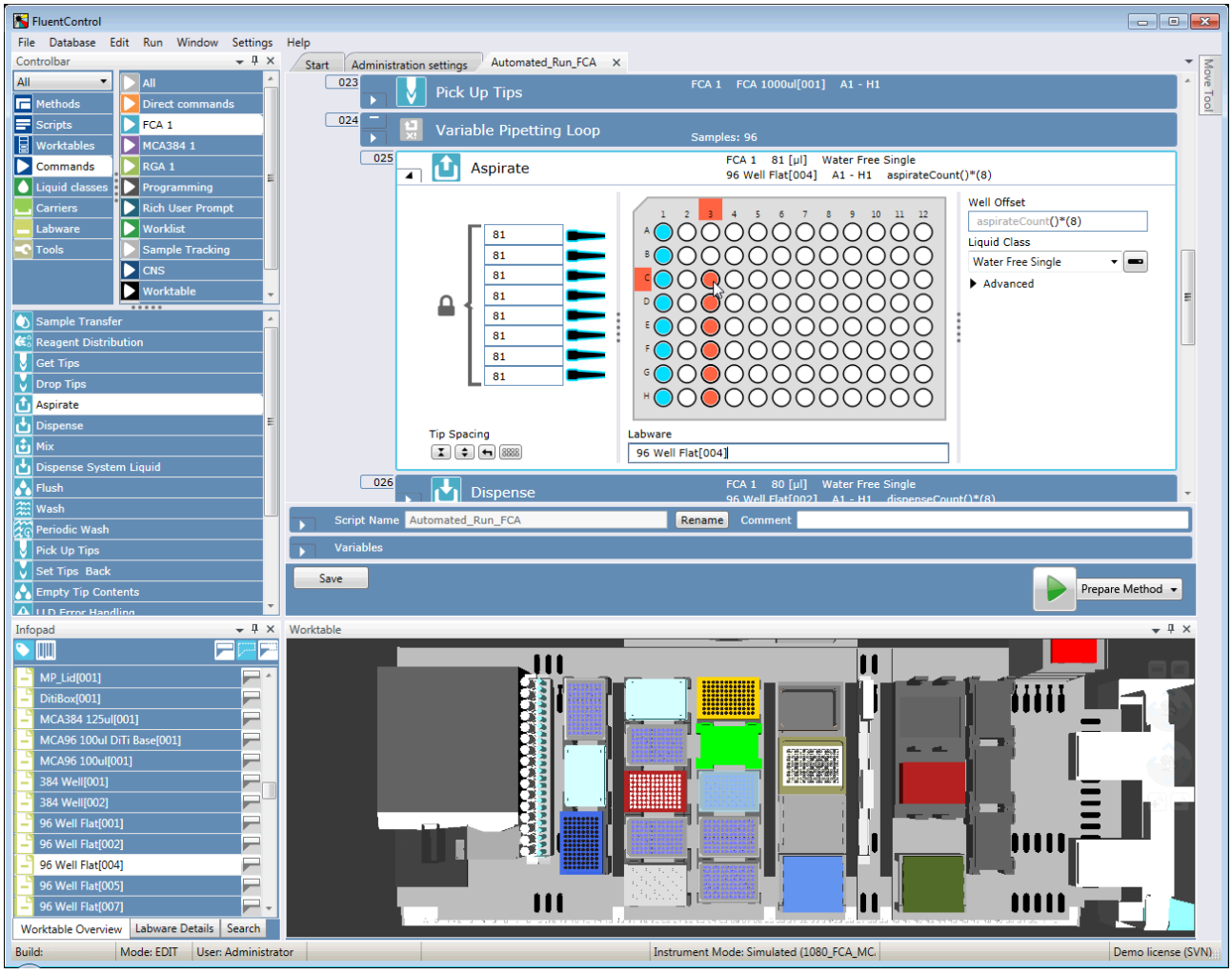
Application Software Manual

FluentControl™



Title:	FluentControl™ Application Software Manual			Part number:	30133329.20
ID:	399935, en, Version 2.9			Translated from:	n.a.
Version:	**Revision:**	**Issue:**	**Document History:**		
1	0	2017-08-15	First edition		
1	1	2017-08-30	Section "Intended Use" updated		
1	2	2017-11-07	Section "Intended Use" updated		
1	3	2018-01-31	Section "Intended Use" updated		
1	4	2018-07-16	6. User Administration 11.3 FCA Gripper		
1	5	2018-10-24	15 API 11.7.25 Switch UV Light On 11.7.26 Switch UV Light Off		
1	6	2019-05-13	Sections 7.6.3, 11.7.24–26, 14.1, 17.1 added Sections 11.2.16, 11.8, 12.2 updated		
1	7	2019-10-23	Sections 1.1, 7.2.2, 8.6.2, 8.6.8, 11.1.16, 11.2.2–3, 11.7.12, 11.5.2, 11.9.1, 12.2 updated Sections 11.16, 15.3.4 added		
1	8	2020-03-06	Sections 4.1, 4.2, 6.10.2, 8.4.2, 8.5.3, 11.2.2, 11.2.13, 11.5.1–4, 11.11, 12.2, 12.4, and 17.2 updated		
1	9	2020-07-15	Sections 2.4, 2.7, 5, 8.6.8, and 11.2.1.2 updated Sections 11.2.20, 11.8.30–11.8.33, and 11.7 added		
1	10	2020-11-05	Sections 5.5, 5.8, 7.4, 8.4.1, 11.8.3, 12.4 updated Sections 5.5.3.9, 11.7, 11.8, 11.9.34, 11.10.4 added		
2	0	2021-05-04	Sections 4.1, 5.5.3.8, 7.4, 8, 10, and 11 updated Sections 7.6.4, 7.6.5, 11.2.20, 11.8.4, and 16 added		
2	1	2021-10-08	Sections 4.3, 5.4, 7.2.2, 18.1.1 updated Section 11.9 added		
2	2	2022-04-13	Sections 4.1, 4.2, 6.7, 7.3, 8.4, 11.6.3, 11.9, 11.11.10, 11.11.17, 11.11.36, 11.11.38, 12.2, 16.1, 18.1.1 and 18.1.2 updated		

© 2025, Tecan Trading AG, Switzerland, all rights reserved.

Information contained in this document is subject to change without notice.

2	3	2022-09-20	Sections 4.1, 5.1, 5.3, 5.5.6.10, 8.3.9, 8.6.5.3, 8.6.5.4, 8.6.5.5, 8.6.5.6, 8.6.5.7, 8.6.5.8, 8.6.5.9, 11.1.5, 11.1.6, 11.1.8, 11.1.9, 11.1.10, 11.1.11, 11.2.17, 11.2.18, 11.6, 11.2.6, 11.7 and 15.2 updated
2	4	2023-03-29	Sections 8.6.5.1, 8.6.5.3, 8.6.5.7.2, 11.6.1, 11.6.4, 11.6.5, 11.1.22 and 16.1.5 updated
2	5	2023-10-05	Sections 8.1, 8.3.9, 8.6.7.4, 16.1.5 and 18.1.1 updated, Section 11.8 newly added
2	6	2024-04-19	Sections 1.2, 8.3.9, 8.6.7.8, 11.2.6, 11.2.6.1, 11.7, 13.4, and 13.4.1 updated Sections 1.2.1, 1.2.2 and 11.5 added
2	7	2024-11-11	Sections 4.1, 11.19, 11.9, 18 updated Sections 7.6.10, 8.6.5.4, 17.1 added Section 5.2 removed
2	8	2025-03-18	Sections 5.2, 6, 6.1, 6.1.2, 6.2, 6.3, 6.4, 6.5, 6.7, 6.10, 7, 7.6, 7.6.5, 8.2.5, 8.6.7.9, 11.2.5, 11.2.7, 11.5, 11.5.1, 11.15.20, 11.15.25, 11.16.2, 11.20.2, 13.3, 13.4, 19.1 updated Sections 1.5.1, 6.1.1, 7.6.7, 7.6.8 added
2	9	2025-10-28	Sections 4.1, 4.2, 4.3, 5.11, 7.2.1, 7.2.2, 7.6.5.2, 8.3.9, 8.6.5, 8.6.7.9, 11.2.1, 11.2.2, 11.2.4, 11.2.5, 11.2.5.1, 11.2.5.2, 11.2.7, 11.3.4, 11.3.5, 11.4.4, 11.4.6, 11.4.7, 11.5.4, 11.6.1, 11.6.2, 11.8.5, 11.9.3.1, 11.13.5, 11.15.11, 11.15.12, 11.15.23, 11.15.30, 11.15.31, 11.15.33, 11.25, 12.2, 13.4, 15.3.6, 16.11.1, 17.1, 18.1.3 updated Sections 5.11, 11.1.15, 11.1.16, 11.3.5, 11.4.6, 11.4.7, 11.15.36, 11.15.37, 12.5, 14.6, 15.3.6, 17.3 added Sections 15.3.6, 15.3.7, 15.3.8, 15.3.9 removed

© 2025, Tecan Trading AG, Switzerland, all rights reserved.

Information contained in this document is subject to change without notice.

Table of Contents

1 About This Manual 10
1.1 Scope of This Manual 10
1.2 Intended Use 10
1.3 Improper Use 10
1.4 Trademarks 11
1.5 Reference Documents 11
1.6 License Information 11
1.7 Camera Privacy Statement 12
1.8 Document Conventions 13

2 Safety 14
2.1 Safety Message Conventions 14
2.2 General Safety Information 16
2.3 Operating Company 17
2.4 Method and Process Validation 18
2.5 User Qualification 18
2.6 Safety Elements 19
2.7 Laser Radiation 24
2.8 Optical Radiation (UVC) 25
2.9 Decontamination Declaration 26

3 Description 27
3.1 Overview 27
3.2 Supported Instruments and Options 27

4 Installation 28
4.1 Computer Requirements 28
4.2 Installing Application Software 29
4.3 Activate License 35

5 Getting Started 39

5.1 Starting the Software ... 39
5.2 Logging In ... 39
5.3 Simulation Mode ... 40
5.4 The Configuration Editor ... 44
5.5 TouchTools ... 64
5.6 Scripts ... 66
5.7 Modules ... 79
5.8 PathFinder ... 79
5.9 Lid Handling ... 84
5.10 Regrip ... 85
5.11 Runtime ... 87

6 User Administration ... 90
6.1 Main Features ... 90
6.2 Login Settings ... 92
6.3 Password Settings ... 93
6.4 Notify Settings ... 95
6.5 User – Role permission ... 96
6.6 Role – Method permission ... 98
6.7 Roles and Access Rights ... 98
6.8 Electronic Signature ... 102
6.9 Deactivate User Management ... 103
6.10 Installing and Configuring Tecan IAM ... 103

7 Menus ... 115
7.1 File ... 115
7.2 Database ... 118
7.3 Edit ... 122
7.4 Run ... 123
7.5 Window ... 124
7.6 Settings ... 125
7.7 Help ... 130

8 Editors 132
8.1 Worktable Editor 132
8.2 Carrier Editor 141
8.3 Labware Editor 152
8.4 Script Editor 163
8.5 Method Editor 165
8.6 Liquid Class Editor 168

9 Specifying Your Instrument Setup 198
9.1 Configuring Your Drivers 199

10 Validation Checklist 200

11 Commands 205
11.1 Direct Commands 205
11.2 FCA 226
11.3 FCA Gripper 272
11.4 RGA 277
11.5 PreciseFlex 400 283
11.6 MCA384 290
11.7 MCA96 305
11.8 MCA96 Gripper 319
11.9 Resolvex i300 323
11.10 Fluent ID 341
11.11 FRIDA Reader 349
11.12 Tube Rotator 356
11.13 DeckCheck 359
11.14 Magellan 365
11.15 Programming 367
11.16 Touch Tools 386
11.17 Worklist 404
11.18 Sample Tracking 418

11.19 Introspect 423
11.20 Worktable 424
11.21 Te-Shake Shaker 427
11.22 Heated Incubator 429
11.23 Te-VacS 431
11.24 Communication 434
11.25 Te-Chrom 437
11.26 Capture Worktable Command 440

12 Functions and Operators 442
12.1 Operators Supported by Expression Solver 442
12.2 Functions 444
12.3 Operands 454
12.4 Worktable Lookup Functions 455
12.5 Internal Variables 460

13 Data Handling 462
13.1 Saving User Data 462
13.2 Backup 462
13.3 Restoring 463
13.4 Archiving Log and Audit Trail Data 463

14 Error Handling and Recovery 467
14.1 Method Recovery 467
14.2 Errors Requiring User Interaction (FCA) 473
14.3 Error Requiring User Interaction (RGA) 476
14.4 Error Requiring User Interaction (Fluent ID) 478
14.5 Detection Sequence 479
14.6 Error Requiring User Interaction (i300) 480

15 API 481
15.1 Execution Channels 481
15.2 Functions 482

15.3 Examples 485

16 Scheduler 491

16.1 The Process Editor 491

16.2 Define Methods Including Processes 501

16.3 Running a Process 503

16.4 User Rights 508

16.5 Scheduler Configuration 508

16.6 The Labware Group Editor 509

16.7 Devices 510

16.8 Automated Tip Handling 510

16.9 Validating a Process 512

16.10 Optimizing a Process 512

16.11 Support for Cell-Based Assays 513

16.12 Not Supported 515

17 Utilities 517

17.1 Log Viewer 517

17.2 OLEG Log Viewer 518

17.3 Command Line Parameters 518

18 Customer Support 519

18.1 Data Collection for Customer Support 519

18.2 Contacting Tecan 531

19 Abbrevations 532

19.1 Abbreviations Table 532

Glossary 534

1 About This Manual

This Application Software Manual describes the FluentControl software, provides information required for its proper installation, and instructs how to run the software and how to make use of all its features.

Before installing and running the FluentControl software, read this manual carefully —in particular section “Safety” [▶ 14].

1.1 Scope of This Manual

This manual applies to FluentControl Application Software.

1.2 Intended Use

1.2.1 General Purpose

The Fluent is a fully automated laboratory liquid-handling platform for general laboratory use. It is intended for routine laboratory tasks, such as pipetting, liquid handling, and robotic manipulation of labware in certain test procedures.

The Fluent Gx Assurance Software is required for this intended use.

Certain options from Tecan and third-party devices that can be used with Fluent are for research use only (RUO).

If a research-use-only option or device is integrated with Fluent or if the Fluent Gx Assurance Software is not installed, the intended use changes to: The Fluent is a fully automated laboratory liquid-handling platform for research and industrial applications. It is intended for routine laboratory tasks, such as pipetting, liquid handling, and robotic manipulation of labware in certain test procedures (e.g., cell-based assays, biochemical assays and compound management). This instrument is not intended for use in clinical diagnostics procedures.

1.2.2 In Vitro Diagnostics

An IVD Fluent instrument is required for this intended use.

The Fluent is an automated laboratory liquid handling platform for in vitro diagnostic use. The product is intended for the automation of clinical sample preparation and processing of clinical diagnostic assays using human specimens. The type of specimen and specific diagnostic protocol is defined and validated by the user for their selected clinical assay. The product is intended for professional laboratory use by trained personnel. The product is not intended for self-testing or near patient testing.

If a research-use-only (RUO) option or device is integrated with Fluent, the intended use changes to For Research Use Only. Not for use in diagnostic procedures.

1.3 Improper Use

Improper use may prejudice the Fluent safety concept.

- The Fluent must not be used with options or components that are not approved by Tecan.
- The Fluent is not explosion-proof and should not be installed in locations where there is a hazard of explosion.

- The Fluent should not be used in the absence of functional safety devices.

1.4 Trademarks

The product names, whether registered or unregistered trademarks, mentioned in this manual are reproduced solely for identification purposes and remain the exclusive property of their respective owners. For simplicity reasons, the trademark symbols such as ® and ™ are not repeated in the manual.

1.5 Reference Documents

This section provides a list of the documents that are needed or may be useful when using the Fluent.

The Doc IDs listed below are root numbers. Therefore, they do not contain information about the language, document version, or the medium (data storage medium, hard copy, downloadable file, etc.) of the document.



On the basis of your order configuration, the Operating Manuals for optional equipment apply as well.

Check the scope of the corresponding document to ensure that you are in possession of the correct version.

1.5.1 Instrument Manuals

- Fluent® Operating Manual (Doc ID 399706)
- Fluent® Dx Operating Manual (Doc ID 403096)
- Fluent® Reference Manual (Doc ID 399937)
- Fluent® Dx Reference Manual (Doc ID 403190)

1.5.2 Software Manuals

- Tecan Sample Tracking Add-on Software Manual (Doc ID 393933)
- For Tecan digital solutions, such as Introspect, see https://www.tecan.com/knowledge-portal/digital-solutions

1.5.3 Other Reference Documents

- Fluent® Carousel Operating Manual (Doc ID 398350)
- HEPA hood (Doc ID Caron 70072)
- Frida Reader™ Application Manual (Doc ID 401882)
- Te-Shake™ Operating Manual (Doc ID 391496)
- Te-VacS™ Operating Manual (Doc ID 391236)
- Fluent® Stacker Operating Manual (Doc ID 398658)
- MIO2 Operating Manual (Doc ID 394934)
- Resolvex i300 Operating Manual (Doc ID 402756)

1.6 License Information

Use the **About FluentControl** command on the **Help** menu for detailed information about licenses used in FluentControl.



Fig. 1: License FluentControl

Product includes software components of third parties whose source code is located in the subfolder C:\Program Files (x86)\Tecan\Third Party Source Code. For license overview please press **F2**.

1.7 Camera Privacy Statement

The Fluent system is equipped with cameras mounted on the inside front profile. The cameras are focused on the deck and rear deck. Views downwards through the acrylic-glass side panels are possible.

- The user is responsible for advising people in the room that cameras are in operation.
- The user is responsible for ensuring that the personnel could not be identified from pictures taken, for example if the instrument is adjacent (side-on) to a desk space or if rear or side panel cut-outs are made or if an acrylic-glass panel is used to replace the rear wall.

1.8 Document Conventions

Cross-References

Cross-references appear as follows—e.g.:

Refer to section “Safety” [▶ 14]

- “Safety” refers to the corresponding section header
- The page number is given in square brackets

Prerequisites

Prerequisites appear as follows—e.g.:

✓ “General Safety Information” has been read.

Tips

Additional tips appear as follows—e.g.:



For safety conventions and symbols refer to chapter “Safety” [▶ 14].

Illustrations

The illustrations may show component versions which are not relevant to your Fluent.

2 Safety

This chapter describes the safety concept of Fluent, provides general rules of correct behavior, and warnings concerning hazards associated with the use of the Fluent.

2.1 Safety Message Conventions

2.1.1 Signal Words

Tab. 1: Signal Words

Signal Word	Meaning
⚠ DANGER	Indicates a hazardous situation that, if not avoided, will result in death or serious injury.
⚠ WARNING	Indicates a hazardous situation that, if not avoided, could result in death or serious injury.
⚠ CAUTION	Indicates a hazardous situation that, if not avoided, could result in minor or moderate injury.
NOTICE	Indicates a situation that is not hazard-related but, if not avoided, could result in damage to or malfunctioning of the equipment, or incorrect process results.

2.1.2 Safety Symbols



Crushing of hands



General warning



Laser beam



Optical radiation



Biohazard



No heavy load



Magnetic Field

2.2 General Safety Information

⚠ WARNING

Fluent is designed and built in accordance with the present state-of-the-art technology and the recognized technical safety regulations. Nevertheless, risks to users, property and the environment can arise if the Fluent is used without due care and attention.

The safety of all users and personnel depends on the strict observation of these safety instructions and awareness of the safety-related warnings provided in this manual.

- Please pay great attention to the following general safety information.
- This manual must always be available to all persons performing the tasks described herein.
- Always use the power cable supplied with the instrument.
- Do not use the power cable with other products.

- Legal regulations, such as local, state and federal laws concerning the use or application, as well as the handling, of dangerous materials in connection with the Fluent must be strictly followed.
- The operating company is responsible for defining instructions in accordance with company procedures and local legal requirements. The instructions provided by the operating company must be strictly observed.
- Observe the correct environmental conditions for storage and operation.
- Structural changes to the safety devices are forbidden.
- Damaged safety devices must be replaced immediately as described in this manual.
- The Fluent must not be modified in any way without prior consultation and written approval of Tecan. Authorized modifications to the system may only be performed by an FSE certified for the repair and upgrading of the Fluent. Tecan will reject any claim resulting from unauthorized modifications.
- Fire hazard caused by the improper use of the Fluent. The Fluent should not be installed in locations where there is a hazard of explosion.
- Fire hazard caused by flammable liquids or system liquid.
- Avoid the formation and accumulation of flammable vapors.
- Chemical, biological, and radioactive hazards can be associated with the substances used or the samples and reagents processed with the Fluent (e.g., during loading and unloading). The same applies to waste disposal.
  - Always be aware of possible hazards associated with these substances.
  - Use appropriate protective clothing, safety goggles, respirators, and gloves.
  - The handling of substances and the disposal of waste may be subject to local, state, or federal law, or to regulations with regard to health, environment, or safety. Strictly observe the corresponding provisions.
- Any contamination must be dealt with immediately as described in this manual.
- The user is responsible for ensuring that the Fluent is always operated under proper conditions, and that maintenance, service, and repair tasks are performed with care, on schedule, and only by authorized personnel.

- Risk of incorrect measuring results. After system care or maintenance has been performed, operation must only be resumed after the correct system operating conditions have been verified.
- Always use recommended consumables within expiration date and original spare parts for maintenance and repair to assure good system performance and reliability.
- Injury could result if skin comes in contact with system liquid on the instrument.
  - Always wear protective clothing according to GLP.
- Heavy load! Do not lift the instrument.
- Do not operate the system without deck trays and deck segments.
- Deck trays capture liquid spills that may occur in the manual deck loading area. The system should be operated with as many deck trays as possible installed below the deck to collect all liquid spills. Do not operate the system without deck trays.
- If carry-over is not tolerated, the use of disposable tips with filters is strongly recommended.
- Possible Crash. Do not place devices without Tecan model data on the deck.
- Extension 300 is designed for a maximum load of 40 kg (88 lbs.), and only for use with options that are lighter than 40 kg (88 lbs.).
- The Fluent is supplied with a biohazard safety sign which should be applied by the user in the event of use of biohazardous substances. Apply the label on the front door in a position visible to the user and convenient for the application. Refer to section Product Safety Signs.
- Options used on the worktable of the Fluent can generate strong magnetic fields, which may interfere with the function of medical devices implanted into or worn by an operator, such as pacemakers or insulin pumps. The Fluent is supplied with a safety sign of strong magnetic fields which should be applied by the user to the front door in a position visible to the user and suitable for the application in the event of use of options generating strong magnetic fields.
- The ethernet cable of the DeckCheck cameras will be installed by an FSE and must be installed on the Fluent PC at all times (EMC). The Ethernet interface is not allowed to be connected to a network.
- For California residents only: This product can expose you to lead which is known to the State of California to cause cancer and birth defects or other reproductive harm. For more information go to www.P65Warnings.ca.gov.

2.3 Operating Company

The operating company must ensure that the Fluent and in particular the safety features, function properly and that all the personnel in contact with the instrument are adequately trained.

Responsibilities

- Method and process validation.
- Defining the processes in compliance with the Standard Operating Procedures.
- Ensuring that installation and operational qualifications (IQ OQs) have been completed.
- Ensuring that all personnel in contact with the Fluent are adequately trained.
- Ensuring the availability of appropriate protective clothing and equipment.

- Ensuring the maintenance and safe operation of the Fluent.
- Requiring adherence to laboratory safety regulations and directives.

2.4 Method and Process Validation

While performing method and process validation, pay attention to the following:

- If using fixed tips with MCA 384 or FCA, ensure that the wash procedure is effective for the expected sample concentration range and assay sensitivity.
- Check that pipetted volumes meet the precision and accuracy requirements of the process being automated.
- When using non-Tecan or custom labware and aspirating with tracking, ensure that the container definition is correct (i.e., the appropriate speed is used for tracking) to avoid air aspiration.
- The Phase Separator functionality has been verified for use with standard Tecan 1 ml disposable tips and 1 ml Tecan wide-bore disposable tips. For more information on supported Tecan consumables, refer to the Reference Manual (see Reference Documents).

Responsibility of the Key Operator

- Validate liquid detection on the Fluent Stacker transfer station.
- Validate the correct usage of the MCA wash station by the application.
- Validate the application with regard to correct pipetting volumes and tracking.
- Validate the application to prevent aspiration too close to Z-max of custom labware.
- Validate piercing applications with regard to the downholders needed (active or passive).
- If chemicals and labware are not removed, the impact of UVC light on chemicals and labware present on the deck has to be evaluated and the assay validated.
- Include a manual post-run check for correct pipetting volumes.
- Personnel must be informed regarding the camera privacy statement (refer to “Camera Privacy Statement” [▷ 12]).

2.5 User Qualification

The laboratory personnel must be fully qualified and trained to operate the Fluent. The work described in this Application Software Manual must only be performed by authorized personnel with the qualifications prescribed below.

Laboratory personnel must:

- have suitable technical training,
- be familiar with the laboratory safety regulations and directives,
- be familiar with the instructions for the safety elements of the instrument,
- use protective clothing and equipment,
- be familiar with and adhere to good laboratory practices,
- and have read and understood the instructions in the Operating Manual.

Tecan recommends that the operator attends an operator training course. Please ask the Tecan Customer Service about available courses. Refer to section “Customer Support” [▷ 519].

2.5.1 Operator

The operator (lab technician) works for the operating company.

Required Skills

- No specific application or system knowledge
- Command of local languages
- Command of English is preferable

The operator has application software access rights allowing him to run methods and perform system care and will receive necessary training from the Key Operator.

2.5.2 Key Operator

The key operator (application specialist) supports the operating company or works for the same company.

Required Skills

- Extensive application knowledge
- Limited system knowledge
- Command of local languages
- Command of English
- In-depth knowledge of the corresponding software manual

Responsibilities

- Instructing the operator
- Writing, running and validating methods
- Helping the operator to solve problems with the instrument

2.6 Safety Elements

⚠ CAUTION

Moving parts

The protection and safety elements installed on the Fluent must not be removed, disabled or overridden during operation.

- If any devices are removed (e.g., for maintenance work), all protection and safety devices must be reinstalled, re-enabled and checked before resuming operations.

Safety panels and safety sensors are integral parts of the Fluent, whereas instrument door locks and cabinet door locks can be included only in certain system configurations.

2.6.1 Safety Panels

Fluent is protected with safety panels:

The **front safety panel** can be opened and is fitted with door sensors that trigger an active stop. The front safety panel can be locked with optional door locks.

The Fluent with either MCA 96 or MCA 384 arms may only be used with Full Front Safety Panel.

The door opening is supported by gas filled springs. The operator should open the door fully for safe/ complete access to the instrument.

The **diluter panel** can be opened without affecting Fluent operation (except for Fluents with UVC light option installed – the diluter panel sensor will trigger a fast stop when the panel is opened).

The **top and side safety panels** are fixed.

2.6.1.1 Front Safety Panels

The front safety panel prevents direct access to the robotic arms and to the elements on the instrument deck during operation. This is for the benefit of personal safety and improves method security. In addition, the front safety panel protects the user against spilling sample or reagent. There are different types of front safety panels.

Full Front Safety Panel

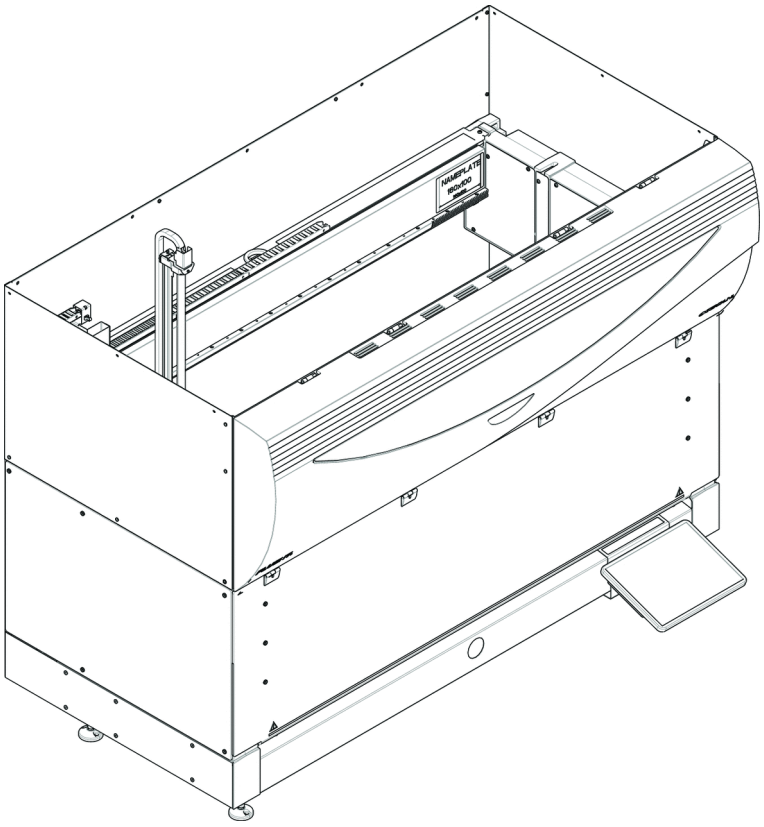


Fig. 2: Full Front Safety Panel

The full front safety panel has the following features:

- No access to moving parts (moving parts, mechanical hazards)
- Protection of the samples against outside influence (method safety)
- Protection against spilling sample or reagent

With full front safety panels, only batch-wise loading is possible.

Full Front Safety Panel (UVC)

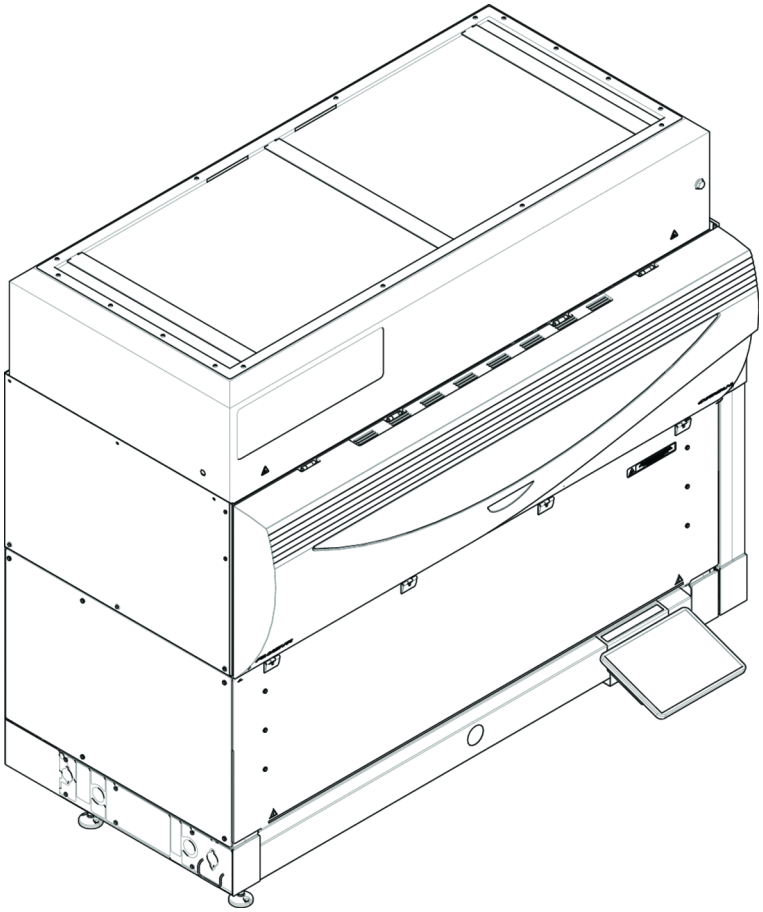


Fig. 3: Full Front Safety Panel (UVC)

The full front safety panel (UVC) has the following features:

- No access to moving parts (moving parts, mechanical hazards)
- Protection of the samples against outside influence (method safety)
- Protection against spilling sample or reagent
- Protection against optical radiation (UVC)

With full front safety panels, only batch-wise loading is possible.

⚠ CAUTION

Moving parts!

Moving MCA, FCA and Air FCA can cause hand injuries when reaching through the half front safety panel or of the front safety panel with expansion into the instrument during a run.

- Do not reach into the instrument during a run.

Half Front Safety Panel

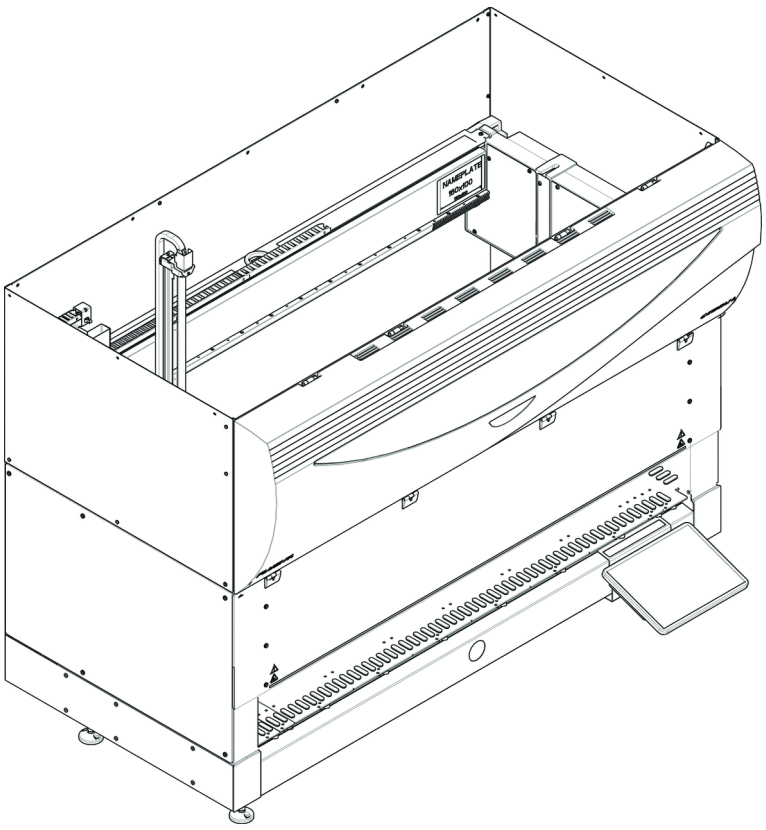


Fig. 4: Half Front Safety Panel

The half front safety panel has the following features:

- Restricted access to moving parts (moving parts, mechanical hazards)
- Protection against spilling sample or reagent

With the half front safety panel the operator has restricted access to the instrument deck. Loading and unloading runners is possible without opening the panel; i.e., the operator is enabled to reload samples or reagents during the method run.

Front Safety Panel with Expansion

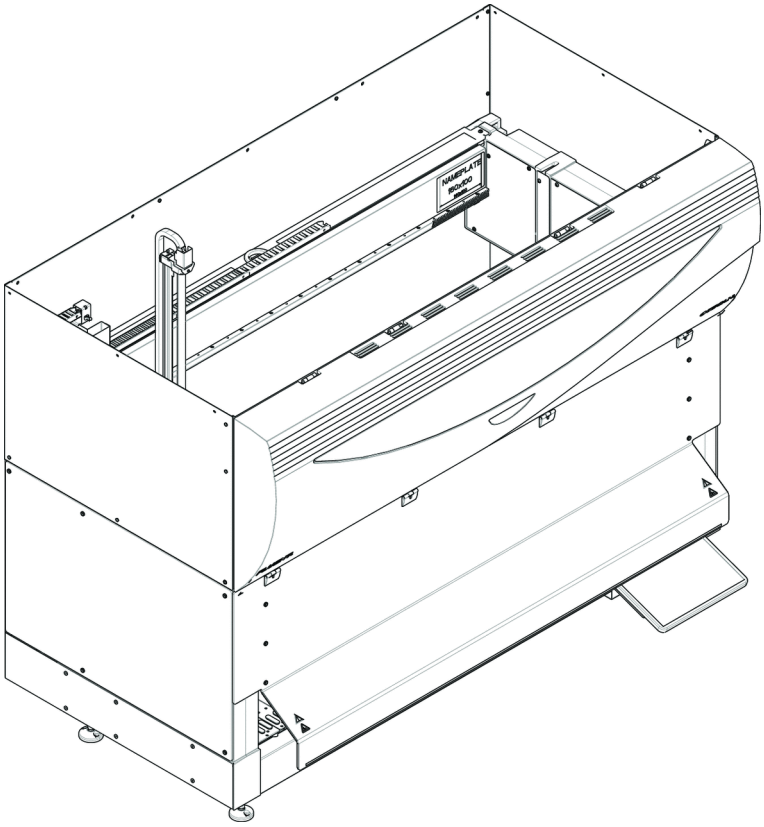


Fig. 5: Front Safety Panel with Expansion

The front safety panel has the following features:

- Restricted access to moving parts (moving parts, mechanical hazards)
- Protection against spilling sample or reagent
- Allows for the use of a front DiTi waste station, which protrudes from the deck and requires a downward-facing opening in the front safety panel.

With the front safety panel with expansion, only batch-wise loading is possible.

2.6.1.2 Safety Panels for Optional Devices

If an optional device is added to, or removed from, the side of the Fluent, an appropriate side safety panel must be installed. Please consult the “Customer Support” [▷ 519].

2.6.2 Instrument Door Locks (optional)

Two optional door locks can prevent the front safety panel from being opened and protect the ongoing process. This prevents unwarranted interruption of the process run. To stop a process, a pause request can be entered by means of the touchscreen.

2.6.3 Cabinet Door Locks

If an RGA long axis has access below the deck, the cabinet door closest to the access point must be equipped with a door lock sensor option. If more than one access point below the deck is implemented or if the access point is changed during the life of the instrument, then each door near the access point must be equipped with a door lock sensor.

If the instrument has a HEPA hood, all cabinet doors must be equipped with a door lock sensor.

2.6.4 External Door Locks

External door locks will be implemented on Fluent installations in an external enclosure. The door panels of the external enclosure replace the mechanical safety function of the Fluent front safety panel and the cabinet doors, and the external door docks with integral sensors replace the door sensor and door lock functions of the Fluent front safety panel and cabinet doors.



External door locks do not allow an ActiveStop. To stop or pause the process a pause request can be entered by means of the touch screen.

2.7 Laser Radiation

Fluent can be equipped with laser barcode scanners. The laser radiation from these barcode scanners is a low-power, collimated beam in the visible spectrum. The laser classes of each barcode scanner, and of the entire Fluent system, are indicated on the laser safety label affixed to the corresponding hardware.

All modules with lasers are marked with the appropriate laser safety labels.

The Fluent instrument has been tested and certified according to IEC 60825-1:2007 and IEC 60825-1:2014.



⚠ CAUTION

The Fluent is a class 1 laser product pursuant to IEC 60825-1:2014 that emits laser radiation.

Dazzle, flash-blindness and afterimages may be caused by the laser beam.

- Do not stare into the laser beam or into its reflections.

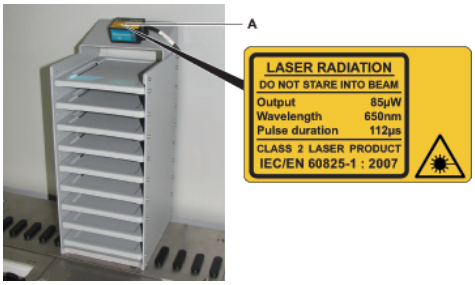
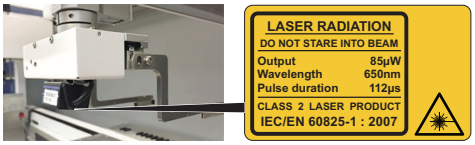
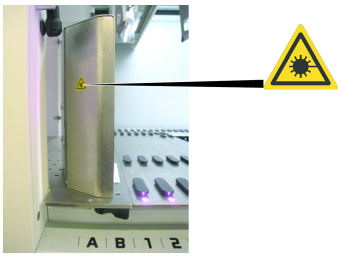
2.7.1 Laser Radiation Devices

A stand-alone barcode scanner can be mounted on a device.

Please ensure that the safety label is correctly affixed to the barcode scanner at all times:

- Explanatory Laser Radiation Label (A): Identifies a CLASS 2 LASER PRODUCT according to IEC 60825-1 that contains an embedded visible low power laser barcode scanner. Instructs the user not to stare into laser beam or its reflection.

- Class 2 lasers are only operated when the system is running and have no interface to the operator.

Label placement	Explanation
	Stand-alone barcode scanner mounted on a hotel: label located underneath the scanner.
	Stand-alone barcode scanner mounted on Robotic Gripper Arm: label located on the scanner.
	Loading ID: label located on the rear side of the scanner housing.
	Loading ID: label located on the side of the scanner housing.

2.8 Optical Radiation (UVC)

The Fluent can be equipped with an optional HEPA hood which includes a UVC light, or a separate UVC light option.

Exposure to UVC light radiation must be avoided as it can lead to injury. The UVC light switches off automatically when the front safety panel is opened, and in the case of the UVC light option also when the diluter cover is opened. Special UVC-resistant safety panels are installed on the Fluent in conjunction with UVC light.

UVC light can be used in decontamination procedures. The suitability and effectiveness of using UVC for individual processes must be validated by the user.

Please also refer to the manual provided by the HEPA hood manufacturer.

2.9 Decontamination Declaration

In addition to regular system care, and in accordance with standard laboratory regulations, the Fluent and its parts and accessories must be thoroughly decontaminated in the following circumstances:

- Before any maintenance or service work is performed on the Fluent and, in particular, before an FSE intervention on the Fluent
- In the event of accidents (e.g., crash, spillage, etc.)
- Before returning the Fluent or its parts or accessories, to Tecan (e.g., for repair)
- Prior to storage
- Prior to disposal
- In general, before moving the Fluent or its parts from its location

The owner of the instrument has full responsibility for the effective decontamination of all the equipment.

Before any intervention on the Fluent by an FSE, and before returning the Fluent or its parts or accessories to Tecan, the owner of the instrument must complete and sign the Decontamination Declaration form, confirming that the decontamination has been performed in accordance with good laboratory practice guidelines. Contact your local service organization to obtain this form and refer to section Decontamination.



Tecan reserves the right to refuse to deal with any Fluent or its parts or accessories that is not accompanied by the Decontamination Declaration form.

3 Description

This chapter provides a brief overview of the FluentControl software and defines its scope of application. In addition, it describes the features and specifies the computer requirements for running the software.

FluentControl allows the user to interface, control and monitor the instrument.

3.1 Overview

FluentControl provides a graphical user interface for controlling Tecan's Fluent series of pipetting instruments.

The FluentControl software allows the user to implement protocols (scripts) and to quickly visualize how a laboratory process or experiment can be implemented. Pipetting is modeled by individual script lines. Commands are provided for a wide range of Tecan hardware options and third-party devices and the scripting system is designed for maximum flexibility when developing complex custom methods.

3.2 Supported Instruments and Options

FluentControl is intended for the following instruments:

- Fluent 480
- Fluent 780
- Fluent 1080

FluentControl in this configuration supports the following optional devices, instrument options and systems:

- Flexible Channel Arm (FCA) with 8 channels and disposable tips, also with FCA gripper fingers
- Multiple Channel Arm 96 (MCA96), also with Finger Exchange System (FES) fingers
- Multiple Channel Arm 384 (MCA384) with 96 or 384 tips
  - Wash & Refill Center
- Robotic Gripper Arm (RGA), also with FES
- Fluent ID barcode scanner
- Fluent Stacker
- MIO2
- Fluent Carousel with or without barcode scanner
- Keyence Laser Bar Code Reader BL-1300 Series
- Mettler Toledo WXS and SAG weighing modules
- Tecan Safety features (door locks and sensors)
- Tecan Access/Status feature (status lamps)
- Tecan Magellan data reduction and analysis software for Tecan microplate readers

4 Installation

This chapter describes the installation of the FluentControl software. Please follow the instructions carefully.

FluentControl must be installed on target computers that are or will be connected to the instrument.

To install FluentControl on the target computer, you need local administrator access rights for the Windows operating system.

4.1 Computer Requirements

To run FluentControl, your computer must meet the following requirements:

Component	Minimum requirements FluentControl without DeckCheck	Minimum requirements FluentControl with DeckCheck
Operating System	Windows 11 (IoT) Enterprise LTSC 2024, or Windows 10 (IoT) Enterprise LTSC 2021*	Windows 11 (IoT) Enterprise LTSC 2024, or Windows 10 (IoT) Enterprise LTSC 2021*
Processor**	Intel® Core™ i7 of the 11th generation or higher, or AMD Ryzen 7	Intel® Core™ i7 11th generation and with UHD, or higher***
RAM	16 GB	32 GB
Hard Drive	Solid-state drive (SSD) with 512 GB free space	Solid-state drive (SSD) with 512 GB free space with high transfer speed (6 GB/s)
USB Ports	1 USB 2.0. Additional ports may be needed for additional devices.	1 USB 2.0. Additional ports may be needed for additional devices
Network Interfaces	GigE Network card with Intel® chipset for external network. • Optional: second GigE Network card with Intel® chipset for local network to any 3rd party devices • Optional: internet connectivity for remote support and Introspect.	Two GigE Network cards with Intel® chipset for: • External network • Local network to any 3rd party devices and DeckCheck Optional: internet connectivity for remote support and Introspect.
Resolution of Monitor(s)	1920 x 1080	1920 x 1080

* Windows 10 (IoT) Enterprise LTSC2021 does not support all Intel CPU models and generations. For supported CPUs, please refer to https://learn.microsoft.com/en-us/windows-hardware/design/minimum/supported/windows-10-ltsc-2021-supported-intel-processors.

** It is recommended to use an Intel® Core™ i9 11th generation or higher for FluentControl Scheduler applications.

*** Do not use a processor without integrated graphics for a DeckCheck configuration. Do not use a separate graphics card for a DeckCheck configuration, except the NVIDIA RTX 2060, NVIDIA RTX 2080 or the NVIDIA PP2200 Quadro.

NOTICE

Risk of the script or process aborting.

Risk of loosing connection to the PC After a certain time of inactivity from the user, the PC goes into a sleep mode and the connection to the instrument may be lost. Adjust the PC settings to avoid from going into the sleep mode.

- The PC must be configured as to prevent it from going into **Sleep Mode**.
- The USB power management in Windows must be switched off to avoid losing the connection to the pipetting instrument.

Connect the pipetting instrument and other hardware options to the USB 2.0 port of the PC.

The use of an external USB hub should be avoided.

A passive external USB hub may generate data transfer interruptions and loss of data.

- If an external USB hub is absolutely necessary, use an active USB hub (i.e., with its own power supply).
- Please note that some port replicators for laptops contain a USB hub.

4.2 Installing Application Software

With the Tecan FluentControl Preinstaller, you install the necessary components to use FluentControl on your computer.

The Magellan driver requires the Tecan Magellan software, which must be installed separately.

Before installing FluentControl on your PC, make sure it meets the PC requirements..

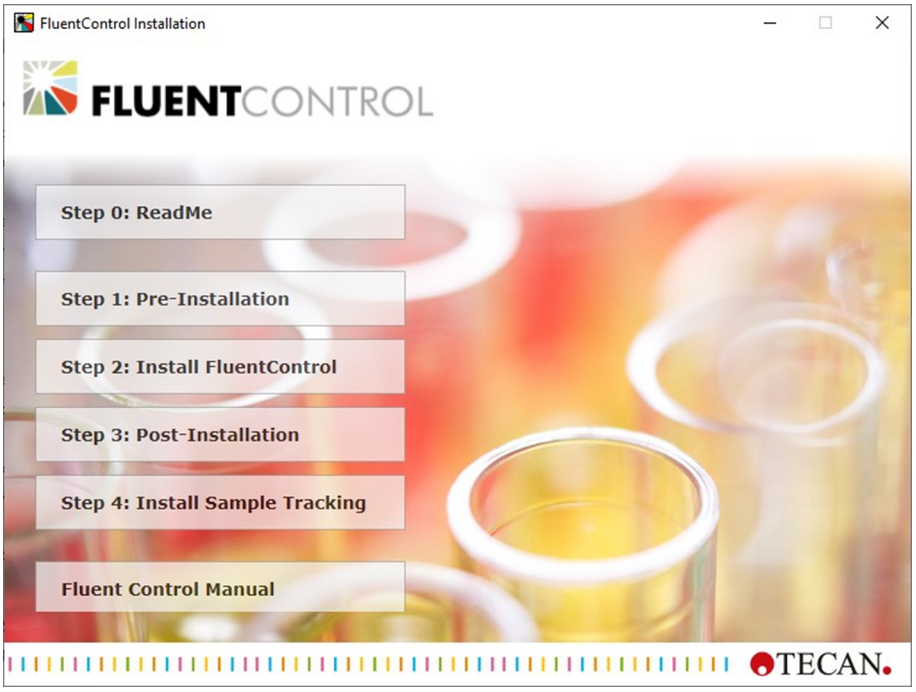


Fig. 6: FluentControl AutoPlay Application

The installation consists of following steps:

- ReadMe
- Pre-Installation
- Install FluentControl
- Post-Installation
- Install Sample Tracking

To install the program, proceed as follows:

1. Click **Step 0: The ReadMe**
   The ReadMe document opens. It contains information about how to upgrade the FluentControl software, remaining issues and new features.
2. Run the autorun application (autorun.exe) as administrator.
3. Click **Step 1 PreInstallation**.
   The Tecan FluentControl Preinstaller appears:

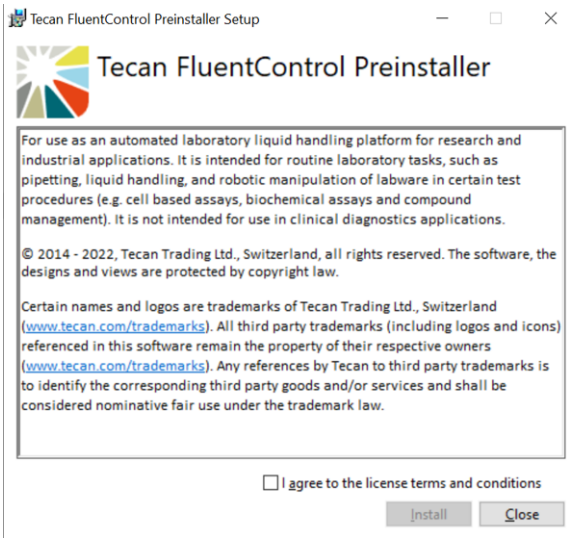


Fig. 7: Tecan FluentControl Preinstaller Setup

4. Acknowledge the license terms and conditions, then click **Install**.
5. In the **User Account Control** dialog box, click **Yes** to allow the Tecan FluentControl Preinstaller Setup to make changes to your computer.

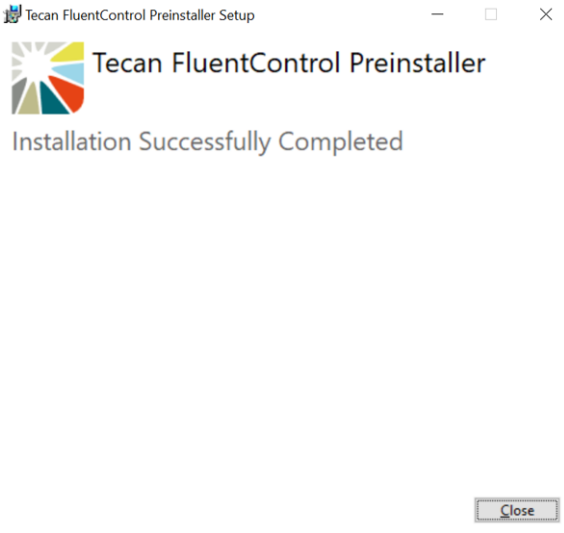


Fig. 8: Tecan FluentControl Preinstaller: Setup Successful

6. After you reach this screen, click **Close**.
7. Click **Step 2: Install FluentControl**.

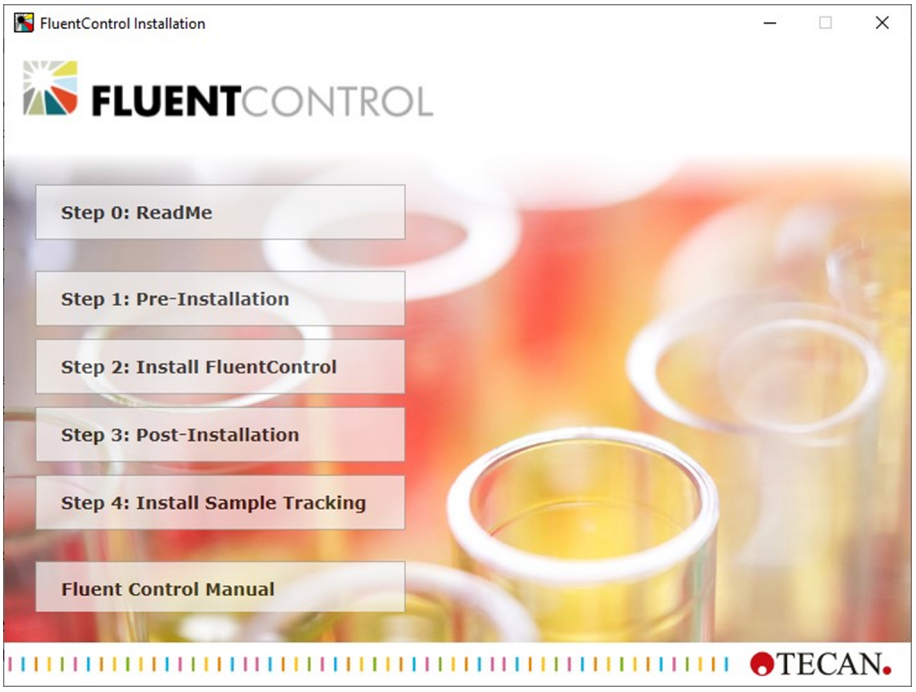


Fig. 9: The Tecan FluentControl Setup Wizard

During the execution of this installation step, an installation log file is generated in the temp folder of the current Windows user. The name of the log file follows the naming pattern 'Fluent_Install_%YEAR%%MONTH%%DAY%__%HOUR%-%MINUTE%-%SECONDS%.log' (e.g., Fluent_Install_2025_09_08__15-30-20.log).

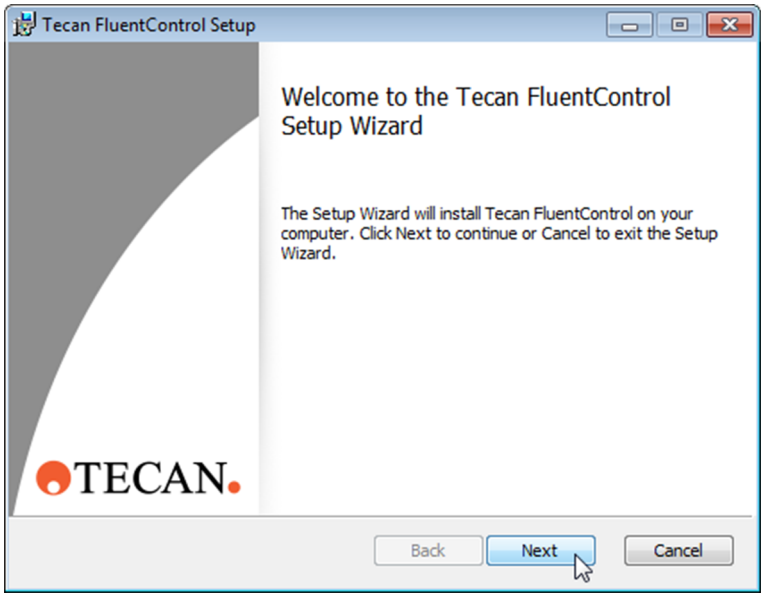


Fig. 10: FluentControl Setup

8. Click **Next** in the Tecan FluentControl Setup screen.

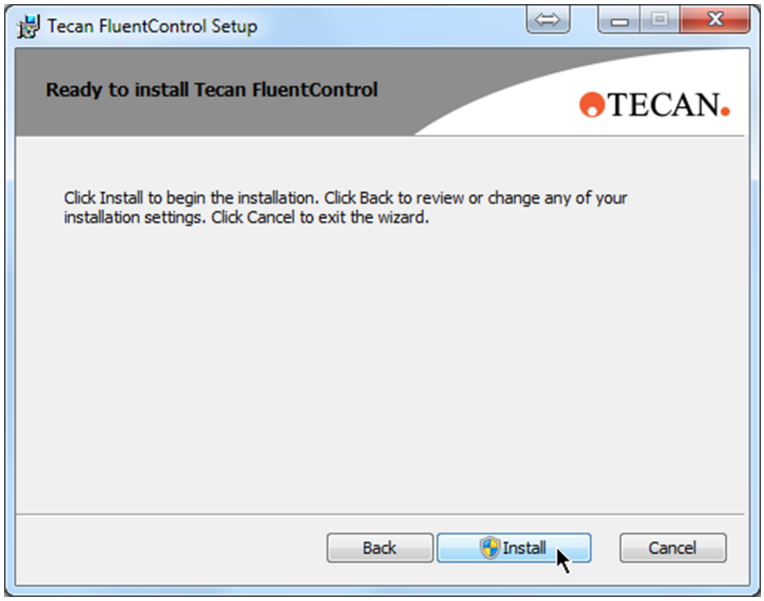


Fig. 11: Tecan FluentControl Setup: Ready to install

9. Click **Install**. The installation procedure may take some time.

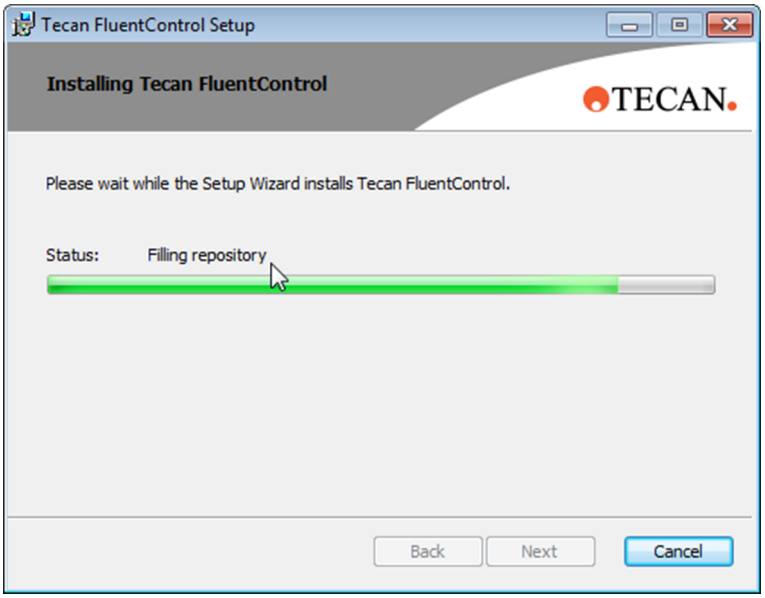


Fig. 12: Installation FluentControl

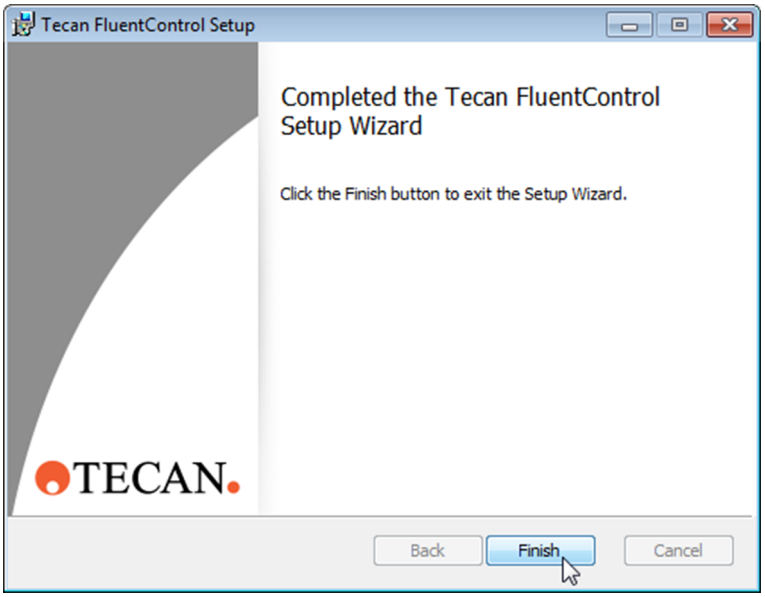
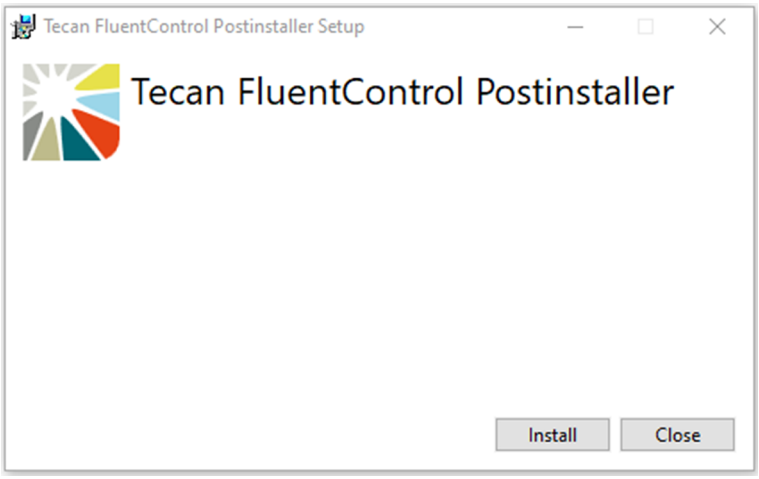


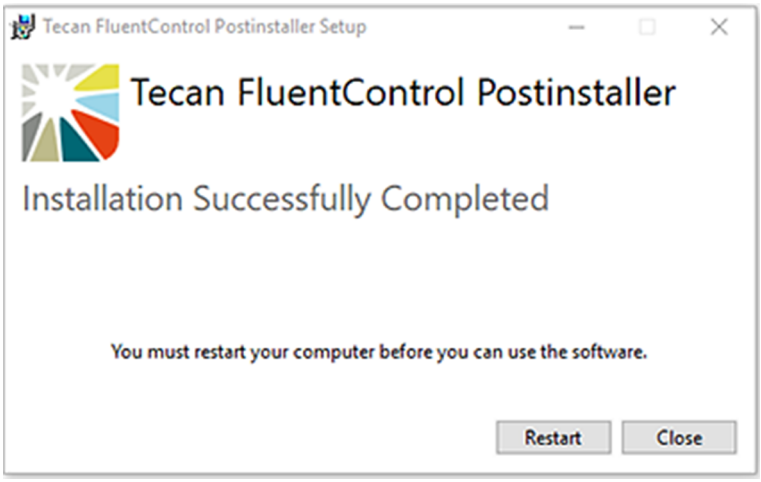
Fig. 13: Installation completed

10. Click **Finish** after the installation is finished. The setup wizard screen closes.

11. Click **Step 3**: The Tecan FluentControl Postinstaller appears:



12. Click **Install**.
13. After the installation is completed, restart the instrument and the connected PC.



14. Optionally, click **Step 4 Install Sample Tracking** and follow the instructions in the Tecan Sample Tracking Add-on Software Manual on the installation medium (refer to section Reference Documents) to install the embedded configuration of the Sample Tracking software.
    If sample tracking is not needed, it is not necessary to run step 4.
15. After setup is complete, restart your computer, start the program as described in **Starting FluentControl**, and then, before any other actions, select the instrument's configuration as described in "Simulation Mode" [▶ 40].

4.3 Activate License

Each instrument is registered on the Tecan Licensing Server with its serial number and the license types that were sold in combination with that instrument. The following two license types are available for FluentControl: The Base license, Gx license, IVD license, Fluent Scheduler license, Pressure Monitored Pipetting AI license and the Phase Separator license. If no existing license is available, you may request your license by typing in your personal data and sending a request to

Tecan. The license for FluentControl is not provided by a dongle or other hardware. The license must be activated on the local PC via the Licensing Client Module. Licenses can be activated online or offline.

Online Activation

1. Start FluentControl and make sure you are connected to an instrument.
2. In the menu **Settings**, click **Tecan Licensing Client**.
3. The **Licensing Client Module** appears:

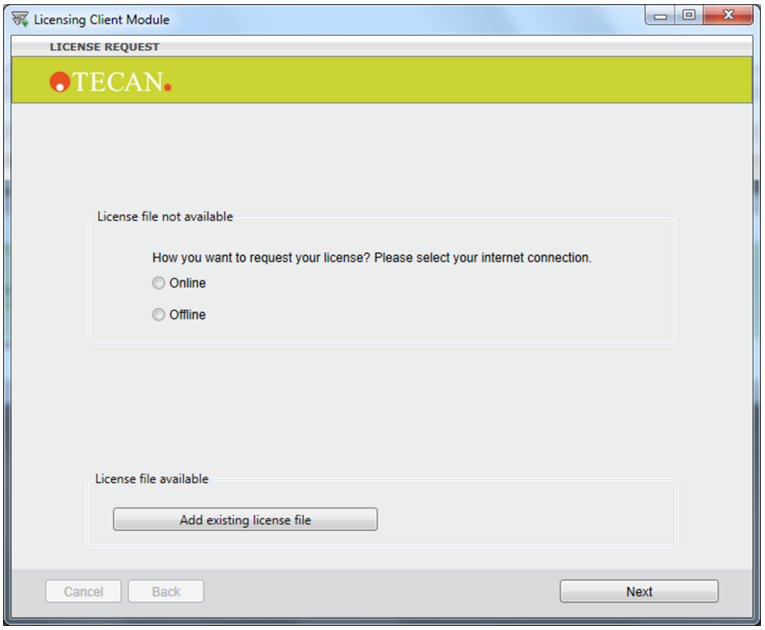


Fig. 14: Add license file

4. Select **Online** and then click **Next**.
5. Leave the **Activation Key** field empty and click **Next**.
6. Fill in the form window accordingly and click **Send Request**. A message appears will appear confirming the successful activation.
7. Click **OK** and restart the FluentControl. The instrument is ready for work.

Offline Activation

1. In the menu **Settings**, click **Tecan Licensing Client**.
2. Select **Offline** and click **Next**.
3. Choose from the following two modes:
   File Mode: A request file will be generated and this file will be uploaded to the webpage.
   Key Mode: A sequence of characters will be generated and displayed for manual entry on the web page.

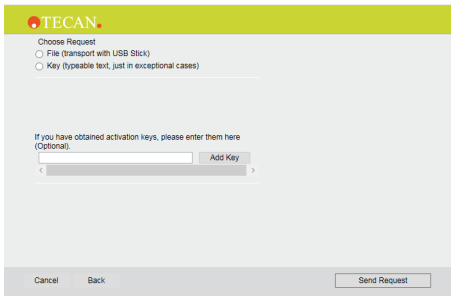


Fig. 15: Choose Request

4. Leave the **Activation Key** field empty and click the **Send Request** button. A window with the path to your license request file or your request key will appear.

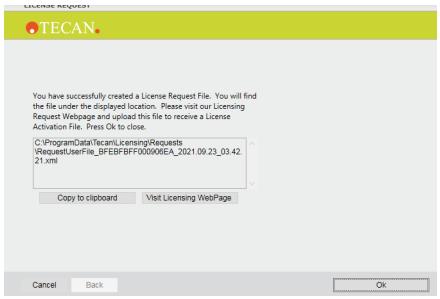


Fig. 16: Licensing Key Path

5. Copy the license request file to an USB storage device.
6. Go to another computer that has internet access to complete the next steps.
7. Open the licensing webpage, which can be accessed by copying the URL below. The URL is also available by clicking the “Visit Licensing Webpage” button in the illustration above.
   https://license.tecan.com/lsprovider/licensingrequest.do
8. Upload and submit your license request file, or click **Manual Request** to manually enter your request key.
9. A license data file specific for your instrument will be available for download. Copy this file to a USB storage device.
10. Go back to the instrument computer with the copy of your license file.
11. Open the Licensing Client Module and click the **Add existing licensing file** button
12. Select the downloaded license file and click the **Add file** button
13. Restart the FluentControl application. You can now work with the new, valid license.

Note: After license activation, the current license type is displayed in the bottom left corner of the FluentControl main window: Simulation mode does not require a license. In simulation mode, the “Demo License” type is always displayed in the bottom right corner of the software. The license file is linked to the serial number of

a specific instrument. It is possible to download the license file multiple times. When replacing the instrument computer, create a new license request and download the license file for the new computer.

5 Getting Started

FluentControl is an application software, which allows the user to program and run fully automated laboratory liquid-handling processes, pipetting and robotic processes on the Fluent automation workstation.

The user can simply drag and drop commands by using the 3D editor to create protocols to configure their worktable elements and labware. Finished protocols can be run first in a 3D simulator, without wasting valuables reagents.

5.1 Starting the Software

- The instrument powers on when the application software is started and the connection is established. When not in use the instrument goes in standby mode.
- The system has to be initialized the first time you power it on. Note: To avoid errors first open a worktable in the SW.
- When FluentControl is started two separate windows will be shown, the main Software interface and the TouchTools interface. The integrated touchscreen monitor is an extended monitor; you can dynamically switch and operate the System from either TouchTools or from the main SW window.
- A SW license is required to run the system in real mode. Simulation and script editing as well as System initialization do not require a license. The license must be activated on the local PC via the Licensing Client Module (see Licensing).
- Press F1 to open the Online Help.

1. Start Microsoft Windows.
2. Click, Start > All Programs > Tecan >FluentControl.

   A startup screen appears during startup.

5.2 Logging In



Contact your FluentControl administrator if you do not have a user name and password.

1. If User Administration is activated, you must enter your password during log in. For more information on User Administration refer to “User Administration” [▷ 90].

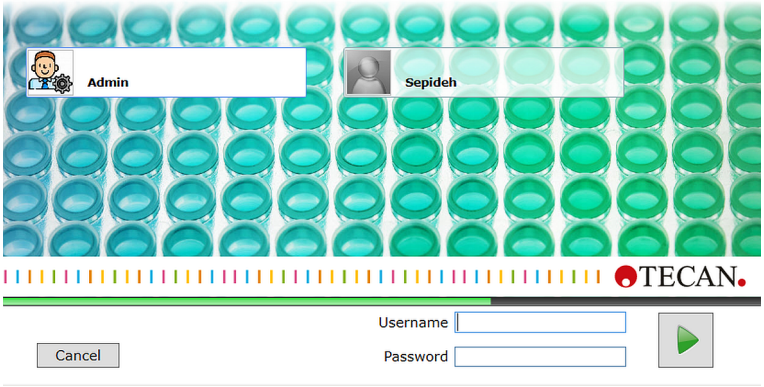


Fig. 17: FluentControl Startup Window

2. Click your user name icon.
3. Type your password and press **Enter** or click the large green arrow on the right.

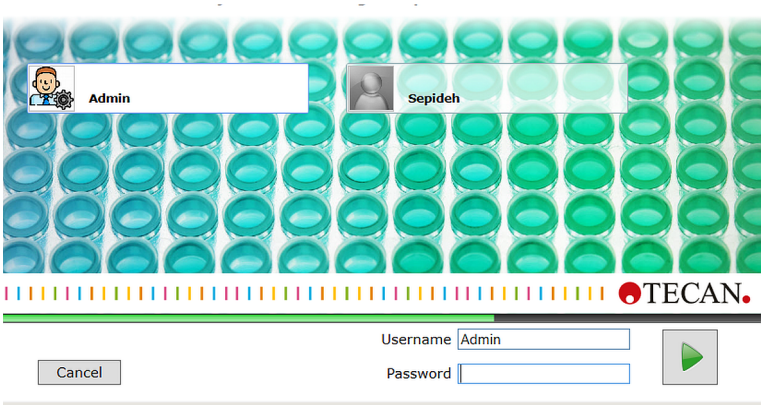


Fig. 18: Password for “Administrator”

5.3 Simulation Mode

You can simulate any Fluent instrument configuration even if no real instrument is connected to the FluentControl software. You can switch between a real instrument and a simulated instrument. You also can install FluentControl on an other PC than the PC connected to the instrument. Whenever FluentControl is connected to a Fluent instrument in real mode, the software automatically downloads and applies the instrument configuration file that is stored on the instrument. This instrument configuration file contains data such as instrument size, arm order, tip type, and arm hardware (e.g., disposable tip adapters, fixed tips, syringe size, RGA finger type, etc.). This process ensures that FluentControl applies the correct hardware configuration in real mode.

For script development, we recommend to use simulation mode. Simulation can be performed either on the computer operating the instrument or on a separate computer.

Whenever FluentControl goes into simulation mode, the selected instrument configuration file that is selected in the drop-down list will be applied.

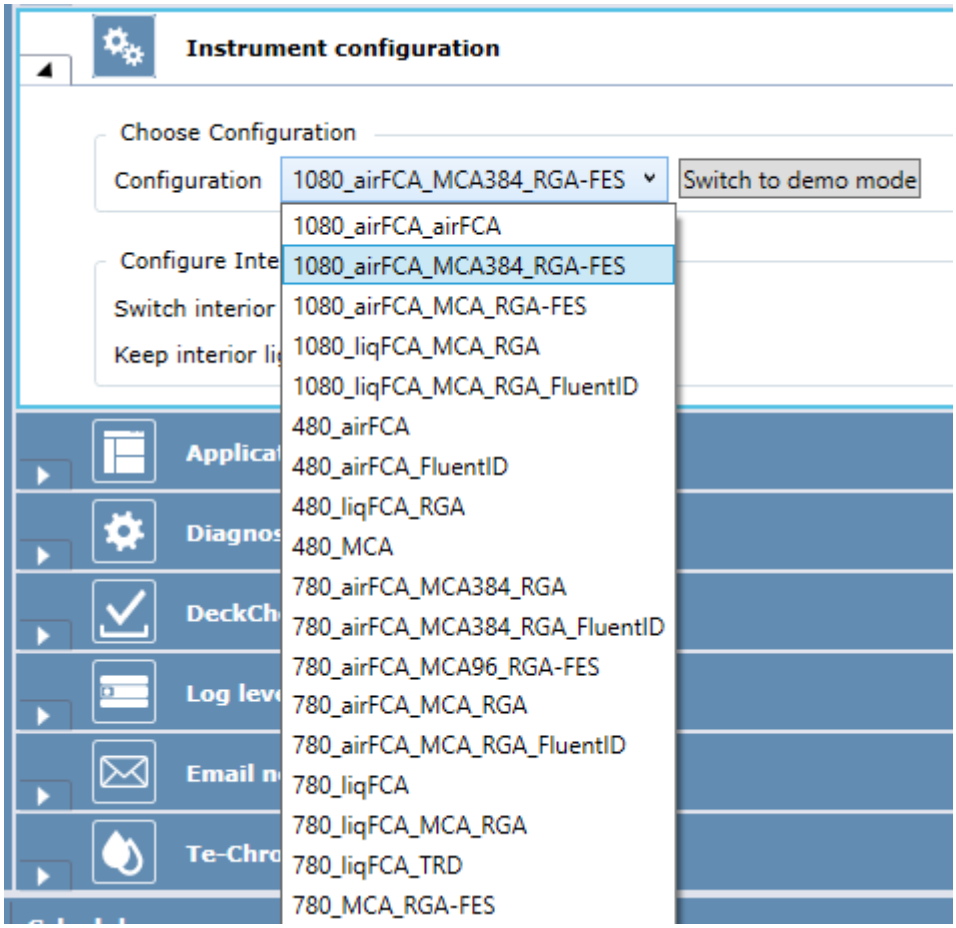


Fig. 19: Instrument Configuration List

1. In menu **Settings** click **Configure System**.
2. Open **Instrument configuration** and select the instrument configuration you want to run. The configurations in this drop-down list reflect the instrument configuration files that are stored under the following file path: C:\ProgramData\Tecan\VisionX\InstrumentConfigurations

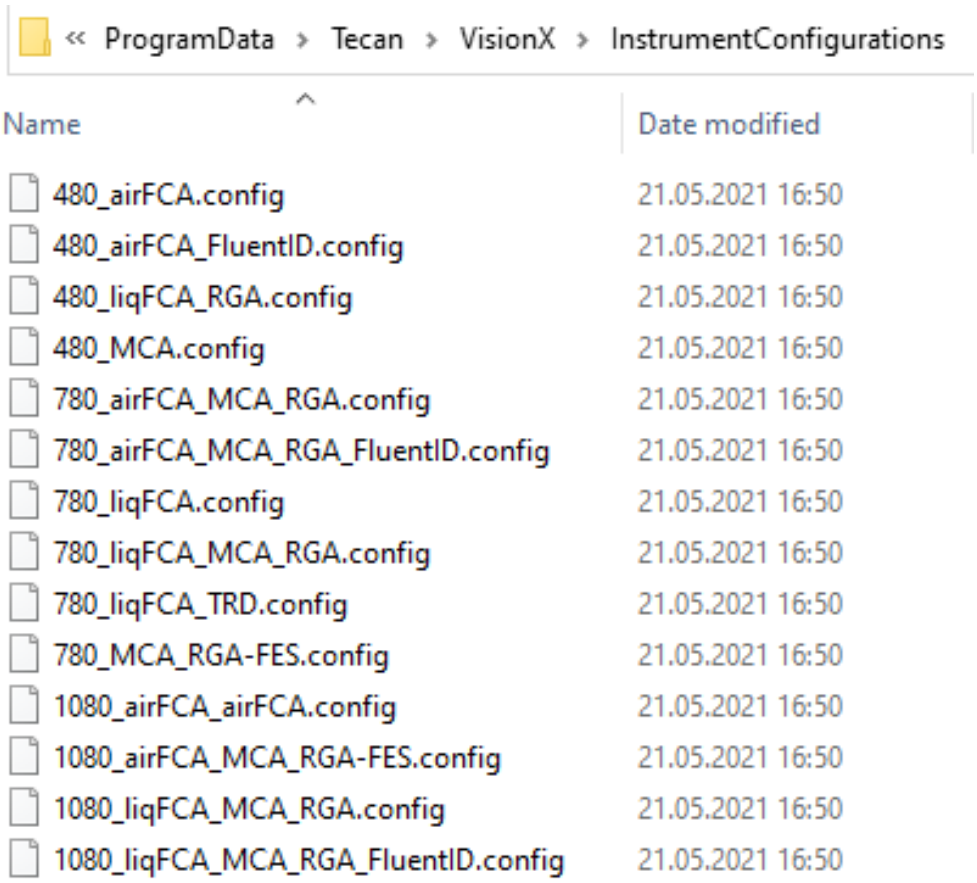


Fig. 20: Instrument configuration files on the C-drive

3. Click **Switch to demo mode button**.

In addition to the demo mode instrument configurations, FluentControl will also download the instrument configuration of the real instrument to the same folder, as long as the PC was previously connected to an instrument.
This configuration file as shown in the illustrations below, contains the serial number of the connected Fluent instrument. Select this instrument-specific configuration file on the computer connected to the instrument to make sure that the same instrument configuration is selected for the simulation mode

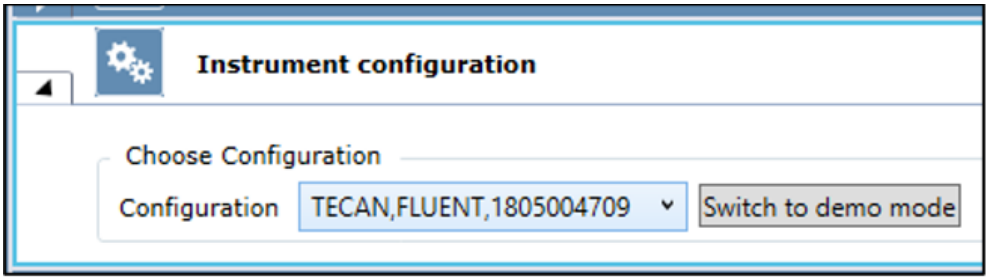


Fig. 21: Instrument configuration connected instrument

To use this instrument configuration in the simulation mode on a separate computer, copy-and-paste this configuration from the current PC and copy the file into the corresponding folder of the FluentControl installation on the separate computer. The next time the **Configure System**tab opens, the drop down list will refresh and the imported instrument configuration can be selected.

If you do not have access to the computer to which the instrument is connected, but do have a Snapshot file that was taken from the relevant instrument in real mode:
By default, the Snapshot file includes a version of the instrument configuration file that has been renamed to system.config. Copy, paste, and rename the file as needed and then select this instrument configuration in FluentControl as described above.

If you do not have access to the instrument configuration file of your real instrument:

1. In **Window** open the **3D Simulator**.
2. Select **File** and **New Configuration**.

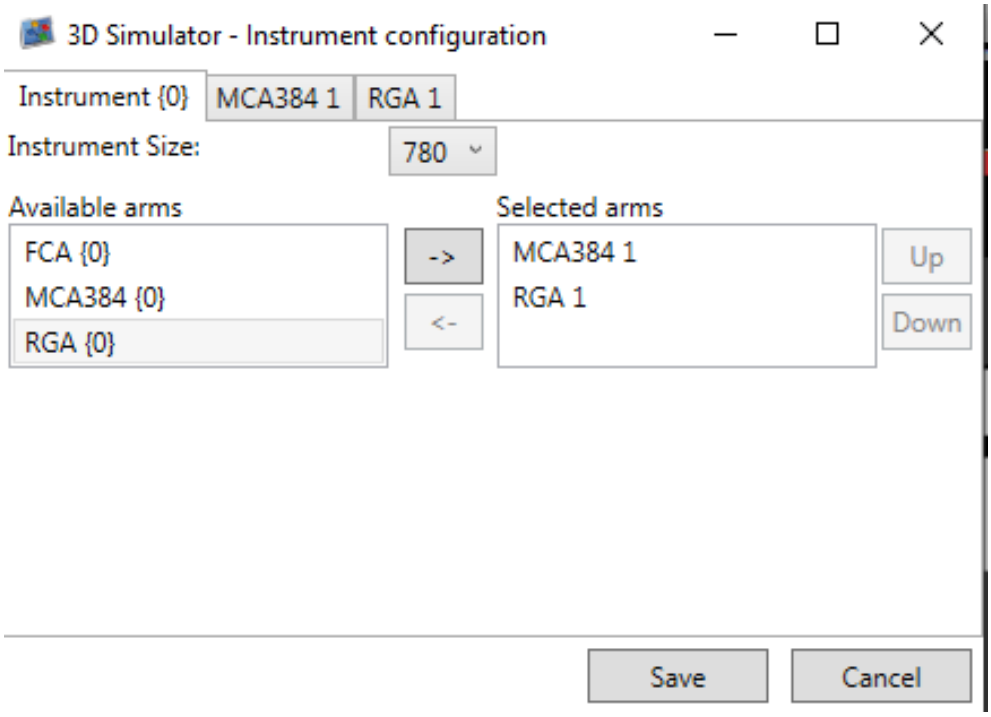


Fig. 22: Instrument configuration 3D Simulator

3. Select the relevant configuration for each arm used by the instrument. Configure the simulated instrument to match the real instrument as closely as possible
4. Click **Save** and rename the configuration file as needed.
5. Select the instrument configuration as described above.

5.4 The Configuration Editor

The Configuration Editor allows the user to configure an entire worktable with all racks, devices, labware and methods by drag and drop. After finishing the user can run the instrument in the simulation mode in order to see whether the configuration was setup in a correct way.

You interact with the user interface documented in this manual or with **TouchTools**, an intuitive, customizable touch monitor interface. **TouchTools** is now an integral part of FluentControl. You can quickly create interfaces using **TouchTools** standard features like easy variable entry, or build completely customized interfaces for your operators.

Overview of the Main Window

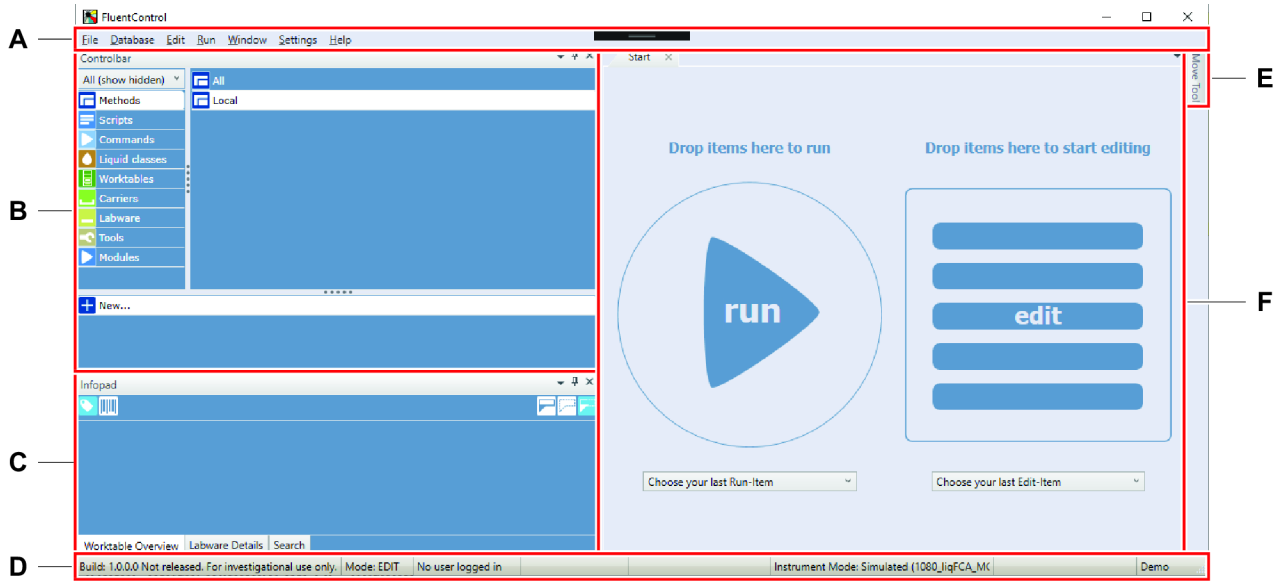


Fig. 23: FluentControl window on PC-screen

A	Menu bar	**B**	Controlbar
C	Infopad	**D**	Status bar
E	MoveTool	**F**	Main section

Menu bar The menus are displayed on the menu bar located near the top of the main window. For a description of the **File**, **Database**, **Edit**, **Run**, **Window**, **Settings** and **Help** menu refer to section “Menus” [▶ 115].

Controlbar The **Controlbar** contains items for methods, scripts, worktables, commands, objects (carriers, labware, and tools), and liquid classes.

Infopad In the **Infopad**, you see information about objects (carrier, labware, and tools) that are currently selected on the worktable.

Start A tabbed page used for dropping items on to run or start editing.

MoveTool Positions are measured and checked by moving the respective arms to the required position with FluentControl **MoveTool** support.

Status bar The area at the bottom of the main window that lists the status of FluentControl and gives other information, such as the build number (version), current software mode, instrument mode, and license.

5.4.1 Differences to Evoware

- Flexible layout: Panels and views in the application software can be arranged and rearranged to allow adaptation to different needs and tastes.
- Flat interface: Several items can be edited at the same time (e.g., Script, Carrier, worktable). No pop-up parameter dialog.
- Move tool: Arms can be moved independently of a script by using the move tool. Zero-G allows moving the arms without motor resistance.
- Worktable concept: FluentControl allows placing a Carrier onto another Carrier and Labware on top of Labware.
- Toggle smart mode: This setting allows you to get relevant information more quickly
- Barcodes and labels are more visible
- The 3D worktable represents the physical worktable in an easily interpretable way
- The 3D simulator is a graphical 3D view visualization of the instrument
- “One-touch” Zero-G teaching for third party devices or position optimization

5.4.2 Arranging Windows

The main window layout is partitioned into resizable areas, which can be adjusted by a grid splitter.

- You can extract the panes of the main window out of the main window by double clicking on the pane header.
- Another double click on the extracted window reintegrates that pane back into the main window. The Window menu provides a functionality to reset the customized panes back to the default.

FluentControl provides various tools and options to help you arranging document and tool windows and navigating between them. FluentControl automatically saves the workspace settings between sessions.

The main window of FluentControl consists of several windows that can be arranged freely by clicking on their title bar and dragging them onto the position tool that appears.

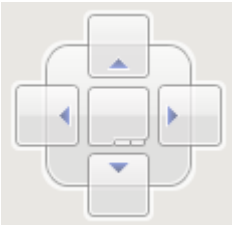


Fig. 24: Position tool

Window Options

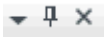


Fig. 25: Options

Windows can be pinned and unpinned. An unpinned window slides aside as soon as another window is activated.

Window Splitters

Windows can also be split. When the pointer becomes a  drag the pointer to move the split line.

5.4.3 Controlbar

The Controlbar is used for selecting objects (**Carriers**, **Labware**, and **Tools**), commands and liquid classes, which are needed to set up a script. The Controlbar is organized in categories. Each category contains a list of objects, which are arranged in groups.

1. Click on the group's title to access group specific items.

   The selection of the displayed items can be changed.

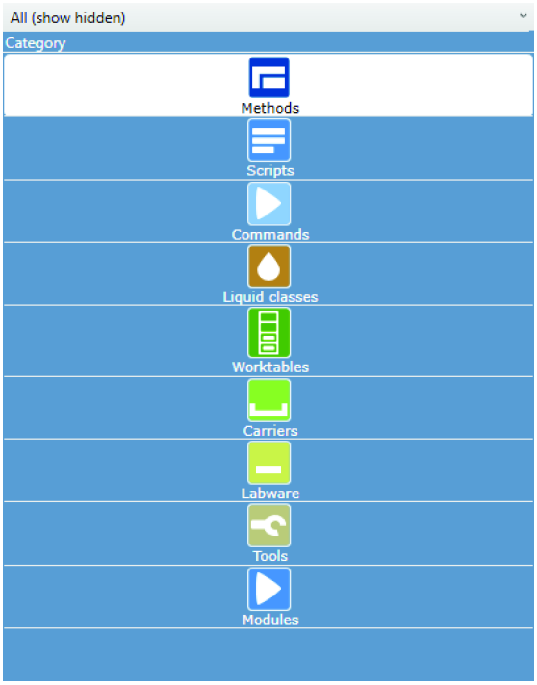


Fig. 26: Controlbar: Category view

2. Click the drop-down arrow to see more options:

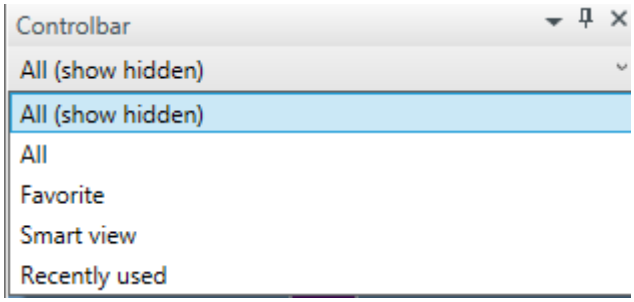


Fig. 27: Controlbar: Selection of displayed items

If you select the Smart view option, for example, only context relevant items are shown in the Controlbar. For the supported views, see also the Controlbar Views command on the Settings menu.

5.4.3.1 Methods

A carrier is a removable support on the instrument's deck. You can use a carrier to hold one or more pieces of labware in place. Carriers can be fitted to the deck at pre-defined grid positions (positioning pins/clips). The pre-defined grid positions are spaced 25.0 mm apart in X-direction. By default, carriers are saved in the base workspace, labware is saved in scripts. Methods can be visible on the touch interface for operators. Methods can also be saved as "Maintenance" or "System Care" to show up under the "System Care" touch monitor option. All scripts in a method must share a base worktable.

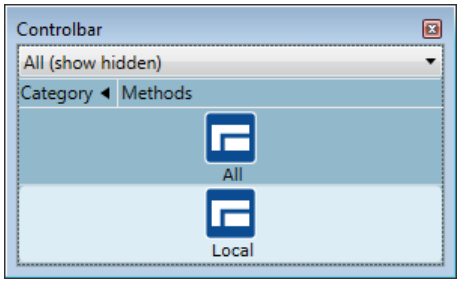


Fig. 28: Methods category in Controlbar

5.4.3.2 Scripts

Scripts are a collection of commands to carry out the actions and interactions required for an assay.

5.4.3.3 Worktables

This editor allows you to edit the base worktable which may be used by one or more scripts.

5.4.3.4 Commands

The **Command** menu containing commands for the installed arms and devices − for example:

- All

- Direct commands
- FCA 1
- MCA384 1
- RGA 1
- Programming
- Touch Tools
- Worklist
- Sample Tracking
- Worktable
- Communication

5.4.3.5 Liquid Classes

A liquid class defines the robotic actions that will take place during a **pipetting script** command.

Liquid classes are defined per liquid type and pipetting mode (e.g., single pipetting or multi-pipetting, free dispense or wet contact). One liquid class covers the whole volume range for FCA, MCA384 and MCA96. Sub-classes are defined for arm type and tip type.

FluentControl comes with pre-defined liquid classes. The liquid classes should always be tested and validated for the target application. The pre-defined Multi-pipetting liquid classes usually need optimization for the target application.

Opening the editor for displaying and editing liquid classes

1. To open the liquid class editor, click **Liquid Classes** in the Controlbar.
2. Double-click on any of the liquid classes.

   *Alternatively, on the File menu, click **Open** > **Liquid Class**.*

   Each liquid class is divided into subclasses depending on arm type and tip type.

   The appropriate subclass is automatically used at runtime.

The pipetting parameters for each pipetting command type are split amongst the **Aspirate**, **Dispense** and **Mix** sections which in turn are divided in up to four sub-sections:

- **Easy Control** (based on Tip Type): A graphical editor that allows to easily set the main parameters.
- **Detection & Positioning** (based on Tip Type): You can configure the liquid level detection, tracking options, error handling, and retract properties here.
- **Formulas** (based on Tip Type): Liquid handling parameters to be entered as functions.
- **MicroScript** (based on Arm Type): Pipetting steps can be defined as a sequence of **MicroScript** commands offering a high degree of flexibility.

5.4.3.6 Carriers

A carrier is a removable support on the instrument's deck. You can use a carrier to hold one or more pieces of labware in place. Carriers can be fitted to the deck at pre-defined grid positions (positioning pins/clips). The pre-defined grid positions are spaced 25.0 mm apart in X-direction. By default, carriers are saved in the base workspace, labware is saved in scripts.

With FluentControl, you define an object (e.g., a microplate) only once, and then, you use it at any required position on the worktable.

Coordinates

FluentControl uses a hierarchical system for the coordinates:

- The labware coordinates are relative to the site on which the labware is placed.
- The site coordinates are relative to the carrier.
- The carrier offsets are relative to the grid position on the worktable
- The grid positions on the worktable have a spacing of 25 mm.

This hierarchy allows you to place the labware at any suitable site, carrier and worktable position without having to redefine its coordinates.

Types of carriers

The **Carriers** menu contains the following types of carriers:

- Deck Segment, which attach to a grid an provide nests for labware
- Grid Segment, which attach to the grids and provide pins for runners to slide on and to position hotels
- Runner, which slide on and off grid segments (e.g., a tube runner)
- Hotel
- Nest, the part of the carrier that provides a position for a microplate or SBS format tip box
- Device
- Deck Extension

5.4.3.7 Labware

Labware refers to consumable items such as microplates or sample tubes. Microplates have a regular arrangement of containers or cavities (wells) for holding liquids or samples.

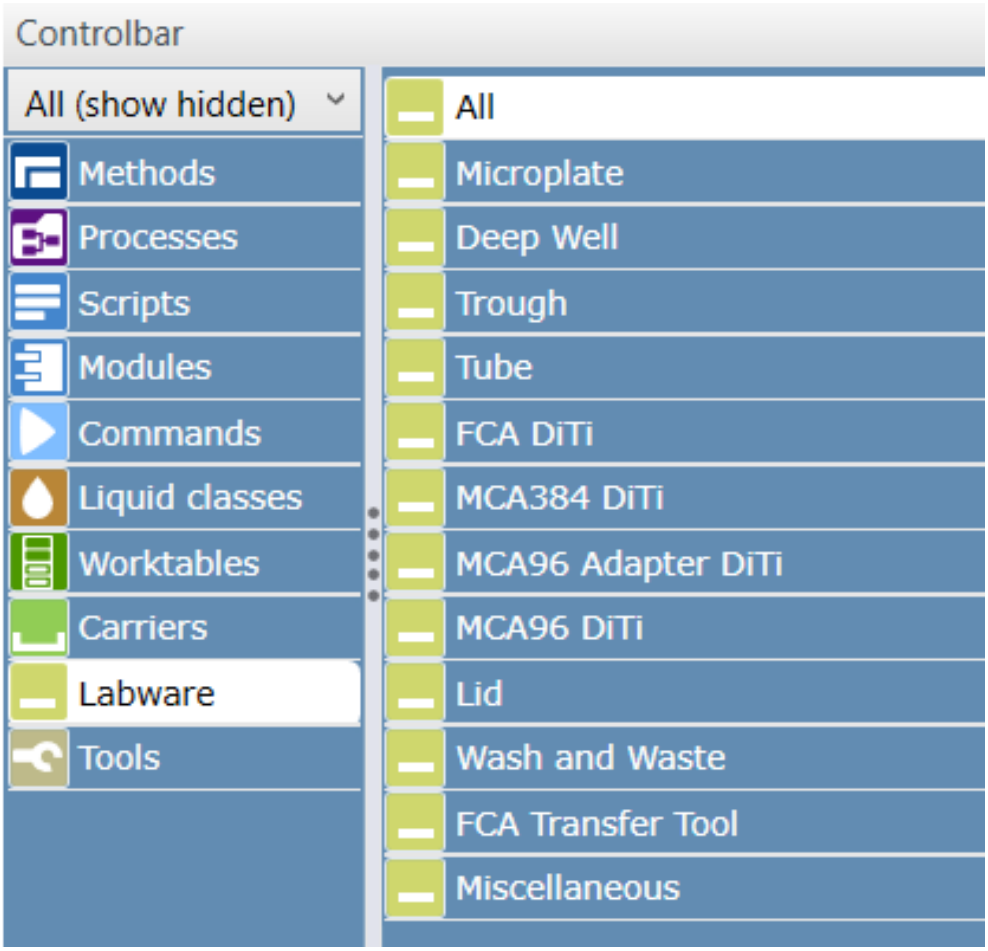


Fig. 29: Labware Tab

The **Labware** menu contains the following sub-categories:

- Deep well Microplates.
- Disposable Tips used only on the FCA arms.
- Lids on Microplates and Lids on Disposable Tip Boxes.
- Disposable Tips used only on the MCA 384 head with one of the 384 Tips Adapterplates.
- Disposable Tips used only on the MCA 96 head.
- Disposable Tips used only on the MCA 384 head with one of the 96 Tips Adapterplates
- Flat Microplates.
- Troughs for the FCA arms and SLAS format troughs.
- Tubes with or without lid of different sizes.
- Wash stations and liquid waste stations.
- FCA Transfer Tool: Disposable to dispose nested DiTi layers.
- Miscallaneous: E.g. spacers.

Allowed Labware

If you have already put a carrier onto the Worktable Editor, click on it to see which labware types are compatible with the currently selected carrier.

The FluentControl database already contains a large selection of pre-defined labware types. You only need to create or edit labware if you cannot find a standard item which is suitable for your needs.

5.4.3.8 Tools

Tools are objects which attach to arms and are not thrown away, for example **MCA384 Adapters, fixed tips, or RGA fingers**.

5.4.3.9 Modules

Modules are specially designated scripts that can be inserted into the other scripts.

5.4.4 Infopad

In the Infopad, you see information about carrier, labware and tools that are selected on the worktable.

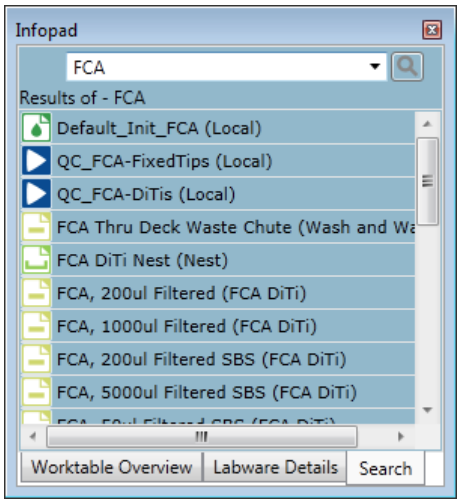


Fig. 30: Search example in Infopad

Errors

Context check errors are displayed in the error tab of the Infopad. It shows the errors of the active editor and is labeled with the name of the current opened object. Click on a line in the error tab of the Infopad to select the corresponding line in the script editor.

5.4.4.1 Worktable Overview

The worktable overview can be easily navigated by clicking on the worktable. It is where you can rename labware and hide or reveal labels.

5.4.4.2 Labware Detail

Labware Details will show a graphic of the selected part, the name, label, and additional information.

5.4.4.3 Search

You can search for objects against labware name and part number. The results are shown in the Infopad area. The result entries have the same drag and drop behavior as in the Controlbar. The search covers the currently opened script editors and method editors as well as all Controlbar elements. You can also search for opened script content (e.g., variables).

5.4.5 Start

In this pane, also called the main section, you find the run and the edit part. By drag and drop to these sections, you can **Edit** or **Run** a script.

From the FluentControl Start pane, you can perform the following tasks:

- **Run**: Run an editor item like a script/method or a direct command.
- **Edit**: Edit an editor item.

Two drop zones of Start: Run and Edit

The Start pane provides two drop zones, which enable the user to run or edit an editor item like a script/method or a direct command. The start of the drag and drop operation is the search and the Controlbar and the drop target is one zone for edit, the other for run direct.

The Start pane has two recent lists, which provide last ran scripts and methods, and a second list, which provides entries for create editor items for script and method and a list of last edited items.

5.4.6 MoveTool

After the initialization of the instrument, the arms cannot be moved without FluentControl **MoveTool** support. If FluentControl is connected to an instrument (real or simulated), you can use the **MoveTool** and move the arms individually by clicking the arrows in the **MoveTool** pane. The arms speed up upon continuous movements.

5.4.6.1 MoveTool Pane

If the instrument is switched to the Zero-G state, then you can move all axis of all the connected arms manually.

⚠ CAUTION

Moving parts!

Moving parts can cause crushing or cutting injury.

- Always keep hands and body away from moving parts while teaching with open safety panels.
- Take unintended motion of robot arms into consideration while teaching with open safety panels.
- Do not attempt to circumvent the door sensors.

5.4.6.2 Teaching and the MoveTool

If you are teaching a custom carrier, labware or vector, there is a direct connection between the corresponding editor and the **MoveTool**:

- When you click **Move** in one of the editors, the robotic arm selected in the **MoveTool** will move to the desired position.
- Transfer will record the actual position of the arm selected in the **MoveTool** in the current field.
- Show Delta will show the difference between the taught position and the current position of the arm selected in the **MoveTool**. It will not change the definition of anything.
- Adjust Site will use the labware or vector definition and the current position of the arm selected in the **MoveTool** to calculate new site offsets for only the topmost carrier.

The **MoveTool** will indicate what tools are mounted on the arms. Tips can be mounted via **Direct Commands**. Positions will be correctly calculated with mounted tips.

NOTICE

Transfer will take the position of the arm, even while the arm is in Zero-G.

In this version of the software, **Show Delta** and **Adjust Site** will only work if the arms are active, i.e. on standby.

- To work around this issue, take the arm out of Zero-G before using **Show Delta** or **Adjust Site** by clicking on a directional button or the Zero-G button in the Move Tool.

While teaching carrier positions or labware with the doors open, the arms will move at half their normal speed. Do not attempt to circumvent the door sensors.

5.4.6.3 Home Position

A home position is defined for each robotic arm. After initialization of the instrument, the robotic arms move to their home positions. The default home position of the FCA module, for example, is over the nearest wash/waste station with Z-axes on Z-travel position. To configure home positions, use the Configure System command on the **Settings** menu, expand the devices section and click **Home-positions**.

5.4.6.4 Zero-G

To release the resistance on the motors to allow the robotic arms to be positioned gently by hand, click **Zero-G**. Click **Zero-G** again to reset the instrument on Standby. Zero-G mode will be canceled automatically when a predefined interval of time has passed without further user interaction.

5.4.6.5 Moving the FCA

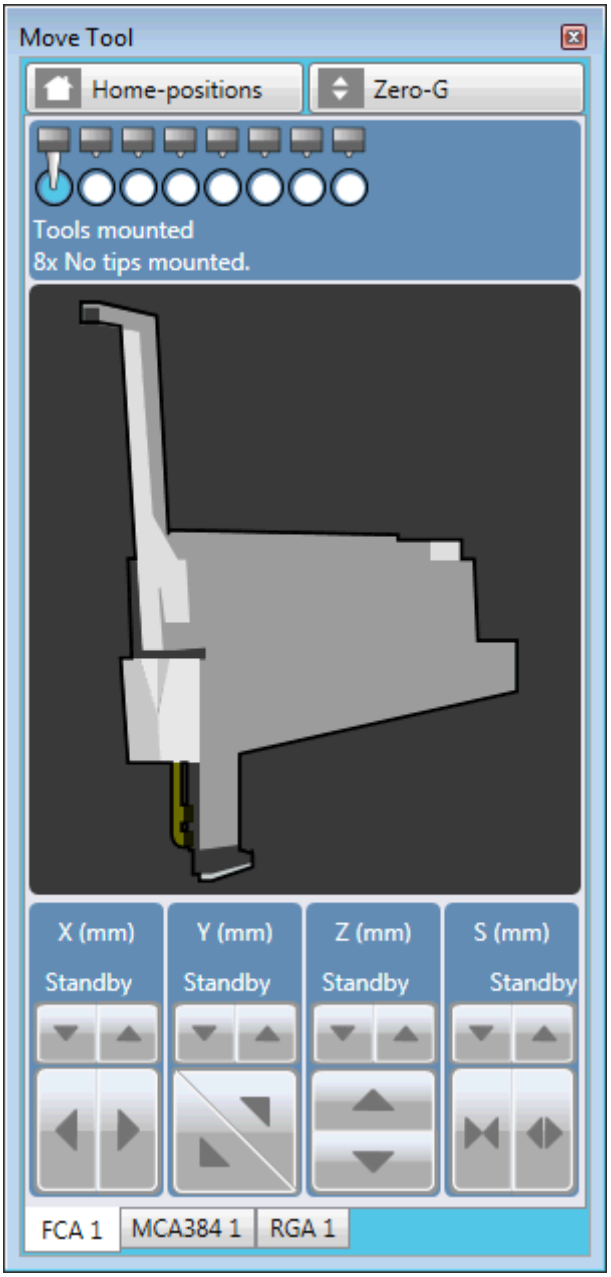


Fig. 31: MoveTool: FCA

X [mm]: Movement axes from left to right: Use the Left Arrow and Right Arrow keys on the numeric keypad or click the appropriate button.

Y [mm]: Movement axes from rear to front: Use the Up Arrow and Down Arrow keys on the numeric keypad or click the appropriate button.

Z [mm]: Movement axes from top to bottom

S [mm]: Spreading of the FCA channels (expand or narrow)

To move the arm one step (0.1 mm), press the appropriate arrow key. If the key is kept pressed the arm is continuously moved in the direction as defined in the table below:

 expand (+S)	 rear (+Y)	 up (+Z)
 left (-X)		 right (+X)
 narrow (-S)	 front (-Y)	 down (-Z)

5.4.6.6 Moving the MCA384

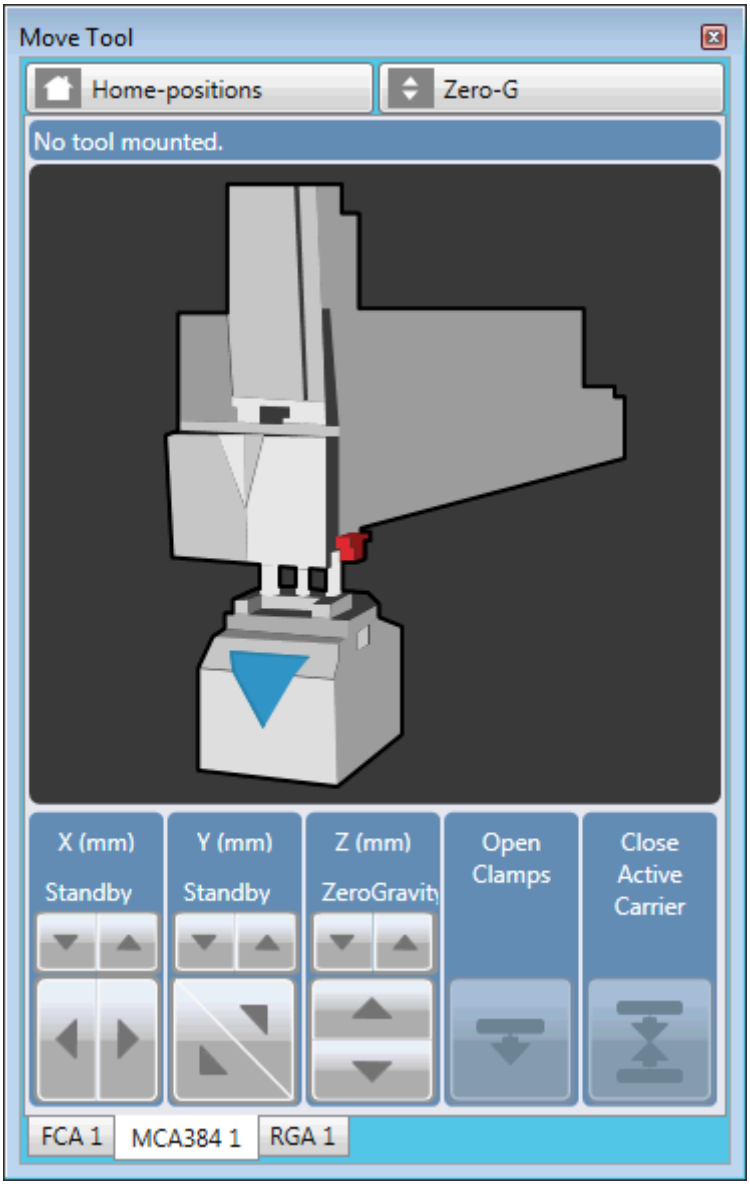


Fig. 32: MoveTool: MCA384

X, Y, and Z as described above.

Open Clamps

⚠ CAUTION

Moving parts!

Moving parts can cause crushing or cutting injury.

- Always keep hands and body away from moving parts while teaching with open safety panels.
- Take unintended motion of robot arms into consideration while teaching with open safety panels.
- Do not attempt to circumvent the door sensors.

Close Active Carrier

In addition to the standard MP nest segments/runners and the DiTi nest segments the MCA384 makes use of the system segment and the active carrier—i.e., an active DiTi nest segment especially designed for the MCA384. Retainers activated by a built-in solenoid keep the DiTi boxes in place during the DiTi pickup procedure. While teaching a DiTi 15, 50, or 125 box on the active carrier, the DiTi box is not locked by the solenoid.

1. To keep the DiTi box in place while teaching, click **Close Active Carrier**.

   *Thus the locked DiTi Box has the same position while teaching as while using the **Get Tips** command.*

2. To open the clamps, click **Close Active Carrier** again.

Using the keys on the numeric keypad to move the MCA384 :

7	8 rear (+Y)	9 up (+Z)
4 left (-X)	5	6 right (+X)
1	2 front (-Y)	3 down (-Z)

5.4.6.7 Moving the RGA

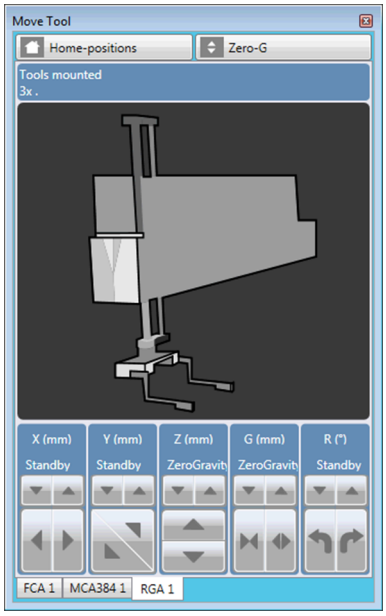


Fig. 33: MoveTool: RGA

X, **Y**, and **Z** as described above.

G [mm]: From narrow to wide gripper distance. Use the 0 and . keys on the numeric keypad as described in the table below or click the appropriate button.

R [°]: Clockwise rotation of the gripper when viewed from above. Gripper rotation is measured in 0.1° steps.

Use the 7 and 1 keys on the numeric keypad as described in the table below or click the appropriate button.

7 clockwise (+R)	8 rear (+Y)	9 up (+Z)
4 left (-X)	5	6 right (+X)
1 counterclockwise (-R)	2 front (-Y)	3 down (-Z)
0 narrow (-G)		. wide (+G)

5.4.6.8 Moving the Arms (Task of Path Finder)

Path Finder controls the movements of all arms between safe points of vectors and travel positions of labware. If necessary arm movements can be influenced by bounding boxes on the worktable, speed parameters in **Labware** attributes or **MicroScript** commands, defining longer **Vectors** and using **Move Arm** commands.

Task of Path Finder

The task of the Path Finder is to move the arms to their target locations and also move other arms out of the way if necessary. Path Finder will avoid all bounding boxes, too. Bounding boxes are the theoretical “no fly zone” around worktable objects and are defined within FluentControl. Bounding boxes are also what glows when the cursor hovers over an object in the **Worktable** editor. Path Finder will do its best to move arms without issues and calculate problems before they arise. But there are the following limits.

Cannot start from colliding position or cannot move to a colliding position

"Cannot start from colliding position or cannot move to a colliding position" is a Path Finder error message. It does not mean that the arm has physically collided with something. A colliding position means that Path Finder believes that arm is inside of a bounding box of another object at the start of a move or would have to enter a bounding box at the end of the move. The message may look complex, but it is giving hints about what objects are causing the issue (e.g., the left Gripper Finger has an issue and is “colliding” with a hotel). If the colliding position is the current arm position at the start of the movement, then manually re-position the arm to free space and press retry may help.

Limits of Path Finder

For the Robotic Gripper Arm, Path Finder will move the arm from its current position to the Safe Point of the target vector. Movements from the Safe Point of the vector to the End of the vector are not under control of Path Finder, this enables vectors to enter the bounding boxes of the target and surrounding labware or carriers.

For the pipetting arms, the Flexible Channel and Multiple Channel arms, Path Finder will move the arm from current position to the Z travel of the target position. If the arm is inside of a bounding box before the move, Path Finder is allowed to lift the arm in Z only to exit the box. From Z Travel through the pipetting step, the arm is under control of the Pipetting engine and will execute what is defined in the liquid class. Path Finder is not in control of the movement during the pipetting step.

During configuration of a Fluent, please verify that all desired movements are physically possible. Path Finder will do its best to move arms without issues and calculate problems before they arise. But there is no software feature that can make a Flexible Channel Arm with a mounted 10 µl Tip reach the bottom of a 100 ml Trough before the ILID block would touch the top.

The pipetting instrument coordinate system

For the instrument coordinate system, the values are relative to the chassis. In the RGA move command, they are relative to that arm. In general the following description applies to all arms, when facing the instrument:

- The **X**,**Y**,**Z** “0 Point” or origin is the Left, Front, Worktable level
- **X**: Negative / lower values move to left; Positive / higher values to the right
- **Y**: Negative / lower values move to the front; Positive / higher values to the back
- **Z**: Negative / lower values move lower; Positive / higher move higher
- **Yspread** (FCA Only): Negative / lower values move closer together; Positive / higher values spread apart
- **G** (RGA Only): Negative / lower values move closer together; Positive / higher values spread apart
- **R** (RGA Only): Negative / lower values move clockwise; Positive / higher values move counter-clockwise (viewed from top)
- The coordinate system extends in the negative direction beyond the origin (e.g., RGA with extend Z can reach below the worktable in Z, or negative)

5.4.6.9 Moving the RGA

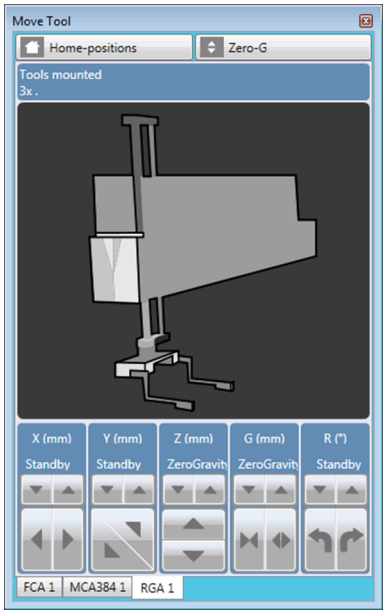


Fig. 34: MoveTool: RGA

X, **Y**, and **Z** as described above.

G [mm]: From narrow to wide gripper distance. Use the 0 and . keys on the numeric keypad as described in the table below or click the appropriate button.

R [°]: Clockwise rotation of the gripper when viewed from above. Gripper rotation is measured in 0.1° steps.

Use the 7 and 1 keys on the numeric keypad as described in the table below or click the appropriate button.

7 clockwise (+R)	8 rear (+Y)	9 up (+Z)
4 left (-X)	5	6 right (+X)
1 counterclockwise (-R)	2 front (-Y)	3 down (-Z)
0 narrow (-G)		. wide (+G)

5.4.6.10 Moving the MCA96

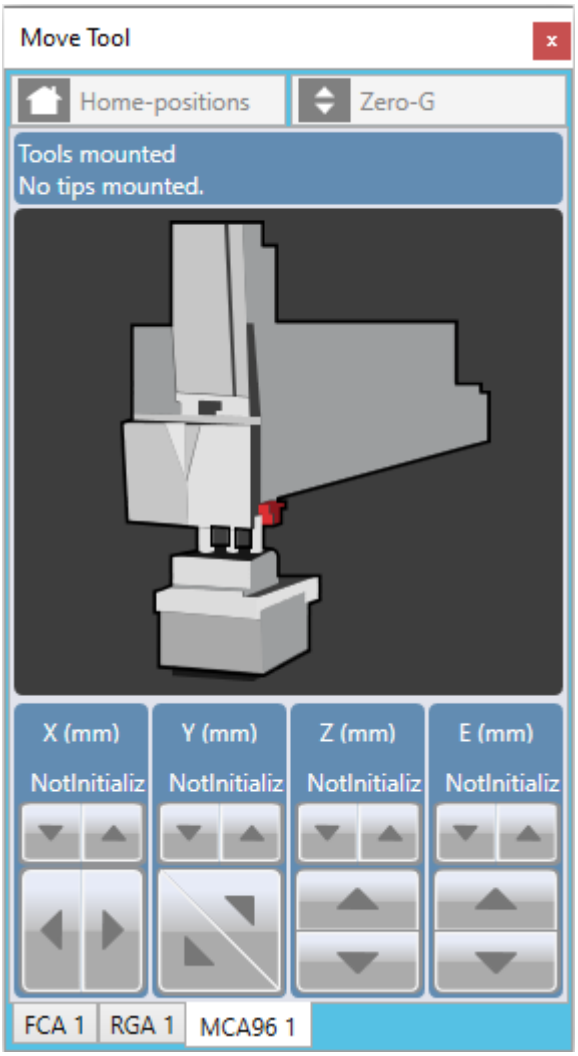


Fig. 35: MoveTool: MCA96

X, **Y**, and **Z** as described above.

E [mm]: Movement of the Ejector axis from top to bottom.

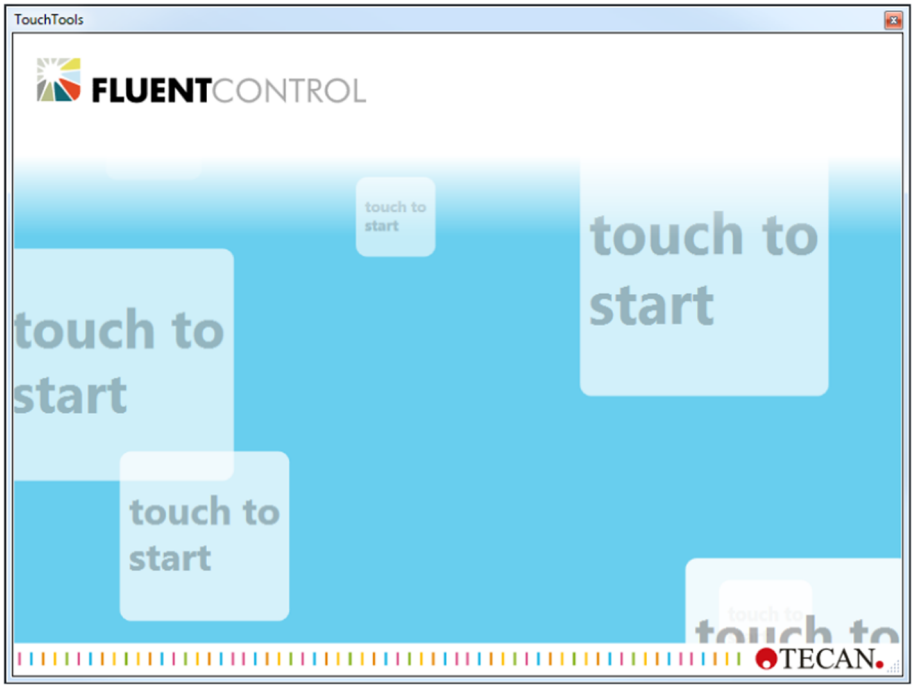
Using the keys on the numeric keypad to move the MCA96.

7	8 Rear (+Y)	9 Up (+Z)
4 left (-X)	5	6 Right (+X)

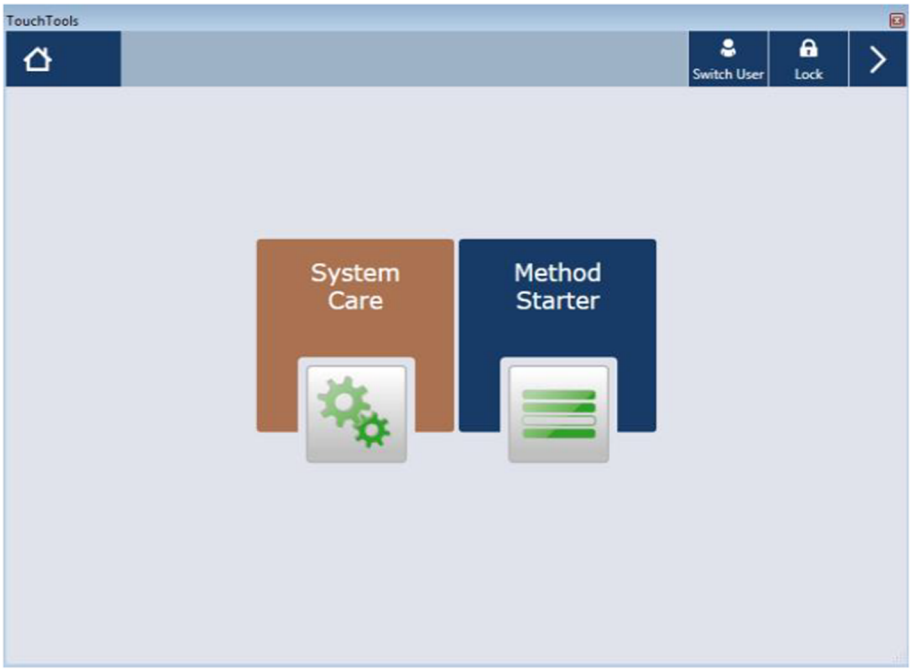
1	2 Front (-Y)	3 Down (-Z)

5.5 TouchTools

The TouchTools interface provides an intuitive operator interface for runtime control. It can display pictures and instructions to avoid common setup problems leading to more uptime and fewer interrupted runs. Instructions can be dynamically altered based on user input and are guaranteed to be viewed by the operator with every execution.



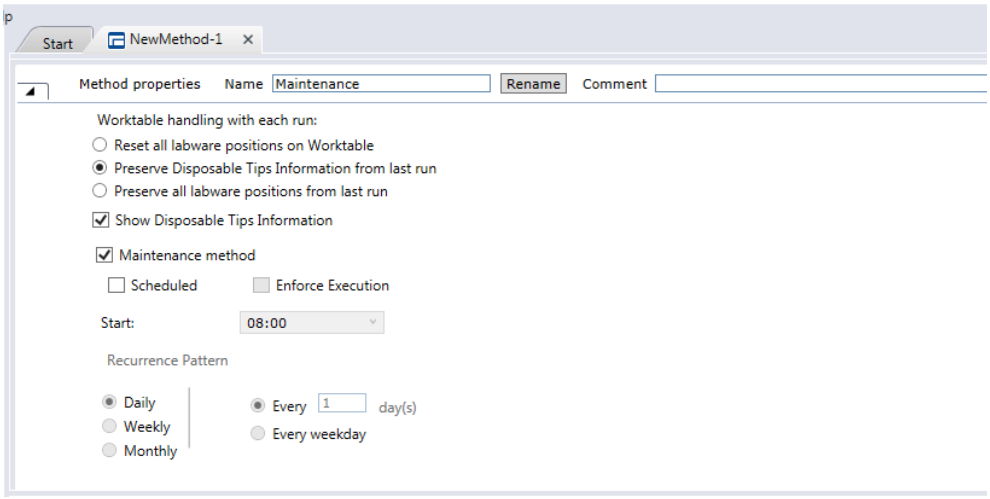
To start TouchTools, touch the screen. After startup, the home screen will show the following categories:



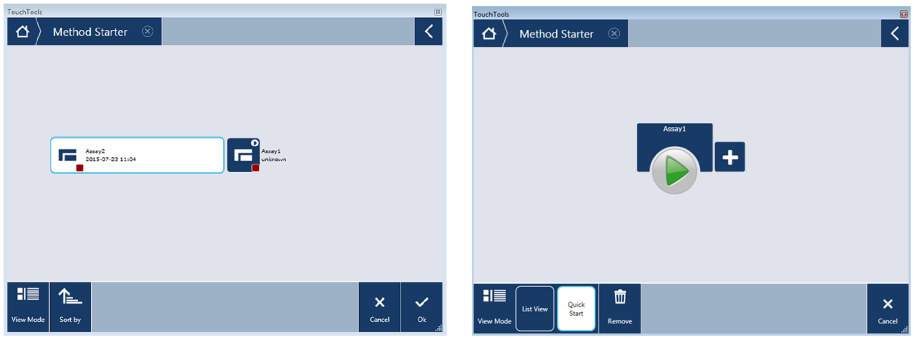
System Care: Select & start maintenance methods. Note: If a system care method is due then the System Care icon will flash.

Method Starter: Opens the menu where you can select the application method to run.

Only methods can be run from the TouchTools interface. To make a script available on TouchTools it must be prepared and saved as a method. However, not all methods will automatically be shown on the TouchTools, only those with the option “Is visible in TouchTools”. Check the option “Maintenance Method” if you want a Method to show up under System Care.



Once you are in the Method Starter, you can switch between the Quick View and the List View. The List View will show you all methods made available on TouchTools, whereas the in the Quick View allows the user to define favorites which can then be started more easily.



If you want to delete favorite methods from the Quick View, press the Remove button and select the method to be removed.

It is also possible, at any time, to switch users or lock FluentControl from TouchTools; to make those options available expand the tab by clicking on the arrow on the top right corner in TouchTools.

5.6 Scripts

5.6.1 Differences to Evoware

- More sophisticated context check. Script lines which are omitted by context check are clearly marked.
- Flags: Script line flags allow you to quickly deactivate script lines or set breakpoints.
- Details: for every command, parameters can be shown or hidden.
- Scripts can be run exclusively or together with other scripts as a method.
- In a method, you can specify how many times each script is run.
- Ability to preserve Labware positions from script to script using a method.
- Step-by-Step mode pauses before each step in a run simplifying development.

5.6.2 How to Create a Script

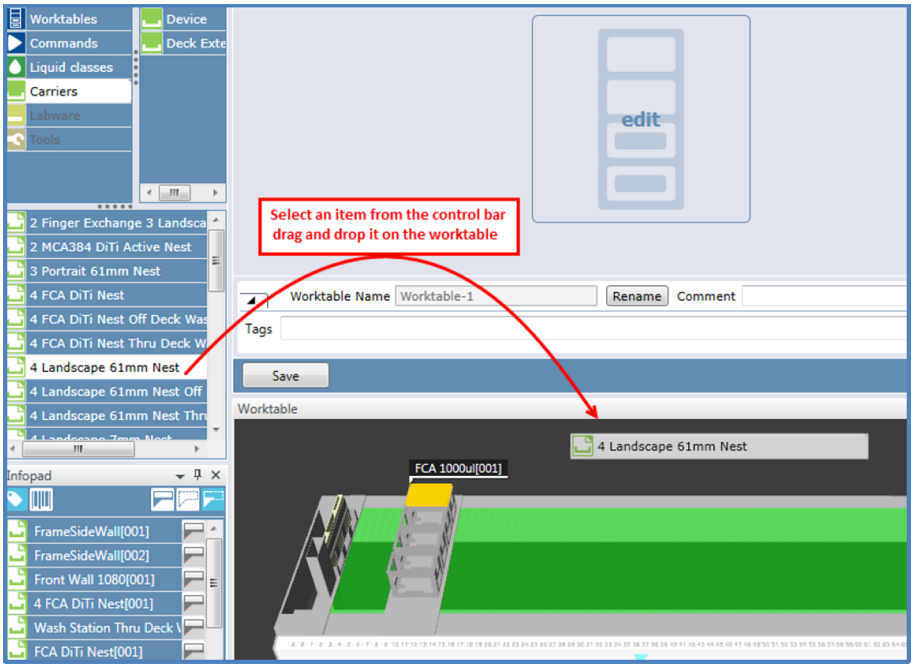
5.6.2.1 Create a new Worktable layout with the worktable editor

The first step in the process of setting up a script is to create the worktable layout that will be linked to the script and will be used to program the sequence of commands.

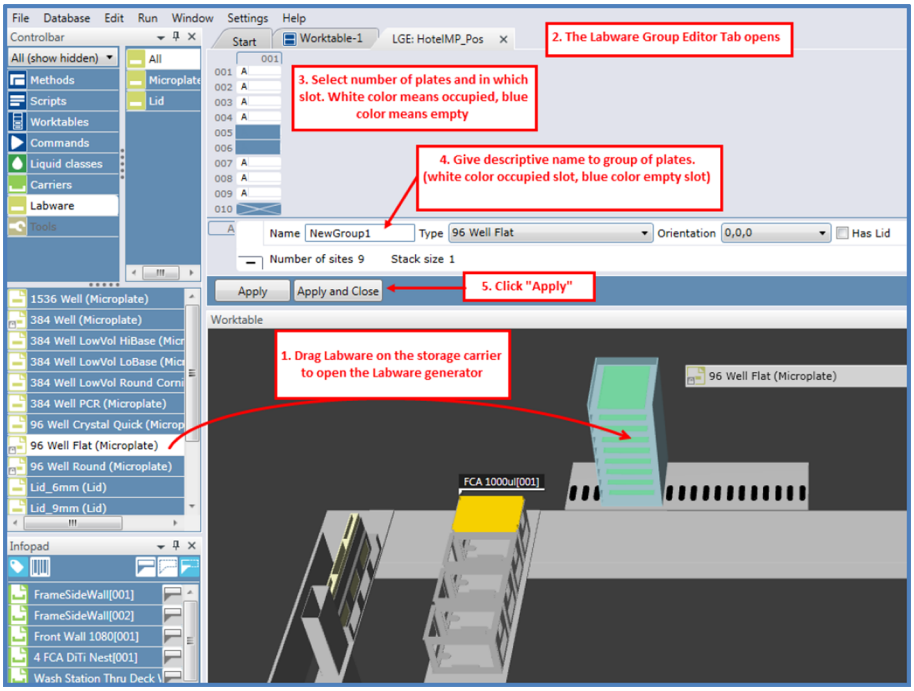
The worktable editor is a graphical representation of the pipetting instrument's worktable. To create a new worktable, select “new” in the control bar under category Worktable; a worktable editor tab will open.

Add elements on the worktable. The base worktable normally contains everything that is static (cannot be moved); Carriers (segments and nests) and Labware such as troughs, DiTi trays etc. In contrast, the worktable saved together with a script

adds elements which are specific to the script and can be dynamically moved during its execution.

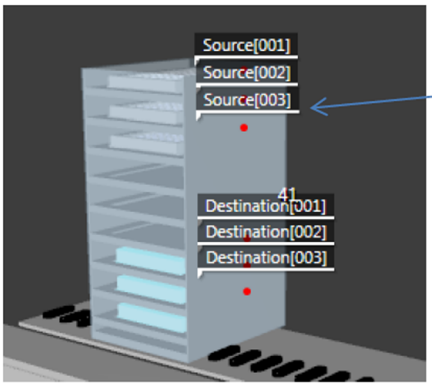


Labware is added to storage carriers (Hotel, Shelf, etc…) or storage devices (Carousel, Te-Stack, etc…) via the Labware Generator.



Specify the quantity of the Labware items required for your process. Select in which sites of the storage carrier you want to create the plates. Give a Labware Group name to the created plates.

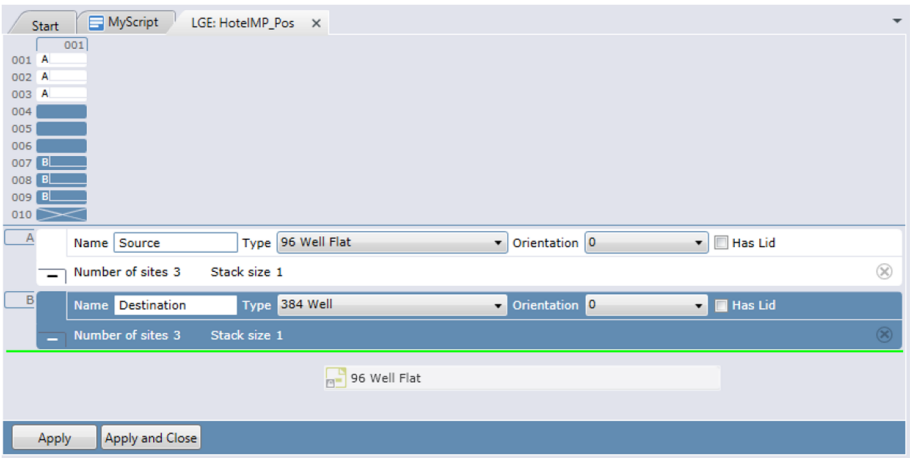
In the Labware Group Editor single or multiple groups of Labware can be created.



Labware Group name:

A group of plates defined in the Labware generator of FluentControl. The group name, type, quantity and storage location are defined. The group name is usually a descriptive name. You might have several different groups of plates consisting of plates of the same Labware type

To create a second group of plates just drag & drop a new Labware below the current group.



With all elements added to the worktable, save it. All worktables saved are listed in the homonym category in the control bar.

5.6.2.2 Create a Pipetting Script in the Script Editor

Open a new script editor (double click “new” in the control under the category Scripts; a new tab will open on the main window).

Link one of the existing worktables to the new script. Expand the Script Name section at the button of the script editor and select the worktable from the drop-down list.

Worktable

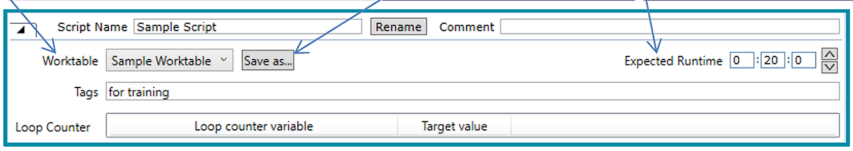
Choose the worktable you want to associate with the script. If you switch to another workspace later, you might be asked to re-connect Labware (orphaned Labware).

Save As

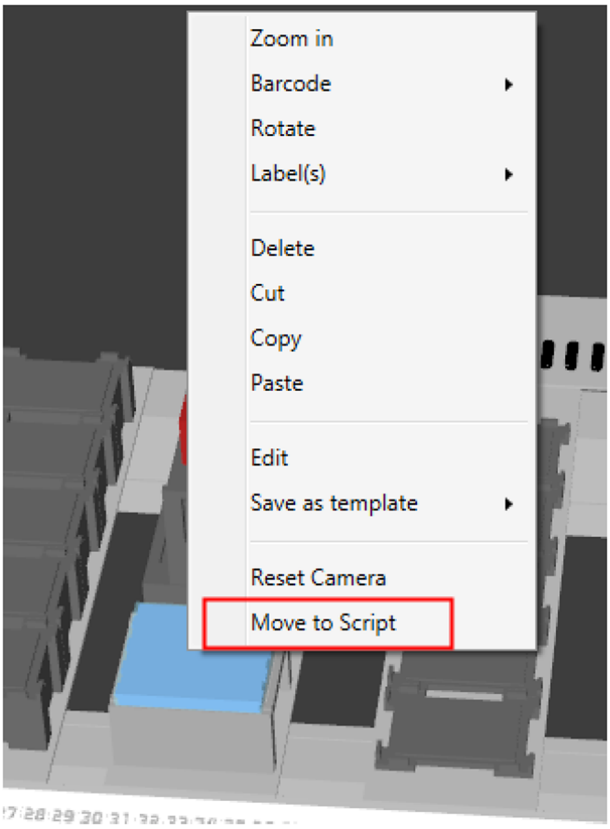
A Modified Worktable in the script editor can be saved as a new worktable template

Expected Runtime

Enter the predicted execution time of the script (based on experience). TouchTools will show a progress bar based on the expected execution time.

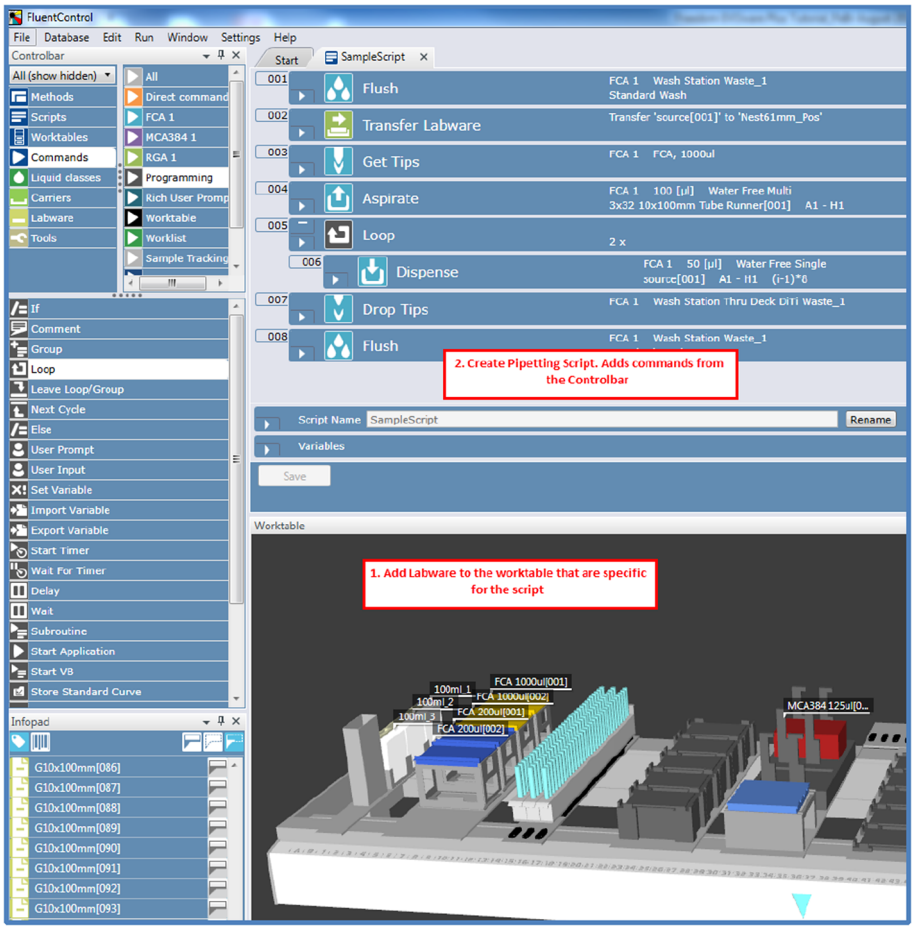


Modifications of the worktable in the script editor do also change the original worktable; however this is only true for objects which are flagged “Base Worktable” only; if the object is flagged “Script” it is only saved together with the worktable linked to the script. You can move an object from script to base worktable level and vice versa by right clicking the object and selecting the option “Move to Base Worktable” or “Move to Script”, respectively. Default, segments.at script level (“Moving Objects in between Worktables” [▶ 137])and nests are at base worktable level whereas Labware are at script level.

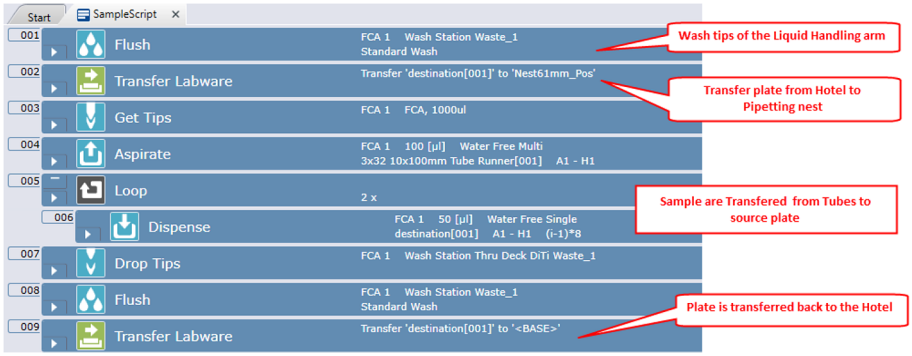


The Script Editor is used to set up a script. Commands for all devices are available in the Script editor.

Program your pipetting script and place the involved Labware on the pipetting carrier.



In FluentControl script lines are executed in a linear sequence. All commands therefore need to be added in the order they shall be executed.



5.6.2.3 Orphaned Labware

Adapting a script to run with another worktable can be done in a few steps. This facilitates sharing of scripts on different machines. In order to switch the worktable your script runs with, simply choose another worktable in the script editor. The software will re-assign the commands in the script to a corresponding Labware

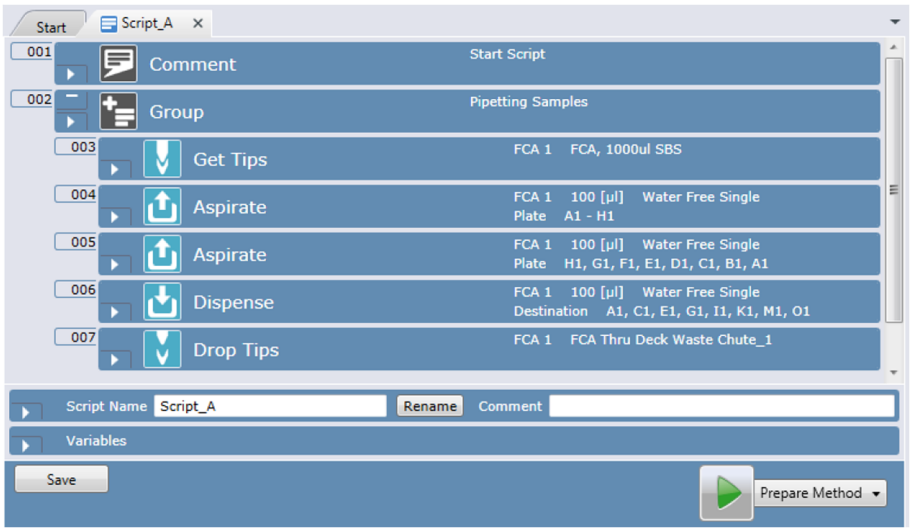
found on the new worktable. If there is Labware missing on the new worktable, it will appear in the list of Orphaned Labware. Simply drag and drop orphaned Labware to the new worktable to re-connect them.



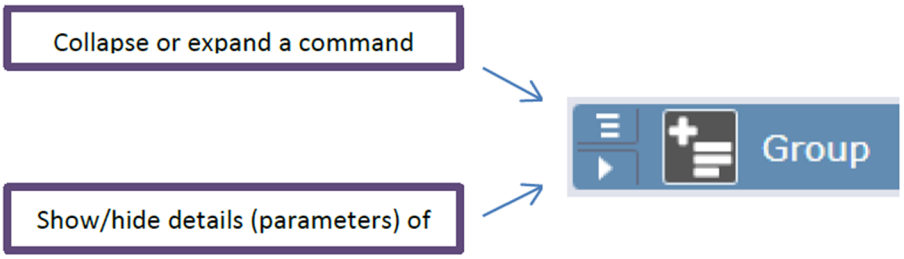
Fig. 36: Orphaned Labware

5.6.3 How to Edit a Script

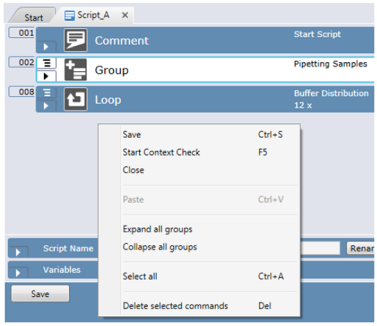
When you open an existing script, it will look like this:



Individual command lines can be expanded/collapsed. This allows you to hide details for a better overview.



By right-clicking outside command lines you get a window that allows you to expand/collapse all groups in the open script:

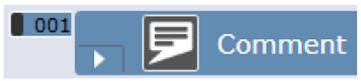


Expand all groups	All groups in the script will be expanded to show all command lines within the group. Groups include all commands that have grouping functions, e.g. loops
Collapse all groups	Groups will collapse to show overview of the script

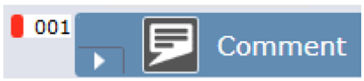
Disable command and Set Breakpoint

Flags next to command lines help you when programming a script and testing.
A black flag stands for a disabled command: The concerned lines are skipped at runtime.
A red dot signifies a breakpoint: when at runtime the script reaches this command line it will pause until the user confirms the run can continue.
Flags can be toggled with the right-click menu.

Disabled command



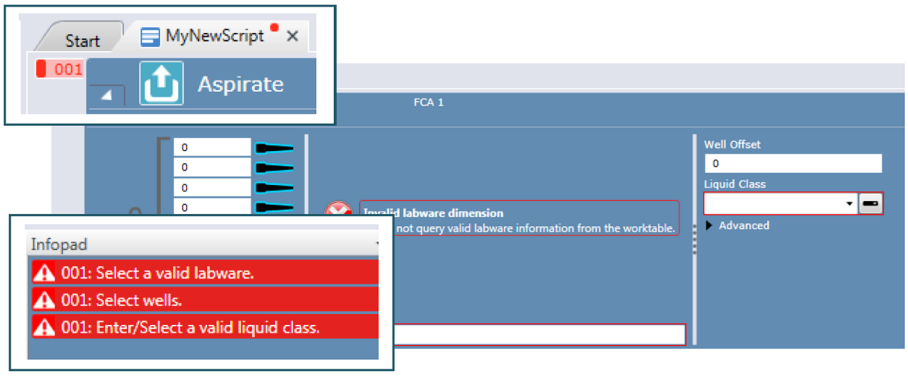
Breakpoint



Context Check

Each time you insert a new command in your script or edit an existing command, FluentControl checks whether the script contains errors. Faulty commands are flagged red. The Infopad will give more specific information or hints on the error. The script cannot be run.

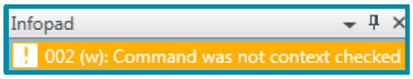
Auto-context check can be disabled from the Run menu. If the script contains errors, the faulty command(s) are not marked red and you can execute a script after confirming a warning that it was not verified.



If your script contains conditional jumps or branches, context check only reviews the parts of the script which are executed with the default values of the variables. Target script lines that could not be context-checked will be made visible with a yellow flag and a warning in the Infopad. The worktable will only show a warning.

The sections of the script skipped by context check have to be separately reviewed. To accomplish this, temporarily modify the values of the variables to ensure all script branches are verified.





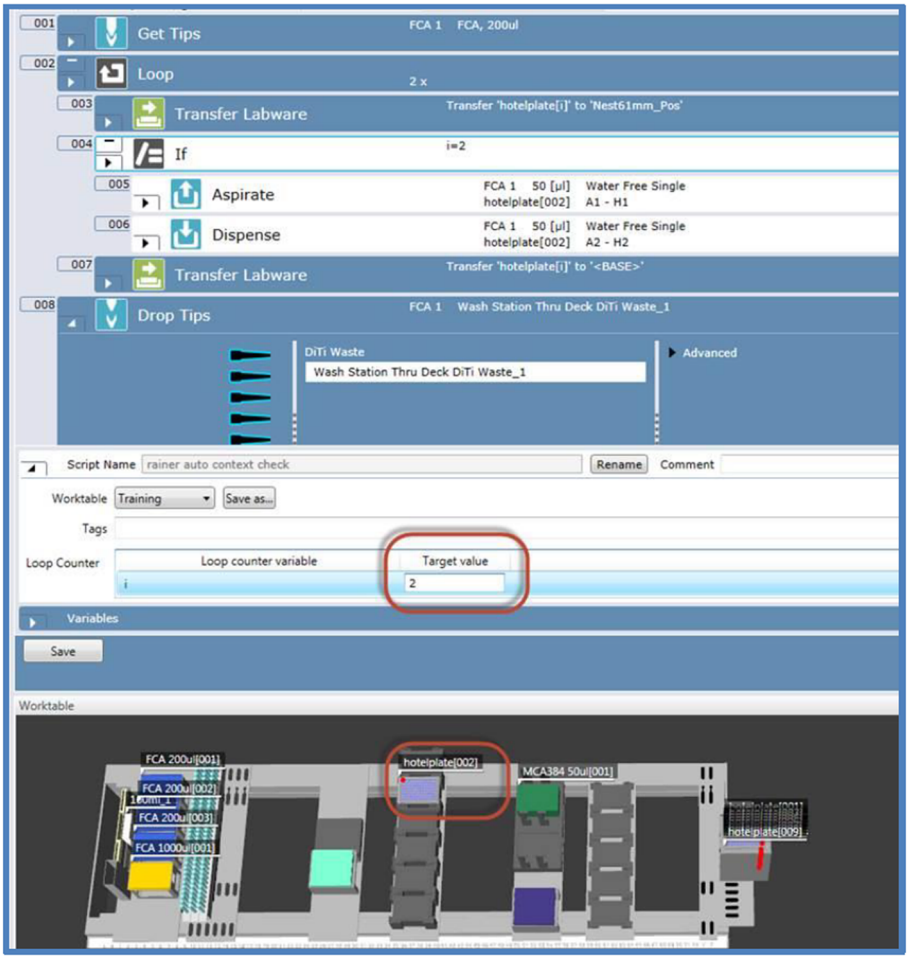
Please note that if the non printing characters (ASCII characters from 0 to 31 except 9, 10 and 13) are used in the input fields there will be no context check error shown. The non printing characters will be ignored and they will not be saved.

Loop Iteration Simulation

FluentControl is object oriented. The worktable will always show you the current Labware Position; at runtime and also within the script editor as you click through the script lines.

The script option Loop Counter will help you outlook the current Labware position during every cycle of the loop. The Loop counter variable parameter corresponds to the loop counter for the selected loop in the script, whereas the Target Value parameter corresponds to the loop cycle to be shown on the worktable.

The example below shows a loop with loop counter “i” where the worktable is simulated for the second cycle of the loop.

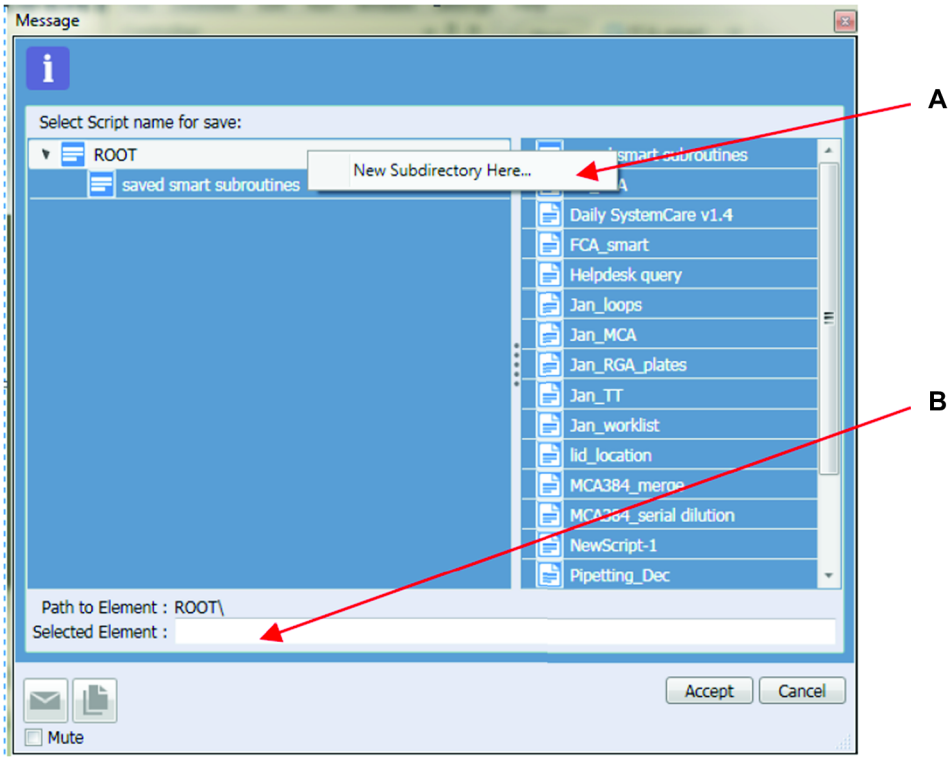


5.6.4 How to Save a Script

Once you are done editing the script, it is necessary to save it. You may rename the script by selecting ‘rename’ which will allow you to then modify the ‘script name field’. Then select save.



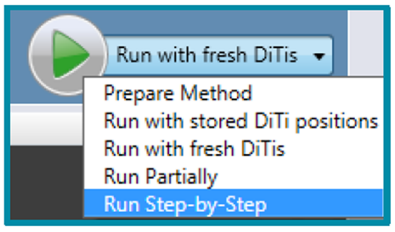
In the case you wish to make a copy of the existing script, choosing ‘rename’ will only overwrite the name of the current script. It does not create a second instance! To create another script select from the File menu ‘save as’. The below user message box will appear:



Location	Description
A	If you wish to save the script to a subdirectory of the root folder, right click on the 'root' header to have the 'new subdirectory here' option populate.
B	Write the new desired script name in the Selected Element box.

5.6.5 How to Run a Script

Scripts can be run independently or combined into a method.



Prepare method



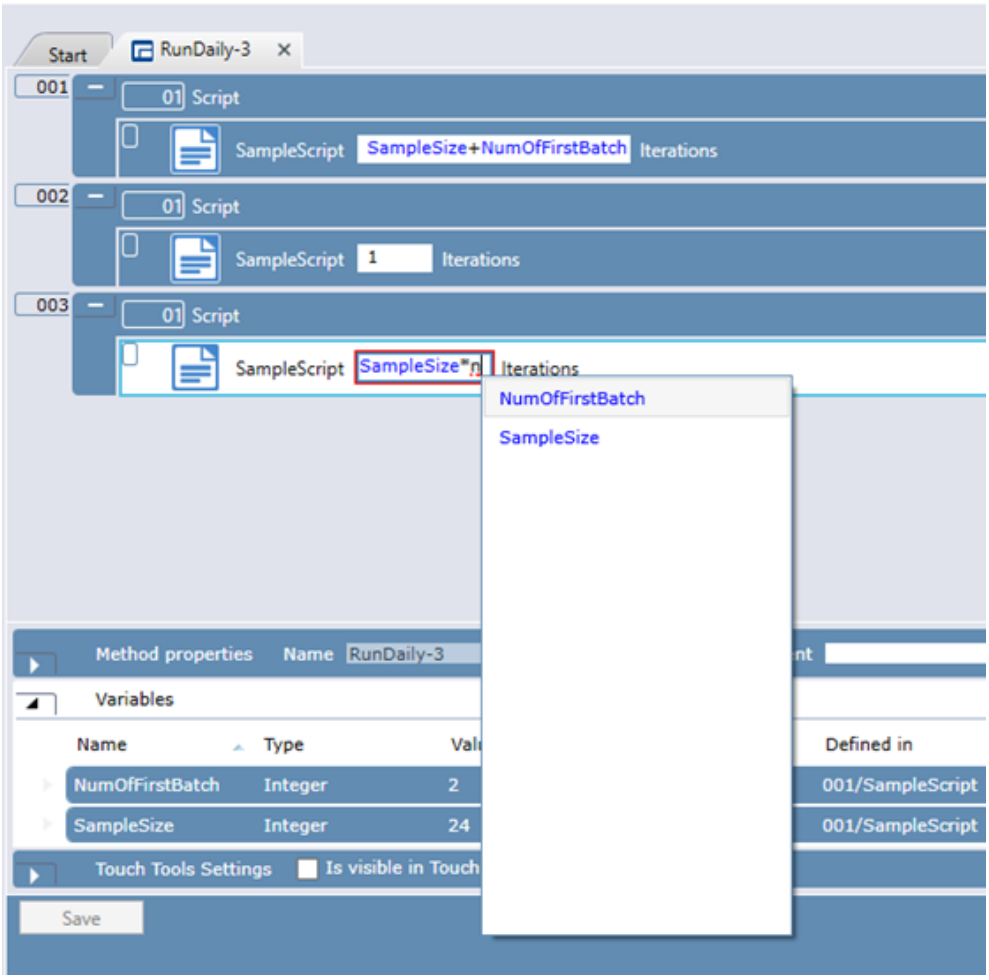
With this option you are shown the method editor tab.

A method consists of one or several different scripts using the same worktable. For every script you can specify how many times it is run (number of iterations). The number of iterations can be defined as an integer, variable with scope Run or expression.
If a variable or expression is used, the number of iterations that shall be executed is calculated just before the batch is started. If the calculated value is not an integer, the default value '-1' will be used. Expressions Floor() and Round() may be used to ensure that an integer value is used. The Trace View and Audit Trail record the number of iterations started for each batch in the method.

The method editor offers additional settings: “Reset all labware positions on Worktable”: After the method is finished the Labware (plate and DiTi) status will be reset. “Preserve Disposable Tips Information from last run”: The plate positions will be reset but the DiTi status will be stored. With the option “Preserve all Labware positions from last run”; FluentControl will update the worktable with Labware information after the run. Labware information includes 1) Preserve the last DiTi position and continue from there the next time a script is started; 2) Update Labware positions on the worktable to the current locations after the run, and 3) Preserve Labware barcodes.

You can make the method visible on TouchTools. Note: individual scripts cannot be made visible on TouchTools, only methods, and by default not all methods will appear on TouchTools.

In the Variable section you will find the list of all variables defined in the individual scripts with scope Run, Process and Iteration. The default values can be modified, allowing different Methods to run the same Scripts and Processes with distinct settings.



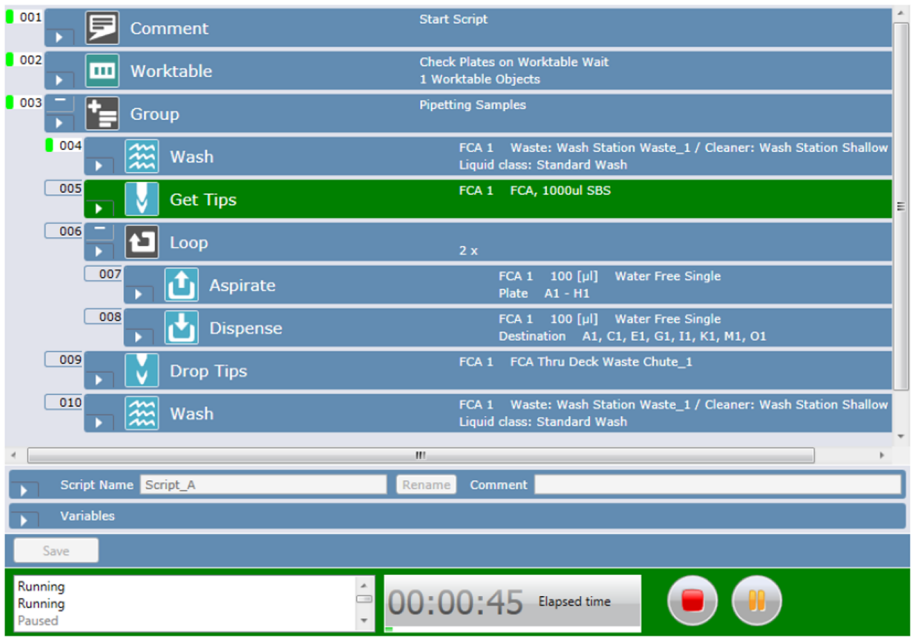
Run a Script

In contrast to running a method, running a script will execute a script directly, and just once.

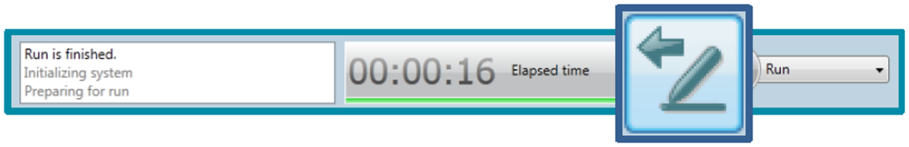
- **Run Partially:** executes the current selected script lines. To specify the command lines to run, highlight them in the script editor
- **Run with stored DiTi positions:** resets all Labware positions on the worktable but preserves DiTi information from last run, including the positions of tip trays, boxes, and layers in nested tip stacks. This applies to FCA, MCA96 and MCA384 tip types.
  The DiTi position is preserved between different scripts if they use the same Worktable or if the Labware label is the same.
- **Run with fresh DiTis:** resets all Labware positions on the worktable and stored disposable tip information from the last run. DiTis are picked up from the first position in the selected DiTi box assuming that the box is full.
- **Run Step-by-Step:** the execution of the script is paused before each step.

Running

When a method is run, the currently running script is shown. The current step being executed is highlighted in green, whereas already executed steps carry a green flag.



After a run is finished switch back to the edit mode by clicking the “Switch to edit mode” button:



Active Stop & Resume

Opening the door or pressing the stop button will trigger an ActiveStop. The system will stop as fast as possible. Some actions will be allowed to finish (e.g. Pickup Tips, Drop Head Adapter). Most commands will stop immediately, including Aspirate & Dispense commands. If a run has been interrupted by an ActiveStop, the run will be “Completed with Warnings” as incubation times or pipetting steps may be influenced. Reports will also indicate which pipetting steps were interrupted.

After the Arms have stopped, they will enter Zero-G mode.

Upon pressing retry or continue, the system will ensure the doors are closed and will resume whatever steps were interrupted or continue with the next action as appropriate.

ActiveStop will not pause the system if there is no mechanical action of the arms. If the front door is not closed by the time the next action starts, the system will pause at that point and arms will not move. This means, you can open the door during an incubation step and if you close it before the next step, the system will simply continue without interrupting the run.

Door locks are optional and can be configured (or removed from the configuration) by a Tecan Field Service Engineer. Door sensors are always active.

5.7 Modules

5.7.1 General

Module in all its creation behavior is the same as the script. The only difference between module and script is that a module can be inserted in the script and will be treated there as a synchronous subroutine. To run a Module asynchronously, use it in a Subroutine command.

5.7.2 How to create a Module

Module creation is the same as the creation of the script. It can be created from the Scripts or Modules section of control bar menu. To designate a script as a module, check the checkmark "Use as module" and save the script. If the module is created from the Modules section of control bar menu, this checkmark will be set by default.

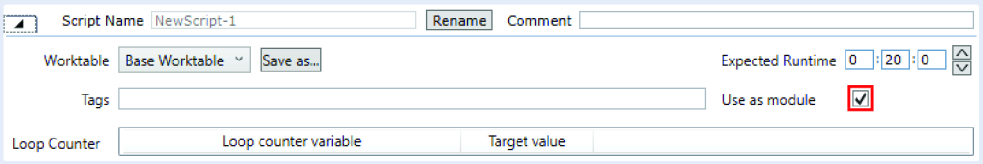


Fig. 37: Creating a module

5.7.3 How to use a Module

A Module can be dragged from the Controlbar and dropped into the script editor where it will appear as a Module programming statement. It will automatically contain the name of the called module and the list of parameter variables including their default values that should be transferred into the module. A tooltip function provides the query text information of each parameter variable, if that has been defined, when hovering over the variable. See section“11.11.36 Module” [▶ 384] for more details on the Module Command.

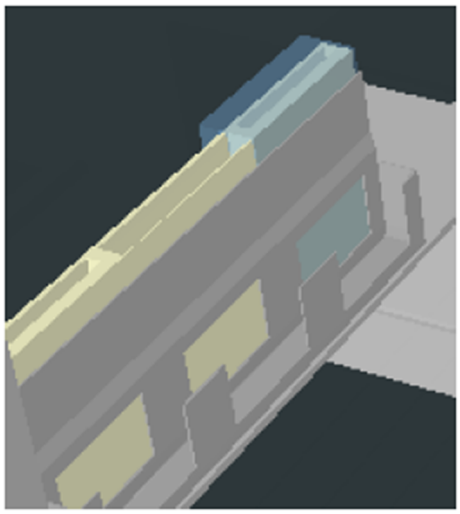
5.7.4 Other Operations with Modules

Modules can be turned in scripts if the **Use as module** checkmark is unchecked, and the Module is saved. Note: If the module was used in the script and then became a script, the caller script will have a context check error.

5.8 PathFinder

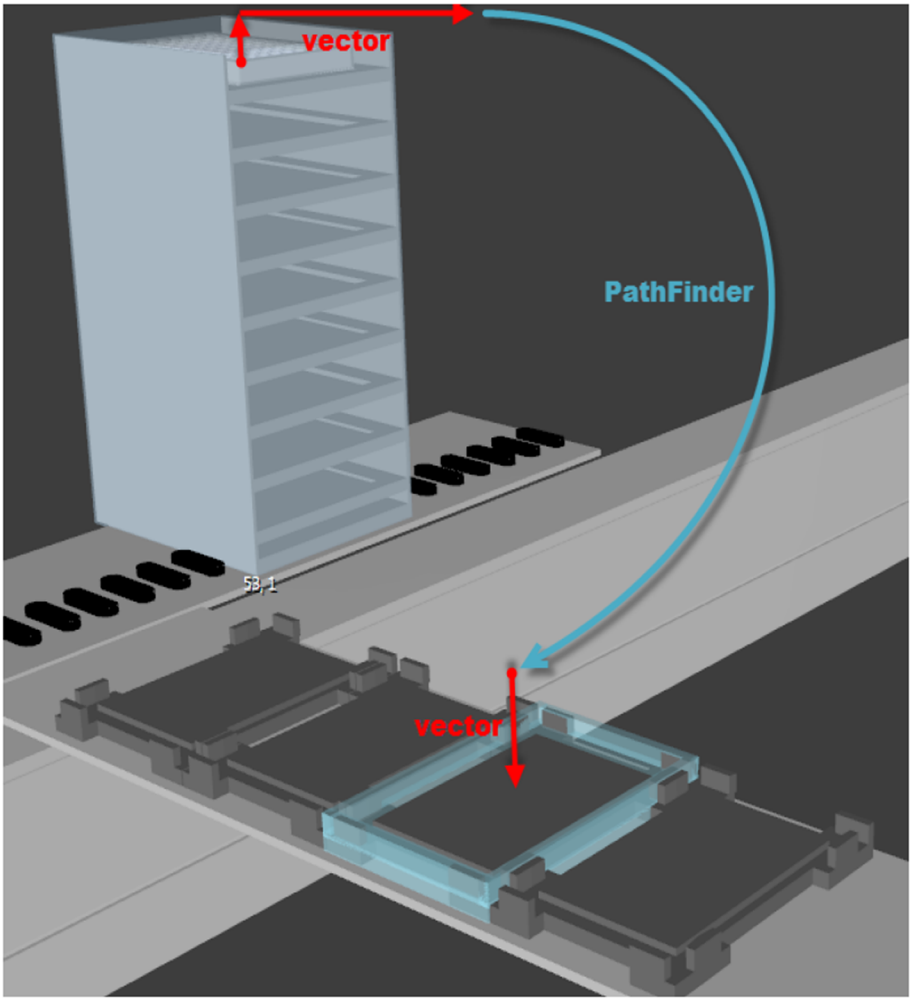
Every object, static objects like the frame of the instrument, the worktable, carriers on the worktable, labware on carriers as well as moving objects like the arms, the mounted tips, MCA adapters, have a bounding box definition which defines the outer dimensions. PathFinder calculates a way for the arm to move from a starting point to an end point making sure there is no collision; by ensuring that no bounding box of the moving parts interferes with any other bounding boxes. If movement in a direct line is not possible, Pathfinder will move tips, grippers or MCA adapters up or even guide them around objects if necessary. Pathfinder will also move other arms out of the way and ensure they don't collide with any object while moving. This is called an evasion path and if the pathfinder can't find any way

to move the arm it will stop and issue a warning to the user. Consider the bounding box as a “no fly zone”.



RGA

For the RGA the Pathfinder calculates the route from the current position to the safe waypoint of the next target vector. However, movements from the Safe Position of a vector to the End Position of the vector are not controlled by PathFinder. The RGA executes the defined vector from safe to end. While the RGA movement is controlled by a vector the arm is able to move inside bounding boxes. This means potential collisions are not controlled and prevented by Pathfinder while the RGA move is controlled with a vector.



Hence for a plate transfer, the movement is divided into a vector controlled part to fetch the microplate and move it to the Safe position. At the Safe position the bounding box of RGA including its gripped plate must be completely outside of the bounding box of the carrier. At this point the PathFinder takes over control of the movement until the Safe Position of the target carrier. Here again bounding boxes of RGA and plate are not allowed to violate the bounding box of the carrier as long the movement is under PathFinder control. Otherwise PathFinder errors such as for example “The arm is unable to move to the specified end position” are thrown. The last part of the movement from Safe to End position on the target carrier is controlled by the corresponding vector.

FCA, MCA96 and MCA384

For the pipetting arms the Pathfinder calculates the route from Z-travel of the current Labware to the Z-travel of the destination Labware. Once at destination at Z-travel, the arm moves along the Z-axis as defined in the MicroScript being executed, with Pathfinder not active. This means potential collisions are not controlled and prevented by the Pathfinder.

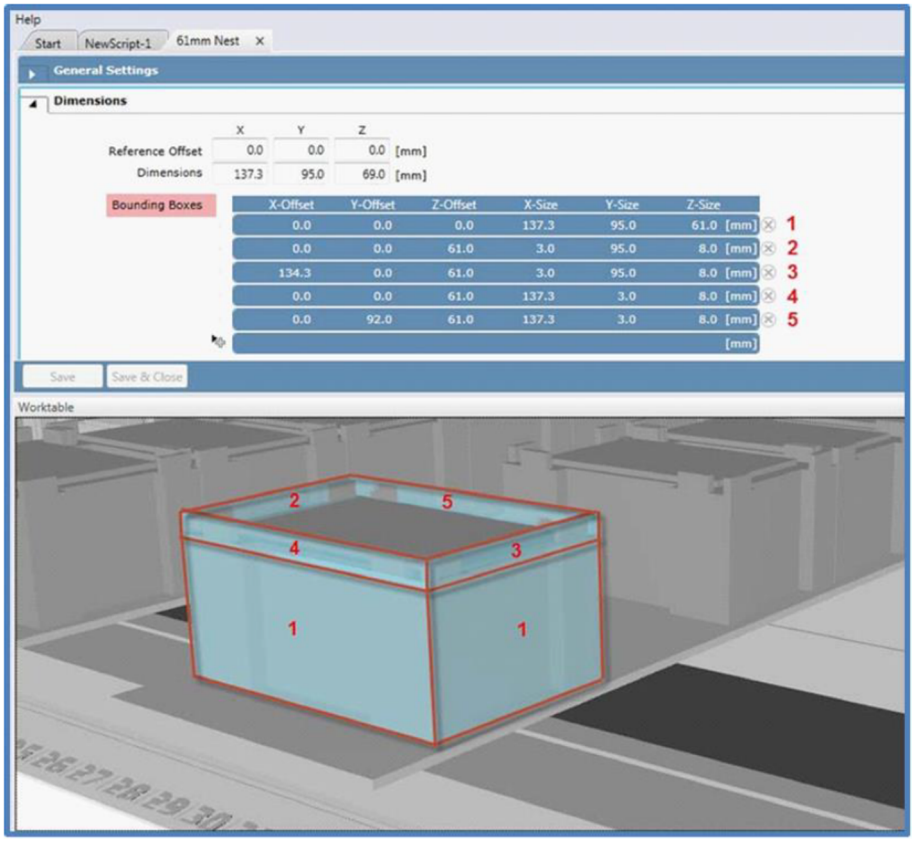
For example the collision of the Tip adapter with the tube rim, if the tube length is larger than DiTi length.

During configuration of a Fluent, please verify all desired movements are physically possible. PathFinder will do its best to move arms without issues and calculate problems before they arise. But there is no software feature that can make a Flexible Channel Arm with a mounted 10µl Tip reach the bottom of a 100ml Trough before the ILID block would touch the top.

Cannot start from colliding position or cannot move to a colliding position. This is a PathFinder error message. It does not mean the arm physically collided. A colliding position means that PathFinder believes the arm is inside of a bounding box of another object at the start of a move or would have to enter a bounding box at the end of the move. The message is giving hints about what objects are causing the issue (e.g. The Left Gripper Finger has an issue and is “colliding” with a hotel).

Dimensions (carrier)

Definition of bounding boxes to define an object.

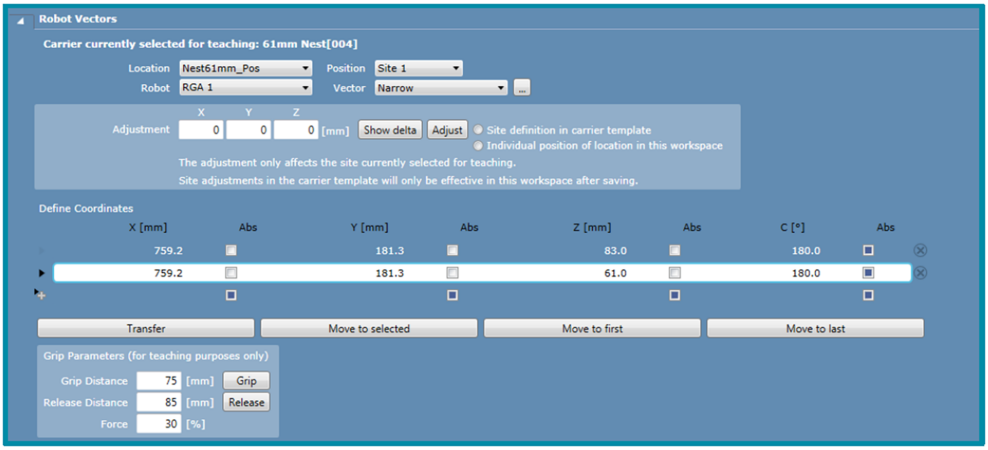


Robot Vectors

Robot vectors define how to approach a site of a Carrier within the bounding box. Changing these is only needed for custom Carriers and for some non-grid based devices: The Fluent is teach free and non-custom carriers have predefined vectors in the software.

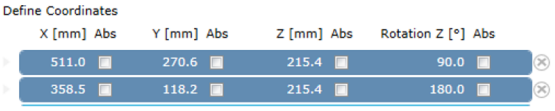
Hard crashes can misalign an arm. If after a hard crash you are no longer able to pipette or transport plates: DO NOT attempt to teach the carrier/labware (e.g. 7mm Nest). If "everything" worked until a crash the appropriate service actions must be run in the service software to re-align the arm.

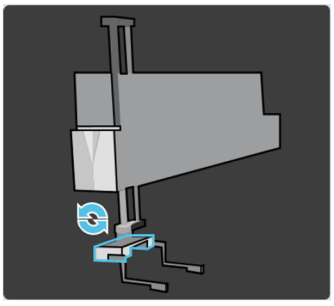
A vector consists of several waypoints. The button Transfer is used to store the current (physical) position of the arm as a waypoint. The three move buttons are used to move to one of the waypoints of the vector.



To create a custom vector click on the button next to the vector dropdown list, define the coordinates for each waypoint and save it. The new vector will then be available to select in the Execute Single Vector Command. Note, that custom vectors are always associated to the carriers where they were created.

Vectors are based on the plate position, not the arm position. This supports different finger types with no need to define a new vector.





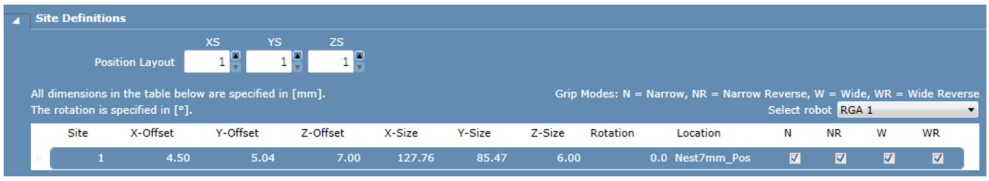
The above two points only rotate the gripper head if eccentric fingers are mounted. Physically these two coordinates are the same in X, Y and Z. The gripper would rotate 90° around Z.

The easiest way to program these coordinates is to put the arm in Zero-G, gently move the arm by hand to the final position, and use the transfer button. FluentControl will do all the calculations necessary. To teach a position while holding a plate, use the “Fetch Plate for Teaching” direct command before entering Zero-G mode.

Site definitions

A site is a defined space for labware. Properties of the sites are defined in the site definition window, including number of sites per Carrier and their sizes as well as which kind of vector is used to approach the sites (N, NR, W, WR). All grip modes are pre-defined and allowed on each site. The default Transfer Labware command

supports all grip modes. The priority however is given to the narrow vector. If you want to change this, only activate the grip mode you want to be used.



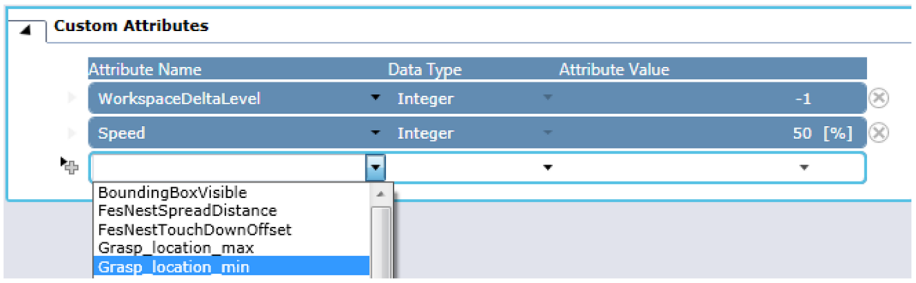
Site coordinates are relative to the carrier.

Allowed Labware types and allowed locations

These two sections contain information on which Labware is allowed on the Carrier, and on which location the Carrier is allowed.

Custom attributes

This section contains attributes, which are settings to define characteristics of Carriers (like is it a waste location), or Carrier-specific settings (e.g. speed for RGA movements from/to this carrier, whether to automatically remove a lid when the microplate arrives at this Carrier, and where to store it and many more).

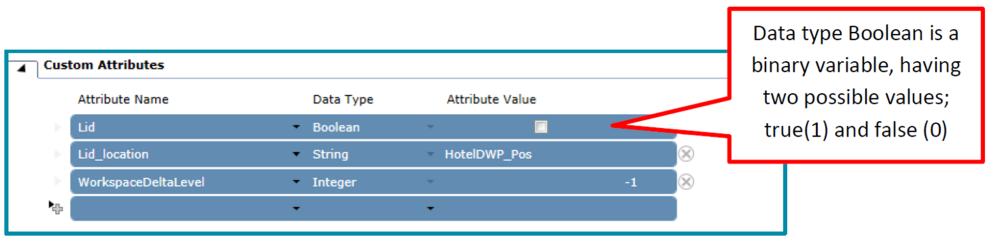


5.9 Lid Handling

In FluentControl lid handling is executed automatically.

First you need to let the software know the labware can have a lid. This is done in the Labware editor under “Robot Specific”. However, it does not yet mean the plate has a lid, this has to be specified in the Labware group editor (pictured above).

The next step is to define on which carrier a microplate with lid is allowed and on which it is not. This is done by activating the attribute **Lid** and **Lid_location** in the carrier editor under **Custom Attributes**. If the attribute Lid is set to false (i.e., unchecked) it means whenever a plate with a lid arrives on this carrier, the lid will be removed. **Lid_location** defines where the lid is stored.



Moreover, since in FluentControl the lid itself is a labware and a microplate is an allowed location where it can be placed, you can directly handle the lid transfer with the **transfer Labware** command outside carrier attributes. This is done by setting the Labware attribute **LidHandlingManual** to true.

How to Change Labware/Carrier attributes on the fly

FluentControl allows you to change Labware/Carrier attributes on the fly. The following example shows how to switch from automatic to manual lid handling within the script. Refer to “Functions andOperators” [442] for details about FluentControl functions.



Don’t forget to use another **Set Variable** command once manual lid handling is no longer required and you would like FluentControl to take over and move the lids according to the Labware attributes.

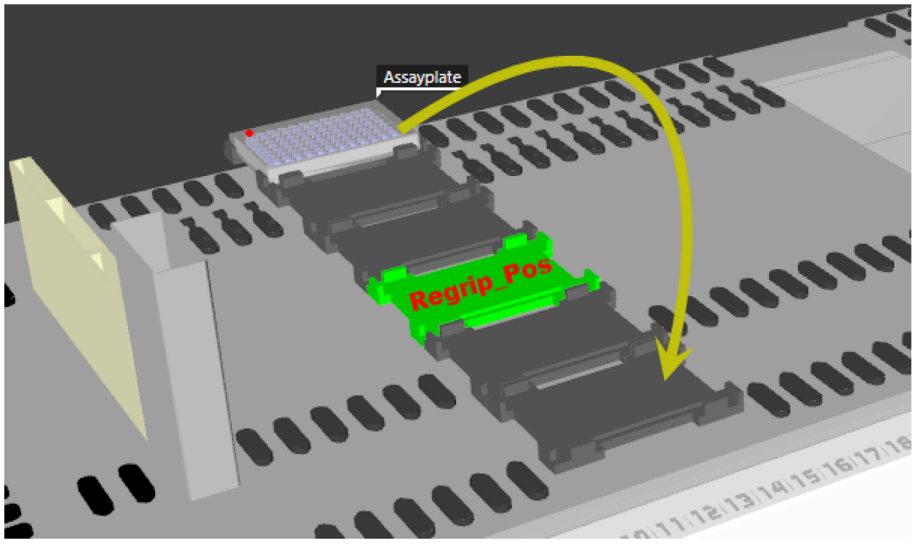
5.10 Regrip

A Regrip station will automatically be used if the source and destination location do not share an allowed vector.

PathFinder does not have any impact on the determination to use a Regrip station. First the RGA driver reads the carrier definitions, figures out which vector to use and whether it has to be offset (e.g., DWP) and then passes the move to PathFinder.

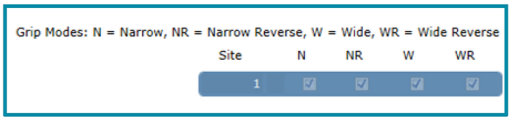
Example

Move a plate from site 1 to 6 and maintain plate orientation

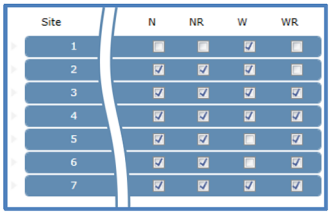


The 6x 7mm nests sit on a deck segment, the 7mm nest allows all vector modes, whereas the deck segment allows different vectors for different sites, based on what is physically accessible. The **Transfer Labware** command picks which vector to use exclusively from the combination of all carriers' allowed vectors under the labware.

7mm Nest

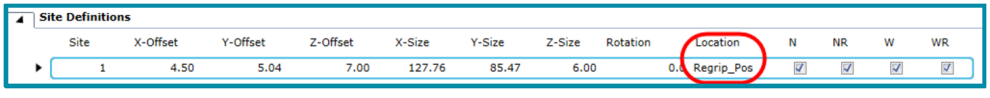


6 Landscape 7mm Nest



Site 1 can only be reached in Wide, whereas the Wide orientation is not allowed on Site 6. To move from Site 1 to Site 6 a Regrip is needed.

There are pre-defined Regrip nests in the FluentControl database, the "7mm Regrip Nest" and the "61mm Regrip Nest", They are physically the same as the 7mm Nest and 61mm Nest, respectively, but are defined as a Regrip location (Location "Regrip_Pos" defined in the Site Definitions section of the carrier editing tab). Requisite for a Regrip station is to allow all grip orientations.



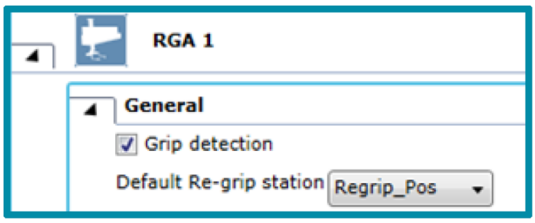
The Transfer Labware command will only use a combination of the vectors:

- Narrow
- Narrow Reverse
- Wide
- Wide Reverse

The order listed above is the order of preference that will be used if more than one combination is feasible.

The grip determination does not check if a vector is defined. It also does not check if it is physically reachable. Undefined Vector & “Colliding Position” are fatal run time errors

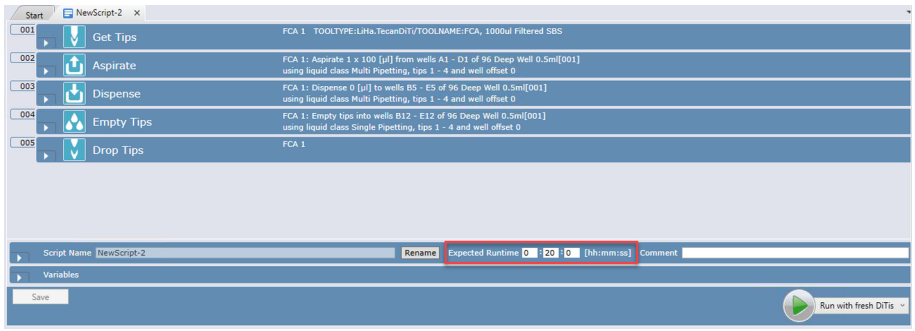
Custom Regrip location names can be configured in the RGA driver configuration tab. The name in the RGA configuration must be a location name on the worktable. By default the RGA configuration matches the Regrip Nests.



5.11 Runtime

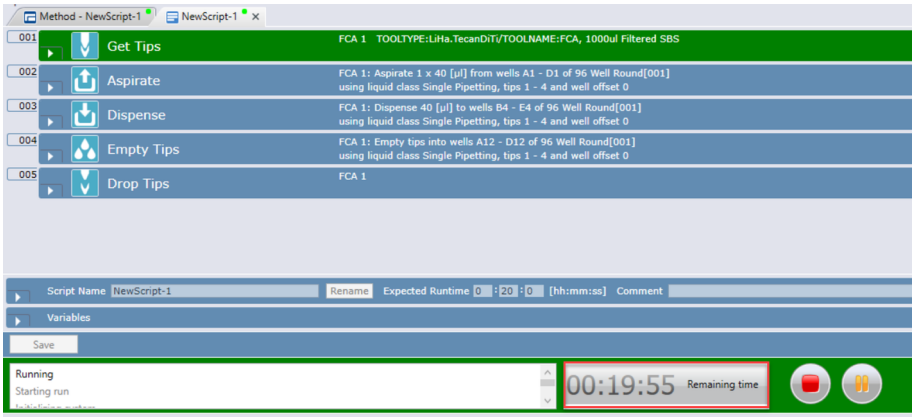
Expected Runtime

The Expected Runtime is defined in the Script Editor for each individual script. It determines the initial Remaining Runtime displayed in the Runtime Controller (refer to section “Runtime” [▶ 87]).
By default, this Expected Runtime is set to 20 minutes.



Remaining Runtime

The Remaining Runtime can be viewed in the Runtime Controller while a script is running.



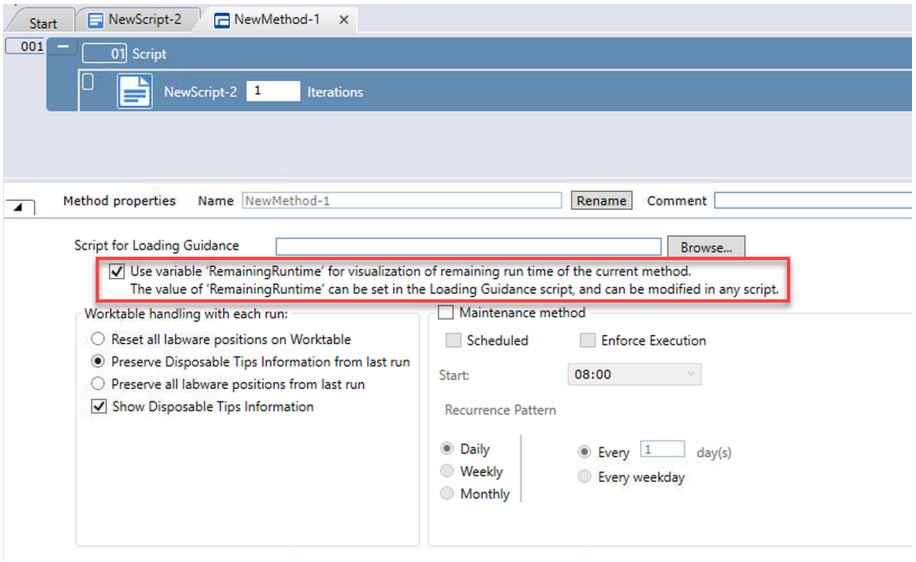
The Remaining Runtime counts down from the Expected Runtime.

Methods and Runtime Calculation

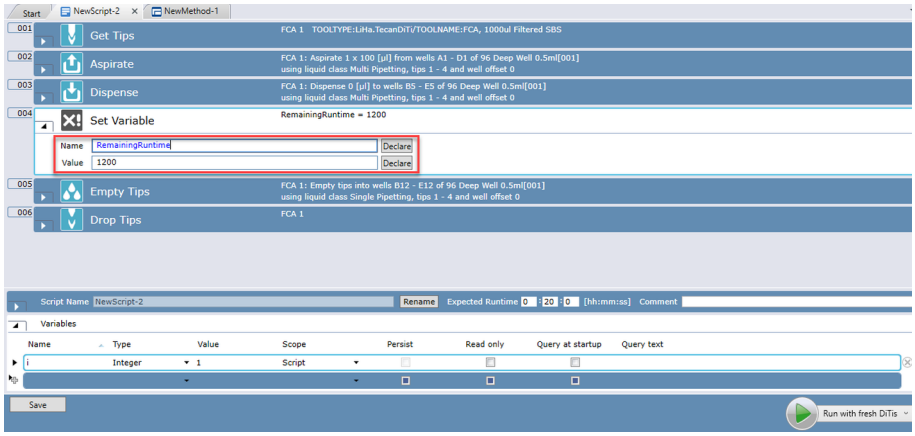
Methods do not allow direct input for an expected runtime. Instead, the Remaining Runtime for a Method is calculated as the total sum of the expected runtimes of all scripts within that Method.

Dynamic Remaining Runtime

Dynamic Remaining Runtime provides additional flexibility by allowing scripts to override the predefined Expected Runtime during execution. To enable this feature, select the Remaining Runtime checkbox under the Method properties.



When it is enabled, the Remaining Runtime value can be dynamically adjusted during script execution by setting the RemainingRuntime variable in the Set Variable script command.



Benefits of Dynamic Remaining Runtime

The Dynamic Remaining Runtime ensures a more accurate and flexible estimation of the script completion time by allowing real-time updates to the Remaining Runtime based on current conditions or specific points within a script.

6 User Administration

To activate user management in FluentControl **select Settings > User Administration... and click Activate in the User Administration tab.**

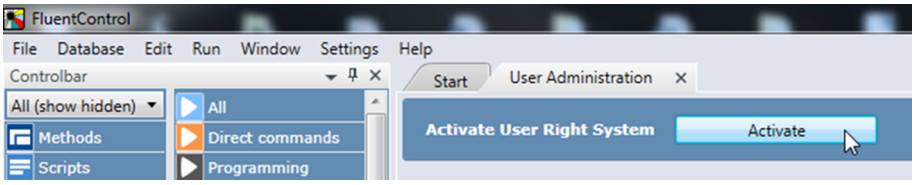


Fig. 38: Activate

You can only change user settings if you have been given FluentControl administrator access rights.

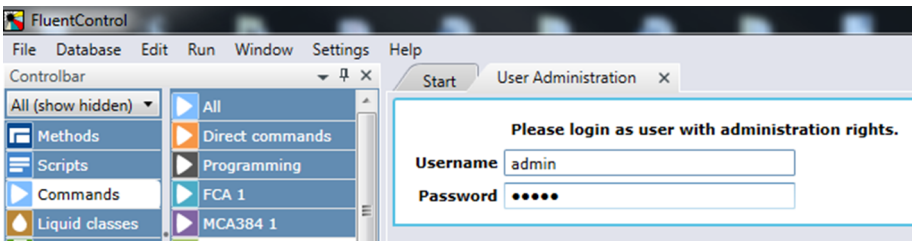


Fig. 39: Log in as administrator

In User Administration you can change the settings for log in, passwords, notifications, user roles, method permissions, and access rights. You can also activate the use of electronic signatures.

User Administration does not rely on the MS Windows access rights and log in mechanisms which have been configured for your PC and/or for your local network. The Tecan Identity Access Management is a central software component which handles user log in and user authorization for all Tecan software components. It also provides facilities to log out users after a specified period of inactivity and to send warnings by e-mail to a specified person (usually the application administrator) if a user is temporarily locked out due to too many failed logins. Furthermore, users can be deactivated by the application administrator.

The Tecan Identity Access Management thus provides an additional security barrier against unauthorized use of the Tecan software. FluentControl writes log files and can create data files. The Tecan Identity Access Management provides its own audit trail logs. It is important to correctly configure the MS Windows access rights to the respective directories on your PC and/or local network to prevent unauthorized access to and tampering with this data.

6.1 Main Features

- All changes relating to user management require FluentControl administrator access rights.
- All changes relating to user management are recorded in the user management audit trail.

- User accounts are automatically disabled following a configurable number of consecutive failed attempts to enter the correct user name and password combination. The disabled account can only be reactivated by the FluentControl administrator.
- FluentControl does not allow more than one person to use the software simultaneously.
- Users can log out without leaving FluentControl or interrupting a running pipetting script or process. The same or a different user can then log in to FluentControl while the script or process is still running.
- Users are automatically logged out after a specified time during which no keyboard or mouse-click activities have been detected.
- User passwords expire after a configurable number of days. The user must then choose another password.
- Passwords must contain a specified minimum number of characters.
- Passwords can be forced to contain numeric digits.
- User names are unique within the system and cannot be assigned more than once.
- The FluentControl administrator provides a temporary password when setting up a user. When the new user first logs on to the system, he is forced to change the password immediately. This ensures that no one other than the individual user knows the particular user name and password combination.
- If a user forgets his password, the FluentControl administrator does not have access to the user's password. The FluentControl administrator is only able to reset the password and provide a new temporary password.
- If the FluentControl administrator forgets his password, no one has access to the password.
- A specified person (typically the FluentControl administrator) can be automatically notified by e-mail in case of potential security threats (e.g., a user account has been temporarily locked for a configurable time after a configurable number of consecutive failed login attempts).

6.1.1 Login Dialog

The last 5 users that have been logged in to the application are listed according their last login in the login dialog.
A click to the according user button fills out the the user name. Only the password is required to complete the login.
Other users need to provide the user name and the password in the corresponding input fields.

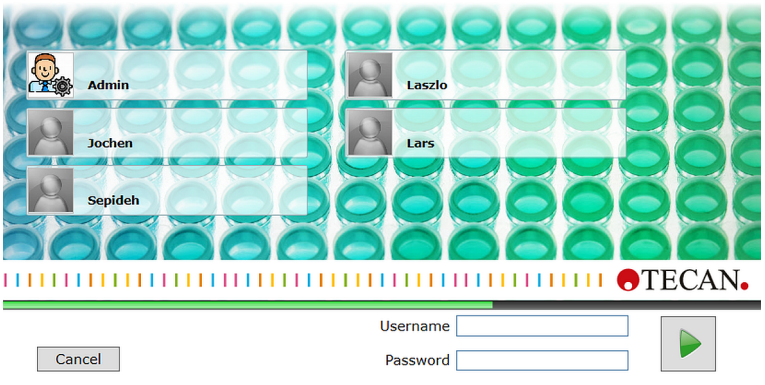


Fig. 40: Login Dialog

6.1.2 Audit Trail

Are you in a regulated industry where audit trails matter?

All changes relating to user management are recorded in the user management audit trail. A log file is written to the following directory:
`%ProgramData%\Tecan\MAP.IAM\DataStore\MAP.Services.Logging.Service\LogFile`

The file name is `LogFile YYYY_MM-DD HH.MM.SS.`, where `YYYY_MM-DD` is the file creation date and `HH.MM.SS.` is the file creation time.

6.2 Login Settings

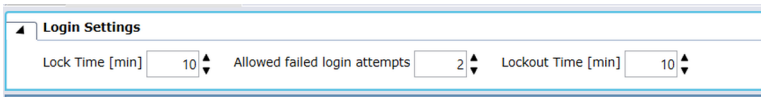


Fig. 41: Login Settings

Lock Time [min]

Users are automatically logged out of FluentControl after a period of inactivity. Specify that period of time. Logged out users must log in to FluentControl again to regain control of the program (range: 2 to 1440).

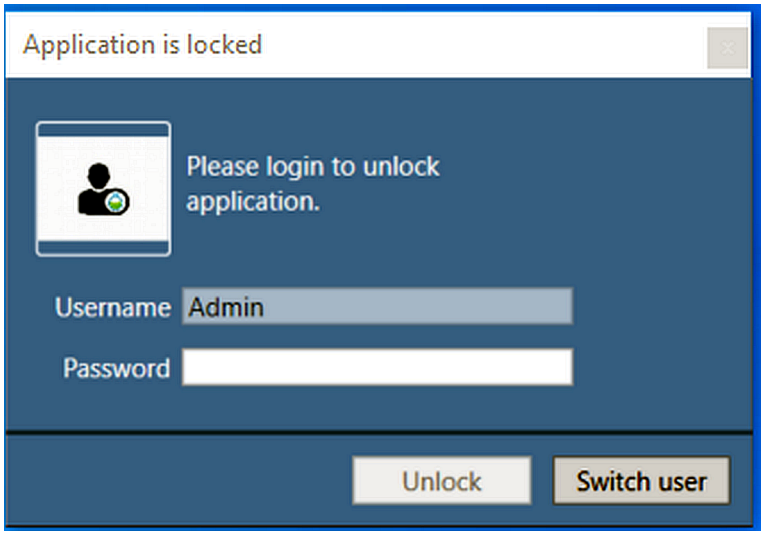


Fig. 42: Log in

Allowed failed login attempts

A user is temporarily locked out for a configurable time after a configurable number of consecutive failed attempts to enter the correct user name and password combination. Specify the maximum number of unsuccessful logins (range: 1 to 10).

Lockout Time

The duration in minutes for which a user is locked out after exceeding the configured number of failed logins.

6.3 Password Settings

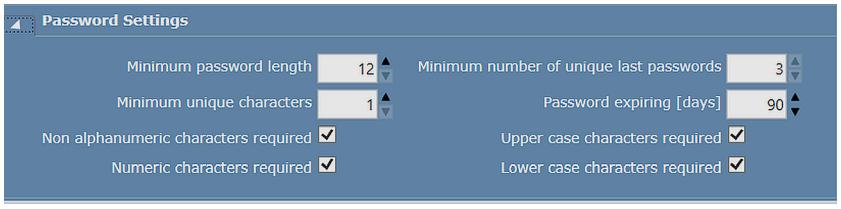


Fig. 43: Password Settings

Minimum password length

Passwords can be forced to contain a specified minimum number of characters. Specify the required number of characters.

Minimum unique characters

Specify the number of unique characters, that the password must contain.

Non alphanumeric characters required

The password requires at least one non alphanumeric character.

Numeric characters required

The password requires at least one numeric character.

Minimum number of unique last passwords

Specify the number of previously used passwords that are not allowed to be reused.

Password expiring [days]

User passwords expire after a configurable number of days; the user must then choose another password. Specify the number of days.

Upper case characters required

The password requires at least one upper case character.

Lower case characters required

The password requires at least one lower case character.

6.4 Notify Settings

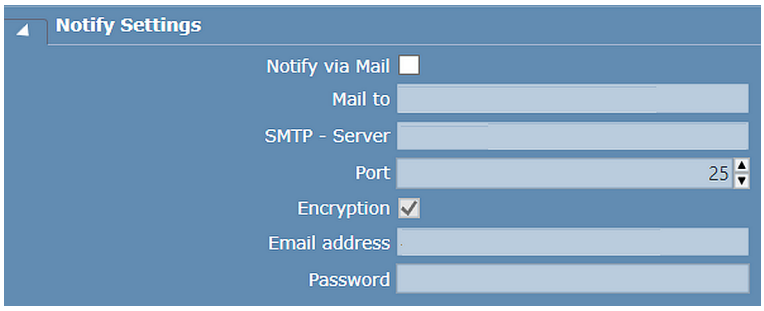


Fig. 44: Notify Settings

A specified person (typically the FluentControl administrator) can be automatically warned by e-mail in case of potential security threats (i.e., a user has been temporarily locked for a configurable time after a configurable number of consecutive failed login attempts). This panel is used to configure the settings for the e-mail warnings.

Notify via Mail

Check this checkbox if you want FluentControl to send security warnings by e-mail. If you disable this feature, this does not delete the e-mail settings and you can re-enable them later if required.

Mail to

Specify the e-mail address of the recipient (e.g., the system administrator).

SMTP - Server

Specify the address of the SMTP server you want to use for sending the e-mails. You should normally specify the SMTP server address that is recommended by your e-mail provider or e-mail administrator.

Port

Specify the SMTP port number.

Encryption

Check this checkbox if encryption is required. This is normally the case.

Email address

Specify the email address for SMTP authentication.

Password

Specify the password for SMTP authentication.

6.5 User – Role permission

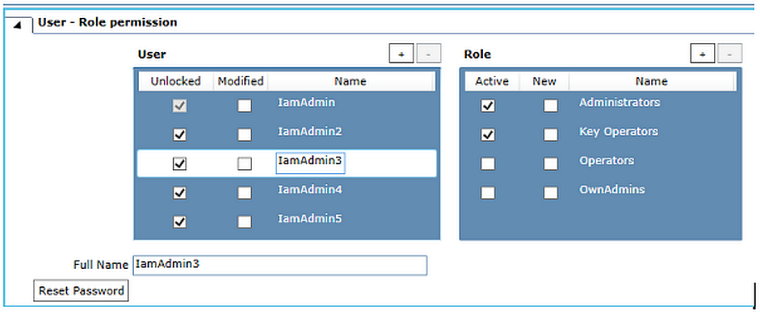


Fig. 45: User - Role permission

The **User** panel lists all users who are currently known to FluentControl. The **Full Name** of the currently selected user is displayed underneath the panel.

Uncheck the **Unlocked** checkbox to disable the log in for the selected user.

*Never uncheck all of the **Unlocked** checkboxes because you will not be able to log in to FluentControl again. Do not disable the FluentControl administrator login.*

The **Role** panel lists all available user groups and shows the groups to which the currently selected user belongs. Check or uncheck the **Active** checkboxes to change the group assignments as required. Users inherit the rights of the groups to which they belong.

Resetting a Password

The administrator can reset a user's password.

The administrator is not able to reset the administrator's password.
To change the administrator's password, the administrator must use the Change Password functionality. Refer to section "Changing Password" [▶ 129].

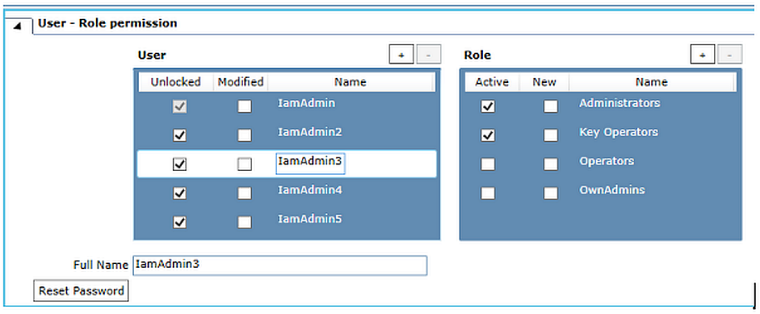
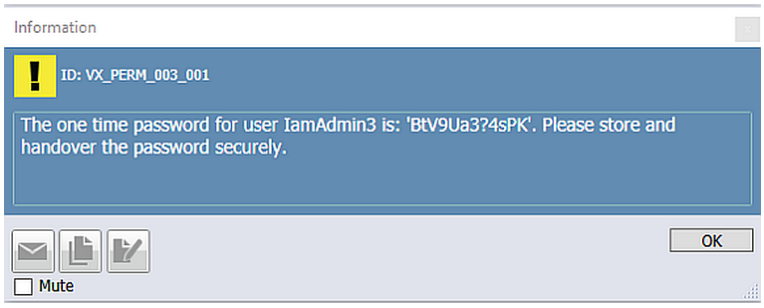


Fig. 46: Resetting a user password

The generated one time password is displayed in a dialog.



Adding a New User

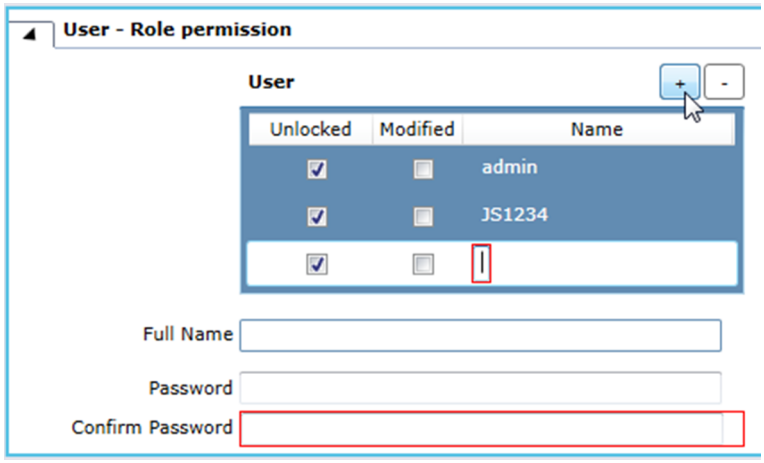


Fig. 47: Adding a new user

To create a new user account, proceed as follows:

1. Click **[+]**.
2. Enter a unique **Name**.
   You can neither change this name nor delete the created user account after clicking **Apply**.
3. You can enter the user's **Full Name**.
4. Enter and confirm a new password.
   The password has to comply to the rules that have been defined in **Password Settings**. This initial password is valid for three days. During this time the user has to log in and change his password or the account will be disabled automatically.
5. Click **Apply** to save your changes or **[-]** to abort.

Adding a New Role

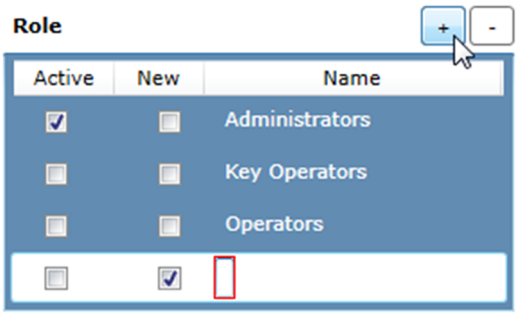


Fig. 48: Adding a new role

To create a new **Role**, proceed as follows:

1. Click **[+]**.
2. Enter a unique **Name**.
3. Click **Apply** to save your changes or **[-]** to abort.

6.6 Role – Method permission

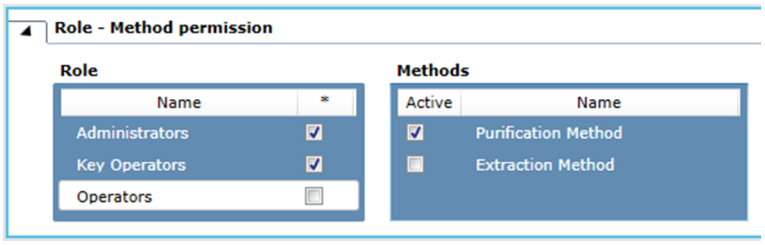


Fig. 49: Method permission

In **Method permission** you can define the methods that are available for a user role. However, what a user can actually do with a method is defined by the access rights of his user role. For more information refer to Role – Access Rights.

By checking or unchecking the * checkbox in the **Role** panel you can assign all methods to the selected role or deny access to all methods, respectively.

By checking or unchecking the **Active** checkbox in the **Methods** panel you can assign a specific method to the selected role or deny access to this method, respectively.

6.7 Roles and Access Rights

The **Role - Access Rights** menu lists all available access rights for the currently installed Tecan software components and shows the access rights that are assigned to the selected user role. Users inherit the rights of the role that has been assigned to them.

By default we differentiate between three available Roles:

- **Administrator**

- **Key Operators**
- **Operators**

The system allows to add new roles which can be a new operator, an operator with special rights etc.. The new created role appears in the menu **Roles**. For more information on how to create a new role please refer to “User – Role permission” [▶ 96].

You can assign **Access Rights** to the selected **Roles** by checking / unchecking the **Active** box in the menu **Access Rights**.

Access Rights

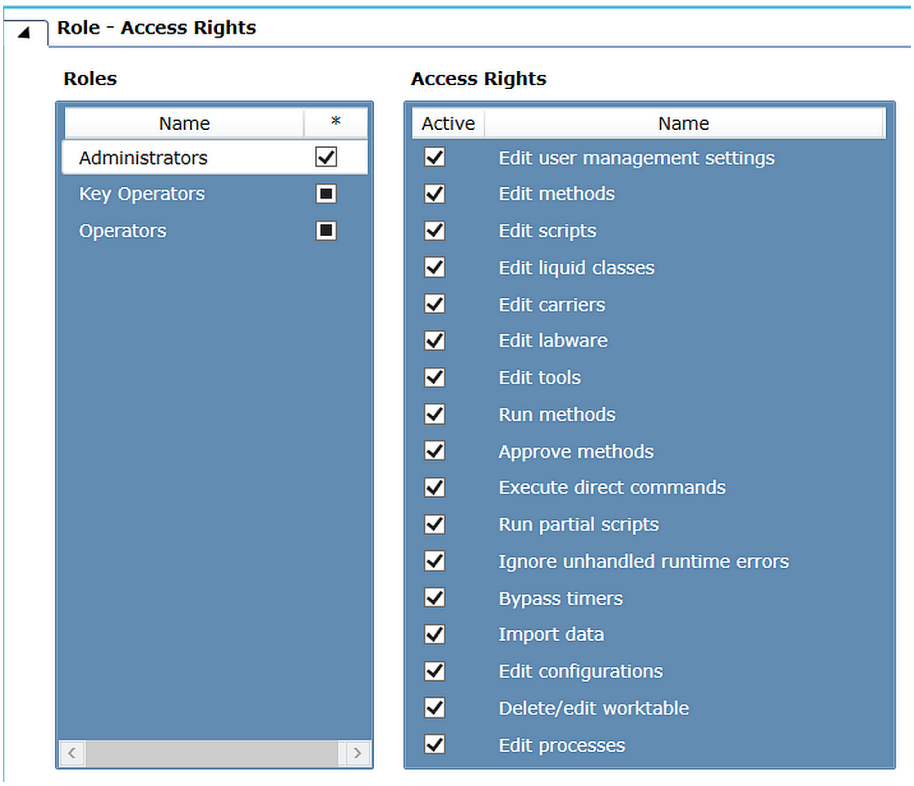


Fig. 50: Access Right Overview

Edit user management settings	The User Administration option in the Settings menu will be visible to users with this role. Roles with this access right can activate or deactivate the User Management System and edit any settings in the User Administration tab. For more information see Chapter 6 “User Administration” [▶ 90].
Edit methods	Roles with this access right can create, edit and delete methods. Roles without this access right may only view Methods.
Edit scripts	Roles with this access right can create, edit and delete Scripts and Modules. Roles without this access right may only view Scripts.

Edit liquid classes	Roles with this access right can duplicate, edit and delete Liquid Classes. Liquid classes with a lock symbol cannot be edited or deleted. Roles without this access right may only view Liquid Classes.
Edit carriers	Roles with this access right can duplicate, edit and delete carriers. Carriers with a lock symbol cannot be edited or deleted. Roles without this access right may only view carriers.
Edit labware	Roles with this access right can duplicate, edit and delete labware. Labware with a lock symbol cannot be edited or deleted. Roles without this access right may only view labware.
Edit tools	Roles with this access right can duplicate, edit and delete tools. Tools with a lock symbol cannot be edited or deleted. Roles without this access right may only view tools.
Run methods	Roles with this access right can run methods via the FluentControl user interface or the touch monitor. A method is only visible in on the touch monitor if the 'Is visible in Touch Tools' box has been checked in the method editor.
Approve methods	This right is required, along with the Edit configurations right, to activate the approval functionality in Configure System. Roles with this access right can approve Methods, Processes, Scripts and Modules. Roles with this access right can run unapproved Methods, Processes, Scripts and Modules. For more information on method approval, see section “Method Approval” [▶ 126].
Execute direct commands	Roles with this access right can run direct commands. For more information on direct commands, see section Direct Commands.
Run partial scripts	Roles with this access right can run partial scripts.
Ignore unhandled runtime errors	Roles with this access right can ignore runtime errors and continue the run without handling them.
Bypass timers	Roles with this access right can bypass timers.
Import data	Roles with this access right can import data into the FluentControl database by using the Import function in the Database menu. For more information on importing data, see section “Import Mode” [▶ 526].

Edit configurations	Roles with this access right can view and change any setting in the Configure System option in the Settings menu.
Delete/edit worktable	Roles with this access right can create, duplicate edit and delete worktables. Roles without this role may only view worktables.
Edit processes	Roles with this access right can create, edit and delete processes. Roles without this role may only view processes.

Administrators By default the **Administrators role** has all **Access Rights** active. Tecan strongly recommends not to change these settings for the **Administrators** role. Tecan also recommends to create at least two users with the **Administrators** role, in case one Administrators user locked out of the software unexpectedly.

Key Operators By default the role **Key Operators** has ALL access rights—except **Edit user management settings**

Operators By default the following access rights are assigned to the role **Operators**:

- **Run methods**
- **Execute direct commands**

6.8 Electronic Signature

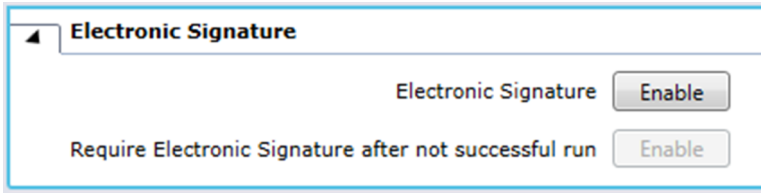


Fig. 51: Enabling electronic signature

Click **Enable** to enable **Electronic Signature**.

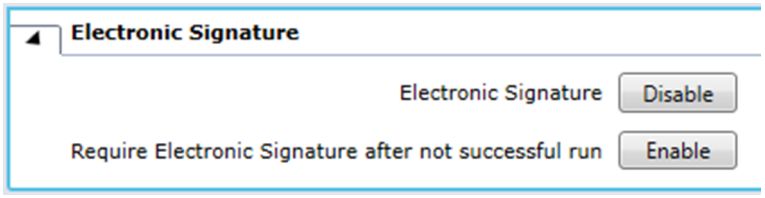


Fig. 52: Disabling electronic signature

If you want the user to sign electronically after each run that has not been successful click **Enable**.

When your electronic signature is required (e.g., for disabling the Electronic Signature) you need to enter a **Reason** and your **Password**.

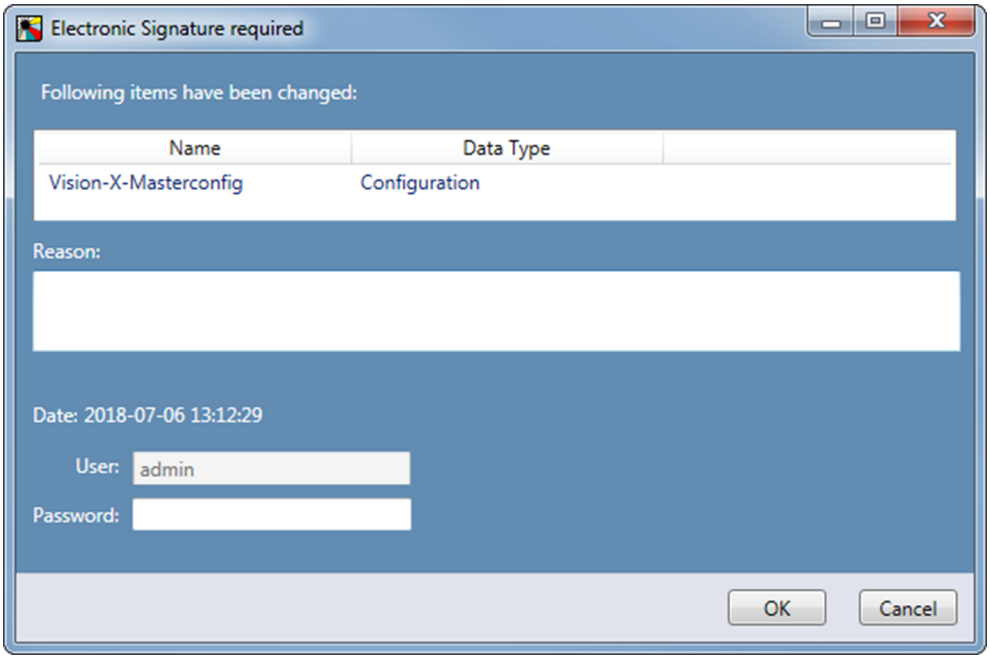


Fig. 53: Electronic signature

6.9 Deactivate User Management

To deactivate user administration select **Deactivate User Management** and click **Deactivate**.

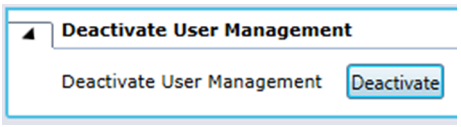


Fig. 54: Deactivating user administration

With user management disabled, it is not necessary to enter a user name or password when FluentControl is started, and all users will be given FluentControl administrator access rights automatically. Be aware of the security risk if you deactivate user management and make sure this is permitted by the security procedures of your laboratory or company.

6.10 Installing and Configuring Tecan IAM

This chapter, designed for IT administrators, provides comprehensive guidance on the installation, configuration, and operation of Tecan IAM (Tecan's Identiy Access Managment).

6.10.1 Installing Tecan IAM

The installation supports the following configurations:

- **Local User Management (one system)**
  All components for Tecan IAM are installed on the local PC (refer to section "Installing IAM Server (standalone)" [▶ 104]).
  Note: This is the default installation.
  Note: No LAN connection is required.
- **Tecan Cloud Identity provider**
  Tecan Cloud Identity provider supports an installation to use a Tecan Cloud Identity (refer to section "Installing IAM Server for Cloud IAM usage" [▶ 105]).
  Note: This requires an Internet connection.
  Note: It allows users to use the same username and password with Tecan's cloud services as well as Instrument Control Software such as FluentControl.
- **LDAP (Local Roles) / OpenID Connect (Local Roles)**
  LDAP (Local Roles) / OpenID Connect (Local Roles) supports an installation that uses the LDAP or OpenID Connect (OIDC) server of the user's company for authentication (refer to section "Installing IAM Server in combination with LDAP/OpenID IdP (Active Directory)" [▶ 107]).
- **Network Installation (several systems)**
  This installation option allows several instrument Control PC on the same LAN to share the same login.
  It describes how to install Tecan IAM to connect to an already installed Tecan IAM Server running on another Instrument Control PC on the user's LAN (refer to section "Installing IAM Server for a network setup" [▶ 110]).

Make sure that endpoint protection settings (e.g., Windows Defender) are configured to allow the installation of IAM without being blocked.
For this, the IAM application must be allowed before starting the installation process.
During installation, the installer will create an initial local IAM Windows service user, which could potentially be flagged as a security threat.

6.10.2 Installing IAM Server (standalone)

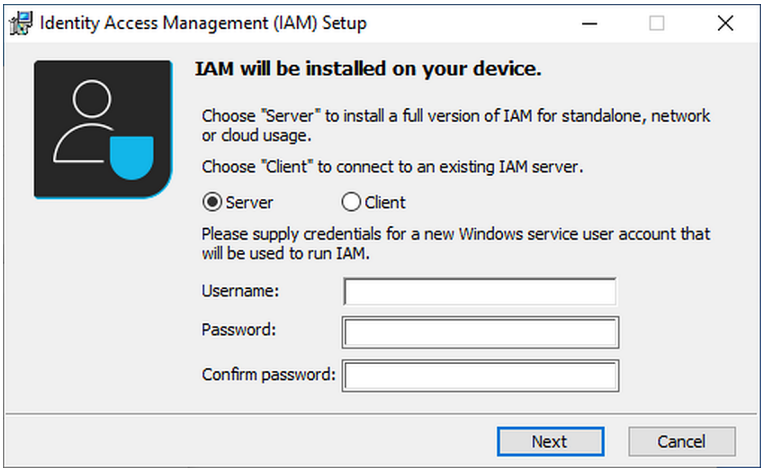
The IAM Server installation (standalone) is used for a single Instrument Control PC. It can also be used as a server in a network set up.
The IAM Server installation (standalone) is the default installation.

6.10.2.1 Pre-Installation Requirements

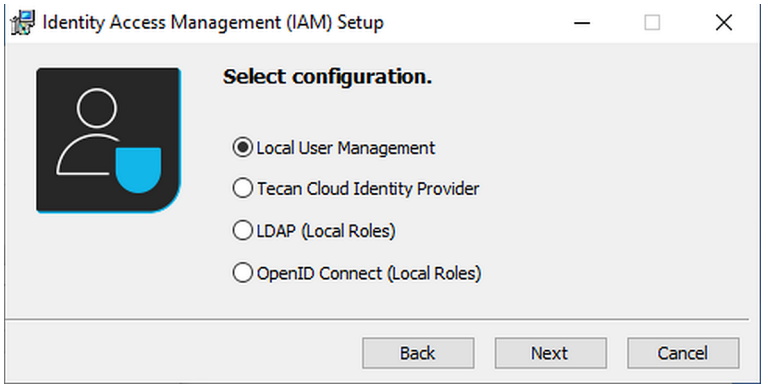
- Local Windows administrator rights on the instrument control PC with the rights to create a local user
- Ability to open a local port

6.10.2.2 Installation process

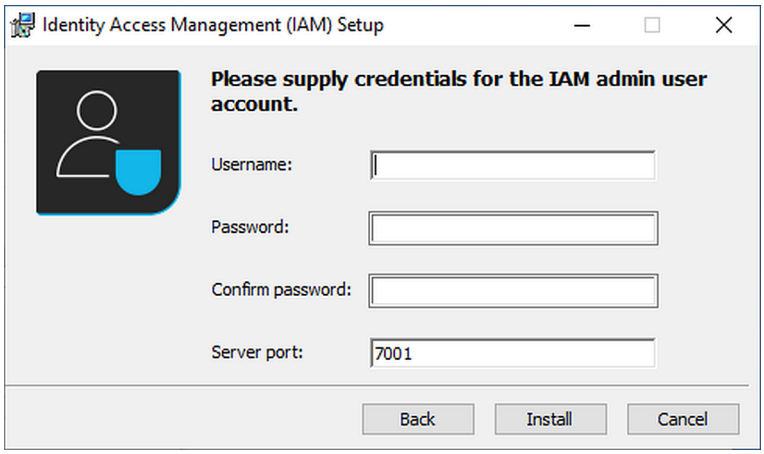
1. Run the installer as administrator.
2. Select “Server” in the first overview window.
3. Enter a new and unique user account name to create a new user account on your machine.
   Note: The Tecan IAM service will run under this newly created Windows Service account after installation.
   Note: The password can be changed after installation.



4. Select “Local User Management” in the next window.



5. Create a Tecan IAM admin.
   Note: The creation of a Tecan IAM admin is required to access the user management system after the installation.
   Note: A free port for the server is determined automatically but can be adjusted if needed. This port will be used by the Tecan applications over localhost.
   Note: Make sure that the Server port is not blocked by the firewall, operation system or endpoint protection.



6.10.3 Installing IAM Server for Cloud IAM usage

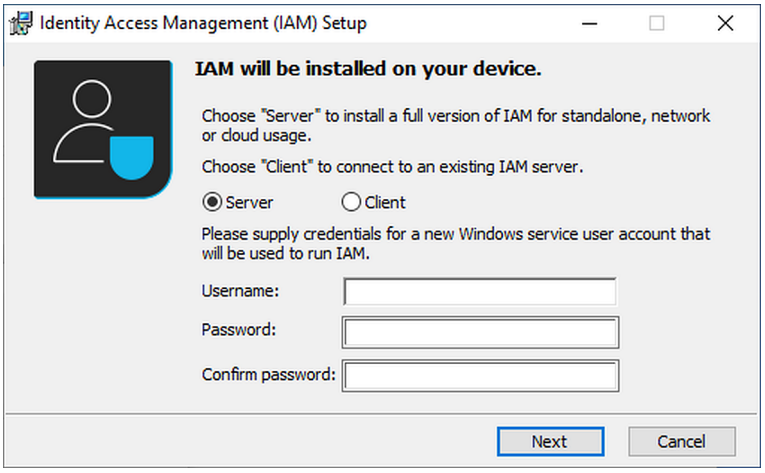
The IAM Server installation for Cloud IAM usage prepares the Tecan IAM for the use with Tecan's cloud-based user management system.

6.10.3.1 Pre-Installation Requirements

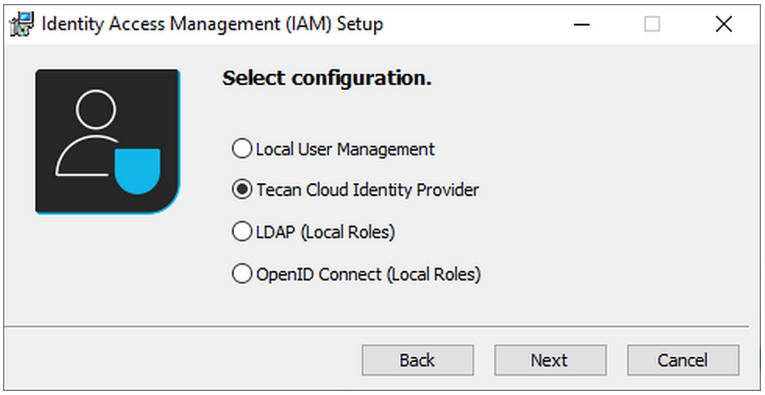
- Local Windows Admin rights to the Instrument Control PC with the rights to create a local user
- Depending on the setup, the chosen ports must be open
- An internet connection with access to Tecan's cloud services
- Instrument type must be supported in Tecan's cloud service, please refer to your instrument's manual

6.10.3.2 Installation process

1. Run the installer as Windows administrator.
2. Select “Server” in the first overview window.
3. Enter a new and unique user account name to create a new user account on your machine.
   Note: The Tecan IAM service will run under this newly created Windows Service account after installation.
   Note: The password can be changed after installation.

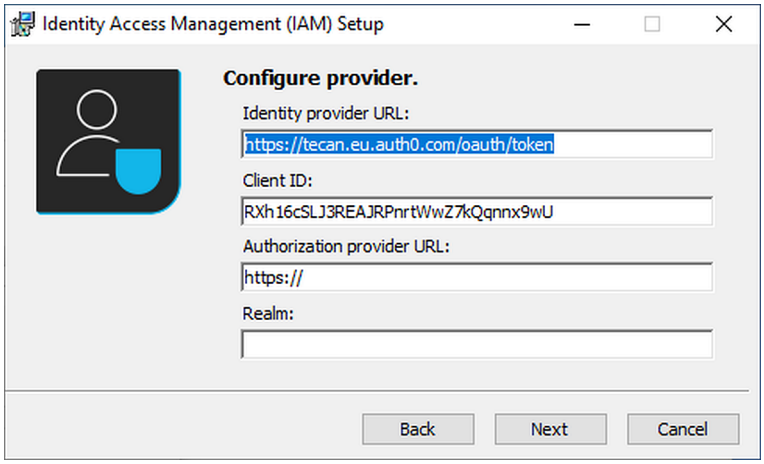


4. Select “Tecan Cloud Identity Provider” in the next window and click next.

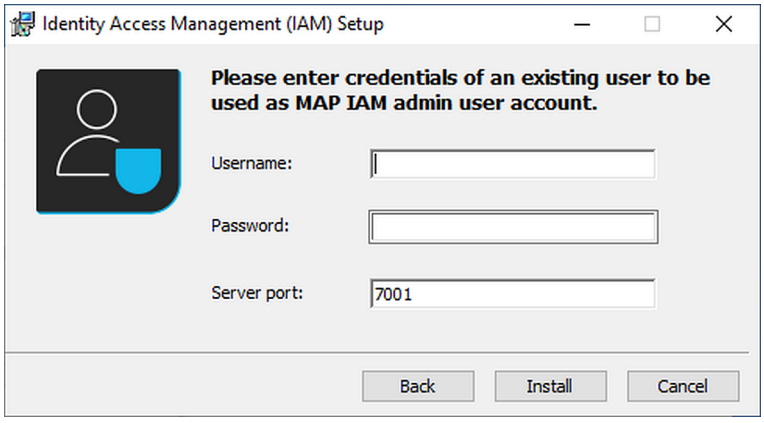


5. Provide the company’s realm and the URL to cloud organization management.
   Note: Tecan's default identity provider is already listed in the window.
   Identity provider URL: OpenID provider’s URL.
   Default: auth0, URL: `https://tecan.eu.auth0.com/oauth/token`
   Client ID: OpenID provider’s client ID.
   Default: `auth0`, ID: `RXh16cSLJ3REAJRPnrtWwZ7kQqnnx9wU`
   Authorization provider URL: URL to Tecan’s organization management, providing user roles and permissions.
   Default: `https://organizationmanagement.tecan.com/api/Users/`

`MyRolesAndPermissions`
Realm: Auth0's user database.
Default: Tecan SSO



6. Create a local Tecan IAM admin.
   Note: The creation of a local Tecan IAM admin is required.
   Note: Provide the user credentials of an existing admin for your organization in Tecan's organization management.
   Note: A free port for the server is determined automatically but can be adjusted if needed. This port will be used by the Tecan applications over localhost.
   Note: Make sure that the port is not blocked by your firewall, operation system or endpoint protection.



6.10.4 Installing IAM Server in combination with LDAP/OpenID IdP (Active Directory)

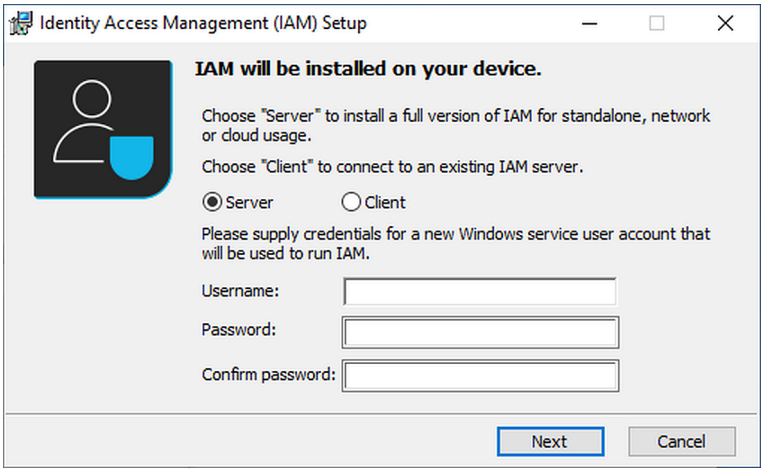
The IAM Server in combination with LDAP/OpenID IdP installation is used for set up Tecan IAM to connect to the LDAP or OpenID server of the user's company.

6.10.4.1 Pre-Installation Requirements

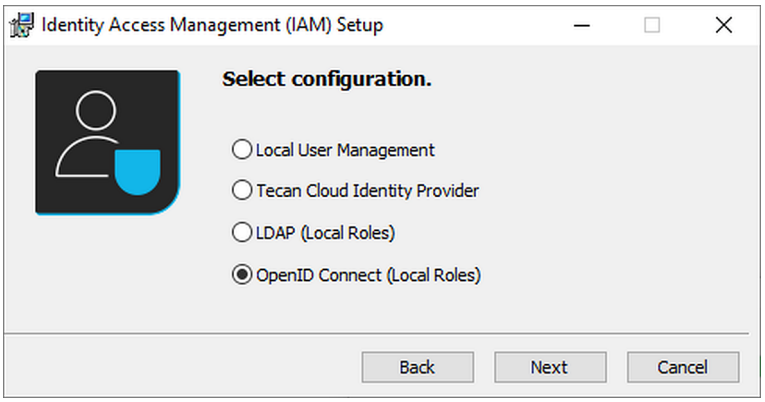
- Local Windows Admin rights to the Instrument Control PC with the rights to create a local user
- Depending on the setup, the chosen ports must be open
- Information on your companies LDAP
- URL and Client ID of the OIDC provider

6.10.4.2 Installation process

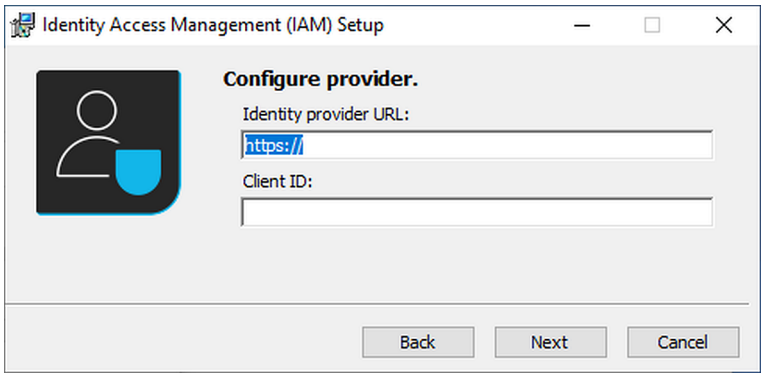
1. Run the installer as administrator.
2. Select “Server”in the first overview window.
3. Enter a new and unique user account name to create a new user account on your machine.
   Note: The Tecan IAM service will run under this newly created Windows Service account after installation.
   Note: The password can be changed after installation.



4. Select LDAP or OpenID in the next window.
   Note: The following steps are required based on the selection.

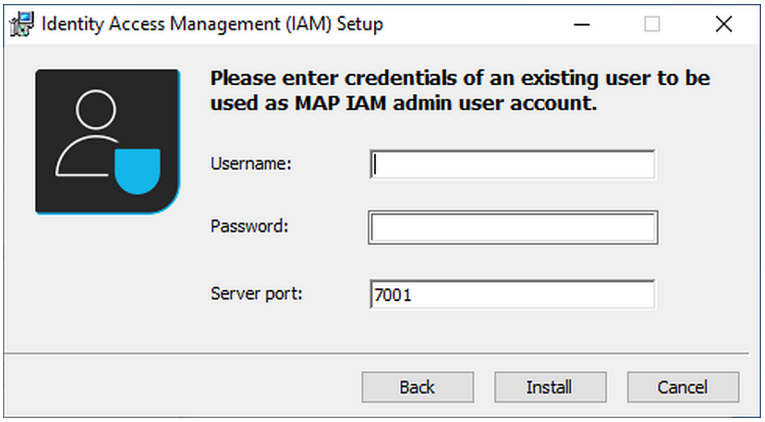


5. LDAP (Local Roles): no further actions.
   Note: The LDAP domain of the current user is taken from the system properties.
6. OpenID Connect: When choosing a combination of OpenID Connect (OIDC) provider and Tecan IAM, enter the URL and Client ID of the OIDC provider.



7. Create a Tecan IAM admin.
   Note: The creation of a Tecan IAM admin is required to access the user management system after the installation.
   Note: Provide the username of an existing user within your company's Active Directory.
   Note: A free port for the server is determined automatically but can be adjusted if needed. This port will be used by the Tecan applications over localhost.
   Note: Make sure that the port is not blocked by your firewall, operation system or endpoint protection.

After entering the login data, the installer connects to LDAP and attempts to authenticate using the provided credentials. If the connection is successful, the installer will proceed with the installation. If the connection is not successful, the corresponding error returned by the server is displayed. A similar situation occurs with OIDC. The following headers are used when sending the request to openID:
grant_type - password
username - userName from the installer
password - password from the installer
scope - openid profile email
client_id - clientId from the installer



6.10.5 Installing IAM Server for a network setup

The IAM Server for a network setup installation is used to set up Tecan IAM in a network environment to use one user management system across multiple instruments.

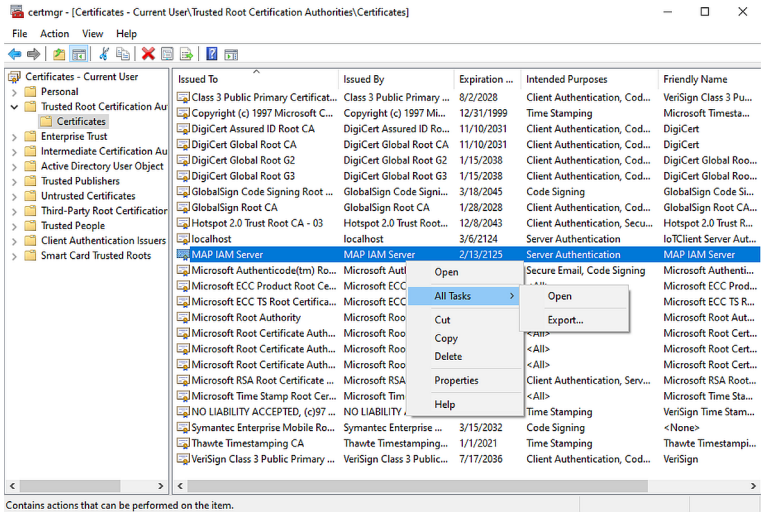
6.10.5.1 Pre-Installation Requirements

- Local Windows Admin rights to the Instrument Control PC with the rights to create a local user
- Access to LAN with a running Tecan IAM server
- Depending on the setup, the chosen ports must be open
- A trusted certificate from the IAM server (see below)

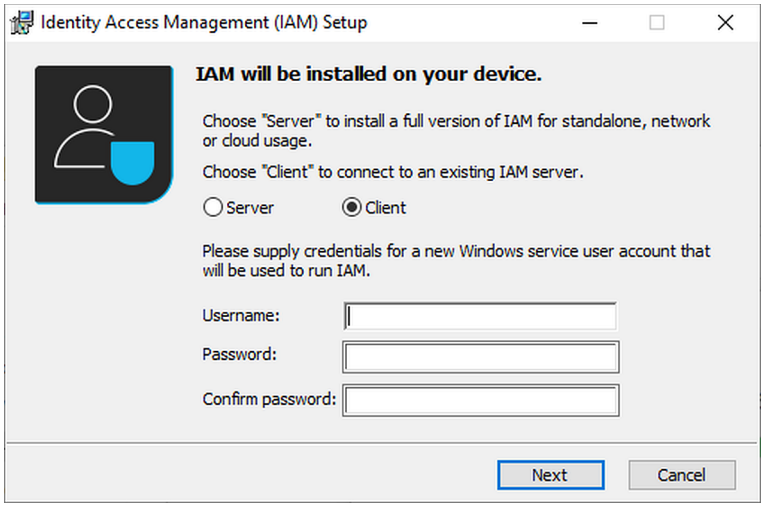
6.10.5.2 Installation process

Make sure that a Tecan IAM Server is installed on one machine in the LAN. If it is not installed, follow the installation guide in chapter "Installing IAM Server (standalone)" [▷ 104], "Installing IAM Server for Cloud IAM usage" [▷ 105], "Installing IAM Server in combination with LDAP/OpenID IdP (Active Directory)" [▷ 107], depending on the authentication provider. After the installation, follow the next steps.

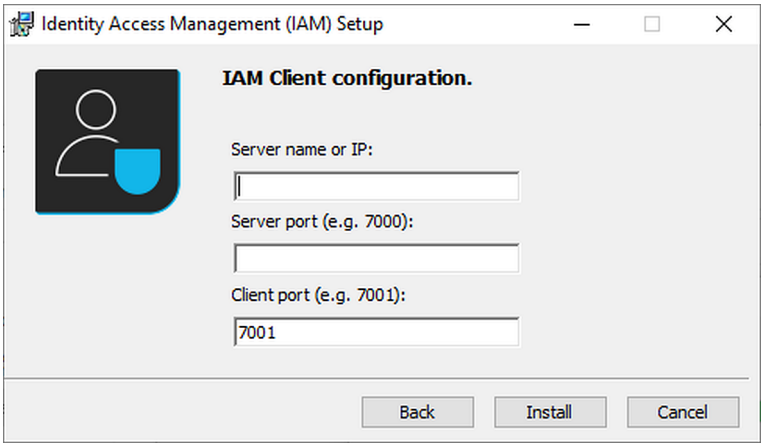
1. Export the Tecan IAM certificate from the IAM server PC and install it in the same folder "Trusted Root Certification Authorities\Certificates" on the client machine.
   Note: To access the windows certificate manager, serach for "Manage computer certificates" or use the command prompt and type "certlm".



2. Run the installer as Windows administrator.
3. Select “Client” in the first overview window.
4. Enter a new and unique user account name to create a new user account on your machine.
   Note: The Tecan IAM service will run under this newly created Windows Service account after installation.
   Note: The password can be changed later.



5. Enter the IP address and port of your Tecan IAM server in the next window.
   Note: The server port is set during the server installation (refer to chapter “Installing IAM Server (standalone)” [▶ 104], step 7).
   Note: A free port for the client is determined automatically but can be adjusted if needed.
   Note: Make sure that the Server and Client ports are not blocked by your firewall, operation system or endpoint protection.



6.10.6 Operating Tecan IAM

Tecan IAM is running as a Windows service under the user account created during installation and provides endpoints for various actions. Tecan's applications use these endpoints to access the user management system.

6.10.6.1 Changing the identity provider

This section addresses transitioning from one identity provider (e.g., LDAP) to another (e.g., OpenID) within Tecan IAM.

Current Limitations: Direct switching between providers is not supported. A full reinstallation of Tecan IAM is required for this change.
Recommendation: Back up essential data before initiating the reinstallation process.

6.10.6.2 Locked out IAM admin account

The IAM administrator accounts can be locked after too many login attempts.
Immediate Action: Wait for the designated lockout period to expire. Attempt to log in again with your correct credentials.
If the password has been forgotten, please contact Tecan Support ("Contacting Tecan" [▶ 531]).

Remember to back up critical data before reinstalling.

6.10.7 Unistalling IAM

Select the type of uninstallation to uninstall the IAM Server.



If "Remove application and all data" is selected, a computer restart is required.

6.10.8 Working with Active Directory

To work with Active Diectory, check the following aspects:

- An Active Directory user must be used during installation.
- The user performing the installation is assigned as an IAM Administrator.
- Roles must be created and assigned to users within Fluent Control.
  The role assignments are stored in the IAM service, not in Active Directory.
- Users do not synchronize automatically. Only those who log in will be added to IAM.
- The full name field for users remains empty.
- Users cannot change their passwords within IAM.
- Password resets must be performed via Active Directory.
- Enabling or disabling users can be done through IAM or Active Directory.
- Renaming a user in Active Directory does not update the user in IAM; instead, a new entry is created.
- Manually adding a user is possible but ineffective, as the user does not synchronize with Active Directory.
  Since authentication is handled by Active Directory, not IAM, changing the password in IAM has no effect in Active Directory mode. Despite this limitation, the user will still appear in the IAM user list.
- Active Directory is used solely for authentication and does not manage application-specific permissions or roles.

6.10.8.1 Steps to Introduce a User from Active Directory to IAM without Prior Login

- **User Creation in FluentControl:**
  The admin manually creates a user in FluentControl with the same username as in Active Directory, regardless of whether the user already exists in Active Directory.
- **Password Assignment:**
  The admin assigns any password during user creation.
  The password is irrelevant, as authentication will be handled by Active Directory.
- **Role Assignment:**
  The admin assigns the necessary roles and permissions to the user in FluentControl.
- **User Synchronization:**
  The system automatically links the user to the corresponding Active Directory account based on the username.
- **Login Process:**
  The user logs in using their Active Directory credentials, with the authentication managed by Active Directory.
- **Access Verification:**
  Upon successful login, the user has access to the assigned roles and permissions in FluentControl.

6.10.9 Support and Installation Logs

The installation logs are in the following folder:
`C:/ProgramData/Tecan/MAP.IAM`

Some errors can be viewed in Windows Event Viewer.

7 Menus

A menu is a group of the FluentControl main commands arranged by category, such as **File**, **Database**, **Edit**, **Run**, **Window**, **Settings** and **Help**. The menus are displayed on the menu bar located near the top of the main window.

The menus of FluentControl are only available if you have been given FluentControl administrator or key operator rights in the User Administration. Operators can only run existing methods.
For details about access rights and user levels, refer to sections User Qualification and Access Right.

*Most of the features and dialog boxes in FluentControl are provided with context-sensitive help that can be accessed with the **F1** key.*

7.1 File

Use this menu to create a Method, Script, Process, Module or Worktable with the **New** command and **Open** an existing **Method**, **ProcessScript**, **Module**,**Liquid Class**, **Worktable**, **Carrier**, **Labware**, or **Tool**. Create a printout of the corresponding editor content, i.e., your method or script etc., with the **Print** command.

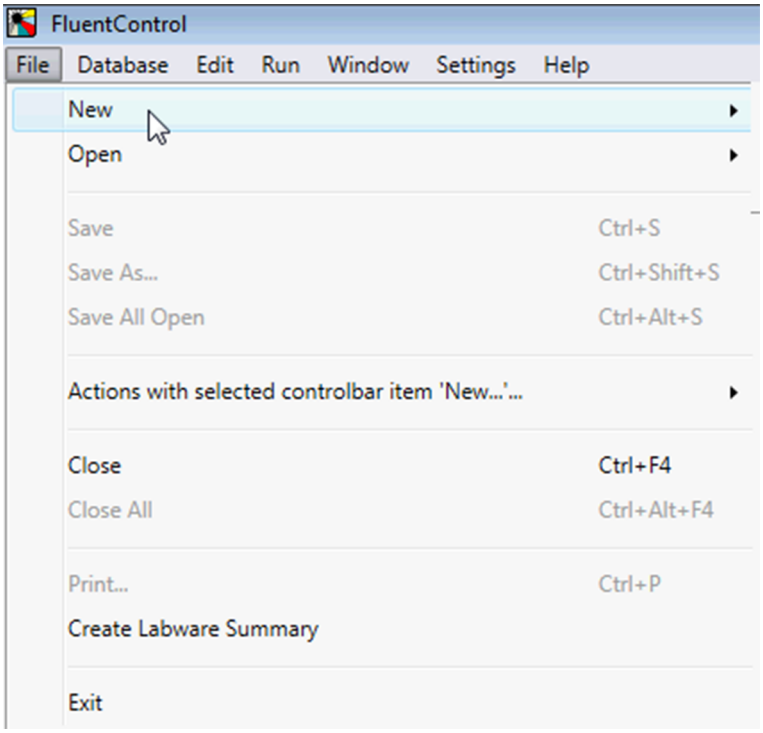


Fig. 55: File Menu

The **File** menu contains the following commands:

- **New**: The command to create a new method, process, script, module or worktable.

- **Open**: The command to edit an existing method, script, process, module, liquid class, worktable, carrier, labware, or tool.
- **Save**: The command saves the current editor content (e.g., a method or a script). Shortcut: Ctrl+S.
- **Save As**: The command allows you to save a copy of the current editor content under a new name. Shortcut: Ctrl+Shift+S.
- **Save All Open**: The command performs the save operation on each editor. Shortcut: Ctrl+Alt+S.
- **Actions with selected Controlbar**: Item to get the shortcut menu of the current object selected in the **Controlbar**.
- **Close**: The command closes the current editor: Shortcut: Ctrl+F4.
- **Close All**: The command closes all editors. Shortcut: Ctrl+Alt+F4.
- **Print**: The command to create a printout of the currently active editor content. If the currently active editor is, for example, a script editor, the script which you are editing will be printed.
- **Create Labware Summary**: The command to export a plain text file with all parts, their part number (Product number), type and quantity present in the currently open Worktable editor.
- **Exit**: The command to close the software. If unsaved data exists, you will be asked if those data objects should be saved before closing the program, or if the exit operation should be cancelled: Shortcut: Alt+F4.

7.1.1 Creating a New Script

1. On the **File** menu, click **New** > **Script**.

   There are more ways to create a script: from the Startpane, or from the Controlbar.

   *You see a **NewScript** tab in the main section and, in the **Controlbar** window, the list of direct commands to start with.*

   In the Controlbar, a layer above, there are more commands.

7.1.2 Editing an Existing Script

1. On the File menu, click **Open** > **Script**.

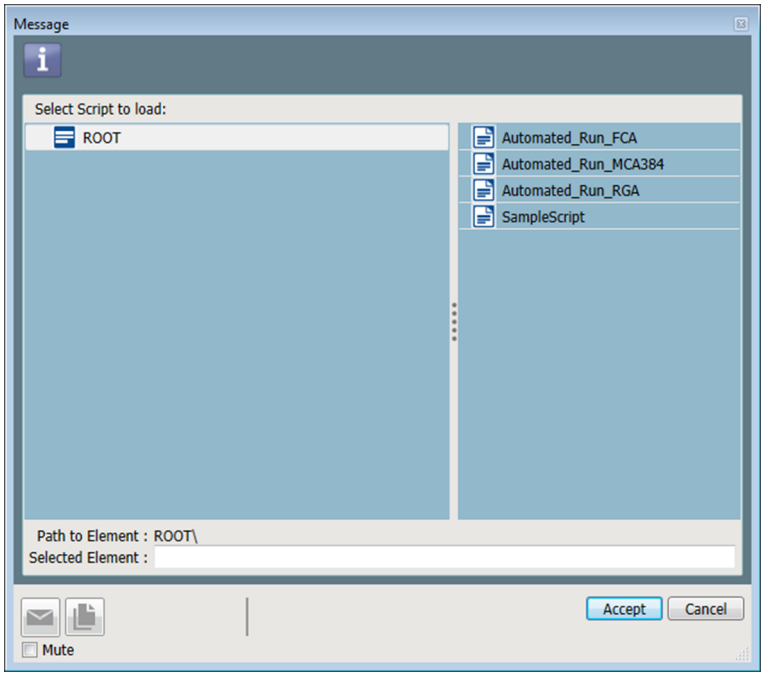


Fig. 56: Select script to load dialog

2. Click the script to load (e.g., Automated_Run_FCA) and click **Accept**.

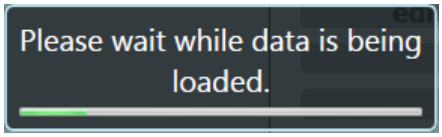


Fig. 57: Data is being loaded

3. The selected script will open in a new tab. It may take some moments to load the data and open the script.

7.2 Database

7.2.1 Overview

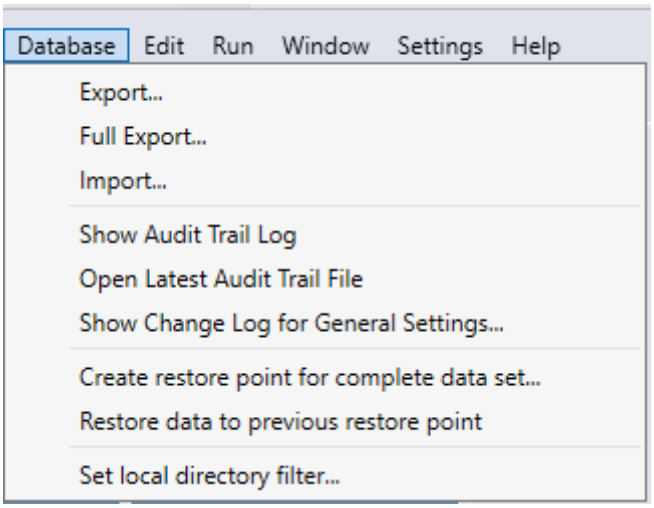


Fig. 58: Database menu

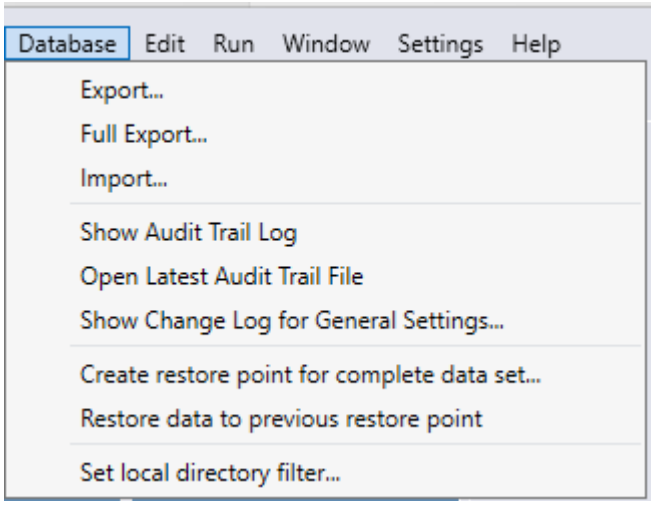
The **Database** menu contains the following commands:

- Use the **Export...** command to save your object, for example, a script to an External Deep Save Data File.
- The **Full Export...** command saves the complete database to a file. Labware and tool templates as well as instances are also part of the exported data set.
- Use the **Import...** command to read a previously exported External Deep Save Data File (a file containing one or more objects) and select the objects to be imported into the database.
- The **Show Audit Trail Log** command displays the audit trail log at the Start tab.
- The **Open Latest Audit Trail File** command opens the latest file in the Audit Trail Folder with the default application of Windows.
- The **Show Change Log** command displays the change log for the selected object (e.g., **General Settings**).
- The **Create restore point for complete data set...** command creates a restore point for the complete data set. Restore points are created automatically with every run for any object involved in the run that has changed. Additionally, restore points can be manually triggered with the **Create restore point for complete data set...** command.
- The **Restore data to previous restore point** command first creates a restore point of the database and then leads to a Revision history window, where the user can select a certain point in the history to go back to.
- The **Set local directory filter...** command changes the local directory for the data set.

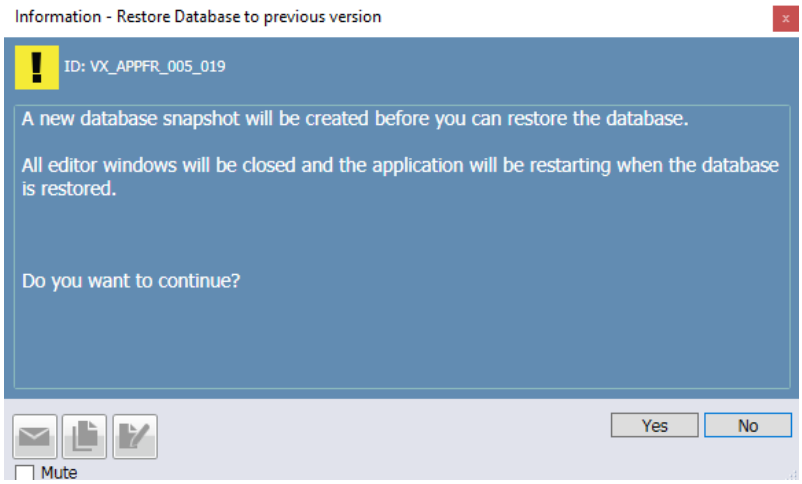
7.2.2 Restore Database

This section explains how to restore a database to a previous version in FluentControl.

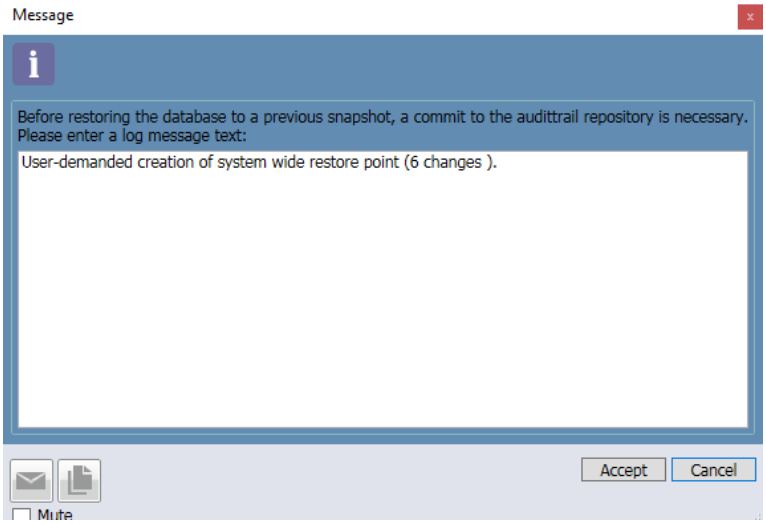
1. Click **Database** in the menu bar
2. Click **Restore data to previous restore point**.



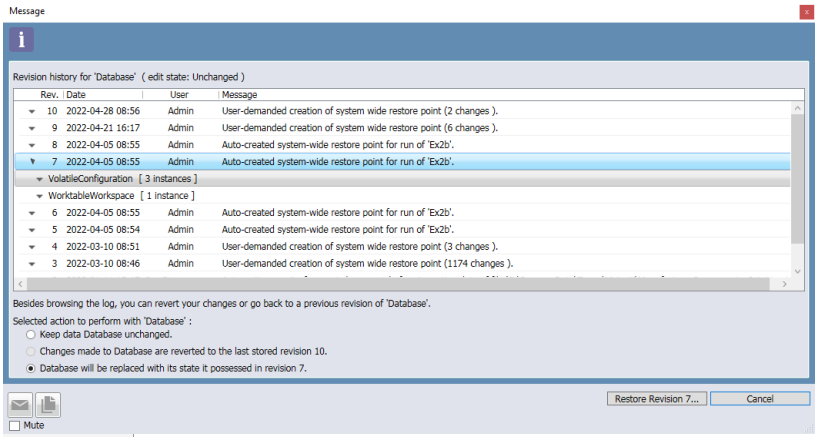
3. Click **Yes** to confirm the next information prompt



4. Edit Restore point's message text if desired, then click **Accept** and then click **OK**.



5. In the next prompt, select the desired restore point from the list.
6. Choose the option **Database will be replaced with its state it possessed in Revision XX**.
7. Click **Restore Revision XX...**, where XX indicates the revision of the chosen restore point.



8. Confirm the next prompt with **Yes, revert**.

FluentControl restores the database and restarts. After the restart, the database is restored to the selected revision. The most recent 5000 restore points are accessible. If you need to restore to an inaccessible restore point, contact our help desk.

7.2.3 Export

When saving a script, dependent objects (e.g., worktable, carrier, labware, liquid class) and external file references used in script commands (e.g., worklists, VB scripts) will be linked to the script. When exporting a script, dependencies and referenced files are included in the export file.

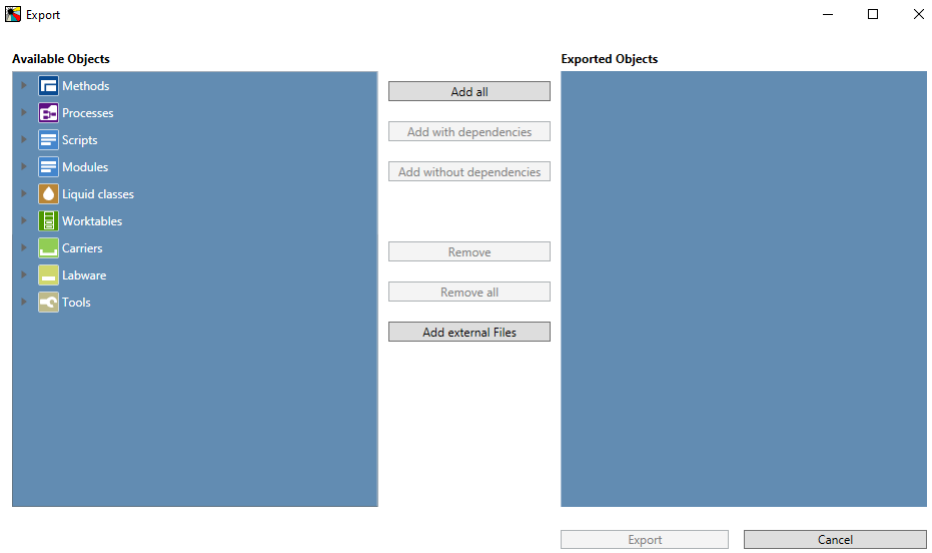
To export a database object proceed as follows:

1. In the **Database** menu, click **Export...** or **Full Export....**

   *The **Export** dialog window opens.*

   *The **Available Objects** list on the left is expandable and contains items that are available for export.*

   *The **Exported Objects** list on the right contains items that are currently selected for export (empty in the image below).*



2. Select the items to be exported by using one of the following options:

   ***Add all** adds all items in the database to the **Exported Objects** list.*

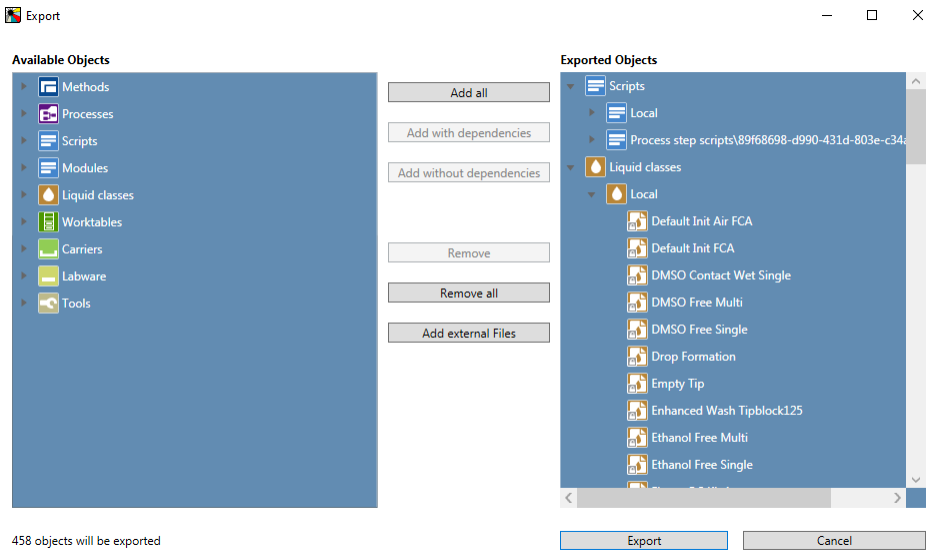
   ***Add with dependencies** adds all selected items to the **Exported Objects** list—including their referenced scripts, liquid classes, worktables, carriers, labware and tools.*

   ***Add without dependencies** adds all selected items to the **Exported Objects** list—without their referenced objects. Note that if any of the referenced items is missing, it may not be possible to import these items successfully to the target system.*

3. To include external files in the export (e.g., images, PDFs, etc.), click **Add external Files**.

   *In the file browser that opens, select the files and click **Open**.*

*The selected external files will be added to the **Exported Objects** list.*



4. Click **Export** to export all items of the **Exported Objects** list.

7.3 Edit

In addition to the **Undo, Redo, Undo List, Redo List, Clear undo/redo history, Cut, Copy, Paste**, and **Delete** commands, this menu also provides a search and replace feature for liquid classes and arms.

The **Edit** menu contains the following commands:

- **Undo/Redo**: The Worktable, Script and Liquid Class editors supports undoing or redoing of changes. The Labware and Carrier editors supports undoing or redoing of changes only after any changes have been saved.
- To **Undo/Redo** a change, use Ctrl+Z, Ctrl+Y or Undo/Redo in the Edit menu.
- **Undo List:** Undo list is an expandable menu showing the available actions to undo.
- **Redo List**: Redo List is an expandable menu showing the available actions to redo.
- **Clear undo/redo history:** Use the command to create a clean starting point or to reduce clutter.

7.3.1 Find and Replace Liquid Classes

With the **Find and Replace Liquid Classes** feature you can change the used liquid classes in open scripts. It allows to replace the used liquid classes for selected script commands by another liquid class in batches instead of having to manually change liquid classes.

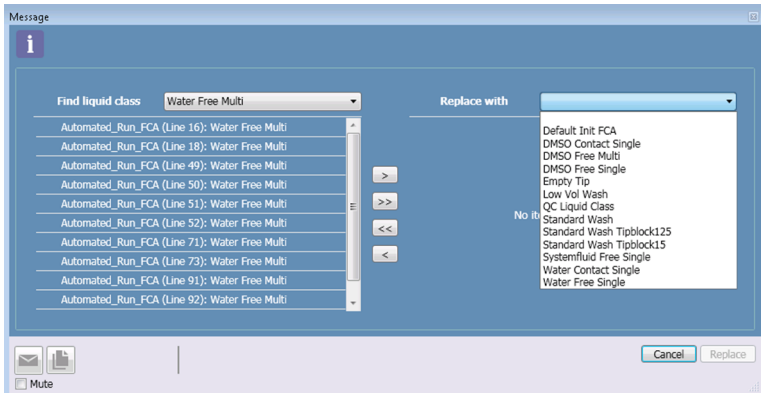


Fig. 59: Search and Replace Liquid Classes dialog

A dialog shows all found script commands in open editors filtered by the currently selected liquid class.

1. To achieve this, in the **Find liquid class** list, click the liquid class you want to replace.
2. in the **Replace with** list, select the new liquid class, which should replace the items found in open editors.
3. If you are satisfied with your selection of commands that you want to get a new liquid class, click **Replace**.
4. Otherwise, click **Cancel**.

7.3.2 Find and Replace Arm

With the **Find and Replace Arm** feature you can change the arms used in open scripts. This feature can only be used if there are two arms of the same type on the instrument, for example 2 Air FCAs or 2 Liquid FCAs.



Fig. 60: Search and Replace Arm dialog

7.4 Run

All scripts that are intended to be used in routines not just by the key operator during development, but also by the operator during operation, should be added to a method, so they can be started from the touch monitor.

The **Run** menu contains the following commands:

- **Initialize Instrument**: The command is only available if no run is active or paused. If the command is not available it is grayed out. If the instrument is not or only partially initialized, this command starts the instrument initialization. If the instrument is already initialized, this command shows a dialogue telling the user that the instrument is already initialized and asking them if they wants to force an initialization, and offering UI-Controls for **Yes** and **No**.
- **Prepare Method**: The command is only available in edit mode, otherwise grayed out. If the edited object is a script, this command opens a new method editor tab with a method containing the edited script and switches the software to run mode. If the edited object is a method, this command switches the software to run mode.
- **Run**: The command can be selected from the menu for a script and then the user is asked if he wants to use the saved DiTi positions or start with fresh DiTi racks. Additionally, the Run action can be aborted by selecting cancel.
- **Pause/Resume**: Pauses/resumes the current run depending on the state. The command reads **Pause** if a run is being executed and the software is in run mode. If a run is paused, the command reads **Resume**. If no run is being executed, the command is grayed out.
- **Stop**: The command is only available in run mode when a run or initialization is being executed, otherwise it is grayed out. Stops the current run or initialization.
- **Auto Context Check**: A check mark indicates that the context check is active during editing. To switch the context check off, remove the checkmark by clicking on this menu option. If the context check is inactive and a run is started, a dialog will warn the user that the context check is switched off and the run may be interrupted.
- **Fast Simulation mode**: A check mark indicates that the Fast Simulation mode is active in the 3D Simulator during simulation.

7.5 Window

- **Lock**: If User Administration has been activated, use this option to lock FluentControl. Active runs will not be interrupted. If User Administration is not activated, the option is grayed out.
- **Switch User**: If User Administration has been activated, use this option to switch between users. If User Administration is not activated, the option is grayed out.
- **Reset Layout**: Reset window placement to the default layout.

A checkmark indicates which windows are currently active:

- Start page
- Controlbar
- Worktable
- Gantt chart
- Worktable Orphaned Labware
- Trace view
- Infopad
- MoveTool
- Variable View

- Open 3D Simulator

7.5.1 Variable View

The system provides a panel that displays the currently defined variables along with their values. It is a read-only view. For more information see section Variables.

7.5.2 Open 3D Simulator

The **3D Simulator** visualizes scripts and methods as they run. This includes picking up tips, adapters and plates, and indicating pipetting steps. It is especially helpful when working with FluentControl when not connected to an instrument. For more information, please refer toSimulation Mode.

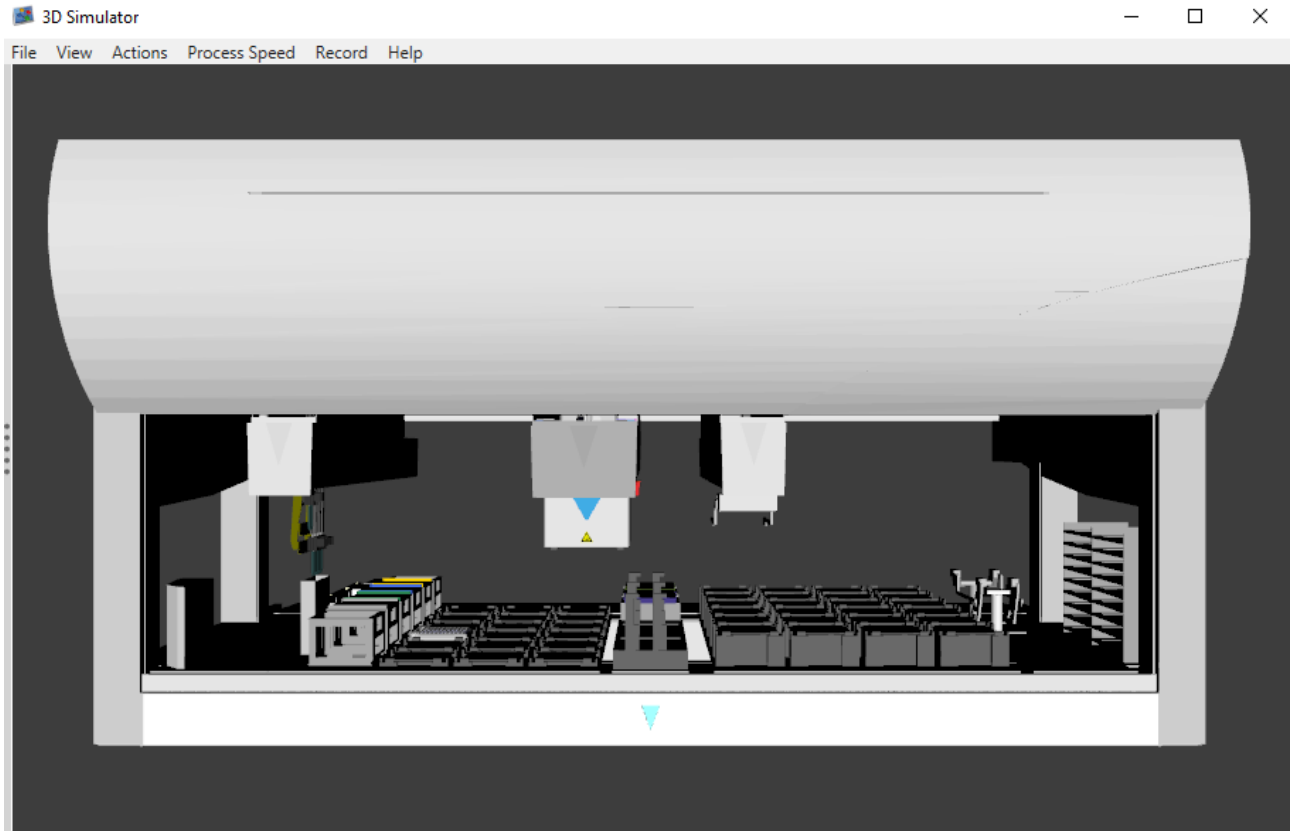


Fig. 61: 3D Simulator

7.6 Settings

The Settings category contains the following commands:

- **Controlbar Views**
- **Color Schemes**
- **External links**
- **Tecan Licensing Client**
- **Configure System**
- **User Administration**
- **Editing User Profile**

- **Changing Password**

7.6.1 Controlbar Views

Click **Settings** > **Controlbar Views** to change the view according to your preference. There are three supported views for the Controlbar:

- **Expert**: Small icons and list view. Individual window for each level.
- **Power**: List view. Windows slide when different levels are selected.
- **Novice**: Icon view: Windows slide when different levels are selected.
- **Toggle smart mode**: Context relevant icons. This option will allow you to switch between default mode (all items visible) and smart view.

We recommend to start with **Novice**.

7.6.2 Color Schemes

This not functional.

7.6.3 External Links

This option will open the available links in your default browser. To add additional links to this menu, copy any URL files to
`C:\ProgramData\Tecan\VisionX\ExternalLinks`
The file name will be displayed in the External Links menu.

7.6.4 Tecan Licensing Client

The client requests a license and add your license file to the Licensing Client Module (separate window). Please refer to"Activate License" [35].

7.6.5 Configure System

This will open the System Configuration panel:
User Administration: This enables you to turn on user administration or edit the current settings.

7.6.5.1 Method Approval

To use Method Approval user management has to be activated. Refer to section "User Administration" [90].

With Method Approval operators can only run methods that have been released by a power user.

All other methods are flagged:

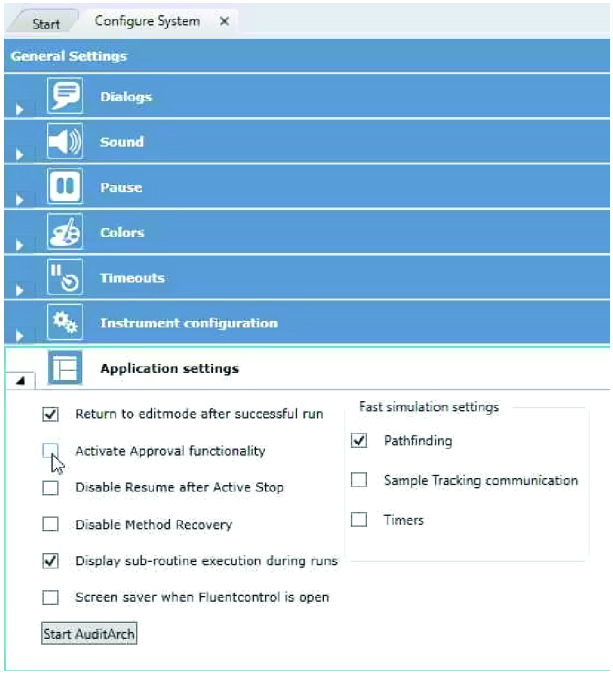
- Methods that are not released are flagged **red**.
- Methods that have been released and subsequently edited are flagged **yellow**.

The power user can release a method by right-clicking the method and selecting **Release** in the context menu.

To (de-)activate Method Approval proceed as follows:

1. Click **Settings > Configure system...**
2. Click **General Settings > Application settings**.

3. (De-)select **Activate Approval functionality**.



Releasing the first method takes a bit of time, as FluentControl approves all of the subcomponents of the method—i.e., liquid classes, carriers, labware, worktables, etc.

Subsequent releases for methods that contain the same subcomponents take much less time, as the components have already been released.

If a script, liquid class, labware, etc., is changed, it will cause all methods using it to revert to unreleased status—but re-releasing it will quickly restore the release status of the methods.

7.6.5.2 Log Level

The default log level in FluentControl is "Debug". For more details in log files, the log level can be changed to "Information".

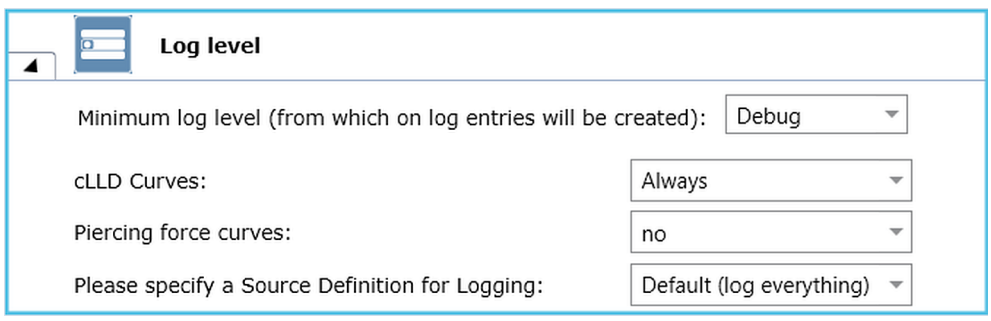


Fig. 62: Log Level

7.6.5.3 Fast Simulation

Fast simulation settings under **Configure System** can be configured to execute (checked) or skip (unchecked) the time consuming parts during the run. Time consuming parts are defined as following:

- Pathfinding
- Sample Tracking communication
- Timers

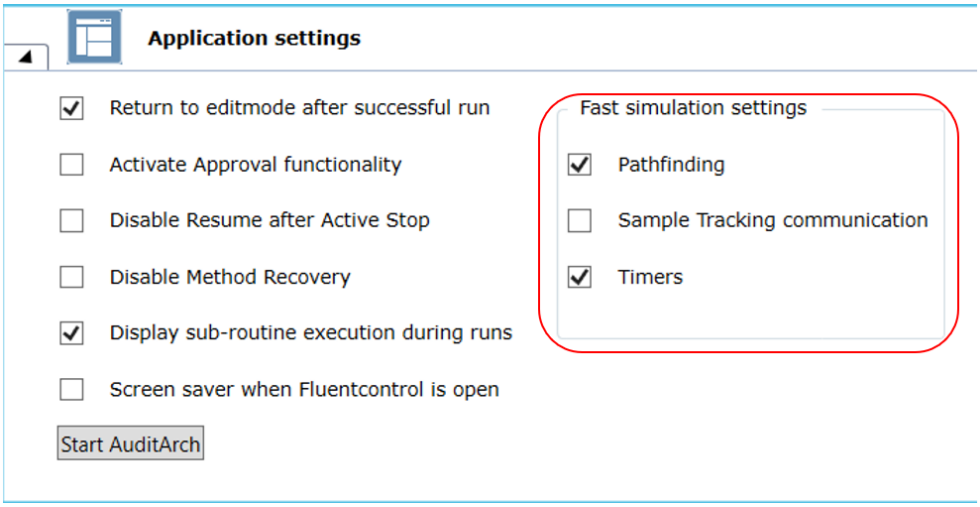


Fig. 63: Fast Simulation

7.6.5.4 Status Light Intensity

The status light intensity is configurable through “Configure System”. The default value is 100%. Lowering the intensity results in a lower brightness of the status light.

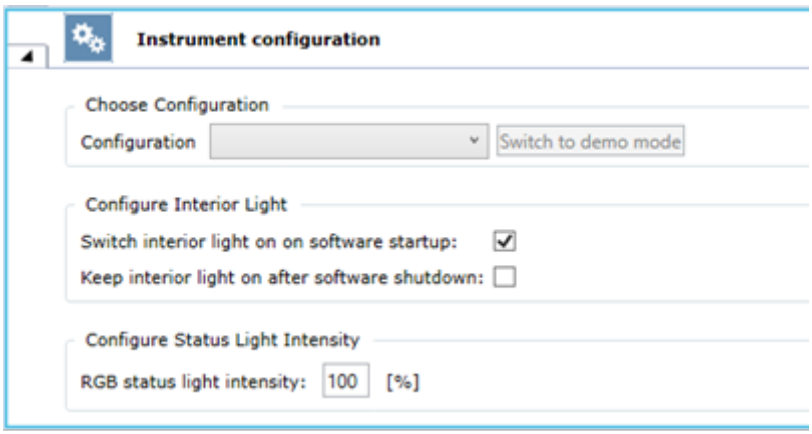


Fig. 64: Instrument Configuration

7.6.6 User Administration

Please refer to “User Administration” [▶ 90].

7.6.7 Editing User Profile

The logged-in user can edit their own profile picture and their own full name.

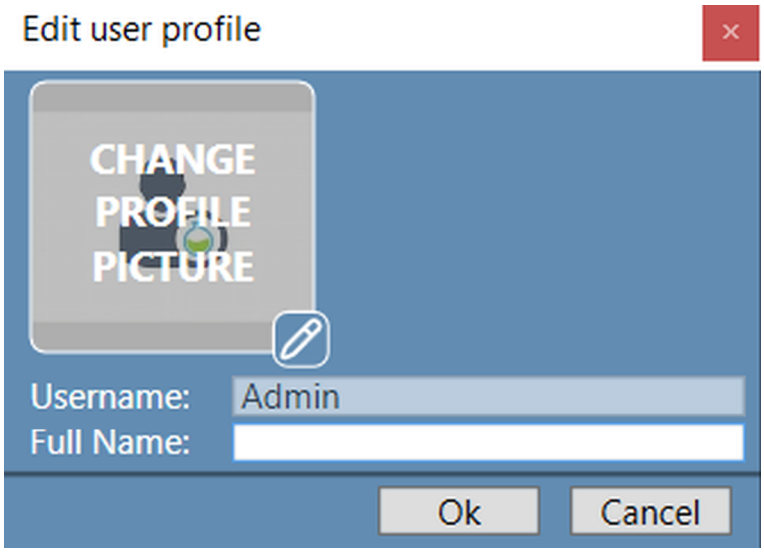


Fig. 65: Editing User Profile

7.6.8 Changing Password

The logged-in user can change their own password.

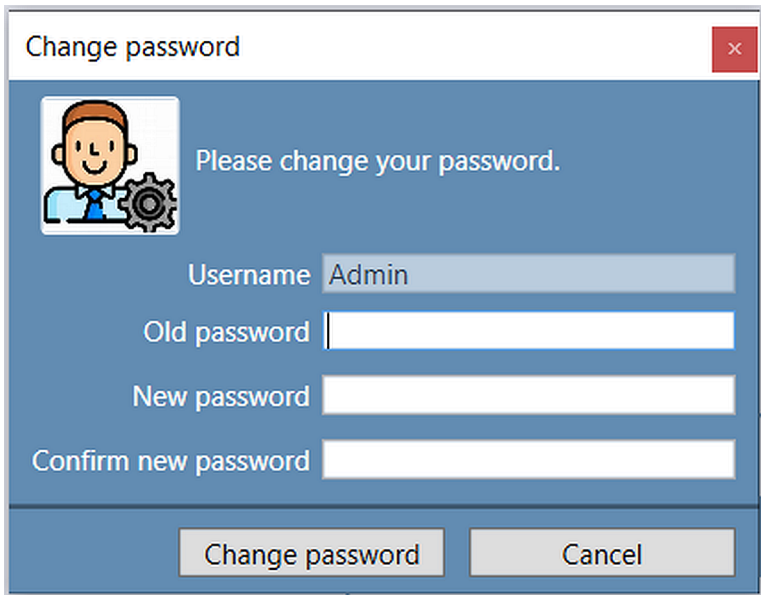


Fig. 66: Changing Password

7.7 Help

*Most of the features and dialog boxes in FluentControl are provided with context-sensitive help that can be accessed with the **F1** key.*

Help can also be accessed by clicking **Open** on the Help menu.

Information about the Software



Fig. 67: About FluentControl

Click **About FluentControl** to get information about:

- Version number and Build number (in case you need to contact customer support)
- Installed Licenses
- Intended Use (The illustration above depicts an example only. The displayed intended use depends on your actual instrument and installation. For more information on the intended use of your instrument please refer to “Intended Use” [▶ 10].)
- Copyright

*The product includes software components of third parties whose source code is located in the subfolder "C:\Program Files (x86)\Tecan\Third Party Source Code". For license overview please press **F2**.*

8 Editors

The Tools Editors allows the user to edit existing devices, carriers, tubes, racks etc. which are available from the FluentControl library.

8.1 Worktable Editor

The **Workable** editor displays the worktable, which belongs to the currently edited script, labware, carrier, or to the executed method. This 3D view allows you to edit and modify the represented worktable. On the right, buttons appear to help you move the camera around the worktable.

The worktable shows the objects that are expected to also be physically present on the Fluent and belong to the currently edited script or to the executed method.



Fig. 68: Worktable editor

NOTICE

Equipment damage!

Incorrect segment and labware positioning on the worktable may cause arms to crash.

- Make sure the Worktable view represents all objects placed on the Fluent deck.
- Only run a script if the Worktable view and the Fluent deck match exactly.

  ⇨ Only those parts of the instrument that are relevant for pipetting and carrier and labware placement are represented, but no movable parts like robot arms.

8.1.1 Default View and Natural View

The Worktable provides you with different views, as you would pan a camera. On the right, buttons appear to help you adjust the worktable view:

- **Default view**: Show the top view
- **Natural view**: Get the oblique projection



Fig. 69: Natural view

8.1.2 Rotate, Pan, Zoom In and Zoom Out

You can move the screen in multiple directions. Using the mouse, click the following buttons:

- **Left click**: Select object
- **Right click**: Display the shortcut menu for the active object
- **Left click & drag**: Move object
- **Right click & drag**: Rotate view
- **Left + right click & drag**: Pan view
- **Scroll wheel**: Zoom in & out

8.1.3 Worktable Overview

1. Click **Worktable Overview**.
   A list of all objects opens currently modeled with your worktable.
   *The **Worktable Overview** is context-sensitive.*
2. On worktable, click, for example, the grid position **1, 1**.
3. Watch the **Worktable Overview** under **Infopad**.
   *The **Wash Station Thru Deck DiTi Waste** labware is highlighted.*

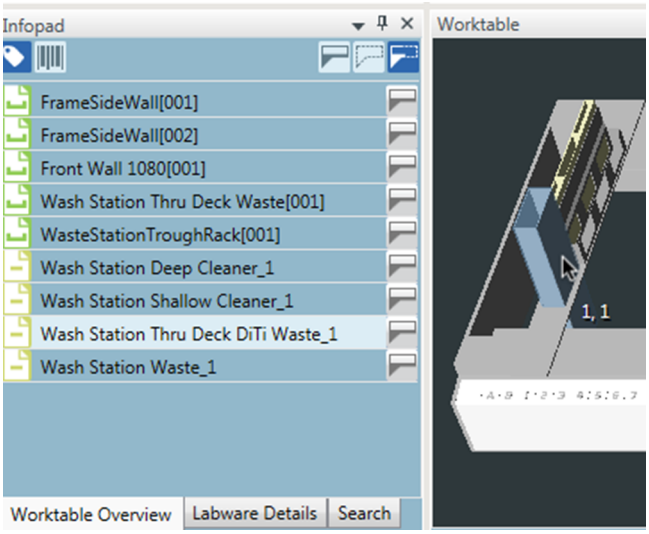


Fig. 70: Worktable Overview: Wash Station Thru Deck DiTi Waste 1

This connection works in both directions.

4. On **Infopad** > **Worktable Overview**, click, for example, the carrier **Wash Station Thru Deck Waste**.

   The corresponding segment on grid position 1 is highlighted on your worktable.

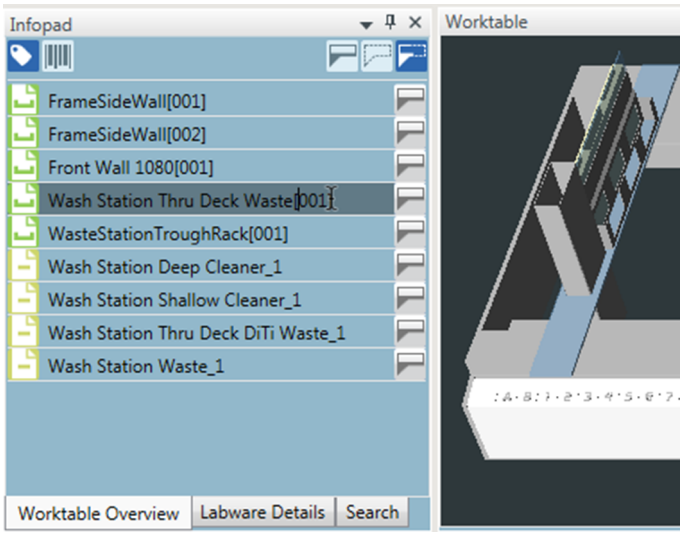


Fig. 71: Wash Station Thru Deck Waste 001

5. Rename worktable objects by editing the names directly in the **Worktable Overview**.
6. Turn labels on or off and toggle between barcodes and names.

8.1.4 Base Worktable

The **Worktable** Editor edits only the Base Worktable. A **Base Worktable** contains everything that is usually static, like the worktable itself, grids, carriers, static labware like troughs etc.. In contrast, the worktable saved together with a script contains all the elements needed in the script (base workspace and moveable labware).

8.1.5 Instrument Worktable

The instrument worktable is a separate worktable layer which is below the base and the script worktable. The instrument worktable file is hidden to the user and can only be edited through the Worktable Editor while editing a worktable, script, or process. The instrument worktable layer is meant to contain carrier or labware data which is mounted or bolted onto the instrument, and which rarely or never changes position. The instrument worktable is shared by all FluentControl by worktables. Therefore, if an item is placed onto the instrument worktable, it will appear on all previously created, as well as all newly created worktables.

Exporting the Instrument Worktable

Whenever a worktable is added to an export, the Instrument Worktable is automatically exported together with the worktable, but not shown in the list of exported objects.

If the worktable is exported with dependencies by clicking the option **Add with Dependencies**, the objects referenced on the Instrument Worktable are explicitly added to the export window and are visible in the export. If the option **Add without dependencies** is selected, the Instrument Worktable objects are not added.

Importing the Instrument Worktable

When a worktable is imported, a checkbox **Overwrite instrument worktable** becomes visible. Enabling this checkbox will overwrite the current Instrument Worktable on the system with the one from the export file.

If the option **Add with dependencies** is selected to import the worktable, the objects referenced on the Instrument Worktable are explicitly added to the import window and are visible in the import. If the option **Add without dependencies** is selected, the referenced objects are not added.

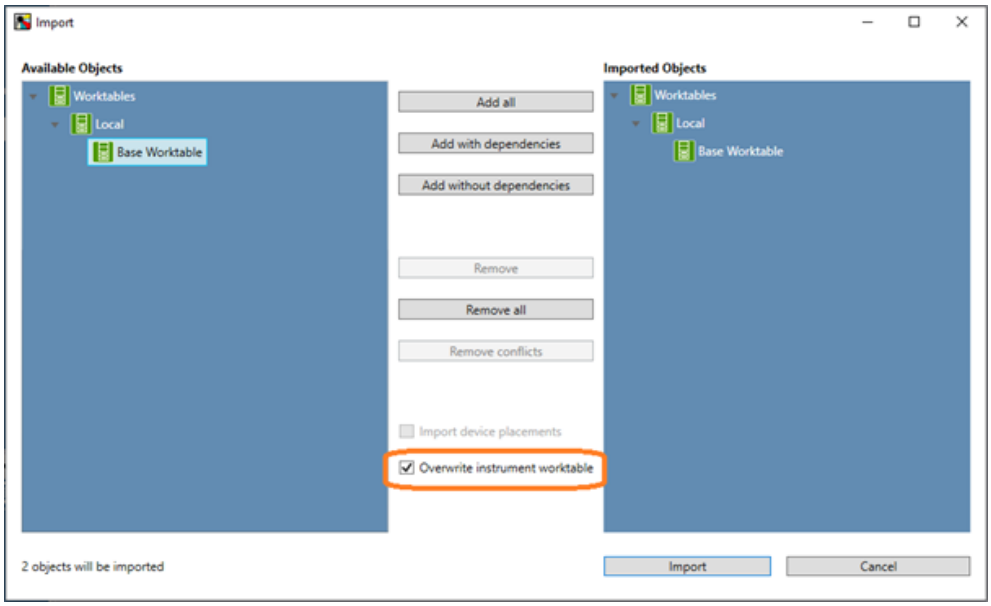
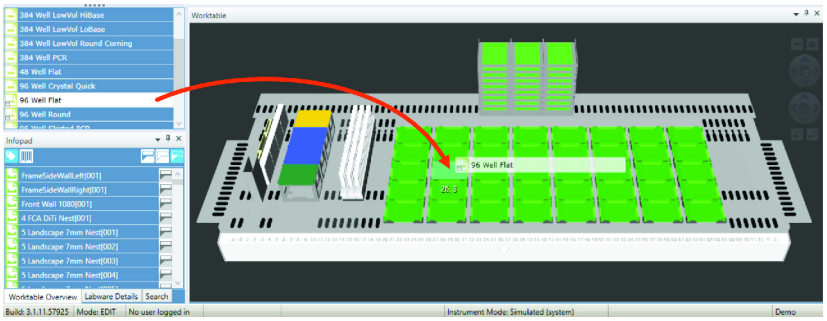


Fig. 72: Import of an Instrument Worktable

There is no necessity to make use of the Instrument worktable. If the user leaves the Instrument worktable empty and untouched, there will be no change to the previous behavior.

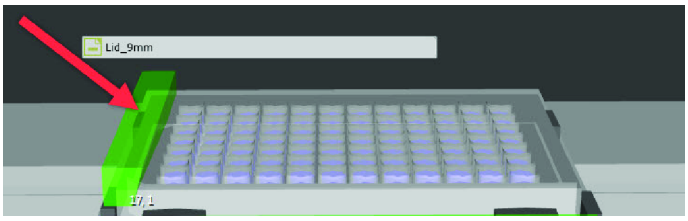
8.1.6 Dragging and Dropping Objects

The Worktable can be edited by dragging and dropping objects onto it. By selecting a carrier, or a labware or tool, the positions that can be occupied by this object are highlighted in green as shown in the illustration below.

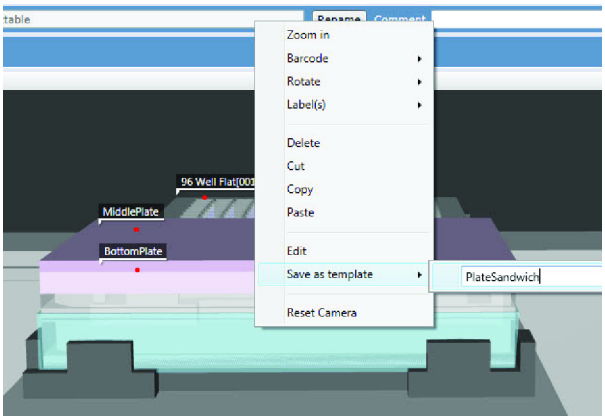


Objects can also be pre-assembled (for example carriers) and can be adapted. For example, "nests" (which are structures to hold SBS labware) can be removed from carriers. In principle, carriers can be put onto carriers and labware can be stacked. What can be put where is defined in the carrier and labware settings. For related information, refer to section Carrier Editor.

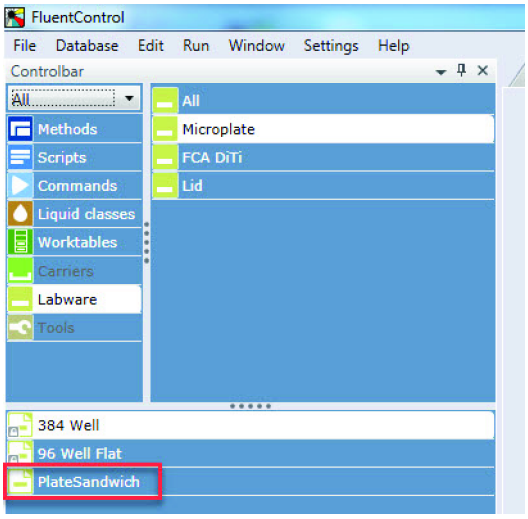
1. To create a stack of labware, place new labware on the CoverSite (see red arrow) of the labware underneath as shown below.



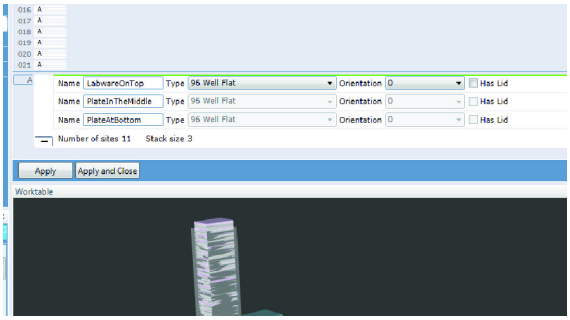
2. To save the stack of labware as a template, start by setting up the stack of plates on the worktable.
3. Right-click the labware at the bottom of the entire stack.
4. Select **Save as template** and give it a name.



5. The entire stack is now available in the Controlbar.



6. To generate plates in a storage carrier (e.g., hotel or carousel tower) in the Labware Group Editor, either generate the stack as a template, or create stacked plates in a storage slot by dragging and dropping plates on top of others in the same plate group.



8.1.7 Moving Objects in between Worktables

Labware items can be moved in between the instrument and the base worktable when the worktable editor is open. Items can be moved between the **Script, Instrument Worktable**, and **Base Worktable**, if the **Script** or **Process Editor** is open. This can be done by right clicking the item which shall be moved to the respective worktable.

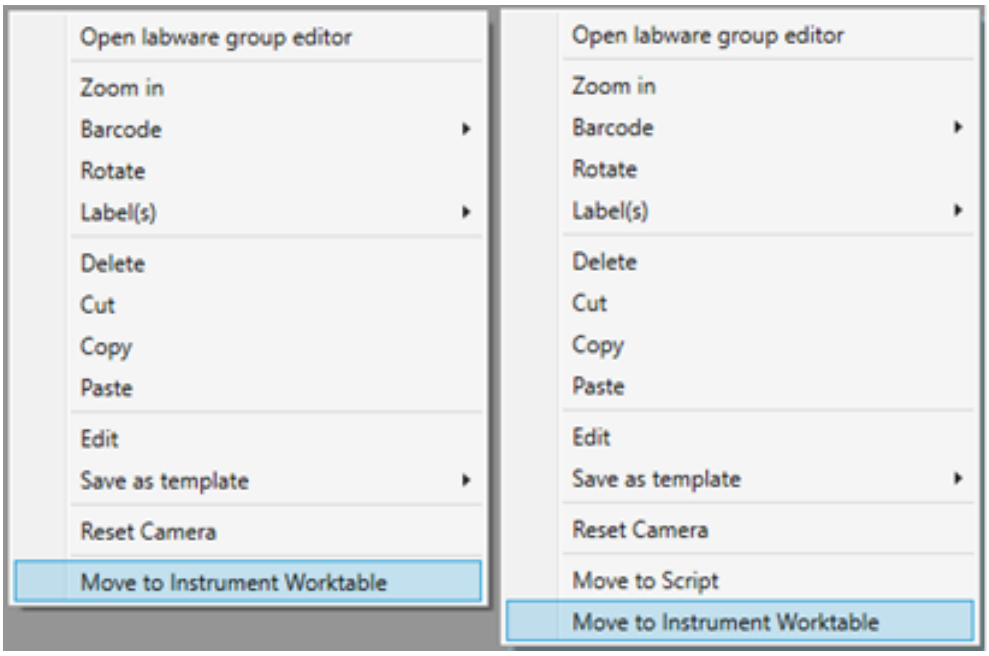


Fig. 73: After right click on in item it can be moved to the respective worktable

It is not possible to move labware to the **Instrument Worktable** when the carrier below is on the **Base** or **Script Worktable**. The option is disabled in that case. It is only possible if all underlying carriers are moved to the **Instrument Worktable** first.

8.1.8 Loading the Instrument and Base Worktable

Whenever a worktable is loaded, it will be merged with the **Instrument Worktable** present on the system. If there are no conflicts between the two worktables (e.g., two different labware items occupy the same site) then the worktable will load without issues. If conflicts do occur, the **Instrument Worktable** will be prioritized over the **Base Worktable**. The items which could not be placed onto the worktable will be shown as orphaned labware and can then be manually placed on or removed from the worktable.

8.1.9 Delta Level Setting of a Carrier or Labware Item

It is possible to define on which worktable a carrier or labware item shall be placed when it is dragged into the worktable editor. This is done by updating the custom attribute **WorkspaceDeltaLevel**

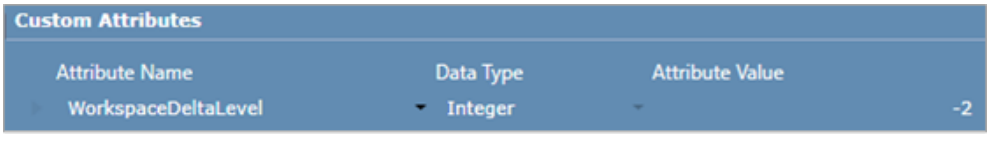


Fig. 74: WorkspaceDeltaLevel can be edited under Custom Attributes of the respective object.

Updating the delta level to the following values will place the item onto the corresponding worktable. Items which are already present on the worktable will keep their delta level and will not be updated retrospectively.

Tab. 2: Values for WorkspaceDeltaLevel and their meaning

Delta Level Description	Delta Level Value
Script Worktable	**1**
Base Worktable	**0**
Undefined	**-1** (default)
Instrument Worktable	**-2**

Saving a worktable when a labware is placed on a carrier that is on a higher layer is not possible and an message is displayed as shown below. The affected labware will automatically be placed on the base worktable when confirming the dialog, otherwise the user can move the affected labware by himself if the save operation is canceled.

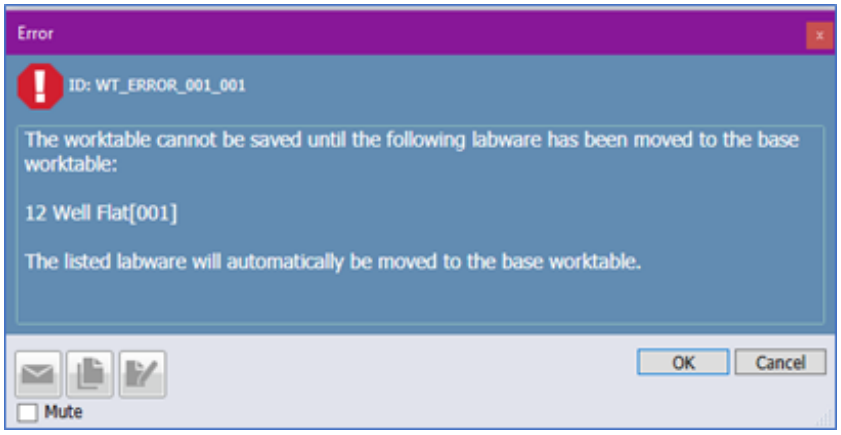


Fig. 75: Error message when trying to save a worktable with incorrect WorkspaceDeltaLevel values

8.1.10 Interconnections

The worktable editor is visible when any of the below are selected:

- **Script** editor
- **Carrier** editor
- **Labware** editor
- **Labware group** editor

8.1.11 Labware Group Editor

You can create, modify, and delete labware in the Labware Group Editor. The software provides a labware generator which will be used to specify the type and quantity of labware stacks on the worktable.

1. Drag and drop the labware on to the storage carrier to open the Labware Generator.
2. The Labware Group Editor opens.

3. Select the number of plates and the slot. White color represents occupied, blue color represents empty slots.
4. Enter a description for the assigned plate.

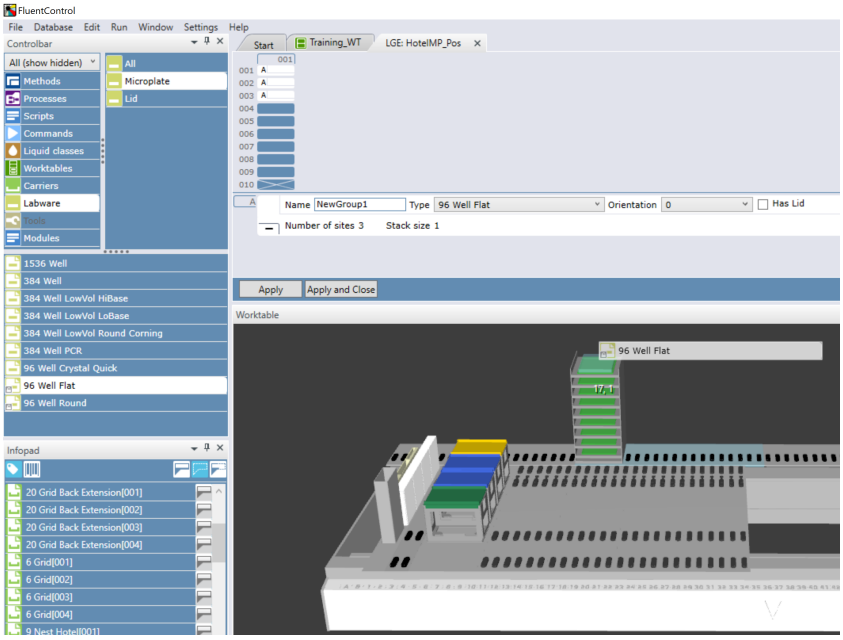


Fig. 76: Labware Group Editor

To create new labware with the Labware Group Editor, drag and drop the desired labware from the Controlbar on the left to the carrier. The Labware Group Editor automatically opens.

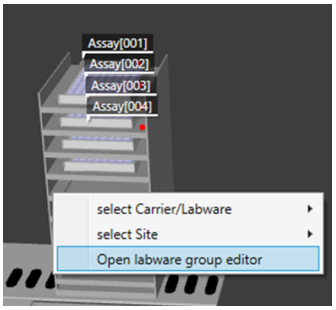


Fig. 77: Labware Group Editor / Carrier/Labware

Click the carrier sites in the graphic to create labware for the selected group. The selected labware group is highlighted in white below the list. If multiple carriers of same type exist on the worktable, the graphic displays multiple towers of carrier sites. Click to select individual carrier sites, or click and drag to select multiple carrier sites.

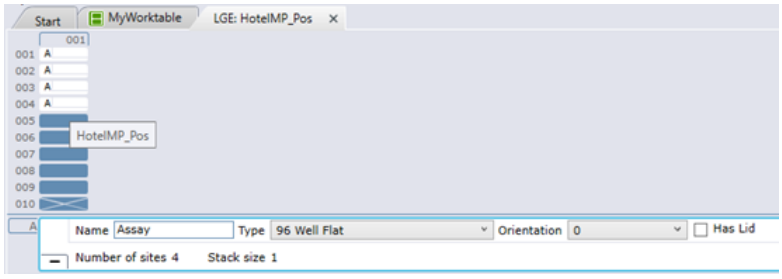
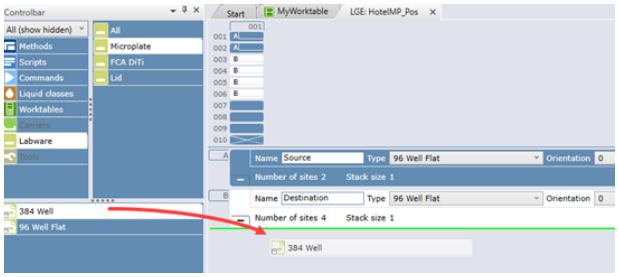
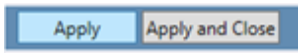


Fig. 78: Labware Hotel Slots

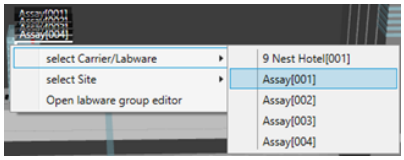
Assign a plate group name; the index numbers are generated automatically (i.e., 4 plates with the group name “Assay” results in 4 plates named Assay[001] - Assay[004] on the worktable). It is also possible to indicate the orientation of 0°/ 180° and whether or not a labware is covered with a lid. Note: For Fluent stacker towers, site number 1 displayed at the top of the graphic corresponds to the plate that comes out of the stacker first. To create different labware groups, drag and drop more labware to the list at the bottom. Each labware group is flagged as A, B, C, etc., as shown in the top of the graphic.



Click “Apply” or “Apply and Close” to create the labware, which is directly visible in the Worktable window. Clicking “Apply and Close” closes the tab automatically after creating the labware.



To select labware while editing a script, click on the carrier and then select the labware from the list that appears.



8.2 Carrier Editor

View or edit carriers with this editor.

1. Right-click the selected carrier in the Controlbar or the **Worktable** editor.

Carrier opens for editing.

2. Click **Open** on the Controlbar shortcut menu.
3. Or click **Edit** on the **Worktable** editor shortcut menu.

8.2.1 General Settings

For related information, refer to section Available Drivers Enabler.

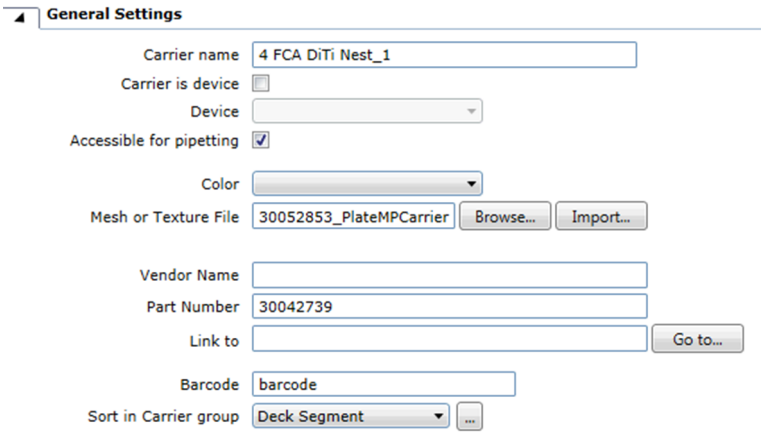


Fig. 79: General settings

Carrier name: Enter the **Carrier name**.

Carrier is device and Device:

1. If you select the **Carrier is device** check box, you can click the **Device** arrow to see more options.
2. You can select the device type from the list.

   If your device is not on the list yet, enable the device driver first.

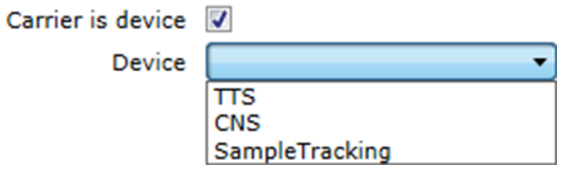


Fig. 80: Select Carrier is device

Accessible for pipetting: The **Accessible for pipetting** check box indicates that in these positions the labware should be accessible to the FCA pipetting arm. This is primarily used for the logic of the Automatic DiTi Handling commands. Select this check box if the carrier can be accessed by a FCA pipetting arm to mount disposable tips (DiTis).

*Clear the **Accessible for pipetting** check box to let the **Replace Used DiTi Racks (RGA)** command automatically replace used racks from this carrier. The **Replace Used DiTi Racks** command will only use labware placed on carriers that are not*

accessible for pipetting when searching for new (full) DiTi racks. If no full racks with the specified DiTi type are found on carriers that are not accessible for pipetting, the command will only remove the old (empty) DiTi rack without replacement.

Color: If you like to use a color to make it easier to identify this carrier in the Worktable Editor, click the Color arrow to see more options. Colors only impact meshes that do not have embedded colors. They do not impact textures.

Mesh or Texture File: A mesh or texture file is used for the visualization of the carrier in the **Worktable** editor. Click Browse and select an existing mesh file or click Import to select texture file(s) to import (X3D graphics or .JPG images).

Vendor Name: You can enter the name of the carrier vendor.

Part Number: You can enter a part number for the carrier.

Link to: You can enter a link for the carrier type. The link may reference a webpage or a file and is displayed in the Infopad when selecting the carrier type. Click **Go to** to test the specified link.

Barcode: This field is currently not used by the software.

Sort in Carrier group: You can select the group in which to display the carrier in the Controlbar. Click the **Sort in Carrier group** arrow to see more options. To add a group to the list, click the ... button, then enter the new group name, and click  .

8.2.2 Dimensions

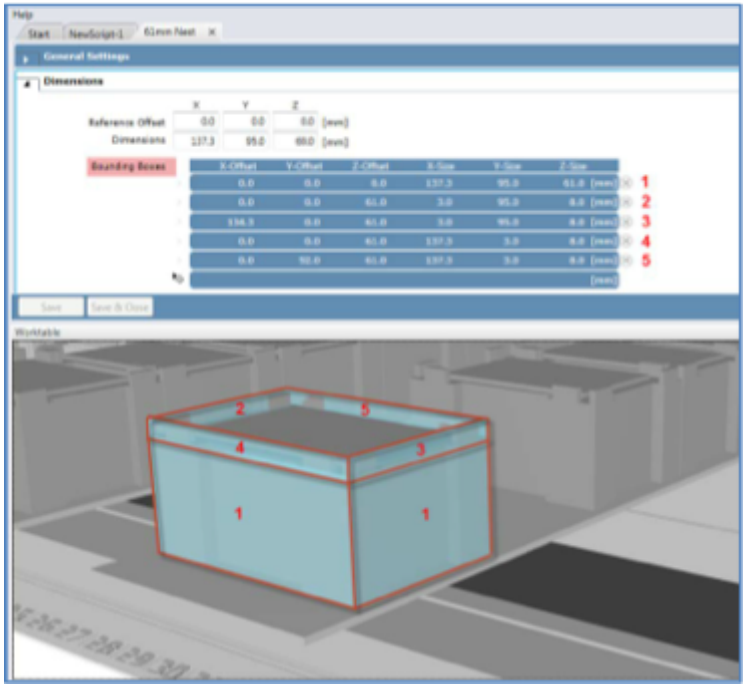


Fig. 81: Dimensions: Definition of bounding boxes that define an object.

Reference Offset: You can specify the carrier reference offset measured in millimeters. The X reference offset is the distance between the left edge of the carrier and the left edge of the grid or site that the carrier is sitting on. The Y reference offset is the distance between the front edge of the carrier and the front edge of the grid or site that the carrier is sitting on. The Z reference offset is the vertical distance between the top surface of the worktable or site the carrier is sitting on and the top of the carrier.

Dimensions: You can specify the outer carrier dimensions in x, y and z measured in millimeters.

Bounding Boxes: You can define a set of bounding boxes representing the carrier outlines. Each bounding box definition consists of the dimensions and the offset from the carrier origin all measured in millimeters. Bounding Boxes are presented to you as a table with the following columns: **X-Offset**, **Y-Offset**, **Z-Offset**, **X-Size**, **Y-Size**, **Z-Size**. See the MCA384 DiTi Active Nest for an example of additional bounding boxes.

8.2.3 Placement of Non-Grid-Based Devices

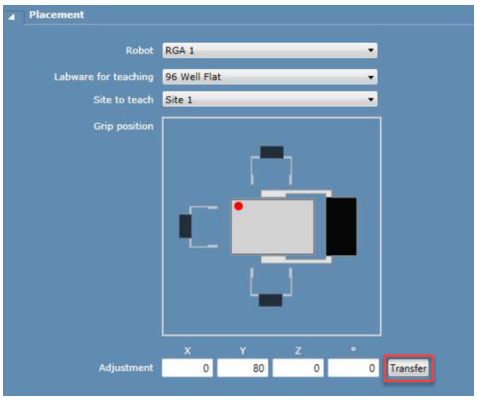
Devices such as shakers and incubators are usually integrated directly on the worktable (grid-based), while devices such as readers and washers are often integrated outside the worktable (non-grid-based), either next to or underneath the worktable. The Placement section is not present by default in the Carrier Editor, but will only appear automatically when a non-grid based device is placed on the workable.

Robot: You can select the robot arm to use for teaching from a list of available robotic arms.

Labware for teaching: You can select a labware type to use for teaching from a list of labware types allowed on the device.

Site to teach: You can select the access site of the device, in case the device has more than one access site.

Grip position: The grip mode defines how the robot arm is approaching the labware. You can select the grip mode of the robot arm during teaching, by clicking on the appropriate robot arm picture in the Grip position figure. The figure shows the relationship of well (red dot) **A1** to the position of the fingers (grip mode).



Prerequisites for Non-Grid-Based Carrier Definitions

- Custom attribute **IsGridBased** is unchecked.
- Reference offsets are set to zero.
- “Allowed Locations the deviceSite 01-50” must be checked.

Teach Location and Orientation of the Device Carrier

The Placement section opens up when you integrate a non-grid based device. Refer to section “Fetch Labware for Teaching (RGA)” [▷ 215].

1. Start the FluentControl application.
2. Fetch labware for teaching via a direct command.
3. Drag and drop a non-grid-based carrier, such as a reader, on to the worktable. The **Placement** section in the device editor opens.
4. Activate **Zero-G** for the RGA and move it to the end position on the plate holder of the device.
5. Click **Transfer** under **Adjustment**.
6. Save the carrier definition. As a result, the positioning of the device on the worktable updates.
7. Go to the **Robot Vector** section and transfer the current position also as the **End Position** of the vector.
8. Teach safe and intermediate waypoints for RGA vector.
9. Return the labware for teaching via a direct command.

Placement only teaches and defines the location of the device in the workspace. A device-specific vector must still be manually defined in the **Robot Vectors** section. It is important that the **Adjustment** in the **Placement** section is the same location as the **End Position** of the vector in order to prevent Pathfinder issues.

The Pathfinder does not work underneath the deck. Therefore, the first waypoint of the RGA vector (the safe position) must be defined above deck level.

8.2.4 Site Definitions

A site is a defined space for labware, tools and other carriers. Properties of the sites are defined in the site definition window, including number of sites per carrier and their sizes as well as which kind of vectors are allowed as well as which grip modes are allowed to move the objects on the sites.

By default, the **Transfer Labware** command supports all grip modes. The priority however is given to the narrow vector. If you want to change this setting, deselect any grip modes that you do not want to be used for that particular site. The site coordinates are relative to the carrier's 0 point defined by its reference offset.

All dimensions in the table are specified in [mm]. The rotation is specified in [°]:.

Select robot: Click the **Select robot** arrow to see more options. The instrument must be equipped with at least one RGA otherwise the list is empty.

N, NR, W, WR: **N**, **NR**, **W**, **WR** are called the Grip Modes: **N** = Narrow, **NR** = Narrow Reverse, **W** = Wide, **WR** = Wide Reverse. Check marks in a Grip Mode indicate it is allowed at that site.

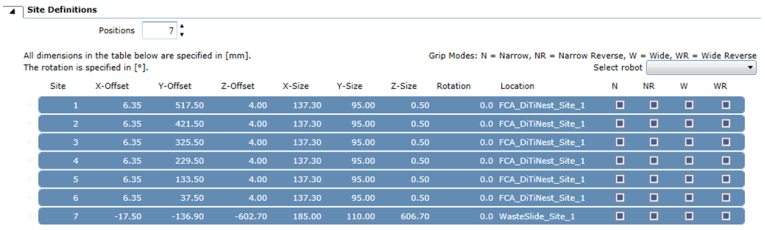


Fig. 82: Site definitions

Site Definitions are presented to you as a table with the following columns:

- **Site**
- **X-Offset**
- **Y-Offset**
- **Z-Offset**
- **X-Size**
- **Y-Size**
- **Z-Size**
- **Rotation**
- **Location**
- **N**
- **NR**
- **W**
- **WR**

Positions: Specify how many sites the carrier has.

If multiple carriers are stacked (e.g. Nests on Base Plates), the Grip Mode used will be the Grip Mode that is allowed by all carriers in the stack. For example, site 1 of the base plate may only have Wide selected. The Nest may have all 4 selected. Transfer Labware will only use the Wide vector at this site as it is the only one that is enabled for the base plate and nest.

The selection of the vector occurs before the Path Finder is asked to move an arm. Potential collisions or impossible moves are not considered during this determination.

Any given layer that has no selected Grip Modes will not impact the determination of the allowed Grip Modes.

If a source position and a target position of a Transfer Labware command do not share allowed Grip Modes, a Regrip Nest which allows both Grip Modes must be present somewhere on the worktable.

You can specify the size and positions of the sites in x, y and z. It is also possible to group the sites by editing the Location name. You can specify the dimensions and offsets for each of the sites of the carrier.

8.2.5 Robot Vectors

During Robot Vector teaching, Pathfinder and collision detection are not always active. To avoid collisions, please carefully read the paragraph ***Move to first; Move to last; Move to selected****.*

This is the vector teaching section of the **Carrier** editor.

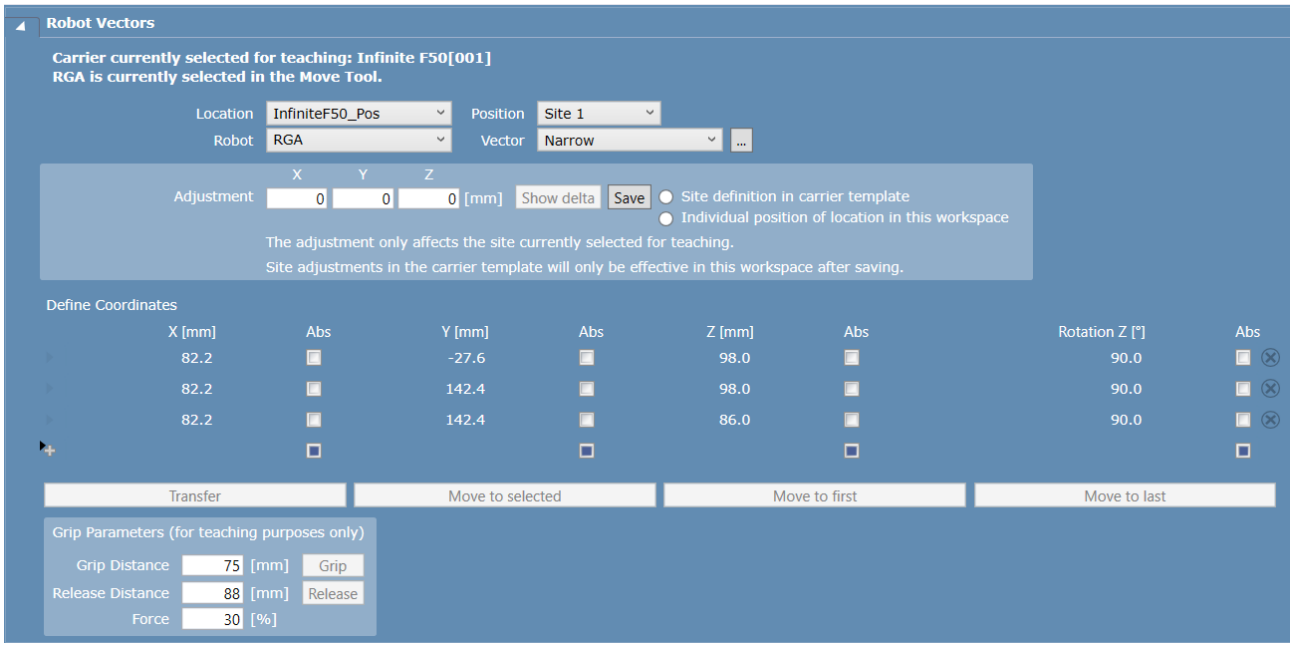


Fig. 83: Robot Vectors of the 7 mm Regrip Nest

Location: A location is a logical group of sites that can split a carrier or combine sites. Click the **Location** arrow to see what locations are available for teaching robot vectors. Select the entry from the list that you want to use.

Position: A position is a 3-dimensional numbering of positions within a location. Click the **Position** arrow to see what positions on the carrier are available for teaching robot vectors.

Robot: Click the **Robot** arrow to see more options. The instrument must be equipped with at least one Robotic Gripper Arm (RGA), otherwise the list is empty.

Vector: Select which Grip Mode vector shall be taught for the selected robotic arm. Click the **Vector** arrow to see more options.

The following default vectors are available for teaching:

- **Narrow**: Labware gripped in portrait orientation.
- **NarrowReverse**: Same as Narrow but Reverse = gripper rotated by 180°.
- **Wide**: Labware gripped in landscape orientation.
- **WideReverse**: Same as Wide but Reverse = gripper rotated by 180°.
- **Custom vectors**: You can teach custom vectors. Click  to add a new vector name.

Adjustment: It is possible to adjust the vectors for each site of a carrier, for example if one of the sites deviates in shape from all others. Two ways of executing the adjustment are possible:

- To change the values for the selected site in all carriers of this type, click the Site definition in carrier template radio button.

Note that with this option, this adjustment is stored only in the local database, not on the connected instrument.

- To change the values for the selected site only in the current worktable, click the Individual position of location in this workspace radio button. With this option, the adjustment is lost when the instance of this carrier is removed from the worktable. With this option, the site adjustment will be stored on the connected instrument (as described in chapter 9.1 - Teaching) and will be applied to all worktables that have the same type of carrier at the same location.

Adjustments applies only to the site which is selected for teaching.

1. To update the values, click **Show delta**.

   *The adjustment is displayed in the **X**, **Y**, and **Z** boxes in [mm].*

2. To apply the new values, click **Save**.

Define Coordinates: The robot vector is displayed as a list of coordinates in the **Define Coordinates** table. The table that represents coordinates has the following columns: **X** [mm] / Abs, **Y** [mm] / Abs, **Z** [mm] / Abs, Rotation **Z** [°] / Abs.

Vector positions are displayed relative to the site and will not match the Move Tool coordinates, which are absolute. Vector positions are plate based, not tool or arm based.

It is important that the last coordinate in the list of the default vectors has a relative value of almost 0 for X,Y and Z to avoid pathfinder issues.

Possible reasons why the value differs from 0 after teaching:

- Wrong site definitions of the current carrier or of the deck/grid segment the carrier is placed onto.
- Non-grid-based device was not teached properly or needs to be re-teached because it moved slightly because of vibrations.

Now, select a row, you want to edit.

Transfer: To include the current coordinates of the robotic arm in the **Define Coordinates** list.

1. Click **Transfer** and let FluentControl do the math for you (convert the actual position to a relative value to be stored).

   You can update coordinates of the coordinates list, or you can add new coordinates at the end of the list.

2. To add new coordinates, click , select the blank row at the end of **Define Coordinates**.
3. Click **Transfer**.

   For convenient teaching, use the robotic arm in Zero-G mode and move it by hand into the desired position.

   Alternatively, you can move the arm with the help of the buttons provided by the Move Tool or the numeric keypad.

Move to selected: To move the robotic arm directly to the selected coordinates (an entry in the Define Coordinates list above), click **Move to selected**:

- No collision checking is done!
- Use **Move to selected** only if the robotic arm is near the selected coordinate.
- Make sure that there are no obstacles in the way!

Move to first; Move to last; Move to selected: To move the robotic arm to the topmost entry on the **Define Coordinates** list, click Move to first. Path Finder will be used, if the robotic arm was not moved to any of the other coordinates, thus avoiding collisions with worktable objects. If the robotic arm was moved to any entry on the list before, the arm will move directly, without Path Finder support. It will move along each entry on the list and finally to the coordinates of the topmost entry. Use **Move to first** in combination with **Move to last** to test the whole robot vector.

To move the robotic arm to the coordinates at the end of the **Define Coordinates** list, click **Move to last**. If the robotic arm was not moved to any other entry on the list, the arm will first move to the coordinates of the topmost entry using Path Finder thus. After that, it will move to the coordinates of the next entry without Path Finder support until the coordinates of the last entry are reached. If the arm was moved to another entry on the list beforehand, it will move directly, without Path Finder support. The arm will move along each entry on the list until the coordinates of the last entry are reached.

If you use **Move to selected**, **Move to first**, or **Move to last** with the door open, the arms will move slowly. The arms will move at full speed if the safety shield is closed.

Grip Parameters (for teaching purposes only)

Use the Grip Parameters (described below) for gripping and releasing labware while teaching. The gripper width is not stored as part of a robot vector. The commands open the grippers automatically to the necessary width. But while teaching, the user is responsible for opening the grippers to an appropriate width. Otherwise, the grippers might collide with the target carrier or labware placed there.

These settings are for teaching only and will not be used in the script commands. The script commands will use the values stored in the labware that is moved.

Grip Distance: To move the fingers from the current position to this position unless the grip force is reached in between, click **Grip**.

Release Distance: To move the fingers from the current position to this position, click **Release**.

Force: Specify the grip force relative to the grippers capabilities that will be reached when gripping.

⚠ CAUTION

Moving parts!

Moving parts can cause crushing or cutting injury.

- Always keep hands and body away from moving parts while teaching with open safety panels.
- Take unintended motion of robot arms into consideration while teaching with open safety panels.
- Do not attempt to circumvent the door sensors.

✓ You can store several robot vectors per carrier.

✓ Open the gripper to an appropriate width before moving.

1. Select a carrier on the worktable to be used for teaching.

   *The text at the beginning of the **Robot Vectors** section tells you the carrier currently selected for teaching, otherwise the following phrase is displayed: Please select a carrier on the worktable to be used for teaching ... .*

2. Select a robotic arm in the Move Tool.

   The second sentence tells you the robotic arm currently selected in the Move Tool.

8.2.6 Allowed Labware Types

This section contains information which labware is allowed on the carrier. You can select the labware types which may be placed on the currently edited carrier. This is what determines the highlighted areas in the worktable editor while placing labware.

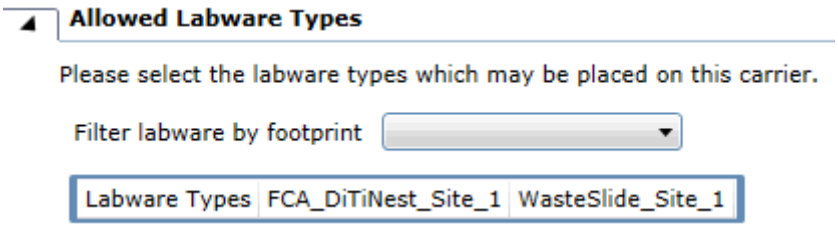


Fig. 84: Filter labware by footprint

Filter labware by footprint: You can specify the footprint of the labware. Click the **Filter labware by footprint** arrow to see more options. The list contains the different footprints of all labware types present on the system.

8.2.7 Allowed Locations

This section contains information on which location the carrier is allowed, including worktable grids or other carriers. You can select the locations on which the currently edited carrier type may be placed.

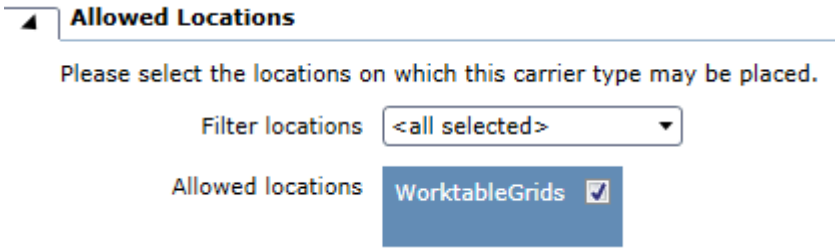


Fig. 85: Filter locations

Filter locations: Click the Filter locations arrow to see more options. The selection made in the **Filter locations** list (e.g., <all selected>) has an influence on the **Allowed locations** list below.

Allowed locations: The **Allowed locations** list shows on which location the carrier or labware is allowed. Each entry comes with a check box, which you can clear or select. To narrow down the list, use **Filter locations**.

8.2.8 Custom Attributes

You can define custom attributes as parameter on carrier or labware to influence specific system (also robot) actions.

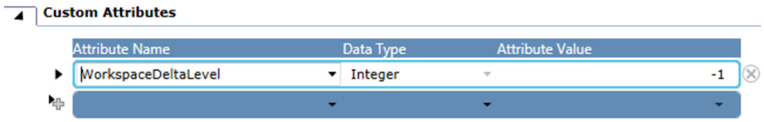


Fig. 86: Custom attributes

1. To delete a custom attribute, click ⊗.
2. Click on row to the custom attributes table.
3. Click the **Attribute Name** arrow to see all attributes:

- **AccessLocation**: Location to access labware for transfer from storage device.
- **BoundingBoxVisible**: Shows the BoundingBox of a carrier.
- **Grasp_location_max**: Maximum value for allowed grasp.
- **Grasp_location_min**: Minimum value for allowed grasp position range.
- **Grasp_location_N_max**: Maximal plate grip offset narrow.
- **Grasp_location_N_min**: Minimal plate grip offset narrow.
- **Grasp_location_W_max**: Maximal plate grip offset wide.
- **Grasp_location_W_min**: Minimal plate grip offset wide.
- **Grip_max**: Maximal gripper distance for labware on this carrier.
- **Grip_min**: Minimal gripper distance for labware on this carrier.
- **IOState**: I/O state of associate tool driver (tool location). If IOState is disabled, then the system does not communicate with the tool location driver.
- **IsFrameElement**: Set object as part of the worktable frame.
- **IsGridBased**: Define if a worktable is grid based or not.

- **Lid**: Whether a lid is allowed on the labware place on this carrier.
- **Lid_location**: Location to store the lid.
- **Lid_position**: Position of lid within lid location.
- **LiddingStation**: Carrier that is allowed to cover or uncover a lid from a labware.
- **OpenCarrierAndSiteList**: Open labware and site list.
- **OpenLabwareGenerator**: Open labware generator.
- **PassiveStack**: Carrier is treated as a passive stack.
- **RegripStation**: Location to use for re-gripping.
- **RenderVisualChildren**: Define if children objects of a worktable object shall be rendered.
- **RobotMove**: On-The-Fly Tools: To enable/disable robot move.
- **Server**: Call name of the device (internal name of driver).
- **Speed**: Maximum speed of robotic arm in percentage of full speed.
- **StartRack**: Defines first position of a group of stacks for the carousel.
- **StartSqueezePosition**: In the context of pipetting for FCA and MCA384.
- **Storage**: Defines if the lid shall remain at the location.
- **WorkspaceDeltaLevel**: Preferred workspace delta level for this labware.
- **WasteLocation**: Location is waste location.
- **PiercingDetectionHeightCompenstation:** In the context of piercing applications to compensate detection height (in mm) due to mechanical deformation.

8.3 Labware Editor

Tecan endeavors to ensure the correctness of the default labware definitions delivered with FluentControl. Many labware definitions are based on a specific manufacturer's part that is listed in the software. However, for non-Tecan labelled labware, these dimensions can change, so you are advised to check that the labware you are using is in agreement with the labware definition and validate your process using the labware you plan to use.

If you use different labware, you must carefully check that the labware definition in the software agrees with the labware you are using. Incorrect compartment definitions can lead to air aspiration or crashes.

View or edit Labware with this editor.



Fig. 87: Labware Editor

8.3.1 General Settings

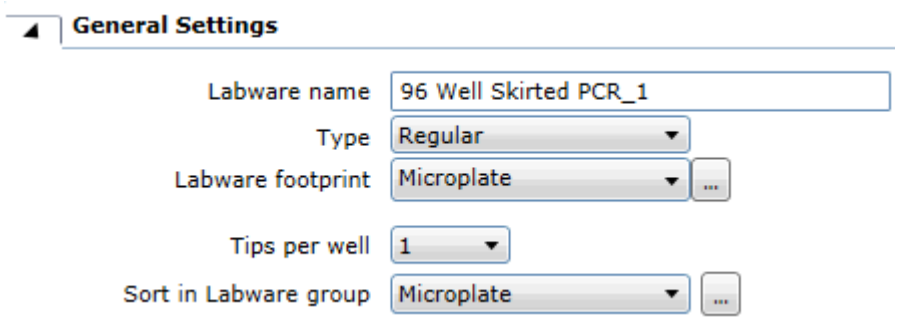


Fig. 88: General settings

Labware name: Defines the display name of the labware.

Type: Click the **Type** arrow to see more options.

Labware footprint:

1. Select the item to which the currently edited labware should be mapped in **Allowed Labware Types**.
2. Click the **Labware footprint** arrow to see more options.
3. If you need another form factor, click ... **to add a new item**.

Tips per well: This parameter is available only in a **Labware** editor. Click the **Tips per well** arrow to see more options.

Sort in Labware group: Defines where the labware should appear on the **Controlbar**.

Tool ID

This parameter is available only in a **Tools** editor.

8.3.2 Positions / Teaching

⚠ CAUTION

Moving parts!

Moving parts can cause crushing or cutting injury.

- Always keep hands and body away from moving parts while teaching with open safety panels.
- Take unintended motion of robot arms into consideration while teaching with open safety panels.
- Do not attempt to circumvent the door sensors.

You can teach the positions of a labware on the worktable.

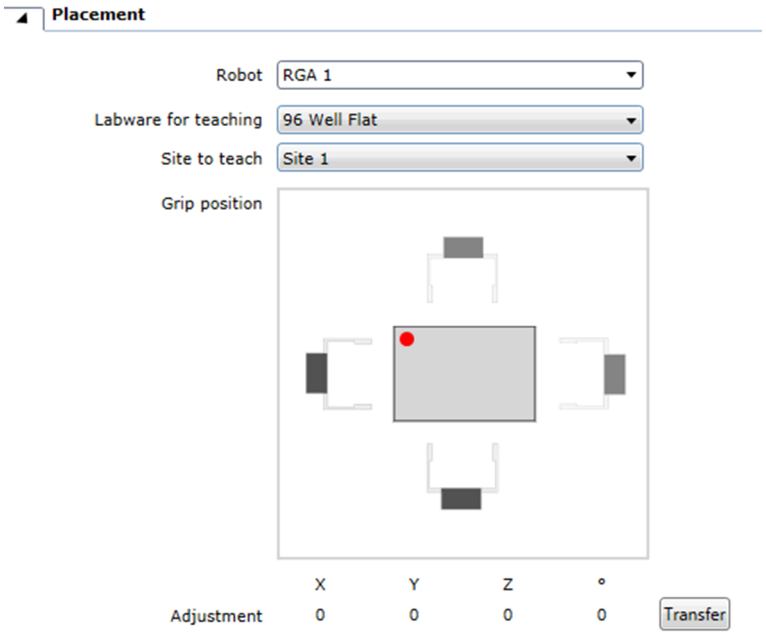


Fig. 89: Positioning / Teaching

Dimensions: You can specify the outer labware dimensions in **X**, **Y** and **Z** measured in millimeters.

Bounding Boxes: The **Bounding Boxes** of the currently edited labware are presented to you as a table with the following columns: **X-Offset**, **Y-Offset**, **Z-Offset**, **X-Size**, **Y-Size**, **Z-Size**. You can add a row to the table and delete one.

Number of wells: These two fields specify the well geometry:

(X): Specify the total number of wells in the X-direction (max. 128).

(Y): Specify the total number of wells in the Y-direction (max. 128).

Well alignment: Select the style of well alignment you want to use. Click the regular or the interleaved option button (also known as radio button).

Type	Name	Comment
	Rectangular labware (regular)	This is labware where the wells are arranged in regular rows and columns (e.g., a standard microplate). The first well is rear left; the last well is right front.
	Rectangular labware with offset (interleaved)	This labware is similar to the rectangular type, but every second column is displaced in the +Y-direction. The spacing of the wells in the Y-direction is the same for all columns. In addition to specifying the X- and Y-positions of the first and the last well, you must also specify the Y-position of the first well in the second row (position #2).

Compartments per well: Some special labware types (e.g. for protein crystallography) have wells with several compartments (cavities). FluentControl can pipette a different liquid into each of the compartments. Each well typically has one main compartment and one or more satellite compartments which are directly adjacent and are often smaller in size. Specify the number of compartments per well in this field. For normal labware, select **1**.

Please select a labware on the worktable to be used for teaching ...

Current compartment: For labware with more than 1 compartment per well, select the Current compartment to be used for teaching.

Pipetting positions: All values are specified in [mm].



The adjustment only affects the site hosting the currently selected labware.

1. Click **<< Transfer**.
2. Use the robotic arm in Zero-G mode and move it by hand into the desired position.

   Alternatively, you can move the arm with the help of the buttons provided by the Move Tool or the numeric keypad.
3. Click **Move to**.

   The robotic arm moves to the transferred coordinates.
4. Click **adjustment** > **Adjust Site** > **Show delta**.

5. Slightly alter the site settings of a custom carrier below the labware in order to achieve the desired fit.
6. **Z-Coordinates:** This part of the visualization shows the actual Z-positions: **Z-Travel**, **Z-Start**, **Z-Dispense**, **Z-Max**, and **Z-Bottom**.

Use **<< Transfer** and **Move to** to teach the Z-coordinates corresponding to the description above.

Y-distance of wells (Visualization):

If the pipetting operation specifies the use of more than one tip, FluentControl adjusts the spacing between the tips in the Y-direction to allow pipetting to take place on several wells simultaneously if possible. The tips are spread by a servo motor on the FCA and the spacing is subject to physical limitations. The minimum spacing between adjacent tips is 9 mm and the maximum spacing is 38 mm, as shown by the light green area in the middle of the indicator. FluentControl calculates the Y-distance (the spacing between adjacent wells) from the labware coordinates and the number of wells you have specified. This is shown by a vertical line which is colored either dark green or dark red. To optimize pipetting speed, the vertical line should lie within the green area. If the vertical line lies within the red area, you may have specified the coordinates incorrectly or inaccurately. Alternatively, you are using labware with a non-standard design. The yellow line shows half of the minimum Y-distance (i.e., 4.5 mm). If you are using microplates with 384 wells and choose this Y-distance, pipetting can take place simultaneously to every other well.

Labware Coordinates: You can define the following different heights (Z coordinates) for the wells.

The coordination system is defined in FluentControl as follows:

- Front, Left, Worktable level of the instrument is the 0,0,0 point.
- Positive X is to the right.
- Positive Y is to the back and Positive Z is Up.
- Negative values are the opposite directions.

See also The pipetting instrument's coordinate system. Labware coordinates are always relative to the carrier. Therefore, coordinates can only be edited if labware is placed on carrier. Values must not be entered manually but added by the **<< Transfer** button.

The following table describes the X, Y and Z coordinates in the positions/teaching section of the **Labware** editor:

X, Y and Z coordinates for labware:

Coordinate	Position / Function
X,Y	At this position, the tip is situated above the well/tube of the first/last position on the labware.

Coordinate	Position / Function
Z-Travel	Tip height at which the tips move from one **X**/**Y** position to another. Choose **Z-Travel** to be higher than all obstacles on the carrier. Movements that pass over different pieces of labware use a tip height of **Z-Travel** (global) instead.
Z-Start	Tip height at which the liquid level detection (ILID) is switched on during pipetting. For tubes and containers (troughs), choose **Z-Start** to be at least 1 mm below the rim but above the liquid surface. For microplates, choose **Z-Start** to be at least 1 mm above the edge of the well.
Z-Dispense	Tip height at which liquid is dispensed. It should be low enough to ensure that no liquid drops can get into adjacent wells or tubes.
Z-Max	Tip height which is very slightly above the lowest point of the well, tube or container. It is the lowest position which the tip is allowed to reach without touching the bottom. When searching for the liquid surface, the pipetting instrument will search from **Z-Start** down to **Z-Max**. If the tip reaches **Z-max** without finding liquid, the pipetting instrument reacts according to the configured liquid level detection error mode. **NOTICE! If "Aspirate Anyway" is selected as an error handling option, the tip will not track lower than Z-Max during aspiration.**
Z-Bottom	It defines physical bottom used for volume calculation.

8.3.3 Compartment Definitions

The compartment definition reflects the inner geometry of one well or cavity. It is used to calculate liquid volume from a detected liquid level. Therefore in order to have an accurate liquid volume calculation it is recommended to have accurate and precise compartment definitions.

Compartment Definition: Example with 3 shapes (Cylinder, (Truncated) Cone and Partial Sphere).

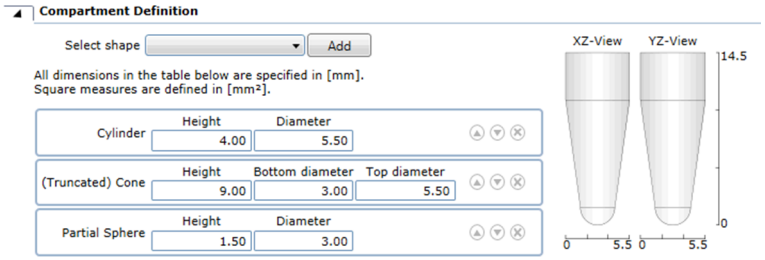


Fig. 90: Compartment Definitions

Select shape:

There are several predefined shapes provided. Depending on the selected shape different parameters are needed. Alternatively a user defined option is available too. This option allows to define a matrix of known height-volume combinations (input either directly or via a file). A cavity can be defined using more than one shape as in the example here. Use the Add button to add more shapes, the x button at the end of each table entry to remove the shape or alternatively the up and down buttons to move the entries.

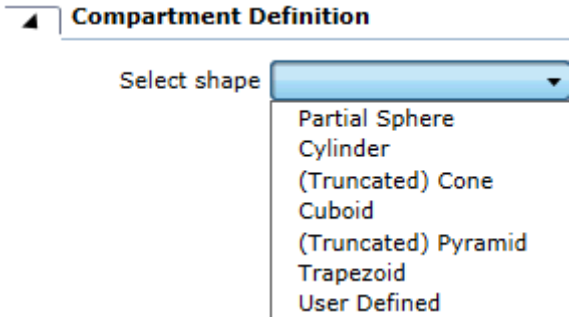


Fig. 91: Select shape arrow

8.3.4 Miscellaneous

This section provides options to adjust the visual appearance of the labware by giving it a (different) color or define a mesh or texture file and to define vendor, part numbers and links to websites.

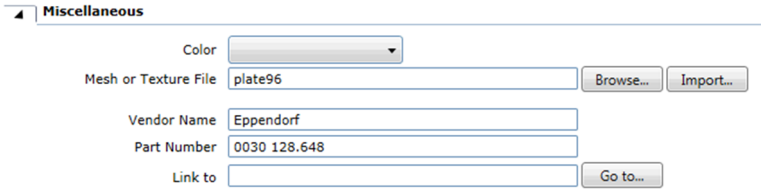


Fig. 92: Miscellaneous

8.3.5 Allowed Labware Types

In this section the labware that can be placed on top of the selected labware can be defined. For plates for example other plates can be placed in order to stack or alternatively lids can be placed onto plates.

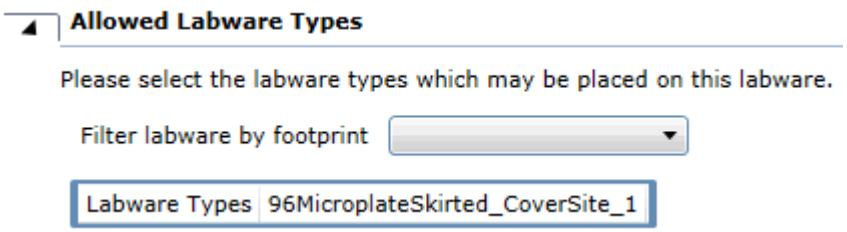


Fig. 93: Filter labware by footprint

You can specify the footprint of the labware.

Filter labware by footprint: Click the **Filter labware by footprint** arrow to see more options. The list contains the different footprints of all labware types present on the system by selecting an option in the dropdown a table appears showing all selected labware that may be placed onto the selected CoverSite of the labware.

8.3.6 Allowed Locations

This section contains information on which underlying location the labware is allowed to be placed. These are mostly carrier sites but also labware sites. A filter allows to filter for all selected, all unselected or all options.

Allowed locations

The **Allowed locations** list shows on which location the carrier or labware is allowed. Each entry comes with a check box, which you can clear or select. To narrow down the list, use **Filter locations**.

8.3.7 FCA Specific

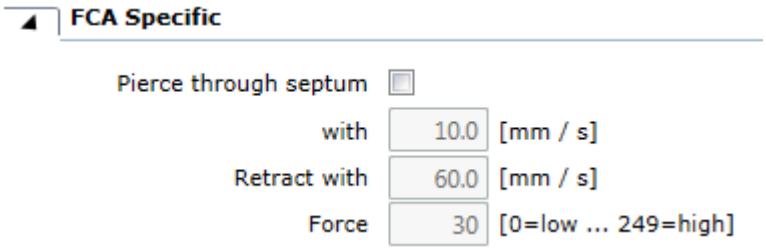


Fig. 94: Pierce through septum

Pierce trough septum: For piercing the septum with the FCA piercing tip(s), select the Pierce through septum check box. The rubber closes the puncture, providing an air and moisture-tight seal to protect the contents of the labware from the atmosphere.

with: Enter the piercing speed in [mm/s].

Retract with: Enter the retract speed in [mm/s].

Force: Enter the piercing force (0= low ... 249=high).

These parameters are optimized. Changes to these values can have an effect on piercing performance.

8.3.8 Robot Specific

This section is relevant if an RGA or a channel gripper is present on the system. Here gripping parameters are defined (only applicable to RGA), offsets between plates when they are stacked and whether the labware allows lids or not.

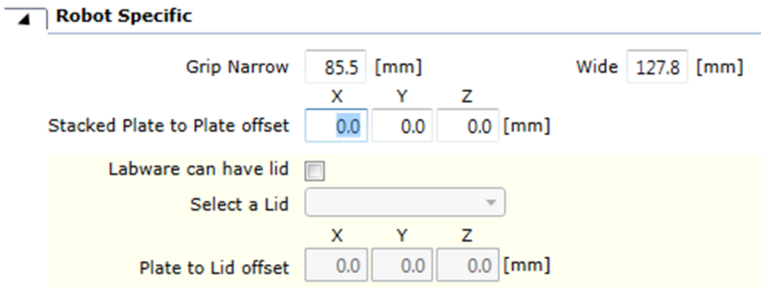


Fig. 95: Robot specific

8.3.9 Custom Attributes

You can define custom attributes as parameter on carrier or labware to influence specific system (also robot) actions. **Custom Attributes** are presented to you as a table with the following columns: **Attribute Name**, **Data Type**, and **Attribute Value**.

1. To delete a custom attribute, click ⊗.
2. Click on row to the **Custom Attributes** table.
3. Click the **Attribute Name** arrow to see all attributes:

- **ClldSensitivityCalculationLabware**: Defines the type used for the cLLD sensitivity calculation
- **ConnectorOffset**: Offset between DiTi and the DiTi adapter Head adapter: Mount point between head and Z axes
- **DisplayToolName**: User-friendly name for Infopad, Controlbar and dropdown lists
- **DiTi_type_group**: Group of DiTi type
- **DiTiToolType**: Identifier for the DiTi type
- **FirstTipPositionOffset**: The offset between the reference MCA384 tip adapter and all other adapters
- **Grasp_location**: Position where the labware should be gripped. This is an offset from the default grasp position

- **Grasp_location_N**: Position where the labware should be gripped when gripping narrow. Offset from the taught position for gripping the plate narrow
- **Grasp_location_W**: Position where the labware should be gripped when gripping wide. Offset from the taught position for gripping the plate wide
- **Grip_delta**: Distance added to the grip distance when releasing the labware or lid
- **GripperCenterOffset**: Offset from the rotation center to plate grip position
- **GripperSpreadOffset**: Measured between the grippers
- **GripperZOffset**: Z offset from the rotation center to plate grip position
- **HeavyObject**: Define if labware is a heavy object and shall have limited load speed
- **InitializationOffset**: The offset which exists during initialization between an axis with no mounted tool and mounted tool
- **IsFcaLiquidWaste**: Definition of the FCA liquid wastes, as fallback the plastic waste should be used
- **IsLiquidAspirateableIntoDevice**: The tool could be used to aspirate liquid
- **IsMcaLiquidWaste**: Definition of the MCA liquid wastes, as fallback, the plastic waste should be used
- **KeepForNewRun**: Indicates that the labware or tool instances of the last runtime workspace shall be taken when starting an new run
- **LeftGripperBoundingBox1Dimension**: Definition of bounding box for the gripper
- **LeftGripperBoundingBox1Offset**: Definition of bounding box for the gripper
- **LeftGripperBoundingBox2Dimension**: Definition of bounding box for the gripper
- **LeftGripperBoundingBox2Offset**: Definition of bounding box for the gripper
- **LeftGripperBoundingBox3Dimension**: Definition of bounding box for the gripper
- **LeftGripperBoundingBox3Offset**: Definition of bounding box for the gripper
- **LeftGripperBoundingBox4Dimension**: Definition of bounding box for the gripper
- **LeftGripperBoundingBox4Offset**: Definition of bounding box for the gripper
- **Lid_arrival**: Movement of the robotic arm to remove a lid of a labware
- **Lid_Force**: Force to grip lid
- **Lid_grasp_location_N**: Offset from the taught position for gripping the lid of a plate in portrait orientation
- **Lid_grasp_location_W**: Offset from the taught position for gripping the lid of a plate in landscape orientation
- **ListNumberInDropDownMenu**: Sorting rule in steps of ten for the list of value in the script command parameter "Adapterplate"
- **MCA384ChannelsPerTip**: Factor for script commands when EVA adapter is mounted
- **MCA96AlignmentMoveRelease**: A special AlignmentMove can be performed when picking up tips

- **MountColumnRowWise**: Define that partial tips with this tool is only possible column- or row-wise
- **MountOffset**: The offset which exists if a tip is mounted and reduces the z-dimension of the tip
- **NbrOfTipsX**: Maximal connectable tips within an adapter in X dimension
- **NbrOfTipsY**: Maximal connectable tips within an adapter in Y dimension
- **OffsetConeIntoTip**: Minus offset to calculate new Z dimension of tool after tip is mounted on the adapter
- **ReferenceWidth**: n/a
- **RenderVisualChildren**: Define if children objects of a worktable object shall be rendered
- **RightGripperBoundingBox1Dimension**: Definition of bounding box for the gripper
- **RightGripperBoundingBox1Offset**: Definition of bounding box for the gripper
- **RightGripperBoundingBox2Dimension**: Definition of bounding box for the gripper
- **RightGripperBoundingBox2Offset**: Definition of bounding box for the gripper
- **RightGripperBoundingBox3Dimension**: Definition of bounding box for the gripper
- **RightGripperBoundingBox3Offset**: Definition of bounding box for the gripper
- **RightGripperBoundingBox4Dimension**: Definition of bounding box for the gripper
- **RightGripperBoundingBox4Offset**: Definition of bounding box for the gripper
- **SafeDropGuideMoveOffsetForMCA384Tips**: Offset to safely move into a waste (with a drop guide) with MCA384 tips. Please note that the z-coordinate is ignored
- **SensitivityOffset**: Offset which shall be applied to cLLD sensitivity group when performing cLLD in labware of this type. No automatic robot actions or influence
- **Speed**: Maximum speed of robotic arm in percentage of full speed
- **Static_Labware**: Define the naming rule for the labware generator
- **TipCapacity**: Capacity of a single tip. Is used for context check in script commands when the user defines a volume to be aspirated
- **TipDimension**: The size of the tip in X, Y, Z
- **TipSpacingX**: Spacing between single tips on the tip adapter (tip block) in X dimension
- **TipSpacingY**: Spacing between single tips on the tip adapter (tip block) in Y dimension
- **ToolIdName**: Identifier for labware/tool
- **ToolType**: Type identifier for labware/tool; together with ToolIdName and ToolType there shall be an unique identifier
- **WasherPlateType**: Washer plate type
- **WorkspaceDeltaLevel**: Preferred workspace delta level for this labware
- **XYTrackingSequence**: Defines the x/y sequence for the microscript command **Aspirate Liquid with XYTracking**

8.4 Script Editor

The script editor is where protocols are being developed. The single script lines are also called commands and those are categorized based on arm and device within the Controlbar. The Script editor allows to develop new scripts from scratch, open existing scripts and adjust by changing order of the statements, adding new commands or deleting commands. Furthermore it allows to test and troubleshoot protocols by providing different tools (refer to chapter 8.4.2).

Do not use breakpoints in validated scripts. They are intended for development purposes only.

When the **Script** editor is open (you are viewing a script), the **Commands** section of the **Controlbar** only shows script commands.

Check for correct execution:

✓ Wrong Application data loaded or saved or deletion not working can cause wrong results.

1. Check the execution of the correct scripts before releasing data.
2. Check for correct execution data being loaded at end of run.

8.4.1 Declaring and using Variables

FluentControl requires explicit declaration of all variables in the Variables section, at the bottom of the editor. The Variables section is available in the Script, Process and Method editors. New variables can either be directly added and defined in the variable pane or within the commands that use the variables (e.g. Set Variable) by using the "Declare"-button. The **declare** button will make a new entry in the variable pane, which may need to be adjusted depending on type, scope, or default value.
FluentControl has 3 types of variables: string (letters and numbers), floating point (numbers with decimals) and integers (numbers without decimals). Setting the type of variable will change how the input screens appear on the touch monitor. String = Full Keyboard, Floating Point = Number pad with decimal point, Integer = Number pad with no decimal point.

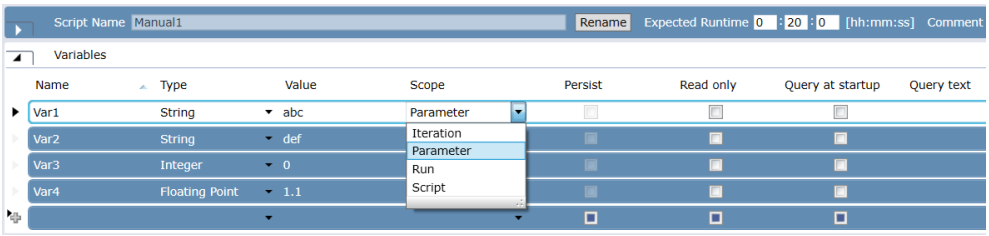


Fig. 96: Variables

Variables in FluentControl can have one of five scopes (**Run**, **Script**, **Iteration**, **Parameter**, and **Process**).

For more information on the use of variables by the FluentControl **Scheduler**, see section “Variables” [▷ 497].

Run: Variables with this scope are shared globally by all the scripts in a run.

Script: Variables with this scope are declared in scripts with the **Set Variable** command. They are not shared with other scripts. If you declare a variable with scope **Run** or **Iteration**, do not declare it again within a script. This will cause an error.

Iteration: Iteration scope is like script scope. However, these values reset to the defaults with each new iteration at run-time. Only impacts Methods that use Scripts or with more than one iteration.

Parameter: Variables of scope **Parameter** are automatically added to the "Transfer values to subroutine" section of the Module and Subroutine commands. The behavior of variables with scope Parameter are otherwise identical to variables with scope Script.

Process: Values of a variable with scope **Process** are only valid and accessible for a specific Process during the run but are not be accessible to other Processes in the run.

Persisted Variable Values

The application allows to persist variable values of variables with scope Run. The current values of these variables are stored at the end of each run and retrieved when a Method is created or loaded. Only variable of scope Run support persistence.
For other scopes the Persist check box is disabled. This feature is especially relevant for Scheduler Cell-based Assay workflows ("Support for Cell-Based Assays" [▷ 513]). If variable values shall not be lost when a run is stopped or the software is restarted, the affected variables can be set to persist their values.
To enable persistence, check the Persist checkbox in the variable definition in the Process or Script declaring the variable. If the Persist checkbox is unchecked, but a stored value is available, the user will be warned that the stored values will be deleted. If the user rejects the deletion, the check box will be checked again.

If a variable is not declared as persisted but a stored value matches the variable definition (name and type) its value will be used when a method is opened. This will be reflected in the Variables section of the Method editor by showing the Persist checkbox as checked.

8.4.2 Run Modes from the Script Editor

To help with script development, the run button in the script editor has four modes:

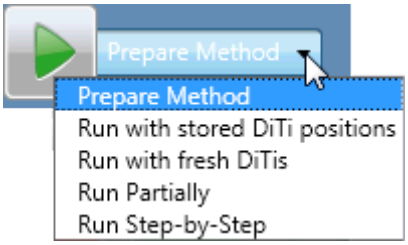


Fig. 97: Script editor

Run with stored DiTi positions: Run with stored DiTi positions resets all labware positions on the worktable but preserves disposable tip information for from last run, including the positions of tip trays, boxes, and layers in nested tip stacks. This applies to Flexible Channel Arm and Multiple Channel Arm tip types.

Run with fresh DiTis: Run with fresh DiTis resets all labware positions on the worktable and stored disposable tip information from the last run.

Run Partialexecute: **Run Partialexecutes** the current selected lines of the script.

Run Step-by-Step mode: Before the start of each command line in the script, the user must click continue in the runtime controller or the touch monitor.

Subroutines: If a synchronous subroutine is present in the script, the complete subroutine runs in step-by-step mode as well. If an asynchronous subroutine is present in the script, then, before the start of this command the user must click continue, but the asynchronous subroutine itself is not run in step-by-step mode.

It is expected that once a script is developed it will be prepared as a method and the operator will execute this method via the interface on the touch monitor.

In addition to these four run modes, FluentControl offers the ability to deactivate script lines or set breakpoints at script lines by right clicking on the command. Deactivated script lines and breakpoints will also be ignored in sub-routines.

Script run is not allowed to start if open editors have unsaved changes. A dialog box will ask whether to save unsaved changes. Press **yes** on the dialog box to save unsaved changes in all open editors and continue with run.

8.4.3 Context Check

When auto context check is observing something inconsistent, it will visually guide you to the respective position in the script. You will see a little red dot on the name part of the tab. Further, the script lines concerned will be marked with a red flag. You will get more specific info about what is inconsistent in the Infopad window, as well as hints and marks how and where to change it in the command itself (red).

In a script, there can be sections that are not reviewed by auto-context-check. The concerned command lines will be pointed out by a yellow flag and a warning in the **Infopad**.

You will get information in the worktable window as well about why you got a warning.

8.5 Method Editor

Create and edit methods with the **Method** editor. Use the **Method** editor to make methods appear to the operator on the touch monitor.

You can drag **Scripts** to your method.

1. Choose **Controlbar** > **Category** > **Scripts**.
2. Dropping the element in place.
3. Move the script to the place shown in the following figure.

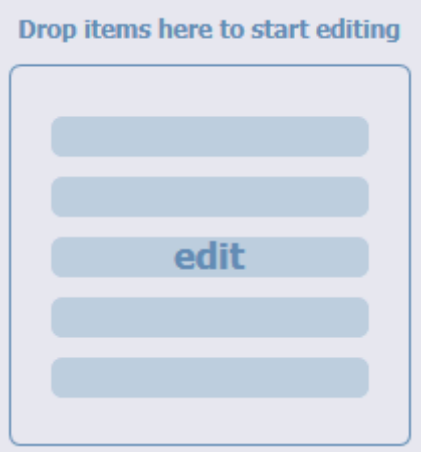


Fig. 98: Drag-and-drop feature: The edit zone

The next figure shows the editor after dropping the element in place.

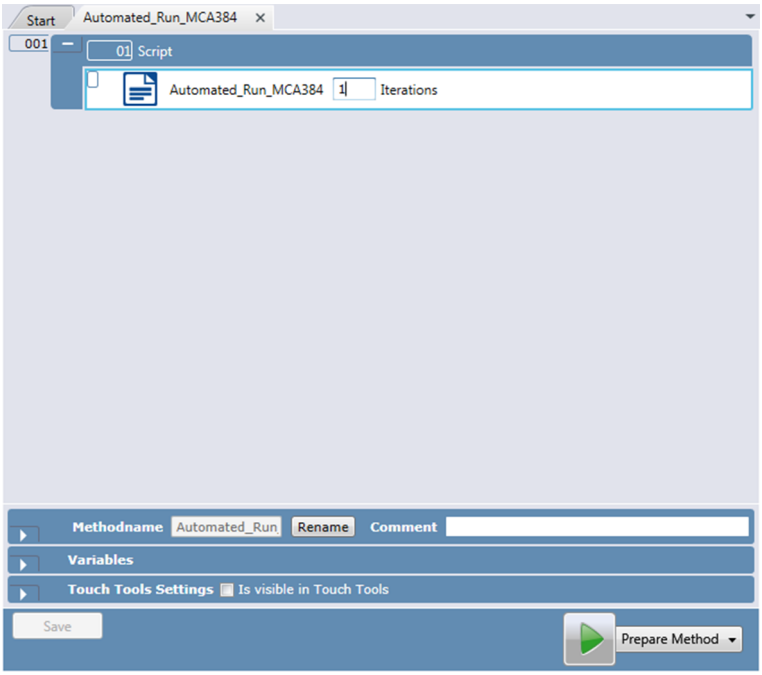


Fig. 99: Method editor with a dropped Script item

Although the edit zone is no longer visible but replaced by the dropped item, the drag-and-drop feature is still available. Think of it as an invisible drop zone in the background.

Iterations: Enter the number of iterations.

8.5.1 Method Properties

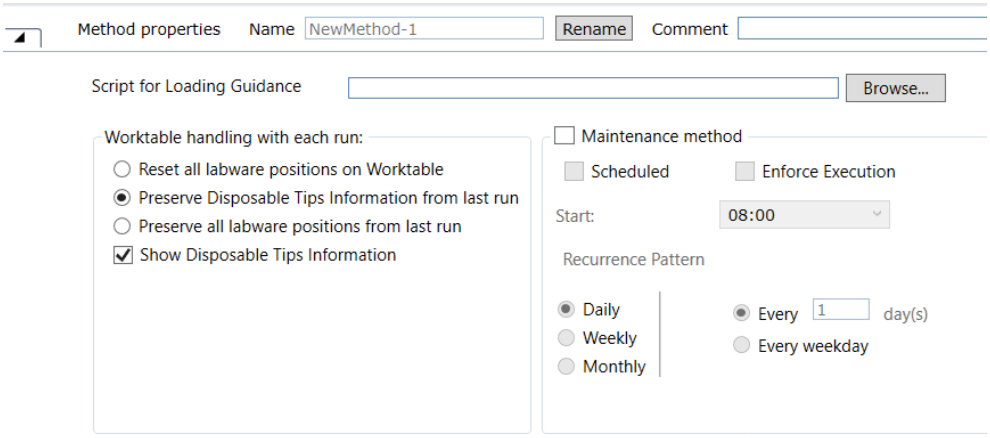


Fig. 100: Method Properties

Method Name: Rename the method name.

Comment: Add a Comment.

Script for Loading Guidance: The Script for Loading Guidance feature allows the user to define a script which is executed before the method is run. This script is intended to guide the operator through the necessary preparation steps before the method can be started. The instrument does not have to be initialized to execute the Loading Guidance script and the doors can remain open. Any commands which do not move the arms can be added to the Loading Guidance script, including DeckCheck commands.

Is visible in Touch Tools: You can look at the Variables and edit the TouchTools Settings. Select **Is visible in Touch Tools** to make methods appear to the operator on the touch monitor.

Maintenance method check box: A method will show up on the touch monitor under system care if you select the **Maintenance method** check box and additionally the **Is visible in Touch Tools** check box in the **TouchTools Settings** section below.

8.5.2 Touch Tools Settings

The **Touch Tools Settings** check box allows the method to be visible on the touch monitor.

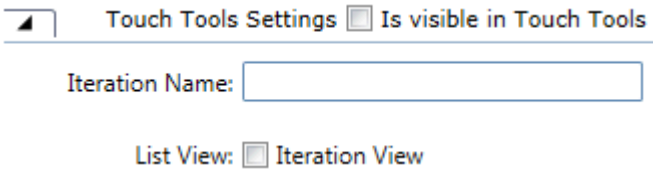


Fig. 101: Is visible in Touch Tools

Iteration Name: Optionally a name can be defined for the Iterations, which will then be visible in the TouchTools.

Is visible in Touch Tools: Select **Is visible in Touch Tools** to make methods appear to the operator on the touch monitor.

List View: To add the iteration controls to the method on the touch monitor when Prepare Method, select the **Iteration View** checkbox.

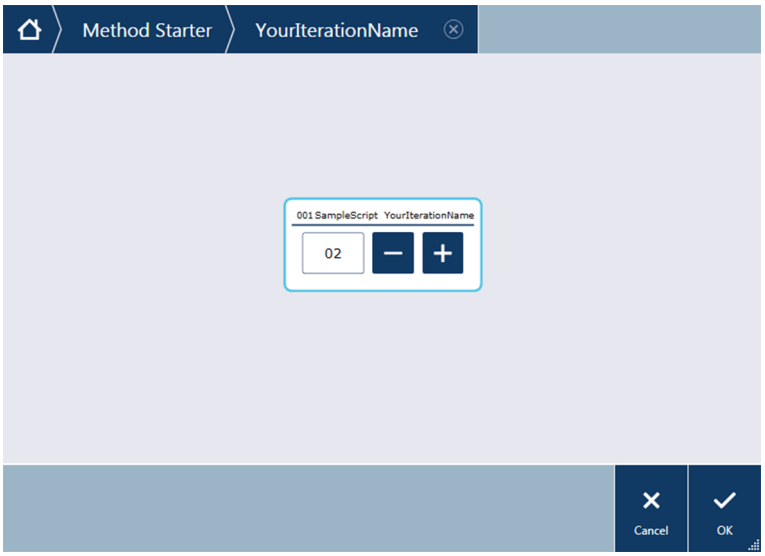
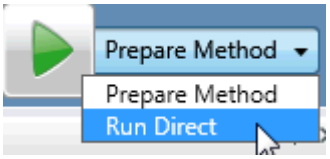


Fig. 102: Touch Tools: Example of a method with Iteration View

8.5.3 Prepare Method or Run Direct





Method run is not allowed to start if open editors have unsaved changes. A dialog box will ask whether to save unsaved changes. Press **yes** on the dialog box to save unsaved changes in all open editors and continue with run.

8.6 Liquid Class Editor

A liquid class defines the robotic actions that will take place during a **pipetting script** command.

Liquid classes are defined per liquid type and pipetting mode (e.g., single pipetting or multi-pipetting, free dispense or wet contact). One liquid class covers the whole volume range for FCA, MCA96 and MCA384. Sub-classes are defined for **Arm Type** and **Tip Type**.

FluentControl comes with pre-defined liquid classes. Additionally, there are special liquid classes for specific application (e.g., piercing, drop formation or QC liquid classes that do not fully follow the convention defined above).

The pre-defined (default) liquid classes in FluentControl should always be tested and validated for the target application. Default Multi-pipetting liquid classes usually need optimization for the target application.

Opening the liquid editor for displaying and editing liquid classes

1. To open the **Liquid Class** editor, click **Liquid Classes** in the Controlbar.
2. Double-click on any of the liquid classes.
   *Alternatively, on the **File** menu, click **Open** > **Liquid Class**.*

8.6.1 Parameters for Dispense

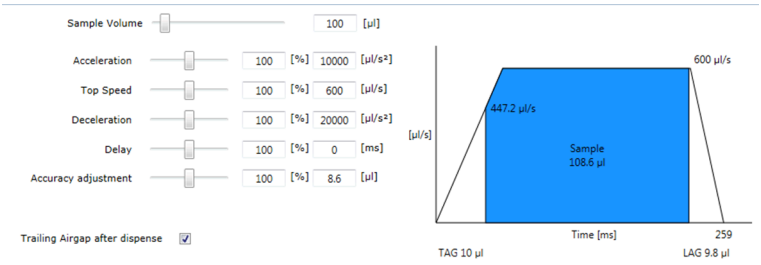


Fig. 103: Dispense: Easy Control

Sample Volume: At run time, the volume given in the pipetting command will be used to calculate all the parameters. **Sample Volume** allows you to select different volumes at edit time to see how the liquid class will behave. To specify the sample volume in microliter, type it in the box or move the slider. Move the slider to the right to increase sample volume up to max. 1000 µl. For large volumes (> 1 ml), type your sample volume in the box (again in µl). When you change the sample volume, FluentControl updates all volume dependent parameters.

Acceleration: To specify the acceleration as a percentage and/or in µl/s 2, type it in one of the boxes or move the slider.

Top Speed: To specify the top speed as a percentage and/or in µl/s, type it in one of the boxes or move the slider.

Deceleration: To specify the deceleration as a percentage and/or in µl/s 2, type it in one of the boxes or move the slider.

Delay: The percentage to multiply against the **Pipetting Speed Formula** result. To change the percentage, type in one of the boxes or move the slider. Move the slider to the right to increase Delay up to max. 200 %. For higher delay time settings (up to max. 60 seconds), type in the second box ([ms]).

Accuracy adjustment: Once precision performance is as desired you can alter the **Accuracy Adjustment** with this percentage to multiply against the formula result. Type in one of the boxes or move the slider. Move the slider to the right to increase Accuracy Adjustment up to max. 200 % or type a value between -10000 and 10000 µl in the second box.

Trailing Airgap after dispense: Check box to activate or deactivate the Trailing Airgap (TAG) after dispense.

8.6.2 Subclasses

Each liquid class is divided into subclasses depending on arm type and tip type. The appropriate subclass is automatically used at runtime.

The pipetting parameters for each pipetting command type are split amongst the **Aspirate**, **Dispense** and **Mix** sections which in turn are divided in up to four sub-sections:

- **Easy Control** (based on Tip Type): A graphical editor that allows to easily set the main parameters.
- **Detection & Positioning** (based on Tip Type): You can configure the liquid level detection, tracking options, error handling, and retract properties here.
- **Formulas** (based on Tip Type): Liquid handling parameters to be entered as functions.
- **MicroScript** (based on Arm Type): Pipetting steps can be defined as a sequence of MicroScript commands offering a high degree of flexibility.

For related information, refer to sections
“Capacitive Liquid Level Detection” [▷ 189],
“MicroScript” [▷ 184],
“Easy Control” [▷ 171],
“Detection and Positioning” [▷ 174],
“Liquid Level Detection (FCA / AirFCA)” [▷ 178],
“Errors Requiring User Interaction (FCA)” [▷ 473],
“Retract Properties (MCA384 and MCA96)” [▷ 177] and
“Formulas” [▷ 184].

8.6.3 Liquid Class Editor

The **Liquid Class** editor will let you open liquid classes to review the settings and enable editing to users that have the appropriate rights. The default liquid classes are locked and must be duplicated in order to edit them. Liquid Class microscripts are specific for each arm.

There are sections (**Aspirate**, **Dispense** and **Mix**) where the pipetting parameters are defined. The **Aspirate** section of the relevant liquid class, for example, gets executed if you use one of the **Aspirate** commands. If there is a conditioning volume it will be dispensed with the **Dispense MicroScript** and parameters of the same liquid class.

FluentControl remembers which settings were last opened and reopens liquid classes with the previous selection. To alert users from unintentionally editing settings from arms or tips not installed on the connected system, those selections appear in the drop down menu with a safety alert symbol.

- Both **Aspirate** and **Dispense** are structured into **Easy Control**, **Detection & Positioning**, **Formulas** and a **MicroScript**.

– The Easy Control, Detection and Positioning and Formula settings are specific to arm and tip type—i.e., a default liquid class may have dozens of independent parameter tabs.

- **Mix**, **Wash**, **Flush** and **Empty Tip** sections only have a MicroScript section.
- Liquid class name with **Rename** and **Comment**.
- **Variables**
- **Save** and **Save & Close** buttons.

For related information, refer to sections
“Liquid Class Editor” [▷ 168],
“MicroScript” [▷ 184],
“Easy Control” [▷ 171],
“Retract Properties (MCA384 and MCA96)” [▷ 177] and
“Formulas” [▷ 184].

8.6.4 Easy Control

Subsection for quickly altering the parameters that most impact accuracy and precision in Aspirate and Dispense. These adjustments impact the full volume range.

Easy Control: The Easy Control visualizes the pipetting parameters set in the Formulas subsection and applies a factor to the result of the expression from the Formulas subsection. The result of this is checked against the minimum and maximum in the Formulas subsection. If it is above or below the range, the minimum or maximum value is used. This value is then used as the initial value of the corresponding variable in the **MicroScript**.

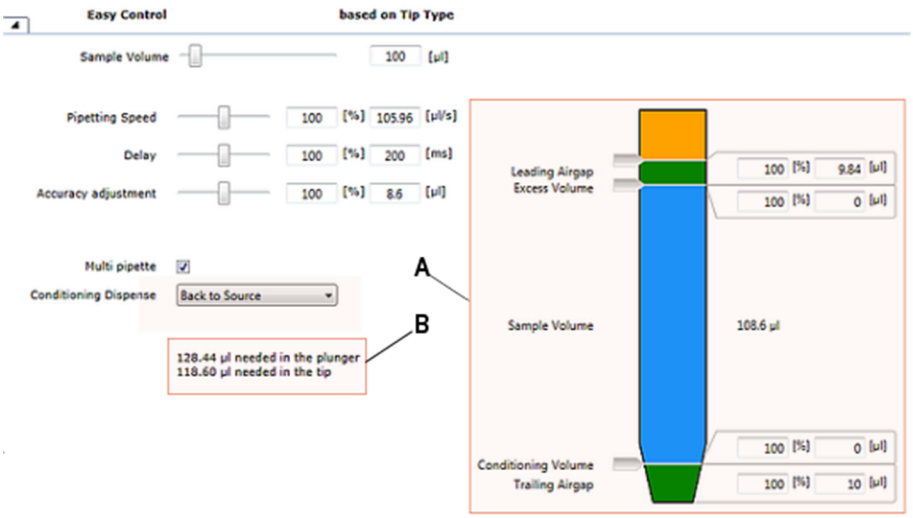


Fig. 104: Easy control

A visualization of the tip displaying the calculated volumes. Not to scale.

B Pipetting summary providing you with volume information how many micro-liters are needed in the plunger and in the tip. The Leading Airgap does not count against the available volume of the tip. This can be helpful when troubleshooting Context Check warnings about needed volumes.

Aspirate: Subsection Easy Control (based on Tip Type)

8.6.4.1 Parameters for Aspirate

Sample Volume: At run time, the volume given in the pipetting command will be used to calculate all the parameters. **Sample Volume** allows you to select different volumes at edit time to see how the liquid class will behave. To specify the sample volume in microliter, type it in the box or move the slider. Move the slider to the right to increase sample volume up to max. 1000 µl. For large volumes (> 1 ml), type your sample volume in the box (again in µl). When you change the sample volume, FluentControl updates all volume dependent parameters.

Pipetting Speed: The percentage to multiply against the **Pipetting Speed Formula** result. To change the percentage, type in one of the boxes or move the slider. Move the slider to the right to increase Pipetting Speed up to max. 200 %. For higher speed (up to max. 800 µl/s), type the desired pipetting speed for the given volume in the second box ([µl/s]).

Delay: The percentage to multiply against the Pipetting Speed Formula result. To change the percentage, type in one of the boxes or move the slider. Move the slider to the right to increase Delay up to max. 200 %. For higher delay time settings (up to max. 60 seconds), type in the second box ([ms]).

Accuracy adjustment: Once precision performance is as desired you can alter the Accuracy Adjustment with this percentage to multiply against the formula result. Type in one of the boxes or move the slider. Move the slider to the right to increase Accuracy adjustment up to max. 200 % or type a value between -10000 and 10000 µl in the second box.

Multi pipette: Select this check box to change the method by which a liquid can be distributed to Multi Pipetting, which should be used when aspirating once and then dispensing several aliquots. If Multi pipette is selected, then the following options become visible:

An Excess Volume helps ensure the final aliquot has similar conditions to the initial aliquots and can help reduce influence of previously pipetted liquids. Alter the Excess Volume factor by typing in one of the boxes ([%] or [µl]). The recommended excess volume is ≥ 30 µl. Ideally it amounts to 15% of the total volume. The excess volume can be gotten rid of by using an Empty Tips command, or by changing or washing tips.

A Conditioning Volume helps the first aliquot have similar conditions to all subsequent aliquots. The conditioning volume is discarded automatically as part of the aspiration step. Change the conditioning volume factor by typing in one of the boxes ([%] or [µl]). The recommended conditioning volume is ≥ 30 µl, or is ideally of the same volume as one aliquot. The conditioning volume is dispensed back into the original container (Back to Source) or into the wash station (Back to Waste).

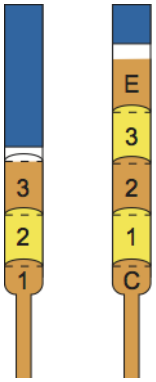


Fig. 105: Conditioning and excess volumes

1, 2, 3 Aliquots

E Excess volume

Recommendations for multi pipetting: - Good precision (CV) requires at least 4 replicates and a recommended maximum of 12 replicates

- After 12 dispenses, the STAG should be replaced (Liquid FCA only)
- When multi pipetting aliquots of various volumes, dispense the smallest volumes before the larger volumes. The largest volume should be dispensed last.
- The default multi pipetting liquid classes in FluentControl usually require optimization for the target application

The conditioning volume aliquot will be dispensed as a free dispense step using the parameters from the Easy Control and Formulas of the dispense section. The dispense **MicroScript** will not be used.

Finally, you can specify two additional parameters in the schematic visualization of the tip:

- The air of the Leading Airgap is dispensed together with the sample volume.
- The resulting tip blowout leads to more accurate pipetting. Type your Leading Airgap volume in one of the boxes ([%] or [µl]).

The Trailing Airgap is used to prevent liquid dripping from the tips when the arm moves. In addition, it ensures that the liquid meniscus is not dragged away from the end of the tip during fast arm movements. Type your Trailing Airgap volume in one of the boxes ([%] or [µl]).

8.6.4.2 Parameters for Dispense

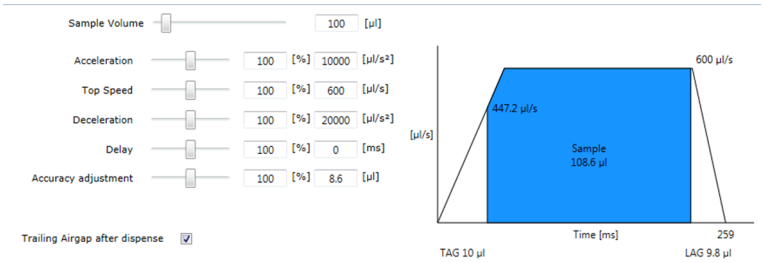


Fig. 106: Dispense: Easy Control

Sample Volume: At run time, the volume given in the pipetting command will be used to calculate all the parameters. **Sample Volume** allows you to select different volumes at edit time to see how the liquid class will behave. To specify the sample volume in microliter, type it in the box or move the slider. Move the slider to the right to increase sample volume up to max. 1000 µl. For large volumes (> 1 ml), type your sample volume in the box (again in µl). When you change the sample volume, FluentControl updates all volume dependent parameters.

Acceleration: To specify the acceleration as a percentage and/or in µl/s 2, type it in one of the boxes or move the slider.

Top Speed: To specify the top speed as a percentage and/or in µl/s, type it in one of the boxes or move the slider.

Deceleration: To specify the deceleration as a percentage and/or in µl/s 2, type it in one of the boxes or move the slider.

Delay: The percentage to multiply against the **Pipetting Speed Formula** result. To change the percentage, type in one of the boxes or move the slider. Move the slider to the right to increase Delay up to max. 200 %. For higher delay time settings (up to max. 60 seconds), type in the second box ([ms]).

Accuracy adjustment: Once precision performance is as desired you can alter the **Accuracy Adjustment** with this percentage to multiply against the formula result. Type in one of the boxes or move the slider. Move the slider to the right to increase Accuracy Adjustment up to max. 200 % or type a value between -10000 and 10000 µl in the second box.

Trailing Airgap after dispense: Check box to activate or deactivate the Trailing Airgap (TAG) after dispense.

8.6.5 Detection and Positioning

Parameters such as liquid detection mode, pipetting position and error handling are configurable within each liquid class in this subsection.

8.6.5.1 FCA/Air FCA Positioning

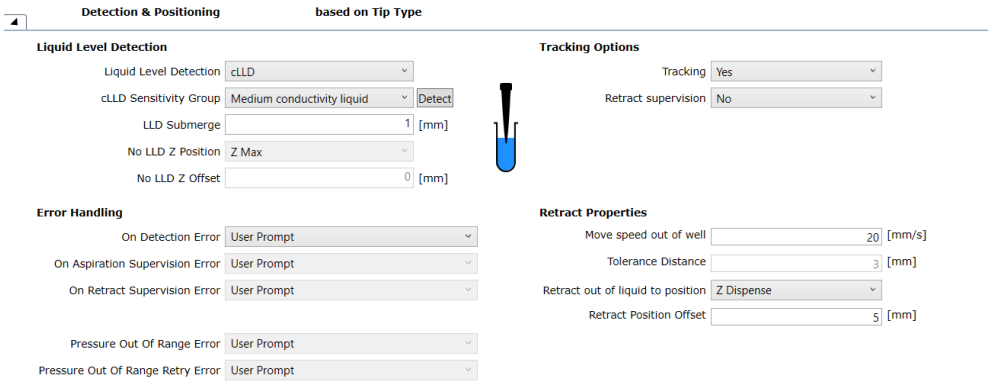


Fig. 107: Detection and Positioning

- Detection & Positioning (based on Tip Type)
- Liquid Level Detection
- Tracking Options (FCA/Air FCA)
- Error Handling (FCA/Air FCA)
- Retract Properties (FCA/Air FCA)

8.6.5.2 MCA384 Positioning

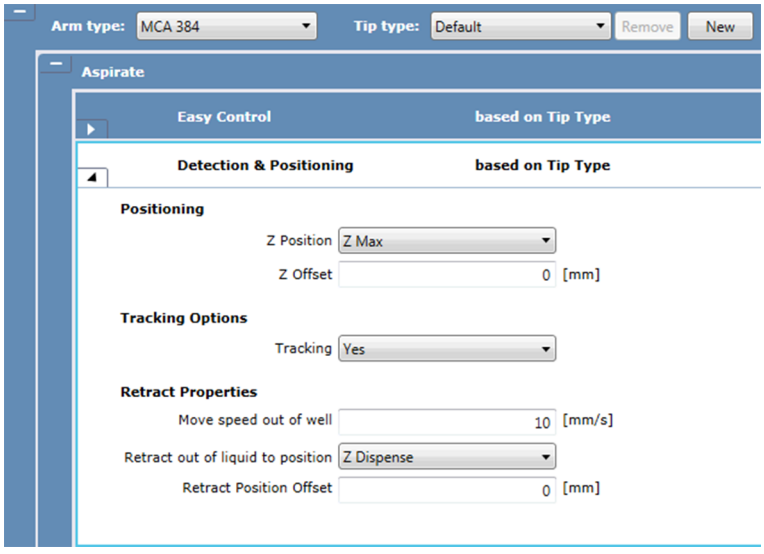


Fig. 108: Positioning for the MCA 384 arm type

Z-Position: Click the **Z Position** arrow to see all options: **Known position**, **Z Max** , or **Z Dispense**.

Z-Offset: You can type an offset in mm from the **Z Position** after movement (positive = higher, negative = lower).

Retract out of liquid to position: Click the **Retract out of liquid to position** arrow to see all options. You can set the distance to retract out of the liquid and can select a retract position form the list.

Retract Position Offset: You can type an offset in mm from the retract position after movement (positive = higher, negative = lower).

For further information, refer to section “Liquid Level Detection (FCA / AirFCA)” [▷ 178].

8.6.5.3 MCA96 Positioning

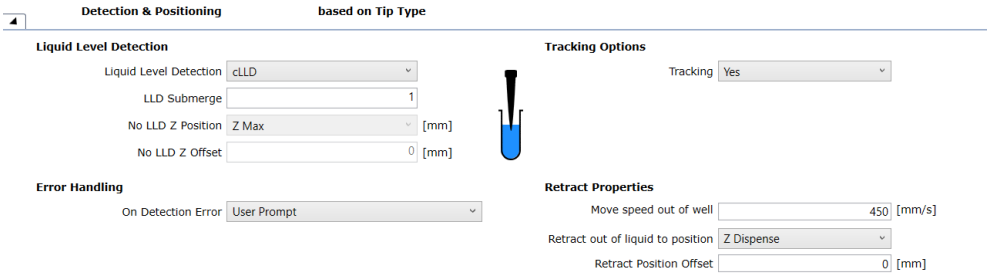


Fig. 109: MCA96 Positioning

Liquid Level Detection : Use cLLD or no liquid level detection. You can specify the submerge depth for tracking and the positioning in Z.

Liquid Level Detection: None or cLLD.

LLD Submerge [mm]: The tips stop at the specified submerge depth. During aspiration of liquid the tip moves down to keep the submerge depth constant (this is called “tracking”, see Tracking Options (MCA384 and MCA96).

No LLD Z Position: In the Liquid Level Detection list, click “None”, then click the Z Position arrow to see the options (Z Max or Known position).

No LLD Z Offset: In the Liquid Level Detection list, click None, then type the z offset in mm.

8.6.5.4 Tracking Options FCA/Air FCA

cLLD monitoring can be kept active while aspirating. Aspiration is supervised and an error is thrown if the monitored capacitance changes. Supervised capacitive detection can also be activated while retracting the tip after aspirate; the system then sends a signal when it is outside the liquid (recorded capacitance change).

Supervised: Capacitive liquid level detection (cLLD) is kept on while aspirating. The system will report an error if the monitored electrical capacitance changes during aspiration. Supervised aspiration is not supported with pLLD

Tracking: If you choose an aspirate position with tracking, the tips will move downwards as the liquid level drops in the wells during aspiration. You can select **Yes**, supervised or **No** to define the tracking method.

Retract supervision: While the tip is retracting from the liquid it ensures that an exit signal occurs at the expected height. You can select **Yes** or **No** from the list. Tip retract supervision is a special function of the capacitive liquid level detection (cLLD) which generates a message if the difference between the liquid levels measured before and after sample **aspiration**/**dispense** does not correspond to the calculated difference. This indicates that the tip may be occluded or that an

object, such as a filament, is hanging from the tip. Retract supervision is not supported with pLLD. Select **Yes** to use this function. To deactivate this function, select **No**.

8.6.5.5 Tracking Options (MCA384 and MCA96)

Click the **Tracking** arrow to see all options. Select **Yes** to use tracking. Select **No** to deactivate tracking.

8.6.5.6 Retract Properties (FCA / Air FCA)

Retract is pulling back up a tip after aspiration or dispense.

Move speed out of well defines the retract speed in mm/s out of the well until "Retract out of liquid to position" position.

Tolerance Distance

Retract out of liquid to position defines the position where the tip shall be retracted to with the speed defined above.

Retract Position Offset defines an additional offset on top of the retract position defined above. Positive values are higher up, negative values lower.

For related information, refer to section “Editors” [▷ 132].

8.6.5.7 Retract Properties (MCA384 and MCA96)

Move speed out of well: You can type the speed in mm/s to retract out of the liquid until retract position is reached.

Retract out of liquid to position: Click the **Retract out of liquid to position** arrow to see all options. You can set the distance to retract out of the liquid and can select a retract position form the list.

Retract Position Offset: You can type an offset in mm from the retract position after movement (positive = higher, negative = lower).

8.6.5.8 Liquid Level Detection (MCA96)

The cLLD functionality of the MCA96 resembles the functionality used by the FCA / Air FCA. The key difference of the MCA96, compared to the FCA / Air FCA, is that only one cLLD signal is registered. This signal is spread across all tips which are mounted. The same physical aspects and constraints of cLLD still exist for the MCA96. For more information regarding cLLD, please refer to the “Capacitive Liquid Level Detection (FCA / Air FCA)” section.

8.6.5.9 Error Handling (MCA96)

These settings define the automatic error handling options for detection errors during pipetting actions.

- On Detection Error: Distinct error handling option are available for aspirate or dispense actions. The following error handling options are available for the aspirate section: Aspirate available volume, aspirate anyway, User Prompt. The following error handling options are available for the dispense section: Ignore and Continue, User Prompt.

8.6.5.10 Liquid Level Detection (FCA / AirFCA)

Use **cLLD**, **pLLD** or **no liquid level detection**. Specify the sensitivity group parameters for cLLD or let the system calculate the sensitivity. You can specify the submerge depth for tracking and the positioning in Z.

When using pierce micro- commands submerge depth and tracking does not apply and any settings here will be ignored.

Liquid Level Detection: None, pLLD or cLLD.

8.6.5.10.1 cLLD Sensitivity Group for non MultiSense Arm

There are different sensitivity groups for cLLD. The liquid to be handled must be classified, accordingly. Correct sensitivity group assignment of the pipetted liquid is critical for the reliability and the sensitivity of the capacitive liquid level detection.

The light signals from the power and instrument status lamps indicate the following states:

Tab. 3: cLLD sensitivity group definition

Liquids/ Groups	Group 1: Low	Group 2: Medium	Group 3: High	Group 7: High High, piercing
Conductivity and relative static permittivity characteristics	< 10 µS/cm and relative static permittivity ≥ 24 and < 80	10 to 100 µS/cm or < 10 µS/cm and relative static permittivity > 80	> 100 µS/cm	> 100 µS/cm in combination with tube piercing
Example liquids that represent the group	Ethanol 96%	Deionized water	Tap water, physiological salt solutions (0.9% NaCl)	physiological liquids (e.g., blood)

Select **Low conductivity liquid**, **Medium conductivity liquid**, or **High conductivity liquid** from the **cLLD Sensitivity Group** list. If the conductivity and the relative static permittivity is not known for a liquid, an automatic method for assignment of the liquid to one of the three groups is available. Click **Detect to calculate the sensitivity** and see **Sensitivity Calculation (FCA)** – Direct Command.

cLLD sensitivity group 7 cannot be selected from the menu and has to be defined in the microscript command.

Non-conductive liquids with a relative static permittivity lower than 24 (e.g., octanol, ε r = 10.3) may be detected with cLLD under certain conditions.

If your liquid falls into this category, please contact your Tecan application specialist to determine how cLLD can be used for your specific labware configuration and application needs.

8.6.5.10.2 cLLD Sensitivity Group for MultiSense Arm

Select MultiSense as cLLD Sensitivity Group only when a MultiSense arm is used. For a MultiSense arm, if another sensitivity group (i.e. low/medium/high conductivity liquid) is selected, a warning will be visible in the liquid class editor. At runtime MultiSense will be used as the sensitivity group.

The MultiSense arm does not support the detection of the sensitivity group by clicking on 'Detect' button.

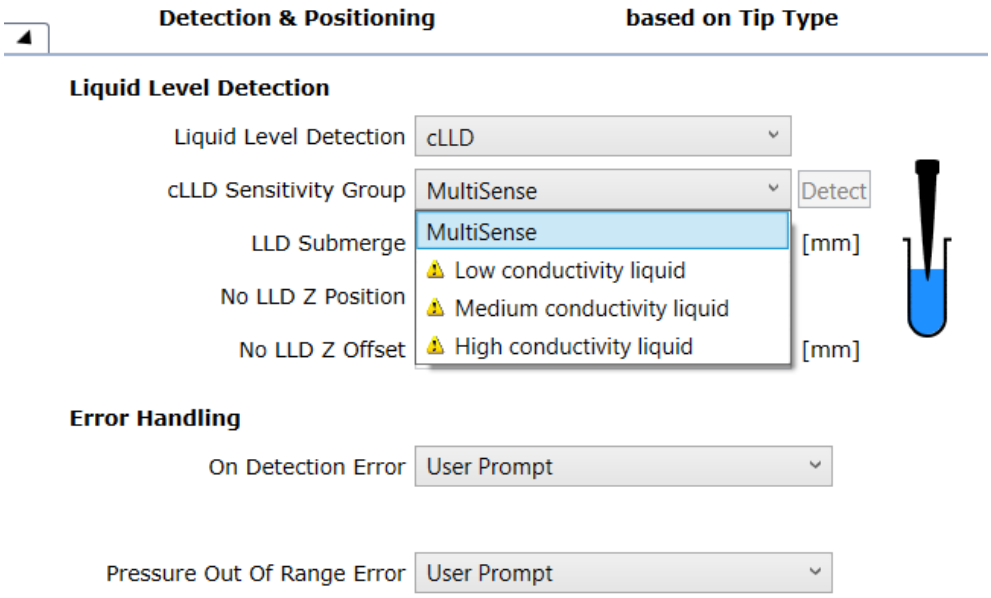


Fig. 110: cLLD Sensitivity Group settings

8.6.5.10.3 Liquid level detection parameters

This information is valid for both MultiSense and non-MultiSense arms, respectively.

LLD Submerge [mm]: The tip stops at the specified submerge depth. During aspiration of liquid the tip moves down to keep the submerge depth constant (this is called “tracking”, see Tracking Options (FCA / Air FCA)).

no LLD Z Position: In the Liquid Level Detection list, click None, then click the Z Position arrow to see more options (Z Max and Known position).

no LLD Z Offset: In the Liquid Level Detection list, click None, then type the z offset in mm.

Capacitive Liquid Level Detection

The capacitive liquid level sensor of the pipetting channel detects the liquid level in a container by measuring the capacitance change between the conductive tip and the grounded deck of the instrument while the tip moves downwards from the reference environment (air) to the liquid.

When using capacitive liquid level detection, pay attention to the following:

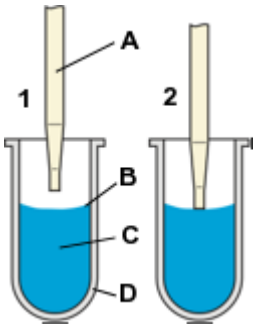
- cLLD only works with conductive tips (and DiTis) and conductive liquids.
- In 1536-well microplates, liquid detection is not possible.
- Labware itself must be non-conductive.

Each tip can individually detect the surface of a conductive liquid by measurement of changes in capacitance. Each channel has an individual liquid detection. Generally, detection of conductive liquids of following volumes is possible:

- 50 µl low-conductive liquid in microplates with round bottoms, for DiTis.
- 100 µl conductive liquid in sample tubes with a diameter of 10 or 13 mm.
- 150 µl conductive liquid in sample tubes with a diameter of 16 mm.
- 5 ml conductive liquid in reagent trough.

Detection Signal

A threshold value for the change in measured capacitance (in the following referred to as “signal”) is defined in order to trigger positive feedback when the tip comes in contact with the liquid surface.

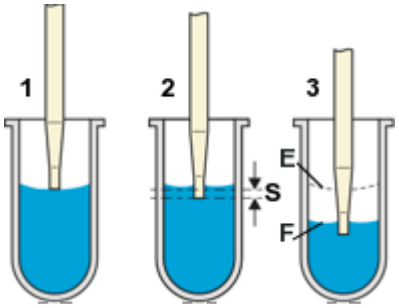


1	Tip moves down towards the liquid.	**2**	Tip reaches liquid surface, triggers the detection signal.
A	Tip	**B**	Liquid level
C	(Sample) liquid	**D**	Container

Liquid Level detection

Suitable Pipetting Tips: cLLD works with conductive disposable tips and fixed tips.

Tracking During Aspiration: At the time the liquid level detection signal is generated the tip does not stop but moves on to the predefined submerge depth to start aspiration of liquid. The example shows a typical workflow:



1	Tip detects and reports the liquid level and keeps moving down without stopping.	**2**	Tip stops at the specified submerge depth.
3	During aspiration: tip moves down to keep the submerge depth constant (“tracking”).	**E**	Original liquid level
F	Liquid level after aspiration	**S**	Submerge depth

8.6.5.10.4 Error Handling (FCA / Air FCA)

These settings define the automatic error handling options for detection errors during pipetting actions. The selection will always apply to all channels.

In case option user prompt is chosen and during runtime no or not enough liquid is detected a user prompt with the following options will be shown: Aspirate available volume, aspirate anyway, deactivate tip, refill labware or retry. The option refill labware will do a retry after the user has confirmed the refill of the labware.

- **Special case for pLLD:** In case no or not enough liquid is detected a automatic retry with a new tip will be performed. When then again no or not enough liquid is detected a user prompt will be shown with the options: Aspirate available volume, aspirate anyway, deactivate tip, refill labware or retry with new tip. The option refill labware will perform a automatic retry with new tip after the user has confirmed the refill of the labware.
- **On Detection Error:** Distinct error handling options are available for aspirate or dispense actions.
  The following error handling options are available for aspirate section: Aspirate available volume, aspirate anyway, user prompt or deactivate tip.
  The following error handling options are available for dispense section: Ignore and Continue, Deactivate tip, User prompt.
- **On Aspiration Supervision Error:** To make this error handling option available, select **Yes, supervised** from the Tracking list under Tracking Options.
  The following error handling options are available: Ignore and continue, dispense back and deactivate tip, dispense back and retry, dispense to waste and deactivate tip, user prompt.
- **On Retract Supervision Error:** Safe Paths for Blocked Tip Error. To make this error handling option available, select Yes from the Retract supervision list under Tracking Options.
  The following error handling options are available: Ignore and continue, dispense back and deactivate tip, dispense back and retry, dispense to waste and deactivate tip, user prompt.

- **Pressure Out Of Range Error:** You can define automatic error handling options for pressure out of range errors on aspirate and dispense within the liquid class. Your selection will always apply to all channels. Only one option can be selected for Pressure out of range during aspirate and one option for Pressure out of range during dispense.
  The following error handling options are available for aspirate: Continue, pipette nothing Continue, dispense back, pipette nothing Continue, drop tip, pipette nothing the tip including its contents is discarded into the waste Retry once with new tip - the tip including its contents is discarded into the waste before a new tip is mounted User prompt.
  The following error handling options are available for dispense: Safe discard tip the tip including its contents is discarded into the waste User prompt.
- **Pressure out of Range Retry Error:**

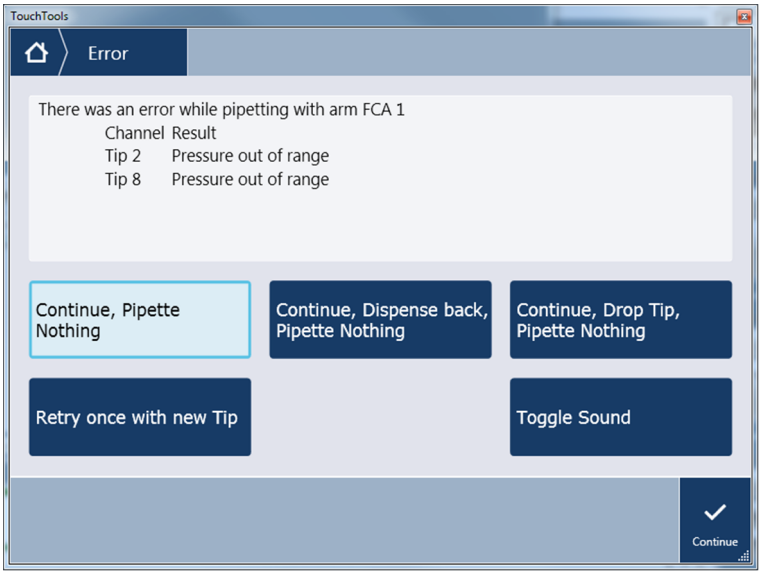


Fig. 111: Pressure out of range errors during aspirate

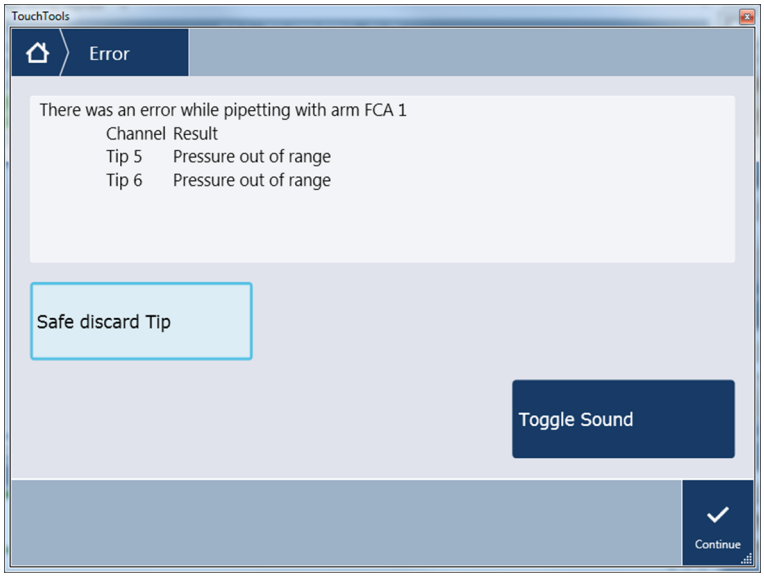


Fig. 112: Pressure out of range during dispense

Configuring Pressure Limits

To configure the upper and lower pressure limits the user can set the following variables:

Tab. 4: Configuring Pressure Limits

Name	Range	Default	Unit
lowerPressure-Limit	-40 ... 200	-40	mBar
upperPressure-Limit	-40 ... 200	200	mBar

When the pressure exceeds the upper or lower limit a “Pressure out of Range” error is triggered.

See also chapter “**Pressure Out Of Range Error**”.

If the user wants to switch off the “Pressure Out Of Range” feature for a LiquidClass, the following variable can be added:

Tab. 5: Switching Pressure Monitoring On/Off

Name	Range	Default
pressureMonitoringOnOff	0(off) ; 1(on)	1

If the variable does not exist, the default value is used.

If “pressureMonitoringOnOff” is set to 0 the liquid class editor warns the user.

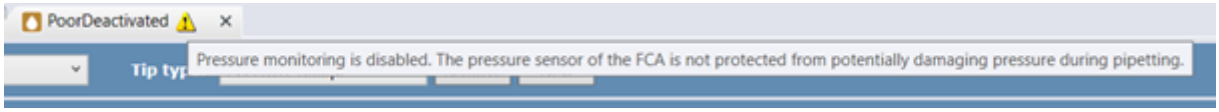


Fig. 113: Warning that pressure monitoring is disabled in this liquid class

8.6.6 Formulas

You can edit the formulas of the main liquid handling parameters for aspiration or dispense, and you can add new rows. The Formula boxes can be filled with specific values—e.g.:
Leading Airgap: a fixed value or a formula in general a pipetting volume dependent expression. This value is then fed into the **Easy Control** subsection and can be further modified by a factor before being fed into the **MicroScript**.

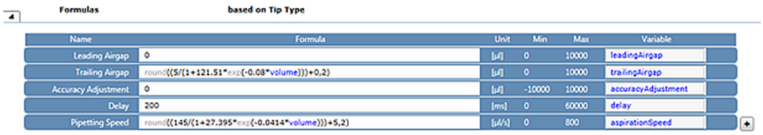


Fig. 114: Formulas (based on Tip Type)

Here you find the default formulas for the transformation of the parameters in **Easy Control** (based on Tip Type) to **MicroScripts** parameters. The figure above shows the **Aspirate** related formulas for **Leading Airgap**, **Trailing Airgap**, **Accuracy Adjustment**, **Delay**, and **Pipetting Speed**.

The volume variable is passed from your script to the Formulas subsection, then fed into the Easy Control subsection and can be further modified by a factor before being fed into the MicroScript.

Add and remove Formulas table

1. Click + to add a row to the Formulas table.
2. Click ⊗ to remove a new row form the table.

You cannot delete a default row but overwrite its Formula box and Variable box.

8.6.7 MicroScript

The base of a FluentControl liquid class is the underlying **MicroScript** and its **MicroScript** commands. The **MicroScript** command defines the sequence of actions during aspiration, dispense and mix. Pipetting parameters (variables) set in the three subsections **Easy Control**, **Positioning & Detection** and **Formulas** can be used in **MicroScript** command. There is a single section **MicroScript** for each arm type within a liquid class but settings for **Easy Control**, **Positioning & Detection and Formulas** are tip-type specific.

The system provides a single section **MicroScript** for each arm type within a liquid class. The section is displayed in the section **Aspirate** and **Dispense** in order to edit **MicroScript** commands for these two sections.

Piercing tips have beveled tips. As a result, the needle lumen is not completely submerged into the liquid when the tip touches the surface.

- The submerge depth has to be at least 4 mm. For piercing micro commands, the submerge depth is by default 4.5 mm (as defined as a custom attribute "TipMinimumSubmergeDepth" in the Piercing tip labware). The parameter "Submerge depth" in the Detection & Positioning editor is ignored.

8.6.7.1 Aspirate Air

Air gap volume [µl]: The volume for the air gap in micro liters.

Aspiration Speed [µl/s]: The maximum speed in micro liter per second used for the aspiration. Default: airspeed.

Backlash volume [µl]: You can define a backlash compensation to be performed when aspirating air and liquid, to compensate for surface tension especially when pipetting small volumes. During runtime the plunger aspirates the defined backlash volume (default = 0µl, range 0 to 10µl) additionally to the defined aspirate volume and subsequently dispense the backlash volume.

8.6.7.2 Aspirate Liquid

The system tracks the aspiration if the user defines any other value than zero for the tracking parameter.

Liquid Volume: The volume to aspirate in µl.

Aspiration Acceleration: The acceleration used for the aspiration in µl/s^2.

Aspiration Speed [µl/s]: The maximum speed in micro liter per second used for the aspiration. Default: aspirationSpeed.

Aspiration Deceleration: The deceleration used for the aspiration in µl/s^2.

Use tracking: Tracking moves the tip down in Z-direction during the aspiration, to keep the distance between tip and liquid level (the submerge depth) constant.

Backlash volume [µl]: You can define a backlash compensation to be performed when aspirating air and liquid, to compensate for surface tension especially when pipetting small volumes. During runtime the plunger aspirates the defined backlash volume (default = 0µl, range 0 to 10µl) additionally to the defined aspirate volume and subsequently dispense the backlash volume.

8.6.7.3 Aspirate Liquid with XYTracking

The microscript command Aspirate Liquid with XYTracking performs synchronous movements of the pipetting channels in the x/y direction during liquid aspiration. Additionally, it allows liquid tracking (in z direction), which can be enabled/disabled by setting the liquid tracking parameter to 1 or 0, respectively.

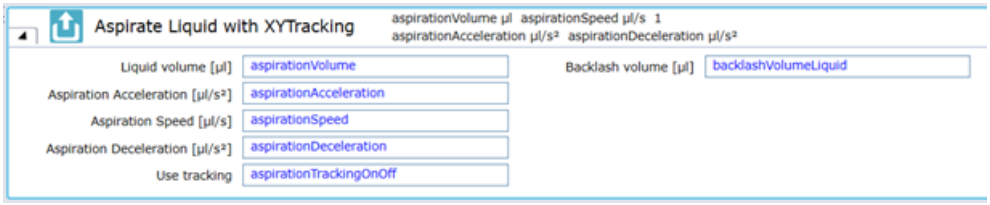


Fig. 115: Aspirate Liquid with XYTracking command

The sequence of x/y coordinates for x/y tracking must be defined as a custom attribute XYTrackingSequence of the target labware (see Fig. above). A guide on how to use the command and create a x/y tracking sequence can be found in the Fluent® Reference Manual (Doc ID 399937).

To protect the FCA hardware, the channels have a preset maximum move speed in x and y. If many x/y coordinates or a high aspiration speed is defined, the system may not be able to synchronize the aspiration and channel x/y movements.

*Pathfinder is not active during the **XYTracking** moves. Make sure that the coordinates specified do not cause the arm or channels to collide with objects on the worktable.*

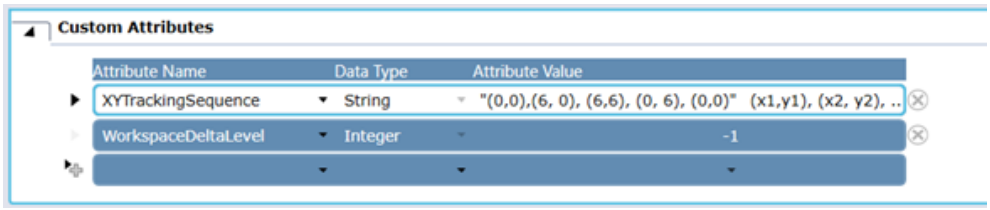


Fig. 116: XYTrackingSequence must be defined in the “Custom Attrributes” section of the labware to be aspirated from.

- **Liquid Volume [µl]**: The volume to aspirate in µl/s^2.
- **Aspiration Acceleration [µl/s^2]**: The acceleration used for the aspiration in µl/s^2.
- **Aspiration Speed [µl/s]**: The maximum speed in micro liter per second used for the aspiration. Default: aspirationSpeed.
- **Aspiration Deceleration [µl/s^2]**: The deceleration used for the aspiration in µl/s^2.
- **Use tracking**: Tracking moves the tip down in Z-direction during the aspiration, to keep the distance between tip and liquid level (the submerge depth) constant.
- **Backlash volume [µl]:** You can define a backlash compensation to be performed when aspirating air and liquid, to compensate for surface tension especially when pipetting small volumes. During runtime the plunger aspirates the defined backlash volume (default = 0µl, range 0 to 10µl) additionally to the defined aspirate volume and subsequently dispense the backlash volume.

Restrictions:

- For the given microscript command Aspirate Liquid with XYTracking it is assumed that the same aspiration volume is used on all active channels.

- The coordinates of the x/y path defined in the XYTrackingSequence custom attribute could be outside the labware geometry. Thus, the user is required to test and optimize all coordinates prior to use (see Reference Manual for further Information).

8.6.7.4 Dispense Liquid with XYTracking

The microscript command Dispense Liquid with **XYTracking** performs synchronous movements of the pipetting channels in the x/y direction during liquid dispense. Additionally, it allows liquid tracking (in z direction), which can be enabled/disabled by setting the liquid tracking parameter to 1 or 0, respectively.

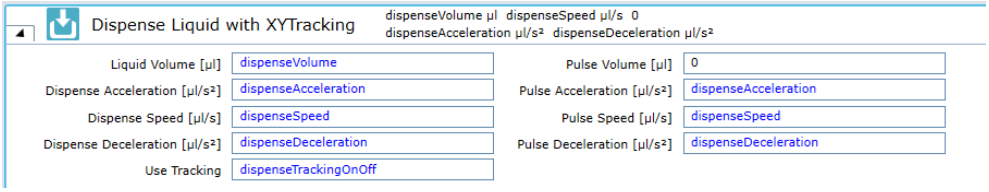


Fig. 117: Dispense Liquid with X / YTracking command

*The sequence of x/y coordinates for x / y tracking must be defined as a custom attribute**XYTrackingSequence** of the target labware (see Fig. above). A guide on how to use the command and create a x / y tracking sequence can be found in the Fluent® Reference Manual (Doc ID 399937).*

To protect the FCA hardware, the channels have a preset maximum move speed in x and y. If many x / y coordinates or a high dispense speed is defined, the system may not be able to synchronize the dispense and channel x/y movements.

*Pathfinder is not active during the **XYTracking** moves. Make sure that the coordinates specified do not cause the arm or channels to collide with objects on the worktable.*

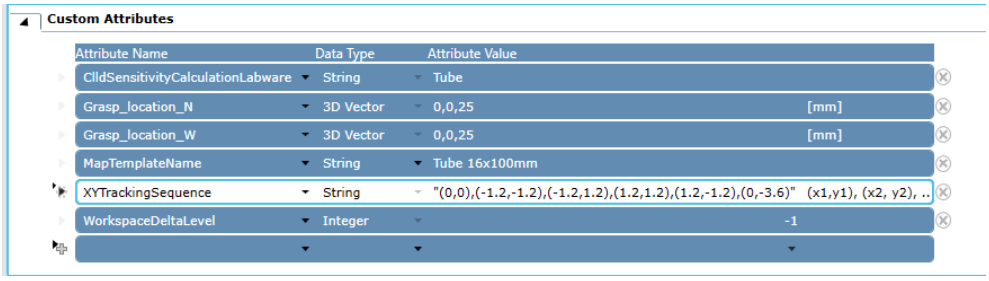


Fig. 118: XYTrackingSequence

The **XYTrackingSequence** must be defined in the **Custom Attributes** section of the labware to be dispensed to.

Parameters:

Liquid Volume [µl]: The volume to dispense in µl.

Dispense Acceleration [µl/s2]: The acceleration used for the dispense in µl/s2.

Dispense Speed [µl/s]: The max. speed used for the dispense in µl/s.

Dispense Deceleration [µl/s2]: The deceleration used for the dispense in µl/s2.

Use tracking: Tracking moves the tip down in Z-direction during the dispense, to keep the distance between tip and liquid level (the submerge depth) constant.

Restrictions:

- For the given microscript command Dispense Liquid with **XYTracking** it is assumed that the same dispense volume is used on all active channels.
- The coordinates of the x/y path defined in the **XYTrackingSequence** custom attribute could be outside the labware geometry. Thus, the user is required to test and optimize all coordinates prior to use (see Fluent Reference Manual for further Information).

8.6.7.5 Aspirate Supervised

Use this command to monitor the cLLD signal tip during aspiration. This allows you to receive an error message in case that the tip moves out of the liquid and starts aspirating air. This could happen when the physical dimension of the labware does not match the defined compartment shape which leads to incorrect calculation of the tracking motion.

Liquid Volume: The volume to aspirate in µl.

Aspiration Acceleration: The acceleration used for the aspiration in µl/s 2.

Aspiration Speed [µl/s]: The maximum speed in microliter per second used for the aspiration. Default: airSpeed.

Aspiration Deceleration: The deceleration used for the aspiration in µl/s^2.

Backlash volume [µl]: You can define a backlash compensation to be performed when aspirating air and liquid, to compensate for surface tension especially when pipetting small volumes. During runtime the plunger aspirates the defined backlash volume (default = 0µl, range 0 to 10µl) additionally to the defined aspirate volume and subsequently dispense the backlash volume.

8.6.7.6 Pressure based liquid level detection

Performs pressure based liquid level detection (pLLD) with one tip per compartment.

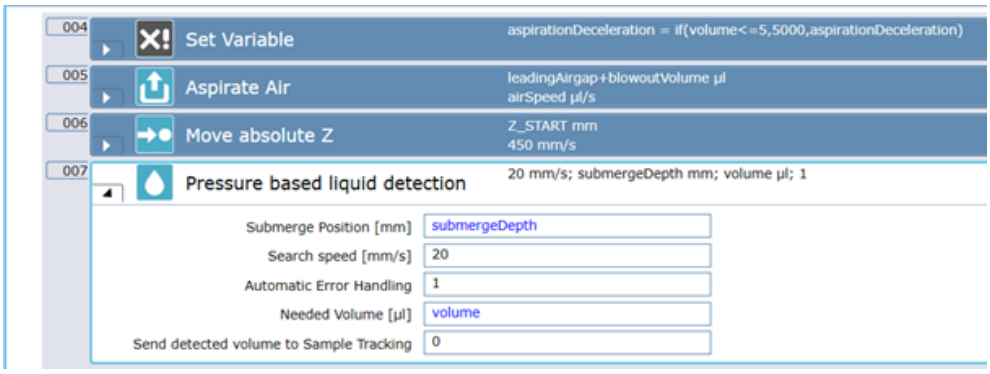


Fig. 119: Pressure based liquid level detection

Parameter

Robotic Action:

- All tips selected in the pipetting script level command start a detection action at the same time.
- The tips travel from the current position toward Z-Max.
- When the pLLD stop event is received per tip, that tip continues from its current position to the submerge depth but not deeper than Z-Max.

Submerge Position [mm]:

Set the submerge depth in mm within the detected liquid (target position of tip after successful detection).

Search Speed [mm/s]:

Set the speed used for the Z movement of the tip in mm/s.

Automatic Error Handling:

Defines whether the error handling defined in the Detection and Positioning section of the Liquid Class applies (=1) or not (=0).

Needed Volume:

Minimal required volume to be detected for successful detection. Send detected volume to Sample Tracking: Optionally the detected volume can be sent to Sample Tracking (=1)

Send detected volume to Sample Tracking:

Optionally the detected volume can be sent to Sample Tracking (=1)

8.6.7.7 Capacitive Liquid Level Detection

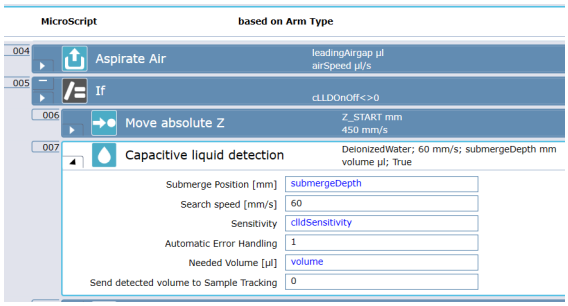


Fig. 120: Capacitive liquid detection

Performs liquid level detection by cLLD with one tip per compartment.

Robotic Action:

- All tips selected in the pipetting script level command start a detection action at the same time.
- The tips travel from the current position toward Z-Max.
- When the cLLD stop event is received per tip, that tip continues from its current position to the submerge depth but not deeper than Z-Max.

Submerge Position [mm]: Set the submerge depth in mm within the detected liquid (target position of tip after successful detection).

Sensitivity: Select the sensitivity group of liquid that is used for the cLLD detection.

Search Speed [mm/s]: Set the speed used for the Z movement of the tip in mm/s.

Automatic Error Handling: Defines whether the error handling defined in the Detection and Positioning section of the Liquid Class applies (=1) or not (=0).

Needed Volume: Minimal required volume to be detected for successful detection.

Send detected volume to Sample Tracking: Optionally the detected volume can be sent to Sample Tracking (=1)

8.6.7.8 Deactivate Tip

This command deactivates further actions with the given tip until the next **Drop Tips**, **Set Tips Back** or **Wash** command.



Fig. 121: Dispense Liquid (MicroScript Command)

8.6.7.9 Detect Phase

Detects the interface between two liquids with different viscosities. Constant pressure monitoring during aspiration enables the detection and trigger upon immediate pressure change within the disposable tip. Both positive and negative pressure changes can be registered by the system, allowing the more viscous liquid to be either the upper or the lower liquid fraction.

To integrate the Phase Separator into a workflow, the Detect Phase command needs to be inserted into a (dedicated) liquid class. The liquid class can be used in an FCA Aspirate command to identify the height of the liquid-liquid interface. The detection height is stored in the well attribute PhaseHeight and can be retrieved using the GetAttribute() function.

Additionally, the value indicating where the last phase detection stopped, regardless of the detection result, is stored in the well attribute LastStopPositionOfPhaseDetection. This value can be retrieved using the GetAttribute() function.

Two template liquid classes for the Transfer Phase command are available in FluentControl: Phase Separator Plasma Contact Wet Single and Phase Separator Plasma Free Multi.
These liquid classes should not be used with liquid handling commands other than Transfer Phase.

The default values given in the detection command are optimized for centrifuged whole blood samples (centrifugation conditions 2500 rcf at room temperature for 10 minutes). For other liquids, the parameters in the microscript command might need to be adjusted.

For more information, consult the Fluent Reference Manual (Doc ID 399937).

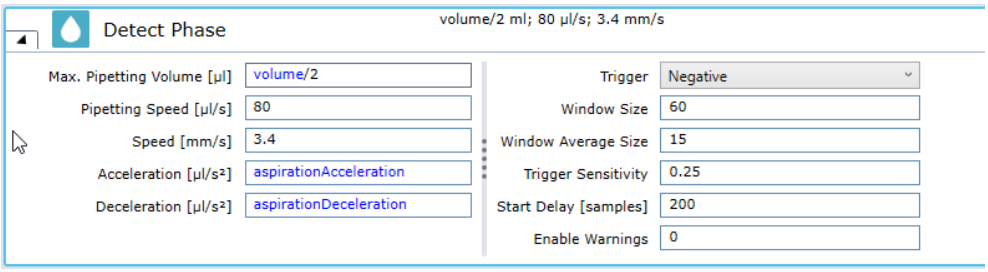


Fig. 122: Detect Phase

Parameters

Max Pipetting Volume [µL]: Define the maximum volume that shall be aspirated while searching the liquid-liquid interface before stopping the detection process. Since the z-movement speed is usually higher than the aspiration speed, the value must be smaller than the total aspiration volume to prevent potential spill-over. If the interface was detected before reaching this volume, the process stops.

Pipetting Speed [µL/s]: Define the aspiration speed.

Speed [mm/s]: Define the movement speed in z. Note: To reach the required pressure change, a higher z-speed than the standard liquid tracking speed might be required. Please keep in mind, that the final submerge will increase accordingly.

Acceleration [µl/s²]: Define the acceleration of the plunger axis.

Deceleration [µl/s²]: Define the deceleration of the plunger axis.

Trigger: Define in which direction the pressure change provides a detection trigger. Negative: Trigger by a decrease in pressure, which is the case when going from a less viscous to a more viscous liquid. Positive: Trigger by an increase in pressure, which is the case when going from a more viscous to a less viscous liquid. PositiveAndNegative: Trigger by either an increase or a decrease in pressure.

Window Size: Define the number of measurement points that will be evaluated in the algorithm for phase detection. A larger window size decreases the chance of false detections as it makes the algorithm more precise, however it also decreases responsiveness of the algorithm, resulting in later detections. Hence the window size should be as small as possible but as large as needed.

Window Average Size: Define the number of measurement points that will be evaluated due to averaging and thus smoothing the values to increase the signal to noise ratio. The higher the value, the larger the true pressure change needs to be however false triggers due to noise can be avoided.

Trigger Sensitivity: Define the trigger threshold. The lower the value, the higher the sensitivity. The threshold value, which is always positive, is applied in both positive and negative direction to detect both positive and negative pressure changes. Note: The active trigger direction needs to be defined in the “Trigger” parameter above.

Start Delay [samples]: Define the initial number of measurement points to ignore before starting pressure value evaluation.

Enable Warnings: Define whether a warning shall be raised if a liquid-liquid interface was not identified after completion. 0: no warning is shown when no interface was identified. 1: when no interface was identified, a warning is shown in TraceView and sent to Sample Tracking.

8.6.7.10 Dispense Liquid

Dispenses liquid and air gaps.

Liquid Volume: The volume to dispense in µl.

Dispense Acceleration: The acceleration used for the dispense in µl/s^2.

Dispense Speed: The max. speed used for the dispense in µl/s.

Dispense Deceleration: The deceleration used for the dispense in µl/s^2.

Use Tracking: When you want to use tracking support (0 = no tracking; 1 = use tracking).

8.6.7.11 Empty Tips

Dispenses the current contents of the selected tips.

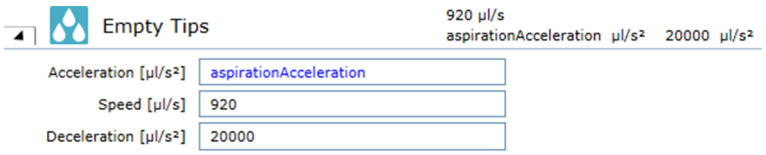


Fig. 123: Empty tips

Acceleration / Speed / Deceleration: Use acceleration, speed and deceleration parameters from dispense liquid.

8.6.7.12 Move Absolute Z

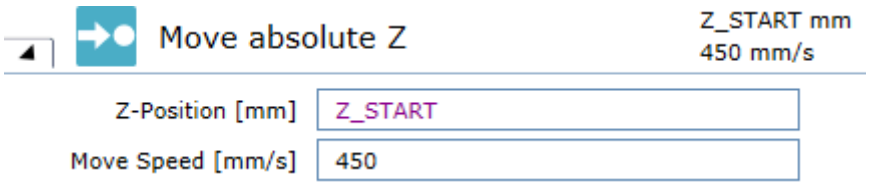


Fig. 124: Z-Position [mm]

Z-Position: Moves the Z-axis to an absolute position. The absolute Z-position to move to in millimeter.

Move Speed [mm/s]: The speed to move in millimeter per second.

8.6.7.13 Move Relative X/Y

If you enter **X-Distance** and **Y-Distance** there is first a move in X-direction and only subsequently (i.e. not simultaneously) in Y.

X-Distance: Enter **X-Distance** in [mm].

Y-Distance: Enter **Y-Distance** in [mm].

Move Speed: Enter the **Move Speed** in [mm/s].

8.6.7.14 Movements Limits

The microscript command **Movement Limits** allows to limit the speed and acceleration/deceleration with which the pipetting arm moves for the rest of the pipetting cycle. The limits are applied for moves outside the labware bounding box (i.e., when moving from the aspirate position to the dispense position).

The movement limits are automatically reset to 100% when a wash, drop tips or empty tips script command is executed.

8.6.7.15 Move Valve

Move the valve to a specific valve position:

- **2**: Tip
- **4**: Bypass
- **3**: System

8.6.7.16 Retract Supervised

Retracts the tip from the liquid while searching for an exit signal. If the signal is not found within the expect range, the error handling selected will be executed.

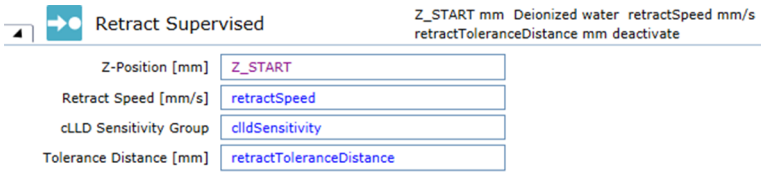


Fig. 125: Retract Supervised

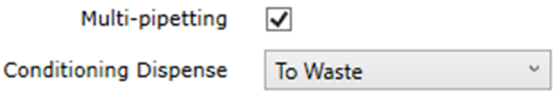
8.6.7.17 Pierce

This FCA MicroScript command allows to pierce tubes with rubber septum by constant force move of piercing tips.

This command also includes automatic retry error handlings and logic to reduce the chance of hardware damages.

*Do not use **Move Absolute Z** Micro Command for tubes with septum. **Move Absolute Z** can only pierce through sealed microplates.*

When selecting **Multi-pipetting**, select **To Waste** from the Conditioning Dispense drop-down list. If **To Source** were selected, an additional piercing move will be executed.



Z-Position: Moves the Z-axis to an absolute position in mm.

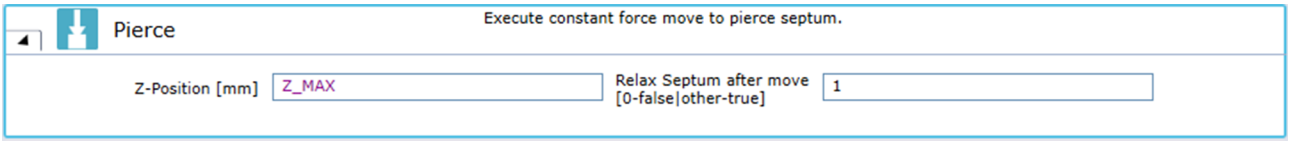


Fig. 126: Pierce

Relax Septum after move: Piercing into septum stresses hardware much and after reaching the target position there will still be pressure against the axis drives, which can cause overcurrent errors. If this option is activated the pressure after the move is released by allowing the septum to push the piercing tip a bit up again.

In addition this command will use following parameters from the Labware to Pierce:

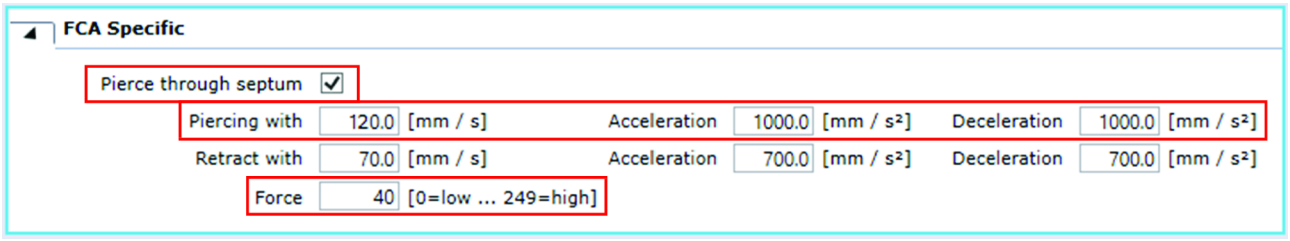


Fig. 127: FCA Specific

Pierce through septum: Must be activated, otherwise MicroScript command will raise error.

For tube types provided by Tecan values are already assigned to the following variables.

The use of custom tubes may require editing these values.

Pierce with: Speed for piercing move

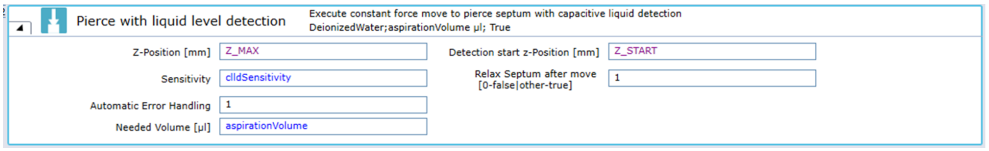
Force: Required Force to pierce through septum.

Acceleration: Acceleration of the piercing tip.

Deceleration: Deceleration of the piercing tip.

8.6.7.17.1 Pierce with Liquid Level Detection

This FCA MicroScript command allows to pierce tubes with rubber septum by constant force move of piercing tips with capacitive liquid level check.



Move logic is the same as the Pierce Micro Command with the following difference:

- Move is split into two parts: Move to Detection start position and then start second move with detection to final position.

Sensitivity: For piercing moves into high conductive liquids (biofluids) the sensitivity group 7 was developed and is recommended to be used. In order to use sensitivity group 7 set the **cLldSensitivity** variable to 7.

Z-Position: Target position to reach after detection.

Detection start z-Position: Position to be reached before detection will start. This first part move is required to get through the septum of a tube.

Automatic Error Handling: Number of attempts to automatically detect the liquid before the main error handling is executed in case of a cLLD error (e.g., no liquid, not enough liquid, etc.).

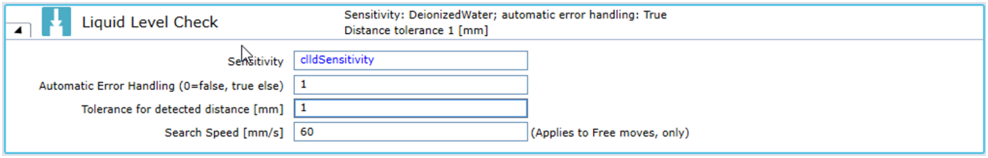
Main Error Handling of Piercing with Liquid Level Detection

When the detected liquid is below **Needed Volume**, the following error handling options are available:

- Aspirate anyway: Ignore the error and continue micro script. Risk of aspirating air.
- Aspirate available volume: Reduce the volume to aspirate to the available volume.
- Deactivate Tip: Do not use to aspirate the liquid and deactivate the tip until next wash command is executed.
- Abort run: Run will be aborted.
- Retry: another retry is executed.

8.6.7.18 Liquid Level Check

This FCA MicroScript command allows to detect the Liquid Level in a cavity. If the level is not equal to the expected level within a tolerance, an error will be shown.



The detection move starts at current position and ends at Z-Max if no liquid was found. In case of a retry and at the end of the measurement, a retract to start position of the move is executed.

Liquid Level Check can be used in both aspirate and dispense section. However, the error handling options is optimized for dispense sections.

Sensitivity: The cLLD Sensitivity that has been defined in the Detection and Position Section of a Liquid Class Section can also be used to overwrite the general settings with another value. For piercing moves into high conductive liquids (biofluids) the sensitivity group 7 was developed and is recommended to be used. In order to use sensitivity group 7 set the clldSensitivity variable to 7.

Automatic Error Handling: If this option is set to a value other than 0, the selected automatic error handling option of the Microscript Section will be executed in case the detection results in an error.

Tolerance for detected distance [mm]: Defines how much the detected liquid level may deviate from the expected result (i.e., mm up and down).

Search speed [mm/s]: Defines the speed for the detection move for labware without septum. In case of labware with septum, piercing with piercing move parameters as defined in the labware itself is used.

Main Error Handling of Liquid Level Check

When the detected liquid level is outside of the range that was defined as acceptable the following options are available:

- Ignore and continue: The detected error will be reported to sample tracking if installed and configured and in the log files. The following actions will be executed as normal.
- Deactivate tip: The tip of the channel that has detected the error will be deactivated. No further actions will be executed for this channel until the fixed tip is washed or the disposable tip is ejected.
- Abort Run: The run will be aborted immediately. No further actions will be executed for any channel.

- Retry: Another retry from start position will be executed with the affected channel.

8.6.7.19 Retract Tip from Septum

This FCA MicroScript command allows to retract a piercing tip from a tube with a septum with constant force move.

Z-Position: Moves the Z-axis to an absolute position. The absolute Z-position to move to in millimeter.

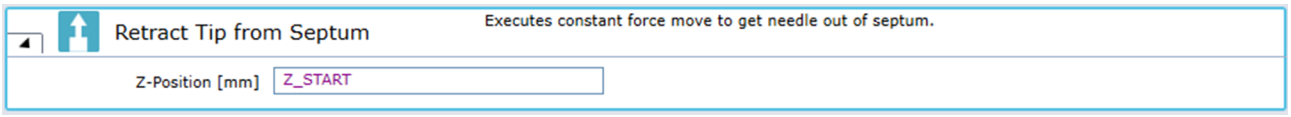


Fig. 128: Retract Tip from Septum

In addition this command will use following parameters from the Labware to Pierce:

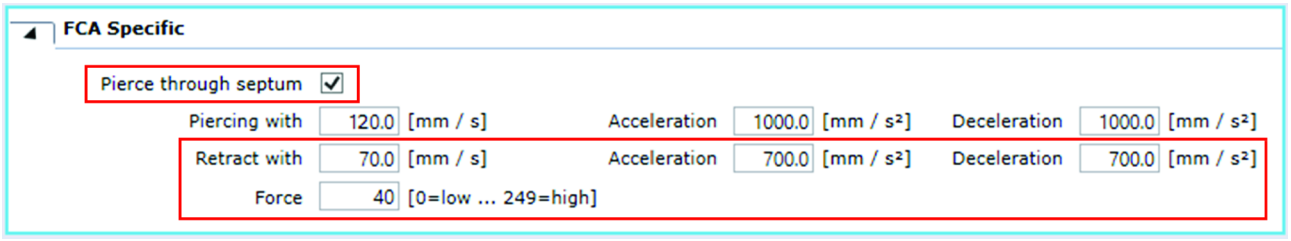


Fig. 129: FCA Specific

Pierce through septum: Must be activated, otherwise MicroScript command will raise an error.

For tube types provided by Tecan values are already assigned to the following variables.

The use of custom tubes may require editing these values.

Retract with: Speed of retract move

Force: Required force to retract the piercing tip from the tube with septum.

Acceleration: Acceleration of the piercing tip.

Deceleration: Deceleration of the piercing tip.

8.6.7.20 Rapid Wash

Dispense using the Rapid Wash Pump. Speed is % of maximum speed of the instrument as configured.

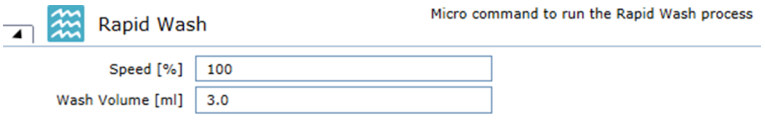


Fig. 130: Rapid Wash

8.6.7.20.1 Start Rapid Wash

Dispense using the Rapid Wash Pump.

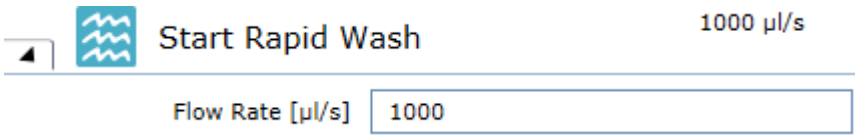


Fig. 131: Start Rapid Wash

8.6.7.20.2 Stop Rapid Wash

Turn off Rapid Wash Pump.

8.6.7.21 Delay

The programming command **Delay** delays the execution of a **MicroScript** for a specified amount of time. You can specify the delay in [ms], either by entering a numerical value or a variable/expression. The value must not be negative. When executing the command, the software simply waits for the specified time, without displaying any prompt.

Delay [ms]: Specify the delay in [ms], either by entering a numerical value or a variable/ expression. The value must not be negative. When executing the command, the software simply waits for the specified time, without displaying any prompt.

On active stop of the run the timer will be stopped and the command will be skipped on retry.

9 Specifying Your Instrument Setup

This topic describes how to specify the setup of your pipetting instrument in FluentControl. FluentControl automatically detects the presence of certain instrument options by interrogating the pipetting instrument. However, it is necessary to provide FluentControl with additional information that cannot be determined by interrogation, such as the dimensions and properties of the carriers and the labware (e.g., microplates) you want to use, and with other details, such as the barcode types that should be accepted by the barcode scanner. The latter depend on the design of your barcode labels.

FluentControl stores information on fixed and variable parts of the pipetting instrument in an internal database. It also stores information on the dimensions of carriers and labware items and the configuration settings for the optional devices.



If you are setting up a new pipetting instrument, FluentSetup must be used. Check and calibrate the reference coordinates of the hardware. This is normally done for you by the Tecan Field Service Engineer when installing the pipetting instrument and the optional devices. For related information on this software refer to section "Reference Documents" [11].

FluentControl enables you to define an object (e.g., a microplate) once, and then to use it at any required position on the worktable. This is because FluentControl uses a hierarchical system for the coordinates:

- **Labware coordinates**: The labware coordinates are relative to the site on which the labware is placed.
- **Site coordinates**: The site coordinates are relative to the carrier.
- **Carrier offsets**: The carrier offsets are relative to the grid position on the worktable.
- **Grid positions**: The grid positions on the worktable have a periodic spacing of 25 mm.

This hierarchy allows you to place the labware at any suitable site, carrier and worktable position without having to redefine its coordinates.

The following lists the items you need to configure when setting up the pipetting instrument in FluentControl (some of the items refer to options that may not be fitted to your instrument):

- **Choose a Carrier**: When you set up the worktable for your FluentControl script, you first choose a carrier with the required number of sites. Each of the sites can hold labware. If you cannot find the carrier you need, you can define a new carrier by duplicating an existing one. Be aware that you are duplicating from a similar carrier. Otherwise this can lead to arms colliding with the worktable. Special carriers are hotels, carousels, stackers and passive stacks. Bounding boxes of labware or carriers placed onto those special carriers will not be considered by the PathFinder.
- **Choose the Labware**: You then choose the labware you want to use. If you cannot find the labware you need, you can define a new labware item.
- **Position the Carriers**: Position the carriers on the worktable and place the labware onto the carriers.

- **Editing an existing Script**: The procedure to follow to set up a script is described in detail in Editing an existing script, refer to section “Editing an Existing Script” [116].



The FluentControl database already contains a large selection of pre-defined carrier and labware types. You only need to create or edit carriers or labware if you cannot find a standard item that is suitable for your needs.

9.1 Configuring Your Drivers

In the Available Drives section you find a list of all available drivers. By clicking the relevant driver checkbox the driver software is enabled. This is only applicable if the assigned device is installed and placed on the deck.

Select the I/O State check box for all devices which are installed and physically connected to the PC FluentControl is running on. If the I/O is on, FluentControl will try to interact with the device or module at run time. If the I/O is turned off, the commands are still visible in the software (for most but not all drivers) and FluentControl will not interact with the device or module at run time.

10 Validation Checklist

To ensure against malfunction and incorrect results, all FluentControl runs must be validated before they are used for pipetting with real samples. To ensure that all your runs operate as intended, it is highly recommended to perform a test run. This topic gives an overview of some important points to be checked.

The following validation checklist is not intended to be exhaustive and does not replace any local laws or statutes that apply.

Important points to be considered when validating FluentControl applications:

Tab. 6: Checklist for validating FluentControl applications

Done (x)	Task
	Check the Log File: Start FluentControl and, before you start pipetting, check the log file for errors. If there are error messages, the cause of all errors must be remedied first!
	Instrument: Make sure that the status bar lists the Instrument Mode as Real. STEP RESULT: In simulation mode, the system will not pipette any liquid! Do not use the simulation mode to validate a run. Make sure that the status bar lists the Instrument Mode as Real.
	Make sure that the correct arms and devices are used during the run.
	Make sure that the script or method is saved and reloaded before you start the validation.
	Make sure that you have loaded the correct script or method for the assigned pipetting task.
	Verify that Context Check is enabled in the Run menu and check the script editor window for errors (commands which have errors are shown in red).
	Check the liquid containers and system liquid of the pipetting instrument for the correct liquid types and concentrations which are required by the script or method.
	Before running the script or method, you must physically set up the worktable with the carriers, labware and devices which are needed by the application or assay. The administrator or owner of the pipetting instrument must establish standard operating procedures for setting up the worktable and make them available to everyone who carries out pipetting.
	After setting up the worktable with the carriers, labware and devices which are needed by the application or assay, inspect the paths that the instrument arms (FCA, RGA, MCA384 and MCA96) will take during the script/method run to make sure that there are no obstacles which could obstruct the movements.

Done (x)	Task
	Observe the arm initialization at the pipetting instrument: • On the Run menu, click Initialize Instrument. • Make a visual check that the initialization sequence has been carried out correctly. • Check the log file for initialization errors. ADDITIONAL INFORMATION: If error messages are found, the cause of the error must be remedied before you start pipetting.
	Before you pipette using real samples, test run the script or method to make sure that: • The correct liquid classes are used for pipetting. • Pipetting takes place at the intended positions on the worktable (e.g., correct micro plate, correct well). • The correct volumes of liquid are pipetted. • Pipetting takes place in the correct sequence as required by the application. • Plate movements are as expected and with no visible drops or collisions. • Plate movements from storage locations (e.g., stacker, carousel, hotels) fetch the expected plates and deliver them to the expected positions. • Taking objects to waste positions functions as expected (with either the FCA or RGA) and that the waste chute/position remains open and accessible throughout the run.
	If your script employs Worklisting, validate the creation of the input files and that the expected pipetting positions are used at run-time with differing input files.
	If you are using a Variable Pipetting Loop or variables to impact the number of samples in a run, verify that the correct number of samples is pipetted.
	If your script uses cLLD functionality (liquid level detection, aspiration supervision, retract supervision), make sure that the related error handling is executed correctly (as selected in the liquid class or the LLD Error Handling script command). • To ensure accurate liquid detection, make sure that the labware is correctly seated on the carriers. • To verify that the cLLD system is operating, execute the cLLD Self Test direct command. NOTE: The capacitive detection system is not suitable for all liquid types (e.g., liquids with foam, non-conductive liquids).

Done (x)	Task
	When setting up an assay for a liquid which can clot, make sure that retraction supervision is enabled in the liquid classes which are used to handle the liquid. Check the execution of the retraction to verify that the tips are outside the liquid afterwards. Refer to the Good Usage information in the Operating Manual for the pipetting instrument for additional recommendations.
	If your script uses septum piercing, make sure that the piercing movement is performed correctly.
	If your script uses arm movement commands, make sure that the arm moves to the correct position.
	Do avoid firmware commands. If your script or method sends firmware commands to the pipetting instrument or to other devices, make sure that the correct firmware commands have been specified and that they initiate the intended actions. Also verify that the results are as expected as firmware commands are not tracked by modules such as Sample Tracking.
	If your script or method uses a microplate reader, check the results (output data) from the reader to make sure that they are plausible.
	Make sure that all liquid transfers work correctly. When setting up an assay which uses a trough (container for a reagent liquid), make sure that the liquid classes which are used for aspirating from the trough are configured to immerse the tips in the liquid by at least 1 mm.
	Verify that the washing actions are executed correctly. Appropriate tip washing actions must be provided in the script or method to avoid cross-contamination of samples and/or reagents. Disposable tips can also be used to avoid cross-contamination of samples and/or reagents. Design and test run the script or method to make sure that cross-contamination will not take place.
	When using disposable tips, make sure that: • The tips are correctly picked up and dropped / set back. • The correct labware is used for picking up and dropping / setting back tips.
	When using the Active Carrier for the MCA384, verify that the carrier is functioning properly when picking up or dropping tips.
	If your script uses a barcode reader, make sure that the barcodes are scanned correctly and are assigned to the correct labware. This includes standalone readers or those integrated in devices like a Stacker or Carousel.
	If your script uses a robot to transport labware, make sure that the robot paths are executed correctly.

Done (x)	Task
	If your script or method uses a shaker or an incubator that shakes, make sure that the shaker parameters (e.g., shaking time, shaking frequency) have been configured correctly and that the shaking functions as intended.
	If your script or method uses an incubator, make sure that the incubator parameters (e.g., incubating time and temperature) have been configured correctly and that the incubator operates as intended, including if applicable the functionality of the doors.
	If your script or method uses devices with heating elements (e.g., shaker, heated incubator, some microplate readers), make sure that the script or method allows enough time for the heater to heat up to or cool down to the intended temperature before it is used. Test run the script or method and measure the temperature which has been achieved.
	If your script or method uses operations where timing is critical (e.g., incubator, shaker), make sure that the timing and delays have been configured correctly and work as intended: Check the time specified in the respective device command. Check the time specified in Wait For Timer commands.
	If the system pauses for any reason when running a script or method, the duration of the pipetting script may differ from the intended time. For example, sample incubation time may be outside of the specified limits. Depending on the application, this can make the assay invalid. A system pause can be caused for example by clicking the Pause button in FluentControl's Runtime Controller dialog box, by pressing the Pause button on the pipetting instrument, by a pipetting error or by an internal timeout if a step has taken longer than expected or specified.
	If your script uses the Touch Tools RUP script command, make sure that: • The contents of the prompt are displayed correctly. • Modifications to variables are executed correctly.
	If your script or method uses e-mail for notification of important events, the notification function must be used to send a test e-mail to make sure that the e-mail is sent correctly and is received by the intended recipient.
	If your script or method imports required information from external files (bar code lists, lists of values for variables, etc.): Make sure that the correct file name has been specified in the script or method. Make sure that the import file exists and contains the intended data.
	If your script exports information (e.g., variable values) to files, make sure that the export file has been generated correctly and contains the correct data.
	If your script uses variables or expressions, make sure that the correct values are used during the run.

Done (x)	Task
	Check that the script or method starts to run as intended when you click the Start button in the Runtime Controller or touch interface.
	If your method consists of more than one batch or contains several scripts, check that the batches and scripts are executed in the correct order.
	If your script uses subroutines, check that: • The subroutines are loaded and executed correctly. • Concurrent arm movements are performed correctly and without collisions.
	When the script or method has completed, check the status information and the Warnings / Errors window in Runtime Controller and Trace View, and check all log files to make sure that the entire script or method has completed and that there are no error messages.
	The results of an assay are only valid if the entire script or method completed without errors. If this is not the case, the entire script or method must be repeated after rectifying the cause of all errors.
	If your script or method uses Sample Tracking to generate a report, verify that all values and actions performed (e.g., pipetting and transferring labware) are correctly reported and assigned to the expected labware and wells.

11 Commands

This chapter contains detailed information about the usage of commands and smart commands. Smart commands are used to program liquid handling workflows for the FCA and MCA arms and comprises the following commands: **Sample Transfer (FCA), Reagent Distribution (FCA), Standard Curve (FCA)**, **Transfer Individual Volumes (FCA)** and **Transfer Liquid (MCA96 and MCA384)**. They are able to create so called smart routines, while normal commands cannot do this.

11.1 Direct Commands

Direct commands are used to execute an action without running a script line (e.g., for picking up DiTis or flushing the tips). They provide a convenient alternative to creating and running a pipetting or maintenance script that carries out the same function.

The Controlbar only shows direct commands for devices whose device driver is installed and configured(e.g., the DiTi commands are only shown if the FCA is configured with DiTis).

To run a Direct command, right-click on the command and then click Run direct in the shortcut menu. Alternatively, you drag the Direct command to the run area of the Start tab.



Direct Commands will use the worktable which is open in the Worktable Editor when the Direct Command is executed. If no worktable is open, the Direct Command will use the worktable that has been most recently used in a run, including the most recent DiTi positions. Before executing a Direct Command, make sure that the arm is outside a contouring-error prone position (i.e., bounding boxes).

11.1.1 Get Tips (FCA)

Use this command together with a FCA to pick up *Disposable Tips (DiTis)* of the specified type from a DiTi rack. Keeps track of their position on the worktable and automatically picks up the next available unused DiTis of the chosen type.

The system tracks available DiTis in the Get Tips script command for the FCA arm by automated disposable tip tracking for single and multiple DiTi types. The system performs automated DiTi tracking by automatic tracking of the position in the labware with the next available DiTi. If the software has restarted or a new power cycle has started, the system continues automatic tracking of the next DiTi from the previous application or power cycle.

Parameters



Fig. 132: Parameters

DiTi Type: Click the DiTi Type arrow to see more options and select the DiTi type of the FCA arm from the list.

Proceed to next position on error: Check box to set whether **Proceed to next position on error** is activated or not. If DiTis cannot be picked up at the current position, the software automatically retries fetching tips. If DiTis could not be picked up after three attempts, error handling is performed as follows:

- If this check box is checked, the software proceeds to the next available position and tries again.
- If this check box is not checked, an error prompt is displayed.

Advanced Parameters

Tip Selection (optional): For this parameter definition, follow the cross reference to **Tip Selection (optional)**.

Air gap volume [µl]: The volume for the air gap in microliters.

Airgap Speed [µl/s]: In the **Airgap Speed** box, enter the air gap speed you want to use in microliters per second.

Robotic Actions

Prior to execution of the script command, the system performs the following:

- If parameter **Airgap volume** is set to zero and the last pipetted volume in variable **tipContentVolumes** is zero, the system aspirates an air gap of volume 20 µl.
- If parameter **Airgap volume** is not set to zero and the last pipetted volume in variable **tipContentVolumes** is zero, the system aspirates the air gap as defined by the **Airgap volume** parameter.

11.1.2 Drop Tips (FCA)

Use this command to discards the specified DiTis into the specified DiTi waste.

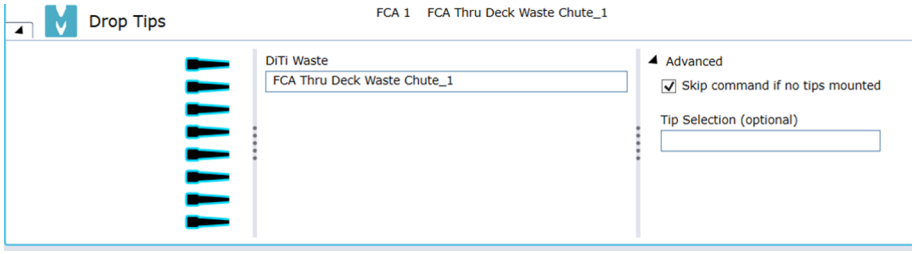


Fig. 133: Drop Tips

Parameters

Tip Selection: Select the tips you want to use. If you do not click a tip, uses all tips by default. To turn off or turn on a tip, click on it.

DiTi Waste: Specify the DiTi waste on the worktable: Click into the DiTi Waste box and select your labware from the Worktable Editor or type the name of your labware.

*The **Skip command if no tips mounted** checkbox is enabled in script commands automatically generated by smart commands and worklists to ensure that the generated script starts without mounted tips while not triggering an unnecessary mechanical move and associated delay.*

11.1.3 Pick Up Tips (FCA)

Robotic Actions: Prior to execution of the script command, the system performs the following:

- If parameter **Airgap Volume** is set to zero and the last pipetted volume in variable **tipContentVolumes** is zero, the system aspirates an air gap of volume 20 µl.
- If parameter **Airgap Volume** is not set to zero and the last pipetted volume in variable **tipContentVolumes** is zero, the system aspirates the air gap as defined by the **Airgap Volume** parameter.

Parameters

Tip Selection: Select the tips you want to use. If you do not click a tip, FluentControl uses all tips by default. To turn off or turn on a tip, click on it.

To avoid errors at run time, the Pick Up Tips command should not be used to fetch tips from nested or stacked labware. The Get Tips command must be used instead.

Labware: In the Labware box, either type your expression or click the labware displayed in the Worktable editor window (Worktable Click).

Well Offset: Select by how many wells the starting point of the currently selected wells is shifted. This is usually expressed as a variable.

Advanced Parameters

Tip Selection (optional): For this parameter definition, follow the cross reference to Tip Selection (optional).

Tip Offset: This parameter lets you modify the tip selection for the pipetting command. The default is 0, which deactivates the offset. A positive value increments the tip number(s) which are specified in the pipetting command. For

example, if the pipetting command specifies tips 2 and 3 and you set this parameter to 1, tips 3 and 4 will be used instead. Negative values of the parameter shift the tip selection in the opposite direction.

Air gap volume [µl]: The volume for the air gap in microliters.

Airgap Speed [µl/s]: In the Airgap Speed box, enter the air gap speed you want to use in microliters per second.

11.1.4 Set Tips Back (FCA)

The Set Tips Back command returns used DiTis to specified positions on a DiTi rack for later use. If the FCA arm is equipped with the **Lower DiTi Eject** option, the system provides a **Set Tips Back** command to return used DiTis to specified positions on a DiTi rack for later use.

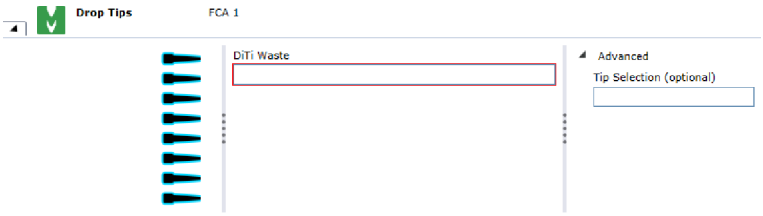


Fig. 134: Drop tips

Parameters

Tip Selection: Select the tips you want to use. If you do not click a tip, FluentControl uses all tips by default. To turn off or turn on a tip, click on it.

To avoid errors at run time, the Pick Up Tips command should not be used to fetch tips from nested or stacked labware. The Get Tips command must be used instead.

Labware: In the Labware box, either type your expression or click the labware displayed in the Worktable editor window (Worktable Click).

Well Offset: Select by how many wells the starting point of the currently selected wells is shifted. This is usually expressed as a variable.

Advanced Parameters

Tip Selection (optional): For this parameter definition, follow the cross reference to Tip Selection (optional).

Tip Offset: This parameter lets you modify the tip selection for the pipetting command. The default is 0, which deactivates the offset. A positive value increments the tip number(s) which are specified in the pipetting command. For example, if the pipetting command specifies tips 2 and 3 and you set this parameter to 1, tips 3 and 4 will be used instead. Negative values of the parameter shift the tip selection in the opposite direction.

Lower DiTi Eject: Hardware option to discard DiTis at a lower height—e.g., to minimize contamination due to aerosols that could arise when the DiTis drop down into the DiTi Waste. This option is required by the Set DiTis Back command.

11.1.5 Flush (FCA)

The **Flush** script command flushes the selected tips. The **Flush** command is used also as a direct command.

*Most of the features and dialog boxes in FluentControl are provided with context-sensitive help that can be accessed with the **F1** key.*

The arm moves over the selected waste and flushes the selected tips according to section "Flush" of the selected liquid class.

After a Flush command was executed, the system resets the maximal speeds, acceleration and deceleration for FCA movements in x, y, and z defined by the MicroScript command to the predefined values in the system configuration file.

Parameters

Waste: Specify the name of the waste on the worktable: Click into the Waste box and select your labware from the Worktable Editor or type the name of your labware.

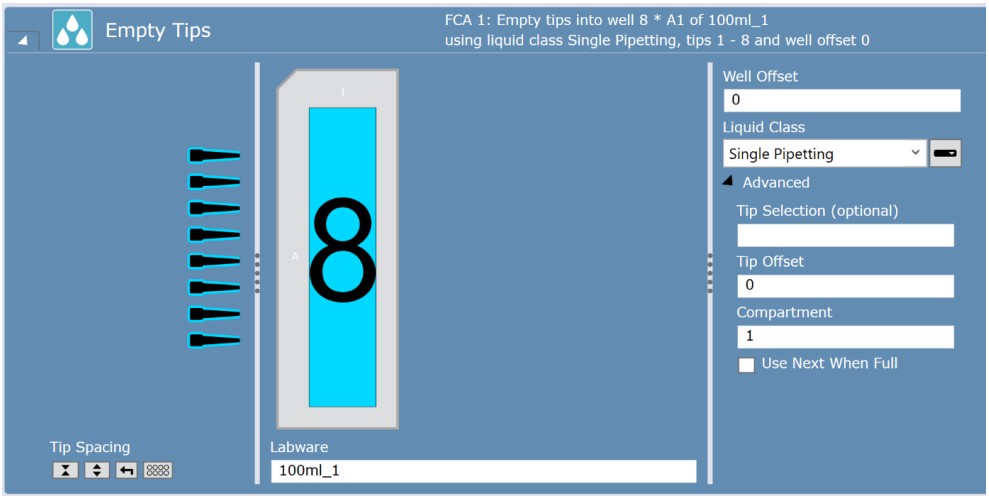
Liquid Class: Choose the required liquid class from the pull down list or by means of an expression typed in (e.g., using a variable defined within your script). The list only shows liquid classes that are suitable for the current configuration of the pipetting instrument. For example, you will not see liquid classes for DiTis if the arm is not currently configured with DiTis.

Flush Volume: Define the flush volume.

Advanced Parameters

Tip Selection (optional): For this parameter definition, follow the cross reference to Tip Selection (optional).

11.1.6 Empty Tip (FCA)



This command dispenses the entire content of the selected tips into specified positions on the worktable.

Parameters

Tip Selection:

Select the tips you want to use. By default, the software selects all tips. To turn off or turn on a tip, click on it. If the box is framed in red, click to select a tip.

Tip Spacing: The Tip Spacing parameter controls the distance between adjacent pipetting tips for this command. Tip spacing is only relevant if you want to use more than one tip. You can only choose values for tip spacing which are meaningful for the labware geometry. The control offers the following options (from left to right):

Contract:

- If wells are already selected in the labware control and the contracted well selection spacing is physically possible, the well selection alters to match the new well selection cursor spacing.
- If wells are already selected in the labware control and the contracted well selection spacing is physically not possible, only the first well is selected for all tips.
- If only one well is selected in the labware control, clicking Contract has no effect on the selected well.

Expand:

- If wells are already selected in the labware control and the expanded selection is physically possible to select, the well selection alters to match the new well selection cursor spacing. The first selected well is not altered by the Tip Spacing buttons.
- If wells are already selected in the labware control and the expanded selection is not physically possible to select, clicking Expand has no effect.

Reset:

- Reset returns the tip spacing to the default.
- If wells are selected, it selects additional wells by basing a default spacing on the selected tips and starting with the first selected well.
- If wells are selected and it is physically not possible to select additional wells by basing a default spacing on the selected tips starting with the first selected well, all selected wells are cleared.

Independent Selection:

- Clicking on an unselected well in the labware control selects it for one tip.
- Clicking on a selected well in the labware control selects it for one additional tip.
- It is not possible to select more wells than the number of tips selected.
- If there are the same number of wells selected as tips and you click on a selected well, it reduces the number of tips in the well by 1 (potentially unselecting the well).
- The tip to well assignment is consecutive where tips are assigned from 1 to n to the corresponding 1 to m wells according to the well selection per tip.

If the number of tips selected is lowered to a number smaller than the number of selected wells, a corresponding number of selected wells switches to not selected starting with the highest numbered well.

Labware: In the Labware box, either type your expression or click the labware displayed in the Worktable editor window (Worktable click).

Well Offset: Select by how many wells the starting point of the currently selected wells is shifted. This is usually expressed as a variable.

Liquid Class: Choose the required liquid class from the pulldown list or by means of an expression typed in (e.g. by using a variable defined within your script). The list only shows liquid classes that are suitable for the pipetting arm and which support the empty tips workflow.

Advanced Parameters

Tip Selection (optional): The Tip Selection (optional) parameter is resolved as integer (integer variables and expressions are allowed) and interpreted then as a bitmask. The tip selection in the script command is calculated considering both the Tip Selection (optional) bitmask and the tips selected via GUI (binary AND). Thus, only tips selected in both masks will be used for pipetting.

The Tip Selection (optional) mask is specified as a numeric, bit-coded value (tip1 = 1, tip2 = 2, tip3 = 4, tip4 = 8, tip5 = 16, tip6 = 32, tip7 = 64, tip8 = 128). Use the sum of the bit-coded values to select more than one tip. In the GUI, tip1 is the topmost tip.

Tip Offset: This parameter lets you modify the tip selection for the pipetting command. The default is 0, which deactivates the offset. A positive value increments the tip number(s) which are specified in the pipetting command. For example, if the pipetting command specifies tips 2 and 3 and you set this parameter to 1, tips 3 and 4 will be used instead. Negative values of the parameter shift the tip selection in the opposite direction.

Compartment: Some special labware types (e.g., for protein crystallography) have wells with several compartments (cavities). Each well typically has one main compartment and one or more satellite compartments which are directly adjacent and are often smaller in size. Choose the compartment which you want to use for pipetting (the main compartment is normally compartment 1). Well compartments are specified in the labware configuration.

X offset [mm]: This is the offset of the tip from the center of the target well (negative = to left, positive = to right).

Y offset [mm]: This is the offset of the tip from the center of the target well (negative = to front, positive = to back).

Use next when full: If this option is selected, the software performs the following checks before executing the dispense:

If the selected labware contains the custom attribute **LiquidWasteCapacity**, the software checks whether the remaining capacity (i.e. the **LiquidWasteCapacity** minus the current volume in the cavity) is sufficient for the volume to be dispensed. If the remaining capacity is insufficient, the software will automatically switch to the next labware in the group (e.g. if “Trough[001]” is full, it will choose “Trough[002]”). If all labware items in the group are full, the software will prompt the user to empty them and start with the first selected labware again.

11.1.7 Sensitivity Calculation (FCA)

The **Sensitivity Calculation** direct command determines the cLLD sensitivity of the unknown liquid by an automatic procedure.

Parameters

- **Labware**: In the Labware box, either type your expression or click the labware displayed in the **Worktable** editor window (*Worktable click*).
- **DiTi Type**: Select the DiTi type of the FCA arm.
- **DiTi Waste**: Specify the DiTi waste on the worktable:

1. Click into the **DiTi Waste box**.
2. Select your labware from the **Worktable** editor or type the name of your labware.

11.1.8 cLLD Self Test (FCA)

The **cLLD Self Test** command executes an on-board self test for the cLLD system. It verifies the functional state of the electronic board and the sensing of interfering signals.

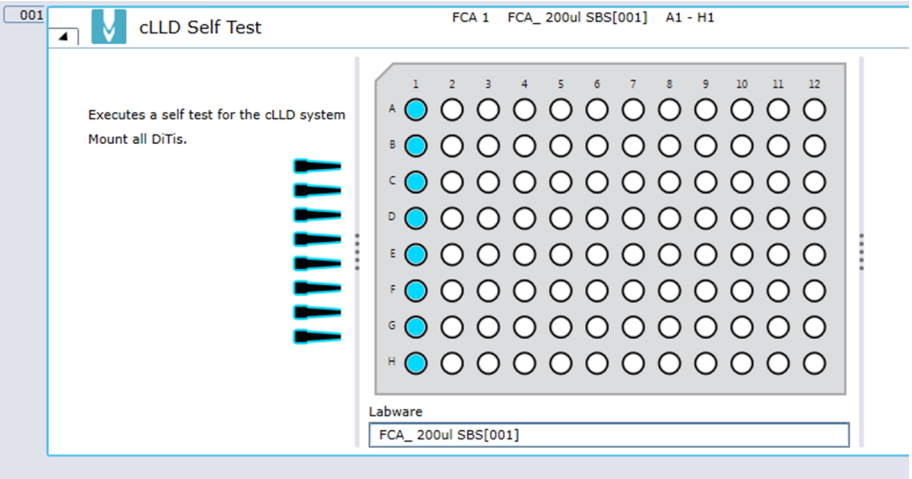


Fig. 135: cLLD Self Test

Description

The system picks up the specified DiTis and performs the cLLD self test.
After the self test is performed, the system puts the DiTis back to their original pick-up position.
If the cLLD self test passes, the system displays the message **cLLD self-test passed**.
If the cLLD self test fails, the system displays the tips that caused the error and the message **Please contact Tecan Helpdesk**.

Parameters

Tip selection: By default, all tips are selected for the self test. Select and deselect tips by clicking on their symbol.

Labware: A DiTi box has to be selected. In the **Labware** box, either type the labware name, or click on the labware displayed in the **Worktable** editor.

11.1.9 Inline Filter Check (Air FCA)

Use this command to detect clogged or damaged filters. Find out whether you need to replace a filter and retry the check to enable plunger motion on that channel.

In order to protect the Air FCA pipetting channels against over-aspiration of liquid (if a DiTi with smaller capacity then intended is used) a control system is installed on each channel. The control system combines a filter inside the DiTi cone and the pressure monitoring of the PMP board. With the Inline Filter Check command you can check the filter inside the DiTi cone of an Air FCA pipetting channel. You can check the presence and integrity of the filters periodically (perhaps at the beginning of a script).

Before processing the command make sure that there are no DiTis mounted on the selected channels.

Failures detected by the test

- Detectable failures:
  - Wet inline filter (due to over-aspiration of liquid)
  - Damaged filter
  - Wrong positioned filter
  - Absence of filter
  - Channel clogged with particles

Procedure

The **Inline Filter Check** is performed at the current position unless Fluent needs to drop DiTis and perform the test over the DiTi waste. If the test succeeds, then the measured values are written to the log file (search for the Action "InlineFilterTest" in the Log Channel "Tecan/FluentControl/Instrument/FCA"). Plunger motion is now enabled for the selected channels. If the test fails, you get an error message and abort the run. Replace the specified filters as described in the Fluent Operating Manual and retry. Refer to section Reference Documents.
Channels that do not pass the test are not available.

Parameters

Tip Selection: Select the tips (channels) for which you want to test the inline filter.

Inline Filter: Process security through inline filter.

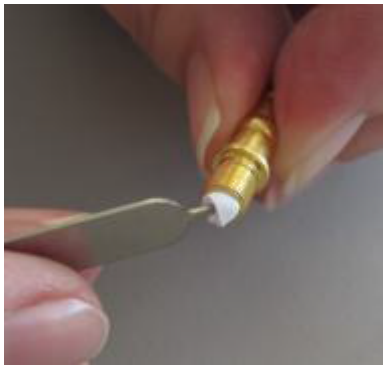


Fig. 136: Remove inlet filter (1)

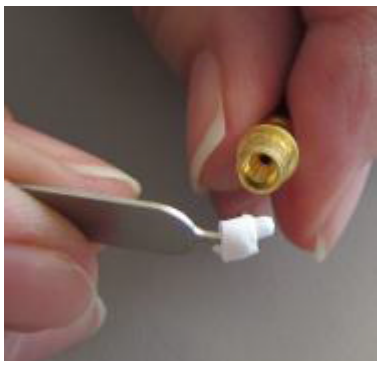


Fig. 137: Remove inline filter (2)

Features

The inline filter has the following features:

- The pressure sensor permanently tracks the pressure in the pipetting channels.
- Unlikely event: Wrong disposable tip size is used (e.g., 50 µl instead of 200 µl DiTi and user tries to pipette 100 µl).
- Liquid comes into contact with the inline filters.
- Pressure drops under a pre-defined threshold.
- Event is detected by the pressure sensor.
- System causes immediate plunger stop.
- By detecting the pressure changes which occur if liquid comes into contact with the system's inline filters, the system is able to stop the aspiration immediately, preventing liquid entering the air chamber.

The inline filter cannot prevent aerosols—only filtered DiTis can.

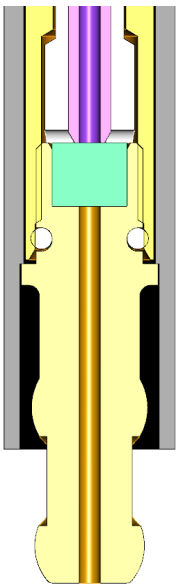


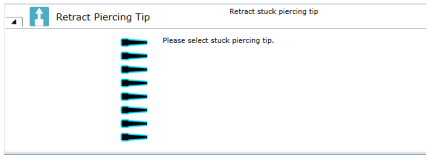
Fig. 138: Air displacement chamber – inline filter – golden DiTi cone



Fig. 139: DiTi cone with inline filter

11.1.10 Retract Piercing Tip (FCA)

This command can be used as direct command or as script command to retract a piercing tip from a tube labware instead of doing it manually. The command will perform a constant torque move to z-travel.



11.1.11 Get Fingers (FCA)

Use this command to pick up the FCA gripper fingers. Refer to section “Get Fingers (FCA Gripper)” [▷ 276].

11.1.12 Drop Fingers (FCA)

Use this command to place the FCA gripper fingers to a specified position on the deck. Refer to section “Drop Fingers (FCA Gripper)” [▷ 276].

11.1.13 Fetch Labware for Teaching (RGA)

Use this command to pick up labware and carriers in order to teach sites with the RGA.

Carriers

When teaching sites with the RGA, the carrier name has to be typed in manually. The carrier needs CustomAttribute **MoveableCarrier** set to true and to specify CustomAttributes **Grip_Narrow** and **Grip_Wide**.
24 Eppendorf Adapter is a moveable carrier and can be used as example.

Parameters

Labware: In the Labware box, either type your expression or click the labware displayed in the **Worktable** editor window (*Worktable click*).

11.1.14 Return Labware for Teaching (RGA)

Use this command to return labware or carriers after using for teaching.

Carriers

When returning carriers, the carrier name has to be typed in manually. The carrier needs CustomAttribute **MoveableCarrier** set to true and to specify CustomAttributes **Grip_Narrow** and **Grip_Wide**.
24 Eppendorf Adapter is a moveable carrier and can be used as example.

Parameters

Labware: In the Labware box, either type your expression or click the labware displayed in the **Worktable** editor window (*Worktable click*).

Target location: Select a target location for used DiTi racks in the list box. The usage of "<BASE>" as an alias for the base location is supported.

Target Position: To specify the target site, click in the **Target Position** box and type the site number (or an expression which resolves into a site number) or select a carrier or site on the worktable.

11.1.15 Get Fingers (RGA)

Use this command to pick up the RGA gripper fingers. Refer to section “Get Fingers (RGA)” [▷ 282].

11.1.16 Drop Fingers (RGA)

Use this command to place the FCA gripper fingers to a specified position on the deck. Refer to section “Drop Fingers (RGA)” [▷ 282].

11.1.17 Pick Up Tips (MCA96)

Use this command to pick up disposable tips for the MCA96 from a specified position on the worktable.

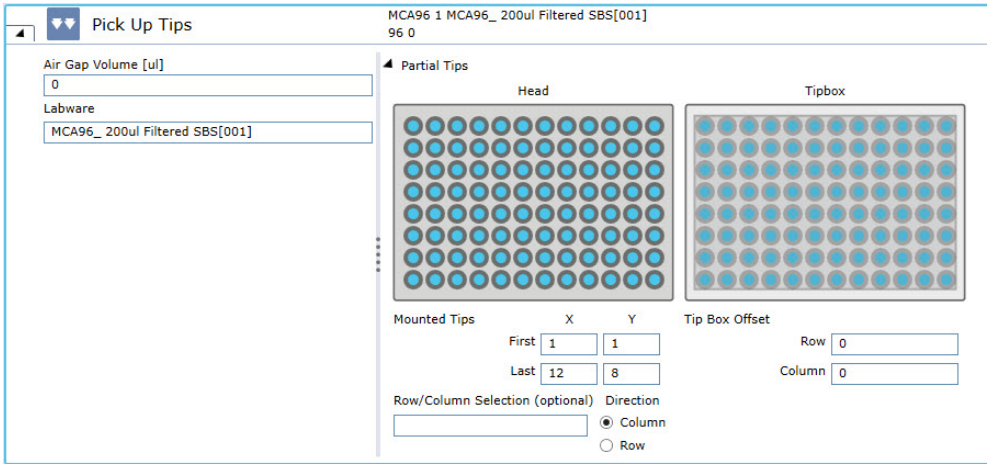


Fig. 140: Pick up Tips

Parameters

Air gap volume [µl]: The volume for the air gap in micro liters.

Labware: In the Labware box, either type your expression or click on a labware displayed in the Worktable editor window (Worktable click).

Tips mounted

First X: Define the first tip position on the head in X-direction where the tips are mounted.

First Y: Define the first tip position on the head in Y-direction where the tips are mounted.

Last X: Define the last tip position on the head in X-direction where the tips are mounted.

Last Y: Define the last tip position on the head in Y-direction where the tips are mounted.

Tip Box Offset

Row: Enter the offset of the tips in rows on the tip box. You can move from one tip tray to another using this parameter.

Column: Enter the offset of the tips in columns on the tip box. It is possible to move from one tip tray to another using this parameter.

Row/Column Selection (optional): Allows working with variable number of columns/rows of MCA96 tips from a tip-box that has been appropriately reracked (i.e., contains only the tips to be mounted). You can set the direction by selecting Column or Row (selection must be consistent between Pick Up Tips, all Aspirate/ Mix/Dispense commands, and the subsequent Set Tips Back command—this is enforced by context check).

11.1.18 Set Tips Back (MCA96)

Use this command to drop the disposable tips of the MCA96 to a specified position on the worktable.

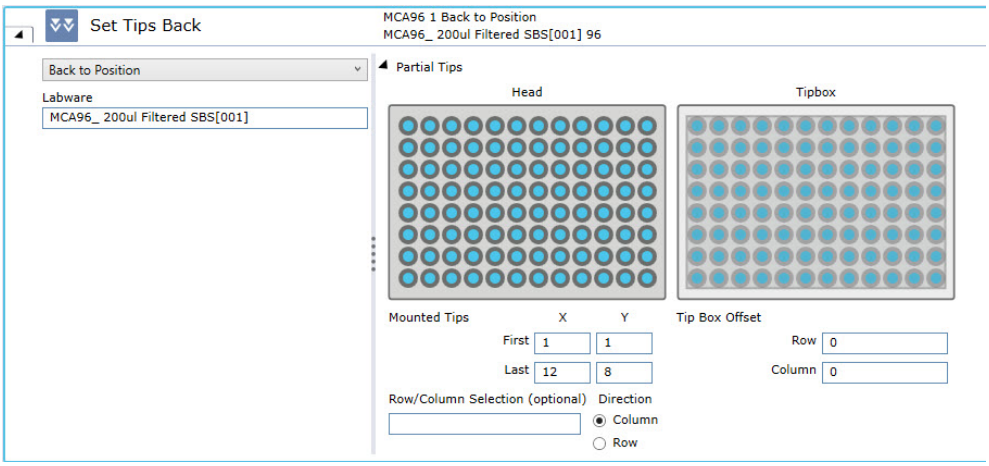


Fig. 141: Set Tips Back

Parameters

Back to Source : This option drops the tips back to the position where they were originally picked up from.

Back to Position : This option drops the tips back to a specified position on the worktable.

Labware: In the Labware box, either type your expression or click on a labware displayed in the Worktable editor window (Worktable click).

Tips mounted

First X: Define the first tip position on the head in X-direction where the tips are mounted.

First Y: Define the first tip position on the head in Y-direction where the tips are mounted.

Last X: Define the last tip position on the head in X-direction where the tips are mounted.

Last Y: Define the last tip position on the head in Y-direction where the tips are mounted.

Tip Box Offset

Row: Enter the offset of the tips in rows on the tip box. You can move from one tip tray to another using this parameter.

Column: Enter the offset of the tips in columns on the tip box. It is possible to move from one tip tray to another using this parameter.

Row/Column Selection (optional): Allows working with variable number of columns/rows of MCA96 tips from a tip-box that has been appropriately reracked (i.e., contains only the tips to be mounted). You can set the direction by selecting Column or Row (selection must be consistent between Pick Up Tips, all Aspirate/ Mix/Dispense commands, and the subsequent Set Tips Back command—this is enforced by context check).

11.1.19 Empty Tip Contents (MCA96)

Use this command to dispense and liquid currently in the tips to a specified position on the worktable.

Parameters

Labware: In the Labware box, either type your expression or click on a labware displayed in the Worktable editor window (Worktable Click).

Labware Offset: The labware offset parameters describe the offset of the mounted tips relative to the labware:

- **Row:** ExpressionInputBox which describes the offset of the mounted tips in rows relative to the labware. It is possible to move from one plate to another using this parameter.
- **Column:** ExpressionInputBox which describes the offset of the mounted tips in columns relative to the labware. It is possible to move from one plate to another using this parameter.

Liquid Class: Choose the required liquid class from the dropdown list or by means of an expression typed in (e.g., by using a variable defined within your script). The list only shows liquid classes that are suitable for the current configuration of the pipetting instrument.

Advanced Parameters

Compartment: Some special labware types (e.g., for protein crystallography) have wells with several compartments (cavities). FluentControl can pipette a different liquid into each of the compartments. Each well typically has one main compartment and one or more satellite compartments which are directly adjacent

and are often smaller in size. Choose the compartment which you want to use for pipetting (the main compartment is normally compartment 1). Well compartments are specified in the labware configuration.

Tips mounted

First X: Define the first tip position on the head in X-direction where the tips are mounted.

First Y: Define the first tip position on the head in Y-direction where the tips are mounted.

Last X: Define the last tip position on the head in X-direction where the tips are mounted.

Last Y: Define the last tip position on the head in Y-direction where the tips are mounted.

Tip Box Offset

Row: Enter the offset of the tips in rows on the tip box. You can move from one tip tray to another using this parameter.

Column: Enter the offset of the tips in columns on the tip box. It is possible to move from one tip tray to another using this parameter.

Row/Column Selection (optional): Allows working with variable number of columns/rows of MCA96 tips from a tip-box that has been appropriately reracked (i.e., contains only the tips to be mounted). You can set the direction by selecting Column or Row (selection must be consistent between Pick Up Tips, all Aspirate/ Mix/Dispense commands, and the subsequent Set Tips Back command—this is enforced by context check).

11.1.20 Get Fingers (MCA96)

Use this command to pick up the MCA96 gripper fingers.

Parameters

Gripper Fingers: Select the MCA96 gripper fingers to be picked up from the worktable.

11.1.21 Drop Fingers (MCA96)

Use this command to demount the MCA96 gripper fingers.

Parameters

Back to Source: By selecting this option, the MCA96 gripper fingers are returned to the position from where they were fetched from.

Back to Position: By selection this option, the user can specify a return position for the MCA96 gripper fingers.

Docking Station: Enter an expression or click a suitable finger nest on the worktable

11.1.22 Get Head Adapter (MCA384)

Use this command to pick up a head adapter for the MCA384 from a specified position on the worktable of the pipetting instrument.

DiTi head adapters can be dropped and picked up with DiTis already mounted. If the target adapter is already mounted, the command is skipped.

Parameters

Head Adapter: Select the available labware on the worktable.

Air gap volume [µl]: The volume for the air gap in microliter.

11.1.23 Drop Head Adapter (MCA384)

Use this command to put down the adapter of the multi-channel arm at a specified position on the worktable, and optionally to get another adapter after the drop.

Be sure the labware is empty to receive the adapter.

Back to Source: When the **Back to Source** option is selected, the adapter is returned back to the grid and site position from where they were fetched with the **MCA384 Get Head Adapter** command.

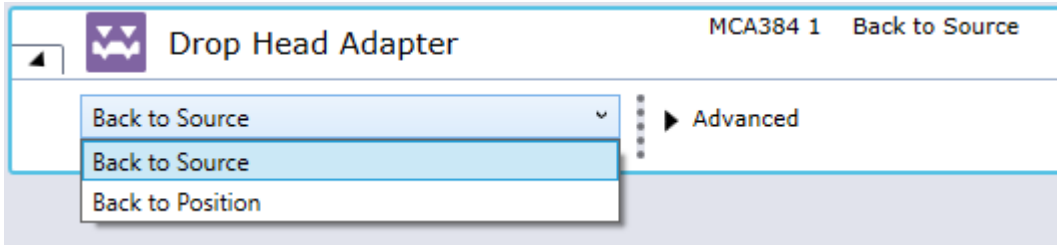


Fig. 142: Back to Source

Back to Position: When the **Back to Position** option is selected, the **Head Adapter box** appears. You can specify the position (labware) where the MCA should return the adapter.

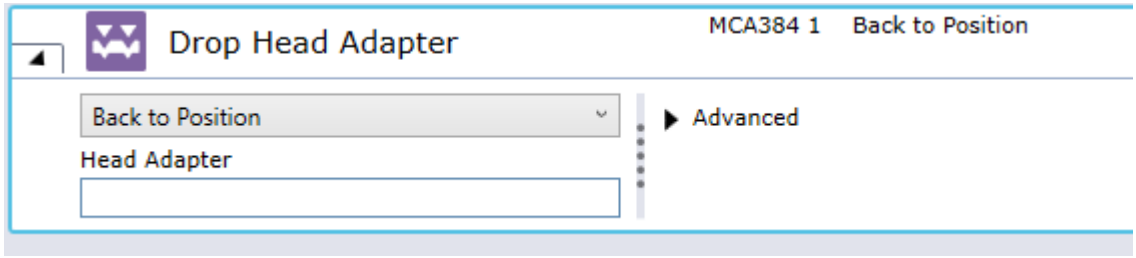


Fig. 143: Back to Position

Advanced: Get Head Adapter After Drop

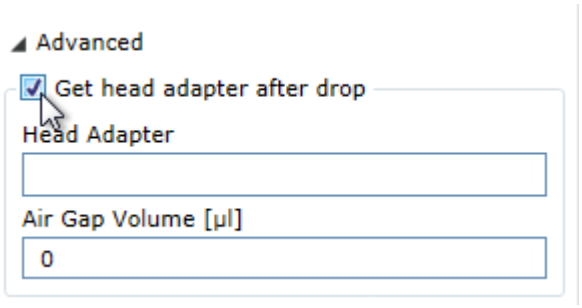


Fig. 144: Get head adapter after drop

Get head adapter after drop: Check this check box if you want to get another adapter after dropping the one which is currently fitted.

Air gap volume [µl]: The volume for the air gap in microliters.

11.1.24 Get Tips (MCA384)

The MCA384 Get Tips command is used to pick up disposable tips for the MCA384 arm with the specified DiTi type on the worktable.

It keeps track of tip positions and automatically picks up the next available and accessible unused tips of the specified type.

Physically inaccessible tip boxes are not considered to pickup tips from.

The order of a list of available tip boxes goes by name in alphabetical order (“MCA96_100ul Filtered_Box[001]”, “MyBox[001]”, “NextBox[001]”), then by index if the names are the same (“MCA96_100ul Filtered_Box[001]”, “MCA96_100ul Filtered_Box[002]”). Physically inaccessible tip boxes are not considered to pickup tips from. Partial tip pickup is supported only for full columns or full rows, not both. For this the First and the Last in X and Y Positions for the channels of the head need to be defined.

If not enough columns or rows of tips are available in one tip box and additional tip boxes with sufficient tips are present on the worktable, the **Get Tips** command automatically executes re-racking of the tips so that the required number of columns or rows can be mounted.

If during the re-racking procedure a partial inaccessibility occurs where tips can't be set back into the target box, no tips will be picked up and an error occurs during runtime. To avoid those errors please validate your application

The re-racking feature is only supported for EVA adapter plates, not for other adapter plates.

Depending on its settings, the **OnEmptyTray** procedure may be called if any empty DiTi rack of the defined type is available.

Parameters

Adapterplate

Select the required adapter plate from the drop-down menu.

Air gap volume [µl]

Defines the volume that is aspirated before mounting any tips.

DiTi Type

Selects the disposable tip type from the dropdown menu or by clicking on the tip box on the worktable.

11.1.25 Drop Tips (MCA384)

Use this command to drop the disposable tips of the multi-channel arm to a specified DiTi Waste.

Adapterplate: Select the appropriate adapter plate by choosing from the drop-down menu or by clicking on the adapter plate on the worktable.

DiTi Waste: Defines the waste where the tips will be discarded.

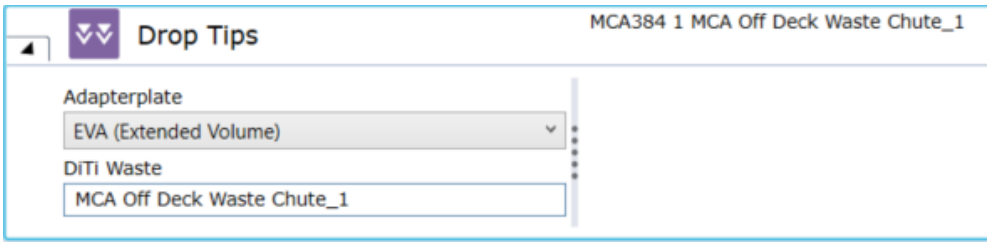


Fig. 145: User interface of 'Drop Tips' command

11.1.26 Pick Up Tips (MCA384)

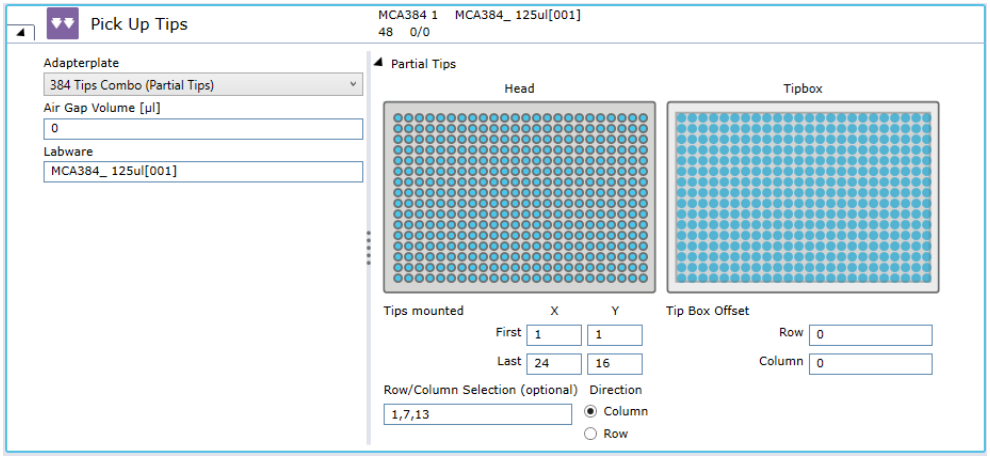


Fig. 146: Pick Up Tips (MCA384)

Use this command to pick up disposable tips for the MCA384 from a specified position on the worktable.

Parameters

Adapterplate: Click the **Adapterplate** arrow to see more options. Select the required adapter plate from the list.

Air gap volume [µl]: The volume for the air gap in microliters.

Labware: In the **Labware** box, either type your expression or click the labware displayed in the **Worktable** editor window (*Worktable click*).

Partial Tips

Tips mounted

First X: Define the first tip position on the head in X-direction where the tips are mounted.

First Y: Define the first tip position on the head in Y-direction where the tips are mounted.

Last X: Define the last tip position on the head in X-direction where the tips are mounted.

Last Y: Define the last tip position on the head in Y-direction where the tips are mounted.

Tip Box offset

Row: Enter the offset of the tips in rows on the tip box. You can move from one tip tray to another using this parameter.

Column: Enter the offset of the tips in columns on the tip box. It is possible to move from one tip tray to another using this parameter.

Row/Column Selection (optional): Allows working with variable number of columns/rows of MCA384 tips for a tip box in the active DiTi carrier with the MCA384 combo adapter from a tip-box that has been appropriately reracked (i.e., contains only the tips to be mounted) outside the active carrier. You can set the **Direction** by selecting **Column** or **Row** (selection must be consistent between Get Tips or Pick Up Tips, all Aspirate/Mix/Dispense commands and the subsequent Drop Tips or Set Tips Back command—this is enforced by context check).

11.1.27 Set Tips Back (MCA384)

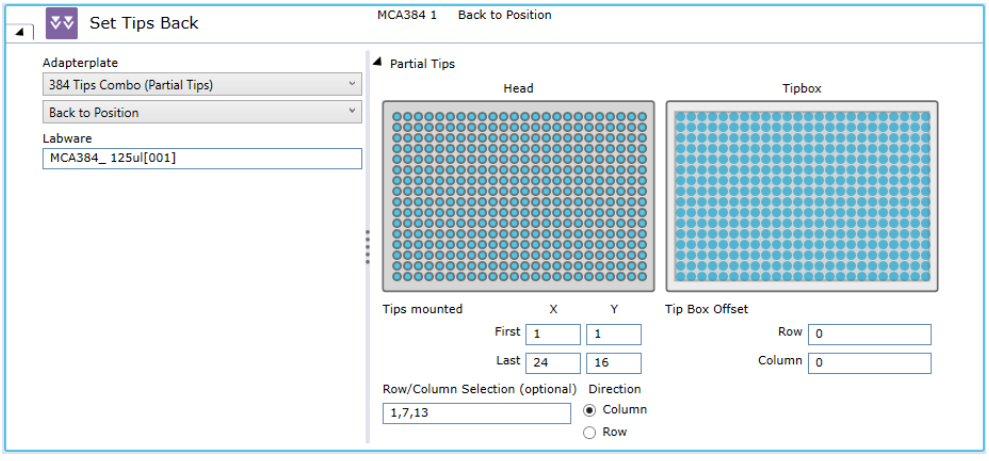


Fig. 147: Set Tips Back

Use this command to drop the disposable tips of the MCA384 to a specified position on the worktable.

Back to Source

Adapterplate: Click the **Adapterplate** arrow to see more options. Select the required adapter plate from the list.

Back to Position

Labware: In the **Labware** box, either type your expression or click the labware displayed in the **Worktable** editor window (*Worktable click*).

Partial Tips

Tips mounted

First X: Define the last tip position on the head in x direction where the tips are mounted.

Last Y: Define the last tip position on the head in y direction where the tips are mounted.

Tip Box offset

Row: Enter the offset of the tips in rows on the tip box. You can move from one tip tray to another using this parameter.

Column: Enter the offset of the tips in columns on the tip box. It is possible to move from one tip tray to another using this parameter.

Row/Column Selection (optional): Allows working with variable number of columns/rows of MCA384 tips for a tip box in the active DiTi carrier with the MCA384 combo adapter from a tip-box that has been appropriately reracked (i.e., contains only the tips to be mounted) outside the active carrier. You can set the **Direction** by selecting **Column** or **Row** (selection must be consistent between Get Tips or Pick Up Tips, all Aspirate/Mix/Dispense commands and the subsequent Drop Tips or Set Tips Back command—this is enforced by context check).

11.1.28 Empty Tip Contents (MCA384)

Use this command to dispense any liquid currently in the tips to a specified position on the worktable.

Parameters

Adapterplate: Click the Adapterplate arrow to see more options. Select the required adapter plate from the list.

Labware Offset: The labware offset parameters describe the offset of the mounted tips relative to the labware:

- **Row**: Enter the offset of the mounted tips in rows relative to the labware. It is possible to move from one plate to another using this parameter.
- **Column**: Enter the offset of the mounted tips in columns relative to the labware. It is possible to move from one plate to another using this parameter. As default a column is selected.

Labware: In the Labware box, either type your expression or click the labware displayed in the **Worktable** editor window (*Worktable click*).

Liquid Class: Choose the required liquid class from the pull down list or by means of an expression typed in, e.g., using a variable defined within your script. The list only shows liquid classes which are suitable for the current configuration of the pipetting instrument. For example, you will not see liquid classes for DiTis if the arm is not currently configured with DiTis.

Advanced Parameters

Compartment: Some special labware types (e.g., for protein crystallography) have wells with several compartments (cavities). can pipette a different liquid into each of the compartments. Each well typically has one main compartment and one or more satellite compartments which are directly adjacent and are often smaller in size. Choose the compartment which you want to use for pipetting (the main compartment is normally compartment 1). Well compartments are specified in the labware configuration. This field is inactive (gray) if the well only has one compartment.

Tips mounted on Head

First X: Define the first tip position on the head in x direction where the tips are mounted.

First Y: Define the first tip position on the head in y direction where the tips are mounted.

Last X: Define the last tip position on the head in x direction where the tips are mounted.

Last Y: Define the last tip position on the head in y direction where the tips are mounted.

X offset [mm]: This is the offset of the tip from the center of the target well (negative is left).

Y offset [mm]: This is the offset of the tip from the center of the target well (Negative = to front, positive = to back).

11.1.29 Prime Wash Station (MCA384)

Use this command to make the wash station ready for use. The command causes the system to fill up again, i.e., it replenishes the wash liquid station completely.

Parameters

Wash Station: Select the wash station.

Duration [s]: Define the duration of a process in seconds.

Wash Liquid Selection: Select the wash liquid.

11.1.30 Empty Wash Station (MCA384)

Use this command to remove the wash liquid of the wash station. The pump is running as long as entered in the **Duration** parameter.

Parameters

Wash Station: Select the wash station.

Duration [s]: Define the duration of the wash process in seconds.

11.1.31 Set Available Tips (Worktable)

Use the **Set Available Tips** command to set the next available position to fetch disposable tips from.

1. First select the labware containing the DiTis (see Labware parameter below).
2. Use the following wildcard characters while typing in the Labware text box:

   ** denoting an arbitrary sequence of characters*

   ? denoting one arbitrary character

3. Select the next available DiTi position in the labware.
4. Mark the positions of the next available DiTi in the plan view of the labware layout.

   FluentControl will update the next available DiTi position for all labware items matching the labware name defined by the user.

Parameters

Labware: In the **Labware** box, either type your expression or click the labware displayed in the Worktable editor window (*Worktable click*).

11.2 FCA

11.2.1 Sample Transfer (FCA)

The 8-tip liquid handling arm driver provides script commands to distribute reagents and transfer samples including wash or get and drop tips.

Instead of a series of **Aspirate** and **Dispense** commands, you can use this smart command to program a sample transfer based on the number of samples, which may be contained in multiple source labware, to destination labware.

A common workflow in the lab is to transfer samples from many tubes to one or more plates. With the **Sample Transfer** smart command, you have an easy and performant way to program a sample transfer based on the number of samples, which may be contained in multiple source labware, to destination labware. In the destination labware you are able to define where the samples may be pipetted, within this range the samples may be pipetted row-wise or column-wise. Optionally, you can define the number of replicates and if they should be pipetted row-wise, column-wise or plate-wise.

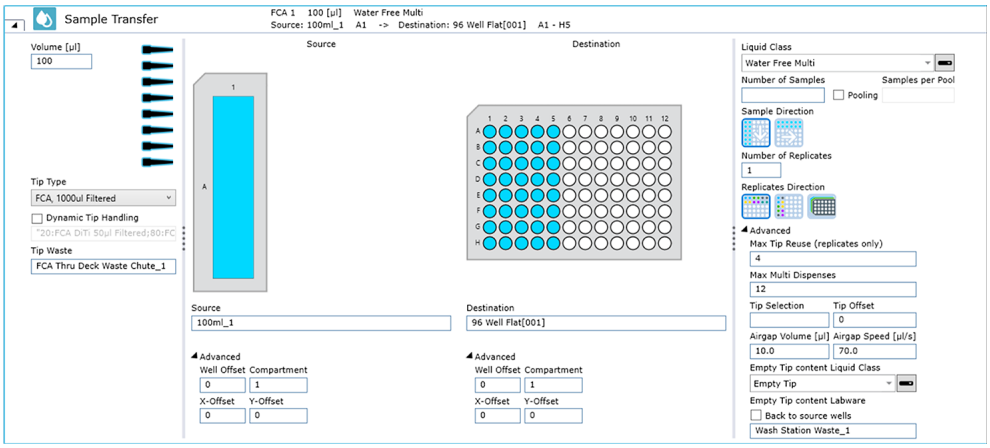


Fig. 148: Transfer Pooling

Pooling: When checking the Pooling option, the number of Samples per Pool has to be entered. The software will then pipette contiguous groups of samples to one pool, i.e. the first group of n samples goes to the first pool (first selected target well). If the total number of samples is not a multiple of the number of samples per pool, the software will issue a context check warning that the last pool contains fewer samples.
When selecting the Pooling option, replicates and tip reuse are not supported and the corresponding input fields are greyed out.

Samples per pool: enter the number of samples per pool.

In addition to a standard Sample Transfer workflow with optional replicates, the command also supports a pooling workflow, i.e., combining a number of samples in one pool.

Defining the pipetting steps for a sample transfer

You can define the pipetting steps for a sample transfer from a source to destination labware. With the **Sample Transfer** command, you can configure the following for the pipetting automation (see script parameters for more details):

- Orientation of samples
- Orientation of replicates (row-wise, column-wise, plate-wise)
- Number of sample replicates
- Orientation of replicates on destination labware
- Selection of which part of the labware will be used (which wells, which compartments)

Selecting the Tip Type (and Tip Waste)

Different parameters are available depending on whether the FCA has Fixed or Disposable tips. For Fixed tip configurations, wash cycles may be defined. For Disposable tip configurations, Tip types and the waste location may be defined.

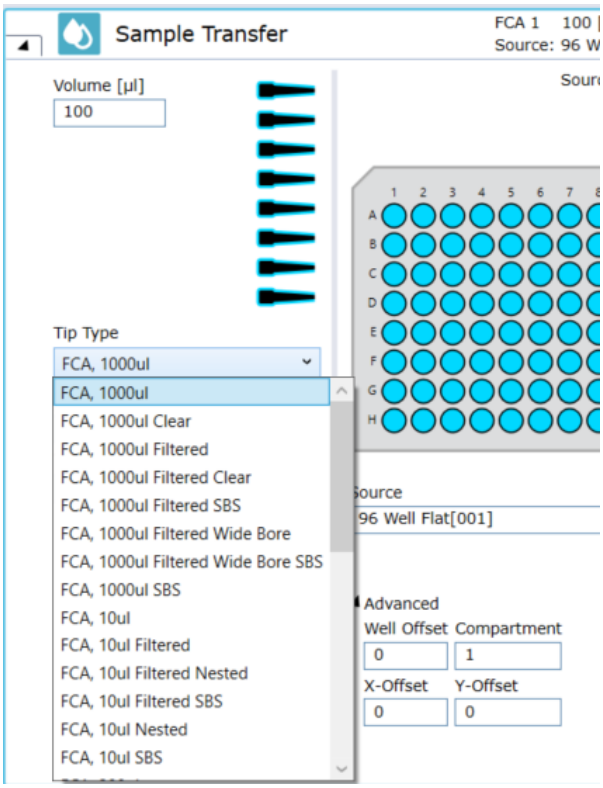


Fig. 149: Select valid Tip Type

Clicking the arrow opens the list.

Tip Waste: If the tips need to be changed (e.g., between samples), the Tip Waste is the position which will be used to discard the tips.

Parameters

The **Sample Transfer** script command provides the following parameters for the configuration of the pipetting automation:

- **Volume [µl]**: Type an expression to specify the volume to be transferred to each target well. If this volume is larger than the available tip capacity or the syringe volume, the system will transfer the volume in multiple equal steps at run-time.
- **Tip selection**: Selects which pipetting channels of the modules are used, optionally a tip mask might be applied.
- **Tip Type**: As described in the Selecting the Tip Type section above.
- **Dynamic Tip Handling:** Dynamic Tip Handling is designed to use different tip types depending on the aspirate volume. If you enter in the Dynamic Tip Handling box – e.g.:

  ```
  30:FCA DiTi 50µl;120:FCA DiTi 200µl
  ```

  this means that:
  - DiTi 50 µl are used for volumes up to 30 µl,
  - DiTi 200 µl are used for volumes > 30 to 120 µl, and
  - for larger volumes the tip type selected in the Tip Type dropdown menu is used.
- **Aspiration position**: Selection from where aspiration is performed by definition of a source labware on the worktable, well selection, well offset, a general X/Y offset within the well and the compartment number.
- **Dispense position**: Selection to where dispensation is performed by selection of the destination labware on the worktable, well selection, well offset, a general X/Y offset within the well and the compartment number.
- Liquid Class
- **Number of Samples**: Number of target wells to be pipetted.
- **Sample Direction**: Defines if the sample should be dispensed with each subsequent sample in the next available well in the row or column. Specify whether you want column-wise or row-wise direction.
- **Number of Replicates**: Default is 1.
- **Replicates Direction**: Defines if replication direction is dispensed in the next available well in the row/column or into the same positions of subsequent plates (plate-wise):



Fig. 150: row-wise



Fig. 151: column-wise or



Fig. 152: plate-wise

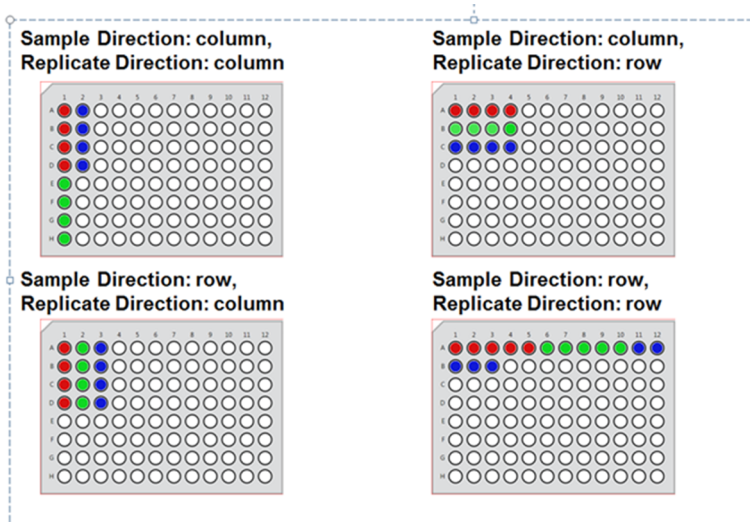


Fig. 153: Sample direction and replicates direction (row-wise and column-wise and their possible directions)

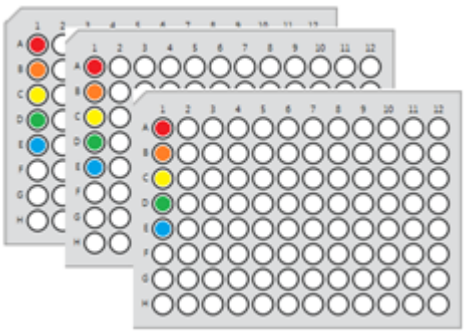


Fig. 154: Sample Direction: column-wise, Replicate Direction: plate-wise (Number of Replicates: 3)

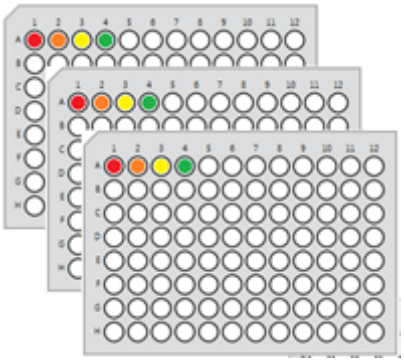


Fig. 155: Sample Direction: row-wise, Replicate Direction: plate-wise (Number of Replicates: 3)

If Replicates Direction row or column is selected, a replicate group (all replicates of a single sample) is kept within a contiguous range and pipetted into the same plate.

Max Multi Dispenses: Maximum number of multi-dispenses

Disposable tips:

- Max DiTi Reuse Maximum number of allowed reuses for an individual disposable tip (i.e., the maximum number of aspirations allowed to be performed with a disposable tip)
- Tip Type to use
- DiTi Waste
- Airgap volume and speed

Fixed tips:

- Wash liquid class
- Waste labware
- Cleaner labware

Example plate map (gaps for controls)

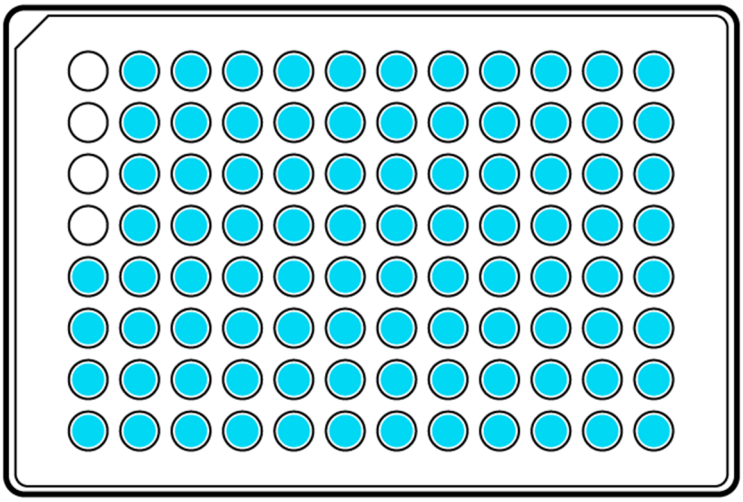


Fig. 156: Selection range leaving 4 wells empty for controls

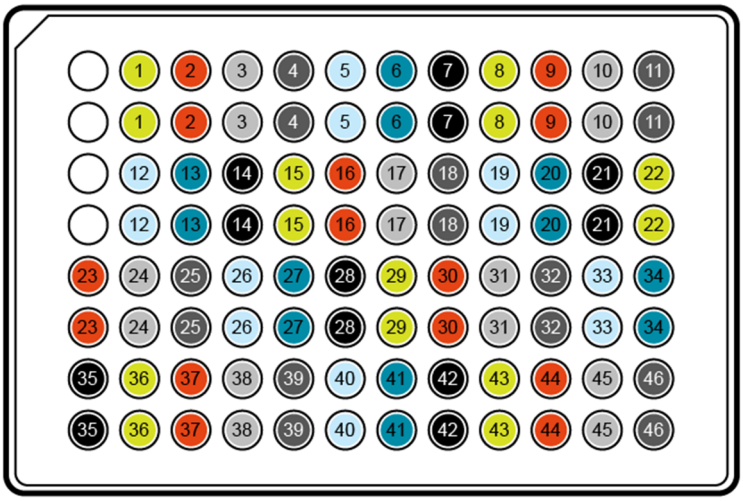


Fig. 157: Samples row-wise, 2 Replicates column-wise and 4 empty wells for controls

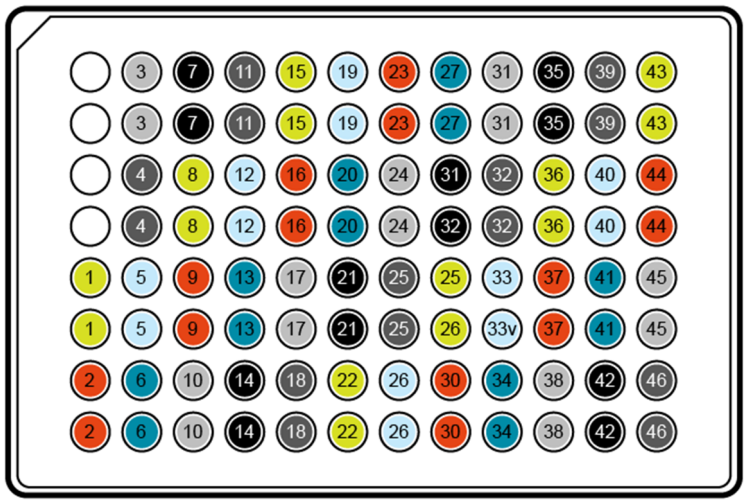


Fig. 158: Samples column-wise, 2 Replicates column-wise and 4 empty wells for controls

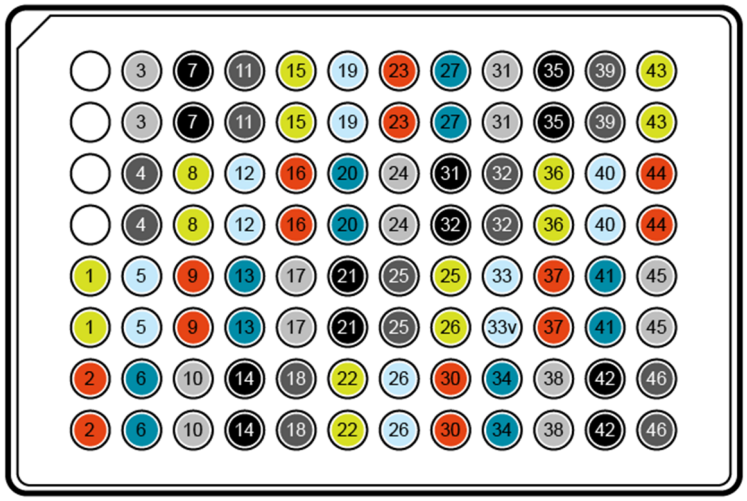


Fig. 159: Samples row-wise, 2 Replicates row-wise and 4 empty wells for controls

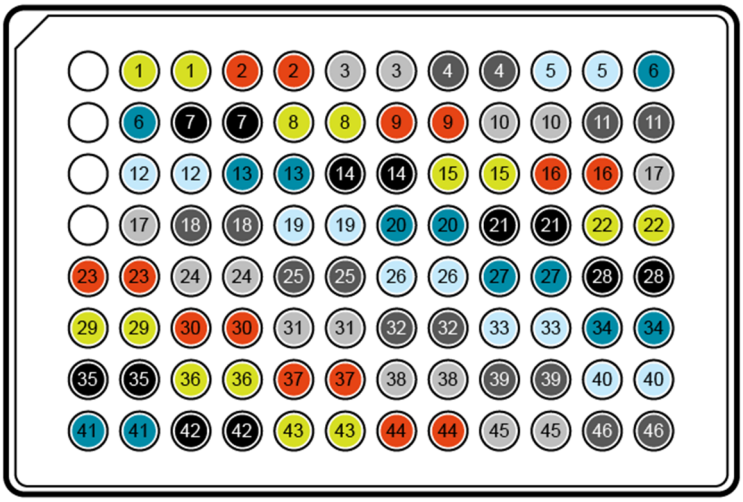


Fig. 160: Samples row-wise, 2 Replicates row-wise and 4 empty wells for controls

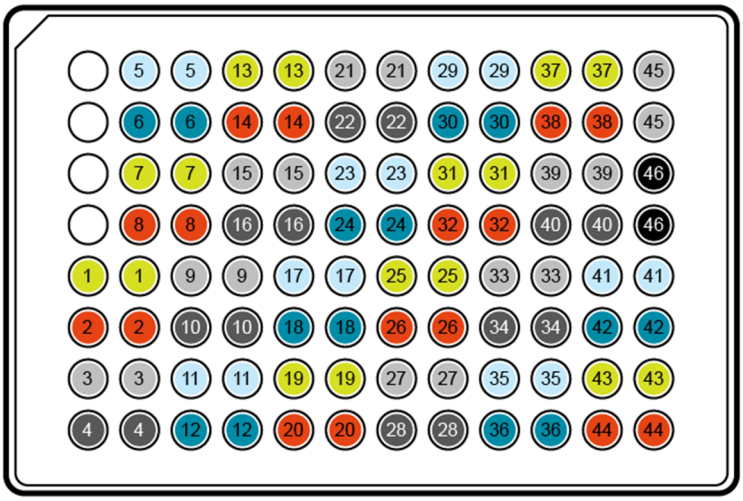


Fig. 161: Samples column-wise, 2 Replicates row-wise and 4 empty wells for controls

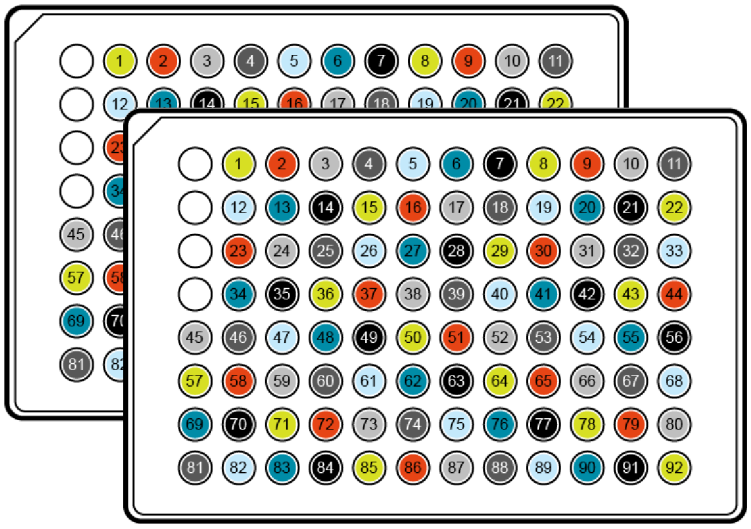


Fig. 162: Samples row-wise, 2 Replicates plate-wise and 4 empty wells for controls

Advanced Parameters

Max Tip Reuse: Maximum number of allowed reuses (i.e. number of aspirations) allowed to be performed with an individual disposable tip.

Max Multi Dispenses: Maximum number of multi-dispenses.

Tip Selection: Optional tip mask to be applied to the tip selection in the UI.

Tip Offset: Optionally shifts the tip selection in the UI.

Airgap volume / Airgap speed: Volume and aspiration speed of blowout airgap, aspirated when picking up disposable tips.

Empty Tip Content Liquid Class: Liquid class to use when performing an Empty Tips (used in multi pipetting sequences to discard the excess volume).

Empty Tip Content Labware: Labware to use when performing an Empty Tips, usually a liquid waste.

Back to source wells: If this option is selected, the tip contents are emptied in the last source that those tips aspirated from. The benefit of this is to save valuable liquids remaining in a tip after a dispense.

11.2.2 Reagent Distribution (FCA)

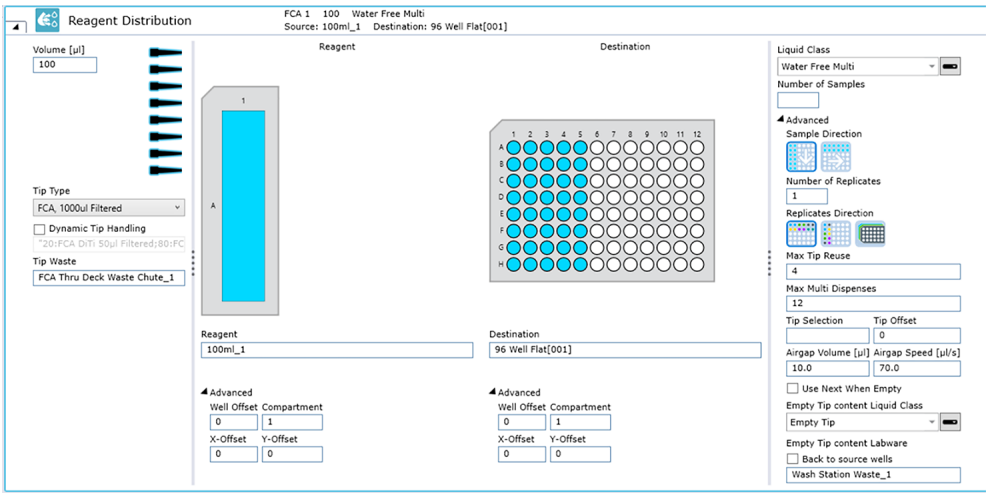


Fig. 163: Reagent Distribution

With the Reagent Distribution script command, you can quickly program a pipetting sequence that transfers liquid (reagent) from a single source across one or multiple plate(s) based on a plate map (gaps for controls) and a number of samples.

Parameters

The Reagent Distribution script command provides the following parameters for the configuration of the pipetting automation:

Volume [µl]: Specifies the volume to be transferred to each target well. If this volume is larger than the available tip capacity or the syringe volume, the system will transfer the volume in multiple equal steps at runtime.

Tip selection: Selects which pipetting channels of the modules are used, optionally a tip mask might be applied.

Tip Type: Select the type of disposable tips to be used for pipetting from the dropdown list.

Dynamic Tip Handling: Dynamic Tip Handling is designed to use different tip types depending on the aspirate volume. If you enter in the Dynamic Tip Handling box – e.g.:

```
30:FCA DiTi 50µl;120:FCA DiTi 200µl
```

this means that: -DiTi 50 µl are used for volumes up to 30 µl, -DiTi 200 µl are used for volumes > 30 to 120 µl, and - for larger volumes the tip type selected in the Tip Type dropdown menu is used.

Tip Waste: If the tips need to be changed (e.g., between samples), the Tip Waste is the position which will be used to discard the tips.

Aspiration position: Selection from where aspiration is performed by definition of a source labware on the worktable, well selection and optionally (Advanced section below the labware) well offset, a general X/Y offset within the well and the compartment number.

Dispense position: Selection to where dispensation is performed by selection of the destination labware on the worktable, well selection and optionally (Advanced section), well offset, a general X/Y offset within the well and the compartment number.

Liquid Class: Select the liquid class to be used for aspirate and dispense.

Number of Samples: Number of target wells to be pipetted. This parameter is optional; if left empty, the number of samples corresponds to the number of selected source wells.

Advanced Parameters

Sample Direction: Defines if the sample should be dispensed with each subsequent sample in the next available well in the row or column. Specify whether you want column-wise or row-wise direction.

Number of Replicates: Allows replication of the selected samples. Default is 1.

Replicates Direction: Defines if replication direction is dispensed in the next available well in the row/column or into the same positions of subsequent plates (plate-wise).

Max Multi Dispenses: Maximum number of multi-dispenses.

Max Tip Reuse: Maximum number of allowed reuses (i.e. number of aspirations) allowed to be performed with an individual disposable tip.

Tip Selection: Optional tip mask to be applied to the tip selection in the UI.

Tip Offset: Optionally shifts the tip selection in the UI.

Airgap volume / Airgap speed: Volume and aspiration speed of blowout airgap, aspirated when picking up disposable tips.

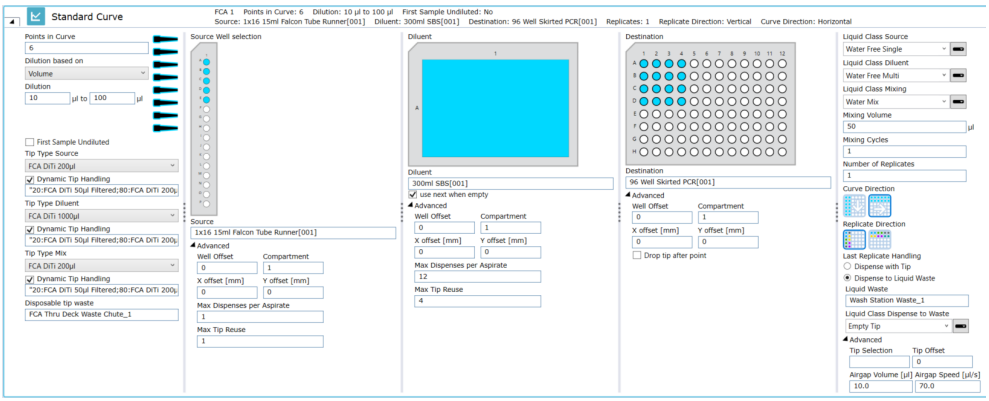
Use Next When Empty: If this option is checked, the system will automatically proceed to the next source container in the same labware group if the first one has been emptied.

Empty Tip Content Liquid Class: Liquid class to use when performing an Empty Tips (used in multi pipetting sequences to discard the excess volume).

Empty Tip Content Labware: Labware to use when performing an Empty Tips, usually a liquid waste.

Back to source wells: If this option is selected, the tip contents are emptied in the last source that those tips aspirated from. The benefit of this is to save valuable liquids remaining in a tip after a dispense. If **Use Next When Empty** is selected as well, the tips empty their contents into the last well that was aspirated from.

11.2.3 Standard Curve (FCA)



You can quickly program a pipetting sequence that pipettes a standard curve from a single source across a plate based on a plate map (gaps for samples), number of points in the curve and dilution options.

A common workflow in the lab is that it is necessary to know the exact concentration of the liquids that are handled. Especially before normalization the concentration of the individual liquids must be known.

A standard curve is used in quantitative research as reference to identify unknown concentrations of substances (e.g., DNA, protein). Multiple samples with known properties are measured and graphed, which then allows the same properties to be determined for unknown samples by interpolation on the graph.

The script command allows configuration of the following for the pipetting automation:

1. Define the curve points and dilution ratios
2. Define one source well
3. Define diluent location
4. Define target wells
5. Define target layout
6. Define pipetting parameters as liquid classes and tip types

The system allows selecting labware by clicking on the worktable:

- Define the source labware (Source)
- Define the diluent labware (Diluent)
- Define the destination labware (Destination)
- Click on a DiTi to define the TipType (DiTi Type)
- Click on a Waste to define the Waste (DiTi Waste)

Points in Curve: Specify the amount serial dilution steps of the curve.

Dilution based on: Specify the ratios that count for each point inside the curve.

- *Volume*
  Specify the exact volumes for the source and the diluent. The diluent volume will be the target volume.

- *Factor*
  Specify the dilution factor—e.g.: 1:3 means 1 part sample and 2 parts diluent. Additionally, the target volume has to be specified. Target volume will be the volume of the dilution.

Tip selection (click): Selects which pipetting channels of the modules are used.

Tip Type Selection (Source, Diluent, Mix): Specify the tip types that shall be used for pipetting.

Dynamic Tip Handling: Dynamic Tip Handling is designed to use different tip types depending on the aspirate volume. If you enter in the Dynamic Tip Handling box – e.g.:

```
30:FCA DiTi 50µl;120:FCA DiTi 200µl
```

this means that:
- DiTi 50 µl are used for volumes up to 30 µl,
- DiTi 200 µl are used for volumes > 30 to 120 µl, and
- for larger volumes the tip type selected in the Tip Type dropdown menu is used.

Disposable Tip waste: Specify the name of the Disposable Tip waste on the worktable which can be added via a worktable click.

Source well selection

Selection from where source aspiration is performed by definition of a source labware on the worktable, well selection, well offset, a general X/Y offset within the well, compartment number, max. dispenses per aspirates and max tip reuse.

Diluent well selection

Selection from where diluent aspiration is performed by definition of a source labware on the worktable, well selection, well offset, a general X/Y offset within the well compartment number, max. dispenses per aspirates and max tip reuse.

Use Next When Empty: If selected, the system uses the next available labware within the same group if the actual labware is empty.

Destination well selection

Selection to where dispensation is performed by selection of the destination labware on the worktable, well selection, well offset, a general X/Y offset within the well and the compartment number. This option is not available for In Situ.

Sample direction: Defines if samples are dispensed into the target labware in the next available well in the row-wise or column-wise. This option is not available for In-Situ.

Replicate direction: Defines if replication direction is dispensed in the next available well in the row/column or into the same positions of subsequent plates (plate-wise).

Well Offset: Specify the offset inside a labware (source, diluent or destination). The wells until the offset will stay empty. In multiple plates this counts for the first plate. Counting is done from top to bottom, then from left to right.

Compartment: Defines the compartment within a well to be used.

X offset [mm]: X-Pipetting offset counted from the middle of the well. Allows pipetting next to the well border.

Y offset [mm]: Y-Pipetting offset counted from the middle of the well. Allows pipetting next to the well border.

Maximum dispenses per Aspirate: Number of dispenses allowed for pipetting until a reaspirate has to be done. After reaspiration or DiTi change this number starts counting from 0.

Maximum Tip reuse: Maximum number of aspirations allowed to be performed with an individual disposable tip.

Tip Selection: Tip mask to define which pipetting channels are used.

Tip Offset: Offset for tips stating which tip shall be used first. The tips until the offset will not be used for pipetting.

11.2.4 Transfer Individual Volumes (FCA)

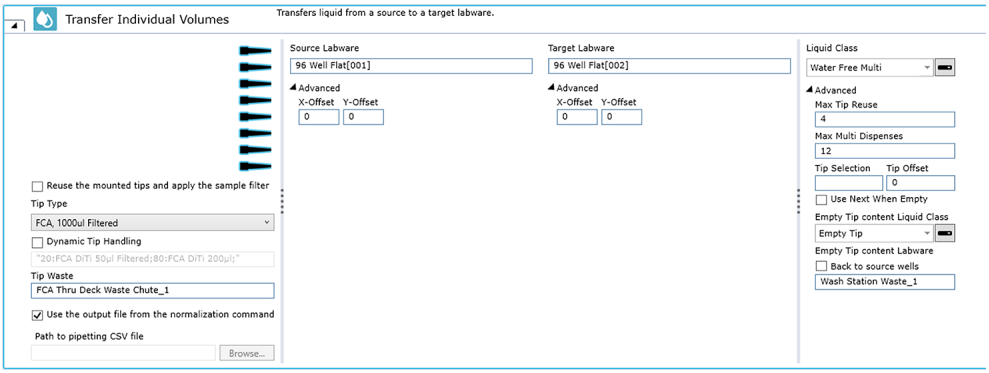


Fig. 164: Transfer Individual Volumes

The Transfer Individual Volumes command can be used to transfer different volumes from one labware to another. The Input of this command is a pipetting instruction file in the form of a CSV file, the separator is a semicolon (;). The user can create their own pipetting instruction file or use the generated pipetting instructions of the Normalization Calculation command. For more information on the Normalization Calculation command, refer to Normalization Calculation. The command will create a subroutine, which contains the optimal pipetting sequence to process the pipetting instruction input file.

Example

Source well index	Target well index	Volume [µl]
1	1	20.000
2	2	22.000

The raw text file of this example would then be:

Source well index;Target well index;Volume [ul]
1;1;20.000

2;2;22.000

The header must be included in the file and must contain these three columns.

Parameters

Tip Selection: Select the tips you want to use. By default the software uses all tips. To turn off or turn on a tip, click on it.

Reuse the mounted Tips and apply the sample filter: Selecting this checkbox will only process source wells from which currently mounted tips have aspirated.

Tip Type: This selection will define what type of disposable tip shall be used during the generated subroutine.

Dynamic Tip handling: The dynamic Tip handling is designed to use different Tip types depending on the aspirate volume, if these are available on the worktable. An example input of this field could be:

```
30:FCA DiTi 50µl; 120:FCA DiTi 200µl;
```

This would mean that:

- DiTi 50 µl are used for volumes up to 30 µl
- DiTi 200 µl are used for volumes > 30 µl to 120 µl

Tip Waste: This field defines where the used tips shall be disposed to.

Use the output from the normalization Command: Enabling this checkbox will use the pipetting instruction files, which were generated in the (In Situ) Normalization Calculation command, to execute the transfer of volumes. These files are protected using a hash value, editing them will cause the script execution to fail.

Path to Pipetting CSV File: If the output from the normalization command shall not be used, or you wish to edit the pipetting instruction files before the transfer of volumes, a CSV file can be selected on the computer and used to transfer the liquids.

Source Labware: This field contains the name of the labware that is used to aspirate the liquid from. If the output from the Normalization command is used, this must be the same labware as the Sample or Diluent labware, depending on which you wish to transfer. Either type your expression or click the labware displayed in the Worktable editor window (Worktable Click).

Target Labware: This field contains the name of the labware which is used to dispense the liquid to. Either type your expression or click the labware displayed in the Worktable editor window (Worktable Click).

Liquid Class: Choose the required liquid class from the dropdown list or by means of an expression typed in (e.g., by using a variable defined within your script). The list only shows liquid classes that are suitable for the current configuration of the pipetting instrument.

Advanced Parameters: The Advanced parameters contain the same functionality as the Sample Transfer and Reagent Distribution Smart Commands. For more information on these, refer to “Sample Transfer (FCA)” [▶ 226] or “Reagent Distribution (FCA)” [▶ 235].

11.2.5 Transfer Phase (FCA)

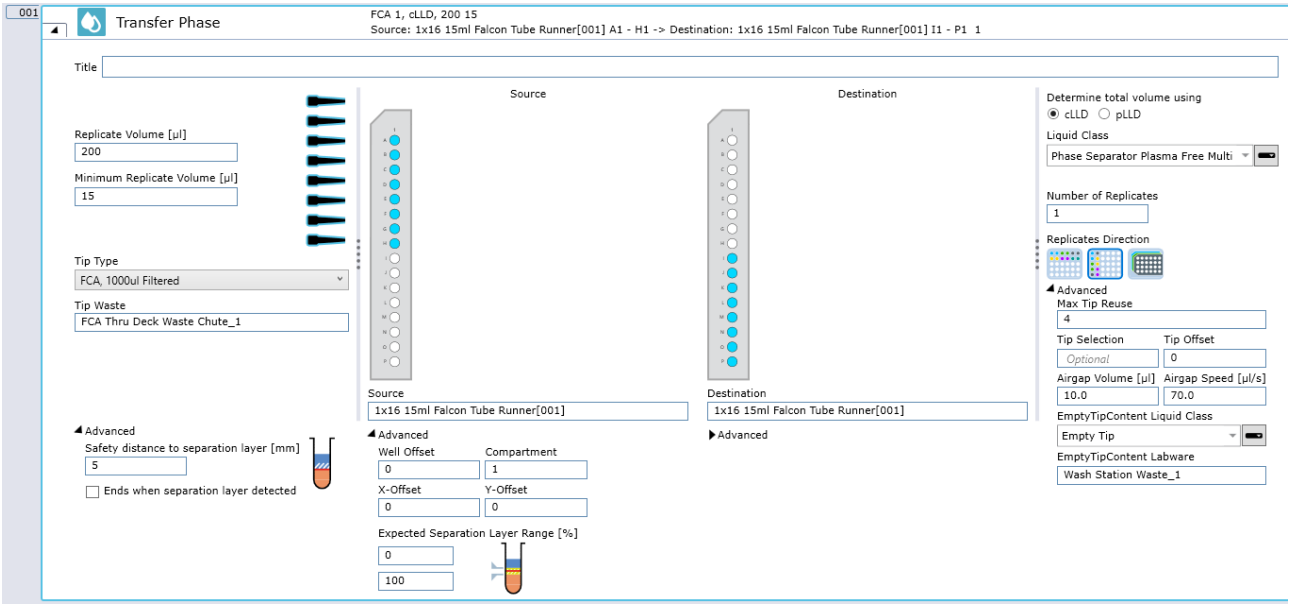


Fig. 165: Transfer Phase HLC Pipetting Command

The Transfer Phase command is designed for the MultiSense AirFCA and is designed to separate the upper liquid phase from the lower liquid phase in samples with a liquid-liquid interface. The goal of the workflow is to transfer the upper phase from the source to the target labware.

Transfer Phase automates the handling of tips. Any tips already present on the AirFCA are automatically discarded before Transfer Phase begins, and the command will start the procedure by mounting a new set of tips of the selected tip type. If the number of aspirations exceeds the **Max Tip Reuse** parameter set in the Advanced section, Transfer Phase will automatically discard tips and mount new tips as needed. Once the command is completed, all tips will be dropped.

At execution, Transfer Phase first determines the total sample volume by detecting the initial liquid level in the source labware. If the recommended liquid class is selected, a spillover prevention procedure, defined in the liquid class, is performed in case the liquid level is too high to allow the tip submergence needed for separation layer detection. Next, the upper phase layer is transferred from source to destination by iterative pipetting sequences of aspirate, dispense (or a series of dispenses), and empty tip. During the liquid transfer, the individual aspiration steps will attempt phase detection until the phase separation layer is identified. The parameters of the aspirations performed are defined in the liquid class and may vary depending on the liquid class used.

The method for calculating the destination indices (or labware with indices) is the same as for the Sample Transfer smart command (“Sample Transfer (FCA)” [▶ 226]).

The empty tip step is only executed for multi-pipetting liquid classes. When multi-pipetting, an additional excess liquid volume is aspirated, which remains in the tips after all dispenses are completed. This remaining volume is dispensed into the labware defined in **EmptyTipContent Labware**, using the liquid class defined in **EmptyTipContent Liquid Class**. If the source labware is defined as the **EmptyTipContent Labware**, the liquid is dispensed into the same wells or tubes

from which it was originally aspirated. If the destination labware is defined as the **EmptyTipContent Labware**, the liquid is dispensed into the wells or tubes used for the last regular dispense.

The user has the possibility to either transfer a distinct number of replicates (sometimes also referred to as “aliquots”) or to extract as much upper liquid as possible.

To transfer a distinct number of replicates, the user must enter a **Replicate Volume** as well as the desired number of replicates.

To transfer as much liquid as possible, the user must type the keyword **MaxVolPhaseSeparator** in the entry field of **Replicate Volume**. Note: This functionality is only compatible with a replicate count of one, corresponding to a direct liquid transfer from one source to one single target.

The information about replicates will be stored as well attribute of the source labware with the dedicated name TransferredReplicateCountForTransferPhase and can be retrieved by using the GetAttribute() functionality. The default value of the attribute is zero.

11.2.5.1 Parameters

Transfer Phase provides the following parameters for the configuration of the pipetting automation:

- **Replicate Volume**: the volume of liquid that is required in the destination well or tube per replicate. For the transferring of all available upper phase liquid to the destination, the keyword **MaxVolPhaseSeparator** shall be entered in the Replicate Volume field.

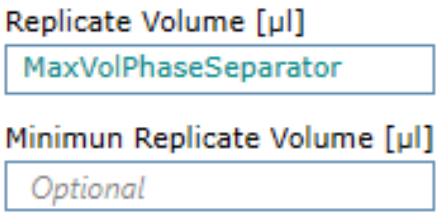


Fig. 166: Replicate volume with MaxVolPhaseSeparator

- **Minimum Replicate Volume:** the minimum volume required in the destination well per replicate. Setting this parameter is optional.
- **Tip Selection:** Selects which pipetting channels of the AirFCA are used.
- **Tip type:** select the type of tips to be used for pipetting from the dropdown list or by clicking on the labware in the Worktable Editor.
- **Tip waste:** If the tips need to be changed (e.g., between samples), the Tip Waste is the position which will be used to discard the tips.
- **Safety distance to separation layer:** the distance in mm from detected phase separation layer to which the tip orifice is allowed to travel. When a separation layer is detected, Transfer Phase will not aspirate any more liquid below the safety distance towards the separation layer level, leaving a certain dead volume behind For example, a value of 5 would mean that the tip is not allowed to go closer to the separation layer than 5 mm, and the corresponding volume below 5 mm would be left in the source tube. This prevents accidental

aspiration or contamination from the lower liquid fraction in unstable samples (e.g. centrifuged whole blood). It is also possible to enter a negative value, to aspirate liquid down to a level below the detected height.

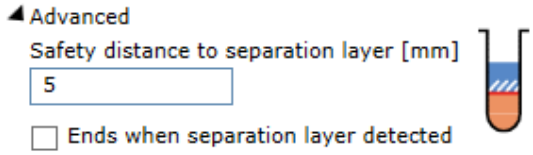


Fig. 167: Safety Distance to Separation Layer

- **Ends when separation layer detected:** if the checkbox is checked, no further liquid is transferred as soon as the separation layer is detected.
- **Source:** Defines the source labware on the worktable.
- **Well Offset:** Specify the offset inside a labware. The wells until the offset will stay empty. In multiple plates this counts for the first plate. Counting is done from top to bottom, then from left to right.
- **Compartment:** Defines the compartment within a well to be used.
- **X-Offset:** Defines the well offset in the x direction.
- **Y-Offset:** Defines the well offset in the y direction.
- **Expected Separation Layer Range:** consists of two parameters, "From" and "To," which are expressed as percentages of the initially detected liquid volume. These parameters define the expected location of the phase separation layer and need to be defined with a “top to bottom” traveling direction in mind.
- The **From** parameter indicates the starting point of the expected range.
- The **To** parameter indicates the ending point of the expected range.

Both parameters are optional:

- If the **From** parameter is not specified, it defaults to 0%.
- If the **To** parameter is not specified, it defaults to 100%.

The Transfer Phase command will not aspirate below the "To" parameter, even if the phase separation layer is not detected.

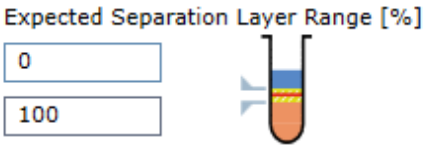


Fig. 168: Expected separation layer range section

- **Destination:** Defines where the dispense is performed by defining the destination labware on the worktable.
- **Well Offset:** Specify the offset inside a labware. The wells until the offset will stay empty. In multiple plates this counts for the first plate. Counting is done from top to bottom, then from left to right.
- **Compartment:** Defines the compartment within a well to be used.

- **X-Offset:** Defines the well offset in the x direction.
- **Y-Offset:** Defines the well offset in the y direction.
- **Determine Total Volume using Selection:** Liquid level detection can be performed using either cLLD (Capacitive Liquid Level Detection) or pLLD (Pressure Liquid Level Detection). The operator can select the preferred detection method. The default detection speed is 60 mm/s for cLLD and 20 mm/s for pLLD.
  The Infopad warning is displayed to alert the operator when the pLLD command is used. With pLLD, liquid level detection is performed only once, and in liquid class the remaining volume is calculated based on the initial pLLD measurement. Operator is informed with warning message “Liquid level detection using pLLD can be less accurate…”.

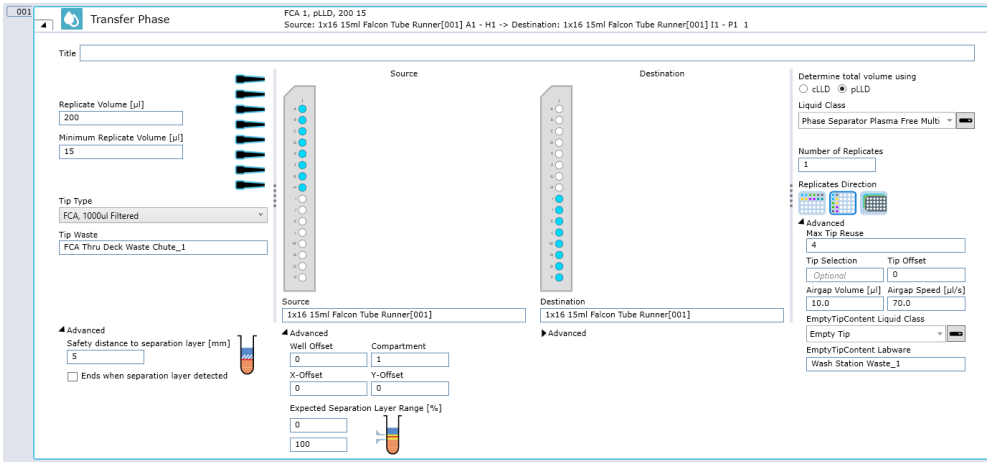


Fig. 169: Determin Total Volume Using Selection



Fig. 170: Infopad Warning

- **Liquid class:** defines the liquid class to be used for the liquid transfers. The defined liquid class must include the phase detection functionality.
- **Number of Replicates:** defines the required number of replicates for each sample. Note: The number of replicates transferred during execution of the command can be impacted by other parameters, e.g., the Minimum Replicate Volume and the liquid available above the detected phase separation layer.
- **Replicates Direction:** Defines if replication direction is dispensed in the next available well in the row/column or into the same positions of subsequent plates (plate-wise).

Advanced Parameters

- **Max Tip Reuse:** Maximum number of allowed reuses for an individual tip (i.e., the maximum number of aspirations) allowed to be performed with a tip.
- **Tip Selection:** The Tip Selection (optional) parameter is resolved as integer (integer variables and expressions are allowed) and interpreted then as a bitmask. Note: The tip selection must be compatible between both the graphical tip selection as well as the optional Tip Selection field in the advanced parameters section. Thus, only tips selected in both masks will be used for pipetting.
  The Tip Selection (optional) mask is specified as a numeric, bit-coded value

(tip1 = 1, tip2 = 2, tip3 = 4, tip4 = 8, tip5 = 16, tip6 = 32, tip7 = 64, tip8 = 128). Use the sum of the bit-coded values to select more than one tip. (e.g. 6 to select tip 2 and 3) In the user interface, tip1 is the topmost tip.

- **Tip Offset:** This parameter lets you modify the tip selection for the pipetting command. The default is 0, which deactivates the offset. A positive value increments the tip number(s) which are specified in the pipetting command. For example, if the pipetting command specifies tips 2 and 3 and you set this parameter to 1, tips 3 and 4 will be used instead. Negative values of the parameter shift the tip selection in the opposite direction.
- **Airgap Volume / Airgap Speed:** Volume and aspiration speed of blowout airgap, aspirated when picking up disposable tips.
- **EmptyTipContent Liquid Class:** Liquid class to use when performing an Empty Tips (used in multi pipetting sequences to discard the excess volume).
- **EmptyTipContent Labware:** Labware to use when performing an Empty Tips, usually a liquid waste.

11.2.5.2 Error Handling

Retry Behavior When No Liquid is Detected:

1. **pLLD (Pressure Liquid Level Detection)**: if no liquid is detected in a channel, the system discards current tips, acquires new tips, and continues the detection process. The following message is displayed:

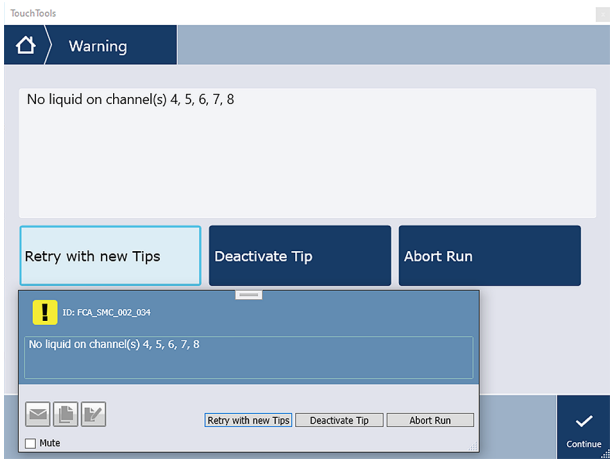


Fig. 171: pLLD no liquid detected

2. **cLLD (Capacitive Liquid Level Detection)**: if no liquid is detected, the system automatically retries only with the channels where liquid was not detected.

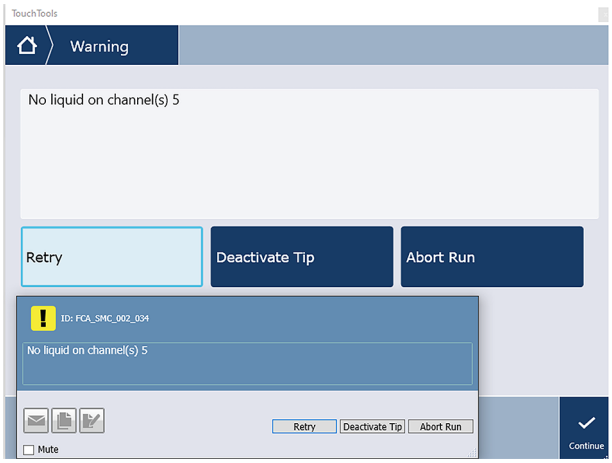


Fig. 172: cLLD no liquid detected

Retry Behavior When Detection is Halted or Aborted (fast stop triggered):

1. **pLLD (Pressure Liquid Level Detection)**: if detection is aborted or halted (fast stop triggered), the system discards all current tips, acquires new tips, and continues the detection process with Transfer Phase.
2. **cLLD (Capacitive Liquid Level Detection)**: If detection is aborted or halted (fast stop triggered), the system continues the detection process with existing tips.

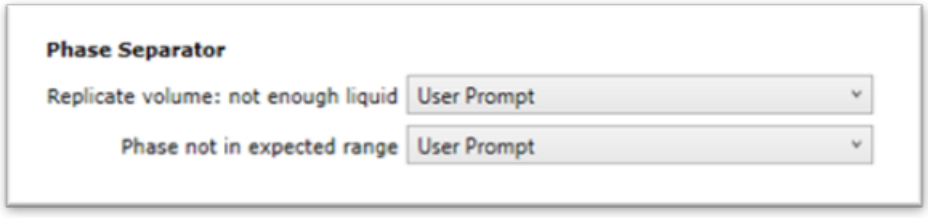


Fig. 173: Transfer Phase Error Handling

Transfer Phase raises an error when not enough liquid is available to achieve the specified replicate volume and number, or if the separation layer is not found within the defined expected range. Automatic error handling settings can be defined in the Detection & Positioning subsection of the Liquid Class.

The following points describe the pipetting behavior of Transfer Phase in specific scenarios:

- A Minimum Replicate Volume is not defined (field is empty). The separation layer is detected, and it is determined that insufficient volume is available to complete a replicate of the defined Replicate Volume. In this scenario, Transfer Phase aspirates as much liquid as possible, respecting the defined safety distance to the separation layer, and dispenses the short replicate to the destination labware. The number of replicates achieved from a specific source well is stored in the well attribute **TransferredReplicateCountForTransferPhase**.

- A Minimum Replicate Volume is defined. The separation layer is detected, and it is determined that insufficient volume is available to complete a replicate of the defined replicate volume. In this scenario, Transfer Phase stops transferring liquid immediately. A ‘not enough liquid’ error is raised when:
  - The total available source volume is less than the total minimum volume required to be transferred (Number of Replicates * Minimum Replicate Volume).
  - The separation layer is found. The liquid volume above the safety distance from the separation layer is less than the total minimum volume (Number of Replicates * Minimum Replicate Volume).
- Whenever Transfer Phase does not detect a separation layer in the source labware, a ‘phase not found in expected range’ error is raised when:
  - The position defined in the field **To** of the Expected Separation Layer Range settings or the bottom of the source labware is reached without detecting a separation layer.
  - If a separation layer is found before the position defined in the field **From** of the Expected Separation Layer Range settings is reached, a ‘phase not in expected range’ error is raised.

The error handling results of each Transfer Phase command execution are stored in the well attribute **TransferPhaseResult** of the source labware and can be retrieved using the GetAttribute() function. The **TransferPhaseResult** well attribute is of type integer with the default value “-1”. The meaning of the values written into the well attribute **TransferPhaseResult** by Transfer Phase are found in the table below.

TransferPhaseResult well attribute value	Error corresponding to the value
-1	Unknown result
0	No errors
1	No liquid detected
2	Not enough liquid detected: total volume is not sufficient (determined after initial liquid level detection)
3	Not enough liquid detected: upper phase volume is not sufficient (determined after aspiration has started and a separation layer has been detected)
4	Phase not found within Expected Separation Layer Range
5	Phase found above Expected Separation Layer Range

11.2.5.3 Liquid Class Requirements

The Transfer Phase command is distributed with special designed liquid class. The liquid class that is designed for the transfer phase script command shall define the following variables that is needed for correct executing of microscript of the liquid class. If one of the variables is not defined in the scope of the LC the corresponding context check error is triggered.

- **endsWhenSeparationLayerDetected** – is an *Integer* type variable. The recommended default value is 0.

- **safetyDistanceToSeparationLayer** – is a *Floating Point* type variable. The recommended default value is 5.
- **topExpectedSeparationLayerHeight** - is a *Floating Point* type variable. The recommended default value is 100.
- **bottomExpectedSeparationLayerHeight** - is a *Floating Point* type variable. The recommended default value is 0.
- **tipDisplacementVolume** - is a *Floating Point* type variable. The recommended default value is 0.
- **maxImmersionDepth** - is a Floating Point type variable. The recommended default value is 0.

11.2.6 Get Tips (FCA)

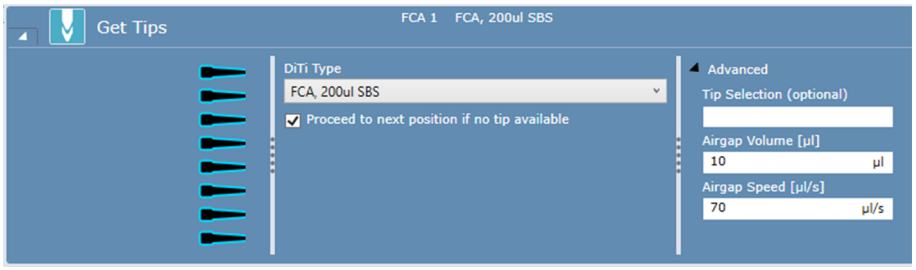


Fig. 174: Get Tips

Use this command together with a FCA (see “FCA” [▷ 226]) to pick up disposable tips (DiTis) of the specified type from a DiTi rack. FluentControl keeps track of their position on the worktable and automatically picks up the next available unused DiTis of the chosen type.

The system tracks available DiTis in the **Get Tips** script command for the FCA arm by automated disposable tip tracking for single and multiple DiTi types. The system performs automated DiTi tracking by automatic tracking of the position in the labware with the next available DiTi. If the FluentControl software has restarted or a new power cycle has started, the system continues automatic tracking of the next DiTi from the previous application or power cycle.

Order of Tip Pickup

If there is more than one tip box containing tips of the selected type, pickup starts at the box with the lowest index number (e.g., FCA_200ul[001] before FCA_200ul[002]). In general tips are picked up from left to right within a box, but preference is given to fast pickup over strict left-to-right use—e.g.: 8 tips will always be picked up from a full column and not in two steps starting from an incomplete column in a tip box.

Pickup of tips from nested FCA tip stacks follows a special logic. Tip stacks in worktable positions outside of common tip range (i.e., those in the very back or front of the worktable) are only used when picking up 8 tips. Those stacks are not used when picking up only partial columns. For stacks of FCA tips within common arm range FluentControl tries to use up one stack completely before starting with the next stack. The order of how the stacks are used depends on the index number of the base element (lowest layer).

Absolute Z-Travel for Nested DiTis

In situations where the actual height of a DiTi stack may not match the height assumed by the software, an “absolute Z-Travel” is used. This applies to:

- Resuming a run after pause
- Starting a new run

In these situations, the software always assumes the worst case—i.e., a full stack—when moving to a stack to fetch tips or traveling over it. The Z-Travel height is only reduced once the actual height is definitely known (i.e., after the first successful pickup attempt). This is a safety mechanism that prevents damaging the tip adapters by crashing into a DiTi stack from the side.

Parameters

Tip Selection: Select the tips you want to use. If you do not click a tip, uses all tips by default. To turn off or turn on a tip, click on it.

DiTi Type: Click the DiTi Type arrow to see more options and select the DiTi type of the FCA arm from the list.

Proceed to next position if no tip available

Check box to set whether **Proceed to next position if no tip available** is activated or not. If no DiTi is present at the current position, the software automatically moves to the next position. If a tip is detected but cannot be successfully picked up an error message is shown. Follow the instructions of the error message.

Advanced Parameters

Tip Selection (optional): For this parameter definition, follow the cross reference to Tip Selection (optional).

Air gap volume [µl]: The volume for the air gap in microliters.

Airgap Speed [µl]: In the **Airgap Speed** box, enter the air gap speed you want to use in microliters per second.

Robotic Actions

Prior to execution of the script command, the system performs the following:

- If parameter **Airgap volume** is set to zero and the last pipetted volume in variable **tipContentVolumes** is zero, the system aspirates an air gap of volume 20 µl.
  - If parameter **Airgap volume** is not set to zero and the last pipetted volume in variable **tipContentVolumes** is zero, the system aspirates the air gap as defined by the **Airgap volume** parameter.

11.2.7 Detect Liquid (FCA)

The Detect Liquid command is used to carry out liquid level detection without pipetting and reports the liquid volume for each of the selected Labware wells. The detected liquid volume value is directly written into the corresponding well attribute and can later be read out via the **GetAttribute(LabwareName)** function. Furthermore, a subsequent pipetting step can pipette from the detected height when using the (Liquid Class) LC setting to pipette from a known position.

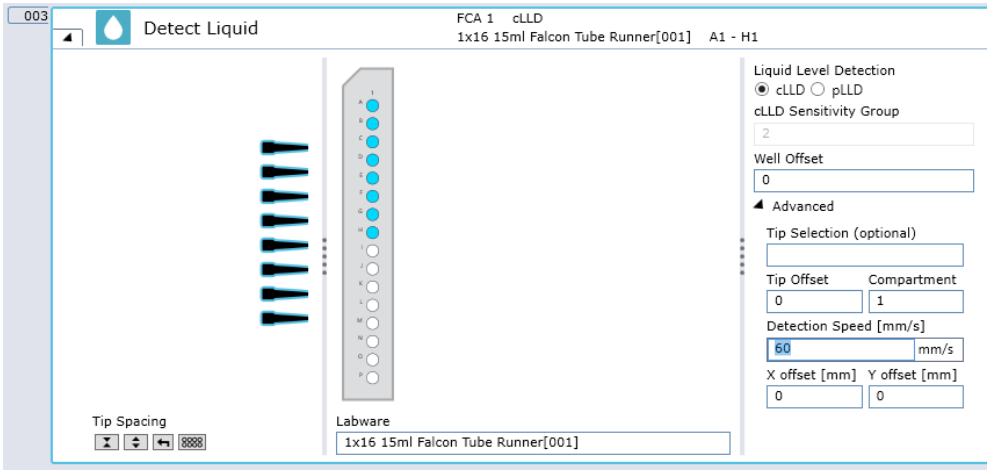


Fig. 175: Detect Liquid cLLD

Liquid Level Detection

Liquid level detection can be performed using either cLLD (Capacitive Liquid Level Detection) or pLLD (Pressure Liquid Level Detection). The operator can select the preferred detection method. The default detection speed is 60 mm/s for cLLD and 20 mm/s for pLLD.

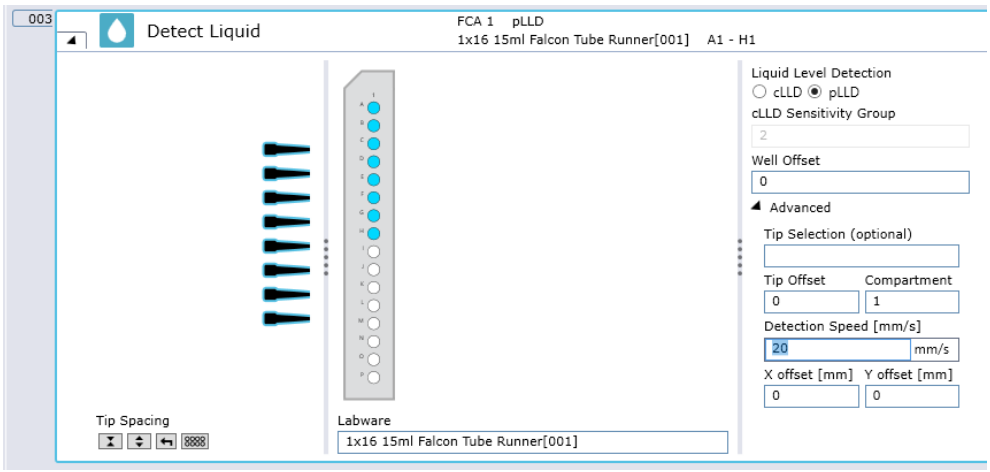


Fig. 176: Detect Liquid pLLD

The pLLD detection method is available only for FCA systems equipped with a MultiSense Board. For other FCA configurations, the pLLD radio button is disabled.

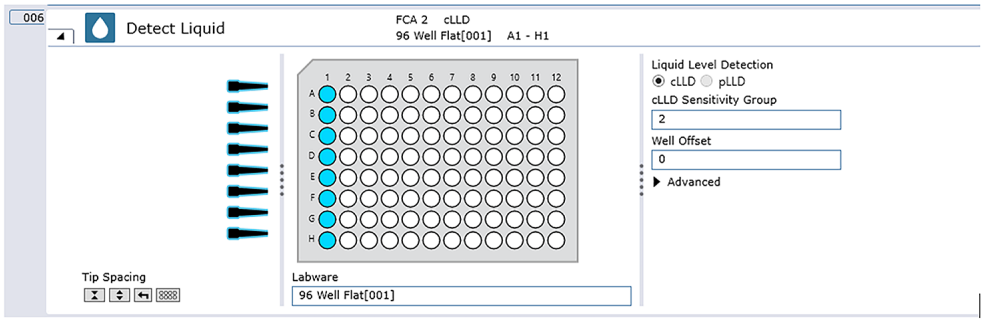


Fig. 177: Detect Liquid pLLD Disabled

cLLD Sensitivity Group

We differentiate between the following **cLLD Sensitivity Group** settings.

Tab. 7: cLLD Sensitivity Group Definition

Liquids / Groups	Group 1: Low	Group 2: Medium	Group 3: High	Group 7: High, Piercing
Conductivity and relative static permittivity characteristics	< 10 µS/cm and relative static permittivity ≥ 24 and < 80	10 to 100 µS/cm or < 10 µS/cm and relative static permittivity > 80	> 100 µS /cm	> 100 µS/cm in combination with tube piercing
Example liquids that represent the group	Ethanol 96%	Deionized water	Tap water, physiological salt solutions (0.9% NaCl)	physiological liquids (e.g., blood)

Well Offset

The Well Offset feature is mainly used when there is a repetitive pipetting procedure defined. Select by how many wells the starting point of the currently selected wells is shifted. The well offset can be greater than the maximum number of wells in a plate, the FCA arm continues with the next plate with the same label and consecutive index.

Tip Spacing

Following **Tip Spacing** arrangements are supported:

- Independent well selection.
- AND several tips in the same well (e.g., 2, 4 and 8 tips)

X-Offset / Y-Offset

Type in the pipetting offset in mm if the pipetting is executed eccentrically.

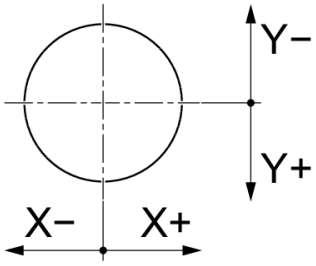


Fig. 178: X- and Y- Well Offset

11.2.8 Aspirate (FCA)

The **Aspirate** script command picks up liquid from specified positions on the worktable.

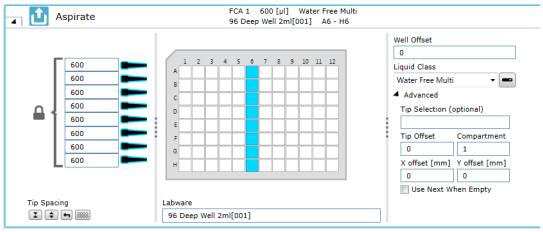


Fig. 179: FCA aspirate script command

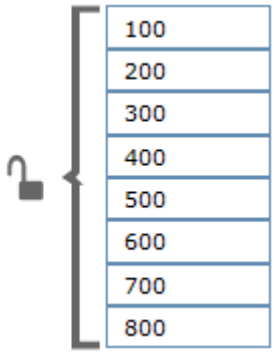


Fig. 180: Individual volumes when padlock icon is opened

Select at least one tip and set Volume ≥ 0.



Fig. 181: Default for tip selection

All tips are selected.

Parameters

Tip Selection:

1. Select the tips you want to use.

   If you do not click a tip, FluentControl uses all tips by default.
2. To turn off or turn on a tip, click on it.
3. If the box is framed in red, click to select a tip.

   Ensure that the Tip Selection is not out of range.
4. Enter the volume to pipette.

   If the padlock icon is opened, individual volumes can be selected.
5. To open the padlock, click on it.

Tip Spacing: The Tip Spacing parameter controls the distance between adjacent pipetting tips for this command. Tip spacing is only relevant if you want to use more than one tip.

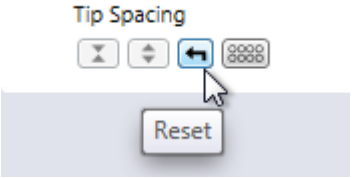


Fig. 182: Tip spacing

Reset: The Reset button is graphically represented with the  icon. Pause on, i.e., briefly resting the mouse pointer on the button to see the description. Reset has the following effects on the tip spacing distance when clicked:

- Reset returns the tip spacing to the default.
- If wells are selected it selects additional wells by basing a default spacing on the selected tips and starting with the first selected well.
- If wells are selected and it is physically not possible to select additional wells by basing a default spacing on the selected tips starting with the first selected well, all selected wells are cleared.

Contract: The Contract button is graphically represented with the  icon. Contract has the following effects on the tip spacing distance when clicked:

- If wells are already selected in the labware control and the contracted well selection spacing is physically possible, the well selection alters to match the new well selection cursor spacing.
- If wells are already selected in the labware control and the contracted well selection spacing is physically not possible, only the first well is selected for all tips.
- If only one well is selected in the labware control, clicking Contract has no effect on the selected well.

Expand: The Expand button is graphically represented with the  icon. Expand has

the following effects when clicked:

- If wells are already selected in the labware control and the expanded selection is physically possible to select, the well selection alters to match the new well selection cursor spacing. The first selected well is not altered by the Tip Spacing buttons.
- If wells are already selected in the labware control and the expanded selection is not physically possible to select, clicking Expand has no effect.

Independent Selection: The Independent Selection button is graphically

represented with the  icon. Independent selection of wells has the following effects on the well selection when clicked:

- Clicking on an unselected well in the labware control selects it for one tip.
- Clicking on a selected well in the labware control selects it for one additional tip.
- It is not possible to select more wells than the number of tips selected.
- If there are the same number of wells selected as tips and you click on a selected well, it reduces the number of tips in the well by 1 (potentially unselecting the well).
- The tip to well assignment is consecutive where tips are assigned from 1 to n to the corresponding 1 to m wells according to the well selection per tip.

If the number of tips selected is lowered to a number smaller than the number of selected wells, a corresponding number of selected wells switches to not selected starting with the highest numbered well.

You can only choose values for tip spacing which are meaningful for the labware geometry. The Flexible Channel Arm (FCA) achieves the highest mechanical accuracy when the tips are not spread. For high-density labware such as 1536-well microplates, you should choose tip spacing such that the tips are adjacent to one another (physical tip spacing 9 mm). Accordingly, for 1536-well microplates you should set tip spacing to 4 (every fourth well).

Labware: In the Labware box, either type your expression or click the labware displayed in the **Worktable** editor window (*Worktable click*).

Well Offset: Select by how many wells the starting point of the currently selected wells is shifted. This is usually expressed as a variable.

Liquid Class: Choose the required liquid class from the pull down list or by means of an expression typed in (e.g., by using a variable defined within your script). The list only shows liquid classes that are suitable for the current configuration of the pipetting instrument. For example, you will not see liquid classes for DiTis if the arm is not currently configured with DiTis.

Advanced Parameters

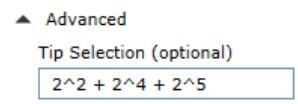


Fig. 183: Tip selection (optional)

Tip Selection (optional): The **Tip Selection (optional)** parameter is resolved as integer (integer variables and expressions are allowed) and interpreted then as a bitmask. Bitmasks are a very resource friendly means of storing settings. The tip selection in the script command is calculated considering both the **Tip Selection (optional)** bitmask and the tips selected via GUI (binary AND). Thus, only tips selected in both masks will be used for pipetting.

The Tip Selection (optional) mask is specified as a numeric, bit-coded value (tip1 = 1, tip2 = 2, tip3 = 4, tip4 = 8, tip5 = 16, tip6 = 32, tip7 = 64, tip8 = 128). Use the sum of the bit-coded values to select more than one tip. In the GUI, tip1 is the upmost tip.

Tip Offset: This parameter lets you modify the tip selection for the pipetting command. The default is 0, which deactivates the offset. A positive value increments the tip number(s) which are specified in the pipetting command. For example, if the pipetting command specifies tips 2 and 3 and you set this parameter to 1, tips 3 and 4 will be used instead. Negative values of the parameter shift the tip selection in the opposite direction.

Compartment: Some special labware types (e.g., for protein crystallography) have wells with several compartments (cavities). FluentControl can pipette a different liquid into each of the compartments. Each well typically has one main compartment and one or more satellite compartments which are directly adjacent and are often smaller in size. Choose the compartment which you want to use for pipetting (the main compartment is normally compartment 1). Well compartments are specified in the labware configuration. This field is inactive (gray) if the well only has one compartment.

X offset [mm]: This is the offset of the tip from the center of the target well (negative is left).

Y offset [mm]:: This is the offset of the tip from the center of the target well (negative = to front, positive = to back).

Use next when empty: When a liquid-detection error occurs (i.e., no liquid/not enough liquid), the FCA will use the next available labware with the same name and next index number—e.g.:
When "Water Trough[001]" is empty, the FCA will log the detection error, but it will not display a user prompt. It will immediately go to the labware named "Water Trough[002]" and try to detect it at that location.
The labware can be a different type, as this feature increments by name only—e.g.:
"Water Trough[001]" can be a 300 µl trough, and "Water Trough[002]" can be a 100 ml trough or a micro plate.

11.2.9 Dispense (FCA)

Dispenses liquid in a labware.

Parameters

Tip Selection: Select the tips you want to use. If you do not click a tip, FluentControl uses all tips by default. To turn off or turn on a tip, click on it.

Volume: Enter the volume to pipette. If the padlock icon is opened, individual volumes can be selected. To open the padlock, click on it.

Labware: In the **Labware** box, either type your expression or click the labware displayed in the **Worktable** editor window (*Worktable click*).

Tip Spacing: For this parameter definition, follow the cross reference to Tip Spacing.

Well Offset: Select by how many wells the starting point of the currently selected wells is shifted. This is usually expressed as a variable.

Liquid Class: Choose the required liquid class from the pull down list or by means of an expression typed in (e.g., by using a variable defined within your script). The list only shows liquid classes that are suitable for the current configuration of the pipetting instrument. For example, you will not see liquid classes for DiTis if the arm is not currently configured with DiTis.

11.2.10 Mix (FCA)

Mixes the liquid(s) in one or more wells of a selected labware. Performs a mix operation with the FCA (see “FCA” [▷ 226]) by aspirating and then dispensing at the same position.

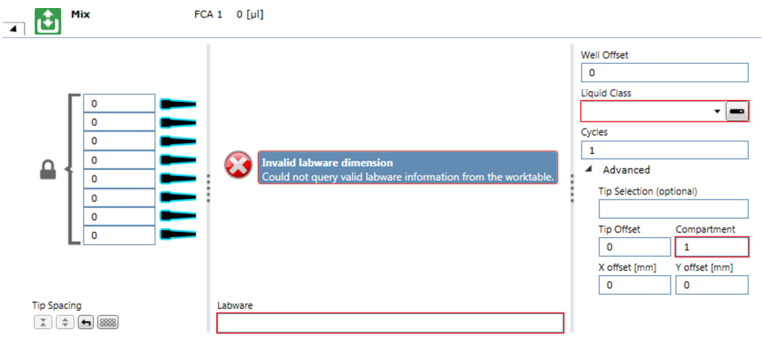


Fig. 184: Mix

Parameters

Tip Selection: Select the tips you want to use. If you do not click a tip, FluentControl uses all tips by default. To turn off or turn on a tip, click on it.

Volume: Enter the volume to pipette. If the padlock icon is opened, individual volumes can be selected. To open the padlock, click on it.

Tip Spacing: For this parameter definition, follow the cross reference to Tip Spacing.

Labware: In the **Labware** box, either type your expression or click the labware displayed in the **Worktable** editor window (*Worktable click*).

Well Offset: Select by how many wells the starting point of the currently selected wells is shifted. This is usually expressed as a variable.

Liquid Class: Choose the required liquid class from the pull down list or by means of an expression typed in (e.g., by using a variable defined within your script). The list only shows liquid classes that are suitable for the current configuration of the pipetting instrument. For example, you will not see liquid classes for DiTis if the arm is not currently configured with DiTis.

Cycles: Set the number of mixing cycles. You can select existing variables.

11.2.11 Drop Tips (FCA)

Use this command to discards the specified DiTis into the specified DiTi waste.

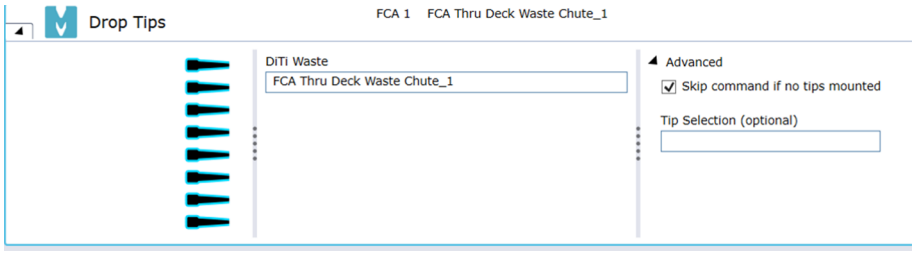


Fig. 185: Drop Tips

Parameters

Tip Selection: Select the tips you want to use. If you do not click a tip, uses all tips by default. To turn off or turn on a tip, click on it.

DiTi Waste: Specify the DiTi waste on the worktable: Click into the DiTi Waste box and select your labware from the Worktable Editor or type the name of your labware.

*The **Skip command if no tips mounted** checkbox is enabled in script commands automatically generated by smart commands and worklists to ensure that the generated script starts without mounted tips while not triggering an unnecessary mechanical move and associated delay.*

11.2.12 Dispense System Liquid (FCA)

It is not possible to use this command with an AirFCA.

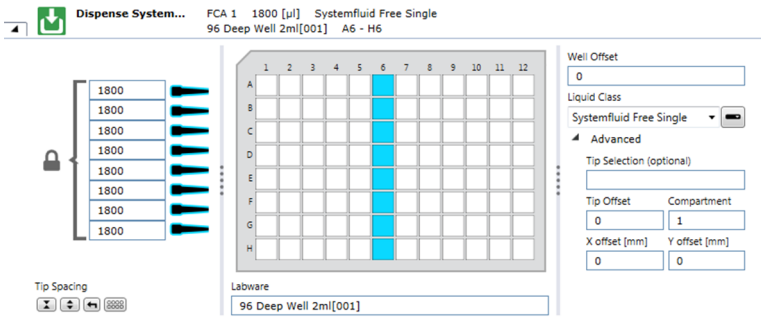


Fig. 186: Dispense system

Parameters

Tip Selection: Select the tips you want to use. If you do not click a tip, FluentControl uses all tips by default. To turn off or turn on a tip, click on it.

Volume: Enter the volume to pipette. If the padlock icon is opened, individual volumes can be selected. To open the padlock, click on it.

Labware: In the **Labware** box, either type your expression or click the labware displayed. In the **Worktable** editor window (*Worktable click*).

Tip Spacing: For this parameter definition, follow the cross reference to Tip Spacing.

Well Offset: Select by how many wells the starting point of the currently selected wells is shifted. This is usually expressed as a variable.

If well offset is invalid, the text box is framed in red. On mouse over, the message: "Enter a valid well offset." appears.

Liquid Class: Choose the required liquid class from the pull down list. The list only shows liquid classes which are suitable for the current configuration of the pipetting instrument. For example, you will not see liquid classes for DiTis if the FCA is not currently configured with DiTis.

Robotic Actions: If the dispense volume is greater than the syringe volume, the system splits the volume into multiple volumes of equal size according to the syringe size and perform multiple dispense actions.

- **Example:** For a dispense volume of 2000µl with syringe size of 1250µl, the system dispenses two times 1000 µl. For a dispense volume of 2700µl, the system dispenses three times 900 µl.

11.2.13 Flush (FCA)

The **Flush** script command flushes the selected tips. The **Flush** command is used also as a direct command.

It is not possible to use this command with an AirFCA.

The arm moves over the selected waste and flushes the selected tips according to section **Flush** of the selected liquid class.

After a **Flush** command was executed, the system resets the maximal speeds, acceleration and deceleration for FCA movements in x, y, and z defined by the **MicroScript** command to the predefined values in the system configuration file.

Parameters

Tip Selection: Select the tips you want to use. If you do not click a tip, FluentControl uses all tips by default. To turn off or turn on a tip, click on it.

Waste: Specify the name of the waste on the worktable: Click into the **Waste** box and select your labware from the **Worktable** editor or type the name of your labware.

Liquid Class: Choose the required liquid class from the pull down list or by means of an expression typed in (e.g., by using a variable defined within your script). The list only shows liquid classes that are suitable for the current configuration of the pipetting instrument. For example, you will not see liquid classes for DiTis if the arm is not currently configured with DiTis.

Flush Volume: Define the flush volume

11.2.14 Wash (FCA)

It is not possible to use this command with an AirFCA.

Use this command to wash the liquid handling system from the FCA (see “FCA” [▷ 226]). The Wash Tips command is used to flush and wash fixed tips or to flush DiTi adapters using a wash station. It is not intended for flushing DiTis (normally, DiTis should not be flushed). If you are using DiTis, make sure in your script that there are none attached when this command is executed.

The arm moves over the selected waste. The arm performs the first wash step according to section Flush of the selected liquid class. The arm performs the second wash step according to section **Wash** of the selected liquid class.

After the execution of a wash tips command, the system resets the maximal speeds, acceleration and deceleration for FCA movements in x, y, and z defined by the **MicroScript** command to the predefined values in the system configuration file.

*The **Wash Tips** command is not intended for flushing DiTis (DiTis should not normally be flushed). If you are using DiTis, make sure in your script that there are none attached when this command is executed.*

Parameters

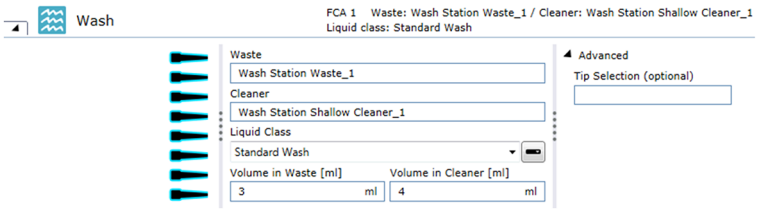


Fig. 187: Wash

Waste: Specify the name of the waste on the worktable: Click into the Waste box and select your labware from the Worktable Editor or type the name of your labware.

Cleaner: Name of the selected cleaner on the worktable. It is possible to edit this name manually. Allowed Labware Types: CleanerShallow, CleanerDeep. Default: “Wash Station Shallow Cleaner_1”

Liquid Class: Choose the required liquid class from the pull down list or by means of an expression typed in (e.g., by using a variable defined within your script). The list only shows liquid classes that are suitable for the current configuration of the pipetting instrument. For example, you will not see liquid classes for DiTis if the arm is not currently configured with DiTis.

Volume in Waste [ml]: Define the flush volume.

Volume in Cleaner [ml]: Define the wash volume

11.2.15 Periodic Wash (FCA)

It is not possible to use this command with an AirFCA.

If DiTis are used on the FCA (see “FCA” [▶ 226]), the system provides a script command **Periodic Wash** to define a periodical washing of the tips in order to reset the system trailing air gap. The **Periodic Wash** script command is not executed if the value of the parameter **Drop Tips Cycle** is zero.

The FCA will move over the selected waste and flush the selected tips according to the **Flush** section of the defined liquid class. This is repeated as many times as specified by **Drop Tips Cycle**.

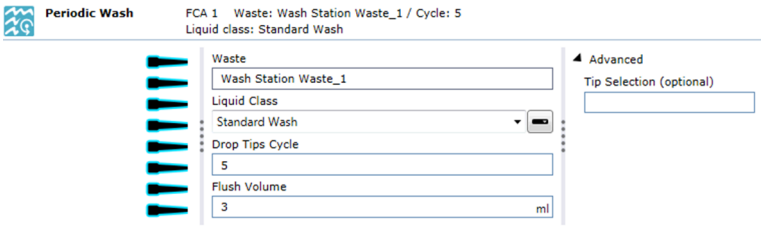


Fig. 188: Periodic wash script command

Waste: Specify the name of the waste on the worktable.

Liquid Class: Click the Liquid Class arrow to see all available liquid classes to select a liquid class. Alternatively, you can use a variable defined within the script to specify the liquid class.

Drop Tips Cycle: Define after how many drop tips executions **Periodic Wash** shall be performed. If **Drop Tips Cycle** is out of range, the text box is framed in red and a message indicating to enter a valid drop tip cycle is shown.

Flush Volume: Define the Flush volume in [ml].

11.2.16 Empty Tip Contents (FCA)

Executes the **Empty Tips** section of the selected liquid class. The system executes the **Empty Tips** section of the selected liquid class and moves the plunger to volume of the system trailing air gap (STAG) which is defined by **tipContentVolumes** as the last pipetted volume in the variable.

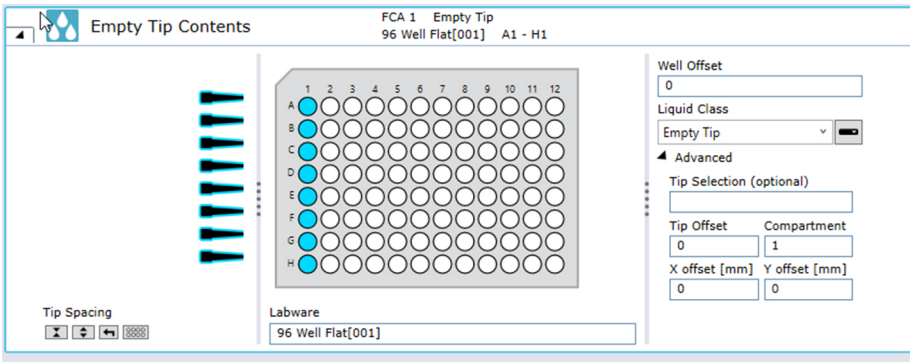


Fig. 189: Empty Tip Contents

Parameters

Tip Selection: Select the tips you want to use. If you do not click a tip, FluentControl uses all tips by default. To turn off or turn on a tip, click on it.

Tip Spacing: For this parameter definition, follow the cross reference to Tip Spacing.

Labware: In the Labware box, either type your expression or click the labware displayed in the Worktable editor window (Worktable Click).

Liquid Class: Choose the required liquid class from the pull down list or by means of an expression typed in (e.g., by using a variable defined within your script). The list only shows liquid classes that are suitable for the current configuration of the pipetting instrument. For example, you will not see liquid classes for DiTis if the arm is not currently configured with DiTis.

Advanced Parameters

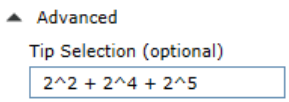


Fig. 190: Tip selection (optional)

Tip Selection (optional): The **Tip Selection (optional)** parameter is resolved as integer (integer variables and expressions are allowed) and interpreted then as a bitmask. Bitmasks are a very resource friendly means of storing settings. The tip selection in the script command is calculated considering both the **Tip Selection (optional)** bitmask and the tips selected via GUI (binary AND). Thus, only tips selected in both masks will be used for pipetting.

The Tip Selection (optional) mask is specified as a numeric, bit-coded value (tip1 = 1, tip2 = 2, tip3 = 4, tip4 = 8, tip5 = 16, tip6 = 32, tip7 = 64, tip8 = 128). Use the sum of the bit-coded values to select more than one tip. In the GUI, tip1 is the upmost tip.

Tip Offset: This parameter lets you modify the tip selection for the pipetting command. The default is 0, which deactivates the offset. A positive value increments the tip number(s) which are specified in the pipetting command. For example, if the pipetting command specifies tips 2 and 3 and you set this parameter to 1, tips 3 and 4 will be used instead. Negative values of the parameter shift the tip selection in the opposite direction.

Compartment: Some special labware types (e.g., for protein crystallography) have wells with several compartments (cavities). FluentControl can pipette a different liquid into each of the compartments. Each well typically has one main compartment and one or more satellite compartments which are directly adjacent and are often smaller in size. Choose the compartment which you want to use for pipetting (the main compartment is normally compartment 1). Well compartments are specified in the labware configuration. This field is inactive (gray) if the well only has one compartment.

X offset [mm]: This is the offset of the tip from the center of the target well (negative is left).

Y offset [mm]:: This is the offset of the tip from the center of the target well (negative = to front, positive = to back).

11.2.17 LLD Error Handling (FCA)

The **LLD Error Handling** command allows to set up default handling options for liquid handling. These options override the settings specified in the liquid class and are valid until the current script ends.

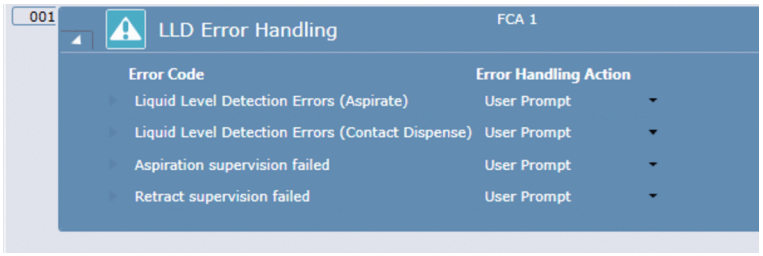


Fig. 191: LLD Error Handling

To disable LLD Error Handling for a part of the script, insert a second **LLD Error Handling** command, and select **User Prompt** or **Liquid Class Defaults** in the **Error Handling Action** box.

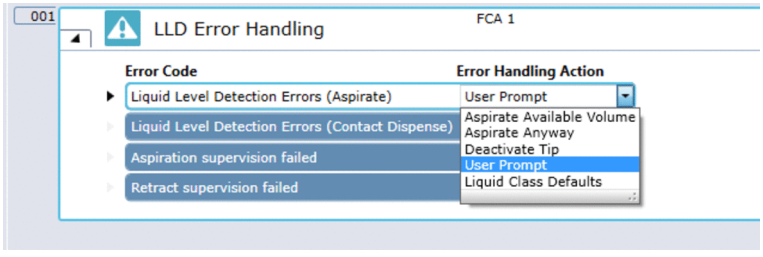
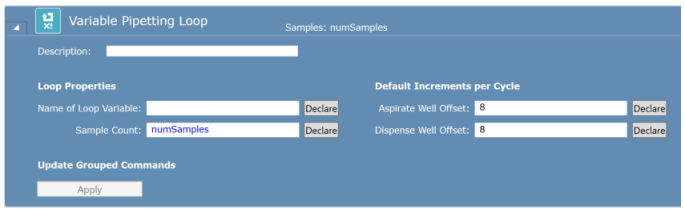


Fig. 192: Error Handling Action box

11.2.18 Variable Pipetting Loop (FCA)



This command allows the automated, iterative execution of a set of script commands – representing a pipetting sequence – with automated variable parameterization.

The system prevents by context check that the user groups the variable pipetting loop command within itself. It is allowed to nest the variable pipetting loop group command inside other group and loop commands.

Parameters

Description: Describe the variable pipetting loop.

Loop Properties

Name of Loop Variable: Define the variable counting the loop cycles. Use the Declare button to register a variable, as defined in this parameter, with its name and type under the script section Variables and with read/write permission.

Sample Count: Stores the expression for the number of samples to pipette. Use the Declare button to register a variable, as defined in this parameter, with its name and type under the script section Variables and with read/write permission.

Default Increments per Cycle

Aspirate Well Offset: Define the well offset for Aspirate commands.

Dispense Well Offset: Define the well offset for Dispense commands.

Update Grouped Commands

Apply: After the group command parameters are set by the user and as each script command is added to the group or when the Apply button is pressed, the system performs automated parametrization for the script parameters Tip Mask and Well Offset for all grouped script commands as follows:

1. For the script parameter Tip Mask, the system uses the expression “tipsNeeded(tipMask, sampleCount)”. You can edit this parameter after it has been automatically set.
2. For script parameter Well Offset:
   a. If you select an automatic parametrization for Aspirate, the system uses the expression “aspirateCount() * (<Aspirate Aspiration Offset>)” for every Aspirate within the Variable Pipetting Loop. You can edit this parameter after it has been automatically set.
   b. If you select an automatic parametrization for Dispense, the system uses the expression “dispenseCount() * (<Dispense Dispense Offset>)” for every Dispense within the Variable Pipetting Loop. You can edit this parameter after it has been automatically set.

If the Variable Pipetting Loop command is executed, the system repeats the sequence of script commands within the loop command until the specified number of samples has been pipetted, i.e. the sampleCount() function equals zero at end of the sequence.

System Functions

sampleCount(): This function returns the number of samples remaining to be pipetted. The initial value is determined from the parameter “Sample Count”. At the end of each Variable Pipetting Loop, the value decrements by the number of distinct tips having aspirated.

aspirateCount(): This function returns the number of aspirate commands having been executed within the Variable Pipetting Loop.

dispenseCount(): This function returns the number of dispense commands having been executed within the Variable Pipetting Loop.

11.2.19 Move Arm (FCA)

Moves the Flexible Channel Arm (FCA) from its current position to another position without performing an aspirate or dispense operation

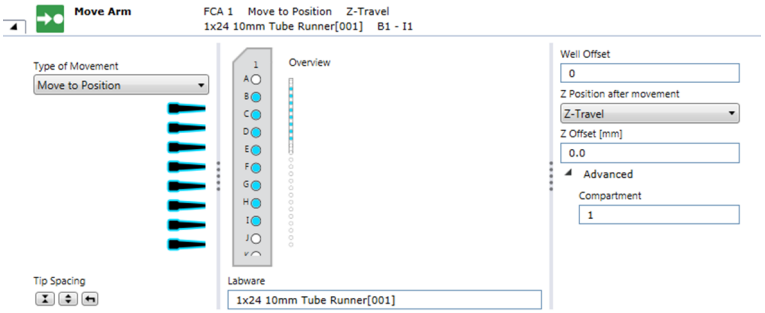


Fig. 193: Move arm

Parameters

Type of Movement: Clicking the arrow opens the Type of Movement list:

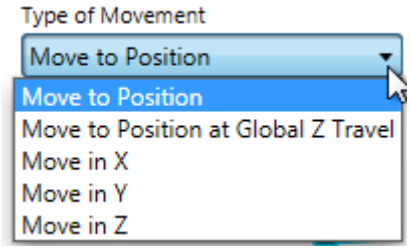


Fig. 194: Move to position

Move to Position: The system uses the standard movements as determined by Pathfinder and moves the arm to the labware with maximum speed.

Move to Position at Global Z Travel: The system first moves the arm in Z to Z-Travel and then moves in X and Y direction to the target position by a Pathfinder controlled movement. Finally, the system moves the arm to the Z position as defined in the move arm command. The system moves the arm to the labware with maximum speed.

Move in X: This selection determines that a relative motion in the direction of X is made.

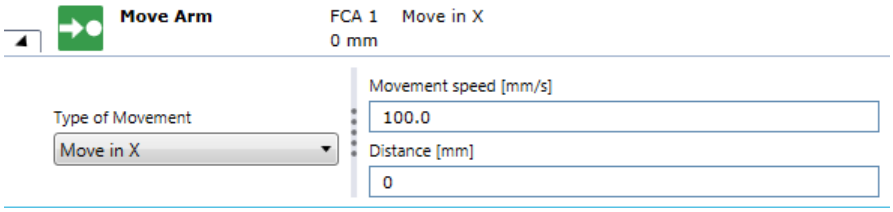


Fig. 195: Move in Y

Move in Y: This selection determines that a relative motion in the direction of Y is made.

Move in Z: This selection determines that a relative motion in the direction of Y is made.

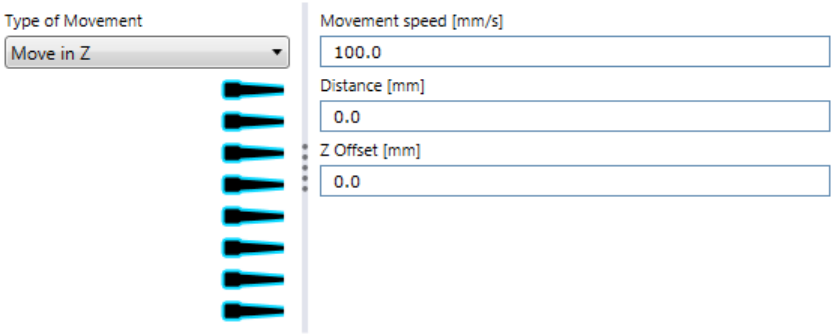


Fig. 196: Movement speed [mm/s]

Movement speed [mm/s]: Set the movement speed in millimeters per second.

Distance [mm]: Distance of arm movement in millimeters if **Move** in X, Move in Y or **Move** in Z is selected.

Z-Offset [mm]: Defines an offset in millimeters to the Z-Position after movement (positive = lower, negative = higher).

Tip Spacing: For this parameter definition, follow the cross reference to **Tip Spacing**.

Labware: In the **Labware** box, either type your expression or click the labware displayed in the **Worktable** editor window (*Worktable click*).

Well Offset: Select by how many wells the starting point of the currently selected wells is shifted. This is usually expressed as a variable.

Z Position after movement: Set the Z position. **Range**: ["Z-Travel", "Z-Start", "Z-Dispense", "Z-Max", "Z-Bottom"]. **Default**: "Z-Travel".

Advanced Parameters

Compartment: Some special labware types (e.g. for protein crystallography) have wells with several compartments (cavities). FluentControl can pipette a different liquid into each of the compartments. Each well typically has one main compartment and one or more satellite compartments which are directly adjacent and are often smaller in size. Choose the compartment which you want to use for

pipetting (the main compartment is normally compartment 1). Well compartments are specified in the labware configuration. This field is inactive (gray) if the well only has one compartment.

11.2.20 Remove Empty Tip Spacers (FCA)

The **Remove Empty Tip Spacers** script command allows the user to remove empty tip spacers or stacked tips with the FCA using the Transfer Tools. The command removes any empty tip spacers on the worktable and moves them to the selected waste. This is useful for systems which do not have a RGA or the RGA is very busy with plate transfers.

Parameters

DiTi Type: Drop-down combo box with name of tip type to observe. Expressions are supported. If you click on a tip type on the worktable, the clicked tip type is entered into the edit field.

DiTi Name Filter: Labware name filter (like in RGA Replace Empty DiTi Rack command). Name of tip labware to observe. If given other labware matching the above selected tip type but not matching the name filter are ignored.

Waste: Waste to use. Expressions are supported. If you click on a waste on the worktable, the name of the clicked waste is entered into the edit field.

Workflow

With the **Remove Empty Tip Spacers** command, you can search for any empty tip spacers on the (pipettable) worktable (same as Replace Empty DiTi Racks command but only for stacked tips).

Then the command does the following for all found empty tip spacers:

Portrait DiTi racks are not checked.

- **Fetch Transfer Tool**: Select next available **Transfer Tool** and fetch it with first free and not broken Tip from Front (normally Tip 8). If no more **Transfer Tools** are available the system shows an error dialog and pause the run. The user has the options to re-load the **Transfer Tool** carriers or abort the run. If a **Transfer Tool** cannot be fetched the system automatically re-tries three times at the same position before raising an error. If the **Transfer Tool** still cannot be fetched the system displays an error dialog and pauses the run. The user has the option to re-try, continue with next position or abort.
- **Fetch empty tip spacer**: **Move over empty spacer**; Tip with Tool over corresponding well Tip with Tool is moved into corresponding well of the tip spacer to be removed and move up again. If no spacer is found (no hard stop detected), the Transfer Tool is lost or a crash occurs the system shows an error dialog and pauses the run. The dialog tells the user to look for a misplaced tip spacer and remove it. The user has the option to continue or abort.

Drop to waste: The command does the following

- **Move to waste**: As Waste it is only allowed the MCA or Plastic waste. **NOTICE! Different position than with DiTis.**
- **Eject Transfer Tool**: If Transfer Tool is still detected after the eject an error dialog is shown by the system and the run is paused. The user has the option to re-try or abort.

11.2.21 Pressure Monitored Pipetting (PMP)

The PMP hardware required by this feature is only available on the AirFCA.

Currently the PMP functionality is limited to the evaluation of liquid aspiration and cannot be used in combination with Aspiration Supervision (tracking option "Yes, supervised").
Currently the PMP functionality cannot be used together with the Phase Separator and the Transfer Phase command, nor with liquid classes using the Detect Phase microscript command.

11.2.21.1 How to Activate the PMP Feature

The availability of the PMP feature depends on a corresponding license being available for the connected instrument. This is checked during startup of the FluentControl.

11.2.21.2 How to Set Up a Liquid Class for PMP

If the PMP feature is activated, the liquid class editor provides additional settings in the Detection and Positioning section (Aspirate) of AirFCA sub classes:

- Density and viscosity of the liquid
- Error handling option in case of an aspiration error
- Evaluation model
- Threshold values depending on selected evaluation model

Currently only the following tip types, and volume rate ranges are supported:
- 200 µl DiTis (filtered and non-filtered): 16 µl/s – 160 µl/s
- 350 µl DiTis:16 µl/s – 160 µl/s
- 1000 µl DiTis (filtered and non-filtered): 37 µl/s – 200 µl/s
- 1000 µl Wide-Bore DiTis: 60 µl/s – 310 µl/s

The PMP feature supports two different means of evaluating an aspiration:

- ADP Clot Detection
- Evaluation by a neural network

FluentControl includes different neural nets for classification and anomaly detection as shown in the drop-down list.

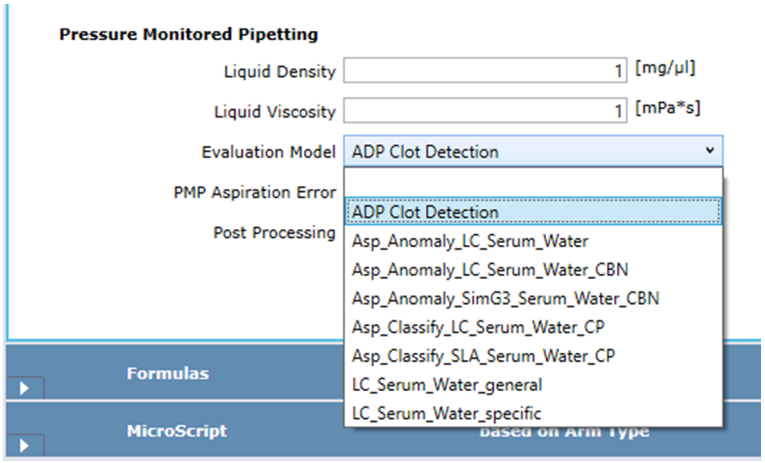


Fig. 197: Liquid Class Editor - Evaluation Model selection

Depending on the selected evaluation model (ADP or neural net), different threshold values can be chosen.

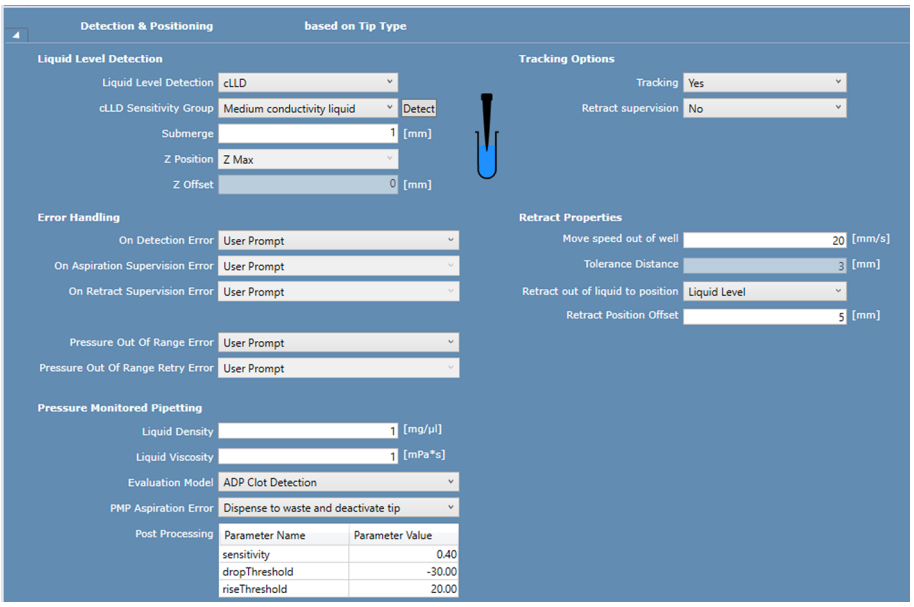


Fig. 198: Liquid Class Editor - ADP Clot Detection

11.2.21.2.1 Threshold Values for ADP Clot Detection

If the ADP Clot Detection algorithm has been selected, the user can define the following parameters:

Parameter	Minimum	Maximum	Description
Sensitivity	0.0	1.0	Allowed pressure variation (factor)

Parameter	Minimum	Maximum	Description
Rise Threshold	0	1000	Threshold by how many [Pa] the pressure needs to rise after aspiration to exclude a clot.
Drop Threshold	-1000	0	Threshold by how many [Pa] the pressure needs to drop to exclude an air aspiration.

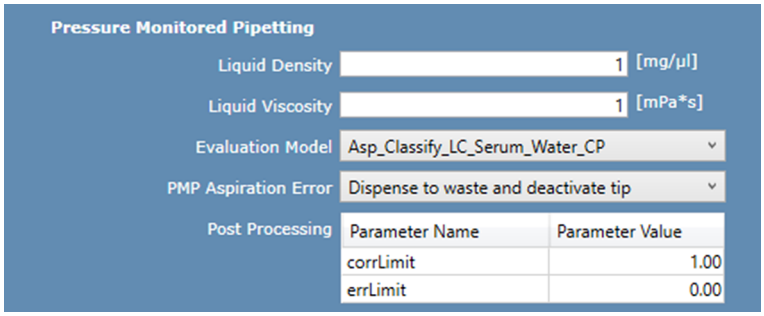


Fig. 199: Liquid Class Editor - Neural Net (Classification)

11.2.21.2.2 Threshold Values for Neural Net

If a neural net for classification has been selected as evaluation model, the user can define the following parameters to control sensitivity. The Neural Net determines 2 probabilities for each pipetting curve:

- The probability that the pipetting is correct
- The probability that the pipetting has an error

Where 1 represents 100% certainty and zero represents no certainty. The user can control the sensitivity of the evaluation by defining how certain they want to be for a Correct or Error determination.

Parameter	Minimum	Maximum	Description
Correct Limit	0.0	1.0	If the “Correct” probability is equal to or greater than this limit, the classification is reported as “Correct” even if there is an error rating with higher probability.
Error Limit	0.0	1.0	If the highest probability of an error rating does not surpass this limit, the classification is reported as “Correct”.

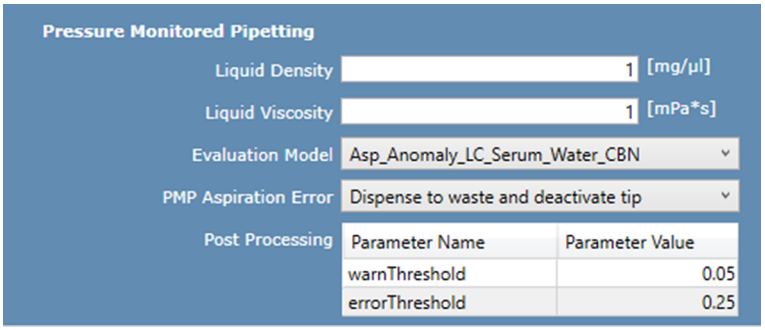


Fig. 200: Liquid Class Editor - Neural Net (Anomaly)

If a neural net for anomaly detection has been selected as evaluation model, the user can define the following parameters to control sensitivity.

Parameter	Minimum	Maximum	Description
warnThreshold	0.0	10.0	If the anomaly value (deviation from ideal pressure curve) exceeds this threshold, a warning is raised and the sample is flagged accordingly, but the run continues. There is no error handling being performed.
errorThreshold	0.0	10.0	If the anomaly value exceeds this threshold, this is treated as a PMP Aspiration Error and the selected error handling is executed.

11.2.21.2.3 Error Handling Options

The following error handling options can be configured in the liquid class to be performed in case of an aspiration error:

User Prompt: A prompt will be shown, requiring the operator to choose the actual error handling option at runtime. The operator can choose one of the following options.

Ignore and Continue: The run continues, but a warning message is shown in the System Trace View.

Dispense back to source and deactivate tip: Liquid is dispensed back to the source well, the trailing air gap is aspirated and the channel is deactivated until the end of the current pipetting sequence.

Dispense back to source and retry with new tip: Liquid is dispensed back to the source well. After the non-affected channels have completed the aspiration, the FCA mounts new DiTis on the affected channels and retries the aspiration once. If the retry also fails, the error handling reverts to “Dispense back to source and deactivate tip”.

Dispense to waste and deactivate tip: After the non-affected channels have completed the aspiration, the FCA moves to the liquid waste, dispenses the liquid and deactivates the channel until the end of the current pipetting sequence.

11.2.21.3 How to Use the PMP Feature in the Script

If a PMP license is available for the connected instrument, the software provides two additional script commands for the AirFCA:

- Switch On PMP
- Switch Off PMP

The **Switch On** PMP command activates pressure curve recording for the selected channels. Those channels will record a pressure curve for each pipetting sequence, starting with the pick up of a disposable tip and concluding with its being discarded, until the channels are deactivated by a **Switch Off** PMP command.

The recorded pressure curves are stored in:
C:\ProgramData\Tecan\VisionX\PressureData

In addition to the curve recording, the aspiration of liquid will be evaluated according to the settings in the liquid class.

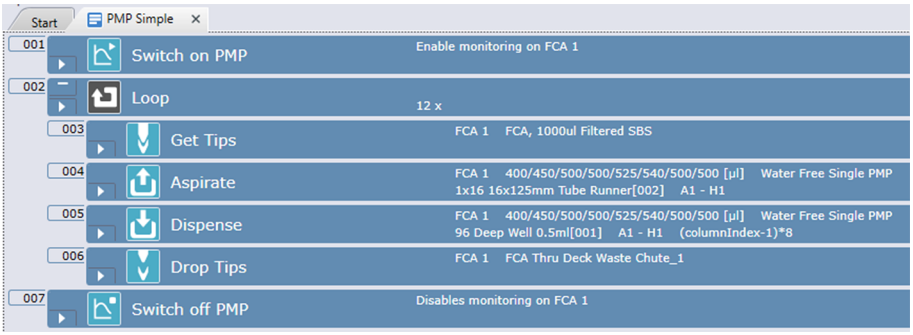
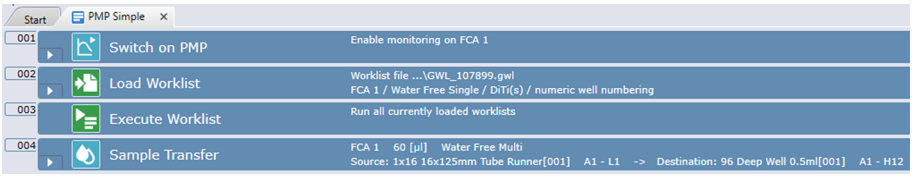


Fig. 201: PMP Usage in Script (Example)

The PMP functionality can also be used with worklists and Smart Commands simply by switching on the channels prior to the pipetting as shown below:



11.2.21.4 Sample Tracking

The User can integrate sample tracking into scripts using PMP. Samples must be registered and have unique barcodes. Reports can be generated by selecting a PMP Plate Map Report which will include the sample ID, the pipetting status, and a link to the curve file.

11.2.21.5 Curve Viewer

A Curve Viewer tool is available on request that can be used to visualize the recorded curve, and also to interactively post-process curves with different parameters for the Neural network.

11.2.21.6 Limitations and Known Issues

The following limitations apply to the use of the PMP functionality in FluentControl :

- The PMP functionality is limited to the evaluation of liquid aspiration.
- The PMP functionality cannot be used in combination with Aspiration Supervision. (tracking option “Yes, supervised”)

11.3 FCA Gripper

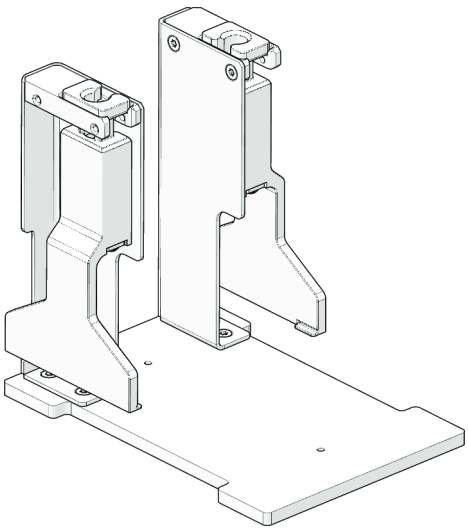


Fig. 202: FCA gripper

The FCA gripper is an option for the FCA that—in addition to pipetting—also allows using the arm for moving labware. Instead of disposable tips FCA gripper fingers are attached to the FCA. The FCA can automatically get and drop the FCA gripper fingers during the run. The FCA gripper allows moving labware (plates and DiTi boxes) with the FCA “Transfer Labware (FCA Gripper)” [▷ 274] command—as an alternative to the RGA Transfer Labware command.

FCA Gripper Fingers

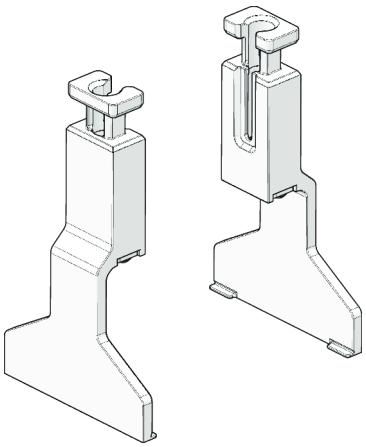


Fig. 203: FCA gripper fingers

The FCA can get and drop the FCA gripper fingers with the “Get Fingers (FCA Gripper)” [276] and “Drop Fingers (FCA Gripper)” [276] commands.

The FCA gripper fingers are designed for up to 20,000 pickup moves. When 90% of this limit has been reached, a different warning message that the fingers need to be replaced will be shown. A new warning message is shown when 100% of the lifetime is reached.
Once the fingers have been replaced, a key operator can confirm the replacement and reset the counter by clicking **Settings > Configure System > Devices > FCA > Reset Counter**.

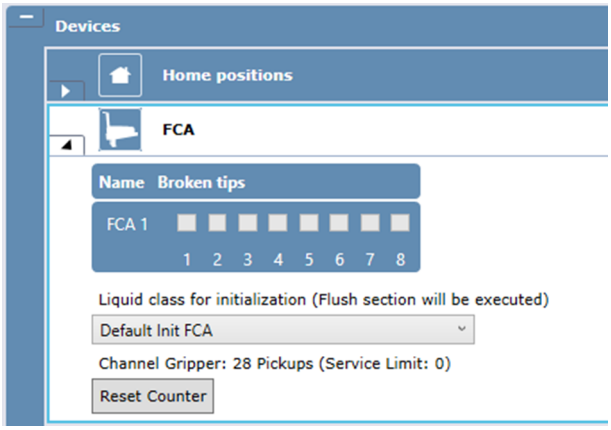


Fig. 204: Reset Counter

FCA Gripper Docking Station

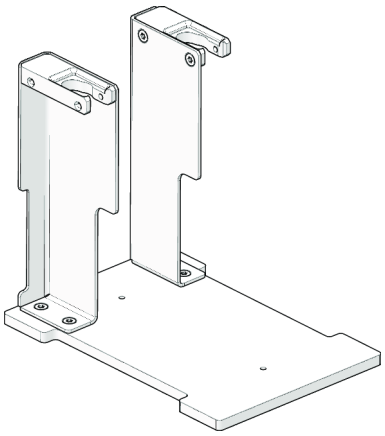


Fig. 205: FCA gripper docking station

The FCA gripper docking station is a docking station for the FCA gripper fingers that is mounted on a standard microplate nest position on the deck of the Fluent. At runtime the FCA gets the FCA gripper fingers from this position with the “Get Fingers (FCA Gripper)” [▷ 276] command and places them back to this position with the “Drop Fingers (FCA Gripper)” [▷ 276] command.
The FCA gripper docking station consists of two upright holders. The distance between them can be specified in the **Gripperwidth** custom attribute of the gripper nest.

11.3.1 Transfer Labware (FCA Gripper)



Fig. 206: Transfer Labware

Use this command in combination with an FCA gripper to move labware on the deck of the Fluent.

Parameters

Back to Base

The system will move the labware to its base. If this option is selected, the UI controls for location and site are dimmed.

To location

The system will move the labware to a specified location.

Labware

Specify the labware to be transferred by the FCA by entering the labware name (or an expression that resolves into a labware name) or click the labware displayed in the Worktable editor window (Worktable click).

Target location

Select a target location. The usage of "<BASE>" as an alias for the base location is supported.

Target Position

By default the system will place the labware on the first free site of the target location. The user can override this behavior by checking **Only use selected site** and specify a dedicated site on the target location. The target site can be specified by entering the site number (or an expression which resolves into a site number) or by selecting a carrier or site on the worktable.

***Context check**: The system will check if the move is possible (labware exists, location/target site is available) and raise context check messages if an error is detected.*

***Move simulation**: The system will simulate the move in Move Simulation mode and update the worktable (move the labware to the destination) if the move is possible.*

***Grip position**: By default the labware is gripped at the very bottom. The two tabs on the bottom edge of each gripper finger are designed to insert beneath the labware object to enable lifting with the weight of the labware on the vertical Z-axis rather than by applying pressure on the FCA Y-axis. It is possible to define a custom attribute "FcaGripZOffset" with the labware which specifies a Z-offset in mm that is applied when gripping a plate in the FCA Transfer Labware command. Such a grip offset is recommended when moving DiTi boxes.*

After transferring the labware the Detect Labware command is automatically executed and an error message is displayed if the labware is not at the expected position.

Sometimes this check can be undesired as it could tilt the plate to the side or compromise sterility. It is possible to disable the presence check in the advanced option of the command. If the presence check is disabled, it must be ensured through other means that the plate transfer was successful—for example, by detecting liquid in the plate.

11.3.2 Detect Labware (FCA Gripper)

This command requires that FCA gripper fingers are mounted. It serves to verify that labware is physically present at the position assumed in the software.

During the execution of the command the FCA moves over the labware and moves down on the edge of the plate with one of the gripper fingers. The height at which the FCA gripper finger touches the plate is detected and compared to the Z-dimension of the plate defined in the software. If this matches within a tolerance of +/- 3 mm, the labware is considered to be present and script execution continues—if not, an error message is displayed.

11.3.3 Get Fingers (FCA Gripper)



Fig. 207: Get Fingers

Use this command to pick up the FCA gripper fingers.

Parameters

Gripper Fingers

Select the FCA gripper fingers to be picked up.

11.3.4 Drop Fingers (FCA Gripper)

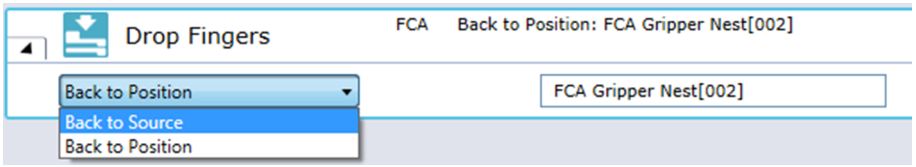


Fig. 208: Drop Fingers

Use this command to place the FCA gripper fingers at a specified position on the deck of the Fluent.

Parameters

Back to Source

By selecting this option the FCA gripper fingers are returned to the position from where they were fetched.

Back to Position

By selecting this option the user can specify a return position for the FCA gripper fingers.

11.3.5 Replace Empty Tip Box (FCA Gripper)

Use this command to replace an empty tip box on the worktable with a full one.

Empty box: The system checks for an empty box of the selected type at a position used for pickup, i.e. on a carrier marked as “accessible for pipetting arm”.

Full box: The system searches for a full box of the selected type at a storage position, i.e. on a carrier not marked as “accessible for pipetting arm”. Boxes in storage are consumed in rear-to-front, left-to-right order. Stacks of boxes are consumed from bottom to top.

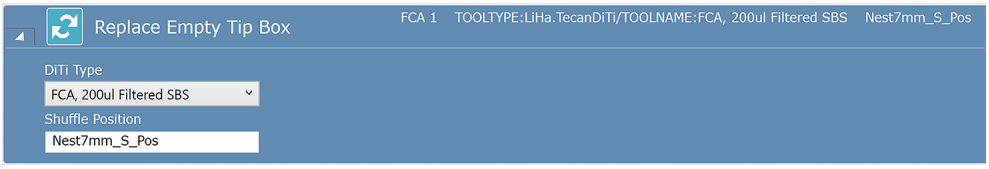


Fig. 209: Replace Empty Tip Box

Parameters

DiTi Type: Select the type of disposable tips to replace from the dropdown list or by clicking a tip box on the worktable.

Shuffle Position: Specify the location to use for stacking and unstacking of boxes and as temporary storage during the exchange.

When using this command with 1000µl tip boxes, two shuffle positions are required, since those boxes cannot be stacked due to their height.

The command is intended to be used in an ADH workflow, i.e. in a subroutine registered for the tip type via an On Empty Tray command.

The command only executes a sequence of Transfer Labware commands to perform the replacement. The ADH subroutine has to include the necessary Get Fingers and Drop Fingers commands.

11.4 RGA

11.4.1 Driver Configuration (RGA)

If the instrument is equipped with an RGA, its general settings can be configured under Configure System|Drivers|RGA …

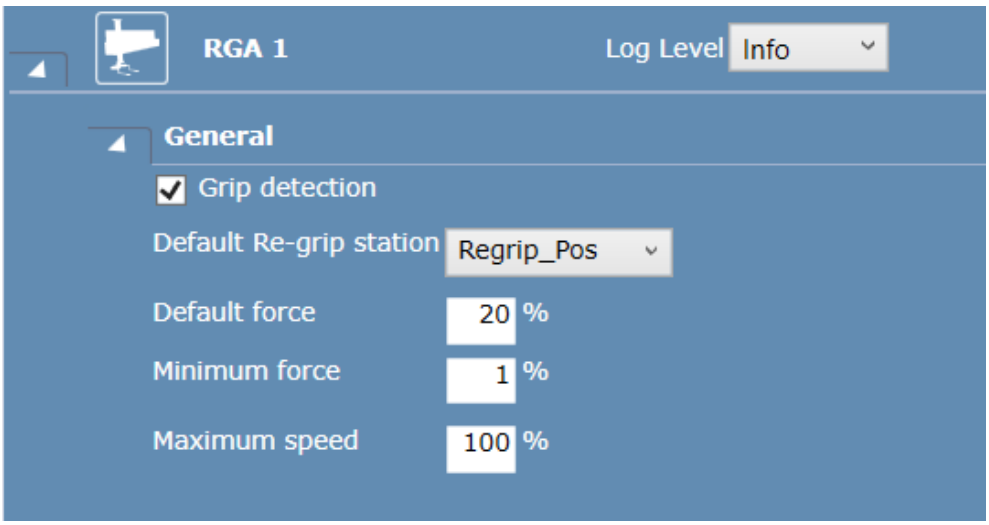


Fig. 210: RGA Driver Configuration

Grip detection: If enabled, the system will check for each plate gripping action whether or not an object was detected by the hardware and trigger the appropriate error handling. If this option is disabled, the system assumes that the grip action is always successful.

Default Re-grip station: Here, the location name of the default re-grip station can be specified. This re-grip station will be used whenever re-gripping is needed for a labware transfer (see Introduction|Regrip) and no re-grip station is specified by any involved location or the labware.

Default force: The default gripping force (in percent) to be used when gripping force is not limited by the labware via the Force custom attributes.

Minimum force: The lower limit for gripping force reduction. This setting can prevent that a too low gripping force is used. A too low gripping force can lead to erroneous object detection and loss of object while moving.

Maximum speed: The upper limit for the movement speed for plate transfers.

11.4.2 Execute Single Vector (RGA)

The **Execute Single Vector** command executes a vector (list of 3D way-points).

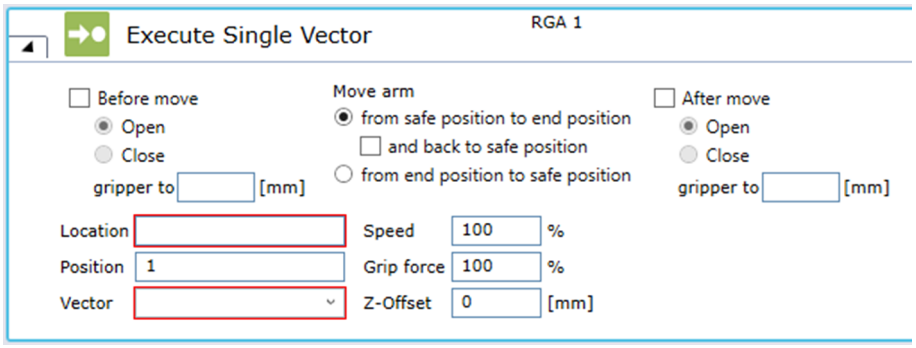


Fig. 211: Execute Single Vector

Move arm

Specify the direction of the move. Either select the **from safe position to end position** option button—with the possibility to select the **and back to safe position** check box—or select the **from end position to safe position** option button.

If the option to **and back to the safe position** of the vector is selected, the vector is now executed in reverse order. If it fails, the command is aborted with an error.

Additionally, you can select the **Before move** check box and the **After move** check box:

Before move

To execute a gripper action before the move check **Before move**.
If the **Close** option button is selected, no object is detected by the gripper and grip detection is activated, the command is aborted with an error. Then, the selected vector is executed for the specified site.
If the **from end position to safe position** option button is selected, the vector is executed in reverse. If it fails, the command is aborted with an error.

After move

If a gripper action is selected after the move, it is executed after this move. If the **Close** option button is selected, grip detection is activated and no object is detected by the gripper.

If **and back to safe position** is not selected, error handling shall be performed.

If **and back to safe position** is selected, error handling shall be performed.

Both before and after move, an action for the gripper has to be selected. Options are **Open** and **Close**. By default, neither **After Move** nor **Open** or **Close** are selected. If **Open** or **Close** are selected, you need to specify a gripper width.

Parameters

Location and **Position**: Specify the name of a location and a site number. This can be done by clicking a carrier, site or labware on the worktable.

Vector: Select a vector name from the **Vector** box. The list contains the names of all vectors for the current robot stored in the selected carrier.

Speed: Specify the move speed in percent (1–100%).

Grip force: Specify the gripper force in percent (1–100%). All edit fields support variables and expressions.

11.4.3 Transfer Labware (RGA)

Use this command together with an RGA to transfer labware objects or labware-like carriers (i.e., less than 400 g) to a location.

Carriers

When transferring carriers, the carrier name has to be typed in manually. The carrier needs CustomAttribute **MoveableCarrier** set to true and to specify CustomAttributes **Grip_Narrow** and **Grip_Wide**.
24 Eppendorf Adapter is a moveable carrier and can be used as example.

Pay close attention to copy and pasting lines before and after Transfer Labware commands. For details, see the Labware rotation and worktable simulation topic.

You can specify the labware to be transferred by the RGA. This can be done by entering the labware name or an expression which resolves into a labware name into the **Labware** box or by selecting labware on the worktable.

You can select from the following, mutual exclusive options:

- **Move to location**: You can specify the location the labware will be transferred to.
  - By default the system will place the labware on the first free site of the target location. The user can override this behavior with a check box and specify a dedicated site on the target location. The target site can be specified by entering the site number or an expression which resolves into a site number into a text box or by selecting a carrier or site on the worktable.
- **Move to base**: The system will move the labware to its base. If this option is selected, the UI controls for location and site are dimmed.
  - By default the system will place the labware on the first free site of the target location. The user can override this behavior with a check box and specify a dedicated site on the target location.

Context check

The system will check if the move is possible (labware exists, location/target site is available) and raise context check messages if an error is detected.

Move Simulation

The system will simulate the move in Move Simulation mode and update the worktable (move the labware to the destination) if the move is possible.

Example

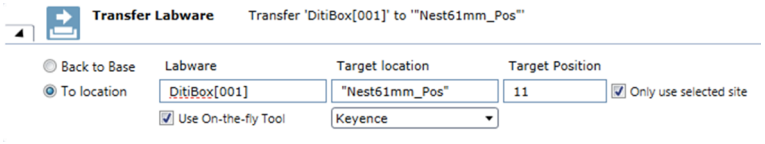


Fig. 212: Transfer Labware

Parameters

Labware: In the Labware box, either type your expression or click the labware displayed in the **Worktable** editor window (Worktable Click).

To location: Click **To location** to specify the location the labware should be moved to. In the **Target location** box, type the location name (or an expression which resolves into a location name) or select a carrier or site on the worktable.

Back to Base: Select **Back to Base**, to move the labware to its base—i.e., the original position where the labware was placed in the initial workspace. If the **Back to Base** option is selected, **Target location** is set to "<BASE>" and appears dimmed. **Target Position** appears dimmed also.

Target location: Select a target location for used DiTi racks in the list box. The usage of "<BASE>" as an alias for the base location is supported.

Target Position: The default is to place the labware on the first free site of the target location.

Only use selected site: To change the default, select the **Only use selected site** check box and specify a dedicated site on the target location.

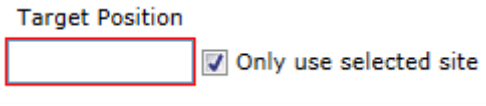


Fig. 213: Target Position

To specify the target site, click in the **Target Position** box and type the site number (or an expression which resolves into a site number) or select a carrier or site on the worktable.

Use On-the-fly Tool: Select this check box to use a barcode reader as on-the-fly tool. The barcode reader must be available on the worktable, otherwise this field is not visible.

The vector, which is used to move the labware to the barcode reader, is selected depending on the grip mode that is used for the transfer from the source to the destination of the transfer.

There is also the possibility to move the RGA while the barcode scanner is scanning the labware. In order to activate this functionality an additional vector must be created in the carrier of the barcode scanner.
The name of these custom vectors must begin with "OnTheFly" and contain the grip mode for which it shall be used (e.g., "OnTheFlyNarrow"). The first point of

such a vector should be identical with the end position of the normal vector for the barcode scanner. When the scan is executed and the corresponding custom vector is present, the vector is moved from start to end position and back again. The speed of the movement can be reduced via the custom attribute **OnTheFlySpeed**. Here a speed factor can be set (1–100%).

Labware rotation and worktable simulation

FluentControl tracks labware rotation. It also simulates the worktable as the program is being written. Both of these features make it very easy to see what will happen at run time when the script/method is executed.

Pay close attention to copy and pasting lines before and after **Transfer Labware** commands. If a labware is rotated after the pipetting commands are written, it may impact the well selections. After you expand the inserted command once, the values get updated as expected.

Because FluentControl is an open system, during method development one must always verify that the intended wells are correctly pipetted. For more information on validation of methods, see “Validation of FluentControl applications” in Help.

11.4.4 Replace Used DiTi Racks (RGA)

Use this command to replace empty DiTi racks on the worktable with unused DiTi racks.



Fig. 214: Replace used DiTi Racks

Parameters

DiTi Type Filter: Select the disposable tip (DiTi) type to replace with from the list box or type an expression to specify the DiTi type.

Labware Name Filter: Select a labware from the list of all available labware names or by clicking it in the Worktable editor. Alternatively, you can type an expression to specify the labware name filter.

Target location: Select a target location for used DiTi racks in the list box. The usage of **<BASE>** as an alias for the base location is supported. Note that in contrast to other commands like Transfer Labware, the base position information is ignored if the target location is the base location of the DiTi rack, i.e. the first free position will be used to store the DiTi rack.

Only remove: To remove the DiTi rack without replacement, select the **Only remove** check box.

11.4.5 Move Arm (RGA)

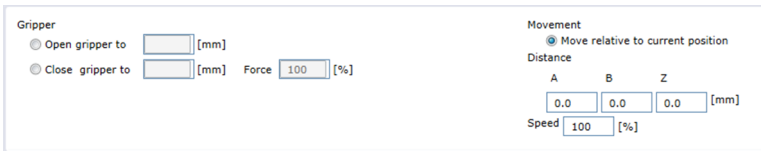


Fig. 215: Gripper

Use this command to open or to close the gripper fingers, or make a relative move of the gripper to its current position.

You can select from one of the following options:

Gripper

- **Open gripper to**: Gripper width in mm.
- **Close gripper to**: Gripper width in mm.
- **Force**: Grip force in percent (range: 1–100%).

Movement

- **Move**: Move relative to current position
- **Distance:**
  - Relative distance to move in **A** (X-direction in gripper coordinate system) in mm. Floating point value.
  - Relative distance to move in **B** (Y-direction in gripper coordinate system) in mm. Floating point value.
  - Relative distance to move in **Z** in mm. Floating point value.
- **Speed**: Speed in percent (range: 1–100%). Integer value.

11.4.6 Get Fingers (RGA)



Fig. 216: Get Fingers (RGA)

Can only be used by RGA with Finger Exchange System (FES). Use this command to pick up RGA FES fingers.

Parameters

Gripper Fingers: Select RGA FES gripper fingers to be picked up.

11.4.7 Drop Fingers (RGA)

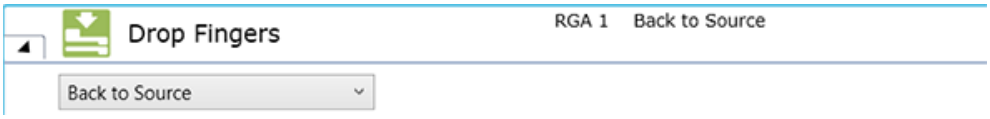


Fig. 217: Drop Fingers (RGA)

Can only be used by RGA with Finger Exchange System (FES). Use this command to place RGA FES fingers at a specified position on the deck of the Fluent.

Parameters

Back to source: By selecting this option the RGA FES fingers are returned to the position they were fetched from.

Back to position: By selecting this option the user can specify a return position for the RGA FES fingers.

11.5 PreciseFlex 400

11.5.1 Configuring FluentControl and PreciseFlex (Precise)

1. Power on and connect to the PreciseFlex WebInterface.

1. Click on 'Admin'.

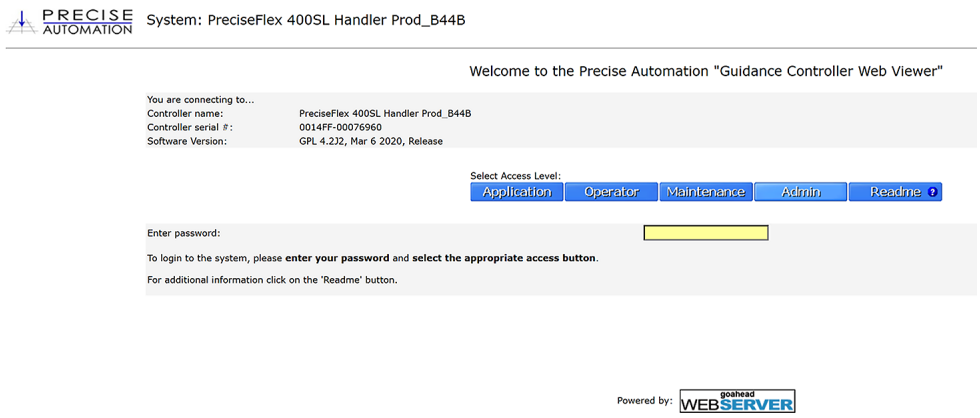


Fig. 218: PreciseFlex 400SL Handler

2. If this has not been done yet, go to 'Setup' > 'First Time Setup' and configure the IP address, e.g. 192.168.0.2, then load the 'TCP Server' GPL onto the internal flash drive.

   If you have a rail, be aware that you have to configure the orientation in which the robot is attached to the rail. (See the PreciseFlex manual.)

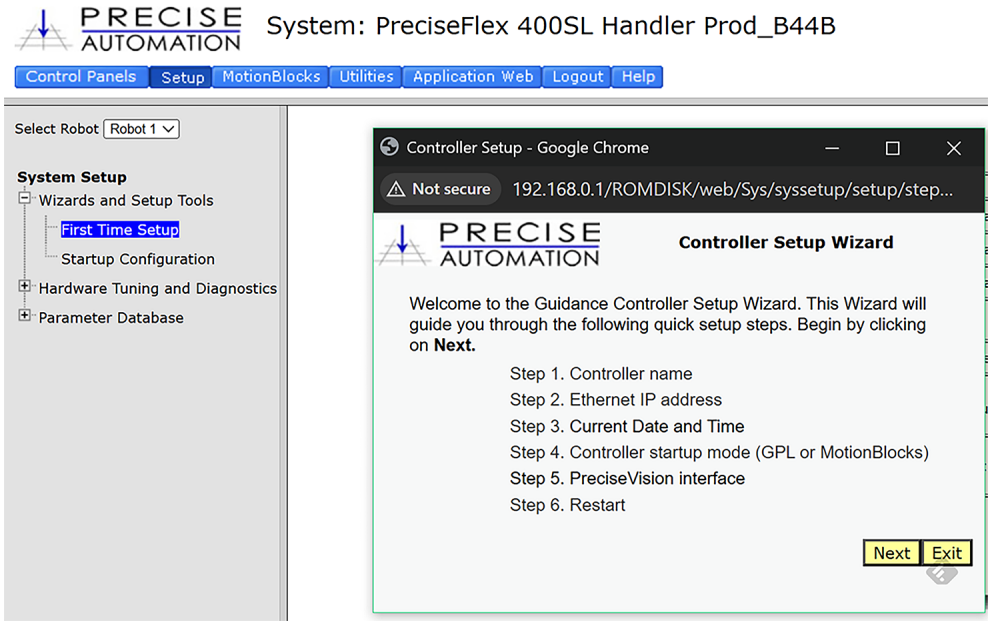


Fig. 219: First Time Setup

3. Go to 'Setup' and set the checkbox for automatic run at startup.

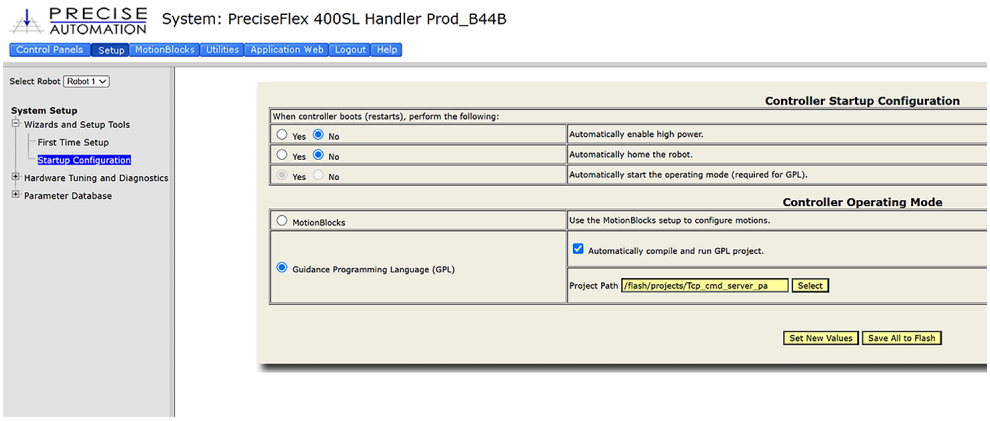


Fig. 220: Startup Configuration

4. Go to 'Control Panel' > 'Operator Control Panel' and check if the GPL project is loaded and started. If it is not loaded, you can load it by clicking on 'Load', and then click on 'Start' to run the GPL project.

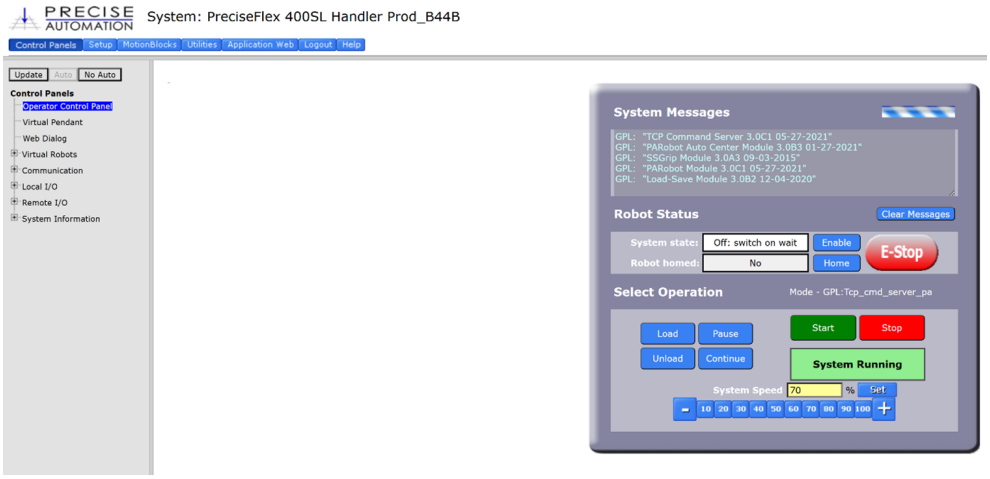


Fig. 221: Operator Control Panel

2. Start FluentControl and open Configuration

1. Go to 'AvailableDrivers'.

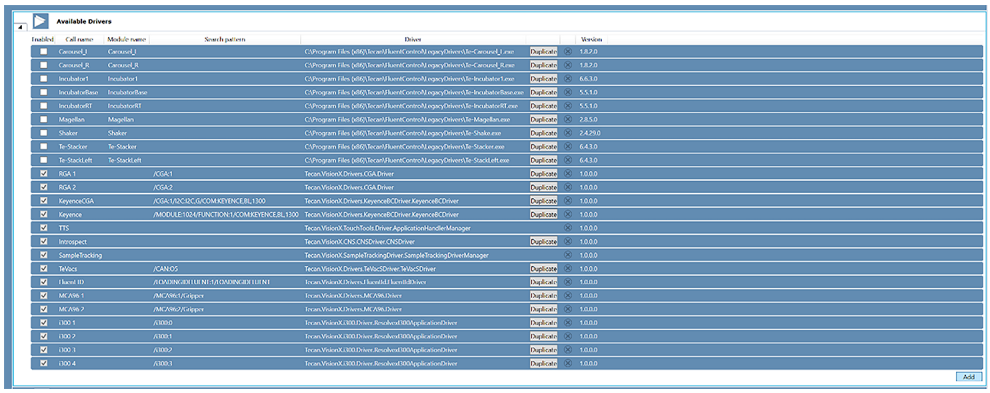


Fig. 222: Available Drivers

2. Click 'Add' to create a new entry.
3. Enter a 'CallName' of your choice. The 'SearchPattern' needs to be '/precise:1' and you need to select the PreciseFlex driver from the drop-down menu.

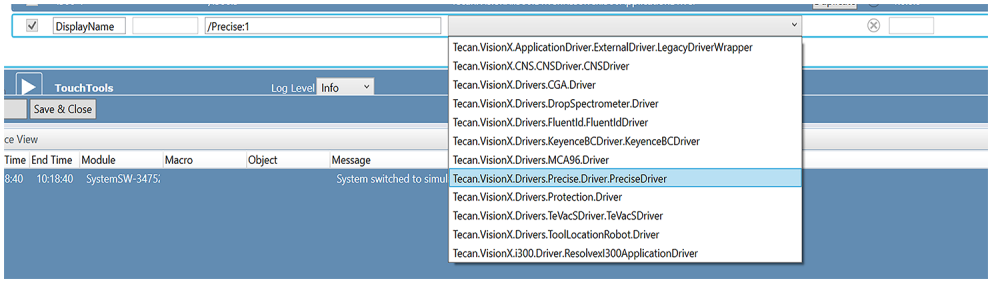


Fig. 223: Select Driver

4. Save and restart FluentControl.
5. Open 'Configuration' and expand PreciseDriver configuration.

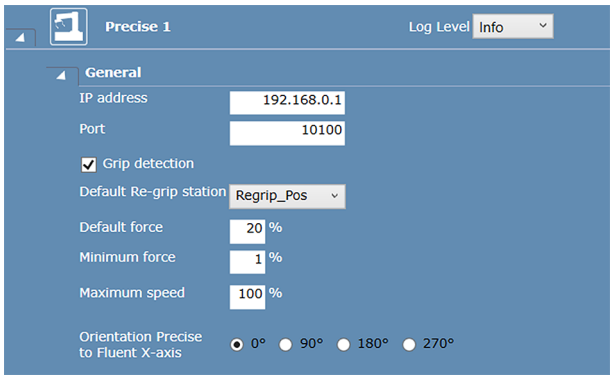


Fig. 224: Configuration

6. Enter the IP address of the Precise PF400.
7. The default port is 10100.
8. Select the orientation between the FluentControl coordinate system and the Precise PF400 coordinate system.

IP address: The IPv4 Network address of the robot.

Port: The Network port specified for the robot. Default is port 10100.

Grip detection: If enabled, the system will check for each plate gripping action whether or not an object was detected by the hardware and trigger the appropriate error handling. If this option is disabled, the system assumes that the grip action is always successful.

Default Re-grip station: Here, the location name of the default re-grip station can be specified. This re-grip station will be used whenever re-gripping is needed for a labware transfer (see Introduction|Regrip) and no re-grip station is specified by any involved location or the labware.

Default force: The default gripping force (in percent) to be used when gripping force is not limited by the labware via the Force custom attributes.

Minimum force: The lower limit for gripping force reduction. This setting can prevent that a too low gripping force is used. A too low gripping force can lead to erroneous object detection and loss of object while moving.

Maximum speed: The upper limit for the movement speed for plate transfers. Orientation Precise to Fluent X-axis: Defines the orientation of the coordinate system between Fluent and Precise. Orientation Precise to Fluent X-axis: Defines the orientation of the coordinate system between Fluent and Precise.

11.5.2 Execute Single Vector (Precise)

Use this command to execute a vector (list of 3D waypoints) with the Precise.

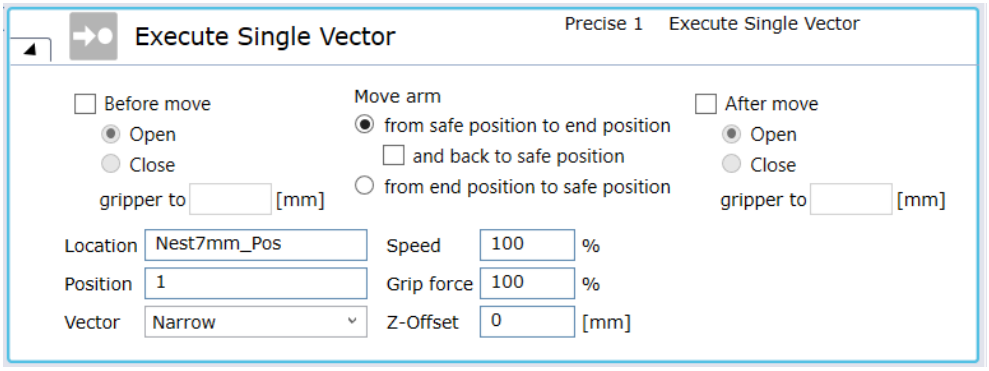


Fig. 225: Executive Single Vector

Move Arm

This option allows the user to specify the movement direction of the selected vector which will be executed. The following directions can be selected:

- From safe position to end position (with the possibility to move back to the safe position)
- From end position to safe position

If the option **From safe position to end position** is selected, the arm will execute the movement vector like it is defined in the selected Location (from top to bottom). Selecting the option **and back to safe position** will execute the vector again in reverse order (e.g. to grip an object and move out again). If the option **From end position to safe position** is selected, the arm will only execute the movement

vector in reverse order. In addition to the execution of the movement vector, the grippers can be opened and closed. This can be done before and/or after the move. The following options can be selected:

Before Move

This option will open or close the gripper to a specified width, in millimeters, before the move vector is executed. The Close option will use grip detection (if it is enabled in the Configure System menu) to detect a labware item in between the grippers. If the gripper reaches the specified position without detecting a labware item, an error is raised. The Open option is executed without grip detection.

After Move

This option will open or close the gripper to a specified width, in millimeters, after the move vector is executed. The **Close** option will enable a grip detection to detect a labware item in between the grippers. If the gripper reaches the specified position without detecting a labware item, an error is raised. The Open option is executed without grip detection. By default, neither **Before Move** nor **After Move** is selected. This setup which will only execute the selected movement vector without moving the gripper.

Parameters

Location and Position: Specify the name of a location and a site number for which the vector shall be executed. This can be done by clicking a carrier, site or labware on the worktable.

Vector: Select a vector name from the Vector box. The list contains the names of all vectors for the current robot stored in the selected carrier.

Speed: Specify the move speed in percent (1–100%).

Grip force: Specify the gripper force in percent (1–100%).

All edit fields support variables and expressions.

11.5.3 Transfer Labware (Precise)

Use this command together with an RGA to transfer labware objects or labware-like carriers (i.e., less than 400 g) to a location.

Carriers

When transferring carriers, the carrier name has to be typed in manually. The carrier needs CustomAttribute **MoveableCarrier** set to true and to specify CustomAttributes **Grip_Narrow** and **Grip_Wide**.
24 Eppendorf Adapter is a moveable carrier and can be used as example.

Pay close attention to copy and pasting lines before and after Transfer Labware commands. For details, see the Labware rotation and worktable simulation topic.

You can specify the labware to be transferred by the RGA. This can be done by entering the labware name or an expression which resolves into a labware name into the **Labware** box or by selecting labware on the worktable.

You can select from the following, mutual exclusive options:

- **Move to location**: You can specify the location the labware will be transferred to.

  - By default the system will place the labware on the first free site of the target location. The user can override this behavior with a check box and specify a dedicated site on the target location. The target site can be specified by entering the site number or an expression which resolves into a site number into a text box or by selecting a carrier or site on the worktable.

- **Move to base**: The system will move the labware to its base. If this option is selected, the UI controls for location and site are dimmed.
  - By default the system will place the labware on the first free site of the target location. The user can override this behavior with a check box and specify a dedicated site on the target location.

Context check

The system will check if the move is possible (labware exists, location/target site is available) and raise context check messages if an error is detected.

Move Simulation

The system will simulate the move in Move Simulation mode and update the worktable (move the labware to the destination) if the move is possible.

Example

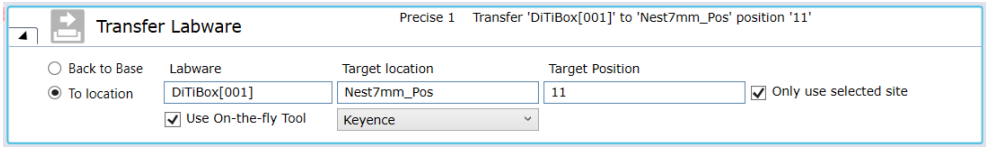


Fig. 226: Transfer Labware

Parameters

Labware: In the Labware box, either type your expression or click the labware displayed in the **Worktable** editor window (Worktable Click).

To location: Click **To location** to specify the location the labware should be moved to. In the **Target location** box, type the location name (or an expression which resolves into a location name) or select a carrier or site on the worktable.

Back to Base: Select **Back to Base**, to move the labware to its base—i.e., the original position where the labware was placed in the initial workspace. If the **Back to Base** option is selected, **Target location** is set to "<BASE>" and appears dimmed. **Target Position** appears dimmed also.

Target location: Select a target location for used DiTi racks in the list box. The usage of "<BASE>" as an alias for the base location is supported.

Target Position: The default is to place the labware on the first free site of the target location.

Only use selected site: To change the default, select the **Only use selected site** check box and specify a dedicated site on the target location.

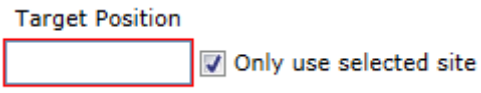


Fig. 227: Target Position

To specify the target site, click in the **Target Position** box and type the site number (or an expression which resolves into a site number) or select a carrier or site on the worktable.

Use On-the-fly Tool: Select this check box to use a barcode reader as on-the-fly tool. The barcode reader must be available on the worktable, otherwise this field is not visible.

The vector, which is used to move the labware to the barcode reader, is selected depending on the grip mode that is used for the transfer from the source to the destination of the transfer.

There is also the possibility to move the Precise while the barcode scanner is scanning the labware. In order to activate this functionality an additional vector must be created in the carrier of the barcode scanner.
The name of these custom vectors must begin with "OnTheFly" and contain the grip mode for which it shall be used (e.g., "OnTheFlyNarrow"). The first point of such a vector should be identical with the end position of the normal vector for the barcode scanner. When the scan is executed and the corresponding custom vector is present, the vector is moved from start to end position and back again. The speed of the movement can be reduced via the custom attribute **OnTheFlySpeed**. Here a speed factor can be set (1–100%).

Labware rotation and worktable simulation

FluentControl tracks labware rotation. It also simulates the worktable as the program is being written. Both of these features make it very easy to see what will happen at run time when the script/method is executed.

Pay close attention to copy and pasting lines before and after **Transfer Labware** commands. If a labware is rotated after the pipetting commands are written, it may impact the well selections. After you expand the inserted command once, the values get updated as expected.

Because FluentControl is an open system, during method development one must always verify that the intended wells are correctly pipetted. For more information on validation of methods, see “Validation of FluentControl applications” in Help.

11.5.4 Replace Used DiTi Racks (Precise)

Use this command to replace empty DiTi racks on the worktable with unused DiTi racks.



Fig. 228: Replace used DiTi Racks

Parameters

DiTi Type Filter: Select the disposable tip (DiTi) type to replace with from the list box or type an expression to specify the DiTi type.

Labware Name Filter: Select a labware from the list of all available labware names or by clicking it in the Worktable editor. Alternatively, you can type an expression to specify the labware name filter.

Target location: Select a target location for used DiTi racks in the list box. The usage of **<BASE>** as an alias for the base location is supported. Note that in contrast to other commands like Transfer Labware, the base position information is ignored if the target location is the base location of the DiTi rack, i.e. the first free position will be used to store the DiTi rack.

Only remove: To remove the DiTi rack without replacement, select the **Only remove** check box.

11.5.5 Move Arm (Precise)

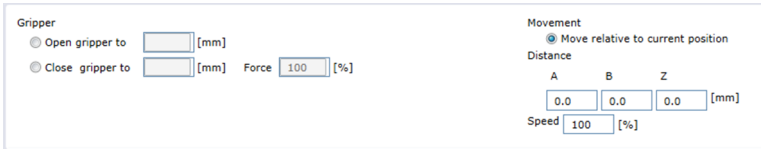


Fig. 229: Gripper

Use this command to open or to close the gripper fingers, or make a relative move of the gripper to its current position.

You can select from one of the following options:

Gripper

- **Open gripper to**: Gripper width in mm.
- **Close gripper to**: Gripper width in mm.
- **Force**: Grip force in percent (range: 1–100%).

Movement

- **Move**: Move relative to current position
- **Distance:**
  - Relative distance to move in **A** (X-direction in gripper coordinate system) in mm. Floating point value.
  - Relative distance to move in **B** (Y-direction in gripper coordinate system) in mm. Floating point value.
  - Relative distance to move in **Z** in mm. Floating point value.
- **Speed**: Speed in percent (range: 1–100%). Integer value.

11.6 MCA384

The MCA384 is an air displacement pipetting system.

The 384-channel pipetting head features a broad volume range with disposable or fixed tips:

- 0.25 µl to 125 µl in the 384-well format
- 0.5 µl to 500 µl in the 96-well format

The washable fixed tips and the disposable tips can be interchanged during a run. The 384-channel pipetting head can be used to pipette with 384, 96, 32, 24, 16, 12 or 8 disposable tips. The head can rapidly change during a run between DiTis and fixed tips by picking up an appropriate adapter.

384 plungers, which are moved simultaneously, are used to vary the volume of the cylinders. When the volume expands causing an underpressure, the liquid is aspirated into the tips. By decreasing the volume causing an overpressure, the liquid can be dispensed all at once or in steps. By means of a control software, the movement of the plungers is controlled very accurately. The channels which are used for pipetting are defined by the type of adapter and / or the DiTis picked up (384, 96, 32, 24, 16, 12 or 8). The plungers are always separated from the liquid to be handled by an air gap. Therefore, only the tips and the inner part of the fixed tip adapter are in contact with the liquid. Other parts of the pipetting head are not exposed to possible contamination.

11.6.1 Transfer Liquid (MCA384)

The Transfer Liquid smart command can be used to transfer a specific number of samples or reagent volume from a source to a target labware using the MCA 384. Instead of a series of Pick Up Tips, Aspirate, Dispense and Set Tips Back commands, the smart command automatically creates the required command sequence at runtime. Depending on the number of samples, the command will get the required number of columns or rows of tips to perform the action and will also perform re-racking of tips if needed.

The Transfer Liquid smart command can only be used if an EVA adapter is mounted on the MCA384. It is currently not compatible with any other adapters.

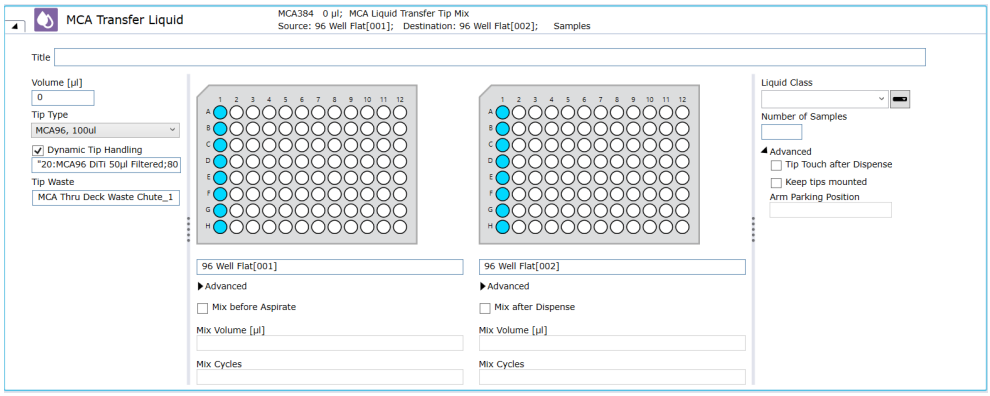


Fig. 230: User interface of 'Transfer Liquid' MCA smart command

Parameters

The **Transfer Liquid** script command provides the following parameters for the configuration of the pipetting automation:

- **Volume [µl]**: Define the volume to be transferred to each target well. If this volume is larger than the tip capacity or the syringe volume, the system will transfer the volume in multiple steps at run-time.
- **Tip Type**: Define the type of disposable tip that will be used for pipetting. Can be selected by using the dropdown menu or by clicking on the worktable.
- **Dynamic Tip Handling**: Dynamic Tip Handling is designed to use different tip types depending on the aspirate volume. If you enter in the Dynamic Tip Handling box – e.g.:

  `20:MCA96 DiTi 50µl Filtered;80:MCA96 DiTi 150µl` Filtered;

  this means that:

- DiTi 50 µl Filtered are for volumes up to 20 µl,
- DiTi 150 µl Filtered are used for volumes > 20 to 180 µl, and
- for larger volumes the tip type selected in the Tip Type dropdown menu is used.

- **Tip Waste**: Define the waste for discarding the disposable tips after pipetting.
- **Aspiration position**: Define where the aspirate is performed by defining the source labware on the worktable, well selection, row offset or column offset. There can be only row or column offset, not both
- **Mix before Aspirate**: Define the settings that will be transferred into the Liquid Class MicroScript to perform a mix step before aspiration: activate or deactivate mixing (MixOnOff), mixing volume (MixVolume) and number of mixing cycles (MixingCycles).
- **Dispense position**: Define where the dispense is performed by defining the destination labware on the worktable, well selection, row offset or column offset. There can be only row or column offset, not both.
- **Mix after Dispense**: Define the settings that will be transferred into the Liquid Class MicroScript to perform a mix step after dispense: An option to activate the mixing (MixOnOff), mixing volume (MixVolume) and number of mixing cycles (MixingCycles).
- **Liquid Class**: The Liquid class serves as template and needs to be validated in the final application.The Liquid Class can be defined by selecting a liquid class from the dropdown menu or alternatively by clicking on the right icon next to the dropdown menu, the input field can be changed to define the liquid class as variable. The default liquid class contains both options to mix before aspirate and after the dispense, as well as the option for a tip touch.
- **Number of Samples**: Number of target wells to be pipetted. By design only full columns or rows of tips can be picked-up by the MCA, hence this number will be rounded up automatically to a multiple of 8 if pipetting column-wise or a multiple of 12 if pipetting row-wise
- **Tip Touch after Dispense**: Define whether a ‘Tip Touch’, i.e., an additional move to the liquid level to remove any potential remaining droplets from the tips, is performed after the dispense.
- **Keep tips mounted**: When this option is checked, the tips will not be discarded after the command execution finishes. Requires that the Arm Parking Position is defined.
- **Arm Parking Position**: Define a labware above which the arm shall be parked if Keep tips mounted is selected.

The well selection pattern for source and destination must match. There can be either a column offset or a row offset, not both. If number of samples is defined, the well selection plus offset define only the start well.

11.6.2 Get Head Adapter (MCA384)

Use this command to pick up a head adapter for the MCA384 from a specified position on the worktable of the pipetting instrument.

DiTi head adapters can be dropped and picked up with DiTis already mounted. If the target adapter is already mounted, the command is skipped.

Parameters

Head Adapter: Select the available labware on the worktable.

Air gap volume [µl]: The volume for the air gap in micro liter. The air gap is only aspirated if an adapter with fixed tips is used, or if the adapter already had disposable tips mounted prior to being picked up by the head.

11.6.3 Drop Head Adapter (MCA384)

Use this command to put down the adapter of the MCA384 at a specified position on the worktable, and optionally to get another adapter after the drop.

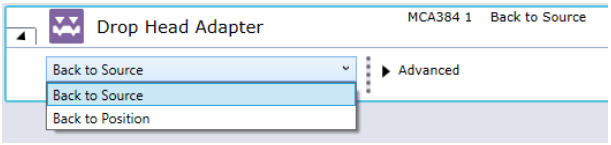


Fig. 231: Drop Head Adapter: Back to Source

Back to Source: When the **Back to Source** option is selected, the adapter is returned back to the grid and site position from where they were fetched with the MCA384 **Get Head Adapter** command.

Back to Position: When the Back to Position option is selected, the Head Adapter box appears. You can specify the position (labware) where the MCA should return the adapter.

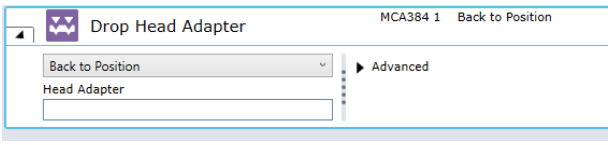


Fig. 232: Drop Head Adapter: Back to Position

Be sure the labware is empty to receive the adapter.

Advanced Parameters

Get head adapter after drop: Check this check box if you want to get another adapter after dropping the one which is currently fitted.

Air gap volume [µl]: The volume for the air gap in micro liters.

11.6.4 Pick Up Tips (MCA384)

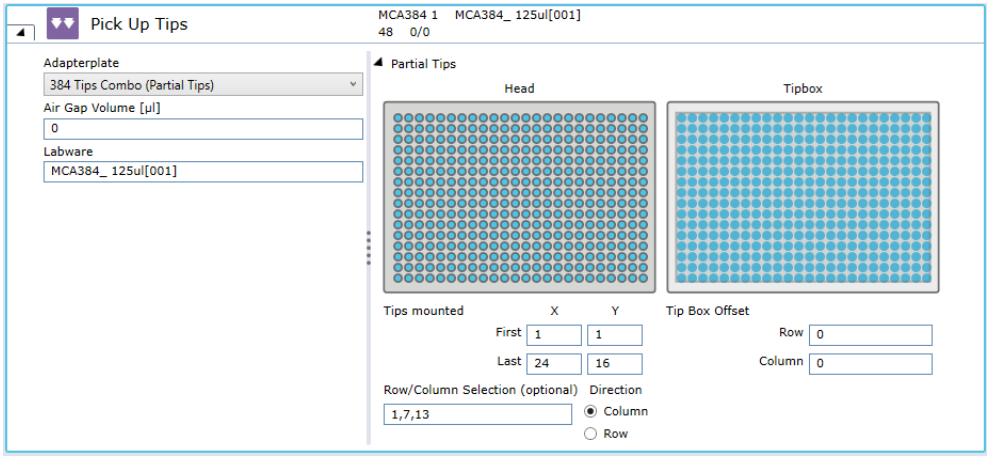


Fig. 233: Pick Up Tips (MCA384)

Use this command to pick up disposable tips for the MCA384 from a specified position on the worktable.

Parameters

Adapterplate: Click the **Adapterplate** arrow to see more options. Select the required adapter plate from the list.

Air gap volume [µl]: The volume for the air gap in microliters.

Labware: In the **Labware** box, either type your expression or click the labware displayed in the **Worktable** editor window (*Worktable click*).

Partial Tips

Tips mounted

First X: Define the first tip position on the head in X-direction where the tips are mounted.

First Y: Define the first tip position on the head in Y-direction where the tips are mounted.

Last X: Define the last tip position on the head in X-direction where the tips are mounted.

Last Y: Define the last tip position on the head in Y-direction where the tips are mounted.

Tip Box offset

Row: Enter the offset of the tips in rows on the tip box. You can move from one tip tray to another using this parameter.

Column: Enter the offset of the tips in columns on the tip box. It is possible to move from one tip tray to another using this parameter.

Row/Column Selection (optional): Allows working with variable number of columns/rows of MCA384 tips for a tip box in the active DiTi carrier with the MCA384 combo adapter from a tip-box that has been appropriately reracked (i.e., contains only the tips to be mounted) outside the active carrier. You can set the

Direction by selecting **Column** or **Row** (selection must be consistent between Get Tips or Pick Up Tips, all Aspirate/Mix/Dispense commands and the subsequent Drop Tips or Set Tips Back command—this is enforced by context check).

11.6.5 Set Tips Back (MCA384)

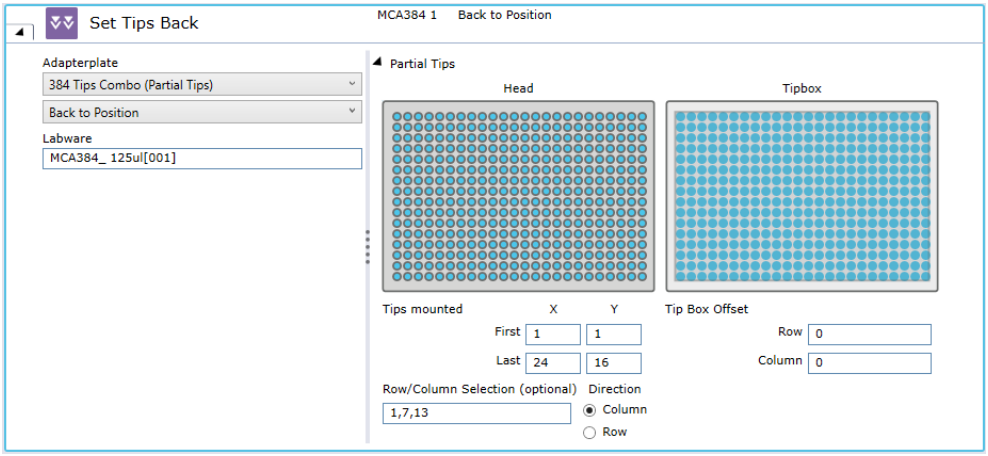


Fig. 234: Set Tips Back

Use this command to drop the disposable tips of the MCA384 to a specified position on the worktable.

Back to Source

Adapterplate: Click the **Adapterplate** arrow to see more options. Select the required adapter plate from the list.

Back to Position

Labware: In the **Labware** box, either type your expression or click the labware displayed in the **Worktable** editor window (*Worktable click*).

Partial Tips

Tips mounted

First X: Define the last tip position on the head in x direction where the tips are mounted.

Last Y: Define the last tip position on the head in y direction where the tips are mounted.

Tip Box offset

Row: Enter the offset of the tips in rows on the tip box. You can move from one tip tray to another using this parameter.

Column: Enter the offset of the tips in columns on the tip box. It is possible to move from one tip tray to another using this parameter.

Row/Column Selection (optional): Allows working with variable number of columns/rows of MCA384 tips for a tip box in the active DiTi carrier with the MCA384 combo adapter from a tip-box that has been appropriately reracked (i.e.,

contains only the tips to be mounted) outside the active carrier. You can set the **Direction** by selecting **Column** or **Row** (selection must be consistent between Get Tips or Pick Up Tips, all Aspirate/Mix/Dispense commands and the subsequent Drop Tips or Set Tips Back command—this is enforced by context check).

11.6.6 Get Tips (MCA384)

The MCA384 Get Tips command is used to pick up disposable tips for the MCA384 arm with the specified DiTi type on the worktable.

It keeps track of tip positions and automatically picks up the next available and accessible unused tips of the specified type.

Physically inaccessible tip boxes are not considered to pickup tips from.

The order of a list of available tip boxes goes by name in alphabetical order (“MCA96_100ul Filtered_Box[001]”, “MyBox[001]”, “NextBox[001]”), then by index if the names are the same (“MCA96_100ul Filtered_Box[001]”, “MCA96_100ul Filtered_Box[002]”). Physically inaccessible tip boxes are not considered to pickup tips from. Partial tip pickup is supported only for full columns or full rows, not both. For this the First and the Last in X and Y Positions for the channels of the head need to be defined.

If not enough columns or rows of tips are available in one tip box and additional tip boxes with sufficient tips are present on the worktable, the **Get Tips** command automatically executes re-racking of the tips so that the required number of columns or rows can be mounted.



If during the re-racking procedure a partial inaccessibility occurs where tips can't be set back into the target box, no tips will be picked up and an error occurs during runtime. To avoid those errors please validate your application



The re-racking feature is only supported for EVA adapter plates, not for other adapter plates.

Depending on its settings, the **OnEmptyTray** procedure may be called if any empty DiTi rack of the defined type is available.

Parameters

Adapterplate

Select the required adapter plate from the drop-down menu.

Air gap volume [µl]

Defines the volume that is aspirated before mounting any tips.

DiTi Type

Selects the disposable tip type from the dropdown menu or by clicking on the tip box on the worktable.

11.6.7 Drop Tips (MCA384)

Use this command to drop the disposable tips of the multi-channel arm to a specified DiTi Waste.

Adapterplate: Select the appropriate adapter plate by choosing from the drop-down menu or by clicking on the adapter plate on the worktable.

DiTi Waste: Defines the waste where the tips will be discarded.

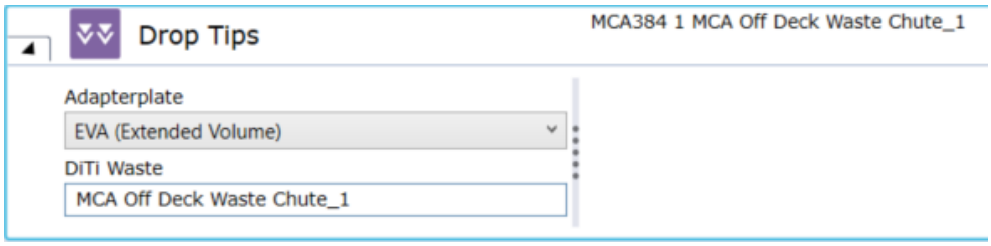


Fig. 235: User interface of 'Drop Tips' command

11.6.8 Aspirate (MCA384)

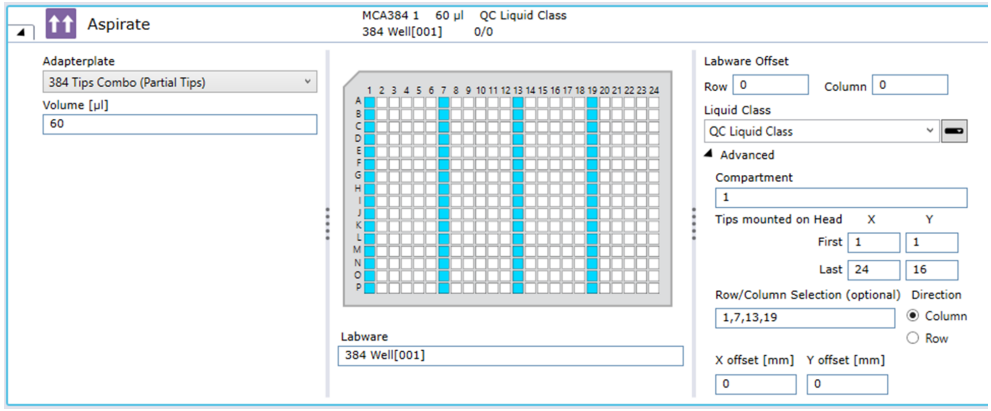


Fig. 236: Aspirate

Use this command to aspirate liquid from specified positions on the worktable with the MCA384.

Parameters

Adapterplate: Clicking the arrow opens the list. In the adapterplate list, click the required item.

Volume: Set the volume for all tips of the MCA. You can specify how many units in microliters (µl) to be pipetted into each well.

Labware: In the Labware box, either type your expression or click the labware displayed in the Worktable editor window (Worktable Click).

Labware Offset: The labware offset parameters describe the offset of the mounted tips relative to the labware:

- Row: **ExpressionInputBox** which describes the offset of the mounted tips in rows relative to the labware. It is possible to move from one plate to another using this parameter.
- Column: **ExpressionInputBox** which describes the offset of the mounted tips in columns relative to the labware. It is possible to move from one plate to another using this parameter.

As default a column is selected.

Liquid Class: Choose the required liquid class from the pull down list or by means of an expression typed in (e.g., by using a variable defined within your script). The list only shows liquid classes that are suitable for the current configuration of the pipetting instrument. For example, you will not see liquid classes for DiTis if the arm is not currently configured with DiTis.

Advanced Parameters

Compartment: Some special labware types (e.g., for protein crystallography) have wells with several compartments (cavities). FluentControl can pipette a different liquid into each of the compartments. Each well typically has one main compartment and one or more satellite compartments which are directly adjacent and are often smaller in size. Choose the compartment which you want to use for pipetting (the main compartment is normally compartment 1). Well compartments are specified in the labware configuration. This field is inactive (gray) if the well only has one compartment.

Tips mounted on Head

First X: Define the first tip position on the head in x direction where the tips are mounted.

First Y: Define the first tip position on the head in y direction where the tips are mounted.

Last X: **ExpressionInputBox** to define the last tip position on the head in x direction where the tips are mounted.

Last Y: **ExpressionInputBox** to define the last tip position on the head in y direction where the tips are mounted.

X offset [mm]: This is the offset of the tip from the center of the target well (negative = left).

Y offset [mm]: This is the offset of the tip from the center of the target well (negative = to front, positive = to back).

Row/Column Selection (optional): Allows working with variable number of columns/rows of MCA384 tips for a tip box in the active DiTi carrier with the MCA384 combo adapter from a tip-box that has been appropriately reracked (i.e., contains only the tips to be mounted) outside the active carrier. You can set the **Direction** by selecting **Column** or **Row** (selection must be consistent between Get Tips or Pick Up Tips, all Aspirate/Mix/Dispense commands and the subsequent Drop Tips or Set Tips Back command—this is enforced by context check).

11.6.9 Dispense (MCA384)

Dispenses liquid into specified labware on the worktable with the MCA384.

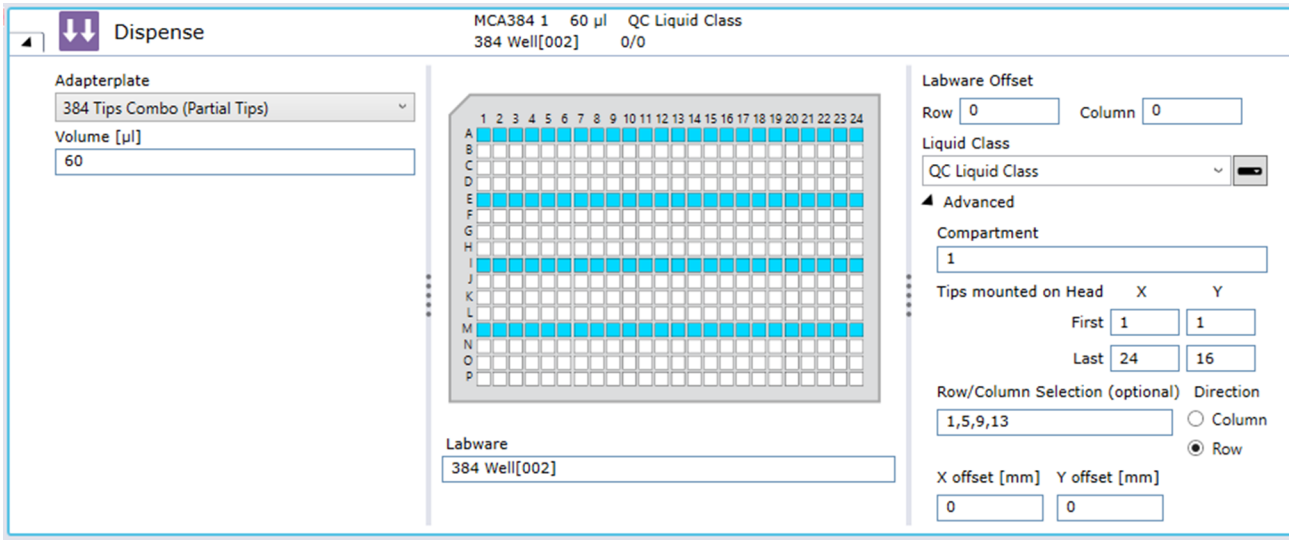


Fig. 237: Dispense

Parameters

Adapterplate: Clicking the arrow opens the list. In the Adapterplate list, click the required item.

Volume: Set the volume for all tips of the MCA. You can specify how many units in microliters (µl) to be pipetted into each well.

Labware: In the **Labware** box, either type your expression or click the labware displayed in the **Worktable** editor window (*Worktable click*).

Labware Offset: The labware offset parameters describe the offset of the mounted tips relative to the labware:

- **Row**: **ExpressionInputBox** which describes the offset of the mounted tips in rows relative to the labware. It is possible to move from one plate to another using this parameter.
- **Column**: **ExpressionInputBox** which describes the offset of the mounted tips in rows relative to the labware. It is possible to move from one plate to another using this parameter.

As default a column is selected.

Liquid Class: Choose the required liquid class from the pull down list or by means of an expression typed in (e.g., by using a variable defined within your script). The list only shows liquid classes that are suitable for the current configuration of the pipetting instrument. For example, you will not see liquid classes for DiTis if the arm is not currently configured with DiTis.

Advanced Parameters

Compartment: Some special labware types (e.g., for protein crystallography) have wells with several compartments (cavities). FluentControl can pipette a different liquid into each of the compartments. Each well typically has one main compartment and one or more satellite compartments which are directly adjacent and are often smaller in size. Choose the compartment which you want to use for pipetting (the main compartment is normally compartment 1). Well compartments are specified in the labware configuration. This field is inactive (gray) if the well only has one compartment.

Tips mounted on Head

First X: Define the first tip position on the head in x direction where the tips are mounted.

First Y: Define the first tip position on the head in y direction where the tips are mounted.

Last X: **ExpressionInputBox** to define the last tip position on the head in x direction where the tips are mounted.

Last Y: **ExpressionInputBox** to define the last tip position on the head in y direction where the tips are mounted.

X offset [mm]: This is the offset of the tip from the center of the target well (negative = left).

Y offset [mm]: This is the offset of the tip from the center of the target well (Negative = to front, positive = to back).

Row/Column Selection (optional): Allows working with variable number of columns/rows of MCA384 tips for a tip box in the active DiTi carrier with the MCA384 combo adapter from a tip-box that has been appropriately reracked (i.e., contains only the tips to be mounted) outside the active carrier. You can set the **Direction** by selecting **Column** or **Row** (selection must be consistent between Get Tips or Pick Up Tips, all Aspirate/Mix/Dispense commands and the subsequent Drop Tips or Set Tips Back command—this is enforced by context check).

11.6.10 Mix (MCA384)

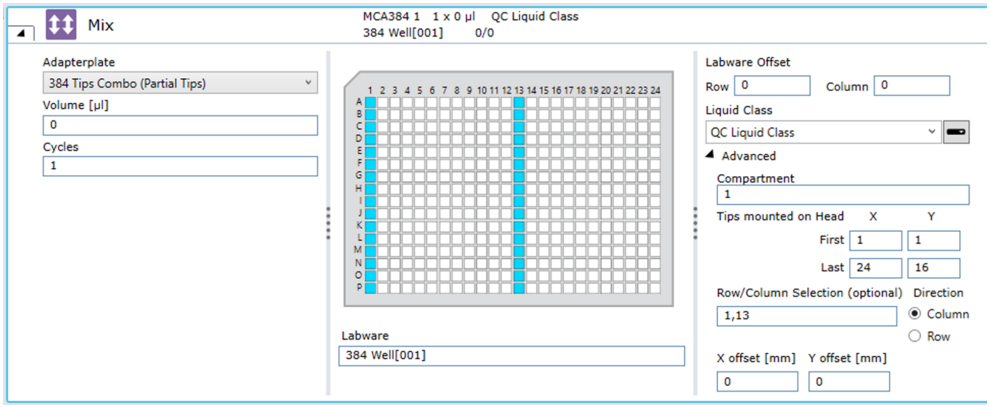


Fig. 238: Mix (MCA384)

Performs a mix operation with the MCA384 (see “MCA384” [▶ 290]) by aspirating and then dispensing at the same position.

Parameters

Adaperplate: Clicking the arrow opens the list. In the adapterplate list, click the required item.

Volume: Set the volume for all tips of the MCA. You can specify how many units in microliters (µl) to be pipetted into each well.

Cycles: Set the number of mixing cycles. You can select existing variables.

Labware: In the **Labware** box, either type your expression or click the labware displayed in the **Worktable** editor window (*Worktable click*).

Labware Offset: The labware offset parameters describe the offset of the mounted tips relative to the labware:

- **Row**: ExpressionInputBox which describes the offset of the mounted tips in rows relative to the labware. It is possible to move from one plate to another using this parameter.
- **Column**: ExpressionInputBox which describes the offset of the mounted tips in columns relative to the labware. It is possible to move from one plate to another using this parameter.

As default a column is selected.

Liquid Class: Choose the required liquid class from the pull down list or by means of an expression typed in (e.g., by using a variable defined within your script). The list only shows liquid classes that are suitable for the current configuration of the pipetting instrument. For example, you will not see liquid classes for DiTis if the arm is not currently configured with DiTis.

Advanced Parameters

Compartment: Some special labware types (e.g., for protein crystallography) have wells with several compartments (cavities). FluentControl can pipette a different liquid into each of the compartments. Each well typically has one main compartment and one or more satellite compartments which are directly adjacent and are often smaller in size. Choose the compartment which you want to use for pipetting (the main compartment is normally compartment 1). Well compartments are specified in the labware configuration. This field is inactive (gray) if the well only has one compartment.

Tips mounted on Head

First X: Define the first tip position on the head in x direction where the tips are mounted.

First Y: Define the first tip position on the head in y direction where the tips are mounted.

Last X: **ExpressionInputBox** to define the last tip position on the head in x direction where the tips are mounted.

Last Y: **ExpressionInputBox** to define the last tip position on the head in y direction where the tips are mounted.

X offset [mm]: This is the offset of the tip from the center of the target well (negative = left).

Y offset [mm]: This is the offset of the tip from the center of the target well (Negative = to front, positive = to back).

Row/Column Selection (optional): Allows working with variable number of columns/rows of MCA384 tips for a tip box in the active DiTi carrier with the MCA384 combo adapter from a tip-box that has been appropriately reracked (i.e., contains only the tips to be mounted) outside the active carrier. You can set the **Direction** by selecting **Column** or **Row** (selection must be consistent between Get Tips or Pick Up Tips, all Aspirate/Mix/Dispense commands and the subsequent Drop Tips or Set Tips Back command—this is enforced by context check).

11.6.11 Wash (MCA384)

Use this command to wash the MCA384 fixed tips using the MCA384 wash station.

Wash Station: Select the wash station.

Liquid Class: Choose the required liquid class from the pull down list or by means of an expression typed in (e.g., by using a variable defined within your script). The list only shows liquid classes that are suitable for the current configuration of the pipetting instrument. For example, you will not see liquid classes for DiTis if the arm is not currently configured with DiTis.

Wash Liquid Selection: Select the wash liquid.

11.6.12 Empty Tip Contents (MCA384)

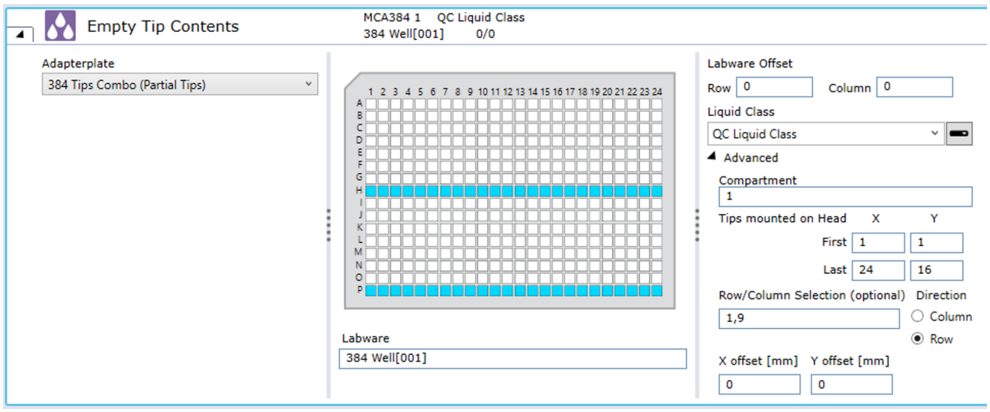


Fig. 239: Empty Tip Contents

Use this command to dispense any liquid currently in the tips to a specified position on the worktable.

Parameters

Adapterplate: Click the **Adapterplate** arrow to see more options. Select the required adapter plate from the list.

Labware Offset: The labware offset parameters describe the offset of the mounted tips relative to the labware:

- **Row**: Enter the offset of the mounted tips in rows relative to the labware. It is possible to move from one plate to another using this parameter.
- **Column**: Enter the offset of the mounted tips in columns relative to the labware. It is possible to move from one plate to another using this parameter.

As default a column is selected.

Labware: In the **Labware** box, either type your expression or click the labware displayed in the **Worktable** editor window (*Worktable click*).

Liquid Class: Choose the required liquid class from the pull down list or by means of an expression typed in (e.g., by using a variable defined within your script). The list only shows liquid classes that are suitable for the current configuration of the pipetting instrument. For example, you will not see liquid classes for DiTis if the arm is not currently configured with DiTis.

Advanced Parameters

Compartment: Some special labware types (e.g., for protein crystallography) have wells with several compartments (cavities). FluentControl can pipette a different liquid into each of the compartments. Each well typically has one main compartment and one or more satellite compartments which are directly adjacent and are often smaller in size. Choose the compartment which you want to use for pipetting (the main compartment is normally compartment 1). Well compartments are specified in the labware configuration. This field is inactive (gray) if the well only has one compartment.

Tips mounted on Head

First X: Define the first tip position on the head in x direction where the tips are mounted.

First Y: Define the first tip position on the head in y direction where the tips are mounted.

Last X: Define the last tip position on the head in x direction where the tips are mounted.

Last Y: Define the last tip position on the head in y direction where the tips are mounted.

X offset [mm]: This is the offset of the tip from the center of the target well (negative = left).

Y offset [mm]: This is the offset of the tip from the center of the target well (Negative = to front, positive = to back).

Row/Column Selection (optional): Allows working with variable number of columns/rows of MCA384 tips for a tip box in the active DiTi carrier with the MCA384 combo adapter from a tip-box that has been appropriately reracked (i.e., contains only the tips to be mounted) outside the active carrier. You can set the **Direction** by selecting **Column** or **Row** (selection must be consistent between Get Tips or Pick Up Tips, all Aspirate/Mix/Dispense commands and the subsequent Drop Tips or Set Tips Back command—this is enforced by context check).

11.6.13 Prime Wash Station (MCA384)

Use this command to make the wash station MCA384 ready for use. The command causes the system to fill up again, i.e., it replenishes the wash liquid station completely.

Parameters

Wash Station: Select the wash station.

Duration [s]: Define the duration of a process in seconds.

Wash Liquid Selection: Select the wash liquid.

11.6.14 Empty Wash Station (MCA384)

Use this command to remove the wash liquid of the wash station MCA384. The pump is running as long as entered in the **Duration** parameter.

Parameters

Wash Station: Select the wash station.

Duration [s]: Define the duration of the wash process in seconds.

11.6.15 Move Arm (MCA384)

Moves the MCA384 from one position to another without performing any other operation

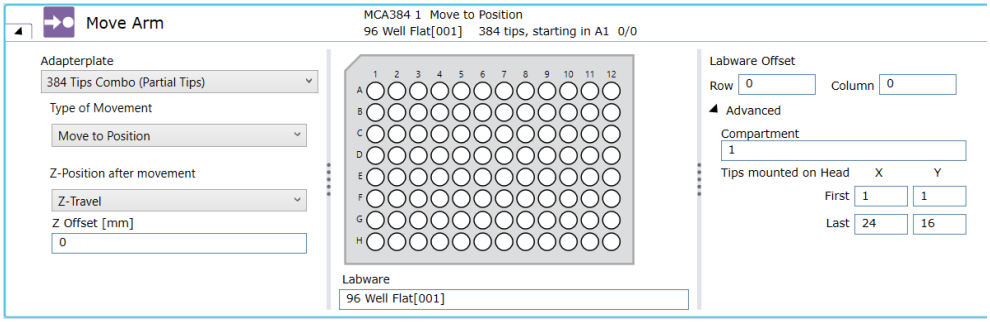


Fig. 240: MCA384: Move Arm

Parameters

Adapterplate: Clicking the arrow opens the list. In the adapterplate list, click the required item.

Type of Movement: List to set the movement type:

- If **Move to Position** or **Move to Position at Global Z Travel** is selected, the system moves the arm to the labware with maximum speed.
- If **Move in X**, **Move in Y** or **Move in Z** is selected, the system uses the speed of the movement value of the parameter **Movement Speed**.
- If **Move to Position** is selected, the system uses the standard movements as determined by PathFinder.
- If **Move to Position at Global Z Travel** is selected, the system first moves the arm in Z to Z travel and then moves in X and Y direction to the target position by a pathfinder controlled movement. Finally, the system moves the arm to the Z position as defined in the **move arm** command.

Site information

Z-Position after movement

Z-Offset [mm]: Defines an offset in millimeters to the Z-Position after movement (positive = higher, negative = lower), if **Move** in Z is selected.

Labware: In the **Labware** box, either type your expression or click the labware displayed in the **Worktable** editor window (*Worktable click*).

Tip Box offset: The tip box offset parameters describes the offset of the tips in the tip box.

Advanced Parameters

Compartment: Some special labware types (e.g., for protein crystallography) have wells with several compartments (cavities). FluentControl can pipette a different liquid into each of the compartments. Each well typically has one main compartment and one or more satellite compartments which are directly adjacent and are often smaller in size. Choose the compartment which you want to use for pipetting (the main compartment is normally compartment 1). Well compartments are specified in the labware configuration. This field is inactive (gray) if the well only has one compartment.

Tips mounted on Head

First X: Define the first tip position on the head in x direction where the tips are mounted.

First Y: Define the first tip position on the head in y direction where the tips are mounted.

Last X: Define the last tip position on the head in x direction where the tips are mounted.

Last Y: Define the last tip position on the head in y direction where the tips are mounted.

11.7 MCA96

The MCA96 is an air displacement pipetting system. The 96-channel pipetting head features a broad volume range of 1 µl to 1000 µl with disposable tips: The 96-channel pipetting head can be used to pipette with 1 to 96 disposable tips. 96 plungers, which are moved simultaneously, are used to vary the volume of the cylinders. When the volume expands, causing an under pressure, the liquid is aspirated into the tips. By decreasing the volume, causing an over pressure, the liquid can be dispensed all at once or in steps. By means of a control software, the movement of the plungers is controlled very accurately. The channels which are used for pipetting are defined by the number of tips picked up. The plungers are always separated from the liquid to be handled by an air gap. Therefore, only the tips are in contact with the liquid. Other parts of the pipetting head are not exposed to possible contamination. The MCA96 also provides cLLD liquid level detection possibility. One cLLD signal is spread across all tips. The first tip which detects a liquid is the liquid volume which is passed to the control software

11.7.1 Transfer Liquid (MCA96)

The Transfer Liquid smart command can be used to transfer a specific number of samples or reagent volume from a source to a target labware using the MCA96 head. Instead of a series of Get Tips, Aspirate, Dispense and Drop Tips commands, the smart command automatically creates the required command sequence at runtime. Depending on the number of samples, the command will get the required number of columns or rows of tips to perform the action and will also perform re-racking of tips if needed.

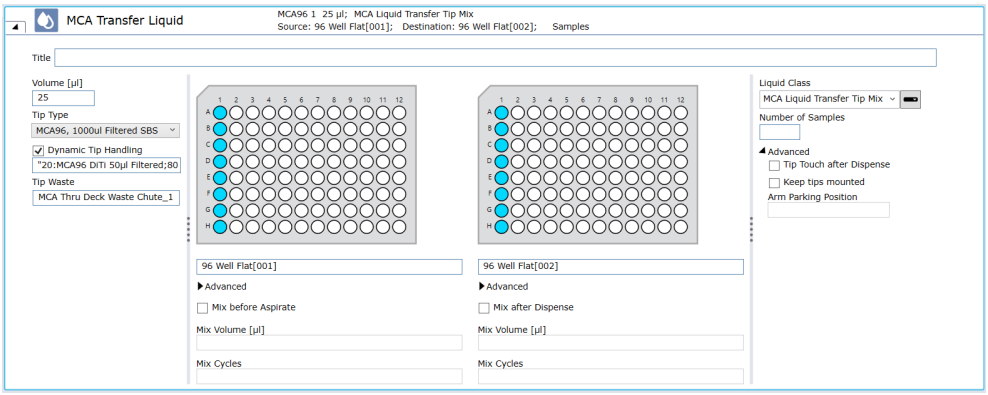


Fig. 241: User interface of 'Transfer Liquid' MCA smart command

Parameters

The Transfer Liquid script command provides the following parameters for the configuration of the pipetting automation:

Volume [µl] Define the volume to be transferred to each target well. If this volume is larger than the tip capacity or the syringe volume, the system will transfer the volume in multiple steps at run-time.

Tip Type Define the type of disposable tip that will be used for pipetting. Can be selected by using the dropdown menu or by clicking on the worktable.

Dynamic Tip Handling: Dynamic Tip Handling is designed to use different tip types depending on the aspirate volume. If you enter in the Dynamic Tip Handling box – e.g.:

```
20:MCA96 DiTi 50µl Filtered;80:MCA96 DiTi 150µl Filtered;
```

this means that:
- DiTi 50 µl Filtered are for volumes up to 20 µl,
- DiTi 150 µl Filtered are used for volumes > 20 to 180 µl, and
- for larger volumes the tip type selected in the Tip Type dropdown menu is used.

Tip Waste Define the waste for discarding the disposable tips after pipetting.

Aspiration position Define where the aspirate is performed by defining the source labware on the worktable, well selection, row offset or column offset. There can be only row or column offset, not both.

Mix before Aspirate Define the settings that will be transferred into the Liquid Class MicroScript to perform a mix step before aspiration: activate or deactivate mixing (MixOnOff), mixing volume (MixVolume) and number of mixing cycles (MixingCycles).

Dispense position Define where the dispense is performed by defining the destination labware on the worktable, well selection, row offset or column offset. There can be only row or column offset, not both.

Mix after Dispense Define the settings that will be transferred into the Liquid Class MicroScript to perform a mix step after dispensing: Activate or deactivate the mixing (MixOnOff), mixing volume (MixVolume) and number of mixing cycles (MixingCycles).

Liquid Class Can be defined by selecting a liquid class from the dropdown menu or alternatively by clicking on the right icon next to the dropdown menu, the input field can be changed to define the liquid class as variable. The default liquid class contains both options to mix before aspirate and after the dispense, as well as the option for a tip touch.



The Liquid class serves as template and needs to be validated in the final application.

Number of Samples Number of target wells to be pipetted. By design only full columns or rows of tips can be picked-up by the MCA, hence this number will be rounded up automatically to a multiple of 8 if pipetting column-wise or a multiple of 12 if pipetting row-wise.

Tip Touch after Dispense Defines whether a ‘Tip Touch’, i.e., an additional move to the side of the well and back before retracting to remove any potential remaining droplets from the tips, is performed after the dispense. The distance of the move is defined within the liquid class itself.

Keep tips mounted When this option is checked, the tips will not be discarded after the command execution finishes. Requires that the Arm Parking Position is defined.

Arm Parking Position Defines a labware above which the arm shall be parked if Keep tips mounted is selected.

The well selection pattern for source and destination must match. There can be either a column offset or a row offset, not both. If number of samples is defined, the well selection plus offset defines only the start well.

11.7.2 Pick Up Tips (MCA96)

Use this command to pick up disposable tips for the MCA96 from a specified position on the worktable.

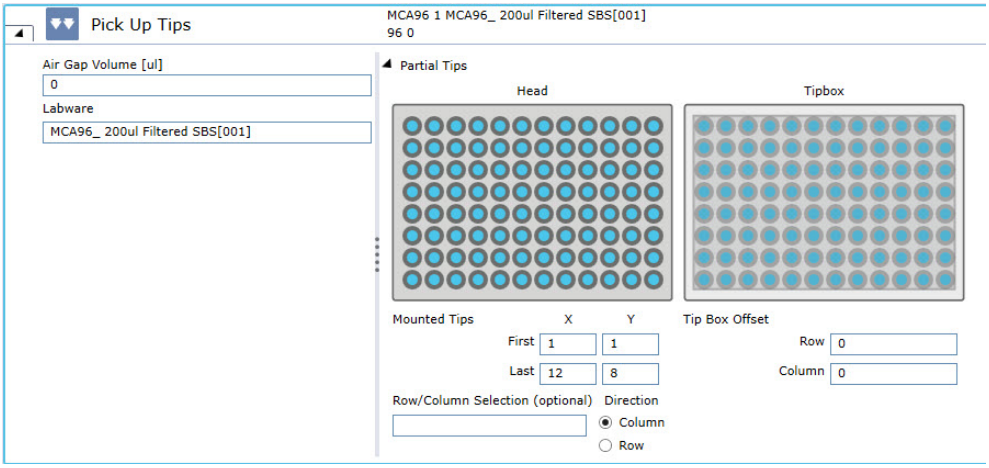


Fig. 242: Pick up Tips

Parameters

Air gap volume [µl]: The volume for the air gap in micro liters.

Labware: In the Labware box, either type your expression or click on a labware displayed in the Worktable editor window (Worktable click).

Tips mounted

First X: Define the first tip position on the head in X-direction where the tips are mounted.

First Y: Define the first tip position on the head in Y-direction where the tips are mounted.

Last X: Define the last tip position on the head in X-direction where the tips are mounted.

Last Y: Define the last tip position on the head in Y-direction where the tips are mounted.

Tip Box Offset

Row: Enter the offset of the tips in rows on the tip box. You can move from one tip tray to another using this parameter.

Column: Enter the offset of the tips in columns on the tip box. It is possible to move from one tip tray to another using this parameter.

Row/Column Selection (optional): Allows working with variable number of columns/rows of MCA96 tips from a tip-box that has been appropriately reracked (i.e., contains only the tips to be mounted). You can set the direction by selecting Column or Row (selection must be consistent between Pick Up Tips, all Aspirate/ Mix/Dispense commands, and the subsequent Set Tips Back command—this is enforced by context check).

11.7.3 Set Tips Back (MCA96)

Use this command to drop the disposable tips of the MCA96 to a specified position on the worktable.

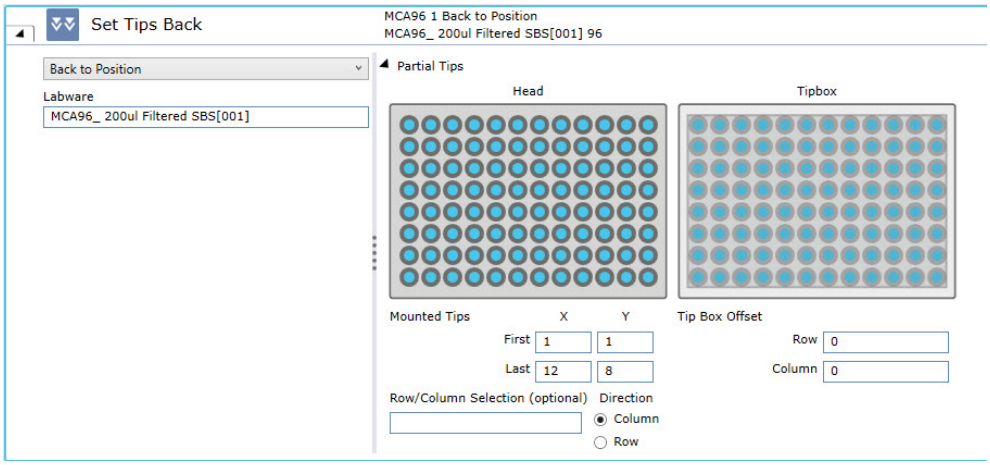


Fig. 243: Set Tips Back

Parameters

Back to Source : This option drops the tips back to the position where they were originally picked up from.

Back to Position : This option drops the tips back to a specified position on the worktable.

Labware: In the Labware box, either type your expression or click on a labware displayed in the Worktable editor window (Worktable click).

Tips mounted

First X: Define the first tip position on the head in X-direction where the tips are mounted.

First Y: Define the first tip position on the head in Y-direction where the tips are mounted.

Last X: Define the last tip position on the head in X-direction where the tips are mounted.

Last Y: Define the last tip position on the head in Y-direction where the tips are mounted.

Tip Box Offset

Row: Enter the offset of the tips in rows on the tip box. You can move from one tip tray to another using this parameter.

Column: Enter the offset of the tips in columns on the tip box. It is possible to move from one tip tray to another using this parameter.

Row/Column Selection (optional): Allows working with variable number of columns/rows of MCA96 tips from a tip-box that has been appropriately reracked (i.e., contains only the tips to be mounted). You can set the direction by selecting Column or Row (selection must be consistent between Pick Up Tips, all Aspirate/ Mix/Dispense commands, and the subsequent Set Tips Back command—this is enforced by context check).

11.7.4 Get Tips (MCA96)

The MCA96 Get Tips command is used to pick up disposable tips for the MCA96 arm with the specified DiTi type on the worktable.

It keeps track of tip positions and automatically picks up the next available and accessible unused tips of the specified type.

Physically inaccessible tip boxes are not considered to pickup tips from.

The order of a list of available tip boxes goes by name in alphabetical order (“MCA96_1000ul SBS[001]”, “MyBox[001]”, “NextBox[001]”), then by index if the names are the same (“MCA96_1000ul SBS[001]”, “MCA96_1000ul SBS[002]”).

Partial tip pickup is supported only for full columns or full rows, not both. For this the First and the Last in X and Y Positions for the channels of the head need to be defined.

If not enough columns or rows of tips are available in one tip box and additional tip boxes with sufficient tips are present on the worktable, the Get Tips command automatically executes re-racking of the tips so that the required number of columns or rows can be mounted.

Depending on its settings, the **OnEmptyTray** procedure may be called if any empty DiTi rack of the defined type is available.



If during the re-racking procedure a partial inaccessibility occurs where tips can't be set back into the target box, no tips will be picked up and an error occurs during runtime. To avoid those errors please validate your application.

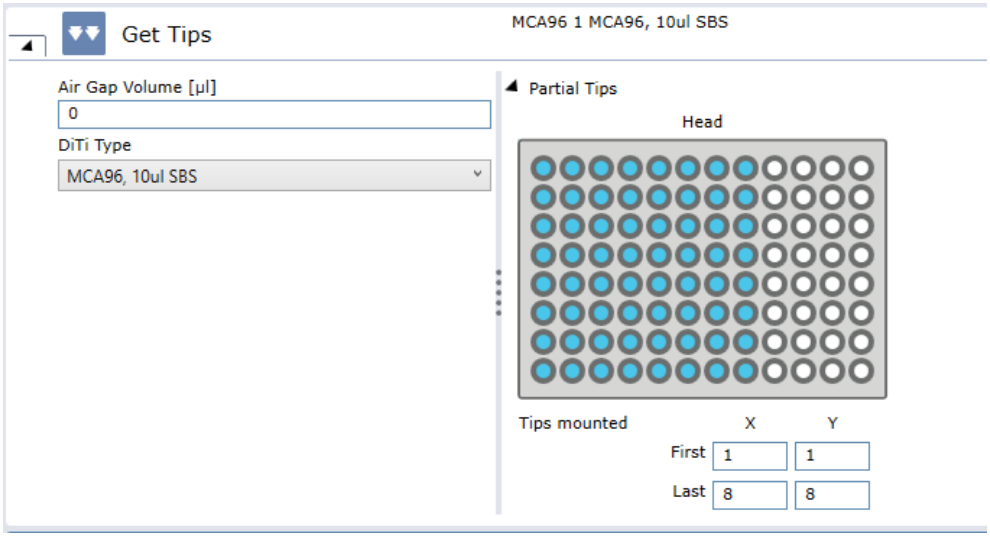


Fig. 244: User interface of 'Get Tips' command

Parameters

Air gap volume [µl] Defines the volume that is aspirated before mounting any tips.

DiTi Type Selects the disposable tip type from the dropdown menu or by clicking on the tip box on the worktable.

Partial Tips Open this section if partial tips shall be mounted. This can be done by selecting the appropriate channels from the head in the UI or by defining the x and y offsets on the head numerically.

- First X: Define the first tip position on the head in x direction where the tips are mounted
- First Y: Define the first tip position on the head in y direction where the tips are mounted.
- Last X: Define the last tip position on the head in x direction where the tips are mounted.
- Last Y: Define the last tip position on the head in y direction where the tips are mounted.

11.7.5 Drop Tips (MCA96)

Use this command to drop the disposable tips of the multi-channel arm to a specified DiTi Waste.

Parameters

DiTi Waste Defines the waste where the tips will be discarded.



Fig. 245: Drop Tips

11.7.6 Aspirate (MCA96)

Use this command to aspirate liquid from specified labware on the worktable with the MCA96.

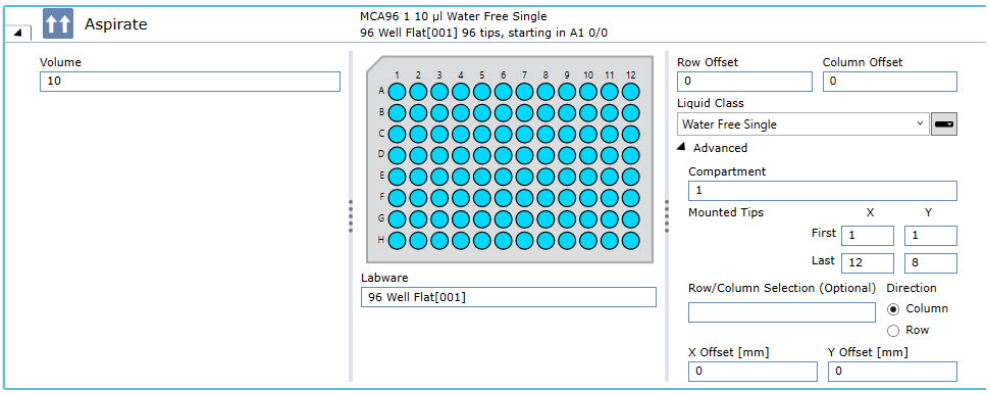


Fig. 246: Aspirate

Parameters

Volume: Set the volume for all tips of the MCA. You can specify how many units in micro liters (µl) to be pipetted into each well.

Labware: In the Labware box, either type your expression or click on a labware displayed in the Worktable editor window (Worktable click).

Labware Offset: The labware offset parameters describe the offset of the mounted tips relative to the labware:

- **Row:** ExpressionInputBox which describes the offset of the mounted tips in rows relative to the labware. It is possible to move from one plate to another using this parameter.
- **Column:** ExpressionInputBox which describes the offset of the mounted tips in columns relative to the labware. It is possible to move from one plate to another using this parameter.

Liquid Class: Choose the required liquid class from the drop-down list or by means of an expression typed in (e.g., by using a variable defined within your script). The list only shows liquid classes that are suitable for the current configuration of the pipetting instrument.

Advanced Parameters

Compartment: Some special labware types (e.g., for protein crystallography) have wells with several compartments (cavities). FluentControl can pipette a different liquid into each of the compartments. Each well typically has one main compartment and one or more satellite compartments which are directly adjacent and are often smaller in size. Choose the compartment which you want to use for pipetting (the main compartment is normally compartment 1). Well compartments are specified in the labware configuration.

Tips mounted

First X: Define the first tip position on the head in X-direction where the tips are mounted.

First Y: Define the first tip position on the head in Y-direction where the tips are mounted.

Last X: Define the last tip position on the head in X-direction where the tips are mounted.

Last Y: Define the last tip position on the head in Y-direction where the tips are mounted.

X offset [mm]: This is the offset of the tip from the center of the target well (negative = left).

Y offset [mm]: This is the offset of the tip from the center of the target well (negative = to front, positive = to back).

Row/Column Selection (optional): Allows working with variable number of columns/rows of MCA96 tips from a tip-box that has been appropriately reracked (i.e., contains only the tips to be mounted). You can set the direction by selecting Column or Row (selection must be consistent between Pick Up Tips, all Aspirate/ Mix/Dispense commands, the Detect Liquid command and the subsequent Set Tips Back command—this is enforced by context check).

11.7.7 Dispense (MCA 96)

Use this command to dispense liquid into specified labware on the worktable with the MCA96.

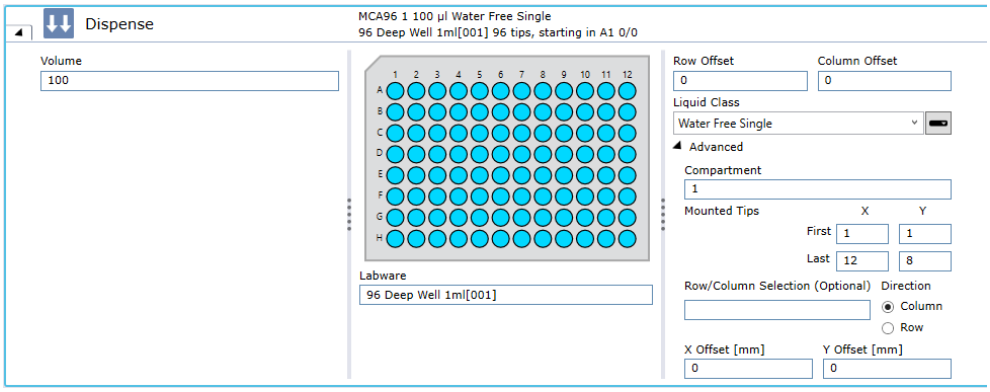


Fig. 247: Dispense

Parameters

Volume: Set the volume for all tips of the MCA. You can specify how many units in micro liters (µl) to be pipetted into each well.

Labware: In the Labware box, either type your expression or click on a labware displayed in the Worktable editor window (Worktable click).

Labware Offset: The labware offset parameters describe the offset of the mounted tips relative to the labware:

- **Row:** ExpressionInputBox which describes the offset of the mounted tips in rows relative to the labware. It is possible to move from one plate to another using this parameter.
- **Column:** ExpressionInputBox which describes the offset of the mounted tips in columns relative to the labware. It is possible to move from one plate to another using this parameter.

Liquid Class: Choose the required liquid class from the drop-down list or by means of an expression typed in (e.g., by using a variable defined within your script). The list only shows liquid classes that are suitable for the current configuration of the pipetting instrument.

Advanced Parameters

Compartment: Some special labware types (e.g., for protein crystallography) have wells with several compartments (cavities). FluentControl can pipette a different liquid into each of the compartments. Each well typically has one main compartment and one or more satellite compartments which are directly adjacent

and are often smaller in size. Choose the compartment which you want to use for pipetting (the main compartment is normally compartment 1). Well compartments are specified in the labware configuration.

Tips mounted

First X: Define the first tip position on the head in X-direction where the tips are mounted.

First Y: Define the first tip position on the head in Y-direction where the tips are mounted.

Last X: Define the last tip position on the head in X-direction where the tips are mounted.

Last Y: Define the last tip position on the head in Y-direction where the tips are mounted.

X offset [mm]: This is the offset of the tip from the center of the target well (negative = left).

Y offset [mm]: This is the offset of the tip from the center of the target well (negative = to front, positive = to back).

Row/Column Selection (optional): Allows working with variable number of columns/rows of MCA96 tips from a tip-box that has been appropriately reracked (i.e., contains only the tips to be mounted). You can set the direction by selecting Column or Row (selection must be consistent between Pick Up Tips, all Aspirate/ Mix/Dispense commands, the Detect Liquid command and the subsequent Set Tips Back command—this is enforced by context check).

11.7.8 Mix (MCA96)

Use this command to perform a mix operation by aspirating and then dispensing in the same position.

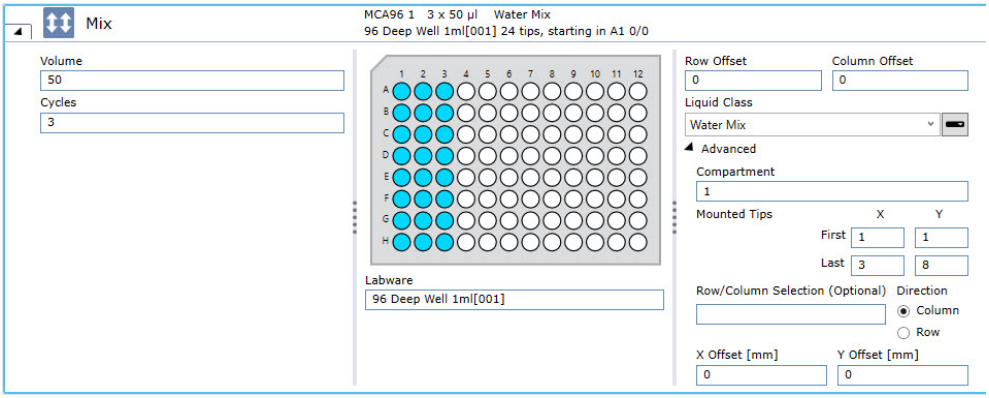


Fig. 248: Mix

Parameters

Volume: Set the volume for all tips of the MCA. You can specify how many units in micro liters (µl) to be pipetted into each well.

Cycles: Set the number of mixing cycles. You can select existing variables.

Labware: In the Labware box, either type your expression or click on a labware displayed in the Worktable editor window (Worktable click).

Labware Offset: The labware offset parameters describe the offset of the mounted tips relative to the labware:

- **Row:** ExpressionInputBox which describes the offset of the mounted tips in rows relative to the labware. It is possible to move from one plate to another using this parameter.
- **Column:** ExpressionInputBox which describes the offset of the mounted tips in columns relative to the labware. It is possible to move from one plate to another using this parameter.

Liquid Class: Choose the required liquid class from the drop-down list or by means of an expression typed in (e.g., by using a variable defined within your script). The list only shows liquid classes that are suitable for the current configuration of the pipetting instrument.

Advanced Parameters

Compartment: Some special labware types (e.g., for protein crystallography) have wells with several compartments (cavities). FluentControl can pipette a different liquid into each of the compartments. Each well typically has one main compartment and one or more satellite compartments which are directly adjacent and are often smaller in size. Choose the compartment which you want to use for pipetting (the main compartment is normally compartment 1). Well compartments are specified in the labware configuration.

Tips mounted

First X: Define the first tip position on the head in X-direction where the tips are mounted.

First Y: Define the first tip position on the head in Y-direction where the tips are mounted.

Last X: Define the last tip position on the head in X-direction where the tips are mounted.

Last Y: Define the last tip position on the head in Y-direction where the tips are mounted.

X offset [mm]: This is the offset of the tip from the center of the target well (negative = left).

Y offset [mm]: This is the offset of the tip from the center of the target well (negative = to front, positive = to back).

Row/Column Selection (optional): Allows working with variable number of columns/rows of MCA96 tips from a tip-box that has been appropriately reracked (i.e., contains only the tips to be mounted). You can set the direction by selecting Column or Row (selection must be consistent between Pick Up Tips, all Aspirate/ Mix/Dispense commands, the Detect Liquid command and the subsequent Set Tips Back command—this is enforced by context check).

11.7.9 Empty Tip Contents (MCA96)

Use this command to dispense and liquid currently in the tips to a specified position on the worktable.

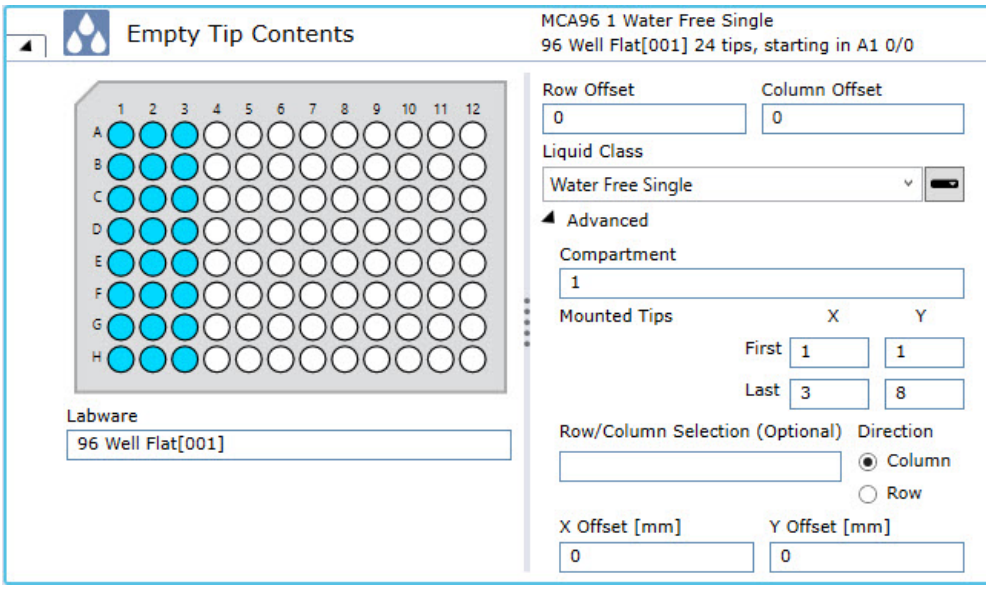


Fig. 249: Empty Tip Contents

Parameters

Labware: In the Labware box, either type your expression or click on a labware displayed in the Worktable editor window (Worktable click).

Labware Offset: The labware offset parameters describe the offset of the mounted tips relative to the labware:

- **Row:** ExpressionInputBox which describes the offset of the mounted tips in rows relative to the labware. It is possible to move from one plate to another using this parameter.
- **Column:** ExpressionInputBox which describes the offset of the mounted tips in columns relative to the labware. It is possible to move from one plate to another using this parameter.

Liquid Class: Choose the required liquid class from the drop-down list or by means of an expression typed in (e.g., by using a variable defined within your script). The list only shows liquid classes that are suitable for the current configuration of the pipetting instrument.

Advanced Parameters

Compartment: Some special labware types (e.g., for protein crystallography) have wells with several compartments (cavities). FluentControl can pipette a different liquid into each of the compartments. Each well typically has one main compartment and one or more satellite compartments which are directly adjacent and are often smaller in size. Choose the compartment which you want to use for pipetting (the main compartment is normally compartment 1). Well compartments are specified in the labware configuration.

Tips mounted

First X: Define the first tip position on the head in X-direction where the tips are mounted.

First Y: Define the first tip position on the head in Y-direction where the tips are mounted.

Last X: Define the last tip position on the head in X-direction where the tips are mounted.

Last Y: Define the last tip position on the head in Y-direction where the tips are mounted.

X offset [mm]: This is the offset of the tip from the center of the target well (negative = left).

Y offset [mm]: This is the offset of the tip from the center of the target well (negative = to front, positive = to back).

Row/Column Selection (optional): Allows working with variable number of columns/rows of MCA96 tips from a tip-box that has been appropriately reracked (i.e., contains only the tips to be mounted). You can set the direction by selecting Column or Row (selection must be consistent between Pick Up Tips, all Aspirate/ Mix/Dispense commands, the Detect Liquid command and the subsequent Set Tips Back command—this is enforced by context check).

11.7.10 Move Arm (MCA96)

Use this command to move the MCA96 from one position to another without performing any other operation.

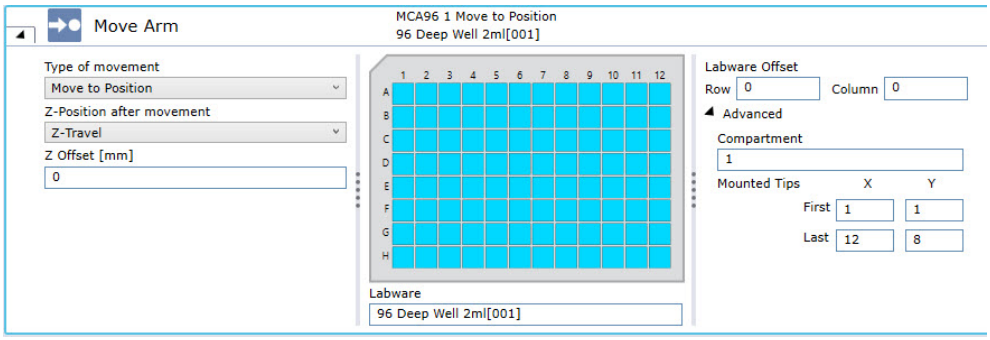


Fig. 250: Move Arm

Parameters

Type of Movement: List to set the movement type:

- If **Move to Position** or **Move to Position at Global Z Travel** is selected, the system moves the arm to the selected labware with maximum speed.
- If **Move in X, Move in Y or Move in Z** is selected, the system uses the speed of the movement value of the parameter Movement Speed.
- If **Move to Position** is selected, the system uses the standard movements as determined by PathFinder.
- If **Move to Position at Global Z Travel** is selected, the system first moves the arm in Z to Z Travel and then moves in X and Y direction to the target position by a PathFinder controlled movement. Finally, the system moves the arm to the Z Position as defined in the move arm command

Site Information

Z-Position after movement

Z-Offset [mm]: Defines an offset in millimeters to the Z-Position after movement (positive = higher, negative = lower), if Move in Z is selected.

Labware: In the Labware box, either type your expression or click on a labware displayed in the Worktable editor window (Worktable click).

Labware Offset: The labware offset parameters describe the offset of the mounted tips relative to the labware:

- **Row:** ExpressionInputBox which describes the offset of the mounted tips in rows relative to the labware. It is possible to move from one plate to another using this parameter.
- **Column:** ExpressionInputBox which describes the offset of the mounted tips in columns relative to the labware. It is possible to move from one plate to another using this parameter.

Liquid Class: Choose the required liquid class from the drop-down list or by means of an expression typed in (e.g., by using a variable defined within your script). The list only shows liquid classes that are suitable for the current configuration of the pipetting instrument.

Advanced Parameters

Compartment: Some special labware types (e.g., for protein crystallography) have wells with several compartments (cavities). FluentControl can pipette a different liquid into each of the compartments. Each well typically has one main compartment and one or more satellite compartments which are directly adjacent and are often smaller in size. Choose the compartment which you want to use for pipetting (the main compartment is normally compartment 1). Well compartments are specified in the labware configuration.

Tips mounted

First X: Define the first tip position on the head in X-direction where the tips are mounted.

First Y: Define the first tip position on the head in Y-direction where the tips are mounted.

Last X: Define the last tip position on the head in X-direction where the tips are mounted.

Last Y: Define the last tip position on the head in Y-direction where the tips are mounted.

11.7.11 Detect Liquid (MCA96)

The Detect Liquid command is used to carry out liquid level detection without pipetting and reports the liquid volume for each of the selected Labware wells. The detected liquid volume value is directly written into the corresponding well attribute and can later be read out via the **GetAttribute(LabwareName)** function. Furthermore, a subsequent pipetting step can pipette from the detected height when using the (Liquid Class) LC setting to pipette from a known position.

The detected liquid is not written into the corresponding well attribute if the labware has more than one well. This is done because the MCA96 only receives one cLLD signal across all tips. Detecting liquid with potentially different volumes in a labware with multiple wells can lead to incorrect results in FluentControl.

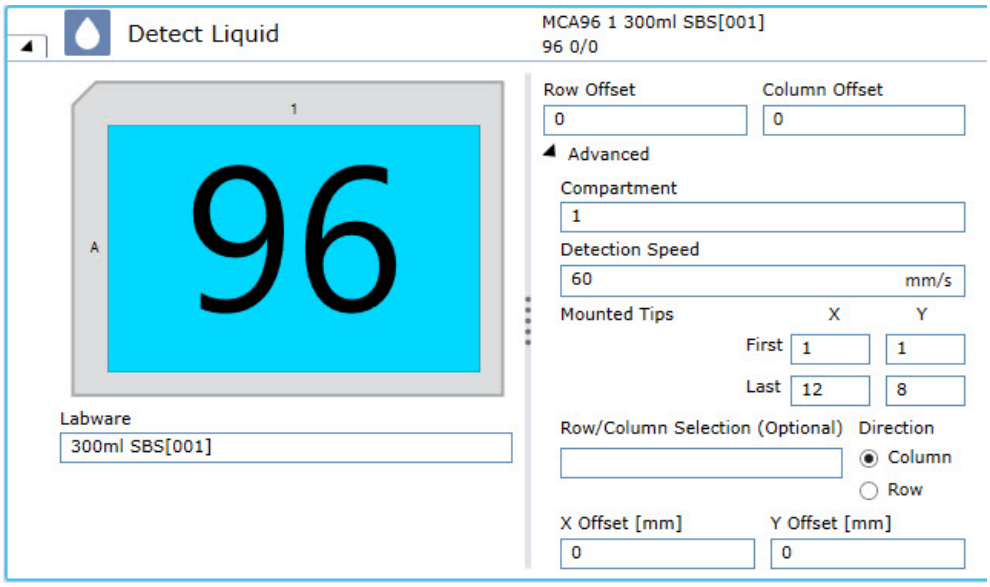


Fig. 251: Detect Liquid

Parameters

Labware: In the Labware box, either type your expression or click on a labware displayed in the Worktable editor window (Worktable click).

Labware Offset: The labware offset parameters describe the offset of the mounted tips relative to the labware:

- **Row:** ExpressionInputBox which describes the offset of the mounted tips in rows relative to the labware. It is possible to move from one plate to another using this parameter.
- **Column:** ExpressionInputBox which describes the offset of the mounted tips in columns relative to the labware. It is possible to move from one plate to another using this parameter.

Advanced Parameters

Compartment: Some special labware types (e.g., for protein crystallography) have wells with several compartments (cavities). FluentControl can pipette a different liquid into each of the compartments. Each well typically has one main compartment and one or more satellite compartments which are directly adjacent and are often smaller in size. Choose the compartment which you want to use for pipetting (the main compartment is normally compartment 1). Well compartments are specified in the labware configuration.

Detection Speed: The speed in mm/s at which the arm shall move towards the liquid during the detection move.

Tips mounted

First X: Define the first tip position on the head in X-direction where the tips are mounted.

First Y: Define the first tip position on the head in Y-direction where the tips are mounted.

Last X: Define the last tip position on the head in X-direction where the tips are mounted.

Last Y: Define the last tip position on the head in Y-direction where the tips are mounted.

X offset [mm]: This is the offset of the tip from the center of the target well (negative = left).

Y offset [mm]: This is the offset of the tip from the center of the target well (negative = to front, positive = to back).

Row/Column Selection (optional): Allows working with variable number of columns/rows of MCA96 tips from a tip-box that has been appropriately reracked (i.e., contains only the tips to be mounted). You can set the direction by selecting Column or Row (selection must be consistent between Pick Up Tips, all Aspirate/ Mix/Dispense commands, the Detect Liquid command and the subsequent Set Tips Back command—this is enforced by context check).

11.8 MCA96 Gripper

The MCA96 Gripper is an option for the MCA96 that allows the arm to mount gripper fingers (similar to the RGA FES). The arm is then able to move labware across the worktable. With the help of the Get Fingers and Drop Fingers commands, the MCA96 can mount and demount fingers during the run. The MCA96 Transfer Labware command allows moving labware (plates and tip boxes) using the MCA96.

Pipetting with the MCA96 can only be done when fingers are not mounted to the MCA96. Similarly, moving labware can only be done when tips are not mounted.

11.8.1 MCA96 Driver Configuration

If the instrument is equipped with an MCA96 with finger exchange system (FES), the general settings for its function as a plate robot can be configured under Configure System|Drivers|MCA96 …

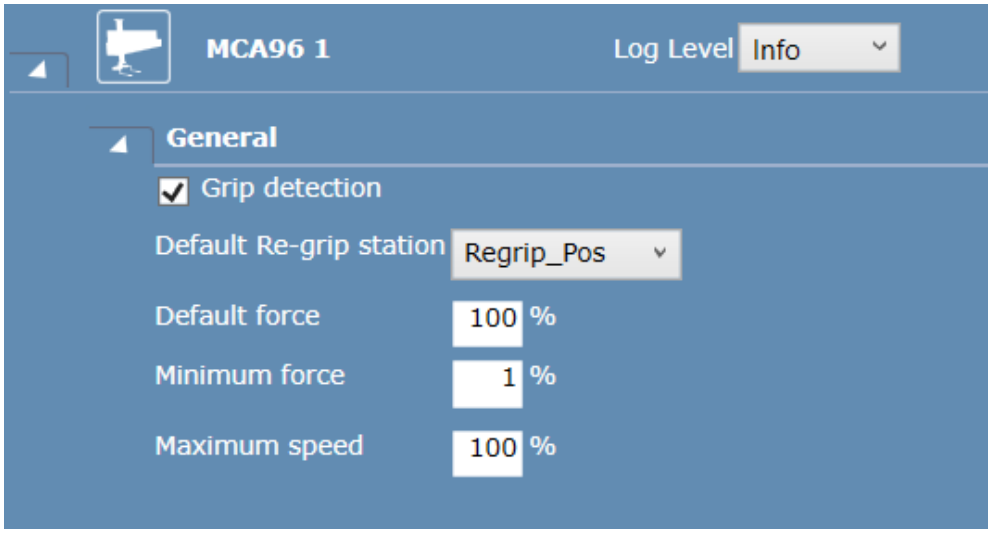


Fig. 252: MCA96 Driver Configuration

Grip detection: If enabled, the system will check for each plate gripping action whether or not an object was detected by the hardware and trigger the appropriate error handling. If this option is disabled, the system assumes that the grip action is always successful.

Default Re-grip station: Here, the location name of the default re-grip station can be specified. This re-grip station will be used whenever re-gripping is needed for a labware transfer (see Introduction|Regrip) and no re-grip station is specified by any involved location or the labware.

Default force: The default gripping force (in percent) to be used when gripping force is not limited by the labware via the Force custom attributes.

Minimum force: The lower limit for gripping force reduction. This setting can prevent that a too low gripping force is used. A too low gripping force can lead to erroneous object detection and loss of object while moving.

Maximum speed: The upper limit for the movement speed for plate transfers.

11.8.2 Get Fingers (MCA96 Gripper)

Use this command to pick up the MCA96 gripper fingers.

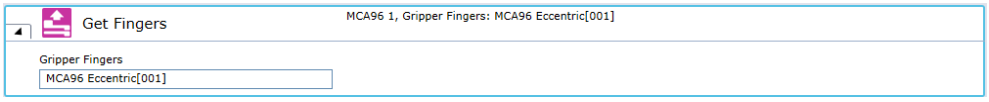


Fig. 253: Get Fingers

Parameters

Gripper Fingers: Select the MCA96 gripper fingers to be picked up from the worktable.

11.8.3 Drop Fingers (MCA96 Gripper)

Use this command to demount the MCA96 gripper fingers.

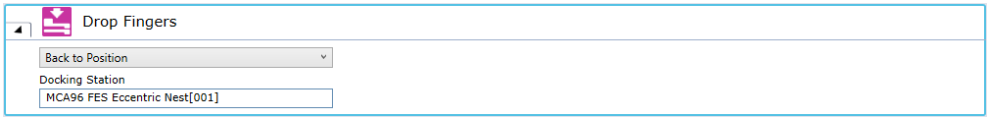


Fig. 254: Drop Fingers

Parameters

Back to Source: By selecting this option, the MCA96 gripper fingers are returned to the position from where they were fetched from.

Back to Position: By selection this option, the user can specify a return position for the MCA96 gripper fingers.

Docking Station: Enter an expression or click a suitable finger nest on the worktable.

11.8.4 Transfer Labware (MCA96 Gripper)

Use this command together with an MCA96 to transfer labware objects or labware-like carriers to a location.

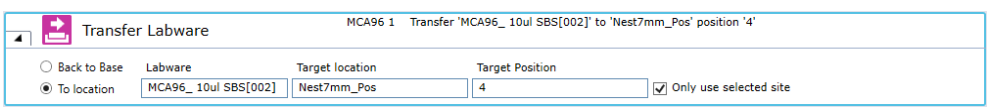


Fig. 255: Transfer Labware

Carriers

When transferring carriers, the carrier's name has to be typed in manually. The carrier needs the CustomAttribute **MoveableCarrier** set to true and to specify the CustomAttributes **Grip_Wide. 24 Eppendorf Adapter** is a moveable carrier and can be used as example.

You can specify the labware to be transferred by the MCA96. This can be done by entering the labware name or an expression which resolves into a labware name into the Labware box or by selecting labware on the worktable.

You can select from the following, mutual exclusive options:

- **Move to location:** You can specify the location the labware will be transferred to.
  By default, the system will place the labware on the first free site of the target location. The user can override this behavior with the check box "Only use selected site" and specify a dedicated site on the target location. The target site can be specified by entering the site number or an expression which resolves into a site number into a text box or by selecting a carrier or site on the worktable.
- **Move to base:** The system will move the labware to its base. If this option is selected, the UI controls for location and site are dimmed.

Context check: The system will check if the move is possible (labware exists, location/target site is available) and raise context check messages if an error is detected.

Move simulation: The system will simulate the move in Move Simulation mode and update the worktable (move the labware to the destination) if the move is possible.

Parameters

Labware: In the Labware box, either type your expression or click the labware displayed in the Worktable editor window (Worktable Click).

To location: Click **To location** to specify the location the labware should be moved to. In the **Labware** box, type the location name (or an expression which resolves into a location name) or select a carrier or site on the worktable

Back to Base: Select **Back to Base**, to move the labware to its base—i.e., the original position where the labware was placed in the initial workspace. If the **Back to Base** option is selected, **Target location** is set to "<BASE>" and appears dimmed. **Target Position** appears dimmed also.

Target location: Select a target location for used tip racks in the list box. The usage of "<BASE>" as an alias for the base location is supported.

Target Position: The default is to place the labware on the first free site of the target location.

Only use selected site: To change the default, select the Only use selected site check box and specify a dedicated site on the target location.

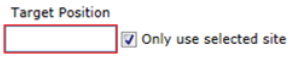


Fig. 256: Target Position

To specify the target site, click in the **Target Position** box and type the site number (or an expression which resolves into a site number) or select a carrier or site on the worktable.

11.8.5 Replace Used DiTi Racks (MCA96 Gripper)

Use this command to replace empty tip racks on the workable with unused tip racks.

The command searches for any tip boxes or tip spacers which are empty or which only contain used tips. This search is limited to carriers which have the **Accessible for pipetting arm** option set.

If such labware is found, it is moved to the target location. If the **Only remove** option is not selected, fresh tips are moved from carriers with **Accessible for pipetting arm** not set (e.g. hotels) to the now free positions.

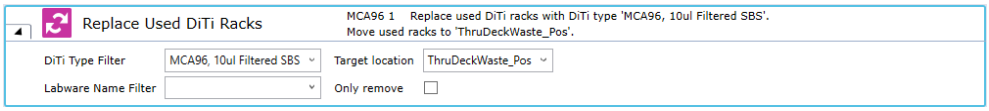


Fig. 257: Replace Used DiTi Racks

Parameters

DiTi Type Filter: Select the disposable tip type to replace with from the list box or type an expression to specify the tip type.

Labware Name Filter: Select a labware from the list of all available labware names or by clicking it in the Worktable editor. Alternatively, you can type an expression to specify the labware name filter.

Target location: Select a target location for used DiTi racks in the list box. The usage of **<BASE>** as an alias for the base location is supported. Note that in contrast to other commands like Transfer Labware, the base position information is ignored if the target location is the base location of the DiTi rack, i.e. the first free position will be used to store the DiTi rack.

Only remove: To remove the tip racks without replacement, select the **Only remove** check box.

11.8.6 Execute Single Vector (MCA96 Gripper)

Use this command to execute a vector (list of 3D waypoints) with the MCA96 gripper.

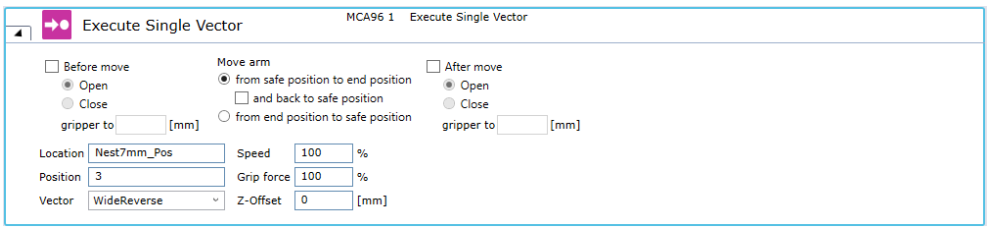


Fig. 258: Executive Single Vector

Move Arm

This option allows the user to specify the movement direction of the selected vector which will be executed. The following directions can be selected:

- From safe position to end position (with the possibility to move back to the safe position)
- From end position to safe position

If the option **From safe position to end position** is selected, the arm will execute the movement vector like it is defined in the selected Location (from top to bottom). Selecting the option **and back to safe position** will execute the vector again in reverse order (e.g. to grip an object and move out again).

If the option **From end position to safe position** is selected, the arm will only execute the movement vector in reverse order.

In addition to the execution of the movement vector, the grippers can be opened and closed. This can be done before and/or after the move. The following options can be selected:

Before Move This option will open or close the gripper to a specified width, in millimeters, before the move vector is executed. The Close option will use grip detection (if it is enabled in the Configure System menu) to detect a labware item in between the grippers. If the gripper reaches the specified position without detecting a labware item, an error is raised. The **Open** option is executed without grip detection.

After Move This option will open or close the gripper to a specified width, in millimeters, after the move vector is executed. The **Close** option will enable a grip detection to detect a labware item in between the grippers. If the gripper reaches the specified position without detecting a labware item, an error is raised. The **Open** option is executed without grip detection. By default, neither **Before Move** nor **After Move** is selected. This setup which will only execute the selected movement vector without moving the gripper.

Parameters **Location and Position:** Specify the name of a location and a site number for which the vector shall be executed. This can be done by clicking a carrier, site or labware on the worktable.

Vector: Select a vector name from the **Vector box**. The list contains the names of all vectors for the current robot stored in the selected carrier.

Speed: Specify the move speed in percent (1–100%).

Grip force: Specify the gripper force in percent (1–100%).

All edit fields support variables and expressions.

11.9 Resolvex i300

11.9.1 Setting up Resolvex i300

The following sections describe how to configure the Resolvex i300 to work with FluentControl. Before scripting, please make sure that you familiarize yourself with the “Configure System” menu of the i300 and the i300 Liquid Classes as the name of the liquids directly impacts the Liquid Class executed. This section also includes instructions of how to create custom extraction labware, how to integrate the device into a FluentControl worktable and how to access the transfer positions. A list of i300 specific commands and a brief description of their use is also included.

11.9.1.1 Configure System

In the **Configure System** section it is possible to disable the I/O state and thereby skip execution of any i300 commands. This prevents context or runtime errors from occurring, even if no Resolvex i300 device is connected. The units used for pressure profiles in the system can be switched between bar and PSI. This does not affect how pressure profiles are saved and executed, only the unit shown in the GUI. Therefore, profiles are executed with the defined pressure, regardless of the display unit setting. The user also has the possibility to define which liquid is connected to which port. The liquid name is used in Liquid Classes to identify the port to which the liquid is connected. The maximum capacity of the bottle used can also be set here, and the current volume is shown. It is strongly recommended to use port 1 for water, as this port is used for initialization of the plungers. The current volume (in milliliters) can be retrieved as a regular expression in a script: `CurrentBottleVolumeMl("Water")`

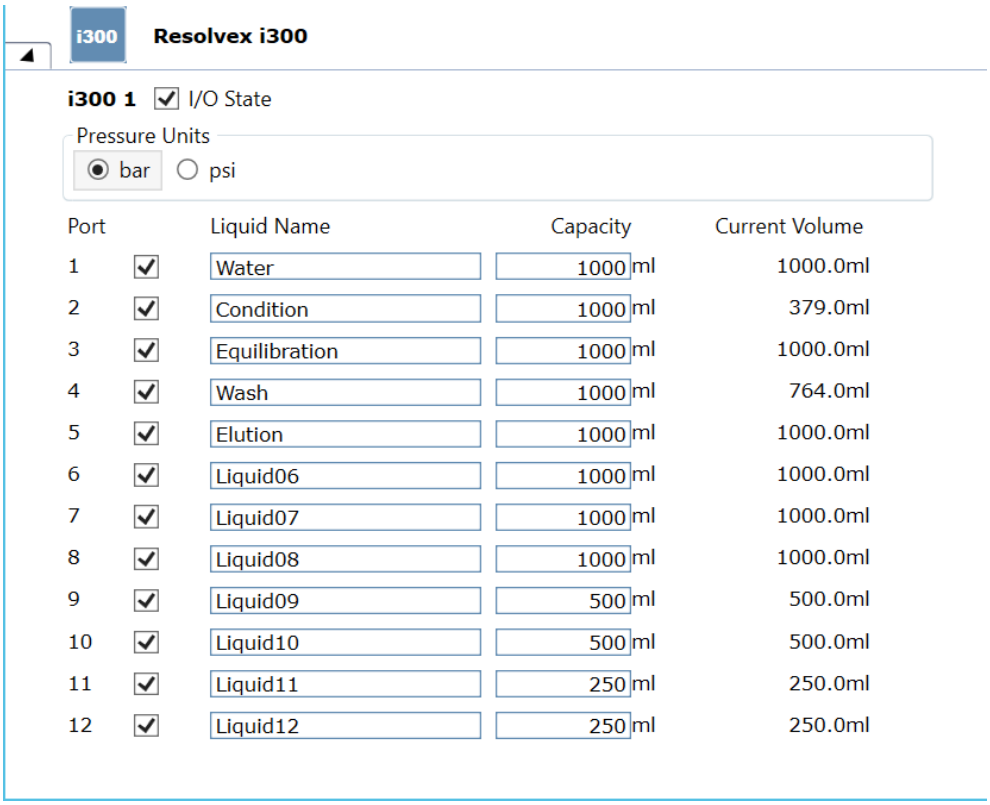


Fig. 259: Configure Settings Resolvex i300

11.9.1.2 Liquid Class Editor

Two distinct categories of Liquid Classes are available for the Resolvex i300

- Dispense Liquid Classes (example shown in first screenshot below)
- Wash Liquid Classes (example shown in second screenshot below)

Dispense Liquid Classes are used by the Dispense command to dispense into filtration or collection labware. Wash Liquid Classes are used by the Wash command to prime the liquid system. The wash command dispenses the liquid into the priming station.

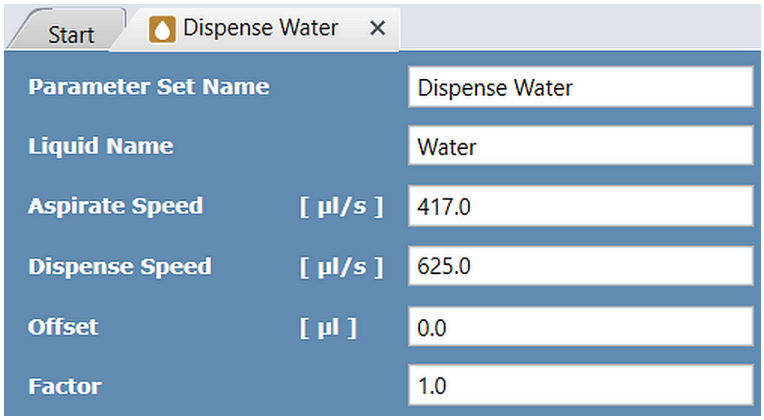


Fig. 260: Liquid Class example for Dispense command

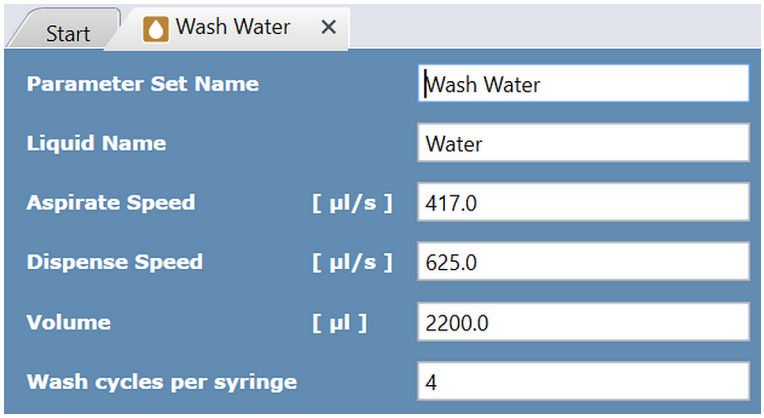


Fig. 261: Liquid Class example for Wash command

Parameters

Parameter Set Name: Name displayed in Dispense Command

Liquid Name: Name of the Liquid to use. This creates with the configuration the linkage to the Port to use.

Aspirate Speed: The speed the Plunger moves while aspirating out of the Bottle.

Dispense Speed: The speed the Plunger moves while dispensing into the Labware.

Offset (Dispense): The offset of Volume to add to reach the desired target volume.

Factor (Dispense): The factor used to dispense. To be able to add a factor if the liquid needs more volume in plunger move to reach the desired target volume in the well.

Volume (Wash): Volume used for one wash per syringe.

Wash cycles per syringe (Wash): Cycles per syringe with the volume to be executed.

11.9.1.3 Move Tool

The move tool is used to adjust the pressure when recording a pressure profile as well as adjusting temperature, flow and **z** height while recording an evaporation profile. After finishing recording, it is recommended to remove intermediate profile points which are not required.

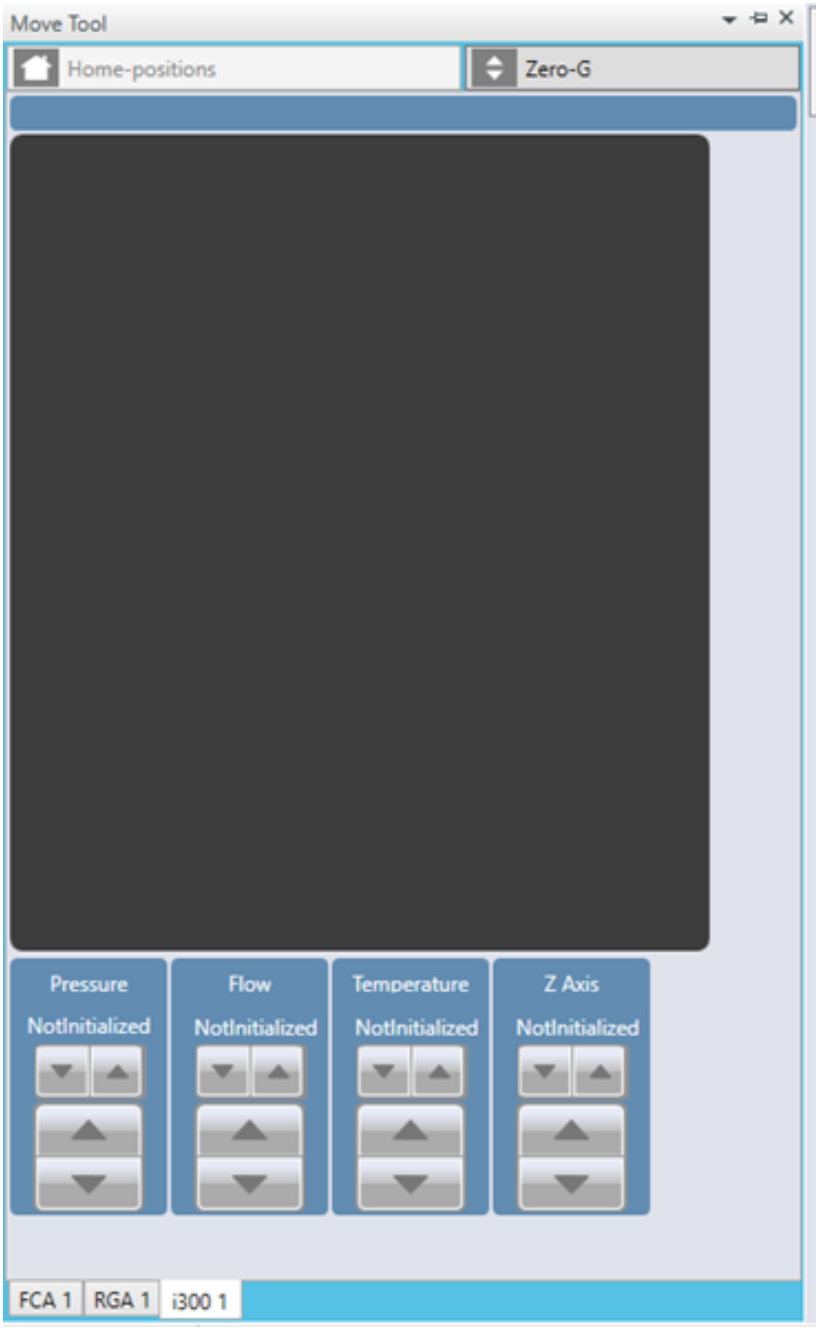


Fig. 262: Move Tool

11.9.2 Use of Resolvex i300 in FluentControl

The following section contains important information about using the i300. Please read carefully and consider during method development and execution.

11.9.2.1 Considerations for scripting and teaching

11.9.2.1.1 Carriers and Labware

Carrier, labware and other objects for use with the Resolvex have **i300** in their names and can therefore easily be found in the Search tab of the Infopad. All labware objects designated for the i300 can be found in the “Extraction” group. Only the designated labware definitions work with the device.
If labware properties need to be adjusted or further labware objects are required, duplicate one of the default objects and modify it to fulfill the actual needs. Do not use other standard FluentControl labware, as these are missing properties required for proper handling in the i300 device.



Fig. 263: Infopad

The following default objects are available:

Tab. 8: Overview Default i300 Objects

Name / Type	Section	Comment
Module and Accessories		
Resolvex i300 Deck Segment	Carrier\Deck Segment	Required to place the device on the worktable. Placing the device in the cabinet can be done directly, without any further supporting object. Additionally, the segment has one position for a 61 mm nest on the 6 mm nest spacer and one position for the 3x100 ml trough holder on the 3 mm trough holder spacer.
Resolvex i300	Carriers\Device	The device carrier has two positions; the y carriage / waste and the lift. Only the y-carriage position should be used for placing labware. The lift position is used by the software for stacking and unstacking.
24 Nozzle Separator i300 **96 Nozzle Separator i300**	Carriers\Nest	A nozzle separator is always required on the i300 device carrier, before labware can be placed. Labware can be placed on or removed from the nozzle separator manually or by a robotic device.
24 Pressure manifold i300 **96 Pressure Manifold i300**	---	Not available or required as objects.
24 Evaporator Manifold i300 **96 Evaporator Manifold i300**	Tools\i300	Must be placed in the holder on the device to exchange format.
Holder for Evaporator Manifold	Carriers\Nest	Required to place the evaporator manifold onto the Y slide of the device.

6mm Nest Spacer i300	Carriers\Nest	Spacer required for the 61 mm nest to be placed on the i300 deck segment. Labware can be placed on or removed from the nest manually or by a robotic device.
61mm Nest	Carriers\Nest	Standard Fluent nest. Optional position to enable placement of ANSI/SLAS labware in front of the device.
3mm Trough Holder Spacer i300	Carriers\Nest	Spacer required for the 3x100 ml trough holder to be placed on the i300 deck segment
3x100ml Trough Holder i300	Carriers\Nest	Optional holder for 3x 100 ml troughs.
Labware Holders		
24x 6ml Column in Holder **i300 24x 3ml Column in Holder** **i300 96x 1ml Column in Holder** **i300 96x 1ml NBE Column in Holder i300**	Labware\Extraction	Optional holders for Cerex columns. The definition includes the columns themselves. The columns are to be physically placed into the holder by the operator. Robotic movement of individual columns is not foreseen. Don't place the rack directly onto a standard Fluent nest as the nozzles of the columns will touch the nest. Instead, use a Tube Holder or deep well plate to place column holders on other positions than the nozzle separator.
24x 16x100mm Tube in Holder i300 **24x 12x75mm Tube in Holder i300**	Labware\Extraction	Optional holder for glass elution tubes. The definition includes the tubes themselves. The tubes are to be physically placed into the holder by the operator. Robotic movement of individual tubes is not foreseen.
Labware		

96 Column Plate 1ml NBE i300	Labware\Extraction	Definition of Cerex NBE column plate.
96 Deep Well 1_2ml for Column Holder i300	Labware\Extraction	Definition of 41.5 mm high 1.2 ml 96 Deep Well propylene round bottom plate. This definition considers a suitable overlap for stacking and unstacking a column holder on top of the plate.
		Definition of 41.5 mm high 1.2 ml 96 Deep Well propylene round bottom plate. This definition considers a suitable overlap when stacking and unstacking an NBE plate on top of the deep well plate.
24 Deep Well 10ml i300	Labware\Extraction	Definition of 24-well Clear V-Bottom 10 mL Polypropylene Rectangular Well Deep Well Plate.

To place the Resolvex i300 on the worktable, place the deck segment in the respective grid, add the module, and the nozzle separator.

The following applies to the labware in the extraction folder:

- Filtration labware definitions are treated as one object. This applies to single piece extraction labware, as well as single columns/tubes in a holder/rack. The addition and removal of single columns or plates from an adapter with the RGA is not foreseen. Also, semi-skirted plates requiring an adapter frame or similar object combinations are to be defined as one object.
- The extraction labware contains designated read-only attributes, required for stacking / unstacking of the labware with the plate lift.
- Do not stack / unstack labware by the RGA or other robotics devices as this might cause cross-contamination. Always use the stacking / unstacking function of the i300 device, which includes a droplet protector.
- The extraction labware is foreseen to be used in pairs. A pair consists of the filtration labware for SPE and the collection labware for collecting the eluate. This is necessary to ensure correct stacking. The actual height of the stack is determined via the stacking offset of the filtration labware, entered in the stacking height of the collection plate. The Z coordinate of the “Stacked Plate to Plate Offset” in the collection labware expresses the height where the bottom rim of the filtration labware is. See illustrations below.

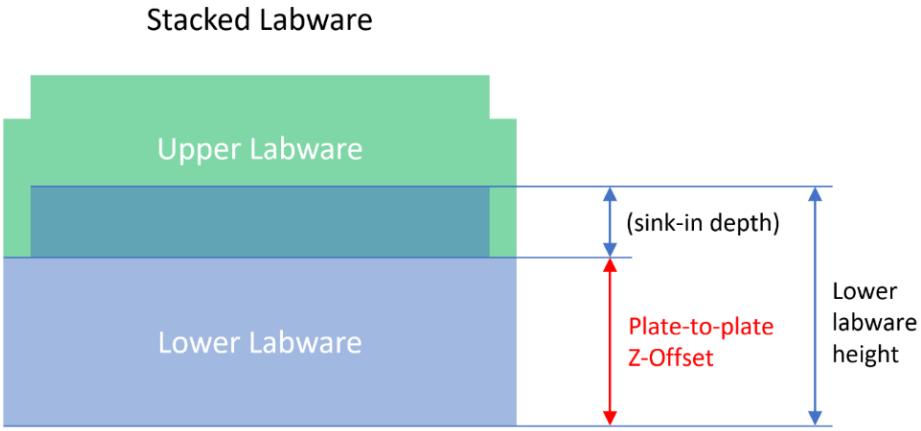


Fig. 264: Stacked Labware

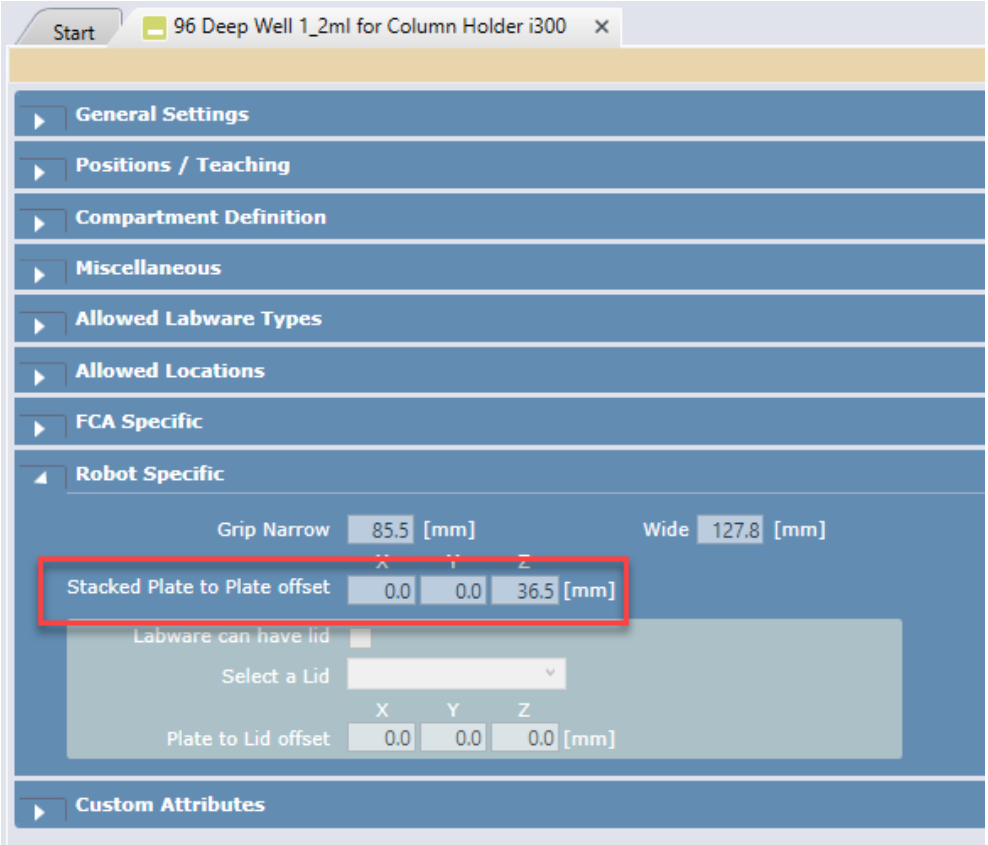


Fig. 265: Offset

- The Positions / Teaching applies to Fluent pipetting arms as well as the i300 dispenser. To use the dispensing function efficiently, Z-Travel and Z-Dispense should have the same value.
- To pipette with the FCA into the wells of the filtration labware, while submerging into the well, the liquid class requires a dispense Z-Offset.

- The high-level Sample Transfer command does not consider tip access on the nozzle separator and might therefore cause runtime errors. Consequently, avoid using this command with the i300, especially with variables.

11.9.2.1.2 Placement and Teaching Below-Deck (in Cabinet)

- To place the Resolvex i300 in the cabinet in FluentControl, follow these steps:
- Initialize the Fluent including the i300
- Duplicate the Resolvex i300 device
- Place the duplicated device to the far left of the worktable (see illustration below)
- Right-click the device in the worktable view and choose Edit
- By following this procedure, the “Placement” section will appear in the editor (see illustration below)
- Open the Placement section
- Grip a plate and teach the position of the Nozzle Separator

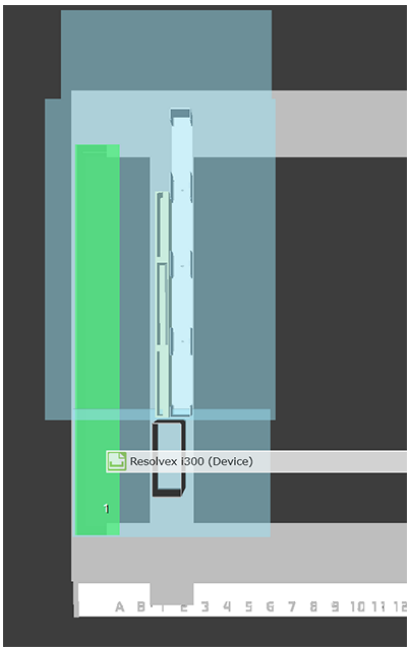


Fig. 266: Placement Resolvex i300 in cabinet

- On placement the rotation must be 0, 90 or 270, otherwise the well selection of the Dispense command does not work correctly.
- The height of the nozzle separator is not considered by the software during teaching. Therefore, the height must be manually deducted after transferring the coordinates in the teach dialog. Specifically, the height of the nozzle separator must be subtracted from the Z coordinate. See steps 1 to 5 in illustration below.

– Relevant height of 24 Nozzle Separator i300: 10 mm
– Relevant height of 96 Nozzle Separator i300: 50 mm



Fig. 267: Placement Resolvex i300 Nozzle Separator

11.9.2.1.3 Accessing the Transfer Position

After executing a command on the Resolvex i300 (e.g. Dispense, Apply Pressure), the Y carriage might be left at “any” position. Therefore, before accessing the site with a Fluent arm or other robotic device, it must be ensured that the carriage is moved to the transfer position by using the Move Carriage command (see illustration below). Please note that some commands might leave the carriage in a position close to the transfer position but not at the transfer position, which might not be obvious for an observer.

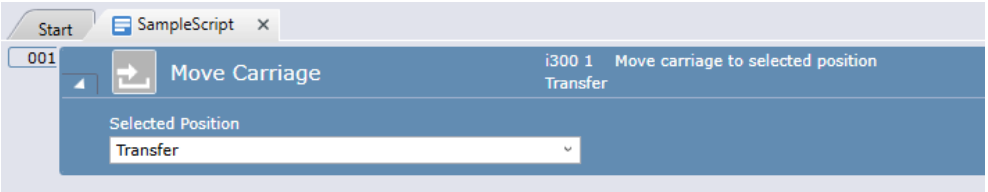


Fig. 268: Move Carriage (i300)

Before executing any Resolvex i300 command moving the carriage, it must be ensured that no robotic device obstructs the movement. In many cases, the FCA and RGA will move away sufficiently to freely move the carriage with labware. However, this is not guaranteed so in certain cases, the arm must be moved away by additional commands (see example illustration with **Move Arm** below).

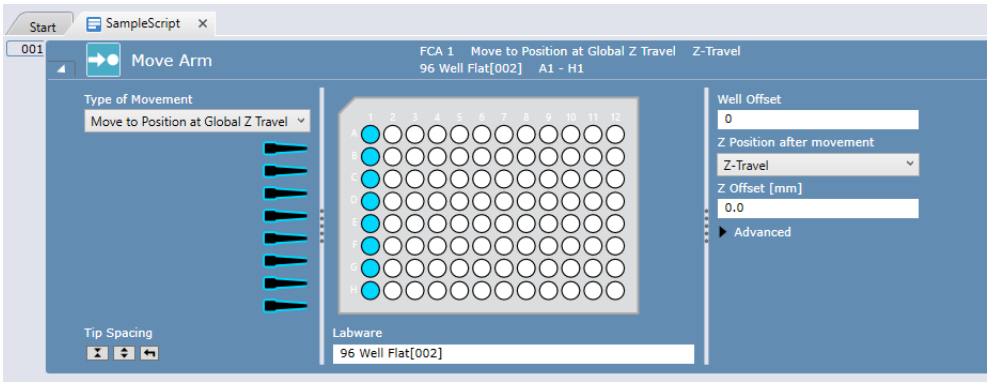


Fig. 269: Move Arm

11.9.2.1.4 Initialization

If RGA with eccentric fingers is used, place the RGA to the left of the module before initialization.

11.9.3 i300 Command Reference

The following commands are available for the Resolvex i300.

11.9.3.1 Apply Pressure

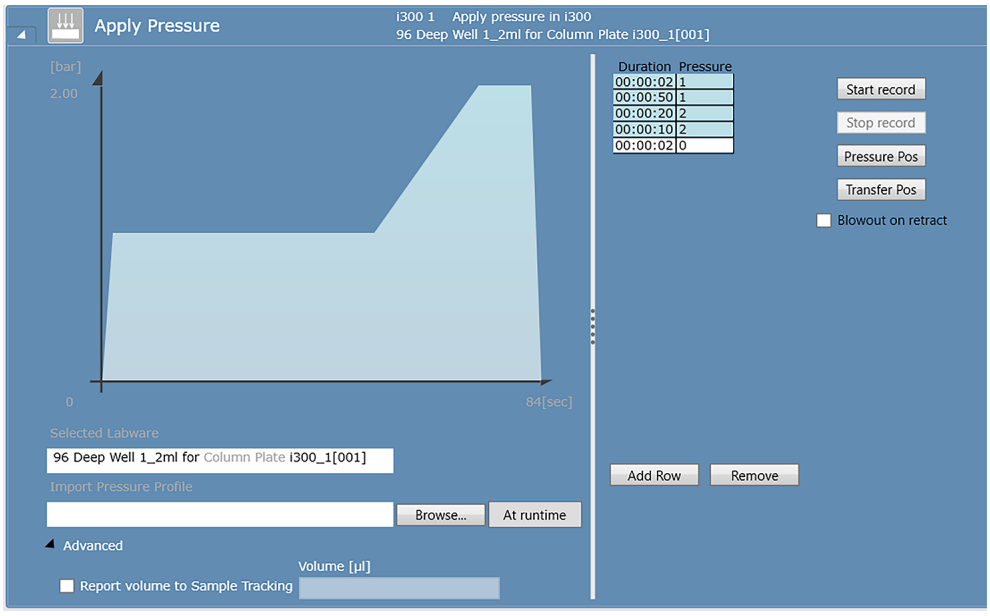


Fig. 270: Apply Pressure

The carrier is moved to pressure application position, pressure manifold travels down, labware is sealed with constant force, pressure is applied, pressure head travels up by 5 mm . The pressure profile is defined in the profile table with the duration for each step and the to be applied pressure. Pressure unit bar or psi is defined in the system configuration.

Parameters

Graph: The pressure executed in time.

Profile Table: Shows the pressure executed in time and the different durations. It is possible to add and remove rows.

Start Record: Starts the recording of a new pressure profile. All data will be overwritten. After starting the recording, the parameters can be changed with the Move Tool. Each click in the Move Tool creates a new row in the table. Note: Pressure profiles with 0 duration in one step are not executable. This can happen on recording the profile. Please edit the profile to have longer durations per step.

Stop Recording: Ends the recording.

Pressure Pos: Moves the carriage to the pressure position to start a recording.

Transfer Pos: Moves the carriage to the Transfer position to allow access to the labware.

Select Labware: In the selected Labware box, either type your expression or click the labware displayed in the worktable editor window (Worktable click).

Import Pressure Profile:

- **Browse**: Click button to select existing pressure profile file.
- **At runtime**: When button is in this state, pressure profile is imported during execution of the command.
- **Import once**: After you click on ‘At runtime’ it switches to ‘Import once’. When you browse for pressure profile file with ‘Import once’, the pressure profile is immediately loaded in the command UI.

Report volume to Sample Tracking: A checkbox along with variable input field is provided to report volume via sample tracking. The volume shall be reported to the sample tracking server and available in the standard views like quick viewer and sample tracking reports.

- **Unchecked**: All volume of top cavity is transferred to bottom cavity and adds warning in sample tracking report for both filter and collection plate.
- **Checked**: Transfer specified volume (variable input field) from top cavity to bottom cavity without warning and keep the difference, if greater than 0, in the top cavity.

Volume: A variable input field to specify the volume.

11.9.3.2 Dispense

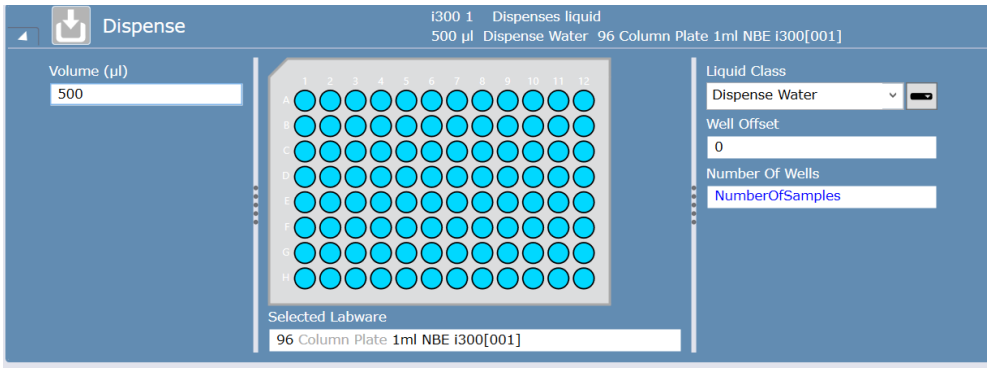


Fig. 271: Dispense

Dispenses the desired volume into selected wells of the selected labware. The Liquid Class in combination with the container configuration defines the solvent bottle used for the dispense.

Parameters

Volume: Enter the volume to dispense.

Labware: In the selected Labware box, either type your expression or click the labware displayed in the worktable editor window (Worktable click).

Wells: Select the wells of the labware to dispense to.

Well Offset: Select by how many wells the starting point of the currently selected wells is shifted. This is usually expressed as a variable.:

Liquid Class: Choose the required liquid class from the pull-down list or by means of an expression typed in (e.g., by using a variable defined within your script). The list only shows liquid classes that are suitable for the current command.

11.9.3.3 Wash

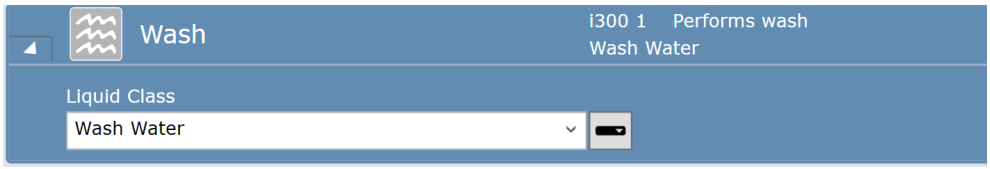


Fig. 272: Wash

Performs a wash into waste with the desired Liquid Class. The Liquid Class in combination with the container configuration defines the solvent bottle used for the wash. This command can either be used to prime the tubing from the bottle to the dispenser tip, to clean the dispenser or to prepare for a new liquid to fill the way from the 12-valve port to the tip of the dispenser channel with the later used solvent.

Parameters

Liquid Class: Choose the required liquid class from the pull-down list or by means of an expression typed in (e.g., by using a variable defined within your script). The list only shows liquid classes that are suitable for the current command.

11.9.3.4 Evaporate

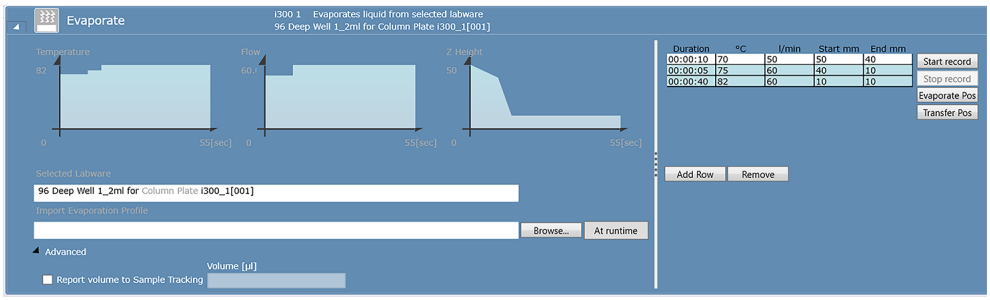


Fig. 273: Evaporate

Executes the evaporation profile on the selected labware.

Parameters

Graph: Shows the Temperature, Flow and Z-Hight over Labware bottom over the time of the evaporation profile.

Profile Table: Shows the duration of having the temperature, flow and Z-Hight (with start and end) which will be traveled) defined in the row of the table. It is possible to add and remove rows.

Start Record: Starts the recording of a new evaporation profile. All data will be overwritten. After starting the recording, the parameters can be changed with the move tool. Each click in the Move Tool creates a new row in the table. Note: Evaporation profiles with 0 duration in one step are not executable. This can happen on recording the profile. Please edit the profile to have longer durations per step.

Stop Record: Ends the recording.

Evaporate Pos: Moves the carriage to the evaporating position to start a recording.

Transfer Pos: Moves the carriage to the Transfer position to allow access to the labware.

Select Labware: In the selected labware box, either type your expression or click the labware displayed in the worktable editor window (worktable click).

Import Evaporation Profile:

- **Browse**: Click button to select existing evaporation profile file.
- **At runtime**: When button is in this state, evaporation profile is imported during execution of the command.
- **Import once**: After you click on ‘At runtime’ it switches to ‘Import once’. When you browse for evaporation profile file with ‘Import once’, the evaporation profile is immediately loaded in the command UI.

Report volume to Sample Tracking: A checkbox along with variable input field is provided to report volume via sample tracking. The volume shall be reported to the sample tracking server and available in the standard views like quick viewer and sample tracking reports.

- **Unchecked**: Set cavity volume to “0” and adds warning in sample tracking report.
- **Checked**: Set specified cavity volume (variable input field) without warning.

Volume: A variable input field to specify the volume.

11.9.3.5 Preheat Evaporator

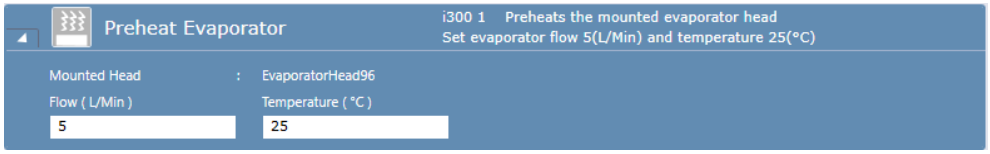


Fig. 274: Preheat Evaporator

Pre heats the evaporator to a temperature so it can be used quicker later in script. This can be used for example in a subscript to warm up the evaporator while doing something else to fasten the heat-up time before usage.

Parameters

Temperature: The temperature the evaporator should be heated up to.

Flow: The flow of air/gas to be used while heating up.

11.9.3.6 Prepare Stack

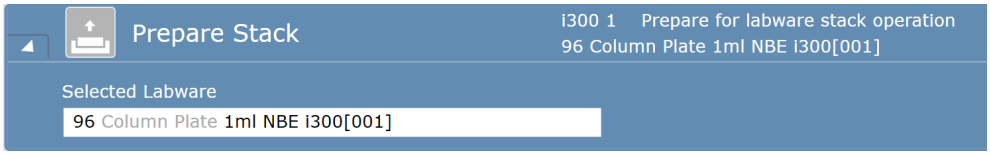


Fig. 275: Prepare Stack

Lift the selected Labware (usually the filtration labware) to make space for a collection labware. With this the droplet protector will be moved under the lifted plate to prevent cross contamination while placing the collection labware below.

Parameters

Labware: In the selected Labware box, either type your expression or click the labware displayed in the worktable editor window (Worktable click).

11.9.3.7 Stack

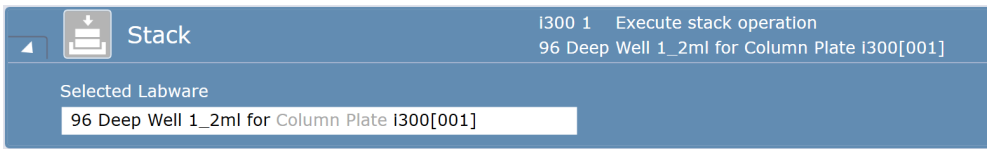


Fig. 276: Stack

Place the lifted Labware (usually the filtration labware) on the selected Labware.

Parameters

Labware: In the selected Labware box, either type your expression or click the labware displayed in the worktable editor window (Worktable click).

11.9.3.8 Prepare Unstack

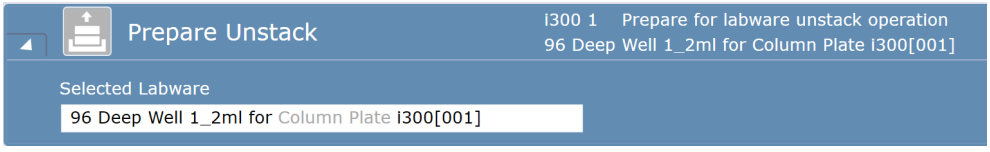


Fig. 277: Prepare Unstack

Lift everything over the selected labware to be able to remove the collection Labware. The droplet protector will prevent cross contamination while removing the collection labware.

Parameters

Labware: In the selected Labware box, either type your expression or click the labware displayed in the worktable editor window (Worktable click). *Select the lower labware for prepare unstack (usually the collection labware).*

11.9.3.9 Unstack

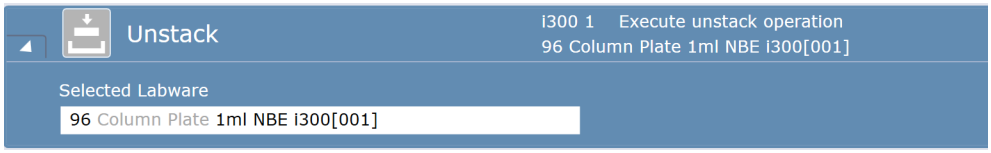


Fig. 278: Unstack

Place the selected labware on the top of the labware on the carriage or the carriage itself.

Parameters

Selected Labware: In the selected Labware box, either type your expression or click the labware displayed in the worktable editor window (Worktable click) (usually the filtration labware).

11.9.3.10 Mount Pressure Manifold

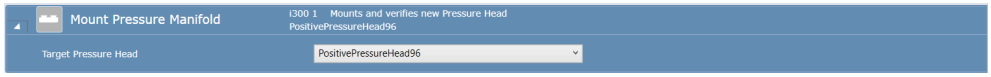


Fig. 279: Mount Pressue Manifold

The Z-Manifold Axis will move upwards to release the springs. After the command the Manifold can be manually removed.

11.9.3.11 Unmount Pressure Manifold



Fig. 280: Unmount Pressure Manifold

The Z-Manifold Axis will move upwards to release the springs. After the command the Manifold can be manually removed.

11.9.3.12 Mount Evaporator Manifold

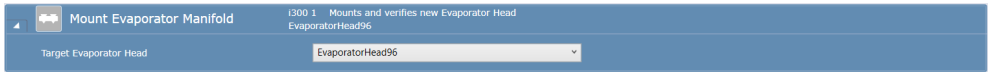


Fig. 281: Mount Evaporator Manifold

An NeedleStorageBox with the selected Manifold needs to be present on the Carriage, both physically and in virtually in the software.

Moves the Carriage back to evaporator and mounts the manifold. After this it will check if the manifold matches the selected manifold.

Parameters

Target Evaporator Head: The manifold which is mounted after the command to ensure the correct manifold is mounted.

11.9.3.13 Unmount Evaporator Manifold



Fig. 282: Unmount Evaporator Manifold

An empty NeedleStorageBox needs to be present on the Carriage, both physically and virtually in the software.

This command will move the Carriage back to the evaporator and unmounts the current Evaporating Manifold and moves the stack to the transfer position to exchange the Manifold manually.

11.9.3.14 Move Carriage



Fig. 283: Move Carriage

Transfer Position: Moves to the Front position. Only this position is accessible by the FCA and RGA. Choose and always move to Transfer Position in order to access the Carriage with the FCA or RGA.

Pressure: Moves the Carriage under the Pressure Manifold

Evaporator: Move the Carriage under the Evaporator Manifold

Restack: Position where the lift can freely move.

Parameters

Moves the carriage to the selected position. It can be choosen from the following positions:

Selected Position: The defined positions of the carriage in the i300.

11.9.3.15 Internal Light



Fig. 284: Internal Light

Switches on and off the internal light in the Resolvex i300. This improves the visibility of the liquid inside the labware to make creation of the pressure- and evaporation profiles easier. Moreover, it is recommended to remove any labware in front of the labware under the evaporator to improve visibility during development processes.

Parameters

Selected State: Selection if the light should be on or off.

11.9.3.16 Set Bottle Volume



Fig. 285: Set Bottle Volume

Sets the current volume of the selected port to the entered volume.

Parameters

Selected Port: The port of the solvent bottle to set the volume.

Volume: The volume to set the bottle to.

11.9.3.17 Refill Bottle

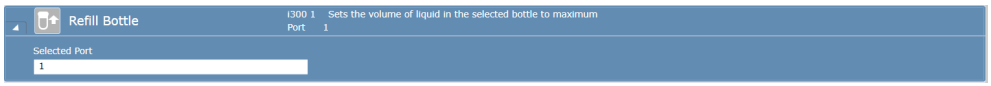


Fig. 286: Refill Bottle

Sets the current volume of the selected port to the maximum.

Parameters

Selected Port: The port of the solvent bottle to set the volume.

11.9.3.18 Replace Seal



Fig. 287: Replace Seal

The command to reset the counter of Pressure profiles in Introspect after the seal is replaced.

11.10 Fluent ID

The commands for this module are explained in this document. For the operating workflow with the Fluent ID module refer to the Fluent Operating Manual (refer to section Reference Documents).

The system allows loading Tube Runners during a run via the Start Loading command. After the command is finished (operators clicks continue), the barcodes will be associated with the appropriate tubes. The loaded runners are supervised from that point forward and are not allowed to be removed until an Allow Removal command has been used.

Once the IDs are set by completion of the Start Loading command, all the standard FluentControl barcode handling is available. The ID slots of the tubes can be read via the GetAttribute function. The barcodes can be viewed via the Infopad. The LabwareCount function can return the number of scanned tubes which can in turn be used to impact things like the sample count of a Sample Transfer command.

Worklists can target individual tubes based on barcodes, instead of absolute position. When using worklist with Fluent ID Runners, be sure your destination labware is in common range of all eight flexible channel arm tips. This will allow FluentControl to always find a tip capable of making the transfer.

Using Fluent ID to capture tube barcodes followed by Sample Tracking Register Labware commands is a good way to track all barcodes and pipetting actions from the tubes to subsequent labware. It is only necessary to register the runners after

the Start Loading command has been used and before the first pipetting step takes place. After all pipetting steps are done, use the Generate Report command and target the destination labware to get an output of all IDs and pipetting actions.

NOTICE

Convention

The following special characters must not be used in barcodes: ! % & ~ ( ) = | as well as carriage return and line feed.

11.10.1 Barcode Types Definition

For each barcode type, up to five lengths can be configured. If a '0' is entered into the length field, the field is ignored. Barcode types and lengths not configured get an **unreadable barcode** error when scanned by the Fluent ID. It is not possible to ignore barcode length.

Code 128 is highly recommended for optimal reading confidence, accuracy and compactness.

- On the **Settings** menu, click **Configure System** and navigate to **Drivers** > **Fluent ID**.

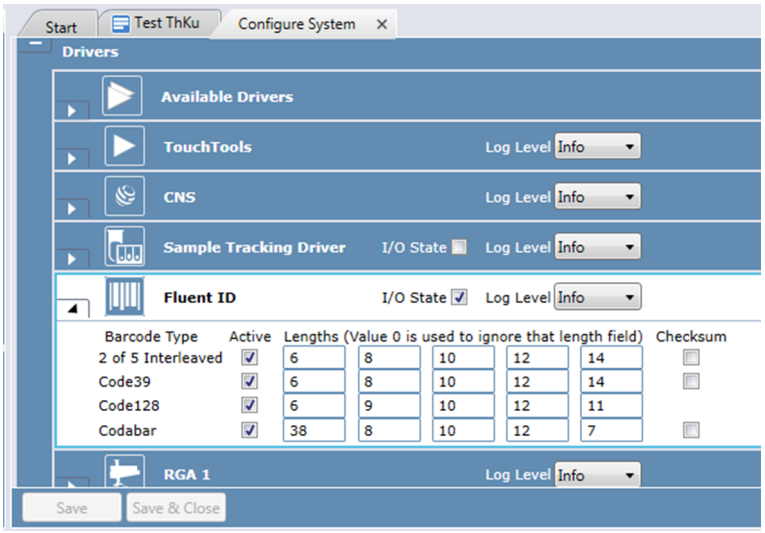


Fig. 288: Configure System

11.10.2 Worktable Setup

The segment needs to be loaded onto the worktable. There is one segment for the leftmost (Fluent ID Left) end of the instrument and one for any other position of the worktable (Fluent ID Middle). There are Loading ID specific tube runners that are needed to load on the segment.

For every connected device a corresponding segment must be placed on the virtual worktable. Every carrier on a virtual worktable must be associated with the driver, refer to section "Configuring Your Drivers" [▷ 199].

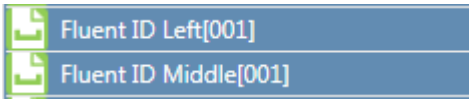


Fig. 289: Connected Fluent ID device

A reflector is required to be located on the first grid, right of each segment. Ensure that the worktable layout allows the reflector to be mounted or the reflector sticker to be affixed. All available tube runners, as listed in Fluent Operating Manual (refer to section Reference Documents) are available in FluentControl labware:



Fig. 290: Available Loading ID tube runners

NOTICE

Ensure that the Path Finder uses the right travel positions for the Loading ID

To ensure that Path Finder uses the right travel positions for the Loading ID, populate the virtual worktable with tube runners on every grid where a runner is expected for loading at any time in the run. This way Path Finder will move always in the safe distance, even in case the user placed a runner on a grid outside of a loading time frame.

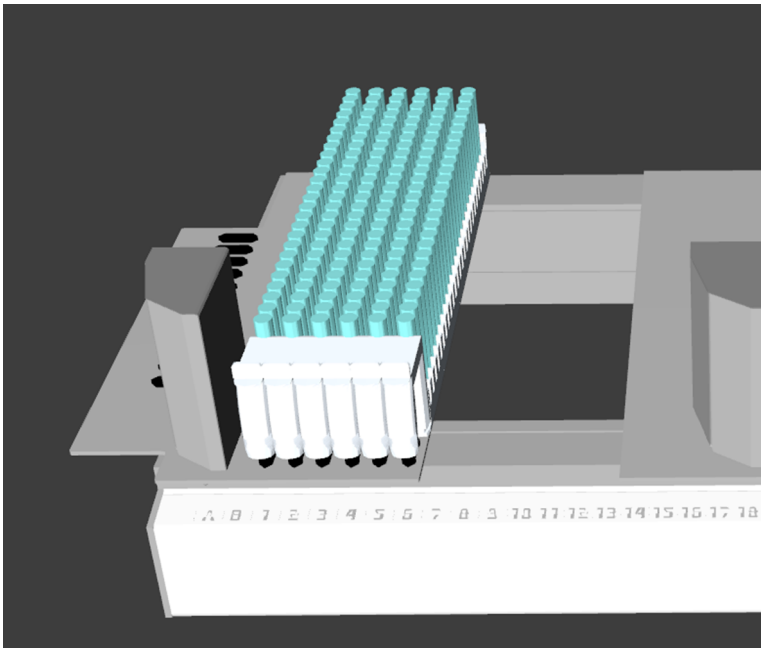


Fig. 291: Virtual worktable layout: runner occupy the 6 grids of the Fluent ID segment

The reflector is mandatory to position physically on the worktable. No additional configuration is needed in the software. The bounding boxes of the modules ensure that Path Finder will account for the presence of the reflector at run time.

11.10.3 Start Loading (FluentID)

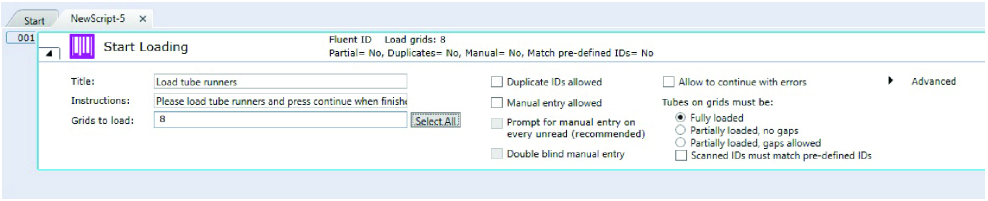


Fig. 292: Start loading command

The **Start Loading** command is used to activate the Fluent ID barcode scanner and invite the user to insert tube runners. Multiple Fluent ID modules can be used in a single **Start Loading** command. At the point where this command is set in the script, the system requests to load runners onto the Loading ID module for barcode scanning.

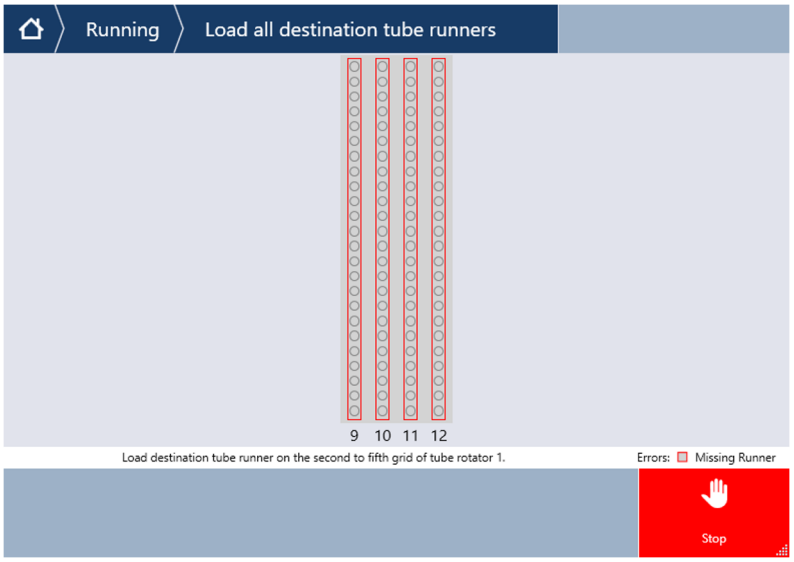


Fig. 293: Load all destination tube runners

When running in simulation mode the command will be skipped and no action is shown to the user.

An Active Stop will occur if a runner is removed from a grid lit in green after the completion on of the Start Loading command. The grid LED will turn red and an error message will be displayed on the touch monitor. If the LED turns red after an unexpected interruption (e.g., runner shifted inadvertently out of its stop position when sliding out a neighbor runner, run-time error dialogue), it must be fully unloaded and reloaded by the operator. The system will verify that the barcodes are the same as read during the initial loading and any mismatch will be displayed on the touch monitor.

*Therefore, it is recommended to set in the script an **Allow Removal** command as soon as the tubes and runners are no longer used for the process (e.g., liquid transfer is completed).*

One part of the **Start Loading** command dialog is for defining the location of Fluent ID runners and loading instructions. The second part shows all the options for error handling and expected behavior depending on application.

Title: Define a title for this command. The title is shown during the run on top of the touch monitor user interface.

Instructions: Define the operator instructions shown on touch monitor.

Grids to Load: Define the grids tube barcodes are to be read. Enter ranges or single grid positions into the textbox or click a tube runner on the virtual worktable. Select all, if barcodes are to be read on all available grids. Enter a single grid, individual grids separated by commas, or a range of grids separated by a hyphen (e.g., 1,2,4-6).

Duplicate IDs allowed:

*Do not use the **Duplicate IDs allowed** option with Sample Tracking.*

Sample Tracking reports are not designed to handle duplicate IDs.

If this is checked, duplicate IDs on one or moreFluent IDs are allowed. If not checked, the operator will get an error message on the touch monitor about duplicate barcodes. Their position will be flagged as errors on the touch monitor.

Manual Entry allowed: When manual barcode entry is checked, the operator is allowed to manually enter barcodes in case of unread barcodes. The touch monitor prompts the user for the unread barcode. The user may enter barcodes directly. The operator may unload the runner. Unread barcodes flagged on the touch monitor can be scanned with a handled barcode reader or typed in. The runner is then loaded again.

Prompt for manual entry on every unread (recommended): If checked, every unread barcode must be manually entered upon loading a runner. If unchecked, the operator will not be required to manually enter a barcode for an unread tube more than once.

Double blind manual entry: For higher security, double blind entry forces the operator to enter the same barcode twice.

When entering barcodes manually, make sure that white spaces are not added by mistake before or after the barcode. Checks for duplicate barcodes take white spaces into account. For example, barcodes " abc123" and "abc123" will not be considered duplicates.

Allow operator to continue with errors

If this box is checked, the run can continue with unresolved errors such as unread barcodes, missing tubes, duplicates, missing runners. LEDs will turn green as soon as the operator continued the run.

Tubes on grid must be:

- **Fully loaded**: Tube and no runner gaps are allowed.
- **Partially loaded, no gaps**: The tube runner may contain more tubes than physical tube runner. The first tube defined in the worktable must be present and no tube or runner are allowed until the last tube physically present in the furthest runner on the right.
- **Partially loaded, gaps allowed**: Tube and runner gaps in runners are allowed.
- **Scanned IDs must match pre-defined IDs**: If this box is checked, the scanned barcodes on the tubes will be verified, if they match the pre-defined barcodes from the barcode attribute of the corresponding tube. Pre-defined barcodes can be set to the tubes via the SetAttribute() function in advance of the scan. Operator is alerted to mis-matching tube-barcodes during loading. If this option is selected, "Allow to continue with errors" is grayed out.

Advanced

Additional settings can be defined in the Advanced tab.

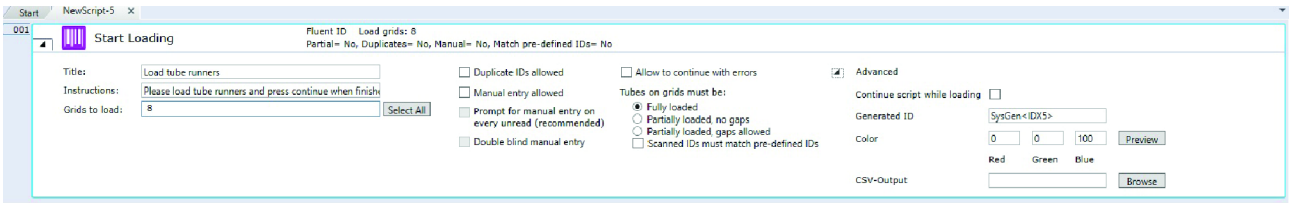


Fig. 294: Advanced settings

Continue script while loading: If checked other steps in the script will run while the operator loads the runners. For every **Start Loading** command a corresponding **Wait for Finish** command must be included right after in the FluentControl Script.

NOTICE

Allowing the script to continue brings a risk of collision.

Allowing the script to continue brings a risk of collision. Ensure that the next steps in the scripts are performed away from the grids.

- Always include a **Wait for Finish** command before the steps in the script requiring arm access to the grids (e.g., pipetting from the barcoded tubes or transporting a tube out of a tube runner).

Generate ID:

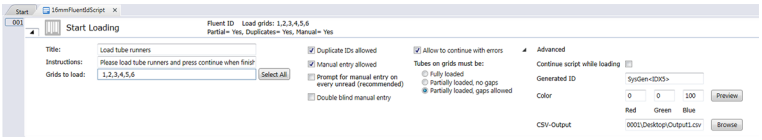


Fig. 295: Generate ID

In case of an unread barcode, with the option **Allow to continue with errors** selected, a unique ID is automatically generated. A default will be suggested with prefix and sequential numbering with <IDX5> that adds a 5-digit sequential number (e.g., 00001). The prefix and sequential number can be changed. If only a text is defined, it will add a single sequential number to that (e.g., barcode-1). The prefix and sequential numbering can be customized.

Preview: Click on the preview button, then load a tube runner on a Fluent ID grid to see the impact of the chosen settings on the barcode reading results on the touch monitor and on the LED color.

Color: Define a custom color for the grid LEDs signaling the invitation to load. Default is set to blue. Preview will show the configured LED color for 5 seconds.

CSV-Output: Define a path and file name for the automatic generation of a CSV-File containing all read tube barcodes. In the current FluentControl version, the headers of the CSV output file do not always correspond to the data in the corresponding column. The file contains the following data for each position in the runner in comma separated columns:

GridPos	Grid position of the runner
SiteOnGrid	Always ‘1’ for tube runners
TubeNumber	Tube position in the runner
ConfigName	Barcode of the runner (the string from Custom Attribute _tUbeRunnerID)
RackLabel	Always ‘N/A’, currently not used
CarrierBarcode	Barcode of the tube, ‘$$$’ if no tube was detected or ‘***’ if the barcode could not be read
SRCRackID	Always ‘N/A’, currently not used

11.10.4 Allow Removal (FluentID)

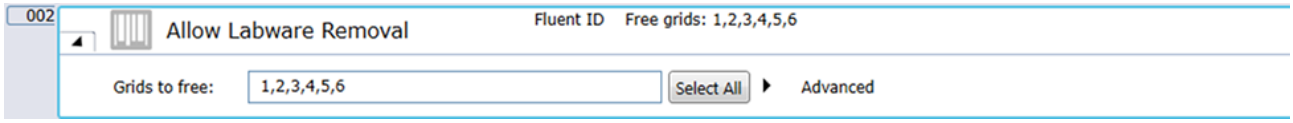


Fig. 296: Allow removal command

Allow operator to unload runners and tubes during the course of a run. The blinking blue (or customized) LED signals the runners that are ready to be unloaded. grids are allowed to unload. The monitoring for these grids will be deactivated. There is no dialog displayed on the touch monitor.

The tubes will be flagged as not present in the worktable. This allows the LabwareCount function to work correctly. Take care not to use Allow Removal before all needed pipetting actions have been executed.

*After a **Allow Removal** command, insert a **Wait for Finish** before the insertion of a second **Start Loading** command in the FluentControl script.*

Grids to free: Define the grid positions where unloading is allowed. Enter ranges or single grid positions into the textbox. Select all, if all available grids shall be unloaded. Enter a single grid, individual grids separated by commas, or a range of grids separated by a hyphen (e.g., 1,2,4-6).

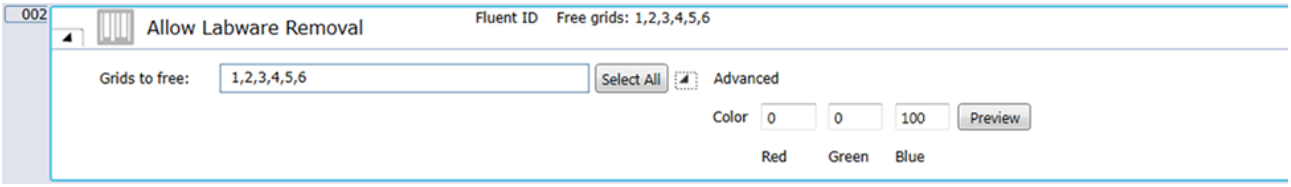


Fig. 297: Grids to free

Advanced

Define a custom color for the grid LEDs to be used for removing is allowed.

NOTICE

Avoid unexpected Active Stops

To avoid unexpected Active Stops of the system, place an **Allow Removal** command as soon as all required steps (e.g., liquid transfer) are successfully executed with the selected runners and tubes.

11.10.5 Wait for Finish (FluentID)

Wait for Finish: The **Wait for Finish** command if the option **Continue** while **Loading is selected**. It is used after a **Start Loading** command to pause the FluentControl Script until the loading of the runners is completed.

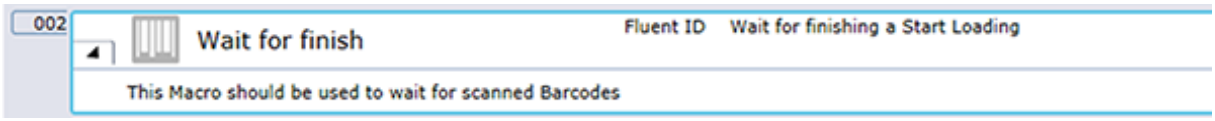


Fig. 298: Wait for finish command

11.11 FRIDA Reader

11.11.1 Measurement Group

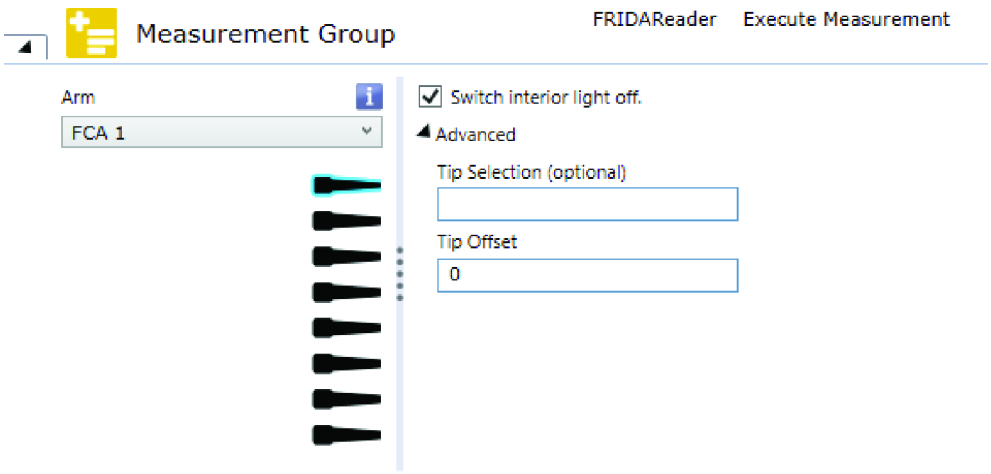


Fig. 299: Measurement Group

The Measurement Group command incorporates the whole measurement workflow and makes sure, that all essential preparation steps are executed at the beginning and all essential finalization steps are executed at the end. Prior to the execution of the command a suitable tip needs to be mounted.

In the beginning, a background measurement is executed for which a single channel with a mounted tip is moved into the FRIDA Reader to block out ambient light. This ambient light could affect the background measurement.

After the background measurement, the measurement group iterates over the embedded script statements.

An error detection is executed at the end of the group to determine whether a FRIDA Reader steploss error has occurred. If it has, the user is notified that all measurement results after the last background measurement may be invalid.

Parameters

Arm: All available FCA pipettors are shown in a dropdown list. The user can choose the pipettor they wish to use from this list.

Tip Selection: Select the tip which shall be used for the background measurement

Tip Selection (optional): The Tip Selection (optional) parameter is resolved as an integer (integer variables and expressions are allowed) and interpreted then as a bitmask. Bitmasks are a very resource friendly means of storing settings. The tip selection in the script command is calculated considering both the Tip Selection (optional) bitmask and the tips selected via GUI (binary AND). Thus, only tips selected in both masks will be used for pipetting.

The Tip Selection (optional) mask is specified as a numeric, bit-coded value (tip1 = 1, tip2 = 2, tip3 = 4, tip4 = 8, tip5 = 16, tip6 = 32, tip7 = 64, tip8 = 128). Use the sum of the bit-coded values to select more than one tip. In the GUI, tip1 is the upmost tip.

Tip Offset: This parameter lets you modify the tip selection for the measurement command. The default is 0, which deactivates the offset. A positive value increments the tip number(s) which are specified in the measurement command. For example, if the measurement command specifies tips 2 and 3 and you set this parameter to 1, tips 3 and 4 will be used instead. Negative values of the parameter shift the tip selection in the opposite direction.

Switch Interior Light Off: This option will turn off the interior light indefinitely. If the interior light is on and the checkbox is checked, the interior light will be switched back on at the end of the measurement group command.

11.11.2 Background Measurement

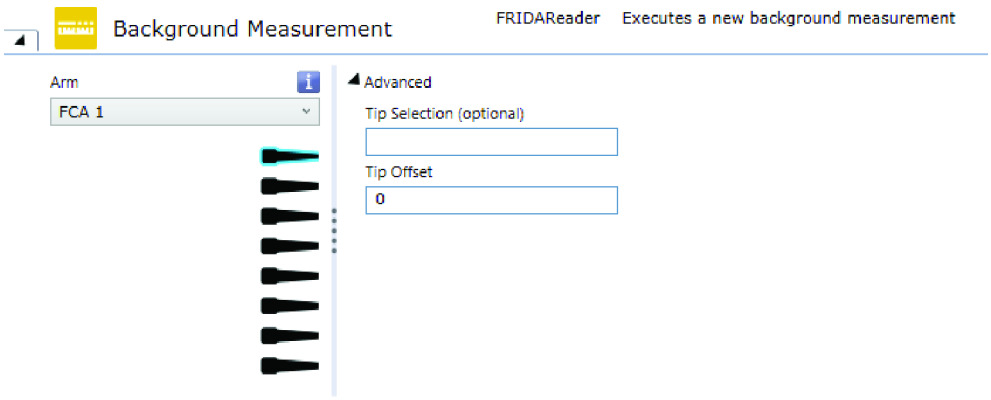


Fig. 300: Background Measurement

The Background measurement command can optionally be used to execute a standalone background measurement with a single channel. Prior to the execution of the command a suitable tip needs to be mounted.

Parameters

Arm: All available FCA pipettors will be shown in a dropdown list. The user can choose the pipettor they wish to use from this list.

Tip Selection: Select the tip which shall be used for the background measurement

Tip Selection (optional): The Tip Selection (optional) parameter is resolved as an integer (integer variables and expressions are allowed) and interpreted then as a bitmask. Bitmasks are a very resource friendly means of storing settings. The tip selection in the script command is calculated considering both the Tip Selection (optional) bitmask and the tips selected via GUI (binary AND). Thus, only tips selected in both masks will be used for pipetting.

The Tip Selection (optional) mask is specified as a numeric, bit-coded value (tip1 = 1, tip2 = 2, tip3 = 4, tip4 = 8, tip5 = 16, tip6 = 32, tip7 = 64, tip8 = 128). Use the sum of the bit-coded values to select more than one tip. In the GUI, tip1 is the upmost tip.

Tip Offset: This parameter lets you modify the tip selection for the measurement command. The default is 0, which deactivates the offset. A positive value increments the tip number(s) which are specified in the measurement command. For example, if the measurement command specifies tips 2 and 3 and you set this parameter to 1, tips 3 and 4 will be used instead. Negative values of the parameter shift the tip selection in the opposite direction.

11.11.3 Buffer Drop Measurement

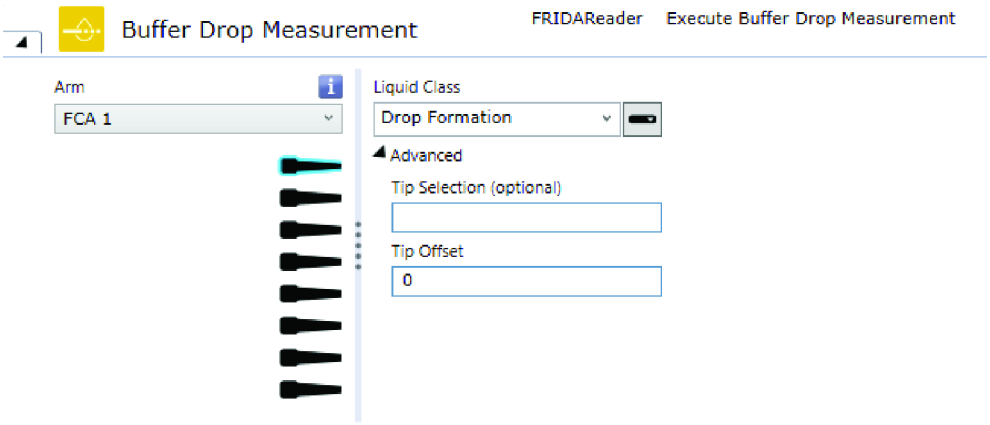


Fig. 301: Buffer Drop Measurement

The Buffer Drop Measurement command can be used to execute a buffer measurement with a single channel. It is only executable within a measurement group. Prior to the execution of the command a suitable tip needs to be mounted and the liquid to be measured needs to be aspirated from the source plate.

The following well attributes will be used to store the measurement results at the source plate:

Tab. 9: Buffer Drop Measurement

Well attribute	Description	Default value
MeasurementError	The error description, if an error has occured	*empty*
MeasurementWarning	The warning description, if a warning has occurred	*empty*
OD230	The optical density at a wavelength of 230 nm	0
OD260	The optical density at a wavelength of 260 nm	0
OD280	The optical density at a wavelength of 280 nm	0
OD320	The optical density at a wavelength of 320 nm	0
Ratio_260_230	The ratio of the optical densities at a wavelength of 260 nm and 230 nm.	0
Ratio_260_280	The ratio of the optical densities at a wavelength of 260 nm and 280 nm	0

Well attribute	Description	Default value
Pathlength	The path length, which corresponds to the diameter of the liquid drop	0

Parameters

Arm: All available FCA pipettors will be shown in a dropdown list. The user can choose the pipettor they wish to use from this list.

Tip Selection: Select the tip which shall be used for the buffer drop measurement.

Liquid Class: The liquid class used to create and reaspirate the hanging drop.

Tip Selection (optional): The Tip Selection (optional) parameter is resolved as an integer (integer variables and expressions are allowed) and interpreted then as a bitmask. Bitmasks are a very resource friendly means of storing settings. The tip selection in the script command is calculated considering both the Tip Selection (optional) bitmask and the tips selected via GUI (binary AND). Thus, only tips selected in both masks will be used for pipetting.

The Tip Selection (optional) mask is specified as a numeric, bit-coded value (tip1 = 1, tip2 = 2, tip3 = 4, tip4 = 8, tip5 = 16, tip6 = 32, tip7 = 64, tip8 = 128). Use the sum of the bit-coded values to select more than one tip. In the GUI, tip1 is the upmost tip.

Tip Offset: This parameter lets you modify the tip selection for the measurement command. The default is 0, which deactivates the offset. A positive value increments the tip number(s) which are specified in the measurement command. For example, if the measurement command specifies tips 2 and 3 and you set this parameter to 1, tips 3 and 4 will be used instead. Negative values of the parameter shift the tip selection in the opposite direction.

11.11.4 Sample Drop Measurement

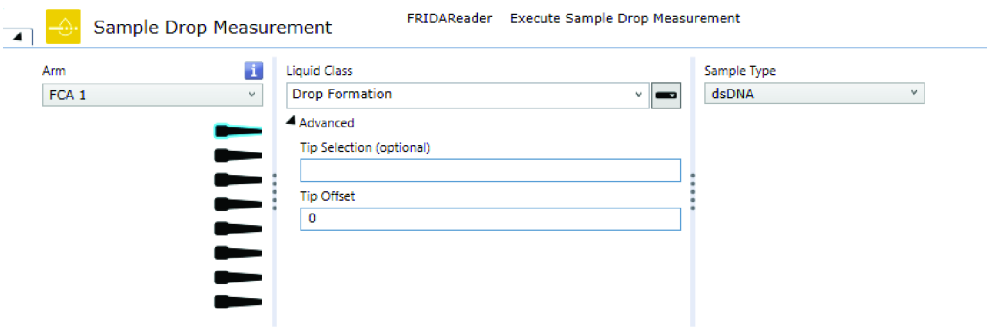


Fig. 302: Sample Drop Measurement

The Sample Drop Measurement command can be used to execute a sample measurement with a single channel. It is only executable within a measurement group. Prior to the execution of the command a suitable tip needs to be mounted and the liquid to be measured needs to be aspirated from the source plate.

The following well attributes will be used to store the measurement results at the source plate:

Tab. 10: Sample Drop Measurement

Well attribute	Description	Default value
Concentration	The measured concentration	-1
MeasurementError	The error description, if an error has occured	*empty*
MeasurementWarning	The warning description, if a warning has occurred	*empty*
OD230	The optical density at a wavelength of 230 nm	0
OD260	The optical density at a wavelength of 260 nm	0
OD280	The optical density at a wavelength of 280 nm	0
OD320	The optical density at a wavelength of 320 nm	0
Ratio_260_230	The ratio of the optical densities at a wavelength of 260 nm and 230 nm.	0
Ratio_260_280	The ratio of the optical densities at a wavelength of 260 nm and 280 nm	0
Pathlength	The path length, which corresponds to the diameter of the liquid drop	0

Parameters

Arm: All available FCA pipettors will be shown in a dropdown list. The user can choose the pipettor they wish to use from this list.

Tip Selection: Select the tip which shall be used for the sample drop measurement.

Liquid Class: The liquid class used to create and reaspirate the hanging drop.

Tip Selection (optional): The Tip Selection (optional) parameter is resolved as an integer (integer variables and expressions are allowed) and interpreted then as a bitmask. Bitmasks are a very resource friendly means of storing settings. The tip selection in the script command is calculated considering both the Tip Selection (optional) bitmask and the tips selected via GUI (binary AND). Thus, only tips selected in both masks will be used for pipetting.

The Tip Selection (optional) mask is specified as a numeric, bit-coded value (tip1 = 1, tip2 = 2, tip3 = 4, tip4 = 8, tip5 = 16, tip6 = 32, tip7 = 64, tip8 = 128). Use the sum of the bit-coded values to select more than one tip. In the GUI, tip1 is the upmost tip.

Tip Offset: This parameter lets you modify the tip selection for the measurement command. The default is 0, which deactivates the offset. A positive value increments the tip number(s) which are specified in the measurement command. For example, if the measurement command specifies tips 2 and 3 and you set this parameter to 1, tips 3 and 4 will be used instead. Negative values of the parameter shift the tip selection in the opposite direction.

Sample Type: The type of sample which can be analyzed with the FRIDA Reader. The following values will be included in the dropdown:

- RNA
- dsDNA

11.11.5 GetLastFridaReaderMeasurementResult (Parser Function)

The function “GetLastFridaReaderMeasurementResult” can be used to retrieve the last measurement results from the FRIDA Reader. The function needs the name of the desired measurement result as an argument.

The following measurement results can be retrieved:

Tab. 11: GetLastFridaReaderMeasurementResult (Parser Function)

Measurement result	Description	Default value
MeasurementError	The error description, if an error has occured	*empty*
MeasurementWarning	The warning description, if a warning has occurred	*empty*
Concentration	The nuclide acid concentration in ng/µl	-1
OD230	The optical density at a wavelength of 230 nm	0
OD260	The optical density at a wavelength of 260 nm	0
OD280	The optical density at a wavelength of 280 nm	0
OD320	The optical density at a wavelength of 320 nm	0
Ratio_260_230	The ratio of the optical densities at a wavelength of 260 nm and 230 nm.	0
Ratio_260_280	The ratio of the optical densities at a wavelength of 260 nm and 280 nm	0
Pathlength	The path length, which corresponds to the diameter of the liquid drop	0

11.12 Tube Rotator

The default Tube Rotator wash station is designed for piercing tips to be used with liquid FCA and not for use with Air FCA or other configurations. In case you have to operate an (additional) Air FCA (or other) please make sure that this arm is using another appropriate wash station accessible for washing and used as home position, otherwise you might run in a hardware crash.

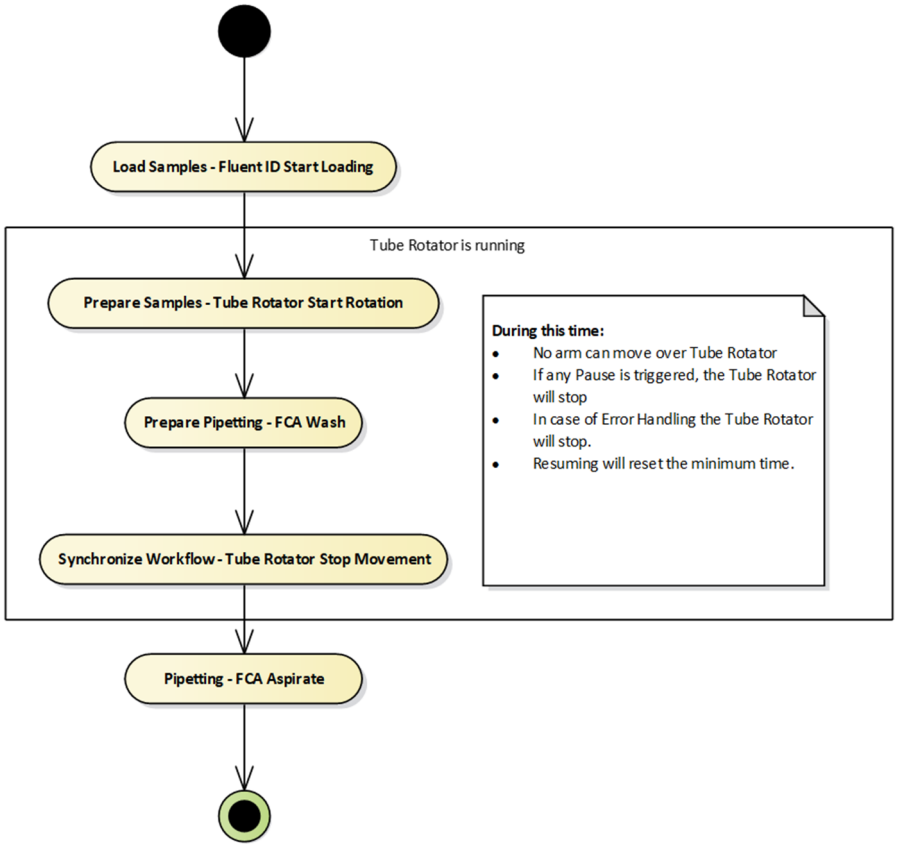


Fig. 303: act Assay Workflow

11.12.1 Start Oscillation

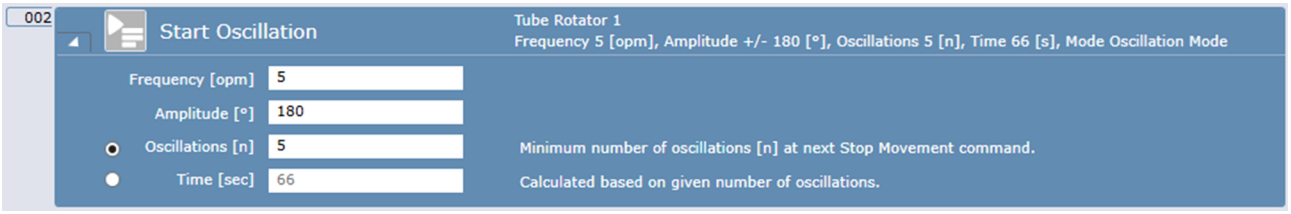
It executes a continuous and symmetric oscillation until stop is called.

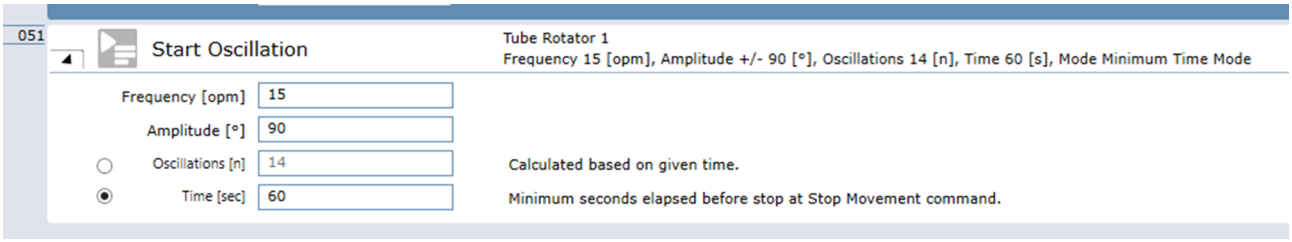
Parameter	Unit	Min and Max Values
Frequency	[OPM]	0.1–15
Amplitude	[°]	1–180 +/- from home
Option 1: Number of Oscillations	[number]	1–10.000
Option 2: Oscillation time	[seconds]	Time for one full swing - 432000

Either **Number of Oscillations** or **Oscillation time** can be entered, depending on the radio button selected. The corresponding value for the other parameter will then be calculated. Please notice that there is some extra time at the beginning and the end of each oscillation for final movements. Therefore, the calculation is not fully linear.

The time and number settings are minimum values, which are minimally executed if no immediate or active stop is triggered.

The command does not wait for the completion of the minimum oscillating time/ number, it just starts the move and then the command ends. For stopping the movement a **Stop Movement Command** is required. The user is responsible to add Stop Movement command in a script and to choose the appropriate mode for the stop.





11.12.2 Start Rotation

The **Start Rotation** command starts rotating the drum. The user can enter number of revolutions or the minimum rotation time depending on the radio button setting.

The command has the following parameters:

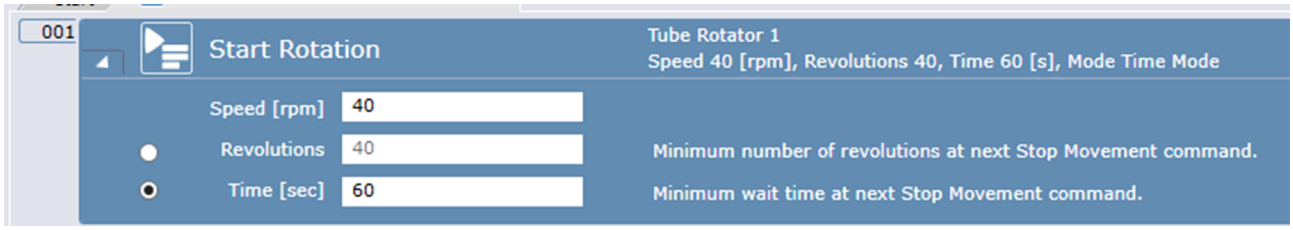
Parameter	Unit	Min and Max Values
Speed	[RPM]	0.1–60
Mode	**Revolutions** setting or **Time** setting	Activates option 1 or option 2
Option 1: Revolutions	[number]	one revolution
Option 2: Rotation time	[seconds]	Time for one rotation – 86400

No default values, all values should be entered by user.

The user can specify the speed. Additionally the number of revolutions or the total time the tube rotator should move. Depending on the selection the equivalent is calculated and displayed to the user.

The command does not wait for the completion of the minimum rotation time, it just starts the move and then the command ends. A following **Stop Movement** command is required to stop the rotation.

The user is responsible to add **Stop Movement** command in a script.



11.12.3 Stop Movement

Rotation and Oscillation movements are running in the background—like a parallel running subroutine—to synchronize the assay workflow (spinning and pipetting) the system provides the **Stop Movement** script command. Same concept like Fluent ID (Start Loading and Wait for Finish) or Touch Tools (Commands and Wait for Close).

The **Stop Movement** command stops a previously started **Rotation** or **Oscillation** command. It has two options:

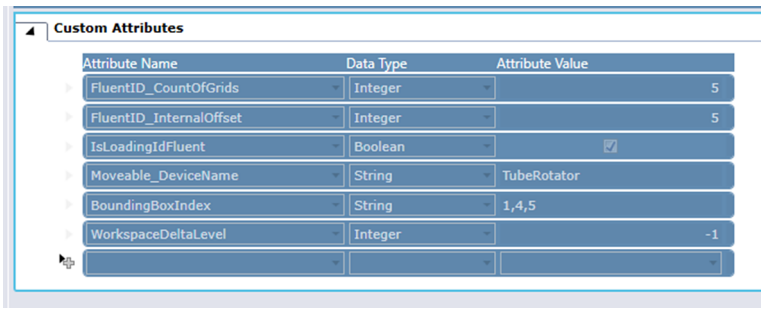
- Stop immediately
- Stop after minimum wait time or number of movements

One **Stop Movement** command needs to be called for each Tube Rotator movement. The next start of a movement may only be called after a previous move has been terminated with a **Stop Movement** command.



11.12.4 Custom Attributes

You can define custom attributes as parameter on the Tube Rotator to influence specific system connection with Loading Id or robotic actions



Parameters

BoundingBoxIndex: It defines a list of indexes to address configured Bounding boxes.

Moveable_DeviceName: Tube Rotator uses this custom attribute to identify their associated carrier

11.13 DeckCheck

The DeckCheck command provides the user with the functionality to execute a discrepancy check on the worktable during runtime. The DeckCheck command can be used multiple times in the script, where each execution is independent to the other executions, i.e., each script command has its own set of pictures which will be used by the DeckCheck function during execution. Discrepancies are not logged as errors. Therefore, the result of the discrepancy check will not have any impact on the result of the run.

11.13.1 DeckCheck Network Settings

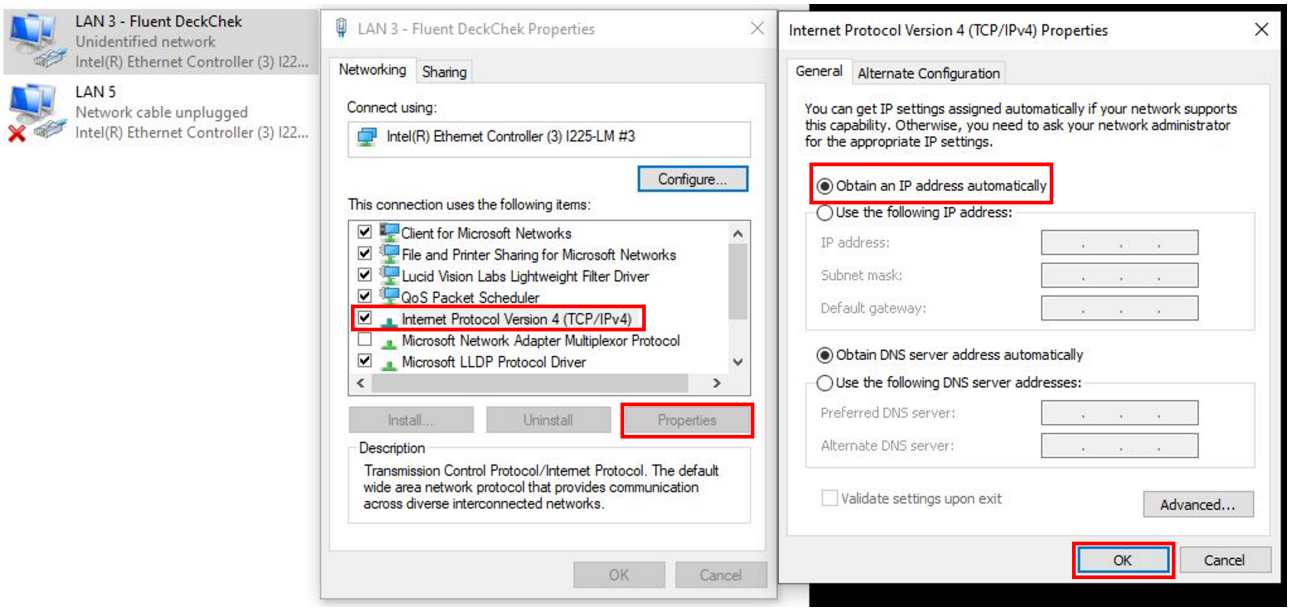


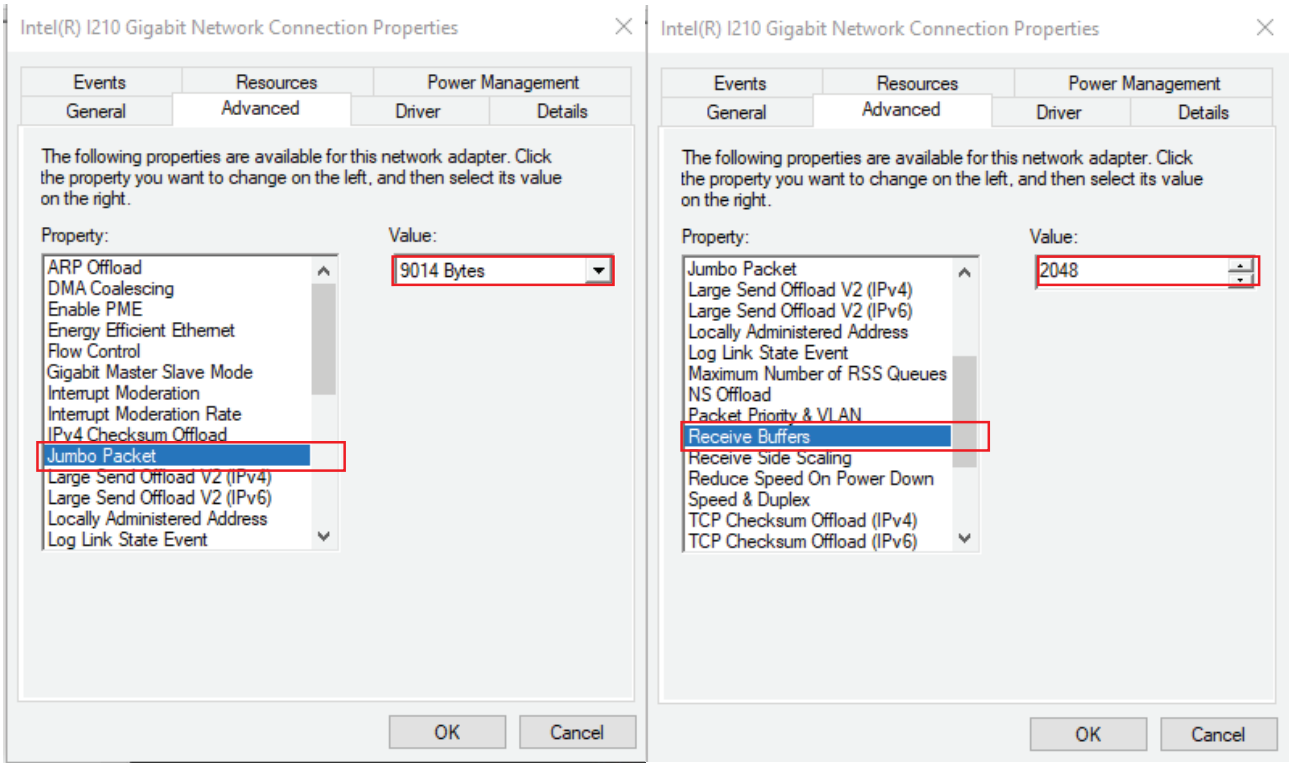
Fig. 304: DeckCheck Network Settings

The following settings are to be performed to use DeckCheck within FluentControl:

1. Open the properties of the respective LAN connection.
2. In the **Networking**tab select **Internet Protocol Version 4** and click **Properties**.
3. In the next window select **Obtain an IP address automatically** and click **OK**.
4. Confirm the changes in the LAN properties window by clicking **OK**.

11.13.2 Ethernet Adapter Settings

Then using DeckCheck, the following settings for the Ethernet adapter are recommended.



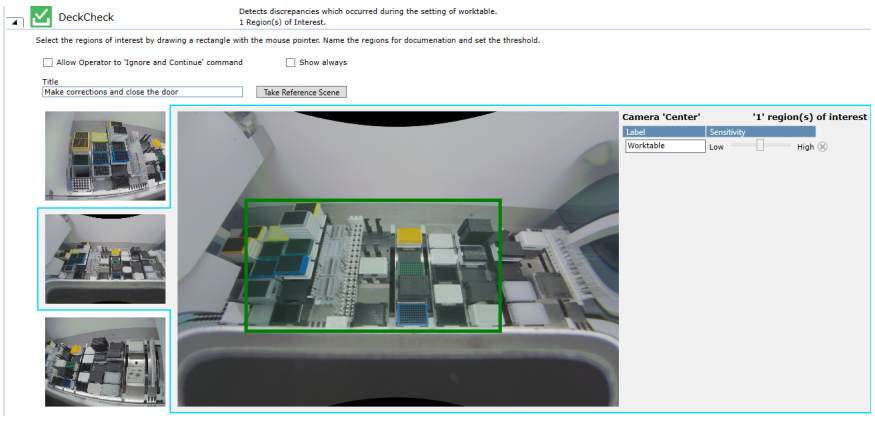
Select Jumbo Packet

Enabling Jumbo Frames on the Ethernet adapter allows a packet size of up to 9014 bytes to be set on the camera. The larger packet size will enable optimal performance on high-bandwidth cameras, and it usually reduces CPU load on the host system. Please note that to set a 9014-byte packet size on the camera, the Ethernet adapter must support a Jumbo Frame size of 9014 bytes or higher.

Adjust Receive Buffer Size

A receive buffer is the size of system memory that can be used by the Ethernet adapter to receive packets. Increasing the receive buffer size will increase the performance but also result in increased system memory usage.

11.13.3 Edit Time



Parameters

Title: The text entered here is displayed on the touch monitor at run time and can be customized to provide information to the operator.

Show always: If this option is selected, the DeckCheck screen is shown on the touch monitor during runtime whether any discrepancies were detected or not. If this option is not selected, the DeckCheck screen will only be shown in case discrepancies were found.

Take Reference Scene: A reference scene is a set of images (one for each camera) that serves as reference for a later comparison with live images. After clicking on **Take reference scene** each camera takes a picture of the worktable. The instrument may execute arm movements to move out of camera view.

Region(s) of Interest: A region of interest defines a section of the worktable where discrepancies are of interest. Discrepancies outside any regions of interest will be ignored. Each camera has its own perspective, and its own set of regions of interests. Select each camera on the left side of the main image to define regions of interest for that view. A region of interest can be defined by dragging a rectangle on the reference image with the mouse. This rectangle can be modified in size and can be deleted by pressing on the **X** on the region(s) of interest list on the right . Each region of interest can be given a name in and a confidence threshold.

Allow Operator to ignore and continue: If this option is selected, the Operator can press **Continue** at runtime without resolving detected discrepancies. If this option is not selected, the run can only be continued if all discrepancies have been corrected.

Confidence threshold (%): The discrepancy detection is based on neural networks, that predict discrepancies between the reference image and the live image with certain confidence. A low confidence threshold allows more potential discrepancies to be detected, however can also lead to more false positive detections meaning it is more likely to detect noise that is no real error. A high confidence threshold is the opposite, it will only show up discrepancies that have high certainty of being real errors, meaning it might miss some potential discrepancies. The values in % for each of the three levels can be set in the Settings > Configure System > DeckCheck.

Note:

- If no region of interest is defined for any camera, the entire image for each camera is considered as region of interest with a medium confidence threshold value.

- If at least one camera has a region of interest defined, all other cameras without any region of interest will be disregarded.

- For best discrepancy detection it is recommended to define the same region of interest for each camera.

- It is recommended to only cover critical areas of the worktable with a region of interest. Areas outside of the worktable like the side panels and the cabinet should not be covered to avoid detection of unwanted discrepancies.

There is no limit to the number of regions of interest that can be set. The number of regions set in one command will not affect the speed of operation.

11.13.4 Configuration

The DeckCheck settings section can be found under: **Configure System**, **General Settings** DeckCheck tab. A user with the corresponding user rights, can configure the confidence threshold settings of each of the three levels low, medium and high. The default recommended threshold values based on empirical data are 80%, 90%

and 95% for low, medium and high confidence respectively. For lower thresholds, reliability of a correct detection result decreases drastically and thresholds below 50% are considered to not add much value anymore to a correct detection.

Disable DeckCheck: If this option is selected, all DeckCheck commands in the script will be ignored and FluentControl just jumps to the next line without executing the command.
In **Configure System**, **Devices**, **Home positions**, the positions the arms move to in order not to block the cameras' view during image acquisition may be defined

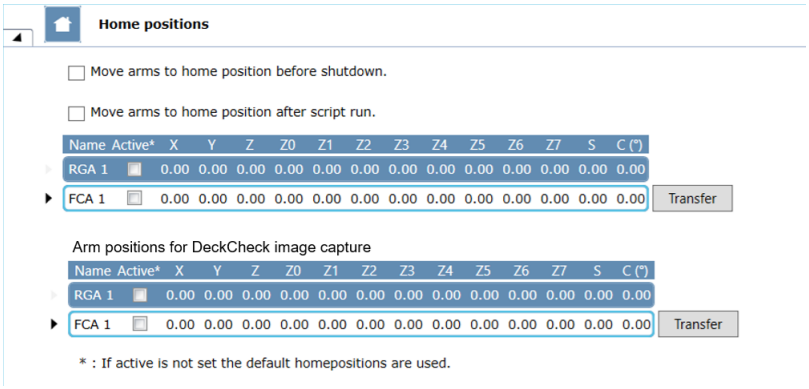


Fig. 305: Home positions

11.13.5 Run Time

During DeckCheck command execution the system performs the following steps before displaying the images:

- Arm movement (if necessary, and if the command is not executed in the scope of a script loading guidance) to allow unobstructed camera views on the worktable.
- Image acquisition
- Evaluation and validation of the images
- Displaying of the images

On the touchscreen the following elements are then displayed:

- Reference image
- Live image

The live and reference images are alternating and detected potential discrepancies are marked with a red square. If there are three cameras on the system, the first camera from the left with any discrepancies will be displayed. If no discrepancies are found, the left camera will be displayed.

On a Fluent with 3 arms the middle arm will always block one of the DeckCheck cameras. This is visible when the DeckCheck live view is shown. During image acquisition two pictures are taken, which allows the middle arm to move out of the way. The two pictures are combined into the final image that is compared to the reference image.

The speed of the DeckCheck execution until image display depends on the PC hardware configuration.

Parameters

Camera View: To navigate between the cameras there is a button for each camera. Clicking on the respective button (left, center, right) will display the corresponding live and reference image of that camera.

Pause Alternate or Resume Alternate: Starts and stops the alternation between the reference and live images.

Check: Performs a re-validation of the live image. The DeckCheck command is executed again and runs through arm movement (if necessary), image acquisition, validation of pictures against reference pictures and display. This button can be used if the user needs feedback on whether all the discrepancies are resolved. An automatic re-validation is executed when closing the front door.

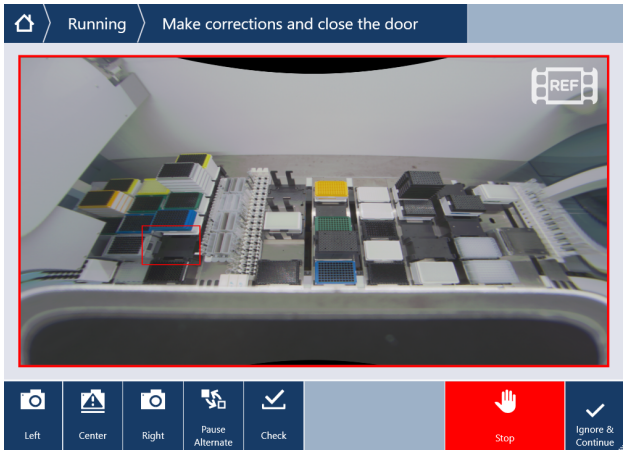
Stop: The run is stopped. The user has the possibility to retry the run or abort it.

Ignore & Continue: This option is only available in case any discrepancies are found and **Allow operator to ignore and continue** has been selected in the DeckCheck command. Pressing this button will log the decision and any discrepancies will be ignored.

Continue: If no discrepancies are found, this button appears to continue the run.

Layouts

Discrepancies have been found:



The area of error depicts the region(s) where discrepancies were found. The area of errors will be highlighted on the live and the reference images with a red square.

The cameras with any detected discrepancy are highlighted with an exclamation mark on the respective camera buttons. Furthermore, a red border indicates that this camera has encountered one or more discrepancies.

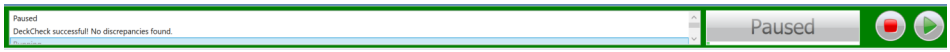
No discrepancies have been found:



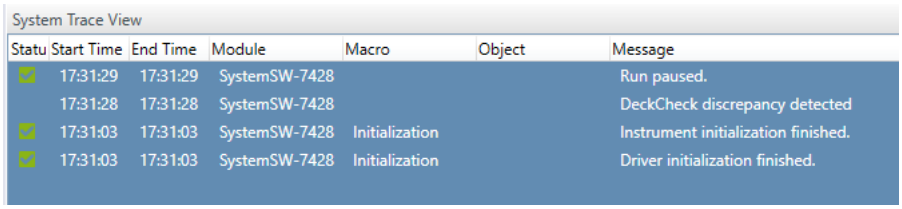
A green border indicates that no camera has encountered any discrepancy.

Notifications

Notification in Runtime Controller, if no discrepancies are found.



Notification in System Trace View if discrepancies are found.



LastDeckCheckResult Function

The function „LastDeckCheckResult()“ can be used to retrieve the result of the last DeckCheck script command. The return value is of type string.

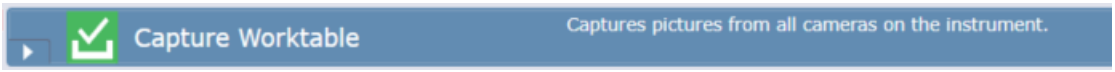
Tab. 12: DeckCheck Result

DeckCheck Result	Type	Description
“true”	string	DeckCheck did not find discrepancies
“false”	string	DeckCheck found discrepancies

11.13.6 Get notified with Introspect

Introspect sends the user a notification if any discrepancies have been detected. For more information about the use of Introspect, please refer to **https://www.tecan.com/knowledge-portal/digital-solutions**.

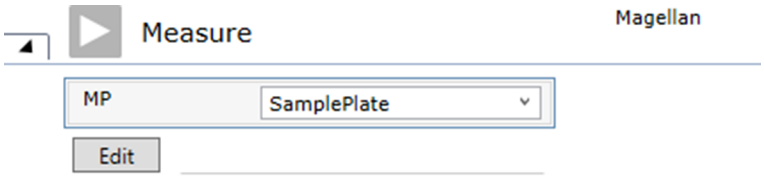
11.13.7 Capture Worktable



The ‘Capture Worktable’ command of the Deck Check driver captures pictures from all cameras on the instrument. The system performs an arm movement (if necessary) to allow unobstructed camera views on the worktable.
The captured worktable images are saved under ‘C:\ProgramData\Tecan\LoggingServer\LogFiles\DeckCheck Pictures’ and can be collected with the Snapshot Tool.

11.14 Magellan

11.14.1 Measure



The measure command of the Magellan driver can be used to execute a measurement with a Tecan Reader controlled by Magellan.

Parameters

MP: This field contains the name of the micro plate, that is to be measured by the Tecan reader. Either type your name or click in the Worktable editor window (Worktable Click).

Edit: Opens an editor to select the Magellan method to be executed for the measurement.

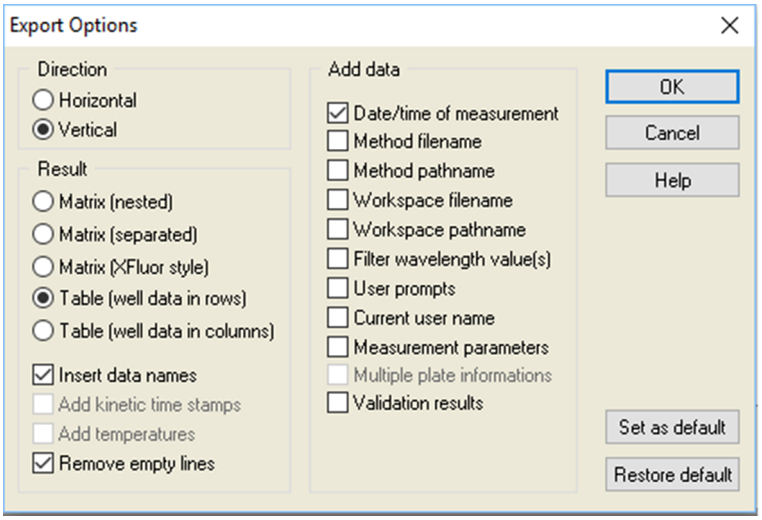
The following are effective whenever the Measure command is executed:

- A previously saved standard curve defined in the layout of the executing Magellan method is saved to a standard curve file and its graphical visualization to a bitmap file. Magellan uses the following target folder: `C:\Users\Public\Documents\Tecan\Magellan Pro\std`
- A previously saved standard curve file, that was previously saved, to apply gain and z-Position of the standard curve to the executing Magellan method. Magellan uses the following source folder: `C:\Users\Public\Documents\Tecan\Magellan Pro\std`
- English is used as the standard language for the measurement output file (.asc)
- If the ‘Use standard output format’ checkbox is unchecked, the output file format is as defined in the Magellan method
- If the ‘Use standard output format’ checkbox is checked, the following export settings are used as standard format for the measurement output file (.asc):

Export Options:

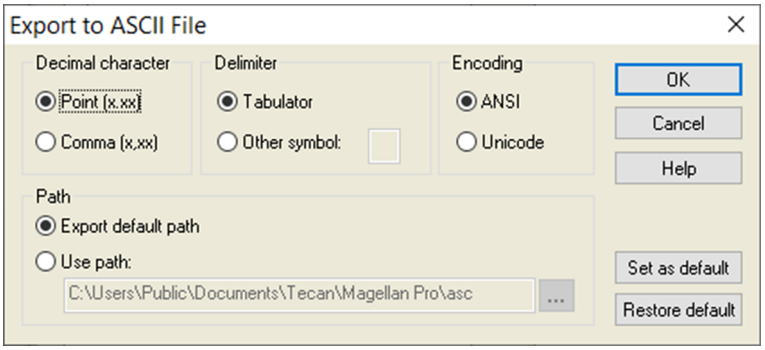
- Direction: Vertical

- Result: Matrix (separated)
- Add data: Date/Time of measurement
- Insert data names
- Remove empty lines



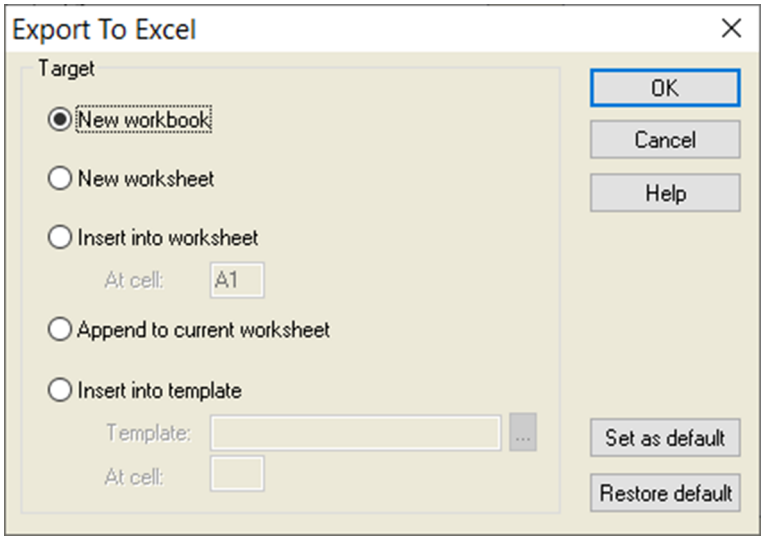
Export to ASCCI file

- Decimal character: Point (x.xx)
- Delimiter: Tabulator
- Encoding: ANSI
- Path: Export default path



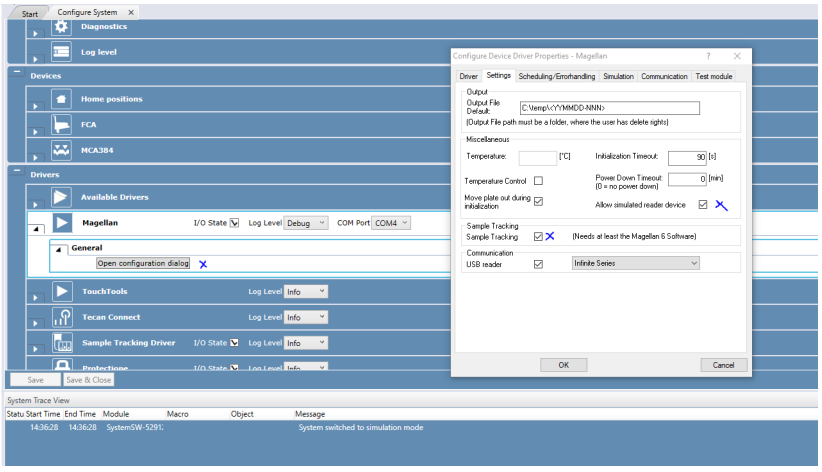
Export to Excel (requires that Excel is installed on the PC)

- New workbook



11.14.2 Driver Settings

Under Configure System > Magellan driver settings (‘open configuration dialog’) it’s important that Sample Tracking is enabled. If not, this is the main reason why Sample IDs will not be available for Magellan and later on Store Well Command. In addition, for running the Magellan driver in simulation the checkbox ‘allow simulated Reader device’ must be activated. Restart FluentControl after these changes.



11.15 Programming

11.15.1 If

With this programming command, you can branch the execution of your script depending on the **Condition** parameter.

If the specified Boolean condition is met (numerical result not equal to 0), the statements within the **If** group are executed. Otherwise, the execution continues in the following branch—either in the optional **Else** group, or after the end of the **If** group.

11.15.2 Comment

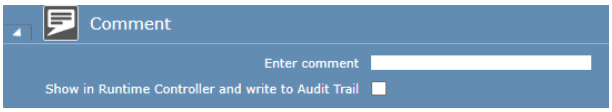


Fig. 306: Programming: Comment

This command is used to insert a comment for a script command or for some other reason. It is used to annotate the script and is not executable or visible to the operator. Use the “User Prompt” [369] command instead if you want to prompt the instrument operator. When you run the script, the comments are shown in the Runtime Controller of FluentControl. If you choose suitable comments, this can help the operator to monitor the progress of the script as it runs.

Settings

Comment: Enter a comment. If the comment will be used for the Audit Trail, check the check box.

11.15.3 Group

The Group command is used to group together script lines to make long scripts easier to read.

Description

Specify a descriptive name of your choice for the group.

11.15.4 Loop

With the Loop command, you group commands and set Begin Loop and End Loop automatically.

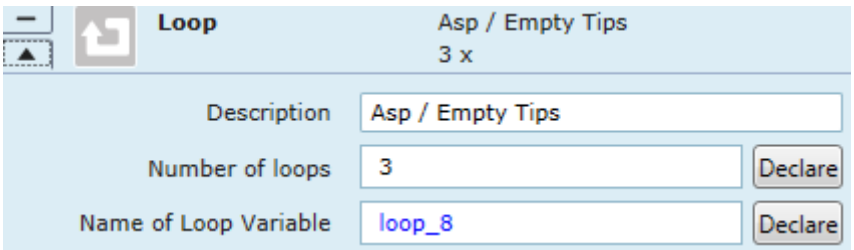


Fig. 307: Programming: Loop

Description

Name of Loop Variable: You can give the Loop a meaningful name or short description.

Number of loops: Specify the required number of times the loop should be repeated (range: 1–999).

11.15.5 Leave Loop / Group

With the Leave Loop command, you can leave the current loop. It skips all possible remaining cycles and continues with the statement behind it.

This programming command has no parameters.

11.15.6 Next Cycle

With the Next Cycle command, you can control the workflow of the script. Next Cycle leaves the current Loop cycle and jumps to the first command in the Loop. On leaving the last Loop cycle, it jumps to the first command after the Loop instead.

This programming command has no parameters.

If the command is outside of a loop in runtime the command will have no effect.

11.15.7 Else

The **Else** branch of a corresponding **If** statement.

If the condition specified in the complementary **If** statement is not met (numerical result equal to 0), the **Else** group of this If statement gets executed. Only one **Else** group can be added after an **If** group. Check that all **Else** branches are preceded by an **If** statement and that there is only one **Else** branch per **If** statement.

11.15.8 User Prompt

Script command to issue a message with requested user feedback.

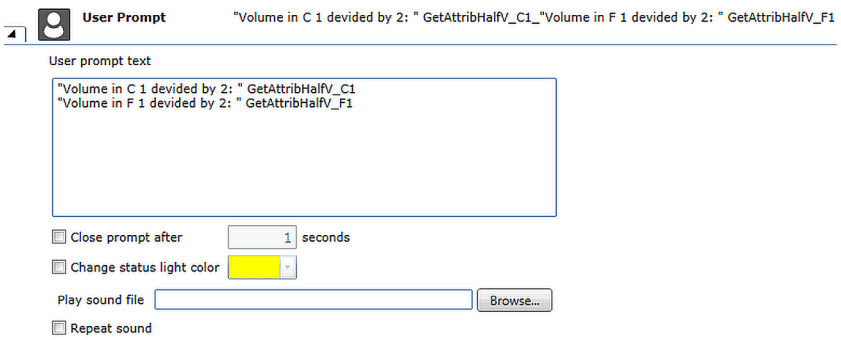


Fig. 308: Programming: User Prompt

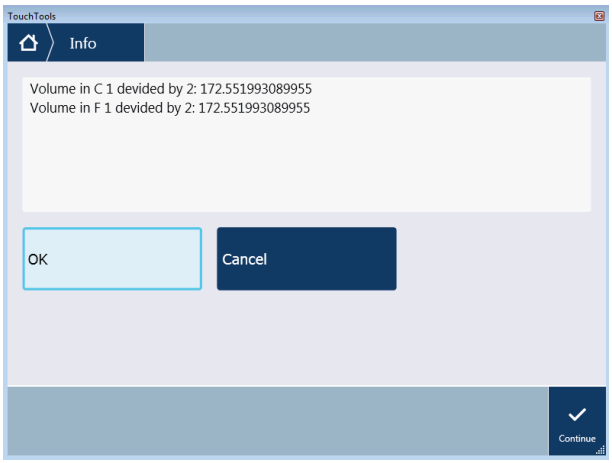


Fig. 309: Programming: User Prompt example

11.15.9 User Input

Use this command to request input from the operator for a variable value.

Name

Declare: Each variable which is used in a User Input command has to be declared first. Type the Name of your variable, and then click **Declare**.

Query Prompt: Type a question to the user.

Limit operator to allowed range: Specify maximum and minimum values for the numeric variable.

11.15.10 Set Variable

Use this command to set the value of a variable.

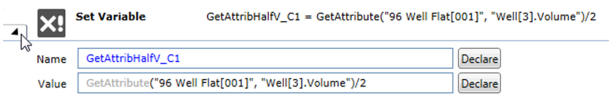


Fig. 310: Programming: GetAttribute Set Variable

The Declare buttons provide a convenient way to declare new variables. Please check the correctness of the declaration in the Variables section of the Script Editor afterwards.
You can use other variables or expressions in the Value field to calculate the new value for the variable. See “Functions and Operators” [▶ 442] Functions and Operators for a list of the available operators and functions.
Note that some functions can be used to manipulate the system database settings. For example, the SetAttribute function can be used to set the barcode information for a labware.

Always use double quotes ("") when assigning values to string variables.

For example, when a String variable is used to hold a barcode, the assigned value must use double quotes like "123456" because the barcode is a String, not a number.

Please note that if the non printing characters (ASCII characters from 0 to 31 except 9, 10 and 13) are used in the input fields there will be no context check error shown. The non printing characters will be ignored and they will not be saved.

11.15.11 Import Variable

The **Import Variable** command is used to import values for script variables from a file during script runtime. You can import values for variables which are already declared (defined) in the script. The import file can be either a comma-separated text file (TXT or CSV) or an Excel file (XLSX).

11.15.12 Export Variable

The **Export Variable** command is used to export the values of script variables to a file during script runtime. The export file can be either a comma-separated text file (TXT or CSV) or an Excel file (XLSX).

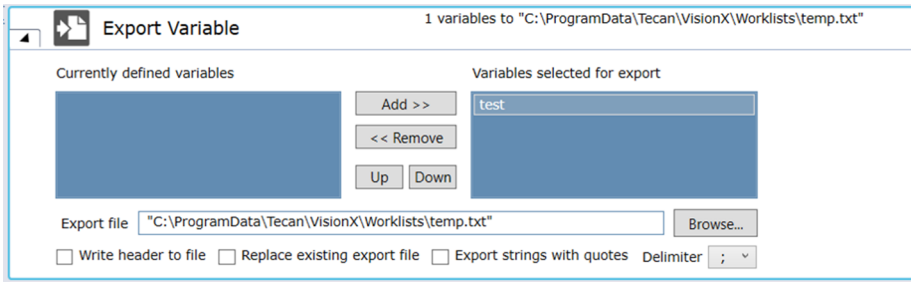


Fig. 311: Export Variable

The **Export strings with quotes** checkbox allows to choose whether to export variables of type string with enclosing quotation marks or without them.

Special characters such as °, µ as well as any superscript numbers will be converted to '?' characters when exported to TXT or CSV file formats.

11.15.13 Start Timer

The **Start Timer** command re-starts one of the 100 internal timers in FluentControl. If required, you can use this command to re-start a specified timer several times in the same script.



Fig. 312: Programming: Start Timer

Timer number

Specify the number of the timer you want to use or specify a numeric variable (range: 1 - 100).

11.15.14 Wait for Timer

Timer number: Specify the number of the timer you want to use or specify a numeric variable (range: 1–100).

Duration [sec]: Specify the time duration which has to expire since the selected timer was started or reset or specify a numeric variable (range: 0.02–86,400 s). The script will be paused until the specified time duration has been reached.

On active stop of the run the timer will not be paused or restarted. It will continue counting down.

11.15.15 Wait

Use this command to have the instrument **Wait** for a specific time duration.

Duration [sec]: Specify the time duration which has to expire since the selected timer was started or specify a numeric variable. With the Declare button you can declare a variable using the current parameter of "Duration".

Display Countdown

Indicates whether to display the countdown when executing the command. In process scripts, this option is deselected by default. On retry after an active stop, the command behaves differently depending on the used option. Selected: Countdown is restarted. Deselected: Wait command is skipped.

11.15.16 Subroutine

The **Subroutine** command executes another script as part of the current script.

Browse: Click this button to look for existing FluentControl scripts or modules and to choose the one you want to use as a subroutine.

Start Subroutine and wait for it to finish: This option causes the main FluentControl script to wait for the subroutine to finish before continuing.

Start Subroutine and continue main script: With this option, the main script continues without waiting for the subroutine to finish (asynchronous subroutine).

Wait for previously started Subroutine to finish: This option causes the main FluentControl script to wait for the specified, previously started subroutine to finish before continuing.

Transfer variable values to Subroutine: this list contains the default values of variables with scope Parameter in the Subroutine. The values can be modified and will then be used by the Subroutine. If the value of a variable from the main script shall be passed to the Subroutine, enter the variable name in this field instead. Values for variables of type String are wrapped in double quotes. Empty string values will lead to context check errors.
This option is not available if “Wait for previously started Subroutine to finish” (asynchronous subroutine) is selected.
Only variables with scope Parameter are shown in the “Transfer values to subroutine” section. It is not possible to remove variables with scope Parameter, however, additional variables may be added to the list to pass them to the Subroutine.

- **Value from main script**: this list contains the values of variables with scope Parameter in the Subroutine. If the value of a variable from the main script shall be passed to the Subroutine, enter the variable name in this field instead

- **Variable in subroutine**: this list contains all variables with scope Parameter used in the main script as well as the Subroutine

Transfer variable values from subroutine to main script: variables whose values shall be transferred from the Subroutine to the main script after the Subroutine has been executed can be manually entered here. This option is not available if “Start Subroutine and continue with main script” (asynchronous subroutine) is selected.

11.15.17 Start Application

Wait for termination and store return code in a variable.

11.15.18 Start VB

Use this command to execute a Visual Basic (VB) script from within a script.

FluentControl supports VB .NET instead of Visual Basic Scripting (VBScript) like EVOware. VB .NET offers many more options for interactions, but may be more complex. To get an idea of how to use this, take a look at the examples.

GetAndSetNumVars.vb

```
Imports Microsoft.VisualBasic
Imports System
Imports Tecan.Core.Scripting
Public Class TestClass
Implements IScriptObject
Private Host As IScriptingHost
Public Property ScriptingHost() As IScriptingHost Implements
IScriptObject.ScriptingHost
Get
Return Host
End Get
Set(ByVal value As IScriptingHost)
Host = value
End Set
End Property
Public Sub Execute() Implements IScriptObject.Execute
Dim factor1 As Double
Dim factor2 As Double
Dim result As Double
factor1 = Host.GetVariable("numValue1")
factor2 = Host.GetVariable("numValue2")
result = factor1 * factor2
Host.SetVariable("result", result)
End Sub
End Class
```

GetAndSetForce.vb

```
Imports Microsoft.VisualBasic
Imports System
Imports Tecan.Core.Scripting
Public Class TestClass
```

```
Implements IScriptObject
Private Host As IScriptingHost
Public Property ScriptingHost() As IScriptingHost Implements
IScriptObject.ScriptingHost
Get
Return Host
End Get
Set(ByVal value As IScriptingHost)
Host = value
End Set
End Property
Public Sub Execute() Implements IScriptObject.Execute
Dim force As Integer
force = Host.ResolveExpression("GetAttribute(""96 Well
Flat[001]"", ""Force"")")
force += 10
Dim expression As String = String.Format("SetAttribute(""96
Well Flat[001]"", ""Force"", {0})", force)
Host.ResolveExpression(expression)

End Sub
End Class
```

The VB.NET program has to implement a certain interface in order to be loaded and compiled by FluentControl. This is a small sample program:

```
Imports Microsoft.VisualBasic
Imports System
Imports Tecan.Core.Scripting

Public Class TestClass
     Implements IScriptObject

     Private Host As IScriptingHost

     Public Property ScriptingHost() As IScriptingHost
Implements IScriptObject.ScriptingHost
      Get
            Return Host
      End Get
      Set(ByVal value As IScriptingHost)
            Host = value
      End Set
     End Property

     Public Sub Execute() Implements IScriptObject.Execute
      // INSERT YOUR CODE HERE
     End Sub

End Class
```

11.15.19 End Script

The **End Script** command is used to stop the script prematurely and to specify a return code (error code) for the script. It is typically used together with error recovery commands.

11.15.20 On Error Goto

The **On Error Goto** command is used to jump to a specified line in the script (error handler) if an error occurs during script runtime. Error recovery commands then allow the script to react automatically to the error.

***On Error Goto** cannot be used to handle fatal errors, such as contouring errors.*

11.15.21 Error Handler

This special group statement represents an error handling routine that can be used by the **On Error Goto** statement.

11.15.22 Raise Error

With this programming command you can raise an error.

Parameters

Severity Level: Drop-down list box to define the severity level of the to be raised error. Range: ["Recoverable", "NonRecoverable"]. **Default value**: "Recoverable".

Error Message: Type the error message to be displayed.

11.15.23 Store Well Concentration

The Store Well Concentration Command is used to save the concentration data from a Magellan ASC File to a selected sample labware.

The quantification labware is used to calculate the dilution factor used during the script execution and apply this to the concentration values of the sample labware accordingly.

The concentration data from the sample labware will then also be sent to Sample Tracking during the execution of the command.

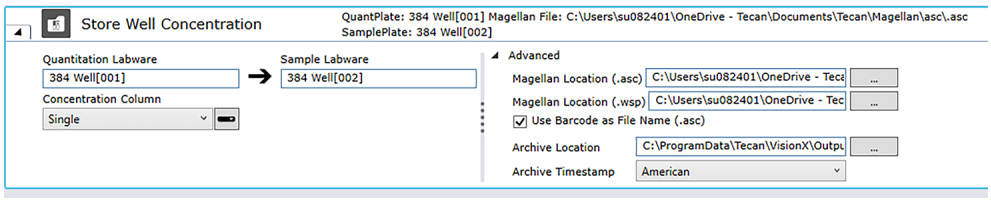


Fig. 313: Store Well Concentration

Parameters

Quantitation Labware: This field specifies the quantification labware used during the execution.

Sample Labware: This field specifies the labware containing the samples.

Magellan Location (.asc) (.wsp): These fields specify the path to the Magellan files used to execute the command.

The Checkbox **“Use Barcode as File Name”** can be checked, if the file name is the same as the barcode of the labware used.

Archive Location: This field specifies the file path of where the Magellan files will be archived after the command is executed.

Archive Timestamp: This drop-down list will change the format of the timestamp added to the archived Magellan files.

11.15.24 Confirm Standard Curve

The Confirm Standard Curve is used to confirm that a displayed standard curve matches the user expectations.

Edittime

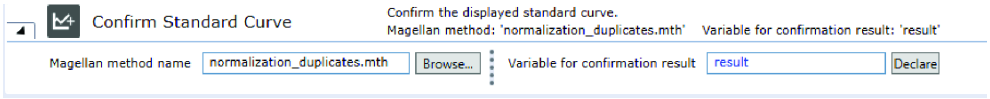


Fig. 314: Confirm Standard Curve

Parameters:

Magellan method name: The name of the Magellan method that has been used for the measurement of the standard curve. The user can select the Magellan method using the “Browse” button. This opens the default location for Magellan methods in the file system. The path to Magellan method in the file system is automatically added by the system.

Variable for confirmation result: An integer variable that is used to store the result of the confirmation dialog.

Runtime

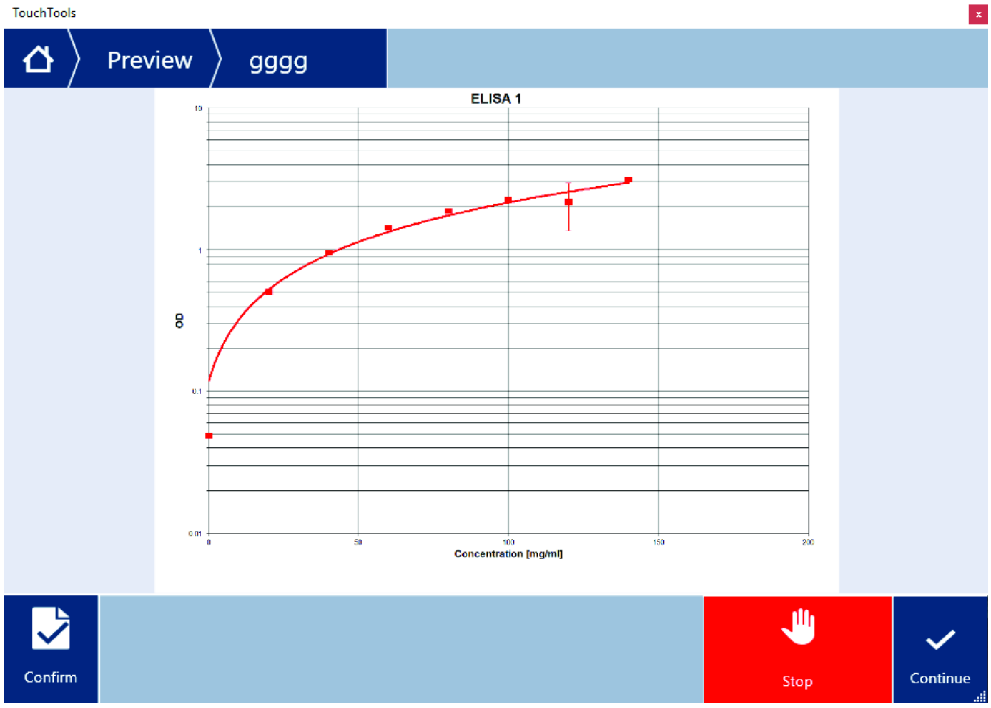


Fig. 315: Runtime

Buttons:

Confirm: Confirms that the displayed standard curve matches the expectations.

Stop: Stops the current run.

Continue: Continues the run.

Result

If the run is continued with confirmation, the value of the confirmation result variable will be set to 1.
If the run is continued without confirmation, the value of the confirmation result variable will be set to 0.

11.15.25 On System Prompt

The **On System Prompt** command allows to automatically respond to an error dialog if an error occurs during script runtime. The settings made in this command are valid until the current script ends or until they are overridden by a subsequent **On System Prompt** command.

The **On System Prompt** command does not allow handling any of the following dialogs:

- User Prompt
- User Input
- RUP

If a runtime error is encountered that matches the filter condition, the system automatically presses the specified button. The dialog is suppressed entirely—i.e., it is neither visible in the application nor on the touch screen.

***On System Prompt** cannot be used to handle fatal errors, such as contouring errors.*

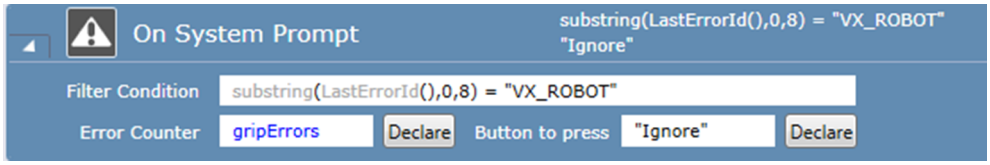


Fig. 316: On System Prompt

Filter Condition

The **Filter Condition** parameter specifies the conditions that must be satisfied for handling an error (e.g., filtered by their error ID). If this parameter is left empty, the specified handling applies to all runtime errors.

*It is possible to define multiple error handlings by using several **On System Prompt** commands with different **Filter Condition** parameters. If the **Filter Condition** parameters overlap, the most recent one in the script applies.*

Error Counter

The **Error Counter** parameter allows specifying a variable that is used to count the number of errors being handled. If this parameter is left empty, no counting will be performed.

Button to press

The **Button to press** parameter is required to be filled in. The button to be pressed can be specified either by its button text (case insensitive) or by its index—starting with 1.

11.15.26 Switch UV Light On



Fig. 317: Switch UV Light On

This command is only available if the UVC light option, which is used for decontamination, is installed.

To avoid shadow casting—and, hence, impaired decontamination—the instrument arms should be moved appropriately during the UVC radiation cycle by the according commands (e.g., “Move Arm (FCA)” [▷ 264], “Move Arm (MCA384)” [▷ 303], “Move Arm (RGA)” [▷ 282], “Subroutine” [▷ 372]). Labware or other objects that might impair the decontamination have to be removed from the worktable. Furthermore, UVC light can damage plastic and reagents.

Parameters

Duration: Define the irradiation duration in minutes (range: 1–30 minutes).

Each On/Off switch cycle reduces lifetime of the UVC light by 30–60 minutes.

*The command **Switch UV Light On** cannot be repeated within a 5-second interval. If this is attempted anyway, an error message is shown.*

When opening the front safety panel or the diluter panel, the UVC light is switched off automatically and an error message is shown.

The UVC light is switched off automatically after the defined irradiation duration—even if FluentControl has stopped running or the connection to the instrument failed.

11.15.27 Switch UV Light Off



Fig. 318: Switch UV Light Off

This command is only available if the UVC light option, which is used for decontamination, is installed.

This command switches the UVC light off.

*The use of the **Switch UV Light Off** command is optional. The UVC light is switched off automatically after the defined irradiation duration—even if FluentControl has stopped running or the connection to the instrument failed.*

11.15.28 Export Well Attributes

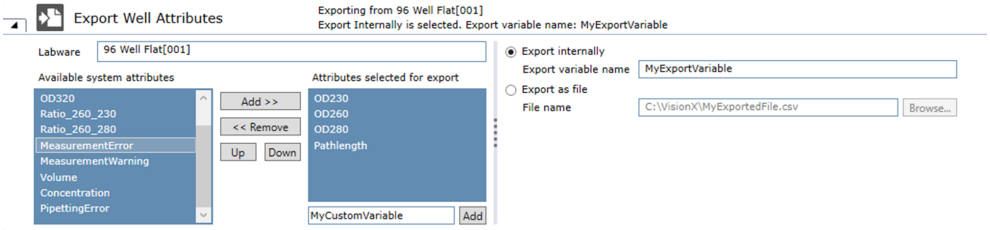


Fig. 319: Export Well Attributes

The **Export Well Attributes** command is used to export the well attribute values of a labware item during script runtime. The Export file is a (.csv) file, a semicolon (;) is used as a separator.

The header of the file contains the names of the exported well attributes. The file must contain a **Well Index** column. Each row below the header represents one well and its corresponding well attribute values.

Example

Well Index	OD230	OD260	OD280	PathLength
1	1.44	12.33	13.55	5.32
2	2.54	33.22	14.43	5.65

The raw text file of this example would then be:

```
Well Index;OD230;OD260;OD280;PathLength
1;1.44;12.33;13.55;5.32
2;2.54;33.22;14.43;5.65
```

Parameters

Labware: This will enable the user to select a labware from the worktable, from which the well attributes shall be exported.

Available System Attributes: This list contains all predefined well attributes that are registered in the system for export.

Attributes Selected for export: This list contains all the well attributes that shall be exported from the selected labware.

Custom Well Attribute Field: This field can be used to add custom attributes, which are not predefined in the system to the list of attributes selected for export. Clicking the **Add** button next to this field will add the custom well attribute to the list above.

Export Options: The export well attribute command contains two ways to export the selected well attributes.

- **Export Internally:** The generated export file is created by the system and is not meant to be edited by the user. This option should be used, if the user wishes to import the well attributes again during the same script runtime. Here, the user can define a unique export variable name which will register the exported well attributes to this variable. The user can then use the Import Well Attributes command (See Import well attributes for more details), enter the same export variable name, and import these attributes again to a different plate. Editing this file will cause the script execution to fail. If the user wishes to edit the export file, they should use the **Export as file** option.
- **Export as file:** The user can select a file using the **browse** button. This will then export the selected well attributes to the given location. This file will not be protected with a hash file and can be edited in between the export and import options. This option should also be used if the export and import commands are in separate script runtimes.

11.15.29 Import Well Attributes

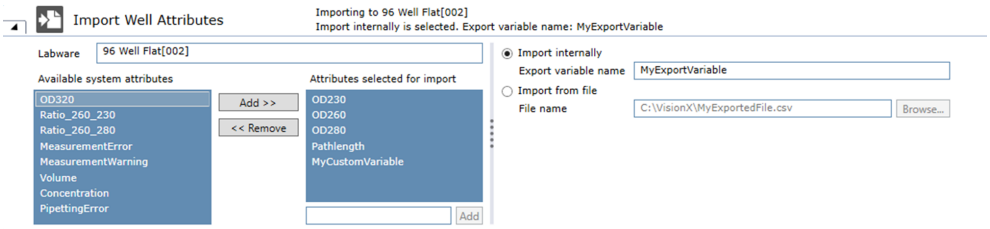


Fig. 320: Import Well Attributes

The **Import Well Attributes** command is used to import well attribute values to a labware item during script runtime. The well attributes from the import file will be written to their corresponding well index of on the labware. The Import/Export functionality supports different labware types. Wells that are not available on the import labware are skipped. The import file must be a (.csv) file, a semicolon (;) must be used as a separator.

The header of the file contains the names of the exported well attributes. The file must contain a **Well Index** column. Each row below the header represents one well and its corresponding well attribute values.

Example

Well Index	OD230	OD260	OD280	PathLength

1	1.44	12.33	13.55	5.32
2	2.54	33.22	14.43	5.65

The raw text file of this example would then be:

Well Index;OD230;OD260;OD280;PathLength
1;1.44;12.33;13.55;5.32
2;2.54;33.22;14.43;5.65

Parameters

Labware: This will enable the user to select a labware from the worktable, to which the well attributes shall be imported.

Available System Attributes: This list contains all predefined well attributes, which are registered in the system for import.

Attributes Selected for import: This list contains all the well attributes that shall be imported to the selected labware.

Custom Well Attribute Field: This field can be used to add custom attributes, which are not predefined in the system, to the list of attributes selected for import. Clicking the **Add** button next to this field will add the custom well attribute to the list above.

Import Options: The import well attribute command contains two ways to import the selected well attributes.

- **Import Internally:** The User can define a variable name that will import the well attributes exported in a previous Export Well Attributes command. For more information on this option, see the **Export Well Attributes** command.
- **Import from file:** The user can select a (.csv) file, which contains the well attributes they wish to import. The file must contain a header, which describes the well attribute names. For more information on this option, see the **Export Well Attributes** command.

11.15.30 Normalization Calculation

The **Normalization Calculation** command calculates how much sample and diluent volume is needed to reach a specific target concentration and volume. The command supports the user to generate a script for a normalization workflow. The command should be used together with the **Transfer Individual Volumes** command, which can read the output files of the **Normalization Calculation** command and transfer the calculated sample and diluent volume to a plate. For more information on this, see **Transfer Individual Volumes**.
The **Normalization Calculation** command will create three CSV files as output. One file contains the pipetting instructions for the sample labware. The second file contains the pipetting instructions for the diluent labware, and the last file contains a plate map confirmation, which can be used by the **Plate Map Confirmation** command to show which wells are going to be normalized.

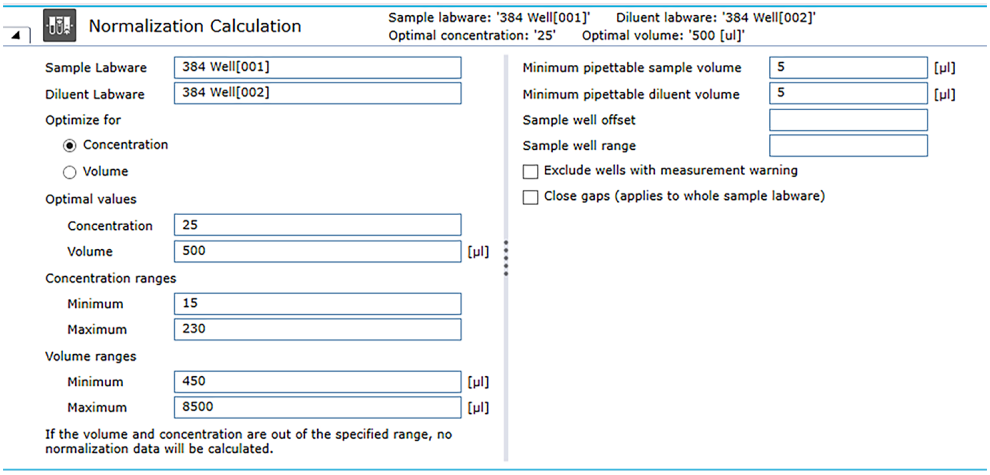


Fig. 321: Normalization Calculation

Parameters

Sample Labware: This field enables the user to select labware that contains the samples.

Diluent Labware: This field enables the user to select labware that contains the diluent.

Optimize for:

- **Concentration:** This selection enables an algorithm that optimizes the normalization according to the concentration. First, the algorithm tries to achieve both optimal values for concentration and for volume. In a second step, if optimal concentration and volume cannot be reached, the algorithm tries to achieve the optimal concentration, within the minimum and maximum volume ranges. If necessary in a third step, both volume and concentration are optimized in the allowed ranges, still prioritizing a result close to the optimal concentration. If no solution within allowed volume and concentration ranges is found the well will not be pipetted.
- **Volume:** This selection enables an algorithm that optimizes the normalization according to the volume. Similar to the previous optimization, both optimal values are tried to be met. In a second step, if optimal concentration and volume cannot be met, the algorithm tries to achieve the optimal volume within the minimum and maximum concentration ranges. If necessary in a third step, both volume and concentration are optimized in the allowed ranges, still prioritizing a result close to the optimal volume. If no solution within allowed volume and concentration ranges is found the well will not be pipetted.

Optimal Values: These fields contain the optimal values for concentration and volume. These are the main values, which the algorithm tries to achieve during normalization.

Concentration and Volume Ranges: These values define the range in which the algorithm can optimize the selected optimization setting. If the optimal values cannot be met, these ranges are taken into account, depending on the radio button selection.

Minimum pipettable sample and diluent volume: These fields define the minimum amount of sample or diluent which can be pipetted.

Exclude wells with measurement warnings: If a well contains a warning, which occurred during the measurement of the sample concentration, enabling this checkbox will exclude it from the normalization.

Sample well offset: Select by how many wells the starting point for the normalization calculation is shifted. This is usually expressed as a variable. No well offset will be applied, if this optional field is empty.

Sample well range: Select how many wells should be considered for the normalization calculation. This is usually expressed as a variable. Considering the shift by the well offset, all further wells of the sample labware will be considered for the normalization calculation, if this optional field is empty.

Close gaps: Enabling this checkbox removes all empty wells, which would occur during the normalization. This also alters the mapping between source and target labware. Close gaps applies to the whole sample labware and not only to a selected range, i.e. all wells up to the sample well offset are considered.

The command marks all wells, that have been considered for the normalization calculation, whether the normalization calculation has been successful or not.

11.15.31 In Situ Normalization Calculation

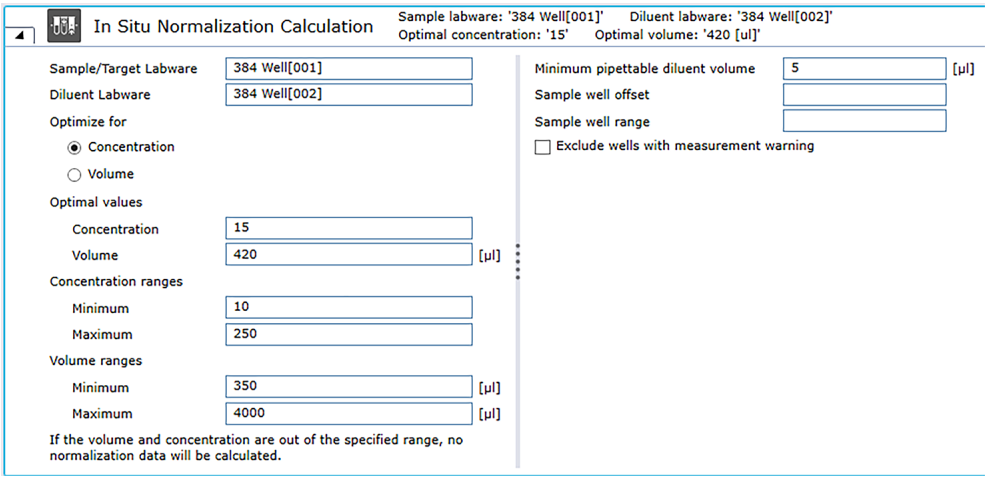


Fig. 322: In Situ Normalization Calculation

The **In Situ Normalization** calculation command performs a similar function to the Normalization Command. The difference between these commands is that the normalization is executed within the sample labware during an **In Situ Normalization Calculation**. This assumes that there is already a volume of sample in the target labware, since the sample and target labware are the same entity. Hence, it is not possible to add or remove a volume of sample from the wells of the source labware. It is only possible to add more diluent to the well.
For more information on fields and algorithm of the command, refer to “Normalization Calculation Command” [▷ 381].

Sample well offset: Select by how many wells the starting point for the normalization calculation is shifted. This is usually expressed as a variable. No well offset will be applied, if this optional field is empty.

Sample well range: Select how many wells should be considered for the normalization calculation. This is usually expressed as a variable. Considering the shift by the well offset, all further wells of the sample labware will be considered for the normalization calculation, if this optional field is empty.

The command marks all wells, that have been considered for the normalization calculation, whether the normalization calculation has been successful or not.

11.15.32 Module

To add a Module to a script, drag the Module from the Module section of the Controlbar into the Script Editor. Module commands are like synchronous Subroutine commands. To run a Module asynchronously, call it using a Subroutine command:

Transfer values to module: If the Module contains variables with the scope Parameter, they will be automatically inserted into the "Transfer values to module" section of the Module command after it is added to the script.
Only variables with scope Parameter are shown in the “Transfer values to module” section. It is not possible to add additional variables to the list to pass them to the Module.

- **Value from main script**: This list contains the default values of variables with scope Parameter in the Module. The values can be modified and will then be used by the Module. If the value of a variable from the main script shall be passed to the Module, enter the variable name in this field instead. Values for variables of type String are wrapped in double quotes. Empty string values will lead to context check errors.
- **Variable in subroutine**: This list contains all variables with scope Parameter used in the Module. These fields cannot be edited.
- **Transfer variable values from module to main script**: Variables whose values shall be transferred from the Module to the main script after the Module has been executed can be manually entered here.

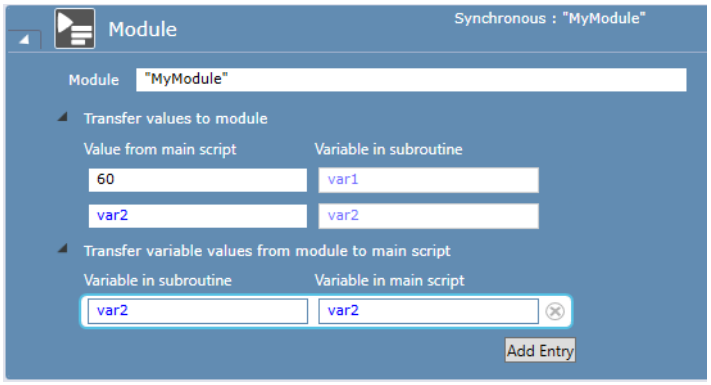


Fig. 323: Module command

11.15.33 Log

This command can be used to log additional information to the logfile. This can help to find relevant information later in the logfile.



Fig. 324: Log

11.15.34 Create Task

The Create Task command can be used to define tasks (Process iterations) to be executed by the system at a future time. One or more Create Task commands need to be followed by a Start Task command which applies the task definitions and stores them in the system's task repository.
To delay the execution of the process iterations, a date and time can be entered. The default of zero for both fields will start the new tasks as soon as possible.
After selecting a Process, the list of labware groups is filled. For each entry (which correspond to the Source Steps of the selected Process), you need to specify a labware name or labware barcode. When executing the Process iteration, this labware will then be assigned to the Source Steps.

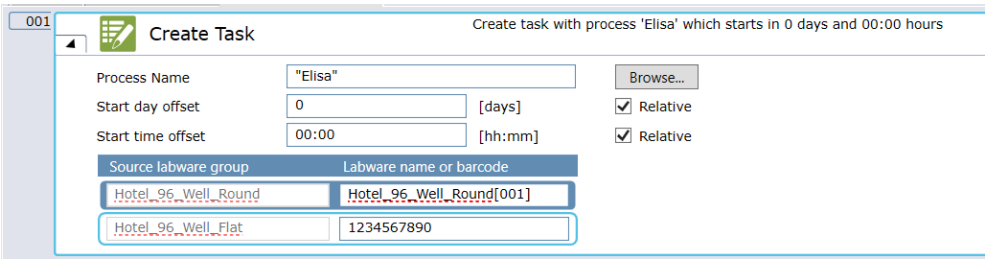


Fig. 325: Create Task

11.15.35 Start Tasks

Apply the tasks defined by preceding Create Task commands and store them in the system's task repository. See "Create Task" [▶ 385].

11.15.36 Register Event Handler

The Register Event Handler command can be used to react to variable value changes (It is also possible to react to variables of a FCA liquid class). When a value of a variable changes and a specified conditional expression becomes true an asynchronous subroutine will be started.

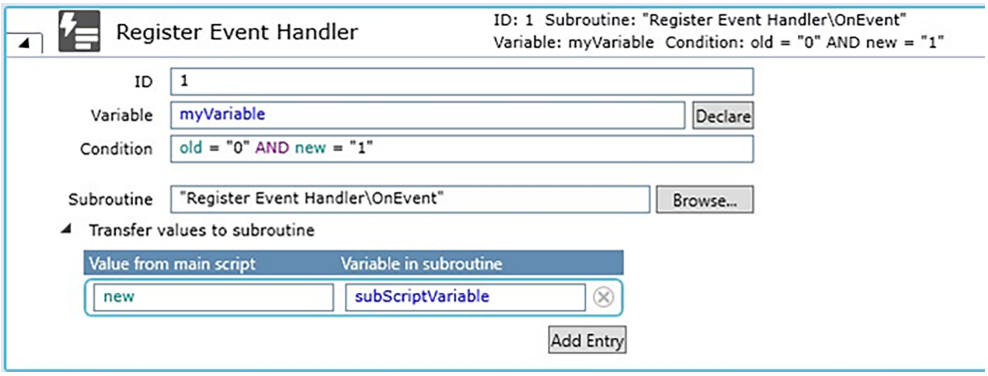


Fig. 326: Register Event Handler

ID: Specify a unique identifier to be able to unregister the event handler via the Unregister Event Handler command if needed.

Variable: The variable to react to. To react to variables of a liquid class the function MicroScriptVariable (see “Functions” [▶ 444]) needs to be used. Note: One Register Event Handler command per channel needs to be configured.

Condition: Specify the condition (expression) to may not react to any variable value change but only start the subroutine under certain conditions. The constants OLD and NEW can be used which represent the previous and the new value of the registered variable. No filter will be applied, and the subroutine will be started on any variable value change in case this optional field is empty.

Browse: Click this button to look for existing scripts or modules and to choose the one you want to use as a subroutine.

Transfer variable values to Subroutine: See “Subroutine” [▶ 372]. Note: Constants OLD and NEW can be passed to the subroutine.

11.15.37 Unregister Event Handler

The Unregister Event Handler command can be used to not react anymore to a variable registered by a Register Event Handler command.

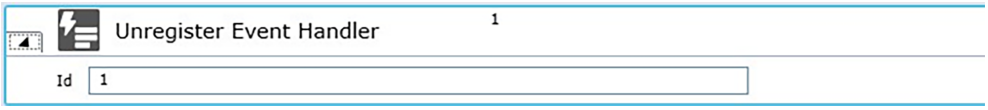


Fig. 327: Unregister Event Handler

ID: The identifier used by an existing Register Event Handler command which shall be unregistered.

11.16 Touch Tools

Rich user prompts provide information (e.g., images, sound, worktable, loading information) that is shown as a prompt in Touch Tools during script or process runtime.

Rich user prompts can be shown in two different modes:

- **Display and Wait:** The script is paused and waits for the operator to touch **Continue**.

- **Display and Continue:** The script continues and runs in the background.

The Rich User Prompt software components are automatically installed when you install TouchTools and are visible as a new group Rich User Prompt with new script commands in the Controlbar.

11.16.1 Wizard Group

This command can be used to display a group of loading instructions to the operator at runtime. While the commands in this group are executed, the doors can be open.

Supported commands are:

TouchTools:
- all commands

Worktable:
- Add Labware
- Remove Labware
- Set Location

Sample-Tracking:
- Register Labware

Programming:
- Import Variable
- Set Variable
- Start Application
- Start Vb
- Log
- Comment
- End Script
- Loop

: Loading ID
- Start loading command

Loop and if-else are not supported. Consider using TouchTools Wizard condition instead of 'if'.

If the wizard group contains only commands from the TouchTools group, it is possible to navigate forward and backward within the wizard.

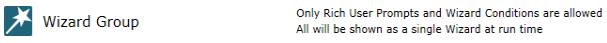


Fig. 328: Wizard Group

11.16.2 Standard (Rich User Prompt Display Command)

Use the Standard command to customize a new dialog to be shown on the touch monitor during script execution. Specify at least the Title and the Prompt Text. Press the Preview button to directly see your Rich User Prompt on the touch monitor.

Runtime behavior

During script execution, **Touch Tool** displays the **Rich User Prompt** template on the touch monitor user interface and displays the following:

- If parametrized, the picture is displayed.

- If parametrized, text is displayed. **Touch Tools** replaces a variable inside the text and XAML files by the corresponding variable value. Used functions are displayed using the total provided screen width and height.
- If parametrized, the sound is played with the configured amount of loops when the user prompt appears on the touch monitor.
- If parametrized, a progress bar is shown (optionally with percentage, remaining time, elapsed time, start time, and/or estimated end time value).

Parameters

The parameters of the Rich User Prompt (Display) command are as follows:

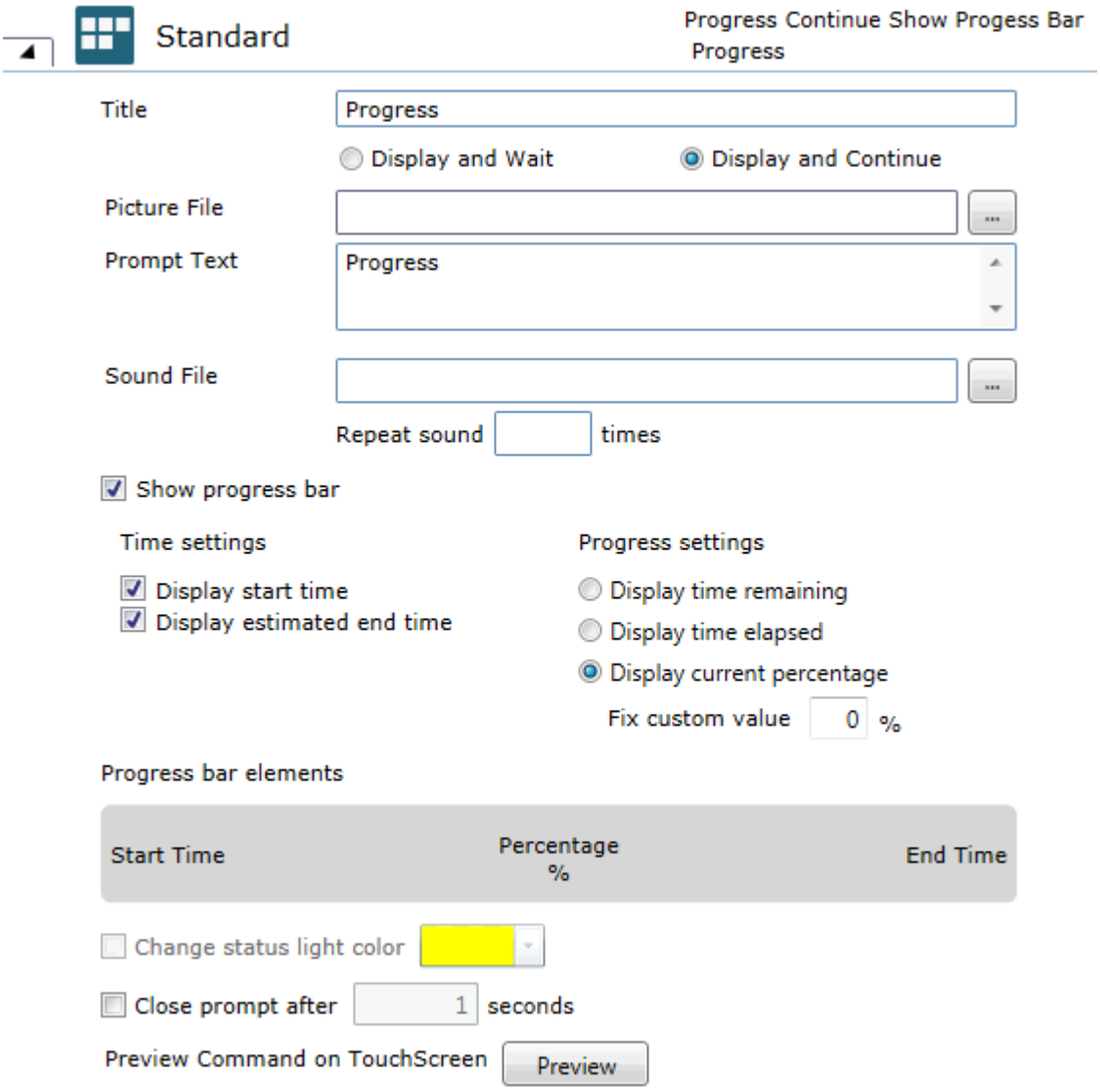


Fig. 329: Standard Rich User Prompt

Title: Specify a title of your choice. The title is shown above the Rich User Prompt during script runtime.

Select Rich User Prompt mode:

- **Display and Wait**: The script is paused and waits for the operator to touch Continue.
- **Display and Continue**: The script continues and runs in the background.

Picture File: Optional file path to a picture. You can choose a still or moving image. Supported formats are: XAML, video (.wav), jpeg, gif, bmp, png

Convention:

- XAML files must be of type UserControl.
- XAML files must not contain the class Attribute.
- You can place Script variable placeholders inside XAML. Placeholders have the following convention: ~VariableName~

Prompt Text: Type the text to be displayed on the touch monitor. Those parts of your typed input which should be treated as text have to be enclosed in quotation marks (double-quotes). All parts in between will be considered as expressions. If the expressions cannot be resolved, the entire comment text is treated as plain text. If the File Path parameter is empty and there is a Text parameter, the font size will be larger.

Sound File: Choose an optional sound file to play.

Repeat sound: Specify the number of times to repeat the sound (no entry = endless loop).

Show progress bar: Display an optional progress bar. When you choose the required progress bar options, a schematic view of the progress bar design is also shown under **Progress bar elements**.

Time settings: You can select the Display start time and Display estimated end time check boxes.

Progress settings: You can select one of the following options: **Display time remaining**, **Display time elapsed**, or **Display current percentage**. In case of the Display current percentageoption, you can type a value in the **Fix custom value** box in percent.

Progress bar elements: Automatic schematic view of the progress bar design reflecting your selected **Time settings** and Progress settings.

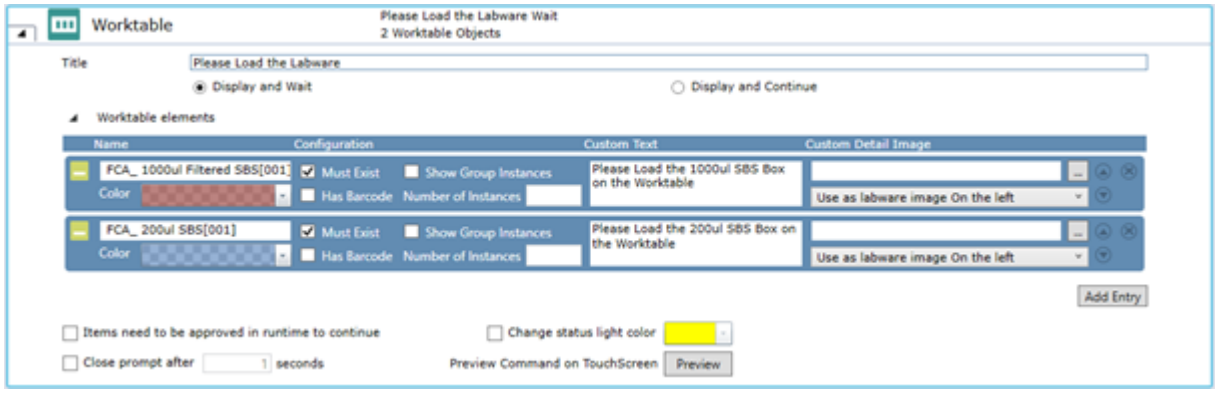
Change status light color: Select the color of the status lamp during the user prompt is shown from a color pallet.

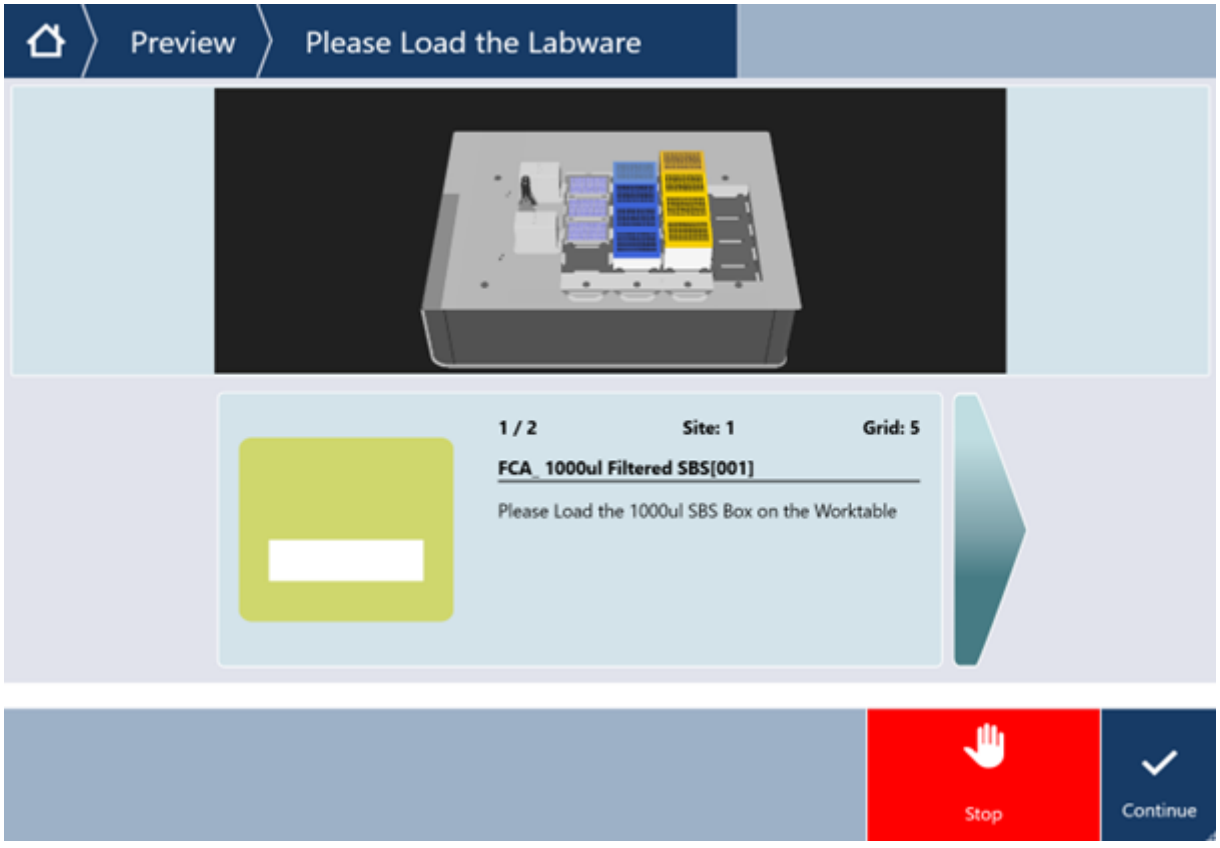
Close prompt after: To close the dialog automatically after a certain time, select the **Close prompt after** check box and then type the number of seconds in the box.

Preview: Click the **Preview** button to directly show your dialog on the touch monitor as it will appear at runtime. The colors you see in the preview may differ from the colors you see at runtime.

11.16.3 Worktable

The worktable command shows the operator a 3D visualization of the worktable with selectable carriers or labware blinking. A Text and Picture can be provided for each selected carrier to guide the operator through the worktable setup.





Name: Selection of the labware/carrier to display. The labware name must be in a non-array variable.

Color: A color picker which defines the color in which the currently shown carrier or labware item should blink in on the TouchTools interface.

Must Exist: If the checkbox **Must Exist** is activated and a specified labware is not present during runtime on the worktable, an error message will be displayed. The operator has the choice to abort or ignore the entire prompt and continue. If the checkbox is deactivated, the Worktable prompt will always be shown and only the existing labware will be displayed in the TouchTools interface.

Show Group Instances: If the **Show Group Instances** checkbox is checked, labware items out of the labware group will be shown in touch tools. The field **Number of Instances** defines how many instances of the labware group shall be shown. The Selected labware item in the **Name** field defines the first labware item of the group which shall be shown in the TouchTools interface. For example, if 10 labware items in the same group are present on the worktable, the operator selected the third labware item of the group, and defines **Number of Instances** with the value 5, the labware items 3 to 7 will be displayed in the TouchTools interface.

Has Barcode: If the Checkbox Has Barcode is checked, a field to enter a barcode will be shown in the TouchTools interface. If the option Show Group Instances is selected, a barcode field will be shown for all the displayed labware items in the TouchTools interface.

Custom Text: Custom Text which can be displayed in the TouchTools interface when the carrier or labware item is shown.

Custom Detail Image: An image which can be displayed alongside of the Custom Text, replace the Custom Text entirely, or access an URI to show an image.

Items need to be approved in runtime to continue: This check box will show an extra Approve button in the TouchTools interface. Each item shown in the Worktable command will have to be approved before the command execution can be completed.

Close Prompt after X seconds: This will close the TouchTools user prompt after the defined number of seconds.

Change status light color: This will change the instrument status light color to a defined status light when the TouchTools interface is shown.

Preview Command on TouchScreen: This will show a preview of the command on the TouchScreen without needing to execute the entire script or method.

11.16.4 XAML

Use the XAML command to customize a new dialog from a XAML file.

11.16.5 Report

Use the Report command to generate reports from template files. The template which is used to convert from XAML to PDF must not contain a scrollviewer.

When the option "Convert XAML to PDF" is selected, the resulting PDF file contains bitmaps (BMP) of the rendered content, not plain text.

11.16.6 Close

Use the **Close** script command to force Touch Tools closing an open rich user prompt (of type Display and Continue) automatically.

11.16.7 Wait for Close

Use the Wait for Close command if a rich user prompt is displayed in Display and Continue mode to stop the script until the rich user prompt is either closed by the user or closed automatically by the timer.

11.16.8 Wizard Condition

If the condition in the Wizard Condition command is true, all Rich User Prompts within the Wizard Condition Group are shown on the touch monitor at run time.

11.16.9 Get File

If the condition in the **Wizard Condition** command is true, all Rich User Prompts within the Wizard Condition Group are shown on the touch screen at run time.

11.16.9.1 Edit Time

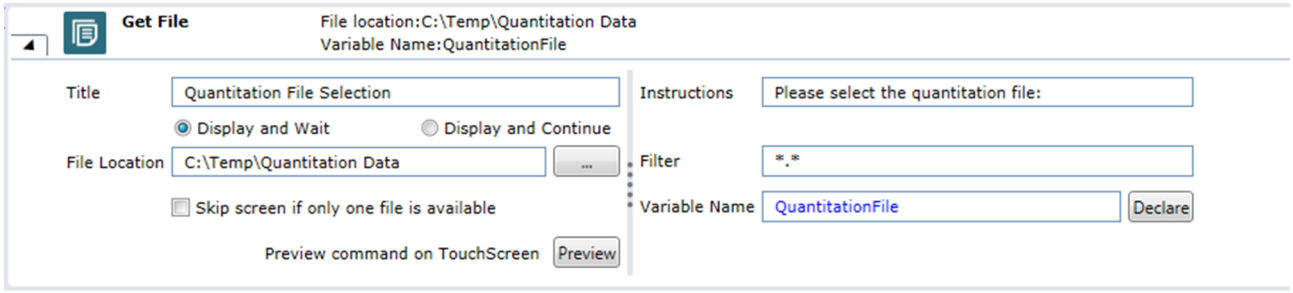


Fig. 330: Edit Time

Specify at least the **Title**, **File Location** and **Variable Name**.

Title: Specify a title of your choice. The title is shown above the Rich User Prompt during script runtime.

Display and Wait: The script is paused and waits for the operator to touch Continue.

Display and Continue: The script continues and runs in the background.

File Location: Specify a folder of your choice where the selectable files are located. Depending on the defined Filter the files inside the defined folder will be displayed during script runtime.

Skip screen if only one file is available: If this option is set, the runtime screen will not be displayed during script runtime if there is only one file available within the defined folder and for the given filter.

Instructions: Specify an instruction text of your choice. The instruction text is shown above the file list during script runtime.

Filter: Specify a filter of your choice. Only the files inside the file location that fit the filter value will be displayed during script runtime.

Tab. 13: Example filters

Filter	Description
*	* is a wild card that filters for any combination of characters and any length string
.	Returns all files in a directory

Filter	Description
*.txt	Only files with the .txt extension will be displayed
QuantitationData.txt	There must be a file with the exact name within the folder, otherwise no file will be displayed.
A*.*	Only files starting with a filename starting with an "A" will be displayed.
.t	Only files with an extension that has a "t" as the first letter will be displayed.

Variable Name: Name of the variable where the file path and name (selected at runtime) will be stored. If the variable is not yet declared it is possible to declare the variable via the declare button.

Preview: Click the **Preview** button to display the prompt on the touchscreen as it will appear at runtime. The colors you see in the preview may differ from the colors used at runtime.

11.16.9.2 Runtime

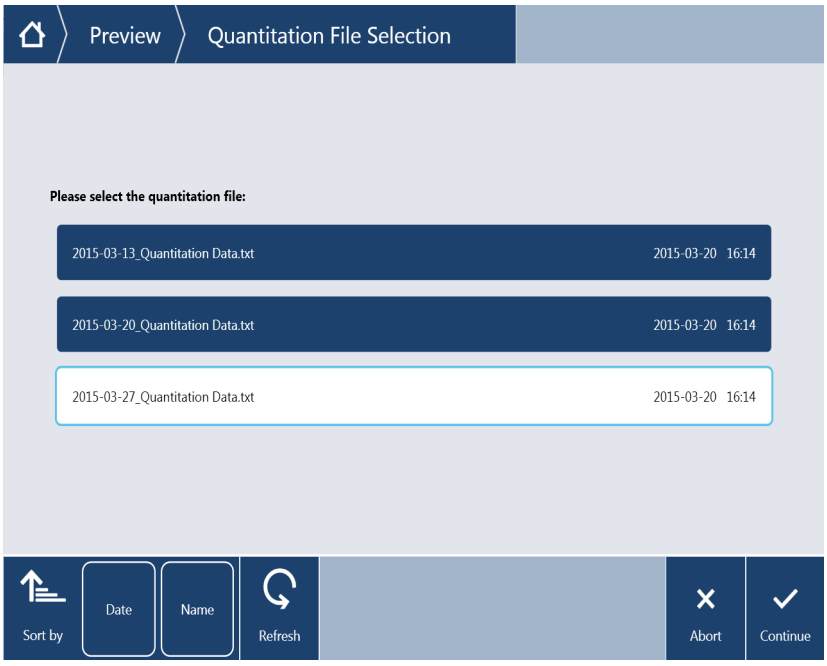


Fig. 331: Runtime

At runtime the touch screen displays a list of the files found inside the defined folder and filter.

Refresh: Will check the target directory again and update the selectable file names, if files that match the filter have been added or removed inside the defined folder.

Abort: Will abort the entire run after operator confirmation.

Continue: Continues with the script.

11.16.10 Variable

The Variable Rich User Prompt allows defining a runtime screen to change variable values during script run. It is possible to assign to each declared variable a control type (e.g., Textbox, Combobox) which will be displayed inside the runtime screen in a defined order. At runtime the operator is able to enter / change / select values which be stored in the associated variable upon pressing continue.

11.16.10.1 Edit Time

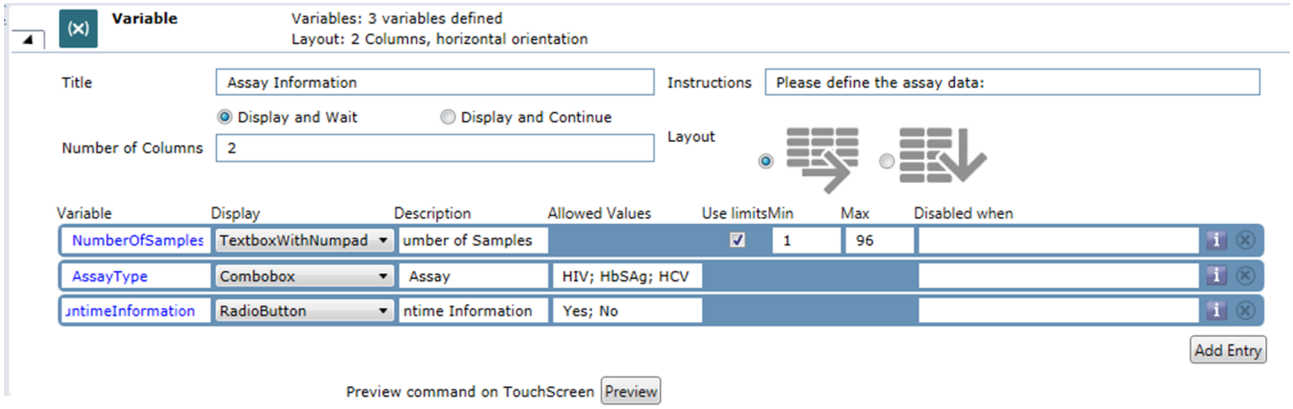


Fig. 332: Edit Time

Specify at least the **Title**, **Number of Columns**, **Layout**, **Variable**, **Display and Description**.

Title: Specify a title of your choice. The title is shown above the Rich User Prompt during script runtime.

Display and Wait: The script is paused and waits for the operator to press Continue.

Display and Continue: The script continues and runs in the background.

Instructions: Specify an instruction text of your choice. The instruction text is shown above the file list during script runtime.

Number of Columns: Defines the number of columns for the runtime screen to define its layout.

Layout: Defines the orientation of the controls within the runtime screen.

Variable: Specify the variable for which a runtime control shall be displayed.

Display: Specify the display runtime display type for the variable.

Tab. 14: Display Types

Display Type	Description
TextBox	Allows entering a text. When entering the textbox the TouchKeyboard will be opened automatically to be able to enter values via the Touch Screen.
TextBox, Numpad	Allows entering numbers. When entering the textbox the Numpad opens automatically to be able to enter values via the Touch Screen.
TextBox, No Keyboard	Allows entering text via the physical Keyboard or a Hand- held scanner. No Touch Support will be opened.
Checkbox	Allows enabling / disabling an option. Will store "True" or "False" to a string variable.
Dropdown	Allows selecting a pre-defined value from a list. Input the list
Radio Button	Allows switching between different options. Only one radio may be selected.
Empty	Placeholder to be able to adapt the layout of the screen. No control will be displayed at runtime at this position.

Description: Specify the header text for the control. For radio buttons this is the header for the full radio button group. For Checkboxes this is the label of the checkbox.

Allowed Values: Only visible for Dropdowns of Radio buttons.
Keep in mind that the number of allowed values for radio button values influences the screen layout.
Specify the values contained inside the dropdown as a semicolon separated list.
Specify the number of radio buttons and their labelone group via a semicolon separated list.
Example for both combobox: OptionA;OptionB;OptionC

Use limit: Only visible for Textboxes with Numpad. Specify if limits shall be checked for the field.

Min / Max: Minimum and maximum limits for the field that will be checked at runtime.

Disabled when: Specify an expression that defines when the field shall be disabled. The expression is resolved before the screen is displayed. A change of a screen value will have no effect on this function.

Add Entry: Adds a new row to the list. Remove rows by clicking the (X) button to the right of the row you would like to remove.

11.16.10.2 Runtime

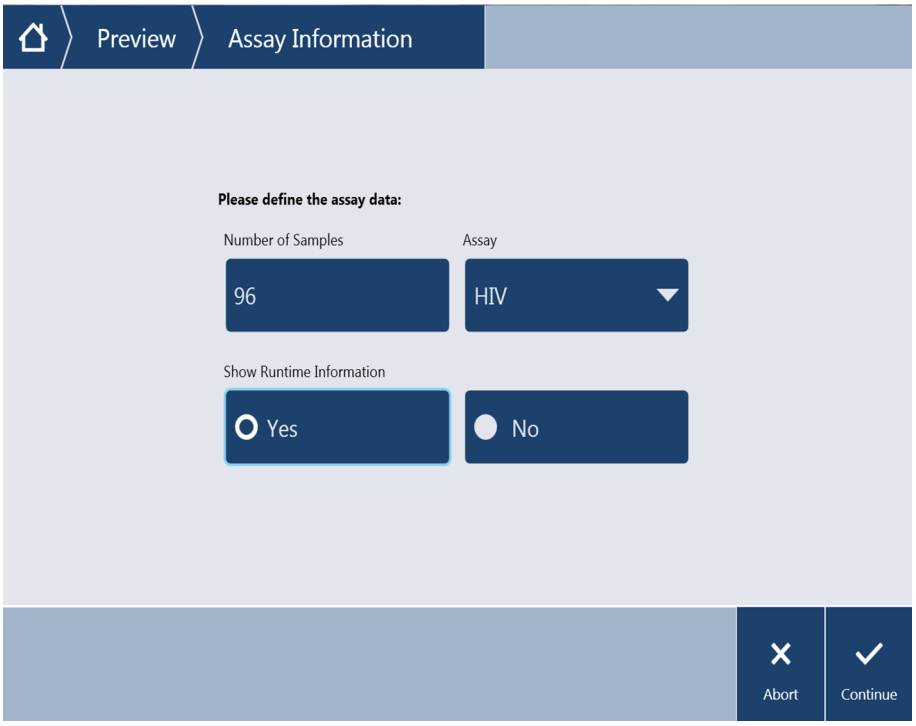


Fig. 333: Runtime

At runtime the touch screen displays the defined controls for the variables according to the defined layout.

The changed / updated / selected values are stored into variables when the operator presses continue.

Abort: Will abort the entire run after operator confirmation.

Continue: Continues with the script.

11.16.11 Confirmation

The Confirmation Rich User Prompt allows defining a runtime screen containing a custom image and text / XAML. This can be used to be displayed before instrument actions start and user confirmation is required, for example worktable loading. The screen can be customized to contain information that helps the user to be able to confirm the current status before continuing.

11.16.11.1 Edit Time

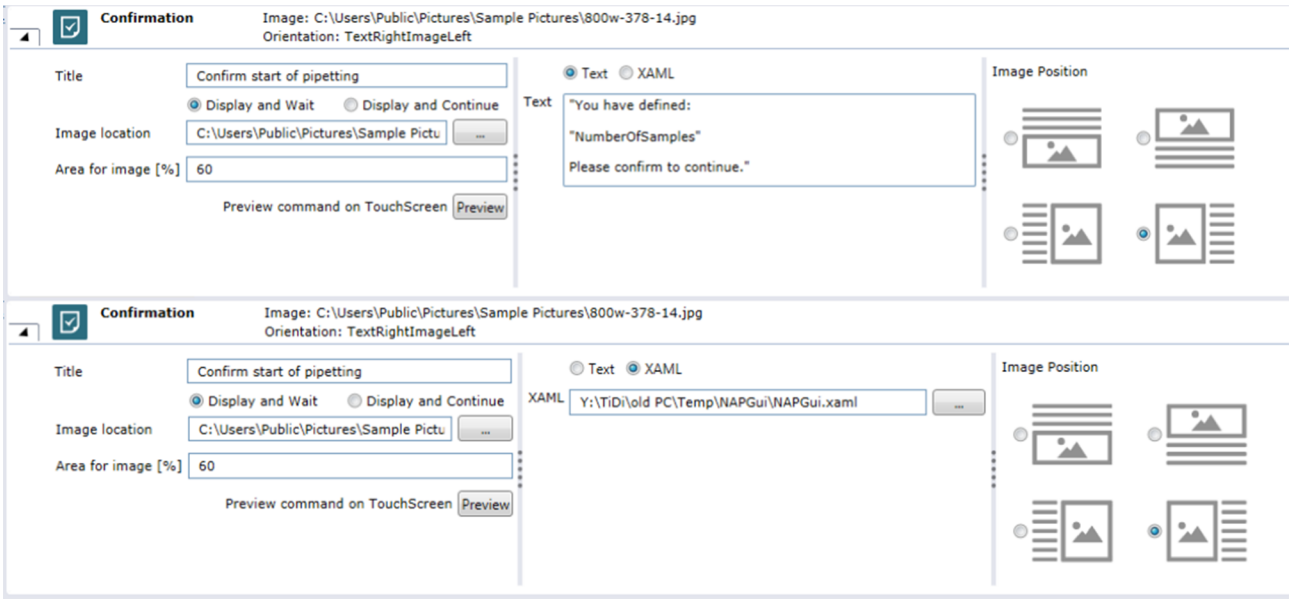


Fig. 334: Edit Time (Text Selected and XAML selected)

Specify at least the **Title**, **Image location**, **Area for image**, **Text** or **XAML**, and **Image Position**.

Title: Specify a title of your choice. The title is shown above the Rich User Prompt during script runtime.

Display and Wait: The script is paused and waits for the operator to touch Continue.

Display and Continue: The script continues and runs in the background.

Image location: Specify the path to the image that shall be displayed at runtime at the defined position.

Area for image: Specify the available space to display the image in percent compared the space used by the text or XAML content. If the area is zero, no image will be displayed.

Text: Specify a custom text that will be displayed at runtime at the defined position. The text allows capture returns and Variables. When using variables the non-variable text has to be within quotation marks.

XAML: XAML is an Extensible Application Markup language developed by Microsoft, allowing defining user interfaces. With XAML it is possible to provide formatted text and images. Specify the path to a valid XAML file that shall be displayed at runtime. This XAML file does not support the binding functions of the XAML Rich User Prompt and just supports standard XAML without Fluent specific bindings.

Example:

```
<UserControl xmlns=http://schemas.microsoft.com/winfx/2006/
xaml/presentation xmlns:x="http://schemas.microsoft.com/
winfx/2006/xaml">
     <Grid>
```

```
      <Grid.RowDefinitions>
            <RowDefinition />
            <RowDefinition />
      </Grid.RowDefinitions>
      <TextBlock Grid.Row="0" FontSize="18">
            You have loaded 96 samples
      </TextBlock>
      <TextBlock Grid.Row="1" FontWeight="Bold"
Margin="0,20,0,0">
            Please confirm to continue
      </TextBlock>
     </Grid>
</UserControl>
```

Image Position: Specify the position for the image inside the runtime screen.

11.16.11.2 Runtime



Fig. 335: Runtime

At runtime the touch screen displays defined image / text / xaml in the defined size and orientation.

Abort: Will abort the entire run after operator confirmation.

Confirm: Continues with the script and logs the decision.

11.16.12 Plate Map Confirmation

The Plate Map Confirmation Rich User Prompt allows defining a runtime screen containing a plate or tube with well information. This can be used to be displayed before instrument actions start and user confirmation is required. The screen can be customized to contain information that helps the user to be able to confirm the following actions.

11.16.12.1 Edit Time

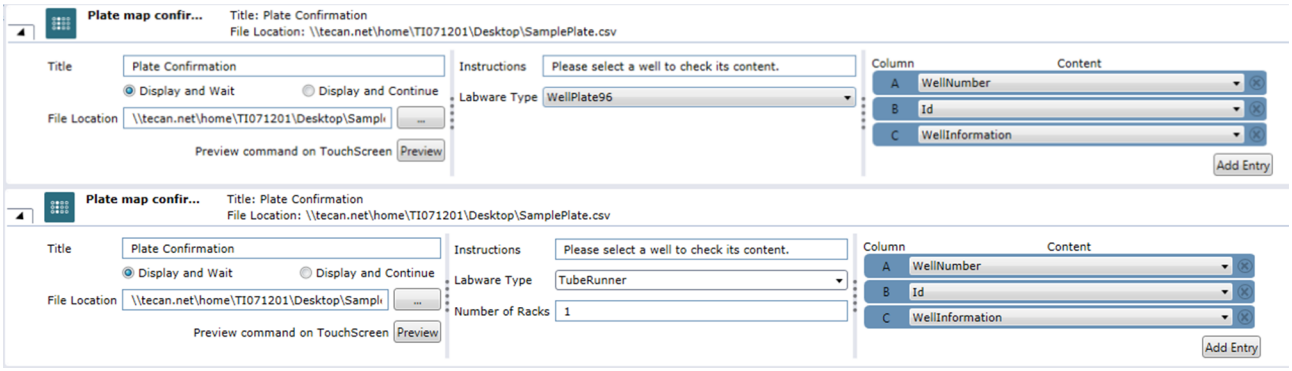


Fig. 336: Edit Time (Text Selected and XAML selected)

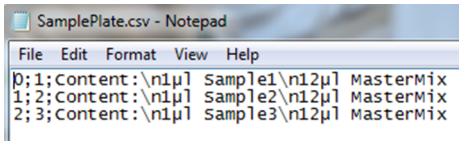
Specify at least the **Title**, **FileLocation**, **Labware Type**, **Well Number** and **WellInformation**.

Title: Specify a title of your choice. The title is shown above the Rich User Prompt during script runtime.

Display and Wait: The script is paused and waits for the operator to touch Continue.

Display and Continue: The script continues and runs in the background.

File location: Specify the path to the file containing the well information. The file must be a semicolon separated CSV file and must contain at least the well number and the well information—e.g.:



Instructions: Specify an instruction text of your choice. The instruction text is shown above the file list during script runtime.

Labware Type: Specify the type of labware that shall be displayed at runtime. Possible selections are:
- 96 Well Plate
- 384 Well Plate
- Tuberunner

Number of racks: Specify the number of tuberunners that shall be displayed at runtime.

Column association: Associate the columns of the file (e.g., A, B, C) to the identifiers.

Well number: 1-based number referring to the well on the plate. It is counted from top to bottom and then left to right.
A1 = 1, A2 = 9… , B1 = 2, B2 =10

Well information: text allowing text formatting via for example \n. The text is displayed at runtime for the selected well.

ID: text that will be displayed at runtime within the well for an easier well identification. This value is optional.

11.16.12.2 Runtime

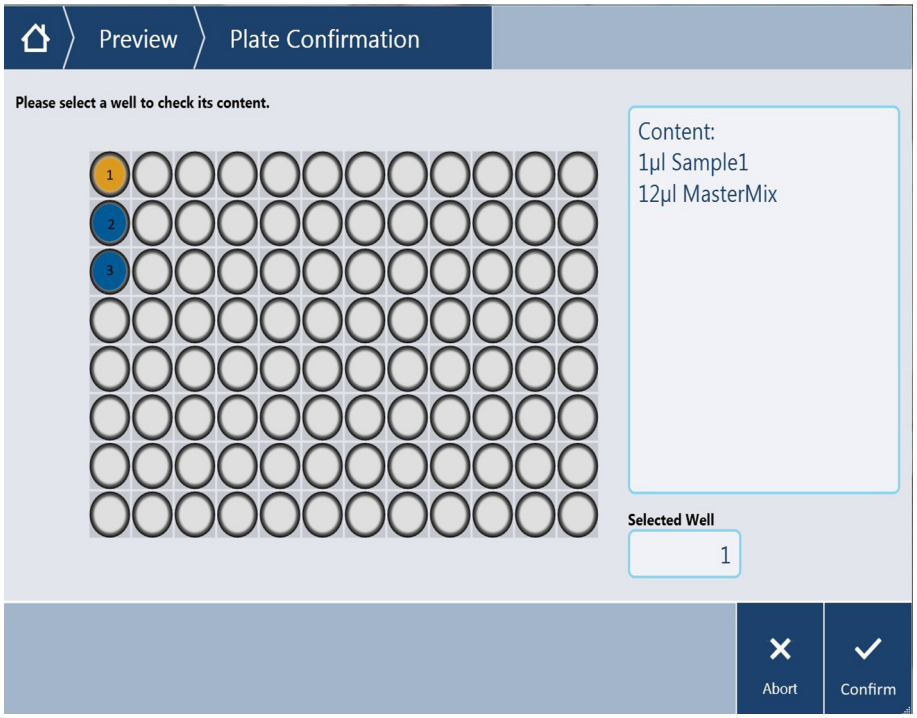


Fig. 337: Runtime 96-well plate

At runtime the touch screen displays the defined labware and well information. Empty wells are displayed gray.

Abort: Will abort the entire run after operator confirmation.

Confirm: Continues with the script and logs the decision.

11.16.13 Select Wells

The Select wells Rich User Prompt allows selecting the number of samples for the current run by defining the start and end well of a pre-defined labware.

There are different selection options that can be defined for the runtime screen.

On continue the selected values are stored into defined variables and can be used for the subsequent script.

11.16.13.1 Edit Time

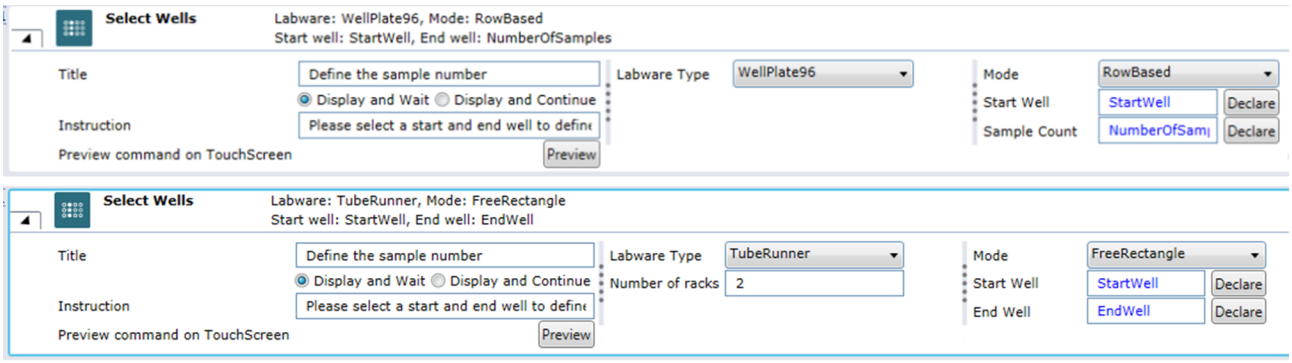


Fig. 338: Edit Time (Well Plate and Tube Runner selected and different modes.)

Specify at least the **Title**, **LabwareType**, **Number of racks**, **Mode**, and **Variables**.

Title: Specify a title of your choice. The title is shown above the Rich User Prompt during script runtime.

Display and Wait: The script is paused and waits for the operator to touch Continue.

Display and Continue: The script continues and runs in the background.

Instructions: Specify an instruction text of your choice. The instruction text is shown above the file list during script runtime.

Labware Type: Specify the type of labware that shall be displayed at runtime. Possible selections are:
- 96 Well Plate
- 384 Well Plate
- Tuberunner

Number of racks: Specify the number of tube runners that shall be displayed at runtime.

Mode: Specify the well selection mode of the runtime screen. For each mode at runtime you have to select the start and the end well. The wells well be selected according to the defined mode.

Rowbased: selection starts at the selected start well and continues row-wise until the selected end well.

Columnbased: selection starts at the selected start well and continues column-wise until the selected end well.

Free Rectangle: selection starts at the selected start well and continues to the end well and builds a rectangle.

Start Well: Specify the variable where the selected start well number will be stored (1 – based) after continuing at runtime.

End Well: Specify the variable where the selected end well number will be stored (1 – based) after continuing at runtime.

Sample Count: Specify the variable where the number of selected samples will be stored after continuing at runtime.

11.16.13.2 Runtime

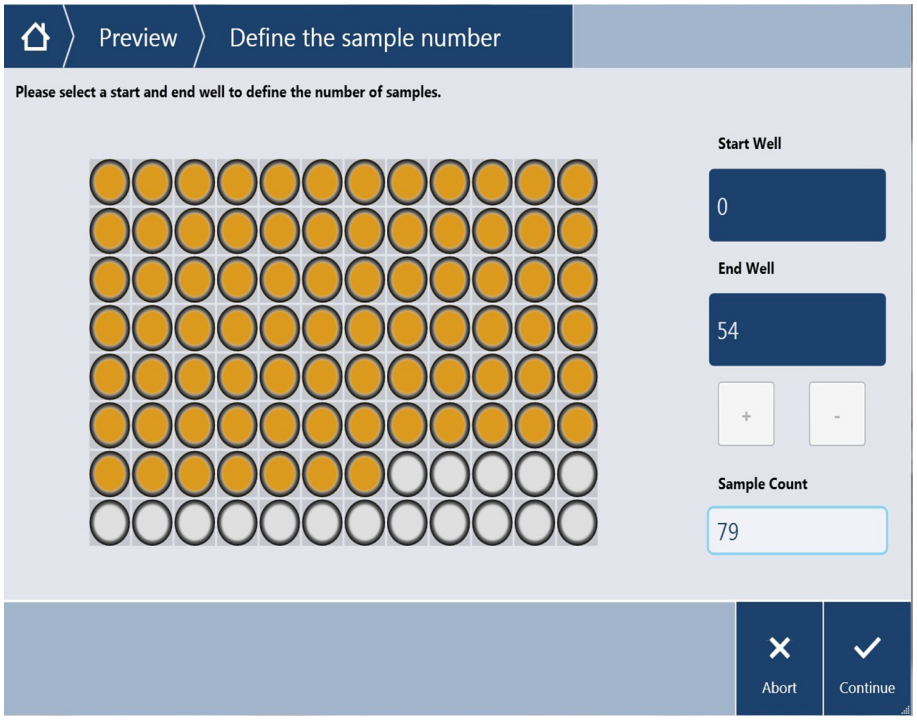


Fig. 339: Runtime

At runtime the touch screen displays the defined labware type and allows selecting the start and end well. Wells are selected automatically according to the defined mode.

Start Well: Specify the start well by either clicking on a well on the labware or using the + and – buttons. If no well is selected yet the first selection will be the start well.

End Well: Specify the end well by either clicking on a well on the labware or using the + and – buttons. If one well is already selected the next selection will be the end well.

Abort: Will abort the entire run after operator confirmation.

Confirm: Continues with the script and logs the decision.

11.16.14 Manually Scan Labware

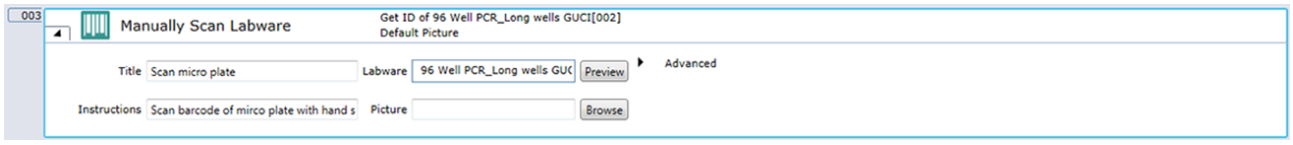


Fig. 340: Manually scan labware

Allows manually entering a barcode or identification to any labware during the run. It does not require a barcode scanner device on the deck. A barcode can be entered by handheld barcode scanner, touch monitor or key board. The software will prompt the user to manually enter a barcode for the specified labware. The command will then assign the entered barcode to the labware.

Title: Define a title for this command. The title is shown during the run on top of the touch monitor.

Instructions: Define instructions for the operator which is shown during the run on touch monitor.

Labware: Define the specific labware. This can be done by click on selected labware on the virtual worktable, entering the name or an expression.

Picture: If a picture is selected here it will be displayed during the run on touch monitor. If no picture is selected, the 3D worktable will be displayed.

Advanced

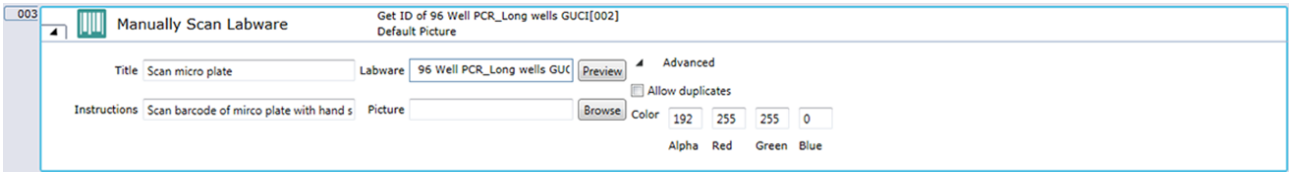


Fig. 341: Manually scan labware advanced

Duplicate IDs allowed: If this is checked duplicate IDs are allowed and operator will get no error message. If not checked, the operator will get an error message on the touch monitor about duplicate barcodes and their position.

Color: Define a custom color to highlight the labware in the default worktable graphic.

11.17 Worklist



Fig. 342: Worklist: Commands

NOTICE

Verify that the desired pipetting positions and results are achieved

When using worklist from either CSV or GWL files, please ensure the format is correct and verify that the desired pipetting positions and results are achieved.

To execute a worklist in a script, use the **Load Worklist** command to load the worklist file and specify additional parameters for the pipetting operations. Then insert an **Execute Worklist** command to run the worklist file. You can load several worklists by inserting several load worklist commands in your script. In this case, a single **Execute Worklist** command runs all of the worklists which you have loaded. They will be executed in chronological order of loading. FluentControl keeps a track of the worklists which have been run, and a second Execute Worklist command will not run them again unless you have reloaded them.

11.17.1 Worklist File Format

A worklist file is a text file containing pipetting instructions.

The worklist commands consist of individual lines (or records). Seven different record types are available: **Aspirate**, **Dispense**, **Wash Tips** / **Replace DiTis**, **Flush**, **Break**, **Set DiTi Type** and **Comment**. Each record starts with a single character indicating the type and is followed by one or more parameters. The parameters are separated by semicolons. Some of the parameters are optional and some record types have no parameters.

Aspirate: The Aspirate record has the following structure:

```
A;RackLabel;RackID;RackType;Position;TubeID;Volume;LiquidClas
s;Tip Type ;TipMask;ForcedRackType
```

Each Aspirate record specifies the aspiration parameters for a single tip (the next unused tip from the tip selection you have specified).

Dispense: The Dispense record has the following structure:

```
D;RackLabel;RackID;RackType;Position;TubeID;Volume;LiquidClas
s;Tip Type
;TipMask;ForcedRackType
```

Each Dispense record specifies the dispensing parameters for a single tip. It uses the same tip which was used by the preceding Aspirate record.

Tab. 15: Parameters for Aspirate and Dispense records in a worklist file

Parameter	Range	Explanation
`RackLabel`	Max. 32 characters	User-defined label (name) which is assigned to the labware.
`RackID`	Max. 32 characters	Labware barcode.
`RackType`	Max. 32 characters	Labware type (configuration name)—e.g.: “384 Well, landscape”.
`Position`	1 .. number of wells	Well position in the labware. The position starts with 1 and increases from rear to front and left to right.
`TubeID`	Max. 32 characters	Tube barcode.
`Volume`	0 .. +7158278	Pipetting volume in µl.
`LiquidClass`	Max. 32 characters	This optional parameter overwrites the liquid class specified in the Worklist command.
`TipType`		Reserved, must be omitted.
`TipMask`	1 .. 128	This optional parameter specifies the tip you want to use. The tip number is bit-coded (i.e., Tip 1 = 1, Tip 2 = 2, Tip 3 = 4, Tip 4 = 8, ...). Please note that you can only use a tip which has been enabled with Tip Selection in the Worklist command.
`ForcedRack- Type`	Max. 32 characters	This optional parameter is ignored by FluentControl.
`MinDetected- Volume`	0 .. +7158278	This optional parameter is ignored by FluentControl.

Wash Tip/Replace DiTi: The Wash Tip/Replace DiTi record has the following structure:

Tab. 16: Parameters for wash tip/replace DiTi

Parameter	Explanation
W;	To use the first wash scheme in the worklist command

The Wash Tip / Replace DiTi record washes the tip (or replaces the DiTi) which was used by the preceding Aspirate record. There are no parameters (the wash position, wash parameters and/or drop DiTis position are specified in the Worklist command).

Decontamination Wash: The Decontamination Wash record has the following structure:

Tab. 17: Parameter for decontamination wash

Parameter	Explanation
WD;	FluentControl converts the 'WD' command into an Aspirate (of the decontamination liquid specified in the Load Worklist command) followed by a wait and then a wash.

Decontamination wash consists of a decontamination wash operation followed by the normal wash procedure. There are no parameters (the parameters are specified in the Worklist command).

Flush: The Flush record has the following structure:

Tab. 18: Parameter for flushing

Parameter	Explanation
F;	It discards the contents of the tips without washing them or dropping DiTis.

Break: It has the following structure:

Tab. 19: Parameter for break

Parameter	Explanation
B;	The Break record forces the execution of previously specified aspirate, dispense or wash actions which have not yet been executed.

If you don’t specify a Break record, FluentControl normally executes pipetting commands in groups to optimize the efficiency. For example, if you have specified four tips in the Worklist command, FluentControl will queue Aspirate records until four of them are ready for execution. This allows pipetting to take place using all four tips at the same time. Specify the Break record if you want to execute all of the

currently queued commands without waiting. You can use the Break record—for example, to create a worklist that pipettes using only one tip at a time (even if you chose more than one tip in the tip selection).

Set DiTi Type: It has the following structure:

```
S;DiTi_Index
```

When using DiTis, S; is used to switch DiTi types from within a worklist. Choose the required DiTi type by specifying the DiTi index. FluentControl automatically assigns a unique index to each DiTi type. The DiTi index is shown in the **General Settings** tab of the DiTi labware.

If the S; record is not used, the worklist uses the DiTi type which you select with the Get DiTi button in the Worklist command.

The Set DiTi Type record can only be used at the very beginning of the worklist or directly after a Break record. A Break record always resets the DiTi type to the type selected in the Worklist command. Accordingly, if your worklist contains a Break record, you may need to specify the Set DiTi Type record again.

Comment: The Comment record has the following structure:

```
C;Comment
```

Start Timer: Start a timer. The Start Timer record has the following structure:
`TS;TimerNumber`

Wait for Timer: Wait for a defined number of seconds after starting the timer started with a previously executed TS; record. The Wait for Timer record has the following structure:

```
TW;TimerNumber;WaitTimeInSeconds
```

Start Timer and Wait for Timer records must not be used within a pipetting sequence using A; and D; records.

Reagent Distribution: The Reagent Distribution record has the following structure:

```
R;AspirateParameters;DispenseParameters;Volume;LiquidClass;No
OfDit iRe uses;NoOfMultiDisp;Direction[;ExcludeDestWell]*
```

where:

```
AspirateParameters =

SrcRackLabel;SrcRackID;SrcRackType;SrcPosStart;SrcPosEnd;
```

and:

```
DispenseParameters =

DestRackLabel;DestRackID;DestRackType;DestPosStart;DestPosEnd
;
```

Tab. 20: Aspirate Parameters for the Reagent Distribution command

Parameter	Range	Explanation
`SrcRackLabel`	Max. 32 characters	User-defined label (name) that is assigned to the source labware.
`SrcRackID`	Max. 32 characters	Source labware barcode.
`SrcRackType`	Max. 32 characters	Source labware type (configuration name)— e.g.: “384 Well, landscape”.
`SrcPosStart`	1 .. number of wells	First well to be used in the source labware.
`SrcPosEnd`	1 .. number of wells	Last well to be used in the source labware.

Tab. 21: Dispense Parameters for the Reagent Distribution command

Parameter	Range	Explanation
`DestRackLabel`	Max. 32 characters	User-defined label (name) that is assigned to the destination labware.
`DestRackID`	Max. 32 characters	Destination labware barcode.
`DestRackType`	Max. 32 characters	Destination labware type (configuration name)— e.g.: “384 Well, landscape”.
`DestPosStart`	1 .. number of wells	First well to be used in the destination labware.
`DestPosEnd`	1 .. number of wells	Last well to be used in the destination labware.

Tab. 22: Additional Parameters for the Reagent Distribution command

Parameter	Range	Explanation
`Volume`	0 .. +7158278	Dispense volume in the destination labware in µl.
`LiquidClass`	Max. 32 characters	This optional parameter overwrites the liquid class specified in the Worklist command.

Parameter	Range	Explanation
NoOfDitiReuses	-	Optional maximum number of DiTi reuses allowed (default 1 = no DiTi reuse).
NoOfMultiDisp	-	Optional maximum number of dispenses in a multidispense sequence (default 1 = no multi-dispense).
Direction	-	Optional pipetting direction (0 = left to right, 1 = right to left; default = 0).
[ExcludeDestWell]*		Optional list of wells in destination labware to be excluded from pipetting.

The Reagent Distribution record specifies the aspiration and dispensing parameters for all of the tips you have chosen (and dispenses the same volume in all of the chosen wells in the destination labware).

The Reagent Distribution record has the following advantages when compared to separate Aspirate and Dispense records:

- Pipetting is automatically optimized during script runtime for the number of tips which are actually fitted to the FCA.

Sample Transfer: The Sample Transfer record has the following structure:

```
T;AspirateParameters;DispenseParameters;Volume;LiquidClass;NoOfDiTiReuses;NoOfMultiDisp;SampleCount;ReplicationCount;SampleDirection;ReplicateDirection[;ExcludedestWell]*
```

where :

```
AspirateParameters = SrcRackLabel;SrcRackID;SrcRackType;SrcPosStart;SrcPosEnd;
```

and: `DispenseParameters = DestRackLabel;DestRackID;DestRackType;DestPosStart;DestPosEnd;`

and:

```
DispenseParameters = DestRackLabel;DestRackID;DestRackType;DestPosStart;DestPosEnd;
```

Worklist Examples Using Labware Labels

Use this worklist format if you want to use labware labels to specify source and/or destination positions in the worklist. This is necessary if your labware does not carry barcodes.

In this case, you only need to define the following parameters in an Aspirate or Dispense record:

```
A;RackLabel;;RackType;Position;;VolumeD;RackLabel;;RackType;P
ositi on;;Volume
```

Parameter RackType is optional. If specified, FluentControl checks whether the labware specified with RackLabel has the configuration name specified with RackType. An error message is displayed if an incorrect labware type is found on the Fluent worktable.

Parameter ForcedRackType is optional. If specified, FluentControl does not use the labware type of the labware specified with RackLabel, but uses ForcedRackType instead. The following worklist example illustrates the use of ForcedRackType in an Aspirate and a Dispense command:

```
A;S1;;;9;;10;;;;Microplate,
landscapeD;D1;;;1;;10;;;;Deepwell, landscape
```

landscape If you specify the ForcedRackType parameter, you must make sure that labware of the required types have been placed in the right positions on the instrument's worktable before the worklist is executed.

The system liquid is associated with a "virtual" labware item of type System and labware label Systemliquid. The following worklist command can be used to aspirate the system liquid:

```
A;Systemliquid;;System;1;;500:Water
```

The following worklist example aspirates 10.1 µl from well A2 in labware "S1", dispenses 6 µl into well A1 in labware "D1", dispenses 4 µl into well B1 in labware "D1" and then washes the tip / changes the DiTi:

```
A;S1;;Microplate, landscape;9;;10.1D;D1;;Microplate,
landscape;1;;6D;D1;;Microplate, landscape;2;;4W;
```

Note that FluentControl counts wells from rear to front and then from left to right, and a Microplate, landscape labware has eight wells in each column. Accordingly, well A2 is specified in the above worklist as "9".

11.17.2 Convert CSV to GWL (Worklist)

The **Convert CSV to GWL** command reads and parses a CSV file after parametrization and generates a corresponding GWL worklist file that can be loaded and executed.

The **Convert CSV to GWL** command is used to create a FluentControl worklist from an Excel file which is stored in CSV format.

Parameters

CSV file to import: Specify the file name and location of the CSV file to import. Enter the path and file name directly or click Browse and choose an existing file.

Start with line: Conversion normally starts with the first line of the import file. Specify a line number which is higher than 1 to skip one or more lines at the beginning of the file.

Stop with last line: Stop conversion with the last line of the import file.

Stop with line: Conversion normally stops with the last line of the import file. Specify the line number where you want the conversion to stop. The remaining lines in the import file are skipped.

Column Separator: Select from the list the separator which is used between the fields (columns) in each of the records (i.e. lines) of the import file. The default is **Comma**.

Generated Script (Subpath and Scriptname): Specify the file name and location of the worklist file (*.gwl) which should be created. Enter the path and file name directly or click **Browse** and choose an existing worklist file. The **Convert CSV to GWL** command overwrites the previous contents of the file each time it is executed.

Content: The **Convert CSV to GWL** command can handle up to 11 different input parameters in the import file (labware name, labware barcode, tube barcode, worktable position for both source and destination carrier; volume, liquid class, and tip). For each column (field) of the import file, specify the assignment to one of the 11 parameters by choosing from the drop-down list. Alternatively, leave the Content field empty to skip the field because it is empty or contains irrelevant data. An error message is output if you assign the same parameter to more than one column of the input file or make a meaningless or contradictory assignment.

11.17.3 Load Worklist

Use this command to parse a GWL worklist file for the selected FCA device and let FluentControl convert it into a subroutine script.

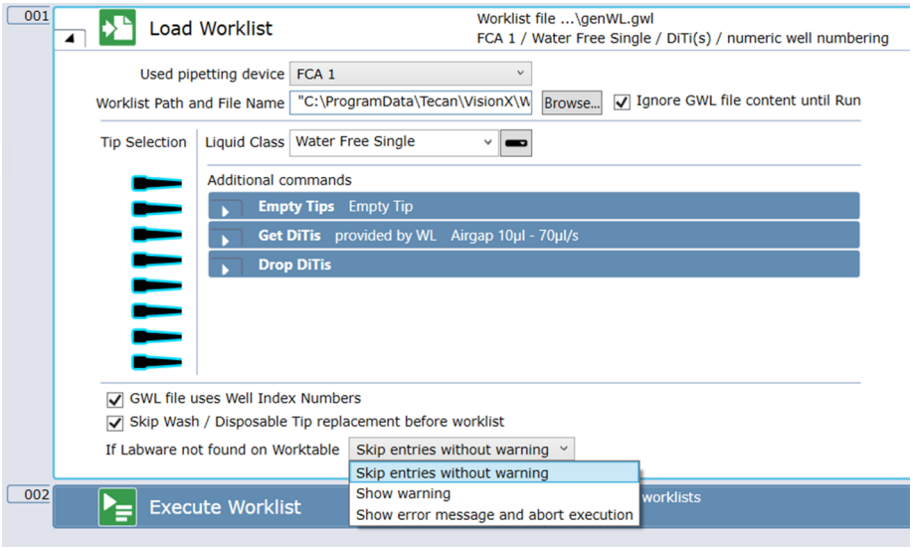


Fig. 343: Load Worklist

The **Load Worklist** command is used to load one or more worklist files, which can then be run with the **Execute Worklist** command. A worklist file is a text file containing pipetting instructions. It contains information on source and destination positions and the volumes to pipette. Worklists can be created using a plain text editor. They can also be created automatically by the **Convert CSV to GWL** command.

You can load several worklists by inserting several **Load Worklist** commands in your script. In this case, a single **Execute Worklist** command runs all the worklists which you have loaded. They will be executed in chronological order of loading. FluentControl keeps a track of the worklists which have been run, and a second **Execute Worklist** command will not run them again unless you have reloaded them.

The **If Labware not found on Worktable** drop-down list is used to specify how to react if the worklist contains labware barcodes or labware labels of labware that is not located on carriers accessible to pipetting arm (as determined by the checkbox accessible to pipetting arm of the carrier).

- **Skip entries without warning**: skips any affected transfers from worklist.
- **Show warning** skips an affected worklist and shows the details of the skipped transfers in the trace view as warning. Run will finish with warnings and if electronic signatures are enabled the operator has to confirm these warnings at the end of the run.
- **Show error message** interrupts execution of worklist and shows an error message to the operator. The operator can choose to ignore and continue (run will finish with errors) or to abort execution of the script.

Additional Parameters

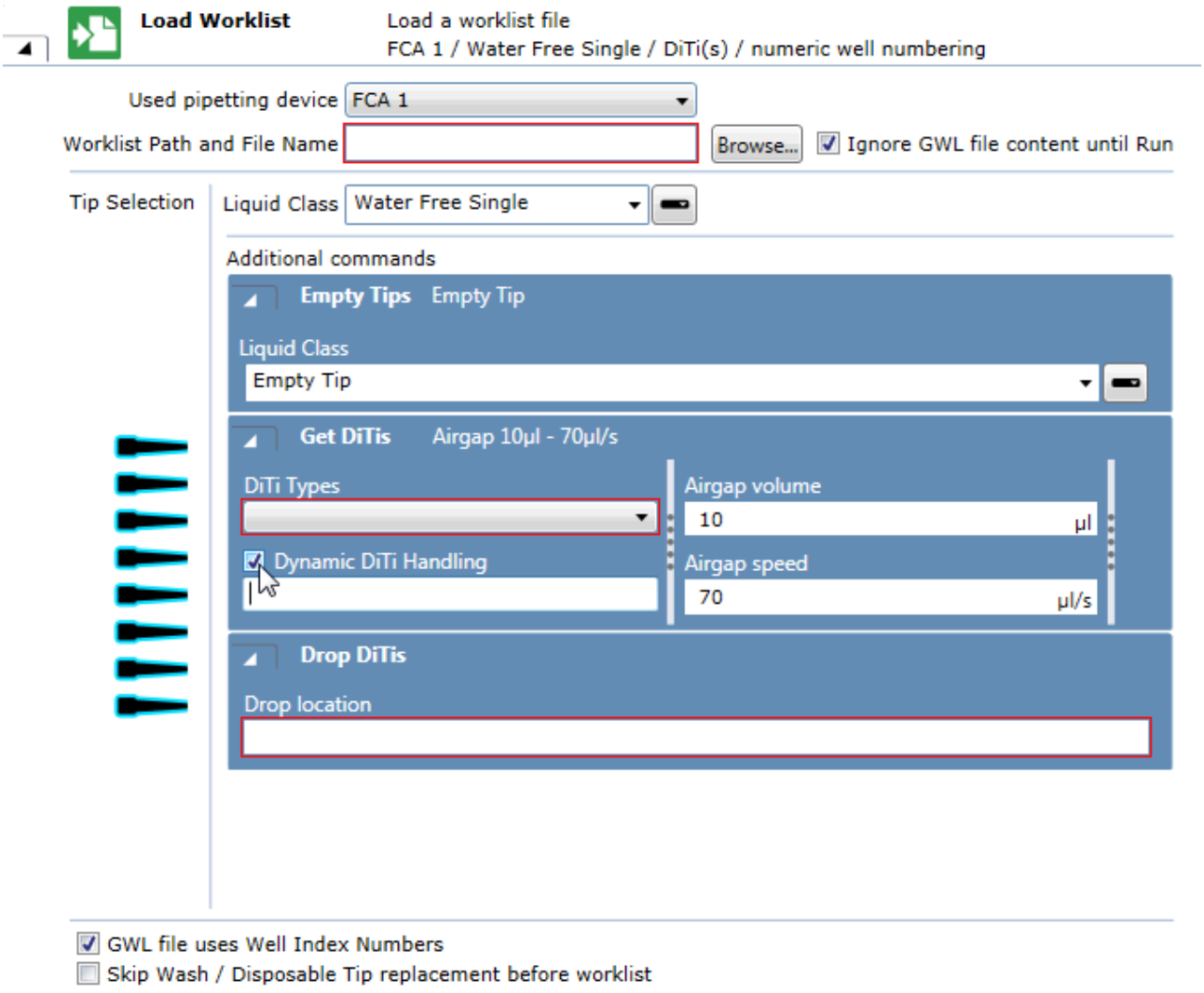


Fig. 344: Load Worklist: Additional Commands

Dynamic DiTi Handling: The dynamic DiTi handling is designed to use different DiTi types depending on the Aspirate volume. If you enter in the Dynamic DiTi Handling box—e.g.:

30:FCA DiTi 50µl;120:FCA DiTi 200µl;

this means that:

- DiTi 50 µl are used for volumes up to 30 µl,
- DiTi 200 are used for volumes > 30 to 120 µl, and
- for larger volumes the selected DiTi type from the Load Worklist command is used.

In the generated script, this leads – depending on which aspirates records got merged by FluentControl – to several **Get Tips** commands. When reusing tips, the tip type corresponding to the largest of the individual A;D; transfers preceding the next W; record is used.

Parameters

Used Pipetting Device: This parameter selects the FCA which is used to execute the worklist.

Worklist Path and File Name: Specify the file name and location of the worklist file (*.gwl). Enter the path and file name directly or click **Browse** and choose an existing file. This parameter accepts the name of a GWL text file in the form of an expression (this means, either a fixed value or script variables can be used to determine the real name of the file).

Ignore GWL file content until Run: Use this check box to indicate whether the GWL file contents shall be examined as part of a context check of the Load Worklist command or not (this is useful in such cases, when the GWL file is not dynamically created / modified during execution of the script). The default state for a new “Load Worklist” command for this check box is selected.

Tip Selection: Select the tips you want to use. If the worklist file specifies the tip selection (with the TipMask parameter) it overrides the settings you make here. If the commands in the corresponding worklist specifies a dedicated tip (with the TipMask parameter), this tip must be part of the tip mask set in the command, otherwise the system will report an error during parsing of the GWL file. It is not be possible to use a mixed tip configuration (fixed tips and DITIs) in the same worklist. Therefore the system does not allow choosing fixed tips if DiTi tips have been already selected and vice versa.

Liquid Class: Select a liquid class from the list of currently known liquid classes to be used as default (i.e., if not explicitly specified in the GWL file) or use a variable. You can choose a different liquid class for each source labware and for the destination labware.

GWL file uses Well Index Numbers: Uncheck this box if the GWL file contains alphanumeric well numbers (A1, A2, …, H11, H12) instead of numeric well numbers (1, 2, …, 95, 96). The box is checked and numeric well numbers are expected by default.

Skip Wash / Disposable Tip replacement before worklist: By default, the execution of a worklist begins with a Wash (in case of fixed tips) or a DiTi replacement (Drop Tips followed by Get Tips, in case of DiTis). You can explicitly skip the default behavior of wash or DiTi replacement at the beginning of the worklist by selecting this check box. The check box is cleared by default.

11.17.4 Advanced Worklist

With an advanced worklist you can add any script command to your worklist. To do so, just add a Break command (**B;**) to your .gwl file followed by the deserialized script command that shall be executed.

Per Break command only one script statement may be specified. And the command must be encapsulated in a **ScriptGroup element:**

```
B;<ScriptGroup><Objects><Object Type="Command Type">Command
Serialization</Object></Objects><Name></
Name><IsBreakpoint>False</
IsBreakpoint><IsDisabledForExecution>False</
IsDisabledForExecution><LineNumber>0</LineNumber></
ScriptGroup>
```

Command Type and Command Serialization are specific to the command. The parameterization of the command is part of the command serialization.

Here is an example for a Transfer Labware command for RGA1 which moves labware “SourcePlate[001]” to **Nest7mm_Pos**:

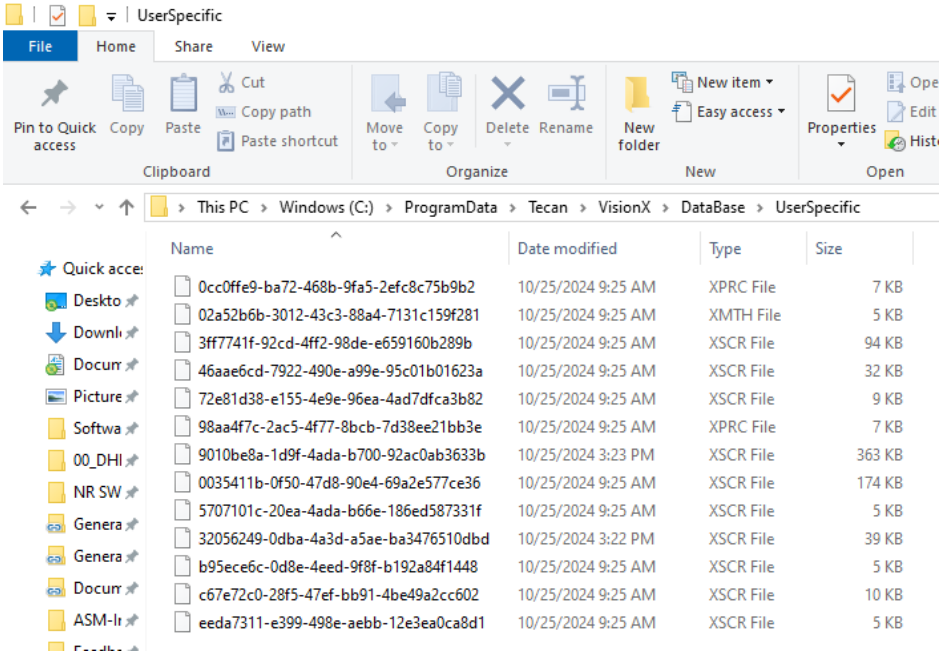
```
B;<ScriptGroup><Objects><Object
Type="Tecan.VisionX.ApplicationDriver.ApplicationDriverBase.A
pplicationDriverMacro"><ApplicationDriverMacro Version="1"
Name="RGA1_TransferLabware" ModuleName="RGA 1"
ExecutionTime="PT2S" IsBreakpoint="false"
IsDisabledForExecution="false"
LineNumber="1"><ExecutionSettings>&amp;lt;TransferLabwareComm
andParameters xmlns:i="http://www.w3.org/2001/XMLSchema-
instance"
xmlns="http://schemas.datacontract.org/2004/07/
Tecan.VisionX.Drivers.RobotDriverBase"&amp;gt;&amp;lt;FixedSi
te&amp;gt;false&amp;lt;/
FixedSite&amp;gt;&amp;lt;Labware&amp;gt;SourcePlate[001]&amp;
lt;/
Labware&amp;gt;&amp;lt;Location&amp;gt;Nest7mm_Pos&amp;lt;/
Location&amp;gt;&amp;lt;MoveToBase&amp;gt;false&amp;lt;/
MoveToBase&amp;gt;&amp;lt;OnTheFlyTool&amp;gt;&amp;lt;/
OnTheFlyTool&amp;gt;&amp;lt;Site&amp;gt;1&amp;lt;/
Site&amp;gt;&amp;lt;UseOnTheFlyTool&amp;gt;false&amp;lt;/
UseOnTheFlyTool&amp;gt;&amp;lt;/
TransferLabwareCommandParameters&amp;gt;</
ExecutionSettings></ApplicationDriverMacro></Object></
Objects><Name></Name><IsBreakpoint>False</
IsBreakpoint><IsDisabledForExecution>False</
IsDisabledForExecution><LineNumber>0</LineNumber></
ScriptGroup>
```

There is an easy way to get the needed serialization:

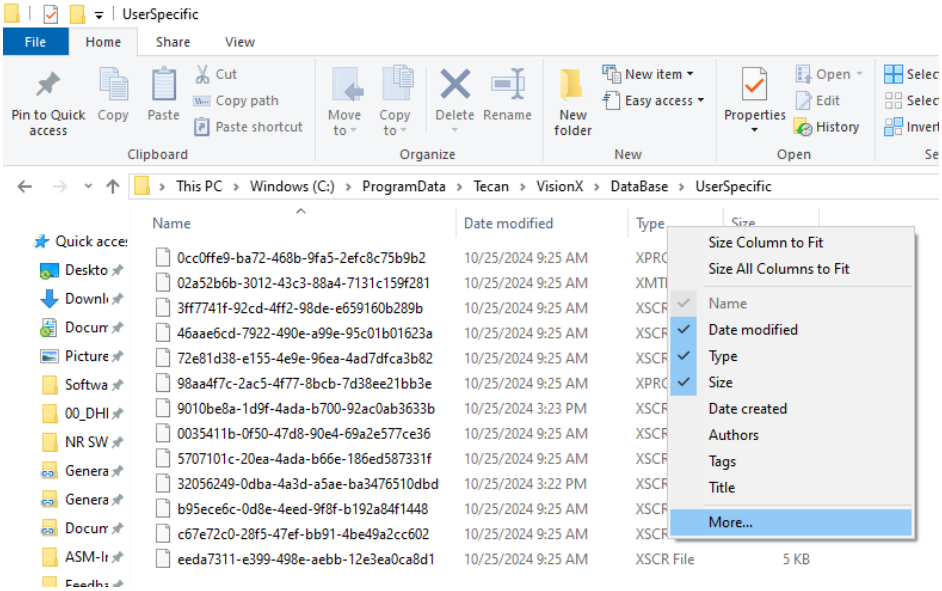
1. Create a Script with the command you want to add.
2. Parameterize and test the command.
3. Navigate to folder C:\ProgramData\Tecan\VisionX\DataBase\UserSpecific.
4. Open the script in a text editor like Notepad or Notepad++
5. Search for “<ScriptGroup>”
6. Copy the text from “<ScriptGroup>” to “</ScriptGroup>”.
7. Add a new line with “B;” to your .gwl file and append the copied text to this line.
   Remove all line breaks which were added.
   The complete text block must be part of the same line.

To find your script, add a column showing the FluentControl names:

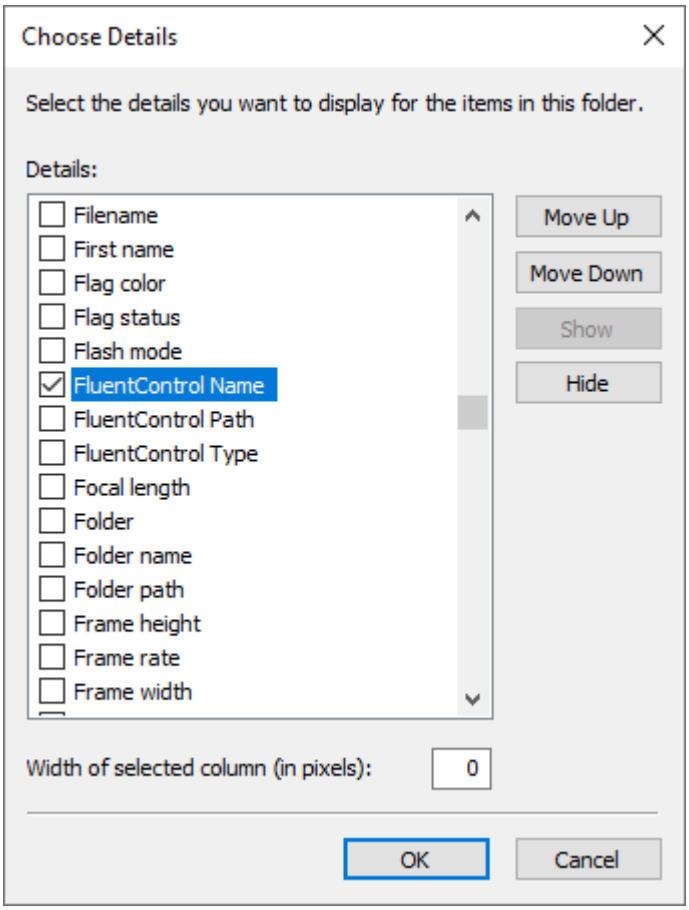
1. Open C:\ProgramData\Tecan\VisionX\DataBase\UserSpecific in the Windows Explorer.



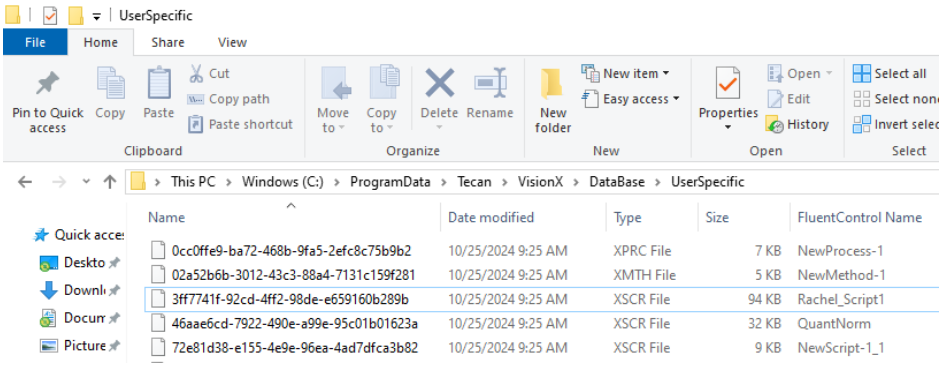
2. Right-click on any column header and select “More…”.



3. Navigate to “FluentControl Name” and check the checkbox. Press OK.



4. Now you have a column displaying the names of the objects:



11.17.5 Execute Worklist

The **Execute Worklist** command processes the output of all **Load Worklist** commands which have been executed either since start of the script or since the execution of a last, previous **Execute Worklist** command, in the chronological order they were loaded.

FluentControl will at runtime create a temporary script from the worklist and execute that temporary script. This allows you to see exactly which steps have already been executed if you pause and resume the pipetting or are using Method recovery.

For very long worklists, displaying the temporary script may cause significant delays. You can avoid those delays when executing very long worklists by disabling the option **display sub-routine execution during runs** in **Configure System**, **General Settings**, **Application Settings**.

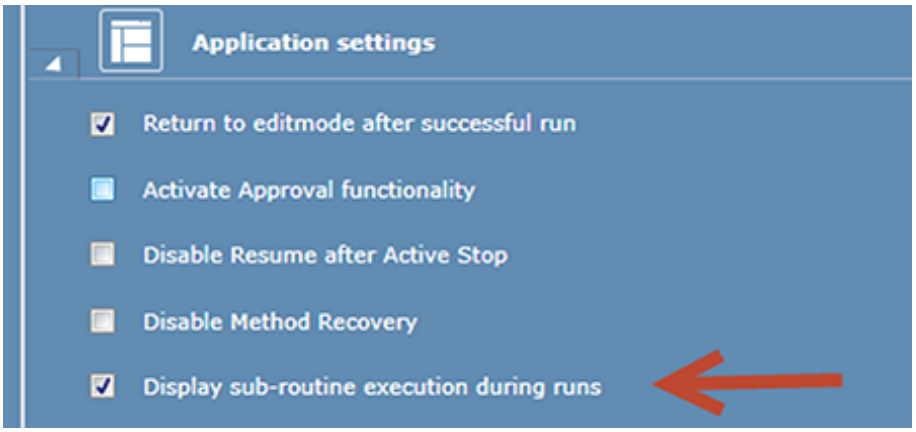


Fig. 345: Application settings

11.18 Sample Tracking

The **Sample Tracking** functions are controlled by special script commands.

Sample Tracking is a software add-on for FluentControl. It allows you to record sample and labware identities, track samples and labware through pipetting and robotic steps and to store process-related information, raw data and reduced data. The recorded data is stored centrally in an SQL database.
For more information about Sample Tracking, refer to the Sample Tracking Manual in the Documentation folder on the FluentControl installation medium.

Sample Tracking is not designed to work with duplicate barcodes. If duplicate barcodes are part of the workflow, make sure they are renamed to be unique before the Labware is registered in Sample Tracking. For example, suffixes or index numbers may be added using the “GetAttribute()” and “SetAttribute()” functions.

The following special characters must not be used in barcodes: ! % & ~ ( ) = | as well as carriage return and line feed.

Prerequisites:

- ✓ Make sure source/destination labware has barcodes.
- ✓ Register source labware that contains liquid that you want to track.
- ✓ Do not register destination labware!

1. Run Step 3 Install Sample Tracking of the FluentControl Installer and install **Sample Tracking Embedded**.
2. On the Settings menu, click **Configure System** and navigate to **Drivers** > **Sample Tracking Driver**.

   Sample Tracking driver in the FluentControl setting page is enabled.
3. To determine the format for the reports, use the **Generate Report** command in your script.

11.18.1 Sample Tracking, Commands, Overview

Tab. 23: Overview of Sample Tracking commands

Command	Cross reference to the Sample Tracking manual
Register Labware	**RegLabware** - This command is used to register labware and cavities.
Generate Report	**PrintRep** - This command is used to automatically generate printed reports.
Set Values	**SetValues** - The SetValue command provides the setting of values of identified tubes and wells.
Get Value	**GetValue** - The GetValue command is used to retrieve values of identified tubes or wells in Sample Tracking to script variables.

11.18.2 Generate Report

The **Generate Report** command is used to generate a sample tracking report as an execution step in a script. Optionally a printed report and a CSV file (e.g. for machine readable export) can be created. The output path for the CSV files can be adjusted under Configure System > Sample Tracking Driver > Report directory. Multiple labware selection is supported and for each selected labware separate reports are being created. PDF reports can be found here:
`C:\ProgramData\Tecan\Sample Tracking\Reports`



Sample Tracking reports will:
- Flag the sample as “Nothing was pipetted” if the error handling option “Ignore and continue” is chosen after an Aspiration Supervision Error.
- Flag the sample as “Not enough liquid” if the error handling option “Dispense back and deactivate tip” is chosen after an Aspiration Supervision Error.
Sample Tracking will show a warning for any pipetting steps that were interrupted by an Active Stop.

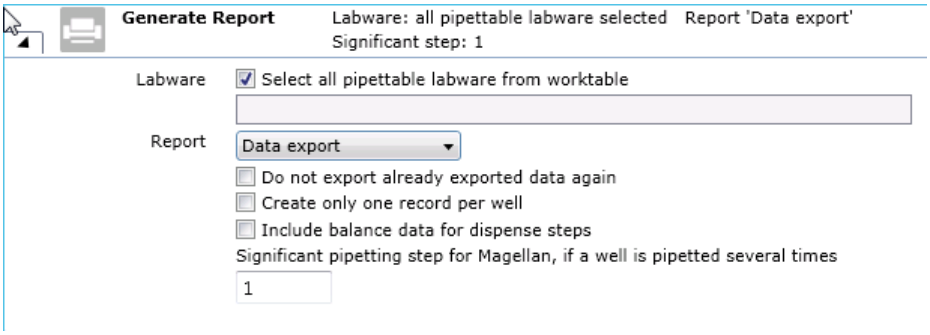


Fig. 346: Sample Tracking: Generate Report

Parameters

Labware

This panel shows the labware for which FluentControl will create output files. A separate report file is created for each labware. Specify a single or multiple labware for which a report shall be generated.

Add labware to the list by first clicking in the **Labware** box and then clicking on the desired labware on the worktable. To remove an entry, select the line in the list and press the **Delete** key (or click the labware on the worktable a second time).

To create reports for all labware on the worktable, select the **Select all pipettable labware from worktable** check box. In this case, the list of labware you have specified is ignored. The description then shows that all labware is selected.

A tube runner will be treated like a regular labware element: the tubes are treated like wells and the tube runner is treated like the labware element.

Do not export already exported data again check box

Do not export already exported data later again. If you use the **Generate Report** command more than once in the same script, select this check box if you don't want to export pipetting data again which has already been exported. This setting deletes the internal pipetting records for all prior pipetting operations, not only for the labware specified in the Generate Report command.

Create only one record per well check box

FluentControl normally outputs one record for each **Aspirate** or **Dispense** command for each well. If you select this check box, FluentControl only writes one record for each well. The record will then contain information on the accumulated errors (bitwise addition of all error codes) and the accumulated total pipetting volume (the sum of dispense volumes—i.e., positive values—and aspirate volumes—i.e., negative values).

Sample Tracking does not track the detected volume. In the 'Data Export' report, the column 'DetectVol' is therefore empty.

Include balance data for dispense steps check box

The **Liquid Arrival Check (LAC)** feature verifies that the correct liquid volumes have been dispensed by weighing the labware before and after pipetting. The labware must be put onto a special balance (pooling balance) for the dispense operation. Select this check box if you want to include the weight measurements in the report file. In this case, the report file will include four extra columns if a **Liquid Arrival Check** was carried out when dispensing.

Significant pipetting step for Magellan, if a well is pipetted several times box

Specify the relevant pipetting step for Magellan.

If you plan to use a Tecan Microplate Reader (photometer) in your script, specify which pipetting step is relevant to the Magellan data reduction software if the same well will be pipetted several times. For example, if you pipette the sample and then pipette a reagent, Magellan needs to know which step pipettes the sample. Information on the significant pipetting step will be included in the labware report file which the **Generate Report** command creates for the Magellan software.

If a report shall be generated that references data stored in external files, such as a plate map CSV files, the required files must be stored locally and not on a network location.

11.18.3 Get Value

The **Get Value** command is used to download values of registered tubes or wells from the Sample Tracking server and to store these values locally in script variables of FluentControl. This command is only used if the labware, such as a microplate, is already known to the Sample Tracking server. Labware is identified by either labware barcode and position in labware or the tube/well barcode. The labeling of all items reflects the function it represents and varies from the titles in this description.

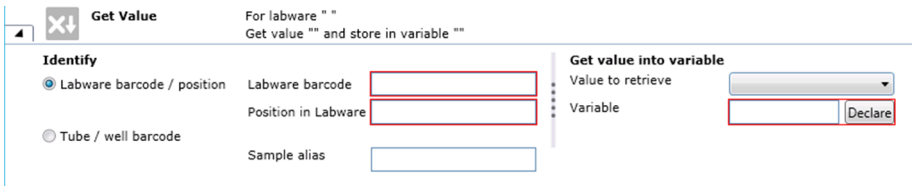


Fig. 347: Get Value

Identify: In this section the user can provide the identification information for a labware.

Labware barcode/position option: In this section the user can provide the identification information for a labware.

- **Labware barcode**: Expression to set a value for the labware barcode.
- **Position in Labware**: Expression to set a value for the container position in labware.

Tube/well barcode: Expression to set a value for the tube/well barcode for the container.

- **Sample Alias**: Optional value as an expression to set a value for the sample alias.

Get value into variable

Value to retrieve: ComboBox to retrieve a value to be stored in the given variable.

Variable: Expression to select a variable and a declare button.

11.18.4 Register Labware

The **Register Labware** command is used to register labware, cavities and the associated samples in the cavities. The labware selected for this command has to be accessible during the execution of this command. The labeling of all items reflects the function it represents and varies from the titles in this description.

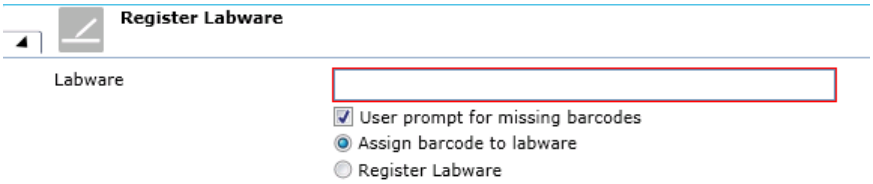


Fig. 348: Register Labware

Labware: Expression to select the available labware on the worktable.

AssignBarcode: To set if a single barcode is to be assigned for the whole labware if it has no barcode.

Register Labware: To generate barcodes for the labware and for all containers within if they have no barcode.

Concerning the labware selection: A tube runner will be selected as a single unit: the tubes are treated like wells and the tube runner is treated a piece of labware. It is not possible to select a single tube on its own. Clicking on a tube will cause its tube runner to be selected.

11.18.5 Set Values

The SetValue command is used to modify values on labware entries on the Sample Tracking Server. The command can thus only be used if the labware to be edited (e.g., microplate) has been previously registered on the Sample Tracking server. The SetValue command provides the setting of values of identified tubes and wells. Labware is identified by either labware barcode and position in labware or the tube/well barcode. The labeling of all items reflects the function it represents and varies from the titles in this description.

The **Set Values** command is used to modify values on labware entries on the Sample Tracking Server.

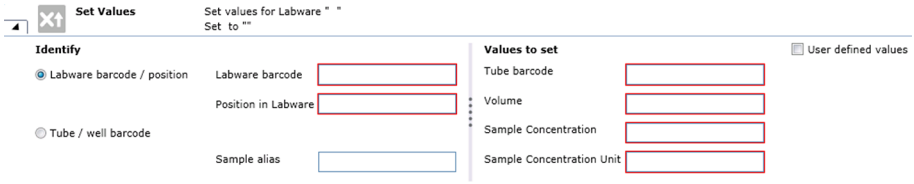


Fig. 349: Set values

Identify: In this section the user can provide the identification information for a labware.

Labware barcode / position

- **Labware barcode**: Expression to set a value for the labware barcode.
- **Position in Labware**: Expression to set a value for the container position in labware.

Tube / well barcode: Expression to set a value for the tube / well barcode for the container.

- **Sample Alias**: Optional value as an expression to set a value for the sample alias.

Values to set: In this section, the user can set further values regarding the liquid found in the tube / wells.

- **Tube barcode**: Expression to set a value for the container barcode.
- **Volume**: Expression to set a value for the container volume.
- **Sample Concentration**: Expressions to set a value for the sample concentration.
- **Sample Concentration**: Expressions to set values for the sample concentration unit.

User defined value: Expression to set a user defined value.

The Set Values command can only be used if the labware to be edited (e.g., microplate) has been previously registered on the Sample Tracking server with the Register Labware command.

11.19 Introspect

Introspect is a cloud-based service that allows for real-time monitoring of instruments.

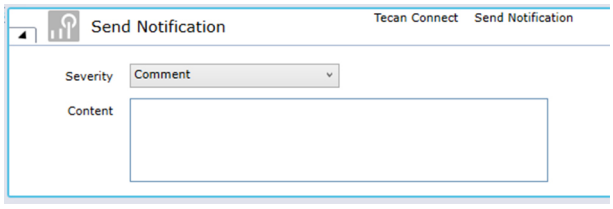


Fig. 350: Introspect

You can use the **Send Notification** script command to push a message to the paired Introspect smart phone application. The Introspect mobile app can be downloaded from the app stores for iOS and Android.

11.20 Worktable

11.20.1 Set Available Tips (Worktable) - Direct Command

Use the **Set Available Tips** command to set the next available position to fetch disposable tips from.

Then select the next available DiTi position in the labware. Mark the positions of the next available DiTi in the plan view of the labware layout. Now, FluentControl will update the next available DiTi position for all labware items matching the labware name defined by the user.

Parameter

Labware: In the **Labware** box, either type your expression or click the labware displayed in the **Worktable** editor window (*Worktable click*).

11.20.2 On Empty Tray

Use the **On Empty Tray** command to define a subroutine handling for the automatic replacement of empty DiTi trays of a specified type on the worktable.

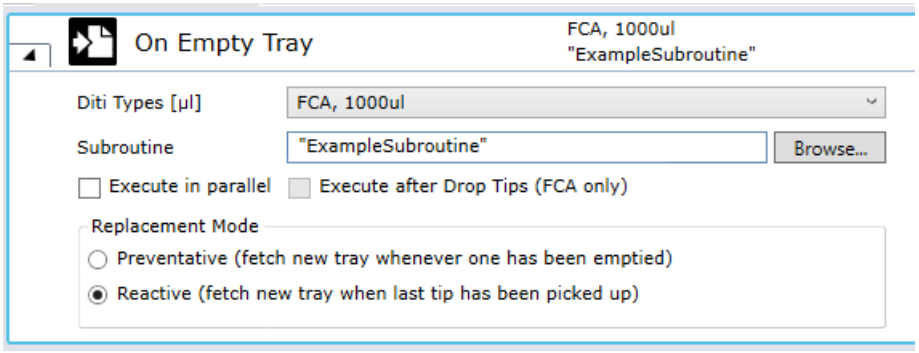


Fig. 351: On Empty Tray command

The **On Empty Tray** command stays active for the rest of the script. Depending on the selected replacement mode, whenever one tray of disposable tips (preventive mode) or when all trays are empty (reactive mode) the specified subroutine is executed.

The **Execute after Drop Tips (FCA only)** option is intended for systems with only one FCA arm and no RGA when working in preventative mode. Unchecked the system replaces an empty tip box as soon as the box has been emptied, which usually happens during or right after a Get Tips command. However, this is not possible with only a FCA on the system, as FCA channel grippers can't be picked up while tips are mounted. Therefore, when the option is checked, the box will only be replaced after the tips have been dropped.

Make sure, there are always at least two tip boxes of the same tip type available on an accessible position on the worktable to ensure enough tips can be mounted for a given pipetting sequence.

*The option **Execute in parallel** should only be chosen if you use the RGA to remove the empty tray and if your worktable layout permits that the RGA removes or replaces the empty tray while the MCA or FCA are pipetting.*



Fig. 352: Replace Used DiTi Racks command

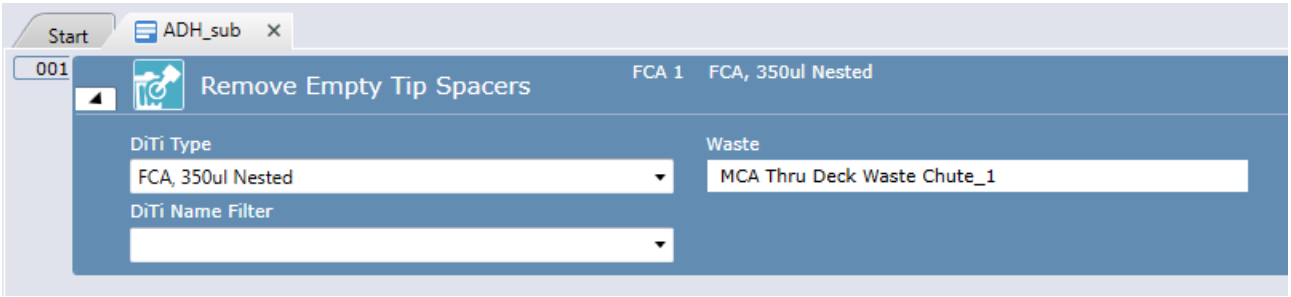


Fig. 353: Remove Empty Tip Spacers command

Typically, the subroutine that you link to will contain an RGA **Replace used DiTi Racks** command OR and FCA **Remove Empty Tip Spacers** command.

11.20.3 Import Sample IDs from File

Loads labware barcodes from a TXT or CSV file.

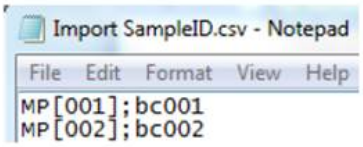
Path and filename of sample ID file box to specify a path and filename of the sample ID file box.

Prerequisites:

- The sample ID file must be a TXT or CSV file containing one sample ID per line.
- Each line consists of the labware name and the sample ID, separated by ',' or ';'.

1. Click **Browse**.
2. Specify the path and file name of the file containing the sample IDs to import.

LabwareLabel;LabwareBarcode



LabwareLabel;WellBarcode;WellIndex *(*where the well index is to be specified 1-based)

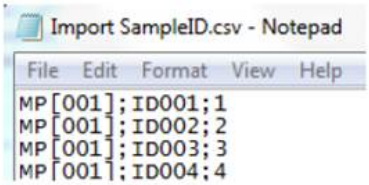


Fig. 354: Import sample IDs

11.20.4 Add Labware

Use the **Add Labware** command to programmatically populate the worktable with labware. You can then choose name, type, location and rotation of the labware at runtime.

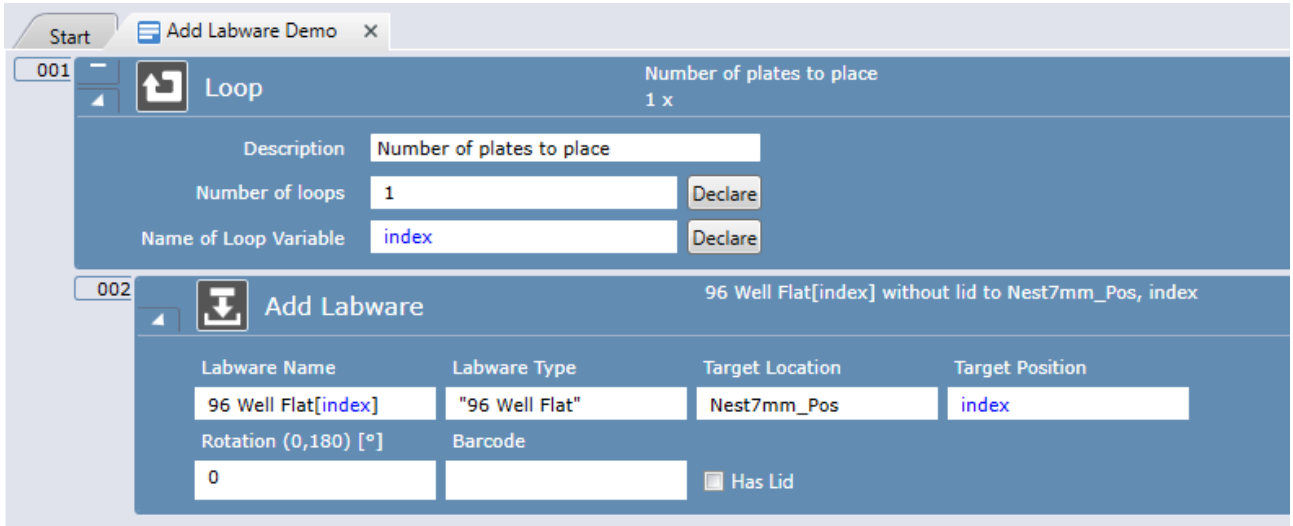


Fig. 355: Add Labware command

It is possible to select **Labware Type** and **Target Location** by clicking on the worktable or also by typing directly into the fields or by using variables.

11.20.5 Remove Labware

Use the **Remove Labware** command to programmatically remove labware from the worktable.



Fig. 356: Remove Labware command

11.20.6 Set Location

Use the **Set Location** command to programmatically set the location and rotation of a labware on the worktable. The set location command will not trigger any robotic action. You should use this command when the position of a labware changed after manual intervention of an operator or if the plate was moved using RGA single vectors or a 3rd party plate transport device.



Fig. 357: Set Location command

11.21 Te-Shake Shaker

The Te-Shake Shaker commands are used for controlling the Tecan orbital shaker for microplates or microplate-like racks. The shaker is used for mixing functions and can be equipped with an optional heating plate. The shaker plate is placed on an eccentric. Rotation of the rotor causes an orbital movement of the shaker plate. The orbit of the movement can be varied by changing the eccentric.



Fig. 358: Te-Shake Shaker

In Configure System at the Te-Shake driver configuration, the Device ID for each Te-Shake driver must be set to match the hardware address of the device.

The carrier definition for the Nest on top of the Te-Shake must be set as 'Carrier is device' with the corresponding Te-Shake driver selected at 'Device'. Further, it requires an attribute called DeviceIDs (String) with the value of the hardware address of that device.

Appropriate Te-Shake carrier(s) must be present on the active worktable during the initialization process. Therefore, open a script or worktable that contains your Te-Shake carrier(s) for successful initialization.

11.21.1 SetFrequency

The SetFrequency command sets the shaking frequency of the shaker.

Specify the required shaking frequency (range: 2–1800 Hz).

11.21.2 SetTemperature

The SetTemperature command sets the shaker to the specified temperature. Changes in temperature are not instantaneous. Make sure you specify the SetTemperature command early enough in your process or script to ensure that the shaker has sufficient time to heat up or cool down before using the Shake command.

Specify the working temperature of the heating plate option (15–90°C).

The heating plate cannot cool. Therefore, the minimum temperature of the labware is room temperature. Specify 0 to switch off the heating plate, or uncheck the Heater checkbox in the device configuration.

11.21.3 Start

The Start command starts the shaking procedure.

Specify whether you want the shaker to shake clockwise or counter-clockwise.

11.21.4 Stop

The Stop command stops the shaking procedure. This command has no execution parameters.

11.22 Heated Incubator



Fig. 359: Commands Heated Incubator

11.22.1 Turning On the Incubator and Turning Off the I/O State

The log files need to be checked for Incubator-related log messages. (See Checking the Log File.)

1. On the **Settings** menu, click **Configure System**.
2. Expand the **Drivers** > **Available Drivers** section.

   ***Available Drivers** are presented to you as a table with the following columns: **Enabled**, **Call name**, **Module name**, **Search pattern**, **Driver**, and **Version**.*



Fig. 360: I/O state selected drivers

3. Select the **Enabled** check box in front of **Incubator1** and click **Save & Close**.
4. Once more on the **Settings** menu, click **Configure System** and expand the **Drivers** section.

   *You find a new entry in the Drivers list named **Heated Incubator1**.*

5. On the **Heated Incubator1** entry, clear the **I/O State** check box and click **Save & Close**.

   The configuration change requires a restart of FluentControl.

11.22.2 How to duplicate the Incubator Driver

1. Open Configure System
2. Duplicate Incubator1 driver and assign callname and module name to Incubator2.
3. Save and close Configure System
4. Duplicate matching Incubator carrier and rename duplicated carrier accordingly (for example "4 Nest Incubator2")
5. Open worktable
6. Place "4 Nest Incubator2" carrier on the worktable
7. Open "4 Nest Incubator2" carrier and change General Settings -> Device to Incubator2
8. Save & Close Carrier editor
9. Open Configure System
10. Open settings for Incubator2 and change device id and number of slots, etc.
11. Save and close Configure System.

11.22.3 CloseDoor

The slot doors of the incubator opens by pressing a labware/finger against it. The doors then stay in the open position. Use this command to release the latches to close all of the doors.

Validate the application and consider the usage controls.

11.22.4 InsertPlate

Use this command to regulate the power of the incubator heater each time the robotic arm inserts a plate into the incubator. The measured temperature is written to the log file.

***InsertPlate** only works if incubator heater is on (temperature != 0). This can be done through driver configuration or SetTemperature command.*

Parameter

Slot: Specify the slot into which the plate should be inserted.

11.22.5 RemovePlate

Use this command to remove a plate from the incubator.

*To ensure that the incubator maintains a stable temperature, add a **CloseDoor** command directly after any **RemovePlate** command.*

Parameter

Slot: Specify the slot from which the plate should be removed.

11.22.6 SetTemperature

Use this command to set one or all slots of the incubator to the specified temperature.

Parameters

Set temperature to: Specify the required temperature of the specified slot (or all slots). The possible temperature range is defined in the firmware of the device. A value of 0 disables the heater.

for slot: Specify the incubator slot whose temperature should be set. Click the for slot arrow to see more options. Select ALL slots or a single slot number.

11.22.7 StartShaker

Use this command to start the optional shaker of the incubator device.

Parameter

Set speed to: Specify the shaking speed (a frequency in the range of 1 to 8.5 Hz).

11.22.8 StopShaker

Use this command to stop the optional shaker of the incubator device.

11.23 Te-VacS

The following table lists the commands for the vacuum separator.

To carry out vacuum separation in FluentControl, you must use a combination of the commands from the table below.

	**Apply Vacuum**: The **Apply Vacuum** command applies the vacuum to the SPE block. You must specify the required vacuum. Refer to section “Apply Vacuum Command” [▶ 432].
	**Check Waste Level**: The **Check Waste Level** command is used to check the liquid level in the waste bottle. Refer to section “Check Waste Level Command” [▶ 432].
	**Deactivate System**: The **Deactivate System** command is used to shut down the Te-VacS when you no longer want to use it in your script. It switches off the vacuum pump and vents all blocks and the vacuum tank. Refer to section “Deactivate System Command” [▶ 433].
	**Open Desalting Valve**: The **Open Desalting Valve** command opens the desalter valve to drain the residues. It is only relevant for Te-VacS versions which are equipped with a desalter valve. Refer to section “Open Desalting Valve Command” [▶ 433].

	**Set Pressure Diff**: The **Set Pressure Diff** command starts the vacuum pump. You must specify the required vacuum. The vacuum is not applied to either of the SPE blocks and the script or process does not wait for the vacuum to build up. Refer to section “Set Pressure Diff Command” [433].
	**Vent**: The **Vent** command vents the vacuum from the SPE block. It is used to stop the solid phase extraction process. Refer to section “Vent Command” [433].
	**Wait For Pressure**: The **Wait for Pressure** command pauses the script until the pressure specified in a previous set pressure command is reached. Refer to section “Wait For Pressure Command” [434].

11.23.1 Apply Vacuum Command

The **Apply Vacuum** command opens a valve which applies the vacuum to the SPE block. You must specify the required vacuum. If you did not specify a previous **Set Pressure Diff** command, the vacuum pump is then started and pumps the entire vacuum path (“soft pumping”, refer to Te-VacS operating manual). The vacuum will be applied until a subsequent **Vent** command is executed.



Fig. 361: Apply Vacuum command

Apply to front: The **front** command opens a valve which applies the vacuum to the front SPE block that is connected to vacuum pipe 1.

Apply to rear: The **rear** command opens a valve which applies the vacuum to the rear SPE block that is connected to vacuum pipe 2.

Pressure: Specify the required vacuum: 30 – 700 mBar

11.23.2 Check Waste Level Command

The **Check Waste Level** command is used to check the liquid level in the waste bottle. The waste level is reported during process runtime in the **Messages**/**Error** window of the runtime controller.

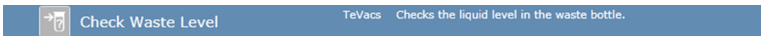


Fig. 362: Check Waste Level command

11.23.3 Deactivate System Command

The **Deactivate System** command is used to shut down the Te-VacS when you no longer want to use it in your script or process. It switches off the vacuum pump and vents all blocks and the vacuum tank. If you do not specify a **Deactivate System** command, the Te-VacS will be deactivated automatically when you quit FluentControl.

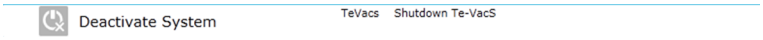


Fig. 363: Deactivate System command

11.23.4 Open Desalting Valve Command

The **Open Desalting Valve** command opens the desalter valve to drain the residues. It is only relevant for Te-VacS versions which are equipped with a desalter valve.

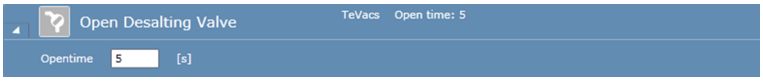


Fig. 364: Open Desalting Valve command

Opentime: Specify how long you want to open the desalting valve: 1 - 60 seconds

To view or edit the execution parameters again later, right-click on the script command in the **Script Editor** and choose **Edit** in the context menu.

11.23.5 Set Pressure Diff Command

The **Set Pressure Diff** command starts the vacuum pump. You must specify the required vacuum. The vacuum is not applied to either of the SPE blocks and the script or process does not wait for the vacuum to build up. Due to the time needed to build up the vacuum, you should specify this command at an earlier stage in your script to ensure that the instrument does not have to wait when you want to apply the vacuum with the **Apply Vacuum** command.



Fig. 365: Set Pressure Diff command

Pressure: Specify the required vacuum: 30 – 700 mBar

To view or edit the execution parameters again later, right-click on the script command in the **Script Editor** and choose **Edit** in the context menu.

11.23.6 Vent Command

The **Vent** command vents the vacuum from the SPE block. It is used to stop the vacuum separation process.



Fig. 366: Vent command

Apply to front: The **front** command vents the vacuum from the front SPE block. It is used to stop the vacuum separation process.

Apply to rear: The **rear** command vents the vacuum from the rear SPE block. It is used to stop the vacuum separation process.

11.23.7 Wait For Pressure Command

The **Wait for Pressure** command pauses the script until the pressure specified in a previous set pressure command is reached.



Fig. 367: Wait For Pressure command

11.24 Communication

Commands from the section **communication** allow to directly interact with the electronic boards of the Fluent.

Send ICP command *and* ***Send I2C command*** *are only shown when connected to a physical Fluent. They allow to send firmware commands directly, bypassing all intermediate software layers. These commands are meant for troubleshooting only. If needed Tecan customer support will advise how to use them.*

The commands **Read Digital Input**, **Set Digital Output**, **Set PWM Output**, and **Wait for Digital Input** allow to control input/output pins that are not used for internal functions such as doorlock and door sensor control, system liquid sensors or clamping of the active MCA384 tip carrier.

An FSE or the integration (Labwerx) group can connect additional devices to the input/output board which can then be controlled with the commands described below.

11.24.1 Read Digital Input

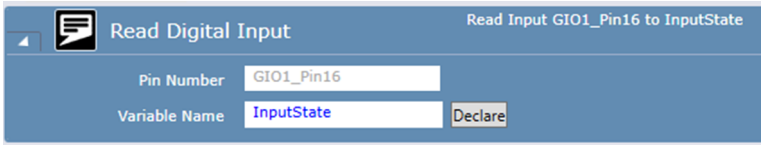


Fig. 368: Read Digital Input

Allows to read out the signal state of a selected pin on the input/output board. Typically, this is used to read a sensor returning a true/false result.

Pin Number: used to specify the identifier of the input pin. Please type in the letter **G** and select one of the available free pins that allow detecting input state. Pin number should be in the format **GIOn_Pinm**, where **n** is the TeGIO board number and **m** is the pin number (e.g., GIO1_Pin10).

Recommendation is to then define a variable of type string with a speaking name and enter the pin identifier as its value. For example, define a variable ReagentReservoirEmptySensor with the value GIO1_Pin16.

Variable Name: variable into which the sensor state is written. The value 0 will be returned when no signal is detected, the value 1 if a signal is detected.

11.24.2 Set Digital Output



Fig. 369: Set Digital Output

Allows to set the signal state of a selected pin on the input/output board. Typically this is used to enable/disable an external device.

Pin Number: used to specify the identifier of the input pin. Please type in the letter **G** and select one of the available free pins that allow detecting input state. Pin number should be in the format **GIOn_Pinm**, where **n** is the TeGIO board number and **m** is the pin number (e.g., GIO1_Pin10).

Recommendation is to then define a variable of type string with a speaking name and enter the pin identifier as its value. For example, define a variable RefillPumpOnOff with the value GIO1_Pin18.

Value [0 or 1]: must be 0 (output disabled) or 1 (output enabled) or a variable resolving to one of those values

Duration [sec]: used to specify for how long the new value is applied (controlled by Fluent firmware). Entering duration 0 means the pin will be enabled/disabled without time restriction. Recommendation is to use a time limit as a safety measure if problems are expected when the connected device is permanently enabled (i.e., use a time limit, if used to enable a refill pump to ensure that the devices switches off after the specified time, even when script execution is stopped, for example, due to the operator pausing the script or opening the door of the Fluent).

11.24.3 Set PWM Output

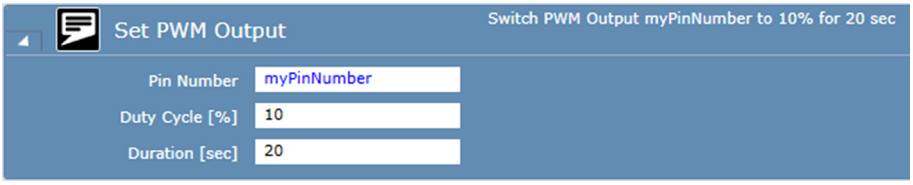


Fig. 370: Set PWM Output

Allows to set control a PWM output pin on the input/output board. Pulse width modulation (pwm) together with the specified duty cycle is used to control the average output power. This is typically used to control the speed/force for the connected external device.

Pin Number: used to specify the identifier of the input pin. Please type in the letter **G** and select one of the available free pins that allow detecting input state. Pin number should be in the format **GIOn_Pinm**, where **n** is the TeGIO board number and **m** is the pin number (e.g., GIO1_Pin10).

Recommendation is to then define a variable of type string with a speaking name and enter the pin identifier as its value. For example, define a variable VialGripper with the value GIO1_Pin18.

Duty Cycle [%]: must be between 0 and 100 percent. The duty cycle specifies the proportion on time during which power is delivered to the specified pin. This corresponds to the time-averaged output power.

Duration [sec]: used to specify for how long the PWM output is applied (controlled by Fluent firmware). Entering duration 0 means the pin's new PWM limit is applied without time restriction. Recommendation is to use a time limit as a safety measure, if problems are expected when the connected device is permanently enabled (i.e., use a time limit if used to enable a refill pump to ensure that the devices switches off after the specified time, even when script execution is stopped, for example, due to the operator pausing the script or opening the door of the Fluent).

11.24.4 Wait for Digital Input



Fig. 371: Wait for Digital Input

Allows to wait for a signal change of an input sensor.

Pin Number: used to specify the identifier of the input pin. Please type in the letter **G** and select one of the available free pins that allow detecting input state. Pin number should be in the format **GIOn_Pinm**, where **n** is the TeGIO board number and **m** is the pin number (e.g., GIO1_Pin10).

Recommendation is to then define a variable of type string with a speaking name and enter the pin identifier as its value. For example, define a variable ReagentReservoirEmptySensor with the value GIO1_Pin16.

Expected Value [0 or 1]: must be 0 (no voltage) or 1 (high voltage) or a variable resolving to one of those values

Timeout [sec]: used to specify the maximal wait time. If the specified event does not occur during this time an error message is shown and script execution then continues. Timeout must be between 1 and 3600 seconds.

11.25 Te-Chrom

The Te-Chrom smart command is a dedicated command for chromatography applications using the Tecan Te-Chrom device with the OPUS® RoboColumn® from Repligen.

The Te-Chrom smart command automatically executes all necessary actions to perform a chromatography workflow with up to 8 channels and processing up to 8 RoboColumns in parallel. Actions include a tip wash at the end of the workflow, one or multiple aspirate – dispense cycles depending on the volume, disengagement of the waste tray, movement of the Te-Shuttle and plate handling when collecting fractions or activation of the waste pump in case the fractions are not collected.
The Te-Chrom Base Segment carrier supports a left and a right location for the shuttle. The shuttle location determines which side is used for transfer labware and waste tray movements.
The collection plate must be a 96 well plate. Other plate types will be supported in future.

If the workflow requires that the collection plate is exchanged while the FCA tips are still in the RoboColumns, ensure that the RGA can execute the labware movements without requiring the FCA to move aside. For example, if the instrument arm order is FCA-RGA, place the Te-Chrom on the left side of the worktable and store the required plate labware on the right side of the worktable.

Parameters

Some of the parameters shown in the UI depend on the selected workflow, which can be either “Discard Fraction” where the liquid is not collected, “Collect Batch”, which collects the whole liquid of a RoboColumn in one destination well (i.e. one fraction is collected) and “Collect Fractions” where the volume is collected in multiple fractions. The default selection when using the first Te-Chrom command is “Discard Fraction”, which contains only parameters that are common to all workflows.

Common Parameters:

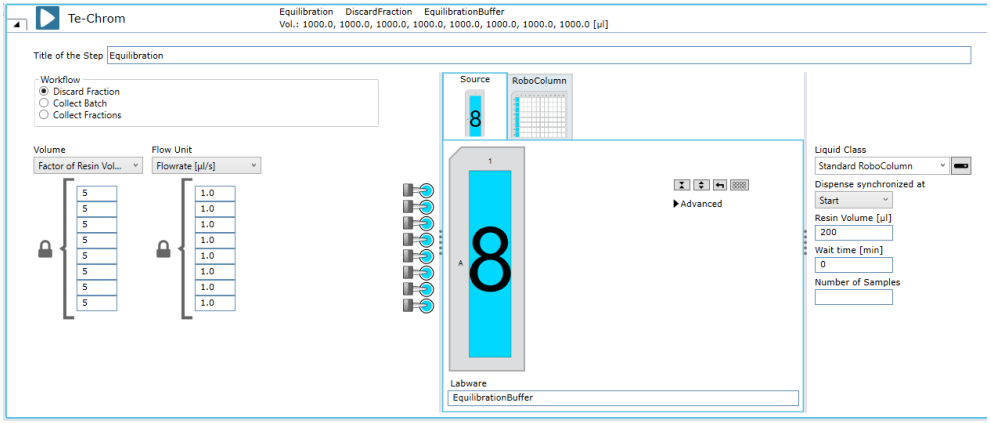


Fig. 372: Common Parameters

Title of the step: Free text field.

Workflow: Radio buttons to select the appropriate workflow, Discard Fraction, Collect Batch or Collect Fractions.

Volume: Define the total volume to pipette through a column. Volume can either be defined in µl or as a factor of the resin volume via the dropdown menu. Each channel can have a distinct volume, which is set by clicking the lock symbol to open the lock. If the volume exceeds the maximum capacity of the syringe, the system will transfer the liquid in multiple aspirate-dispense cycles with a wash in between. The volume per cycle is not divided equally, instead the maximum capacity is transferred first, followed by the remaining volume. Furthermore, it depends on the fraction volume or flowrate if the volumes are different for different channels.

Flow Unit: Defines the speed at which the liquid is dispensed through the columns. Can be either defined as flowrate in µl/s or as residence time in s via the dropdown menu. The units may be converted according to the following formula: Residence time [s] = Resin Volume [µl] / Flowrate [µl /s]. Each channel can have a separate flowrate, which is set by clicking the lock symbol to open the lock. The value for each channel is transferred into the variable “DispenseSpeed” in the MicroScript of the liquid class.

Channel Selection: UI to select the channels to pipette the liquid.

Source Tab: Define the aspiration position by selecting a source labware on the worktable and defining the well selection and tip spacing. Additionally, a well offset, an x- and y-offset and the compartment within the well can be selected in the Advanced section.

RoboColumn Tab: Define the dispense position of the RoboColumn labware on the worktable and define the well selection. Only wells within the same labware column are allowed within a single Te-Chrom smart command. Additionally, a well offset can be selected in the Advanced section. A function called **GetLastRoboColumnColumnIndex()** can be used to query the last used RoboColumn column index. For more information please refer to“Functions” [▶ 444]

Liquid Class: Define the liquid class to be used. This can be either defined from a dropdown list, as expression or as variable.

Dispense synchronized at: Define the pipetting behavior if not all channels pipette the same volume or use the same flowrate. The pipetting can either be synchronized at Start or at End. If synchronization happens at the start all channels will start pipetting simultaneously, but do not necessarily finish pipetting at the same time. If the synchronization happens at the end, the channels might start pipetting at different times but will finish pipetting at the same time.

Resin Volume [µl]: Define the volume of the resin within the RoboColumn. By default, the value is set to 200µl.

Wait time [s]: Define an optional delay at the end of the workflow.

Additional Parameters for Collect Batch and Collect Fractions Workflows

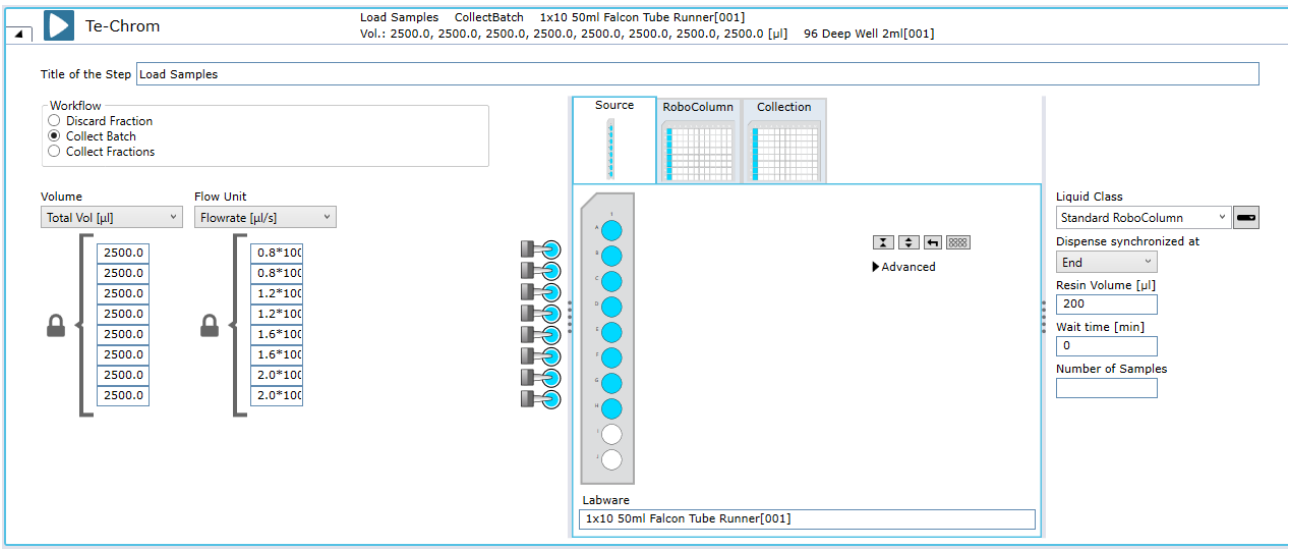


Fig. 373: Additional Parameters for Collect Batch and Collect Fractions Workflows

Collection Tab: Define the position of the collection labware on the worktable with well selection. Only available for Collect Batch and Collect Fractions workflows. The start well for the first fraction is set and if multiple fractions are collected, the well offset for subsequent fractions is automatically increased by 8 (i.e. the next column). Additionally, a well offset can be selected in the Advanced section. The well offset must be a multiple of 8, as the shuttle can only move in x to the next column. A function called **GetLastCollectionPlateColumnIndex()** can be used to query the last used Collection Plate column index. For more information please refer to “Functions” [▶ 444]

Additional Parameters for Collect Fractions Workflows

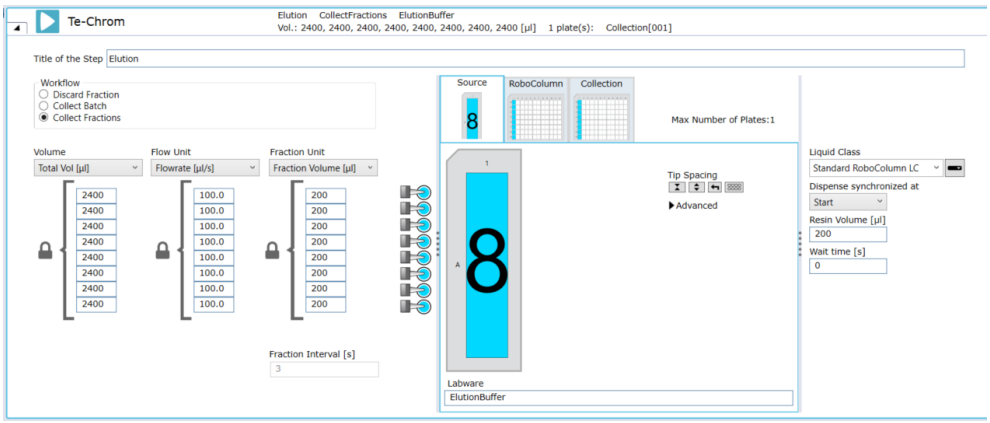


Fig. 374: Additional Parameters for Collect Fractions Workflows

Fraction Unit: Define the fractionation when collecting multiple fractions. Fraction unit can either be defined in µl per channel or a fraction interval in seconds. The fraction interval determines the amount of time a column of the collection plate is located underneath the columns before the shuttle moves the plate to the next

collection location on the collection plate. If fraction interval is used, the total volume is calculated automatically per channel based on flowrate and fraction interval.

Max. Number of Plates: Based on the total volume and the fraction volume the number of fractions and the number of collection plates required is calculated automatically.

Volume, flowrate and fraction volume may be entered individually per channel, but the following conditions apply:

- In a synchronized dispense at start workflow, all channels will start dispensing simultaneously. Channels that have more volume to dispense or are slower will finish pipetting later. This pattern also applies for each fraction if the channels have different dispense times per fraction.
- In a synchronized dispense at end workflow, all channels will finish dispensing simultaneously, so that channels that have more volume to dispense or are slower will start later. This pattern also applies for each fraction if the channels have different dispense times per fraction.
- If the total volume exceeds the syringe volume, the system will execute multiple aspirates, dispenses and wash cycles.
- If multiple aspirate-dispense cycles are required and channels have different flowrates, the volume for each channel per aspirate-dispense cycle is divided in a way that optimizes the total time and not slow down faster channels.
- If multiple fractions are collected and the fraction volume exceeds the total volume, the fraction volume will be adjusted to the total volume.
- If the fraction volume is not a full aliquot of the total volume the remaining volume will be pipetted as separate fraction either as last fraction (in case of a synchronized dispense at end) or as first fraction (in case of a synchronized dispense at start).

Additional Parameters for Collect Batch and Discard Fraction Workflows

Number of Samples: Define the number of samples to be pipetted. If no value is provided the number of selected wells will be used. In case the specified number of samples is smaller or equal to the number of selected wells, the system will use the selected wells until the number of samples has been reached. If the number of samples (source and collection wells) is greater than the number of selected wells, multiple pipetting cycles will be executed until the number of samples has been reached. After each cycle the next source and collection column will be used. If the source labware is a single trough, the system will aspirate from the same source well for each cycle.

11.26 Capture Worktable Command

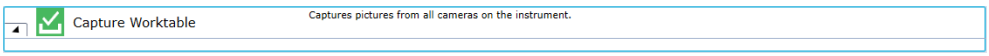


Fig. 375: DeckCheck Capture Worktable

TThe command explicitly takes the picture of the worktable using the DeckCheck cameras.
The worktable pictures taken are saved in the “%programdata% \Tecan\LoggingServer\LogFiles\DeckCheck Pictures”.
Each worktable picture file name contains the cameraID(Name) and a timestamp, the file name format is “<CameraID>_YYYYMMDD_HHmmSS.png”. A log entry is

created per each saved worktable picture, which contains the link to the file. If the camera cannot take a picture, the failure error is logged and the trace view warning is created, the run continues without user interaction and the run execution warning is generated upon the end of the run.
This DeckCheck command has no parameters.

12 Functions and Operators

For the examples below, the following variables were declared:

Name	Type	Value	Scope
DblVar1	Floating Point	12.34	Script
IntVar1	Integer	23	Script
StrVar1	String	StartValue	Script
ArrayInt1[]	Integer	0	Script

12.1 Operators Supported by Expression Solver

The following table lists all operators supported by the Expression Solver in order of precedence, from highest to lowest precedence:

Operator Symbols	Description	Example	Result
()	Parenthesis to group by evaluation order	3*(1+2)	9
^	power of	2^7 2^IntVar1	128 8388608
* / DIV MOD &	multiplication division modulo division (i.e. integer division) modulo logical (bitwise) AND	DblVar1*8 230/IntVar1 IntVar1 Div 8 IntVar1 Mod 8 0101 & 0011 3 & 5 3 & StrVar1	98.72 10 2 7 1 1 ERROR
+ - | XOR 	addition, string concatenation subtraction, unary minus (numeric negation) logical (bitwise) OR logical (bitwise) XOR	IntVar1 + DblVar1 "This value " + IntVar1 + StrVar1 IntVar1 – DblVar1 20 – IntVar1 -0.57 -DblVar1 0101 | 0011 3 | 5 3 | StrVar1 0101 XOR 0011 3 XOR 5 3 XOR StrVar1	35.34 This value 23StartValue 10.66 -3 -0.57 -12.34 111 7 ERROR 110 6 ERROR

Operator Symbols	Description	Example	Result
:=	Variable Name := Value Assigns Value to the Variable Returns Value as the result	Set StrVar1 to IntVar1:=50	StrVar1 = 50 IntVar1 = 50
= <> < <= > >=	Equality Inequality less than less than or equal greater than greater than or equal	IntVar1=DblVar1 IntVar1<>DblVar1 IntVar1<DblVar1 IntVar1<23 IntVar1<=23 IntVar1>=DblVar1 IntVar1>DblVar1	0 1 0 0 1 1 1
NOT	logical negation	NOT 1 NOT 0 IntVar1=23 AND NOT IntVar1=0 NOT IntVar1=23 AND IntVar1=0 NOT (IntVar1=23 AND IntVar1=0)	0 1 1 0 1
AND	conditional AND	IntVar1 =23 AND IntVar1=0 IntVar1 =23 AND IntVar1<>0 1 AND 1 1 AND 0 0 AND 0	0 1 1 0 0
OR	conditional OR	IntVar1=23 OR IntVar1=0 IntVar1=23 OR IntVar1<>0 IntVar1<23 OR IntVar1<>0 1 OR 1 1 OR 0 0 OR 0	1 1 0 1 1 0

The relational and comparison operators (=, <>, <, <=, >, >=) allow the use of numeric as well as string operands. Both operands in a comparison are required to be of the same type. The bitwise operators require both operands to be of integral type. The use of parentheses to influence the order/precedence of evaluation is allowed.

12.2 Functions

The following table lists the basic functions supported by the Expression Solver:

Function	Parameters	Description	Example	Result
abs(x)	x: numerical value	Returns the absolute value of x.	abs(DblVar1-IntVar1)	10.66
AdjustAccuracy()	Many	See AdjustAccuracy Function.xls	Located in the documents directory on the installation CD	
AdjustRangeByChannel()	Many	See AdjustAccuracy Function.xls	Located in the documents directory on the Installation CD	
AvailableVolume(labwareName, wellIndex, compartmentIndex, submergeDepth)	A float value in µl	The driver allows to access the usable volume (with submerge depth and up to z-max) for a given well through a function—e.g., AvailableVolume("Microplate[001].Well[1]",1,1,1.5)) . labwareName is the label of the labware, wellIndex the index of the well (1-based), compartmentIndex the index of the compartment within the well (usually 1) and submergeDepth the distance the tip goes into the liquid—i.e., a positive number means the tip is closer to the worktable).		
avg(x1,x2,…xn)	x1,x2,…xn: list of values	Returns the average of the specified values.	avg(DblVar1+IntVar1,IntVar1,IntVar1)	27.1133333333333

Function	Parameters	Description	Example	Result
binary(x,[L])	x: integral value L: minimum length of returned string (padding with leading zeros if necessary; optional)	Returns the binary (string) representation of an integral value.	binary(IntVar1) binary(IntVar1,8)	10111 00010111
CheckIfPhaseHeightIsInExpectedRange(LiquidLevelBeforePhaseDetection, PhaseHeight, ConfidenceLevel)	LiquidLevelBeforePhaseDetection: The liquid level before starting the phase detection [mm] PhaseHeight: the height of the phase [mm] ConfidenceLevel: allowed values are 90%, 95% and 99%	Checks if a detected phase height is in an expected range of a blood sample.	CheckIfPhaseHeightIsInExpectedRange(“120”, “80”, 95%)	-1: phase is below expected range 0: phase is in expected range 1: phase is above expected range
ceil(x)	x: numerical value	Rounds the given value to the nearest higher integer value.	ceil(3.01) ceil(3.995) ceil(3)	4 4 3
concat(s1,s2,…sn)	s1,s2,…sn: list of strings	Concatenates the specified strings.	concat("This value ", IntVar1, StrVar1)	This value 23StartValue
ConnectionString()	none	Returns the connection identifier for the connected instrument. This can be used to access low level functions.	ConnectionString()	USB:TECAN, FLUENT,1901001234
count(var)	var: array variable	Returns the number of elements present in the specified array (not the upper bound).	count(ArrayInt1[])	0
CurrentUser()	none	This function returns the name of the currently logged in user.	CurrentUser() Dependent on logged in user	No Login Required Admin Etc.

Function	Parameters	Description	Example	Result
CurrentBottleVolumeMl()	i300, port	Returns the current volume of the bottle on given port of given i300	CurrentBottleVolumeMl(0,1)	Current volume in ml e.g. 10.66
date()	none	Returns the current date in YYYY-MM-DD.	Date()	2024-07-27
decimal(x)	x: binary string	Returns the integral value of a binary string.	decimal(11011) decimal(2)	27 Error
DetectedVolume(arm, tip)	arm: Arm Name tip: Tip Index	If liquid level detection is used, the FCA pipetting commands automatically store the detected volumes for each tip in µl. The value is updated by each subsequent FCA pipetting command which uses liquid level detection. The value -1 is reported if the tip was used with liquid level detection disabled or was not part of the pipetting command. This also applies to the Detect Liquid command.	DetectedVolume("FCA 1", 1)	0 (or most recent volume)
DetectedVolumeMCA(arm)	arm: Arm Name	If liquid level detection is used, the MCA96 aspirate command automatically stores the detected volumes for each tip in µl. The value is updated by each subsequent MCA96 aspirate command which uses liquid level detection.	DetectedVolume("MCA96 1")	0 (or most recent volume)
ElapsedTime(x)	x: integer of started timer	Function returning the number of seconds having elapsed since the specified timer was last (re)started in the current script.	ElapsedTime(1) ElapsedTime(StrVar1)	305.4080805 (for example) ERROR
FileExist(str)	str: full path and file name	Check if a file exists and return “True” or “False”.	FileExist(StrVar1) FileExist("C:\Program Files (x86)\Tecan\FluentControl\SystemSW.exe")	=1 (meaning True) <>1 (meaning False)
FileLineCount	File Path	Use to count the number of lines in a text file. Returns an integer with the number of lines before the end of file.	FileLineCount(filePath)	5

Function	Parameters	Description	Example	Result
FileExcelRowCount	file path, (optional) column, (optional) start row	Use to count the number of rows in an excel file until the first blank row. Returns an integer. Column and startRow are optional parameters.	FileExcelRowCount("C:\excelFile.xlsx")	150
floor(x)	x: numerical value	Rounds the given value to the nearest lower integer value.	floor(3.01) floor(3.995) floor(3)	3 3 3
FormatDateTime(double dateTime, string format = YYYY-MM-DD HH:MM:SS)	Parameter format is optional.	If format is not provided, the function takes the yyyy-MM-dd HH:mm:ss as the default format. Otherwise the function takes the given format and converts the DateTime/ Date from a long number to a string. Example formats	FormatDateTime("636841440000000000","yyyy-MM-dd")	2019-01-27
Format(double number, string format)	Both parameters are required.	Takes a number, and uses a format to format the given number. Example formats	Format(1252643,"e2") Format(123.456,"f2")	1.25e+006 123.46
GetAttribute	labwareName, attribute	Gets a labware attribute like barcode, well concentration or custom attribute. For getting the labware barcode attribute name “Barcode” or “ID” can be used. For well barcodes use “Well[<well index>]."Barcode" where <well index> is the 1-based well index. Likewise you can query the well concentration via "Well[<well index>].Concentrat”on".	GetAttribute(“FCA_1000ul Filtered[001]", “ID”) GetAttribute(“Source[001]", “Well[1].Barcode”) GetAttribute(“Source[001]", “Well[96].Concentration”)	Current value of the attribute. Examples: “MyBC 1” 0.4
GetExpectedPhaseHeight(labwareName, well, compartment, liquidVolume)	labwareName: name of the labware. well: 1-based well index. compartment: 1-based compartment index. liquidVolume: volume in [µL]	Gets a height in mm that represents the expected height of a phase in a blood sample.	GetExpectedPhaseHeight("G15ml_Falcon[016]","1","1","6000")	34.45 [mm] (or other value for expected phase height [mm])

Function	Parameters	Description	Example	Result
GetCoverSiteName & GetCoverSiteIndex	labware name	Returns the location and site names of given plate for where a lid or a stacked labware would rest.	GetCoverSiteName (96 Well Flat[001]) & GetCover- SiteIndex(96 Well Flat[002])	GetCoverSiteName = 96_Lid_Site & GetCoverSiteIndex = 3
GetExpectedStepStartTime	script name	Returns the expected start time for the iteration of the process step script. This can be used in combination with the Send Notification command to push the information to Introspect to e.g. show the user when a specific step is finished.	GetExpectedStepStartTime("MyScript1")	21-09-17 15:43:55
GetLabwareByLocation	location name, position	Returns the labware name from a location at a specified position.	GetLabwareByLocation("Incub6_Pos", 4)	384 Well[001]
GetLabwareLabel	barcode	Use to fetch plates from locations when only the barcode is known. For example, can be used with a carousel after an inventory scan to fetch a plate to use with a worklist.	GetLabwareLabel("BC12345")	MTP96[001]
GetLastCollectionPlateColumnIndex()	none	Returns the last pipetted column index of the Collection plate labware as integer. The value will be (re-)set when executing the Te-Chrom Smart Command. The function always returns the highest number if different channels have had different numbers of fractions. The maximum number is not limited and the function considers multiple plates with the same index.	GetLastCollectionPlateColumnIndex()	16 (if the liquid was dispensed into column 4 of the Collection Plate with Index 2 in the Te-Chrom Smart Command) or any other integer.
GetLastRoboColumnColumnIndex()	none	Returns the last pipetted column index of the RoboColumn labware as integer. The value will be (re-)set when executing the Te-Chrom Smart Command. The max.value is 12.	GetLastRoboColumnColumnIndex()	3 (if the liquid was dispensed into column 3 of the RoboColumn labware in the Te-Chrom Smart Command) or any other integer.

Function	Parameters	Description	Example	Result
GetLocation & GetSite	labware name	Returns the location name and site number of a labware. Can be helpful to dynamically alter targets of RGA Execute Vector commands.	GetLocation(Labware- Name) & Get- Site(Labware-Name)	GetLocation = "Incub6_Pos" & GetSite = 3
GetLocationNameOfCarrier(<CarrierLabel>,<SiteIndex as shown in carrier editor>)	CarrierLabel, SiteIndex (optional, default value = 1)	Returns the location name of a site on a carrier that can be used, for example, in transfer-labware or add-labware commands	GetLocationNameOfCarrier("Nest 61mm[007]") GetLocationNameOfCarrier("6 Landscape 61mm Nest[001]",7)	"Nest61mm_Pos" WasteSlide_Site
GetNextIdleTimeForResource	resourceName minimumDurationOfIdleTime (in minutes)	Returns the next idle time which fits the given minimum duration for the given resource. If no idle time is found the end of the run will be returned.	GetNextIdleTimeForResource("MyWasher", 5)	21-09-17 15:56:14
GetNumberOfTipsInTipBox	labware name	Returns the number of available tips in a box.	GetNumberOfTipsInTipBox("FCA_ 1000ul Filtered[001]")	96
GetTimeSinceSuccessfulExecution	method name	Returns the time since a method was successfully executed. Useful if analysis on when a system care method's last run time may impact a given run.	Get- TimeSinceSuccessfulExecution(MethodName)	2.25
GetTipBoxStatus	labware name	Returns the tip status of a tip box, with 0 and 1 indicating which positions of tips in a box are in use or available.	GetTipBoxStatus("FCA_ 10ul SBS[001]")	"0101010101 11111100000 11001"
GetFluentIDLEDStatus & GetFluentIDSensorStatus	grid number	Returns the LED status and sensor status of the Fluent ID module. Useful to dynamically interact with operators at runtime.	GetFluentIDLEDStatus(Grid) & GetFluentIDSensorStatus(Grid)	GetFluentIDLEDStatus = 1 & GetFluentIDSensorStatus = 0
GetLocationNameOfCarrier(CarrierLabel, [SiteIndex])	CarrierLabel name of the carrier on the worktable SiteIndex (optional, default value = location name of site1)	Returns the location name to be used in a transfer labware or add labware command.	GetLocationNameOfCarrier("7mm Nest[009]")	Nest7mm_Pos

Function	Parameters	Description	Example	Result
GetPositionIndex-OfCarrier(Carrier-Label, [SiteIndex])	CarrierLabel name of the carrier on the worktable SiteIndex (optional, default value = location name of site1)	Returns the target position to be used in a transfer labware or add labware command. The position is counted over the full worktable whereas the SiteIndex is within one carrier.	GetPositionIndex-OfCarrier"7mm Nest[009]")	3
Humidity()	none	On systems equipped with a humidity sensors returns the last measured relative humidity value in %. The sensor automatically measures/refreshes the value several times per hour.	Humidity()	48
if(c,e1,e2)	c: condition e1: expression e2: expression	Evaluates the condition c. If c yields true, the value of e1 is returned, otherwise the value of e2.	if(IntVar1<DblVar1,"True","False") if(IntVar1<DblVar1 OR IntVar1>DblVar1, 1+4,"False")	False 5
InStr(inputString, searchString, numberOfOccurrence = 1, startPosition = 0, ignoreCase = false)	numberOfOccurrence, startPosition, ignoreCase are optional paramenters	If the search string is found in Input string, the function returns the index number (zero based) depending on the other parameters provided. Returns -1 if not found.	InStr("abccba","b",2)	4 (start position of the 2nd occurrence of the string "b" is 4)
LabwareCount(labwareGroupName)		Returns the number of labware (i.e., tubes) that have a barcode assigned. Usually used to count the number of tubes scanned with Fluent ID to use this number as sampleCount for further processing	LabwareCount("G13x100mm")	18 (assuming that 18 13x100mm tubes were scanned)
LastErrorId() *Fore more information see Infobox at end of this table*	none	This function returns the error message ID of the last command which reported an error during script runtime, example APP_xxx_yyy_zzz.	LastErrorId()	Depends on the error
LastErrorMessage() *Fore more information see Infobox at end of this table*	none	This function returns the error message of the last command which reported an error during script runtime.	LastErrorMessage()	Depends on the error

Function	Parameters	Description	Example	Result
LastErrorSeverity() *Fore more information see Infobox at end of this table*	none	This function returns the error severity of the last command which reported an error during script runtime as an integer.	LastErrorSeverity()	Depends on error
LastMovementResult(x, y)	x= Devicename, y= ChannelNumber	Returns 1= if an over current Error during Piercing Return 0 if there is no an over current error during Piercing	LastMovementResult("FCA 1",1)	
length(x)	x: any variable, string, constant or expression	Returns the total number of characters of the parameter.	length(StrVar1) length(IntVar1) length(DblVar1)	10 2 5
ln()	x: numerical value	Returns the natural (base-E) logarithm of the specified value.	ln(100) ln(IntVar1)	4.60517018598809 3.13549421592915
log()	x: numerical value	Returns the base-10 logarithm of the specified value.	log(100) log(IntVar1)	2 1.36172783601759
max()	X1,X2,...Xn: list of values	Returns the maximum of the specified values.	max(DblVar1,IntVar1,17,20.6) max(DblVar1,IntVar1,StrVar1)	23 ERROR
min()	X1,X2,...Xn: list of values	Returns the minimum of the specified values.	min(DblVar1,IntVar1,17,20.6) min(DblVar1,IntVar1,StrVar1)	12.34 ERROR
MicroScriptVariable (arm, tip, liquidClassName, variableName)	arm: Arm Name tip: Tip Index liquidClassName: Name of Liquid Class variableName: Name of variable of a MicroScript	Returns the internal address of a variable of a MicroScript. Can be used for the Register Event Handler command to react to value changes of the MicroScript variable.	MicroScriptVariable("FCA 1", 1, "Water Free Single", "dispenseVolume")	channel1|Water Free Single_Fca_default|dispenseVolume
MountedFESfinger([RGA_ID])	default RGA_ID = 1	In case of fixed finger returns tool name otherwise text (<no arm>, <nothing mounted>)		"Centric[001]"

Function	Parameters	Description	Example	Result
MountedM-CA384Adapter()		otherwise text (<no arm>, <nothing mounted>)		"384 Tips Combo[001]"
MountedM-CA384Adapter-Type()		otherwise text (<no arm>, <nothing mounted>)		"EVA"
ParseDate-Time(string date-Time, string format = YYYY-MM-DD HH:MM:SS)	Parameter format is op-tional.	If format is not provided, the function takes the yyyy-MM-dd HH:mm:ss as the default format. Otherwise the function takes the given format and converts the DateTime/ Date to a long number.	ParseDate-Time("Jan27-2019" ,"MMMdd-yyyy")	6368414400000000000
pow(x,y)	x: the base y: the power	Raises the specified base x to the power y.	pow(IntVar1,3) pow(3,3)	12167 27
random()	none	Generates a random floating-point number between 0 and 1. Is seeded by the system clock.	random() round(ran-dom()*(IntVar1-1)) +1	0 to 1, for ex-ample: 0.325044045841808 Integers from 1 to 23
ResetLastMove-mentResult(x,y)	x= Device-name, y= Channel-Number, Channel num-ber is op-tional, if not specified then reset on given FCA arm	It resets state for given FCA arm back to Success.	ResetLastMove-mentResult("FCA 1",1) ResetLastMove-mentResult("FCA 1")	
round(x,[p])	x: numerical value p: number of fractional dig-its (optional)	Rounds a specified value to the specified number of frac-tional digits (0 if not specified).	round(0.49) round(0.5) round(round(0.49,1)) round(10,2) round(10.195,2)	0 1 1 10 10.2
ScriptIteration()	none	Returns the iteration of the current script.	ScriptIteration()	1
ScriptName()	none	Returns the name of the cur-rent script.	ScriptName()	NewScript-1
SerialNumber()	none	Returns the serial number of the connected instrument.	SerialNumber()	1901001234

Function	Parameters	Description	Example	Result
SetAttribute	labware-Name, attribute, value	Sets a labware attribute like the barcode. For setting the barcode “Barcode” or “ID” can be used. For well barcodes use “Well[<well index>].Barco”de" where <well index> is the 1-based well index. Likewise you can set the well concentration via "Well[<well index>].Concentration".	SetAttribute(“FCA_1000ul Filtered[001]", “ID”, “MyBC”) SetAttribute(“Source[001]", “Well[1].Barcode”, “Well1”) SetAttribute(“Source[001]", “Well[96].Concentration”, “0.4”)	The assigned value
SetLastCollectionPlateColumnIndex(integer)	integer	Sets the last pipetted column index of the Collection plate labware as integer. This happens automatically inside the created subroutine of the TeChrom Smart Command, where the value is set to 0 at beginning of the subroutine and then set every time after the column index has increased.	SetLastCollectionPlateColumnIndex(13)	Sets the column index to 13 of the collection plate in use, which is the first column index of the second collection plate.
SetLastRoboColumnColumnIndex(integer)	integer	Sets the last pipetted column index of the RoboColumn labware as integer. This happens automatically inside the created subroutine of the TeChrom Smart Command, where the value is set to 0 at beginning of the subroutine and then set to the appropriate value.	SetLastRoboColumnColumnIndex(5)	Sets the column index to 5 of the RoboColumn labware in use.
SetTipBoxStatus	labware name, status	Sets the tip status of a tip box.	SetTipBoxStatus("FCA_ 1000ul Filtered[001]", "01010111010001111111101010101010101")	1 = successful, otherwise 0
sqrt(x)	x: numerical value	Returns the square root of the specified value.	sqrt(9) sqrt(InVart1)	3 4.79583152331272
substring(s,start,[count])	s: string value start: start position count: optional length	Extracts a substring starting at position start from specified string; optionally limiting the length of the returned string.	substring(StrVar1,5,3) substring(StrVar1,2)	Val artValue

Function	Parameters	Description	Example	Result
SyringeSize(arm,x)	arm: the name of the arm x: channel number	Returns the configured syringe size for the given channel.	SyringeSize("FCA 1", 1)	1250
teMotionPosition(identifier)	identifier: the device ID of the Te-Motion	Returns the current position of the given Te-Motion drive.	teMotionPosition("USB:TECAN, MYRIUS,1310005667/LIHA:1/DRIVE:X")	102.499
Temperature()	none	Returns the last measured temperature in °C on systems equipped with a temperature sensor. The temperature is automatically measured/refreshed several times per hour.	Temperature()	23.7
time()	none	Returns the current time in 24 hour format: HH:MM:SS.	time()	21:55:27
WorktableName()	none	Returns the name of the current worktable	WorktableName()	Base Worktable

The functions LastErrorId(), LastErrorMessage(), LastErrorSeverity() return an empty string if they are used without “Error Handler” command. If these functions are to be used, first an “Error Handler” needs to be defined. “OnErrorGoTo” command needs to be used to determine which error handler will be acting upon the occurrence of an error. Within the error handler the functions return the correct values.

12.3 Operands

An operand used in an expression can be any of the following:

- fixed numerical value, using the '.' character as decimal separator in case of a floating-point value
- string constant enclosed in double-quotes
- variable
- carrier, labware or location attribute

The use of array variables is permitted. Array elements are accessed using brackets to enclose the index; multi-dimensional arrays are supported. An array index itself may be an expression. To declare an array variable, enter the variable name directly in the Variables panel and add [] to the end of the name, for example arrayVar[] (see figure above). It is also possible to declare a multi-dimensional array by using a series of [] brackets: arrayVar[][][].

Variables in expressions are specified without scope identifier(s). The Expression Solver uses the scope of the current script to access the Variable Engine, which in turn tries to resolve variables within that scope first, working its way upwards in the scope hierarchy until the variable is found.

Carrier/Labware or location attributes are resolved using the current workspace. (At runtime, there is exactly one current workspace for all running scripts and/or processes.)

The labware index, site index and well number can be specified using a numeric expression.

12.4 Worktable Lookup Functions

All results in this table are dependent on labware selected and worktable layout.

This section documents the functions provided by the expression solver to lookup different attributes of labware and cavities:

Function	Parameters	Description	Example	Result
alphaNumeric(wellIndex, labwareName, [lengthOfNumericPart])	wellIndex: index of well within labware (1-based) labwareName: label of labware lengthOfNumericPart: optional length of numeric part in output	Returns the alphanumeric well identifier corresponding to the given well index, according to the orientation of the labware on the worktable.	alphaNumeric(19, "96 Well Flat[001]", 3) alphaNumeric(96, "96 Well Flat[001]")	C003 H12
calculateHeight(labwareName, well, compartment, volume)	labwareName: LabwareName[Index] well: LabwareName[Index].Well[WellNo] compartment: index > 0 volume: volume in µl	height within cavity in mm (floating-point)	calculateHeight("96 Well Flat[001]", 1, 1, 100)	3.013584721266666
calculateVolume(labwareName, well, compartment, height)	labwareName: LabwareName[Index] well: LabwareName[Index].Well[WellNo] compartment: index > 0 height: height within cavity in mm	volume at specified height in µl (floating-point)	calculateVolume("96 Well Flat[001]", 1, 1, 2.3)	76.321066528147

Function	Parameters	Description	Example	Result
column(loopIteration, labwareName)	loopIteration: current value of loop counter (> 0) labwareName: LabwareName[Index]	Well index to apply to pipette into column corresponding to loop counter.	column(3, ”96 Well Flat[001]”)	16
GetAttribute(labwareName)	labwareName: LabwareName[Index] attributeName: name of attribute to read	Retrieve the value of an attribute from a labware. Also allows accessing attributes of a well within the labware.	GetAttribute("Microplate[001]", "Barcode") GetAttribute("Microplate[001]", "Well[23].Volume")	"Sample-017" 73.2783
GetLocation(labwareName)	labwareName: LabwareName[Index]	Determines the location (= location group name) where a labware is currently placed on the worktable.	GetLocation("Microplate[001]")	"Nest_61 mm"
GetSite(labwareName)	labwareName: LabwareName[Index]	Determines the site (= site index) within the location group where a labware is currently placed on the worktable.	GetSite("Microplate[001]")	1
numeric(wellIdentifier, labwareName)	wellIdentifier: alphanumeric identifier of well labwareName: label of labware	Returns the numeric well index corresponding to the given well index, according to the orientation of the labware on the worktable.	numeric(A004, "96 Well Flat[001]") numeric(H12, "96 Well Flat[001]")	25 96

Function	Parameters	Description	Example	Result
plate(labware-TypeName, loopIteration)	labwareTypeName: name of labware type being pipetted loopIteration: current value of loop counter (> 0)	The label of the labware instance to use for pipet-ting, corre-sponding to the loop counter. (string)	plate(“96 Well Flat”, 4)	“96 Well Flat[004]”
quadrant(lab-wareName, loopIteration, headRows, head-Columns)	labwareName: Lab-wareName[Index] loopIteration: current value of loop counter (> 0) headRows: number of rows on pipetting head headColumns: number of columns on pipet-ting head	Well index to apply to pipette into plate quadrant corresponding to loop counter.	quad-rant(”384 Well [001]”,3, 8, 12)	16
row(loopItera-tion, labware-Name)	loopIteration: current value of loop counter (> 0) labwareName: Lab-wareName[Index]	Well index to apply to pipette into row corre-sponding to loop counter.	row(3, ”96 Well Flat[001]”)	2
SetAt-tribute(lab-wareName,at-tributeName)	labwareName: Lab-wareName[Index] at-tributeName: name of attribute to write value: New value to assign	Update the value of an at-tribute of a labware. Re-turns the new value having been as-signed.	SetAt-tribute("Mi-croplate[0 01]", "Bar-code", "Sam-ple-123") SetAt-tribute("Mi-croplate[0 01]", "Well[23]. Volume", "150")	"Sam-ple-123" 150
SetLoca-tion(labware-Name,loca-tion,Site)	labwareName: Lab-wareName[Index] lo-cation: Location group name Site: Index of site within location group (1 … n)	Moves a lab-ware to a given location and site on the worktable.	SetLoca-tion("Mi-croplate[0 01]", "Ho-tel_9Pos", 5)	

Function	Parameters	Description	Example	Result
tipAccess(labwareName, [well, compartment])	labwareName: LabwareName[Index] well: index of well within labware (1-based) compartment: optional index of compartment within well (1-based)	number of tips allowed to access the well in parallel	tipAccess("96 Well Flat[001]", 71) tipAccess("100ml_1", 1)	1 8
tipTouchDistance(labwareName, well, compartment, direction[, height])	labwareName: LabwareName[Index] well: index of well within labware (1-based) compartment: index of compartment within well (1-based) Direction: 0 = X, 1 = Y height: optional height within cavity in mm; if not specified, current Z-axis position of calling channel is used	Tip-touching distance (i.e. distance to rim of cavity) at given height in [mm] (floating-point)	tipTouchDistance(“96 Well Flat[001]”, 1, 1, 0)	3.3
xPosition(labwareName, well[, compartment])	labwareName: LabwareName[Index] well: index of well within labware (1-based) compartment: optional index of compartment within well (1-based)	X-Position of well (center) in mm	xPosition("96 Well Flat[001]", 1)	209.3
yPosition(labwareName, well[, compartment])	labwareName: LabwareName[Index] well: index of well within labware (1-based) compartment: optional index of compartment within well (1-based)	Y-Position of well (center) in mm	yPosition("96 Well Flat[001]", 71)	350.775
zBottom(labwareName, well[, compartment])	labwareName: LabwareName[Index] well: index of well within labware (1-based) compartment: optional index of compartment within well (1-based)	Z-Bottom coordinate in mm	zBottom("96 Well Flat[001]", 1, 1)	11.2

Function	Parameters	Description	Example	Result
zDispense(labwareName, well[, compartment])	labwareName: LabwareName[Index] well: index of well within labware (1-based) compartment: optional index of compartment within well (1-based)	Z-Dispense coordinate in mm	zDispense("96 Well Flat[001]", 1, 1)	19.6
zMax(labwareName, well[, compartment])	labwareName: LabwareName[Index] well: index of well within labware (1-based) compartment: optional index of compartment within well (1-based)	Z-Max coordinate in mm	zMax("96 Well Flat[001]", 1, 1)	11.7
zStart(labwareName, well[, compartment])	labwareName: LabwareName[Index] well: index of well within labware (1-based) compartment: optional index of compartment within well (1-based)	Z-Start coordinate in mm	zStart("96 Well Flat[001]", 1, 1)	21.6
zTravel(labwareName, well[, compartment])	labwareName: LabwareName[Index] well: index of well within labware (1-based) compartment: optional index of compartment within well (1-based)	Z-Travel coordinate in mm	zTravel("96 Well Flat[001]", 1, 1) zTravel("96 Well Flat[001]", 1)	29.6 29.6

Tab. 24: Syntax table

Syntax	Example
LabwareName[Index].Attribute	Microplate[001].Barcode
LabwareName[Index].Well[No].Attribute	Microplate[001].Well[1].Volume
LocationName.Attribute	DitiWaste.Barcode
LocationName[SiteIndex].Attribute	MP4Pos[1].Barcode

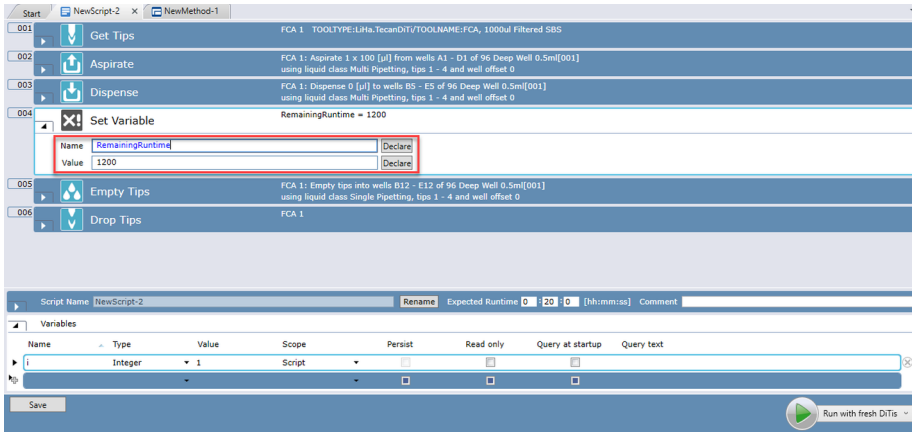
Other Variable Pipetting Loop functions that can be used outside of a Variable Pipetting Loop:

Function	Arguments	Description	Example	Result*
tipsNeeded(samples)	samples: number of samples to pipette	Tip mask required to pipette the remaining number of samples. Same as 2^samples – 1	tipsNeeded(5)	31
tipUseCount(tipNo, tipCount, samples)	tipNo: index of tip to evaluate (1-based) tipCount: number of tips available samples: total number of samples to pipette	Number of dispense steps required for the specified tip to pipette the total number of samples.	tipUseCount(3,8,27) tipUseCount(3,8,26)	4 3
tipsUsed(samples)	tipMask: bitmask of tips selected for pipetting	Tip mask required to pipette the remaining number of samples. In addition, the function reduces the number of remaining samples by the number of selected tips.	tipsUsed(5)	31

12.5 Internal Variables

12.5.1 Remaining Runtime

The internal variable "RemainingRuntime" enables a dynamical adjustment of the remaining runtime during script execution.
It can be set via the command "Set Variable" script command.
This allows the definition of dynamic remaining runtimes and real-time updates based on known interactions or different conditions to get a more accurate and flexible estimation of script completion.


Start
NewScript-2
NewMethod-1
001
Get Tips
FCA 1 TOOLTYPE:LiHa.TecanDiTi/TOOLNAME:FCA, 1000ul Filtered SBS
002
Aspirate
FCA 1: Aspirate 1 x 100 [µl] from wells A1 - D1 of 96 Deep Well 0.5ml[001]
using liquid class Multi Pipetting, tips 1 - 4 and well offset 0
003
Dispense
FCA 1: Dispense 0 [µl] to wells B5 - E5 of 96 Deep Well 0.5ml[001]
using liquid class Multi Pipetting, tips 1 - 4 and well offset 0
004
Set Variable
RemainingRuntime = 1200
Name
RemainingRuntime
Declare
Value
1200
Declare
005
Empty Tips
FCA 1: Empty tips into wells B12 - E12 of 96 Deep Well 0.5ml[001]
using liquid class Single Pipetting, tips 1 - 4 and well offset 0
006
Drop Tips
FCA 1
Script Name
NewScript-2
Rename
Expected Runtime
0
20
0
[hh:mm:ss]
Comment
Variables
Name
Type
Value
Scope
Persist
Read only
Query at startup
Query text
i
Integer
1
Script
Save
Run with fresh DiTis

13 Data Handling

This chapter provides information about the handling of user data. Log files may become huge. Depending on company standards you may wish to backup log files in regular intervals. After backup delete the log file to start afresh.

13.1 Saving User Data

Describes the concepts backup and restore of FluentControl.

If FluentControl is already installed on your PC, we recommend you to back up your FluentControl data files, scripts and processes to exchangeable media or to a network or temporary directory (not within the FluentControl <install_path>) before running the installation program.

Saving these files is especially recommended in the following cases:

- Before uninstalling (removing) the FluentControl software.
- Before installing a newer version of FluentControl.
- Before modifying an existing FluentControl installation.

13.2 Backup

When a computer write to disk action is interrupted, for example by power loss, it may be possible that restoring a backup is necessary to make FluentControl operational again. If the SVN Repository is impacted by such data corruption restoring a backup may be necessary to maintain the integrity of the Audit Trail.

Data to backup:

- **Database Folder**: This contains all the files necessary to run assays. It includes the definitions of Methods, Scripts, Liquid Classes, Carriers, Labware, etc.
- **SVN-Repository**: This contains the change history for all objects in the database. It enables the database restore points and “rolling back” changes to individual objects.
- **SVN-Password**: This is the system password FluentControl uses to communicate with the SVN. This is not a protected password, nor is it related to User Management.
- **Audit Trail Folder** (optional): This should be backed up if the change history of FluentControl’s electronic records is critical to maintain in the event of computer failure.

1. Shutdown FluentControl.
2. Shutdown SVN Service.
3. Backup Target: Create new Folder if necessary.
4. Copy Folders (with all subfolders and files) to Backup Target Folder: C:\ProgramData\Tecan\VisionX**Database**, C:\ProgramData\Tecan**SVNRoot** and optional C:\ProgramData\Tecan\VisionX\AuditTrail.
5. Create Registry Export of SVN-Password (and other FluentControl data): HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Tecan.
6. Following Command Line will do this for you: regedit /E BackupTargetFolder/Registry.reg HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Tecan.
7. Start SVN-Service again.

To restore a backup:

1. Shutdown FluentControl.
2. Shutdown SVN-Service.
3. Delete current Folders: C:\ProgramData\Tecan\VisionX**Database** and C:\ProgramData\Tecan**SVNRoot**.
4. Copy backed up folders back to original source.
5. If registry entries were changed: Import Registry File.

   This is required when FluentControl was removed completely and SVN-Password is lost.
6. Start SVN-Service again.

13.3 Restoring

1. Install FluentControl in the same version the backup has been run on.
2. Close FluentControl in case it is running.
3. Terminate all FluentControl processes in case they are running. These are for example:

   Mcs.exe

   Tecan.Licensing.ClientModule.exe
4. Stop the following services:

   Tecan FluentControl SVN

   Tecan FluentControl Watchdog

   Tecan.MAP.IAM.Service

   Tecan.LoggingService
5. Copy the data you previously backed up to their origin folders. Make sure the access rights of the files and directories are restored as they were on the backed-up system.
6. Restart the services that were stopped before.
7. Start FluentControl.

13.4 Archiving Log and Audit Trail Data

If you are required to comply with the FDA regulations, make sure archived copies of all Audit Trail and log files are kept. The relevant files are stored in the following directories and their subdirectories:

- `C:\ProgramData\Tecan\VisionX\AuditTrail`
- `C:\ProgramData\Tecan\LoggingServer\`
- `C:\ProgramData\Tecan\DataStore\IoT-Client\MAP.Services.Logging.Service\LogFile`
- `C:\ProgramData\Tecan\MAP.IAM`
- `C:\ProgramData\Tecan\Sample Tracking\AuditTrail\log`

The C:\ProgramData\Tecan\VisionX\AuditTrail directory is protected against unauthorized access. The AuditArch tool allows you to copy or move the log files and audit trail data files to another directory of your choice.

13.4.1 AuditArch Data Archiving Tool

The Audit Trail Archiving tool **AuditArch** allows you to archive FluentControl log files and audit trail data either manually or automatically. **AuditArch** will start in the background each time you start FluentControl and archive the log files if this is required by the currently configured archiving actions.

FluentControl can output a large amount of logging information, especially if the log level is set to 'Debug'. It is recommended to back up and then delete the old log files on a regular basis.

By default, log files from `C:\ProgramData\Tecan\VisionX\LoggingServer\` are moved to `C:\AuditTrail\Backup\AuditTrail` after 24 hours.

Start this tool via **Settings > Configure System > Application settings > Start AuditArch**.

The AuditArch dialog box is as follows:

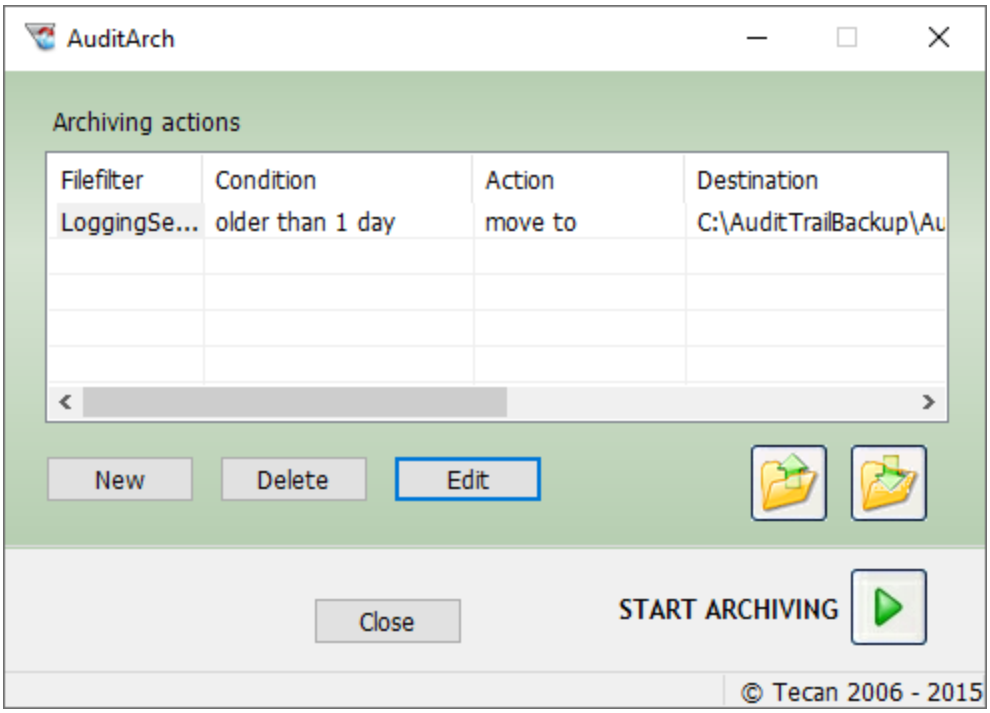


Fig. 376: AuditArch Data Archiving Tool

Click New to create a new archiving action. Click Delete to delete an existing archiving action.
Click the yellow buttons on the right to:

- Save the current set of archiving actions in a .aal file, and
- Open a previously stored set of archiving actions.

Click the large green arrow on the right to start the archiving procedure manually using the currently configured archiving actions.
Click **Edit** to configure the selected archiving action. This opens the Edit Archiving Action dialog box:

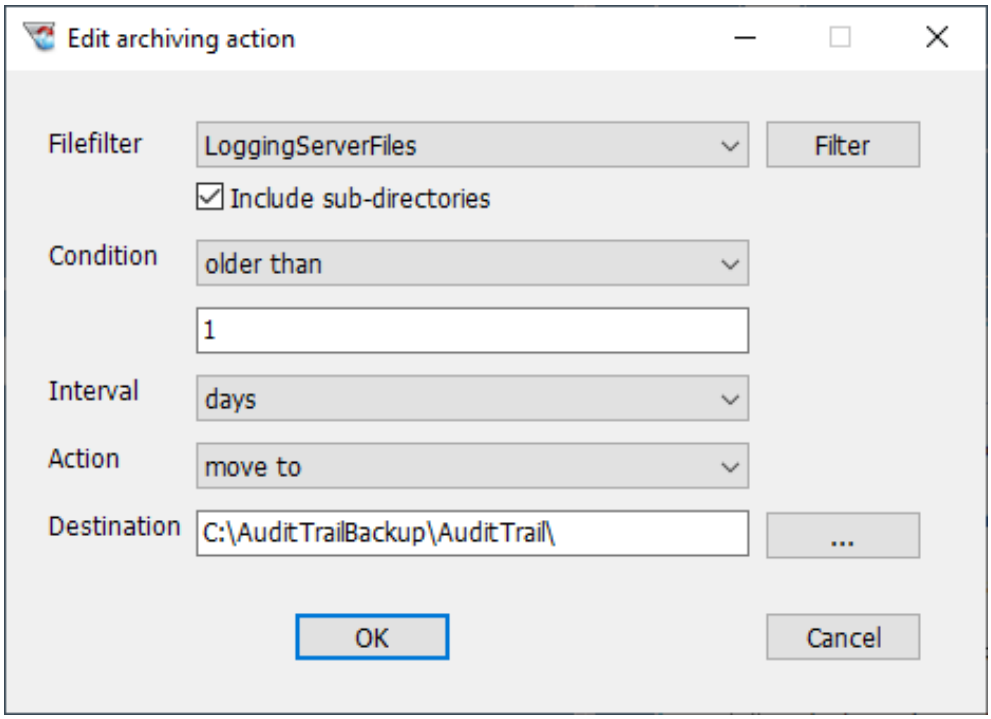


Fig. 377: Editing an archiving action

Choose the files you want to archive with this archiving action by choosing a file filter from the FileFilter pull-down list. Choose All files to archive all log files and audit trail data. Click Filter to create or edit a file name filter in the Edit Filter dialog box (see below).

Check the Include sub-directories checkbox if you want the file name filter to apply to log files in sub-directories of the ../AuditTrail directory as well. You can also choose when you want to archive the files (e.g. when they have reached a specified age) and the action to take (move or copy them to the specified destination directory or delete them).

If you are required to comply with the FDA regulations, make sure you keep archive copies of all audit trail and log files. Do not delete them from the ../AuditTrail directory without making an archive copy, which should be kept as a permanent record.

Edit Filter dialog box

The Edit Filter dialog box lets you specify the files to archive using wildcards:

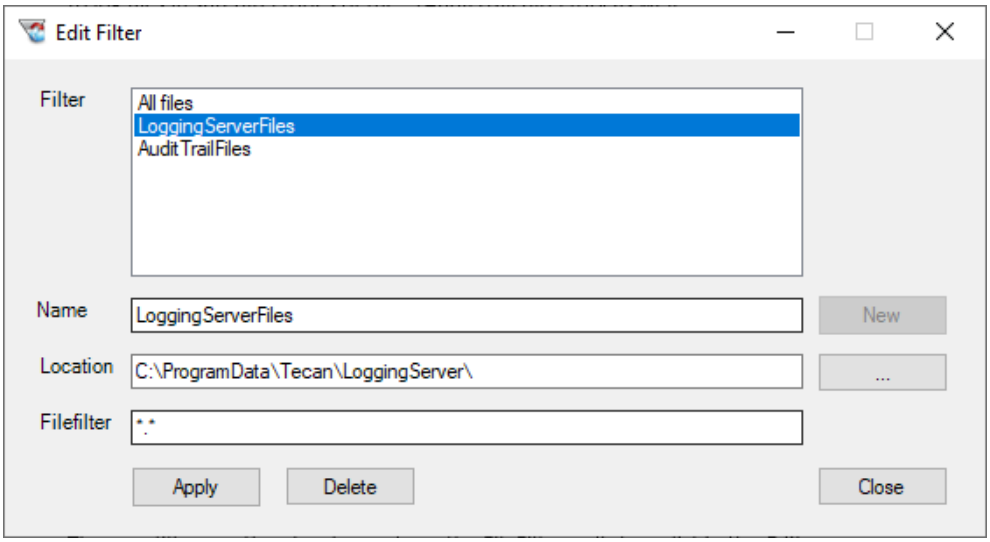


Fig. 378: Edit Filter dialog box

To create a new filter, enter a name for the filter in the Name field, click New, specify the filter settings and click Apply. Click Close when you are finished. The new filter can then be chosen from the FileFilter pull-down list in the **Edit archiving action** dialog box.

14 Error Handling and Recovery

This chapter describes runtime error messages and the associated dialog boxes that can be issued by the software. It provides information on the cause of the error and instructs how to correct the error, where applicable.

14.1 Method Recovery

In FluentControl you can recover runs that have been interrupted, aborted, or had fatal errors.

Not all runs can be recovered.

- Only methods created with and run from the method editor can be recovered.
- Methods created from the script editor with the option “prepare method” cannot be recovered.

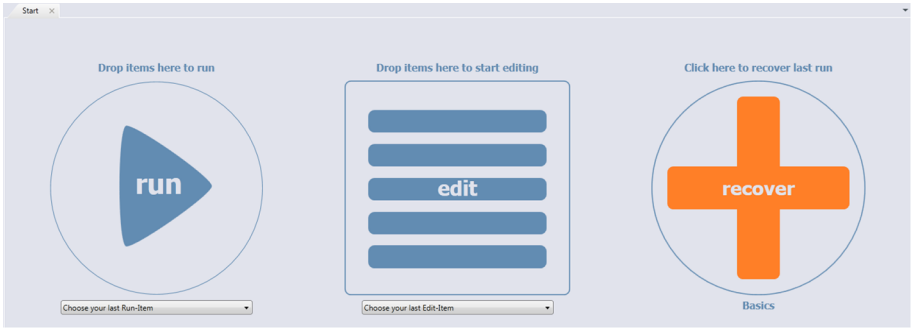
Before trying to recover, executing the following is allowed:

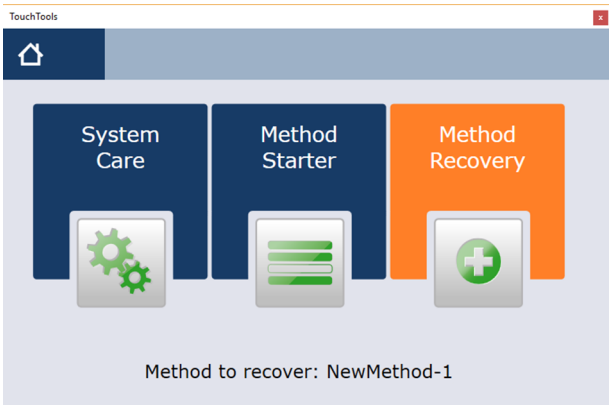
- System Care / Maintenance methods
- Direct Commands
- Scripts & Partial Scripts
- Other methods (without recovery mode)

  ⇨ Making alterations to the worktable definition or to script commands that have already been executed may result in the method no longer being recoverable.

To recover a method proceed as follows:

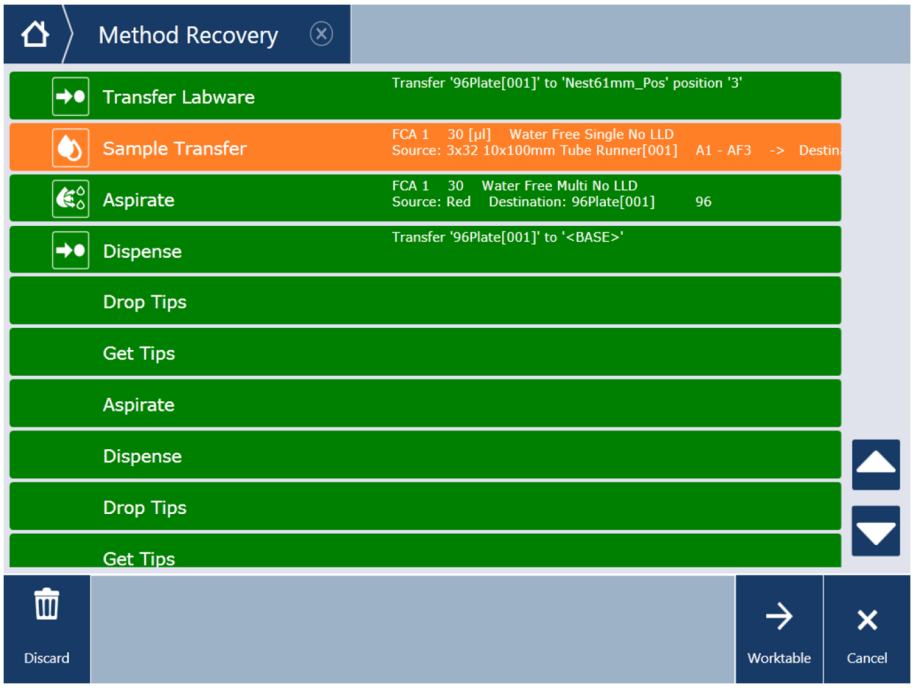
1. Click **recover** on the Start page or **Method Recovery** in TouchTools.



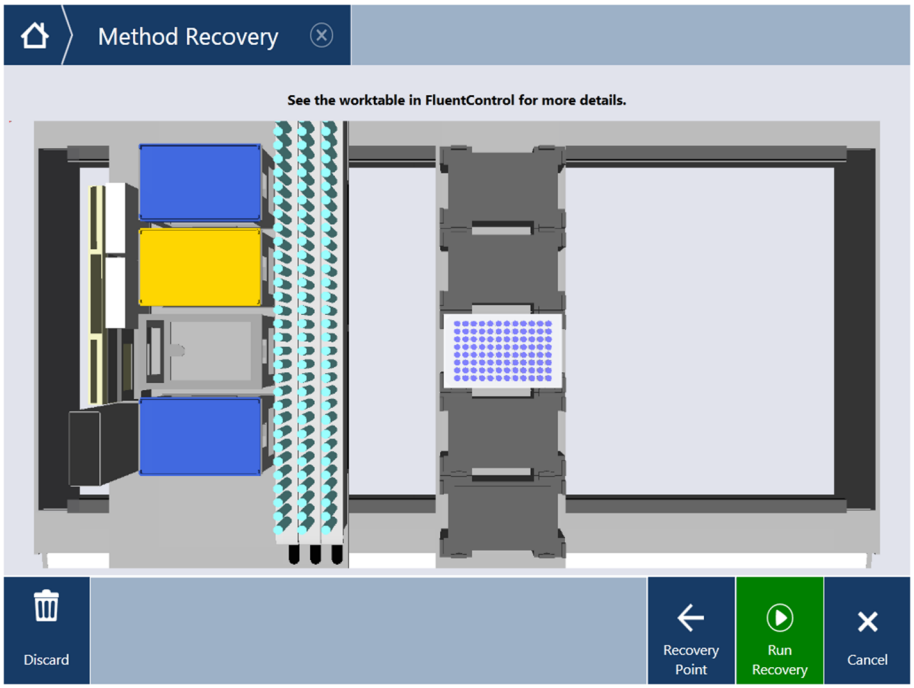


The specific line to be recovered from is highlighted in orange.

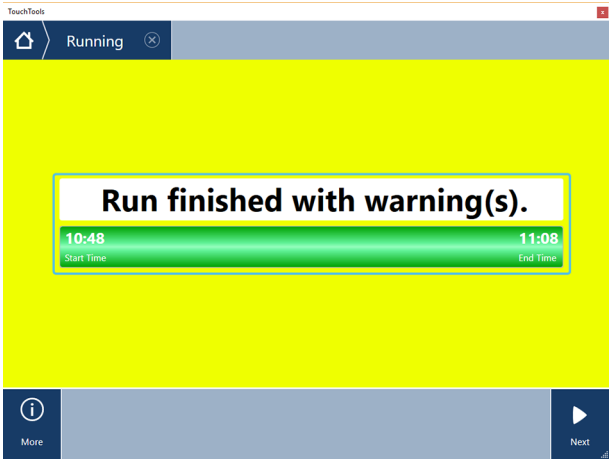
2. Click **Worktable** in TouchTools.



3. Compare the physical worktable with the representation in TouchTools, and ensure that they match.



4. Click **Run Recovery**.

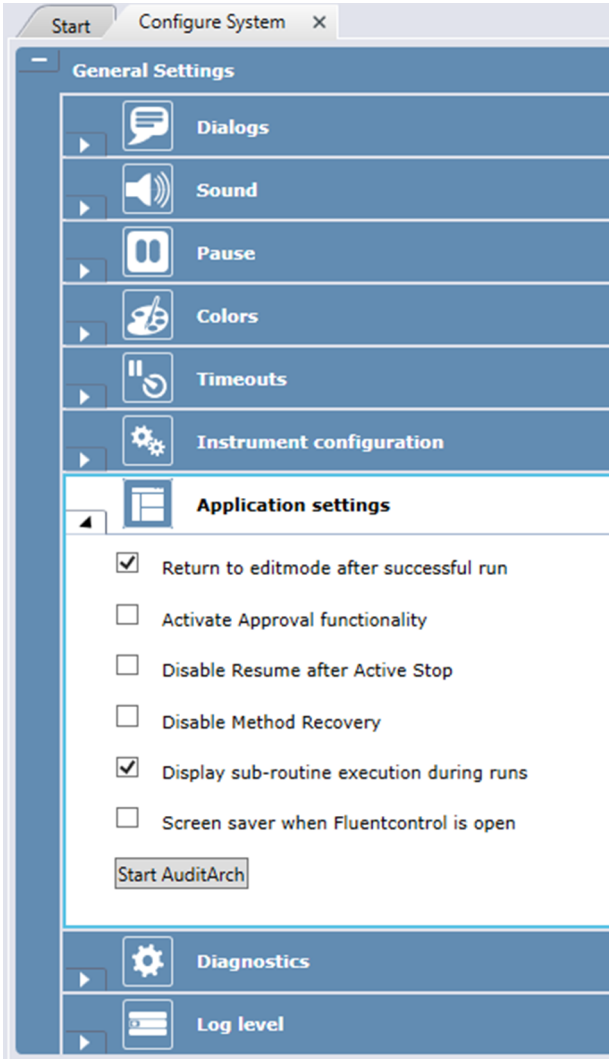


All runs that have been recovered will finish with warnings. Any pipetting steps that were interrupted and recovered will be flagged in Sample Tracking, as it is likely pipetting performance was negatively impacted for this transfer.

14.1.1 Activating and Deactivating Method Recovery

To (de-)activate Method Recovery proceed as follows:

1. Click **Settings > Configure System...**
2. Click **Application settings**.
3. (De-)select **Disable Method Recovery**.



14.1.2 Send Email on Error

FluentControl can send email notifications on errors using Simple Mail Transfer Protocol (SMTP). The email contains the instrument configuration name, time, error identifier. The script name is optional.The email notifications are sent for each error occurrence when the feature is enabled. This behavior does not depend on the error handling (e.g.: ignore and continue, automatic retry and etc.).

Settings

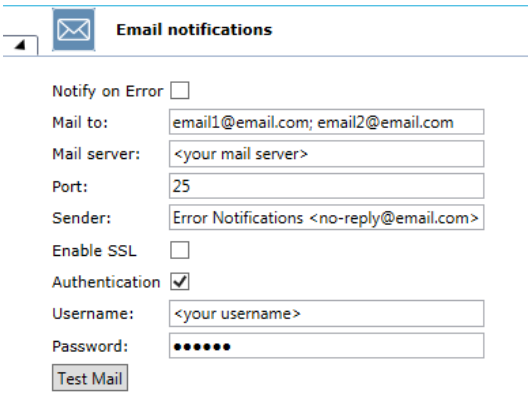
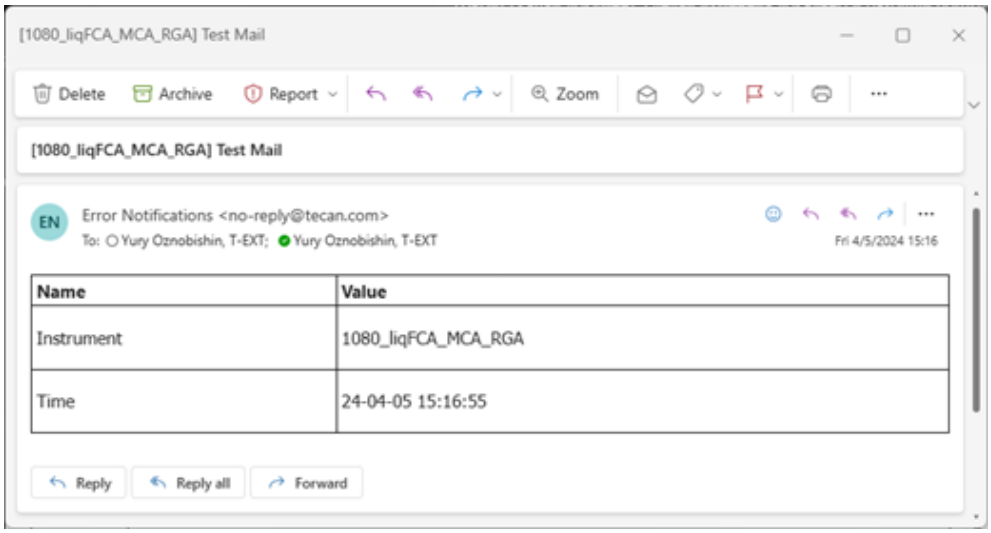


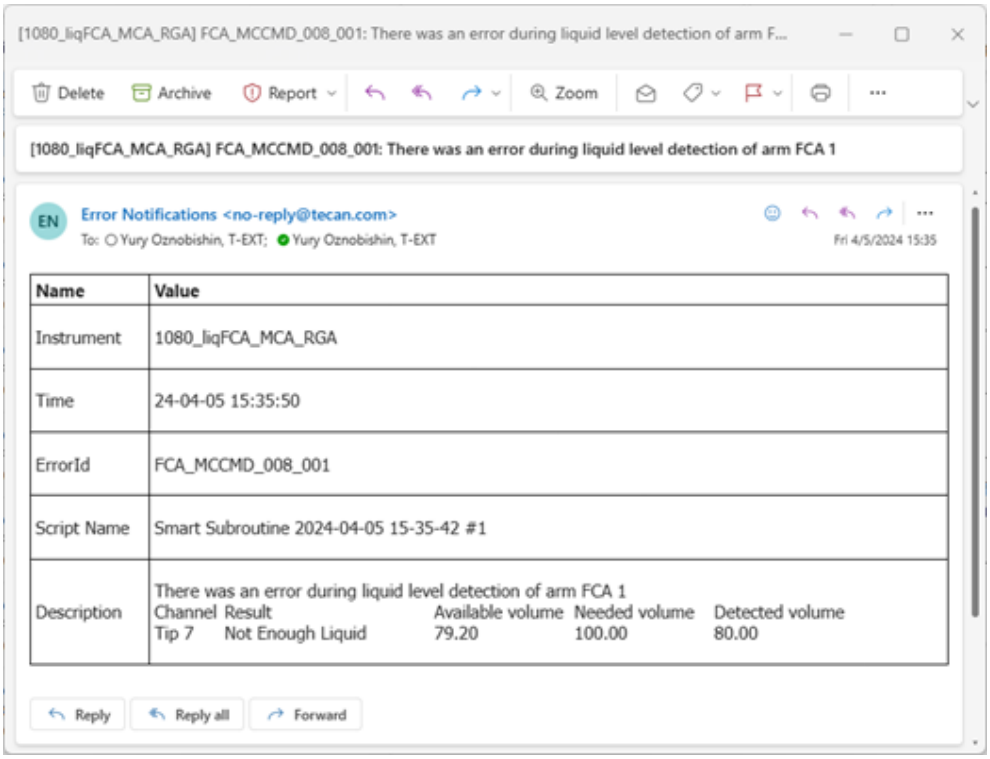
Fig. 379: Email Notification

To configure email notifications, proceed as follows:

1. Click Settings > Configure System.
2. Expand Email notifications section.
3. Change settings.

Examples of Emails





[1080_liqFCA_MCA_RGA] FCA_MCCMD_008_001: There was an error during liquid level detection of arm F...

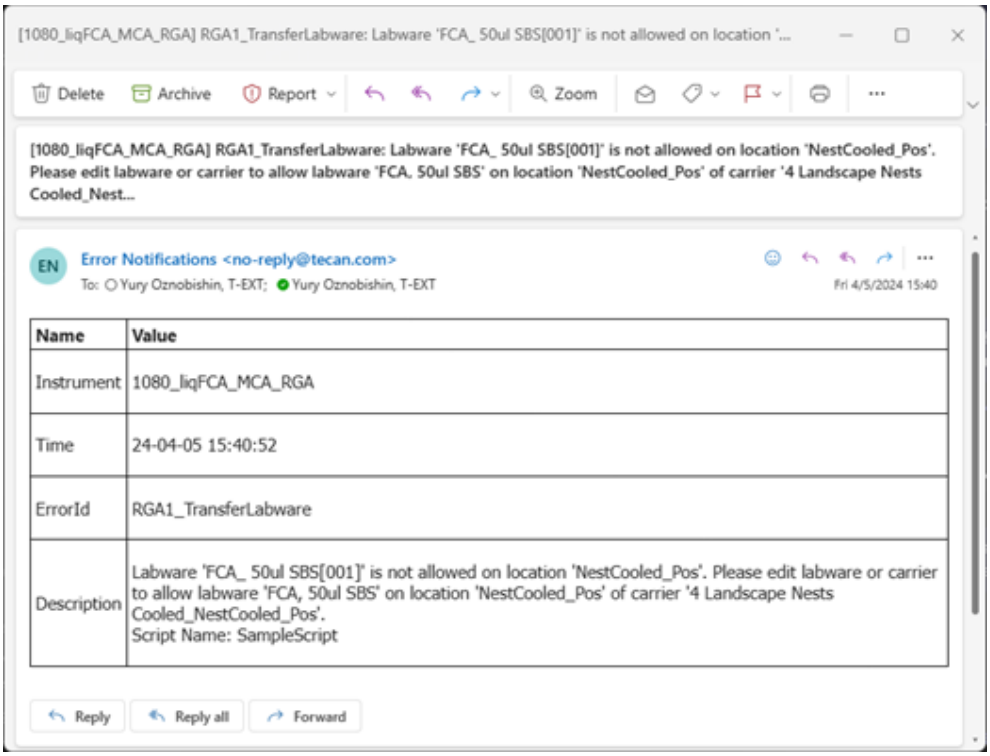
Delete Archive Report Zoom

[1080_liqFCA_MCA_RGA] FCA_MCCMD_008_001: There was an error during liquid level detection of arm FCA 1

EN Error Notifications <no-reply@tecan.com>
To: Yury Oznobishin, T-EXT; Yury Oznobishin, T-EXT
Fri 4/5/2024 15:35

Name	Value
Instrument	1080_liqFCA_MCA_RGA
Time	24-04-05 15:35:50
ErrorId	FCA_MCCMD_008_001
Script Name	Smart Subroutine 2024-04-05 15-35-42 #1
Description	There was an error during liquid level detection of arm FCA 1 Channel Result Available volume Needed volume Detected volume Tip 7 Not Enough Liquid 79.20 100.00 80.00

Reply Reply all Forward



[1080_liqFCA_MCA_RGA] RGA1_TransferLabware: Labware 'FCA_ 50ul SBS[001]' is not allowed on location '...

Delete Archive Report Zoom

[1080_liqFCA_MCA_RGA] RGA1_TransferLabware: Labware 'FCA_ 50ul SBS[001]' is not allowed on location 'NestCooled_Pos'. Please edit labware or carrier to allow labware 'FCA, 50ul SBS' on location 'NestCooled_Pos' of carrier '4 Landscape Nests Cooled_Nest...

EN Error Notifications <no-reply@tecan.com>
To: Yury Oznobishin, T-EXT; Yury Oznobishin, T-EXT
Fri 4/5/2024 15:40

Name	Value
Instrument	1080_liqFCA_MCA_RGA
Time	24-04-05 15:40:52
ErrorId	RGA1_TransferLabware
Description	Labware 'FCA_ 50ul SBS[001]' is not allowed on location 'NestCooled_Pos'. Please edit labware or carrier to allow labware 'FCA, 50ul SBS' on location 'NestCooled_Pos' of carrier '4 Landscape Nests Cooled_NestCooled_Pos'. Script Name: SampleScript

Reply Reply all Forward

14.1.3 Method Recovery and Sample Tracking Embedded

An existing method recovery point makes the Sample Tracking Embedded database temporary persistent in order to preserve pipetting data of the aborted run. This feature allows to have complete Sample Tracking data in case of an aborted method that is followed by a successful method recovery.

In case you do not plan to recover the aborted method, please discard the method recovery point. The Sample Tracking Embedded database content will then be cleared again at the next run start as usual.

14.2 Errors Requiring User Interaction (FCA)

This section describes pipetting error messages which require user interaction. Please see the description of the output file format of the Export Data command for information on the pipetting error codes which are reported. Errors are also reported in the application log files.

14.2.1 FCA error: DiTi(s) not dropped

Error received if one or more DiTis were not properly ejected.

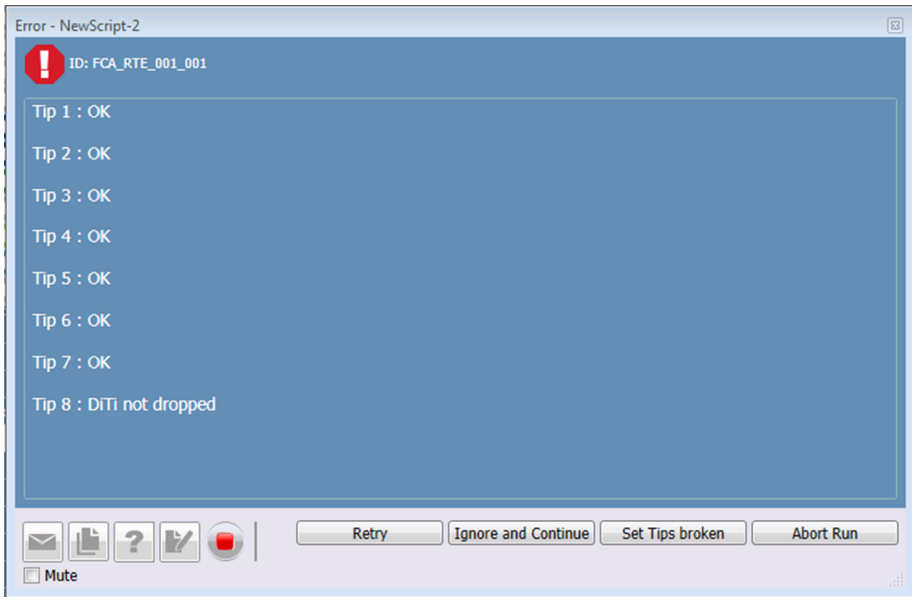


Fig. 380: Error message

When receiving this error message, the user has the following options to handle the error properly:

- **Retry**: The Drop Tips command executes again trying to eject the tips.
- **Ignore and Continue**: FluentControl continues the execution of the script regardless of whether the tips have been properly ejected.
- **Set Tips broken**: Channels with not dropped DiTis are marked as broken and are not used further in the pipetting sequence.
- **Abort Run**: The whole script is aborted.

14.2.2 FCA error: DiTi has been lost

Error received if one or more DiTis were lost during the execution of a script.

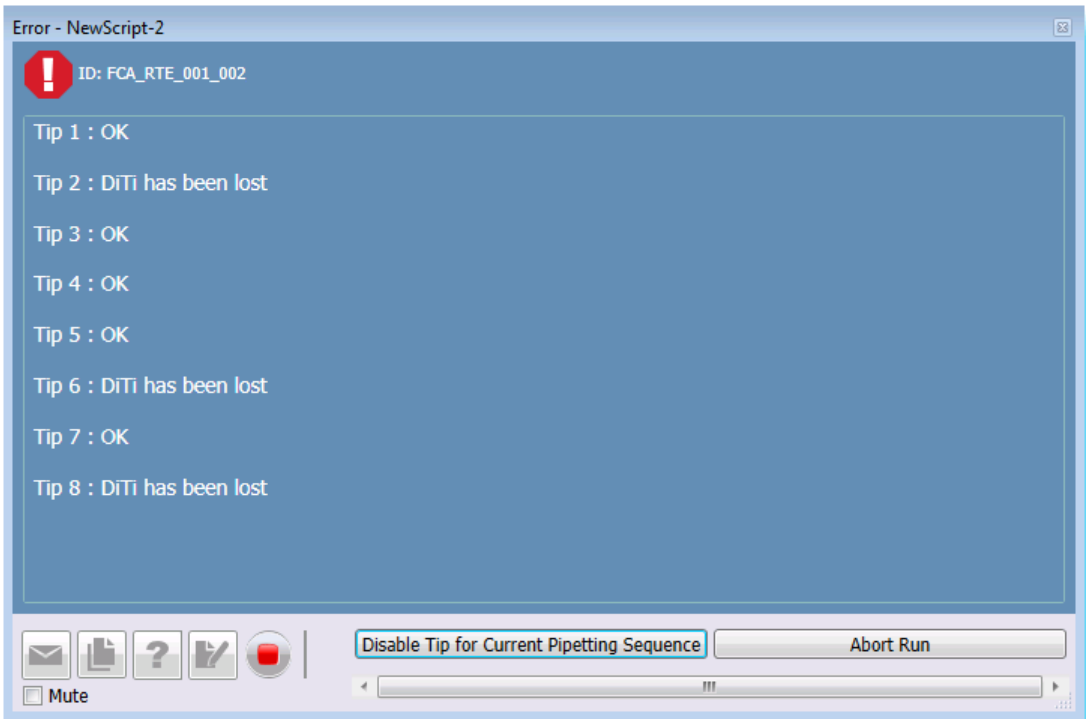


Fig. 381: Error message

When receiving this error message, the user has the following options to handle the error properly:

- **Disable Tip for Current Pipetting Sequence**: Channel(s) with lost DiTis are disabled for the current pipetting sequence and the execution of the script continues.
- **Abort Run**: The whole script is aborted.

14.2.3 FCA error: DiTi(s) not fetched

Error received if one or more DiTis were not properly fetched. The worktable is accessible for the user to fill the DiTi rack with new DiTis or to replace the whole DiTi rack.

When replacing the whole DiTi rack, put the new DiTi rack on exactly the same worktable position.

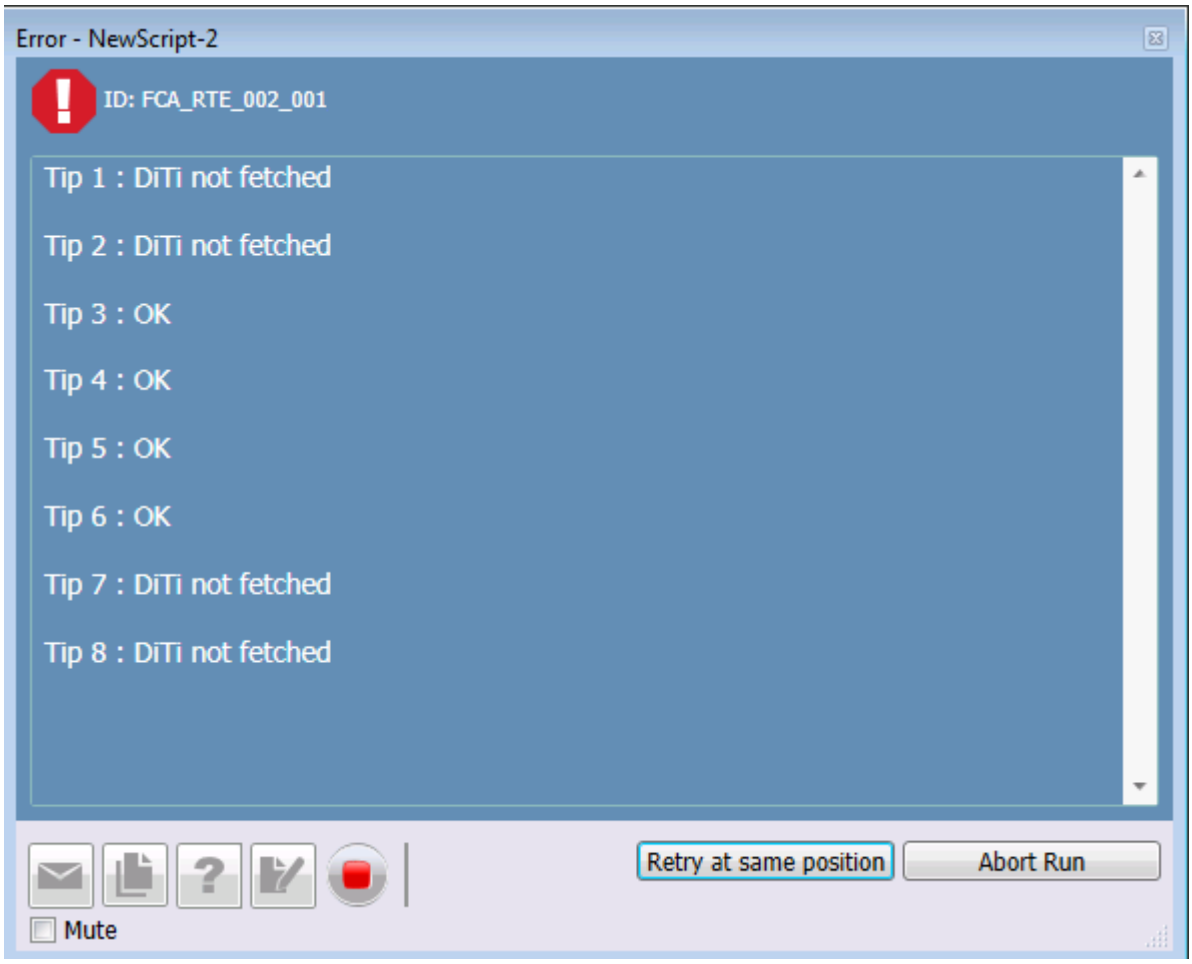


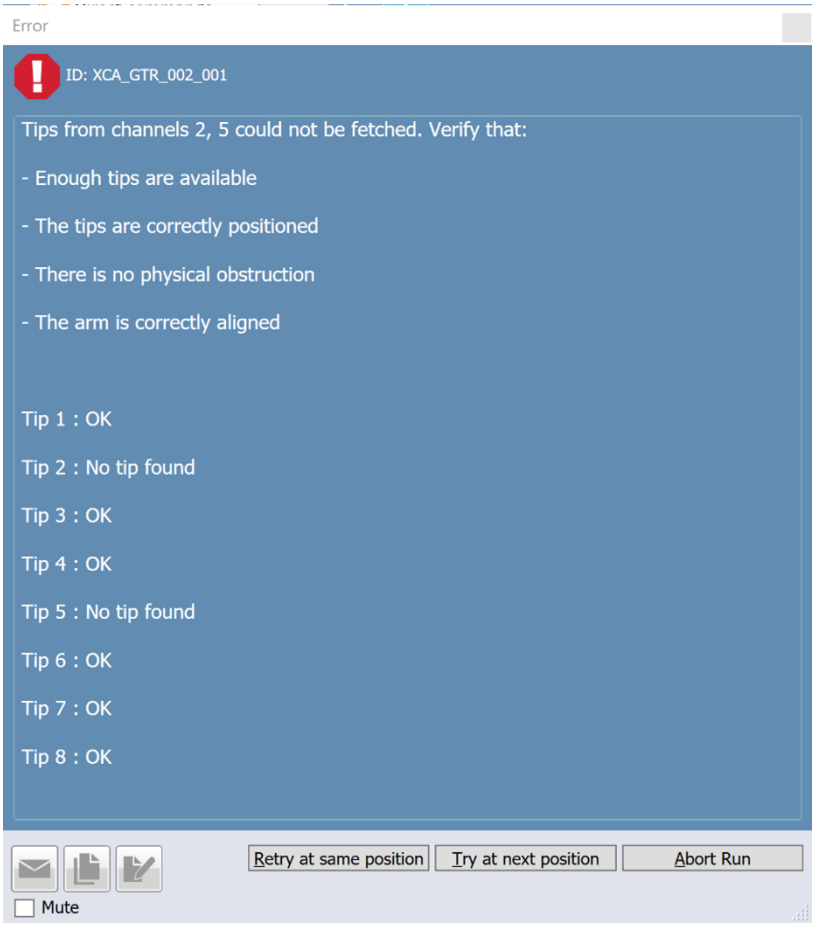
Fig. 382: Error message

When receiving this error message, the user has the following options to handle the error properly:

- **Retry at same position**: The FCA tries to get DiTis again from the same position as before.
- **Abort Run**: The whole script is aborted.

14.2.4 FCA error: DiTi(s) not fetched

This error message is shown if one or more disposable tips were not properly fetched. The worktable is accessible for the user to fill the tip box with new DiTis or to replace the whole box.



The user has the following options to handle the error:

Retry at same position: The FCA tries to get DiTis again from the same position as before.

Try at next position: The FCA moves to the next available position and tries to fetch tips.

Abort Run: The entire run is aborted.

14.3 Error Requiring User Interaction (RGA)

This section describes pipetting and hardware device error messages which require user interaction. Please see the description of the output file format from the Export Data command for information on the pipetting error codes which are reported. Errors are also reported in the FluentControl log files.

- RGA Error: Labware lost
- RGA Error: Labware not gripped on source location
- RGA Error: Object not gripped on source location

14.3.1 RGA error: Labware lost

Error received if the labware being transferred has been lost.

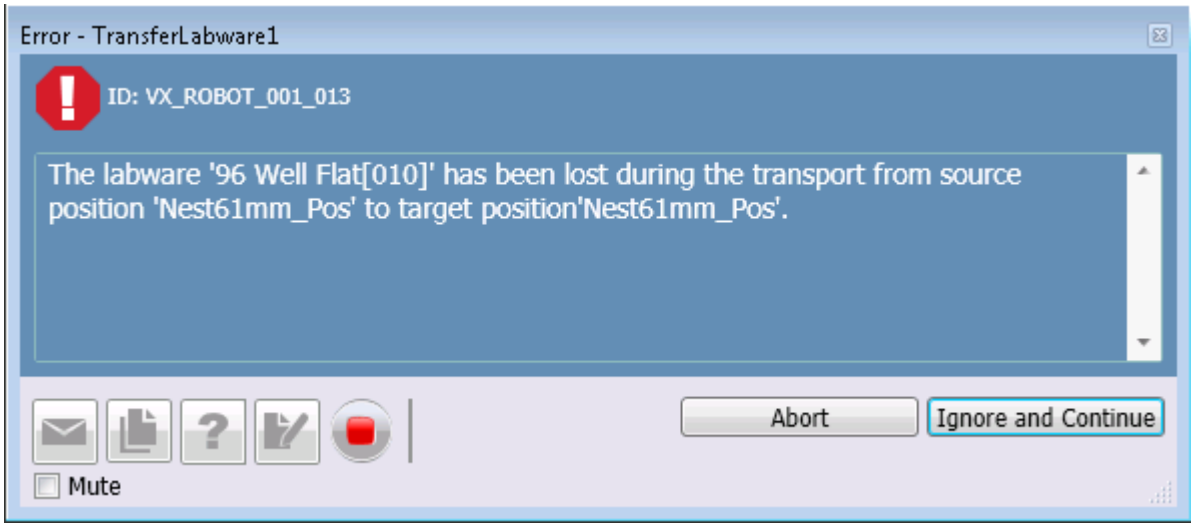


Fig. 383: Error message

When receiving this error message, the user has the following options to handle the error properly:

- **Abort Run**: The whole script is aborted.
- **Ignore and Continue**: FluentControl continues the execution of the script ignoring that the labware was lost.

14.3.2 RGA error: Labware not gripped on source location

Error received if the labware was not successfully gripped at the source location.

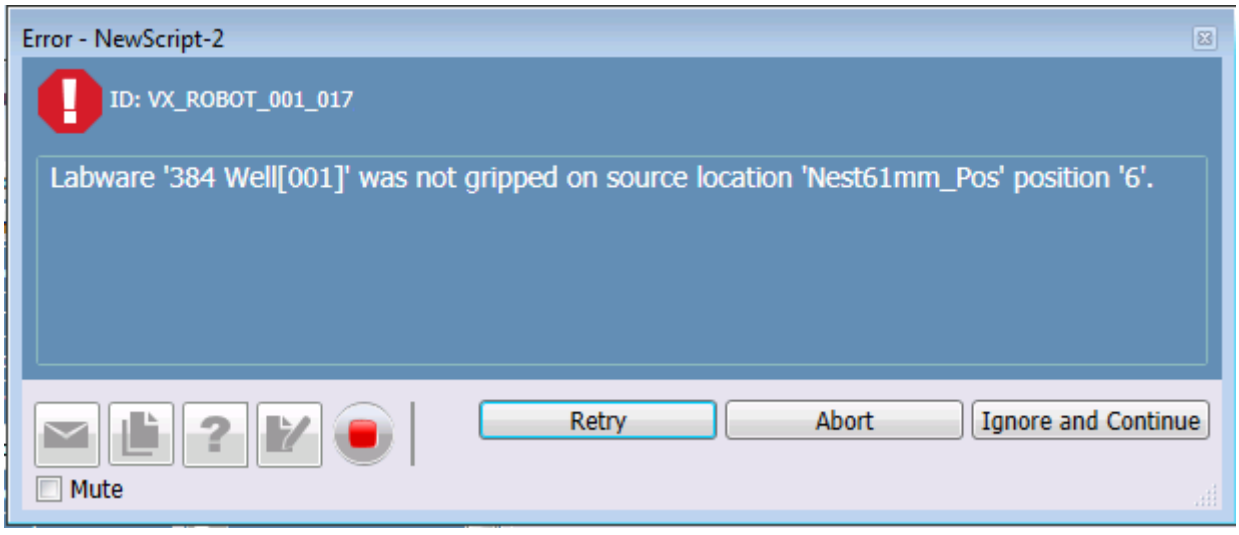


Fig. 384: Error message

At this point, the worktable is accessible to the user. When receiving this error message, the user has the following options to handle the error properly:

*Before clicking **Retry**, the user can access the worktable and place the object on the source position.*

- **Retry**: The RGA tries to grip the labware from the same position as before.
- **Abort**: The whole script is aborted.
- **Ignore and Continue**: FluentControl continues the execution of the script ignoring that the labware was not fetched.

14.3.3 RGA error: Object not gripped on source location

Error received if the object was not successfully gripped at the source location.

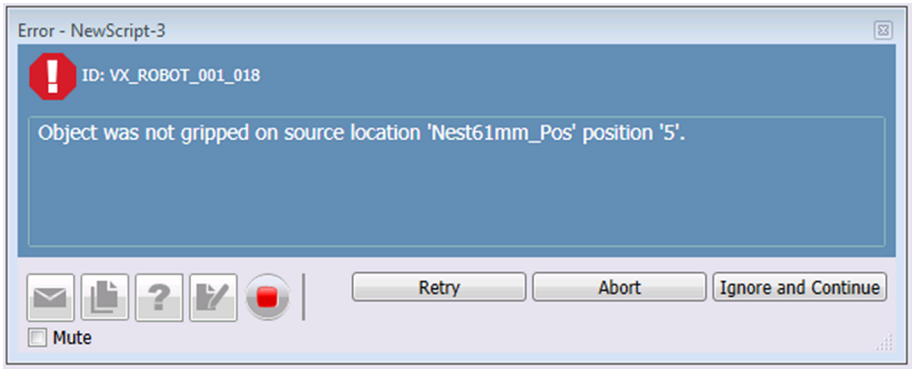


Fig. 385: At this point, the worktable is accessible to the user

When receiving this error message, the user has the following options to handle the error properly:

*Before clicking **Retry**, the user can access the worktable and place the object on the source position.*

- **Retry**: The RGA tries to grip the object from the same position as before.
- **Abort**: The whole script is aborted.
- **Ignore and Continue**: FluentControl continues the execution of the script ignoring that the labware not fetched.

14.4 Error Requiring User Interaction (Fluent ID)

This section describes pipetting and hardware device error messages that require user interaction. Please see the description of the output file format from the Export Data command for information on the pipetting error codes which are reported. Errors are also reported in the FluentControl log files.

14.4.1 Unreadable Tube Barcode

When receiving this error message, the user has the following options to handle the error properly:

- Check barcode label and position on tube.
- Reload with slower move.
- Check allowed barcode configuration in settings.
- Clean window of the scanner.

14.4.2 Unreadable Runner Barcode

When receiving this error message, the user has the following options to handle the error properly:

- Check runner barcode.
- Check runner type.

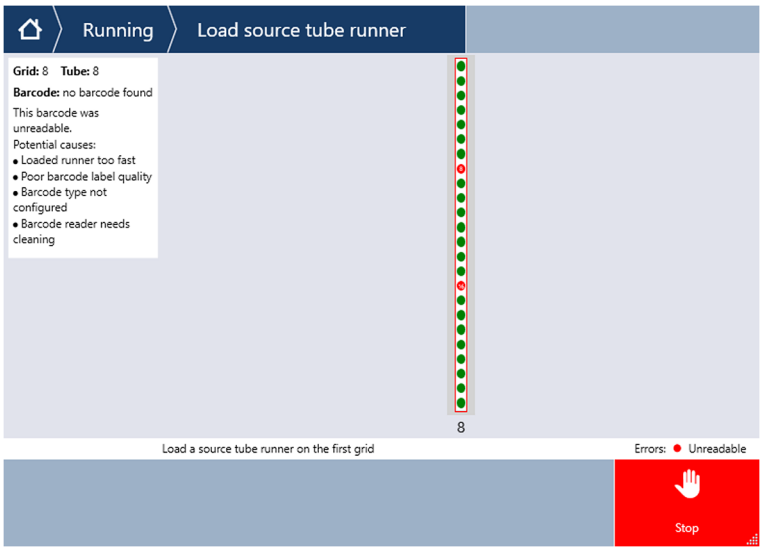


Fig. 386: No barcode found error

14.4.3 Runner Not Detected (Hall Sensor)

When receiving this error message, the user has the following options to handle the error properly:

- Check runner type (Fluent ID runner).

14.4.4 Runner Not Defined in Worktable

When receiving this error message, the user has the following options to handle the error properly:

- Loaded wrong runner type according to worktable, not defined runner, on wrong position.

14.4.5 Continue Not Allowed after Loading

When receiving this error message, the user has the following options to handle the error properly:

- Check definition in the **Start Loading** statement.

14.5 Detection Sequence

If the detection sequence is wrong, the system does not accept the tube runner and informs the user.

The software only accepts barcodes from the firmware if:

- Barcode could be read
- Checksum is correct
- Barcode type is allowed
- Detection sequence is correct (checked by firmware, no software needed):
  - Runner Start barcode read
  - Tube Barcodes read
  - Runner Endbarcode read
  - Hall Sensor detects contact

14.6 Error Requiring User Interaction (i300)

14.6.1 Bubble Detected Event Handling

If a bubble is detected during a dispense action in an active run, the system will prompt a Bubble Detected Error. At this point, the user has two options: **Ignore and Continue** or **Abort**.

Ignore and Continue:
Selecting this option will ignore the detected bubble. The run will continue running and will finish with a warning.

Abort:
Selecting this option will abort the current run.

In case the Bubble Detected Error occurs during a Method run, it is also possible to use the Method Recovery feature after aborting the run. The user must ensure a safe operating state before resuming. The following steps are recommended.

1. Resolve the root cause of the bubble.
2. Prime the dispenser with the appropriate solvent using the Wash command via Direct Commands.
3. The recovery will resume with the dispense action from the first selected well. If there is a risk of overflowing wells, it is recommended to empty the filter plate using the Apply Pressure command via Direct Commands.



Depending on the application step, the liquid will be discarded or collected. Before executing the Apply Pressure Direct Command, the user must adapt the worktable layout in the Worktable View to reflect the correct plate setup and destination (waste or collection plate).

4. Before the method recovery, ensure that the physical worktable reflects the Worktable View at the corresponding step when the error occurred.
5. Use the method recovery, which restarts the method run with the Dispense command beginning at the first selected well.

15 API

The FluentControl API allows external applications and scripts to control FluentControl. It allows to run methods, read/write variables, stop/pause/resume a run, trigger execution of script commands and much more.

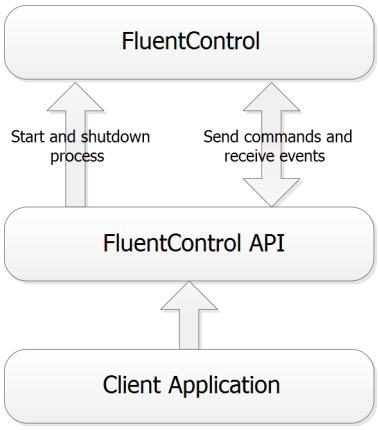


Fig. 387: Overview

Why to use the API?

FluentControl's main feature is to automate assay execution. However, to start and configure an assay run the user still needs to interact with the application:

1. Start a method.
2. Enter parameters (e.g., sample count, barcodes, etc.).
3. Monitor progress and react on errors.

The API allows for automating these steps with external applications and scripts.

Example

- Connect to a database or other external resources
  An external application can read data from a database or from a web service and dynamically start a FluentControl method and parametrize the run.

15.1 Execution Channels

Execution Channels allow for triggering script command executions in FluentControl from an API client.

In FluentControl a **script with worktable** and a **Create API Channel** script command is required for creating an Execution Channel.

When **Create API Channel** is executed, a new subroutine is started. The event **ChannelOpens** assigns an Execution Channel instance that can now receive script commands via API.

The API client can now use the Execution Channel to send commands and to trigger **Abort** or **Finish** of the Execution Channel.

- **Abort**: The complete run will abort.
- **Finish**: Only the selected Execution Channel ends. Other Execution Channel can still send commands and the script execution can finish successfully.

To achieve parallel execution multiple Execution Channels need to be created with multiple **Create API Channel** script commands using a unique name for each Execution Channel.

15.2 Functions

The functionality available in the interfaces shown in the diagram is:
FluentControl provides functions to start/close/login to FluentControl.
IRuntimecontroller allows to list, start, stop and interact with methods.
Both of these also send events.

ICedInfo allows to view and react to errors&dialogs.

IExecutionChannel allows to open parallel interfaces for interaction while a method is running.

RuntirneController
+ CloseMethod(): void
+ CloseQueryAtStartUpDialog(bod): void
+ GetAllMaintenanceMethods(): string[]
+ GetAllRunnableMethods(): string[]
+ GetAlllUserNames(): string[]
+ GetCurrentUserName(): string
+ GetFluentStatus(): StateMachineStates
+ GetProgress(): int
+ GetprogressInitialization(): int
+ GetQueryVariableNames(): string[]
+ GetStateAsString(StateMachineStates): string
+ GetVariableNames(): string[]
+ GetVariableValue(string): string
+ InitializeInstrument(WorkspaceSelection, string): void
+ InteriorLight(bool): void
+ IsReady(): bool
+ LoginUser(string, string): bool
+ PauseRun(): void
+ PrepareMethod(string): void
+ ResolveExpression(string): string
+ ResumeRun(): void
+ RunMethod(): void
+ SetVariableValue(string, string): void
+ StopMethod(): void
+ ValidateUser(string,string): bool
<<event>>
+ ChannelCloses(): ChannelChange
+ ChannelOpens(): ChannelChange
+ CommonErrorDialog(): CEDNotification
+ EnterReadyMode(): Acation
+ Error(): Notification
+ ModeChanged(): ModeSwitch
+ ProgressChanged(): Progress
+ DeckCheckDiscrepancyDetected(): DeckCheckDiscrepancyEventArgs

NOTICE

SetVariableValue(string,string): void

Valid only for variables of scope Run. They can be set at runtime only.

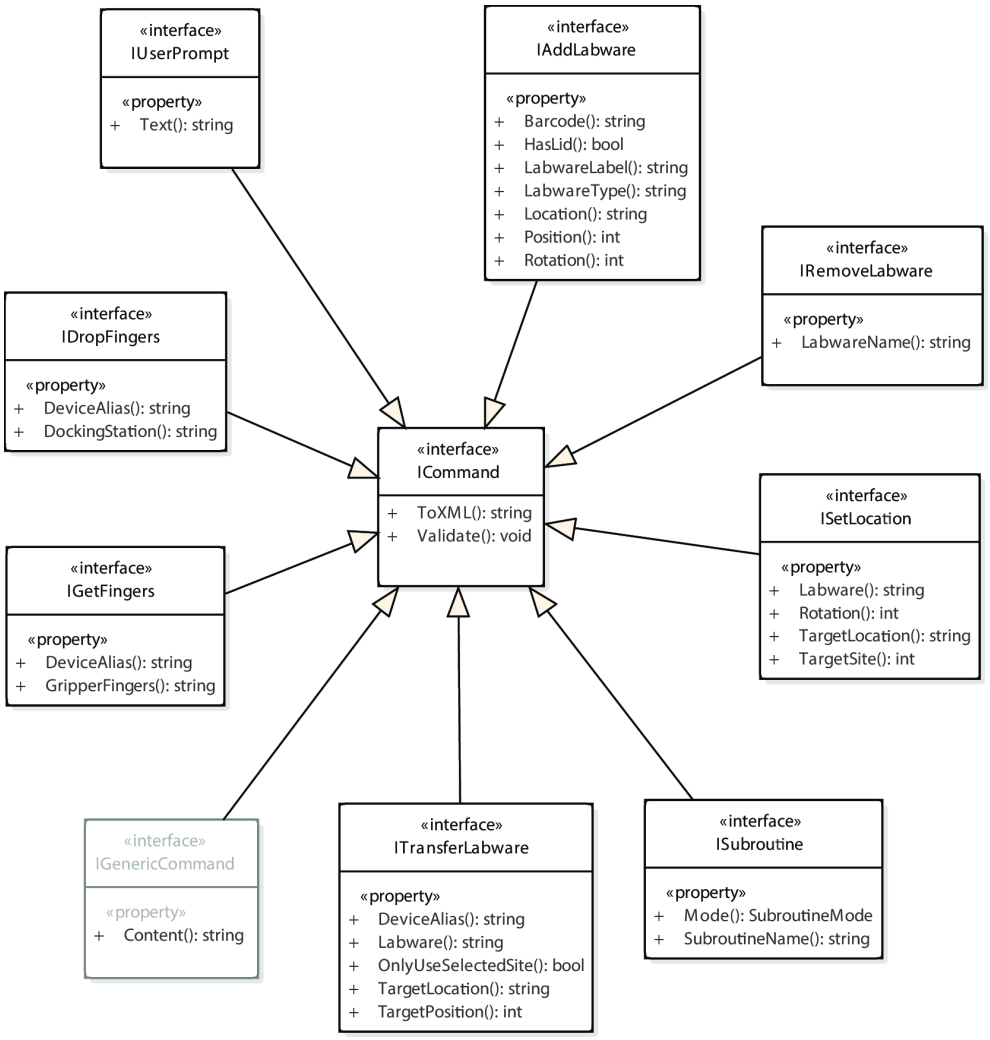


Fig. 388: Commands

*The **IGenericCommand** is meant as a supplement for actions that need to be used but were not foreseen in the API (e.g., to set temperatures of devices controlled through third-party drivers or similar). Since the command is very generic is not possible to test in detail what will and will not work. The XML input string is not documented and can change between software versions—i.e., a generic command that worked with a specific FluentControl version may not work with a more current one.*

*The function **ReadyToConnect** is deprecated and should not be used anymore. Please use **RuntimeIsAvailable** to determine when to request a runtime. See example Start FluentControl and use API for details.*

15.3 Examples

15.3.1 Start FluentControl and use API

The API can be used to start FluentControl. During the first seconds of the startup process the API cannot do anything with FluentControl. As soon as FluentControl has loaded and prepared enough data, the following event will be triggered: `RuntimeIsAvailable`.

When this event is triggered the API user can call all available methods of the **RuntimeController**. The **RuntimeController** provides access to the main functionalities of the API to control a running FluentControl process. Some of the functions depend on certain states (e.g., `Prepare Method`—which only works in **EditMode**).

C#

```
private static ManualResetEventSlim
_waitForRuntimeIsReady=new ManualResetEventSlim(false);
public static void StartFluentControl()
{
     FluentControl process = new FluentControl();
     bool alreadyStarted = process.IsRunning();
     process.RuntimeIsAvailable +=
_process_RuntimeIsAvailable;
     process.StartOrAttach();

     if( !alreadyStarted )
     {
       _waitForRuntimeIsReady.Wait();
     }
     RuntimeController runtime =
(RuntimeController)process.GetRuntime();
}

private static void _process_RuntimeIsAvailable()
{
     _waitForRuntimeIsReady.Set();
}
```

VBA

```
Dim WithEvents vision_x As FluentControl

Dim WithEvents run As RuntimeController

Public Sub StartFluentControl()
     Set vision_x = New FluentControl
     vision_x.StartOrAttach
End Sub

' Event Handler
Public Sub vision_x_RuntimeIsAvailable()
     Set run = vision_x.GetRuntime
End Sub
```

15.3.2 Prepare a Method Run

Prerequisites for preparing a method for execution with API:

- RuntimeController instance is available
- Otherwise the example code will wait until EditMode is reached.

C#

```
private static readonly ManualResetEventSlim _waitForEditMode
= new ManualResetEventSlim( false );

RuntimeController runtime =
(RuntimeController)_process.GetRuntime();
runtime.ModeChanged += Runtime_ModeChanged;
if (_runtime.GetFluentStatus() !=
StateMachineStates.EditMode)
{
     _waitForEditMode.Wait();
}
runtime.PrepareMethod("Method to prepare");

private static void Runtime_ModeChanged( StateMachineStates
old, StateMachineStates current )
{
     if (current == StateMachineStates.EditMode)
     {
      _waitForEditMode.Set();
     }
}
```

15.3.3 Get ExecutionChannel and Execute Command on Channel

Prerequisites for passing script commands from the API to FluentControl:

- A script with a **Create Channel** script command

The API Client needs to attach to following **EventHandler** to get an Instance of **IExecutionChannel**:

```
runtime.ChannelOpens += Runtime_ChannelOpens;
```

As soon as the script run is executing the **Create Channel** script command the event handler will be triggered and the API client is now able to send script commands via the **IExecutionChannel** to FluentControl:

```
private static void Runtime_ChannelOpens( IExecutionChannel
openedChannel )
{
     openedChannel.ExecuteCommand(new
TransferLabware("labware","targetLocation",2,true));
}
```

15.3.4 Asynchronous Subroutines

Once you opened an API-Channel you might want to start another routine that runs in parallel to your main script.

To do so please create a new API-Channel via the already opened execution channel and after receiving the new generated channel by the **ChannelOpen** Event Handler you can execute the second subroutine there. Subroutines that are started within the Execution Channel of the main script will be run immediately and will block the API-Client Call of **ExecuteCommand** until the complete subroutine is executed.



The execution of asynchronous subroutines or the creation of an API channel from within another API channel is not supported. For multiple API channels, a script with multiple Create Channel script commands must be used

```
class Program
{
     private static readonly ManualResetEvent _waitForFinish
= new ManualResetEvent(false);
     private static readonly SubroutineMode subroutineMode =
SubroutineMode.StartSubroutineAndWaitForItToFinish;
     static void Main(string[] args)
     {
      FluentControl fluent = new FluentControl();
      Console.WriteLine("Created Fluent instance");
      fluent.StartOrAttach();
      Console.WriteLine("Attached to Fluent instance");
      RuntimeController runtime =
(RuntimeController)fluent.GetRuntime();
      Console.WriteLine("Created runtime instance");
      runtime.ChannelOpens += Runtime_ChannelOpens;
      Console.WriteLine("Attached Runtime Controller");
      _waitForFinish.WaitOne();
     }

     private static void
Runtime_ChannelOpens(IExecutionChannel openedChannel)
     {
      If(openendChannel.Name.Equals(”secondChannel”))
            //Things you want to do inside the second API-
Execution Channel openedChannel.ExecuteCommand(new
Subroutine(…));
      }
      Else
      {
            // Main Channel used to start secondChannel:
            StartNewChannel(openendChannel);
            // Main Channel can now be used for other things
      }
     }

     private static void StartNewChannel (IExecutionChannel
```

```
currentChannel)
     {
      string text = "<ScriptGroup>
            < Objects >
            < Object Type =
\"Tecan.Core.Scripting.Statements.CreateChannelStatement\" >
            <CreateChannelStatement >
            <ApiChannelName> secondChannel </ApiChannelName>
            <IsBreakpoint> False </IsBreakpoint>
            <IsDisabledForExecution> False </
IsDisabledForExecution>
            <LineNumber> 1 </LineNumber>
            </CreateChannelStatement></ Object >
            </ScriptGroup>"
      openedChannel.ExecuteCommand(new GenericCommand(text));
     }
}
```

15.3.5 Resolving Expression from PowerShell command line

When FluentControl is running, the API can be used to get information from the current run. API does not support expressions for script or worktable specific functions.

PowerShell script

```
$fluent = New-Object -COM "Tecan.FluentControl"
$fluent.StartOrAttach()
$runtime=$fluent.GetRuntime()
$iteration=$runtime.ResolveExpression("loop_iteration")
$volumeWellOne=$runtime.ResolveExpression("GetAttribute(""tar
getLabware"",""Well[1].Volume"")")
$volumeWellTwo=$runtime.ResolveExpression("GetAttribute(""tar
getLabware"",""Well[2].Volume"")")
$status = @("Iteration: ", $iteration,"Volume 1: ",
$volumeWellOne,"Volume 2", $volumeWellTwo)
$status | format-table
```

15.3.6 Close FluentControl

The API can be used to close FluentControl. FluentControl needs to be in **EditMode** to be able to properly shutdown via API function **Shutdown**. It has a parameter timeout which defines how many milliseconds to wait for the shutdown to be finished. In case the SW is unresponsive API function **ForceShutdown** can be used. It works same than API function **Shutdown** but will kill the FluentControl process when the process is still running after the timeout.

C#

```
private static FluentControl _process;
public static void CloseFluentControl()
{
     if(!_process.Shutdown(20000))
     {
       _process.ForceShutdown(20000);
     }
}
```

16 Scheduler

The FluentControlScheduler requires an additional license to activate the features described in this chapter.

16.1 The Process Editor

A Process is a set of connected steps. It can be executed as a part of a Method together with FluentControl standard Scripts.

16.1.1 Creating a Worktable

For setting up processes it is important to create a worktable beforehand. By drag-and-drop all carriers and labware (movable and static) can be placed on the worktable. It is the same procedure as known from scripting in FluentControl. Labware that will be moveable during a process such as assay microplates are typically placed in storage carriers. For some of those storage carriers like hotels or carousel towers, FluentControl opens the Labware Group Editor tab. Drag-and-drop the labware into the list below the graphic, assign a labware group name and select at which carrier sites the labware of the selected group is generated. It is recommended to add all fixed Worktable elements like carriers, devices to the Base Worktable. Elements associated with an individual protocol can be added via the Process Worktable.

16.1.2 Open a Process

Navigate to the Control Bar item **Processes** and open a new process by double-click on **New…**. Under **Process properties** a process name can be assigned and a worktable from the drop-down list can be linked to the process.

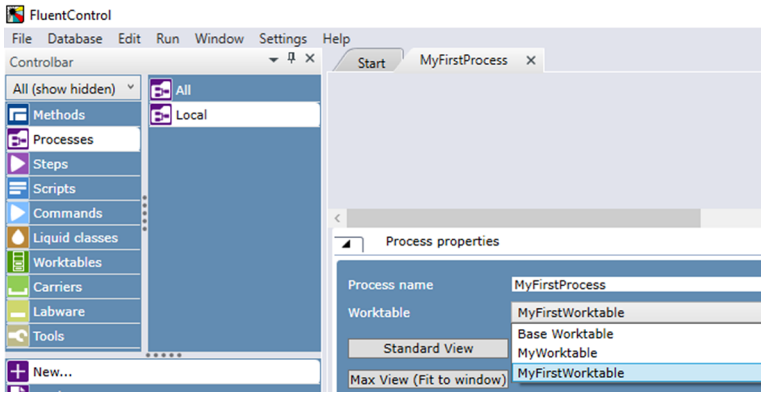


Fig. 389: Open a Process

16.1.3 Process Worktable

It is recommended to add source Labware Groups in the process editor worktable. The added Labware will not be shown in the processing scripts editor worktable, which allows you to use the storage carriers e.g. Hotels for Base Incubation even though it is fully occupied.

16.1.4 Process Steps

For creating a process three different process steps are available: Processing, Source and Destination step.
By drag-and-drop the steps can be added to the process and labware sequences can be created.



Fig. 390: Process Steps

16.1.4.1 Source Step

The Source step is the first step of a labware sequence that defines the labware group which is used in a process sequence. The name of the labware group can be added in the text field of the Source step. It is also possible to use variables in the labware name field. The Source steps can be connected to Processing Steps.

16.1.4.2 Processing Step

A Processing step is a generic node that contains a script (equivalent to FluentControl Script Editor) when you double-click on it. This script defines which action takes place (pipetting with FCA or MCA, Incubation, etc.). The Processing step is set after a Source step or another Processing step.

A descriptive Alias can be given to its text field. A Processing Step by itself is not pre-assigned to a certain driver or resource. The software does automatically figure out which driver's script commands are used within the step script and dynamically change the color of the node according to the assigned driver after the user defined commands in the step. The colors that are assigned to a driver can be changed at **Configure System**.

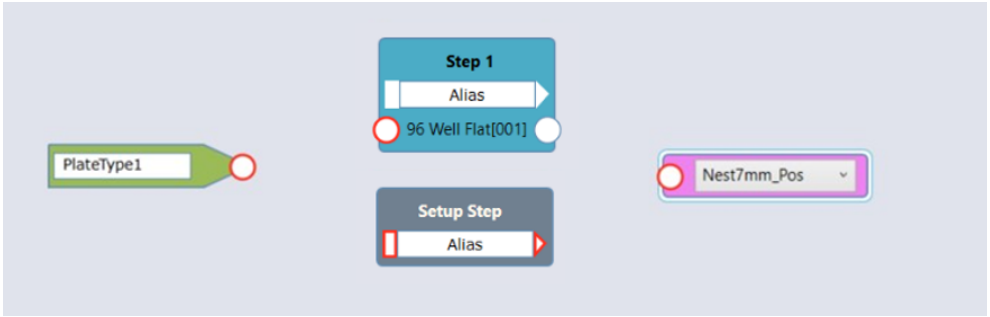


Fig. 391: Processing Steps

After opening a Processing step the Processing step script name and the expected runtime of this step can be set. Both can be edited in script settings. Changing the expected duration of the processing steps in the corresponding script helps to have a better schedule (refer to section “Expected Duration of a Processing Step” [▶ 496]). Fresh context-check errors within a script can be indicated at Step level too (refer to “Context Check” [▶ 500]).

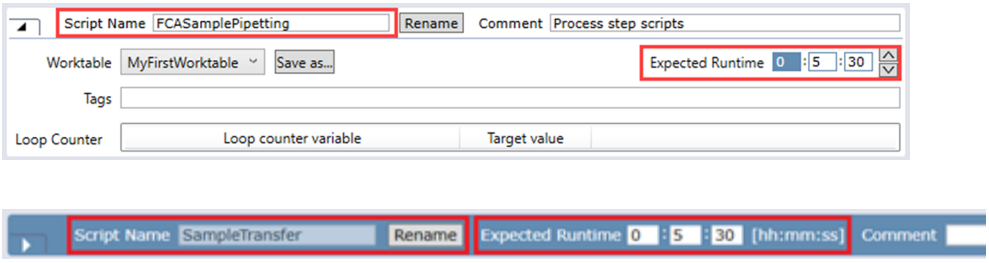


Fig. 392: Sample Transfer

Scripts from Processing Steps are not accessible via Controlbar category Scripts. However, the script names are used at Timing Constraints and Reports. Hence giving descriptive script names is useful.

Processing Step Scripts cannot be saved via the File menu ‘Save’ or ‘Save as…’ options. The Save button below the Script Editor must be used instead.

By drag-and-drop moveable labware has to be set on the worktable (directly onto a Nest of that location) which the system shall automatically transfer to a certain location. The commands can be used as in FluentControl standard Scripts to program the pipetting or other actions of the Processing step script using the previously added labware.

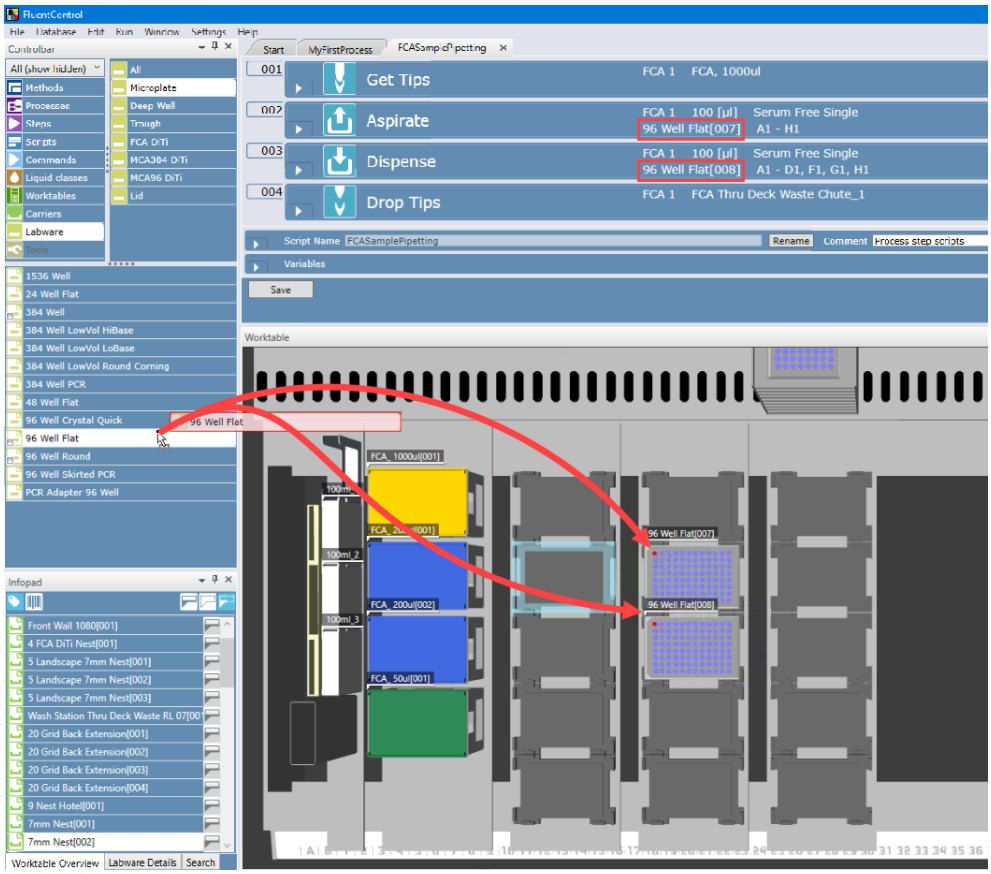


Fig. 393: Moveable Labware

It is possible to add one or more dynamic labware to a processing step. Depending on the complexity of a given process more than five dynamic labware per step can lead to reduced performance. It is also possible to define script variables in the Processing step script, which is similar to the FluentControl standard Script. Once you have finished the Processing step script, save it, and return to Process.

16.1.4.3 Destination Step

The destination step is an optional final step of a labware sequence and defines the destination of the labware. If no destination step is added the default is to return the labware to its base location.

16.1.5 Process Workflow Definition

The software automatically detects the differences between the *Base Worktable* of the *Process* and the script specific worktable and displays additional labware from the script as labware parameters at the bottom part of the Processing steps.

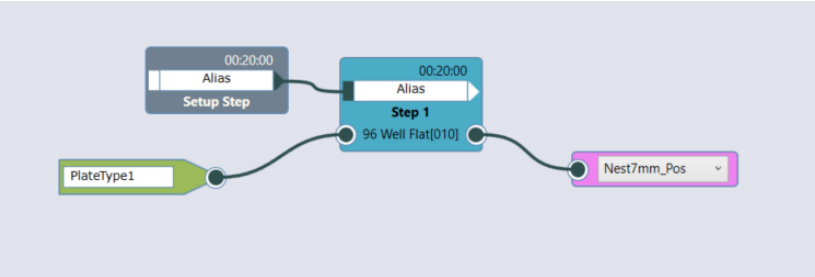


Fig. 394: Connected Steps

Intended use of Transfer Steps and correct use of Destination Step

Only one plate handling command must be used in a Transfer Step. The intended and optional use of customized Transfer Steps is to define additional:
- Commands to move disposable tip labware
- FES finger exchange commands
- MCA384 adapter plate exchange commands

Each labware defined in a Source Step in the Process will be brought back to its <base> after all Process Steps have been executed. There is no need to add a Destination Step with location <base>. If a plate shall be brought to any other place than <base>, a Destination Step defining the desired location must be added as the final Process Step of that labware (pink Step in figure below).

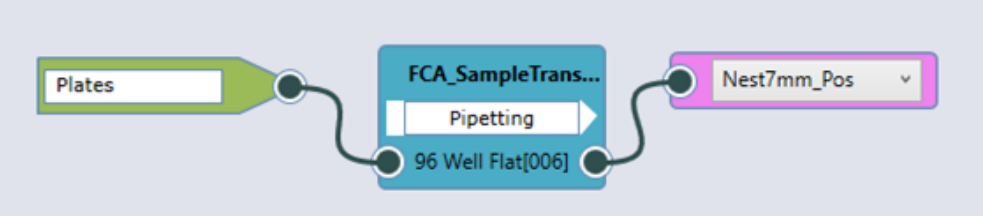


Fig. 395: Connected Steps

If process steps shall be used more than once in the same way as defined, the user has the possibility to duplicate single steps by using copy and paste via context menu or keyboard shortcuts. In the same way the steps can also be deleted. To define a process sequence the labware from Source steps is to be connected to the defined labware of a *Processing* step to give them an execution order. All the process sequences define the later workflow for the process. This means at runtime a labware from this labware group will be assigned to the connected labware parameter of a Processing step and the RGA automatically transfers that labware to the location used on the worktable of the *Processing* step. If necessary, a default transfer step can be customized to add or remove commands to adjust the labware transfer to the actual needs. To do that right click the connection to be customized and select the "*Customize selected connection*" option. A new tab will be opened with the commands of the transfer script and a selection of available commands like for scripts will be shown. A customized transfer step will be indicated by a blue square with three white lines.

Do not change the predefined variables of the "*Transfer Labware*" script command when customizing a transfer step, otherwise there can be problems in combination with a destination step

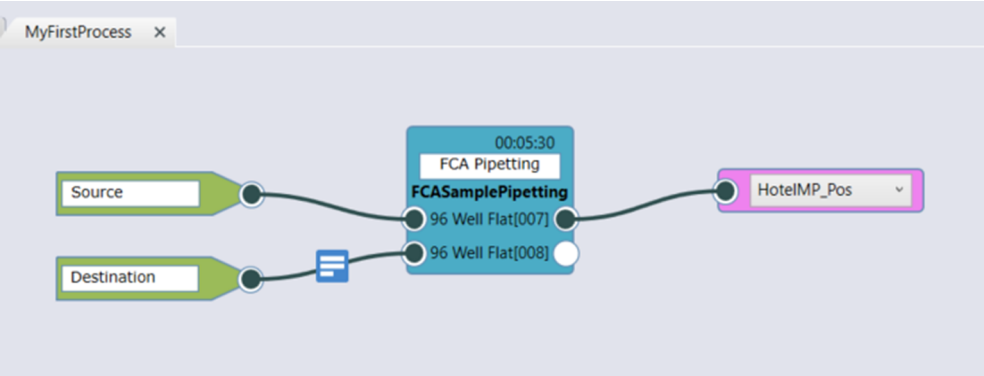


Fig. 396: Connected Steps

The steps can only be connected when the same labware type is used. Note that a step can have different labware types assigned (e.g., '96 Well Flat' and '384 Well'). The connections can also be deleted via context menu or using the Delete key of your keyboard.

The Process Editor can be zoomed in and out diagonally using Ctrl + mouse wheel.

16.1.6 Process Properties

16.1.6.1 Expected Duration of a Processing Step

The expected duration of a Processing step can be set manually in a section of the script.



Fig. 397: Expected Runtime

The expected duration time will also be displayed in the tool tip of the Steps in the Process editor tab.

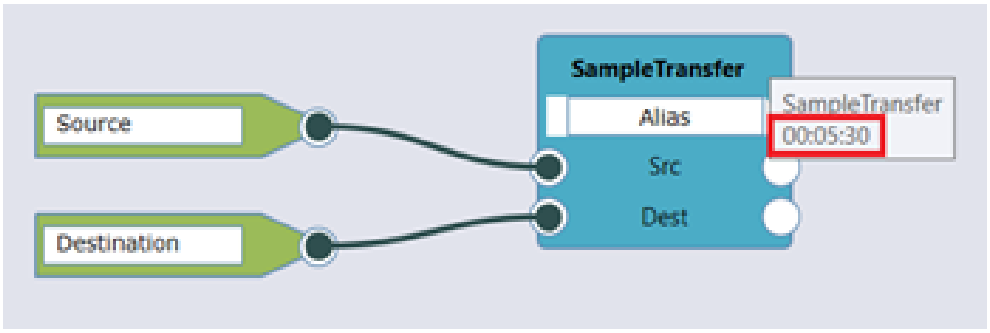
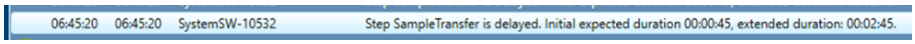


Fig. 398: Connected Steps

The expected runtime of Processing step script shall be as accurate as possible to obtain the best possible schedule. However please consider the longest possible cases one should use the time for the highest number of samples (e.g., when pipetting a variable number of samples). Or you may need to account a bit of extra

time if pipetting is expected to be interrupted at runtime by events such as automated tip handling (refer to section “Automated Tip Handling” [▷ 510]) to exchange disposable tip boxes.

If processing of the pipetting step at runtime takes longer than the expected duration, the scheduler dynamically re-schedules and assigns more time to the Processing step in 2 minute increments.



Such a 2-minute time extension can occur multiple times. Keep in mind that future steps in the run may be affected when re-scheduling or it could lead to a violation of a Timing Constraint (refer to “Timing Constraints” [▷ 498]).

16.1.6.2 Expected Transfer Step Duration

The expected transfer step duration is set to 15 s by default but can also be changed (refer to “Expected Transfer Duration” [▷ 508]).

16.1.7 Variables

It is also possible to declare variables in the Process Editor for using them in the created process. The user can set different scopes.

16.1.7.1 Scope of Variables

The Scope defines where a variable is valid and which items can set or get the value of a variable. The variable scope depends on if you plan to exchange a variable value with other Processes, Iterations or Scripts in the same Run. Do not mix up variables by having multiple variables with the same variable name but having different scopes!

Run: Values of a variables with scope Run are valid and accessible to all Processes, Iterations and Scripts involved in that Run. Please refer to 8.4.1 Variables for more information on persisting variables with scope Run.

Process: Values of a variables with scope Process are valid and accessible to all iterations of a Process during the run but it won’t be accessible to other Processes in the run.

Iteration: Values of a variables with scope Iteration are only valid and accessible for a specific Process iteration. Each Process iteration gets its own instance of this variable on which it could work on and reset itself to the default setting for the next iteration.

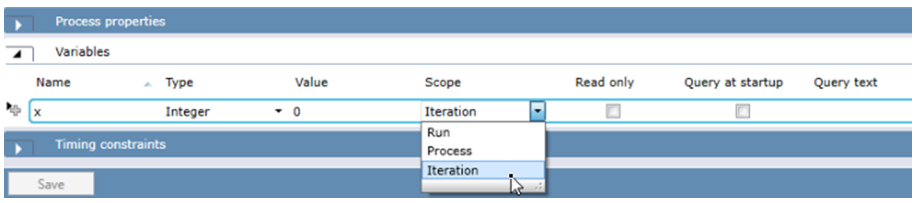


Fig. 399: Process editor tab

Script: Values of variables with Scope Script are only valid and accessible within the Script of a Processing Step.

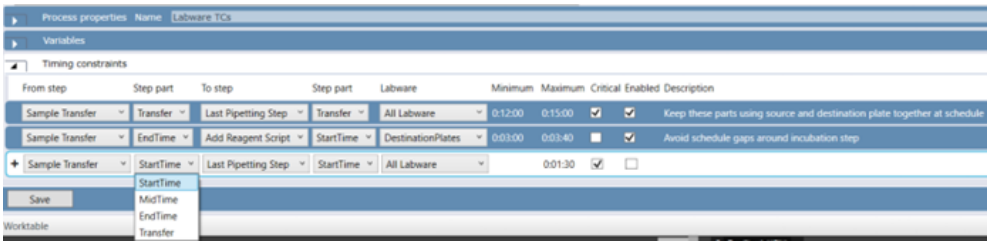
Scope Script can only be declared when opening the script of a Processing Step.

16.1.7.2 ProcessIteration() function

FluentControl contains a function called ProcessIteration() which returns the current iteration number of this Process. Imagine it being similar like a loop counter but counting Process iterations instead. When you assign the result of this function to a variable you can use it for programming.

16.1.8 Timing Constraints

Timing Constraints can be set between Process steps, influence the way the schedule of the run is planned and assist you creating a schedule the way you want it to be. Of course, timing constraints will in turn reduce the flexibility of the scheduler in creating a schedule.
They can be defined at a section at the bottom of the Process tab. Select the Step [script name] – from step and to step – and the Step part [StartTime; MidTime; EndTime; Transfer] involved in the time definition. Enter a minimum and maximum duration time and a description.
Note that timing constraints can only be created between process steps which are connected. Either by labware connections or labware-free connections.
When a step is selected in one of the drop-down boxes the other drop-down lists are updated to only show valid entries.



Labware

By default a timing constraint is applied to all labware common to the two selected end points / process nodes. It is also possible to restrict timing constraints to a specific labware. Then the timing constraint only applies to the labware selected in the **Labware** column.
The drop-down list of the Labware column list all labware common to the selected steps as well as a **All Labware** entry.
Selecting a single labware is useful:

- When you want to affect the loading or unloading of a specific labware, e.g. sample plate.
- When you want to limit the reporting of timing constraint violations to specific labware.

Non-critical timing constraints

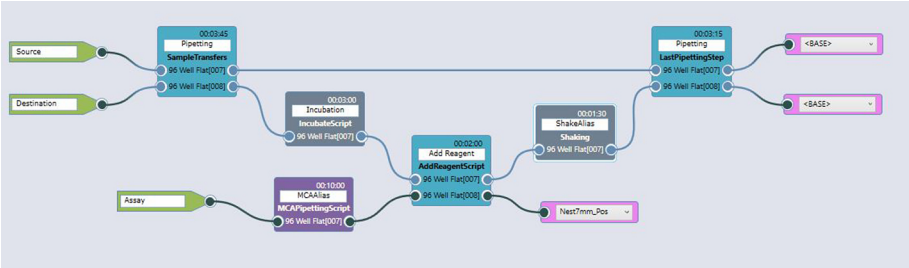
By default timing constraints are critical, i.e. a violation of a timing constraint will show an error in the Trace View and the run will finish with warnings.
Often a timing constraint is introduced to improve the sequence of the steps or limit on-deck-time of important labware. For these cases a timing constraint can be marked as non-critical in the **Critical** column. If a non-critical timing constraint is

violated during run, only an informational entry is shown in the Trace View. Please note that non-critical timing constraints are not reported in the timing constraints report file.

Enable

In order to quickly test the effect of timing constraints on the schedule, individual timing constraints can be disabled via the **Enabled** column. Please note that disabled timing constraints are not used for scheduling and are not monitored or reported.

Connections with Timing Constraints are highlighted. Connections without Timing Constraints are shown in black. If Step part 'Transfer' is part of the timing constraint, the connected line is colored with a color gradient.



Example **Use of a Timing Constraint**

Avoid gaps in the schedule around an incubation since another reagent shall be added after a certain amount of time.



If timers are used to keep the schedule gap closed during an incubation add 10 seconds to the expected duration of the step for step preparation and closing of the step to avoid step delays—e.g.:
A script only containing a Timer and a Wait command that is set to 30 minutes -> use 30 minutes and 10 seconds as the expected duration of that step.

16.1.9 Gantt Chart

During the design time the Gantt chart visualizes the created schedule of the process and provides information about the used labware in the process and the allocated resources. The expected run- and end-time is also displayed, and the user is informed about when a schedule is being calculated or if no schedule is available.

Zooming of the Gantt Chart (not supported by TouchTools)

Mouse Wheel: Scroll Up & Down

[Ctrl] + Mouse Wheel: Extend/Contract X-axis

[Shift] + Mouse Wheel: Zoom, Extend/Contracted X- and Y-axis simultaneously

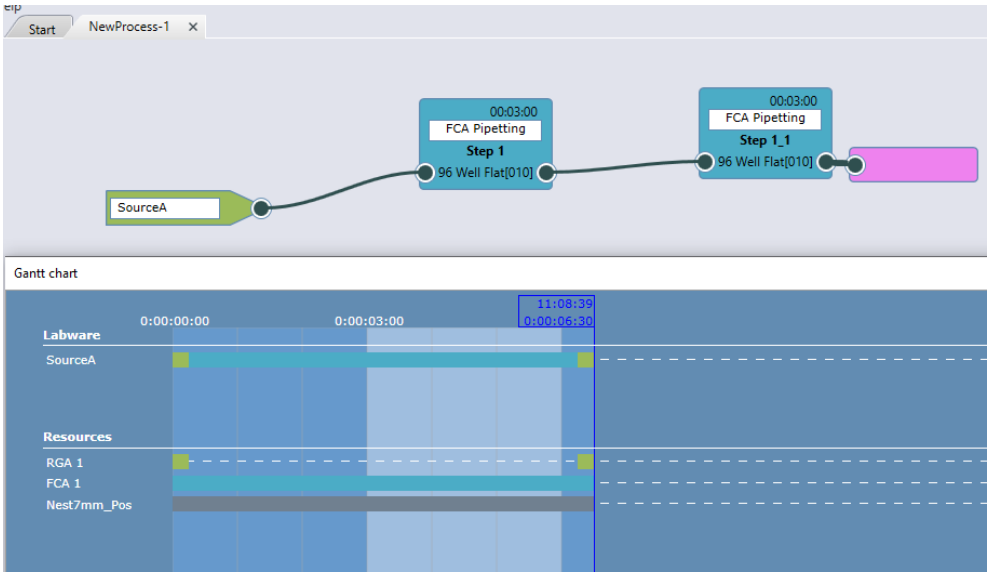


Fig. 400: Gantt chart

For Gantt chart description during run time refer to section “Gantt Chart” [▶ 503].

16.1.10 Menu Options for Processes

16.1.10.1 Import and Export of Processes

Importing and exporting of processes is possible and works similar to the Import/ Export functionality known in FluentControl for methods and standard scripts. If the user is doing a full export, processes are included. For complex scheduling scenarios a lot of files are needed. It can easily take more than 10 minutes to import such a case.

16.1.10.2 File menu

The user has the option to close an open process when using **Close** and also save an open process under a different name when using the **Save as**.

16.1.10.3 Edit menu

The user has the option to copy, paste and delete processing steps in the opened process.

16.1.11 Context Check

A meaningful context check error text is displayed in the Infopad when something inconsistent is preventing the schedule creation. The process tab is marked with a red flag when a context check error occurs in the process. When the processing step script is marked with a red flag then a context check error exists in this script.

16.2 Define Methods Including Processes

16.2.1 Setting up a Method

When setting up a Method one or multiple processes can be combined into one batch in order to be scheduled in parallel. It is possible to use several script batches in a method.

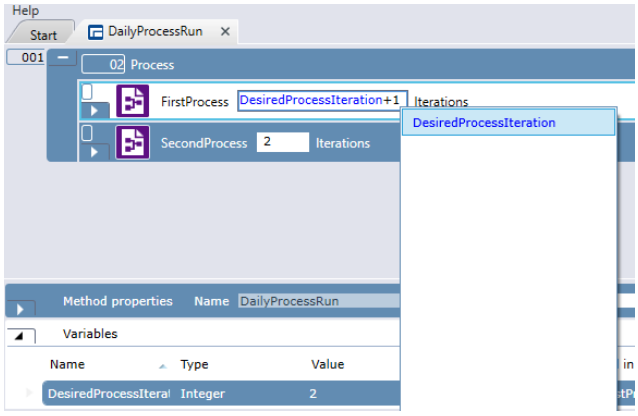


Fig. 401: Setting up a Method

The number of iterations can be defined as an integer, variable with scope Run or expression. If a variable or expression is used, the number of iterations that shall be executed is calculated just before the batch is started.
If the calculated value is not an integer, the default value '-1' will be used. Expressions Floor() and Round() may be used to ensure that an integer value is used. The Trace View and Audit Trail record the number of iterations started for each batch in the method.

16.2.2 Assign Labware by Barcode File

The **Assign labware by barcode file** option allows you to run processes using specific labware items which are identified by their barcodes instead of using the next available labware. Each process batch item in the **Method Editor** has a small arrow, this can be expanded to enable the option by checking the Checkbox. If you choose to use **Assign labware by barcode file** you should provide either a .txt or .csv file. The file should have the following structure:

MyLabware001,MyLabware002,Sample1,Reagent1
ABB123,BAB321,DCC123,AAA123,BBA321
DED123,BAB321,BCD123,BBB123,CDA321 ...

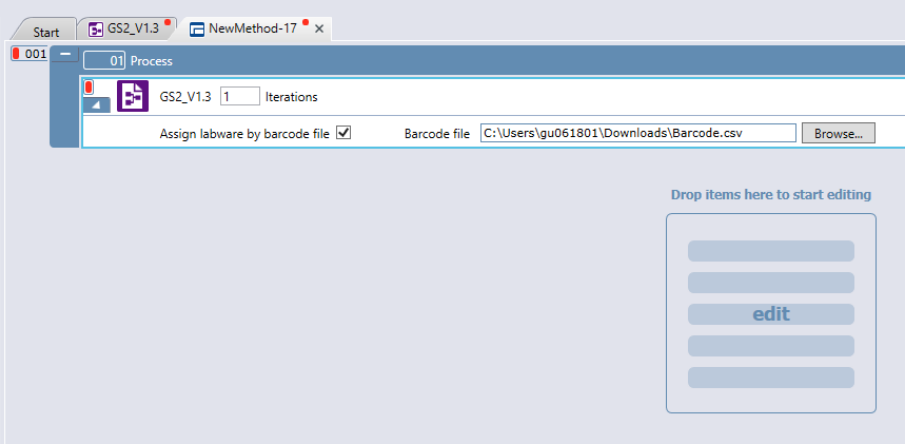


Fig. 402: Process

The first line should contain the list of the source labware group names used in the process Source Steps. The following lines contain the barcodes of the source labware to be used. Each line should include the same number of barcodes as the number of the source labware names in the first line. Each line is used for one process iteration. Empty lines in the file are ignored. You can use semicolons ";" or pipe characters "|" as the separator character instead of commas. The barcodes must not include any of these three separator characters.

The **Assign labware by barcode file** field does not accept variables.
If the field is left empty, the operator is prompted to browse for a .csv or .txt file before the run begins. On the touch monitor, it is not possible to browse for the barcode file, this must be done on the PC running FluentControl.

16.2.3 Touch Tools Settings

In the method's Touch Tools Settings section one can enable **Iteration View** and enter a text that will be shown to the operator at runtime.

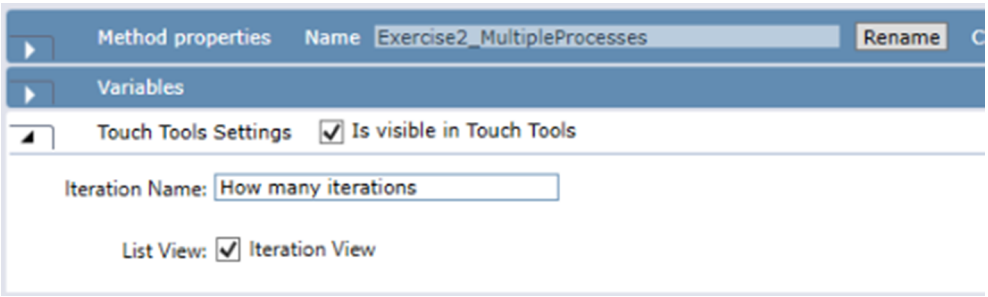


Fig. 403: Touch Tool Setting

Using this method setting allows the operator at Run start to choose how many iterations of each process will be executed in this run.

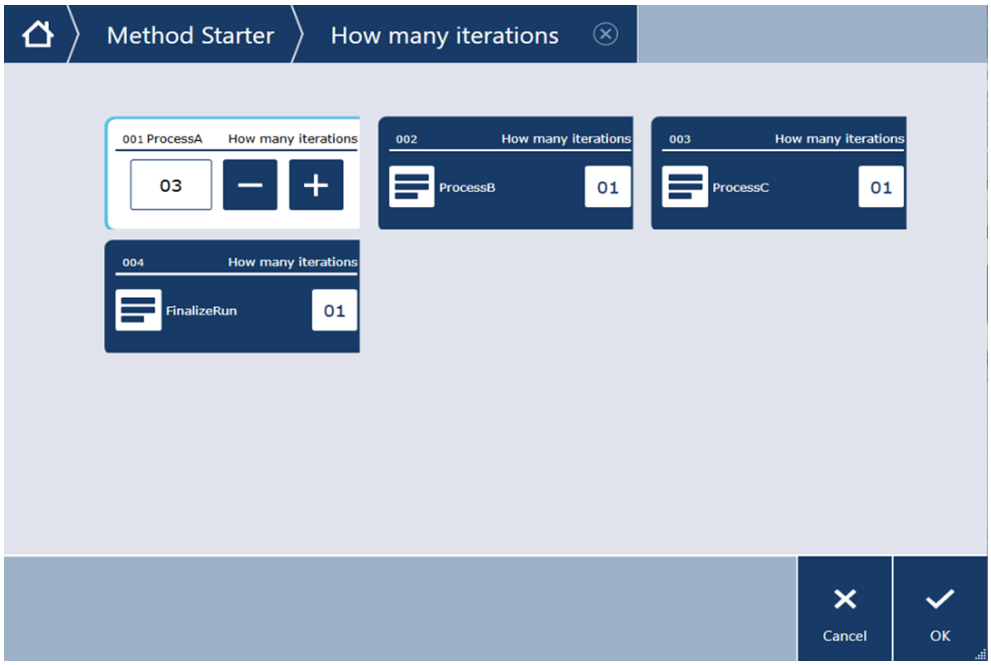


Fig. 404: Touch Tool

16.3 Running a Process

16.3.1 How to Start a Process

A process is started via Method Editor after a method was created including the process of interest as it is common in FluentControl.

16.3.2 Gantt Chart

During the run time the Gantt chart visualizes the timing of the run. As described before, it provides information about the used labware in the process and the allocated resources. The expected run- and end-time is also displayed. A red line (Runtime indicator) moves through the chart and indicates which step is currently running. The user is informed in the Gantt Chart with the help of Overlays when a schedule is being calculated, no schedule available, run was paused, stopped, etc. Whenever a Processing step completes 5 seconds earlier than expected the entire schedule will update dynamically. The expected total duration and end time is visible at the right side of the chart. These times as well as the progress bar in TouchTools adapts dynamically to schedule updates.

The user is also able to have a step information overview by mouse over of a particular step (see in the picture below). Here information like script name, start and end time, expected duration, actual start and end time, actual duration, allocated resources and resources after finishing the run are displayed.

During the run a compact view of the Gantt chart can also be displayed via **TouchTools** on a touch screen. There the Gantt chart will show maximum the next 6 hours of a schedule and each process iteration in one line for better visibility. This provides the user the ability to see when a process iteration is finished and to identify possible idle times of the instrument.

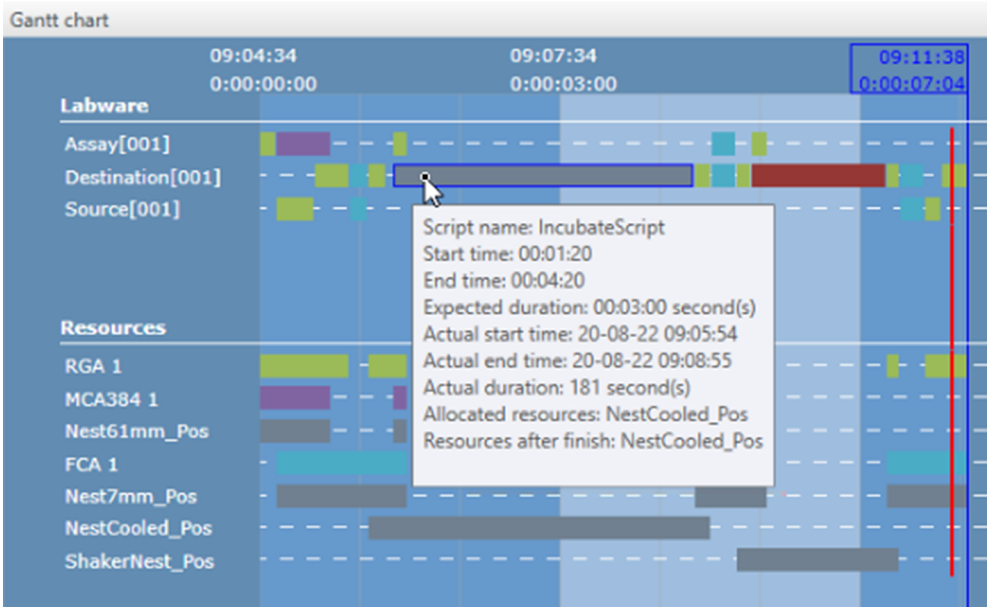


Fig. 405: Gantt Chart

Zooming of the Gantt Chart (not supported by TouchTools)

Mouse Wheel: Scroll Up & Down

[Ctrl] + Mouse Wheel: Extend/Contract X-axis

[Shift] + Mouse Wheel: Zoom, Extend/Contracted X- and Y-axis simultaneously

16.3.3 Abort Labware

It is also possible to abort a labware during the execution via right click on the step and selecting ‘Abort’. The currently running step will finish its execution and the following affected steps won’t be executed and are removed from the schedule. After aborting a labware all affected labware is brought back to base.

It is also possible to abort a complete process iteration in the Gantt chart in **TouchTools**.

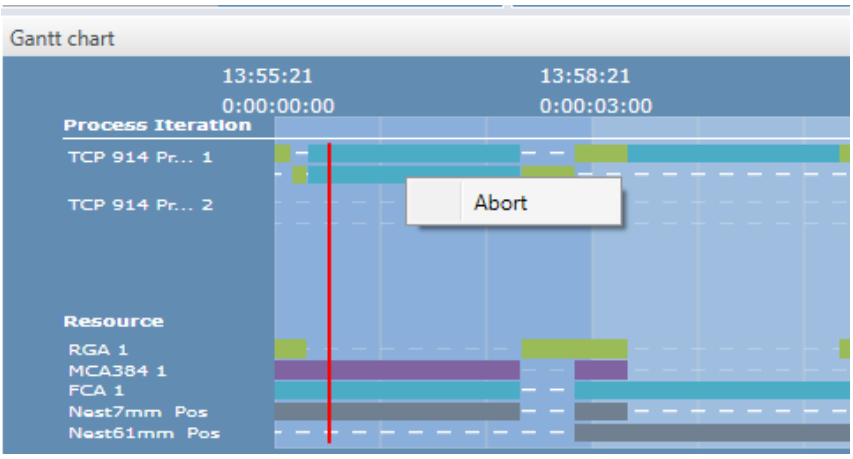


Fig. 406: Gantt Chart

16.3.4 Delay Handling

During execution, it can happen that steps may not finish within their expected duration (as configured in the scripts). In such cases, additional time is added to the duration of delayed step. This additional time is user-configurable in the **Configure System** section under **Scheduler and Parameters** and **Additional time for delayed Steps**. Please refer to section “Additional Time for Delayed Steps” [509] for more information.

While any step is delayed, no new steps will be started. In the case of Automatic DiTi handling which adds new steps to the schedule, where a new schedule is needed, the newly added DiTi replacement and pipetting-resuming steps will be started. A new schedule will be created when all delays are finished.

After a delay occurs, start times of all non-started steps are postponed by the configured time to allow the delayed steps to finish. In case a delayed step is still not finished after the added time has passed, its duration is prolonged again.

The “GanttChart” [503] shows a delay overlay with details as follows:

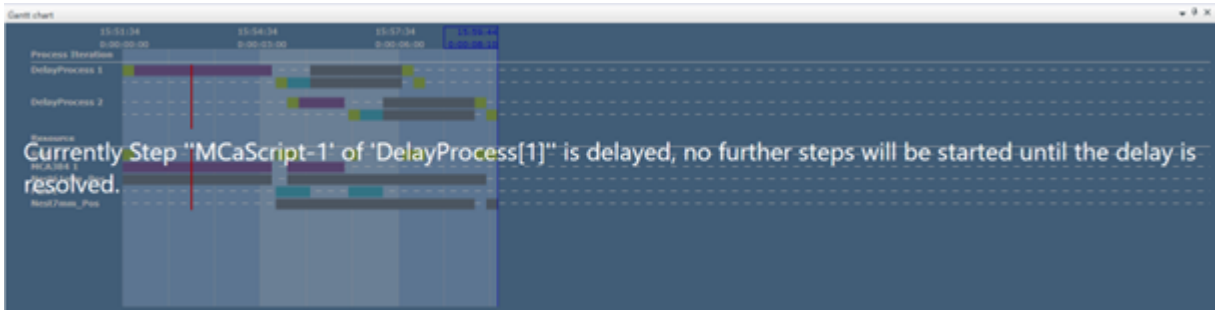


Fig. 407: GanttChart

16.3.4.1 Limitations

Dynamic Loading and the Add button is only available when a Process batch is active and a schedule is running. The Method which will be added may only contain Processes. These Processes will be added to the Process Batch of the active Method.

Both the running and to be added Methods and their Processes must be based on the same Base Worktable. The added items may contain additional labware, but they need to be placeable on the current worktable.

16.3.5 Dynamic Loading

Via Touch Tools it is possible to add additional Process iterations to an ongoing run. Any Process can be added if it uses the same base worktable. For further details see “16.3.4.1 Limitations” [506]. The items to be added need to be part of a Method and this Method needs to be visible in Touch Tools. To add additional workload, press the Add button in Touch Tools.

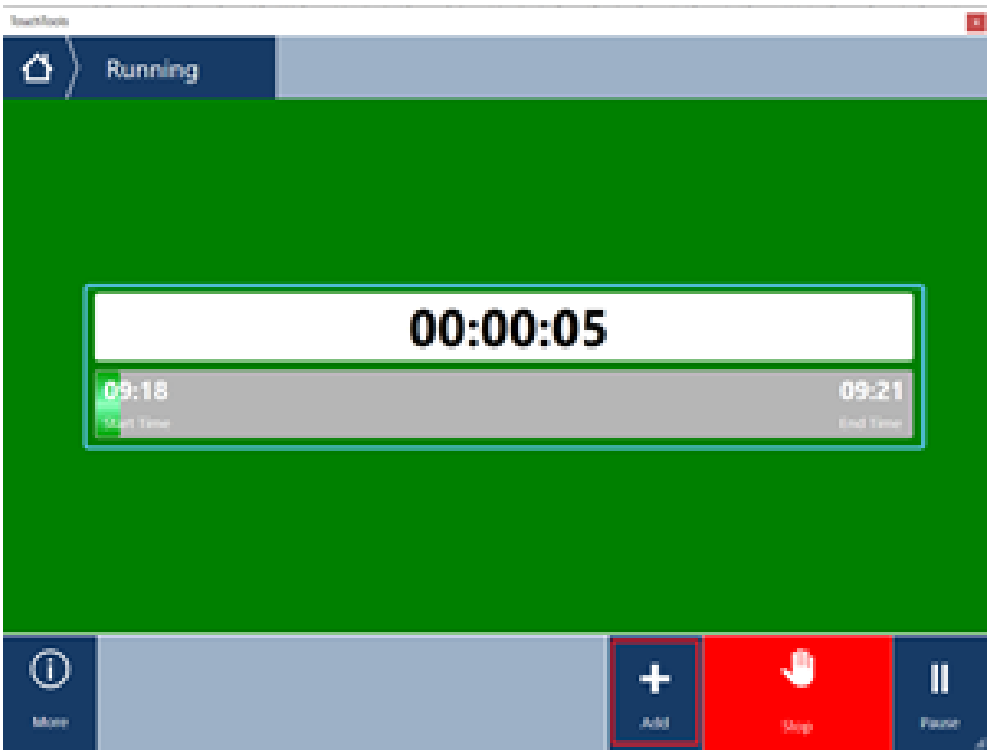


Fig. 408: Dynamic Loading

The workflow is the same as when starting a Method via Touch Tools. See “5.6 TouchTools” [▶ 64] for details on the Touch Tools workflow. Instead of starting a new run, the content of the selected Method is added to the currently running Method and incorporated into the ongoing schedule. Depending on the needed resources the start times of the newly added Process iterations are planned as early as possible.

16.3.5.1 Limitations

Dynamic Loading and the Add button is only available when a Process batch is active and a schedule is running. The Method which will be added may only contain Processes. These Processes will be added to the Process Batch of the active Method.

Both the running and to be added Methods and their Processes must be based on the same Base Worktable. The added items may contain additional labware, but they need to be placeable on the current worktable.

16.3.6 Generated Reports

There is a Labware Report and a Timing Constraint Report created for each run.

The data is stored at:
C:\ProgramData\Tecan\VisionX\AuditTrail\Scheduler

16.3.6.1 Labware Report

Plate	Step	Iteration	Process name	Actual start time	Actual end time	Expected	Actual dur	CompletionStatus
Destination[001]	MyProcess1_DefaultTransferScript	1	MyProcess1	08/17/2020 07:12:15	08/17/2020 07:12:29	15	14	Completed
Destination[001]	DiTiManagement	1	MyProcess1	08/17/2020 07:12:30	08/17/2020 07:12:31	5	2	Completed
Destination[001]	TransferringSamples	1	MyProcess1	08/17/2020 07:12:35	08/17/2020 07:18:06	290	332	Completed
Destination[001]	MyProcess1_DefaultTransferScript	1	MyProcess1	08/17/2020 07:18:26	08/17/2020 07:18:39	15	13	Completed
Destination[001]	IncubationScript	1	MyProcess1	08/17/2020 07:18:41		480	0	Incomplete
Destination[001]	MyProcess1_DefaultTransferScript	1	MyProcess1 from sequence 1	08/17/2020 07:18:56	08/17/2020 07:19:17	15	21	Completed
Source[001]	MyProcess1_DefaultTransferScript	1	MyProcess1	08/17/2020 07:12:00	08/17/2020 07:12:11	15	11	Completed
Source[001]	DiTiManagement	1	MyProcess1	08/17/2020 07:12:30	08/17/2020 07:12:31	5	2	Completed
Source[001]	TransferringSamples	1	MyProcess1	08/17/2020 07:12:35	08/17/2020 07:18:06	290	332	Completed
Source[001]	MyProcess1_DefaultTransferScript	1	MyProcess1	08/17/2020 07:18:11	08/17/2020 07:18:25	15	14	Completed
Destination[002]	MyProcess1_DefaultTransferScript	2	MyProcess1	08/17/2020 07:16:30	08/17/2020 07:17:18	15	48	Completed
Destination[002]	DiTiManagement	2	MyProcess1	08/17/2020 07:18:56	08/17/2020 07:18:58	5	2	Completed
Destination[002]	TransferringSamples	2	MyProcess1	08/17/2020 07:19:01	08/17/2020 07:24:09	290	309	Completed
Source[002]	MyProcess1_DefaultTransferScript	2	MyProcess1	08/17/2020 07:18:41	08/17/2020 07:18:53	15	12	Completed
Source[002]	DiTiManagement	2	MyProcess1	08/17/2020 07:18:56	08/17/2020 07:18:58	5	2	Completed
Source[002]	TransferringSamples	2	MyProcess1	08/17/2020 07:19:01	08/17/2020 07:24:09	290	309	Completed
798603A6C3C63A5AF4E0E95F3B6BAD2ACE9AA6C47D45A1C6BFE5C7497EA3960B								

16.3.6.2 Timing Constraint Report

Plate	FromStep	FromAnchorPoint	ToStep	ToAnchorPoint	Actual Dur	ProcessName	Proc	Min-Constrai	Max-Cor	Violated	Description
Destination[001]	DiTiManagement	Transfer	TransferringSamples	Transfer	00:05:57.0	MyProcess1	1	00:00:00	00:06:00	FALSE	Keep this part together in schedule
Source[001]	DiTiManagement	Transfer	TransferringSamples	Transfer	00:06:00.3	MyProcess1	1	00:00:00	00:06:00	TRUE	Keep this part together in schedule
8131C8F9066BAA7C2C517749F326A183E80846071E8932DA0DB06AFED45AD82A											

16.3.6.3 Customized Labware and Timing Constraint Report

Based on ‘BatchId’ variable defined (Type: String, Scope: run) If set, labware and timing constraint reports name can be customized. Labware report file name shall be "{BatchID}_LabwareReport"

Timing constraint report name shall be "{BatchID}_TimingConstraint"

SetVariable can be used to set the values of the variables if customization is desired.

If ‘BatchId’ variable is not defined or set(i.e. BatchId= empty or whitespace), it uses the default naming convention (i.e. ”{MethodName}_LabwareReport” and ”{MethodName}_ TimingConstraint”

16.3.7 Error Handling

The following error handling is supported during execution of processes:

- In case a Timing Constraint is violated during runtime the description entered is displayed at System Trace, to help identifying the timing constraint. Giving a meaningful short description is helpful.
- If a transfer step is aborted during the run, affected plates are automatically aborted. Unaffected iterations will continue. For the failed transfer, the source location will be blocked the scheduler assumes that the destination location can still be used for subsequent iterations.
- If a step duration takes longer than expected, refer to “Expected Duration of a Processing Step” [▶ 496] and “Schedule Update Timeout” [▶ 508].
- Context check in Process Editor and Processing step Script Editor (refer to “Context Check” [▶ 500]).

16.3.8 Introspect

If you are using the scheduler Introspect can also inform you about start and endtime of a run and if the run takes longer than expected.

16.4 User Rights

As with standard FluentControl scripts and methods only the admin user and key operator user have access rights to edit processes. Operators can only run a validated method.

16.5 Scheduler Configuration

Under 'Configure System' the following parameters can be edited.

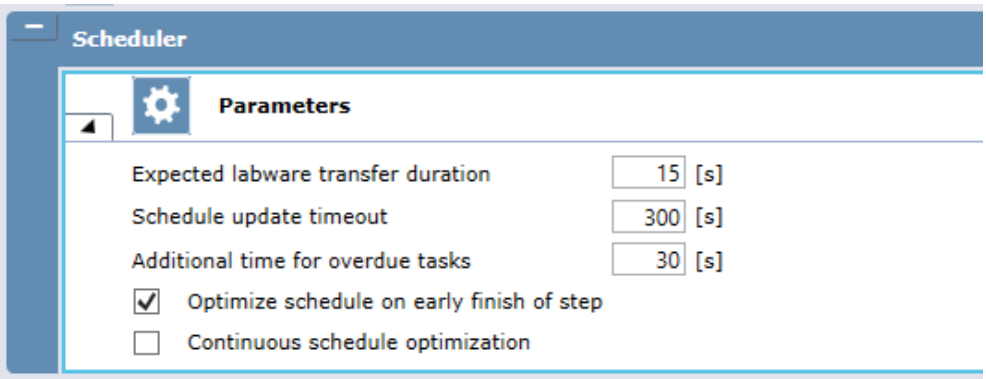


Fig. 409: Scheduler

16.5.1 Expected Transfer Duration

The default expected duration of a transfer step is 15 s. The user has the option to set a new expected duration for transfer steps in **Configure System** under **Scheduler** and **Parameters**.

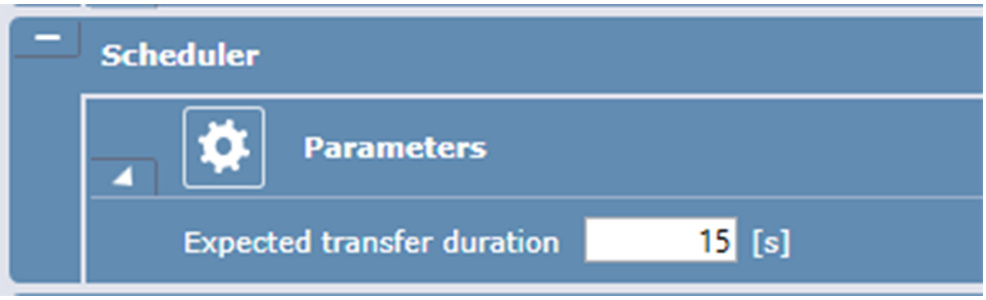


Fig. 410: Expected Transfer Duration

16.5.2 Schedule Update Timeout

The schedule update calculation time has a default timeout of 5 minutes—i.e., if a schedule calculation takes more than 5 minutes, the execution continues with the existing schedule. The user has the option to change the **Schedule update timeout** in the **Configure System** under **Scheduler** and **Parameters**.



16.5.3 Additional Time for Delayed Steps

If a processing step doesn't finish in expected time, additional time for this step to complete is provided. Default is 30 Sec. The user has the option to change the **Additional time for delayed steps** in the **Configure System** under **Schedule** and **Parameters**.

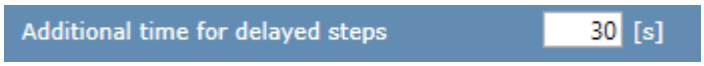
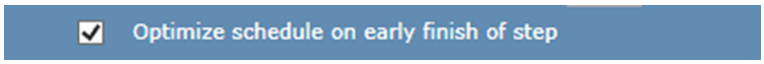


Fig. 411: Additional time for delayed Steps

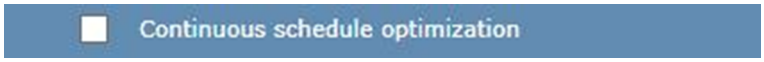
16.5.4 Optimize Schedule on Early Finish of Step

If a processing step completes earlier than expected the entire schedule will update dynamically (to save time if possible). This setting is enabled by default. Depending on the complexity of the process, this can lead to a deterioration of the schedule. The user has the option to disable the re-scheduling on early finish of steps in the configuration settings (in **Configure System** under **Scheduler** and **Parameters**: disable checkbox **Optimize schedule on early finish of step**). If the re-scheduling is disabled, the system would wait to continue with the following step as it was initially scheduled.



16.5.5 Continuous Schedule Optimization

If a processing starts or completes the entire schedule will update dynamically (to save time if possible). This setting is disabled by default. Depending on the complexity of the process, this can lead to a deterioration of the schedule. The user has the option to enable the continuous optimization in the configuration settings (in **Configure System** under **Scheduler** and **Parameters**: enable checkbox **Continuous schedule optimization**).



16.6 The Labware Group Editor

As already described in section "The Process Editor" [▶ 491] when using Scheduler functionality there is a difference between static and movable labware. The movable labware is used for automated transfer to certain locations and defines the labware sequences of a process. The movable labware can be set when designing a worktable. Therefore, storage carriers are used (e.g., hotels or carousel towers). With these settings the source for the labware sequences can be defined and used in process creation. For this, FluentControl opens the Labware Group Editor tab and the user is able to select the required labware type in the Controlbar. In the Processing step Scripts this movable labware can be set on different worktable locations, like nests. In this way, the user tells the software where the previously defined labware from the source or storage location shall be transferred to in order to be processed.

Static labware can also be set when creating the worktable. This usually encompasses non-processable labware like troughs, reagent reservoirs or carrier-like labware. Static means, that this labware is not automatically transferred from a source or a previous location to another defined location. But note, if placing the labware on a storage carrier, like hotels or carousel towers, the labware will be automatically defined as a dynamic labware. Labware which is placed on the worktable (e.g., an Nest location) in the Processing step Script is not a static labware.

16.7 Devices

Using devices for Process execution is generally working the same as known from FluentControl. At this point there are some workarounds for driver usage and the usage of duplicated drivers.

When using the **Hydroflex washer** during execution the user normally has the option to use variables for the ‘Wash’ command to use a flexible number of strips. Note that so far the usage of variables is not possible in combination with this command, so a fixed number of strips must be used.

When using **shakers** (MIOs or other shakers) and more than one plate in the incubator, the user need to do the following: If one of the plates shall be transferred from the shaking incubator to another location, the device stops shaking. The RGA transfers the plate but so far it is not working that the shaking continues for the rest of the plates after this transfer. To ensure continue shaking, the user has the option to start it manually or working with ‘If’ commands (system shall check locations, if there is a plate and start shaking again, if there is one).

Also, device duplication is generally working the same as known from FluentControl. There is a workaround when duplicating devices: to control more than one device of a device group it is also recommended to work with ‘If’ commands to have a better control about where the plates has to be brought to and executed in. E.g., when using more than one shaker, to have a better control about which shaker has to stop shaking or continue shaking; when using more than one reader, to have a better control about which reader has to be used and start a method.

16.8 Automated Tip Handling

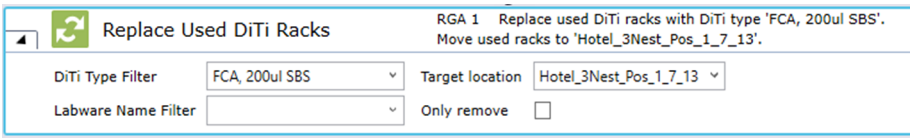
16.8.1 Automated Tip Handling (ADH)

To prevent user interactions during a scheduling process it's possible to program an automated DiTi handling (ADH) option. Disposable tip boxes can be stored in a storage such as a hotel, carousel, or stacker and whenever a tip box on the worktable deck is consumed, the ADH exchanges the empty box with a full box from the storage location. Setting up an automatic DiTi handler works the same way as with conventional FluentControl scripts.



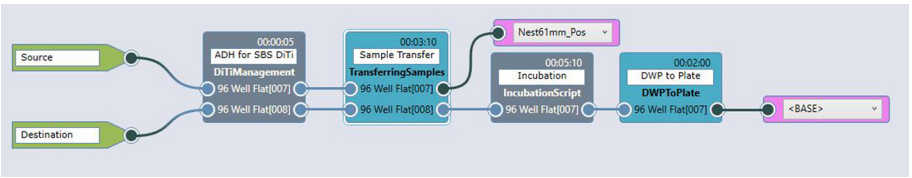
Fig. 412: On Empty Tray

The subroutine can contain any script you like to program, but typically it involves a RGA – Replace Used DiTi Racks command to exchange the tip boxes (or FCA – Remove Empty Tip Spacers in case of Nested FCA Diti & Transfer Tool).



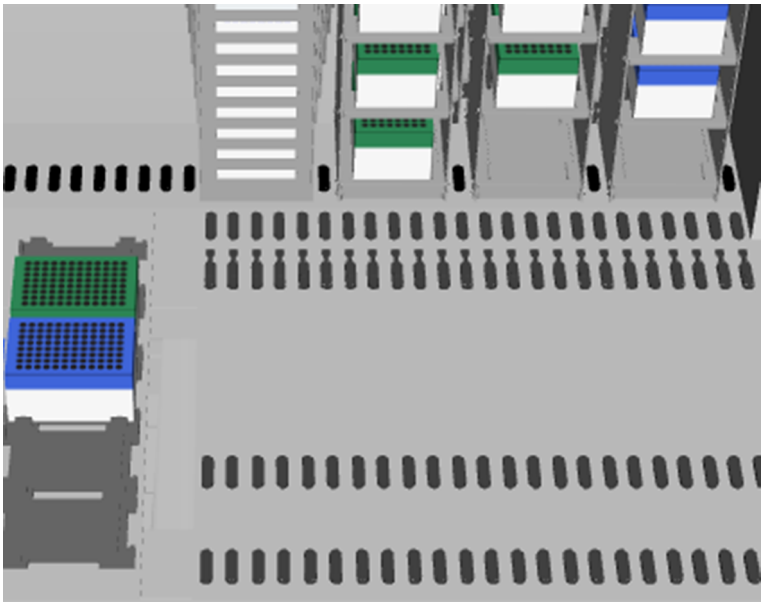
16.8.2 Example Process with ADH for FCA SBS DiTi Boxes

Here is an example script using Automated DiTi Handling. The DiTi Handling is defined in a separate Step at the beginning of the Process and applies then to the entire process. It's not mandatory to have this DiTi management as a separate step, the 'On Empty Tray' commands could also be included at the start of the first FCA step. It is sufficient to have the 'On Empty Tray' command executed once at the beginning it will apply to other Processing Steps as well.



The DiTi management step contains 'On Empty Tray' commands that trigger the corresponding subroutine to exchange the FCA DiTi boxes.

Please consider in case the first set of tip boxes initially starts on the worktable deck, enough storage space needs to be available for the new boxes at the first tips exchange.



16.9 Validating a Process

For further reducing the risk of potential failure effects, consequences of failures or potential hazards during the execution of processes, pay attention to the following when setting up processes:

- Check that Schedule creation is logical and correct to avoid wrong or partial execution.
- Check after the validating run that the correct process step scripts were executed in the correct order and all timing constraints were met.
- Check that process variables (process-scoped and process-iteration-scoped) are defined correctly to avoid wrong results or any error during execution.
- Check that the correct labware is used to avoid wrong results or any error during execution.
- Consider the usage of controls for all runs to improve process safety.

16.10 Optimizing a Process

For optimizing the execution of a process, the following has to be considered:

- Using minimum incubation across multiple steps: by creating steps that have an expected duration close to the duration of the timing constraint the quality of the resulting overall schedule can be significantly improved.

  Bad example:

  Good example:

- For creating a better initial schedule, the report can be used to check step duration of the single planned processing steps. ca. 10 s can be added to the actual execution duration of each step. This time can be set as expected runtime of the processing step.
- If a step takes longer than the expected duration, the scheduler dynamically re-schedules and assigns more time to the Processing step in 2 minute increments. This can lead to deterioration of the schedule and bad results. To avoid unnecessary re-scheduling and 2 minutes delay when using a timer or other commands which can be used for an incubation time, please set the expected runtime not as the exact planned and set incubation/delay time. Increase the expected duration by 10 seconds, e.g. the planned and set incubation is 1 h and the set expected runtime is set to 1 h and 10 sec.
- If a step completes earlier than expected the entire schedule will update dynamically (refer to “Optimize Schedule on Early Finish of Step” [▶ 509]). Depending on the complexity of the process, this can also lead to a deterioration of the schedule.

16.11 Support for Cell-Based Assays

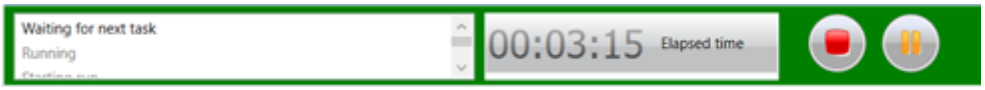
For long running assays (multiple days to weeks), the system supports the execution of Tasks. This allows the system to be shut down and restarted for maintenance or other usages during long runs.

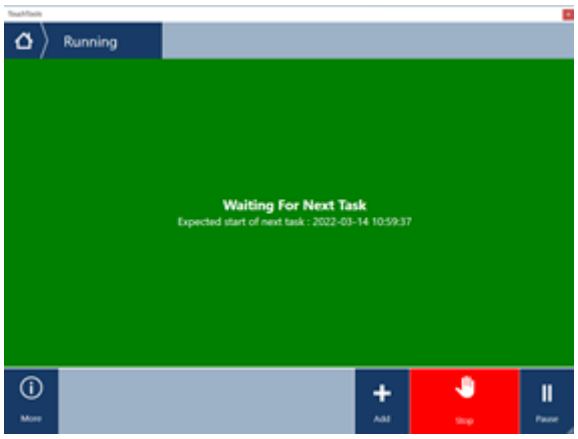
Each Task links a Process, along with the labware to use, to a requested start time. This way an assay with incubation times of multiple days can be split into several short Processes. Between these Processes the labware is returned to its base location, for example an incubator. At the requested start time the Process iteration will either be added to an active schedule, or a new schedule will be created.

To define Tasks the “Create Task” [▶ 385] and “Start Task” [▶ 385] commands are used. See the commands for details.
When Tasks are added to the system, the system keeps track of them and will try to execute them at the requested start times. If pending Task are present at FluentControl start up, the software will automatically start a run to execute the Tasks. The user can cancel this automatic run via a dialog which is shown for 30 seconds. If the start of the run is not aborted within this timespan, the run will be started.

The system may be used as normal while no Tasks are active. FluentControl will inform the user via the status bar about the time when the system is supposed to start the next Task. If the start time has come, the system will show a dialog as a reminder. Via this dialog the execution can be started or suppressed. If the start is suppressed, the status bar element with the Task information can be clicked to re-open the dialog.

Whenever the Task execution is running and no Tasks are due to be started, the system will stay in Run mode. The FluentControl main UI will as well as Touch Tools will show “Waiting for next task”. The requested start time of the next task is shown in the status bar and on Touch Tools.





While the system is waiting for the next task to become due, it is safe to press the Stop button to end the run. After the run is stopped the system can be used as normal, for example to run maintenance Methods. Be sure not to alter the Runtime worktable unintentionally. Select “Preserve all labware positions from last run” in the Method properties if you are in doubt.

After the system is no longer needed by the user, the status bar can be used to go back to Task execution mode. Click on the element showing the requested start time of the next task in the middle of the status bar.

To allow internal software clean-up processes to run, FluentControl will perform a daily shut down and restart. The default time for this restart is 23:59. The default can be changed via a .twl file.

For ease of use when defining Processes for cell-based assays, we recommend to add all commonly used labware in the Base Worktable. This way they do not need to be defined in every Process individually and will simplify later changes.

16.11.1 Task Input File (.twl)

To add new Tasks, either use the Create Task and Start Task script commands or add a .twl input file to the Worklist folder: C:\ProgramData\Tecan\VisionX\Worklists. The Worklist folder is constantly scanned for new .twl files while FluentControl is running.
When a new file is detected, it is processed and removed from the folder. A copy of the file is stored in the Audit Trail as an electronic record.

The structure of the file is defined as follows:
P, Date or relative Day, Time or relative Time, ProcessName, LabwareSource
R, Time, Weekdays

The following table specifies these parameters.

Parameter	Valid Inputs	Explanation
Indicator	P or R	“P” indicates that it is a line to add a new process task. “R” indicates that the restart time is set.
Date or relative Day	+0, 2022-03-10, 2022-03-10T15:00, 2022-03-10T15:00Z	+ indicates relative day offset. Absolute Date is given as UCT time or as local time without time zone identifier.
Time or relative Time	+1:00, 15:00	+ indicates relative time starting with the time when the file is read. Absolute time is given as HH:mm
ProcessName	MyProcess	A string of a Process name
LabwareSource	Source(Source[001]), Barcode	A string of the LabwareGroup name or a Barcode
Weekdays	1, 2, 3, 4, 5, 6, 7	A list of integers representing the weekdays a restart is allowed to be performed. 1: Monday, 2: Tuesday, etc. **This parameter is optional.** A restart is allowed on all weekdays when nothing is specified.

Examples:
CellBasedAssay.twl => P, +0, 15:00, TestProcess, NewGroup1(NewGroup1[001])
Restart.twl => R, 23:59, 1, 2, 3, 4, 5

16.12 Not Supported

Please check the following functionalities, which are not supported:

- When using Subroutines or Modules in Processing Step Scripts, make sure that these are not using other resources than the calling Script. FluentControl will only reserve the resources (arms, devices and Worktable positions) used by the Script of the Processing Step itself. Usage of other resources can lead to unexpected behaviors and abortion of the run.
- More than one process batch per method
- Methods containing processes cannot be recovered in case of an abort of the run

- Worktable manipulation via commands to add or remove labware (e.g., SetLocation) may lead to an unexpected behavior.

17 Utilities

17.1 Log Viewer

1. To start the **LogViewer** tool, click **Start > All Programs > Tecan > Tools > LogViewer**.
2. Click **Tools > Filter**.

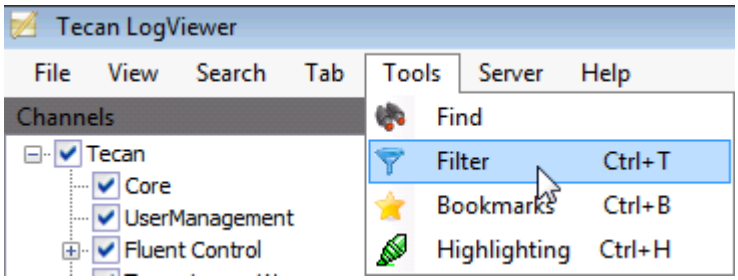


Fig. 413: Tecan LogViewer

3. In the Filter dialog, click **Add**.

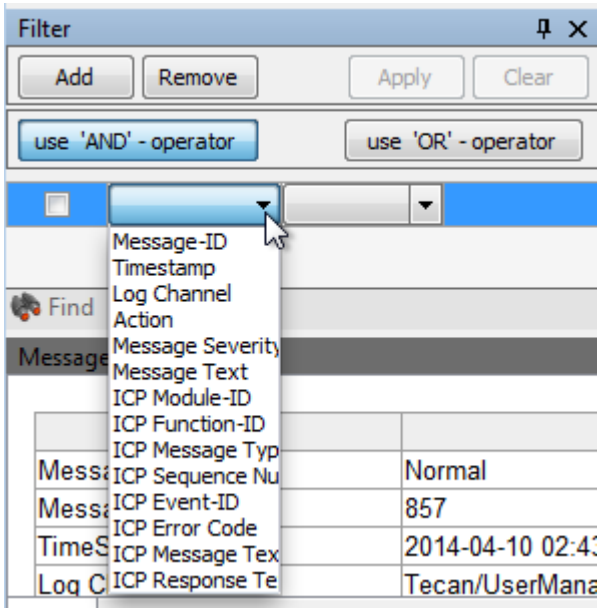
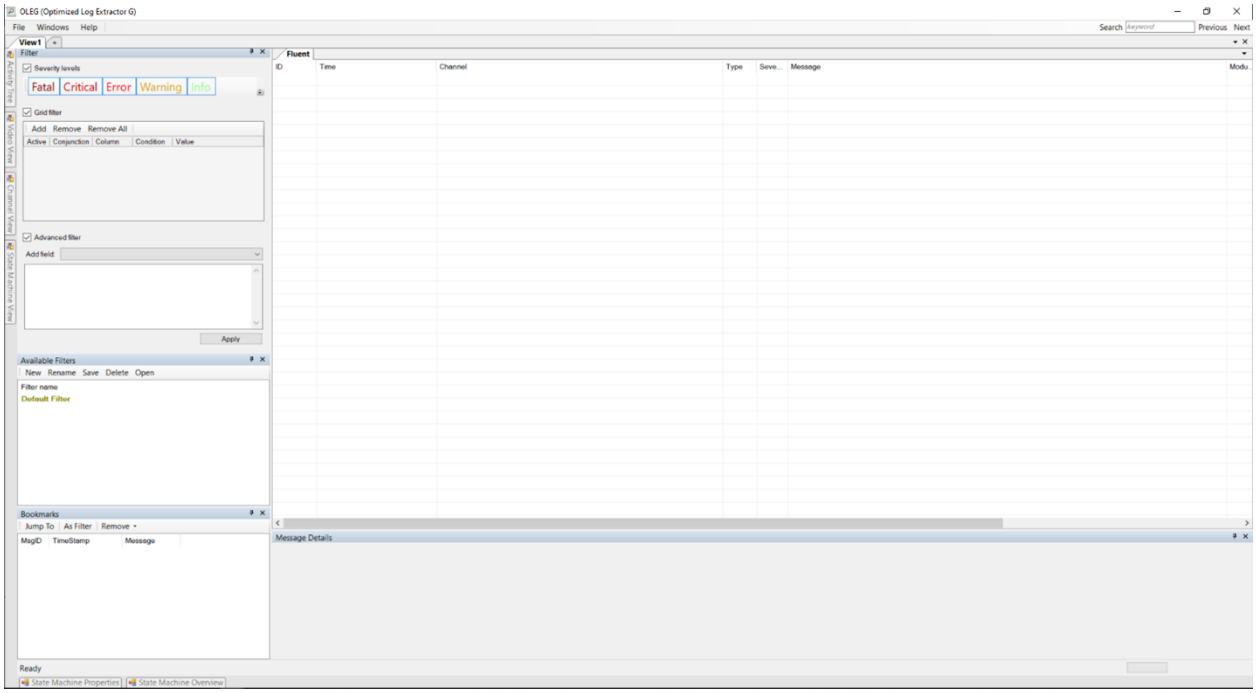


Fig. 414: Select Message Severity.

4. Click the arrow of the first box, and select **Message Severity**.
5. Click the arrow of the second box, and select =.
6. Type `Error`.
7. Select the check box, and then, click **Apply**.
   You get a list of all errors for the selected Channels.

17.2 OLEG Log Viewer

1. To start the **OLEG** tool, click **Start > All Programs > Tecan > OLEG**



2. For more information on how to use the OLEG tool consult the OLEG manual accessible through the Help menu within the OLEG application.

17.3 Command Line Parameters

FluentControl supports the following command line parameters:

-u <user> with activated user management: pre-select user for login
-w <password> with activated user management: provide password for automatic login
-sim enforces FC start in simulation mode and skips startup questions if no instrument found
-method <methodName> opens specified method
-script <scriptName> opens specified script
-process <processName> opens specified process

18 Customer Support

This chapter explains which files and information Tecan requires to perform a first assessment of an issue.

18.1 Data Collection for Customer Support

When contacting customer support, please provide the following information:

- **Serial number** of the instrument.
- **Detailed description of the problem** that includes:
  - Timestamp of the error
  - Instruction on how to reproduce the error
  - Has daily and weekly maintenance been performed on the system?
  - Video or photo that shows the error
  - LOT number of item for complaints related to consumables
  - etc.
- **Description of preliminary troubleshooting:**
  What has been already attempted to resolve the problem?
- **Instrument hardware and software configuration:**
  - Arm order, tip configuration and integrated devices
  - Software and driver versions
- The ZIP file generated with the **Snapshot Tool** as well as **additional files**, if such as:
  - Names of affected scripts or methods and a ZEIA export file including these. “See section 18.1.2” [▶ 519] on how to create an export file.
  - Additional screenshots illustrating the issue.
  - Additional log files. FluentControl file archiving might be active. To gather archived log files, check the destination folder defined in any active AuditArch archiving actions. For more information on AuditArch,see section 13.4.1

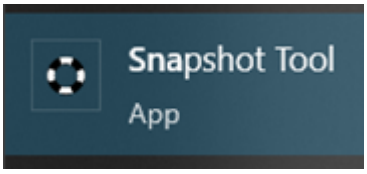
DumpFiles are files that are created when FluentControl unexpectedly closes. In case FluentControl freezes and does not respond it is possible to create DumpFiles manually from the jump list, or alternatively from the task manager. To do so, right click FluentControl in the taskbar and click **Create Dump File**.

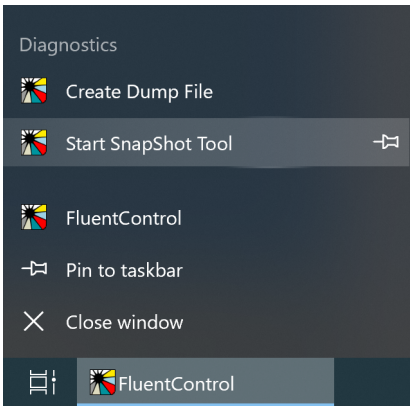
18.1.1 SnapShot Tool

The **Snapshot Tool** is used for collecting e.g., *Instrument Files, Screenshot, Instrument Configuration, Firmware Versions, Windows Event Logs, Journal Files and Audit Trail and Sample Tracking Log Files*and logfiles from other devices installed if needed, over certain period of time. The collected information will be stored in a ZIP file for further use and a pop-up window will show where the ZIP file is stored at the end of the collection process.

To create a Snapshot, proceed as follows:

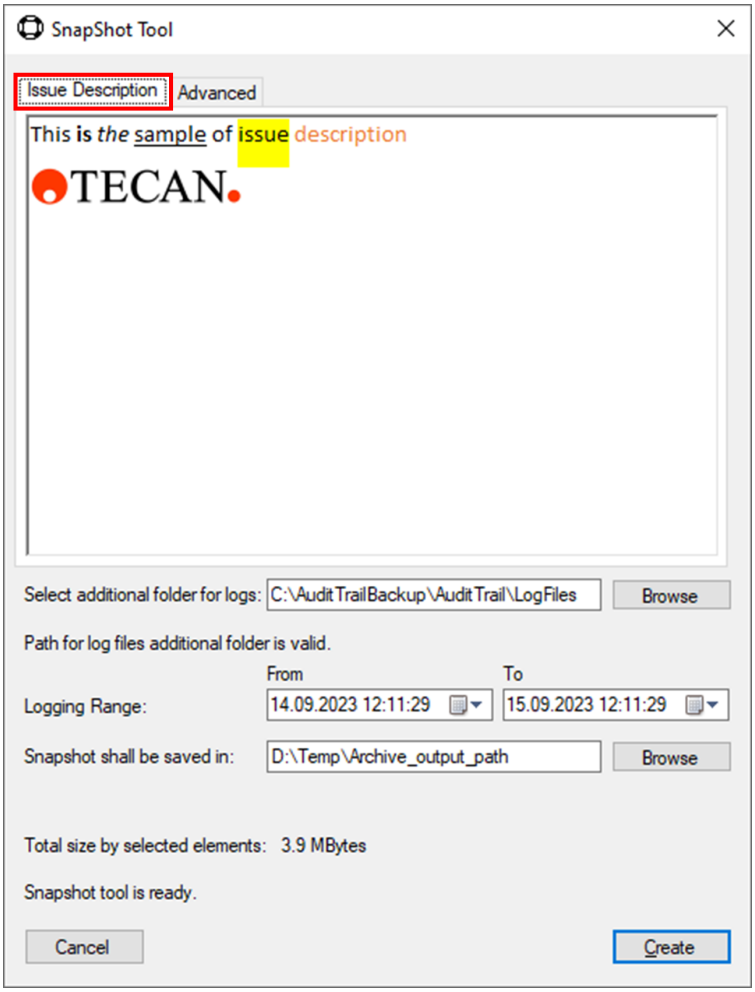
1. Open the **SnapShot Tool** from the Windows Start menu, or from the FluentControl jump list.





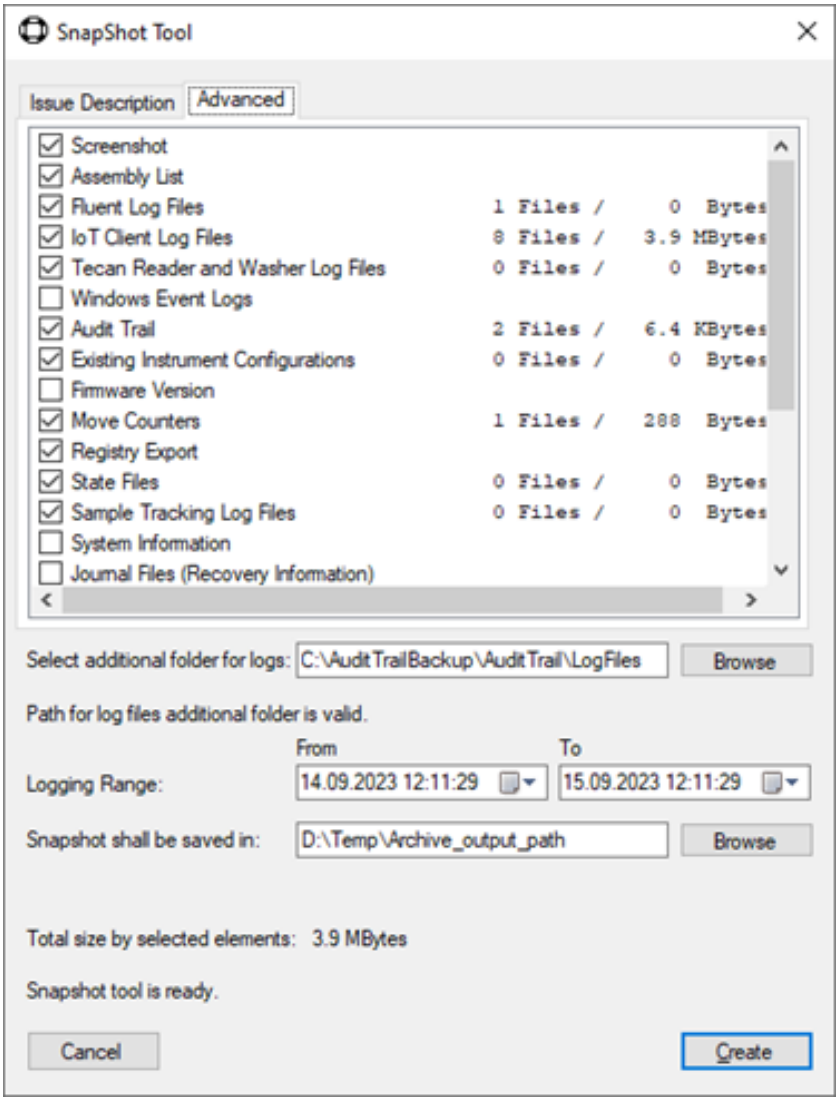
2. The **SnapShot Tool**starts with the **Issue Description** tab. Please describe the issue in detail in this window. You can also add drawings, images or further screen shots.
Select additional folder for logs: The user can specify a folder for additional log files which will be included in the ZIP (e.g. DriverFramework log files).
Logging Range: Select the time range within which the issue(s) or error(s) occurred. Narrow down the time frame the issue was observed to limit the amount of data to be sent.
Snapshot shall be saved in: Define the path to save the ZIP-file of the snapshot.

Total size by selected elements: Depicts the total file size of the snapshot.

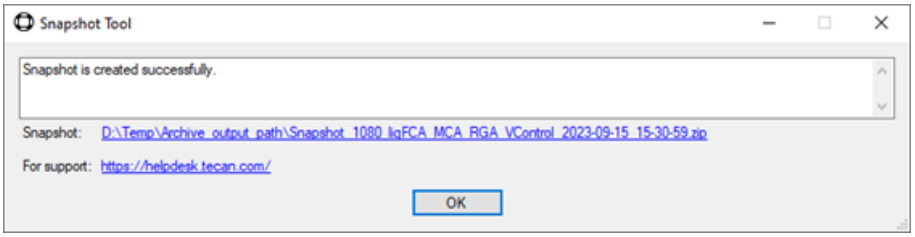


3. Switch to the **Advanced** tab to select which data shall be collected. Additional data can be added, for example: *Add Sample Tracking Log Files* for Sample Tracking-related issues, add *Journal Files* to collect method recovery data for

issues related to recovery. In case you observed an unexpected shutdown, or a software freeze include *Dump Files*.



4. When all data are selected, and a valid path is chosen the “**Snapshot tool is ready**” sign appears.
5. Click **Create** to start collecting the data. This procedure might take some minutes. Once data collection is completed a message displaying where the ZIP-file is stored is shown. Click on the **For support** link to create an online support request directly.



Tab. 25: : List of elements that can be included in the snapshot

Screenshot	Takes a screenshot of each monitor, including the touch monitor, after clicking the **Create** button. Make sure that the FluentControl user interface is visible on one of the monitors.
Assembly list	Collects the FluentControl installation files and their version numbers.
Instrument log files	Collects the instrument log files.
IoT Client log files	If the system uses any cloud services, such as Introspect, activate the checkbox.
Tecan Reader and Washer log files	Collects Tecan supplied washer or reader log files.
Windows Event logs	Collects FluentControl information, if the software has crashed or is frozen.
Audit trail	Collects audit trail information.
Instrument configuration	Collects instrument configuration information. In demo mode, or simulation, no data is collected.
Firmware version	Collects instrument firmware version information. In demo mode, or simulation, no data is collected.
Move counters	Collects move information about arm. In demo mode, or simulation, no data is collected.
Registry Export	Collects all related Tecan entries in the Windows registry.
State Files	Collects the state of each arm at the time of the SnapShot creation.
Sample Tracking log files	Collects Sample Tracking information, if this option is installed on the system.
FRIDA reader log files	Collects FRIDA reader log files, if a FRIDA reader is connected to the system.
System Information	Collects information about the connected computer.
Journal files (recovery information)	Collects Method recovery files. Needed if the issue is connected to Method Recovery. Can be large in size and their collection time-consuming

Database	Collects the content of the FluentControl database. Only needed upon request by Tecan support.
SVN root	Collects the FluentControl SVN root, but only upon request by Tecan.
Path Finder Swift Dumps	Collects **PathFinder** swift dumps. Needed for analysis of pathfinding and collision avoidance issues .
Dump Files	Collects the memory dump files which are automatically created when FluentControl has crashed.
Driver configurations	Collects the FluentControl driver settings, but only upon request by Tecan.
User Description	Contains the data the user enters in the *User Description field*.

Add new entry called "DeckCheck pictures": Collects the pictures, that were taken by the Capture Worktable command of the Deck Check driver in png-format. They are available here:\ProgramData\Tecan\LoggingServer\LogFiles\DeckCheck Pictures.

18.1.2 Export / Import Tool

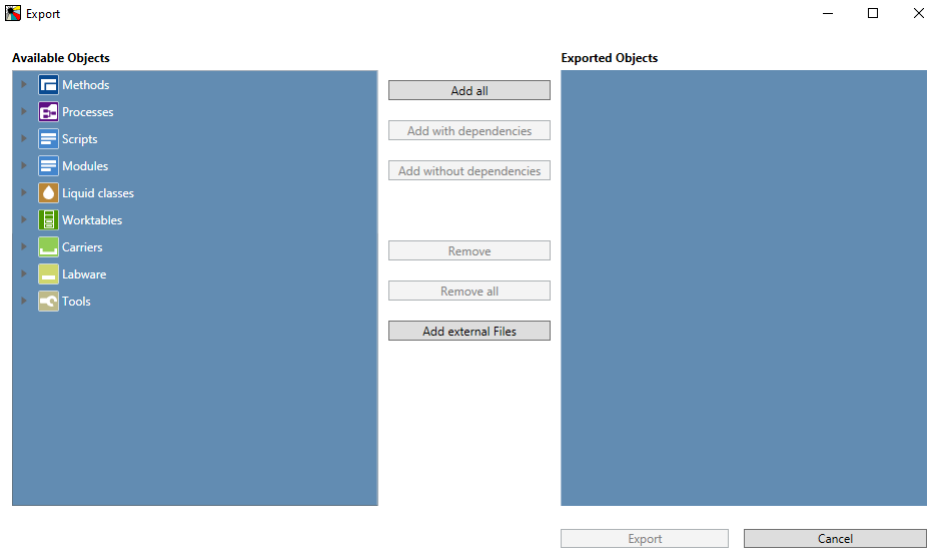
FluentControl includes an Export / Import tool for exchanging scripts, liquid classes, carriers and labware definitions with other installations, for technical support or for backup use. The Export / Import tool can be accessed from the main FluentControl application window and produces files with the *.zeia extension. Note: For troubleshooting, use the Export tool to export relevant methods and/or scripts including dependencies as a *.zeia file. To add other external files that are not automatically referenced, such as worklist files, click the “Add external Files” to the exported objects. Make sure all dependencies of a method or script such as carriers, labware, and liquid classes are always included.

1. In the **Database** menu, click **Export...** or **Full Export....**

   *The **Export** dialog window opens.*

   *The **Available Objects** list on the left is expandable and contains items that are available for export.*

   *The **Exported Objects** list on the right contains items that are currently selected for export (empty in the image below).*



2. Select the items to be exported by using one of the following options:

   ***Add all** adds all items in the database to the **Exported Objects** list.*

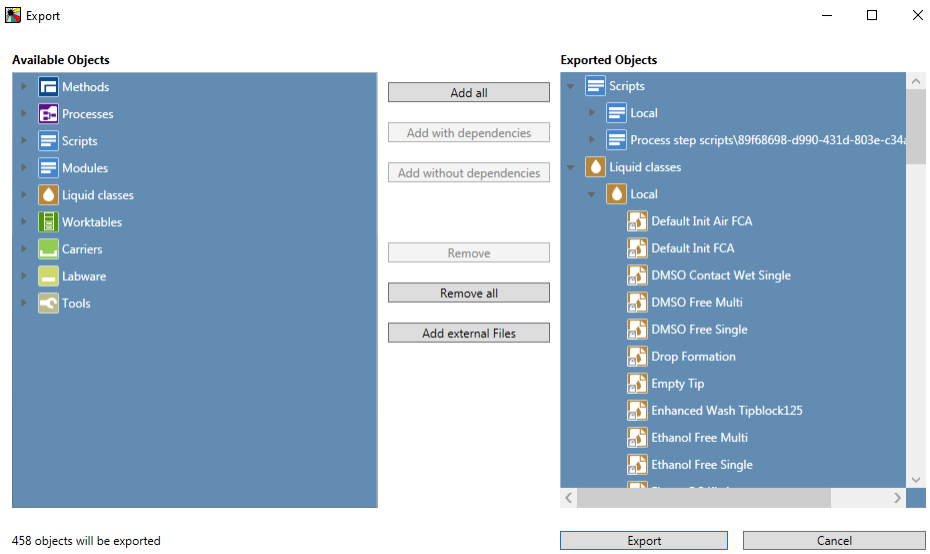
   ***Add with dependencies** adds all selected items to the **Exported Objects** list—including their referenced scripts, liquid classes, worktables, carriers, labware and tools.*

   ***Add without dependencies** adds all selected items to the **Exported Objects** list—without their referenced objects. Note that if any of the referenced items is missing, it may not be possible to import these items successfully to the target system.*

3. To include external files in the export (e.g., images, PDFs, etc.), click **Add external Files**.

   *In the file browser that opens, select the files and click **Open**.*

*The selected external files will be added to the **Exported Objects** list.*



4. Click **Export** to export all items of the **Exported Objects** list.

18.1.3 Import Mode

To import a file proceed as follows:

1. On the **Database** menu, click **Import...**
2. In the file explorer browse for and select a .ZEIA file to be imported.

   *The **Import** dialog box opens.*

   *The expandable **Available Objects** table contains all items that are available for import. They are arranged in the same way as in the **Controlbar**.*

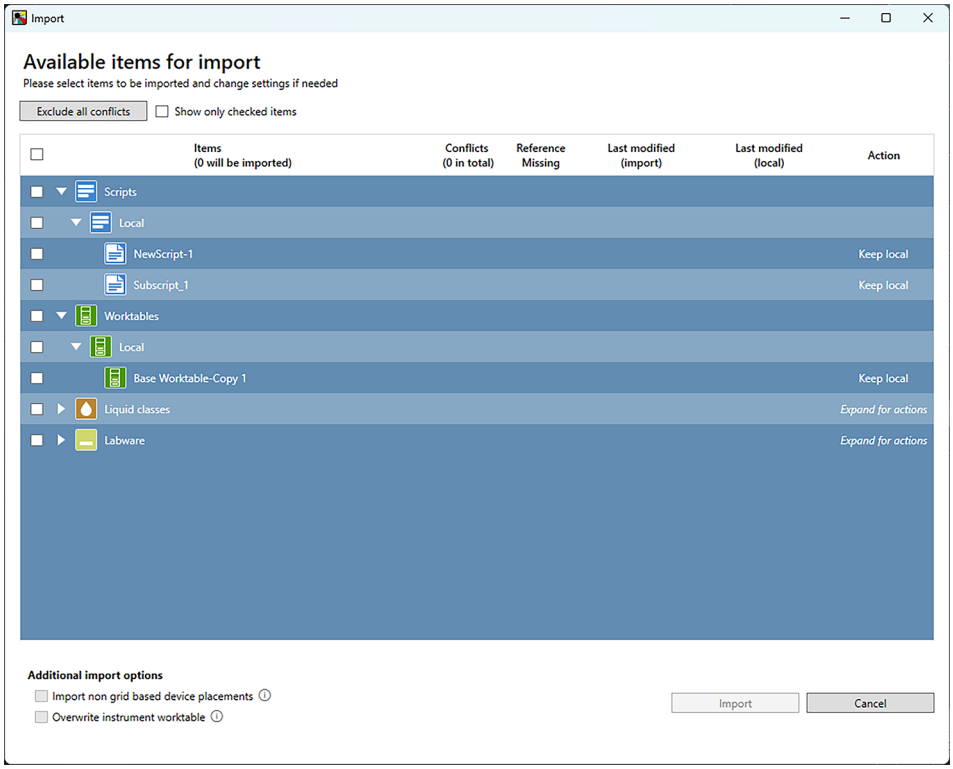


Fig. 415: Default item view

3. Select the items to be imported by using the following options:

   Select the checkbox next to a specific item (e.g. script) to select that item for import and any items it depends on. For example, when a script is selected, all labwares, liquid classes etc. that it requires, will be selected automatically as well.

   Select the checkbox for a certain category of items (e.g. Worktables) to select all items contained under it.

   Select the checkbox in the table header to select all items for import. Clicking the checkbox when all items are already selected will unselect all items instead.

   It is also possible to exclude individual items from import by unchecking them, but this may result in context-check errors with imported scripts or methods or missing components on the imported worktable. Carriers, labware or tools that are not present on the target system – but are used by a worktable to be imported – must also be imported.

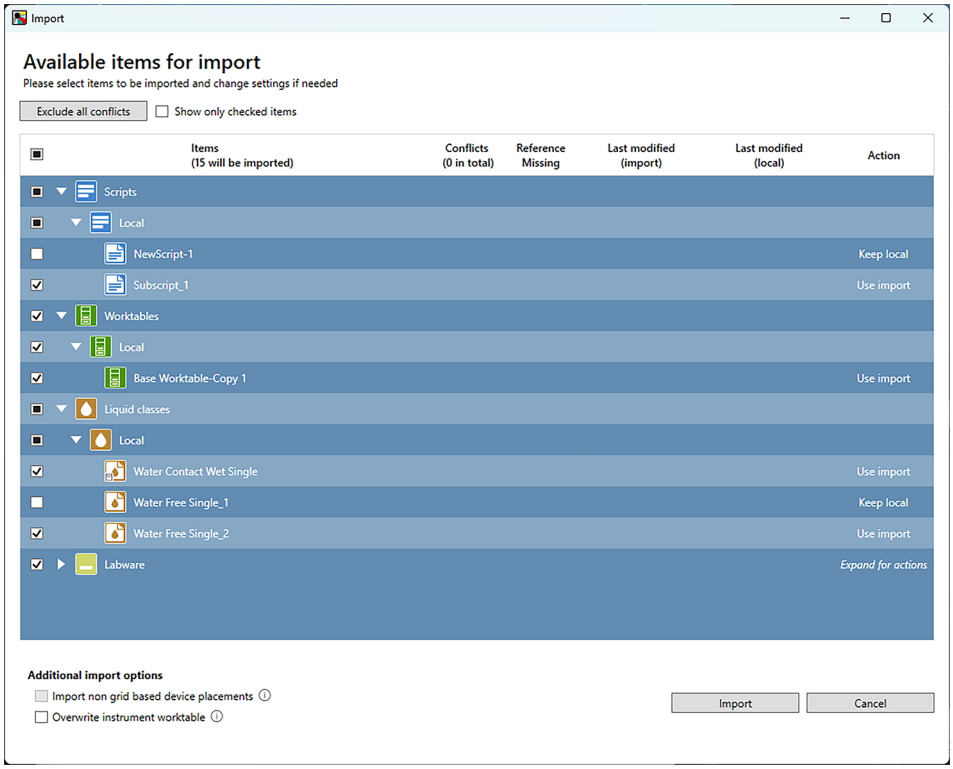


Fig. 416: Some selected items

4. When available items already exist on the target system and conflict with the existing item, they are marked with an entry in the “Conflicts” column. Conflicts occur when items have the same name, type and subfolder but different content. For each conflicting item, decide whether the conflicting item shall be imported or not with the following options:

   Resolve individual conflicts by selecting the checkbox next to a specific item (e.g. script) to decide whether to import it or not. If selected, the imported item will replace the local version. If not selected, the local version remains unchanged. The action selected for each item will also be displayed in the “Action” column.

   Use the button “Exclude all conflicts” to bulk unselect all conflicting items, therefore keeping the current local state of the files that would conflict.

   For conflicts, the columns “Last modified (local)” and “Last modified (import)” give some context as to whether the local version of the item or the import version of the item is more recent.

   Additionally, the “Show only checked” items button can be used to filter the table to only show items that will be imported (i.e. are checked) or to switch back to showing all items again.

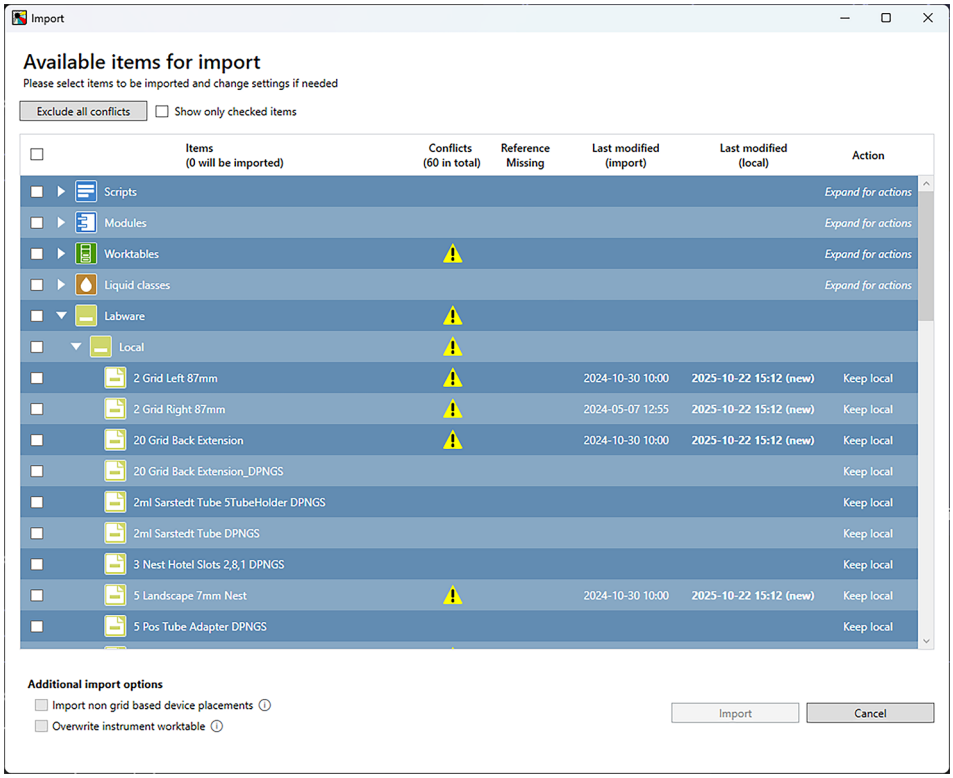


Fig. 417: Some conflicts

The ‘Reference Missing’ column flags items that depend on objects not included in the import (e.g. required liquid classes or labwares).”

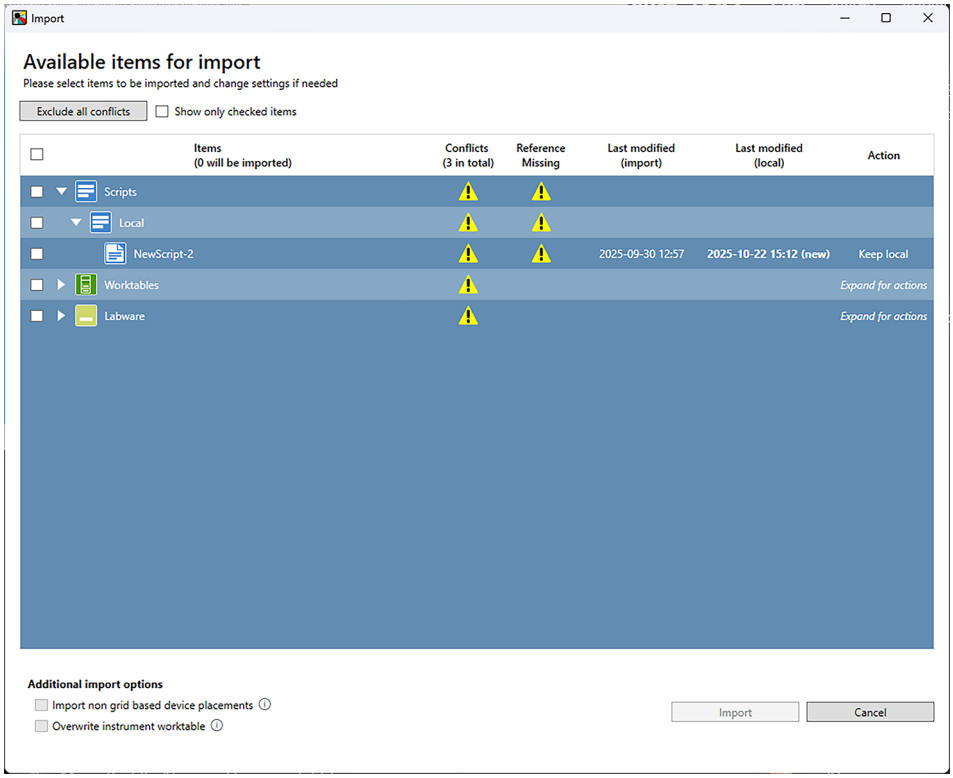


Fig. 418: Some missing references

5. If a worktable is imported, it is possible to overwrite the instrument worktable currently present in the application with the one provided in the *.zeia file. To overwrite the instrument worktable, select the checkbox “Overwrite instrument worktable”. For more information on the instrument worktable, please see the chapter [instrument worktable].
6. The checkbox Import device placements specifies whether it will import placement data for non-grid based device carriers. Overwriting potentially already existing data has only an effect in a simulated environment but not if a physical device is connected, since device placement is stored on the device itself (Te-Control) and will be synchronized automatically.
7. To perform the import, click the “Import” button or alternatively the “Cancel” button to abort the import process.

18.2 Contacting Tecan

For more information on Tecan Support, please refer to https://www.tecan.com/support.

19 Abbrevations

19.1 Abbreviations Table

This tables establishes a list of abbreviations for the Fluent:

Tab. 26: Abbreviations table

Abbreviation	Definition
ARP	Air Restriction Pipetting
cLLD	capacitive LLD (pLLD -> pressure LLD)
CV	Coefficient of variance or variation
cXP	cXP Diluter
DiTi	Disposable Tip
DMSO	Dimethyl Sulfoxide
DWP	Deep Well Plate
EN	European Norm
FCA	Flexible Channel Arm
FES	Finger Exchange System
FSE	Field Service Engineer
IAM	Identity Access Management
LDT	Liquid Displacement Technology
LLD	Liquid Level Detection
MCA	Multiple Channel Arm
MCH	Multiple Channel Head
MP	Microplate
MS	MultiSense
PC	Personal Computer
PMP	Pressure Monitored Pipetting
RGA	Robotic Gripper Arm / standard
RGA-Z	Robotic Gripper Arm / long Z
RWP	Rapid Wash Pump
SVN	Subversion (Open source from Apache)

Abbreviation	Definition
TES	Disposable Tip Ejection System
USB	Universal Serial Bus

Glossary

Accuracy

The degree of closeness of measurements to a standard or true value. It is expressed as a percentage, represented by the function: Difference between expected value and actual value, divided by the expected value, multiplied by 100%.

Active Wash Waste System

A liquid active wash and waste station for low volume fixed tips for deep and shallow cleaning positions.

Adapter DiTi

Adapter for the MCA384 pipetting head for mounting disposable tips.

Adapter Fixed

Adapter for the MCA384 pipetting head with 384 or 96 fixed tips.

Adapter QC

Adapter for setups and tests with the MCA384.

Adapter Rack

A special rack for the System Carrier of the multiple channel arm (MCA384). Various versions to hold tip adapters, DiTi boxes or microplates.

Application

Generally refers to a s specific procedure or test, such as RIA (Radio Immuno Assay), EIA (Enzyme Immuno Assay), etc. A corresponding method is programmed in the relevant applicationl software and the application is performed with defined samples and reagents.

Base Carrier

A special carrier for the multiple channel arm (MCA384), which is equipped with a wash station and adapter racks to hold tip adapters, DiTi boxes or microplates.

Base Worktable

All carriers, labware and tools that are not saved only with the script. In general all carriers and tools are by default a part of the base worktable. The assignment to the base worktable can be altered with the shortcut menu of any active worktable object. Base worktables can be shared between scripts.

Batch

A batch is one part of a method and consists of scripts.

Breakpoint

The system will pause before the execution of the command with a break point. The Operator must press continue on the touch interface or runtime controller.

Capacitive Liquid Level Detection (cLLD)

This is a standard feature of the liquid handling channels of the flexible channel arm. It measures the capacitance between the pipetting tip and the electrical ground (instrument deck). As soon as the tip touches the liquid, the change in capacitance triggers a signal, which is used for liquid detection. The special features of the cLLD also monitor the tip for an exit signal during aspiration and when the tip retracts from the liquid. The function only works with conductive tips (and DiTis) and conductive liquids.

Carrier

The generic term for components that can be placed on the instrument's deck. Carriers are categorized as nest segments, deck segments, runners (placed on deck segments) and special deck components (e.g. disposable tip waste station, combo carriers).A carrier is a removable support on the instrument's deck. You can use a carrier to hold one or more pieces of labware in place. Carriers can be fitted to the deck at pre-defined grid positions (positioning pins/ clips). The pre-defined grid positions are spaced 25.0 mm apart in X-direction. By default, carriers are saved in the base workspace, labware is saved in scripts. NOTE: Carriers are also used to represent positions on devices which can be accessed by a robot. For example, a robot can put a micro plate onto the shuttle of the Te-Stack stacker when the shuttle is at the transfer position. The carrier name indicates the labware type for which it is designed (for device carriers, it indicates the device type).

Carry Over

Residue of any liquid that remains in a tip after rinsing at the end of a pipetting cycle. Such residue is "carried over" to the next cycle. Where carry over needs to be kept to a minimum, disposable tips (DiTis) must be used.

Cleaner

The position in the wash station into which a tip is placed during a wash cycle. System liquid is pumped through the tip. The liquid rising in the cleaner rinses the outer surface of the tips. Overflowing liquid is collected in the waste.

Coefficient of Variation (CV%)

The statistical representation of the precision of a measurement. It is expressed as a percentage, represented by the function: Standard deviation divided by the mean value, multiplied by 100%.

Container

Any vessel placed on or under the deck and containing a liquid or other chemical, e.g. one well in a microplate, a sample tube or a system liquid bottle.

Control

A liquid containing a known concentration of the substance which is to be tested. Used to determine (high/ low/cutoff) limits and/or as reference for quality control. The properties of the control are well known and stable.

Deck

Part of the instrument where carriers and options are placed for access by the robotic arm(s).

Diluter

A precision pump used for aspiration and dispense of exactly defined liquid volumes. It mainly consists of a syringe with a motor-driven plunger and a valve to control flow direction. Each pipetting channel is equipped with a diluter.

Disposable tip (DiTi)

Tip often made of plastic that will be discarded after a single aspiration/ dispense cycle. Used whencarry over needs to be kept to a minimum.

DiTi Transfer Rack

A special rack for the multiple channel arm, which is used to pick up or park the disposable tips.

EMC

Electromagnetic Compatibility

Excess Volume

The volume of excess liquid which is aspirated together (not separated by an air gap) with the liquid to be distributed. It is not dispensed anywhere, but discarded to waste (or a special position) after use, and serves to minimize dilution of the reagents by the system liquid.

Field Service Engineer (FSE)

A person especially trained by the manufacturer for the performance of maintenance and service jobs on the instrument. Also carries out the initial installation of the instrument and its options as well as product upgrades at the user's site.

Fixed Tip

General term for a washable and reusable tip for aspiration/dispense that is fixed to the pipetting device.

Flexible Channel Arm (FCA)

Robotic arm positioned above the instrument's deck area. It holds the pipetting tips and controls the liquid handling. Robot arm on the pipetting instrument which is mounted on the X-slide and typically holds four or eight pipetting tips. It is used for pipetting samples in different volume ranges, depending on the tip types used and the features of the liquid system.

Fluent Carousel

An optional module (Fluent Carousel) to store microplates in stacks that are arranged in a circle. It offers random access to the microplates and can be used for microplate feed and incubation purposes.

FluentControl

FluentControl is the software for the Fluent.

Flush

The procedure which rinses the complete liquid system with the purpose of removing outgassed air or exchanging the system liquid. It is executed only at the beginning or the end of a pipetting cycle.

Free Dispense

A dispense mode in which liquid is dispensed while the pipetting tip does not touch the liquid surface in the container. In this mode carry over is minimal, however, droplets may remain sticking to the tip due to surface tension of the liquid.

FSE

Field Service Engineer

GLP

Good Laboratory Practice

HEPA

High-Efficiency Particulate Arrestance

IAM

Identity Access Management

Labware

Disposable item such as a microplate or a tube, able to hold liquid in an arrangement of containers or cavities; placed on a carrier for pipetting. Labware can be moved using one of the robotic arms on the instrument.

LIMS

Laboratory Information Management System

Liquid Class

Script defining a theoretical model for a particular type of liquid; includes all liquid handling parameters which are required to pipette liquids of that type. A set of properties defining a theoretical model of a liquid for a specific

pipetting device. The liquid classes are set up in the applicationl software. Standard liquid classes are identified by a generic name and include all default liquid handling parameters required to properly handle these liquids. The application specialist may define new liquid classes for optimization or liquids with special properties.

Liquid System

All instrument modules and parts which contain or directly influence liquid (tubing, diluters, valves, tips, etc.).

Location

A logical group of sites that can split a carrier or combine sites.

MCA

Multiple Channel Arm

MCA384

The MCA384 is a special liquid handling arm which can be fitted with 96 or 384 fixed tips or disposable tips (DiTis). The spacing between the tips is fixed (9 mm and/or 4.5 mm). So-called head adapters are used to adapt the pipetting head to the different tip types. The MCA384 can be optionally fitted with a gripper system (CGM, common gripper module) which can be used to pick up and move labware. The gripper module can be rotated.

MCA96

The MCA96 is a special liquid handling arm which can pick up up to 96 disposable tips (DiTis). The spacing between the tips is fixed (9 mm). The MCA96 can be optionally fitted with a gripper system which can be used to pick up and move labware. The gripper module cannot be rotated.

Method

A method is a collection (sequence) of batches. The method can be executed in a run.

Microplate

A labware in the shape of a rectangular plate of standardized size and layout, mostly comprising 96, 384 or 1536 cavities (wells).

Multi Pipetting

The pipetting mode in which a larger amount of liquid is aspirated and then several aliquots are dispensed to different positions.

Multiple Channel Arm (MCA)

Robotic arm with a multiple channel pipetting head (with 96 or 384 channels) fixed to it. All channels of the pipetting head can aspirate/dispense liquid simultaneously.

Nest Segment

A nest segment is a mount for microplates or other labware. It is placed on the instrument's deck.

Piercing

The penetration of a sealing membrane on a container, such as a microplate or sample tube, by means of a pipetting tip in order to make the liquid inside the container accessible for pipetting.

Pipetting Cycle

A sequence of aspiration and dispense steps to complete a pipetting task.

Pipetting Mode

Describes the main method by which a liquid can be distributed: either by single pipetting or multi pipetting.

Plunger

The piston in a syringe or a channel of a multiple channel pipetting head. It aspirates liquid by moving in one direction and dispenses it when moving in the opposite direction.

Position

A 3-dimensional numbering of positions within a location.

RGA

Robotic Gripper Arm

Robotic Gripper Arm (RGA)

Robotic arm which can be equipped with different types of gripper fingers to pick up, transport and place objects (plates and tubes) within the working area of the pipetting instrument.

Run

A run is the execution of a method. Only one run can be executed at a time.

Runner

Holds sample tubes or reagent troughs. The runners are placed on a deck segment and can slide along the positioning pins, e.g. for barcode identification.

Script

A script is a collection of commands to carry out the actions and interactions required for an assay protocol. More technically speaking it is a sequence of statements and can be executed in a run. Control structures can influence the order of execution of the statements.

Segment

A plate with positioning pins. It is placed on the instrument's deck and positions the runners.

Site

A continuous numbered 1-dimensional identification of a position on a carrier.

Syringe

Part of the diluter. In consists of a glass barrel and a movable plunger to aspirate/dispense the required quantity of liquid.

System Liquid

System liquid refers to a liquid which fills the liquid system and is used as wash fluid. • Standard system liquid: Deionized or distilled water with a conductivity between 0.5 µS/cm and 10 µS/ cm or an aqueous buffer solution with a conductivity of less than 500 µS/cm. NOTE: Recommendation: Add 1% isopropanol to the above. • Special system liquid: DMSO NOTE: Make sure that the system liquid container is clean. NOTE: The system liquid must be free of particles. NOTE: The system liquid must be free of air bubbles and must be at room temperature. NOTE: In order to ensure that during operation no air bubbles form in the pipetting tubing, a sufficient quantity of system liquid must circulate in the system. We recommend at least 60 ml per hour. NOTE: To reach the pipetting performance we recommend degassing the system liquid. For further information on this issue, please contact your responsible application specialist. NOTE: Any additives to the system liquid must be validated to evaluate the influence on the pipetting performance and the overall analytical process.

Tip

A needle-like device that can be mounted to a pipetting device for aspirating or dispensing liquids. The following tip types are used with Tecan instruments: • Standard tip • DiTi (disposable tip) • ActiveTip • Fixed tip • Tip block (for multipipetting heads)

Tip Retract Supervision

A special function of the capacitive liquid level detection (cLLD) which generates a message if the difference between the liquid levels measured before and after sample aspiration/dispense does not correspond to the calculated difference. This indicates that the tip may be occluded or that an object, such as a filament, is hanging from the tip.

Wash

Aspirating system liquid from the system liquid container and dispensing it through the pipetting tip into the wash position to clean the inside and the outside of the pipetting tip.

Well

One of the cavities in a microplate.

Z-dispense

The height of the point of the pipetting tip at which liquid is dispensed.

Z-max

The lowest possible position the pipetting tip is allowed to reach. During a “search liquid command” the instrument will search for liquid from Z-start down to Z-max. If the tip reaches Z-max without finding liquid, the instrument reacts according to the liquid detection error mode selected.

Z-start

The height of the pipetting tip at which the cLLD is switched on during a “search liquid command”. It is usually slightly above the rim of the liquid container.

Z-travel

The height at which the pipetting tip moves from one X/Y-position to another. Moves passing over different labware always use the highest Z-travel defined.