Marcus Thum 1801321I

How-To Guide

Modularity of Programming WSS Stations

ASP 2020

Contents

Setting

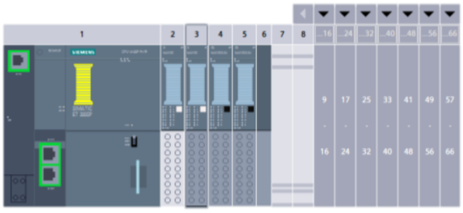
PLC

Steps to setting up PLC

Choose Model Number of PLC

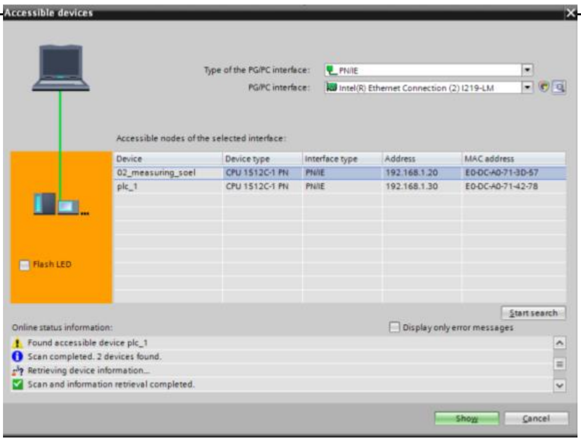
In the project tree

"



Scan PLC

"



Type of the PG/PC interface: PN/IE

PG/PC interface: Intel(R) Ethernet Connection (2) I219-LM

Accessible nodes of the selected interface:

Device	Device type	Interface type	Address	MAC address
02_measuring_soel	CPU 1512C-1 PN	PN/IE	192.168.1.20	E0-DC-A0-71-3D-57
plc_1	CPU 1512C-1 PN	PN/IE	192.168.1.30	E0-DC-A0-71-42-78

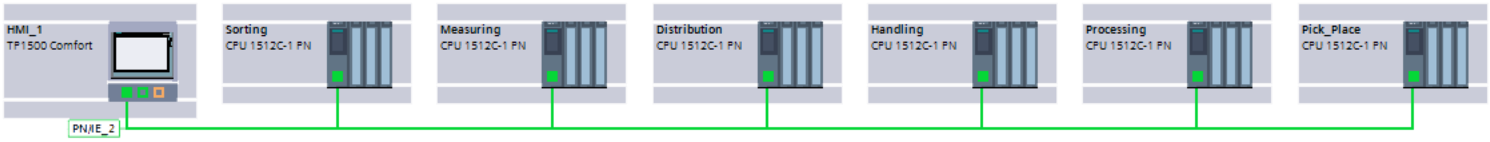
Online status information:

□ Display only error messages

```
Found accessible device plc_1
Scan completed. 2 devices found.
Retrieving device information...
Scan and information retrieval completed.
```

Connect all PLC to the same Subnet

"

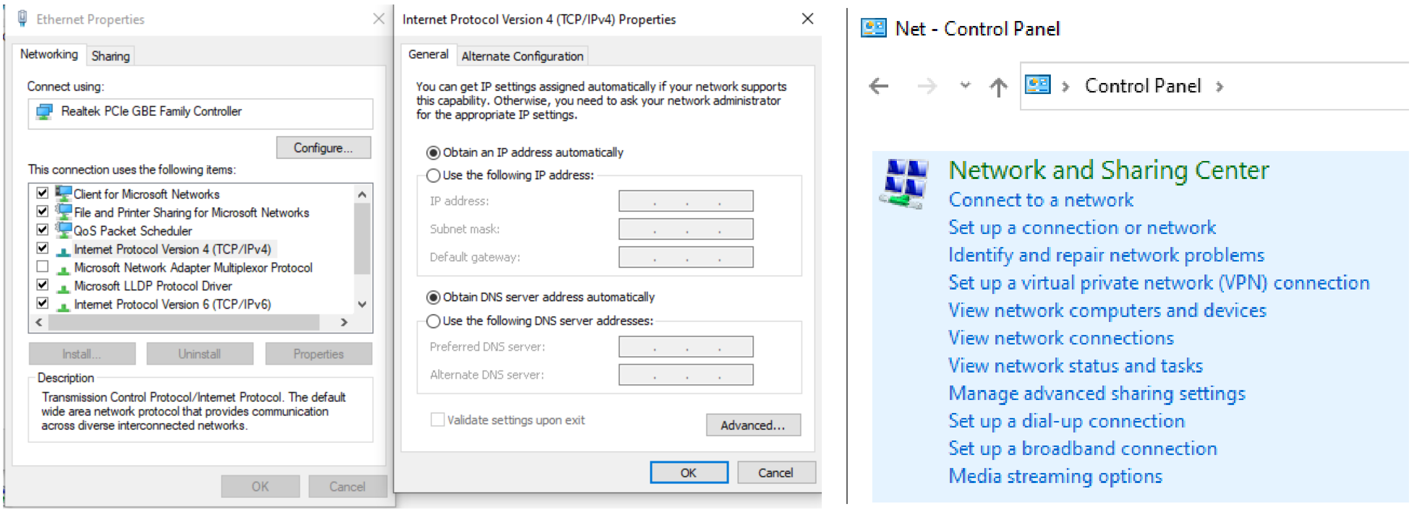


✓Drag Connections between Stations

✓Use Windows Network Manager to change IPV4

✓Windows Network Manager to change IPV4

"



Ethernet Properties

Networking | Sharing

Connect using:

Realtek PCIe GBE Family Controller

Configure...

This connection uses the following items:

- ☑ Client for Microsoft Networks
- ☑ File and Printer Sharing for Microsoft Networks
- ☑ QoS Packet Scheduler
- ☑ Internet Protocol Version 4 (TCP/IPv4)
- □ Microsoft Network Adapter Multiplexor Protocol
- ☑ Microsoft LLDP Protocol Driver
- ☑ Internet Protocol Version 6 (TCP/IPv6)

Install... | Uninstall | Properties

Description

Transmission Control Protocol/Internet Protocol. The default wide area network protocol that provides communication across diverse interconnected networks.

OK | Cancel

Internet Protocol Version 4 (TCP/IPv4) Properties

General | Alternate Configuration

You can get IP settings assigned automatically if your network supports this capability. Otherwise, you need to ask your network administrator for the appropriate IP settings.

● Obtain an IP address automatically

○ Use the following IP address:

IP address: . . .

Subnet mask: . . .

Default gateway: . . .

● Obtain DNS server address automatically

○ Use the following DNS server addresses:

Preferred DNS server: . . .

Alternate DNS server: . . .

□ Validate settings upon exit

Advanced...

OK | Cancel

Net - Control Panel

Control Panel ›

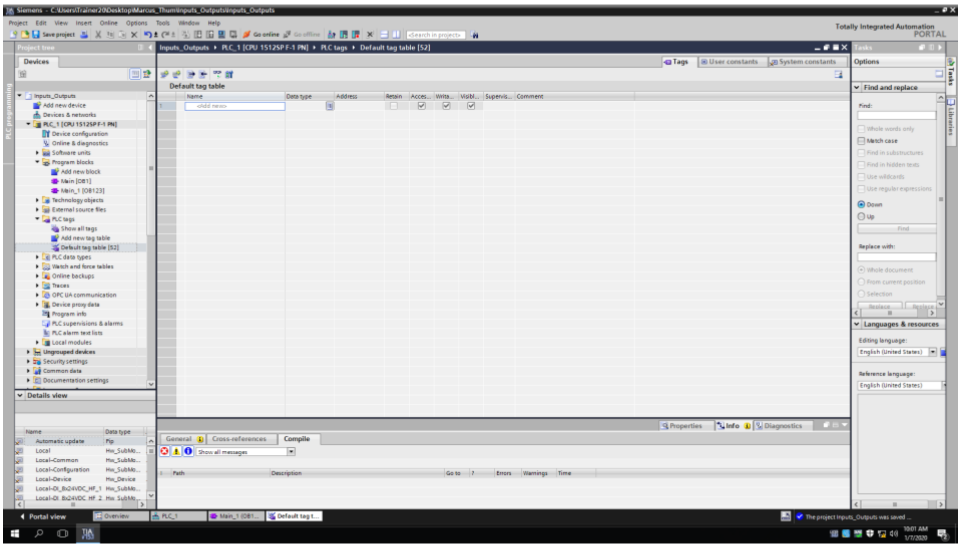
Network and Sharing Center

- Connect to a network
- Set up a connection or network
- Identify and repair network problems
- Set up a virtual private network (VPN) connection
- View network computers and devices
- View network connections
- View network status and tasks
- Manage advanced sharing settings
- Set up a dial-up connection
- Set up a broadband connection
- Media streaming options

Tags

Tag Table

Can be used to reference various types of variables.



The project tree lists the default tag table. It is a subset of the station. Additional tables can be added. These are the type of variables I use.

1

Boolean

This is used to define variables that are True/False

2

Int

Integer variables are used to store numerical values

1

%I

Inputs

2

%Q

Outputs

3

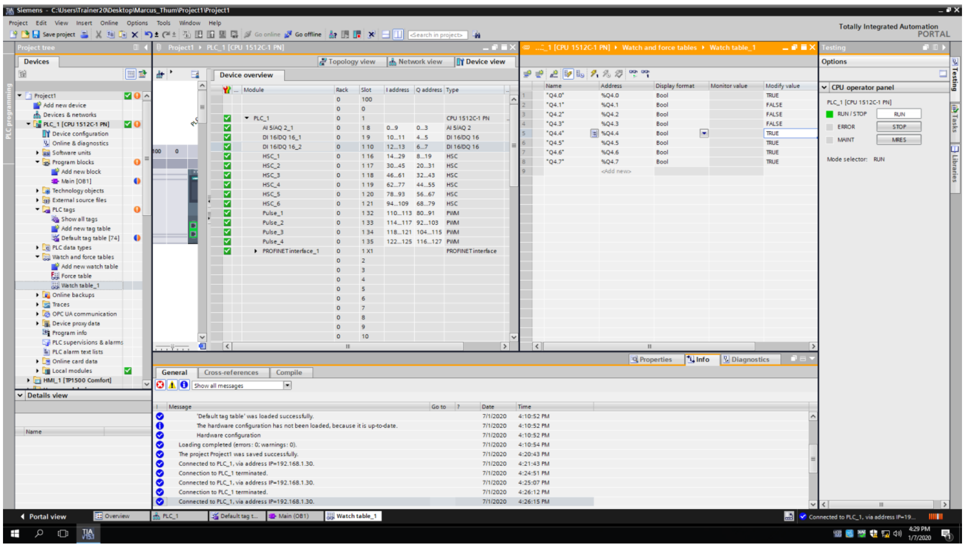
%M

Memory

Watch Table

Monitor Variables

Can be used to monitor any variables



...1 [CPU 1512C-1 PN] ▸ Watch and force tables ▸ Watch table_1

	Name	Address	Display format	Monitor value	Modify value
1	"Q4.0"	%Q4.0	Bool		TRUE
2	"Q4.1"	%Q4.1	Bool		FALSE
3	"Q4.2"	%Q4.2	Bool		FALSE
4	"Q4.3"	%Q4.3	Bool		FALSE
5	"Q4.4"	%Q4.4	Bool		TRUE
6	"Q4.5"	%Q4.5	Bool		TRUE
7	"Q4.6"	%Q4.6	Bool		TRUE
8	"Q4.7"	%Q4.7	Bool		TRUE
9		<Add new>			

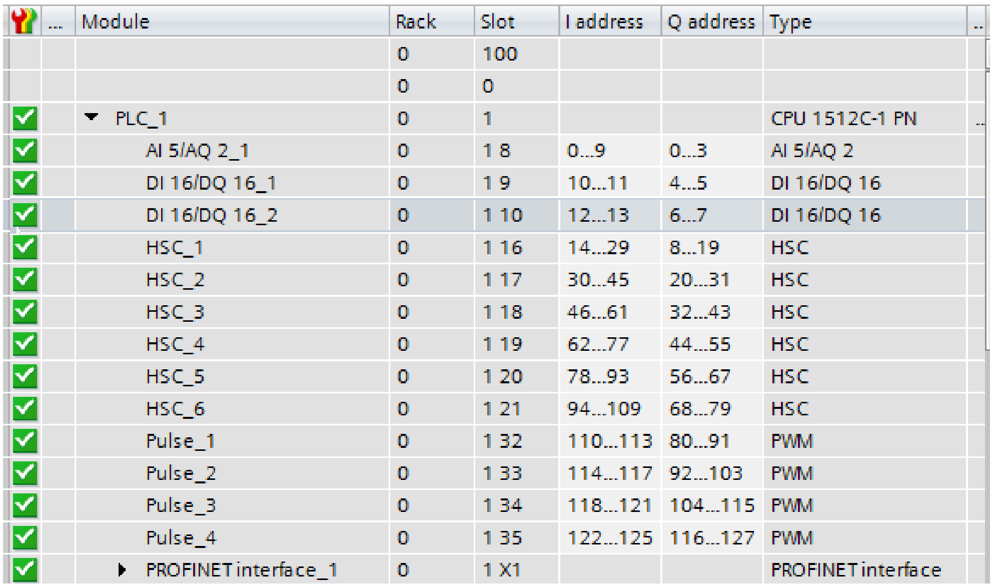
Message	Go to	?	Date	Time
'Default tag table' was loaded successfully.			7/1/2020	4:10:52 PM
The hardware configuration has not been loaded, because it is up-to-date.			7/1/2020	4:10:52 PM
Hardware configuration			7/1/2020	4:10:52 PM
Loading completed (errors: 0; warnings: 0).			7/1/2020	4:10:54 PM
The project Project1 was saved successfully.			7/1/2020	4:20:43 PM
Connected to PLC_1, via address IP=192.168.1.30.			7/1/2020	4:21:43 PM
Connection to PLC_1 terminated.			7/1/2020	4:24:51 PM
Connected to PLC_1, via address IP=192.168.1.30.			7/1/2020	4:25:07 PM
Connection to PLC_1 terminated.			7/1/2020	4:26:12 PM
Connected to PLC_1, via address IP=192.168.1.30.			7/1/2020	4:26:15 PM

Addresses

Input / Output

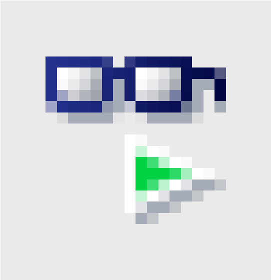
I10 to I11 is Input

Q4 to Q5 is output

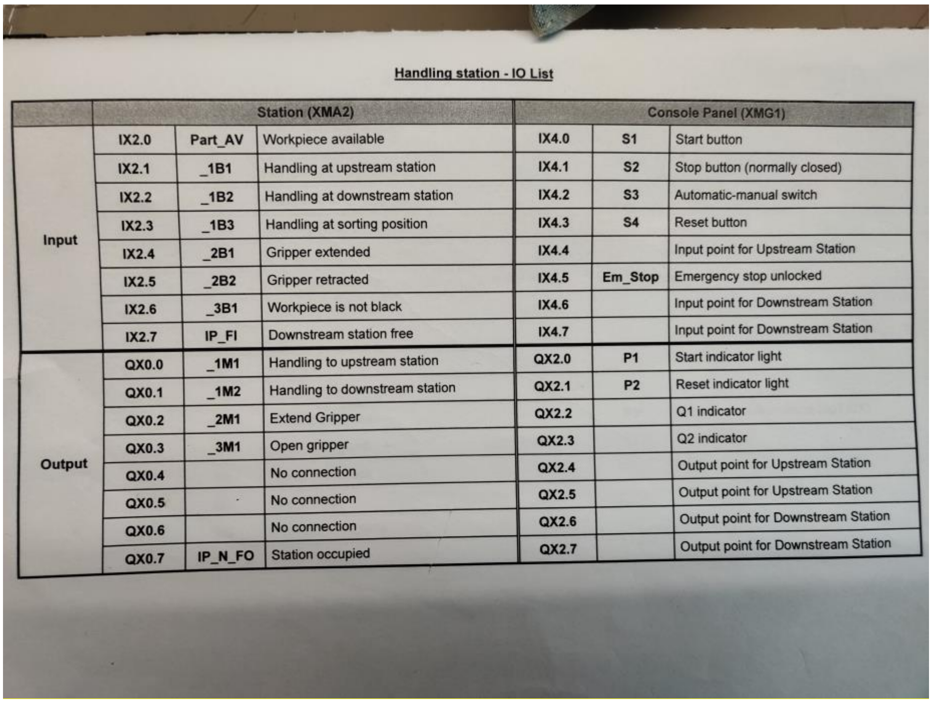


	...	Module	Rack	Slot	I address	Q address	Type
			0	100			
			0	0			
✓		▾ PLC_1	0	1			CPU 1512C-1 PN
✓		AI 5/AQ 2_1	0	1 8	0...9	0...3	AI 5/AQ 2
✓		DI 16/DQ 16_1	0	1 9	10...11	4...5	DI 16/DQ 16
✓		DI 16/DQ 16_2	0	1 10	12...13	6...7	DI 16/DQ 16
✓		HSC_1	0	1 16	14...29	8...19	HSC
✓		HSC_2	0	1 17	30...45	20...31	HSC
✓		HSC_3	0	1 18	46...61	32...43	HSC
✓		HSC_4	0	1 19	62...77	44...55	HSC
✓		HSC_5	0	1 20	78...93	56...67	HSC
✓		HSC_6	0	1 21	94...109	68...79	HSC
✓		Pulse_1	0	1 32	110...113	80...91	PWM
✓		Pulse_2	0	1 33	114...117	92...103	PWM
✓		Pulse_3	0	1 34	118...121	104...115	PWM
✓		Pulse_4	0	1 35	122...125	116...127	PWM
✓		▸ PROFINETinterface_1	0	1 X1			PROFINETinterface

Start Monitoring



Reference



Handling station - IO List

	Station (XMA2)			Console Panel (XMG1)		
Input	IX2.0	Part_AV	Workpiece available	IX4.0	S1	Start button
	IX2.1	_1B1	Handling at upstream station	IX4.1	S2	Stop button (normally closed)
	IX2.2	_1B2	Handling at downstream station	IX4.2	S3	Automatic-manual switch
	IX2.3	_1B3	Handling at sorting position	IX4.3	S4	Reset button
	IX2.4	_2B1	Gripper extended	IX4.4		Input point for Upstream Station
	IX2.5	_2B2	Gripper retracted	IX4.5	Em_Stop	Emergency stop unlocked
	IX2.6	_3B1	Workpiece is not black	IX4.6		Input point for Downstream Station
	IX2.7	IP_FI	Downstream station free	IX4.7		Input point for Downstream Station
Output	QX0.0	_1M1	Handling to upstream station	QX2.0	P1	Start indicator light
	QX0.1	_1M2	Handling to downstream station	QX2.1	P2	Reset indicator light
	QX0.2	_2M1	Extend Gripper	QX2.2		Q1 indicator
	QX0.3	_3M1	Open gripper	QX2.3		Q2 indicator
	QX0.4		No connection	QX2.4		Output point for Upstream Station
	QX0.5		No connection	QX2.5		Output point for Upstream Station
	QX0.6		No connection	QX2.6		Output point for Downstream Station
	QX0.7	IP_N_FO	Station occupied	QX2.7		Output point for Downstream Station

This is the Input / Output Table for the Handling Station.

WARNING: The addresses are different for Siemens.

Label

Use this as reference to label and name your variables

Identify

Use this to identify the order of Inputs / Outputs

Code

All the outputs & Inputs are Boolean.

They can either be set TRUE / FALSE

TRUE

1 Set (Output) HIGH

2 Check Condition (Input) HIGH

Examples: Turn on / Extend / Sensor HIGH

FALSE

1 Set (Output) LOW

2 Check Condition (Input) LOW

Examples: Turn off / Retract / Sensor LOW

Code

Most common Loops I use

IF Loop

```
IF <Boolean Expression>
THEN
        <Condition>
ELSE
        <Condition>
END_IF
```

Usually used with Boolean Expressions

This is used to program a condition.

```
CASE <INT> OF
        0:
           <code>
        1:
           <code>
END_CASE
```

CASE Statement

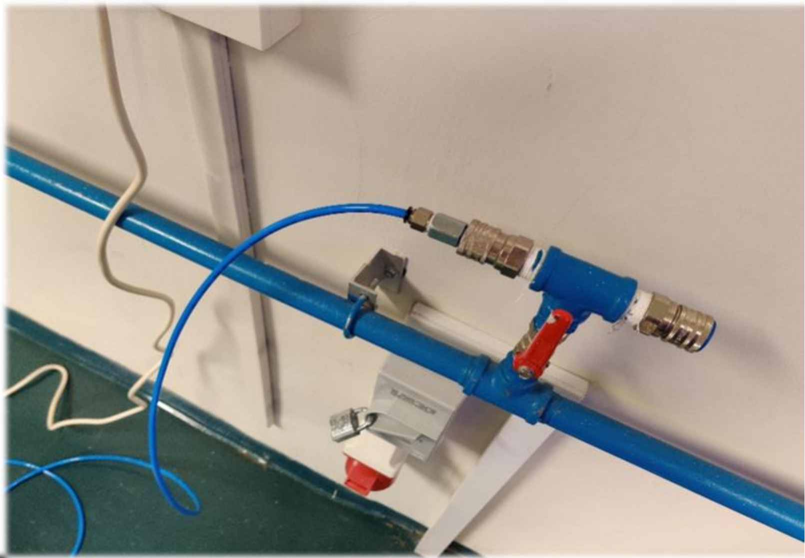
Excellent for sequencing. Code is modular based on STEPS.

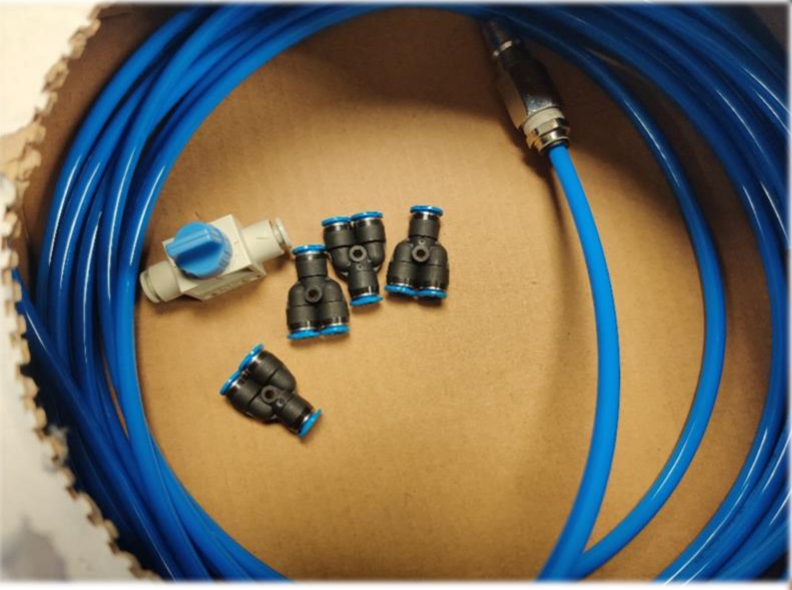
STEPS can be looped back.

Setup

Pneumatics Supply

Hook Up to Pneumatics Supply



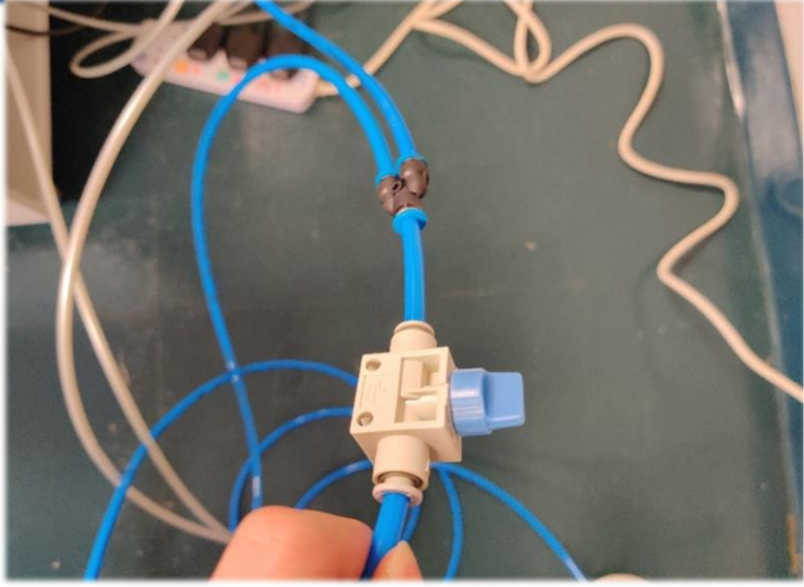


Pneumatics

Supply Valves and Splitters

Install

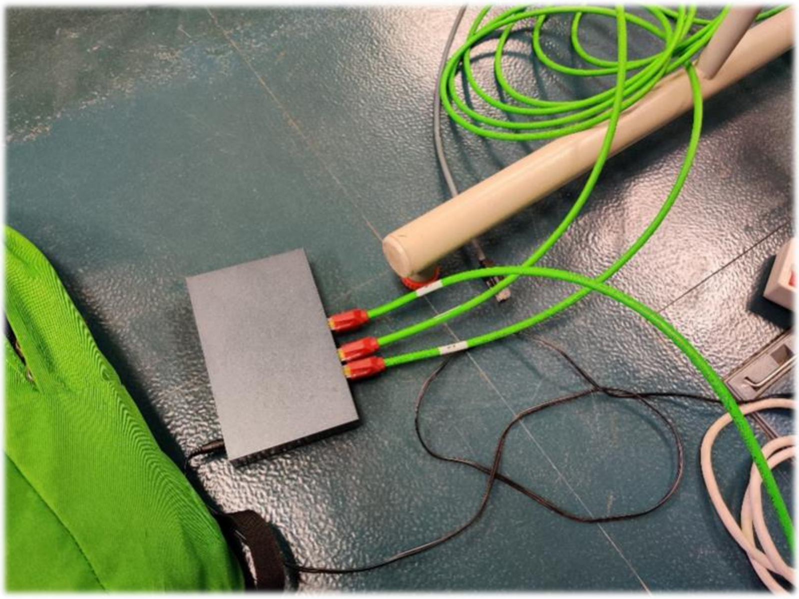
Cut to size and Attach



Setup

Power

Connect Stations to a Power Brick

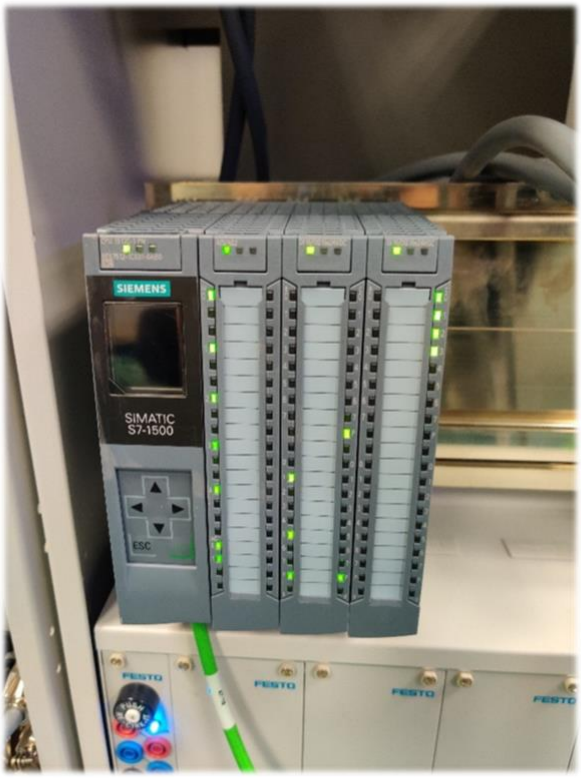


LAN

Connect to Network Switch

LAN

Connect to Station's PLC



Setup

Sample Project File

Let us start FIRST by using my project file

https://github.com/MarcusThum/Siemens-PLC-Programming

3 Stations

“Combined_3_Stations_FINAL”

Distribution > Measuring > Sorting

6 Stations

“Combined_6_Stations_200817_LATEST_Evening”

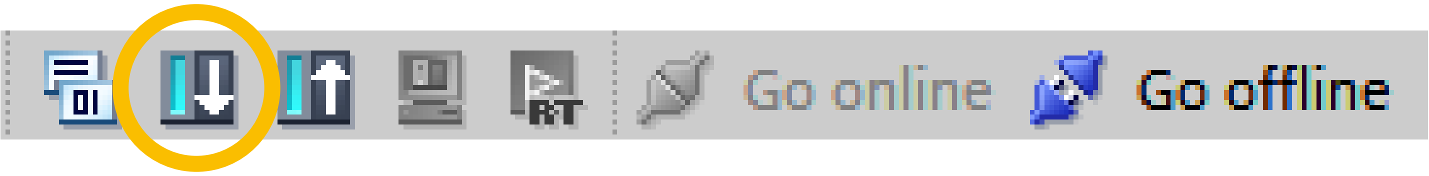
Distribution > Measuring > Handling > Processing > Pick Place > Sorting

Requirements

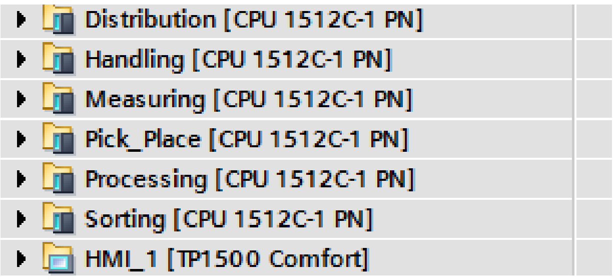
TIA PORTAL with License Key

(Inform your lecturer if your license key is invalid / expiring)

Setup

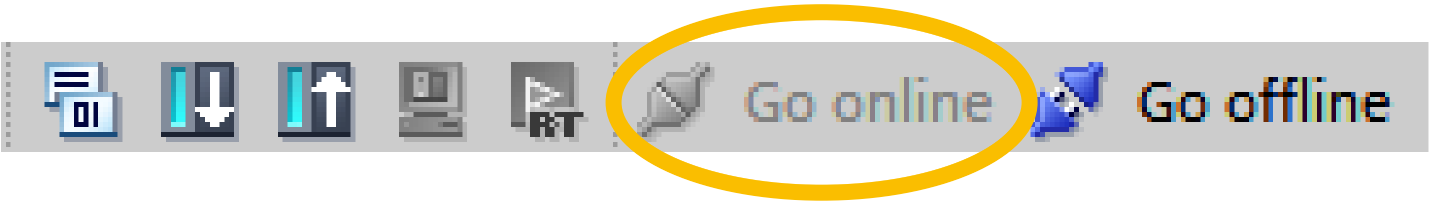


Download Code



- Distribution [CPU 1512C-1 PN]
- Handling [CPU 1512C-1 PN]
- Measuring [CPU 1512C-1 PN]
- Pick_Place [CPU 1512C-1 PN]
- Processing [CPU 1512C-1 PN]
- Sorting [CPU 1512C-1 PN]
- HMI_1 [TP1500 Comfort]

For each Station and HMI



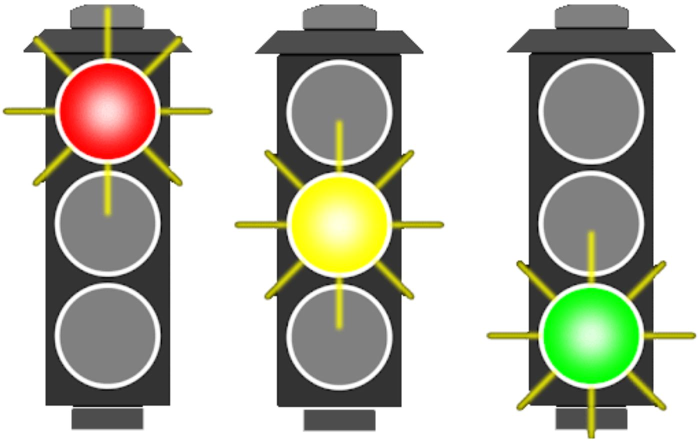
Go Online to Check

✓

If successful Sync / Download

Setup

Busy Signal



Busy Signals are intentionally used to inform the upstream station that the downstream station is busy and cannot services additional workpieces at the moment.

Wire

It can be done using wires connected on the control panel.

The outputs and inputs need to be assigned to a variable.

The outputs can be set HIGH or LOW.

Finally, an IF ELSE Statement can be used for the inputs to check for availability.

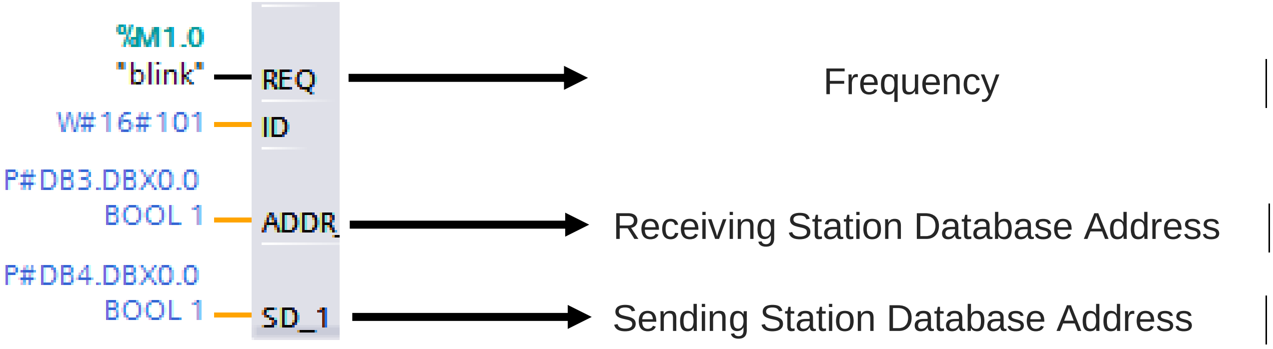
Put Block

A PUT Block can be added in Siemens that provides similar functionality.

It sends any variables you assign in the block and sends over whenever REQ is HIGH.

Thus, it needs a frequency to send data. I used my blink function as the frequency.

Databases is also necessary to send and receive data.



```
%M1.0
"blink"      — REQ
W#16#101     — ID
P#DB3.DBX0.0
BOOL 1       — ADDR
P#DB4.DBX0.0
BOOL 1       — SD_1
```

Frequency

Receiving Station Database Address

Sending Station Database Address

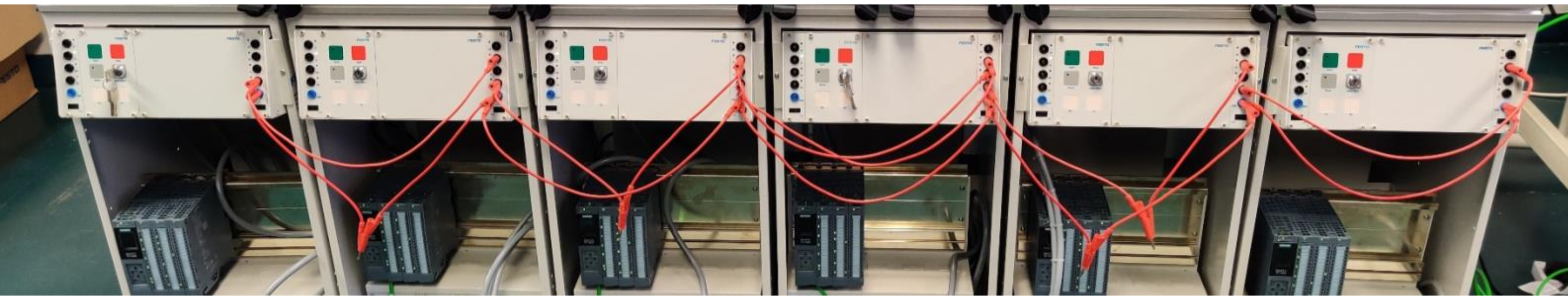
Setup

Busy Signal

6 Stations

“Combined_6_Stations_200817_LATEST_Evening”

Distribution > Measuring > Handling > Processing > Pick Place > Sorting



Wire Up Busy Signal (All Stations)

Downstream Q7

Upstream I7

There is a second signal

Handling to Processing Second Feedback

Handling Station Q7

Processing Station I6

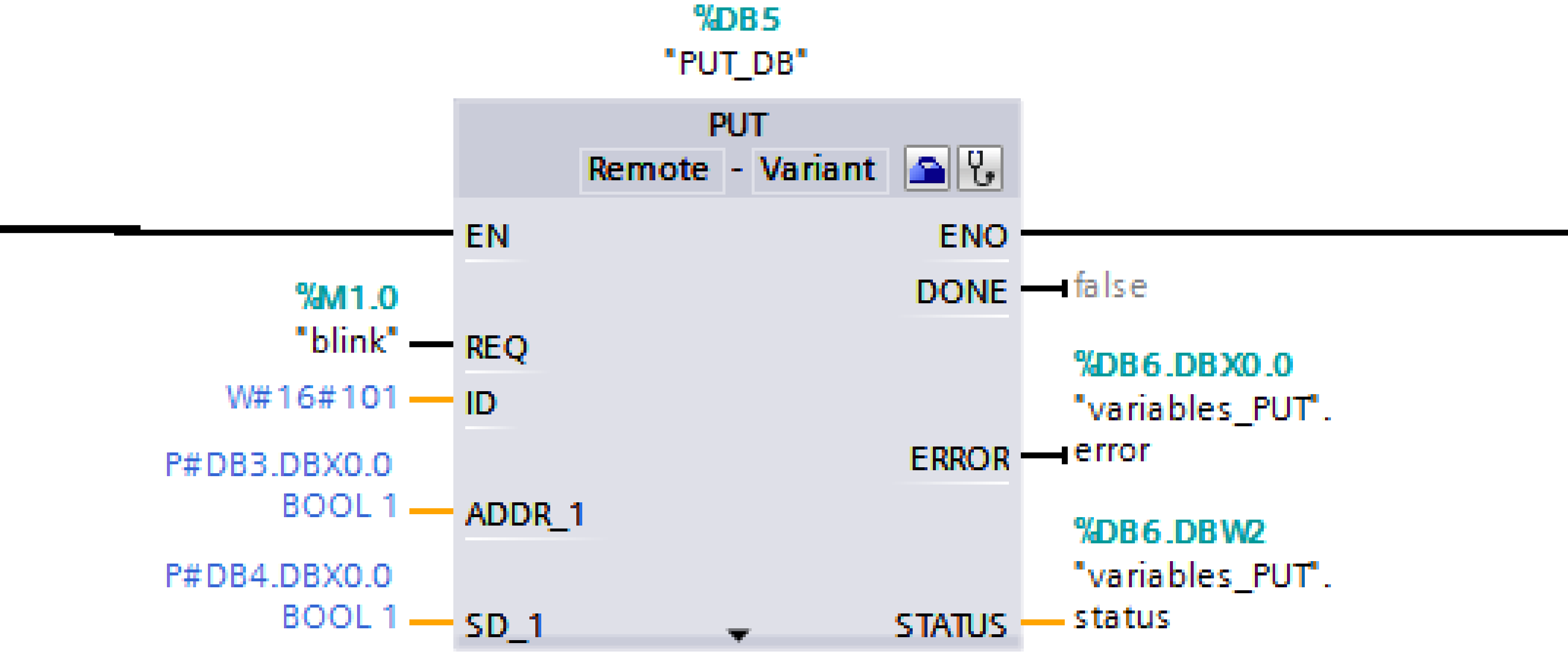
Setup

Busy Signal

3 Stations

“Combined_3_Stations_FINAL”

Distribution > Measuring > Sorting



%DB5
"PUT_DB"

PUT
Remote - Variant

EN
ENO

DONE — false

%M1.0
"blink" — REQ

W#16#101 — ID

%DB6.DBX0.0
"variables_PUT".error — ERROR

P#DB3.DBX0.0
BOOL 1 — ADDR_1

%DB6.DBW2
"variables_PUT".status — STATUS

P#DB4.DBX0.0
BOOL 1 — SD_1

Put Block

Upstream to Downstream Station

Code Explanation

CODE SNIPPET

```
"blinker".TON(IN := TRUE,
          PT := T#1s);
IF "blinker".Q THEN
   "blinker".TON(IN := FALSE,
               PT := T#0s);
   IF "blink" THEN
      "blink" := FALSE;
   ELSE
      "blink" := TRUE;
   END_IF;
END_IF;


IF "resetOn" OR ("resetBlink" AND "blink") THEN
   "resetButtonLight" := 1;
ELSE
   "resetButtonLight" := 0;
END_IF;

IF "startOn" OR ("startBlink" AND "blink") THEN
   "startButtonLight" := 1;
ELSE
   "startButtonLight" := 0;
END_IF;
```

BLINKER

1. Blink LED's of the Reset and Start Button

2. Blink Indicators of the HMI

3. Used as 1 second frequency to trigger PUT BLOCK

IMAGE

Code Explanation

CODE SNIPPET

```
"Timer".TON(IN := TRUE,
                PT := T#1s);
      IF "Timer".Q THEN
         "Timer".TON(IN := FALSE,
                   PT := T#0s);
         <condition / result>
      END_IF;
```

Timers

Timers are often used to create delays between each step since cylinders and arms do not move instantaneously. It is to prevent jamming.

Delays are also useful for sensors. The sensors need to be TRUE for a SET DELAY PERIOD to trigger the Timer. This is useful when detecting workpieces and position of arms and cylinders.

TIMER DELAY

1 Use TON to create a 1 Second Delay

2 Move to next step when (timer.Q) timer is up

Code Explanation

STOP BUTTON

1 Check For Stop Button Press (Physical & HMI)

2 Reset all Outputs

3 Reset all variables

4 Blink Reset Button Light

5 Check For Reset Button Press (Physical & HMI)

Stop Button Presses

This is necessary to stop the stations in the event it jams or the user decides that he/she wants to stop production.

CODE SNIPPET

```
IF NOT "stopButton" OR "hmiStopButton" THEN
  "drillMotorOn" := 0;
  "turnTable" := 0;
  "lowerDrillingUnit" := 0;
  "raiseDrillingUnit" := 0;
  "clampWorkpiece" := 0;
  "checkHole" := 0;
  "pushOutWorkpiece" := 0;
  "step" := 0;
  "turning" := 0;
  "checking" := 0;
  "drilling" := 0;
END_IF;
CASE "step" OF
  0:
    "holePresentForDrilling" := FALSE;
    "doneDrilling" := FALSE;
    "holePresent" := FALSE;
    "feedback" := FALSE;
    "resetCount" := 0;
    "Timer".TON(IN := FALSE,
          PT := T#0s);
    "resetOn" := FALSE;
    "resetBlink" := TRUE;
    "startOn" := FALSE;
    "startBlink" := FALSE;
    IF "resetButton" AND "autoManualSwitch" THEN
      "step" := 1;
    END_IF;
    IF "hmiResetButton" AND "hmiAutoManualSwitch" THEN
      "step" := 1;
    END_IF;
    IF "hmiMasterReset" THEN
      "step" := 1;
    END_IF;
```

Code Explanation

RESET SEQUENCE

1 Reset Button Light On

2 Reset Sequence Begin

CODE SNIPPET

```
1:
    "resetOn" := TRUE;
    "resetBlink" := FALSE;
    "raiseDrillingUnit" := 1;
    "lowerDrillingUnit" := 0;
    "Timer".TON(IN := "drillInUpper",
                PT := T#1s);
    IF "Timer".Q THEN
      "Timer".TON(IN := FALSE,
                  PT := T#0s);
      "step" := 10;
    END_IF;
```

Reset Sequence

It is necessary to perform a reset sequence before you start the machine. This is done to clear any remaining workpieces left after stopping.

Difficulty

Some station’s reset sequence tends to be harder. Handling Station’s Reset Sequence uses part of it’s start sequence to clear any workpiece available. This shows the reusability of code among sequences.

Code Explanation

CODE SNIPPET

```
70:
    "resetOn" := FALSE;
    "resetBlink" := FALSE;
    "startOn" := FALSE;
    "startBlink" := TRUE;
    IF "startButton" AND NOT "autoManualSwitch" THEN
      "step" := 80;
    END_IF;
    IF "hmiStartButton" AND NOT "hmiAutoManualSwitch" THEN
      "step" := 80;
    END_IF;
    IF "hmiMasterStart" THEN
      "step" := 80;
    END_IF;

  80:
    "startOn" := 1;
    "startBlink" := 0;
    IF "workpieceAvailable" AND "turning" = 0 AND "checking" = 0 AND "drilling" = 0 THEN
      "step" := 91;
    END_IF;
    "Timer_1".TON(IN := "turning" = 0 AND "checking" = 0 AND "drilling" = 0,
           PT := T#5s);
    IF "Timer_1".Q THEN
      "Timer_1".TON(IN := FALSE,
             PT := T#0s);
      "step" := 90;
    END_IF;
```

START BUTTON

1. Check For Start Button Press (Physical & HMI)
2. Start Button Light On
3. Start Sequence Begin

Start Sequence

Start Sequence is where the stations performs its main automated sequence such as transferring a workpiece to the next downstream station.

Code Explanation

Unique “Case”

Processing Station

1 4 Case Statements

2 Can be done using separate programs on CoDeSys

3 5 second delay to wait for workpiece

Case Loops:

Steps

Turning

Checking

Drilling

Code Explanation

View Raw Code Siemens

My Website

www.marcusthum.com/sixstations

www.marcusthum.com/siemens

Video Demonstration

YouTube

https://www.youtube.com/channel/UCflRSA8qN4-dR7nhpM1nMXA/videos

Sample Project

OneDrive SharePoint

(Login to Temasek)

https://studenttpedu-my.sharepoint.com/:f:/g/personal/1801321i_student_tp_edu_sg/Ets0srNBOXVFk_5Dq3BnEIoBIAvT7NkE7Uymmm9DTqQNsQ?e=PRHUDq

View Raw Code CoDeSys

My Website

(Not meant to be part of this guide

BUT I started learning CoDeSys first)

You may too!

www.marcusthum.com/codesys

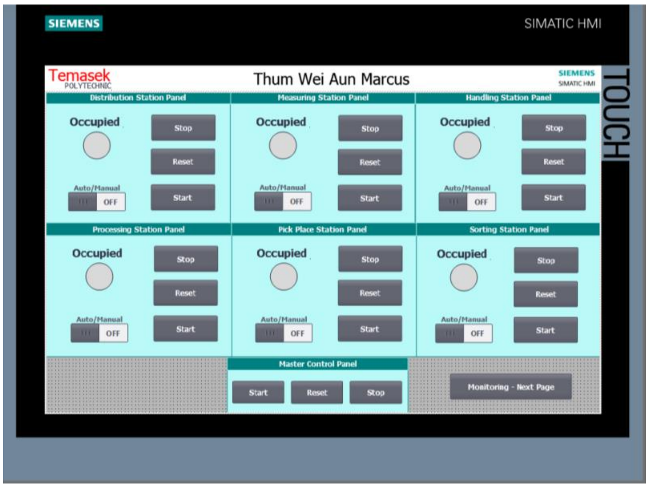
Need more help with CoDeSys?

Ask

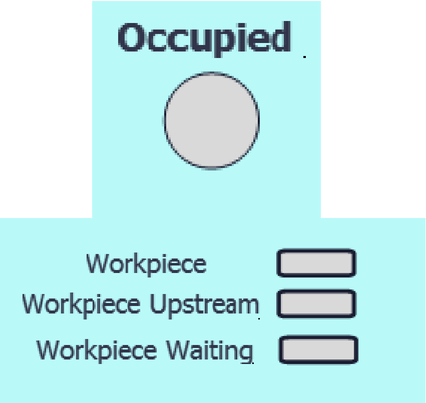
Suriya or Lennard

(Provided they are still in WSS)

HMI



Human Machine Interface is made for interactions between human and machines



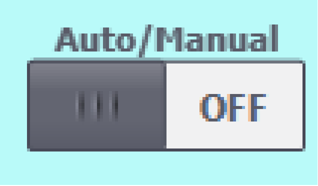
Indicators

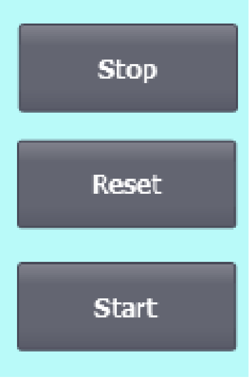
This is a substitute for physical LED’s. It can blink any colour or stay on for a period of time. Very Useful!

Switches

It emulates a physical switch. Either **ON** or **OFF**.

TRUE or **FALSE**. 2 State Configuration.

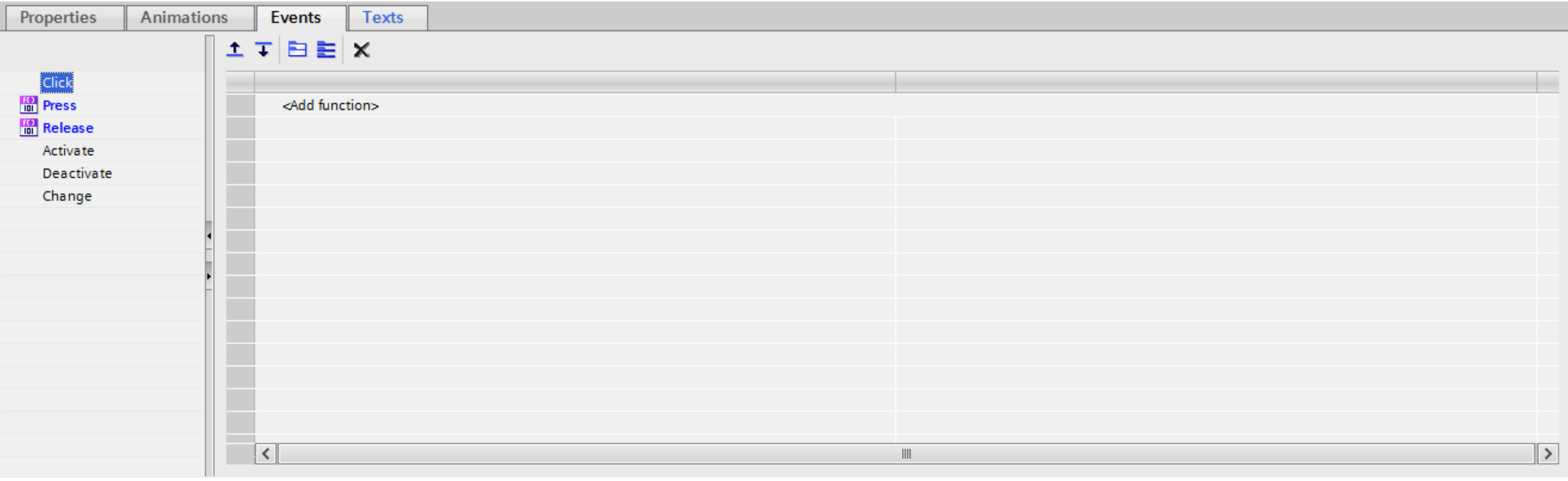




Buttons

Press / Hold / Let Go for a programmed condition to happen. You can **SET** or **RESET BITS** for variables.

HMI

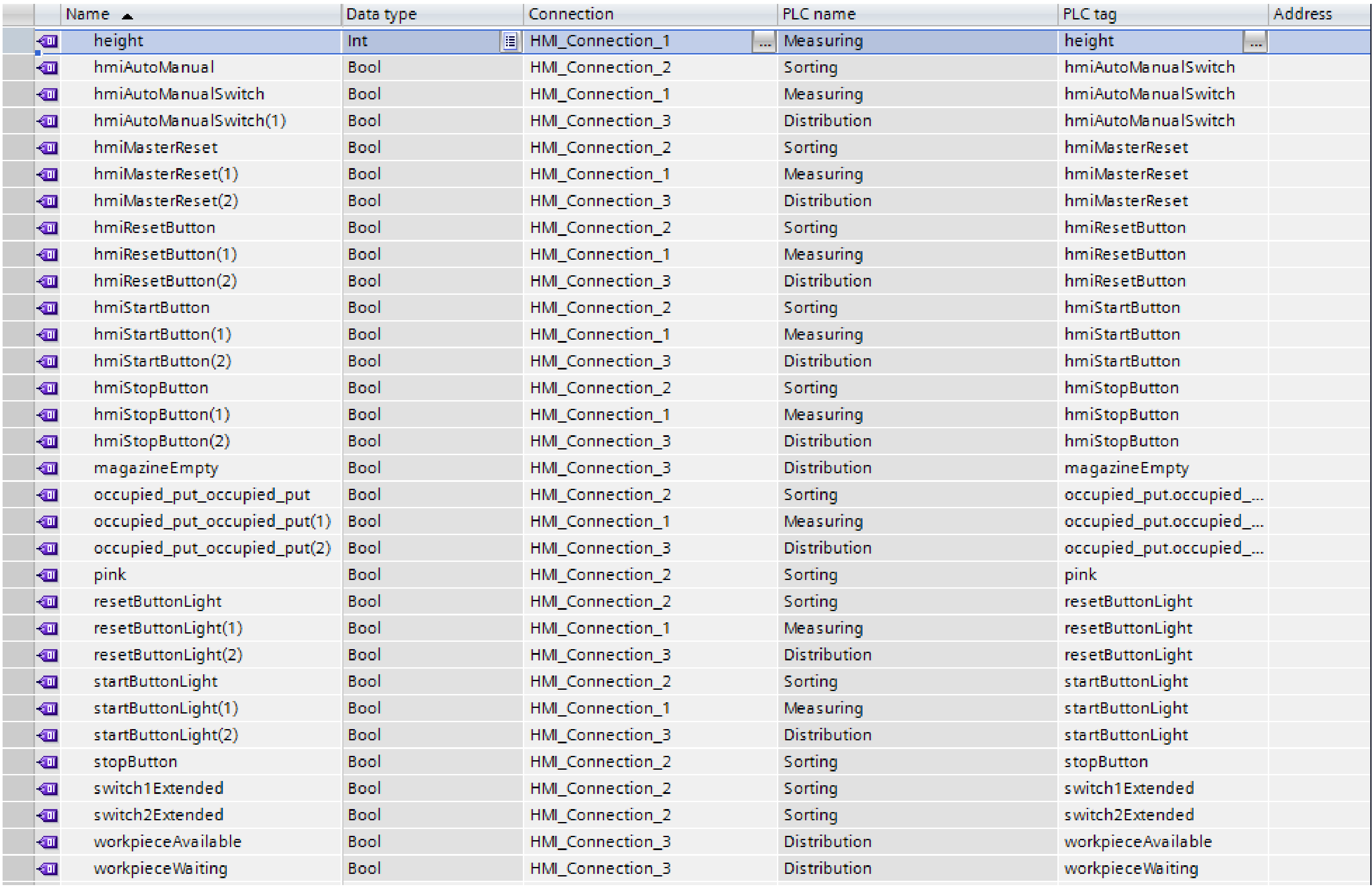


Events

Modify Button Events

Animation

Use To Function as Indicators

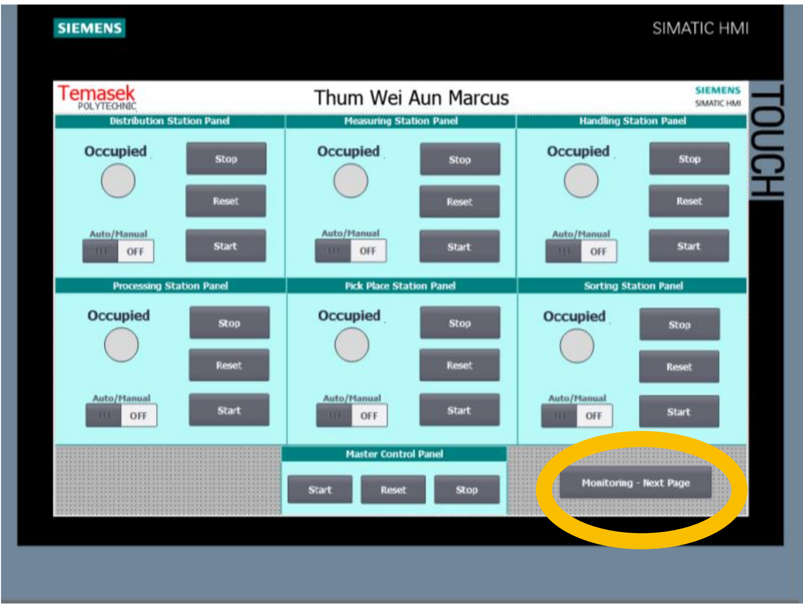


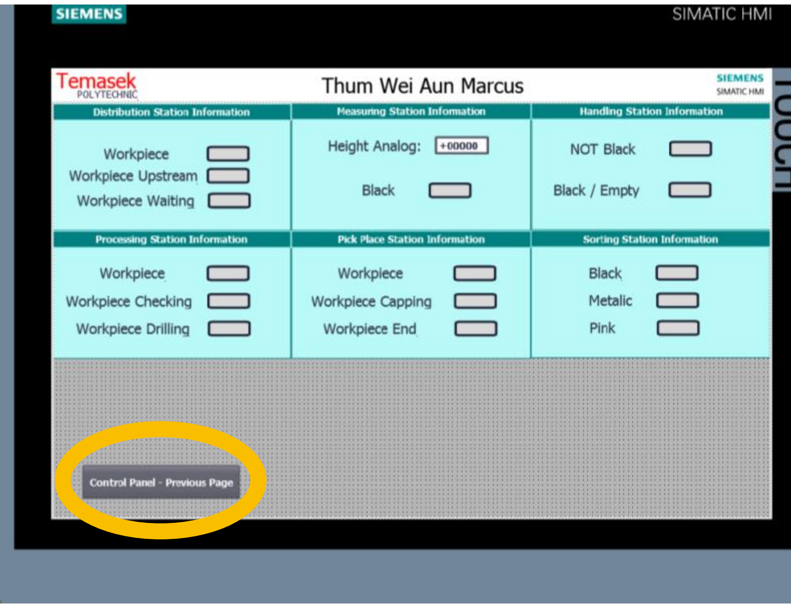
Name	Data type	Connection	PLC name	PLC tag	Address
height	Int	HMI_Connection_1	Measuring	height	
hmiAutoManual	Bool	HMI_Connection_2	Sorting	hmiAutoManualSwitch	
hmiAutoManualSwitch	Bool	HMI_Connection_1	Measuring	hmiAutoManualSwitch	
hmiAutoManualSwitch(1)	Bool	HMI_Connection_3	Distribution	hmiAutoManualSwitch	
hmiMasterReset	Bool	HMI_Connection_2	Sorting	hmiMasterReset	
hmiMasterReset(1)	Bool	HMI_Connection_1	Measuring	hmiMasterReset	
hmiMasterReset(2)	Bool	HMI_Connection_3	Distribution	hmiMasterReset	
hmiResetButton	Bool	HMI_Connection_2	Sorting	hmiResetButton	
hmiResetButton(1)	Bool	HMI_Connection_1	Measuring	hmiResetButton	
hmiResetButton(2)	Bool	HMI_Connection_3	Distribution	hmiResetButton	
hmiStartButton	Bool	HMI_Connection_2	Sorting	hmiStartButton	
hmiStartButton(1)	Bool	HMI_Connection_1	Measuring	hmiStartButton	
hmiStartButton(2)	Bool	HMI_Connection_3	Distribution	hmiStartButton	
hmiStopButton	Bool	HMI_Connection_2	Sorting	hmiStopButton	
hmiStopButton(1)	Bool	HMI_Connection_1	Measuring	hmiStopButton	
hmiStopButton(2)	Bool	HMI_Connection_3	Distribution	hmiStopButton	
magazineEmpty	Bool	HMI_Connection_3	Distribution	magazineEmpty	
occupied_put_occupied_put	Bool	HMI_Connection_2	Sorting	occupied_put.occupied_...	
occupied_put_occupied_put(1)	Bool	HMI_Connection_1	Measuring	occupied_put.occupied_...	
occupied_put_occupied_put(2)	Bool	HMI_Connection_3	Distribution	occupied_put.occupied_...	
pink	Bool	HMI_Connection_2	Sorting	pink	
resetButtonLight	Bool	HMI_Connection_2	Sorting	resetButtonLight	
resetButtonLight(1)	Bool	HMI_Connection_1	Measuring	resetButtonLight	
resetButtonLight(2)	Bool	HMI_Connection_3	Distribution	resetButtonLight	
startButtonLight	Bool	HMI_Connection_2	Sorting	startButtonLight	
startButtonLight(1)	Bool	HMI_Connection_1	Measuring	startButtonLight	
startButtonLight(2)	Bool	HMI_Connection_3	Distribution	startButtonLight	
stopButton	Bool	HMI_Connection_2	Sorting	stopButton	
switch1Extended	Bool	HMI_Connection_2	Sorting	switch1Extended	
switch2Extended	Bool	HMI_Connection_2	Sorting	switch2Extended	
workpieceAvailable	Bool	HMI_Connection_3	Distribution	workpieceAvailable	
workpieceWaiting	Bool	HMI_Connection_3	Distribution	workpieceWaiting	

HMI Tags

Create HMI Tags by linking to Station’s Tags

HMI





Create Multiple Screens

Switch Between Screens

Design

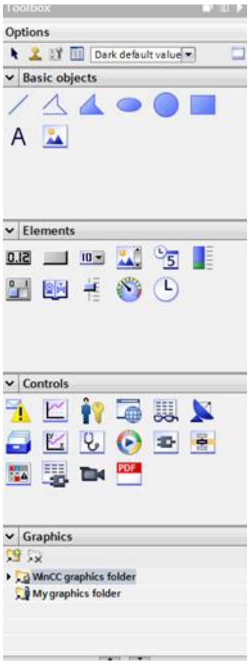
Buttons and indicators can be placed in different positions for aesthetically pleasing control panel

Functionality

The Siemens Comfort Panel is touch screen. It has a touch screen panel to detect button presses

User Friendly

It is easy to interact with



UI Panel

Contains Elements, Objects and Controls to insert into the HMI Screen. The Buttons and Shapes are customizable to suit each designer’s desired look and feel, as well as Functionality.

THE END