LYNQ

LYNQ GEN 2

Factory Automation Guide

OCTOBER 2024

LYNQ

Contents

Factory Automation Overview

140+ Industrial Drivers
LYNQ supports 140+ industrial drivers out of the box.

OEE
Automation increases the accuracy of overall equipment effectiveness with minimal human interaction.

Realtime Posting
Post machine data instantly to your ERP/PLM system.

With over 140 industrial drivers to the most common PLC/IO devices including Allen Bradley, GE, Honeywell, Mitsubishi, Siemens and more. LYNQ provides the platform to support factory digitalisation by connecting machines to read data automatically into LYNQ. Manual and hybrid data capture methods are also supported using the workbench and timesheet features.

Automated data collection allows manufacturers to:

- Improve visibility across manufacturing operations
- Improve effectiveness
- Reduce cost of production
- Reduce time to market
- Increase factory performance
- Gain competitive advantage

LYNQ's factory automation solution provides controlled management of machine data including error handling; live or controlled data posting to ERP/PLM systems; data collection from multiple geographical facilities; data historian and supports these features and capabilities:

- Active job or multiple jobs
- Good parts count including serial numbers, weight etc
- Scrap parts count including scrap reasons codes etc
- Machine status including downtime, breakdown etc
- Job completion status
- Real-time OEE/OLE/TEEP visibility
- Alerting and production issues assignment
- Real-time ERP WIP integration
- Intelligent error handing
- Generation of time-series data for conditional monitoring

It is important to note that this feature guide does not cover the setup and maintenance requirements for the PLC or IO device and/or the OPC server. Configuration of these hardware and software devices will be completed by a recommended industrial engineer.

Within this feature guide, you will learn:

- System requirements to support factory automation
- How to enable factory automation
- How to create automated devices
- How to configure signals and processors
- How to configure automation examples
- How to use the factory automation simulator
- How to monitor a live automated environment
- How to resolve dataflow errors
- How to setup alerts for monitoring automation errors
- How to troubleshoot failures

System Requirements

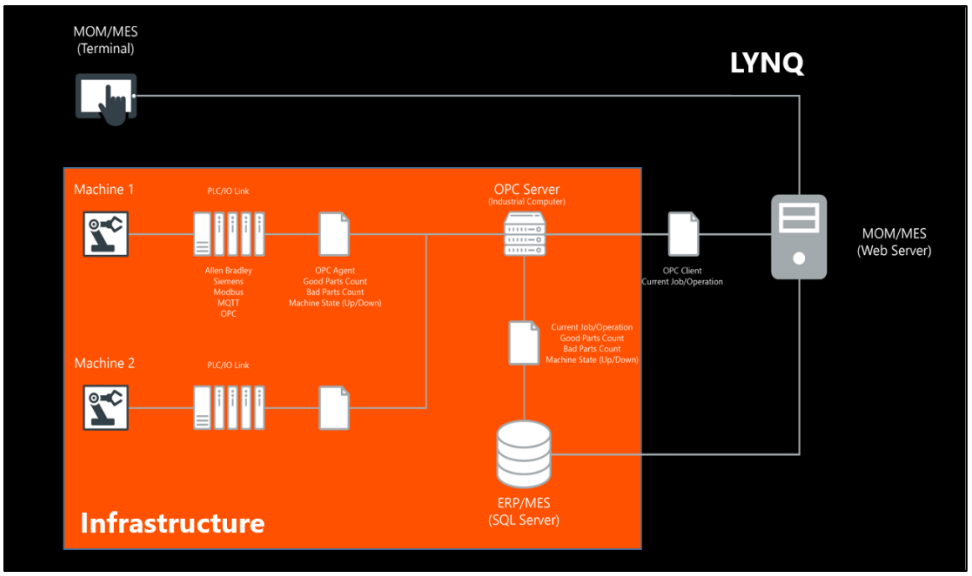
The Solution
Factory Automation combines hardware and software to create a unique solution.

System Requirements
Read the System Requirements for OPC Server specifications.

Device Licenses
Automated device licenses can be purchased separately.

Factory automation is entirely dependent on the infrastructure illustrated in the Factory Automation Topology Diagram. You must have LYNQ implemented to utilise factory automation.

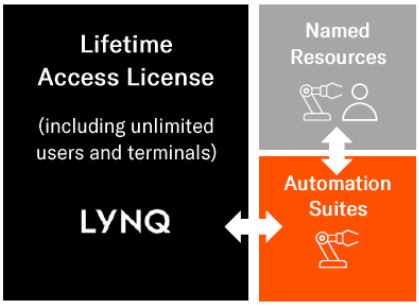
Factory Automation Topology Diagram

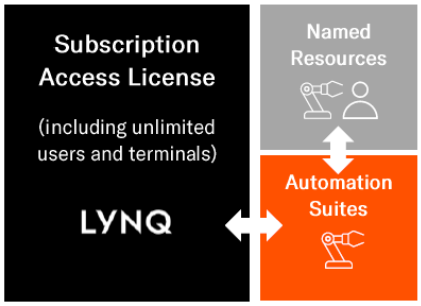


Before configuring automated devices in LYNQ, the following pre-requisites must be completed outside of LYNQ.

- Factory floor digitalised to a level where machine data is available on PLC/IO (provided and supported by end user)
- LYNQ machine connectivity middleware (OPC Server) is installed, connected to machine, and configured to log machine data in format of tag-value (provided and supported by LYNQ)

To automate equipment, an automation suite is required which differs to the named resource license. Suites provide access to collections of industrial drivers, which make it possible for LYNQ to connect to the most common PLC/IO devices used in the manufacturing industry.





To implement factory automation, refer to the following documents

- LYNQ System Requirements
- Licensing Guide

The Concept of Automation

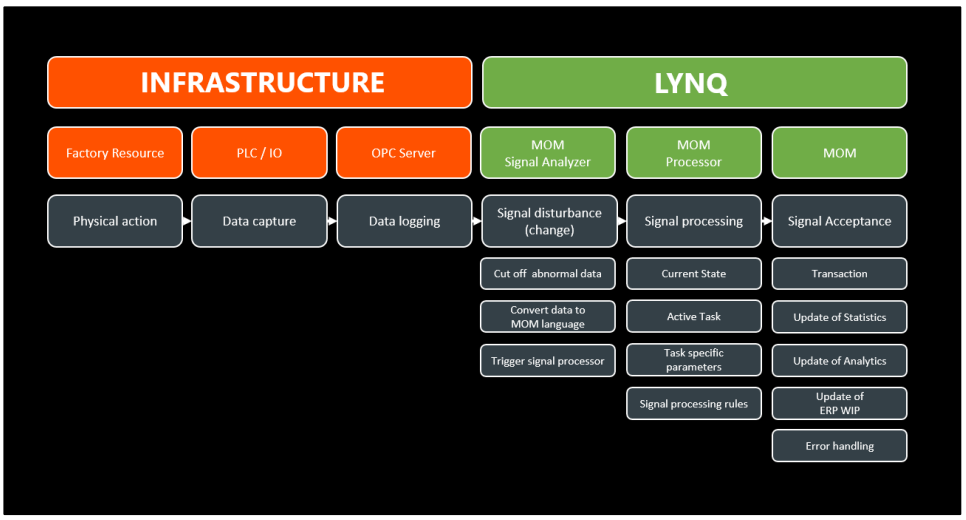
Data Logging
LYNQ combines the data logged by the OPC server to active jobs running in LYNQ.

Real-Time Analysis
When the Factory Automation feature is correctly configured and all components of the infrastructure are working seamlessly you will be able to see the data from your automated devices appear in LYNQ. Quantity, Scrap and Status related Data coming into LYNQ will be updated on the Workbench and on all other live status screens in real-time.

Transaction Review
All posted Factory Automation Transactions will appear in the Transaction Review screen and will be set to the approval status based on the default approval status specified in Transaction Rules (Advanced Settings). These transactions will still need to go through your standard approval process to be posted to your ERP application.

Factory automation in LYNQ has a simple concept. The Data Flow Diagram illustrates the overall factory automation solution. The orange columns (Infrastructure) illustrate the flow of data between the various hardware and software components before reaching LYNQ. The green columns (LYNQ) illustrate the flow of data once the data has reached LYNQ.

Data Flow Diagram



It is important that all these aspects of the factory automation solution are understood and configured correctly. To successfully adopt automation in your factory, you will require a resource, such as an Industrial Engineer that is sufficiently skilled to maintain your factory network including PLC/IO devices and the OPC Server hardware and software. Note: LYNQ can only take responsibility for the support and maintenance of its own software.

Whilst this guide demonstrates how connected machines can be automated, automation may also be used for employee resources.

For example:

- To automatically start/end an employee's day with RFID
- To automatically report quantity for an employee with RFID
- To automatically report quantity when an employee drops production into a box where the box is equipped with Sensor and IO Device

Enabling Factory Automation

Intervals
Interval settings can be adjusted as required to suit the requirements of your factory. However, setting these values to very low numbers may place additional load on your infrastructure in high data volume environments. Fine tune these settings for optimal performance.

Polling Interval
The polling interval determines how frequently data will be read in the factory automation database to LYNQ.

Data Buffer Interval
The data buffer interval determines how frequently the processor will translate machine data into LYNQ transactional data.

Transaction Interval
The transaction generation interval determines how frequently LYNQ will generate transactions that can be approved and posted to the ERP system. Transactions will appear in the Transaction Review Screen. Note, this setting also applies to events performed in the Workbench and when using the REST API feature.

By default, factory automation is disabled. The following steps explain how to turn on automation within LYNQ.

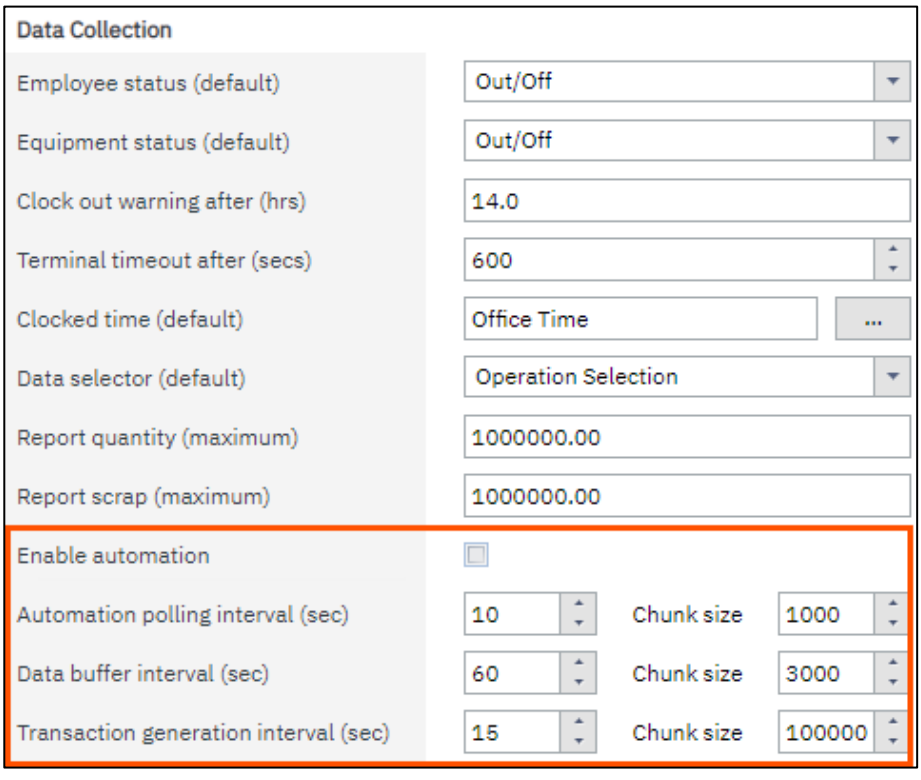
LYNQ must understand which database is configured to store data logged by the OPC Server.

Configure the Factory Automation Database:

1. Logon to LYNQ as an administrator
2. Select Settings from the home page
3. Select Settings
4. Navigate to the Database Settings section
5. Enter the factory automation database name in the Automation row
6. Select Save

Enable Factory Automation:

1. Logon to LYNQ as an administrator
2. Select Settings from the home page
3. Select Advanced Settings
4. Select General
5. Check the Enable automation option
6. Set the Automation polling interval (secs) as required
7. Set the Data buffer interval (secs) as required
8. Set the Transaction generation interval (sec) as required
9. The chunk size determines the maximum number of records that can be processed at one time. Set these values as required



Configuring Automated Resources

Automation Settings
Except for the automation check box setting, all other fields in this section are for informational purposes only.

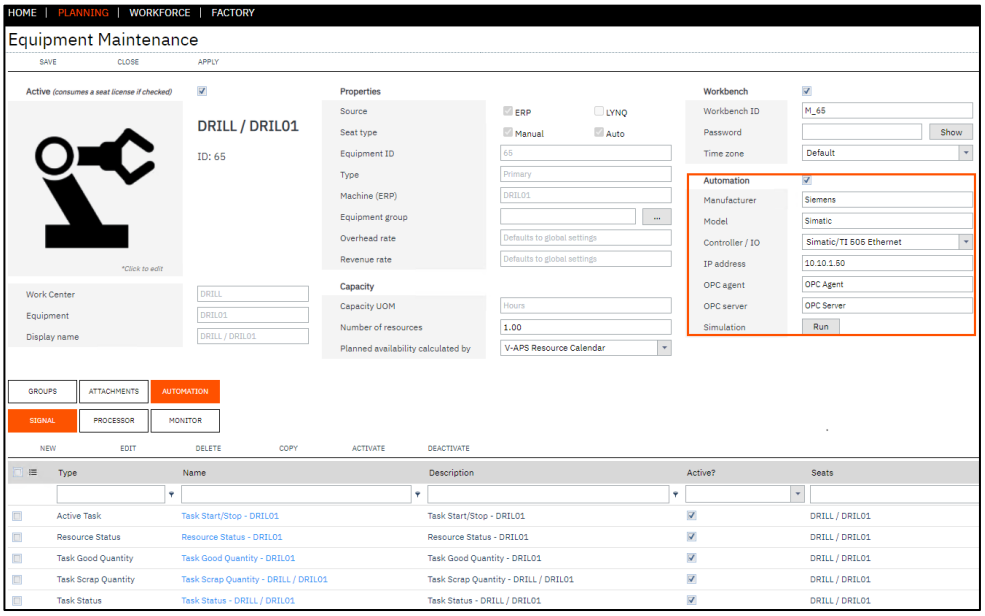
Seat Type
Once the automation checkbox is ticked, the seat type will change to automated device.

Manual Data Collection
It is possible to turn off manual data collection for an automated device by deselecting the Workbench Checkbox in Equipment Maintenance.

To enable equipment or employees for automation:

1. Select Seat Maintenance
2. Select the Equipment/Employee record and click Edit
3. Check the Automation tick box
4. For informational purposes only, enter the manufacturer's details, IP Address of the automated device, the OPC Agent and the IP Address of the OPC Server.
5. Click Save

Equipment Maintenance



Against each Automated Device within LYNQ, you must configure both:

1. Signals
2. Processors

It is recommended that you read and understand the sections on understanding signals and understanding processors before trying to configure automation in LYNQ.

Understanding Signals

Activate/Deactive
Signals can be activated or deactivated as required by double clicking on the signal.

Copying Signals
Signals can be copied to simplify the setup process.

Signals are used to read any changes in tag values, logged in the OPC Server. You must create a signal in LYNQ to match each tag configured in the OPC server.

LYNQ supports five (5) different signals types:

Signal	Purpose
Resource Status	The resource status signal can be used to read the status of a resource. (i.e. the resource is on, off, or any other status value).
Task Start/Stop	The task start/stop signal can be used to start or stop a task running on a resource.
Task Good Quantity	The task good quantity signal can be used to update the good quantity produced for a job/operation.
Task Scrap Quantity	The task scrap quantity signal can be used to update the scrap quantity produced for a job/operation.
Task Status	The task status signal can be used to update the status of a job/operation. (i.e update the operation to complete or update the operation to open)

Whilst the primary purpose of a signal is to prepare the tag data coming from an automated resource for a particular type of event in LYNQ, the signal does support some basic transformation options.

Including the transformation of:

- Incoming automated data to a different value using the Mapping option
- Empty tag values to a different value using the Settings option

Further automation rules and transformation settings are configured against the corresponding processor.

Signals read incoming machine data from the Lynq_ME_TF_InputTagData table. The table resides in the Factory Automation database.

Time series data such as pressure, temperature, spindle speed, etc may be recorded and stored within the LYNQ factory automation SQL database. LYNQ signal processing will ignore this data but the data can be used for custom reporting (i.e Power BI, Microsoft SQL Server Reporting Services). You can then provide visibility of this data in LYNQ via the Webhook functionality.

Alternatively, a dedicated LYNQ time series factory automation database may be used. It is recommended to keep time series data in a separate database when significant volumes of data is being capture, for performance and maintenance purposes.

Resource Status Signal

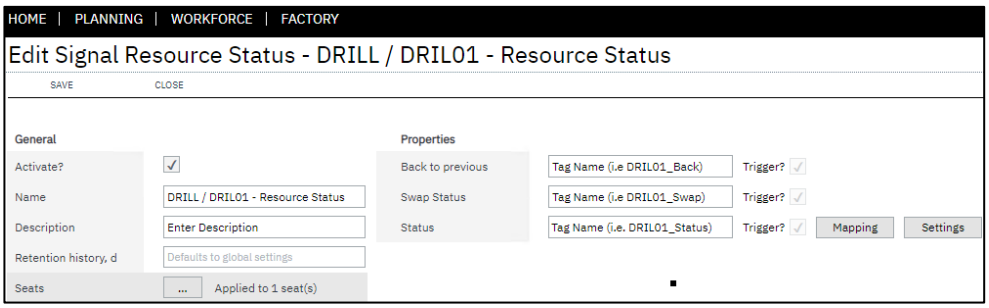
Activate/Deactive
Signals can be activated or deactivated as required by double clicking on the signal.

Copying Signals
Signals can be copied to simplify the setup process.

The resource status signal can be used to read tag information related to the status of a resource. (i.e. resource is on, off, or any other type of status).

By default, the resource status signal supports tags to automate:

- Status
- Swapping of the status
- Changing the status to back to previous



A single signal can be associated to more than one seat.

Task Start/Stop Signal

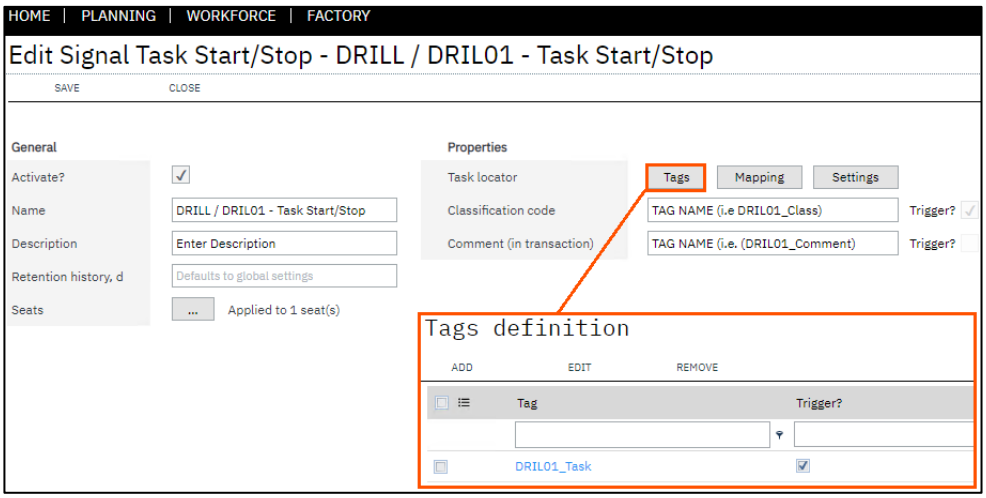
Activate/Deactive
Signals can be activated or deactivated as required by double clicking on the signal.

Copying Signals
Signals can be copied to simplify the setup process.

The task start/stop signal can be used to start or stop a task running on a resource.

By default, the task start/stop signal supports tags to automate:

- Starting of single or multiple tasks
- Association to a classification code
- Creation of a transaction comment for the task



A single signal can be associated to more than one seat and a signal can be defined to start/stop multiple tasks. Supporting environments where machines may be performing more than one job at a same time.

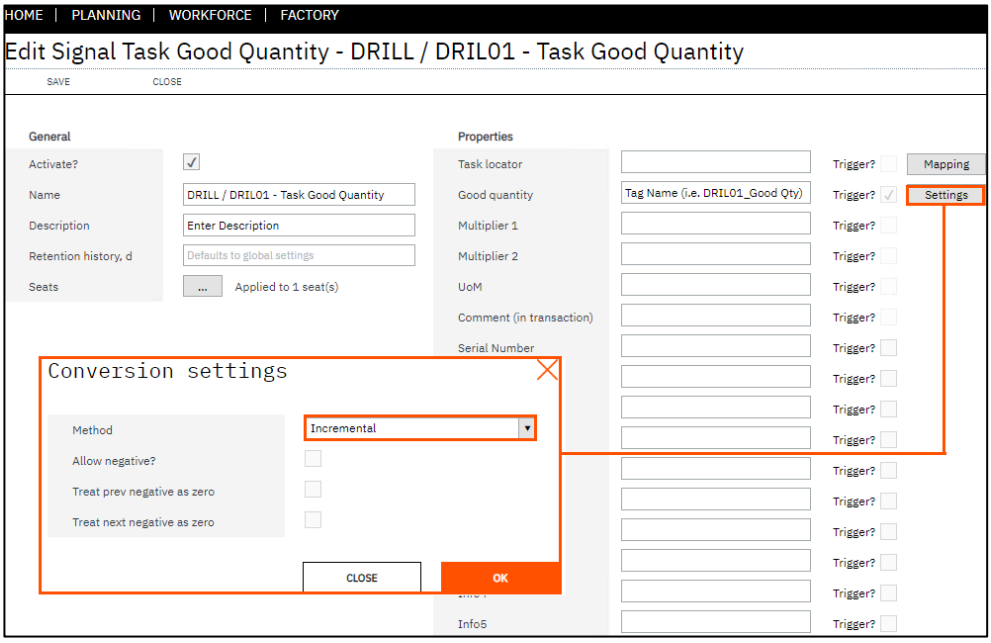
Task Good Quantity Signal

Activate/Deactive
Signals can be activated or deactivated as required by double clicking on the signal.

Copying Signals
Signals can be copied to simplify the setup process.

The task good quantity signal can be used to update the good quantity produced for a job/operation. By default, the task good quantity signal supports tags to automate:

- Good quantity reported
- Report extended reporting details such as:
  - Serial, Bin and Lot Numbers
  - Location/Warehouse
  - UoM
  - Multiplier quantities
  - Relevant transaction comments



Conversion Settings:

Method	Purpose
Incremental	Use when quantity is incrementally increasing. The transaction value is calculated as the difference between current value and the previous value.
Simple Change	Use when quantity should be increased to +1 every time the tag value is changed
Zero to Value	Use when quantity is not incremental. The transaction value is calculated as the difference between current value and zero. (2, 8 and 16 would result in a quantity increase of 26)

A single signal can be associated to more than one seat.

In the event that the machine performs multiple jobs at the same time, multiple good quantity signals must be created, each with a definition of task locator and quantity tag.

Task Scrap Quantity Signal

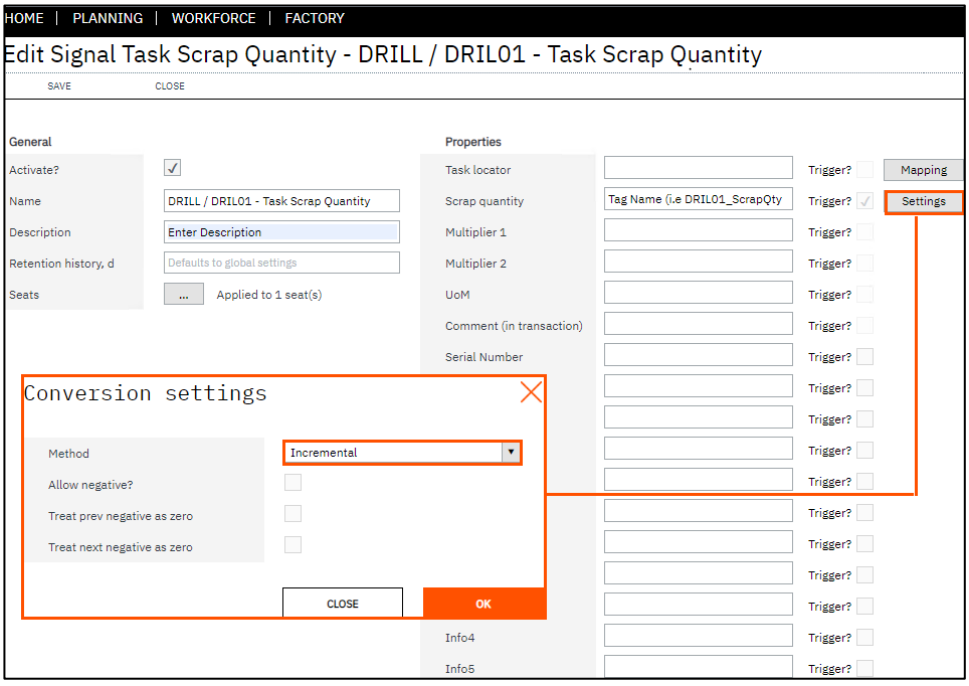
Activate/Deactive
Signals can be activated or deactivated as required by double clicking on the signal.

Copying Signals
Signals can be copied to simplify the setup process.

Scrap Reasons
In order to capture scrap for 5 different reasons codes, the OPC Server must be configured to store 5 separate tags and distinct counters. LYNQ must be setup with 5 separate Scrap Signals each with a unique Scrap Reason code.

The task scrap quantity signal can be used to update the scrap quantity produced for a job/operation. By default, the task scrap quantity signal supports tags to automate:

- Scrap quantity reported
- Report extended reporting details such as
  - Serial, Bin and Lot Numbers
  - Location/Warehouse
  - UoM
  - Multiplier quantities
  - Relevant transaction comments



Conversion Settings:

Method	Purpose
Incremental	Use when quantity is incrementally increasing. The transaction value is calculated as the difference between current value and the previous value.
Simple Change	Use when quantity should be increased to +1 every time the tag value is changed
Zero to Value	Use when quantity is not incremental. The transaction value is calculated as the difference between current value and zero. (2, 8 and 16 would result in a quantity increase of 26)

A single signal can be associated to more than one seat.

In the event that the machine performs multiple jobs at the same time, multiple scrap quantity signals must be created, each with a definition of task locator and quantity tag.

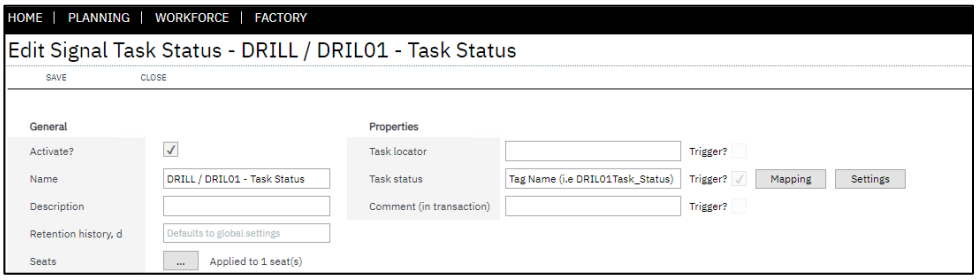
Task Status Signal

Activate/Deactive
Signals can be activated or deactivated as required by double clicking on the signal.

Copying Signals
Signals can be copied to simplify the setup process.

The task status signal can be used to update the status of a job/operation. (i.e update the operation to complete or update the operation to open). By default, the task status signal supports tags to automate:

- Task status
- Creation of a transaction comment for the task



A single signal can be associated to more than one seat.

Creating Signals

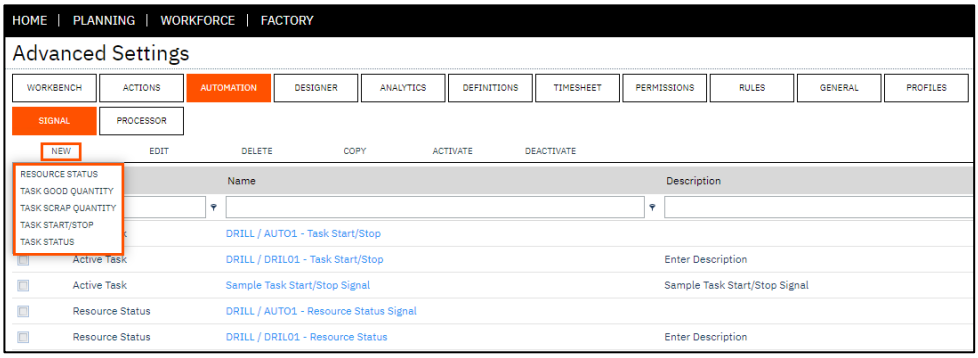
Activate/Deactive
Signals can be activated or deactivated as required by double clicking on the signal.

Copying Signals
Signals can be copied to simplify the setup process.

Signals can be created and configured in advanced settings or directly from the equipment maintenance screen.

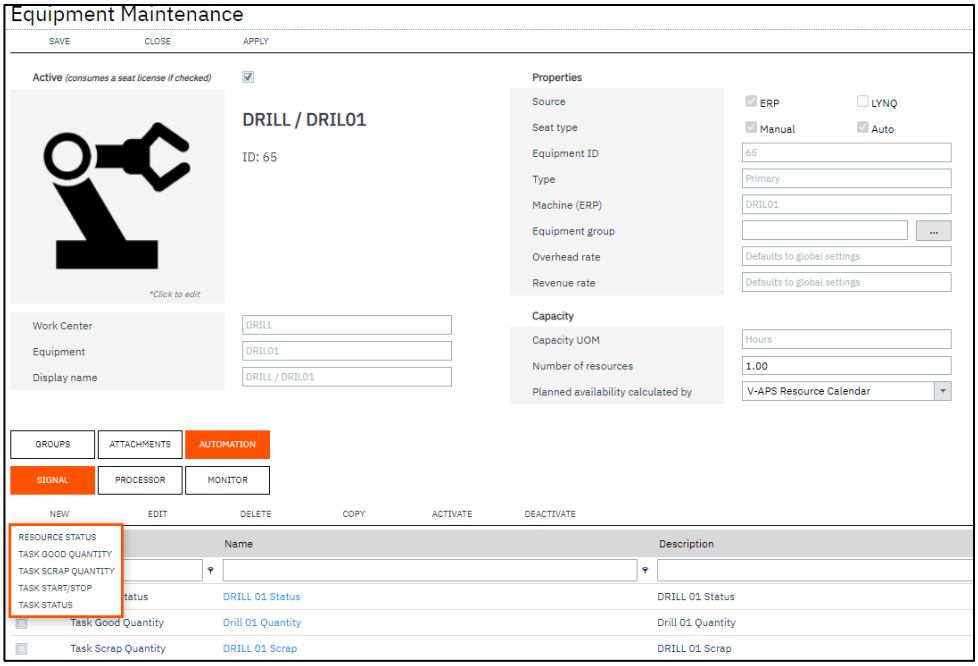
To create signals from advanced settings:

1. From the LYNQ mom home page select Settings
2. Select Advanced Settings
3. Select Automation
4. Select Signals



To create signals from equipment maintenance:

1. Select Seat Maintenance from the Resource Management menu area
2. Select the Equipment record and click Edit
3. Select the Automation Tab
4. Select Signals



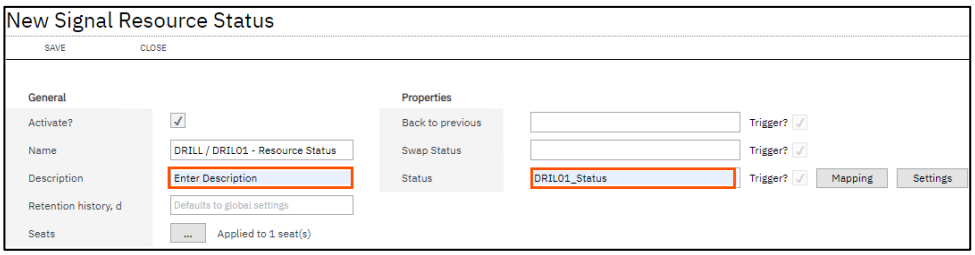
This guide provides step by step instructions on how to create signals from the equipment record.

Resource Status Signal

OPC Tag
The OPC Tag is written to the factory automation logging database along with all other values captured by the PLC/IO device. The data is stored in the Lynq_ME_FA_InputTagData table.

To create a signal for Resource Status:

1. Select Seat Maintenance from the Resource Management menu area
2. Select the Equipment record and click Edit
3. Select the Automation Tab
4. Select Signals
5. Select New
6. Select Resource Status
7. Enter a Description
8. Enter the Tag Name in the Status field (i.e. DRIL01_Status)
9. Select Save
10. Select Apply



The mapping option allows for basic transformation of an incoming value to a new value before the value is read by the processor. In this example, mapping is not configured as further mapping options exist against the processor. The settings option can be used to specify a default value if the incoming tag value is left empty. (see processor section)

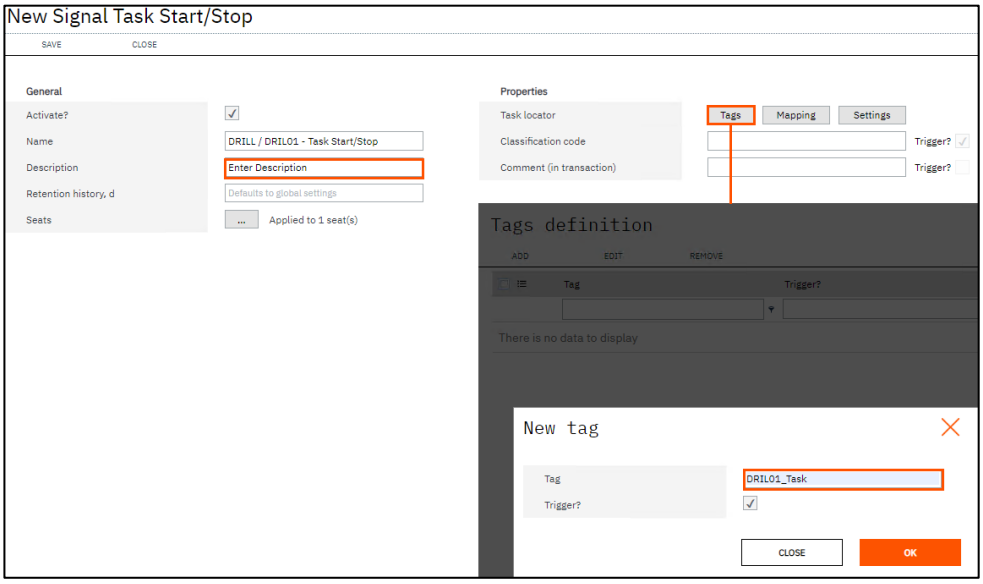
When creating signals from the equipment maintenance screen, the signal is automatically associated to the seat record, You can see which seats the signal is associated to, by clicking on the seat lookup [...] button. Signals are activated by default and observe the default retention history setting in advanced settings, general. The number of days for retention history, can be set differently for each signal.

Task Start/Stop Signal

OPC Tag
The OPC Tag is written to the factory automation logging database along with all other values captured by the PLC/IO device. The data is stored in the Lynq_ME_TF_InputTagData table.

To create a signal for Task Start/Stop:

1. Select Seat Maintenance from the Resource Management menu area
2. Select the Equipment record and click Edit
3. Select the Automation Tab
4. Select Signals
5. Select New
6. Select Task Start/Stop
7. Enter a Description
8. Select Tags
9. Select Add
10. Enter the Tag Name in the Tag field (i.e DRIL01_Task)
11. Select OK
12. Select OK
13. Select Save
14. Select Apply



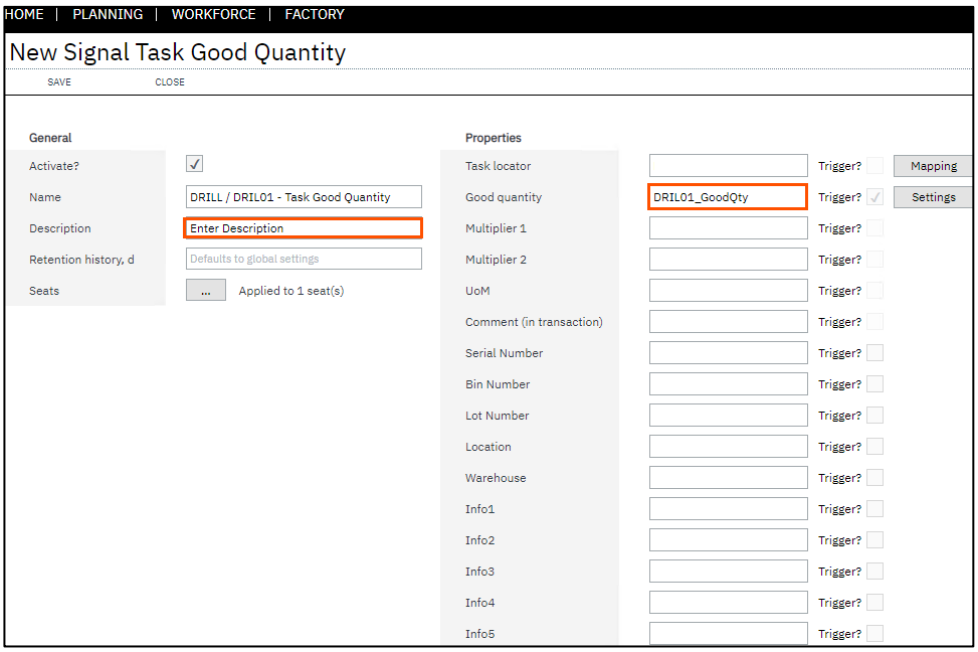
The mapping option allows for basic transformation of an incoming value to a new value before the value is read by the processor. In this example, mapping is not configured as further mapping options exist against the processor.

Task Good Quantity Signal

OPC Tag
The OPC Tag is written to the factory automation logging database along with all other values captured by the PLC/IO device. The data is stored in the Lynq_ME_TF_InputTagData table.

To create a signal for Task Good Quantity:

1. Select Seat Maintenance from the Resource Management menu area
2. Select the Equipment record and click Edit
3. Select the Automation Tab
4. Select Signals
5. Select New
6. Select Task Good Quantity
7. Enter a Description
8. Enter the Tag Name in the Good Quantity field (i.e. DRIL01_GoodQty)
9. Select Save
10. Select Apply



The mapping option allows for basic transformation of an incoming value to a new value before the value is read by the processor. In this example, mapping is not configured as further mapping options exist against the processor. The settings option can be used to specify a default value if the incoming tag value is left empty. (see processor section)

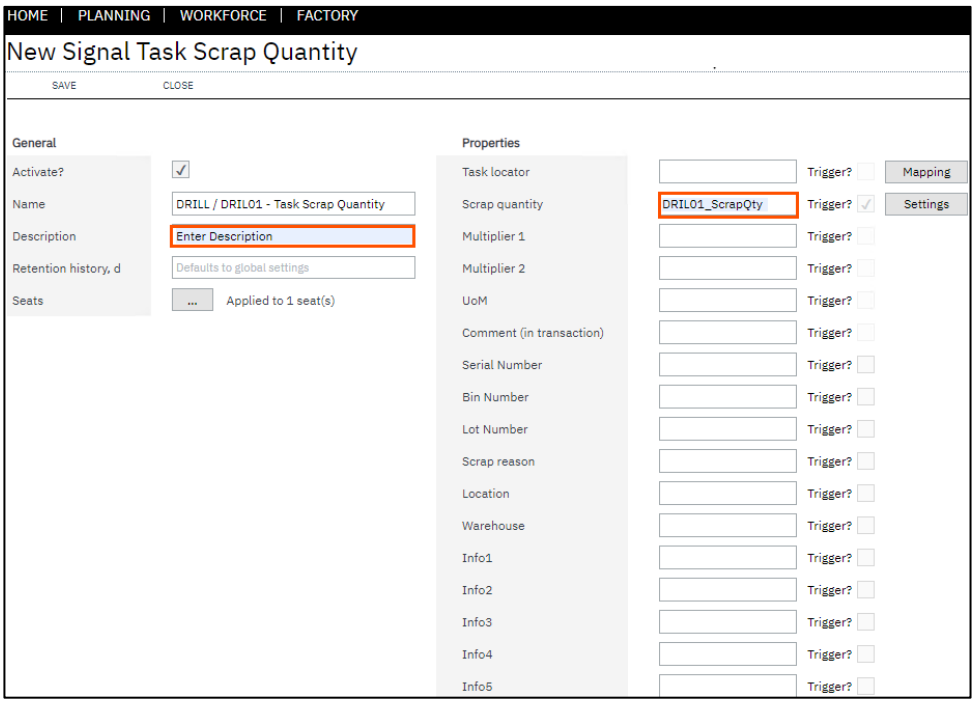
When creating signals from the equipment maintenance screen, the signal is automatically associated to the seat record, You can see which seats the signal is associated to, by clicking on the seat lookup [...] button. Signals are activated by default and observe the default retention history setting in advanced settings, general. The number of days for retention history, can be set differently for each signal.

Task Scrap Quantity Signal

OPC Tag
The OPC Tag is written to the factory automation logging database along with all other values captured by the PLC/IO device. The data is stored in the Lynq_ME_TF_InputTagData table.

To create a signal for Task Scrap Quantity:

1. Select Seat Maintenance from the Resource Management menu area
2. Select the Equipment record and click Edit
3. Select the Automation Tab
4. Select Signals
5. Select New
6. Select Task Scrap Quantity
7. Enter a Description
8. Enter the Tag Name in the Scrap Quantity field (i.e. DRIL01_ScrapQty)
9. Select Save
10. Select Apply



The mapping option allows for basic transformation of an incoming value to a new value before the value is read by the processor. In this example, mapping is not configured as further mapping options exist against the processor. The settings option can be used to specify a default value if the incoming tag value is left empty. (see processor section)

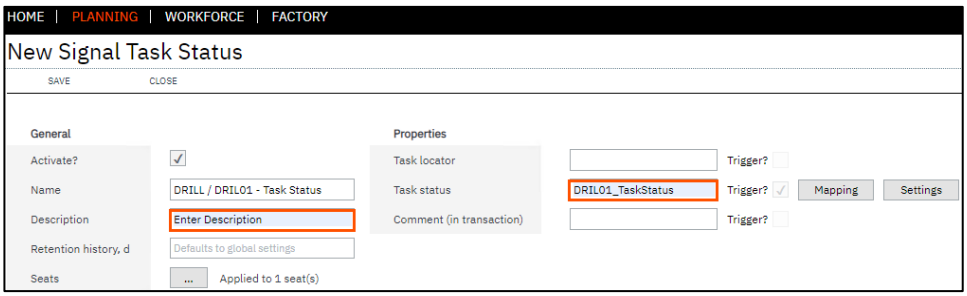
When creating signals from the equipment maintenance screen, the signal is automatically associated to the seat record, You can see which seats the signal is associated to, by clicking on the seat lookup [...] button. Signals are activated by default and observe the default retention history setting in advanced settings, general. The number of days for retention history, can be set differently for each signal.

Task Status Signal

OPC Tag
The OPC Tag is written to the factory automation logging database along with all other values captured by the PLC/IO device. The data is stored in the Lynq_ME_TF_InputTagData table.

To create a signal for Task Status:

1. Select Seat Maintenance from the Resource Management menu area
2. Select the Equipment record and click Edit
3. Select the Automation Tab
4. Select Signals
5. Select New
6. Select Task Status
7. Enter a Description
8. Enter the Tag Name in the Task Status field (i.e. DRIL01_TaskStatus)
9. Select Save
10. Select Apply



LYNQ

Understanding Processors

New Processors
Use the New option to create your own custom processor.

Processors are used to transform signal data into transactional data that LYNQ understands. Within the processor settings, you may also apply specific business rules to enable certain actions to be performed. Similar to those types of actions a user can experience in the workbench.

LYNQ is shipped with pre-defined global processors for:

- Resource Status
- Active Task
- Task Good Quantity
- Task Scrap Quantity
- Task Status

Against each processor property you can define where the property value is read from.

The methods for reading property values:

Method	Purpose
Incoming	The data will be read from the incoming signal
Predefined	The data is predefined within the processor
By Seat	The data is read from the associated seat record
Mapping	The data is read from the mapping table
Scheme	The data is read from the scheme definition

Signal data consists of only 4 attributes:

- Tag Name
- Tag Value
- Moment (Timestamp)
- Quality

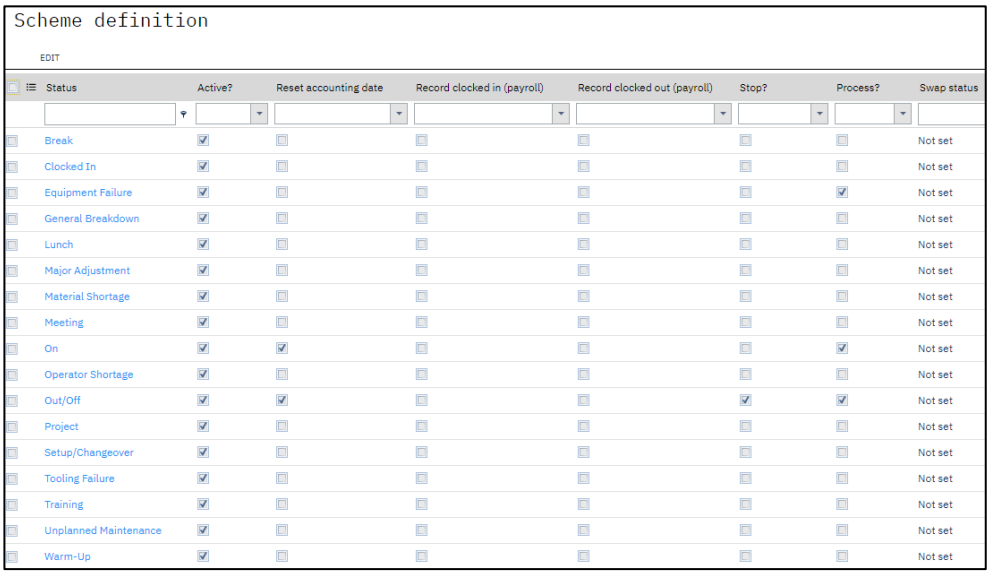
Signal data in this format may not be sufficient for automation purposes in LYNQ. Let's assume that you have a PLC for a digitalised machine that is only capable of recognising parts count. The PLC has no understanding of the job it was performing at the time. You want the parts count to be captured in LYNQ against Job 100. By using processor settings, it's easy to overcome this limitation. There are many other examples where signal data may be insufficient.

Scheme Definition

Resource Status

The resource status processor includes a scheme definition feature.

Scheme Definition Properties:



Refer to the table below to understand the purpose of each property.

Property	Purpose
Active	The process is active for the type of status
Rest Accounting Date	Whether the accounting date should be reset after the transaction is created
Record Clocked in (payroll)	Whether the clocked in date should be reset after the transaction is created
Record Clocked out (payroll)	Whether the clocked out date should be reset after the transaction is created
Stop	Whether all active tasks should be stopped after the transaction is created
Process	Whether additional data should be created after the transaction is created. This behaviour works the same as the process data function within on screen elements.
Swap Status	Whether the status of the resource should be swapped after the transaction is created

`Task Locator

Task Locator
Task Locator reads incoming data from a signal and can perform data transformation services. In addition, the Task Locator can be used to change the key that Factory Automation will read for incoming task information and can be configured to limit the selection or records and use data filters.

LYNQ includes a task locator feature which can be used in the processors:

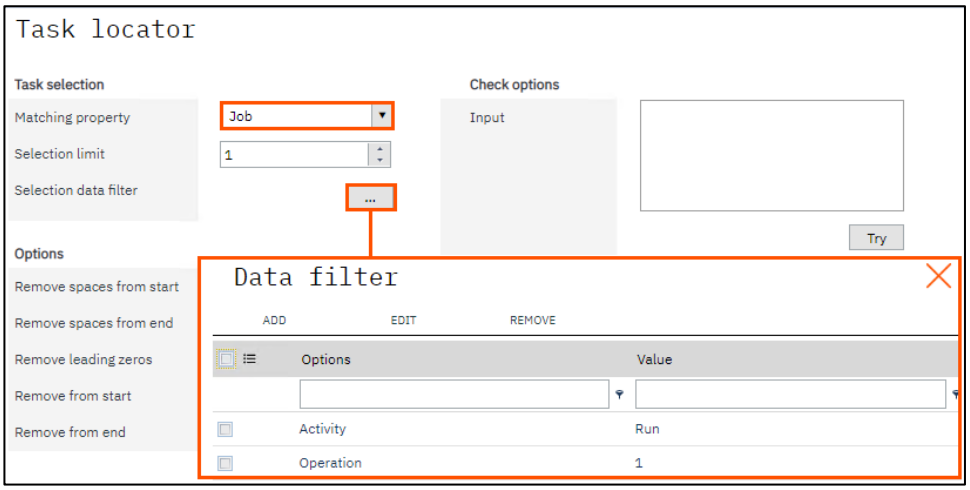
- Task Start/Stop
- Task Good Quantity
- Task Scrap Quantity
- Task Status

The task locator feature may be used to locate which tasks the processor should be applied to. For example, assume the PLC controller only stores the job number (123) in the tag value. The job number (123) alone, is not sufficient information to start a task, as LYNQ requires 3 attributes to start a task, by default:

- Job Number (i.e. 123)
- Operation Number (i.e 1)
- Activity (i.e Run)

To accommodate this scenario, the task locator would be configured as:

- Matching property equals job
- Data filters to return the operation and task



Task Selection Property Values:

Property	Code	Option	Value
Matching Property	Job	Select the 2nd job code from the list	N/A
Selection Data Filter	Activity	Select the first activity code from the list	Run
	Operation	N/A	1

Task Locator

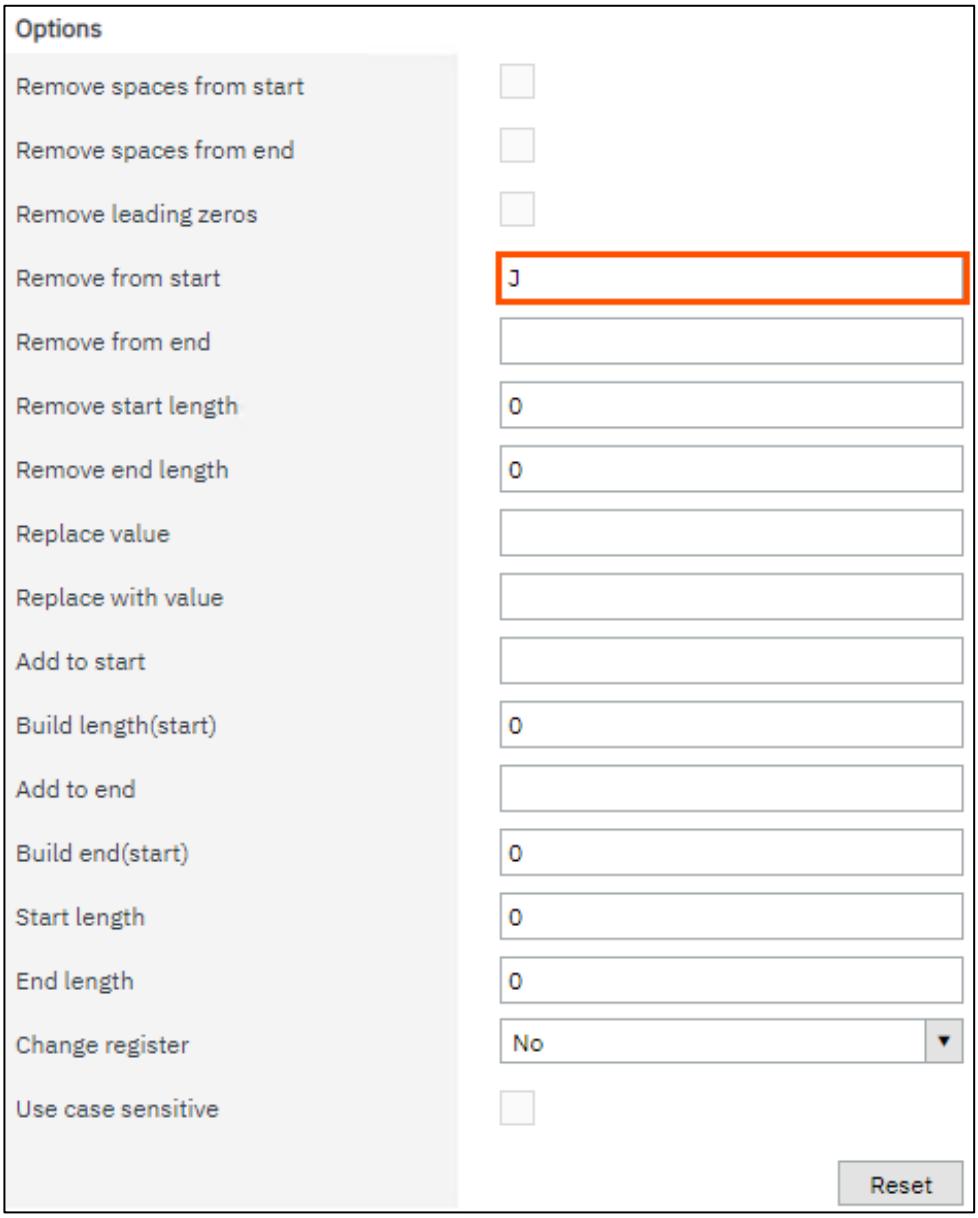
Task Locator
Task Locator reads incoming data from a signal and can perform data transformation services. In addition, the Task Locator can be used to change the key that Factory Automation will read for incoming task information and can be configured to limit the selection or records and use data filters.

The task locator feature includes a comprehensive set of options to transform data received from the signal.

For example, assume that the PLC controller stores the job number with a (J) as a leading character in the tag value and you need to remove the leading character so the job is matched in LYNQ. By using the task locator Remove from Start option, the leading character is easily removed.

The task locator may be used to remove spaces, remove leading zeros, remove characters, replace characters, change values to upper and lower case in the incoming signal data

Task Locator Options:



Resource Status Processor

Non Supported Property
Contact the Support Team to use a non-supported property. Non-supported properties are not supported at the signal level but can be supported by enabling settings within the database. All property methods are supported via REST API.

Refer to the table below to understand the purpose of each property and where the property can read its settings from.

Resource Status Processor Properties:

Property	Method	Supported	Purpose
Employee	Incoming Predefined By Seat	No Yes Yes	Employee recorded against the status transaction
Equipment	Incoming Predefined By Seat	No Yes Yes	Equipment recorded against the status transaction
Swap State	Incoming Predefined	Yes Yes	Resource status to be swapped based on scheme
Remain in Current Status	Incoming Predefined	No Yes	Resource to remain in status
Back to Previous	Incoming Predefined	Yes Yes	Resource to change to previous status
Status	Incoming Predefined Mapping	Yes Yes Yes	New status for the resource
Reset Accounting Date	Incoming Predefined Scheme	No Yes Yes	Accounting date should be reset for the resource
Clock In	Incoming Predefined Scheme	No Yes Yes	Clocked In Date to be reset after the transaction is created
Clock Out	Incoming Predefined Scheme	No Yes Yes	Clocked Out Date to be reset after the transaction is created
Stop task(s)	Incoming Predefined Scheme	No Yes Yes	All active tasks to be stopped when status changes
Processing Types	Incoming Predefined Scheme	No Yes Yes	Types of transactions to generate when status changes
Apply to Whole Crew	Incoming Predefined	No Yes	Status should apply to whole crew

Note: a mapping is required to map the machine status code to the MOM status code. More than one mapping can be configured.

Task Start/Stop Processor

Non Supported Property
Contact the Support Team to use a non-supported property. Non-supported properties are not supported at the signal level but can be supported by enabling settings within the database. All property methods are supported via REST API.

Refer to the table below to understand the purpose of each property and where the property can read its settings from.

Task Start/Stop Processor Properties:

Property	Method	Supported	Purpose
Employee	Incoming Predefined By Seat	No Yes Yes	Employee recorded against the labour transaction
Equipment	Incoming Predefined By Seat	No Yes Yes	Equipment recorded against the machine time transaction
Trigger Employee	Incoming Predefined	No Yes	Employee to be triggered for labour time collection
Trigger Equipment	Incoming Predefined	No Yes	Equipment to be triggered for machine time collection
Start/Stop Directive	Incoming Predefined	Yes Yes	
Task Locator	Incoming Predefined Locate	Yes Yes Yes	Which task to be started or stopped
Employee Split Time Mode	Incoming Predefined	No Yes	How labour time is to be split when multi-jobbing
Employee Split Time Parameter	Incoming Predefined Task Context Employee Equipment	No Yes Yes Yes Yes	How labour time is to be split when multi-jobbing in proportional split mode
Equipment Split Time Mode	Incoming Predefined	No Yes	How equipment time is to be split when multi-jobbing
Equipment Split Time Param	Incoming Predefined Task Context Employee Equipment	No Yes Yes Yes Yes	How machine time is to be split when multi-jobbing in proportional split mode
Classification Code	Incoming Predefined Task Context Employee Equipment	Yes Yes Yes Yes Yes	Which classification code to assign to the labour or machine time transaction
Comment	Incoming Predefined	Yes Yes	Comments for the labour or machine time transaction
Processing Types	Incoming Predefined	No Yes	Types of transactions to generate when task is started or stopped

Task Good Quantity Processor

Non Supported Property
Contact the Support Team to use a non-supported property. Non-supported properties are not supported at the signal level but can be supported by enabling settings within the database. All property methods are supported via REST API.

Refer to the table below to understand the purpose of each property and where the property can read its settings from.

Task Good Quantity Processor Properties (Properties):

Property	Method	Supported	Purpose
Employee	Incoming Predefined Task Context By Seat	No Yes Yes Yes Yes	Employee recorded against the quantity transaction
Equipment	Incoming Predefined Task Context By Seat	No Yes Yes Yes Yes	Equipment recorded against the quantity time transaction
Task Locator	Incoming Predefined Active Tasks Locate	Yes Yes Yes Yes	Task to report the quantity transaction against
Good Quantity	Incoming Predefined	Yes Yes	Good quantity value
Multiplier 1	Incoming Predefined Task Context Employee Equipment	Yes Yes Yes Yes Yes Yes	Good quantity multiplier value
Multiplier 2	Incoming Predefined Task Context Employee Equipment	Yes Yes Yes Yes Yes Yes	Good quantity multiplier value
Comments	Incoming Predefined	Yes Yes	Comments for the quantity transaction
Processing Types	Incoming Predefined	No Yes	Types of transactions to generate when quantity is reported

Multiplication factors can be static (i.e. defined against the Processor) or dynamically assigned (i.e. defined in the Routing Operation, Stock Code or Custom Form Fields etc). Multiplicators are typically used to indicate the number of units produced per operation cycle recorded with the PLC (i.e. multi-die forms when one punch of press creates multiple units)

Task Good Quantity Processor

Non Supported Property
Contact the Support Team to use a non-supported property. Non-supported properties are not supported at the signal level but can be supported by enabling settings within the database. All property methods are supported via REST API.

Task Good Quantity Processor Properties (Details):

Details	Method	Supported	Purpose
Location	Incoming Predefined Task Context Employee Equipment	Yes Yes Yes Yes Yes Yes	Location (site) recorded against the quantity transaction
Warehouse	Incoming Predefined Task Context Employee Equipment	Yes Yes Yes Yes Yes Yes	Warehouse recorded against the quantity transaction
Bin	Incoming Predefined Task Context Employee Equipment	Yes Yes Yes Yes Yes Yes	Bin number recorded against the quantity transaction
Traceability Code	Incoming Predefined Task Context Employee Equipment	Yes Yes Yes Yes Yes Yes	Lot number recorded against the quantity transaction
Serial Number	Incoming Predefined Task Context Employee Equipment	Yes Yes Yes Yes Yes Yes	Serial number recorded against the quantity transaction
Info 1 Info 2 Info 3 Info 4 Info 5	Incoming Predefined Task Context Employee Equipment	Yes Yes Yes Yes Yes Yes	Information recorded against the quantity transaction

Task Scrap Quantity Processor

Non Supported Property
Contact the Support Team to use a non-supported property. Non-supported properties are not supported at the signal level but can be supported by enabling settings within the database. All property methods are supported via REST API.

Refer to the table below to understand the purpose of each property and where the property can read its settings from.

Task Scrap Quantity Processor Properties (Properties):

Property	Method	Supported	Purpose
Employee	Incoming Predefined Task Context By Seat	No Yes Yes Yes Yes	Employee recorded against the scrap transaction
Equipment	Incoming Predefined Task Context By Seat	No Yes Yes Yes Yes	Equipment recorded against the scrap time transaction
Task Locator	Incoming Predefined Active Tasks Locate	Yes Yes Yes Yes	Task to report the scrap transaction against
Scrap Quantity	Incoming Predefined	Yes Yes	Good scrap value
Multiplier 1	Incoming Predefined Task Context Employee Equipment	Yes Yes Yes Yes Yes Yes	Good scrap multiplier value
Multiplier 2	Incoming Predefined Task Context Employee Equipment	Yes Yes Yes Yes Yes Yes	Good scrap multiplier value
Comments	Incoming Predefined	Yes Yes	Comments for the scrap transaction
Processing Types	Incoming Predefined	No Yes	Types of transactions to generate when scrap is reported

Multiplication factors can be static (i.e. defined against the Processor) or dynamically assigned (i.e. defined in the Routing Operation, Stock Code or Custom Form Fields etc). Multiplicators are typically used to indicate the number of units produced per operation cycle recorded with the PLC (i.e. multi-die forms when one punch of press creates multiple units)

Task Scrap Quantity Processor

Non Supported Property
Contact the Support Team to use a non-supported property. Non-supported properties are not supported at the signal level but can be supported by enabling settings within the database. All property methods are supported via REST API.

Task Scrap Quantity Processor Properties (Details):

Details	Method	Supported	Purpose
Location	Incoming Predefined Task Context Employee Equipment	Yes Yes Yes Yes Yes	Location recorded against the scrap transaction
Warehouse	Incoming Predefined Task Context Employee Equipment	Yes Yes Yes Yes Yes	Warehouse recorded against the scrap transaction
Bin	Incoming Predefined Task Context Employee Equipment	Yes Yes Yes Yes Yes	Bin number recorded against the scrap transaction
Traceability Code	Incoming Predefined Task Context Employee Equipment	Yes Yes Yes Yes Yes	Lot number recorded against the scrap transaction
Serial Number	Incoming Predefined Task Context Employee Equipment	Yes Yes Yes Yes Yes	Serial number recorded against the scrap transaction
Scrap Reason	Incoming Predefined Task Context Employee Equipment	Yes Yes Yes Yes Yes	Scrap reason recorded against the scrap transaction
Info 1 **Info 2** **Info 3** **Info 4** **Info 5**	Incoming Predefined Task Context Employee Equipment	Yes Yes Yes Yes Yes	Information recorded against the scrap transaction

Task Status Processor

Non Supported Property
Contact the Support Team to use a non-supported property. Non-supported properties are not supported at the signal level but can be supported by enabling settings within the database. All property methods are supported via REST API.

Refer to the table below to understand the purpose of each property and where the property can read its settings from.

Task Status Processor Properties:

Property	Method	Supported	Purpose
Employee	Incoming Predefined Task Context By Seat	No Yes Yes Yes	Employee recorded against the status change transaction
Equipment	Incoming Predefined Task Context By Seat	No Yes Yes Yes	Equipment recorded against the status change transaction
Task Locator	Incoming Predefined Locate Active Tasks	No Yes Yes Yes	Task to report the status change transaction against
Status	Incoming Predefined	No Yes	Status to change the task to (complete or open)
Comment	Incoming Predefined	Yes Yes	Comments for the status change transaction
Processing Types	Incoming Predefined	Yes Yes	Types of transactions to generate when status change is reported

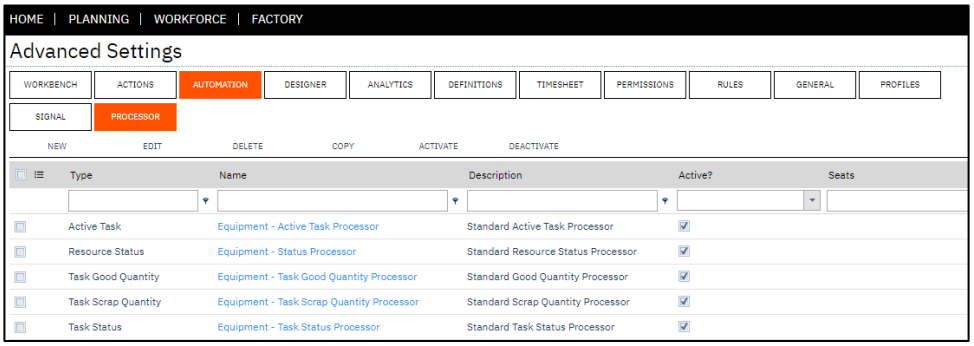
Enabling Default Processors

Caution
When editing a default processor, the settings will apply to all seats. Create new processors for each seat if required.

It is possible to configure automation without creating custom processors, assuming each seat does not require unique processor properties. Pre-shipped processors can be viewed and edited from Advanced Settings. Alternatively, a processor can be enabled and edited directly from Seat Maintenance.

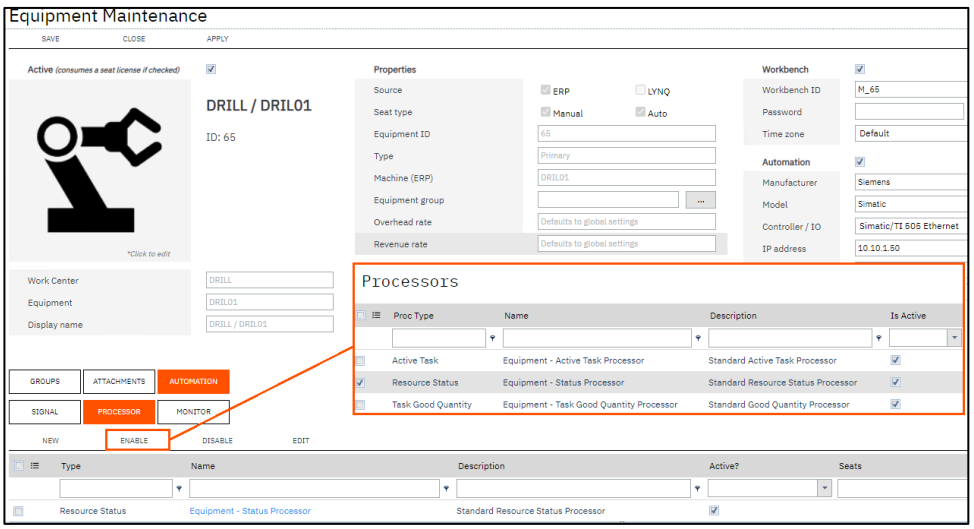
To view and edit processors in Advanced Settings:

1. From the LYNQ mom home page select Settings
2. Select Advanced Settings
3. Select Automation
4. Select Processor
5. Double click on the Processor record to view or edit the settings



To enable a processor in Seat Maintenance:

1. Select Seat Maintenance from the Resource Management menu area
2. Select the Equipment record and click Edit
3. Select the Automation Tab
4. Select Processor
5. Select Enable
6. Check the Processor(s) to enable
7. Select OK
8. Select Apply



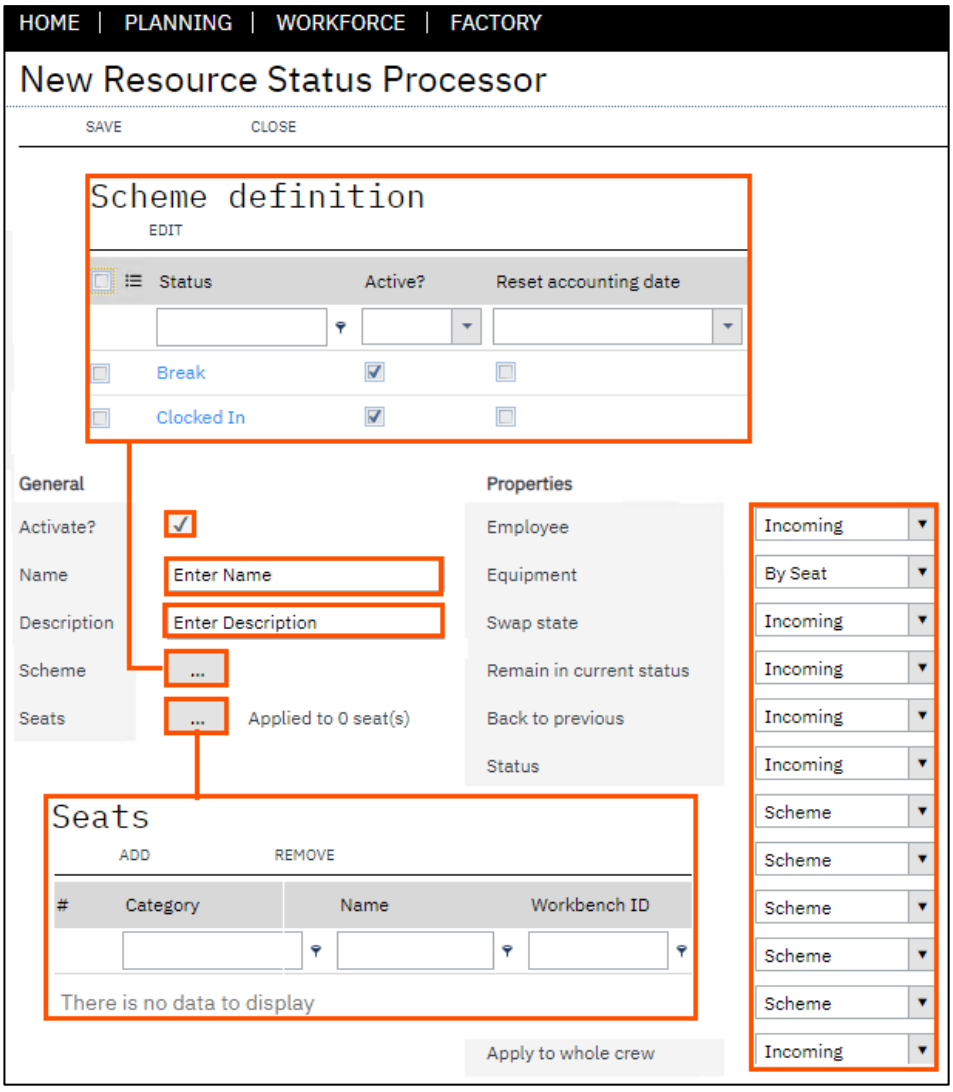
Creating Custom Processors

Processor Properties
Refer to the previous sections on processor properties

Where the default pre-shipped processors are insufficient for automation purposes, additional processors can be created. Processors are created and configured in advanced settings or directly from the equipment maintenance screen.

To create processors from advanced settings:

1. From the LYNQ mom home page select Settings
2. Select Advanced Settings
3. Select Automation
4. Select Processor
5. Select New
6. Select the Processor Type
7. Enter the Processor Name and Description
8. Using the Scheme Lookup [...] configure the scheme definition
9. Using the Seat Lookup [...] select the seats to apply the processor to
10. Select Add to select the seats
11. Select the relevant processor property setting methods
12. Select Save



Creating Custom Processors

Activated
By default, all statuses are activated. You only need to deactivate a status if you want to stop the processor from processing data for that particular status code.

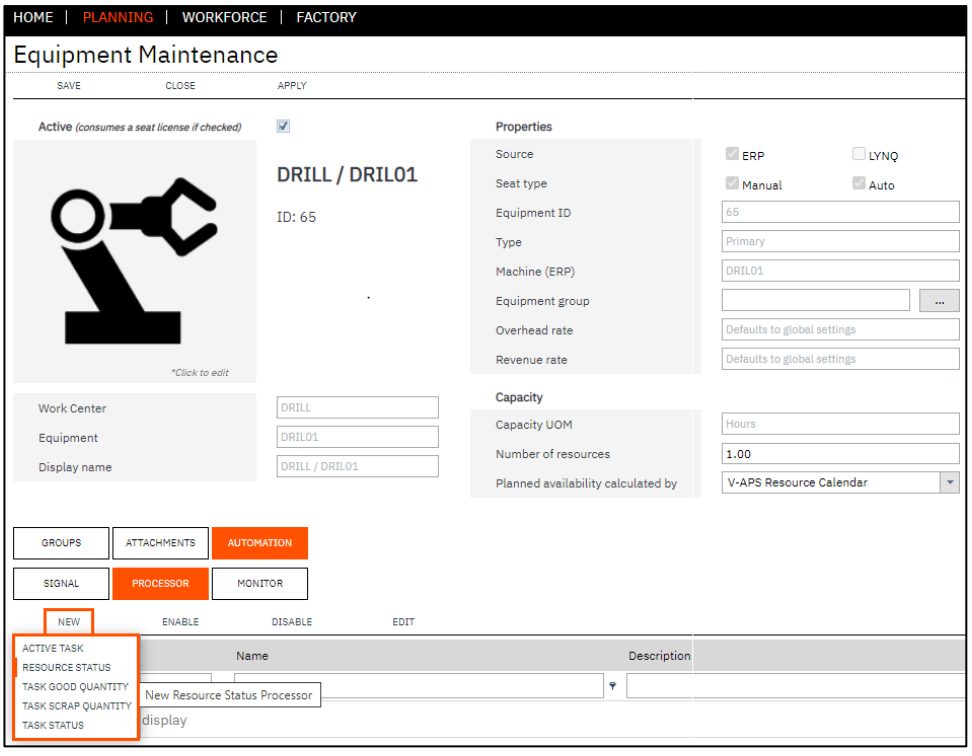
LYNQ REST API
Refer to the Rest API documentation in advanced settings/general to understand how to communicate with the processor outside of LYNQ.

New Processors
Use the New option to create your own custom processor

To create processors from equipment maintenance:

1. Select Seat Maintenance from the Resource Management menu area
2. Select the Equipment record and click Edit
3. Select the Automation Tab
4. Select Processor
5. Select New
6. Select the Processor Type
7. Enter the Processor Name and Description
8. Using the Scheme Lookup [...] configure the scheme definition
9. Select the relevant processor property setting methods
10. Select Save

Note: creating the processor from Equipment maintenance will automatically assign the processor to the seat.



Factory Automation Examples

Interval Settings
Please remember to change the interval settings back to the original values once you have completed these automation examples.

Now that you have a better understanding of how to create and configure the key components of factory automation (signals and processors), this section provides some examples that may help to further your learning. The examples will explain how to configure LYNQ end to end, to provide a factory automated environment using a few different simple scenarios. You may take these examples and apply them in your test or production environments or you may use this content for training purposes. It is recommended that you follow these examples in the order they are provided. Be careful if turning on automation in a production environment. It is recommended that you follow automation examples in a test environment first to ensure metrics such as OEE, OLE and TEEP and not affected, as a result.

The step by step instructions will explain how you can use REST API to simulate the creation of the factory data as if it is coming from the digitalised resource. This allows automation to be configured and tested prior to digitalisation of the physical equipment and the configuration of the OPC Server.

LYNQ provides a simulator feature that does much of this work for you but by using REST API, you will gain a better understanding of how automated data flows into LYNQ. The factory simulator is covered later in this guide.

Before following any step by step instructions, you must ensure your environment is prepared for factory automation. Factory automation does require separate licensing to be applied. Contact your customer success manager to understand how you can obtain the required licensing.

To prepare your environment:

1. Enable Factory Automation
    a. From the LYNQ mom home page select Settings
    b. Select Advanced Settings
    c. Select General
    d. Check the option to enable automation
2. Enable REST API
    a. From the LYNQ mom home page select Settings
    b. Select Advanced Settings
    c. Select General
    d. Enable REST API from the API section
    e. Select Save
3. Change the automation intervals
    a. From the LYNQ mom home page select Settings
    b. Select Advanced Settings
    c. Select General
    d. Set the Automation Polling Interval to 3 seconds
    e. Set the Data Buffer Interval to 3 seconds
    f. Set the Transaction Generation Interval to 3 seconds

Automating Resource Status

Follow All Examples
To gain a good understanding of how factory automation is configured, complete all examples provided in sequence.

Complex Scenarios
Complex scenarios are not covered in this guide and it is recommended that you partner with LYNQ to implement Factory Automation.

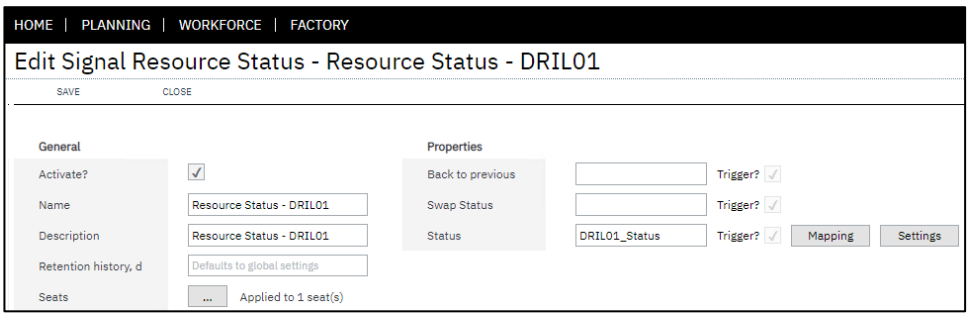
Let's assume you have a requirement to automate the status of a resource in LYNQ. Your factory equipment is capable of storing the status in the PLC controller. Each time the resource is turned on or off, you want to see the resource status update appropriately in LYNQ. The PLC stores the status of the equipment as 00 when the equipment is turned off and 01 when the equipment is turned on.

To demonstrate how this is configured, these steps by step instructions will use the default signals and processors pre-shipped with LYNQ.

Step 1 – Configure the Resource Status Signal:

1. Select Seat Maintenance from the Resource Management menu area
2. Select the Equipment record and click Edit (i.e. DRIL01)
3. Select the Automation Tab
4. Select the Signal Tab
5. Select New
6. Select the Resource Status Signal
7. Enter a name (i.e. Resource Status -DRIL01)
8. Enter a description (i.e. Resource Status – DRIL01)
9. Enter a unique tag name in the Status field (i.e. DRIL01_Status)
10. Select Save
11. Select Apply

Your signal settings should match the image below.



Step 2 – Configure the Resource Status Processor:

1. Select Seat Maintenance from the Resource Management menu area
2. Select the Equipment record and click Edit (i.e. DRIL01)
3. Select the Automation Tab
4. Select the Processor Tab
5. Select Enable
6. Select the Resource Status Processor
7. Select OK
8. Select the enabled Resource Status Processor (step 6)
9. Select Edit

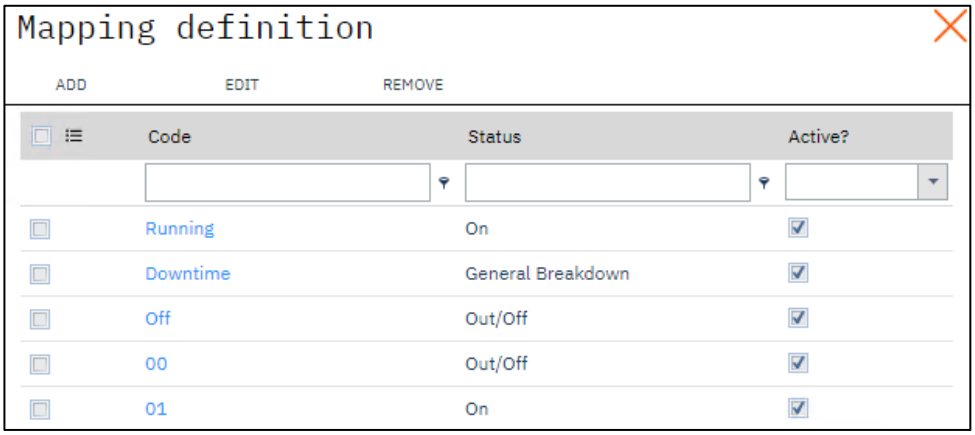
Automating Resource Status

Follow All Examples
To gain a good understanding of how factory automation is configured, complete all examples provided in sequence.

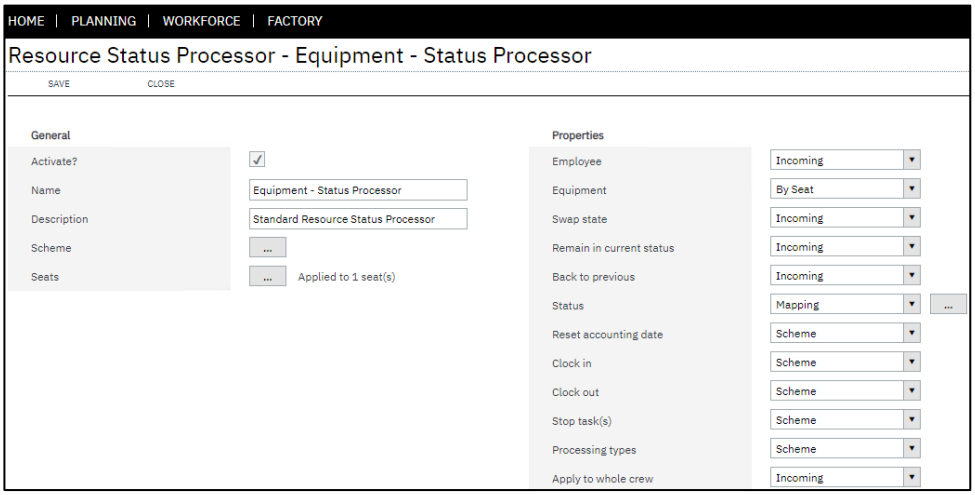
Complex Scenarios
Complex scenarios are not covered in this guide and it is recommended that you partner with LYNQ to implement Factory Automation.

10. Select Mapping from the drop down in the Status Properties field
11. Select the Mapping Lookup [...]
12. Select Add
13. Type 00 in the Code field
14. Change the status in the dropdown to Out/Off
15. Select OK
16. Select Add
17. Type 01 in the Code field
18. Change the status in the dropdown to On
19. Select OK
20. Select OK
21. Select Save

Your mapping definition should match the image below:



Your processor settings should match the image below:



Automating Resource Status

Follow All Examples
To gain a good understanding of how factory automation is configured, complete all examples provided in sequence.

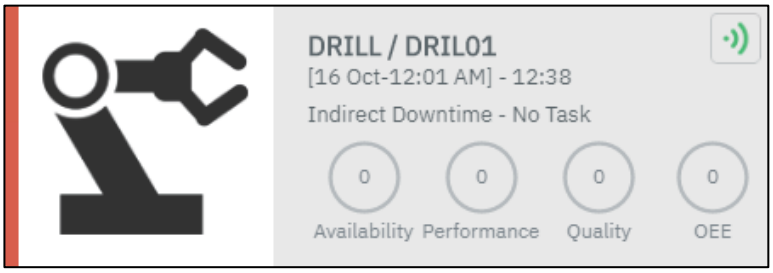
Complex Scenarios
Complex scenarios are not covered in this guide and it is recommended that you partner with LYNQ to implement Factory Automation.

REST API Documentation
You can access REST API documentation from http://localhost/lynqmom/DC/API/Docs/index.html# Remember to replace localhost/lynqmom with your LYNQ mom server and site names.

In this step you will send data to LYNQ using the REST API GET method via the browser to turn the equipment on/off. Prior to testing the automation, check that the seat you have configured for automation is turned off in LYNQ.

Step 3 – Send the tag data values to LYNQ to turn equipment on:

1. From a web browser copy the URL in step 2. You must edit the URL to match your environment:
    a. Change localhost/lynqmom to the correct mom website
    b. Change the tag name DRIL01_Status to the correct tag name (if created differently to example)
    c. These values are underlined below for easy recognition
2. **localhost/lynqmom**/api/TF/Accept/**DRIL01_Status**?moment=GetDateTime&tagValue=01&quality=5.2
3. Once the URL is edited, run the URL
4. If the URL is correctly formatted, you will not see any errors returned in the browser and the equipment status will be on



To turn the equipment off, run the same URL but change the tag value to 00. These values are underlined for quick recognition.

For example:

localhost/lynqmom/api/TF/Accept/DRIL01_Status?moment=GetDateTime&tagValue=**00**&quality=5.2

When valid automation data is received, you will see the indicator icon light up green. If you are seeing a red indicator, check the signal and processor settings are correctly configured.

Valid Signal Data Indicator 

Automating Task Start/Stop

Follow All Examples
To gain a good understanding of how factory automation is configured, complete all examples provided in sequence.

Complex Scenarios
Complex scenarios are not covered in this guide and it is recommended that you partner with LYNQ to implement Factory Automation.

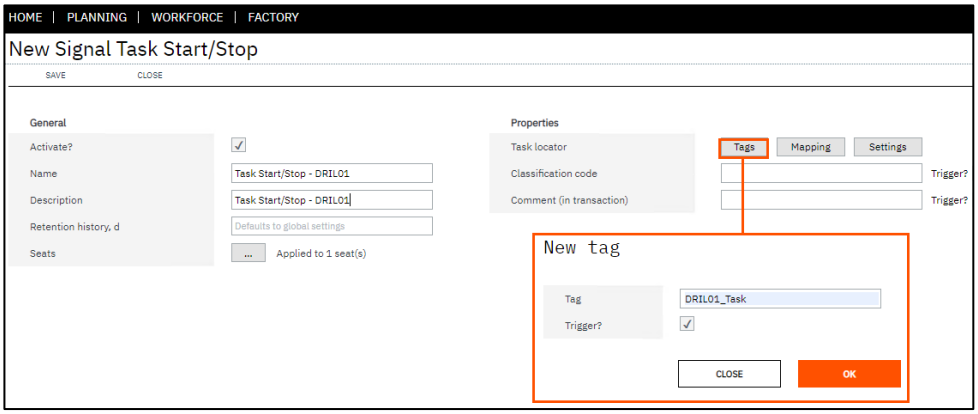
Let's assume you have a requirement to automate a task starting on a resource in LYNQ. Your factory equipment is capable of storing the task or job number in the PLC controller. The PLC controller stores the task as Job123.1.Run, which is in the same format as the task recognised by LYNQ. When the task starts the resource should track equipment time in LYNQ.

To demonstrate how this is configured, these steps by step instructions will use the default signals and processors pre-shipped with LYNQ.

Step 1 – Configure the Task Start/Stop Signal:

1. Select Seat Maintenance from the Resource Management menu area
2. Select the Equipment record and click Edit (i.e. DRIL01)
3. Select the Automation Tab
4. Select the Signal Tab
5. Select New
6. Select the Task Start/Stop Signal
7. Enter a name (i.e. Task Start/Stop -DRIL01)
8. Enter a description (i.e. Task Start/Stop – DRIL01)
9. Select Tags
10. Select Add
11. Enter a unique tag name in the Tag field (i.e. DRIL01_Task)
12. Select OK
13. Select OK
14. Select Save
15. Select Apply

Your signal settings should match the image below.



Automating Task Start/Stop

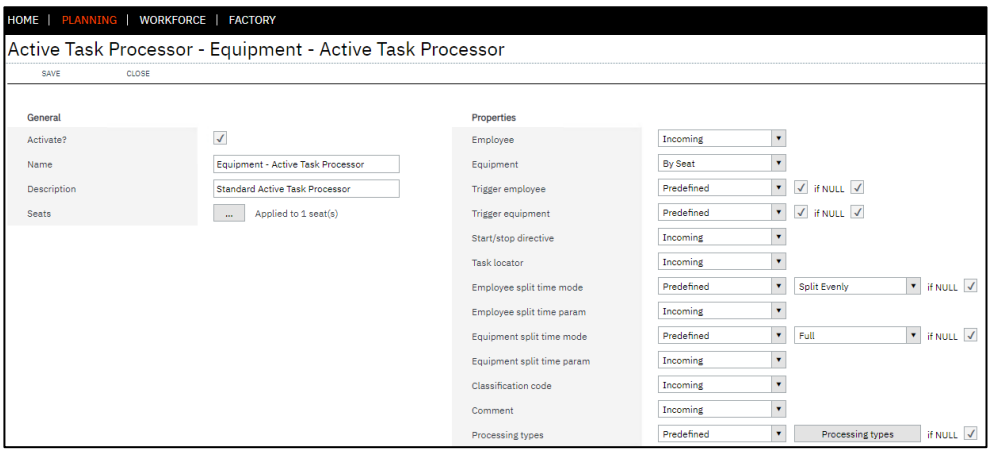
Follow All Examples
To gain a good understanding of how factory automation is configured, complete all examples provided in sequence.

Complex Scenarios
Complex scenarios are not covered in this guide and it is recommended that you partner with LYNQ to implement Factory Automation.

Step 2 – Configure the Active Task Processor:

1. Select Seat Maintenance from the Resource Management menu area
2. Select the Equipment record and click Edit (i.e. DRIL01)
3. Select the Automation Tab
4. Select the Processor Tab
5. Select Enable
6. Select the Active Task Processor
7. Select OK
8. Select Apply
9. Select the enabled Active Task Processor (step 6)
10. Select Edit
11. Ensure the Equipment dropdown value is set to By Seat
12. Ensure the Trigger Equipment dropdown value is set to Predefined
13. Ensure the Task Locator dropdown value is set to Incoming
14. Select Save

Your processor settings should match the image below.



Automating Task Start/Stop

Follow All Examples
To gain a good understanding of how factory automation is configured, complete all examples provided in sequence.

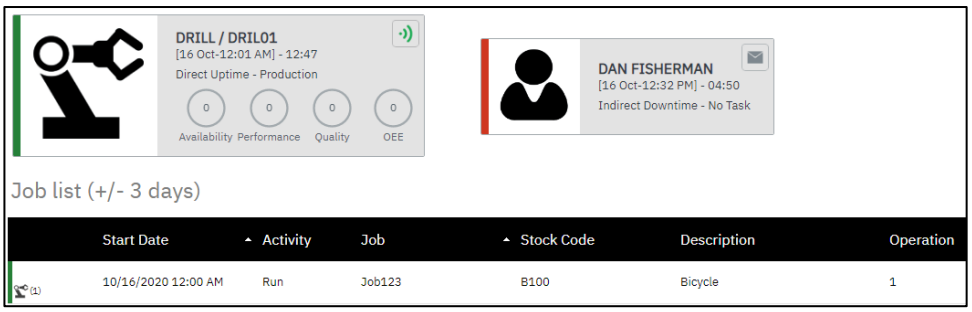
Complex Scenarios
Complex scenarios are not covered in this guide and it is recommended that you partner with LYNQ to implement Factory Automation.

REST API Documentation
You can access REST API documentation from http://localhost/lynqmom/DC/API/Docs/index.html# Remember to replace localhost/lynqmom with your LYNQ mom server and site names.

In this step you will send data to LYNQ using the REST API GET method via the browser to start the task on the equipment. Prior to testing the automation, check that the seat you have configured for automation is turned on from the previous example.

Step 3 – Send the tag data values to LYNQ to start a task:

1. From a web browser copy the URL in step 2. You must edit the URL to match your environment:
    a. Change localhost/lynqmom to the correct mom website
    b. Change the tag name DRIL01.Task value to the correct tag name (if created differently to example)
    c. Change the job number, operation number and activity to match a job in LYNQ. Note the %E2 syntax for a period (.) Make sure you do not change this syntax. LYNQ recognises tasks in the format JOB123.1.Run (Job.Operation.Task)
    d. These values are underlined below for easy recognition
2. **localhost/lynqmom**/api/TF/Accept/**DRIL01_Task?**moment=GetDateTime&tagValue=**Job123%2E1%2ERun**&quality=5.2
3. Once the URL is edited, run the URL
4. If the URL is correctly formatted, you will not see any errors returned in the browser and the job will be running



If the task is not started on the equipment, check the signal and processor settings are correctly configured and you have specified the correct job number, operation number and task value in the URL. The task must be visible in LYNQ, so check you can see this in Job Status

.

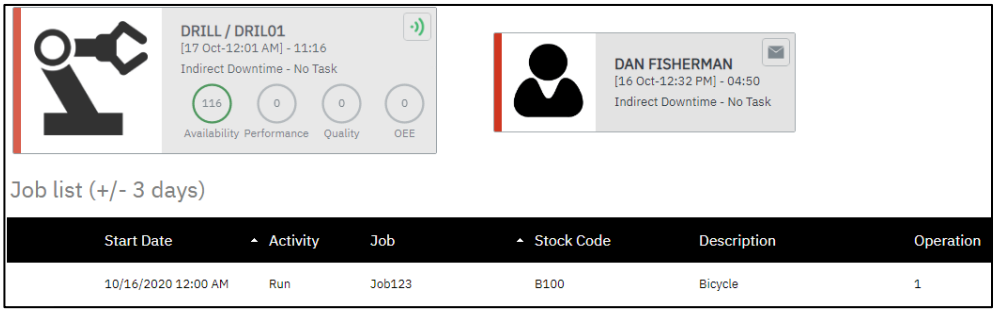
Automating Task Start/Stop

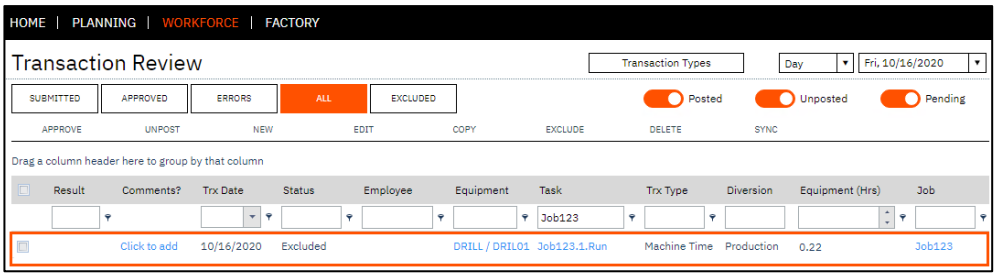
Follow All Examples
To gain a good understanding of how factory automation is configured, complete all examples provided in sequence.

Complex Scenarios
Complex scenarios are not covered in this guide and it is recommended that you partner with LYNQ to implement Factory Automation.

Step 4 – Send the tag data values to LYNQ to stop a task:

1. From a web browser copy the URL that you used in the previous step to start a task. To stop the task, the tag value can be removed. See step 2 for an example.
2. localhost/lynqmom/api/TF/Accept/DRIL01_Task?moment=GetDateTime&**tagValue=**&quality=5.2
3. Once the URL is edited, run the URL
4. If the URL is correctly formatted, you will not see any errors returned in the browser and the task will be stopped on the equipment and a machine time transaction will be generated.





NOTE: if following examples in a production environment. Test transactions should be deleted manually from Transaction Review.

If the equipment sends another task code in the same tag, the default behaviour in LYNQ is to stop the previous task and start the new task. LYNQ supports the ability to configure unique tags for each start and stop directive. In addition, LYNQ can be configured to run multiple tasks simultaneously and locate tasks where the incoming tag value does not match a task value in LYNQ.

Automating Good Quantity

Follow All Examples
To gain a good understanding of how factory automation is configured, complete all examples provided in sequence.

Complex Scenarios
Complex scenarios are not covered in this guide and it is recommended that you partner with LYNQ to implement Factory Automation.

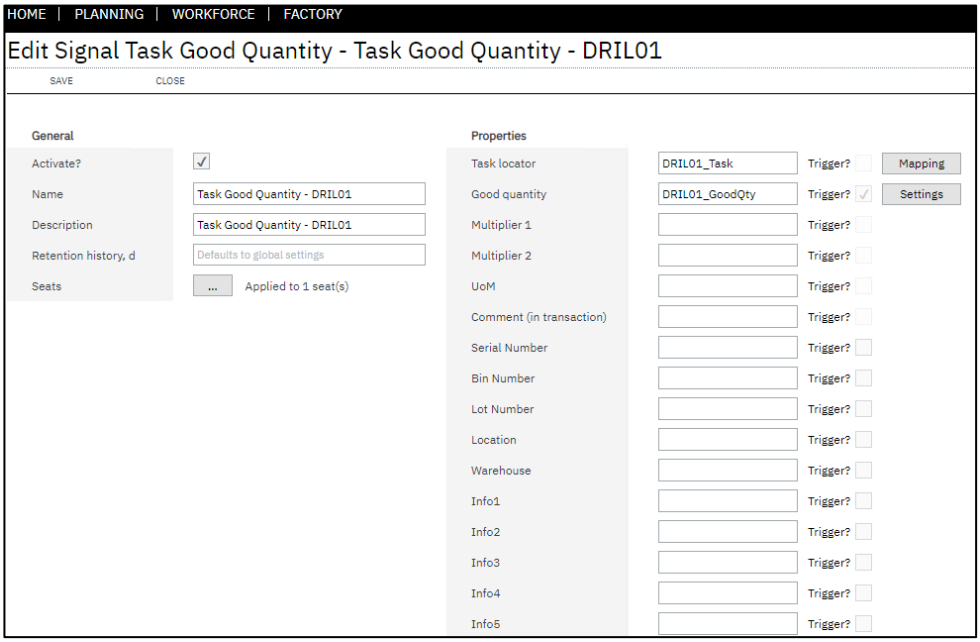
Let's assume you have a requirement to automate the good quantity produced in LYNQ. Your factory equipment is capable of storing the good quantity in the PLC controller. The PLC controller stores the quantity as an incremental value each time a quantity is produced. The good quantity must be reported against the runtime portion of operation 1 for the Job 123 in LYNQ.

To demonstrate how this is configured, these steps by step instructions will use the default signals and processors pre-shipped with LYNQ.

Step 1 – Configure the Task Good Quantity Signal:

1. Select Seat Maintenance from the Resource Management menu area
2. Select the Equipment record and click Edit (i.e. DRIL01)
3. Select the Automation Tab
4. Select the Signal Tab
5. Select New
6. Select the Task Good Quantity Signal
7. Enter a name (i.e. Task Good Quantity -DRIL01)
8. Enter a description (i.e. Task Good Quantity – DRIL01)
9. Enter the tag name DRIL01_Task in the Task Locator field
10. Enter the tag name DRIL01_GoodQty in the Good Quantity field
11. Select Save
12. Select Apply

Your signal settings should match the image below.



Automating Good Quantity

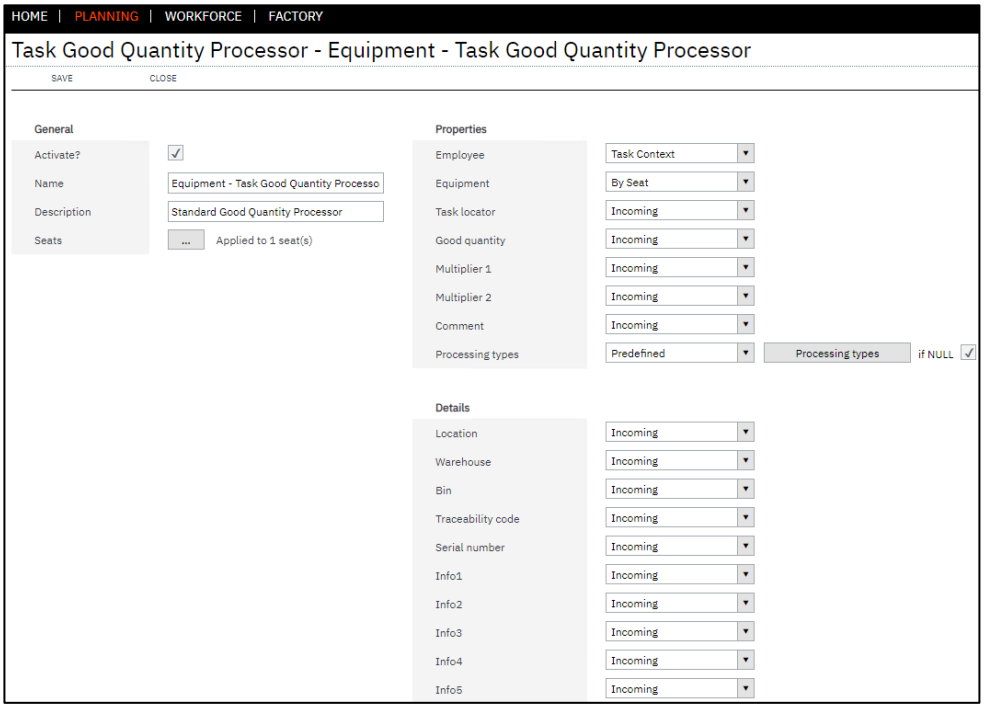
Follow All Examples
To gain a good understanding of how factory automation is configured, complete all examples provided in sequence.

Complex Scenarios
Complex scenarios are not covered in this guide and it is recommended that you partner with LYNQ to implement Factory Automation.

Step 2 – Configure the Task Good Quantity Processor:

1. Select Seat Maintenance from the Resource Management menu area
2. Select the Equipment record and click Edit (i.e. DRIL01)
3. Select the Automation Tab
4. Select the Processor Tab
5. Select Enable
6. Select the Task Good Quantity Processor
7. Select OK
8. Select Apply
9. Select the enabled Task Good Quantity Processor (step 6)
10. Select Edit
11. Ensure the Task Locator property value is set to Incoming
12. Ensure the Good Quantity property value is set to Incoming
13. Select Save

Your processor settings should match the image below.



Automating Good Quantity

Follow All Examples
To gain a good understanding of how factory automation is configured, complete all examples provided in sequence.

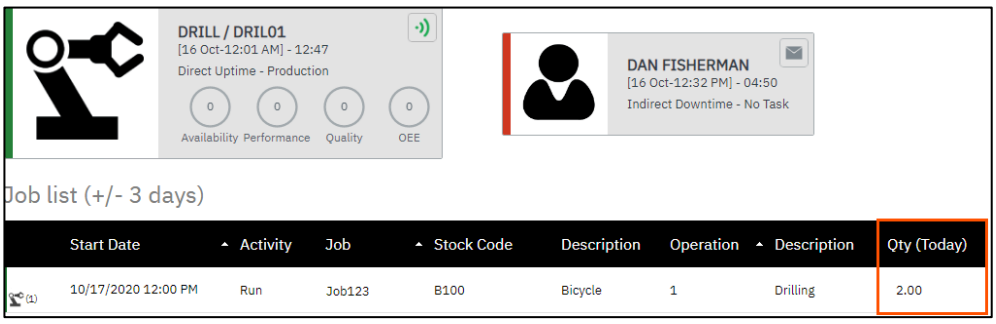
Complex Scenarios
Complex scenarios are not covered in this guide and it is recommended that you partner with LYNQ to implement Factory Automation.

REST API Documentation
You can access REST API documentation from http://localhost/lynqmom/DC/API/Docs/index.html# Remember to replace localhost/lynqmom with your LYNQ mom server and site names.

In this step you will send data to LYNQ using the REST API GET method via the browser to report good quantity for the task. Prior to testing the automation, check that the seat you have configured for automation is turned on and running the task from the previous automation example.

Step 3 – Send the tag data values to LYNQ to report good quantity:

1. From a web browser copy the URL in step 2. You must edit the URL to match your environment:
    a. Change localhost/lynqmom to the correct mom website
    b. Change the tag name DRIL01.Task value to the correct tag name (if created differently to example)
    c. Change the job number, operation number and activity to match a job in LYNQ. Note the %E2 syntax for a period (.) Make sure you do not change this syntax. LYNQ recognises tasks in the format JOB123.1.Run (Job.Operation.Task)
    d. These values are underlined below for easy recognition
2. **localhost/lynqmom/**api/TF/Accept/**DRIL01_GoodQty**?moment=GetDateTime&tagValue=0&quality=5.2
3. Once the URL is edited, run the URL
4. Now run the same URL again but change the tagValue to 1
5. **localhost/lynqmom/**api/TF/Accept/**DRIL01_GoodQty**?moment=GetDateTime&tagValue=1&quality=5.2
6. Once the URL is edited, run the URL
7. Now run the same URL again but change the tagValue to 2
8. **localhost/lynqmom/**api/TF/Accept/**DRIL01_GoodQty**?moment=GetDateTime&tagValue=2&quality=5.2
9. Once the URL is edited, run the URL
10. If the URL is correctly formatted, you will not see any errors returned in the browser and the good quantity will be reported against the task and quantity transactions will be generated



Note: when running the tests in a production environment. Test transactions should be deleted manually from Transaction Review.

Automating Scrap Quantity

Follow All Examples
To gain a good understanding of how factory automation is configured, complete all examples provided in sequence.

Complex Scenarios
Complex scenarios are not covered in this guide and it is recommended that you partner with LYNQ to implement Factory Automation.

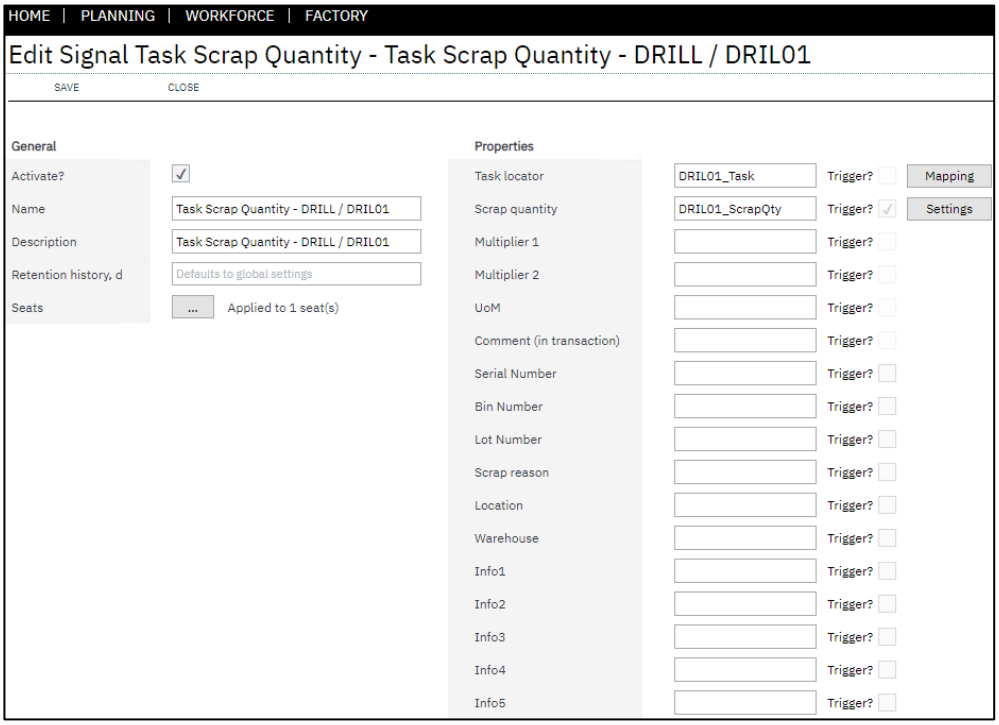
Let's assume you have a requirement to automate the scrap quantity produced in LYNQ. Your factory equipment is capable of storing the scrap quantity in the PLC controller. The PLC controller stores the quantity as an incremental value each time a quantity is produced. The scrap quantity must be reported against the runtime portion of operation 1 for the Job 123 in LYNQ.

To demonstrate how this is configured, these steps by step instructions will use the default signals and processors pre-shipped with LYNQ.

Step 1 – Configure the Task Scrap Quantity Signal:

1. Select Seat Maintenance from the Resource Management menu area
2. Select the Equipment record and click Edit (i.e. DRIL01)
3. Select the Automation Tab
4. Select the Signal Tab
5. Select New
6. Select the Task Scrap Quantity Signal
7. Enter a name (i.e. Task Scrap Quantity -DRIL01)
8. Enter a description (i.e. Task Scrap Quantity – DRIL01)
9. Enter the tag name DRIL01_Task in the Task Locator field
10. Enter the tag name DRIL01_ScrapQty in the Scrap Quantity field
11. Select Save
12. Select Apply

Your signal settings should match the image below.



Automating Scrap Quantity

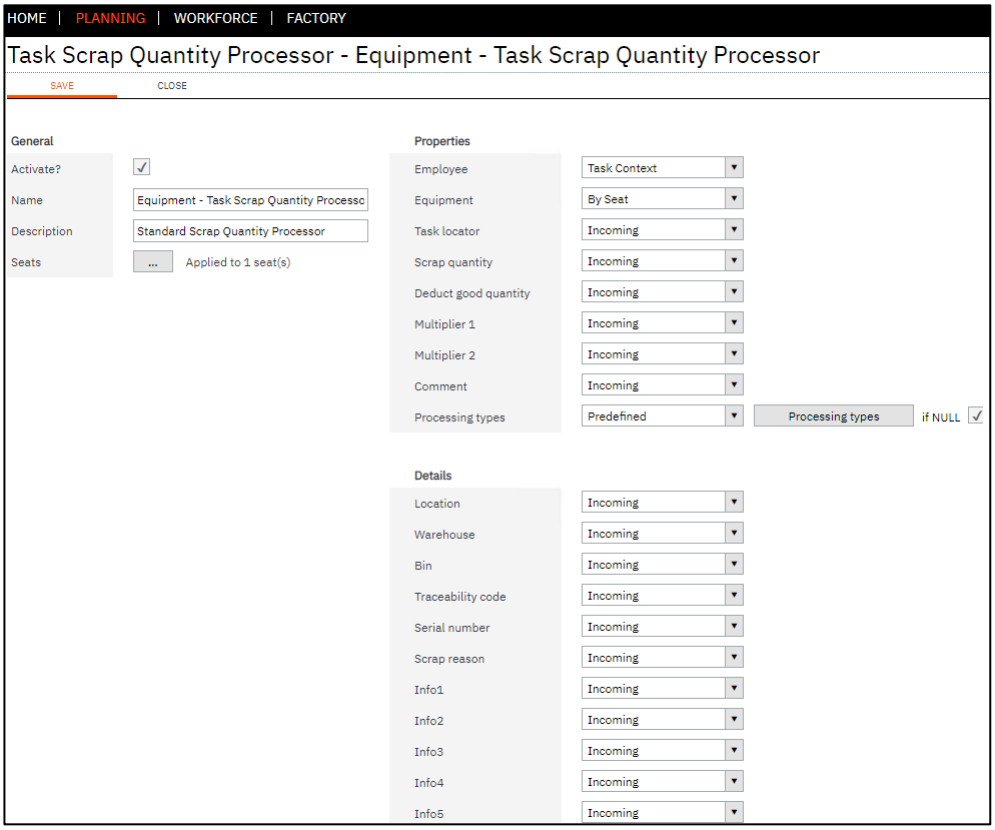
Follow All Examples
To gain a good understanding of how factory automation is configured, complete all examples provided in sequence.

Complex Scenarios
Complex scenarios are not covered in this guide and it is recommended that you partner with LYNQ to implement Factory Automation.

Step 2 – Configure the Task Scrap Quantity Processor:

1. Select Seat Maintenance from the Resource Management menu area
2. Select the Equipment record and click Edit (i.e. DRIL01)
3. Select the Automation Tab
4. Select the Processor Tab
5. Select Enable
6. Select the Task Scrap Quantity Processor
7. Select OK
8. Select Apply
9. Select the enabled Task Scrap Quantity Processor (step 6)
10. Select Edit
11. Ensure the Task Locator property value is set to Incoming
12. Ensure the Scrap Quantity property value is set to Incoming
13. Select Save

Your processor settings should match the image below.



Automating Scrap Quantity

Follow All Examples
To gain a good understanding of how factory automation is configured, complete all examples provided in sequence.

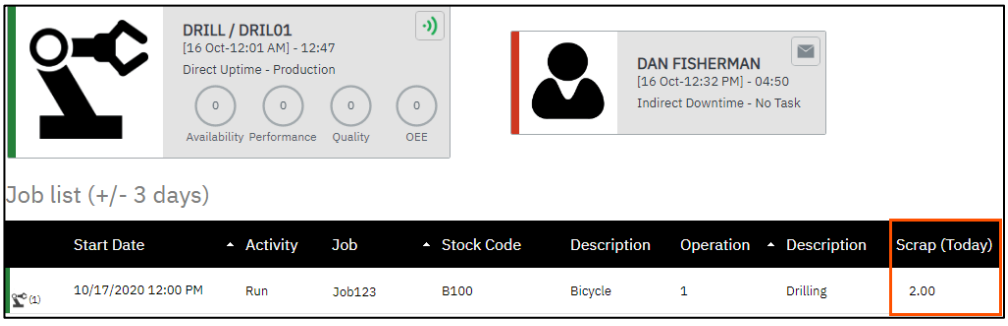
Complex Scenarios
Complex scenarios are not covered in this guide and it is recommended that you partner with LYNQ to implement Factory Automation.

REST API Documentation
You can access REST API documentation from http://localhost/lynqmom/DC/API/Docs/index.html# Remember to replace localhost/lynqmom with your LYNQ mom server and site names.

In this step you will send data to LYNQ using the REST API GET method via the browser to report scrap quantity for the task. Prior to testing the automation, check that the seat you have configured for automation is turned on and running the task from the previous automation example.

Step 3 – Send the tag data values to LYNQ to report scrap quantity:

1. From a web browser copy the URL in step 2. You must edit the URL to match your environment:
    a. Change localhost/lynqmom to the correct mom website
    b. Change the tag name DRIL01.Task value to the correct tag name (if created differently to example)
    c. Change the job number, operation number and activity to match a job in LYNQ. Note the %E2 syntax for a period (.) Make sure you do not change this syntax. LYNQ recognises tasks in the format JOB123.1.Run (Job.Operation.Task)
    d. These values are underlined below for easy recognition
2. **localhost/lynqmom**/api/TF/Accept/**DRIL01_ScrapQty**?moment=GetDateTime&tagValue=0&quality=5.2
3. Once the URL is edited, run the URL
4. Now run the same URL again but change the tagValue to 1
5. **localhost/lynqmom**/api/TF/Accept/**DRIL01_ScrapQty**?moment=GetDateTime&tagValue=1&quality=5.2
6. Once the URL is edited, run the URL
7. Now run the same URL again but change the tagValue to 2
8. **localhost/lynqmom**/api/TF/Accept/**DRIL01_ScrapQty**?moment=GetDateTime&tagValue=2&quality=5.2
9. Once the URL is edited, run the URL
10. If the URL is correctly formatted, you will not see any errors returned in the browser and the scrap quantity will be reported against the task



Note: when running the tests in a production environment. Test transactions should be deleted manually from Transaction Review.

Automating Task Status

Follow All Examples
To gain a good understanding of how factory automation is configured, complete all examples provided in sequence.

Complex Scenarios
Complex scenarios are not covered in this guide and it is recommended that you partner with LYNQ to implement Factory Automation.

Expected Values
To change the status of a task, the processor expects to receive the values as 'C' for complete and 'N' for Re-open (without the single quotes).

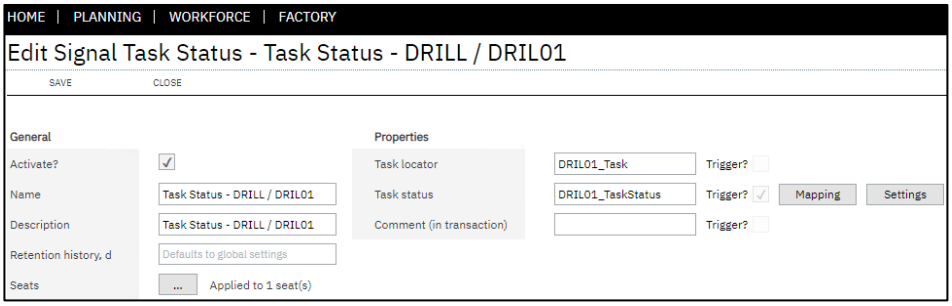
Let's assume you have a requirement to automate the status of a task in LYNQ. Your factory equipment is capable of storing the task status in the PLC controller. The PLC controller stores the status as C for complete. The task status must be reported against the runtime portion of operation 1 for the Job 123 in LYNQ.

To demonstrate how this is configured, these steps by step instructions will use the default signals and processors pre-shipped with LYNQ.

Step 1 – Configure the Task Status Signal:

1. Select Seat Maintenance from the Resource Management menu area
2. Select the Equipment record and click Edit (i.e. DRIL01)
3. Select the Automation Tab
4. Select the Signal Tab
5. Select New
6. Select the Task Status Signal
7. Enter a name (i.e. Task Status -DRIL01)
8. Enter a description (i.e. Task Status – DRIL01)
9. Enter the tag name DRIL01_Task in the Task Locator field
10. Enter the tag name DRIL01_TaskStatus in the Task Status field
11. Select Save
12. Select Apply

Your signal settings should match the image below.



Automating Task Status

Follow All Examples
To gain a good understanding of how factory automation is configured, complete all examples provided in sequence.

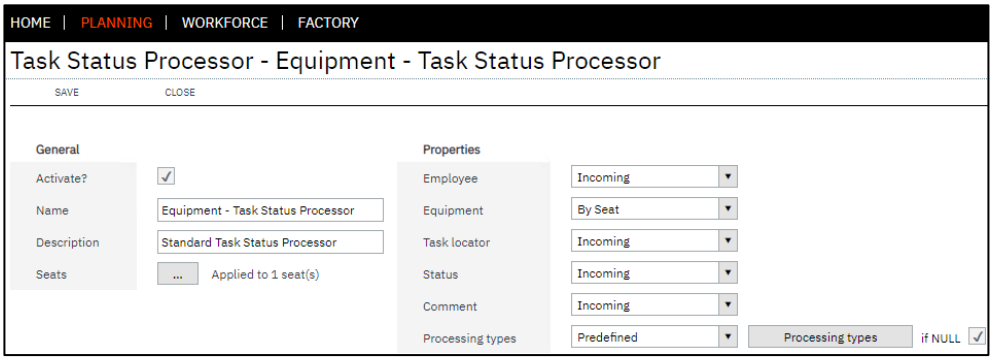
Complex Scenarios
Complex scenarios are not covered in this guide and it is recommended that you partner with LYNQ to implement Factory Automation.

Expected Values
To change the status of a task, the processor expects to receive the values as 'C' for complete and 'N' for Re-open (without the single quotes).

Step 2 – Configure the Task Status Processor:

1. Select Seat Maintenance from the Resource Management menu area
2. Select the Equipment record and click Edit (i.e. DRIL01)
3. Select the Automation Tab
4. Select the Processor Tab
5. Select Enable
6. Select the Task Status Processor
7. Select OK
8. Select Apply
9. Select the enabled Task Status Processor (step 6)
10. Select Edit
11. Ensure the Task Locator property value is set to Incoming
12. Ensure the Status property value is set to Incoming
13. Select Save

Your processor settings should match the image below.



Automating Task Status

Follow All Examples
To gain a good understanding of how factory automation is configured, complete all examples provided in sequence.

Complex Scenarios
Complex scenarios are not covered in this guide and it is recommended that you partner with LYNQ to implement Factory Automation.

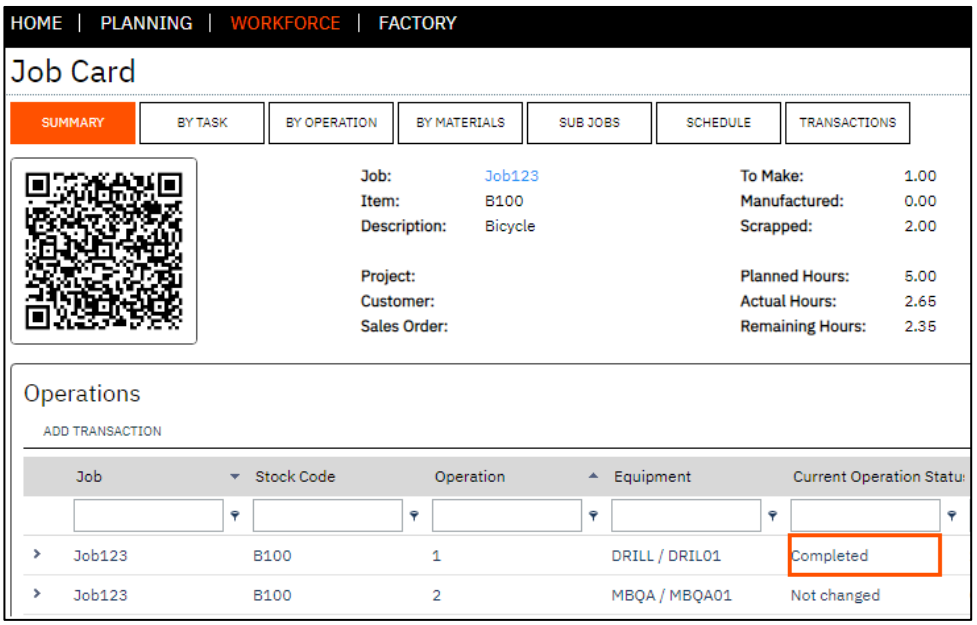
Expected Values
To change the status of a task, the processor expects to receive the values as 'C' for complete and 'N' for Re-open (without the single quotes).

REST API Documentation
You can access REST API documentation from http://localhost/lynqmom/DC/API/Docs/index.html# Remember to replace localhost/lynqmom with your LYNQ mom server and site names.

In this step you will send data to LYNQ using the REST API GET method via the browser to change the task status to complete. Prior to testing the automation, check that the seat you have configured for automation is turned on and running the task from the previous automation example.

Step 3 – Send the tag data values to LYNQ to report the complete task status:

1. From a web browser copy the URL in step 2. You must edit the URL to match your environment:
    a. Change localhost/lynqmom to the correct mom website
    b. Change the tag name DRIL01_TaskStatus value to the correct tag name (if created differently to example)
    c. Change the job number, operation number and activity to match a job in LYNQ. Note the %E2 syntax for a period (.) Make sure you do not change this syntax. LYNQ recognises tasks in the format JOB123.1.Run (Job.Operation.Task)
    d. These values are underlined below for easy recognition
2. **localhost/lynqmom**/api/TF/Accept/**DRIL01_TaskStatus**?moment=GetDateTime&tagValue=C&quality=5.2
3. Once the URL is edited, run the URL
4. If the URL is correctly formatted, you will not see any errors returned in the browser and the status of the task will change to completed. You will also see a status transaction in Transaction Review.



Note: when running the tests in a production environment. Test transactions should be deleted manually from Transaction Review.

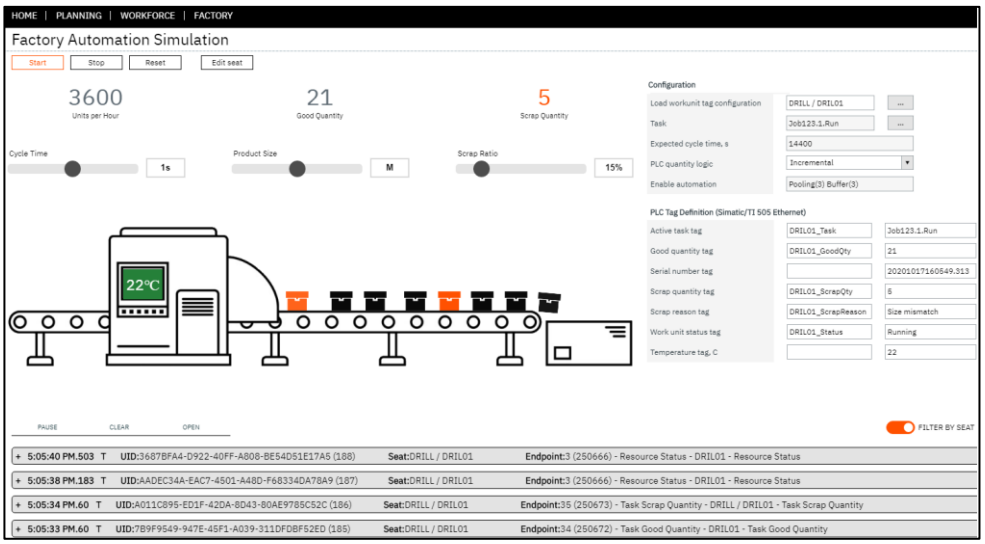
Factory Automation Simulation

20 Minute Timeout
The simulator will stop automatically after 20 minutes if left running.

LYNQ includes a feature that will allow you to simulate your factory automation environment. The simulator recognises the different signals and processors that have been applied to the seat. When the simulation is started, LYNQ will automatically send values to the signals based on the options you provide for:

- Cycle Time
- Product Size
- Scrap Ratio

The simulator works much in the same way as sending tag values to LYNQ using REST API. This feature provides a very quick and easy way to test your factory automation configuration and is an excellent tool for those users that are less technically minded.



NOTE: running this simulator will generate transactions but these can be deleted afterwards from the Transaction Review screen.

To start the Simulator:

1. Select Seat Maintenance from the Resource Management menu area
2. Select the Equipment record and click Edit (i.e. DRIL01)
3. Select Run from the Automation section
4. From the Simulation screen select the task lookup [...]
5. Select a task
6. Adjust the cycle time, product size and scrap ratio as required
7. Select Start

If everything has been configured correctly you will see transactions for the signals defined, reported against the job.

To stop the Simulator:

1. Select Stop

Monitoring Live Data

System Insights
Monitoring can also be run from the Automation Tab in the System Insights page.

No Data in Monitor?
Refer to the Troubleshooting section if you are not seeing any data in the Monitor.

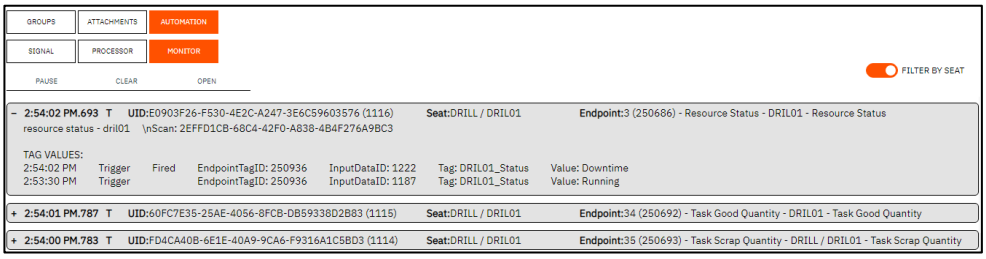
The actual live data coming from the factory floor can be visualised in the monitor. The monitor is optimised for mobile devices and is a useful tool for troubleshooting physical dataflow issues between multiple layers of the automated solution. (Machine, PLC, Network, OPC Server, LYNQ).

The monitor displays information relating to the signals that are active in LYNQ.

To start the monitor:

1. Select Seat Maintenance from the Resource Management menu area
2. Select the Equipment record and click Edit
3. Select the Automation Tab
4. Select Monitor
5. Select Start

Expand the row to view the detail captured for the OPC Tag



To pause the monitor:

1. Select Monitor
2. Select Pause

To clear the monitor:

1. Select Monitor
2. Select Clear

To stop the monitor:

1. Select Monitor
2. Select Stop

Monitoring Live Data

Red Indicator Icon
You will continue to see a Red Indicator Icon for an automated device until you resolve all invalid data entries for the accounting day.

The Automation Status Indicator on the Equipment Status screen and on the Workbench screen provides another useful tool for monitoring the health of an automated device.

The Automation Status Indicator has 5 different status meanings:

Indicator	Color/Image Code	Meaning	
	Finger	Factory Automation is disabled but Manual Data Collection is enabled	
	WIFI Grey with Diagonal Line	Device is not activated for Automation and Manual Data Collection is disabled	
	WIFI Grey	Device is activated for Automation but no data has been received	
	WIFI Green	Device is activated for Automation and valid data has been received on the accounting day	
	WIFI Red	Device is activated for Automation and unresolved invalid data has been received on the accounting day	

Error Handling

No Data visible in Factory Automation
Check that you have enabled signals and processors for the data you are expecting LYNQ to capture.

Source Column
The source column indicates where the data originated from. Filter the Source column by the value 'automation' to only show the transactions that have been created by the Factory Automation Process. By default, you will see records for manual actions and workbench.

Customise Screen
You can add/remove columns to the Factory Automation screen by clicking on Customise.

Deleting Transactions
You cannot delete transactions with a flow status of Trx Generated.

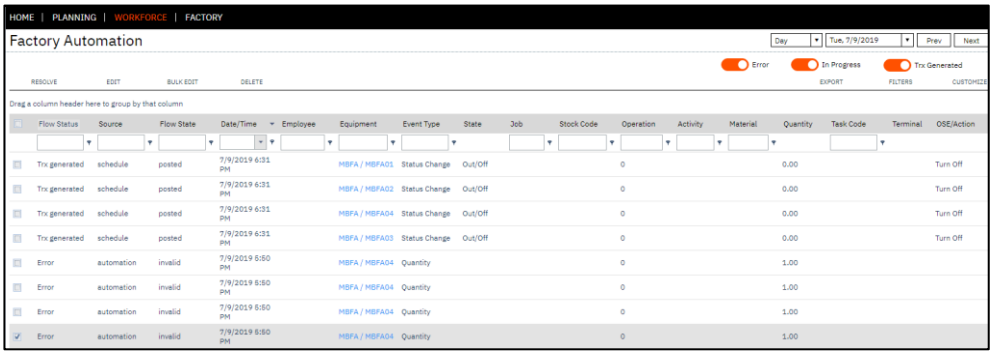
System Insights
An ungrouped view of the factory automation data can be found on the Events Tab of the System Insights screen.

There may be occasions where data received from an automated device is not visible in the workbench or in reports and/or dashboard screens in LYNQ. If you are seeing the data on the Monitor but cannot report on the data in LYNQ you should use the Factory Automation screen to investigate the reasons for this problem.

Similar to the process flow in the Transaction Review Screen, Factory Automation data must pass through stages before the data becomes a valid transaction that can been seen in the Transaction Review Screen and in turn, in other LYNQ dashboard and reporting screens.

Data from Automated Devices must pass through these flow states:

Factory Automation Screen



There are in total 5 flow states in LYNQ.

Flow State	Type	Meaning
	Invalid	Invalid data received or no Job running (Qty)
	Pending Accept	Waiting for the Processor to run based on Data Buffer Interval
	Accepted	Data accepted by the LYNQ Platform Service (Service must be started, service polls every 1 minute)
	Accept Error	Internal Acceptance error due to data integrity issues
	Instant Accept	Transaction was generated via the Workbench

The Factory Automation screen groups these different flow states into a simplified view of the flow status for quick troubleshooting purposes.

Error Handling

Invalid Data
If the signal receives data from a tag that is not recognised in LYNQ, LYNQ will treat the data as invalid. The tag value specified against the Signal must match the Tag value specified on the OPC Server. In addition to the Tag value, the value passed as a good quantity or scrap quantity value must be numeric.

Pending Accept
Check the Data Buffer Interval if you notice a number of transactions that have remained at the status of Pending Accept for some time. You may need to reduce the Data Buffer Interval if this is set to a high value and you wish to update LYNQ more frequently.

LYNQ Platform Service
The polling interval of the LYNQ Platform Service cannot be adjusted.

Accept Errors
These errors should be fixed to ensure the status changes to Trx Generated.

The Factory Automation Screen has 3 filters:

- In progress
- Errors
- Trx Generated

Relationship of filter to flow state value

Relationship	Flow State	Flow Status
	Invalid	Errors
	Pending Accept	In Progress
	Accepted	Trx Generated
	Accept Error	Errors
	Instant Accept	Trx Generated

Transactions displayed on the Factory Automation Screen with a status of **Error** should be investigated promptly.

Errors will happen if:

1. Data received includes invalid data (i.e. string value received instead of a numerical value for good quantity/scrap quantity).
2. No Job/Operation running at the time when the Quantity/Scrap processor created the transaction. When this happens, the data cannot be processed correctly, as LYNQ does not know which Job/Product to process the data against.
3. Task code cannot be located.
4. Internal data acceptance issues due to violation of key constraints.
5. Other types of error not listed above

You should also investigate transactions in the status of **In Progress,** if these transactions have been in this status longer than the data buffer interval.

In Progress will happen if:

1. The processor is still waiting to process the data based on the Data Buffer Interval Setting (Advanced Settings/General).

Some of these issues will resolve themselves, however there may be times when you need to resolve error data to allow the LYNQ transaction to be generated.

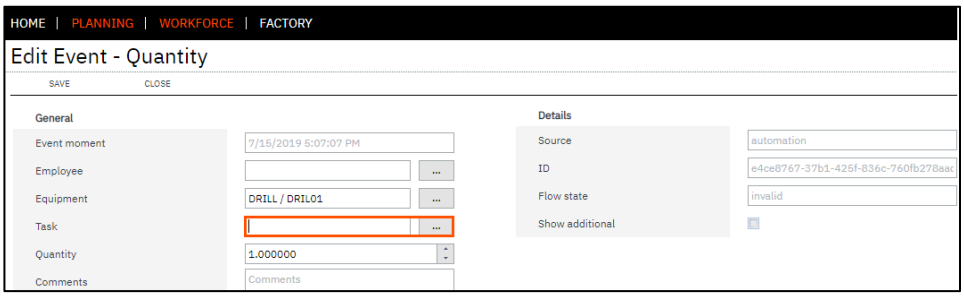
Correcting Invalid Errors

Bulk Edit
Use the Bulk Edit option in the Factory Automation screen to update multiple records at a time.

Transactions with a flow state of Invalid should be corrected to ensure data flow in LYNQ completes successfully. Corrections can only be applied to quantity, scrap or task status transactions. Resource status transactions cannot be corrected as correction requires changes of time-based calculations in the past. Invalid resource status transactions are shown for visualisation purposes only.

Invalid errors happen when the Factory Automation Processor cannot generate the final transaction for the data received from the automated device. This typically happens when the equipment was not running a Job at the time of the record creation.

Double clicking on the record will show a blank Task value.



To edit/correct these Transactions, complete these steps:

1. Select Factory Automation
2. Filter the screen to show the correct date range
3. Filter the screen to show only records where event type = Quantity
4. Filter the screen to show only records where Flow State = Error
5. Double Click on the Transaction that does not have a Job No
6. Using the Task Lookup [...] select a Job/Task
7. Select Save
8. Select the Checkbox in the Row Data (first column) and select Resolve

The Flow Status will change to Pending Accept and the next time the processor runs the transaction should update to a Flow Status of Trx Generated.

When a record in the Factory Automation screen is updated to Trx Generated the data will be visible in the standard LYNQ screens.

If you are experiencing high volumes of Invalid Errors see the following Alerts Section.

Correcting Accept Errors

Support Team
Support can be reached at support@lynqmes.com

Contact the LYNQ Support Team if you receive any transactions with a flow state of **Accept Error**. Accept Errors will occur if the transaction cannot be generated due to internal data integrity issues. LYNQ will investigate these issues with you to determine the root cause.

Alerts

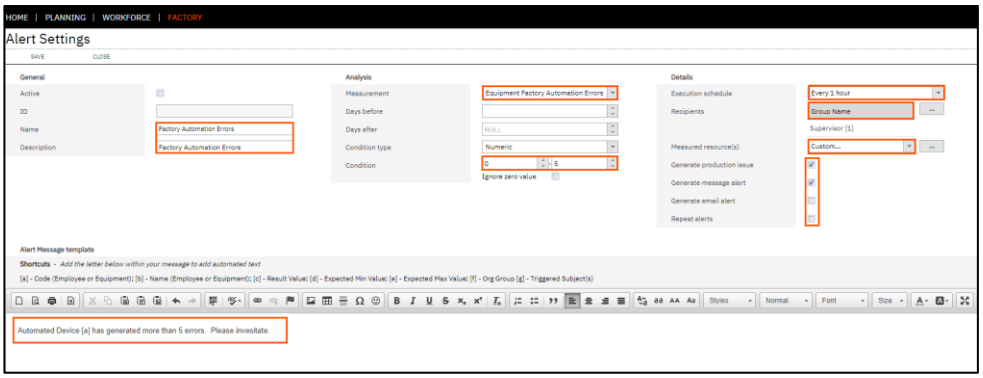
Alerts Setup
Search the Knowledgebase for articles relating to the setup of Alerts. You will need to ensure you have configured Alerts before using this function successfully.

LYNQ can be configured to alert you when errors are logged during the data flow process to help you promptly react to problems.

As an example, to configure an alert to notify recipients every hour when 5 or more errors have been logged:

1. Select Alert Maintenance
2. Select New
3. In the Name field enter Factory Automation Errors
4. In the Description field enter Factory Automation Errors
5. In the Measurement field select Equipment Factory Automation Errors
6. In the Condition field enter 0 and then 5
7. In the Execution Schedule field select Every 1 Hour
8. Select the Recipient Group for this alert
9. Select the Measured resources
10. Select whether the alert should create
    a. Product Issue
    b. Message Alert
    c. Email Alert
11. Select whether the Alert can be repeated.
12. Check the Active field

Alert Maintenance Settings



After 5 errors have been generated, the Alert will create a Production Issue and will send a message.

Troubleshooting

Troubleshooting
LYNQ can only process automated device data once the infrastructure is performing correctly.

The overall Factory Automation solution will have various points for failure. Investing in redundant network infrastructure will ensure higher availability of the Factory Automation solution.

It is important to understand when Factory Automation is not working, where to start troubleshooting. Troubleshooting can take place at the infrastructure layer and at the LYNQ application layer.

Use the table below as a guide for troubleshooting purposes

Issue	Troubleshooting Steps	Layer
No Data in Monitor	• Check that you have correctly configured the automated device seat in LYNQ. (i.e. Equipment has been imported, Equipment has been activated for Automation)	• LYNQ Application
	• Check there are no communication issues between the automated device/PLC/OPC Server or other underlying network issues	• Infrastructure
	• Check there are no communication issues between OPC server and the LYNQ Web Server	• Infrastructure
	• Check the Signals are activated in LYNQ with the correct Tag Values and Status Codes	• LYNQ Application
Data in Monitor but no Transactions in LYNQ	• Check the Processors are Enabled in LYNQ and that the correct seats have been associated to the Processor	• LYNQ Application
	• Check the Factory Automation screen to see if there are any flow status errors. Resolve these where appropriate	• LYNQ Application

	• Check System Insights
	  ○ Are there any error packets?
	  ○ Are there any invalid events?
	  ○ Are there any application general log errors? Review the log by clicking on [...]
		• LYNQ Application

	○ For DTC error messages check that the firewall on the SQL servers has an inbound rule to enable Distributed Transaction Coordinator and check the component services configuration on the SQL Server	• Operating System

LYNQ Ltd. 8 Barnes Wallis Road,
Fareham, Hampshire, UK PO15 5TU

LYNQ Your business. Understood.