Software instructions | matriq AG

October 2026 | As of software version 3.2.3

Table of contents

1 Preconditions 1

2 Software layout 1

2.1 Navigation area (left sidebar) 2

2.1.1 Language Settings 3

2.2 Status bar (top bar) 3

2.3 Content area (central content area) 4

3 Operating status 4

3.1 System states in the software 4

3.2 Error messages and warnings 4

3.2.1 UI interaction with errors and warnings 5

4 Markings 5

4.1 Assigning a cavity 6

4.2 Setting the marking patterns 6

4.3 Fine-tuning the marking contrast 9

4.4 Design of Experiments (DoE) 9

4.4.1 Step 1 10

4.4.2 Step 2 10

4.4.3 Step 3 11

4.4.4 Execution 12
4.4.5 Result 13
4.5 Display of the markings 14
4.6 Exporting the markings 16
5 Login and User Management 17
5.1 Logging in as an Admin 17
5.2 Managing Users 17
5.3 Forgotten Admin PIN 18
6 Interfaces 18
6.1 Network 19
6.2 OPC UA 20
6.2.1 OPC UA client mode 20
6.2.1.1 Functional principle (example Arburg machine) 20
6.2.1.1.1 Workflow at Arburg-IMM 21
6.2.1.1.2 Configuration in the LinQ UI 21
6.2.2 OPC UA server mode (beta version) 22
6.2.2.1 Node structure 22
6.2.2.1.1 Central nodes 23
7 MQTT 23
7.1 Requirements 23
7.2 General technical specifications 24
7.3 User Interface 24
7.3.1 Production view 24

7.3.2 Mode selection 26

7.4 MQTT operating modes 29

7.4.1 MQTT server mode 29

7.4.2 MQTT client mode 30

8 Settings 31

8.1 NTP server and time zones 31

8.1.1 Procedure for setting up the time server 31

8.1.2 Error Indication 32

9 Updates to the matriq platform 32

9.1 Procedure for matriq updates 32

10 Information and status report 34

10.1 Contact information 34

10.2 Status report 34

10.3 Overview of the connected hardware 35

11 Diagnostics 35

11.1 Insert Health 35

11.2 Live Sensors 37

1 Preconditions

These instructions assume that the operating instructions have been successfully completed in accordance with matriq_DM-qode_BA1_Rev03. The device must be switched on and the matriq tablet must be connected to it. The dashboard is loaded automatically.

2 Software layout

The user interface can be used across all platforms, both on touch-based tablets and on desktop PCs with mouse and keyboard input. The dashboard can be accessed from non-matriq devices by entering the IP in the browser.

The application is divided into three central areas: Navigation area, status bar and the content area.

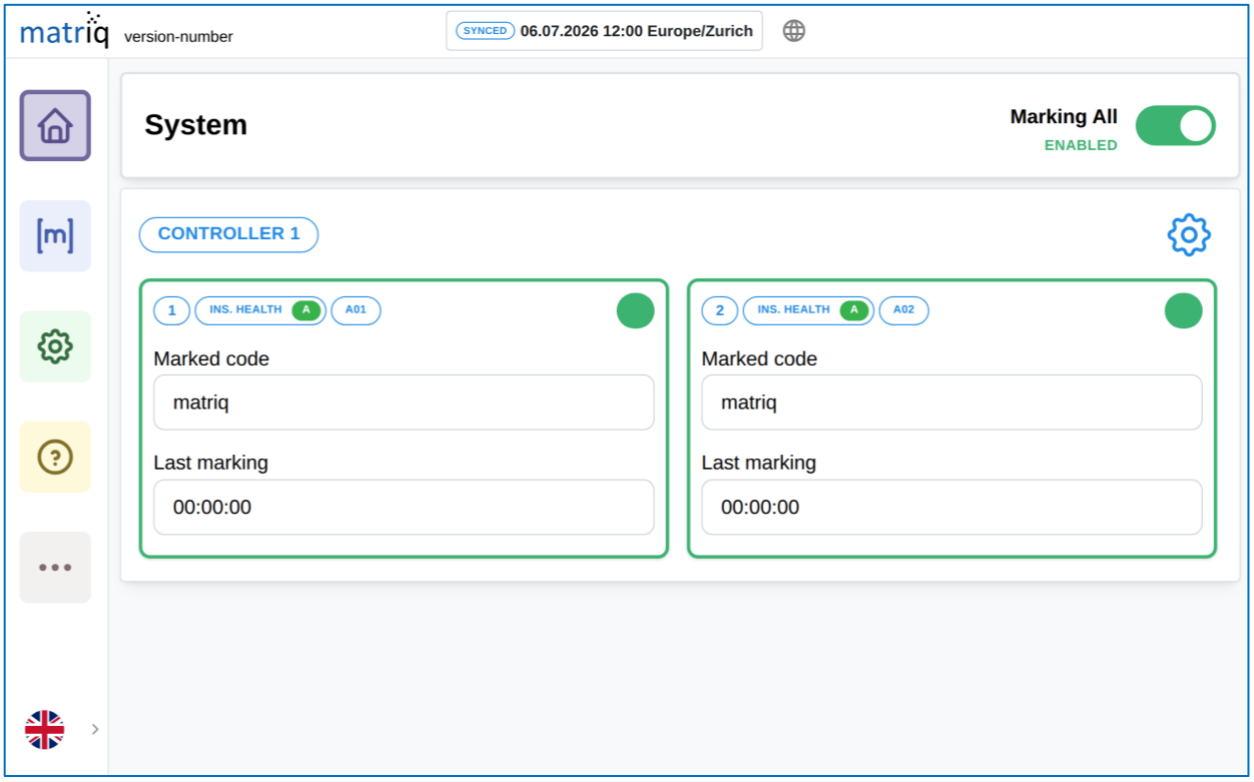

Figure 1 Dashboard display with navigation area (left), status bar (top), content area (center)

2.1 Navigation area (left sidebar)

On the left is a vertical sidebar with icons for navigation:

Home
Overview with all central information at a glance: current production status, system messages, access to marking control.

Markings
Shows a history of the most recently created Data Matrix codes including their metadata and status information.

System settings
Access to system configurations: Network and device settings, as well as customization of interfaces (e.g. OPC UA, Time Server etc.).

About
System information such as software version, current hardware configuration and connected modules.

More Options
Offers additional functionality: Diagnostics, Refresh, Screenshot and Admin login. The Image below shows the Dashboard in Admin Mode.

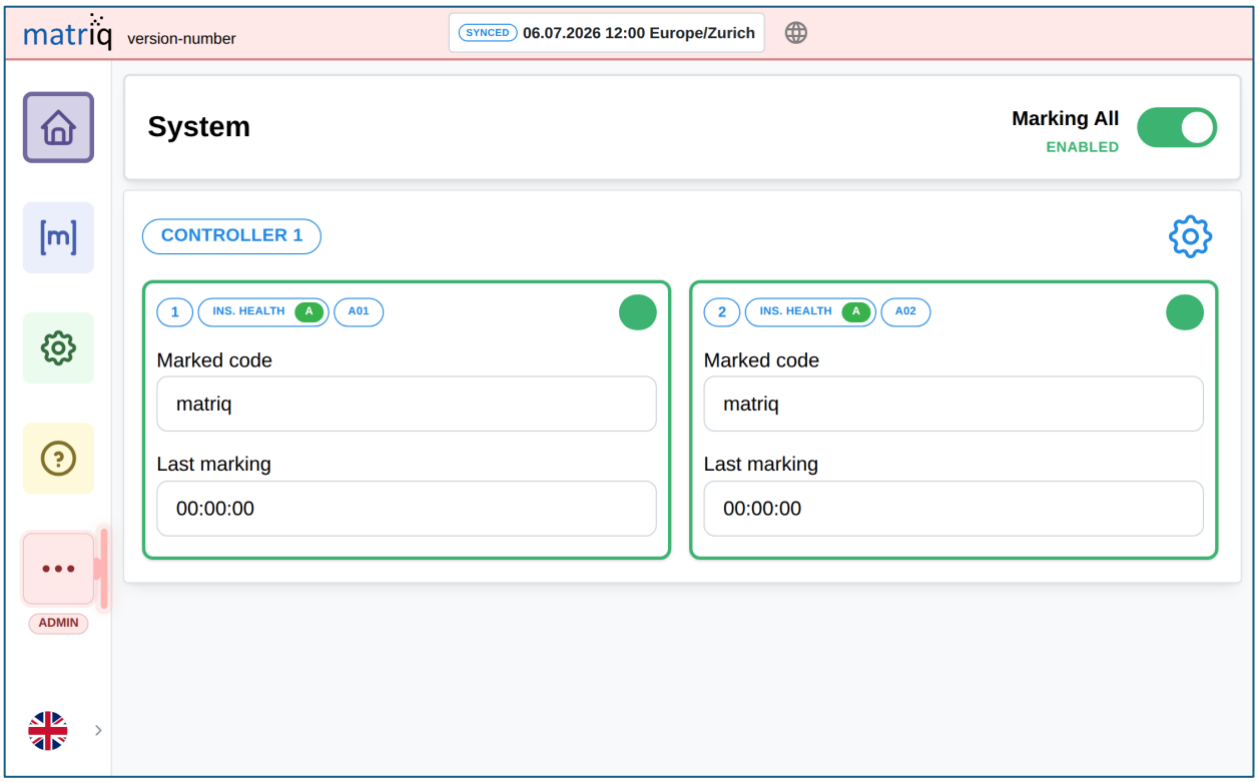

Figure 2 Active Admin mode

To exit Admin mode, click the three-dot button and select “Exit Admin Mode” from the popover menu.

2.1.1 Language Settings

The language selection is located at the bottom of the navigation area. The language can be changed via the country icon. Currently, German and English are supported.

2.2 Status bar (top bar)

There is a horizontal status bar at the top of the application:

- Centered: The current system time combined with the display of whether the time has been actively synchronized with configured time servers.
- Right-aligned: Context-dependent system status messages, such as "Restart required", "Update ready", etc.

The status bar provides continuous feedback on the state of the system.

2.3 Content area (central content area)

The main area of the user interface displays the currently selected view, depending on the selection made in the navigation menu. Typical contents are

- Overview of the connected modules
- Detailed views of individual markings
- Form masks for configuring devices or interfaces
- System logs or diagnostic information

3 Operating status

The DynamicMold® software signals different operating states of the application, both in the dashboard and on the housing of the LinQ using the status LEDs. There is also an error contact for forwarding the status to the machine. These statuses reflect both software and hardware-related system information.

3.1 System states in the software

There are several states that the system can assume:

Condition	Description	UI - Status Color	LED on the LinQ	Error Contact
Idle	The system is ready for operation but does not carry out any markings.	Orange	Blue	Active
Active	The marking process is running without errors. All systems are in normal operation.	Green	Green	Inactive
Error	A critical error has occurred. The marking process is not guaranteed.	Red	Red	Active
Boot mode	The system is in boot mode. This mode is primarily relevant for technicians.	Blue	Blue	Active

3.2 Error messages and warnings

The controller reports both error messages and warning messages which are displayed in the UI dashboard:

- Warnings: Have no direct influence on the system status.
- Errors: Lead directly to the error state being triggered. In this case, the error contact is activated.

3.2.1 UI interaction with errors and warnings

The tablet dashboard always shows which channel or module has reported a warning or error. When the affected module is selected, an information window opens containing detailed information on the respective message, including classification (warning/error), description and possible recommendations for action (such as resetting the module).

4 Markings

The system can be equipped with up to four channels with one insert each.

The individual channels are configured using:

- Selection of the marking content (e.g. alphanumeric or coded content).
- Adaptation of the marking behavior depending on the material used, and the definition of the marking mode per module.

The marking function must be active:

- The controller is correctly connected to the LinQ.
- At least one insert is connected to a channel of the controller.
- The relevant channel has marking set to "ENABLED" and is in an error-free state.

As soon as all requirements are met, the marking process runs completely autonomously:

- The system recognizes the incoming plastic at the insert (via sensor).
- The marking is triggered automatically, without manual intervention.

All active marking processes can be viewed in the UI dashboard of the matriq platform. The following information is displayed for each active module:

- The current status.
- Visual feedback on the last successful marking.
- Status of the connected insert.
- Possible warnings or error messages.

The markings are created continuously as long as the machine is producing components, and marking is set to enabled (“ENABLED”) for the corresponding channel.

4.1 Assigning a cavity

Each channel can be assigned a cavity number. This allows markings to be associated with the corresponding cavity in the mold.

In the Dashboard, select the desired channel and enter the cavity number in the “Cavity” field at the top right.

4.2 Setting the marking patterns

The matriq platform enables the definition and configuration of individual marking patterns that are used during the injection molding marking process. Depending on the application, different pattern types (code sources) are available to design the marking content dynamically or in a fixed way.

To configure these settings, the corresponding channel must be selected in the UI dashboard. This opens a pop-up with the "Marking" pattern configuration section preselected. To change the settings, the marking function must be disabled. If a warning or error is present, the detailed display for the warning or error is shown as the first window.

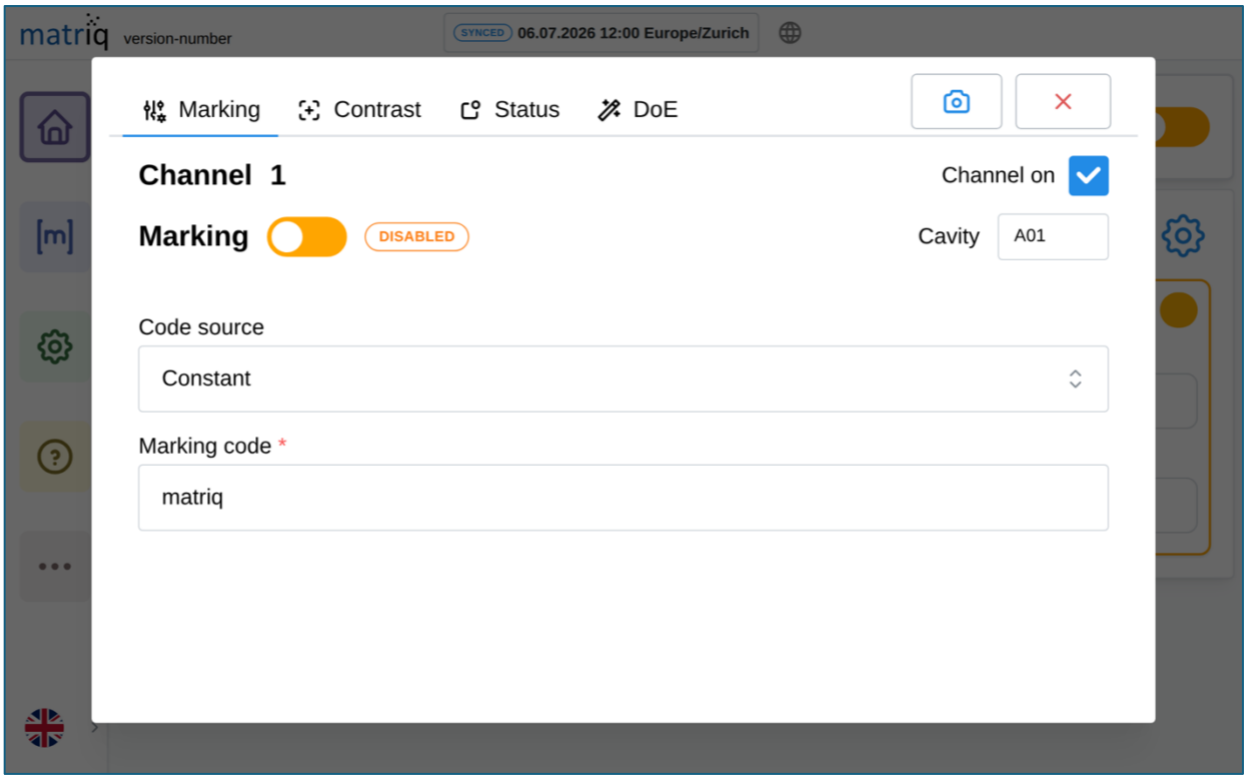

Figure 3 Marking Configuration Popup

Available marking patterns:

- **Constant**: a constant, alphanumeric code (characters A-Z, a-z, 0-9)
- **Count Up**: A ten-digit number which is counted up automatically; starting value and increment can be set. After a power interruption to the LinQ, five serial numbers per channel are skipped when marking resumes to prevent duplicates.
- **Constant + Count Up**: A combination of a fixed prefix (consisting of two alphanumeric characters) and a counter with six digits. Example: AB000001, AB000002, AB000003 etc. After a power interruption to the LinQ, five serial numbers per channel are skipped when marking resumes to prevent duplicates.
- **Unix timestamp**: The current timestamp of the marker in UTC time. The display is in the format of a 10-digit Unix timestamp (seconds since 01.01.1970, 00:00 UTC). Example: 1747778400 corresponds to May 21, 2025, 00:00:00 UTC.
  Note: The accuracy of the timestamp depends directly on the synchronization with a reliable time server (see chapter *8.1 NTP server and time zones*).
- **Random**: A quasi-random ten-digit number is marked for each component. This mode is primarily suitable for test purposes and is not recommended for productive series production.
- **MQTT Service**: Marks serial numbers received via the MQTT interface. Available numbers are retrieved by the channels. Detailed instructions can be found in chapter *7 MQTT*.

All configuration changes are saved automatically, provided that no errors are present. The marking process starts as soon as “Marking” is set to ENABLED, which activates the configured marking logic for the channel. This is shown in the image below.

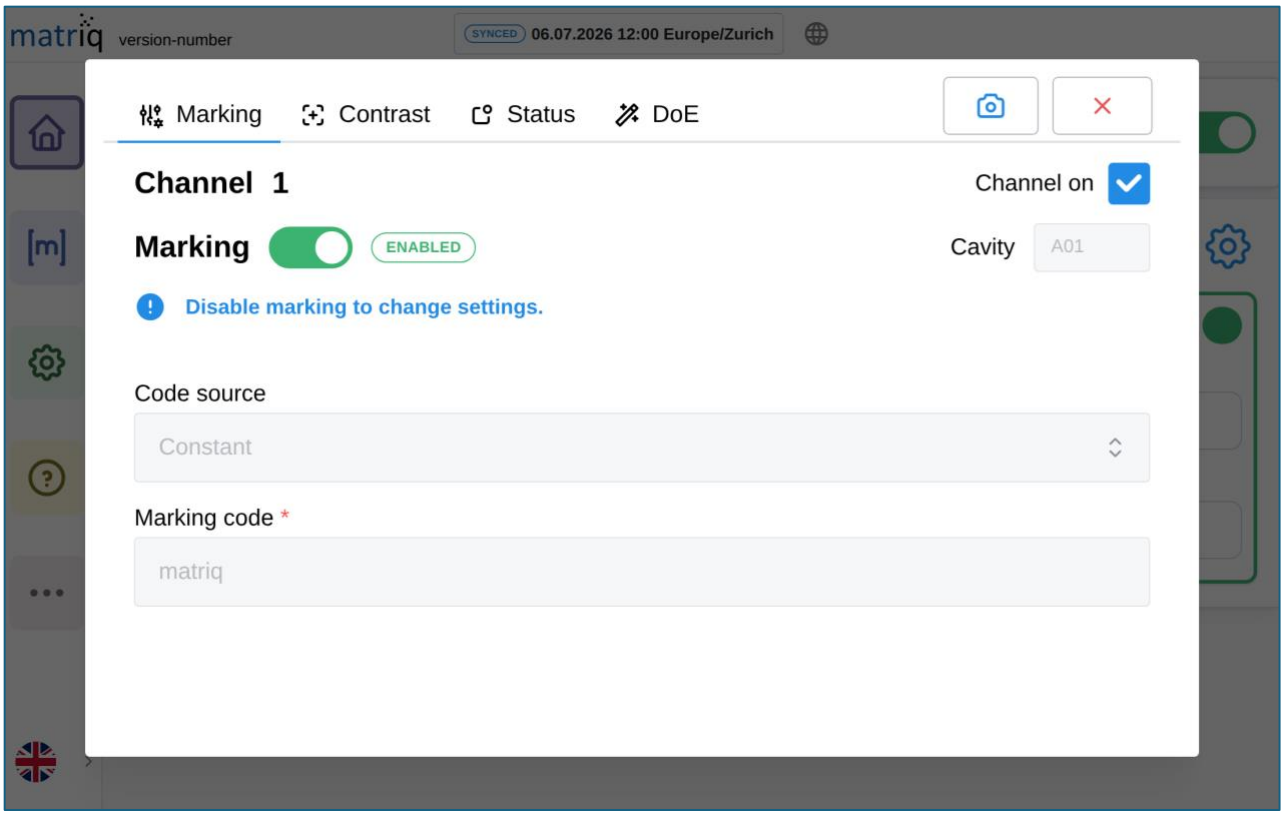

Figure 4 Channel status ENABLE view

4.3 Fine-tuning the marking contrast

Depending on the material and component shape, different heating parameters must be set to achieve the best marking contrast. These parameters can be adjusted both when marking is enabled and when it is disabled. For guidance on determining suitable parameter combinations, see chapter *4.4 Design of Experiments (DoE)*.

Manual adjustments are done in the channel configuration view under “Contrast”:

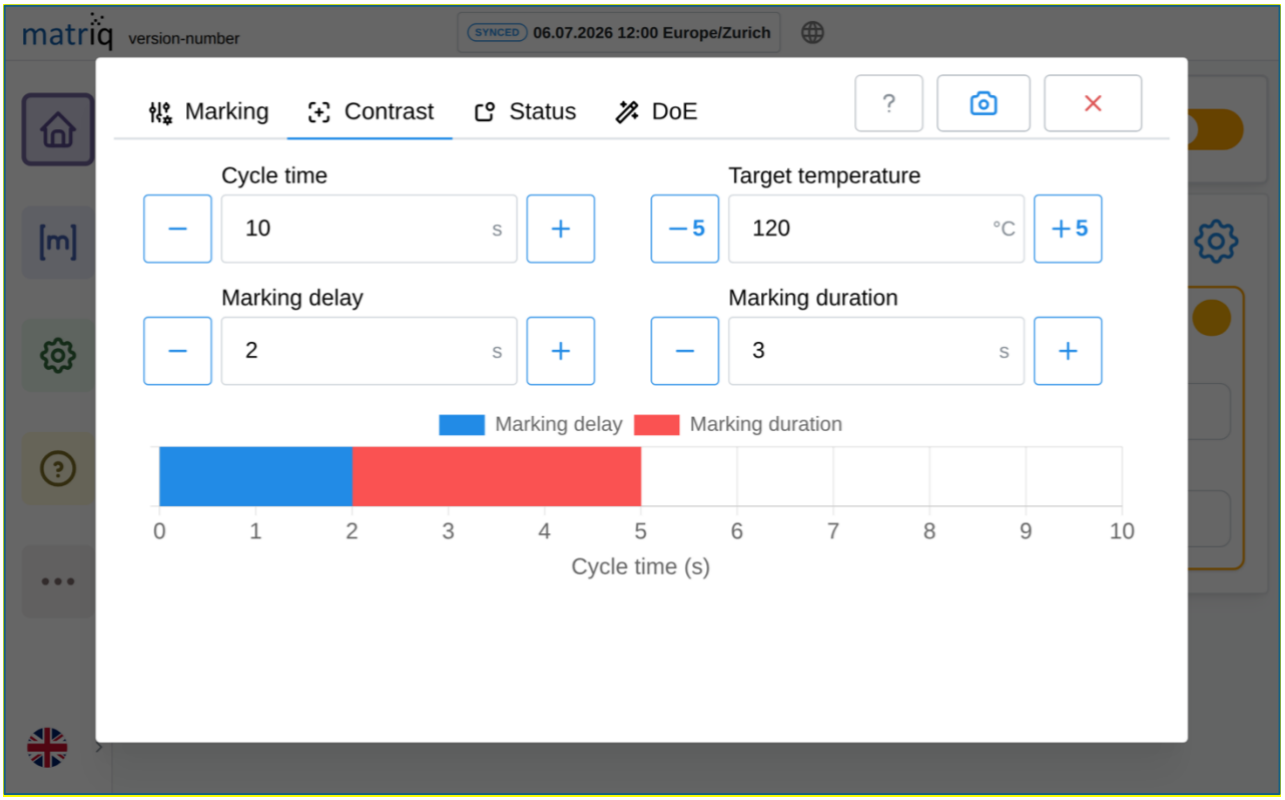

Figure 5 Contrast settings pop-up

- Cycle time: Total duration of an injection molding cycle, so that the marking process stops before the next component is produced (typically 5-20s).
- Marking delay: Delay between material detection (trigger) and start of marking (typically 0-10s).
- Marking duration: Time window of the marking process.
- Target temperature: Target temperature insert (typically 90-150°C).

Changes are saved automatically and applied from the next marking onward.

4.4 Design of Experiments (DoE)

The Design of Experiments (DoE) functionality provides a structured approach for identifying the optimal parameter combination for a specific polymer type. Those three

parameters are: Marking Duration, Marking Delay and Target Temperature. The process can be started from the channel view under the “DoE” tab and contains three steps.

4.4.1 Step 1

To begin, a preset value must be selected. It serves as the starting point for the parameter optimization. The parameter values suggested in the next step are derived from the selected preset. By default, the current configuration of Channel 1 is used.

The cycle time of the corresponding machine and a code prefix must also be specified. The prefix is used as the first part of the marking code. The complete marking code is generated from the prefix, the channel number, and a consecutive number starting at 001.

Finally, the channels to be included in the Design of Experiments run can be selected. The same DoE configuration is applied to all selected channels and inserts.

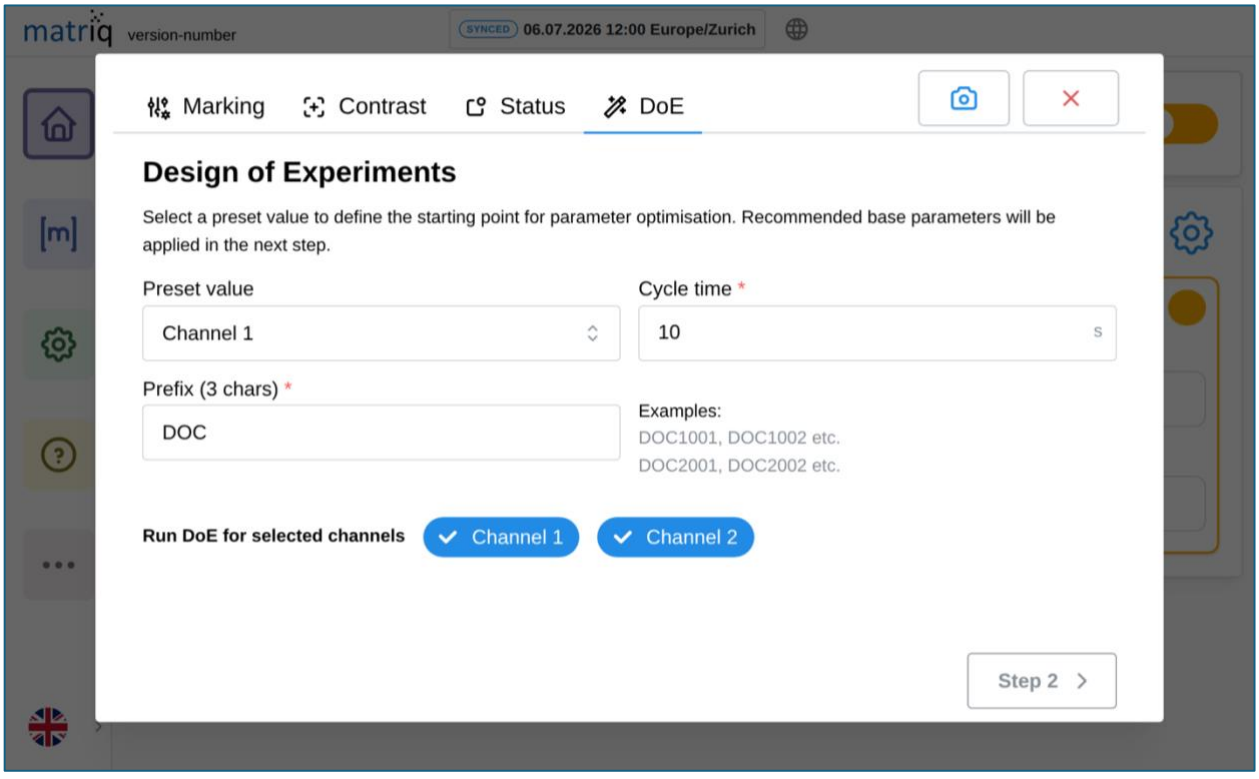

Figure 6 DoE configuration - step 1

4.4.2 Step 2

In the second step, the parameter values to be tested during the experiment are defined.

For each parameter, the number of values to be tested can be selected between one and seven. The values are generated automatically based on a configurable middle value and

step size. These settings can be adjusted individually for Marking Duration, Marking Delay, and Target Temperature.

All possible parameter combinations are automatically generated from the selected parameter values.

The number of iterations can also be configured. An iteration value greater than one causes each parameter combination to be marked multiple times, allowing the consistency and repeatability of the marking results to be evaluated.

Based on the selected parameter values, iterations, and machine cycle time, the system displays both the total number of shots and the estimated execution time. A maximum of 999 shots is permitted for a single DoE run.

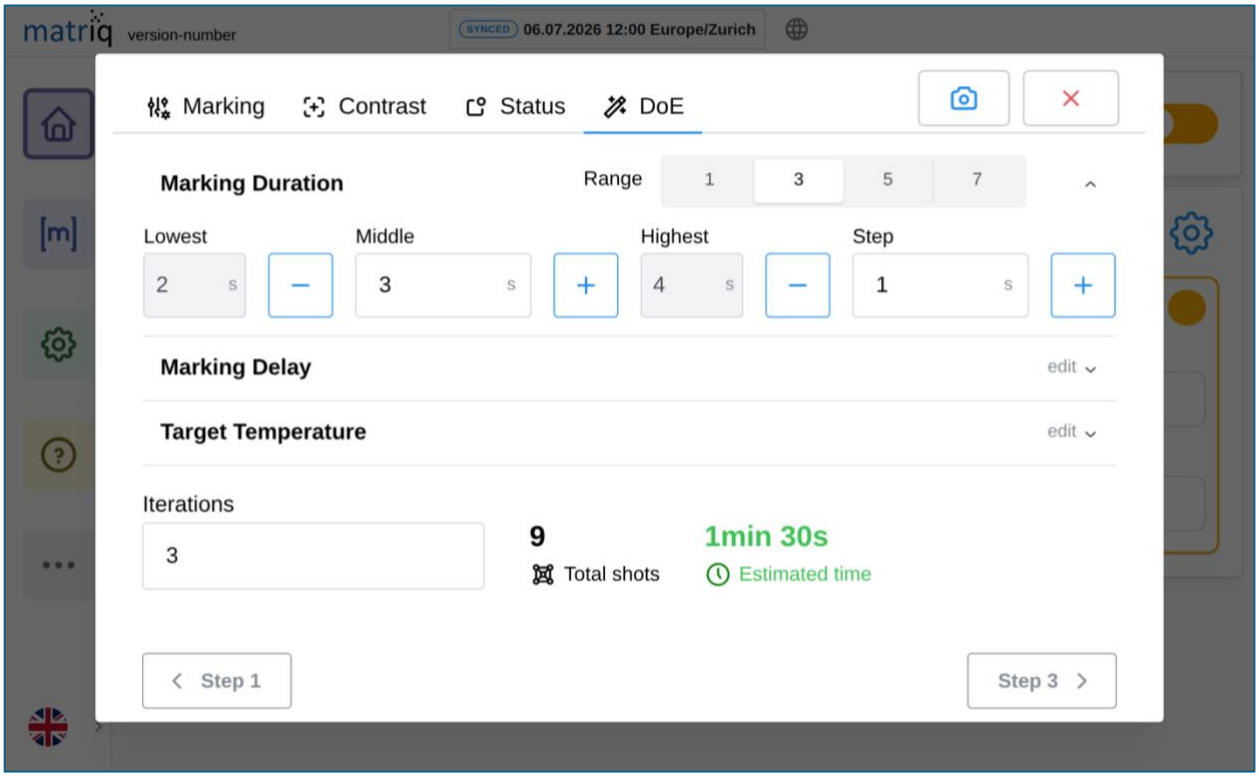

Figure 7 DoE configuration - step 2

4.4.3 Step 3

The final step provides an overview of all selected settings, including the configured parameter ranges, code format, selected channels, total number of shots, and estimated execution time. This overview allows the configuration to be reviewed before execution.

Once the configuration has been verified, the DoE run can be started using the Start button. The system then begins the automated marking process using the defined

parameter combinations.

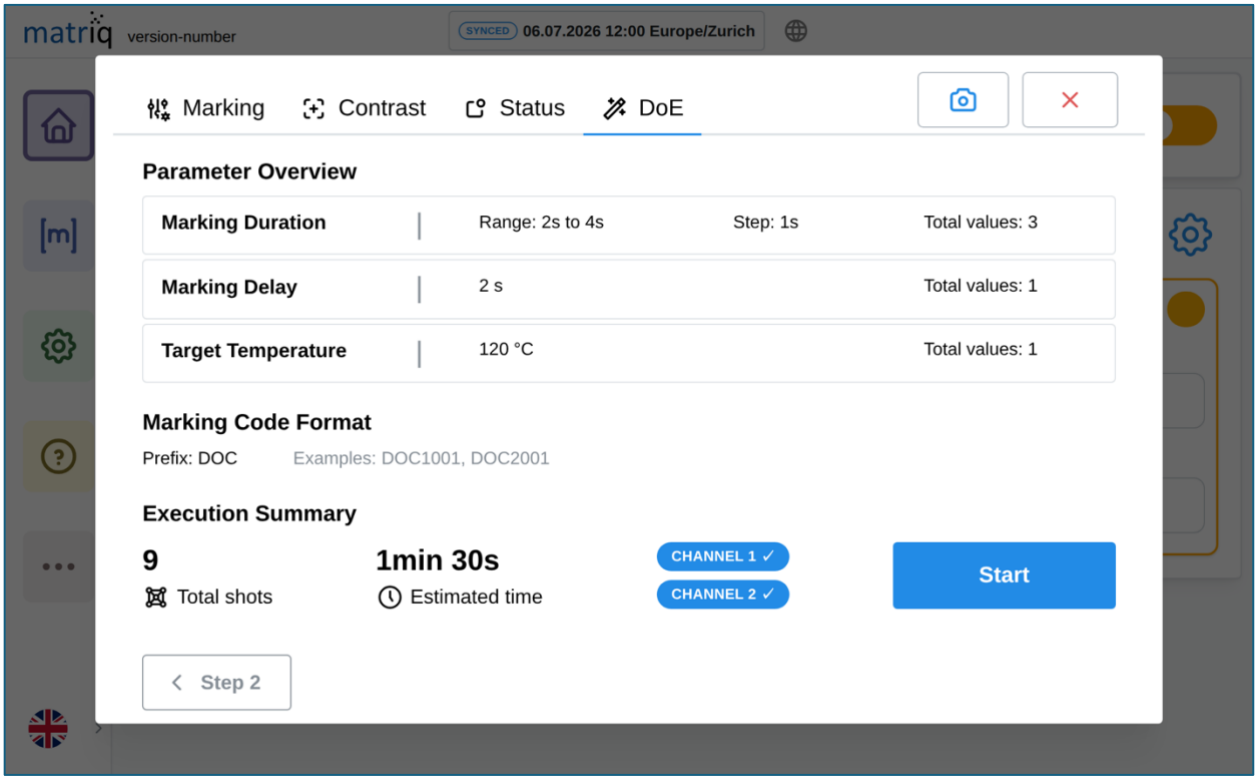

Figure 8 DoE configuration - step 3

4.4.4 Execution

During execution, the current progress of the experiment is displayed by a progress bar.

The experiment can be aborted at any time. All parameter combinations remain available in the results view. Combinations that were not physically marked before the abortion are displayed as not completed.

Once aborted, a DoE run cannot be resumed and must be restarted from the beginning.

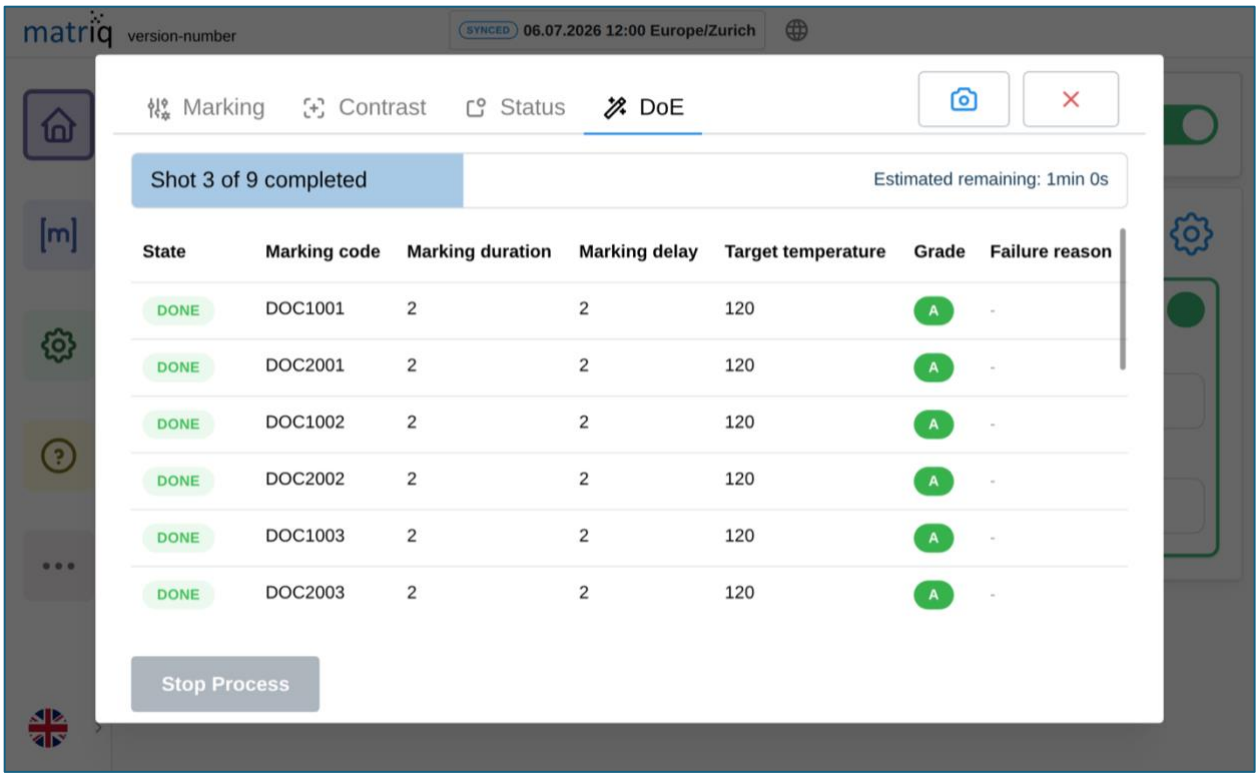

Figure 9 DoE execution

4.4.5 Result

Once the DoE run has finished or has been aborted, the results view is displayed. In addition to the parameter values, each marking shows its code grade and current status. Any errors that occurred during processing are also indicated in the status column.

The results table provides an overview of all tested parameter combinations. The marked parts must then be evaluated manually to identify the most suitable marking configuration for the specific application.

Once a suitable marking configuration has been identified, its contrast settings can be applied directly. To do so, select the desired marking for each channel and click Apply selected.

The complete results table, including all parameter combinations and the configuration of the corresponding DoE run, can be exported as a CSV file using the Export results button.

Once the evaluation is complete, the session can be closed using the Finish session button.

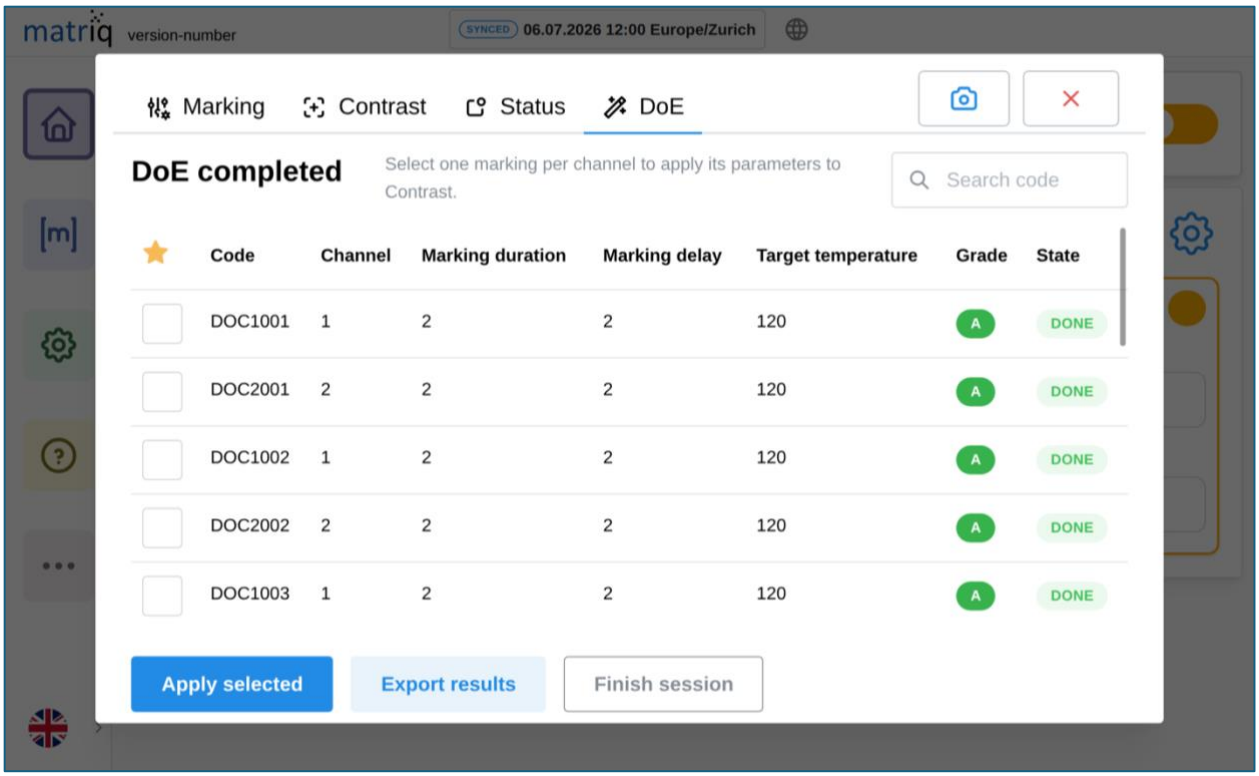

Figure 10 DoE results

All markings generated as part of the DoE run are also available in the Markings table alongside regular markings.

4.5 Display of the markings

The matriq platform saves all created markings continuously in a local database on the LinQ system. This memory is continuously filled with new marking data as long as sufficient memory space is available. When the memory is full, the oldest marking data is overwritten.

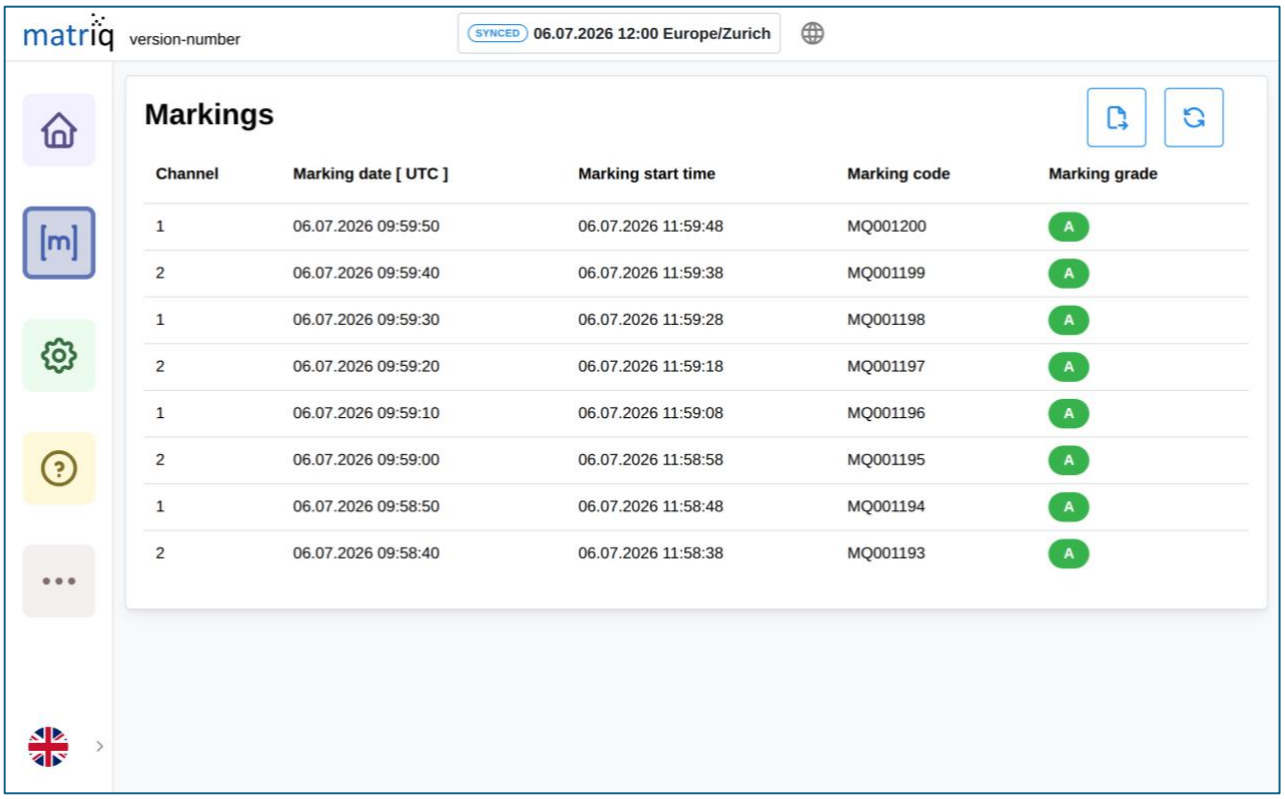

Figure 11 View of the last 15 markers

The overview table contains the following columns:

- Channel: Indicates which module or insert generated the respective marking.
- Marking Date [UTC]: Timestamp of the marking in UTC time, for cross-system synchronization.
- Marking Start Time: Timestamp of the starting point of the marking in the injection mold marking process.
- Marking code: The marked Data Matrix code or serial value.
- Marking grade: Evaluation of the error correction used on a scale from A (optimal) to F (unsatisfactory). Is calculated from the distribution of existing defective heater pixels for the respective code. However, it does not provide any information about the marking quality in terms of legibility (e. g. contrast).

4.6 Exporting the markings

The export button (left button) is located at the top right of the marking view. Markers can be exported as CSV or TXT. Three different time ranges are available: the last 100 markings, markings from the last 24 hours or a user-defined time period. In addition, the level of detail of the exported data can be selected: Basic or Expert.

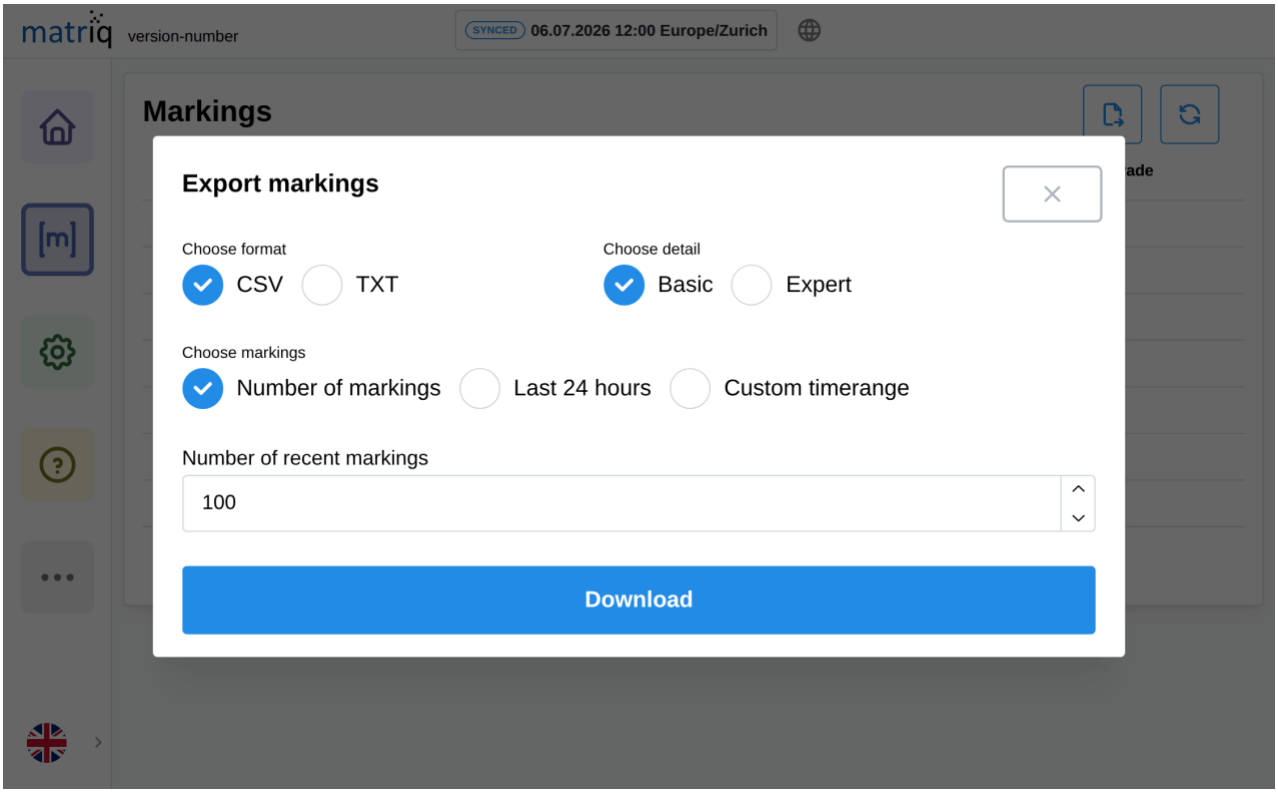

Figure 12 Export options for markings

In Expert mode, all available attributes of the markings are exported. This includes general marking information as well as marking parameters, including parameters that are not configurable through the tablet user interface.

In Basic mode, only parameters that can be configured via the user interface are included. These are listed and briefly described below:

- requested_code: Originally requested code
- code_grade: Evaluation of the error correction used (A - optimal ... F - unsatisfactory)
- marking_time: Timestamp of the marking created
- channel_id: ID of the associated module
- trigger_to_marking_delay: Marking delay in ms
- marking_duration: Marking duration in ms

- target_temperature_phase_2: Target temperature in °C

5 Login and User Management

The tablet connected to the LinQ via Ethernet is automatically logged in as an **Operator**. An **Admin** login is required to change protected system settings.

5.1 Logging in as an Admin

- Open the three-dot menu in the left sidebar and select “Sign in”.
- Select an Admin account. When logging in for the first time, select the default “customer admin” account.
- Enter the six-digit PIN. The default PIN for the “customer admin” account corresponds to the last six digits of the LinQ serial number, unless it has already been changed.

5.2 Managing Users

Once logged in as an Admin, accounts can be created, edited or deleted under “Settings” → “Users”. Their PINs can also be changed here. The account you are currently logged in to and the “customer admin” account cannot be deleted.

The “Allow device reset” option enables or disables resetting the LinQ from the login screen. If this option is disabled and all Admin credentials are lost, the LinQ must be returned to matriq for a reset.

The user management interface is shown below.

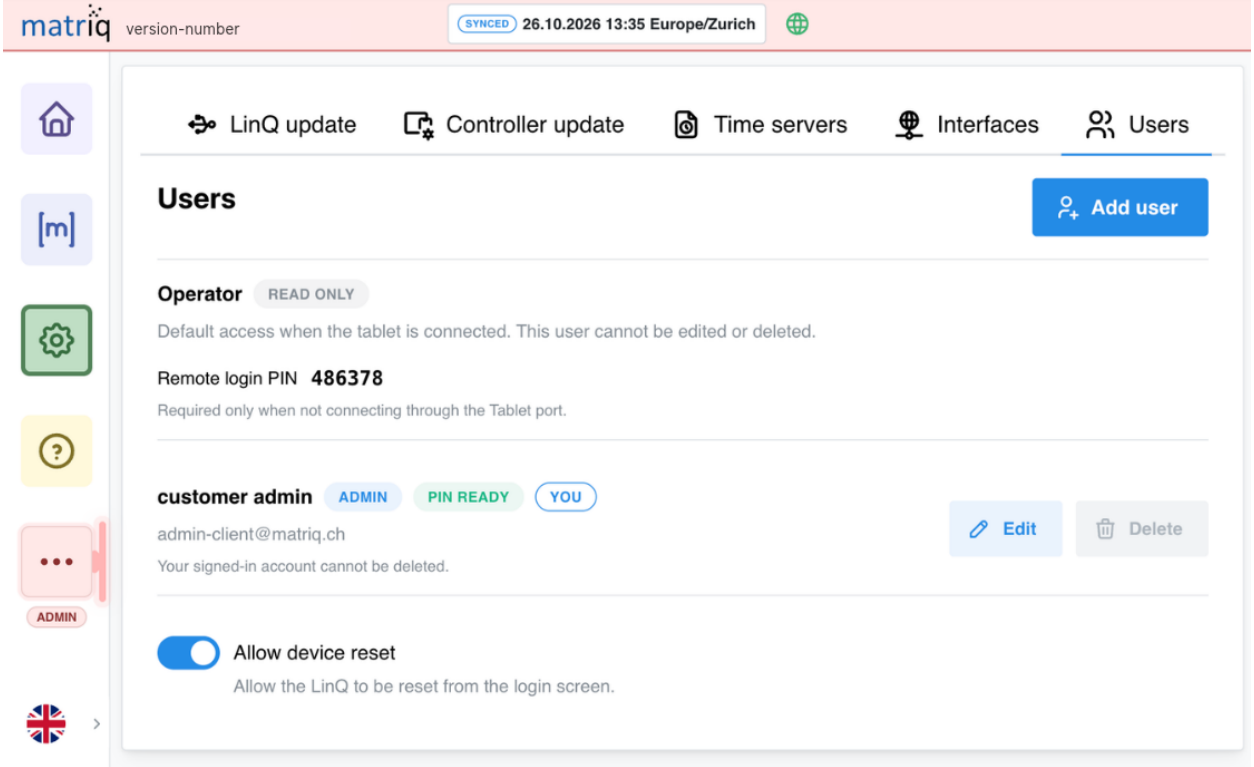

Figure 13 User management

5.3 Forgotten Admin PIN

If the PINs for all Admin accounts have been forgotten, access can be restored by resetting the LinQ. The reset button is located in the bottom-right corner of the standard login screen. All modules must be disabled, no error condition may be present and the “Allow device reset” option must be enabled.

Warning: Resetting permanently deletes all stored data and settings, including all marking data and user accounts. Pressing the reset button displays detailed information and instructions before the reset is confirmed.

6 Interfaces

This chapter describes the two interfaces “Network” and OPC UA” and their functionalities. The MQTT interface is described separately in Chapter *7 MQTT*.

All functionalities described in this chapter can only be modified by users with the “Admin” role (See *5.1 Logging in as an Admin*). Users with the Operator role have read-only access and cannot make any changes.

6.1 Network

The matriq platform enables flexible network configuration to allow access to external systems and integration into existing infrastructures. All settings are made directly via the UI and can be adapted to specific network requirements.

To do this, navigate to the "Settings" page in the menu on the left and select "Interfaces". The network settings only relate to the LAN output. The network settings of the "Tablet" output are static and cannot be configured.

- DHCP: If the checkbox is activated, the IP address is automatically obtained from the network. This requires a correctly configured DHCP server in the same network segment.
- The LinQ MAC address is available via the info button associated with the IP address.
- Gateway, subnet mask and DNS: These parameters can be set manually if required, for example to connect to specific subnets or DNS servers.

6.2 OPC UA

The client-server URL can be entered individually for access to external OPC UA servers. In addition, the username and password for establishing the connection can be configured.

The OPC UA interface of the matriq platform has been successfully tested with Arburg brand machines. However, for smooth integration with other machine manufacturers, it may be necessary to make device-specific adaptations. Such manufacturer-specific extensions are maintained as customized OPC UA configurations within the matriq software and can be distributed and imported via the software update system if required (see Software updates).

The matriq platform supports two operating modes for OPC UA, which can be used depending on the application, client or server mode.

6.2.1 OPC UA client mode

In client mode, the platform actively connects to external OPC UA servers (e.g. injection molding machines, MES systems or PLC controllers) and reads the data they provide. In addition, the matriq client can also write back marking data, e.g:

- Current Data Matrix code in the machine's production log.
- Status information on the marking process.
- Marking parameters.

6.2.1.1 Functional principle (example Arburg machine)

The LinQ controller connects as an OPC UA client to the OPC UA server of the injection molding machine (IMM) to write the individual Data Matrix code to the machine protocol before each injection cycle. The prerequisite is that LinQ and IMM are in the same network segment and the IP address of the machine is known. The IMM must allow external fields to be written. On Arburg machines, this is done via the External Part ID field (f96358).

Important: As each machine (even identical models) has its own OPC UA tree, the implementation can vary from machine to machine.

6.2.1.1.1 Workflow at Arburg-IMM

Step	**Action at the IMM**	**Action on the LinQ**
1	Set the External Part ID field on the control panel to external	•
2	Select start trigger (see below)	Adapt corresponding ENV variables or load specific implementation from matriq via USB memory.
3	Call the SetExternalPartID method	Automatically triggered by the client as soon as the trigger signal is received
4	Include field f96358 in the production log	-

The SetExternalPartID method expects two arguments:

- station_number (UInt32) - for standard machines 1
- external_part_id (string, under 256 characters) with the desired marker

Trigger variants

Variant	OPC-UA field	Time	Remark
1 (Standard)	Start Cycle ns= 2;i=64572	Approx. 1s before the start of the cycle	Works on most Arburg machines
2 (fallback)	F14001-Load ns= 2;i=245533	End of cycle / immediately before new start	Recommended if Start Cycle is not active

Both variants must not be active at the same time.

6.2.1.1.2 Configuration in the LinQ UI

1. Settings → Network | Open OPC UA
2. Activate OPC UA Client (checkbox selection)
3. Enter server URL, user name "host_computer" and password "dummy_password"
4. Save

Note: The Arburg OPC UA server accepts a maximum of **3** parallel clients. If the slots of other OPC UA clients are already occupied, the connection with the matriq LinQ client will not be established.

6.2.2 OPC UA server mode (beta version)

In server mode, the matriq platform itself **provides** an OPC UA interface on port **4840** and offers other systems (e.g. host computers or QA tools) access to marking-relevant data. This includes

- Recently generated marking codes
- Status of individual modules
- Time stamp, error or warning messages
- Diagnostic and configuration information

The server can be reached via the configured IP address of the device and the standardized OPC UA port **4840**.

To establish a connection using an OPC UA client software, the following steps are required:

- In the Security Mode section, select “Sign&Encrypt”
- In the Security Policy section, select “Basic256Sha256”
- Confirm with OK

In the next step (the exact interface may vary depending on the software):

- Set User Authentication to “Anonymous”
- Confirm with OK

A certificate request will usually appear:

- Permanently accept the offered certificate

After a short waiting period, the connection will be established, and access to the device’s OPC UA interface will be possible.

6.2.2.1 Node structure

The server follows a fixed NodeId division in namespace 2.

Type	ID area	Example enum
Objects	10000-19999	matriqLinq (= root)
Variables	30000-39999	systemStatus, lastMarking
Methods	50000-59999	requestAllPattern
Events	derived	markingEvent, allPatternEvent

The complete mapping is embedded in the OPC-UA component of the matriq software platform.

6.2.2.1.1 Central nodes

Node	Type	Purpose	Data type
matriqLinq	Object	Root node	-
operation/systemStatus	Variable	Operating status (RUNNING, ERROR, IDLE, NOT_CONNECTED)	Enum
marking/markingEvent	Event	Contains last marker as JSON CodeInfoType	-
pattern/requestAllPattern	Method	Provides all currently valid marking patterns	-
pattern/allPatternEvent	Event	Result of requestAllPattern (AllPatternType)	-
version/softwareVersion	Variable	OPC UA server version	String

7 MQTT

The matriq platform supports the exchange of marking data via MQTT (Message Queuing Telemetry Transport).

The MQTT interface is primarily used for the transmission of serial numbers that should be marked. In addition, it supports feedback on the current usage of serial numbers, the number of remaining entries in the buffer, resetting the internal number storage, as well as monitoring device startup events and error states.

There are two different approaches to use this functionality: MQTT server mode and MQTT client mode.

7.1 Requirements

The following system requirements must be met for MQTT-based serial number transmission:

- matriq software version **3.1.4** or higher
- Yocto version **2.2.55** or higher

In addition to that, the implementation of an external system to provide serial numbers must be completed.

7.2 General technical specifications

This section describes the general technical characteristics of the MQTT communication used by the matriq platform.

- All MQTT payloads are sent as JSON objects encoded in UTF-8.
- All timestamps will be provided as Unix Epoch timestamp in seconds. This is also the only format accepted by the matriq system.
- To establish the connection, the external system must be located within the same network as the LinQ device and be able to reach it.
- Communication takes place via the standard mqtt://, mqtts:// or wss:// protocols.

7.3 User Interface

This chapter provides an overview of the MQTT-related user interface. It describes the activation of the functionality as well as relevant configuration aspects.

7.3.1 Production view

To use the MQTT functionality, MQTT must be selected as the current pattern. This is done by selecting the corresponding channel in the dashboard and choosing “MQTT Service” from the code source drop-down menu. All displayed fields are read-only. They provide an overview of the current MQTT connection state and the serial number processing status during operation.

The connection to the MQTT broker depends on the selected MQTT mode. Once the connection has been successfully established, the connection status is shown as “Connected”. Additional connection details can be viewed via the information icon.

As soon as the matriq system receives a batch of serial numbers, the displayed values are updated and show the number of available codes currently stored on the device. When marking is enabled and the channel starts processing markings, the “Next marking“ field is continuously updated. In addition, the “Marked codes” field displays the number of serial numbers that have already been marked by this channel.

If a power failure, device restart or change of code source occurs during marking, unconfirmed serial numbers are marked as unusable. The number is displayed in the “unavailable codes” field. This prevents serial numbers that may already have been used from being assigned again.

In the rare case that no serial numbers are available for marking, the default behavior of the matriq system is to mark Unix timestamps instead. If this behavior is not desired, it can be disabled via the corresponding switch in the MQTT configuration.

An example of this view is shown below.

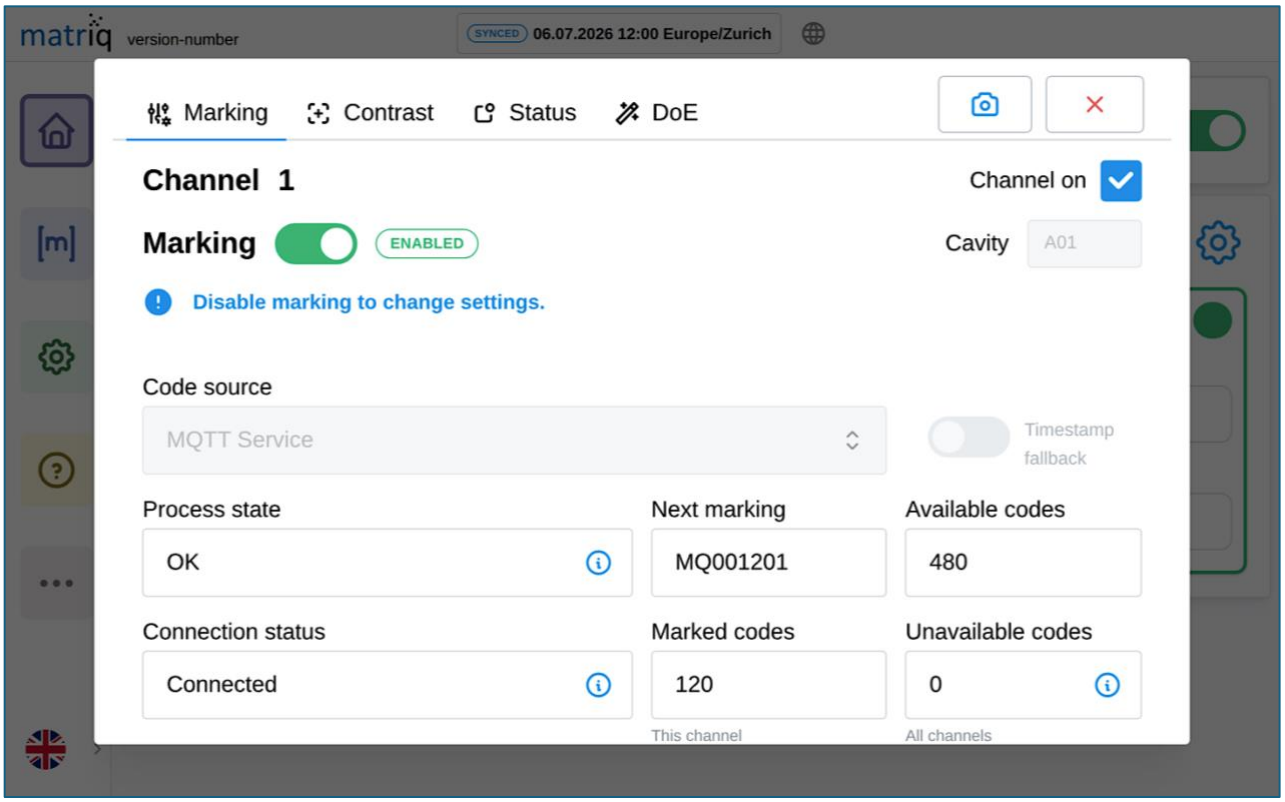

Figure 14 MQTT service overview

7.3.2 Mode selection

Generally, MQTT server mode is used as the default mode. To switch between server and client mode, you must be logged in with an Admin account, as users with the Operator role have read-only access and cannot modify these settings. To log in as an Admin, see *5.1 Logging in as an Admin*.

After logging in, navigate to Settings → Interfaces → MQTT. There, a checkbox can be selected to enable MQTT client mode, see *Figure 15*.

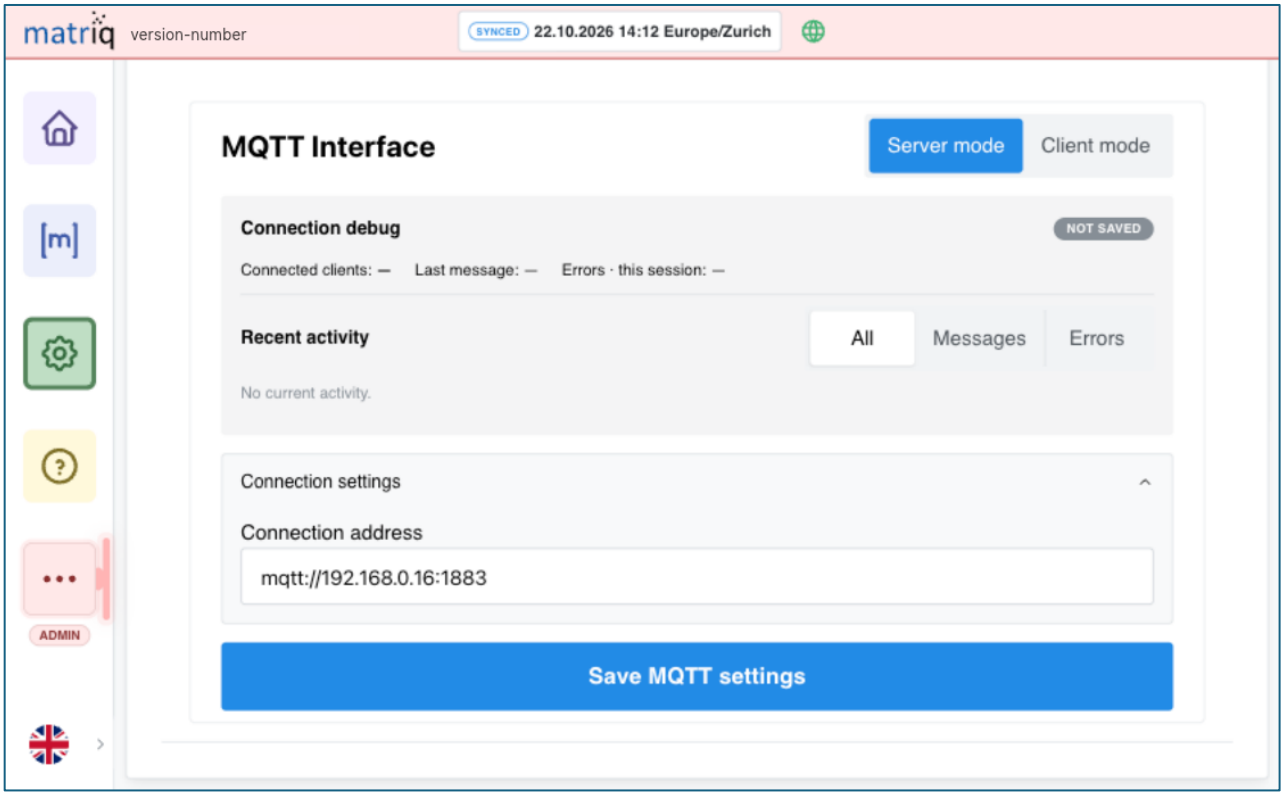

Figure 15 Switch between MQTT server and client mode

To establish a connection to the external MQTT broker, a primary broker URL (add up to 5 different URLs), caller ID, username, and password must be configured. More detailed information about these parameters can be found in Chapter *7.4.2 MQTT client* mode.

Once all parameters have been configured and saved, the matriq system tests the connection and displays the result directly inside the MQTT client configuration window, as shown in the example below for a successful connection.

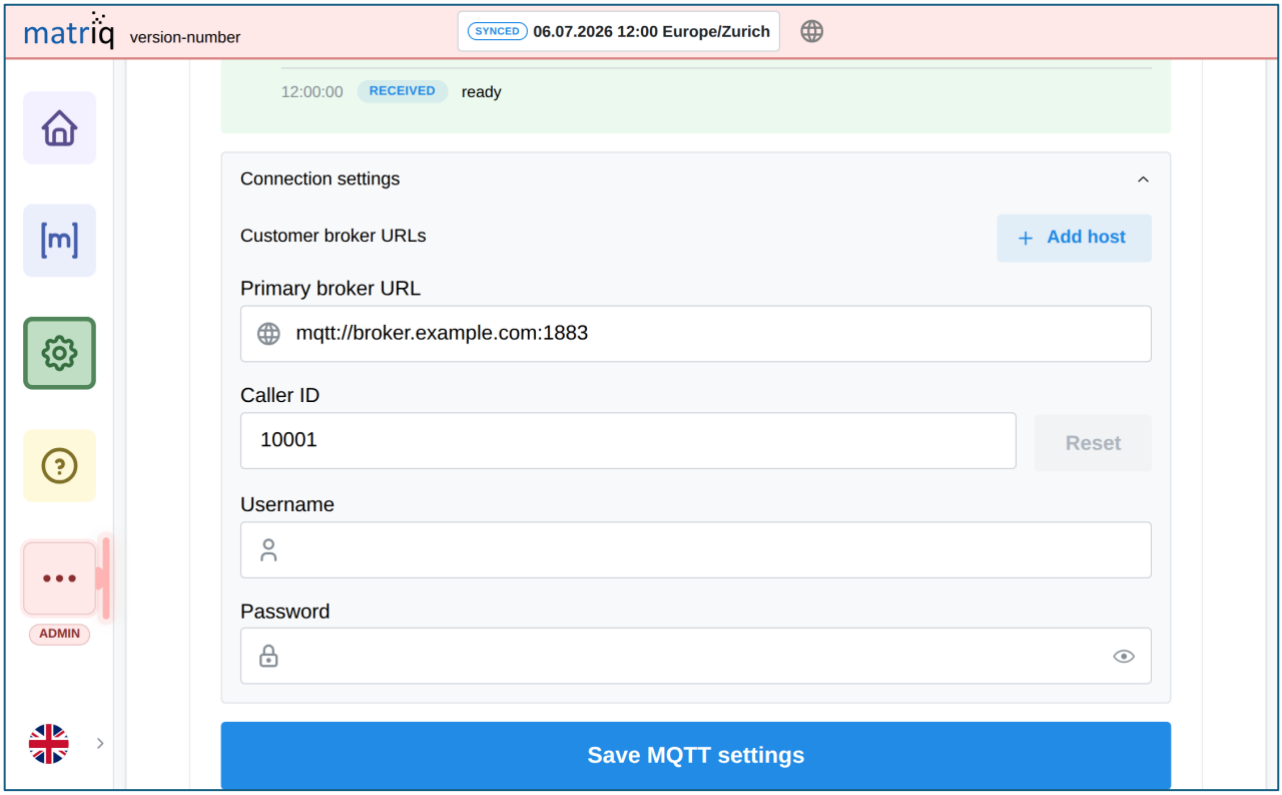

Figure 16 Successful MQTT connection test

When establishing a connection to an encrypted and previously unknown host for the first time, a trusted certificate is required. In this case, the connection test provides the option to review the certificate, see *Figure 17*.

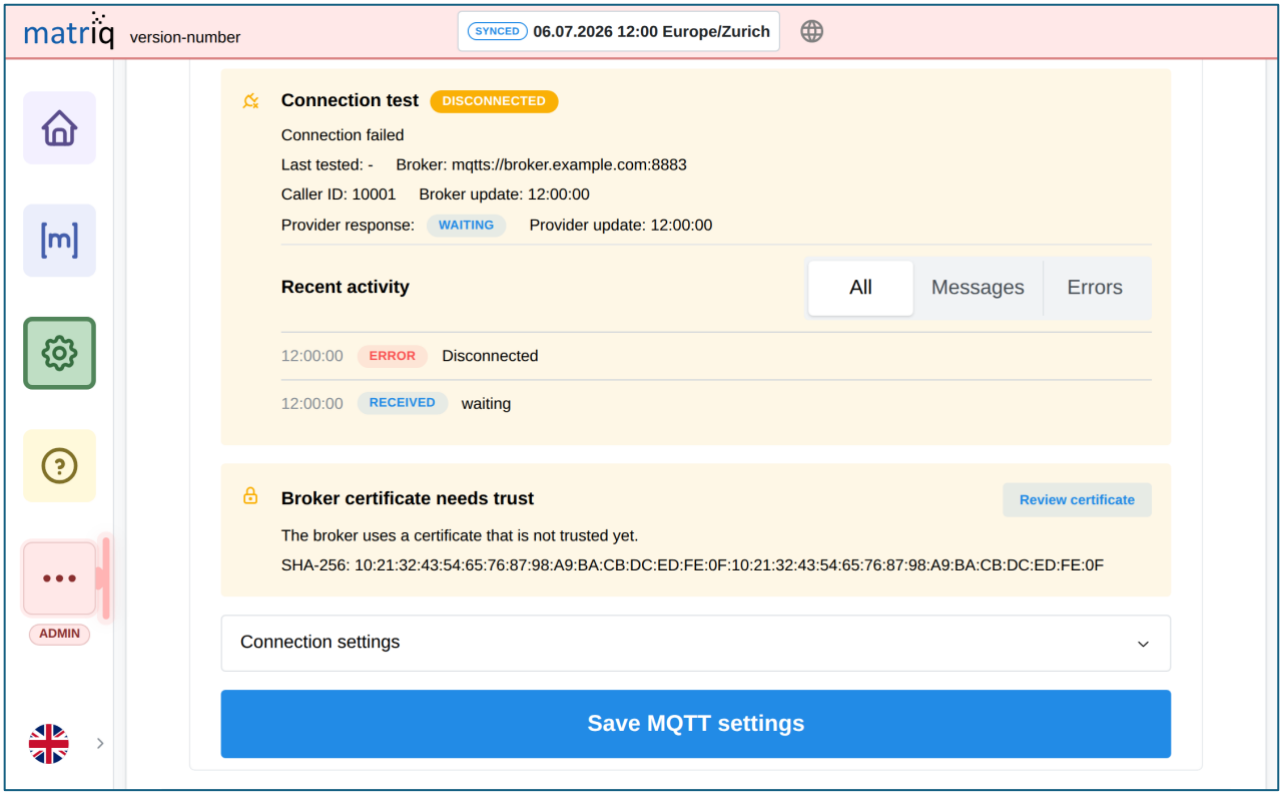

Figure 17 MQTT connection test detects an untrusted certificate

After selecting the “Review certificate” button, an overlay displaying the certificate details appears, as shown in *Figure 18*. The certificate can then be trusted, and the connection test is executed again automatically.

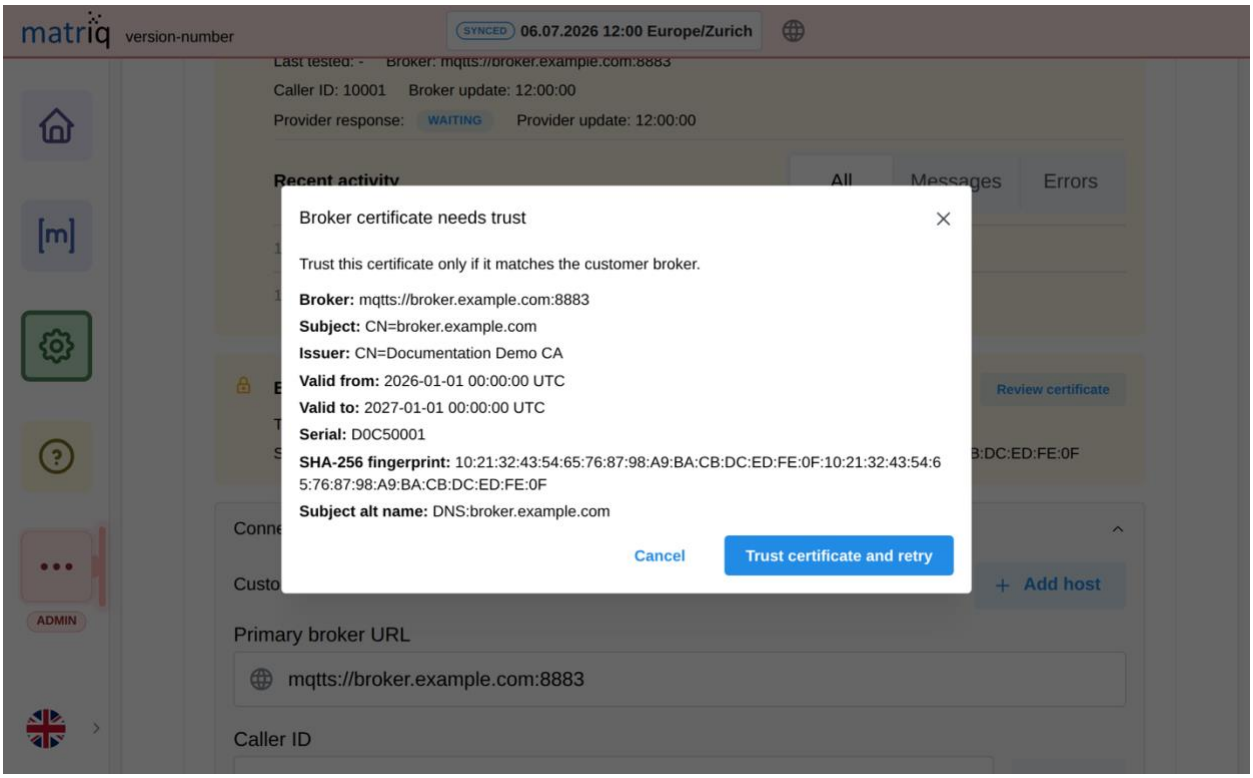

Figure 18 Certificate details

7.4 MQTT operating modes

For both modes, the important user choice is the same: the marking setup must use MQTT Service as the code source. The difference is only where the MQTT connection is configured and who initiates the connection. In server mode, the external system is configured to connect to the matriq system via the LinQ device IP address. In client mode, matriq is configured to connect to the external customer broker.

7.4.1 MQTT server mode

When using MQTT server mode, the matriq system acts as the MQTT endpoint. As mentioned previously, this is the standard setup. The external system must be configured with the following parameters:

- the matriq MQTT broker address
- credentials (username and password), provided by matriq
- the documented request and response topics

The external system then sends serial-number batches to matriq, requests the current number state when required, and can reset the stored external numbers. The matriq system stores the received numbers, uses them during the marking process, and publishes responses so that the external system can verify whether the operation was successful.

Only one batch can be processed by the matriq system at a time. Sending a new batch replaces the previously stored serial numbers.

7.4.2 MQTT client mode

In MQTT client mode, matriq connects to a customer MQTT broker instead. This mode is used when the customer provides the broker and expects matriq to actively establish the connection.

As described in chapter *7.3.2 Mode selection* MQTT client mode must be activated manually. Inside the MQTT client configuration, the following parameters must be configured:

- The customer broker URL (optionally, up to four additional fallback URLs can be configured. This enables the matriq platform to try another broker if the primary broker is not reachable)
- MQTT credentials (username and password)
- The caller ID, which by default uses the unique hardware ID of the controller. This allows the matriq controller to be identified on the customer broker. Alternatively, the caller ID can be overwritten with a custom identifier.

If the broker uses a secure MQTT connection, such as mqtts:// or wss://, matriq validates the broker certificate. When the certificate is not yet trusted, the tablet displays the certificate details, and the user must review and trust the certificate before the connection can be established.

After saving the settings, matriq tests and establishes the connection to the customer broker. The tablet displays the current broker connection status, the active broker URL, the last update time, and the latest response from the serial-number provider.

During production in client mode, matriq automatically requests new serial numbers from the external system when the local number pool becomes low (less than 10 numbers available). The external system responds on the configured provider topics with the requested serial numbers. matriq stores the received numbers and uses them during the marking process.

After each marking, the matriq system reports the used serial number back to the customer broker, including the channel information. If duplicate numbers are received, matriq warns the operator. If already-used numbers are returned, matriq rejects the provider response and waits for operator recovery.

8 Settings

Central system settings such as time, network or communication interfaces can be configured directly via the matriq platform's UI using the "Settings" menu item.

All functionalities described in this chapter can only be modified by users with the Admin role. Users with the Operator role have read-only access and cannot make any changes. To log in as an Admin, click the three-dot button in the menu on the left side of the screen and select Admin Login from the popover menu.

8.1 NTP server and time zones

Precise system time is essential for time-based markings (Unix timestamp marking patterns), logging of events and synchronization with other systems - especially in automated production environments.

The matriq platform supports the synchronization of the system time via so-called Network Time Protocol (NTP) servers. Synchronization takes place automatically, provided there is Internet access and the predefined public time servers are accessible.

In isolated networks without internet access, for example in internal production intranets, the system time can be provided via user-defined NTP sources. Up to four time servers can be configured manually for this purpose, either as IP addresses or as resolvable host names (DNS).

The system uses a hierarchical fallback strategy:

1. An attempt is first made to synchronize with the primary (top-level) time server.
2. If this attempt fails, the next configured server in the sequence is contacted automatically.
3. This process is repeated until an accessible server successfully transmits a valid time message.

8.1.1 Procedure for setting up the time server

The configuration is carried out via the Settings → Time Server area within the user interface.

1. Navigate to the Settings page.
2. Select the Time Server area.
3. Enter up to four time servers - either as IP addresses (e.g. 192.168.0.100) or as host names (e.g. ntp.localdomain).
4. Save the settings.

Note: It should be ensured that the servers used are permanently accessible and correctly registered in the internal DNS if host names are used.

The status of the time synchronization is displayed visually on the right-hand side of the user interface. Among other things, the following information is available here:

- Server reached (host name/IP)
- Synchronization status (successful/failed)
- Time deviation in milliseconds compared to the server

This information helps with the diagnosis of time deviations and network problems.

8.1.2 Error Indication

A correctly synchronised system time is required for production. If no NTP server is configured or time synchronisation is not confirmed, a warning triangle appears in the top-right corner. Tap it to view details.

After saving the NTP settings, allow two minutes for synchronisation. A warning is displayed during this period. If synchronisation is not confirmed within this time, an error message appears.

9 Updates to the matriq platform

The matriq platform updates contain updates for all operational application components, including the user interface (UI) on the tablet as well as the browser-based applications and the operating system. These updates provide new functions, stability improvements and consistently up-to-date application logic.

Software updates can only be performed by users with the Admin role. Users with the Operator role have read-only access and cannot make any changes. To log in as an Admin, click the three-dot button in the menu on the left side of the screen and select Admin Login from the popover menu.

9.1 Procedure for matriq updates

1. Connect USB stick

2. Navigate to the matriq UI settings section.
3. Open the "LinQ Update" area (this area is already open by default).
4. Enter the provided update file (*.tar.gz or *.rauc) in the corresponding file field or navigate to the file manually.
5. Start the update.

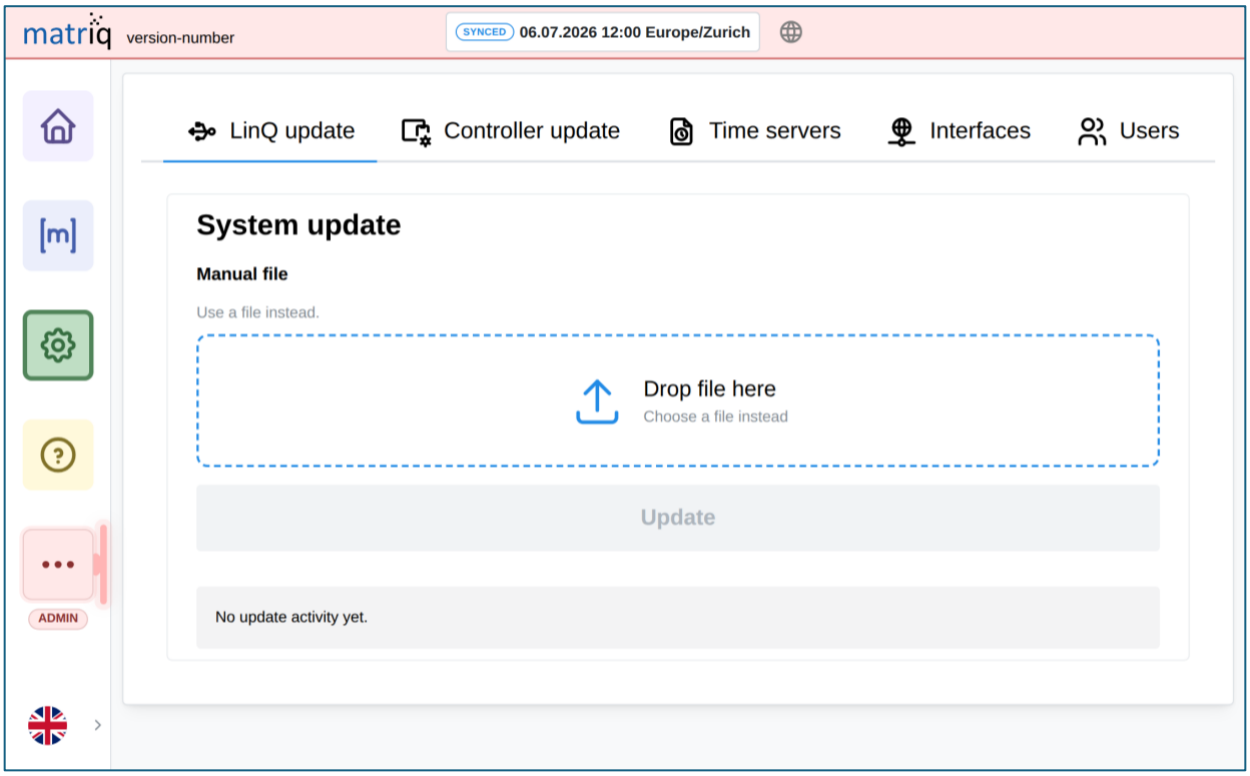

Figure 19 Settings page with active "LinQ Update" tab

After starting, the platform automatically recognizes from the file whether it is an operating system update or a software update. The update logic is then activated, and the update process begins.

During the process:

- The progress is displayed in real time in the log area at the bottom of the user interface.
- Error messages and status information are clearly documented.
- The overall status section in the upper right corner of the UI will display the status of the update as long as the update process is in progress or a restart is pending.

Once the update has been successfully completed, a message will appear indicating that a restart is required. This restart can only be performed once marking has been disabled for

all channels. The updated system will only be fully activated after this restart. If an error occurs when updating the system, the system automatically reverts to the last functioning version and starts with this version. The currently installed software version can be always seen in the user interface.

Important notes

- The update process must not be interrupted. The USB stick may only be removed after the update has been completed.
- Ideally, matriq updates should be carried out outside of productive operating hours.
- After the restart, the DM-LinQ system is fully operational and up to date.

10 Information and status report

The information page can be accessed by clicking the question mark icon in the left-hand navigation bar of the user interface. It provides a central overview of the current system status, connected hardware components and relevant support contact data. This view is used both for transparency during operation and for quick orientation in the event of support.

10.1 Contact information

The support contact details of matriq AG are displayed at the top of the page. These include the following:

- Address of matriq AG
- E-mail address for support requests
- Technical support telephone hotline (also accessible via QR code)
- Link to online support (also accessible via QR code)

This information is deliberately placed in a prominent position to enable quick access to assistance in the event of a fault.

10.2 Status report

The information page provides an integrated diagnostic function to assist with support cases. This makes it possible to automatically collect system-relevant log files and provide them as a compact report. A status report can be generated using the "Create" button. This process includes:

- the compilation of all relevant system and error logs,

- the protocols of active modules and the controller,
- as well as other status data that is helpful for error analysis.

The collected information is automatically summarized in a JSON file. Once the file has been successfully created, it can be downloaded directly and copied to a USB stick, for example, or forwarded to the matriq AG support team by email.

The status report serves as a central error analysis basis and enables technical support to respond more quickly and in a more targeted manner. The structured collection of all log data eliminates the need for queries or manual exports of individual files.

10.3 Overview of the connected hardware

Below this is a dynamic system overview that lists all currently recognized hardware components in the system, including

- Device name (e.g. LinQ Controller, insert module)
- Serial number (if available)
- Current firmware version
- Current software version

This display is constantly updated and always reflects the current hardware inventory. If a new module or an additional controller is connected, it automatically appears in the overview.

11 Diagnostics

Diagnostics can be accessed by clicking the More Options button (three dots) in the menu on the left side of the screen. It can be used to review the current condition of the inserts and monitor live sensor values.

11.1 Insert Health

The Insert Health tab provides an overview of defective pixels and switches and their impact on Data Matrix code quality and readability. *Figure 20* shows an overview of the page.

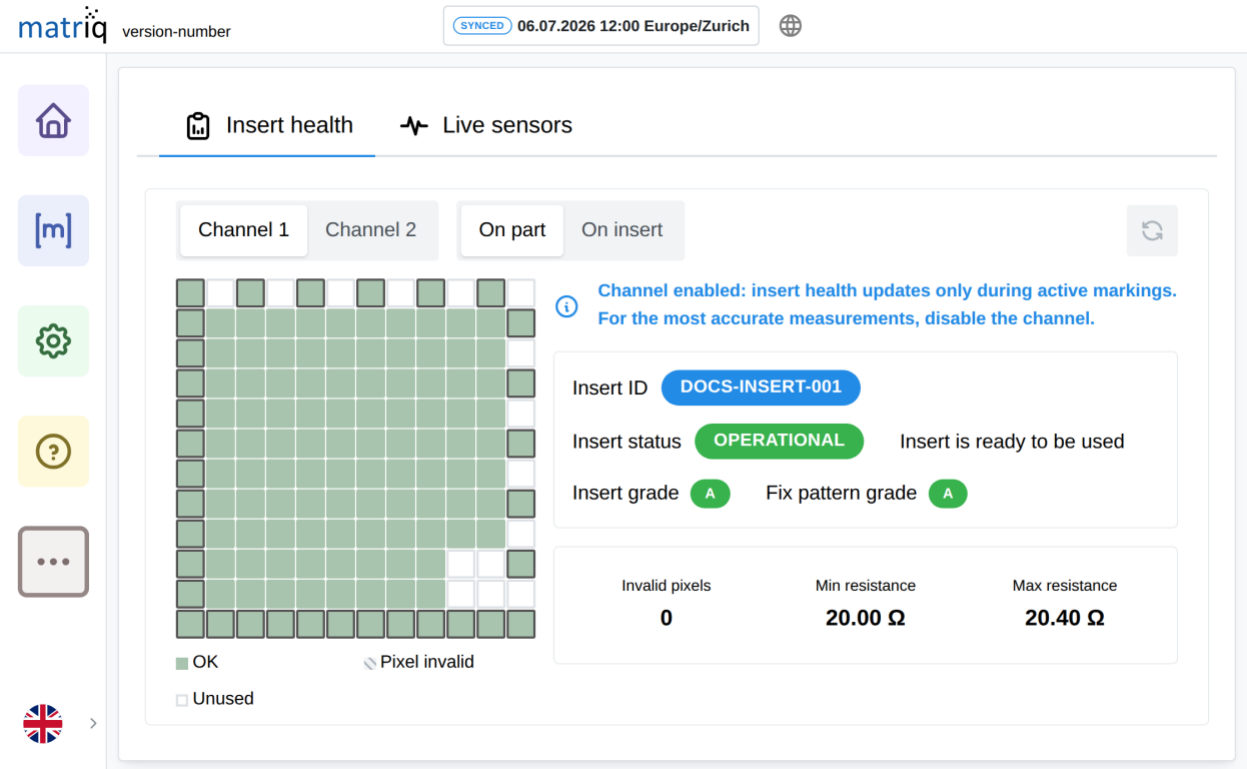

Figure 20 Insert health overview

The view allows switching between different channels and between the following display modes:

- On Part: Displays the pixel positions as they appear on the generated Data Matrix code on the physical part.
- On Insert: Displays the physical layout of the insert, including the position of individual pixels and switches. For easier orientation, the location of the insert's flat cable is also shown.

The Insert Health can be measured while a channel is either enabled or disabled. However, the most accurate measurements are obtained when the channel is disabled. Additionally, switch values cannot be measured while the channel is enabled. When the channel is enabled, active markings must be in progress for the measurement to be performed.

The Insert Health view indicates which pixels or switches are invalid and visualizes their location within the Data Matrix code. This information can be used to assess the potential impact of hardware defects on code generation and readability.

In general, a Data Matrix code consists of two main areas:

- Finder Pattern: the fixed structural elements of the code, including the solid L-shaped border (bottom left) and the alternating timing pattern (top right). These elements are used by readers to locate and orient the code.
- Data Region: the area containing the encoded information in the form of data codewords.

The Insert health is evaluated based on the number of defective codewords. A codeword consists of a group of eight pixels within the Data Matrix code. If one or more pixels belonging to a codeword are defective, the entire codeword is considered affected.

The 12×12 Data Matrix code used by the system contains 12 codewords: 5 data codewords and 7 error-correction codewords. The following error scale is used for matriq Inserts:

- Grade A – excellent: up to one defective codeword. Code grade A or better is guaranteed.
- Grade D – sufficient: two defectice codewords. Code grade D or better is guaranteed.
- Grade F – insufficient: three defectice codewords. Code grade F or better is guaranteed.
- NOT READABLE: more than three defectice codewords

The calculated code grade only reflects the principle ability of the insert to write a certain code. Factors like the actual contrast or modulation that is achieved on the part or if there are any surface defects cannot be measured by the system and are therefore not taken into account.

In accordance with ISO/IEC 15415:2011 the use of inserts with a grade below D is not recommended.

11.2 Live Sensors

The Live Sensors tab allows live analysis of three different measurement parameters: the insert ASIC temperature, the pixel heating current (I Heat) and the trigger sensor measurement (R Trigger). One channel in combination with one parameter can be displayed at a time. The displayed time range of the graph can be adjusted to 10 s**,** 30 s, or 100 s**.**

For the measurement to take place, the selected channel has to be enabled. Additionally, “I Heat” can only be measured while the insert is actively performing markings.

The user interface is shown below.

Figure 21 Live sensors overview

Important information for each parameter:

- **ASIC:**
  Inside the insert there are 4 ASICs. Two of these are located closer to the insert front surface, and two are located closer to the back surface. The measurement shows the actual temperature of one ASIC at each position. ASIC 1 is located closer to the insert surface and therefore typically shows higher temperatures than ASIC 0, which is located closer to the back surface.

  With the ASIC temperature, the following aspects can be assessed:

  - the local mold temperature can be estimated
  - whether the optional cooling ring is properly set
  - the impact of the hot plastic material on the insert temperature
  - the impact of the marking process on the ASIC temperature (electronic waste heat)

  When the heating for the marking process is active, no temperature data is recorded from the ASICs.

- **I Heat:**

With this function, the required drive current of the heater pixels can be assessed. Depending on the configured target temperature, the thermal properties of the mold material, etc., a certain drive current is required to reach and maintain the designated target temperature throughout the entire heating period. The ASIC can supply the heater pixel with up to 72 mA of current for a duration of up to 20 seconds.
The graph in *Figure 22* shows two consecutive marking processes.It is clearly visible that, when the first marking process starts, the drive current immediately rises to approximately 68 mA and then decreases rapidly to maintain the target temperature. After approximately 4 seconds, there is a significant drop in the required current. The current then remains at approximately 57 mA for another 5 seconds (with small fluctuations visible). There is no current measurement while the heating is off. A straight line connects the last measurement point of the first marking process with the first measurement point of the second marking process (the immediate rise).

The following interpretation can be made:

- increasing the target temperature is still possible, and the temperature can be reached, as the maximum drive current of 72 mA is neither reached nor continuously applied in the graph
- the plastic material started shrinking and is no longer touching the insert surface after 4 seconds of heating. A thin layer of air insulates the heaters from the

plastic part. In this case, the heating duration (set to 9 seconds) exceeds the useful duration.

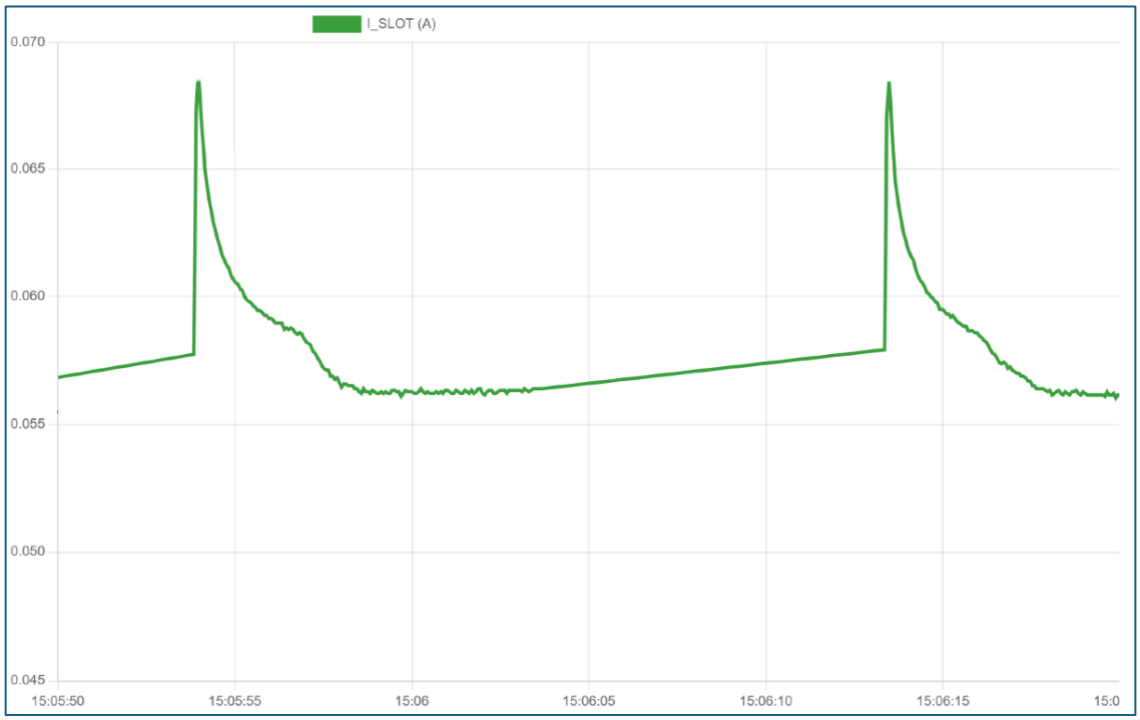

Figure 22 typical pixel current measurement graph of two consecutive marking processes

- **R Trigger:**
  The heater pixels are monitored by means of their resistance. One of the pixels (the trigger pixel) is used to detect the hot melt material and generate the trigger signal for the marking process. The graph of this resistance can be analyzed to gain a deeper understanding of what happens inside the mold.
  In the example shown in *Figure 23*, one complete molding cycle is recorded. It shows the resistance of the trigger pixel in ohms. The resistance of this trigger pixel is proportional to the temperature of the insert surface. The graph clearly shows an immediate rise at the beginning, indicating that the hot melt material has reached the insert surface. During the timeframe labeled "1", the marking process is active, and no resistance measurement is performed. The heated surface (from the marking process) then cools down until "2", where the curve bends. At this point, the hot plastic material shrinks and detaches from the surface by a few microns, causing the insert surface to cool more rapidly. After continuously cooling for another few seconds, the curve bends again. This indicates that the cavity opens and the part is ejected. As a result, the cooling rate increases further because fresh air enters the cavity.

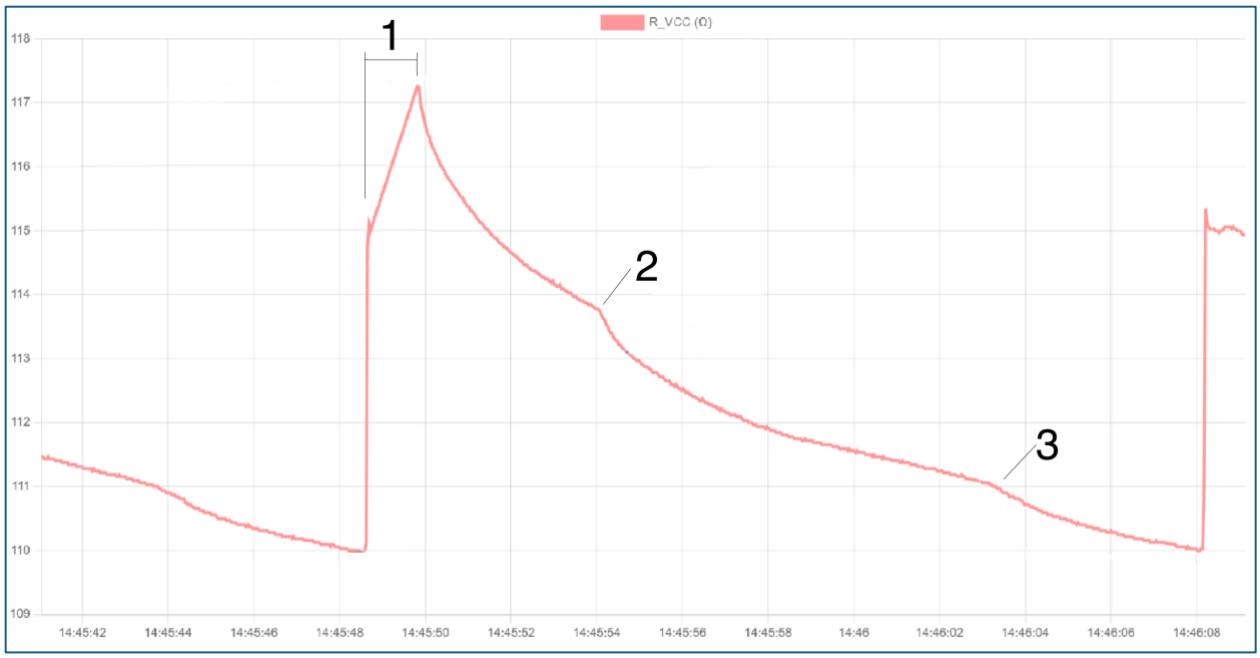

Figure 23 Graph of a typical resistance measurement of the trigger pixel