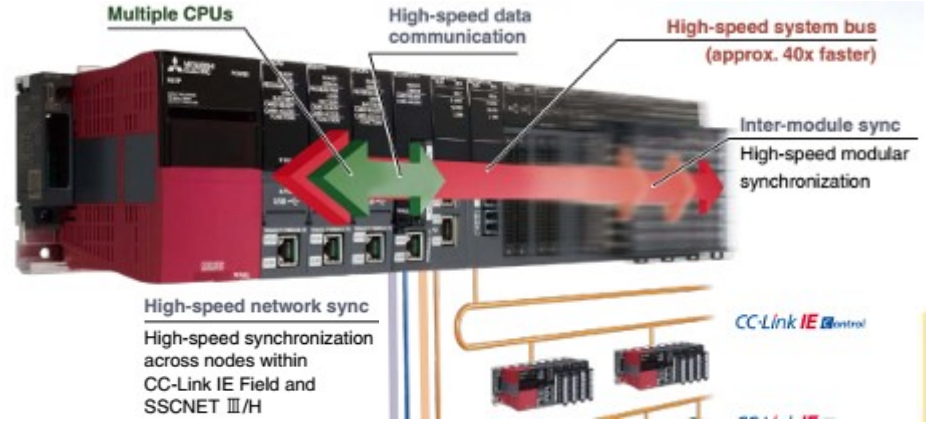
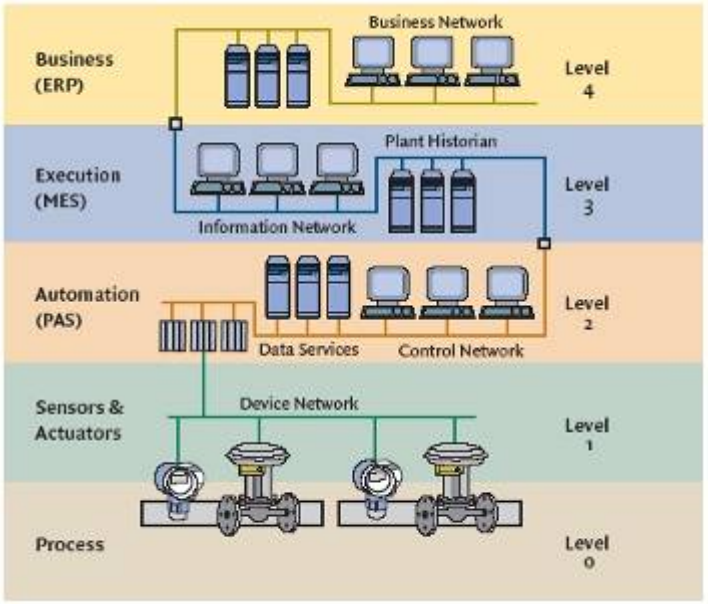
Programmable Logic Controllers (PLC) & Industrial Automation Overview

Nereus Fernandes





Agenda

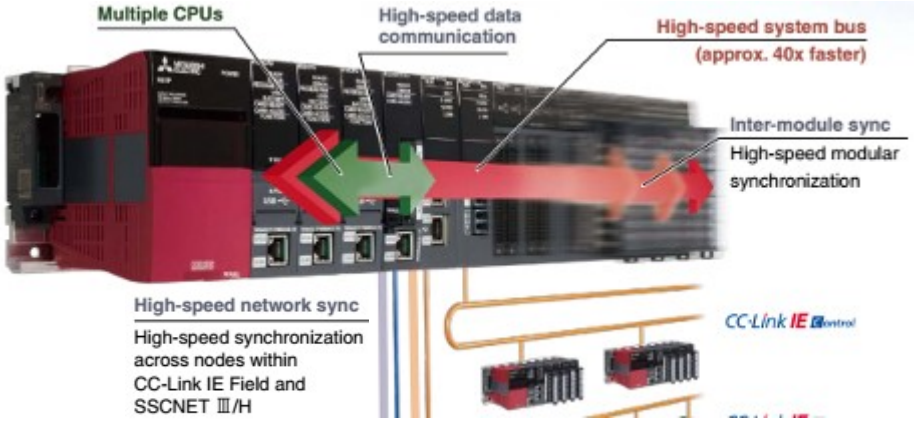
- Basic Theory, PLC Types
- SCADA-HMI
- Drive Control using PLC
- Protocols-Modbus, Profinet, OPC UA
- PLC Selection, Programming Guidelines
- Industrial Automation Hierarchy
- Other Closely Related Controllers and Networks
  - Safety Instrumented Systems
  - Building Automation Systems
  - Substation Automation Systems & Electrical SCADA
  - DCS (Distributed Control System)
  - RTU (Remote Terminal Unit)
  - Machine Vision
  - Position & Motion Systems
  - Robotics
  - CNC Machines
  - PAC (Programmable Automation Controller)
- Emerging Technologies
  - MQTT Protocol & PLC intergation with (Industrial) IIoT and the Cloud
  - Assisted / Augmented Reality
  - Simulation & Virtual Reality
- SCADA – support for Pharma Regulations, GeoSCADA
- H/w & S/w to build a PLC, HMI, SCADA & IoT Systems
- Industrial Cyber Security
- Dual Use Technology

PLC Trends…

- Today the PLC is increasingly integrated into the Industrial Automation and IT networks
- PLCs today have extensive functionality and are much more than just Ladder Logic…
  - Besides programming the PLC, the Automation Engineer is also required to set-up network infrastructure for Ethernet & Field-buses, to code scripts for SCADA, SQL Queries for Database & API-calls for ERP and integration with the Cloud
- Wide and often confusing choices of PLC vendors, hardware platforms and software programming options are now available
  - The ultimate choice of the hardware and software is driven by : required features, cost, expertise available, vendor support, historical use in the application – existing installations and the need for uniformity in the plant / across plants of an organization
- Most books and presentations available - only focus on basics and do not highlight the breadth of advanced PLC features and technology
- The aim of this presentation is
  - to provide a broad overview of the Industrial Automation market-place and show where the PLC fits in. The reader is encouraged to look at the technical documentation of the various vendors to learn more about products of interest
  - Highlight the advanced features available on most PLCs now-a-days
  - to stimulate interest in the inner workings, design and construction of the PLC hardware and software - rather than treating it as a black-box

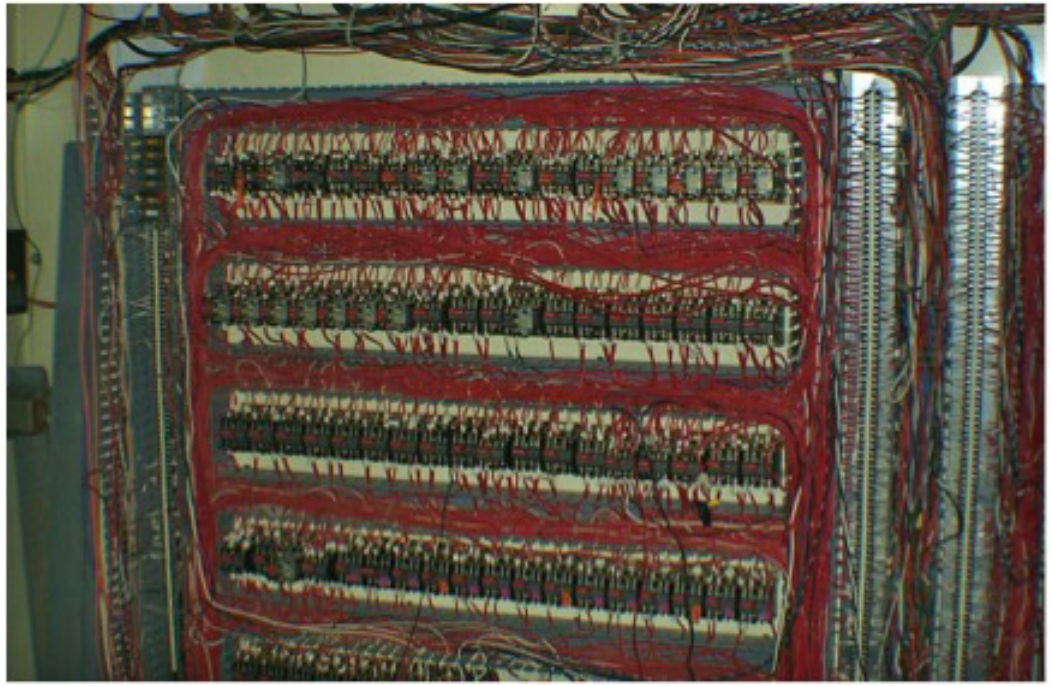
PLC Trends… Into the future…

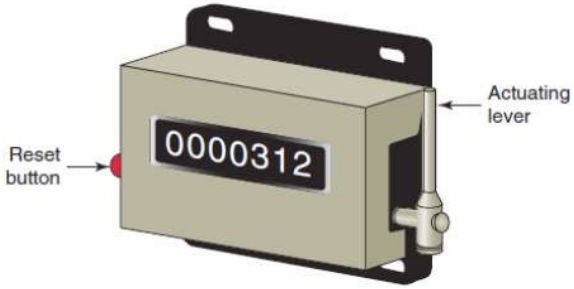
- The PLC has been the control and computational work-horse of the industry for the past 50 years
- Industrial IoT documentation in the early 2010s largely mentioned examples based on Arduino and RasberryPi hardware. This was to encourage trial of the principles on economical hardware. However it could lead to the assumption that PLCs were not suitable for IoT and did not have much of a future. While initially PLC manufacturers were a little slow to react – today there is broad and robust support for IoT Technologies in PLC hardware and almost all Industrial Automation Software. We will also look at the software that transforms Arduinos and RasberryPis into fully functional PLCs.
- Ability to simultaneously use Multiple Processors – PLC-CPU, Motion Controller, CNC Controller, Robot Controller, Programmable Embedded System & FPGA Special Purpose Controllers, Tensor/Neural Processing Units, High Speed Data-Logger, Communication Modules with multiple network support, Large Memory – turn today's PLCs into powerful multiple processor platforms linked by High-Speed PLC Back-Plate Bus and other High-Speed Communication Busses
- With the continuous innovations in the PLC world .. the PLC will continue to be the industrial work-horse to well into the foreseeable future.
- There will be new ideas and platforms – the very mention of a non-domain company like Amazon in this presentation indicates major shifts in the industry. Amazon has propelled itself into this discussion by embracing the latest technologies in robotics, material handling, vision, AI, Big Data, Cloud Computing and leveraging them to improve it's efficiencies – at the same time providing frameworks and best practices for others to use and benefit.



PLC Basics

Relay Control Logic – used for machine control before PLCs





Solid-state timing relay

Pneumatic timing relay

Origins of the PLC

- 1968 General Motors controller requirements specification with Design Criteria for a ”standard machine controller”
  - Elimination of costly scrapping of assembly-line relays during model changeovers
  - Replace unreliable electro-mechanical relays
  - Reduction of machine downtime related to controls problems
  - Provide for future expansion, it had to be modular
  - Should work in an industrial environment

The First PLC

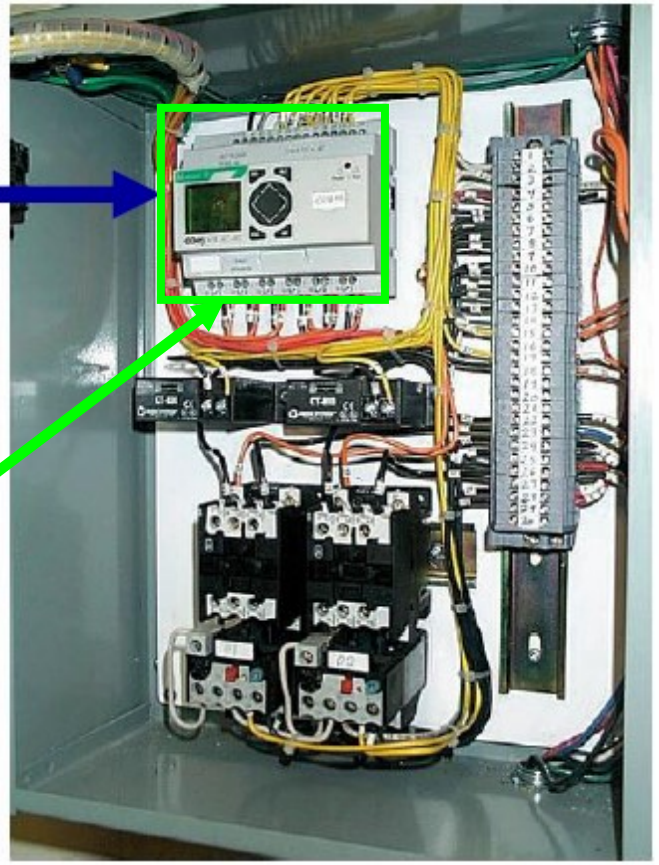
- In 1968 by Mr Richard Morley from Bedford Associates
- Modicon (Modular Digital Controller) sold the first commercial PLC - Modicon084
- 1979 Modbus Industrial Communication network specification

PLC Advantages-1

- Component count and space reduction
- Simplify wiring, reduce failures
- Flexible to easily accommodate changes
- When more than 10 relays are needed, a PLC will be cheaper

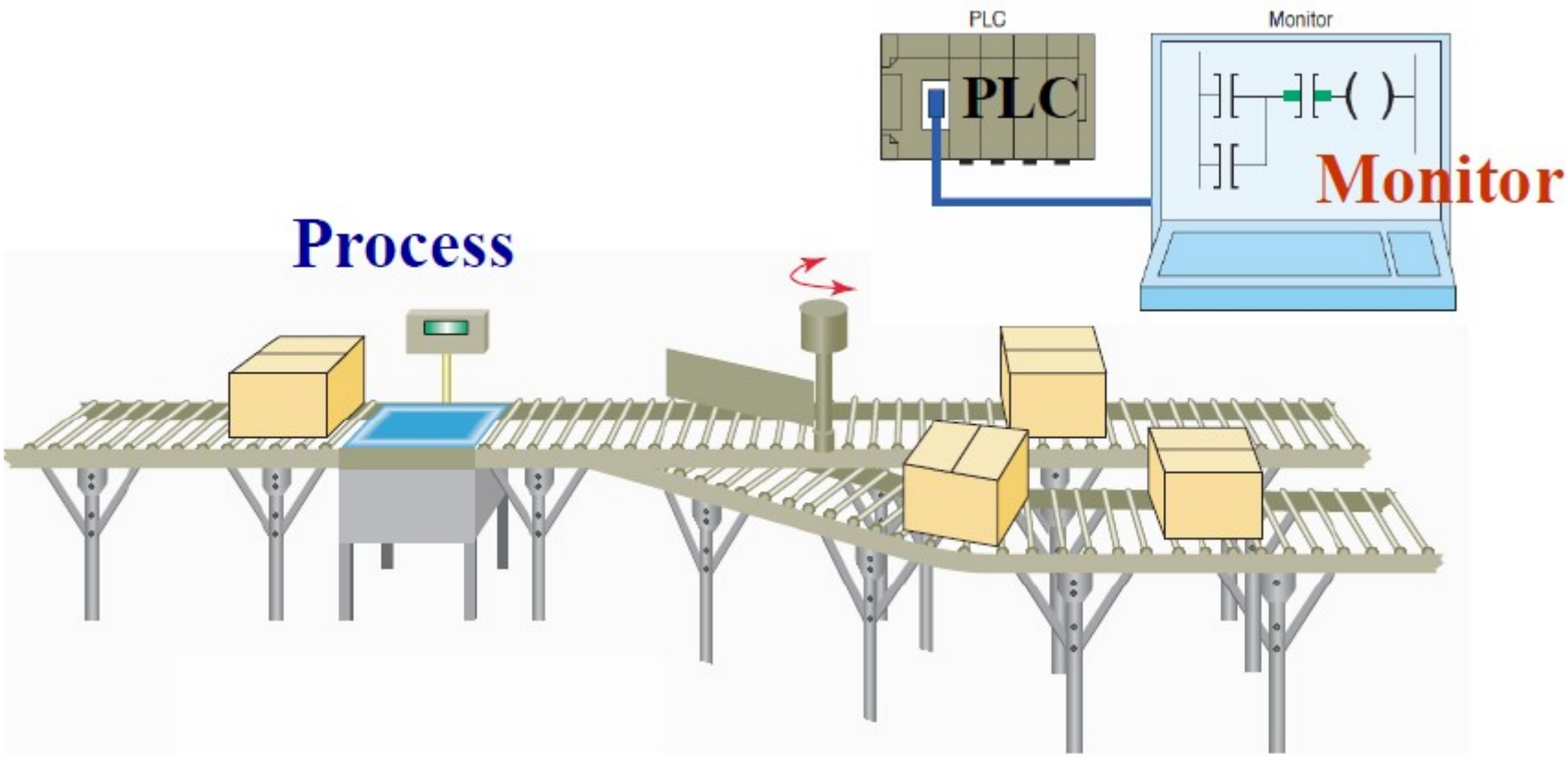
PLC control panel

Relay control panel

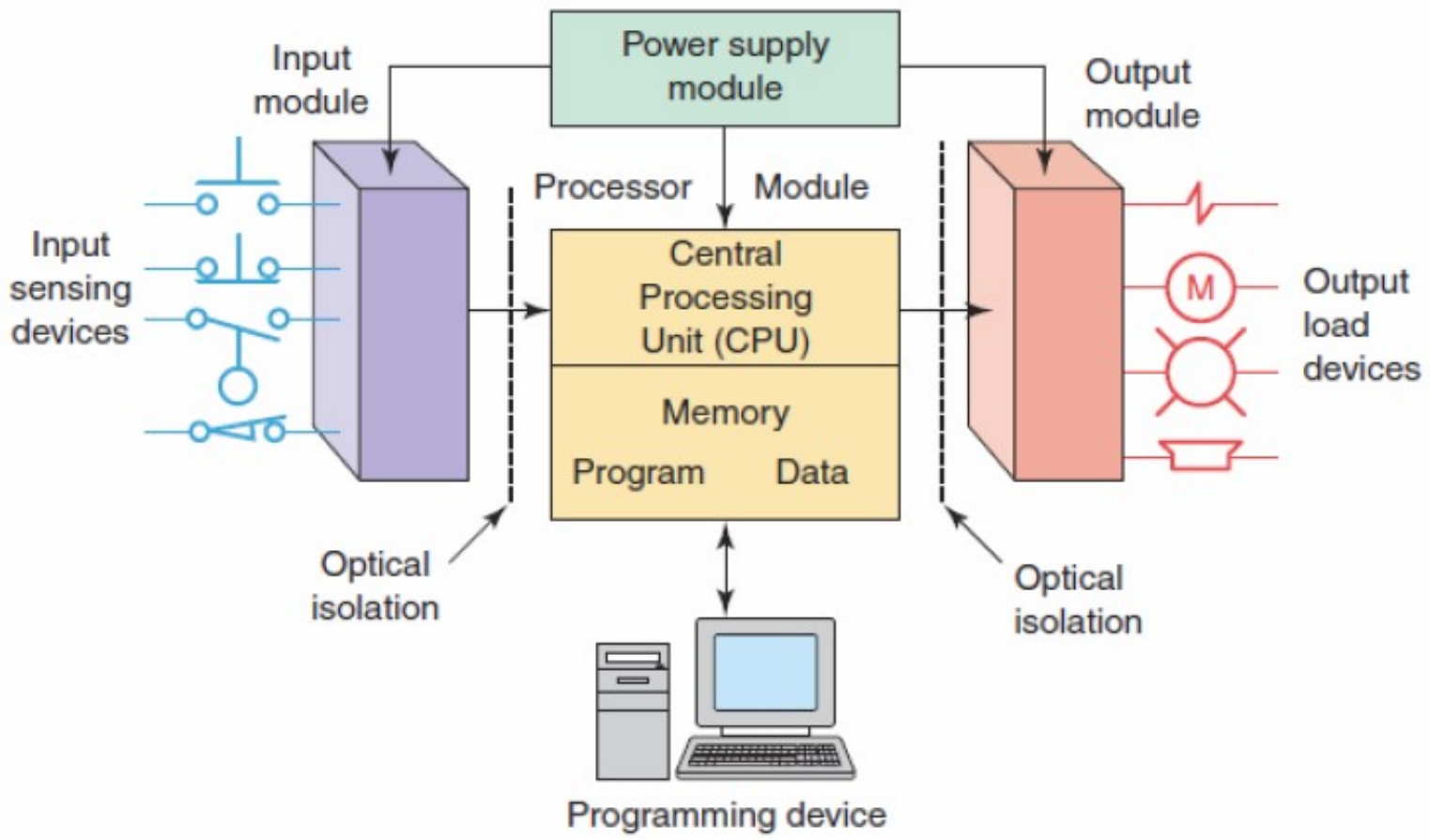


PLC Advantages-2

- Communication Interfaces
- Easier to troubleshoot



PLC – major sections



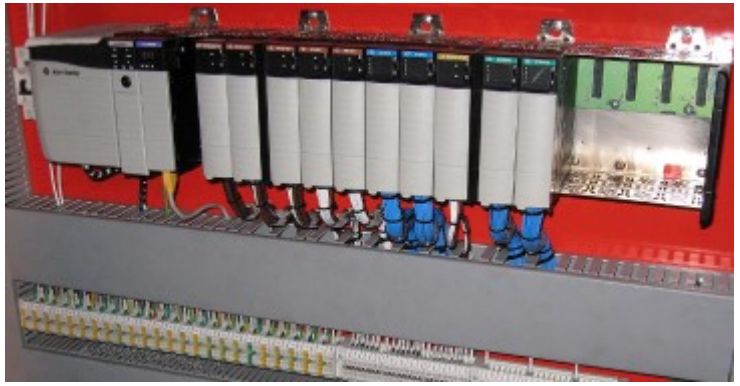
Compact PLC

The Compact PLCs have the PS, CPU, COMM and IOs in a single unit. It may be extended by plugging in additional modules

Mitsubishi

Modular PLC

In the Modular PLC the Backplane carries the power, address and data buses to the modules installed in the rack.
Modules have separate functionality : PS, CPU, COMM and IOs



Allen Bradley

IO Addressing
CPU needs to be configured with the locations and types of IO modules. DIP switches on the IO module may need to be set to the slot number
Alternatively the CPU may automatically recognize the plugged in module and assign it an address in continuation from left to right :
-- right most Input card is I0-7 next one is I8-15.. and so on.

Some More Types of PLCs

MM1010-V/I-230V

I5	I4	I3	I2	I1	I0	ANA IP+		ANA IP -	
1	2	3	4	5	6	7	8	9	10
22	I6							RS485 +	24
21	I7							RS485 -	23
11	12	13	14	15	16	17	18	19	20
+24V	GND	NO0	NO1	NO2	NO3	NO4	COM	L(+)	N(-)

Input Device:

Device	X0	X1	X2	X3	X4	X5	X6	X7	X10	X11	X12	X13	X14	X15	X16	X17
1	FWD	REV	M11	M12	M13	M14	M15	M16	M17	M18						

1: Control board I/O

Output Device:

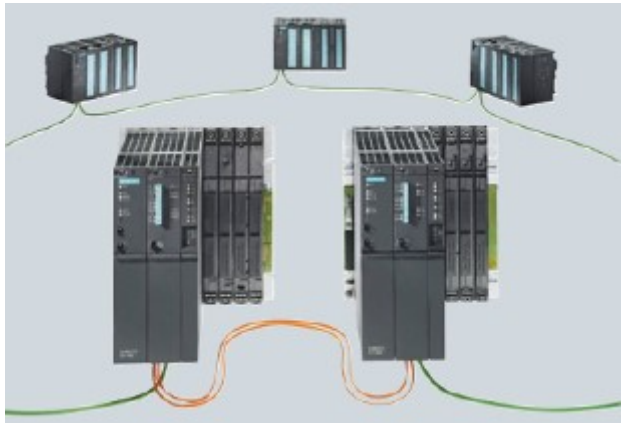
Device	Y0	Y1	Y2	Y3	Y4	Y5	Y6	Y7	Y10	Y11	Y12	Y13	Y14	Y15	Y16	Y17
1	RY1	RY2		DFM 1	DFM 2											

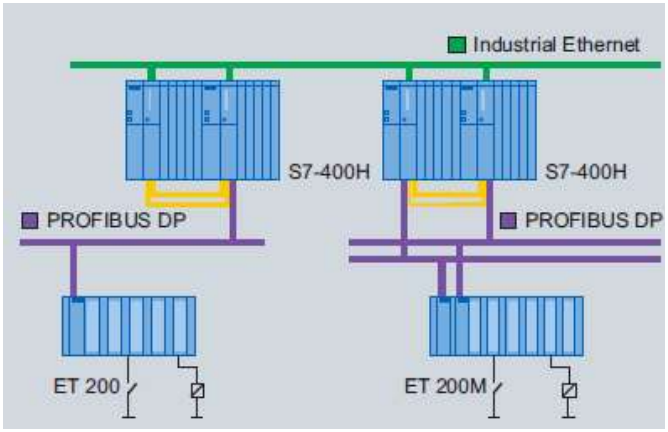
1: Control board I/O

PLCs with the form factor of a Digital Panel Meter
PLCs are also embedded in HMI and Drives

Siemens PLCs – various ranges

S7-400(H) CPU Hot-standby with redundant CPU configuration & event synchronization bus.
Changeover in 100ms max.
Redundant CPUs at max10Km apart.
Upto 65356 DIOs, 4096 AIOs
Redundancy is enabled by configuration and no modifications are required in the PLC program



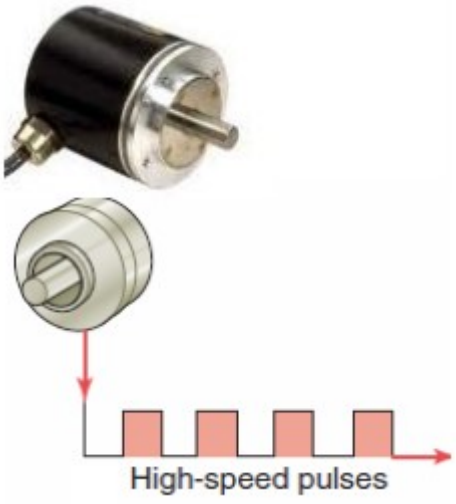


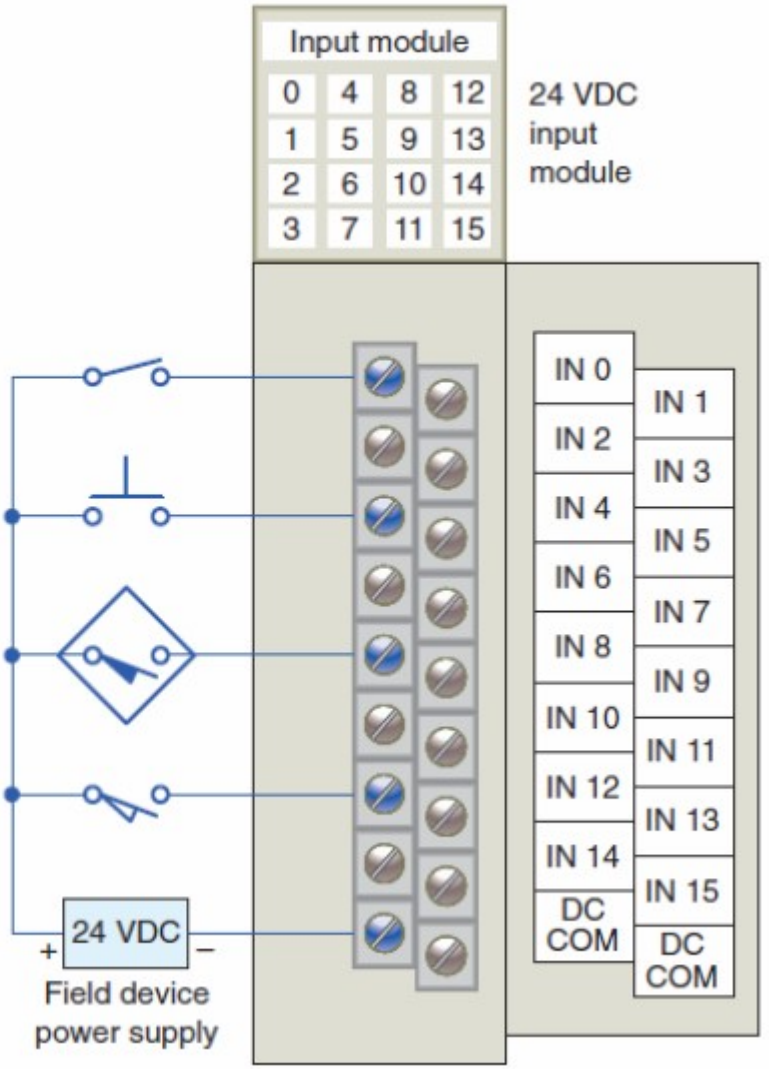
Mid-range **S7-300**(left) is superseded by S7-1500(right) with modern features for easy IT integration and networking.
S7-1500 is available in the traditional PLC hardware format and as a Soft-PLC running on an IPC platform

ET-200 Modular IO with S7-300/S7-1500 type CPUs for distributed computing

Low-end **S7-1200** for simple applications generally stand-alone

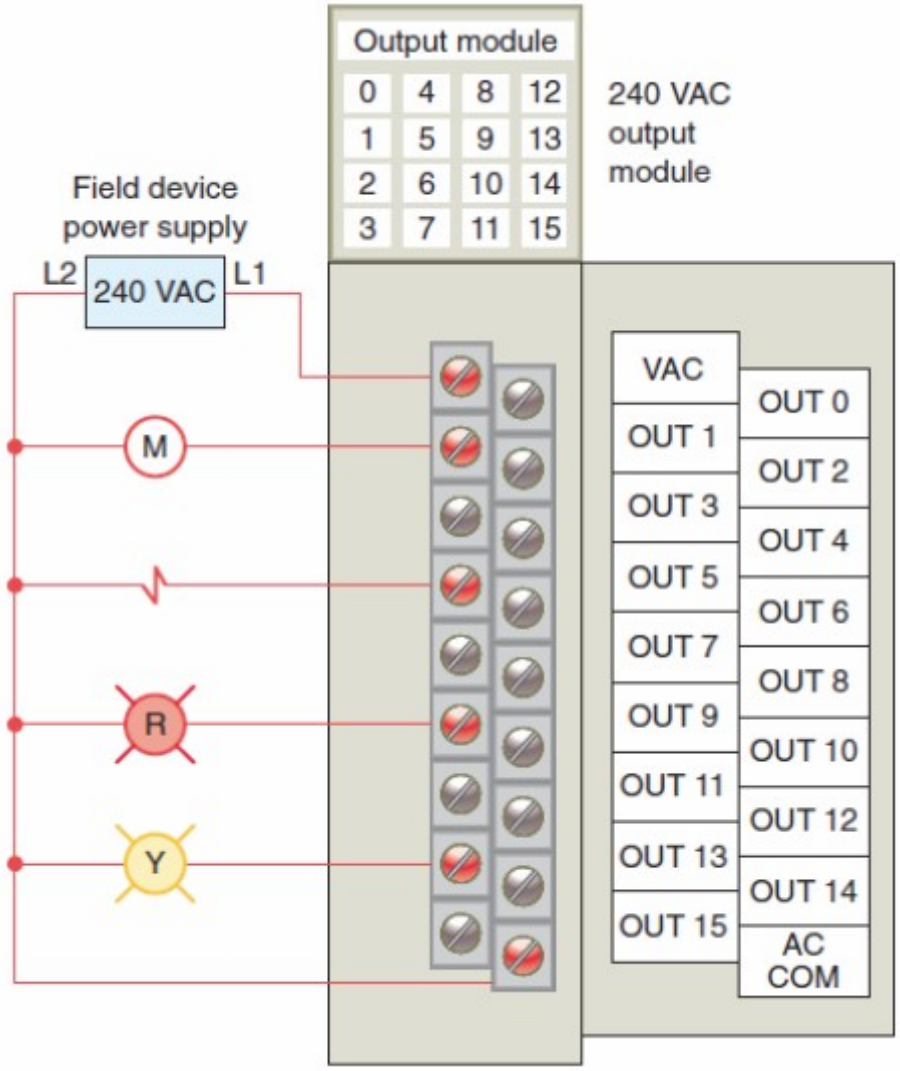
PLC-DI


High-speed pulses

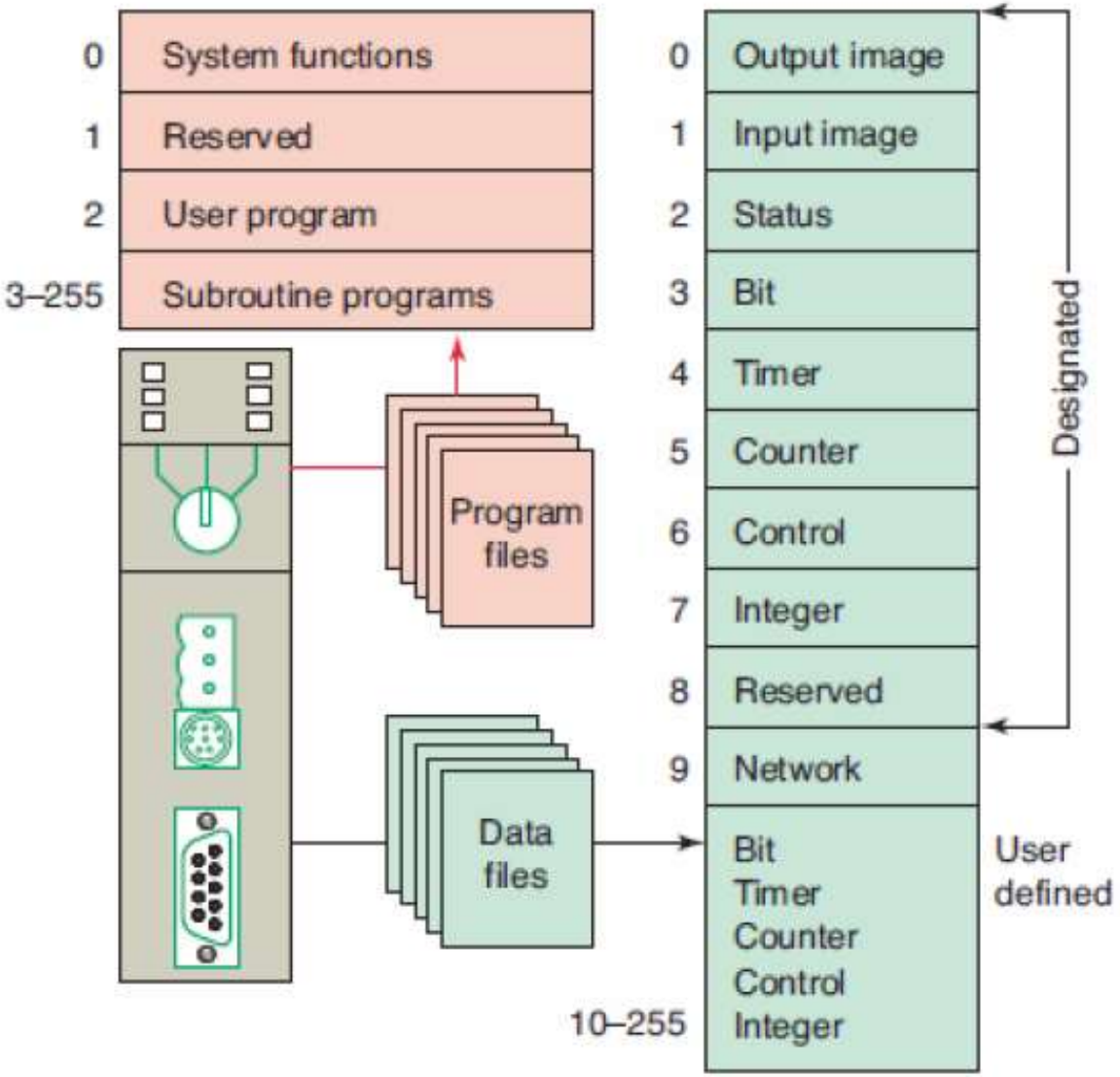

Input module
0 4 8 12
1 5 9 13
2 6 10 14
3 7 11 15
24 VDC input module
IN 0
IN 1
IN 2
IN 3
IN 4
IN 5
IN 6
IN 7
IN 8
IN 9
IN 10
IN 11
IN 12
IN 13
IN 14
IN 15
DC COM
DC COM
24 VDC
+
–
Field device power supply

PLC-DO

Signaling column

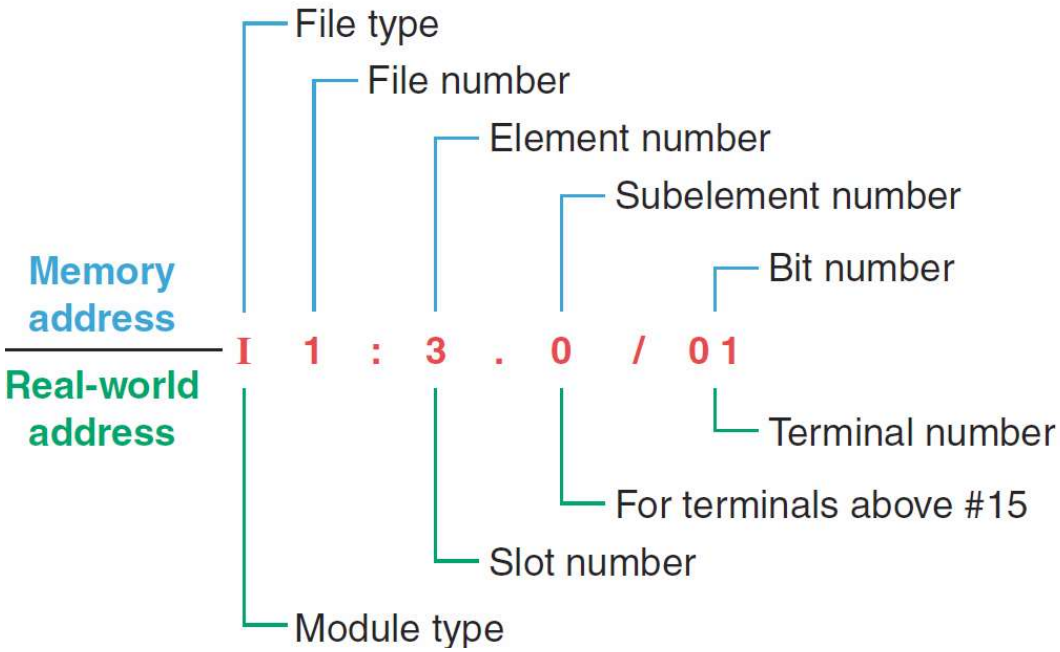


PLC Memory Space



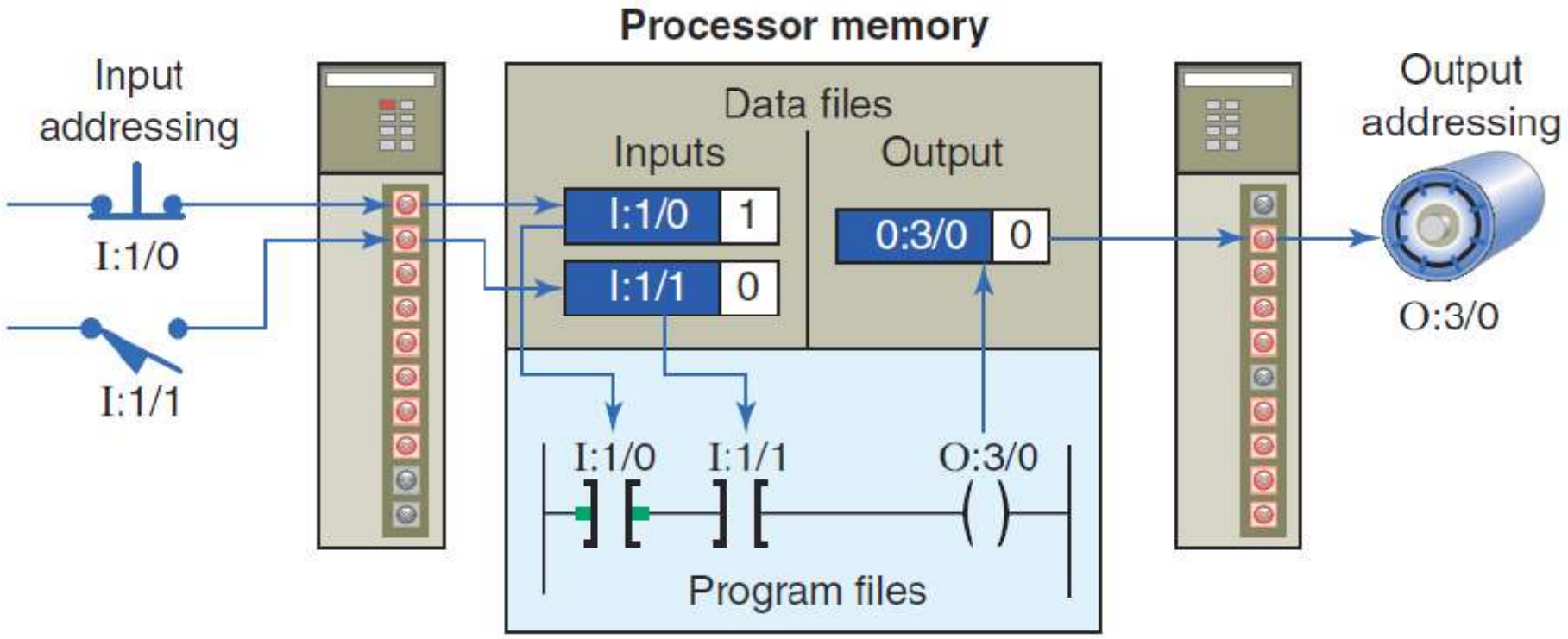
- Divided into
  - Program
  - Data

IO Addressing

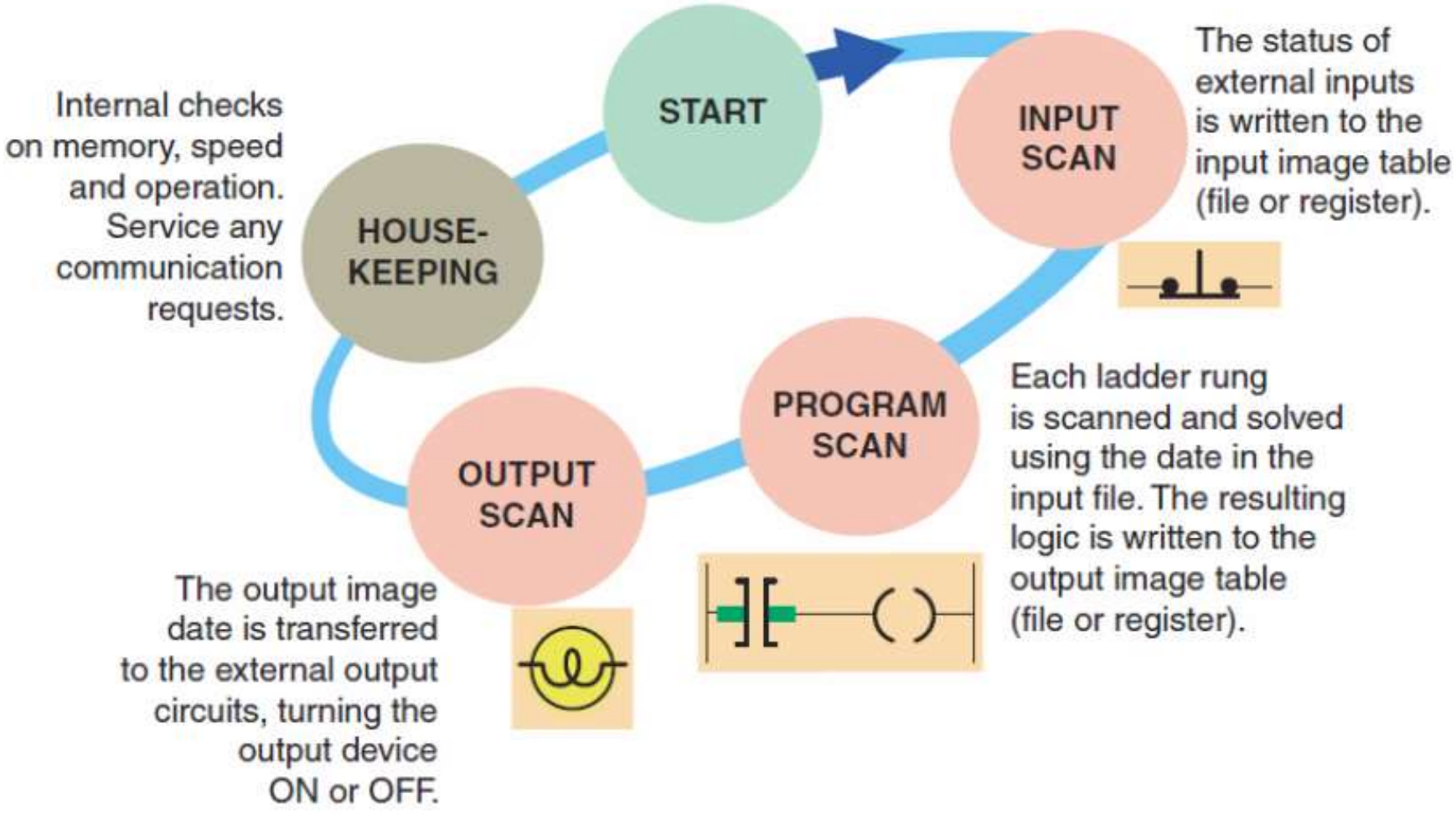


- Input Module: Slot 3 / Channel 1
- Some PLCs enable definition of an alias Tag Name that is more user understandable

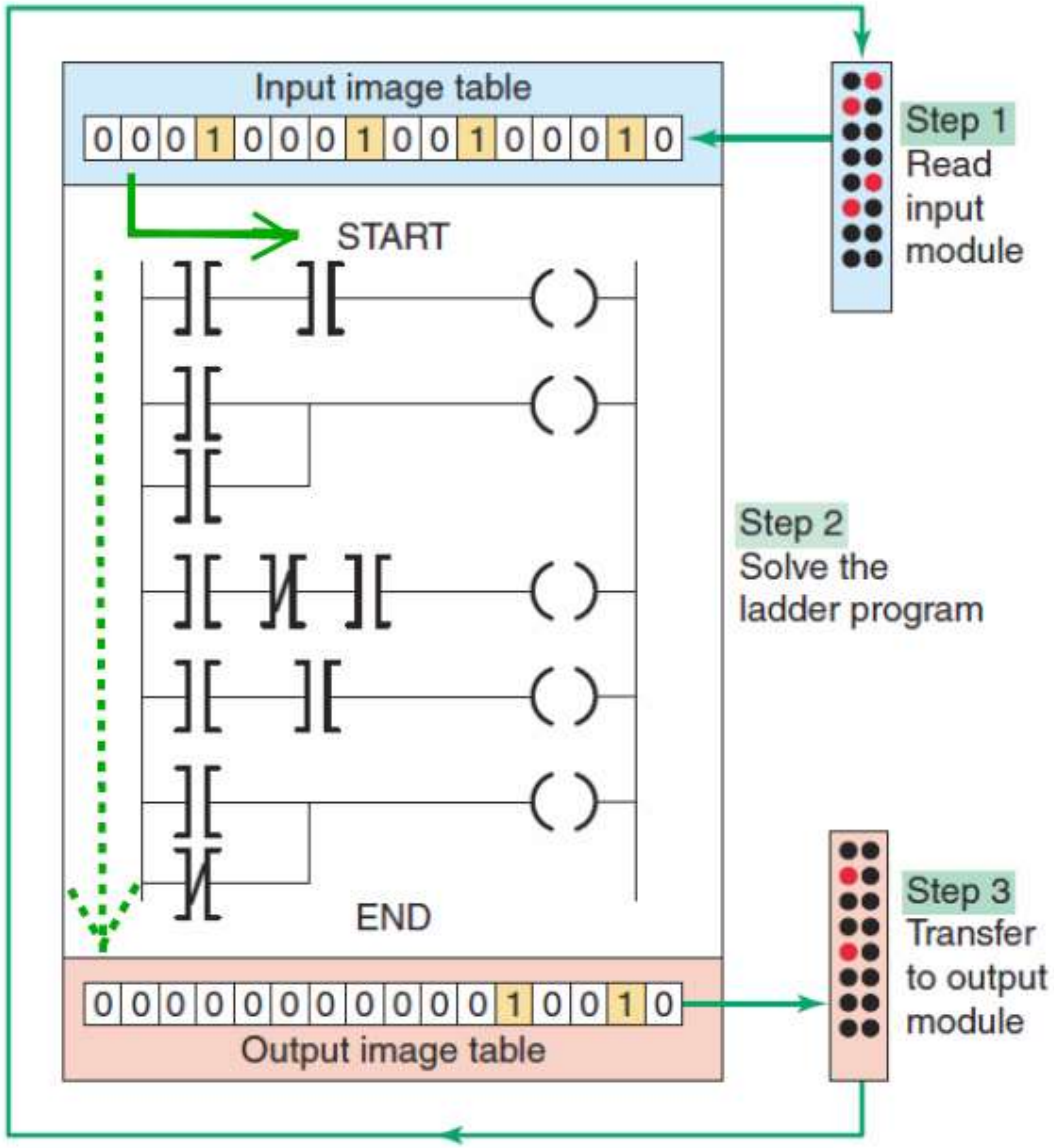
IO Address in LD Logic



PLC Program- Scan Cycle



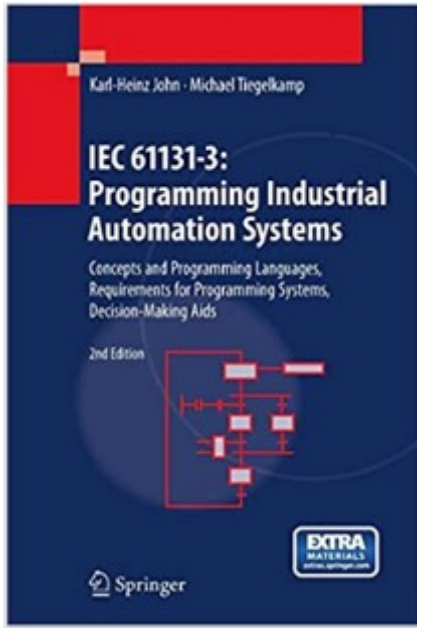
PLC Program - Scan Cycle

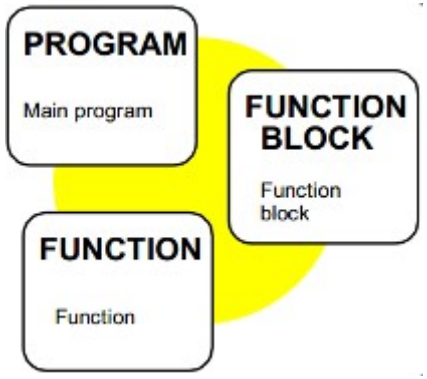


PLC Programming Languages

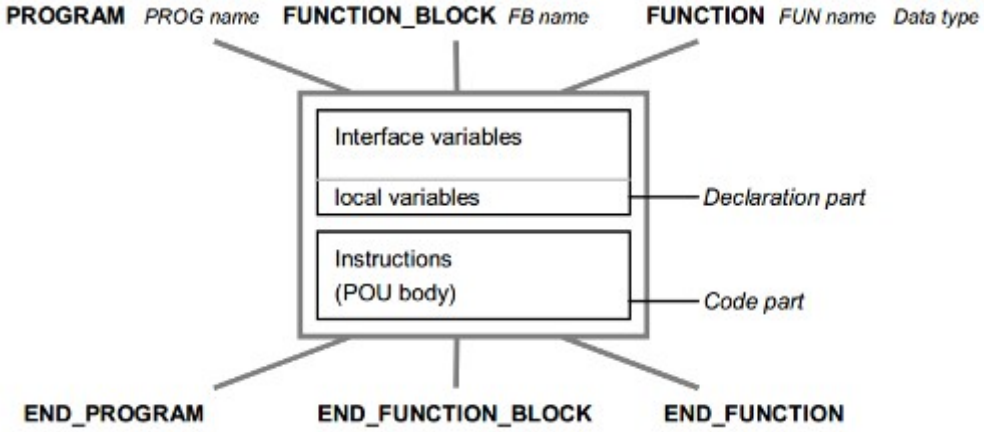
PLC Programming Languages

- IEC 61131-3 Standard specifies features, syntax and semantics that must be supported in PLC Program structure and programming languages
- Program is structured as Program Organization Units (POU)
  - Programs can be configured for
    - cyclic execution in configured order (Prog1, Prog5, Prog2,..)
    - Fixed Interval execution (Prog3 @ 100ms say)
    - Triggered Execution (Prog4 on DI1↑)
    - Can use Function Blocks and Functions
  - Function Blocks
    - Instantiated as “local to a POU” or Global and are called by Instance Name (Counter1 is instance of Counter Function Block)
    - Can use Functions
  - Functions
    - No retentive memory after return (Maths, Comparison … ADD, ISEQUAL…)
    - Can use other Functions
- Data
  - Global Variables are accessible by all Programs
  - Local Variables are accessible only in POU where they are declared





POUs in IEC 61131-3



PLC Programming Languages

IL: Instruction list - *deprecated*

```
LD      Var1
OR      Var2
AND     Var3  (* Comment *)
ST      Coil
```

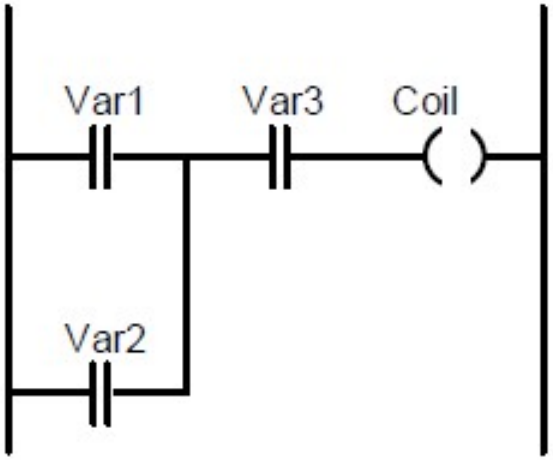
ST: Structured Text

```
Coil := (Var1 OR Var2) AND
            (* Comment *) Var3;
```

LD: Ladder diagram

0002

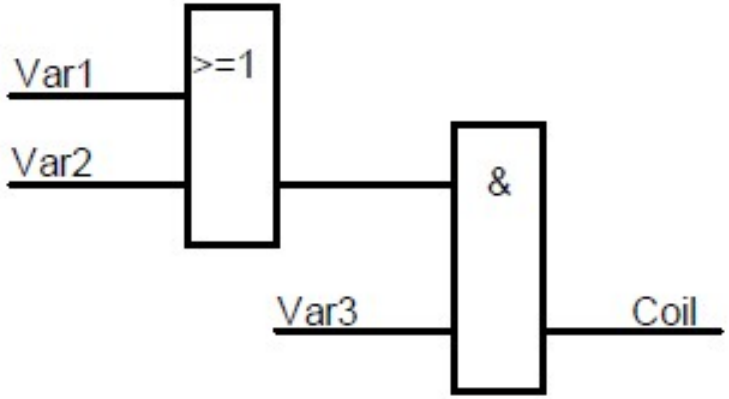
(* Comment *)



FBD: Function block Diagram

0002

(* Comment *)



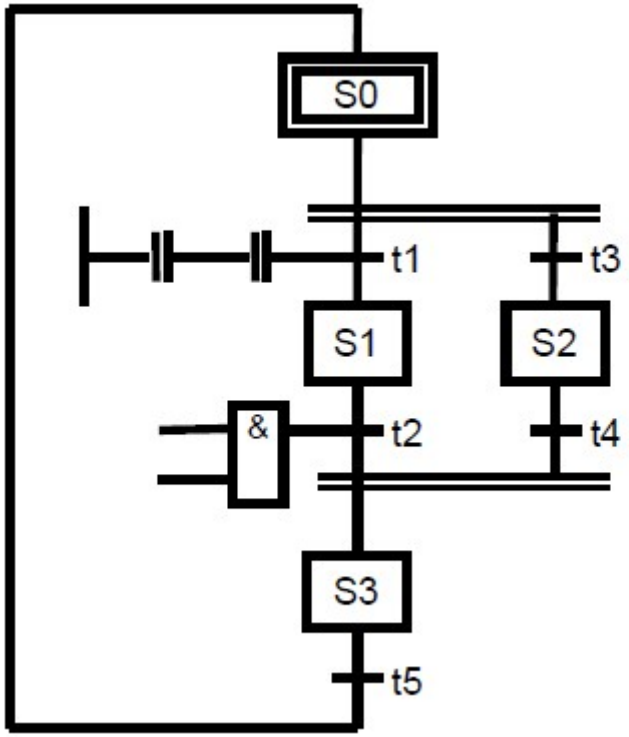
PLC Programming Languages – Some Comments

- **IL** similar to assembly
- **ST** similar to a higher level language like C.
  - **Adv** : Good for implementation for Indexing arrays, IF THEN, CASE, FOR, WHILE constructs and Maths operations

    Good for memory management & Communications – open and write to Serial port OR TCP socket
  - **Disadv** : No Visualization as in LD.
- **LD** is similar to an electrical circuit. It uses the same structure as the relay logic that existed before the invention of the PLC.
  - **Adv** - Can be understood by anyone with basic electrical knowledge. Software skills are not needed.
  - Easy to visualize interlocks, edge detection.
  - **Disadv** : No straight forward implementation for Indexing arrays, CASE, FOR, WHILE.
- **FBD** good to visualize signal flow and control loops AI1 went to that PI1 controller whose output went to that AO1

PLC Programming Languages

SFC: Sequential function chart



SFC used for sequential control system

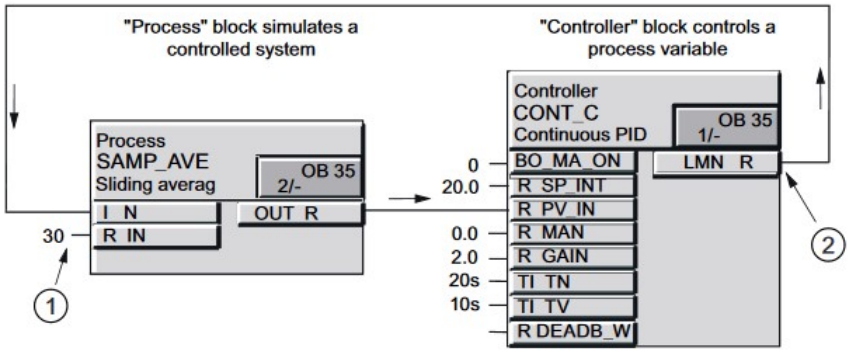
• A state is typically characterized by the set-points, logic and outputs.

• When input condition required for a transition is sensed the system moves to the next state

Example

- •Tank Filling State
- •Mixing State
- •Tank Emptying State

CFC: Continuous Function Chart



CFC :

- • No networks needed.
- • Loops can be created.
- • Order of execution of blocks can be specified.
- • Used in Process Industries, Motion Control, Data Streams

Writing Portable Applications with IEC61131-3

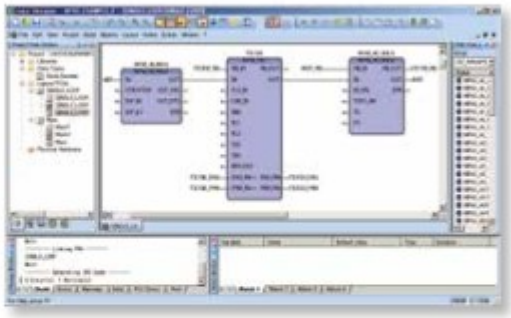
Network Based Control System STARDOM

Low Power Autonomous Controller FCN-RTU

YOKOGAWA

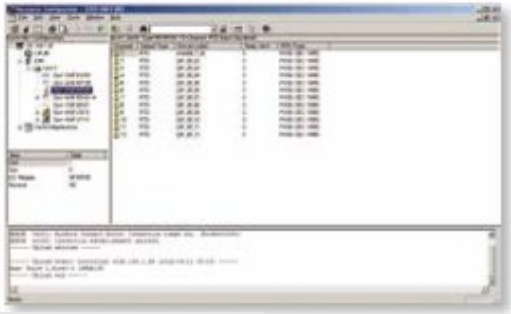
Enhanced Application Portability through Division of Logical and Physical Layers

A platform independent architecture enhances application portability.



Logic Designer: Control application development tool

- Supports all five IEC 61131-3 languages
- Intuitive look & feel with automated application layout
- Project comparison function for confirming modifications



Resource Configurator: Environment configuration tool for control applications

- Connects control application logical I/O with actual hardware I/O
- Configures hardware settings for IP addresses, serial ports, etc.

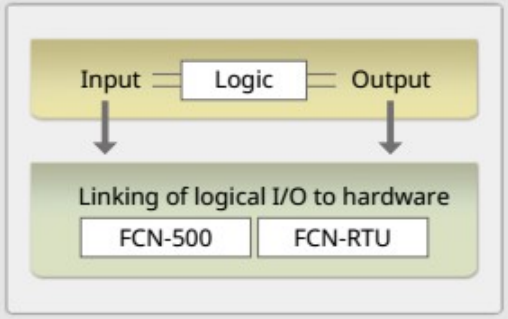
When Porting is required across PLC Families/Vendors use only Standard IEC61131-3 function blocks

Application portability by separating logic from hardware

►**Logic Designer**
- Platform independent programming tool

►**Resource Configurator**
- Hardware configuration tool for linking logical I/O with hardware

With Logic Designer, programming and debugging are platform independent, and with Resource Configurator logic can be easily ported to other hardware platforms.



IEC 61131-3 & PLCOpen

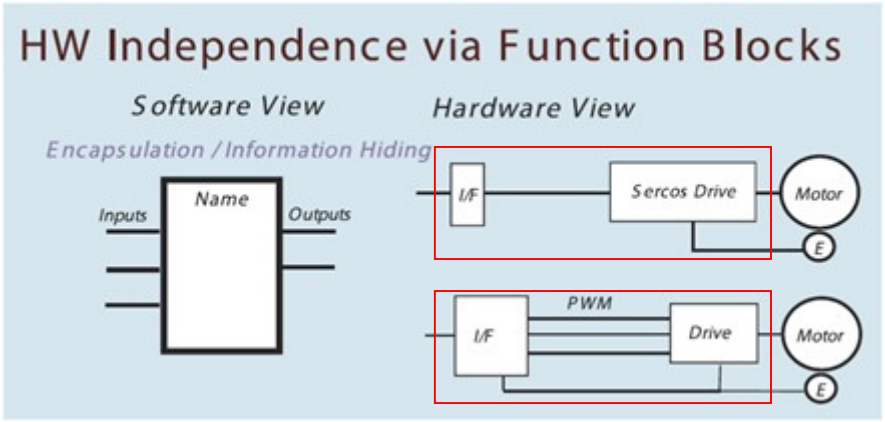
- PLCOpen had come up with standards that enables export/import of IEC 61131-3 compliant PLC programs and Function Blocks in XML format
- While most PLC Manufacturers support the IEC 61131-3 Programming Standard, differences in the underlying hardware and use of specialized Function Blocks means that programs ***may*** not be easily ported from the PLC of one manufacturer to that of another. On import to another manufacturer's editor, the program would require significant rework for hardware/Function Block specific customization
- The PLCOpen XML Exchange Standard has enabled development of 3rd party Function Blocks containing Logic for specific functions (eg. Motion/Robot Interface) that can be imported into the Editors of multiple PLC vendors (Siemens, Allen Bradley, Mitsubishi, Beckhoff..) and then configured to connect with the PLC specific IO and Communication registers. This enables code reuse across PLC systems from different manufacturers

PLCopen logic
ST LD IL
SFC FBD
IEC 61131-3

PLCopen XML
EXTENSIBLE <MARKUP> LANGUAGE

PLCopen motion control

PLCopen
ERP MES HMI
IEC 61131-3
C T R L
OPC UA



Soft PLCs

IPC based PLCs

- Panel PCs with Built-in Screen
- Rail Mounted
- Rack Mounted

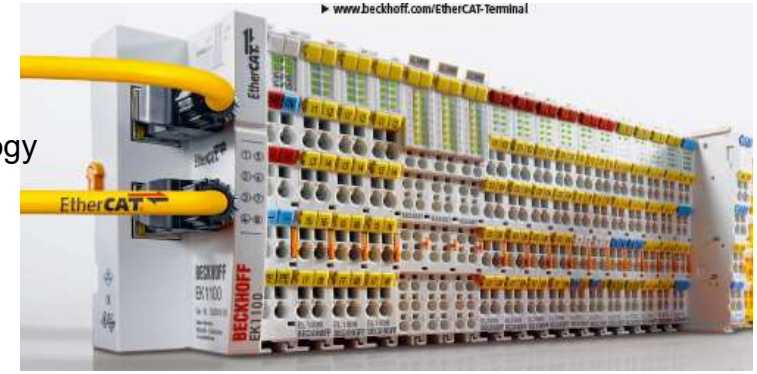
BECKHOFF

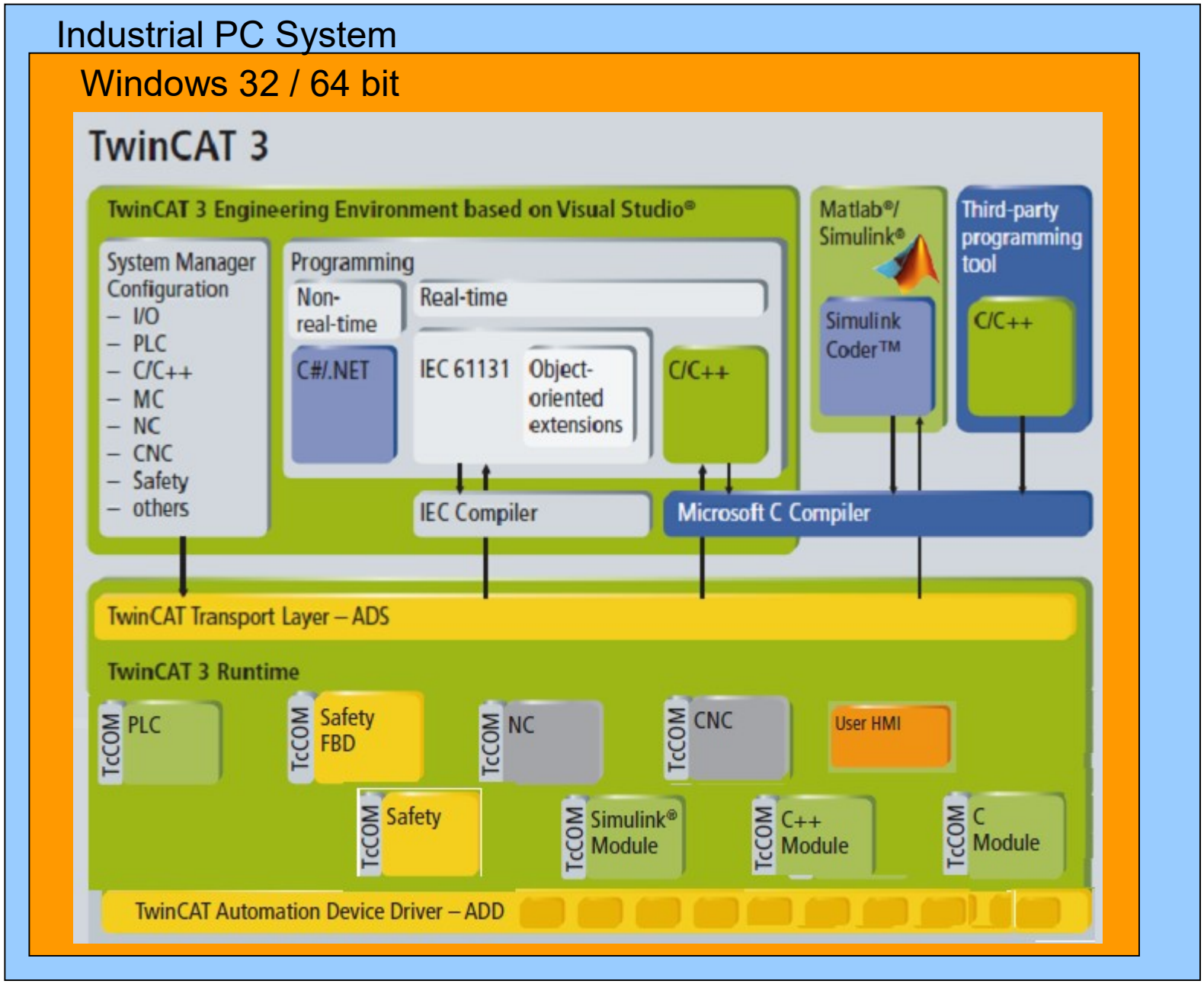
TwinCAT®

IPC based PLC system connected to IOs on high speed real time Ethernet Fieldbus

EtherCAT®

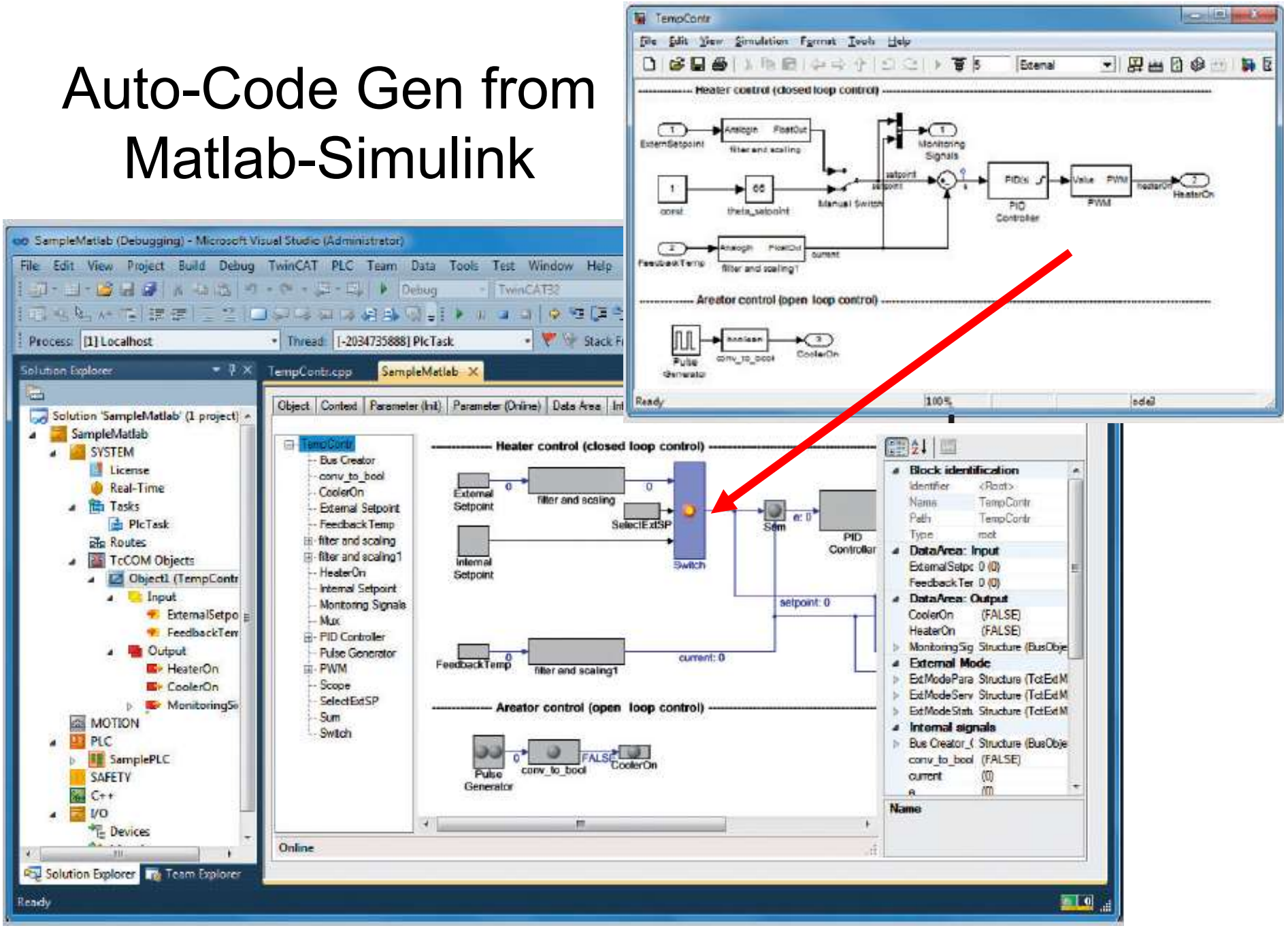
CAT: Control Automation Technology





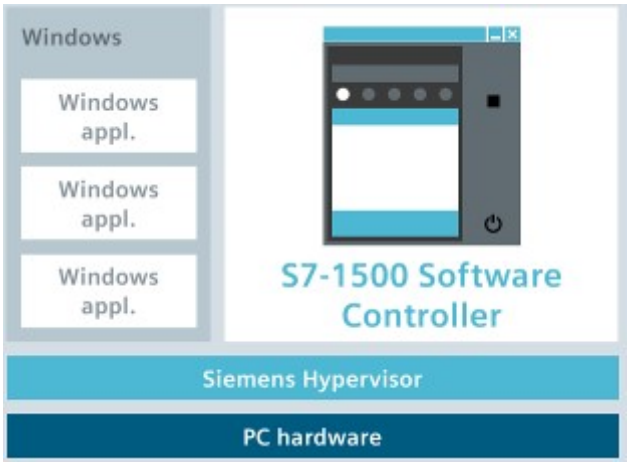
On Multi-core machines each of the Runtime Processes can be assigned to a separate core
Visual Studio based development environment. Windows based system.

Auto-Code Gen from Matlab-Simulink



Typically used for process where Fuzzy Logic Control or some complex algorithm is needed – Supported by the TwinCat PC based PLC System

Siemens Soft PLC – **S7-1500 Software Controller**



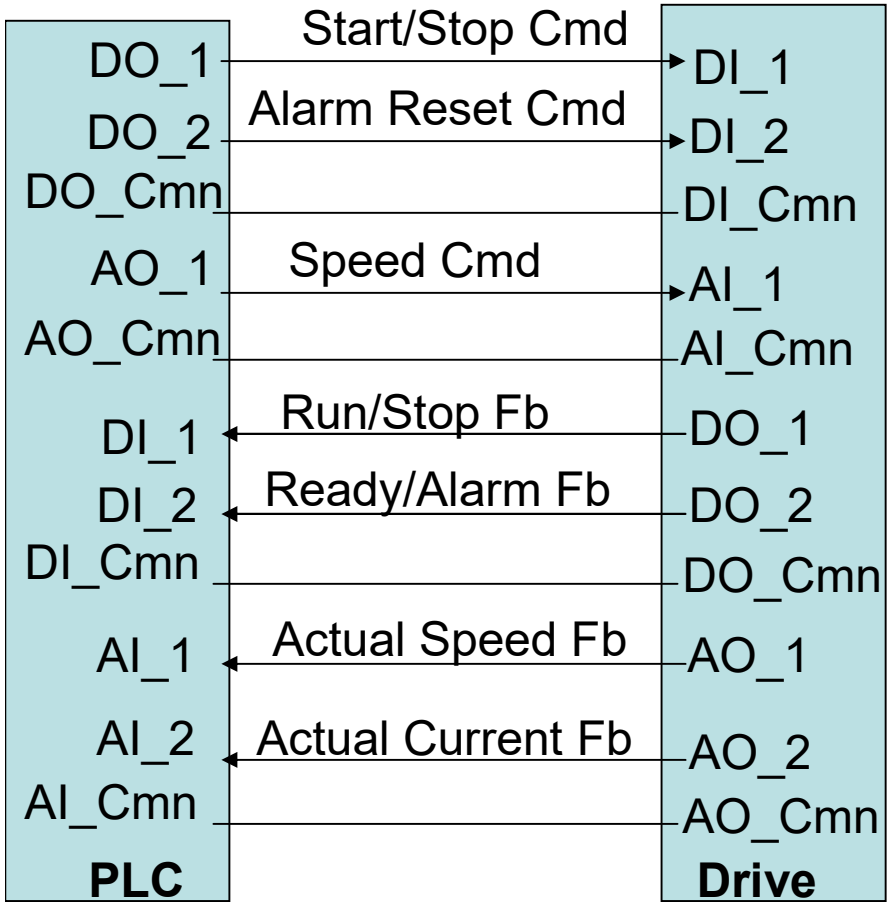
- Industrial PC formats like Panel PC, Rack-mounted PC, **Hybrid-PC with backplate for ET200 modules (effectively making it a PAS – see later slides)**
- Siemens hypervisor allocates the PC resources such as interfaces, processor cores or memory directly and exclusively to the OS and to the **Software Controller**. Even an OS crash or restart has no effect whatsoever on the execution of the control program
- Communication between the OS and the Software Controller
  - TCP/IP Communication using S7 blocks
  - OPC UA Server for the Software Controller
  - ODK1500S Open Development Kit for real-time programming in C/C++
- Communication with Remote I/O and Field Devices over Profibus, Profinet
- Applications : Operation of PLC, HMI, image processing and other software on a single hardware platform, linking to databases, complex algorithms, Model-based development with Simulink..

PLCs and Drives

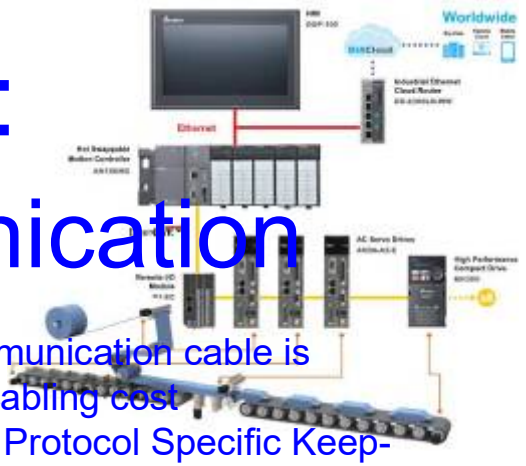
PLC Control of Drive using :

1. I/O

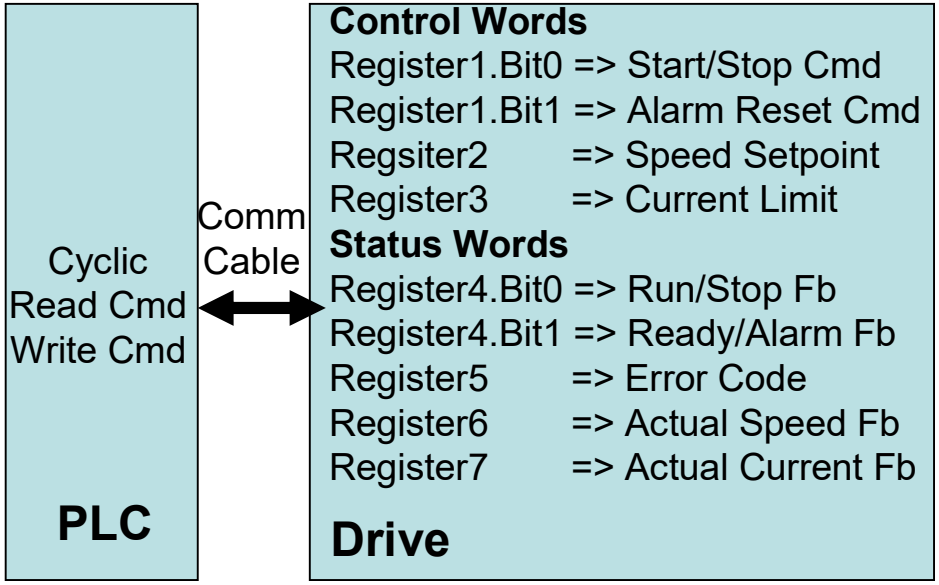
- Expensive Analog I/O required
- 11 wires required per Drive



2. Communication



- Only single multi-drop communication cable is required – greatly reducing cabling cost
- Cable break is detected by Protocol Specific Keep-Alives OR Pre-set Cyclic Read/Write Cmd intervals – if R/W Cmd is not received every say 1s then communication break is assumed and Drive goes to configured safe state
- Large number of Cmd/Status Signals can be R/W
- Drive can be remote configured using the communication cable

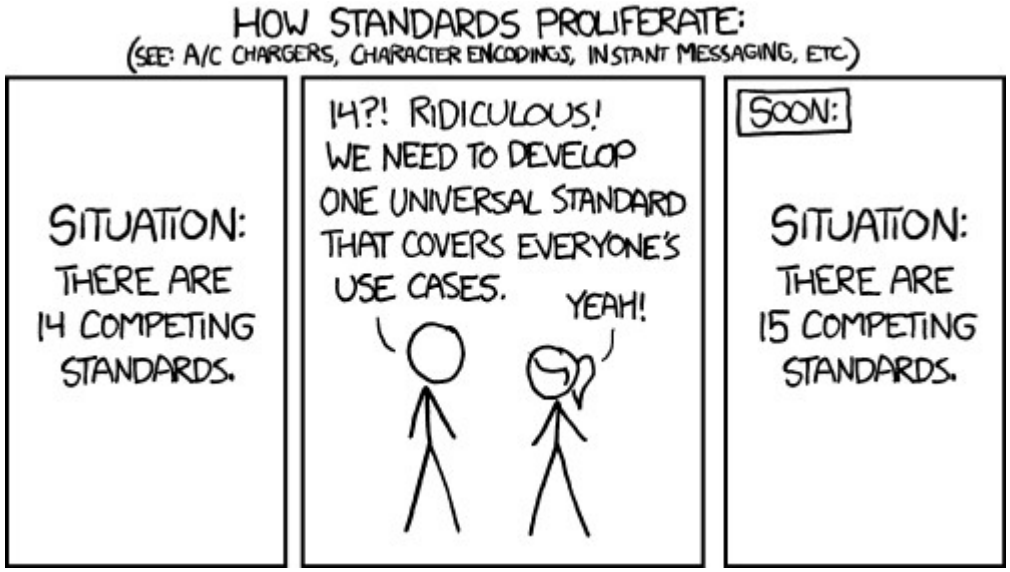


Industrial Protocols

Industrial Protocols

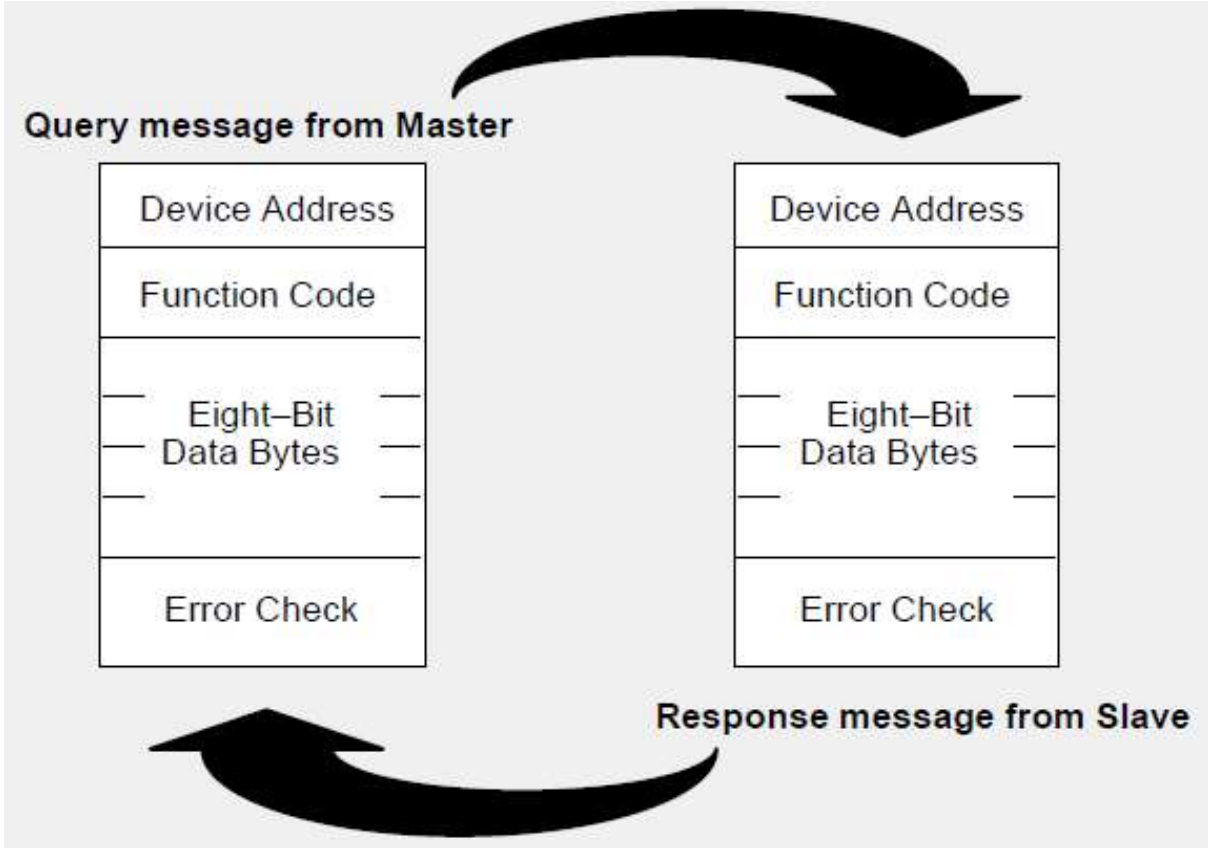
- Protocol used depends on the support by Plant Instruments and PLC

- **IEE802.1 Time Sensitive Network (TSN)** OSI Level-2 Specification for deterministic Ethernet Communications.
- Supported Network Infrastructure (switches / routers..) must be used for TSN.
- Same network can be used for IT and OT


HOW STANDARDS PROLIFERATE:
(SEE: A/C CHARGERS, CHARACTER ENCODINGS, INSTANT MESSAGING, ETC.)
SITUATION: THERE ARE 14 COMPETING STANDARDS.
14?! RIDICULOUS! WE NEED TO DEVELOP ONE UNIVERSAL STANDARD THAT COVERS EVERYONE'S USE CASES.
YEAH!
SOON:
SITUATION: THERE ARE 15 COMPETING STANDARDS.


HOW DID THE INDUSTRY STANDARDS MEETING GO?
DID YOU CONVINCE 83 COMPANIES TO ADOPT STANDARDS THAT BENEFIT ONLY US WHILE DOOMING THE ENTIRE INDUSTRY IN THE LONG RUN?
OR ARE YOU A COMPLETE FAILURE?
CAN I HEAR THOSE CHOICES AGAIN?
DilbertCartoonist@gmail.com
Dilbert.com
9-2-09
©2009 Scott Adams, Inc./Dist. by UFS, Inc.

PLC Communications- Modbus

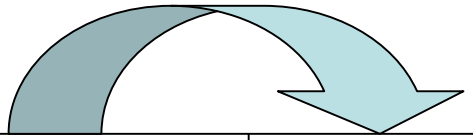


Code	Name
01	Read Coil Status
02	Read Input Status
03	Read Holding Registers
04	Read Input Registers

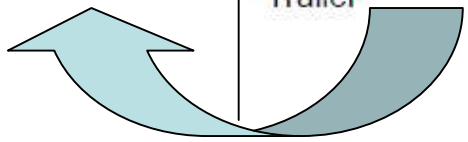
START	ADDRESS	FUNCTION	DATA	LRC CHECK	END
1 CHAR :	2 CHARS	2 CHARS	*n* CHARS	2 CHARS	2 CHARS CRLF

Modbus ASCII Messages

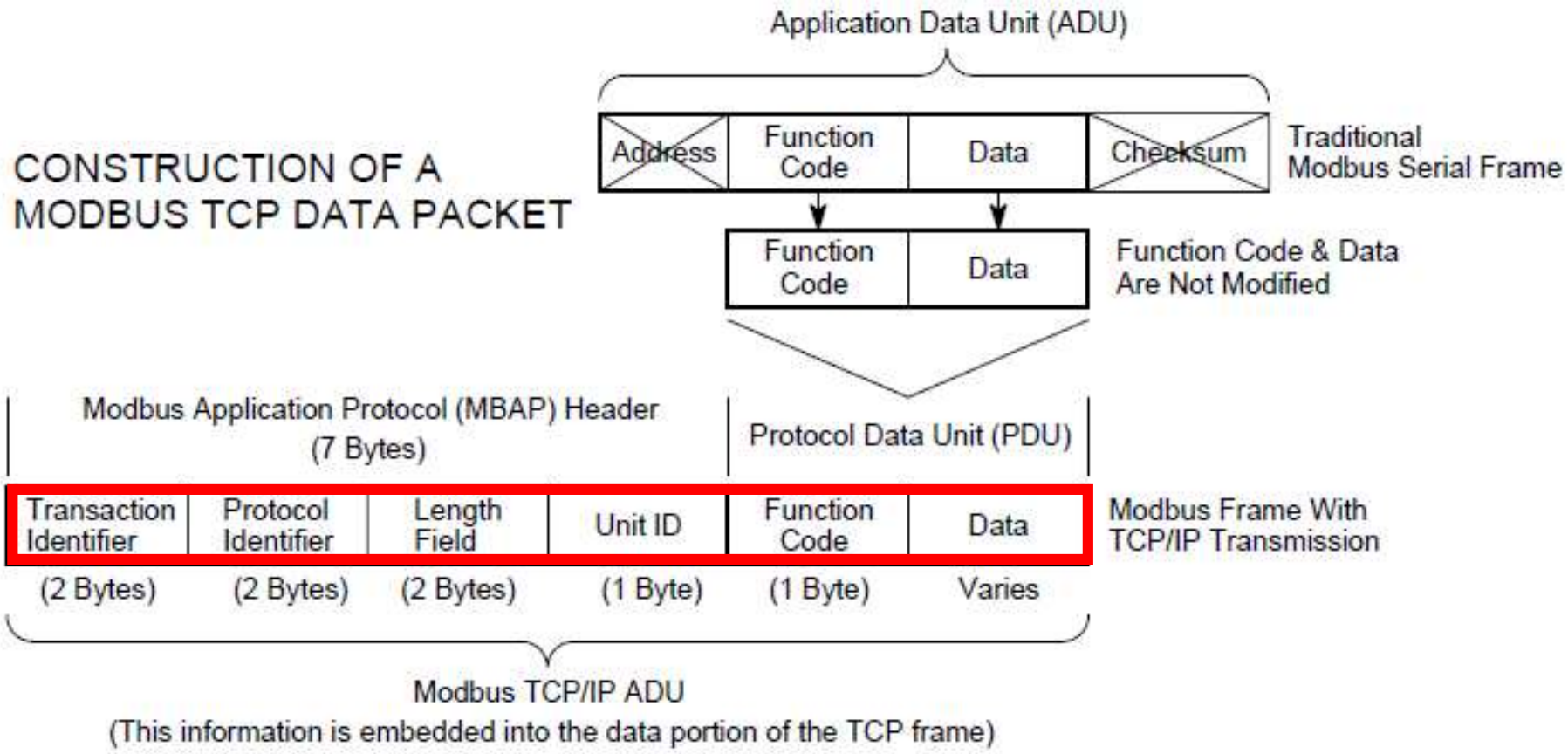
- Master query to Slave at device address 06
- Requests data from 40108 through 40110
  - Start Address 0107 (006B hex)
  - Number of registers 03
- Slave response 40108(02 2B) 40109(00 00) 40110 (00 063)



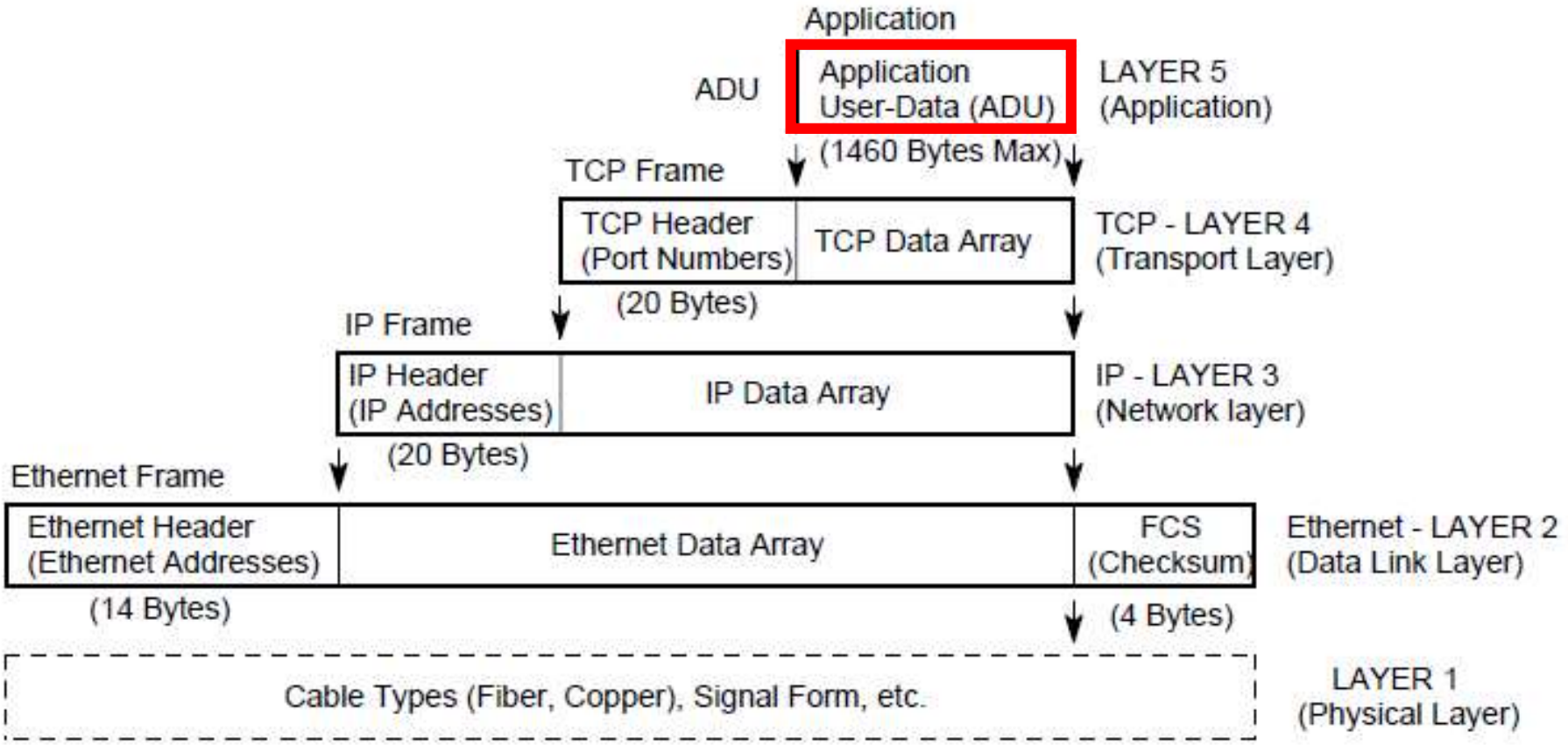
QUERY			RESPONSE		
Field Name	Example (Hex)	ASCII Characters	Field Name	Example (Hex)	ASCII Characters
Header		: (colon)	Header		: (colon)
Slave Address	06	0 6	Slave Address	06	0 6
Function	03	0 3	Function	03	0 3
Starting Address Hi	00	0 0	Byte Count	06	0 6
Starting Address Lo	6B	6 B	Data Hi	02	0 2
No. of Registers Hi	00	0 0	Data Lo	2B	2 B
No. of Registers Lo	03	0 3	Data Hi	00	0 0
Error Check		LRC (2 chars.)	Data Lo	00	0 0
Trailer		CR LF	Data Hi	00	0 0
			Data Lo	63	6 3
	Total Bytes:	17	Error Check		LRC (2 chars.)
			Trailer		CR LF
				Total Bytes:	23



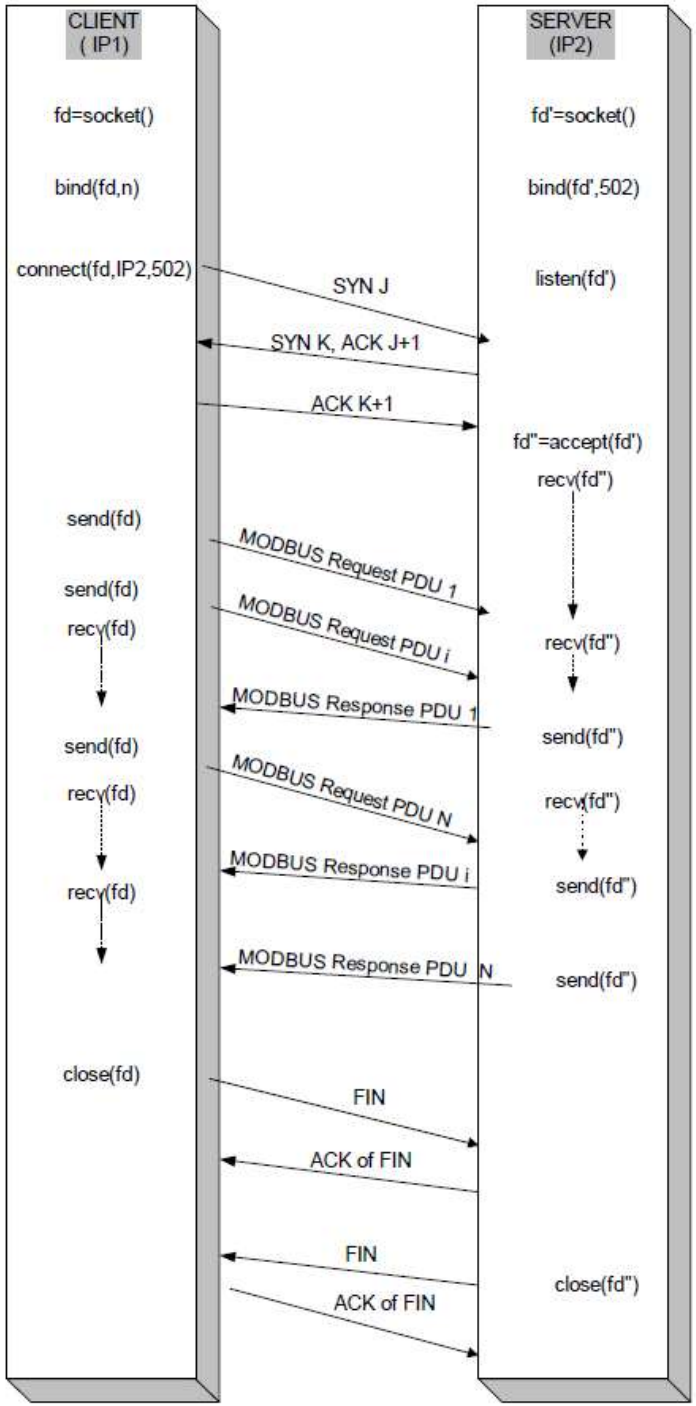
Modbus TCP


CONSTRUCTION OF A
MODBUS TCP DATA PACKET
Application Data Unit (ADU)
Address
Function Code
Data
Checksum
Traditional Modbus Serial Frame
Function Code
Data
Function Code & Data Are Not Modified
Modbus Application Protocol (MBAP) Header
(7 Bytes)
Protocol Data Unit (PDU)
Transaction Identifier
Protocol Identifier
Length Field
Unit ID
Function Code
Data
Modbus Frame With TCP/IP Transmission
(2 Bytes)
(2 Bytes)
(2 Bytes)
(1 Byte)
(1 Byte)
Varies
Modbus TCP/IP ADU
(This information is embedded into the data portion of the TCP frame)

Modbus TCP Message Structure



Modbus-TCP Exchanges



Modbus Ethernet Port 502

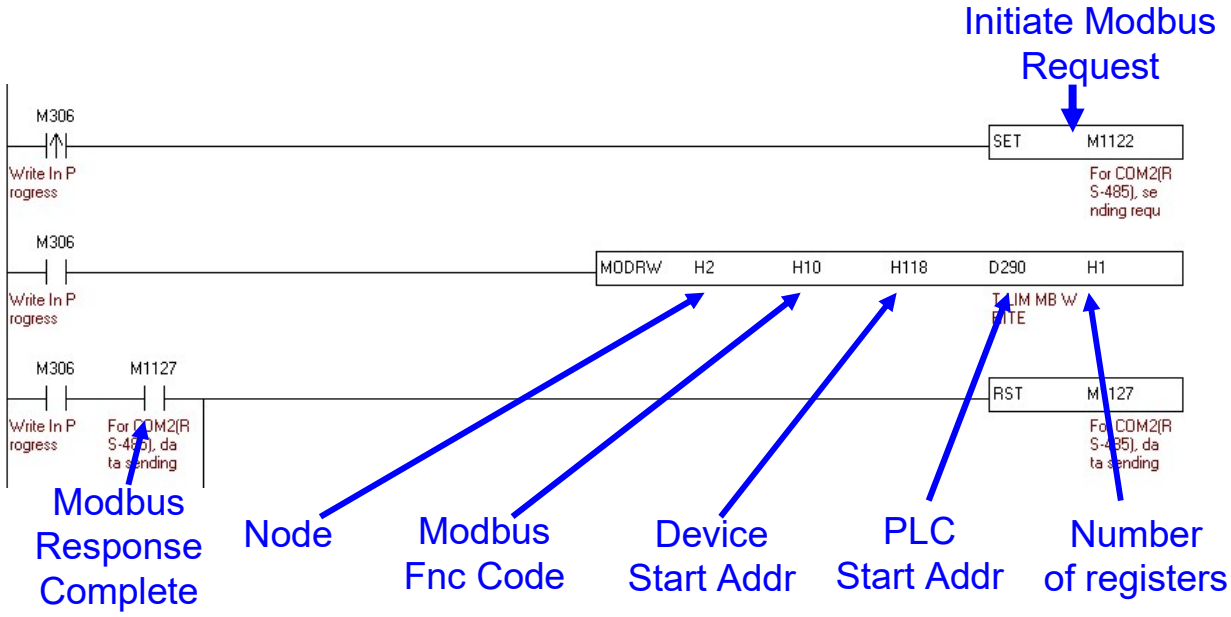
PLC as Modbus - Master & Slave

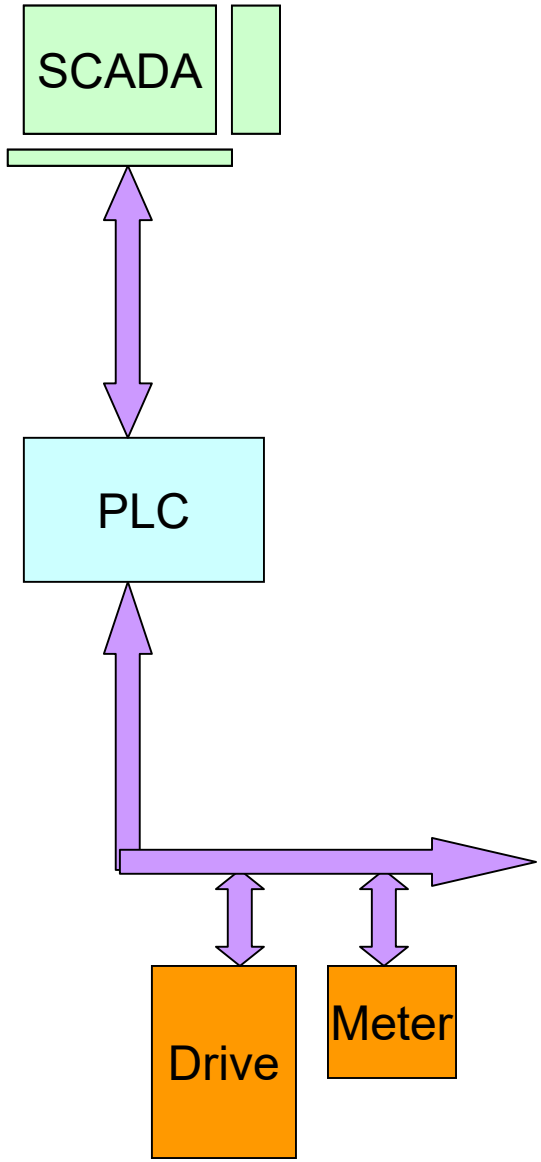
- SCADA is Modbus Master and **PLC is Modbus Slave**
- PLC registers are mapped to Modbus Registers

PLC Device Address

Device	Range	Effective Range			MODBUS Address	Address
		ES2/EX2	SS2	SA2/SX2		
X	000~377 (Octal)	000~377	000~377		101025~101280	0400~04FF
Y	000~377 (Octal)	000~377	000~377		001281~001536	0500~05FF
D	000~255				404097~405376	1000~10FF
D	256~511					1100~11FF
D	512~767					1200~12FF
D	768~1023					1300~13FF
D	1024~1279					1400~14FF

- **PLC is Modbus Master** and Field-Device is Modbus Slave
- PLC accesses Field-Device registers using Modbus Functions





Common Interface for Standard Peripherals

Taking the Example of an AC Drive

- Analog and Digital IO Interface is fairly Drive Vendor Independent
- Modbus Interface may have different Addresses for different Vendors, requiring rework with a different Drive Vendor to re-map to the new Addresses
  - Alternative is for all Drive Vendors to agree on standard Modbus Addresses for Control and Status Regsiters

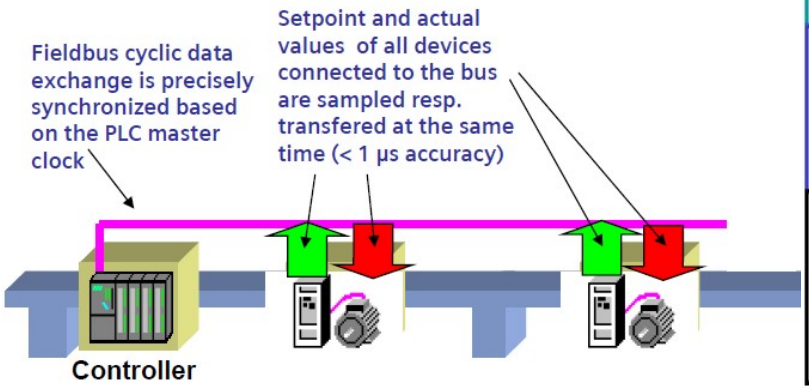
- Profinet (at Physical & Network Layer) ensures that Devices are ***Interconnectable & Interoperable***
- ProfiDrive is an Application Layer Specification and ensures that devices are ***Interchangable***

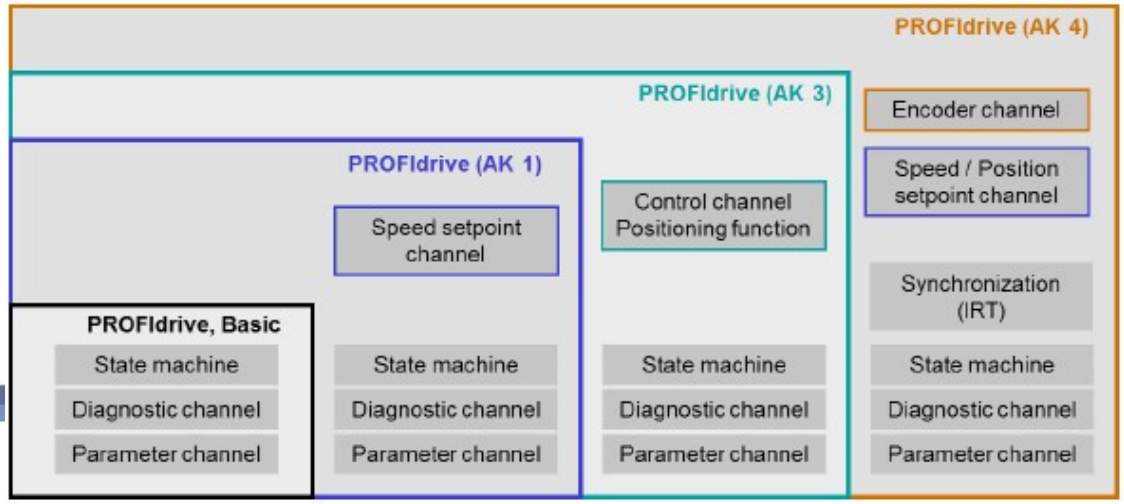
– PROFIdrive is a vendor-neutral application profile from PROFIBUS and PROFINET International (PI) which is focused on drives, encoders, motors, and their applications, which range from simple to very demanding motion control tasks. PROFINET allows scalable update times starting from 31.25µs and a jitter of less than 1µs for the most demanding motion applications while supporting open TCP communications. PROFIdrive provides interoperability through standardized drive IO data sets, messages, and parameter sets as well as manufacturer-specific options to allow vendors to implement additional features to foster competition and innovation.

... Isochronous mode assures a bus cycle with 1 µs accuracy

All drive application processes are synchronized to the bus cycle ...

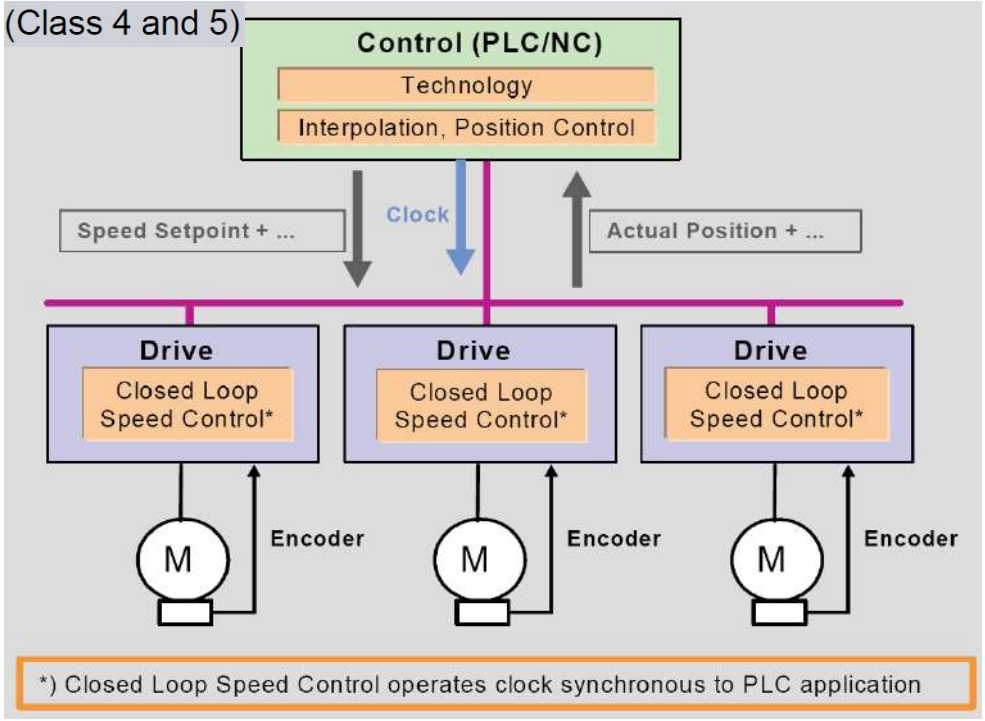
... allowing high performance and high precision motion control applications!





Standard telegram for Central motion control

(Class 4 and 5)



Telegram 1 16-bit speed setpoint

PZD01	PZD02
STW1	NSOLL_A
ZSW1	NIST_A

Receive user data

Send user data

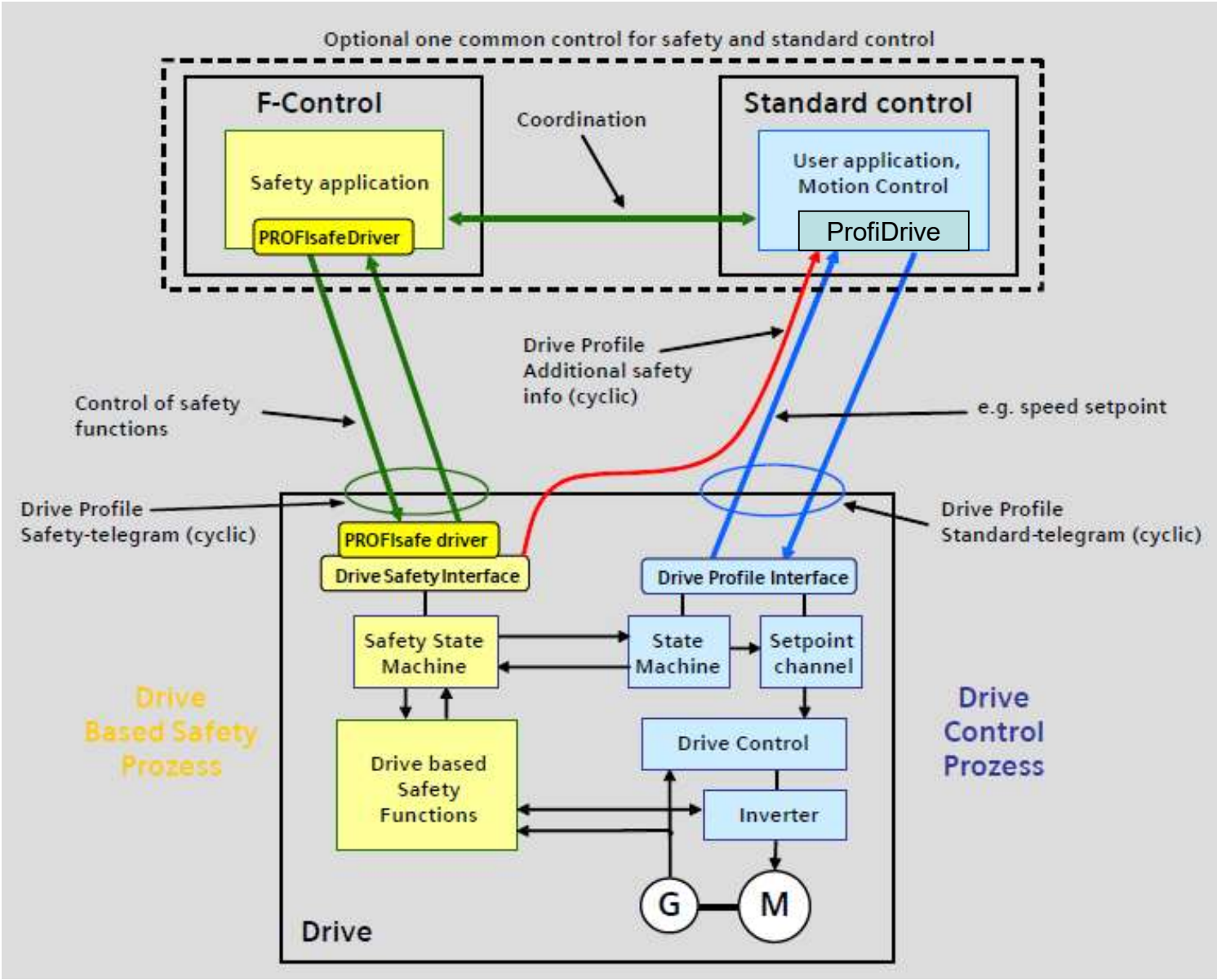
Telegram 2 32-bit speed setpoint

PZD01	PZD02	PZD03	PZD04
STW1	NSOLL_B		STW2
ZSW1	NIST_B		ZSW2

Telegram 3 32-bit speed setpoint with 1 position encoder

PZD01	PZD02	PZD03	PZD04	PZD05	PZD06	PZD07	PZD08	PZD09
STW1	NSOLL_B		STW2	G1_STW				
ZSW1	NIST_B		ZSW2	G1_ZSW	G1_XIST1		G1_XIST2	

Profisafe & ProfiDrive

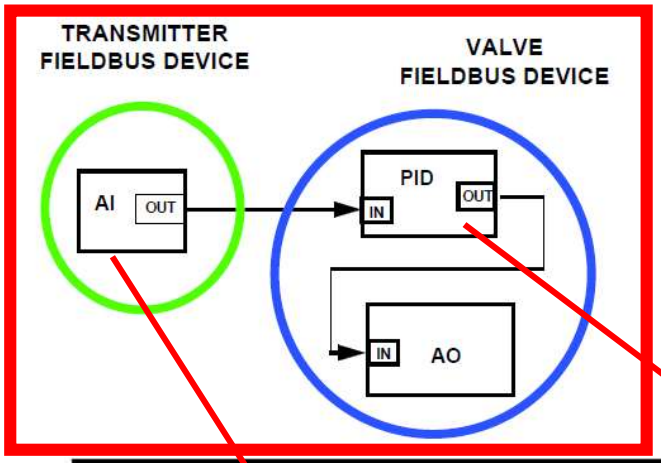


Set-up Profinet Communication between Drive(Device) and PLC(Controller)

- Profinet Devices Manufacturers provide associated GSD*ML* (General Station Description - *XML*) files
- Import the GSD*ML* file into a Standard Profinet Configuration Tool *(typically your Controller Programming Environment)*, the Device Type now shows in the catalog
- For Manual Configuration
  - Drag and Drop the Device from Catalog into the Configuration View to create a Device Instance.
- For Auto Configuration
  - Scan the network to automatically add Device Instances for all connected devices
- Configure the Device Instance with Parameters like – Names, Network IP Address, Add Device Options like for Drive Devices - Power Card, Control Card, Encoders, Default Fail-safe IO State in case of loss of Communication
- Configure the Profinet Telegrams, map to PLC Registers. Write the PLC program to Read Status/Write Setpoints to the mapped PLC Registers
- Download the configuration to the Field Devices(Drive) & Controller(PLC)

- Foundation Fieldbus **- Example of Distributed Control**
- Function blocks can be built into Fieldbus Devices as needed to achieve the desired device functionality.
  - For example,
    - a simple temperature transmitter may contain an AI function block.
    - A control valve might contain a PID function block as well as the expected AO block.
- Thus, a complete control loop can be built using only a simple transmitter and a control valve
- Function Blocks (FB) provide the control system behavior. The input and output parameters of Function Blocks can be linked over the Fieldbus.
- The execution of each Function Block is precisely scheduled. There can be many function blocks in a single User Application.
- Function Block Name Symbol
  - Analog Input AI
  - Analog Output AO
  - Bias/Gain BG
  - Control Selector CS
  - Discrete Input DI
  - Discrete Output DO
  - Manual Loader ML
  - Proportional/Derivative PD
  - Proportional/Integral/Derivative PID
  - Ratio RA

Distributed Control Configuration



As long as the two configured devices are powered and online (communicating with each other on the Fieldbus Network), the control loop will continue to execute – irrespective of the status of the rest of the plant.

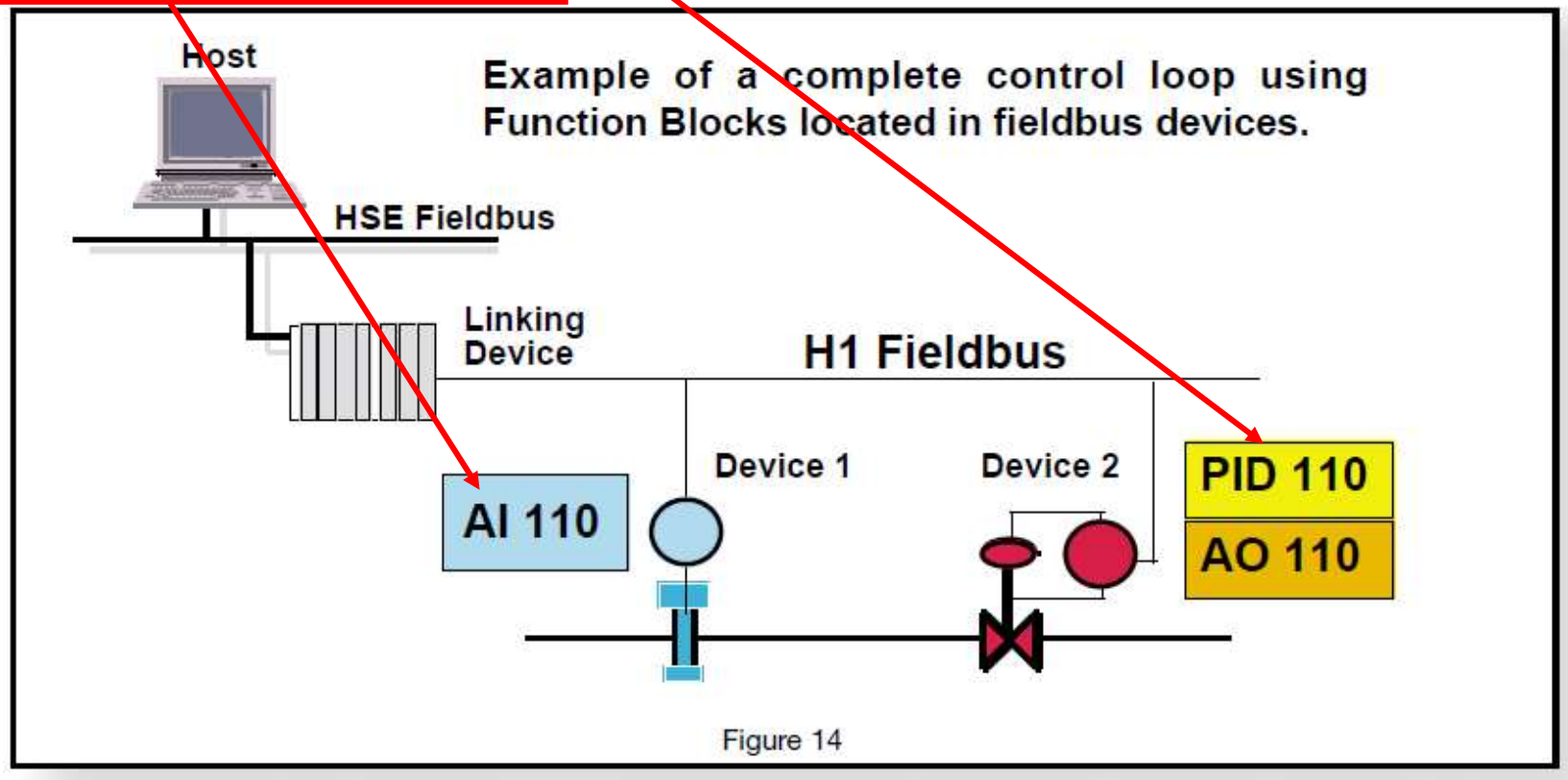
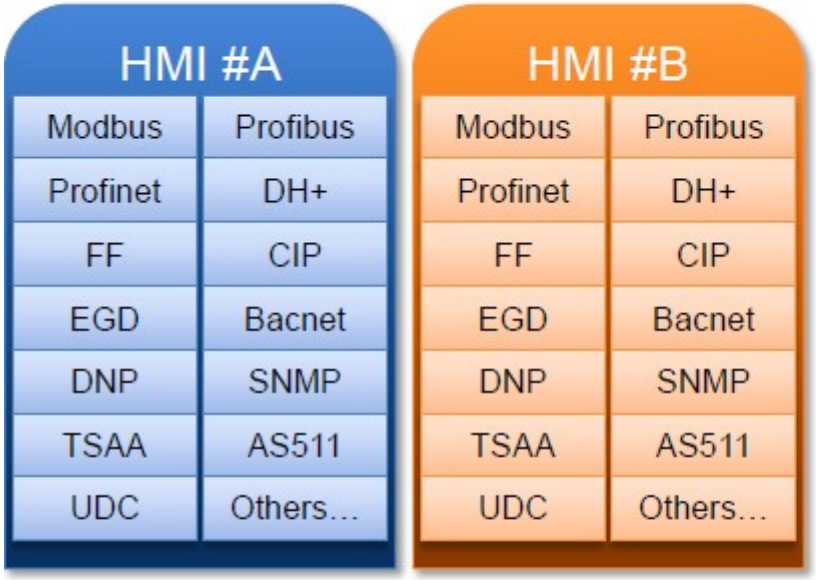
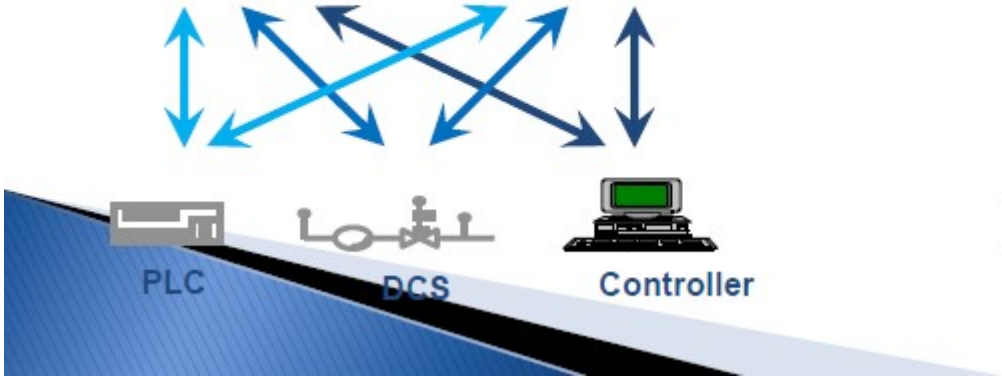


Figure 14

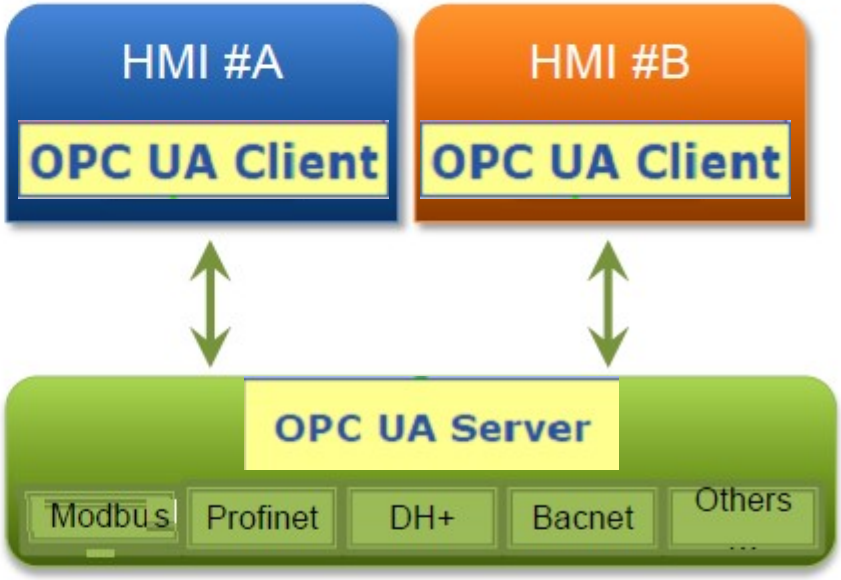
PLC-SCADA Connectivity

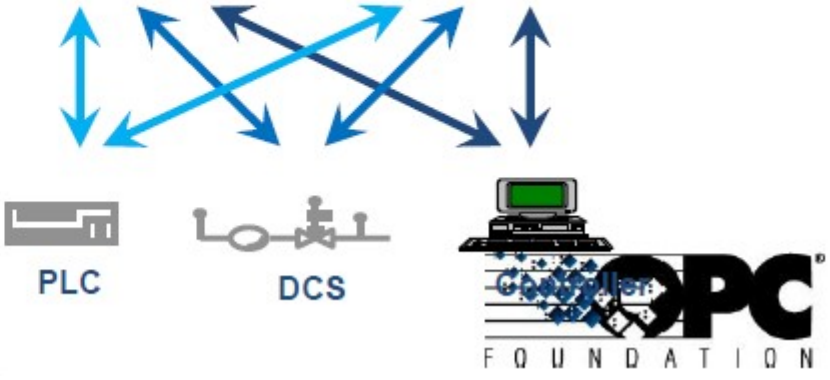
- Before OPC



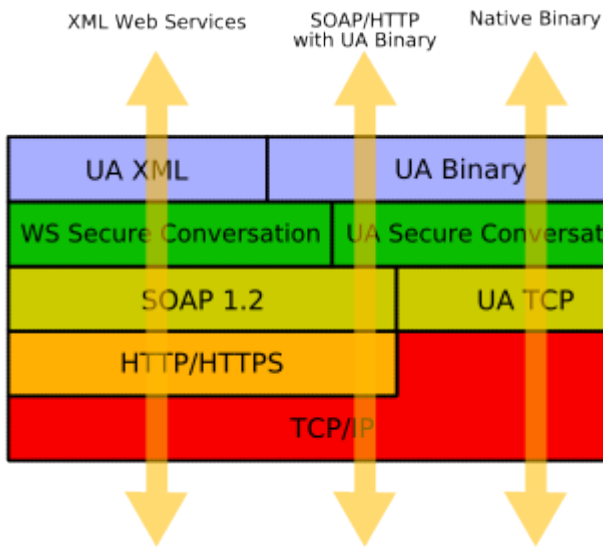


- With OPC

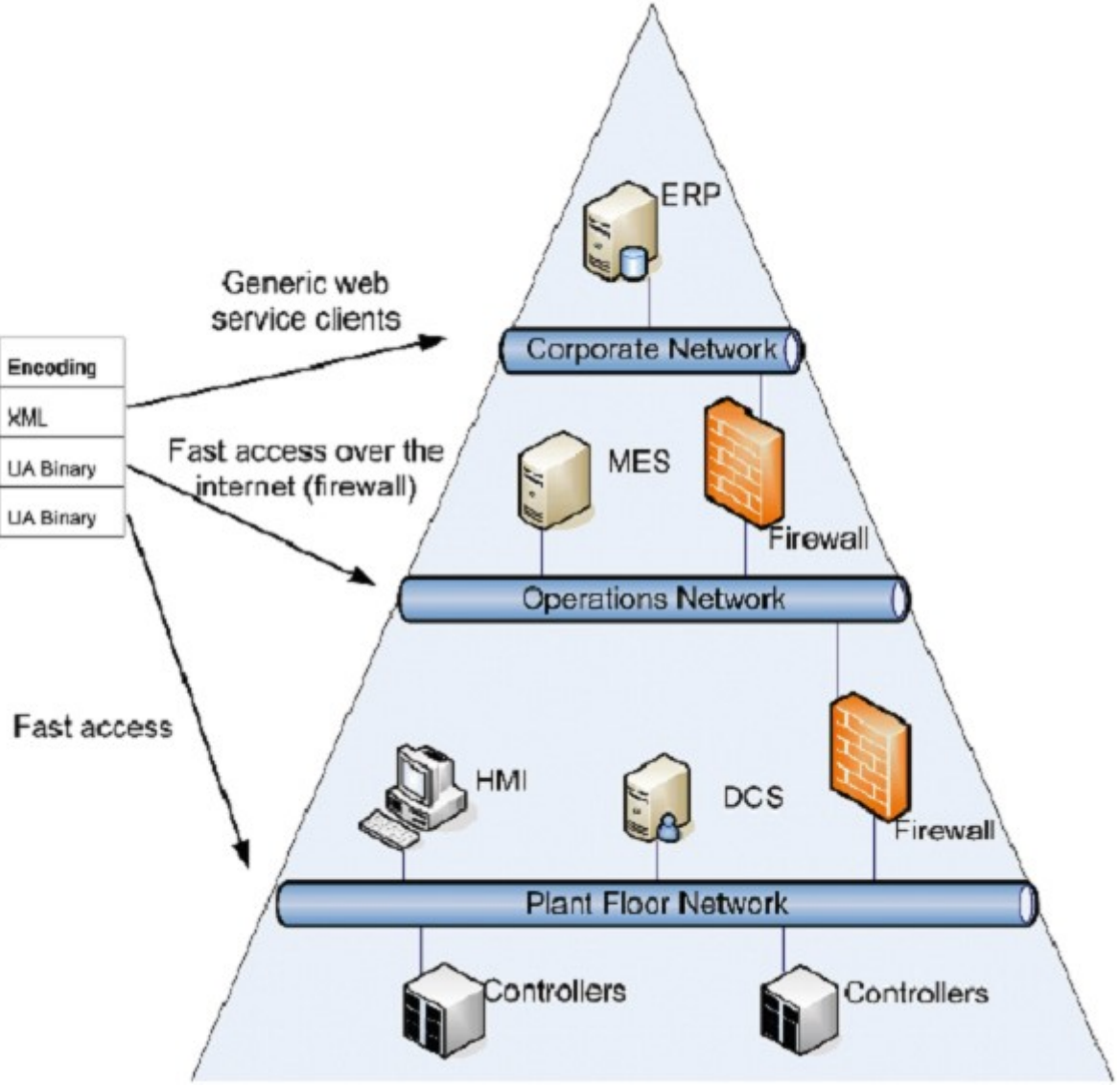




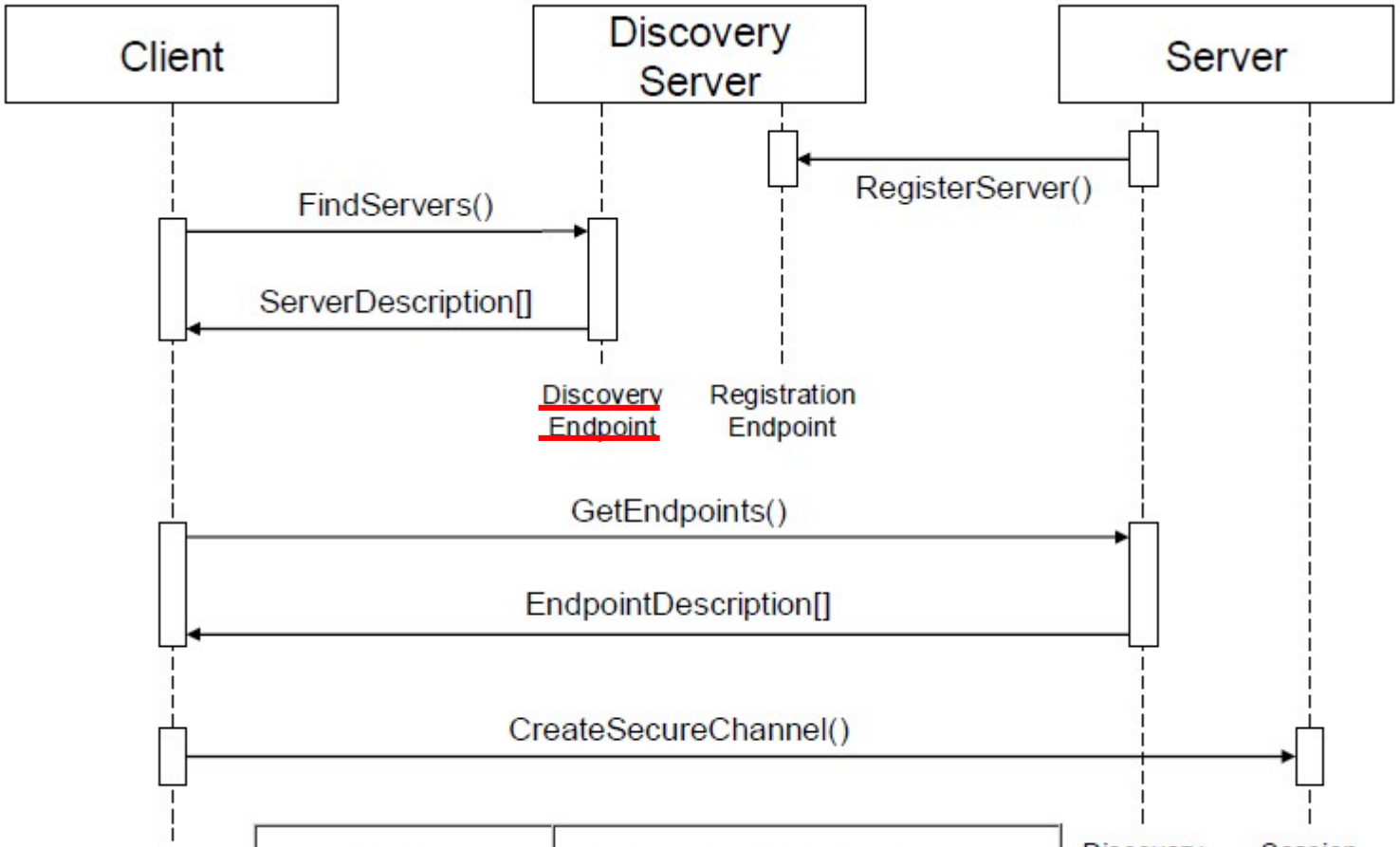
OPC UA Protocols and Encoding



Protocol	Encoding
HTTP	XML
HTTP	UA Binary
UATCP	UA Binary

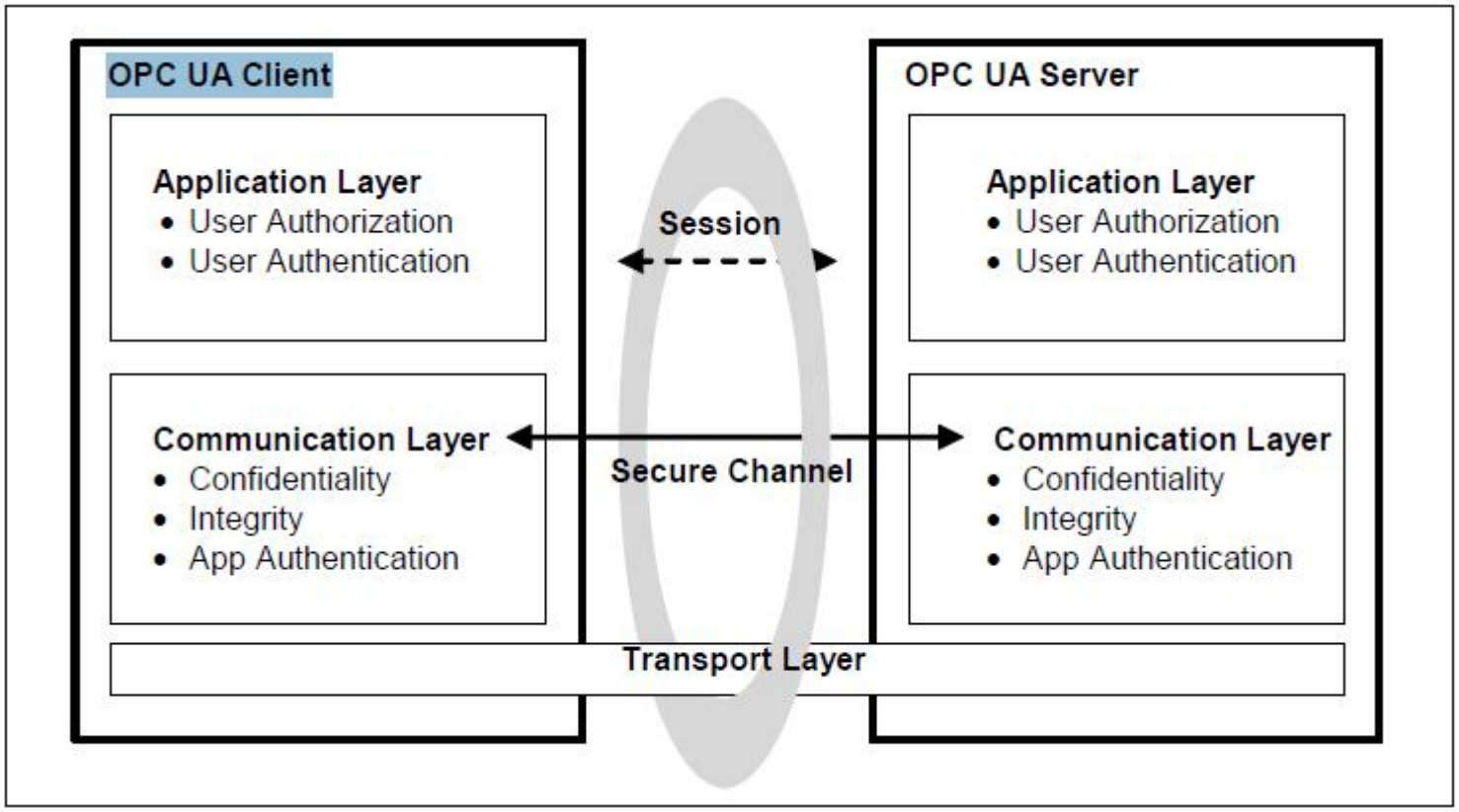


OPC UA Discovery

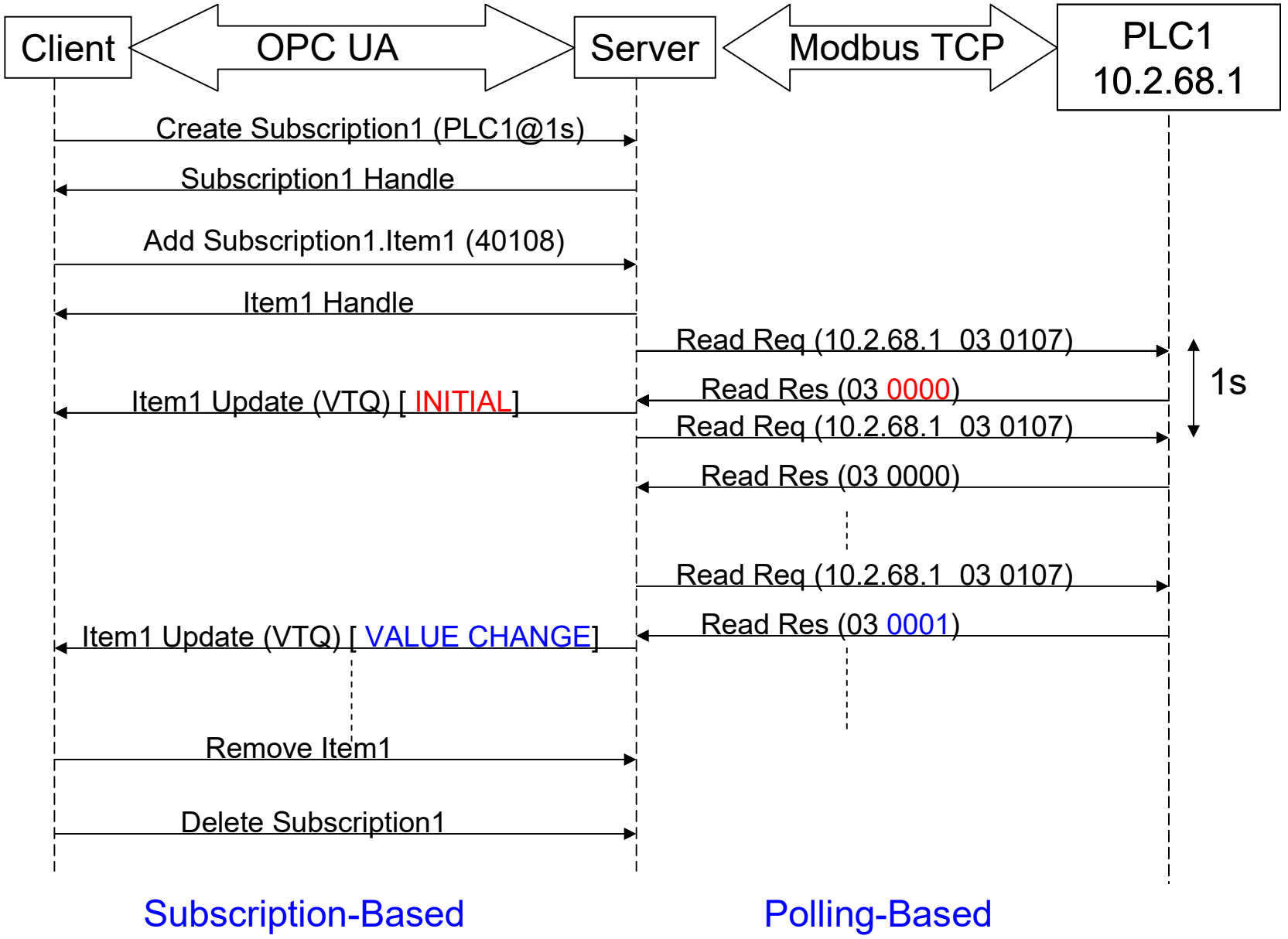


TRANSPORT	WELL KNOWN ENDPOINT
HTTP	http://localhost/UADiscovery
OPC UA TCP	opc.tcp://localhost:4840/UADiscovery
HTTPS	https://localhost:4843/UADiscovery

OPC UA Session



OPC UA Subscription



User Interfaces
HMI & SCADA

Supervisory Control
Before the PC/Microprocessor

Line/Diagram of entire plant with indicator lamps and dial guages for process status.
Physical Knobs and Switches for Supervisory Control

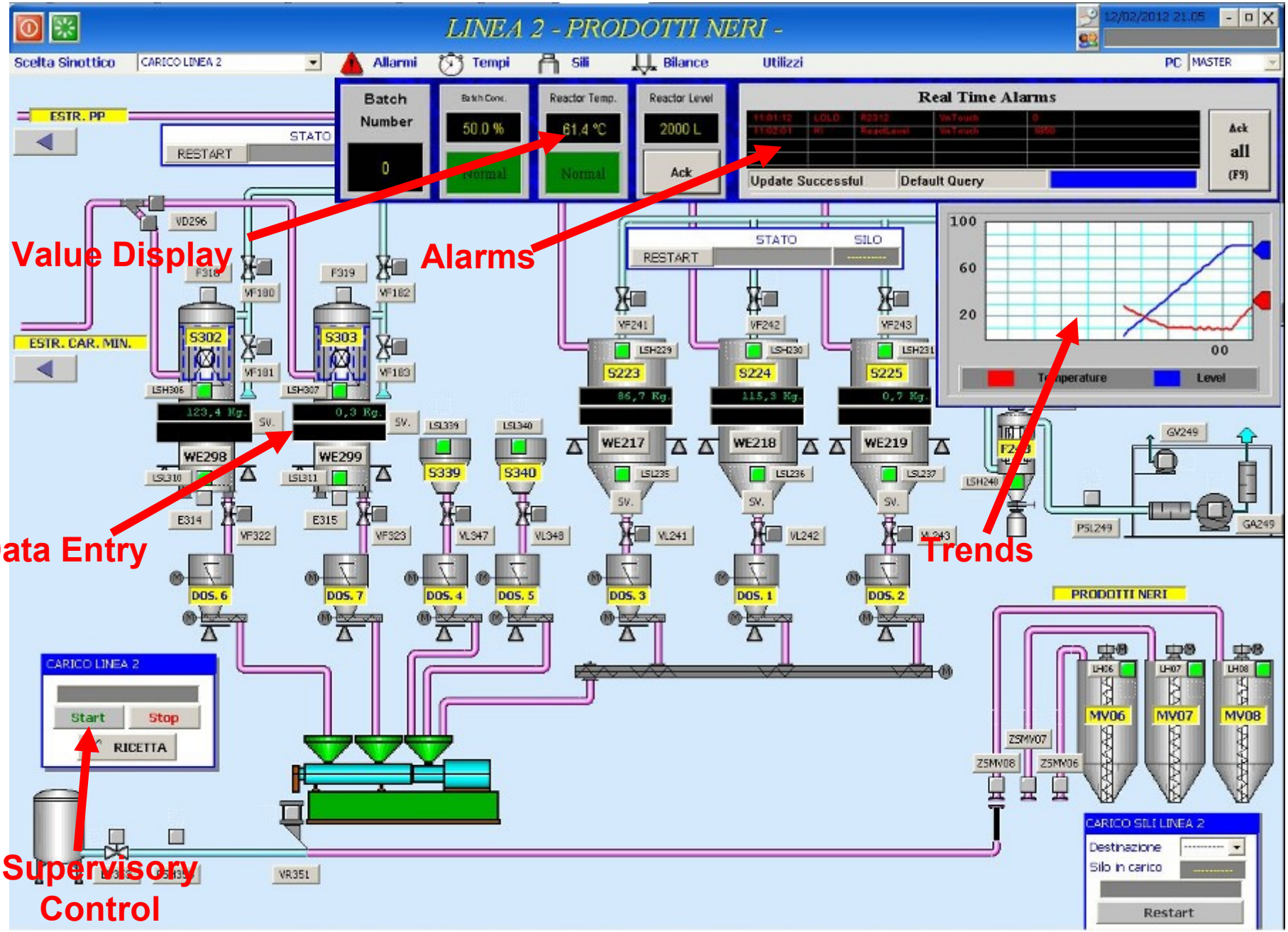
Modern Control Room with PC based SCADA

SCADA-HMI

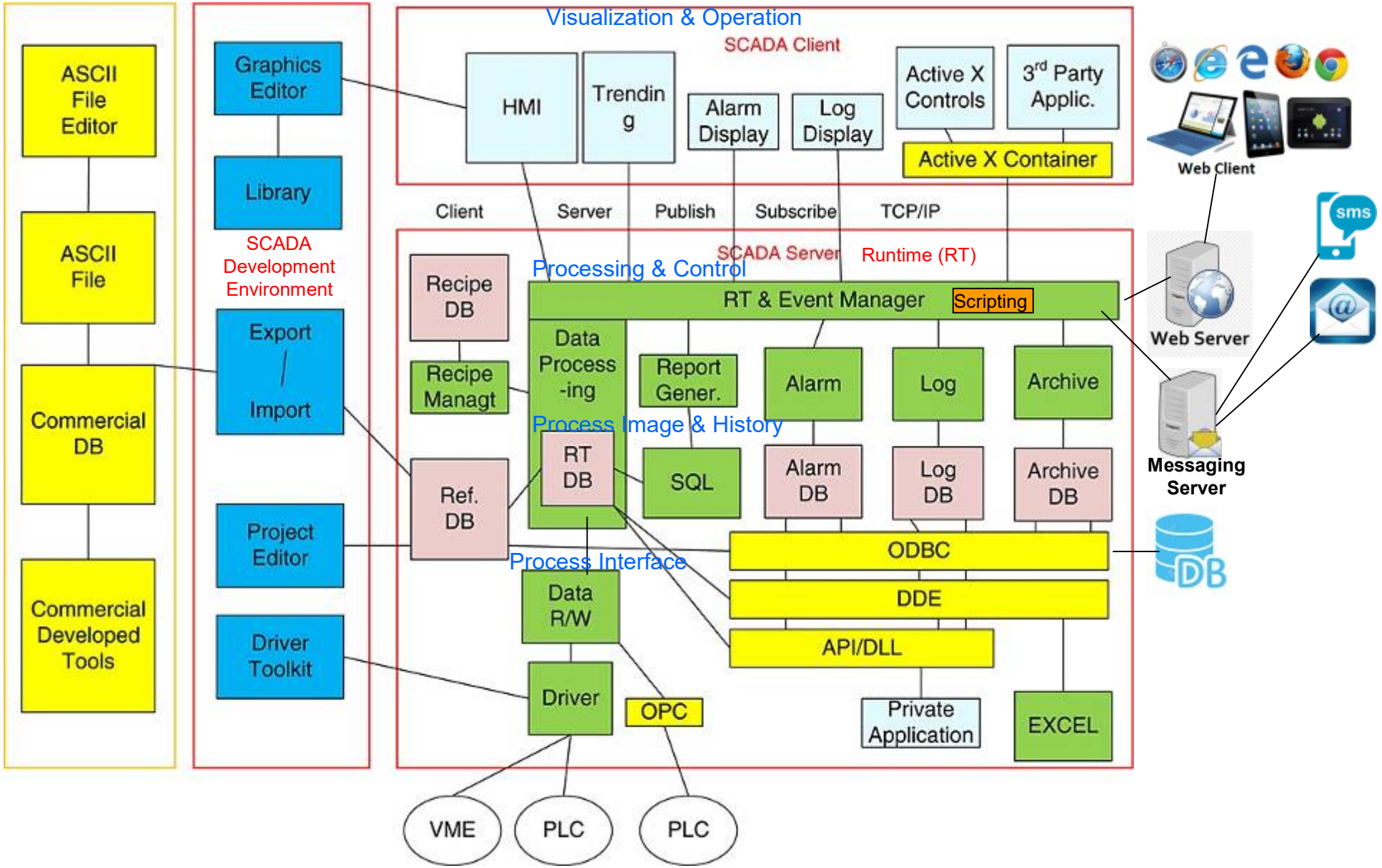
- Industrial Panels
- Panel PCs
- Industrial PCs
- Wireless Tablets

- Control Room Multi-monitor PCs
- WebServer (HTML5) for Browser Access using Remote PCs and Cell-phones

SCADA - HMI


LINEA 2 - PRODOTTI NERI -
Scelta Sinottico
CARICO LINEA 2
Allarmi
Tempi
Sili
Bilance
Utilizzi
PC
MASTER
Batch Number
Batch Conc.
Reactor Temp.
Reactor Level
50.0 %
61.4 °C
2000 L
Ack
Real Time Alarms
Update Successful
Default Query
Ack all (F9)
ESTR. PP
ESTR. CAR. MIN.
RESTART
STATO
SILO
S302
S303
S223
S224
S225
WE298
WE299
WE217
WE218
WE219
S339
S340
DOS. 6
DOS. 7
DOS. 4
DOS. 5
DOS. 3
DOS. 1
DOS. 2
Temperature
Level
PRODOTTI NERI
MV06
MV07
MV08
CARICO LINEA 2
Start
Stop
RICETTA
CARICO SILI LINEA 2
Destinazione
Silo in carico
Restart
Value Display
Alarms
Data Entry
Trends
Supervisory Control

SCADA - Typical Internal Software Architecture



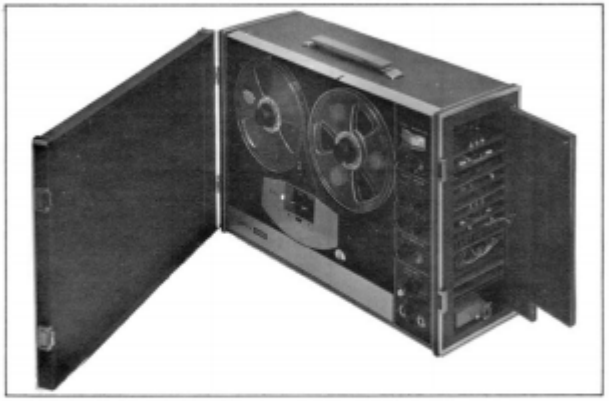
Security and Communication Networks, Volume: 7, Issue: 1, Pages: 175-194, First published: 24 January 2013, DOI: (10.1002/sec.698)

Data Logging

Data Logging
Before the PC/Microprocessor

INSTRUMENTATION/DIGITAL RECORDERS **AMPEX**

SP-700 series 33 channel
instrumentation recorders/reproducers



GENERAL DESCRIPTION

The standard SP700 is capable of three different modes of recording on each of its four magnetic tracks: TIME SHARED, FREQUENCY MODULATION and VOICE. Types of recording can be intermixed and varied to suit different applications simply by substituting plug-in electronics cards.

TIME SHARED Unique time shared recording on the SP700 allows up to 30 multiplexed input channels to be recorded on one magnetic tape track. Time shared electronics sample each input 30 time/second and allow reconstruction of nominal 5Hz data. Standard output electonics incorporate a "sample-and-hold" card for any 5 of the 30 input channels; this provides a nearly analog equivalent of the input to drive readout devices.

FM The standard SP700 can record and reproduce up to four FM channels, each with response from dc to 2.5kHz. Two electronics cards and one magnetic tape track are required for each FM channel. FM recording on the SP700 offers exceptionally good signal-to-noise and is useful for recording low-level acoustic noise, EKG, EEG, vibration, and similar measurements.

VOICE The standard SP700 provides space for one voice electronics card, allowing voice annotation or time reference to be recorded on one magnetic track. An output on the front panel provides power for an external, low-impedance speaker in audio playback.

SPECIFICATIONS

TAPE SPEEDS:
$7\frac{1}{2}$, $3\frac{3}{4}$, $1\frac{7}{8}$ inches per second standard; transport speed control also switches FM center frequency unit for each speed.

TAPE SPEED DEVIATION:
0.25% maximum, long terms, with constant 60Hz or 50Hz source.

REELS AND TAPE:
7" plastic reels standard, $\frac{1}{4}$" width; 1.0 or 1.5-mil acetate or polyester base.

FM RECORDING
FREQUENCY RESPONSE:

Tape Speed	Frequency Response (±1 db)	RMS* S/N Ratio
1 7/8 ips	0 to 625Hz	43 db
3 3/4 ips	0 to 1250 Hz	44 db
$7\frac{1}{2}$ ips	0 to 2500 Hz	45 db

*RMS S/N ratio at center carrier

PHYSICAL CHARACTERISTICS
POWER REQUIREMENTS:
115VAC ±10%, 60Hz single phase; 230 VAC ±10%, 50Hz single phase; 100VA maximum.
TEMPERATURE AND HUMIDITY:
32°F to 110°F operating; 0% to 85% R.H., non-condensing.
DIMENSIONS:
20" x $13\frac{1}{2}$" x 8"
WEIGHT:
38 pounds approximately (with 3 FM record/reproduce and 1 voice channel).

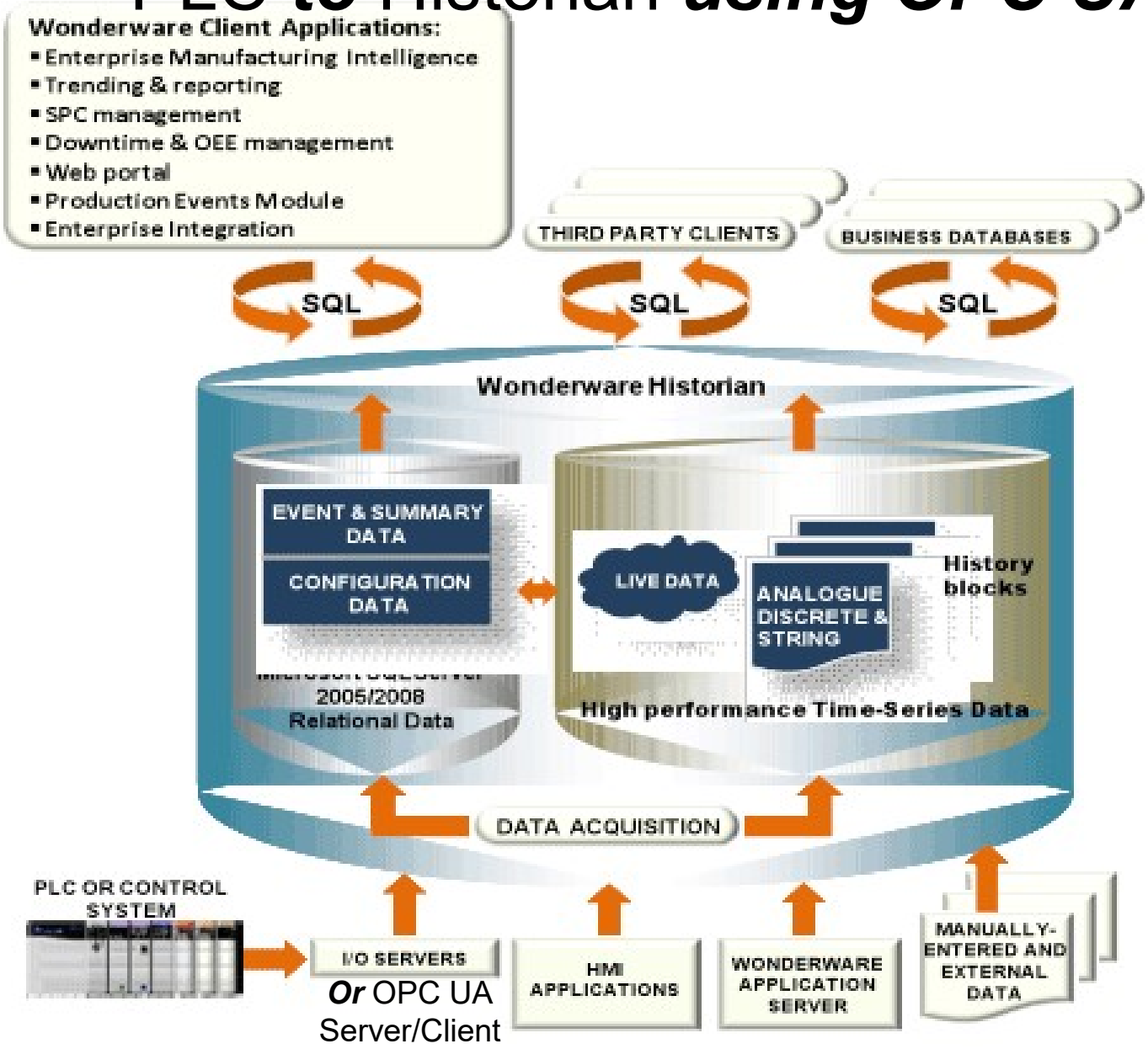
Magnetic Tape Recorder recording – upto 7 channels with Time and Voice Annotations

Strip Chart Paper Recorder

Up to 24 analogue inputs, 36 digital inputs & 36 digital outputs

Data Logging

PLC ***to*** Historian ***using OPC UA Server***

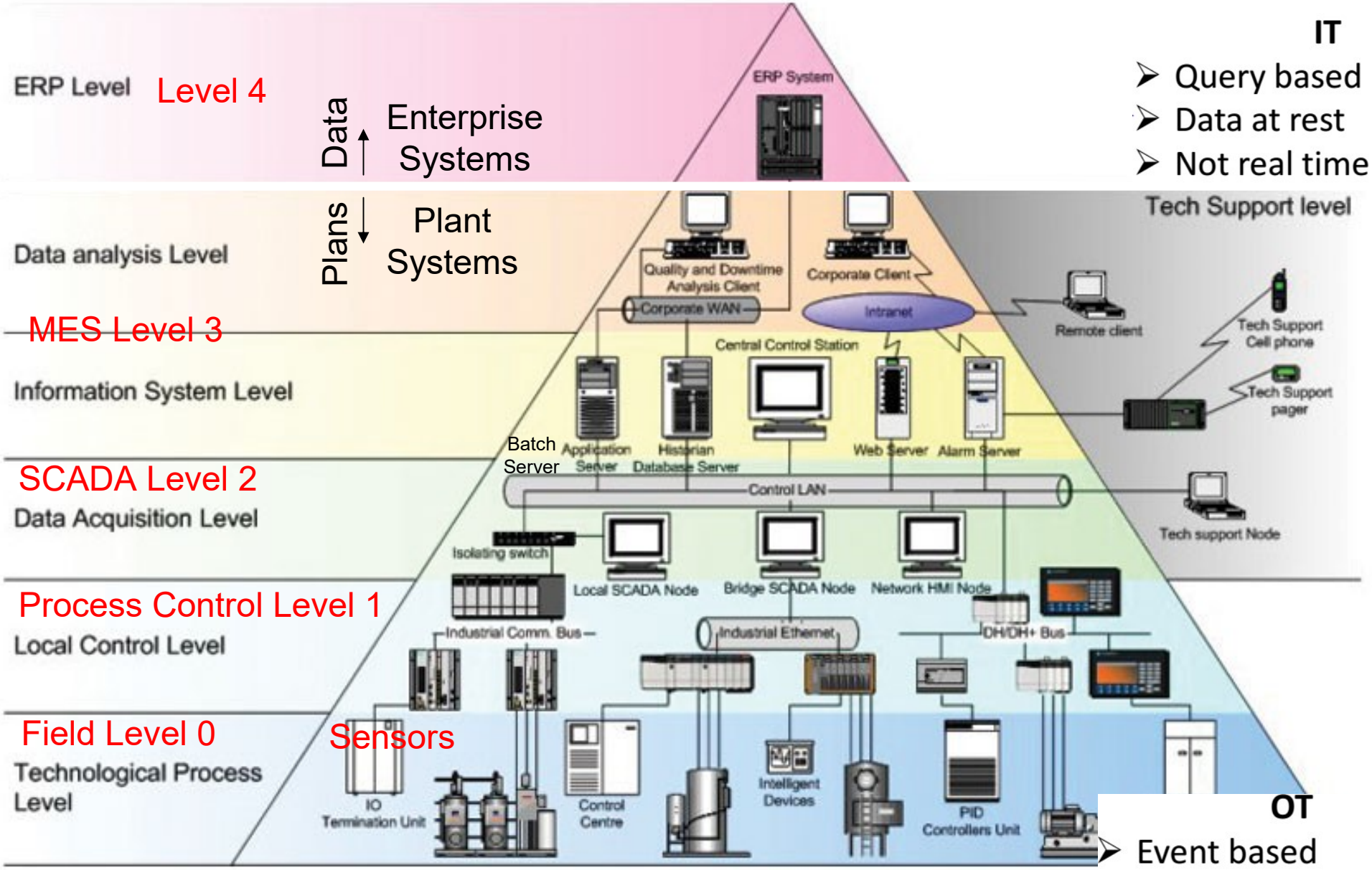


Or OPC UA Server/Client

Refer Later Slide for ***PLC direct access to SQL database***

Industrial Automation Hierarchy

Industrial Automation Levels



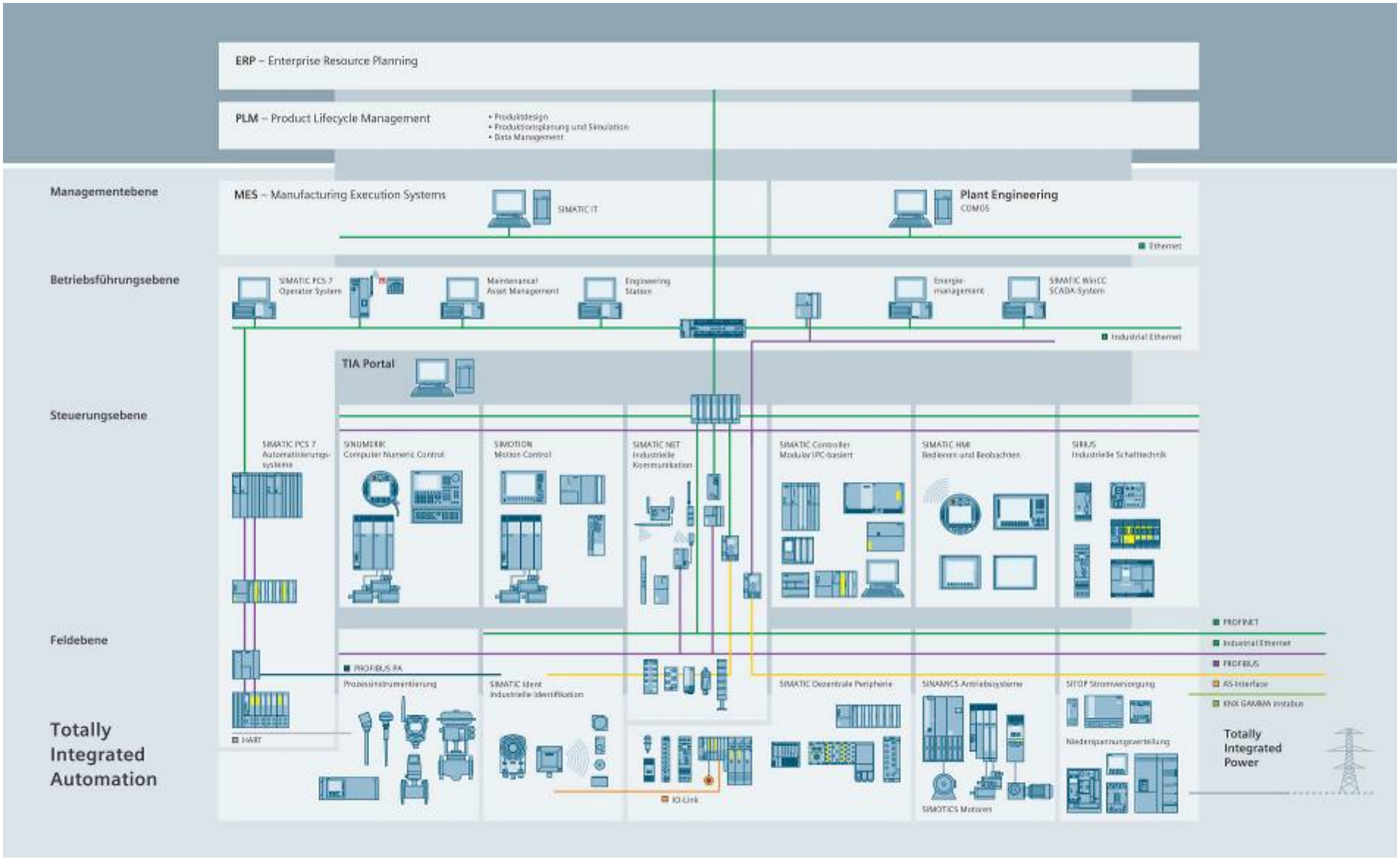
IT

- Query based
- Data at rest
- Not real time

OT

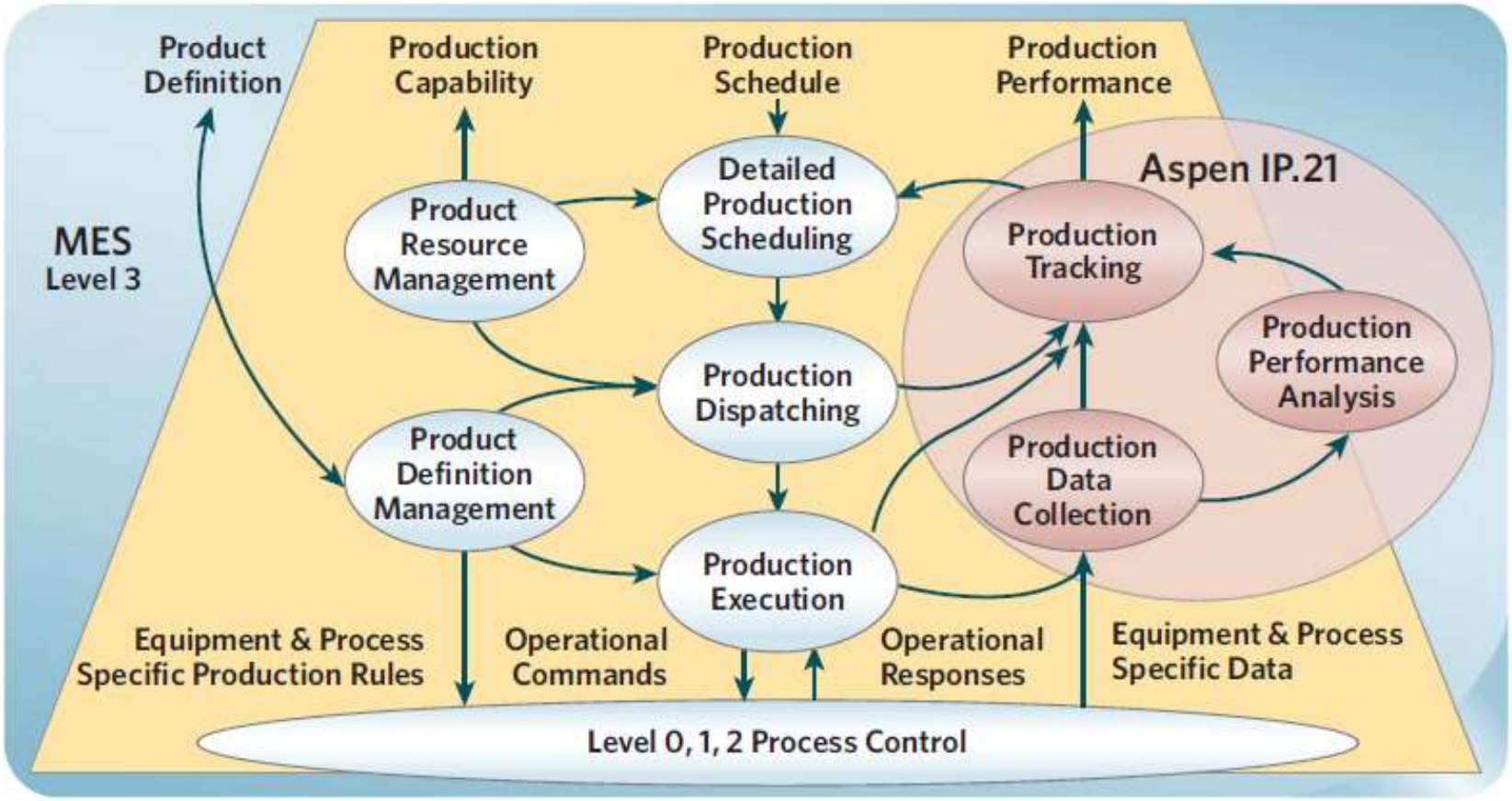
- Event based
- Data in motion
- (Near) real time

Siemens Automation Hierarchy



Level-3 MES Work Flows

Production Maintenance Quality Inventory



ANSI ISA Standards S88 for process control

S95 for integrating enterprise and control systems

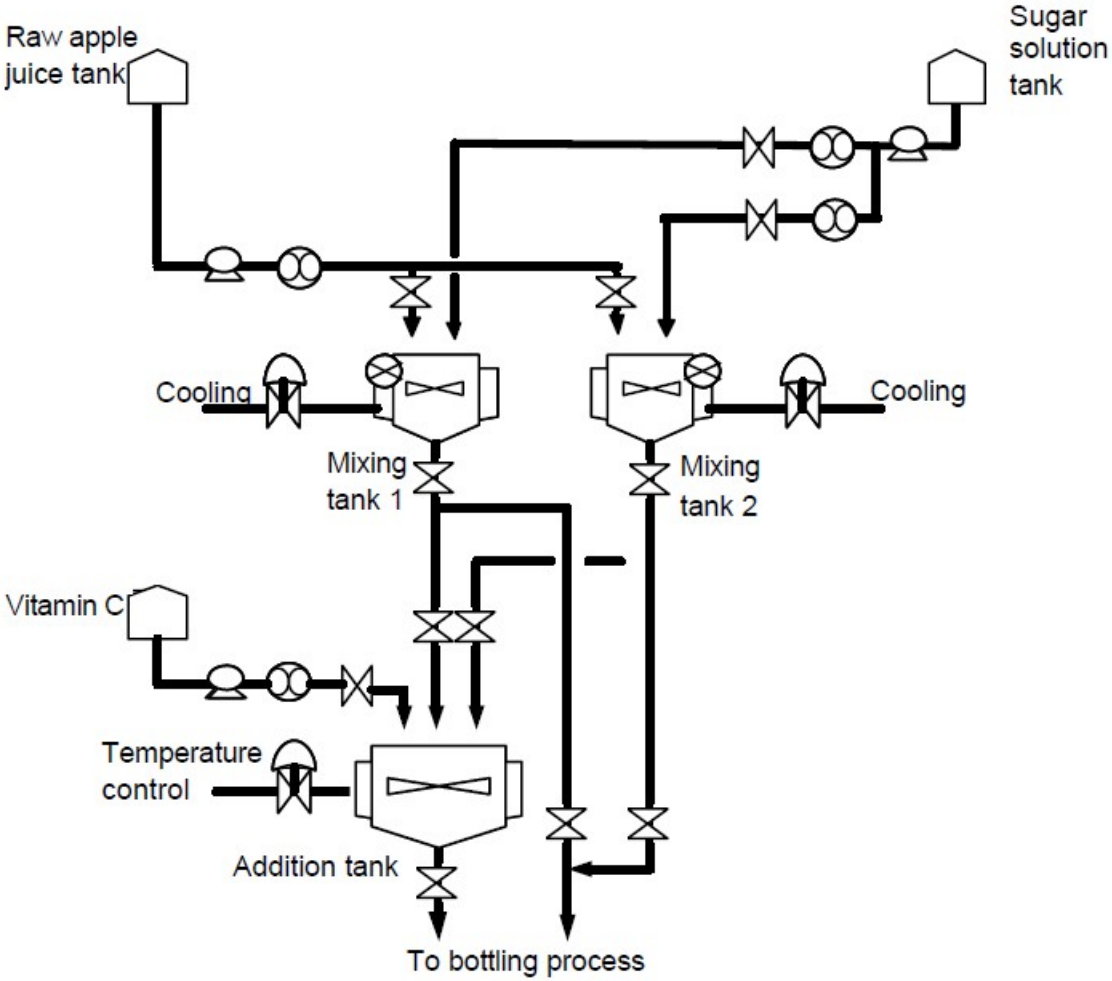
EBR Electronic Batch Records – FDA requirement for pharmaceuticals

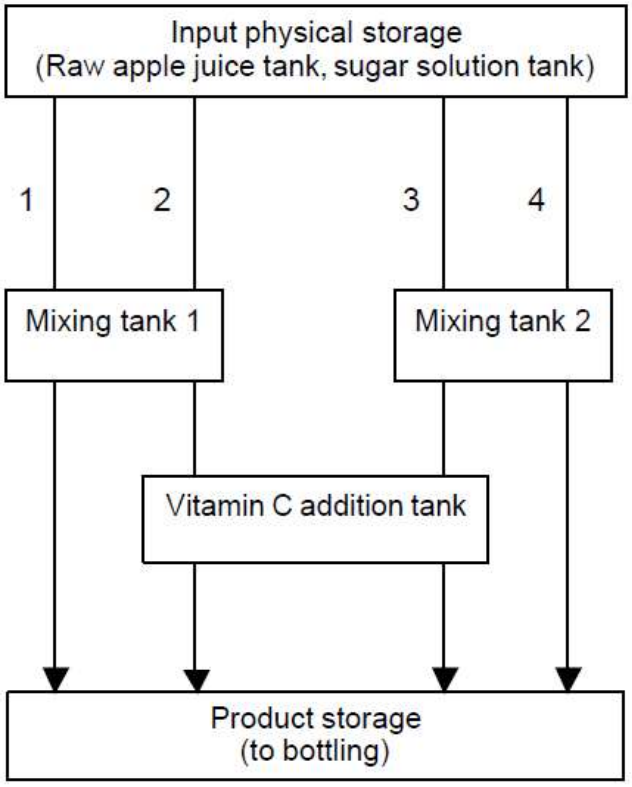
Level 3 Standards

- S88
  - object-oriented programming to conceptualize batch control
  - specifies that the recipe be separated from the process unit or control system where that recipe would be executed
  - shows how the same recipe could be run on different process units, as long as those process units had comparable processing capabilities.
  - The reusability also means that once a “library” of process and recipe models was defined, writing the actual control code could proceed rapidly from those generic models.
- the basic elements of S88 are as follows:
  - a physical model of a manufacturing process, which is a description of what the process does, and what its capabilities are
  - a recipe, which describes the formula (the physical ingredients of a batch) and what actions are to be taken in what order. The recipe is written as a procedural model, which can be subdivided into unit procedures, the unit procedures into operations, and the operations into phases
  - the equipment logic, which is the description of how the procedure (production) is implemented
- S95
  - identifies the interactions between ERP Layer4 and Process Control Layer3.
  - XML Schemas are defined to standardize the data passed between the layers
  - defines terminologies and good practises

- Level3 ->what will be produced
- Level0,1,2 -> how will it be produced

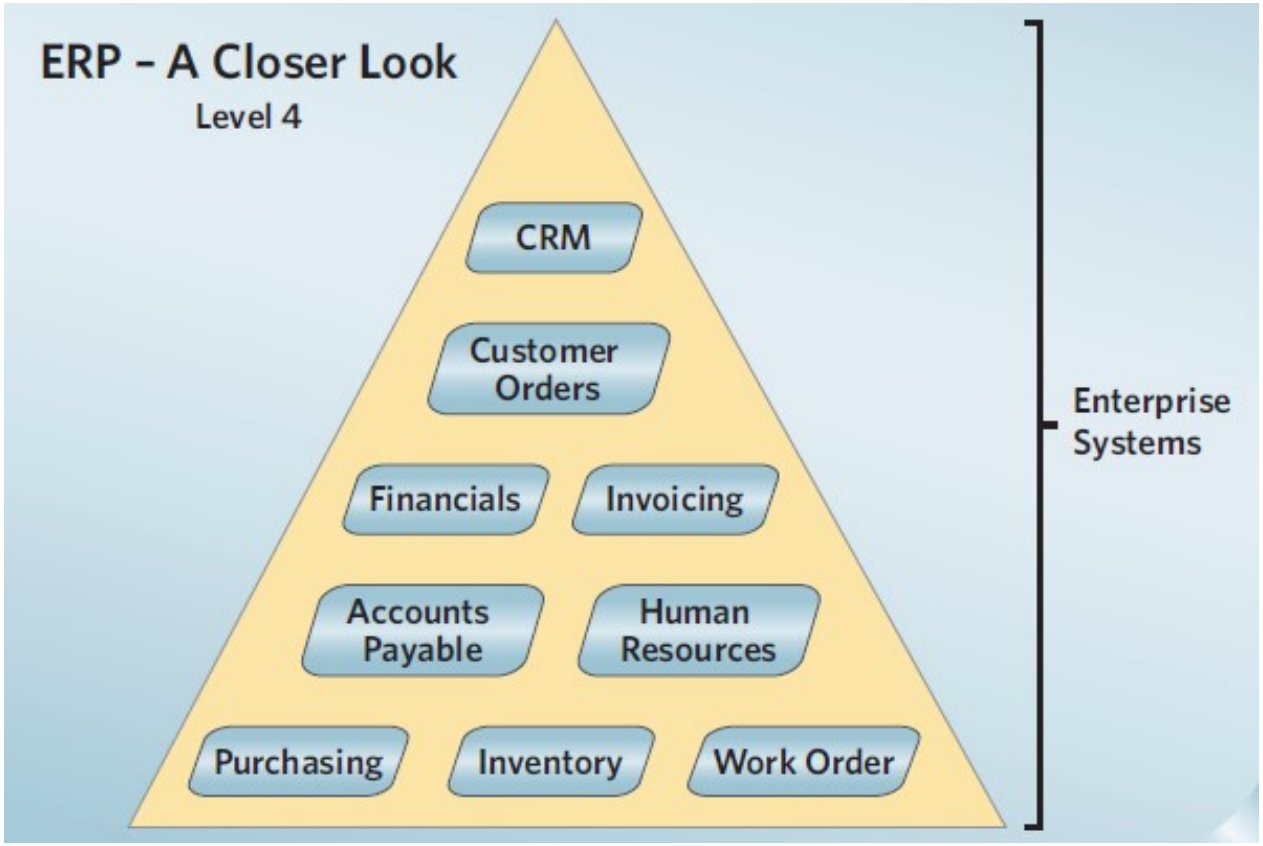
Common Resource Management - Route Management





MES (Batch) System selects the equipment to be used based on the recipe and other batches scheduled / in progress.
Process Data (temperature, ph, density, color ….) is logged for quality control and regulatory bodies

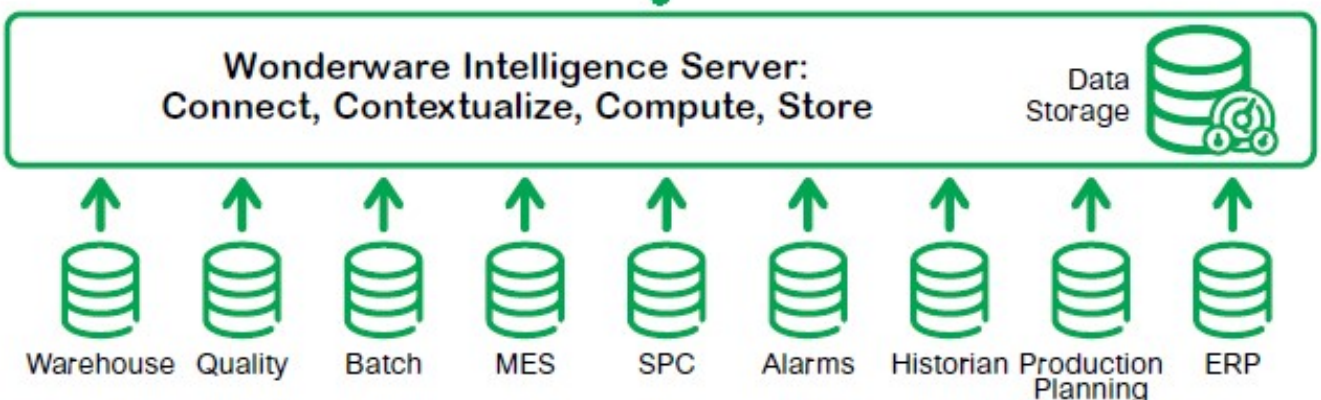
Level-4 ERP



Customer Relationship Management (CRM)
SCM (Supply Chain Management)
PLM (Product Life-cycle Management)

Level-4++
Value Addition by the IT Industry
Manufacturing Intelligence

Monitoring of Key Performance Indicators

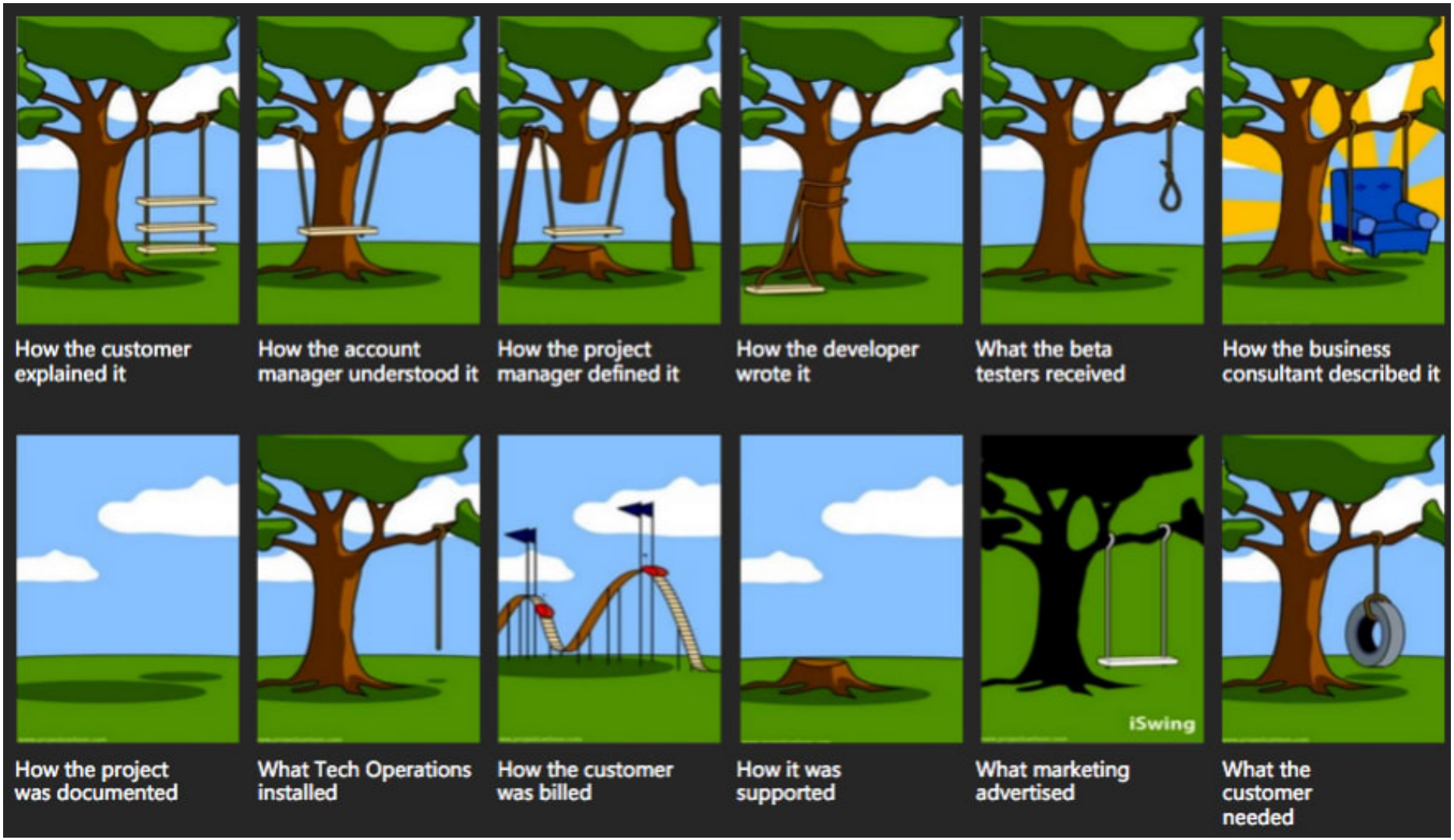


Production, Energy, Inventory.. And other typical “Cloud” Applications

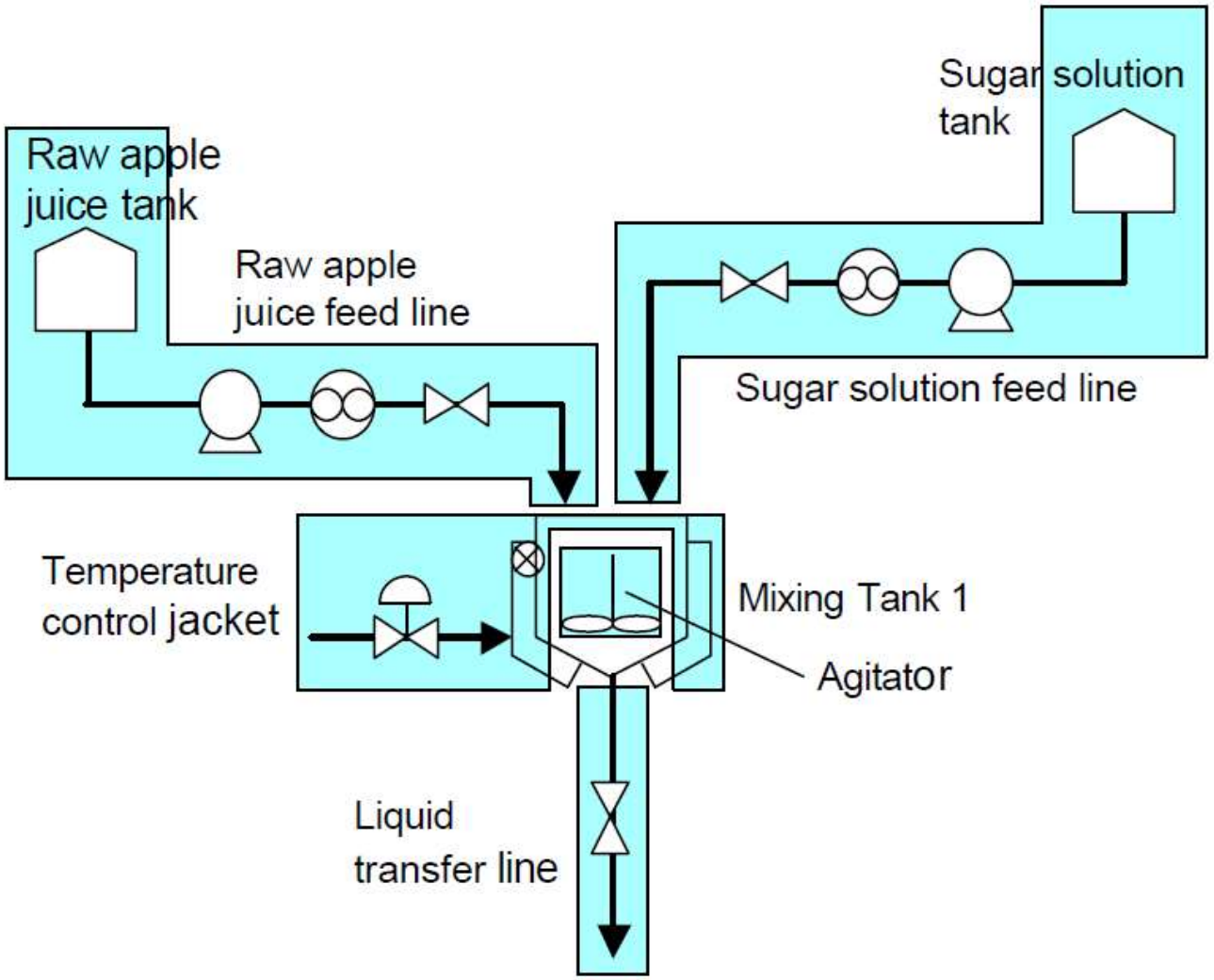
Integrated Workflow

1. Sales force registers enquiry
2. ERP system checks inventory, raw material order and production lead times and provides sales with estimated delivery dates
3. Sales books order in the Sales Module
4. ERP System checks if order can be fulfilled with
   a. Existing finished goods stock
   b. Or using existing raw materials
   c. Or else places orders for raw materials in the Purchase Module
5. Once material has been received and passes Incoming Quality Checks the Inventory System Raw Material stock is updated and Accounting Module is notified to release payments to vendors
6. Based on the availability of raw materials the ERP system places the Production Order in the MES execution queue
7. Based on availability of production line the MES selects the Production Order from the execution queue
8. MES writes the product type and quantity to the SCADA system for the production line that is available
9. SCADA system selects the recipes corresponding to the product type and writes the set-points, quantity and quality criteria to the PLCs / Controllers
10. PLCs execute the production sequences to produce and test the product
11. PLC/SCADA updates the MES with the Finished goods produced
12. MES updates the ERP System with Finished Goods Produced
13. ERP updates the Inventory Module – Finished Goods Stock
14. ERP system checks Accounting Module for customer payments received and updates Dispatch / Delivery Module with the Order
15. After dispatch the ERP system tracks Accounting module for Payments received and closes the order
16. ERP generates alerts to the Service Module at end of warranty period and prompts for AMC and spare sales
17. ERP alerts the Sales & Service Module on End-Of-Life Time to push for sales of new versions. ERP alerts Inventory Module to dispose unused raw materials / finished goods that cannot be diverted to other products

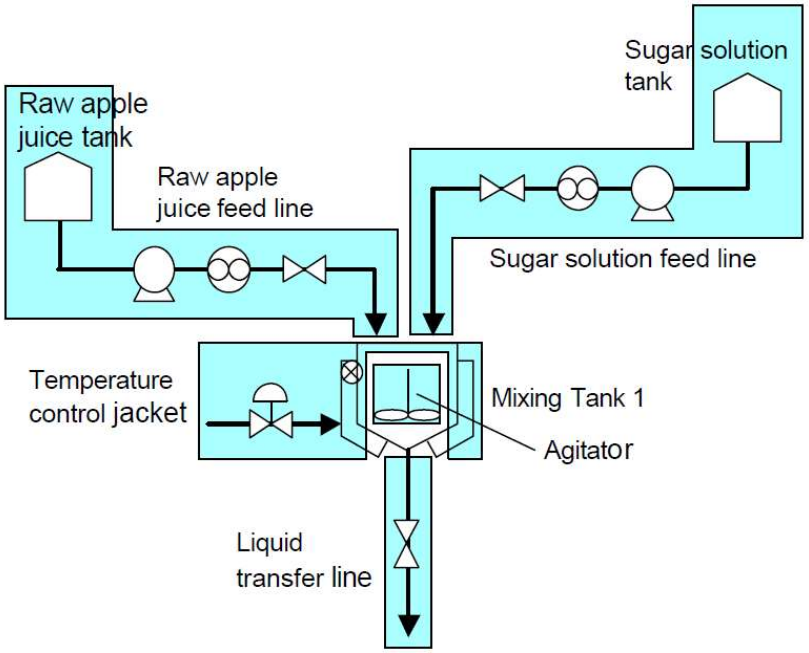
Building A PLC Application



Application Example



PLC Selection – IO Sizing



- Inputs
  - Digital :
    - Start
    - Stop
  - High Speed Digital : Encoder, Pulse Train
  - Analog :
    - Raw Juice Tank Level
    - Sugar Soln. Tank Level
    - Actual Raw Juice Flow Rate
    - Actual Sugar Soln. Flow Rate
    - Mixing Tank Level
    - Mixing Tank
    - Temperature
- Outputs
  - Digital :
    - Raw Juice Valve
    - Sugar Soln. Valve
    - Mixing Tank Out Level
  - High Speed Digital : PWM
  - Analog :
    - Raw Juice Valve Command
    - Actual Sugar Soln. Valve Command

PLC Selection- considerations

- IO Count
- IO Voltage/Current Levels
- IO Distance – individual cables to the PLC or Remote IO System
- Analog Resolution 8/12/16/32-bit
- Control Algorithms – PID, Math Functions, Timers, Counters
- Process Speed – PLC Scan times and IO read/update rates
- Existing PLC / SCADA installations
- Field Networks, Inter-PLC communication
- Operator Station / HMI
- SCADA
- Industry Regulations : Marine, Safety, Redundancy
- Cost of Programming Software
- Cost of PLC Hardware

PLC Programming - Guidelines

- Establish the Customer's Requirement
- Identify the different users of the machine
  - HMI Roles : Operator, Supervisor, Engineer
- Understand the Upstream and Downstream processes
- Identify Normal Sequence of Operation
- Identify Faults/Safety/Shutdown Operation
- Identify Special features (eg. cold start)
- Identify states of the system
  - Identify state entry conditions (Inputs)
  - Identify state conditions (Outputs)
  - Identify state exit conditions (Inputs)
- Based on IO Lists, HMI Setpoints assign PLC IO and Memory
  - Retentive
  - User Access Levels

PLC Ladder Programming - Good Practices

- Let the logic flow downwards -- so at runtime while doing monitor/debug you keep viewing with the execution by pressing Page Down
- No Loops – in certain cases you could use successive scans to simulate loop like behaviour
- Scale and Calculate Analog Inputs at the start
- Update PLC outputs at the end
- Avoid multiple ***coils*** with the same Tag – as the last coil state will determine the output for the remainder of the program and output update
- Right justify coils
- Use SET / RESET of variables with care
- Divide your code into sections, functions
- Test the sections of code as you write them
- Identify and handle fault conditions
- Test extensively in a safe simulated environment before plant trials

HMI-UI Design is just as Critical !!!

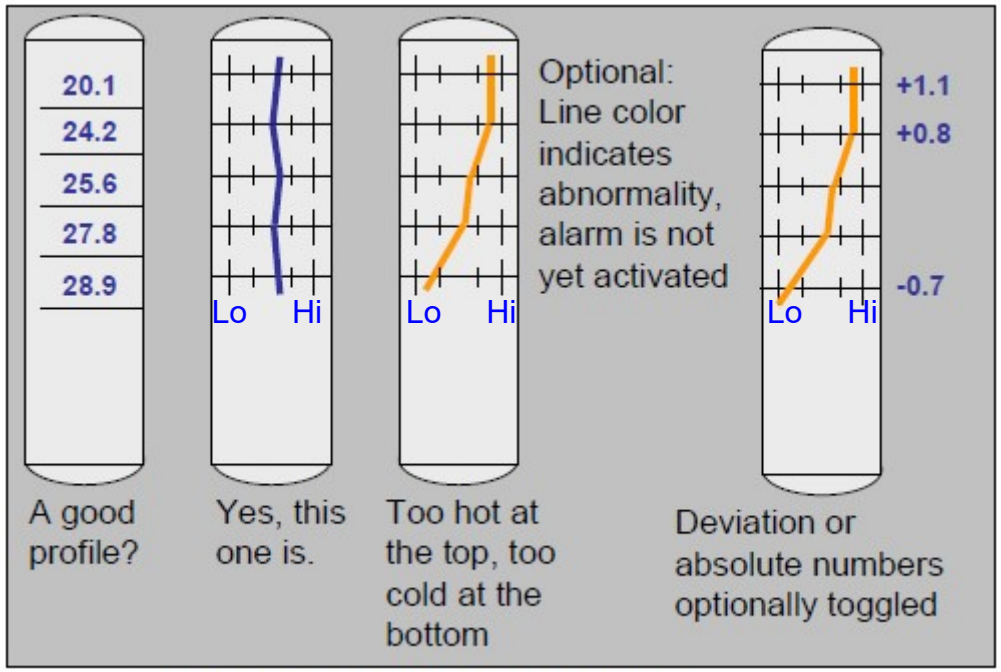
- Logical, readable and in-context presentation of data => Information
- Intuitive, follows operator work-flow, logical navigation with hierarchical content – overview, details, diagnostics
- Clear display of Process State V/s Desired Conditions
- Trend and Status information
- On occurrence of Alarm Conditions guides the Operator with resolution steps for corrective action
- The Controls (on screen elements) that the Operator can use to manipulate the process should be clearly identifiable
- Role based display – Operator need not see settings meant for a Commissioning Engineer
- Consistent colors in line with industry standards
- Avoid excessive detail, flashy graphics, animations

The

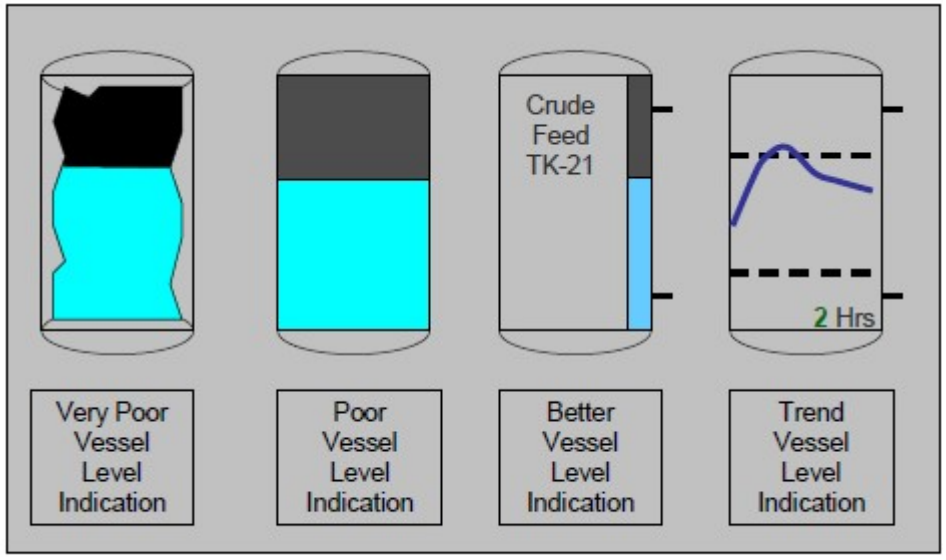
High Performance HMI Handbook

A Comprehensive Guide to Designing, Implementing and Maintaining Effective HMIs for Industrial Plant Operations

By Bill Hollifield, Dana Oliver, Ian Nimmo, &Eddie Habibi



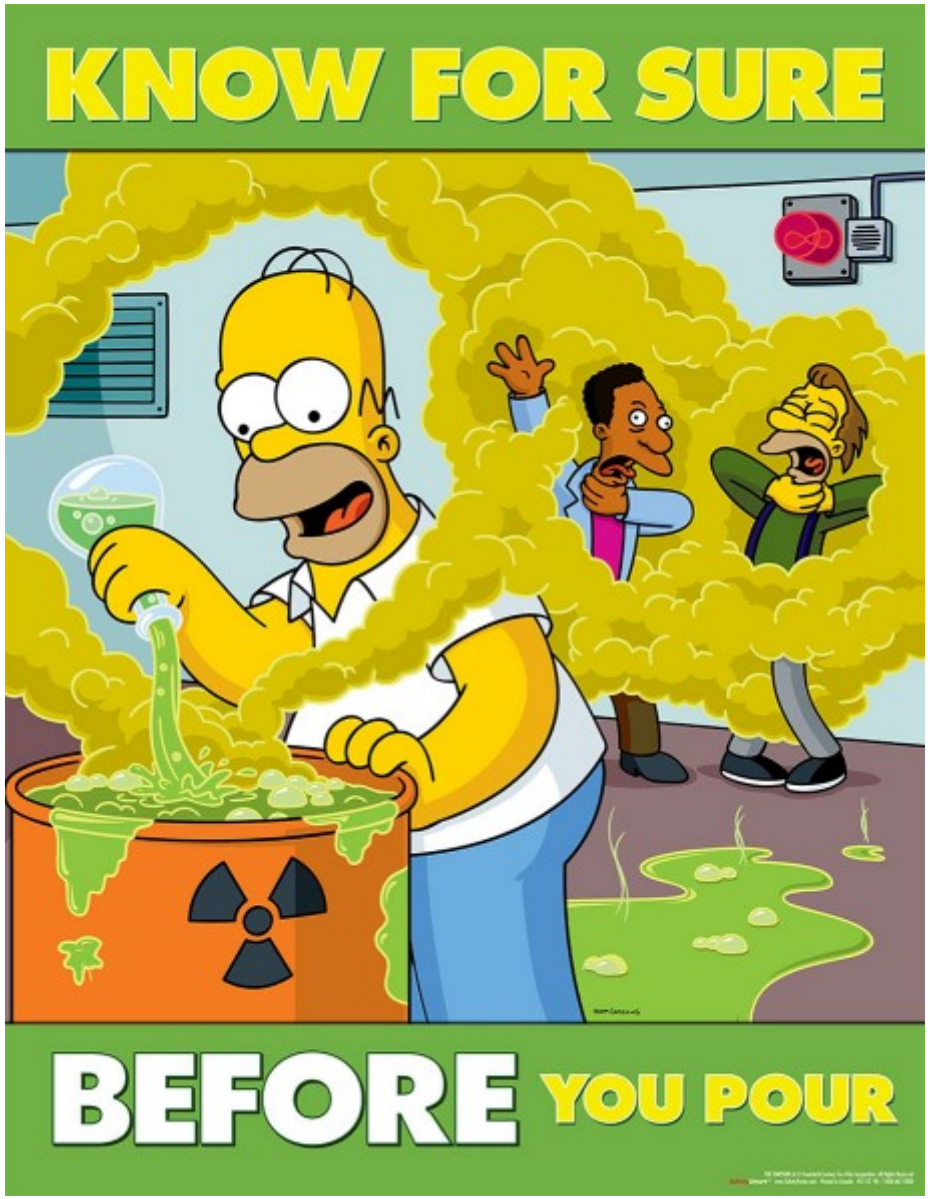
At-A-Glance Indicators – Column Temperatures

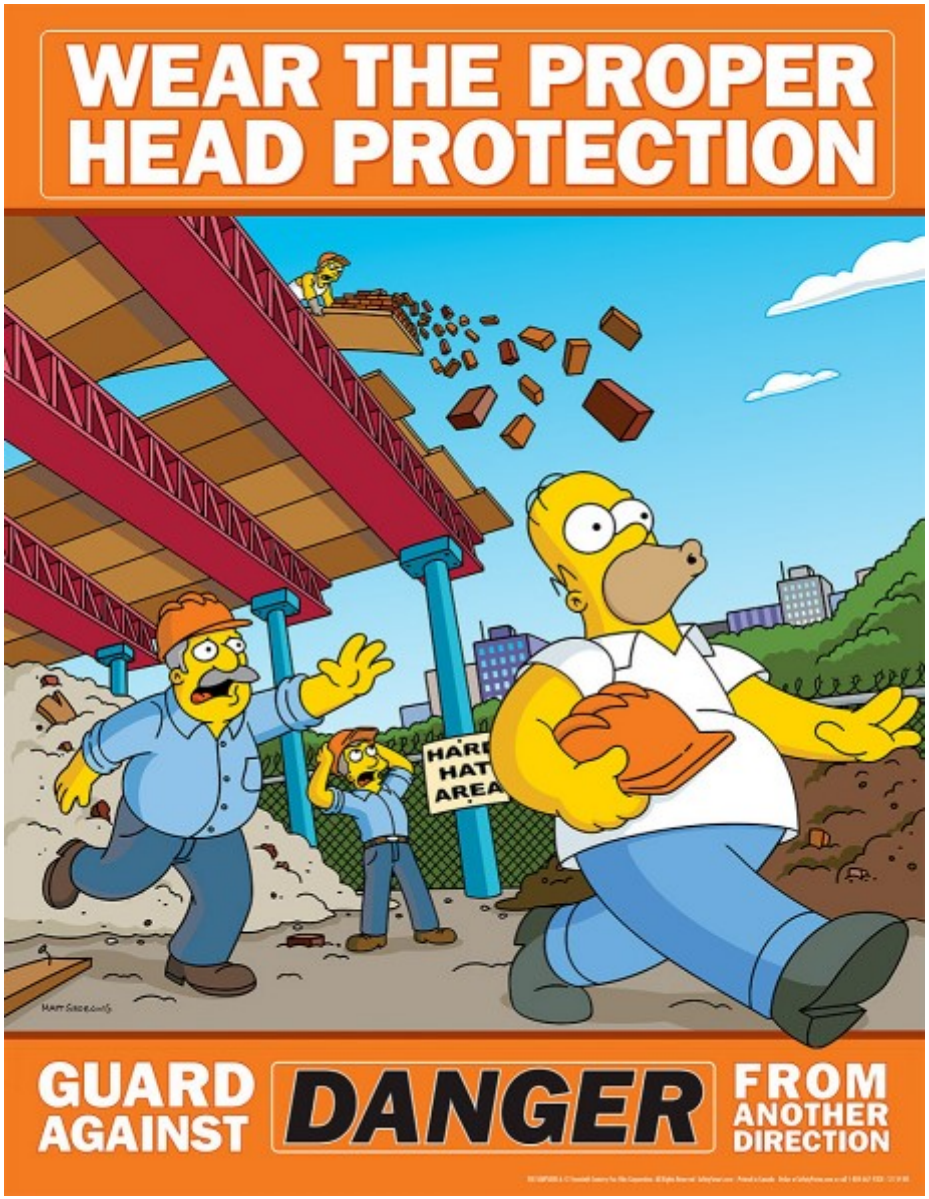


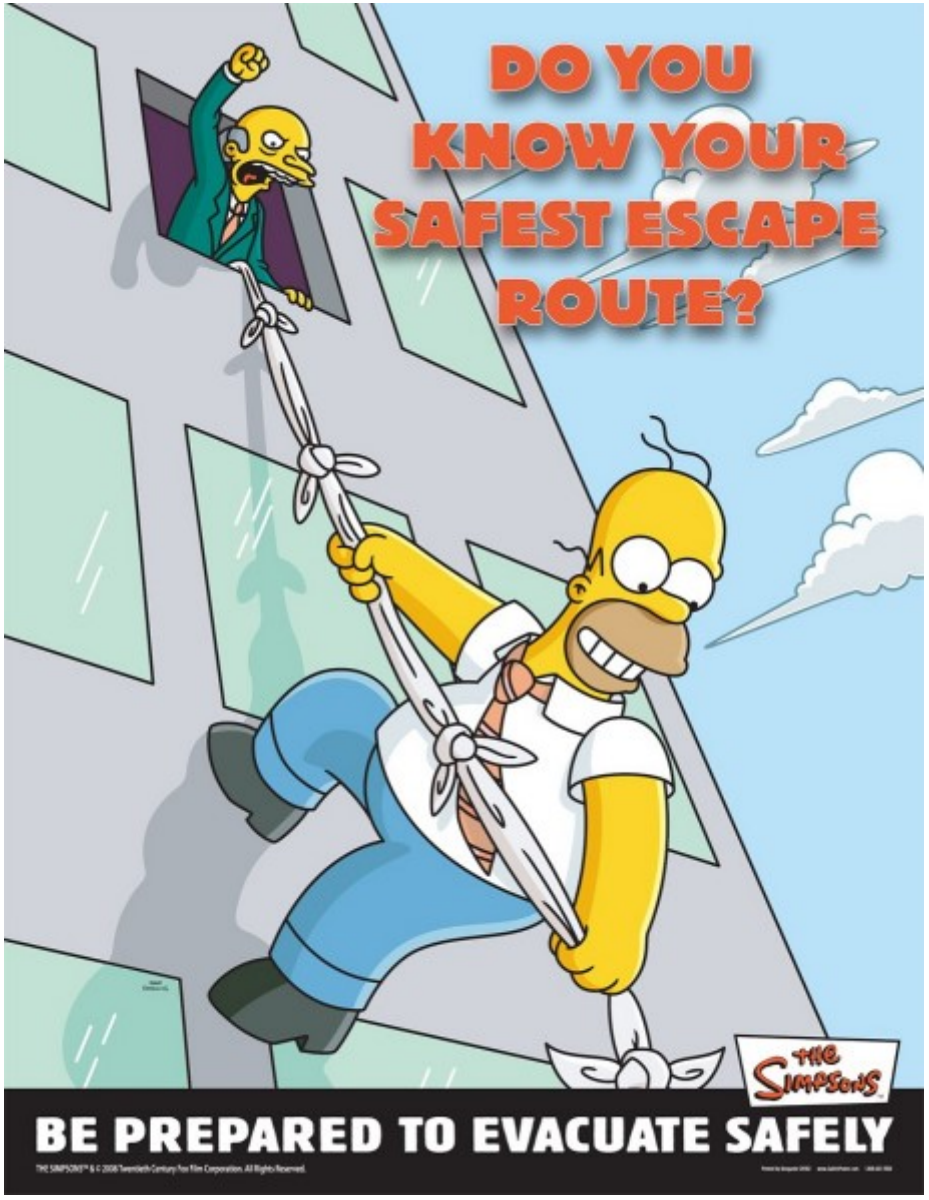
Example Practices for Vessel Levels

PLC Programming – System Startup

- Take interest in understanding the process. Do not blindly follow instructions by the Customer. Is the material corrosive, toxic, explosive…? Flooding, Fire..?
- Ensure that you are wearing the right PPE and follow the safety processes as per the work environment
- Ensure that you know of alternate safe escape/evacuation paths and assembly areas
- Ensure that you know about all the movement paths of the system and are in a safe area while operating the machine.
- Be aware of the other machinery operating near you, wires, pipelines, etc
- Ensure that you know where the Mains Switch, Manual Shut-off Valve is located so that you can react quickly to an emergency
- During wiring ensure that you follow Mains safety Lock-out/Tag-out procedures
- Before starting plant/machine trails ensure that the system is mechanically, hydraulically and electrically sound. All mechanical, hydraulic and electrical interlocks are operational. Mechanical stoppers at the end of strokes, Safety Gate Interlocks… Hydraulic fittings are secure and there are no leakages, Pressure Relief Valves are working… Electrical interlocks – Emergency Stop Switches, Safe Torque Off interlocks on Drives…
- In case of any doubt ensure that you are convinced with the system safety and received full clarifications before you proceed
- Design the program so that you can operate the system in Manual/Test Mode and test each movement/operation a low speed and torque/pressure – this must be verified before full Auto Mode Operation of the System
- Note and investigate any erratic behaviour, glitches in the system operation. Do not ignore warning signs.

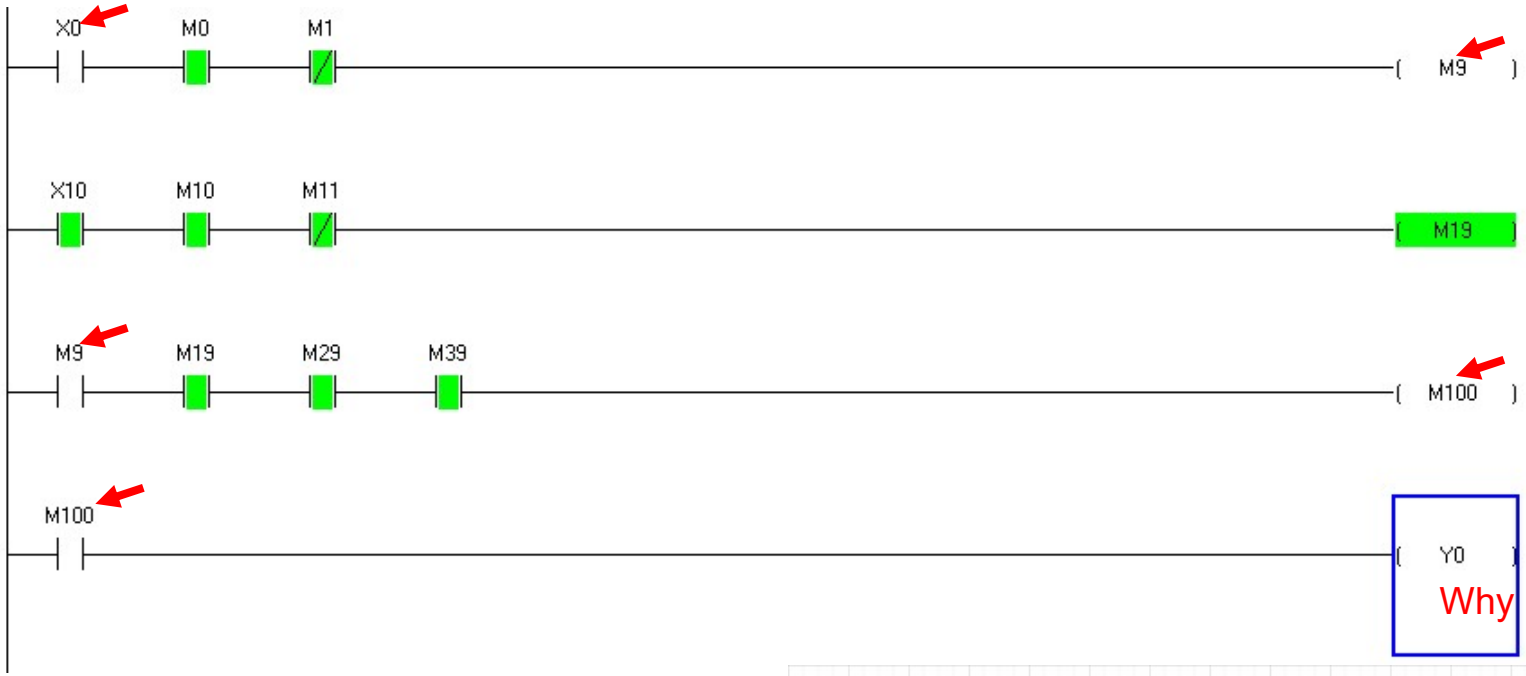

KNOW FOR SURE
BEFORE YOU POUR


WEAR THE PROPER HEAD PROTECTION
GUARD AGAINST DANGER FROM ANOTHER DIRECTION


DO YOU KNOW YOUR SAFEST ESCAPE ROUTE?
THE SIMPSONS
BE PREPARED TO EVACUATE SAFELY


REPORT IT, DON'T IGNORE IT.
ACCIDENT REPORT
EXPLAIN EXACTLY WHAT HAPPENED.

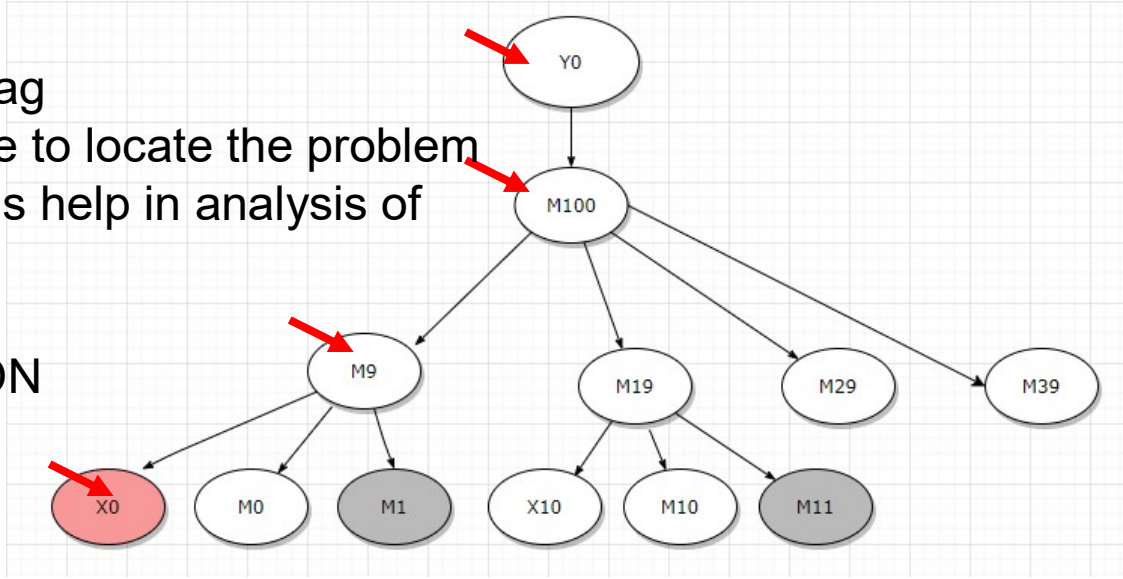
Debugging Ladder Logic



Why is Y0 not ON?

- Back Trace from the Problematic Tag
- Examine the Tag's dependency tree to locate the problem
- Timing diagrams of Dependent Tags help in analysis of Transient conditions

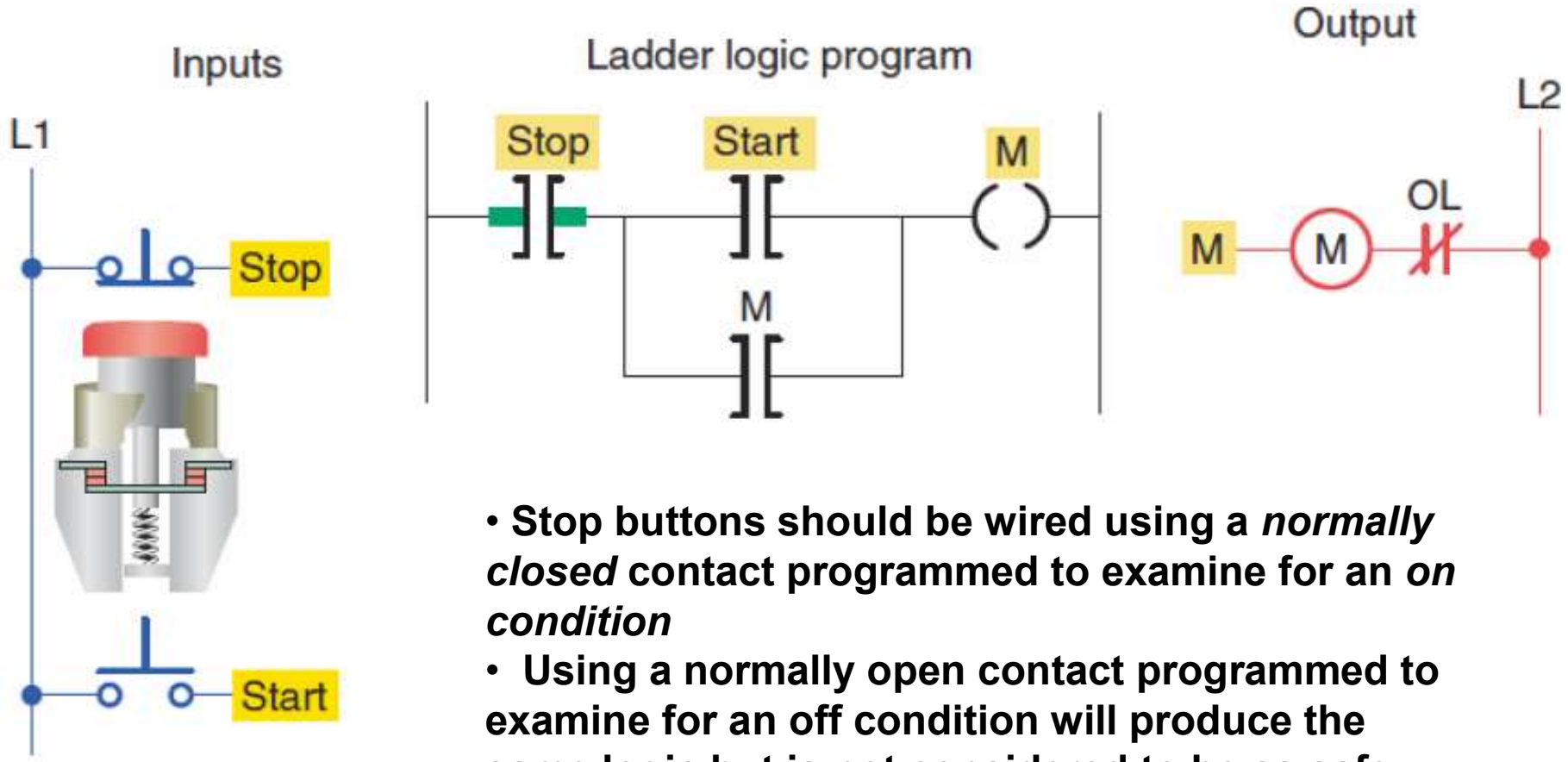
Example Y0 is not ON as X0 is not ON



PLC Code Quality

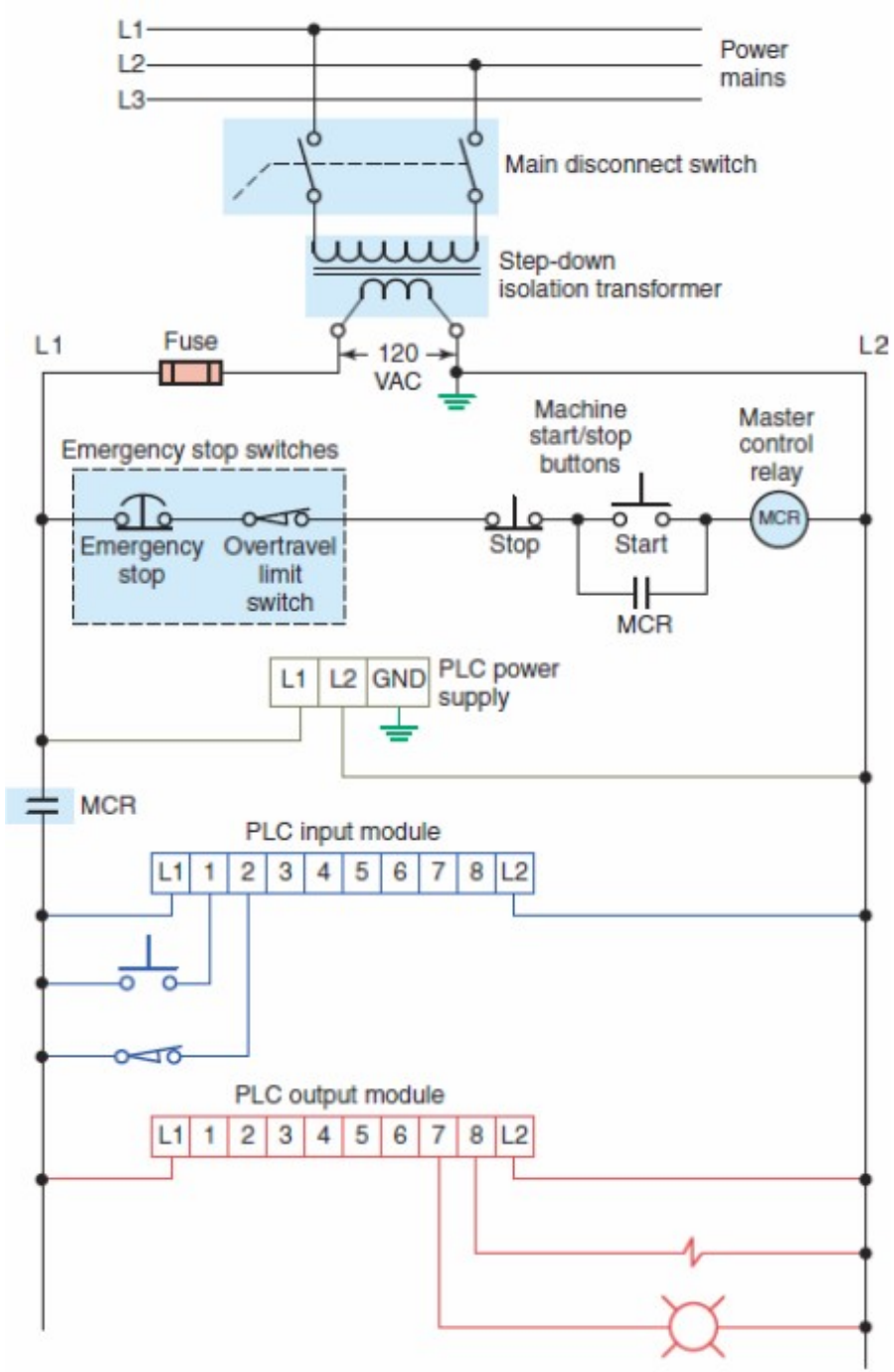
- **Cyclomatic number v(G) :** number of different paths that can be executed in a routine
- **Essential Complexity ev(G) :** number of different unstructured instructions in a routine
- Dead Code
- Unused variables
- Comments

PLC Programming - Safety



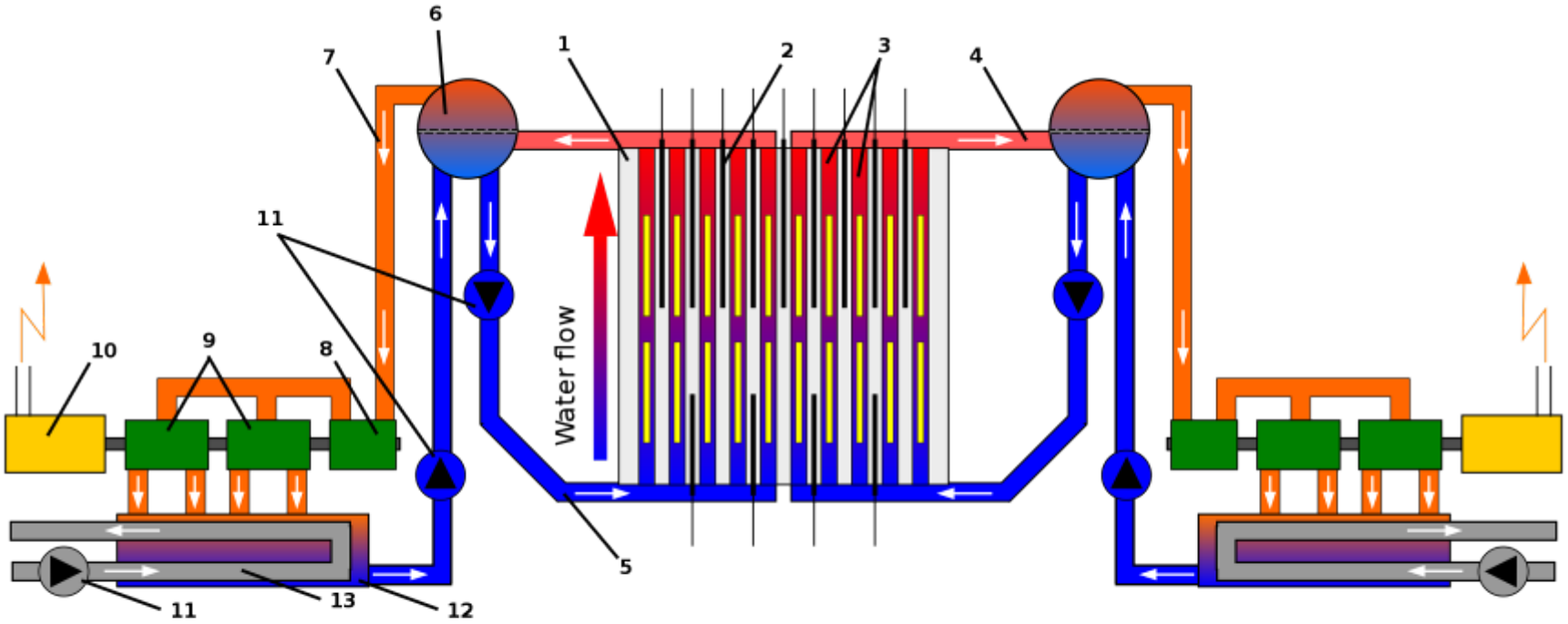
- **Stop buttons should be wired using a *normally closed* contact programmed to examine for an *on condition***
- **Using a normally open contact programmed to examine for an off condition will produce the same logic but is not considered to be as safe.**

PLC Basic Wiring Safety



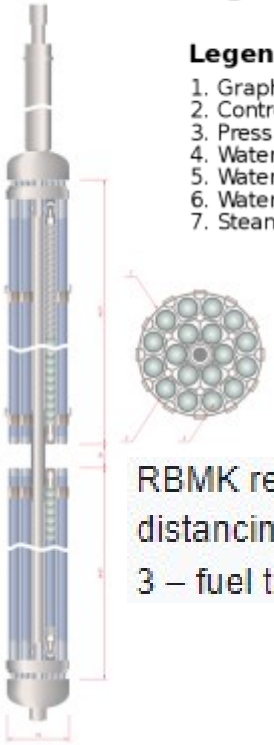
Case Study :
Nuclear Reactor Control

- The Basic Physical System to be controlled should preferably be designed to be inherently stable
- Control Systems should resist operator errors and attempts to be bypassed – ***"A Good Control System Protects Against Human Mistakes"***
- Overall knowledge and "feeling" of the System is very important for the Control Engineer and the Operator.
- If only limited knowledge is available then run the system in the "normal operation zone". Do not push an unfamiliar system to it's limits.

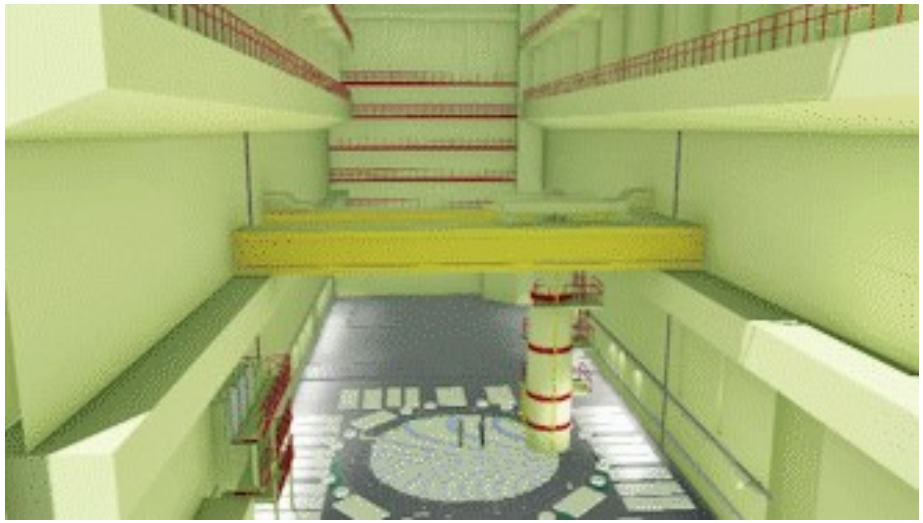


Legend :

1. Graphite moderated reactor core
2. Control rods
3. Pressure channels with fuel rods
4. Water/steam mixture
5. Water
6. Water/steam separator
7. Steam inlet
8. High-pressure steam turbine
9. Low-pressure steam turbine
10. Generator
11. Pump
12. Steam condenser
13. Cooling water (from river, sea, etc.)



RBMK reactor fuel rod holder 1 – distancing armature; 2 – fuel rods shell; 3 – fuel tablets.



The RBMK-1000 Reactor Theory

- Graphite moderated pressure tube type reactor, using slightly enriched (2% U-235) uranium dioxide fuel. It is a boiling light water reactor, with two loops feeding steam directly to the turbines, without an intervening heat exchanger. Water pumped to the bottom of the fuel channels boils as it progresses up the pressure tubes, producing steam which feeds two 500 MWe turbines. The water acts as a coolant and also provides the steam used to drive the turbines. The vertical pressure tubes contain the zirconium alloy clad uranium dioxide fuel cells around which the cooling water flows.
- Surrounding the pressure tubes is the graphite moderator that slows down neutrons to make them more efficient in producing fission in the fuel . A mixture of nitrogen and helium is circulated between the graphite blocks to prevent oxidation of the graphite and to improve the transmission of the heat produced by neutron interactions in the graphite to the fuel channel. In each of the two loops, there are four main coolant circulating pumps, one of which is always on standby.
- The reactivity or power of the reactor is controlled by raising or lowering 211 Boron Carbide Control Rods, which, when lowered into the moderator, absorb neutrons and reduce the fission rate. The power output of this reactor is 3200 MW thermal, or 1000 MWe. Various safety systems, such as an emergency core cooling system, are incorporated into the reactor design.
- The reactor design is simple, scalable and economical. Low levels of fuel enrichment and low fuel density meant that a nuclear explosion (like a nuclear bomb) is not possible.
- Remote controlled refuelling even while the reactor is operating enabled improved uptime and also plutonium production.

The RBMK-1000 Reactor Constraints

- The RBMK reactor can possess a 'positive void coefficient', where an increase in steam bubbles ('voids') is accompanied by an increase in core reactivity.
- The Graphite tips on the Boron Carbide Control Rods can increase core reactivity when the rods are in partially inserted condition. Atleast 15 control rods should be fully inserted in the reactor at all times to maintain stability. But there is no indication of the effective rods in the reactor.
- The Control Rod motion system is slow requiring 18-20s for full insertion.
- At low power levels it is difficult to maintain power output as reactivity decreases due to build-up of neutron absorbing Xenon-135 decreasing power output even further
- The massive graphite core runs very hot – even hotter than the fuel. Air leakage could cause the graphite core to ignite.
- 1700 Pressure tubes run through the hot graphite core resulting in piping that is difficult to maintain. A rupture of a cooling pipe would bring the water directly in contact with the hot graphite risking a steam explosion.
- Steam-Zirconium or Steam-Graphite reactions can produce Hydrogen risking a Hydrogen explosion
- No Reinforced Concrete Containment Structure over the reactor to guard against an explosion
- In the event of a total loss of electrical power the diesel generators require 60-75s to run-up and provide full power to the cooling pumps. There is a 60s window when the reactor cooling could be compromised.

The Safety Test Procedure

- To check ride-through of the core cooling system in case of loss of electrical power - Using the rotational energy of the turbine as it winds down under residual steam pressure to generate electricity to run the cooling pumps
- Steps
  - Reactor must be running at reduced power 700-800 MW
  - Steam-Turbine Generator must be run up to full speed
  - Steam supply of Turbine to be closed
  - Turbine Generator performance to be recorded to verify if it can provide bridging power to the cooling pumps while the diesel generators run-up to full power.
  - Steam Turbine then allowed to free-wheel down
  - Reactor to be shut-down for scheduled maintenance

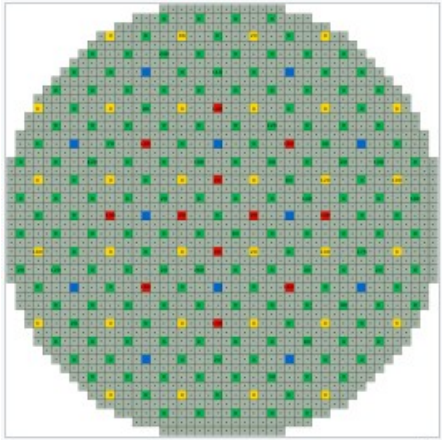
The Sequence of ~~Events~~ *Errors*

- The test is planned for afternoon and output power is gradually reduced to 50% ~ 1500MW
- The Emergency Core Cooling system is turned off in the afternoon and never turned on again.
- Due to Grid Load Issues, the Electrical Grid Controller allows further output power reduction only after midnight.
- The Test proceeds with the night shift personnel who are less experienced
- Power is reduced to 720MW, but power continues to decrease to 500MW due to Xenon-135 build-up.
- The operators try to manually control the power output but it continues to fall to 30MW. Operators respond by raising the Control rods and power increases to 200MW.
- It is decided that the tests be carried out at this power level even though it is much lower than prescribed.
- As part of the test program two additional coolant pumps are activated reducing steam voids and decreasing reactivity. Operators respond by removing more control rods to maintain power. At around this time the number of control rods falls to 8 -- well below the minimum mandated 15.
- The reactor is now in a very unstable state – nearly all control rods are manually raised, excessively high coolant flow rates through the core mean that the coolant is entering the reactor very close to the boiling point. The formation of steam bubbles (voids) from boiling cooling water intensifies the nuclear chain reaction owing to voids having lower neutron absorption than water. The reactor is now at risk of a runaway increase in its core power with nothing to restrain it

The Outcome

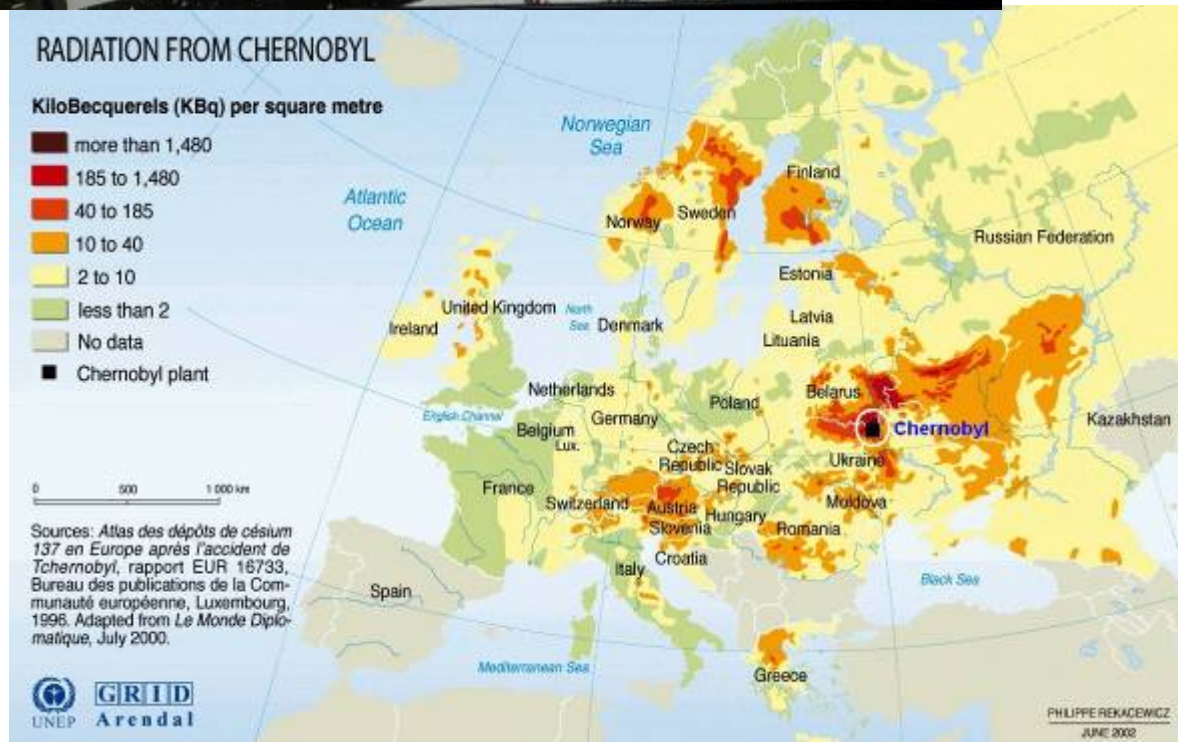
- As per the Test, the steam to the turbines are shut off, beginning a run-down of the turbine generator. The diesel generators start and sequentially pick up loads
- As the momentum of the turbine generator decreases, so does the power it produces for the pumps. The water flow rate decreases, leading to increased formation of steam voids in the coolant flowing up through the fuel pressure tubes.
- A scram (emergency shutdown) of the reactor is initiated as the Test is wrapping up – with the intention to shut-down the reactor for scheduled maintenance.
- Control Rods are lowered to reduce power. Graphite Tips of the Boron Carbide control rods increase rate of fission when the control rods are in partially inserted position causing a Power Surge. **- This feature of control rod operation is counter-intuitive and not known to the reactor operators**. The core overheats, causing some of the fuel rods to fracture - releasing the fuel elements into the coolant and rupturing the channels in which these elements are located. Due to the intense heat the control rods distort and jam at 1/3rd insertion. Power increases to 500MW in 3s and jumps to +30GW, 10+ times the rated output. The cooling water comes in contact with the hot graphite core and flashes to steam causing a steam explosion that blows out the lid of the reactor
- Power continues to increase to 100 times or more of the reactor output creating intense heat. Zirconium-steam/graphite-steam reactions produce hydrogen that explodes, disperses the core and stops the chain reaction.
- The hot graphite from the core comes in contact with air, catches fire and spews radioactive material high into the atmosphere

Few Design Mistakes + Few Human Errors = Continent-wide Disaster



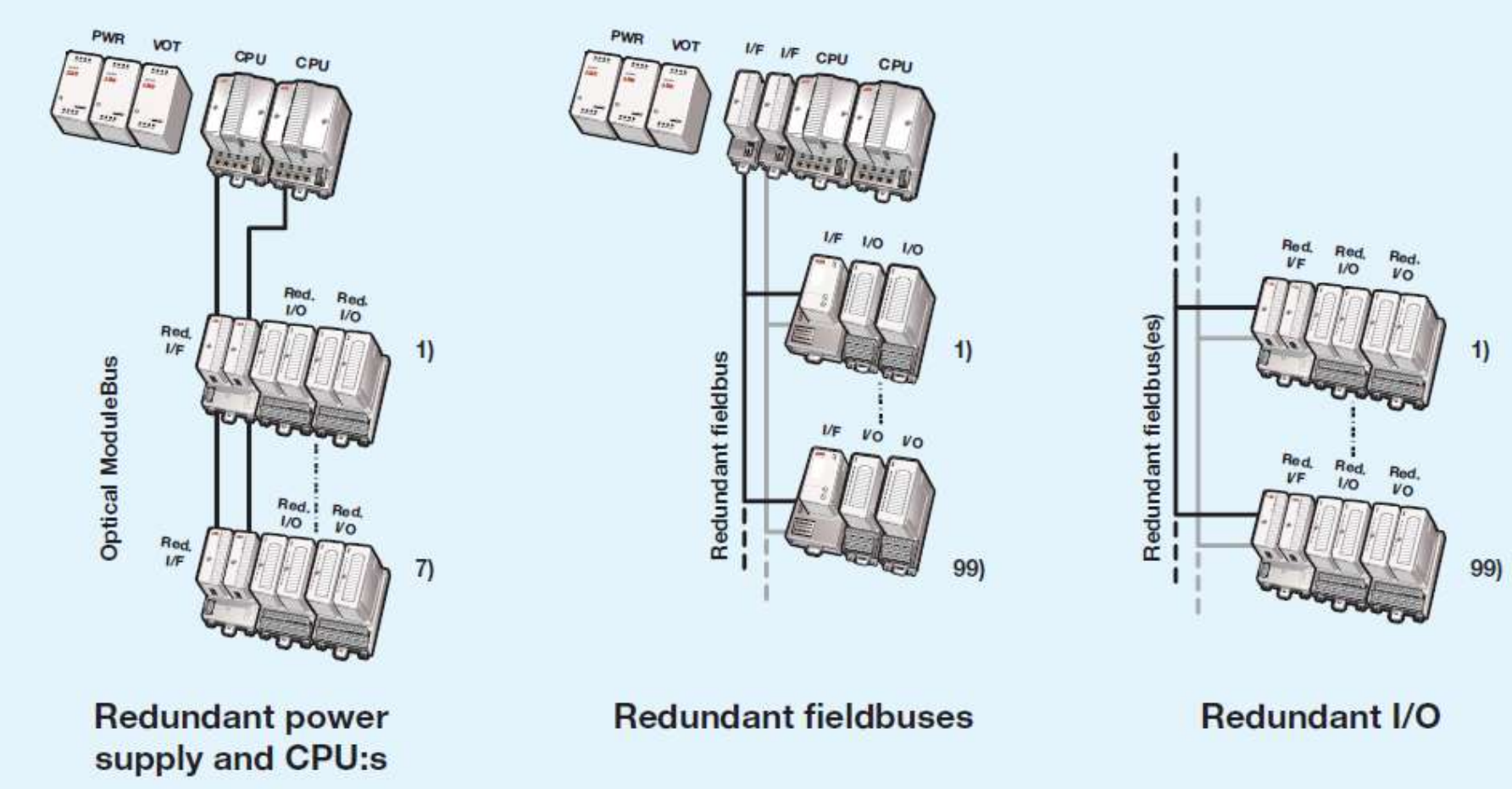
Plan view of reactor No. 4 core. Numbers show insertion depths of control rods in centimeters one minute prior to the explosion.

- startup neutron sources (12)
- control rods (167)
- short control rods from below reactor (32)
- automatic control rods (12)
- pressure tubes with fuel rods (1661)



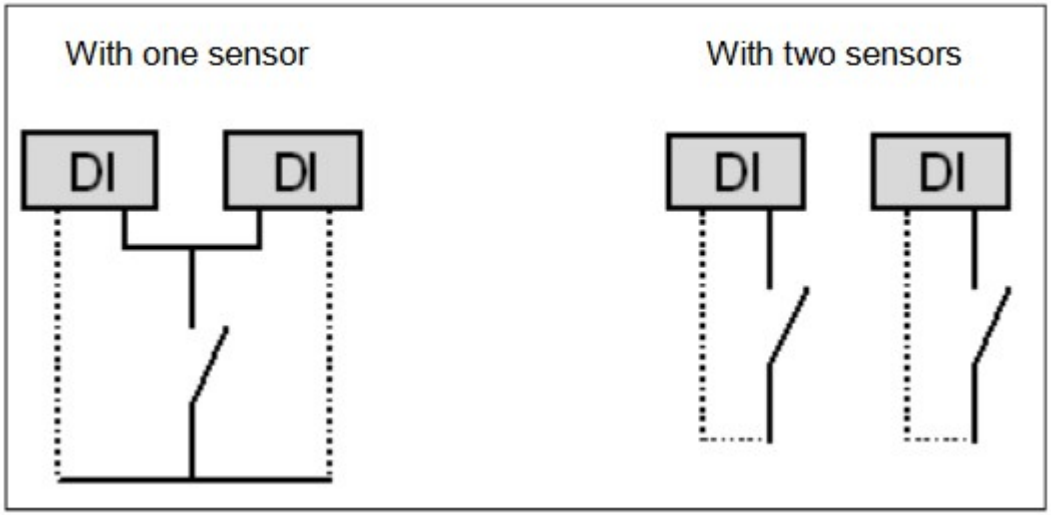
Redundancy

PLC Redundancy

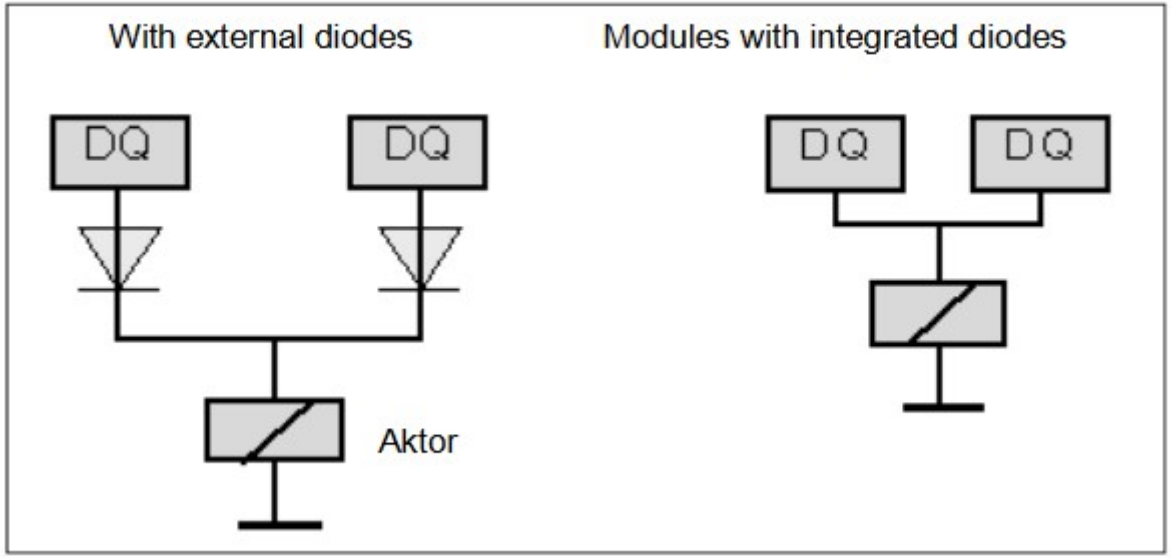


Redundancy is enabled by configuration and no modifications are required in the PLC program

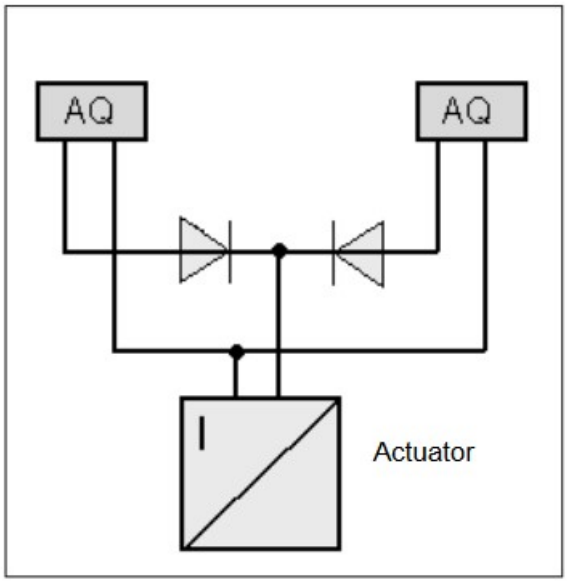
Digital IO Redundancy



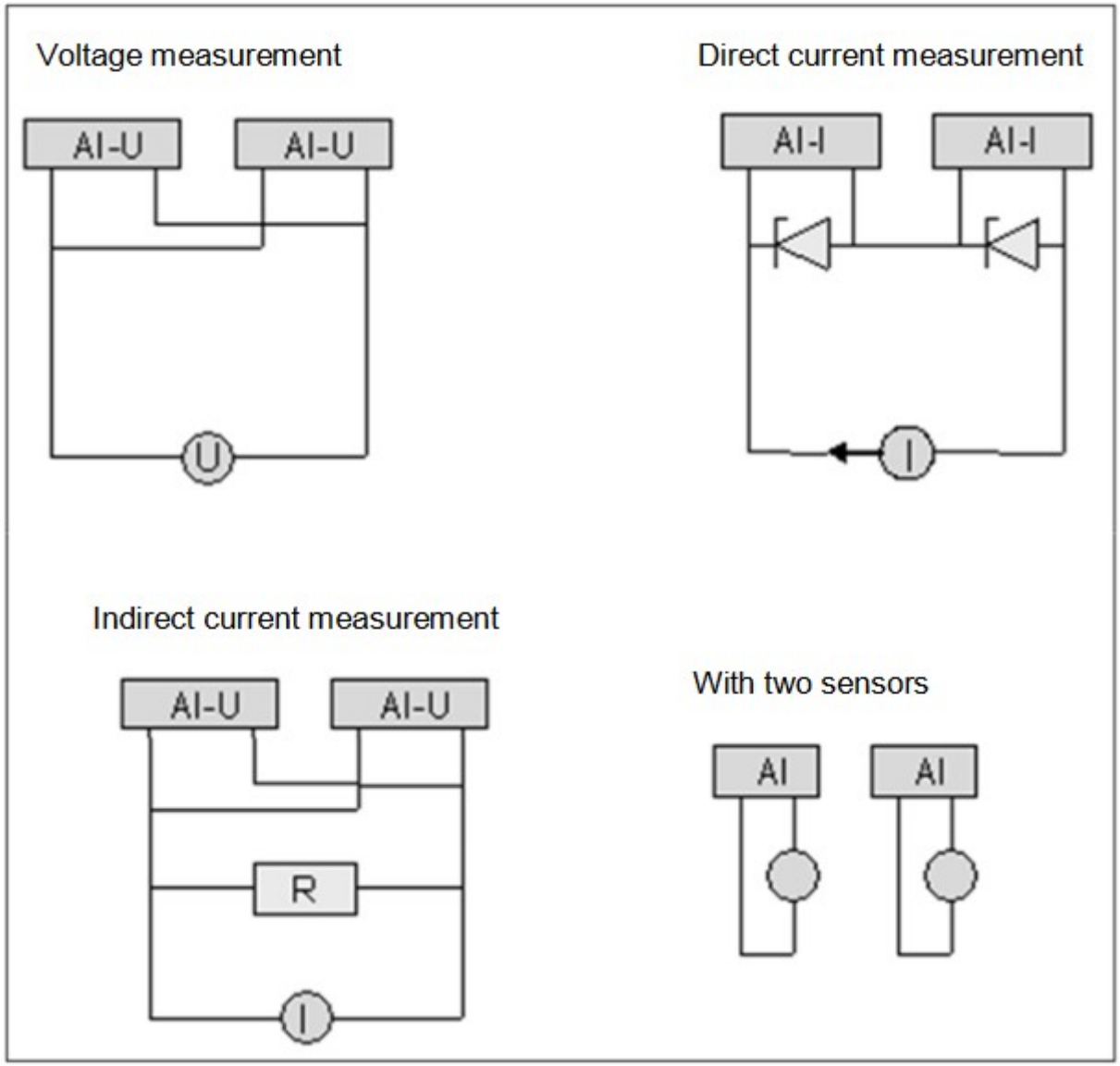
It is recommended to use
2 sensors for max. availability



Analog IO Redundancy

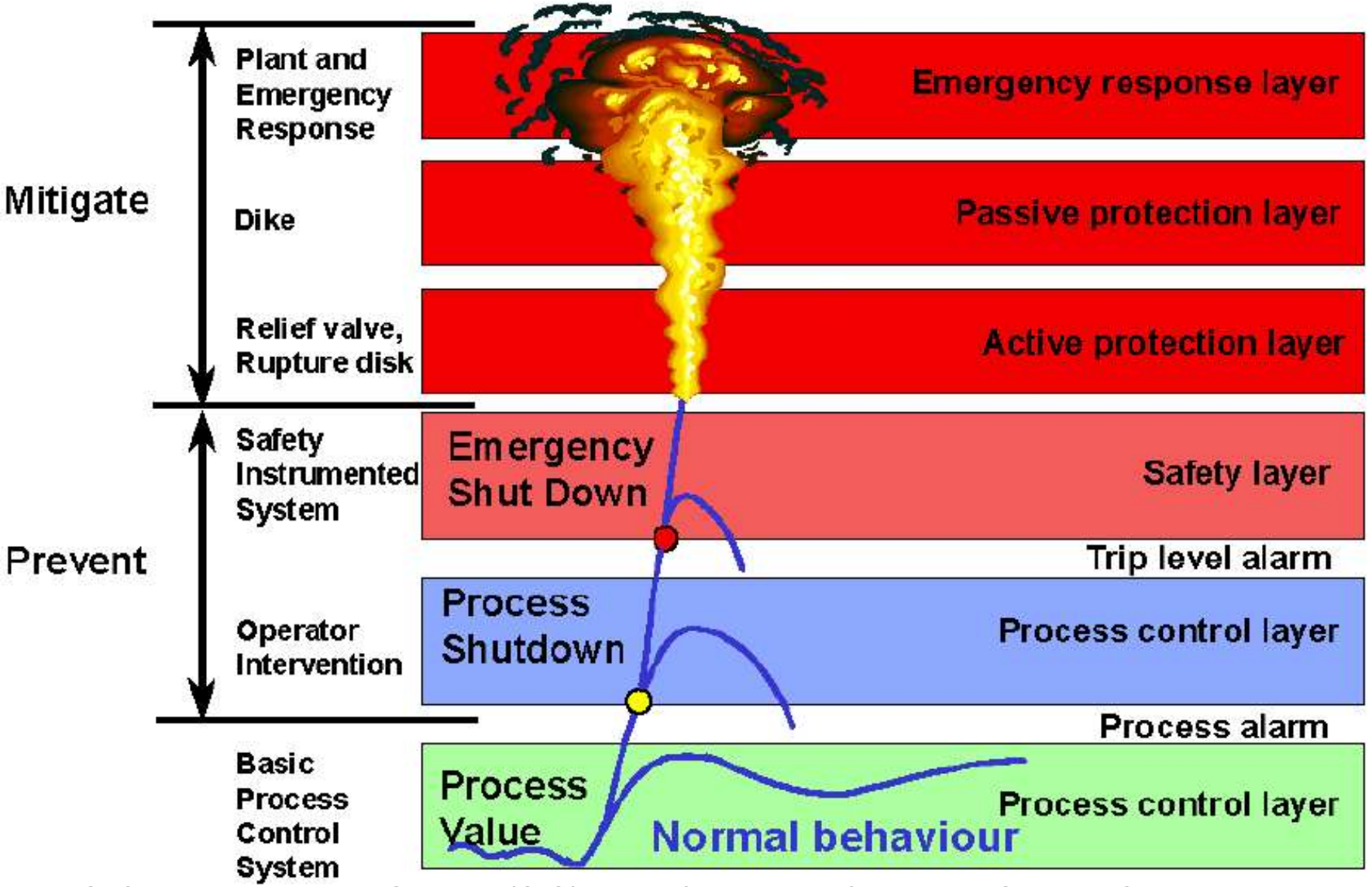


Both outputs provide half of the value.
If one of the modules fails, the active output delivers the complete value.



Safety Systems

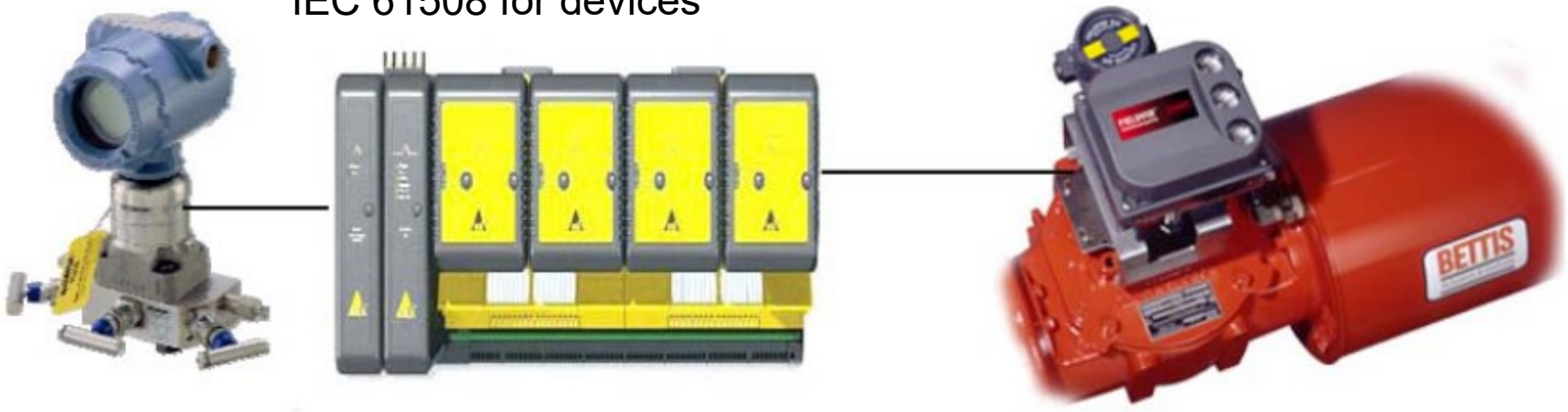
Layers of Protection



The Safety Instrumented System (SIS) is the final level after which System Control is beyond the power of the Control System Engineer and depends on Mechanical / Civil Designs for containment.

Safety Instrumented System (SIS)

Standards : IEC 61511 (ISA 84) for process / plant application
IEC 61508 for devices



Sensor Logic Solver Final Control Element

Typically Redundant Systems
Ability to self test Inputs and Outputs, by connecting to loop-backs

*"The **process control system** affects the size of your paycheck;*

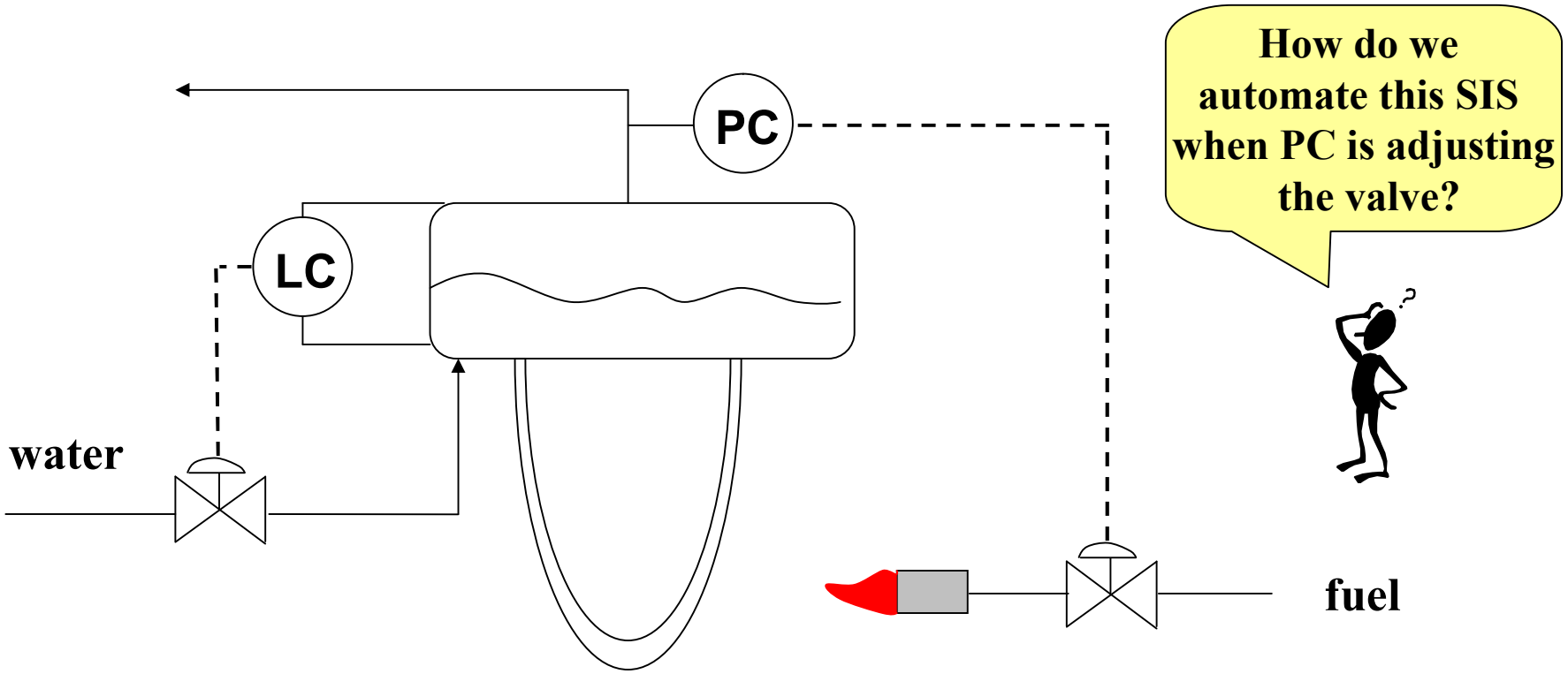
*the **safety control system** affects whether or not you will be around to collect it."*

-- Rinard (1990)

Safety Instrumented (Interlocked) System

- **The automation strategy is usually simple, for example,**

If L123 < $L123_{min}$; then, reduce fuel to zero

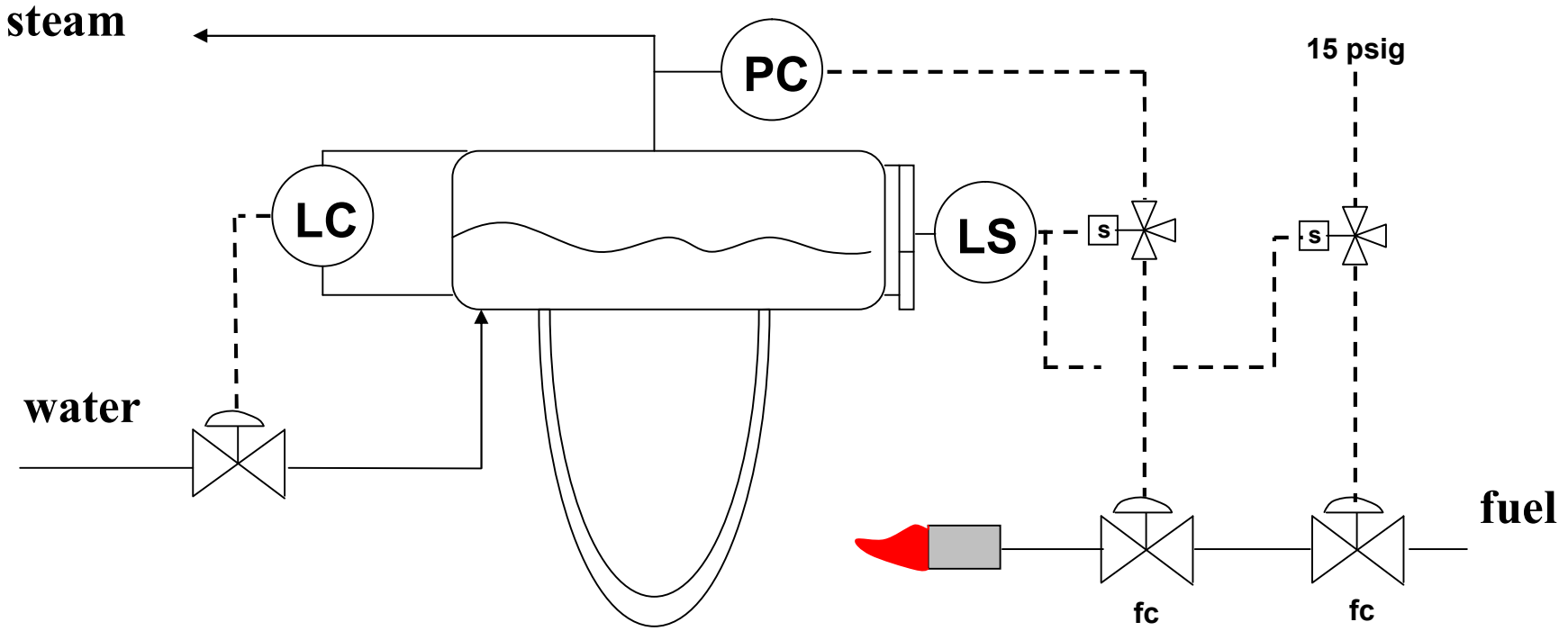


If L123 < $L123_{min}$; then, reduce fuel to zero

LS = level switch, note that separate sensor is used

= solenoid valve (open/closed)

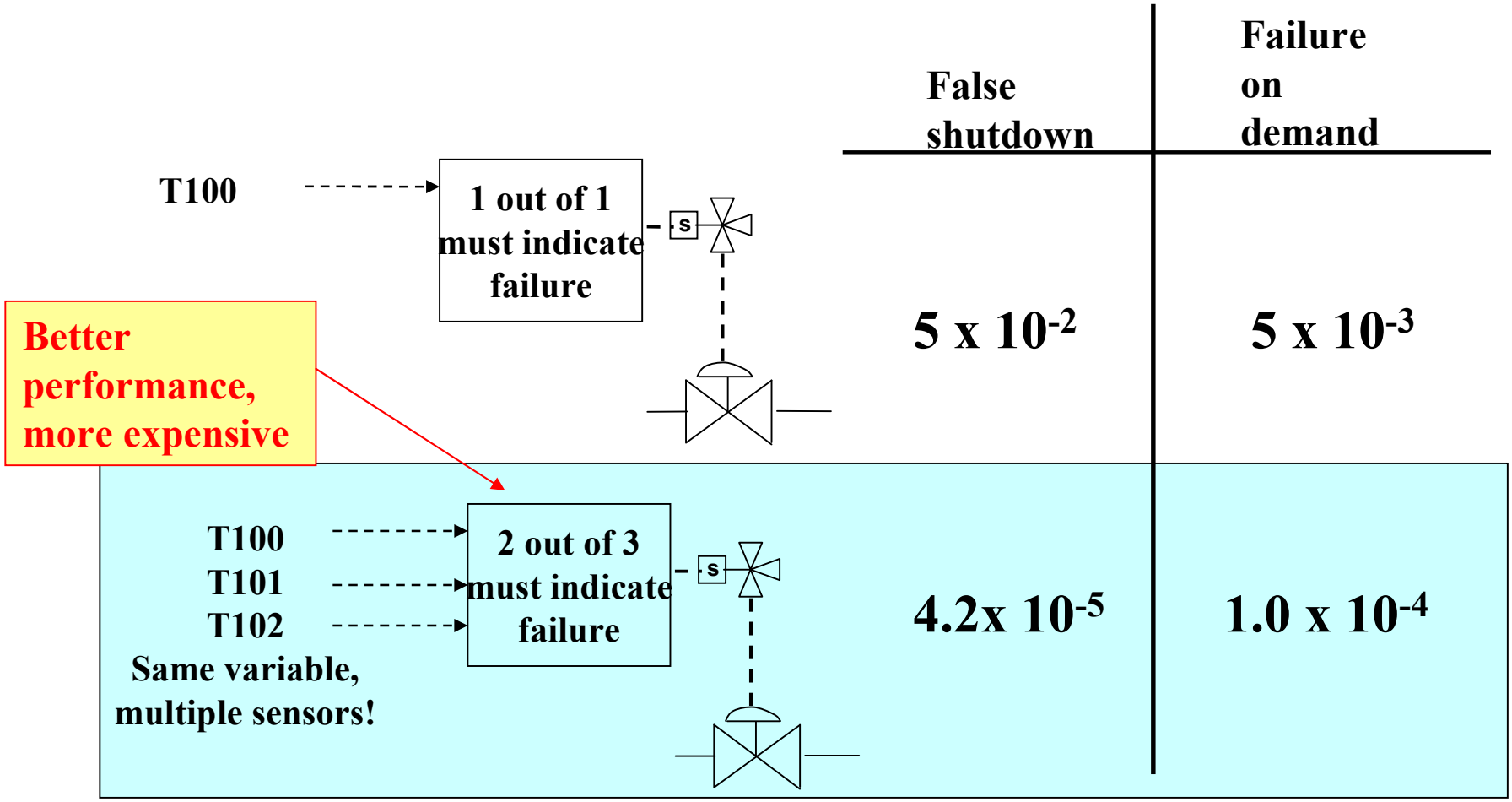
fc = fail closed



Note the “Fail Closed” valves. That fail to a safe state.

SIS

- **The SIS saves us from hazards, but can shutdown the plant for false reasons, e.g., instrument failure.**



RISK MATRIX FOR SELECTING Safety Integrity Level SIL

Event Severity	low	moderate	high
extensive	Medium 2	Major 3	Major 3
serious	Minimal 1	Medium 2	Major 3
minor	Minimal 1	Minimal 1	Medium 2

Event Likelihood

Table entries

word = qualitative risk description

number = required safety integrity level (SIL) ⟶

Safety Integrity Levels

(Prob. Of failure on demand)

1 = .01 to .1

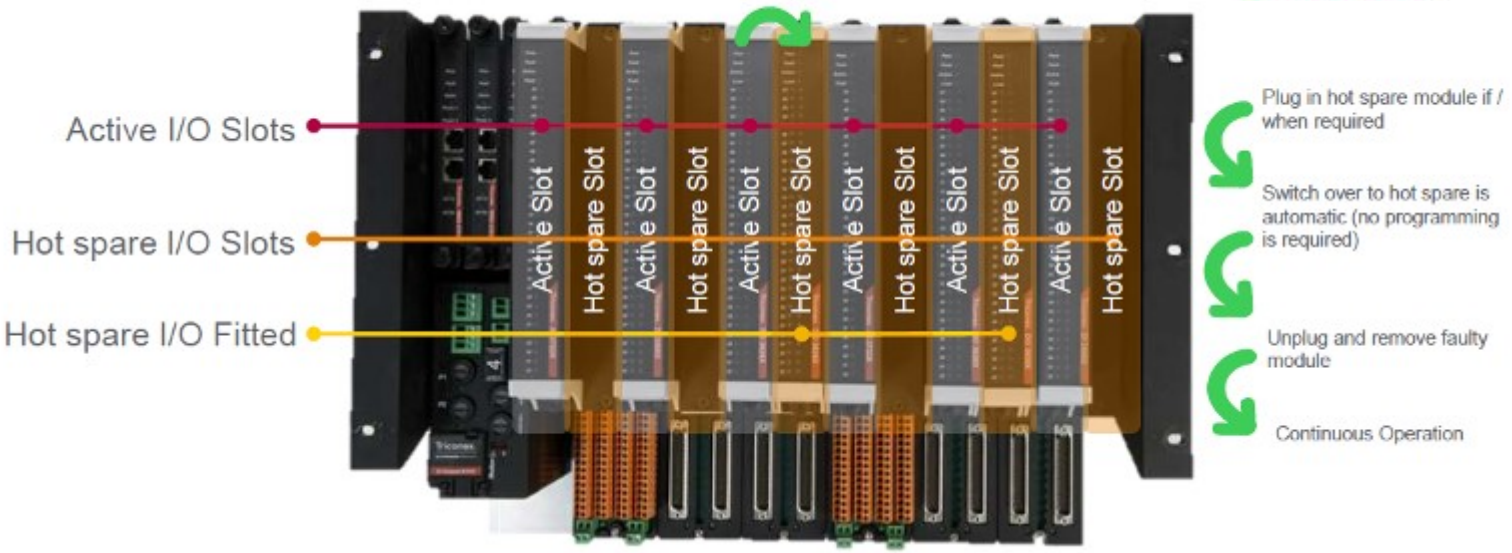
2 = .001 to .01

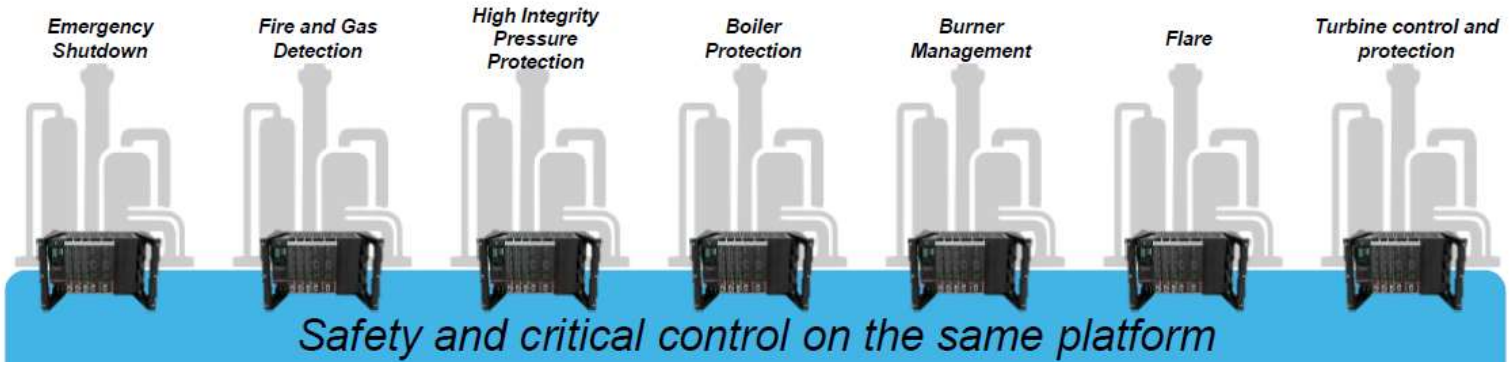
3 = .0001 to .001

Selection documented for legal requirements

Tricon CX – delivering continuous operation

Life Is On | Schneider Electric





Triconex General Purpose System
TÜV Certified SIL2
Fail Safe, Fault Tolerant
Central and distributed installation
Broad range of I/O modules

Trident
TÜV Certified SIL3
Fail Safe, Fault Tolerant
Central and distributed installation
Broad range of I/O modules

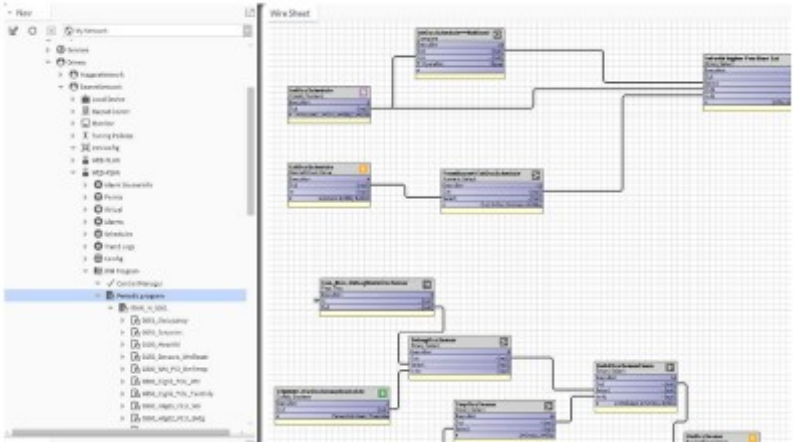
Tricon (Nuclear)
TÜV Certified SIL3
Nuclear 1E Approved

Tricon and Tricon CX
TÜV Certified SIL3
Fail Safe, Fault Tolerant
Choice of form factors

Allied Fields
Building Automation

Building Automation Systems

- Protocols : OPC, BACnet, LON, Modbus, SNMP, KNX, DALI, Zigbee
- BACnet Controllers support
  - Node discovery and browsing
  - Configuration
    - Node, IO
  - Programs
    - Periodic Functions
    - Event Driven Functions
      - Time-of-day, Input state
    - Access to IO on other nodes for sensing and control



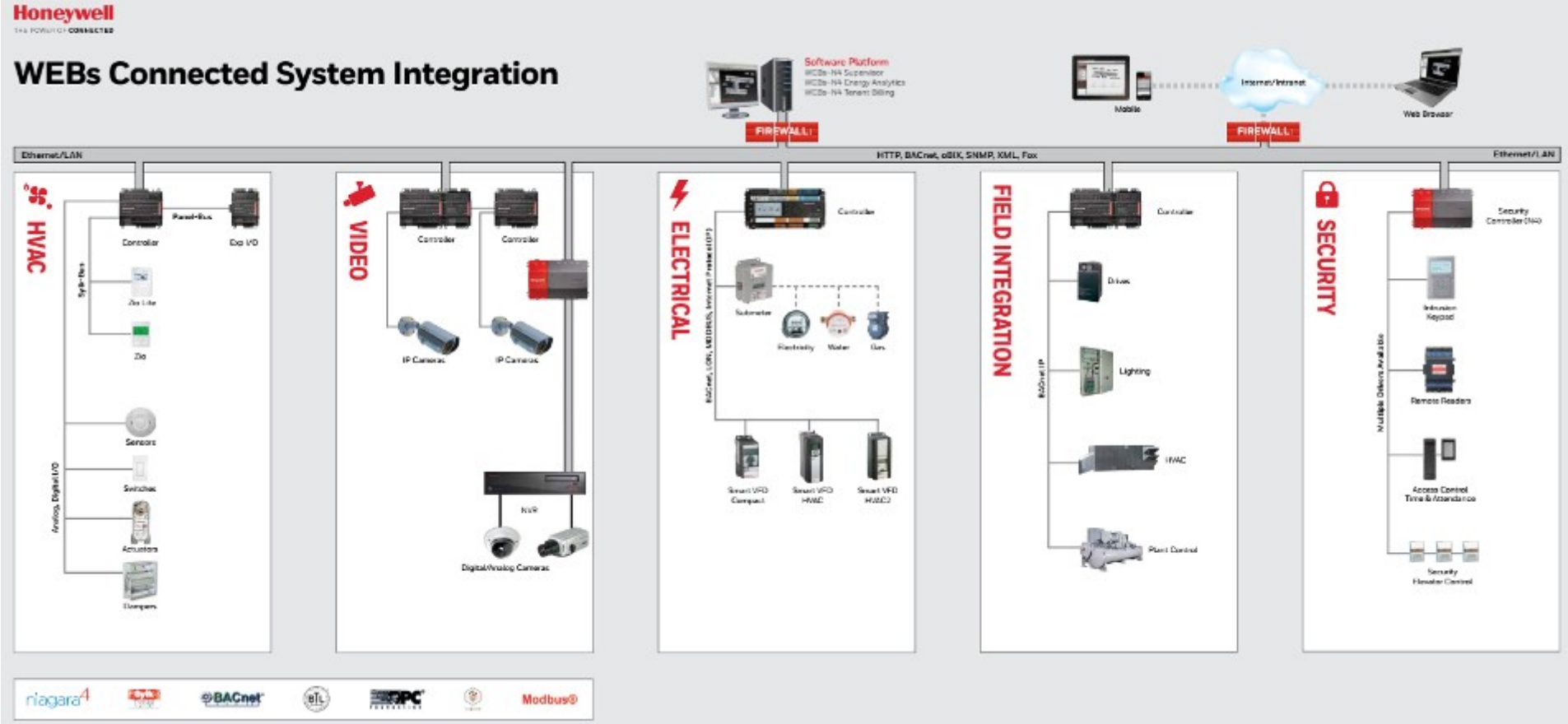
Johnson Controls

Life Is On Schneider Electric

Honeywell

Building Automation Systems

Zoom to see details

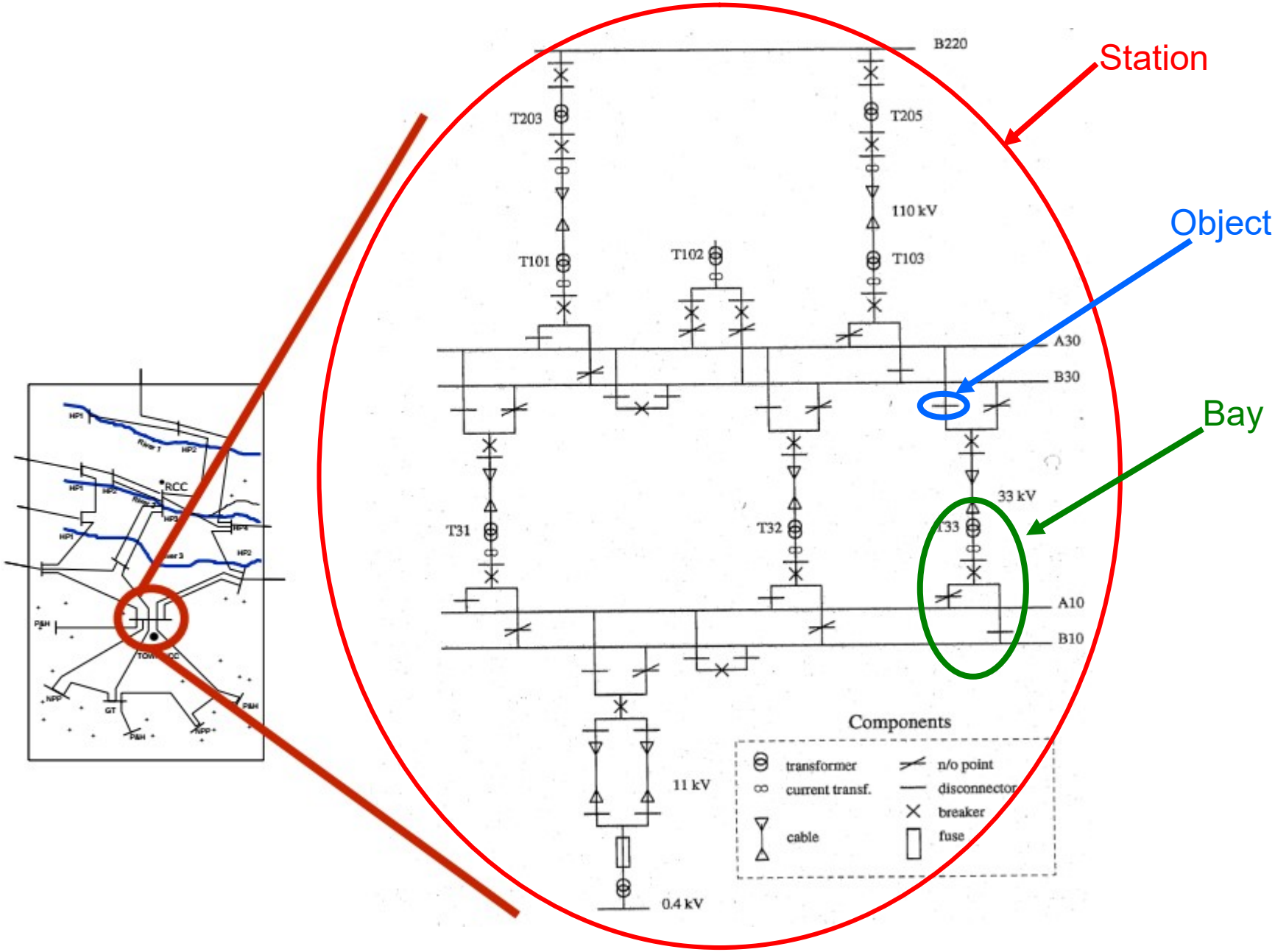


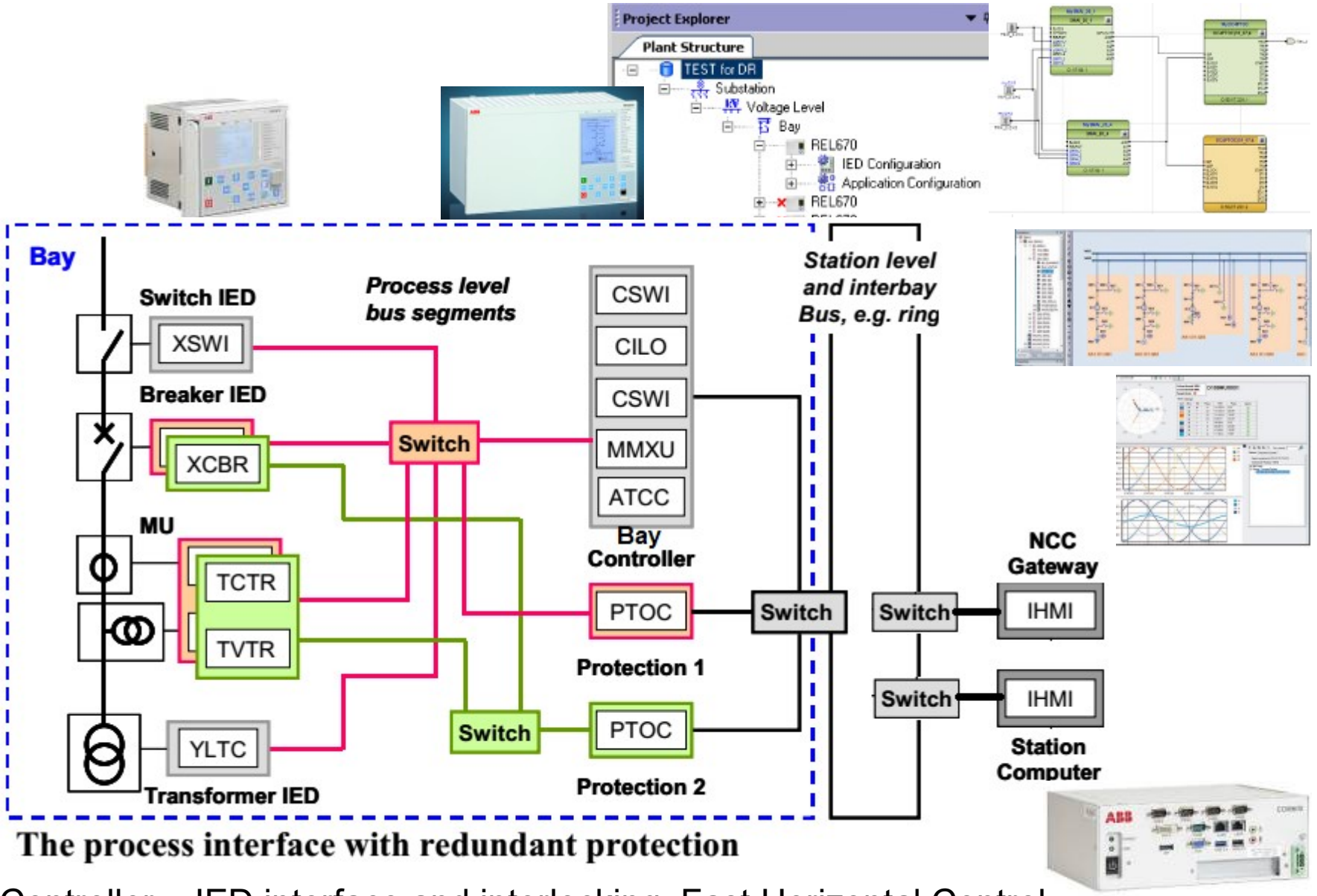
- The Web-8000 Series WebController (on the extreme right – in the security block) enables internet access for configuration and user operation
- During on-line configuration the engineer can browser to discover Controllers on the network. The IO points of one controller can be referenced by another controller.

Allied Fields
Substation Automation
&
Electrical SCADA System

Substation Automation

- Protocol – IEC 61850. IEC60870, Modbus, DNP3, ICCP
- Intelligent Electronic Devices (IED) for control, protection, monitoring, metering
- Specialized hardware rather than general-purpose PLCs
- Programming – Step-wise with Wizards and Specialized Function Block Programs

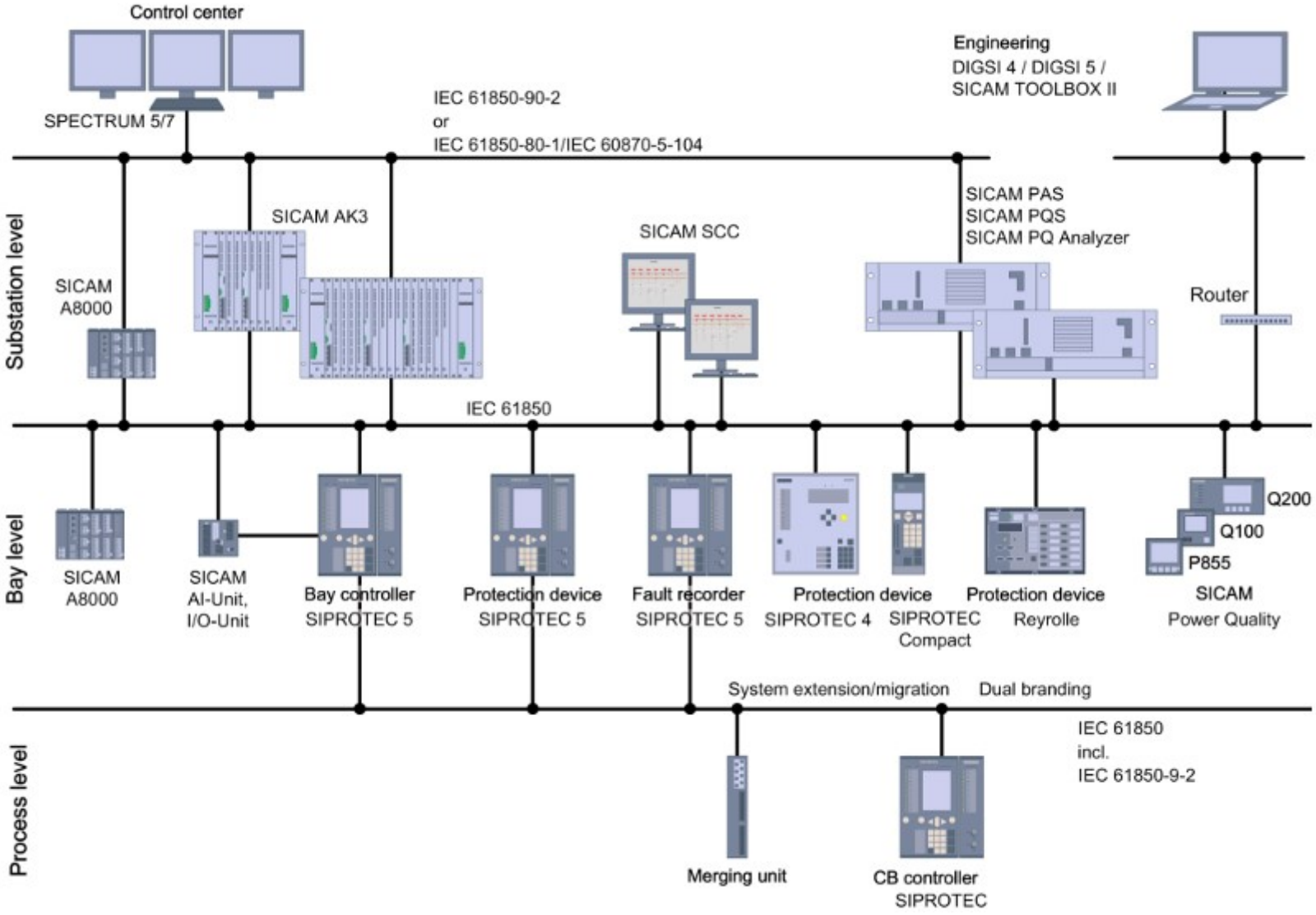

Station
Object
Bay
B220
T203
T205
110 kV
T101
T102
T103
A30
B30
T31
T32
T33
33 kV
A10
B10
11 kV
0.4 kV
Components
transformer
current transf.
cable
n/o point
disconnector
breaker
fuse



The process interface with redundant protection

- Bay Controller – IED interface and interlocking, Fast Horizontal Control
- IEDs – Protection and Switching
- Station Computer – Station-Level Control, HMI and Configuration
- Network Control Center Gateway – Enables Remote Control at Grid Level

Siemens Sub-station Automation Hierarchy



Electric SCADA Functionality

Intelligent Devices
Feeder Automation
Substation Automation
ETAP Microgrid Controller Distributed Resource
ETAP Energy Management System
ETAP Automated Fault Analysis System
Fleet / ERP
Interactive Voice Response (IVR)

REAL-TIME NETWORK

DMS
Load Flow Analysis
Short Circuit Analysis
Fault detection and location automated switching
Overload reduction Switching
Volt / Var Optimization

SCADA
Data Acquisition
Data Archiving
Monitoring & Event Processing
Supervisory Control & Interlocking
Calculation & Reports
Human Machine Interaction
Inter-Center Communication

OMS
Outage Analysis
Crew Management
Trouble Call Management
Operations Management
Switch Order Management
Referral Work Orders

ADMS
Advanced Distribution Management System

Visual & Reporting
Web Portal
Dashboards
HMI

Mobile
Work Orders
Vehicle Tracking
Outage Schedule

Workforce Management
Dispatch
Crew Management
Dynamic Scheduling
Auto Dispatch

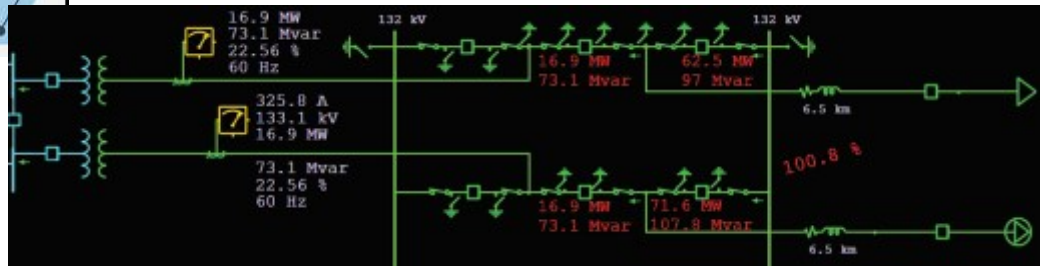
ENTERPRISE NETWORK

Call Management — CIS
Work Order — WMS
AMI / MDM — Smart Meters
Network Model Management — GIS
Energy Market
Enterprise Asset Management
Enterprise Resource Planning

Unified Smart Grid Management Architecture

etap Powering Success

Storm Assessment
- Planned Outage
- Damage Location

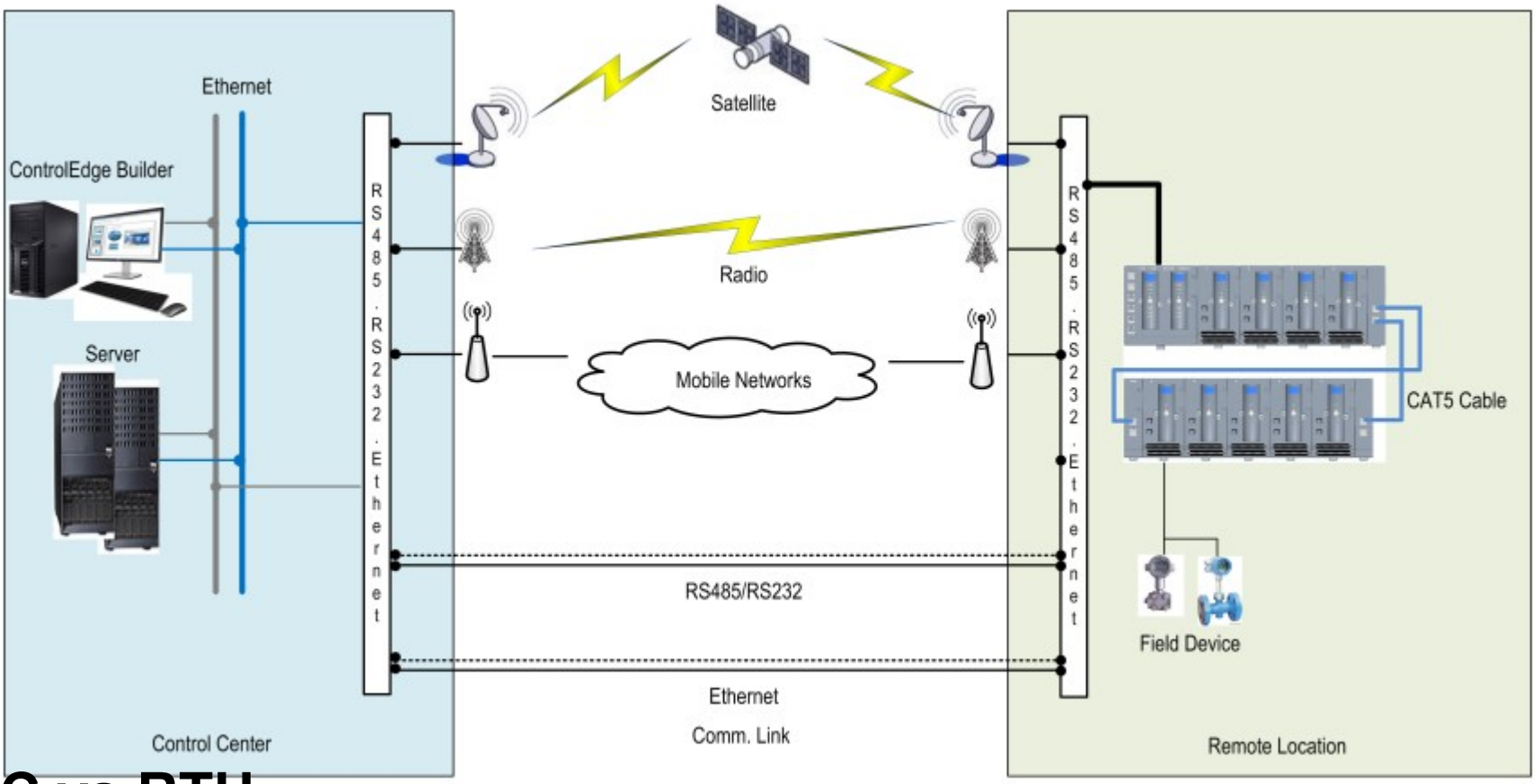


Allied Fields
Telemetry, Telecontrol, Teleservice

RTUs
Remote Terminal Units

- Used for geographically distributed applications
  - Oil, Water Pipelines, Canals, Street lights, Traffic Lights
  - Electrical Power Grid, Wind/Solar Generation
- Have inherent battery-back-up, redundant power supply options, solar power options, low power consumption, wide operating temperatures and rugged construction to survive for extended periods in **harsh conditions at remote unmanned locations**
- Typically support fewer IO points per node – in the tens
- Historically had limited and simple logic compared to PLC
- Modern RTUs however may support programming using IEC 61131-3 languages like Ladder, Function Block, STL, SFC
- Communicate with DCS or SCADA base-station using wired or wireless communication
- Inherent support to communication using wireless technology like 2G, 3G, LTE, 4G, Radio, Satellite Link
- Support Application Layer protocols like Modbus RTU, DNP3,
- Typically have buffering capacity to store logged data and transmit in bursts to the base-station
- Network Time Sync of system clock, Local Time Stamping
- Extensive Remote Diagnostics and Control - Modern RTUs may have embedded Webserver, Email, SMS, File Logging and FTP server

RTU Remote Communications

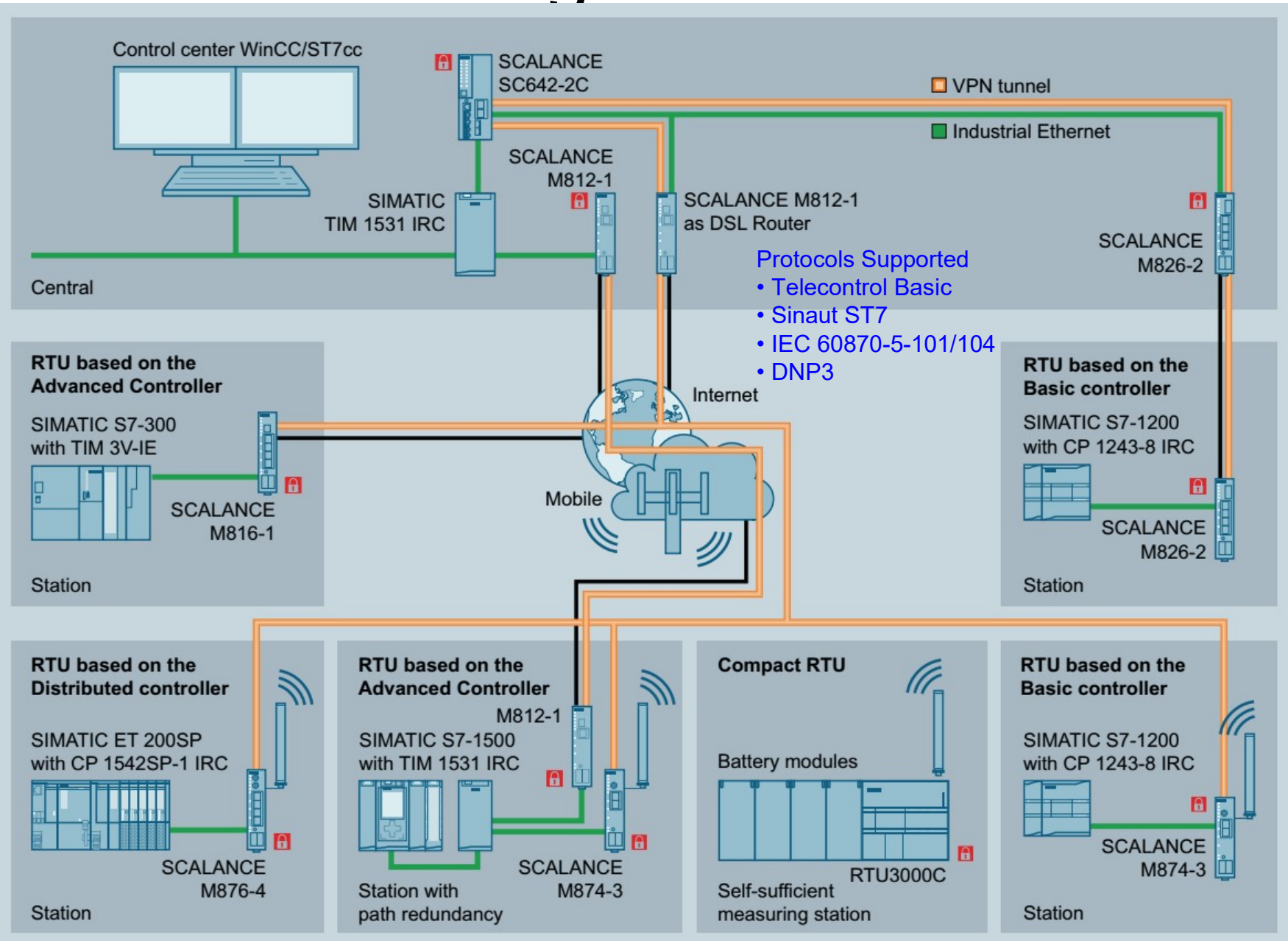


PLC vs RTU

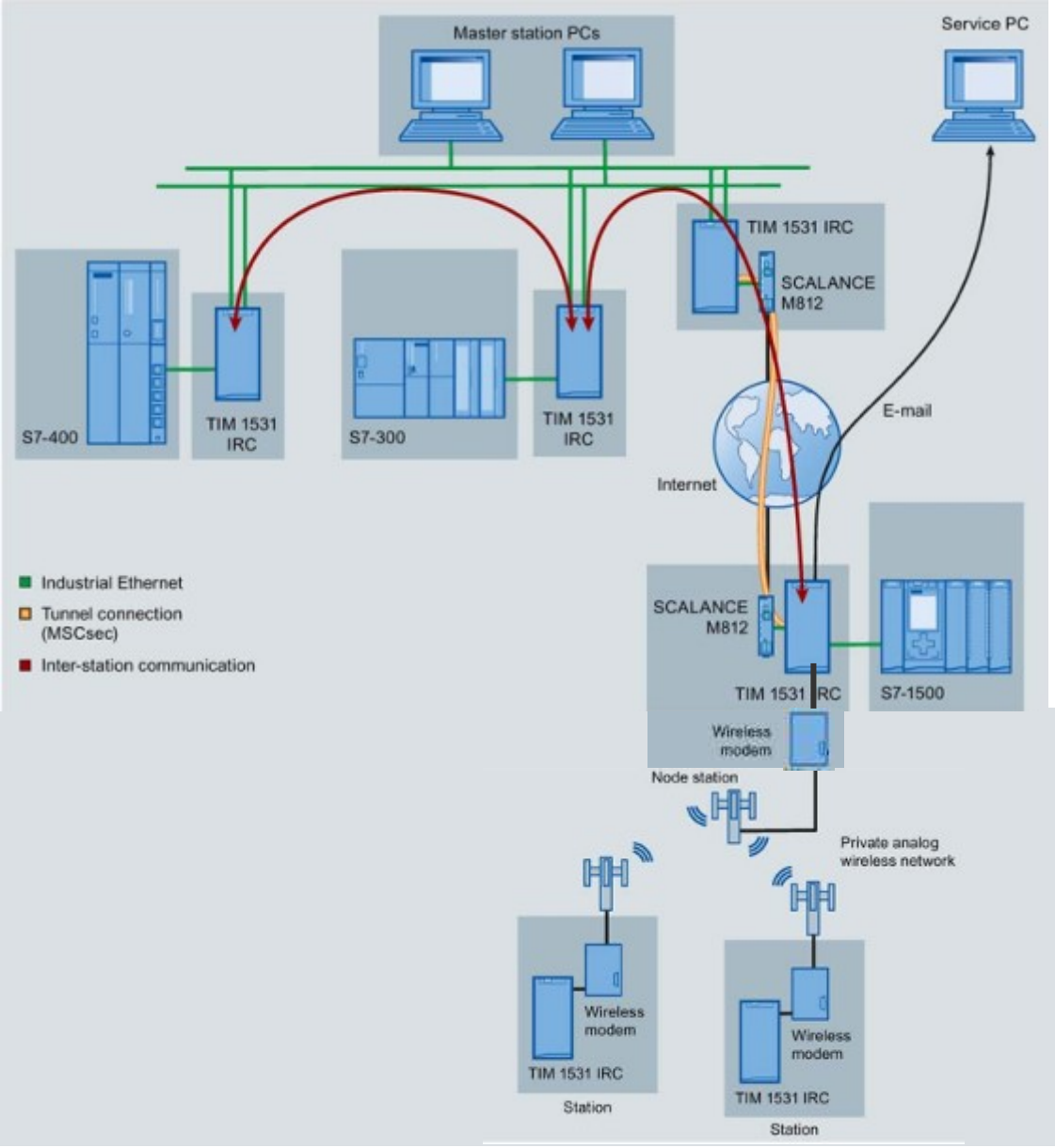
Modern general-purpose PLCs paired with suitable UPS, Modems and special environment-proof enclosures ***could be*** used in place of RTUs … ***but*** … the RTUs are built and certified specifically for these applications so makes sense to use them here.

Some Vendors (Siemens) may use PLC hardware as RTUs

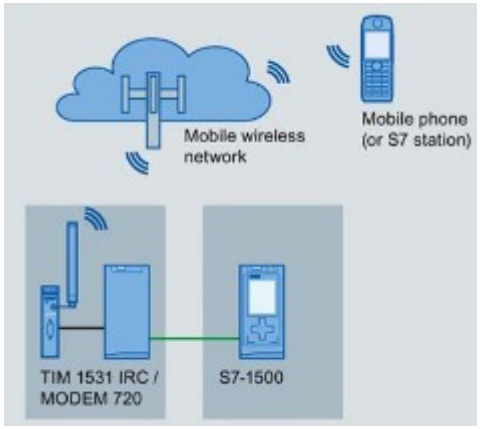
Wide Range of Networks



SMS, Email, Radio



Communications can be set-up by configuration of the Communication Modules with very little programming effort



Allied Fields
DCS

What is a DCS?

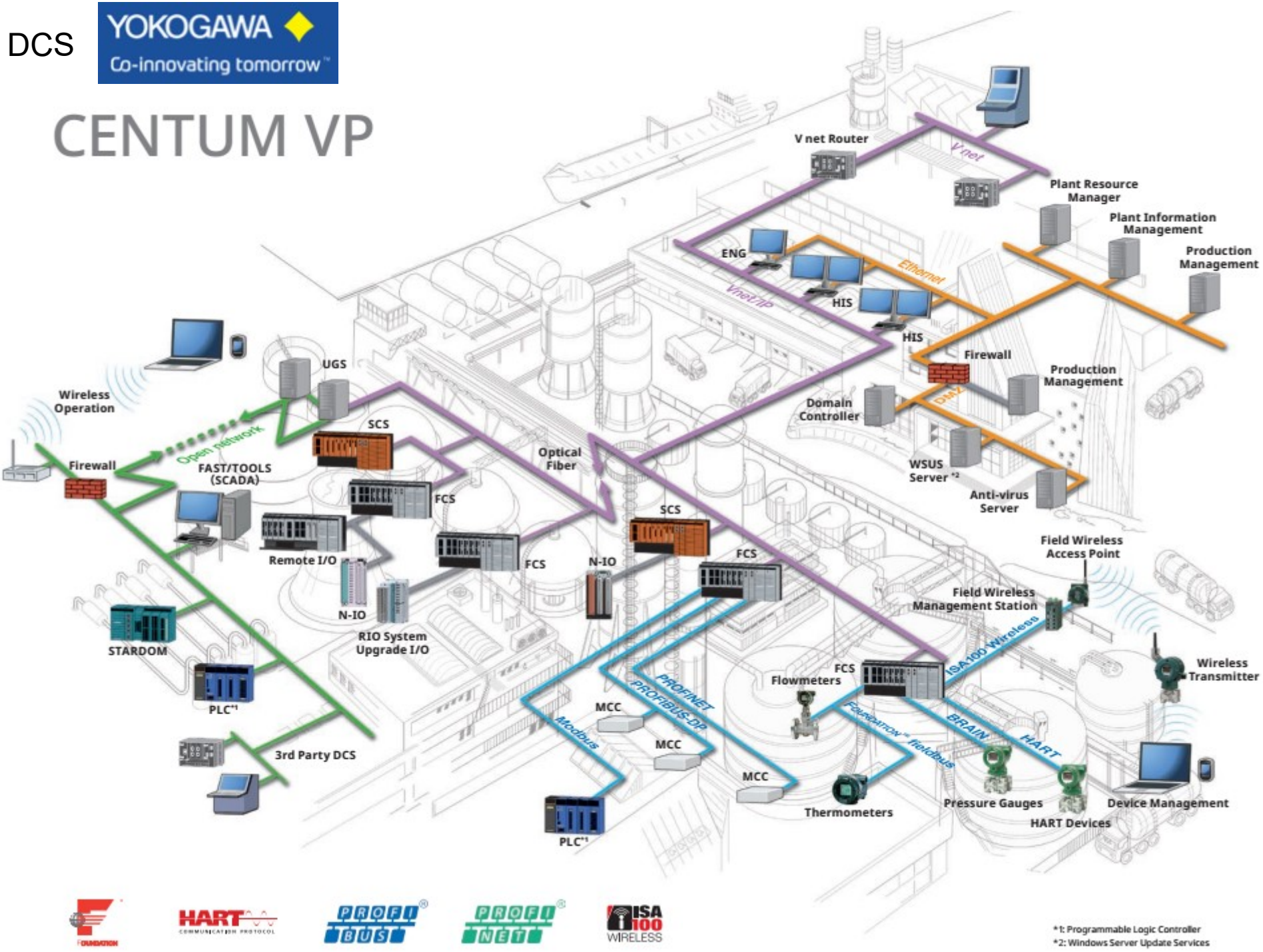
- The classic DCS originated from an overall system approach
- A DCS is a **complete tightly integrated package from a single vendor** consisting of
  - Hardware
    - Process Controllers, including redundant controllers & IO and other certified safety systems
    - Networking - wired, fibre-optic, wireless, mesh, redundant
    - Server & PC hardware, including industrial PCs, redundant drive systems, multi-monitor systems
  - Software
    - Engineering and Configuration Environment with in-built Change Tracking and Audit Trails
    - **Single Data Model – One data owner – System-wide Tag.**
      - Example : Tag defined in Process Controller with HMI Alarming, History Logging is immediately available with HMI and Historian
    - Visualization
    - Batch Support at Controller-level
    - Process Database
    - System-wide Modelling and Simulation
- Multiple Process Controllers autonomously control large sections of the plant and they can inherently communicate with each other.
- The entire system is tested, certified and guaranteed to handle high-speed data communications between the nodes
- Coordination, synchronization and integrity of process data over a high-performance and deterministic network are at the core of the DCS architecture
- 20-30 year Vendor guarantee for entire system support

ABB YOKOGAWA Co-innovating tomorrow Schneider Electric SIEMENS Ingenuity for life Honeywell EMERSON

DCS

YOKOGAWA
Co-innovating tomorrow™

CENTUM VP



*1: Programmable Logic Controller
*2: Windows Server Update Services

PLC vs DCS

- Historically the DCS was designed for
  - integrated control of entire scope of large process plants with millions of IO points
  - Guaranteed network performance for communication between the distributed Process Controllers and also with the HMI, Historian, etc..
  - Redundancy and high availability
  - Large Number of Analog Control Loops – programmed with Function Blocks
    - Standard Function Blocks (PID, etc), Fuzzy Logic, Advanced Algorithms and Visualization Templates(Refining Column, etc) are available out-of-the-box in the DCS software
    - New Function blocks can be created and reused
    - Of-course Digital IO is also supported!
    - Configured and maintained by Trained Engineers
  - Guaranteed fixed interval scan. *Example : @0.1s @1s @5s*

- Historically the PLC was designed for
  - Local control with a few 100 IO points
  - Stand-alone operation
  - Low Cost and Simplicity
  - Relay logic Replacement – programmed with Ladder Logic
    - Initial emphasis on simplicity to enable even electricians to understand
  - Ladder Logic is continuously executed cyclically
    - Fast response guaranteed below the Scan Watchdog Time (say 0.1 ms)
    - Fast but non-deterministic scan scheduling interval – sometimes say every 50ns, other times 52 ns…

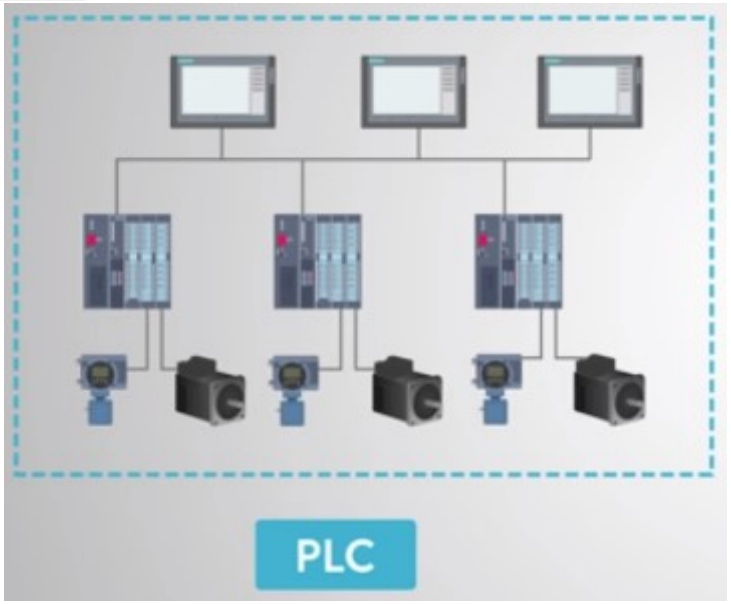
- Many machines ship with built-in PLCs for their control and may be installed in a plant that is controlled by a DCS. The DCS may communicate with these PLCs using IO or a field-bus.

PLC vs DCS

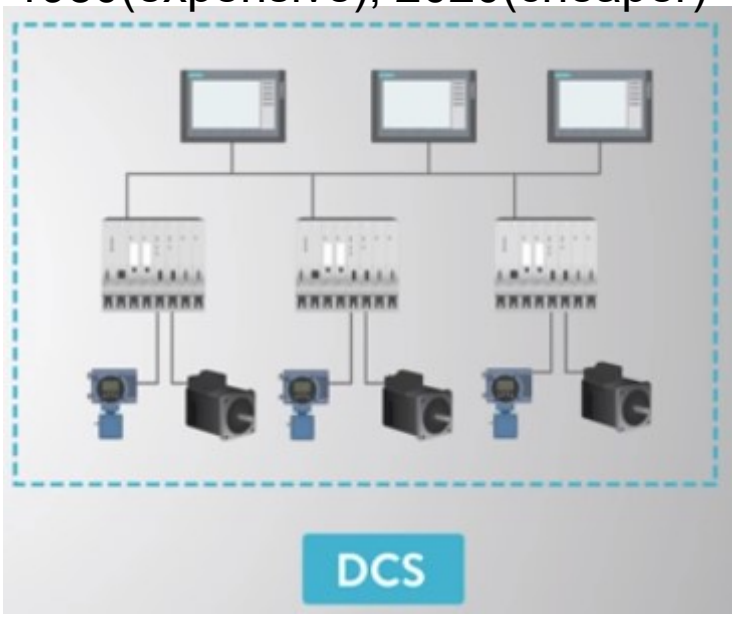
https://realpars.com

1980

2020



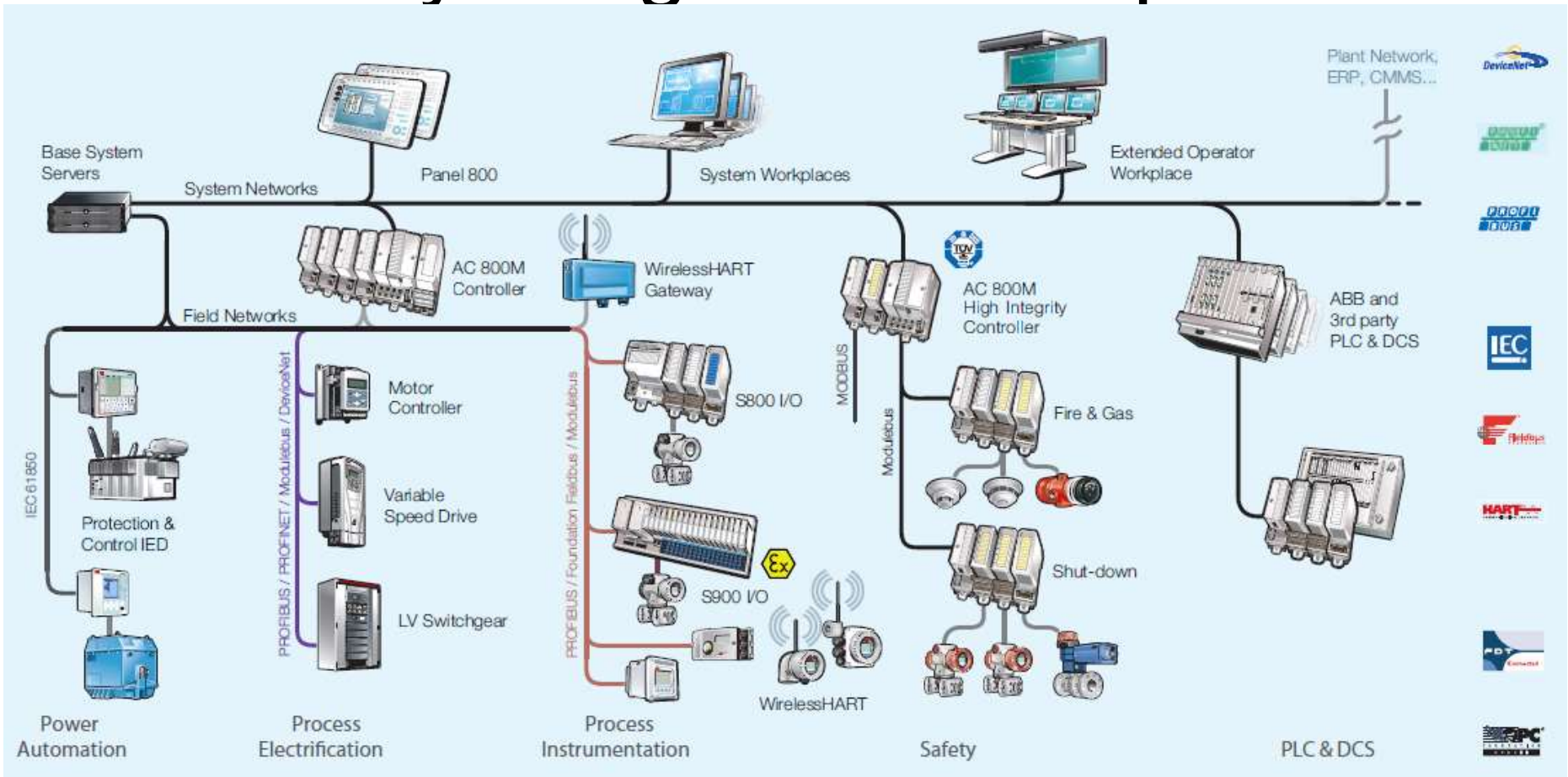
1980(expensive), 2020(cheaper)



PLC vs DCS
Today – PLCs come close.. but..

- PLCs **today** support
  - Networking using standard Field Buses *(but through-put and reliability may not be guaranteed for your system configuration)*
  - Function Block Programming
  - Redundancy *(but at cost comparable to a DCS)*
  - Configurable Scan Intervals
  - PLC Tag data-base integrated with SCADA of the same vendor (or uploaded to SCADA for a different vendor – here syncing changes could be an issue)
- Using PLCs, SCADA, Historian… a system comparable to a DCS could be built
- But… using networked PLCs+SCADA as a DCS … is very much a "do-it-yourself" (DIY) approach with plenty of technical risk as well as added costs that are not always immediately obvious.
- Today due to significant price reduction in DCS systems…it still makes good Engineering and Economic sense to use a DCS than attempt to *"club together"* your own system using networked-PLCs and SCADA
  - For distributed systems with large IO counts
  - Proven DCS applications – refineries, chemical processes
  - Where long support guarantees are required for the system – and individual PLC and SCADA vendors may not provide 20-30 year support guarantees
- Several DCS Systems of today actually use PLC hardware and SCADA software with some tweaking. However here the integration is done by the equipment manufacturer and a pre-packaged system tested and certified for the IO Counts and data-transfer rates is supplied. Example Siemens PCS7 DCS that uses S7-400 PLCs and WinCC HMI, Rockwell's ProcesLogix

ABB System 800xA AC800M -Fully Integrated Enterprise

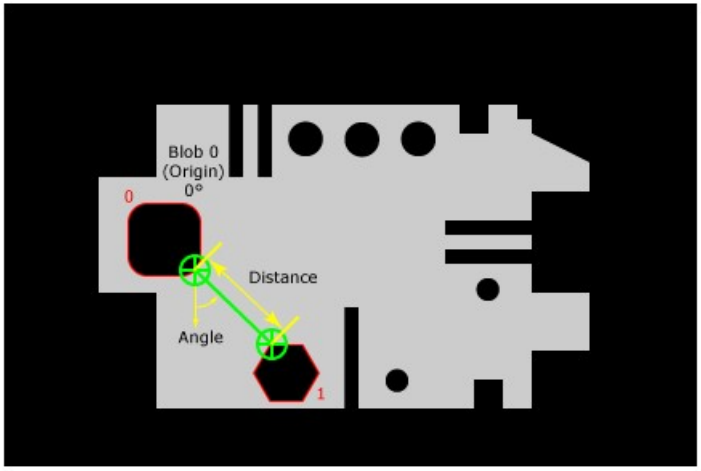


With System 800xA, users enjoy a fully integrated enterprise where PLCs, DCS systems, safety systems, electrical systems, live video, maintenance systems, ERPs and more all work together in an seamless environment. The result is improved plant visibility and fast access to relevant information in real time, allowing operational issues to be resolved before downtime occurs.

Machine Vision Interface

Machine Vision Components

- Digital greyscale or color camera to acquire images
- Imager to digitize them
- Processor to run the programmed image analysis
- Input/Output hardware and communication links to report results and send/receive messages from other devices, such as an HMI, PLC or Robot.
- Optional components include:
  - Object-sensing synchronizing sensor (usually an optical or magnetic sensor) that triggers the vision system to acquire and process an image.
  - Actuator to reject or sort parts based on the results o f the image analysis.
- Machine Vision Programming Software



COGNEX

Machine Vision System Programming

- **Functions**
  - **Inspection**: Presence/Absence, Defect Analysis, Pattern Matching
  - **Identification**: Colors, Data Codes, Lot Codes, Serial Numbers, Text
  - **Measurements**: Gauging, Dimensions, Counting, Locating
- Programming a vision system involves the following:
  - Teaching the vision system what to look for, based on distinctive "features of interest" (also known as features) in the image.
  - When to take a picture, and how best to illuminate the part.
  - How to communicate the results of its analysis, and the actions to take, based on the results.

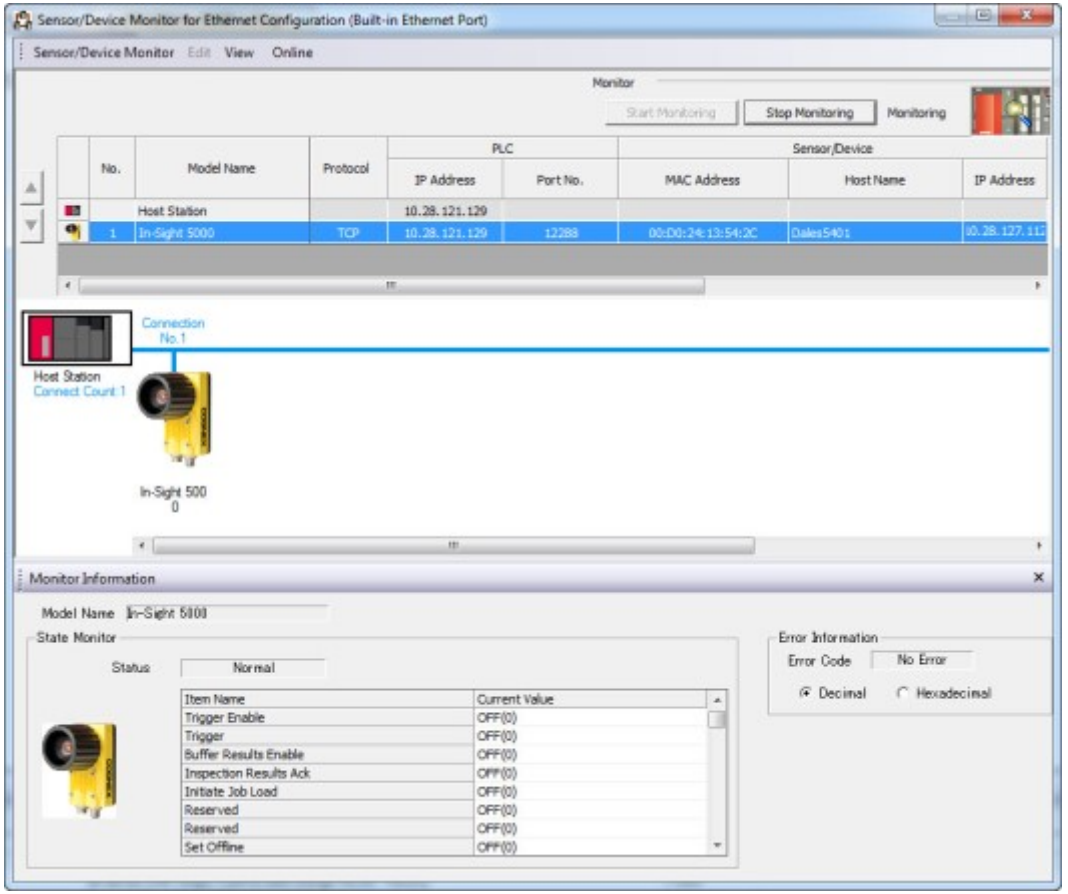
PLC communication with the Machine Vision System - Configuration

PLCs may enable browsing Cameras on the network and quick set-up of map between the Camera parameters and PLC registers

Communication is over Ethernet or Serial :

Using Custom Messages

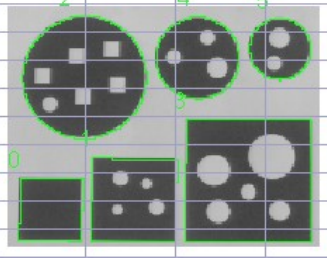
OR Protocols like Modbus, FTP, SMTP, Ethernet/IP, Melsec, Profinet, PowerLink



PLC communication with the Machine Vision System - Runtime

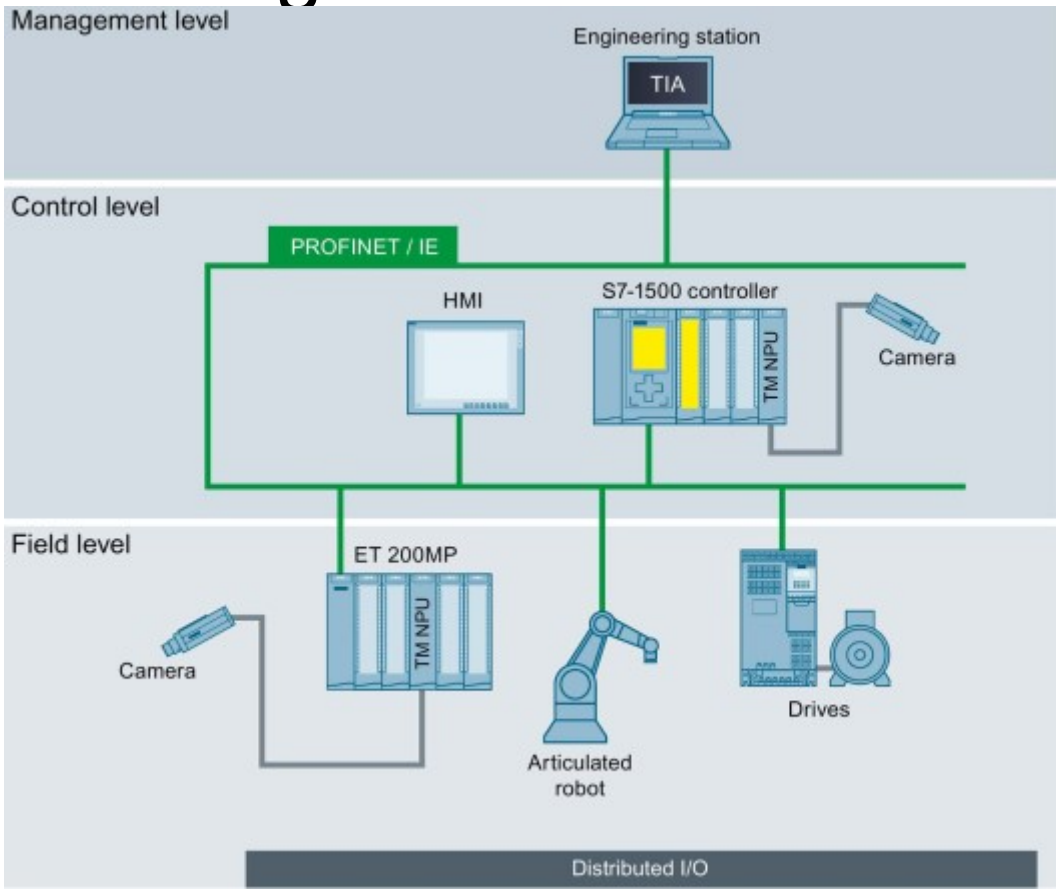
- PLC triggers the Machine Vision System which captures and processes the image and generates the results
- Machine Vision System sets PLC register to indicate completion of image processing
- The results are written to / read by the PLC system into PLC registers

	Index	Row	Col	Angle	Color	Score	Area	Elongation	Holes	Perimeter	Spread
Blobs	0.000	297.087	262.208	-26.328	0.000	100.000	1057.000	0.000	0.000	123.000	0.168
	1.000	292.623	305.169	-36.801	0.000	100.000	1768.000	0.000	4.000	164.000	0.186
	2.000	228.483	279.699	57.403	0.000	100.000	2865.000	0.000	6.000	199.000	0.185
	3.000	282.991	364.814	37.764	0.000	100.000	3328.000	0.000	5.000	243.000	0.234
	4.000	218.453	338.029	-66.475	0.000	100.000	1277.000	0.000	3.000	134.000	0.189
	5.000	213.722	381.447	10.766	0.000	100.000	675.000	0.000	2.000	99.000	0.209



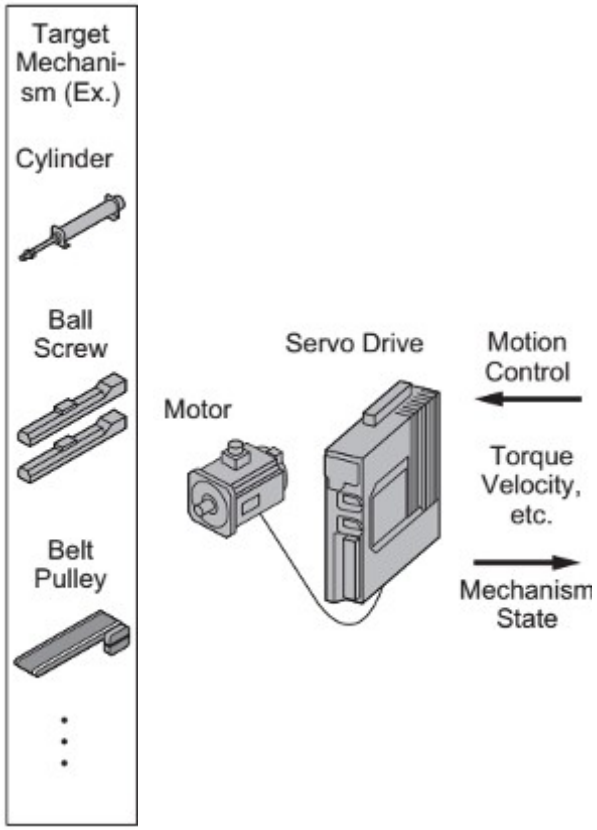
PLC & AI

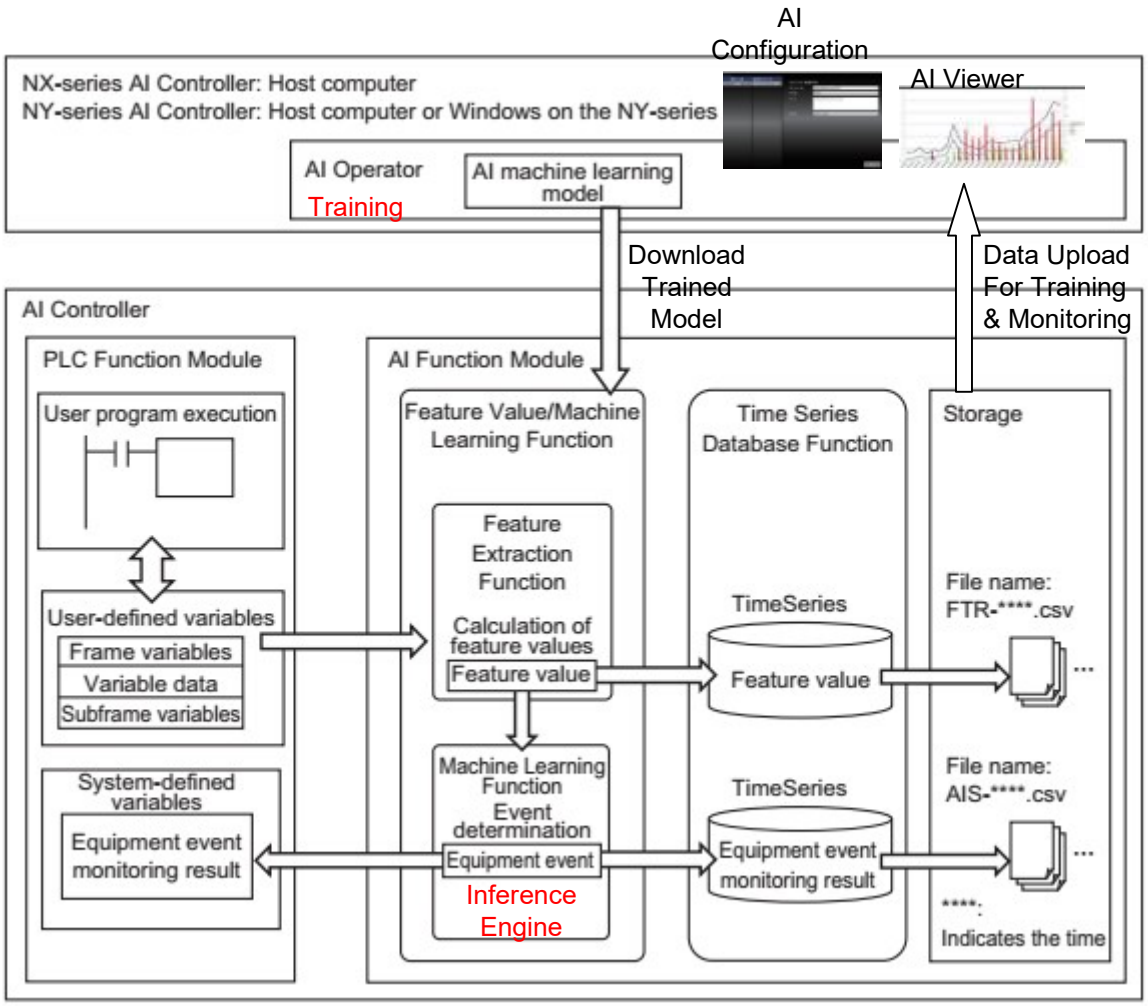
Siemens S7-1500/ET200 NPU
Neural Processing Unit



- NPU can input data (Camera images / files) from its USB, Ethernet ports or the PLC CPU NPU processes data through its Neural networks and passes the result to the PLC CPU
- Uses Intel Myriad X AI Accelerator
- Works with NNs designed and trained with Tensorflow and Caffe ML Frameworks
- Suitable for Image Processing using CNN, Audio Processing using RNN
- Applications – Vision controlled Motion, Sorting, Pick & Place

AI-Based Preventive Maintenance





OMRON

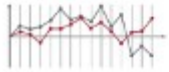
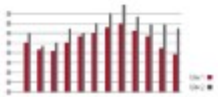
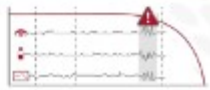
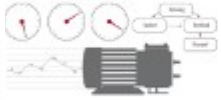
NX701-Z□□□
NY532-Z□□□
NY512-Z□□□

- AI Inference on the PLC Rack
- Inference Engine on the AI Processor analyses Field Signals from the PLC CPU for patterns that suggest performance degradation

Developing and using analytics

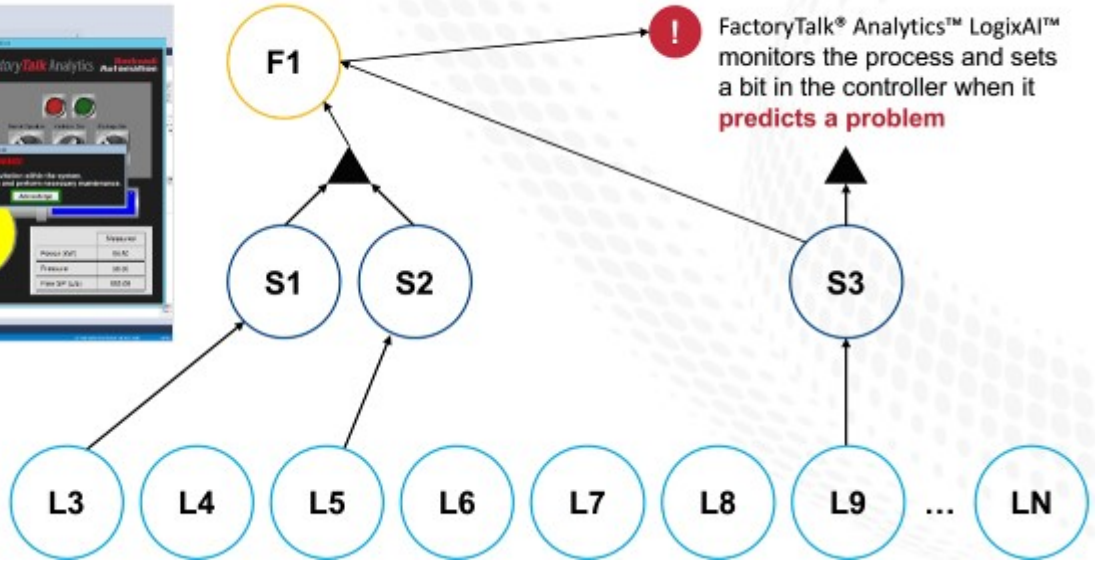
Scalable analytics

Where FactoryTalk® Analytics™ LogixAI™ fits into the ecosystem

ENTERPRISE	 What plant performed the best?	 Why is site A throughput below plan?	 Will I meet plan today? Tomorrow?	 How can I change operations to improve profitability? Yield? Quality?
SYSTEM	 Is Line 1 running ok?	 Why is Line 1 quality poor?	 I predict that Line 1 quality is moving out of tolerance.	 What action helps the operator to avoid poor quality?
DEVICE	 Am I running ok?	 Why did a fault happen?	 I predict a fault will happen soon.	 What action helps to avoid the fault?

FactoryTalk® Analytics™ LogixAI™

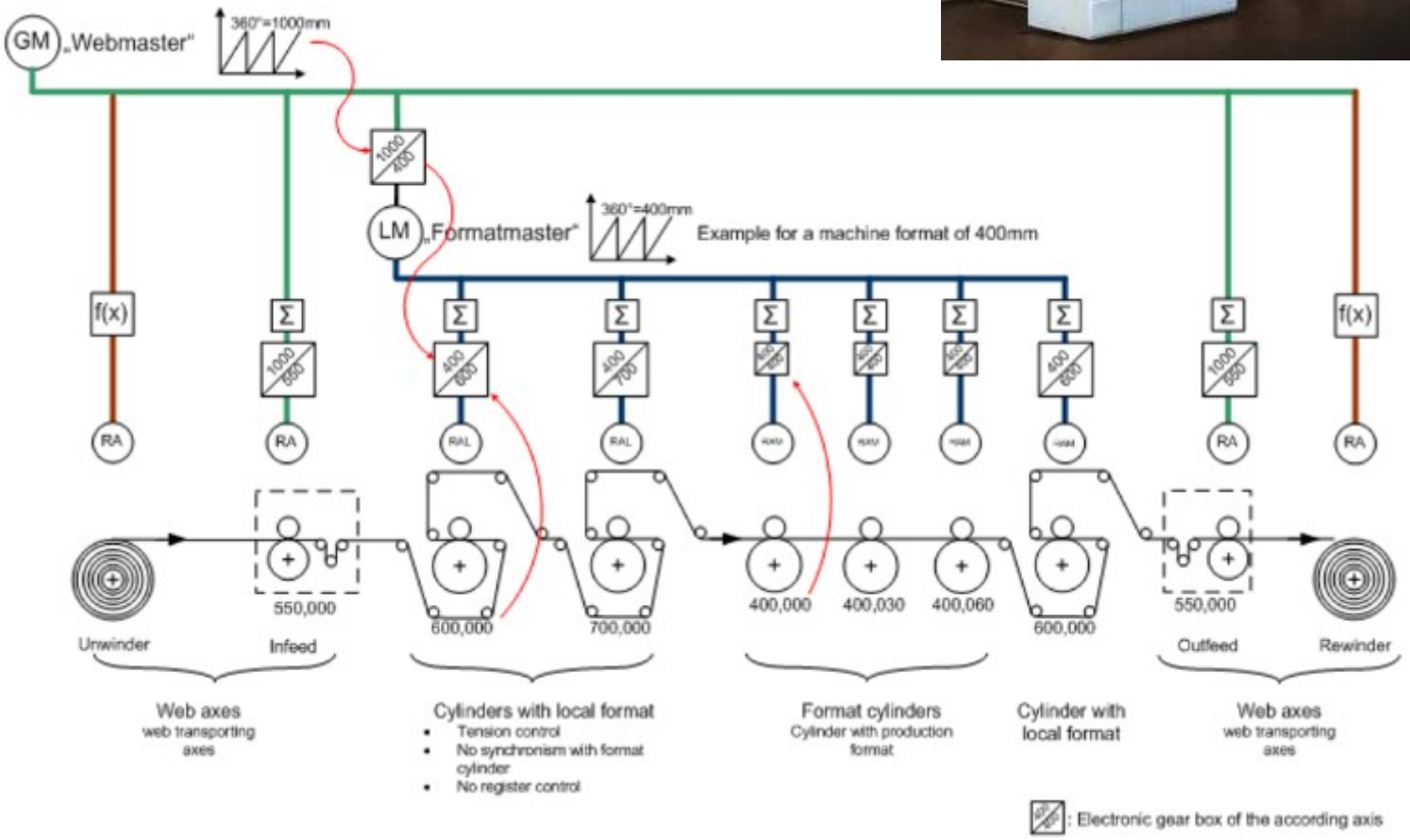
- Anomaly Detection
  - Create a model of normal operation. Detect change in the value of system parameters
    - Example – Chem X is produced by mixing Chem A,B,C in certain proportion, so variation in the flow of A beyond certain limits will be flagged
- Soft Sensor
  - Use data that exists in the controller to estimate data that does not exist in the controller
    - Use inputs of easily measurable system parameters to estimate system parameters that are difficult to measure. Example – sharpness of a cutting knife estimated from cutting time, drive current during the cutting operation, dimensions of the cut product



Motion & Position Systems

Gravure Printing Process

- All axes in the machine have the ***same web speed***
- The virtual master generates the position and speed set value as electronic line shaft (ELS) by math and software, which replaces the mechanical line shaft (MLS) for synchronization
- The format depending angular reference for the printing cylinder axes are generated by the local master using the electronic gear box function



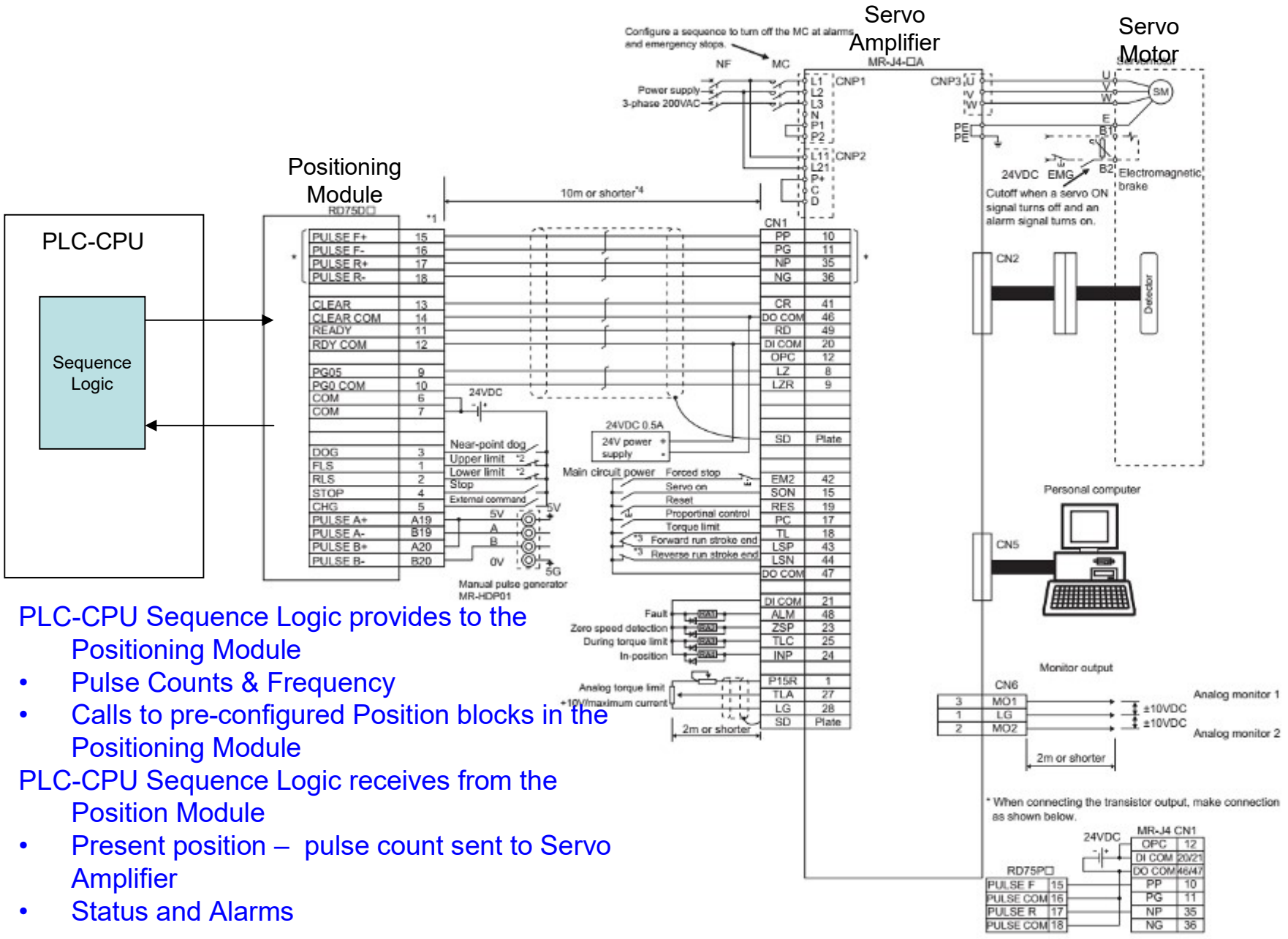
PLC- Position and Motion Systems

Types of PLC Motion System

- **Positioning Module** (may be built-in using on-board PLC outputs)
  - Pulse output – Open Collector OR Differential, PULSE/SIGN OR CW/CCW
  - Upto 8 axes
  - Logic in PLC-CPU sequence program
  - Operations
    - Electronic Gear
    - OPR (Original Point Return) Control – Zero Return Control
    - Positioning Control (PTP (Point To Point), Path) – Preset Segment Data, Absolute/Incremental
    - Modes - Speed Control, Position Control, Speed / Position Control switching, Jog
    - Interpolation - Linear, circular, helical - sealant-glue applicator, milling
- **Simple Motion Module**
  - Communications
  - Upto 16 axes
  - Module has predefined configuration, PLC CPU sequence triggers the execution
  - Operations (supports all Positioning Module Operations)
    - Electronic Cam Function
    - Synchronous Control - for Printing Lines, Packing Machines, Sheet Metal working
    - Speed-Torque Control –Press-Fit Control cap tightening
- **Motion Controller** (PLC-CPU and Motion Control in the same module)
- **Motion Module** (separate CPU dedicated for Motion in PLC rack)
  - Communications
  - Upto 256 axes
  - Motion Module is programmed in ST, PLC CPU sequence triggers the execution
  - Operations (supports all Simple Motion Module Operations)
    - G-code program execution
    - Simplified Robot Control

Electronic Cam / Synchronous control – movements previously used mechanical coupling for coordinated motions can now be controlled by separate (smaller) motors that are electronically coordinated with no mechanical linkage.

PLC-CPU with Positioning Module



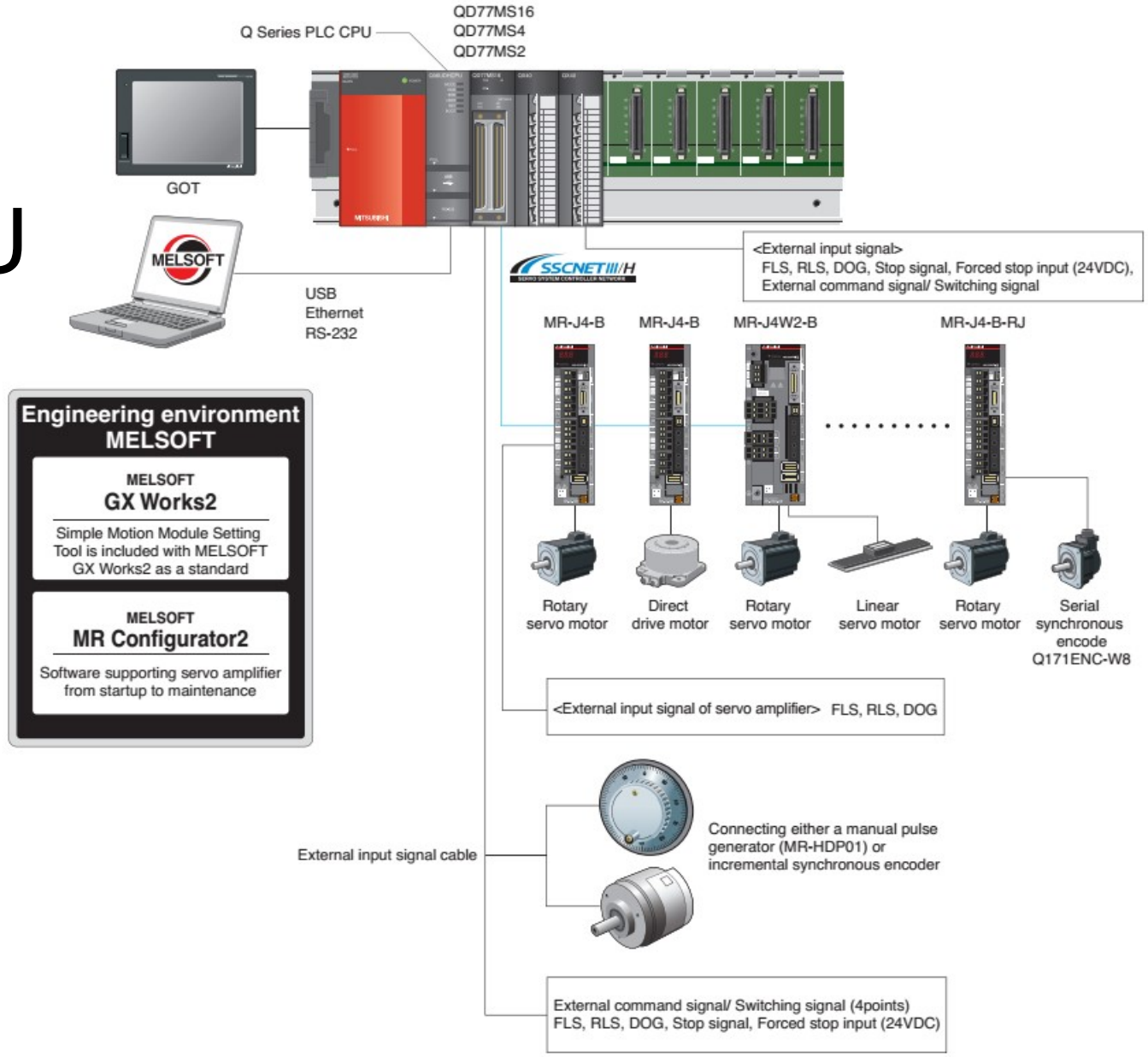
PLC-CPU Sequence Logic provides to the Positioning Module

- Pulse Counts & Frequency
- Calls to pre-configured Position blocks in the Positioning Module

PLC-CPU Sequence Logic receives from the Position Module

- Present position – pulse count sent to Servo Amplifier
- Status and Alarms

PLC-CPU *with* Simple Motion Module



Engineering environment MELSOFT

MELSOFT GX Works2

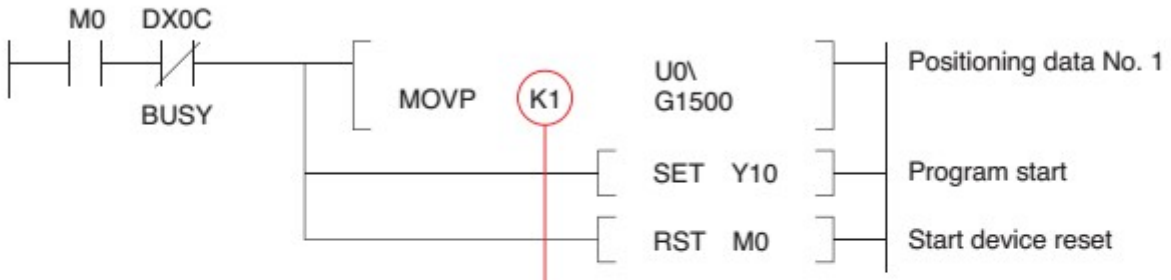
Simple Motion Module Setting Tool is included with MELSOFT GX Works2 as a standard

MELSOFT MR Configurator2

Software supporting servo amplifier from startup to maintenance

PLC-CPU with Simple Motion Module

PLC CPU



Simple Motion module

Axis #1 Positioning Data

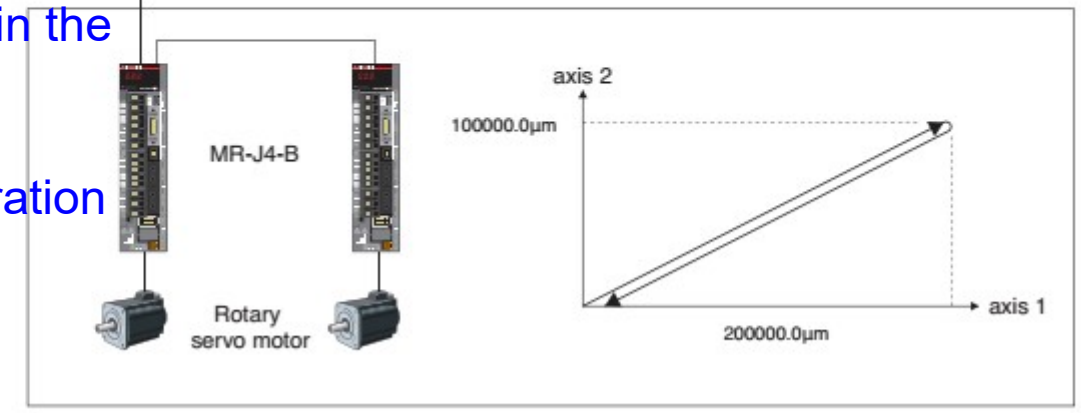
No.	Operation pattern	Control system	Axis to be interpolated	Acceleration time No.	Deceleration time No.	Positioning address	Arc address	Command speed
1	1: CONT	0Bh: INC Linear 2	Axis #2	0: 1000	0: 1000	200000.0 µm	0.0 µm	20000.00 mm/min
	<Positioning Comment>							
2	0: END	0Bh: INC Linear 2	Axis #2	0: 1000	0: 1000	-200000.0 µm	0.0 µm	10000.00 mm/min
	<Positioning Comment>							

Axis #2 Positioning Data

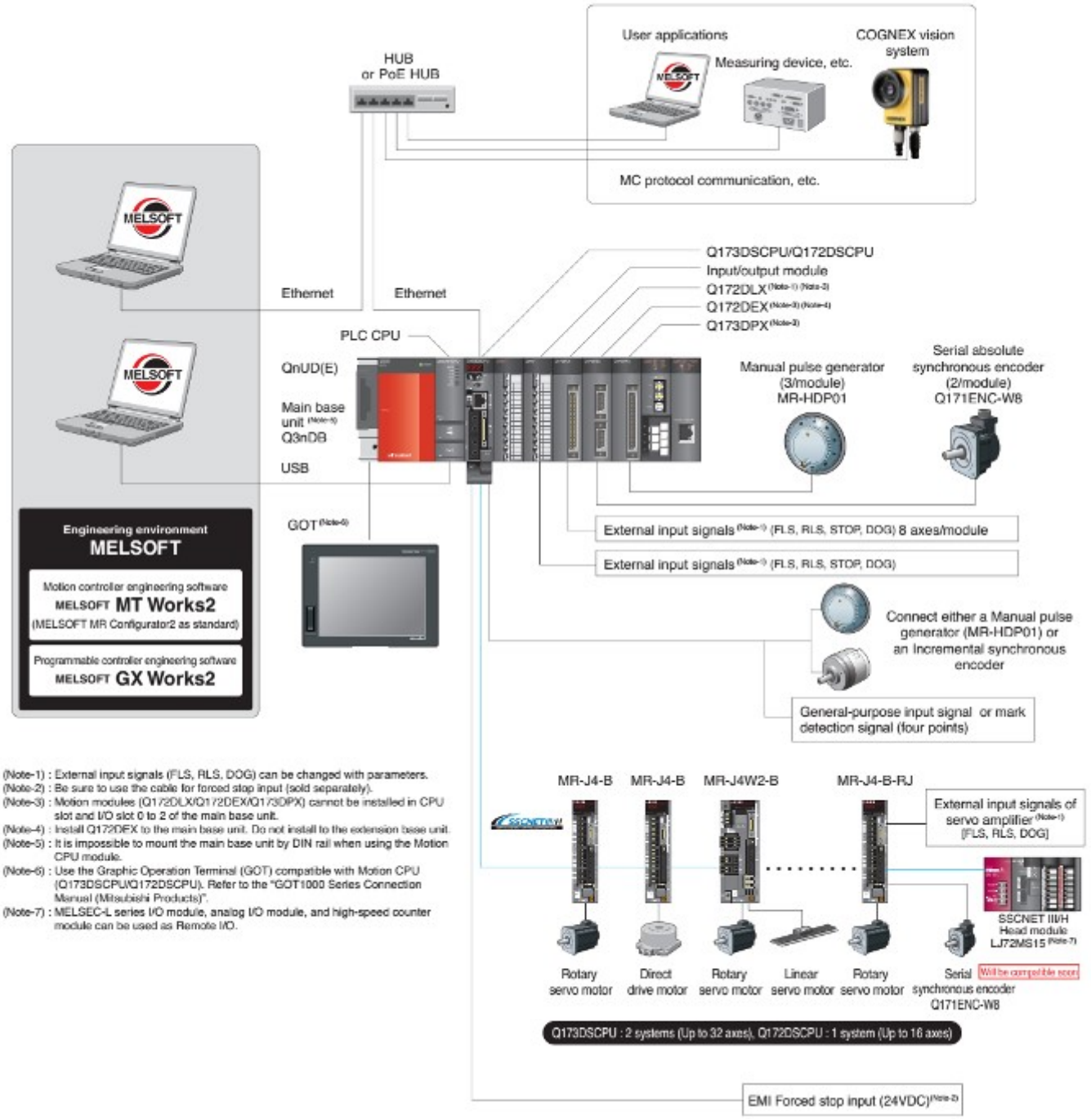
No.	Operation pattern	Control system	Axis to be interpolated	Acceleration time No.	Deceleration time No.	Positioning address	Arc address	Command speed
1						100000.0 µm	0.0 µm	0.00 mm/min
	<Positioning Comment>							
2						-100000.0 µm	0.0 µm	0.00 mm/min
	<Positioning Comment>							

Motion Operations stored in Tables in the Simple Motion Module

PLC-CPU triggers the Position Operation in the Simple Motion Module

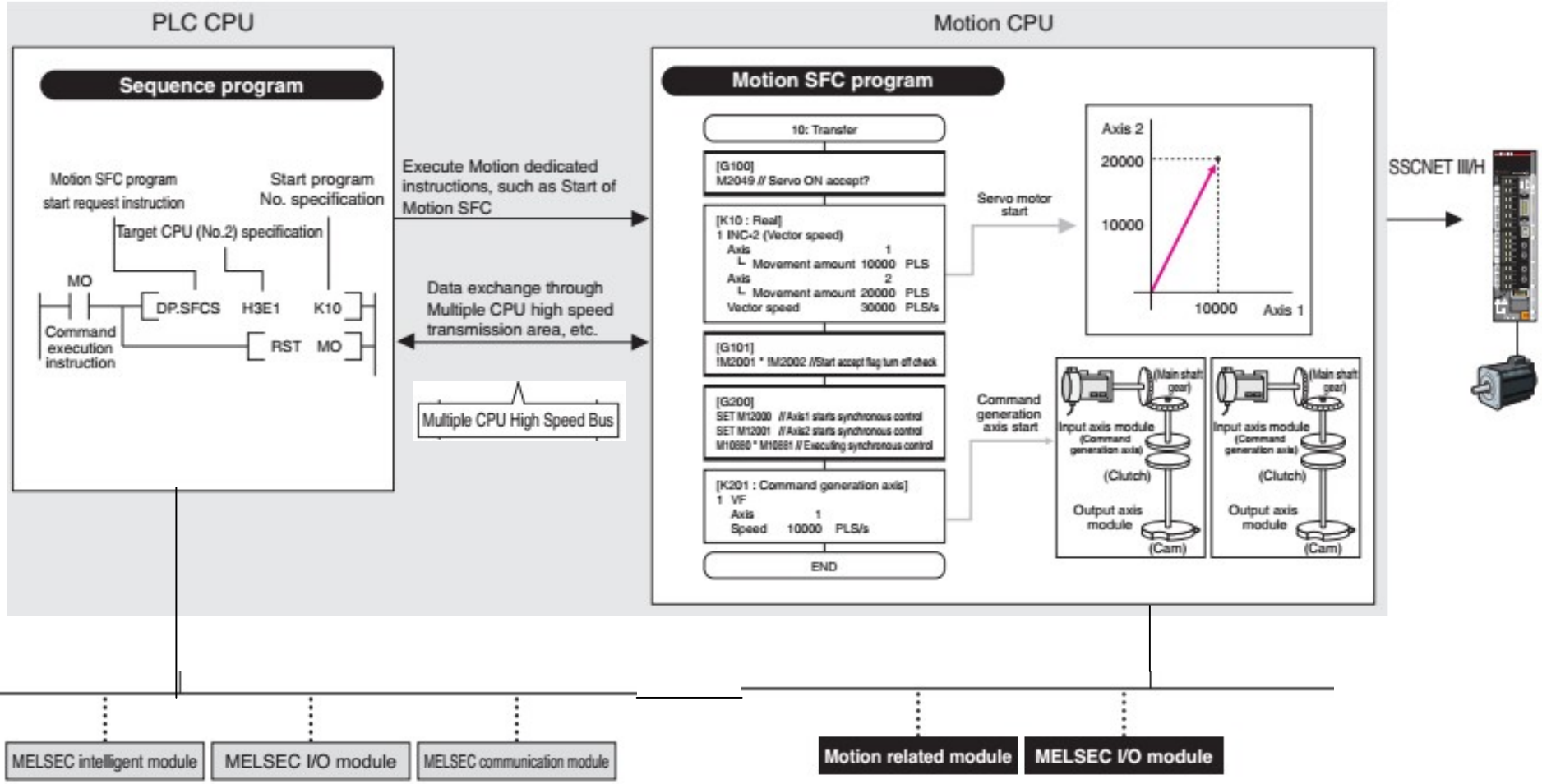


PLC-CPU with Motion-CPU



(Note-1) : External input signals (FLS, RLS, DOG) can be changed with parameters.
(Note-2) : Be sure to use the cable for forced stop input (sold separately).
(Note-3) : Motion modules (Q172DLX/Q172DEX/Q173DPX) cannot be installed in CPU slot and I/O slot 0 to 2 of the main base unit.
(Note-4) : Install Q172DEX to the main base unit. Do not install to the extension base unit.
(Note-5) : It is impossible to mount the main base unit by DIN rail when using the Motion CPU module.
(Note-6) : Use the Graphic Operation Terminal (GOT) compatible with Motion CPU (Q173DSCPU/Q172DSCPU). Refer to the "GOT1000 Series Connection Manual (Mitsubishi Products)".
(Note-7) : MELSEC-L series I/O module, analog I/O module, and high-speed counter module can be used as Remote I/O.

PLC-CPU & Motion Controller Communication



[Sequence program]

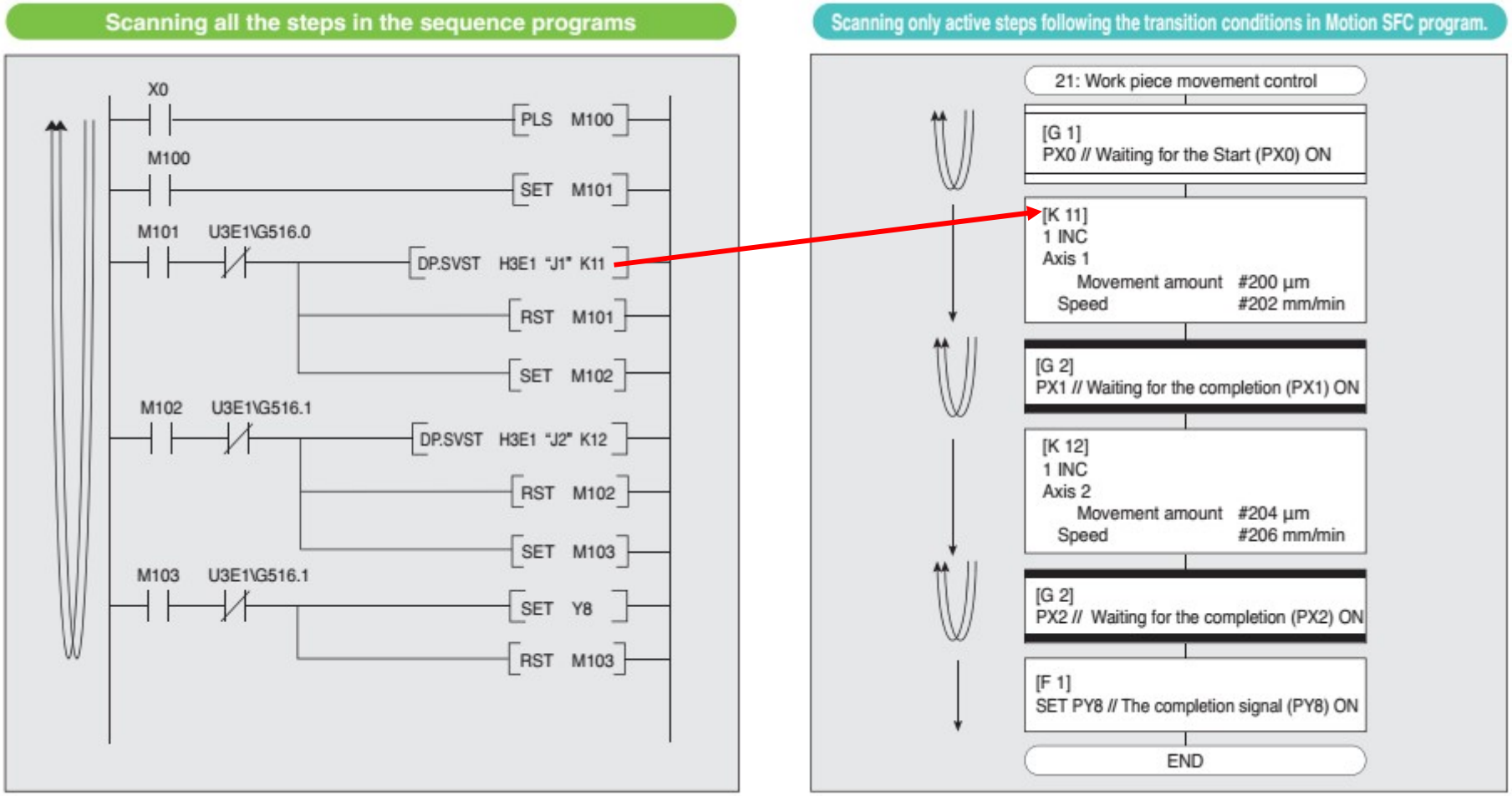
- Ladder description is suitable for scan process (Importance focused on condition control)
- Sequence control (Compatible with multiple I/O points, multiple operations)
- System stop processing at error detection

[Motion SFC program]

- Motion SFC description is suitable for event processes (Importance focused on sequential control, pursuit of event responsiveness)
- Servo high-speed response (Start)
- Positioning address, speed data operation, speed change
- High functionality with multitasking and branching

Motion CPU executes independently & directly accesses IO - thereby reducing load on the PLC CPU

PLC-CPU and Motion Controller Scans

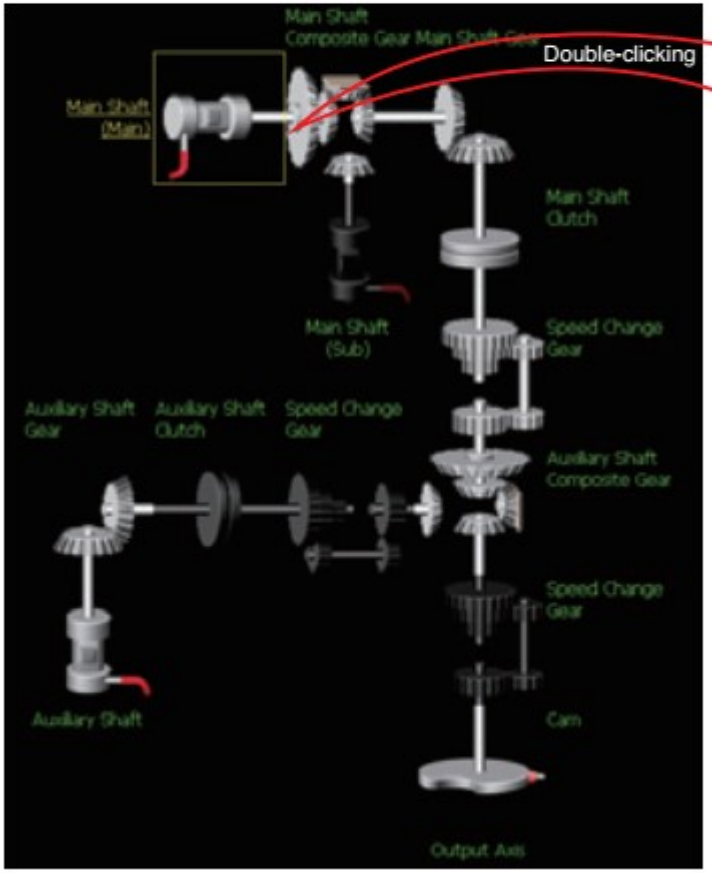


Synchronous Control

●Synchronous control is easily executed by setting parameters.

●The movement amount of the main shaft can be transmitted to output axes via the clutch.

●"Command generation axis" is not considered as a control axis; therefore the output axes can be set using all of the available control axes.



Double-clicking

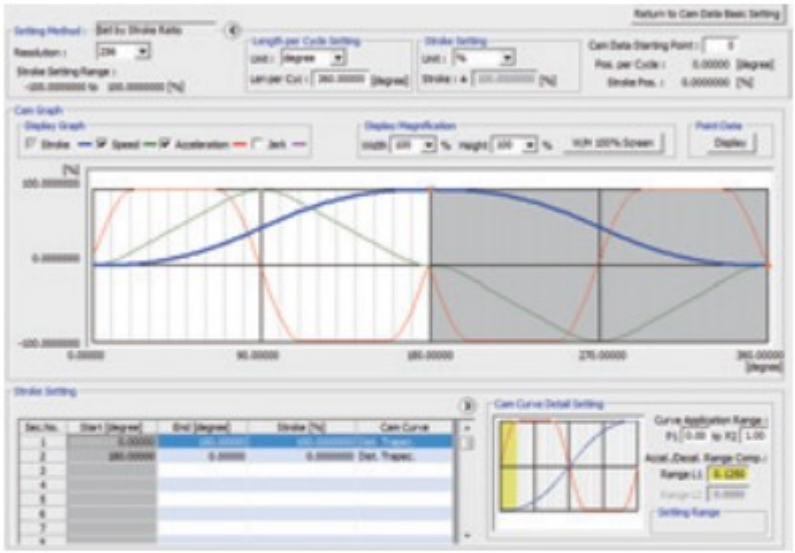
Item	Setting Value
Synchronous Control Module Setting	Set each module parameter.
Main Shaft	
Main Input Axis	
Type	201:Command Generation Axis
Axis No.	
Sub Input Axis	
Type	
Axis No.	
Main Shaft Composite Gear	
Main	1:Input+
Sub	0:No Input
Main Shaft Gear	
Numerator	1
Denominator	1
Main Shaft Clutch	
Main Shaft Clutch Control Setting	
ON Control Mode	1:Clutch Command ON/OFF
OFF Control Mode	0:OFF Control Invalid
High-speed Input Request Signal	0
Main Shaft Clutch Reference Address Setting	0:Current Value after Main Shaft Composite Gear
Main Shaft Clutch ON Address	0 PLS
Travel Value before Main Shaft Clutch ON	0 PLS
Main Shaft Clutch OFF Address	0 PLS
Travel Value before Main Shaft Clutch OFF	0 PLS
Main Shaft Clutch Smoothing System	0:Direct
Main Shaft Clutch Smoothing Time Constant	0 ms
Slippage at Main Shaft Clutch ON	0 PLS
Slippage at Main Shaft Clutch OFF	0 PLS

0:Invalid
1:Servo Input Axis
201:Command Generation Axis
801:Synchronous Encoder Axis

Electronic Cam

A wide variety of cam patterns can be easily created.

[Cam Data Creation Screen]



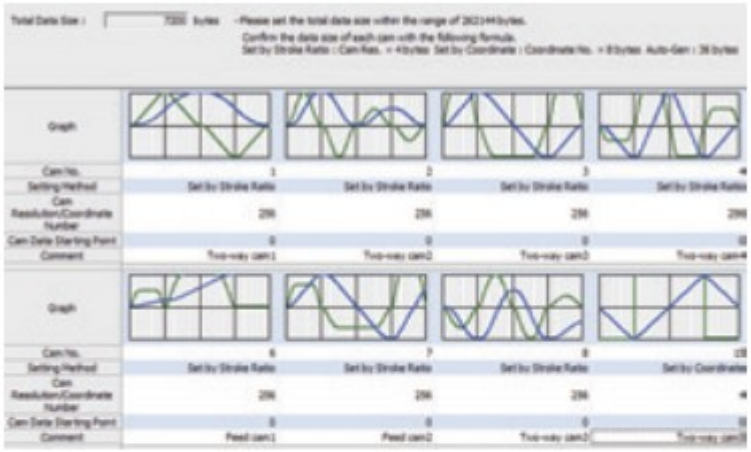
●Cam data has been created more freely than the previous ones. Various cam data is available.

●Click the graph and drag it, which causes the waveform to automatically change according to the pointer's movement.

●Stroke, speed, acceleration, and jump of speed can be set while checking the change of the graph.

●Cam data can be imported and exported in CSV format.

[Cam Data List]



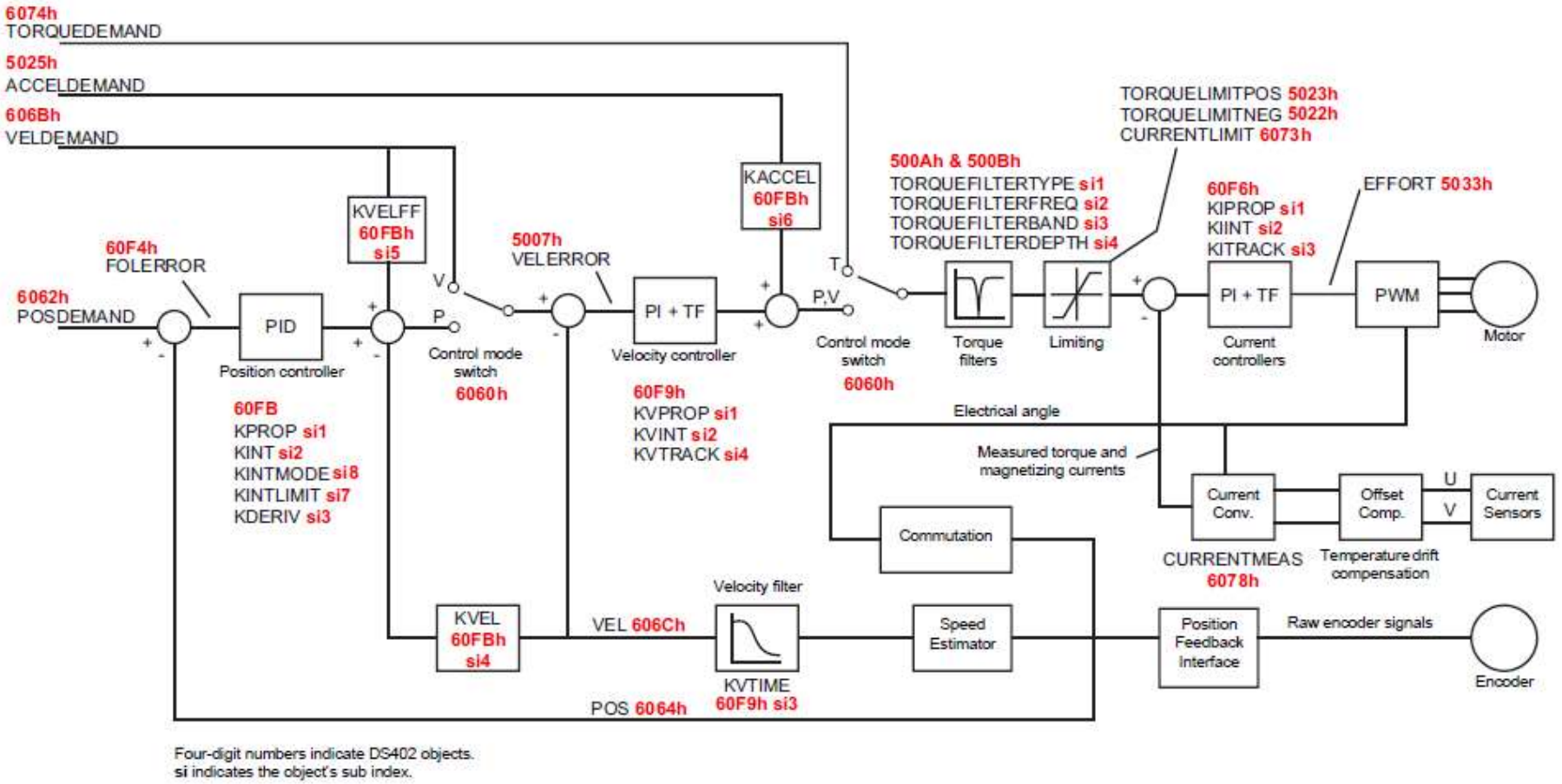
●The created cam data is easily checked with the thumbnail display.

●The screen for cam data creation will open by double-clicking the cam data to be edited.

Servo Amplifiers

- Network Compatible
  - synchronous control and interpolation control by sequential commands
  - Built-in Positioning – only the pre-configured seq no selected by Master
    - Point Table
    - Program
    - Indexer
- Multi-Axis models

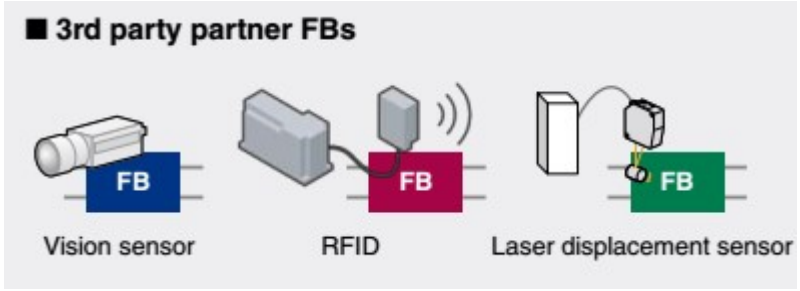
Servo-Amplifier Control Loops – Position, Speed, Torque

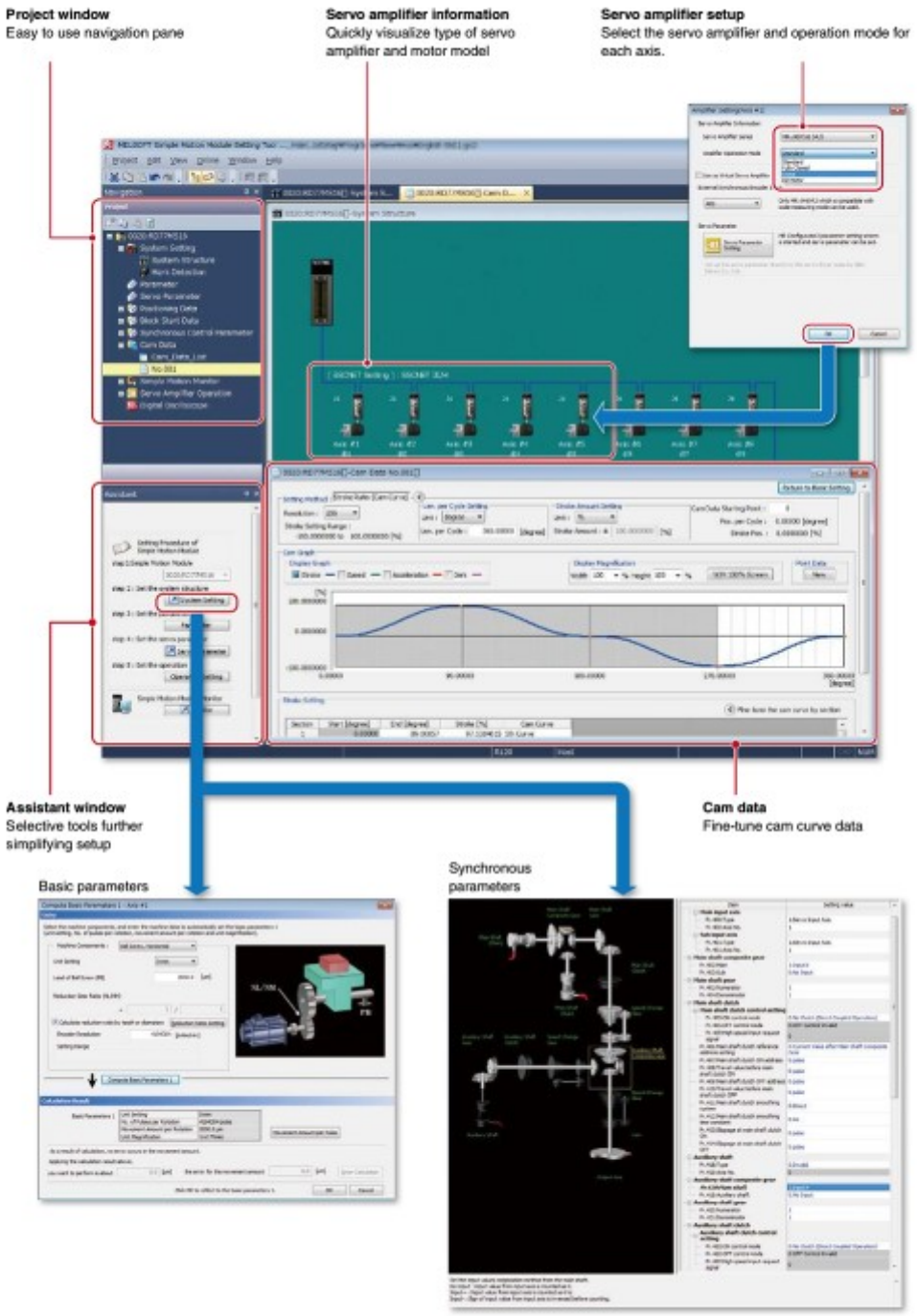


Four-digit numbers indicate DS402 objects.
si indicates the object's sub index.

Integrated Software

- Programming
  - PLC-CPU, Motion-CPU, Servo Amplifier
  - Assistants for data setting, arc calculation
  - Motion Simulation
  - Motion Controller Programming
  - Synchronous Parameter setting
    - electronic gears, shaft, cams
  - Cam data creation
- Start-up and Adjustment
  - Monitor & Alarm
  - Digital Oscilloscope
  - Multi-axis adjustment
  - Tuning

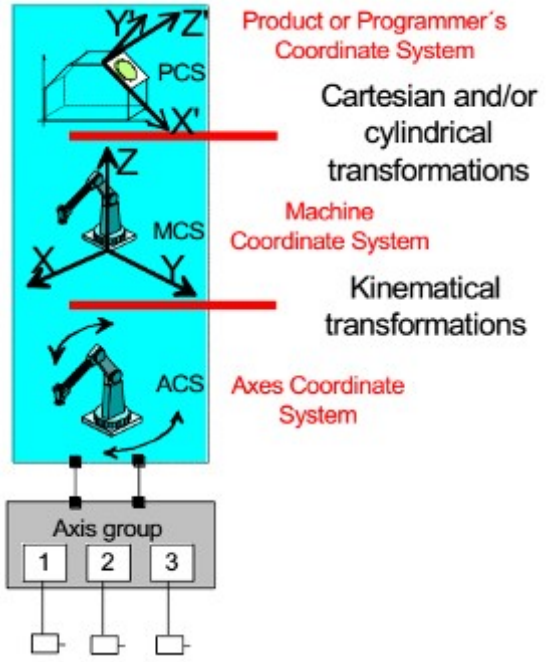




Allied Fields
Robotics

A Typical Robot

Commercially available robots have an optimized set of Kinematic Equations based on the manufacturer's knowledge of their mechanical Construction.
The Robot Controller has the Processor for calculations and Power Electronics Motor Drive Hardware to control the motors



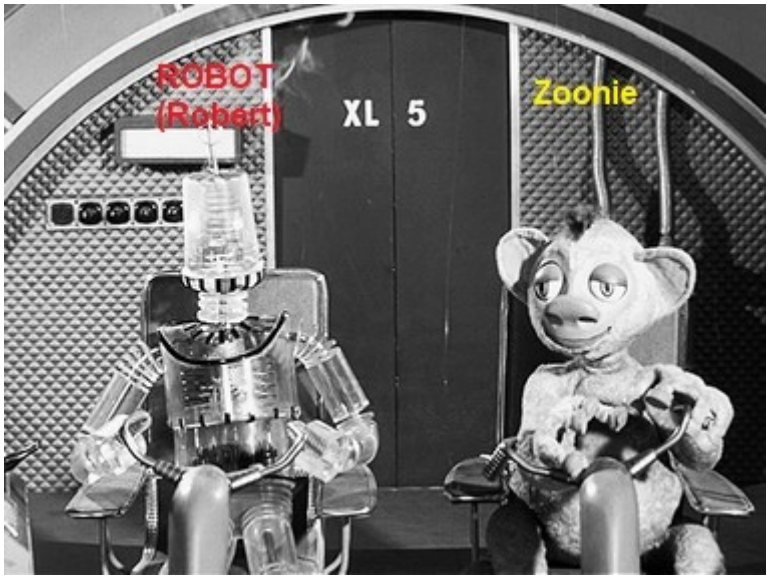
Robot-Mechanics

Teach-Pendent

Robot-Controller

No.	Component	Function
1	Manipulator	The manipulator represents the actual robot mechanics, i.e. the kinematics, which executes the ordered commands.
2	smartPAD programming handset	Settings can be entered and checked on the robot controller using the smartPAD programming handset. Likewise, the robot can be moved manually and automatically using the programming handset.
3	Connecting cable/smartPAD	
4	Robot controller	The robot controller coordinates the movements of the robot. The calculation of the coordinate transformation for the robot movements and the control of the robot axis motors occur in this controller. The robot controller may also contain the power units for the robot axis motors.
5	Connecting cable/data cable	
6	Connecting cable/engine cable	

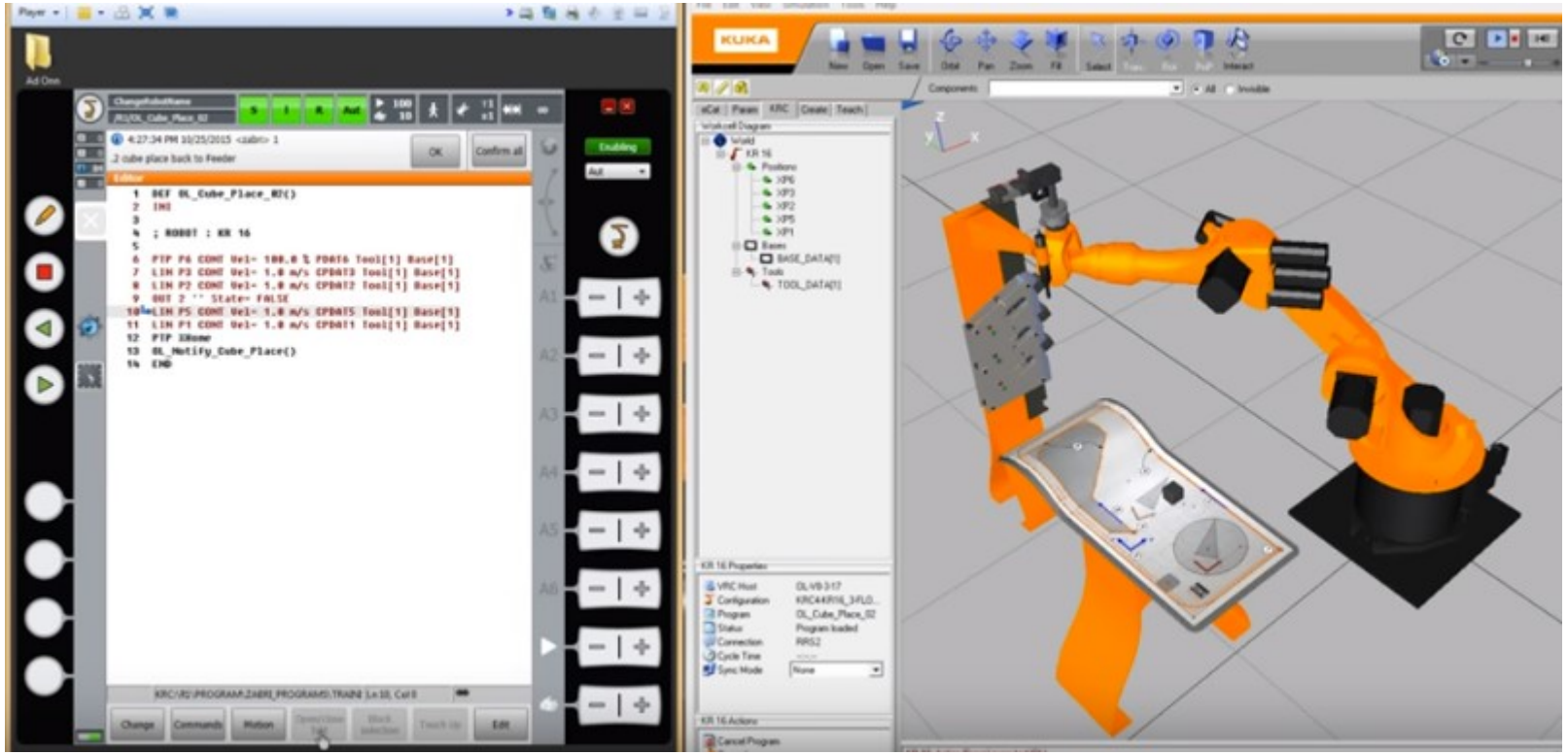
The PLC and The Robot



- The PLC is the Robot
  - For simple applications like bang-bang, x-y motion
  - Coupled with suitable Drives or Motion Controllers
- The Robot is the PLC
  - Application with complex Kinematic calculations but few IO points can be controlled by only the Robot Controller using on-board soft-PLC
- The PLC and Robot co-operate together
  - For large application with large number of IO points, the PLC executes the machine logic and co-ordinates over high-speed communication busses with one or more robots that execute the Kinematic calculations

Typical Robot Languages

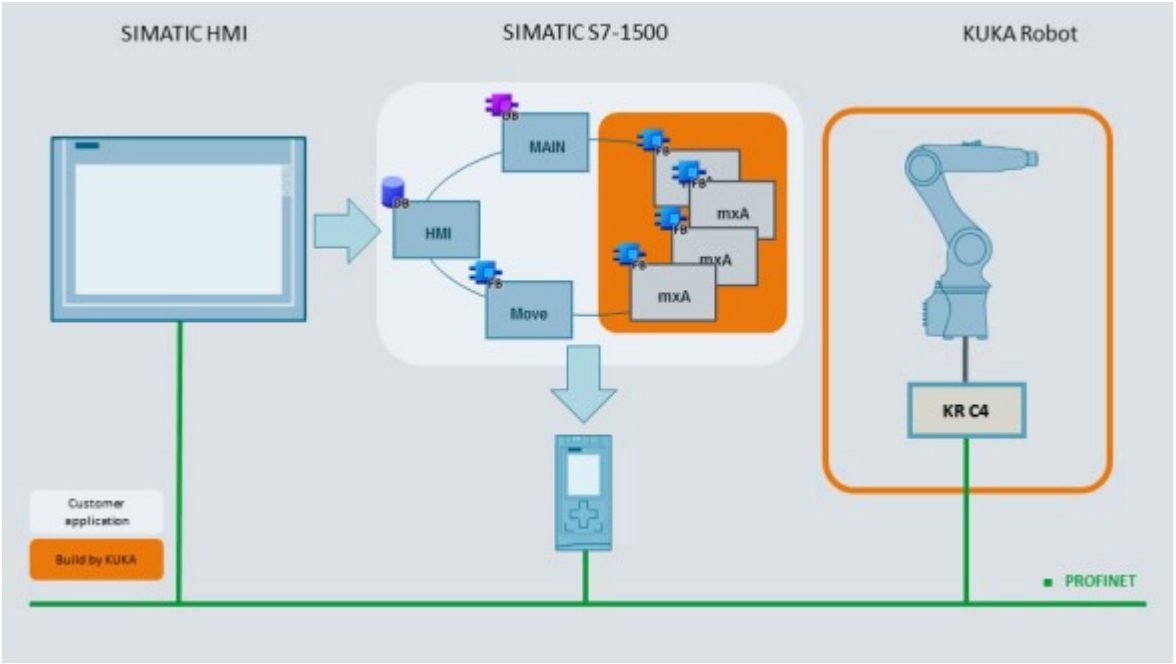
- Kuka Robot Language (KRL)
  - Similar to Python

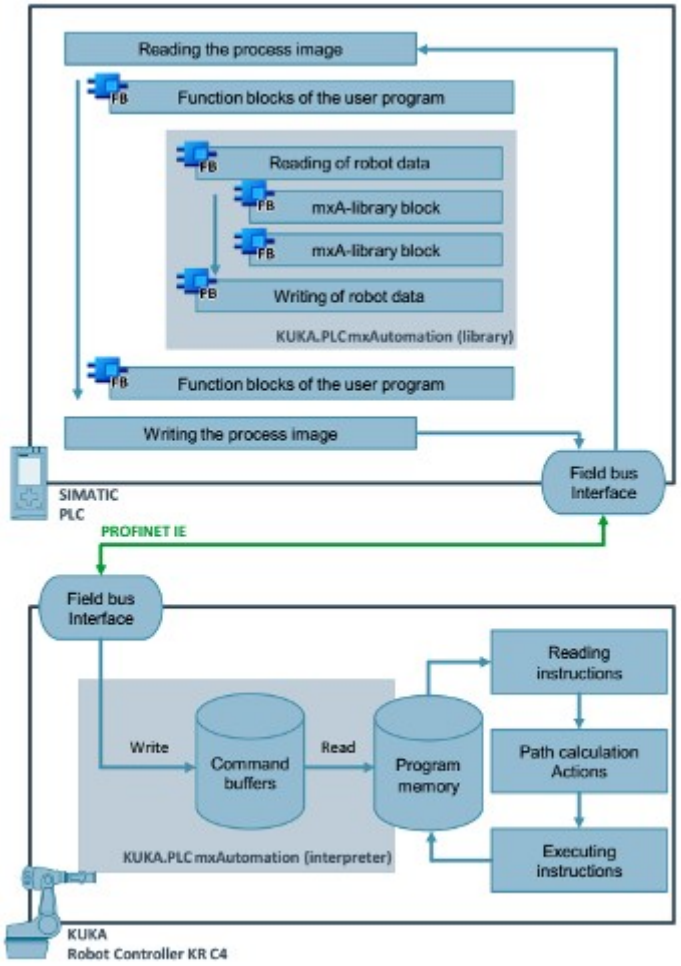


The PLC – Robot Interface

- Siemens TIA - Combines PLC and Robotic Programming Environments
  - Robot's Teach Pendent is no longer the primary interface with the robot controller.
  - With PLC-based robotic controls, the Human Machine Interface (HMI) is now the same throughout the system.
  - The alarming system, fault recording, data monitoring, and the other functions that are available to HMI now directly interface with the robot controller.
  - Unique faults and custom operations can be added and changed directly to the robot controller.
  - An HMI interface allows for a much greater application-specific focus, as well as a considerably more agile structure.
  - Robots from multiple vendors can be programmed from the same PLC programming environment, thereby greatly reducing engineering effort as there is no need to learn the Robot Specific Programming Language

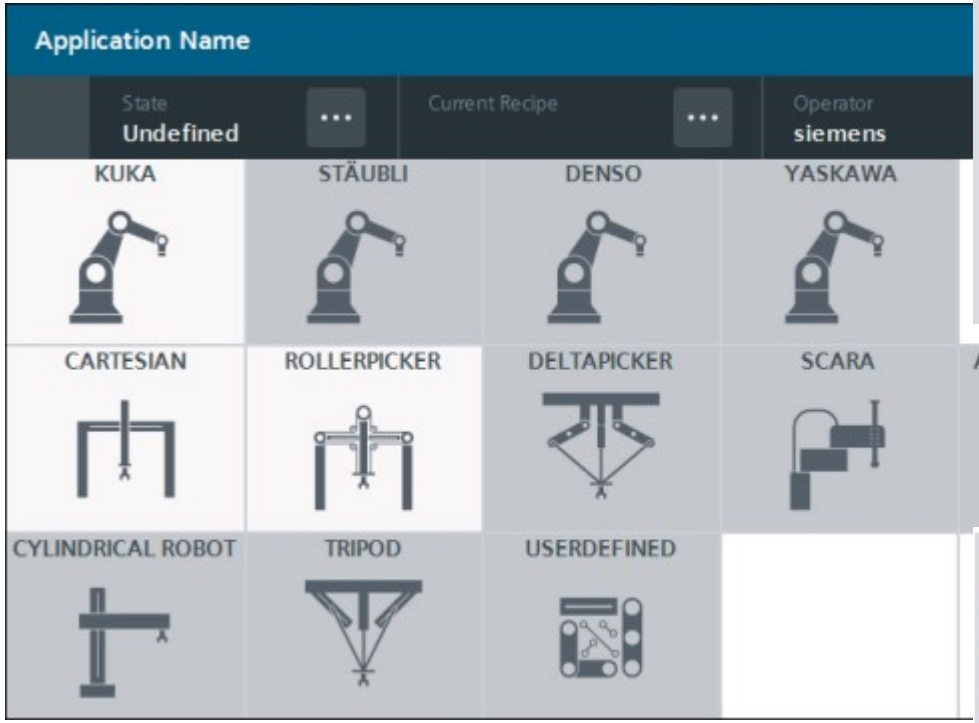
PLC –Robot Interface

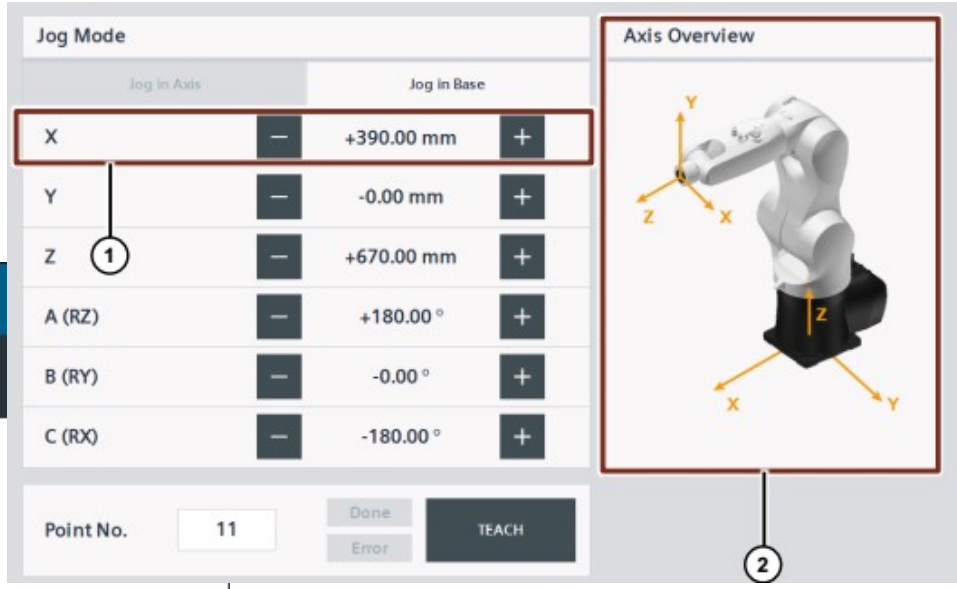




• The KUKA industrial robot consists of the KUKA KR C4 Robot Controller and the robot's mechanical system
• Using the KUKA.PLC mxAutomation block library, the SIMATIC controller controls the robot.
• The interpreter for the commands of the KUKA.PLC mxAutomation block library, on the Robot Controller, receives the commands from the SIMATIC controller, computes kinematic transformations and executes them on the robot's mechanical system.

- Uniform UI to program and control Robots from different vendors and different coordinate systems





- System can be taught in jog Mode – using cartesian or axial co-ordinates
- Both individual Positions and trajectory curves Paths can be saved

(PLC's) HMI is used to interact with the Robot Controller

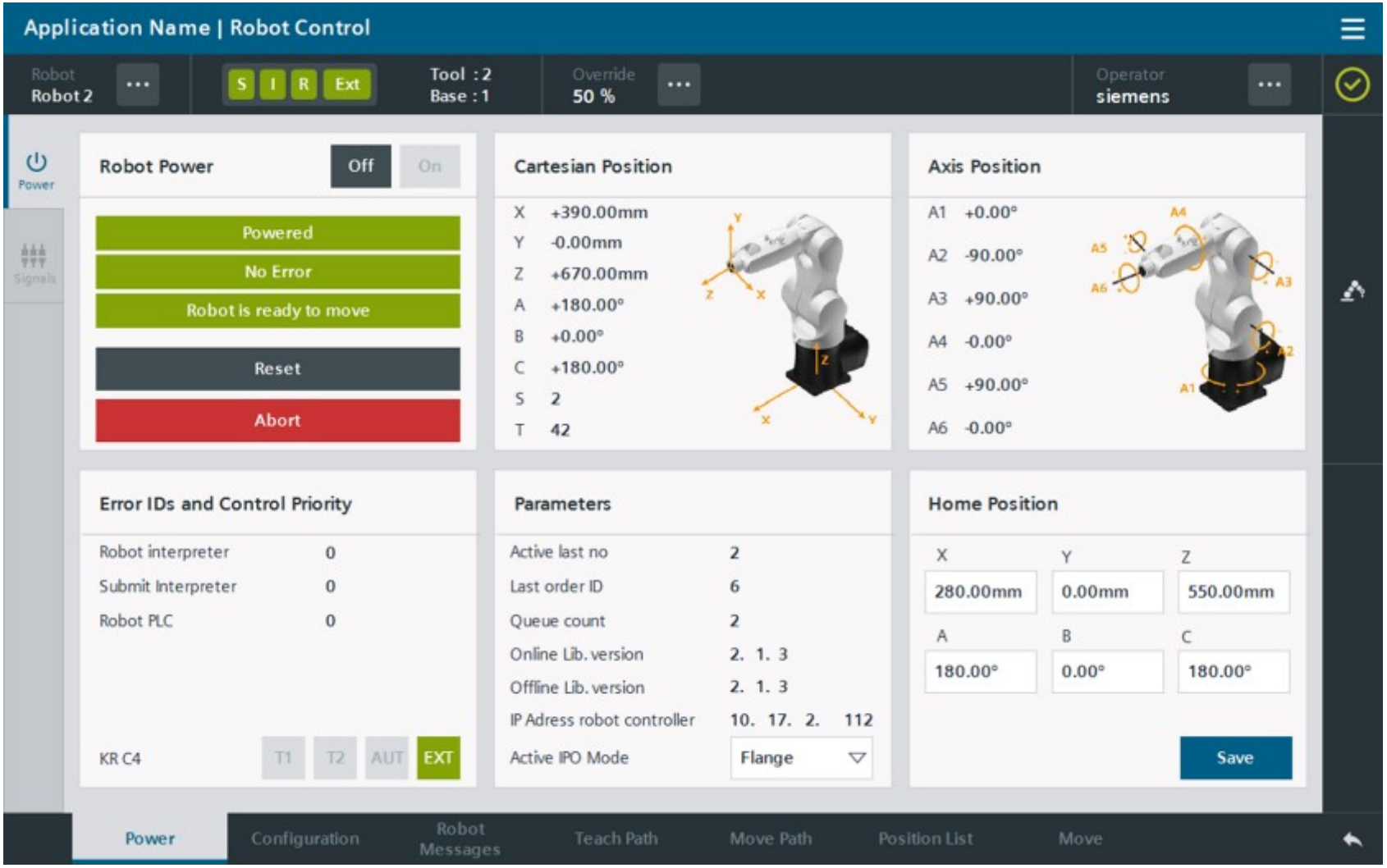
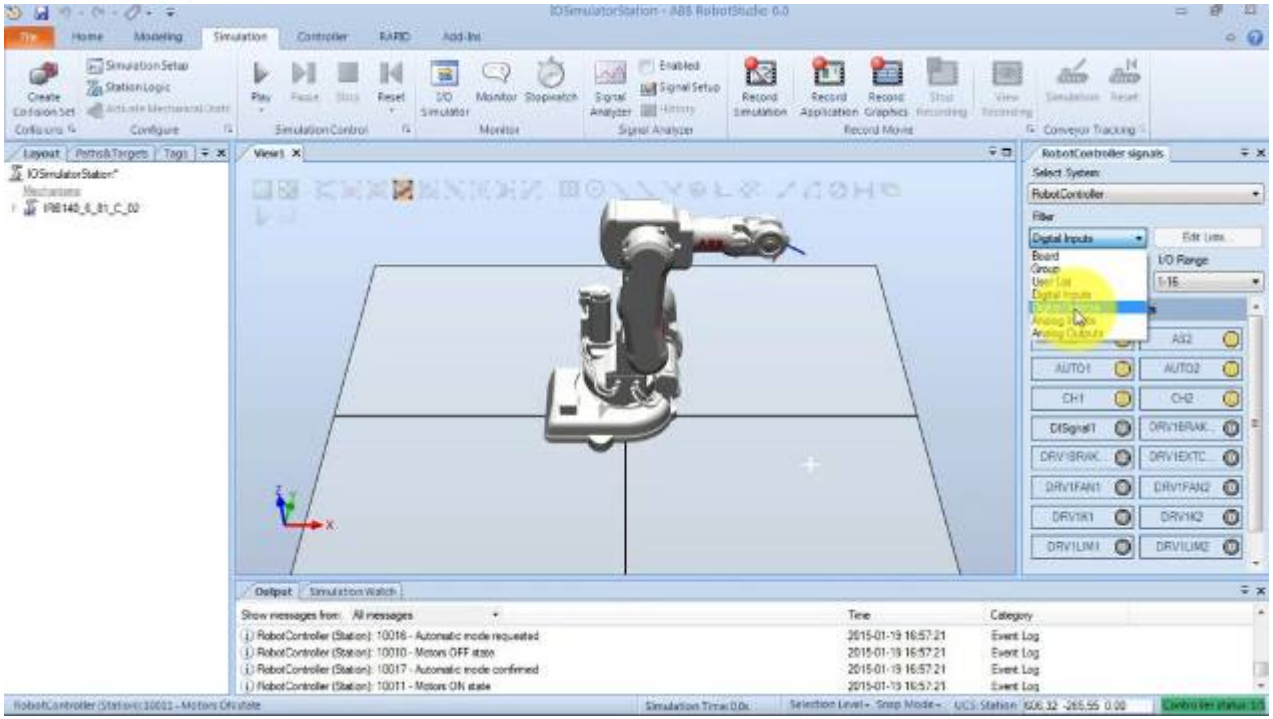
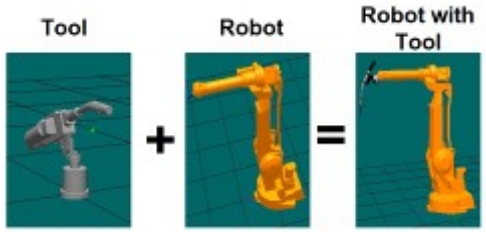


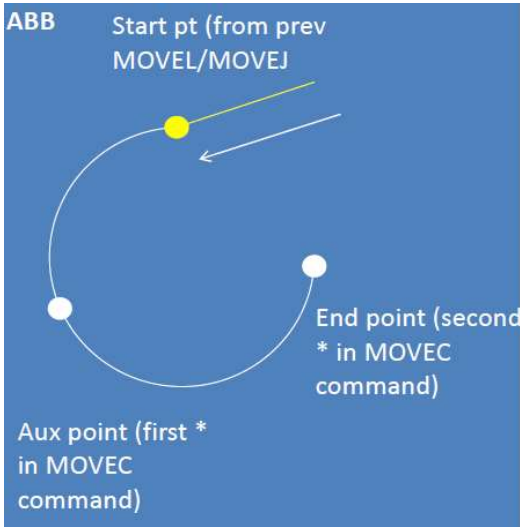
ABB RobotStudio®






```
MOVEL [start pt] , [vel mm/s] , [Zone] , [tool]
MOVEC [Aux pt] , [End pt] , [vel mm/s] , [Zone] , [tool]
```

```
MODULE MainModule
  PROC main()
    MOVEL *, v1000, z50, tool0;
    MOVEC *,*,v1000, z10, tool0;
  ENDPROC
ENDMODULE
```



1. Import Workbench CAD
2. Import WorkPiece CAD
3. Select Robotic Arm
4. Select Tool
5. Set the Axes
6. Program the Tool Movements
7. Run the System in Off-line Simulation Mode and Optimize
8. Download Program to Shop-Floor Robot for use in On-line Production

RoboDK GAZEBO Webots FANUC ROBOGUIDE

ABB RobotStudio®

Virtual Meetings

Is a collaboration feature allowing to share the digital robot solutions in web meetings. The participants are immersed in the virtual room, using a VR headset connected to RobotStudio, where the RobotStudio station can be shared for making design reviews and sales proposals without travelling

Digital Twin

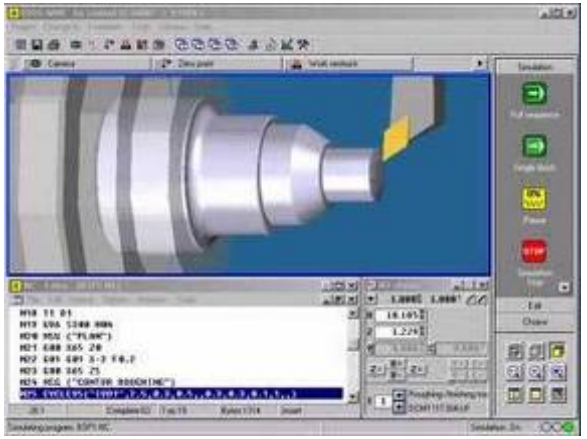
is a concept to monitor and optimize the automation solution without disturbing the ongoing production. It enables real-time simulation of the production system , like a digital shadow, allowing the users to try changes and do optimization in the virtual world without affecting the production.

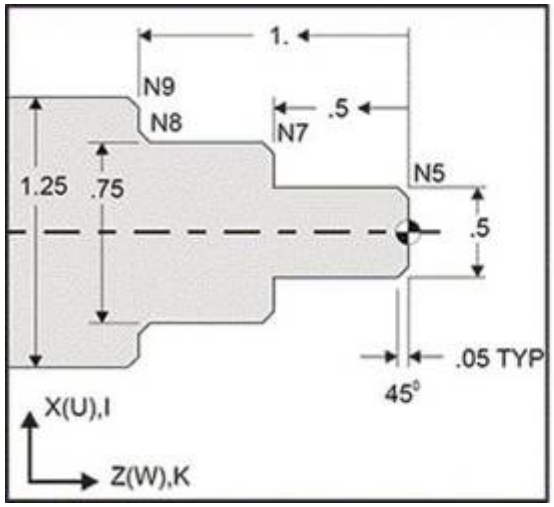
Virtual Commissioning

ABBs virtual commissioning solutions speed up commissioning by simulating an exact replica of the production cell in RobotStudio so that all technical issues can be solved in advance. RobotStudio allows to connect to PLCs and other external devices to fully virtually test the complete logic and safety of the cell prior to installing the physical line.

Allied Fields
CNC

CNC ***has it's own*** PLC



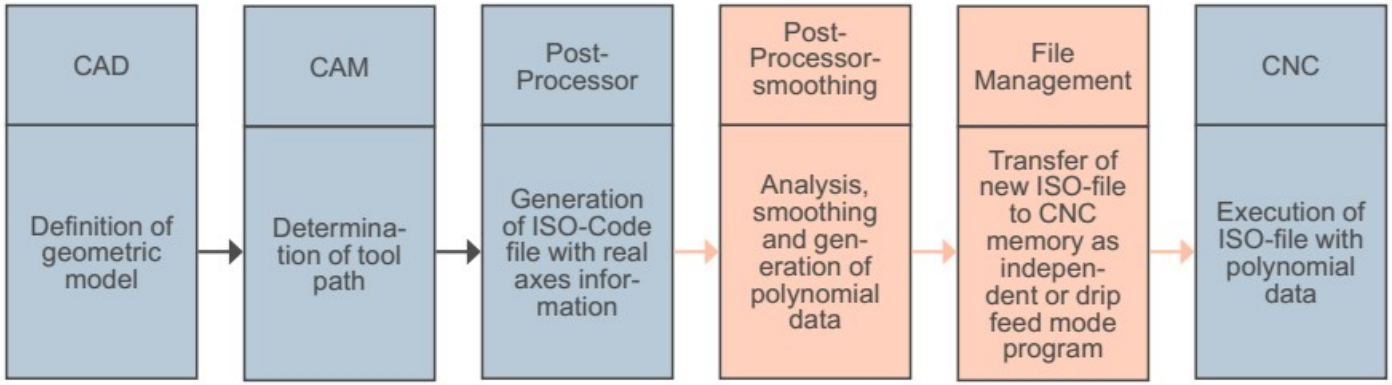



```
%
O0001 (Chamfering)
N1 G50 S1500
N2 G00 T101 G97 S500 M03
N3 G00 X0 Z0.25
N4 G01 Z0 F0.005
N5 G01 X0.50 K-0.050
N6 G01 Z-0.50
N7 G01 X0.75 K-0.050
N8 G01 Z-1.0 I0.050
N9 G01 X1.25 K-0.050
N10 G01 Z-1.5
N11 G00 X1.5 Z0.25
M30
%
```

CNC - Computer Numerical Control

- Evolved from Punch-Card programmed Numerical Controlled (NC) Machines

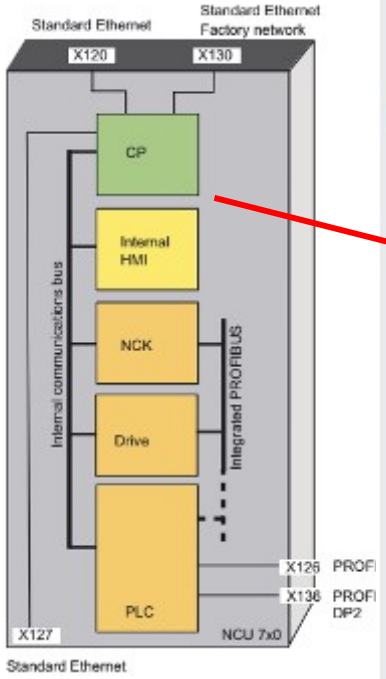
- **CNC Machining Process**
  - CAD software is used to generate 2-D or 3-D Model of the required product
  - CAM software processes the CAD Model
    - Based on constraints like
      - Part Material and Geometry, Tool Properties, Jig and Fixture Properties
    - To generate part-program code for the CNC Machine
      - Geometric G-code – Tool Path, Feed Speed..
      - Miscellaneous M-code – Coolant, Tool changes
      - While maximizing savings of time, material and energy
  - CNC machine controller executes the part-program code and together with drives and motors, controls the various machine axis to move the Tool, Work-piece and other machine parts to generate the required product
  - CNC machine also interacts with I/O, Safety Interlocks, Conveyor lines, Loading Robots, Part Inspection Systems (Contact/Non-Contact Probes, Cameras,..) using the built-in PLC



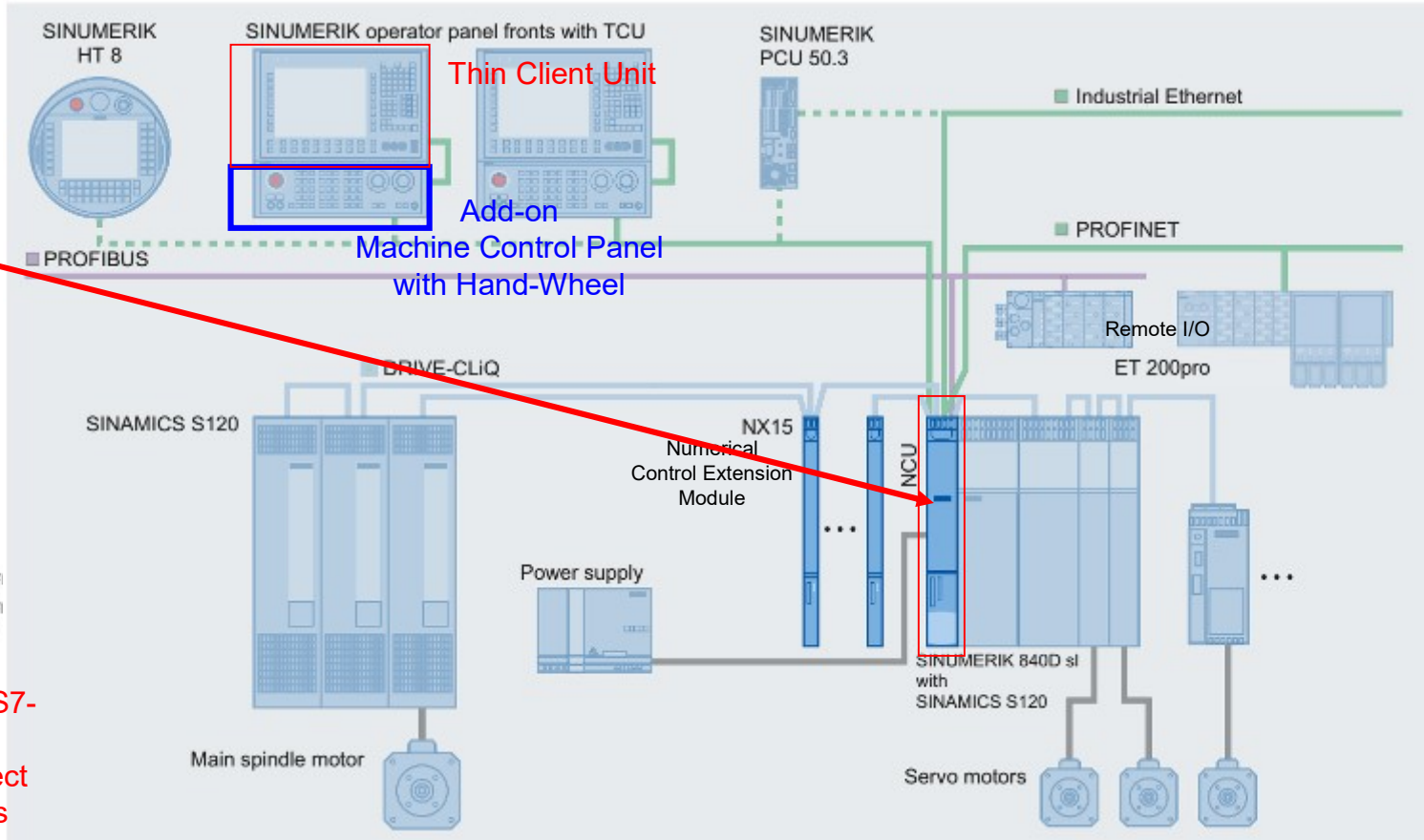
CNC Controller Hardware

NCU – NC Control Unit consists of

1. PLC **(integral part of the CNC)**
2. NCK (Numerical Control Kernel)
3. CP (Communications Processor)
4. Operator Panel / HMI (TCU+MCP ***OR*** PC for integrated CAM support)
5. Drive
6. NX (optional)
7. Remote IO



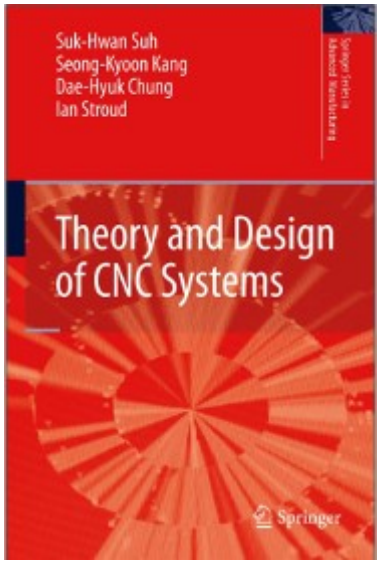
NCU7 is a specialized PLC S7-300 with NCK (Nunerical Control Kernel) that has direct communication bus access



CNC Controller Main Hardware

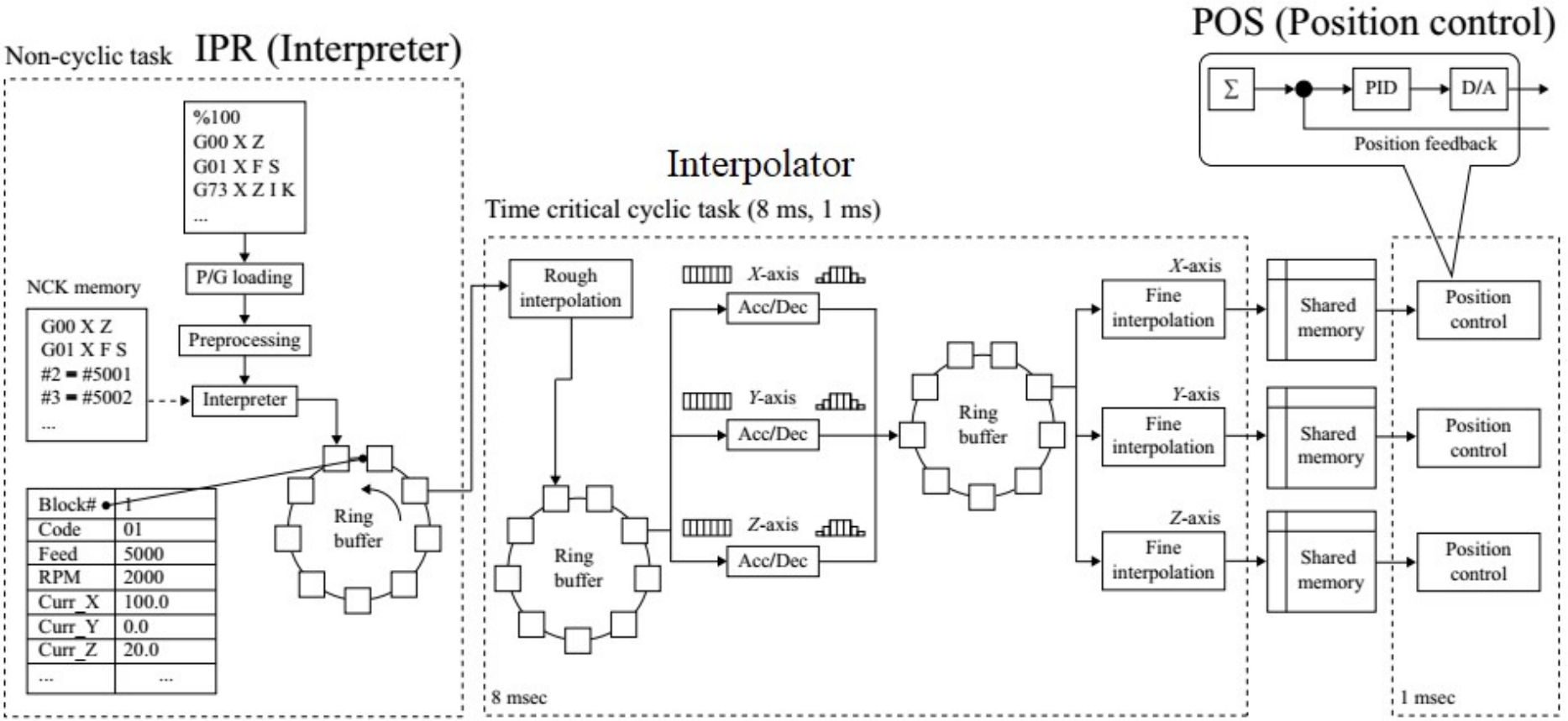
- Operator Panel (HMI) & Machine Control Panel (MCP)
  - Operator Functions – Start, Stop, Jog using Key and Digital Feed-Wheel
  - Parameter Settings
  - Part-Program Import, Editing, Test, Collision Detection
  - Monitoring & Alarm
  - Service and Utility
- NCK
  - Part-Program Interpretation
  - Interpolation for Position & Speed Set-points
  - Position Control – Accl/Dccl, stiffness, etc.. by communicating with the Servo Drives
  - sophisticated algorithms in-advance automatically optimize parameters such as machining rate or damping while taking account of friction and the contours of a workpiece
- PLC
  - Machine Logic and Sequencing
  - I/O Interlocks can make calls to the NCK to start / stop operations
  - Additional Function Blocks & Axis – like Conveyor.., can be called from the NCK
  - Interface with Loading Robot

NCK Operation



Operator Panel MMC	NC part program generation Program loading	NC program G00 X150 Z300 G54 DO M3 M8 G01 Z20 F100	X, Z, Work piece Working profile interpretation
NCK	Code interpreter Tool compensation	Code interpreter	X, Z End point calculation
	Motion data generation (linear/circular)	Interpolation	V, X, V, Z Basic velocity curve generation
	Acceleration/ deceleration	Acceleration/ deceleration control	V, X, V, Z Fitted velocity curve generation
	Backlash/pitch error compensation PID control	Position control	X, Z Tool movement
DRV	Servo drivers/motor control	Velocity control	

NCK Operation

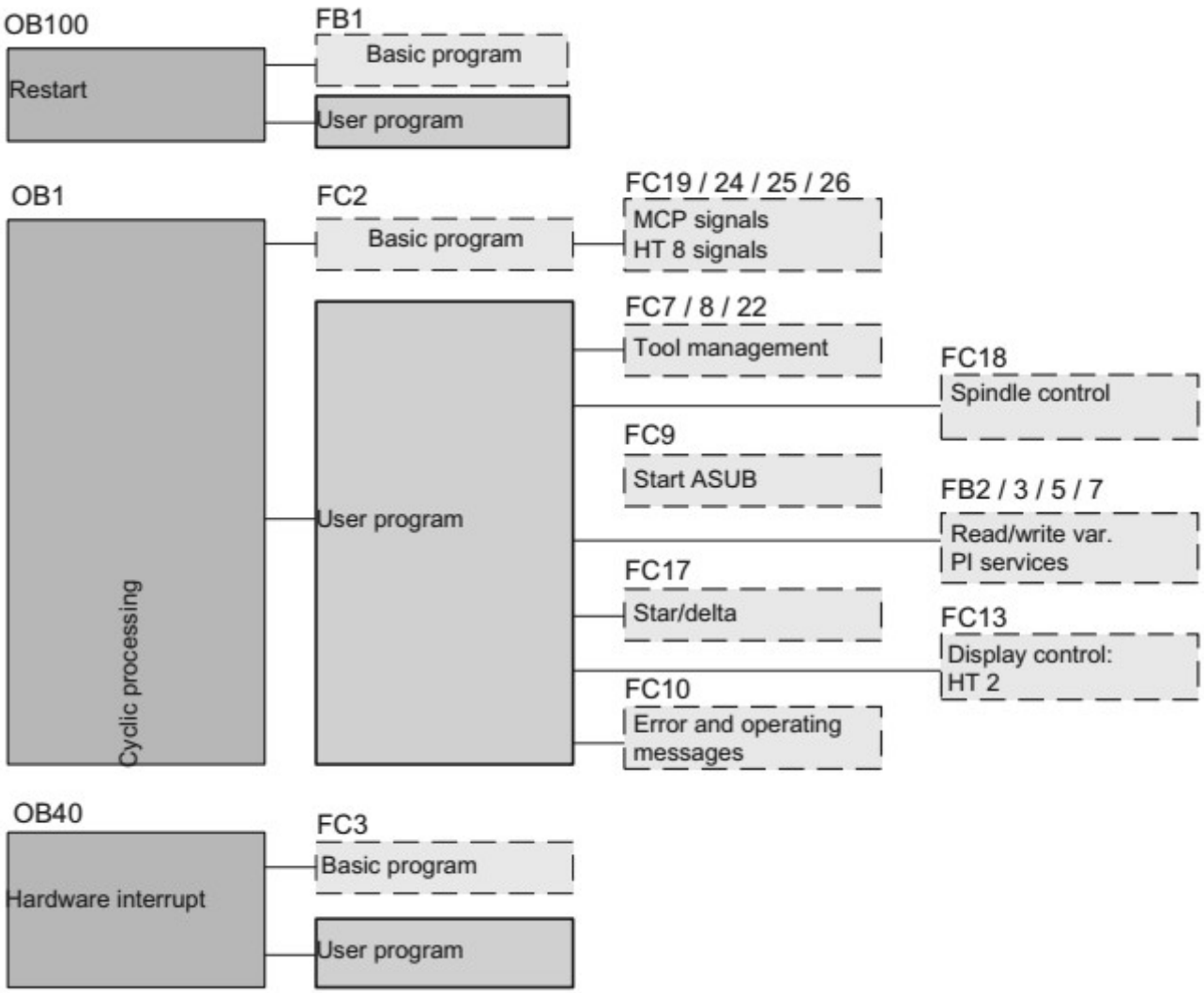


CNC Controller Software

- Machine Manufacturer uses the ***Sinumerik S7 Toolbox*** with the ***Simatic Step7 Professional (TIA Portal)***
- Machine can be programmed in PLC programming languages like Ladder, STL, FBD, SCL(Structured Control Language), etc
- HMI screens configured using standard and user controls

Sinumerik Interfaces for Machine Builders

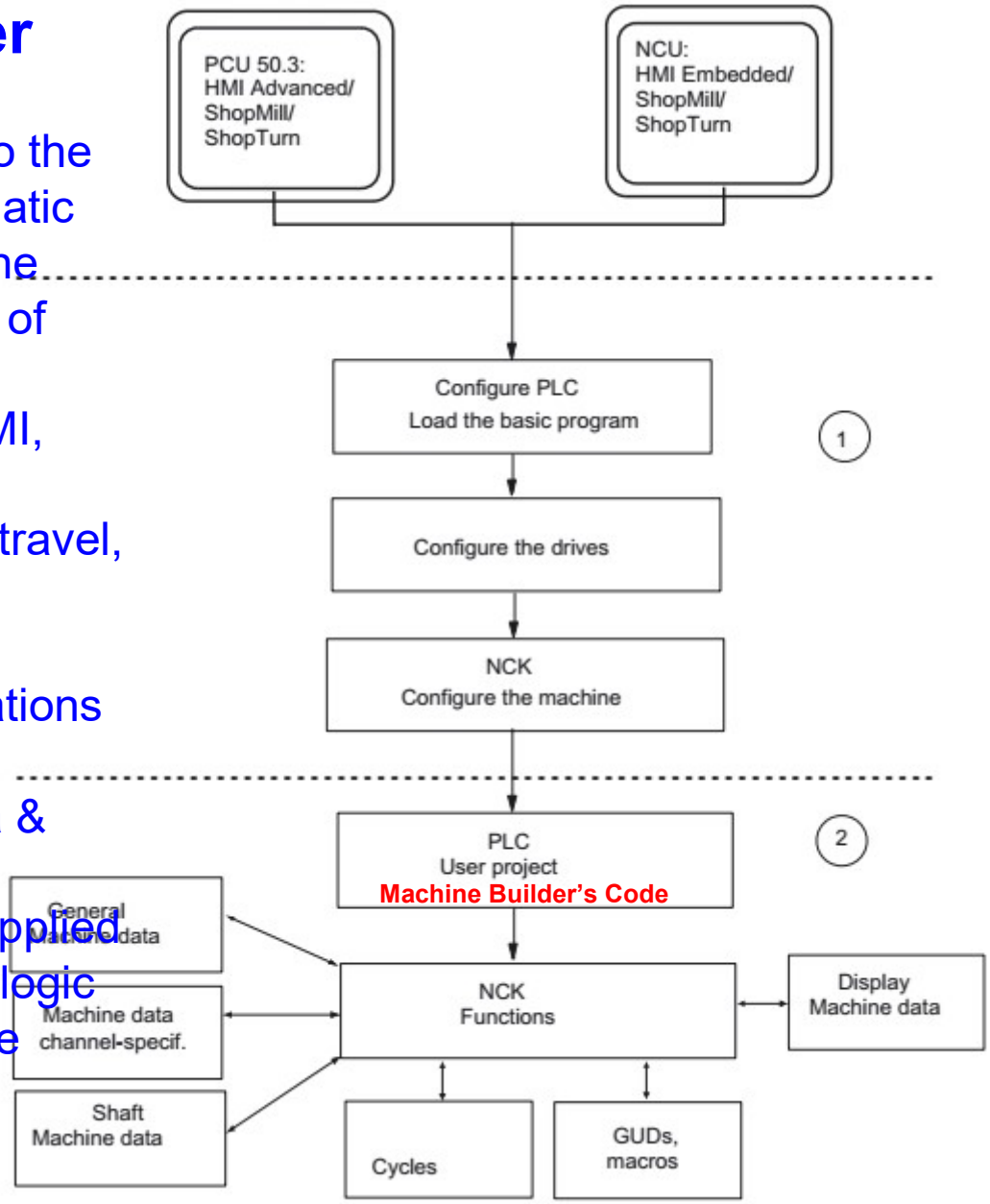
PLC Basic Program FBs



Sinumerik CNC - Machine Builder Process

Steps for the Machine Builder

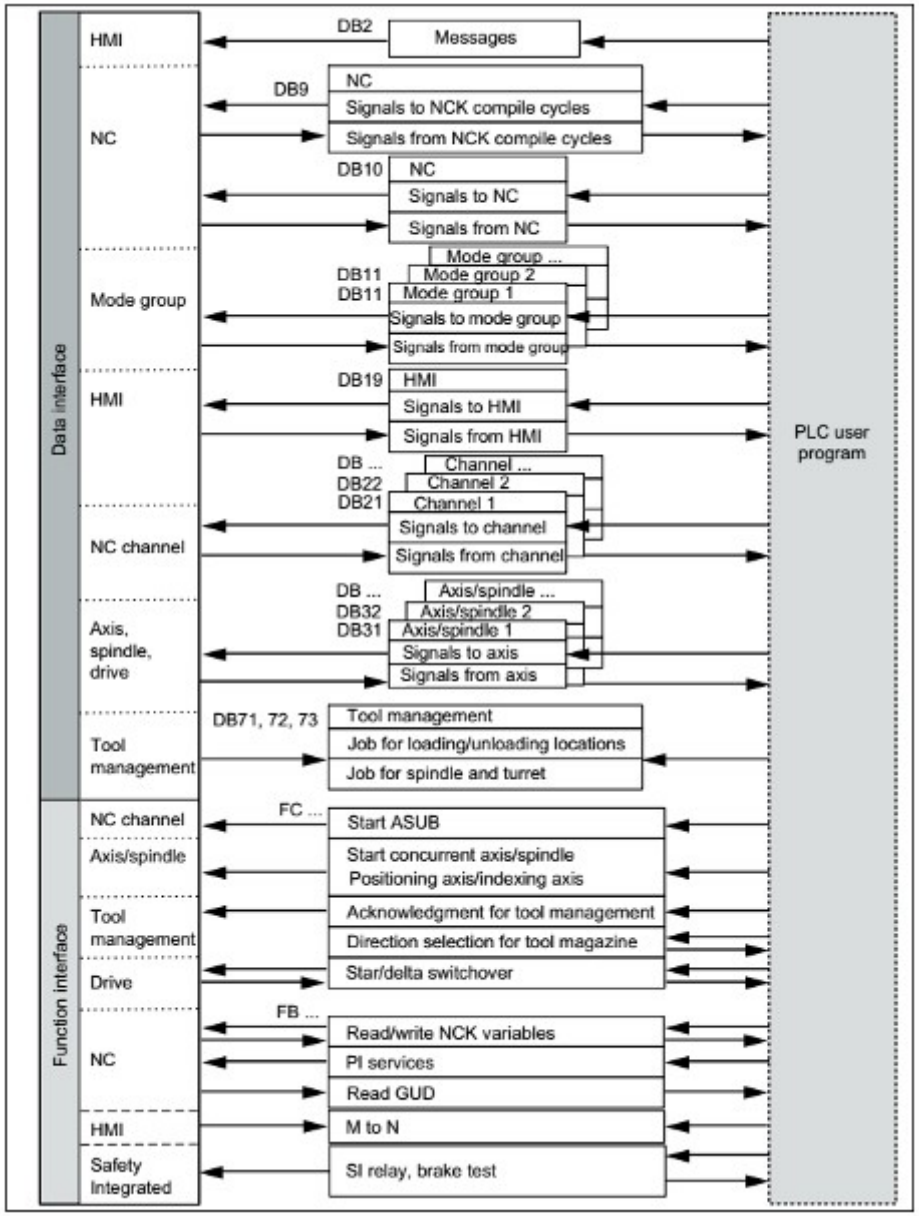
- **PLC Basic Program** is built by adding Standard FBs, example Axis, Spindle,.. to the project depending on the machine Kinematic Structure, example : milling, turning,… The PLC Basic Program organizes exchange of signals of data between the various components of the NCU – PLC, NCK, HMI, MCP
- **Drive Configuration** for Spindle, Axes travel, interpolation, resolutions,...
- **NCK Configuration**
  - Drive, PLC, HMI, MCP Communications
  - Scaling Machine Data
  - Parameterization axis, spindle data & Measurement Systems
- **PLC User Project** - use subroutines supplied by the basic program and implement the logic operations and sequences of the machine



Data Exchange between PLC, NCK, HMI, MCP, Drives

- **Cyclic at start of every cycle of PLC OB1**
  - PLC to NCK Commands
  - NCK to PLC Status
  - For Mode Groups, Channels, Axes/Spindle, NCK
- **Event**
  - NCK→ PLC : PLC functions to be executed as part of the workpiece program
  - PLC → NCK : PLC request to NCK (eg traversal of auxillary axis)
- **Messages**
  - HMI /MCP ← → PLC

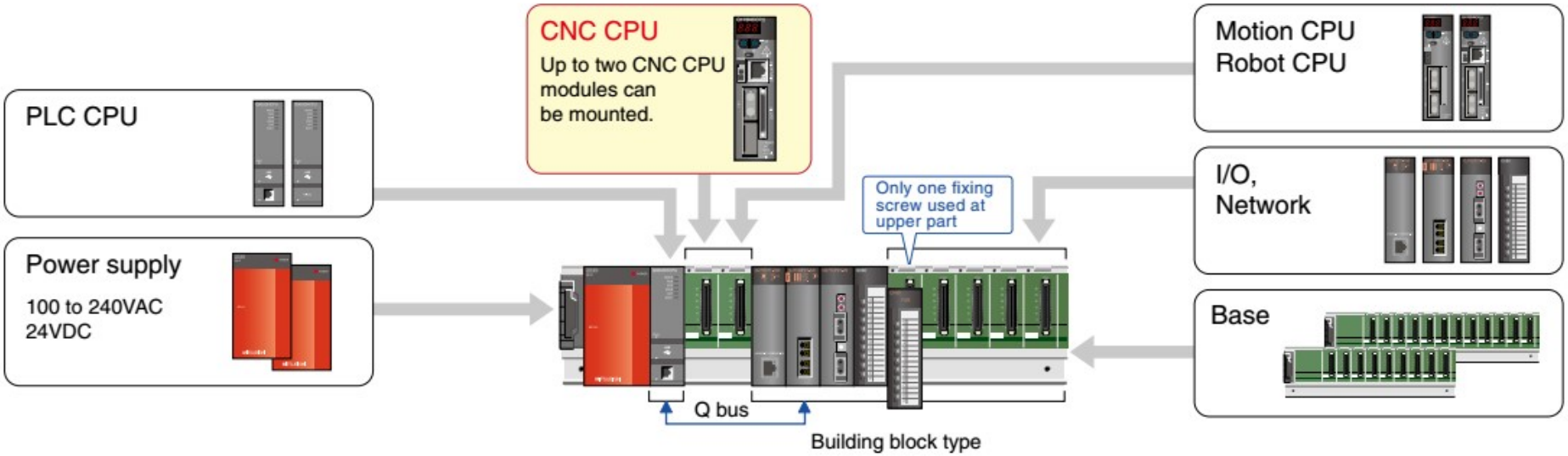
PLC-NCK User Program Interface



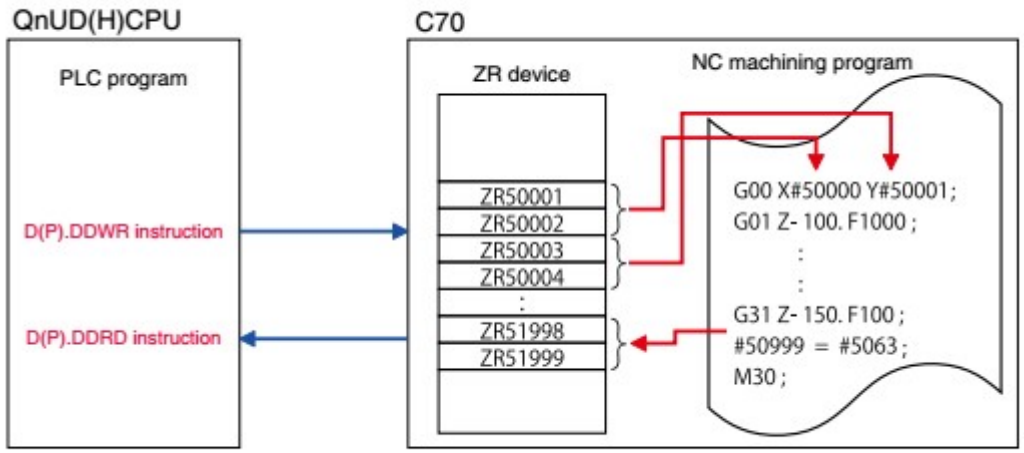
La CNC et les ROBOTs

- **The Robot is the CNC**
  - Example : **KUKA.CNC** complete software-based CNC implementation for execution of machine tool code (G-code) directly on the Robot Controller. This turns the robot, with its accuracy and stiffness, into a machining center for path supported processes. Example : Drilling, Welding…
- **The CNC is the Robot**
  - No separate Robot Controller is required as the NCK is used to calculate the Robot's inverse-kinematic equations and control the Robot as a channel of the CNC
  - Robot Kinematics are also available for Siemens NX CAM for part-program code generation
  - Sinumerik's **RunMyRobot /DirectControl** extension enables programming, control and diagnostics of the supported Robots (Kuka, Comau, etc..) from the CNC HMI
  - Industrial sectors targeted are additive manufacturing, fiber placement, metal-cutting, carbon fiber-reinforced polymer, and laser machining.
- **The CNC and the Robot co-operate together**
  - ***CNC's built-in PLC*** communicates with the Robot Controller - typically for job part loading, unloading…

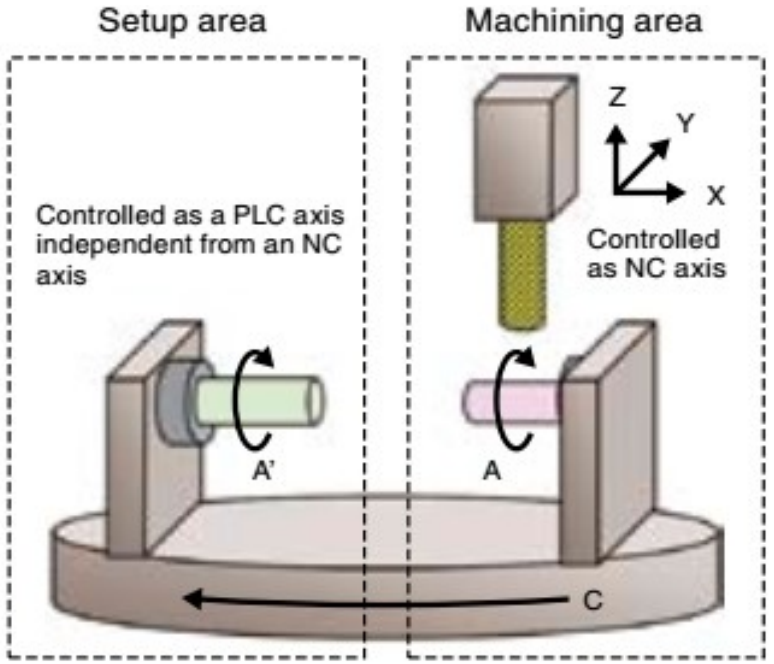
Mitsubishi PLC, CNC, Robot on the same Base



IOs may be assigned to PLC-CPU or CNC-CPU or Robot-CPU



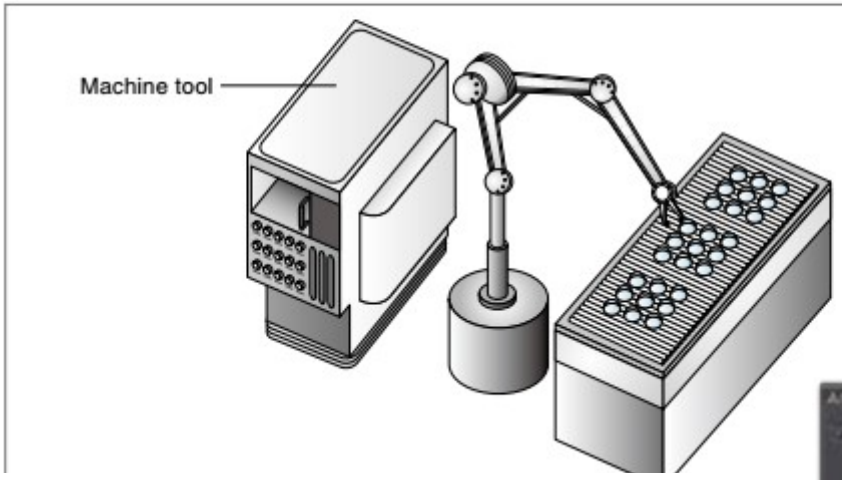
Extended macro interface for system variables



Example of PLC axis mixed control with a pallet changer

Processing robot cell

One PLC CPU plus up to three other CPUs (CNC, robot and motion controller) can be mounted on a single base.
Note that if two CNC CPUs are used, up to three CPUs including the PLC CPU can be mounted. ⇒ helps reduce size and wiring of the control panel.
Each CPU transfers the data using a high-speed bus. ⇒ helps reduce cycle time.



PLC CPU: controls peripheral device
CNC CPU: controls machine tool
Robot CPU: controls robot

Allows editing of C70's ladder program on a GOT

Machine operation panel computerized and aggregated

Possible to operate machine with the touch panel screen, instead of the conventional machine operation panel.

Conventional machine operation panel

GOT Displays
(GOT2000/GOT1000)

CNC monitor installed

CNC monitor has been installed, which allows editing of machining programs and setting each CNC data.

Allied Fields
PAC

National Instruments Labview PAC

http://www.ni.com/en-in/shop/compactrio.html

Packaged Controller

Combines customizable software with powerful processing and I/O for any measurement, control, or monitoring application.

Board-Level Controller

Combines a processor, a programmable FPGA, memory, and I/O in a small form factor for custom embedded design.

Conditioned I/O Modules

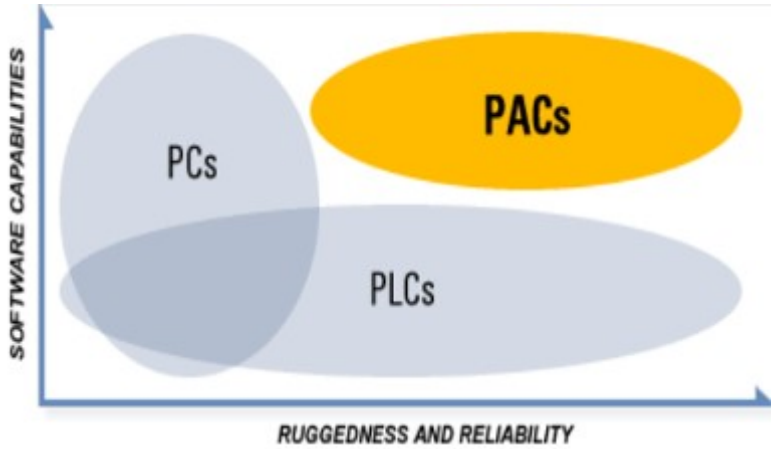
Connects to many sensors and buses and support measurements such as temperature, voltage, resistance, audio frequency, and more.

Software

Includes a portfolio of highly interoperable software products to meet your needs from interactive exploration to custom engineering design.

PCs … PACs … PLCs

Programmable Automation Controllers

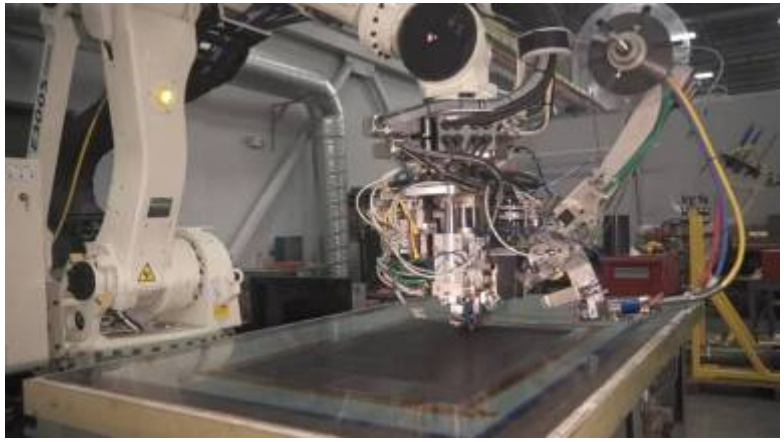


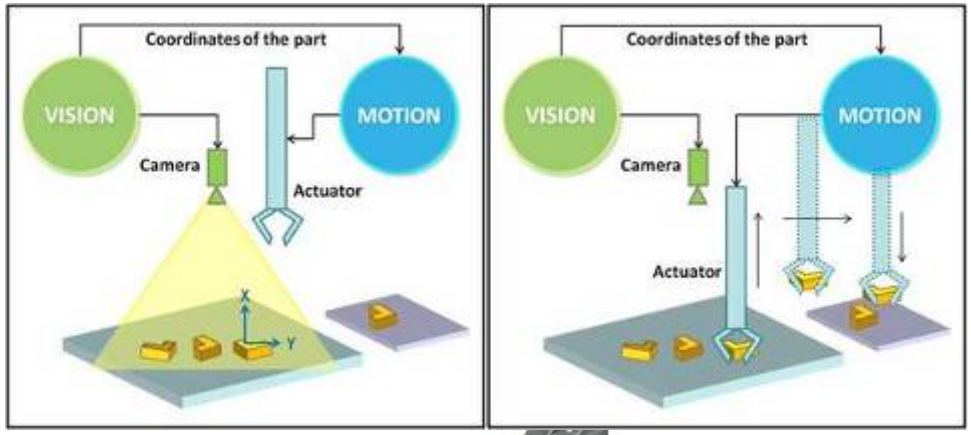
- PACs bridge the gap between PCs and PLCs
- Rugged Industrial Construction
- Real-Time OS
- Multi-processor and Multi-tasking
- Deterministic Scan times for Logic Execution and IO update
- Redundancy as required
- Large Non-volatile memory / SSD to log data. Time stamping in the PAC itself.
- Multiple Network Ports Ethernet, USB, RS485, Field-buses.. Compatible with Enterprise Networks with High Level Protocol Support like OPC, SQL Database, MES, ERP & IoT connectivity, Fault-tolerant File System, Batch Process Control, large number of PID loops and WebServer
- Programming in C, C++ and also Ladder, FB, SFC
- Multi-domain functionality and Multi-discipline Development Platform – Integrated Tag Database, Sequential Control, Process Control, Robotics and Multi-Axis Motion Control, Machine Vision, Communications, Intensive Calculations, Statistical Process Control (SPC), SIL certified modules, Diagnostics, Predictive Maintenance and Operations Monitoring, Testing and Quality Control
- Open Modular Architecture with support to large number of Digital and Analog IO and custom extension IO interfaces with user-programmable FPGA interfaced to the PAC’s data-bus / mapped to the PAC memory.
- In-built display port & visualization capabilities so the logic and HMI is developed in the same software package

PAC *vs* PLC & PC+SoftPLC

- PLCs are becoming more powerful and feature-rich
- PLC vendors now designate their high-end PLCs as PACs
- PCs are becoming rugged, compact and can be configured for real-time support.
- Soft-PLC systems running on Compact PCs could provide most of the features of a PAC.
- A feature-rich PLC + Compact PC with SCADA, Databases and customized-applications written in VB, C++ could also provide most of the features of a PAC.
- PAC required for high-performance applications with many interfaces

Interface - Motion, Vision, Test Instruments, Data Log

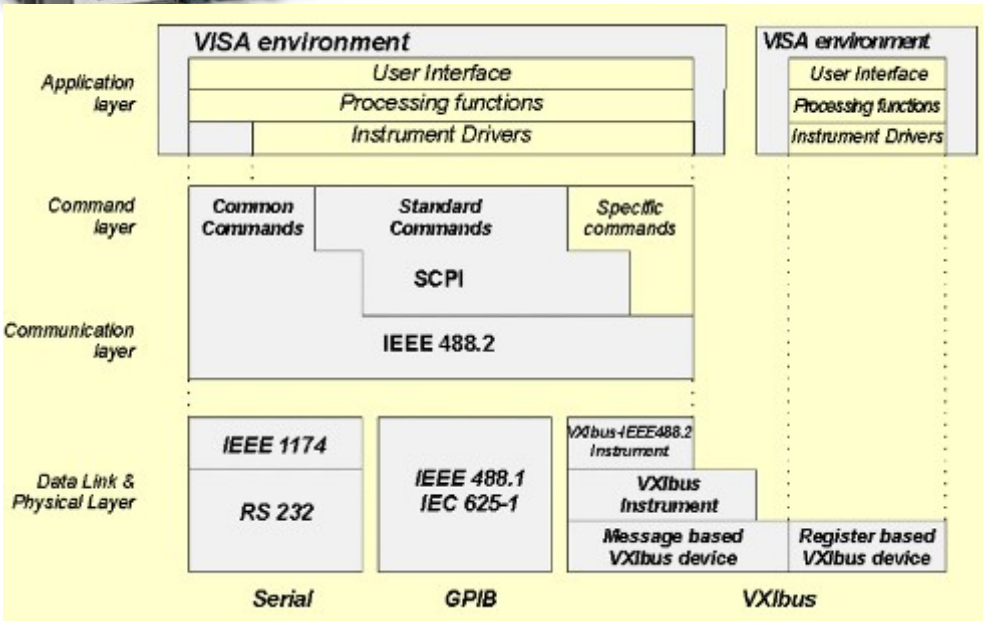




NATIONAL INSTRUMENTS™ LabVIEW™

Typical Application of PAC performing all the below functions

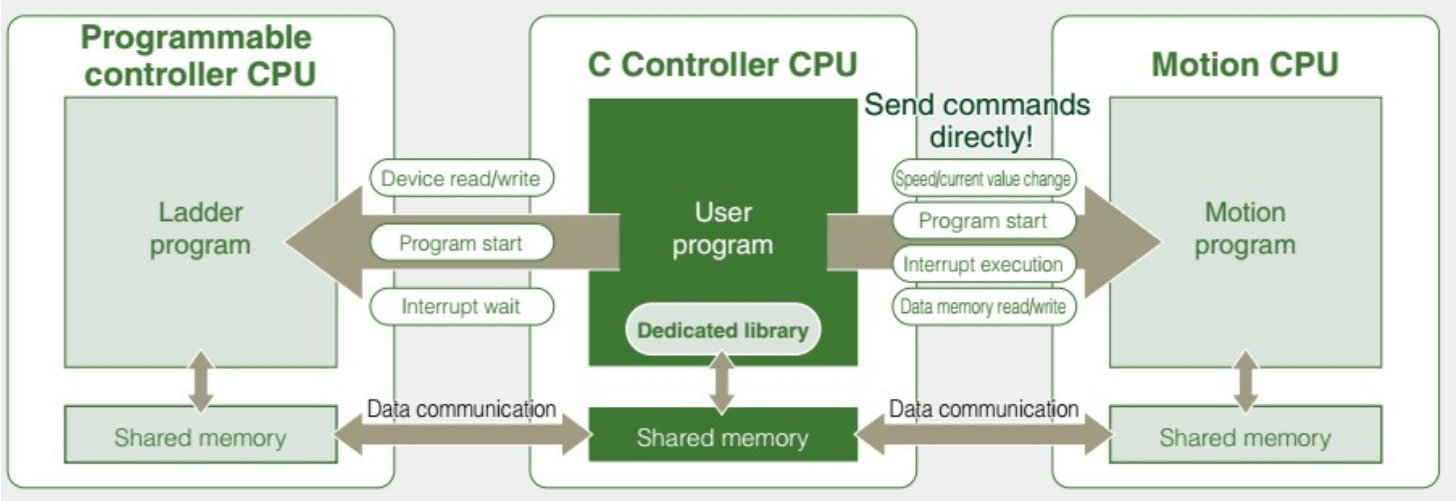
- Vision based Multi-axis Motion Control
- Quality Inspection using Vision, interfaces with Instruments using GPIB, SCPI, VISA
- SPC
- High-Speed Data-Logging



PLC as an Embedded Platform

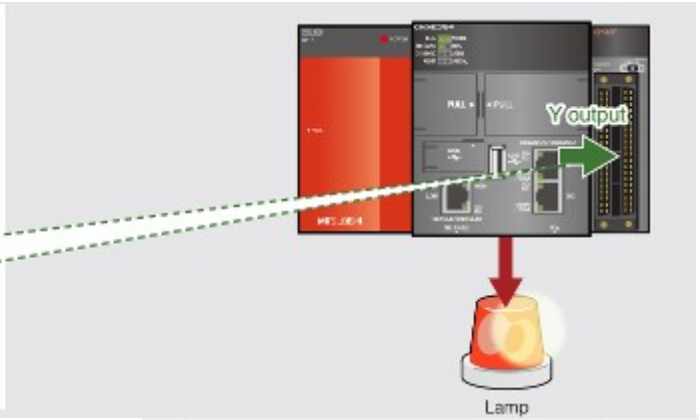
PLC as an Embedded System Platform

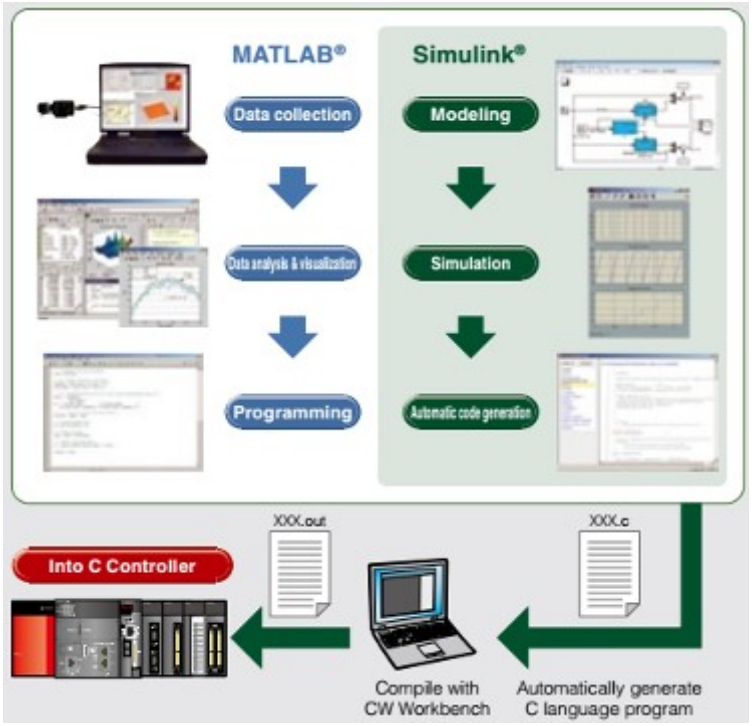
- Embedded Controller based on Mitsubishi iQ PLC system Architecture
- VxWorks RTOS
- C-language programming
- Auto-code Gen from Matlab-Simulink
- Industry Proven and Certified Hardware
  - API access to all iQ series PLC hardware and Communication Interfaces
  - Multi-processor Mode with -PLC-CPU Motion Controllers, Robot Controllers, CNC, .. **(effectively as a PAS)**
  - Standard Mitsubishi HMIs can be used (as they can access variables in the embedded controller)
  - Large 16GB+ data storage, High Speed Data Logging, FTP
  - PCI expansion bus
- Guaranteed long support cycles

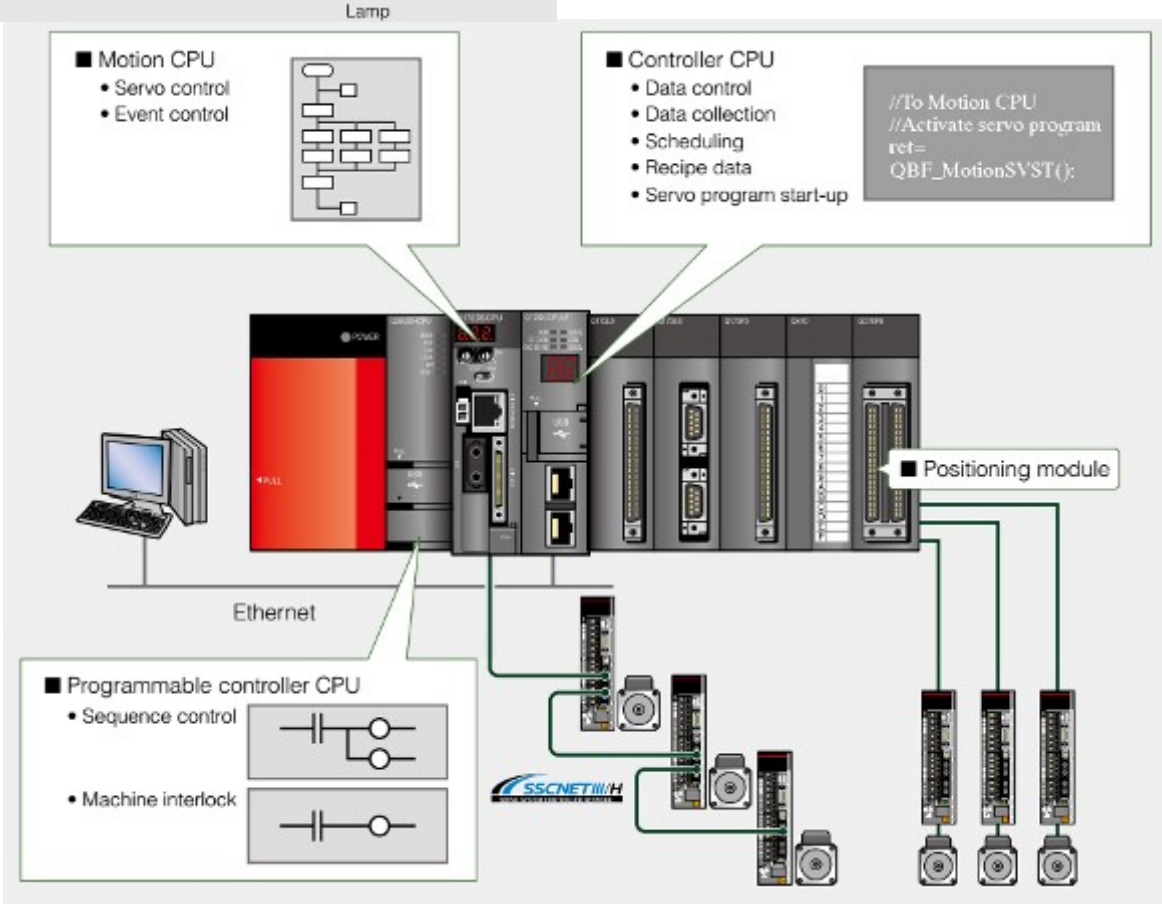


Mitsubishi C-Controller –Embedded Platform

```
void vSample()
{
    short   sRet;   /* QBF Function Return Value */
    long    lPath;  /* Bus Path */
    short   i;

    /* Open Bus */
    sRet = QBF_Open(2, &lPath);

    for(i = 0; i < 0x40; i++){
        /* Read data from Y output */
        sRet = QBF_Y_Out_BitEx(lPath,0,i,1);
    }

    /* Close Bus */
    sRet = QBF_Close(lPath);
    return;
}
```







High-Speed Data-Logging
- Local at PLC

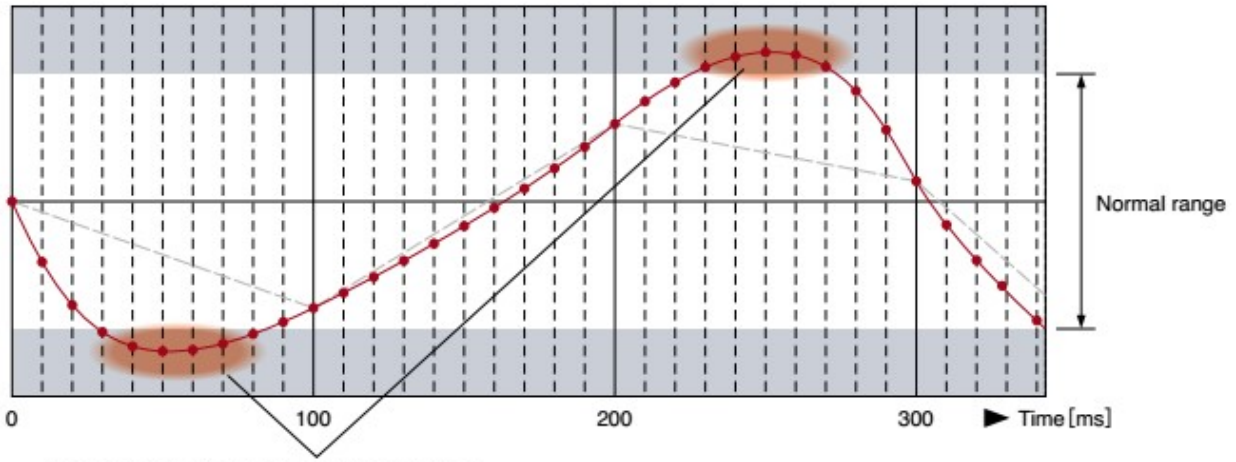
Fast Local Data Logging @ 1ms

SCADA/Historian/SQL PC-based logging is typically slower. Refer earlier Historian Slide

■ **High Speed Data Logger Module sample data at 10ms intervals** (fastest down to 1ms)

Control value
Control value (Personal computer)

The control value has exceeded the limit. Let's adjust the equipment!



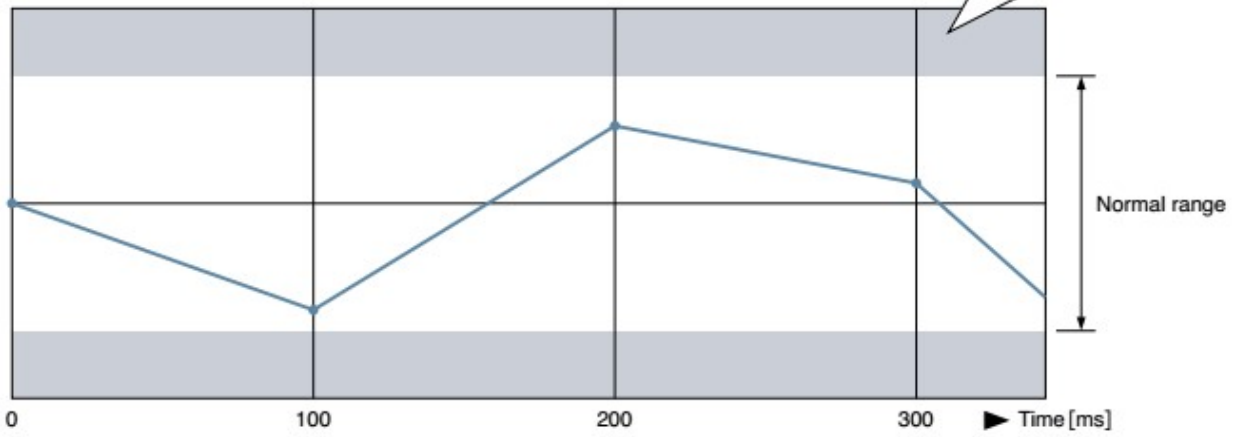
Existing data logging might miss the causes of the errors.

■ **Generic sample data from personal computer or external device at 100ms intervals**

Control value

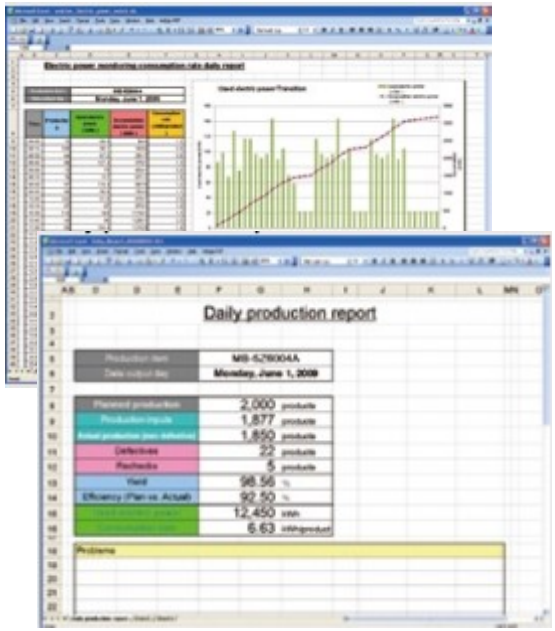
Logging data does not seem to indicate any problems...

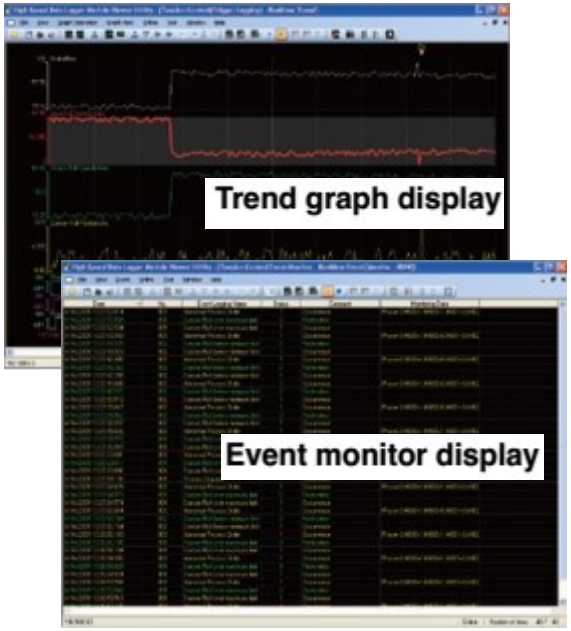
Error causes cannot be detected with the logging data.

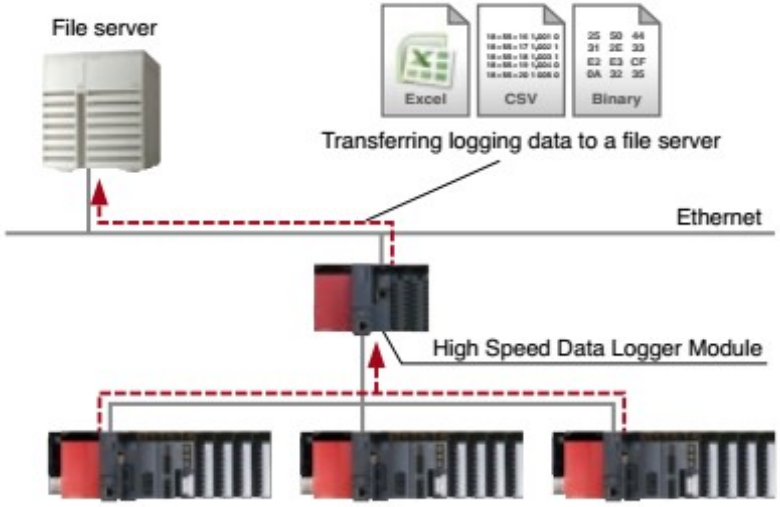


Other Features/Advantages of PLC based Data-Logging

- Used to capture high-speed events
- Logging is set-up using simple configuration
- Trigger based logging
  - Only log data near trigger events, thus saving memory space
- Time Synchronization with SNTP Server
  - Data across multiple PLC systems can be analyzed for plant-wide issues
- Ability to access and log data from multiple PLCs connected by high-speed networks
- FTP Transfer to File Server
- Multiple data-formats, Batch/Lot wise logging possible
- Excel File can be used for Report Generation right on the PLC rack



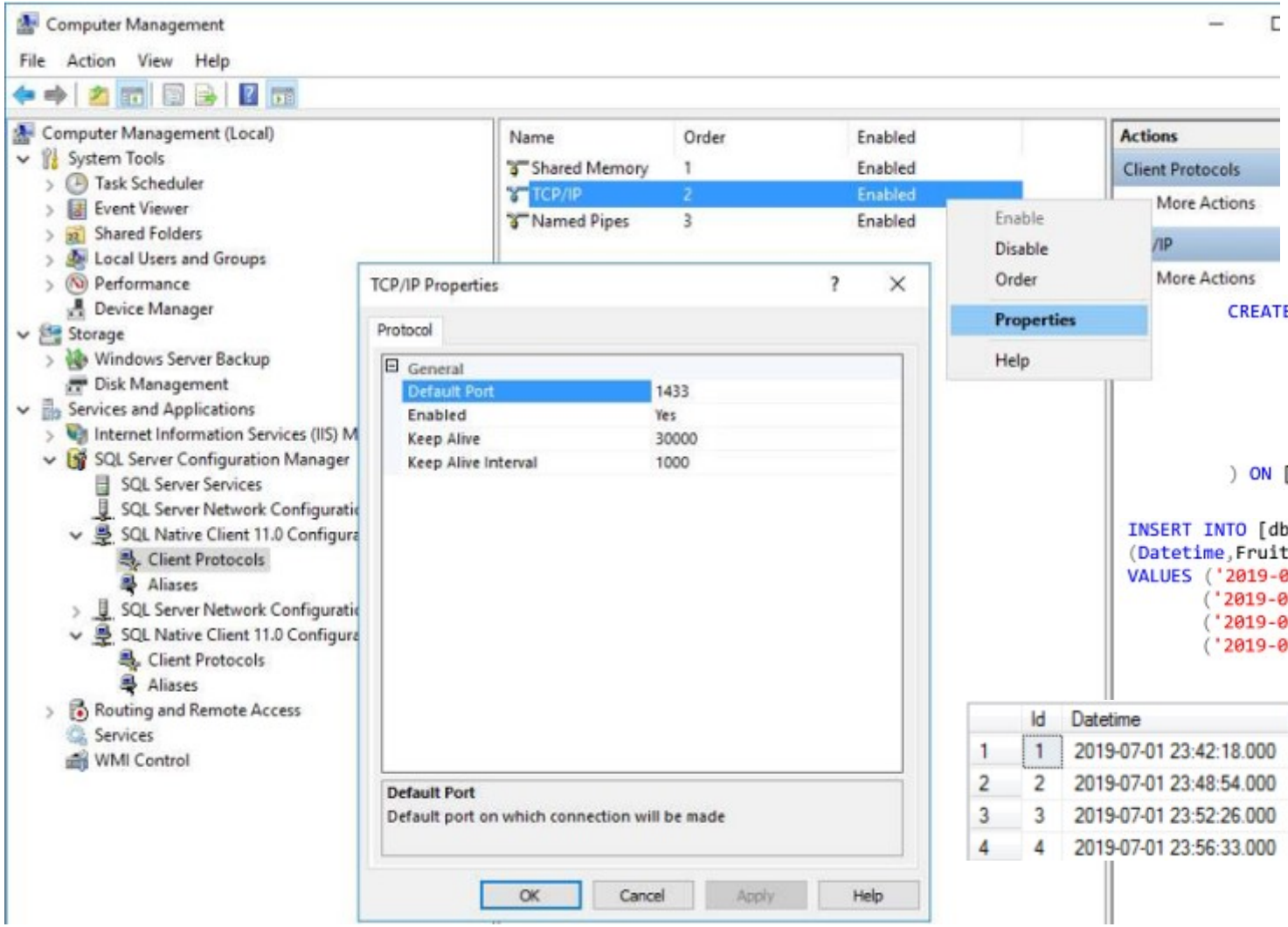




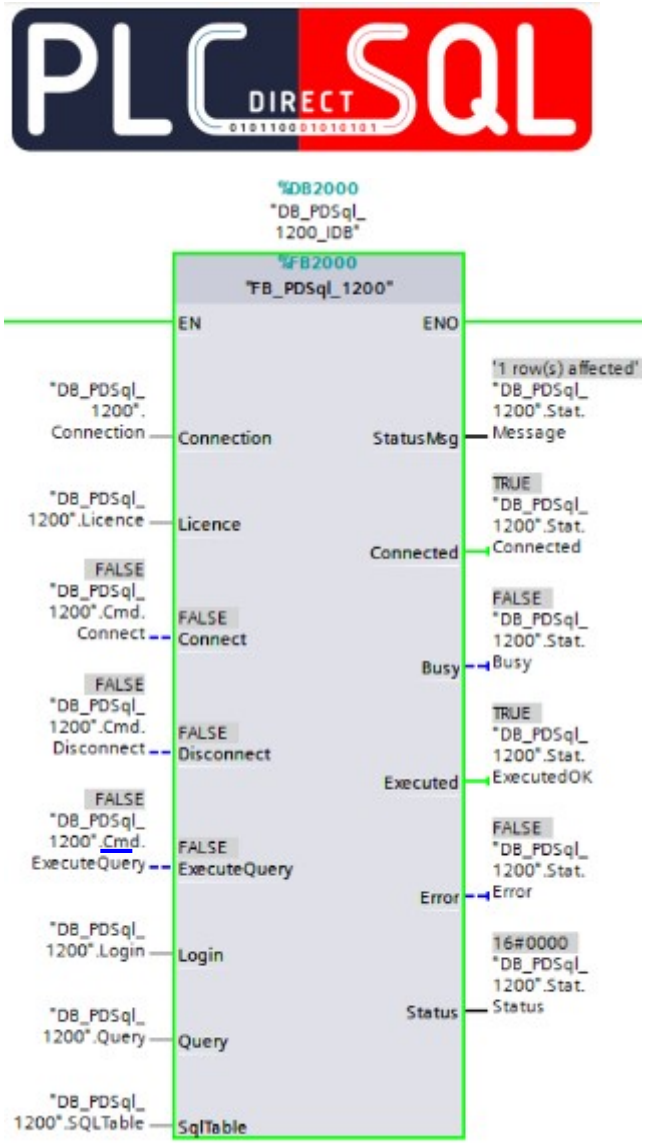
PLC access to Databases

PLC direct access to SQL Database

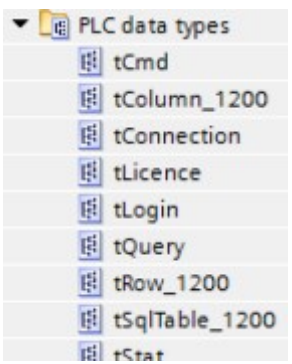
- Configure SQL Server Security, Authentication, TCP/IP Port, DB Tables..
- Configure Firewalls Exceptions for the configured TCP/IP Port

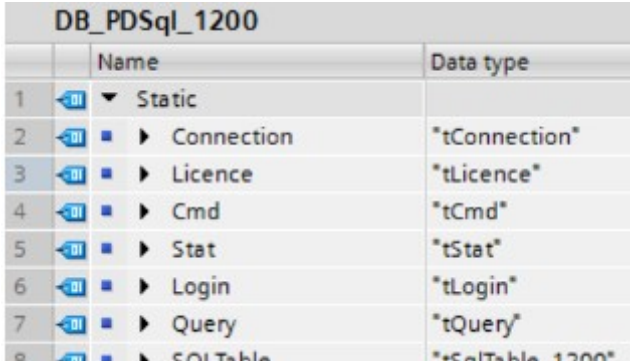


PLC direct access to SQL Database



1. Import the SQL Library and Instantiate the FB and Data-Blocks






```
#dtlDateTime := '2019-07-02 10:42:56';
#sFruitName := 'Banana';
#iQuantity := 24;
#rWeight := 8.48;
#bCitrusType := FALSE;
```

2. Code STL Functions to “connect” to the SQL database and “select”, “modify” or “insert” rows

```
#str := 'INSERT INTO TestTable(Datetime,FruitName,Quantity,Weight,CitrusType) VALUES($'';
#str := CONCAT(IN1 := #str, IN2 := "fcDTLString"(#dtlDateTime));
#str := CONCAT(IN1 := #str, IN2 := '$',$'');
#str := CONCAT(IN1 := #str, IN2 := #sFruitName);
#str := CONCAT(IN1 := #str, IN2 := '$',');
#str := CONCAT(IN1 := #str, IN2 := INT_TO_STRING(#iQuantity));
#str := CONCAT(IN1 := #str, IN2 := ',');
#str := CONCAT(IN1 := #str, IN2 := REAL_TO_STRING(#rWeight));
#str := CONCAT(IN1 := #str, IN2 := ',');
#str := CONCAT(IN1 := #str, IN2 := INT_TO_STRING(BOOL_TO_BYTE(#bCitrusType)));
#str := CONCAT(IN1 := #str, IN2 := ')');

"DB_PDSql_1200".Query.Query[1] := #str;
"DB_PDSql_1200".Query.Query[2] := '';
```

```
Cmd.ExecuteQuery := TRUE;
```

3. Trigger the SQL Query

	Id	Datetime	FruitName	Quantity	Weight	CitrusType
1	1	2019-07-01 23:42:18.000	Apple	10	2.54	0
2	2	2019-07-01 23:48:54.000	Orange	16	3.28	1
3	3	2019-07-01 23:52:26.000	Lemon	8	1.64	1
4	4	2019-07-01 23:56:33.000	Mango	7	4.32	0
5	5	2019-07-02 10:42:56.000	Banana	24	8.48	0

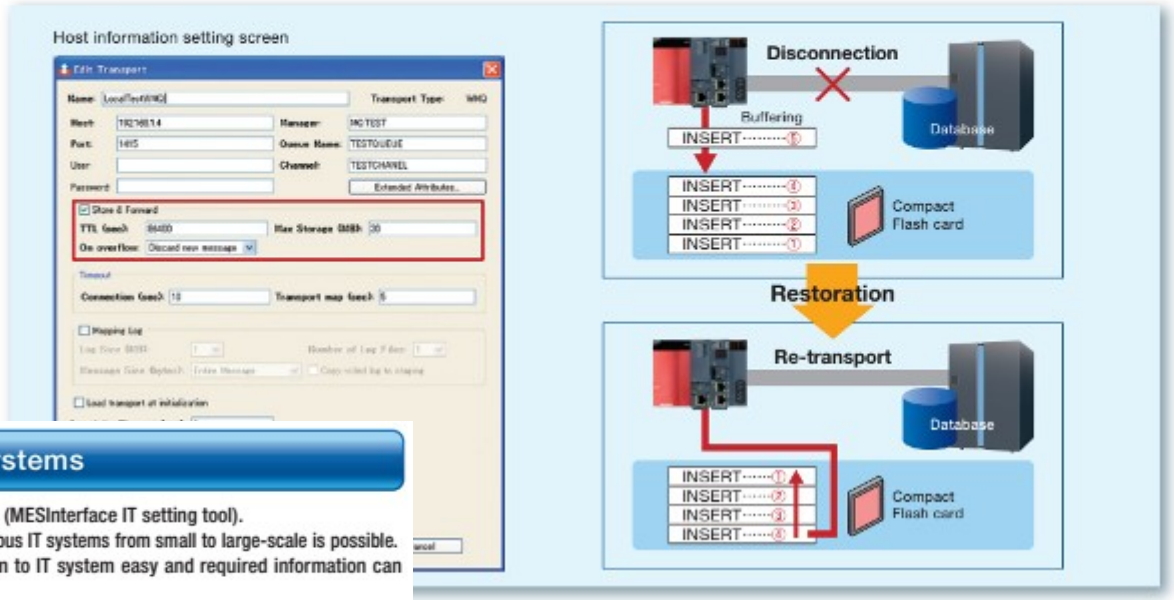
4. Query the Database and confirm that the data is added

Mitsubishi MES-Interface IT Module

No need for communication gateway PCs or programs!

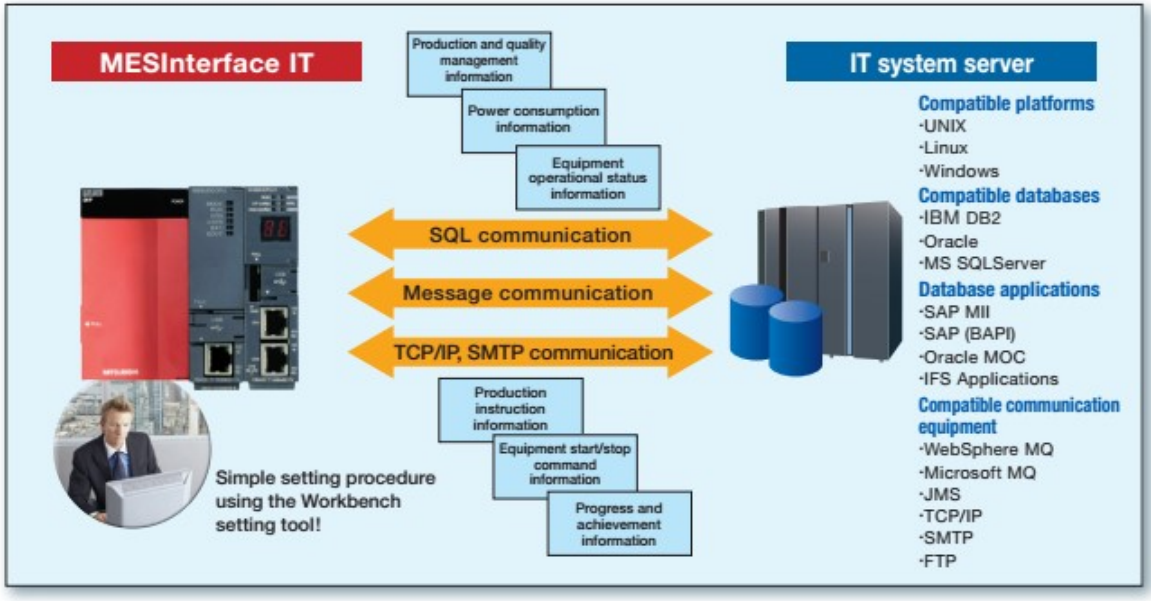
Store & Forward function ensures reliable data collection

●Simply by placing a check in the "Store & Forward" option box in the setting screen of the host IT system server, MESInterface IT automatically buffers the data from the shop floor on a CompactFlash card. This ensures full transport of data even when a communication error occurs, because the buffered data is automatically forwarded to the IT system after the connection is restored.



Provides easy connection to diverse information systems

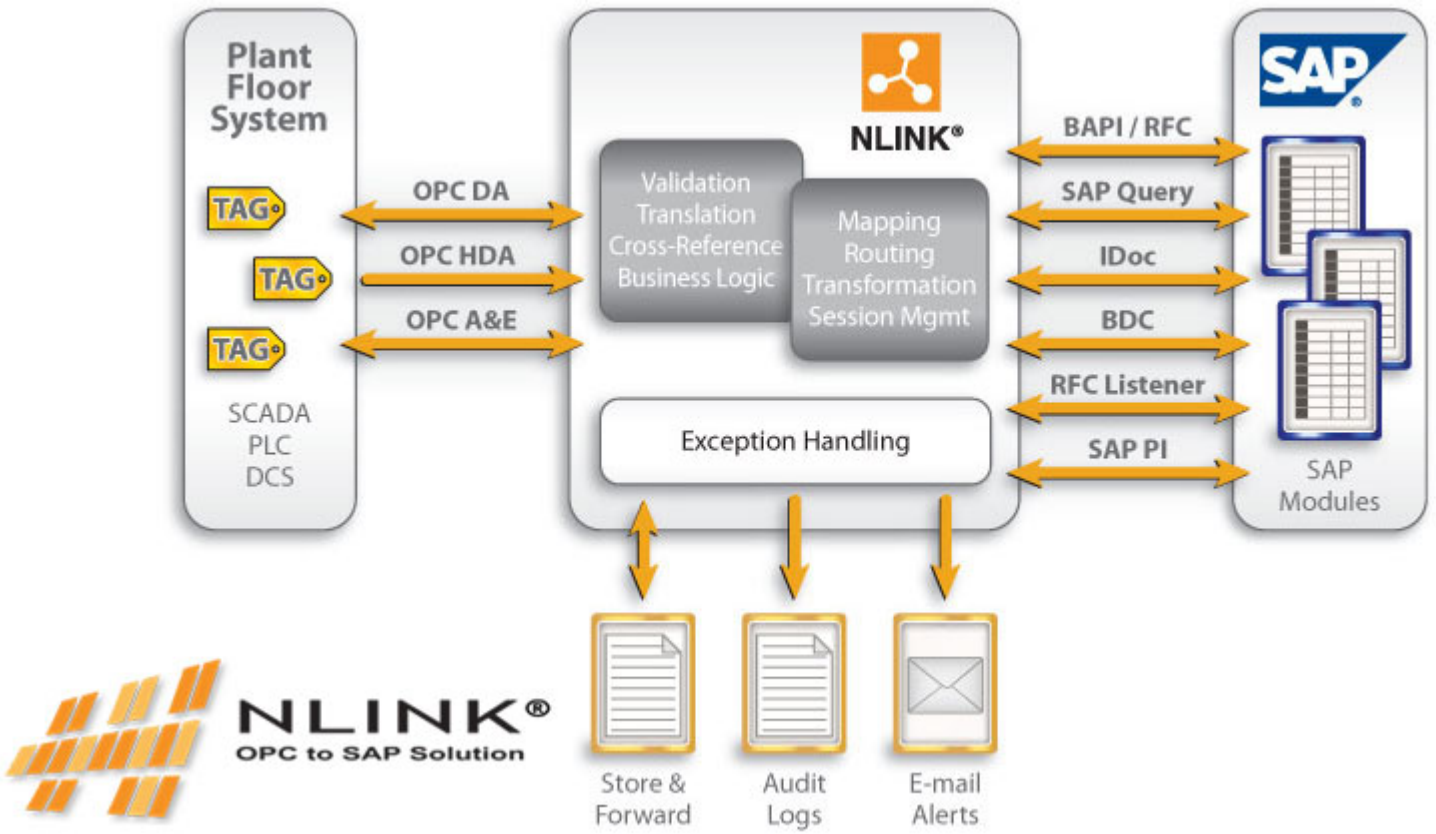
●Connectivity is bi-directional and occurs by mapping data using the Workbench Software (MESInterface IT setting tool).
●Corresponding to a wide range of platform and database, and direct communication with various IT systems from small to large-scale is possible.
●The correspondence of diverse communication functions makes the data communication to IT system easy and required information can be communicated when needed.



1. From the Production Control Database acquire Target Production Number of Units, Lot Numbers, Serial Number Sequence – this is stored in the local database of the MES-Interface IT Module
2. As PLC-CPU completes execution update the date-time of manufacture, Serial Numbers and quality related parameters in the Production Control Database

PLC and ERP(SAP)

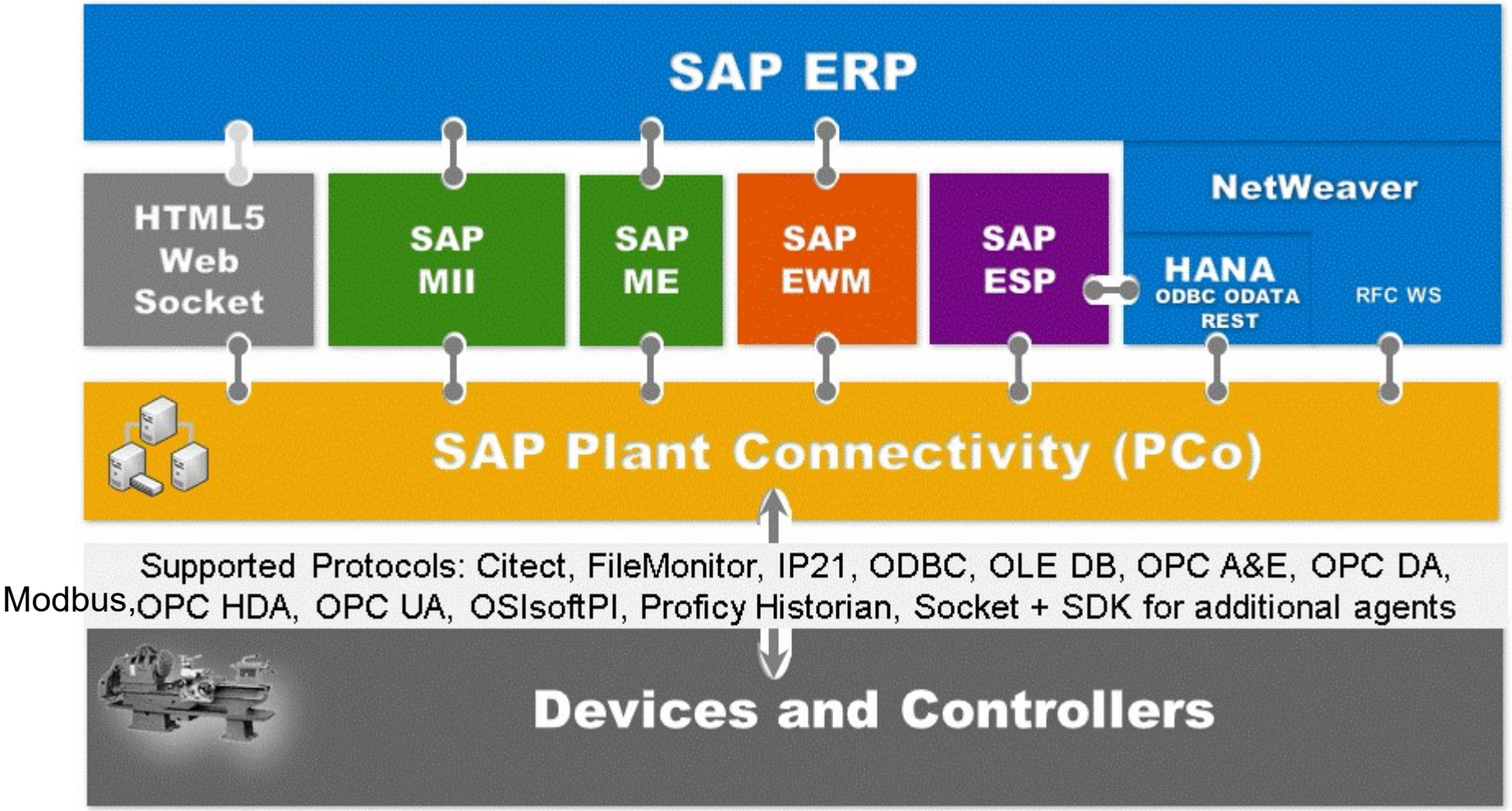
Using 3rd Party Software



Despite extensive SAP interfaces available for automatic data transfer with the Industrial Automation System – many IT Departments may conservatively insist on .csv file load/dump using a programmed SAP GUI Buttons for Import/Export, where the user controls the data exchange and data is validated as per the SAP GUI rules similar to manual data entry

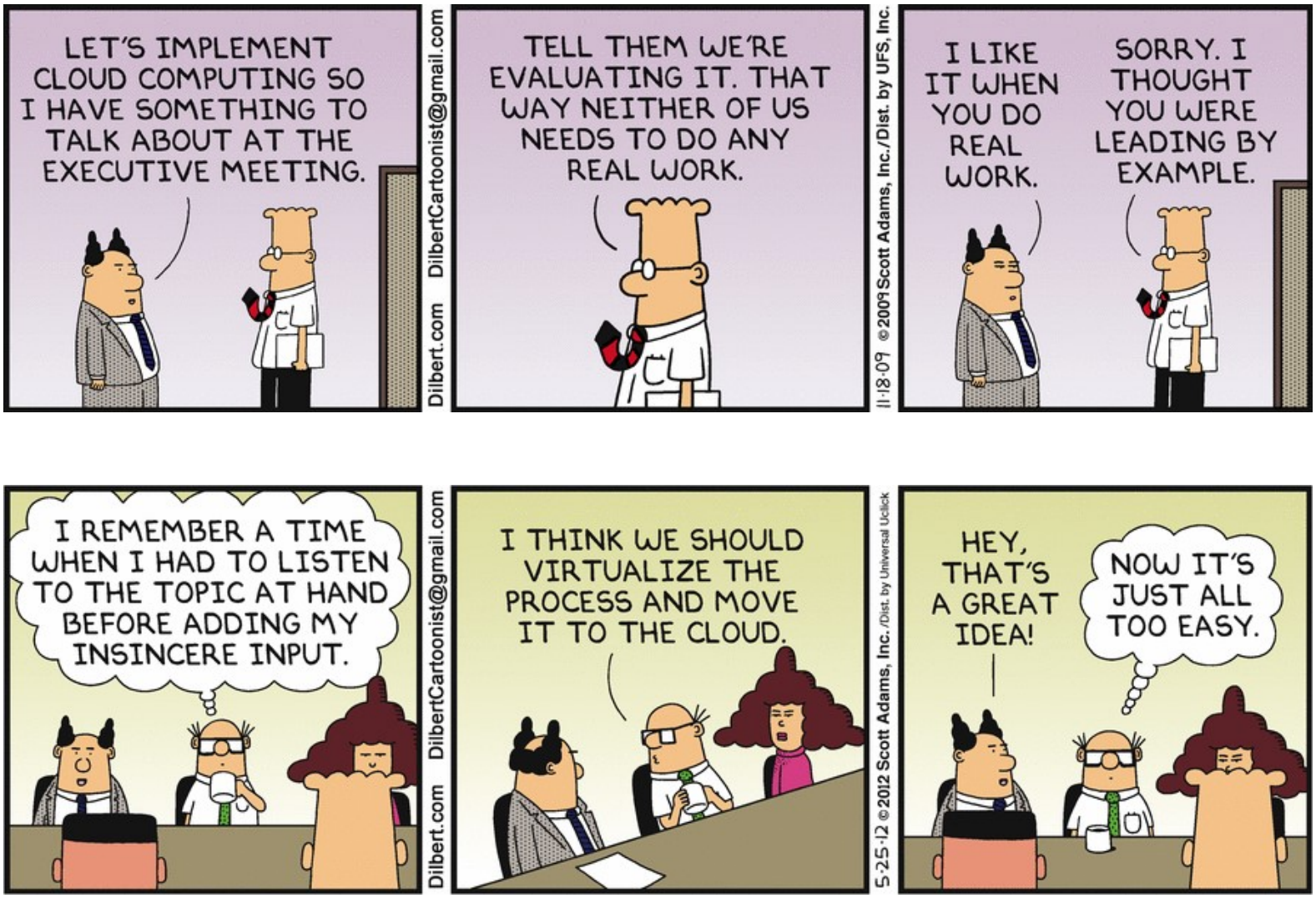
High License Costs for additional connectivity software components may also be the reason

Using SAP Plant Connectivity Component



Connectivity with SAP can be at the PLC, SCADA or MES layer as per the requirement

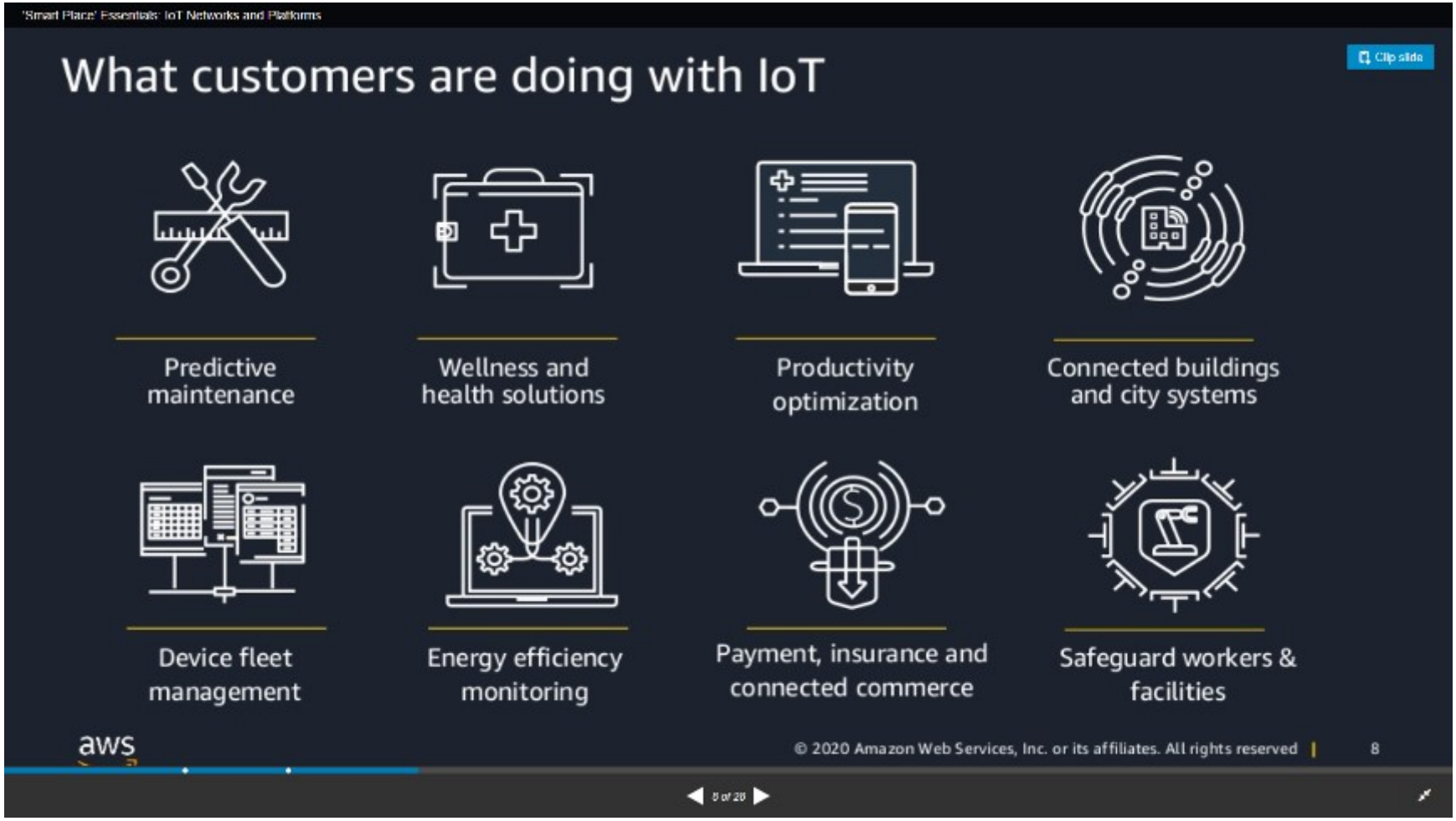
PLC and IoT


LET'S IMPLEMENT CLOUD COMPUTING SO I HAVE SOMETHING TO TALK ABOUT AT THE EXECUTIVE MEETING.
Dilbert.com DilbertCartoonist@gmail.com
TELL THEM WE'RE EVALUATING IT. THAT WAY NEITHER OF US NEEDS TO DO ANY REAL WORK.
11-18-09 ©2009 Scott Adams, Inc./Dist. by UFS, Inc.
I LIKE IT WHEN YOU DO REAL WORK.
SORRY. I THOUGHT YOU WERE LEADING BY EXAMPLE.
I REMEMBER A TIME WHEN I HAD TO LISTEN TO THE TOPIC AT HAND BEFORE ADDING MY INSINCERE INPUT.
Dilbert.com DilbertCartoonist@gmail.com
I THINK WE SHOULD VIRTUALIZE THE PROCESS AND MOVE IT TO THE CLOUD.
5-25-12 ©2012 Scott Adams, Inc./Dist. by Universal Uclick
HEY, THAT'S A GREAT IDEA!
NOW IT'S JUST ALL TOO EASY.

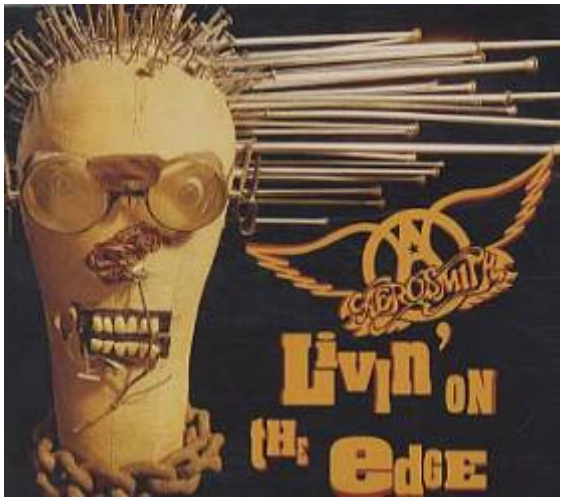
IoT Applications and Advantages

- Typical Applications
  - Log and Analysis of sensor information for preventive maintenance, productivity optimization and energy efficiency monitoring
- Advantages of IoT with Cloud Computing
  - Data Aggregation
  - Data Storage – huge capacity and redundancy
  - Data Analysis, Modeling and Simulation, Optimizations – access to huge computing power and latest technologies, GPUs, AI software stacks, floating software licenses on-demand
  - Monetization of Data – lower down time, better efficiency, increased production through-put, improved safety

Industrial IoT (IIoT)



The IIoT Edge



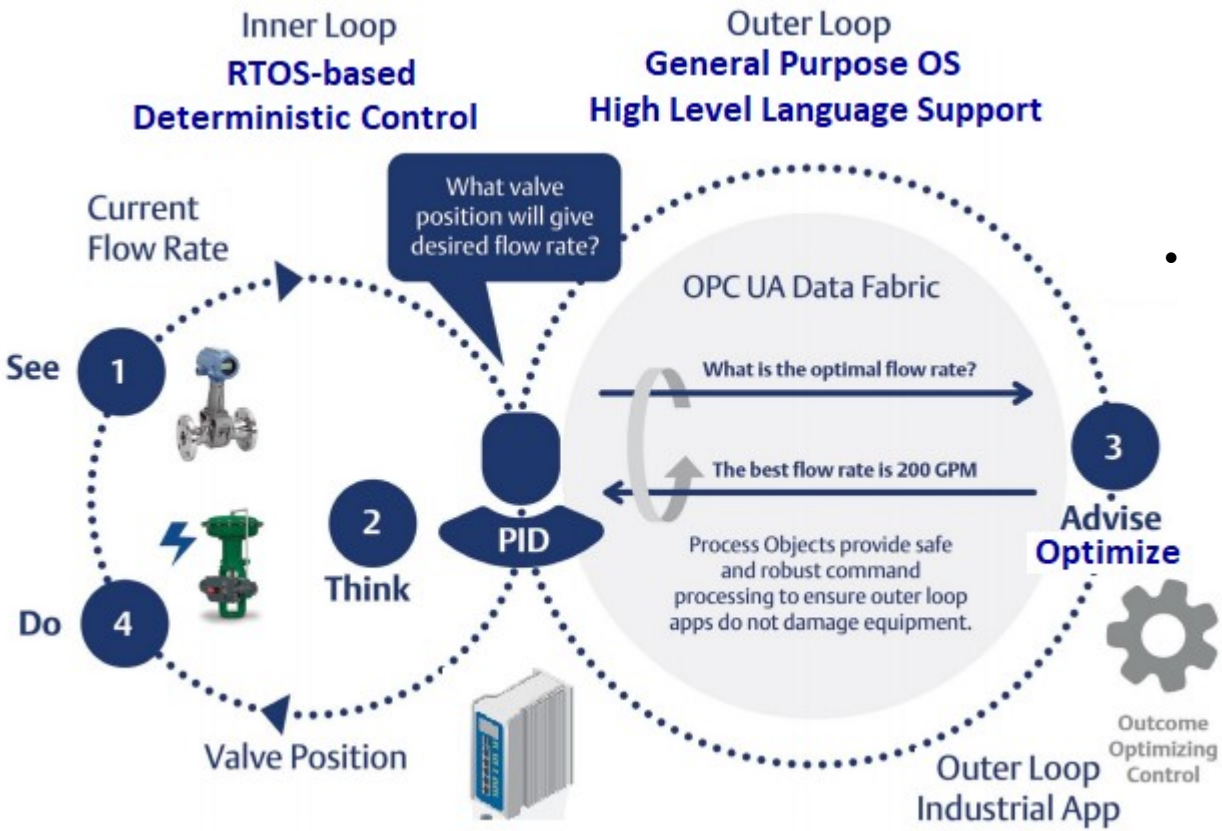
- Industrial *Control* Applications are implemented on the Edge
  - Latency Sensitive, need quick response
  - Economics, cost of moving data that could be reduced after some processing
  - Regulatory, data must be maintained on-site as per requirement of Regulatory Authorities

Industrial IOT (IIOT) Devices

- Edge Gateway
  - Collects Data from Sensors and Controllers and offloads the Data for ***Processing*** to the Cloud or Higher-Level Computing System
- Edge Device
  - Collects Data from Sensors and Controllers and ***Processes*** the Data Locally
  - Suitable for Brown-field projects with existing controllers
- Edge Controller
  - Integrates Control and Data Processing
  - real-time deterministic control & non-deterministic applications that leverage external data to analyze and optimize business operations
  - Suitable for Green-field projects
  - Typically IPC or PAC Type Devices

Devices can be a combination of two or more of the above. Some data being processed locally and pre-processed data offloaded to the cloud

IIOT Loops and Devices

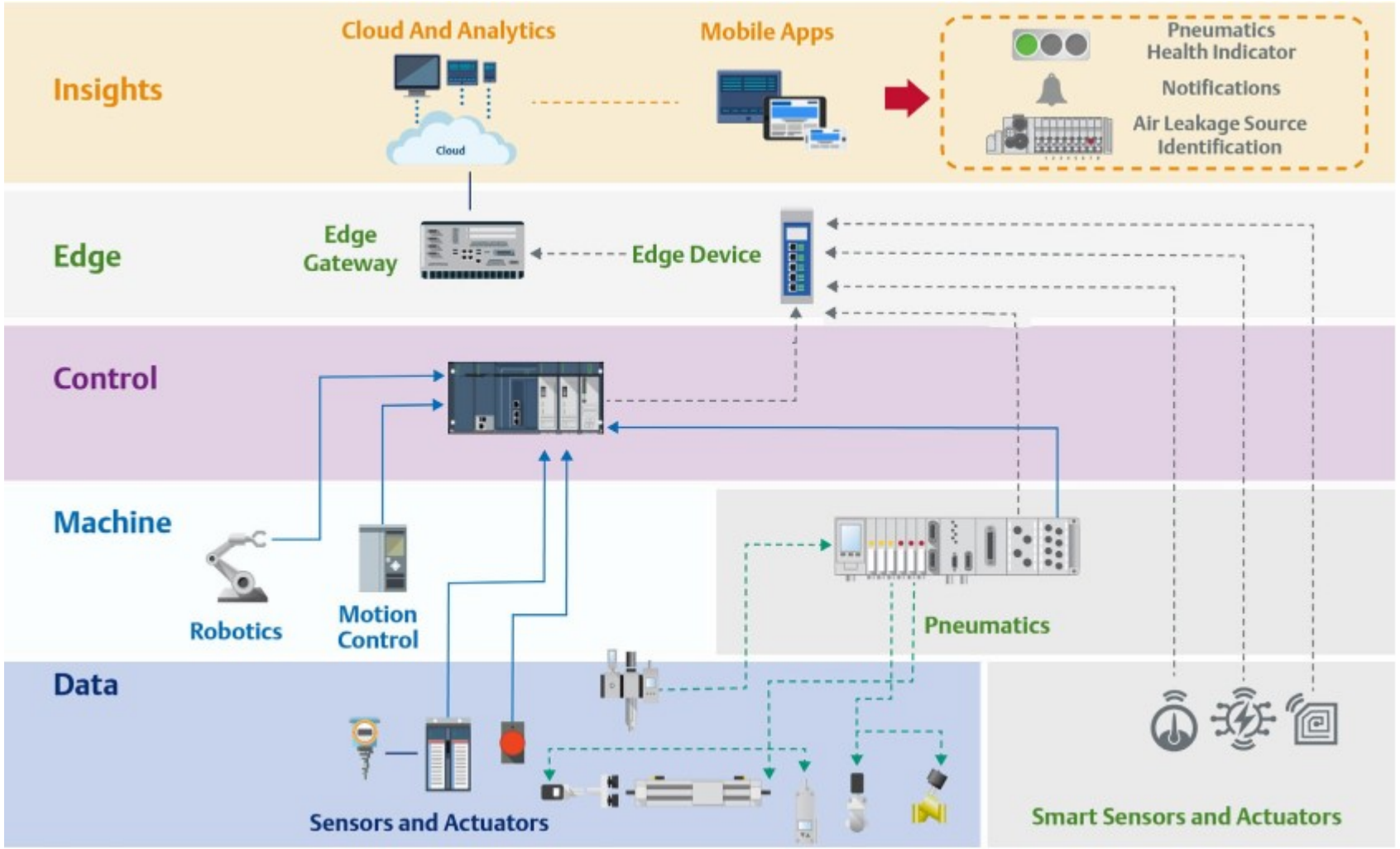


- Edge Gateway
  - Offloads Plant Data to Outer Loop in the Cloud
- Edge Device
  - Executes the Outer Loop in the Plant
  - PLC/Controller executes the Inner Loop
- Edge Controller
  - Executes Both Loops locally

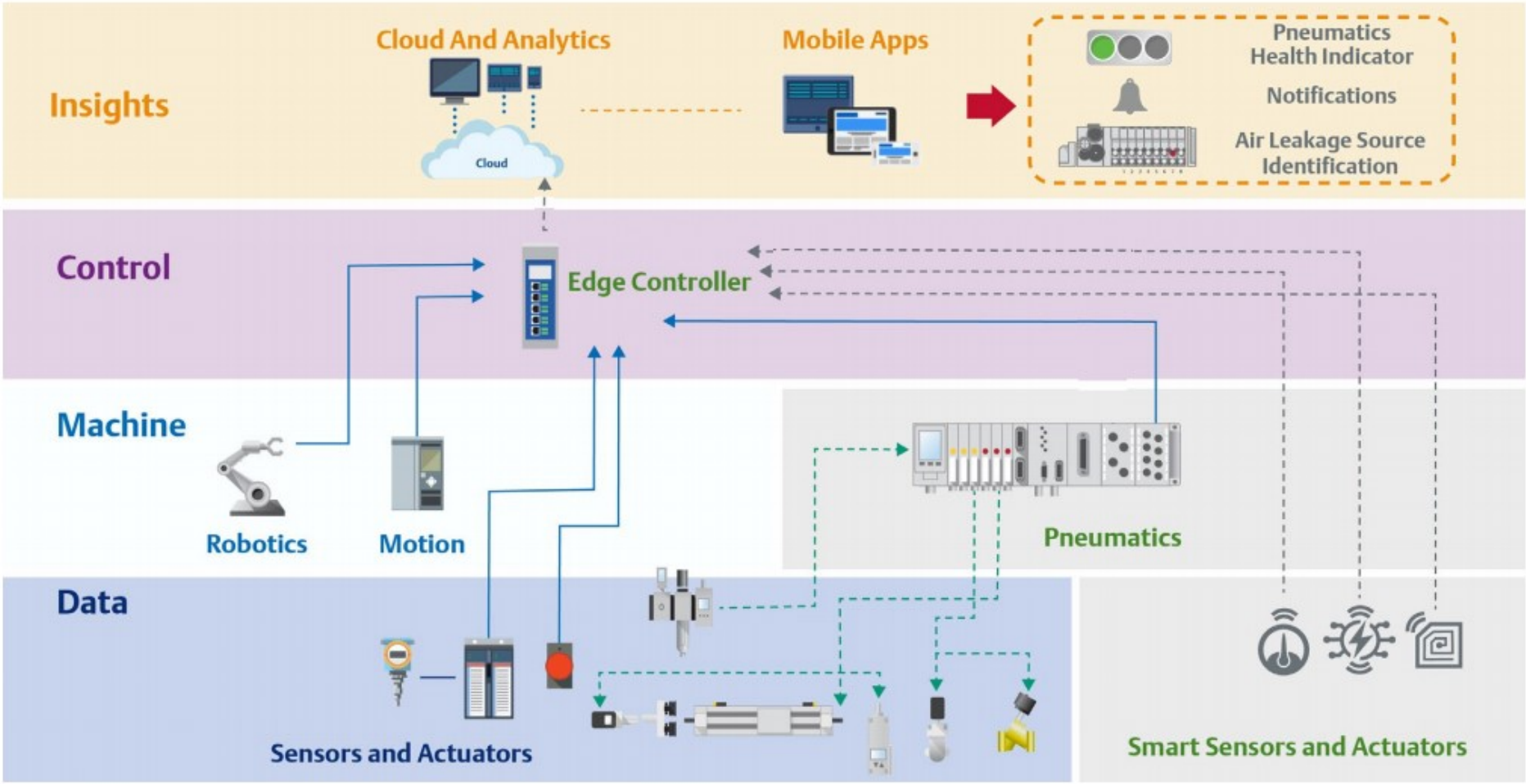
Industrial Simulation and Optimization Software can now run both locally and in the Cloud
Refer Section:
• Plant Simulation & Virtual Reality
• Alternatively general purpose Cloud Analytics and Data Science may be applied to industrial Data

For Hardware Example, refer Slide :
Siemens Soft PLC – **S7-1500 Software Controller**

Edge Gateways & Edge Devices



Edge Controller



MQTT Protocol for IoT

- Very Simple Protocol (~50 page spec)
- Binary Protocol
  - Low Bandwidth
  - Small Memory Footprint
- Widely Accepted in the IoT world
  - Supported by major cloud players like Amazon, Microsoft, IBM
- Reconnection and resending messages possible
- User Authentication with Username & password
- Message can be encrypted using TLS
- Local Message Broker can be used for forwarding to cloud over SSL, VPN

Beckhoff IoT scenarios overview

BECKHOFF

HMI Server in the Cloud

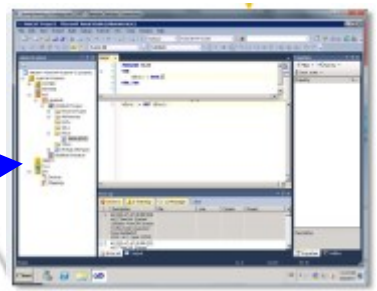
aws

TwinCAT Cloud Control - Virtual Machine

TwinCAT Engineering | TwinCAT Runtime | TwinCAT Analytics | TwinCAT IoT

TwinCAT I/O | TwinCAT HMI | TwinCAT Scope | TwinCAT Database

Message Broker



Engineering/Diagnostics in the Cloud

C6015

MQTT

MQTT

MQTT

MQTT

AWS Lambda function

AWS Greengrass

Beckhoff IoT Gateway

Beckhoff IoT Gateway

Vendor-specific communication via ADS

Standardized communication via OPC UA

to AWS IoT

```
{
  "ProductionLast10s": {
    "Black": 3,
    "Blue": 0,
    "Green": 2,
    "Null": 0,
    "Red": 1,
    "Yellow": 0
  },
  "ProductionTotal": {
    "Black": 6,
    "Blue": 1,
    "Green": 6,
    "Null": 0,
    "Red": 5,
    "Yellow": 1
  },
  "TotalProductionCount": 19
}
```

Field Level

Machine Level

New Beckhoff Controller with TC3

Old Beckhoff Controller with TC2

3rd party control

EK9160

aws IoT Hardware Program

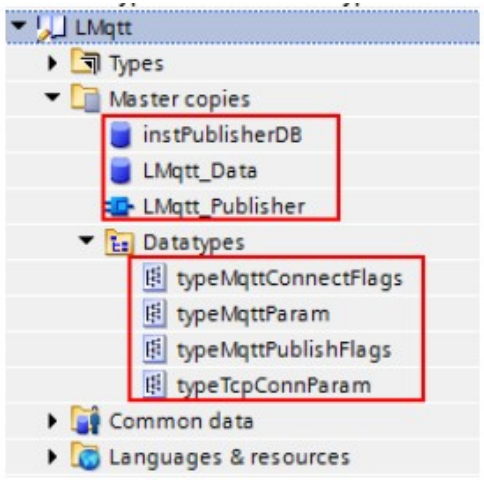
Beckhoff IIoT Devices

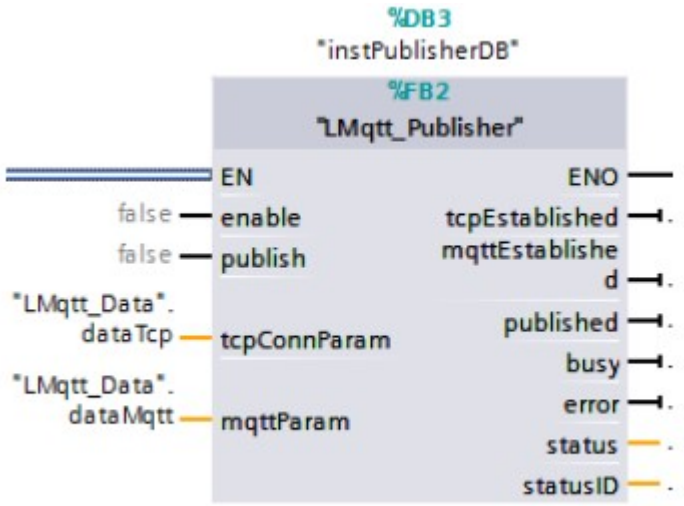
- **TF6701 IoT Communication Function Blocks (software)**
  - IoT Controller & Gateway
  - Function blocks for MQTT communication from within the PLC
  - PLC library "Tc3_JsonXml" to support JSON parsing
- **EK9160 IoT Coupler**
  - IoT Gateway
  - Easily and securely push I/O data to AWS IoT
  - Easy configuration via integrated website
  - Automatic I/O detection
- **C6015 | Ultra compact control cabinet Industrial PC**
  - Configured as IoT Gateway
  - **AWS Greengrass**
  - AWS Lambda function retrieves process data from machine
  - AWS Lambda function publishes aggregated production data

Native MQTT Support by Modern PLCs

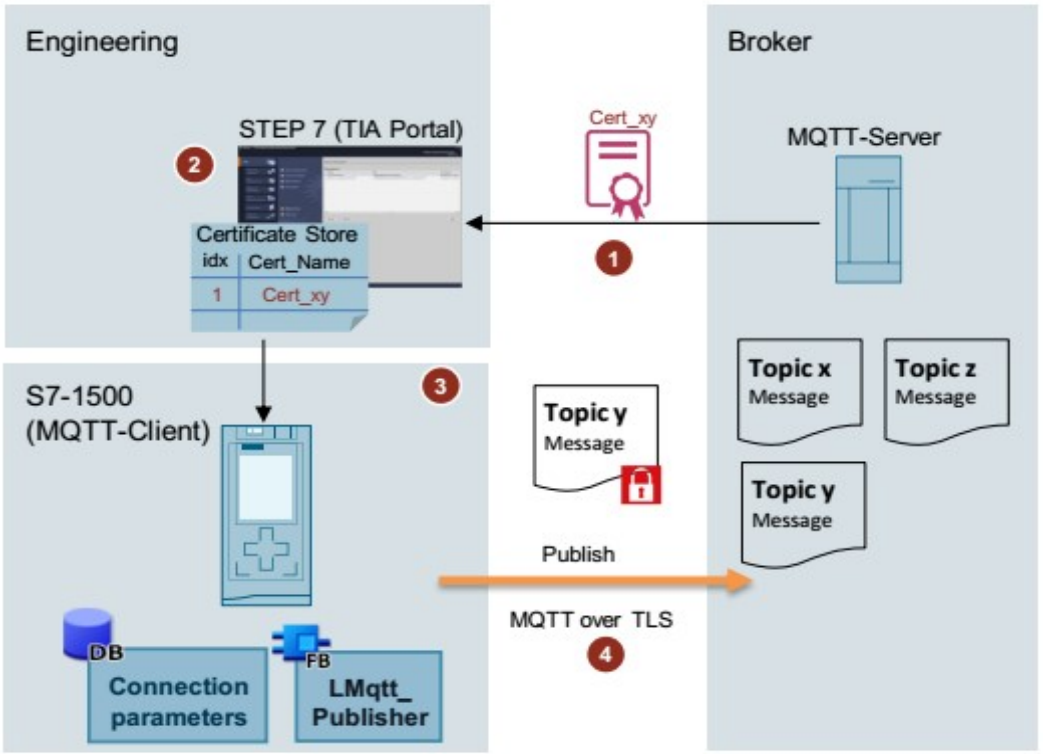
SIEMENS
Ingenuity for life

Function Blocks



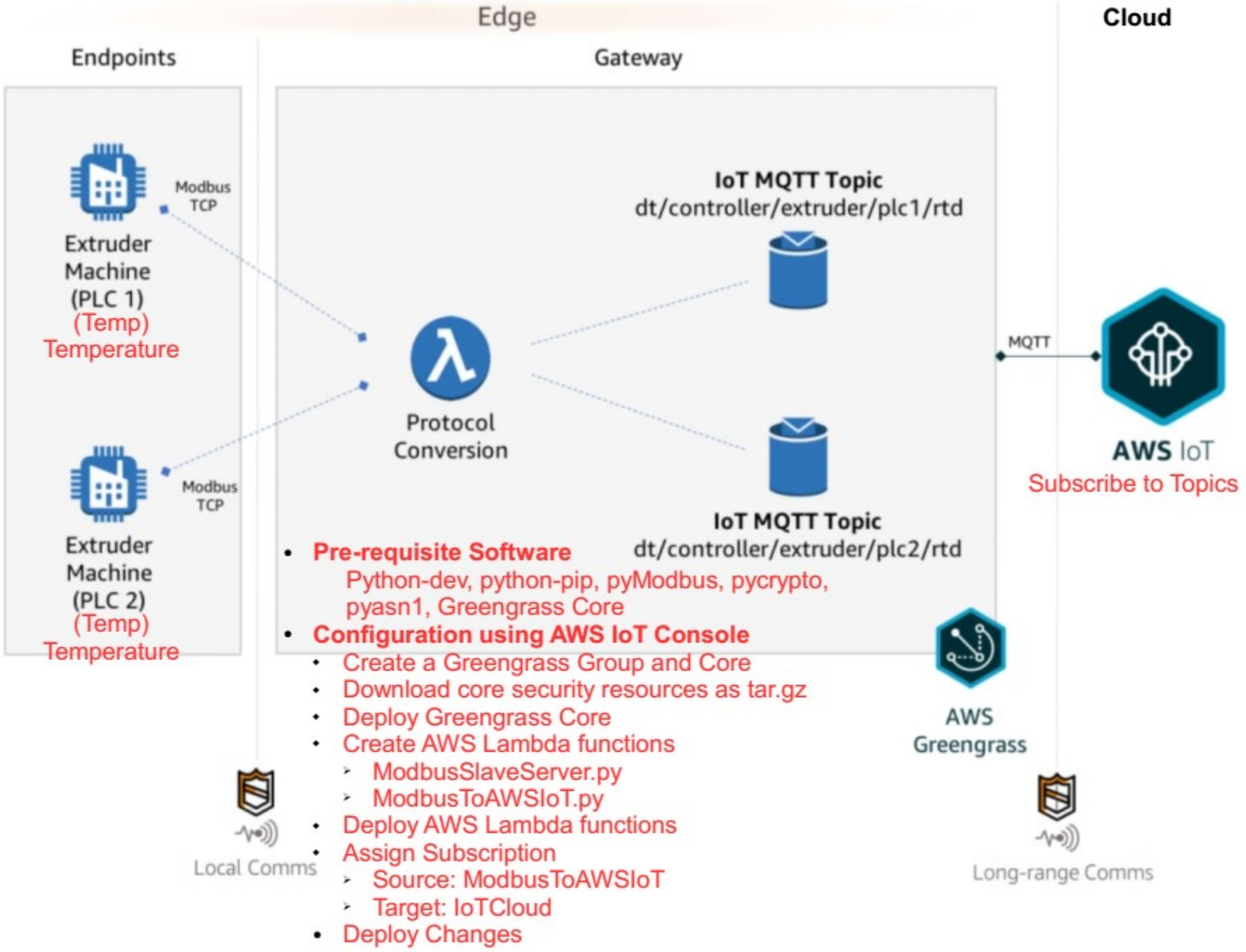


TLS Certificates

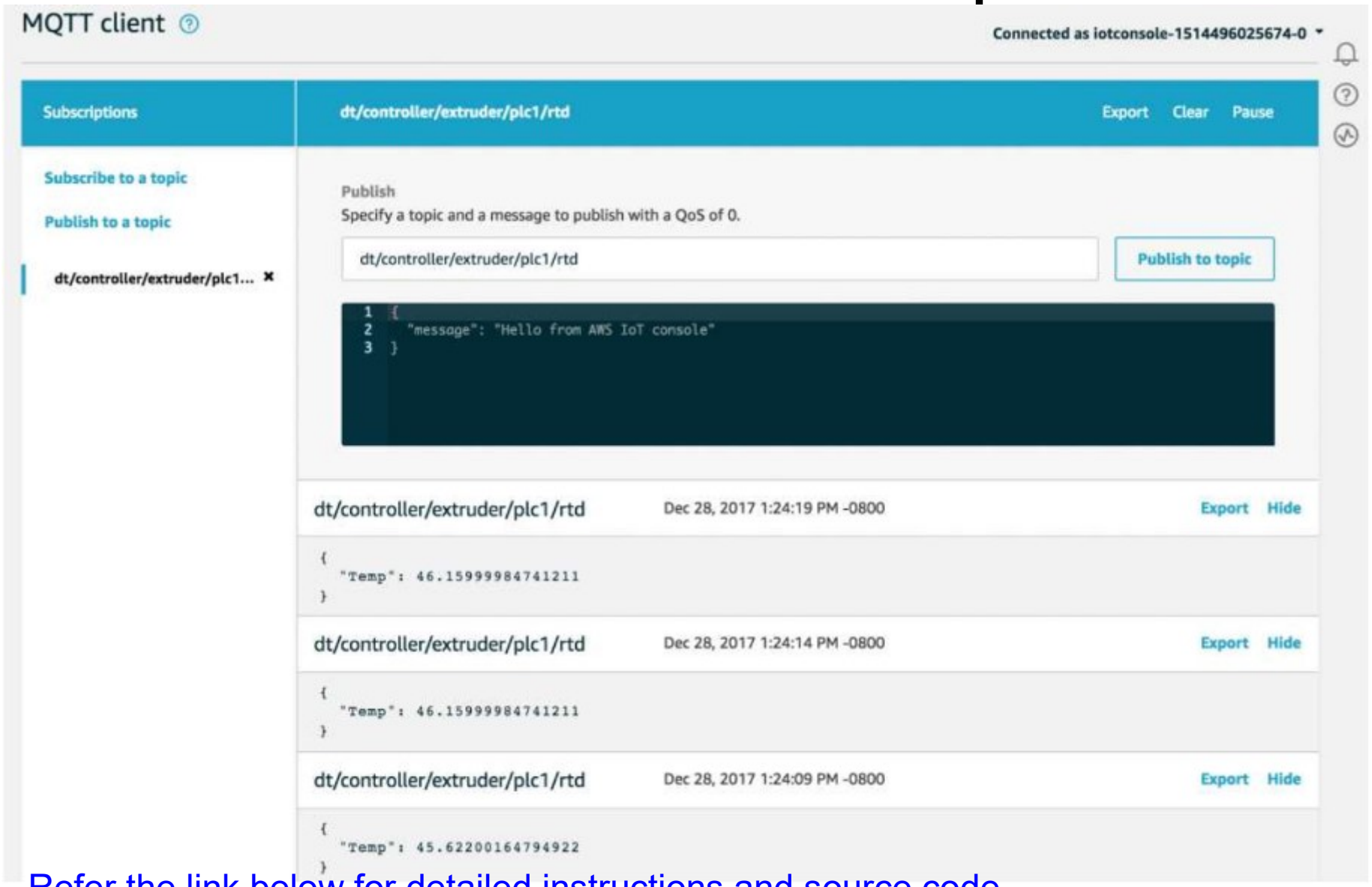


The next slide shows how to manually set-up a Modbus to MQTT protocol converter on a computer acting as an Edge Node to connect an older PLC that supports Modbus Communications to the Cloud

Modbus to MQTT Conversion to support older-PLCs



MQTT Client Subscription



Refer the link below for detailed instructions and source code

https://aws.amazon.com/blogs/iot/perform-protocol-conversion-at-the-edge-with-aws-lambda-and-aws-greengrass/

BRX Do-more!

Cloud-Capable PLCs

JSONPARSE Instruction

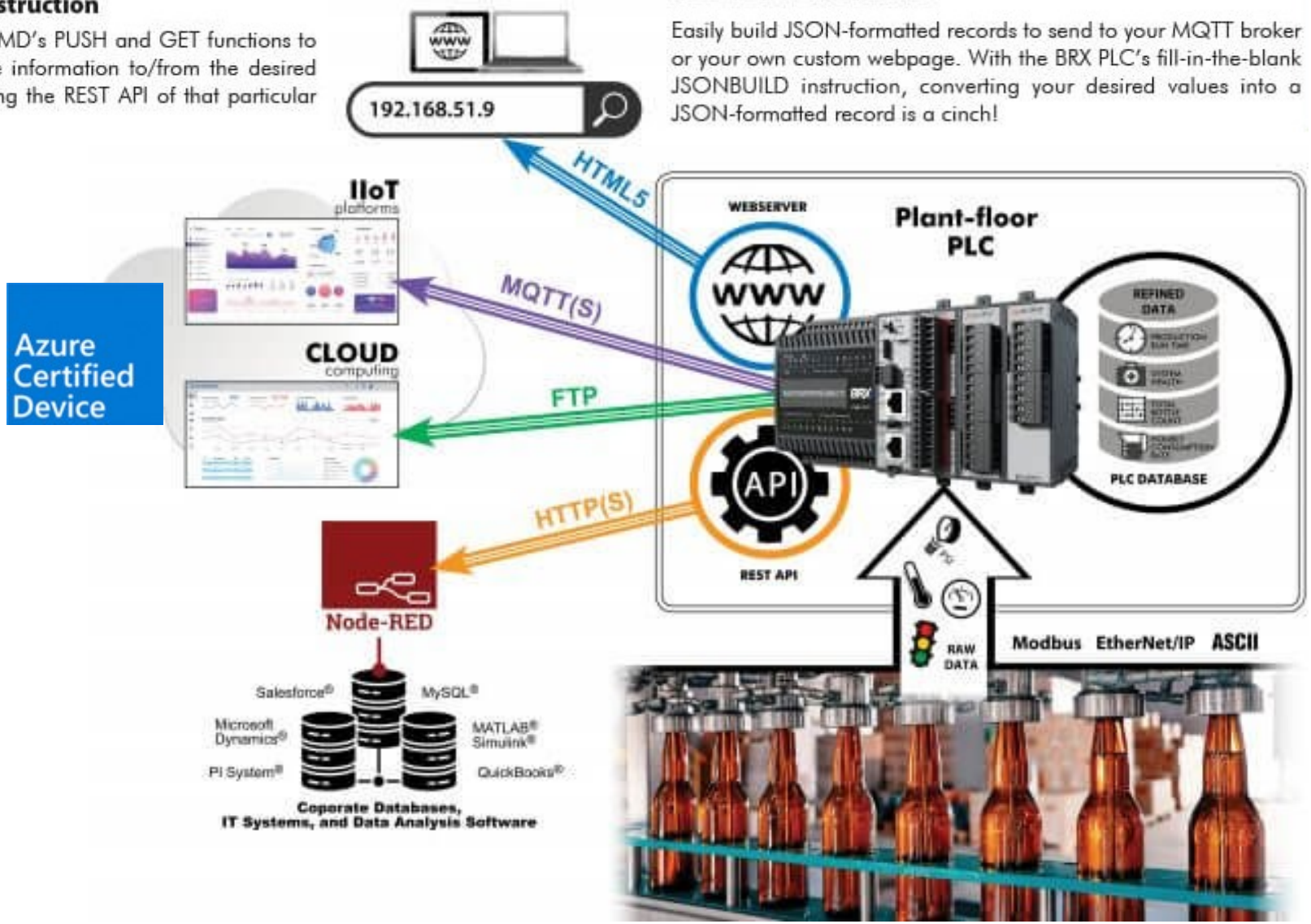
The JSONPARSE instruction allows the BRX PLC to decipher the information within the JSON-formatted responses received from many Web servers. This instruction looks up a value based on a field name or a 0-based array index within a JSON input record. Once found, the value can be returned as text or as numeric or bit values.

HTTPCMD Instruction

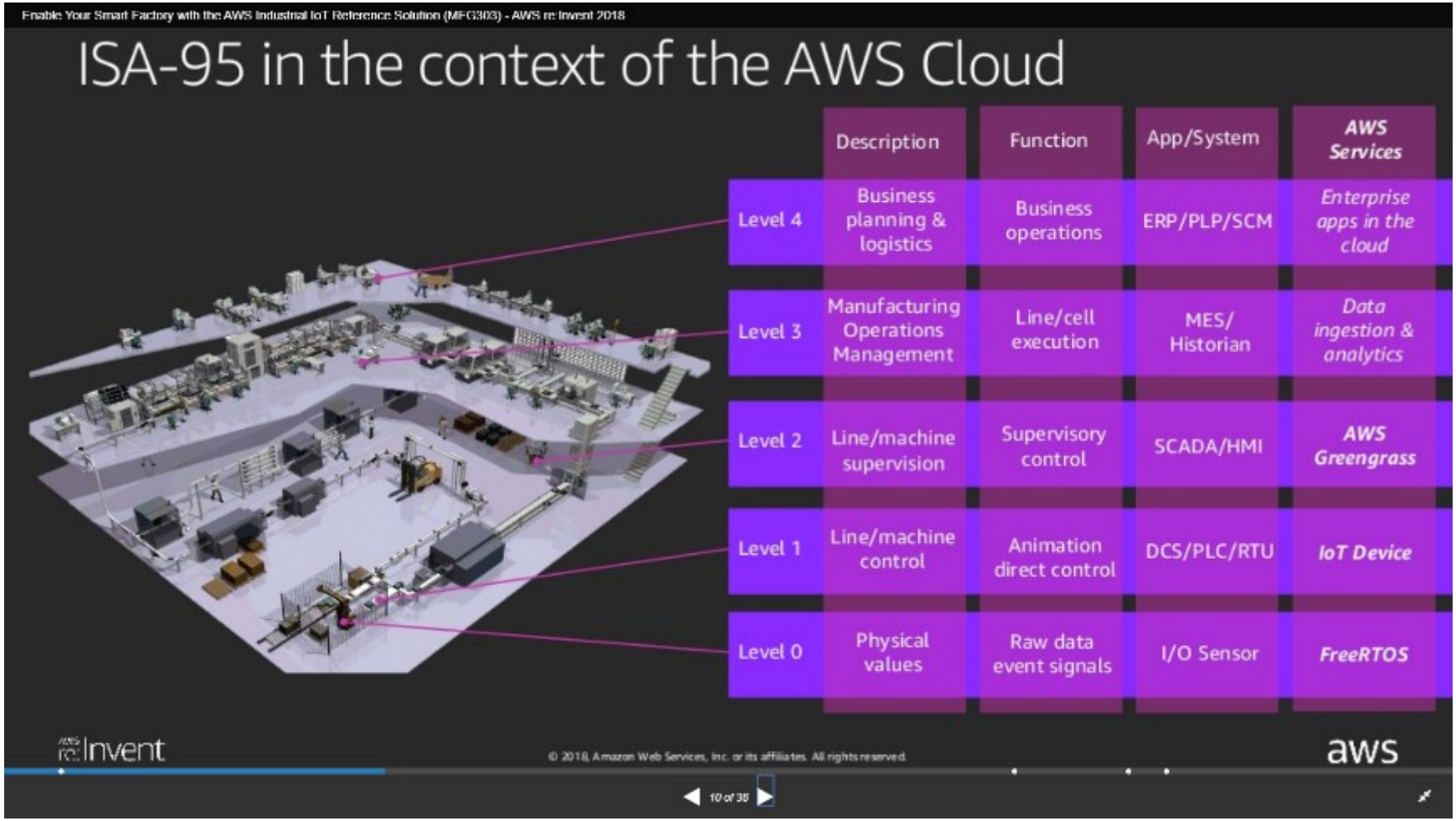
Use the HTTPCMD's PUSH and GET functions to send or receive information to/from the desired Web server using the REST API of that particular site.

JSONBUILD Instruction

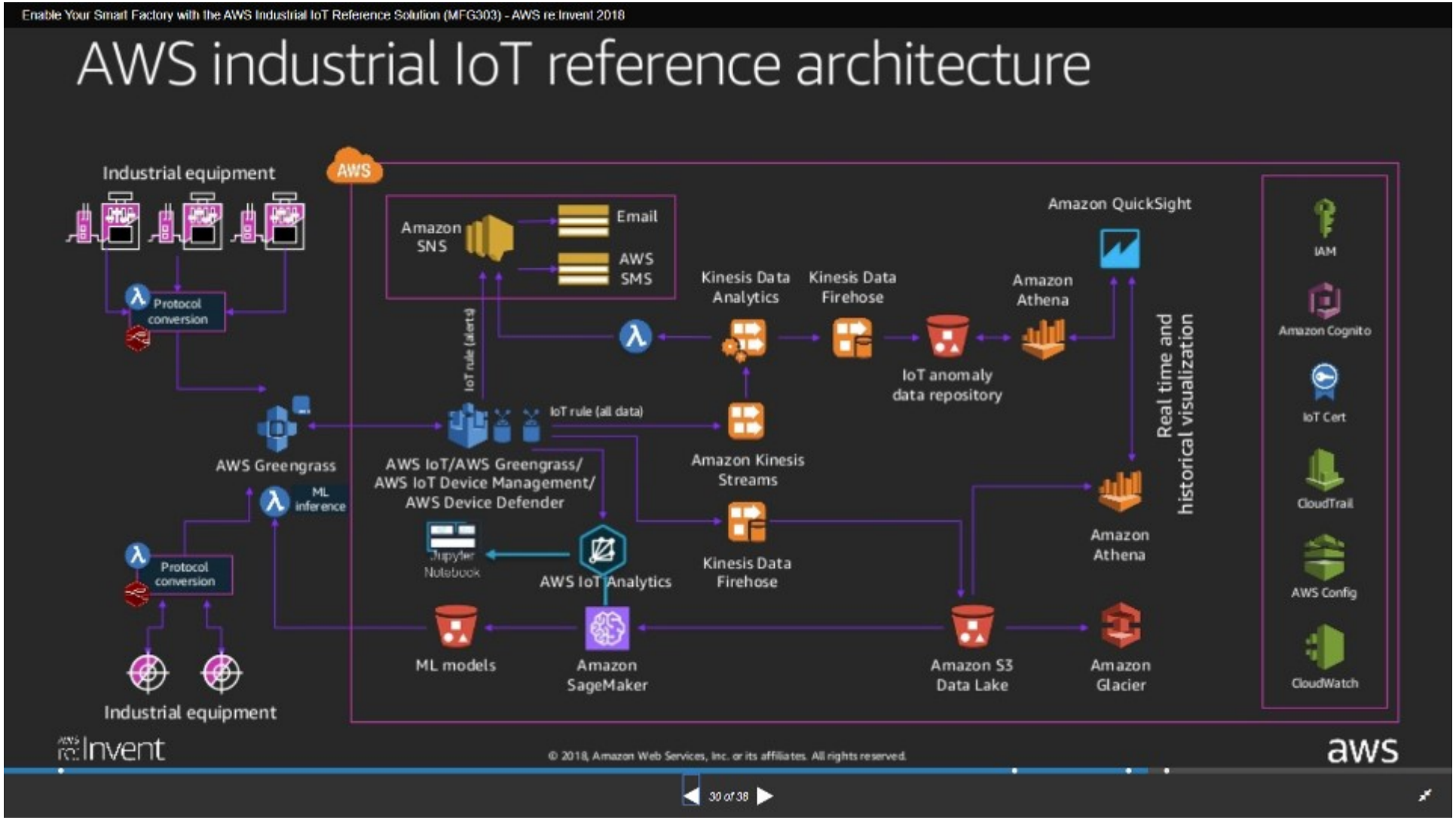
Easily build JSON-formatted records to send to your MQTT broker or your own custom webpage. With the BRX PLC's fill-in-the-blank JSONBUILD instruction, converting your desired values into a JSON-formatted record is a cinch!



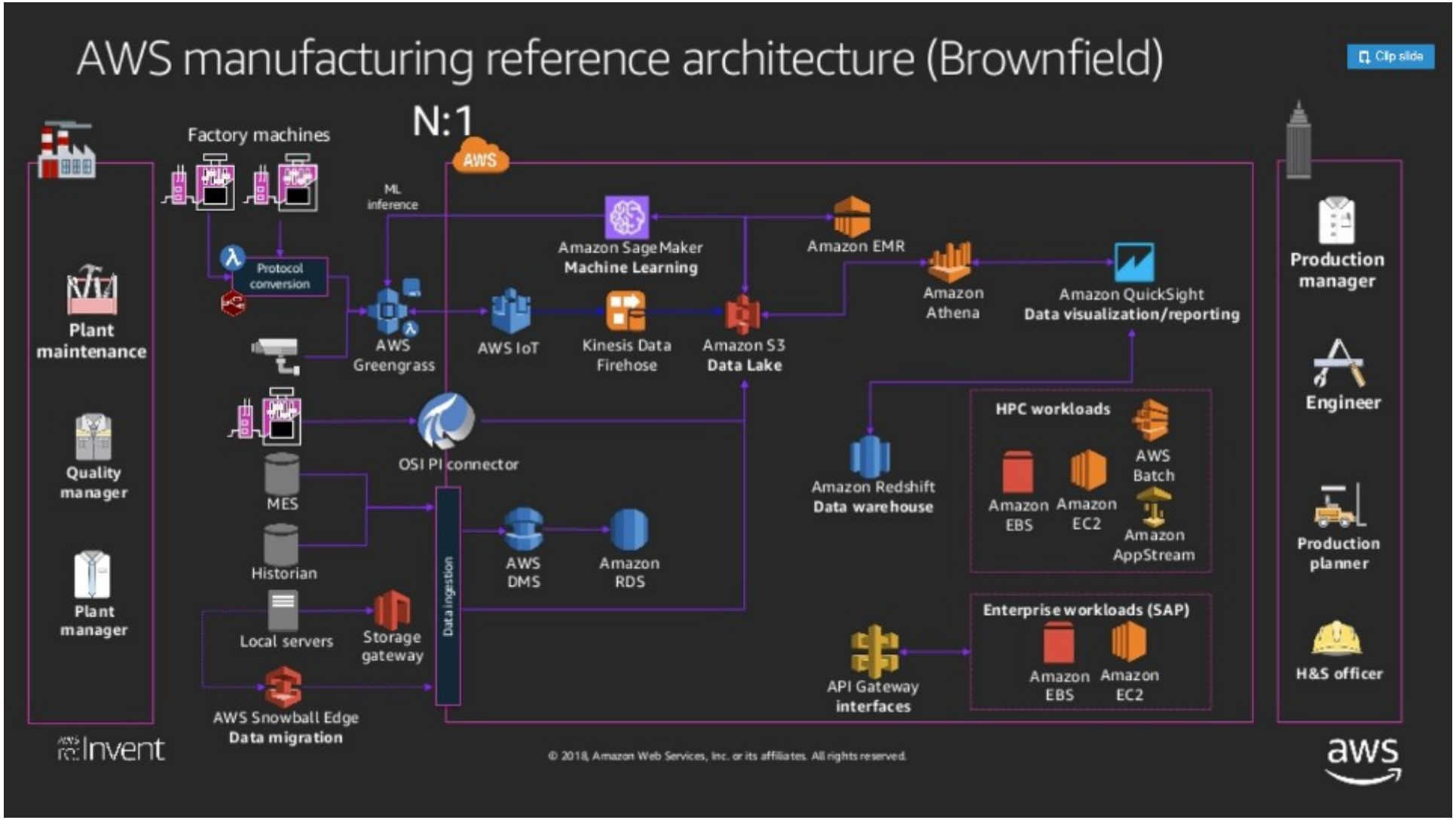
Amazon Industrial IoT Stack



Amazon IoT Architecture -1



Amazon IoT Architecture - 2



Amazon IoT Edge

- *AWS Greengrass is a software that extends AWS Cloud capabilities to local devices, making it possible for those devices to collect and analyze data closer to the source of information, while also securely communicating with each other on local networks.*

- So, Greengrass is a software that will enable you to perform, in a local network, basically the same process as if your Things were connected to a cloud but this time they are connected to a local device.
- This solves the connectivity issue with the cloud servers. If your Greengrass device loses connection to the cloud, it won't stop working. It still gather data from the Things and perform automated actions (that you made before deploying Greengrass, we will see that later on).
- Once the Greengrass device reconnects to the cloud it can perform updates to its core and send you back the data it collected when it couldn't reach the cloud.
- AWS Greengrass provides the core software that you can install on your device, an SDK and an API.

AWS Greengrass Service Features

- A lambda runtime that allows you to execute serverless instructions if needed.
- A shadow implementation, this means that a Thing has a JSON file (the shadow) where all of its parameters/variables are set and can be modified with lambda functions from the core or from the cloud.
- A message manager, for instance, when the core needs to restart, the Things can still send messages and the core will save them until the core has restarted again.
- A group management, a group is comprised of the Things and the Greengrass Core.
- A discovery service, it is a service mainly used by the Things to get certificates to connect to a Greengrass Core.
- An Over-the-air update agent that allows updating one or more Greengrass Cores on a network at the same time or on predefined schedules. The devices with the Greengrass core will need to have the WiFi activated for this feature to work.
- Local resources access on the Greengrass Core device if it is needed, the resources can be anything.
- A machine learning inference, meaning that the training is done on the cloud servers but the model (the “brain” of the AI) is on the Greengrass device and can perform in real time what it was trained to do.
- Since version 1.7, Connectors help you implement reusable business logic, interact with cloud and local services (including AWS and third-party services), ingest and process device data, enable device-to-device calls using MQTT topic subscriptions and user-defined Lambda functions. This module works as packages that you can deploy on your core without the hassle of learning new APIs or protocols. It’s meant to make your life easier.
- Refer : https://medium.com/smileinnovation/aws-greengrass-the-forefront-of-edge-computing-8ec2098a33b7

IoT vs IIoT Suites from Leading Automation Vendors

- An IoT platform will give you the middleware
- But you'll still have to..
  - physically connect up your machines and sensors
  - install and configure a gateway device
  - write the code to pull and capture the data streams
  - write the code that manages the data and contextualizes it
  - write the algorithms that turn that data into useful KPIs
  - write the code that displays within a user interface
  - develop code and a UI to capture what the operators are doing
- IIoT suites like Siemens Mindsphere, Beckhoff TwinCAT IoT,.. provides pre-packaged Industrial Connectivity, UI and Programming Applications to enable remote Configuration, Monitoring, Control and Optimization of Plant Operations that can be deployed on the AWS, Azure or Google Cloud

SIEMENS
Ingenuity for life

(I)IoT Players

You will typically use the (I)IoT cloud of your PLC/Automation System manufacturer

If you want to set-up your own IoT cloud you will first encounter the BIG THREE

- https://aws.amazon.com/ aws
- https://azure.microsoft.com/ Azure
- https://cloud.google.com/ Google Cloud

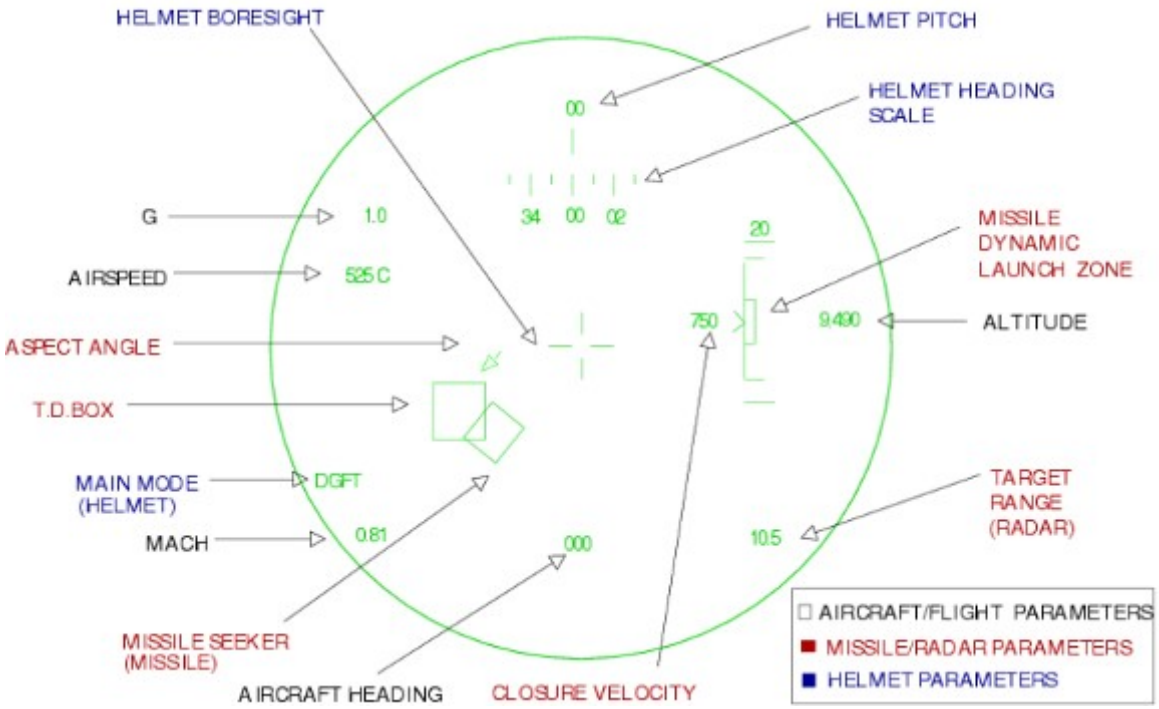
Of all the (I)IoT platforms out there the most economical and easy to use is https://www.kaaiot.com/ KAA

Assisted / Augmented Reality

The Next Generation of HMI - Assisted Reality / Augmented Reality (AR)

Augmented Reality
- Looks can Kill

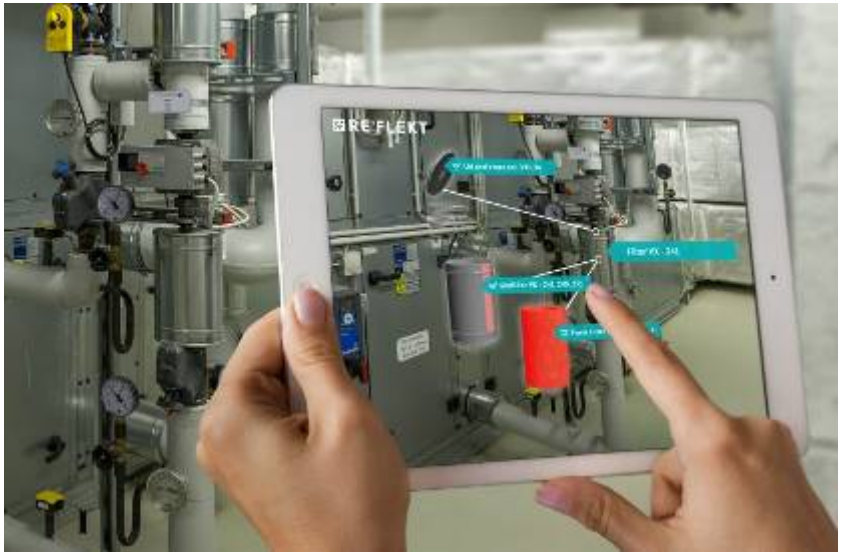
Since the 1980s, with the Mig-29's Helmet Mounted Display, Sights & Cueing system, the pilot is able to target enemy aircraft simply by looking at them within a specific range of view



Augmented Reality (AR)

- Assisted reality refers to projecting additional information into a user's field of vision, importantly hands-free. It differs from augmented reality in that it doesn't change what the user is seeing, only adds an extra layer of information into their peripheral vision.
- An engineer using augmented reality equipment, sees a virtual display on real environment along with instructions to follow while manufacturing/maintaining a product. Thereby improving engineer efficiency(with approx. 99% accuracy and 30% of less time consumption).

Applications of AR



- Maintenance
- Monitoring & Control
- Assembly / Disassembly

AR Types

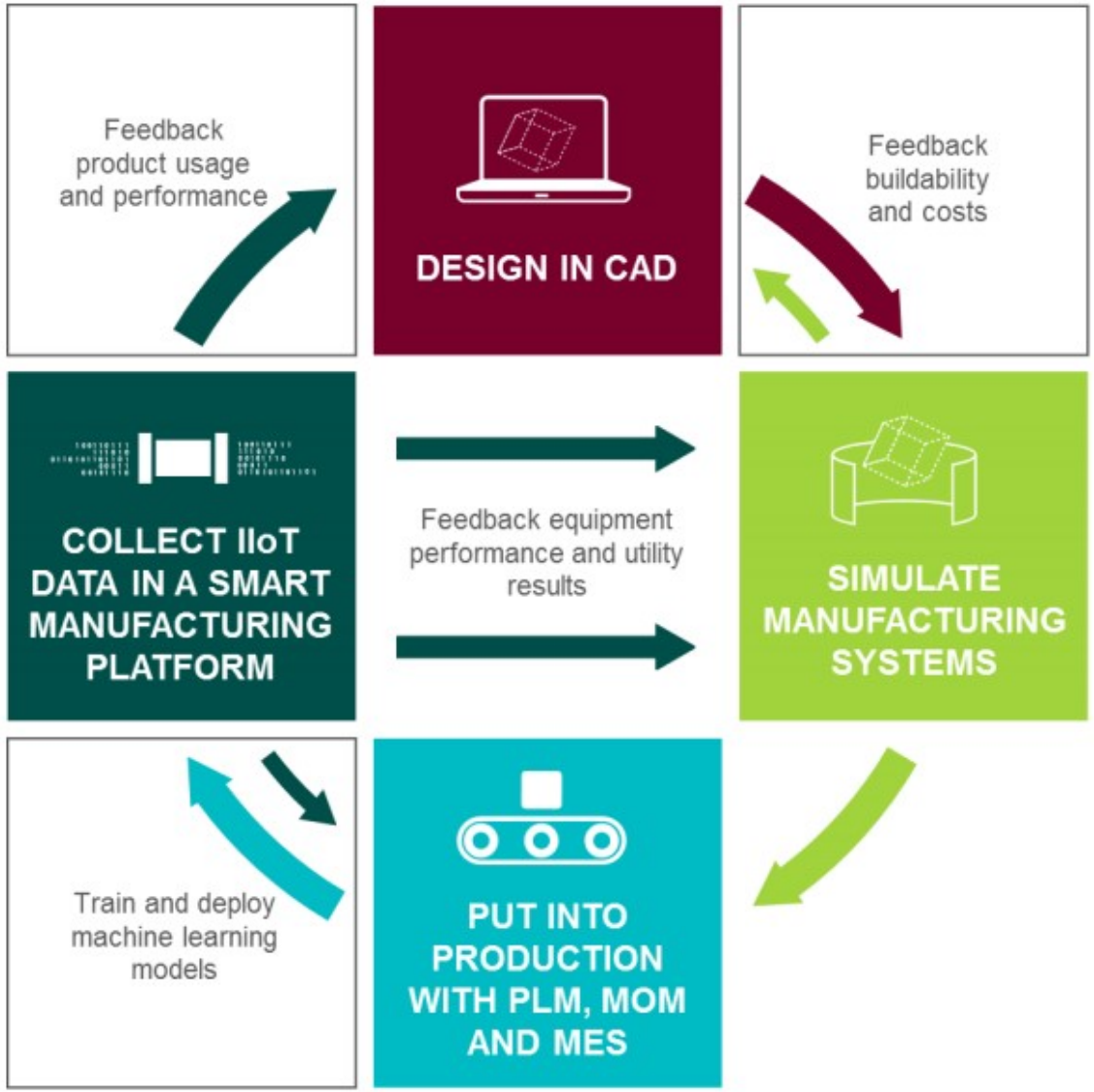
- Marker-based AR
  - uses a visual marker, usually in the form of a 2D QR code
  - Bluetooth Low Energy (BLE) Beacon
- Marker-less AR - uses positional information collected from the device's camera (Image Recognition), GPS, digital compass, and accelerometer. The inputs from these data points allow the system to understand the 3D environment through a process known as ***Simultaneous Localization and Mapping (SLAM).***

AR Hardware

- Smart Phones and Tablets
- AR Headsets
  - Google AR Glasses
  - Microsoft HoloLens

Simulation
&
Virtual Reality

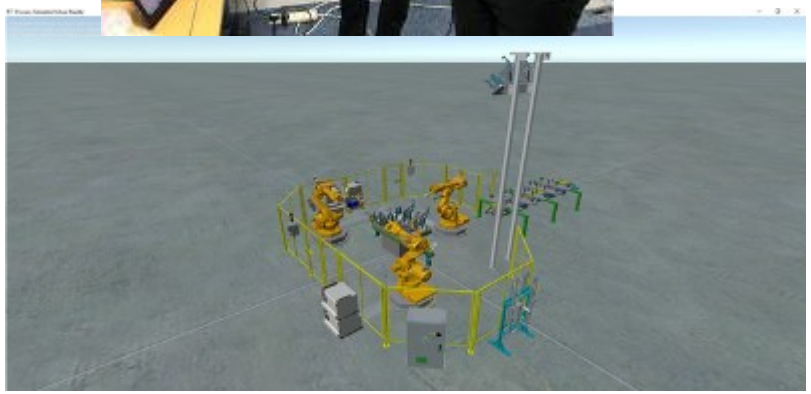
Plant Simulation & Virtual Reality



- Verify Designs
- Investigate Problems
- Check Optimizations
- Analyze Disaster Scenarios
  - Plan & Train for Avoidance & Recovery
  - Refinery Shut-down & Restart

Plant Simulation – Discrete Manufacturing

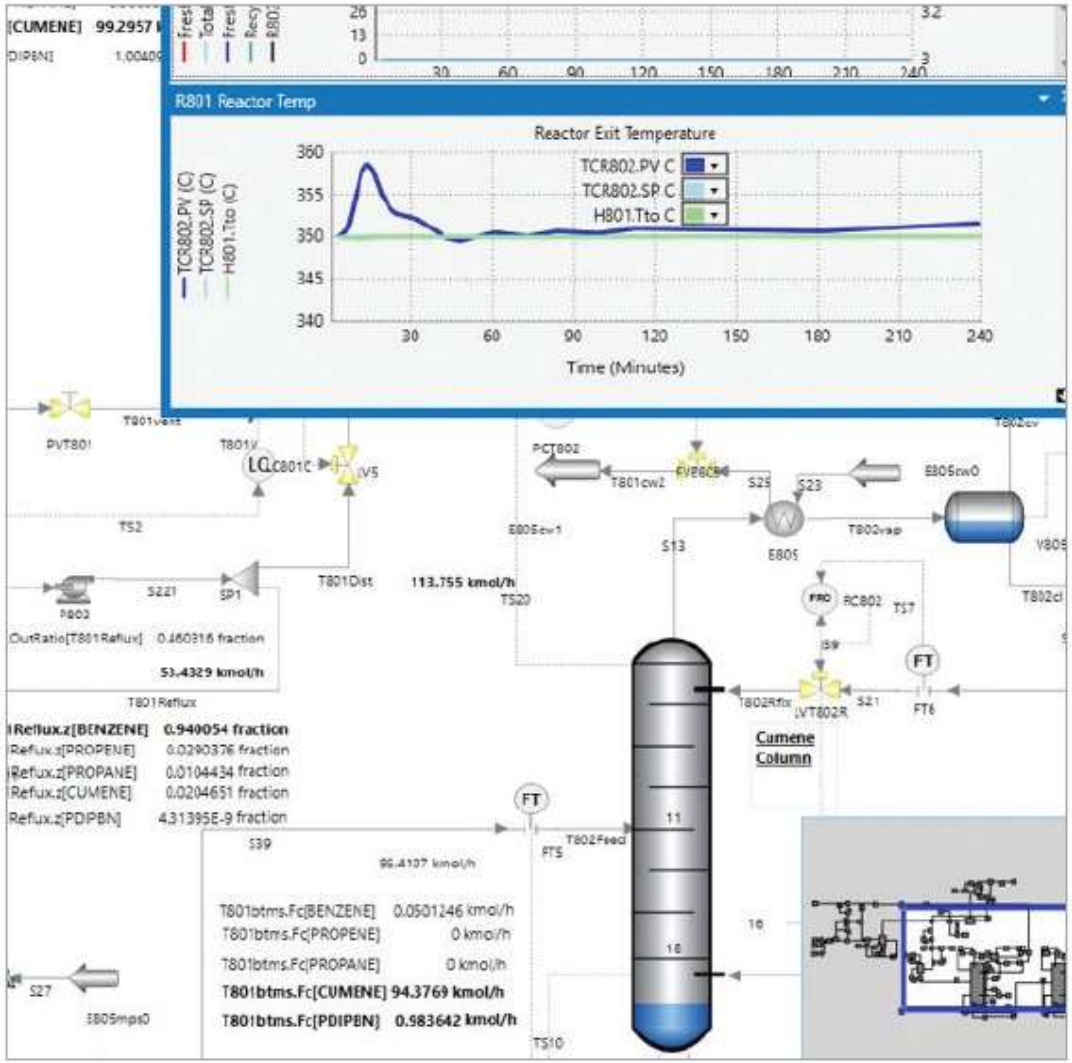
SIMSOL SIEMENS anylogic Arena® Simulation Software Simio Forward Thinking



- Process definition
- Layout definition and integration
- Robotic programming
- Mechanical sequencing
- Material flow
- Automation control
- Virtual commissioning
- Man-machine interaction
- Safety Systems
- Virtual reality for presentation and collaboration
- Validating PLC code offsite with virtual commissioning

Virtual commissioning is the simulation and debug of PLC code using a 'digital twin'
this is an exact functional replica of the production line in the virtual world, where unlimited testing can take place in an environment with no risk. The virtual commissioning approach facilitates interdisciplinary collaboration between mechanical and electrical teams much earlier on in a project, allowing more of the programming to be completed and tested before physical installation. This leads to less time spent on the factory floor

Plant Simulation – Continuous Process



AVEVA™ Process Simulation

aspentech

- Process library with distillation columns, reactors, heat exchangers, compressors and other common unit operations
- Fluid thermodynamics methods such as SRK, SRKM, PR, PRM, NRTL, UNIQUAC, UNIFAC, Wilson, Hayden O'Connell, IF97..
- Integrated dynamic simulation for better distillation column relief load calculations
- Steam library with extraction turbines, desuperheaters and condensers
- Cooling water library with supply, return, pipes, pumps and exchangers
- Transient Flow library for water hammer and pressure surge analysis
- Fluid thermodynamics methods such as steam (IF97), cooling water, other heat transfer mediums
- Flare library with relief valves, tail pipes and flare stacks

Operator Training System and DCS Logic integrated in the Simulation

SYSTEM REQUIREMENTS

- Windows Mixed Reality Gaming laptop with Windows 10 or Xbox One
- Windows Mixed Reality Immersive Headset and Motion Controllers

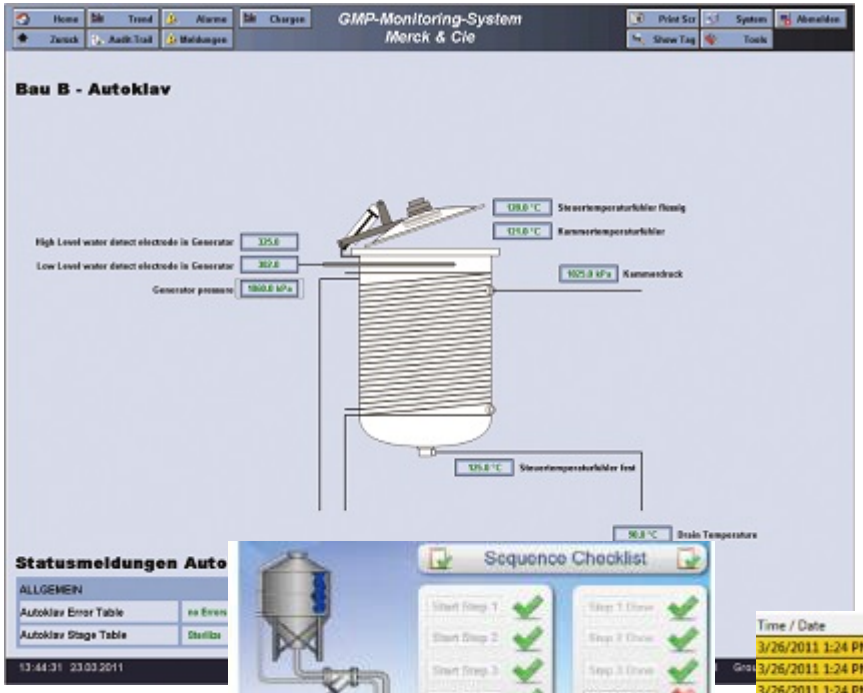
Pharmaceutical Standards

Pharmaceutical Standards

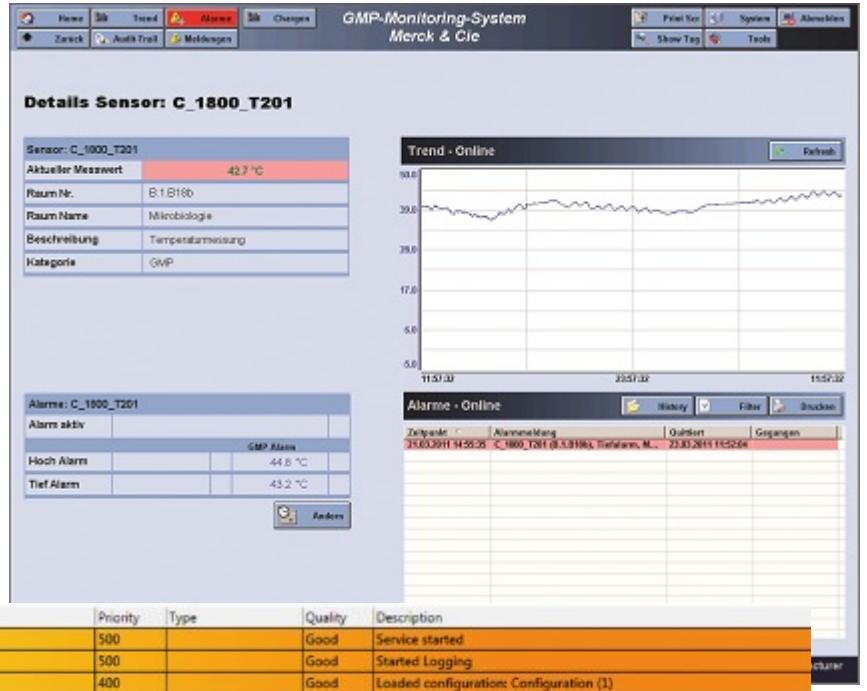
- Driven by the need to digitalize record keeping
- CGMP, GAMPx, EU Annex 11, FDA 21 CFR Part 11,…
- These standards and regulations make quality testing an integral part of each stage of manufacturing, including facilities, equipment, materials acquisition and staff hygiene. This rigorous employment of standard of operating procedures (SOP) helps ensure purity of manufactured pharmaceuticals.
- Under these guidelines, not only must process data be recorded, but also environmental conditions of rooms, storage and production facilities. In addition, security is mandated to keep records of user access to equipment and materials, alarms, change-logs, etc... This requires traceability features such as individual user access with protected user names and passwords, and the verification of eSignatures.
- **Benefits of the regulations**
  - Protection and retrieval of electronic records
  - Operational consistency
  - Improve productivity and efficiency through automation
  - Minimize or eliminate management of paper documentation
  - Enable faster data-related searches
  - Enable trending
  - Electronic submissions to the FDA

Meeting Pharma Standards with modern SCADA & Historian

- SCADA systems capabilities for reports, traceability, and environmental control procedures allow for these pharmaceutical regulations to be met with a greater degree of accuracy and quality control.
- Clear limits for values and carefully planned guidelines give operators the precise information needed for quality assurance. Operator interface screens use standard colors and graphics to make analysis easy, from historical data and trends to everyday operations.



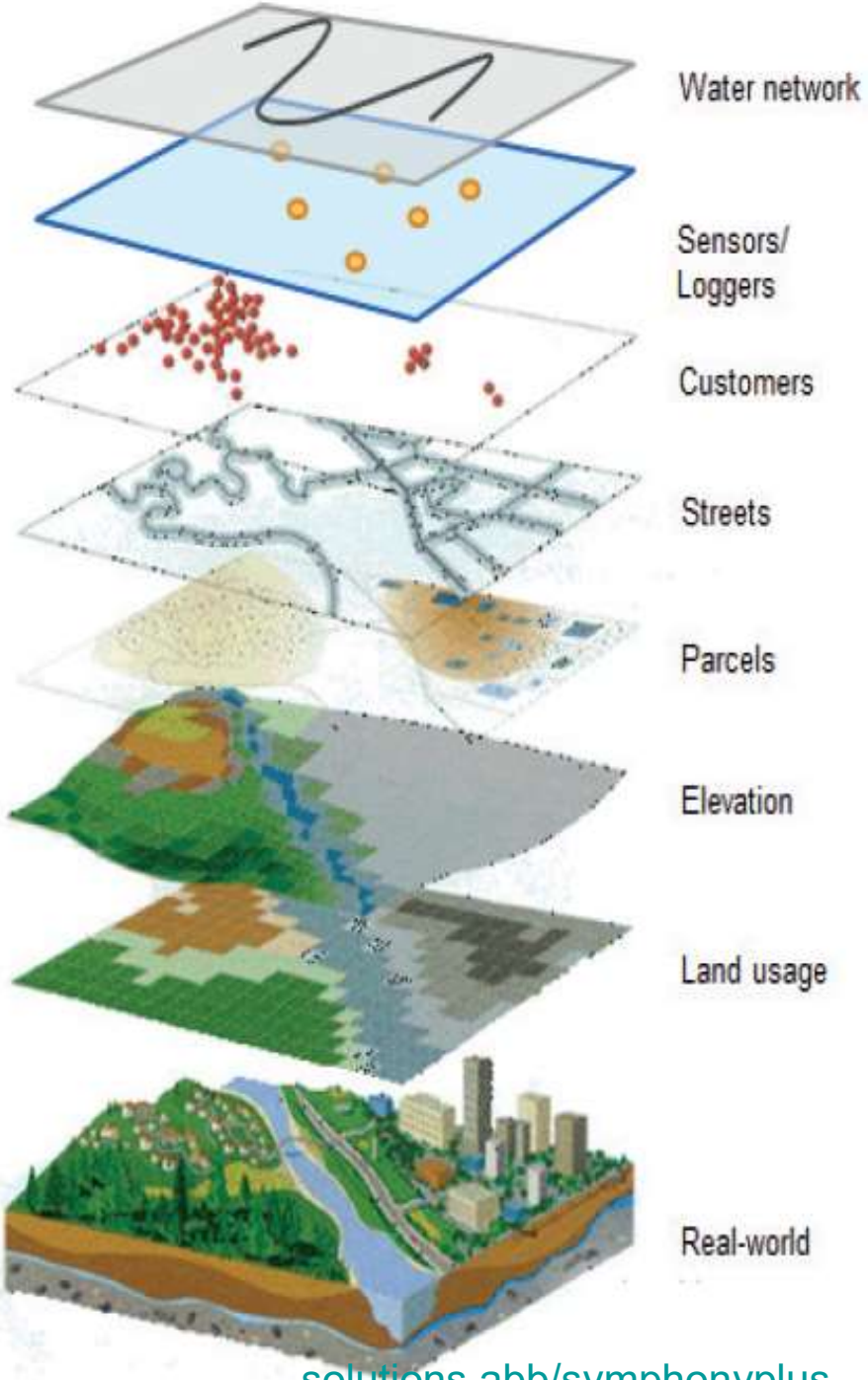
Operation check function



Time / Date	Tag	Priority	Type	Quality	Description
3/26/2011 1:24 PM		500		Good	Service started
3/26/2011 1:24 PM	TwxLog64	500		Good	Started Logging
3/26/2011 1:24 PM	TwxLog64	400		Good	Loaded configuration: Configuration (1)
3/26/2011 1:24 PM		500		Good	Service stopped
3/26/2011 1:23 PM	TwxLog64	500		Good	Stopped Logging
3/26/2011 1:23 PM	FrameWorX Server	600		Good	User "admin" logged in
3/26/2011 1:22 PM	AWXLog32	500		Good	License Enabled
3/26/2011 1:22 PM	AWXLog32	500		Good	ICONICS AlarmWorX64 Logger Started.
3/26/2011 1:22 PM	AlarmWorX64	500		Good	AlarmWorx64 Viewer is correctly connected to the subscription @M:Com

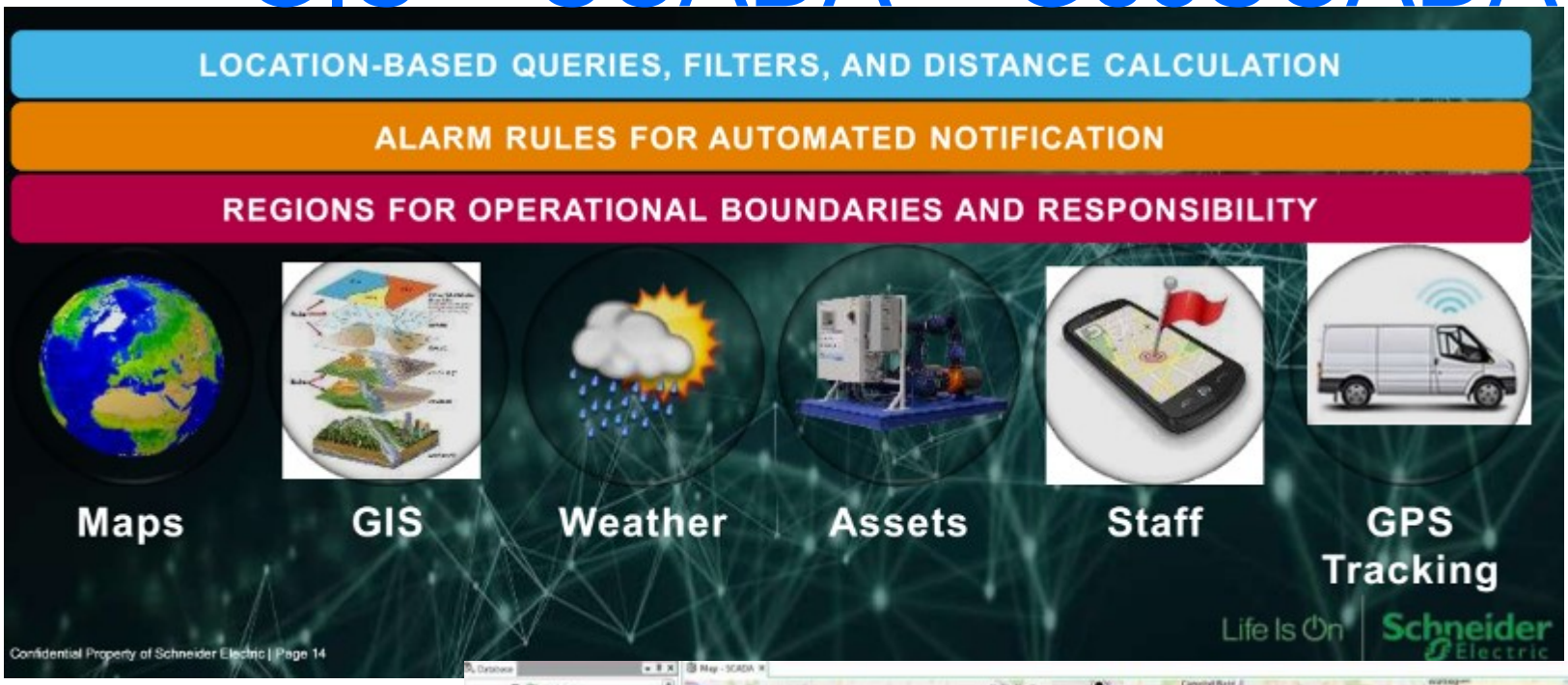
Audit trail function

GeoSCADA

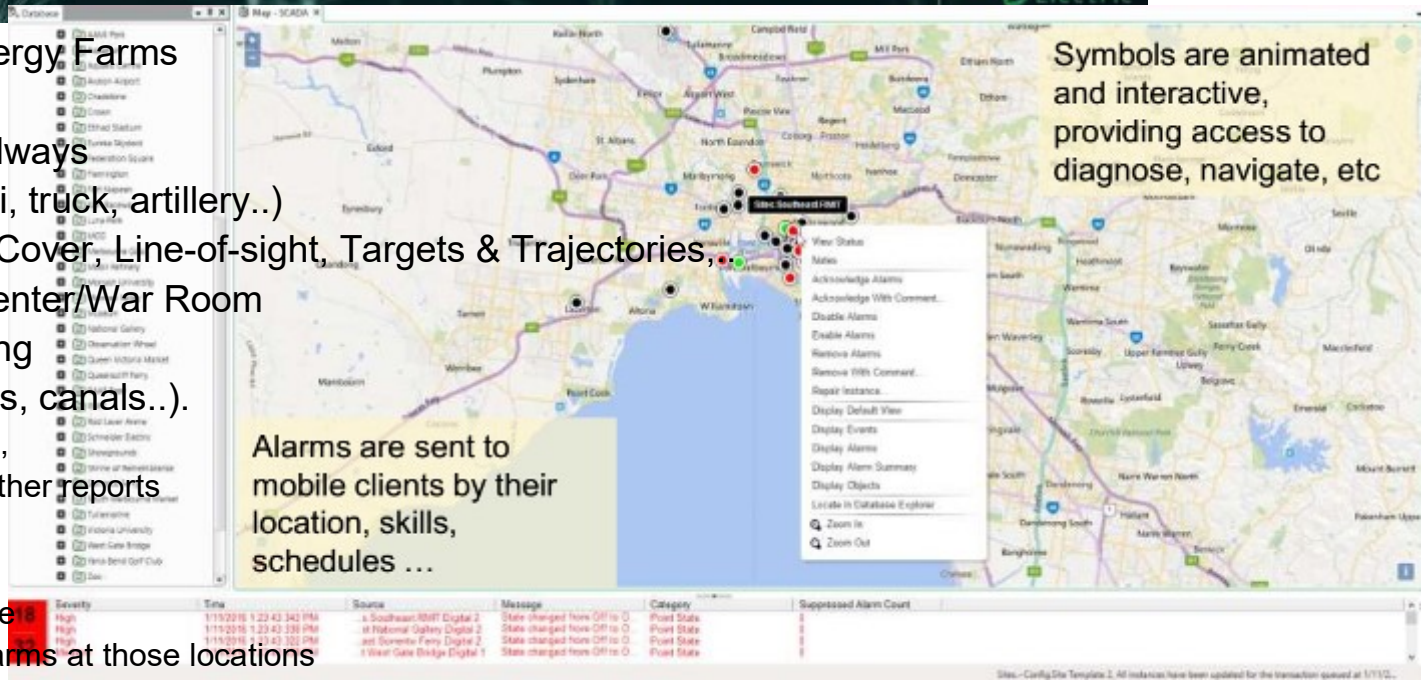


solutions.abb/symphonyplus

GIS + SCADA = GeoSCADA

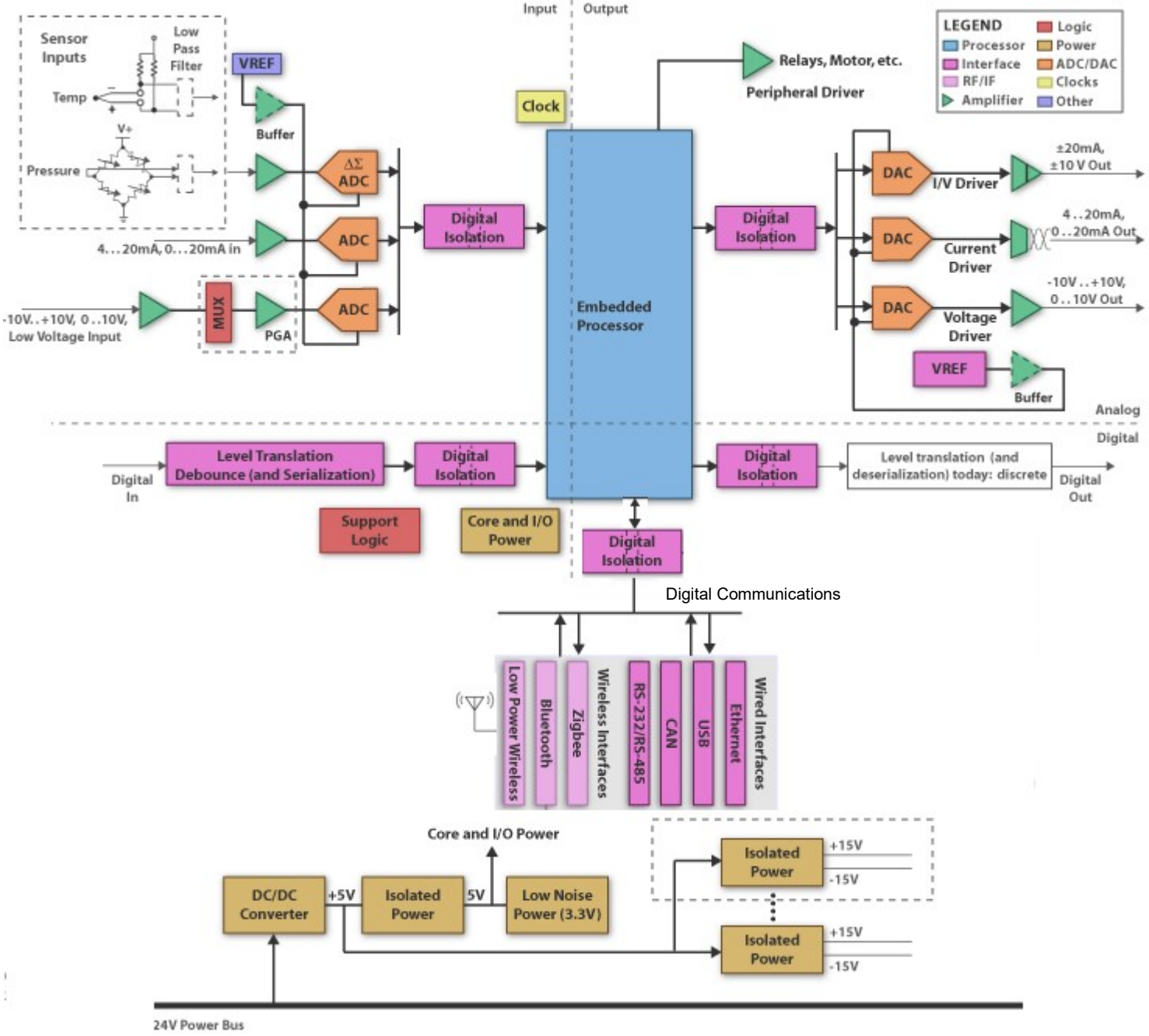


- Electric Grids, Wind Energy Farms
- Oil and Gas Pipelines
- Transport Systems, Railways
- Fleet management (taxi, truck, artillery..)
- Logistics, Terrain, Soil, Cover, Line-of-sight, Targets & Trajectories, ..
- Battlefield Command Center/War Room
- Perimeter/Border Fencing
- Irrigation Systems (dams, canals..).
  - Using upstream level data, online location based weather reports land contours / elevations to forecast down-stream flood locations and activate levee/flood-gates/flood-alarms at those locations

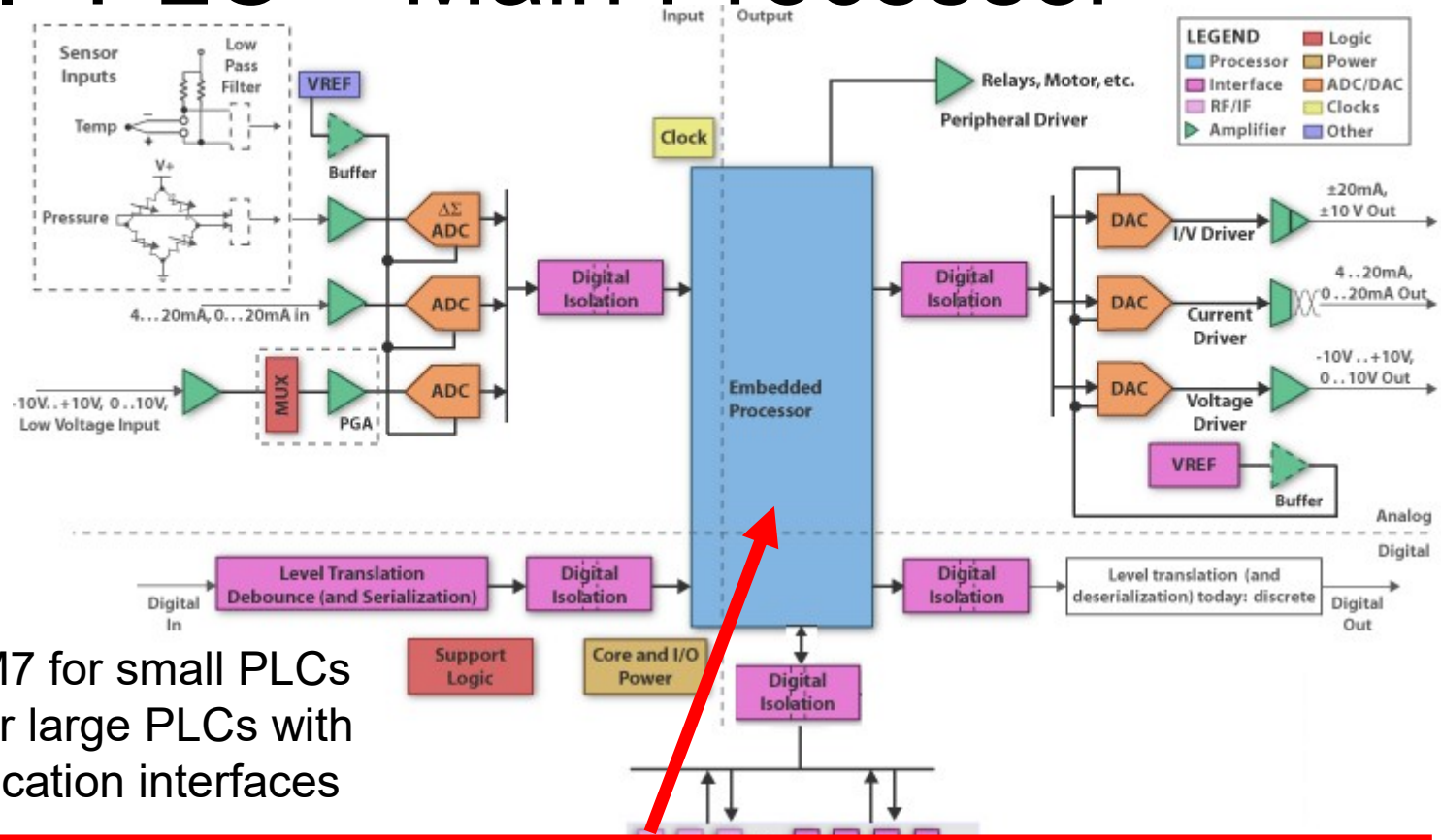


PLC Hardware Design

TI- PLC Internal Architecture



TI- PLC Main Processor

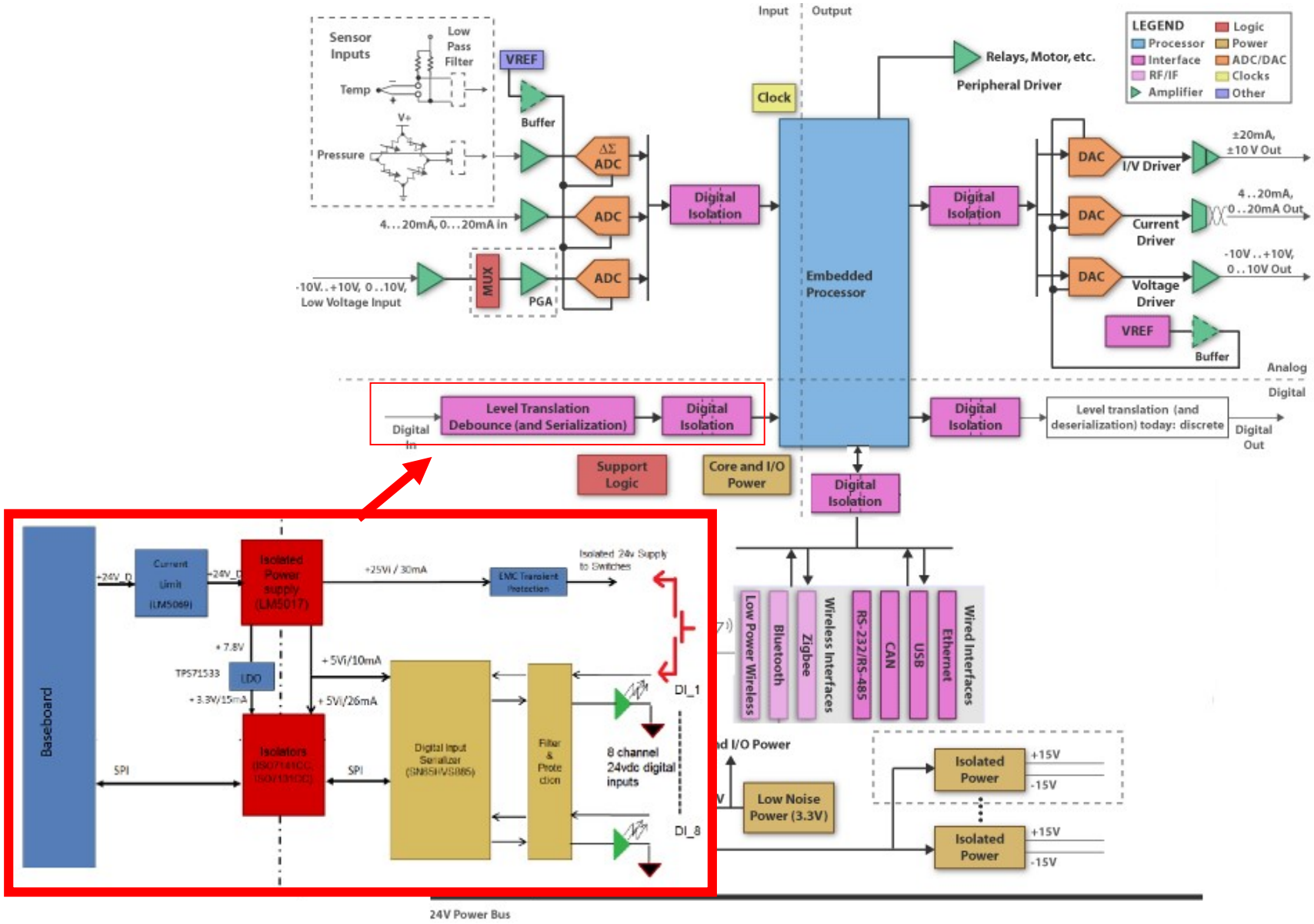


- Arm Cortex-M4/M7 for small PLCs
- Arm Cortex-Ax for large PLCs with multiple communication interfaces

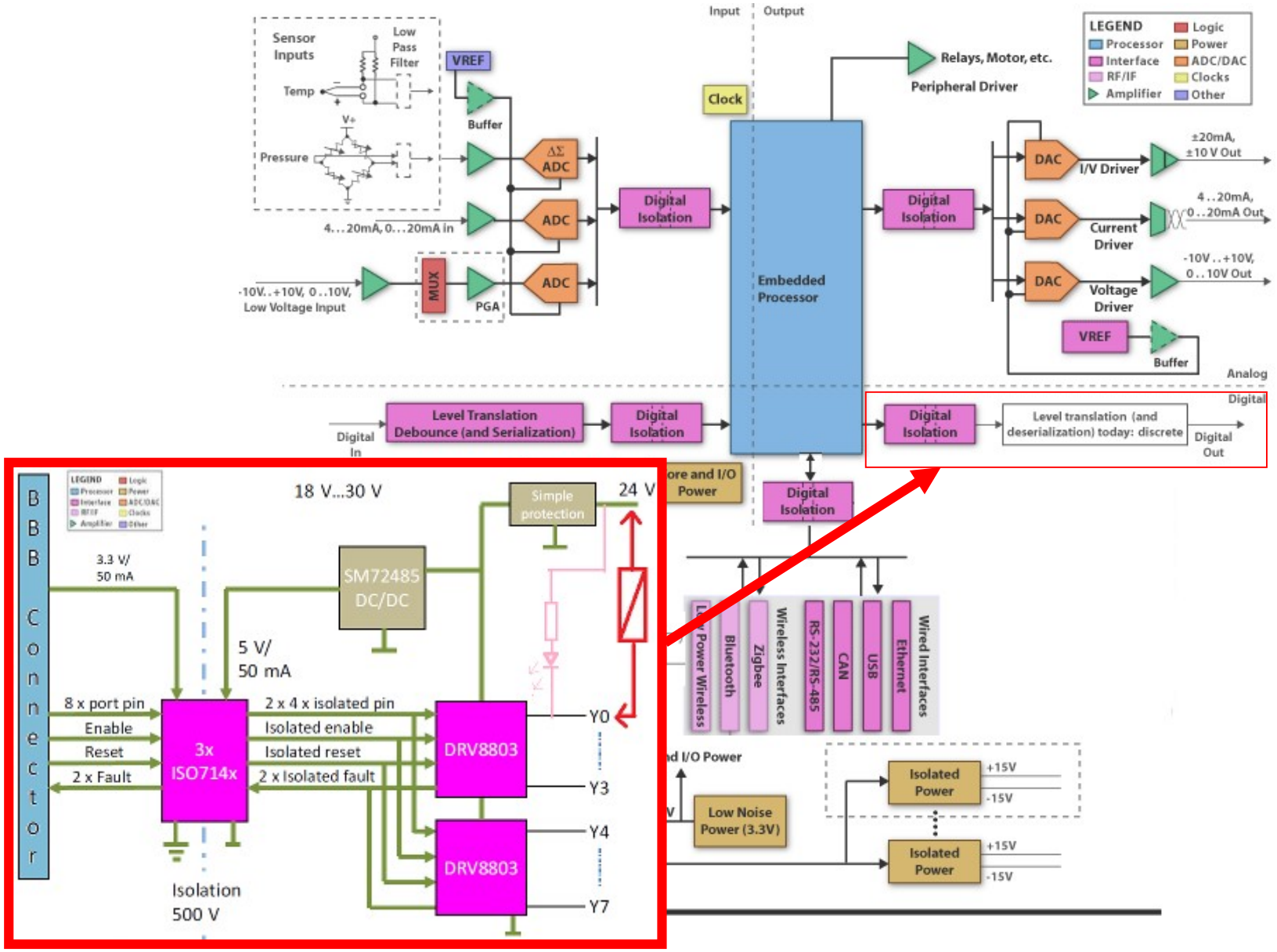
Microprocessors

Part Number	Description	Core	Communication Interface	Analog I/O	End Equipment
Sitara™ AM335x	Sitara™ processors with ARM® Cortex™-A8 core and integrated industrial communications support	ARM® Cortex™ - A8	USB, UART, I²C, SPI, Industrial Ethernet	ADC	Communication module PLC controller (CPU)
Sitara AM4xxx	Sitara processors with ARM Cortex-A9 core with integrated industrial communications support	ARM Cortex - A9	USB, UART, I²C, SPI, Industrial Ethernet	12-bit SAR ADC0 12-bit SAR ADC1	Motor control PLC controller (CPU)
Sitara AM5xxx	Sitara processors with ARM Cortex-A15 core with integrated industrial communications support and DSP acceleration for analytics processing	ARM Cortex - A15	USB, UART, I²C, SPI, Industrial Ethernet	PMIC	PLC controller (CPU)
66AK2L06	High-performance DSP+ARM SoC with integrated digital front end (DFE) and JESD204B	ARM Cortex-A15 and C66x DSP	UART, I²C, SPI, Ethernet, USB, JESD204B	DFE	Communication modules PLC controller (CPU)

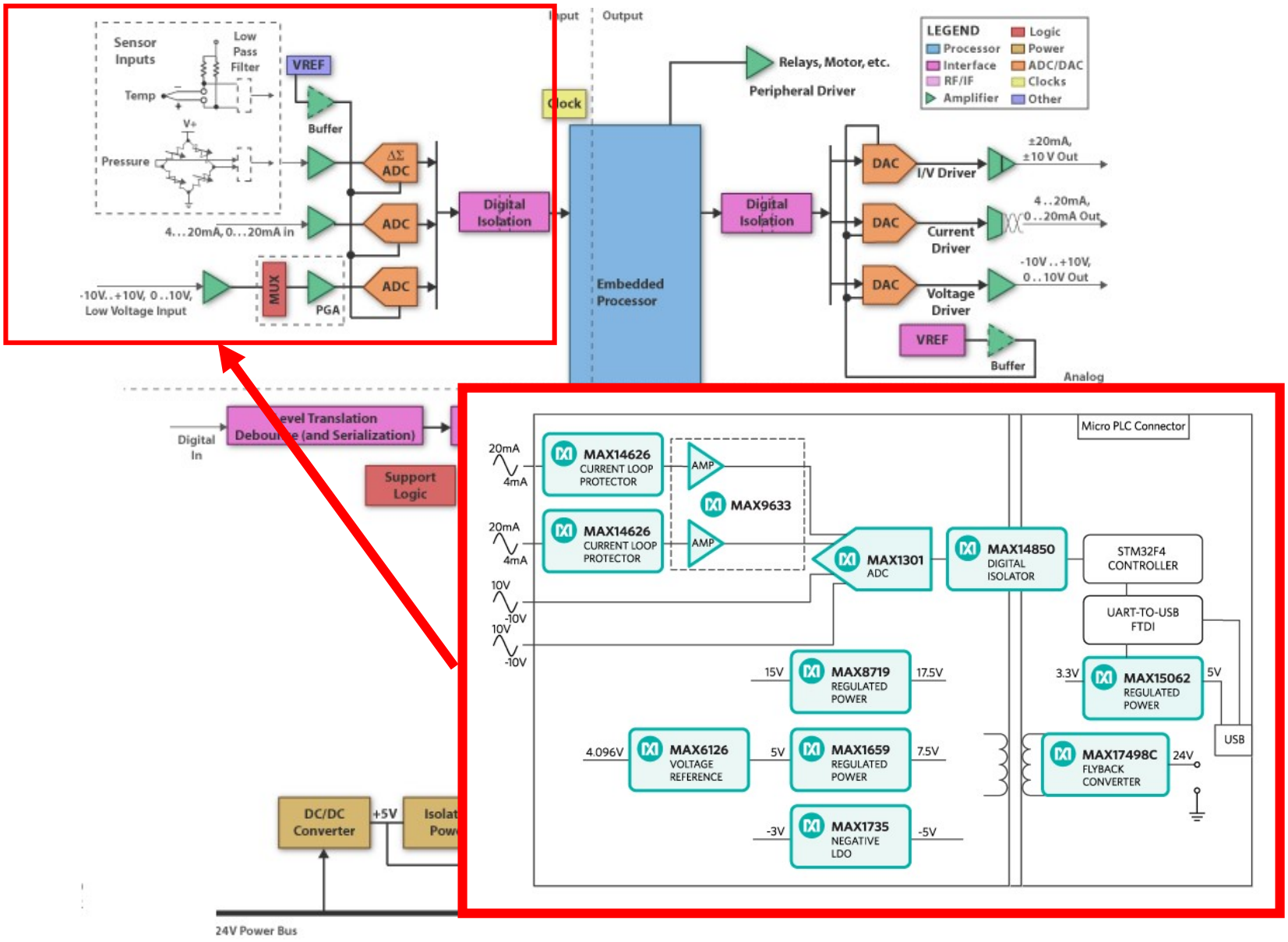
TI-PLC DI Section



TI- PLC DO Section



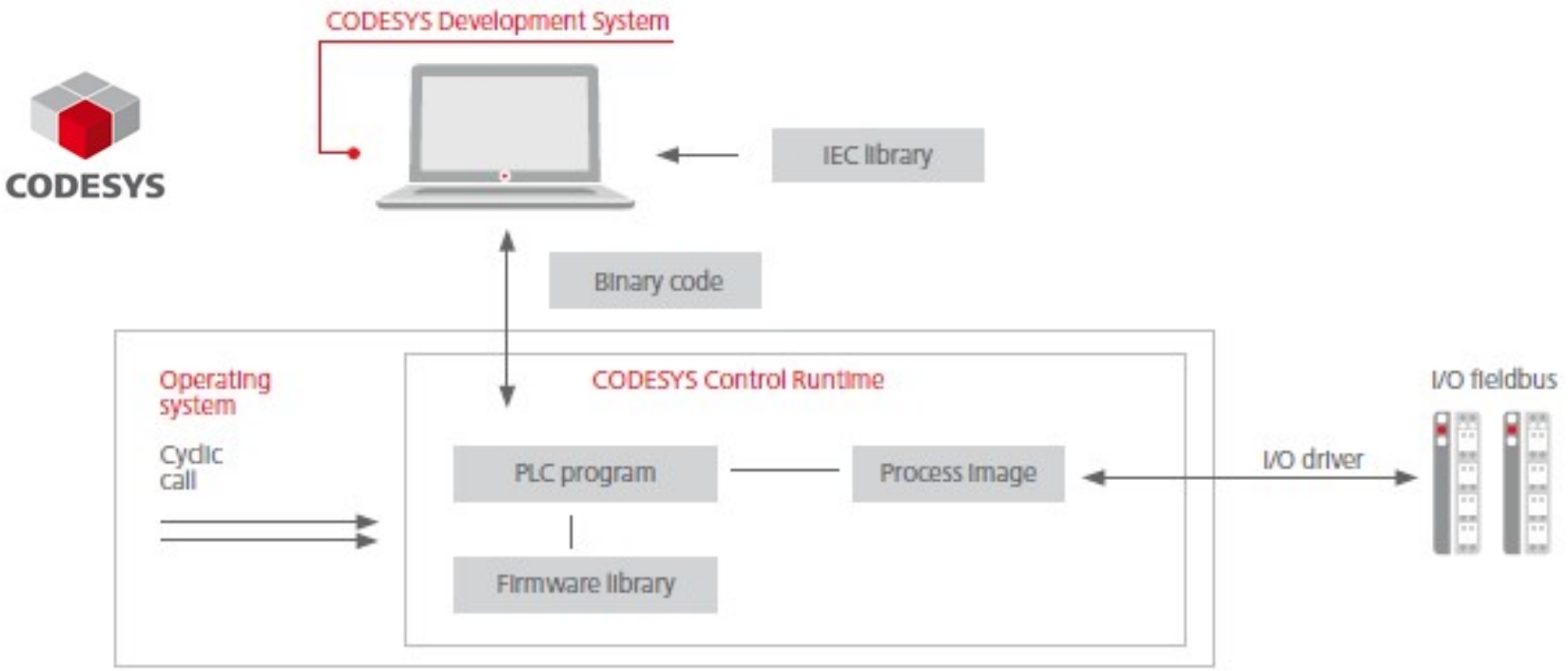
TI PLC AI Section



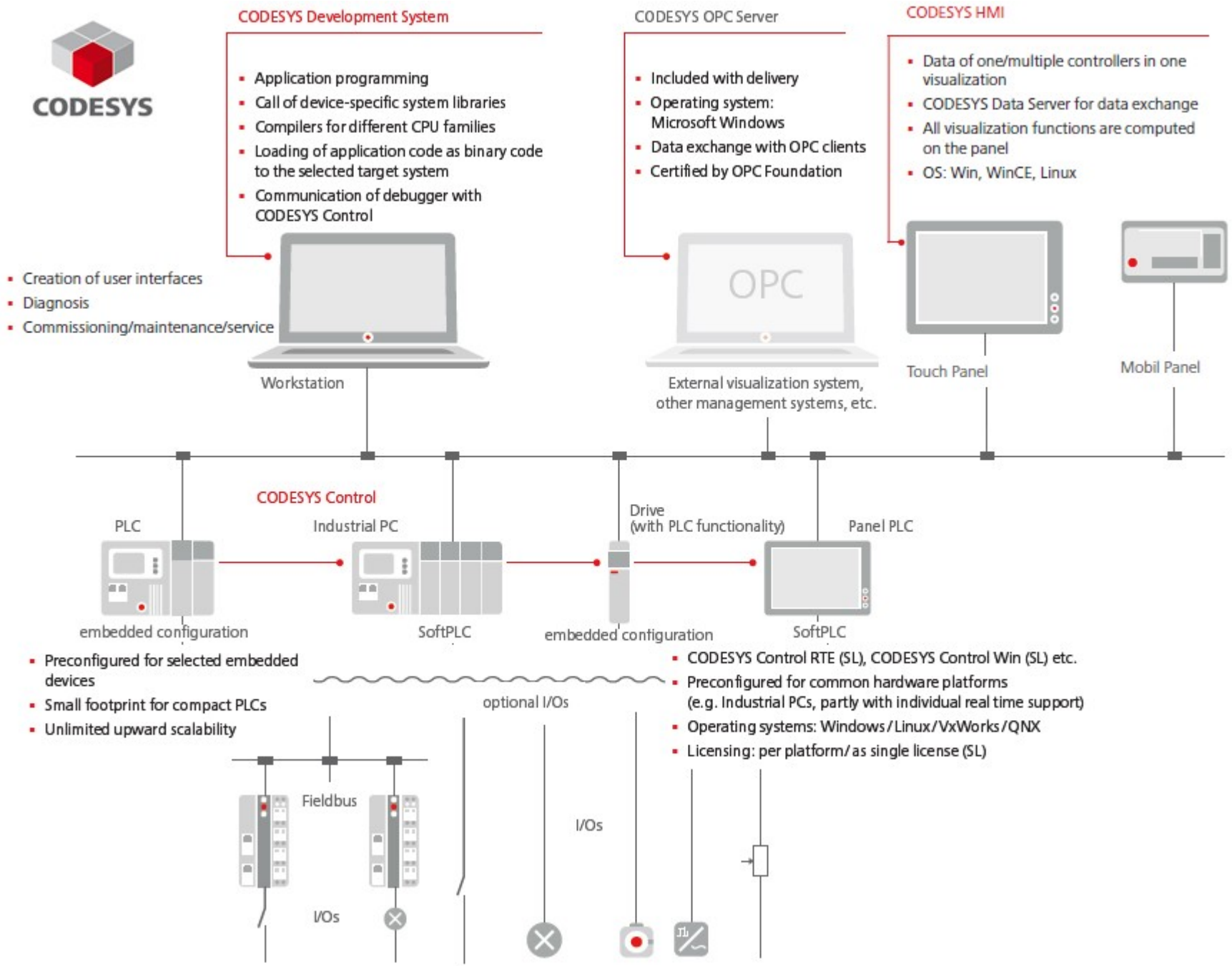
PLC Software Design

- Editors for User Applications
- Embedded Firmware

CODESYS Dev & Run-time



Codesys provides Industry Standard Editors
Codesys generates a runtime binary for PC based system or embedded controller.


CODESYS
CODESYS Development System
Application programming
Call of device-specific system libraries
Compilers for different CPU families
Loading of application code as binary code to the selected target system
Communication of debugger with CODESYS Control
CODESYS OPC Server
Included with delivery
Operating system: Microsoft Windows
Data exchange with OPC clients
Certified by OPC Foundation
CODESYS HMI
Data of one/multiple controllers in one visualization
CODESYS Data Server for data exchange
All visualization functions are computed on the panel
OS: Win, WinCE, Linux
Creation of user interfaces
Diagnosis
Commissioning/maintenance/service
Workstation
OPC
External visualization system, other management systems, etc.
Touch Panel
Mobil Panel
CODESYS Control
PLC
Industrial PC
Drive (with PLC functionality)
Panel PLC
embedded configuration
SoftPLC
embedded configuration
SoftPLC
Preconfigured for selected embedded devices
Small footprint for compact PLCs
Unlimited upward scalability
CODESYS Control RTE (SL), CODESYS Control Win (SL) etc.
Preconfigured for common hardware platforms (e.g. Industrial PCs, partly with individual real time support)
Operating systems: Windows / Linux / VxWorks / QNX
Licensing: per platform / as single license (SL)
optional I/Os
Fieldbus
I/Os
I/Os

Open-Source Systems

Role of Open Source in the PLC / Industrial Automation Domain

- Limited by
  - Stringent Safety, Performance and Liability Requirements, High Cost, limited access for trails at large installations, proprietary nature of existing hardware
  - Customer Risk Aversion and tendency to order what is proven from large reputable vendors
  - Customer lack of IT Expertise and focus on Production core-competency
  - Educational Institutes treat PLCs as Black-Boxes and do not focus on how the Industry Standards should be used to design the underlying hardware and software
- Almost all Open Source hardware offerings are related to the Arduino(roughly ARM Cortex-M4) and RasberryPi(ARM Cortex-A) platforms

Contribuições de nossos amigos brasileiros …

https://www.openplcproject.com/
http://scada-lts.org/

The PLC Runtime is targeted at RasberryPI devices with Arduino devices treated as Remote-IO Modbus Slaves

Contribuciones de nuestros amigos españoles

https://www.industrialshields.com/

PLC based on Arduino

Programmed using Arduino IDE or Ladder logic using

http://soapboxautomation.com/

Editor may not be IEC61131-3 compliant.

https://rapidscada.org/

HMI based on TouchberryPi

Programmed using

Rapid SCADA

Contributions from the US https://cq.cx/ladder.pl

Ladder Editor with compiler for ATMega and PIC targets.
Alternatively ANSI C code can be generated and used with the C-Compiler for another hardware target
Alternatively Byte-Code can be generated and used with target specific interpreters

Editor is old with 1990s look and feel and may not be IEC61131-3 compliant. Not Industry-grade, but should be a good starting point for students

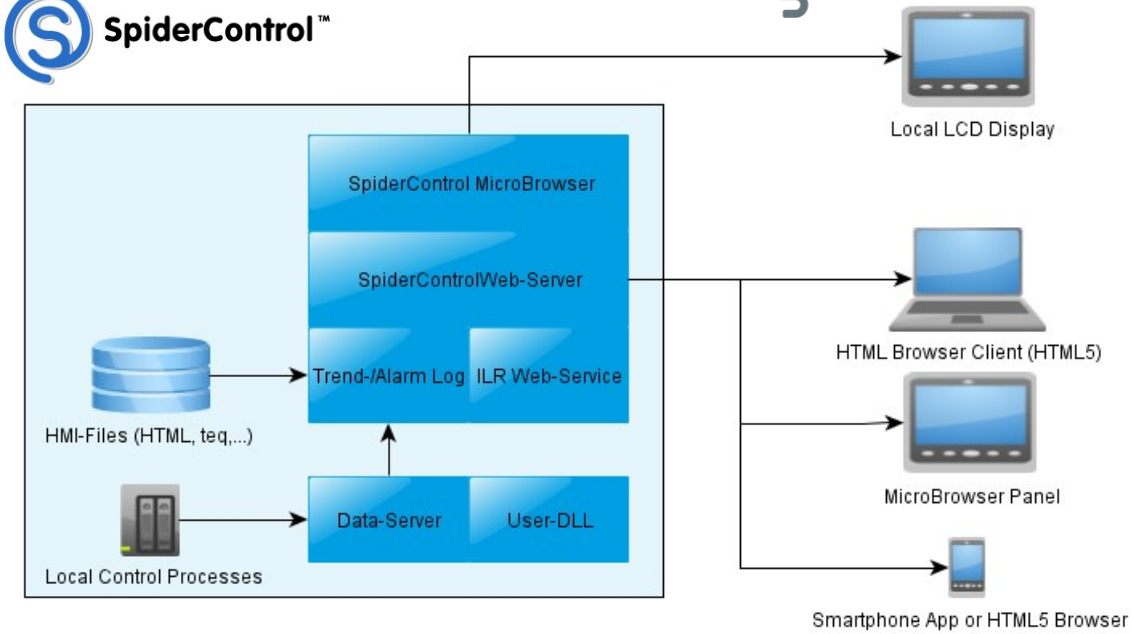
NOTE : The hardware and software mentioned MAY NOT BE 100% free and/or open-source
Some may require significant effort to set-up and use and some are fairly expensive

Beiträge unserer österreichischen freunde https://www.controllino.biz/

PLC based on Arduino
Programmed using Arduino IDE or
Ladder logic using https://www.logicals.com/en/

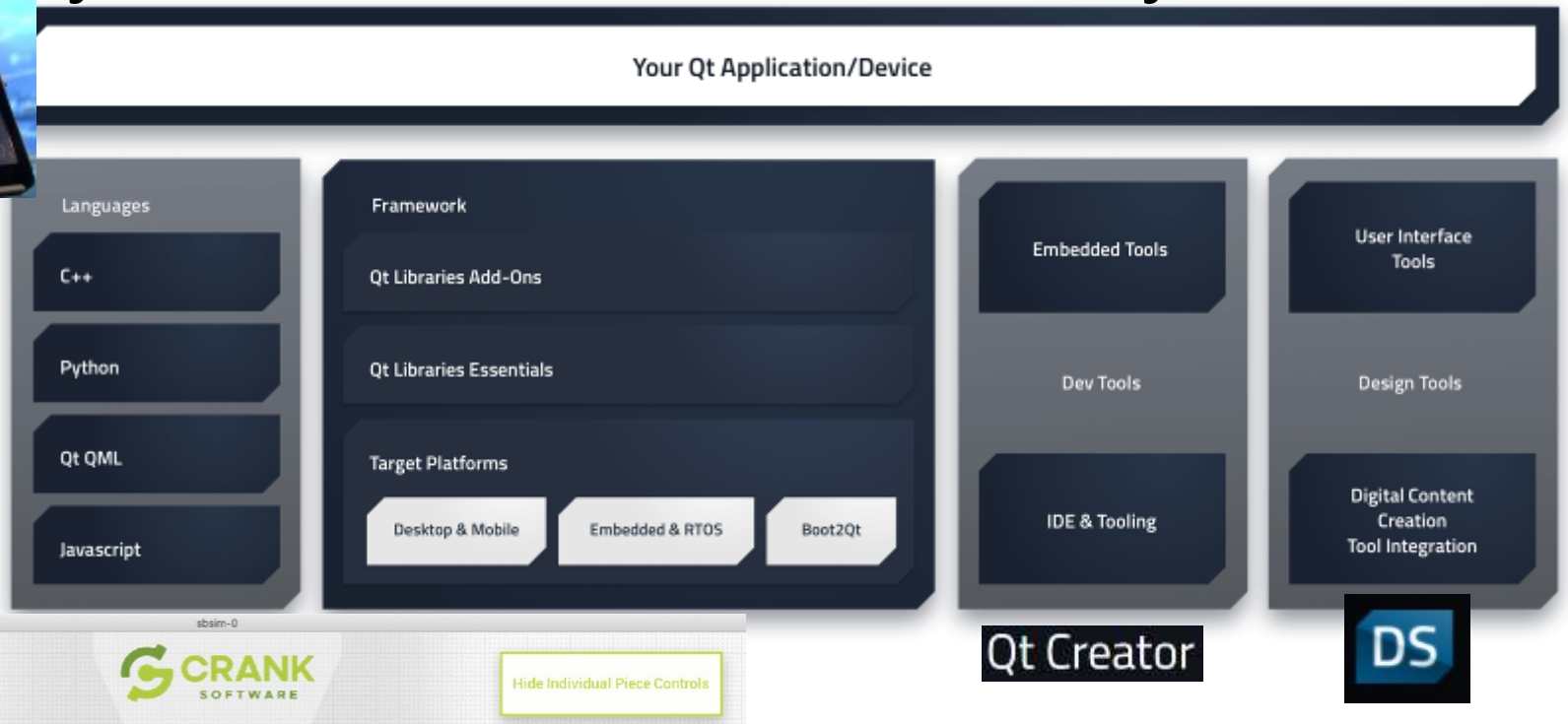
The RevolutionPi eco-system https://revolution.kunbus.com/

- Hardware built around RasberryPi Compute Module with Industry certifications
- Software
  - OS uses Rasbian with real-time patch for kernel
  - Supports CodeSys Editor and CodeSys Control Runtime
  - Supports logi.Cad3 Editor and logi.RTS Runtime
  - SpiderControl WebHMI
  - IoT Support
- IPC with PAS-type capability



Cross Platform Graphic Frameworks to build your own HMI and SCADA System

https://www.qt.io/



CRANK SOFTWARE AMETEK

https://www.cranksoftware.com/

wxWidgets Cross-Platform GUI Library

http://www.wxwidgets.org/

GTK

https://www.gtk.org/

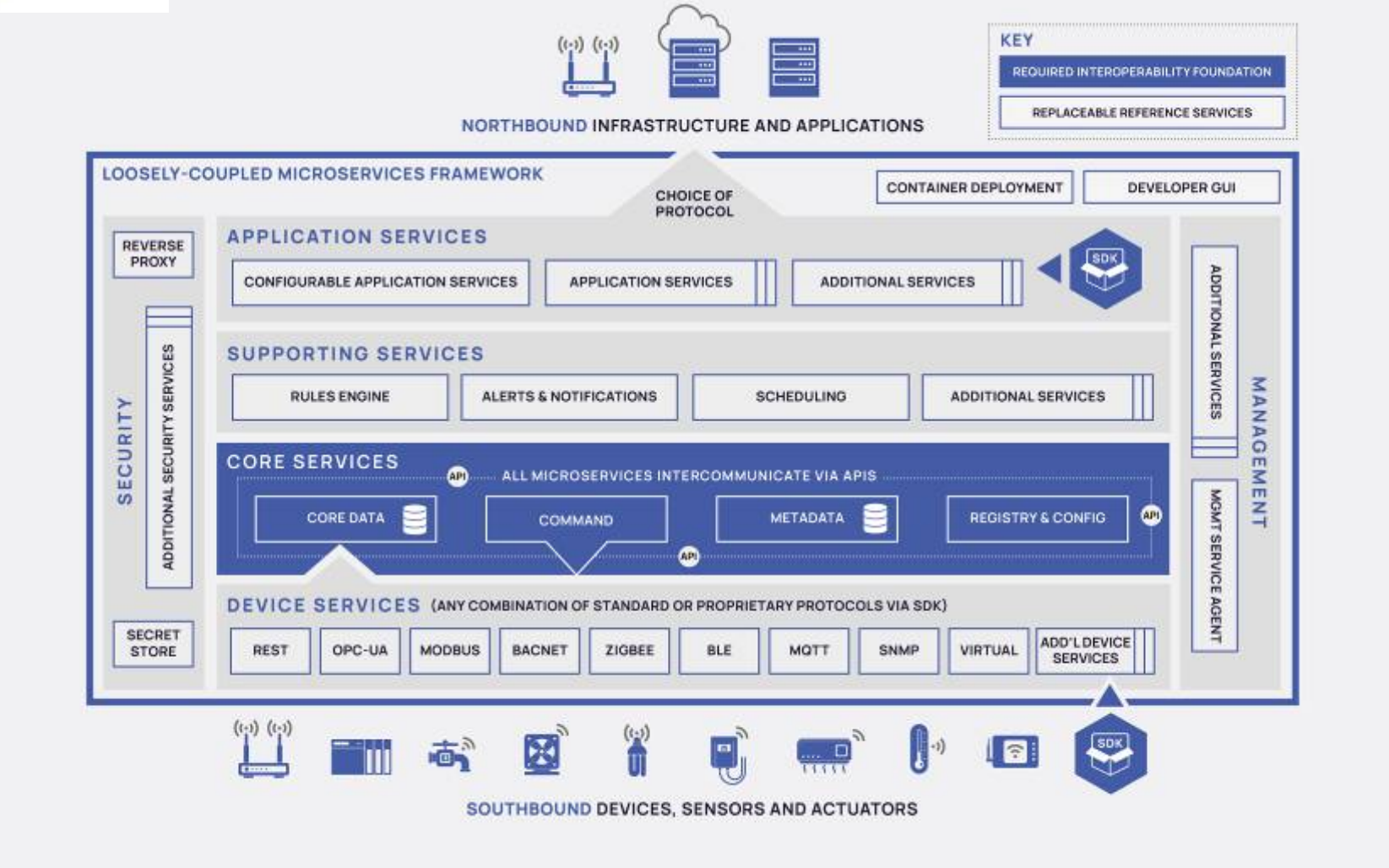
Free SCADA for the Brave

Home Site	Source	Architecture	Technology
http://rapidscada.org/	https://github.com/RapidScada	Communicator+Server+ Webstation	C#
http://pvbrowser.de/pvbrowser/index.php	https://pvbrowser.de/pvbrowser/download.php?file=pvb.tar.gz	Driver+WebServer	C++, QT
http://oscada.org/	http://oscada.org/main/download/	Driver+WebServer	C++, QT
http://scada-lts.org/	https://github.com/SCADA-LTS/Scada-LTS/releases/tag/v2.2.1	Webserver	Java
http://www.proview.se/	https://sourceforge.net/projects/proview/files/proview/	Webserver	Java
https://www.openapc.com/	https://halaser.eu/download.php	Webserver	C, Java
https://szarp.org/en/	https://gitlab.newterm.pl/Newterm/szarp	Webserver	C, C++, QT, Lua
IndigoSCADA download | SourceForge.net	https://sourceforge.net/projects/indigoscada/files/ https://github.com/jonathanxavier/IndigoSCADA/tree/master/src	Observer Pattern	C, C++, QT
Scilab-SCADA https://www.scilab.org/	https://atoms.scilab.org/toolboxes/opc_client/1.3.1 http://atoms.scilab.org/toolboxes/modbus	Scilab	Scilab
http://taurus-scada.org/	https://github.com/taurus-org/taurus	Model-View-Controller	Python, QT, PyQT, PyTango, numpy,..

https://www.edgexfoundry.org/
https://www.lfedge.org/

LF Edge is an umbrella organization that aims to establish an open, interoperable framework for edge computing independent of hardware, silicon, cloud, or operating system.

May prove interesting… but currently documentation and examples seem to be limited



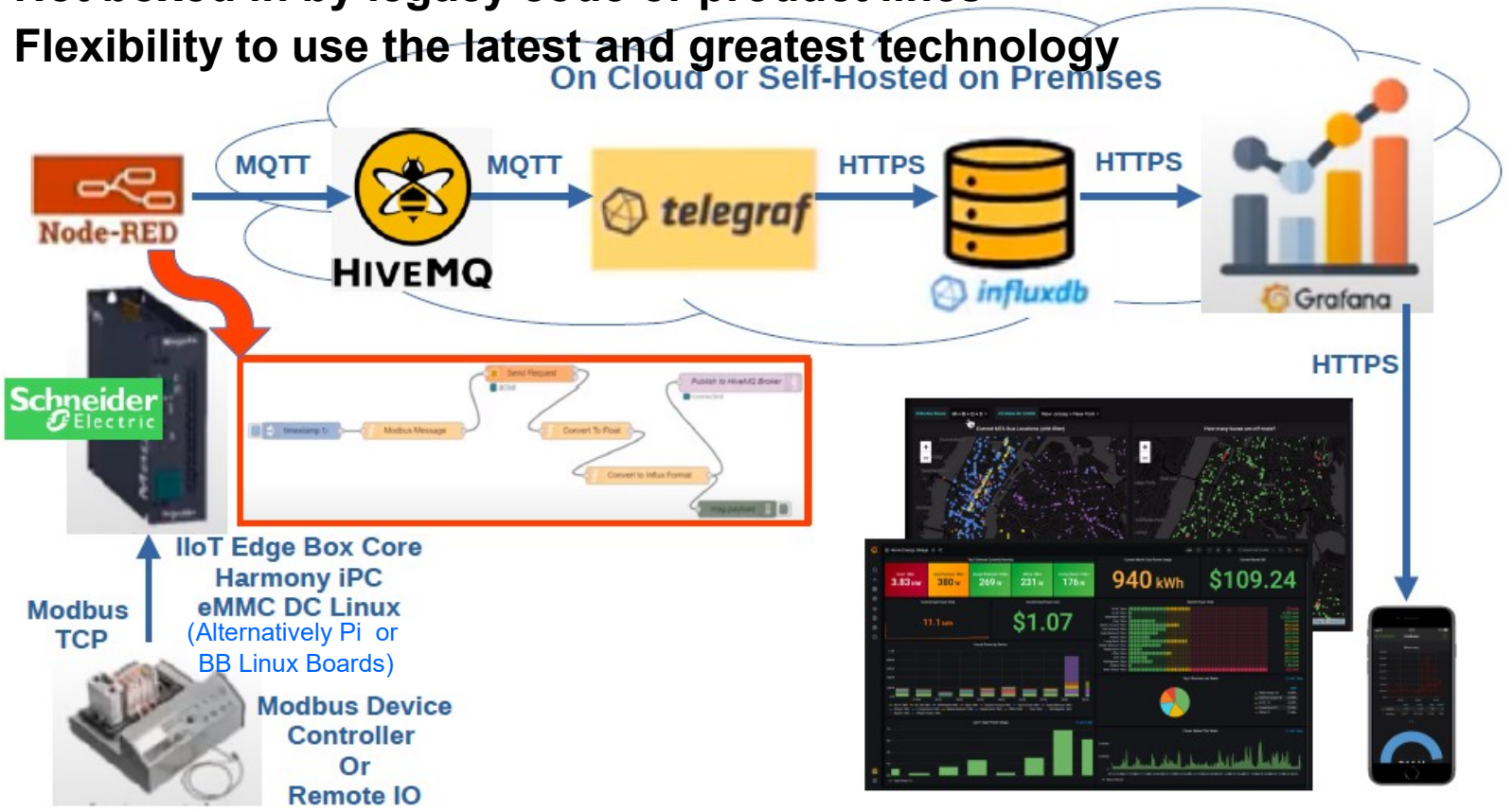
Truly Open SCADA/IoT

Proven Standard IT tools to build a SCADA/IoT system:

Linux, MySQL, MongoDB, PostgreSQL/TimescaleDB, InfluxDB, Node.js, Node-RED, C#, Golang, Grafana, HTML5 Web interface, SVG, Vue.js, Nginx, Apache, PHP and protocol stacks like OpenDNP3, lib60870, lib6150, MQTT, HTTPS

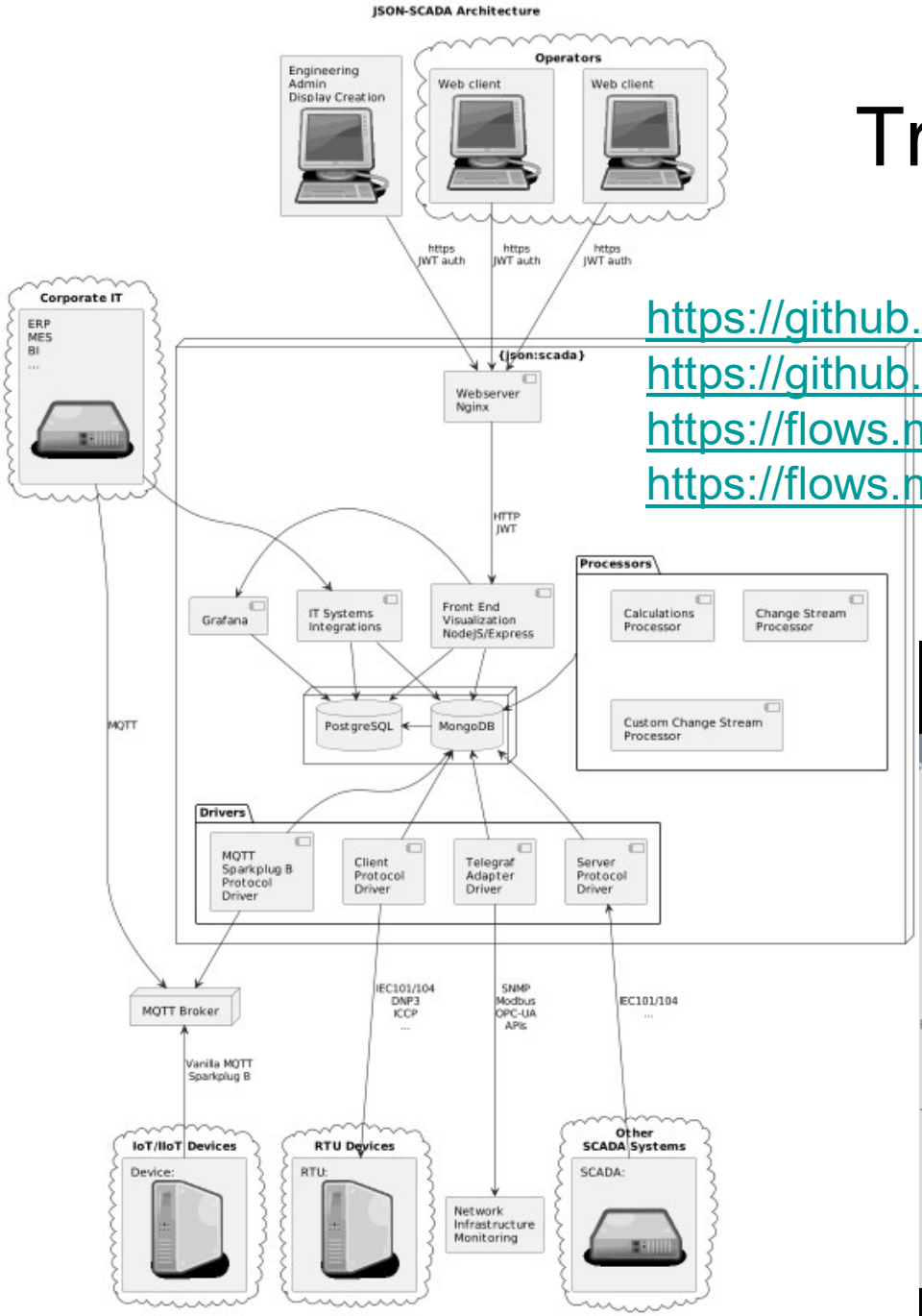
Not boxed in by legacy code or product lines

Flexibility to use the latest and greatest technology

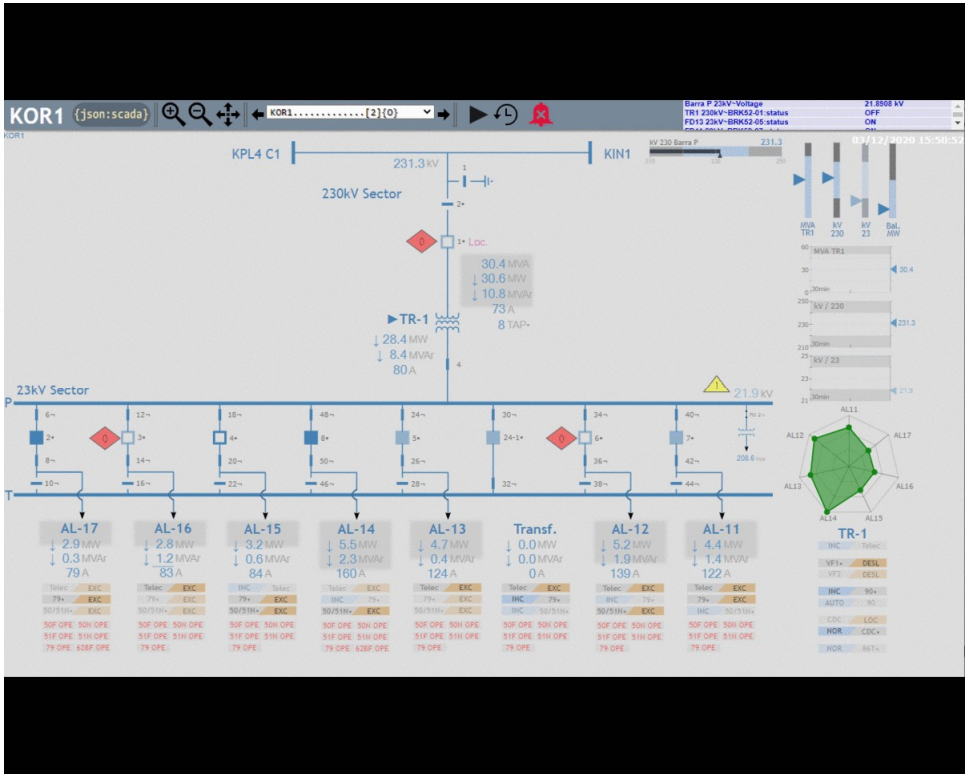


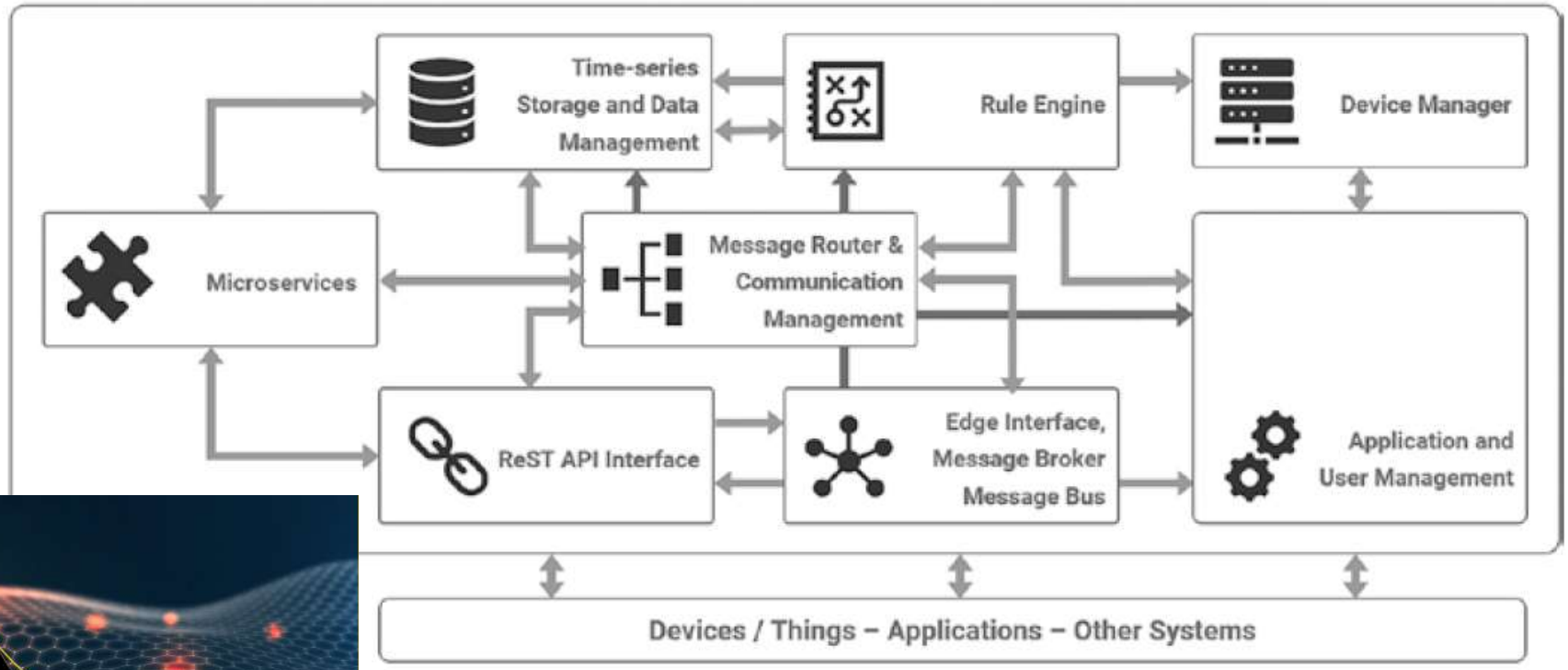
https://www.youtube.com/watch?v=UVWatCq77B0

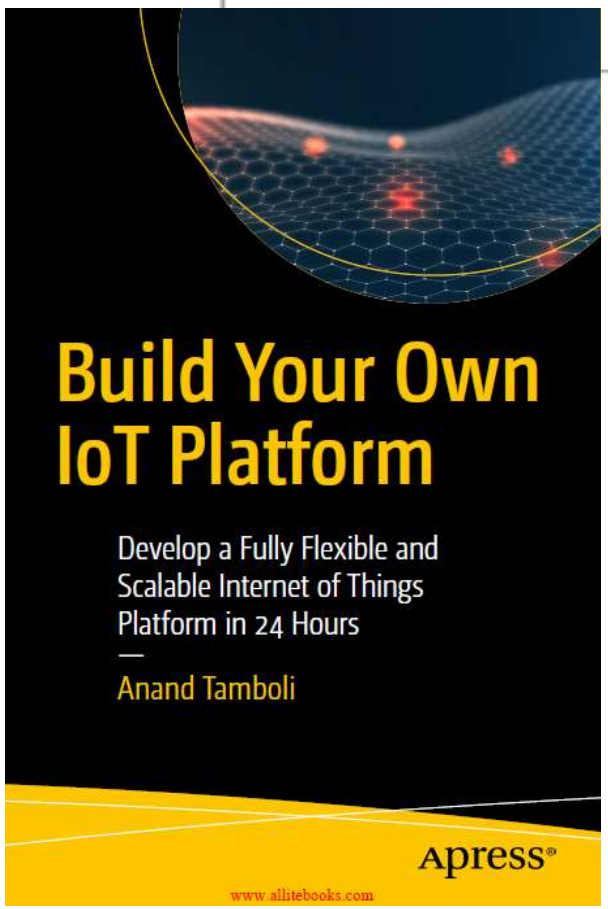
Truly Open SCADA/IoT



https://github.com/riclolsen/json-scada
https://github.com/riclolsen/OSHMI
https://flows.nodered.org/node/node-red-dashboard
https://flows.nodered.org/node/node-red-contrib-uibuilder







MySQL

Node-RED

node

php

mosquitto

APACHE
HTTP SERVER PROJECT

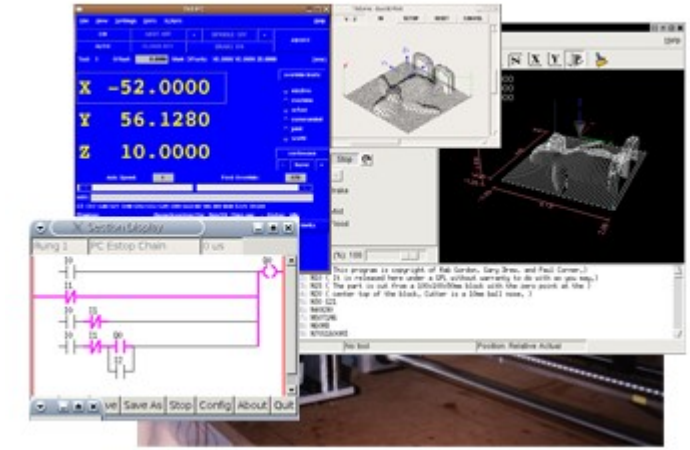
Let's Encrypt

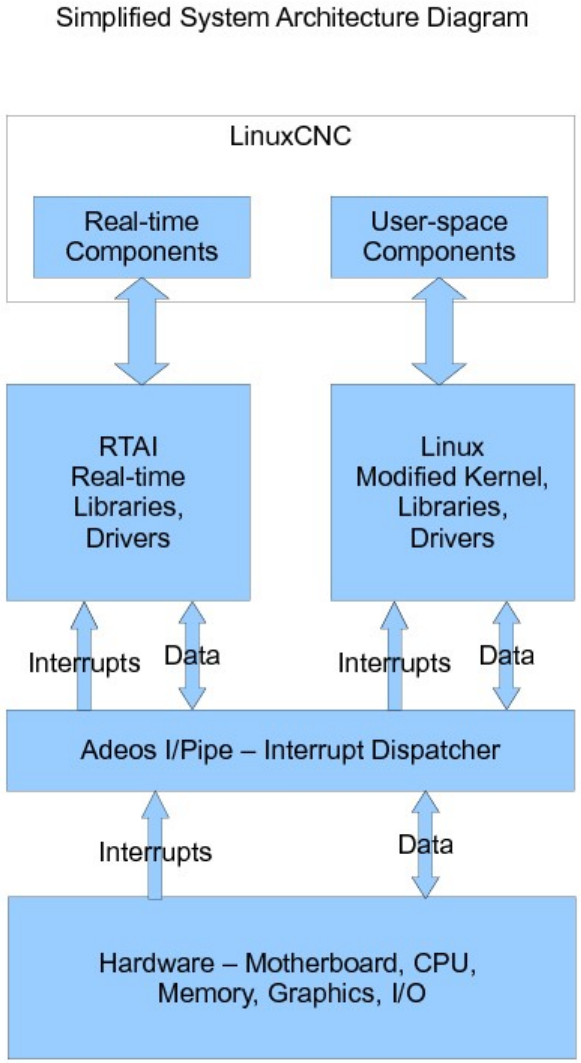
ubuntu

DigitalOcean

CNC Freeware

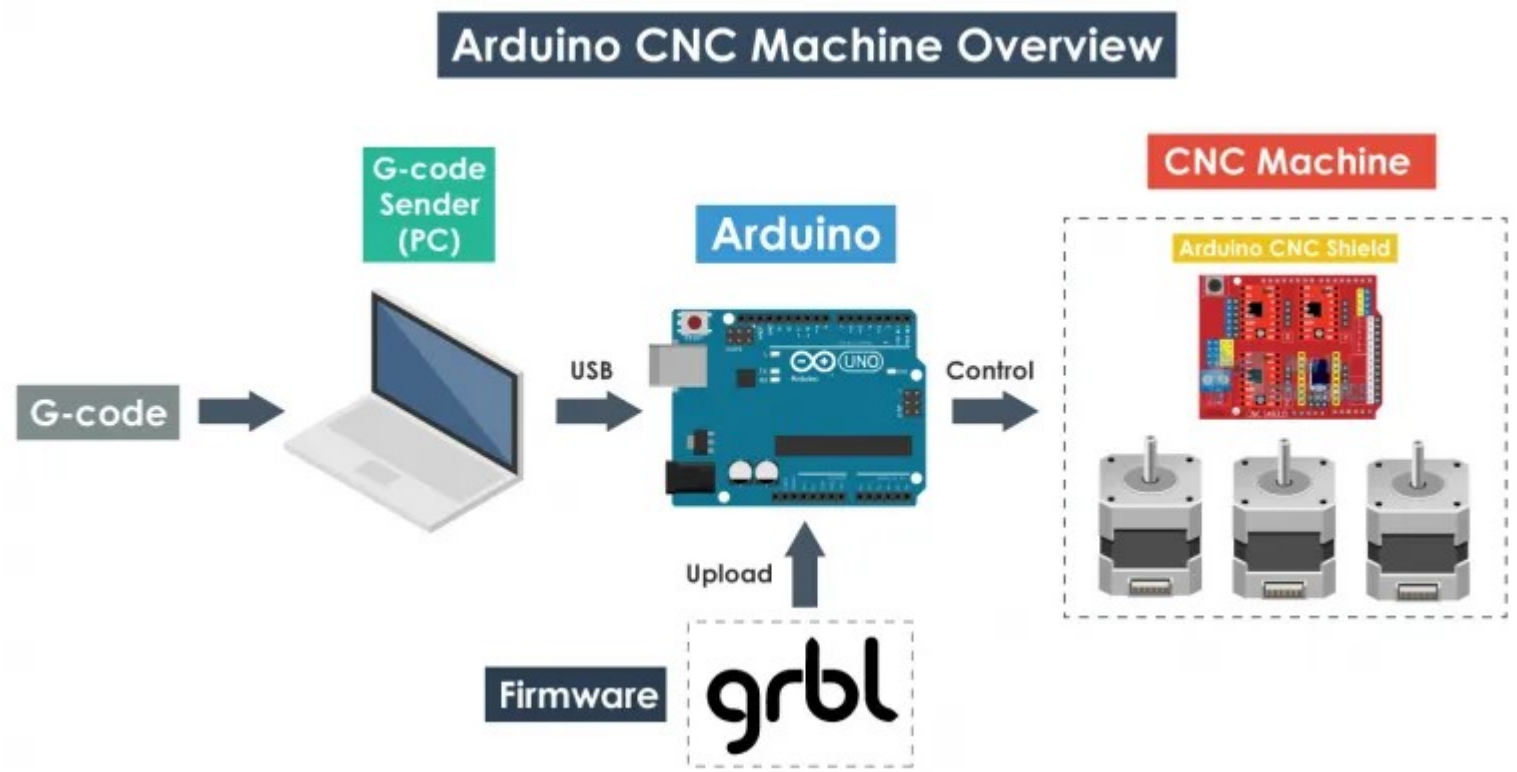
- http://linuxcnc.org/ LinuxCNC (formerly Enhanced Machine Controller or EMC2) implements numerical control capability using general purpose computers to control CNC machines





Kent A Reed, 20120518

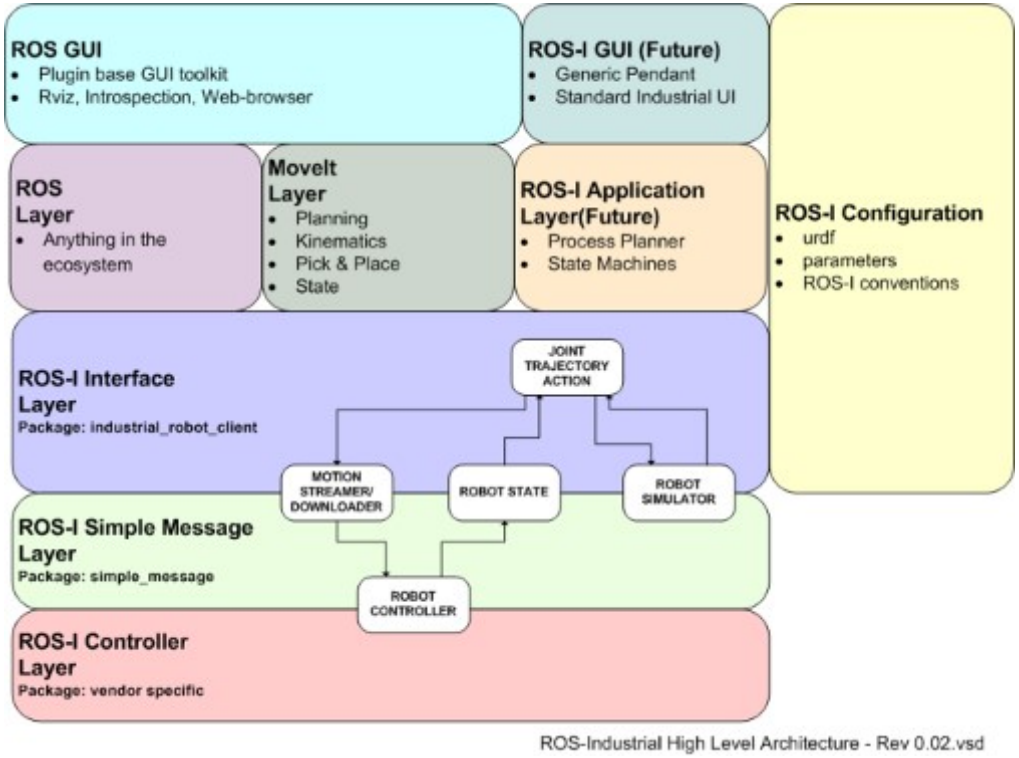
CNC Freeware



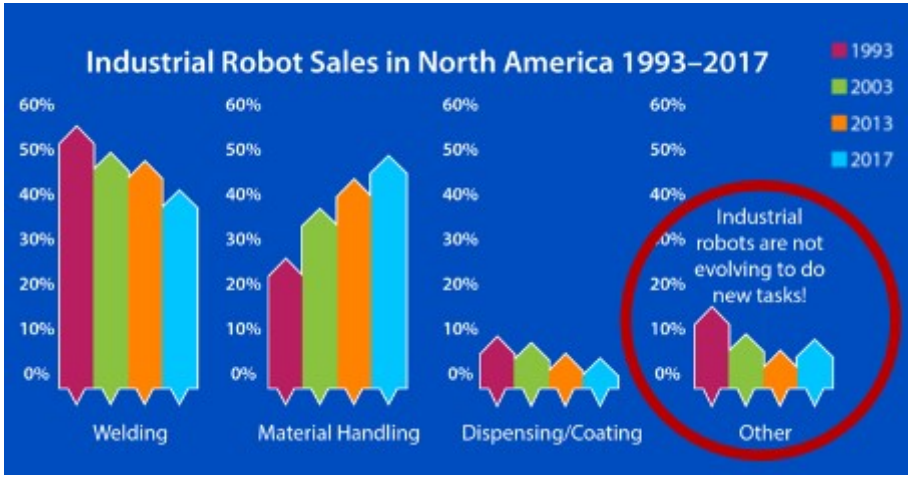
https://howtomechatronics.com/tutorials/how-to-setup-grbl-control-cnc-machine-with-arduino/
https://github.com/grbl/grbl
https://github.com/gnea/grbl/releases

Robot Freeware

- https://www.ros.org/ The Robot Operating System (ROS) is a flexible framework for writing robot software. It is a collection of tools, libraries, and conventions that aim to simplify the task of creating complex and robust robot behavior across a wide variety of robotic platforms.
- https://rosindustrial.org/ to solve the issue that despite billions of dollars of research by governments and academia, new robotic applications were not being introduced in the industry due to incompatibility between the research and the industrial robotic platforms



ROS-Industrial High Level Architecture - Rev 0.02.vsd



Industrial Cyber Security

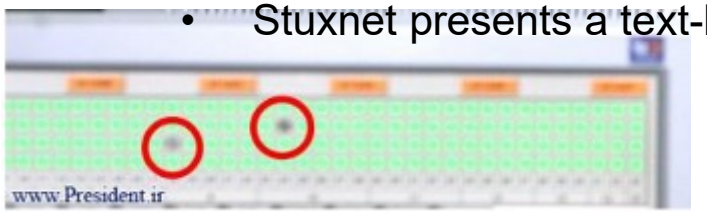

Since 2010, there have been several sophisticated cyberattacks that targeted ICS networks such as Stuxnet and Industroyer.
Year
Major ICS Security Incidents
Attack Targets
2010 Stuxnet PLC (Nuclear Power Plant in Iran)
2011 Duqu Computer/Server (Public Utility in Multiple Countries)
2012 Disttrack/Shamoon Computer/Server (Oil Company in Saudi Arabia)
2014 Sandworm SCADA/HMI (Factory Floor in Multiple Countries)
2015 BlackEnergy/Killdisk HMI/Serial Device (Power Grid in Ukraine)
2016 Industroyer Circuit Breaker (Power Substation in Ukraine)
2017 Dragonfly Computer/Server (Public Utility in US/EU)
2018 WannaCry Computer/Server (Factory Machines in Asia)

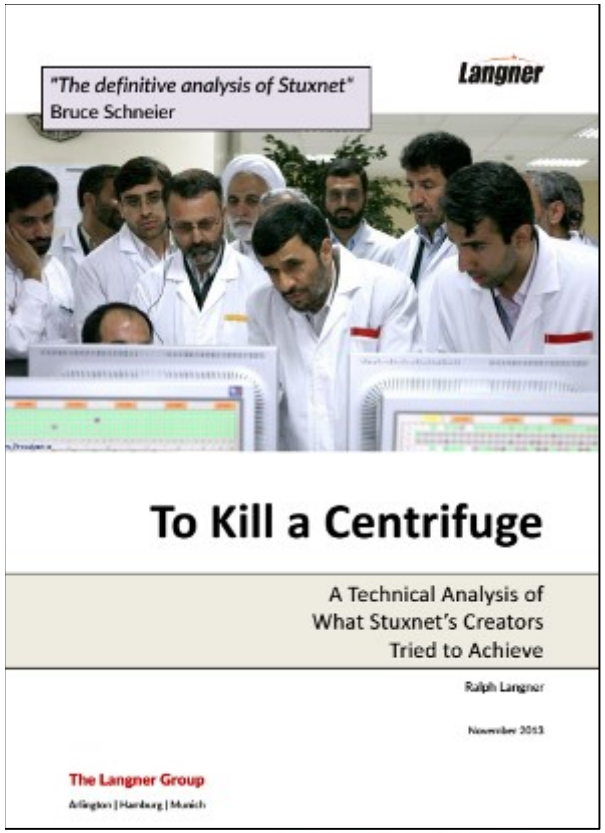
Stuxnet… Worm, Trojan, Virus

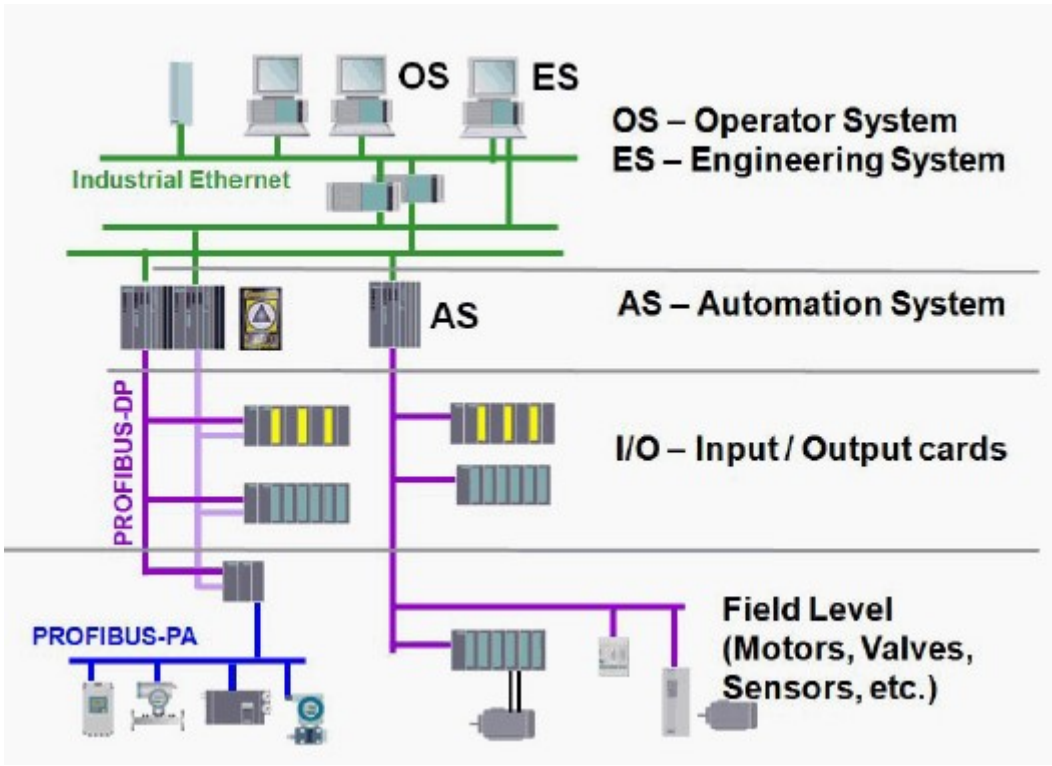
- State sponsored Cyber Weapon specifically targeted Iranian Uranium refining centrifuges
  - Delivery Introduced into the computer networks of several Iranian Industrial Automation companies by USB pen-drives “found” near the organizations

    - Spread through the company networks using multiple “zero-day exploits” on vulnerabilities in RPC, Printer sharing on Windows PCs and replicated itself on all USB pen-drives used on the networks
    - Jumped the “air-gap” to the isolated Industrial Control System network (ICS n/w) at the Iranian Natanz Nuclear Facility on a USB pen-drive and replicated across the Natanz ICS n/w
    - On PCs with Siemens PCS7 (DCS) & Step7 Programming Software used an unchanged “Default Password” that gave Admin access to install/replace a .dll file using a genuine (but stolen) authentication key
  - Payload Scanned the PLC program for certain key parameters example “164” => typical centrifuges in a cascade and configured Profibus network for Control Valves & Motor Control Drives

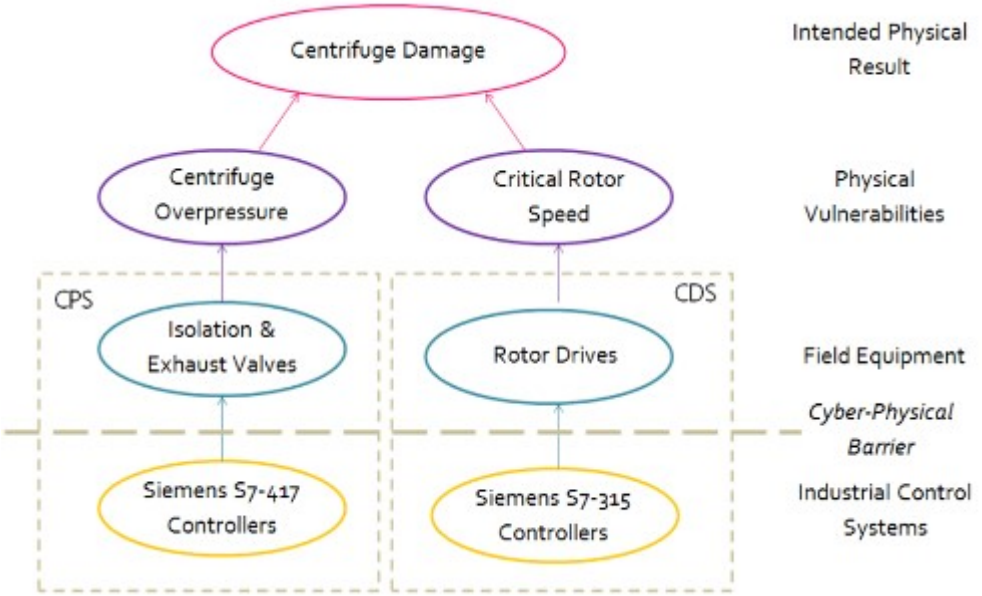
    - Updated the Controller Program when Step7 connected to S7-400/S7-300 Controller Hardware
    - At pre-programmed intervals every around 2 weeks
      - Varied gas pressure to over-pressurize a centrifuge stage
      - varied the set-speed of the drives drastically taking them near their critical resonance speeds
    - The Controller could continue to report correct Gas Pressure & Drive Speed values to the SCADA so there would be no automatic shutdown by safety routines or chance for manual intervention from the Control Room or post-mortem troubleshooting when analyzing logged process data. The durations of the injected malfunctions were short so as to increase wear and tear on the motor and mechanical systems but not cause immediate catastrophic destruction
    - The slow delayed action – an event every 2 weeks
      - made problem isolation difficult
      - could destroy even replacement hardware
      - caused frustration and loss of morale among the project team
  - Effect
    - 2000 centrifuges around 1/5th of the total installation were destroyed and had to be completely replaced
- Stuxnet presents a text-book methodology of “Attack Engineering”







Core Functional Components of the Siemens SIMATIC PCS 7 Control System



- A cyber-physical attack involves three layers and their domain-specific vulnerabilities:
  - ➢ IT layer which is used to spread the malware – exploiting
  - ➢ Control system layer which is used to manipulate (but not disrupt) process control
  - ➢ Physical layer where the actual damage is created

Industrial Cyber Security

- Open communication and the increased networking of production systems involve not only huge opportunities, but also high risks
- Effects of a Cyber attack on Industrial system
  - Safety
  - Quality
  - Uptime
  - Espionage
  - Ultimately resulting in revenue loss
- IEC 62443 International Industrial Security Standard

Cisco Industrial Security Appliance ISA 3000
-Firewall & N/w Security Policies

IEC 62443 security levels

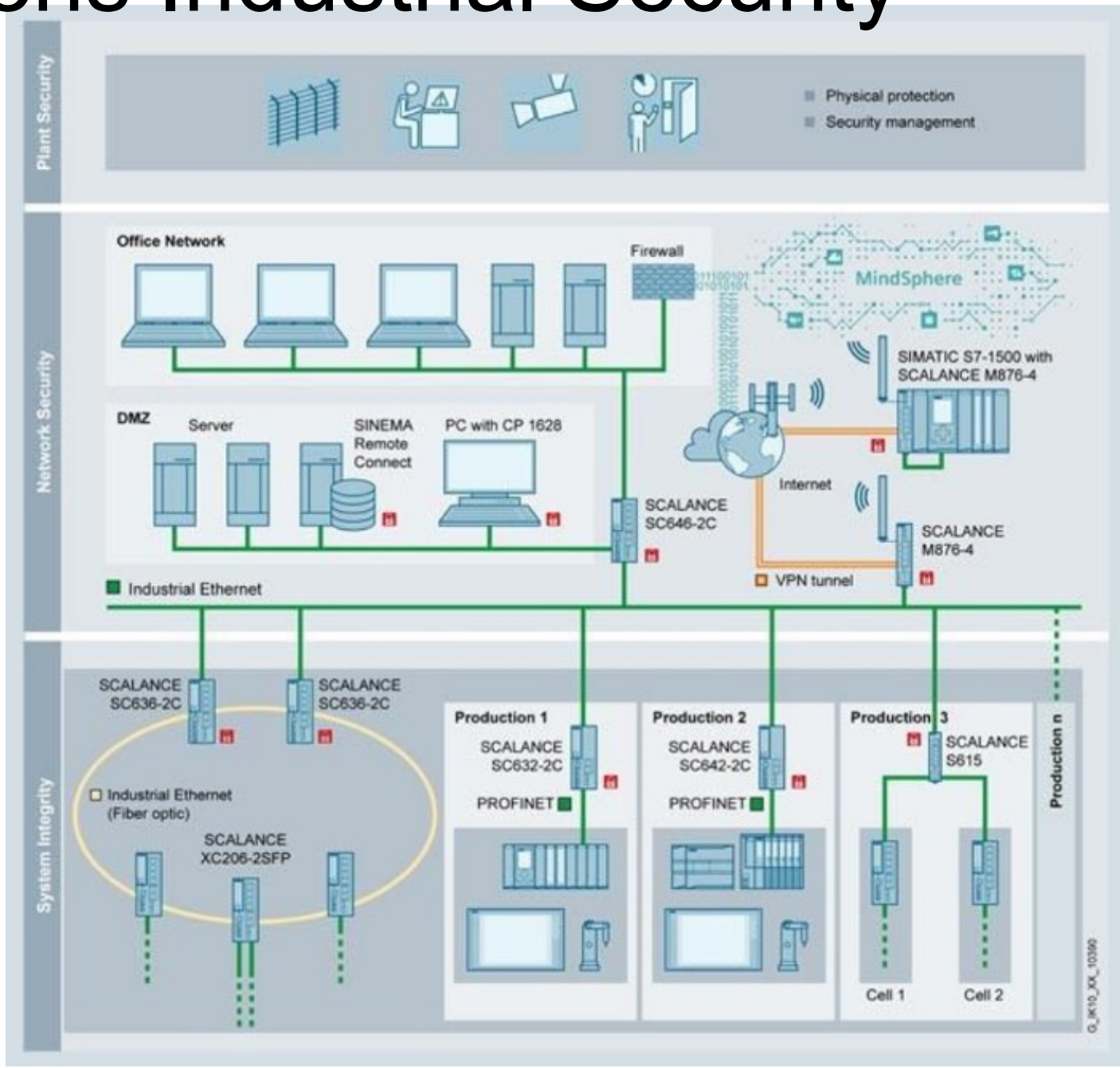
IEC 62443	Skills	Motivation	Means	Resources
SL4 Nation-state	ICS Specific	High	Sophisti-cated (Campaign)	Extended (multi-disciplinary teams)
SL3 Hacktivist, Terrorist	ICS specific	Moderate	Sophisti-cated (Attack)	Moderate (groups of hackers)
SL2 Cyber crime, Hacker	Generic	Low	Simple	Low (Isolated individuals)
SL1 Careless employee, contractor	No attack skills	Mistakes	Non-intentional	Employee, contractor

Layered Security

- Plant security
  - Protection against access by unauthorized persons – fenced zones based on roles
  - Physical access protection for critical components – locked cabinets
- Network security
  - Controlled interfaces between the office and plant networks, e.g., using firewalls, DMZ
  - Segmentation of the plant network (could be using vLANs)
    - Group on need-to-communicate basis
    - Group Legacy systems / out-dated OS
    - Groups behind VPN, Firewalls
  - Network Monitoring – traffic density, intrusion, blocking of devices attempting to flood the network at port / segment level
  - Secure IT Functions
    - Enable Encrypted Communications where possible
    - Web-Access using encrypted protocols, VPN, IPSec, HTTPS, MQTT over TLS, FTPS, NTP(secured), Secure SNMP
- System integrity
  - Use of antivirus software with regular updates of virus signatures
  - Only “Whitelisted” programs are allowed to execute on PCs
  - Maintenance and update processes
  - User authentication for machine or plant operators – use Windows Users, Policies – password life
  - Device authentication by passwords and certificates
  - Integrated access protection mechanisms in automation components
  - Disable unused ports Ethernet/USB/Fieldbus/ WLAN/Bluetooth..
  - Accept messages only from known sources
  - Disable unused functions Webserver, FTP..
  - PLC program locking, function block locking, upload prevention, Audit Trails

Siemens Industrial Security

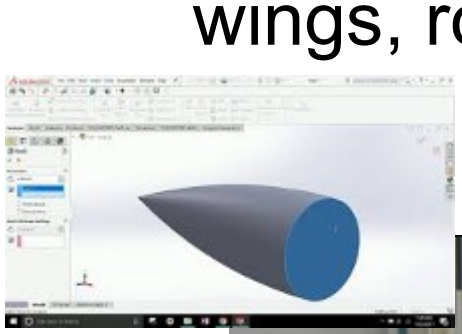
- Industrial Grade Networking Security Modules – Firewall, VPN
- PLC Communication Processors can also act as Firewalls

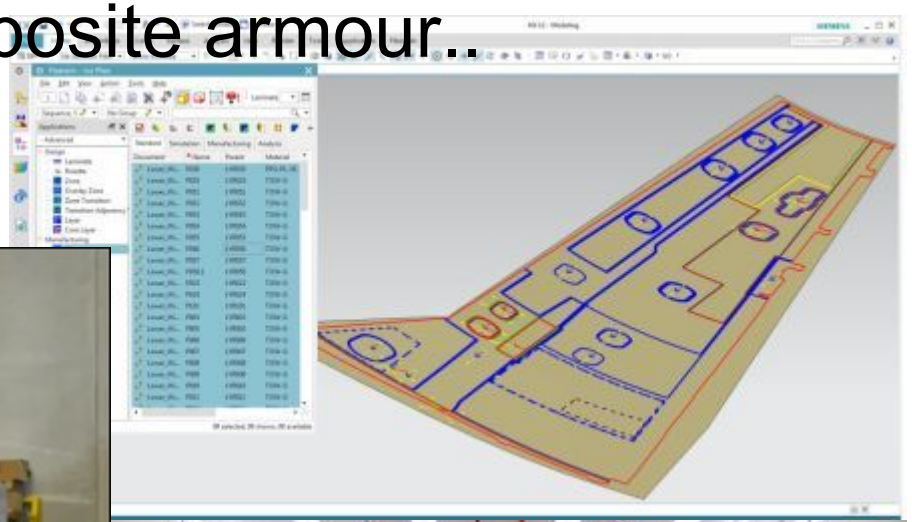


Strategic Role of Industrial Automation

Machining for Nuclear, Weapon Systems

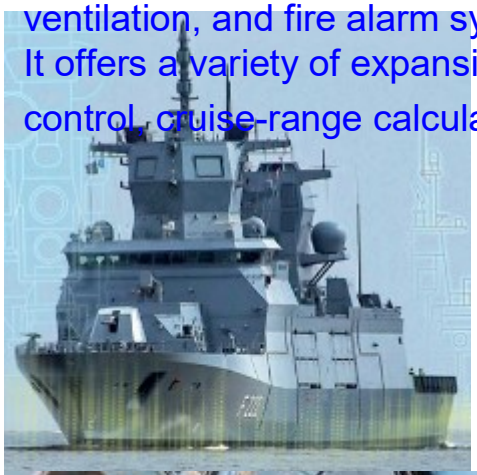
- CAD/CAM for design and analysis
- CNC machining for parts of weapon systems
- Additive manufacturing for composites – aircraft wings, rotor blades, composite armour..

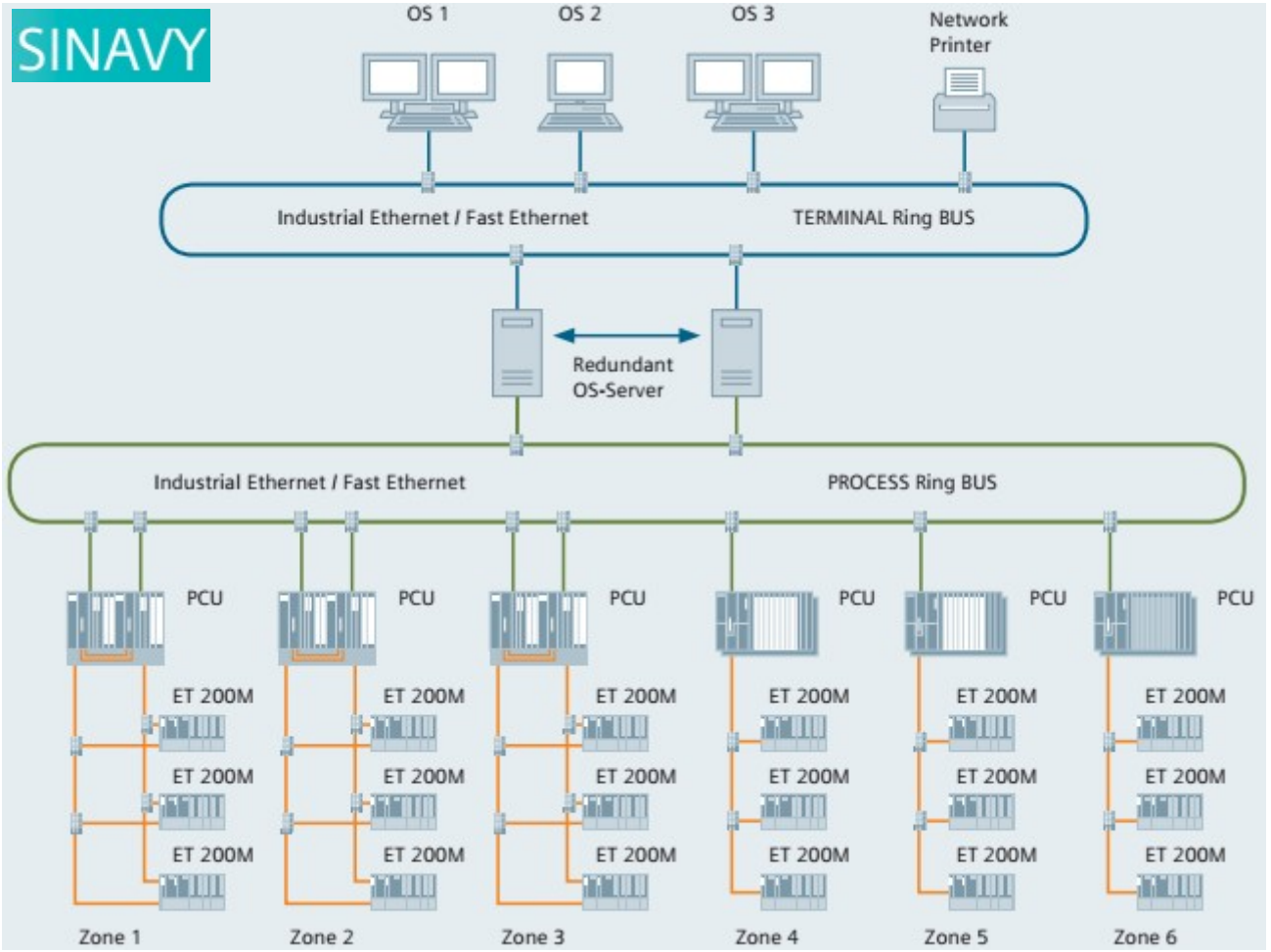




Control of Military Platforms

SINAVY Automation controls and monitors all vessel systems, including diesel engines, gas turbines, gear levels, clutches, water jet pumps, exhaust systems and fuel cells.
It also handles auxillary modules, such as fuel supply and distribution, ventilation, and fire alarm systems.
It offers a variety of expansion possibilities, including battle damage control, cruise-range calculation, central battery and pier monitoring



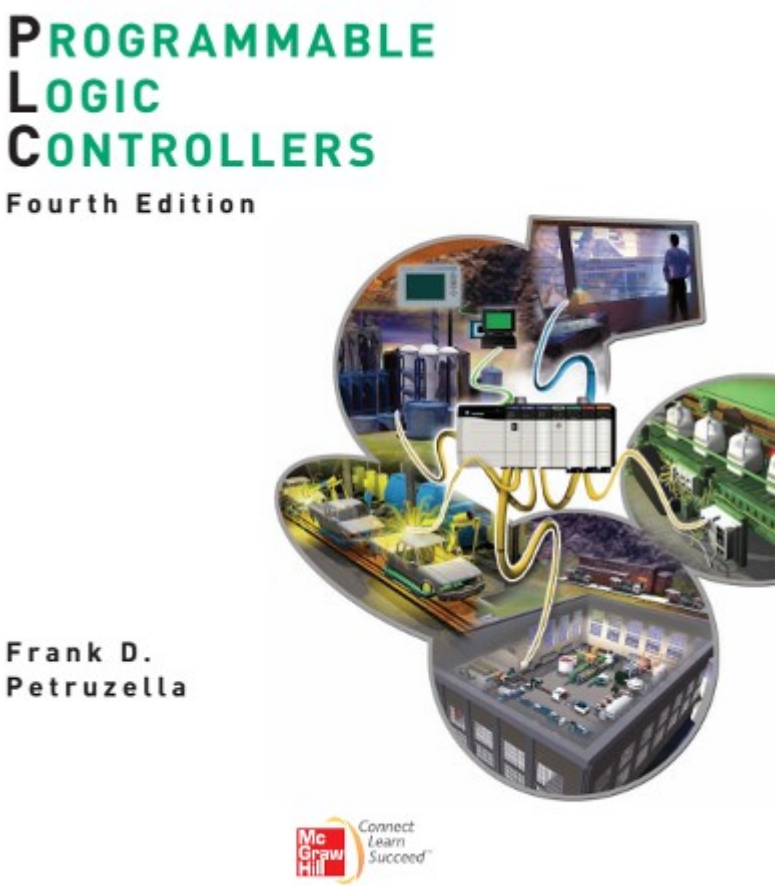


Dual Use Technology

- From fertilizer, food processing, pharmaceuticals, automobiles, steel, cement, chemicals, refineries, nation-wide utility grids electricity/fuel/water, to nuclear, weapons, ammunition, warships…
- PLCs and allied Industrial Automation fields can be applied to the manufacture and control of military systems
- Nations that invest in the developing competency and expertise in research, design, development, production, operation and maintenance of these technologies ensure self-reliance in industrial and military systems

References

- Technical Documents of Equipment Manufacturers and Industry Standards
- Books/Sites referred are mentioned in the presentation



From Small to Strategic
You now see the Bigger Picture
Hopefully this presentation answered many of your questions..and created some new ones..

All the Very Best in Your Automation Endeavors

Stay Safe!

Thank You

Also check out the Industrial Drives Presentation at
https://www.researchgate.net/publication/354336214_Overview_of_Low_Voltage_Industrial_Drives