

**DataMan[®] Fixed-Mount
WebUI
Reference Manual**

2025 December 17

Legal Notices

The software described in this document is furnished under license, and may be used or copied only in accordance with the terms of such license and with the inclusion of the copyright notice shown on this page. Neither the software, this document, nor any copies thereof may be provided to, or otherwise made available to, anyone other than the licensee. Title to, and ownership of, this software remains with Cognex Corporation or its licensor. Cognex Corporation assumes no responsibility for the use or reliability of its software on equipment that is not supplied by Cognex Corporation. Cognex Corporation makes no warranties, either express or implied, regarding the described software, its merchantability, non-infringement or its fitness for any particular purpose.

The information in this document is subject to change without notice and should not be construed as a commitment by Cognex Corporation. Cognex Corporation is not responsible for any errors that may be present in either this document or the associated software.

Companies, names, and data used in examples herein are fictitious unless otherwise noted. No part of this document may be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose, nor transferred to any other media or language without the written permission of Cognex Corporation.

Copyright © 2025. Cognex Corporation. All Rights Reserved.

Portions of the hardware and software provided by Cognex may be covered by one or more U.S. and foreign patents, as well as pending U.S. and foreign patents listed on the Cognex web site at: cognex.com/patents.

The following are registered trademarks of Cognex Corporation:

2DMax, Advantage, AlignPlus, AssemblyPlus, Checker, Cognex, Cognex Vision For Industry, CVL, DataMan, DVT, EasyBuilder, Edge Intelligence, Hotbars, In-Sight, MX, OmniView, PatFind, PatFlex, PatInspect, PatMax, PatMax RedLine, PatQuick, PowerGrid, Smart View, SmartAdvisor, UltraLight, VisionPro, VisionView

The following are trademarks of Cognex Corporation:

The Cognex logo, 1DMax, 3D-Locate, 3DMax, BGAI, CheckPoint, Cognex VSoC, CVC-1000, FFD, iLearn, In-Sight (design insignia with cross-hairs), In-Sight 2000, InspectEdge, Inspection Designer, MVS, NotchMax, OCRMax, PatMax RedLine, ProofRead, SmartSync, ProfilePlus, SmartDisplay, SmartSystem, SMD4, VisiFlex, Xpand


Portions copyright © Microsoft Corporation. All rights reserved.


Portions copyright © MadCap Software, Inc. All rights reserved.


Other product and company trademarks identified herein are the trademarks of their respective owners.

Symbols

The following symbols indicate safety precautions and supplemental information:

 **WARNING:** This symbol indicates a hazard that could cause death, serious personal injury or electrical shock.

 **CAUTION:** This symbol indicates a hazard that could result in property damage.

 **Note:** This symbol indicates additional information about a subject.


 **Tip:** This symbol indicates suggestions and shortcuts that might not otherwise be apparent.

Table of Contents

Legal Notices	2
Symbols	3
Table of Contents	4
About the DataMan WebUI	6
Connecting to WebUI	7
Troubleshooting	8
Setting your IP Address to DHCP	8
Using Static IP addresses	8
Connecting on a Managed Network	8
Connecting through USB-C	8
Application Steps	9
Home	9
Code Setup	11
Positioning	11
Tuning	12
Test	17
Review	17
Application Details	20
Trigger Settings	20
Code Quality	21
Data Validation	22
Data Formatting	24
Communications	28
Communications	28
Inputs	29
Outputs	30
File Transfer	32
Runtime	37
TruCheck™ Grading	39
Configuring Grading	39
Setting up TruCheck™	40
Application Standards	41
Calibration Process	42
Transferring Verification Reports	42
Examining TruCheck™ Results	43
ISO/IEC 15416 Quality Parameters	44
ISO/IEC 15415 Quality Parameters	46
ISO/IEC TR 29158 AIM-DPM-1-2006 Quality Parameters	50
Settings	52
Overview	52
Configure Codepage	52
Licenses	55
Date & Time	56

Password Protection	57
Turning on Password Protection	57
Application Lock	57
Forgotten Password	58
Removing the Password	58
Program Management	58
Add New Program	59
Program Management with DMCC	60
Network Settings	61
Firmware Update	62
Backup Device	62
Restore Device	62
Factory Reset	63
Device Log	64
Field Diagnostics	64
Reboot	65
Secure Connection	66
Image Viewer	67

About the DataMan WebUI

The DataMan WebUI provides an intuitive interface for managing your DataMan readers. The WebUI streamlines configuration and monitoring tasks. Key features include:

- Live code review: View real-time images of the codes being read by your DataMan reader directly within the WebUI.
- Unified platform: The DataMan WebUI is consistent across all models, simplifying deployment by consolidating common controls into a single page.
- Real-time impact: Explore how different options affect the reader in real time.
- No additional downloads required: You no longer need to download and install additional software.
- Fast setup: Save time by not having to download and install software. Systems take just a few minutes to deploy.
- Programs: Programs provide a convenient method of grouping and storing multiple read setups and device-level parameters, enabling you to easily switch your reader between different applications.
- The WebUI offers the same functionality as the DataMan Setup Tool, with the following differences:
 - Settings changes are automatically saved in WebUI, while DataMan Setup Tool tool requires you to manually save settings.
 - The DataMan Setup Tool tool only provides a program-level view.

You can access the WebUI from the following browsers:

- Google Chrome
- Microsoft Edge
- Apple Safari
- Mozilla Firefox
- Opera

The following DataMan reader models are compatible with WebUI:

- DataMan 80 series
- DataMan 280 series
- DataMan 290 series
- DataMan 390 series

Connecting to WebUI

Perform the steps described to connect your DataMan reader to WebUI. These steps apply to both Ethernet and USB readers.

Note: USB readers always receive a random link local IPv4 address automatically to connect with.

Power on your reader and connect it to a PC. There are two ways of connecting your reader to the WebUI: using MAC address, serial number, host name, or IP address.

MAC address, Serial Number, or Host Name

1. Locate the MAC address and serial number on the sticker of your reader:



Alternatively, you can use the host name of the reader. By default, the host name of your reader is the model of your reader followed by the last six digits of the MAC address, using the following format: DM290-3BF53C.

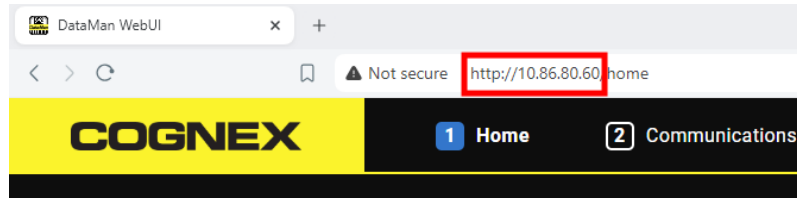
2. Type in the MAC address, serial number, or host name into the web browser followed by `.local`. For example: `00D0243BF53C.local`.

Note: Make sure that your computer and your reader are on the same subnet, otherwise you cannot use the MAC address or serial number to connect to the reader from your computer. This restriction does not apply if you are using the host name of the reader to establish the connection.

IP address

1. Find the IP address of the reader in Station Manager:
 - a. Download Station Manager. See the Cognex support site for additional information:
 - [DataMan 290 series](#)
 - [DataMan 390 series](#)
 - b. Use Station Manager to view and find the IP address of all Cognex devices on the network. To connect to a specific reader, select it and click **Connect**. Your browser automatically opens the WebUI.

2. Type your IP address into the web browser: For example: 10.86.80.60



Note: The WebUI uses a regular HTTP connection by default. If you want to set up a secure HTTPS connection, see [Secure Connection on page 66](#).

Troubleshooting

Setting your IP Address to DHCP

You must ensure that the Ethernet port of your computer is set to DHCP when connecting to a reader for the first time.

In Windows, go to **Network and Sharing Center > Your Ethernet Connection > Properties > Internet Protocol Version 4 (TCP/IPv4) > Obtain an IP Address Automatically**.

Using Static IP addresses

You can use the mDNS features of the WebUI on a static network.

Ensure that the reader and your computer are on the same subnetwork, and you can connect using the MAC address, serial number or host name to connect to the reader.

Connecting on a Managed Network

Managed networks with DNS servers behave differently than connecting the reader directly into your computer. Most internal corporate networks define different domains than `.local`, and have an internal DNS server that connects computers on the internal network through hostname.

.local Addresses do not Work

A managed network might define a different domain than `.local`. For example, an internal network may use `.computer.corporate.com` instead of `.local`. In this case you need to replace `.local` in all examples above with `.computer.corporate.com`.

To check the domain of your network:

1. Type in *About* in the search bar of your computer
2. Find Domain or Workgroup settings.
3. Verify that your local domain is listed.

Connecting through Device Name Works, but MAC and Serial do not

You can only connect using the Device Name on a managed network with a DNS server. This is because managed networks typically support a single non-IP address name for each device.

Connecting through USB-C

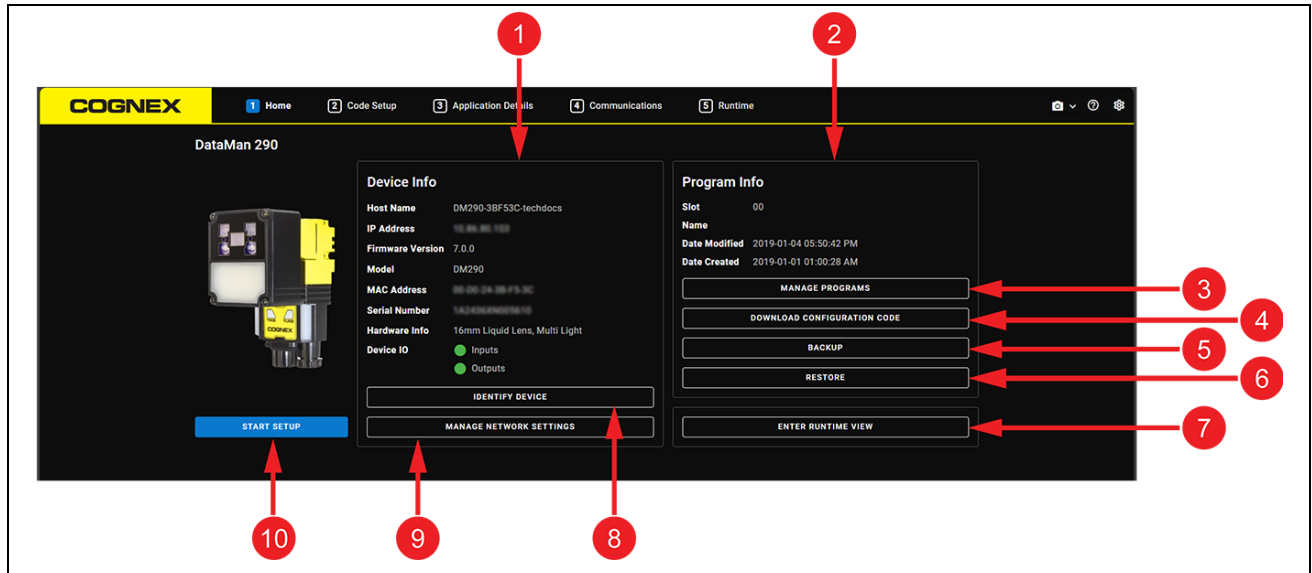
The WebUI does not support mDNS connections through USB-C.

Application Steps

WebUI provides a workflow that allows you to set up your reader through a series of guided steps.

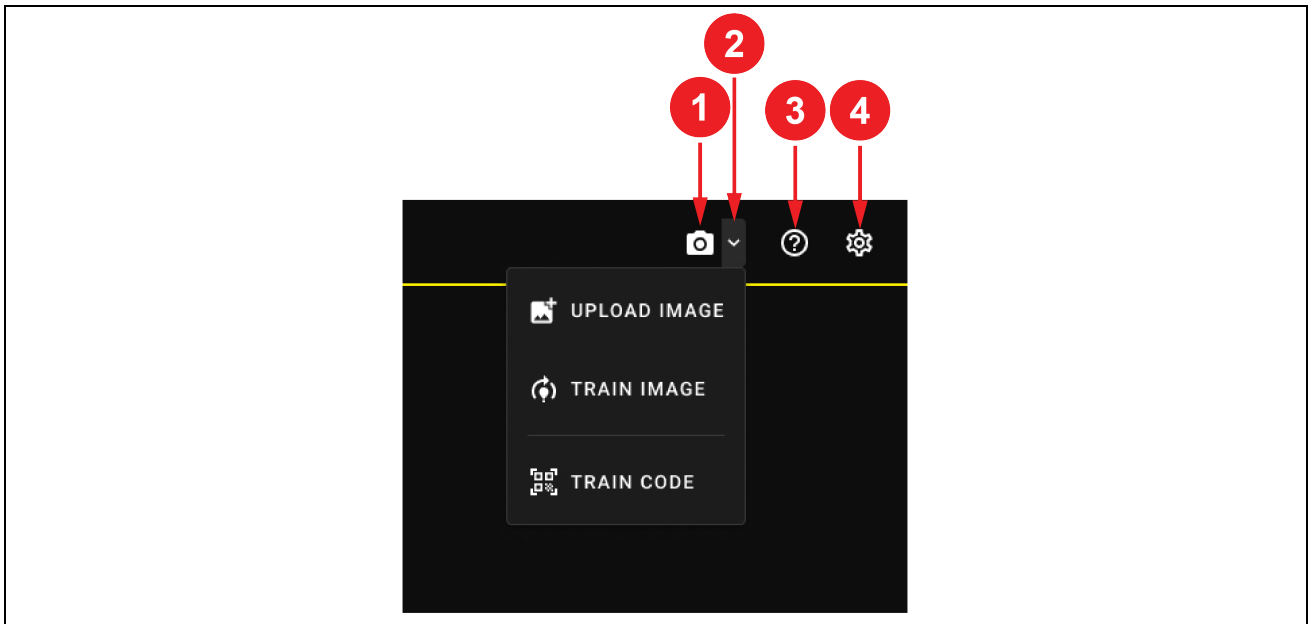
Home

The **Home** step provides an overview of your reader and serves as a central hub for reader management and configuration.



Number	Element
1	Information about the reader.
2	Information about the currently active program on the reader.
3	The Manage Programs button opens the window where you can manage the programs on the reader. For more information, see Program Management on page 58 .
4	The Download Configuration Code button downloads the current reader configuration as a series of 2D codes.
5	The Backup button opens the window to save a backup file.
6	The Restore button opens the window to upload the selected backup file.
7	The Enter Runtime View button takes you to the Runtime step, where you can see the reader reading barcodes in real time.
8	The Identify Device button activates the ring light indicator of the reader.
9	The Manage Network Settings button opens the connectivity settings panel.
10	The Start Setup button takes you to the Code Setup step, where you can perform the initial positioning and tuning for your application.

The top right corner has the following options:



Number	Element
1	The Trigger button manually triggers the reader.
2	<p>The dropdown provides additional image options:</p> <ul style="list-style-type: none"> • Click Upload Image to open the file browser, and select the image you want to upload for processing. • Click Train Image to open the file browser, and select the image you want to train the reader on without tuning. • Click Train Code to acquire an image and train the image without tuning.
3	<p>The available help information, including page help and documentation:</p> <ul style="list-style-type: none"> • Click Page Help to get additional information for the step that you are on. • Click Getting Started Guide to open an external website with extended help materials. • Click Launch Help Walkthrough to get an interactive overview of the given step.
4	The Settings panel. For more information, see Settings on page 52 .

Code Setup

The Code Setup step allows you to set up, test, and review the positioning and tuning of your reader to ensure that your reader settings are optimized for the code that you are trying to read. Follow the substeps in the menu on the left side to configure your reader.

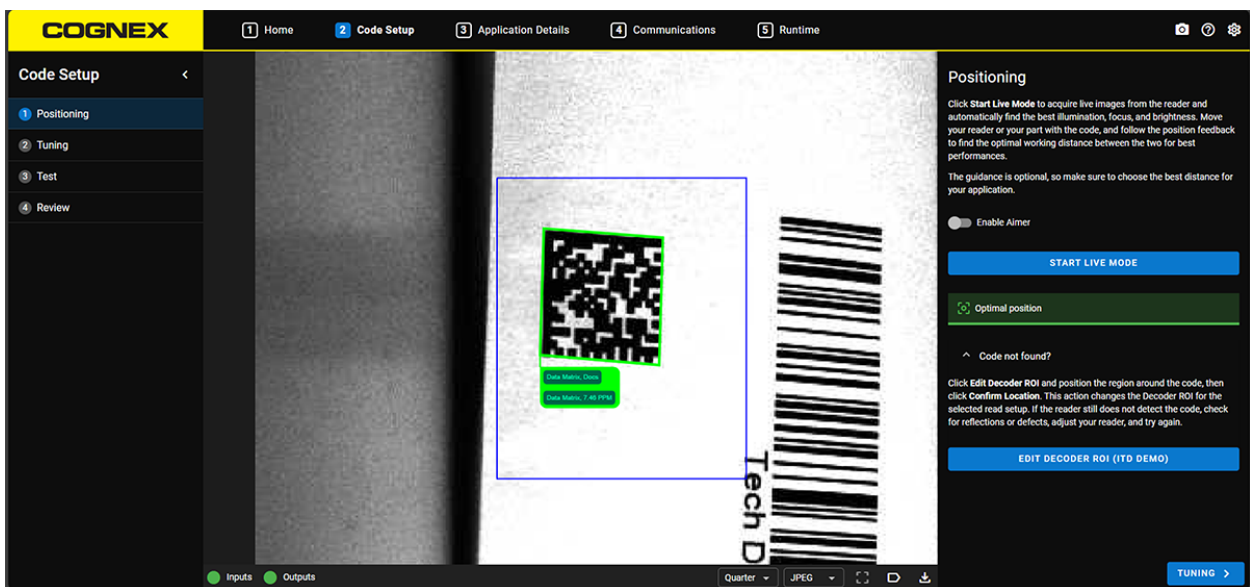
Positioning

The **Positioning** substep allows you to view live images from your reader, and read codes as you install and mount your reader.

The **Positioning** substep provides real time position guidance for the distance between the reader and the part, such as instructing you to move your reader closer to or further away from the part with the code.

Note: Position guidance is an optional guideline for finding the best possible decoding speed and consistency on challenging part markings. You can proceed with code setup regardless of your current position guidance.

You can trigger the reader manually using the **Trigger** button, or you can click **Start Live Mode** to automatically adjust the lighting, focus, and exposure for optimal results while you adjust the positioning.



Click **Start Live Mode** to acquire live images from the reader and automatically find the best illumination, focus, and brightness. Move your reader or your part with the code, and follow the position feedback to find the optimal working distance between the two for best performances.

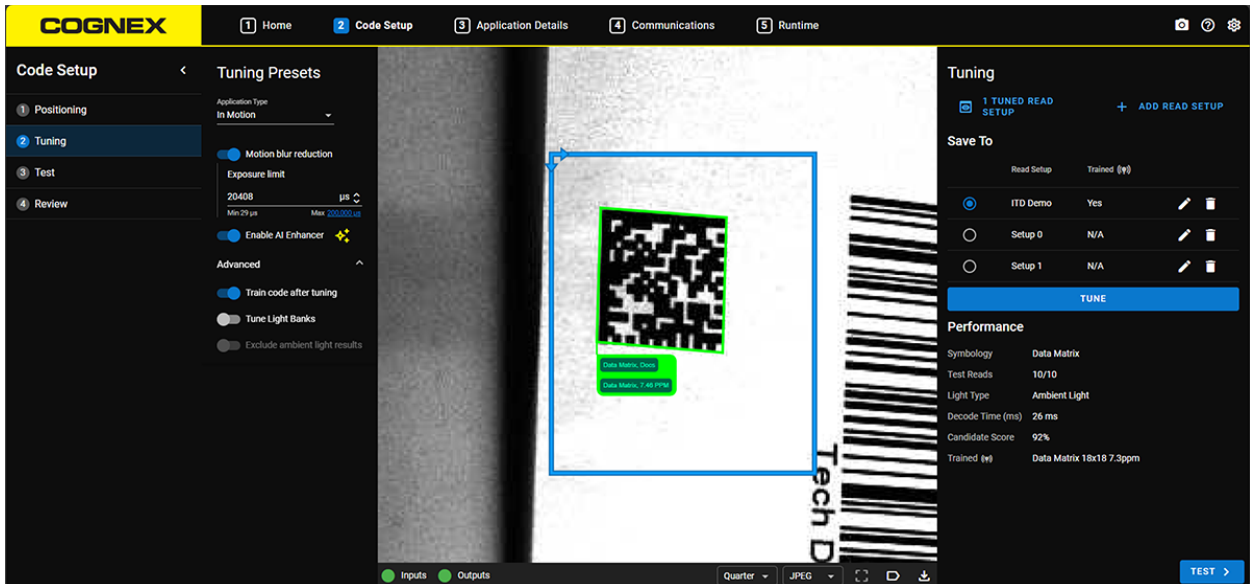
The guidance is optional, so make sure to choose the best distance for your application.

If the reader cannot find the code, open the **Code not found?** dropdown to mark the code manually.

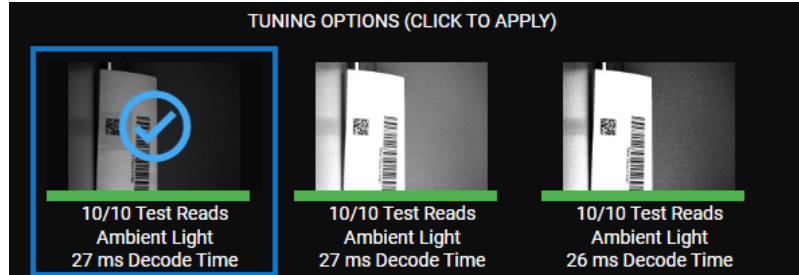
Click **Edit Decoder ROI** and position the region around the code, then click **Confirm Location**. This action changes the Decoder ROI for the selected read setup. If the reader still does not detect the code, check for reflections or defects, adjust your reader, and try again.

Tuning

The **Tuning** substep allows you to automatically find the best code reading settings for the reader.



Drag and drop the blue rectangle to define the decoder Region of Interest (ROI). The reader only decodes symbols that appear in the ROI. Click the **Tune** button to start the process of tuning the reader to the code. Once tuning is complete, WebUI displays the tuning options with the highest candidate scores, and the best candidate of the tuning options is automatically applied. You can switch to any of the other generated tuning options if you do not want to use the highest-ranked candidate.



The tuning procedure optimizes the following image acquisition parameters to obtain the best decode time and decoder performance:

- reader settings
- Light banks
- Lens Focus
- Image-filters
- Symbology attributes

You can also configure your tuning settings in the **Tuning Presets** column. Open the **Advanced** dropdown to access advanced parameters.

Select from the preset options to refine the tuning procedure:

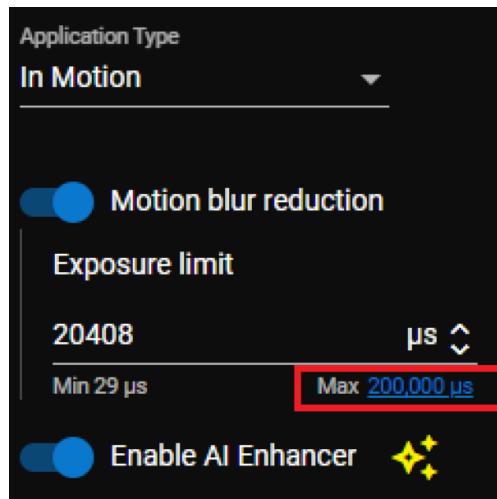
- **Application Type:** choose between **Stationary** or **In Motion**. If your application type is In Motion, you can reduce motion blur by properly adjusting the maximum exposure, either manually or using the embedded assistant.
- **Enable AI Enhancer:** use the AI Enhancer to improve decoding of damaged and degraded barcodes. The AI Enhancer uses artificial intelligence to enhance image regions which can contain a code to help improve overall decoding performance. The **Visibility** slider allows to dynamically overlay AI-adjusted regions based on the top three tuning candidates. AI Enhancer can help to increase code readability with only minimal impact to decoding speed.

Note: The DataMan 80 and DataMan 280 do not support the AI Enhancer.

- **Advanced options:**
 - **Train code after tuning:** enable this option if the codes you want to decode have a consistent symbology type and characteristics. Training the reader with the tuned symbology and its characteristics can decrease the time required to decode subsequent symbols, and help increase readability and read rate.
 - **Tune Light Banks:** enable to tune the reader illumination type.
 - **Exclude ambient light results:** the tuning procedure ignores images acquired without illumination.
 - **Field of View:** shows the current dimensions of the FoV and allows you to edit the FoV.

For **In Motion** type applications with **Motion blur reduction**, you can use the **Max Exposure Calculator** to calculate the recommended maximum exposure time:

1. Click on the **Suggested Max** number to open the **Max Exposure Calculator** dialog.



2. Enter the **Code Element Size** and **Max Line Speed** values.
The WebUI automatically calculates the recommended maximum exposure time.
3. Click **Apply** to use the calculated value as your **Exposure limit** for tuning.

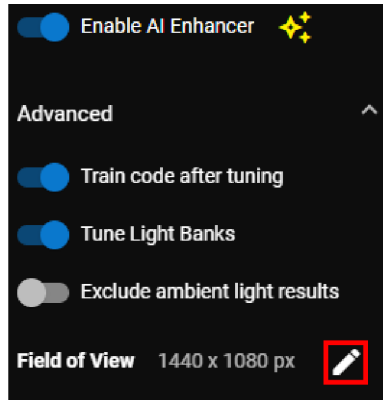
You can apply the tuning results to an existing read setup by selecting the target read setup in the **Save To** column. You can also click **Add Read Setup** to create a new read setup for the current tuning settings. You can use multiple read setups when there is wide variation between parts.

When you create a new read setup, you can copy the settings from an existing read setup, or you can perform tuning to save a new set of settings to the read setup.

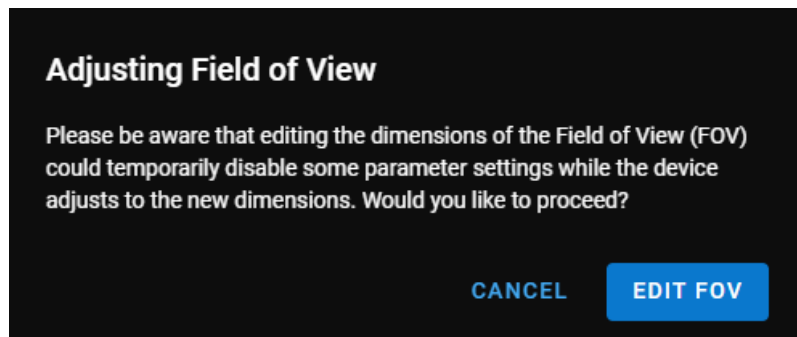
Adjusting the FoV

To resize the FoV, do the following:

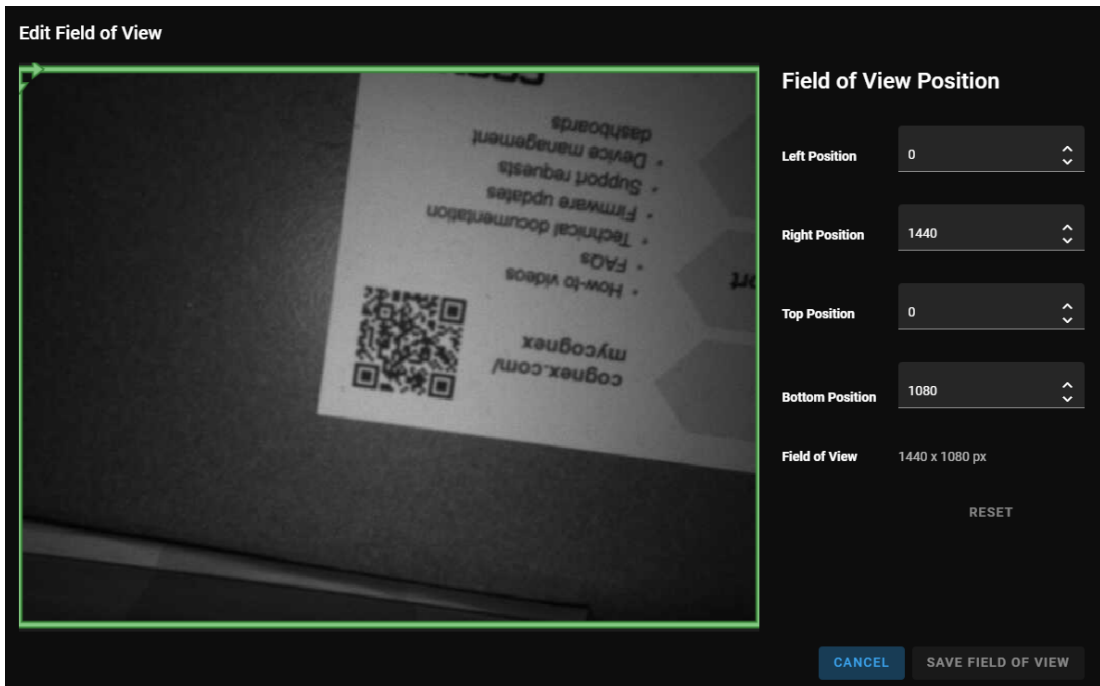
1. Click **Edit** next to the **Field of View**.



2. A dialog pops up informing you that the reader needs time to adjust to the new FoV settings, which means some of the functions are disabled during the adjustment process. Click **Edit FoV** to continue.



3. The **Edit Field of View** dialog pops up.



Edit the size of the FoV in one of the following ways:

- Drag the green rectangle to resize the FoV.
- Write values into the FoV positions.

The minimum value is 80 pixels for width and 64 pixels for height. For example, if you write 200 pixels into the **Left Position**, then you must write at least 280 pixels into the **Right Position**.

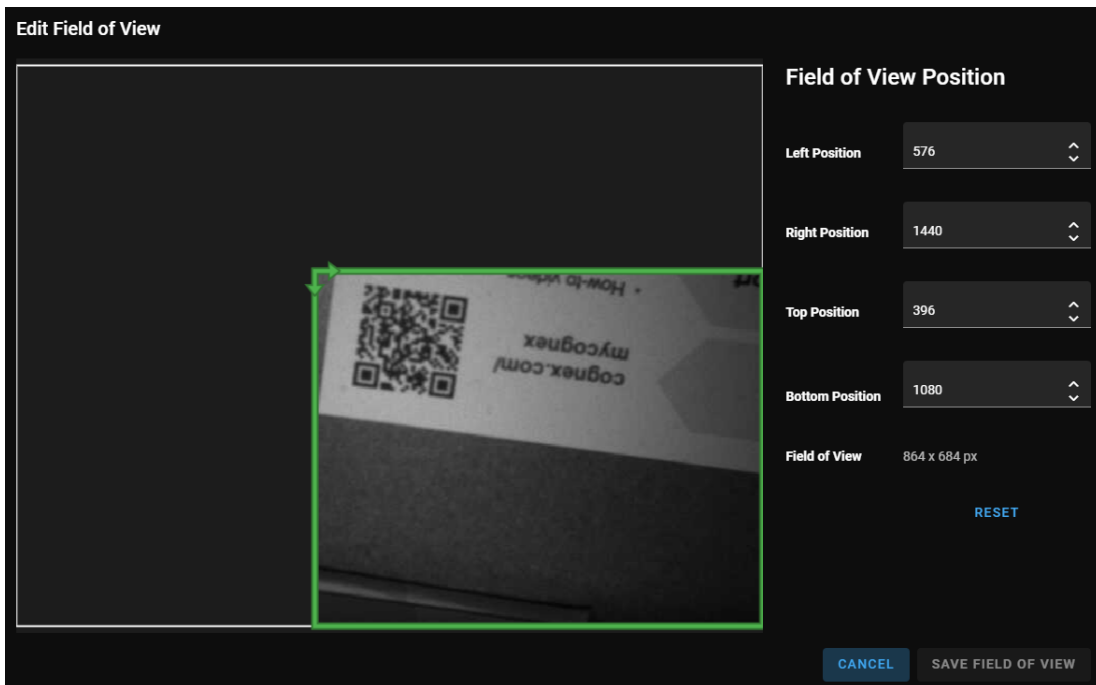
- Increase or decrease the position value by clicking on the arrows.

One click changes the value by eight pixels for width and four pixels for height. If you type in any number, the reader adjusts the value to the next lowest valid value. For example, if you type 100 pixels for the left position, the reader automatically adjusts the value to 96 pixels because 100 is not divisible by eight.

- Click **Reset** to reset the position values to the default values of the reader.

4. Click **Save Field of View** to finish. The reader acquires a new image with the new FoV.

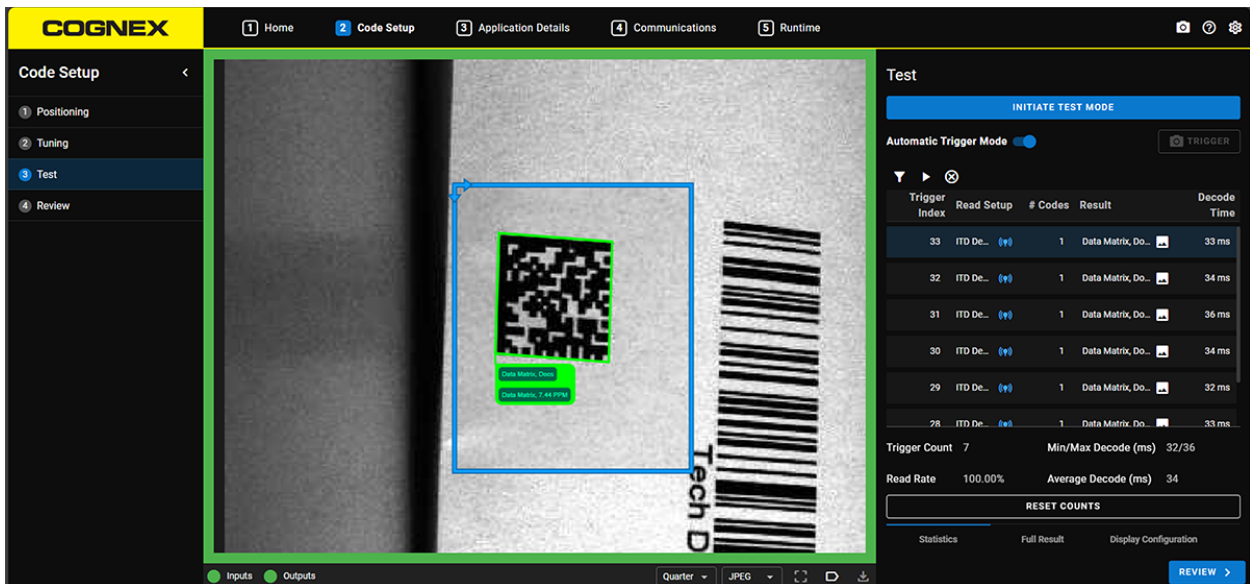
If you click **Edit Field of View** again, you can see the new FoV as the subregion of the full FoV, showing where the new FoV is compared to the full FoV.



Note: When clicking **Save Field of View**, tuning and live mode become temporarily unavailable. The decode time can change during the adjustment process if you trigger the reader. Once the reader is ready with the FoV adjustments, functions and speed resume back to standard.

Test

The **Test** substep allows you to test your tuning results without configuring a live trigger.



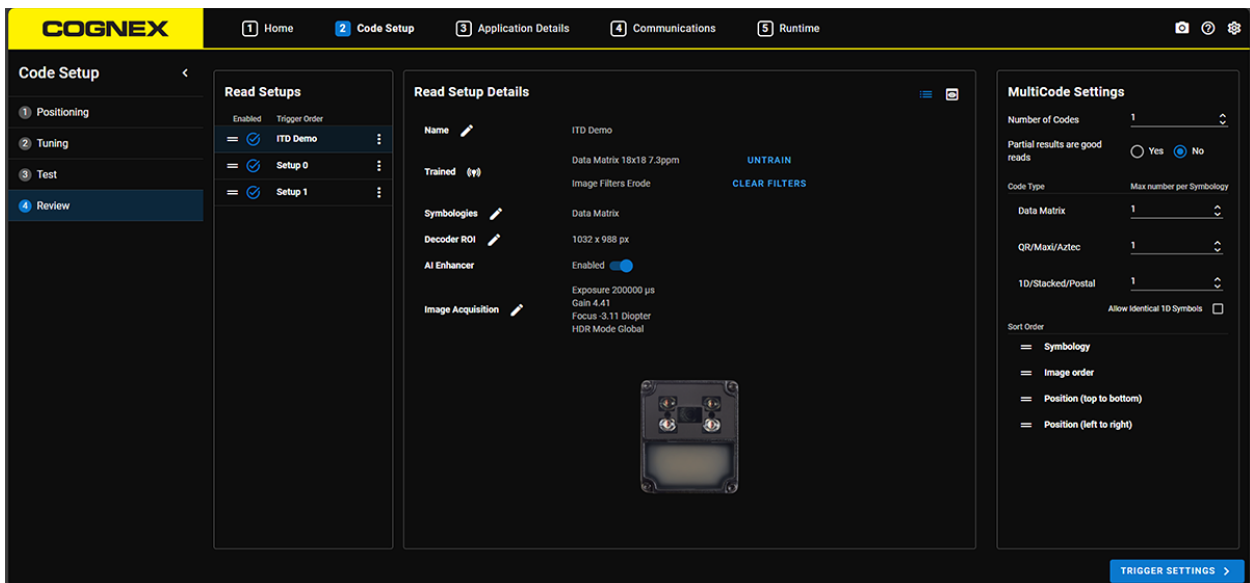
You can trigger the reader with the manual trigger button or switch to automatic mode. In **Automatic Trigger Mode**, the reader triggers as fast as possible.

Test mode automatically uses all enabled read setups. You can select the read setups you want to use for runtime in the **Review** substep.

Use the result history and statistics to determine if your current reader settings are suitable for your application.

Review

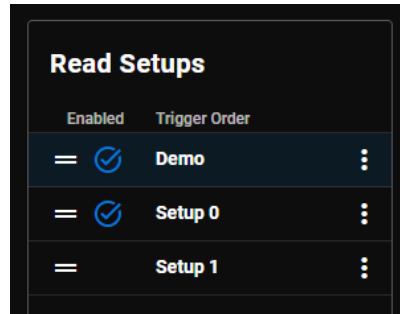
The **Review** substep allows you to view and modify your tuned settings, select which read setups to enable, delete read setups, or compare settings between them.



The **Review** substep provides a high-level overview of the tuning parameters, such as symbology, gain, and exposure for each read setup. You can also edit current read setups and configure **MultiCode Settings**.

A read setup is a set of image acquisition and decoding settings. The reader cycles through all enabled read setups until the trigger ends or the read result is complete. Use the **Read Setups** pane to enable or disable read setups, and change their **Trigger Order** by dragging the handles.

Manage all your configured read setups on the **Review** substep. The checkmark near a read setup means that this setup is currently enabled.



Click the three dots to the right of a read setup to open the read setup dropdown, where you can edit, compare, disable, or delete that setup.

You can check and configure each read setup individually. The settings shown in the **Read Setup Details** pane are based on the results of the **Tuning** substep. You can also view the image from which the tuned settings were derived, and compare the settings of two different read setups.

The following information is available for each read setup. Click on the pen icon to change a given setting:

- **Name:** The name of the currently selected read setup.
- **Trained:**
 - The type and size of the code that was trained during the Tuning step. Click **Untrain** to reset the trained code.
 - The filters that the Tuning procedure applies to the image to achieve the tuned image. Click **Clear Filters** to remove the filter.

Note: Use Setup Tool or DMCC commands to modify the training and filter configuration.

- **Symbologies:** The list of enabled symbologies that the reader recognizes.
- **Decoder ROI:** The Region of Interest (ROI) the decoder operates on.
- **AI Enhancer:** Enable or disable the use of AI enhancer.
- **Image Acquisition:**
 - **Exposure** and **Gain** affect the brightness of the image. Click **Optimize** to automatically determine good values.
 - **Focus** affects the distance where the image is the most clear. Click **Optimize** to automatically determine the best value.
 - **HDR Mode** affects whether HDR is enabled. This setting affects all read setups.
 - **High Frequency Light (HFL)** prevents the reader from rapidly flashing. In combination with the **Enabled** option, the **Always On** setting turns on the HFL at the initial trigger and maintains it after a trigger ends. You can set the **Timeout** measured in seconds after which the HFL turns off if no trigger occurs for a long period of time. This setting affects all read setups.
 - Configure which light bank of the reader is on by clicking on the reader image and selecting a light bank.

Use the **MultiCode Settings** pane to configure how to read multiple codes in one image. Determine the overall number of codes to read and the number of codes for each symbology. If you disable the **Partial results are good reads**, the reader does not return any codes, including single code images when the image contains fewer codes than specified with **Number of Codes**. If you enable **Partial results are good reads**, the reader returns a successful read result with fewer codes. The **Sort Order** list allows you to determine how the reader reports multiple results:

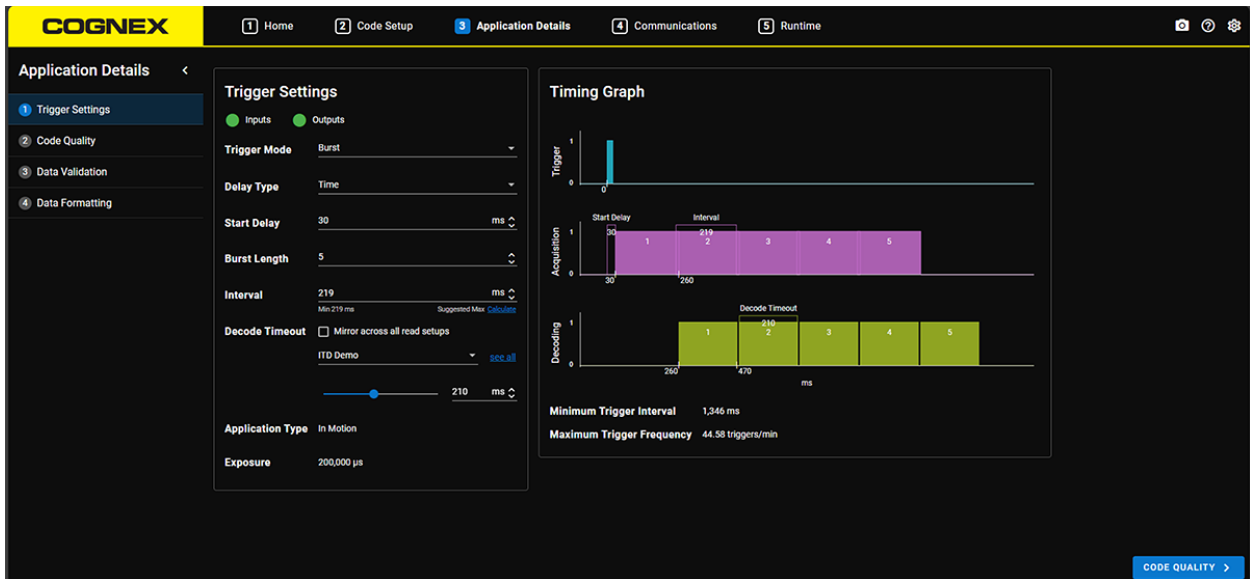
- Use the handles to change the priority order.
- The **Position (top to bottom)** and **Position (left to right)** options refer to the center of the symbols.
- Use **Image order** for Burst or Continuous trigger modes, where symbols that are decoded in earlier images are reported before symbols decoded in later images.

Application Details

The **Application Details** step allows you to configure the details of your application. Follow the substeps in the menu on the left side to configure your application.

Trigger Settings

The **Trigger Settings** substep allows you to configure your runtime processing and decoder timeout settings. This substep features a timing graph that reflects the trigger settings to give feedback on the current application setup.



Triggers that initiate a decode attempt either come from a source external to the reader or are generated by a timing mechanism internal to the reader itself.

When the reader is configured to respond to an external trigger, any of the following events signal the reader to perform a decode attempt:

- Pushing the physical button on the reader
- Sending the DataMan Control Command (DMCC) **trigger** when connected to the reader over a serial or Ethernet connection
- Receiving a digital input signal over the trigger lines of the I/O cable
- Clicking the **Manual Trigger** in WebUI

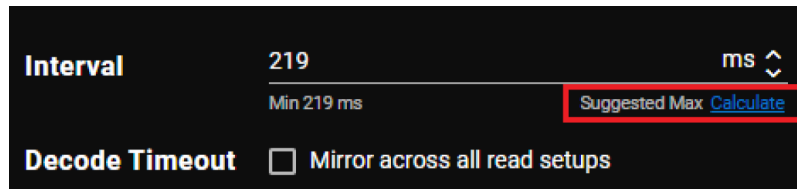
The following trigger types are available:

- **Self** trigger mode continuously acquires images at a configurable interval, and attempts to decode symbols found.
- **Single** trigger mode acquires a single image and attempts to decode any symbols found.
- **Burst** trigger mode acquires a set number of images and decodes until the expected codes are found. You can configure the number of images within each burst and the interval between each image acquisition. This trigger mode supports a read timeout. This timeout is the maximum amount of time spent on each image in the burst, and not on the burst as a whole.

- **Continuous** trigger mode acquires images as long as the trigger signal remains active, where the reader acquires images at a specific interval and attempts to scan any symbols each successive image contains. If the desired number of codes is decoded, the reader ends the trigger and sends out results. Until the desired number of codes is decoded, image acquisition continues as long as the trigger signal remains active. If the trigger signal ends and the desired number of codes is not decoded, the reader returns a NoRead result.

For **Burst** and **Continuous** trigger modes, you can use the **Max Interval Calculator** to calculate the recommended maximum interval:

1. Click **Calculate** to open the **Max Interval Calculator** dialog.



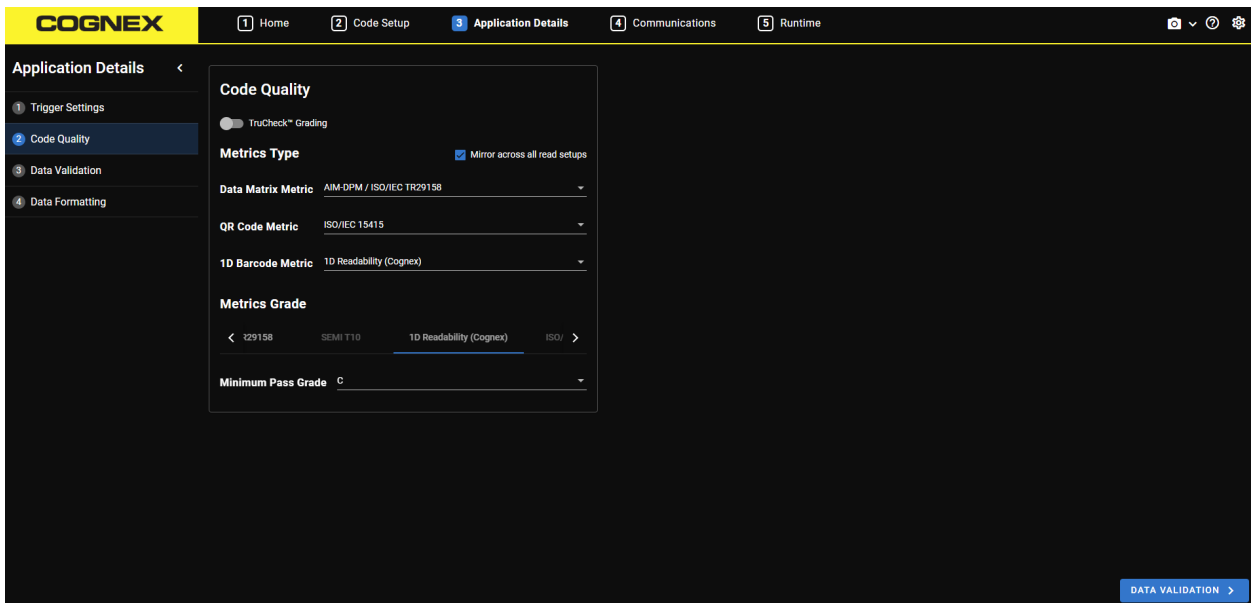
2. Choose your **Application Type** to display different options for calculating the necessary data:
 - Select **Field of View Size** to calculate the longest possible interval time by measuring the physical **Field of View Size** and giving the **Maximum Line Speed** and the size of the **Longest Code**.
 - Select **Lens Distance to Code** if you want to give the distance of the code from the lens. Give the **Lens Distance** and select the **Motion Direction** in which the code is traveling. Here you also need to give the **Max Line Speed** and the size of the **Longest Code**.
 - Select **Code Element Size** to calculate with the size of the most narrow code element. Press the **Trigger** button to read your code and obtain the **Code PPM** value. You also need to give the **Motion Direction** in which your code is traveling in, the **Max Line Speed** and the size of the **Longest Code**.

The WebUI automatically calculates the recommended maximum interval time when you fill in all the required fields.

3. Click **Confirm** to use the calculated value as your maximum **Interval** time.

Code Quality

The **Code Quality** substep allows you to validate that the symbols you use meet the quality guidelines as outlined in one of the code quality standards supported by Cognex. Codes get an overall grade after grading all or selected metrics for a given code quality standard.



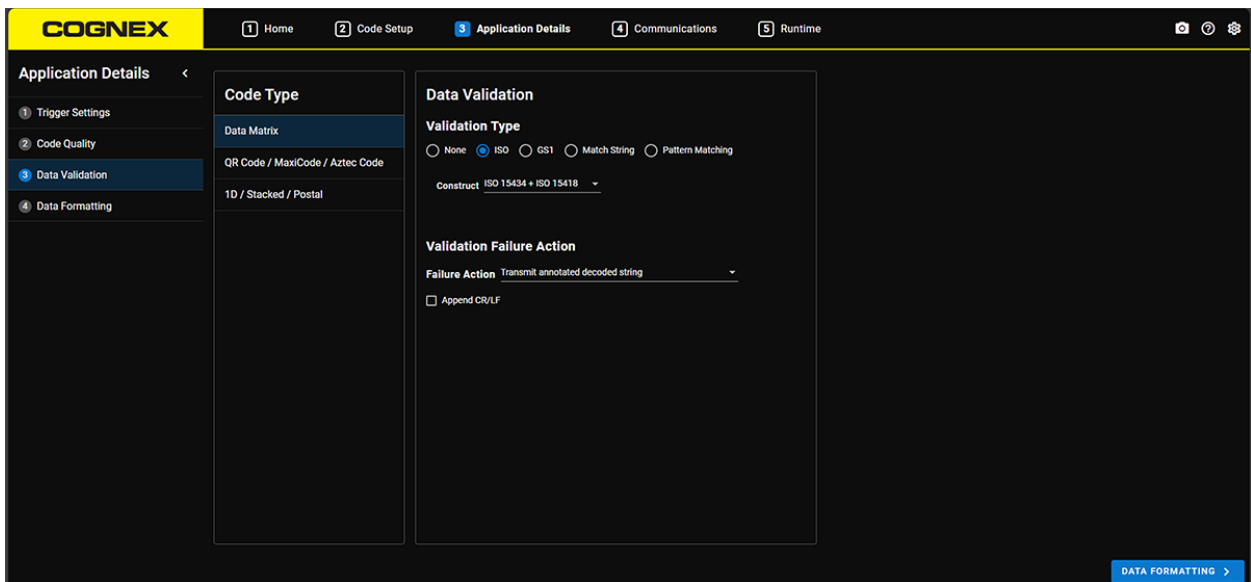
For more detailed information on specific code quality standards and their metrics, see the *DataMan Communications and Programming Guide* or the corresponding ISO standard documentation, which can be found at www.iso.org.

Configure a grade threshold for acceptable code quality with the **Minimum Pass Grade** toggle.

To enable Standards-Based Grading, enable the **TruCheck™ Grading** toggle. For more information, see [Configuring Grading on page 39](#).

Data Validation


The **Data Validation** substep allows you to confirm that the data encoded by a symbol is in the correct format for a particular company, industry, or international standard. The settings shown on the page depend on the **Validation Type** you choose.



Select the **Code Type** and assign a **Validation Type** to it.

- Use the **None** option if there is no need for any validation type.
- Use the **ISO** option to specify an ISO standard for the strings.

- Use the **GS1** to specify an application identifier for the strings.
 - In addition to the list of specific AIs that you can select, the following special tokens are available:
 - <AI(.)> to signify AI
 - <AI(*)> for an arbitrary number of AIs
 - DataMan software supports GS1 Digital Link but only for syntax checking. With GS1 Digital Link enabled, the reader ignores any specified pattern, as if the pattern was set to <AI(*)>.
- Use the **Match String** option to specify an exact string to match against the string encoded by the symbol. Only symbols containing a string that matches the specified string generate a **Pass** result.
- Use the **Pattern Matching** option to input a regular expression to confirm if the data string includes a specific string of characters.

 **Note:** The **Match String** function is case sensitive.

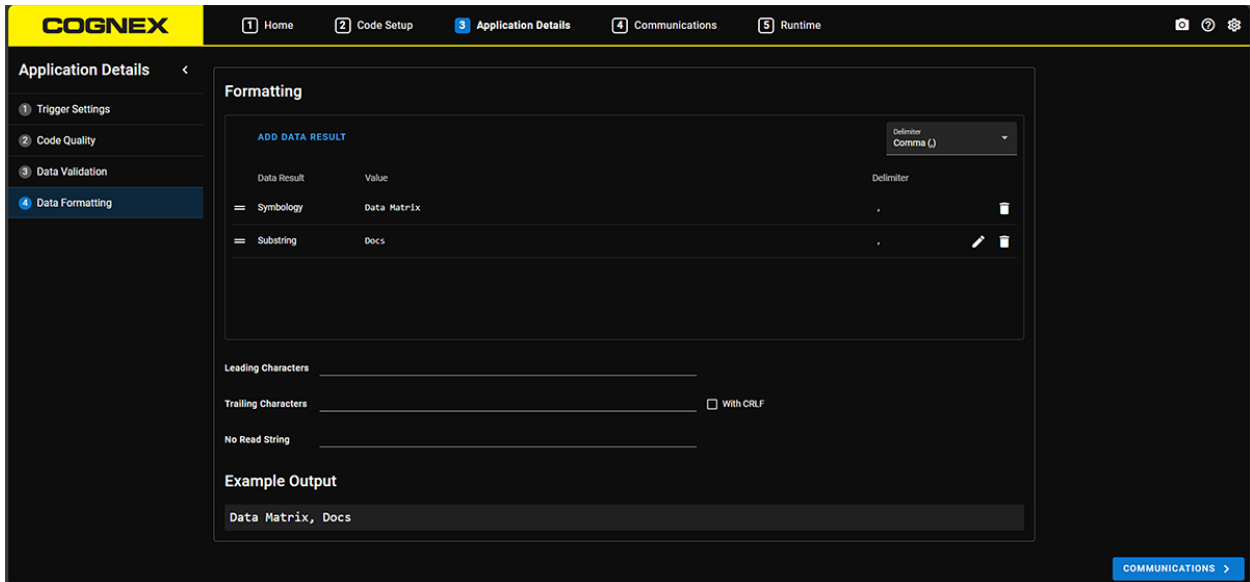
If a code fails validation, you can configure the reader to output a specific string in the **Outputs** substep.

Choose one the following options for the Validation Failure Action:

- **Transmit “Validation Failure”:** The reader outputs the string “*Validation Failure*” after a code fails validation.
- **Transmit “Validation Failure” + decoded string:** The reader outputs the string “*Validation Failure*” followed by the complete decoded string.
- **Transmit “Validation Failure” + quoted decoded string:** The reader outputs the string “*Validation Failure*” followed by the quoted part of the decoded string.
- **Transmit annotated decoded string:** The reader outputs the decoded string along with information on the error type and error location in the string.
- **Transmit nothing:** The reader outputs no information when data validation fails.

Data Formatting

In the **Data Formatting** step, you can construct a customized output string using pre-defined building blocks. You can also test your output before proceeding to the next step.



The output string starts with the text you enter in the **Leading Character** field. It continues with the **Data Results** you add, in the order you define. Each item is separated by the **Delimiter**. The output string ends with the text in the **Trailing Character** field.

To construct the output string:

1. Click **Add Data Result** to add a new item to the data formatting.
2. Reorder your data by dragging items to their new position.
3. Configure data formatting options that apply to the whole output string.
4. Open the **Delimiter** dropdown and select what string delimiter you want to use.
5. In the **String Options** section, define the leading characters, trailing characters, and No Read string you want to use.
6. Make sure to review your final output in the **Example Output** field before moving on to the next step.

You can configure options that impact that overall output string, such as specifying the **Leading Character** and the **Trailing Character**. The reader transmits the **No Read String** if it cannot read a given code. This can be useful for verifying that the reason you do not see any output is because the reader did not read data.

You have to configure an SNTP server to add timestamps to the output string.

Data items to construct your output string:

Category	Data	Description
----------	------	-------------

Result	Full string	All of the characters encoded in the symbol.
	Sub-string	A sub-string of the characters encoded in the symbol. You must specify a range for the sub-string when using this data. In Simple mode, you have to provide the index of starting and ending characters to construct the sub-string. In Advanced mode, you can provide a range, such as <i>1-3,5,8,10-</i> , for selecting the characters from the full string.
	Symbology	The symbology of the decoded symbol.
Formatting	Space	A space character.
	CR/LF	Carriage Return with Line Feed.
	Tab	A tab character.
Result Details	Symbology Identifier	The symbology identifier as specified by ISO/IEC 15424:2008.
	Symbol Size	The size of the symbol in pixels.
	Module Size	The width or height, in pixels, of each module in the symbol.
	Code Position	The image coordinates of each corner of the decoded symbol.
	Code Center	The image coordinates of the center of the decoded symbol.
	Code Orientation	The orientation of the decoded symbol reported in degrees. Clockwise rotations of the decoded symbol are indicated as positive changes in orientation.
	Contrast	A value, between 0 and 1, to indicate the total amount of contrast within the decoded symbol. A high value indicates the light and dark modules in the symbol have a wide range of grey values, while a low value indicates the light and dark modules consist of similar grey values.
	Unused Error Correction	The amount of available unused error correction in the symbol. Error correction is a method of reconstructing or replacing data that is lost through symbol damage. A value of <i>100</i> is ideal. A 1D symbol always returns <i>-1</i> .

TruCheck™	TruCheck™ Application Standard Name	The name of the application standard that is used for grading the code.
	TruCheck™ Application Standard Result	The result based on the application standard.
	TruCheck™ Grading Standard Name	The name of the grading standard.
	TruCheck™ Overall Grade Letter	The letter of the overall grade, for example, A.
	TruCheck™ Overall Grade Value	The value of the overall grade, for example, 3.9.
	TruCheck™ XDimension (mil)	The value of the custom X dimension in mils.
	Metadata	Input string
Passed Validations		The number of validation attempts the decoded symbol passed.
Failed Validations		The number of validation attempts the decoded symbol failed.
Result source		The name of the reader.
Code Quality Grade		The grade that the code receives during validation.
Timing	Decode time	The amount of time, in milliseconds, it took the reader to decode the symbol. This period of time does not include image acquisition, pending image processing, communication time and additional processing such as data formatting.
	Trigger time	The amount of time, in milliseconds, that elapses between receiving the trigger and completing the decode operation.
	Filter time	The duration of filtering in milliseconds.

Timestamp	Trigger Creation Local Time	The local time when the reader created the trigger.
	Trigger Creation UTC Time	The UTC time when the reader created the trigger.
	Trigger Execution Local Time	The local time when the reader executed the trigger.
	Trigger Execution UTC Time	The UTC time when the reader executed the trigger.
	Image Acquisition Local Time	The local time when the reader acquired the image of the symbol.
	Image Acquisition UTC Time	The UTC time when the reader acquired the image of the symbol.
	Subresult Creation Local Time	The local time when the reader created the subresult.
	Subresult Creation UTC Time	The UTC time when the reader created the subresult.
	Current Local Time	The current local time.
	Current UTC Time	The current UTC time.

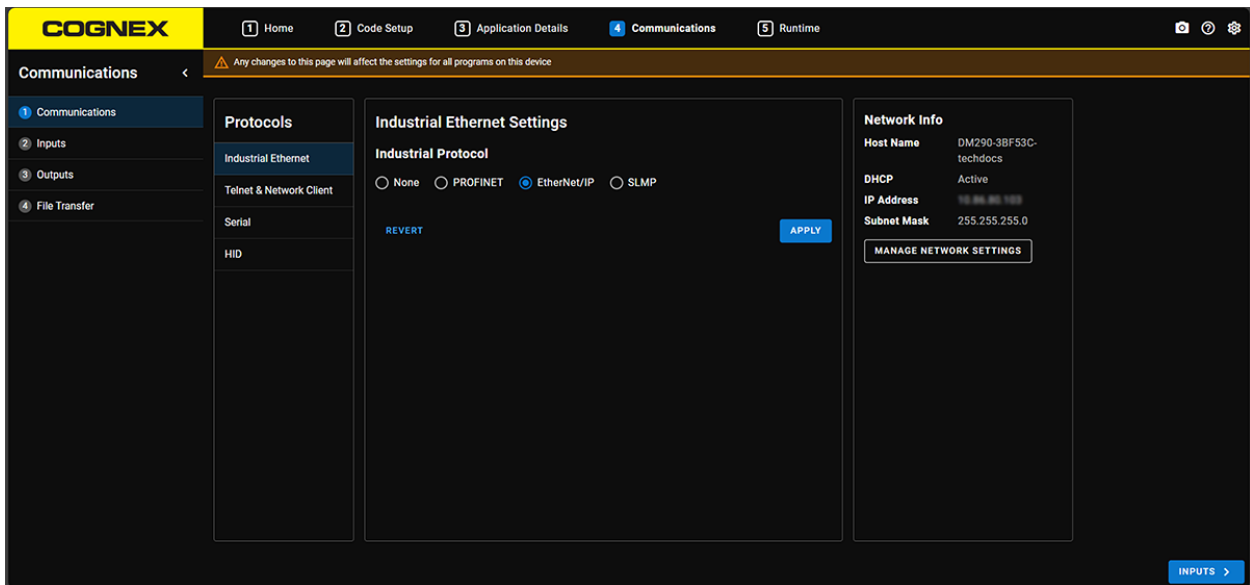
Communications

The **Communications** step allows you to configure the network, I/O, and file transfer settings for your reader. Follow the substeps in the menu on the left side to configure your communications settings.

Note: Any changes you make on the **Communications** step are applied to all programs on the reader.

Communications

The **Communications** step allows you to configure the communication protocols of your reader.



Industrial Ethernet

Industrial protocols enable the reader to exchange I/O data, alarms, and diagnostics with a PLC. Configure the industrial protocol you want to use:

Protocol	PLC
PROFINET	Siemens S7
EtherNet/IP	Rockwell ControlLogix
SLMP	Mitsubishi Q, iQ and L series

To select the protocol, click **Apply**. To disable a protocol, click **Revert**. This reverts the settings to the previous state. When changing industrial protocol settings, you have to restart the reader for the changes to take effect.

For more information, see the *Industrial Protocols Manual*.

Telnet & Network Client

Configure the **Telnet Port** to be used for Telnet communication.

Disable the **Telnet Port** to stop sending data from the reader through Telnet port. Disabling Telnet push messages still supports DMCC communication through the Telnet port.

The **Network Client** behavior is a means for the reader to actively open an Ethernet connection to another device on the network. The connection has the same basic operational behavior as a Telnet connection: the reader can transmit results to the external device and process DMCC commands received from the external device.

To use the reader as a network client, enable the **Network Client** toggle and provide the **Host Address**.

Serial Settings

Set the values to match the reader serial settings to the device you are connecting to:

- Baud Rate
- Data Bits
- Stop Bits
- Parity
- Handshake

HID Settings

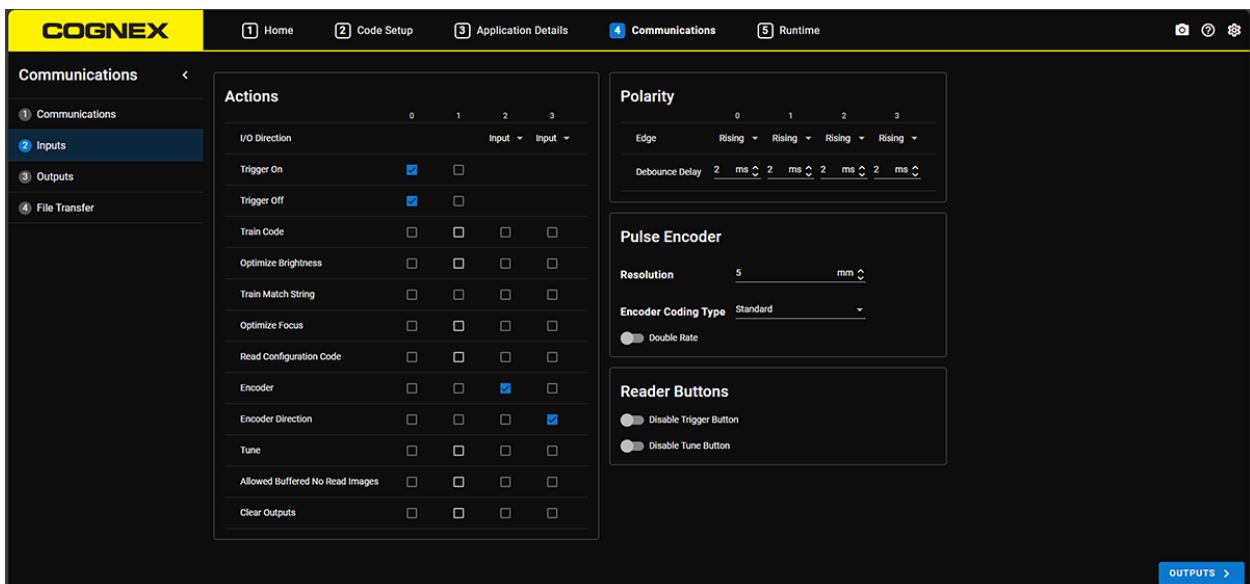
Turn on the **Enable HID interface** toggle.

- **Keyboard language:** Set the language. For Alt Key Codes, you can enter special characters using the **Alt** key and the numeric keypad. The read results are then interpreted as numeric keystrokes.
- **Inter-character delay:** Set the delay time between characters when in HID keyboard mode.
- **Keyboard report interval:** Set the delay between key down and key up HID messages.

Note: Additional communications protocols are available through the DataMan Setup Tool.

Inputs

The **Inputs** substep allows you to configure what action the reader takes when it receives an input signal.




By default, the reader responds to a signal over input line 0. The reader only supports trigger signals on input lines 0 and 1.

You can configure the reader to take any of the following actions:

- **Trigger On** and **Trigger Off** are enabled by default on the same input line 0. **Trigger Off** is automatically the inverse of **Trigger On**. If you put the two signals on separate lines, you can set the polarity independently according to your needs.
- **Train Code:** the reader is trained with the expected symbology, which can decrease the time required to decode subsequent symbols.

- **Optimize Brightness:** the reader finds the best possible light and imager settings to read a code.
- **Train Match String:** the reader uses the decoded code to match and validate subsequent codes against.
- **Optimize Focus:** the reader adjusts to the best possible focus settings to read a code. This option is only available for readers with liquid lens.
- **Read Configuration Code:** the reader reads the code as a configuration command instead of a result that needs to be reported. This read does not increase trigger count and does not produce a NoRead result.
- **Encoder:** connect your encoder to this input line.
- **Encoder Direction**
- **Tune:** the reader finds the best possible settings to read a code. This is the equivalent of rerunning the Tuning substep.
- **Allow Buffered No Read Images:** when this input is active and connected to a sensor that detects a part or package, the reader buffers No Read images if it is configured for no-read image saving. This is useful, for example, when you only want to store No Read images if an actual part or package is in the field of view of the reader.
- **Clear Outputs:** all outputs are reset immediately if the input gets triggered. This option does not reset any output delay settings.

 **Note:** The input line actions Train Code, Optimize Brightness, Train Match String, Read Configuration Code, and Tune are carried out only with the next trigger.

You can also configure the following signal detection options for each input line:

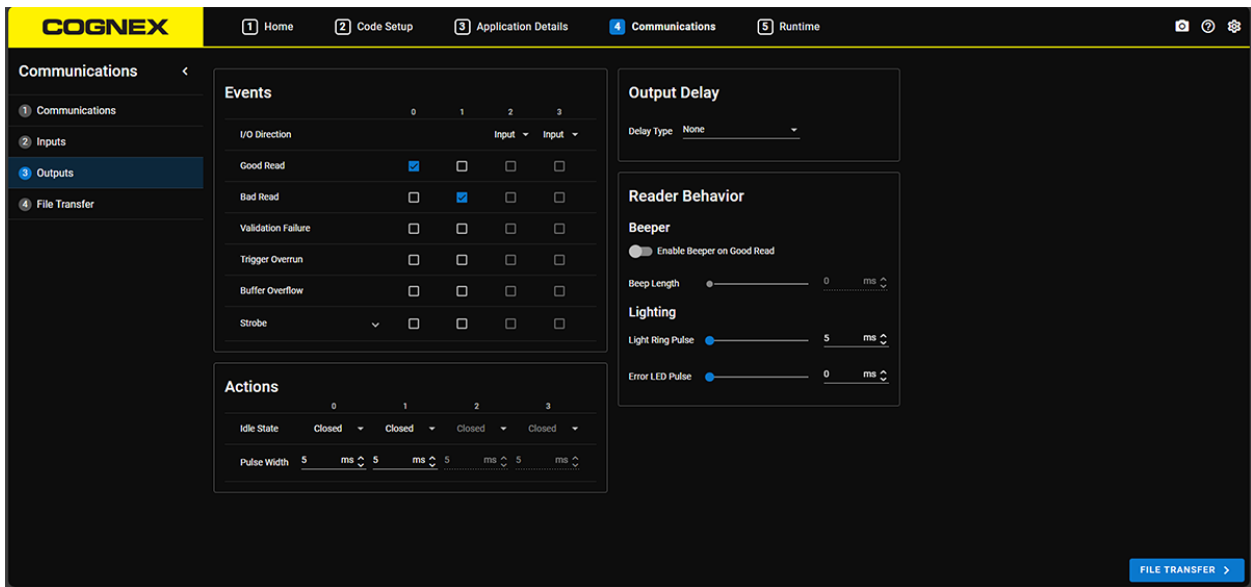
- The **Polarity** option allows you to set the change in the input signal that triggers an image acquisition.
- The **Debounce Delay** text box allows you to define how long the trigger signal must be detected in order to be recognized as valid. Use a shorter value to compensate for ESD line noise, and higher values to compensate for noise in electromechanical relays. Experiment with different **Debounce Delay** values to determine the most appropriate value for your production environment.

You can also configure whether the buttons on the reader are active. Turn on the **Disable Trigger Button** or **Disable Tune Button** toggles to prevent the reader from initiating a read, train, or tune attempt when you press the physical button on the reader.

Outputs

The **Outputs** substep allows you to configure what output signals the reader produces for different events.

You can set up an output delay of either time or distance, and update reader beeper and lighting behavior for different events such as Good Reads or Bad Reads.



I/O Direction

Configure whether I/O lines 2 and 3 behave as an Input or Output. Lines 0 and 1 are fixed output lines.

Good Read and Bad Read

If data validation is activated, the reader only signals a Read event if a code was read and passed validation. With Good Read and No Read events assigned to two outputs, the following logic applies:

- Good read, validation pass: **Good Read** output
- Good read, validation failure: **no** output, or Validation Failure if enabled
- Bad read: **Bad Read** output

Validation Failure

The reader signals the **Validation Failure** event when a code fails data validation, and outputs the error string configured in the **Data Validation** substep.

Trigger Overrun

The reader signals the **Trigger Overrun** event when an image could not be acquired because the reader was busy. Possible reasons include an ongoing burst or a long exposure for a previous trigger.

Buffer Overflow

The reader signals the **Buffer Overflow** event when the reader can acquire an image, but does not have any available internal image memory to save it. This happens when images are acquired faster than they can be processed and cleared from the buffer. A possible reason is that the decoding time is longer than the trigger interval.

Strobe

You can strobe an external illumination by checking **Strobe** on one of the output lines.

File Transfer

You can configure the reader to automatically transfer images, result data, and code quality verification reports from its memory to a destination on your network. The settings in the **File Transfer** substep allow you to configure file transfer using the following methods:

- **FTP:** The reader transfers files to an FTP server you set up on your network, for example, using FileZilla. This option uses insecure HTTP, which means that the data is not encrypted.
- **FTP Mitsubishi GOT:** The reader transfers files through the FTP Server/Client function built into Mitsubishi hardware. This option enhances FTP file transfer on Mitsubishi GOT and other FTP servers by allowing:
 - Idle timeout through periodic NOOPs
 - Optional timeout to close an idle control connection
 - Mitsubishi GOT write mode support
- **SFTP:** Same as the FTP option but with secure file encryption. Use this option when transferring files to the cloud or over untrusted networks.
- **TCP:** For image transfer only. The reader transfers image data to a TCP server. The reader opens the connection and expects a listening socket on the receiving end. This is similar to FTP.
- **Local TCP Server:** For image transfer only. The reader acts as the TCP server. You configure a listening socket on the reader that waits for an incoming connection from your network. When using the **Local TCP Server** option, keep the following in mind:

- You can have maximum one open connection.
- No buffering takes place. You cannot retrieve images captured before the connection was established. You might lose images if the connection drops and is re-established.

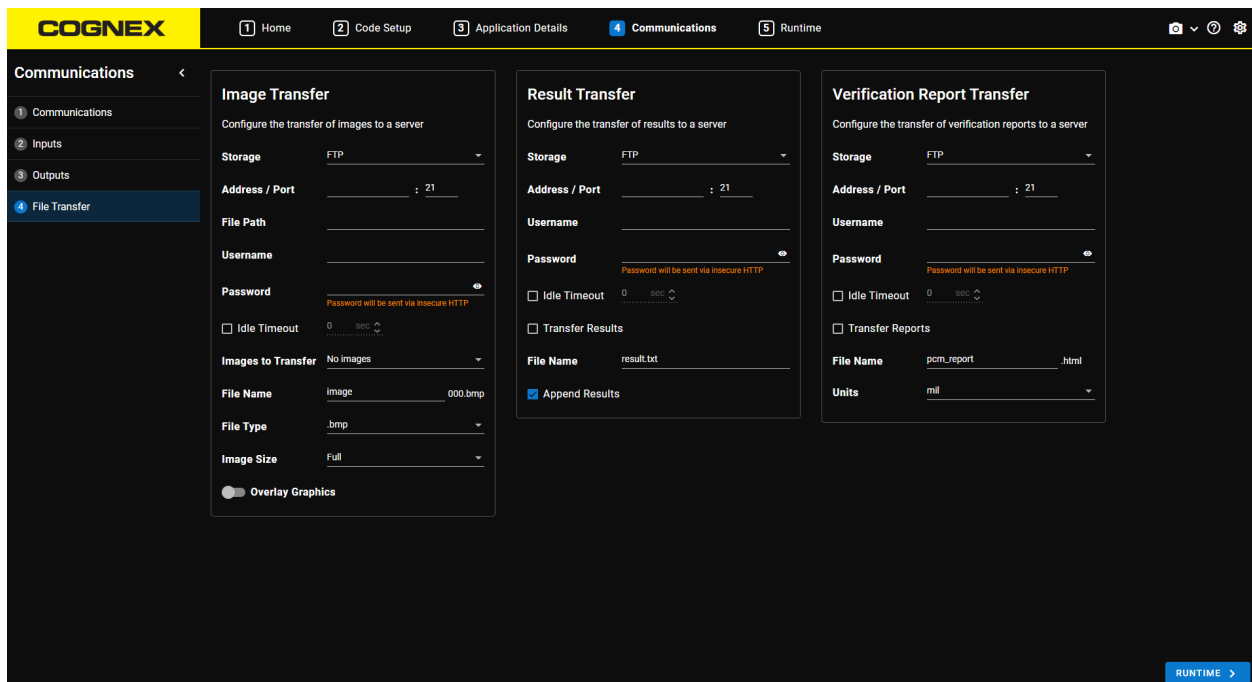


Image Transfer

The **Image Transfer** option allows you to transfer image files, with optional .svg overlay graphics, to a destination on your network.

The reader cannot guarantee that all images are sent. This can happen when the reader is operating at a higher frequency and cannot schedule time for transferring images.

The following options are available:

- **Storage:** the transfer method you want to configure for image transfer, such as FTP or TCP. Select the option that best fits the requirements of your network environment.
- **Server Address:** the address and port number of the file server on your network.
 - For **FTP** connections, normally you do not need to change the default port.
 - For **Local TCP Server** connections, the address field is not available because the reader acts as the TCP server. You only need to specify the port the reader leaves open to listen for client requests.
- **SFTP Fingerprint** (only available with SFTP option): the unique hash of the public key for the SFTP server.
- **File Path:** the file path in relation to the root directory of the FTP server, for example, */failed_images/image*. The directory must be available and accessible. The reader cannot create directories on the file server.
- **Username:** the name of the user that the reader can use to access the FTP server.
- **Password:** the password for the user that the reader can use to access the FTP server.
- **Idle Timeout:** enable or disable an idle timeout, defined in seconds.
- **Images to Transfer:** select the category of images to transfer based on the inspection results.
- **File Name:** define the name to use for transferred images, appended with a three-digit index.
- **File Type:** select the image type (*.bmp*, *.jpg*, or *.png*) for the transferred images.
- **Image Size:** select the degree of downscaling, if any, for the transferred images.
- **Overlay Graphics:** enable or disable the SVG overlay of result graphics on the transferred images.

Result Transfer

The **Result Transfer** option allows you to append result data to a text file stored on an FTP server.

The following options are available:

- **Storage:** the transfer method you want to configure for result data transfer. TCP is not supported.
- **Server Address:** the address of the FTP server.
- **SFTP Fingerprint** (only available with SFTP option): the unique hash of the public key for the SFTP server.
- **Port Number:** the port number of the FTP server.
- **Username:** the name of the user that the reader can use to access the FTP server.
- **Password:** the password for the user that the reader can use to access the FTP server.
- **Idle Timeout:** enable or disable an idle timeout, defined in seconds.
- **Transfer Results:** enable or disable transferring results.
- **File Name:** allows you to specify the name of the file that collects the results. If you enable **Append Results**, make sure that the FTP user has Append permissions, which is different from Write permissions.

Transferring Images over TCP

TCP provides an alternative to FTP for transferring images from the reader. When you use TCP, you send image data directly to a remote destination on your network. This approach avoids saving images to a file server and then copying them to a database. Using TCP simplifies integration and reduces latency.

In contrast, FTP requires you to set up a file server, monitor the file share for new files, and copy each file to your database. This process is more complex and can slow down your workflow.

When you enable TCP image transfer, the reader sends image files over a TCP connection to a destination on your network. Your application on the remote destination receives the image data and processes each file as needed.

You can configure TCP on the reader as an alternative to FTP or SFTP. At the same time, you must implement an application on your network to receive and process images from the reader.

TCP Message Format

When you use TCP for transferring images, the reader sends each image using a simple binary message format. The TCP message consists of a header followed by the body, which contains the image data described in the header.

The header size is fixed. It contains information about the image size, image type, and the image file name. The body size is variable and depends on the image.

The following table describes the TCP message format and the size of each field:

Section	Contents	Size								
Header	Image Size	4 bytes								
	Image Type	4 bytes Valid values:								
		<table border="1"> <tbody> <tr> <td>0</td> <td>BMP</td> </tr> <tr> <td>1</td> <td>PNG</td> </tr> <tr> <td>2</td> <td>JPG</td> </tr> <tr> <td>9</td> <td>SVG</td> </tr> </tbody> </table>	0	BMP	1	PNG	2	JPG	9	SVG
	0	BMP								
1	PNG									
2	JPG									
9	SVG									
Image File Name	128 bytes									
Body	Image Data	Variable								

Sample TCP Server for TCP Option

To help you implement a TCP server that receives images from the reader, the following Python sample script demonstrates how to read and process the TCP message format described above.

The script performs the following steps:

1. Accept a TCP connection from the reader.
2. Read the fields of the fixed-size header.
3. Read the image data using the image size to determine how many bytes to receive.
4. Process or store the image.

By following the structure of the TCP message format, you can adapt the sample script to the requirements of your application.

```

"""
This code is provided as an example. It is not intended for use in a production environment
without further testing and validation. The author and the organization they represent are not
responsible for any issues
that may arise from using this code.
"""

import socket
import struct

```

```
# host machine IP address and port
server_ip = "<insert host ip>"
server_port = 4444

# Create a socket object
s = socket.socket(socket.AF_INET, socket.SOCK_STREAM)

# Bind the socket to a specific address and port
s.bind((server_ip, server_port))

# Start listening for incoming connections
s.listen(1)

print("Server is listening on port", server_port)

filesReceived = 0

try:
    # Receive data from the client
    while True:
        # Accept a connection
        print("Waiting on connection...")
        c, addr = s.accept()
        print("Got connection from", addr)

        data = ""

        while True:
            # Read header
            data = c.recv(8 + 128)
            print("Received header length", len(data))
            if not data:
                print("Header not received, aborting read")
                break

            fileSize = 0
            fileSizeBuffer = data[:4]
            fileSize = struct.unpack("<I", fileSizeBuffer)[0]
            print("FileSize", fileSize)
            fileTypeBuffer = data[4:8]
            fileType = struct.unpack("<I", fileTypeBuffer)[0]
            print("FileType", fileType)
            fileNameBuffer = data[8:]
            fileName = struct.unpack("128s", fileNameBuffer)[0]
            print("FileName", fileName.decode())
```

```
# Read data in 1024 packets

received = 0
while received < fileSize:
    readPacketSize = 1024
    if fileSize - received < 1024:
        readPacketSize = fileSize - received
    data = c.recv(readPacketSize)
    received += len(data)

    if not data:
        print("Got bad data packet aborting file read")
        break

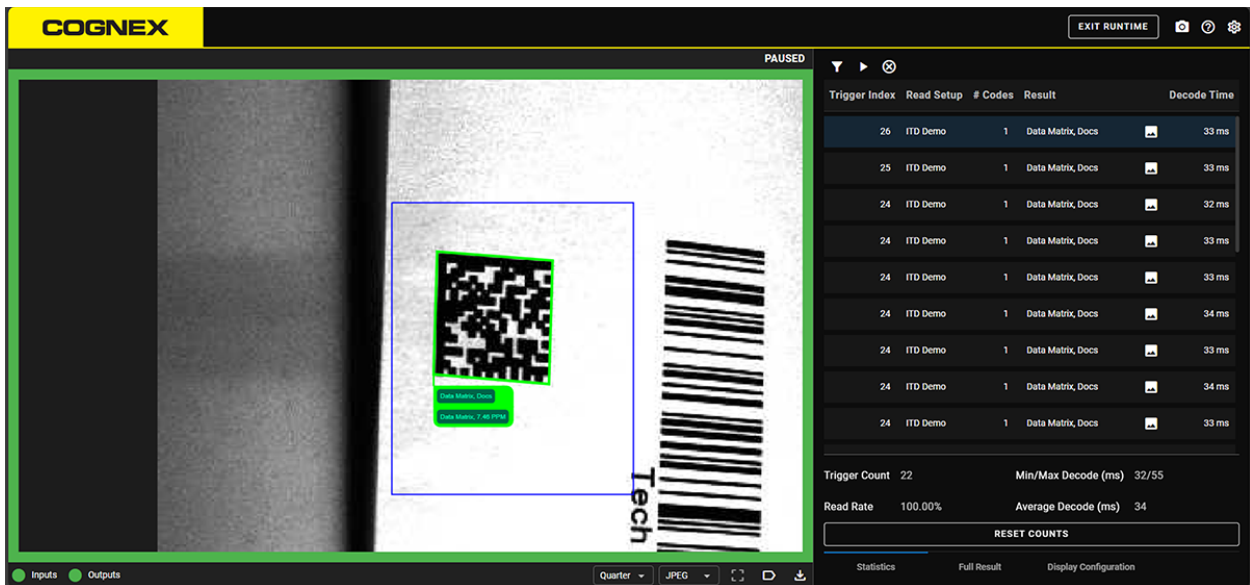
# Save the file
filesReceived += 1
print("Files Received", filesReceived, "Last file size:", fileSize)
print("")

except KeyboardInterrupt:
    print("\nExiting")
```

Runtime

The **Runtime** step allows you to monitor the read results and performance of the reader in production.

The **Runtime** step provides a summary table showing trigger counts, current read rates, and decode time statistics.



The result table lists the read codes. The results table can display the **Full Result** string, provide **Statistics** about reader performance, or display the SBG results with **TruCheck™**. Click **Display Configuration** to customize the data shown for individual reads and overall statistics.

You can filter, pause, or clear the incoming result data.

Click on a result with the image icon to display the image on the screen.

The screen shows the last image of the reader, or the image acquired for the selected read. You can move the image within the screen, and zoom with the mouse wheel.

Enable the **Result Status** option on the **Display Configuration** tab for more detailed status information. The **Status** column appears with the following icons to indicate different read results:

Icon	Meaning
	Good read
	Used AI enhancer (feature key required)
	Validation failure
	Code found but could not be read
	No read

Configure the displayed image with the buttons on the bottom of the screen:

- Choose the image resolution and format.
- Fit the image to the screen.

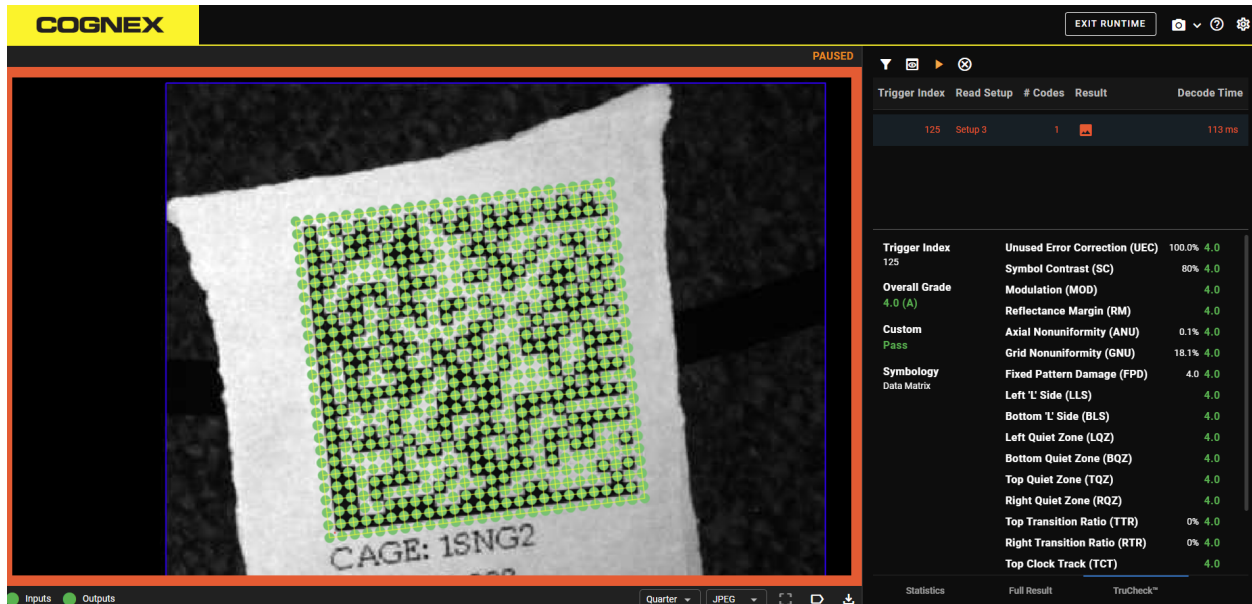
- Turn labels on or off.
- Download the latest image in full resolution.

 **Note:** You can password protect the UI through the **Settings** menu to make only the **Home** and **Runtime** steps accessible. For more information, see [Password Protection on page 57](#).

TruCheck™ Grading

TruCheck™ is a feature that implements ISO/IEC 15415, ISO/IEC 15416, and ISO/IEC 29158 barcode quality grading. TruCheck™ is available on verifiers and select readers. When you use TruCheck™ on a reader that does not use strictly compliant lighting, the results are called Standards-Based Grading (SBG). SBG provides grades based on an image captured by a reader that might not be ISO-compliant, but calculates the grades based on ISO standards. Therefore, the results of SBG can correlate to ISO-compliant verification results depending on the reader setup.

SBG is useful in applications where ISO compliance is not possible, for example, when there is not enough space to mount a reader and lighting at required distances and angles. SBG setups might also provide a Field of View or resolution that is different from that of verifiers.

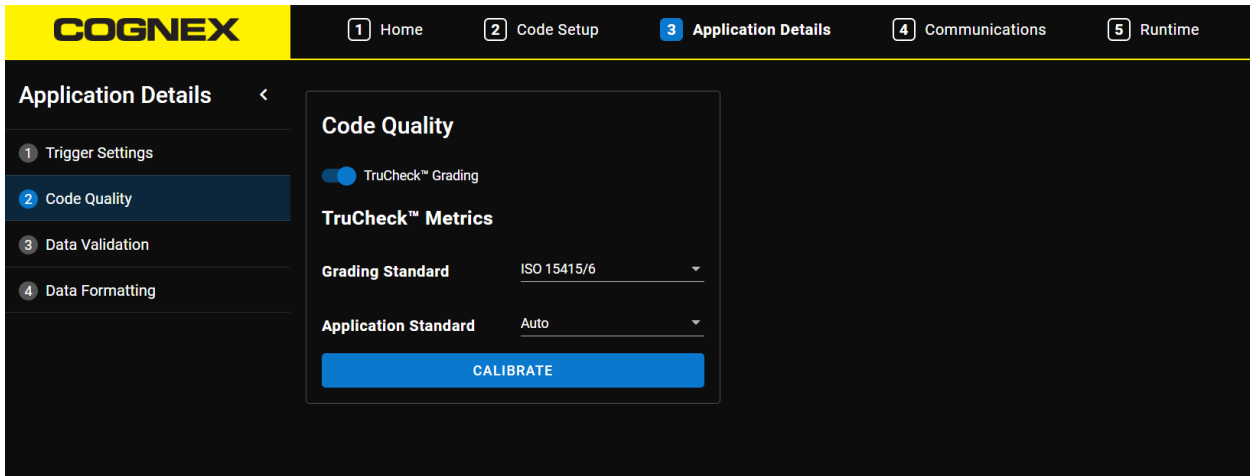


To use TruCheck™ Grading, set up your reader using the regular application steps and perform the TruCheck™-specific actions in the following substeps:

- **Code Quality:** Enable and configure TruCheck™ Grading. For more information, see [Configuring Grading on page 39](#).
- **Data Formatting:** Add data results from the TruCheck™ category to the string. For more information, see [Data Formatting on page 24](#).
- **File Transfer:** Set up the **Verification Report Transfer** to send SBG results to an FTP server. For more information, see [Transferring Verification Reports on page 42](#).
- **Runtime:** Check the SBG quality parameters under the TruCheck™ tab. For more information, see [Examining TruCheck™ Results on page 43](#).

Configuring Grading

The **Code Quality** substep includes the settings for enabling and configuring TruCheck™.



Setting up TruCheck™

To set up TruCheck™, do the following:

1. Enable **TruCheck™ Grading**.

As a result, the TruCheck™-specific options appear and the process control metrics are disabled.

2. Select the **Grading Standard** and **Application Standard**.

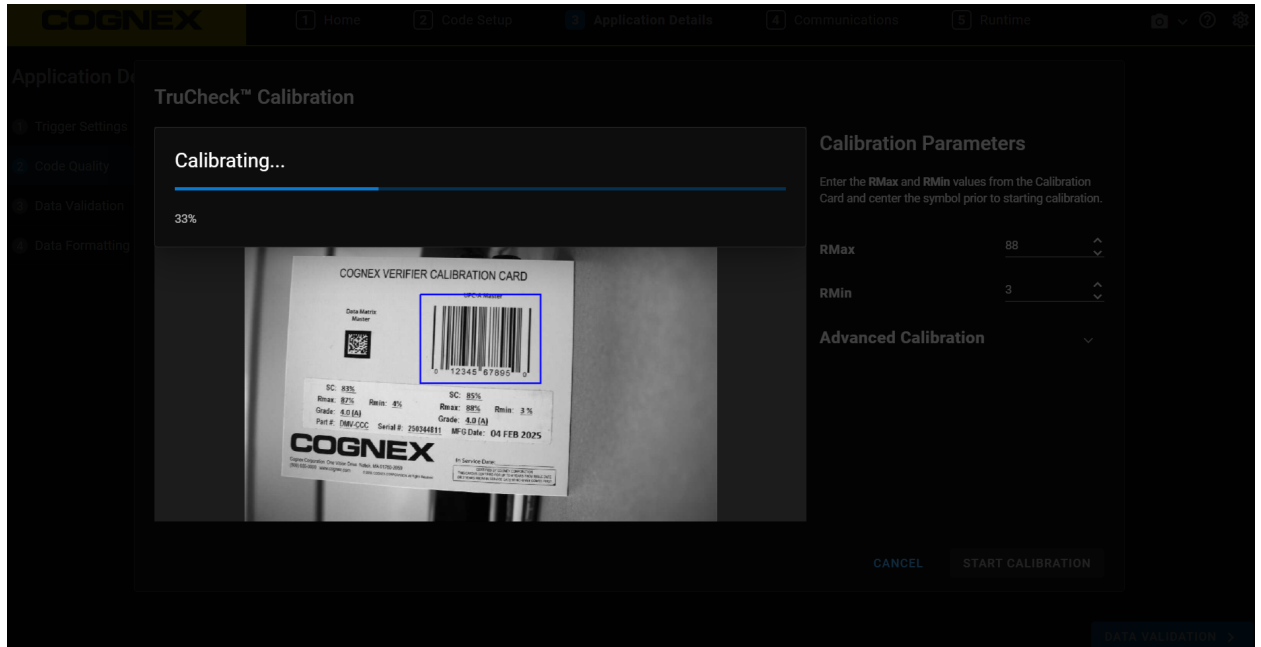
The **Dot Peen** option is available if you select **ISO 29158 (AIM-DPM)** for the **Grading Standard**.

3. Click **Calibrate**.
4. Set the **RMax** and **RMin** values to the values that your calibration card displays.
5. Place the code in the center of the Field of View and edit the ROI.



Tip: If you have the DMV-CCC Cognex Calibration Card, you can use either the UPC or the Data Matrix code for calibration, regardless of your application. If the Field of View is not large enough for the UPC code, you can use the Data Matrix code.

6. Click **Start Calibration**.



7. When the calibration is complete, click **Save Calibration**.

Application Standards

The following table details the application standards you can select when setting up TruCheck™ Grading:

Application Standard	Description
GS1	The GS1 application standard follows GS1 General Specification guidelines in code verification.
HIBCC	The HIBCC application standard follows the HIBCC standard in code verification.
UDI (GS1 or HIBCC)	The UDI (GS1 or HIBCC) application standard checks symbols that meet UDI requirements for medical devices using GS1 or HIBCC guidelines.
UID (MIL-STD-130)	The UID (MIL-STD-130) application standard reports the quality standard according to the MIL-STD-130, specifying UID marks Construct 1 and Construct 2 that use data structure and code grade for verification. The MIL-STD-130 defines acceptable grades and requirements for data format.
Custom	<p>The Custom application standard option allows you to customize the settings for grading. This option is useful when the code grading is not expected to adhere to any predefined industry conformance standards. The option has the following settings:</p> <ul style="list-style-type: none"> • Minimum and Maximum X-dimension • Overall Pass Grade • Data Format Check • Aperture Setting <p>Note: The Aperture Setting option is automatically set to Auto Aperture when the Grading Standard is ISO 29158 (AIM-DPM).</p>

Application Standard	Description
Auto	The Auto application standard option allows the reader to select the correct application standard based on the format of the data encoded in the symbology. If the reader detects GS1, HIBCC, or MIL-STD-130 standards within a symbology during verification, the reader uses the correct application standard.
Cryptocode	The Cryptocode application standard checks symbols that meet Russian Cryptocode standard or other Cryptocode applications.

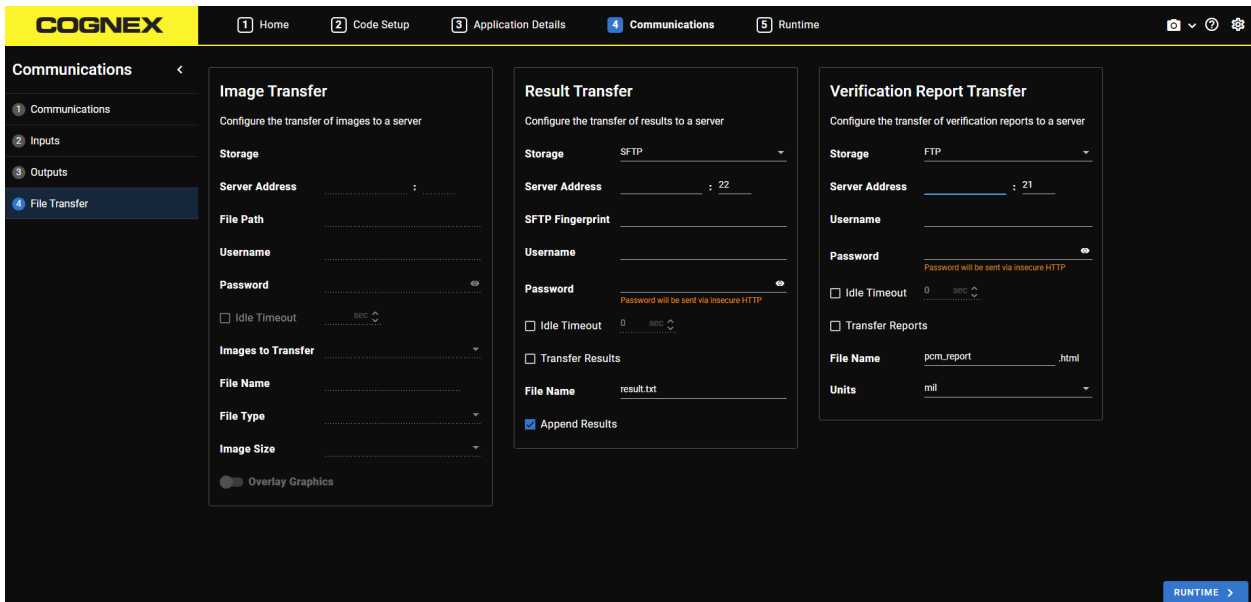
Calibration Process

Calibration sets reflectance and spatial references by checking grey scale levels to measure contrast and pixel size to measure code dimensions.

Note: You must use a code for calibration that has known RMin and RMax values, for example, a Cognex Calibration Card, such as the DMV-CCC or DMV-DMCC calibration cards.

Transferring Verification Reports

The **File Transfer** substep includes an option to send verification reports with a selected protocol with TruCheck™ Grading. The settings to configure sending the TruCheck™ results appear here if you enable **TruCheck™ Grading** in the **Code Quality** substep.



The **Verification Report Transfer** option allows you to send a report with code quality or TruCheck™ details to an FTP server. Every trigger generates a new report file. These settings are only available if you have enabled any of the verification settings under **Code Quality** in the **Application Details** step.

The following options are available:

- **Storage:** the transfer method you want to configure for result data transfer. TCP is not supported.
- **Server Address:** the address and port number of the FTP server.
- **SFTP Fingerprint** (only available with SFTP option): the unique hash of the public key for the SFTP server.
- **Username:** the name of the user that the reader can use to access the FTP server.

- **Password:** the password for the user that the reader can use to access the FTP server.
- **Idle Timeout:** enable or disable an idle timeout, defined in seconds.
- **Transfer Reports:** enable or disable transferring reports.
- **File Name:** allows you to specify the name of the file that collects the results.
- **Units:** specifies the unit of measurement. Available if TruCheck™ Grading is enabled.

Examining TruCheck™ Results

The **Runtime** step displays TruCheck™ results for the selected trigger. This includes, for example, **Trigger Index**, **Symbology**, and SBG quality parameters. The grading results appear below the result table, the **TruCheck™** tab is available when you enable **TruCheck™ Grading** in the **Code Quality** substep.

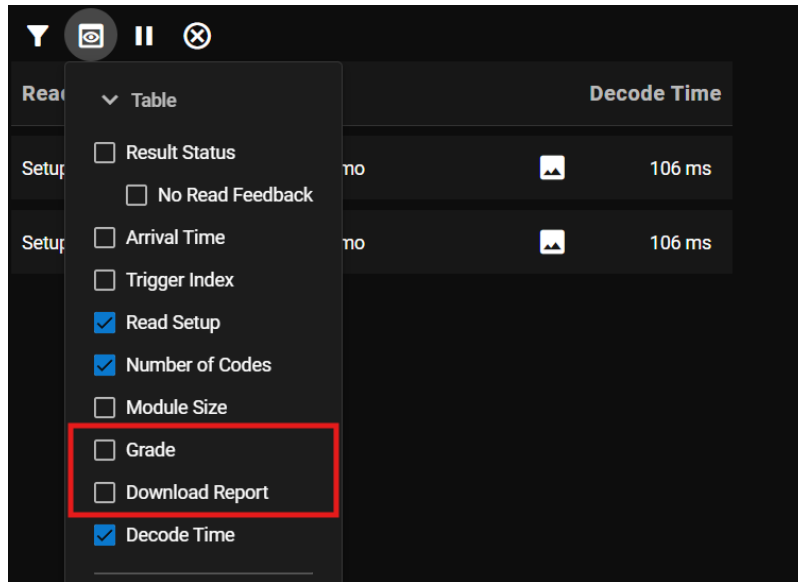
Trigger Index	Read Setup	# Codes	Result	Decode Time
125	Setup 9	1		113 ms

Trigger Index	Unused Error Correction (UEC)	100.0%	4.0
125	Symbol Contrast (SC)	80%	4.0
Overall Grade	Modulation (MOD)		4.0
4.0 (A)	Reflectance Margin (RM)		4.0
Custom	Axial Nonuniformity (ANU)	0.1%	4.0
Pass	Grid Nonuniformity (GNU)	18.1%	4.0
Symbology	Fixed Pattern Damage (FPD)		4.0
Data Matrix	Left 'L' Side (LLS)		4.0
	Bottom 'L' Side (BLS)		4.0
	Left Quiet Zone (LQZ)		4.0
	Bottom Quiet Zone (BQZ)		4.0
	Top Quiet Zone (TOZ)		4.0
	Right Quiet Zone (RQZ)		4.0
	Top Transition Ratio (TTR)	0%	4.0
	Right Transition Ratio (RTR)	0%	4.0
	Top Clock Track (TCT)		4.0

The tab shows the quality parameter grades in letter or number form. The format depends on the version of the standard. The following revisions replaced the letters, ranging from F to A, with numbers, ranging from 0.0 to 4.0 in increments of 0.1:

- ISO/IEC 15416:2016
- ISO/IEC 15415:2024
- ISO/IEC TR 29158:2020

With TruCheck™ enabled, the result table can display the grade of the trigger, and the verification report for the latest trigger can be downloaded. To see these columns in the result table, enable them in the **Display Configuration**:



Note:

When you download the verification report, consider the following:



- The download button only appears for the latest report and if the column is enabled.
- Depending on your browser settings, the browser might warn you that the download is not secure. This is because the reader uses HTTP, not HTTPS.

ISO/IEC 15416 Quality Parameters

The ISO/IEC 15416 standard defines the grading standards for 1D barcodes.

Trigger Index 154	Edge	0 F
	Reflectance Light / Reflectance Dark (RI/Rd)	0/0 F
Overall Grade 0.0 (F)	Symbol Contrast (SC)	-
	Minimum Edge Contrast (MinEC)	-
GS1 Fail (Quality)	Modulation (MOD)	74% A
	Defect (Def)	80% F
Symbology Code 128	Decode (DCD)	0/10 F
	Decodability (DEC)	0% F
	Minimum Quiet Zone (MinQZ)	1 F

Statistics Full Result **TruCheck™**

The following table lists and details the quality parameters for ISO/IEC 15416 that the **TruCheck™** tab shows in the **Runtime** step.

Quality Parameter	Parameter Description	Value Description	Grading
Edge	Edge determination. The parameter checks that the reader detects each edge between bars and spaces. The edges can blur due to splitting of bars or ink splattered into spaces.	The Edge value is the number of bars and spaces without quiet zones.	Pass/Fail grade (A or F , respectively). Pass grade if the number of bars and spaces is correct.
Reflectance Light / Reflectance Dark (RI/Rd)	The ratio of minimum reflectance. The parameter checks whether the darkness of the bars is sufficient. The light reflected by the bars must be less than half of the light reflected by the spaces.	The first value represents the lightest value in spaces. The second value represents the darkest value in bars.	Pass/Fail grade (A or F , respectively). Pass grade if the reflectance is sufficient.
Symbol Contrast (SC)	The parameter measures the contrast between the brightest space and the darkest bar.	The value is a percentage of contrast. It ranges from 100, which is the highest contrast, to 0, which is the lowest contrast.	A > 70 B 55 — 69 C 40 — 54 D 20 — 39 F < 20
Minimum Edge Contrast (MinEC)	The parameter checks the contrast between bars and spaces. As opposed to symbol contrast, minimum edge contrast checks for the worst contrast difference between each bar and space.	The value is a percentage of edge contrast.	Pass/Fail grade (A or F , respectively). Pass grade if the value exceeds 15%.
Modulation (MOD)	The parameter checks the edge contrast as the ratio of the minimum edge contrast and symbol contrast. It represents the consistency of the darkness of the bars and the lightness of the spaces.	The value is a percentage of modulation.	A >= 70 B 60 — 69 C 50 — 59 D 40 — 49 F < 30
Defect (Def)	The parameter checks for defects in bars and spaces. Out of all the changes in reflectance in bars and spaces, Defect takes the highest value, which is the worst one and could influence the readability of the code the most.	The value is a percentage of the worst change in reflectance of a single bar or space.	A <= 15 B 16 — 20 C 21 — 25 D 26 — 30 F > 30
Decode (DCD)	The parameter checks whether the code can be decoded based on a specific formula.	The value is a score out of ten.	Pass/Fail grade (A or F , respectively).

Quality Parameter	Parameter Description	Value Description	Grading
Decodability (DEC)	The parameter determines how accurate the bar and space widths are and how easy it is to determine the widths. The lower the score, the more inconsistent and distorted the bar and space widths are.	The value is a percentage of the overall tolerance range for a bar or space range that is accurate.	A >= 62 B 50 — 61 C 37 — 49 D 25 — 36 F < 25
Minimum Quiet Zone (MinQZ)	The quiet zone represents the empty spaces to the left and right of the code. Different symbologies have different requirements for the minimum quiet zone.	The value is the width of the blank spaces on either side of the code, measured in modules.	Pass/Fail grade (A or F, respectively). Pass grade depends on the symbology.

ISO/IEC 15415 Quality Parameters

The ISO/IEC 15415 standard defines the grading standards for 2D codes.

Trigger Index 125	Unused Error Correction (UEC)	100.0%	4.0
	Symbol Contrast (SC)	80%	4.0
Overall Grade 4.0 (A)	Modulation (MOD)		4.0
	Reflectance Margin (RM)		4.0
Custom Pass	Axial Nonuniformity (ANU)	0.1%	4.0
	Grid Nonuniformity (GNU)	18.1%	4.0
Symbology Data Matrix	Fixed Pattern Damage (FPD)	4.0	4.0
	Left 'L' Side (LLS)		4.0
	Bottom 'L' Side (BLS)		4.0
	Left Quiet Zone (LQZ)		4.0
	Bottom Quiet Zone (BQZ)		4.0
	Top Quiet Zone (TQZ)		4.0
	Right Quiet Zone (RQZ)		4.0
	Top Transition Ratio (TTR)	0%	4.0
	Right Transition Ratio (RTR)	0%	4.0
	Top Clock Track (TCT)		4.0

The following table lists and details the quality parameters for ISO/IEC 15415 that the TruCheck™ tab shows in the Runtime step.

Quality Parameter	Parameter Description	Value Description	Grading
Unused Error Correction (UEC)	Measures the remaining capability for error correction that is available for further incorrect modules.	The value is the percentage of the remaining error correction capability.	A > 62 B 50 — 61 C 37 — 49 D 25 — 36 F < 25
Symbol Contrast	The parameter measures the contrast between the brightest space and the darkest bar.	The value is a percentage of contrast. It ranges from 100, which is the highest contrast, to 0, which is the lowest contrast.	A > 70 B 55 — 70 C 40 — 55 D 20 — 40 F < 20
Print Growth (PG)	Measures the deviation from the original, intended sizes of the modules. Note: The Print Growth (PG) quality parameter is only present from ISO/IEC 15415:2024.	A percentage of deviation from the original size of modules.	4.0 <= 10 3.9 — 10.1 — 3.0 28 2.9 — 28.1 — 2.0 34 1.9 — 34.1 — 1.0 40 0.9 — 40.1 — 0.1 60 0.0 > 60
Modulation (MOD) and Reflectance Margin (RM)	Measures the amount of reflectivity of the modules. The grade is calculated using a formula that compares the reflectivity of each module to the global threshold and the overall symbol contrast. Note: The Reflectance Margin (RM) quality parameter is not displayed in ISO/IEC 15415:2024.	A percentage of modulation.	A > 50 B 40 — 49 C 30 — 39 D 20 — 21 F < 20
Axial Nonuniformity (ANU)	Measures the overall aspect ratio of the symbol and checks for uneven scaling of the modules.	A percentage of deviation from a square shape of the modules.	A <= 6 B 6 — 7 C 8 — 9 D 10 — 11 F >= 12

Quality Parameter	Parameter Description	Value Description	Grading
Grid Nonuniformity (GNU)	Measures the biggest deviation from the calculated center of the module and the ideal location for the center of the module.	Deviation from the center in percentage of module size.	A ≤ 38 B 39 — 50 C 51 — 63 D 64 — 74 F ≥ 75
Fixed Pattern Damage (FPD)	The overall grade of the fixed pattern components.	N/A	It is equal to the lowest grade among LLS, BLS, LQZ, BQZ, TQZ, and RQZ.
Left 'L' Side (LLS)	Measures imperfections in the 'L' finder pattern. The parameter checks the pattern for the following: <ul style="list-style-type: none"> The gaps must be three modules or less. The gaps must be separated by at least four correct modules. 	The percentage of incorrect modules.	A 0 B 1 — 9 C 10 — 13 D 14 — 17 F < 17
Bottom 'L' Side (BLS)	Measures imperfections in the 'L' finder pattern. The parameter checks the pattern for the following: <ul style="list-style-type: none"> The gaps must be three modules or less. The gaps must be separated by at least four correct modules. 	The percentage of incorrect modules.	A 0 B 1 — 9 C 10 — 13 D 14 — 17 F < 17
Left Quiet Zone (LQZ)	Measure imperfections in the quiet zone, which is a one-module area on either side of the code.	The percentage of incorrect modules.	A 0 B 1 — 9 C 10 — 13 D 14 — 17 F < 17
Bottom Quiet Zone (BQZ)	Measure imperfections in the quiet zone, which is a one-module area on either side of the code.	The percentage of incorrect modules.	A 0 B 1 — 9 C 10 — 13 D 14 — 17 F < 17
Top Quiet Zone (TQZ)	Measure imperfections in the quiet zone, which is a one-module area on either side of the code.	The percentage of incorrect modules.	A 0 B 1 — 9 C 10 — 13 D 14 — 17 F < 17

Quality Parameter	Parameter Description	Value Description	Grading
Right Quiet Zone (RQZ)	Measure imperfections in the quiet zone, which is a one-module area on either side of the code.	The percentage of incorrect modules.	A 0 B 1 — 9 C 10 — 13 D 14 — 17 F < 17
Top Transition Ratio (TTR)	Measures the imperfections in the transition from the clock track to the adjoining quiet zone.	The ratio of the number of transitions from light to dark or dark to light in the quiet zone divided by the number of transitions in the clock track.	A < 6 B 7 — 8 C 9 — 10 D 11 — 12 F > 12
Right Transition Ratio (RTR)	Measures the imperfections in the transition from the clock track to the adjoining quiet zone.	The ratio of the number of transitions from light to dark or dark to light in the quiet zone divided by the number of transitions in the clock track.	A < 6 B 7 — 8 C 9 — 10 D 11 — 12 F > 12
Top Clock Track (TCT)	Measures the imperfections in the clock track. Three out of five modules in every five modules in the clock track must be correct for the passing grade.	N/A	A grade of the highest modulation level for which the test passes.
Right Clock Track (RCT)	Measures the imperfections in the clock track. Three out of five modules in every five modules in the clock track must be correct for the passing grade.	N/A	A grade of the highest modulation level for which the test passes.
Average Grade (AG)	Average grade of damage across the finder pattern.	The average value of the following: <ul style="list-style-type: none"> • Lowest of values associated with the clock track segment: TCT, TTR, TQZ, TCT, RTR, and RQZ. • LLS • BLS • LQZ • BQZ 	A 4.0 B 3.5 — 3.9 C 3.0 — 3.4 D 2.5 — 2.9 F <= 2.5
Decode	The parameter checks whether the code can be decoded based on a specific formula.	The value is a score out of ten.	Pass/Fail grade (A or F, respectively).

ISO/IEC TR 29158 AIM-DPM-1-2006 Quality Parameters

The ISO/IEC TR 29158 standard is an extension of ISO/IEC 15415 for Direct Part Marking use case.

The following table lists and details the quality parameters for ISO/IEC TR 29158 that the **TruCheck™** tab shows in the **Runtime** step.

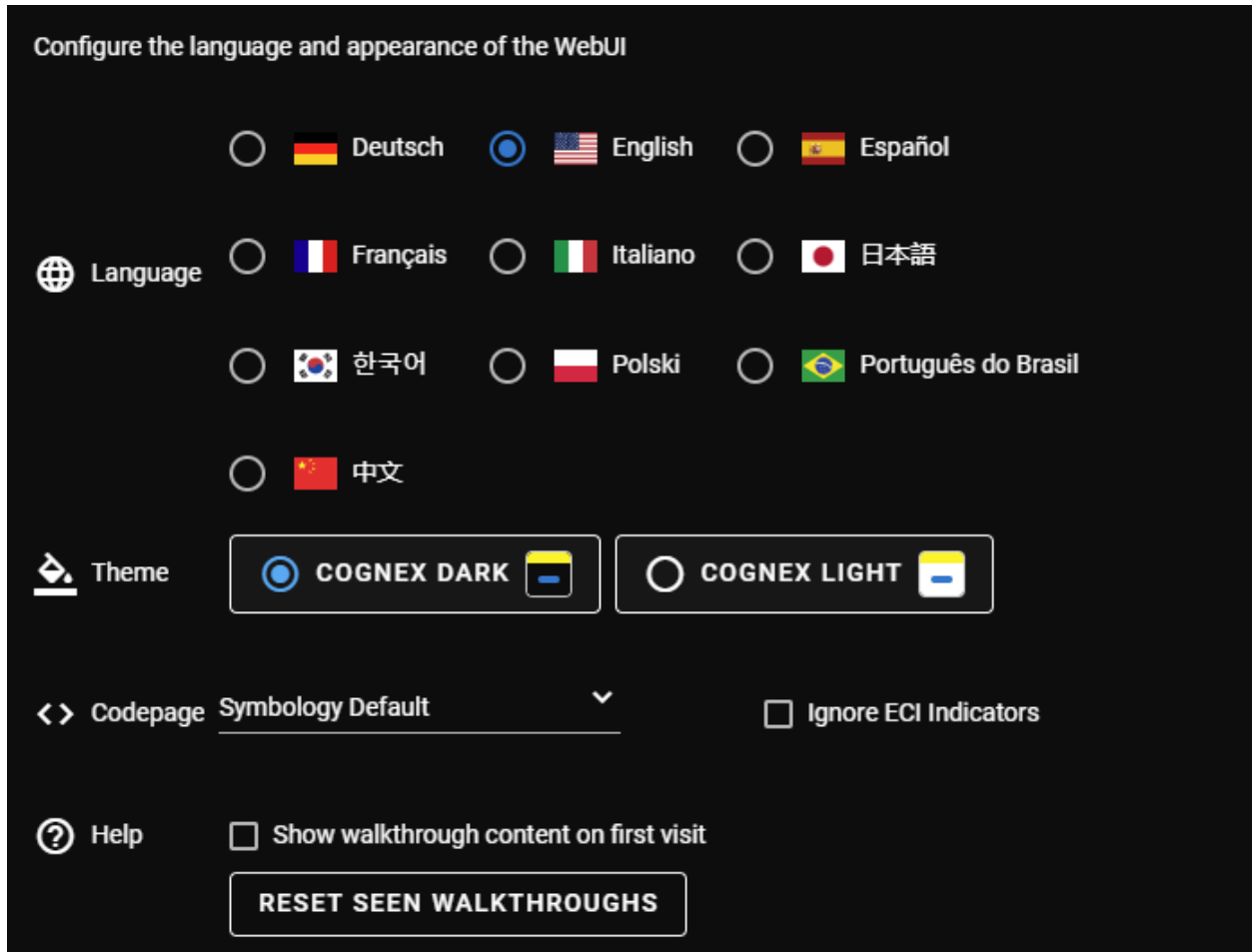
Quality Parameter	Parameter Description	Value Description	Grading
Cell Contrast (CC)	The relative contrast value between bars and spaces. It is the mean value of the light and dark elements.	A percentage of contrast value.	A >= 30 B 25 — 29 C 20 — 24 D 15 — 19 F < 15
Cell Modulation (CMOD)	Measures the amount of reflectivity of the modules. The grade is calculated using a formula that compares the reflectivity of each module to the global threshold and the overall symbol contrast. Each module is graded based on the reflectivity of the modules, then error correction capability is calculated to negate one or few elements with low values.	A percentage of modulation.	
Distributed Damage Grade (DDG)	Average grade of the fixed pattern elements.	N/A	The result of fixed pattern damage patterns.
Minimum Reflectance (MR)	The minimum reflectance between light and dark elements.	A percentage of the lowest value of reflectance between dark and light elements.	For ISO/IEC TR 29158:2011: Pass/Fail grade (A or F , respectively). The value must be at least 5% for a pass grade. For ISO/IEC TR 29158:2020 and later: 4.0 >= 20 3.5 — 15 — 4.0 19.9 2.5 — 10 — 3.4 14.9 1.5 — 5 — 2.4 9.9 1.4 — < 5 0.0

Quality Parameter	Parameter Description	Value Description	Grading
Decode	The parameter checks whether the code can be decoded based on a specific formula.	The value is a score out of ten.	Pass/Fail grade (A or F , respectively).

Settings

Overview






In the **Overview** tab of **Settings**, you can set up the language and theme of the WebUI. You can also customize the behavior of the in-application help.




Configure Codepage





You can change the codepage to be able to decode non-ASCII characters.

Click **Symbology Default** to open the drop-down of available codepages and select one of the options. Read at least one non-ASCII code and use the **Decoded Result** column to help determine which codepage is the right one for your reading. You can also use the **Filter** field to narrow down the list of available options or disable the **Show Invalid Decodings** checkbox to only show valid codepages.


COGNEX EXIT RUNTIME     




QR, 9.52 PPM

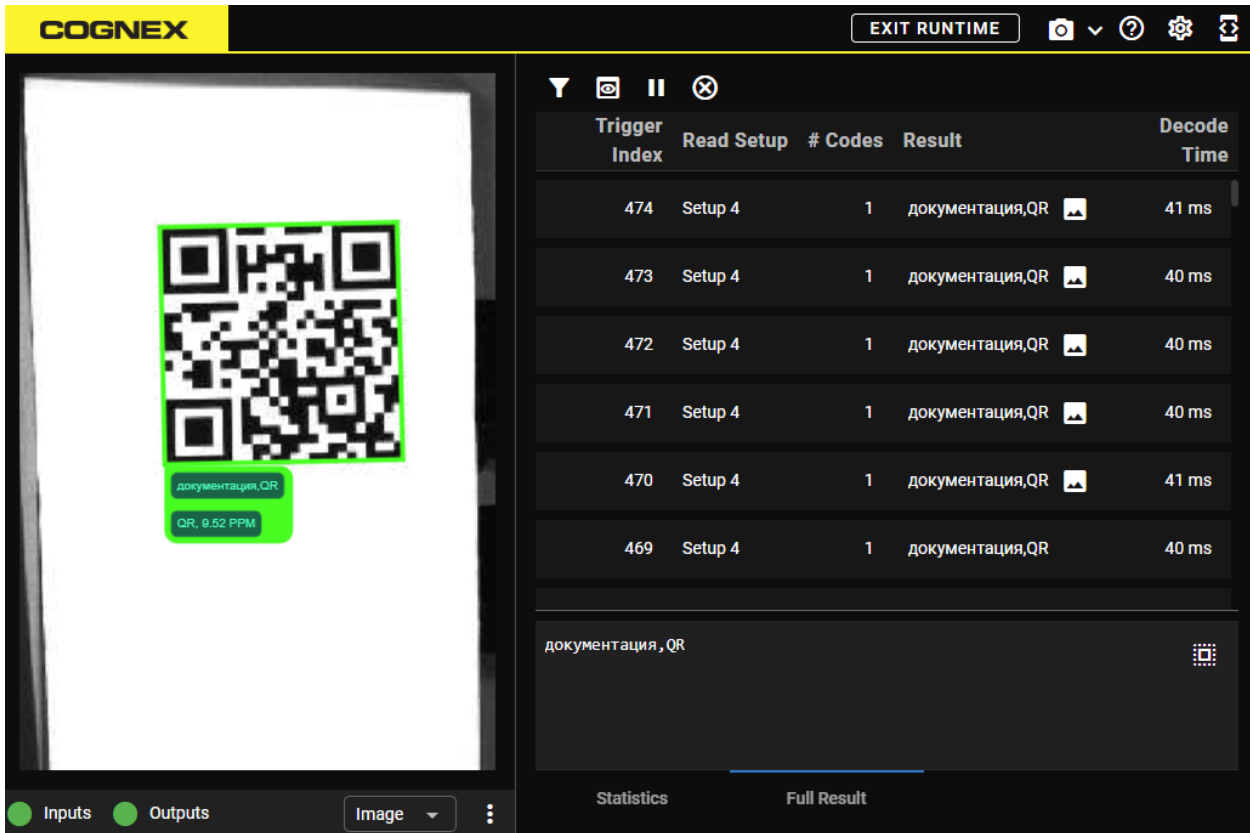
   

Trigger Index	Read Setup	# Codes	Result	Decode Time
474	Setup 4	1	Ð'Ð%Ð°Ñ@Ð¼Ð½Ñ@Ð°Ñ...	41 ms
473	Setup 4	1	Ð'Ð%Ð°Ñ@Ð¼Ð½Ñ@Ð°Ñ...	40 ms
472	Setup 4	1	Ð'Ð%Ð°Ñ@Ð¼Ð½Ñ@Ð°Ñ...	40 ms
471	Setup 4	1	Ð'Ð%Ð°Ñ@Ð¼Ð½Ñ@Ð°Ñ...	40 ms
470	Setup 4	1	Ð'Ð%Ð°Ñ@Ð¼Ð½Ñ@Ð°Ñ...	41 ms
469	Setup 4	1	Ð'Ð%Ð°Ñ@Ð¼Ð½Ñ@Ð°Ñ@,...	40 ms

Ð'Ð%Ð°Ñ@Ð¼Ð½Ñ@Ð°Ñ@,Ñ@,QR 

Statistics Full Result

Inputs Outputs Image 



The screenshot shows the COGNEX WebUI interface. On the left, a camera view displays a QR code with a green bounding box. Below the QR code, a green box contains the text "документация,QR" and "QR, 9.52 PPM". At the bottom left, there are "Inputs" and "Outputs" indicators, and an "Image" dropdown menu. On the right, a table displays the following data:

Trigger Index	Read Setup	# Codes	Result	Decode Time
474	Setup 4	1	документация,QR	41 ms
473	Setup 4	1	документация,QR	40 ms
472	Setup 4	1	документация,QR	40 ms
471	Setup 4	1	документация,QR	40 ms
470	Setup 4	1	документация,QR	41 ms
469	Setup 4	1	документация,QR	40 ms

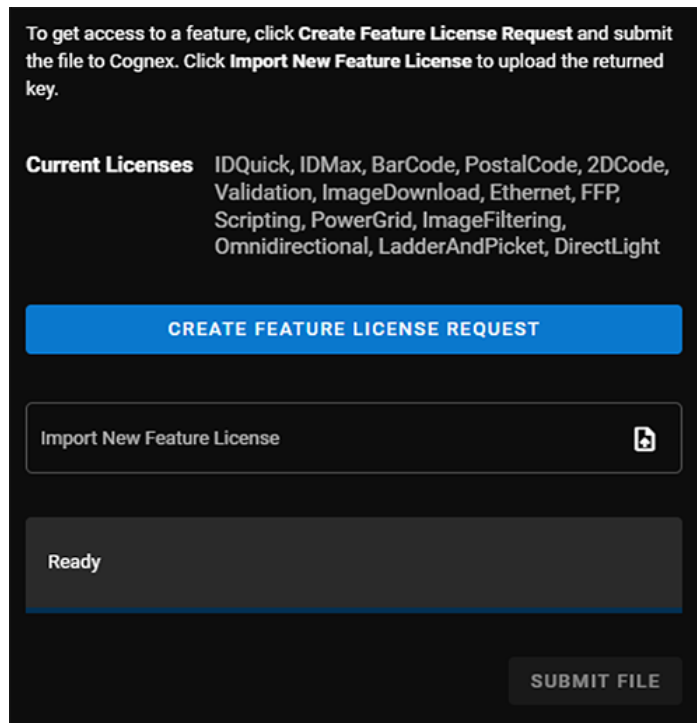
Below the table, there is a "Full Result" section showing "документация, QR" with a small QR code icon. At the bottom right, there are "Statistics" and "Full Result" tabs.

By default, the WebUI recognizes Extended Channel Interpretation (ECI) indicators compliant with AIM ECI standards. When reading compliant codes, this allows the WebUI to automatically identify and use the correct codepage. Check the **Ignore ECI Indicators** box to disable support for ECI indicators, so the WebUI uses the configured codepage instead.

Note: The WebUI has limited support for non-codepage ECI indicators. The WebUI still attempts to display codes with these indicators, but correct interpretations are only possible in the expected downstream program.

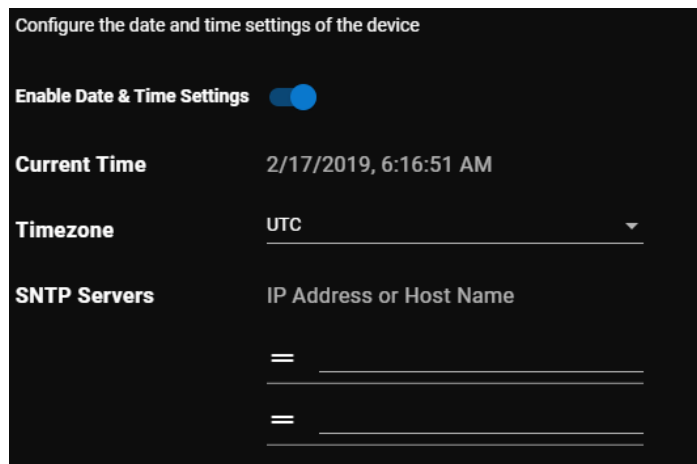
Licenses

In the **Licenses** tab of **Settings**, you can view and manage the Feature Keys of your reader.



Date & Time

In the **Date & Time** tab of **Settings**, you can configure date and time settings for your reader by clicking on **Enable Date & Time Settings**.

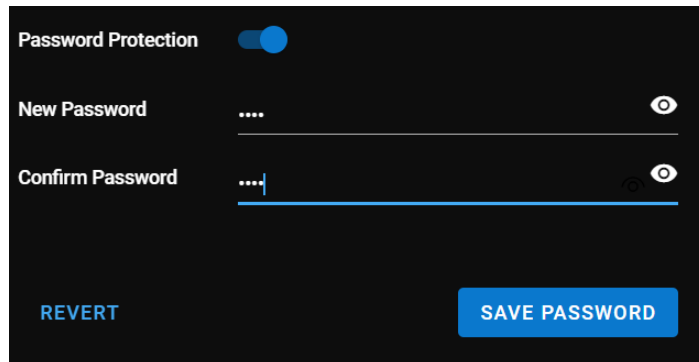


If you choose **SNTP Service**, you can configure clock time sources between your servers. Choose **Timezone** and enter the IP Addresses or Host Names of the servers.

Note: You have to configure an SNTP server to add timestamps to the output string. For more information on the output string, see [Data Formatting on page 24](#).

Password Protection

In the **Password Protection** tab of **Settings**, you can set up a password for the reader. When you enable password protection and lock the WebUI, users cannot modify the reader configuration without a password. Use this feature to protect your settings from unwanted changes.



The screenshot shows the 'Password Protection' settings interface. At the top, the 'Password Protection' toggle is turned on. Below this, there are two password input fields: 'New Password' and 'Confirm Password'. Both fields have masked characters (dots) and a visibility icon (an eye) to the right. At the bottom of the screen, there are two buttons: 'REVERT' on the left and 'SAVE PASSWORD' on the right.

Turning on Password Protection

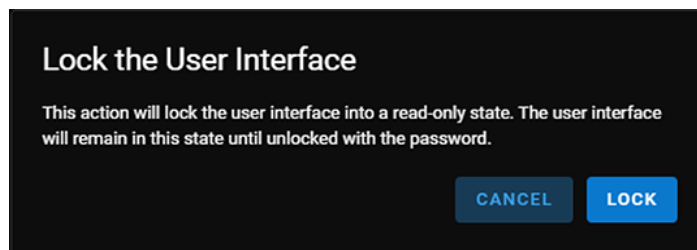
To turn on password protection:

1. Enable the **Password Protection** toggle.
2. Enter the new password in the **New Password** and **Confirm Password** fields.
3. Click **Save Password**.

Application Lock

To lock the application:

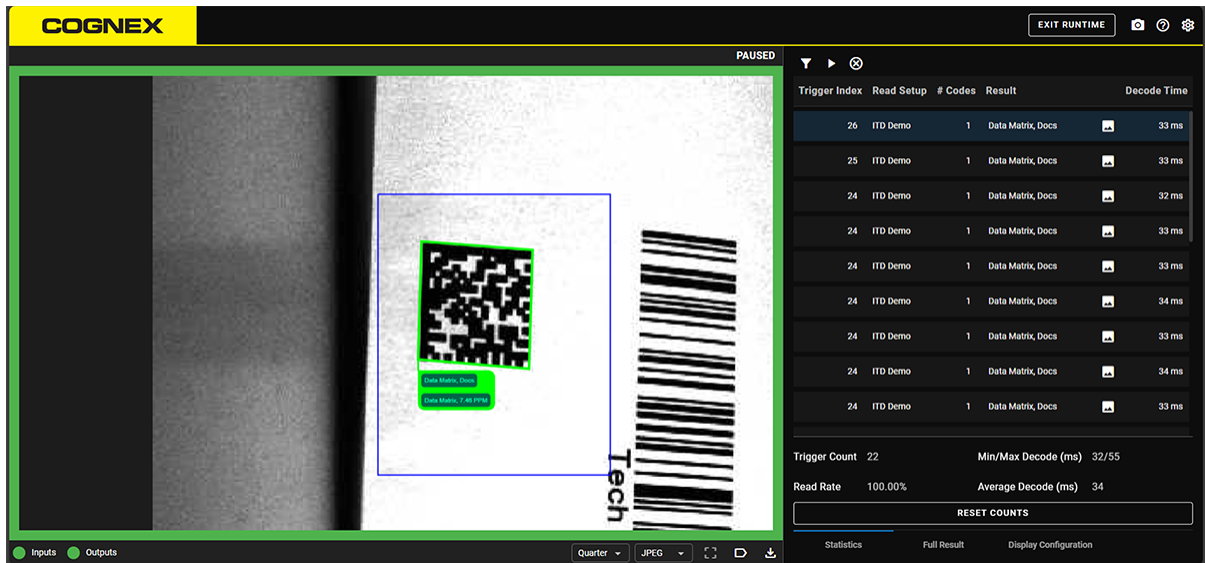
1. Turn on password protection.
2. Click on the lock icon at the top right of the screen. The **Lock the User Interface** dialog appears.



The screenshot shows a dialog box titled 'Lock the User Interface'. The text inside the dialog reads: 'This action will lock the user interface into a read-only state. The user interface will remain in this state until unlocked with the password.' At the bottom of the dialog, there are two buttons: 'CANCEL' on the left and 'LOCK' on the right.

- Click **Lock**.

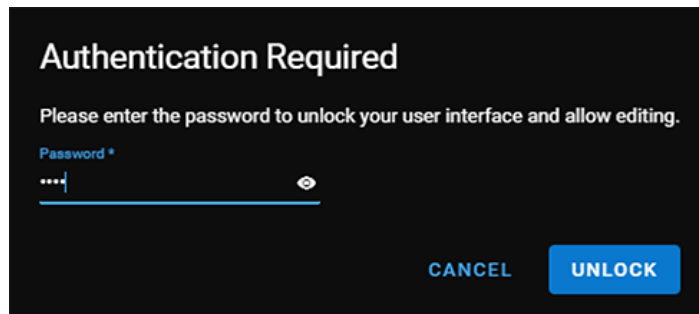
The application is locked, and you can only view the **Runtime** step.



To unlock the application:

- Click **Exit Runtime**.

The **Authentication Required** dialog appears:



- Fill in the **Password** field and click **Unlock**.

Note: The WebUI locks you out after a few minutes of inactivity.

Forgotten Password

Reset your DataMan reader to factory settings. For more information, see [Factory Reset on page 63](#).

Removing the Password

Disable the **Password Protection** toggle, then click **Turn off** on the pop-up.

Program Management

The **Program Management** tab of **Settings** allows you to see information about and perform actions relating to your programs. A blue checkmark appears next to the current active program.

A program is a collection of device-level configuration settings, including one or more read setups, application details, and communication setup.

#	Name	Created	Modified
00	Program 1	2018-02-01 01:00:00 AM	2018-02-01 01:00:00 AM
01	Program 2	2024-11-25 11:26:33 AM	2024-11-25 11:26:33 AM
02	+ PROGRAM		
03	+ PROGRAM		
04	+ PROGRAM		
05	+ PROGRAM		
06	+ PROGRAM		
07	+ PROGRAM		
08	+ PROGRAM		

The following actions are available by clicking on the **three dots** next to each program:

- Set as active
- Rename
- Clone
- Swap slots
- Export
- Delete

When cloning a program, the program name is kept by default, and you can change the name in the cloning window. Multiple programs can have the same name.

You can see the slot number, example images, and creation and modification dates of your previously created programs. The programs are sorted by slot number in ascending order by default, but you can click on any column to sort the programs by that value. The arrow next to **#** (slot number), **Name**, **Created**, or **Modified** signifies what metric the programs are sorted by and whether it is in ascending or descending order.

Add New Program

You can create a new program by clicking **+ Program** and choosing an **Add New Program** option:

Add New Program

Add a new active program using one of the following options

Create a new blank program

Import a program file

Program Name

CANCEL ADD NEW PROGRAM

The following options are available when adding a new program:

- **Create a new blank program:** the new program has all the default settings and no program name. You can change the default name by typing in the **Program Name** field.
- **Import a program file:** choose and import a previously exported program file. The new program copies all settings, including program name and the creation date of the exported program.

Click **Add New Program** to create the program.

Program Management with DMCC

You can also set the active program with the PROGRAM.ENABLE command, using the following syntax:

Name	Type	Range	Meaning
slot	int	[0-19]	program slot

For example, use the following command to update the current program to program slot 1:

```
SET PROGRAM.ENABLE 1
```

Network Settings

In the **Network Settings** tab of **Settings**, you can configure network settings for your reader. To save network settings, click on **Apply**.

Configure network settings for the device

Host Name

DHCP

IP Address

Subnet Mask

Default Gateway

DNS Server

Domain Name

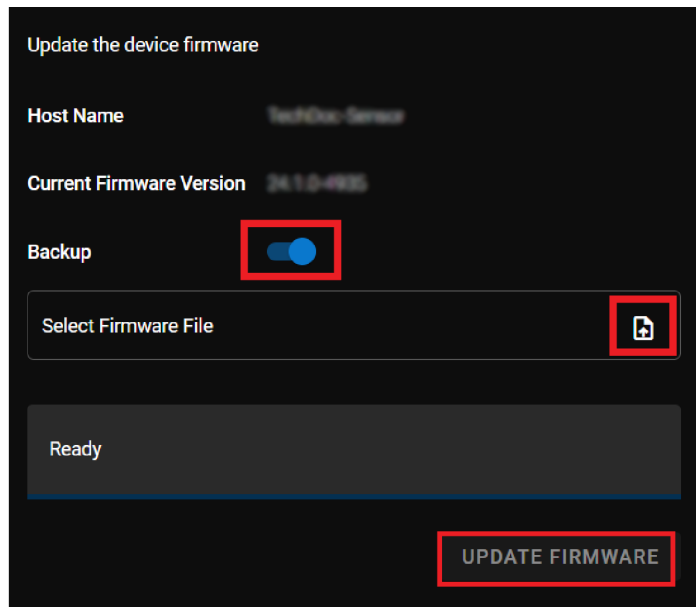
Ready

[REVERT](#) [APPLY NETWORK SETTINGS](#)

Firmware Update

In the **Firmware Update** tab of **Settings**, you can update the firmware of your reader by uploading the firmware file from your PC and then clicking on **Update Firmware**.

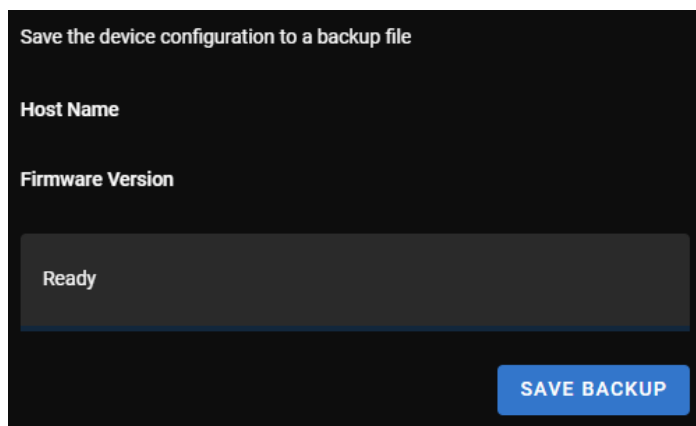
Turn on the **Backup** toggle to save a backup file before the update.



Backup Device

In the **Backup** tab of **Settings**, you can save the reader configuration to a backup file by clicking on **Save Backup**. The backup file includes all programs and global communication settings.

Note: Secure Connection settings are not included.



Restore Device

In the **Restore Device** tab of **Settings**, you can restore the reader from a backup file by selecting your backup file and clicking on **Restore**.

Switching on the **Restore Network Settings** is optional. Switch the toggle on if you want to override the current network settings of the device with the network settings of the backup file.

Note:

- If you enable the **Restore Network Settings**, the [Secure Connection on page 66](#) settings are copied from the backup file to the target reader.
- The Secure Connection files are only restored if the **Restore Network Settings** toggle is enabled.

Restore the device from a backup file

Host Name

Firmware Version

IP Address

Restore Network Settings

Select Backup File

Ready

RESTORE FROM BACKUP



Note: If restoring your reader changes the IP address of the reader, your connection to the reader is lost when the restore process finishes. You can find the new IP address of the reader using device discovery in Station Manager.

Factory Reset

In the **Factory Reset** tab of **Settings**, you can reset the reader to factory settings by clicking **Factory Reset**.

Reset the device to factory defaults

Host Name

Firmware Version

Ready

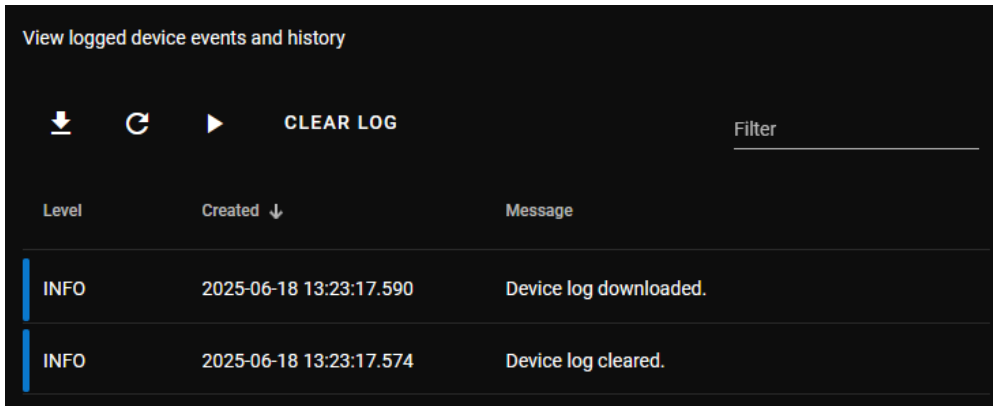
FACTORY RESET



Note: If resetting your reader changes the IP address of the reader, your connection to the reader is lost when the reset process finishes. You can find the new IP address of the reader using device discovery in Station Manager.

Device Log

In the **Device Log** tab of **Settings**, you can access the device log of your reader.



The reader generates logged events with timestamps. The WebUI displays these events in the device log.

The reader fetches the device log automatically when you open the **Device Log** tab. Click **Refresh** to retrieve any updates manually, or the **Play** button to start receiving updates automatically.

The device log contains the following message types:

- **Info**: general information about the reader.
- **Warning**: a non-critical issue.
- **Error**: a problem that needs to be resolved.

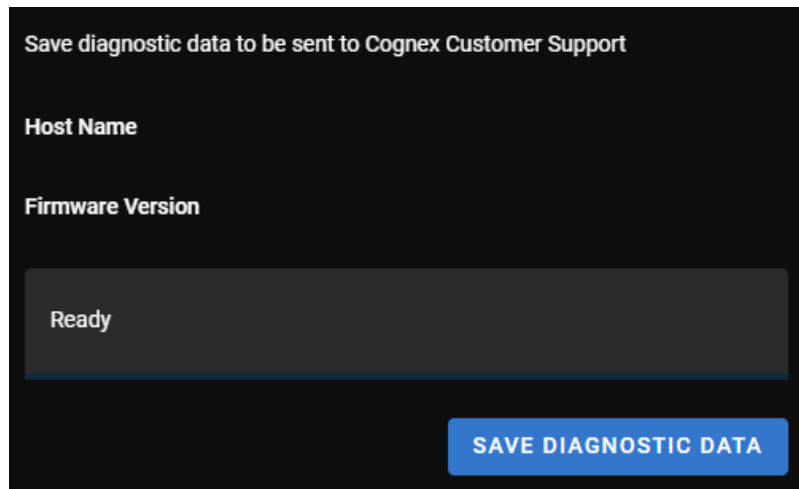
Note: Error events trigger the error LED on the reader. Configure the pulse duration of the error LED on the **Outputs** substep.

The following actions are available in the **Device Log** tab:

- **Download**: download the log in .csv format.
- **Refresh**: get the latest log entries.
- **Clear**: remove all entries from the log.
- **Filter**: input any string to filter the displayed log entries.
- **Sort**: click on a column header to sort the log entries in descending or ascending order.

Field Diagnostics

In the **Field Diagnostics** tab of **Settings**, you can save diagnostics data by clicking on **Save Diagnostic Data**. If necessary, you can send this data to Cognex Customer Support.



Save diagnostic data to be sent to Cognex Customer Support

Host Name

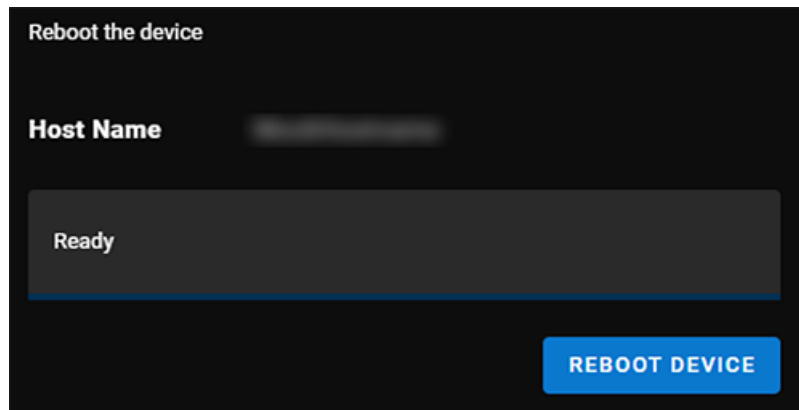
Firmware Version

Ready

SAVE DIAGNOSTIC DATA

Reboot

In the **Reboot** tab of **Settings**, you can reboot the device.



Reboot the device

Host Name

Ready

REBOOT DEVICE

After clicking **Reboot Device**, the screen confirms that the reader initiated the reboot. Rebooting takes a few seconds to finish.

Secure Connection




In the **Secure Connection** of **Settings**, you can set up a secure communication channel between the reader and the browser by uploading and using an SSL certificate.

Configure a secure connection between the device and the browser

Applied SSL Certificate

None Applied

Upload SSL Certificate

Server Certificate	Server Certificate *	
Intermediate Certificate (optional)	Intermediate Certificate	
Certificate Key	Certificate Key *	

Status

UPLOAD CERTIFICATE

The **Applied SSL Certificate** shows you which SSL certificate you are currently using.

To upload an SSL certificate, click on the **Upload** button and select the following files from your computer:

- **Server Certificate**
- **Intermediate Certificate (optional)**
- **Certificate Key**

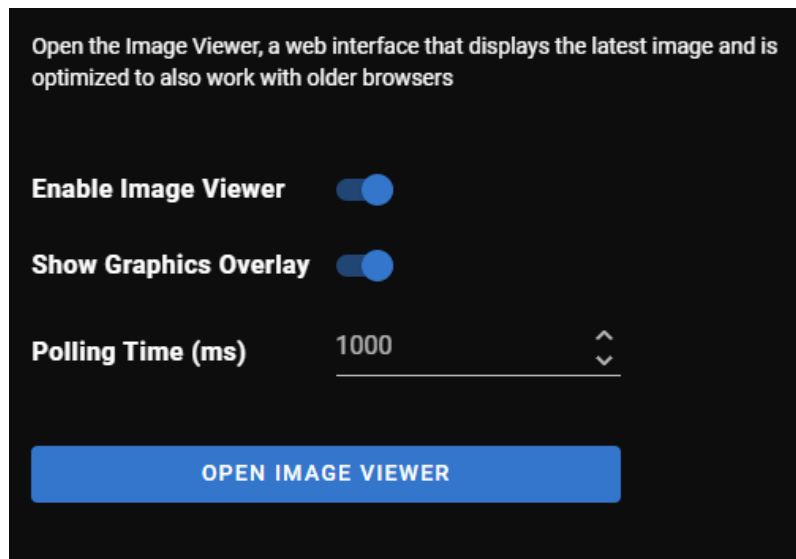
The **Status** shows the statuses of the uploaded files.

Click on **Upload Certificate** to upload the chosen files.

Note: If you delete your certificate, your connection is no longer secure. In this case, you need to manually remove the "s" from the protocol in the URL to reconnect to the reader.

Image Viewer

In the **Image Viewer** tab of **Settings**, you can configure and open the Image Viewer, which is accessible outside of WebUI. The Image Viewer is a HMI that displays the latest image captured by the reader with optional overlay graphics. The Image Viewer is optimized to also work with older browsers.



The **Polling Time (ms)** defines how often the web interface displays a new image. The minimum value is 500 ms. Modern browsers ignore the polling time and display the latest trigger.

