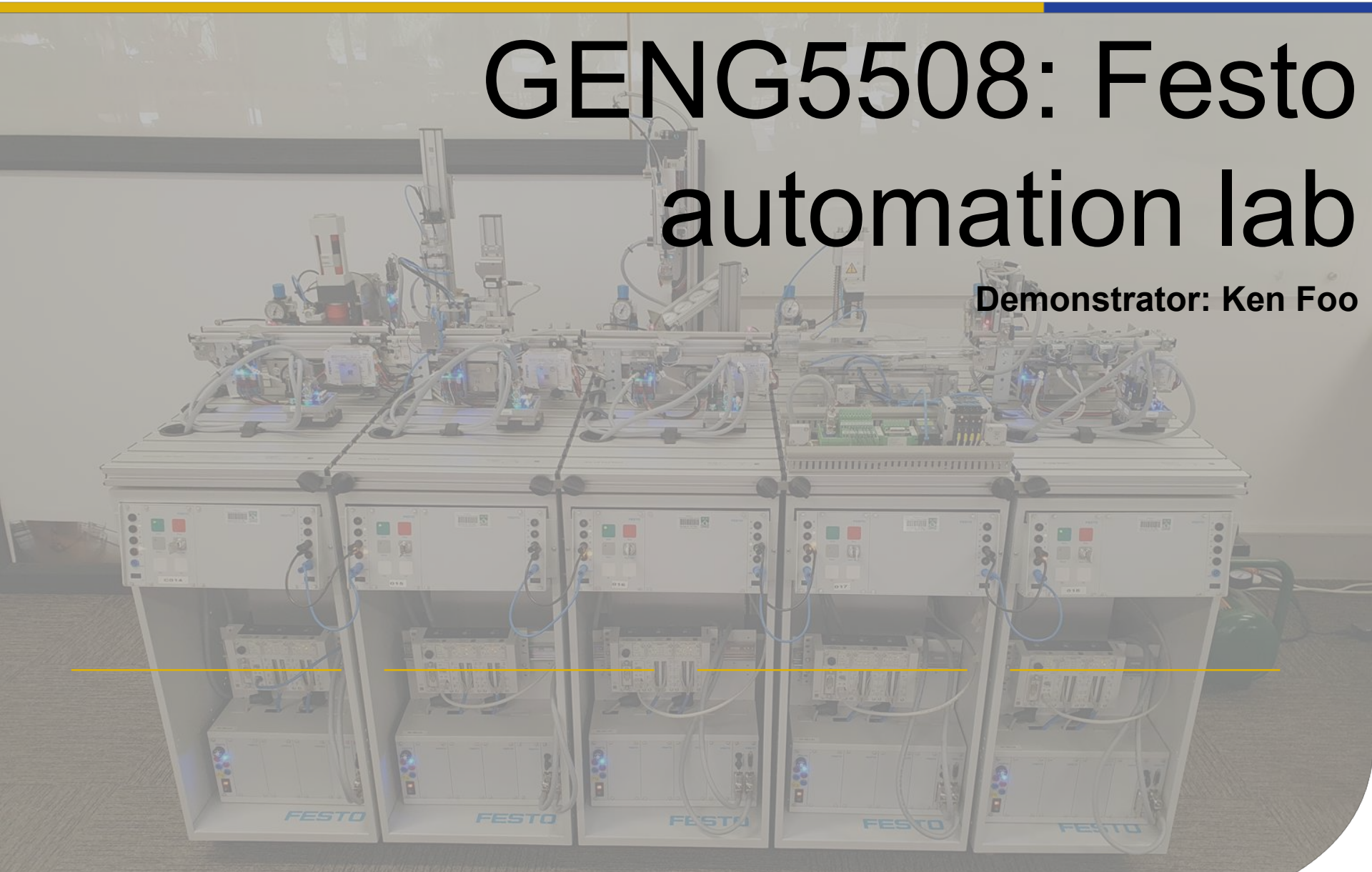
THE UNIVERSITY OF WESTERN AUSTRALIA

GENG5508: Festo automation lab



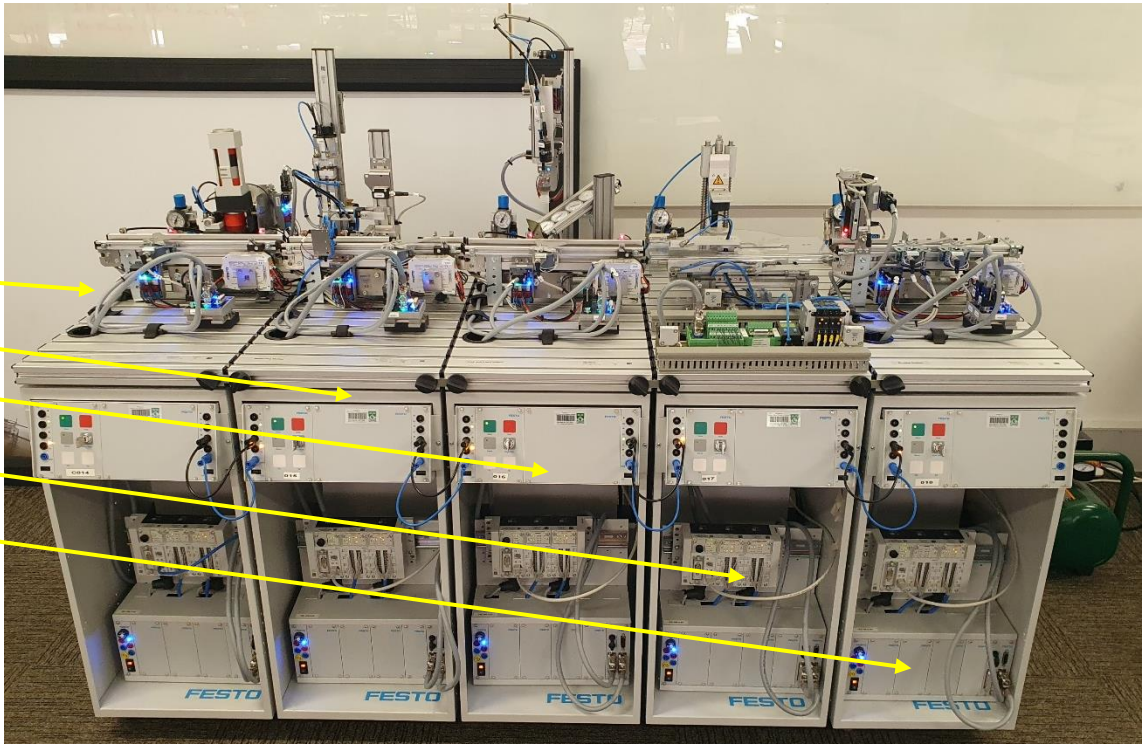
Demonstrator: Ken Foo

Contents

- Introduction to Festo
- Description of lab project
- Introduction to CODESYS
- Sequential Flow Chart (SFC) example
- Practical details and lab preparation

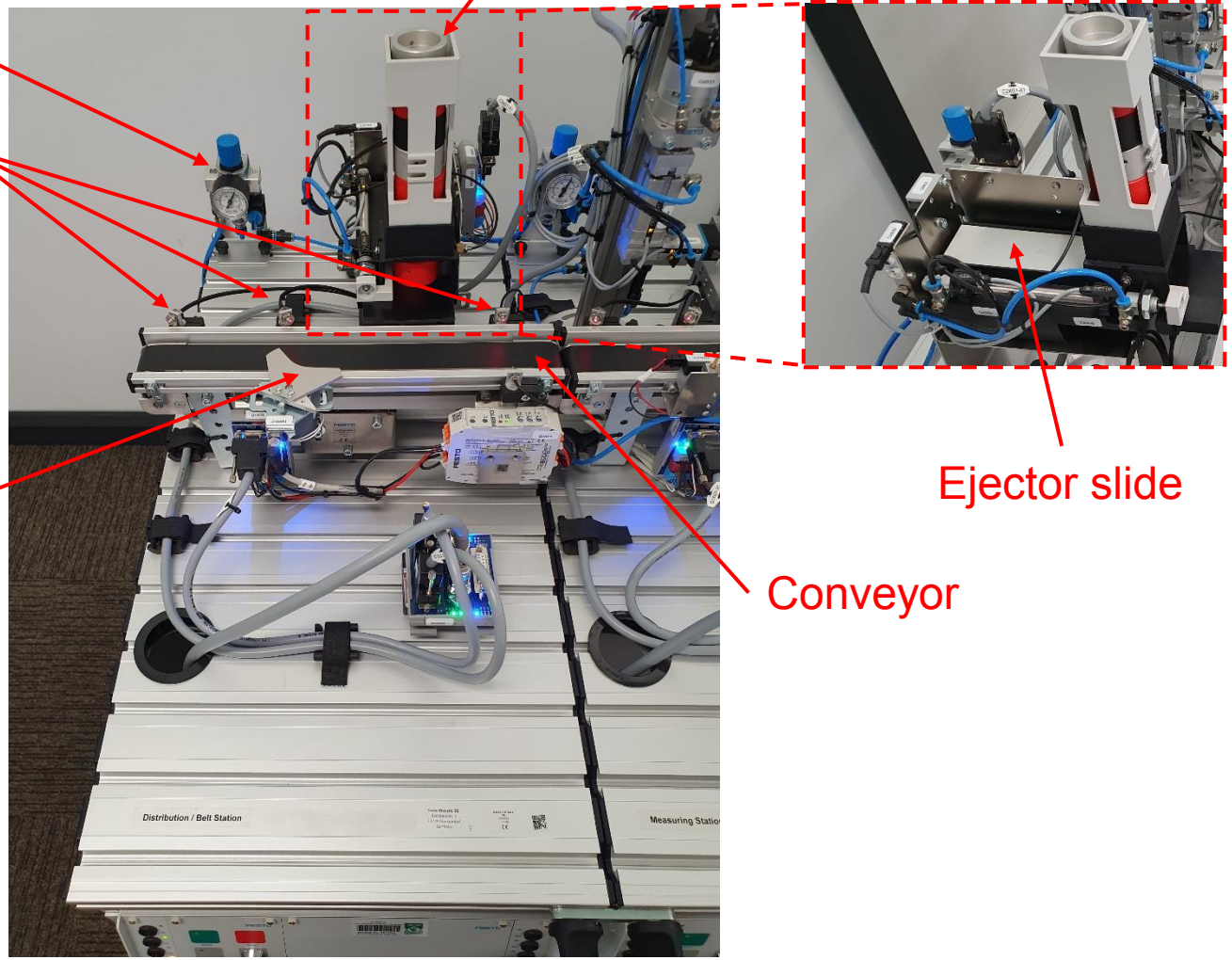
Festo introduction

- Festo is a German industrial control and automation company
  - Festo Didactic: provides educational and consultation services
- Festo Modular Production System (MPS): miniaturised production line used for educational purposes
- Simulates production of humidity (hygrometer) sensors
- Comprised of both electrical and pneumatic systems
- Composed of 5 stations:
  - Distribution station
  - Measuring station
  - Pick and place station
  - Pressing station
  - Sorting station



Festo: Distributing station

- Task: store and feed workpiece housings



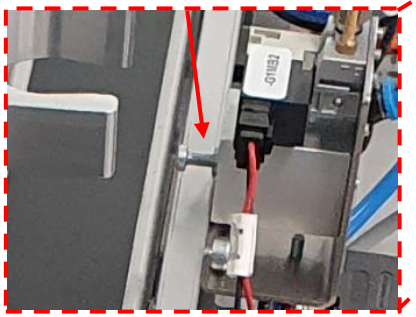
Festo: Measuring station

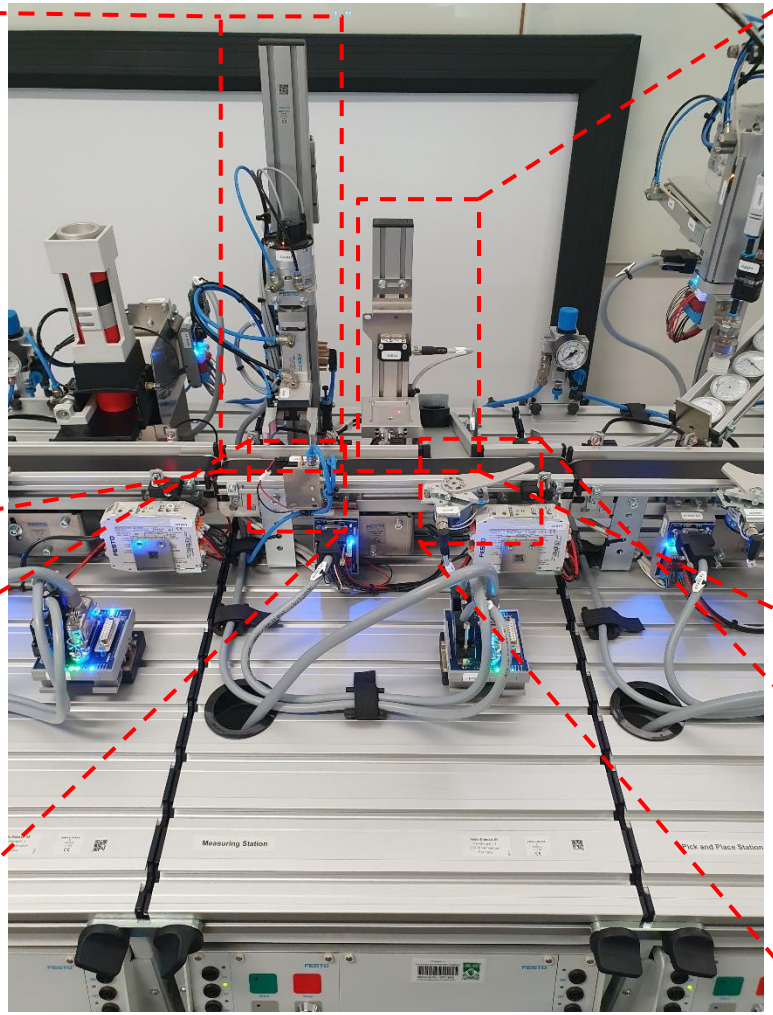
- Task: measure workpiece housing height

Lift and rotate module

Gripper

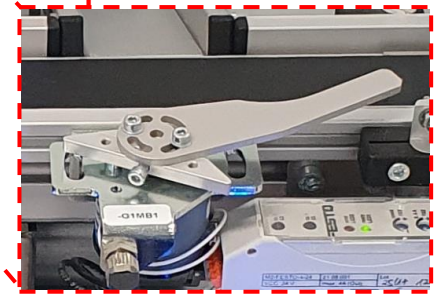
Stopper





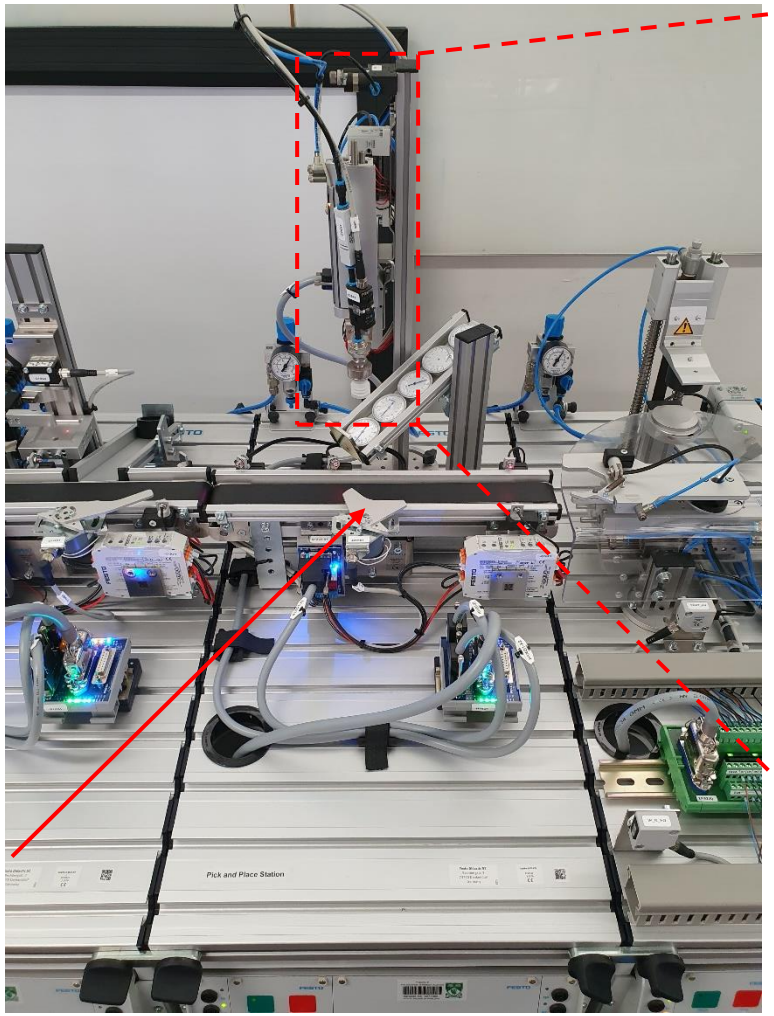
Measurement module

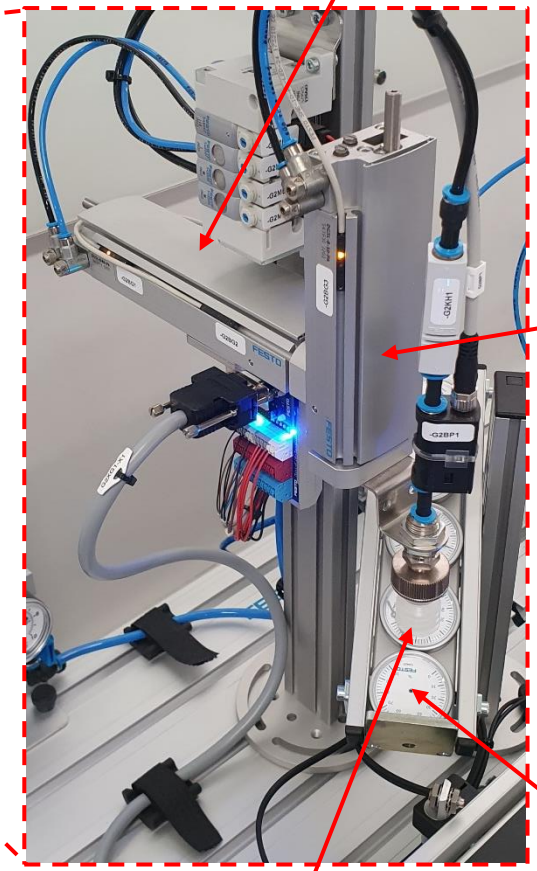
Separator



Festo: Pick and place station

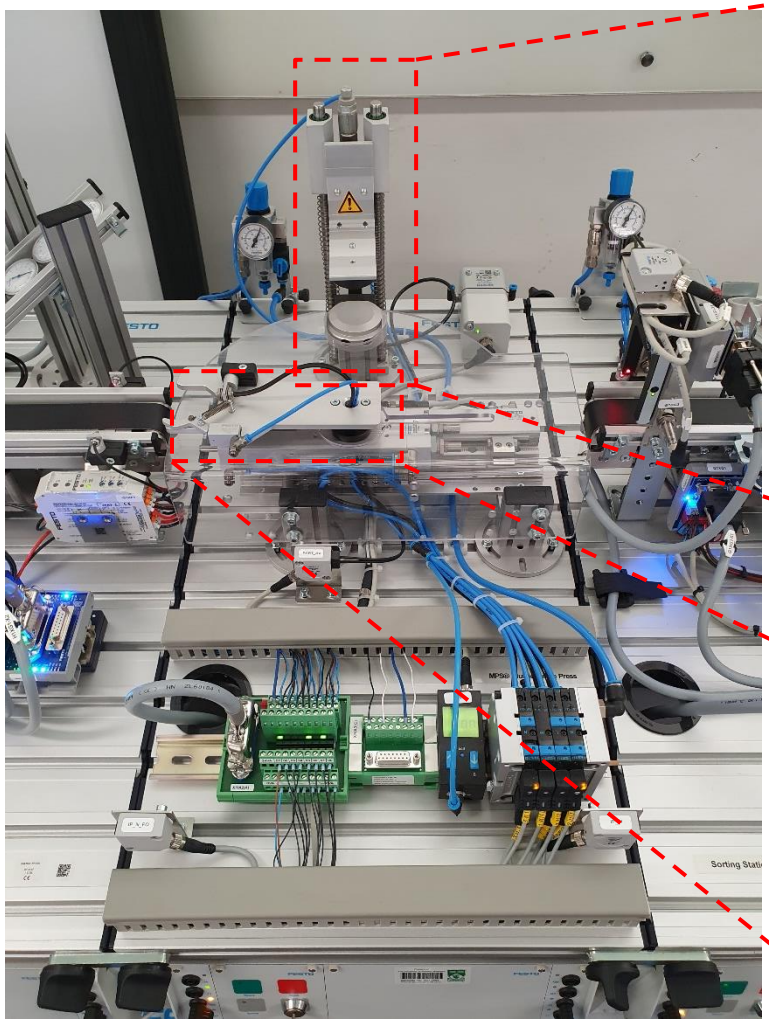
- Task: insert workpiece insert into workpiece housing





Festo: Pressing station

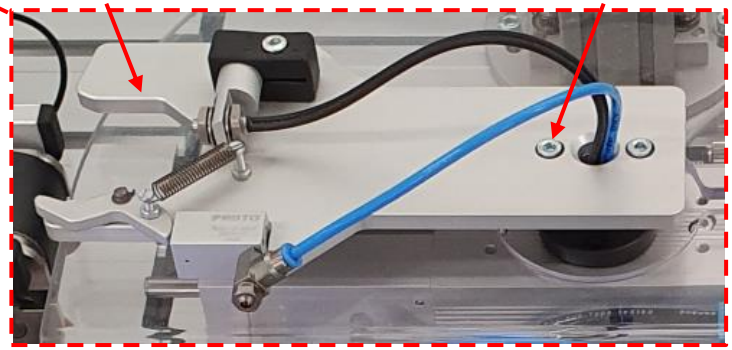
- Task: press workpiece insert into housing



Fluidic muscle press

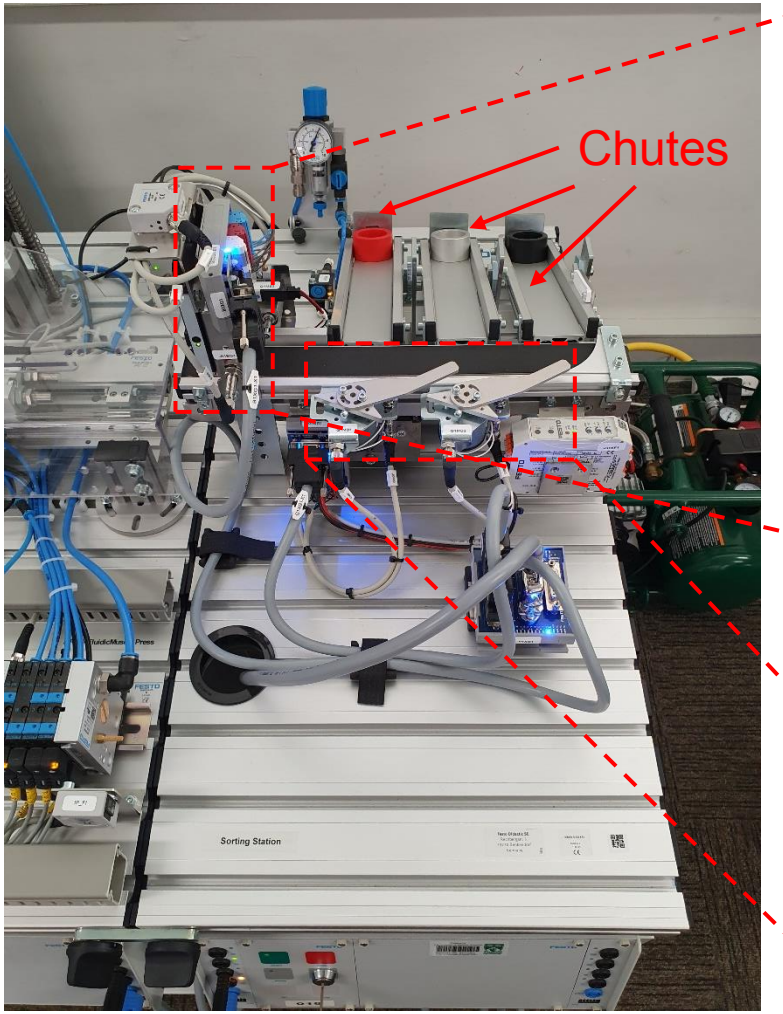
Gripper

Rotary drive

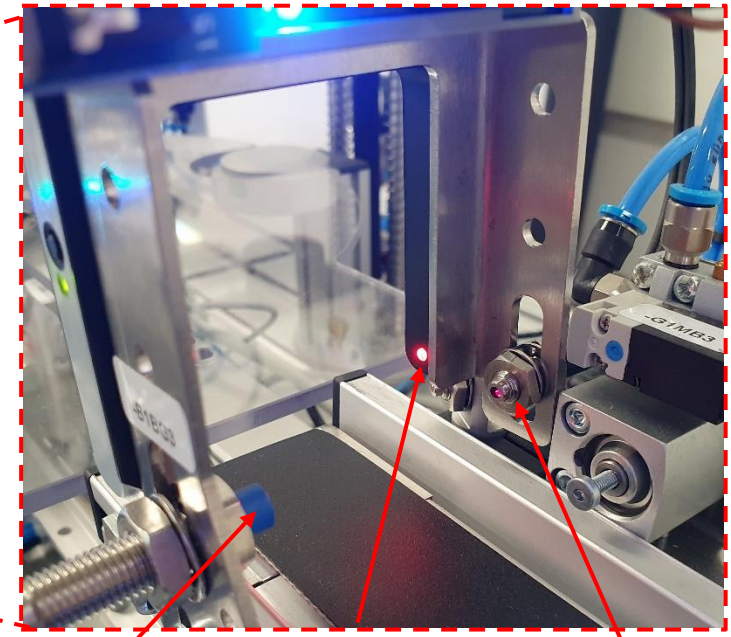


Festo: Sorting station

- Task: sorts assembled workpieces by housing



Detector module

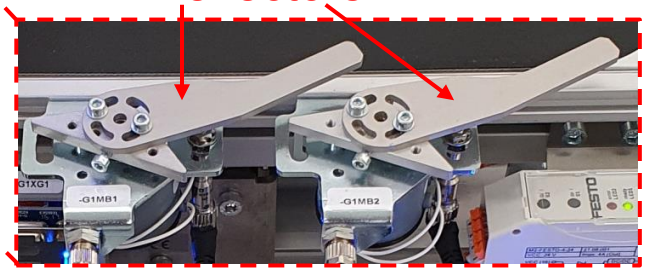


Inductive proximity sensor

Fork light barrier

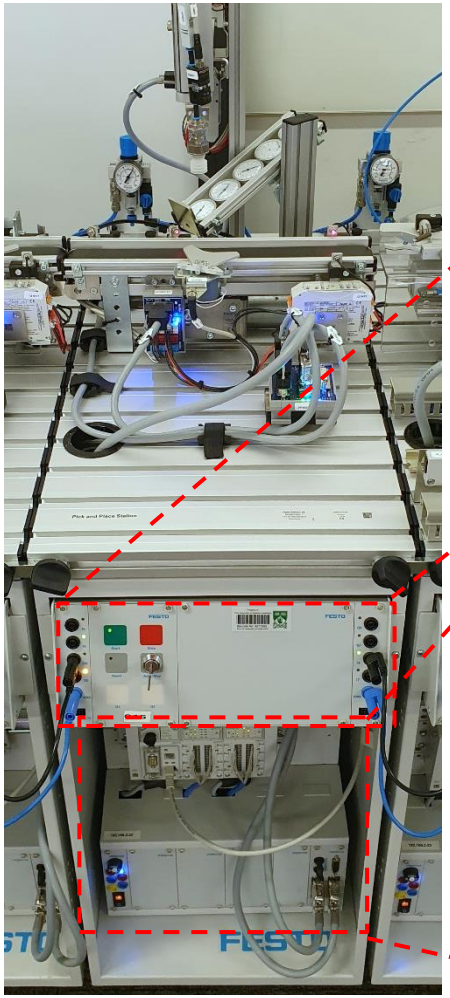
Diffuse light sensor

Deflectors

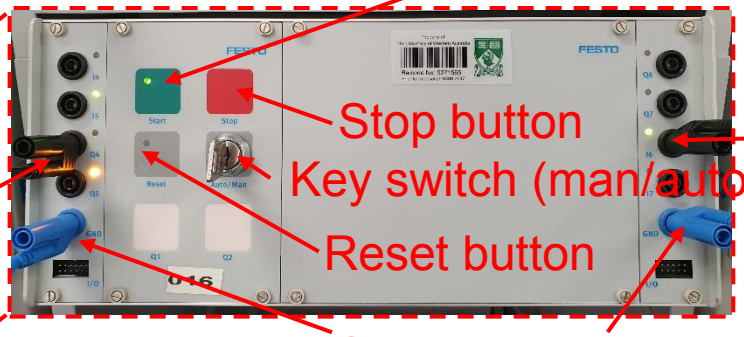


Festo: PLCs and control panels

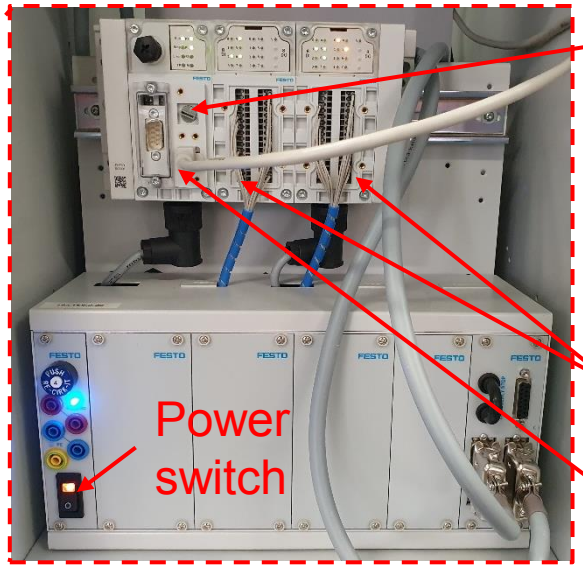
- Each station is programmed by a programmable logic controller (PLC) module and manually controlled from the control panel



Control panel

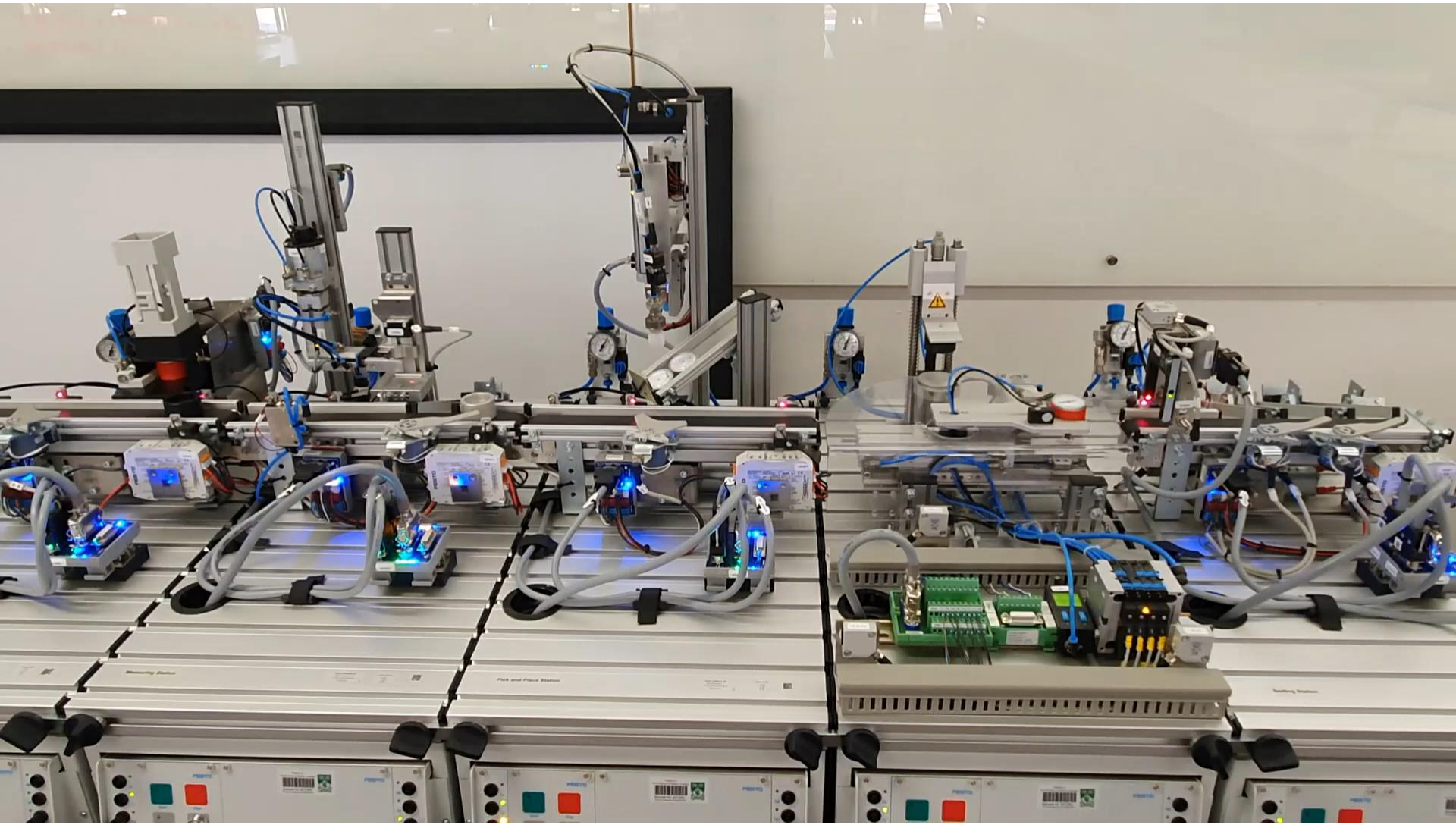


PLC module



Switch position	Setting	Meaning
	0	STOP Codesys controller stopped
	1 ... F	RUN Codesys controller started. Factory setting: 1

Festo: Video demonstration



Project description

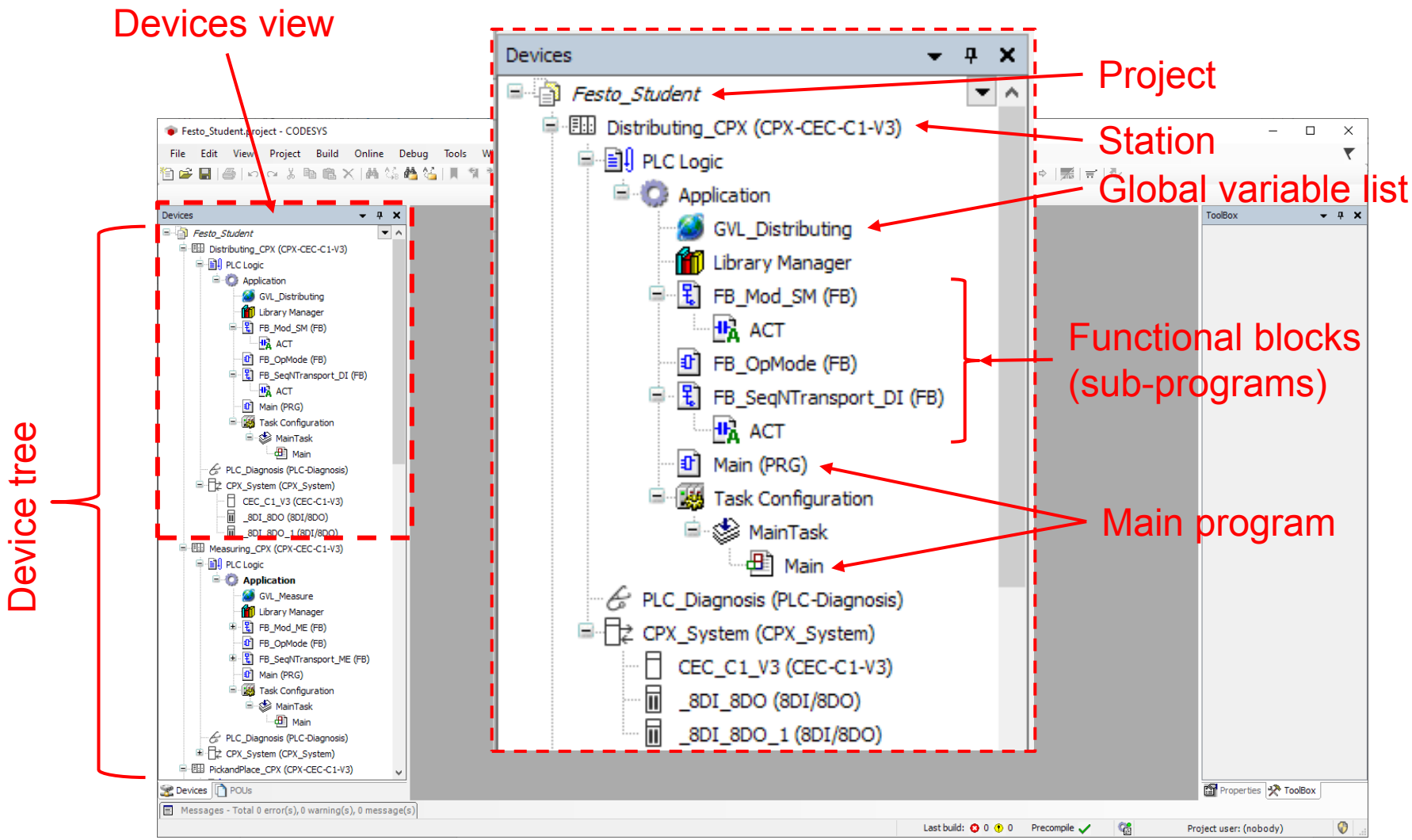
- The aim of the project is to program one Festo MPS station to complete the tasks previously outlined
- Students will work in groups of 3–4
- The stations will be programmed using CODESYS
- Partially complete code will be provided – students are required to complete the missing sections
- The lab consists of three parts:
  - Prelab: Read provided documentation and complete prelab questions
  - First lab (Week 3): Match variables defined in CODESYS with the corresponding PLC input, and describe the physical action associated with it
  - Second lab (Week 4): Each group in a session will be assigned one station and complete the code for that station. The code will then be uploaded to the station to demonstrate functionality.

CODESYS overview

- CODESYS (COntroller DEvelopment SYStem) is a development environment for PLCs
- **Windows required – not available for MAC!**
- Programming is implemented in the open international standard IEC61131-3
- This includes three graphical and two textual languages:
  - Ladder diagram (LD, graphical)
  - Functional block diagram (FBD, graphical)
  - Structured text (ST, textual)
  - Instruction list (IL, textual, deprecated)
  - Sequential function chart (SFC, graphical)

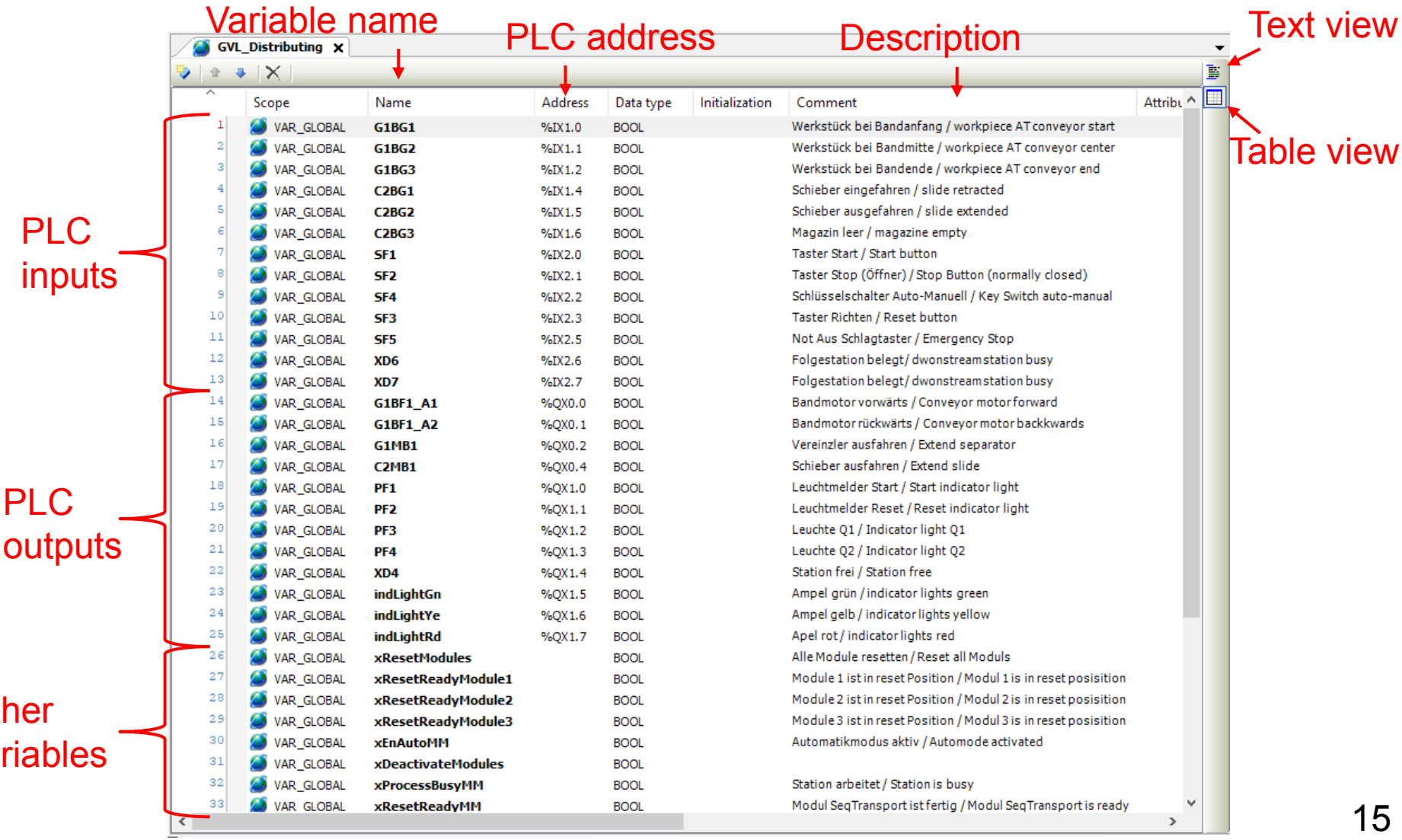
CODESYS

CODESYS Main interface



Global variable list (Distributing station)

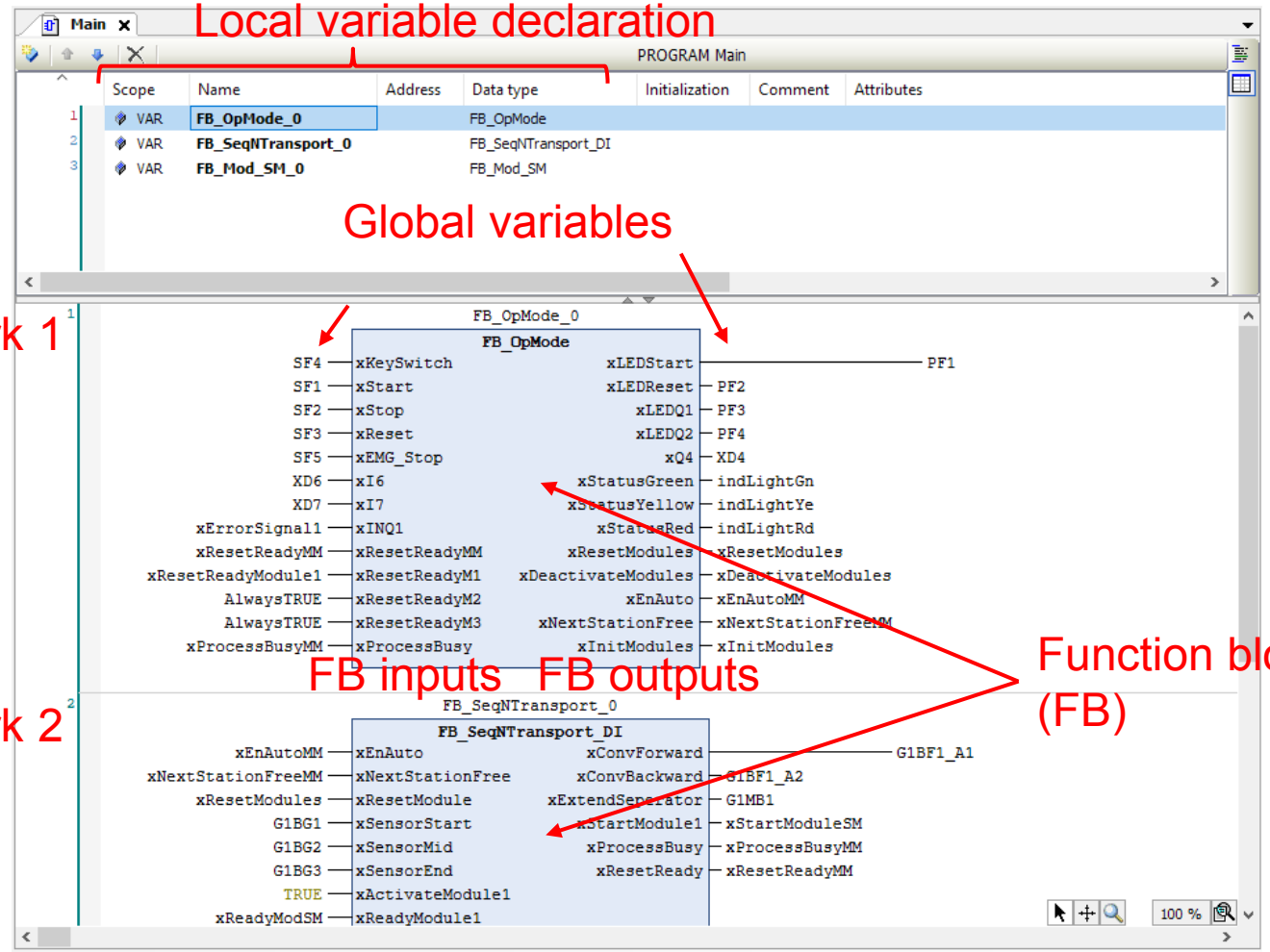
- Defines/declares global variables (i.e., variables available in all sub-programs)



	Scope	Name	Address	Data type	Initialization	Comment	Attribu
1	VAR_GLOBAL	G1BG1	%IX1.0	BOOL		Werkstück bei Bandanfang / workpiece AT conveyor start	
2	VAR_GLOBAL	G1BG2	%IX1.1	BOOL		Werkstück bei Bandmitte / workpiece AT conveyor center	
3	VAR_GLOBAL	G1BG3	%IX1.2	BOOL		Werkstück bei Bandende / workpiece AT conveyor end	
4	VAR_GLOBAL	C2BG1	%IX1.4	BOOL		Schieber eingefahren / slide retracted	
5	VAR_GLOBAL	C2BG2	%IX1.5	BOOL		Schieber ausgefahren / slide extended	
6	VAR_GLOBAL	C2BG3	%IX1.6	BOOL		Magazin leer / magazine empty	
7	VAR_GLOBAL	SF1	%IX2.0	BOOL		Taster Start / Start button	
8	VAR_GLOBAL	SF2	%IX2.1	BOOL		Taster Stop (Öffner) / Stop Button (normally closed)	
9	VAR_GLOBAL	SF4	%IX2.2	BOOL		Schlüsselschalter Auto-Manuell / Key Switch auto-manual	
10	VAR_GLOBAL	SF3	%IX2.3	BOOL		Taster Richten / Reset button	
11	VAR_GLOBAL	SF5	%IX2.5	BOOL		Not Aus Schlagtaster / Emergency Stop	
12	VAR_GLOBAL	XD6	%IX2.6	BOOL		Folgestation belegt/ dwonstreamstation busy	
13	VAR_GLOBAL	XD7	%IX2.7	BOOL		Folgestation belegt/ dwonstreamstation busy	
14	VAR_GLOBAL	G1BF1_A1	%QX0.0	BOOL		Bandmotor vorwärts / Conveyor motor forward	
15	VAR_GLOBAL	G1BF1_A2	%QX0.1	BOOL		Bandmotor rückwärts / Conveyor motor backkwards	
16	VAR_GLOBAL	G1MB1	%QX0.2	BOOL		Vereinzler ausfahren / Extend separator	
17	VAR_GLOBAL	C2MB1	%QX0.4	BOOL		Schieber ausfahren / Extend slide	
18	VAR_GLOBAL	PF1	%QX1.0	BOOL		Leuchtmelder Start / Start indicator light	
19	VAR_GLOBAL	PF2	%QX1.1	BOOL		Leuchtmelder Reset / Reset indicator light	
20	VAR_GLOBAL	PF3	%QX1.2	BOOL		Leuchte Q1 / Indicator light Q1	
21	VAR_GLOBAL	PF4	%QX1.3	BOOL		Leuchte Q2 / Indicator light Q2	
22	VAR_GLOBAL	XD4	%QX1.4	BOOL		Station frei / Station free	
23	VAR_GLOBAL	indLightGn	%QX1.5	BOOL		Ampel grün / indicator lights green	
24	VAR_GLOBAL	indLightYe	%QX1.6	BOOL		Ampel gelb / indicator lights yellow	
25	VAR_GLOBAL	indLightRd	%QX1.7	BOOL		Apel rot / indicator lights red	
26	VAR_GLOBAL	xResetModules		BOOL		Alle Module resetten / Reset all Moduls	
27	VAR_GLOBAL	xResetReadyModule1		BOOL		Module 1 ist in reset Position / Modul 1 is in reset posisition	
28	VAR_GLOBAL	xResetReadyModule2		BOOL		Module 2 ist in reset Position / Modul 2 is in reset posisition	
29	VAR_GLOBAL	xResetReadyModule3		BOOL		Module 3 ist in reset Position / Modul 3 is in reset posisition	
30	VAR_GLOBAL	xEnAutoMM		BOOL		Automatikmodus aktiv / Automode activated	
31	VAR_GLOBAL	xDeactivateModules		BOOL			
32	VAR_GLOBAL	xProcessBusyMM		BOOL		Station arbeitet / Station is busy	
33	VAR_GLOBAL	xResetReadyMM		BOOL		Modul SeqTransport ist fertig / Modul SeqTransport is ready	

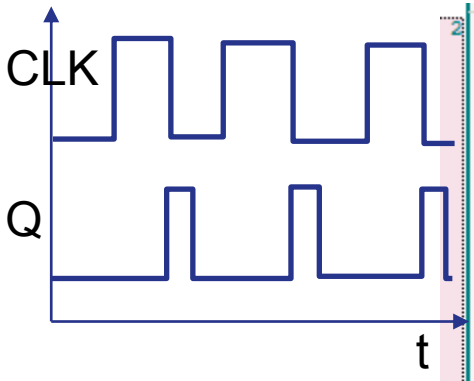
Main program (Distributing station)

- PLC loops over Main program every 20ms
- In each cycle, networks in Main are executed sequentially

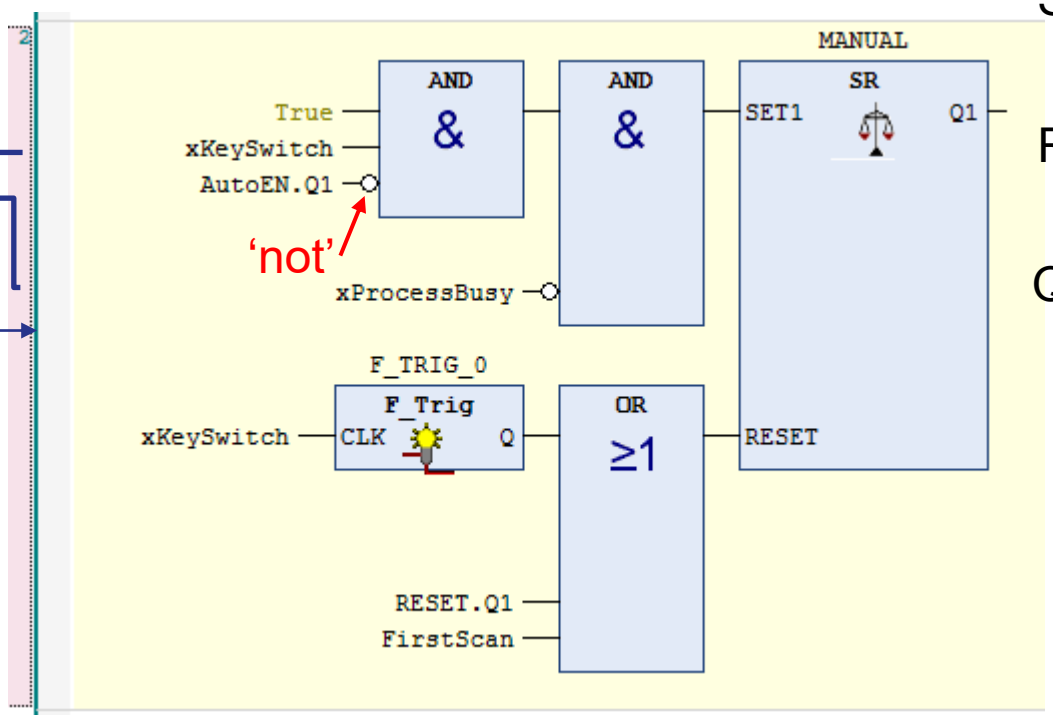


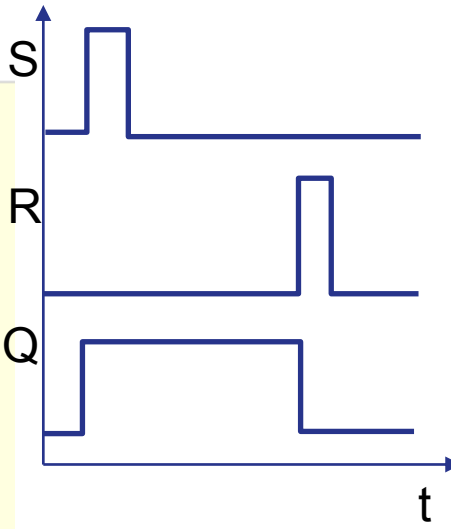
Functional block diagram (FBD) example

- Denoted by  in Device tree
- Networks executed sequentially
- Each network executed left-to-right (inputs on left, outputs on right)
- Common functions (AND, OR) and function blocks (SR latch, falling trigger) included in base library



Falling trigger: Set Q=1 when CLK transitions 1->0 for one cycle, then reset Q=0

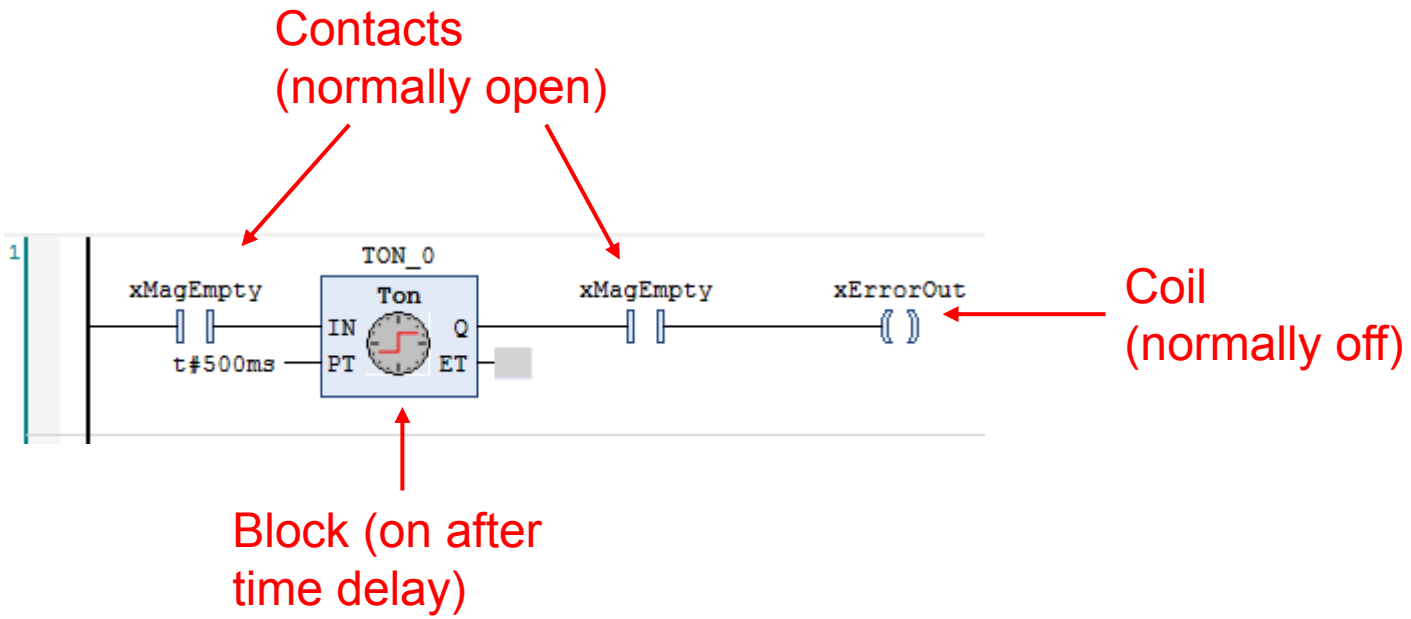




SR latch: Set Q=1 on rising edge of S, reset Q=0 on rising edge of R

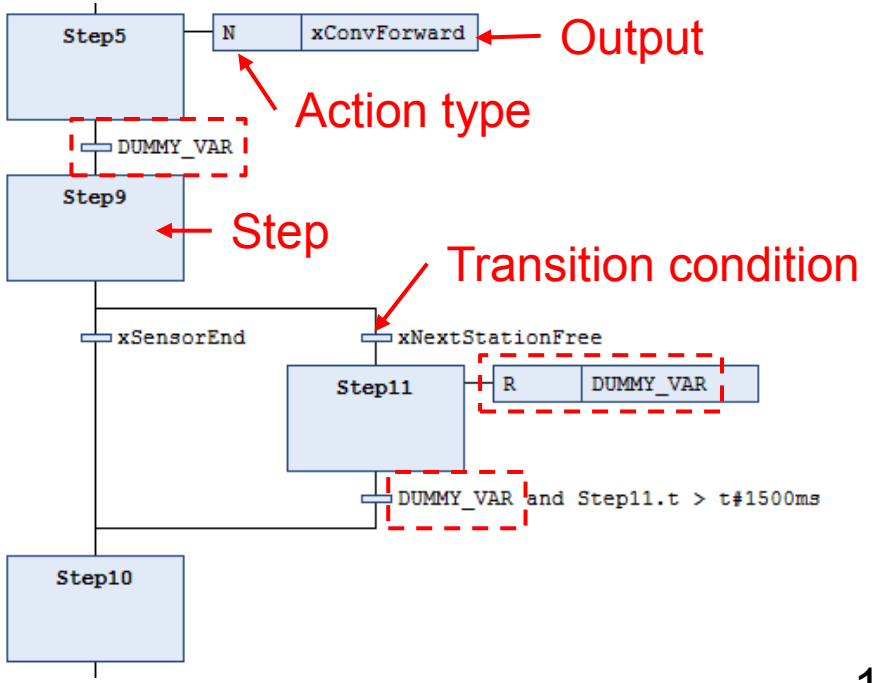
Ladder diagram (LD) example

- Denoted by  in Device tree
- Networks executed sequentially
- Each network executed left-to-right
- Each network consists of ‘contacts’, ‘blocks’, and ‘coils’:
  - Contacts: analogous to switches. Denoted | | (normally open) or | \ | (normally closed)
  - Blocks: implements higher-level functions
  - Coils: analogous to devices, e.g. lightbulb. Denoted ( ) (normally off) or (\ ) (normally on)

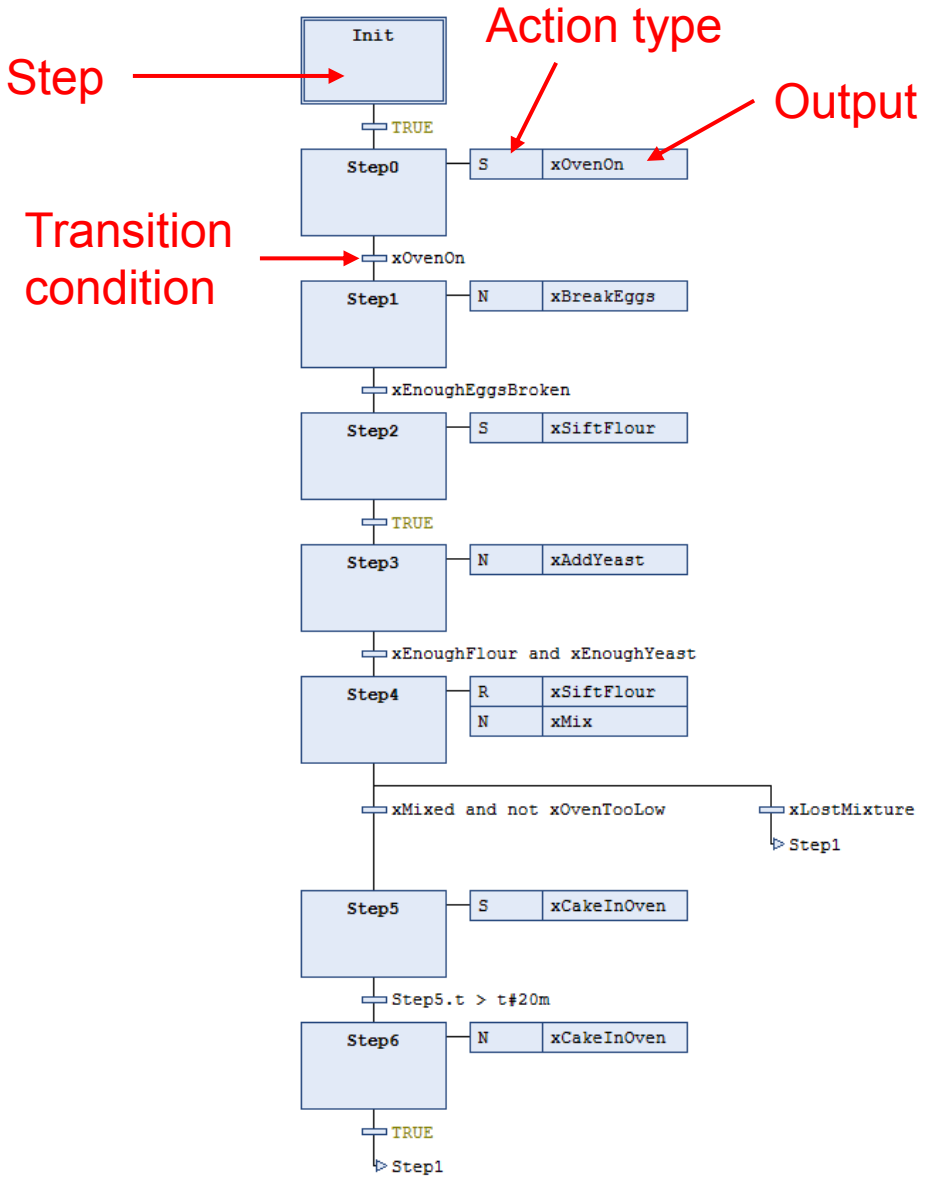


Sequential Flow Chart (SFC) example

- Denoted by  in Device tree
- Flows from top to bottom, and consists of steps, actions and transitions
- A step may be **active** or **inactive**.
  - A step becomes active when it is the first step, or when all immediately preceding steps are active and the transition condition is True.
  - A step becomes inactive when the succeeding transition is true
- Actions associated with steps are only executed when the step is active.
- Actions have an output and type. Types include:
  - S: Set output to 1
  - R: Reset output to 0
  - N: Set output to 1 while step is active, then reset to 0 when step is inactive



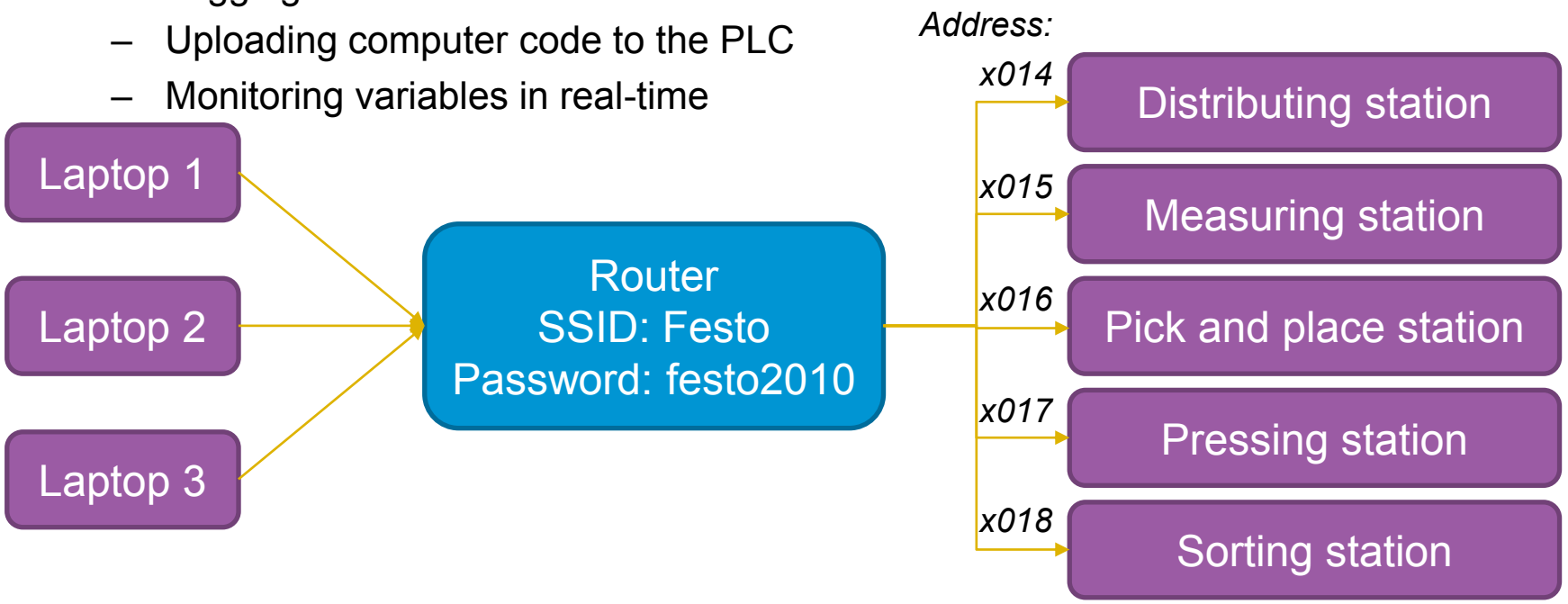
SFC example: Baking a cake



- Action types:
  - S: sets output high
  - N: sets output high while step is active
  - R: sets output low
- Step0: oven is set (S) to on and will stay on until it is reset (R)
- Step1: output *xBreakEggs* will remain on only while the condition *xEnoughEggsBroken* is not true (N)
- Step2-3: The Boolean *True* can be used to proceed immediately from Step 2 to 3
- Step3-4: Functions such as *and* and *not* can be used in the conditions
- Step4: Multiple actions can be associated with each step (executed sequentially)
- Step4-5: The branch implements *or*. Conditions are evaluated left-to-right until one is true.
- Step5-6: The time taken to complete a step can also be used as a condition

Connecting to Festo MPS

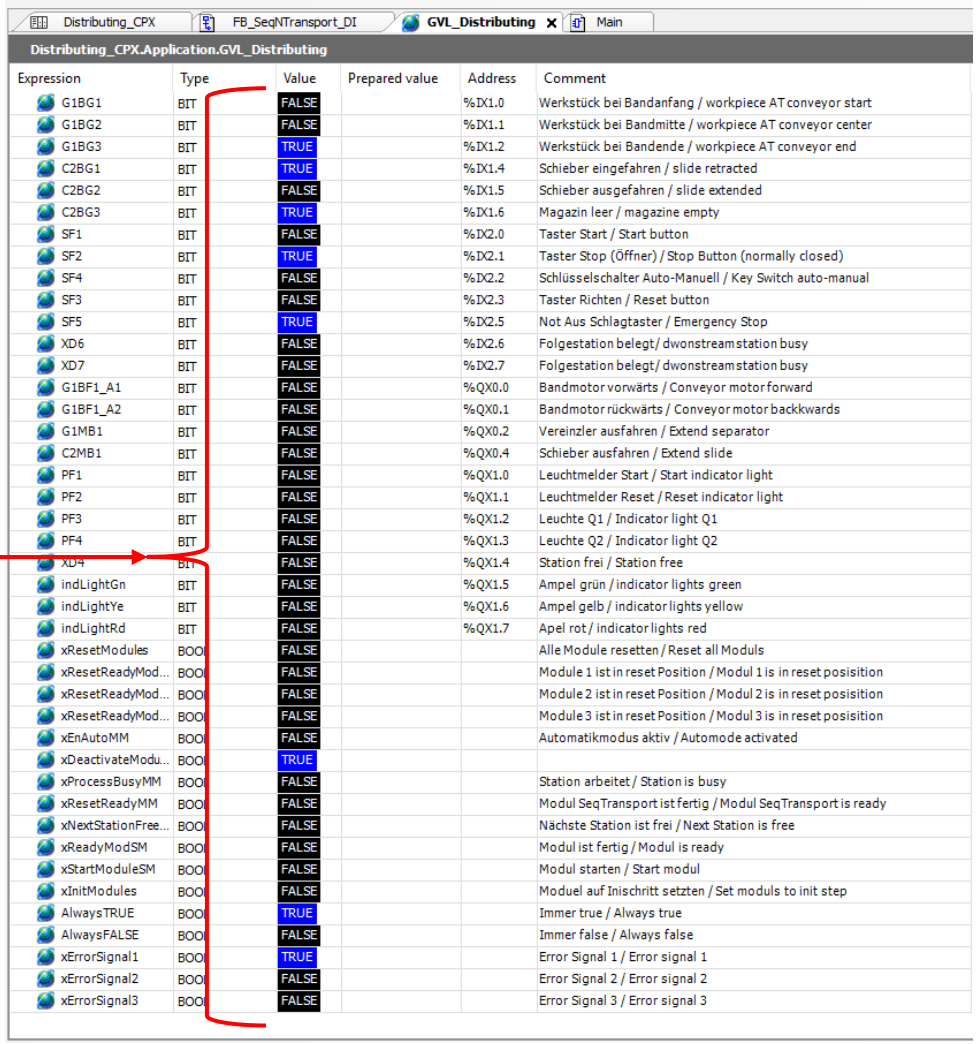
- The steps required to connect to Festo MPS are provided at the unit website in LabA-Festo_CODESYSBasics.pdf
- These steps included:
  - Connecting to the router:
    - All stations are connected to the same router
    - Note: no internet connection available through router
  - Assigning the station to a network address in CODESYS
  - Logging in to the station in CODESYS
  - Uploading computer code to the PLC
  - Monitoring variables in real-time



Monitoring variables in real-time

- Once connected to a Festo station, the state of variables can be monitored in real-time (this will be particularly useful for the first lab)

Current variable state



Distributing_CPX | FB_SeqNTransport_DI | GVL_Distributing | Main

Distributing_CPX.Application.GVL_Distributing

Expression	Type	Value	Prepared value	Address	Comment
G1BG1	BIT	FALSE		%IX1.0	Werkstück bei Bandanfang / workpiece AT conveyor start
G1BG2	BIT	FALSE		%IX1.1	Werkstück bei Bandmitte / workpiece AT conveyor center
G1BG3	BIT	TRUE		%IX1.2	Werkstück bei Bandende / workpiece AT conveyor end
C2BG1	BIT	TRUE		%IX1.4	Schieber eingefahren / slide retracted
C2BG2	BIT	FALSE		%IX1.5	Schieber ausgefahren / slide extended
C2BG3	BIT	TRUE		%IX1.6	Magazin leer / magazine empty
SF1	BIT	FALSE		%IX2.0	Taster Start / Start button
SF2	BIT	TRUE		%IX2.1	Taster Stop (Öffner) / Stop Button (normally closed)
SF4	BIT	FALSE		%IX2.2	Schlüsselschalter Auto-Manuell / Key Switch auto-manual
SF3	BIT	FALSE		%IX2.3	Taster Richten / Reset button
SF5	BIT	TRUE		%IX2.5	Not Aus Schlagtaster / Emergency Stop
XD6	BIT	FALSE		%IX2.6	Folgestation belegt/ dwonstreamstation busy
XD7	BIT	FALSE		%IX2.7	Folgestation belegt/ dwonstreamstation busy
G1BF1_A1	BIT	FALSE		%QX0.0	Bandmotor vorwärts / Conveyor motor forward
G1BF1_A2	BIT	FALSE		%QX0.1	Bandmotor rückwärts / Conveyor motor backkwards
G1MB1	BIT	FALSE		%QX0.2	Vereinzler ausfahren / Extend separator
C2MB1	BIT	FALSE		%QX0.4	Schieber ausfahren / Extend slide
PF1	BIT	FALSE		%QX1.0	Leuchtmelder Start / Start indicator light
PF2	BIT	FALSE		%QX1.1	Leuchtmelder Reset / Reset indicator light
PF3	BIT	FALSE		%QX1.2	Leuchte Q1 / Indicator light Q1
PF4	BIT	FALSE		%QX1.3	Leuchte Q2 / Indicator light Q2
XD4	BIT	FALSE		%QX1.4	Station frei / Station free
indLightGn	BIT	FALSE		%QX1.5	Ampel grün / indicator lights green
indLightYe	BIT	FALSE		%QX1.6	Ampel gelb / indicator lights yellow
indLightRd	BIT	FALSE		%QX1.7	Apel rot / indicator lights red
xResetModules	BOO	FALSE			Alle Module resetten / Reset all Moduls
xResetReadyMod...	BOO	FALSE			Module 1 ist in reset Position / Modul 1 is in reset posisition
xResetReadyMod...	BOO	FALSE			Module 2 ist in reset Position / Modul 2 is in reset posisition
xResetReadyMod...	BOO	FALSE			Module 3 ist in reset Position / Modul 3 is in reset posisition
xEnAutoMM	BOO	FALSE			Automatikmodus aktiv / Automode activated
xDeactivateModu...	BOO	TRUE			
xProcessBusyMM	BOO	FALSE			Station arbeitet / Station is busy
xResetReadyMM	BOO	FALSE			Modul SeqTransport ist fertig / Modul SeqTransport is ready
xNextStationFree...	BOO	FALSE			Nächste Station ist frei / Next Station is free
xReadyModSM	BOO	FALSE			Modul ist fertig / Modul is ready
xStartModuleSM	BOO	FALSE			Modul starten / Start modul
xInitModules	BOO	FALSE			Moduel auf Inischritt setzten / Set moduls to init step
AlwaysTRUE	BOO	TRUE			Immer true / Always true
AlwaysFALSE	BOO	FALSE			Immer false / Always false
xErrorSignal1	BOO	TRUE			Error Signal 1 / Error signal 1
xErrorSignal2	BOO	FALSE			Error Signal 2 / Error signal 2
xErrorSignal3	BOO	FALSE			Error Signal 3 / Error signal 3

Practical details and preparation

- Schedule: 2x 3-hour (2 hours supervised) labs during Weeks 3–4
- Location: Clough Engineering Centre (Engineering building 224)
- Groups of 3–4
- Preparation:
  - Download lab worksheet (LabA-Festo_Prelab&Instructions.pdf [1]) and do prelab using Festo PrelabDocumentation/ [2]
  - Review LabA-Festo_CODESYSBasics.pdf [3]
  - If possible, form groups before lab
  - Download CODESYS3.5.13.0.exe [4] and source code (Festo_Student.projectarchive [5]) and install CODESYS (1 computer per group)
    - Charge your laptops! Not many power points available in venue

week	cal-wk	week start	lectures A (2h)	lectures B (2h)	reading	project	lab prep.	lab	lab contents
1	8	22-Feb	Org., 0.Introduction	1. Automotive	Ch.1+26		--		---
2	9	1-Mar	2. Automation	3. Manipulators	Ch. 14		Automation		---
3	10	8-Mar	3. Manipulators	4.Driving Robots	8 – 13		--	A1	Automation
4	11	15-Mar	5. Kinematics	6. Walk + 7.AUV/UAV	Ch. 16		EyeSim	A2	Automation