

Release: March 2025 - R24D

Paint control standard package documentation

['Introduction'](#)

Reference Manual

['Installation'](#)

['Operation and safety'](#)

['Control system'](#)

['HMI system'](#)

['Robot system'](#)

['System functionalities'](#)



Paint Controls





ROBOTICS

Reference manual

PaintControls



Reference manual

PaintControls

3HNA035518-001 Rev.:1.5

Fifth edition: 11 March 2025

The information in this manual is subject to change without notice and should not be construed as a commitment by ABB. ABB assumes no responsibility for any errors that may appear in this manual.

Except as may be expressly stated anywhere in this manual, nothing herein shall be construed as any kind of guarantee or warranty by ABB for losses, damage to persons or property, fitness for a specific purpose or the like.

In no event shall ABB be liable for incidental or consequential damages arising from use of this manual and products described herein.

This manual and parts thereof must not be reproduced or copied without ABB's written permission.

Additional copies of this manual may be obtained from ABB.

The original language for this manual is English. Any other languages that are supplied have been translated from English.

Overview of this manual

Abbreviations

1 Introduction	14
2 Safety	16
2.1 Safety information	16
2.2 General hazards	16
2.3 Specific hazards	17
3 Installation	18
3.1 ABB Paint Controls system requirement	18
3.2 Time synchronization	19
3.2.1 Purpose and basic description	19
3.2.2 Synchronization structure	19
3.2.3 Windows OS setting	20
3.2.4 PLC setting	28
3.2.5 Robot setting	28
3.3 Robot system installation	29
3.3.1 Hardware description	29
3.3.2 Naming convention	30
3.3.3 Setup network	33
3.3.4 Install the Add-In to RobotStudio	34
3.3.5 Create a new system	35
3.3.6 Modify the existing system	38
3.3.7 Generic paint control Add-In options	38
3.3.8 Setup UAS	40
3.3.9 Safety configuration	43
3.3.10 Setting date and time	51
3.3.11 System parameter configuration	51
3.4 PLC system installation	64
3.4.1 Introduction	64
3.4.2 Requirement	64
3.5 HMI system installation	65
3.5.1 Introduction	65
3.5.2 Hardware requirement	66
3.5.3 Software components	66
3.5.4 Disk partitions	67
3.5.5 Windows 10 OS setting	67
3.5.6 Tool software	71
3.5.7 SQL Server Installation	71
3.5.8 IRC5P OPC Server Installation	77
3.5.9 RobotStudio Installation	80
3.5.10 RobView5 Installation	83
3.5.11 Redis Database	85
3.5.12 SQL Server Management Configuration	86
3.5.13 IRC5P OPC Configuration	90
3.5.14 Program preparation	91
3.5.15 Authorization description	91
4 Operation and safety	92
4.1 Introduction	92
4.2 Operation panel	92
4.2.1 Operation panel overview	92
4.2.2 Operation panel layout	93
4.2.3 Operation panel/Button description	93
4.3 Operation modes	101

Table of contents

4.3.1	Station mode	101
4.3.2	Station production mode	103
4.3.3	Robot operating mode	104
4.4	Station light indicator	105
4.5	Safety system	105
4.5.1	Safety layout	106
4.5.2	Safety matrix	106
4.5.3	Personal safety	111
4.5.4	Fire Protection	116
4.5.5	Cabin ventilation	117
5	Control system	118
5.1	Network structure	118
5.2	Robot Naming	120
5.2.1	Exterior station	120
5.2.2	Interior station	121
5.2.3	Sealing station	122
5.3	Type Verification	123
5.4	Interface	123
5.4.1	Interface description between PLC and Robot	124
5.4.2	Type and Vehicle Data	125
5.4.3	Conveyor Equipment	126
5.4.4	Process Equipment	127
5.4.5	Fire Protection	127
5.4.6	Central Control Room	127
5.4.7	Paint Supply	128
5.4.8	Special Colour Supply	130
5.4.9	PDA	131
5.5	Blow Off	132
5.6	3D Vision system	133
5.6.1	3D Camera system	133
5.6.2	BK Laser System	134
5.6.3	Toolbox3D	134
6	HMI system	136
6.1	HMI Overview	137
6.1.1	Screen layout	138
6.1.2	Header	139
6.1.3	Menu	141
6.1.4	Alarm Line	145
6.1.5	Main window	146
6.1.6	Additional window	147
6.1.7	Status Bar	148
6.2	View	153
6.2.1	Station view	153
6.2.2	Zone overview	159
6.2.3	Production status	159
6.2.4	Vision system status (VMT)	160
6.3	Robot	162
6.3.1	Positions	164
6.3.2	Application	169
6.3.3	Function - Painting	187
6.3.4	Function - PixelPaint	195
6.3.5	Function - Dispensing	199
6.3.6	Backup	200
6.3.7	Reset Program Pointer (PP)	204
6.3.8	Brush Editor	205
6.3.9	Refresh Log (command-IO)	208
6.3.10	Shaftlock - Bell (paint robot)	209

6.3.11 Interlock	209
6.4 Control	210
6.4.1 FIFO	211
6.4.2 Simulation	214
6.4.3 Bypass	216
6.4.4 Opener	217
6.5 Setting	218
6.5.1 Type	219
6.5.2 Colour (painting)	220
6.5.3 Skid slip sensor check (line tracking station)	221
6.5.4 Parameter	222
6.5.5 Alarm	223
6.5.6 Security	225
6.5.7 Miscellaneous	227
6.6 Diagnose	229
6.6.1 Alarm	229
6.6.2 WAN (Factory Network)	230
6.6.3 Bus	230
6.6.4 Mixroom	230
6.6.5 IO	231
6.7 Report	232
6.7.1 Production	233
6.7.2 Consumption	234
6.7.3 Operation	234
6.7.4 Alarm	235
6.7.5 Command log	236
6.8 Program	237
6.8.1 RobotStudio	237
6.8.2 RobView	237
6.8.3 Gateway RD	238
6.8.4 VMT RD	238
6.9 Help	239
6.9.1 RobotUI	239
6.9.2 Robot	239
6.9.3 Exit	239
7 Robot system	240
7.1 System description	240
7.1.1 Robot type	240
7.1.2 Application type	240
7.1.3 IO protocol type	241
7.1.4 Safety interface	241
7.2 Paint control IO Interface	241
7.2.1 I/O interface	241
7.2.2 Paint control IO (Paint, Pixel paint, Clean robot, Flaming robot, Dispense robot I/O signals)	242
7.3 Paint control command interface	258
7.3.1 Paint robot command	258
7.3.2 Pixel paint command	277
7.3.3 Clean robot command	282
7.3.4 Flaming robot command	283
7.3.5 Dispense robot command	283
7.4 Paint instructions	284
7.4.1 File structure	284
7.4.2 Program structure	296
7.4.3 User data	300
7.5 Paint control functions	305
7.5.1 General functions	305
7.5.2 Paint robot functions	312
7.5.3 PixelPaint Application functions	332

Table of contents

7.5.4	Clean robot functions	339
7.5.5	Flaming robot functions	340
7.5.6	Dispense robot functions	341
7.6	Paint control instructions.	342
7.6.1	General instructions	342
7.6.2	Paint robot system instructions.	356
7.6.3	Pixel Paint robot system instructions	373
7.6.4	Feather duster robot system instructions	376
7.6.5	Sword brush robot system instructions.	378
7.6.6	Paint checker robot system instructions.	378
7.6.7	Quality check instructions.	382
7.6.8	Dispense robot system instructions	385
7.7	Troubleshooting	385
7.7.1	General information	385
7.7.2	How to save diagnostics information	388
7.7.3	Troubleshooting with log file.	397
7.7.4	Troubleshooting with e-log	398
8	System functionalities	404
8.1	Program execution flow	404
8.1.1	Paint robot workflow	404
8.1.2	PixelPaint robot workflow	407
8.1.3	Clean robot workflow	407
8.1.4	Flaming robot workflow.	408
8.1.5	Dispense robot workflow	408
8.2	Robot mode	408
8.2.1	Job IO structure	409
8.2.2	Variable Car Type.	410
8.2.3	Degrade mode	411
8.2.4	Repair Mode.	412
8.2.5	Bypass mode	414
8.3	Interior concept.	416
8.3.1	Interior station data.	416
8.3.2	Stop&Go concept	417
8.3.3	Line tracking concept	417
8.4	Create a new program	419
8.4.1	Paint robot	419
8.4.2	Pixel paint robot	425
8.4.3	Clean robot.	425
8.4.4	Flaming robot	425
8.4.5	Dispense robot	429
Appendix		
Revision Log		

Overview of this manual

About this manual

This manual provides information and instructions for installation, configuration and usage of the HMI.

It also provides description of all functionalities included in the package in the HMI part.



WARNING! Before performing any work described in this manual, the Safety manual must be read and understood. Work must only be performed by skilled personnel with the proper training.

The Safety manual is included in the software and documentation DVD following each robot, and is also shipped with the robot as paper copy, included in the control cabinet.

Network security

This product is designed to be connected to and to communicate information and data via a network interface. It is your sole responsibility to provide and continuously ensure a secure connection between the product and to your network or any other network (as the case may be). You shall establish and maintain any appropriate measures (such as but not limited to the installation of firewalls, application of authentication measures, encryption of data, installation of anti-virus programs, and so on) to protect the product, the network, its system and the interface against any kind of security breaches, unauthorized access, interference, intrusion, leakage and/or theft of data or information.

ABB Ltd and its entities are not liable for damages and/or losses related to such security breaches, any unauthorized access, interference, intrusion, leakage and/or theft of data or information.

This page is intentionally left blank.

Abbreviations

1	BPP	Bits Per Pixel
2	CBS	Cartridge Bell System
3	CCV	Colour Changing Valve
4	COL	Colour
5	EMU	Electro Mechanical Unit
6	FIFO	First In, First Out
7	FOSI	Fibre Optic and Sensor Interface
8	GAP	Global Application Platform
9	HDC	Head Drive Card
10	HMI	Human Machine Interface
11	IDFP	Integrated Dispensing Function Package
12	IPS	Integrated Process System
13	MC	Material Change
14	MCB	Manipulator Control Board
15	MIB	Manipulator Interface Board
16	PS	Pressure Sensor
17	PLC	Programmable Logic Controller
18	PC	Personal Computer
19	PCC	Print Controller Card
20	PIB	Process Interface Board
21	POS	Position
22	PP Gateway	PixelPaint Gateway
23	PSIB	Pressure Sensor Interface Board
24	RCC	Rotating Colour Change
25	TBD	To Be Discussed
26	TCP	Tool Center Point
27	TPU	Teach Pendant Unit
28	VCBI	Voltage Control Board Interlock
29	VIN	Vehicle Identification Number Code
30	VAR	Variable Assembly Language

This page is intentionally left blank.

1 Introduction

About PaintControls	PaintControls is a standard Paint system function package. It can be used at Paint Robot Automotive and Tier1 Project. It contains robot functions, naming convention, job mode control, IO interface, communication and rapid instructions. It has a standard IO interface and a standard variable interface with PLC and HMI. The package supports both normal functions and safety functions. Both Stop&Go and conveyor tracing lines are supported. Some parameters and configuration files may be different according to different project specifications. A standard system commissioning checklist and commissioning steps can be used to guide system commissioning. Error handling and error logs can be used to support troubleshooting when there is a problem. Operators can follow the troubleshooting guidelines to fix the problem.
About this manual	This manual contains information about and instructions for: • Terms and concepts of PaintControls. • Installation steps. • Operation mode & safety. • IO interface. • Operate the system & instruction. • The functions description & Error information. • Description of the robot configuration, user files and RAPID instructions. • Description of the HMI functionalities.
Usage	This manual should be used during: • Installation. • Commissioning. • Programming work. • Maintenance work. • Repair work.
Who should read this manual?	This manual is intended for: • Installation personnel. • Robot programmers. • Maintenance personnel. • Repair personnel.

Prerequisites

The reader should :

- Be a trained installation engineer, programmer or operator.
- Have the required knowledge of operating the robot during installation, commissioning, production and repair work.
- Read the Safety Manual and follow directions at all times.

Organization of chapters

The manual is organized in the following chapters.

Number	Chapter	Description
1	Introduction	A short overview of the different sections contained in this manual.
2	Safety	Safety information must be needed before starting any work on the system.
3	Installation	Guidelines for system installation steps.
4	Operation and safety	Description of the available robot and station operation modes and the underlying safety setup.
5	Control system	Detailed description of the control system.
6	HMI system	A manual from the user's point of view describing the HMI functionalities.
7	Robot system	A manual from the user's point of view describing how to configure the robot and work with the robot files like application programs or brush tables.
8	System functionalities	A functional description of the PaintControls functions and how the concept is designed.

References

Following manuals are referred to in this manual:

Document ID	Manual
3HNA008924-001	Safety manual, IRC5P
3HNA026812-001	Application manual, Functional safety and SafeMove2 for paint
3HNA025117-001	Operating manual, IRC5P RW6.0
3HNA024494-001	Reference manual, paint commands and PLC interface IRC5P RW6.0
3HNA013917-001	Operator's manual, RobView5
3HAC032104-001	Operating manual - RobotStudio
3HAC020738-001	Operating manual - Troubleshooting IRC5P
3HNA025116-001	Installation manual - IRC5P RW6.0.

2 Safety

2.1 Safety information

Before performing any work described in this manual, it is extremely important that all safety information is observed!

There are general safety aspects that must be read through, as well as more specific safety information that describes danger and safety risks when performing the procedures. Read the Safety manual before performing any service work.



WARNING!

Before performing any work described in this manual, the Safety manual must be read and understood. Work must only be performed by skilled personnel with the proper training.



Important:

The description in this chapter is intended as a short reference and is no substitute for reading the complete Safety Manual.

The Safety manual is included in the ABB Standard user documentation IRB Paint robots and the IRC5P controller.

2.2 General hazards

The following lists some of the most relevant hazards.

- Before you enter the cell, make sure that all robots and other equipment have stopped, and also make sure that you wear appropriate safety clothes, antistatic shoes, gloves, safety glasses, etc. In case of danger from toxic gases, make sure that you wear appropriate safety equipment.
- Always attend to your personal safety.
- The motion of the manipulator is powerful. Always make sure that nobody is within the reach of the robot when running the robot for testing, etc.
- The manipulator is normally installed in an ex-area where there are risks of explosion. Always consider these risks when working with the robot system and bringing tools and equipment into the area.
- Always be aware of hazards associated with electric power when working with the robot system.
- The use of thin layers of non-conductive plastics for covering parts of the manipulator against exposure of coating materials must be avoided, as it could present a hazard of propagating brush discharges.
- Releasing the robot axis brakes can be potentially dangerous. Never release the brakes unless you know the risks involved.

- Electronic units are sensitive to static electricity. It is extremely important that these units are handled with care in order not to damage the boards or introduce errors.
- Do not disconnect any connectors, except service ports, while power is applied to the controller.

2.3 Specific hazards

The following lists some hazards specific to the operating activities described in this manual.

- Always pay attention to the network security.
- The operating mode selector on the controller or control desk must be in the manual reduced speed mode position () or Manual high-speed mode () to render the enabling device operative and block operation from a computer link or remote control panel. The key must be brought to the working site to prevent another person from accidentally changing the operating mode of the robot.
- Before starting any kind of operation of the robot, always pay attention to hazards related to the specific robot. These hazards are described in chapter ‘Safety’ in the Product manual for the specific robot.

3 Installation

About this chapter

This chapter provides information about the requirements and how to install the packages on robot, PLC and HMI.

The following sections are included:

['ABB Paint Controls system requirement'](#) on page 18

['Time synchronization'](#) on page 19

['Robot system installation'](#) on page 29

['PLC system installation'](#) on page 64

['HMI system installation'](#) on page 65

3.1 ABB Paint Controls system requirement

ABB software requirements

Software product	Software version
RobotWare	6.15 or higher
IPS	4.60.36 or higher
PAP package	4.0.408 or higher
PaintControl package	Project related
RobotStudio	2023 or higher
RobView5	5.2.XXXX
ABB IRC5P OPC server	6.10 or higher.

PC hardware requirements

Hardware	Minimum Requirements	Recommended Requirements
CPU	2.0 GHz or faster processor	2.4 GHz or faster processor
Memory	4 GB	16 GB
Operation system	Window 10 or 11.	Window 10 or 11.

3 Installation

3.2 Time synchronization

3.2 Time synchronization

The following sub-sections are included in this section:

'Purpose and basic description' on page 19

'Synchronization structure' on page 19

'Windows OS setting' on page 20

'PLC setting' on page 28

'Robot setting' on page 28

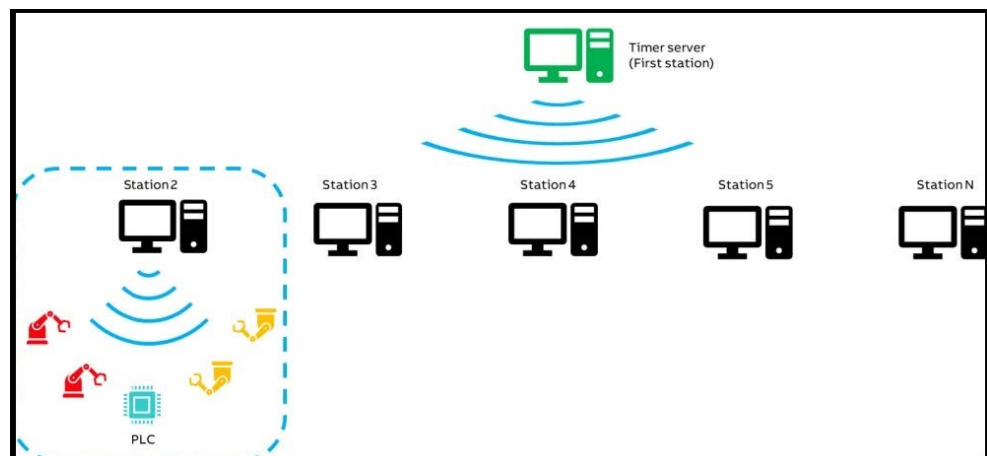
3.2.1 Purpose and basic description

Overview

The purpose of this document is to provide a comprehensive guide on the implementation of a time synchronization service between HMI computer systems, Programmable Logic Controllers (PLCs) and robots.

Achieving precise time synchronization among these devices is crucial for commissioning and issue traceback.

3.2.2 Synchronization structure



- If the customer does not provide a dedicated time synchronization server, it is recommended to designate a computer within the local area network (LAN) as the time synchronization server.
- This internal server will act as the centralized timekeeping reference for all other computers within the network.
- The selection of this server should consider factors such as stability, reliability, and accessibility to ensure accurate time synchronization across the network.
- Generally, choose the first computer in the line as the server.
- Alternatively, when the host supplies the time synchronization server, all subscribed computers synchronize their time with this central server.

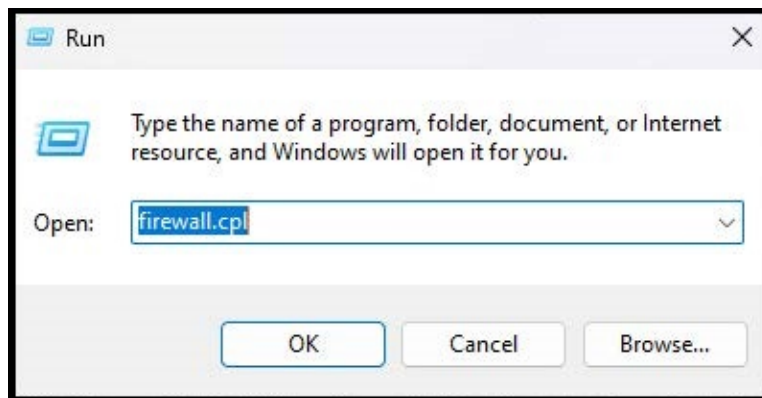
- Each client computer, acting as a server within its station, facilitates time synchronization for sub-devices such as PLC CPUs and Robots.
- In this setup, the HMI computer serves as the time source for all connected devices, ensuring a unified and accurate timekeeping system across the network.

3.2.3 Windows OS setting

3.2.3.1 System Setting

Add Firewall Inbound Rule

1. Open Windows Defender Firewall:
 - a. Press Win + R to open the Run dialog.
 - b. Type control firewall.cpl and press Enter to open the Windows Defender Firewall settings.
2. Access Advanced Settings:

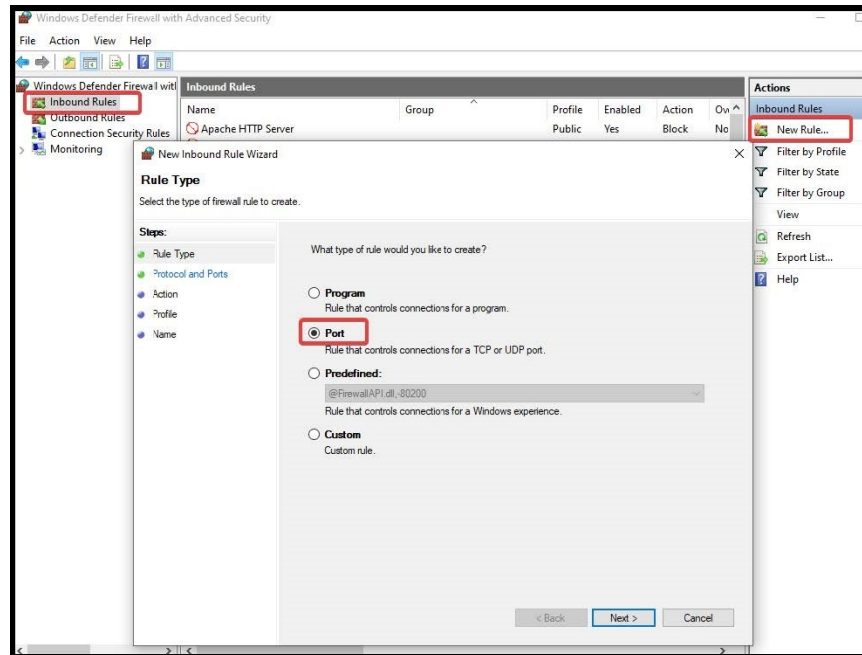


- a. In the left pane, click on "Advanced settings." This will open the Windows Defender Firewall with Advanced Security.
3. Create a New Inbound Rule:
 - a. In the left pane, select "Inbound Rules."
 - b. Right-click and choose "New Rule..." from the context menu.

3 Installation

3.2 Time synchronization

4. Choose Rule Type:

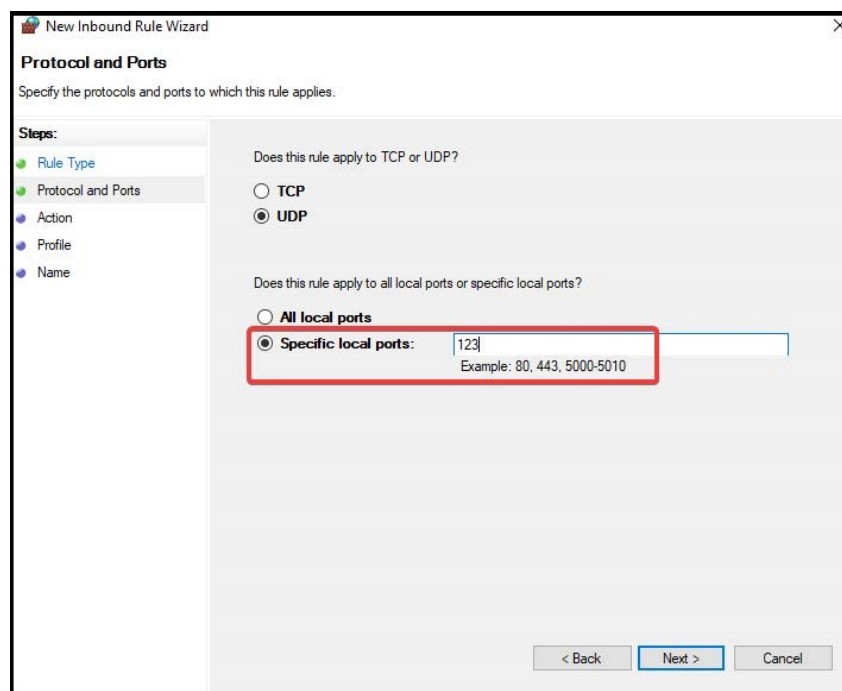


a. In the Rule Wizard, select "Custom" and click "Next."

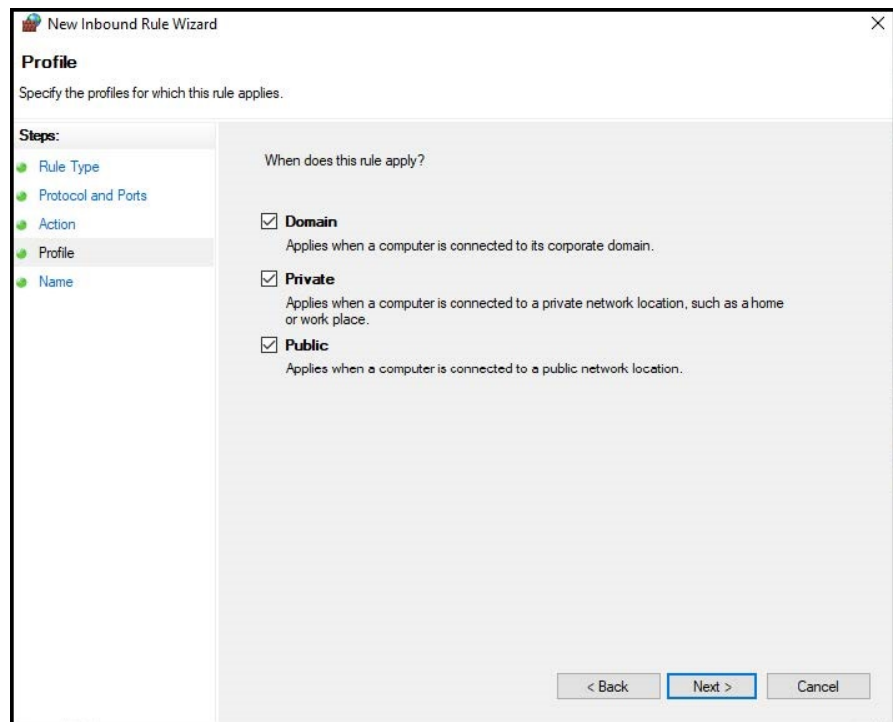
5. Choose Program and Protocol:

a. Leave "This program path" as it is and click "Next."

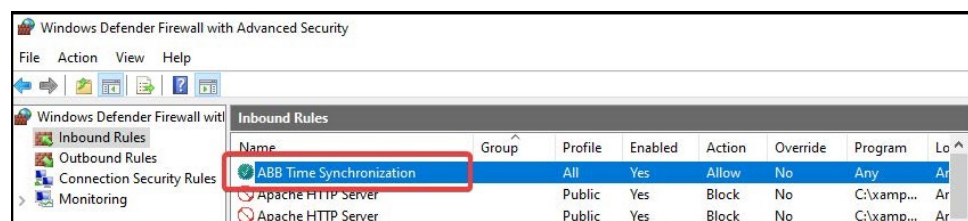
b. Select "This program uses the following protocol and ports" and choose "UDP." Enter "123" in the "Specific local ports" field.



6. Choose Action:
 - a. Select "Allow the connection" and click "Next."
7. Choose Profiles:
 - a. Leave all profiles checked and click "Next."



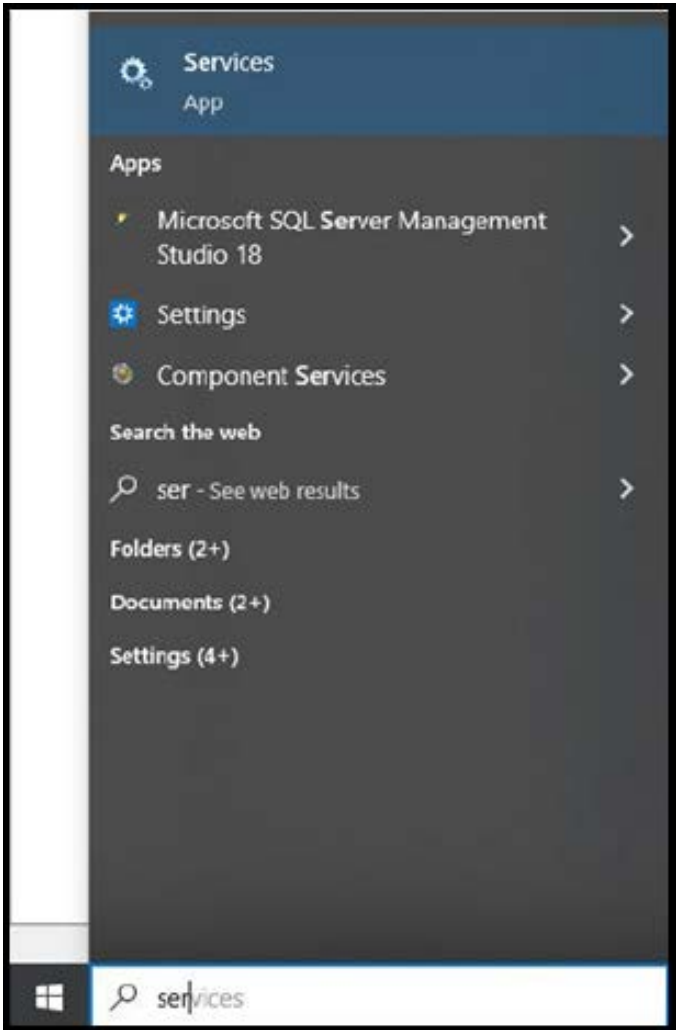
8. Specify Name and Description:
 - a. Enter a name "ABB Time Synchronization", with description "For ABB devices time synchroni-zation".
 - b. Click "Finish" to create the rule.



3.2 Time synchronization

Disable Windows Time Service

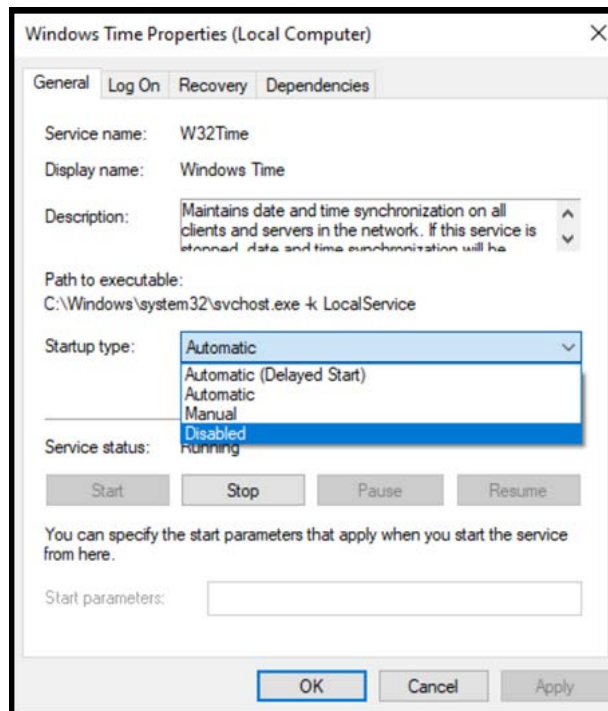
1. Open the Services.



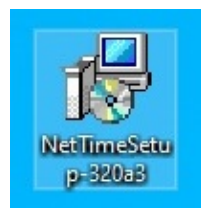
2. Double click on the Windows Time service.

	Windows Remote Manage...	Windows R...	Manual	Network S...
	Windows Search	Provides co...	Running	Automatic (...)
	Windows Security Service	Windows Se...	Running	Manual
	Windows Time	Maintains d...	Starting	Automatic (T...
	Windows Update	Enables the ...	Manual (Trig...	Local Syste...
	Windows Update Medic Ser...	Enables rem...	Manual	Local Syste...

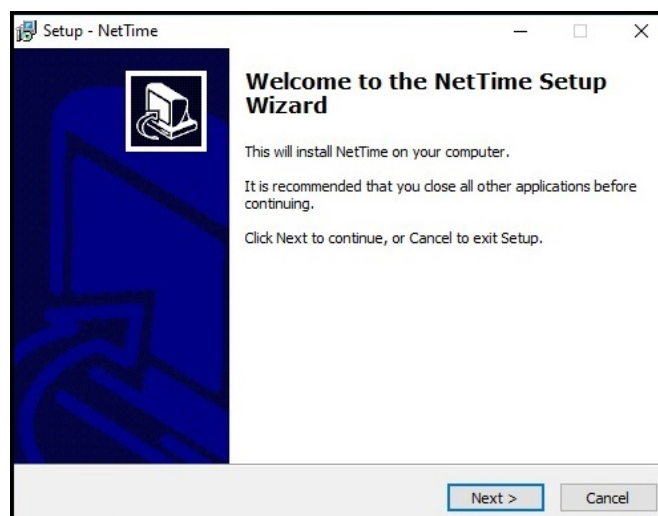
3. Select Disabled from the drop-down list, click on Stop and Apply button to Disable Windows Time service.



Install NetTime software



Always click “Next” button until finish installation.



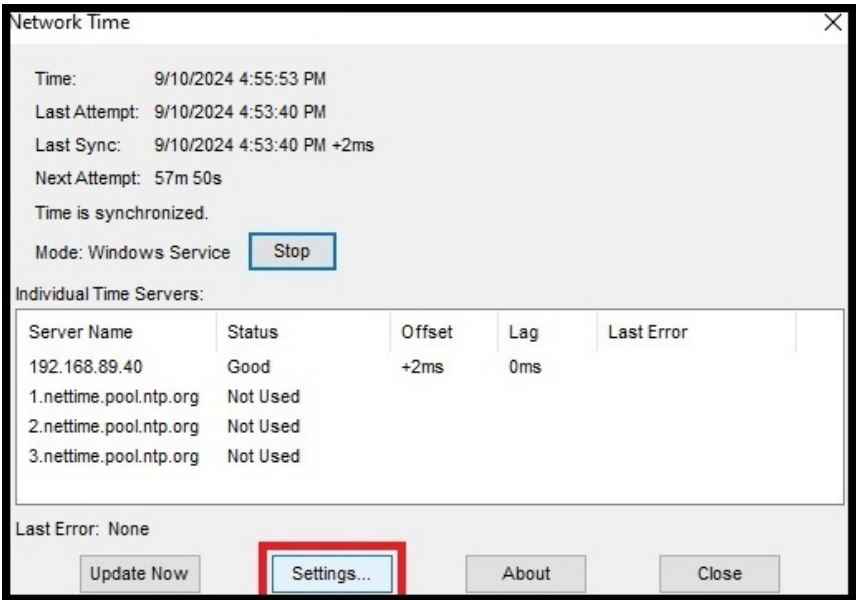
NetTime software will automatically start after windows system start.

3 Installation

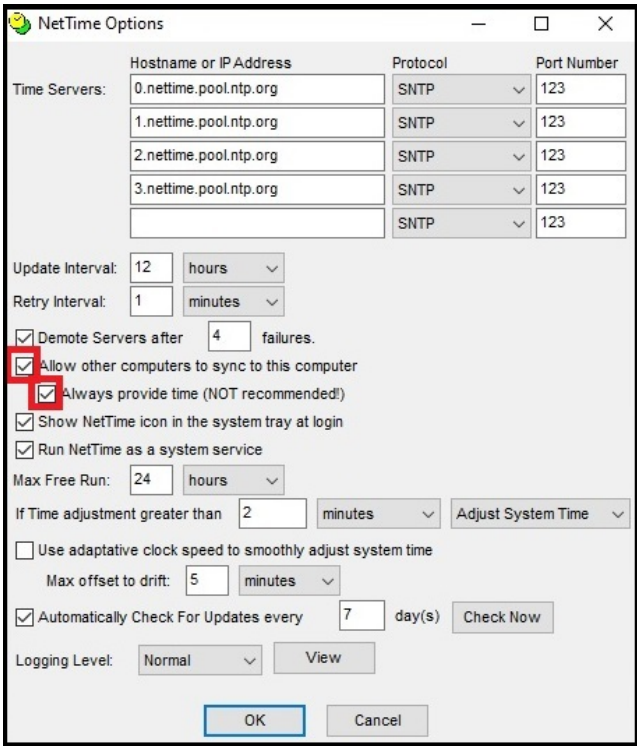
3.2 Time synchronization

Set time server

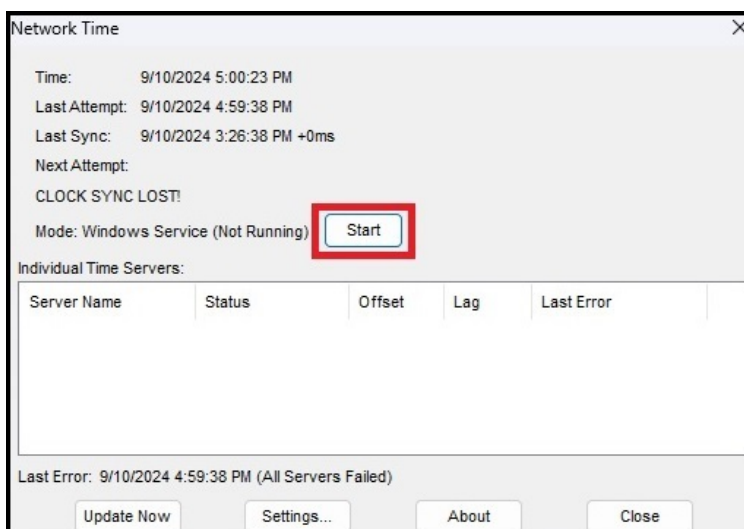
Click ‘Settings’ button.



Select ‘Allow other computers to sync to this computer’ and ‘Always provide time (NOT recommended)’. Click ‘OK’ button.

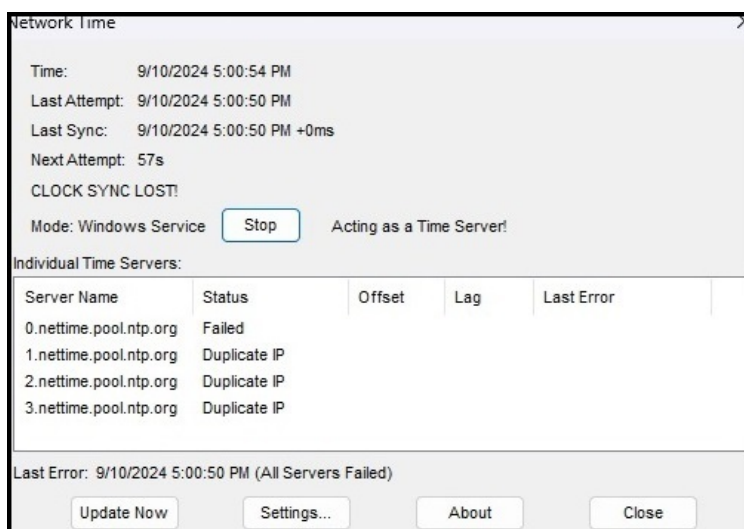


Click ‘Start’ button to start time service.



Set time client

Click 'Settings' button.



Input time server IP address and set update interval as 1 hour. Click 'OK' button.

3 Installation

3.2 Time synchronization

NetTime Options

Time Servers:	Hostname or IP Address	Protocol	Port Number
	192.168.89.40	SNTP	123
	1.nettime.pool.ntp.org	SNTP	123
	2.nettime.pool.ntp.org	SNTP	123
	3.nettime.pool.ntp.org	SNTP	123
		SNTP	123

Update Interval: 1 hours

Retry Interval: 1 minutes

☒ Demote Servers after 4 failures.

☐ Allow other computers to sync to this computer

☒ Always provide time (NOT recommended!)

☒ Show NetTime icon in the system tray at login

☒ Run NetTime as a system service

Max Free Run: 24 hours

If Time adjustment greater than 2 minutes Adjust System Time

☐ Use adaptative clock speed to smoothly adjust system time

Max offset to drift: 5 minutes

☒ Automatically Check For Updates every 7 day(s) Check Now

Logging Level: Normal View

OK Cancel

Click 'Update Now' button to check synchronization result.

Network Time

Time: 9/10/2024 4:59:05 PM

Last Attempt: 9/10/2024 4:59:00 PM

Last Sync: 9/10/2024 4:59:00 PM +1ms

Next Attempt: 59m 56s

Time is synchronized.

Mode: Windows Service Stop

Server Name	Status	Offset	Lag	Last Error
192.168.89.40	Good	+1ms	0ms	
1.nettime.pool.ntp.org	Not Used			
2.nettime.pool.ntp.org	Not Used			
3.nettime.pool.ntp.org	Not Used			

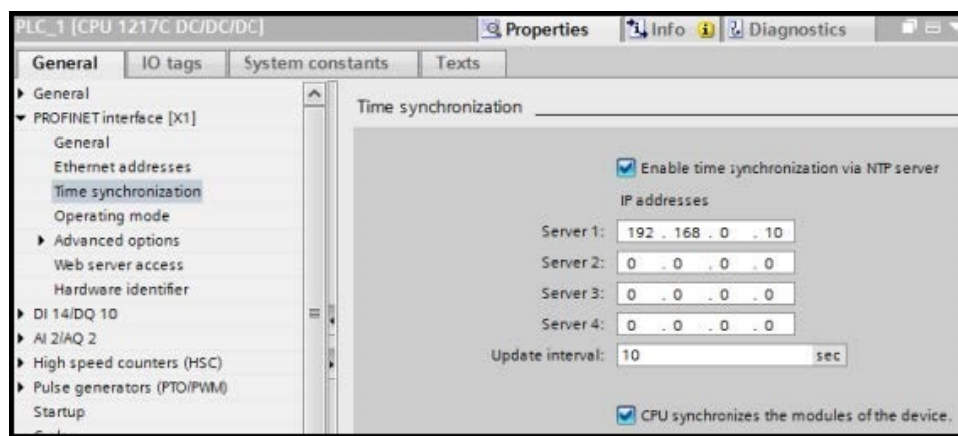
Last Error: 9/10/2024 4:58:07 PM (All Servers Failed)

Update Now Settings... About Close

3.2.4 PLC setting

3.2.4.1 Siemens S7-1500 Series

Locate and select the CPU module within the hardware configuration of your project.



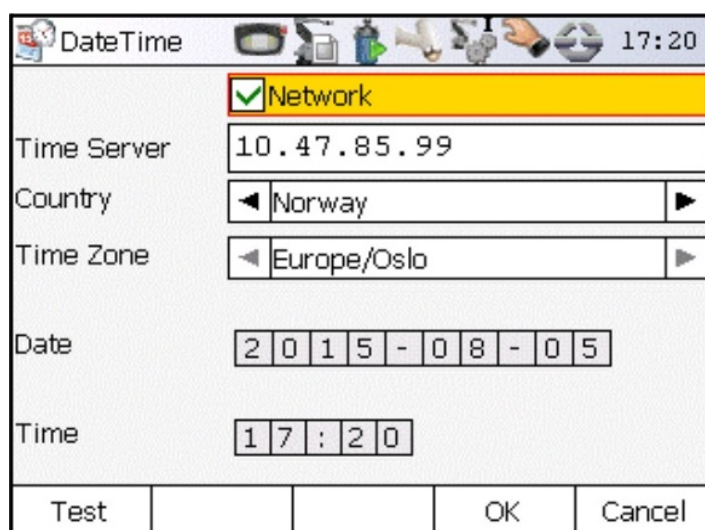
Activate the Enable time synchronization via NTP server, and input the IP address of current station computer, set the Update Interval to 10 seconds.

3.2.5 Robot setting

In each Robot teach pendant, set the IP address of current HMI computer, and enable the Net- work selection.

3.2.5.1 Teach Pendant Setting

In Service -> Data Time module, input current station HMI computer IP address, set the location and time zone accordingly.



Note: There is setting in Robot Studio, but seems it's not stable, as of the time of writing this document, it is recommended utilizing this approach by teach pendant.

3 Installation

3.3 Robot system installation

3.3 Robot system installation

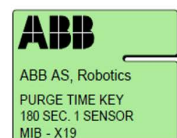
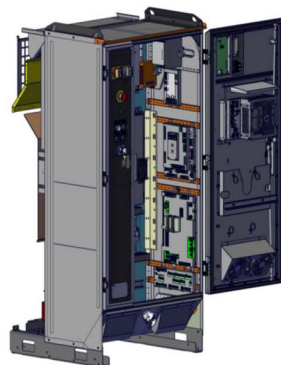
The following sub-sections are included in this section:

- 'Hardware description' on page 29
- 'Naming convention' on page 30
- 'Setup network' on page 33
- 'Install the Add-In to RobotStudio' on page 34
- 'Create a new system' on page 35
- 'Modify the existing system' on page 38
- 'Generic paint control Add-In options' on page 38
- 'Setup UAS' on page 40
- 'Safety configuration' on page 43
- 'Setting date and time' on page 51
- 'System parameter configuration' on page 51

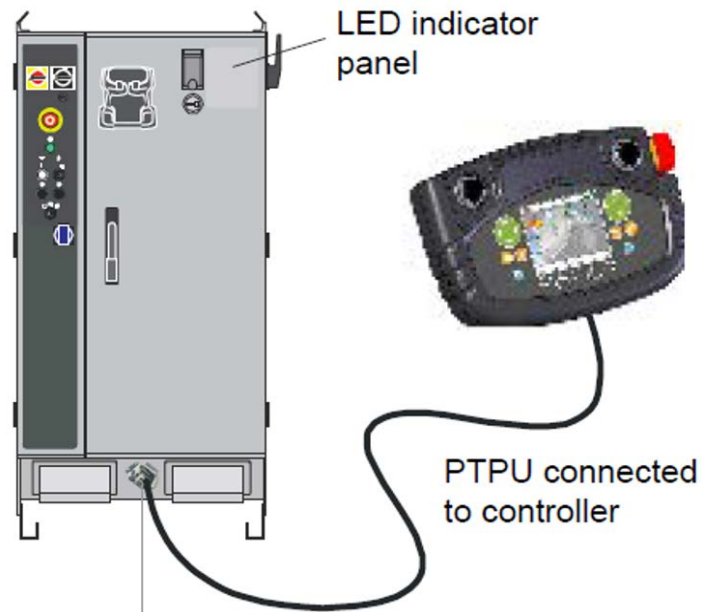
3.3.1 Hardware description

Steps

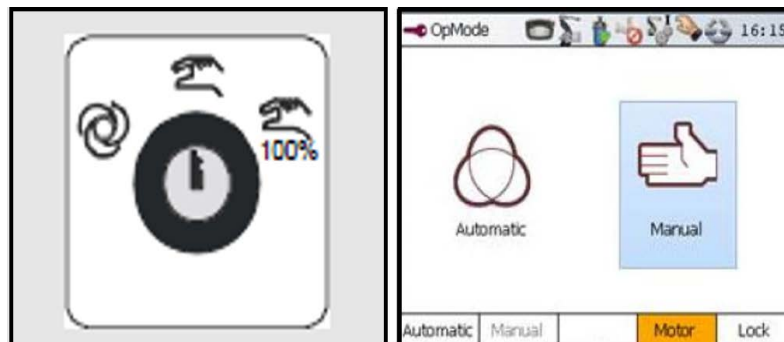
1. If the air supply is not ready during installation, install the non-ex purge time key to start up the robot controller.



2. The TPU should be connected with the robot controller.



3. For the remote panel with SCB-01, the X8 connection should be in Manual or Auto mode.



3.3.2 Naming convention

The following sub-sections are included in this section:

'Station name' on page 30

'Robot name' on page 32

'Installation package name' on page 33

3.3.2.1 Station name

Robot station naming convention:

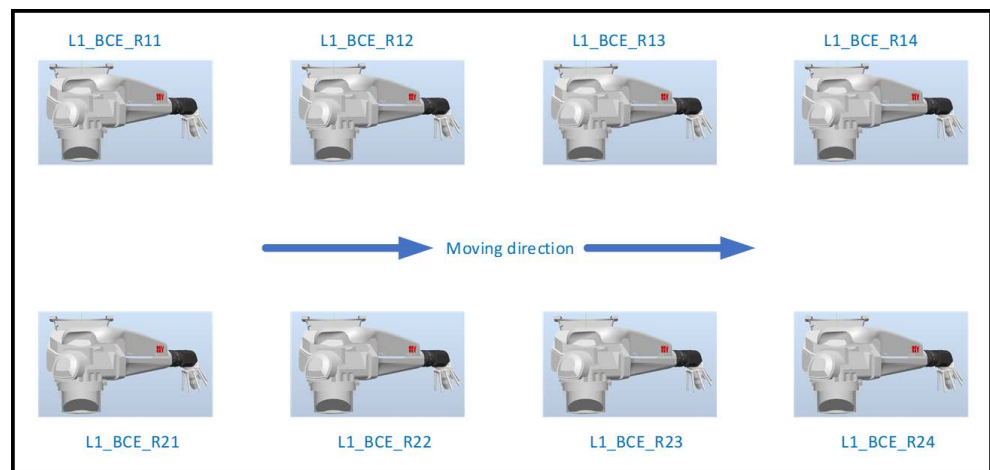
Station area	Station name
Primer Exterior	PRE
Primer Interior	PRI
Base Coat Exterior	BCE

3 Installation

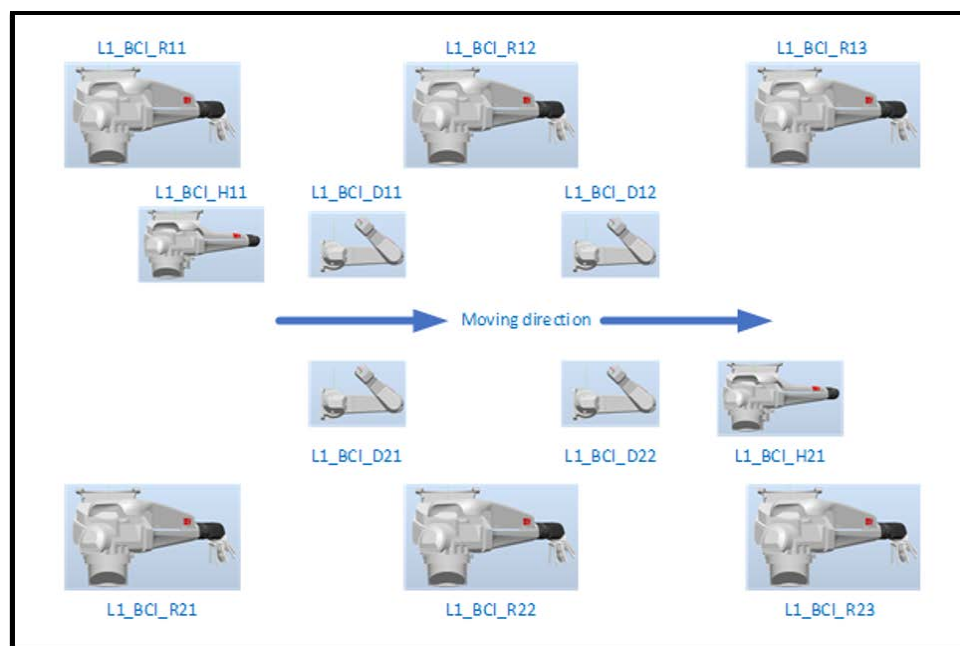
3.3 Robot system installation

Station area	Station name
Base Coat Interior	BCI
Clear Coat Exterior	CCE
Clear Coat Interior	CCI
Feather Duster	FD
Sword Brush	SW
Paint Check	PC
Flaming	FL
PixelPaint	PxP
Dispenser	UBC, UBS, ISS, FAD...
TBD	TBD

Exterior station layout overview (BC exterior station)



Interior station layout overview (BC interior station)



3.3.2.2 Robot name

There are three criteria for robot name, like “L1_BCI_R11”, “L1” means line 1 and “BCI” means base coat interior station. “R11” means the first paint robot on left side.

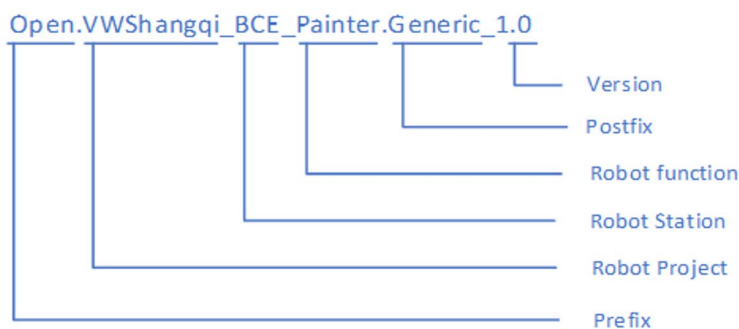
Robot side of conveyor	Robot index from entry to exit	Paint robot	Door Opener	Hood/Trunk
Left	1	R11	D11	H11
	2	R12	D12	H12
	3	R13	D13	H13
	4	R14	D14	H14
	X	R1X	D1X	H1X
Right	1	R21	D21	H21
	2	R22	D22	H22
	3	R23	D23	H23
	4	R24	D24	H24
	5	R2X	D2X	H2X

If customers have other requirements for robot naming, the project can be defined by them.

3 Installation

3.3 Robot system installation

3.3.2.3 Installation package name



Example:

The project name is VW Shanghai.

Station	Robot function	PaintControl Package name
Primer Interior	Painter	Open.VWShangqi_PRI_Painter.Generic_1.0
Primer Interior	Door opener	Open.VWShangqi_PRI_Door.Generic_1.0
Primer Interior	Hood or trunk opener	Open.VWShangqi_PRI_HoodTrunk.Generic_1.0
Primer exterior	Painter	Open.VWShangqi_PRE_Painter.Generic_1.0
BC Interior	Painter	Open.VWShangqi_BCI_Painter.Generic_1.0
BC Interior	Door opener	Open.VWShangqi_BCI_Door.Generic_1.0
BC Interior	Hood or trunk opener	Open.VWShangqi_BCI_HoodTrunk.Generic_1.0
BC exterior	Painter	Open.VWShangqi_BCE_Painter.Generic_1.0
CC Interior	Painter	Open.VWShangqi_CCI_Painter.Generic_1.0
CC Interior	Door opener	Open.VWShangqi_CCI_Door.Generic_1.0
CC Interior	Hood or trunk opener	Open.VWShangqi_CCI_HoodTrunk.Generic_1.0
CC exterior	Painter	Open.VWShangqi_CCE_Painter.Generic_1.0

3.3.3 Setup network

The following sub-sections are included in this section:

['Network settings, MC'](#) on page 33

['Network settings, PIB'](#) on page 33

3.3.3.1 Network settings, MC

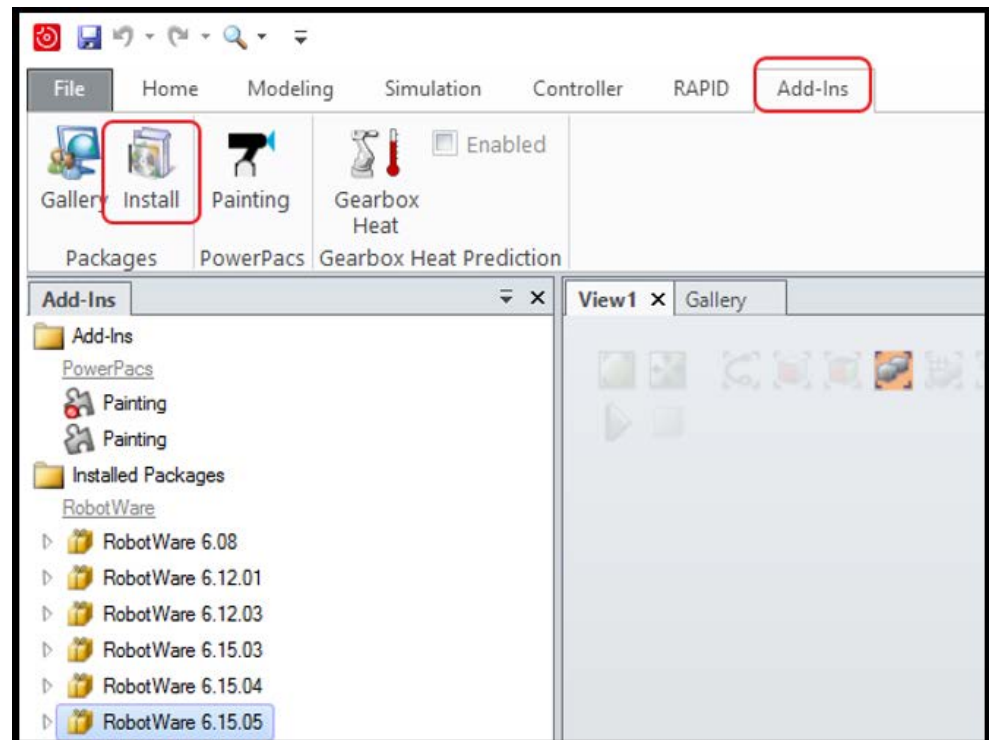
For more information about Network (WAN) setting, see *Bootloader*, in *Installation manual - IRC5P RW6.0* listed in ['References'](#) on page 15.

3.3.3.2 Network settings, PIB

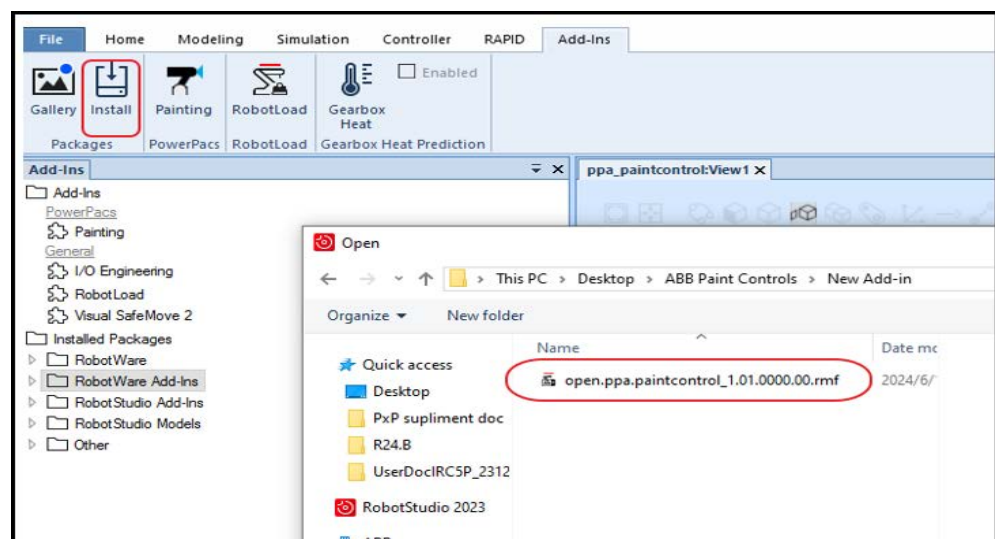
For more information about Network settings, see *PIB*, in *Installation manual - IRC5P RW6.0* listed in ['References'](#) on page 15.

3.3.4 Install the Add-In to RobotStudio

- Open the **RobotStudio** > go to the menu “**Add-Ins tab**” and click “**Install**”.



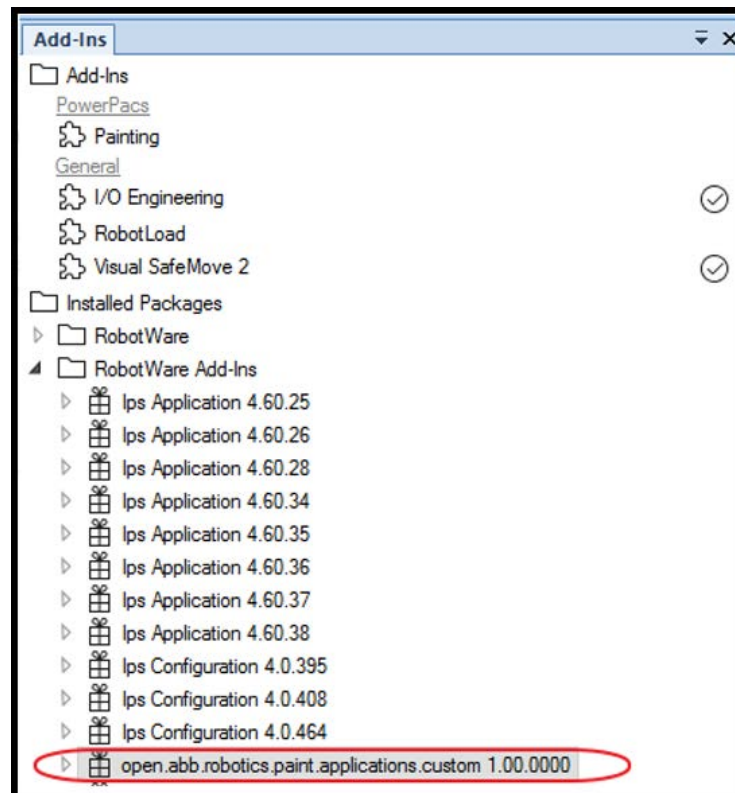
- Select the package that needs to be installed and click “**Open**”. The package will be installed in RobotStudio.



3 Installation

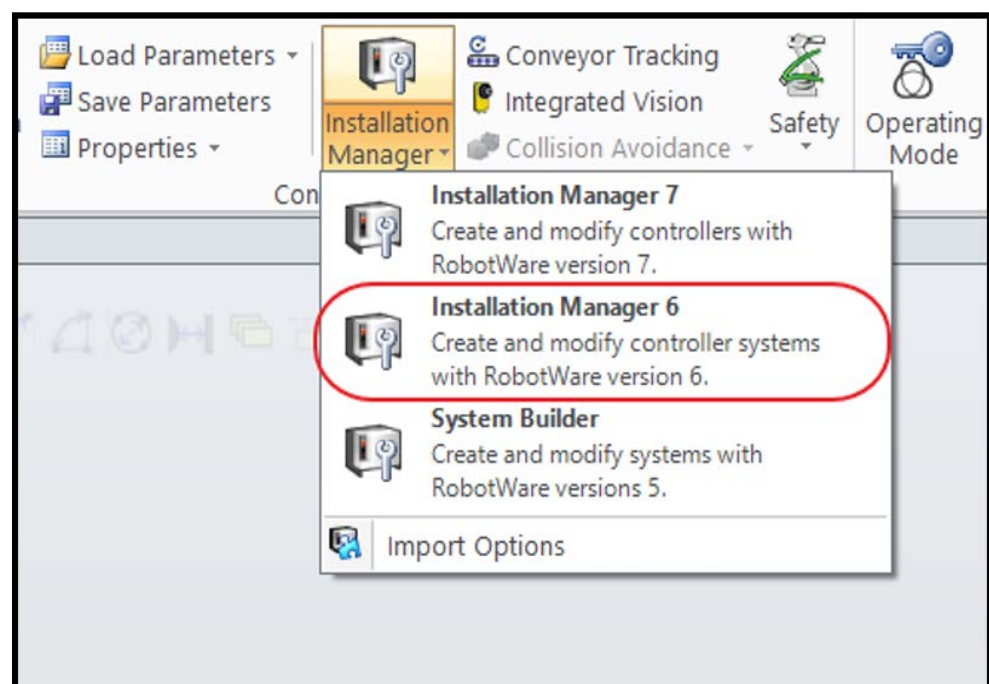
3.3 Robot system installation

- After the installation, the package can be found in the left “**Add-Ins**” window.



3.3.5 Create a new system

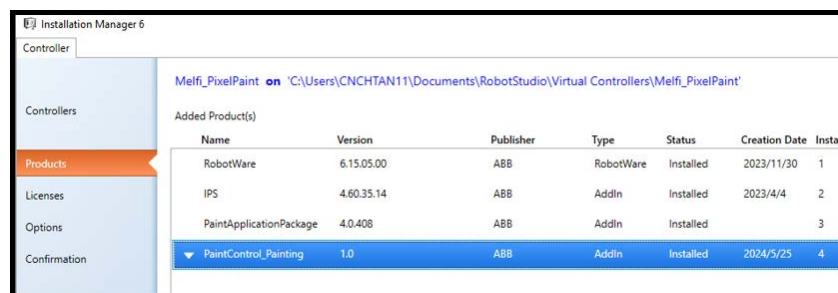
- Build a connection with the controller > start **RobotStudio** > go to the **Controller** tab and click “**Installation Manager**”.



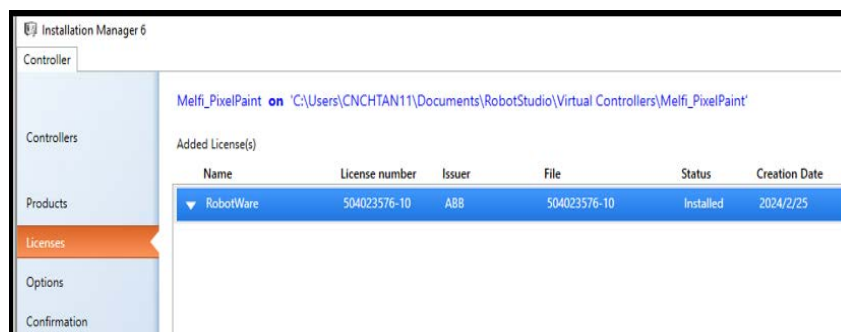
- Click network to select the **robot IP**.



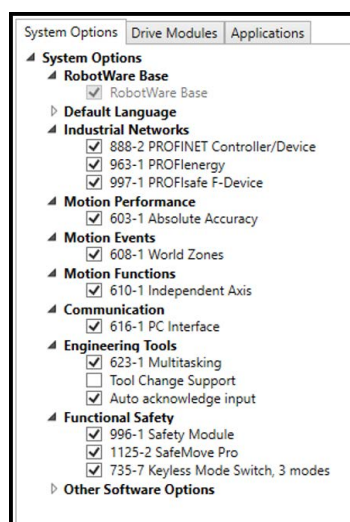
- Click **New** and input the system name.
- Add or replace the **Robotware, IPS**, and the system add-in package.



- Choose the **Robot** license file.



- Choose the options of the **Robot**.



3 Installation

3.3 Robot system installation

- Select options in “Applications”.

The screenshot shows the 'Applications' tab in the installation manager. It is divided into two panes. The left pane shows the 'Paint Control' section with options for 'Painting', 'General Option', 'Station', 'Robot Function', 'Move Home', 'Smart Home', 'Manipulator', and 'Application'. The right pane shows the 'ZKP' section with options for 'Applicator', 'Atomzier', 'Gun', 'Opener Tool', 'Pixel Paint', 'Applicator Clean', 'Bell Clean', 'Safety', 'PaintOptions', 'Application Test', 'HV Safety Test', 'Application Functions', 'Paint Motion Data', 'Vision', '3D Vision', and 'Line Runner'. The 'Pixel Paint' option is selected in the right pane.

System Options	Drive Modules	Applications
Paint Control		
Painting		
General Option		
Station		
Interior		
Exterior		
Robot Function		
Painter		
Opener		
PainterWithOpener		
Move Home		
Normal Home		
Smart Home		
EscapePoint		
Manipulator		
IRB5500		
IRB5350		
Rail		
Application		
Application Type		
1KP		
1KP CCV 1cavity		
1KP CCV 2cavity		
1KP RCC 1cavity		
1KP RCC 2cavity		
1KI PXP		
1KP CBS CCV		
2KP		
Applicator		
Atomzier		
RB1000i		
RB1000i-S		
Gun		

System Options	Drive Modules	Applications
ZKP		
Applicator		
Atomzier		
RB1000i		
RB1000i-S		
Gun		
Opener Tool		
Pixel Paint		
PXP PH		
Applicator Clean		
HubClean		
GunClean		
DumpClean		
PixelPaintClean		
ToolClean		
Bell Clean		
Safety		
ProfiSafe		
PaintOptions		
Material Change		
Application Supervision		
Solvent Supervision		
Material Supervision		
Shaft Supervision		
Application Test		
Non Motion Execution		
HV Safety Test		
Application Functions		
Paint Motion Data		
Vision		
3D Vision		
VMT 3D		
Line Runner		
VMT LineRunner		

- Confirmation and download to **Controller**.

The screenshot shows the 'Confirmation' window in the installation manager. It displays the configuration for the 'Melfi_PixelPaint' application. The left pane shows the 'Confirmation' tab selected. The right pane shows the configuration details for the application, including the 'Mainipulator', 'Application', 'Applicator', 'Applicator Clean', 'Safety', 'PaintOptions', 'Application Test', and 'Vision' sections. The 'Non Motion Execution' option is highlighted in blue.

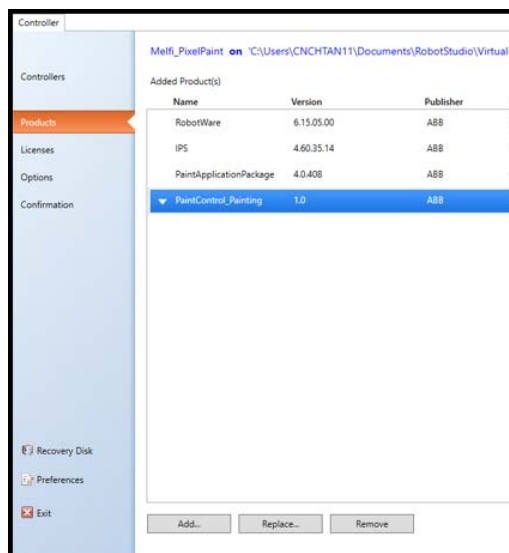
Controller
Melfi_PixelPaint on 'C:\Users\CNCHTAN11\Documents\RobotStudio\Virtual Controllers\Melfi_PixelPaint'

Controllers
Products
Licenses
Options
Confirmation

Confirmation
EscapePoint
Mainipulator
IRB5500
Application
Application Type
1KP
1KI PXP
Applicator
Pixel Paint
PXP PH
Applicator Clean
PixelPaintClean
Safety
ProfiSafe
PaintOptions
Application Test
Non Motion Execution
Vision
3D Vision
VMT 3D
Line Runner
VMT LineRunner

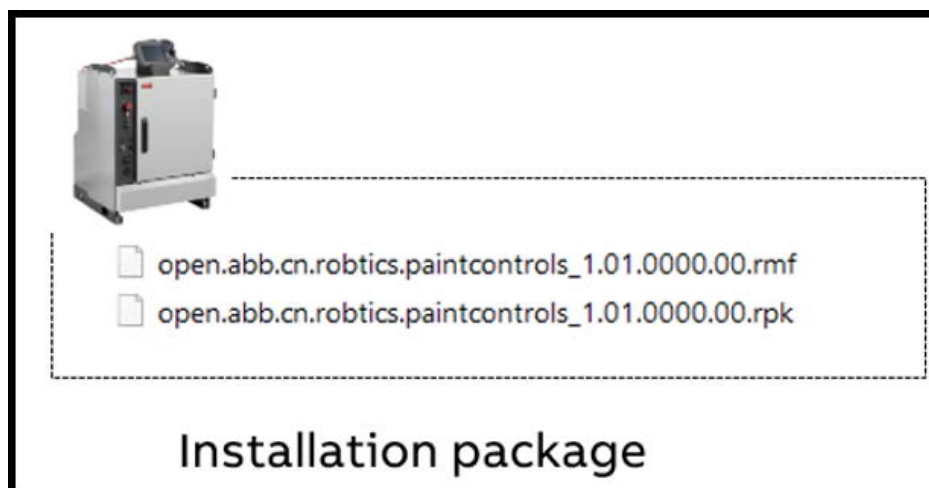
3.3.6 Modify the existing system

- Setup computer network > build a connection with the controller > start **RobotStudio** > go to the **Controller** tab > click **“Installation Manager”** > click network to select the robot IP > click the **“Next”** and replace the correct **Robotware, IPS or PaintControl** package or options if these items.



- Select **Options** and download to **Controller**.

3.3.7 Generic paint control Add-In options



The generic paint control package is a package which includes most of the standard options for paint applications.

3 Installation

3.3 Robot system installation

General Configuration:

Product	Version
Station Type	Interior, Exterior
Line type	Stop&Go, Line tracking
Robot type	IRB5500, IRB5350
Atomizer	RB1000i GAP type
Color changer	MPAC, RCC
Application	1K, 2K, CBS, PxP
Cleaner Type	CCC, HD

Communication and SafeMove:

BUS type	All ABB Standard BUS type
IO Size	64 bytes
Command	Full command
SafeMove	SafeMove 2

GAP Solution:

Product	Support
GAP 1K	Yes
GAP 2K	Yes
GAP CBS	Yes
GAP PXP	Yes

Applicator Type:

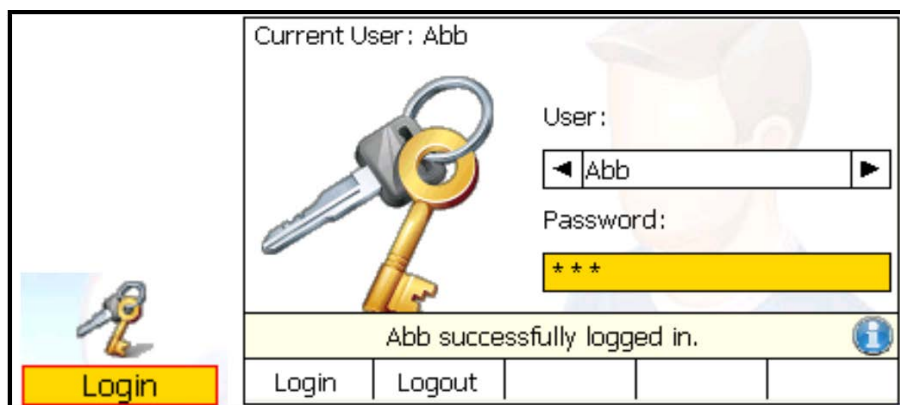
Product	Support
RB1000i SAD	Yes
RB1000i SAD NO HV	Yes
RB1000i SSD	Yes
RB1000i EXT	Yes
RB1000i S SSD	Yes
AGMD Gun	Yes
Opener	Yes
Pixel painter	Yes

3.3.8 Setup UAS

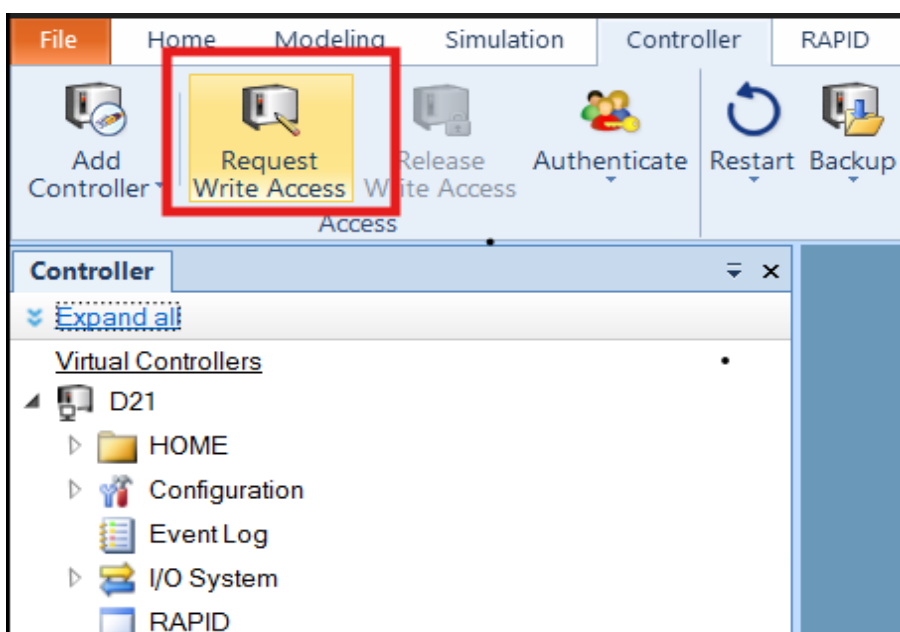
All the setup work is done in RobotStudio, like setting up user accounts. See 'Operating manual, RobotStudio', Working online/User authorization.

For more information on Login/logout on the TPU, see 'Operating manual, IRC5P RW6.0', Basic operation/User authorization.

- Go to the Login menu from the main menu on the TPU.



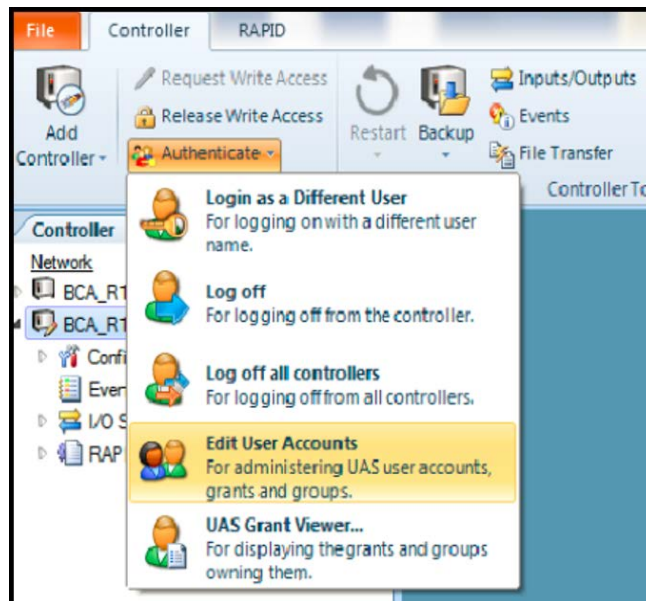
- Select your user name, type the Password and click “Login”.
- Logout the current user account and login the Default account as “Default User”, Password is “robotics”. You will see the access level change according to the user privileges.
- Start RobotStudio.
- Connect to **Controller** and **Request Write Access**.



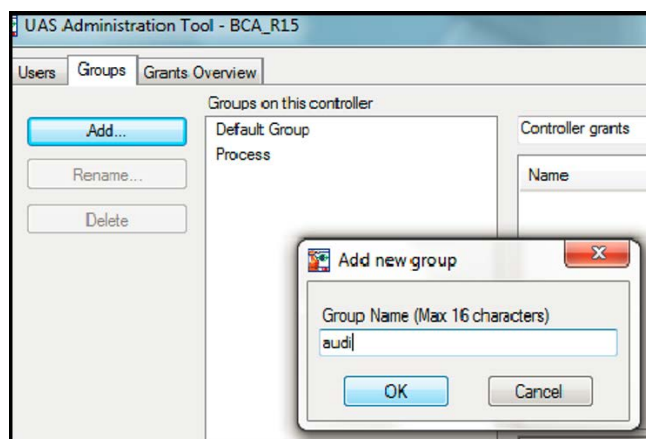
3 Installation

3.3 Robot system installation

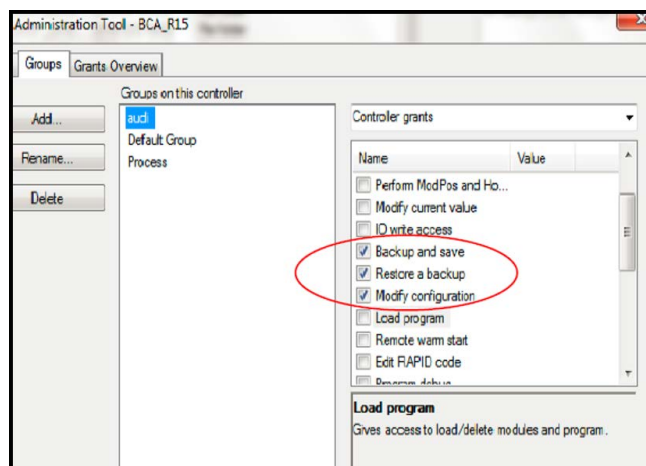
- Select **Authenticate** and “**Edit User Accounts**”.



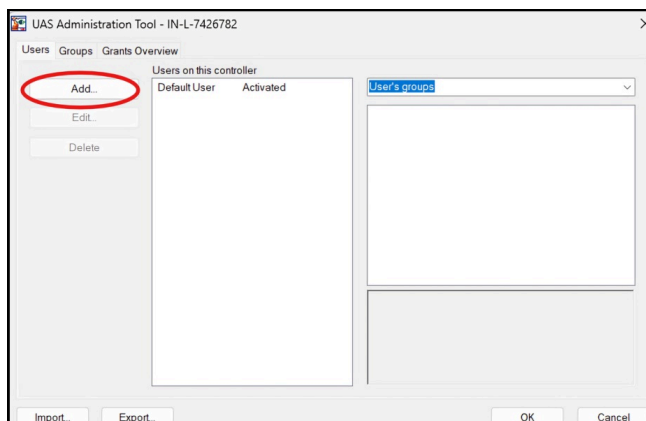
- Select “**Groups**” and Add new group.



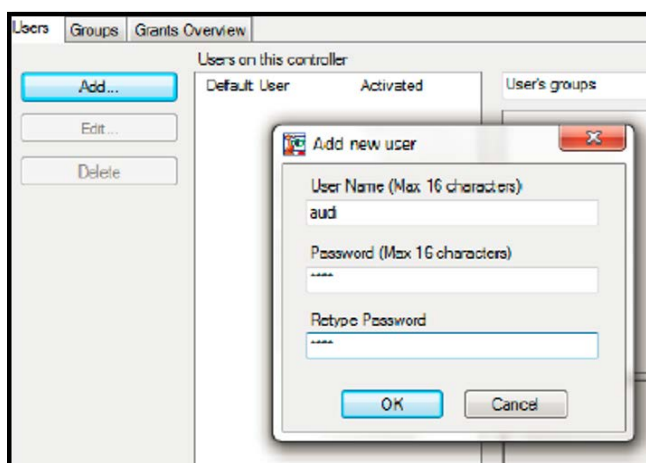
- Select the groups access.



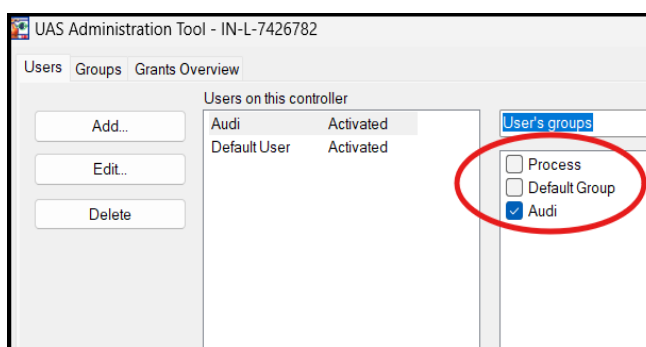
- Select **Users** and **Add** to create a new user.



- Set the new user account and password.



- Set the new user access and choose the groups which need to access.



- Restart controller.



Important: The account have to be changed to restrict access for unauthorized personal. Especially the administrator and safety user must not be available for personal use without safety training.

3 Installation

3.3 Robot system installation

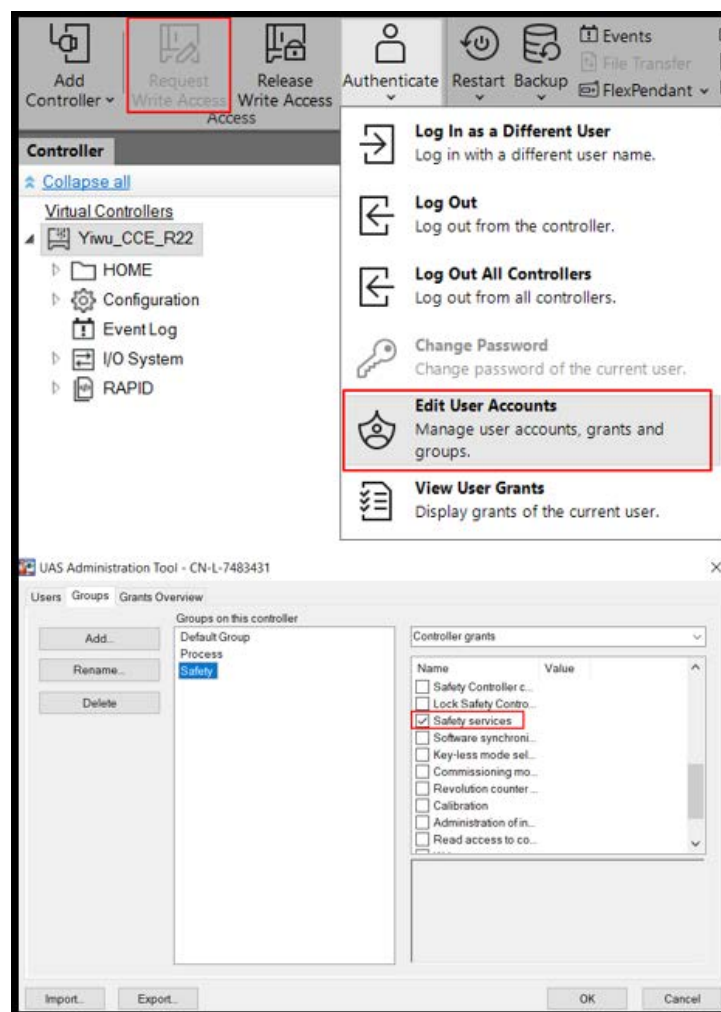
3.3.9 Safety configuration

The purpose of the safety module and the functional safety options is to provide a robust and easy-to-use safety controller in the robot system. Functional safety includes a complete software and hardware solution that is fully integrated with the robot controller and the RobotStudio programming environment.

Operation:

Use the following steps :

1. Configuration Safety user.

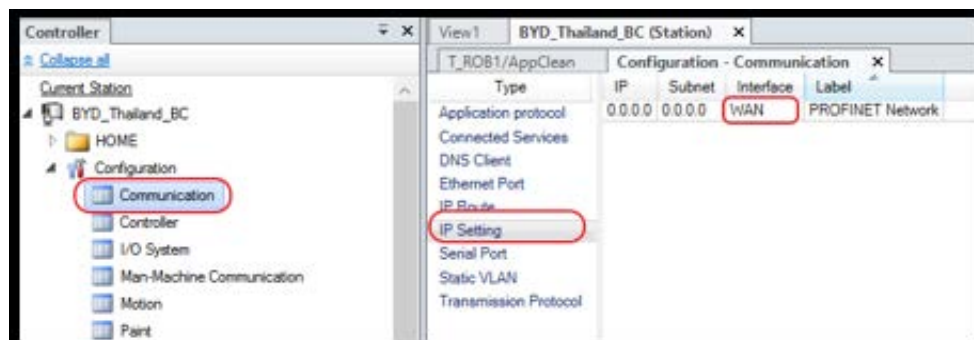


2. Configuration safety network.

PROFINET:

- Set the PROFINET interface.
- Using the “WAN” port means PROFINET and Ethernet share the same port and address.

- Using the “LAN3” port means PROFINET have a separate Ethernet port.



Configuration - Communication X				
Type	IP	Subnet	Interface	Label
Application protocol	192.168.127.1	255.255.255.0	N/A	Axis computer 1
Connected Services	192.168.126.1	255.255.255.0	LAN	FlexPendant Network
DNS Client	192.168.125.1	255.255.255.0	LAN	Private Network
Ethernet Port	192.168.1.128	255.255.255.0	LAN3	PROFINET Network
IP Route	10.47.69.128	255.255.254.0	WAN	Public Network
IP Setting				
Serial Port				
Static VLAN				
Transmission Protocol				

Configuration - I/O System X				
Type	Name	Connection	Identification Label	PROFINET Station Name
Access Level	EtherNet/IP	Private Network	EtherNet/IP Adapter Network	N/A
Cross Connection	ICI	N/A	ICI Connection	N/A
Device Trust Level	Local	N/A	Local	N/A
EtherNet/IP Command	PROFINET	PROFINET Network	PROFINET Device Network	robotbasicio128
EtherNet/IP Device	Virtual	N/A		N/A
ICI Device				
Industrial Network				
PROFINET Common Data				

3.3 Robot system installation

Configuration - Communication X				
Type	IP	Subnet	Interface	Label
Application protocol	192.168.127.1	255.255.255.0	N/A	Axis computer 1
Connected Services	192.168.1.12	255.255.255.0	LAN3	EtherNet/IP Network
DNS Client	192.168.126.1	255.255.255.0	LAN	Plex Pendant Network
Ethernet Port	192.168.125.1	255.255.255.0	LAN	Private Network
IP Route	10.47.89.135	255.255.254.0	WAN	Public Network
IP Setting				
Serial Port				
Configuration - I/O System X				
Type	Name	Connection	Identification Label	
Access Level	EtherNet/IP	EtherNet/IP Network	EtherNet/IP Adapter Network	N/
Cross Connection	ICI	N/A	ICI Connection	N/
Device Trust Level	Local	N/A	Local	N/
EtherNet/IP Command	Virtual	N/A		N/
EtherNet/IP Device				
EtherNet/IP Internal Device				
ICI Device				
Industrial Network				
Route				

- Source and Destination address for PROFIsafe must match the configuration in the PLC.

Configuration

- L3_CL1 (172.27.41.100)
 - Communication
 - I/O System
 - PROFINET
 - Device
 - PN_Internal_Device
 - 01: DO_64_bytes*
 - 02: DI_64_bytes*
 - 03: SDO_8_bytes*
 - 04: SDI_8_bytes*

Signal Editor

Name	Assigned to Device	Type of Signal
fi_EnableNonMotionExecution	SDI_8_bytes	Digital Input
fi_StandStillSupervision	SDI_8_bytes	Digital Input
fi_Zone08PosSupervision	SDI_8_bytes	Digital Input
fi_Zone07PosSupervision	SDI_8_bytes	Digital Input
fi_Zone06PosSupervision	SDI_8_bytes	Digital Input
fi_Zone05PosSupervision	SDI_8_bytes	Digital Input
fi_Zone04PosSupervision	SDI_8_bytes	Digital Input
fi_Zone03PosSupervision	SDI_8_bytes	Digital Input
fi_Zone02PosSupervision	SDI_8_bytes	Digital Input
fi_Zone01PosSupervision	SDI_8_bytes	Digital Input
fi_TestModeStop	SDI_8_bytes	Digital Input
fi_DelayedStop	SDI_8_bytes	Digital Input
fi_GeneralStop	SDI_8_bytes	Digital Input
fi_AutoStop	SDI_8_bytes	Digital Input
fi_ExtEmyStop	SDI_8_bytes	Digital Input
fi_EmyStopReset	SDI_8_bytes	Digital Input
fi_CBSDisCon	SDI_8_bytes	Digital Input
fi_ServoDisCon	SDI_8_bytes	Digital Input
fi_MainAirSupp	SDI_8_bytes	Digital Input
fi_TurbineLLK	SDI_8_bytes	Digital Input
fi_Sys2ILK	SDI_8_bytes	Digital Input
fi_ProcILK	SDI_8_bytes	Digital Input

Properties

System

Name	SDI_8_bytes
Identification Label	
Vendor Name	
Product Name	

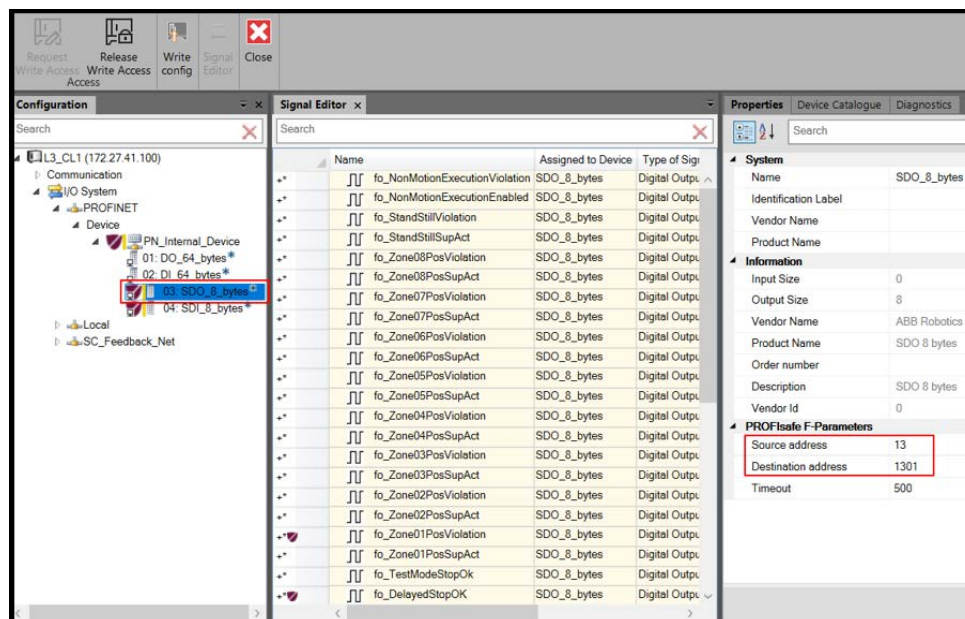
Information

Input Size	8
Output Size	0
Vendor Name	ABB Robotics
Product Name	SDI 8 bytes
Order number	
Description	SDI 8 bytes
Vendor Id	0

PROFsafe F-Parameters

Source address	13
Destination address	1302
Timeout	500

Output :



4. Stop configuration.

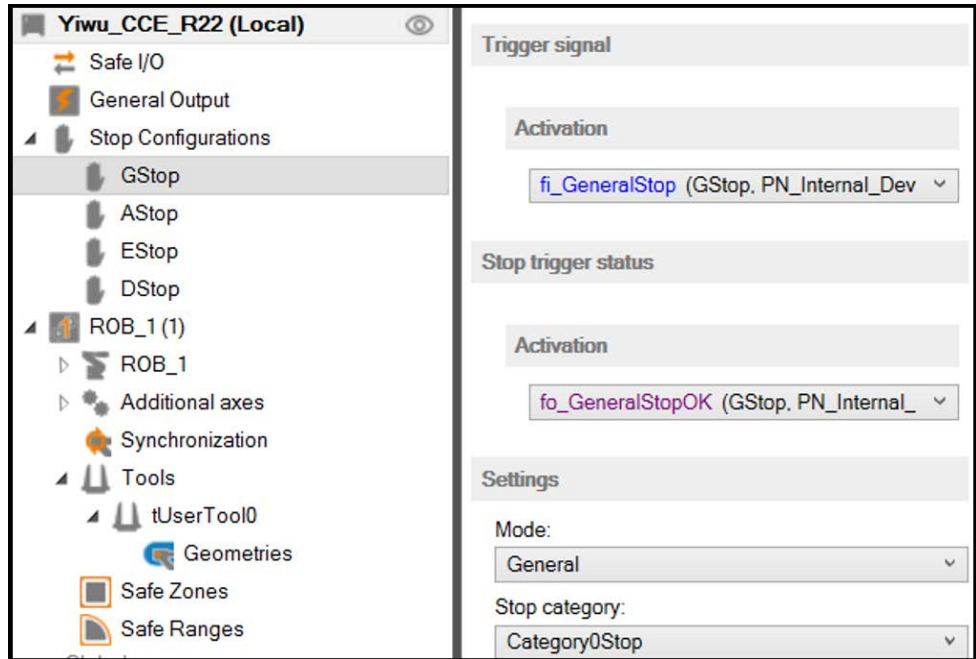
The **Stop Configuration** is used to configure a safety stop that is sent on the safe field bus from the safety PLC to the robot controller.

- **Category0Stop** - Stop by immediate removal of power to the actuators. Mechanical brakes are applied. A robot that is stopped with a category 0 stop does not follow its programmed path while decelerating.
- **Category1Stop** - Controlled stop with power available to the actuators to achieve the stop. Power is removed from the actuators when the stop is achieved.

3 Installation

3.3 Robot system installation

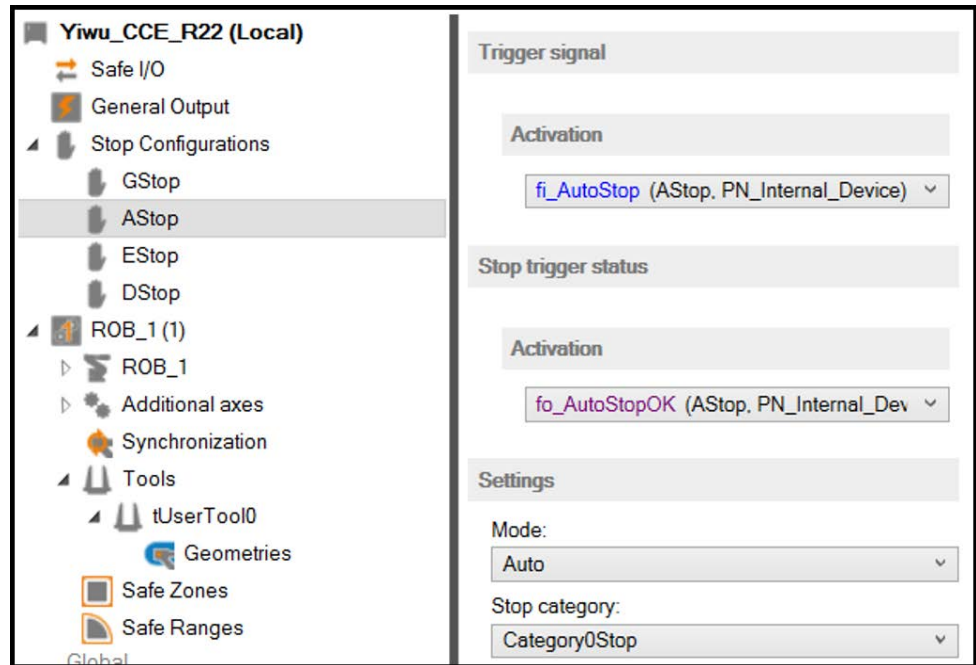
GStop



General Stop -

It sets the IRC5P controller in general stop mode (GS). GS is Category0Stop.

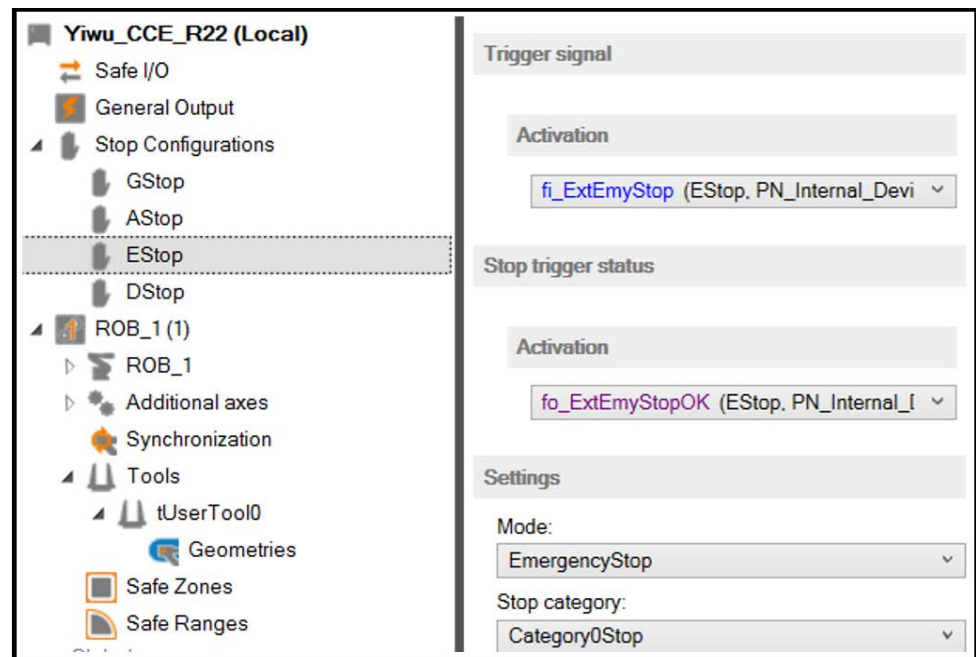
AStop



Auto Stop -

It sets the robot controller in auto-stop mode (AS). AS is Category0Stop.

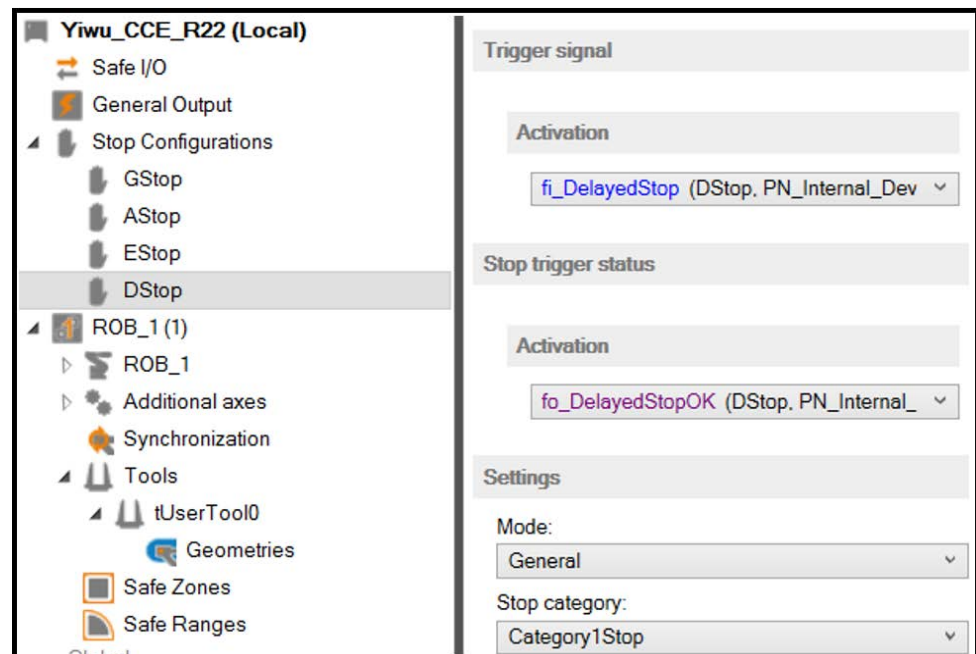
EStop



Emergency Stop -

It puts the robot controller in emergency stop mode (ES). ES is Category0Stop.

DStop



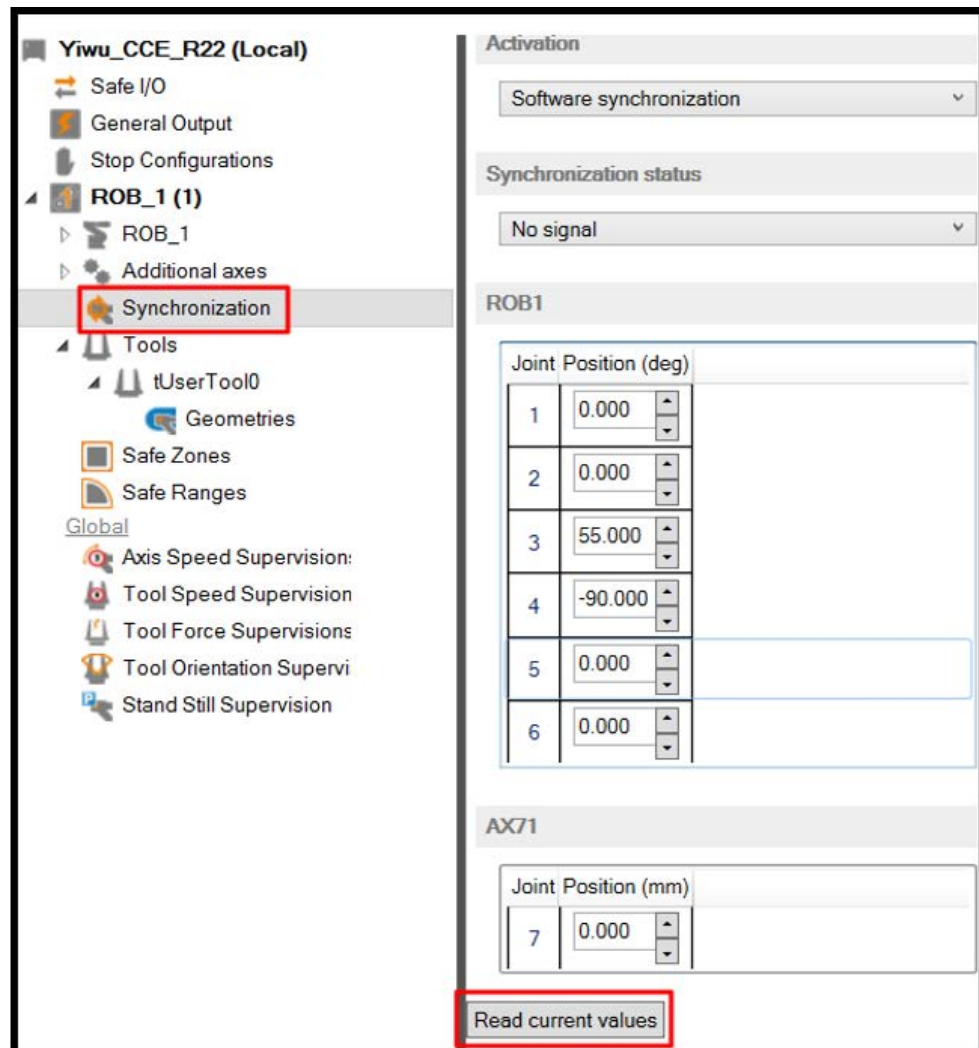
Delayed Stop -

If you do not need to stop the robot right away, delayed stop is used, which is determined by the stop level. DStop is a Category 1 Stop.

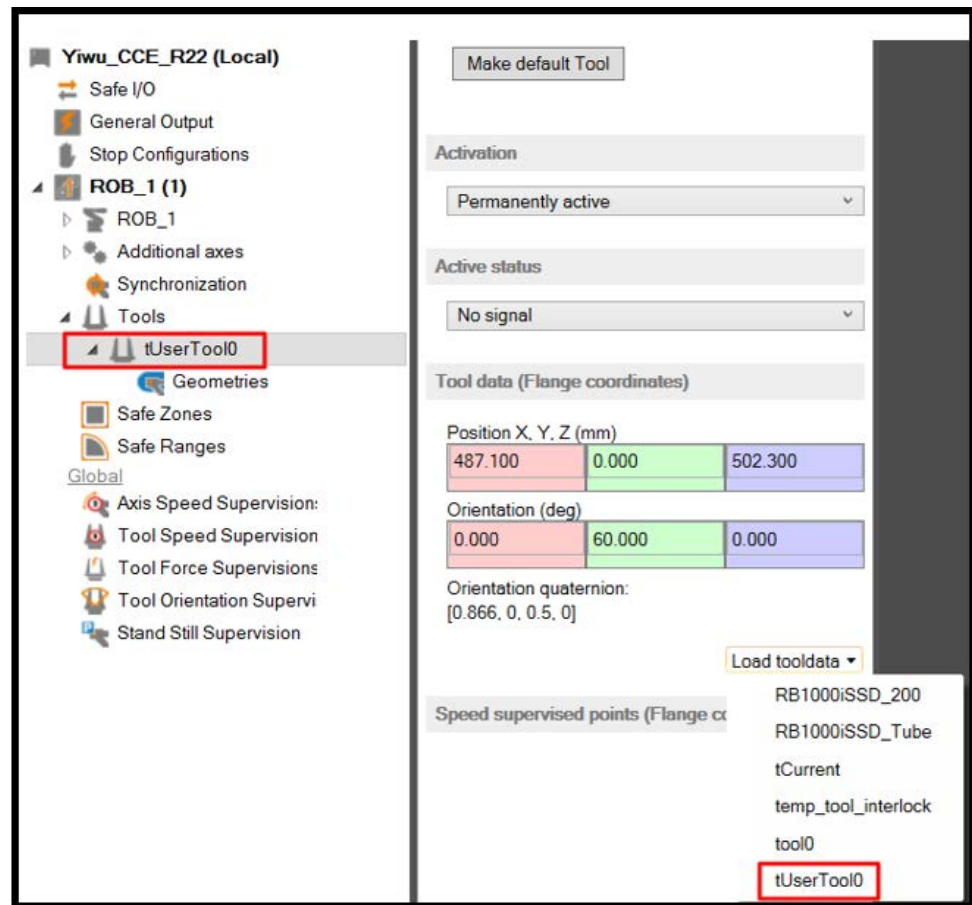
3 Installation

3.3 Robot system installation

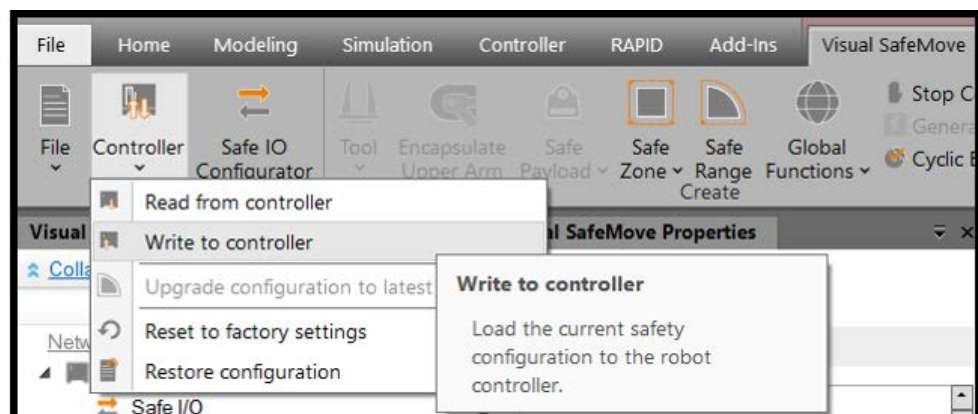
5. Synchronization - Robot needs calibration position.



6. Tool configuration.



7. Download safety configuration.



3.3 Robot system installation

3.3.10 Setting date and time

The following sub-sections are included in this section:

['Setup time with TPU'](#) on page 51

['Setup time with Robotstudio'](#) on page 51

3.3.10.1 Setup time with TPU

For more information, see the Installation manual, IRC5P RW6.0 / 5 Configuration / 5.2 General Configuration / 5.2.8 Setting date and time.

3.3.10.2 Setup time with Robotstudio

For more information, see the Operating manual – RobotStudio / 18 Controller tab / 18.6 Properties / Setting the controller date and time.

3.3.11 System parameter configuration

The following sub-sections are included in this section:

['PaintWare system parameters'](#) on page 52

['General system parameters'](#) on page 56

['Paint robot system parameters'](#) on page 58

['Opener robot system parameters'](#) on page 61

['PixelPaint robot system parameters'](#) on page 61

['Flaming robot system parameters'](#) on page 61

['Cleaning robot system parameters'](#) on page 64

['PaintCheck robot system parameters'](#) on page 64

['Dispensing robot system parameters'](#) on page 64

Some parameters of RobotWare Paint can be configured through the Paint/Process domain, also known as the PROC domain. Like other system parameters, these require a controller warm start to activate parameter changes.

3.3.11.1 PaintWare system parameters

PaintControl package sets some of the PaintWare system parameters to predefined values.

System parameter	Value	Description
A1BrushDevice	A1Brush	Name of the brush IPS device to interact with.
A1BrushTabLimEnSignal	A1BrushTabLimEn	Name of the brush table limit enable EIO signal to interact with.
A1BrushTabNdxSignal	A1BrushTabNdx	Name of the brush table EIO signal to interact with.
A1EnableSignal	A1Enable	Name of the applicator enable EIO signal to interact with.
A1ErrorSignal	A1Error	Name of the applicator error EIO signal to interact with.
A1FluidAccuSignal	A1FluidAccu	Name of the accumulated fluid EIO signal to interact with.
A1FluidEnSignal	A1FluidEn	Name of the fluid enable EIO signal to interact with.
A1HVENSignal	A1HVEN	Name of the high voltage enable EIO signal to interact with.
A1HVErrNoSignal	A1HVErrNo	Name of the high voltage error EIO signal to interact with.
A1MaterialSignal	A1Material	Name of the material EIO signal to interact with. Note: Using a blank value will prevent the system from setting it.
A1NeedleEnSignal	A1NeedleEn	Name of the needle valve enable EIO signal to interact with.
AppCleanModuleFile		Path to an index file containing a mapping between material index and module name. Can be used to select different modules to run depending on material.
BrushMapFilter	999999	Any result from the brush map formula calculation that is above this value will not be set.
BrushMapFormula	(P*100)+M	Brush formula calculation.
BrushMapZeroDefault	0	Specifies another value to set when brush table 0 would otherwise be set.
BrushTableDirectory	HOME:/ipsdata/node 1/A1Brush/	The path of brush table.
BrushTrackDirecotry	HOME:/btrack/	The path of brush tracking.
BrushTrackFormula	P	The brush tracking file formula.
BrushTrackSignal	A1Brush	The brush tracking signal.
CmdLogDirectory	HOME:/logs	Location of where to save command logging files.
CmdLogSize	100000	Size is in bytes, and defines when the old log file will be rotated by a new one. Only two files are kept, so if there is another old file present, this will be deleted. A value of 0 disables logging.

3 Installation

3.3 Robot system installation

System parameter	Value	Description
ExternalStart	False	Enables or disables external start, meaning an extra sync signal is needed before the paint program starts. Typically used on stop-and-go lines (no conveyor tracking).
HVAckDelay	200	Time in milliseconds between setting high voltage acknowledge and attempting enable of the high voltage IPS device. Used by the high voltage enable paint command.
HVAckSignal	HV_ILCK_ACK	Name of high voltage acknowledge signal. Set by high voltage enable paint command.
JobQueueIgnoreMaterial	False	Is set to True on systems without a material change engine that can pop the material number from the job queue.
JobQueueSize	2	Defines an upper limit on the job queue. No more jobs can be appended or inserted if it's 'full'.
LoadFailureModule		If a module is specified, this will run when failures relating to program loading occurs, instead of stopping. Execution will also continue afterwards, potentially loading the next program from the queue.
MaterialAliasFile	HOME:/alias/material.map	Location of material alias file on controller.
MCDDataDirectory	HOME:/mcdata/	Location of material change index files and sequences.
MCDisablePreStart	False	If set to True, then the next material will not be immediately retrieved from the job queue, but instead later when the program has been finished. This prevents the job queue from being locked for editing, but will affect cycle time on systems like CBS and AB where pre-processing is needed.
MCRecoverOnStart	True	If set to True, the system will automatically attempt to resume suspended material change on regular robot (program) start.
OptionAliasFile	HOME:/alias/option.map	Location of option alias file on controller.
ProdLogDirectory	HOME:/logs	Location of where to save production logging files.
ProdLogFormat	%D,Prog:%P,Mtrl:%M,JobMode:%1,Part:%2,Skid:%I,Brush:%B,Usage:%C,CTime:%T,%a,%U	Format in which to write to the production log file.
ProdLogSize	100000	Size is in bytes, and defines when the old log file will be rotated by a new one. Only two files are kept, so if there is another old file present, this will be deleted. A value of 0 disables logging.
ProgramAliasFile	HOME:/alias/program.map	Location of program alias file on controller.

System parameter	Value	Description
ProgramBackupSave	False	If set to True, then a backup of paint program that are replaced (unloaded and re-loaded) will be saved with the ".bak" extension.
ProgramCache	False	If set to True, program modules will not be unloaded after use. This can in some cases improve cycle time.
ProgramDirectory	HOME:/programs/	Location of paint programs.
ProgramIndexFile	HOME:/programs/segments.ndx	Location of program index and segmentation file.
ProgramStaticLoad	False	If set to True, then paint programs will be loaded statically instead of dynamically into RAPID memory.
RobotDataDirectory	HOME:/robdata	Location of RAPID data system modules.
MCEngine_nAForceInterval	0	How many times the same material is loaded before a material change is forced automatically. (A value of 0 Disables this function.) Note: The RAPID variable 'nAForceInterval' can also be modified after startup to change the interval while the system is running.
MCEngine_CleanIndex	999	Material Index recognized as running cleaning sequence only.
MCEngine_ClnGradeDefault	1	When the index file for cleaning sequences does not exist or when a particular combination is not found in the index file, this grade is chosen.
MCEngine_DebugEventLog	False	Enables or disables material change event logging to the debug log.
MCEngine_DebugStateLog	False	Enables or disables material change state change logging to the debug log.
MCEngine_FillGradeDefault	1	When the index file for filling sequences does not exist or when a particular combination is not found in the index file, this grade is chosen.
Single2K_CleanCatOn1K	False	Enables or disables cleaning of catalyst when switching from 2K to 1K material.
Single2K_CleanThroughApp	False	Enables or disables the use of the mixer cleaning sequence.
Single2K_ParallelClean	False	Enables or disables cleaning of resin and catalyst in parallel.
Single2K_ParallelFill	False	Enables or disables filling of resin and catalyst in parallel.
RCC2K_ParallelMixCln	True	Enables or disables cleaning of mixer in parallel with resin processing.
JobQueueSave	True	If set to True, the job queue data will be saved on shutdown/restart, and re-load on startup. Mostly helpful if running without conveyor tracking, since the conveyor queue is lost anyway.

3 Installation

3.3 Robot system installation

System parameter	Value	Description
JobQueueExpediteBegin	10001	Specifies the beginning for the range for program (indexes) that should be automatically put into the high priority slot instead of the queue. Note: Only applies for the job queue append command (421), and all input except the program index will be discarded.
JobQueueExpediteEnd	10040	Specifies the end of the range for expedited program indexes.
MCPosModule	yMtrlChPos	Module name to load for moving to material change position (and synchronize).

The below configuration can be changed based on the project requirements.

System parameter	Value	Description
BrushMapFormula	$(P \cdot 100) + M$	Brush formula calculation
BrushTrackFormula	P	If need brush tracking function
BrushTrackSignal	A1Brush	If need brush tracking function.

For more information, see the Installation manual, IRC5P RW6.0 / 8 Paint user files/8.5 PaintWare system parameters.

3.3.11.2 General system parameters

The following parameters is the default value form paint controls.

System parameter	Value	Description
CustData_TuneServoEnable	True	Enable or disable tune servo function.
CustData_RobotID	L1_BCE_R11	Format: line number_station type_robot name 1. Line number define which line the robot is installed. Example : L1, L2... 2. Station type define the robot station name. Example: • "BCE" is BC external station, • "CCI" is CC interior station. 3. Robot name define : • 1 st string: Robot type is painter (R), Door opener (D) and Hood/Trunk opener (H). • 2 nd string: Robot is left (1) of right (2). • 3 rd string: Robot installation position. Example: • "R12" is a painter and the 1 st robot on the left. • "D22" is a door opener and the 2 nd robot on the right. • "H13" is a hook/trunk opener and the 3 rd robot on the left.
CustData_Station	BCE	The station name.
CustData_StationType	Int	Int: Interior station, Ext: Exterior station.
CustData_RobotName	R11	The name of robot.
CustData_RobotSide	L	R: Right side; L: Left side.
CustData_RobotPosition	1	The position of robot.
CustData_RobotFunc	Painter	Painter: paint robot; HoodTrunk: Hood trunk opener; Door: Door opener.
CustData_SubFolder	True	Enable or disable load program module in sub folder.
CustData_RobotnamePrefix	False	Add the robot name in the head for paint program name. If set to true. Example : D11_BCI_H5.
CustData_RobotnamePostfix	True	Add the robot name in the tail for paint program name. If set to true. Example : BCI_H5_D11.
CustData_CheckHomePos	True	Check if the robot is at home position before load program.
JobQueueIgnoreMaterial	False	Is set to True on systems without a material change engine that can pop the material number from the job queue.
CustData_ExtaxControl	False	Enable or disable whether need to get the position of the robot external axis.
CustData_nExtraControlOffset	20000	The external axis position offset value. If the "ExtaxControl" set to true, robot will sent the calculate result that "actual external axis position" + "offset value" to PLC.
CustData_WZEnable	True	True: Enabled; False: Disable.

3 Installation

3.3 Robot system installation

System parameter	Value	Description
CustData_WZLowPoint	[-20000, -20000, -20000]	The position value of lower point.
CustData_WZHighPoint	[20000,20000 ,20000]	The position value of higher point.
CustData_ServicePosMoveEn	False	Check if service portion need push enable button to move.
CustData_CnvSpeed	0.5	The real conveyor speed.
CustData_JobPassControl	True	True: When the job passes the start window, it will raise a robot error. False: No function.
CustData_JobPassException	M99	When the module name same as this parameter, there will no robot error when job pass start window.
CustData_SmarHome	True	True: Smart home Enable False: Smart home disable, use normal smart home.
CustData_SmartBlock	,,,,	Set which robot will block this robot move to home when others are not at home position.

The below parameter needs to be changed based on project requirements.

System parameter	Value	Description
CustData_RobotID	L1_BCE_R11	The robot ID.
CustData_Station	BCE	The station name.
CustData_StationType	Int	Int: Interior station, Ext: Exterior station.
CustData_RobotName	R11	The name of robot.
CustData_RobotSide	L	R: Right side; L: Left side.
CustData_RobotPosition	1	The position of robot.
CustData_RobotFunc	Painter	Painter: paint robot; HoodTrunk: Hood trunk opener; Door: Door opener.
CustData_SubFolder	True	Enable or disable load program module in sub folder.
CustData_RobotnamePostfix	True	Add the robot name in the tail for paint program name. If set to true. Example : BCI_H5_D11.
CustData_WZLowPoint	[-20000,-20000,-20000]	The position value of lower point.
CustData_WZHighPoint	[20000,20000,20000]	The position value of higher point.
CustData_ServicePosMoveEn	False	Check if service position need push enable button to move.
CustData_SmartHome	True	True: Smart home Enable . False: Smart home disable, use normal smart home.
CustData_SmartBlock	,,,,	Set which robot will block this robot move to home when others are not at home position.

3.3.11.3 Paint robot system parameters

The following parameters are available only on a painting robot:

System parameter	Value	Description
CustData_CheckAPP	True	Check if the paint is filled or if the atomizer is enabled before load paint program.
CustData_Reopenvalve	True	Check if color value will reopen when job start, this is used for cavity store function.
CustData_AppMasterPLC	True	True: Applicator enable control by PLC False: Applicator enable control by Robot.
CustData_IdleBrush	901	Called when the robot is idle.
CustData_StartBrush	902	Called when the robot start to production.
CustData_HaltBrush	903	Called when the robot is halt.
CustData_MtrlChBrush	904	Called when the robot is executing the material change.
CustData_StaticBrush	905	Called when the robot executing the static brush test program.
CustData_DynamicBrush	906	Called when the robot executing the dynamic brush test program.

3 Installation

3.3 Robot system installation

System parameter	Value	Description
CustData_GunClnBrush	907	Called when the robot executing the Gun clean program.
CustData_BellClnBrush	908	Called when the robot executing the Bell clean program.
CustData_HubClnBrush	909	Called when the robot executing the Hub clean program.
CustData_HVDyTestBrush	910	Called when the robot executing the HV Dynamic test program.
CustData_HVStaTestBrush	911	Called when the robot executing the HV Static test program.
CustData_AutoShapeBrush	912	Called when the robot mode switch switches to Maintenance or Clean mode.
CustData_stBypassSafeCheck	False	Whether or not bypass the HV safe check When calling global cleaning brush. If the value set to false, means not bypass, the HV actual value must be reduced to < 5kv.
CustData_CnvHoldEnable	True	Enable or disable the material change conveyor hold request function. If the value set to false, the conveyor will not hold when the conveyor reach the robot start painting position during executing the material change.
CustData_CnvHoldBuff	0.5	The time to calculate the conveyor buffer position for stop the conveyor.
CustData_RunAdvanceMC	False	Enable or disable the advance material change function.
CustData_BellCln	True	True bell clean enable, false bell clean disable.
CustData_BellClnModule	BellClean	Define the bell clean module name.
CustData_BellClnSequence	BellCln	Define the bell clean sequence name.
CustData_ForceBellCln	True	Enable or disable the Force bell clean. If set to true, and the "MaxSkipBell" set to >0, When the count for painting the same material that reached to the "MaxSkipBell" set value, the bell clean program will force executing.
CustData_MaxSkipBell	0	How many times the same material has been painted before a bell clean is forced executing.
CustData_HubClnModule	HubClean	Define the Hub clean module name.
CustData_HubClnSequence	HubCln	Define the Hub clean sequence name.
CustData_DumpCln	True	Enable or disable the dump clean function.
CustData_DumpClnInt	1	The Dump clean frequency.
CustData_DumpClnSequence	DumpCln	Define the Dump clean sequence name.
CustData_GunClnModule	GunClean	Define the gun clean module name.
CustData_GunClnSequence	GunCln	Define the gun clean sequence name.
CustData_ToolClnModule	ToolClean	Define the tool clean module name.
CustData_ToolClnSequence	ToolCln	Define the tool clean sequence name.
CustData_CheckSolvPS	True	Enable or disable the solvent pressure check function. The function will check the solvent pressure status before executing the cleaning program.
CustData_MaintShape1	200	The Shape1 brush value in Maint mode . When the robot mode switch switches to Maint or Clean mode, the MaintShape1 will be activated.

System parameter	Value	Description
CustData_MaintShape2	200	The Shape2 brush value in Maint mode. When the robot mode switch switches to Maint or Clean mode, the MaintShape1 and MaintShape2 brush will be activated.
PntPush_PushEnable	True	Enable or disable the paint push function.
PntPush_GloablFactor	1	The Global factor to calculate the push volume.
PntPush_DefPushVolume	0	Define the volume inside the pipe. Used for push volume calculate.
PntPush_DefPushResinVolume	0	Define the volume inside the resin pipe. Used for push volume calculate (2K type).
PntPush_DefPushMixerVolume	0	Define the volume inside the mixer pipe. Used for push volume calculate (2K type).
PntPush_DefPushMethod	2	When the index file for pushing methods does not exist or when a particular combination is not found in the index file, this method is chosen. The following values can be used: 0 = No push, 1 = Push paint, 2 = Push solvent.
PntPush_MaxPaintPush	3	Determines how many times pushing can occur before a standard material change is forced. (A value of 0 disables this function).
PntPush_PushCavity	2	Chooses the cavity that will use paint push. The following values can be used: 1 = Mix cavity, 2 = Resin cavity. 3 =Catalyst cavity.
PntPush_PushMargin	3	Chooses how much paint can be pushed after pushing should be finished before it is detected as an overpush.
CustData_MixerForceInterval	0	The interval for force mixer clean with same color
CustData_UndockSafe	True	True: RCC will undock after clean system False: No function.
CustData_RccCavity	2	The cavity number of RCC.
CustData_BypassUndock	False	For test purpose.
CustData_OverrideBrush	False	True: Brush table will be send from PLC False: Use robot formula to define brush table.

The below parameter needs to be changed based on project requirements.

System parameter	Value	Description
CustData_RunAdvanceMC	False	Pre-start MC
PntPush_DefPushVolume	0	1K
PntPush_DefPushResinVolume	0	2K
PntPush_DefPushMixerVolume	0	2K

3 Installation

3.3 Robot system installation

3.3.11.4 Opener robot system parameters

The following parameters are available only on an opener robot:

System parameter	Value	Description
CustData_RobotArray	R11,R12, R13,R14, R15 R21,R22, R23,R24, R25	For Door opener, Show the painter robot name along the conveyor from entrance to exit.
CustData_RobotArray	R11,R21 R12,R22 R13,R23 R14,R24 R15,R25	For Hood/Trunk opener, Show the painter robot name for "Left, Right" Painter.
CustData_AutoSetHomePos	False	Set home position to opener interface
CustData_CheckHomeSensor	False	Check opener sensor whether on home status before load paint program.
CustData_OpenerAutoRecover	True	Enable or disable the auto-rise function after the gripper alarm.
CustData_OpenerErrMove	80	Define the auto rising height after the gripper alarm.
CustData_GripperSensorMaxCheckTime	0.5	The check time for sensor missing alarm(s)
CustData_RelatedAreaBits		Reserved
CustData_WorkAreaBit		Reserved

3.3.11.5 PixelPaint robot system parameters

The following parameters are available only on an PixelPaint robot:

System parameter	Value	Description
TBD	TBD	TBD

3.3.11.6 Flaming robot system parameters

The below system parameter is only for flaming robots:

System parameter	Value	Description
CustData_FlameOnStart	True/ False	True: System will turn on flaming when new job coming False: No Action

The following common parameters are available for all paint robots:

System parameter	Value	Description
CustData_EmptySkid	M99	The empty skid name

File name	Config	Value
Proc_common	CustData_FastTest	FALSE
	CustData_ErrLog	TRUE
	CustData_Reopenvalve	FALSE

File name	Config	Value
Proc_worldZone	CustData_WZLowPoint	[-20000,-20000,-20000]
	CustData_WZHighPoint	[20000,20000,20000]
	CustData_WZEnable	TRUE
	CustData_CleanerRadius	200
	CustData_CleanerHeight	400
	CustData_CleanerZOffset	100
Proc_appcIn	CustData_AppCInModule	TipClean
	CustData_AppCInSequence	TipcIn
	CustData_AppCInEnable	TRUE
	CustData_AppCInIndx	HOME:/mcdata/TipcIn.ndx
	CustData_AppCleanPre	FALSE
	CustData_AppCleanPost	TRUE
Proc_HubcIn	CustData_HubCInModule	HubClean
	CustData_HubCInSequence	HubcIn
	CustData_HubCInEnable	TRUE
	CustData_HubCInIndx	HOME:/mcdata/hubcIn.ndx
	CustData_ServPosControl	FALSE
Proc_extaxCheck	CustData_ExtaxControl	FALSE
	CustData_nExtraControlOffset	20000
Proc_cnv	CustData_CnvHoldBuff	0.5
	CustData_CnvHoldEnable	TRUE
Proc_appCheck	CustData_CheckAPP	TRUE
	CustData_AppMasterPLC	TRUE
Proc_sensor	CustData_CheckHomeSensor	FALSE
	CustData_GripperSensorMaxCheckTime	0.5
Proc_vmt3d	CustData_VMT3DEnable	TRUE
Proc_intopn	CustData_RobotArray	,,,
	CustData_AutoSetHomePos	FALSE
Proc_rcc_precc	CustData_RunAdvanceMC	TRUE
Proc_safemaint1	CustData_MaintShape1	200
	CustData_MaintShape2	200
Proc_intopn	CustData_OpenerAutoRecover	TRUE
Proc_idResolve	CustData_OpenerErrMove	90
	CustData_RobotID	L1_PRI_R11
	CustData_Station	PRI
	CustData_StationType	Int
	CustData_RobotName	R11
	CustData_RobotSide	R
	CustData_RobotPosition	1
	CustData_RobotFunc	Painter
Proc_mixerForceFlush	CustData_MixerForceInterval	0

3 Installation

3.3 Robot system installation

File name	Config	Value
Proc_rcc	CustData_UndockSafe	TRUE
	CustData_HubClnRotateCC	TRUE
	CustData_RccCavity	2
	CustData_BypassUndock	TRUE
Proc_smarthome	CustData_SmartHome	TRUE
	CustData_SmartBlock	,,,,,,,,,,,,,
Proc_tuneservo	CustData_TuneServoEnable	TRUE
Proc_solvsup	CustData_CheckSolvPS	TRUE
Proc_loadModule	CustData_SubFolder	TRUE
	CustData_RobotnamePostfix	TRUE
	CustData_RobotnamePrefix	TRUE
Proc_JobControl	CustData_JobPassControl	TRUE
	JobQueueSize	2
	CustData_JobPassException	M99
Proc_BrushOverride	CustData_OverrideBrush	FALSE
	BrushMapFormula	(P*1000)+(M*10)
	CustData_IdleBrush	901
	CustData_StartBrush	902
	CustData_HaltBrush	903
	CustData_MtrlChBrush	904
	CustData_StaticBrush	905
	CustData_DynamicBrush	906
	CustData_GunClnBrush	907
	CustData_BellClnBrush	908
	CustData_HubClnBrush	909
	CustData_HVDyTestBrush	910
	CustData_HVStaTestBrush	911
	CustData_AutoShapeBrush	912
	CustData_BypassSafeCheck	FALSE
Proc_smarthome	CustData_SmartHome	FALSE
Proc_dumpclean	CustData_DumpCln	TRUE
	CustData_DumpClnInt	1
	CustData_DumpClnSequence	DumpCln
Proc_Prevent	CustData_PreventTeachMode	TRUE
Proc_rcc	CustData_RccCalibClockWise	TRUE
Proc_Flaming	Proc_preprogram	Proc_preprogram
	Proc_preprogram	Proc_preprogram

3.3.11.7 Cleaning robot system parameters

The following parameters are available only on an cleaning robot:

System parameter	Value	Description
DefaultFDSpeed	60	Default FD move speed
ServiceMoveHold	False	True: Need push release button to enable robot move for service position False: No restrict

The following parameters are available only on an cleaning robot:

System parameter	Value	Description
TornadoCln	Tornado Service	Sword brush clean sequence
ServiceMoveHold	False	True: Need push release button to enable robot move for service position False: No restrict

3.3.11.8 PaintCheck robot system parameters

The following parameters are available only on an PaintCheck robot:

System parameter	Value	Description
TBD	TBD	TBD

3.3.11.9 Dispensing robot system parameters

The following parameters are available only on an dispensing robot:

System parameter	Value	Description
TBD	TBD	TBD

3.4 PLC system installation

The following sub-sections are included in this section:

['Introduction'](#) on page 64

['Requirement'](#) on page 64

3.4.1 Introduction

TBD

3.4.2 Requirement

TBD

3.5 HMI system installation

The following sub-sections are included in this section:

- 'Introduction' on page 65
- 'Hardware requirement' on page 66
- 'Software components' on page 66
- 'Disk partitions' on page 67
- 'Windows 10 OS setting' on page 67
- 'Tool software' on page 71
- 'SQL Server Installation' on page 71
- 'IRC5P OPC Server Installation' on page 77
- 'RobotStudio Installation' on page 80
- 'RobView5 Installation' on page 83
- 'Redis Database' on page 85
- 'SQL Server Management Configuration' on page 86
- 'IRC5P OPC Configuration' on page 90
- 'Program preparation' on page 91
- 'Authorization description' on page 91

3.5.1 Introduction

This document is intended for the installation of RobotUI software.

The following sub-sections are included in this section:

- 'How to use the document' on page 65
- 'Safety Precautions' on page 65

3.5.1.1 How to use the document

This manual contains information, specifications, and warnings for software installation.

3.5.1.2 Safety Precautions

If the operations in this document are incorrect, contact professional personnel. Otherwise, the production process may be affected.

3.5.2 Hardware requirement

The following sub-sections are included in this section:

['Minimum requirements'](#) on page 66

['Recommendation'](#) on page 66

3.5.2.1 Minimum requirements

1. CPU Intel i5-12600 K or AMD 5600 X or higher
2. Integrated GPU
3. System Windows 10 (64 bit)
4. RAM 8 GB
5. 256 GB SSD disk
6. Monitor 1920 * 1080 resolution
7. 100 m network card
8. Keyboard and mouse.

3.5.2.2 Recommendation

1. CPU Intel i7-12700 K or AMD 7700 X or higher
2. GPU professional graphics card with 2 GB memory
3. System Windows 10 (64 bit)
4. RAM 16 GB
5. 512+ GB SSD disk
6. Monitor 1920 * 1080 resolution
7. 1000 m network card
8. Keyboard and mouse.

3.5.3 Software components

Operating System Windows 10

1. IRC5 OPC Server
2. Microsoft SQL Server
3. Robot Studio (optional)
4. FFFTP (optional)
5. RobView 5 (optional).

3 Installation

3.5 HMI system installation

Note: Unless specified, the latest available version of all software will be installed by default. Software versions are subject to updates and may differ from those currently documented.

3.5.4 Disk partitions

The following sub-sections are included in this section:

['General Directory Structure'](#) on page 67

- Divide the hard disk into at least two partitions, each at least 256 GB.
- Drive C: Contains system software (operating system, Robot Studio, IRC5 OPC, SQL database, RobView5, RobotUI configuration file).
- Drive D: Installs RobotUI, RobotLog, and stores robot backups, database backups, tools, and installation software packages.

Drive	C	D
Name	OS (C)	Program (D)
Storage capacity	100 GB	200+ GB
File system	NTFS	NTFS
Partition type	Primary	Logical Drive

3.5.4.1 General Directory Structure

Drive	Catalog	Description
C	\Program Files (x86)\	A directory of all system programs (Robot Studio, IRC5 OPC etc.)
D	\ABB_RobotUI	Robot UI program, configuration, log

3.5.5 Windows 10 OS setting

Computer settings update continuously.

The following sub-sections are included in this section:

['Computer username'](#) on page 68

['Computer resolution'](#) on page 68

['Setting the system time format'](#) on page 69

['.NET framework 3.5 installation'](#) on page 69

['Power supply settings'](#) on page 70

['Ethernet configuration'](#) on page 70

3.5.5.1 Computer username

If there are no customer requirements:

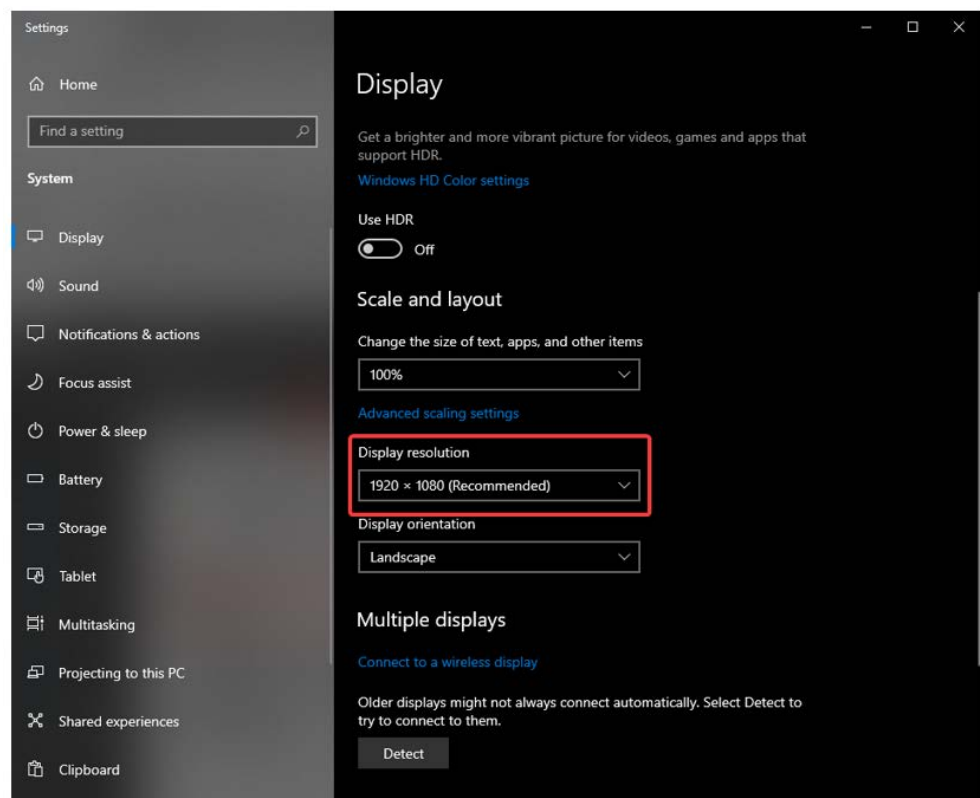
Default username: “ABB”

Default password: “ABB”

User must be an administrator of the user group.

3.5.5.2 Computer resolution

Right-click “This PC” > select “Display Settings,” and set the display resolution to 1920 x 1080.

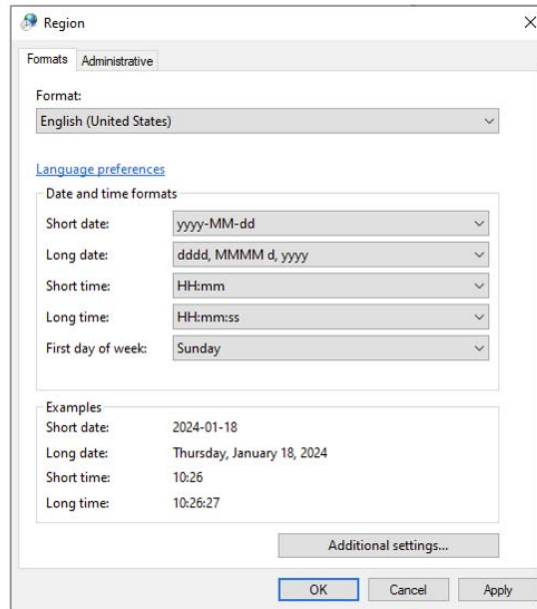


3 Installation

3.5 HMI system installation

3.5.5.3 Setting the system time format

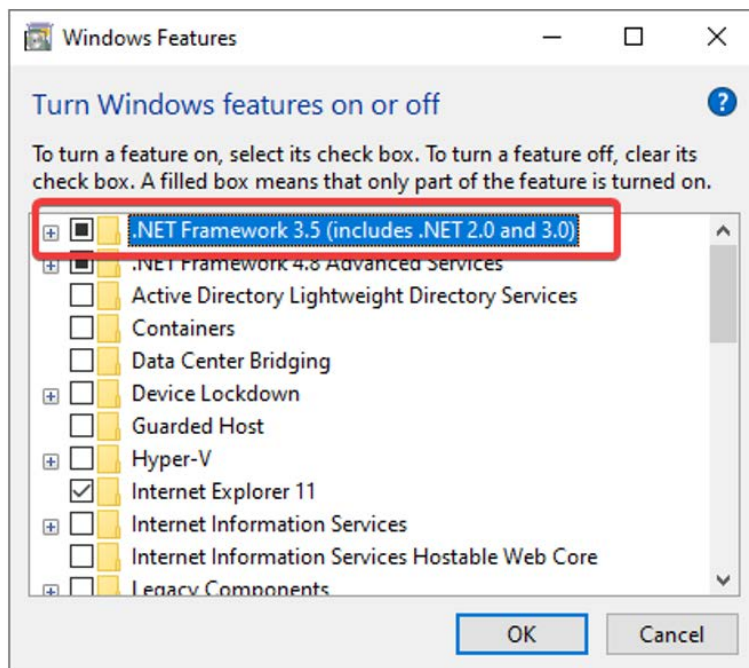
1. Open the Control Panel > go to “Date and Time” > select “Date and Time Settings” > click “Change date and time” > choose “Change calendar settings” > go to “Formats” and set the format as shown.



3.5.5.4 .NET framework 3.5 installation

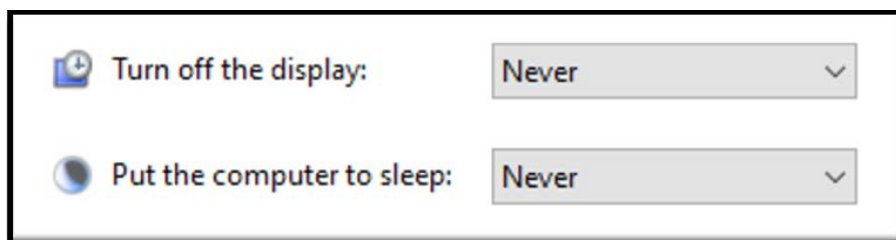
Open the Control Panel > go to “Programs” > select “Programs and Features” > click “Turn Windows features on or off” > find “.NET Framework 3.5 (includes .NET 2.0 and 3.0)” and check the box and click “OK” to install.

Tip: This method requires an internet connection.



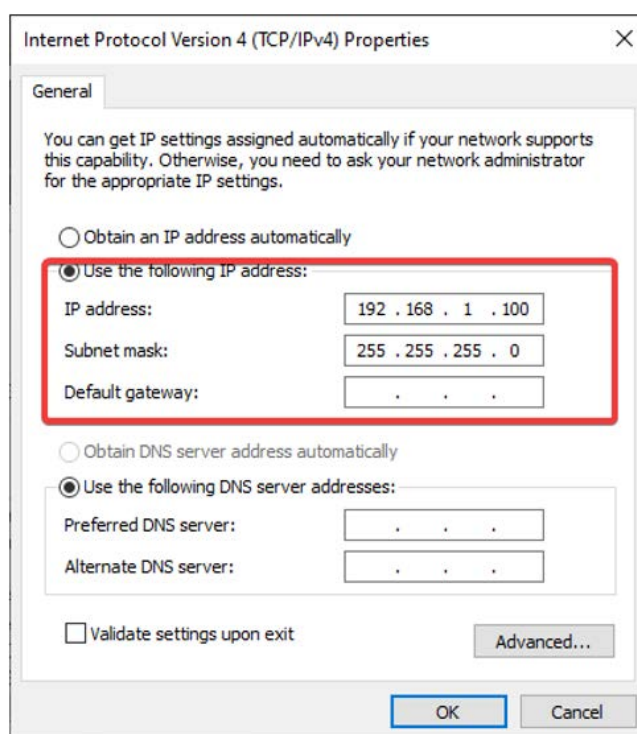
3.5.5.5 Power supply settings

Open the Control Panel > go to “Power Options” > select the time to turn off the monitor > set it to “Never” and set the computer to never go to sleep.



3.5.5.6 Ethernet configuration

Open the Control Panel > go to “Network and Sharing Center” > click “Change adapter settings” > right-click on “Ethernet” and select “Properties” > select “Internet Protocol Version 4 (TCP/IPv4)” > enter the assigned IP address > ensure the subnet mask matches the robot’s subnet mask > if the subnet mask is different, the robot won’t be found during IRC5 OPC configuration.



3 Installation

3.5 HMI system installation

3.5.6 Tool software

Install this program to support the tool software.

The following sub-sections are included in this section:

'7 - zip' on page 71

'Adobe reader' on page 71

3.5.6.1 7 - zip

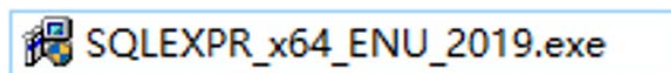
Download and install the latest 7-Zip from the '<https://www.7-zip.org>' website. You can also use other compression software like WinRAR.

3.5.6.2 Adobe reader

Download and install the latest Adobe Acrobat Reader from the '<https://www.adobe.com/acrobat/pdf-reader.html>' website.

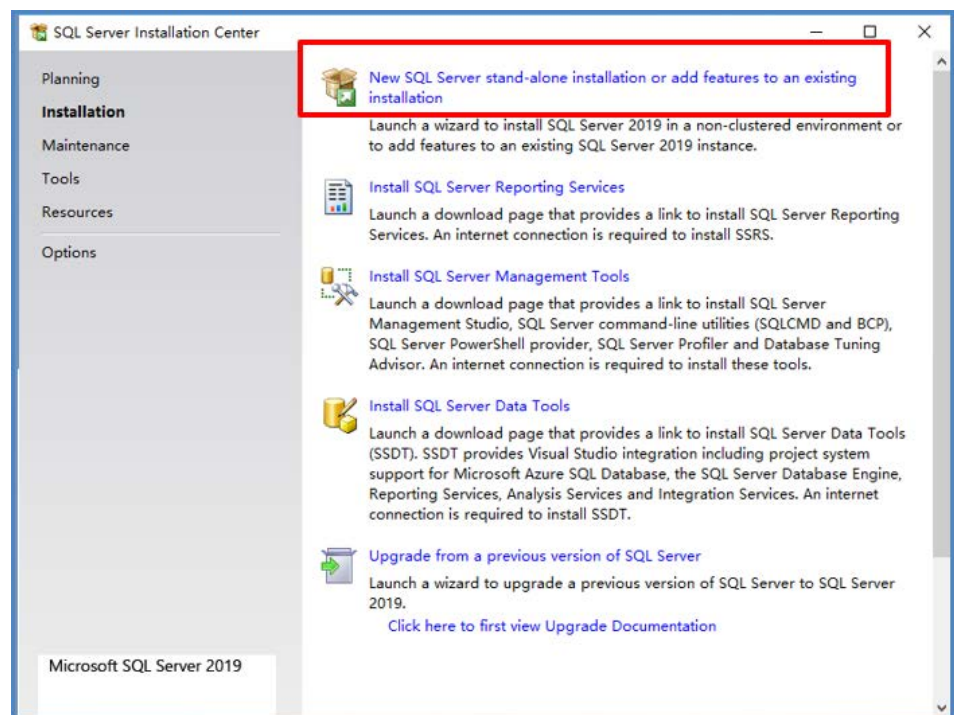
3.5.7 SQL Server Installation

1. Find the SQL2019 folder in the installation package and double-click it.

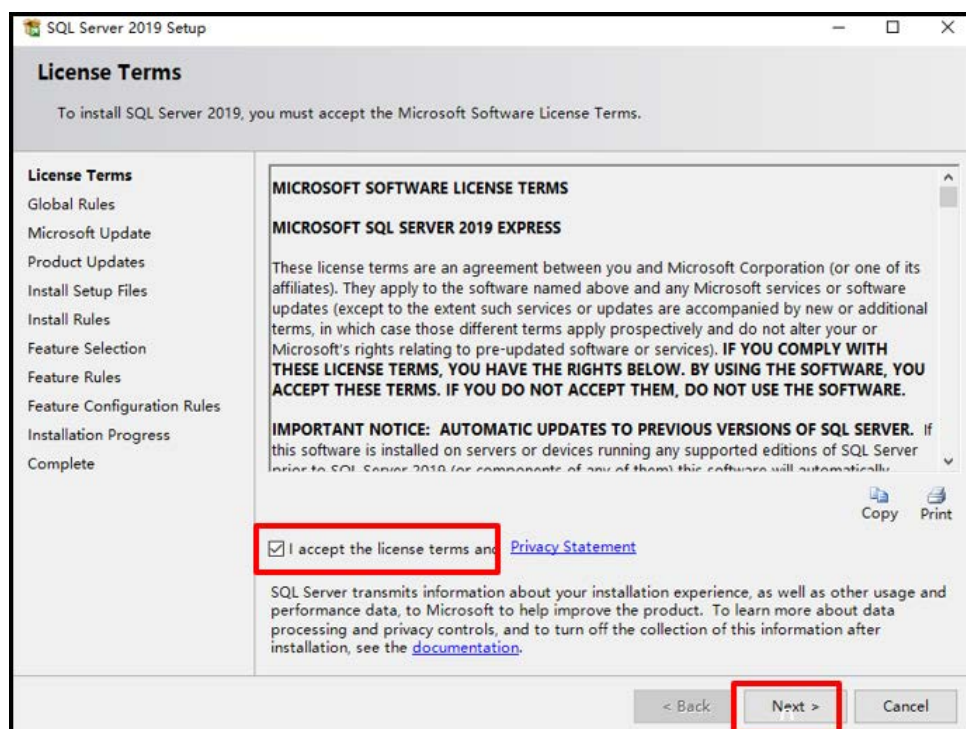


Begin the installation.

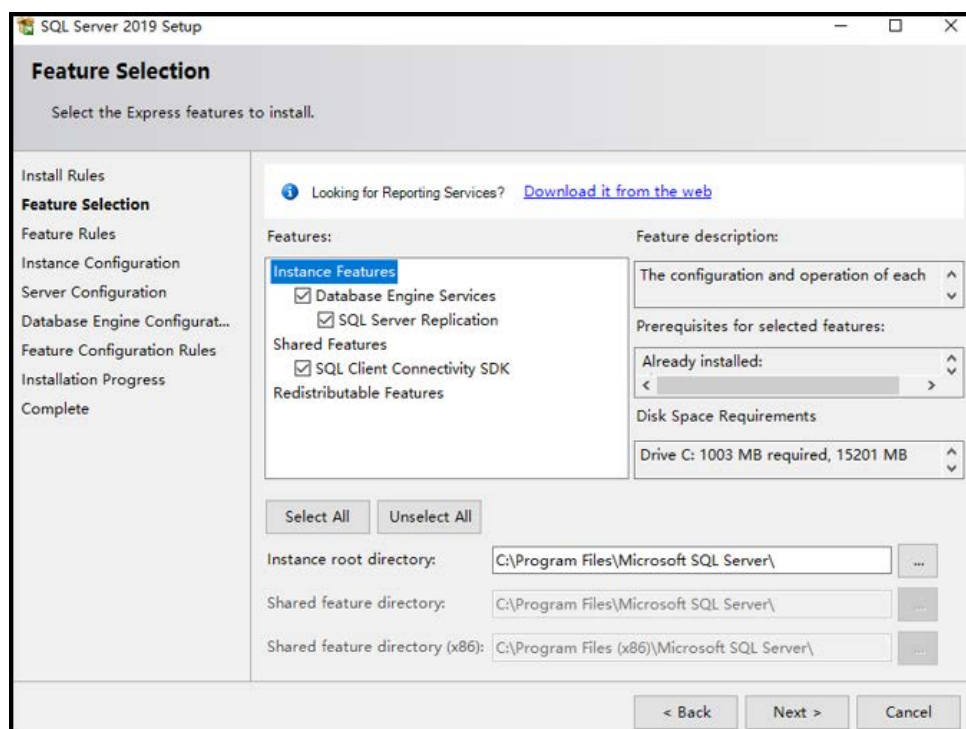
2. Choose the options shown in the figure.



3. After selecting the items in the figure, click Next.

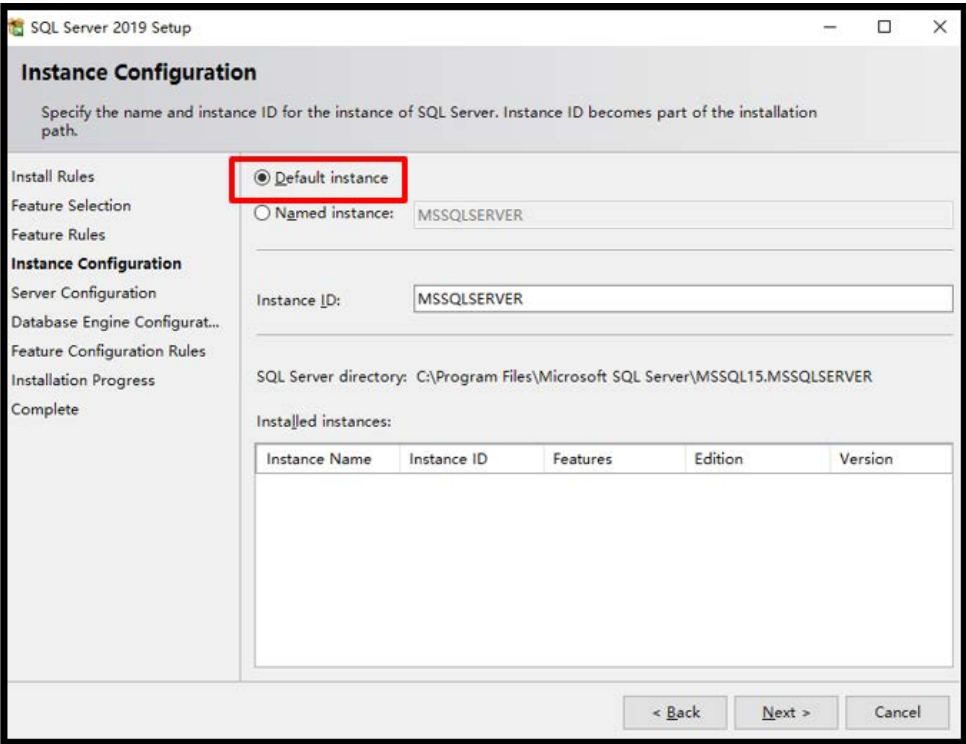


4. Go to the next step.

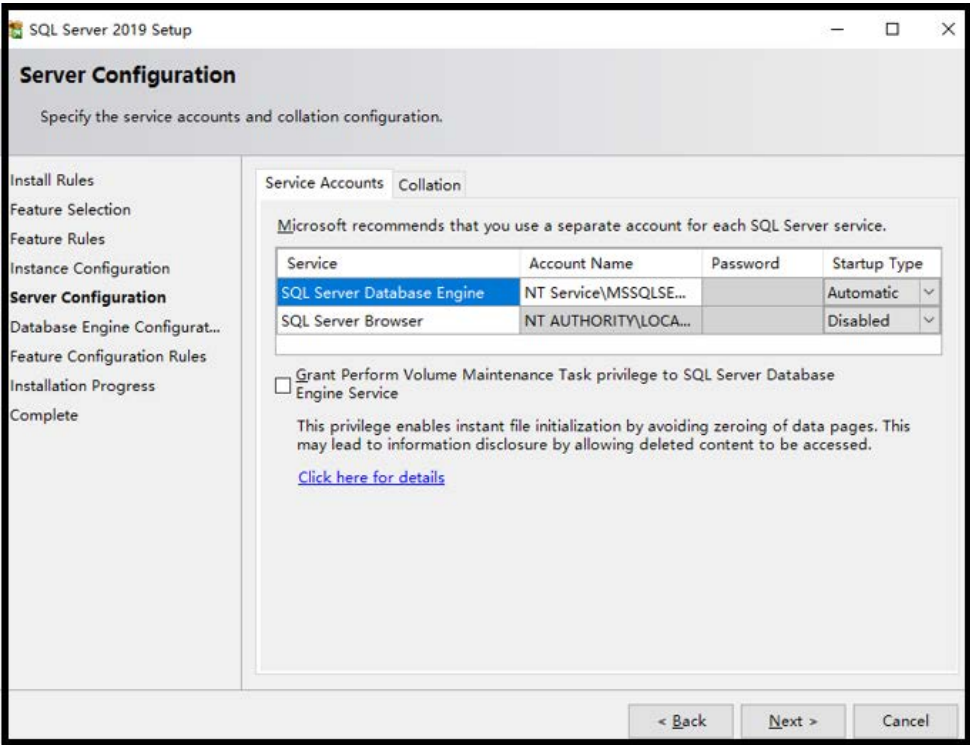


3.5 HMI system installation

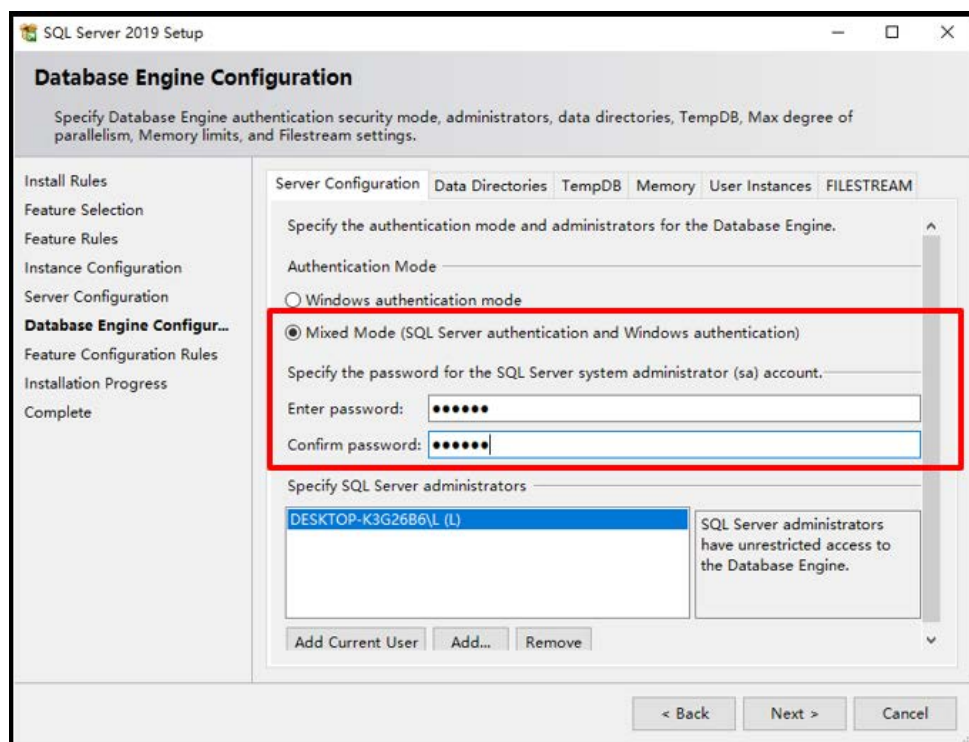
5. Choose the Default instance and go to the next step.



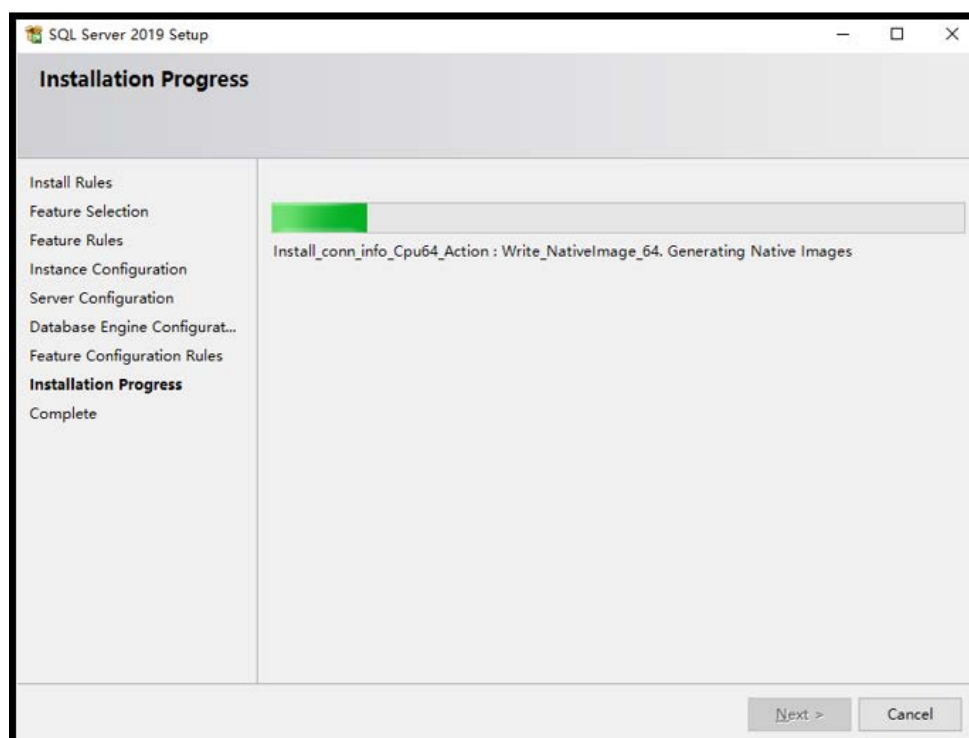
6. Proceed to the next step.



7. Choose Mixed Mode. Enter the password (123456) and confirm it in the specified fields. Keep the password confidential. Proceed to the next step.



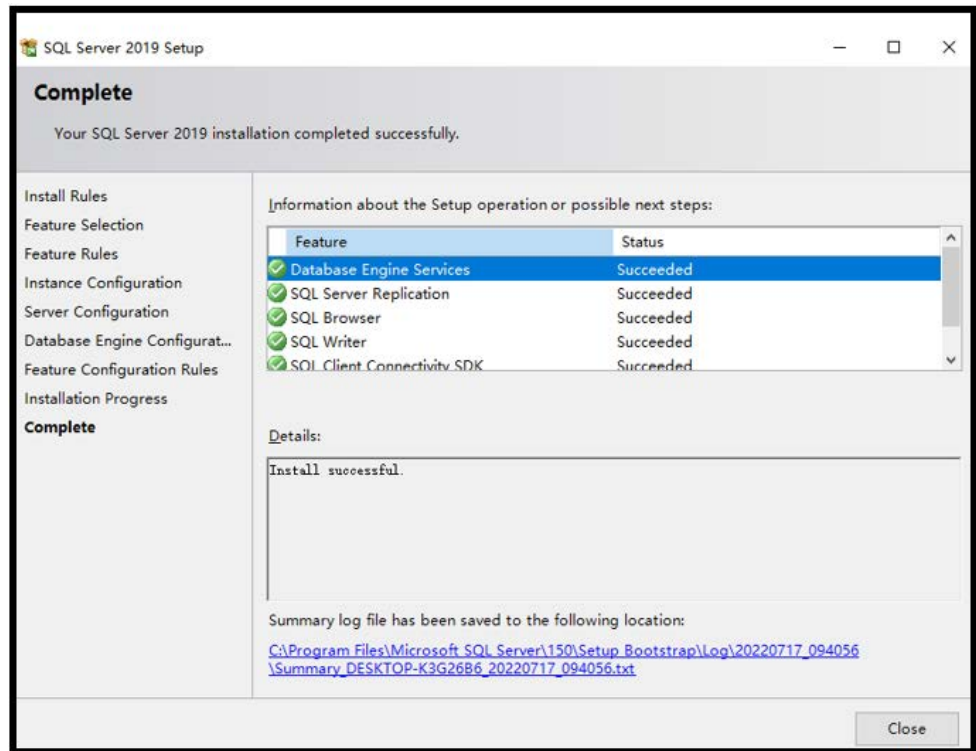
8. Go to the final installation step and wait for the process to complete.



3 Installation

3.5 HMI system installation

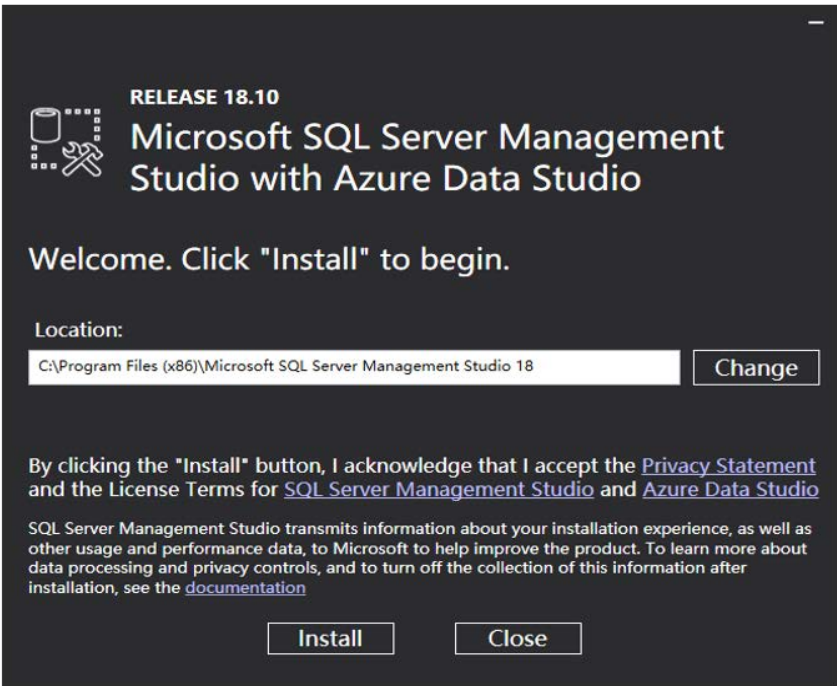
9. When you see this screen, the installation is complete. Click “Close.”



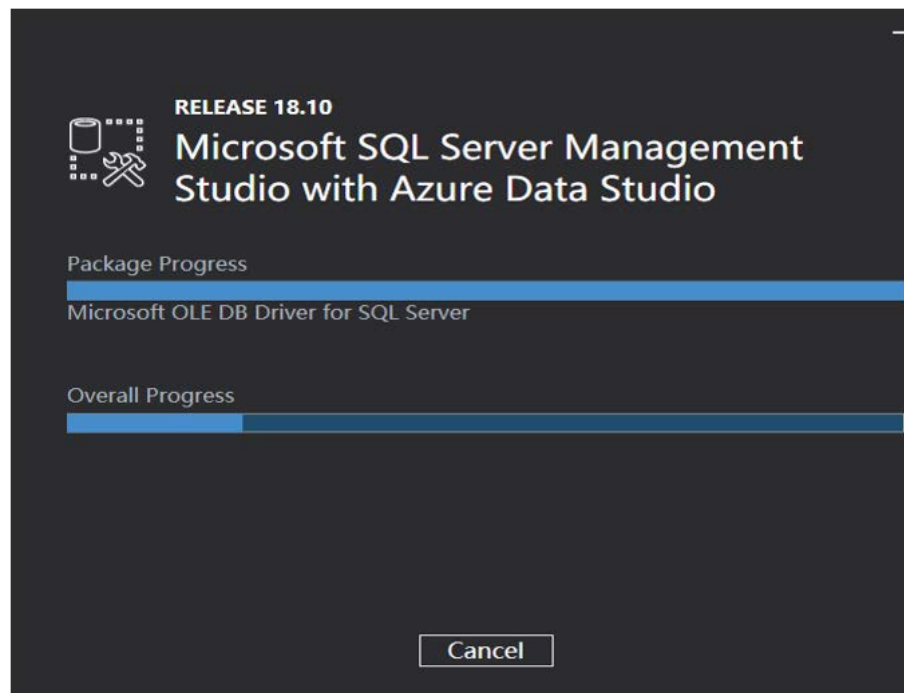
10. Find the SQL2019 folder in the installation package and double-click it.
Start the installation.



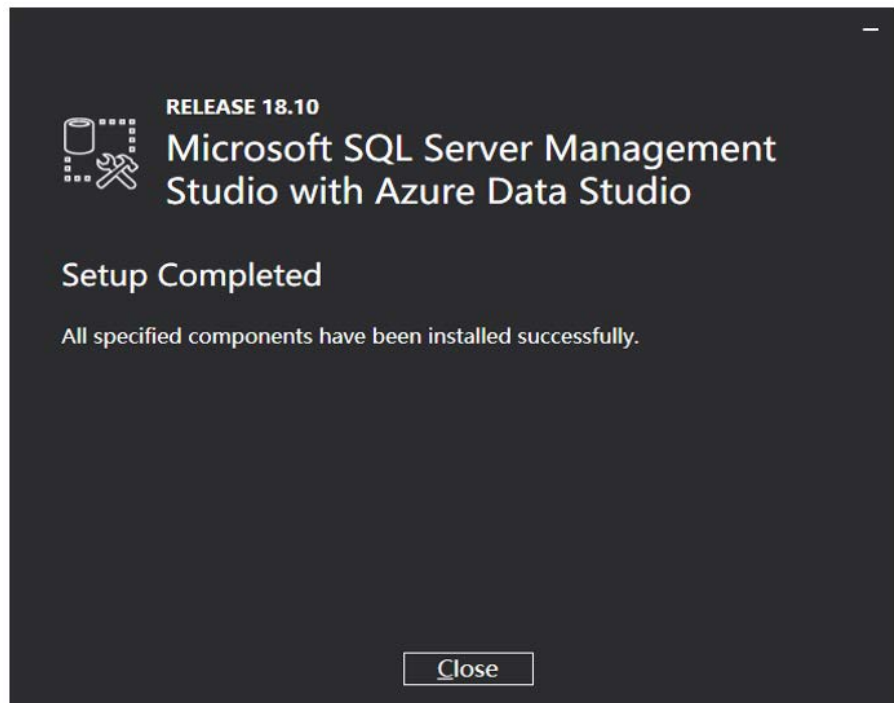
11. Click Install.



12. Enter the installation step and wait.



13. When the below screen appears, the installation is complete. Click "Close."

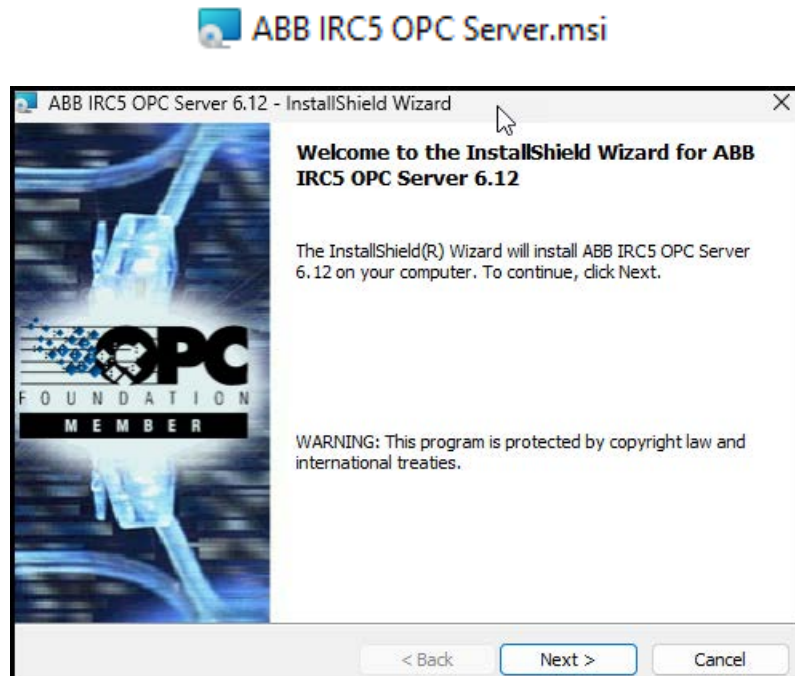


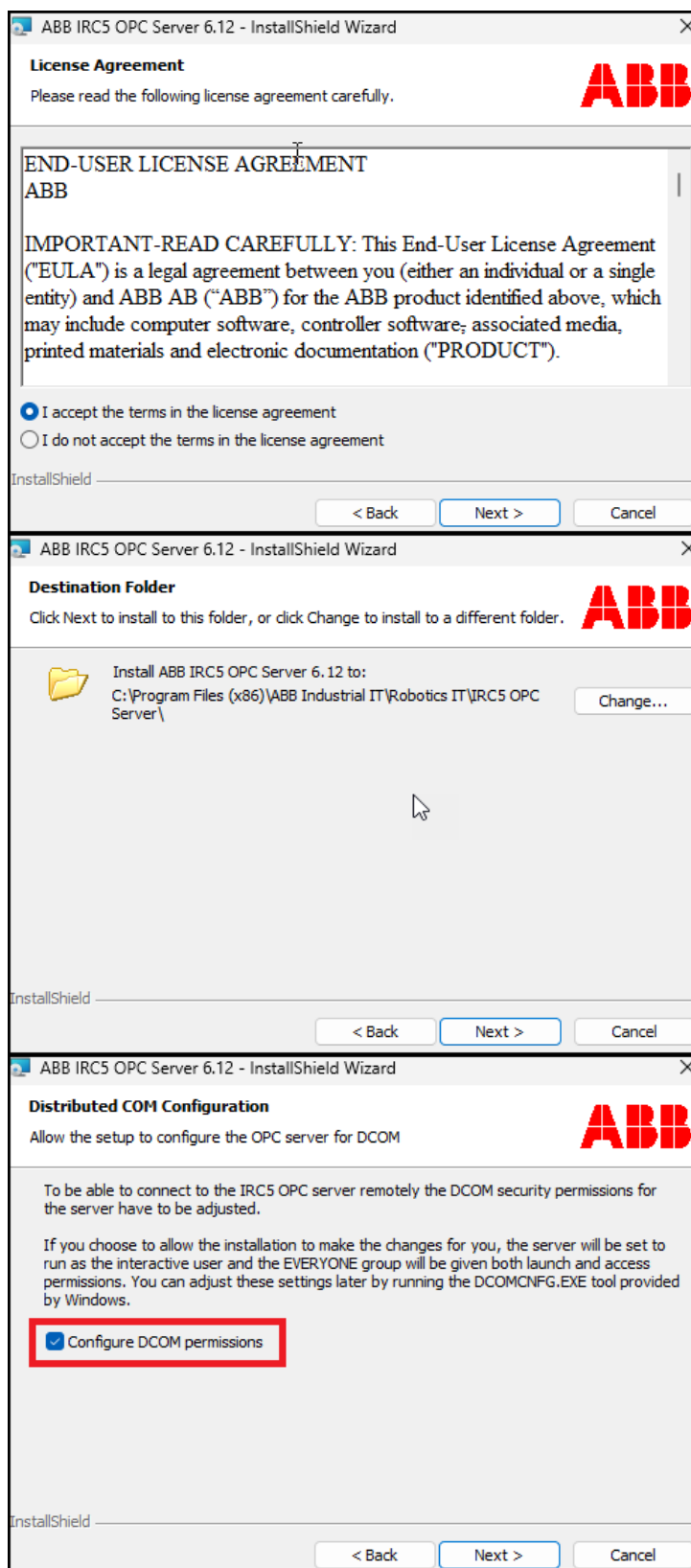
3 Installation

3.5 HMI system installation

3.5.8 IRC5P OPC Server Installation

In the installation package, find and click the IRC5OPCServer.6.12 folder.

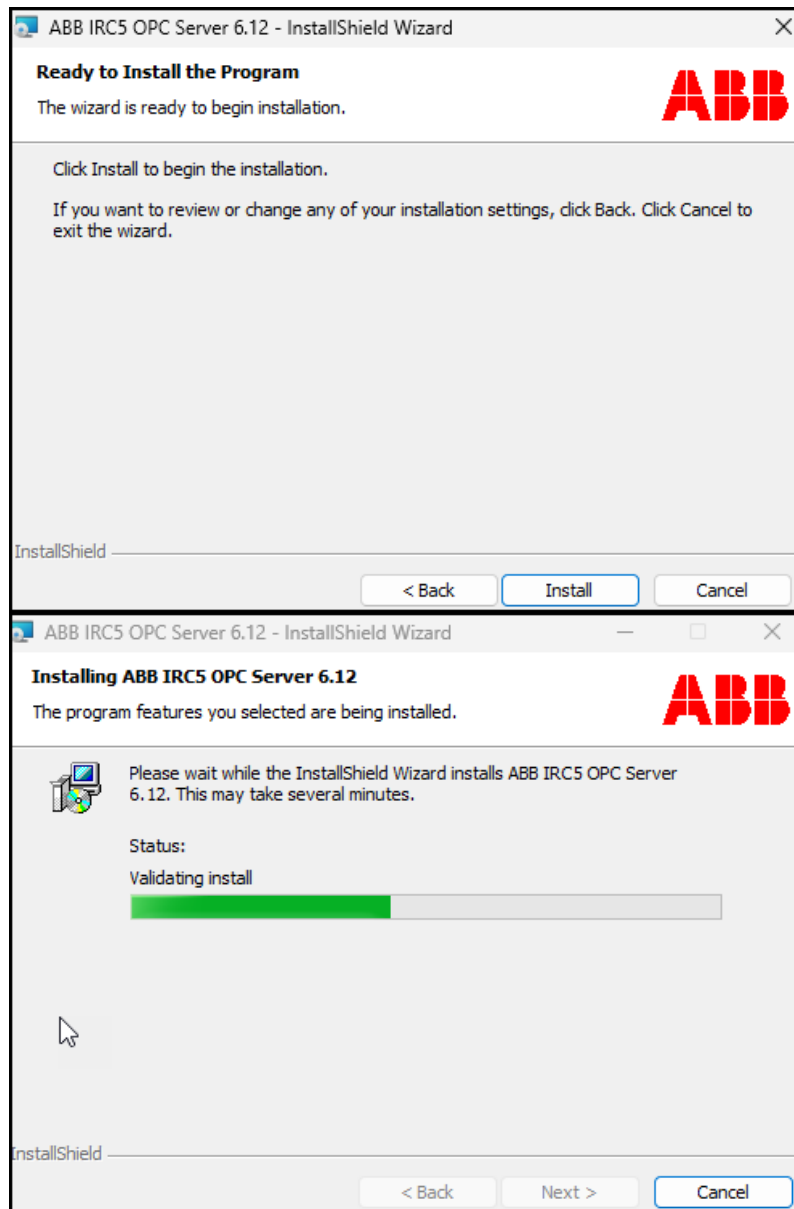




The figure above must be selected. Otherwise, the robot will not be connected.

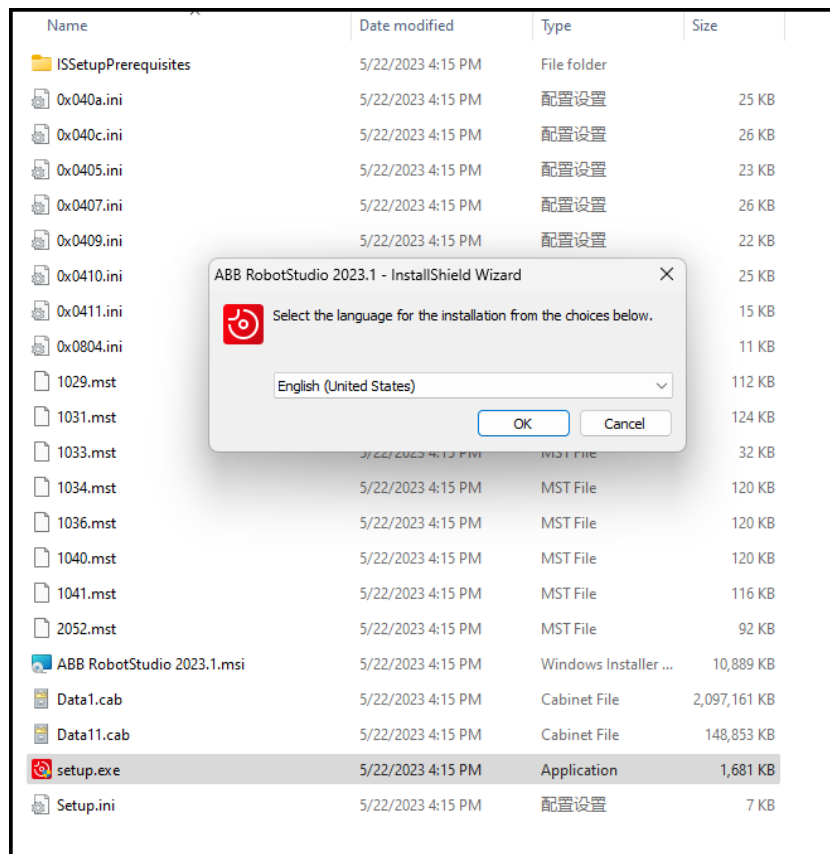
3 Installation

3.5 HMI system installation



3.5.9 RobotStudio Installation

1. Click “Install application” and choose the language version.



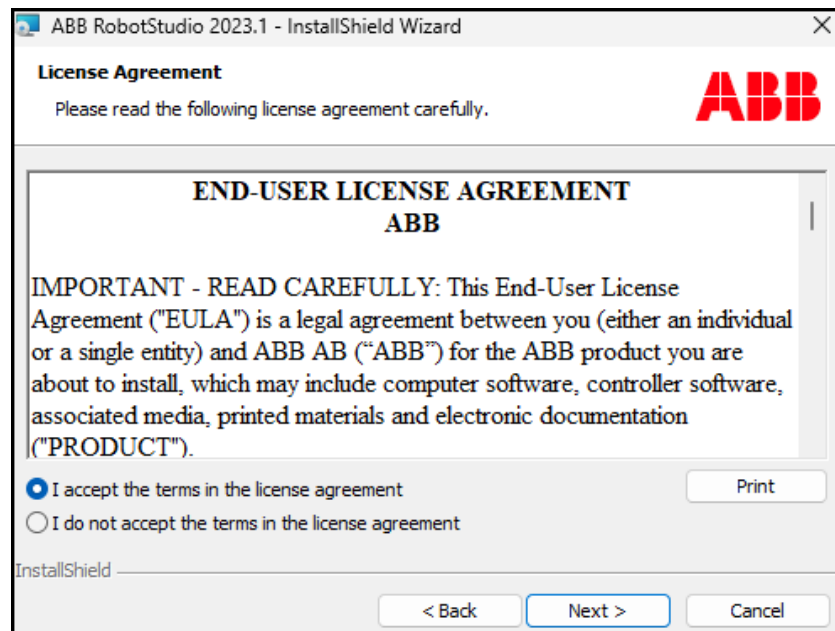
2. Click “Install”.



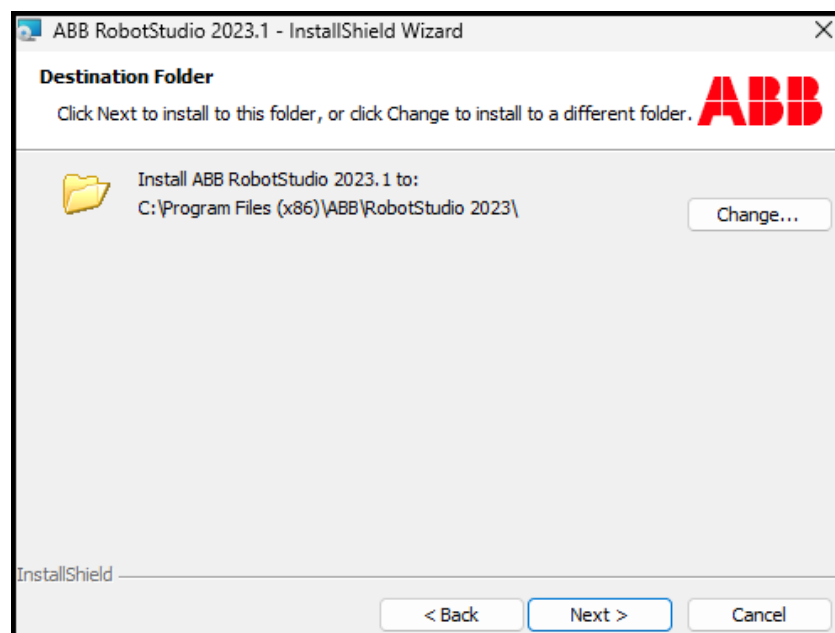
3 Installation

3.5 HMI system installation

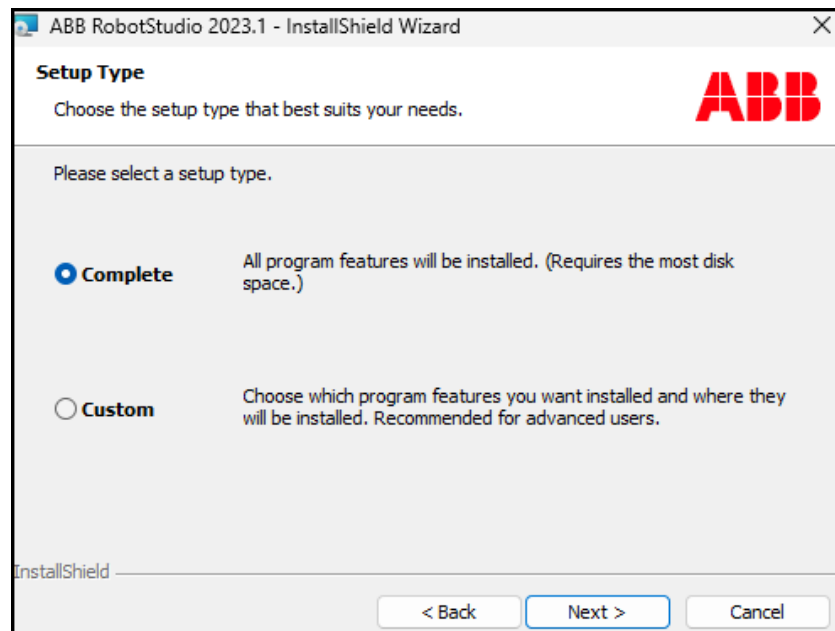
3. Click “Accept and next”.



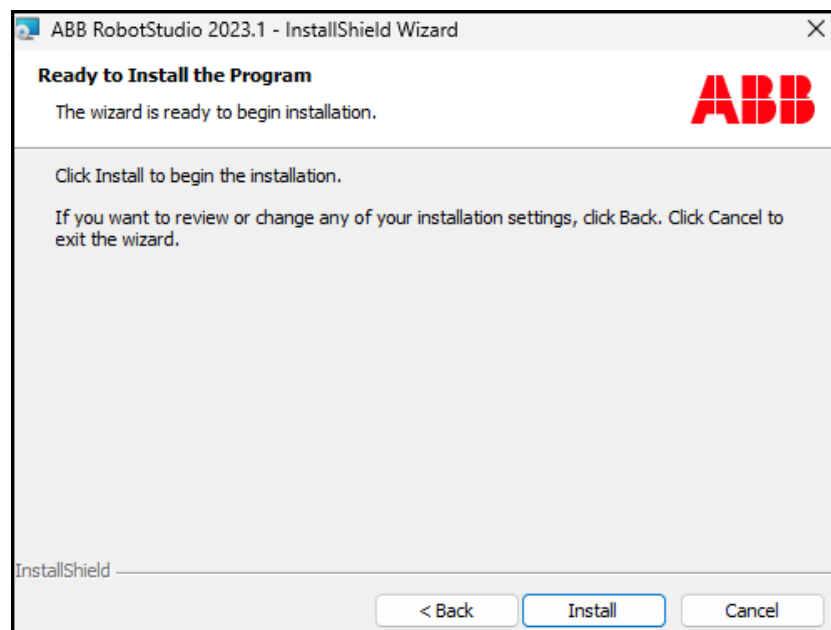
4. The default path stays the same. Click “Next”.



5. The installation type dialog box appears, showing that the next step is done.

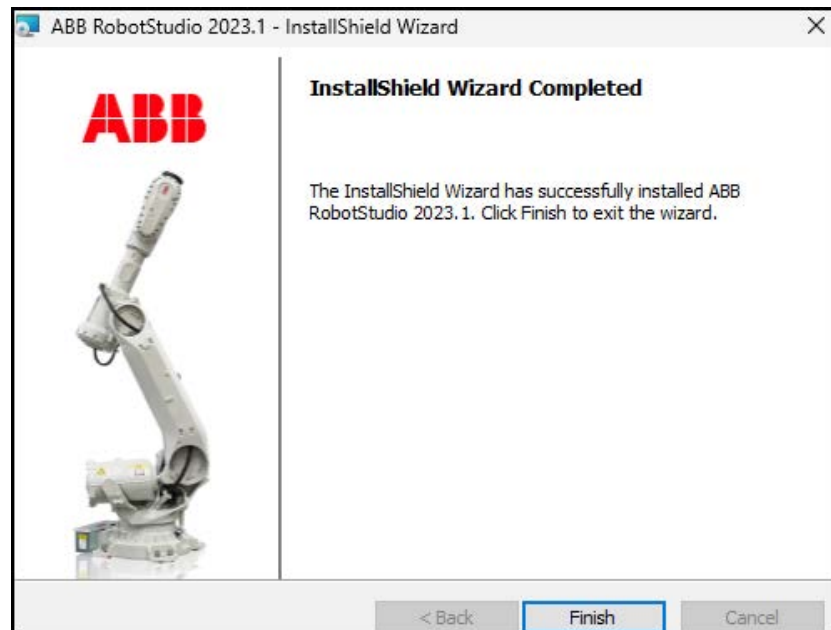


6. Click "Install Next" and wait until the installation finishes.



3 Installation

3.5 HMI system installation

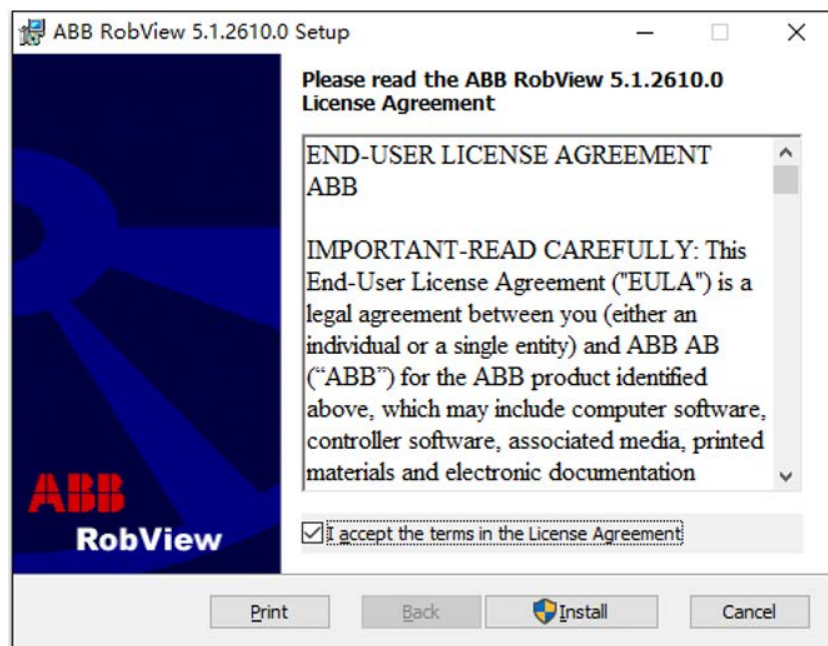


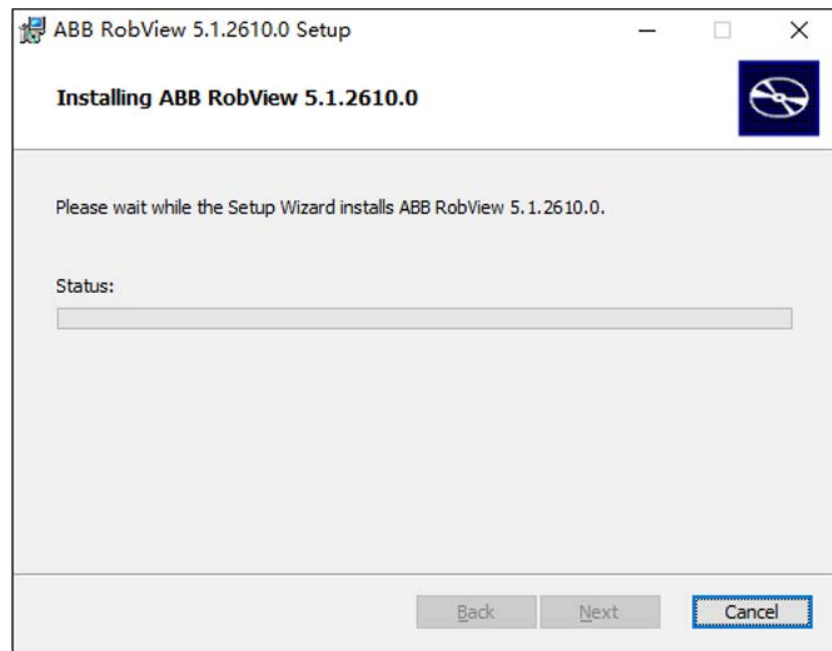
3.5.10 RobView5 Installation

1. Double-click the installation package RobView5 Setup.msi.

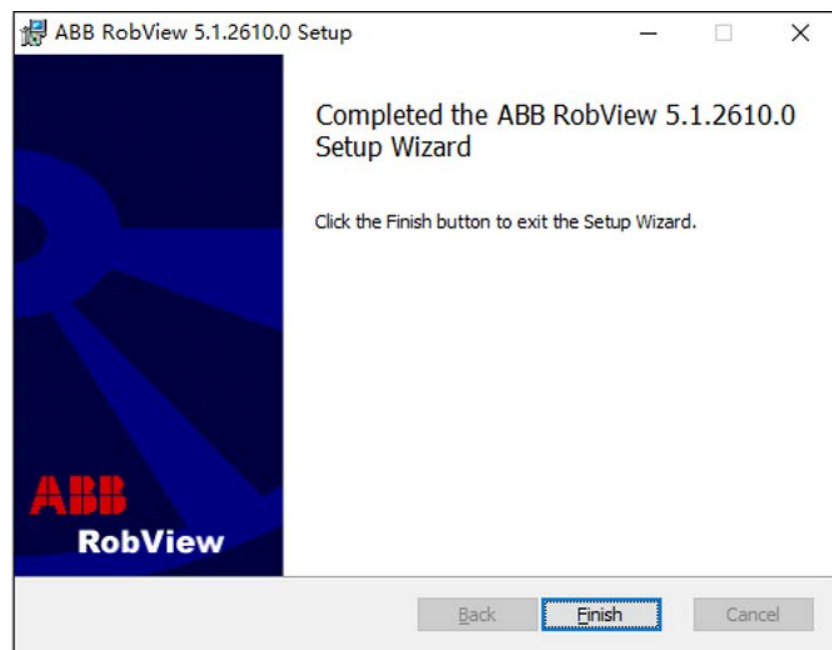
名称	修改日期	类型	大小
RobView5 Setup.msi	2014/6/11 19:29	Windows Install...	22,100 KB
setup.exe	2019/2/22 11:12	应用程序	643 KB

2. Select the checkbox and click “Install”.





3. Click the “Finish” button to end the installation.

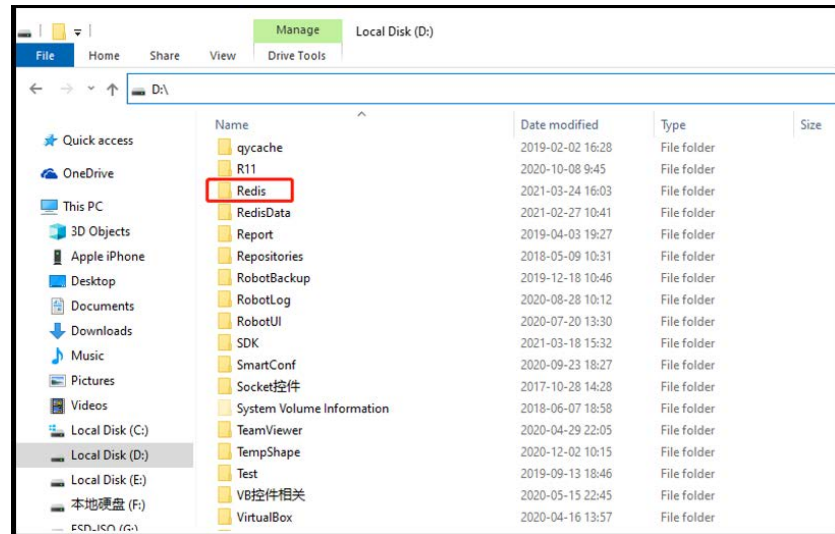


3 Installation

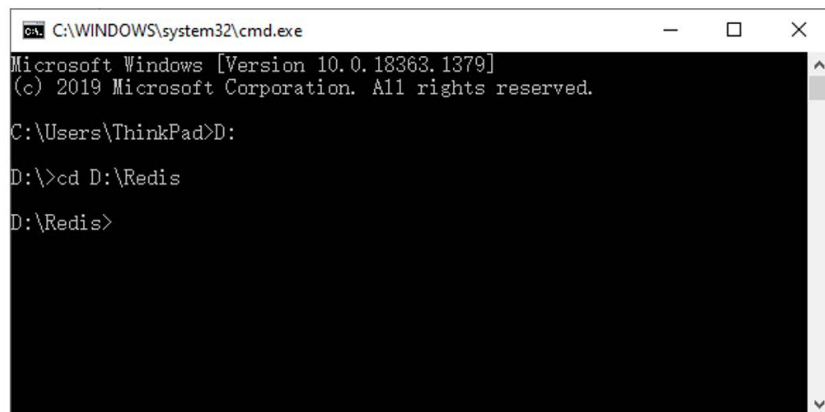
3.5 HMI system installation

3.5.11 Redis Database

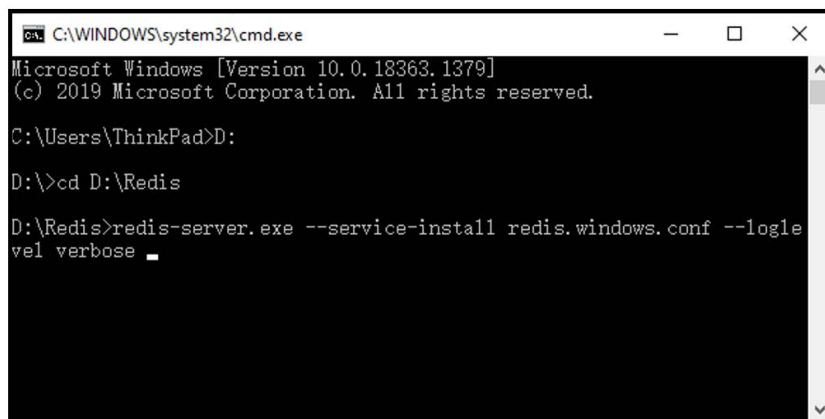
1. Save the Redis folder from the installation package to the D:\ drive.



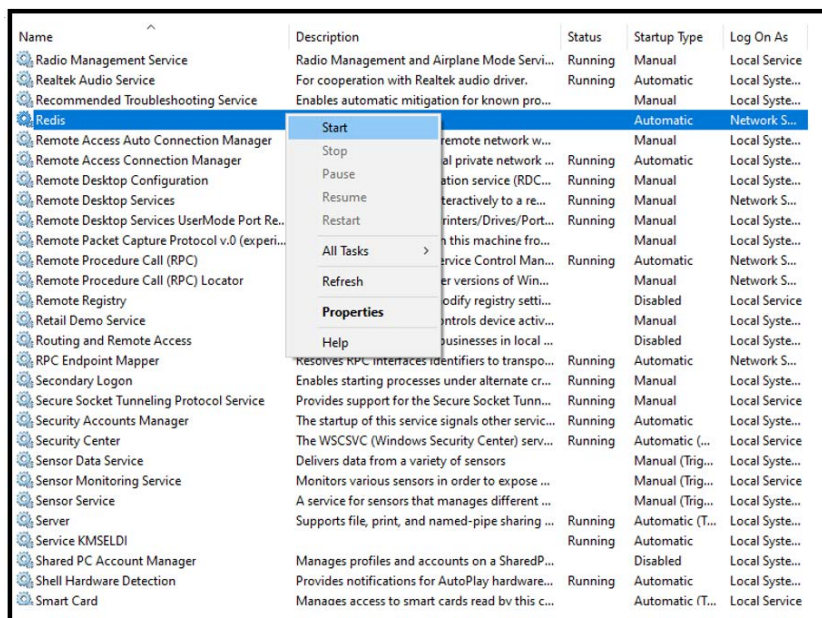
2. Open cmd.exe as an administrator and navigate to D:\Redis by the following the statement shown below.



3. Type this command to finish the database registration service:
redis-server.exe --service-install redis.windows.conf --loglevel verbose



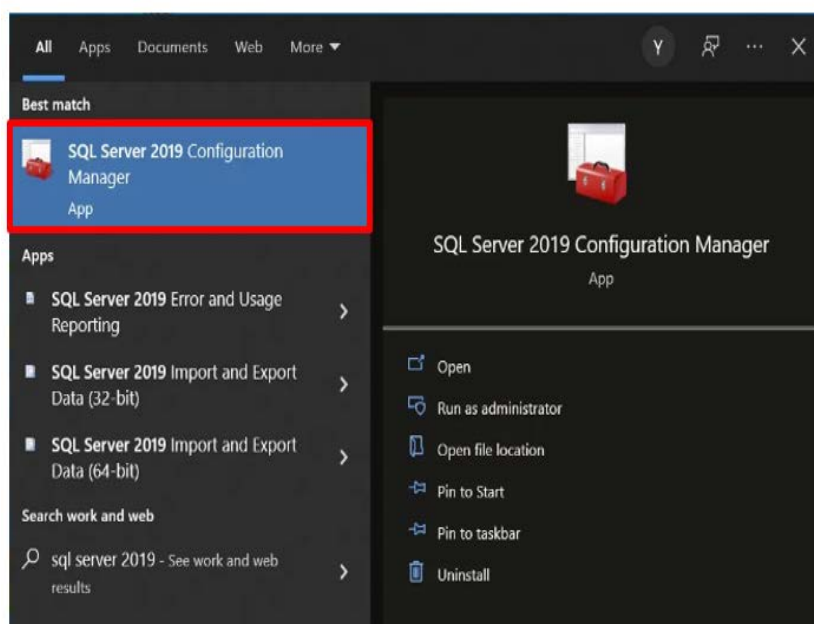
4. Open “Control Panel”, then “Administrative Tools”, and click on ”Services”. Find the Redis service, right-click it, and select “Start”.



3.5.12 SQL Server Management Configuration

Start the remote database service:

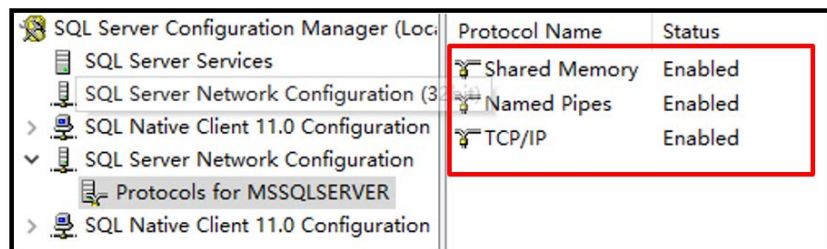
1. Open the configuration tool.



3 Installation

3.5 HMI system installation

2. Turn on all the settings shown in the figure.



Database SQL authentication (skip if already set up during database installation)

1. Open Microsoft SQL Server Management Studio 18.

Login with

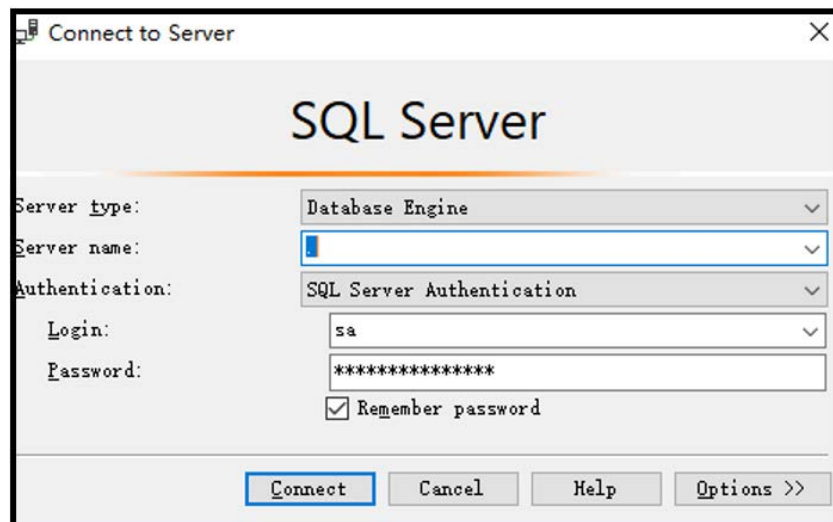
Username: sa

Password: 123456

Select “Remember password.”

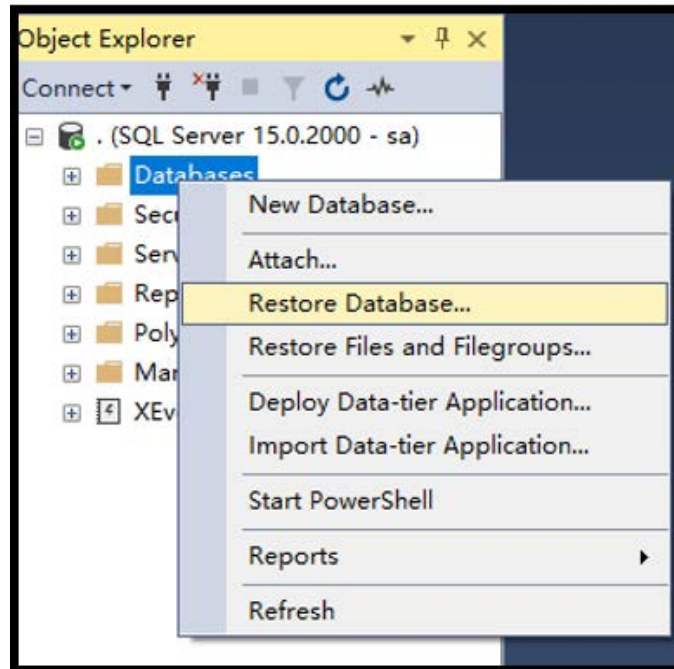
Click the “Connect” button to connect to the database.

Note: This default password should be different per project and customer requirements.

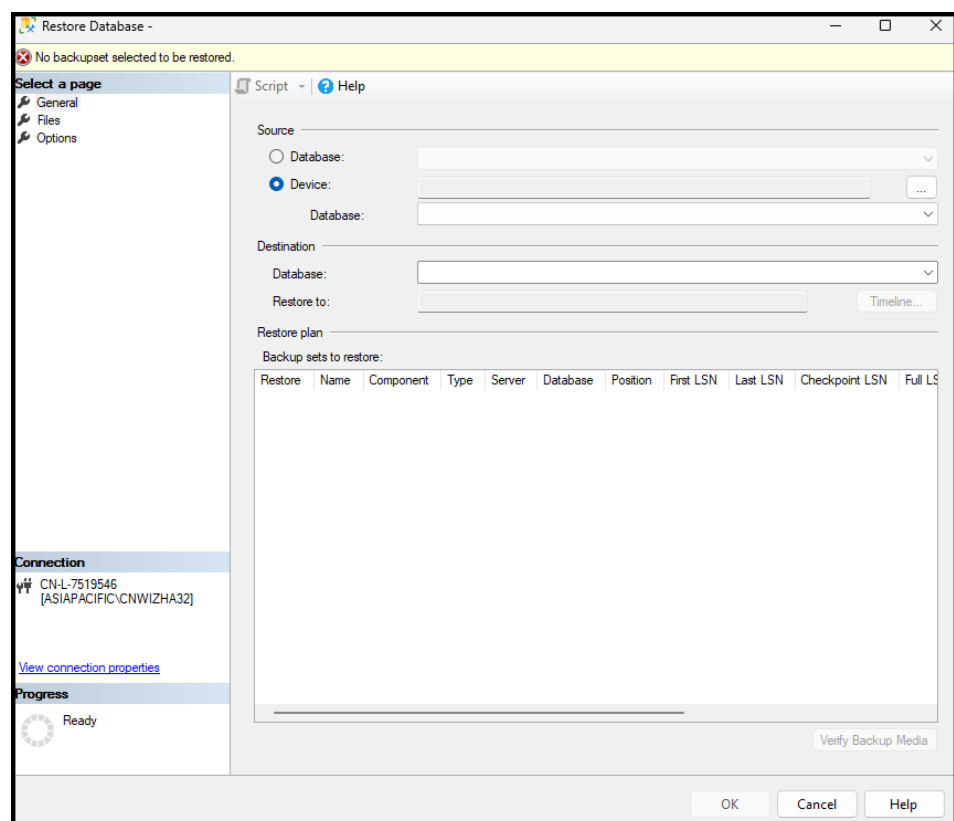


Restore the database file:

1. Right-click on the Database. Select “Restore Database”.



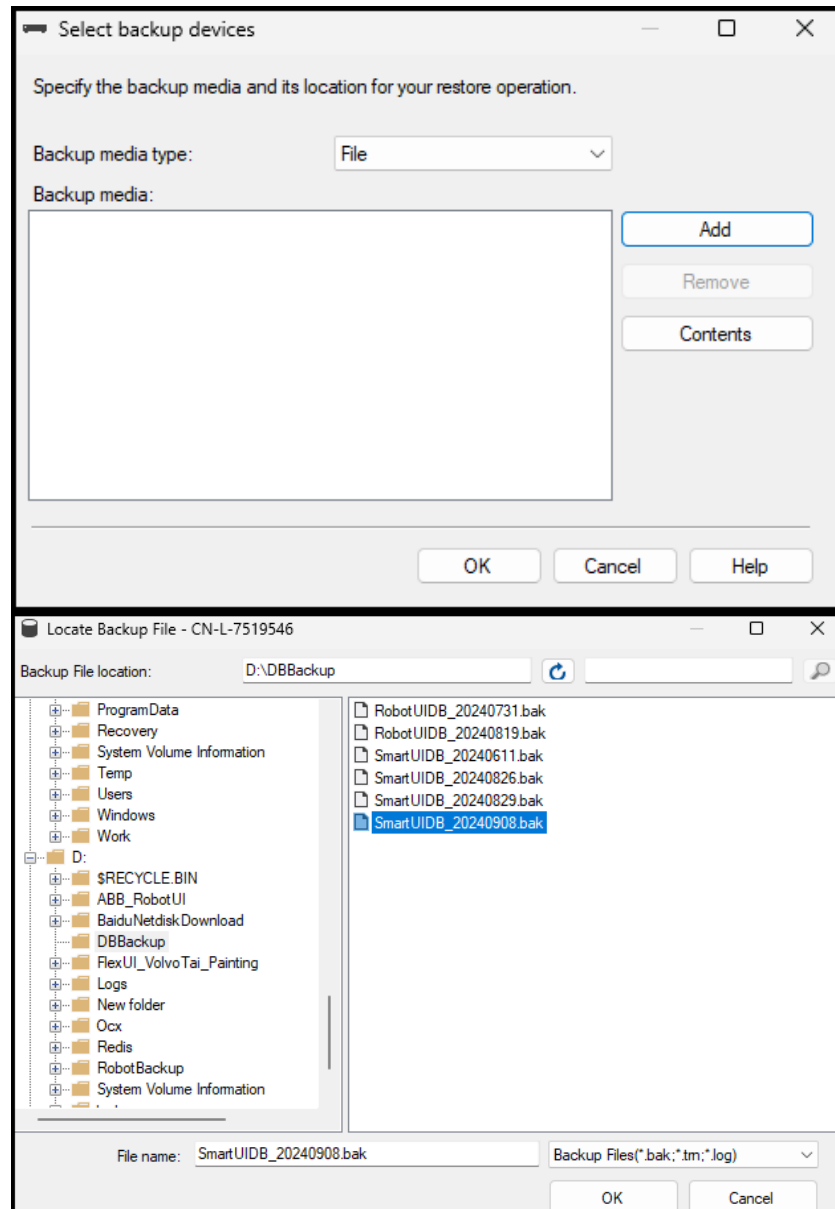
2. Choose Device (the second option).



3 Installation

3.5 HMI system installation

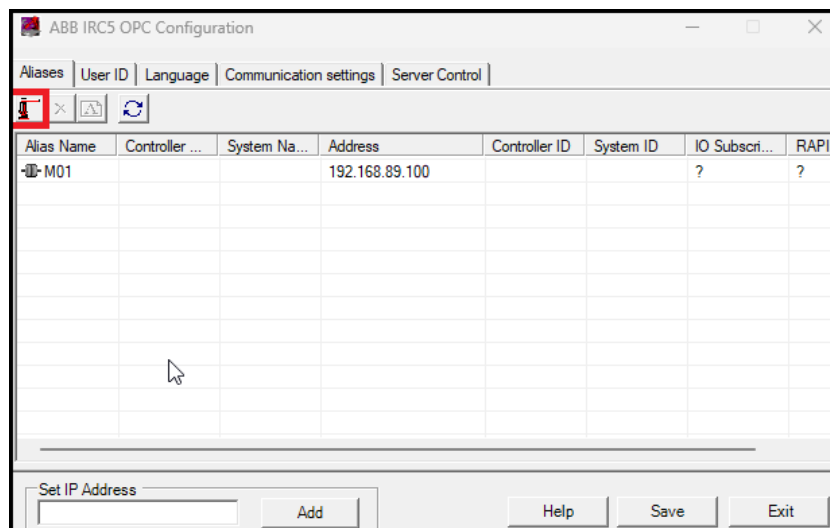
3. Click “Add” in the right side of the image. The path option is shown in the left side of the image. Select the database backup file, usually named SmartUIDB_XXXX.bak. Click “OK” to restore the database.



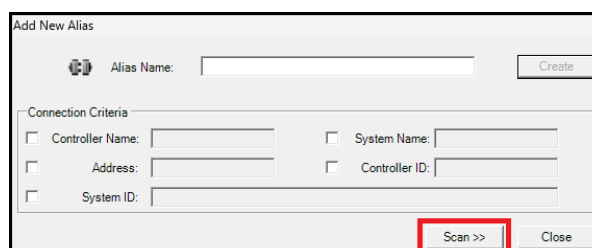
Note: If you want to run RobotUI on a local PC, not in a real project, change the configuration file "C:\ProgramData\SmartUI\Configure\Configure.xml", update the item “PC IP”, and change the value to 127.0.0.1, which means using the local PC database.

3.5.13 IRC5P OPC Configuration

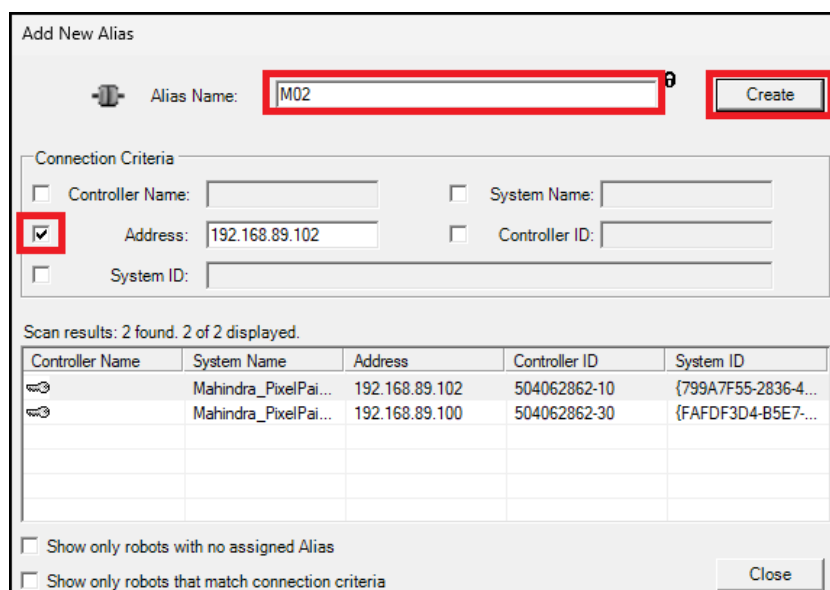
1. Open the ABB IRC5 OPC Configuration software. Click the small robot icon in the red box.



2. Click the Scan button to find robot configuration items in the network segment, as shown in the figure below.



3. Select an item from the list, check the box, and type the robot name in the Alias Name field.



Note :

Facing the entrance, the first painting robot on the left is M01.

The second painting robot on the right is M02.

The second painting robot on the left is M03, and so on.

The door opening robot is named after the painting robot.

The cover opening robot is named after the door opening robot.

Click the “create” button to add a robot.

After all robots are created, click the Save button on the start page to save and exit the software.

Example: 4 painting robots, 4 door opening robots, and 2 lid opening robots in the inner painting station.

Naming rule:

M01 (for R11 painting robot)

M02 (for R21 painting robot)

M03 (for R12 painting robot)

M04 (for R22 painting robot)

M05 (for D11 painting robot)

M06 (for D21 painting robot)

M07 (for D12 painting robot)

M08 (for D22 painting robot)

M09 (for H11 painting robot)

M10 (for H21 painting robot)

3.5.14 Program preparation

1. Place the ABB_RobotUI folder in the D:\ folder.

3.5.15 Authorization description

Regularly using Robot UI requires ABB to provide a legitimate dongle authorization. If there is no legitimate authorization, only a 7-day temporary approval is available, and the Robot UI function cannot be used commonly when the authorization expires.

4 Operation and safety

About this chapter

This chapter gives an overview of the station operation modes which are specified and what the purpose is used for. Further, it focuses on safety and how the PLC reacts to certain situations and external inputs.

The following sections are included:

['Introduction'](#) on page 92

['Operation panel'](#) on page 92

['Operation modes'](#) on page 101

['Station light indicator'](#) on page 105

['Safety system'](#) on page 105

4.1 Introduction

“The term ‘Operation and Safety’ primarily refers to the mode of operation and the safety devices implemented in robotic applications.”

4.2 Operation panel

The following sub-sections are included in this section:

['Operation panel overview'](#) on page 92

['Operation panel layout'](#) on page 93

['Operation panel/Button description'](#) on page 93

4.2.1 Operation panel overview

Overview

Paint Controls necessitate a specific set of push buttons and key switches, irrespective of the operator panel being used, to function immediately upon setup.

The following chapters provide an overview of the required setup for each application type.

The following abbreviations are used:

Abbreviation	Description
SB	Push button
SBH	Illuminated Push button
SK/SW	Key Switch

4.2 Operation panel

Each station comes with an operation desk, and the following is the panel layout of the operation desk.

```

graph TD
    subgraph Left_Flow [Left Logic]
        SM1((Station Mode)) --> HV1([HV])
        SM1 --> PR1([Paint/Robot])
        SM1 --> WP1([Whole Process])
        HV1 --> RM1{Robot Mode}
        PR1 --> RM2{Robot Mode}
        WP1 --> RM3{Robot Mode}
        RM1 -- "T1 Manual 25%" --> B1[No robot Movement  
Cycled Command OFF  
No Paint Command  
HV Test While Pushing  
HV authorization button  
HV Indication  
No Area Protection]
        RM2 -- "T1 Manual 25%" --> B2["1. Robot manual mode 25%  
2. Movement of the robot by TPU  
3. Cycle command OFF  
4. No area Protection"]
        RM3 -- "Auto" --> B3["MMI master granted  
1. Robot Position test  
2. Robot Paint process Test  
3. Securities actives"]
    end

    subgraph Right_Flow [Right Logic]
        SM2((Station Mode)) --> RM4{Robot Mode}
        RM4 -- "Auto" --> B4["Automatic Mode simulated  
- Without conveyor  
- Without Paint  
- Without HV  
Securities activities"]
        RM4 -- "Deep" --> B5["End of Production Deep Cleaning  
No power supply when robot is in calibration  
position no robot or paint action available  
Securities Deactivated"]
        RM4 -- "ON" --> B6["End of production Cleaning  
Bell and circuit cleaning  
No cycled commands  
System waits for a new body  
Securities activate"]
    end

```

The following sub-sections are included in this section:

- 'Emergency stop button' on page 94
- 'Control voltage switch' on page 94
- 'Control voltage two-position on/off switch' on page 94
- 'Conveyor bypass switch' on page 95
- 'Two position key switch' on page 95
- 'Conveyor run button' on page 96
- 'Station work mode switch' on page 96
- 'Three position key switch' on page 96
- 'Robot mode button' on page 97
- 'Robot mode key switch' on page 97

'Resume' on page 98

'Station mode button' on page 99

'Cycle start button' on page 100

'HV enable button' on page 100

'Lamp test button' on page 100

'Fault reset button' on page 101

Emergency stop button

The emergency stop button is utilized to manage emergency situations by ceasing movement and the operation of the robot. This action will initiate an emergency stop for the robot.

Operation panel layout

 E-STOP SE1	Control Voltage On	Station Auto/Manual	Conv. On/Off	Spare	Spare	Start On	Robot Mode Auto/T1	HV. Enable	MAINT WORK CLEAN
	控制电压开	站自动/站手动	输送使能	备用	备用	启动	机器人自动/T1	高压使能	
	SBH10	SBH11	SBH11			SBH12	SK31	SBH13	SW100
	Control Voltage Off	Lamp Test	Conv. Bypass OFF	Spare	Spare	Start Off	FaultReset	HV.Auth OFF ON	Robot Select(Optional)
	控制电压关	灯测试	输送旁通	备用	备用	停止	故障复位	高压操作	
	SBH20	SB40	SK32			SB41	SBH21	SK34	16

Control voltage switch

The control voltage switch is displayed in the operating panel layout as shown in the figure below.

 E-STOP SE1	Control Voltage On	Station Auto/Manual	Conv. On/Off	Spare	Spare	Start On	Robot Mode Auto/T1	HV. Enable	MAINT WORK CLEAN
	控制电压开	站自动/站手动	输送使能	备用	备用	启动	机器人自动/T1	高压使能	
	SBH10	SBH11	SBH11			SBH12	SK31	SBH13	SW100
	Control Voltage Off	Lamp Test	Conv. Bypass OFF	Spare	Spare	Start Off	FaultReset	HV.Auth OFF ON	Robot Select(Optional)
	控制电压关	灯测试	输送旁通	备用	备用	停止	故障复位	高压操作	
	SBH20	SB40	SK32			SB41	SBH21	SK34	16

Control voltage two-position on/off switch

This switch is utilized as a precondition for enabling or disabling the high voltage (HV) function. When switched to 'off', the following actions occur:

- The robot's movement is halted.
- The release of high voltage is reset.
- The process (bear turbine/pumps/valves) is turned off.
- The station maintains its previously active mode, and the cycle can resume after performing the restart procedure.

4 Operation and safety

4.2 Operation panel

Conveyor bypass switch

The conveyor bypass switch is displayed in the operating panel layout as shown in the figure below.

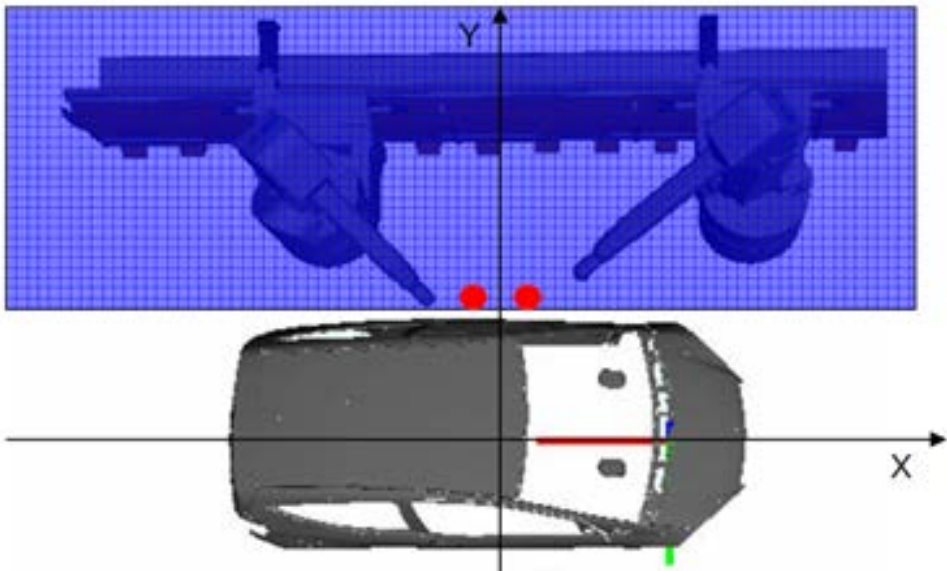
 E-STOP SE1	Control Voltage On	Station Auto/Manul	Conv. On/Off	Spare	Spare	Start On	Robot Mode Auto/T1	HV. Enable	MAINT	WORK	CLEAN
	控制电压开	站自动/站手动	输送使能	备用	备用	启动	机器人自动/T1	高压使能			
	SBH10	SBH11	SK32			SBH12	SK31	SBH13			
	Control Voltage Off	Lamp Test	Conv. Bypass Off	Spare	Spare	Start Off	FaultReset	HV.Auth OFF	SW100		
	控制电压关	灯测试	输送旁通	备用	备用	停止	故障复位	高压操作	Robot Select(Optional)		
	SBH20	SB40	SK32			SB41	SBH21	SK34			
											16

Two position key switch

ON position - Bypass conveyor

OFF position - Enable conveyor

- To bypass the conveyor, the station must switch to manual mode, and the robot does not control the conveyor. The robot must maintain a safe position. The robot's automatic movements and actions are prohibited. If the conveyor is operational, the robot must remain in a safe position.
- During the conveyor bypass, manual movement of the robot and manual process testing is permitted, but the robot must stay in a safe position. If the robot is disabled and immobile, the HMI can enforce a safe position.
- There are settings for axis limits and safe move zones to keep the robot within a restricted position. The HMI can only enforce the home position, but even in this situation, the robot remains within the safe zone due to the axis limit.



- To ensure the robot's safety position, it is only necessary to verify the Y coordinate of the robot's position using the SafeMove system. For more information about safety position, see interlock chapter. This function facilitates the smooth operation of the conveyor.

Conveyor run button

 E-STOP SE1	Control Voltage On	Station Auto/Manual	Conv. On/Off	Spare	Spare	Start On	Robot Mode Auto/T1	HV. Enable	MAINT WORK CLEAN
	控制电压开	站自动/站手动	输送使能	备用	备用	启动	机器人自动/T1	高压使能	
	SBH10	SBH11	SBH14			SBH12	SK31	SBH13	SW100
	Control Voltage Off	Lamp Test	Conv. Bypass OFF	Spare	Spare	Start Off	FaultReset	HV.Auth OFF	Robot Select(Optional)
	控制电压关	灯测试	输送旁通	备用	备用	停止	故障复位	高压操作	
	SBH20	SB40	SK32			SB41	SBH21	SK34	
									16

It is used to release conveyors, which can be released in several conditions.

Explanation of the button light:

- The light remains off if the conveyor button is not engaged.
- The light flashes rapidly if the station conveyor is not engaged.
- The light flashes slowly if any other station conveyor is not engaged.
- The light stays on continuously if all conditions are met.

Station work mode switch

 E-STOP SE1	Station Auto	Conv. On/Off	L.B. Entrance bypass OFF	Motor On	Start On	Robot Auto Mode	Robot T1 Mode	HV. Enable	Lamp Test	Power On	MAINT WORK CLEAN
	站自动	输送运行	入口光栅旁通关	马达使能	启动	机器人自动模式	机器人T1模式	高压使能	灯测试	电源指示	
	SBH10	SBH11	SK31	SBH12	SBH13	SBH14	SBH15	SB43	HB10		SW100
	Station Manual	Conv. Bypass ON	L.B. Exit bypass ON	Motor Off	Start Off	Robot Mode Disable Enable	Spare	HV. Operate OFF	FaultReset	Spare	Robot Select(Optional)
	站手动	输送旁通关	出口光栅旁通关	马达断开	停止	机器人模式使能禁止使能	预留	高压操作关	故障复位	预留	
	SBH20	SK32	SK33	SB41	SB42	SK34		SK35	SBH21		

Three position key switch

Maintain position - Maintenance mode

Clean position - Clean Mode

Work position - Working Mode

The switch allows the selection of three modes: maintenance, work, and cleaning. It is crucial to adhere to safety protocols before entering the booth.

- Maintenance Mode:** In this mode, the robot can be manually moved. If the FlexPendant is to be used, the system must be switched to maintenance mode and T1 mode. This is independent of whether the station is in manual or automatic mode.

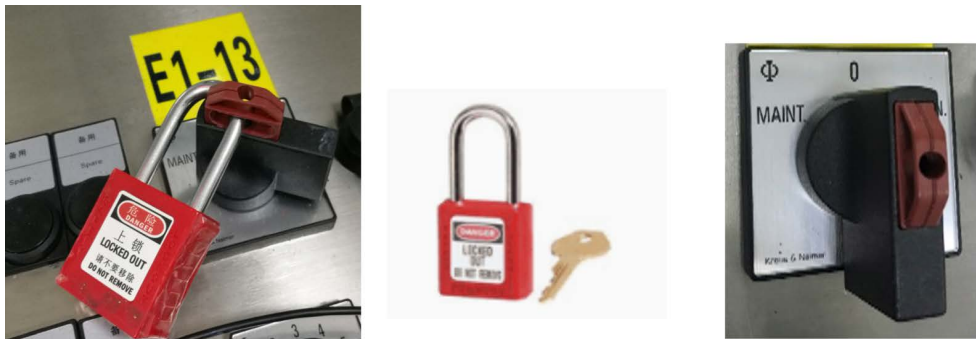
4 Operation and safety

4.2 Operation panel

- **Clean Mode:** This mode is primarily for cleaning the bell and atomizer. Robot movement is prohibited, and low-shape air is released from the shaping air ring. The operator will secure it before entering the booth.
- **Work Mode:** In this mode, the robot can be run in auto mode, and production can be initiated by pressing ‘Cycle Start’. If the station is operational, ‘Stop Cycle’ must be pressed.

Once the station has stopped, the switch can be turned to another function, such as maintenance or cleaning.

The robot can be stopped by turning off the motor or brake, although this is not recommended for the brake system. Alternatively, the conveyor run can be used to stop the robot.



Robot mode button

 E-STOP SE1	Control Voltage On	Station Auto/Manual	Conv. On/Off	Spare	Spare	Start On	Robot Mode Auto/T1	HV. Enable	MAINT WORK CLEAN
	控制电压开	站自动/站手动	输送使能	备用	备用	启动	机器人自动/T1	高压使能	
	SBH10	SBH11	SBH11			SBH12	SK31	SBH13	SW100
	Control Voltage Off	Lamp Test	Conv. Bypass OFF	Spare	Spare	Start Off	Fault/Reset	HV Auth OFF ON	Robot Select(Option)
	控制电压关	灯测试	输送旁通	备用	备用	停止	故障复位	高压操作	
	SBH20	SB40	SK32			SB41	SBH21	SK34	
									16

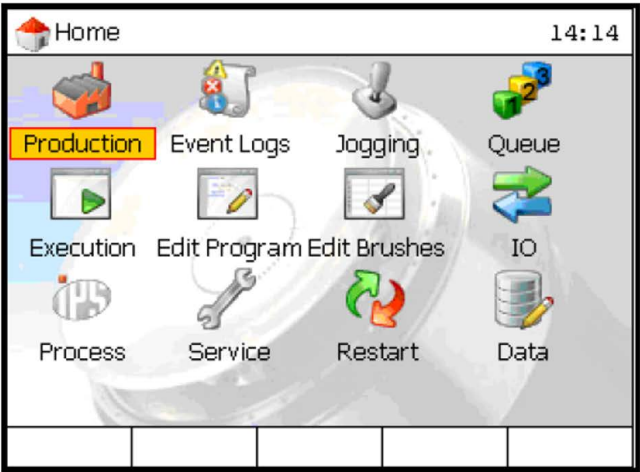
Robot mode key switch

The Robot mode button allows the selection of two robot movement modes: “Robot Auto” and “Robot T1”.

- **Robot T1 Mode:** In this mode, the robot moves at a teaching speed of 25%, which is less than 250 mm/s. Only one robot is permitted to move at a time.
- **Robot Auto Mode:** In this mode, the operator can test robot positions and other functions using the HMI. The entry protection device is activated, and both process and HV tests are allowed. The FlexPendant can be used to check the robot’s status, change the brush ratio, and adjust the movement speed ratio.

Resume

Resume matrix says which menu can be used in different robot modes.



TPU menu	T1 mode	Auto mode	
		Station auto	Station manual
Production	x	○	x
Event logs	x	x	x
Jogging	x		
Queue	x	○	x
Execution	x	x	x
Edit program	x		
Edit Brushes	x		
IO	x		
Process	x		
Service	x	x	x
Restart	x		x
Data	x		

- - Limit access
- x - Full access

4.2 Operation panel

Station mode button

 E-STOP SE1	Control Voltage On	Station Auto/Manual	Conv. On/Off	Spare	Spare	Start On	Robot Mode Auto/T1	HV. Enable	MAINT WORK CLEAN
	控制电压开	站自动/站手动	输送使能	备用	备用	启动	机器人自动/T1	高压使能	
	SBH10	SBH11	SBH11			SBH12	SK31	SBH13	
	Control Voltage Off	Lamp Test	Conv. Bypass OFF ON	Spare	Spare	Start Off	FaultReset	HV Auth OFF ON	SW100 Robot Select(Optional)
控制电压关	灯测试	输送旁通	备用	备用	停止	故障复位	高压操作		
SBH20	SB40	SK32			SB41	SBH21	SK34		

-
- 99

Cycle start button

Cycle start button is used to facilitate the movement of the robot.

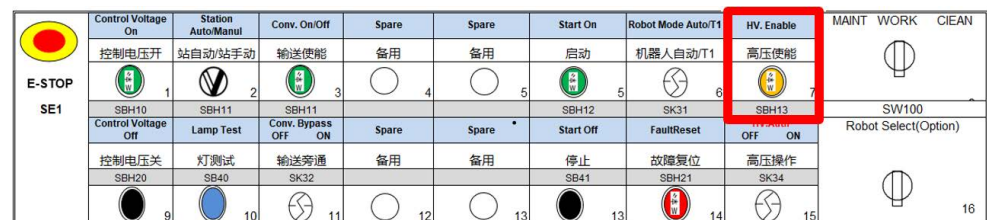


The button has an integrated light that indicates the status of the robot's movement:

- **Light OFF:** Indicates that no robot movement is enabled.
- **Light Flashing:** Signifies that one or more robot movements are disabled due to a robot error.
- **Light ON:** Denotes that all robot movements are enabled.

HV enable button

The HV enable button is used to control the high voltage (HV) of the robot.

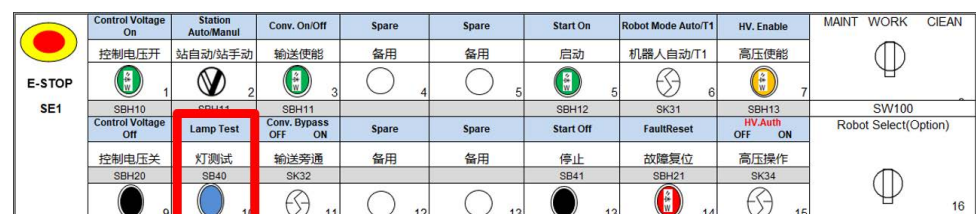


- **Light ON:** If the HV of all robots is enabled, the indicator light remains on.
- **Light OFF:** If the HV of all robots is off, the indicator light is turned off.
- **Light Flashing:** If the HV of at least one, but not all, enabled robots is on, the indicator light flashes.

Note: The HV ON status is pre-enabled. After each cycle, once the painting of each car body is completed, the HV current value is reset to 0.

Lamp test button

The lamp test button is used to test all lamps of the panel. All button lights will turn on if this button is pushed.



4 Operation and safety

4.3 Operation modes

Fault reset button

Fault reset button is used to reset all faults.

 E-STOP SE1	Control Voltage On	Station Auto-Manul	Conv. On/Off	Spare	Spare	Start On	Robot Mode Auto/T1	HV. Enable	MAINT WORK CIEAN
	控制电压开	站自动/站手动	输送使能	备用	备用	启动	机器人自动/T1	高压使能	
	SBH10	SBH11	SBH11			SBH12		SBH13	SW100
	Control Voltage Off	Lamp Test	Conv. Bypass OFF	Spare	Spare	Start Off	FaultReset	HV.Auth OFF ON	Robot Select(Optional)
	控制电压关	灯测试	输送旁通	备用	备用	停止	故障复位	高压操作	
	SBH20	SB40	SK32			SB41	SBH21	SK34	

Fault reset button contains a light that operates as follows:

- The light flashes, if a fault is detected.
- The light remains off, if no faults are present.

4.3 Operation modes

The following sub-sections are included in this section:

- 'Station mode' on page 101
- 'Station production mode' on page 103
- 'Robot operating mode' on page 104

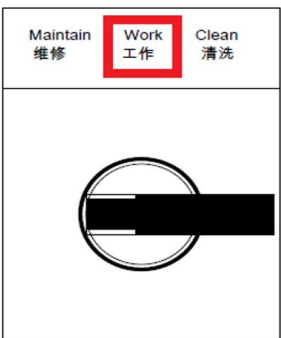
4.3.1 Station mode

The operating modes of the station are chosen via the station control panel. These modes include:

- 'Work mode' on page 101
- 'Maintenance mode' on page 102
- 'Cleaning mode' on page 102

4.3.1.1 Work mode

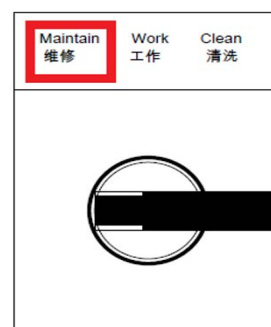
Production can only commence when the switch is in this position. Additionally, the station mode must be set to either auto or manual mode.



4.3.1.2 Maintenance mode

Maintenance mode is activated using a lockable rotary switch. In this mode, the robot can be moved, and the operator has the ability to replace parts, inspect the hardware, and program trajectories.

Functional testing of process components such as valves, pumps, and controllers is also possible. Manual control of valves and setting setpoints for the dosing pump and air volume controller can be done. When the robot is in T1 mode, all doors in the zone and the light grid pairs are deactivated. Typically, it is locked with an operator lock to prevent unauthorized operation.



4.3.1.3 Cleaning mode

The cleaning mode is activated using a lockable rotary switch. When in this mode, the robot can be cleaned and serviced. Once the rotary switch is set to the cleaning position, the power to the drives, valves, high voltage facilities, and air volume controllers is turned off, except for a minimal airflow through the shaping air nozzles to prevent material from clogging the orifices. The safety measures for personnel are deactivated and the conveyor equipment is turned off.



4 Operation and safety

4.3 Operation modes

4.3.2 Station production mode

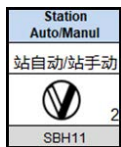
The following sub-sections are included in this section:

'Station Auto Mode' on page 103

'Ghost Vehicle Administration' on page 103

'Station Manual Mode' on page 103

4.3.2.1 Station Auto Mode



The robot is only allowed to operate in automatic mode. Once the robot is switched to T1, the permission for the robot to move is withdrawn. All safety measures are in effect, and entry into the cell is prohibited.

Neither movement nor time programs, such as fixed position rides or purging programs, can be initiated at the station control panel or vision system. Atomizers can be activated or deactivated within the vision system. If “No Paint” is detected, car bodies are automatically deselected.

4.3.2.2 Ghost Vehicle Administration

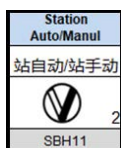


The painting processes can be simulated not only with actual car body data but also with ghost car body data. On the vision system, up to eight ghost vehicles encompassing all potential types, paints, and options can be chosen. It allows for the selection of individual ghost car bodies or a continuous ring buffer of ghost car bodies. Furthermore, the ghosts can be simulated without any process.

Ghost Mode
Prerequisites

- The robot and the station must both operate in automatic mode.
- The station should not contain any car body.
- If a car body nears the type delivery while the ghost with conveyor is active, the ghost is instantly terminated and an error report is produced.
- If the operating mode (of the robot and/or station) is altered while the ghost is active, the ongoing program is dismissed.

4.3.2.3 Station Manual Mode



- The robot must operate solely in automatic mode.
- All safety mechanisms are enabled, prohibiting entry into the cell.
- The vision system can initiate movement or service programs.
- Manual control of valves and necessary adjustments for dosing pumps and air volume controllers are permitted.
- Fixed positions can be accessed only from the home position and when the cell is empty.

- The home position can be reached from any other position. Users are responsible for preventing collisions caused by robot movement.
- Robots can be selected or deselected within the vision system. However, deselection is only feasible in the “Deselection Position”.
- Atomizers can also be selected or deselected within the vision system.

4.3.3 Robot operating mode

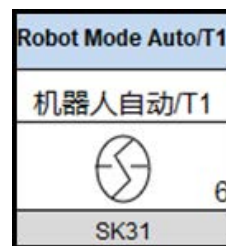
The following sub-sections are included in this section:

[‘Auto Mode’](#) on page 104

[‘Robot T1 Mode’](#) on page 104

Description

The mode of operation for the robot is determined using a key-operated switch located on the control panel of the station.



4.3.3.1 Auto Mode

- The robot’s production programs can only be initiated via the PLC.
- The robot operates at the velocities and accelerations specified in the teach program.
- Modifications to the teaching programs cannot be made through the FlexPendant.
- However, it is feasible to halt teaching programs using the stop button and adjust the speed overrides via the FlexPendant.
- When in manual station operation mode, the application can be managed through the vision system.

4.3.3.2 Robot T1 Mode

- In T1 test mode, you can create and test movement programs.
- The robot’s movement programs, also known as teaching programs, can only be initiated and halted at the FlexPendant.
- These movement programs can be tested either in single-step mode or continuously.
- The robot’s maximum speed is 250 mm/s.
- To operate the robot, the acknowledge switch needs to be enabled.

4 Operation and safety

4.4 Station light indicator

- T1 operation is only feasible in the “Maintenance” station operating mode. If the key-operated switch is set to T1 in any other mode, the robot will be immediately stopped via an emergency off.
- When operating the robot in T1 mode, the conveyor equipment must be turned off.
- It is not possible to move the conveyor and robots at the same time.
- The application cannot be managed through the vision system.

4.4 Station light indicator

Every station is outfitted with a four-color Indicator lamp.

Light Status	Description
Red	Fault
Blue	Conveyor running or release
Yellow	Warning message
Green	Auto



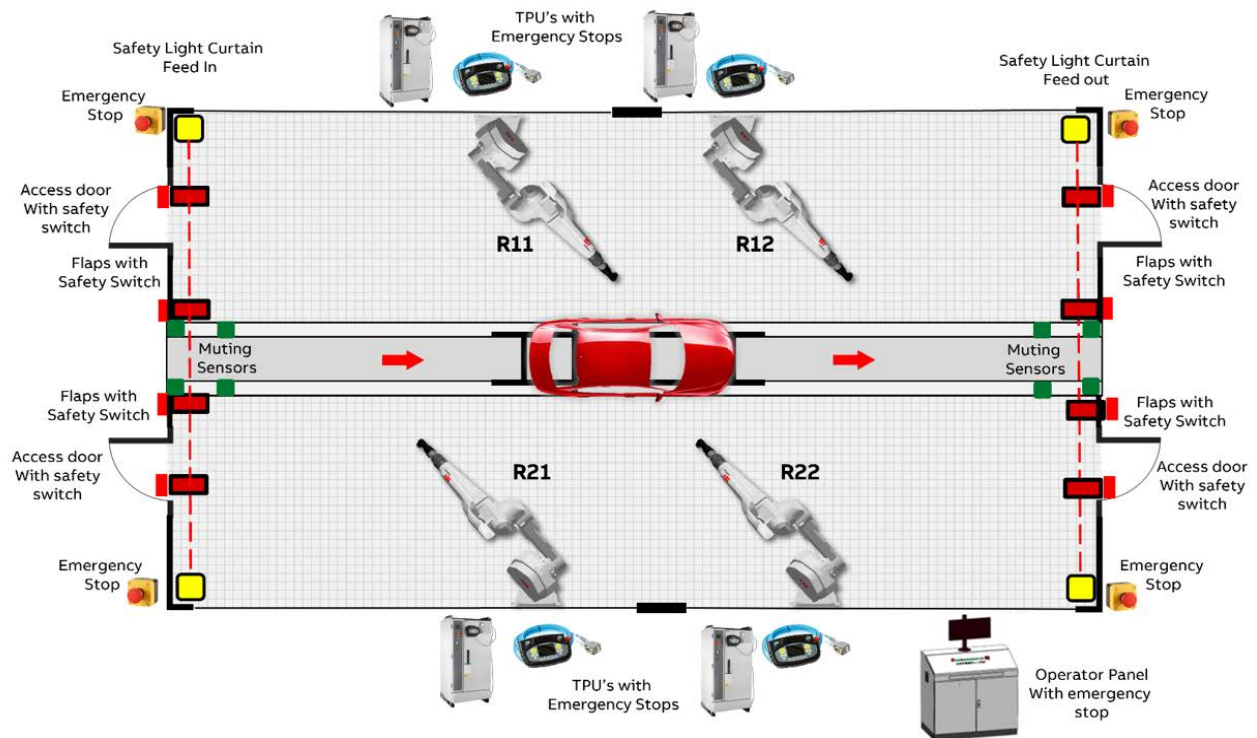
4.5 Safety system

The following sub-sections are included in this section:

- ['Safety layout'](#) on page 106
- ['Safety matrix'](#) on page 106
- ['Personal safety'](#) on page 111
- ['Fire Protection'](#) on page 116
- ['Cabin ventilation'](#) on page 117

4.5.1 Safety layout

The below image presents a standard safety arrangement, also available as a template in the Paint Controls package.



4.5.2 Safety matrix

The following sub-sections are included in this section:

['Safety matrix overview'](#) on page 107

['Robot safety stop matrix'](#) on page 108

['Station safety stop matrix'](#) on page 110

4 Operation and safety

4.5 Safety system

4.5.2.1 Safety matrix overview

The below table represents the primary safety matrix of the robot station.

Safety Matrix	Emergency mode Stop	General mode Stop	Test mode Stop	Auto mode Stop	Delayed Stop	Cabin Interlock	High Voltage Interlock	Turbine Interlock	System2 Interlock	Process Interlock	Main air
OP E-Stop	x										
Robot E-Stop	x										
Fire alarm	x										
Conveyor E-Stop	x										
Booth E-Stop	x										
Light barrier stop				x							
Swing door stop				x							
Booth ventilation open						x					
Booth clean mode		x				x					
Booth maintenance mode				x							
Booth auto mode											
Robot jogging			x								
Robot auto cleaning							x				
CBS door switch				x							
TPU door switch				x							

4.5.2.2 Robot safety stop matrix

The table outlines the safety stop procedures for safety devices based on the safety inputs of the robot control system.

For robot stop:

There are several types of stop for robot:

General Mode Stop (GSTOP)	Activating the General Stop Mode will halt the robot and disconnect the electric power from the axis motors, similar to triggering the emergency stop chain. However, unlike the emergency stop, the General Mode Stop doesn't require a reset signal to clear the stop condition. To re-energize the axis motors, the reset button on the main panel or an external panel or PLC must be activated. This falls under Stop category 0.
Auto Mode Stop (ASTOP)	Operating the Auto Stop Mode function will remove power from the axis motors when the robot mode selector is in Automatic mode. In robot manual mode, you can move the robot using a TPU. The difference between using the emergency stop and the auto mode stop function is that a reset signal is not required to reset the stop condition. Power must be re-applied to the axis motors by activating a reset button on the main panel or from an external panel or PLC. Stop category 0.
Emergency Stop (ESTOP)	The Emergency Stop chain operates on a dual channel system, requiring all switches and relay contacts in the chain to be closed for power to reach the axis motors. If this chain is broken, power to the robot's axis motors will be cut off, triggering the axis brakes (this is referred to as breaking the run chain). Additionally, breaking the chain will result in the paint system being disabled, high voltage being switched off, and pump motors being stopped, among other things. To restore power to the axis motors, the reset button on the main panel or an external panel or PLC stop. This is classified as Stop category 0.

4 Operation and safety

4.5 Safety system

HV interlock	The risk to personnel from high voltage is mitigated through the installation of either a manual interlock switch or a switch function managed by the plant's safety supervision system, a PLC, cabin door, etc. The high-voltage interlock chain serves to prohibit personnel from entering the cabin when the high voltage is active.
Cabin interlock	The cabin interlock chain is designed to deactivate paint operations that release combustible substances (such as paint, solvent, etc.) into the cabin in the absence of essential cabin functions like ventilation (cabin fans) or the fire extinguishing system.

In summary, protective and emergency stops are outlined by the IEC 60204-1 standard. Stops can fall under category 0. As per IEC 60204, a category 0 stop involves an immediate power cut to the machine actuators, resulting in an uncontrolled stop. This is executed in IRC5P by instantaneously cutting off power in the drive units.

The table illustrates the safety stop procedures of safety devices based on the safety inputs of the robot control system.

Safety Matrix	Emergency mode Stop	General mode Stop	Test mode Stop	Auto mode Stop	Delayed Stop	Cabin Interlock	High Voltage Interlock	Turbine Interlock	System2 Interlock	Process Interlock	Main air
OP E-Stop	x										
Robot E-Stop	x										
Fire alarm	x										
Conveyor E-Stop	x										
Booth E-Stop	x										
Light barrier stop				x							
Swing door stop				x							
Booth ventilation open						x					
Booth clean mode		x				x					
Booth maintenance mode				x							
Booth auto mode											
Robot jogging			x								

Safety Matrix	Emergency mode Stop	General mode Stop	Test mode Stop	Auto mode Stop	Delayed Stop	Cabin Interlock	High Voltage Interlock	Turbine Interlock	System2 Interlock	Process Interlock	Main air
Robot auto cleaning							×				
CBS door switch				×							
TPU door switch				×							

For instance, if the OP E-Stop is activated, it will trigger an emergency stop, causing the robot to halt movement, painting, and air circulation.

4.5.2.3 Station safety stop matrix

The table shows safety stops of safety devices on safety inputs of the robot control.

Station mode					
Level 1	Station		Teaching	Cleaning	Maintenance
Level 2	Station Automatic	Station Manual	-		
Robot mode	Automatic		T1		
Safety signal to the robot controller "Cabin interlock"	Stop ①	Stop	NA	NA	NA
Safety signal to the robot controller, "Auto mode stop"	Stop	Stop	NA	NA	Stop
Safety signal to the robot controller, "Test mode stop"	NA	NA	Stop (only one robot)	NA	NA

4 Operation and safety

4.5 Safety system

Station mode					
Level 1	Station		Teaching	Cleaning	Maintenance
Level 2	Station Automatic	Station Manual	-		
Robot mode	Automatic		T1		
Safety signal to the robot controller, "General mode stop"	Stop	Stop	Stop	Stop	Stop
Safety signal to the robot controller, "Emergency mode stop"	Stop	Stop	Stop	Stop	Stop
Main air	Stop	Stop	NA	NA	NA

Stop ①” implies that whether the station is in auto or manual mode, if the robot is in automatic mode and the Cabin interlock is triggered, the robot will halt.

“NA” signifies that there will be no impact and the robot will not stop.

In the event of a safety stop (Estop), the painting process will resume once the faults have been reset.

To resume the production process after a safety stop has been activated, follow the steps below:

1. Press the fault reset button and verify on the HMI if the fault has been cleared.
2. Press the cyclic start button.
3. Activate the high voltage by pressing the corresponding button.
4. Initiate the conveyor by pressing the run button.

4.5.3 Personal safety

The following sub-sections are included in this section:

[‘SafeMove’](#) on page 112

[‘Cabin entry protective stops’](#) on page 112

[‘Emergency stops’](#) on page 114

[‘HV safety functions’](#) on page 115

4.5.3.1 SafeMove

Overview



Paint Controls is entirely based on the PROFIsafe / CIP safety interface, which was launched alongside SafeMove2. This makes the safety interface a compulsory component in robotics. However, it is important to note that SafeMove, with its movement supervision, is not required to operate this package.

The default configuration used is the one provided in the ABB paint documentation, specifically in the “Application manual, Functional safety and SafeMove for paint” document.

Furthermore, Paint Controls enhances the system by adding some signals and a predefined safety configuration in both the robot and the PLC.

Cyclic brake check

Once SafeMove is implemented on the robot, it necessitates a periodic brake check to guarantee the safety (performance level) of the movement supervision.

A preset function can initiate the cyclic brake check from the station HMI. Alternatively, the brake check can be initiated using the TPU.

Safety synchronization

Upon the installation of SafeMove on the robot, it is necessary to synchronize the safety module with the axis computer. By default, this safety synchronization is executed through a software synchronization, with only the status being reported to the PLC.

4.5.3.2 Cabin entry protective stops



Light Barrier

The zone light barriers are positioned at the entrance and exit of the booth. Activation occurs when the robot is in automatic mode, similar to the safety/swing door.

These light barriers are linked with muting sensors to distinguish between a person intruding into the booth and the normal entry or exit of a skid.

If a person is detected instead of a vehicle, it triggers an immediate emergency stop. The robot ceases movement, and the process is halted.



Muting Operation

The light barriers are controlled at the entry and exit of each application zone, with two photocells installed on the booth and a light barrier positioned behind the booth glass.



Operation of the Shielded Sensor

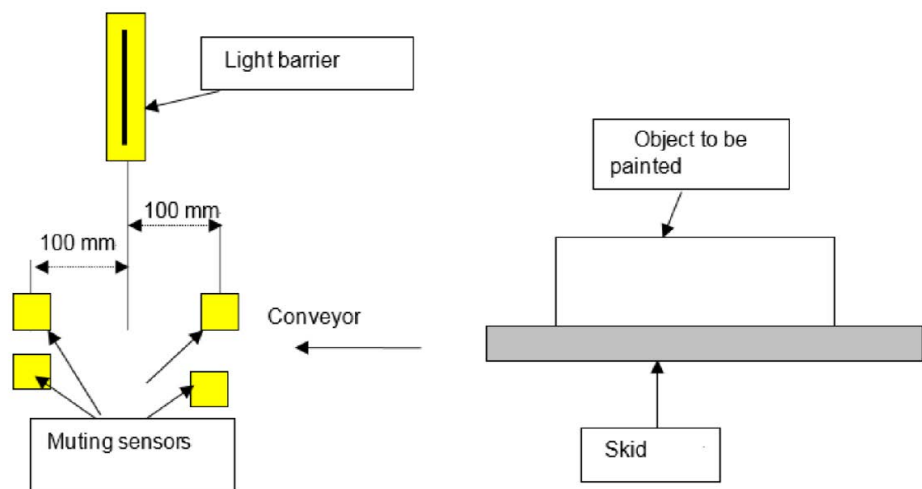
The shielded sensor operates as follows:

When a skid enters the booth, the inclined photocell detects the first part of the car body before the light barriers are blocked. The muting function is activated based on the car body's position (monitored via the body position monitor window), allowing the skid to pass through the booth without triggering a fault. This same photocell also functions when the skid exits the booth.

In other scenarios, such as when people or other objects block the light barriers, a fault will be triggered because their positions cannot be tracked even though the photocell is activated.

The car body's position is determined using the station start sensor, which checks whether it is a car body and not a person who has interrupted the light barrier.

Please note that the distances mentioned in the muting example are illustrative and may not match the exact layout.



Safety Doors in Booths

Typically, each booth is outfitted with safety doors. These doors are fitted with switches that interrupt the double-channel safety signal when the door is opened.

Automatic Mode in Robots

The safety measures in the area are linked to the robot's automatic Stop function.

In automatic mode, if a door is opened or a light barrier is interrupted, the robots halt. This also results in the cessation of the power supply to the motors and the application.

T1 Mode in Robots

In this mode, the doors and light barriers are not active.

This facilitates the operator's work by allowing proper use of the access doors.

Swing Doors in Booths

Each booth comes with four swing doors. These doors are fitted with switches that interrupt the dual channel safety signal when opened. Their functionality largely mirrors that of the safety door.

4.5.3.3 Emergency stops

In all operating modes, emergency stop devices are active. Each station is equipped with the following emergency stop devices:

- One emergency stop switch on the control panel/visualization.
- One emergency stop switch on every robot FlexPendant.
- Four emergency stop switches in the booth.



Actions Performed in the Stations :

- The robot's movement is halted by the emergency stop.
- The high voltage release is reset.
- The application, including the rotation colour changer, pumps, and valves, is turned off.
- The station maintains its previously active mode.
- The conveyor system's release is reset.

Note: Safety signals are handled by the fail-safe module of the PLC, which operates on two channels. In the event of a channel error, the corresponding emergency stop button must be activated and then deactivated before the error can be recognized and addressed.

- The safety stop function is station-specific and does not affect other stations.
- The emergency stop switch, linked to the robot's external emergency Stop, deactivates the motor drive when pressed. To reactivate the motor, the corresponding switch that initiated the security stop must be released.
- If an application was running, it would be interrupted, affecting both compressed air and high-voltage systems. The emergency stop signal also halts the conveyor system.
- Once the emergency stop is disengaged, the ongoing cycle resumes upon restarting the robot station, allowing production to continue.

4.5.3.4 HV safety functions

The painting robot is equipped with a High Voltage (HV) safety function.

Each robot station features a high-voltage warning light located outside the booth, indicating the following statuses:

– **Red light On:** High voltage is active.

– **Green light On:** High voltage is grounded.

- The robot's high-voltage can be toggled on and off by pressing the "HV Enable" button on the operation panel. This function can also be bypassed using the HMI.
- If the robot loses high voltage feedback, the conveyor release will be turned off. Therefore, it is necessary to check it with the HMI.
- The high voltage interlock is active to ensure personal safety regardless of working mode. It is forbidden to enter the booth when high voltage is displayed.
- Pressing the Emergency stop button, interrupting the light curtain, or opening the safety door will deactivate the high-voltage enable function. If the high voltage is disabled, it would have a warning message in the vision system.



4.5.4 Fire Protection

When a signal is triggered from the fire fault system, the following actions occur:

- The robot's movement is halted by activating the emergency stop mode.
- The release of high voltage is reset.
- The application (including rotation colour changer, pumps, and valves) is turned off.
- The station continues to operate in the previously active mode.
- The release of the conveyor system is reset.
- The safety valve in the relevant booster cabinets closes immediately.

The interlock signal comes from the existing firefighting system provided by the customer; the interlock between Robot PLC and the fire fault system is connected by one cable.

The cycle in progress will continue after the fire fault is switch off.

4.5.5 Cabin ventilation

Cabin Ventilation refers to the supply and exhaust air system. Signals from the dry contact are linked to the DI module of the PLC and ultimately processed for the robot cabin inter-lock. It is crucial for the booth ventilation to be operational for automatic production spray.

When the Booth Ventilation is not functioning, the robot stations respond as follows:

- The high voltage is turned off.
- The application (including rotation colour changer, pumps, and valves) is deactivated.
- The station continues in its previously active mode.
- The conveyor system's release is reset.

5 Control system

About this chapter

This chapter provides information on the interface used to communicate between the PLC control systems.

The following sections are included:

'Network structure' on page 118

'Robot Naming' on page 120

'Type Verification' on page 123

'Interface' on page 123

'Blow Off' on page 132

'3D Vision system' on page 133

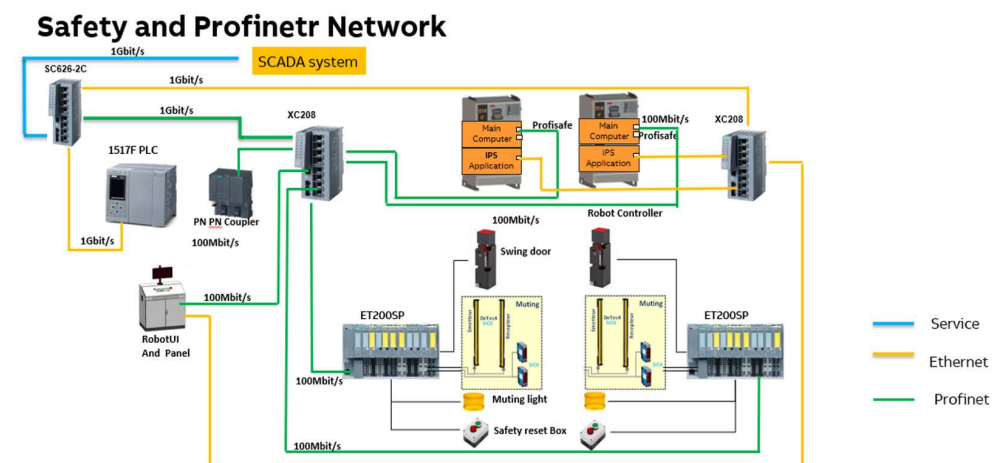
5.1 Network structure

The control system primarily outlines the fundamental electrical concepts, which include the network structure and external interfaces.

Overview

The image below illustrates the universal network architecture that is applicable to all types of applications.

Note: It is recommended that the industrial Ethernet network be physically separated.



Required IP addresses:

Each robot requires the subsequent number of IP addresses:

Application type	No.	Usage
Painting	3	1 x PROFINET, 1 x Main Computer, 1 x Process Interface Board (PIB)
Dispensing	3	1 x PROFINET, 1 x Main Computer, 1 x Process Interface Board (PIB)
Cleaning	2	1 x PROFINET, 1 x Main Computer

When PROFINET and Ethernet are merged, the individual PROFINET address is no longer necessary.

Required network cables:

The following number of network cables are required per robot:

Application type	No.	Usage
Painting	3	1 x PROFINET, 1 x Main Computer, 1 x Process Interface Board (PIB)
Dispensing	2	1 x PROFINET, 1 x Built in Ethernet switch in controller cabinet
Cleaning	2	1 x PROFINET, 1 x Main Computer

When PROFINET and Ethernet are merged, the individual PROFINET address is no longer necessary.

Paint

In a paint booth, all the application equipment is directly linked to the robots.

Dispensing

The IDFP autonomously consolidates all application tools and manages the communication among its hardware components.

Note: When a path correction sensor is installed on the robots, it establishes an independent connection for communication between the robot (acting as the master) and the measurement PC (acting as the client). Any communication intended for a camera PC for 3D displacement is consistently directed via the PLC.

Cleaning

Within a cleaning cell, all application equipment is fully connected and managed by the PLC, which also controls the movement of the blower portals. The cleaning station comprises two types: “Feather dust” and “Sword brush”.

5.2 Robot Naming

The process of naming a robot is a component of the control system. Here are the guidelines for assigning names to robots.

The following sub-sections are included in this section:

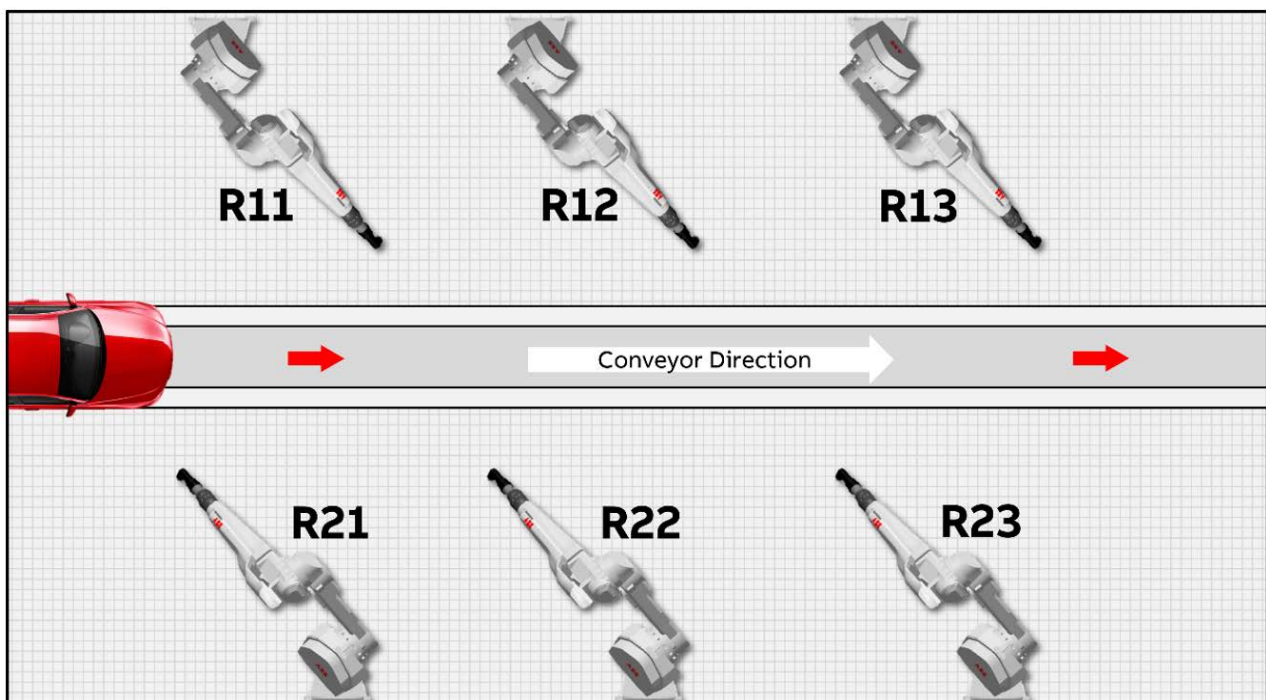
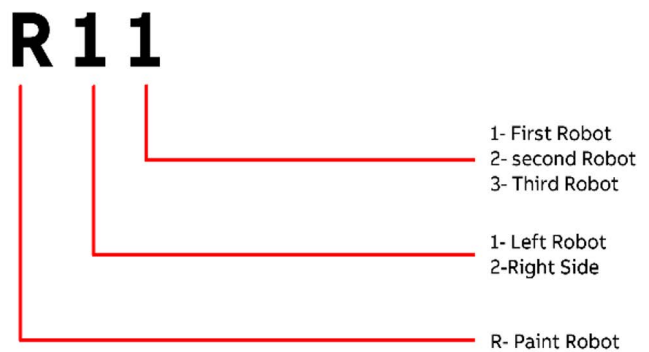
'Exterior station' on page 120

'Interior station' on page 121

'Sealing station' on page 122

5.2.1 Exterior station

- Exterior Station
- Paint Robot



5 Control system

5.2 Robot Naming

5.2.2 Interior station

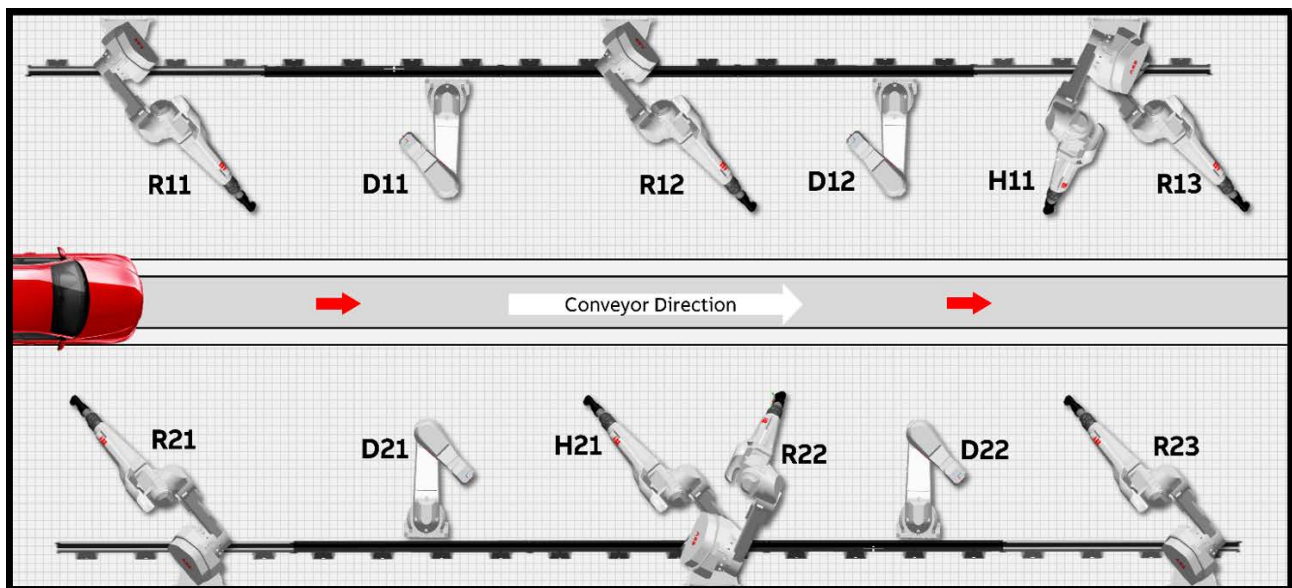
- Interior Station
- Paint Robot
- Door Opener
- Hood & Trunk Opener

D 1 1

1- First Robot
2- second Robot
3- Third Robot

1- Left Robot
2- Right Side

D- Door Opener
H- Hood opener

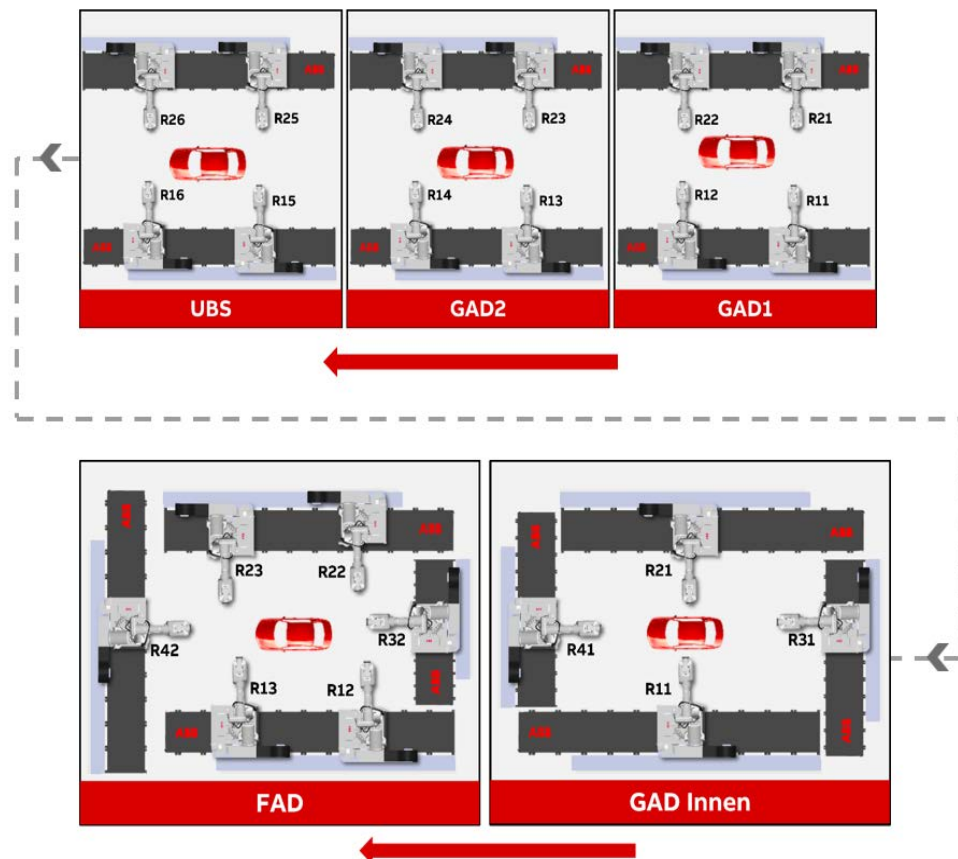


5.2.3 Sealing station

Opener Robot:

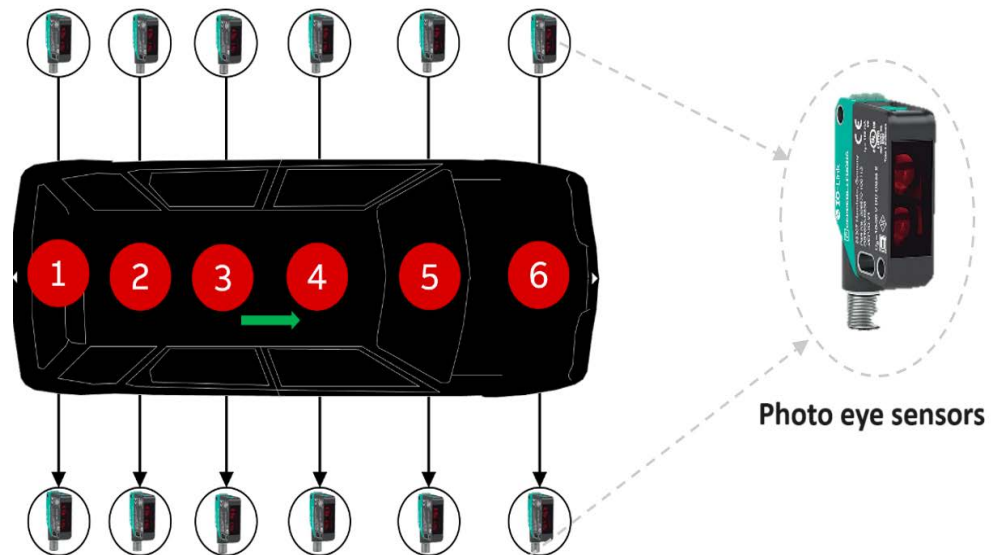
H - Open trunk/hood robot

D - Open door robot



5.3 Type Verification

Type verification involves reconfirming the car body type when the APT's initial station receives data from the conveyor or reading station.



There are six pairs of sensors in total. These sensors can be arranged in numerous combinations, such as 1,...; 12,...; 123,...; 1234,...; 12345,... and so on.

Different car models have unique characteristic points. By using the sensors to detect these points, the car model can be identified.

5.4 Interface

The following sub-sections are included in this section:

['Interface description between PLC and Robot'](#) on page 124

['Type and Vehicle Data'](#) on page 125

['Conveyor Equipment'](#) on page 126

['Process Equipment'](#) on page 127

['Fire Protection'](#) on page 127

['Central Control Room'](#) on page 127

['Paint Supply'](#) on page 128

['Special Colour Supply'](#) on page 130

['PDA'](#) on page 131

5.4.1 Interface description between PLC and Robot

Robot Releases	Corresponding to the robot selections of the vision system, the drives and the movement release are given to the robot controller and the atomizer.
Transfer of Application Data and Profile Data	Based on the car body's current position in the inlet zone, the robots receive the car body details and type information.
Fixed Position Interlocks	The station manager oversees the robot's movement towards fixed positions. Essentially, the robot can only approach these positions from the home position, provided there is no vehicle in the area. Moreover, when entering or exiting atomizer or air cap cleaning devices, the robot is restricted to vertical movements only.
Collision Locks	Specifically, when multiple application robots are involved in coating, it is crucial to prevent any collisions. To achieve this, the station controller is aware of various collision fields, each of which can only be occupied by one robot at a time. Before entering such a field, each robot must first obtain permission from the station controller. If another robot is already in the field, the permission is withheld until the first robot has exited the field.
Alarm Handling	The station controller is responsible for assessing all station hardware alarms, the safety system's alarms, and the collective alarms from each robot.
Personnel Protection	The safety PLC monitors all personnel protection facilities, including emergency stop switches, monitor switches at booths and swing doors, and acknowledging switches at FlexPendants.

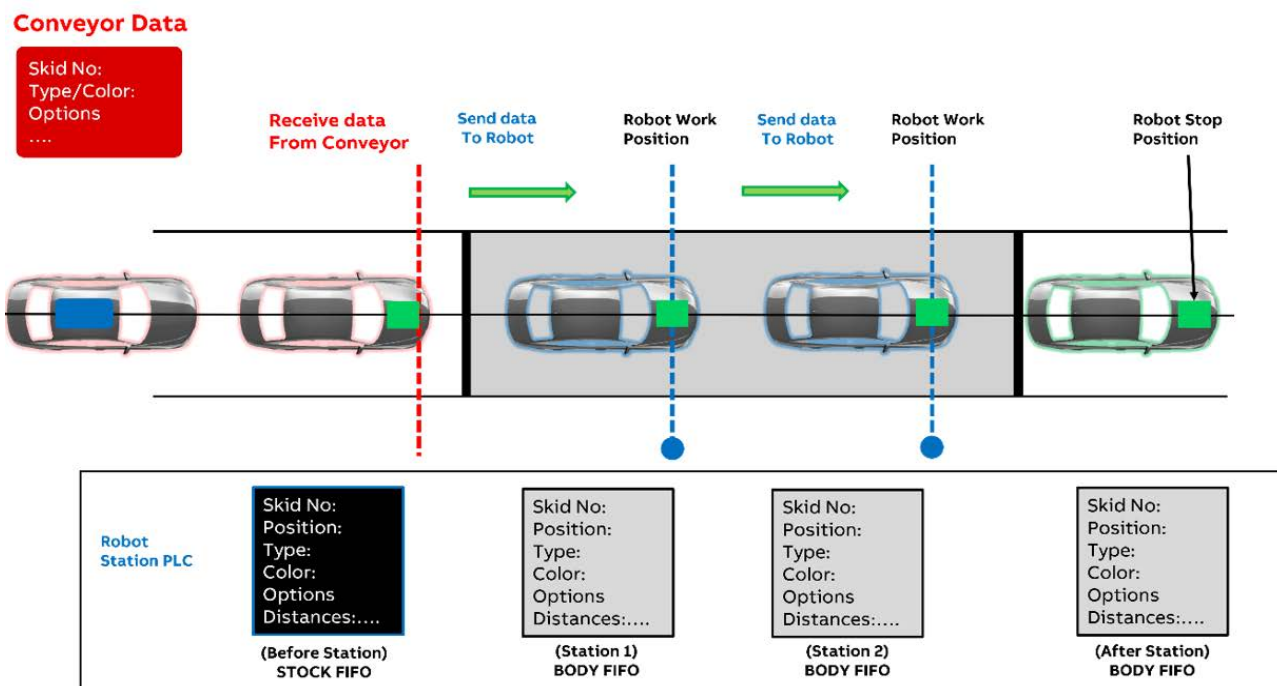
5.4.2 Type and Vehicle Data

The booth is segmented into two sections: the area before the station and the area within the station. For the carbody located before the station, the data is stored in the STOCK FIFO. For the carbody situated within the station, the data is stored in the BODY FIFO.

The BODY data structure encompasses the following elements:

- Position of the body
- Type code
- Colour code
- Skid number
- VIN code
- Sunroof status
- Degrade mode
- Among others

In extreme conditions, it is allowed to clear Booth Piles and robot job queues.

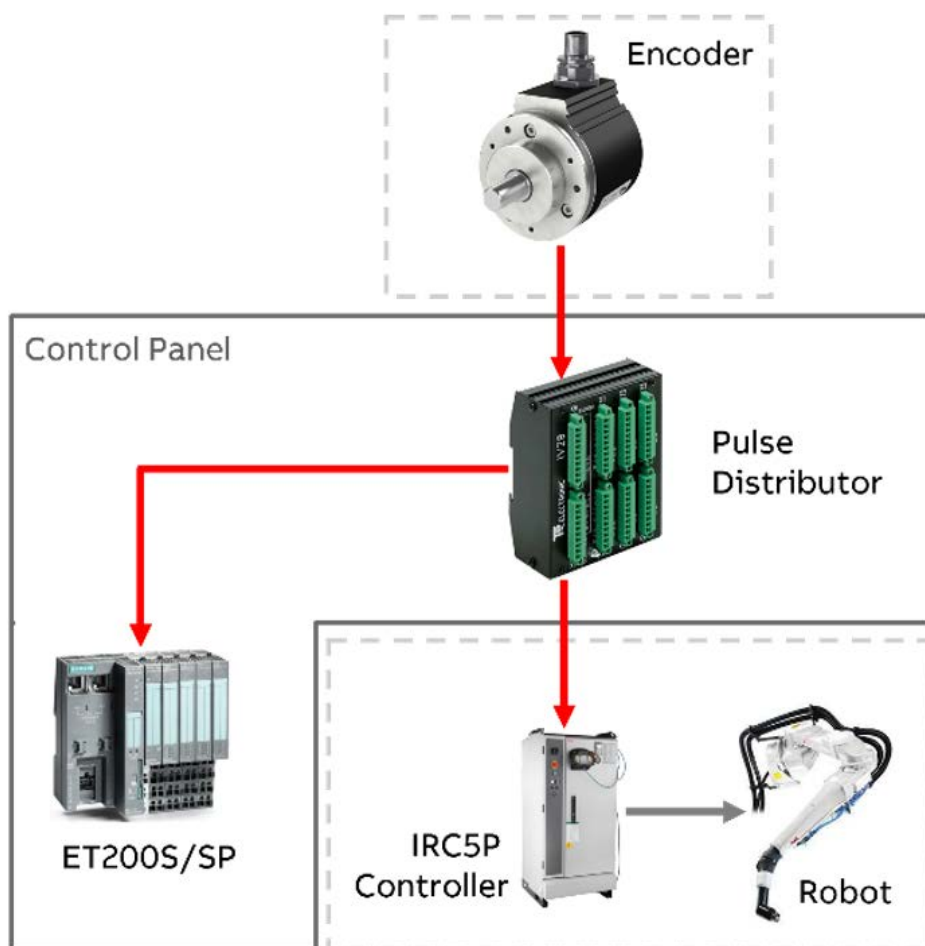


5.4.3 Conveyor Equipment

The conveyor run operation can be classified into two types: “Tracking” and “Stop&Go”. The interface to the conveyor system, for both safe and non-safe signals, is facilitated through a PN/PN coupler. These couplers are housed in the PLC cabinet provided by APT. In this scenario, the conveyor system is responsible for providing the cables.

Conveyor Increments

The increments are identified using a dedicated pulse encoder that is connected to the conveyor motor and shared across all stations. Each robot and PLC receives a signal. For example, at the initial station, an ET200S/SP counter card is employed. The signals are then transmitted to other stations using an IV20 board as an example. It is worth nothing that other types of boards may be used as well.



5.4.4 Process Equipment

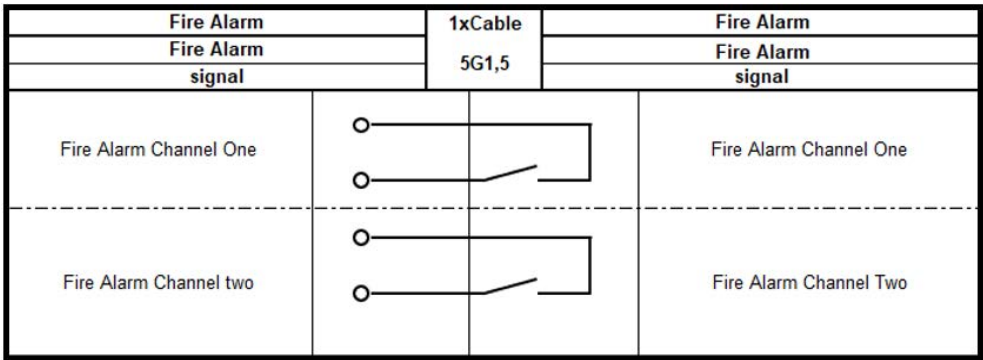
The humidity data for the cabins is relayed to the cabin ventilation system via a PN/PN coupler. The station visual displays the analogue values for “temperature” and “humidity”. Digital signals transmit warning and alarm messages, which must be shown on the APT controller.

The APT does not generate any alarm or warning messages. These are sent by the VT and displayed on the APT VISU. Furthermore, the displays for a warning or fault are activated.

5.4.5 Fire Protection

The fire protection interface is structured as follows:

Safety-critical signals are executed as dual-channel signals through secure, potential-free contacts. The fire protection system communicates signals via hardware interlocks, which shut off in the event of a fire. If the fire protection system is not activated, all paint and solvent valves are immediately closed. The drives are also promptly deactivated.



5.4.6 Central Control Room

The connection to the central control room is established through the Ethernet port on the root switch at each station.

5.4.7 Paint Supply

The design of the interface to the paint supply is as follows: It is through a ProfiNet PN/PN coupler. There's no need for safe signals.

Paint Supply - APT			
O_Cs_Heartbeat	0.0	BOOL	Heartbeat signal
O_Cs_BC-Solvent 1	0.1	BOOL	BC-Solvent 1 ok
O_Cs_BC-Solvent 2	0.2	BOOL	BC-Solvent 2 ok
Waste_Solvent_Pump	0.3	BOOL	Waste solvent pump ok
Waste_Solvent_Enable	0.4	BOOL	Waste solvent enable drain
Spare	0.5	BOOL	Spare
Spare	0.6	BOOL	Spare
Spare	0.7	BOOL	Spare
Array[1....20] of PipeInfo_UDT BC (1-20)			
Pipe[1]	2.0	Bool	
Release	2.0	Bool	Release
CCR Enable	2.1	Bool	Enable
Sleep	2.2	Bool	Sleep
Color code	4.0	Array[1....8]	Color code
Color code[1]	4.0	Char	Color code
Color code[2]	5.0	Char	Color code
Color code[3]	6.0	Char	Color code
Color code[4]	7.0	Char	Color code
Color code[5]	8.0	Char	Color code
Color code[6]	9.0	Char	Color code
Color code[7]	10.0	Char	Color code
Color code[8]	11.0	Char	Color code
Pipe[2]	12.0	Bool	
Release	12.0	Bool	Release
CCR Enable	12.1	Bool	Enable
Sleep	12.2	Bool	Sleep

5 Control system

5.4 Interface

APT - Paint Supply			
I_Cs_Heartbeat	0.0	BOOL	Heartbeat signal
I_Cs_BC-Solvent 1 Request	0.1	BOOL	BC-Solvent 1 request
I_Cs_BC-Solvent 2Request	0.2	BOOL	BC-Solvent 2 request
I_Cs_BC1 Request	0.3	BOOL	BC1 paint request
I_Cs_BC2 Request	0.4	BOOL	BC2 paint request
I_Cs_BC3 Request	0.5	BOOL	BC3 paint request
I_Cs_BC4 Request	0.6	BOOL	BC4 paint request
I_Cs_BC5 Request	0.7	BOOL	BC5 paint request
I_Cs_BC6 Request	1.0	BOOL	BC6 paint request
I_Cs_BC7 Request	1.1	BOOL	BC7 paint request
I_Cs_BC8 Request	1.2	BOOL	BC8 paint request
I_Cs_BC9 Request	1.3	BOOL	BC9 paint request
I_Cs_BC10 Request	1.4	BOOL	BC10 paint request
I_Cs_BC11 Request	1.5	BOOL	BC11 paint request
I_Cs_BC12 Request	1.6	BOOL	BC12 paint request
I_Cs_BC13 Request	1.7	BOOL	BC13 paint request
I_Cs_BC14 Request	2.0	BOOL	BC14 paint request
I_Cs_BC15 Request	2.1	BOOL	BC15 paint request
I_Cs_BC16 Request	2.2	BOOL	BC16 paint request
I_Cs_BC17 Request	2.3	BOOL	BC17 paint request
I_Cs_BC18 Request	2.4	BOOL	BC18 paint request
I_Cs_BC19 Request	2.5	BOOL	BC19 paint request
I_Cs_BC20 Request	2.6	BOOL	BC20 paint request
Spare	2.7	BOOL	Spare
Spare	3.0	BOOL	Spare
Spare	3.1	BOOL	Spare

5.4.8 Special Colour Supply

The design for the interface to the special colour supply is as follows: It utilizes a ProfiNet PN/PN coupler. There's no necessity for safe signals.

KVS - APT			
RobotIn	ToRobotIn_UDT	0	
Heartbeat	Bool	0	Heartbeat 1s
Spare1	Bool	0.1	Spare
Spare2	Bool	0.2	Spare
Spare3	Bool	0.3	Spare
Spare4	Bool	0.4	Spare
Spare5	Bool	0.5	Spare
Spare6	Bool	0.6	Spare
Spare7	Bool	0.7	Spare
Release	Bool	1	Release
Color code	Array[1....4] of Char	2	Color code
Color code[1]	Char	2	Color code
Color code[2]	Char	3	Color code
Color code[3]	Char	4	Color code
Color code[4]	Char	5	Color code
Body type	Array[1....4] of Char	6	Body type
Body type[1]	Char	6	Body type
Body type[2]	Char	7	Body type
Body type[3]	Char	8	Body type
Body type[4]	Char	9	Body type
Body ID	Array[1....4] of Char	10	Body ID
Body ID[1]	Char	10	Body ID
Body ID[2]	Char	11	Body ID
Body ID[3]	Char	12	Body ID
Body ID[4]	Char	13	Body ID

5 Control system

5.4 Interface

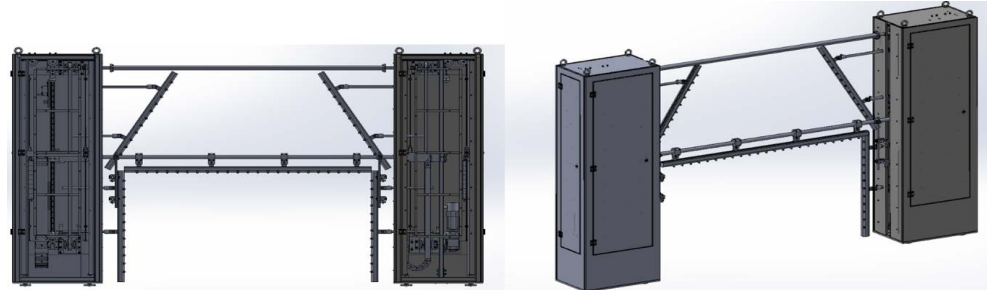
APT - KVS			
RobotIn	FromRobotics_UDT	148	From robot inner
Heartbeat	Bool	148	Heartbeat 1s
Request	Bool	148.1	Request
Pressure_Low	Bool	148.2	Pressure low
Spare3	Bool	148.3	Spare
Spare4	Bool	148.4	Spare
Spare5	Bool	148.5	Spare
Spare6	Bool	148.6	Spare
Spare7	Bool	148.7	Spare
BodyInfo	BodyInfo_UDT	150	Access BC Body information
ID	Array[1....4] of Char	150	BodyID (PIN)
ID[1]	Char	150	BodyID (PIN)
ID[2]	Char	151	BodyID (PIN)
ID[3]	Char	152	BodyID (PIN)
ID[4]	Char	153	BodyID (PIN)
ID[5]	Char	154	BodyID (PIN)
ID[6]	Char	155	BodyID (PIN)
ID[7]	Char	156	BodyID (PIN)
ID[8]	Char	157	BodyID (PIN)
ID[9]	Char	158	BodyID (PIN)
ID[10]	Char	159	BodyID (PIN)
ID[11]	Char	160	BodyID (PIN)
ID[12]	Char	161	BodyID (PIN)
ID[13]	Char	162	BodyID (PIN)
ID[14]	Char	163	BodyID (PIN)

5.4.9 PDA

The items are not sold at the moment.

5.5 Blow Off

The blow-off is typically set up either before the Sword brush station or after the Feather dust station. The mechanical layout of the blow-off is in 3D.

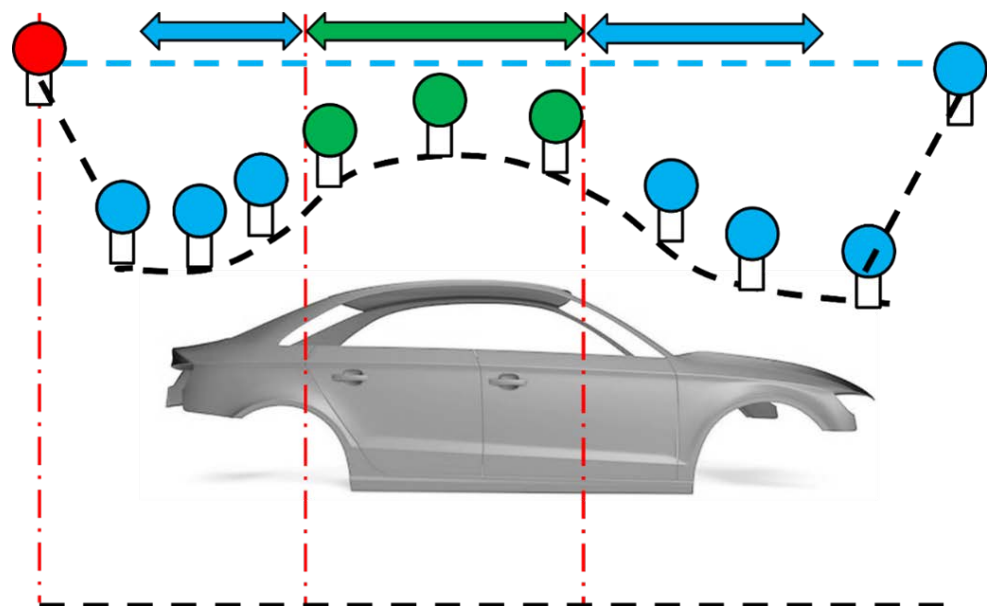


Principle

The principle involves adjusting the ion balance between the entrance environment of topcoat lines in the car painting workshop and the car body. This adjustment helps to prevent dust adhesion due to electrostatic adsorption. As a result, it reduces defects such as fibre hair and dust adsorption on the car body surface and in the cracks, thereby increasing the pass rate of one-time inspection. It also helps to reduce the number of purge personnel.

Schematic diagram of the blowing trajectory on the top of the vehicle body

The ION wind-blowing device employs a servo motor to operate a lifting mechanism that moves up and down in sync with the model. A horizontal blowing unit affixed to the lifting mechanism, carries out extensive dust removal and cleaning on the front cover, top cover, and tail cover. Additionally, a pair of vertical blowing units are positioned on the left and right.



5.6 3D Vision system

The following sub-sections are included in this section:

'3D Camera system' on page 133

'BK Laser System' on page 134

'Toolbox3D' on page 134

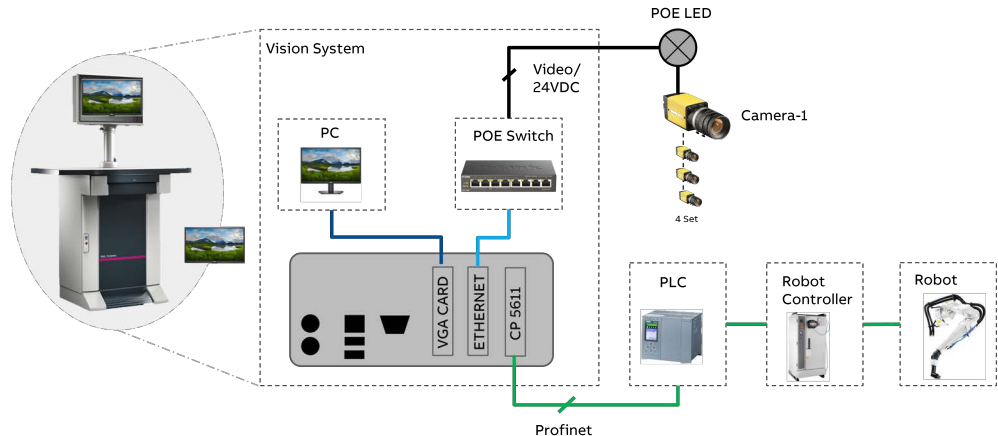
5.6.1 3D Camera system

This refers to the conventional camera positioning system, which solely determines the three-dimensional position of the car body using cameras. This system is compatible with PE/GAD/UBS/GAD/RP. Due to mechanical variations, the vehicle cannot halt at the exact same position every time. The control system calculates the current orientation of the vehicle body by identifying fixed characteristics and marks on the car body.

The system performs a six-coordinate conversion (X, Y, Z, RX, RY, RZ) and sends a rectified vector to the robot. This enables the robot to locate the body accurately.

The electrical structure is comprised of two parts: the connection between the camera and light and the PROFINET communication with the PLC and Robot.

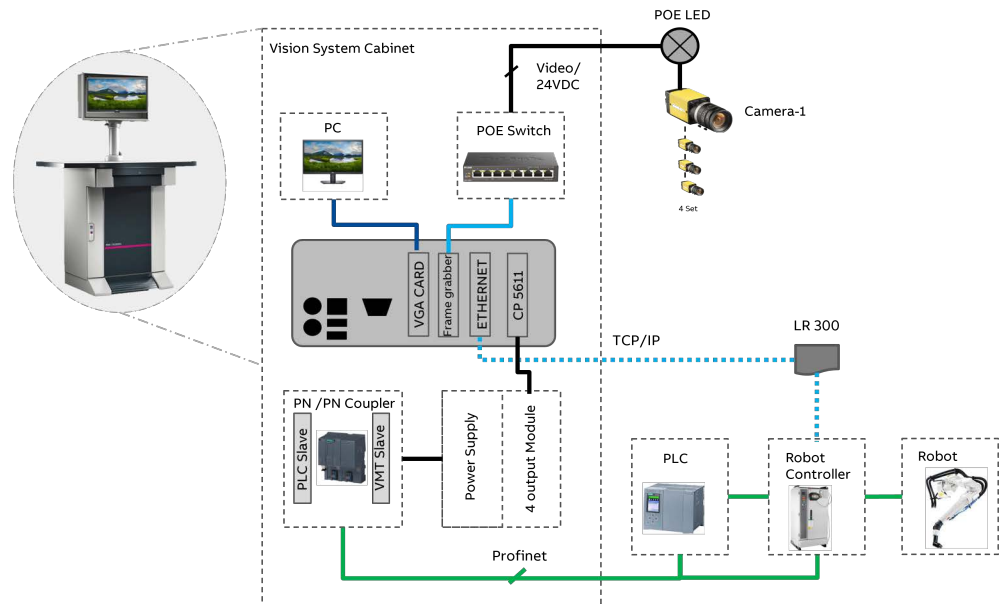
Electrical Structure



5.6.2 BK Laser System

This system is based on 3D camera positioning. It uses a laser to scan and measure specific parts of the car body closely, accurately determining their three-dimensional positions. The SS station is outfitted with two sets of the BK system and a 3D system housed in two PC cabinets.

Electrical Structure



5.6.3 Toolbox3D

This task is relatively independent and primarily operates in the same way as the 3D camera system. To ensure consistently high quality in automated production processes, regular validation of the Tool Center Points (TCP) is crucial.

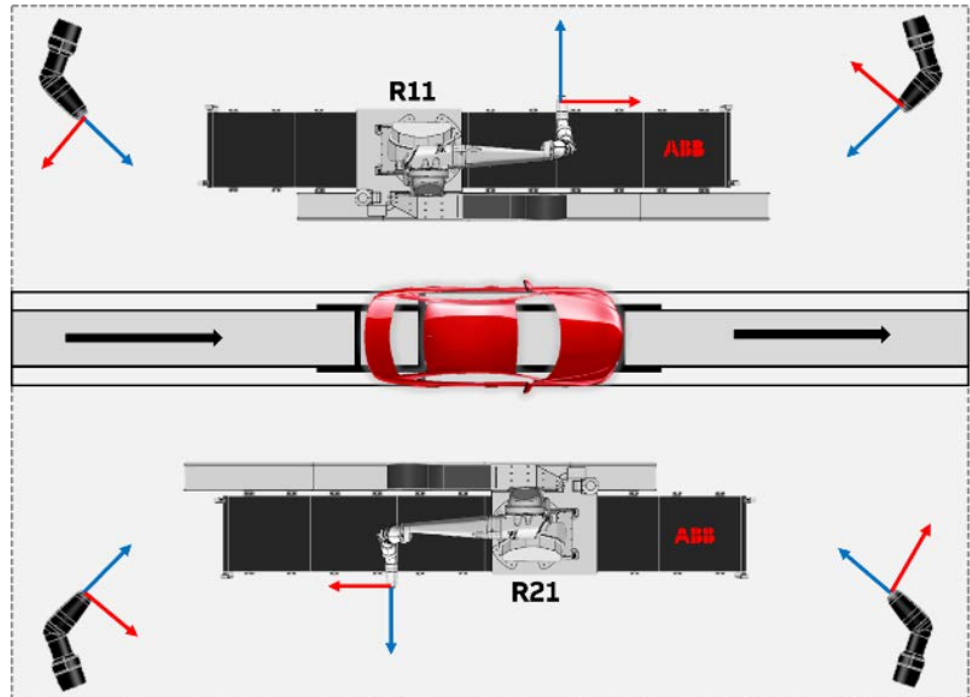
Especially in high-precision application processes, accurate tool positioning is crucial. In addition to the basic TCP Check, the VMT Toolbox 3D provides the automatic TCP Correction.

SS station is equipped with two sets of Toolbox3D, connected with two BK systems.

5 Control system

5.6 3D Vision system

SS station



6 HMI system

About this chapter

This chapter describes the most crucial station functions and provides user-level information on operating the station from the HMI.

This chapter describes the most crucial station functions. Paint and process knowledge is necessary to carry out the following operations.

It covers the general architecture and interface of the HMI screen.



WARNING! Before starting the tasks below, read the safety instructions in the system documentation. If you are not sure about handling the system components, then do not operate the system.



Important: In general, it is only possible to activate robot functions (for example, position runs, edit values, etc.) from the visualization when the robot operating switch is set to “Auto” position.

Following sections are included in this chapter:

[‘HMI Overview’](#) on page 137

[‘View’](#) on page 153

[‘Robot’](#) on page 162

[‘Control’](#) on page 210

[‘Setting’](#) on page 218

[‘Diagnose’](#) on page 229

[‘Report’](#) on page 232

[‘Program’](#) on page 237

[‘Help’](#) on page 239

6.1 HMI Overview

Robot UI

“Robot UI is a flexible software solution for ABB robot painting stations, developed using the .NET framework.”

The software supports various technologies, including electrostatic painting, pixel paint, cleaning, dispensing and quality inspections.

Following sub-sections are included in this section:

6.1.1 [‘Screen layout’](#) on page 138

6.1.2 [‘Header’](#) on page 139

6.1.3 [‘Menu’](#) on page 141

6.1.4 [‘Alarm Line’](#) on page 145

6.1.5 [‘Main window’](#) on page 146

6.1.6 [‘Additional window’](#) on page 147

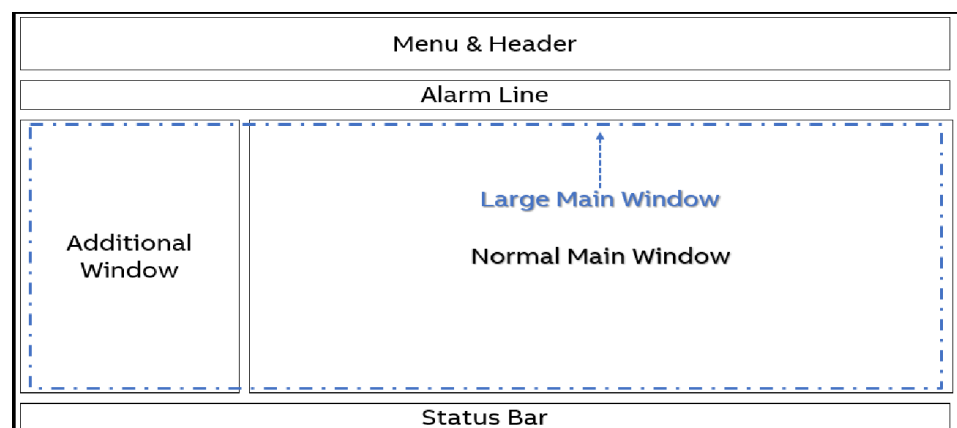
6.1.7 [‘Status Bar’](#) on page 148

6.1.1 Screen layout

Overview

The screen layout is clearly structured, making the information easy for the user to visualize and interpret. This ensures easy and practical training for the user.

Header	The header includes general information, such as station name, date, time, shift, user and language switch button.
Menu	The menu area contains the menu system, which opens the required interface by clicking the menu.
Alarm Line	The alarm line exhibits the current alarm messages. Click the “Alarm” button on the alarm, to directly navigate to the alarm interface.
Main Window	The main window reveals the most essential information (e.g. station overview). Depending on the representation, it is shown in conjunction with an additional window or as a total view.
Additional Window	The additional window typically presents the status or control functions for the main window.
Status bar	The status bar displays station states (e.g. station mode). Quick selection keys are also integrated here (e.g. control robot position).



6 HMI system

6.1 HMI Overview

6.1.2 Header

The header shows general information, including software version, license info, station name, ABB logo, language switch button, user login button, username, user level, date, time, and shift information.



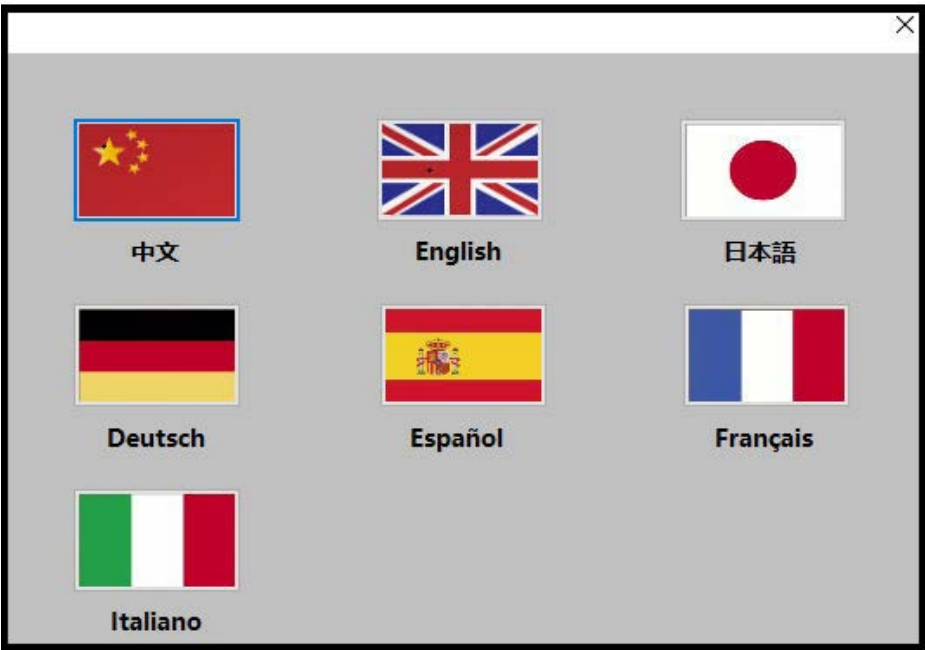
A dongle is a device that connects to a PC via a USB port. Once the dongle is plugged in, the RobotUI software initiates communication with the PLC and Robot. ABB tailors the dongle's information according to the customer's details. By inserting the dongle into the PC, the RobotUI software can be activated. The software's successful authorization is confirmed when "Runtime" appears in the interface's top right corner.

The current dongle reads "Factory Name" and "Robot Number" from the config file, so the customer does not need to provide PC information. A new software license will be created and a section about it will be included later.



Language

Users can change the language by clicking the language button.



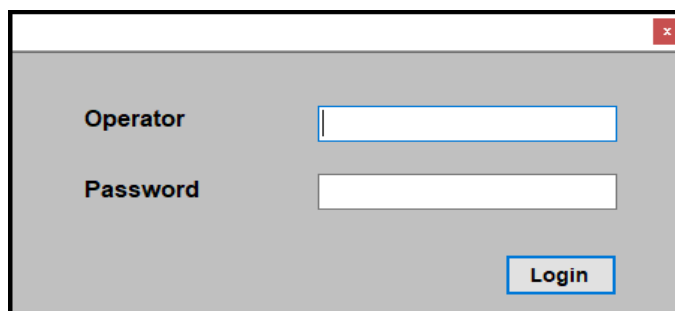
Login

Click the login button  and input the username and password to log in.

Each menu binds with the user access level. If the user does not have written permission, the user cannot operate related menu pages.

The default user name is "Null". Operators can only perform basic operations such as viewing station status. Without a login, the operator can view interfaces but cannot control them. For more information, see Security chapter 6.5.6 on how administrators can add and manage operator levels.

ABB engineers can use the "abb" account to commission internal systems and then delete it. Accounts will be created for customers.



A screenshot of a login dialog box. It has a title bar with a close button (X). The dialog contains two labels: "Operator" and "Password". Next to each label is a text input field. At the bottom right of the dialog is a button labeled "Login".

Shift

Shows current shift information, which will be recorded in the alarm, operation, and production reports.

2023-09-12	Shift
10:56:46	1

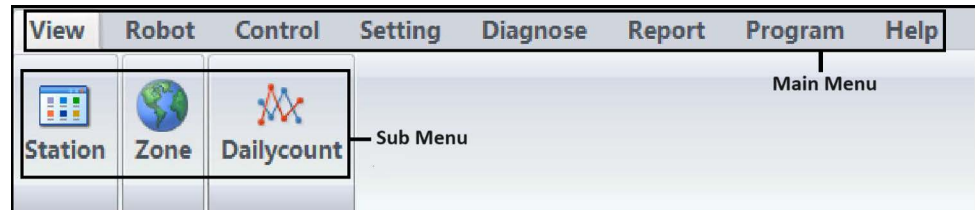
For more information, see Miscellaneous chapter 6.5.7, operator can set shift configuration in Miscellaneous Setting menu (shift start time and end time).

For more information, see Production chapter 6.7.1, header will display current shift and production report, shift information will be recorded in the production report, consumption report, operation report and alarm report.

6.1.3 Menu

The menu system contains a main menu and a sub-menu. The visualization offers the following possible selections.

Select the main menu, then the sub menu; function interfaces will display in the main window.



PP = Pixel Paint

FD = Feather Duster

SWB =Sword Brush

IDFP =Integrated Dispensing Function Package

DP = Dispense Pac

RobotUI		Painting		Cleaning		Sealing		Link	Buttons
Main Menu	Sub Menu	1KP CCV /1KP RCC/ 2KP CCV/ CBS CCV	PP	FD	SWB	IDFP	DP		
View	Station view	☒	☒	☒	☒	☒	☒	'Station view' on page 153	 Station
	Zone overview	☒	☒	☒	☒	☒	☒	'Zone overview' on page 159	 Overview
	Production status	☒	☒	☒	☒	☒	☒	'Production status' on page 159	 Dailycount
	Vision system status	☒	☒	☒	☒	☒	☒	'Vision system status (VMT)' on page 160	 Vision

RobotUI		Painting		Cleaning		Sealing		Link	Buttons
Robot	Position	☒	☒	☒	☒	☒	☒	'Positions' on page 164	 Position
	Application	☒	☒	☒	☒	☒	☒	'Application' on page 169	 Application
	Function-Painting	☒	☐	☐	☐	☐	☐	'Function - Painting' on page 187	 Function
	Function-PixelPaint	☐	☒	☐	☐	☐	☐	'Function - PixelPaint' on page 195	 Function
	Function-Dispensing	☐	☐	☐	☐	☒	☒	'Function - Dispensing' on page 199	 Function
	Backup	☒	☒	☒	☒	☒	☒	'Backup' on page 200	 BackUp
	Reset Program Pointer	☒	☒	☒	☒	☒	☒	'Reset Program Pointer (PP)' on page 204	 ResetPP
	Brush Editor	☒	☒	☐	☐	☐	☐	'Brush Editor' on page 205	 Brush Edit
	Log Refresh	☒	☒	☒	☒	☐	☐	'Refresh Log (command-IO)' on page 208	 LogRefresh
	Shaftlock	☒	☐	☐	☐	☐	☐	'Shaftlock - Bell (paint robot)' on page 209	 ShaftLock

6 HMI system

6.1 HMI Overview

RobotUI		Painting		Cleaning		Sealing		Link	Buttons
Control	FIFO	☒	☒	☒	☒	☒	☒	'FIFO' on page 211	
	Simulation	☒	☒	☒	☒	☒	☒	'Simulation' on page 214	
	Bypass	☒	☒	☒	☒	☒	☒	'Bypass' on page 216	
	Opener	☒	☐	☐	☐	☐	☐	'Opener' on page 217	
Setting	Type	☒	☒	☒	☒	☒	☒	'Type' on page 219	
	Colour	☒	☒	☒	☒	☐	☐	'Colour (painting)' on page 220	
	Skid slip sensor check	☒	☐	☐	☐	☐	☐	'Skid slip sensor check (line tracking station)' on page 221	
	Parameter	☒	☐	☐	☐	☐	☐	'Parameter' on page 222	
	Alarm	☒	☒	☒	☒	☒	☒	'Alarm' on page 223	
	Security	☒	☒	☒	☒	☒	☒	'Security' on page 225	
	Miscellaneous	☒	☒	☒	☒	☒	☒	'Miscellaneous' on page 227	

RobotUI		Painting		Cleaning		Sealing		Link	Buttons
Diagnose	Alarm	☒	☒	☒	☒	☒	☒	'Alarm' on page 229	 Alarm
	WAN	☒	☒	☒	☒	☒	☒	'WAN (Factory Network)' on page 230	 LAN
	Bus	☒	☒	☒	☒	☒	☒	'Bus' on page 230	 Bus
	Mixroom	☒	☒	☒	☒	☒	☒	'Mixroom' on page 230	 Mixroom
Report	Production	☒	☒	☒	☒	☒	☒	'Production' on page 233	 Production
	Consumption	☒	☒	☒	☒	☒	☒	'Consumption' on page 234	 Consumption
	Operation	☒	☒	☒	☒	☒	☒	'Operation' on page 234	 Operation
	Alarm	☒	☒	☒	☒	☒	☒	'Alarm' on page 235	 Alarm
	Command Log	☒	☒	☒	☒	☒	☒	'Command log' on page 236	 Command
Program	RobotStudio	☒	☒	☒	☒	☒	☒	'RobotStudio' on page 237	 Robot Studio
	RobView	☒	☒	☒	☒	☒	☒	'RobView' on page 237	 Robview

RobotUI		Painting		Cleaning		Sealing		Link	Buttons
Help	RobotUI	☒	☒	☒	☒	☒	☒	'RobotUI' on page 239	
	Robot	☒	☒	☒	☒	☒	☒	'Robot' on page 239	
	Exit	☒	☒	☒	☒	☒	☒	'Exit' on page 239	

Note:

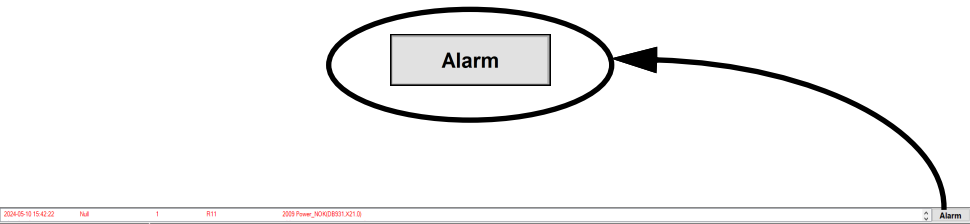
1. For more information, see Security Chapter 6.5.6 on how users can manage menu access levels.
2. Control - Shaftlock is for painting robot with bell cup.
3. Control - Opener is for paint station with door or cover opener robot.
4. Setting - Skid slip sensor check is for line tracking station.

6.1.4 Alarm Line

The alarm line shows the latest alarm details, including time, operator name, shift, error source from PLC or Robot, and alarm description.

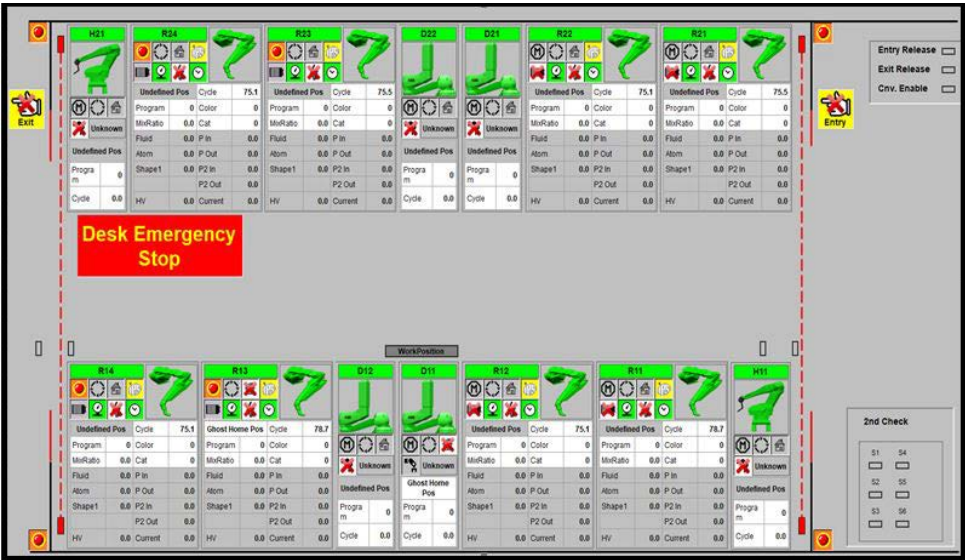
Tapping the “Alarm” button on the alarm line in the RobotUI directly switches to the alarm interface.

For more information on alarm diagnose, see chapter 6.6.1.



6.1.5 Main window

The main window displays the most crucial information (e.g. station overview). Depending on the representation, it is shown in conjunction with an additional window or as a total view.



Application Type	Main Window
FD (Feather Duster)	For main windows, see Appendix 'F (6.1.5)' on page 430.
PixelPaint	
Sealing	

6.1.6 Additional window

The additional window typically displays the status or control functions for the main window.

Usually, the sub-menus in “View”, “Robot”, and “Control” include additional windows.

The following table contains information about additional windows.

Main Menu	Sub Menu
View	Station view
	Zone overview
	Production status
	Vision system status
Robot	Position
	Application
	Function-Painting
	Function-PixelPaint
	Function-Dispensing
	Backup
	Reset Program Pointer
	Brush Editor
	Log Refresh
	Shaftlock
Control	FIFO
	Simulation
	Bypass
	Opener

View:Robot

Control Unit

Interface State

Connected

Controller State

Motor Off

Operating Mode

Auto Mode

Master Mode

PLC Master

Speed Ratio

100%

Job Unit

Job Status

Stopped

Run Chain

Closed

Program No.

0

Last Cycle Time

75.11

CCV Unit

Current Material

0

Fill Material

0

Brush Table

0

Material Usage

0.00

H21

R24

R23

D22

D21

R22

R21

R14

R13

D12

D11

R12

R11

H11

View:Robot	Robot: Position	Robot: Reset PP																														
Control Unit Interface State Lost Connection Controller State Init State Operating Mode Auto Mode Master Mode No Master Speed Ratio 0%	☐ Home Position ☐ MtrChPos ☐ CalPos ☐ MaintPos ☐ BypassPos ☐ ManCleanPos ☐ DressPos ☐ User Position ☐ User Position																															
Job Unit Job Status Stopped Run Chain Open Program No. 0 Last Cycle Time 0.00																																
CCV Unit Current Material 0 Fill Material 0 Brush Table 0 Material Usage 0.00																																
	 	<table border="1"> <thead> <tr> <th>Robot</th> <th>Reset PP</th> </tr> </thead> <tbody> <tr><td>R11</td><td>Reset</td></tr> <tr><td>R21</td><td>Reset</td></tr> <tr><td>R12</td><td>Reset</td></tr> <tr><td>R22</td><td>Reset</td></tr> <tr><td>R13</td><td>Reset</td></tr> <tr><td>R23</td><td>Reset</td></tr> <tr><td>D11</td><td>Reset</td></tr> <tr><td>D21</td><td>Reset</td></tr> <tr><td>D12</td><td>Reset</td></tr> <tr><td>D22</td><td>Reset</td></tr> <tr><td>H11</td><td>Reset</td></tr> <tr><td>H21</td><td>Reset</td></tr> <tr><td>H12</td><td>Reset</td></tr> <tr><td>All</td><td>Reset</td></tr> </tbody> </table>	Robot	Reset PP	R11	Reset	R21	Reset	R12	Reset	R22	Reset	R13	Reset	R23	Reset	D11	Reset	D21	Reset	D12	Reset	D22	Reset	H11	Reset	H21	Reset	H12	Reset	All	Reset
Robot	Reset PP																															
R11	Reset																															
R21	Reset																															
R12	Reset																															
R22	Reset																															
R13	Reset																															
R23	Reset																															
D11	Reset																															
D21	Reset																															
D12	Reset																															
D22	Reset																															
H11	Reset																															
H21	Reset																															
H12	Reset																															
All	Reset																															
Please pay attention to whether there are people or objects obstructing																																

The left side of the status bar shows a shortcut for important pages, and the right side shows the most essential station conditions. The icons are collective messages and change regarding their status. The alarm window displays detailed fault messages.





Network

Shows the network status of the robot to PLC.

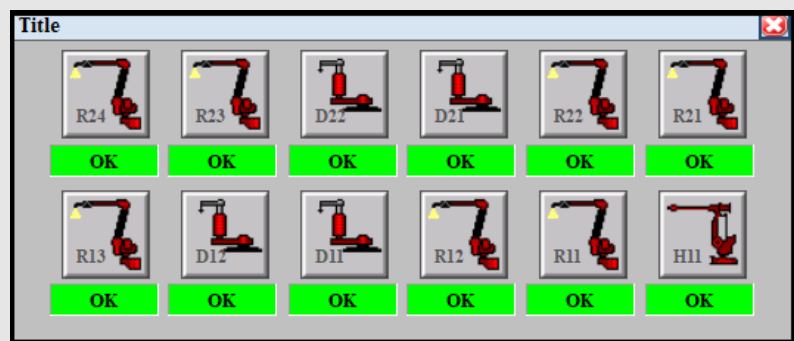
Grey: Indicates the system is down or shows a lost connection.

Green: Signifies that everything is operating normally (OK).

Red: Indicates the presence of an error.

Clicking the red button reveals the robot network status.

Its only Ethernet communication between RobotUI and robot. Its Ethernet/IP.



Fire Alarm

Displays the fire alarm status.

Grey: System is down or there's a lost connection between RobotUI and PLC, not related to certain PLC signals.

Green: Everything is operating normally (OK).

Red: Indicates a problem, not OK.

White: Indicates that the system is bypassed.



Booth (Paint and Bumper available)

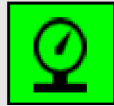
Presents the status of the booth.

Grey: System is down or there's a lost connection between RobotUI and PLC, not related to certain PLC signals.

Green: Everything is operating normally (OK).

Red: Indicates a problem, not OK.

White: Indicates that the booth status is bypassed.



Air Pressure

Depicts the status of the air pressure.

Grey: System is down or there's a lost connection between RobotUI and PLC, not related to certain PLC signals.

Green: Everything is operating normally (OK).

Red: Indicates a problem, not OK.

The lower right corner displays the air pressure value.



E-Stop

Exhibits the emergency stop status.

Grey: System is down or there's a lost connection between RobotUI and PLC, not related to certain PLC signals.

Green: Indicates that the station is not in emergency stop status.

Red: Indicates that the station is in emergency stop status, clicking the red button reveals the detailed emergency stop status.

Desk EStop	NO		R11	R21
Entry Left EStop	NO		R12	R22
Entry Right EStop	NO		R13	R23
Exit Left EStop	NO		R14	R24
Exit Right EStop	NO		D11	D21
Conveyor EStop	NO		D12	D22



Safety

Flashes the safety status.

Grey: System is down or there's a lost connection between RobotUI and PLC, not related to certain PLC signals.

Green: Station is in a safe status.

Red: Station is not in a safe status, clicking the red button shows the detail safety status.

White: Some station safe status is bypassed.

Safety Signal Detail	
Entry Light Barrier	OK
Exit Light Barrier	OK



Conveyor

This symbol presents the status of the conveyor system.

Grey: System is down or there's a lost connection between RobotUI and PLC, not related to certain PLC signals.

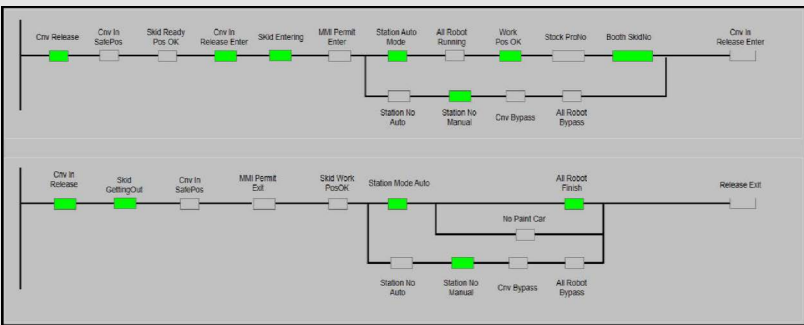
Green: The conveyor is running.

Yellow: The conveyor is stopped.

Red: Indicates an error status with the conveyor.

White: Indicates that the conveyor is bypassed.

Clicking the conveyor button shows the required conditions for the conveyor operation.



Application

Displays the atomizers status.

Grey: System is down or there's a lost connection between RobotUI and PLC, not related to certain PLC signals.

Green: Everything is operating normally (OK).

Red: Indicates a problem, not OK. clicking the red button shows the detail atomizer status.

White: Indicates that the atomizer status is bypassed.

Bell rotation error	R11	R21	R12	R22	R13	R23	R14
Shape1 error	R11	R21	R12	R22	R13	R23	R14
Shape2 error	R11	R21	R12	R22	R13	R23	R14
Servo error	R11	R21	R12	R22	R13	R23	R14
Pump In pressure value too low	R11	R21	R12	R22	R13	R23	R14
Pump In pressure value too High	R11	R21	R12	R22	R13	R23	R14
Pump Out pressure value too low	R11	R21	R12	R22	R13	R23	R14
Pump Out pressure value too high	R11	R21	R12	R22	R13	R23	R14

	Robot Shows the operating status of the robot. Grey: System is down or there's a lost connection between RobotUI and PLC, not related to certain PLC signals. Green: Automatic mode. Blue: Teach mode. Red: Error status. White: The robot status is bypassed.
	PLC Power Shows the status of the PLC power. Grey: The power is off. Green: The power is on.
	Station Mode This symbol appears in automatic mode.
	Station Mode This symbol appears in manual mode.
	Station Mode This symbol appears in maintenance mode.
	Station Mode This symbol appears in clean mode.
	Station Mode This symbol appears in teach mode.
	Station Mode This symbol appears in unknown status.

6.2 View

Following sub-sections are included in this section:

6.2.1 'Station view' on page 153

6.2.2 'Zone overview' on page 159

6.2.3 'Production status' on page 159

6.2.4 'Vision system status (VMT)' on page 160

6.2.4.1 'PixelPaint' on page 160

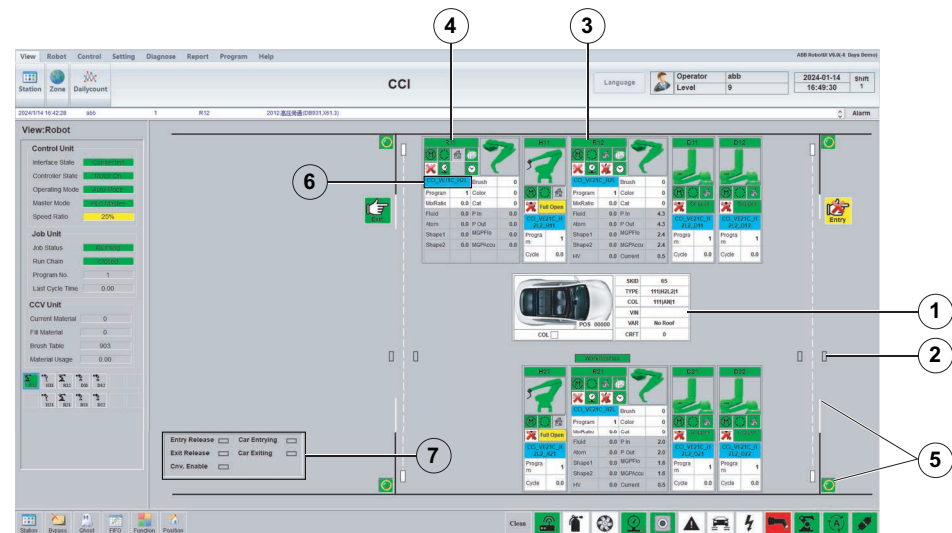
6.2.4.2 'Dispensing' on page 161

6.2.1 Station view

Overview

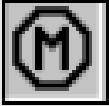
The main window of the station displays all essential station states, while additional windows provide detailed status information for individual robots.

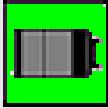
The following figure shows some examples:



Pos.	Description
1	Car body data (type, colour, status [enabled/disabled] etc.)
2	Status of the individual station devices
3	Status of the application system
4	Robot status and position reports
5	Status of the station safety devices (e.g. doors, emergency stops, etc.)
6	Position reports
7	Conveyor status

Symbols

	Robot name Displays robot name and background colour. Green: The station is in automatic mode. Blue: The station is in manual mode. Grey: The station is in an unknown mode.
	Robot main status Red: Indicates a robot error. Green: The robot is in automatic mode. Blue: The robot is in manual mode. White: The robot is bypassed. Yellow: The robot is disconnected. Grey: Indicates an unknown status for the robot.
	Robot motor status Green: The motor is on. Grey: The motor is off.
	Robot cycle status Green: The cycle is on. Grey: The cycle is off.
	Robot home position Green: The robot is at the home position. Grey: The robot is not at the home position.
	Robot IPS status (only paint robot) Green: The IPS is connected. Grey: The IPS is disconnected. Yellow: Indicates an IPS error.

 or 	Application status (only paint robot) Green: The application is okay. Red: Indicates an error in the application. White: The application is bypassed. Yellow: Indicates a shaft lock in the application. Grey: Shows an unknown status of the application.
	Air pressure status (only paint robot) Green: The air pressure is okay. Red: An error in the air pressure. Grey: An unknown status of the air pressure.
	High voltage status (only paint robot with high voltage) Green: The high voltage is okay. Red: Indicates an error in the high voltage. White: The high voltage is bypassed. Grey: Indicates an unknown status of the high voltage.
	Potlife status (only paint robot with potlife) Green: The potlife is okay. Red: Indicates an error in the potlife. Yellow: Shows a warning related to the potlife. Grey: Flashes an unknown status of the potlife.

Paint Robot

Undefined Pos		Cycle	75.1
Program	0	Color	0
MixRatio	0.0	Cat	0
Fluid	0.0	P In	0.0
Atom	0.0	P Out	0.0
Shape1	0.0	P2 In	0.0
		P2 Out	0.0
HV	0.0	Current	0.0

Dispensing

Home Pos	Cycle	0.0	
Program	0	Skid	0
Fluid	0.0	P In	0.1
Consum	0.0	P Out	-1.2
Templn	29.7	TempOut	31.0
OffSetX	0.0	OffSetRx	0.0
OffSetY	0.0	OffSetRy	0.0
OffSetZ	0.0	OffSetRz	0.0

Pixel Paint

Undefined Pos		Brush	0
Program	0	Color	0
Fluid Out	0.000	Fluid In	0.000
Fluid Rtn	0.000	FluidSup	0.000
State Machine		Primed State	

Robot detail data

Displays robot position information, cycle time, program, colour and process data for paint robot, dispensing and Pixel Paint.

Booth car data

Displays booth car information, including:

Ghost mode and **paint** status.

SKID: Skid number.

TYPE: Car type.

COL: Colour type (for paint station only).

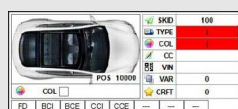
POS: Current position from zero point (for tracking station only).

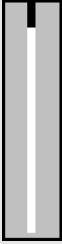
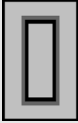
VIN: Vicode.

VAR: Roof class for car body.

CRFT: Two-tone code for car body.

Station Enable: Indicates which station is painted or not.



	Swing door status Green: The swing door is okay. Red: A fault with the swing door. White: The swing door is bypassed.
	Light barrier status Green: The light barrier is okay. Red: A fault with the light barrier. White: The light barrier is bypassed.
	Sensor Muting sensor, station sensor or slip sensor (tracking station). Green: The sensor is on. Grey: The sensor is off.
	Emergency stop status Displays when emergency stop is active.
	Entry/Exit release status (only Stop&Go station) Green: Entry/exit release is active . Yellow: Entry/exit is blocked.

The additional window displays detailed status information for a single robot. Users can switch between robots by clicking on the robot name at the bottom.

View:Robot

Control Unit

Interface State	Connected
Controller State	Motor Off
Operating Mode	Auto Mode
Master Mode	PLC Master
Speed Ratio	100%

Job Unit

Job Status	Stopped
Run Chain	Closed
Program No.	0
Last Cycle Time	75.11

CCV Unit

Current Material	0
Fill Material	0
Brush Table	0
Material Usage	0.00

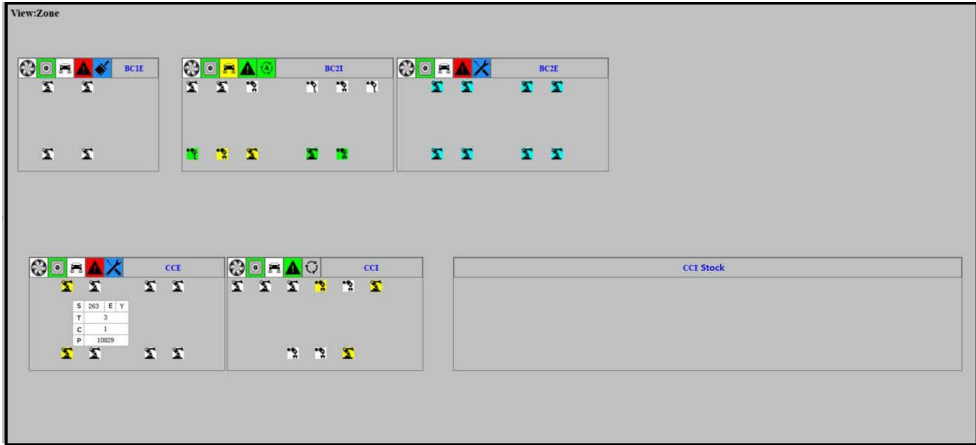
H21	R24	R23	D22	D21	R22	R21
R14	R13	D12	D11	R12	R11	H11

6.2.2 Zone overview

Overview

The zone includes all station information about each robot's status, conveyor status, safety status and painting vehicle information in the booth.

The symbols below indicate the station page and status bar page.

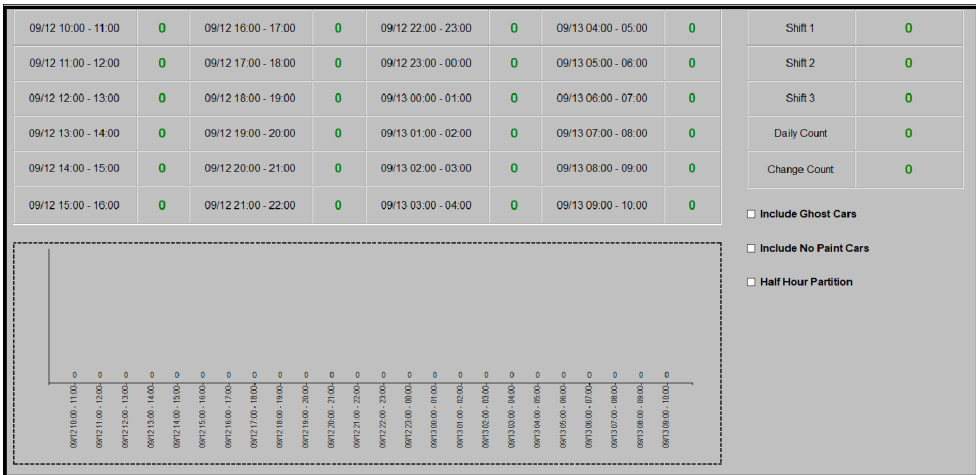


6.2.3 Production status

Overview

The below screen shows detailed production status. It offers an in-depth overview of production status, showcasing production data for each hour and shift. Users can choose to view production details, including instances with no painted cars or ghost cars.

When users select an hour partition, the displayed hour span will appear in the format of 10:30 – 11:30.



6.2.4 Vision system status (VMT)

6.2.4.1 PixelPaint

This image shows the vision system status, both 3D and BK status.

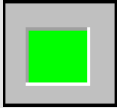
3D Status	Description
Automatic	VMT is in automatic mode.
Simulation	VMT is in simulation mode.
Measurement Running	VMT measurement is running.
Vector Sending	VMT is sending the Vector data.
Type Code	Car Type Code number.

3D Measurement	Description
Direct X	X value of Vector data.
Direct Y	Y value of Vector data.
Direct Z	Z value of Vector data.
Direct Rx	Rx value of Vector data.
Direct Ry	Ry value of Vector data.
Direct Rz	Rz value of Vector data.

BK Status	Description
General Error	General Error of BK sensor.
Measure Error	Measure Error of BK sensor.
Communication Error	Communication Error of BK sensor.

Menu items

Signal status:

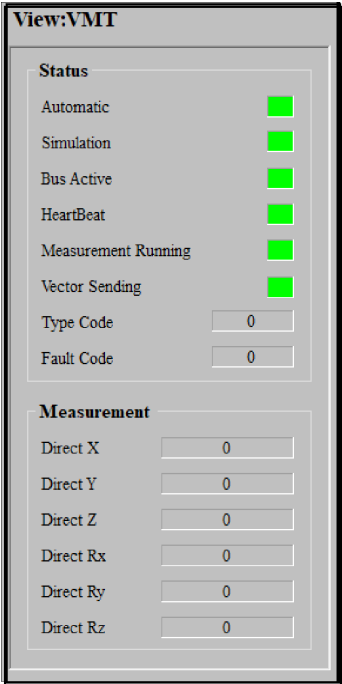
	Green: On (The signal value is 1, signal has been activated).
	Grey: Off.
	Red: Error.

6.2.4.2 Dispensing

This image shows the vision system status.

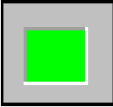
Status	Description
Automatic	VMT is in automatic mode.
Simulation	VMT is in simulation mode.
Bus Active	VMT communication BUS is active.
HeartBeat	Heart Beat status.
Measurement Running	VMT measurement is running.
Vector Sending	VMT is sending the Vector data.
Type Code	Car Type Code number.
Fault Code	VMT Fault Code number.

Measurement	Description
Direct X	X value of Vector data.
Direct Y	Y value of Vector data.
Direct Z	Z value of Vector data.
Direct Rx	Rx value of Vector data.
Direct Ry	Ry value of Vector data.
Direct Rz	Rz value of Vector data.



Menu items

Signal status:



Green: On.

Grey: Off.

6.3 Robot

Following sub-sections are included in this section:

- 6.3.1 ['Positions'](#) on page 164
 - 6.3.1.1 ['Positions - Painting'](#) on page 165
 - 6.3.1.2 ['Positions - PixelPaint'](#) on page 166
 - 6.3.1.3 ['Positions - Dispensing'](#) on page 167
 - 6.3.1.4 ['Positions - Cleaning'](#) on page 168
- 6.3.2 ['Application'](#) on page 169
 - 6.3.2.1 ['Painting'](#) on page 169
 - 6.3.2.2 ['PixelPaint'](#) on page 176
 - 6.3.2.3 ['Dispensing'](#) on page 180
 - 6.3.2.4 ['Cleaning'](#) on page 184
- 6.3.3 ['Function - Painting'](#) on page 187
 - 6.3.3.1 ['Fill and Clean'](#) on page 188
 - 6.3.3.2 ['Flow test'](#) on page 190
 - 6.3.3.3 ['Brush test'](#) on page 191
 - 6.3.3.4 ['Brush factors'](#) on page 192
 - 6.3.3.5 ['High voltage test'](#) on page 192
 - 6.3.3.6 ['IPS Calibration'](#) on page 194
- 6.3.4 ['Function - PixelPaint'](#) on page 195
 - 6.3.4.1 ['Commands'](#) on page 196
 - 6.3.4.2 ['Brush test'](#) on page 196
 - 6.3.4.3 ['PxP Service'](#) on page 197
 - 6.3.4.4 ['Custom'](#) on page 198
- 6.3.5 ['Function - Dispensing'](#) on page 199
 - 6.3.5.1 ['Function - IDFP'](#) on page 199
- 6.3.6 ['Backup'](#) on page 200
- 6.3.7 ['Reset Program Pointer \(PP\)'](#) on page 204
- 6.3.8 ['Brush Editor'](#) on page 205
- 6.3.9 ['Refresh Log \(command-IO\)'](#) on page 208

6.3.10 '[Shaftlock - Bell \(paint robot\)](#)' on page 209

6.3.11 '[Interlock](#)' on page 209

Overview

This chapter provides an overview of the functionalities supported by the Robot User Interface.

The below table describes each main menu.

Menu	Description
View	This menu is used to monitor the station, zone booth, and robot status.
Robot	This menu allows for manual command sending to the robot and modification of the robot file, primarily operating in station manual mode.
Control	This menu is designed for controlling the station FIFO and device, predominantly operating in station automatic mode.
Setting	This menu is dedicated to setting station parameters.
Diagnose	This menu displays the running status of the station.
Report	This menu presents historical data of production, operation, and alarm.
Program	This menu provides links to other software.
Help	This menu offers links to help documents and software exit options.

6.3.1 Positions

General

HMI can control the position of each robot under certain conditions.

Operation conditions:

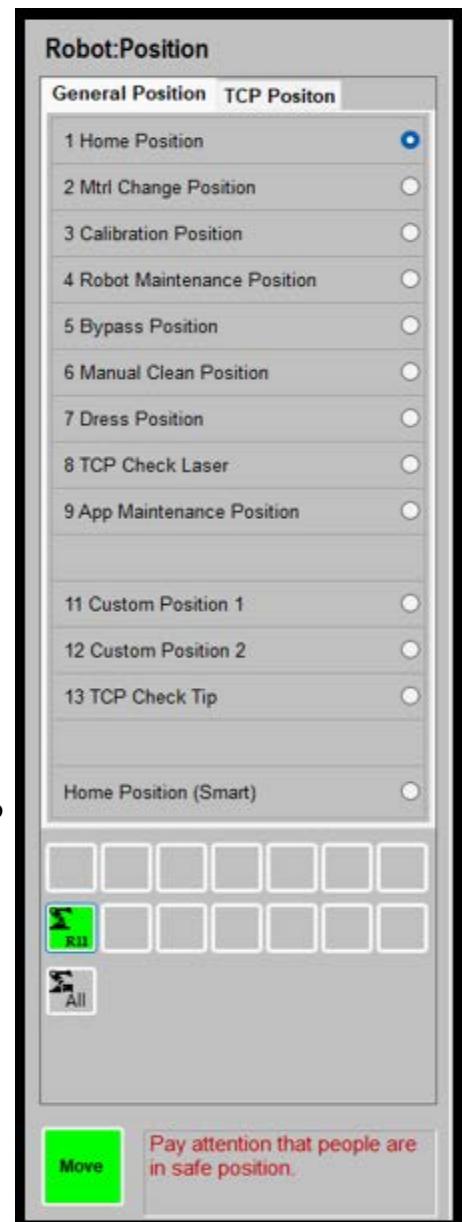
1. Station manual mode.
2. Robot automatic mode.
3. Robot motor on and cycle on.
4. The robot must be in a home position before moving to a non-home position.

Steps:

1. Select a defined position.
2. Select a robot.
3. Click the “move” button.

The RobotUI software will display a pop-up window to alert the operator when prerequisites are not fulfilled, helping the operator in troubleshooting.

A brief introduction to the paint robot position is in chapter 7.3.1.2, ‘[Service position command](#)’ on page 260, which contains a detailed introduction to each position.



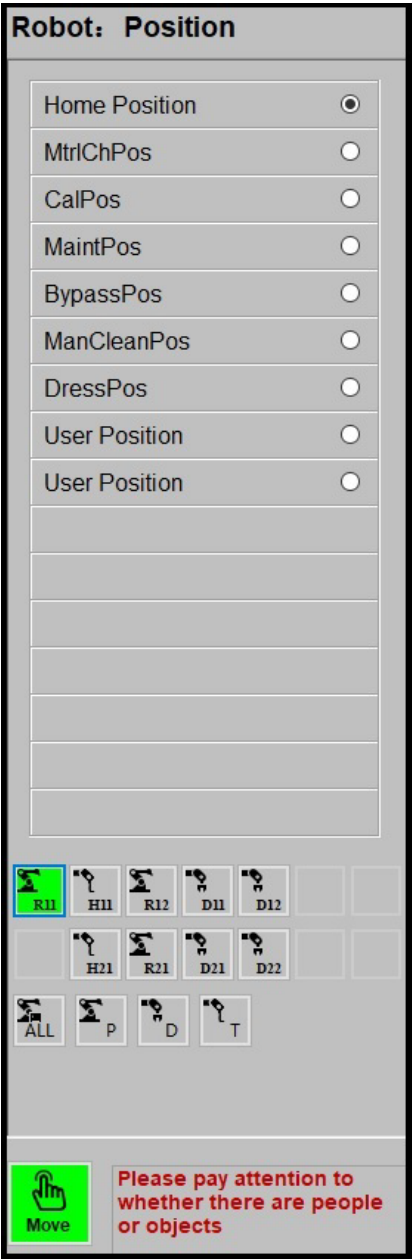
WARNING!

Before moving the robot position, check whether any people or objects are in the station.

6.3.1.1 Positions - Painting

Overview

This window facilitates the movement of multiple assigned robots from their home positions to defined positions as needed.



WARNING!

Before moving the robot position, check whether any people or objects are in the station.

Overview

This window facilitates the movement of multiple assigned robots from their home positions to specified positions, both general and TCP (Tool Center Point).

[illegible]

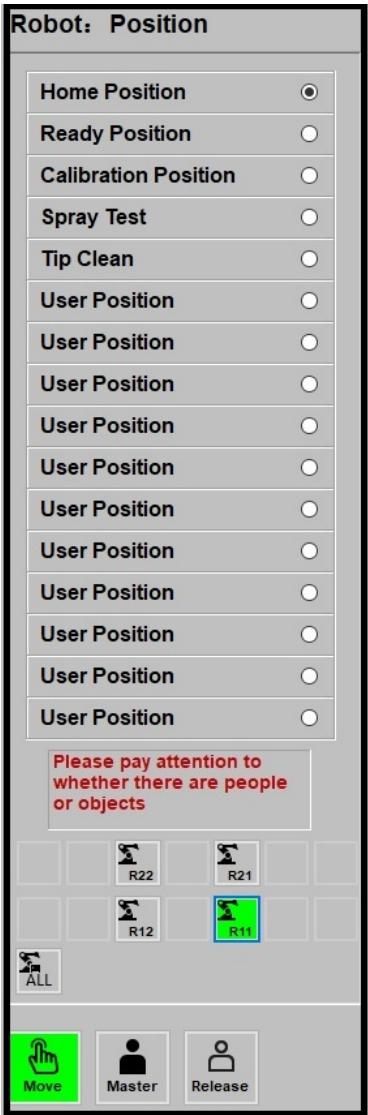
Before moving the robot position, check whether any people or objects are in the station.

6.3.1.3 Positions - Dispensing

Overview

Dispensing robot must get master before moving and release master after moving. This window facilitates the movement of multiple assigned robots from their home positions to defined positions as needed.

The dispense robot system and paint robot system are different. Paint robots and other robots with paintware systems receive the command-IO commands; in station manual and robot automatic mode, RobotUI can send commands to the robots. The dispensing robot receives an IO signal, and RobotUI needs to get a master from the robot before controlling the robot's position.



WARNING!



Before moving the robot position, check whether any people or objects are in the station.

6.3.1.4 Positions - Cleaning

Overview

This window facilitates the movement of multiple assigned robots from their home positions to defined positions as needed.

FD position and SB position

WARNING!



Before moving the robot position, check whether any people or objects are in the station.

6.3.2 Application

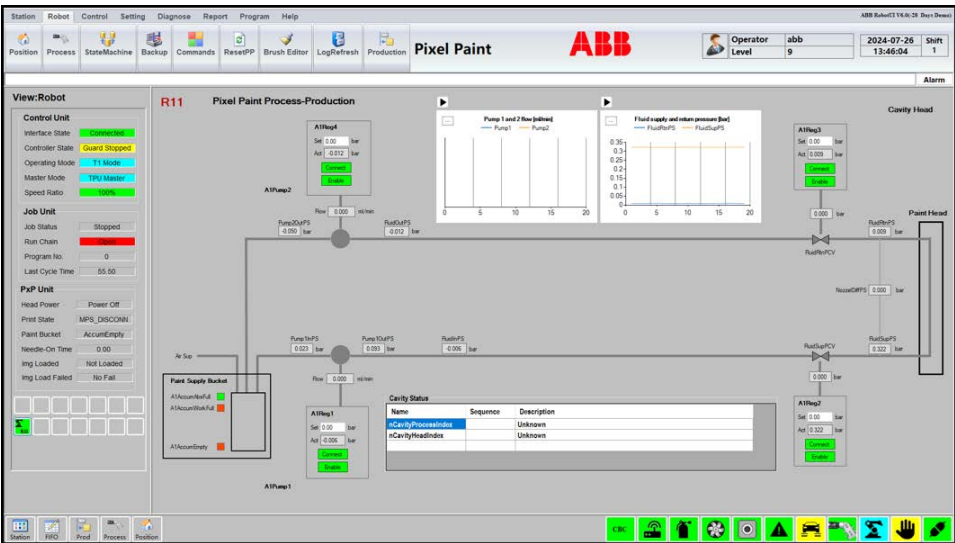
General

The fluid system page shows the 2D robot paint system pipe and valves. Users can open or close the valve but must first log into the system under certain conditions which will be discussed in operation manual.

Operation conditions:

- 1. Station manual mode.
- 2. Robot automatic mode.

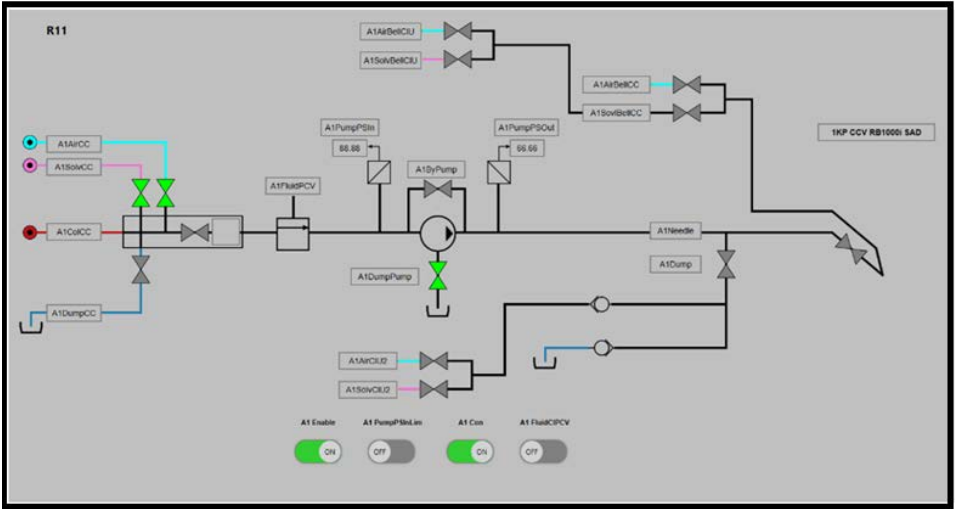
In the simplified process view, all valves are incorporated. This window offers a more comprehensible system overview, promoting operational efficiency.



6.3.2.1 Painting

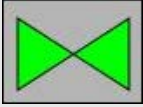
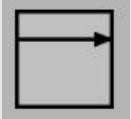
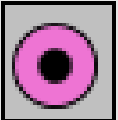
1K paint system

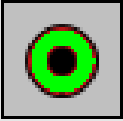
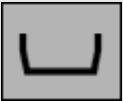
This window displays the process and management of the 1KP CCV paint mixture flow.



Symbols

1K paint system

	Valve Shows the operating status of the valve. Green: The valve is open. Grey: The valve is close.
	Pump Displays the pump inlet pressure and outlet pressure, as well as the fluid pumped. Green: The pump is in operation. Grey: The pump is not in operation.
	Atomizer signal The atomizer signal status is indicated as either "on" or "off."
	PCV Shows the operating status of pressure control valve. Green: The valve is open. Grey: The valve is close.
	Air Supply Schematic diagram of the air supply.
	Solvent Supply Schematic diagram of the solvent supply.

	Paint Supply Schematic diagram of the paint supply.
	Dump Schematic diagram of dump.

General

Users can observe the atomizer status in the interface below.

The process data can be manually set and fluid tests can be conducted under certain conditions.

Operation conditions:

1. Station manual mode.
2. Robot automatic mode.

Steps:

1. Select a robot.
2. Click the “Save” button.

View: Dispense System

App Parameter	Set	Actual
Fluid(ml)	0	0.0
Atom(nl)	0	0.0
Shape1(nl)	0	0.0
Shape2(nl)	0	0.0
HV(KV)	0	0.0
HVCur(μA)		0.0

R11

R12

R13

R14

R21

R22

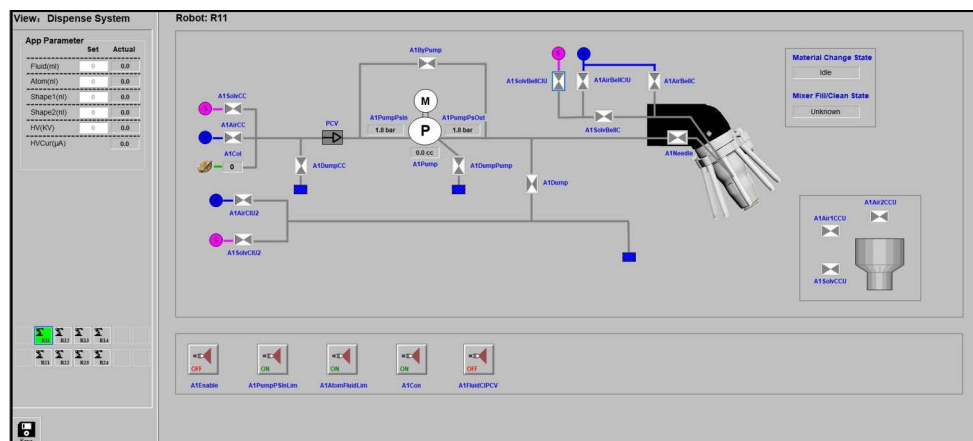
R23

R24


Save

1KP CCV process

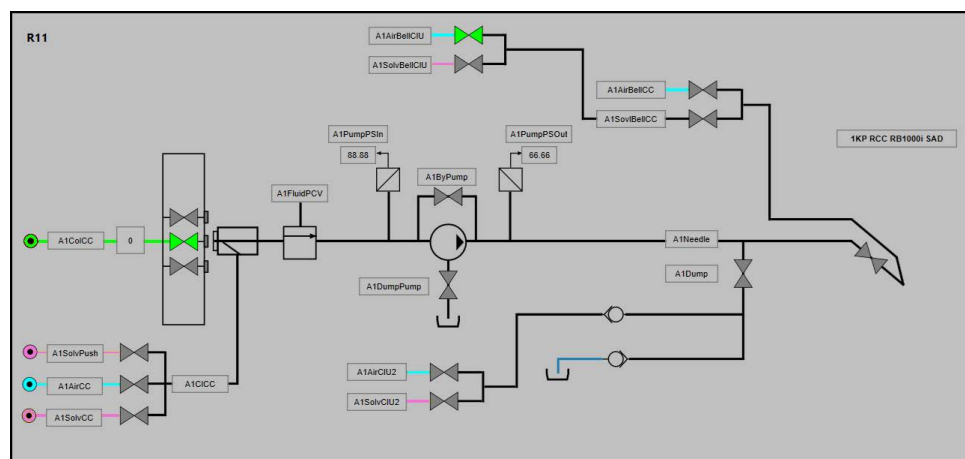
Users can select a robot to switch between robots. The left side displays the set value and actual value of the atomizer parameters, while the right side shows a system picture of the atomizer and clean bucket.

**Symbols**

For more information on Valve, Pump, Pipe and Atomizer signal symbols, see '[1K paint system](#)' on page 170.

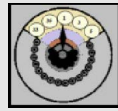
1KP RCC process

Users can click on the robot name to switch between robots. On the left side, the interface displays the set value and actual value of the atomizer parameters, while the right side shows a system picture of the atomizer and clean bucket.



Symbols

For more information on Valve, Pump, and Atomizer signal symbols, see '[1K paint system](#)' on page 170.

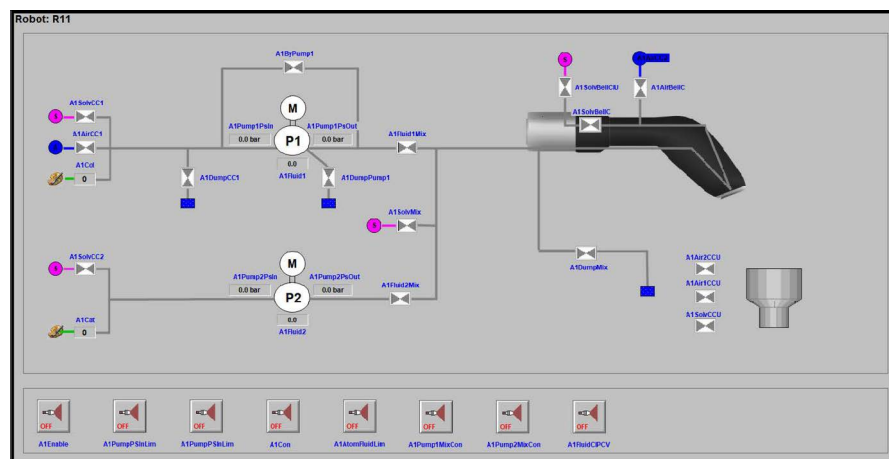


RCC Valve

Shows the current RCC valve number.

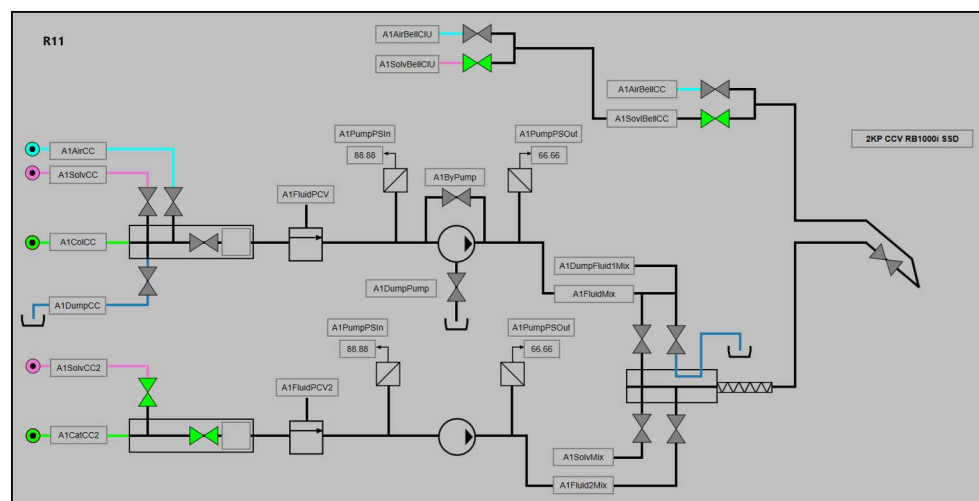
2K paint system

The operation method of the 2K paint system interface is identical to that of the 1K paint system.



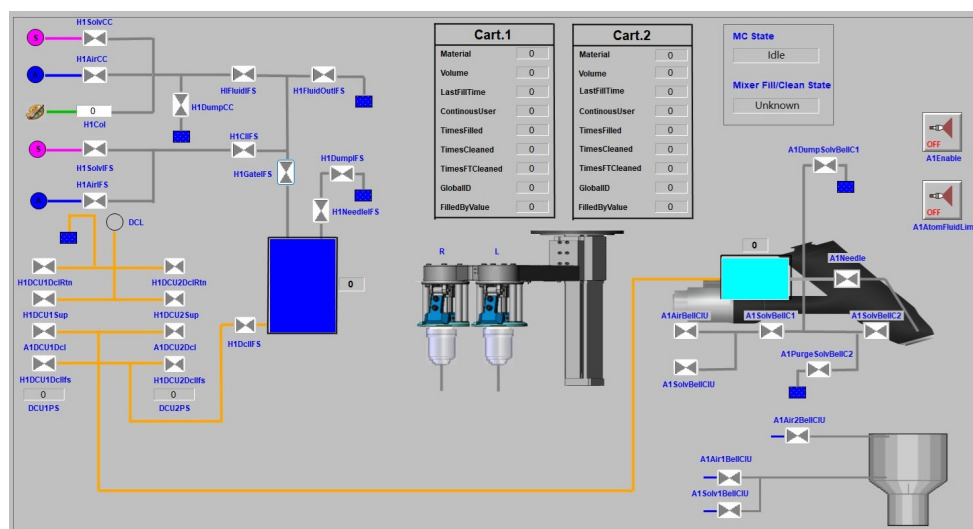
2KP CCV process

Users can click on the robot name to switch between robots. The left side displays the set value and actual value of the atomizer parameters, while the right side shows a system picture of the atomizer and clean bucket.

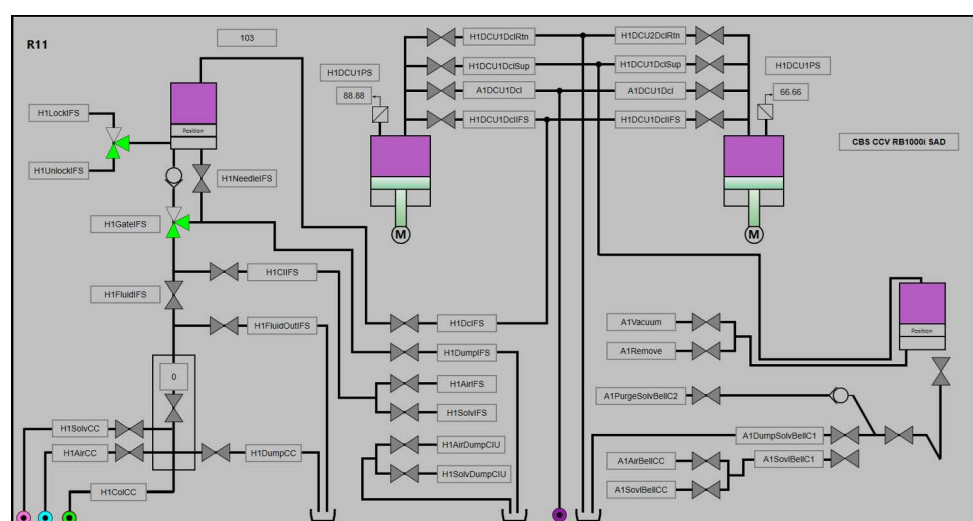


For more information on Valve, Pump and Atomizer signal symbols, see '**1K paint system**' on page 170.

The operation method of the CBS paint system interface mirrors that of the 1K paint system.

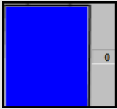
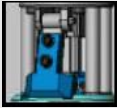
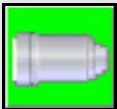
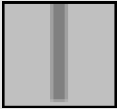


Users can click on the robot name to switch between robots. On the left side, the interface displays the set value and actual value of the atomizer parameters, while the right side shows a system picture of the atomizer and clean bucket.



Symbols

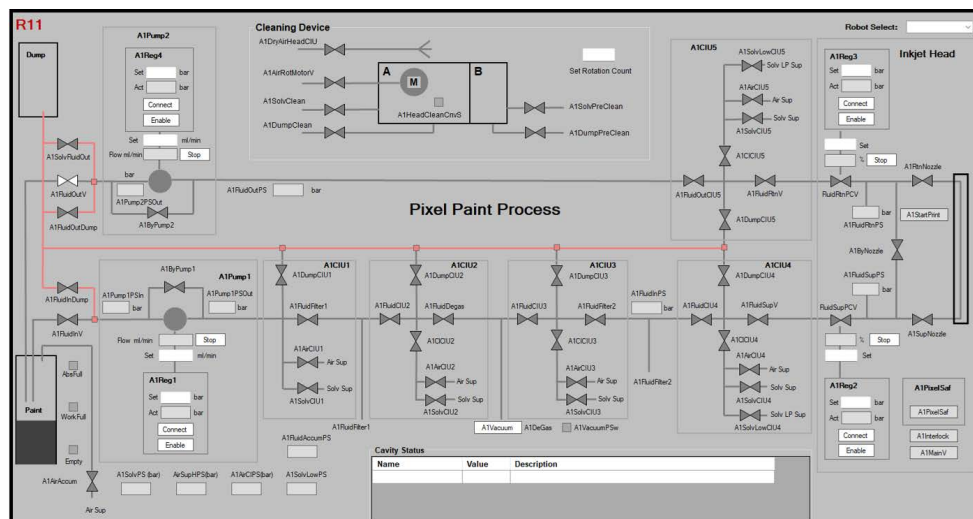
For more information on Valve, Pump, Pipe and Atomizer signal symbols, see '[1K paint system](#)' on page 170.

	Fluid fill Fluid fill level.
	Fluid consumption Fluid consumption level.
	Gripper Gripper status. Blue: The gripper is closed. Green: The gripper is opened. Grey: The gripper is not gripped.
	Cart Cartridge status. Visible: Cartridge is in the gripper. Invisible: Cartridge is not in the gripper.
	Needle Needle status. Green: The needle is open. Grey: The needle is closed.

6.3.2.2 PixelPaint

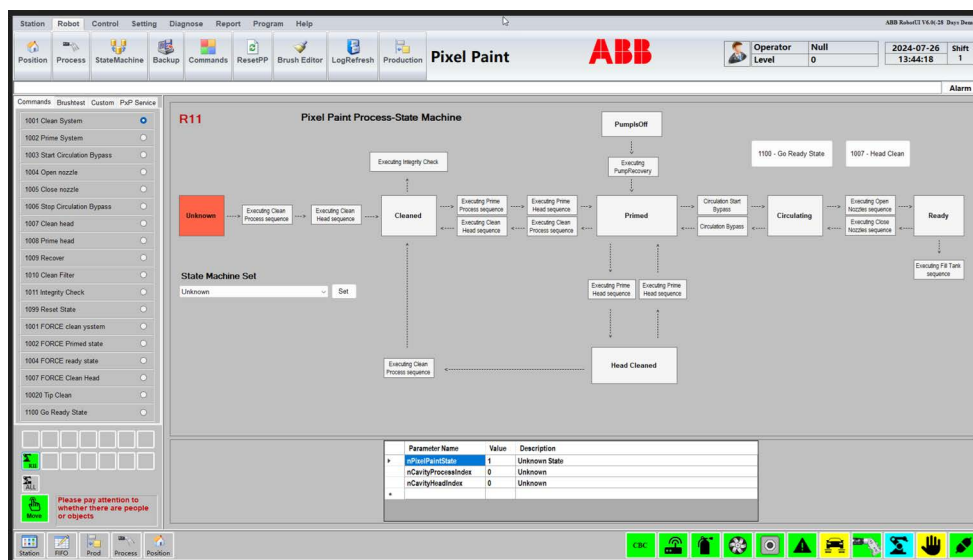
1KPP PxP Process

This image monitors and controls valves, pumps, and sensors. Valves shown in **grey** are **off**, while those in **green** are **on**, as are the pump or other signals. The status below reflects the running state, although it differs slightly from the latest layout. (Note: This image is not updated to the latest version; it is solely for displaying the running status).



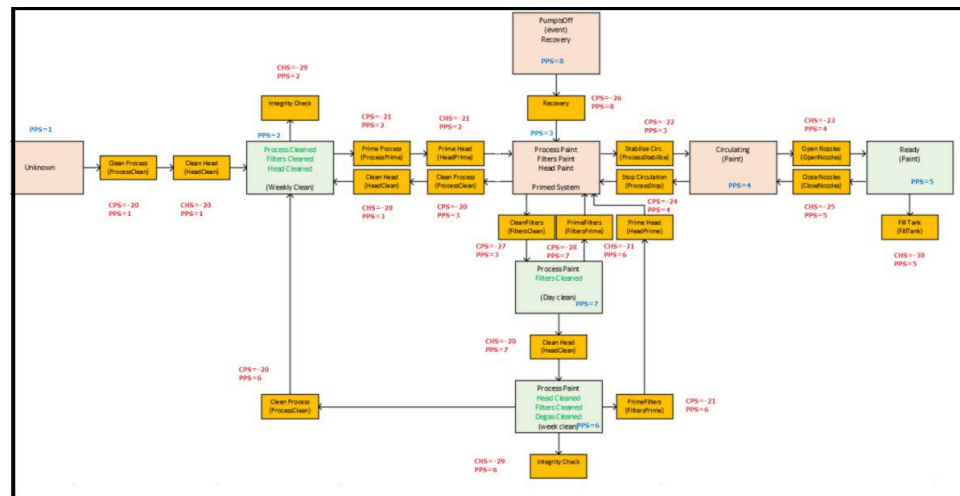
1KPP PxP State machine

The Pixel Paint working status transfer is shown in this image.



The status machine logic operates as follows.

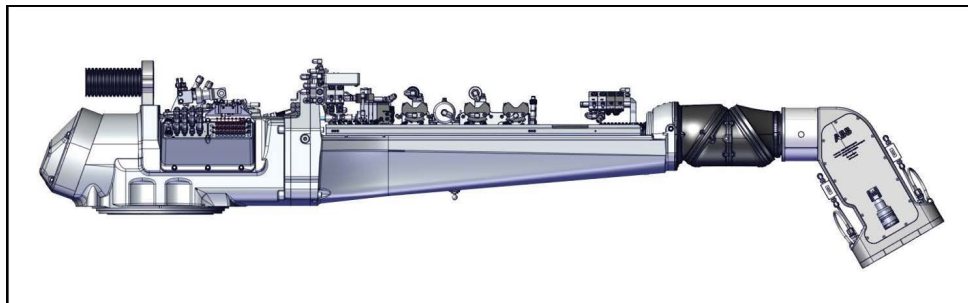
6.3 Robot



Parameter

Values	Name	Description
0	UNKNOWN	Unknown
-15	CAV_FILLED_WATER	Filled water
-16	CAV_FILLED_COLOUR	Filled colour
-17	CAV_STABILIZED_COLOUR	Stabilized colour
-18	CAV_IS_READY	Ready
-19	RESERVE	Status not in use
-20	TRS_FILL_WATER	Filling water
-21	TRS_FILL_COLOUR	Filling colour
-22	TRS_STABILIZING_COLOUR	Stabilizing colour
-23	TRS_GETTING_READY	Getting ready
-24	TRS_STOPPING_CIRCULATION	Stopping circulation
-25	TRS_CLOSING_NOZZLE	Closing nozzle
-26	TRS_RECOVER_CIRCULATION	Recover circulation
-27	TRS_FILTER_CLEAN	Filter clean
-28	TRS_FILTER_PRIME	Filter prime
-29	TRS_INTEGRITY_CHK	Integrity check
-30	TRS_FILL_TANK	Fill tank

Head Cavities:



Value	Name	DESCRIPTION
0	UNKNOWN	Unknown
-15	CAV_FILLED_WATER	Filled water
-16	CAV_FILLED_COLOR	Filled color
-17	CAV_STABILIZED_COLOR	Stabilized color
-18	CAV_IS_READY	Ready
-19	reserve	Status not in use
-20	TRS_FILL_WATER	Filling water
-21	TRS_FILL_COLOR	Filling color
-22	TRS_STABILIZING_COLOR	Stabilizing color
-23	TRS_GETTING_READY	Getting ready
-24	TRS_STOPPING_CIRCULATION	Stopping circulation
-25	TRS_Closing_Nozzle	Closing nozzle
-26	TRS_RECOVER_CIRCULATION	Recover circulation
-27	TRS_FILTER_CLEAN	Filter clean
-28	TRS_FILTER_PRIME	Filter prime
-29	TRS_INTEGRITY_CHK	Integrity check
-30	TRS_FILL_TANK	Fill tank

head cavities.

Figure 29 Cavity overview



(1)nCavityProcessIndex -- Process Cavity(CPS)

(2)nCavityHeadIndex -- Head Cavity(CHS)

Process cavity(CPS) and head cavity(CHS):

This is the current operational instance. The main state, represented by the 'ready' state, is highlighted in green, while the sub-state is depicted in dark grey, as observed in the 'managing close nozzles' and 'executing fill tank' sequences. The arrow's direction signifies the state transition; during this shift, the substrate is highlighted in dark grey.

Ready Button: This button sends a command to the robot, prompting it to switch its state to 'Ready'.

Clean Button: This button dispatches a command to the robot to perform a cleaning operation.

(Note: This image is not the most recent one, it is merely to demonstrate the running status).

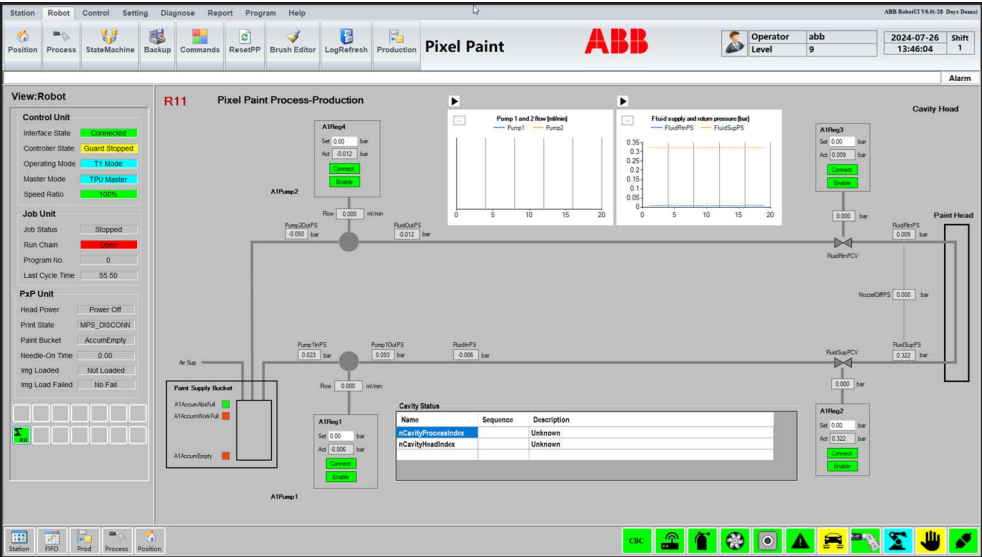
6 HMI system

6.3 Robot

1KPP PxP Production

Changes in the production state are directed by an arrow; when transitioning from one state to another state, the **sub-state is highlighted in dark grey**.

Two charts are available to display the live value of pressure in the below image.



For Information about

1KPP PxP Process

1KPP PxP State machine

1KPP PxP Production

See

HMI on chapter 6.3.2.2 PixelPaint

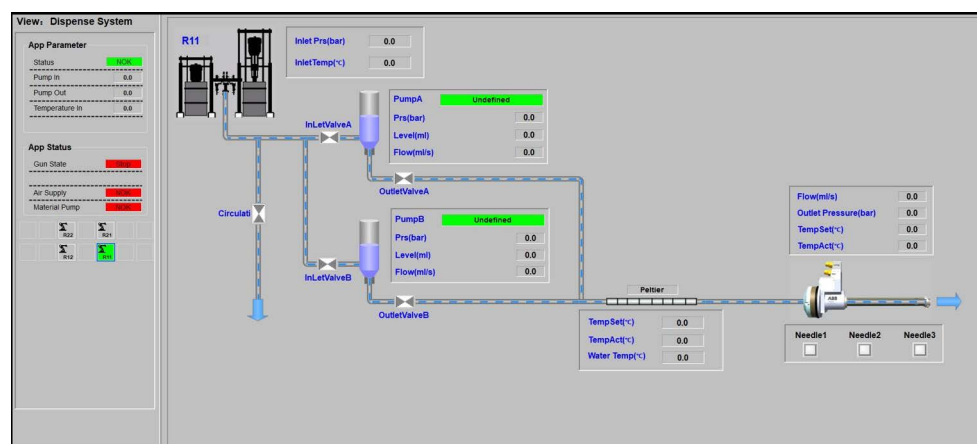
6.3.2.3 Dispensing

Dispense IDFP process

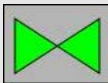
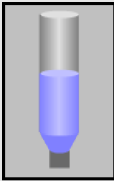
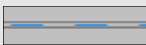
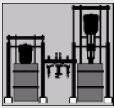
Users can click on the robot name to switch between robots. The left side displays the actual value of the dispense system parameters and dispense status, while the right side shows a system picture of the dispense.

This window displays the process and management of the dispensing system.

Dispensing includes two systems: the IDFP system and the dispense pac system, which are discussed in the sub chapter.



Symbols

	Valve Green: The valve is open. Grey: The valve is closed.
	Doser pump Shows the doser pump status, pressure, glue level and flow speed.
	Pipe Calculates the pipeline flowing animation based on valve status and glue supply status.
	Inlet Pump Indicates inlet pump status, inlet pressure, and glue temperature.

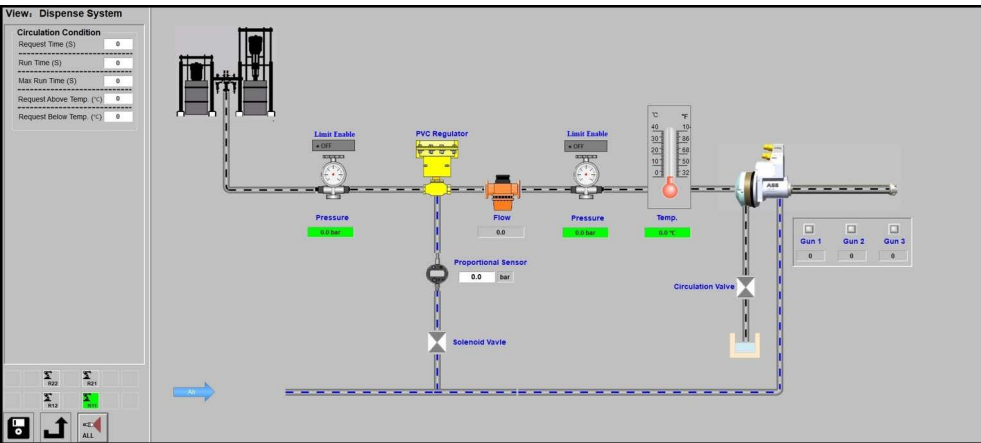
6 HMI system

6.3 Robot

	Peltier Flashes the Peltier temperature set value and actual value, and presents the Peltier heating on/off status.
	Glue gun Displays the speed at which glue is dispensed from the glue gun, outlet pressure, temperature set value and actual value.
	Needle Represents the needle on/off status.

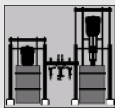
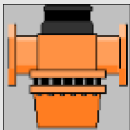
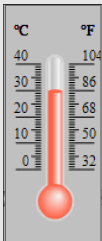
Dispense PAC process

Users can click on the robot name to switch between robots. On the left side, the dispense circulation set value is displayed, while the right side shows a system picture of the dispense.



Symbols

	Valve Green: The valve is open. Grey: The valve is closed.
	Pipe Calculates the pipeline flowing animation based on valve status and glue supply status.

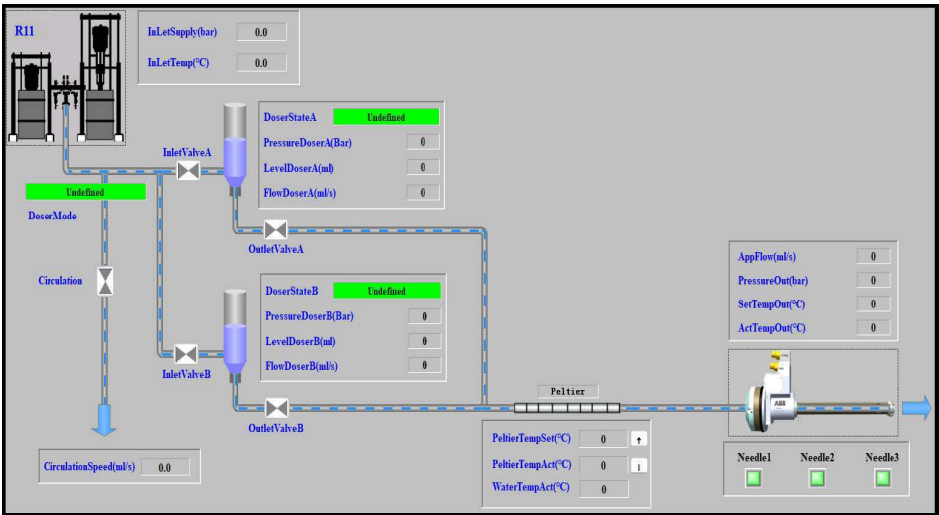
	Inlet Pump Shows the inlet pump picture.
	Pressure sensor Displays the pipe pressure value and the pressure limit enable signal.
	PVC Regulator Shows the PVC regulator picture.
	Proportional Sensor Displays the proportional sensor value.
	Flow meter Indicates the flow meter value.
	Temperature sensor Flashes the temperature sensor value.
	Glue gun Illustrates a glue gun picture.
	Needle Represents needle on/off status and needle on/off delay time.

6 HMI system

6.3 Robot

IDFP sealing system

The operation method of the IDFP sealing system interface mirrors that of the 1K paint system.



Dispense Pac sealing system

The operation method of the dispense pac sealing system interface mirrors that of the 1K paint system.

App Parameter

Status

OK

Pump In

0.0

Pump Out

0.0

Temperature In

0.0

Temperature Out

0.0

App Status

Gun State

OK

Temperature Condition

OK

Air Supply

OK

Material Pump

OK

Related Information

For Information about	See
Dispense IDFP process	Reference manual - HMI on chapter 6.3.2.3 Dispensing
Dispense PAC process	
IDFP sealing system	
Dispense Pac sealing system	

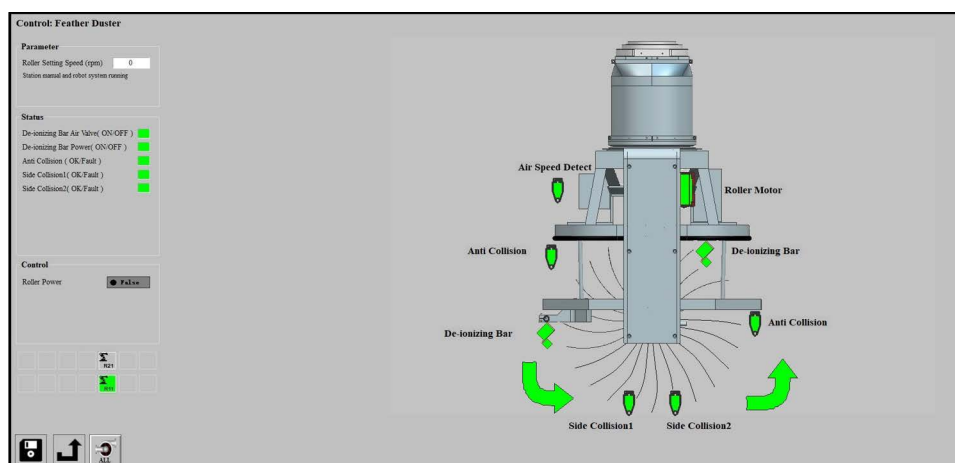
6.3.2.4 Cleaning

Cleaning FD process

Users can click on the robot name to switch between robots. On the left side, the interface displays the feather duster roller setting speed sensor status and roller power status, while the right side shows a system picture of the feather duster.

This window displays the process and management of the cleaning system.

Cleaning includes a feather duster system and a sword brush method; introduce the two systems in the sub-chapter.



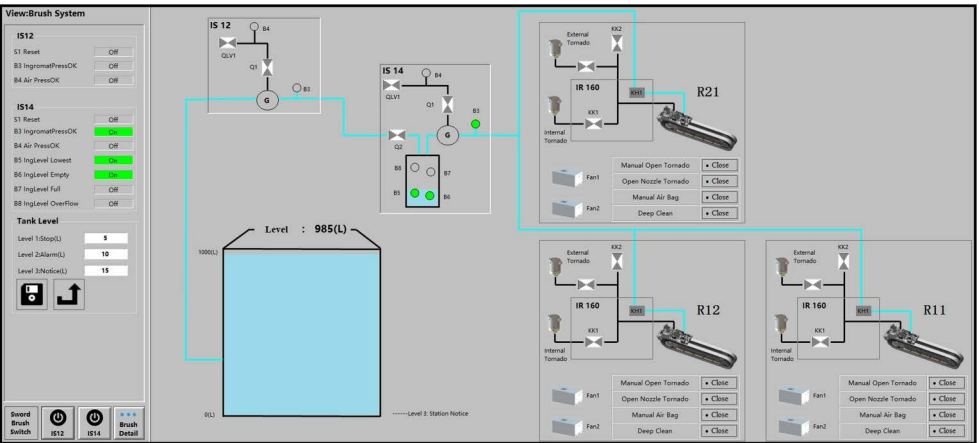
Symbols

	Roller motor Green: The roller is running. Grey: The roller is stopped.
	Detect sensor Green: The detect sensor feedback is safe. Grey: The detect sensor feedback is in alarm.

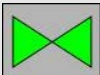
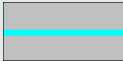
	De-ionizing bar power Green: The de-ionizing bar power is on. Grey: The de-ionizing bar power is off.
	Roller running Visible when the roller is running.

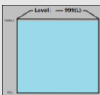
Cleaning SB process

The left side displays the status of the sword brush IS12, IS14, and cleaning liquid tank, while the right side shows a system picture of the sword brush.



Symbols

	Valve Green: The valve is open. Grey: The valve is closed.
	Pipe Displays the picture of the cleaning liquid pipe.
	Level sensor Green: The level sensor is detected. Grey: The level sensor is not detected. Red: Indicates a high-level alarm.

	Tornado Green: The tornado is on. Grey: The tornado is off.
	Fan Green: The fan is on. Grey: The fan is off.
	Sword brush Shows the picture of the sword brush. Green: The sword brush air bag is on. Grey: The sword brush air bag is off.
	Cleaning liquid tank Displays the cleaning liquid tank value.
	Pump Flashes the pump picture.

6.3.3 Function - Painting

Overview

Users can do the robot function with certain conditions.

Operation conditions:

1. Station manual mode
2. Robot automatic mode.

Steps:

1. Select function.
2. Select robot.
3. Click  button to send the command to the robot.
4. Click  button if you need to do a flow test.

WARNING!



Before doing the function, check whether there are any people or objects in the station.

Fill	Clean	Fluid	Calib	User																						
<table border="1"><thead><tr><th>Fill And Clean</th><th>Color</th></tr></thead><tbody><tr><td>Material Fill</td><td>☐ 1</td></tr><tr><td>Air Fill</td><td>☐</td></tr><tr><td>Solvent Fill</td><td>☐</td></tr><tr><td>Bell Clean</td><td>☐</td></tr><tr><td>Hub Clean</td><td>☐</td></tr><tr><td>All Cavity Clean</td><td>☐</td></tr><tr><td>Cat Fill</td><td>☐ 1</td></tr><tr><td>Mixer Clean</td><td>☐</td></tr><tr><td>Color Clean</td><td>☐</td></tr><tr><td>Cat Clean</td><td>☐</td></tr></tbody></table>					Fill And Clean	Color	Material Fill	☐ 1	Air Fill	☐	Solvent Fill	☐	Bell Clean	☐	Hub Clean	☐	All Cavity Clean	☐	Cat Fill	☐ 1	Mixer Clean	☐	Color Clean	☐	Cat Clean	☐
Fill And Clean	Color																									
Material Fill	☐ 1																									
Air Fill	☐																									
Solvent Fill	☐																									
Bell Clean	☐																									
Hub Clean	☐																									
All Cavity Clean	☐																									
Cat Fill	☐ 1																									
Mixer Clean	☐																									
Color Clean	☐																									
Cat Clean	☐																									
Please pay attention to whether there are people or objects																										
<table border="1"><thead><tr><th>Fluid Flow</th><th>Flow</th><th>Time</th></tr></thead><tbody><tr><td>Hub Flow</td><td>☐</td><td>0</td></tr><tr><td>Dyna. Brush</td><td>☐</td><td></td></tr><tr><td>Fluid Volume</td><td>☐ 0</td><td>0</td></tr><tr><td>Static Brush</td><td>☐ 0</td><td>0</td></tr></tbody></table>					Fluid Flow	Flow	Time	Hub Flow	☐	0	Dyna. Brush	☐		Fluid Volume	☐ 0	0	Static Brush	☐ 0	0							
Fluid Flow	Flow	Time																								
Hub Flow	☐	0																								
Dyna. Brush	☐																									
Fluid Volume	☐ 0	0																								
Static Brush	☐ 0	0																								
<table border="1"><thead><tr><th>Set Brush Factor</th><th>Perce nt</th><th>Set</th></tr></thead><tbody><tr><td>Fluid</td><td>☐ 0</td><td>0</td></tr><tr><td>Atom</td><td>☐ 0</td><td>0</td></tr><tr><td>Shape1</td><td>☐ 0</td><td>0</td></tr><tr><td>Shape 2</td><td>☐ 0</td><td>0</td></tr><tr><td>High Voltage</td><td>☐ 0</td><td>0</td></tr></tbody></table>					Set Brush Factor	Perce nt	Set	Fluid	☐ 0	0	Atom	☐ 0	0	Shape1	☐ 0	0	Shape 2	☐ 0	0	High Voltage	☐ 0	0				
Set Brush Factor	Perce nt	Set																								
Fluid	☐ 0	0																								
Atom	☐ 0	0																								
Shape1	☐ 0	0																								
Shape 2	☐ 0	0																								
High Voltage	☐ 0	0																								
Please pay attention to whether there are people or objects																										
<table border="1"><thead><tr><th>Calibrate IPS Curve</th><th>Value</th></tr></thead><tbody><tr><td>Channel</td><td>Shape1</td></tr></tbody></table>					Calibrate IPS Curve	Value	Channel	Shape1																		
Calibrate IPS Curve	Value																									
Channel	Shape1																									





6.3.3.1 Fill and Clean

This window carries out both the fill and clean operations of the paint flow.

Fill And Clean		Color
Material Fill	☐	1  
Air Fill	☐	
Solvent Fill	☐	
Bell Clean	☐	
Hub Clean	☐	
All Cavity Clean	☐	
Cat Fill	☐	1  
Mixer Clean	☐	
Color Clean	☐	
Cat Clean	☐	
Please pay attention to whether there are people or objects		



WARNING!

Before moving the robot position, check whether any people or objects are in the station.

Parameter

Function	Description	Command				
		Command No.	In Param 1	In Param 2	In Param 3	In Param 4
Material fill	Fill the selected colour material into pipe	401	Colour Number	0	0	0
Air fill	Fill the air into pipe	401	999	2	0	0
Solvent fill	Fill solvent into pipe	401	999	3	0	0
Bell clean	Clean bell	421	10022	0	0	0
Hub clean	Clean hub	421	10023	0	0	0
All cavity clean	Clean all pipe	401	999	0	0	0
Catalyst fill	Fill selected catalyst into pipe	401	Catalyst Number	0	3	0
Mixer clean	Clean only mixer pipe	401	999	0	1	0
Colour clean	Clean only colour pipe	401	999	0	2	0
Catalyst clean	Clean only catalyst pipe	401	999	0	3	0

Filling with air

Instruct the HMI to run the service program while ensuring that the robot is in auto mode and the station is in manual mode.

When utilizing the function to fill with air from the HMI, the application line can be left empty. However, if the line is not cleaned, it will initiate the cleaning process first. Subsequently, the system will actively blow air to remove all solvents from the hose.

Filling with solvent

Initiate the service program on the HMI while ensuring that the robot is in auto mode and the station is in manual mode.

This function is designed for a 2K robot. During prolonged production stops, the system can be filled with solvent to prevent curing. When executing a command from the HMI, the catalyst, resin, and mixer will be filled with solvent.

For more information about commands and relevant parameters, see the description in the chapter 7.3 ['Paint control command interface'](#) on page 258.

6.3.3.2 Flow test

Fluid flow test

Fluid Flow	Flow	Time
Hub Flow ☐		0
Dyna. Brush ☐		
Fluid Volume ☐	0	0
Static Brush ☐	0	0

Function	Command				
	Command No.	In Param1	In Param2	In Param3	In Param4
Hub flow test	421	10028	0	0	0
Dynamic brush test	421	10026	0	0	0
Fluid volume test	421	10025	0	0	0
Static brush test	421	10027	0	0	0

Overview

Condition:

Execute the process service program in RobotUI under the condition that the robot is in auto mode and the station is in manual mode.

During the fluid flow test, the robot will move to a programmed position and execute a timed brush in a predefined position. The material collected in a beaker will allow for comparison of the actual flow rate with the accumulated volume. If the volumetric accuracy is not within the technical specifications for the gear pump, corrections can be made. Volumetric measurement can be performed on every robot with the ability to measure the actual flow rate. Individual parameters must be adjusted from the visualization before the test.

The measured time can be configured from RobotUI.

Hub clean flow test

Condition:

Execute the process service program in RobotUI while ensuring that the robot is in auto mode and the station is in manual mode.

During the hub clean flow test, the robot will move to a programmed position and execute a timed brush in a predefined position. The material collected in a beaker will allow for comparison of the flow rate with the recommended settings for the actual atomizer. If the volumetric accuracy is not within the technical specification for the atomizer, manual adjustment on the solvent flow regulator is required.

Volumetric measurement can be performed on every robot with the ability to measure the actual flow rate. Individual parameters must be adjusted from the visualization before the test.

6.3.3.3 Brush test

Dynamic brush test

Condition:

With the robot in auto mode and the station in manual mode, execute the process service program in the HMI.

During the dynamic brush test, the robot moves along the created path and executes the brush event in the predefined positions. This test can be individually operated for every robot with reproducible spray times. Mode selection and individual parameters must be adjusted ahead of the test from the visualization.

Dynamic brush tests can be used for different purposes:

- Creating spray patterns.
- Checking for irregularities in pattern control.
- Implementing dynamic delay compensation.

Dynamic brush test overlap

Condition:

With the robot in auto mode and the station in manual mode, initiate the process service program in RobotUI.

During the dynamic brush test, the robot moves along the created path and executes the brush event in the predefined positions. The path overlaps. This test can be individually operated for every robot with reproducible spray times. Mode selection and individual parameters must be adjusted from the visualization before the test.

Dynamic brush tests serve various purposes:

- Creating spray patterns.
- Checking for irregularities in pattern control.
- Implementing dynamic delay compensation.

Static brush test

With the robot in auto mode and the station in manual mode, initiate the service program in RobotUI.

During the static brush test, the robot moves to a programmed position and executes a brush event in the predefined position. This test can be individually operated for every robot with reproducible spray times. Mode selection and individual parameters must be adjusted from the visualization before the test.

Static brush tests serve different purposes, including:

- Checking for irregularities in pattern control
- Identifying contamination in the air cap or shape air-ring.

6.3.3.4 Brush factors

Brush factor by percentage

To adjust Brush Offset, select the brush factor channel, input the desired percentage, and click the "Move" button to modify the brush factor percentage.

Set Brush Factor		Percent	Set
Fluid	☐	0	0
Atom	☐	0	0
Shape 1	☐	0	0
Shape 2	☐	0	0
High Voltage	☐	0	0

Function	Command					
	Command No.	In Param 1	In Param 2	In Param 3	In Param 4	In Param 5
Set fluid brush factor	320	0	0	1	FluidSetValue	0
Set atom brush factor	320	0	0	2	AtomSetValue	0
Set shape1 brush factor	320	0	0	3	Shape1SetValue	0
Set shape2 brush factor	320	0	0	4	Shape2SetValue	0
Set HV brush factor	320	0	0	5	HVSetValue	0

6.3.3.5 High voltage test

This window conducts the HV test under both static and dynamic conditions.

HV Test	
Dynamic Test	Start
Static Test	Start

Function	Command				
	Command No.	In Param1	In Param2	In Param3	In Param4
HV Dynamic Test	421	10029	0	0	0
HV Static Test	421	10030	0	0	0

High voltage
dynamic test

Dynamic test

Dynamic Test

☐ R11HVTest

☐ R21HVTest

☐ R12HVTest

☐ R22HVTest

Dynamic Test

Description

The HV dynamic test can only be initiated from the home position. This test measures the current value for three different positions to ensure the HV device is functioning properly and to determine if the HV cascade shuts off in the event of a short circuit. It is triggered from RobotUI.

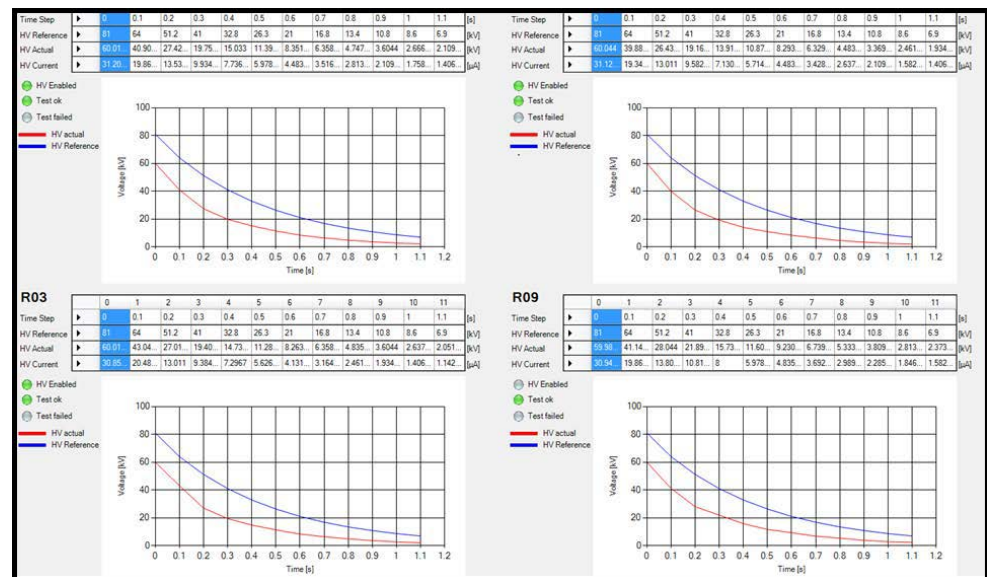
This test must be successfully completed every 24 hours to enable production. If the test fails or is not conducted for more than 24 hours, the station cannot be switched to automatic mode.

During this test, the robots move from the home position to the special HV test plate. High voltage is activated with a setpoint of 60 kV prior to reaching the plate.

The test is considered successful if the tested current values do not deviate significantly from the stored current value, and an HV error occurs upon successful completion. The test state is displayed in RobotUI.

High voltage static test

Static test



Description

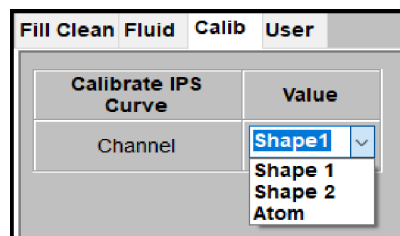
The HV static test can only be initiated from the home position and is necessary to determine the shut-down behavior of the HV cascade. It is triggered from RobotUI. It is recommended to run this test at least once a week.

The test begins with a voltage set point of 70 kV, turns off the voltage, and measures the voltage and current values every 0.1 seconds until 1.2 seconds have elapsed. The measured voltage curve is then compared to a reference curve. If the measurement at every step is below the corresponding reference value, the test is considered successful. All values and curves are displayed in RobotUI.

6.3.3.6 IPS Calibration

Parameter

In a closed-loop configuration for atom or shape air, calibrating the dynamic curve of the regulator is necessary.

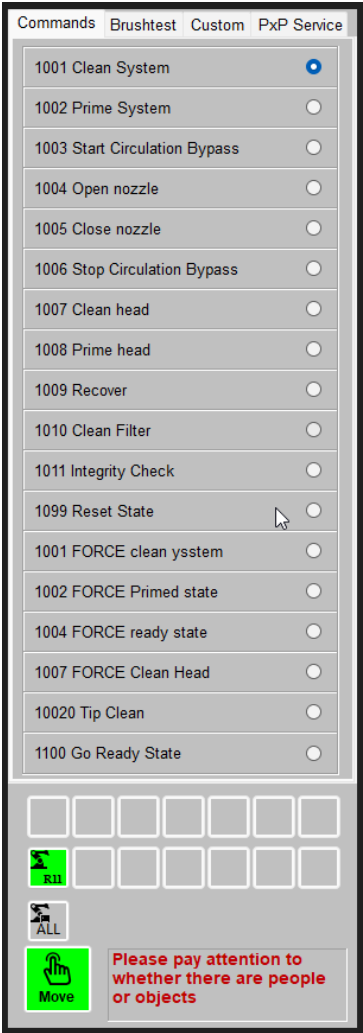


Function	Command				
	Command No.	In Param1	In Param2	In Param3	In Param4
IPS Calibration	421	10010	0	0	0

This calibration process can be performed in RobotUI.

6.3.4 Function - PixelPaint

Description This window include Commands, Brush Test, Custom and PxP Service command.



WARNING!

Before moving the robot position, check whether any people or objects are in the station.

6.3.4.1 Commands

Parameter

The Process application has a command on its command page.

Command	Description
Clean System	Clean process cavity and head
Prime System	Prime Process cavity and head
Stabilize Circulation Bypass	Stabilize circulation (without head nozzles)
Open nozzle	Opening valves to circulate paint through print head.
Close nozzle	Closing valves to circulate through bypass print head.
Stop Circulation Bypass	Stop paint circulation
Clean head	Clean print head
Prime head	Prime print head
Recover	Recover from a pump stop (release pressure and recover)
Clean Filter	Clean filter cavity
Integrity Check	Integrity Check sequence when required
Reset State	Reset PixelPaint state to unknown
FORCE clean system	FORCE clean system
FORCE Primed state	FORCE primed state
FORCE ready state	FORCE ready state
FORCE Clean Head	FORCE clean Head

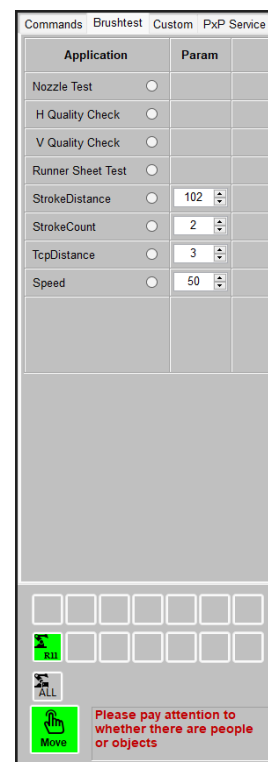
6.3.4.2 Brush test

This window displays the brush test.



WARNING!

Please pay attention to check whether any people or objects are in the station.



6.3.4.3 PxP Service

This window displays the PxP service.



WARNING!

Before doing the function, check whether there are any people or objects in the station.

Parameter

Commands for PxP Service:

Command	Description
Colour Select	Select colour number
Push Back Colour	Push back colour to supply system
Clean	Clean colour supply pipe
Prime	Prime colour supply pipe
Dry	Dry colour supply pipe
Fill With Air	Fill with air into the colour supply pipe
Fill With Air/Solvent	Fill with solvent into the colour supply pipe

6.3.4.4 Custom

Overview

A custom page is available for sending customized commands.

The format includes one command and ten parameters. Input all the parameters, then click the Move button to send them to the selected robot.

Parameter	Value
Command	421
Param1	0
Param2	0
Param3	0
Param4	0
Param5	0
Param6	0
Param7	0
Param8	0
Param9	0
Param10	0

Clear

Move

Please pay attention to whether there are people or objects



WARNING!

Before moving the robot position, check whether any people or objects are in the station.

6 HMI system

6.3 Robot

6.3.5 Function - Dispensing

6.3.5.1 Function - IDFP

Menu items Users can set clean times (frequency) and view actual clean times and clean counts.

Clean Times

Input	0
Actual	0
Count	0

Users can turn on/off temperature control and view temperature control status.

Temperature

Temp. Control On	☐
Temp. Control Ready	☐
Temp. Control Error	☐
Temp. Control	☒ NO

Users can click the “Start” button to release pressure.

Release Pressure

Release Pressure Request	Start
--------------------------	--------------------------------------

Users can turn on/off circulation mode, start/end circulation and view circulation status.

Circulation

Circle Status	☐
Circle Mode	☒ OFF
Circle Start	☒ OFF
Circle Flow	0.0
Circle Delay On	0.0
Circle Delay Off	0.0

Users can view the doser status.

Doser

Doser Ready ☐

Doser Error ☐

Users can view the material supply status.

Material Supply

Material Supply Ready ☐

Material Supply Error ☐

6.3.6 Backup

Overview

Users can do the robot backup manually and automatically.

The status will show a different description.

Nos.	Status	Description
1	Connection Lost	The robot is offline and cannot perform a backup.
2	Not at T1 mode	The robot is not in manual mode and cannot do backup.
3	Backup Running	The robot is doing a backup.
4	Backup Finish	Robot backup finished.
5	Backup Error	Some error occurs when doing a backup.

Robot: Backup

Robot	Status
R11	Backup Finish
R21	Backup Finish
R12	Backup Finish
D11	Backup Finish
D21	Backup Finish
D12	Backup Finish
D22	Backup Finish
H11	Backup Finish
H21	Backup Finish

R11

H11

R12

D11

D12

H21

R21

D21

D22

ALL

P

D

T

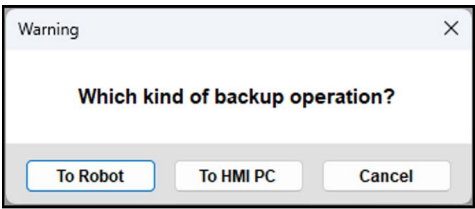
Backup



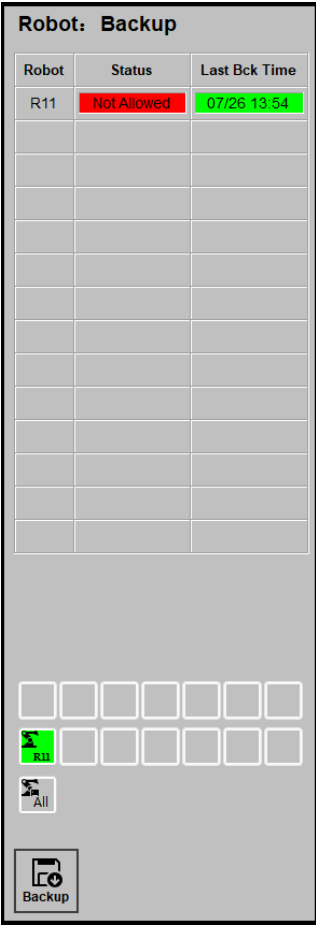
Important: The system will automatically do a backup for all robots and the backup time can be configured by the user. The user must not set the backup time during production because the robot load will increase when doing a backup, and the robot system will be influenced. Maybe it will cause a collision accident.

The backup function has been redesigned. Earlier, the robot's status wasn't updated promptly. Now, it refreshes in real time to display the current robot status accurately.

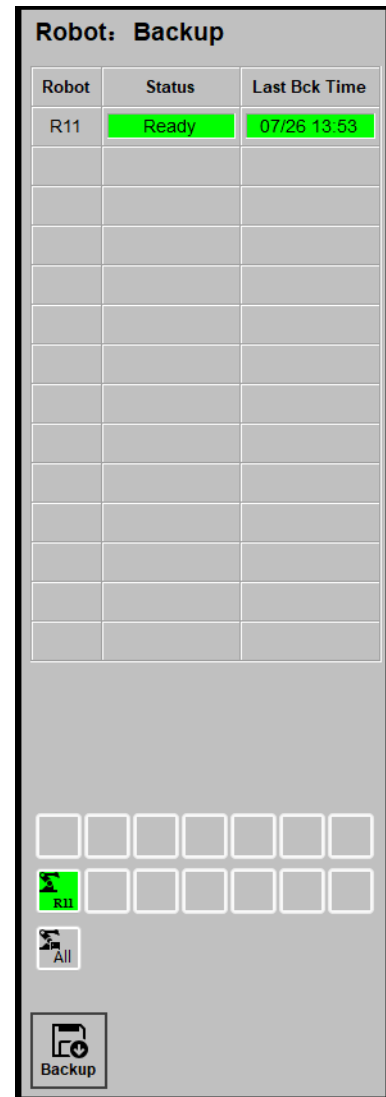
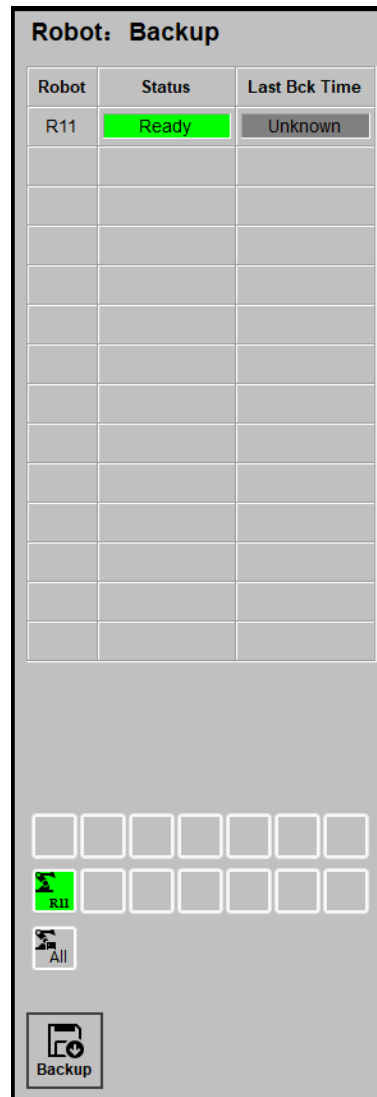
Upon clicking the Backup button, two options appear. The 'To Robot' option performs a backup solely within the Robot controller. The 'To HMI PC' option, after creating a backup in the robot controller, transfers it to the HMI PC, not retaining it in the robot controller.



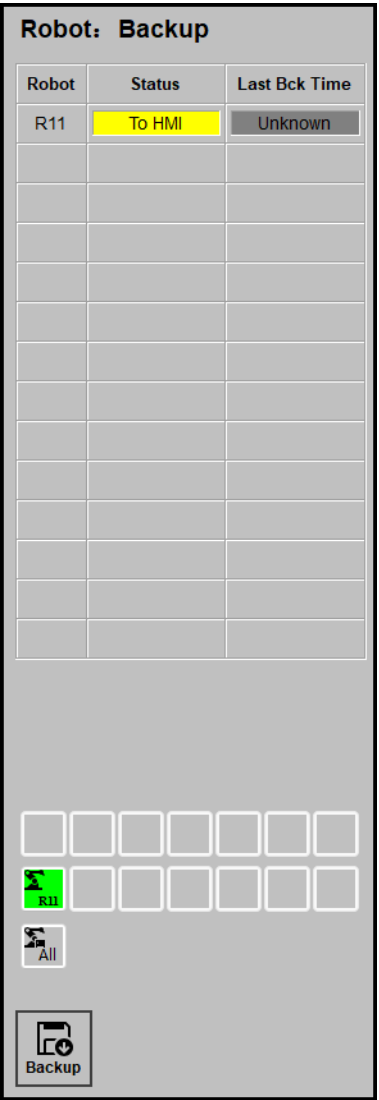
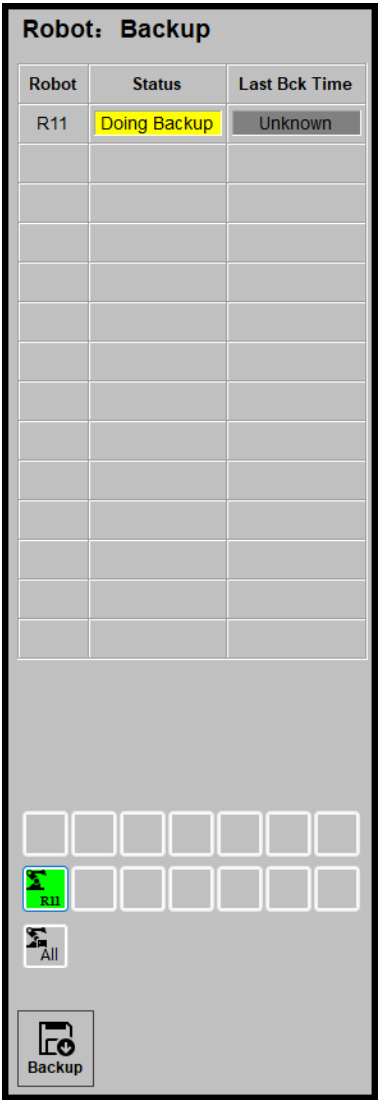
1. If the Robot is in auto mode or a cycle is running, backup execution is prohibited. This restriction is indicated by a red label displaying 'Not Allowed'.



2. When all conditions are satisfied, 'Ready' is displayed in green. After the RobotUI starts and before any backup is run, the last backup time displays 'Unknown'. Once a backup is completed, it shows the timestamp of the last successful backup.



3. When a backup is in progress, the robot's IO signals 'Backup Running' as 1, and the UI displays 'Doing Backup' in yellow. If the 'To HMI' option is selected for backup, once the robot backup completes, the UI changes to 'To HMI'. This indicates that files are being transferred from the Robot controller to the HMI PC.



6.3.7 Reset Program Pointer (PP)

Overview

Users can clear the robot's current task and reset the robot's main program point on this page.

The system input for the robot includes commands that configure the signals received from the user interface, directing them towards the main program (PP to main).

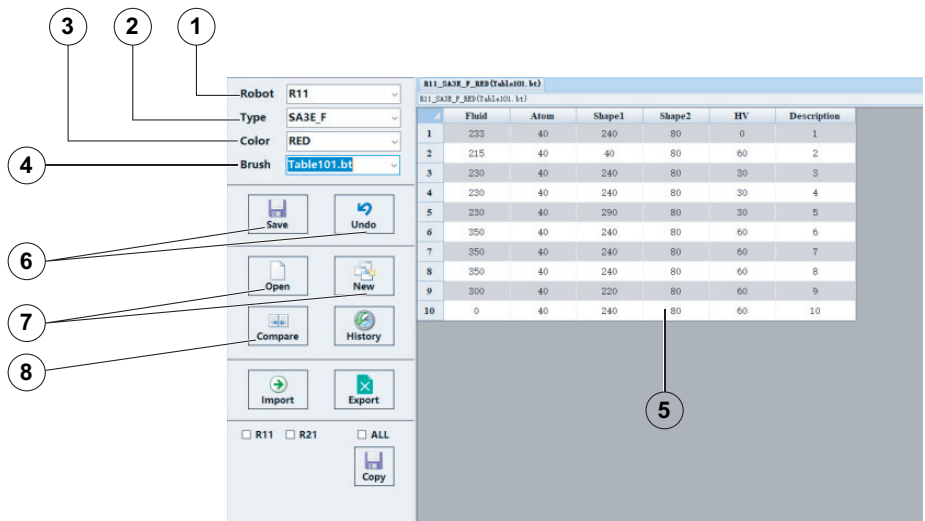
Skipping these commands will prompt the system to transition directly to the main program's first position.

This Command Run Controller cancels program execution (CMD 104).

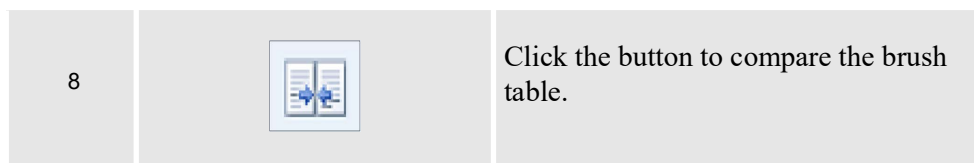
Robot	Reset PP
R11	Reset
R21	Reset
R12	Reset
R22	Reset
R13	Reset
R23	Reset
R14	Reset
R24	Reset
D11	Reset
D21	Reset
D12	Reset
D22	Reset
H11	Reset
H21	Reset
All	Reset

6.3.8 Brush Editor

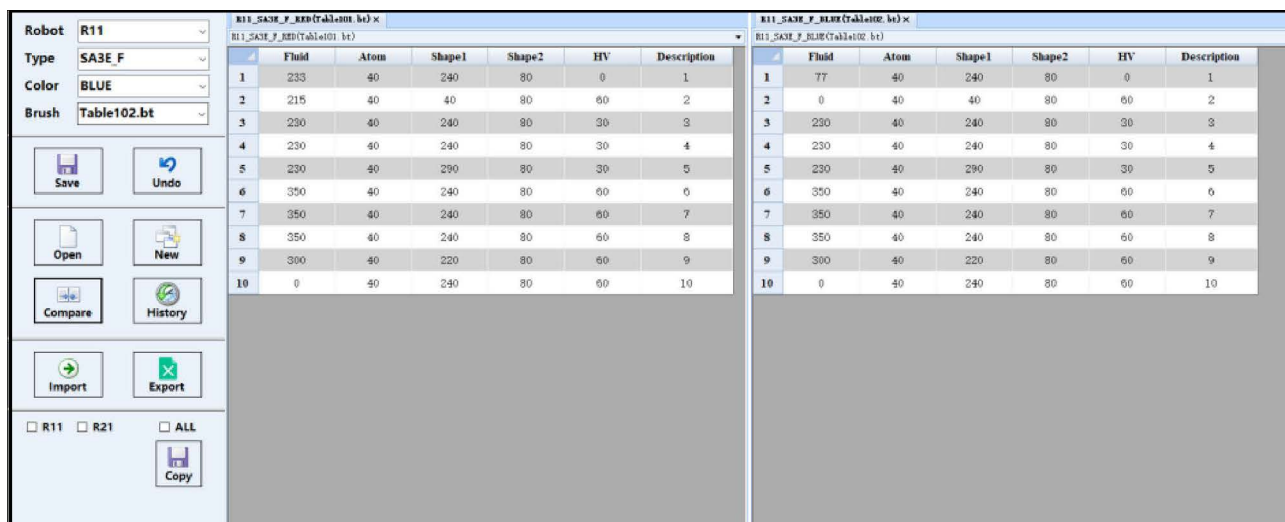
Description Users can edit the brush table with the brush table editor. Any changed data will be recorded in the operation log.



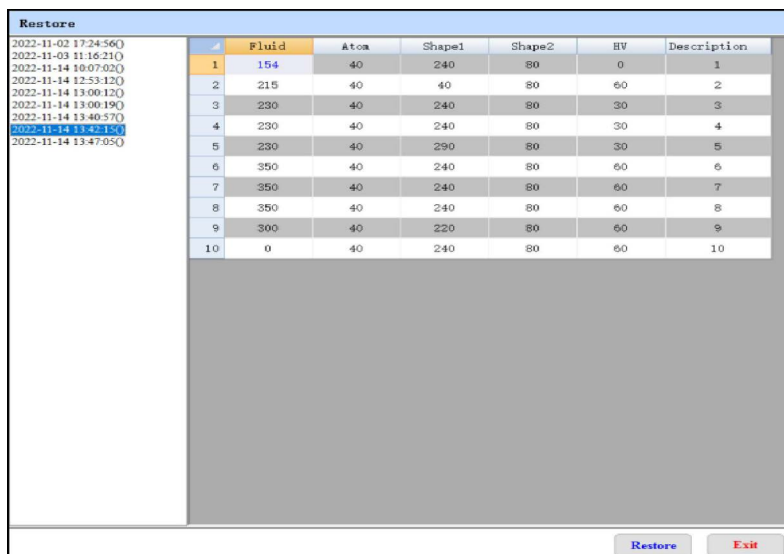
Pos.	Buttons	Description
1	-	Select the robot.
2	-	Select the type code.
3	-	Select the colour code.
4	-	Select the brush table.
5	-	Select on one cell and modify the data.
6	 or 	Click the button to save or click the button to cancel the modify.
7	 or 	Click the button to open the saved brush table or click the button to open the new brush table.



User can compare the two different brush table contents as shown in the figure below.



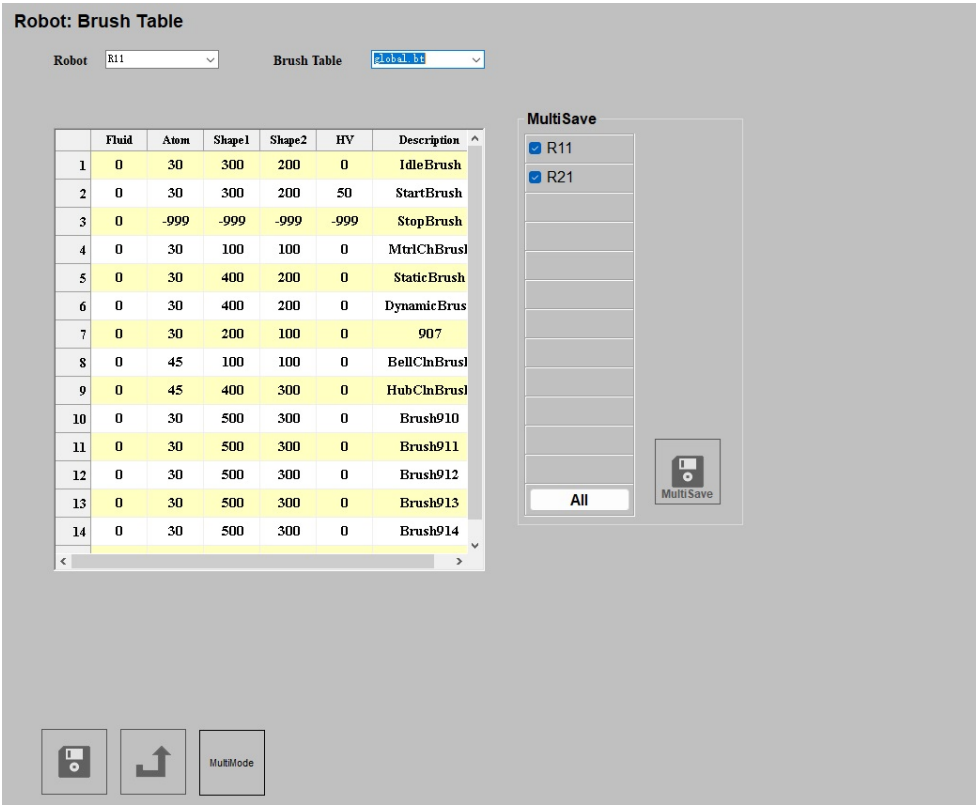
Click  to view the history brush table and restore the history brush table as shown in the figure below.



Click  button to import the brush table file, and click  button to export brush table file.

Click  button to copy the current brush table to selected robots.

The Brush Editor has been simplified and no longer includes complex features like history comparison. To modify the active brush file, adjust the Robot and Brush Table combo box. This process is similar to making changes in Excel. Once changes are made, save them back to the robot controller.



6.3.9 Refresh Log (command-IO)

Overview

The robot with the paint system receives a command IO signal from the PLC or RobotUI, and the robot system records a command log. However, the log is not refreshed in real time. The refresh log function facilitates the manual refresh of the robot log.

Operation conditions:

1. Communication with the robot is OK.
2. The robot is in automatic mode.
3. The robot master is released.

Robot	Refresh Log
R11	Refresh
R21	Refresh
R12	Refresh
R22	Refresh
R13	Refresh
R23	Refresh
R14	Refresh
R24	Refresh
D11	Refresh
D21	Refresh
D12	Refresh
D22	Refresh
H11	Refresh
H21	Refresh
All	Refresh

6.3.10 Shaftlock - Bell (paint robot)

Overview

The paint robot requires a lock air motor before removing and installing the bell cup.

Operation conditions:

- 1. Robot is in T1 mode.

Steps:

- 1. Select a robot and engage shaft lock.
- 2. Remove or install the bell cup.
- 3. Unlock the shaft after installing the cup.

Robot	Shaft Lock
R11	Not Locked
R21	Not Locked
R12	Not Locked
R22	Not Locked
R13	Not Locked
R23	Not Locked
R14	Not Locked
R24	Not Locked
All	All

6.3.11 Interlock

Overview

The paint robot displays the interlock status between robots.

Robot: Interlock									
discrete lock	R11	R12	H13	H14	discrete lock	R11	R12	H13	H14
1					1				
2					2				
3					3				
4					4				
5					5				
6					6				
7					7				
8					8				
9					9				
10					10				
11					11				
12					12				
13					13				
14					14				
15					15				
16					16				

6.4 Control

Following sub-sections are included in this section:

6.4.1 ['FIFO'](#) on page 211

6.4.1.1 ['Line tracking station'](#) on page 212

6.4.1.2 ['Stop&Go station'](#) on page 213

6.4.2 ['Simulation'](#) on page 214

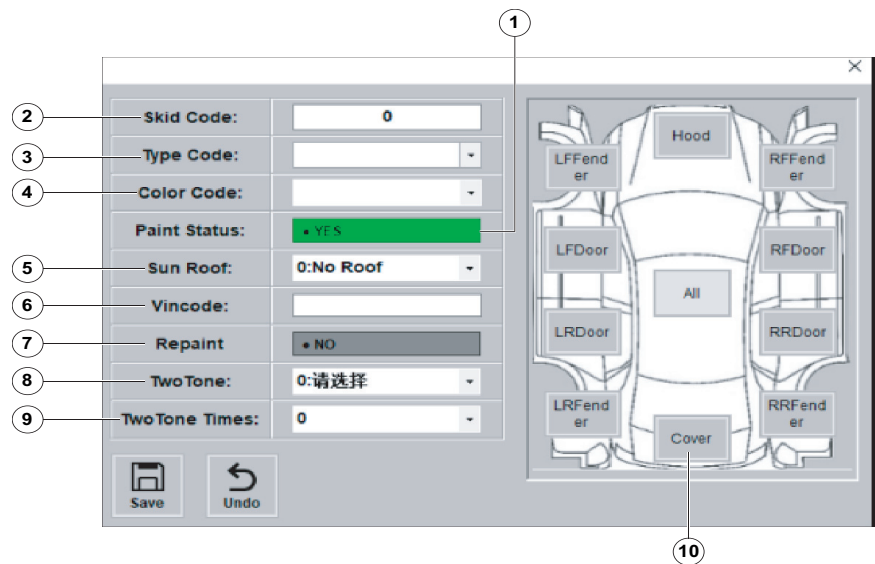
6.4.3 ['Bypass'](#) on page 216

6.4.4 ['Opener'](#) on page 217

6.4.1 FIFO

Overview

The FIFO displays car information as it enters the station.



Pos.	Description
1	Paint Enable Status
2	Skid Code
3	Type Code
4	Colour Code (paint or bumper system)
5	Sun Roof
6	Vincode
7	Repaint
8	TwoTone
9	TwoTone paint number of times
10	Paint part selection

Update button: Updates the car information.

Insert button: Inserts a car in front of the current car.

Delete button: Deletes the current car.

Steps to update car information:

1. Click the “Update” button.
2. Select or input the right car information.
3. Click  to save car information or click  to cancel the modification.

6.4.1.1 Line tracking station

Overview

The tracking station shows the current pitch and data check distance.

Control: FIFO Overview

01	Updat	Insert	Delet
	Skid	566	
	Type	111 H2L2 1	
	Col	SD 0	
COL ☐	VIN		

02 Updat Insert Delet

03 Updat Insert Delet

04 Updat Insert Delet

05 Updat Insert Delet

<< < 1 > >>

Stock Infomation

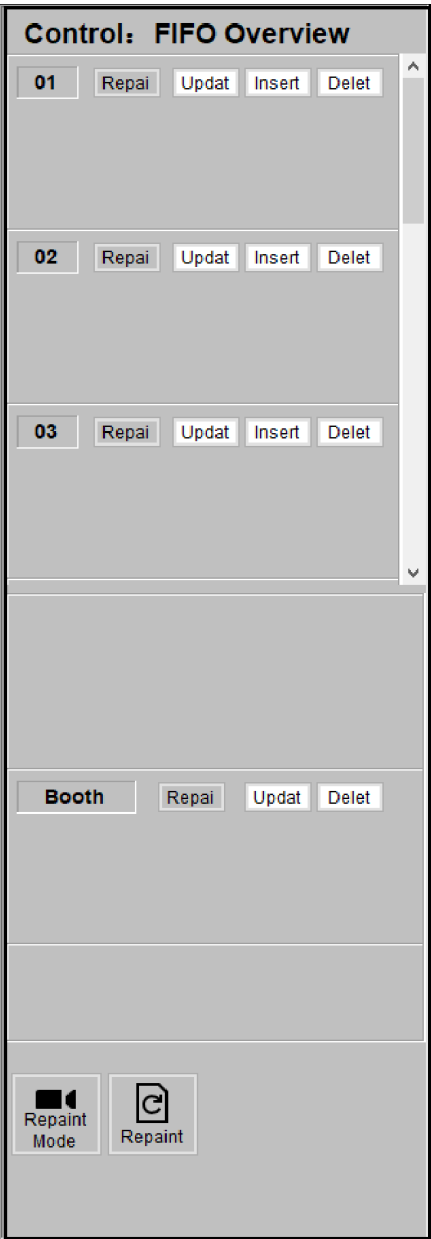
Current Pitch 13457

6.4.1.2 Stop&Go station

Overview

The Stop&Go station displays cars in the station.

It features a repaint mode button and a repaint start button. To initiate the repainting process, the user must first activate the repaint mode and then click the repaint start button.



6.4.2 Simulation

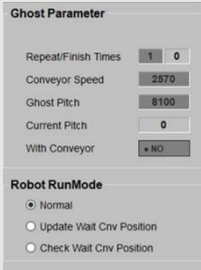
Users can do a simulation (ghost) before production.

Ghost information includes:

1. **Repeat/Finish Times:** Set all repeat times and current finished times.
2. **Conveyor Speed:** Establish the speed of the ghost conveyor at the tracking station only.
3. **Ghost Pitch:** Maintain a safe distance between two cars (Only tracking station).
4. **Current Pitch:** View the current distance from the last car (Only tracking station).
5. **With Conveyor:** Select whether ghost with conveyor running (Only tracking station).
6. **Robot RunMode:** The waiting point function is tested in a robot simulation.
 - Select normal mode.
 - Update mode will update the robot wait conveyor position when the ghost appears.
 - Check mode will check the robot wait conveyor position when the ghost appears. (Only tracking station).

Buttons

Ghost steps:

Steps	Buttons	Description
1		Click the button to turn on ghost mode.
2		Select ghost car information; the operation method is the same as FIFO.
3		Select the ghost parameter and robot parameter.
4	 or 	Click the button to save the ghost parameter or click the button to cancel or modify the ghost parameter.
5		Click the button to start the ghost.
6		Click the button to stop the ghost (not mandatory).
7		Click the button to turn off ghost mode.

6.4.3 Bypass

Users can bypass the robot, atomizer, HV, home position and other station devices on the screen.

Symbols

	Atomizer bypass: The robot will not paint.
	Home bypass: System runs without home position condition judgement.
	High voltage bypass: Robot paint without High voltage.
	Robot bypass: Do not use this robot.
	Gripper bypass: System runs without opener robot gripper sensor status condition judgement.

Users can bypass the safety device if needed with the permission of assigned authorities if the device is broken or the device status is not okay.

R11	B •	B •	• E	• E	
R21	B •	B •	• E	• E	
R12	B •	B •	• E	• E	
R22	B •	B •	• E	• E	
R13	B •	B •	B •	• E	
R23	B •	B •	• E	• E	
R14	B •	B •	• E	• E	
R24	B •	B •	• E	• E	
D11			• E	• E	• E
D21			• E	• E	B •
D12			• E	• E	B •
D22			• E	• E	B •
H11			• E	• E	B •
H21			• E	• E	B •
	All	All	All	All	All
Page1					
Fire Alarm		• ENABLE			
Booth		• ENABLE			
Safety Entry Left		• ENABLE			
Safety Entry Right		• ENABLE			
Safety Exit Left		• ENABLE			
Safety Exit Right		• ENABLE			
Entry LB		• ENABLE			
Exit LB		• ENABLE			
Conveyor Estop		BYPASS •			

6.4.4 Opener

Overview

Users can control the robot gripper movement on this screen. The gripper can be moved to full open, half open, closed, hook, unhook, ready, and home position.

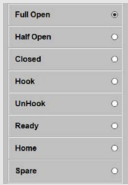
For the Stop&Go interior paint station, users can control the robot gripper movement on this screen. The gripper can be moved to full open, half open, closed, hook, unhook, ready, and home position.

Operation conditions:

1. Station manual mode
2. Robot automatic mode.

Buttons

Steps:

Pos.	Buttons	Description
1		Select robot.
2		Click to get the robot master control.
3		Select target position.
4		Click to move the robot position.
5		Click to release the robot master control.
6		Click to cycle on selected robot.



6.5 Setting

Following sub-sections are included in this section:

6.5.1 'Type' on page 219

6.5.2 'Colour (painting)' on page 220

6.5.3 'Skid slip sensor check (line tracking station)' on page 221

6.5.4 'Parameter' on page 222

6.5.5 'Alarm' on page 223

6.5.6 'Security' on page 225

6.5.7 'Miscellaneous' on page 227

Overview

The setting menu is designated for configuring station parameters, with most parameters saved in both the PLC and PC databases.

The following buttons are common across sub-menus.

	PLC Click to get data from PLC.
	PC Click to get data from the PC database.
	Save Click to save current page data. If the PLC button is selected, save the current page data to PLC. If the PC button is selected, save the current page data to the PC database.
	Restore Click to cancel or modify data.
	Synchronization Click to save all page data from PLC to the PC database or from the PC database to PLC. If the PLC button is selected, save all page data from the PLC to the PC database. If the PC button is selected, save all the page data from the PC database to PLC.

6 HMI system

6.5 Setting

6.5.1 Type

Description

The type setting interface offers a selection of hundred car types to configure.

The settings include:

No.	Header	Description
1	Enable	Enables painting if set to off.
2	Type Name	Description of the car type, user-defined.
3	Type Code	Car type code provided by the customer, where the code binds with the trajectory.
4	Trajectory	Number corresponding to the robot move trajectory.
5	Robot Selection	Specifies which robot requires spraying.
6	Picture	Image associated with the vehicle model.

Serial	Enable	Type Name	Type Code	Trajectory	Robot Selection	Picture
1	• Disable	Pegasus		0	R11 R12 R13 R14 R21 R22 R23 R24	
2	• Disable	Aries		0	R11 R12 R13 R14 R21 R22 R23 R24	
3	• Disable	Lya		0	R11 R12 R13 R14 R21 R22 R23 R24	
4	• Disable	Orion		0	R11 R12 R13 R14 R21 R22 R23 R24	
5	• Disable	LEO		0	R11 R12 R13 R14 R21 R22 R23 R24	
6	• Disable			0	R11 R12 R13 R14 R21 R22 R23 R24	
7	• Disable			0	R11 R12 R13 R14 R21 R22 R23 R24	
8	• Disable			0	R11 R12 R13 R14 R21 R22 R23 R24	
9	• Disable			0	R11 R12 R13 R14 R21 R22 R23 R24	
10	• Disable	Pegasus_5		0	R11 R12 R13 R14 R21 R22 R23 R24	
11	• Disable	Pegasus_G1.2		0	R11 R12 R13 R14 R21 R22 R23 R24	
12	• Disable	Aries_BR		0	R11 R12 R13 R14 R21 R22 R23 R24	
13	• Disable	Pegasus_Handle		0	R11 R12 R13 R14 R21 R22 R23 R24	
14	• Disable			0	R11 R12 R13 R14 R21 R22 R23 R24	
15	• Disable			0	R11 R12 R13 R14 R21 R22 R23 R24	

6.5.2 Colour (painting)

Overview

The colour setting interface allows configuration for a hundred colour types.

The settings comprise:

No.	Header	Description
1	Enable	Enables painting if set to off.
2	Colour Name	Description of the colour type, user-defined.
3	Colour Code	Colour code provided by the customer, binding with colour number and ring pipe.
4	Colour Number	Robot colour number, associated with the brush table.
5	Supplier	Name of the supplier, defined by the user.
6	Display	Selection of the corresponding colour for this colour pipe.
7	Robot Selection	Specifies which robot requires painting.

Serial	Color Enable	Color Name	Color Code	Color Number	Supplier	Display	Robot Selection			
1	☒	BLACK	C001	1	BASF		R11	R12	R13	R14
2	☒	WHITE	C002	2	PPG		R21	R22	R23	R24
3	☐			0			R11	R12	R13	R14
4	☐			0			R21	R22	R23	R24
5	☐			0			R11	R12	R13	R14
6	☐			0			R21	R22	R23	R24
7	☐			0			R11	R12	R13	R14
8	☐			0			R21	R22	R23	R24
9	☐			0			R11	R12	R13	R14
10	☐			0			R21	R22	R23	R24
11	☐			0			R11	R12	R13	R14
12	☐			0			R21	R22	R23	R24
13	☐			0			R11	R12	R13	R14
14	☐			0			R21	R22	R23	R24
15	☐			0			R11	R12	R13	R14

6.5.3 Skid slip sensor check (line tracking station)

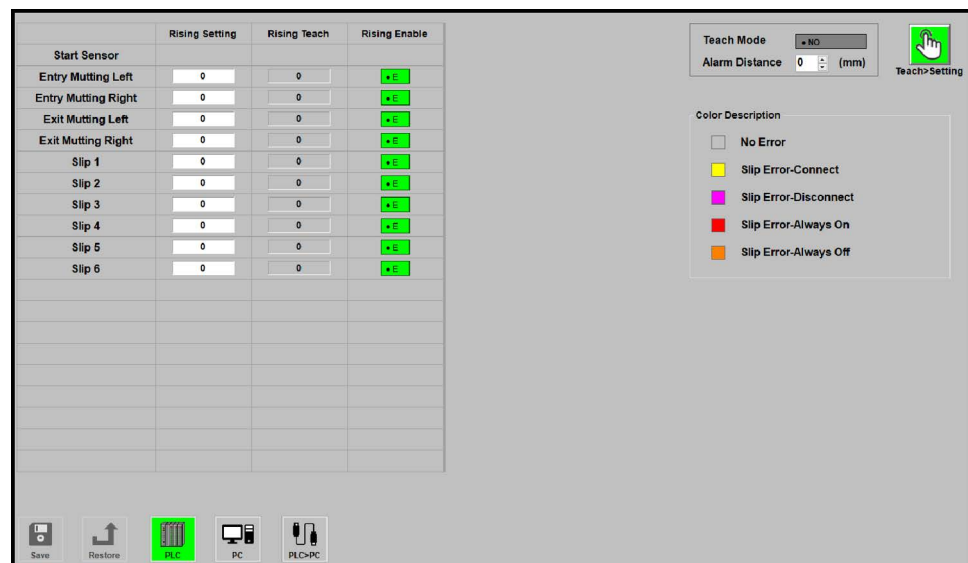
Overview

The slip sensor check verifies whether the skid has slipped. If the variance between the current slip sensor value and the preset value exceeds the safety range, the system will trigger an alarm.

Setting steps:

1. Turn on the teach mode.
2. Set the alarm distance.
3. As a car smoothly passes through the station, the rising teach value will be automatically generated.
4. Click  the button to raise the setting value by saving and teaching value.
5. Turn off the teach mode.

Users also have the option to manually adjust the rising setting values and disable certain slip sensors.



The screenshot shows the HMI interface for the Skid slip sensor check settings. It includes a table for sensor settings, a Teach Mode section, a Color Description legend, and a bottom toolbar.

Start Sensor	Rising Setting	Rising Teach	Rising Enable
Entry Mutting Left	0	0	
Entry Mutting Right	0	0	
Exit Mutting Left	0	0	
Exit Mutting Right	0	0	
Slip 1	0	0	
Slip 2	0	0	
Slip 3	0	0	
Slip 4	0	0	
Slip 5	0	0	
Slip 6	0	0	

Teach Mode: ☒ NO

Alarm Distance: 0 (mm)

Teach Mode Setting:

Color Description:

- ☐ No Error
- Slip Error-Connect
- Slip Error-Disconnect
- Slip Error-Always On
- Slip Error-Always Off

Bottom toolbar: Save, Restore, PLC, PC, PLC-PC

Overview

These parameters typically do not require modification once debugging is complete, as altering them can affect the system.

Serial	PLC Description	Value	Unit
61	Conveyor Speed Design	0	mm/min
62	Zone Length	10000	mm
63	Skid Length	0	mm
64	Minimum Pitch	0	mm
65	Air_Press_Set	0	

<
1
>

Serial	Robot Description	Value	Unit
83	R11 Zero Position	0	mm
84	R21 Zero Position	0	mm
85	R12 Zero Position	0	mm
86	R22 Zero Position	0	mm
88	R23 Zero Position	0	mm
95	Send Data Position	0	mm
96	Send Start Position	0	mm
99	Ready_Check_Position_R11	0	mm
100	Ready_Check_Position_R21	0	mm
101	Ready_Check_Position_R12	0	mm
102	Ready_Check_Position_R22	0	mm
104	Ready_Check_Position_R23	0	mm

<
1
>

Save

Undo

PLC

PC

PLC<PC

6.5.5 Alarm

Overview

The alarm settings consist of PLC alarm settings and robot alarm settings. The left side of the interface displays the PLC alarm settings, while the right side displays the robot alarm settings in the screen. Users can click on the robot name to switch between robots and click the “All” button to copy the current robot alarm settings to all other robots.

Alarm settings include:

No.	Header	Description
1	Fault	The alarm will be displayed in the alarm bar and will disappear when the alarm clears and is manually acknowledged (by clicking the reset button on the operation desk).
2	Stop	The alarm will cause the conveyor to stop.
3	Unack	Alarm display information will disappear without manual acknowledgment after the alarm clears in the alarm bar.

Users can click the “All” button at the top of the interface to set all current column alarms as Yes or No.



Clicking “Update” allows for editing alarm display information.

Description

Information includes:

No.	Header	Description
1	Alarm ID	Each alarm is assigned a unique alarm number.
2	English/Chinese Message	The message displayed in the alarm bar, available in both English and Chinese.
3	PLC Address	Alarm PLC signal address (robot alarm are also read from PLC).
4	Alarm Type	Different types of alarm/warning/information are displayed in different colours: alarms in red, warnings in yellow, and information in blue.
5	Alarm Class	The enforced alarms do not allow the user to set the alarm type.
6	Alarm Description	A detailed alarm description displayed when the user clicks the alarm in the alarm bar.

The screenshot shows a configuration window for an alarm. The fields are as follows:

- Alarm ID:** 1017
- English Message:** Not_Closed_SwDoor11
- Local Message:** 入口左侧摆门未关闭
- PLC Address:** DB931,X2.0
- Alarm Type:** Alarm
- Alarm Class:** Normal (selected), Enforced

A 'Save' button is located at the bottom of the dialog.

6.5.6 Security

Overview

Administrators can configure access for each screen and set the user level.

Each user account is associated with a specific level, ranging from level 0 to level 9.

The setting information includes:

No.	Header	Description
1	Title	Menu title name.
2	Menu enable	Disabling the menu will prevent it from being displayed.
3	Shortcut	Creating a shortcut on the left side of the status bar.
4	Level	The operational relationship between the user level and the interface. Authorized indicates authorized. Read Only indicates read only. Not Visible means not visible.



Clicking  the button will prompt a dialog to configure the user.

The settings include:

No.	Header	Description
1	Add	Enter the username, password, and user level, then click the “Add” button to create a new user.
2	Update	Select a username, make changes to the password or level as needed, then click the “Update” button to save the updated user information.
3	Delete	Select a username and click the “Delete” button to remove the user from the system.

A dialog box titled 'User Management' with a close button (X) in the top right corner. It contains a list box labeled 'User Name' with the following entries: administrator, abb, abc, saic, op, and new. The 'administrator' entry is selected. To the right of the list box are buttons for 'OK', 'Cancel', 'Add', 'Update', and 'Delete'. Below the list box are two text input fields: 'Password:' and 'Confirm:', both containing '*****'. At the bottom left is a 'Level:' label next to a numeric spinner set to '10'.

Automatic logout time setting:

The waiting time for automatic logout when user has no operation.

A dropdown menu for 'Level 9' is open, showing four options: 'Authorized' (selected), 'Not Visible', 'Read Only', and 'Authorized'.

The setting information includes:

No.	Header	Description
1	Level	The operational relationship between the user level and the interface. • Authorized indicates authorized. • Read Only indicates read only. • Not Visible means not visible.

The 'Setting: Security' screen displays menu settings. On the left, there are two lists: 'Main Menu' (Station, Robot, Control, Setting, Diagnose, Report, Program, Help) and 'Sub Menu' (Station, DailyReport). The 'Station' item in the 'Sub Menu' is selected. In the center, the 'Sub Menu Property Setting' section shows a 'Shortcut' set to 'YES' and a list of levels from 0 to 9, each with a dropdown menu. Level 0 is 'Not Visible', Level 1 is 'Read Only', and Levels 2 through 9 are 'Authorized'. On the right, the 'Automatic logout time setting' is set to '1750 (Seconds)'. At the bottom left are icons for 'Save', 'Undo', and 'User'.

6.5.7 Miscellaneous

Overview

Miscellaneous settings are displayed on this page, covering various configurations and options that do not fit into specific categories.

Setting: Miscellaneous

1 **Shift Configuration**

	Start Time	End Time
Shift 1	9:00:00 AM	6:00:00 PM
Shift 2	6:30:00 PM	6:00:00 AM
Shift 3	12:00:00 AM	12:00:00 AM

2 **Store Configuration**

Database Auto Backup ☒ YES

Backup Path ...

Manual Restore Manual Backup

3 Database Auto Delete ☒ YES

Keep Month Data

4 Robot Event Log

Backup Path

Keep Day Log

5 Robot Auto Backup ☒ YES

Robot Backup Path

Backup Time : :

Cut Time : :

Frequency

Controller Date Synchronization

Pos.	Components	Description
1	Shift	Configure shift start and end times, which will be logged in various reports such as production, consumption, operation, and alarm reports.
2	Database Auto Backup	Determine whether to automatically back up the database and specify the backup path. Manual backup and restoration options are also available.
3	Database Auto Delete	Choose whether to automatically delete database entries and set the duration for retaining data (e.g., one month). Manual deletion of database data is also possible.
4	Robot Event Log	Specify the backup path and retention period for the robot event logs obtained by RobotUI. Users can manually delete these logs.
5	Robot Auto Backup	Define whether the robot should perform automatic backups, set the backup path, timing, duration for retaining backups, and backup frequency (e.g., every few days). Users can synchronize the PC time with the robot.



WARNING! The system automatically backs up all robots, and the backup time can be configured by the user. However, it is crucial not to set the backup time during production. Doing so can increase the load on the robot system during backup, potentially leading to collisions or other accidents.

6.6 Diagnose

Following sub-sections are included in this section:

- 6.6.1 'Alarm' on page 229
- 6.6.2 'WAN (Factory Network)' on page 230
- 6.6.3 'Bus' on page 230
- 6.6.4 'Mixroom' on page 230
- 6.6.5 'IO' on page 231

6.6.1 Alarm

Overview

The alarm page shows the current alarms and the robot event log.

Alarm and error messages are displayed in red, warnings in yellow, and information in blue. Clicking an alarm message reveals a detailed description at the bottom of the screen.

Users can switch between robot event logs by clicking the robot button on the right. Filtering options include Info, Warning, Error, or viewing All logs. Navigation through log entries can be done using the "last" and "next" buttons or by selecting dates on the calendar.

Diagnose: Alarm					Robot Alarm			
Date	Operator	Shift	Device	Comment	Type	Code	Title	Date
2024-07-18 09:14:39	abb	1	PLC	1013 Station maintenance mode (DB001.X1.5)	Warning	40210	Interrupt removed from queue	2024-07-18 09:27:03
2024-07-18 09:14:01	abb	1	PLC	1073 Sensor fault - station start sensor (DB001.X5.0)	Information	133208	AtHV/24VotPower: Val lo	2024-07-18 09:14:40
2024-07-18 09:07:33	abb	1	R22	2011 Robot high voltage bypass (DB001.X81.1)	Information	10015	Manual mode selected	2024-07-18 09:14:39
2024-07-18 09:07:33	abb	1	R12	2011 Robot high voltage bypass (DB001.X81.1)	Error	90208	Chain switches open	2024-07-18 09:14:39
					Warning	20141	Motors Off rejected	2024-07-18 09:14:39
					Information	20577	HV On open	2024-07-18 09:14:38
					Error	20507	Hardware chain open	2024-07-18 09:14:38
					Information	10125	Program stopped	2024-07-18 09:14:38
					Warning	20141	Motors Off rejected	2024-07-18 09:14:38
					Information	10012	Safety guard stop state	2024-07-18 09:14:38
					Information	10151	Program started	2024-07-18 09:14:01
					Information	10053	Regain ready	2024-07-18 09:14:01
					Information	10052	Regain start	2024-07-18 09:14:01
					Information	10002	Program pointer has been reset	2024-07-18 09:14:01
					Warning	116033	Conveyor status change	2024-07-18 09:14:01
					Warning	40210	Interrupt removed from queue	2024-07-18 09:14:01
					Information	10138	Program stopped	2024-07-18 09:14:01
					Warning	116033	Conveyor status change	2024-07-18 09:07:43
					Information	10156	Program restarted	2024-07-18 09:07:33
					Information	10053	Regain ready	2024-07-18 09:07:33
					Information	10052	Regain start	2024-07-18 09:07:33
					Information	10011	Motors On state	2024-07-18 09:06:46
					Information	10010	Motors OFF state	2024-07-18 09:06:43
					Error	90208	Chain switches open	2024-07-18 09:06:42
					Information	10017	Automatic mode confirmed	2024-07-18 09:06:42
					Information	10016	Automatic mode requested	2024-07-18 09:06:42
					Error	20507	Hardware chain open	2024-07-18 09:06:36

6.6.2 WAN (Factory Network)

Overview

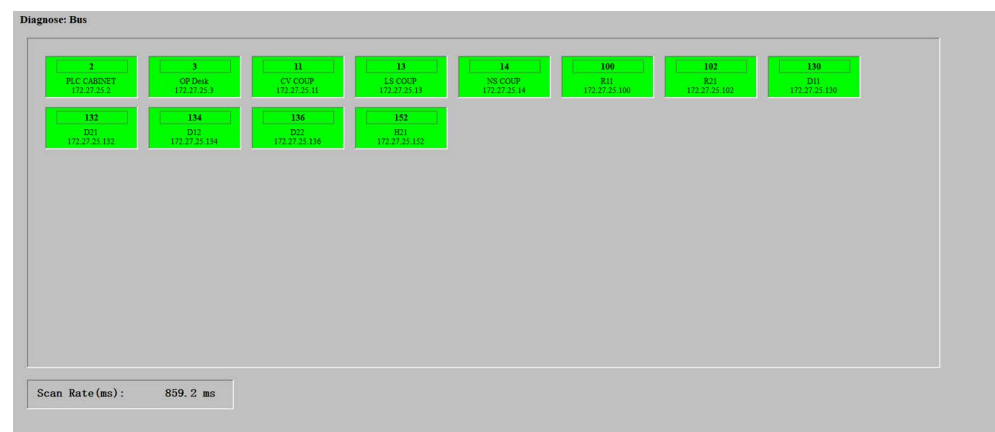
Indicate the network status as either connected or disconnected.



6.6.3 Bus

Overview

Display the status of ProfiNet, ProfiBus, and CC-Link on this screen.



6.6.4 Mixroom

Overview

Displays the status of the mix room, including the colour, solvent, and catalyst pipe statuses. Also, shows the pipe readiness and enable out statuses.

Status																								
Color	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	Solv	1	2	Cat	1	2
Ready	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	Ready	☐	☐	Ready	☐	☐
Enable Out	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	Enable Out	☐	☐	Enable Out	☐	☐

6 HMI system

6.6 Diagnose

6.6.5 IO

Display the status of both input and output signals, including boolean, integer, and string formats.

Diagnose: IO

ID	Input Comment	Status	ID	Output Comment	Status
1	FK_Gstop_Input_PL		1	Robot_Mode_Auto_Output_PL	
2	FK_Astop_Input_PL		2	Robot_Mode_T1_Output_PL	
3	Power_OK_R21_Input_PL		3	Robot_Mode_T2_Output_PL	
4	Robot_Mode_Auto_Input_PL		4	Motor_On_Output_PL	
5	Robot_Mode_T1_Input_PL		5	Motor_Off_Output_PL	
6	Robot_Mode_T2_Input_PL		6	Fault_Reset_Output_PL	
7	Station_Mode_Auto_Input_PL		7	G_Stop_Output_PL	
8	Station_Mode_Manual_Input_PL		8	A_Stop_Output_PL	
9	Cycle_Start_Input_PL		9	Start_On_Lamp_Output_PL	
10	Cycle_Stop_Input_PL		10	Motor_On_Lamp_Output_PL	
11	Robot_Motor_On_Input_PL		11	Fault_Reset_Lamp_Output_PL	
12	Robot_Motor_Off_Input_PL		12	Tower_Green_Lamp_Output_PL	
13	Lamp_Test_Input_PL		13	Tower_Yellow_Lamp_Output_PL	
14	Fault_Reset_Input_PL		14	Tower_Blue_Lamp_Output_PL	
15	Activate_Input_PL		15	Tower_Red_Lamp_Output_PL	
16	Buzzer_Stop_Input_PL		16	Tower_Buzzer_Lamp_Output_PL	
17	Clean_PB_Input_PL		17	Robot_Mode_Auto_Lamp_Output_PL	
18	Work_Mode_Input_PL		18	Robot_Mode_T1_Lamp_Output_PL	
19	Maintenance_Mode_Input_PL		19	Robot_Clean_Lamp_Output_PL	
20	Clean_Mode_Input_PL		20	Activate_Lamp_Output_PL	
21	Operation_Ready_On_Input_PL		21	Estop_Box_Off_Light_Lamp_Output_PL	
22	Operation_Ready_Off_Input_PL		22	Operation_Ready_On_Lamp_Output_PL	
23	Work_Position1_Input_PL				
24	Work_Position2_Input_PL				
25	Safe_Door1_1_Input_PL				
26	Safe_Door1_2_Input_PL				
27	Safe_DoorR1_1_Input_PL				
28	Safe_DoorR1_2_Input_PL				
29	Safe_DoorR2_1_Input_PL				
30	Safe_DoorR2_2_Input_PL				
31	In_Left_Estop_1_Input_PL				
32	In_Left_Estop_2_Input_PL				

PLC

Robot

6.7 Report

Following sub-sections are included in this section:

6.7.1 ['Production'](#) on page 233

6.7.2 ['Consumption'](#) on page 234

6.7.3 ['Operation'](#) on page 234

6.7.4 ['Alarm'](#) on page 235

6.7.5 ['Command log'](#) on page 236

Description

All report interfaces follow a similar structure, comprising a date filter, additional filtering options, time sorting functionality, a search button, an export button, a data display interface, and pagination controls.

The operation steps are:

1. Choose the desired time range using the date filter.
2. Apply specific filter conditions as needed.
3. Select the sorting order by time, either ascending or descending.
4. Click the "Search" button to retrieve the relevant data.
5. Navigate through the pages if the data spans multiple pages.
6. Export the data if required by clicking the "Export" button.

6.7.1 Production

Overview

The production report provides detailed information about each car body processed in the production line. It includes the following data:

Report: Production		Enter Time	Exit Time	Skid	Type	Color	Carvin	Duration	Paint Off	Dailycount	Shift	Shiftcount	Roof
Date Filter Start Time2019-01-15 00:00:00End Time2023-09-15 10:06:58 Last DayTodayNext Day Other Filter TypeALLColorALLShiftALLSkid0VincodePaint OffSimulationRoofALLTwo ToneALLProcessALL Order By Time AscDesc SearchExport Total 370 pages	1	1/11/2023 2:26:23 PM	1/11/2023 2:23:46 PM	2234	10	AB20	232018125	00:01:31	Paint	125	1	125	0
	2	1/11/2023 2:22:14 PM	1/11/2023 2:22:04 PM	2403	10	AB20	232017954	00:01:31	Paint	124	1	124	0
	3	1/11/2023 2:20:32 PM	1/11/2023 2:22:04 PM	2257	10	AB20	232018137	00:01:31	Paint	123	1	123	0
	4	1/11/2023 2:18:51 PM	1/11/2023 2:20:23 PM	2067	10	AB20	232018025	00:01:31	Paint	122	1	122	0
	5	1/11/2023 2:14:00 PM	1/11/2023 2:15:32 PM	2348	10	AB20	232018131	00:01:31	Paint	121	1	121	0
	6	1/11/2023 2:12:18 PM	1/11/2023 2:13:51 PM	2203	10	AB20	232018065	00:01:32	Paint	120	1	120	0
	7	1/11/2023 2:10:21 PM	1/11/2023 2:11:53 PM	2226	10	AB10	232018133	00:01:31	Paint	119	1	119	0
	8	1/11/2023 2:08:31 PM	1/11/2023 2:10:03 PM	2266	10	AB10	232018047	00:01:32	Paint	118	1	118	0
	9	1/11/2023 2:06:49 PM	1/11/2023 2:08:21 PM	2008	10	AB10	232018120	00:01:31	Paint	117	1	117	0
	10	1/11/2023 2:05:07 PM	1/11/2023 2:06:39 PM	2186	10	AB10	232018114	00:01:32	Paint	116	1	116	0
	11	1/11/2023 2:03:27 PM	1/11/2023 2:04:59 PM	2095	10	AB10	232018093	00:01:31	Paint	115	1	115	0
	12	1/11/2023 2:00:55 PM	1/11/2023 2:02:27 PM	2379	10	AB10	232018100	00:01:31	Paint	114	1	114	0
	13	1/11/2023 1:59:13 PM	1/11/2023 2:00:44 PM	2075	10	AB10	232018106	00:01:31	Paint	113	1	113	0
	14	1/11/2023 1:56:12 PM	1/11/2023 1:57:45 PM	2108	10	AB02	232018111	00:01:33	Paint	112	1	112	0
	15	1/11/2023 1:54:30 PM	1/11/2023 1:56:03 PM	2209	10	AB02	232017538	00:01:33	Paint	111	1	111	0
	16	1/11/2023 1:52:49 PM	1/11/2023 1:54:20 PM	2291	10	AB02	232017403	00:01:31	Paint	110	1	110	0
	17	1/11/2023 1:51:06 PM	1/11/2023 1:52:38 PM	2259	10	AB06	232018091	00:01:31	Paint	109	1	109	0
	18	1/11/2023 1:49:24 PM	1/11/2023 1:50:56 PM	2376	10	AB06	232017909	00:01:31	Paint	108	1	108	0
	19	1/11/2023 1:47:43 PM	1/11/2023 1:49:14 PM	2180	10	AB06	232018099	00:01:31	Paint	107	1	107	0
	20	1/11/2023 1:42:34 PM	1/11/2023 1:44:06 PM	2151	10	AB06	232018095	00:01:31	Paint	106	1	106	0
	21	1/11/2023 1:40:52 PM	1/11/2023 1:42:24 PM	2164	10	AB06	232018104	00:01:31	Paint	105	1	105	0
	22	1/11/2023 1:39:09 PM	1/11/2023 1:40:43 PM	2342	10	AB06	232018094	00:01:33	Paint	104	1	104	0
	23	1/11/2023 1:29:37 PM	1/11/2023 1:31:09 PM	2160	10	AB06	232018079	00:01:31	Paint	103	1	103	0
	24	1/11/2023 1:27:54 PM	1/11/2023 1:29:28 PM	2123	10	AB06	232018086	00:01:33	Paint	102	1	102	0
	25	1/11/2023 1:25:57 PM	1/11/2023 1:27:30 PM	2222	10	AB04	232018066	00:01:33	Paint	101	1	101	0
	26	1/11/2023 1:24:01 PM	1/11/2023 1:25:32 PM	2341	10	AB02	232018096	00:01:31	Paint	100	1	100	0
	27	1/11/2023 1:22:18 PM	1/11/2023 1:23:51 PM	2405	10	AB02	232018074	00:01:33	Paint	99	1	99	0
	28	1/11/2023 1:20:22 PM	1/11/2023 1:21:54 PM	2360	10	AB04	232018029	00:01:32	Paint	98	1	98	0
	29	1/11/2023 1:11:04 PM	1/11/2023 1:12:36 PM	2216	10	AB04	232018035	00:01:32	Paint	97	1	97	0
	30	1/11/2023 1:09:18 PM	1/11/2023 1:11:02 PM	2093	10	AB04	232018028	00:01:44	Paint	96	1	96	0

Header	Description
Skid Number	Unique identifier for the skid on which the car body is mounted.
VIN Code	Vehicle Identification Number, a unique code assigned to each vehicle.
Type Code	Code indicating the type of vehicle.
Colour Code	Code representing the colour of the vehicle.
Paint Duration	Duration of the painting process for the car body.
Paint Status	Status of the painting process (e.g., completed, ongoing, or pending).
Daily Count	Total count of car bodies processed during the day.
Shift Status	Status of the shift during which the car body was processed (e.g., morning, afternoon, or night shift).

6.7.2 Consumption

Overview

The consumption report provides detailed information about the consumption of materials by each robot for each car body processed in the production line.

Report: Consumption		Enter Time	Skid	Type	Color	Guest Car	Consumption	Paint Off	Dailycount	Shift	Shiftcount	R11	R12	R13	R14	R15	R16	R17	R18	R19	R20	R21	R22	R23	R24	R25	R26	R27	R28	R29	R30	R31	R32	R33	R34	R35	R36	R37	R38	R39	R40	R41	R42	R43	R44	R45	R46	R47	R48	R49	R50	R51	R52	R53	R54	R55	R56	R57	R58	R59	R60	R61	R62	R63	R64	R65	R66	R67	R68	R69	R70	R71	R72	R73	R74	R75	R76	R77	R78	R79	R80	R81	R82	R83	R84	R85	R86	R87	R88	R89	R90	R91	R92	R93	R94	R95	R96	R97	R98	R99	R100	R101	R102	R103	R104	R105	R106	R107	R108	R109	R110	R111	R112	R113	R114	R115	R116	R117	R118	R119	R120	R121	R122	R123	R124	R125	R126	R127	R128	R129	R130	R131	R132	R133	R134	R135	R136	R137	R138	R139	R140	R141	R142	R143	R144	R145	R146	R147	R148	R149	R150	R151	R152	R153	R154	R155	R156	R157	R158	R159	R160	R161	R162	R163	R164	R165	R166	R167	R168	R169	R170	R171	R172	R173	R174	R175	R176	R177	R178	R179	R180	R181	R182	R183	R184	R185	R186	R187	R188	R189	R190	R191	R192	R193	R194	R195	R196	R197	R198	R199	R200	R201	R202	R203	R204	R205	R206	R207	R208	R209	R210	R211	R212	R213	R214	R215	R216	R217	R218	R219	R220	R221	R222	R223	R224	R225	R226	R227	R228	R229	R230	R231	R232	R233	R234	R235	R236	R237	R238	R239	R240	R241	R242	R243	R244	R245	R246	R247	R248	R249	R250	R251	R252	R253	R254	R255	R256	R257	R258	R259	R260	R261	R262	R263	R264	R265	R266	R267	R268	R269	R270	R271	R272	R273	R274	R275	R276	R277	R278	R279	R280	R281	R282	R283	R284	R285	R286	R287	R288	R289	R290	R291	R292	R293	R294	R295	R296	R297	R298	R299	R300	R301	R302	R303	R304	R305	R306	R307	R308	R309	R310	R311	R312	R313	R314	R315	R316	R317	R318	R319	R320	R321	R322	R323	R324	R325	R326	R327	R328	R329	R330	R331	R332	R333	R334	R335	R336	R337	R338	R339	R340	R341	R342	R343	R344	R345	R346	R347	R348	R349	R350	R351	R352	R353	R354	R355	R356	R357	R358	R359	R360	R361	R362	R363	R364	R365	R366	R367	R368	R369	R370	R371	R372	R373	R374	R375	R376	R377	R378	R379	R380	R381	R382	R383	R384	R385	R386	R387	R388	R389	R390	R391	R392	R393	R394	R395	R396	R397	R398	R399	R400	R401	R402	R403	R404	R405	R406	R407	R408	R409	R410	R411	R412	R413	R414	R415	R416	R417	R418	R419	R420	R421	R422	R423	R424	R425	R426	R427	R428	R429	R430	R431	R432	R433	R434	R435	R436	R437	R438	R439	R440	R441	R442	R443	R444	R445	R446	R447	R448	R449	R450	R451	R452	R453	R454	R455	R456	R457	R458	R459	R460	R461	R462	R463	R464	R465	R466	R467	R468	R469	R470	R471	R472	R473	R474	R475	R476	R477	R478	R479	R480	R481	R482	R483	R484	R485	R486	R487	R488	R489	R490	R491	R492	R493	R494	R495	R496	R497	R498	R499	R500	R501	R502	R503	R504	R505	R506	R507	R508	R509	R510	R511	R512	R513	R514	R515	R516	R517	R518	R519	R520	R521	R522	R523	R524	R525	R526	R527	R528	R529	R530	R531	R532	R533	R534	R535	R536	R537	R538	R539	R540	R541	R542	R543	R544	R545	R546	R547	R548	R549	R550	R551	R552	R553	R554	R555	R556	R557	R558	R559	R560	R561	R562	R563	R564	R565	R566	R567	R568	R569	R570	R571	R572	R573	R574	R575	R576	R577	R578	R579	R580	R581	R582	R583	R584	R585	R586	R587	R588	R589	R590	R591	R592	R593	R594	R595	R596	R597	R598	R599	R600	R601	R602	R603	R604	R605	R606	R607	R608	R609	R610	R611	R612	R613	R614	R615	R616	R617	R618	R619	R620	R621	R622	R623	R624	R625	R626	R627	R628	R629	R630	R631	R632	R633	R634	R635	R636	R637	R638	R639	R640	R641	R642	R643	R644	R645	R646	R647	R648	R649	R650	R651	R652	R653	R654	R655	R656	R657	R658	R659	R660	R661	R662	R663	R664	R665	R666	R667	R668	R669	R670	R671	R672	R673	R674	R675	R676	R677	R678	R679	R680	R681	R682	R683	R684	R685	R686	R687	R688	R689	R690	R691	R692	R693	R694	R695	R696	R697	R698	R699	R700	R701	R702	R703	R704	R705	R706	R707	R708	R709	R710	R711	R712	R713	R714	R715	R716	R717	R718	R719	R720	R721	R722	R723	R724	R725	R726	R727	R728	R729	R730	R731	R732	R733	R734	R735	R736	R737	R738	R739	R740	R741	R742	R743	R744	R745	R746	R747	R748	R749	R750	R751	R752	R753	R754	R755	R756	R757	R758	R759	R760	R761	R762	R763	R764	R765	R766	R767	R768	R769	R770	R771	R772	R773	R774	R775	R776	R777	R778	R779	R780	R781	R782	R783	R784	R785	R786	R787	R788	R789	R790	R791	R792	R793	R794	R795	R796	R797	R798	R799	R800	R801	R802	R803	R804	R805	R806	R807	R808	R809	R810	R811	R812	R813	R814	R815	R816	R817	R818	R819	R820	R821	R822	R823	R824	R825	R826	R827	R828	R829	R830	R831	R832	R833	R834	R835	R836	R837	R838	R839	R840	R841	R842	R843	R844	R845	R846	R847	R848	R849	R850	R851	R852	R853	R854	R855	R856	R857	R858	R859	R860	R861	R862	R863	R864	R865	R866	R867	R868	R869	R870	R871	R872	R873	R874	R875	R876	R877	R878	R879	R880	R881	R882	R883	R884	R885	R886	R887	R888	R889	R890	R891	R892	R893	R894	R895	R896	R897	R898	R899	R900	R901	R902	R903	R904	R905	R906	R907	R908	R909	R910	R911	R912	R913	R914	R915	R916	R917	R918	R919	R920	R921	R922	R923	R924	R925	R926	R927	R928	R929	R930	R931	R932	R933	R934	R935	R936	R937	R938	R939	R940	R941	R942	R943	R944	R945	R946	R947	R948	R949	R950	R951	R952	R953	R954	R955	R956	R957	R958	R959	R960	R961	R962	R963	R964	R965	R966	R967	R968	R969	R970	R971	R972	R973	R974	R975	R976	R977	R978	R979	R980	R981	R982	R983	R984	R985	R986	R987	R988	R989	R990	R991	R992	R993	R994	R995	R996	R997	R998	R999	R1000	R1001	R1002	R1003	R1004	R1005	R1006	R1007	R1008	R1009	R1010	R1011	R1012	R1013	R1014	R1015	R1016	R1017	R1018	R1019	R1020	R1021	R1022	R1023	R1024	R1025	R1026	R1027	R1028	R1029	R1030	R1031	R1032	R1033	R1034	R1035	R1036	R1037	R1038	R1039	R1040	R1041	R1042	R1043	R1044	R1045	R1046	R1047	R1048	R1049	R1050	R1051	R1052	R1053	R1054	R1055	R1056	R1057	R1058	R1059	R1060	R1061	R1062	R1063	R1064	R1065	R1066	R1067	R1068	R1069	R1070	R1071	R1072	R1073	R1074	R1075	R1076	R1077	R1078	R1079	R1080	R1081	R1082	R1083	R1084	R1085	R1086	R1087	R1088	R1089	R1090	R1091	R1092	R1093	R1094	R1095	R1096	R1097	R1098	R1099	R1100	R1101	R1102	R1103	R1104	R1105	R1106	R1107	R1108	R1109	R1110	R1111	R1112	R1113	R1114	R1115	R1116	R1117	R1118	R1119	R1120	R1121	R1122	R1123	R1124	R1125	R1126	R1127	R1128	R1129	R1130	R1131	R1132	R1133	R1134	R1135	R1136	R1137	R1138	R1139	R1140	R1141	R1142	R1143	R1144	R1145	R1146	R1147	R1148	R1149	R1150	R1151	R1152	R1153	R1154	R1155	R1156	R1157	R1158	R1159	R1160	R1161	R1162	R1163	R1164	R1165	R1166	R1167	R1168	R1169	R1170	R1171	R1172	R1173	R1174	R1175	R1176	R1177	R1178	R1179	R1180	R1181	R1182	R1183	R1184	R1185	R1186	R1187	R1188	R1189	R1190	R1191	R1192	R1193	R1194	R1195	R1196	R1197	R1198	R1199	R1200	R1201	R1202	R1203	R1204	R1205	R1206	R1207	R1208	R1209	R1210	R1211	R1212	R1213	R1214	R1215	R1216	R1217	R1218	R1219	R1220	R1221	R1222	R1223	R1224	R1225	R1226	R1227	R1228	R1229	R1230	R1231	R1232	R1233	R1234	R1235	R1236	R1237	R1238	R1239	R1240	R1241	R1242	R1243	R1244	R1245	R1246	R1247	R1248	R1249	R1250	R1251	R1252	R1253	R1254	R1255	R1256	R1257	R1258	R1259	R1260	R1261	R1262	R1263	R1264	R1265	R1266	R1267	R1268	R1269	R1270	R1271	R1272	R1273	R1274	R1275	R1276	R1277	R1278	R1279	R1280	R1281	R1282	R1283	R1284	R1285	R1286	R1287	R1288	R1289	R1290	R1291	R1292	R1293	R1294	R1295	R1296	R1297	R1298	R1299	R1300	R1301	R1302	R1303	R1304	R1305	R1306	R1307	R1308	R1309	R1310	R1311	R1312	R1313	R1314	R1315	R1316	R1317	R1318	R1319	R1320	R1321	R1322	R1323	R1324	R1325	R1326	R1327	R1328	R1329	R1330	R1331	R1332	R1333	R1334	R1335	R1336	R1337	R1338	R1339	R1340	R1341	R1342	R1343	R1344	R1345	R1346	R1347	R1348	R1349	R1350	R1351	R1352	R1353	R1354	R1355	R1356	R1357	R1358	R1359	R1360	R1361	R1362	R1363	R1364	R1365	R1366	R1367	R1368	R1369	R1370	R1371	R1372	R1373	R1374	R1375	R1376	R1377	R1378	R1379	R1380	R1381	R1382	R1383	R1384	R1385	R1386	R1387	R1388	R1389	R1390	R1391	R1392	R1393	R1394	R1395	R1396	R1397	R1398	R1399	R1400	R1401	R1402	R1403	R1404	R1405	R1406	R1407	R1408	R1409	R1410	R1411	R1412	R1413	R1414	R1415	R1416	R1417	R1418	R1419	R1420	R1421	R1422	R1423	R1424	R1425	R1426	R1427	R1428	R1429	R1430	R1431	R1432	R1433	R1434	R1435	R1436	R1437	R1438	R1439	R1440	R1441	R1442	R1443	R1444	R1445	R1446	R1447	R1448	R1449	R1450	R1451	R1452	R1453	R1454	R1455	R1456	R1457	R1458	R1459	R1460	R1461	R1462	R1463	R1464	R1465	R1466	R1467	R1468	R1469	R1470
---------------------	--	------------	------	------	-------	-----------	-------------	-----------	------------	-------	------------	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------

6.7.4 Alarm

Overview

The alarm report provides a detailed history of alarm information, including the data for each alarm event.

Report: Alarm

Option

☒ Normal

☐ Top Frequency

Date Filter

Start Time

2023-10-01 00:00:00

End Time

2024-02-19 14:37:17

Last Day

Today

Next Day

Other Filter

Device

ALL

Shift

ALL

Message

Order By Time

☐ Asc

☒ Desc

	Time	Duration	Device	Code	Operator	Shift	Message(English)	Address
211	2023-11-22 15:07:02	00:05:44	D21	2881	abb	1	HeartBeat_Fault	DB931.X360.0
212	2023-11-22 15:07:02	00:05:45	R12	482	abb	1	Emergency_Stop	DB931.X60.1
213	2023-11-22 15:07:01	00:05:46	R11	162	abb	1	Emergency_Stop	DB931.X20.1
214	2023-11-22 15:07:01	00:05:45	D12	3041	abb	1	HeartBeat_Fault	DB931.X380.0
215	2023-11-22 15:07:00	00:05:46	D11	2721	abb	1	HeartBeat_Fault	DB931.X340.0
216	2023-11-22 15:07:00	00:05:47	H21	3522	abb	1	Emergency_Stop	DB931.X440.1
217	2023-11-22 15:06:58	00:05:49	D22	3202	abb	1	Emergency_Stop	DB931.X400.1
218	2023-11-22 15:06:57	00:05:50	D21	2882	abb	1	Emergency_Stop	DB931.X360.1
219	2023-11-22 15:06:55	00:05:52	D12	3042	abb	1	Emergency_Stop	DB931.X380.1
220	2023-11-22 15:06:54	00:05:53	D11	2722	abb	1	Emergency_Stop	DB931.X340.1
221	2023-11-22 15:06:28	00:06:18	D11	2739	abb	1	RunChain_Not_Closed	DB931.X342.2
222	2023-11-22 15:06:28	00:06:18	D21	2899	abb	1	RunChain_Not_Closed	DB931.X362.2
223	2023-11-22 15:06:28	00:06:18	R23	979	abb	1	RunChain_Not_Closed	DB931.X122.2
224	2023-11-22 15:06:28	00:06:18	R22	659	abb	1	RunChain_Not_Closed	DB931.X82.2
225	2023-11-22 15:06:28	00:06:18	R12	499	abb	1	RunChain_Not_Closed	DB931.X62.2
226	2023-11-22 15:06:28	00:06:18	R21	339	abb	1	RunChain_Not_Closed	DB931.X42.2
227	2023-11-22 15:06:28	00:06:18	R11	179	abb	1	RunChain_Not_Closed	DB931.X22.2
228	2023-11-22 15:06:28	00:06:18	D12	3059	abb	1	RunChain_Not_Closed	DB931.X382.2
229	2023-11-22 15:06:28	00:06:18	D22	3219	abb	1	RunChain_Not_Closed	DB931.X402.2
230	2023-11-22 15:06:28	00:06:18	H21	3539	abb	1	RunChain_Not_Closed	DB931.X442.2
231	2023-11-22 15:06:28	00:06:18	H11	3379	abb	1	RunChain_Not_Closed	DB931.X422.2
232	2023-11-22 15:06:25	00:06:21	R21	326	abb	1	Applicator_Error	DB931.X40.5
233	2023-11-22 15:06:25	00:06:21	R11	166	abb	1	Applicator_Error	DB931.X20.5
234	2023-11-22 15:05:18	00:07:28	R12	486	abb	1	Applicator_Error	DB931.X60.5
235	2023-11-22 15:05:03	00:07:43	R22	646	abb	1	Applicator_Error	DB931.X80.5
236	2023-11-22 15:04:45	00:08:01	R23	966	abb	1	Applicator_Error	DB931.X120.5
237	2023-11-22 15:04:34	00:00:03	R11	166	abb	1	Applicator_Error	DB931.X20.5
238	2023-11-22 15:01:14	00:01:56	R13	802	Null	1	Emergency_Stop	DB931.X100.1
239	2023-11-22 15:00:54	00:02:16	R12	482	Null	1	Emergency_Stop	DB931.X60.1
240	2023-11-22 14:42:17	00:17:00	R21	321	Null	1	HeartBeat_Fault	DB931.X40.0

Search

Export

8

395

In addition to the standard alarm report format, the alarm report offers a unique feature known as Top Frequency mode. When enabled, Top Frequency mode aggregates alarm occurrences and arranges them in descending order based on frequency. This allows users to quickly identify and prioritize the most frequently occurring alarms.

Report: Alarm		
Option		
☐ Normal ☒ Top Frequency		
Date Filter		
Start Time 2023-11-01 00:00:00		
End Time 2024-02-19 14:44:27		
Last Day Today Next Day		
Other Filter		
Device ALL		
Shift ALL		
Message		
Order By Time		
☐ Asc ☒ Desc		
Search Export		
	Message(English)	Count
1	RunChain_Not_Closed	456
2	Emergency_Stop	232
3	HeartBeat_Fault	165
4	Robot_Bypassed	163
5	Not_At_Home_Position	52
6	Applicator_Error	38
7	System_Error	36
8	Command_Failed	30
9	Maintenance_Mode	7
10	Applicator_Bypassed	5
11	CmdExecuting_Overtime	5
12	PowerOff_Power_Consumpt	4
13	CommErr_CV	4
14	Safety_Module_AckReq	4
15	Comm_Err_LS	4
16	High_Voltage_Error	3
17	Not_In_Automatic_Mode	3
18	AStop_Active	3
19	Gstop_Active	3
20	Bus_Fault	2
21	LC_Bypassed_Outlet	2
22	PowerOff_Sensor	2
23	Estop_Active_Outlet_R	2
24	LC_Bypassed_Inlet	2
25	Estop_Active_CV	2
26	System_Not_Running	1

6.7.5 Command log

Overview

Command report displays PLC and HMI command records sent to a robot with a paint system. Command report displays command sent time, robot name, command source, command number, command parameters and command feedback time.

Report: Command Log						
Date Filter						
Start Time 2023-12-01 00:00:00						
End Time 2024-02-19 14:56:16						
Last Day Today Next Day						
Other Filter						
Device ALL						
Source ALL						
Command No. R11 R21 R12 R22 R13 R23 R14 R24						
Order By Time						
☐ Asc ☒ Desc						
Search Export						
	Time	Device	Source	Command	Parameter	Feedback
31	2024-01-11 11:56:45	R23	PLC	103	0.0.0.0.0.0.0.0	2024-01-11 11:56:45
32	2024-01-11 11:56:45	R22	PLC	103	0.0.0.0.0.0.0.0	2024-01-11 11:56:45
33	2024-01-11 11:56:45	R22	PLC	103	0.0.0.0.0.0.0.0	2024-01-11 11:56:45
34	2024-01-11 11:56:45	R12	PLC	103	0.0.0.0.0.0.0.0	2024-01-11 11:56:45
35	2024-01-11 11:56:45	R23	PLC	103	0.0.0.0.0.0.0.0	2024-01-11 11:56:45
36	2024-01-11 11:56:45	R13	PLC	103	0.0.0.0.0.0.0.0	2024-01-11 11:56:45
37	2024-01-11 11:56:45	R14	PLC	103	0.0.0.0.0.0.0.0	2024-01-11 11:56:45
38	2024-01-11 11:56:45	R21	PLC	103	0.0.0.0.0.0.0.0	2024-01-11 11:56:45
39	2024-01-11 11:56:45	R11	PLC	103	0.0.0.0.0.0.0.0	2024-01-11 11:56:45
40	2024-01-11 11:56:44	R11	PLC	314	0.0.0.0.0.0.0.0	2024-01-11 11:56:44
41	2024-01-11 11:56:44	R11	PLC	314	0.0.0.0.0.0.0.0	2024-01-11 11:56:44
42	2024-01-11 11:56:44	R11	PLC	314	0.0.0.0.0.0.0.0	2024-01-11 11:56:44
43	2024-01-11 11:56:44	R11	PLC	314	0.0.0.0.0.0.0.0	2024-01-11 11:56:44
44	2024-01-11 11:56:44	R21	PLC	314	0.0.0.0.0.0.0.0	2024-01-11 11:56:44
45	2024-01-11 11:56:44	R21	PLC	314	0.0.0.0.0.0.0.0	2024-01-11 11:56:44
46	2024-01-11 11:56:44	R21	PLC	314	0.0.0.0.0.0.0.0	2024-01-11 11:56:44
47	2024-01-11 11:56:44	R12	PLC	314	0.0.0.0.0.0.0.0	2024-01-11 11:56:44
48	2024-01-11 11:56:44	R12	PLC	314	0.0.0.0.0.0.0.0	2024-01-11 11:56:44
49	2024-01-11 11:56:44	R22	PLC	314	0.0.0.0.0.0.0.0	2024-01-11 11:56:44
50	2024-01-11 11:56:43	R22	PLC	2	0.0.0.0.0.0.0.0	2024-01-11 11:56:43
51	2024-01-11 11:56:43	R12	PLC	2	0.0.0.0.0.0.0.0	2024-01-11 11:56:43
52	2024-01-11 11:56:43	R21	PLC	2	0.0.0.0.0.0.0.0	2024-01-11 11:56:43
53	2024-01-11 11:56:43	R11	PLC	2	0.0.0.0.0.0.0.0	2024-01-11 11:56:43
54	2024-01-11 11:56:43	R13	PLC	2	0.0.0.0.0.0.0.0	2024-01-11 11:56:43
55	2024-01-11 11:56:43	R14	PLC	2	0.0.0.0.0.0.0.0	2024-01-11 11:56:43
56	2024-01-11 11:56:43	R24	PLC	2	0.0.0.0.0.0.0.0	2024-01-11 11:56:43
57	2024-01-11 11:56:43	R12	PLC	2	0.0.0.0.0.0.0.0	2024-01-11 11:56:43
58	2024-01-11 11:56:43	R22	PLC	2	0.0.0.0.0.0.0.0	2024-01-11 11:56:43
59	2024-01-11 11:56:43	R22	PLC	2	0.0.0.0.0.0.0.0	2024-01-11 11:56:43
60	2024-01-11 11:56:43	R21	PLC	2	0.0.0.0.0.0.0.0	2024-01-11 11:56:43

6.8 Program

Following sub-sections are included in this section:

6.8.1 '[RobotStudio](#)' on page 237

6.8.2 '[RobView](#)' on page 237

6.8.3 '[Gateway RD](#)' on page 238

6.8.4 '[VMT RD](#)' on page 238

6.8.1 RobotStudio

A shortcut for the Robot Studio software which provides convenient and direct access to the application from the system interface. This feature eliminates the need for manual navigation through file directories or searching for the application, allowing users to quickly utilize the powerful functionality of Robot Studio.



6.8.2 RobView

A shortcut to the RobView software, which is frequently used for visualizing and monitoring robot operations, programming, and diagnostics, allows users to access the application directly from the system interface with a single click. This eliminates the need for manual navigation through file directories or searching for the application.



6.8.3 Gateway RD

A shortcut to the Gateway RD software, which is frequently used for designing seamless connectivity.



6.8.4 VMT RD

A shortcut to the VMT RD software, which is dynamic and visually distinctive road that adds a unique character. It often features vibrant, pixel aesthetics and is designed to facilitate smooth movement. With its carefully crafted intersections and engaging visual elements, VMT RD enhances game play by providing an interesting and efficient route for players to explore and connect different areas.



6.9 Help

Following sub-sections are included in this section:

6.9.1 ['RobotUI'](#) on page 239

6.9.2 ['Robot'](#) on page 239

6.9.3 ['Exit'](#) on page 239

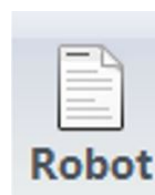
6.9.1 RobotUI

A shortcut for the RobotUI software manual provides users with quick access to the documentation or user manual of the RobotUI software directly from the system interface. This shortcut is typically linked to a specific file or URL where users can find detailed instructions, guidelines, and information on using the RobotUI software effectively.



6.9.2 Robot

A shortcut for the robot user manual ensures easy access to documentation for users whenever assistance or clarification is needed on operating the robot, programming it for specific tasks, or troubleshooting any issues that may arise during operation. This promotes user self-sufficiency and helps enhance overall user experience by providing easy access to helpful resources.



6.9.3 Exit

Exit the RobotUI software.



7 Robot system

About this chapter

This chapter provides information on the user level of the robot file structure and how to create or change files. It also references all available RAPID functions and procedures offered by the reference manual.

The following sections are included:

'System description' on page 240

'Paint control IO Interface' on page 241

'Paint control command interface' on page 258

'Paint instructions' on page 284

'Paint control functions' on page 305

'Paint control instructions' on page 342

'Troubleshooting' on page 385

7.1 System description

The following sub-sections are included in this section:

'Robot type' on page 240

'Application type' on page 240

'IO protocol type' on page 241

'Safety interface' on page 241

7.1.1 Robot type

The system supports the following robots.

No.	Robot	Application
1	IRB 5500	Paint application
2	IRB 5350	Paint application
3	IRB 6700	Cleaning (FD robot)
4	IRB 4600	Checking

7.1.2 Application type

The system is compatible with the following application types.

No.	Application type
1	MPAC 1K
2	MPAC 2K
3	RCC 1K

7 Robot system

7.2 Paint control IO Interface

No.	Application type
4	RCC 2K
5	CBS
6	Dual system (1 Resin + 2 Cat)
7	Dual system (2 Resin + 1 Cat)
8	Dual system (2 Resin + 2 Cat)
9	Pixel paint

7.1.3 IO protocol type

The system accommodates the following IO protocols.

No.	Application type	IO type
1	Command IO	CC-Link, PROFIBUS
2	Full Command IO	PROFINET, Ethernet/IP

The default support IO type is “Full command IO”.

7.1.4 Safety interface

The system supports the following safety interface.

No.	Safety type
1	Hardware safety connection (SCB-01)
2	PROFIsafe (SafeMove 2)

7.2 Paint control IO Interface

Overview

This chapter provides information on the interface used to communicate between the robot and the PLC (Programmable Logic Controller).

The following sub-sections are included in this section:

[‘I/O interface’](#) on page 241

[‘Paint control IO \(Paint, Pixel paint, Clean robot, Flaming robot, Dispense robot I/O signals\)’](#) on page 242

7.2.1 I/O interface

Description

The Command I/O Protocol is a communication channel utilized between the station PLC and the robot. For more information about this protocol and its implementation, see ABB paint documentation [‘Reference manual, paint commands and PLC interface IRC5P RW6.0’](#) on page 15".

Regarding the full scalability, all I/O signals described in the documentation of the Command I/O Protocol are defined and used identically.

The robot supports multiple fieldbus types to communicate with PLC, as shown in the table below.

Fieldbus type	Bytes	Device
Profinet	64	Robot < -- > PLC
Ethernet/IP	64	Robot < -- > PLC
CC-Link	32	Robot < -- > PLC
Profibus	32	Robot < -- > PLC
Profisafe	8	Robot < -- > PLC
CIP	8	Robot < -- > PLC

The robot signals are defined in the robot EIO file; the robot signal type consists of digital input, digital output, group input and group output.

Input	Description	Output	Description
DI	Digital input	DO	Digital output
GI	Group input	GO	Group output

7.2.2 Paint control IO (Paint, Pixel paint, Clean robot, Flaming robot, Dispense robot I/O signals)

The following sub-sections are included in this section:

['Paint I/O signals'](#) on page 242

['Pixel paint I/O signals'](#) on page 256

['Clean robot I/O signals'](#) on page 256

['Flaming robot I/O signals'](#) on page 256

['Dispense robot I/O signals'](#) on page 257

7.2.2.1 Paint I/O signals

Address	Type	Signal name	Description	Station
0	DO	AutomaticMode	Auto mode feedback	All
1	DO	MasterGranted	Master status feedback	All
2	DO	Run chain closed	Run chain closed	All
3	DO	RobotError	Robot error status	All
4	DO	JobInProgress	Executing a program, including pre and post events	All
5	DO	JobPending	A program is waiting to be executed	All
6	DO	ReadyHighPri	Ready for high priority program	All
7	DO	SystemRunning	System is in its production state	All
8	DO	MtrlChRunning	Material change is not in the suspended state	All

7 Robot system

7.2 Paint control IO Interface

Address	Type	Signal name	Description	Station
9	DO	MtrlChangeOn	Material change is pending or executing	All
10	DO	ObjConnected	Conveyor object connected	All
11	DO	MtrlSupplyError	Material supply is in error state	All
12	DO	AppEnabled	Applicator is enabled	All
13	DO	AppError	Applicator is in error state	All
14	DO	HVEnabled	High voltage is enabled	All
15	DO	HVError	High voltage is in error state	All
16	DO	HeartBeat	This signal changes state at a 2 Hz frequency	All
17	DO	PotLifeWarning	Potlife timers in warning state	All
18	DO	PotlifeTimeOut	Potlife timers is timed out	All
19	DO	HomePos	Manipulator is in home position	All
20	DO	MtrlChangePos	Manipulator is in material change position	All
21	DO	ReadyPos	Manipulator is in ready position	All
22	DO	UserPos	Manipulator is in user position	All
23	DO	ReposFinished	Indicates that repos is finished	All
24	DO	EmergencyStop	Controller is in emergency stop state	All
25	DO	JobComplete	A signal is pulsed when a paint program has completed its path	All
26	DO	GoingToAuto	Operating mode selector is in auto, but auto mode not yet acknowledged	All
27	DO	ReadyToReceive	Controller is ready to receive normal jobs (the queue is not full)	All
28	DO	MotorIsOn	Motors are on	All
29	DO	ConveyorRunning	Conveyor is running from the robot's point of view	All
30	DO	CBSMotorIsOn	CBS servo motor is enabled	All
31	DO	ProgramLoaded	If the external start option is enabled, then this signal tells that a program is loaded and the controller is waiting for an external start command	All
32	DO	CommandOK	Command OK	All
33	DO	CommandFailed	Command Failed	All
34	DO	BellRotating	Indicates if bell on atomizer is rotating	All
35	DO	MotionSupTrigg	Activated when motion supervision is triggered	All
36	DO	ESavingBlocked	Status of system output "EnergySavingBlocked" (Only available with ProfiEnergy option)	All
37	DO	PowerError	A problem has been detected with the power supply	All

Address	Type	Signal name	Description	Station
38	DO	AutoModeReqAck	System is ready to accept acknowledge of automatic mode after remote automatic mode request (Only available with SafeMove option)	All
39	DO	IPSNodesReady	All IPS nodes have finished loading configuration and the paint process is ready for operation	All
40-46	DO	Spare	Spare	All
47	DO	CommandAck	Acknowledge of the command. This bit is set when the command has finished, and the results are available in OutData	All
48-63	GO	OutData1	Output parameter # 1 in the command I/O protocol (If the command fails, this output group contains the error number)	All
64-79	GO	OutData2	Output parameter # 2 in the command I/O protocol	All
80-95	GO	OutData3	Output parameter # 3 in the command I/O protocol	All
96-111	GO	goProgramNo	Program number feedback	All
112-127	GO	goMaterialNo	Colour number feedback	All
128-143	GO	goSkidNo	Skid number feedback	All
144-159	GO	clpositionFB	Robot Conveyor Position feedback	Tracking
160-175	GO	goWorkflowStatus	Workflow status	All
176	DO	doRobotOK	Show robot is ok to PLC	Stop&Go
177	DO	doRobIsPainting	Robot is painting	All
178	DO	dopaintfinish	Robot painting finish	All
179	DO	doGhostNonMotion	Status show robot is in non-motion mode	All
180	DO	doCheckCMD421	Set flag bit when executed CMD421	All
181	DO	doPLCBypassHV	Status show PLC no need check HV enable signal	Painter
182	DO	DegradeReadyPos	Robot in degrade ready position	All
183	DO	doHVChainIsOn	HV chain is on	Painter
184	DO	CalPos	Robot in calibration position	All
185	DO	MaintPos	Robot in maintain position	All
186	DO	BypassPos	Robot in bypass position	All
187	DO	ManCleanPos	Robot in manual clean position	All
188	DO	DressPos	Robot in dress position	All
189	DO	GunCleanPos	Robot in gun clean position	Painter
190	DO	BellCleanPos	Robot in bell clean position	Painter
191	DO	HubCleanPos	Robot in hub clean position	Painter
192	DO	ToolCleanPos	Robot in opener tool clean position	Opener
193	DO	AppMaintPos	Robot in Applicator Maintenance position	All

7 Robot system

7.2 Paint control IO Interface

Address	Type	Signal name	Description	Station
194	DO	TCPCheckPos	Robot in calibration checking position	All
195	DO	Spare	Spare	All
196	DO	Spare	Spare	All
197	DO	Spare	Spare	All
198	DO	Spare	Spare	All
199	DO	Spare	Spare	All
200-201	GO	goConveyorNo	Send conveyor device number to PLC	All
202	DO	doGreaseStart	Start grease for automatic lubricating device (Optional function)	E-axis
203	DO	doGreaseMove	Move grease for automatic lubricating device (Optional function)	E-axis
204	DO	doMotorOn	Motors are on	All
205	DO	doBellCupMiss	Bell cup miss alarm	Painter
206	DO	doPumpNoSignal	Pump not running	Painter
207	DO	MtrlUsageReady	Material right for PLC reading	Painter
208	DO	doA1AirSupPSLow	Main air pressure low	Painter
209	DO	doA1AirHPSupPSLow	High pressure air pressure low	Painter
210	DO	doA1BearPSLow	Bearing air pressure low	Painter
211	DO	doA1SolvPSLow	Solvent pressure low	Painter
212	DO	doA1Servo1Dev	Servo 1 deviation alarm	Painter
213	DO	doA1Servo2Dev	Servo 2 deviation alarm	Painter
214	DO	doA1Servo3Dev	Servo 3 deviation alarm	Painter
215	DO	doA1Pump1PSOutHigh	Pump1 out pressure high	Painter
216	DO	doA1Pump2PSOutHigh	Pump2 out pressure high	Painter
217	DO	doA1Pump3PSOutHigh	Pump3 out pressure high	Painter
218	DO	doA1Pump1PSInLow	Pump1 in pressure low	Painter
219	DO	doA1Pump2PSInLow	Pump2 in pressure low	Painter
220	DO	doA1Pump3PSInLow	Pump3 in pressure low	Painter
221	DO	doA1AtomDev	Atom deviation alarm	Painter
222	DO	doA1Shape1Dev	Shape1 deviation alarm	Painter
223	DO	doA1Shape2Dev	Shape2 deviation alarm	Painter
224-227	GO	GoA1HVErrNo	1 - Current value too high 2 - Slope current value too high 3 - Transformer current high 4 - Actual voltage too high/low 5 - Watchdog 6 - Internal voltage error 7- 24 volt power missing 8- Cascade error 9- Dsp communication error	Painter
228-231	GO	Spare	-	-
232	DO	doPlcCheckJobInfo	Set flag bit for PLC check job information	All

Address	Type	Signal name	Description	Station
233	DO	doHVTestFinished	Status show all HV test finished	Painter
234	DO	doHVDyTestPassed	Status show HV dynamic test finished	Painter
235	DO	doHVStaticTestPassed	Status show HV static test finished	Painter
236	DO	ConveyorHoldReq	Set flag bit for PLC hold conveyor	All
237	DO	doCnvSyncAck	Set flag bit for sync start signal acknowledge	Tracking
238	DO	SafePosition	Robot in safe position	All
239	DO	doReleasePin	Release pin for PLC to transport system	Stop&Go
240-253	GO	goPositionCode	Position code for robot	All
254-255	GO	Spare	-	-
256-271	GO	Go_extaxpos	Rail position feedback	All
272	DO	doFB_R01	Send to painter 01 master feedback	Opener
273	DO	doFB_R02	Send to painter 02 master feedback	Opener
274	DO	doFB_R03	Send to painter 03 master feedback	Opener
275	DO	doFB_R04	Send to painter 04 master feedback	Opener
276	DO	doFB_R05	Send to painter 05 master feedback	Opener
277	DO	doFbPLC	Send to PLC master feedback	Opener
278-281	GO	goFbPosition	Feedback door position to Painter	Opener
282	DO	GripperErr	Gripper error	Opener
283-303	GO	Spare	-	-
304-307	GO	goPos_H01	Request position for Hood	Painter
308-311	GO	goPos_D01	Request position for front door	Painter
312-315	GO	goPos_D02	Request position for back door	Painter
316-319	GO	goPos_H02	Request position for trunk	Painter
320	DO	doReq_H01	Request master hood opener	Painter
321	DO	doReq_D01	Request master front door opener	Painter
322	DO	doReq_D02	Request master back door opener	Painter
323	DO	doReq_H02	Request master trunk opener	Painter
324-335	GO	Spare	Spare	
336	DO	doRobotInterlock1	Robot interlock signal	All
337	DO	doRobotInterlock2	Robot interlock signal	All
338	DO	doRobotInterlock3	Robot interlock signal	All
339	DO	doRobotInterlock4	Robot interlock signal	All
340	DO	doRobotInterlock5	Robot interlock signal	All
341	DO	doRobotInterlock6	Robot interlock signal	All
342	DO	doRobotInterlock7	Robot interlock signal	All
343	DO	doRobotInterlock8	Robot interlock signal	All
344	DO	doRobotInterlock9	Robot interlock signal	All

7 Robot system

7.2 Paint control IO Interface

Address	Type	Signal name	Description	Station
345	DO	doRobotInterlock10	Robot interlock signal	All
346	DO	doRobotInterlock11	Robot interlock signal	All
347	DO	doRobotInterlock12	Robot interlock signal	All
348	DO	doRobotInterlock13	Robot interlock signal	All
349	DO	doRobotInterlock14	Robot interlock signal	All
350	DO	doRobotInterlock15	Robot interlock signal	All
351	DO	doRobotInterlock16	Robot interlock signal	All
352	DO	doProcessStatus_1	Process status send to PLC	Painter
353	DO	doProcessStatus_2	Process status send to PLC	Painter
354	DO	doProcessStatus_3	Process status send to PLC	Painter
355	DO	doProcessStatus_4	Process status send to PLC	Painter
356	DO	doProcessStatus_5	Process status send to PLC	Painter
357	DO	doProcessStatus_6	Process status send to PLC	Painter
358	DO	doProcessStatus_7	Process status send to PLC	Painter
359	DO	doProcessStatus_8	Process status send to PLC	Painter
360	DO	doProcessStatus_9	Process status send to PLC	Painter
361	DO	doProcessStatus_10	Process status send to PLC	Painter
362	DO	doProcessStatus_11	Process status send to PLC	Painter
363	DO	doProcessStatus_12	Process status send to PLC	Painter
364	DO	doProcessStatus_13	Process status send to PLC	Painter
365	DO	doProcessStatus_14	Process status send to PLC	Painter
366	DO	doProcessStatus_15	Process status send to PLC	Painter
367	DO	doProcessStatus_16	Process status send to PLC	Painter
368	DO	CBCAct	Cyclic brake check is currently running	All
369	DO	CBCErr	Cyclic brake check performed with an error	All
370	DO	CBCOK	Cyclic brake check performed successfully	All
371	DO	CBCPreWarn	Pre-warning time for cyclic brake check expired	All
372	DO	CBCReq	Cyclic brake check required	All
373	DO	CBCWarn	Cyclic brake check performed with a warning	All
374-375	GO	Spare	-	-
376	DO	doFlame_Burning	Flame or Feather Duster is running	FL&FD
377	DO	doFlame_Ready	Flame or Feather Duster is ready	FL&FD
378-383	GO	Spare	Spare	Paint Checker
384	DO	do_Measure_Job_Start	Paint check job start (AIM)	Paint Checker
385	DO	do_Measure_Trigger	Paint checker trigger measurement (AIM)	Paint Checker
386	DO	do_Measure_Job_Finished	Paint checker Job finish (AIM)	Paint Checker

Address	Type	Signal name	Description	Station
387	DO	do_CSVFileReady	Paint checker CSV file ready (AIM)	Paint Checker
388-391	GO	Spare	Spare	
392	DO	PumpIsOn	PixelPaint PumpsOn state (precondition)	Pixel paint
393	DO	PixelPaintOK	PixelPaintOK state (precondition)	Pixel paint
394	DO	ImagesLoaded	Image load succeed	Pixel paint
395	DO	ImageLoadFailed	Image load Failed	Pixel paint
396-397	GO	HeadPowerState	Head Power State: 0 - Power Off 1 - HDC Only 2 - Power ON	Pixel paint
398-401	GO	PrinterState	Printer States: 0 - MPS_DISCONNECTED 1 - MPS_CONNECTED 2 - MPS_IDLE 3 - MPS_READY 4 - MPS_PRINTING 5 - MPS_INITIALIZING 6 - MPS_STARTUP 7 - MPS_FAULT 8 - MPS_LOADING	Pixel paint
402-405	GO	Spare	-	Pixel paint
406	DO	do_Vision_RevDataOK	Vision system receiver data ok	Pixel paint
407	DO	do_Vision_SendDataOK	Vision system send data ok	Pixel paint
408	DO	do_Vision_ManualStart	Vision system manual start	Pixel paint
409	DO	do_Vision_BK_Error	BK error	Pixel paint
410	DO	do_Vision_BK_MeasError	BK measure error	Pixel paint
411	DO	do_Vision_BK_ComError	-	Pixel paint
412	DO	do_Vision_BK_ColliError	-	Pixel paint
413	DO	do_Vision_NoZMeasError	-	Pixel paint
414	DO	do_Vision_NoNozOK	-	Pixel paint
415	DO	Spare	-	Pixel paint
416-431	GO	Go_TransX	The offset in X	Pixel paint
432-447	GO	Go_TransY	The offset in Y	Pixel paint
448-463	GO	Go_TransZ	The offset in Z	Pixel paint
464-479	GO	Go_RotX	The rotation Rx	Pixel paint
480-495	GO	Go_RotY	The rotation Ry	Pixel paint
496-511	GO	Go_RotZ	The rotation Rz	Pixel paint

7 Robot system

7.2 Paint control IO Interface

Input signal mapping

Address	Type	Signal name	Description	Used by
0-15	GI	Command	Paint command number	All
16-31	GI	InParam1	Input parameter # 1	All
32-47	GI	InParam2	Input parameter # 2	All
48-63	GI	InParam3	Input parameter # 3	All
64-79	GI	InParam4	Input parameter # 4	All
80-95	GI	InParam5	Input parameter # 5	All
96-111	GI	InParam6	Input parameter # 6	All
112-127	GI	InParam7	Input parameter # 7	All
128-143	GI	InParam8	Input parameter # 8	All
144-159	GI	InParam9	Input parameter # 9	All
160-175	GI	InParam10	Input parameter # 10	All
176	DI	CommandToggle	Executes Paint command # <Command> with the InParam(1~10)	All
177-191	DI	Spare	Spare	All
192	DI	diGhostMode	Set robot to ghost mode	All
193	DI	diStationOK	Station ok, robot could execute	All
194	DI	diJobsAreDone	Job finish from PLC	All
195	DI	diSolvPSLow	Solvent pressure low (Only external solvent pressure sensor)	All
196	DI	diGhostNonMotion	Ghost non motion mode from PLC	All
197	DI	DiBoothIsEmpty	Status show station is not empty	All
198	DI	diHomeMoveEn	Home position enable	All
199-201	DI	Spare	Spare	All
202	DI	diMotorOff	Robot Motor off	All
203	DI	diResetErr	Reset error	All
204	DI	diStopSignal	Robots stop signal	All
205	DI	diGhostCnv	Ghost conveyors select (1-Ghost cnv; 0-Real cnv)	All
206	DI	diSimCnvRun	Ghost conveyor running(1-run;0-stop)	All
207	DI	diGreaseRunning	Grease system running (With rail)	E-axis
208	DI	diGreaseMove	Grease moves for automatic lubricating device (Option function)	E-axis
209	DI	diAppEnable	Applicator enable	Painter
210	DI	diHvEnable	HV enable	Painter
211	DI	AutoModeRequest	Robot auto mode request (Profisafe)	All
212	DI	ManModeRequest	Robot manual mode request (Profisafe)	All
213	DI	diCNVReady_PLC	Convery running is ready	Flame

Address	Type	Signal name	Description	Used by
214	DI	Spare	Spare	All
215	DI	diStationAuto	1-Station auto; 0-Station manual	All
216-219	GI	GiStationOpMode	Station status 3-Station maint mode 4-Station cleaning mode 5-Station teach mode	All
220-223	GI	Spare	Spare	
224-239	GI	GiSimSpeed	Ghost conveyor speed to robot (xx mm/s * 100)	All
240	DI	diSafePos_R11	R11 in safe position	All
241	DI	diSafePos_R21	R21 in safe position	All
242	DI	diSafePos_R12	R12 in safe position	All
243	DI	diSafePos_R22	R22 in safe position	All
244	DI	diSafePos_R13	R13 in safe position	All
245	DI	diSafePos_R23	R23 in safe position	All
246	DI	diSafePos_R14	R14 in safe position	All
247	DI	diSafePos_R24	R24 in safe position	All
248	DI	diSafePos_D11	D11 in safe position	Interior
249	DI	diSafePos_D21	D21 in safe position	Interior
250	DI	diSafePos_D12	D12 in safe position	Interior
251	DI	diSafePos_D22	D22 in safe position	Interior
252	DI	diSafePos_H11	H11 in safe position	Interior
253	DI	diSafePos_H21	H21 in safe position	Interior
254	DI	diSafePos_H12	H12 in safe position	Interior
255	DI	diSafePos_H22	H22 in safe position	Interior
256	DI	DiBypassed_R11	R11 bypassed	All
257	DI	DiBypassed_R21	R21 bypassed	All
258	DI	DiBypassed_R12	R12 bypassed	All
259	DI	DiBypassed_R22	R22 bypassed	All
260	DI	DiBypassed_R13	R13 bypassed	All
261	DI	DiBypassed_R23	R23 bypassed	All
262	DI	DiBypassed_R14	R14 bypassed	All
263	DI	DiBypassed_R24	R24 bypassed	All
264	DI	DiBypassed_D11	D11 bypassed	Interior
265	DI	DiBypassed_D21	D21 bypassed	Interior
266	DI	DiBypassed_D12	D12 bypassed	Interior
267	DI	DiBypassed_D22	D22 bypassed	Interior
268	DI	DiBypassed_H11	H11 bypassed	Interior
269	DI	DiBypassed_H21	H21 bypassed	Interior
270	DI	DiBypassed_H12	H12 bypassed	Interior
271	DI	DiBypassed_H22	H22 bypassed	Interior

7 Robot system

7.2 Paint control IO Interface

Address	Type	Signal name	Description	Used by
272	DI	diReqMaster_R01	Master control request from R01	Opener
273-276	GI	GiReqPosition_R01	Request opener position from R01	Opener
277	DI	diReqMaster_R02	Master control request from R02	Opener
278-281	GI	GiReqPosition_R02	Request opener position from R02	Opener
282	DI	diReqMaster_R03	Master control request from R03	Opener
283-286	GI	GiReqPosition_R03	Request opener position from R03	Opener
287	DI	diReqMaster_R04	Master control request from R04	Opener
288-291	GI	GiReqPosition_R04	Request opener position from R04	Opener
292	DI	diReqMaster_R05	Master control request from R05	Opener
293-296	GI	GiReqPosition_R05	Request opener position from R05	Opener
297	DI	diReqMaster_PLC	Master control request from PLC	Opener
298-301	GI	GiReqPosition_PLC	Request opener position from PLC	Opener
302	DI	diSensorEnable	Sensor enables	Opener
304-307	GI	giPos_H01	H01 opener position	I-Painter
308-311	GI	giPos_D01	D01 opener position	I-Painter
312-315	GI	giPos_D02	D02 opener position	I-Painter
316-319	GI	giPos_H02	H02 opener position	I-Painter
320	DI	diMaster_H01	Get H01 opener master	I-Painter
321	DI	diMaster_D01	Get D01 opener master	I-Painter
322	DI	diMaster_D02	Get D02 opener master	I-Painter
323	DI	diMaster_H02	Get H02 opener master	I-Painter
324-335	GI	Spare	Spare	-
336	DI	diRobotInterlock1	Interlock signals from PLC	All
337	DI	diRobotInterlock2	Interlock signals from PLC	All
338	DI	diRobotInterlock3	Interlock signals from PLC	All
339	DI	diRobotInterlock4	Interlock signals from PLC	All
340	DI	diRobotInterlock5	Interlock signals from PLC	All
341	DI	diRobotInterlock6	Interlock signals from PLC	All
342	DI	diRobotInterlock7	Interlock signals from PLC	All
343	DI	diRobotInterlock8	Interlock signals from PLC	All
344	DI	diRobotInterlock9	Interlock signals from PLC	All
345	DI	diRobotInterlock10	Interlock signals from PLC	All
346	DI	diRobotInterlock11	Interlock signals from PLC	All
347	DI	diRobotInterlock12	Interlock signals from PLC	All
348	DI	diRobotInterlock13	Interlock signals from PLC	All
349	DI	diRobotInterlock14	Interlock signals from PLC	All
350	DI	diRobotInterlock15	Interlock signals from PLC	All
351	DI	diRobotInterlock16	Interlock signals from PLC	All
352	DI	diProcessStatus1	Process status from PLC	All
353	DI	diProcessStatus2	Process status from PLC	All

Address	Type	Signal name	Description	Used by
354	DI	diProcessStatus3	Process status from PLC	All
355	DI	diProcessStatus4	Process status from PLC	All
356	DI	diProcessStatus5	Process status from PLC	All
357	DI	diProcessStatus6	Process status from PLC	All
358	DI	diProcessStatus7	Process status from PLC	All
359	DI	diProcessStatus8	Process status from PLC	All
360	DI	diProcessStatus9	Process status from PLC	All
361	DI	diProcessStatus10	Process status from PLC	All
362	DI	diProcessStatus11	Process status from PLC	All
363	DI	diProcessStatus12	Process status from PLC	All
364	DI	diProcessStatus13	Process status from PLC	All
365	DI	diProcessStatus14	Process status from PLC	All
366	DI	diProcessStatus15	Process status from PLC	All
367	DI	diProcessStatus16	Process status from PLC	All
368	DI	DiBrakeReleaseEn	Brake release enable signal from PLC	All
369	DI	DiBrakeAxis1	Release axis 1 brake	All
370	DI	DiBrakeAxis2	Release axis 2 brake	All
371	DI	DiBrakeAxis3	Release axis 3 brake	All
372	DI	DiBrakeAxis4	Release axis 4 brake	All
373	DI	DiBrakeAxis5	Release axis 5 brake	All
374	DI	DiBrakeAxis6	Release axis 6 brake	All
375	DI	DiBrakeAxis7	Release axis 7 brake	All
376	DI	FD_IonicBarBypass	Feathers duster iron bar bypass	FD
377	DI	FD_CollisionBypass	Feathers duster collision bypass	FD
378	DI	FD_AirSpeedBypass	Feathers duster air speed bypass	FD
379-383	GI	Spare	Spare	-
384	DI	Di_MeasureSytemEnabl e	Measurement system enable	Paint Checker
385	DI	Di_MeasureSytem_Ok	Measurement system ok	Paint Checker
386	DI	Di_colour_Ok	Colour ok	Paint Checker
387	DI	Di_Measure_Done	Measurement done	Paint Checker
388	DI	Di_Measure_DataReady	Measurement data ready	Paint Checker
389	DI	Di_Measure_Error	Measurement error	Paint Checker

7 Robot system

7.2 Paint control IO Interface

Address	Type	Signal name	Description	Used by
390	DI	Di_CSV_Read_Done	CSV ready finished from HMI	Paint Checker
391-405	GI	Spare	Spare	Paint Checker
406	DI	Di_Vision_SendDataOK	-	Paint Checker
407	DI	Di_Vision_ReqReadData	-	Paint Checker
408	DI	Di_Vision_DataCompareNOK	-	Paint Checker
409	DI	Di_Vision_Dist_Offs1	-	Paint Checker
410	DI	Di_Vision_Dist_Offs2	-	Paint Checker
411	DI	Di_Vision_Dist_Offs3	-	Paint Checker
412	DI	Di_Vision_Angl_Offs1	-	Paint Checker
413	DI	Di_Vision_Angl_Offs2	-	Paint Checker
414	DI	Di_Vision_Simulation	-	Paint Checker
415	DI	Di_BKEnable	-	Paint Checker
416-431	GI	Gi_TransX	-	Paint Checker
432-447	GI	Gi_TransY	-	Paint Checker
448-463	GI	Gi_TransZ	-	Paint Checker
464-479	GI	Gi_RotX	-	Paint Checker
480-495	GI	Gi_RotY	-	Paint Checker
496-511	GI	Gi_RotZ	-	Paint Checker

Safety signals

Safety output signals from robot to PLC:

Address	Type	Signal name	Description	Station
512	DO	Fo_SafetyControllerOperational		All
513	DO	Fo_SafetySynchronized	Safety synchronized status	All
514	DO	Fo_ExtComShutdownReq	External shut down request	All
515	DO	Fo_ConfigurationLocked	Safety configuration locked	All
516	DO	Fo_CommissioningMode	Commissioning mode	All
517	DO	Fo_ServiceMode	Service mode	All
518	DO	Fo_Stop0Status	Stop 0 status	All
519	DO	Fo_Stop1Status	Stop 1 status	All

Address	Type	Signal name	Description	Station
520	DO	Fo_LocalEmergencyStopStatus	Local Emergency stop status	All
521	DO	Fo_DriveEnable	Drive enable	All
522	DO	Fo_AutomaticMode	Auto mode status	All
523	DO	Fo_ManualFullSpeedMode	Manual full speed mode	All
524	DO	Fo_ManualMode	Manual mode status	All
525	DO	Fo_EnablingDevice	TPU enable device ok	All
526	DO	Fo_ExtEmyStopOK	External Empty stop ok	All
527	DO	Fo_AutoStopOK	Auto stop ok	All
528	DO	Fo_GeneralStopOK	General mode stop ok	All
529	DO	Fo_DelayedStopOK	Delay stop ok	All
530	DO	Fo_TestModeStopOk	Test mode stop ok	All
531	DO	Fo_Zone01PosSupAct	Zone 1 supervision active	All
532	DO	Fo_Zone01PosViolation	Zone 1 violation	All
533	DO	Fo_Zone02PosSupAct	Zone 2 supervision active	All
534	DO	Fo_Zone02PosViolation	Zone 2 violation	All
535	DO	Fo_Zone03PosSupAct	Zone 3 supervision active	All
536	DO	Fo_Zone03PosViolation	Zone 3 violation	All
537	DO	Fo_Zone04PosSupAct	Zone 4 supervision active	All
538	DO	Fo_Zone04PosViolation	Zone 4 violation	All
539	DO	Fo_Zone05PosSupAct	Zone 5 supervision active	All
540	DO	Fo_Zone05PosViolation	Zone 5 violation	All
541	DO	Fo_Zone06PosSupAct	Zone 6 supervision active	All
542	DO	Fo_Zone06PosViolation	Zone 6 violation	All
543	DO	Fo_Zone07PosSupAct	Zone 7 supervision active	All
544	DO	Fo_Zone07PosViolation	Zone 7 violation	All
545	DO	Fo_Zone08PosSupAct	Zone 8 supervision active	All
546	DO	Fo_Zone08PosViolation	Zone 8 violation	All
547	DO	Fo_StandStillSupAct	Standstill supervision active	All
548	DO	Fo_StandStillViolation	Standstill violation	All
549	DO	Fo_NonMotionExecutionEnabled	Non motion mode enabled	All
550	DO	Fo_NonMotionExecutionViolation	Non motion mode violation	All

Safety input signals to robot from PLC:

Address	Type	Signal name	Description	Station
512	DI	Fi_SafetyEnable	Safety enable from PLC	All
513	DI	Fi_SafetySynchronization	Safety synchronization from PLC	All
514	DI	Fi_ExtComShutdownAck	External communication shut down acknowledge	All
515	DI	Fi_EnableRemoteModeChange	Remote mode change enable from PLC	All
516	DI	Fi_PaintSafetyInterlock	Paint safety interlock	All
517	DI	Fi_CabILK	Cabin interlock input	All
518	DI	Fi_HvILK	HV interlock input	All
519	DI	Fi_ProcILK	Process interlock input	All
520	DI	Fi_Sys2ILK	System 2 interlock input	All
521	DI	Fi_TurbineILK	Turbine interlock input	All
522	DI	Fi_MainAirSupp	Main air interlock input	All
523	DI	Fi_ServoDisCon	Servo disconnect input	All
524	DI	Fi_CBSDisCon	CBS servo disconnect input	All
525	DI	Fi_EmyStopReset	Emergency stop reset	All
526	DI	Fi_ExtEmyStop	External emergency stop	All
527	DI	Fi_AutoStop	Auto mode stop	All
528	DI	Fi_GeneralStop	Generic mode stop	All
529	DI	Fi_DelayedStop	Delay mode stop	All
530	DI	Fi_TestModeStop	Test mode stop	All
531	DI	Fi_Zone01PosSupervision	Zone1 supervision	All
532	DI	Fi_Zone02PosSupervision	Zone2 supervision	All
533	DI	Fi_Zone03PosSupervision	Zone3 supervision	All
534	DI	Fi_Zone04PosSupervision	Zone4 supervision	All
535	DI	Fi_Zone05PosSupervision	Zone5 supervision	All
536	DI	Fi_Zone06PosSupervision	Zone6 supervision	All
537	DI	Fi_Zone07PosSupervision	Zone7 supervision	All
538	DI	Fi_Zone08PosSupervision	Zone8 supervision	All
539	DI	Fi_StandStillSupervision	Standstill supervision	All
540	DI	Fi_EnableNonMotionExecution	Enable non motion execution	All

7.2.2.2 Pixel paint I/O signals

Pixel paint I/O signals for paint control is same as '[Paint I/O signals](#)' on page 242.

7.2.2.3 Clean robot I/O signals

Feather duster IO
signals

TBD

Sword brush I/O
Signals

TBD

7.2.2.4 Flaming robot I/O signals

Robot to PLC

IO List

For flaming robot IO signals-

Signal name	Data type	Address	Description
doFlame_On	BOOL	do416	Flame is burning
doFlame_Ready	BOOL	do417	Flame ready
doFlame_Err	BOOL	do418	Flame error
doFlame_En	BOOL	do419	Flame enable
doFlameIgnition_En	BOOL	do420	Flame ignition enable
doFlame_1	BOOL	do421	Flame level 1
doFlame_2	BOOL	do422	Flame level 2
doFlame_3	BOOL	do423	Flame level 3
	BOOL	do424	
	BOOL	do425	
	BOOL	do426	
	BOOL	do427	
	BOOL	do428	
	BOOL	do429	
	BOOL	do430	
	BOOL	do431	

The other IO addresses are the same as those for a common paint robot. However, the IO map from do416 to do431 is specifically designed for the flame robot.

7 Robot system

7.2 Paint control IO Interface

PLC to Robot

IO List

For flaming robot IO signals-

Signal name	Data type	Address	Description
diFlame_Sys_Rel	BOOL	di416	Flame is burning
diFlame_Sys_On	BOOL	di417	Flame system power On
diFlame_Sys_Auto	BOOL	di418	Flame system auto mode
diFlame_Sys_Ok	BOOL	di419	Flame system OK
diFlame_AirFlowErr	BOOL	di420	Flame system air flow
diFlame_GasFlowErr	BOOL	di421	Flame system gas flow
	BOOL	di422	
	BOOL	di423	
	BOOL	di424	
	BOOL	di425	
	BOOL	di426	
	BOOL	di427	
	BOOL	di428	
	BOOL	di429	
	BOOL	di430	
	BOOL	di431	

The IO addresses are the same as those for a common paint robot. However, the IO map from di416 to di431 is specifically designed for the flame robot.

7.2.2.5 Dispense robot I/O signals

TBD

7.3 Paint control command interface

Description

The robot and the station PLC communicate with each other via the Command I/O Protocol. ABB paint documentation '[Reference manual, paint commands and PLC interface IRC5P RW6.0](#)' on page 15" describes the default accessible commands in addition to certain unique ones that are utilized for particular functions.

This chapter contains a list and description of all custom commands.

The following sub-sections are included in this section:

['Paint robot command'](#) on page 258

['Pixel paint command'](#) on page 277

['Clean robot command'](#) on page 282

['Flaming robot command'](#) on page 283

['Dispense robot command'](#) on page 283

7.3.1 Paint robot command

The following sub-sections are included in this section:

['Paintware command'](#) on page 259

['Service position command'](#) on page 260

['Application robot command'](#) on page 261

['Material change command'](#) on page 265

['Customized command'](#) on page 266

7.3.1.1 Paintware command

The following Paintware commands are full command protocols which are frequently used in the robot system. For more information, see the ABB paint documentation '[Reference manual, paint commands and PLC interface IRC5P RW6.0](#)' on page 15".

Command No	Command name	Description
1	Get Master	Client get master of robot
2	Release Master	Client release master of robot
100	Reset system	Reset robot system
101	Reset Error	Reset any process or controller error
102	Stop Execution	Stop the manipulator (not CBS or pump motors)
103	Start/Continue Execution	Resume production / program / material change
104	Cancel Execution	Cancel programs and stops execution (after that, a start command is needed)
105	INHIBIT_PROD	-
106	Mode Change Acknowledge	Acknowledge switching from manual to automatic mode
107	System recovery	Recover the robot system
110	Motors On	Turn the robot motors on
111	Motors Off	Turn the robot motors off
150	Backup	Program backup of robot
151	Restore	Restore the program backup of robot
302	Applicator Enable	Activate painting, AppEnabled = 1
303	Applicator Disable	Deactivate painting, AppEnabled = 0
314	High Voltage Enable	Activate high voltage, HVActive = 1
315	High Voltage Disable	Deactivate high voltage, HVActive = 0
320	Set Brush Factor	Set brush factor to be used for the specified material or global factor.
400	Execute program	Load and execute a program that bypasses the job queue.
413	External start	External start of the paint program.
416	Job queue erase	Erase jobs from queue, cancel program & material change.
419	Get current	-
421	Job queue append	Append a job in the queue.
610	Dock or undock RCC.	Dock Cmd610,1
		Undock Cmd610,2
611	Move RCC to requested position	Cmd611, n n = requested valve no.
612	Calibrate RCC	For PLC Cmd612
		Internal paint command Cmd401,2001 Cmd422,0,2001
613	Enable or disable RCC motor	Enable Cmd613,1
		Disable Cmd613,2

Command No	Command name	Description	
614	Port Cleaning	For PLC	Cmd614,nFrom,nTo
		Internal paint command	Cmd401,2002,nFrom,nToCmd422,0,2002,0,0,0,nFrom,nTo
	Clean current valve	For PLC	Cmd614,1001
		Internal paint command	Cmd401,2002,1001 Cmd422,0,2002,0,0,0,1001
	Clean all valve	For PLC	Cmd614,1002
		Internal paint command	Cmd401,2002,1002 Cmd422,0,2002,0,0,0,1002
	Cancel ongoing operation	For PLC	Cmd614,1003
		Internal paint command	None

A return code is given when the execution of the paint command is completed. This indicates there is an error if it is non-zero, and the robot error log may contain details on the cause of the error.

Return code	Name	Description
0	OK	The command executed successfully
-1	Command failed	Command failed for an unspecified reason
-2	Unknown command	No command defined with this number
-3	Not master	The client did not have the privilege to execute the command
-4	Timeout	The command timed out
-5	Invalid value	A parameter value was outside valid range
-6	Illegal system mode	The command could not be executed in the current system mode
-7	Resource unavailable	Some resource required to execute the command was unavailable, e.g. a signal has not been defined
-8	IPC queue error	Internal error
-9	Semaphore timeout	Internal error, check that CmdTask is running
-10	Buffer full	Internal error, check that CmdTask is running

7.3.1.2 Service position command

Description

Service position command is used to move the robot to a defined service position for cleaning, dressing or maintenance purposes. The command number is 421, with different input parameters. All service position programs are high-priority programs, which means the program can be run with the job queue paused. The position can be selected from HMI. The command can only be used in the robot in automatic mode and the station in manual mode.

Note: All commands, except the home command, need the robot to start in the home position.

Command No	Program number	Description
421	10001	Home position
421	10002	Material change position
421	10003	Calibration position
421	10004	Robot Maintain position
421	10005	Bypass position
421	10006	Manual clean position
421	10007	Dress position
421	10008	Reference position
421	10009	Applicator maintenance position
421	10010	Spare
421	10011	User position 1
421	10012	User position 2
421	10013	User position 3
421	10014	User position 4
421	10015	User position 5
421	10016	Spare
421	10017	Spare
421	10018	Spare
421	10019	Spare
421	10020	Spare

7.3.1.3 Application robot command

Description

The application function command is used to order the robot to run special maintenance work, such as bell cleaning, hub cleaning, fluid volume calibration, and so on. The command can be selected and sent using HMI software. The command is only applicable to robots in automatic mode and stations in manual mode.

Note: All commands need to be sure the robot starts at the home position.

Command No	Program number	Description
421	10021	Gun outside clean
421	10022	Atomizer outside clean
421	10023	Hub clean
421	10024	Opener tool clean
421	10025	Fluid flow test
421	10026	Dynamic brush test
421	10027	Static brush test
421	10028	Hub clean flow test
421	10029	HV alarm test
421	10030	HV shutdown test

Command No	Program number	Description
421	10031	Spare
421	10032	Perform a brake check operation
421	10033	Reserved
421	10034	Reserved
421	10035	Reserved

Description of application command:**1. Fluid flow test**

Purpose This command is used to test fluid volume for the gear pump.

Preconditions

- Station : Manual mode
- Robot : Auto mode

Parameters

Command	421
In param 1	10025
In param 2	0
In param 3	Flow volume
In param 4	Flow on time
In param 5	0
In param 6	0
In param 7	0
In param 8	0
In param 9	0
In param 10	0

7 Robot system

7.3 Paint control command interface

2. Dynamic brush test

Purpose This command is used to test the brush dynamic pattern.

Preconditions

- Station : Manual mode
- Robot : Auto mode

Parameters

Command	421
In param 1	10026
In param 2	0
In param 3	Path number
In param 4	Path offset
In param 5	0
In param 6	0
In param 7	0
In param 8	0
In param 9	0
In param 10	0

3. Static brush test

Purpose This command is used to test the brush static pattern.

Preconditions

- Station : Manual mode
- Robot : Auto mode

Parameters

Command	421
In param 1	10027
In param 2	0
In param 3	Path number
In param 4	Path offset
In param 5	0
In param 6	0
In param 7	0
In param 8	0
In param 9	0
In param 10	0

4. Hub clean flow test

Purpose This command is used to test the hub clean flow.

Preconditions

- Station : Manual mode
- Robot : Auto mode

Parameters

Command	421
In param 1	10028
In param 2	0
In param 3	0
In param 4	0
In param 5	0
In param 6	0
In param 7	0
In param 8	0
In param 9	0
In param 10	0

5. HV alarm test

Purpose This command is used to test the HV alarm work status.

Preconditions

- Station : Manual mode
- Robot : Auto mode

Parameters

Command	421
In param 1	10029
In param 2	0
In param 3	0
In param 4	0
In param 5	0
In param 6	0
In param 7	0
In param 8	0
In param 9	0
In param 10	0

7 Robot system

7.3 Paint control command interface

6. HV shutdown test

Purpose This command is used to test the HV voltage release time.

Preconditions

- Station : Manual mode
- Robot : Auto mode

Parameters

Command	421
In param 1	10030
In param 2	0
In param 3	0
In param 4	0
In param 5	0
In param 6	0
In param 7	0
In param 8	0
In param 9	0
In param 10	0

7.3.1.4 Material change command

Material change command can be used to run cleaning, filling and special colour change programs. The command can be sent from PLC or HMI. The command from HMI can only be used in the robots in automatic mode and stations in manual mode.

Note: All commands need to be sure the robot starts at the home position.

Command No	Material number	Material Option1	Material Option2	Description
421	Colour number	0	0	Filling colour according to colour number.
421	Colour number	5<X<20	0	Filling material supply (VW).
421	Colour number	X>20	0	Cleaning material supply (VW).
421	999	0	0	Cleaning the line.
421	999	0	1	Cleaning cavity 1.
421	999	0	2	Cleaning cavity 2
421	999	0	3	Cleaning cavity 3.
421	999	1	0	Force cleaning the line.
421	999	2	0	Maintenance function, filling air.
421	999	3	0	Maintenance function, filling solvent.

7.3.1.5 Customized command

Commands List

Command No	Command name	Description
1020	Robot backup	The command is used to save robot backup.
1030	Paint push enable	The command is used to enable paint push function.
1031	Paint push disable	The command is used to disable paint push function.
1040	Pump shaft supervision	The command is used to enable pump shape supervision function.
1041	Pump shaft supervision	The command is used to disable pump shape supervision function.
1050	Enable leakage check	The command is used to enable leakage check function.
1051	Disable leakage check	The command is used to disable leakage check function.
1060	Get brush parameter	The command is used to get brush parameters.
1061	Set brush parameter	The command is used to save them into variable from HMI.
1070	Fluid enable	The command is used to enable fluid device.
1071	Fluid disable	The command is used to disable fluid device.
1080	Calculate volume	The command is used to calculate the paint accumulate volume.
1090	Set Painter interior bypass	The command is used to set painter interior logic to bypass.
1091	Rest Painter interior bypass	The command is used to reset painter interior logic to bypass.

7 Robot system

7.3 Paint control command interface

Description of customized command

1. Start backup - Cmd1020

Purpose	This command is used to start a backup of the current system.
Preconditions	When the Robot's "Automatic mode" and "JobInProgress" signals are present at the same time, the backup will be disabled.

In Parameters

Command	1020
In param 1	0
In param 2	0
In param 3	0
In param 4	0
In param 5	0
In param 6	0
In param 7	0
In param 8	0
In param 9	0
In param 10	0

Out Parameters

dosignal	"sdoStartBackup"
----------	------------------

2. Paint push - Cmd1030

Purpose	This command is used to enable the paint push function.
Preconditions	None

In Parameters

Command	1030
In param 1	0
In param 2	0
In param 3	0
In param 4	0
In param 5	0
In param 6	0
In param 7	0

Command	1030
In param 8	0
In param 9	0
In param 10	0

Out Parameters None

Errors

Command failed	-
----------------	---

3. Paint push - Cmd1031

Purpose This command is used to disable the paint push function.

Preconditions None

In Parameters

Command	1031
In param 1	0
In param 2	0
In param 3	0
In param 4	0
In param 5	0
In param 6	0
In param 7	0
In param 8	0
In param 9	0
In param 10	0

Out Parameters None

Errors

Command failed	-
----------------	---

7 Robot system

7.3 Paint control command interface

4. Paint shaft supervise- Cmd1040

Purpose This command is used to enable the paint shaft supervise function.

Preconditions

- Station : Manual mode
- Robot : Auto mode

In Parameters

Command	1040
In param 1	1 = Enable pump (pump1 – for 2K) shaft supervise 2 = Enable pump2 shaft supervise 3 = Enable pump3 (for 2K) shaft supervise 4 = Enable pump4 (for 2K) shaft supervise
In param 2	0
In param 3	0
In param 4	0
In param 5	0
In param 6	0
In param 7	0
In param 8	0
In param 9	0
In param 10	0

Out Parameters None

Errors

Command failed

-

5. Paint shaft supervise- Cmd1041

Purpose This command is used to enable and disable the paint shaft supervise function. The enable or disable is set by the input parameter.

Preconditions

- Station : Manual mode
- Robot : Auto mode

In Parameters

Command	1041
In param 1	1 = Enable pump (pump1 – for 2K) shaft supervise 2 = Enable pump2 shaft supervise 3 = Enable pump3 (for 2K) shaft supervise 4 = Enable pump4 (for 2K) shaft supervise
In param 2	0
In param 3	0
In param 4	0
In param 5	0
In param 6	0
In param 7	0
In param 8	0
In param 9	0
In param 10	0

Out Parameters

None

Errors

Command failed	-
----------------	---

6. Enable leakage check- Cmd1050

Purpose

This command is used to enable the paint leakage check function.

Preconditions

None

In Parameters

Command	1050
In param 1	0
In param 2	0
In param 3	0
In param 4	0
In param 5	0
In param 6	0
In param 7	0
In param 8	0
In param 9	0
In param 10	0

7 Robot system

7.3 Paint control command interface

Out Parameters None

Errors

Command failed	–
----------------	---

7. Disable leakage check- Cmd1051

Purpose This command is used to disable the paint leakage check function.

Preconditions None

In Parameters

Command	1051
---------	------

In param 1	0
------------	---

In param 2	0
------------	---

In param 3	0
------------	---

In param 4	0
------------	---

In param 5	0
------------	---

In param 6	0
------------	---

In param 7	0
------------	---

In param 8	0
------------	---

In param 9	0
------------	---

In param 10	0
-------------	---

Out Parameters None

Errors

Command failed	–
----------------	---

8. Get brush factor - Cmd1060

Purpose This command is used to get the brush factor. By using this command, the HMI can get the current brush offset value.

Preconditions None

In Parameters

Command	1060
In param 1	0
In param 2	0
In param 3	0
In param 4	0
In param 5	0
In param 6	0
In param 7	0
In param 8	0
In param 9	0
In param 10	0

Out Parameters

nBrushOffset1	Fluid offset
nBrushOffset2	Atom offset
nBrushOffset3	Shape1 offset
nBrushOffset4	Shape2 offset
nBrushOffset5	HV offset

Errors

Failed	Error IPS parameter
--------	---------------------

9. Set brush factor - Cmd1061

Purpose

This command is used to set the brush factor. By using this command, the HMI can set the current brush offset value.

Preconditions

None

In Parameters

Command	1061
In param 1	0
In param 2	0
In param 3	0
In param 4	0
In param 5	0
In param 6	0
In param 7	0

7 Robot system

7.3 Paint control command interface

Command	1061
In param 8	0
In param 9	0
In param 10	0

Out Parameters

nBrushOffset1	Fluid offset
nBrushOffset2	Atom offset
nBrushOffset3	Shape1 offset
nBrushOffset4	Shape2 offset
nBrushOffset5	HV offset

Errors

Failed	Error IPS parameter
--------	---------------------

10. Fluid Enable - Cmd1070

Purpose

This command is used to enable and disable the fluid gear pump.

Preconditions

None

In Parameters

Command	1070
In param 1	0
In param 2	0
In param 3	0
In param 4	0
In param 5	0
In param 6	0
In param 7	0
In param 8	0
In param 9	0
In param 10	0

Out Parameters

None	–
------	---

Errors

Command failed	–
----------------	---

11. Fluid Disable - Cmd1071

Purpose This command is used to disable the fluid gear pump.

Preconditions None

In Parameters

Command	1071
In param 1	0
In param 2	0
In param 3	0
In param 4	0
In param 5	0
In param 6	0
In param 7	0
In param 8	0
In param 9	0
In param 10	0

Out Parameters

None	–
------	---

Errors

Command failed	–
----------------	---

12. Paint volume calculation - Cmd1080

Purpose This command is used to calculate the paint volume for specific job data and send it back to PLC.

Preconditions None

7 Robot system

7.3 Paint control command interface

In Parameters

Command	1080
In param 1	Program index
In param 2	Material Index
In param 3	Option1
In param 4	Option2
In param 5	Option3
In param 6	0
In param 7	0
In param 8	0
In param 9	0
In param 10	0

Out Parameters

The calculated volume

Errors

Command failed	–
----------------	---

13. Set interior painter bypass state - Cmd1090

Purpose

This command is used to set the interior paint to bypass state.

Preconditions

None

In Parameters

Command	1090
In param 1	0
In param 2	0
In param 3	0
In param 4	0
In param 5	0
In param 6	0
In param 7	0
In param 8	0
In param 9	0
In param 10	0

Out Parameters

None

Errors

Command failed	–
----------------	---

14. Reset interior painter bypass state - Cmd1091

Purpose

This command is used to reset the interior paint to bypass the state.

Preconditions

None

In Parameters

Command**1091**

In param 1

0

In param 2

0

In param 3

0

In param 4

0

In param 5

0

In param 6

0

In param 7

0

In param 8

0

In param 9

0

In param 10

0

Out Parameters

None

Errors

Command failed	–
----------------	---

7.3.2 Pixel paint command

The following sub-sections are included in this section:

'Cavity transition state' on page 278

'Go Ready State' command' on page 278

'Clean head (Day clean)' on page 280

'MC suspend handling' on page 281

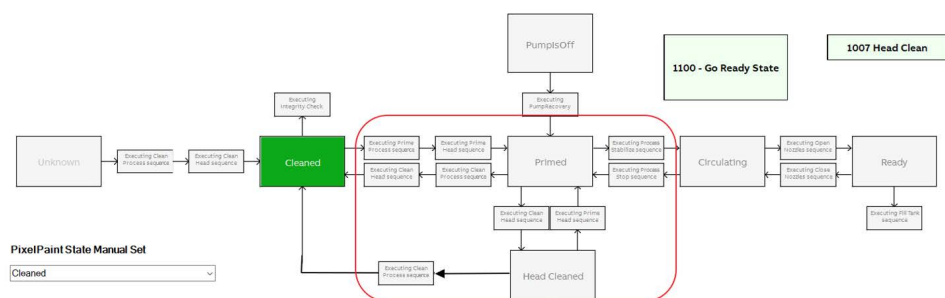
These PixelPaint commands are part of the robot command protocol. The material indexes must be ordered as regular program selection to perform operations to move from one logical state to another. The command with the specific index can be ordered from any client that is allowed to add a job. The PixelPaint logic prevents illegal operations.

Example:

It is not allowed to execute the circulation sequence in a cleaned or unknown state. The system will respond with an error message indicating an illegal material operation.

It is possible to force a command to execute to bypass the rules in the logic by setting the force flag to "1" when ordering the command to execute. For information on how to set this force flag, see 'Reference manual, Paint commands and PLC interface RW6', System description / Python-based material change / Material option codes.

PixelPaint Commands and State Machine



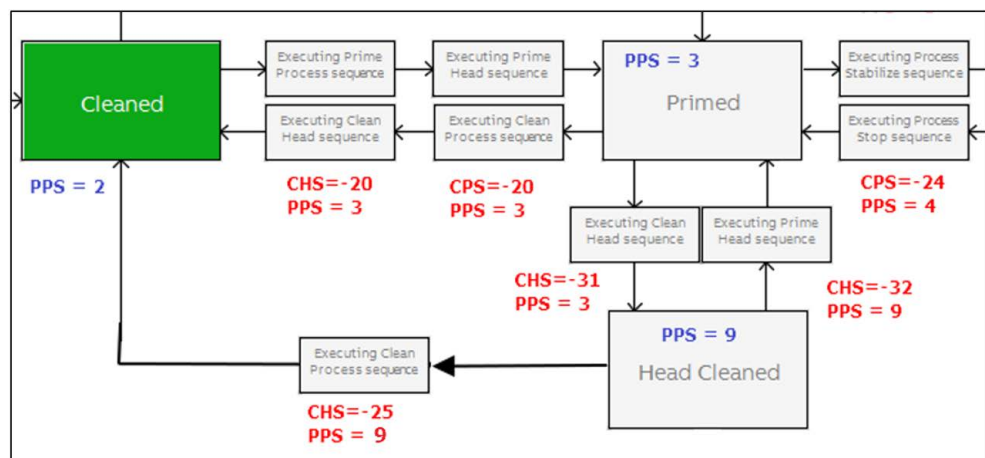
Index	Description of action
1001	Clean process cavity and head
1002	Prime process cavity and head
1003	Stabilize circulation (without head nozzles)
1004	Opening valves to circulate paint through print head
1005	Closing valves to circulate through bypass print head
1006	Stop paint circulation
1007	Clean print head
1008	Prime print head
1009	Recover from a pump stop (release pressure and recover)

Index	Description of action
1010	Clean filter cavity
1011	Integrity check sequence when required
1099	Reset PixelPaint state to Unknown
1100	Go Ready State from any state.

7.3.2.1 Cavity transition state

Description

In Pixelpaint, the cavity transition state refers to a visual effect or design element that represents a shift or change within a game environment. This transition often involves a pixelated or abstract depiction of a cavity or void, signaling a transformation.



Primed (PPS=3) ↔ HeadClean (PPS=9) :

Clean Head: nCavityHeadIndex = -31; nPixelPaintState=3

Prime Head: nCavityHeadIndex = -32; nPixelPaintState=9

Head Cleaned (PPS=9) =>Cleaned:

Clean Filter: nCavityProcessIndex = -25; nPixelPaintState=9

7.3.2.2 “Go Ready State” command

Description

The "Go Ready" state is a visual cue indicating that a system is prepared for the next action or phase. This state typically involves a clear, attention-grabbing design or animation, signaling that all necessary conditions are met and that can proceed with the next step.

Make the system automatically transition to “Ready State” internally, step by step.

```
! Commands:
local const num CMD_CLEAN:=1;
local const num CMD_PRIME:=2;
local const num CMD_STABILIZE_CIRCULATION:=3;
local const num CMD_OPEN_NOZZLES:=4;
local const num CMD_CLOSE_NOZZLES:=5;
local const num CMD_STOP_CIRCULATION:=6;
local const num CMD_CLEAN_HEAD:=7;
local const num CMD_PRIME_HEAD:=8;
local const num CMD_ACK_RECOVERY:=9;
local const num CMD_CLEAN_FILTER:=10;
local const num CMD_INTEGRITY_CHECK:=11;
local const num CMD_RESET_STATE:=99;
local const num CMD_READY_STATE:=100;
```

```
case CMD_READY_STATE:
  while nPixelPaintState<>PPS_READY do
    test nPixelPaintState
    CASE PPS_UNKNOWN:
      UnknownToClean;
    CASE PPS_CLEAN:
      CleanToPrimed;
    CASE PPS PRIMED:
      StabilizeCirculation;
    CASE PPS_CIRCULATING:
      OpenNozzles;
    CASE PPS_PP_FC_HC:
      WeekCleanToPrimed;
    CASE PPS_FC_HP:
      DayCleanToPrimed;
    CASE PPS_RECOVERY:
      Recovery;
    CASE PPS_HEAD_CLEAN:
      HeadCleanToPrimed;
    ENDTEST
  endwhile
```

7.3.2.3 Clean head (Day clean)

From-To-Status	1.PPS_Unknown	2.PPS_Clean	3.PPS_Primed	4.PPS_Circulating	5.PPS_Ready	9.PPS_Head_Clean
1.PPS_Unknown	X	1001	1002	1003	1100	X
2.PPS_Clean	X	X	1002	1003	1100	X
3.PPS_Primed	X	1001	X	1003	1100	1007
4.PPS_Circulating	X	1001	1006	X	1100	1007
5.PPS_Ready	X	1001	1006	1005	X	1007
9.PPS_Head_Clean	X	1001	1002	1003	1100	X

For example: Want operate from PPS_Unknown to PPS_Ready, send Command 421,0,1100.

Example:

Go to Clean_Head.

Can from ready, circulating, Primed status.

```

case CMD_CLEAN_HEAD:
    !Check if the nPixelPaintState still in Ready or Circulating, otherwise opeate to PPS_Primed state
    IF (nPixelPaintState=PPS_CIRCULATING or nPixelPaintState=PPS_READY) THEN
        WHILE nPixelPaintState<>PPS_PRIMED DO
            TEST nPixelPaintState
            CASE PPS_READY:
                CloseNozzles;
            CASE PPS_CIRCULATING:
                StopCirculation;
            endtest
        endwhile
    endif

    if (nPixelPaintState=PPS_FC_HP) or bForce then
        DayCleanWeekendClean;
    elseif (nPixelPaintState=PPS_PRIMED) then
        PrimedToHeadClean;
    elseif (nPixelPaintState=PPS_PP_FC_HC) then
        PntElog 4765,PNT_ELOG_ERROR\Arg1:=stCommandName(nOperationCmd);
    else
        PntElog 4762,PNT_ELOG_ERROR\Arg1:=stCommandName(nOperationCmd);
    endif

```

7 Robot system

7.3 Paint control command interface

```
case CMD_PRIME:
    if (nPixelPaintState=PPS_UNKNOWN) or bForce then
        UnknownToClean;
    endif
    if (nPixelPaintState=PPS_FC_HP) or bForce then
        DayCleanToPrimed;
    elseif (nPixelPaintState=PPS_PP_FC_HC) or bForce then
        WeekCleanToPrimed;
    elseif (nPixelPaintState=PPS_CLEAN) or bForce then
        CleanToPrimed;
    elseif (nPixelPaintState=PPS_HEAD_CLEAN) or bForce then
        HeadCleanToPrimed;
    elseif (nPixelPaintState=PPS_PRIMED) then
        PntElog 4765,PNT_ELOG_ERROR\Arg1:=stCommandName(nOperationCmd);
    else
        PntElog 4762,PNT_ELOG_ERROR\Arg1:=stCommandName(nOperationCmd);
    endif
```

Go to Primed_Status

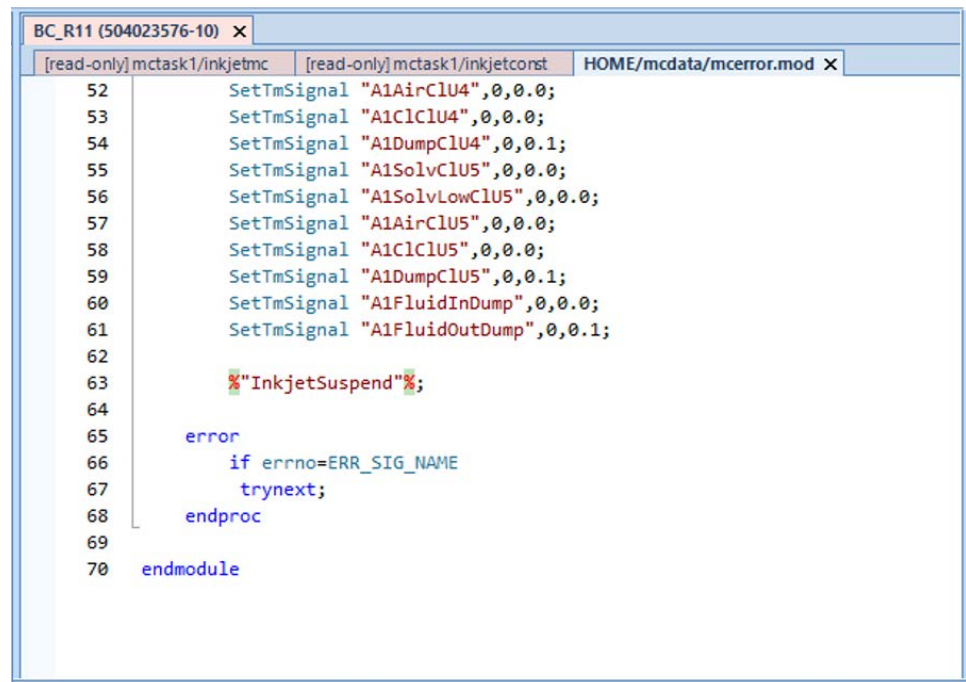
It can change status from ready, circulating, head clean, system clean status.

```
case CMD_STOP_CIRCULATION:
    !Check if the nPixelPaintState in Ready state, other with CloseNozzles first.
    IF nPixelPaintState=PPS_READY THEN
        CloseNozzles;
    endif
    if (nPixelPaintState=PPS_CIRCULATING) or bForce then
        StopCirculation;
    elseif (nPixelPaintState=PPS_PRIMED) then
        PntElog 4765,PNT_ELOG_ERROR\Arg1:=stCommandName(nOperationCmd);
    else
        PntElog 4762,PNT_ELOG_ERROR\Arg1:=stCommandName(nOperationCmd);
    endif
```

7.3.2.4 MC suspend handling

Description

"MC Suspend Handling" refers to the mechanism for managing and responding to the suspension or pausing of a main character (MC) or components. This system ensures that when it is paused or the main character is temporarily inactive, all relevant processes, such as elements, are properly managed or halted. It aims to maintain stability and provide a seamless experience when resuming from suspension.

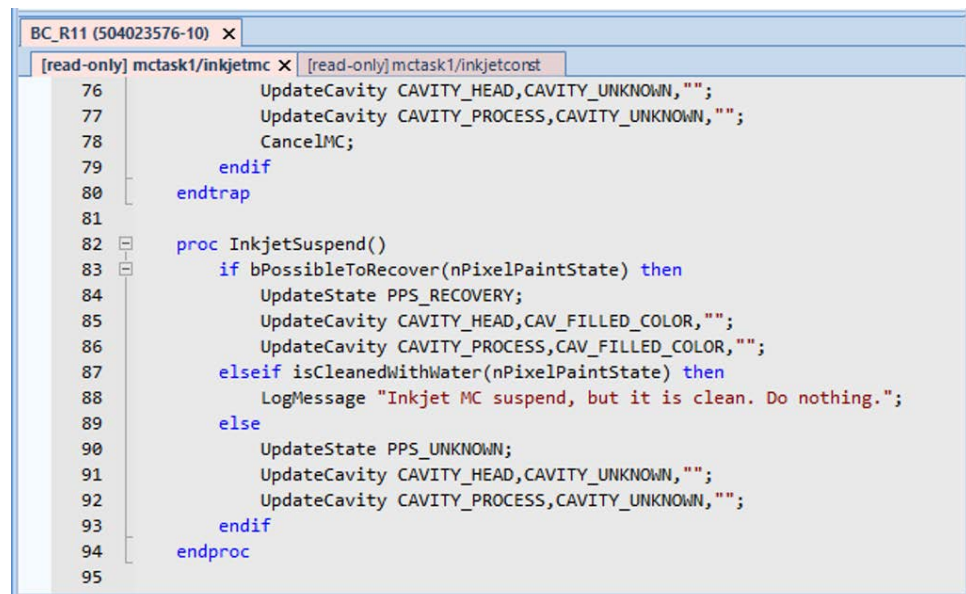


```

BC_R11 (504023576-10) X
[read-only] mctask1/inkjetmc [read-only] mctask1/inkjetconst HOME/mcdata/mcerror.mod X
52 SetTmSignal "A1AirClU4",0,0.0;
53 SetTmSignal "A1ClClU4",0,0.0;
54 SetTmSignal "A1DumpClU4",0,0.1;
55 SetTmSignal "A1SolvClU5",0,0.0;
56 SetTmSignal "A1SolvLowClU5",0,0.0;
57 SetTmSignal "A1AirClU5",0,0.0;
58 SetTmSignal "A1ClClU5",0,0.0;
59 SetTmSignal "A1DumpClU5",0,0.1;
60 SetTmSignal "A1FluidInDump",0,0.0;
61 SetTmSignal "A1FluidOutDump",0,0.1;
62
63 %"InkjetSuspend"%;
64
65 error
66     if errno=ERR_SIG_NAME
67         trynext;
68     endproc
69
70 endmodule

```

MC error handling, to close valves/regulator, and Update state/cavity.



```

BC_R11 (504023576-10) X
[read-only] mctask1/inkjetmc X [read-only] mctask1/inkjetconst
76 UpdateCavity CAVITY_HEAD,CAVITY_UNKNOWN,"";
77 UpdateCavity CAVITY_PROCESS,CAVITY_UNKNOWN,"";
78 CancelMC;
79 endif
80 endtrap
81
82 proc InkjetSuspend()
83     if bPossibleToRecover(nPixelPaintState) then
84         UpdateState PPS_RECOVERY;
85         UpdateCavity CAVITY_HEAD,CAV_FILLED_COLOR,"";
86         UpdateCavity CAVITY_PROCESS,CAV_FILLED_COLOR,"";
87     elseif isCleanedWithWater(nPixelPaintState) then
88         LogMessage "Inkjet MC suspend, but it is clean. Do nothing.";
89     else
90         UpdateState PPS_UNKNOWN;
91         UpdateCavity CAVITY_HEAD,CAVITY_UNKNOWN,"";
92         UpdateCavity CAVITY_PROCESS,CAVITY_UNKNOWN,"";
93     endif
94 endproc
95

```

7.3.3 Clean robot command

TBD

7 Robot system

7.3 Paint control command interface

7.3.4 Flaming robot command

The following flaming robot commands are full command protocols which are frequently used in the robot system. For more information, see the ABB paint documentation '[Reference manual, paint commands and PLC interface IRC5P RW6.0](#)' on page 15".

Command No	Command name	Description
1	Get Master	Client get master of robot
2	Release Master	Client release master of robot
101	Reset Error	Reset any process or controller error
102	Stop Execution	Stop the manipulator (not CBS or pump motors)
103	Start/Continue Execution	Resume production / program / material change
104	Cancel Execution	Cancel programs and stops execution (after that, a start command is needed)
106	Automode Acknowledge	Acknowledge switching from manual to automatic mode
110	Motors On	Turn the robot motors On
111	Motors Off	Turn the robot motors Off
302	Applicator Enable	Activate painting, AppEnabled = 1
303	Applicator Disable	Deactivate painting, AppEnabled = 0
406	Inhibit Production	Stop job queue at the end of current job.
413	External start	Start the robot paint program when robots are ready to paint.
416	Job queue erase	Erase jobs from queue, cancel program & material change.
421	Job queue append	Append a job in the queue.

7.3.5 Dispense robot command

TBD

7.4 Paint instructions

The following sub-sections are included in this section:

'File structure' on page 284

'Program structure' on page 296

'User data' on page 300

7.4.1 File structure

The following sub-sections are included in this section:

'Paint robot file structure' on page 284

'Pixel paint file structure' on page 290

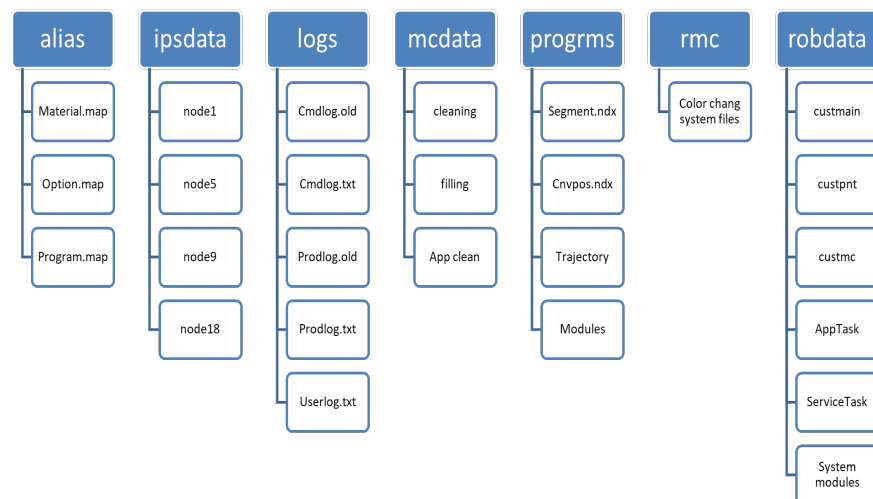
'Clean robot file structure' on page 291

'Flaming robot file structure' on page 291

'Dispense robot file structure' on page 295

7.4.1.1 Paint robot file structure

Description	The robot system program and trajectory program are saved in the robot file system. This chapter will give a description of the robot file structure.
Home folder	The home folder is where the main files are located. The home structure is as below:



Home folder & files:

Folders	Function
alias	• Program alias • Material alias • Option alias
ipsdata	• IPS configuration file • Process index file • Calibration file
logs	• Production log • Command log • User log
mcddata	• Colour change files • Application files • Valve index files
Programs	• Trajectory files • Position and application programs • Program index • Conveyor position index
rmc	• Colour change system
robdata	• Custmain • Custpnt • Custmc
SafeCfgTemplate	• Safety configuration template file
PxTPU.config.xml	• TPU command file

Alias Folder

This folder contains the alias related to the name of the programs, materials and options.

Alias file	Description
Program.map	Programs alias
Option.map	Options alias
Material.map	Materials alias

Logs folder

The logs folder contains the logs of the robot command log, conveyor log, production log and user logs.

Logs File	Description
Cmdlog.old	The old logs of the commands which the robot have received.
Cmdlog.txt	The last logs of the commands which the robot have received.
Cnv.log	The time information when and where the robot get the synchronous signal and when and where it offload the synchronous signal, it can help us to diagnose if the job queue problems happens.
Prodlog.old	The old logs of the productions information which the robot have executed, like programs number, materials number, job options, brush number, fluid consumptions, cycle time and so on.
Prodlog.txt	The last logs of the productions information which the robot have executed, like programs number, materials number, job options, brush number, fluid consumptions, cycle time and so on.
User.log	The log is for customers to export the information needed.

Logs File	Description
Hvlog.log	The log of the auto HV test contains the value of the test and reference and the result of the test.

Mcddata folder

The mcddata folder is where the main material change sequence, it contains the cleaning, filling, hub clean, bell clean modules, and the index, which is necessary for colour change.

File name	Description
Cxgxcln.mod	Cavity x grade x cleaning sequence (x indicates 1,2,3...)
Cxgxfill.mod	Cavity x grade x filling sequence (x indicates 1,2,3...)
Cxfill.ndx	Cavity x filling grade index (x indicates 1,2,3...)
Cxclean.ndx	Cavity x cleaning grade index (x indicates 1,2,3...)
Cxvalve.ndx	Cavity x valve mapping index (x indicates 1,2,3...)
CxfillA.mod	Cavity x filling air sequence (x indicates 1,2,3...)
CxfillS.mod	Cavity x filling solvent sequence (x indicates 1,2,3...)
CxfillMS.mod	Cavity x filling material supply (x indicates 1,2,3...)
CxclnMS.mod	Cavity x cleaning material supply (x indicates 1,2,3...)
Bellcln.mod	Atomizer outside clean sequence program
Bellcln.ndx	Atomizer outside clean index file
Hubcln.mod	Hub clean sequence program
Hubcln.ndx	Hub clean index file
Mcerror.mod	Colour change error sequence
potlerr.mod	Potlife error handling sequence
potlwarn.mod	Potlife warning handling sequence
dumpcln.mod	Dump clean sequence
PushFactor.ndx	Paint push factor index file
Pushmethod.ndx	Paint push method file
psreg.ndx	PCV control regulator index file

Programs folder

The following picture shows the main service programs module and its index numbers, user can set the program number to run the program.

Program No.	File name	Description
10001	yHomePos	Home position module
10002	yMtrChPos	Material position module
10003	yCalPos	Calibration position module
10004	yMaintPos	Maintenance position module
10005	yByPassPos	Bypass position module
10006	yManCleanPos	Manual clean position module
10007	yDressPos	Dress position module
10008	yTCPCheckPos	TCP check position module
10009	yAppMaintPos	Applicator maintenance module
10010	Sapre	Spare
10011	yUserPos1	User position 1

7 Robot system

7.4 Paint instructions

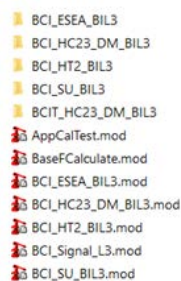
Program No.	File name	Description
10012	yUserPos2	User position 2
10013	yUserPos3	User position 3
10014	yUserPos4	User position 4
10015	yUserPos5	User position 5
10016-10020	Spare	Spare
10021	GunClean	Gun clean module
10022	BellClean	Atomizer clean module
10023	HubClean	Hub clean module
10024	ToolClean	Tool clean module
10025	yFluidFlowTest	Fluid flow volume test module
10026	yDynBrushTest	Dynamic brush test module
10027	yStatBrushTest	Static brush test module
10028	yHubCleanFlowTest	Hub clean flow test module
10029	yHvDynamicTest	High voltage error test module
10030	yHvStaticTest	High voltage shut down test module
10031	Spare	Spare
10032	yCyclicBrakeCheck	Cycle brake check module
10033	Spare	Spare
10034	Spare	Spare
10035	AppCalTest	Applicator calibration module

File name	Description
Cnvpos.ndx	Conveyor start position
ValveCheck.ndx	The opening time of the valves is too long
Segment.ndx	Program number mapping index file

Trajectory folder

The trajectory folder is where the routine modules are located. To distinguish each part of the routine, the module is usually named after the painting part of the object.

The following image or picture shows the common abbreviation and corresponding whole name of the car body. All the parts will composite the complete painting trajectory.



BCI_ESEA_BIL3
BCI_HC23_DM_BIL3
BCI_HT2_BIL3
BCI_SU_BIL3
BCI_T_HC23_DM_BIL3
AppCalTest.mod
BaseFCalculate.mod
BCI_ESEA_BIL3.mod
BCI_HC23_DM_BIL3.mod
BCI_HT2_BIL3.mod
BCI_Signal_L3.mod
BCI_SU_BIL3.mod

For example, the above image, “BCI_ESEA_BIL3”, is the trajectory module for the car body. The main module name is “BCI_ESEA_BIL3.mod”.

Folder path: Controller-Home-Programs.

Robdata folder

The system modules and logic functions are saved in the “robdata” folder.

Location	Module name	Intro
AppTask	AppLoop.sys	Application task main loop file
	AppFunc.sys	Application task function file
	AppPreventMaint.sys	Application task prevent maintenance file
	ValveCheck.sys	Valve open check system file
ServiceTask	ServiceLoop.sys	Service task main loop file
	Qnext.sys	Read next job information
	CnvService.sys	Conveyor ghost, conveyor halt function
	ShapeMaint.sys	Shape air for maintenance function
Robdata	Robdata.sys	Tool, speed, zone and work object data
	Sysdata.sys	System positions and world zones connected to IO signals
	Custmain.sys	User module for custom cycle event procedures
	Custcmd.sys	User module for custom application commands
	Custvar.sys	User modules for custom to define the variables which will be used
	CheckSystem.sys	Check the system license
	Cnvdata.sys	Conveyor user module
	Csvparse.sys	CSV parse module
	Ctsync.sys	Conveyor sync module
	Custconfig.sys	Customized configuration file
	Custdata.sys	Global user data and functions
	Errmessage.sys	Error message handling
	Userlog.sys	User log system file
Custmain	Appinital.sys	Check and set applicator enable signal before painting
	appclean.sys	Applicator clean handling logic
	ldparse.sys	Parse the robot name, station name and robot type
	CheckJobmode.sys	Check the PLC interface get job module
	Infotopl.sys	Export the information communicate with PLC
	Interlock.sys	The interlock logic
	MixerRatioCheck.sys	Check the mixer ratio
	PrepostProgram.sys	Function for pre-program and post-program in trajectory module
	OverrideBrush.sys	Override the system default brush mapping
	Processflowstatus.sys	Export the process of the current job queue
	workflowStatus.sys	Set the function for workflow status
	Programload.sys	The function to loading sub trajectory modules.
	Waitconveyorpos.sys	For trajectory to update, refresh, check the conveyor waiting position
	ServiceControl.sys	Function to allow push button to release robot moving to home position
	Softservo.sys	Enable tuneservo for paint robot
	Wzoneuser.sys	Robot work zones and safety box

7 Robot system

7.4 Paint instructions

Location	Module name	Intro
custmc	Appalarm.sys	Applicator error handling during colour change process
	AutoMixerFlush.sys	Automatic clean mixer with fixed interval
	Mtrloption.sys	Special clean sequence logic
	Starmc.sys	Set global brush and start material change
	SovlSup.sys	Solvent pressure supervision
	Mixratio.sys	Set mix ratio for new program
	Waitmtrlpos.sys	Wait robot move to material change position to execute material change sequence
Location	Module name	Intro
custpnt	Cmdhooks.sys	Events hook with the command
	Hvsolventinterlock.sys	Error handling for high voltage and solvent interlock.
	JobPassControl.sys	Function to check conveyor sync signal and log the signals
	Reseterr.sys	Event trip for general event and error reset function
guisock	Guisock.sys	Socket interface for HMI
	Guisocks.sys	Socket variable for HMI

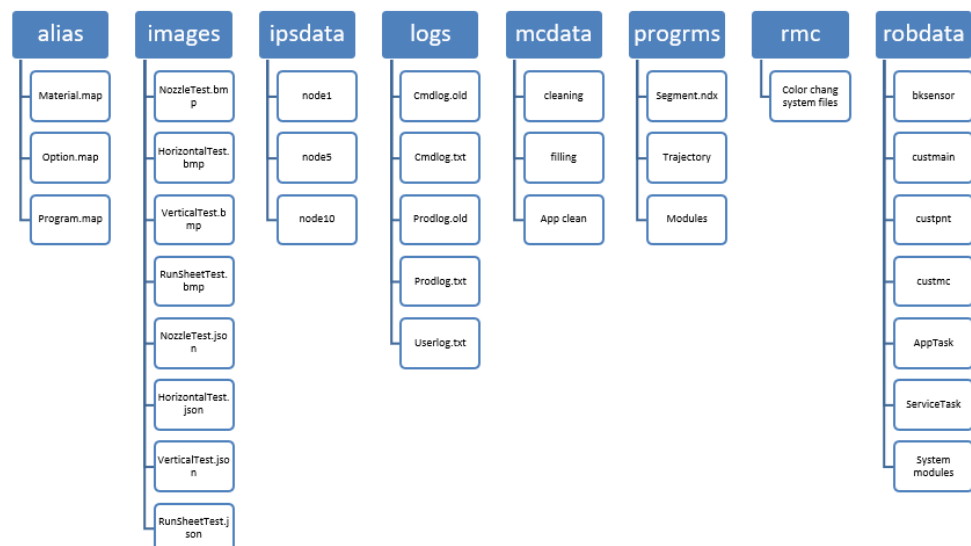
IPS data folder

The IPS folder is a place to save IPS configuration files. There are normally sub folders named “node1”, “node5”, “node9”, and “node18”. Every node corresponds to an IPS device: node 1 corresponds with the PIB board, node 5 corresponds with the MCB board, node 9 corresponds with ACU board, node 18 corresponding with HVC board, the node1 is the most important folder, most modules located at this folder, the following table shows some necessary modules for the node1 configuration.

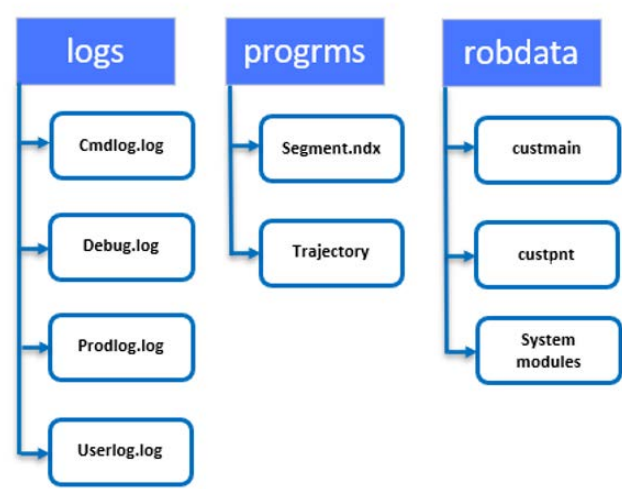
Location	Module name	Intro
A1Brush	Global.bt	Global brush
	tableXXX.bt	Trajectory brush table
calib	A1Atom.c00	The calibration value of Atom
	A1Atom.log	The calibration log of Atom
	A1shape1.c00	The calibration value of shape1
	A1Shape1.log	The calibration log of shape1
	A1shape2.c00	The calibration value of shape2
	A1Shape2.log	The calibration log of shape2
a_physical	A1AirFS	Shape1, shape2 feedback sensor configuration
	A1AtomRS	A1 Atom real speed Curve
	A1Valve.cfg	Define the valve mac address
	A1Valvecol.cfg	Define the colour valve mac address
	A1Valvecat.cfg	Define the cat valve mac address
	A1ValvesHVSolvInt.cfg	The interlock between solvent valves with HV

Location	Module name	Intro
b_devices	A1AtomRoboBell951_1000.cfg	The atom regulation configuration
	A1FluidRed6cc.cfg	Fluid configuration
	A1Shape12RoboBellACU.cfg	The shape regulation configuration
	A1FluidPCV.cfg	Fluid PCV curve configuration
c_applicator	A1AtomRegEn.cfg	Atom regulator Enable configuration
	A1ShapeRegEn.cfg	Shape regulator Enable configuration
	A1Shaftpurge.cfg	Shaft purge Enable configuration
d_brush	A1Brush.cfg	Brush table configuration
e_custom	A1BellCupChk.cfg	Check the bell cup missing
Index	A1shape1Tab.ndx	For ACU regulate output pressure of the shape1 tune
	A1shape2Tab.ndx	For ACU regulate output pressure of the shape2 tune
	A1CpsTab.ndx	For I-Drive regulate the angel of the colour index of RCC
Param	A1BellDelay.par	Needle valve delayup and delaydown
	A1Fluid.par	Punpin and Pumpout parameter
	A1FluidDelay.par	Pump delayup and delaydown parameter
	cnv_pib.par	Conveyor parameter
	A1HVRoboBell1000i.par	HV alarm parameter

7.4.1.2 Pixel paint file structure



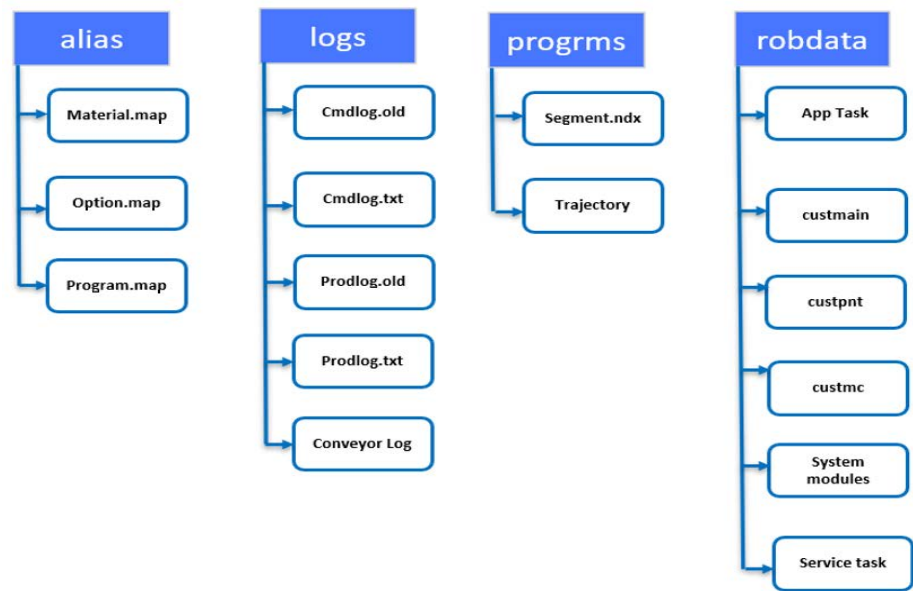
7.4.1.3 Clean robot file structure



7.4.1.4 Flaming robot file structure

Description The robot system program and trajectory program are saved in the robot file system. This chapter will give a description of the flaming robot file structure.

Home folder The home folder is where the main files are located. The home structure is as below:



Home folder & files:

Folders	Function
alias	• Material alias • Option alias • Program alias
logs	• Production log • Command log • Conveyor log
programs	• Trajectory files • Segment index
robdata	• Custmain • Custpnt • Custmc
safeCfgTemplate	• Safety configuration template file
PxTPU.config.xml	• TPU command file

Alias Folder

This folder contains the alias related to the name of the programs, materials and options.

Alias file	Description
Program.map	Programs alias
Option.map	Options alias
Material.map	Materials alias

Logs folder

The logs folder contains the logs of the robot command log, conveyor log, production log and user logs.

Logs File	Description
Cmdlog.old	The old logs of the commands which the robot have received.
Cmdlog.txt	The last logs of the commands which the robot have received.
Prodlog.old	The old logs of the productions information which the robot have executed, like programs number, materials number, job options, brush number, fluid consumptions, cycle time and so on.
Prodlog.txt	The last logs of the productions information which the robot have executed, like programs number, materials number, job options, brush number, fluid consumptions, cycle time and so on.
Conveyor.log	The log is for conveyors to export the information needed.

Programs folder

The following picture shows the main service programs module and its index numbers, user can set the program number to run the program.

Program No.	File name	Description
10001	yHomePos	Home position module
10002	yMtrChPos	Material position module
10003	yCalPos	Calibration position module
10004	yMaintPos	Maintenance position module
10005	yFireTempPos	Fire and flame temperature position module

7 Robot system

7.4 Paint instructions

Program No.	File name	Description
10006	yManCleanPos	Manual clean position module
10007	yDressPos	Dress position module
10008	yTCPCheckPos	TCP check position module
10009	yAppMaintPos	Applicator maintenance module
10010	Spare	Spare
10011	yUserPos1	User position 1
10012	yUserPos2	User position 2
10013	yUserPos3	User position 3
10014	yUserPos4	User position 4
10015	yUserPos5	User position 5
10016-10020	Spare	Spare
10021	GunClean	Gun clean module
10022	TipClean	Tip clean module
10023	HubClean	Hub clean module
10024	ToolClean	Tool clean module
10025	yFluidFlowTest	Fluid flow volume test module
10026	yDynBrushTest	Dynamic brush test module
10027	yStatBrushTest	Static brush test module
10028	yHubCleanFlowTest	Hub clean flow test module
10029	yHvDynamicTest	High voltage error test module
10030	yHvStaticTest	High voltage shut down test module
10031	Spare	Spare
10032	yCyclicBrakeCheck	Cycle brake check module
10033	Spare	Spare
10034	Spare	Spare
10035	AppCalTest	Applicator calibration module

File name	Description
ValveCheck.ndx	The opening time of the valves is too long
Segment.ndx	Program number mapping index file

Trajectory folder

The trajectory folder is where the routine modules are located. To distinguish each part of the routine, the module is usually named after the painting part of the object.

The following image or picture shows the common abbreviation and corresponding whole name of the car body. All the parts will composite the complete painting trajectory.

BCI_ESEA_BIL3

BCI_HC23_DM_BIL3

BCI_HT2_BIL3

BCI_SU_BIL3

BCIT_HC23_DM_BIL3

AppCalTest.mod

BaseCalculate.mod

BCI_ESEA_BIL3.mod

BCI_HC23_DM_BIL3.mod

BCI_HT2_BIL3.mod

BCI_Signal_L3.mod

BCI_SU_BIL3.mod

For example, the above image, “BCI_ESEA_BIL3”, is the trajectory module for the car body. The main module name is “BCI_ESEA_BIL3.mod”.

Folder path: Controller-Home-Programs.

Robdata folder

The system modules and logic functions are saved in the “robdata” folder.

Location	Module name	Intro
AppTask	AppLoop.sys	Application task main loop file
	AppFunc.sys	Application task function file
	AppPreventMaint.sys	Application task prevent maintenance file
	ValveCheck.sys	Valve open check system file
SeviceTask	ServiceLoop.sys	Service task main loop file
	Qnext.sys	Read next job information
	CnvService.sys	Conveyor ghost, conveyor halt function
	ShapeMaint.sys	Shape air for maintenance function
Robdata	Robdata.sys	Tool, speed, zone and work object data
	Sysdata.sys	System positions and world zones connected to IO signals
	Custmain.sys	User module for custom cycle event procedures
	Custcmd.sys	User module for custom application commands
	Custvar.sys	User modules for custom to define the variables which will be used
	CheckSystem.sys	Check the system license
	Cnvdata.sys	Conveyor user module
	Csvparse.sys	CSV parse module
	Ctsync.sys	Conveyor sync module
	Custconfig.sys	Customized configuration file
	Custdata.sys	Global user data and functions
	Errmessage.sys	Error message handling
	Userlog.sys	User log system file

7 Robot system

7.4 Paint instructions

Location	Module name	Intro
Custmain	Appinital.sys	Check and set applicator enable signal before painting
	appclean.sys	Applicator clean handling logic
	ldparse.sys	Parse the robot name, station name and robot type
	CheckJobmode.sys	Check the PLC interface get job module
	Infotopl.sys	Export the information communicate with PLC
	Interlock.sys	The interlock logic
	MixerRatioCheck.sys	Check the mixer ratio
	PrepostProgram.sys	Function for pre-program and post-program in trajectory module
	OverrideBrush.sys	Override the system default brush mapping
	Processflowstatus.sys	Export the process of the current job queue
	workflowStatus.sys	Set the function for workflow status
	Programload.sys	The function to loading sub trajectory modules.
	Waitconveyorpos.sys	For trajectory to update, refresh, check the conveyor waiting position
	ServiceControl.sys	Function to allow push button to release robot moving to home position
	Softservo.sys	Enable tuneservo for paint robot
	Wzoneuser.sys	Robot work zones and safety box
Custmc	Appalarm.sys	Applicator error handling during colour change process
	AutoMixerFlush.sys	Automatic clean mixer with fixed interval
	Mtrloption.sys	Special clean sequence logic
	Starmc.sys	Set global brush and start material change
	SovlSup.sys	Solvent pressure supervision
	Mixratio.sys	Set mix ratio for new program
	Waitmtrlpos.sys	Wait robot move to material change position to execute material change sequence
Location	Module name	Intro
Custpnt	Cmdhooks.sys	Events hook with the command
	Hvsolventinterlock.sys	Error handling for high voltage and solvent interlock.
	JobPassControl.sys	Function to check conveyor sync signal and log the signals
	Reseterr.sys	Event trip for general event and error reset function
Guisock	Guisock.sys	Socket interface for HMI
	Guisocks.sys	Socket variable for HMI

7.4.1.5 Dispense robot file structure

TBD

7.4.2 Program structure

The following sub-sections are included in this section:

'Paint robot program structure' on page 296

'Pixel paint program structure' on page 297

'Clean robot program structure' on page 297

'Flaming robot program structure' on page 298

'Dispense robot program structure' on page 299

7.4.2.1 Paint robot program structure

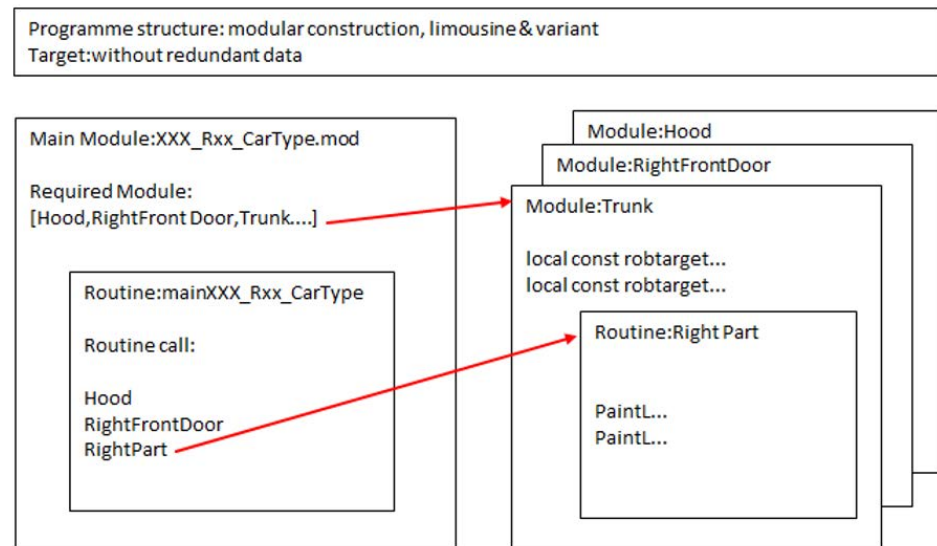
In order to avoid data redundancies in the movement programs, a modular program structure was implemented. The program flow is controlled in the main program. The trajectory program is loaded by the routine main XXX_RXX_CarType. The necessary movement programs are reloaded by a system call or directly via the index file.



In the main module, as above module "BCE1_Q6L_R11.mod", an array with a sub-module was defined like below. The sub-module is saved in a folder with the same name as the module name. The instruction in the main module will load the sub-modules to the T_ROB1 task.

```
LOCAL PERS string stModuleExecution{20}:=["R11_BDBF61_Q6L",
"R11_BDBF62_Q6L",
"R11_BDBF63_Q6L",
"R11_CP_Q6L",
"R11_CPtoTKV_Q6L",
"R11_TKV_Q6L",
"R11_TKVtoHome_Q6L",
"","","","","","","","","","","","","",""];

```



Main program

```

PrepareProgram stModuleExecution,jHomePos\vhHome:=vPaint400\zReady:=fine\nReadyBrush:=3\nSpeedRatio:=100\nAccRatio:=100;
IF nRepairTrunk=1 OR nRepairLFDDoor=1 OR nRepairLBDDoor=1 or nRepairLBFender=1 THEN
  velset 45,1500;
  R11_norpaint;
  velset 100,1500;
  FinishProgram jHomePos\vhHomeSpeed:=vPaint600\nStopBrush:=1,"Q6L_R11 Normal mode";
ELSE
  WaitTime 5;
  FinishProgram jHomePos\vhHomeSpeed:=vPaint500\nStopBrush:=1,"Q6L_R11 for repaired mode";
ENDIF
  
```

1. Prepare Program:

- Load program modules.
- Move to home position.
- Move to ready position.
- Set speed ratio.

2. Trajectory program

3. Finish Program:

- Move to home position.
- Output cycle time.

7.4.2.2 Pixel paint program structure

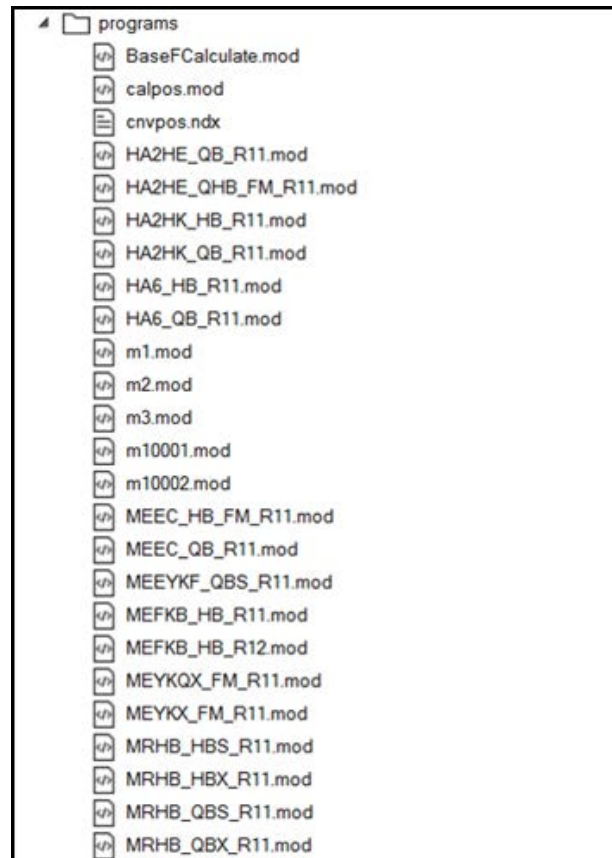
Pixel paint program structure is same as 'Paint robot program structure' on page 296.

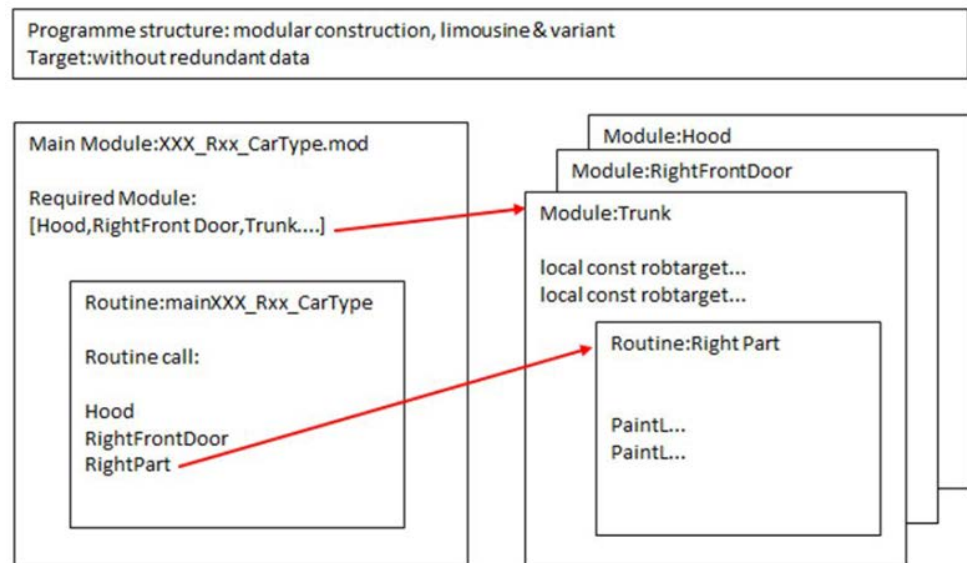
7.4.2.3 Clean robot program structure

Clean robot program structure is same as 'Paint robot program structure' on page 296.

7.4.2.4 Flaming robot program structure

In order to avoid data redundancies in the movement programs, a modular program structure was implemented. The program flow is controlled in the main program. The trajectory program is loaded by the routine main XXX_RXX_CarType. The necessary movement programs are reloaded by a system call or directly via the index file.





In the main module, similar to the above module, the sub-module is stored in a folder named after the module. The instructions in the main module will load the sub-modules into the T_ROB1 task.

Main program:

1. Prepare Program:

- Load program modules.
- Move to home position.
- Move to ready position.
- Set speed ratio.

2. Trajectory program

3. Finish Program:

- Move to home position.
- Output cycle time.

7.4.2.5 Dispense robot program structure

TBD

7.4.3 User data

The following sub-sections are included in this section:

'Paint robot user data' on page 300

'PixelPaint user data' on page 304

'Clean robot user data' on page 304

'Dispense robot user data' on page 305

7.4.3.1 Paint robot user data

The user data is saved in the module custvar.sys.

Below is the interlock data. If the following name does not meet the project requirements, the user can define the name by themselves. The value must be 1-6.

Variable name	Value
nExt_Hood	1
nExt_R_FD	2
nExt_L_FD	3
nExt_X_XX	4

Variable name	Value
nInt_Hood	1
nInt_R_FD_Opener	2
nInt_L_FD_Opener	3
nInt_R_BD_Opener	4
nInt_L_BD_Opener	5
nInt_Trunk	6

Below is the position code data. If the following name does not meet the project requirements, the user can define the name by themselves. The value must be between 101 and 65535.

Variable name	Value
Position_R_HD	121
Position_L_HD	122
Position_R_SR	123
Position_L_SR	124
Position_X_XX	...

Below is the process state data. If the following name does not meet the project requirements, the user can define the name by themselves. The value must be between 1-16.

Variable name	Value
State_RFrontDoorFinished	1
State_RBackDoorFinished	2
State_RHoodFinished	3
State_RTrunkFinished	4
State_LFrontDoorFinished	5
State_LBackDoorFinished	6
State_LHoodFinished	7
State_LTrunkFinished	8
Spare	9-16

The user can use the default clean configuration file as follows. The user can define the configuration according to the project configuration.

File name	Description
Bellcln.ndx	Bell clean index file
Hubcln.ndx	Hub clean index file
Guncln.ndx	Gun clean index file
Toolcln.ndx	Tool clean index file
BellClean.mod	Bell clean sequence program
HubClean.mod	Hub clean sequence program
GunClean.mod	Gun clean sequence program
ToolClean.mod	Tool clean sequence program

Variable name	Value
APPClean_NDX_FILE	["bellcln.ndx", "Hubcln.ndx", "Guncln.ndx", "Toolcln.ndx", ""]
APPClean_stSequence	["BellClean.mod", "HubClean.mod", "GunClean.mod", "ToolClean.mod", ""]

The colour valve name is used to save the valve value. This value will be used to restore the value to the colour changer when restarting the controller. The user can change the valve name if the default valve name does not include the project data.

Variable name	Value
ncolourValveVal	[0, 0, 0, 0, 0, 0]
APPClean_stSequence	["A1Col", "A1Cat", "A1Col1", "A1Col2", "A1Cat1", "A1Cat2"]

Below data is station mode. The user should not change the value.

Const var name	Value
ST_MAN_MODE	1
ST_AUTO_MODE	2
ST_MAINT_MODE	3
ST_CLEAN_MODE	4
ST_TEACH_MODE	5

Below is the interior robot data. The user should not change the value.

Below is the variable used for the painter robot. The variable represents the name of the opener.

Variable name	Value	Description
nH01	1	The left hood/trunk opener
nD01	2	The first door opener
nD02	3	The second door opener
nH02	4	The right hood/trunk opener

Below is the variable used for the opener robot. The variable represents the name of the painter from entry toward exit.

Variable name	Value	Description
nR01	1	The first painter
nR02	2	The second painter
nR03	4	The third painter
nR04	8	The fourth painter
nR05	16	The fifth painter

Door/hood opener variable (Conveyor tracking)

Variable name	Value
nDoor_F_Full_Opened	1
nDoor_F_Half_Opened	2
nDoor_F_Half_Closed	3
nDoor_F_Full_Closed	4
nDoor_B_Full_Opened	5
nDoor_B_Half_Opened	6
nDoor_B_Half_Closed	7
nDoor_B_Full_Closed	8
nHoodTrunk_Full_Opened	9
nHoodTrunk_Half_Opened	10
nHoodTrunk_Half_Closed	11
nHoodTrunk_Full_Closed	12
nHomePos	15

Door/hood opener variable (Stop&Go)

Variable name	Value
nFullOpenPos	1
nHalfOpenPos	2
nClosedPos	3
nHookPos	4
nUnHookPos	5
nReadyPos	6
nErrHandlerPos	11
nMaxPos	15
nHomePos	15

The variable shows the next moving position. (Stop&Go).

Variable name	Description
nNextPos	The next moving position

The variable for car variants.

Normally, the car type will use the variable “nCarType”. If a bool type variable is needed, the PLC also needs to send an option with a bit.

Variable name	Description	Option1 bit
nCarType	The num value for cartype	0-7
bCarType1	The bool value for cartype1	0
bCarType2	The bool value for cartype2	1
bCarType3	The bool value for cartype3	2
bCarType4	The bool value for cartype4	3
bCarType5	The bool value for cartype5	4
bCarType6	The bool value for cartype6	5
bCarType7	The bool value for cartype7	6
bCarType8	The bool value for cartype8	7

The variable for degrade mode.

Variable name	Description	Option1 bit
nPntStrategy	The num value for degrade mode	8-15
bPntStrategy1	The bool value for degrade strategy 1	8
bPntStrategy2	The bool value for degrade strategy 2	9
bPntStrategy3	The bool value for degrade strategy 3	10
bPntStrategy4	The bool value for degrade strategy 4	11
bPntStrategy5	The bool value for degrade strategy 5	12
bPntStrategy6	The bool value for degrade strategy 6	13
bPntStrategy7	The bool value for degrade strategy 7	14
bPntStrategy8	The bool value for degrade strategy 8	15

The repair part mode flag.

Variable name	Description
nRepairPart	The dnum value for repair part
bRepairSwitch	The bool flag for repair mod

The variable for repair mode, used in trajectory.

Variable	Description	Option2 bit
nRepairHood	Repair value for hood	1
nRepairRFFender	Repair value for front fender	2
nRepairRFDoor	Repair value for right front door	3
nRepairRBDoor	Repair value for right back door	4
nRepairRBFender	Repair value for right front fender	5
nRepairTrunk	Repair value for trunk	6
nRepairLBFender	Repair value for left back fender	7
nRepairLBDoor	Repair value for left back door	8
nRepairLFDoor	Repair value for left front door	9
nRepairLFFender	Repair value for left front fender	10
nRepairRoof	Repair value for roof	11

7.4.3.2 PixelPaint user data

The variable data for PixelPaint brush test:

Variable	Description	Default Value
nGUI_StrokeDistance	Stroke distance value for PixelPaint brush test (mm)	101.76
nGUI_StrokeCount	Stroke Counts for 1 brush test cycle	3
nGUI_TCPDistance	Tcp distance from print head to object	3
nGUI_Speed	Tcp speed during brush test (mm/s)	50

7.4.3.3 Clean robot user data

The user data saved in the module custvar.sys.

Below is the interlock data. If the following name does not reach the project require, the user can define the name by themselves. The value must be 1-16.

Variable name	Value
nExt_Hood	1
nExt_R_FD	2
nExt_L_FD	3
nExt_X_XX	4

Variable name	Value
nInt_Hood	1
nInt_R_FD_Opener	2
nInt_L_FD_Opener	3
nInt_R_BD_Opener	4

7 Robot system

7.5 Paint control functions

Variable name	Value
nInt_L_BD_Opener	5
nInt_Trunk	6

It is used for opener painter safe interlock.

Variable name	Value
nInt_HoodnInt_D12_SafeLock_1	8
nInt_D22_SafeLock_1	9
D12_SafeLock_2	13
nInt_D22_SafeLock_2	14
nInt_R13_BlockOpener	15
nInt_R23_BlockOpener	16

7.4.3.4 Dispense robot user data

TBD

7.5 Paint control functions

The following sub-sections are included in this section:

['General functions'](#) on page 305

['Paint robot functions'](#) on page 312

['PixelPaint Application functions'](#) on page 332

['Clean robot functions'](#) on page 339

['Flaming robot functions'](#) on page 340

['Dispense robot functions'](#) on page 341

7.5.1 General functions

The following sub-sections are included in this section:

['Job information check'](#) on page 306

['Stop Concept'](#) on page 306

['External Axis check'](#) on page 307

['Ghost conveyor for painting'](#) on page 307

['Conveyor halt for painting'](#) on page 308

['TeachPos function'](#) on page 309

['Smart Home'](#) on page 309

['Robot interlock'](#) on page 310

['Robot process status'](#) on page 311

['CBC function'](#) on page 312

7.5.1.1 Job information check

PLC send job information to the robot system. The robot system will save the job info to the job queue. Job information is critical data, and its accuracy is very important. There are several information handshakes to ensure the data is correct.

1. Job command handshake

When the robot receives the job data, normally the command 421, the robot will send the program number and material number back to PLC. The information will be kept available until the next job append information is sent. The PLC needs to check if the data is correct; if not, the PLC will trigger error information and ask the operator to double-check.

2. Job info handshake

Before the robot starts executing a new job, the robot will send the job information and a 2s pulse high signal to PLC to double-check if the current job information is correct or not. If not, PLC should trigger an error information and ask the operator to double-check.

7.5.1.2 Stop Concept

When the robot makes a fatal error, it will be stopped. For the other robots in one interior station, the default strategy is for the other robot to be stopped at the same time. For external station, the PLC will only stop conveyor, the other robot will move until reach interlock zone or wait conveyor position, waiting conveyor resume.

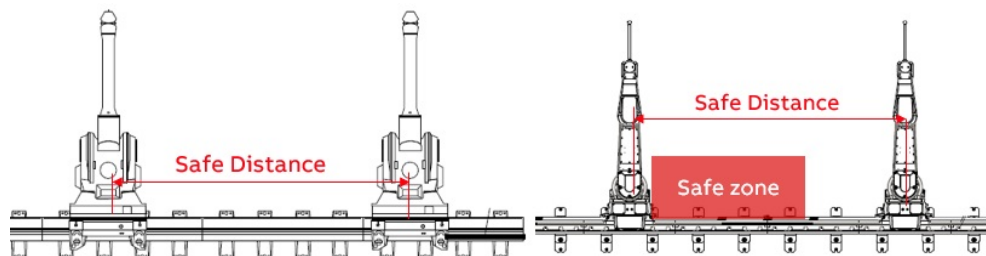
For the special request station, the stop concept can be changed based on the project request.

The following errors will stop the robot.

- System internal fatal errors, like collision, execution error, etc.
- System communication error
- Severe application error
- External stop command from PLC.

ExtaxCheck is for this situation; if there are two or more robots located at the same rail, there must be a monitor to make that the two trolleys cannot collide each other.

There is a minimum distance between the two trolleys, so you should define the distance.



The following par how to define the data.

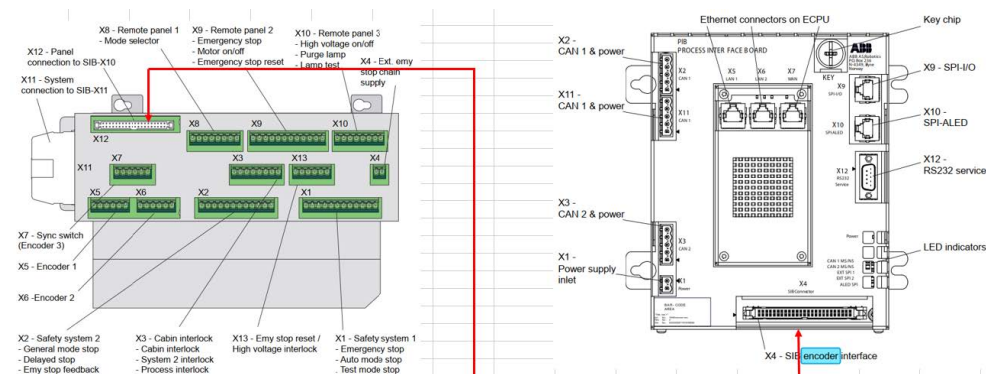
Example:

Name "CustData DoorlockZone Low" -Value "-1500"

Name "CustData DoorlockZone High" -Value "1500"

Name "CustData nExtraControlOffset" -Value "10000

For the conveyor tracking station, the conveyor position can be from a real conveyor or a simulated conveyor in a robot IPS system. Therefore, the robot can run with or without a conveyor in simulation mode. In ghost conveyor mode, the operator can run the update or check the conveyor wait position (the position to sync the robot with the conveyor). You can also start the production simulation with the correct parameter.



The ghost conveyor is controlled by PLC from the field bus.

Signal name	Description
GiSimSpeed	The ghost conveyor speed (from PLC)
diGhostCnv	The enable signal for ghost conveyor
diSimCnvRun	Run/stop signal for ghost conveyor

Ghost conveyor operation:

1. Entry ghost mode.
2. Station in auto mode.
3. Robot in auto mode and cycle on.
4. Click ghost with the conveyor.
5. Set the ghost job information and start the ghost mode.

7.5.1.5 Conveyor halt for painting

To keep the robot paint with a fixed start position in any case, the system needs to have a function to control the conveyor position before starting. The function will monitor the conveyor position and compare it with the proper start position; when the conveyor position is over the limit, the robot will request a conveyor halt, and when all the cleaning and filling sequences are done, the system will resume the conveyor.

The system will monitor conveyor position when the robot TCP in the colour change position, bell clean position, hub clean position and gun clean position.

The conveyor start position was defined in the file “Cnvpos.ndx”.

_____ Program Index

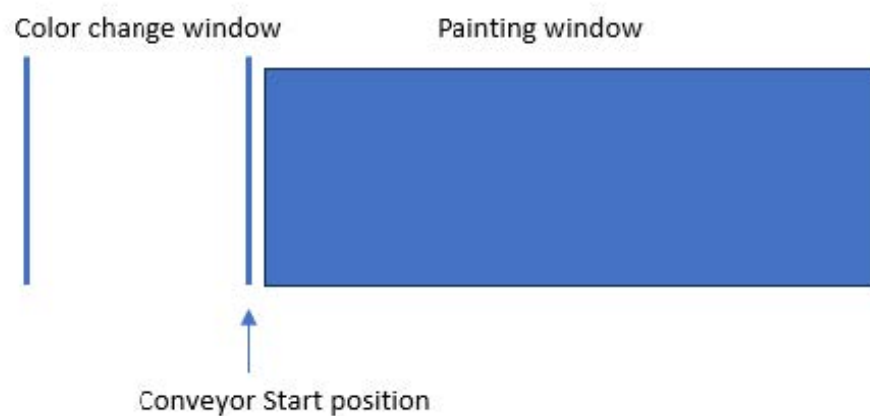
| _____ Initial Wait Position for Conveyor

| |

1, 1800

2, 1800

3, 1800



7.5.1.6 TeachPos function

TeachPos Function is for us to fast teach the service position of the robot with some reference from some known position. Before user start to use the function, user should make sure the value of these positions verified well, or the operation will be dangerous.

The following shows the steps of this function:

Step 1: Copy the sysdata.sys file from the certified robot;

Step 2: Upload the file and P-Start;

Step 3: Open the ServposTeach.mod module. The following example shows the usage of TeachRobTarget instruction.

Example:

```
TeachRobTarget JhomePos, JhomePos, pHomePos, v100;  
| | | _____ The RobotTarget in sysdata for service position  
| | _____ The JointTarget in sysdata for service position  
| _____ The Known JointTarget already taught or copy from  
other station, the name can be any name, no need the same name as the JointTarget  
want to teach, here only a example will the same name.
```

As shows, The pHomepos is the robottarget which will be defined, the middle jHomepos is the Jointtarget which will be defined, the first Jhomepos is the joint name of the service position, normally it can be the same with the second JointTarget, but it can be any other joint target.

Step 4: Manually run the program in TPU or in AUTO mode.

Step 5: After run the robot, need restart the robot to active the service position.



Important: System positions which are connected to a digital output on the fieldbus need to be taught with robtargets as well as jointtargets. For these positions, a MoveL and MoveAbsJ instruction is already available in the routines mentioned above. Make sure to teach both targets on the same position.

7.5.1.7 Smart Home

The smart home is the function that allows at any position robots in one station to move back to the home position at any position. When a robot has an error and is unable to continue moving, the operator must manually jog the other robots back to their home position. This jogging operation will take a long time. To save error recovery time, with the smart home function, the operator can simply click the “Smart Home” button, and all robots can move home together.

Smart home function can support exterior station, conveyor tracking line and Stop&Go line. If another type of station needs this function, it needs analysis separately.

For exterior station:

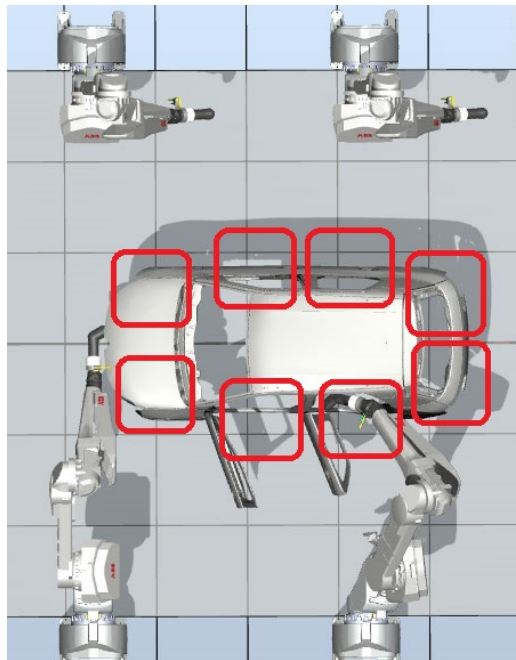
1. Cancel the current program by sending the command “Reset PP”.
2. Cycle on the robot in station - manual mode and click “Smart home”.

For interior station:

1. Cancel the paint robot’s current program by sending the command “Reset PP”, not cancel the opener’s current job.
2. Cycle on the robot in station - manual mode and click “Smart home” for paint robots.
3. Wait for the painter in the home position.
4. In the opener’s operation window, manipulate the opener back to home.

7.5.1.8 Robot interlock

There are 16 interlocked areas supported in a station. Before entering the working area, the robot will check if the area has been released by other robots. If the area is locked, the robot will wait for the area to be released. The robot will send the locked signal to the PLC when entering a released working area.



The below table shows the default variable data:

Exterior zone	Interior zone	Bit position
nExt_Hood	nInt_Hood	1
nExt_R_FD	nInt_R_FD_Opener	2
nExt_L_FD	nInt_L_FD_Opener	3
nExt_R_BD	nInt_R_BD_Opener	4
nExt_L_BD	nInt_L_BD_Opener	5
nExt_R_FF	nInt_Trunk	6
nExt_L_FF	nInt_D12_SafeLock_1	7
nExt_R_BF	nInt_D22_SafeLock_1	8
nExt_L_BF	nInt_D12_SafeLock_2	9
nExt_SunRoof	nInt_D22_SafeLock_2	10
nExt_Roof	nInt_R13_BlockOpener	11
nExt_Trunk	nInt_R23_BlockOpener	12
nExt_R_7Axi	Not Define	13
nExt_L_7Axi	Not Define	14
nExt_L_CoverArea1	Not Define	15
nExt_R_CoverArea2	Not Define	16

7.5.1.9 Robot process status

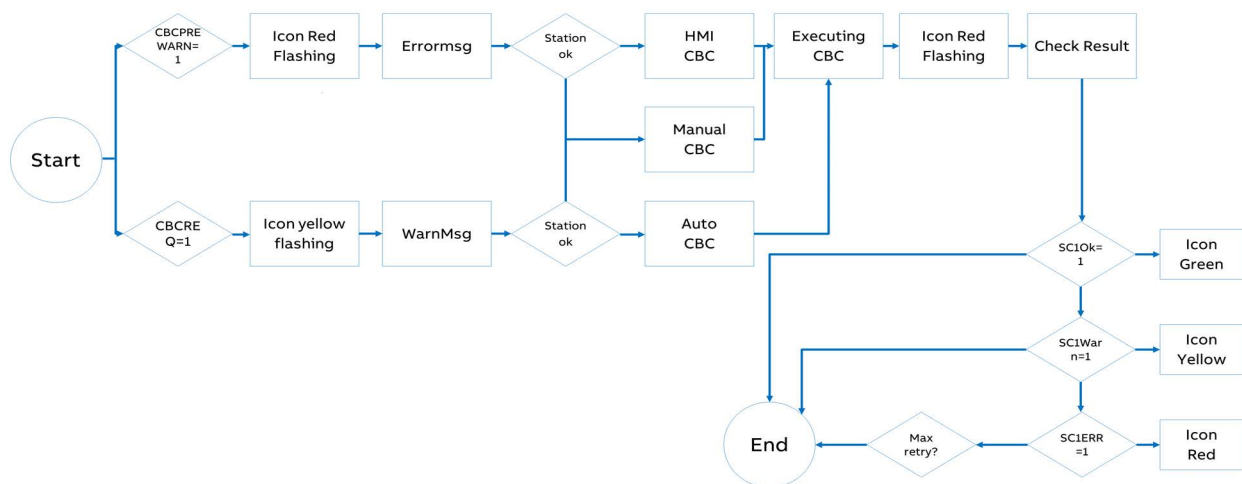
There are 16 process status areas supported in a station. When one area is painted or work done, the robot can set the status of this area. This status can be used by other robots to read the station painting status.

Status	Bit position
State_RFrontDoorFinished	1
State_RBackDoorFinished	2
State_RHoodFinished	3
State_RTrunkFinished	4
State_LFrontDoorFinished	5
State_LBackDoorFinished	6
State_LHoodFinished	7
State_LTrunkFinished	8
Not Define	9
Not Define	10
Not Define	11
Not Define	12
Not Define	13
Not Define	14
Not Define	15
Not Define	16

7.5.1.10 CBC function

CBC function is used to test the brake status of the robot motor. The test interval and parameter can be set by the safety user. The CBC check can be run both in robot manual mode and auto mode. In manual mode, the operator manually runs the CBC check with TPU. In automatic mode, the robot can get the CBC command from PLC automatically or with the operator manually clicking the CBC request command.

The CBC loading workflow is as follows.



7.5.2 Paint robot functions

The atomizer becomes more dirty when production takes a long time. So after one cycle time, an applicator clean must be done to ensure production.

The following sub-sections are included in this section:

- 'Applicator clean' on page 313
- 'Material Change' on page 319
- 'Global brush settings' on page 324
- 'Volume calculation' on page 325
- 'Application supervision' on page 327
- 'Application test' on page 328
- 'Preventive maintenance functions' on page 330

7 Robot system

7.5 Paint control functions

7.5.2.1 Applicator clean

Bellclean - function

Bellclean cleans SA Nozzle of atomizer after the production of some cars (like batchwise cars), reducing some quality problems of the car (like paint residue problems).



For example:

If user don't have the clean box, user can change the cleaning sequence to 0 or change the clean time to 0 in bellcln.ndx (file path: Home/mcdata). If user have, user need to set some value of sequence and clean time.

User should know the first column is material, the second column is the frequency of production cars, and the third is the time of bellclean. If user have, user change the air and solvent alternation sequence in bellcln.mod (file path: Home/mcdata) to get the purpose of cleaning.

alias	2024/3/7 17:04	文件夹	bellcln.cfg	2024/3/7 17:04	CFG 文件	1 KB
GSDML	2024/3/7 17:04	文件夹	bellcln.mod	2024/3/7 17:04	MOD 文件	1 KB
ipsdata	2024/3/7 17:04	文件夹	bellcln.ndx	2024/3/10 20:54	NDX 文件	1 KB
logs	2024/3/7 17:04	文件夹	c1.cfg	2024/3/7 17:04	CFG 文件	1 KB
mcdata	2024/3/10 21:15	文件夹	c1clean.ndx	2024/3/7 17:04	NDX 文件	1 KB
programs	2024/3/7 17:04	文件夹	c1fill.ndx	2024/3/7 17:04	NDX 文件	1 KB
rmc	2024/3/7 17:04	文件夹	c1fillA.mod	2024/3/7 17:04	MOD 文件	1 KB
robdata	2024/3/7 17:04	文件夹	c1fills.mod	2024/3/7 17:04	MOD 文件	1 KB
SafecfgTemplate	2024/3/7 17:04	文件夹	c1g1cln.mod	2024/3/7 17:04	MOD 文件	2 KB
syspar	2024/3/7 17:04	文件夹	c1g1fill.mod	2024/3/7 17:04	MOD 文件	1 KB
CBC_REPORT.log	2024/3/7 17:04	文本文档	c1valve.ndx	2024/3/7 17:04	NDX 文件	1 KB
LIC_503868146.sys	2024/3/7 17:04	SYS 文件	c2.cfg	2024/3/7 17:04	CFG 文件	1 KB
LIC_VIRTUAL_USE.sys	2024/3/7 17:04	SYS 文件	c2clean.ndx	2024/3/7 17:04	NDX 文件	1 KB
PxTPU.config.xml	2024/3/7 17:04	Microsoft Edge ...	c2fill.ndx	2024/3/7 17:04	NDX 文件	1 KB
sc_vsm_metadata.xml	2024/3/7 17:04	Microsoft Edge ...	c2fillA.mod	2024/3/7 17:04	MOD 文件	1 KB
user.sys	2024/3/7 17:04	SYS 文件	c2fills.mod	2024/3/7 17:04	MOD 文件	1 KB

User can call bellclean.mod through defined some variable value (file path Home/robdata/custvar.sys) and use the function (file path Home/robdata/custmain/appclean.sys).

alias	2024/3/7 17:04	文件夹	AppTask	2024/3/7 17:04	文件夹	
GSDML	2024/3/7 17:04	文件夹	custmain	2024/3/7 17:04	文件夹	
ipsdata	2024/3/7 17:04	文件夹	custmc	2024/3/7 17:04	文件夹	
logs	2024/3/7 17:04	文件夹	custpnt	2024/3/7 17:04	文件夹	
mcddata	2024/3/10 21:15	文件夹	guisock	2024/3/7 17:04	文件夹	
programs	2024/3/7 17:04	文件夹	ServiceTask	2024/3/7 17:04	文件夹	
rmc	2024/3/7 17:04	文件夹	backlash.sys	2024/3/7 17:04	SYS 文件	12 KB
robdata	2024/3/7 17:04	文件夹	base_copy.sys	2024/3/7 17:04	SYS 文件	3 KB
SafecfgTemplate	2024/3/7 17:04	文件夹	checksystem.sys	2024/3/7 17:04	SYS 文件	3 KB
syspar	2024/3/7 17:04	文件夹	cnvdata.sys	2024/3/7 17:04	SYS 文件	2 KB
CBC_REPORT.log	2024/3/7 17:04	文本文件	csvparse.sys	2024/3/7 17:04	SYS 文件	4 KB
LIC_503868146.sys	2024/3/7 17:04	SYS 文件	ctsync.sys	2024/3/7 17:04	SYS 文件	10 KB
LIC_VIRTUAL_USE.sys	2024/3/7 17:04	SYS 文件	custcmd.sys	2024/3/7 17:04	SYS 文件	10 KB
PxTPU.config.xml	2024/3/7 17:04	Microsoft Edge ...	custconfig.sys	2024/3/7 17:04	SYS 文件	3 KB
sc_vsm_metadata.xml	2024/3/7 17:04	Microsoft Edge ...	custdata.sys	2024/3/7 17:04	SYS 文件	5 KB
user.sys	2024/3/7 17:04	SYS 文件	custmain.sys	2024/3/7 17:04	SYS 文件	2 KB
appclean.sys	2024/3/7 17:04	SYS 文件	custvar.sys	2024/3/7 17:04	SYS 文件	8 KB
appinital.sys	2024/3/7 17:04	SYS 文件				
CheckJobMode.sys	2024/3/7 17:04	SYS 文件				
idparse.sys	2024/3/7 17:04	SYS 文件				
interlock.sys	2024/3/7 17:04	SYS 文件				
OverrideBrush.sys	2024/3/7 17:04	SYS 文件				
PrePostProgram.sys	2024/3/7 17:04	SYS 文件				
ProcessFlowStatus.sys	2024/3/7 17:04	SYS 文件				
ProgramLoad.sys	2024/3/7 17:04	SYS 文件				
ServicePosControl.sys	2024/3/7 17:04	SYS 文件				
SoftServo.sys	2024/3/7 17:04	SYS 文件				
WaitConveyorPos.sys	2024/3/7 17:04	SYS 文件				
WorkFlowStatus.sys	2024/3/7 17:04	SYS 文件				
wzoneuser.sys	2024/3/7 17:04	SYS 文件				

User can change the speed of the robot to go to the clean box and back to home or how to dry solvent in the bellclean.mod.

alias	2024/3/7 17:04	文件夹	BCE_EM2E_L1	2024/3/7 17:04	文件夹	
GSDML	2024/3/7 17:04	文件夹	AppCalTest.mod	2024/3/7 17:04	MOD 文件	6 KB
ipsdata	2024/3/7 17:04	文件夹	BasefCalculate.mod	2024/3/7 17:04	MOD 文件	1 KB
logs	2024/3/7 17:04	文件夹	BCE_EM2E_L1.mod	2024/3/7 17:04	MOD 文件	11 KB
mcddata	2024/3/10 21:15	文件夹	BCE_HeadTest_L1.Mod	2024/3/7 17:04	MOD 文件	1 KB
programs	2024/3/7 17:04	文件夹	BellClean.mod	2024/3/7 17:04	MOD 文件	2 KB
rmc	2024/3/7 17:04	文件夹	calpos.mod	2024/3/7 17:04	MOD 文件	5 KB
robdata	2024/3/7 17:04	文件夹	cnvpos.ndx	2024/3/7 17:04	NDX 文件	1 KB
SafecfgTemplate	2024/3/7 17:04	文件夹	HubClean.mod	2024/3/7 17:04	MOD 文件	2 KB
syspar	2024/3/7 17:04	文件夹	segments.ndx	2024/3/7 17:04	NDX 文件	1 KB
CBC_REPORT.log	2024/3/7 17:04	文本文件	ServPostTeach.mod	2024/3/7 17:04	MOD 文件	2 KB
LIC_503868146.sys	2024/3/7 17:04	SYS 文件	ValveCheck.ndx	2024/3/7 17:04	NDX 文件	1 KB
LIC_VIRTUAL_USE.sys	2024/3/7 17:04	SYS 文件	WobjfCalculate.mod	2024/3/7 17:04	MOD 文件	1 KB
PxTPU.config.xml	2024/3/7 17:04	Microsoft Edge ...	WobjfCalculate_EM2E.mod	2024/3/7 17:04	MOD 文件	2 KB
sc_vsm_metadata.xml	2024/3/7 17:04	Microsoft Edge ...	yAppMaintPos.mod	2024/3/7 17:04	MOD 文件	1 KB
user.sys	2024/3/7 17:04	SYS 文件				

Hubclean - function

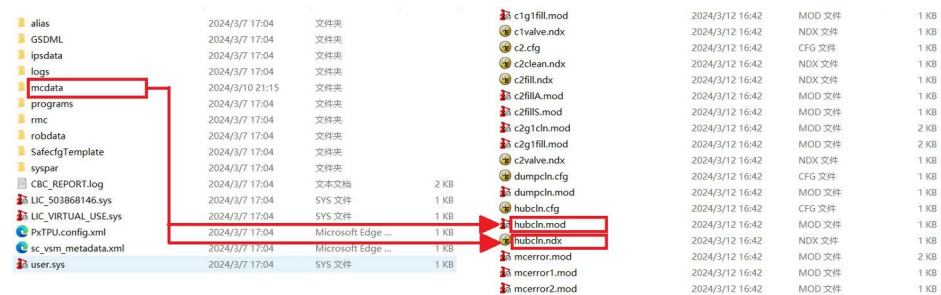
Hubclean is a clean hub of atomizer after the production of some cars (like one or two cars), reducing some quality problems in the car (like paint residue problems).



For example:

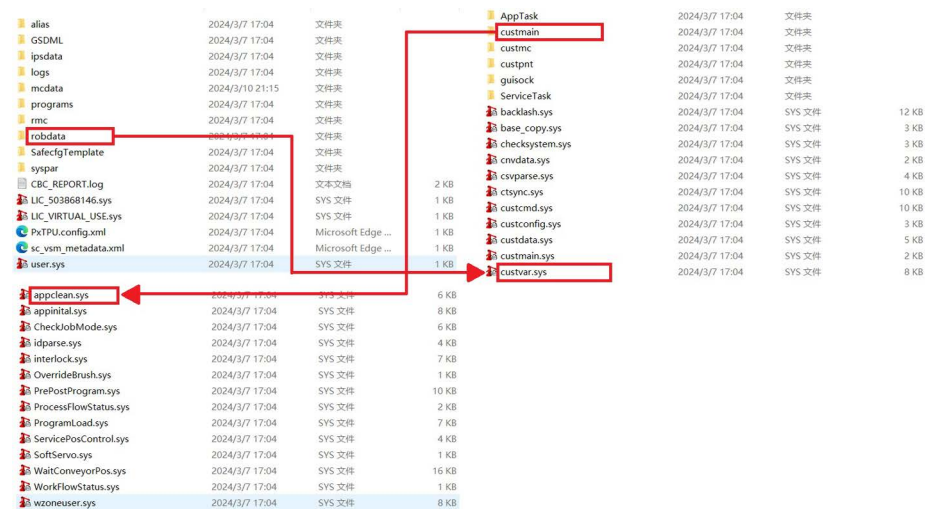
If user don't clean the hub, user can change the cleaning sequence to 0 or change the clean time to 0 in Hubcln.ndx (file path: Home/mcdata).

If user need to, user should set some values for the sequence and clean time. User should know the first column is material, the second column is the frequency of production cars, and the third is the time of HubClean. User can control air and solvent in hubcln.mod (file path: Home/mcdata) to make sure the hub is cleaned.



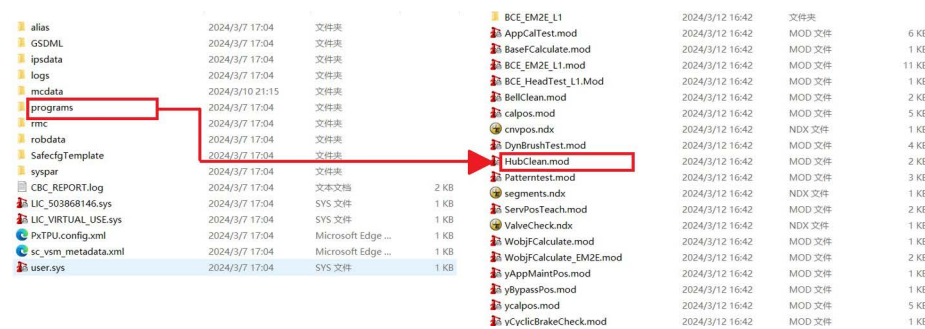
File Name	Date	Type	Size
alias	2024/3/7 17:04	文件夹	
GSDML	2024/3/7 17:04	文件夹	
ipsdata	2024/3/7 17:04	文件夹	
logs	2024/3/7 17:04	文件夹	
mcdata	2024/3/10 21:15	文件夹	
programs	2024/3/7 17:04	文件夹	
rmc	2024/3/7 17:04	文件夹	
robdata	2024/3/7 17:04	文件夹	
SafecfgTemplate	2024/3/7 17:04	文件夹	
syspar	2024/3/7 17:04	文件夹	
CBC_REPORT.log	2024/3/7 17:04	文本文件	2 KB
LUC_503868146.sys	2024/3/7 17:04	SYS 文件	1 KB
LUC_VIRTUAL_USE.sys	2024/3/7 17:04	SYS 文件	1 KB
PxTPU.config.xml	2024/3/7 17:04	Microsoft Edge ...	1 KB
vc_vsm_metadata.xml	2024/3/7 17:04	Microsoft Edge ...	1 KB
user.sys	2024/3/7 17:04	SYS 文件	1 KB
c1g1fill.mod	2024/3/12 16:42	MOD 文件	1 KB
c1valve.ndx	2024/3/12 16:42	NDX 文件	1 KB
c2.cfg	2024/3/12 16:42	CFG 文件	1 KB
c2clean.ndx	2024/3/12 16:42	NDX 文件	1 KB
c2fill.ndx	2024/3/12 16:42	NDX 文件	1 KB
c2fillA.mod	2024/3/12 16:42	MOD 文件	1 KB
c2fillB.mod	2024/3/12 16:42	MOD 文件	1 KB
c2g1cln.mod	2024/3/12 16:42	MOD 文件	2 KB
c2g1fill.mod	2024/3/12 16:42	MOD 文件	2 KB
c2valve.ndx	2024/3/12 16:42	NDX 文件	1 KB
dumpcln.cfg	2024/3/12 16:42	CFG 文件	1 KB
dumpcln.mod	2024/3/12 16:42	MOD 文件	1 KB
hubcln.cfg	2024/3/12 16:42	CFG 文件	1 KB
hubcln.mod	2024/3/12 16:42	MOD 文件	1 KB
hubcln.ndx	2024/3/12 16:42	NDX 文件	1 KB
mcerror.mod	2024/3/12 16:42	MOD 文件	2 KB
mcerror1.mod	2024/3/12 16:42	MOD 文件	1 KB
mcerror2.mod	2024/3/12 16:42	MOD 文件	1 KB

User can call HubClean.mod through defined some variable value (file path Home/robdata/custvar.sys) and use the function (file path Home/robdata/custmain/appclean.sys).



File Name	Date	Type	Size
alias	2024/3/7 17:04	文件夹	
GSDML	2024/3/7 17:04	文件夹	
ipsdata	2024/3/7 17:04	文件夹	
logs	2024/3/7 17:04	文件夹	
mcdata	2024/3/10 21:15	文件夹	
programs	2024/3/7 17:04	文件夹	
rmc	2024/3/7 17:04	文件夹	
robdata	2024/3/7 17:04	文件夹	
SafecfgTemplate	2024/3/7 17:04	文件夹	
syspar	2024/3/7 17:04	文件夹	
CBC_REPORT.log	2024/3/7 17:04	文本文件	2 KB
LUC_503868146.sys	2024/3/7 17:04	SYS 文件	1 KB
LUC_VIRTUAL_USE.sys	2024/3/7 17:04	SYS 文件	1 KB
PxTPU.config.xml	2024/3/7 17:04	Microsoft Edge ...	1 KB
vc_vsm_metadata.xml	2024/3/7 17:04	Microsoft Edge ...	1 KB
user.sys	2024/3/7 17:04	SYS 文件	1 KB
AppTask	2024/3/7 17:04	文件夹	
custmain	2024/3/7 17:04	文件夹	
custmc	2024/3/7 17:04	文件夹	
custpnt	2024/3/7 17:04	文件夹	
guisock	2024/3/7 17:04	文件夹	
ServiceTask	2024/3/7 17:04	文件夹	
backlash.sys	2024/3/7 17:04	SYS 文件	12 KB
base_copy.sys	2024/3/7 17:04	SYS 文件	3 KB
checksystem.sys	2024/3/7 17:04	SYS 文件	3 KB
cnadata.sys	2024/3/7 17:04	SYS 文件	2 KB
csparse.sys	2024/3/7 17:04	SYS 文件	4 KB
ctsync.sys	2024/3/7 17:04	SYS 文件	10 KB
custcmd.sys	2024/3/7 17:04	SYS 文件	10 KB
custconfig.sys	2024/3/7 17:04	SYS 文件	3 KB
custdata.sys	2024/3/7 17:04	SYS 文件	5 KB
custmain.sys	2024/3/7 17:04	SYS 文件	2 KB
custvar.sys	2024/3/7 17:04	SYS 文件	8 KB
appclean.sys	2024/3/7 17:04	SYS 文件	6 KB
appinit.sys	2024/3/7 17:04	SYS 文件	8 KB
CheckJobMode.sys	2024/3/7 17:04	SYS 文件	6 KB
idparse.sys	2024/3/7 17:04	SYS 文件	4 KB
interlock.sys	2024/3/7 17:04	SYS 文件	7 KB
OverrideBrush.sys	2024/3/7 17:04	SYS 文件	1 KB
PrePostProgram.sys	2024/3/7 17:04	SYS 文件	10 KB
ProcessFlowStatus.sys	2024/3/7 17:04	SYS 文件	2 KB
ProgramLoad.sys	2024/3/7 17:04	SYS 文件	7 KB
ServicePosControl.sys	2024/3/7 17:04	SYS 文件	4 KB
SoftServo.sys	2024/3/7 17:04	SYS 文件	1 KB
WaitConveyorPos.sys	2024/3/7 17:04	SYS 文件	16 KB
WorkflowStatus.sys	2024/3/7 17:04	SYS 文件	1 KB
wzoneuser.sys	2024/3/7 17:04	SYS 文件	8 KB

User can change the speed of the robot to go to the position of HubClean and back home in the Hubclean.mod.



File Name	Date	Type	Size
alias	2024/3/7 17:04	文件夹	
GSDML	2024/3/7 17:04	文件夹	
ipsdata	2024/3/7 17:04	文件夹	
logs	2024/3/7 17:04	文件夹	
mcdata	2024/3/10 21:15	文件夹	
programs	2024/3/7 17:04	文件夹	
rmc	2024/3/7 17:04	文件夹	
robdata	2024/3/7 17:04	文件夹	
SafecfgTemplate	2024/3/7 17:04	文件夹	
syspar	2024/3/7 17:04	文件夹	
CBC_REPORT.log	2024/3/7 17:04	文本文件	2 KB
LUC_503868146.sys	2024/3/7 17:04	SYS 文件	1 KB
LUC_VIRTUAL_USE.sys	2024/3/7 17:04	SYS 文件	1 KB
PxTPU.config.xml	2024/3/7 17:04	Microsoft Edge ...	1 KB
vc_vsm_metadata.xml	2024/3/7 17:04	Microsoft Edge ...	1 KB
user.sys	2024/3/7 17:04	SYS 文件	1 KB
BCE EM2E L1	2024/3/12 16:42	文件夹	
AppCallTest.mod	2024/3/12 16:42	MOD 文件	6 KB
BaseCalculate.mod	2024/3/12 16:42	MOD 文件	1 KB
BCE EM2E L1.mod	2024/3/12 16:42	MOD 文件	11 KB
BCE HeadTest_L1.Mod	2024/3/12 16:42	MOD 文件	2 KB
BelClean.mod	2024/3/12 16:42	MOD 文件	2 KB
calpos.mod	2024/3/12 16:42	MOD 文件	5 KB
cnvpos.ndx	2024/3/12 16:42	NDX 文件	1 KB
DynBrushTest.mod	2024/3/12 16:42	MOD 文件	4 KB
HubClean.mod	2024/3/12 16:42	MOD 文件	2 KB
PatternTest.mod	2024/3/12 16:42	MOD 文件	3 KB
segments.ndx	2024/3/12 16:42	NDX 文件	1 KB
ServPosTech.mod	2024/3/12 16:42	MOD 文件	2 KB
ValveCheck.ndx	2024/3/12 16:42	NDX 文件	1 KB
WebCalculate.mod	2024/3/12 16:42	MOD 文件	1 KB
WebCalculate_EM2E.mod	2024/3/12 16:42	MOD 文件	2 KB
yAppMainPos.mod	2024/3/12 16:42	MOD 文件	1 KB
yBypassPos.mod	2024/3/12 16:42	MOD 文件	1 KB
yCalpos.mod	2024/3/12 16:42	MOD 文件	5 KB
yCyclicBrakeCheck.mod	2024/3/12 16:42	MOD 文件	1 KB

Gunclean - function

Gunclean is a cleaning cap of the gun after the production of some bumpers (like one or two bumpers), reducing some quality problems of bumpers (like paint residue problems).



For example:

If user don't have gunclean box, user can change the cleaning sequence to 0 or change the clean time to 0 in Guncln.ndx (file path: Home/mcdata). If user have, user should set some value of sequence and clean time.

User should know the first column is material, the second column is the frequency of production cars, and the third is the time of GunClean.

alias	2024/3/7 17:04	文件夹	c2fillA.mod	2023/11/9 15:29	MOD 文件	1 KB
GSDML	2024/3/7 17:04	文件夹	c2fills.mod	2023/11/9 15:29	MOD 文件	1 KB
lpsdata	2024/3/7 17:04	文件夹	c2g1cln.mod	2023/11/9 15:29	MOD 文件	2 KB
logs	2024/3/7 17:04	文件夹	c2g1fill.mod	2023/11/9 15:29	MOD 文件	1 KB
mcdata	2024/3/7 17:04	文件夹	c2valve.ndx	2023/11/9 15:29	NDX 文件	1 KB
programs	2024/3/7 17:04	文件夹	c3.cfg	2023/11/9 15:29	CFG 文件	1 KB
rnc	2024/3/7 17:04	文件夹	c3clean.ndx	2023/11/9 15:29	NDX 文件	1 KB
robodata	2024/3/7 17:04	文件夹	c3fill.ndx	2023/11/9 15:29	NDX 文件	1 KB
SafeCfgTemplate	2024/3/7 17:04	文件夹	c3fillA.mod	2023/11/9 15:29	MOD 文件	1 KB
syspar	2024/3/7 17:04	文件夹	c3fills.mod	2023/11/9 15:29	MOD 文件	1 KB
CBC_REPORT.log	2024/3/7 17:04	文本文件	c3g1cln.mod	2023/11/9 15:29	MOD 文件	1 KB
LIC_503868146.sys	2024/3/7 17:04	SYS 文件	c3g1fill.mod	2023/11/9 15:29	MOD 文件	1 KB
LIC_VIRTUAL_USE.sys	2024/3/7 17:04	SYS 文件	c3valve.ndx	2023/11/9 15:29	NDX 文件	1 KB
PxTpu.config.xml	2024/3/7 17:04	Microsoft Edge ...	guncln.cfg	2023/11/9 15:29	CFG 文件	1 KB
sc_vsm_metadata.xml	2024/3/7 17:04	Microsoft Edge ...	guncln.mod	2023/11/9 15:29	MOD 文件	1 KB
user.sys	2024/3/7 17:04	SYS 文件	guncln.ndx	2023/11/9 15:29	NDX 文件	1 KB
			mcerror.mod	2023/11/9 15:29	MOD 文件	2 KB
			mcerror1.mod	2023/11/9 15:29	MOD 文件	1 KB
			mcerror2.mod	2023/11/9 15:29	MOD 文件	1 KB
			potterr.mod	2023/11/9 15:29	MOD 文件	1 KB
			potwarn.mod	2023/11/9 15:29	MOD 文件	1 KB

7 Robot system

7.5 Paint control functions

User can call GunClean.mod through defined some variable value (file path Home/robdata/custvar.sys) and use the function (file path Home/robdata/custmain/appclean.sys).

alias	2024/3/7 17:04	文件夹		
GSDML	2024/3/7 17:04	文件夹		
ipsdata	2024/3/7 17:04	文件夹		
logs	2024/3/7 17:04	文件夹		
mcdata	2024/3/10 21:15	文件夹		
programs	2024/3/7 17:04	文件夹		
rmc	2024/3/7 17:04	文件夹		
robdata	2024/3/7 17:04	文件夹		
SafecfgTemplate	2024/3/7 17:04	文件夹		
syspar	2024/3/7 17:04	文件夹		
CBC_REPORT.log	2024/3/7 17:04	文本文档	2 KB	
LIC_503868146.sys	2024/3/7 17:04	SYS 文件	1 KB	
LIC_VIRTUAL_USE.sys	2024/3/7 17:04	SYS 文件	1 KB	
PxTPU.config.xml	2024/3/7 17:04	Microsoft Edge ...	1 KB	
sc_vsm_metadata.xml	2024/3/7 17:04	Microsoft Edge ...	1 KB	
user.sys	2024/3/7 17:04	SYS 文件	1 KB	
appclean.sys	2024/3/7 17:04	SYS 文件	6 KB	
appinit.sys	2024/3/7 17:04	SYS 文件	8 KB	
CheckJobMode.sys	2024/3/7 17:04	SYS 文件	6 KB	
idparse.sys	2024/3/7 17:04	SYS 文件	4 KB	
interlock.sys	2024/3/7 17:04	SYS 文件	7 KB	
OverrideBrush.sys	2024/3/7 17:04	SYS 文件	1 KB	
PrePostProgram.sys	2024/3/7 17:04	SYS 文件	10 KB	
ProcessFlowStatus.sys	2024/3/7 17:04	SYS 文件	2 KB	
ProgramLoad.sys	2024/3/7 17:04	SYS 文件	7 KB	
ServicePostControl.sys	2024/3/7 17:04	SYS 文件	4 KB	
SoftServo.sys	2024/3/7 17:04	SYS 文件	1 KB	
WaitConveyorPos.sys	2024/3/7 17:04	SYS 文件	1 KB	
WorkFlowStatus.sys	2024/3/7 17:04	SYS 文件	1 KB	
wzoneuser.sys	2024/3/7 17:04	SYS 文件	8 KB	
AppTask	2024/3/7 17:04	文件夹		
custmain	2024/3/7 17:04	文件夹		
custmc	2024/3/7 17:04	文件夹		
custpnt	2024/3/7 17:04	文件夹		
guisock	2024/3/7 17:04	文件夹		
ServiceTask	2024/3/7 17:04	文件夹		
backlash.sys	2024/3/7 17:04	SYS 文件	12 KB	
base_copy.sys	2024/3/7 17:04	SYS 文件	3 KB	
checksystem.sys	2024/3/7 17:04	SYS 文件	3 KB	
cnvdata.sys	2024/3/7 17:04	SYS 文件	2 KB	
csvparse.sys	2024/3/7 17:04	SYS 文件	4 KB	
ctsync.sys	2024/3/7 17:04	SYS 文件	10 KB	
custcmd.sys	2024/3/7 17:04	SYS 文件	10 KB	
custconfig.sys	2024/3/7 17:04	SYS 文件	3 KB	
custdata.sys	2024/3/7 17:04	SYS 文件	5 KB	
custmain.sys	2024/3/7 17:04	SYS 文件	2 KB	
custvar.sys	2024/3/7 17:04	SYS 文件	8 KB	

User can change the speed of the robot to go to the position of HubClean and back home in the gunclean.mod. User can change the clean speed and times during the clean time in gunclean.mod.

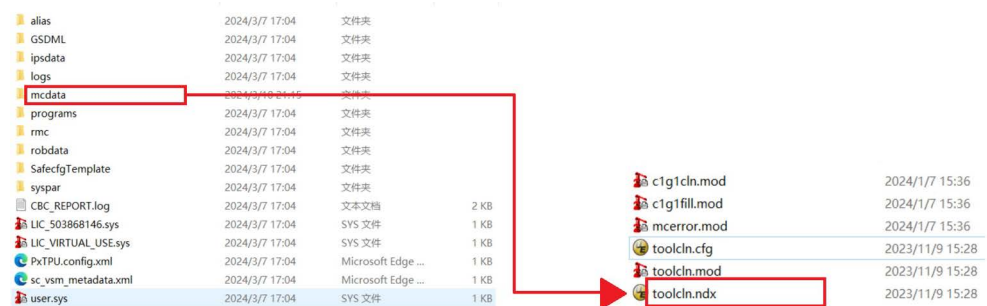
alias	2024/3/7 17:04	文件夹		
GSDML	2024/3/7 17:04	文件夹		
ipsdata	2024/3/7 17:04	文件夹		
logs	2024/3/7 17:04	文件夹		
mcdata	2024/3/10 21:15	文件夹		
programs	2024/3/7 17:04	文件夹		
rmc	2024/3/7 17:04	文件夹		
robdata	2024/3/7 17:04	文件夹		
SafecfgTemplate	2024/3/7 17:04	文件夹		
syspar	2024/3/7 17:04	文件夹		
CBC_REPORT.log	2024/3/7 17:04	文本文档	2 KB	
LIC_503868146.sys	2024/3/7 17:04	SYS 文件	1 KB	
LIC_VIRTUAL_USE.sys	2024/3/7 17:04	SYS 文件	1 KB	
PxTPU.config.xml	2024/3/7 17:04	Microsoft Edge ...	1 KB	
sc_vsm_metadata.xml	2024/3/7 17:04	Microsoft Edge ...	1 KB	
user.sys	2024/3/7 17:04	SYS 文件	1 KB	
AppCalTest.mod	2023/11/9 15:28	MOD 文件	5 KB	
BaseCalculate.mod	2023/11/9 15:28	MOD 文件	1 KB	
calpos.mod	2023/11/9 15:28	MOD 文件	5 KB	
cnvpos.ndx	2023/11/9 15:28	NDX 文件	1 KB	
Degrade.ndx	2023/11/9 15:28	NDX 文件	1 KB	
fluidtest.mod	2023/11/9 15:28	MOD 文件	4 KB	
GunClean.mod	2023/11/9 15:28	MOD 文件	2 KB	
HA2_BFR_R24.mod	2023/11/9 15:28	MOD 文件	56 KB	
HA2EN_QB_R24.mod	2023/11/9 15:28	MOD 文件	26 KB	
HA2HE_FR_R24.mod	2023/11/9 15:28	MOD 文件	45 KB	
HA3_F_R24.mod	2023/11/9 15:28	MOD 文件	43 KB	
HA3_KG_R24.mod	2023/11/9 15:28	MOD 文件	46 KB	
HA3_R_R24.mod	2023/11/9 15:28	MOD 文件	43 KB	
HKHA_F_R24.mod	2023/11/9 15:28	MOD 文件	66 KB	
HKHA_R_R24.mod	2023/11/9 15:28	MOD 文件	46 KB	
HX_FX_R24.mod	2023/11/9 15:28	MOD 文件	28 KB	

Toolclean (only for opener) - function

Toolclean is a clean tool for hood opener or truck opener after production of some cars (like one or two cars), reducing some quality problems of cars (like drip paint problems).

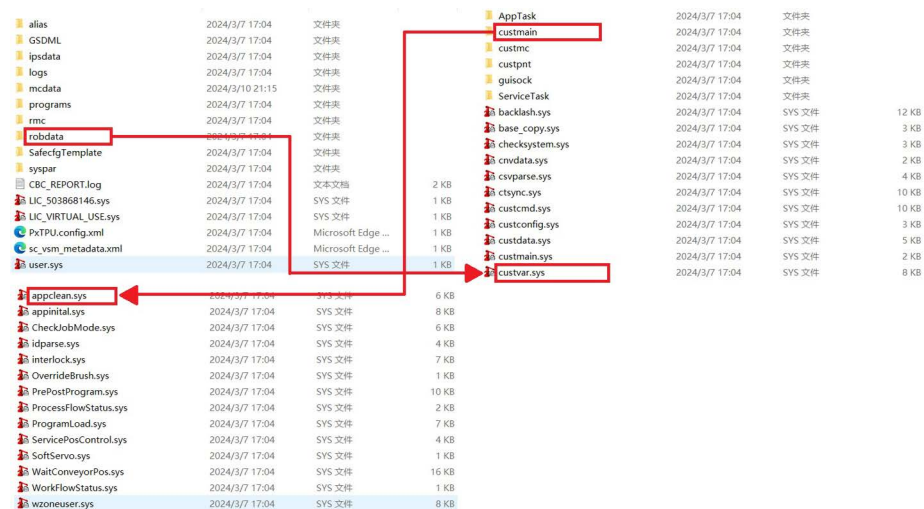
For example:

If user don't clean tool, user can change the change cleaning sequence to 0 or change the clean time to 0 in toolcfn.ndx (file path: Home/mcdata). If user need to, user should set some values of sequence and clean time. User should know the first column is type of car, second column is frequency of production cars, third is the time of toolcfn.



alias	2024/3/7 17:04	文件夹	
GSDML	2024/3/7 17:04	文件夹	
ipsdata	2024/3/7 17:04	文件夹	
logs	2024/3/7 17:04	文件夹	
mcdata	2024/3/10 21:15	文件夹	
programs	2024/3/7 17:04	文件夹	
rmc	2024/3/7 17:04	文件夹	
robdata	2024/3/7 17:04	文件夹	
SafecfgTemplate	2024/3/7 17:04	文件夹	
syspar	2024/3/7 17:04	文件夹	
CBC_REPORT.log	2024/3/7 17:04	文本文档	2 KB
LIC_503868146.sys	2024/3/7 17:04	SYS 文件	1 KB
LIC_VIRTUAL_USE.sys	2024/3/7 17:04	SYS 文件	1 KB
PxTPU.config.xml	2024/3/7 17:04	Microsoft Edge ...	1 KB
sc_vsm_metadata.xml	2024/3/7 17:04	Microsoft Edge ...	1 KB
user.sys	2024/3/7 17:04	SYS 文件	1 KB
c1g1cfn.mod	2024/1/7 15:36		
c1g1fill.mod	2024/1/7 15:36		
mcerror.mod	2024/1/7 15:36		
toolcfn.cfg	2023/11/9 15:28		
toolcfn.mod	2023/11/9 15:28		
toolcfn.ndx	2023/11/9 15:28		

User can call ToolClean.mod through defined some variable value (file path Home/robdata/custvar.sys) and use the function (file path Home/robdata/custmain/appclean.sys).



alias	2024/3/7 17:04	文件夹	
GSDML	2024/3/7 17:04	文件夹	
ipsdata	2024/3/7 17:04	文件夹	
logs	2024/3/7 17:04	文件夹	
mcdata	2024/3/10 21:15	文件夹	
programs	2024/3/7 17:04	文件夹	
rmc	2024/3/7 17:04	文件夹	
robdata	2024/3/7 17:04	文件夹	
SafecfgTemplate	2024/3/7 17:04	文件夹	
syspar	2024/3/7 17:04	文件夹	
CBC_REPORT.log	2024/3/7 17:04	文本文档	2 KB
LIC_503868146.sys	2024/3/7 17:04	SYS 文件	1 KB
LIC_VIRTUAL_USE.sys	2024/3/7 17:04	SYS 文件	1 KB
PxTPU.config.xml	2024/3/7 17:04	Microsoft Edge ...	1 KB
sc_vsm_metadata.xml	2024/3/7 17:04	Microsoft Edge ...	1 KB
user.sys	2024/3/7 17:04	SYS 文件	1 KB
AppTask	2024/3/7 17:04	文件夹	
custmain	2024/3/7 17:04	文件夹	
custmc	2024/3/7 17:04	文件夹	
custpnt	2024/3/7 17:04	文件夹	
guisock	2024/3/7 17:04	文件夹	
ServiceTask	2024/3/7 17:04	文件夹	
backlash.sys	2024/3/7 17:04	SYS 文件	12 KB
base_copy.sys	2024/3/7 17:04	SYS 文件	3 KB
checksystem.sys	2024/3/7 17:04	SYS 文件	3 KB
cnvdata.sys	2024/3/7 17:04	SYS 文件	2 KB
csvparse.sys	2024/3/7 17:04	SYS 文件	4 KB
ctsync.sys	2024/3/7 17:04	SYS 文件	10 KB
custcmd.sys	2024/3/7 17:04	SYS 文件	10 KB
custconfig.sys	2024/3/7 17:04	SYS 文件	3 KB
custdata.sys	2024/3/7 17:04	SYS 文件	5 KB
custmain.sys	2024/3/7 17:04	SYS 文件	2 KB
custvar.sys	2024/3/7 17:04	SYS 文件	8 KB
appclean.sys	2024/3/7 17:04	SYS 文件	6 KB
appinit.sys	2024/3/7 17:04	SYS 文件	8 KB
CheckJobMode.sys	2024/3/7 17:04	SYS 文件	6 KB
idparse.sys	2024/3/7 17:04	SYS 文件	4 KB
interlock.sys	2024/3/7 17:04	SYS 文件	7 KB
OverrideBrush.sys	2024/3/7 17:04	SYS 文件	1 KB
PrePostProgram.sys	2024/3/7 17:04	SYS 文件	10 KB
ProcessFlowStatus.sys	2024/3/7 17:04	SYS 文件	2 KB
ProgramLoad.sys	2024/3/7 17:04	SYS 文件	7 KB
ServicePosControl.sys	2024/3/7 17:04	SYS 文件	4 KB
SoftServo.sys	2024/3/7 17:04	SYS 文件	1 KB
WaitConveyorPos.sys	2024/3/7 17:04	SYS 文件	16 KB
WorkFlowStatus.sys	2024/3/7 17:04	SYS 文件	1 KB
wzoneuser.sys	2024/3/7 17:04	SYS 文件	8 KB

User can change the speed of the robot to go to the position of ToolClean and back home in the ToolClean.mod. User can change the clean speed and times during the clean time in ToolClean.mod.



alias	2024/3/7 17:04	文件夹	
GSDML	2024/3/7 17:04	文件夹	
ipsdata	2024/3/7 17:04	文件夹	
logs	2024/3/7 17:04	文件夹	
mcdata	2024/3/10 21:15	文件夹	
programs	2024/3/7 17:04	文件夹	
rmc	2024/3/7 17:04	文件夹	
robdata	2024/3/7 17:04	文件夹	
SafecfgTemplate	2024/3/7 17:04	文件夹	
syspar	2024/3/7 17:04	文件夹	
CBC_REPORT.log	2024/3/7 17:04	文本文档	2 KB
LIC_503868146.sys	2024/3/7 17:04	SYS 文件	1 KB
LIC_VIRTUAL_USE.sys	2024/3/7 17:04	SYS 文件	1 KB
PxTPU.config.xml	2024/3/7 17:04	Microsoft Edge ...	1 KB
sc_vsm_metadata.xml	2024/3/7 17:04	Microsoft Edge ...	1 KB
user.sys	2024/3/7 17:04	SYS 文件	1 KB
BaseFCalculate.mod	2024/1/7 15:36		
calpos.mod	2024/1/7 15:36		
CCI_EQ_CTOL1.mod	2024/1/7 15:36		
CCI_UR_CTOL1.mod	2024/1/7 15:36		
Degrade.ndx	2024/1/7 15:36		
segments.ndx	2024/1/7 15:36		
SenPosTeach.mod	2024/1/7 15:36		
ToolClean.mod	2024/1/7 15:36		
WobjFCalculate.mod	2024/1/7 15:36		
yAppMaintPos.mod	2024/1/7 15:36		
yBypassPos.mod	2024/1/7 15:36		
yCyclicBrakeCheck.mod	2024/1/7 15:36		
yDressPos.mod	2024/1/7 15:36		
yHomePos.mod	2024/1/7 15:36		
yMaintPos.mod	2024/1/7 15:36		
yManCleanPos.mod	2024/1/7 15:36		
yReferencePos.mod	2024/1/7 15:36		

7.5.2.2 Material Change

Overview

The generic material change model is a state machine which starts with receiving the next material number and ends after the robot has started painting. Material change typically consists of cleaning a number of cavities, opening a valve and then filling these cavities with new paint. Some cavities can be cleaned and then filled while the robot is executing a paint program, some cannot.

When using material number 0 as the material for a job in the job queue, the whole material change process will be skipped. A special material number is reserved to only perform cleaning. By default, this is set to 999, but it can be changed through the configuration file for the material change engine.

Note: Material option 1 to force material change does not affect cleaning.

Cleaning can be graded based on how easy the paint is to clean and how compatible it is with the next paint. The filling can be graded based on viscosity, density, or circulating pressure. If positive displacement (pump) is used to fill, no grading is necessary. The filling and cleaning programs reside under the HOME:/mcdata directory and follow the following naming convention:

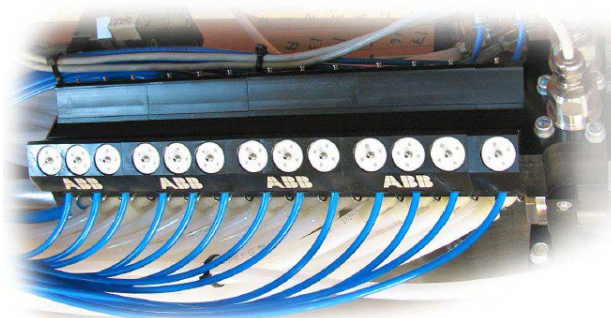
C<Cavity number>G <Grade><Process>.mod

Cavity number is further explained for each colour change type. In a simple single-line system, 1 Grade describes how thoroughly the cavity should be cleaned/filled. The default value is 1. This can be changed by the c<cavity>clean.ndx and the c<cavity> fill.ndx files. Process is clean ('cln') or fill ('fill') processes.

In most material change schemes, it will be important to close valves and set analogue settings to zero or another defined value when the material change process is interrupted. In order to facilitate this, an exception handler is implemented. The module 'mcerror.mod' under HOME:/mcdata will be called, and should contain these instructions.

Single Line 1KP CCV

The single-line colour change is the simplest of the material change processes. Only one cavity is used, and material change happens between programs.



The Single Line 1KP CCV can be used for “1” Cavity or “2” Cavity.

“1” Cavity

Cavity number	Cavity name	Clean Module	Fill Module
1	Single Line	c1g<grade>cln.mod	c1g<grade>fill.mod

“2” Cavity

Cavity number	Cavity name	Clean Module	Fill Module
1	APP Line	c1g<grade>cln.mod	c1g<grade>fill.mod
2	Paint Line	c2g<grade>cln.mod	c2g<grade>fill.mod

Separate cavities for 1KP CCV can be used for pre-start MC since the execution of clean or fill-in Paint line will not go through the needle of the applicator but from the dump line inside the atomizer.

Single Line 1KP RCC

Single Line 1KP RCC uses an RCC colour changer to replace the CCV colour changer.

The rotating colour changer is designed to connect one colour at a time to the applicator. The rotating colour changer is fail-safe; only one paint inlet can be connected to the paint outlet at a time.



Cavity number	Cavity name	Clean Module	Fill Module
1	Mixer	c1g<grade>cln.mod	c1g<grade>fill.mod
2	Resin Line	c2g<grade>cln.mod	c2g<grade>fill.mod
3	Catalyst Line	c3g<grade>cln.mod	c3g<grade>fill.mod

Single Line 2KP CCV

The single-line colour change with two components is an extension of the single line, where two lines supply materials to mix one material for use. Three cavities are used; one for the resin line, one for the catalyst line and one for the mixer.



7 Robot system

7.5 Paint control functions

Cavity number	Cavity name	Clean Module	Fill Module
1	Mixer	c1g<grade>cln.mod	c1g<grade>fill.mod
2	Resin Line	c2g<grade>cln.mod	c2g<grade>fill.mod
3	Catalyst Line	c3g<grade>cln.mod	c3g<grade>fill.mod

Single Line CBS CCV

The cartridge bell system CBS is a rotary atomizing spray coating equipment using cartridges designed to provide an efficient colour change system for both solvent-borne and water-borne paint.

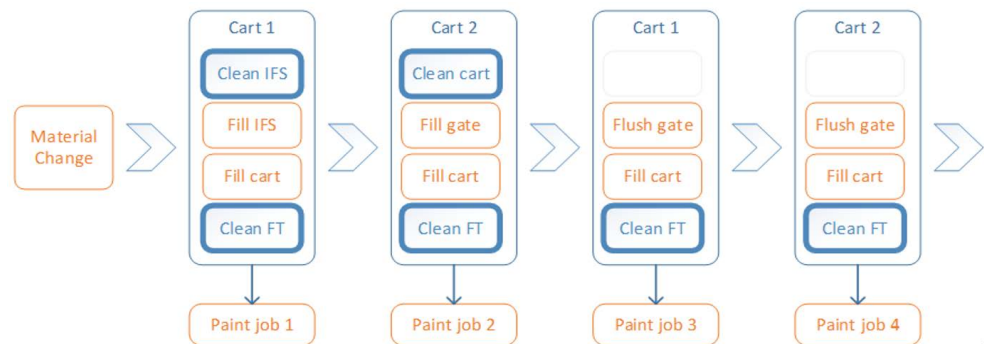
The use of cartridges enables direct charge painting and eliminates the need for paint tube connections to the bell. The CBS robot is equipped with all required cleaning facilities, which guarantee no paint loss and a minimum flushing solvent consumption.



Below is the module and description of CBS mcddata.

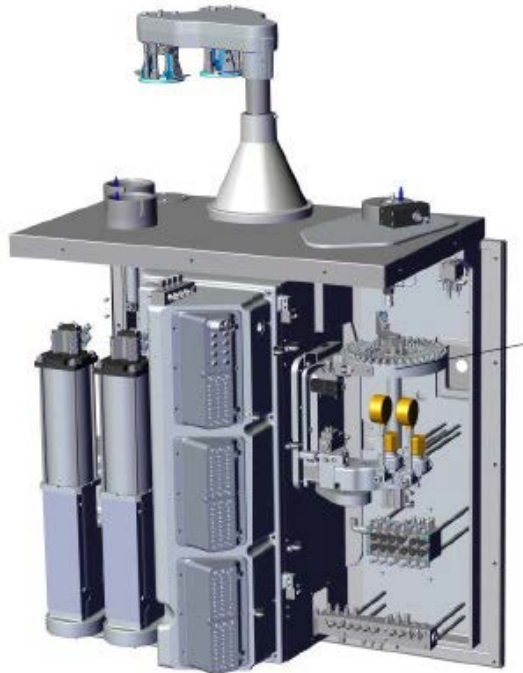
mcddata module	Description
clnifs.mod	Clean all system including colour changer, IFS, cartridge
clncart.mod	Only clean the cartridge
fillifs.mod	Prime the colour changer, IFS, and the gate port of cartridge
fillgate.mod	Prime the inside channel of gate port, inlet and outlet the gate port channel
flushgate.mod	Prime the outside channel of gate port in IFS, keep the gate port close
fillcart.mod	Fill the cartridge with job paint
ftclean.mod	Clean the feed tube of cartridge

The typical running of CBS colour changes with production like below.



Single Line CBS RCC

CBS uses an RCC colour changer to replace the CCV colour changer. The mechanism is the same as CBS, but the colour changer is different to keep fail-safe; only one paint inlet can be connected to the paint outlet at a time.



Dump Clean

The Dump Clean function means the system can clean the dump line of the paint system.

Material change type	Yes or No	Clean Area
1K CCV	Yes	Atomizer dump line
1K RCC	Yes	Atomizer dump line
CBS CCV	Yes	CBS dump line
CBS RCC	Yes	CBS dump line
2KP CCV	No	2K system has no dump line inside atomizer

For dump cleaning, use “dumpcln” . mod in mdata, and it can be called during or after the material change, you can also set the frequency to call the procedure.

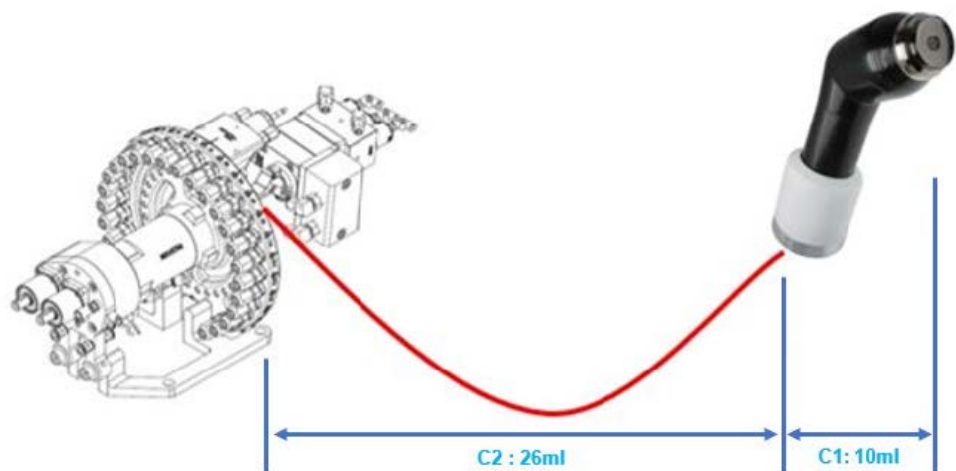
PaintPush

The paint push function is an extension of the standard material change system where an additional function is included to push a given volume of the medium towards the end of a program. The paint push is mainly used to save paint material, but in some cases, the colour change time can also be reduced as the “cleaning” has already started when pushed with solvent. The push medium can be the next paint or solvent. If the paint push has been disabled, the system will use a different clean sequence to avoid incomplete colour change.

If a rotating colour changer is used, the pushing medium can only be solvent. Each program and material combination must run at least once to learn the consumed volume and calculate when the pushing should start. If a different push volume for each material is required, an individual factor or volume can be used. An alarm will be raised if the accumulated fluid is different from the learned volume.

To avoid many combinations, paint push is only used during normal production and not if the production is in repair mode. The paint push can be enabled or disabled by the visualization or any other available client, e.g., TPU.

The following diagram shows the paint push position and volume.



Relevant parameter:

- **Push volume:** The pipe volume + atomizer volume ($c2 + c1$)= 36 ml.
- **Push method:** 0 = No push, 1 = Push paint, 2 = Push solvent.
- **Volumes percentage:** The push percentage of different paint can be separated.

The push method and volume percentage can be set according to different from/to valve combinations by HMI or robot index file.

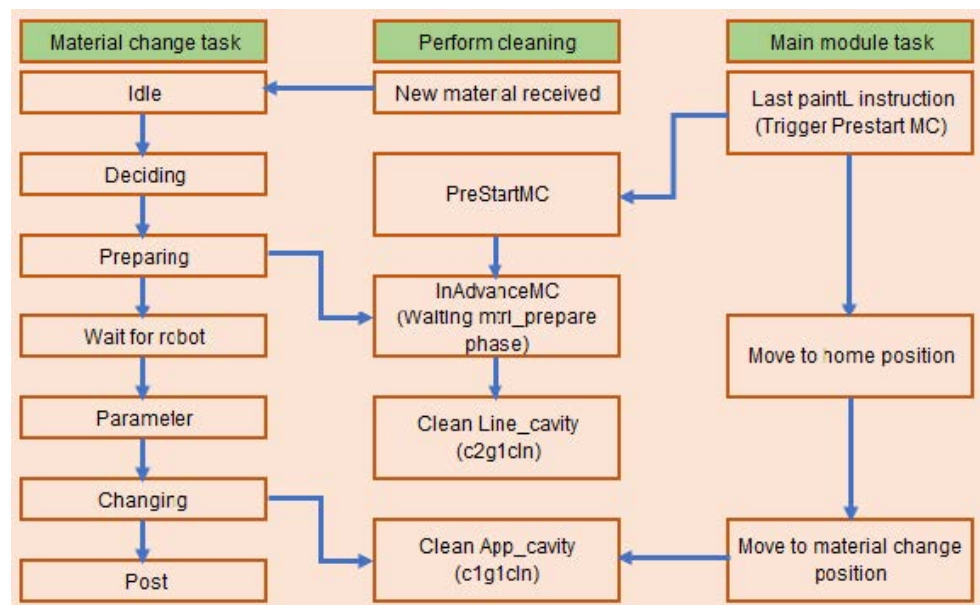
PreStart MC

The material change pre-start is an extension function on the standard material change system where an additional function is included.

The function will start executing the colour change after the last movement before the robot goes back to the home position. The feature makes it possible to reduce the colour change time to a minimum.

Note: This option is only available if the main part of the colour change can be done through an dump valve (as show in the below diagram “Clean line cavity”). e.g. in the atomizer or 2K mixer.

The following diagram shows the process sequence for the Prestart MC function:

**7.5.2.3 Global brush settings**

SetGlobalBrush is used to set service function brush for special functions. When a global brush has been set, the robot will wait for the atom and shape the actual value to reach the set point in a preset time, or an error will arise. If the brush is a cleaning brush, the system will also need to wait for the HV value below a safety level.

Global brush definition

Brush name	Brush number	Description
IdleBrush	901	The idle brush
StartBrush	902	The brush while start robot
HaltBrush	903	The brush while waiting conveyor
MtrlChBrush	904	The brush while material change
StaticBrush	905	The brush during static brush check
DynamicBrush	906	The brush during dynamic brush check
GunClnBrush	907	The brush while gun clean
BellClnBrush	908	The brush while bell clean
HubClnBrush	909	The brush while hub clean
HVDyTestBrush	910	The brush while static HV check
HVStaTestBrush	911	The brush while dynamic HV check
AutoShapeBrush	912	The brush for auto shape during clean mode

7.5.2.4 Volume calculation

Volume calculation uses a Brush tracking mechanism, with Brush tracking files and options.

Brush tracking files are used to record the amount of time each brush is active for a program. These files are later used to calculate the paint consumption by combining it with a brush table. The exact filename is dependent on the BrushTrackFormula system parameter, but by default, this will be the program number.

Volume calculation parameter	Numeric
	1 = Program index
	2 = Material Index
	3 = Option1
	4 = Option2
	5 = Option3

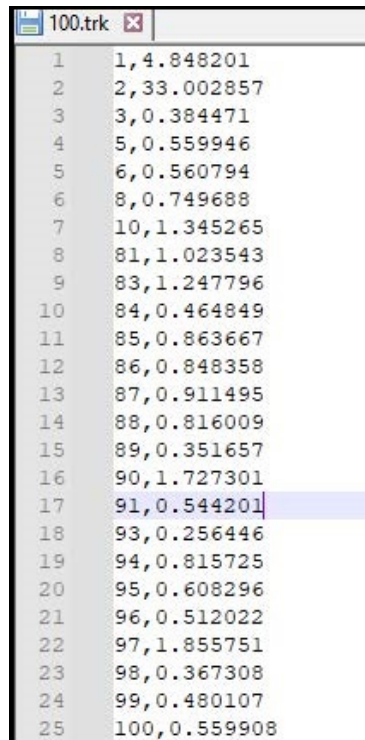
The calculation process is as follows:

Every time the paint brush formula will save the brush applying time in the brush track file. Before next calculate for the paint volume, it will calculate all the brush fluid and brush time during the cycle time.

Calculation volume (cc) =

$(\text{Brush1}(\text{fluid} \times \text{time}) + \text{Brush2}(\text{fluid} \times \text{time}) + \dots + \text{LastBrush}(\text{fluid} \times \text{time})) / (60 \text{ s/min})$

To get the accurate volume for the next paint job.



1	1,4.848201
2	2,33.002857
3	3,0.384471
4	5,0.559946
5	6,0.560794
6	8,0.749688
7	10,1.345265
8	81,1.023543
9	83,1.247796
10	84,0.464849
11	85,0.863667
12	86,0.848358
13	87,0.911495
14	88,0.816009
15	89,0.351657
16	90,1.727301
17	91,0.544201
18	93,0.256446
19	94,0.815725
20	95,0.608296
21	96,0.512022
22	97,1.855751
23	98,0.367308
24	99,0.480107
25	100,0.559908

The configuration in process:

BrushTrackSignal	A1Brush	Name of the EIO signal to monitor with brush tracking.
BrushTrackDirectory	HOME:/btrack/	Location of where to save and load brush tracking data files.
BrushTrackFormula	P,a,b,c	Formula used to calculate brush tracking data files for current program.

7.5.2.5 Application supervision

There are specific application supervision functions for different processes as below.

Paint pressure supervision

The paint pressure sensor is installed by default on the robot process device, normally at the inlet and outlet of the pump, DCU devices etc. The pressure sensor measures a liquid or gas pressure in a process system. The pressure is converted into an electrical signal. The electric signal is normally transferred to the control system via the sensor interface on the fibre optic and sensor interface (FOSI) or the pressure sensor interface board (PSIB).

Solvent pressure supervision

Operating the system without solvent will have a big impact on the colour change performance and, in the worst case, cause damage to the equipment. To ensure that the solvent supply is active.

The following are three methods for solvent state detection:

1. Connect a pressure sensor to the solvent supply. The sensor is monitored and controlled by the robot system or PLC to avoid any process-related operation being performed without solvents.
2. Check the solvent pressure in the material change preparation phase, which is disabled after the material change is finished. If the solvent pressure detected is lower than the setting value. A defined signal, “diSolvPSlow” , will be sent from the PLC.

The solvent pressure setting value can be set by PLC or HMI.

Use the robot pump pressure sensor. Release the solvent pressure before cleaning and then build up the solvent pressure to detect whether the pressure is normal.

Related parameter:

Action	Sensor name	Pressure range	Duration time
Release pressure	A1PumpPSIn (for 1k)	< 1 bar	3 s
Build pressure	A1PumpPSIn (for 1k)	> 4 bar	3 s

Pump shaft supervision

The function includes an inductive sensor on the pump shaft. It is meant to verify that the pump is rotating (e.g. that motor drive shaft and pump shaft are reconnected after service).

Brush table supervision

The function used for supervising the Brush table is loaded successfully, in case there is no brush table load, and the robot will move the path but without painting out.

Valve status supervision	The robot system will supervise the valve open or close status, and if the supervised valve keeps open for a long time and exceeds the maximum time, the robot system will activate the error message and reset the valve automatically.
Applicator enable supervision	The applicator enable check function lets the robot check if the applicator is enabled or not before the program is executed. The function can avoid bypassing the applicator enabled in HMI by mistake. The function can be enabled or disabled in Proc.cfg.

7.5.2.6 Application test

There are specific application test functions for operation from robUI.

Dynamic brush test	When performing the dynamic brush test, the robot will move along the created path and execute the brush event in the predefined positions. The path is overlap. The dynamic brush test can be individually operated for every robot with reproducible spray times. Mode selection and individual parameters need to be adjusted from the visualization ahead of the test. Dynamic brush tests can be used for different purposes: • Creating spray pattern • Check if there are any irregularities in the pattern control • Dynamic delay compensation.
Static brush test	When performing the static brush test, the robot will move to a programmed position and execute a brush event in the predefined position. The static brush test can be individually operated for every robot with reproducible spray times. Mode selection and individual parameters need to be adjusted from the visualization ahead of the test. Static brush tests can be used for different purposes: • Check if there are any irregularities in the pattern control • Contamination in the air cap or shape air-ring.
Fluid flow test	When performing the fluid flow test, the robot will move to a programmed position and execute a timed brush in a predefined position. The material should be collected in a beaker to compare the actual flow rate with the accumulated volume. A correction can be made if the volumetric accuracy is not within the technical specifications for the gear pump.

Volumetric measurement can be operated on every robot to measure the actual flow rate. Individual parameters need to be adjusted from the visualization ahead of the test.

Solvent flow test

When performing the hub clean flow test, the robot will move to a programmed position and execute a timed brush in a predefined position. The material should be collected in a beaker to compare the flow rate with the recommended settings for the actual atomizer. A manual adjustment on the solvent flow regulator is required if the volumetric accuracy is not within the technical specification for the atomizer.

Volumetric measurement can be operated on every robot to measure the actual flow rate. Individual parameters need to be adjusted from the visualization ahead of the test.

High voltage test

The high voltage test includes the dynamic HV test and static HV test.

The dynamic HV test measures the current value for three different positions to make sure the HV device works and determines whether the HV cascade shuts off in case of a short circuit. When executing this test, the robots move from home to the special HV test plate. Before reaching the plate, the high voltage is turned on with a setpoint of 60 kW. Neither the tested current values have a big deviation from the stored current value, and an HV error occurs when the test is successfully completed.

The static HV test starts with a voltage setpoint of 70 kW, then it turns the voltage off and measures the voltage and current values every 0.1 s until 1.2 s have passed. This measured voltage curve is compared to a reference curve. If the measurement at every step is below the corresponding reference value, the test was successful.

Nonmotion mode

The non-motion mode function can be used to simulating the robot's complete running cycle for production with painting but without movement.

This needs to be performed at home position and with a cup below to measure the actual volume of a typical car painting program or with a specific option number.

7.5.2.7 Preventive maintenance functions

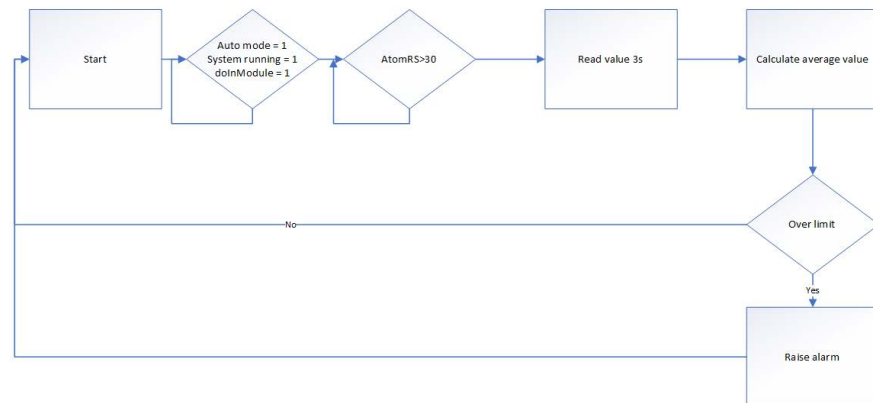
Vibration check function

The vibration check function uses IMS to get a vibration signal to check the air motor and bell cup status. The system will read every 3 s the vibration data when the turbine speed is over 30 krmp; when the value is over the limit value, the system will raise an error message.

The below conditions need to be met for check:

- Robot in Auto mode
- Robot in trajectory program
- Rotation speed over 30 krmp

The vibration amplitude limit value is 20.



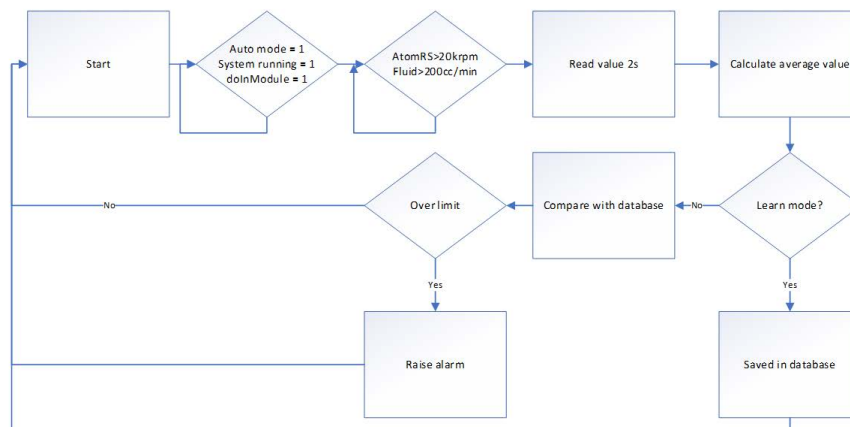
Bell cup missing checking

The bell cup's missing function checks whether the bell cup is installed or not. When the bell cup is missing, the system will raise an alarm to the operator.

The bell cup checking function need learn the reference value first. The variable "bLearnMode" need be set to true to enable learn mode. In learn mode, the system will recode the reference value of Atom IP. and pump pressure out value. In check mode, the system will compare the actual value of Atom IP and pump pressure out value, when the value over limit, the system will raise an error.

The below conditions need to be met for check:

- Robot in Auto mode
- Robot in trajectory program
- Robot rotation speed over 25 krmp
- Robot fluid value over 200 cc/min.



Pump pressure check

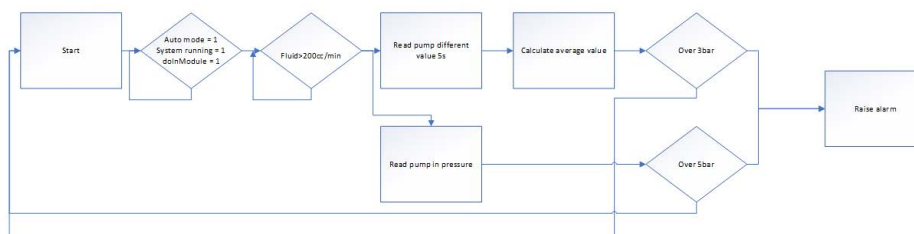
To keep the stable painting, the difference in pump pressure needs to be kept to a small value. This function is to monitor the pressure difference of the pump; when the difference value is over the limit, an error will be raised. The default alarm value is 3 bar.

The gear pump input value also importation for pump lifetime, the system will monitor the pump in pressure alarm, when the alarm over 5 bar, an error will be raised.

When the gear pumps out too big pressure, the system will automatically open the dump valve to release the pressure, and each release will last 100 ms. The valve will be closed when the pump pressure high alarm recovers.

The below conditions need to be met for check:

- Robot in Auto mode
- Robot in trajectory program
- Robot fluid value over 200 cc/min.

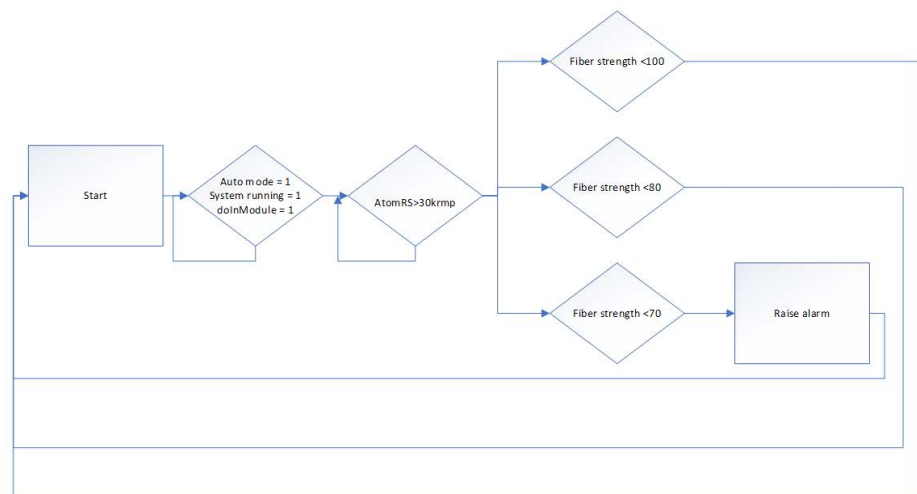


Fiber quality check

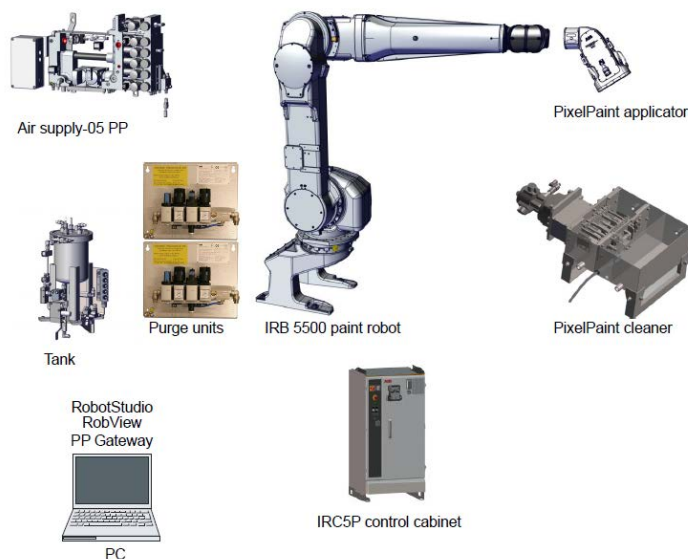
The fibre quality is essentially a parameter for painting. The function will monitor the strength of the fibre; when the light strength is below the limit value, an error will be raised.

The below conditions need to be met for check:

- Robot in Auto mode
- Robot in trajectory program
- Robot rotation speed over 30 krmp.

**7.5.3 PixelPaint Application functions**

The PixelPaint applicator is a non-over spray paint applicator based on the ink-jet principle. The applicator is used for two-tone painting without masking (e.g. black roofs), as well as decor printing (logos, graphics, etc).



The following sub-sections are included in this section:

'Material change' on page 333

'Brush test' on page 336

'Gateway' on page 337

'BK sensor' on page 337

7.5.3.1 Material change

The material change of PixelPaint uses a state machine, which is the logic executing the sequences going from one state to another. It will limit the user from executing a command that is not allowed in that state.

There is no automatic way of sending just one command to end up in "Ready State" in GAP robware.

For project use, the Paint control function updates the GAP material change logic to make it go "Ready State" step by step internally with a simple operation.

Cavity states:

Value	Name	DESCRIPTION
0	UNKNOWN	Unknown
-15	CAV FILLED WATER	Filled water
-16	CAV FILLED COLOR	Filled color
-17	CAV STABILIZED COLOR	Stabilized color
-18	CAV IS READY	Ready
-19	reserve	Status not in use
-20	TRS FILL WATER	Filling water
-21	TRS FILL COLOR	Filling color
-22	TRS STABILIZING COLOR	Stabilizing color
-23	TRS GETTING READY	Getting ready
-24	TRS STOPPING CIRCULATION	Stopping circulation
-25	TRS Closing Nozzle	Closing nozzle
-26	TRS RECOVER CIRCULATION	Recover circulation
-27	TRS FILTER CLEAN	Filter clean
-28	TRS FILTER PRIME	Filter prime
-29	TRS INTEGRITY CHK	Integrity check
-30	TRS FILL TANK	Fill tank



(1)nCavityProcessIndex -- Process Cavity(CPS)

(2)nCavityHeadIndex -- Head Cavity(CHS)

PixelPaint States:

ID	NAME	DISCRIPTION
1	PPS UNKNOWN	Unknown state
2	PPS CLEAN	Clean state
3	PPS PRIMED	Primed state
4	PPS CIRCULATING	Circulating state
5	PPS READY	Ready state
6	PPS_PP_FC_HC	Process primed, filter cleaned, head cleaned, degas cleaned(week clean)
7	PPS_FP_HC	Filter Primed, Head Cleaned(day clean)
8	PPS RECOVERY	Recovery state

nPixelPaintState -- Current logic state

PPS

Command:

INDEX	Description of action
1001	Clean process cavity and head
1002	Prime Process cavity and head
1003	Stabilize circulation (without head nozzles)
1004	Opening valves to circulate paint through print head
1005	Closing valves to circulate through bypass print head
1006	Stop paint circulation
1007	Clean print head
1008	Prime print head
1009	Recover from a pump stop (release pressure and recover)
1010	Clean Filter cavity
1011	Integrity Check sequence when required
1099	Reset PixelPaint state to Unknown
1100	Go Ready State from any state

There are different clean processes for the PixelPaint system. For more information about sequence description, see the product manual - PixelPaint application.

Clean

For simplify the operation, normally day clean/week clean could be use directly after finish the production/test.

Day Clean means the system will clean the print head for daily production/test; this will run the state machine depending on the current state and finally lead to the head clean sequence. That means if the pixel paint system is in circulation or ready state, then it will execute close nozzles, stop circulation, and then do head clean to update the states. If the pixel paint system is already cleaned for the head, then it will just clean again and keep the previous states.

Week clean is similar to day clean, while it mainly checks the filter clean status.

Day Clean

From state	Run sequence	To State
PPS_READY	CloseNozzles StopCirculation HeadClean	PPS_FP_HC
PPS_CIRCULATING	StopCirculation HeadClean	PPS_FP_HC
PPS PRIMED	HeadClean	PPS_FP_HC
PPS_FP_HC	HeadClean	PPS_FP_HC
PPS_PP_FC_HC	HeadClean	PPS_PP_FC_HC

7 Robot system

7.5 Paint control functions

Week Clean

From state	Run sequence	To State
PPS_READY	CloseNozzles StopCirculation HeadClean FilterClean	PPS_PP_FC_HC
PPS_CIRCULATING	StopCirculation HeadClean FilterClean	PPS_PP_FC_HC
PPS_PRIMED	HeadClean FilterClean	PPS_PP_FC_HC
PPS_FP_HC	FilterClean	PPS_PP_FC_HC
PPS_PP_FC_HC	FilterClean	PPS_PP_FC_HC

Prime

The system can go to prime state automatically from different clean statuses.

From state	Run sequence	To State
PPS_FP_HC	PrimeHead	PPS_PRIMED
PPS_PP_FC_HC	PrimeFilter PrimeHead	PPS_PRIMED
PPS_CLEAN	PrimeProcess PrimeHead	PPS_PRIMED

Circulation

The system can go to the circulation state only from the primed state.

From state	Run sequence	To State
PPS_PRIMED	StabilizeCirculation	PPS_CIRCULATING
PPS_CIRCULATING	StopCirculation	PPS_PRIMED

Integrity check

The integrity check is to build and release pressure for both the inlet and outlet unit of the print head, then check the record pressure value of the “A1FluidSupPS” and “A1FluidRtnPS” to compare with the threshold. Check that the sensor is working fine with the process.

From state	Run sequence	To State
PPS_CLEAN	IntegrityCheck	PPS_CLEAN
PPS_PP_FC_HC	IntegrityCheck	PPS_PP_FC_HC

Production

Go to the ready state automatically from any current state; this will execute the internal state machine step by step to get the state of PPS_Ready to print for production or test.

7.5.3.2 Brush test

The pixelpaint brush test is similar to executing operations using conventional commands. But for definite check function.

Nozzle test

The nozzle test is for static printing but check the print head nozzle status. This will print with the staticBrushtest.bmp.



01_staticBrushtest.bmp

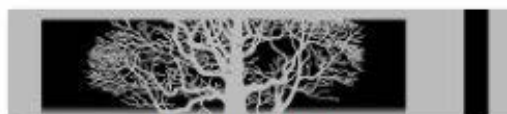
Horizontal test

The horizontal test checks the quality of printing with three images combined. It contains:

- Check the pattern of the print head
- Tree image with complicated detail
- Bpp = 4 to check the bitmap grade of deviation
- Overlap part to check the printing quality.



03_quality_check.bmp



02_quality_check.bmp



01_quality_check.bmp

Vertical test

The vertical test prints the image in a vertical direction but with the same images as the horizontal test. To check the sagging performance of the printing.

Runtime test

The run sheet test is for setting multiple times to print on the same area to check the film thickness, sagging, quality, and so on. It is the same as a statistic test with an image.



01_staticBrushstest.bmp

7.5.3.3 Gateway

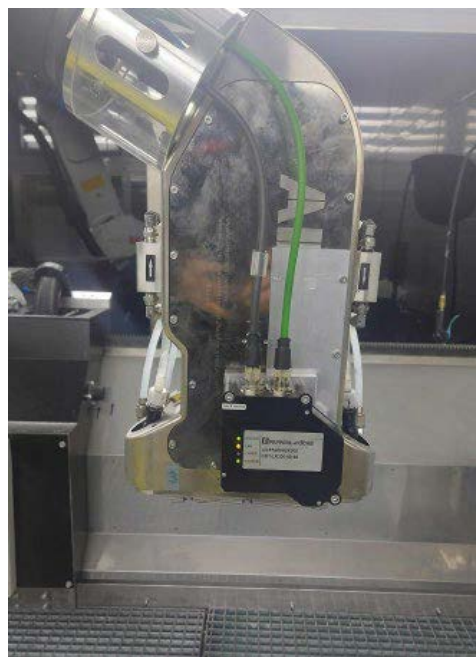
Gateway is a developed tool to integrate the IRC5P controller with meteor software and a PCCE board to automatically send the image from the robot for adaptation to production.

For more information, see the product manuals - pixel paint applications.

7.5.3.4 BK sensor

The IRB 5500 is purpose-built for wet painting applications, demanding precision and accuracy. However, when employing the same paint robot model for specialized tasks like Pixel Paint, additional sensors become essential. To achieve impeccable print quality and precise edges, our setup incorporates both a vision sensor and a BK sensor. Leveraging external sensors to enhance process accuracy is a widely adopted practice.

BK sensor is an option to use line runner to scan the feature of parts. The sensor is mounted on the robot's side plate. It can give the program displacement for path correction during printing.



BK sensor calibration

Calibration for the BK sensor is to measure the BK sensor tool data on the robot. Currently, the PixelPaint system needs multiple BK sensor TCPs for better compensation to the robot in different reorientation poses.

The calibration process is:

Define 12 points with tool0 and World frame:

- X1, X2 (X direction)
- Y1, Y2 (Y direction)
- Z1, Z1 (Z direction)
- RX1, RX2 (Reorient of X direction)
- RY1, RY2 (Reorient of Y direction)
- RZ1, RZ2 (Reorient of Z direction)

Trigger the sensor at these points, check the distances, VMT calculate and feedback the Tool data of BK Sensor mount on robot to the system.

BK sensor check

BK sensor check checks the TCP data of the BK sensor. It scans and measures the points with the definite TCP calibrated and compares the measured position value with the reference position value. If the result matches well with low deviation. Then, the BK sensor mounting position, TCP data, and robot accuracy with the compensation are suitable for printing in this area.

7.5.4 Clean robot functions

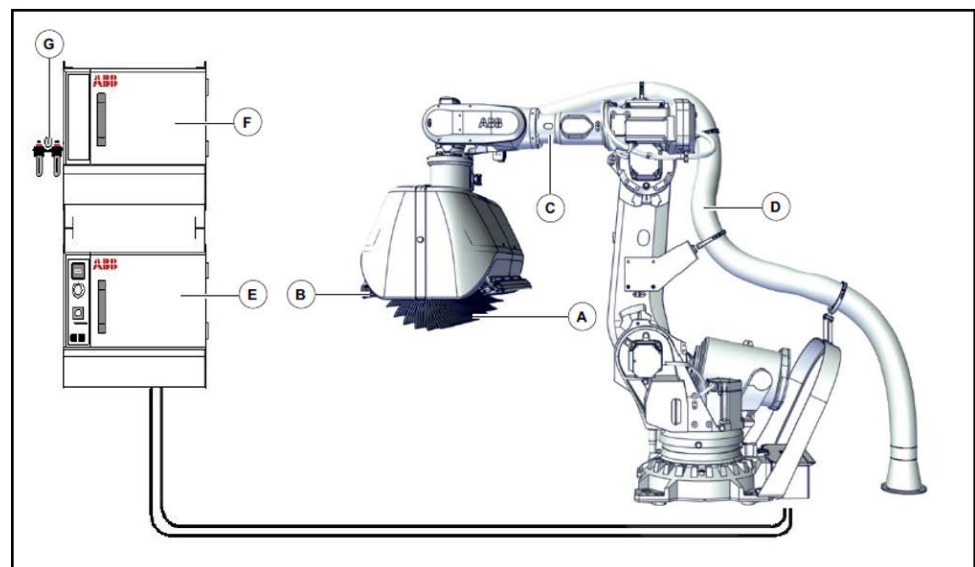
The following sub-sections are included in this section:

'Ionization unit' on page 339

'Exhaust system' on page 340

'Compressed air system' on page 340

Feather Duster V2 is designed to remove contaminants from car bodies and to-be-painted surfaces using rotating ostrich feather brushes. The brushes come across the body shell with an immersion depth of approximately 30 mm to 40 mm. Through this movement, the contaminants are absorbed by the brushes and then vacuumed with an exhaust fan.



Pos.	Description
A	Feather roller
B	Ionization unit
C	Robot
D	Exhaust system controller
E	Controller
F	Process cabinet
G	Compressed air system

7.5.4.1 Ionization unit

Feather Duster V2 is equipped with an ionizer unit to reduce or eliminate static charge from the cleaned surface. The ionizer unit includes an ionizing bar and an ionizing air knife.

7.5.4.2 Exhaust system

Feather Duster V2 works with an exhaust fan to suck and exhaust dust particles from the feathers. The exhaust system of Feather Duster V2 mainly includes a suction slot with two flexible lips, an exhaust manifold, and an exhaust hose.

The 1,016 mm (diameter: 40") long suction slot is installed on the cover and spans the entire length of the feather shaft. The suction slot collects dust and contaminants from the feathers, and then the dust and contaminants are exhausted through the exhaust manifold and exhaust hose.

The exhaust air volume varies according to the arrangement of the exhaust fan location, fan power and ductwork. An airflow of at least 1000 m³/h must be supplied to Feather Duster V2 at the entrance on the manifold.

A flow sensor is integrated into the system to monitor the airflow at the fixed checking point. An alarm (alarm delay: five seconds) is triggered if the airflow is lost or less than the predefined value. The default configured set point of air flow velocity is 5 m/s, which is a hard-coded value and cannot be changed by users.

To ensure that the switch actuates within its published specifications, proper mounting precautions must be taken. The main airflow cavity surrounding the flow thermistor must be orientated perpendicular to the airflow being monitored.

7.5.4.3 Compressed air system

The compressed air system is required to supply air to the ionizing air knife. The compressed air system is provided with a solenoid controlled air valve mounted on the process cabinet. If the air pressure is lower than the setting value, a fault will be generated.

Item	Value
Pressure	3 bar
Filter accuracy	5 µm
Volume	3m ³ /min

7.5.5 Flaming robot functions

Flame type

1. The flame is created by combining and igniting gas (such as natural gas or propane) with oxygen. The size of the flame is regulated by the control cabinet.
2. The adjustment, ignition, and flame size control of the flame head are controlled by the Rapid Flame P104A module.

Burner

The flame head consists of a gas pipeline, a rotary joint (which includes an adapter, balance weight, and other installation materials), and a fixed bracket.

7 Robot system

7.5 Paint control functions

Electrical control cabinet	The electrical control cabinet contains the ignition system, flame detection and diagnostic system, program control system, and manages all operation and monitoring data. It also features a text-based operation interface for reading and an interface module for external communication (such as communication with robots).
Control	The main equipment includes the flame head, control cabinet, high-voltage electric fire module, flame monitoring module, temperature compensation module, flame size selection module, side air input and compression channels, and robot communication module. In this system, the gas system, electrical system, and control module are all independently structured and separated from each other.
Mechanical control cabinet	The mechanical control cabinet houses gas safety valves, pressure switches, air and gas electronic flow meters, electric valves, and a one-way stop valve for the gas and auxiliary combustion air supply unit.
Technical details of Power	- Max. Burner output power: 36 KW - Electric Feeding: 3X380/400V, 50HZ, 16ASI - Max. N.G. consumption: 5 m³/ h NG*/1 per - Max. N.G. consumption: 10 m³/ h NG*/2 per - Gas supply: NG 170~200 mbar (stable)s
The burner (D-6-280)	



The image provided is for reference only. The hose will be installed on the robot arm.

7.5.6 Dispense robot functions

TBD

7.6 Paint control instructions

The following sub-sections are included in this section:

- 'General instructions' on page 342
- 'Paint robot system instructions' on page 356
- 'Pixel Paint robot system instructions' on page 373
- 'Feather duster robot system instructions' on page 376
- 'Sword brush robot system instructions' on page 378
- 'Paint checker robot system instructions' on page 378
- 'Quality check instructions' on page 382
- 'Dispense robot system instructions' on page 385

7.6.1 General instructions

The following sub-sections are included in this section:

- 'PrepareProgram' on page 343
- 'FinishProgram' on page 344
- 'BlockZone - Interlock function block a zone' on page 345
- 'FreeZone - Interlock function free a zone' on page 345
- 'PaintLock - Interlock function with movement' on page 345
- 'BlockCheckZone – Interlock function check and block a zone' on page 346
- 'CheckZone - Interlock function check a zone' on page 347
- 'SetProcessStatus' on page 347
- 'CheckProcessStatus' on page 348
- 'StartWaitConveyorPos' on page 349
- 'RunModeOverride' on page 349
- 'WaitConvPosMove' on page 350
- 'PaintSync' on page 351
- 'WaitWobjCust' on page 351
- 'EndWaitConveyorPos' on page 352
- 'Wait_ExitStart' on page 353
- 'WaitStationOK' on page 353
- 'ReleasePin' on page 354

['WaitStartHome'](#) on page 354

['LoadProgramMod'](#) on page 354

['TeachRobTarget'](#) on page 355

7.6.1.1 PrepareProgram

Description	PrepareProgram is used at the beginning of the trajectory program. The “PrepareProgram” instruction can make the program in good structure and no mistake. It contains: • Load program modules. • Move to Home position. • Move to Ready Position. • Set Ready Brush. • Connect conveyor work object. • Set speed ratio. • Set acceleration ratio. • Set conveyor waiting distance. • Set maximum allowed conveyor position. • Handshake okay between robot and PLC. • Wait for the external start signal. • Switch bypass fluid check.
Example	<pre>PrepareProgram tNormalModule, jhomepos, \vHome:=vpaint1000\jReadyPoint:=jReadyPos\vR eady:=vpaint1000\zReady:=fine\wobj:=wobj_A02\nSpeedRatio:=100\nRelD ist:=nStartPaintCnvPos\nRelDistMax:=nStartPaintCnvPos+200</pre>
Argument	<pre>PrepareProgram (string stSequences{*}, jointTarget jHomePoint \Speeddata vHome \jointTarget jReadyPoint</pre>

```

\Speeddata vReady
\Zonedata zReady
\num nReadyBrush
\pers wobjdata wobj
\num nSpeedRatio
\Num nAccRatio
\num nRelDist
\num nRelDistMax
\Switch CacheMode
\Switch ExitStart
\bool bCheckACCU)

```

7.6.1.2 FinishProgram

Description FinishProgram is used at the end of the trajectory. It contains lots of functions to make the trajectory look like a good structure.

It contains:

- Move back to the home position.
- Drop work object.
- Set stop brush.
- Output the cycle time of the current program.

Example

```

FinishProgram
jhomepos, \vHomeSpeed:=vHFB1000\wobj:=wobj_A02\nStopBrush:=1,
"BC2E_A02";

```

Arguments

```

FinishProgram
(jointTarget jHomePoint
\Speeddata vHomeSpeed
\pers wobjdata wobj
\num nStopBrush,
string stJobName)

```

7 Robot system

7.6 Paint control instructions

7.6.1.3 BlockZone - Interlock function block a zone

Description BlockZone is used for interlock function. The robot will block the zone.

Example BlockZone nExt_Roof;

Arguments BlockZone nZone;

Argument	Data Type	Description
nZone	num	The zone value wants to block

7.6.1.4 FreeZone - Interlock function free a zone

Description FreeZone is used for the interlock function. The robot will free the zone.

Example FreeZone nExt_Roof;

Arguments FreeZone nZone;

Argument	Data Type	Description
nZone	num	The zone value wants to free

7.6.1.5 PaintLock - Interlock function with movement

Description PaintLock is used for the interlock function. The instruction will move the robot to the target and check if the zone is blocked; if yes, the robot will wait until the zone is free, and then the zone will be blocked. If the zone is free during check, it will be blocked directly.

While waiting for the zone free, the brush will be set to wait for brush 903. When a robot target is defined, the robot will move the point and set the wait brush, so there is no need to change the brush to idle brush before this instruction.

Example PaintLock nExt_Roof , p80, vPaint1000, zPaint50,
RB1000EXTL_200\WObj:=wobj_Bora_MQB;

Arguments

PaintLock nZone, ToPoint, Speed, Zone, Tool, \WObj, \RelDist

Argument	Data Type	Description
nZone	num	The zone value wants to block
ToPoint	robtarg	The robot target for painting
Speed	speeddata	The speed for painting
Zone	zonedata	The zone for painting
Tool	tooldata	The tool for painting
\ WObj	wobjdata	The work object for painting
\ RelDist	num	The wait conveyor distance during conveyor stop

Interlock status related variables:

Variable name	Value	Description
nExt_Hood	1	The interlock zone for hood
nExt_R_FD	2	The interlock zone for right front door
nExt_L_FD	3	The interlock zone for left front door
nExt_R_BD	4	The interlock zone for right back door
nExt_L_BD	5	The interlock zone for left back door
nExt_R_FF	6	The interlock zone for right front fender
nExt_L_FF	7	The interlock zone for left front fender
nExt_R_BF	8	The interlock zone for right back fender
nExt_L_BF	9	The interlock zone for left back fender
nExt_SunRoof	10	The interlock zone for sun roof
nExt_Roof	11	The interlock zone for roof
nExt_Trunk	12	The interlock zone for trunk
nExt_R_7Axi	13	The interlock zone for right rail
nExt_L_7Axi	14	The interlock zone for left rail
nExt_L_CoverArea1	15	The interlock zone for left cover
nExt_R_CoverArea2	16	The interlock zone for right cover

7.6.1.6 BlockCheckZone – Interlock function check and block a zone

Description

BlockCheckZone is used for interlock function. The interlock function is to prevent two robots from working in the same area. The interlock function uses opposite logic; 1 indicates free, and 0 indicates blocks. The instruction will check if the zone is blocked; if yes, the robot will wait until the zone is free, and then the zone will be blocked. If the zone is free during check, it will be blocked directly. While waiting for the zone free, the brush will be set to wait for brush 903. When a robot target is defined, the robot will move the point and set the wait brush, so there is no need to change the brush to idle brush before this instruction.

7 Robot system

7.6 Paint control instructions

Example

```
BlockcheckZone nExt_Roof;  
  
BlockcheckZone nExt_Roof \ WaitPoint := P10;
```

Arguments

```
BlockcheckZone nZone \ WaitPoint;
```

Argument	Data Type	Description
nZone	num	The zone value wants to check and block
\WaitPoint	robtarget	The wait position for checking the zone

7.6.1.7 CheckZone - Interlock function check a zone

Description

CheckZone is used for interlock function. The instruction will check if the zone is blocked or free; if the condition does not match, the robot will wait until the zone is free or blocked. While the robot is waiting, the brush will be set to wait for brush 903. When a robot target is defined, the robot will move the point and set the wait brush, so there is no need to change the brush to idle brush before this instruction.

Example

```
CheckZone nExt_Roof;  
  
CheckZone nExt_Roof \ CheckValue := 1 \ WaitPoint := P10;
```

Arguments

```
CheckZone nZone \ CheckValue \ WaitPoint;
```

Argument	Data Type	Description
nZone	num	The zone value wants to check
\CheckValue	num	The want zone value
\WaitPoint	robtarget	The wait position for checking the zone

7.6.1.8 SetProcessStatus

Description

SetProcessStatus is used to mark the paint finish status for different parts. When a part, like a door, is painted, the instruction can be used to mark it as painted and the information will be transferred to PLC. PLC will transfer the information to other robots. The whole station paint process status will be synchronized.

Example

```
SetProcessStatus State_RHoodFinished;
```

Arguments

```
SetProcessStatus nProecssNum;
```

Argument	Data Type	Description
nProecssNum	num	The process state want to set

7.6.1.9 CheckProcessStatus

Description CheckProcessStatus is used to check if the painting of a part is finished or not. If not, the robot will wait until the painting of the check part is finished.

Example CheckProcessStatus State_RHoodFinished;

Arguments CheckProcessStatus nProecssNum;

Argument	Data Type	Description
nProecssNum	num	The process state want to set

Process status related variables:

Variable name	Value	Description
State_RFrontDoorFinished	1	The process finish state for right front door
State_RBackDoorFinished	2	The process finish state for right back door
State_RHoodFinished	3	The process finish state for right hood
State_RTrunkFinished	4	The process finish state for right trunk
State_LFrontDoorFinished	5	The process finish state for left front door
State_LBackDoorFinished	6	The process finish state for left back door
State_LHoodFinished	7	The process finish state for left hood door
State_LTrunkFinished	8	The process finish state for left trunk door

Argument	Data Type	Description
pAboveHook	Robtarget	The robot target above the hook position
HookPoint	Robtarget	The position gripper attach the door
SearchSpeed	Speeddata	The door search speed
tTool	Tooldata	The opener using tool data
\ WObj	WorkObject	The opener using work object
\ nSearchoffset	num	The distance opener searching from hook position
\ UpperLim	num	The search position z direction allowed upper deviation base on hook position
\ LowerLim	num	The search position z direction allowed lower deviation base on hook position
\ Xoffset	num	The increase of start moving distance along x direction when retry search the door
\ Yoffset	num	The increase of start moving distance along y direction when retry search the door
\ Zoffset	num	The increase of start moving distance along z direction when retry search the door
\ checktime	num	The search maximum retry time
\ UpDistance	num	The distance the gripper will move up when searched the door
\ Offset	switch	If the robot will use sensor position to compensation the rest of movement

Argument	Data Type	Description
\ NoStop	Switch	If the robot should stop moving when find the door

7.6.1.10 StartWaitConveyorPos

Description StartWaitConveyeorpos is used in the conveyor wait position function to initial the conveyor track. The conveyor tracking modes “Limit checking mode” and “Update mode” can be set; the mode will override the mode selection in HMI. If the mode parameter is missing, the system will use the HMI setting mode from the conveyor wait position function. A global retract distance in the Z direction can also be set with this instruction.

Example

```
StartWaitConveyorPos nWCP\nRetract_Z := 50 \nMode := nUpdateMode;
StartWaitConveyorPos nWCP\nRetract_Z := 50 \nMode := nCheckMode;
StartWaitConveyorPos nWCP\nRetract_Z := 50 \nMode := nNormalMode;
StartWaitConveyorPos nWCP\nRetract_Z := 50;
```

Arguments StartWaitConveyorPos nTempCnvPos{*}, \ nRetract_Z\ nMode;

Argument	Data Type	Description
nTempCnvPos{*}	Num array{256}	The wait position array input parameter
nRetract	Num	The global retract distance along z direction
nMode	Num	The conveyor tracking mode

7.6.1.11 RunModeOverride

Description RunModeOverride is used to override the conveyor sync mode. There are three modes: nNormalMode, nUpdateMode, and nCheckMode. Generally, the conveyor sync mode will be selected by the HMI job mode setting, but this instruction can be used in the program to override the selected mode.

Example

```
RunModeOverride nNormalMode;    override to normal mode!
RunModeOverride nUpdateMode;    override to position update mode!
RunModeOverride nCheckMode;     override to position check mode!
```

Arguments RunModeOverride Mode;

Argument	Data Type	Description
Mode	Num	The override job mode.

7.6.1.12 WaitConvPosMove

Description

WaitConvPosMove is the instruction to let the robot stop at the specific position when the conveyor stops. The stop point can be located not in the paint quality critical area. The stop point can be defined by the programmer and needs a learning process to save all teach conveyor positions. The position learning and checking can be selected simply on the HMI simulation page.

The “SubEnd” parameter needs to be declared for the last point in a routine. This will be used in bypass mode. I recommend using the instructions “PaintSync” to achieve this function.

Example

```
PaintL p310, vPaint1000, zpaint50, RB1000EXTL_200\Wobj:=Wobj_A26;
```

```
WaitConvPosMove p310;
```

Used after a fly point.

```
PaintL p310, vPaint1000, fine, RB1000EXTL_200\Wobj:=Wobj_A26;
```

```
WaitConvPosMove p310\Zone := fine;
```

Used after a fine point.

```
PaintL p310, vPaint1000, fine, RB1000EXTL_200\Wobj:=Wobj_A26;
```

```
WaitConvPosMove p310\Zone := fine\ SubEnd;
```

Used during the end of routine.

Arguments

Argument	Data Type	Description
WaitPoint	robtarget	The robot wait point when conveyor stop
\Zone	zonedata	The zone data from robot wait point
\nRetract_Z	num	The retract point for this specific wait point
\RelDist	num	The tracking conveyor position for the wait point
\SubEnd	switch	The switch flag to indicate the last wait point in a routine

7.6.1.13 PaintSync

Description PaintSync is used to move the robot quickly from one point to another with a brush and will wait until the point when the conveyor stops.

Example PaintSync p580, vHFB500, zPaint50, RB1000EXTL_200\Wobj:=Wobj_A12;

The robot will move to p580 with speed 500 mm/s, the zone is zPaint50, the tool data is “RB1000EXTL_200”, the work object is “Wobj_A12”. When the conveyor stop before this point, the robot will wait at point “p580”.

PaintSync
p580, vHFB500, zPaint50, RB1000EXTL_200\Wobj:=Wobj_A12\SubEnd;

\SubEnd means this is the last wait position in the routine.

Arguments PaintSync ToPoint, Speed, Zone, Tool \Wobj \RelDist \SubEnd

Argument	Data Type	Description
ToPoint	robtarg	The next point robot will move to
Speed	speeddata	The speed the robot move to next point
Zone	zonedata	The zone data the robot move to next point
Tool	tooldata	The tool data the robot move to next point
\Wobj	wobjdata	The work object the robot move to next point
\nRetract_Z	num	The retract distance that the robot will move during conveyor stop
\RelDist	num	The conveyor waiting distance that robot will keep tracking when conveyor stop
\SubEnd	switch	The flag indicate this is the last instruction in the routine

7.6.1.14 WaitWobjCust

Description WaitWobjCust is a similar function as “WaitWobj”. The function will reset the signal “ConveyorHoldReq” and can check the maximum conveyor position. If the system is using the wait conveyor position function, all “WaitWobj” should be replaced with “WaitWobjCust”. When the conveyor position is over the maximum position, the system will raise an error and stop the robot.

Example WaitWobjCust Wobj_A26\ RelDist:= 3000\ MaxDist := 3200;

Arguments

WaitWobjCust Wobj, \ RelDist \ MaxDist;

Argument	Data Type	Description
Wobj	wobjdata	The workobject need to connect
\RelDist	num	The conveyor waiting distance
\MaxDist	num	The maximum allow conveyor distance

7.6.1.15 EndWaitConveyorPos

Description

EndWaitConveyorPos is used to indicate the end of the conveyor wait in the module. During update mode, the updated array of conveyor wait positions will be saved to a specified array. The module name can also be declared in the instructions to ease the module saving. The variable “nWCPArrayName” will store the correct array name, which can also be declared in the instruction, so there is no need to manually copy the value in array “nWCP” to the destination array.

Example

```
EndWaitConveyorPos nWCP;

Advance solution:

Proc StrategyCheck()

test nPntStrategy

case 0:

test nCarType

case 1:

...

nWCPArrayName := “nL1_A12” ;

...

...

Endproc

EndWaitConveyorPos nWCP \ stModuleName := “L1_BCE_A12_R14” \
StWCPName := nWCPArrayName;
```

Arguments

Argument	Data Type	Description
nTempCnvPos {*}	Array{Num}	The wait position array
\stModuleName	String	The name of current module
\StWCPName	String	The variable that store the array of wait point

Related variables:

nWaitPos {256}: the total wait conveyor array have 256 elements.

The variable name	Index range	Description
nWaitPos	1-150	Wait conveyor positions
nWaitPos	151-200	Each routine conveyor wait positions
nWaitPos	201-256	Conveyor tracking Max tracking distance

nWCPArrayName

The variable name for the conveyor wait position. This variable needs to be assigned the value for the conveyor wait position array.

7.6.1.16 Wait_ExitStart

Description	Wait_ExitStart is used to block the robot program point movement and wait for the start-allowed command from PLC. When the PLC send command 413, the signal “doRobotIsPainting” will be on. Normally, the instructions will be used for the stop-and-go station and painter robot.
Example	Wait_ExitStart;
Arguments	No arguments for this instruction.

7.6.1.17 WaitStationOK

Description	WaitStationOK is used to block the robot's movement and send the robot's prepared ok signal to PLC. PLC will check the whole station's robot status, conveyor and other related conditions; if all conditions are ok, PLC will give feedback on the "Station OK" to the robot. Then, the robot will pass the station check and be ready to paint.
Example	<pre>Moulde XXX ... Proc mainXXX() LoadProgramMod XXX; WaitStationOK; ... Endproc endmodule</pre>

Arguments No arguments for this instruction.

7.6.1.18 ReleasePin

Description ReleasePin is used to send a signal to release the skid lock pin. Normally, the lock pin will be released when all robots finish the entire job; this will waste some cycle time. By using this signal, the lock pin signal can be sent to PLC before the robot program is finished. The function is used to save cycle time.

Example ReleasePin;

Arguments No arguments for this instruction.

7.6.1.19 WaitStartHome

Description WaitStartHome is used to reset the signal “doRobotIsPainting”. The instruction can be put before the robot moves to the home position.

Example WaitStartHome;

Arguments No arguments for this instruction.

7.6.1.20 LoadProgramMod

Description LoadProgramMod is used to load a batch of modules specified in the array. The instruction is used for modular program structure. An optional parameter, “CacheMode”, can be declared, and then the loaded module will not be deleted after the program is executed; this can save the program load time for next time.

Example LoadProgramMod stModuleDegrade;

Load the programs module in the array “stModuleDegrade” ; the modules will be erased if the next job does not need them.

LoadProgramMod stModuleDegrade \ CacheMode;

Load the programs module with cache mode in array “stModuleDegrade” ; the modules won’t be erased in program memory after the program is executed.

Arguments

Argument	Data Type	Description
stModules{*}	Array {string}	The modules name to be load
\CacheMode	Switch	The flag if in cache mode

7.6.1.21 TeachRobTarget

Description

TeachRobTarget is used to quickly teach the service positions. The instruction can save the robot target and joint target based on the input joint target or robot target.

Example

```
TeachRobTarget \ jPoint := jTempHome, jHomePos, pHomePos;
TeachRobTarget \ jPoint := jTempMtrl, jMtrlChPos, pMtrlChPos;
TeachRobTarget \ jPoint := jTempMaint, jMaintPos, pMaintPos;
TeachRobTarget \ jPoint := jTempBypass, jByPassPos, pByPassPos;
```

The above program will quickly teach home, material change, maintenance and bypass position based on jTempHome, JTempMtrl, jTempMaint and jTempByass. The joint values of jTempHome, JTempMtrl, jTempMaint and jTempByass need to be defined by the user in advance.

```
TeachRobTarget \ pPoint:= pTempHome, jHomePos, pHomePos;
TeachRobTarget \ pPoint:= pTempMtrl, jMtrlChPos, pMtrlChPos;
TeachRobTarget \ pPoint:= pTempMaint, jMaintPos, pMaintPos;
TeachRobTarget \ pPoint:= pTempBypass, jByPassPos, pByPassPos;
```

The above program will quickly teach home, material change, maintenance and bypass position based on pTempHome, pTempMtrl, pTempMaint, and pTempByass. The user needs to define the joint values of pTempHome, pTempMtrl, pTempMaint, and pTempByass in advance.

Arguments

TeachRobTarget \ jPoint \ pPoint JTarget, pTarget

Argument	Data Type	Description
\ jPoint	jointtarget	The source of joint target value
\ pPoint	robtarg	The source of robot target value
JTarget	jointtarget	The destination of joint target value
pTarget	robtarg	The destination of robot target value

7.6.2 Paint robot system instructions

The following sub-sections are included in this section:

- 'PaintLSmart' on page 357
- 'BookEscapePoints' on page 358
- 'SmartHome_PushCurrPos' on page 358
- 'SmartHome_Push' on page 358
- 'SmartHome_Pop' on page 359
- 'SmartHome_PoptoAir' on page 359
- 'SmartHome_Clear' on page 359
- 'RunSmartHome' on page 359
- 'DoorRequest - Request a door position' on page 360
- 'DoorCheck – check an opener's position' on page 360
- 'CheckDoorPos - check an opener's position without check master' on page 361
- 'DoorRelease – Release the control of an opener' on page 363
- 'JudgeOpener – Select the executed routine' on page 363
- 'PreStartMC' on page 364
- 'SetGlobalBrush' on page 364
- 'PaintSync' on page 365
- 'WaitPainterRequest - Wait the position command from painter' on page 366
- 'FeedbackPosition – Feedback the door position' on page 367
- 'SearchLin – Search the door' on page 368
- 'PreOpen – Opener functions' on page 370
- 'GripperSensorWatch – Start grip sensor watch' on page 370
- 'GripperSensorSleep – Stop grip sensor check' on page 371
- 'CheckGripperSensor' on page 371
- 'CheckGripperSensorOn' on page 372
- 'CheckGripperSensorOff' on page 372

7.6.2.1 PaintLSmart

Description	PaintLSmart is used to move the robot to the target, and at the same time, the flag signal will be set to indicate the robot's position. The instruction will be used after "BookEscapePoints".
Example	!Book the first smart home array <pre>BookEscapePoints [MQB_RF1, pEmpty, pEmpty, pEmpty, pEmpty, pEmpty] PaintLSmart p10, vPaint450, zPaint50, RB1000EXTL_200\WObj:=wobj_Bora_MQB; PaintL p20, vPaint450, zPaint50, RB1000EXTL_200\WObj:=wobj_Bora_MQB; PaintL p30, vPaint450, zPaint50, RB1000EXTL_200\WObj:=wobj_Bora_MQB; PaintL p40, vPaint450, zPaint50, RB1000EXTL_200\WObj:=wobj_Bora_MQB;</pre> !Book the second smart home array <pre>BookEscapePoints [MQB_RF2, MQB_RF3, pEmpty, pEmpty, pEmpty, pEmpty] PaintLSmart p50, vPaint450, zPaint50, RB1000EXTL_200\WObj:=wobj_Bora_MQB; PaintL p60, vPaint450, zPaint50, RB1000EXTL_200\WObj:=wobj_Bora_MQB; PaintL p70, vPaint450, zPaint50, RB1000EXTL_200\WObj:=wobj_Bora_MQB; PaintL p80, vPaint450, zPaint50, RB1000EXTL_200\WObj:=wobj_Bora_MQB;</pre> - When the robot moved to the point "P10", it will use the first smart home array to move home. - When PP moved over second smart home array, if the robot still not reach the point "P50", it will still use the first smart home array to back home. - When the robot moved to the point "P50", it will use the second smart home array to move home.
Arguments	<pre>PaintLSmart ToPoint, Speed, Zone, Tool \WObj \nRetract_X \nRetract_Y \nRetract_Z\ RelDist;</pre>

7.6.2.2 BookEscapePoints

Description BookEscapePoints is used to save the points target safe back to home when need to cancel the robot painting and move the robot back to the home position. The instruction needs to be used in the trajectory program before the PaintLSmart instruction. Normally, it will used together with PaintLSmart.

Example BookEscapePoints [MQB_RF, pEmpty, pEmpty, pEmpty, pEmpty, pEmpty]

Arguments BookEscapePoints array{6}

7.6.2.3 SmartHome_PushCurrPos

Description SmartHome_PushCurrPos is used to save the current robot target, joint target and active tool into a smart home array. The smart home target will be saved in an array “pEscapeList”. The array has a maximum of 20 elements; each element contains a robot target, a joint target and an active tool. The array works like a stack, first in, last out. The instruction is only used for service positions back home.

Example SmartHome_PushCurrPos;
The current position’s information is saved in smart home array.

Arguments SmartHome_PushCurrPos \ bPushJoint \ bPushPoint;

Argument	Data Type	Description
\ bPushJoint	bool	The flag only saves joint target
\ bPushPoint	bool	The flag only saves robot target

7.6.2.4 SmartHome_Push

Description SmartHome_Push is used to push the input robot target, joint target and active tool into a smart home array. The array works like a stack, first in, last out. The instruction is only used for service positions back home.

Example SmartHome_PushCurr pTarget, jTarget, tool0;

Arguments

Argument	Data Type	Description
pTemp	robtarg	The input value of robot target
jTemp	jointtarget	The input value of joint target
tTemp	tooldata	The input value of tool data

7 Robot system

7.6 Paint control instructions

7.6.2.5 SmartHome_Pop

Description	SmartHome_Pop is used to pop an element from the smart home array “pEscapeList”. The pop value will be saved in our parameters and used to move the robot safely back home. The function is only used in the system module, not in the user program.
Example	EscapeTemp:= SmartHome_Pop();
Arguments	No arguments for this instruction.

7.6.2.6 SmartHome_PoptoAir

Description	SmartHome_PoptoAir is used to pop an element from the smart home array “pEscapeList”. The instruction is only used for service positions back home.
Example	SmartHome_PoptoAir;
Arguments	No arguments for this instruction.

7.6.2.7 SmartHome_Clear

Description	SmartHome_Clear is used to clear all saved service smart home arrays and trajectory smart home arrays.
Example	SmartHome_Clear;
Arguments	No arguments for this instruction.

7.6.2.8 RunSmartHome

Description	RunSmartHome is used to move the robot back to home according to the saved service smart home array or trajectory smart home array. The instruction will be used in the program “HomePos”.
Example	If RunSmartHome() then ... endif

Arguments No arguments for this instruction.

7.6.2.9 DoorRequest - Request a door position

Description DoorRequest is used by a painter robot to request a door/hood/trunk opener robot to open a door/hood/trunk to a position. The instruction won't block the program execution. It contains the parameter of the opener's name and position's name. The system defines the variable name.

Example DoorRequest nD01, nDoor_F_Full_Closed;

The paint robot request front opener to manipulate the front door to full closed position.

Arguments DoorRequest nDoorOpener, nDoorPos;

Argument	Data Type	Description
nDoorOpener	num	The door opener's name
nDoorPos	num	The destination of door opener

7.6.2.10 DoorCheck – check an opener's position

Description A door check is used by a painter robot to check the door/hood/trunk opener's position. The instructor can check two positions at the same time.

For the conveyor tracking station, when the opener is bypassed, the instruction will directly bypass the checking and run a bypass callback declared procedure. In normal mode, a normal callback routine can also be declared executed. When the robot tracks the distance over the allowed value, the robot will send the error to PLC and stop the robot from moving.

Example For Stop&Go station:

DoorCheck nD01, nFullOpenPos;

For tracking line station:

DoorCheck nD01, nDoor_B_Full_Opened;

\ nDoorPos_Sub := nHomePos,

\ bypassSignal := " Bypassed_D11" ,

\MinTracking := 2500

\MaxTracking := 4000

7 Robot system

7.6 Paint control instructions

```
\ CallbackProc := "PaintBackDoor"  
\ CallbackBypassProc := "BypassBackDoor" ;
```

Arguments

For Stop&Go station:

```
DoorCheck nDoorOpener, nDoorPos;
```

For tracking line station:

```
DoorCheck nDoorOpener, nDoorPos ;
```

```
\nDoorPos_Sub
```

```
\ bypassSignal
```

```
\ MinTracking
```

```
\ MaxTracking
```

```
\ CallbackProc
```

```
\ CallbackBypassProc;
```

Argument	Data Type	Description
nDoorOpener	Num	The opener's name
nDoorPos	num	The opener's position
\nDoorPos_Sub	num	Additional opener's position (for tracking line)
\ bypassSignal	string	The signal indicator open is bypassed (for tracking line)
\ MinTracking	num	The minimize tracking distance (for tracking line)
\ MaxTracking	num	The Maximum tracking distance (for tracking line)
\ CallbackProc	string	The routine will run when opener's position is match (for tracking line)
\CallbackBypassProc	string	The routine will run when opener is bypassed (for tracking line)

7.6.2.11 CheckDoorPos - check an opener's position without check master

Description

CheckDoorPos is used by a painter robot to check a door/hood/trunk opener's position for the Stop&Go station. The difference between "CheckDoorPos" and "DoorCheck" is that "CheckDoorPos" will not check if the opener has got mastership or not.

Example

```
CheckDoorPos nD01, nFullOpen;
```

Arguments

CheckDoorPos nDoorOpener, nDoorPos;

Argument	Data Type	Description
nDoorOpener	Num	The opener's name
nDoorPos	Num	The opener's position

Related data: Door position variable for conveyor tracking station.

Variable name	Value	Description
nDoor_F_Full_Opened	1	Front door in full open position
nDoor_F_Half_Opened	2	Front door in half open position
nDoor_F_Half_Closed	3	Front door in half close position
nDoor_F_Full_Closed	4	Front door in full close position
nDoor_B_Full_Opened	5	Back door in full open position
nDoor_B_Half_Opened	6	Back door in half open position
nDoor_B_Half_Closed	7	Back door in half close position
nDoor_B_Full_Closed	8	Back door in full close position
nHoodTrunk_Full_Opened	9	Hood & Trunk in full open position
nHoodTrunk_Half_Opened	10	Hood & Trunk in half open position
nHoodTrunk_Half_Closed	11	Hood & Trunk in half close position
nHoodTrunk_Full_Closed	12	Hood & Trunk in full close position
nHomePos	15	Home position

Door position variable for Stop&Go station:

Variable name	Value	Description
nFullOpenPos	1	Door full open position
nHalfOpenPos	2	Door half open position
nHookPos	3	Door in hook position
nUnHookPos	4	Door in unhook position
nReadyPos	5	Door in ready position
nHomePos	6	Door in home position
nErrHandlerPos	7	Door in error handler position
nErrHookPos	10	Door in error hook position (internal use)
nErrHalfOpenPos	20	Door in error half open position (internal use)
nErrFullOpenPos	30	Door in error full open position(internal use)

Door opener name:

Variable name	Value	Description
nH01	1	Robot open hood/Trunk
nD01	2	Robot located in front along car move direction
nD02	3	Robot located in back along car move direction
nH02	4	Robot open hood/Trunk

7 Robot system

7.6 Paint control instructions

Painter name variables:

Variable name	Value	Description
nR01	1	Robot located in front along car move direction
nR02	2	Robot located in front along car move direction
nR03	4	Robot located in front along car move direction
nR04	8	Robot located in front along car move direction
nR05	16	Robot located in front along car move direction

7.6.2.12 DoorRelease – Release the control of an opener

Description DoorRelease is used to release the control of an opener. The instructions will clean the opener's request for door position value and mastership. The instruction should avoid being used right behind "DoorRequest", as this will reset the previous request directly and cause the request signal can't be transferred to the destination robot.

Example DoorRelease nD01;

Reset the position request and mastership request to the front opener.

Arguments DoorRelease nOpener;

Argument	Data Type	Description
nOpener	num	The opener's name

7.6.2.13 JudgeOpener – Select the executed routine

Description JudgeOpener is used based on the input signal to check which to run which routine. It can be used for interior bypass mode.

Example JudgeOpener
"DiBypassed_H", "HOME_TO_BP01", "Hood_painting", \ReIDist:=5100;

Check which procedure the opener should run based on the check signal "DiBypassed_H". If the signal is on, the procedure "HOME_TO_BP01" will be executed, and at the same time, the robot will wait for the conveyor position to a distance of 5100 mm. If the signal "DiBypassed_H" is off, the procedure "Hood_painting" will be executed.

Arguments

JudgeOpener
BypassSignal,
ProcName1,
ProcName2
\ Reldist;

Argument	Data Type	Description
BypassSignal	string	The name of bypass signal
ProcName1	string	The execute routine when bypass signal is 1
ProcName2	String	The execute routine when bypass signal is 0
\ Reldist	num	The wait position when bypass signal is 1

7.6.2.14 PreStartMC

Description

PreStartMC is used to trigger the material change in the material preparation stage. It can be used in a trajectory program before the robot moves to the home position. The brush needs to be turned off before executing this instruction; normally, a fine point can be put before the instruction.

Example

PaintL p1220, vHFB500, fine, RB1000EXTL_200\Wobj:=Wobj_A12;
PreStartMC;
The material change will start after the robot reaches point p1220.

Arguments

No arguments for this instruction.

7.6.2.15 SetGlobalBrush

Description

SetGlobalBrush is used to set service function brush for special functions. When a global brush has been set, the robot will wait for the atom and shape the actual value to reach the set point in a preset time, or an error will arise. If the brush is a cleaning brush, the system will also need to wait for the HV value below a safety level.

Example

SetGlobalBrush "BellClnBrush";
The system will load the global brush "BellClnBrush", which is 908 according to the predefined configuration.

7 Robot system

7.6 Paint control instructions

Arguments

SetGlobalBrush stBrushParam \ NoWait;

Argument	Data Type	Description
stBrushParam	string	The brush no need to set
\NoWait	Switch	The flag if need wait the actual value reach set point

Related variable:

Brush name	Brush number	Description
IdleBrush	901	The idle brush
StartBrush	902	The brush while start robot
HaltBrush	903	The brush while waiting conveyor
MtrlChBrush	904	The brush while material change
StaticBrush	905	The brush during static brush check
DynamicBrush	906	The brush during dynamic brush check
GunClnBrush	907	The brush while gun clean
BellClnBrush	908	The brush while bell clean
HubClnBrush	909	The brush while hub clean
HVDyTestBrush	910	The brush while dynamic HV check
HVStaTestBrush	911	The brush while static HV check
AutoShapeBrush	912	The brush for auto shape during clean mode

7.6.2.16 PaintSync

Description

PaintSync is used to move the robot quickly from one point to another with a brush and will wait until the point when the conveyor stops.

Example

PaintSync p580, vHFB500, zPaint50, RB1000EXTL_200\Wobj:=Wobj_A12;

The robot will move to p580 with a speed of 500 mm/s; the zone is zPaint50, the tool data is “RB1000EXTL_200”, and the work object is “Wobj_A12”. When the conveyor stops before this point, the robot will wait at point “p580”.

PaintSync

p580, vHFB500, zPaint50, RB1000EXTL_200\Wobj:=Wobj_A12\SubEnd;

\SubEnd means this is the last wait position in the routine.

Arguments

PaintSync ToPoint, Speed, Zone, Tool \WObj \RelDist \SubEnd

Argument	Data Type	Description
ToPoint	robtarget	The next point robot will move to
Speed	speeddata	The speed the robot move to next point
Zone	zonedata	The zone data the robot move to next point
Tool	tooldata	The tool data the robot move to next point
\WObj	wobjdata	The work object the robot move to next point
\nRetract_Z	num	The retract distance that the robot will move during conveyor stop
\RelDist	num	The conveyor waiting distance that robot will keep tracking when conveyor stop
\SubEnd	switch	The flag indicate this is the last instruction in the routine

7.6.2.17 WaitPainterRequest - Wait the position command from painter

Description

WaitPainterRequest is used to wait for the position command from a painter. The robot program point will block until the position is received. The instruction can accept a single painter or two paints with a “+” in the middle.

A backup robot can also be declared; when the main robot is bypassed, the backup robot will take over the control. A maximum tracking distance can be set; the robot will have an error when the opener can't get the correct position before the conveyor reaches the limit.

The parameter “AutoTracking” is used when required robot bypassed, the opener will automatically skip checking the position when conveyor reaches the setting distance, if “AutoTracking” diRobotdefine, the robot will use learned value as auto-tracking value. The auto skip distance is a conflict with the parameter of “Backup robot”. When “Backup robot” is defined, the system will only wait for a command from the bootup selected robRoboten the selected robot is bypassed.

Example

```
WaitPainterRequest nR01, nHoodTrunk_Full_Closed;
```

```
WaitPainterRequest nR01, nHoodTrunk_Full_Closed\ MaxTracking := 3000;
```

```
WaitPainterRequest nR01+ nR02, nHoodTrunk_Full_Closed;
```

```
WaitPainterRequest nR01+ nR02, nHoodTrunk_Full_Closed\nBackup:=nRightPainter;
```

```
WaitPainterRequest nR01+ nR02, nHoodTrunk_Full_Closed\ AutoTracking:= 5000;
```

7 Robot system

7.6 Paint control instructions

Arguments

WaitPainterRequest

nRobot,

nReqPos

\ nBackup

\ ForbidenBypassAll

\ MinTracking

\ MinTrackingBackup

\ MaxTracking

\ MaxTrackingBackup

\ AutoTracking

Argument	Data Type	Description
nRobot	num	The selected robot name
nReqPos	num	The required input position
\ nBackup	num	The selected backup robot
\ ForbidenBypassAll	Switch	Not allowed bypass all selected robots
\ MinTracking	num	Minimum tracking distance
\ MinTrackingBackup	num	Minimum backup tracking distance
\ MaxTracking	num	Maximum tracking distance
\ MaxTrackingBackup	num	Maximum backup robot tracking distance
\ AutoTracking	num	Auto skip checking tracking distance

7.6.2.18 FeedbackPosition – Feedback the door position

Description

FeedbackPosition is used to give feedback on the door position when the opener opens or closes the door. The feedback door position will be set to the painter.

Example

FeedbackPosition \nDoor, nDoor_B_Full_Closed;

Arguments

FeedbackPosition nPos;

Argument	Data Type	Description
\Door	num	The car body position
nPos	num	The opener feedback position

7.6.2.19 SearchLin – Search the door

Description	SearchLin is used by the opener to search the door position. When the door is found, the position when the sensor is on will be stored. The opener can keep moving or stop based on the parameter setting. The search distance and search limit can be set to filter unwanted sensors in a situation.
Example	Tracking station: <pre>SearchLin pStartHook, pTouchHook, v300, tRBDDoorOpener\Wobj:=wobj_DC\nSearchoffset:=0\NoStop;</pre> Robot will move to pStartHook position and move to search target with speed v300, tool tRBDDoorOpener and wobj_DC work object.. The search target is nSearchoffset plus pTouchhook in Z direction. Because of present “NoStop”, the robot will not stop when gripper sensor trigger, robot will keep moving until it reach to target position. Stop&GO station: <pre>SearchLin pStartHook, pTouchHook, v40, tRBDDoorOpener\Wobj:=wobj_DC\nSearchoffset:=15 \UpperLim:=40\LowerLim:=30\Xoffset:=-10\Yoffset:=0\Zoffset:=0\checktime:=0\UpDistance:=-5\Offset;</pre> Robot will move to pStartHook position and move to search target with speed v40, tool tRBDDoorOpener and wobj_DC work object. The search target is nSearchoffset plus pTouchhook in Z direction. Because of not present “NoStop”, the robot will stop when gripper sensor trigger and store the position which have the sensor on. Then the robot will move up the distance according to “upDistance” and then stop. If the store position Z value out of the limit “UpperLim” or “LowerLim”, an error will raise. When an error happen during search process, robot will retry based on parameter of “Checktime”, and retry position will offset based one “Xoffset”, “Yoffset”, “Zoffset”.

7 Robot system

7.6 Paint control instructions

Arguments

SearchLin startPoint,
HookPoint,
SearchSpeed,
tTool
\ WObj
\ searchoffsetX
\ searchoffsetY
\ searchoffsetZ
\ UpperLim
\ LowerLim
\ offsetX
\ offsetY
\ offsetZ
\ checktime
\ UpDistance
\ Offset
\ NoStop
\ CheckAfter

Argument	Data Type	Description
startPoint	robtarg	The start robot target before searching.
HookPoint	robtarg	The touch position which just have the gripper sensor is on.
tTool	tooldata	The tool data during searching.
\ WObj	wobjdata	The work object data using during searching.
\ searchoffsetX	num	The search offset Xdirection base on HookPoint.
\ searchoffsetY	num	The search offset Y direction base on HookPoint.
\ searchoffsetZ	num	The search offset Z direction base on HookPoint.
\ UpperLim	num	The upper limit value for store position compare with hookpoint.
\ LowLim	num	The low limit value for store position compare with hookpoint.
\ offsetX	num	The offset for retry in X-direction.
\ offsetY	num	The offset for retry in Y-direction.

Argument	Data Type	Description
\ offsetZ	num	The offset for retry in Z-direction.
\ Checktime	num	The allowed retry time when have searching error.
\ UpDistance	num	The distance moving up when searching movement finished
\ Offset	switch	The flag show if the frame offset base on store position and hookpoint.
\ NoStop	switch	The flag show if the robot should stop or not when gripper sensor on.
\ CheckAfter	switch	The flag use move! first, then check the sensor, normally used for tracking station.

7.6.2.20 PreOpen – Opener functions

Description PreOpen is used for an opener to open a door/hood/trunk to a certain position before an order. There are three positions supported that is “Hook”, “Half Open”, and “Full Open” position. The default position is the hook position. If there is an error occurs during the opening of the door/hood/trunk, the robot will execute the error handler procedure to move the robot to the error position. This instruction is used for the Stop&Go station.

Example `preopen\ndoorsetpos:=nhookpos;`
Allow opener moving to hook position before receiver order from opener.

Arguments PreOpen \ nDoorSetPos;

Argument	Data Type	Description
\nDoorSetPos	num	The destination position of pre-open.

7.6.2.21 GripperSensorWatch – Start grip sensor watch

Description GripperSensorWatch is used to start monitoring the gripper sensor. Once the monitor is active, the robot will keep watching the sensor signal; when the sensor is missing, the robot will stop and raise an error until the sensor signal is back. Normally, the instruction can be used together with MoveLSync to avoid wasting cycle time with a fine-point solution.

Example `MoveLSync pTouchHook, v800, fine, tHoodTrunkOpener, "
GripperSensorWatch ";`

Arguments No arguments for this instruction.

7.6.2.22 GripperSensorSleep – Stop grip sensor check

Description	GripperSensorSleep is used to stop monitoring the gripper sensor. Normally, the instruction can be used together with MovLSync to avoid wasting cycle time with a fine-point solution.
Example	<pre>MovLSync pOnClose, v800, fine, tHoodTrunkOpener, " GripperSensorSleep ";</pre>
Arguments	No arguments for this instruction.

7.6.2.23 CheckGripperSensor

Description	CheckGripperSensor is used as an opener to check if the gripper is holding a door/hood/truck. There are two modes for checking: in the first mode, the robot will move and then check the sensor value, and in the second mode, the robot will only check the sensor without moving. For the first mode, the opener can also move to check position and then move back to the original position with flexible distance, tool and work object frame; the moving speed is fixed at 500 mm/s. When the gripper sensor does not meet the required value, you can set if the error can be reset. For the second mode, the opener will check if the gripper sensor met require value. If not, the robot will be blocked and raise an error. The error can be defined can be reset or not with the input parameter.
Example	Mode 1: <pre>CheckGripperSensorpOnHalf, tRBDDoorOpener\WObj:=wobj_Amarok_DC\nDown0 ffset:=-1\KeepCheckOn;</pre> The robot will move to pOnHalf position and move downward -1 mm, then it will check if the gripper sensor is on. After checking, the robot will keep the position. Mode 2: <pre>CheckGripperSensor\CheckValue := 1;</pre> The robot will check if the gripper sensor is 1. <pre>CheckGripperSensor\CheckValue := 0;</pre> The robot will check if the gripper sensor is 0;

Arguments

CheckGripperSensor \robtarg startPoint\ robtarg checkPoint\pers
 tooldata tTool\pers wobjdata WObj\ num nDownOffset\switch
 KeepCheckOn\Switch bCheckOnce\ num CheckValue;

Argument	Data Type	Description
\ startPoint	robtarg	The original position robot will move to.
\ checkPoint	robtarg	The check position robot will move to.
\ tTool	tooldata	The tool data using during moving.
\ WObj	wobjdata	The work object data using during moving.
\ nDownOffset	num	The down distance during moving.
\ KeepCheckOn	num	If the robot need move back to original position. If not present, will need move back, if not present, robot will keep the current position as final position.
\ bCheckOnce	switch	If the check error can be reset or not when sensor signal missing. If not present, the error can't be reset until the sensor is ok. If present, the error only show one time and can be reset even the sensor is not ok.
\ CheckValue	num	The sensor value to check.

7.6.2.24 CheckGripperSensorOn

Description

CheckGripperSensorOn is used to check if the gripper sensor is 1. If not, the robot will stop moving and raise an error. The error can't be reset until the sensor signal is on. The instruction is a simple form of instruction "CheckGripperSensorOn".

Example

CheckGripperSensorOn;

Arguments

No arguments for this instruction

7.6.2.25 CheckGripperSensorOff

Description

CheckGripperSensorOff is used to check if the gripper sensor is 0. If not, the robot will stop moving and raise an error. The error can't be reset until the sensor signal is off. The instruction is a simple form of instruction, "CheckGripperSensorOff".

Example

CheckGripperSensorOff;

Arguments

No arguments for this instruction

7.6.3 Pixel Paint robot system instructions

The following sub-sections are included in this section:

- 'BK_INIT' on page 373
- 'BK_Start' on page 373
- 'BK_Ende' on page 374
- 'BKScanL' on page 374
- 'BKScanC' on page 374
- 'BK_SensorTrig' on page 375
- 'BK_DoCorrection' on page 375
- 'BK_ResetErrors' on page 376
- 'BK_ResetData' on page 376

The instructions for using Pixel Paint are identical to those for a standard paint robot system.

7.6.3.1 BK_INIT

Description	BK_INIT used to check the initializing important variables and signals and prepare to start.
Example	BK_INIT;

7.6.3.2 BK_Start

Description	BK_Start used to start the BK sensor work.		
Example	BK_Start 255, 1, BK_MODE_TCP;		
Arguments	BK_Start TypeID, PathID, MODE;		
	Argument	Data Type	Description
	TypeID	num	Type number
	PathID	num	Path number
	MODE	num	BK sensor mode.

7.6.3.3 BK_End

Description BK_Start used to stop the BK sensor work.

Example BK_End 12;

Arguments BK_End PointCount;

Argument	Data Type	Description
PointCount	num	Totally point counts number during this BK job.

7.6.3.4 BKScanL

Description BKScanL used to work for linear robot movements with BK sensor measurement trigger event.

Example BKScanL p30, vHFB20, zpaint1, tBK_Sensor\Wobj:=wobj_Honda;

Arguments BKScanL \num PointID, \num SensorID, \switch Group, robtarget point, speeddata speed, zonedata zone, INOUT tooldata tool \INOUT wobjdata wobj;

Argument	Data Type	Description
\PointID	num	ID of measurement point.
\SensorID	num	ID of BK sensor.
\Group	num	Group of measurement.
Point	Robtarget	The position robot will move to.
Speed	Speeddata	The speed robot moving.
Zone	Zonedata	The zone robot moving.
Tool	Tooldata	The tool of robot moving.
\wobj	Wobjdata	The work object frame using for robot moving.

7.6.3.5 BKScanC

Description BKScanC used to work for circular robot movements with BK sensor measurement trigger event.

Example BKScanC p20, p30, vHFB20, zpaint1, tBK_Sensor\Wobj:=wobj_Honda;

7 Robot system

7.6 Paint control instructions

Arguments

BKScanC \num PointID, \num SensorID, \switch Group, robtarget point1, robtarget point2, speeddata speed, zonedata zone, INOUT tooldata tool \INOUT wobjdata wobj;

Argument	Data Type	Description
\PointID	num	ID of measurement point.
\SensorID	num	ID of BK sensor.
\Group	num	Group of measurement.
Point1	Robtarget	Cirpoint of robot moving.
Point2	Robtarget	To point of robot moving
Speed	Speeddata	The speed robot moving.
Zone	Zonedata	The zone robot moving.
Tool	Tooldata	The tool of robot moving.
\wobj	Wobjdata	The work object frame using for robot moving.

7.6.3.6 BK_SensorTrig

Description

BK_SensorTrig used to trigger the BK sensor without a movement.

Example

BK_SensorTrig 1,1;

Arguments

BK_SensorTrig (num PointID,num SensorID,\switch Group)

Argument	Data Type	Description
\PointID	num	ID of measurement point.
\SensorID	num	ID of BK sensor.
\Group	num	Group of measurement.

7.6.3.7 BK_DoCorrection

Description

BK_DoCorrection used to get the correction with nPointID parameter.

Example

BK_DoCorrection,1;

Arguments

BK_DoCorrection(num nPointID)

Argument	Data Type	Description
nPointID	num	ID of measurement point.

7.6.3.8 BK_ResetErrors

Description	BK_ResetErrors used to reset the BK sensor errors.
-------------	--

Example	BK_Reseterrors;
---------	-----------------

7.6.3.9 BK_ResetData

Description	BK_ResetData used to reset the BK sensor data.
-------------	--

Example	BK_ResetData;
---------	---------------

7.6.4 Feather duster robot system instructions

The following sub-sections are included in this section:

['FDMoveL'](#) on page 376

['SetFDSpeed'](#) on page 377

['StopFDDevice'](#) on page 378

7.6.4.1 FDMoveL

Description	FDMoveL is used for feather duster robots. The robot moves to the target and can set the feather duster rotation speed at the same time.
-------------	--

Example	<pre>FDMoveL p1, vPaint300, zPaint50, Tool_Cleaner\Wobj:=Wobj_SA2FC; FDMoveL p1, vPaint300, zPaint50, Tool_Cleaner\Wobj:=Wobj_SA2FC\nFDSpeed :=60; FDMoveL p1, vPaint300, zPaint50, Tool_Cleaner\Wobj:=Wobj_SA2FC\ RelDist:=1000;</pre>
---------	---

Arguments

Stop&GO:

FDMoveL robtarget ToPoint, speeddata Speed, zonedata Zone,
pers tooldata Tool, \pers wobjdata WObj\ num nFDSpeed

Argument	Data Type	Description
ToPoint	robtarget	The position robot will move to.
Speed	speeddata	The speed robot moving.
Zone	zonedata	The zone robot moving.
Tool	tooldata	The tool of robot moving.
\ WObj	wobjdata	The work object frame using for robot moving.
\ nFDSpeed	num	The feather duster rotation speed.

Conveyor tracking:

FDMoveL robtarget ToPoint, speeddata Speed, zonedata Zone, pers
tooldata Tool, \pers wobjdata WObj\ num RelDist\ num nFDSpeed

Argument	Data Type	Description
ToPoint	robtarget	The position robot will move to.
Speed	speeddata	The speed robot moving.
Zone	zonedata	The zone robot moving.
Tool	tooldata	The tool of robot moving.
\ WObj	wobjdata	The work object frame using for robot moving.
\ RelDist	num	The conveyor related distance.
\ nFDSpeed	num	The feather duster rotation speed.

7.6.4.2 SetFDSpeed

Description

SetFDSpeed is used to set feather duster rotation speed.

Example

SetFDSpeed 60;

Arguments

SetFDSpeed nSpeed

Argument	Data Type	Description
nSpeed	num	The feather duster rotation speed.

7.6.4.3 StopFDDevice

Description StopFDSpeed is used to stop feather duster rotation speed.

Example StopFDDevice;

Arguments No arguments for this instruction

7.6.5 Sword brush robot system instructions

The following sub-section is included in this section:

['SwordMoveL'](#) on page 378

7.6.5.1 SwordMoveL

Description SwordMoveL is used for the sword brush robot. The robot moves to the target.

Example SwordMoveL p1, vPaint300, zPaint50, Tool_Cleaner\WObj:=WObj_SA2FC;

Arguments SwordMoveL robtarget ToPoint, speeddata Speed, zonedata Zone, pers tooldata Tool, \pers wobjdata WObj

Argument	Data Type	Description
ToPoint	robtarget	The position robot will move to.
Speed	speeddata	The speed robot moving.
Zone	zonedata	The zone robot moving.
Tool	tooldata	The tool of robot moving.
\ WObj	wobjdata	The work object frame using for robot moving.

7.6.6 Paint checker robot system instructions

The following sub-sections are included in this section:

['IRYS_Message'](#) on page 379

['IRYS_MeasureL'](#) on page 380

['StartMeasurement'](#) on page 381

['EndMeasurement'](#) on page 381

7 Robot system

7.6 Paint control instructions

7.6.6.1 IRYS_Message

Description IRYS_Message is used for the Das Nano IRYS paint thickness measure system. The instruction can send a command to the IRYS system to trigger different operations.

Example IRYS_Message IRYS_Ref_Measurement, "123456", "T_R011", "KTL_P01_010E_C01" ;

IRYS_Message IRYS_Ref_Measurement, "123456", "T_R011", "KTL_P01_010E_C01" \ SimpleName := " 01" ;

Arguments IRYS_Message num nMode, string CarID, string PointName, string colour\pers string SimpleName \string stCustTag \string stCustData

Argument	Data Type	Description
nMode	num	The position robot will move to.
CarID	string	The speed robot moving.
PointName	string	The zone robot moving.
Colour	string	The tool of robot moving.
\ SimpleName	string	The work object frame using for robot moving.
\ stCustTag	string	The customized tag value for IRYS message
\ stCustData	string	The customized data for IRYS message

IRYS Mode	Description
IRYS_Ref_Measurement	Reference panel measurement
IRYS_Normal_Measurement	Normal mode measurement
IRYS_Sim_Measurement	Simulation mode measurement
IRYS_Health_Measurement	Health mode measurement
IRYS_Modeler_Measurement	Reserved
IRYS_Cancel_Measurement	Cancel measurement
IRYS_Info	IRYS information message
IRYS_Calibrate_Position	Calibration position

Point name format:

T_L011

T_R011

- T is the car type ID, defined together with Das Nano and customer
- L/R is the point location in the car body, L means left side, R means right side.
- 011 is the point ID.

Colour name format:

KTL_P01_010E_C01

- KTL is an e-coat ID
- P01 is the primer coat colour ID
- 010E is the base coat colour ID
- C01 is a clear coat colour ID.

7.6.6.2 IRYS_MeasureL

Description	IRYS_Message is used for the Das Nano IRYS paint thickness measure system. The instruction can send a command to the IRYS system to trigger different operations.
Example	IRYS_MeasureL T_R011, v200, z10, IRYS_Tool_100_1\WObj:=wobj0;
Arguments	IRYS_MeasureL robtarget pTarget, speeddata vSpeed, zonedata Zone, pers tooldata ptool, \pers wobjdata WObj\num nMode\Switch IgnoreCorr\Switch IgnoreNameCheck

Argument	Data Type	Description
pTarget	robtarget	The position robot will move to.
vSpeed	speeddata	The speed robot moving.
Zone	zonedata	The zone robot moving.
ptool	tooldata	The tool of robot moving.
\ WObj	wobjdata	The work object frame using for robot moving.
\ nMode	num	The IRYS measure mode
\ IgnoreCorr	Switch	-
\ IgnoreNameCheck	switch	-

The parameter “nMode” refers to the “IRYS_Message”. If not mentioned here, the default normal mode will be used.

The parameter “IgnoreCorr” is used to bypass the correction of the IRYS positioning system. Normally, it should not bypass the correction.

The parameter “IgnoreNameCheck” is used to check if the IRYS result has the same point name, if the information does not match.

7 Robot system

7.6 Paint control instructions

7.6.6.3 StartMeasurement

Description StartMeasurement is used for the Das Nano IRYS paint thickness measure system. The instructions can initial the test status, data and reference measurements. The movement will use work object wobj0. If the system needs a reference measurements, the robot will go to do reference measurements first.

Example StartMeasurement M1_Ref_point, IRYS_Tool_100_1;

Arguments StartMeasurement robtarget pTarget, pers tooldata ptool\switch ignoreCheckRef

Argument	Data Type	Description
pTarget	robtarget	The position robot will move to.
ptool	tooldata	The tool of robot moving.
\ ignoreCheckRef	wobjdata	The work object frame using for robot moving.

7.6.6.4 EndMeasurement

Description EndMeasurement is used for the Das Nano IRYS paint thickness measure system. The instruction shows the measurement job has been done. Normally used at the end of the program.

Example EndMeasurement SEW_316_T";

Arguments EndMeasurement string stModuleName

Argument	Data Type	Description
stModuleName	string	The position robot will move to.

7.6.7 Quality check instructions

The following sub-sections are included in this section:

['PrepareBYKMeasurement'](#) on page 382

['Mac_MeasureL'](#) on page 383

['Wave_MeasureL'](#) on page 384

['EndBYKMeasurement'](#) on page 385

7.6.7.1 PrepareBYKMeasurement

Description PrepareBYKMeasurement is used for BYK MAC and WAVE measurement devices. The instructions can initial the test status, data, and reference measurements.

Example PrepareBYKMeasurement nNormalMode, nBYK_All;

Arguments PrepareBYKMeasurement num nBYKMode, num nBYKDevice

Argument	Data Type	Description
nBYKMode	num	The position robot will move to.
nBYKDevice	num	The tool of robot moving.

BYK Mode:

BYK Mode	Description
nNormalMode	The normal mode for measurement
nCalibMode	The calibration mode for MAC
nMasterPannel	The master panel mode
nDailyCheck	The daily check mode

Device select:

Device	Description
nBYK_All	Enable both Mac and Wave
nBYK_Mac_Only	Enable Mac only
nBYK_Wave_Only	Enable Wave only

7 Robot system

7.6 Paint control instructions

7.6.7.2 Mac_MeasureL

Description Mac_MeasureL is used for the BYK measure system for MAC measurement.

Example
Mac_MeasureL
1, " Hood ", P10, v100, z10, BYK_Tool1_100_1\WObj:=wobj0\nUpDistance
:=100;

Arguments
Mac_MeasureL num nPointNr, string Check_Zone, robtarget
pTarget, speeddata vSpeed, zonedata Zone, pers tooldata ptool, \pers
wobjdata WObj\num nUpDistance \Switch TeachMode \Switch BypassCorr

Argument	Data Type	Description
nPointNr	num	The position robot will move to.
Check_Zone	string	The zone name sent to BYK.
pTarget	robtarget	The robot target need measurement.
vSpeed	speeddata	The robot moving speed.
Zone	zonedata	The zonedata using for measurement movement.
ptool	tooldata	The TCP data for measurement device.
\ WObj	wobjdata	The work object data for measurement move.
\ nUpDistance	num	The upward moving distance after measurement done.
\ TeachMode	switch	The flag for teach mode selection
\ BypassCorr	switch	The flag for bypass sensor positioning correction.

The parameter “nPointNr” refers to the point number, the value should be increased by 1 for each measurement point.

The parameter “Check_Zone” is used to send check zone data to BYK system. The maximum string length is four characters.

The parameter “nUpDistance” is used to set the move up distance after the measurement finishes.

The parameter “TeachMode” is used to set the running mode to teach. The teach mode is to teach the correction sensor value.

The parameter “BypassCorr” is used to bypass the sensor correction. Normally it should not used.

7.6.7.3 Wave_MeasureL

Description	Wave_MeasureL is used for the BYK measure system for WAVE measurement.
Example	Wave_MeasureL 1, "Hood", P10, P20 \ pTarget3 := P30, v100, z10, BYK_Tool_100_1\WObj:=wobj0\nUpDistance :=100;
Arguments	Wave_MeasureL num nPointNr, string Check_Zone, robtarget pTarget1, robtarget pTarget2, \robtarget pTarget3, speeddata vSpeed, zonedata Zone, pers tooldata ptool, \pers wobjdata WObj \num nUpDistance\Switch TeachMode \Switch BypassCorr

Argument	Data Type	Description
nPointNr	num	The position robot will move to.
Check_Zone	string	The speed robot moving.
pTarget1	robtarget	The target1 robot moving to.
pTarget2	robtarget	The target2 robot moving to.
\ pTarget3	robtarget	The target3 robot moving to (Option parameter).
vSpeed	speeddata	The robot moving speed.
Zone	zonedata	The zonedata using for measurement movement.
ptool	tooldata	The TCP data for measurement device.
\ WObj	wobjdata	The work object data for measurement move.
\ nUpDistance	num	The upward moving distance after measurement done.
\ TeachMode	switch	The flag for teach mode selection
\ BypassCorr	switch	The flag for bypass sensor positioning correction.

The parameter “nPointNr” refers to the point number, the value should be increased by 1 for each measurement point.

The parameter “Check_Zone” is used to send check zone data to the BYK system. The maximum string length is four characters.

The parameters “Target1, Target2, and Target3” are used during wave scan measurement; at least two points need to be used; if the surface is complicated, the third parameter “Target3” can be used.

The parameter “nUpDistance” is used to set the move-up distance after the measurement finishes.

The parameter “TeachMode” is used to set the running mode to teach. The teach mode is to teach the correction sensor value.

The parameter “BypassCorr” is used to bypass the sensor correction. Normally, it should not be used.

7 Robot system

7.7 Troubleshooting

7.6.7.4 EndBYKMeasurement

Description	EndBYKMeasurement is used to measure the BYK system. The instruction shows the measurement job has been done. Normally used at the end of the program.
Example	EndBYKMeasurement ;
Arguments	No arguments for this instruction.

7.6.8 Dispense robot system instructions

TBD

7.7 Troubleshooting

The following sub-sections are included in this section:

['General information'](#) on page 385

['How to save diagnostics information'](#) on page 388

['Troubleshooting with log file'](#) on page 397

['Troubleshooting with e-log'](#) on page 398

7.7.1 General information

PaintControl supports four types of event log messages:

Type	Description
Information	These messages are used to log information into the event log but do not require specific actions from the user. Informational messages do not focus on a display device for the controller.
Warning	These messages are used to remind the user that something is not necessarily correct in the system, but that operation continues. These messages are placed into the event log but do not focus on a display device.
Error with robot stop	These messages indicate something is seriously wrong with the system, and that operation has stopped. These are used when the user should take immediate action.
Error without robot stop	These messages indicate something is seriously wrong with the system, but the operation can still continue. These are used when the user should take immediate action.

Note: The message type is not contained in the actual message text but in the code itself when the message is generated. Messages can only appear as one type, and the message should be written to provide information about the type.

Note: Event number series

The event messages are divided into the following groups depending on which part of the robot system it refers to.

Number series	Error source	Type of events
1 xxxx	Robot	Operational events - events dealing with handling of the system.
2 xxxx	Robot	System events - events dealing with system functions, system states, and so on.
3 xxxx	Robot	Hardware events - events dealing with system hardware, manipulators as well as controller hardware.
4 xxxx	Robot	Program events - events dealing with RAPID instructions, data, and so on.
5 xxxx	Robot	Motion events - events dealing with the control of the manipulator movements and positioning.
6 xxxx	Robot	Not used
7 xxxx	Robot	I/O events - events dealing with inputs and outputs, data buses, and so on.
8 xxxx	Robot	User events - events defined by the user.
9 xxxx	Robot	Functional safety events - events related to functional safety.
10 xxxx	Robot	Not used
11 xxxx	Robot	6000 - 6999 FlexControl events
12 xxxx	Robot	Configuration events - events dealing with the configuration of the system.
13 xxxx	Robot	Paint
14 xxxx	Robot	Rapid
15 xxxx	Robot	Connected Service Embedded event logs which are generated during starting, registering, unregistering, losing connectivity, and so on.

The following chapters describe different error scenarios and how the system reacts to them. The chapters are split by application type and each chapter takes a look at the robot and the PLC reaction separately.

Error Information

Error	Robot reaction	PLC reaction
General robot error	Stop execution and set info bit to PLC: RobotError = 1	Stop robot and conveyor
Emergency stop	Stop robot and program execution	Set emergency stop signal to the robots
Air supply error	Stop execution and set info bit to PLC: AppError = 1	Stop robot and conveyor
Conveyor stop	No reaction	Warning message, no reaction
Paint rapid error	Stop execution and set info bit to PLC: RobotError = 1	Stop robot and conveyor
Gripper error	Stop execution and set info bit to PLC: RobotError = 1	Stop robot and conveyor
Application error	Stop execution and set info bit to PLC: AppError = 1	Stop robot and conveyor
Material supply error	Stop execution and set info bit to PLC: AppError = 1	Stop robot and conveyor
High voltage error	Stop execution and set info bit to PLC: AppError = 1	Stop robot and conveyor

7.7.2 How to save diagnostics information

The following sub-sections are included in this section:

'Collect diagnostic information' on page 388

'Create a system diagnostics file' on page 390

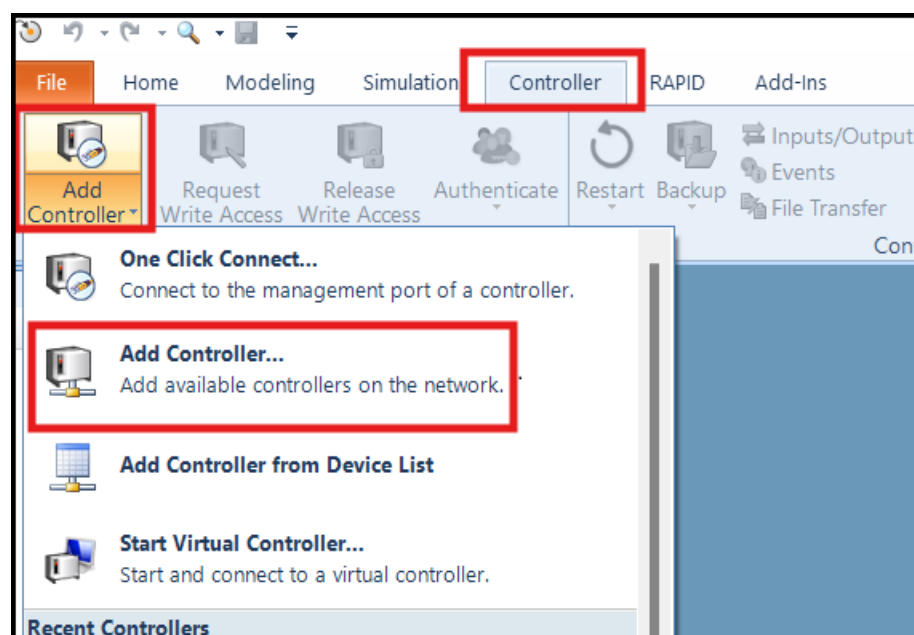
7.7.2.1 Collect diagnostic information

Instruction:

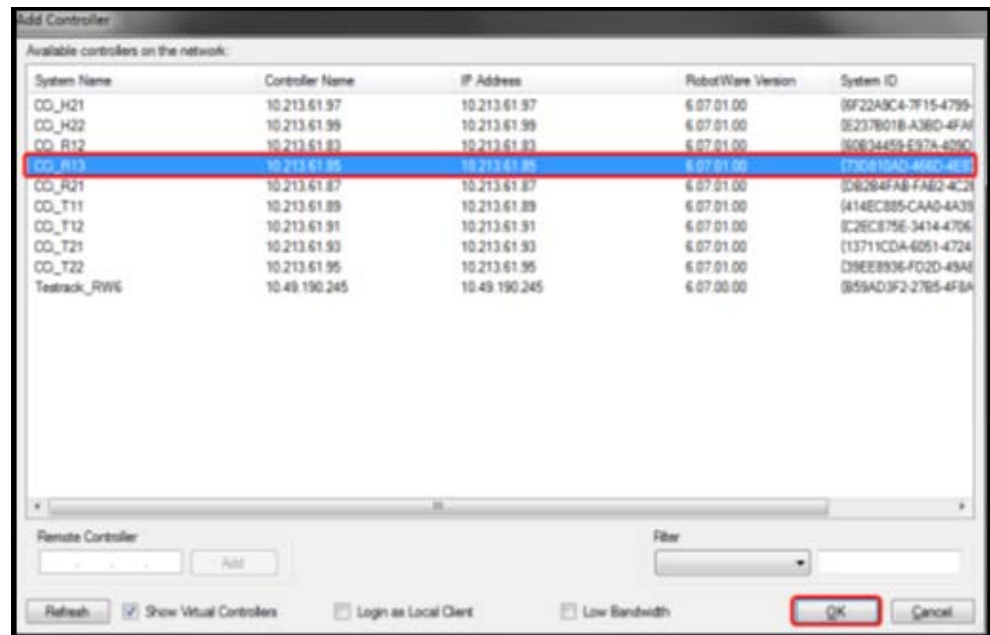
1. Take a photo of the problem on-site.
2. Record a video while the problem happens.
3. Record the program point and motion point while the problem happens.
4. Save the event log by RobotStudio in the Controller tab.
5. Make a photo or screenshot of the error messages displayed on the HMI.
6. Make a note of the exact date and time when the problem occurred.
7. Create a system diagnostic file with RobotStudio or the TPU.
8. Save the signal monitor with Robview if needed.

Connect controller with RobotStudio

1. Start **RobotStudio**.
2. Click on Controller tab.
3. Click the arrow next to the "Add Controller".
4. Click on "Add Controller".

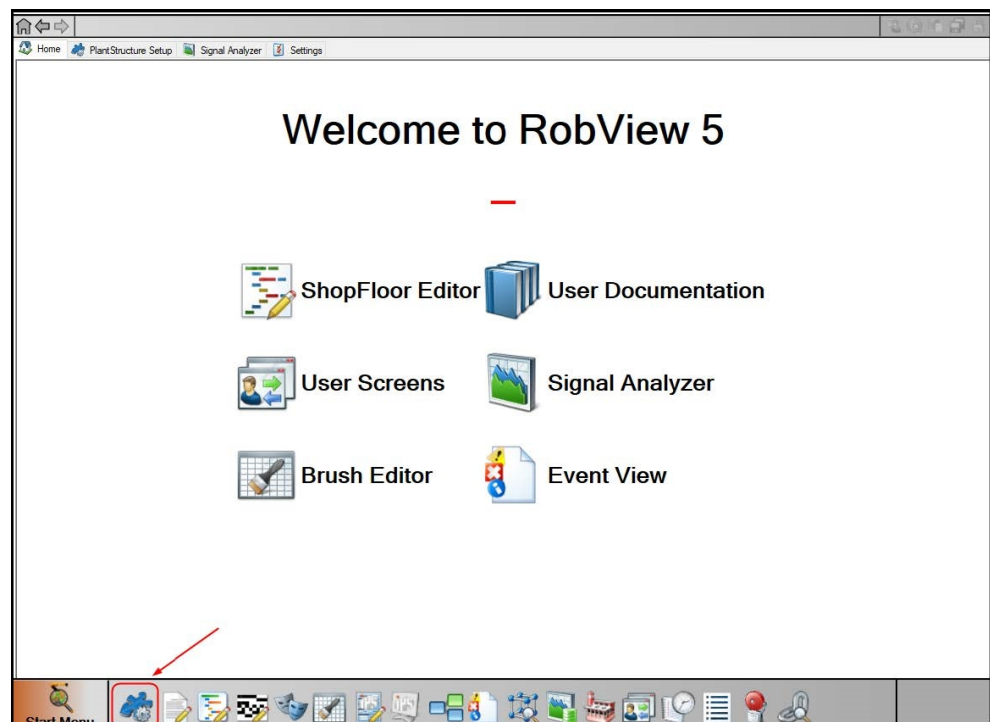


5. Select a robot to connect with.

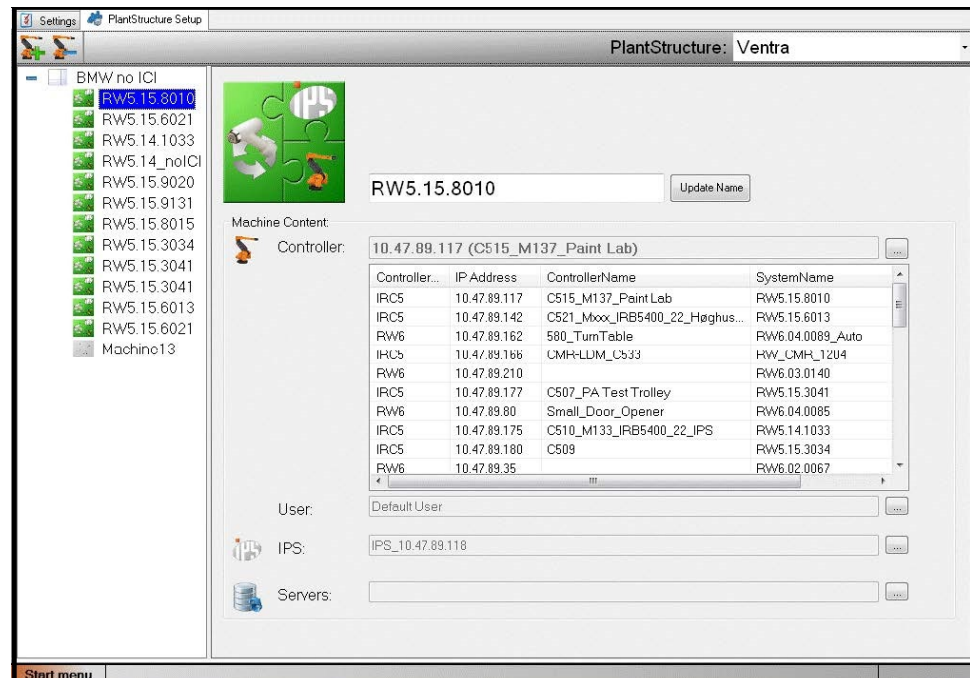


Connect robot with Robview:

1. Start Robview.
2. Go to the plant Structure tab.



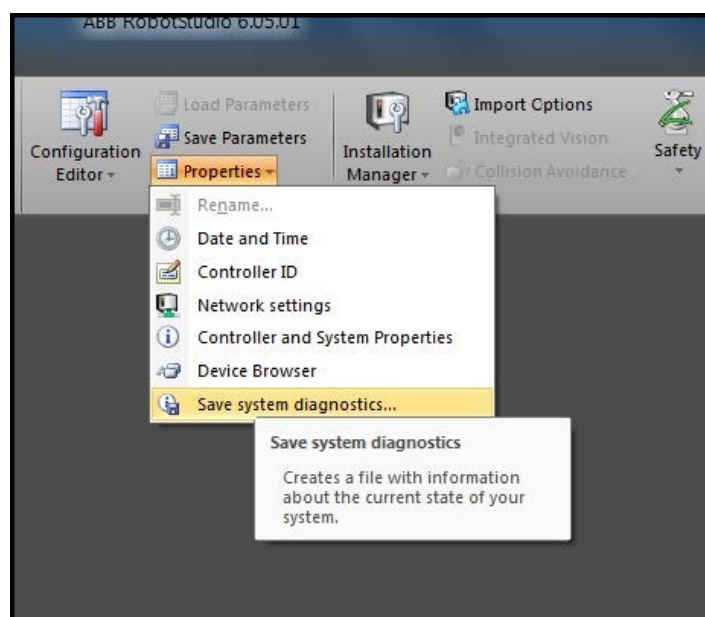
3. By clicking on any of the machines in the left panel, you get the option of adding or removing any machine and associated IPS nodes connected to the network. If the icon becomes green, it means the connection is ok.



7.7.2.2 Create a system diagnostics file

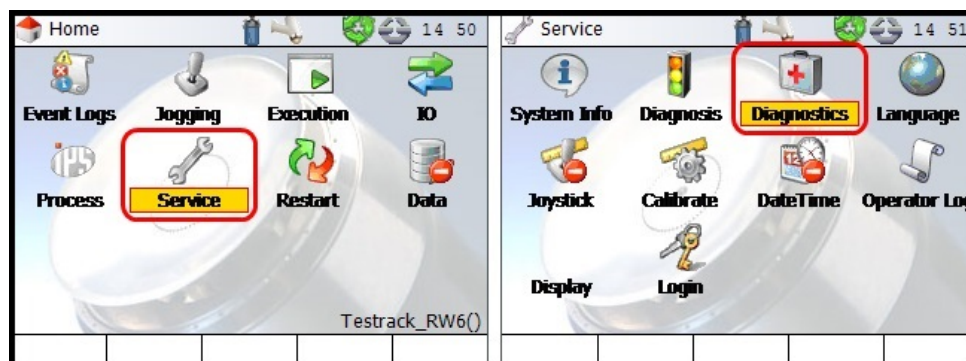
1. With RobotStudio

- Connect the robot with the RobotStudio. See the description in the section '[Connect controller with RobotStudio](#)' on page 388.
- Under "Properties", click on "Save system diagnostics". A diagnostics file will then be stored on the local computer.

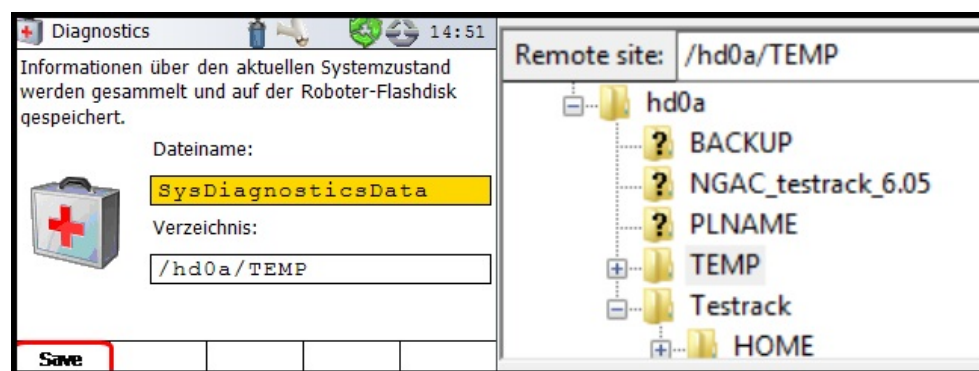


2. With the paint TPU

- Open the menu Service → Diagnostics.



- By clicking on "Save", a system diagnostic file will be saved to the robot flash disk under "/hd0a/TEMP".



3. With main computer console

- Once a connection has been established with a terminal, enter the command: `sysdmp_trigger`.

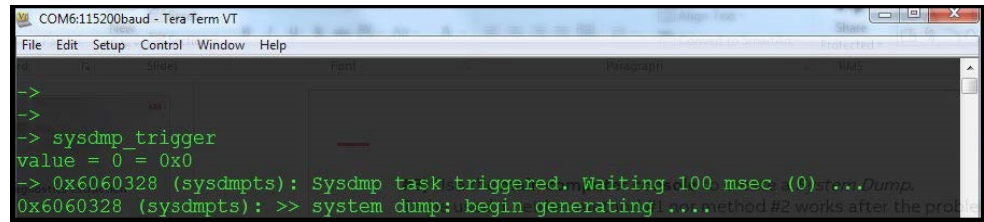
The console should respond with something like the following:

```
0x897e7f0 (sysdmpts): Sysdmp task triggered. Waiting for 100 msec (0) ...
```

```
0x897e7f0 (sysdmpts): >> system dump: begin generating ....
```

```
0x897e7f0 (sysdmpts): << system dump: generated in  
'/hd0a/<system>/INTERNAL/SYSDMP/0' .
```

- Restart the controller later.
- Use TPU or RobotStudio to save the diagnostic file.



```

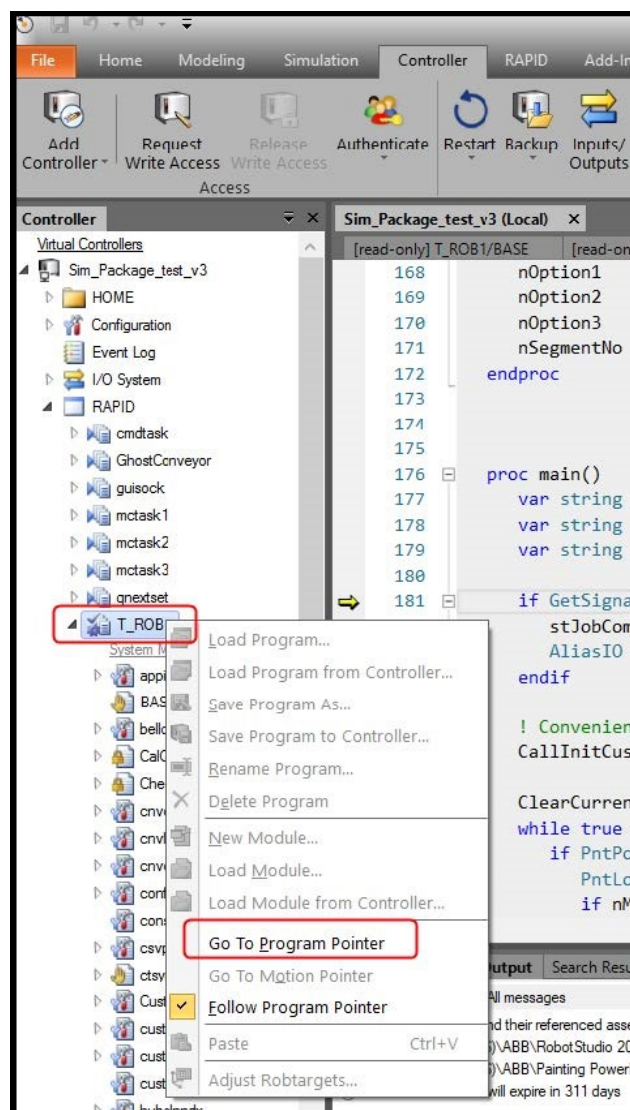
COM6:115200baud - Tera Term VT
File Edit Setup Control Window Help
-->
-->
--> sysdmp_trigger
value = 0 = 0x0
--> 0x6060328 (sysdmp): Sysdmp task triggered...Waiting 100 msec (0)...Dump.
0x6060328 (sysdmp): >> system dump: begin generating...method #2 works after the pro

```

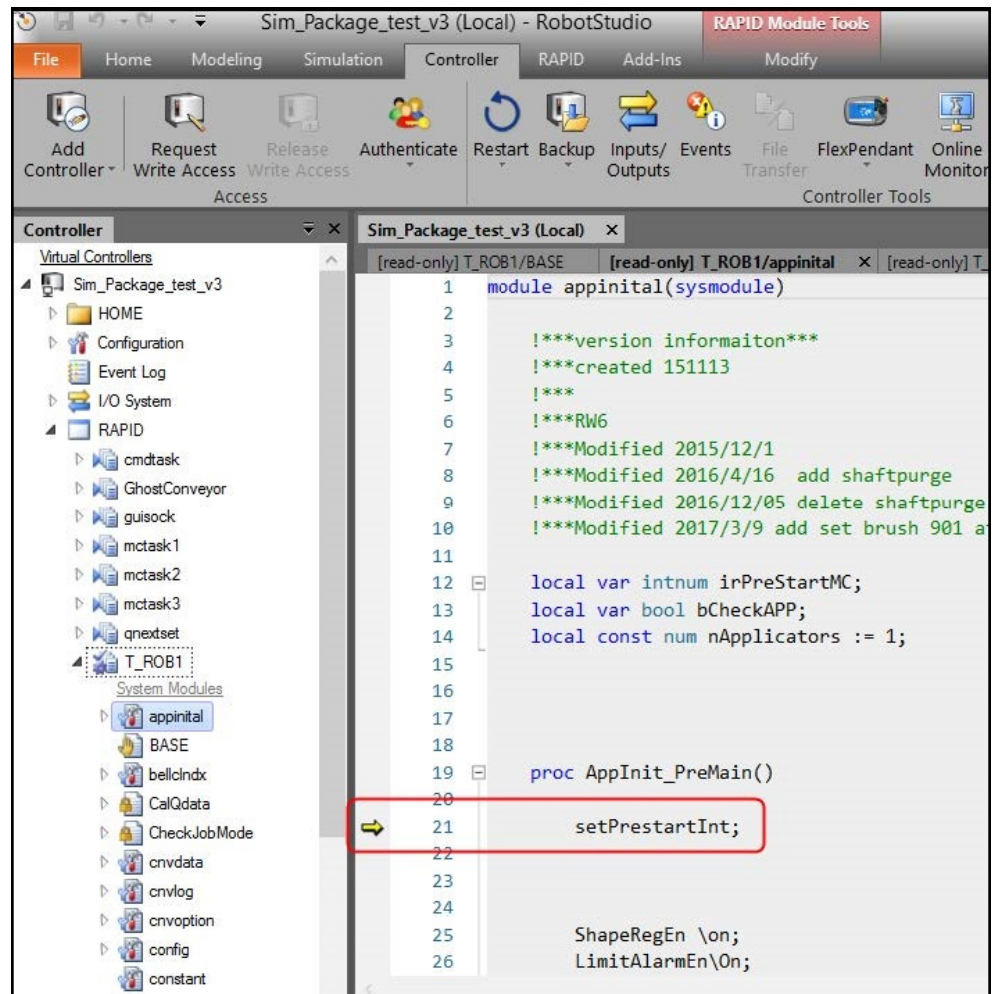
4. Record program pointer with RobotStudio

Instructions

- Connect to the robot. See the description in the section '[Connect controller with RobotStudio](#)' on page 388.
- Expand the menu "RAPID" .
- Right click on task "T_ROB1".
- Click on "Go To Program Pointer".

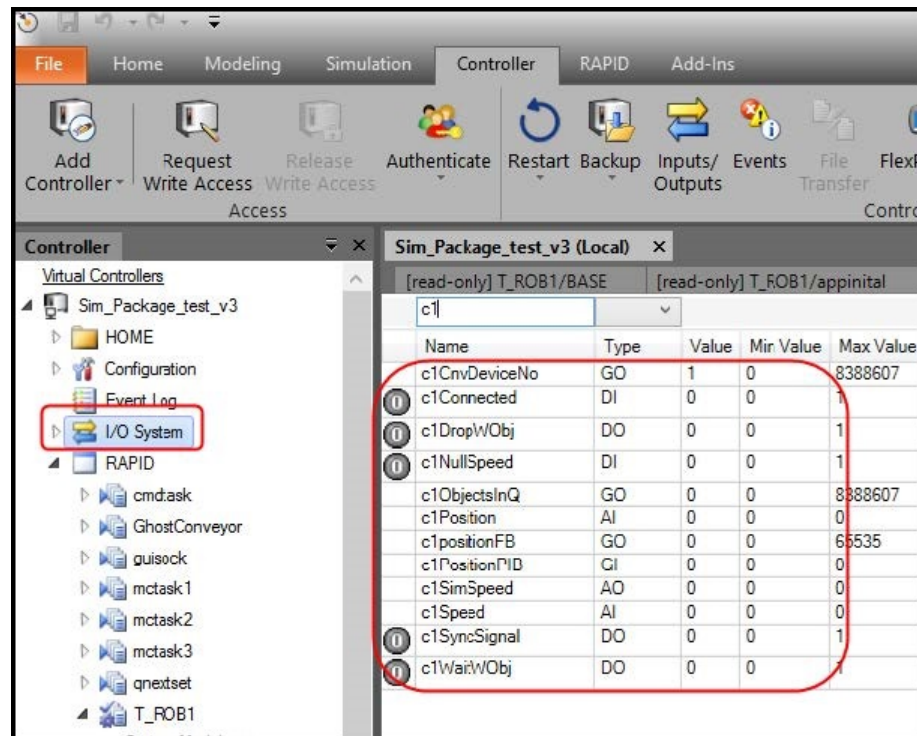


- Create a screenshot or take a photo where the program point stops.



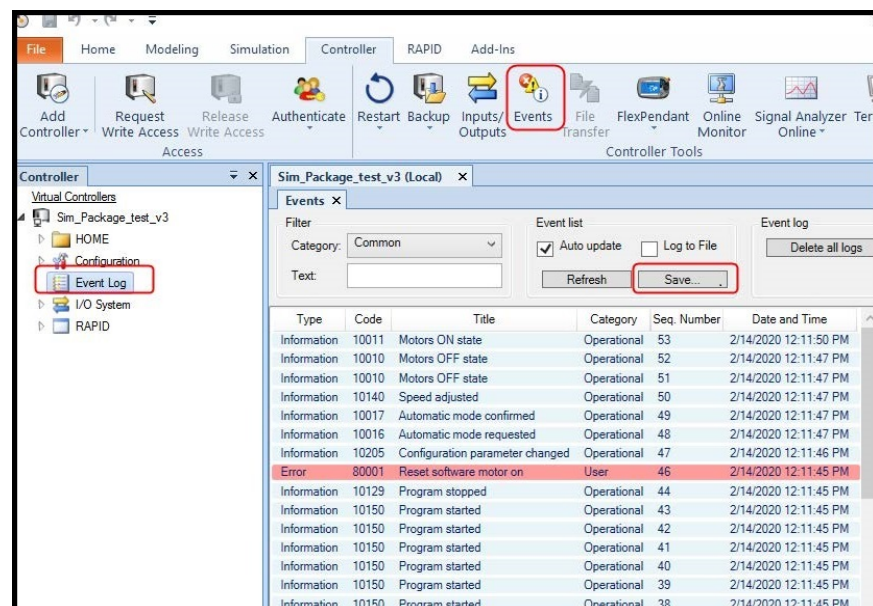
5. Log conveyor signals

- Connect to the robot. See the description in the section '[Connect controller with RobotStudio](#)' on page 388.
- In the controller tab, click on "I/O window".
- Set the filter to "c1" and take a photo or screenshot.



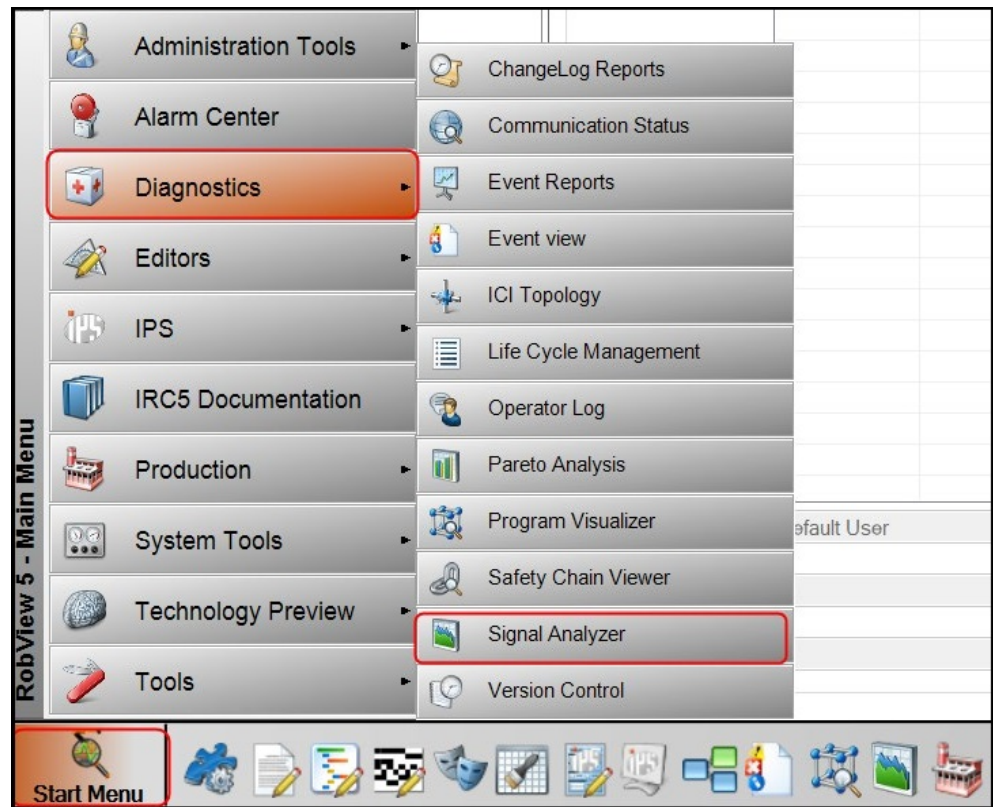
6. Save error log with RobotStudio

- Connect to the robot. See the description in the section '[Connect controller with RobotStudio](#)' on page 388.
- On the Controller tab, click Events to view the Event Log.
- Click the save button and choose the path to save the event log.

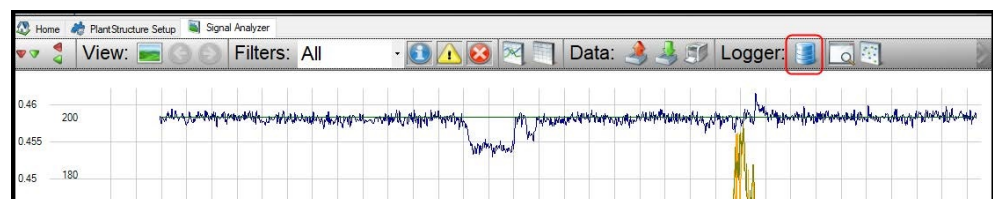


7. Save signal diagnostic with Robview

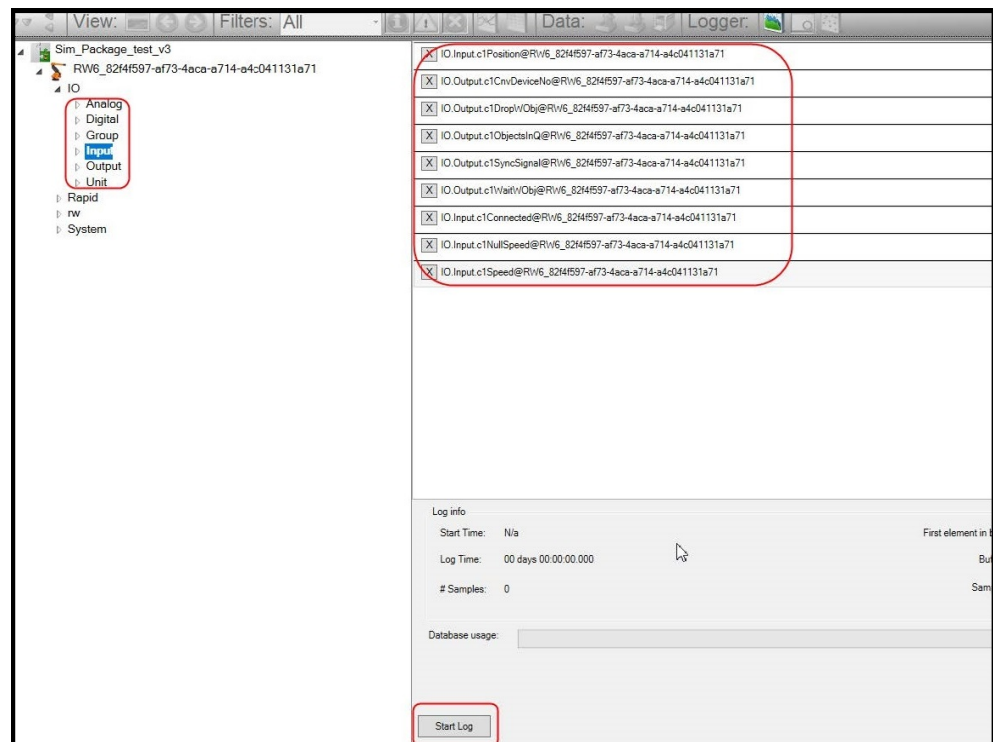
- Connect review with the controller. See the description in the section “Connect controller with Robview.”
- Open the signal analyzer by clicking the window.
- Click the Start Menu → Diagnostics → Signal Analyzer.



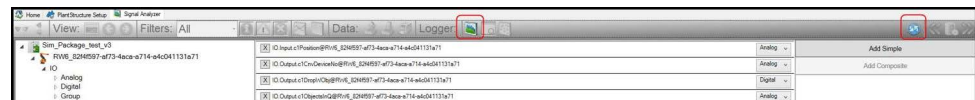
- Click the data log icon and go to the window to add monitor signals.



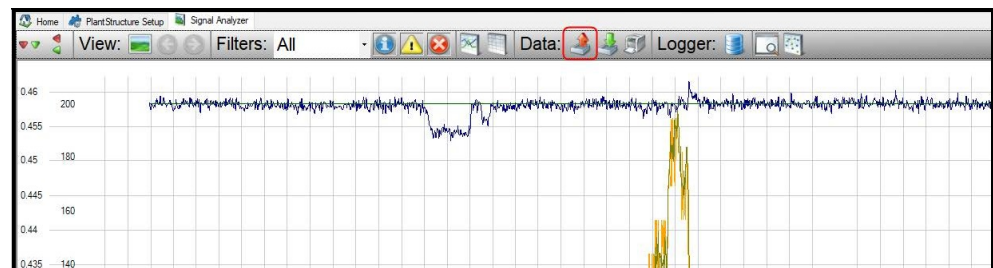
- Add the conveyor-related signal or other needed signal for the monitor, then click the Start Log button in the middle bottom.

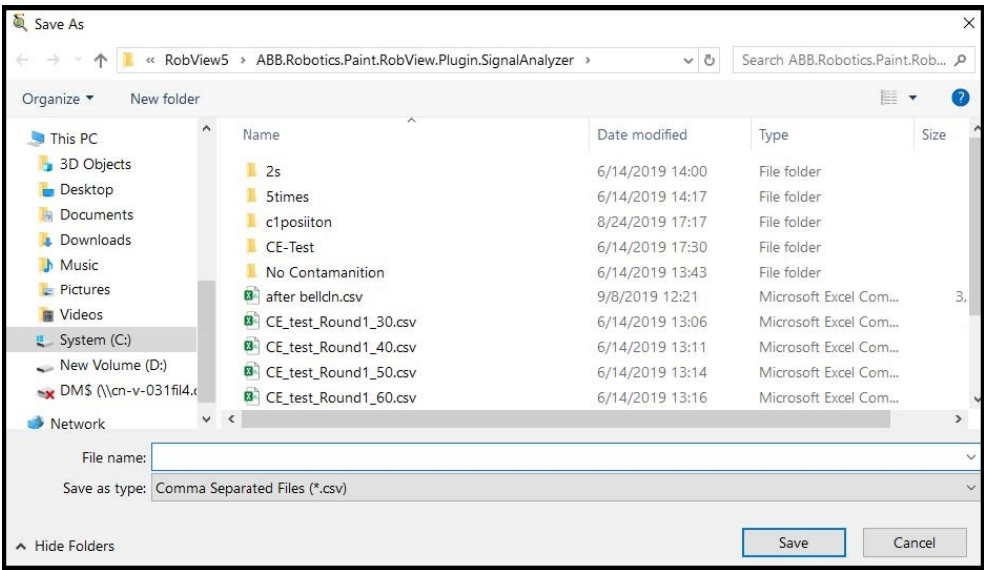


- Click auto-refresh data and click show analyze.



- Export log signal to CSV file.





7.7.3 Troubleshooting with log file

Description The robot system contains several different log files stored in the Log folder within the robot's HOME directory, which can be used to analyze and troubleshoot the system's behaviour.

The PaintWare creates several log files. For more information of the configuration options, as well as a description of the production log, see the *PaintWare system parameters*, in *Operating manual - IRC5P RW6.0* listed in 'References' on page 15.

The following log files are available:

Name	Log file name	Description
Command log	cmdlog.txt	Logging of all internal and external commands executed by the PaintWare.
	cmdlog.old	
Production log	prodlog.txt	Logging of all executed jobs including the job data and cycle time.
	prodlog.old	
User log	Userlog.txt	Logging of all customized information programmed by user.
Conveyor log	Cnv1.log	Logging all conveyor related signal status change.

Production log SmartContrl configures the production log in the following format:
%D, Prog:%P, Mtrl:%M, JobMode:%1, RePart:%2, Skid:%3, Brush:%B, Usage:%C,
CycleTime:%T

With this configuration, the maximum available information is stored in the production log in the following order:

- Date and time stamp
- Program index
- Material index
- Job mode
- Repair part value
- Skid number
- Brush table number
- Material consumption
- Cycle time

7.7.4 Troubleshooting with e-log

Error number	Error ID	Error description
116001	Atomzier not enabled!	Robot applicator not enable.
116002	Fluid feedback error	No Fluid volume.
116003	Signal not online	Signal \$arg1 not online. The function \$arg2 may not work properly.
116004	Shape air Maintenance error	\$arg1 Maintenance no air.
116005	Shape air Maintenance ok	\$arg1 air Maintenance OK.
116006	Robot stop by PLC	Robot stop by PLC, please check PLC error log for reason.
116007	Wait signal ready timeout	Wait \$arg1 reach setpoint timeout.
116008	High voltage ready timeout	Wait High voltage reduced to \$arg1 overtime.
116009	Atom calibration error	No material load for atom calibration.
116010	Applicator calibration error	Applicator calibration failed.
116011	Applicator calibration succeed	Applicator calibration succeed.
116012	Applicator calibration error	Applicator not ready for calibration, check robot mode and applicator status.
116013	CBS semi command	Command 750 to put cartridge back to CBS from atomizer.
116014	CBS semi command	Command 750 to put cartridge back to CBS and clean.
116015	CBS move	No cartridge in robot, just pass the execution.
116016	App disable failed	Robot applicator disable failed from PLC.
116017	Bell Cup check	The bell cup detects missing.
116018	Vibration check	The detect vibration is \$arg1 too big.
116019	Rotation issue	Rotation not calibrated.
116020	Set ready brush timeout	Atom can't reach the setpoint for ready brush.
116021	Init routine missing	Init error, file \$arg1 is missing init routine or not loaded.
116022	Brush factor info	Brush factor file updated.

Error number	Error ID	Error description
116023	Brush factor failed	Brush factor update failed.
116024	Fiber issue	The fiber strength is \$arg1, value low.
116025	Pump pressure diff too high	The \$arg1 pressure diff is \$arg2. Value over 3 bar.
116026	Pump in pressure too high	The \$arg1 pressure n is \$arg2. Value over 5 bar.
116027	Release pressure	Open dump valve to release pressure.
116028	Release pressure	Close dump valve.
116029	Paint leakage	Leakage happens, close the colour valve.
116030	Conveyor stop	Conveyor stop waiting \$arg1 finish.
116031	Conveyor resume	The \$arg1 finished. Conveyor resume!
116032	Conveyor position error	Conveyor position too big. The current conveyor position is \$arg1 mm, over the allowed value \$arg2 mm.
116033	Conveyor status change	The \$arg1 conveyor selected.
116034	Valve auto close	The valve \$arg1 open too long time.
116035	Update wait pos succeed	Update conveyor sync position succeed.
116036	Conveyor update issue	Missing program module name, need manually save the file.
116037	Conveyor update failed	Update sync position failed because of conveyor stop. Update failed, try again!
116038	Conveyor sync error	The variable stWCPname is not accessible. Check if the module assigns the correct name.
116039	Conveyor sync error	The variable stWCPname have different dimension. Check if the variable array dimension.
116040	Conveyor sync error	Unknown error!
116041	Conveyor sync error	Update mode and limit check mode conflict!
116042	-	-
116043	Conveyor sync info	Skip wait cnvpos.
116044	Conveyor sync error	Cannot skip wait cnvpos in update cnv mode!
116045	Conveyor sync point error	The conveyor nPosNr \$arg1 is over limit \$arg2.
116046	Conveyor sync point info	Set wait brush, retract and sync with point \$arg1 c1position is \$arg2.
116047	Conveyor sync point info	Set last brush, retract and sync with point \$arg1 c1position is \$arg2.
116048	Conveyor sync point info	Set wait brush, sync with point \$arg1 c1position is \$arg2.
116049	Conveyor sync point info	Set wait brush, sync with point \$arg1 c1position is \$arg2.
116050	Conveyor position point error	Conveyor position \$arg1 is over limit \$arg2.
116051	Conveyor sync joint info	Position update \$arg1. c1position = \$arg2 Joint 456 = \$arg3; \$arg4; \$arg5.
116052	Cycle time report	The cycle time for \$arg1 is \$arg2.
116053	Volume calculation	The volume for \$arg1 is \$arg2 ml.
116054	Conveyor position error	Conveyor position \$arg1 is over the limit position \$arg2.

Error number	Error ID	Error description
116055	Mixer ration error	Mixer ratio signal error. The desire mixer ration is \$arg1. The actual mixer ration is \$arg2.
116057	Paint push error	Paint over push.
116059	Position move	Robot is waiting for \$arg1 enable from PLC!
116060	Position error	Robot not at \$arg1 position!
116061	Service position update	Service position \$arg1 teach by program.
116062	Wait position error	Wait robot in \$arg1 position timeout!
116063	Calcu count per meter	The counts per meter is \$arg1.
116064	Work object calcu error	Work object calculation error.
116080	Missing Proc configuration	Read the procedure option \$arg1 error!
116084	Smart home error	Robot in singularity area, move home manually.
116085	Smart home error	Smart home save array overflow.
116086	High voltage disable	High voltage disabled.
116087	High voltage power off	High voltage power is off.
116088	HV value too high	HV Value is over limit during cleaning.
116089	HV power on	HV hardware on during cleaning.
116090	HV disable	Disable HV during cleaning.
116091	HV solvent interlock error	HV solvent interlock open!
116092	HV test error	Initial HV dynamic test error.
116093	HV test error	Initial HV static test error.
116094	Position not taught	The service position \$arg1 not taught!
116095	Unhandle Error	The error in routine \$arg1 unhandle.
116096	Move to home	Go home bypass smart home.
116097	Smart home error	Wait other robot home position timeout. Please check proc SmartBlock setting.
116098	Smart home error	Move to home with smarhome failed.
116099	Move home error	Can't move to home position. Please check station status.
116100	Robot ID parse error	The robot ID format not correct, ID must have R,D and H.
116101	HMI command failed	HMI command failed, the robot not in station manual mode.
116102	HMI command succeed	Received HMI command \$arg1 succeed.
116103	Pump shaft enable	Set \$arg1 enable.
116104	Pump shaft disable	Set \$arg1 disable.
116105	Brush test failed	No material in system, please load material first.
116110	Interlock error	The zone number \$arg1 over the limit.
116120	Door Request error	Requested door position out of range. DoorPos: \$arg1.
116121	Door Request error	Requested door opener out of range. DoorOpener: \$arg1.
116126	Door Check bypassed	Request Opener \$arg1 bypassed. The bypass signal \$arg2 is 1.

7 Robot system

7.7 Troubleshooting

Error number	Error ID	Error description
116127	Door Check conveyor over limit	Wait time door position \$arg1. c1position \$arg2 over maximum value \$arg3.
116128	Judge Opener bypassed	Bypass signal \$arg1 is on. Run ProcName1 \$arg2 Skip ProcName2 \$arg3
116131	PLC request position	The PLC send position is \$arg1
116132	Opener master error	More then one master request at same time.
116133	Opener moving error	Gripper Error, Go to position \$arg1 failed.
116134	Opener moving error	Request position \$arg1 is illegal.
116135	Opener moving error	Robot not move from \$arg1 position.
116138	Opener type error	Opener type not define, not door and hood trunk opener. Check the variable stRobotFunc.
116139	Painter request error	PLC request is not reset before running.
116140	Wait painter conveyor over limit	Wait painter request \$arg1 over conveyor limit. The conveyor position is \$arg2. The limit c1position is \$arg3.
116142	Painter request bypassed	Wait Painter request \$arg1 bypassed, because of signal \$arg2 is 1.
116143	Opener wait position update	Update the counter is \$arg1, the conveyor position is \$arg2.
116144	Painter request bypass error	All wait robot bypassed.
116145	Auto bypass painter request	Robot \$arg1 wait request \$arg2 is bypassed. The bypass distance is \$arg3.
116146	Bypass mode error	Bypass mode wrong.
116147	Opener feedback error	Illegal position \$arg1 feedback.
116148	Opener name error	Wrong robot name \$arg1 use wait station safe.
116160	Gripper position error	Gripper home sensor no signal.
116161	Part search error	Gripper sensor activated at \$arg1, out of the limitation \$arg2, \$arg3.
116162	Part search error	Maximum search time \$arg1 reach.
116163	Part search error	Sensor signal \$arg1 not define.
116164	Part search error	Gripper sensor not activated during search.
116165	Sensor bypassed	Part search sensor bypassed in repair mode.
116166	Part search error	Gripper sensor is on before the check.
116167	Gripper error	Gripper sensor missing.
116168	Gripper error	Gripper sensor is on before checking.
116169	Gripper error	Gripper sensor is on before checking. Robot in ghost or sensor bypassed mode.
116170	Gripper sensor disabled	Gripper sensor disabled.
116171	Gripper Error	Sensor lost during sensor watching.
116180	Job queue error	Job queue full, delete first object in queue.
116183	Job pass start window	Program \$arg1 passed start window.
116185	Conveyor queue change	Conveyor object queue \$arg1.
116186	Sync signal received	Sync signal from PLC received.
116200	Module load error	Module \$arg1 load failed.

Error number	Error ID	Error description
116210	SmartHomeList stack full	SmartHomeList stack full, tried to address an element over the max.
116211	Smart home error	Cnv position saved wrong, saved c1position \$arg1 actual value \$arg2.
116212	Smart home executed succeed	Smart home executed succeed.
116213	Smart home error	No point saved and robot not moved.
116214	Smart home error	Missing book parameter nCnvPos for conveyor tracking station.
116215	Smart home error	BookEscapePoint and paintsmart wrong ord. Current nSmartCounter is \$arg1.
116216	Smart home warning	Signal \$arg1 not define.
116217	Smart home error	The nSmartCounter is over limit 100 or below 0.
116218	Smart home error	Save c1position error.
116230	Check home position error	Robot not in home position when run none motion operation. Need Move to home position!
116240	Paint push error	Push volume error, Total volume: \$arg1.
116241	Paint push error	Job received too late, paint push abort. The actual fluid value is \$arg1. The need value is \$arg2.
116244	Paint push feedback error	Feedback program, result: \$arg1.
116260	Solvent pressure check disable	Solvent pressure check disable.
116261	Solvent pressure check error	Solvent pressure release over time.
116262	Solvent pressure too low	Solvent pressure low.
116263	Solvent pressure check info	Solvent PS checked time \$arg1 s.
116560	PLC send to Robot for Work object	Work object data is : \$arg1.\$arg2
116561	Wait PLC get data timeout	Please check it.
116562	Vision system send data is wrong	Please check it.
116570	BK Error	General error.
116571	BK Error	Invalid state.
116572	BK Error	Timeout while sending data.
116573	BK Error	Timeout while receiving data. Code Position: \$arg1.
116574	BK Error	Invalid mode. BK Mode: \$arg1.
116575	BK Error	Invalid type. Code position: \$arg1.
116576	BK Error	Invalid path. Path: \$arg1.
116577	BK Error	Invalid point. Point: \$arg1.
116578	BK Error	Mismatch of number of points between set point and actual.
116579	BK Error	Invalid measurement for point: &arg1.
116580	BK Error	Collision warning for point: \$arg1.
116581	BK Error	Configuration error for type / path: \$arg1.
116582	BK Error	Measured value too big for point: \$arg1.
116583	BK Error	General error at point: \$arg1.
116584	BK Error	No correction for point: \$arg1.

This page is intentionally left blank.

8 System functionalities

This chapter lists all functionalities that are provided with reference manual and gives an overview over the available functions and used concepts which are the basis for the package to work.

Following sections are included:

'Program execution flow' on page 404

'Robot mode' on page 408

'Interior concept' on page 416

'Create a new program' on page 419

8.1 Program execution flow

Following sub-sections are included in this section:

'Paint robot workflow' on page 404

'PixelPaint robot workflow' on page 407

'Clean robot workflow' on page 407

'Flaming robot workflow' on page 408

'Dispense robot workflow' on page 408

8.1.1 Paint robot workflow

Following sub-sections are included in this section:

'T_ROB1 task' on page 405

'Command task' on page 406

'Colour change task' on page 406

'Application task' on page 407

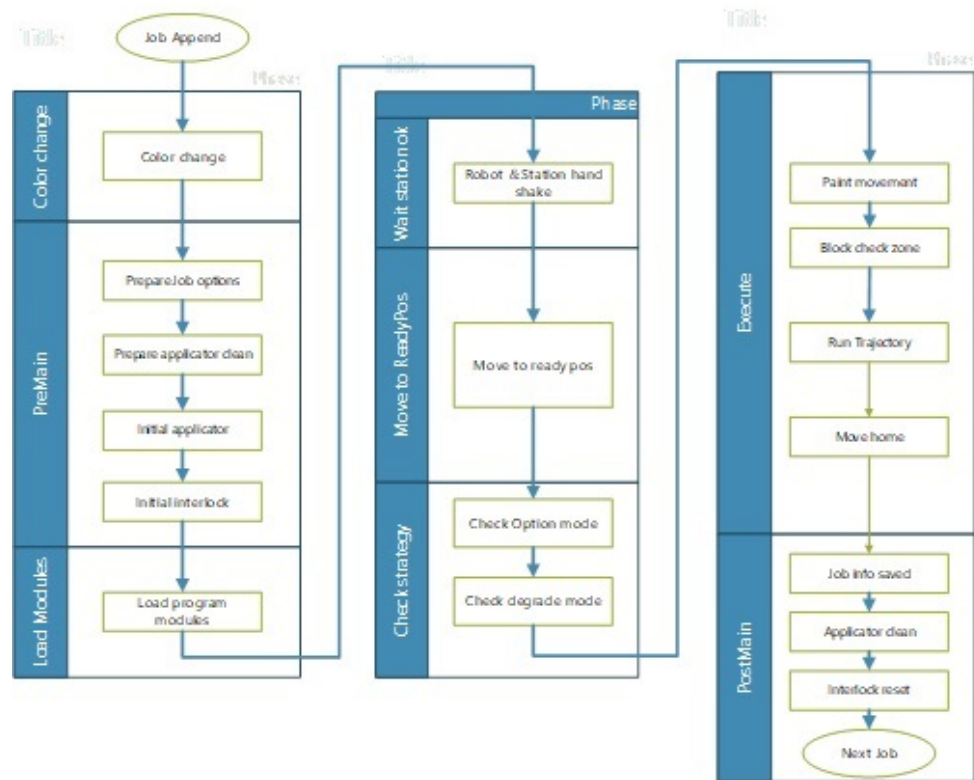
'Service task' on page 407

8 System functionalities

8.1 Program execution flow

8.1.1.1 T_ROB1 task

The T_ROB1 is the main task of paint robot system. Beside of the motion control, the application, position check and job information handle is also in this main task.



The flowing functions are handled by T_ROB1 task.

- Job mode analyze
- Job information check
- Robot station handshake
- Station interlock
- Conveyor synchronization
- Application related function.

8.1.1.2 Command task

Command task is the interface with PLC about command. The customized command, command hooks and system related interrupt have been handled by command task.

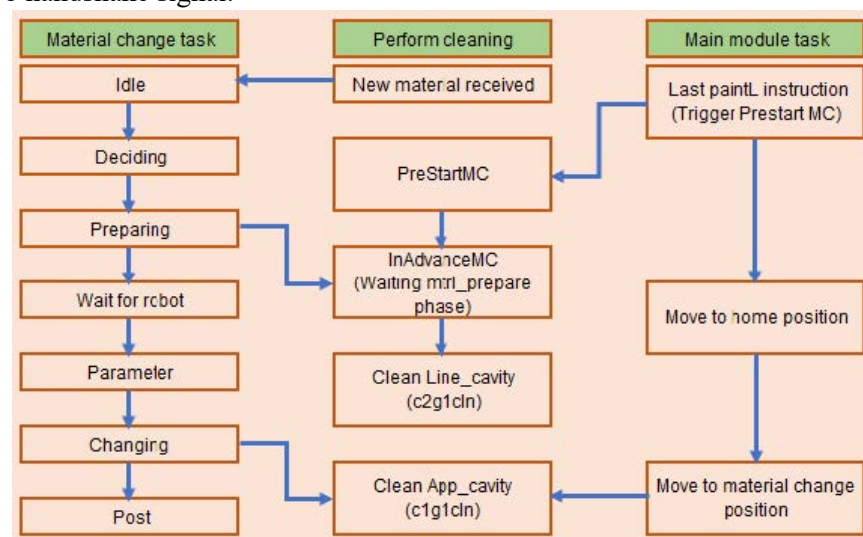
Command hooks	Descriptions
Cmd101PreHook	External reset when command 101 executed
Cmd104PreHook	Reset signals when command 104 executed
Cmd104PostHook	Handle colour change when command 104 executed
Cmd103PreHook	Handle start brush when command 103 executed
Cmd421PostHook	Handle job information when command 421 executed
Cmd416PostHook	Reset signals when command 416 executed

Reset Error handle:

- PLC stop information
- Automatic mode acknowledgement
- Colour change suspend when A1Error.

8.1.1.3 Colour change task

The colour change task is responsible for the colour change logic for the MC Engine and related application files. The colour change task communicates with T_ROB for a job handshake signal.



- Per-Start MC helps to save colour change time, it lets the colour change run before the robot arrives at the colour change position.
- Separate cavity for a 1K system makes it possible to run colour change through dump-like and makes the pre-start colour change safer.
- Solvent supervision make it possible to suspend colour change when solvent pressure low.

8 System functionalities

8.1 Program execution flow

8.1.1.4 Application task

Application tasks handle application-related functions. The task will run with a fixed loop time of 100 ms. The application task will handle the below functions.

Functions	Descriptions
AppEnableControl	Application enable check function
HVEnableControl	HV enable check function
HVSolventCheck	Solvent HV interlock check function
ResetAppError	Reset application error
BrushTabCheck	Brush table loading check function
AppPreventMaint	Prevent maintenance function
ValveCheck	Valve check function

8.1.1.5 Service task

Service tasks handle general related functions. The task will run with a fixed loop time of 100 ms. The below functions will be handled by service task.

Functions	Descriptions
Ghost conveyor	Ghost conveyor
Conveyor stop function	Halt conveyor for colour change or application clean
Digital signals output function	Provide digital interface for external software
Shape air purge function	Provide purge air during manual clean

8.1.2 PixelPaint robot workflow

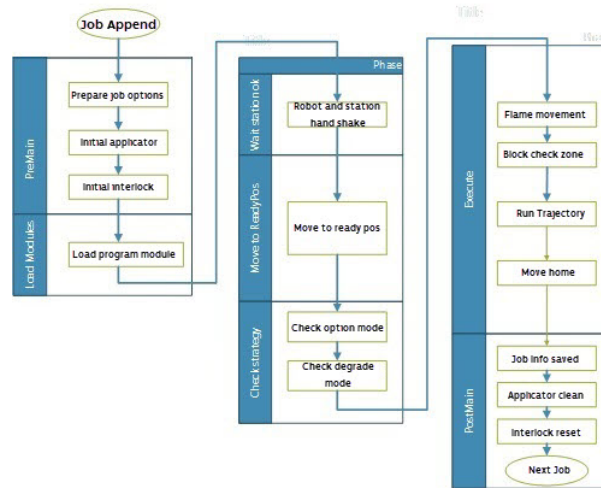
For the normal task workflow, please take the reference of Paint robot workflow.

8.1.3 Clean robot workflow

TBD

8.1.4 Flaming robot workflow

A "flaming robot workflow" typically refers to a process or set of operations involving a robot, possibly with elements such as fire, flames, or intense activity.



8.1.5 Dispense robot workflow

TBD

8.2 Robot mode

Following sub-sections are included in this section:

'Job IO structure' on page 409

'Variable Car Type' on page 410

'Degrade mode' on page 411

'Repair Mode' on page 412

'Bypass mode' on page 414

General

During production, the robot has a fixed IO interface for job information. It also provides different variables for car type and options for process routine execution.

When one or multiple robots fail, bypassing them allows the whole station to run in a degraded mode for continued production. Also, in some cases, the car body could be delivered in repair mode, containing only the parts needed to be painted.

8 System functionalities

8.2 Robot mode

8.2.1 Job IO structure

As the command interface is defined between PLC and Robot, the job information is sent with the Job queue append command 421 and bound ten parameters input.

Parameters	Descriptions	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	Addresses	
Param1	Program																	di17-31	
Param2	Material																	di32-47	
Param3	O1	Degrade Mode									Car Option								di48-63
Param4	O2	B	H	L		Repair parts Option													di64-79
Param5	O3	Brush No (Option)																di80-111	
Param6	M1																	di112-127	
Param7	M2																	di128-143	
Param8	M3	Valve Index																di144-159	
Param9	Quatity																	di160-175	
Param10	Skid ID																	di176-191	

- **Degrade Mode** - Station Degrade Mode number
- **Car Option** - Variables number for Car Option
- **Repair parts option** - Repair mode information
- **B** - Bell clean flag
- **H** - Hub clean flag
- **L** - Learn Mode, learn the leakage check and other related function
- **Brush No** - Option for Brush number sent from PLC
- **Valve Index** - Valve address ready to use in next job

8.2.2 Variable Car Type

Description

Usually, there are some standard car options for painting, like the roof, two-tone, cover for fuel/charging, and some particular use for the same car type. However, there is no standard input interface from the conveyor line or customer to provide information on the car option.

Concept

For robot applications, a standard interface should be used to collect all the variables representing different options for car types.

This is useful for processing with a common standard and interface; for process programming, it can use the same variables and even represent different options of car information for other customers; for robot systems, it also provides a simple interface to analyze the job information, and prepare or provide required information based on the job information. E.g. paint volume calculation, Brush table indexing, etc.

Car option	Roof	Two-tone	Customized option
0	0	0	...
1	1	0	...
2	2	0	...
3	0	1	...
4	1	1	...
5	2	1	...
6	0	2	...
7	1	2	...
8	2	2	...
...	...	...	...

- **Roof** : 0 - No roof, 1 - Small roof and 2 - Big roof.
- **Two-tone** : 0 - No two tone, 1 - Single two tone and 2 - Twice two tone.
- **Customized option** : 0 - ..., 1 - ..., 2 - ...

For example, PLC might get different options for roof, two-tone, or more customized options; while it resolves these and gets indexing from this table, the PLC sends the single variable “car option” to the robot.

The user can use RobUI to define the various option groups, but with a single indexing car option, the robot reads the single variables from the car option and resolves their detailed information using this table index. If the car option is 2, the painting car has a big roof, without two-tone.

The table index can be edited from the RobUI interface and saved or loaded from PLC.

8.2.3 Degrade mode

Description

The definition of degraded mode is the possibility of running production even if one of the robots is out of service. This function is designed to run with a certain amount of maintenance and supervision. In degraded mode, the defective robot must be moved in a bypass position to avoid interference with the other robots in the station.

The operator must select the actual robot as “out of service” and choose the degraded mode on the operator desk – visualization. The PLC will send the information about the production mode to the robot. The degrade mode concept and solution should be agreed upon in advance.

Concept

- For exterior painting stations with conveyor tracking lines, it is possible to choose a degraded mode with higher robot painting speed, two-layer painting decreased to single-layer painting or decreased conveyor moving speed. For the station to have a spare robot, the painting speed, conveyor speed, and paint quality can remain the same.
- For the exterior painting station with Stop&Go line, choose degrade mode with higher robot painting speed or long cycle time if robot reachability is ok. The spare robot can achieve the same cycle time and paint quality.
- For Interior stations with conveyor tracking lines, degrade mode is more complex. There are two possibilities: opener degrade and painter degrade. If the opener degrades, the painter should have the correct tool to open the door, hood and trunk, and the other painter should paint the left area. The other painter must take over the degraded robot’s location if the painter degrades. The total cycle time will increase.

Operation

Enable degrade mode:

1. To enable degrade mode, one global degrade enable button needs to be ON.
2. The robot needs to be degraded in the bypass position (the bypass position can be the same as the home position; in some cases, the bypass position may differ from the home position, such as the interior station). If the robot cannot jog to the bypass position, the ghost home position can also function like the bypass position.
3. The robot that needs to be degraded can be set as a “Degrade robot”.
4. HMI will send specific degraded information to PLC.
5. PLC will send degraded mode information together with job queue information.

Disable degrade mode:

1. Switch the degrade mode button to disable.
2. HMI will automatically deselect the degrade robot and turn off the degrade mode.
3. HMI will send degrade mode 0 to PLC.

Rapid Interface

Degrade mode Variable:

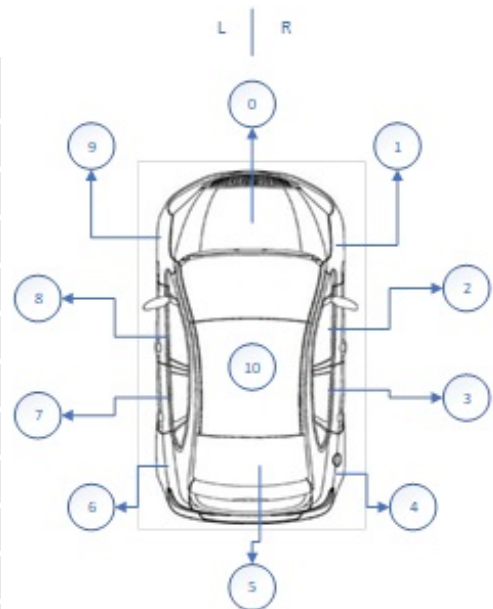
Degrade variable	Value	Description
nPntStrategy	0	Normal mode
	1	Degrade mode 1
	2	Degrade mode 2
	3	Degrade mode 3
	4	Degrade mode 4
	...	...

8.2.4 Repair Mode

The repair mode is used to paint car spare parts. The spare parts consists of, usually, the roof is not a repair part, but in some situations, the whole roof will not be painted; the bit is used to select or paint roofs.

The giant roof and sunroof will be selected in the job options section.

Number	Parts name	System variable name
0	Hood	nRepairHood
1	Right front fender	nRepairRFFender
2	Right front door	nRepairRFDoor
3	Right rear door	nRepairRBDoor
4	Right rear fender	nRepairRBFender
5	Trunk	nRepairTrunk
6	Left rear fender	nRepairLBFender
7	Left rear door	nRepairLBDoor
8	Left front door	nRepairLFDoor
9	Left front fender	nRepairLFFender
10	Roof	nRepairRoof



The repair variable is a clockwise sequence starting from the roof and ending at the roof. The bit is from bit0 to bit10.

Repair Parts name	Bit	Descriptions	Repair Value
nRepairHood	0	Repair Hood	0,1,2,3,4
nRepairRFFender	1	Repair front fender	0,1,2,3,4
nRepairRFDoor	2	Repair right front door	0,1,2,3,4
nRepairRBDoor	3	Repair right back door	0,1,2,3,4
nRepairRBFender	4	Repair right back fender	0,1,2,3,4
nRepairTrunk	5	Repair trunk	0,1,2,3,4
nRepairLBFender	6	Repair left back fender	0,1,2,3,4
nRepairLBDoor	7	Repair left back door	0,1,2,3,4
nRepairLFDoor	8	Repair left front door	0,1,2,3,4
nRepairLFFender	9	Repair left front fender	0,1,2,3,4
nRepairRoof	10	Repair roof	0,1,2,3,4

There are four states for each repair variable.

- Zero is for previous, following, and itself; paint is unnecessary.
- One means itself needs paint.
- Two indicates only the previous part needs paint.
- Three shows only the next part needs paint.
- Four indicates both the previous and next parts need paint, but there is no paint.

Repair value	Descriptions	Previous Part	Parts	Next Part
0	nRepairNothing	×	×	×
1	nRepairAll	×	✓	×
2	nRepairFront	✓	×	
3	nRepairBack	×	×	✓
4	nRepairBoth	✓	×	✓

8.2.5 Bypass mode

The definition of bypass mode is the possibility of running the whole station without one or two robots involved. When the robot has a problem and cannot start production, the robot can be deselected and kept in a bypass position without influencing the other robot's painting. The bypass mode is an easy way to increase the whole station's production ability.

Concept

- For the exterior station with a tracking line and Stop&Go line, when the robot is in bypass mode, the interlock zone and process zone will be released automatically so the other robot can continue painting. The robot has to jog to bypass position or ghost home position.
- For the interior station with a Stop&Go line, when the robot is in bypass mode, there are two possibilities: the opener bypassed, and the painter bypassed.
- When the opener is bypassed, the painter's job is to prevent the opener's area from being executed; the painter needs to jump to the other paint sequence and area to continue painting. If the painter bypasses, the opener needs to be controlled by the different living painters to open and close the door.
- For the interior station with a tracking line, when the robot is in bypass mode, there are two possibilities: the opener bypassed, and the painter bypassed.
- When the opener is bypassed, the painter's job to paint the opener's area cannot be executed. The painter needs to jump to the other paint sequence and location, and at the same time, the conveyor waiting position needs to skip the no paint area to let the robot continue running the conveyor and wait for the other area.
- If the painter bypasses, the opener will open and close the door based on the conveyor position; the conveyor position can be set or learned before. The other painter can also control the opener to open and close the door.

Operation

Enable bypass mode:

1. The robot needs to be bypassed in the bypass position (the bypass position can be the same as the home position; in some cases, the bypass position may differ from the home position, such as the interior station). If the robot can't jog to the bypass position, the ghost home position can also have the same function as the bypass position.
2. The robot that needs to be bypassed can be chosen as a "bypassed robot".
3. HMI will send specific bypass information to PLC.
4. PLC will send bypass mode information with discrete IO.

Disable bypass mode:

1. HMI will deselect the bypass robot to normal mode.
2. HMI will send bypass mode 0 to PLC.

Rapid Interface

Station	Robot	Bypass signal	Signal address	Descriptions
Interior Station	R11	diBypassed_R11	di256	R11 Bypassed
	R21	diBypassed_R21	di257	R21 Bypassed
	R12	diBypassed_R12	di258	R12 Bypassed
	R22	diBypassed_R22	di259	R22 Bypassed
	R13	diBypassed_R13	di260	R13 Bypassed
	R23	diBypassed_R23	di261	R23 Bypassed
	R14	diBypassed_R14	di262	R14 Bypassed
	R24	diBypassed_R24	di263	R24 Bypassed
	D11	diBypassed_D11	di264	D11 Bypassed
	D21	diBypassed_D21	di265	D21 Bypassed
	D12	diBypassed_D12	di266	D12 Bypassed
	D22	diBypassed_D22	di267	D22 Bypassed
	H11	diBypassed_H11	di268	H11 Bypassed
	H21	diBypassed_H21	di269	H21 Bypassed
	H12	diBypassed_H12	di270	H12 Bypassed
	H22	diBypassed_H22	di271	H22 Bypassed
Exterior Station	R11	diBypassed_R11	di256	R11 Bypassed
	R21	diBypassed_R21	di257	R21 Bypassed
	R12	diBypassed_R12	di258	R12 Bypassed
	R22	diBypassed_R22	di259	R22 Bypassed
	R13	diBypassed_R13	di260	R13 Bypassed
	R23	diBypassed_R23	di261	R23 Bypassed
	R14	diBypassed_R14	di262	R14 Bypassed
	R24	diBypassed_R24	di263	R24 Bypassed

8.3 Interior concept

Following sub-sections are included in this section:

'Interior station data' on page 416

'Stop&Go concept' on page 417

'Line tracking concept' on page 417

8.3.1 Interior station data

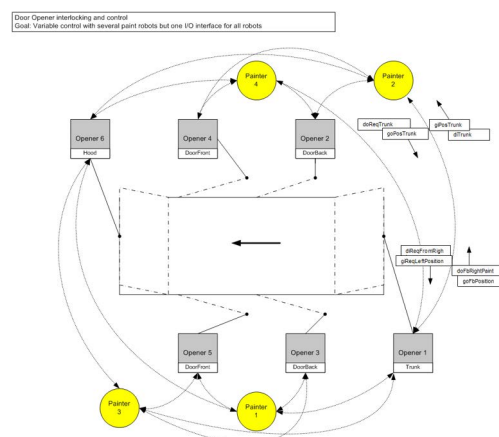
Before painting the inner cavity of a car body, the door opener robots need to open the doors and the front and rear covers of the car body to a position that does not interfere with the painting robot. In this process, each painting robot can request the door opener robot on the same side to open the door to the specified position, and the door opener robot will give the current position information to all painting robots after running to the specified position.

Each door opener and lid opener robot is equipped with sensors on the door opening tool to detect whether the door is open or closed.

Only when both the doors are in place and the sensor signal is triggered will the door opener return the current position information of the door opener robot to the painting robot; if the sensor is not triggered in this process, the current door opener or lid opener robot enters the fault processing program, the sensor fault alarm and the robot is automatically stopped by PLC.

If the sensor is bypassed, the door opener or lid opener robot in the production process will ignore the sensor signal detection after running to the position requested by the painting robot and directly feedback the current position information to the robot.

The following diagram shows the possible communication channels:



8.3.2 Stop&Go concept

After receiving the WaitStationOK and external startup, the painting robot will request a door opener to open the door. Then, the door opener robot will open the door.

The opener robot in the Hook/Half Open/Full Open/Closed position will check whether the door sensor is triggered; the painting robot performs the painting task after receiving the position feedback of the door opener robot corresponding to the requested position of the painting robot. Suppose the door-opening robot performs the door-opening operation and does not interfere with the painting robot.

In that case, the door-opening robot can also open the door to the position required by the painting robot before receiving the door-opening request from the painting robot. In this way, the painting robot can perform the painting task directly after checking the location of the door without waiting. After painting, the painting robot requests the door opener to close the door and lid. Then, release door opener control.

8.3.3 Line tracking concept

Following sub-sections are included in this section:

['Ghost conveyor'](#) on page 418

['Conveyor wait position'](#) on page 418

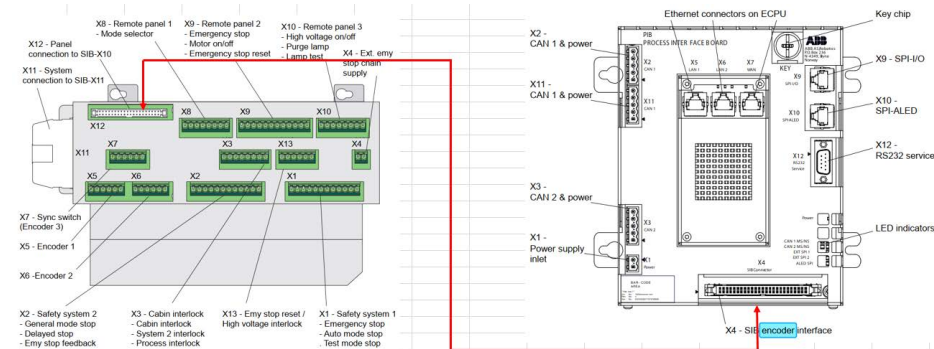
['Job start signal check'](#) on page 418

['Conveyor position check'](#) on page 419

['Job pass start window check'](#) on page 419

8.3.3.1 Ghost conveyor

The conveyor tracking station's conveyor position can be from an accurate conveyor or a simulated conveyor in a robot IPS system. Therefore, the robot can run in simulation mode with or without a conveyor. In ghost conveyor mode, the operator can run the update or check the conveyor wait position (the position to sync the robot with the conveyor). You can also start the production simulation with the correct parameter.



Ghost conveyor operation:

1. Entry ghost mode.
2. Set the station to auto mode.
3. Set the robot in auto mode and cycle on.
4. Click ghost with conveyor.
5. Set the ghost job information and start the ghost mode.

8.3.3.2 Conveyor wait position

To ensure the quality of painting products, effectively control the production time for the conveyor tracking station. Avoid robot joints exceeding the maximum working area range. The robot needs to be synchronous with the conveyor. When the conveyor stops, the robot should move to a synchronized point and stop painting. The synchronize point can be defined with the instruction “waitConveyorPos”.

8.3.3.3 Job start signal check

The sync signal (synchronization signal, sometimes called start signal) is used to enter the position of the work object on the conveyor into the robot control system. The sync signal is entered by installing a sync switch on the conveyor.

After the robot receives the synchronous start signal, it will give feedback to the PLC and record the corresponding information to facilitate problem tracing when there is a problem.

8.4 Create a new program

8.3.3.4 Conveyor position check

CheckLimit is used to check the wait position, which can be used commonly; any stop of the conveyor during the painting period can make sure the execution of the trajectory is smooth. Checklist mode can be chosen by HMI, or **LimitCheckMode** can be used to switch this function. The robot will request the conveyor to stop when it reaches the wait, then wait until PP moves to the next wait, after which the robot will release the conveyor. It will check whether the trajectory can be executed smoothly or not if there is any stop of the conveyor during painting.

8.3.3.5 Job pass start window check

The start window defines the area that if a program starts using an object within the window, then all program coordination can end before the maximum distance or work area is reached. All objects within this window are tracked and are eligible for use in a coordinated work object. A WaitWObj instruction will connect to the first object in the window. If the object has passed the StartwindowWidth without connection, the robot will stop running and output an alarm message.

8.4 Create a new program

Following sub-sections are included in this section:

- 'Paint robot' on page 419
- 'Pixel paint robot' on page 425
- 'Clean robot' on page 425
- 'Flaming robot' on page 425
- 'Dispense robot' on page 429

8.4.1 Paint robot

Following sub-sections are included in this section:

- 'Segment file' on page 420
- 'Create motion program' on page 420
- 'Create Brush table' on page 422
- 'Create colour change sequence' on page 423
- 'Create start position file' on page 424

8.4.1.1 Segment file

Description	This file defines what program file(s) should be loaded and executed by what program index. If the segment file does not exist or an entry for a specific program index is missing, the default naming will be used: m<program index>.mod. Note: The program extension (.mod) should not be included in the definitions. Alias names cannot be used. The segment system refers only to module filenames on the controller disk. Changes in this file are detected automatically when changed on the controller.
Limitation	The file must conform to the Comma Separated Value (CSV) convention. The segment limit for each program is 80 characters due to the RAPID string limitation.
File name	segment.ndx
Default URL location	ftp://<IRC5>/<system>/HOME/programs/
Format	<program index>, <program file>, <program file>, <program file>
Example	

```
1,BCI_HC23_DM
2,BCI_ESEA
3,BCI_SU
4,BCI_HT2
5,car1,car2,car3
10001,yHomePos
10002,yMtrlChPos
10003,yCalPos
10004,yMaintPos
10005,yBypassPos
```

8.4.1.2 Create motion program

Description	The motion program files contain the paint program or parts of it. Paint program files contain a module with a 'main<name>' .
--------------------	--

8 System functionalities

8.4 Create a new program

The module and file names must be identical, for example, the example program file must be named FRONT.MOD.

It is highly recommended that robtargets be defined as LOCAL CONST. ShopFloorEditor supports this and can be configured as default on the pendant.

Using LOCAL routines for all other routines than the 'main' routine in your paint program is highly recommended. This will prevent name ambiguity between paint programs if the same routine name is used in different programs.

File name

m<program index>.mod or <name>.mod

Default URL location

ftp://<IRC5>/<system>/HOME/programs/

Example

```
MODULE M10
  local var num movetime:=0;
  LOCAL VAR string Pos1;
  LOCAL CONST robtarget p10:= [[-1999.84,716.53,-414.39],[0.305924,0.9363;
  LOCAL CONST robtarget p20:= [[-1701.82,718.79,-383.12],[0.55264,0.75425;
  LOCAL CONST robtarget p30:= [[-1502.27,1443.88,-243.9],[0.82767,0.26566;
  LOCAL CONST robtarget p40:= [[-1640.18,1441.76,-243.93],[0.827668,0.2656;
  LOCAL CONST robtarget p50:= [[-3276.02,1446.62,-242.01],[0.82767,0.2656;
  LOCAL CONST robtarget p60:= [[-3494.17,1443.12,-242.04],[0.827669,0.2656;
  LOCAL CONST robtarget p70:= [[-3449.33,1572.24,-255.33],[0.821783,0.1349;
  LOCAL CONST robtarget p80:= [[-3262.56,1572.2,-255.25],[0.821782,0.1349;
  LOCAL CONST robtarget p90:= [[-1644.56,1571.26,-249.65],[0.82178,0.1349;
  LOCAL CONST robtarget p100:= [[-1469.88,1571.14,-249.62],[0.821778,0.1349;

  PROC mainM10()
    MoveAbsJ jHomePos\NoEOffs,vPaint600,fine,tUserTool0;
    WaitWobj wobjtrack\RelDist:=1000;
    ClkReset clock1;
    ClkStart clock1;
    setbrush 2;
    PaintL p10,vPaint600,z50,tRB1000iWSC_200\Wobj:=wobjtrack;
    PaintL p20,vPaint600,fine,tRB1000iWSC_200\Wobj:=wobjtrack;
    WaitWobj wobjtrack\RelDist:=1300;
    PaintL p30,vPaint600,z50,tRB1000iWSC_200\Wobj:=wobjtrack;
    SetBrush 13\X:=-1579.6;
    PaintL p40,vPaint600,z50,tRB1000iWSC_200\Wobj:=wobjtrack;
    PaintL p50,vPaint600,z50,tRB1000iWSC_200\Wobj:=wobjtrack;
    PaintL p60,vPaint600,z50,tRB1000iWSC_200\Wobj:=wobjtrack;
    PaintL p70,vPaint600,z50,tRB1000iWSC_200\Wobj:=wobjtrack;
    SetBrush 14\X:=-3395;
    PaintL p80,vPaint600,z50,tRB1000iWSC_200\Wobj:=wobjtrack;
    PaintL p90,vPaint600,z50,tRB1000iWSC_200\Wobj:=wobjtrack;
    SetBrush 2\X:=-1546.4;
    PaintL p100,vPaint600,z50,tRB1000iWSC_200\Wobj:=wobjtrack;
    MoveAbsJ jHomePos\NoEOffs,vPaint600,fine,tUserTool0;
    ClkStop clock1;
    MoveTime:=ClkRead(clock1);
    TPWrite " Last Cycle Time : "\Num:=ClkRead(clock1);
    DropWobj wobjtrack;
  ENDPROC
ENDMODULE
```

8.4.1.3 Create Brush table

Description	A brush file for an applicator contains one or more collections of process settings, for example, fluid delivery, atomizing air, etc. The Brush data file contains the process data that is often changed along the path. Using linear paint instructions with different brush numbers makes it possible to achieve optimum control of the paint equipment along the entire route. Brush data are applied to the paint process when a valid trig plane is intersected.
File name	Table<brush table number>.bt
Default URL location	ftp:// <IRC5>/<system>/HOME/ipsdata/node1/A1Brush/
Format	From IPS 4.60.12, there is support for an optional brush file comment. This is limited to 1 line at the top of the file before brush data. This comment field tracks the RFID article number from connected atomizers. Brush file syntax: <pre> <digit> ::= 0 1 2 3 4 5 6 7 8 9 <character> ::= --ISO 8859-1 (Latin-1)-- <comment> ::= '#' { <character> <tab> } <integer> ::= <digit> {<digit>} <float> ::= <integer> '.' [<integer>] <brush> ::= < float > [, < float >] [<comment>] <newline> <brushFile> ::= [<comment>] <brush> {<brush>} </pre> Each row (line) contains the values of one brush. Line number = brush number. Each column contains the values for one process (e.g. Fluid, Atom, Shape1, Shape2, HV). All rows must have the same number of columns. Note: Brush tables with comments are not valid on an old IPS system.

8 System functionalities

8.4 Create a new program

Example

Four brushes (lines, rows), Five channels (columns):

```
0,40,400,200,0
0,40,400,200,60
300,40,400,200,60
300,40,200,400,20
300,40,200,400,20
300,40,400,200,60
```

8.4.1.4 Create colour change sequence

Description

Contain a cleaning sequence.

Remember to add instructions on how to wait for the robot to enter the material change position. Otherwise, the cleaning sequence may start before the robot has reached this position.

File name

c<Cavity>g<Grade>cln.mod

Default URL location

ftp://<IRC5>/<system>/HOME/mcdata/

Format

Rapid syntax

Example

```
MODULE c1g1cln
  PROC mainc1g1cln()
    SetTmSignal "A1PumpPSInLimEn",0,0\Reset;
    SetTmSignal "A1AtomFluidLimEn",1,0;
    SetTmSignal "A1Con",1,0;
    SetTmSignal "A1DumpCC",0,0;
    SetTmSignal "A1DumpPump",1,0;
    SetTmSignal "A1Dump",0,0;
    SetTmSignal "A1AirCC",0,0;
    SetTmSignal "A1FluidClPCV",1,0;
    SetTmSignal "A1ByPump",1,0;
    SetTmSignal "A1SolvCC",1,0;
    SetTmSignal "A1Fluid",240,0;
    SetTmSignal "A1SolvCC",0,0.5;
    SetTmSignal "A1AirCC",1,0.05;
    SetTmSignal "A1AirCC",0,0.4;
    SetTmSignal "A1SolvCC",1,0.05;
    SetTmSignal "A1SolvCC",0,0.5;
    SetTmSignal "A1AirCC",1,0.05;
    SetTmSignal "A1Fluid",0,0.2;
    SetTmSignal "A1AirCC",0,0.4;
    SetTmSignal "A1FluidClPCV",0,0;
    SetTmSignal "A1DumpPump",0,0;
    SetTmSignal "A1ByPump",0,0;
    SetTmSignal "A1Con",1,0\Wait;
  ENDPROC
ENDMODULE
```

8.4.1.5 Create start position file

Description	This file defines the data according to motion programs' start position; when the c1position value over this data is the current program index, if the program is still not loaded, the robot system will send a signal to PLC and then stop the conveyor running. The conveyor will resume running until the motion program is loaded.
Limitation	The file must conform to the Comma Separated Value (CSV) convention.
File name	cnvpos.ndx
Default URL location	ftp://<IRC5>/<system>/HOME/programs/
Format	<program index>,< PntStrategy1 start position>,< PntStrategy2 start position>,<PntStrategy3 start position>,...
Example	

```
1,1500,1700,1000
2,1600,1700,1000
3,1600,1700,1000
4,1600,1700,1000
5,1500,1700,1000
```

8 System functionalities

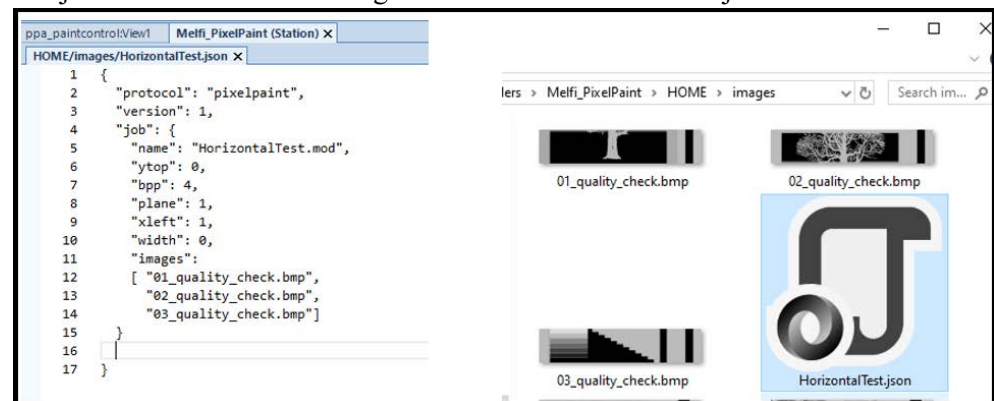
8.4 Create a new program

8.4.2 Pixel paint robot

Create a new program for pixel paint robot can basically take the reference of normal paint robot. Additionally, a job is defined as the images belonging to the motion program. The job is located in the images folder and has the job name with the ".json" extension.

Example:

Program name M1, which is in the programs folder, must have a corresponding M1.json file located in the images folder. The content of the job looks like this:



The name of the job is only used for reference to the program module. The images part of this are the important part. The print order is decided from the order in this job, first 01_quality_check, 02_quality_check and then 03_quality_check.

8.4.3 Clean robot

TBD

8.4.4 Flaming robot

Creating a new program for a flaming robot is the same as for a paint robot.

Following sub-sections are included in this section:

['Segment file'](#) on page 425

['Create motion program'](#) on page 426

['Create start position file'](#) on page 428

8.4.4.1 Segment file

Description

This file defines what program file(s) should be loaded and executed by what program index. If the segment file does not exist or an entry for a specific program index is missing, the default naming will be used: m<program index>.mod.

Note: The program extension (.mod) should not be included in the definitions. Alias names cannot be used. The segment system refers only to module filenames on the controller disk.

Changes in this file are detected automatically when changed on the controller.

Limitation

The file must conform to the Comma Separated Value (CSV) convention.

The segment limit for each program is 80 characters due to the RAPID string limitation.

File name

segment.ndx

Default URL location

ftp://<IRC5>/<system>/HOME/programs/

Format

<program index>, <program file>, <program file>, <program file>

Example

```

1 12,MEYKQX_FM_R11
2 20,MRHB_QBS_R11
3 21,MRHB_QBX_R11
4 22,MRHB_HBS_R11
5 23,MRHB_HBX_R11
6 24,HA6_QB_R11
7 25,HA6_HB_R11
8 26,HA2HK_QB_R11
9 27,HA2HK_HB_R11
10 29,HA2HE_QHB_FM_R11
11 32,MEEYKF_QBS_R11
12 36,MEEC_HB_FM_R11
13 34,MEFKB_HB_R11
14 35,MEEC_QB_R11

```

8.4.4.2 Create motion program**Description**

The motion program files contain the paint program or parts of it. Flaming program files contain a module with a 'main<name>' .

The module and file names must be identical, for example, the example program file must be named FRONT.MOD.

It is highly recommended that robtargets be defined as LOCAL CONST.

ShopFloorEditor supports this and can be configured as default on the module.

8 System functionalities

8.4 Create a new program

Using LOCAL routines for all other routines than the ‘main’ routine in your paint program is highly recommended. This will prevent name ambiguity between paint programs if the same routine name is used in different programs.

File name

m<program index>.mod or <name>.mod

Default URL location

ftp://<IRC5>/<system>/HOME/programs/

Example

```
MODULE MRHB_QBX_R11
IFM11

LOCAL CONST robtarget p10:=[[1000.18,1114.74,967.65],[0.464485,-0.4989,0.626253,0.378365],[0,0,1,0],[9e+009,9e+009,9e+009,9e+009,2113.05]];
LOCAL CONST robtarget p20:=[[67.06,1495.59,1931.25],[0.879488,-0.475877,0.00398406,-0.00506355],[-1,1,-2,0],[9e+009,9e+009,9e+009,9e+009,1577.8]];
LOCAL CONST robtarget p30:=[[214.14,305.76,1889.72],[0.867957,0.224879,-0.0579365,0.439003],[-1,1,-1,0],[9e+009,9e+009,9e+009,9e+009,1577.8]];
LOCAL CONST robtarget p40:=[[-181.92,57.87,1884.05],[0.867956,0.224881,-0.0579373,0.439003],[-1,2,-1,0],[9e+009,9e+009,9e+009,9e+009,1577.8]];
LOCAL CONST robtarget p50:=[[-554.18,324.82,1927.22],[0.821061,0.207858,-0.0722613,0.526718],[0,2,-1,0],[9e+009,9e+009,9e+009,9e+009,1577.8]];
LOCAL CONST robtarget p60:=[[-919.78,353.74,1928.05],[0.787137,0.211908,-0.059347,0.576185],[0,-1,-1,0],[9e+009,9e+009,9e+009,9e+009,1577.8]];
LOCAL CONST robtarget p70:=[[-1384.05,792.45,1792.48],[0.687401,-0.320447,-0.465585,0.456096],[-1,0,-2,-1],[9e+009,9e+009,9e+009,9e+009,1922.35]];
LOCAL CONST robtarget p80:=[[-990.34,647.95,2008.9],[0.776481,0.0635885,-0.176267,0.601634],[-1,0,-3,-1],[9e+009,9e+009,9e+009,9e+009,1922.35]];
LOCAL CONST robtarget p90:=[[-505.08,584.29,2094.57],[0.741686,0.301665,-0.296782,0.520404],[-1,2,-1,0],[9e+009,9e+009,9e+009,9e+009,2222.79]];
LOCAL CONST robtarget p100:=[[-170.96,307.89,2052.98],[0.717536,0.333878,-0.544475,0.277877],[-1,1,0,0],[9e+009,9e+009,9e+009,9e+009,2222.79]];
LOCAL CONST robtarget p110:=[[-105.73,142.28,2005.84],[0.760457,0.206505,-0.565268,0.243992],[-1,1,0,0],[9e+009,9e+009,9e+009,9e+009,1577.8]];
LOCAL CONST robtarget p120:=[[283.27,59.07,1949.43],[0.495268,0.612903,-0.31513,0.528916],[-1,1,0,0],[9e+009,9e+009,9e+009,9e+009,1577.8]];
LOCAL CONST robtarget p130:=[[-211.507,680.645,2194.97],[0.49546,0.622014,-0.318203,0.516106],[0,1,1,0],[9e+009,9e+009,9e+009,9e+009,1577.8]];
LOCAL CONST robtarget p140:=[[-391.75,590.761,2231.54],[0.380017,0.695515,-0.438994,0.423239],[0,0,1,0],[9e+009,9e+009,9e+009,9e+009,1577.8]];
LOCAL CONST robtarget p150:=[[-578.48,638.18,2229.85],[0.439876,0.628849,-0.514948,0.381951],[0,0,1,0],[9e+009,9e+009,9e+009,9e+009,1577.8]];
LOCAL CONST robtarget p160:=[[-807.138,668.931,2223.45],[0.460849,0.61366,-0.501712,0.400077],[0,0,1,0],[9e+009,9e+009,9e+009,9e+009,1577.8]];
LOCAL CONST robtarget p170:=[[-1134.5,703.449,2219.11],[0.481643,0.528599,-0.554029,0.425038],[0,0,1,0],[9e+009,9e+009,9e+009,9e+009,1577.8]];
LOCAL CONST robtarget p180:=[[-1327.42,1064.56,2234.13],[0.664961,0.263103,-0.68527,0.337869],[0,0,0,0],[9e+009,9e+009,9e+009,9e+009,1577.8]];
LOCAL CONST robtarget p190:=[[-914.49,1085.35,2200.03],[0.187874,0.731791,-0.319206,0.572095],[0,0,1,0],[9e+009,9e+009,9e+009,9e+009,2222.79]];
LOCAL CONST robtarget p200:=[[-1142.29,609.507,2214.37],[0.368992,0.627378,-0.46963,0.499688],[0,0,1,0],[9e+009,9e+009,9e+009,9e+009,2222.79]];
LOCAL CONST robtarget p210:=[[-1583.43,639.937,2211.53],[0.368993,0.627378,-0.469628,0.499689],[0,0,1,0],[9e+009,9e+009,9e+009,9e+009,2222.79]];
LOCAL CONST robtarget p220:=[[-1787.27,540.336,2215.64],[0.35288,0.636581,-0.456738,0.511499],[0,0,1,0],[9e+009,9e+009,9e+009,9e+009,2222.79]];
LOCAL CONST robtarget p230:=[[-2214.63,529.991,2229.67],[0.592191,0.423165,-0.632221,0.265589],[0,0,1,0],[9e+009,9e+009,9e+009,9e+009,2222.79]];
LOCAL CONST robtarget p240:=[[-2251.43,166.44,1980.59],[0.70173,0.213952,-0.490695,0.470126],[0,0,1,0],[9e+009,9e+009,9e+009,9e+009,2222.79]];
LOCAL CONST robtarget p250:=[[-2053.61,99.39,1971.49],[0.398604,0.49398,-0.264024,0.726216],[0,0,-2,0],[9e+009,9e+009,9e+009,9e+009,2222.79]];
LOCAL CONST robtarget p260:=[[-1970.28,326.31,2024.72],[0.398599,0.49398,-0.264026,0.726218],[0,0,-2,0],[9e+009,9e+009,9e+009,9e+009,2222.79]];
LOCAL CONST robtarget p270:=[[-1691.58,535.455,2015.81],[0.588473,0.359349,-0.15517,0.707453],[0,-1,-2,0],[9e+009,9e+009,9e+009,9e+009,2222.79]];
LOCAL CONST robtarget p280:=[[-900.362,571.455,2014.93],[0.650911,0.331881,-0.169697,0.661342],[0,2,-1,0],[9e+009,9e+009,9e+009,9e+009,2222.79]];
LOCAL CONST robtarget p290:=[[-602.18,1008.29,1919.58],[0.450167,0.562153,0.118863,0.683525],[0,2,-1,0],[9e+009,9e+009,9e+009,9e+009,2222.79]];
LOCAL CONST robtarget p300:=[[-905.451,340.706,1889.69],[0.627907,0.356063,-0.0099029,0.691993],[0,2,-1,0],[9e+009,9e+009,9e+009,9e+009,2222.79]];
LOCAL CONST robtarget p310:=[[-1646.74,280.93,1928.81],[0.651101,0.308049,0.0431548,0.692323],[0,-1,-1,0],[9e+009,9e+009,9e+009,9e+009,2222.79]];
LOCAL CONST robtarget p320:=[[-2075.3,280.93,1928.77],[0.6511,0.308049,0.043147,0.692324],[0,-1,-1,0],[9e+009,9e+009,9e+009,9e+009,2222.79]];
LOCAL CONST robtarget p330:=[[-2426.86,381.16,1868.72],[0.693175,-0.195792,-0.42253,0.550129],[0,-1,-2,0],[9e+009,9e+009,9e+009,9e+009,2222.79]];

PROC mainMRHB_QBX_R11()
MoveAbsJ jHomePos,vPaint2000,fine,tUserTool0;
WaitObj wobjtrack;
SetDO doFlame_3,0;
WaitObj wobjtrack\RelDist:=1800;
!
ClkReset clock1;
ClkStart clock1;

PaintL p20,v600,z50,FM_Fire_100\Wobj:=wobjtrack;
PaintL p30,v600,z50,FM_Fire_100\Wobj:=wobjtrack;
PaintL p40,v600,z50,FM_Fire_100\Wobj:=wobjtrack;
PaintL p50,v600,z50,FM_Fire_100\Wobj:=wobjtrack;
PaintL p60,v600,z50,FM_Fire_100\Wobj:=wobjtrack;
PaintL p70,v600,z50,FM_Fire_100\Wobj:=wobjtrack;
PaintL p80,v600,z50,FM_Fire_100\Wobj:=wobjtrack;
PaintL p90,v600,z50,FM_Fire_100\Wobj:=wobjtrack;
PaintL p100,v600,z50,FM_Fire_100\Wobj:=wobjtrack;
PaintL p110,v600,z50,FM_Fire_100\Wobj:=wobjtrack;
PaintL p120,v600,z50,FM_Fire_100\Wobj:=wobjtrack;
PaintL p130,v600,z50,FM_Fire_100\Wobj:=wobjtrack;
PaintL p140,v600,z50,FM_Fire_100\Wobj:=wobjtrack;
PaintL p150,v600,z50,FM_Fire_100\Wobj:=wobjtrack;
PaintL p160,v600,z50,FM_Fire_100\Wobj:=wobjtrack;
PaintL p170,v600,z50,FM_Fire_100\Wobj:=wobjtrack;
PaintL p180,v600,z50,FM_Fire_100\Wobj:=wobjtrack;

!
IF cIPosition>2.6 THEN
PaintL p190,v600,z50,FM_Fire_100\Wobj:=wobjtrack;
ELSE
PaintL p180,v600,FINE,FM_Fire_100\Wobj:=wobjtrack;
WaitObj wobjtrack\RelDist:=2600;
ENDIF
!
```



```

PaintL p190,v600,z50,FM_Fire_100\WObj:=wobjtrack;
PaintL p200,v600,z50,FM_Fire_100\WObj:=wobjtrack;
PaintL p210,v600,z50,FM_Fire_100\WObj:=wobjtrack;
PaintL p220,v600,z50,FM_Fire_100\WObj:=wobjtrack;
PaintL p230,v600,z50,FM_Fire_100\WObj:=wobjtrack;
PaintL p240,v600,z50,FM_Fire_100\WObj:=wobjtrack;
PaintL p250,v600,z50,FM_Fire_100\WObj:=wobjtrack;
PaintL p260,v600,z50,FM_Fire_100\WObj:=wobjtrack;
PaintL p270,v600,z50,FM_Fire_100\WObj:=wobjtrack;
PaintL p280,v600,z50,FM_Fire_100\WObj:=wobjtrack;
PaintL p290,v600,z50,FM_Fire_100\WObj:=wobjtrack;
PaintL p300,v600,z50,FM_Fire_100\WObj:=wobjtrack;
PaintL p310,v600,z50,FM_Fire_100\WObj:=wobjtrack;
PaintL p320,v600,z50,FM_Fire_100\WObj:=wobjtrack;
PaintL p330,v600,z50,FM_Fire_100\WObj:=wobjtrack;

MoveAbsJ jHomePos\NoEOffs,vPaint600,fine,tUserTool0;
ClkStop c1ock1;
Dropwobj wobjtrack;
Tperase;
Tpwrite "cycle time is "\num:=ClkRead(c1ock1);
CnvSync \Off;
TPWrite "MRHG_QBX_R_122 : "\num:=ClkRead(c1ock1);
ErrWrite "MRHG_QBX_R_122 : "+ValToStr(ClkRead(c1ock1)),"";

ENDPROC

ENDMODULE

```

8.4.4.3 Create start position file

Description	This file defines the data according to motion programs' start position; when the c1position value over this data is the current program index, if the program is still not loaded, the robot system will send a signal to PLC and then stop the conveyor running. The conveyor will resume running until the motion program is loaded.
Limitation	The file must conform to the Comma Separated Value (CSV) convention.
File name	cnvpos.ndx
Default URL location	ftp://<IRC5>/<system>/HOME/programs/
Format	<program index>,< PntStrategy1 start position>,< PntStrategy2 start position>,<PntStrategy3 start position>,...

8 System functionalities

8.4 Create a new program

Example

```
1,1500,1700,1000  
2,1600,1700,1000  
3,1600,1700,1000  
4,1600,1700,1000  
5,1500,1700,1000
```

8.4.5 Dispense robot

TBD

Appendix

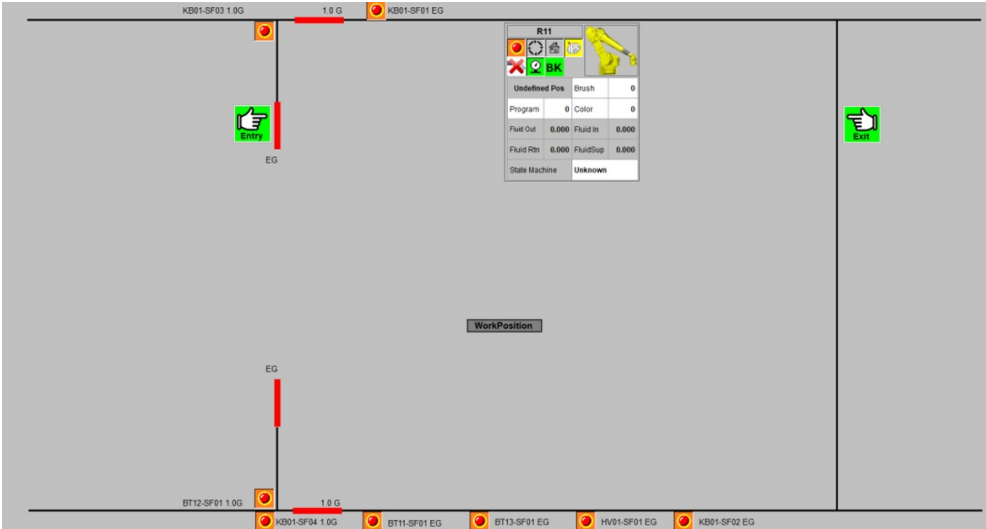
F (6.1.5)

Main windows

FD (Feather Duster)



PixelPaint



Sealing



Revision Log

3HNA035518-001 Rev.1.5 - Reference manual, PaintControls

Revision 1.5 - 250311

1. 3.3.11.6 Flaming robot system parameter updated. (61)
2. 7.2.2.4. Flaming robot IO signals added. (256)
3. 7.3.4 Flame robot command modified. (283)
4. 7.4.1.4 Flaming robot file structure updated. (291)
5. 7.4.2.4 Flaming robot program structure modified. (298)
6. 7.4.3.4 Flaming robot user data deleted. (305)
7. 7.5.5 Flaming robot function added. (340)
8. 8.1.4 Flaming robot workflow modified. (408)
9. 8.4.4 Flaming robot (Creating new program) updated. (425)

Revision 1.4 - 250124

1. 3.2 Modified Time Synchronization - [Disable windows Time service]. (23)
2. 6.3.2.1 Updated 1K paint system Process image. (169)
3. Deleted pipe symbol and description in the symbol table. (170)
4. 6.3.11 Added interlock chapter. (209)
5. 6.6.5 Added IO chapter. (231)
6. Image modified-
 - a. 6.3.2.1 Valve (170)
 - b. 6.3.2.1 Pump (170)
 - c. 6.3.2.1 Atomizer signal (170)
 - d. 6.3.2.1 PCV (170)
 - e. 6.3.2.1 Air Supply (170)
 - f. 6.3.2.1 Solvent Supply (170)
 - g. 6.3.2.1 Paint Supply (171)
 - h. 6.3.2.1 Dump (171)
 - i. 6.3.2.1 CBS CCV (174)

-
- j. 6.3.2.1 2KP CCV (173)
 - k. 6.3.2.1 1KP RCC (172)
 - l. 6.3.2.1 1KP CCV (172)
 - m. 6.6.5 IO (231)
 - n. 6.3.11 RobotInterlock (209)

Revision 1.3 - 240927

- 1. 3.2 Time Synchronization modified in detail. (20)
- 2. 3.5 HMI system installation contents inserted. (62)
- 3. 3.5 Install software images added. (62)
- 4. 6.5.6 and 6.5.7 Security Setting and Miscellaneous Setting images with table updated. (223)

Revision 1.2- 240906

Robot UI Updates:

- 1. Station Mode (124). This symbol appears in teach mode.  This symbol appears in unknown status. 
- 2. 6.2.4.1 Image of vision system status, both 3D and BK status changed along with signal status table.(132)
- 3. 6.3.1 Positions Image modified.(136)
- 4. 6.3.2.2 PxP Process Image modified.(148)
- 5. 6.3.2.2 PxP State machine Image modified. (148)
- 6. 6.3.4 Removed External Material and added PxP Service tab. (167)
- 7. 6.3.2 Application simplification of process view.(141)
- 8. 6.3.6 Backup five images with details(173)
- 9. 6.3.8 Brush Editor (179)
- 10. 6.5.6 Security: Two images added.(197)

Revision 1.1- 240827

Robot Updates:

- 1. 7.3.2 Paint robot program structure. (245)
- 2. 7.3.2.1 Cavity transition state. (245)

-
3. 7.3.2.2 “Go Ready State” command. (246)
 4. 7.3.2.3 Clean head (Day clean). (247)
 5. 7.3.2.4 MC suspend handling. (248)
 6. 7.3.1.1 Paintware command. (228)

Revision 1.0- 240430

First release.

This page is intentionally left blank.

List of Imported Graphics

Graphic files imported in manual:

Pg.:	Illustration:	Pg.:	Illustration:	Pg.:	Illustration:
1	../Images/Paint controls loading page banner.png	updates/Set time server_3.jpg	3.3.5.4.Choose the robot license file.png		
1	../Images/Paint controls label.png	26 ../Images/Installation/ Updates/Time synchronization updates/Set time client_1.jpg	36 ../Images/Installation/ 3.3.5.5.Choose options the robot.png		
2	FrontImage.jpg @ 300 dpi	27 ../Images/Installation/ Updates/Time synchronization updates/Set time client_2.jpg	37 ../Images/Installation/ 3.3.5.6.Select options in "Applications".png		
10	../Images/Warning.JPG	27 ../Images/Installation/ Updates/Time synchronization updates/Set time client_3.jpg	37 ../Images/Installation/3.3.5.7. Download to controller.png		
16	../Images/Warning.JPG	28 ../Images/Installation/ Updates/Time synchronization updates/Siemens S7-1500 Series.jpg	38 ../Images/Installation/3.3.6 Modify the existing system.png		
16	../Images/Important.JPG	28 ../Images/Installation/ Updates/Time synchronization updates/Teach pendant time setting.jpg	38 ../Images/Installation/3.3.7 Add-In options.png		
17	../Images/HMI/Ch-1.1.JPG	29 ../Images/Installation/3.3.1 Hardware description.png	40 ../Images/Installation/ 3.3.8.1.Setup UAS Login menu.png		
17	../Images/HMI/Ch-1.2.JPG	30 ../Images/Installation/ 3.3.1.1.The TPU should connect with the robot controller.png	40 ../Images/Installation/ 3.3.8.2.Connect to Controller, Get Write access.png		
19	../Images/Installation/ Updates/Time synchronization updates/Synchronization structure.jpg	30 ../Images/Installation/ 3.3.1.2.The X8 connection in Manual or Auto mode selected.png	41 ../Images/Installation/ 3.3.8.3.Select Authenticate + "Edit User Accounts".png		
20	../Images/Installation/ Updates/Time synchronization updates/Access advanced settings.jpg	31 ../Images/Installation/ 3.3.2.1.Exterior station layout overview (BC exterior station).png	41 ../Images/Installation/ 3.3.8.4.Select "Groups", add new group.png		
21	../Images/Installation/ Updates/Time synchronization updates/Choose rule type.jpg	32 ../Images/Installation/ 3.3.2.1.Interior station layout overview (BC interior station).png	41 ../Images/Installation/ 3.3.8.5.Select the groups access.png		
21	../Images/Installation/ Updates/Time synchronization updates/Choose program and protocol.jpg	33 ../Images/Installation/3.3.2.3 Installation package name.png	42 ../Images/Installation/ 3.3.8.6.Select User, create a new user.jpg		
22	../Images/Installation/ Updates/Time synchronization updates/Choose Profiles.jpg	34 ../Images/Installation/ 3.3.4.1.Click the button "Install".png	42 ../Images/Installation/ 3.3.8.7.Set the new user account and password.png		
22	../Images/Installation/ Updates/Time synchronization updates/Specify Name and Description.jpg	34 ../Images/Installation/ 3.3.4.2.Select the package need install and click "open", installed to Robotstudio.png	42 ../Images/Installation/ 3.3.8.8.Set the new user access, choose the groups which need to access.png		
23	../Images/Updates R24D/3.2 Open Services.png	35 ../Images/Installation/ 3.3.4.3.After installed, found in the left "Add-Ins" window.png	42 ../Images/Important.JPG		
23	../Images/Updates R24D/ Select Windows Time service.png	35 ../Images/Installation/ 3.3.5.1.Click "Installation Manager".png	43 ../Images/Installation/ 3.3.9.1.Configuration Safety user.png		
24	../Images/Updates R24D/ Disable Windows Time Service.png	36 ../Images/Installation/ 3.3.5.2.Click network to select the robot IP.png	44 ../Images/Installation/ 3.3.9.2.Configuration safety network.1.png		
24	../Images/Installation/ Updates/Time synchronization updates/Net time setup.jpg	36 ../Images/Installation/ 3.3.5.3.Add or replace Robotware, IPS, system add-in package..png	44 ../Images/Installation/ 3.3.9.2.Configuration safety network.2.png		
24	../Images/Installation/ Updates/Time synchronization updates/Install NetTime Software.jpg	36 ../Images/Installation/	45 ../Images/Installation/ 3.3.9.3.Ethernet or IP.1.png		
25	../Images/Installation/ Updates/Time synchronization updates/Set time server_1.jpg		45 ../Images/Installation/ 3.3.9.4.Input.2.png		
25	../Images/Installation/ Updates/Time synchronization updates/Set time server_2.jpg		46 ../Images/Installation/		
26	../Images/Installation/ Updates/Time synchronization				

List of imported graph

3.3.9.4.Configure safety address.1.png	77 ../Images/Installation/ Updates/3.5.8_02_Step1.png	88 ../Images/Installation/ Updates/3.5.12_02_Choose device.png
47 ../Images/Installation/ 3.3.9.5.Stop configuration.GStop.png	77 ../Images/Installation/ Updates/3.5.8_01_ABB IRC5 OPC Software Package.png @ 72 dpi	89 ../Images/Installation/ Updates/3.5.12_03_Add.png
47 ../Images/Installation/ 3.3.9.5.Stop configuration.AStop.png	78 ../Images/Installation/ Updates/3.5.8_03_Step2.png	89 ../Images/Installation/ Updates/3.5.12_03_SelectFile.png
48 ../Images/Installation/ 3.3.9.5.Stop configuration.EStop.png	78 ../Images/Installation/ Updates/3.5.8_04_Step3.png	90 ../Images/Installation/ Updates/ 3.5.13_01_OpenSoftware.png
48 ../Images/Installation/ 3.3.9.5.Stop configuration.DStop.png	78 ../Images/Installation/ Updates/3.5.8_05_Step4.png	90 ../Images/Installation/ Updates/3.5.13_02_Scan.png
49 ../Images/Installation/ 3.3.9.5.Synchronization.png	79 ../Images/Installation/ Updates/3.5.8_06_Step5.png	90 ../Images/Installation/ Updates/3.5.13_03_Select.png
50 ../Images/Installation/ 3.3.9.5.Tool configuration.png	79 ../Images/Installation/ Updates/3.5.8_07_Step6.png	93 ../Images/Operation and safety/2.2 Operation panel layout.1.png
50 ../Images/Installation/ 3.3.9.5.Download safety configuration.png	80 ../Images/Installation/ Updates/3.5.9_01_Step1.png	93 ../Images/Operation and safety/2.2 Operation panel layout.2.png
68 ../Images/Installation/ 3.5.5.2 Computer Resolution.png	80 ../Images/Installation/ Updates/ 3.5.9_06_Step6_Finish.png	94 ../Images/Operation and safety/2.3 Emergency stop button.png
69 ../Images/Installation/ 3.5.5.3 Setting the System Time Format.png	81 ../Images/Installation/ Updates/3.5.9_03_Step3.png	94 ../Images/Operation and safety/2.3 Control voltage switch.png
69 ../Images/Installation/ 3.5.5.4 .NET Framework 3.5 Installation.png	81 ../Images/Installation/ Updates/3.5.9_04_Step4.png	95 ../Images/Operation and safety/2.3 Conveyor bypass switch.png
70 ../Images/Installation/ 3.5.5.5 Power Supply Settings.png	82 ../Images/Installation/ Updates/3.5.9_05_Step5.png	95 ../Images/Operation and safety/2.3 Two position Key switch.png
70 ../Images/Installation/ 3.5.5.6 Ethernet configuration .png	82 ../Images/Installation/ Updates/3.5.9_06_Step6.png	96 ../Images/Operation and safety/2.3 Conveyor run button.png
71 ../Images/Installation/3.5.7 SQL Server Installation_1.png	83 ../Images/Installation/ Updates/ 3.5.9_06_Step6_Finish.png	96 ../Images/Operation and safety/2.3 Station Work mode switch.png
71 ../Images/Installation/3.5.7 SQL Server Installation_2.png	83 ../Images/Installation/3.5.10 RobView5 Installation_1.png	97 ../Images/Operation and safety/2.3 Maintenance, Work, and Clean.png
72 ../Images/Installation/3.5.7 SQL Server Installation_3.png	83 ../Images/Installation/3.5.10 RobView5 Installation_2.png	97 ../Images/Operation and safety/2.3 Robot mode button.png
72 ../Images/Installation/3.5.7 SQL Server Installation_4.png	84 ../Images/Installation/3.5.10 RobView5 Installation_3.png	98 ../Images/Operation and safety/2.3 Resume.png
73 ../Images/Installation/3.5.7 SQL Server Installation_5.png	84 ../Images/Installation/3.5.10 RobView5 Installation_4.png	99 ../Images/Operation and safety/2.3 Station mode button.png
73 ../Images/Installation/3.5.7 SQL Server Installation_6.png	85 ../Images/Installation/3.5.11 Redis Database Installation_1.png	100 ../Images/Operation and safety/2.3 Cycle start button.png
74 ../Images/Installation/3.5.7 SQL Server Installation_7.png	85 ../Images/Installation/3.5.11 Redis Database Installation_2.png	100 ../Images/Operation and safety/2.3 HV enable button.png
74 ../Images/Installation/3.5.7 SQL Server Installation_8.png	85 ../Images/Installation/3.5.11 Redis Database Installation_3.png	100 ../Images/Operation and safety/2.3 Lamp test button.png
75 ../Images/Installation/3.5.7 SQL Server Installation_9.png	86 ../Images/Installation/3.5.11 Redis Database Installation_4.png	101 ../Images/Operation and safety/2.3 Fault reset button.png
75 ../Images/Installation/3.5.7 SQL Server Installation_13.png	86 ../Images/Installation/3.5.12 SQL Server Management configuration instruction_1.png	101 ../Images/Operation and safety/3.1.1 Work mode.png
75 ../Images/Installation/3.5.7 SQL Server Installation_10.png	87 ../Images/Installation/3.5.12 SQL Server Management configuration instruction_2.png	102 ../Images/Operation and safety/3.1.2 Maintenance mode.png
76 ../Images/Installation/3.5.7 SQL Server Installation_11.png	87 ../Images/Installation/3.5.12 SQL Server Management configuration instruction_3.png	102 ../Images/Operation and safety/3.1.3 Cleaning mode.png
76 ../Images/Installation/3.5.7 SQL Server Installation_12.png	88 ../Images/Installation/3.5.12 SQL Server Management configuration instruction_4.png	

103	../Images/Operation and safety/3.2.1 Station Auto Mode.png	Toolbox3D.png	142	../Images/HMI/Tick.png	
103	../Images/Operation and safety/3.2.2 Ghost Vehicle Administration.png	136	../Images/Warning.JPG	142	../Images/HMI/Cross.png
103	../Images/Operation and safety/3.2.3 Station Manual Mode.png	136	../Images/HMI/Important.JPG	142	../Images/HMI/Cross.png
104	../Images/Operation and safety/3.3 Robot operating mode.png	138	../Images/HMI/6.1.1.png	142	../Images/HMI/Cross.png
105	../Images/Operation and safety/4. Station light indicator.png	139	../Images/HMI/6.1.2.1.png	142	../Images/HMI/Cross.png
106	../Images/Operation and safety/5.1 Safety layout.png	139	../Images/HMI/Dongle.jpg	142	../Images/HMI/Cross.png
112	../Images/Operation and safety/5.3.1 SafeMove.jpg	139	../Images/HMI/Language.jpg	142	../Images/HMI/Robot/Function.jpg
112	../Images/Operation and safety/5.3.2 Light Barrier.png	139	../Images/HMI/6.1.2.2.png	142	../Images/HMI/Cross.png
112	../Images/Operation and safety/5.3.2 Muting Operation.png	140	../Images/HMI/login.png	142	../Images/HMI/Tick.png
113	../Images/Operation and safety/5.3.2 Shielded Sensor.png	140	../Images/HMI/6.1.2.3.png	142	../Images/HMI/Cross.png
113	../Images/Operation and safety/5.3.2 Operation of the Shielded Sensor.png	140	../Images/HMI/Shift.png	142	../Images/HMI/Cross.png
114	../Images/Operation and safety/5.3.3 Emergency stops.png	141	../Images/HMI/Tick.png	142	../Images/HMI/Cross.png
115	../Images/Operation and safety/5.3.4 HV safety functions.png	141	../Images/HMI/Tick.png	142	../Images/HMI/Cross.png
118	../Images/Control System/1 Introduction.png	141	../Images/HMI/Tick.png	142	../Images/HMI/Robot/Function.jpg
120	../Images/Control System/2.1 Exterior Station .png	141	../Images/HMI/Tick.png	142	../Images/HMI/Cross.png
120	../Images/Control System/2.1 Paint Robot.png	141	../Images/HMI/Cross.png	142	../Images/HMI/Cross.png
121	../Images/Control System/2.2 Exterior Station .png	141	../Images/HMI/Tick.png	142	../Images/HMI/Tick.png
121	../Images/Control System/2.2 Paint Robot.png	141	../Images/HMI/Tick.png	142	../Images/HMI/Tick.png
122	../Images/Control System/2.3 Sealing station.png	141	../Images/HMI/Tick.png	142	../Images/HMI/Robot/Function.jpg
123	../Images/Control System/3. Type Verification.png	141	../Images/HMI/Cross.png	142	../Images/HMI/Tick.png
125	../Images/Control System/4.1.2 Type and Vehicle Data.png	141	../Images/HMI/Tick.png	142	../Images/HMI/Tick.png
126	../Images/Control System/4.1.3 Conveyor Equipment.png	141	../Images/HMI/Tick.png	142	../Images/HMI/Tick.png
127	../Images/Control System/4.1.5. Fire Protection.png	141	../Images/HMI/Cross.png	142	../Images/HMI/Tick.png
132	../Images/Control System/5 Blow Off.png	141	../Images/HMI/View/Zone.jpg	142	../Images/HMI/Tick.png
132	../Images/Control System/5. top of the vehicle body.png	141	../Images/HMI/Tick.png	142	../Images/HMI/Tick.png
133	../Images/Control System/6.1 3D Camera system.png	141	../Images/HMI/Tick.png	142	../Images/HMI/Tick.png
134	../Images/Control System/6.2 BK Laser System.png	141	../Images/HMI/Tick.png	142	../Images/HMI/Tick.png
135	../Images/Control System/6.3 Application.jpg	141	../Images/HMI/Tick.png	142	../Images/HMI/Robot/Backup.jpg
		141	../Images/HMI/View/Production.jpg	142	../Images/HMI/Tick.png
		141	../Images/HMI/Cross.png	142	../Images/HMI/Tick.png
		141	../Images/HMI/Tick.png	142	../Images/HMI/Tick.png
		141	../Images/HMI/Cross.png	142	../Images/HMI/Tick.png
		141	../Images/HMI/Cross.png	142	../Images/HMI/Tick.png
		141	../Images/HMI/Tick.png	142	../Images/HMI/Tick.png
		141	../Images/HMI/Cross.png	142	../Images/HMI/Tick.png
		141	../Images/HMI/Tick.png	142	../Images/HMI/Robot/ResetPP.jpg
		141	../Images/HMI/Tick.png	142	../Images/HMI/Tick.png
		141	../Images/HMI/View/Vision.jpg	142	../Images/HMI/Tick.png
		141	../Images/HMI/6.1.3.png	142	../Images/HMI/Cross.png
		142	../Images/HMI/Tick.png	142	../Images/HMI/Cross.png
		142	../Images/HMI/Tick.png	142	../Images/HMI/Cross.png
		142	../Images/HMI/Tick.png	142	../Images/HMI/Cross.png
		142	../Images/HMI/Tick.png	142	../Images/HMI/Robot/Brush Editor.jpg
		142	../Images/HMI/Tick.png	142	../Images/HMI/Tick.png
		142	../Images/HMI/Robot/Position.jpg	142	../Images/HMI/Tick.png
		142	../Images/HMI/Tick.png	142	../Images/HMI/Tick.png
		142	../Images/HMI/Tick.png	142	../Images/HMI/Cross.png
		142	../Images/HMI/Tick.png	142	../Images/HMI/Cross.png
		142	../Images/HMI/Tick.png	142	../Images/HMI/Robot/LogRefresh.jpg
		142	../Images/HMI/Robot/Application.jpg	142	../Images/HMI/Tick.png
		142	../Images/HMI/Robot/	142	../Images/HMI/Cross.png
		142	../Images/HMI/Robot/	142	../Images/HMI/Cross.png

List of imported graph

142	../Images/HMI/Cross.png	143	../Images/HMI/Setting/Skid	144	../Images/HMI/Cross.png
142	../Images/HMI/Cross.png		slip sensor check.jpg	144	../Images/HMI/Cross.png
142	../Images/HMI/Cross.png	143	../Images/HMI/Tick.png	144	../Images/HMI/Cross.png
142	../Images/HMI/Robot/ ShaftLock.jpg	143	../Images/HMI/Cross.png	144	../Images/HMI/Cross.png
143	../Images/HMI/Tick.png	143	../Images/HMI/Cross.png	144	../Images/HMI/Cross.png
143	../Images/HMI/Tick.png	143	../Images/HMI/Cross.png	144	../Images/HMI/Diagnose/ Mixroom.jpg
143	../Images/HMI/Tick.png	143	../Images/HMI/Cross.png	144	../Images/HMI/Tick.png
143	../Images/HMI/Tick.png	143	../Images/HMI/Setting/ Parameter.jpg	144	../Images/HMI/Tick.png
143	../Images/HMI/Tick.png	143	../Images/HMI/Tick.png	144	../Images/HMI/Tick.png
143	../Images/HMI/Control/ FIFO.jpg	143	../Images/HMI/Tick.png	144	../Images/HMI/Tick.png
143	../Images/HMI/Tick.png	143	../Images/HMI/Tick.png	144	../Images/HMI/Tick.png
143	../Images/HMI/Tick.png	143	../Images/HMI/Tick.png	144	../Images/HMI/Report/ Production.jpg
143	../Images/HMI/Tick.png	143	../Images/HMI/Tick.png	144	../Images/HMI/Tick.png
143	../Images/HMI/Tick.png	143	../Images/HMI/Tick.png	144	../Images/HMI/Tick.png
143	../Images/HMI/Tick.png	143	../Images/HMI/Setting/ Alarm.jpg	144	../Images/HMI/Cross.png
143	../Images/HMI/Tick.png	143	../Images/HMI/Tick.png	144	../Images/HMI/Cross.png
143	../Images/HMI/Control/ Simulation.jpg	143	../Images/HMI/Tick.png	144	../Images/HMI/Tick.png
143	../Images/HMI/Tick.png	143	../Images/HMI/Tick.png	144	../Images/HMI/Tick.png
143	../Images/HMI/Tick.png	143	../Images/HMI/Tick.png	144	../Images/HMI/Report/ Consumption.jpg
143	../Images/HMI/Tick.png	143	../Images/HMI/Tick.png	144	../Images/HMI/Tick.png
143	../Images/HMI/Tick.png	143	../Images/HMI/Tick.png	144	../Images/HMI/Tick.png
143	../Images/HMI/Tick.png	143	../Images/HMI/Setting/ Security.jpg	144	../Images/HMI/Tick.png
143	../Images/HMI/Tick.png	143	../Images/HMI/Tick.png	144	../Images/HMI/Tick.png
143	../Images/HMI/Control/ Bypass.jpg	143	../Images/HMI/Tick.png	144	../Images/HMI/Tick.png
143	../Images/HMI/Tick.png	143	../Images/HMI/Tick.png	144	../Images/HMI/Tick.png
143	../Images/HMI/Cross.png	143	../Images/HMI/Tick.png	144	../Images/HMI/Report/ Operation.jpg
143	../Images/HMI/Cross.png	143	../Images/HMI/Tick.png	144	../Images/HMI/Tick.png
143	../Images/HMI/Cross.png	143	../Images/HMI/Tick.png	144	../Images/HMI/Tick.png
143	../Images/HMI/Cross.png	143	../Images/HMI/Setting/ Miscellaneous.jpg	144	../Images/HMI/Tick.png
143	../Images/HMI/Cross.png	144	../Images/HMI/Tick.png	144	../Images/HMI/Tick.png
143	../Images/HMI/Control/ Opener.jpg	144	../Images/HMI/Tick.png	144	../Images/HMI/Tick.png
143	../Images/HMI/Tick.png	144	../Images/HMI/Tick.png	144	../Images/HMI/Tick.png
143	../Images/HMI/Tick.png	144	../Images/HMI/Tick.png	144	../Images/HMI/Report/ Alarm.jpg
143	../Images/HMI/Tick.png	144	../Images/HMI/Tick.png	144	../Images/HMI/Tick.png
143	../Images/HMI/Tick.png	144	../Images/HMI/Diagnose/ Alarm.jpg	144	../Images/HMI/Cross.png
143	../Images/HMI/Tick.png	144	../Images/HMI/Tick.png	144	../Images/HMI/Tick.png
143	../Images/HMI/Setting/ Type.jpg	144	../Images/HMI/Tick.png	144	../Images/HMI/Cross.png
143	../Images/HMI/Tick.png	144	../Images/HMI/Tick.png	144	../Images/HMI/Cross.png
143	../Images/HMI/Tick.png	144	../Images/HMI/Tick.png	144	../Images/HMI/Report/ CommandLog.jpg
143	../Images/HMI/Tick.png	144	../Images/HMI/Tick.png	144	../Images/HMI/Tick.png
143	../Images/HMI/Tick.png	144	../Images/HMI/Diagnose/ LAN.jpg	144	../Images/HMI/Tick.png
143	../Images/HMI/Cross.png	144	../Images/HMI/Tick.png	144	../Images/HMI/Tick.png
143	../Images/HMI/Cross.png	144	../Images/HMI/Tick.png	144	../Images/HMI/Tick.png
143	../Images/HMI/Setting/ Colour.jpg	144	../Images/HMI/Tick.png	144	../Images/HMI/Tick.png
143	../Images/HMI/Tick.png	144	../Images/HMI/Tick.png	144	../Images/HMI/Tick.png
143	../Images/HMI/Cross.png	144	../Images/HMI/Tick.png	144	../Images/HMI/Program/ RobotStudio.jpg
143	../Images/HMI/Cross.png	144	../Images/HMI/Tick.png	144	../Images/HMI/Tick.png
143	../Images/HMI/Cross.png	144	../Images/HMI/Diagnose/ Bus.jpg	144	../Images/HMI/Cross.png
143	../Images/HMI/Cross.png	144	../Images/HMI/Tick.png	144	../Images/HMI/Tick.png

144	../Images/HMI/Tick.png	152	../Images/HMI/Clean mode.png	166	../Images/HMI/6.3.1.2 Positions - PixelPaint2.jpg
144	../Images/HMI/Tick.png	152	../Images/HMI/Updates/Teach mode.png	166	../Images/HMI/6.3.1.2 Positions - PixelPaint1.jpg
144	../Images/HMI/Program/RobView.jpg	152	../Images/HMI/Updates/Unknown status.png	166	../Images/HMI/Updates/6.3.1.2 Positions - PixelPaint.png
145	../Images/HMI/Tick.png	153	../Images/HMI/1.eps	166	../Images/HMI/Warning.JPG
145	../Images/HMI/Tick.png	154	../Images/HMI/Robot name.png	167	../Images/HMI/6.3.1.3 Positions - Dispensing.jpg
145	../Images/HMI/Tick.png	154	../Images/HMI/Robot colour.png	167	../Images/HMI/Warning.JPG
145	../Images/HMI/Tick.png	154	../Images/HMI/Robot motor status.png	168	../Images/HMI/6.3.1.4 Positions - Cleaning2.jpg
145	../Images/HMI/Help/RobotUI.jpg	154	../Images/HMI/Robot cycle status.png	168	../Images/HMI/6.3.1.4 Positions - Cleaning1.jpg
145	../Images/HMI/Tick.png	154	../Images/HMI/Home.png	168	../Images/HMI/Warning.JPG
145	../Images/HMI/Tick.png	154	../Images/HMI/IPS.png	169	../Images/HMI/Updates/Production_Process view.png
145	../Images/HMI/Tick.png	155	../Images/HMI/Application Status_1.png	169	../Images/Updates R24D/6.3.2.1 Painting-1K paint system.png
145	../Images/HMI/Tick.png	155	../Images/HMI/Application Status_2.png	170	../Images/Updates R24D/6.3.2.1 Valve.png
145	../Images/HMI/Tick.png	155	../Images/HMI/Air Pressure.png	170	../Images/Updates R24D/6.3.2.1 Pump.png
145	../Images/HMI/Tick.png	155	../Images/HMI/High Voltage Status.png	170	../Images/Updates R24D/6.3.2.1 Atomizer signal.png
145	../Images/HMI/Tick.png	155	../Images/HMI/Potlife Status.png	170	../Images/Updates R24D/6.3.2.1 PCV.png
145	../Images/HMI/Help/Exit.jpg	156	../Images/HMI/Robot detail data.png	170	../Images/Updates R24D/6.3.2.1 Air Supply.png
145	../Images/HMI/AlarmButton.png	156	../Images/HMI/Sealing_Robot detail.png	170	../Images/Updates R24D/6.3.2.1 Solvent Supply.png
145	../Images/HMI/AlarmLine.png	156	../Images/HMI/Updates/6.2.1 PixelPaint robot detail data.png	171	../Images/Updates R24D/6.3.2.1 Paint Supply.png
146	../Images/HMI/6.1.5.png	156	../Images/HMI/Booth car data.png	171	../Images/Updates R24D/6.3.2.1 Dump.png
147	../Images/HMI/6.1.6.png	157	../Images/HMI/Swing door status.png	171	../Images/HMI/6.3 OverView.jpg
148	../Images/HMI/Additional window.png	157	../Images/HMI/Light barrier status.png	172	../Images/HMI/6.3.2.1 1KP CCV process.jpg
148	../Images/HMI/6.1.7.png	157	../Images/HMI/Sensor.png	172	../Images/Updates R24D/6.3.2.1 1KP RCC.png
149	../Images/HMI/Network.png	157	../Images/HMI/Emergency stop staus.png	173	../Images/HMI/RCC Valve.png
149	../Images/HMI/6.1.7.1.png	157	../Images/HMI/Entry or Exit release.png	173	../Images/HMI/2K paint system.png
149	../Images/HMI/Fire Alarm.png	158	../Images/HMI/6.2.1.png	173	../Images/Updates R24D/6.3.2.1 2KP CCV.png
149	../Images/HMI/Booth.png	159	../Images/HMI/6.2.2.png	174	../Images/HMI/Robot/CBS CCV process.jpg
150	../Images/HMI/Air Pressure.png	159	../Images/HMI/6.2.3.png	174	../Images/Updates R24D/6.3.2.1 CBS CCV.png
150	../Images/HMI/E-Stop.png	160	../Images/HMI/6.2.4.1.png	175	../Images/HMI/Fluid fill.png
150	../Images/HMI/6.1.7.2.png	160	../Images/HMI/Signal Status.png	175	../Images/HMI/Fluid consumption.png
150	../Images/HMI/Safety.png	161	../Images/HMI/6.2.4.2.png	175	../Images/HMI/Gripper.png
150	../Images/HMI/6.1.7.3.png	161	../Images/HMI/Signal Status.png	175	../Images/HMI/Cart.png
151	../Images/HMI/Conveyor.png	164	../Images/HMI/Updates/Position view.png	175	../Images/HMI/Needle.png
151	../Images/HMI/6.1.7.4.png	164	../Images/HMI/Warning.JPG	176	../Images/HMI/Updates/Application_Pixel Paint Process.png
151	../Images/HMI/Application.png	165	../Images/HMI/6.3.1 and 6.3.1.1 Positions.jpg		
151	../Images/HMI/6.1.7.5.png	165	../Images/HMI/Warning.JPG		
152	../Images/HMI/Robot.png				
152	../Images/HMI/PLC Power.png				
152	../Images/HMI/Atomatic mode.png				
152	../Images/HMI/Manual mode.png				
152	../Images/HMI/Maintenance mode.png				

List of imported graph

176	../Images/HMI/Updates/	Cleaning SB process.jpg	199	../Images/HMI/		
1KPP PxP State machine.png		185	../Images/Updates R24D/	Circulation.png		
177	../Images/HMI/1KI PXP	6.3.2.1 Valve.png	200	../Images/HMI/Doser.png		
Status machine.png		185	../Images/HMI/	200	../Images/HMI/Material	
178	../Images/HMI/Head	Pipe_Blue_1.png	200	Supply.png		
Cavities.png		185	../Images/HMI/Level	200	../Images/HMI/6.3.6	
178	../Images/HMI/Table.png	Sensor.png	201	Backup.jpg		
179	../Images/HMI/Updates/6.3	186	../Images/HMI/Tornado.png	201	../Images/HMI/Important.JPG	
1KP PxP Production.png		186	../Images/HMI/Fan.png	201	../Images/HMI/Updates/	
180	../Images/HMI/6.3.2.3	186	../Images/HMI/Sword	Warning.png		
Dispense IDFP.jpg		brush.png	201	../Images/HMI/Updates/		
180	../Images/Updates R24D/	186	../Images/HMI/Cleaning liquid	6.3.6.1 No Allowed.png		
6.3.2.1 Valve.png		tank.png	202	../Images/HMI/Updates/		
180	../Images/HMI/Doser	186	../Images/Updates R24D/	6.3.6.2 Ready.png		
pump.png		6.3.2.1 Pump.png	202	../Images/HMI/Updates/		
180	../Images/HMI/Pipe_Blue.png	187	../Images/HMI/	6.3.6.2 Ready 2.png		
180	../Images/HMI/Inlet	Run_green.png	203	../Images/HMI/Updates/6.3.6		
Pump.png		187	../Images/HMI/	Doing Backup.png		
181	../Images/HMI/Peltier.png	Run_yellow.png	203	../Images/HMI/Updates/6.3.6		
181	../Images/HMI/Glue Gun.png	187	../Images/HMI/Warning.JPG	To HMI.png		
181	../Images/HMI/Needle	187	../Images/HMI/6.3.3 Function	204	../Images/HMI/6.3.7 Reset	
Status.png		- Painting3.jpg	Program Pointer(PP).png	205	../Images/HMI/Save.png	
181	../Images/HMI/6.3.2.3	187	../Images/HMI/6.3.3 Function	205	../Images/HMI/Undo.png	
Dispense PAC.jpg		- Painting2.jpg	187	../Images/HMI/6.3.3 Function	205	../Images/HMI/Open.png
181	../Images/Updates R24D/	187	../Images/HMI/6.3.3 Function	205	../Images/HMI/6.3.3 Function	
6.3.2.1 Valve.png		Fill Clean.jpg	188	../Images/HMI/6.3.3.1	205	../Images/HMI/New Brush
181	../Images/HMI/	188	../Images/HMI/6.3.3.1	table.png		
Pipe_Black.png		Function Fill Clean.jpg	188	../Images/HMI/Warning.JPG	205	../Images/HMI/3 2.eps
182	../Images/HMI/Inlet	190	../Images/HMI/Fluid flow	206	../Images/HMI/Compare.png	
Pump.png		test.png	206	../Images/HMI/6.3.8 Brush		
182	../Images/HMI/Pressure	192	../Images/HMI/Brush factor	Editor_1.png		
sensor.png		by Percentage.png	206	../Images/HMI/History.png		
182	../Images/HMI/PVC	192	../Images/HMI/High voltage	206	../Images/HMI/6.3.8 Brush	
Regulator.png		test.png	206	Editor_2.png		
182	../Images/HMI/Proportional	193	../Images/HMI/HV Dynamic	206	../Images/HMI/Import.png	
Sensor.png		Test.png	206	../Images/HMI/Export.png		
182	../Images/HMI/Flow	194	../Images/HMI/High voltage	206	../Images/HMI/Copy.png	
meter.png		static Test.png	207	../Images/HMI/Updates/		
182	../Images/HMI/Temperature	194	../Images/HMI/IPS	BrushTable.jpg		
sensor.png		Calibration.png	208	../Images/HMI/6.3.9 Log		
182	../Images/HMI/Glue Gun.png	195	../Images/HMI/Updates/6.3.4	refresh (Command-IO).png		
182	../Images/HMI/Needle	Function - PixelPaint.png	209	../Images/HMI/6.3.10		
Status.png		195	../Images/HMI/Warning.JPG	Shaftlock - Bell (paint robot).png		
183	../Images/HMI/IDFP sealing	196	../Images/HMI/Warning.JPG	209	../Images/Updates R24D/	
system.png		196	../Images/HMI/Updates/	6.3.11 Interlock.png		
183	../Images/HMI/App	6.3.4.2 Brush test.png	211	../Images/HMI/2.eps		
Parameter Dispense Pac Sealing		197	../Images/HMI/Updates/	211	../Images/HMI/	
System.png		6.3.4.3 PxP Service.png	211	Save_Black_1.png		
184	../Images/HMI/6.3.2.4	197	../Images/HMI/Warning.JPG	211	../Images/HMI/	
Cleaning FD process.jpg		198	../Images/HMI/Updates/	Undo_Black_1.png		
184	../Images/HMI/Roller	6.3.4.4 Custom.png	198	../Images/HMI/6.4.1.1 Line		
motor.png		198	../Images/HMI/Warning.JPG	tracking station.png		
184	../Images/HMI/Detect	199	../Images/HMI/6.3.5.1	213	../Images/HMI/6.4.1.2 Stop &	
sensor.png		Function - IDFP.png	214	Go station.png		
185	../Images/HMI/De-ionizing	199	../Images/HMI/	214	../Images/HMI/6.4.2	
bar power.png		Temperature.png	215	Simulation.png		
185	../Images/HMI/Roller	199	../Images/HMI/Release	215	../Images/HMI/Ghost.png	
running.png		pressure.png	215	../Images/HMI/Car		
185	../Images/HMI/6.3.2.4		215	information.png		

215	../Images/HMI/Ghost parameter.png	229	../Images/HMI/Updates/6.6.1 Alarm display.png	Trajectory folder.png	
215	../Images/HMI/Save_Black_1.png	230	../Images/HMI/Updates/6.6.2 Network image.png	290	../Images/Robot/7.4.1.2 Pixel paint file structure.png
215	../Images/HMI/Undo_Black_1.png	230	../Images/HMI/Updates/6.6.3 Bus.png	291	../Images/Robot/Updates/Flame robot/Clean robot file structure.png
215	../Images/HMI/Start.png	230	../Images/HMI/6.6.4 Mixroom.png	291	../Images/Robot/Updates/Flame robot/Flaming robot file structure.png
215	../Images/HMI/Abort.png	231	../Images/Updates R24D/6.6.5 IO.png	294	../Images/Robot/Updates/Flame robot/Flaming robot file structure_1.png
216	../Images/HMI/Atomizer.png	233	../Images/HMI/6.7.1 Production.png	296	../Images/Robot/7.4.2.1 Paint robot program structure.png
216	../Images/HMI/Home.png	234	../Images/HMI/Updates/6.7.2 Consumption report.jpg	296	../Images/Robot/Updates/7.4.2.1 Paint robot program structure_1.png
216	../Images/HMI/High Voltage_Green.png	234	../Images/HMI/Updates/6.7.3 Operation report.jpg	297	../Images/Robot/Updates/7.4.2.1 Paint robot program structure_2.png
216	../Images/HMI/Robot.png	235	../Images/HMI/Updates/6.7.4 Alarm report .jpg	297	../Images/Robot/Updates/7.4.2.1 Paint robot program structure_2.png
216	../Images/HMI/Gripper_Green.png	236	../Images/HMI/Updates/6.7.4 Alarm report_2.jpg	297	../Images/Robot/7.4.2.1 Paint robot program structure.3.png
216	../Images/HMI/6.4.3 Bypass.png	236	../Images/HMI/Updates/6.7.5 Command report.jpg	298	../Images/Robot/Updates/Flame robot/Flaming robot program structure-1.png
217	../Images/HMI/Select image.png	237	../Images/HMI/Robostudio.png	299	../Images/Robot/Updates/Flame robot/Flaming robot program structure-2.png
217	../Images/HMI/Master.png	237	../Images/HMI/6.8.2 RobView.png	307	../Images/Robot/7.5.1.3 External Axis check.jpg
217	../Images/HMI/Select target position.png	238	../Images/HMI/Updates/6.8.3 IoT Gateway.png	307	../Images/Robot/7.5.1.4 Ghost conveyor for painting.jpg
217	../Images/HMI/Move.png	238	../Images/HMI/Updates/6.8.4 VMT.png	308	../Images/Robot/7.5.1.5 Conveyor halt for painting.jpg
217	../Images/HMI/Release.png	239	../Images/HMI/RobotUI.JPG	309	../Images/Important.JPG
217	../Images/HMI/Cycle on.png	239	../Images/HMI/Robot.JPG	310	../Images/Robot/7.5.1.7 Robot interlock.jpg
217	../Images/HMI/6.4.4 Opener.jpg	239	../Images/HMI/exit.JPG	312	../Images/Robot/7.5.1.10. CBC function.png
218	../Images/HMI/PLC.png	277	../Images/Robot/Updates/7.3.2 Paint robot program structure.png	313	../Images/Robot/7.5.2.1.1 Bellclean.1.jpg
218	../Images/HMI/PC.png	278	../Images/Robot/Updates/7.3.2.1 Cavity transition state.png	313	../Images/Robot/7.5.2.1.1 Bellclean.2.jpg
218	../Images/HMI/Save.png	279	../Images/Robot/Updates/7.3.2.2 "Go Ready State" command_1.png	314	../Images/Robot/7.5.2.1.1 Bellclean.3.jpg
218	../Images/HMI/Restore.png	279	../Images/Robot/Updates/7.3.2.2 "Go Ready State" command_2.png	314	../Images/Robot/7.5.2.1.1 Bellclean.4.jpg
218	../Images/HMI/Synchronization.png	281	../Images/Robot/Updates/7.3.2.2 "Go Ready State" command_3.png	314	../Images/Robot/7.5.2.1.1 Hubclean.1.jpg
219	../Images/HMI/6.5.1 Type.png	281	../Images/Robot/Updates/7.3.2.2 "Go Ready State" command_4.png	315	../Images/Robot/7.5.2.1.1 Hubclean.2.jpg
220	../Images/HMI/Updates/6.5.2 Colour (painting).png	282	../Images/Robot/Updates/7.3.2.2 "Go Ready State" command_5.png	315	../Images/Robot/7.5.2.1.1 Bellclean.3.jpg
221	../Images/HMI/6.5.3 Skid slip sensor check (Line tracking station).png	282	../Images/Robot/Updates/7.3.2.2 "Go Ready State" command_6.png	315	../Images/Robot/7.5.2.1.1 Hubclean.3.jpg
222	../Images/HMI/6.5.4 Parameter.png	284	../Images/Robot/7.4.1.1 Home Folder.png	316	../Images/Robot/7.5.2.1.1 Gunclean.1.jpg
223	../Images/HMI/Updates/6.5.5 Alarm setting .png	287	../Images/Robot/7.4.1.1	316	../Images/Robot/7.5.2.1.1 Gunclean.2.jpg
224	../Images/HMI/Updates/6.5.5 Alarm setting .jpg				
225	../Images/HMI/Updates/6.5.6 Security_1.png				
225	../Images/HMI/Updates_24.C/User.png				
226	../Images/HMI/6.5.6 Security_1.png				
226	../Images/HMI/Updates/security level.png				
226	../Images/HMI/Updates_24.C/6.5.6 Security.png				
227	../Images/HMI/Updates_24.C/Miscellaneous.png				
228	../Images/HMI/Warning.JPG				

List of imported graph

317	../Images/Robot/7.5.2.1.1	Collect diagnostic information.1.png	424	../Images/Robot/8.4.1.5	
Bellclean.3.jpg		389	../Images/Robot/7.7.2.1	Create start position file.jpg	
317	../Images/Robot/7.5.2.1.1	Collect diagnostic information.2.png	425	../Images/Robot/8.4.2. Pixel	
Gunclean.3.jpg		389	../Images/Robot/7.7.2.1.	paint robot.png	
318	../Images/Robot/7.5.2.1.1	Collect diagnostic information.3.jpg	426	../Images/Robot/Updates/	
Toolclean.1.jpg		390	../Images/Robot/7.7.2.1.	Flame robot/Segment file - Flaming	
318	../Images/Robot/7.5.2.1.1	Collect diagnostic information.4.jpg	robot.png		
Bellclean.3.jpg		390	../Images/Robot/7.7.2.3 With	427	../Images/Robot/Updates/
318	../Images/Robot/7.5.2.1.1	RobotStudio.jpg	Flame robot/Create motion program-	1.png	
Toolclean.2.jpg		391	../Images/Robot/7.7.2.4 With	427	../Images/Robot/Updates/
319	../Images/Robot/7.5.2.2	the paint TPU.1.jpg	Flame robot/Create motion program-	2.png	
Single Line 1KP CCV.jpg		391	../Images/Robot/7.7.2.4 With	428	../Images/Robot/Updates/
320	../Images/Robot/7.5.2.2	the paint TPU.2.jpg	Flame robot/Create motion program-	3.png	
Single Line 1KP RCC.jpg		392	../Images/Robot/7.7.2.5 With	429	../Images/Robot/8.4.1.5
320	../Images/Robot/7.5.2.2	main computer console.jpg	Create start position file.jpg	430	../Images/HMI/FD.jpg
Single Line 2KP CCV.jpg		392	../Images/Robot/7.7.2.6	430	../Images/HMI/PixelPaint.jpg
321	../Images/Robot/7.5.2.2	Record program pointer with	431	../Images/HMI/Sealing.jpg	
Single Line CBS CCV.jpg		RobotStudio.jpg			
322	../Images/Robot/7.5.2.2	393	../Images/Robot/7.7.2.6		
Single Line CBS CCV.png		Record program pointer with			
322	../Images/Robot/7.5.2.2	RobotStudio.1.jpg			
Single Line CBS RCC.jpg		394	../Images/Robot/7.7.2.7 Log		
323	../Images/Robot/7.5.2.2	conveyor signals.jpg			
PaintPush.jpg		394	../Images/Robot/7.7.2.8		
324	../Images/Robot/7.5.2.2	Save error log with RobotStudio.jpg			
PreStart MC.jpg		395	../Images/Robot/7.7.2.9		
326	../Images/Robot/7.5.2.4	Save signal diagnostic with			
Volume calculation.jpg		Robview.1.jpg			
330	../Images/Robot/7.5.2.7	395	../Images/Robot/7.7.2.9		
Vibration check function.png		Save signal diagnostic with			
331	../Images/Robot/7.5.2.7 Bell	Robview.2.jpg			
cup missing checking.png		396	../Images/Robot/7.7.2.9		
331	../Images/Robot/7.5.2.7	Save signal diagnostic with			
Pump pressure check .png		Robview.3.jpg			
332	../Images/Robot/7.5.2.7.4	396	../Images/Robot/7.7.2.9		
Fiber quality check .png		Save signal diagnostic with			
332	../Images/Robot/7.5.3	Robview.4.jpg			
PixelPaint Application functions.jpg		396	../Images/Robot/7.7.2.9		
333	../Images/Robot/7.5.3.1	Save signal diagnostic with			
Material change.jpg		Robview.5.jpg			
333	../Images/Robot/7.5.3.1	397	../Images/Robot/7.7.2.9		
Material change.1.jpg		Save signal diagnostic with			
336	../Images/Robot/7.5.3.2	Robview.6.jpg			
Nozzle test.jpg		405	../Images/Robot/8.1.1.1		
336	../Images/Robot/7.5.3.2	T_ROB1 task.jpg			
Horizontal test.jpg		406	../Images/Robot/8.1.1.3		
336	../Images/Robot/7.5.3.2	Color change task.jpg			
Horizontal test.1.jpg		408	../Images/Robot/Updates/		
336	../Images/Robot/7.5.3.2	Flame robot/8.1.4 T_ROB1 task.jpg			
Horizontal test.2.jpg		412	../Images/Robot/8.2.4 Repair		
337	../Images/Robot/7.5.3.2	Mode.jpg			
Nozzle test.jpg		420	../Images/Robot/8.4.1.1		
337	../Images/Robot/7.5.3.4 BK	Segment file .jpg			
sensor .jpg		421	../Images/Robot/8.4.1.2		
339	../Images/Robot/Updates/	Create motion program .jpg			
7.5.4 Clean robot functions.png		423	../Images/Robot/8.4.1.3		
341	../Images/Robot/Updates/	Create Brush table.jpg			
Flame robot/Burner D-6-280.png		423	../Images/Robot/8.4.1.4		
388	../Images/Robot/7.7.2.1	Create color change sequence .jpg			



Contact us

ABB AS, Robotics

Robotics and Discrete Automation

Nordlysvegen 7, N-4340 BRYNE, Norway

Box 265, N-4349 BRYNE, Norway

Telephone: +47 22 87 2000

ABB Engineering (Shanghai) Ltd.

Robotics and Discrete Automation

No.4528, Kangxin Highway,

Pudong New District,

Shanghai, 201319, P.R.China

Tel: +86 (0) 21 6105 6666

www.abb.com/robotics