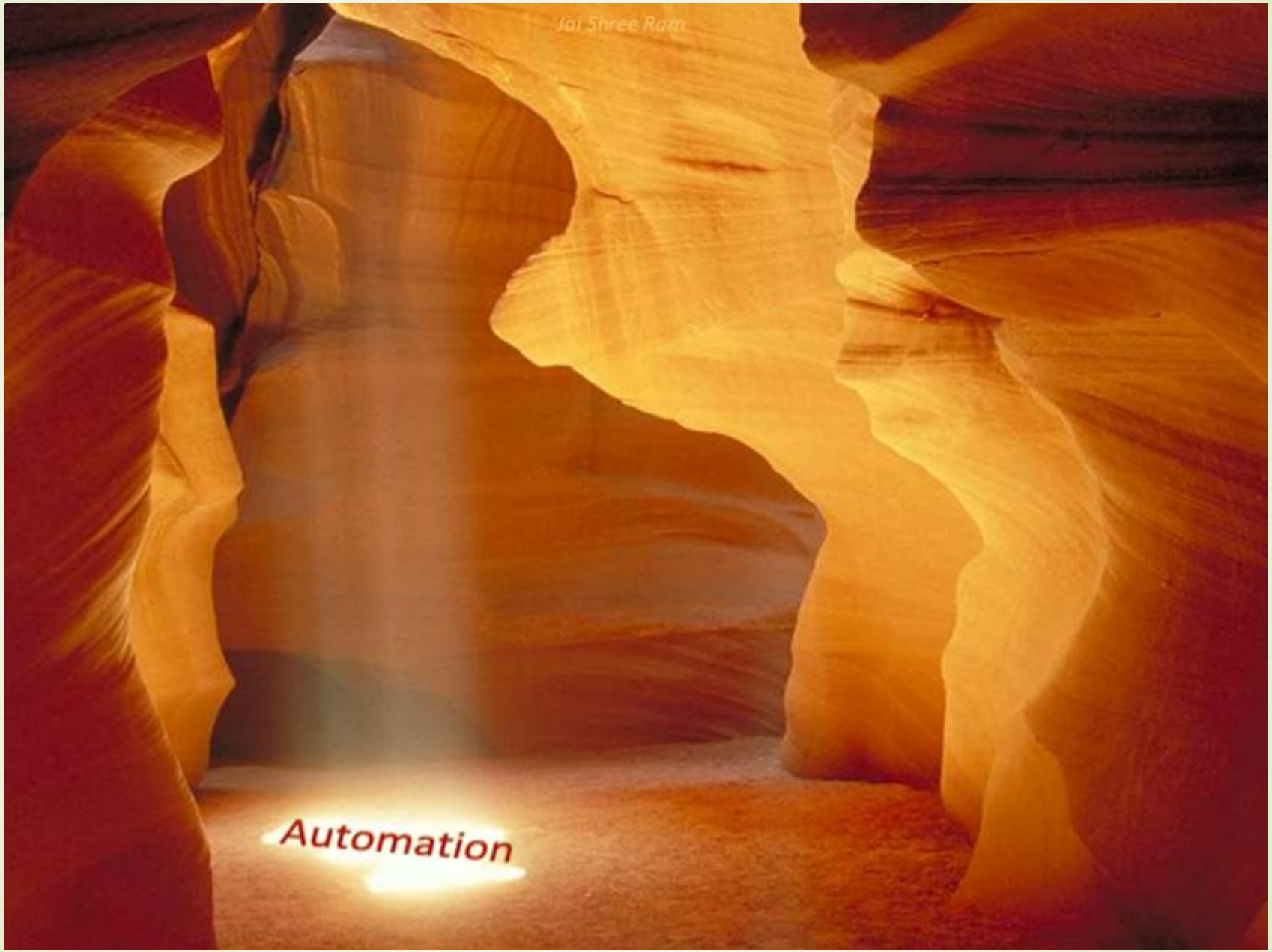
Introduction to PLC and SCADA


CONTENTS
AUTOMATION
PLC
SCADA


Jai Shree Ram
Automation

History of Automation

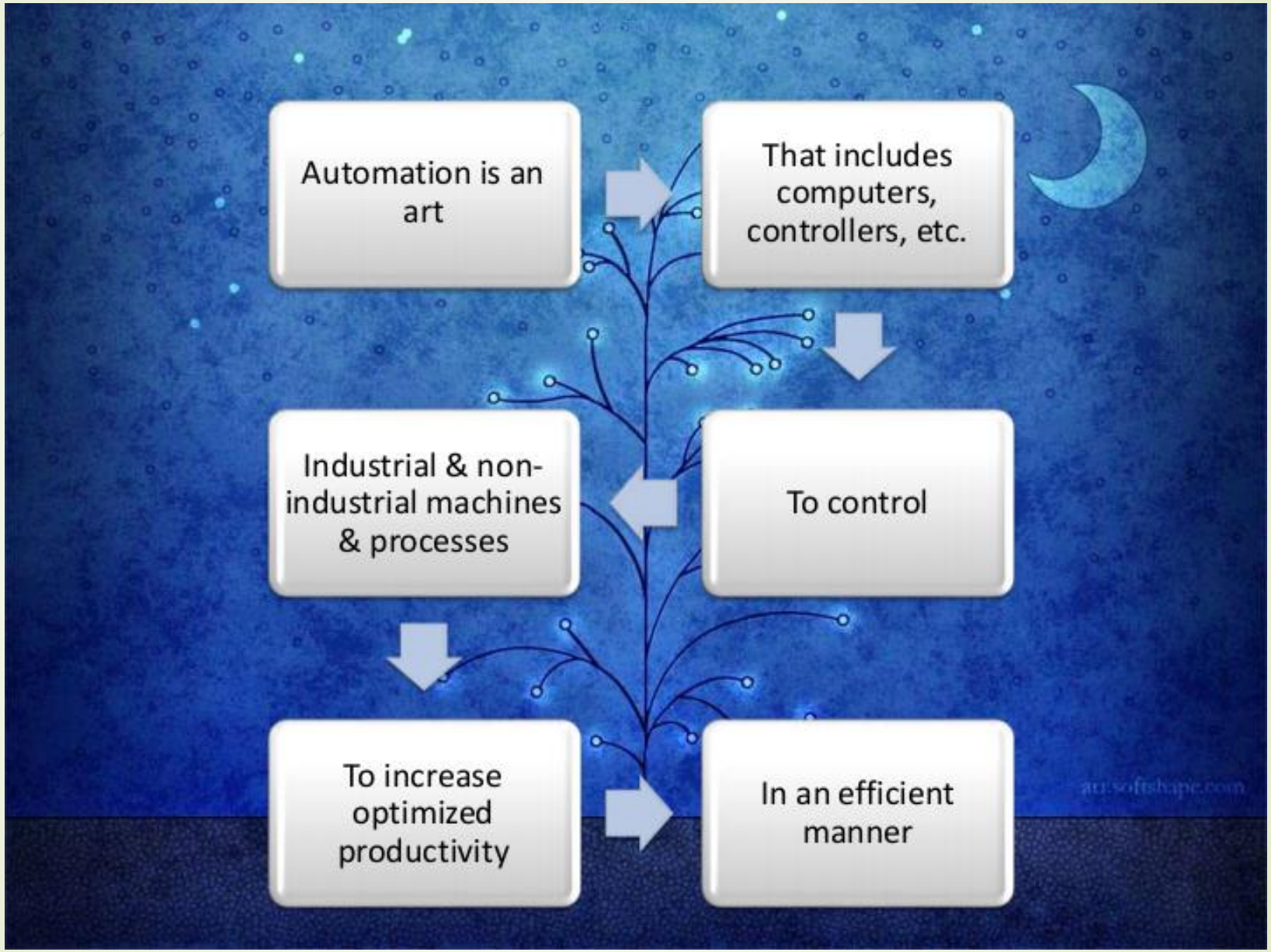
- Manual Control
- Pneumatic Control
- Relay Logic Control
- Electronic Logic Gate Control
- Programmable Logic Gate Control

When the first electronic machine control was designed, relays were to control the machine logic.
Relay logic has its own limitations.

- 1.Less reliability
- 2.The delay involved in switching of contacts
- 3.Less flexibility and difficult troubleshooting due to hard wired connection

Automation

- AUTOMATION is basically the delegation of human control functions to technical equipment aimed towards achieving:
  - Higher productivity.
  - Superior quality of end product.
  - Efficient usage of energy and raw materials.
  - Improved safety in working conditions etc.


Automation is an art
That includes computers, controllers, etc.
Industrial & non-industrial machines & processes
To control
To increase optimized productivity
In an efficient manner

TYPES OF AUTOMATION

- Building automation

  Example: lifts, smoke detectors

- Scientific automation

  Example: rocket launching

- Industrial automation

  Example: automated bottle filling stations , steel factories etc

- Office automation

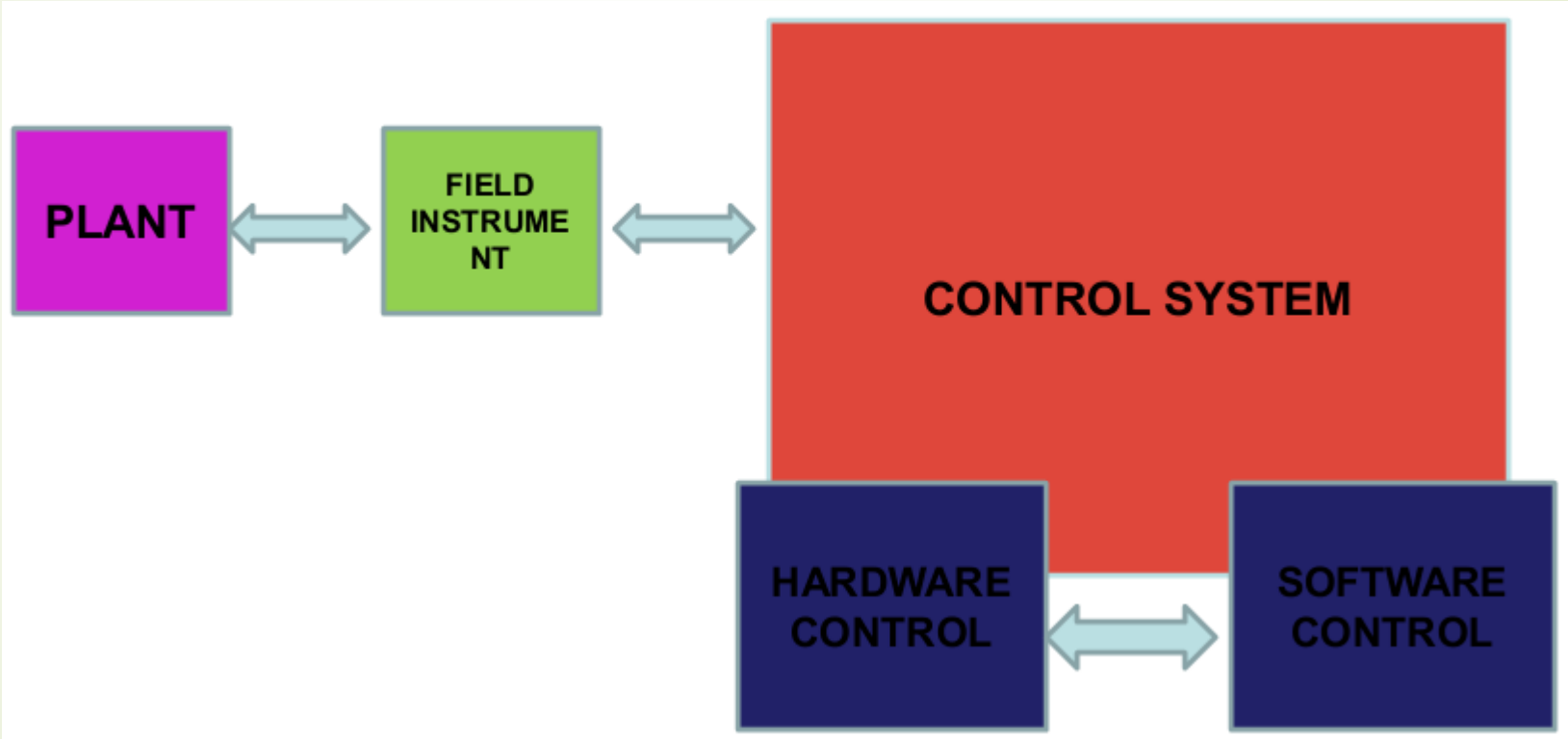
  Example: printers, cctv cameras

- Light automation

  Example: street solar lightening

INDUSTRIAL AUTOMATION

- The use of Computerized or robotic devices to complete manufacturing tasks.



Programmable Logic Controller

PLC

Stands
For

Programmable
Logic
Controller

WHAT IS PLC?

- PLC is an industrial computer that monitors inputs, makes decisions based on its program and controls outputs to automate a process or machine. A PLC has a built-in operating system(OS). This OS is highly specialized to handle incoming events in real-time, or at the time of their occurrence.

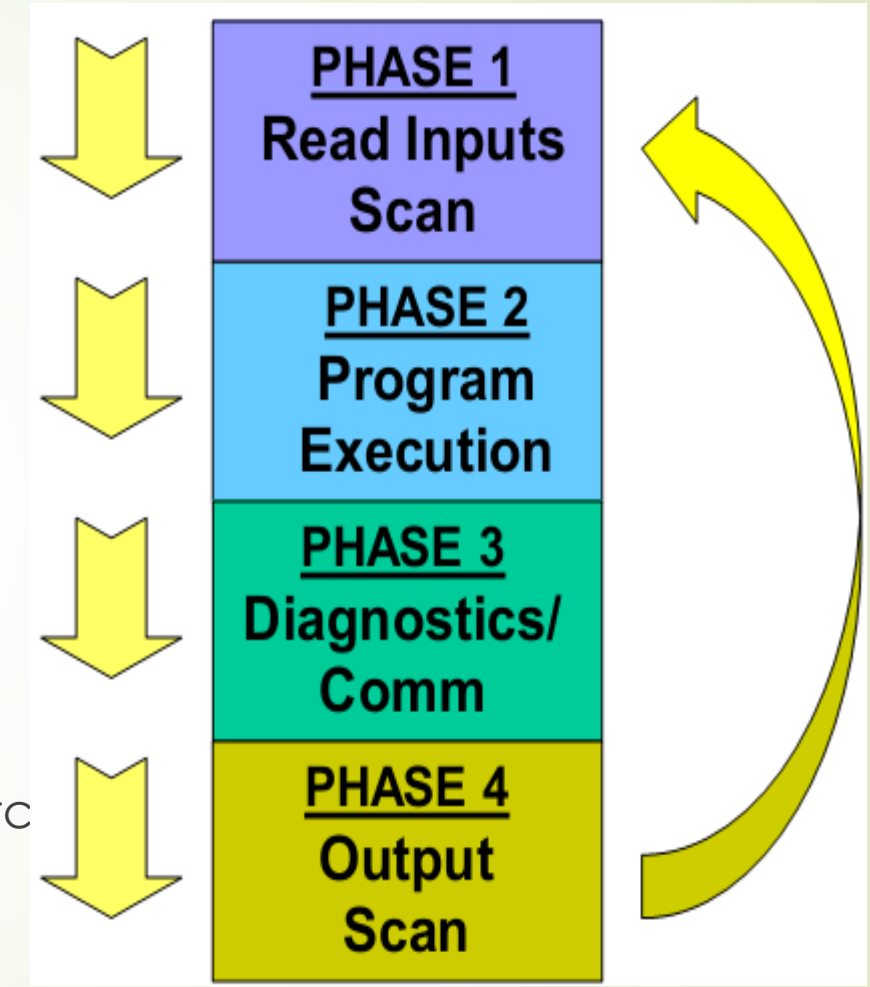
- PLC is a digital computer designed for multiple inputs and output arrangements, extended temperature ranges, immunity to electrical noise, and resistance to vibration and impact. A PLC is an example of a real time system.

- The PLC has input lines where sensors are connected to notify upon events (e.g. temperature above/below a certain level, liquid level reached, etc.), and output lines to signal any reaction to the incoming events (e.g. start an engine, open/close a valve, etc.).

- It uses a programmable memory to store the instructions and specific functions that include On/Off control, timing counting, sequencing, arithmetic and data handling.

PLC SCAN CYCLE & SCAN TIME

- While the PLC is running, the scanning process includes the four phases, which are repeated continuously as individual cycles of operation:



- Input Scan - Scan the state of the Inputs.
- Program Execution – Processes and executes the program logic.
- Housekeeping - This step includes communications, Internal Diagnostics, etc

.

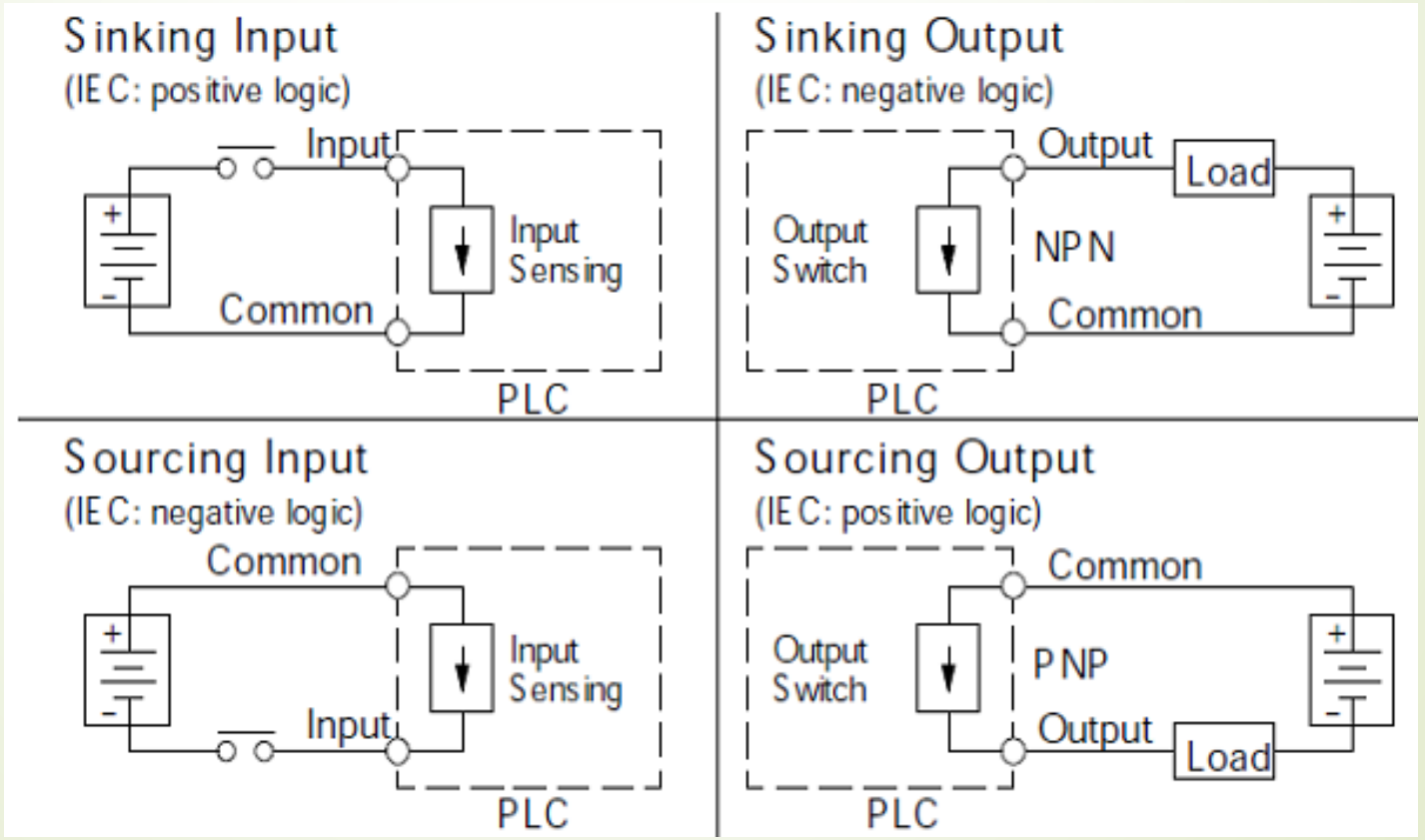
- Output Scan - Energize/de-energize the outputs

PLC SCAN CYCLE & SCAN TIME

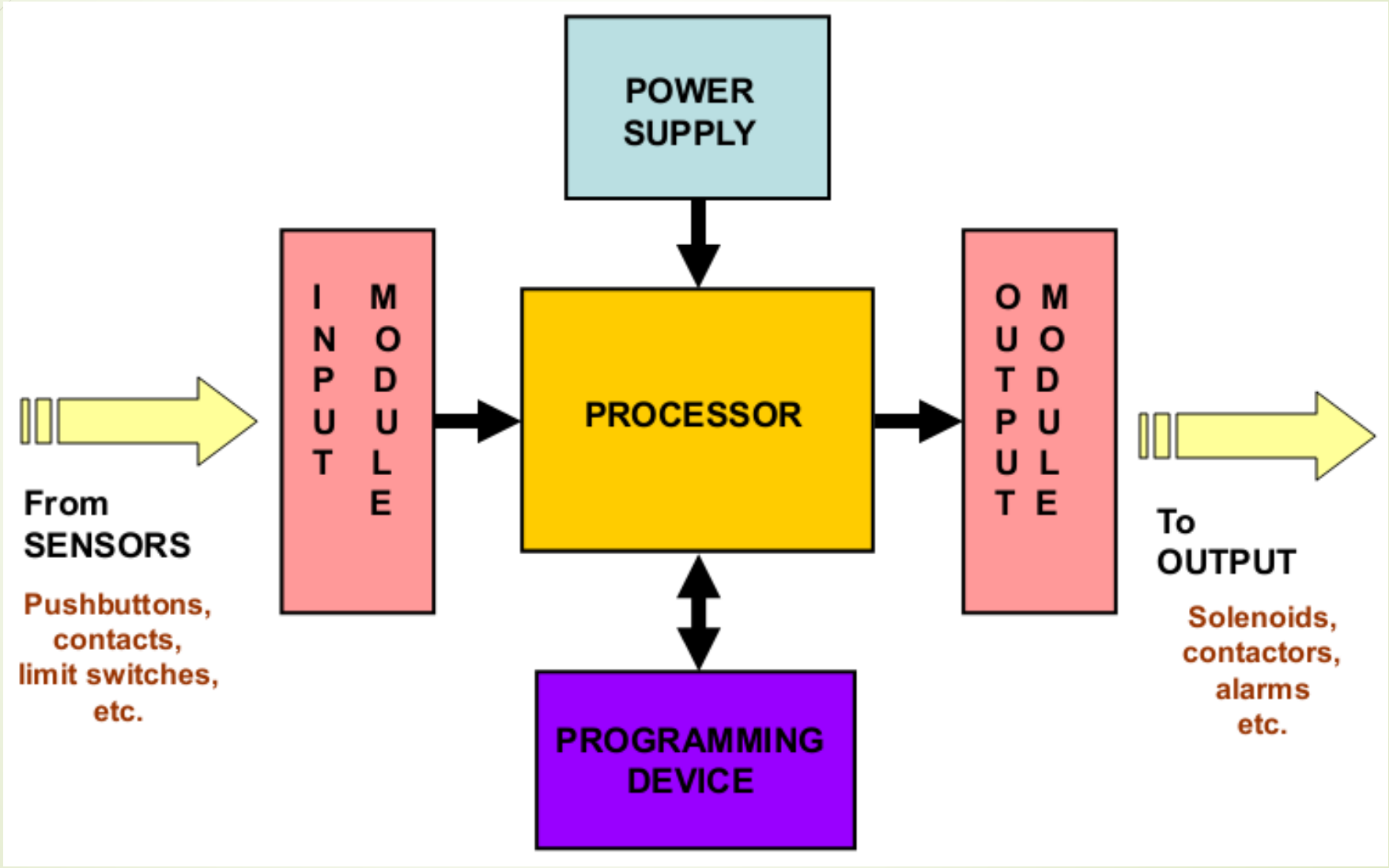
- The time it takes to implement a scan cycle is called SCAN TIME.
- The scan time composed of the program scan time, which is the time required for solving the control program, and the I/O update time, or time required to read inputs and update outputs

WORKING

- Source mode: +ve supply is given to the PLC.
- Sink mode: -ve supply is given to the PLC.
- The sink and source modes mainly define the way in which the I/O is connected to the I/O of the PLC.



MAJOR COMPONENTS OF A COMMON PLC



Major Components Of A Common PLC

1. INPUT MODULES accepts and converts signals from sensors into a logic signal. Ex. : Switches, Pushbuttons.

2. OUTPUT MODULES that convert control instructions a signal that can be used by actuators. Ex. : lamps, alarm.

3. CENTRAL PROCESSING UNIT(CPU)

- It is the brain of PLC and governs the activities of the entire PLC systems
- The CPU performs the SCAN CYCLE for PLC.
- The CPU consists of following blocks : Arithmetic Logic Unit (ALU), Internal memory of CPU, Internal timers, counters and Flags .

a) Timer is basically used add some delay in the programming circuit by adding a particular time to each circuit.

- Types of timers : - ON delay timer, OFF delay timer and Retentive Timers.

b) Counter works on the method of counting the number of pulse.

- Types of Counters: UP counter, DOWN counter

4. MEMORY is the component that stores information, programs and data in a PLC.

- Types of memories used in PLCs are read only memory (ROM) and random access memory (RAM).

5. POWER SUPPLY : Provides the voltage needed to run the primary PLC components.

6. PROGRAMMING DEVICE: The programming terminal is used for programming the PLC and monitoring/sequencing PLCs operation.

Distinct Features of PLC

Fig: - 1 MicroLogix 1000 PLC

Fig: - 2 Siemens S7-1200 PLC

- Power Supply : - DC - 24 V
  AC - 110 V
  220 V
- Inputs : -
  - Analog :
    a. Pressure
    b. Temperature
    c. Viscosity
    d. Level
    e. Flow
    f. Density
  - Discrete :
    a. Push button
    b. Toggle switch
    c. Relay
    d. Proximity sensor
- Outputs : -
  - Analog :
    a. Motor's parameters
    b. Conveyor's speed
    c. Control valve
  - Discrete :
    a. LEDs
    b. Lights

PLC Advantages And Disadvantages

ADVANTAGES

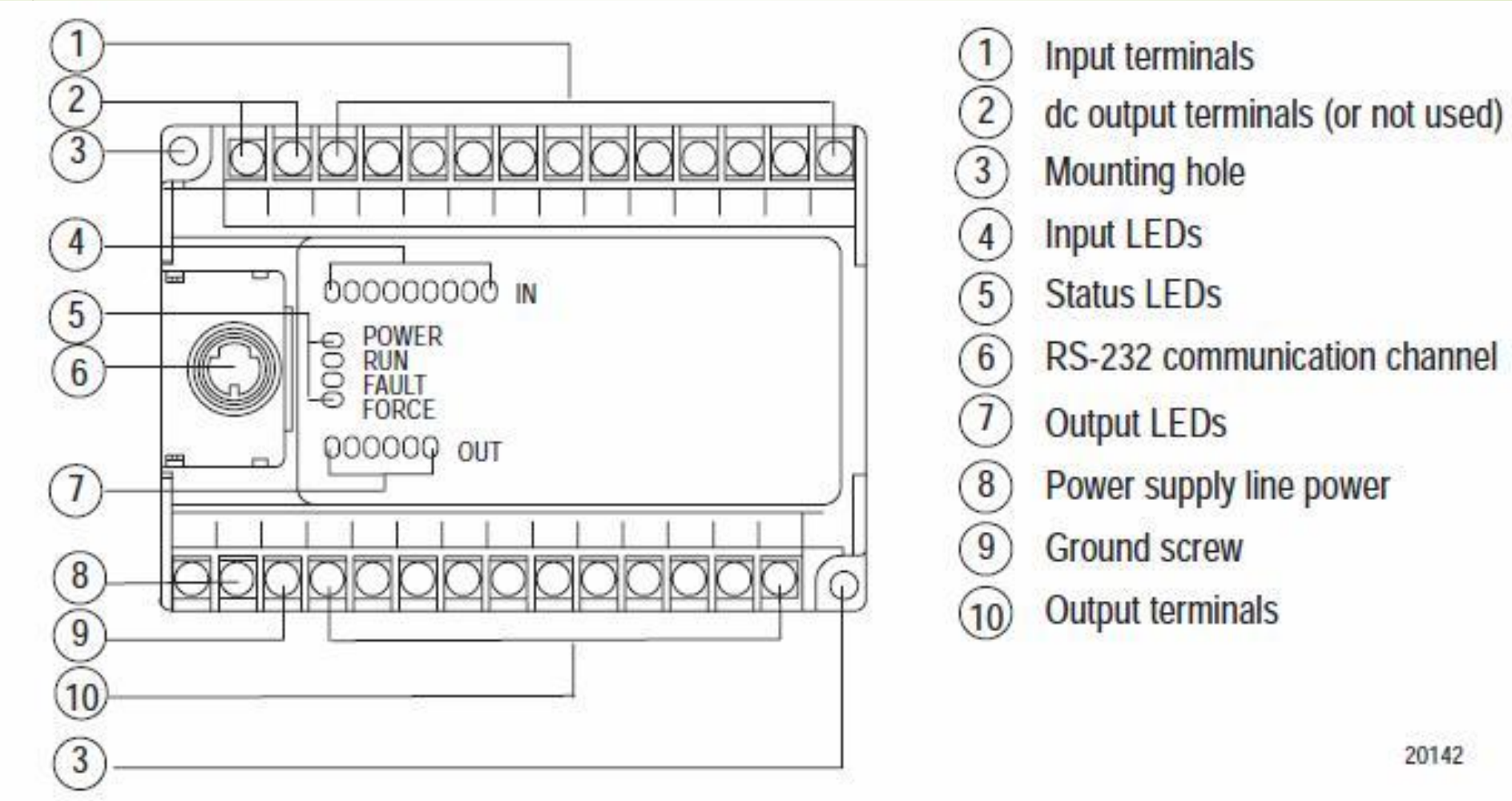
- Increased Reliability, flexibility and accuracy.
- Easier to troubleshoot.
- Remote control capability.
- Communication Capability.
- Handles much more complicated systems.

DISADVANTAGES

- Initial Costs Are High
- Skilled technicians required.
- There's Too Much Work Required In Connecting Wires.
- Error detection is challenging.

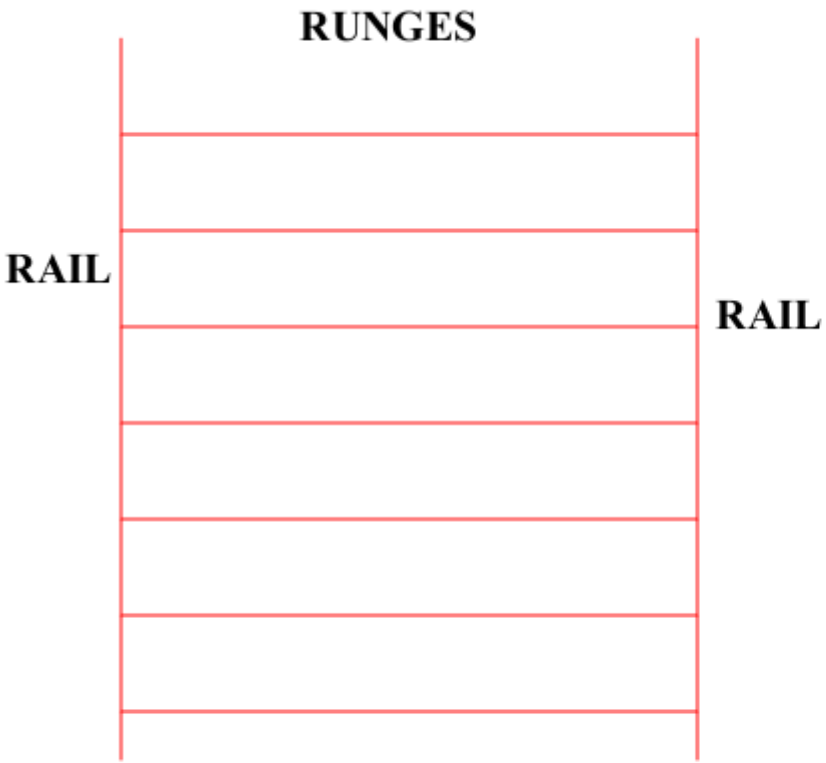
Micrologix 1000 Programmable Controller

- The hardware features of controller are :



PLC PROGRAMMING

ELEMENTS OF LADDER LOGIC

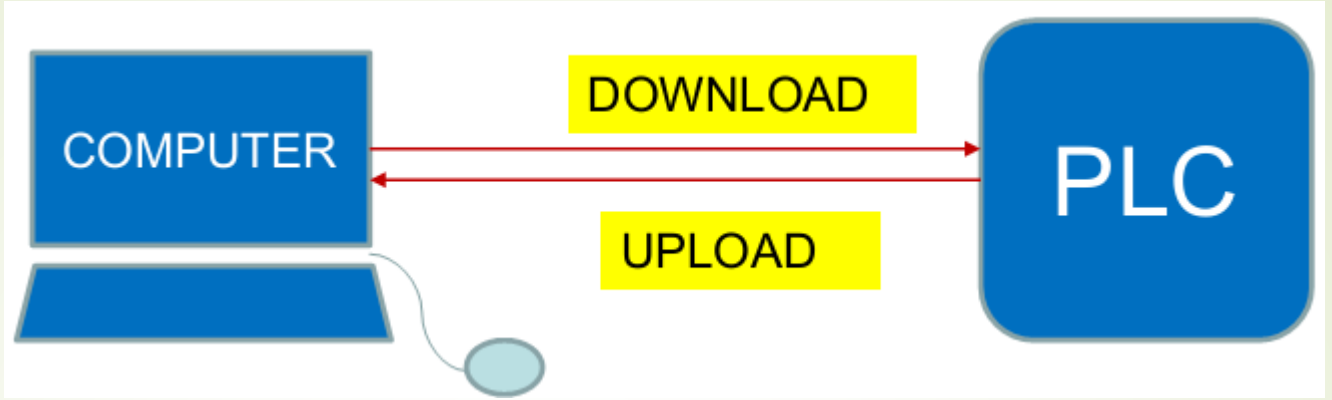


NORMALLY OPEN
(CONTACT)

NORMALLY CLOSED
(CONTACT)

COILS

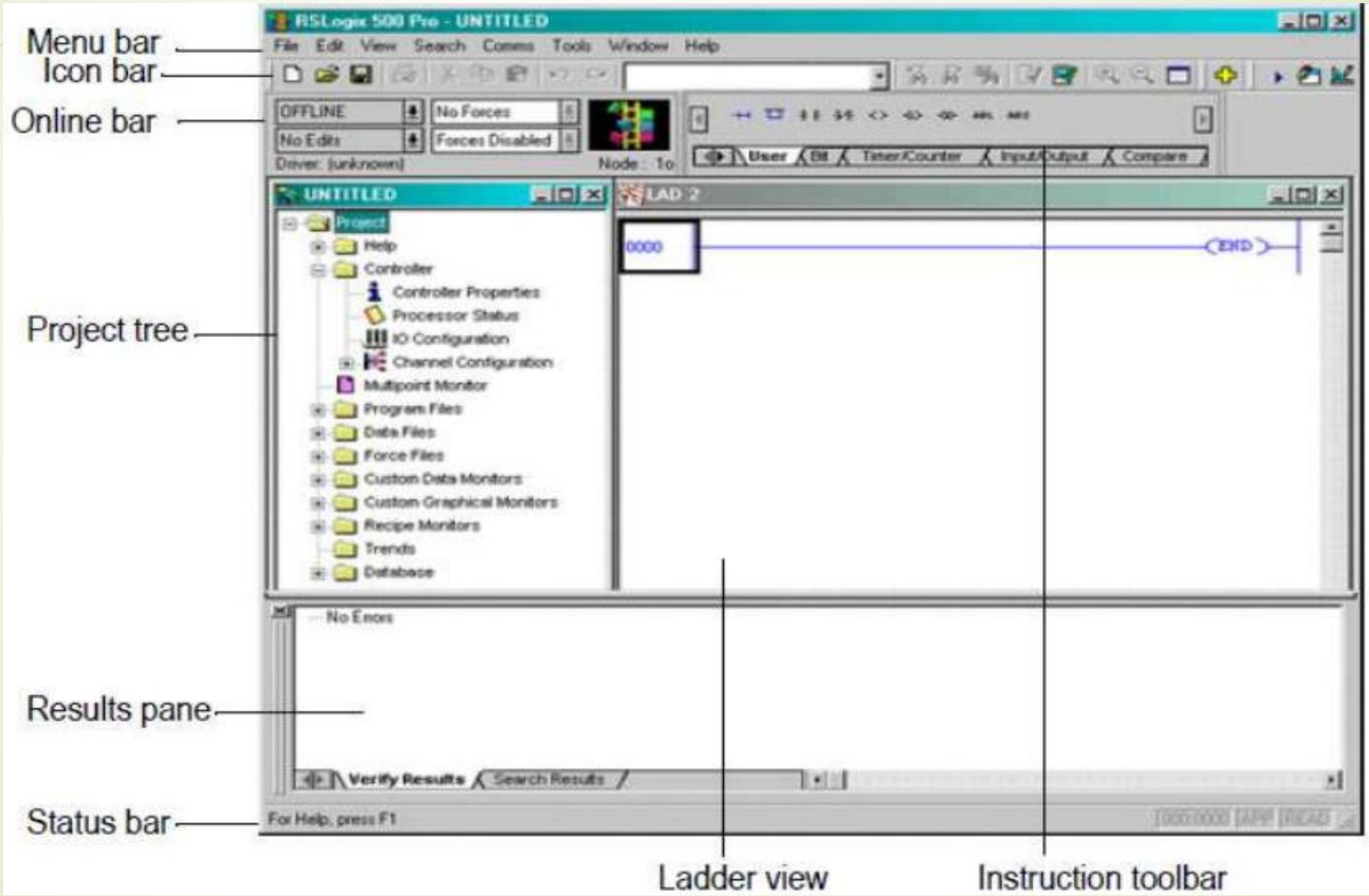
Ladder logic is a programming language used to develop software for PLC used in industrial control applications.



- DATA TRANSMISSION-SERIAL
- Baud Rate- 9.6 Kbps to 19.2 Kbps
- Driver softwares- RS Linx Classic, RS 232 & DF Drivers
- Programming software- RS Logix 500
- SYSTEM(PORT) PLC(PORT)

  RS 232 DF port

RS LOGIX 500 ENGLISH



BASIC SYMBOLS:

- Input symbols:

XIC- 

XIC- Examine If Closed

XIO- 

XIO- Examine if Open

- Output symbols:

OTE- 

OTE-Output Energize

OTL- 

OTL- Output Latch

OTU- 

OTU- Output Unlatch

- **XIC -** The XIC, also known as Examine If Closed, instruction is one of the fundamental instructions used in ladder logic programming for Programmable Logic Controllers (PLCs). This instruction is always found on the left side of a ladder rung and will verify if the specified bit is in a logic HIGH state. If that's the case, the instruction will evaluate to true and allow the rest of the rung to execute.
- **XIO -** The XIO, also known as Examine If Open, instruction perform the function opposite to the XIC (Examine if Closed). It's a fundamental instruction for working with Programmable Logic Controllers (PLCs). This instruction can be found on the left side of a ladder logic rung and will evaluate to true if the specified bit is set to a LOW state. If that's the case, the instruction will allow the rest of the rung to execute.
- **OTE -** The OTE, also known as Output Energize, instruction will energize a single bit of data if the input leading to it is true. It's a fundamental instruction used in Programmable Logic Controllers (PLCs). This instruction will be found on the right side within a ladder logic structure and turn a bit to a HIGH state if the preceding instructions evaluate to true. If the same instructions evaluate to false, the OTE instruction will set the specified bit to a LOW state.

- **OTL -** The OTL, also known as the Output Latch, instruction will force a single bit of logic into a high state if all the conditions leading to it are true. It's a basic instruction which is powerful but can cause programmers a lot of grief if used improperly or too frequently. This instruction is found on the right side of a ladder logic rung and will switch a bit to a HIGH state once it executes. Unlike the OTE Instruction, the OTL will never turn the bit LOW. In order to make that happen, you can leverage other instructions which accomplish exactly that. That being said, the most common pairing with an OTL is an OTU (Output Unlatch).
- **OUT -** The OTU, also known as Output Unlatch, instruction will set a bit to LOW if all the conditions leading to it evaluate to true. This instruction is one of the fundamental instructions for Programmable Logic Controllers (PLCs), but it should be used with caution. This instruction will be found on the right side within a ladder logic structure and turn a bit to a LOW state if the preceding instructions evaluate to true.

PROGRAMMING BASICS

- ADDRESSING FORMAT : Identifier: slot address(if any).word address/bit
- DATA FILES:
  - Micrologix 1000 contain the status information associated with the external I/O and all other instructions we use in our main and subroutine ladder program files.
  - These files are organized by the type of data they contain. Data file types are:

File type	Identifier	File number
Output	O	0
Input	I	1
Status	S	2
Bit	B	3
Timer	T	4
Counter	C	5
Control	R	6
Integer	N	7

PLC TIMER/COUNTER FUNCTION

Instruction		Purpose
Mnemonic	Name	
TON	Timer On-Delay	Count timebase intervals when the instruction is true.
TOF	Timer OFF-Delay	Count timebase intervals when the instruction is false.
RTO	Retentive Timer	Count timebase intervals when the instruction is true and retains the accumulated value when the instruction goes false or when power cycle occurs.
CTU	Count Up	Increments the accumulated value at each false to true transition and retains the accumulated value when the instruction goes false or when power cycle occurs.
CTD	Count Down	Decrements the accumulated value at each false to true transition and retains the accumulated value when the instruction goes false or when power cycle occurs.
RES	Reset	Resets the accumulated value and status bits of a timer or counter. Do not use with TOF timers.

MATH INSTRUCTIONS

Instruction		Purpose
Mnemonic	Name	
ADD	Add	Add source A to source B and store the result in the destination.
SUB	Substract	Substracts source B from source A and store the result in the destination.
MUL	Multiply	Multiply source A by source B and store the result in the destination.
DIV	Divide	Divide source A by source B and store the result in the destination and ther math register.
DDV	Double Divide	Divides the containts of the math register by the source and stores the result in the destination and the math register.
CLR	Clear	Sets all bits of a word to zero.
SQR	Square Root	Calculates the square root of the source and places the integer result in the destination.
SCL	Scale Data	Multilplies the source by the specified rate, adds to an offset value, and stores the result in the destination.

SCADA


SCADA
System Information
Vehicle Monitoring
J1939 Information
CAN Information
GPS
WiFi
Supervisory Control And Data Acquisition

SCADA

- SCADA is "Supervisory Control and Data Acquisition" – real time industrial process control systems used to centrally monitor and control remote or local industrial equipment such as motors, valves, pumps, relays, etc

- A SCADA system gathers information (such as where a leak on a pipeline has occurred), transfers the information back to a central site, then alerts the home station that a leak has occurred, carrying out necessary analysis and control, such as determining if the leak is critical, and displaying the information in a logical and organized fashion.

WHY SCADA?

- Saves Time and Money
  - Requires less time for acquiring data.
  - Reduces man-power needs
  - Increases production efficiency of a company
  - Cost effective for power systems
  - Saves energy
- Reliable
- Stores collected data on regular basis.
- Yields better visibility for distant machinery.
- Gives facility of historical analysis.
- Supervisory control over a particular system.

TAG'S

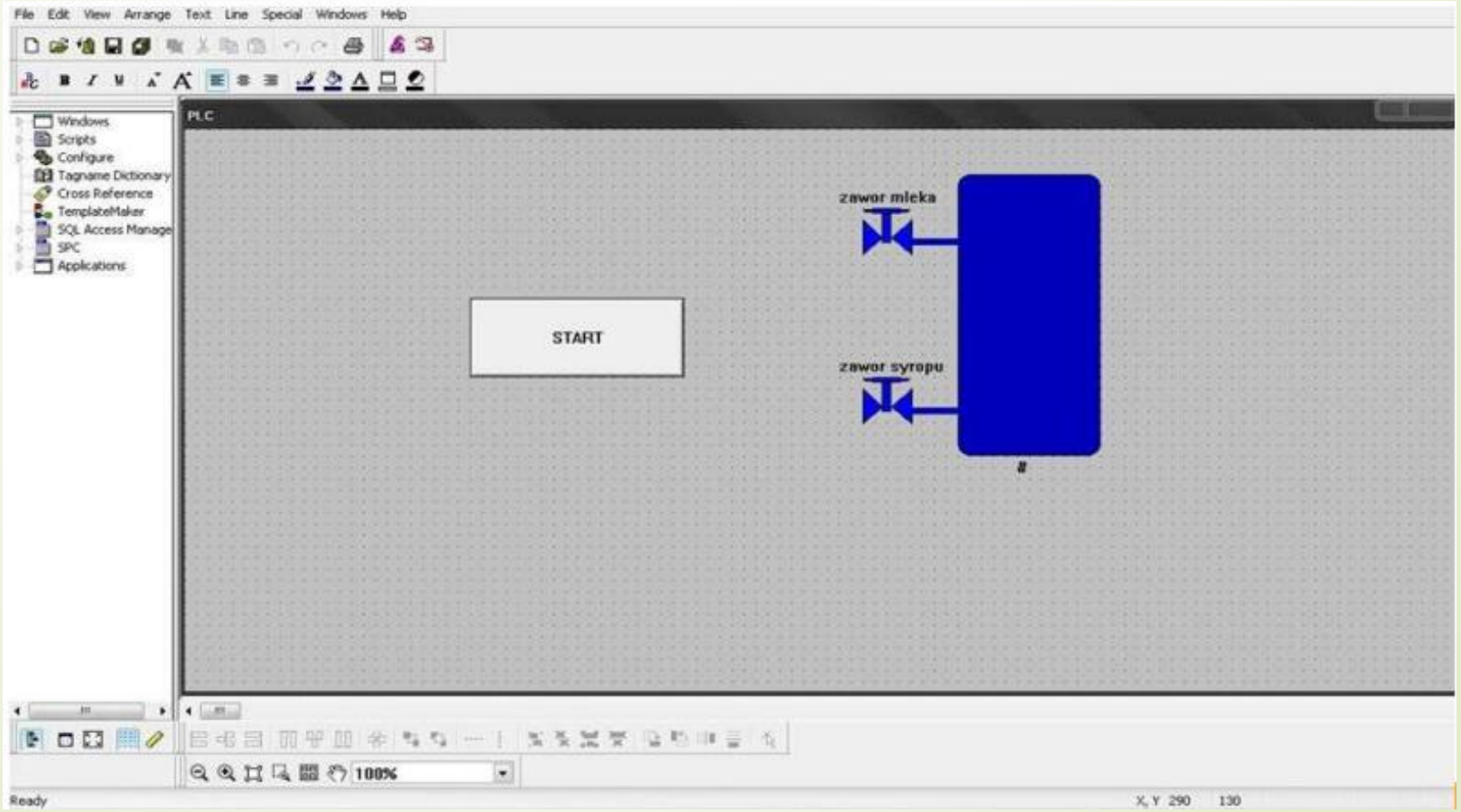
- It is the address of the memory location where signals are being saved.
- We define a TAG in order to use it in the SCADA software(Intouch wonderware).
- TYPES : 1) Analog 2) Digital 3) Strings
- These are further associated with either "Memory"or "I/O" type tags.
- PC-SCADA –Memory analog/digital/string tag.
- PLC-SCADA - I/O analog/digital/string tag

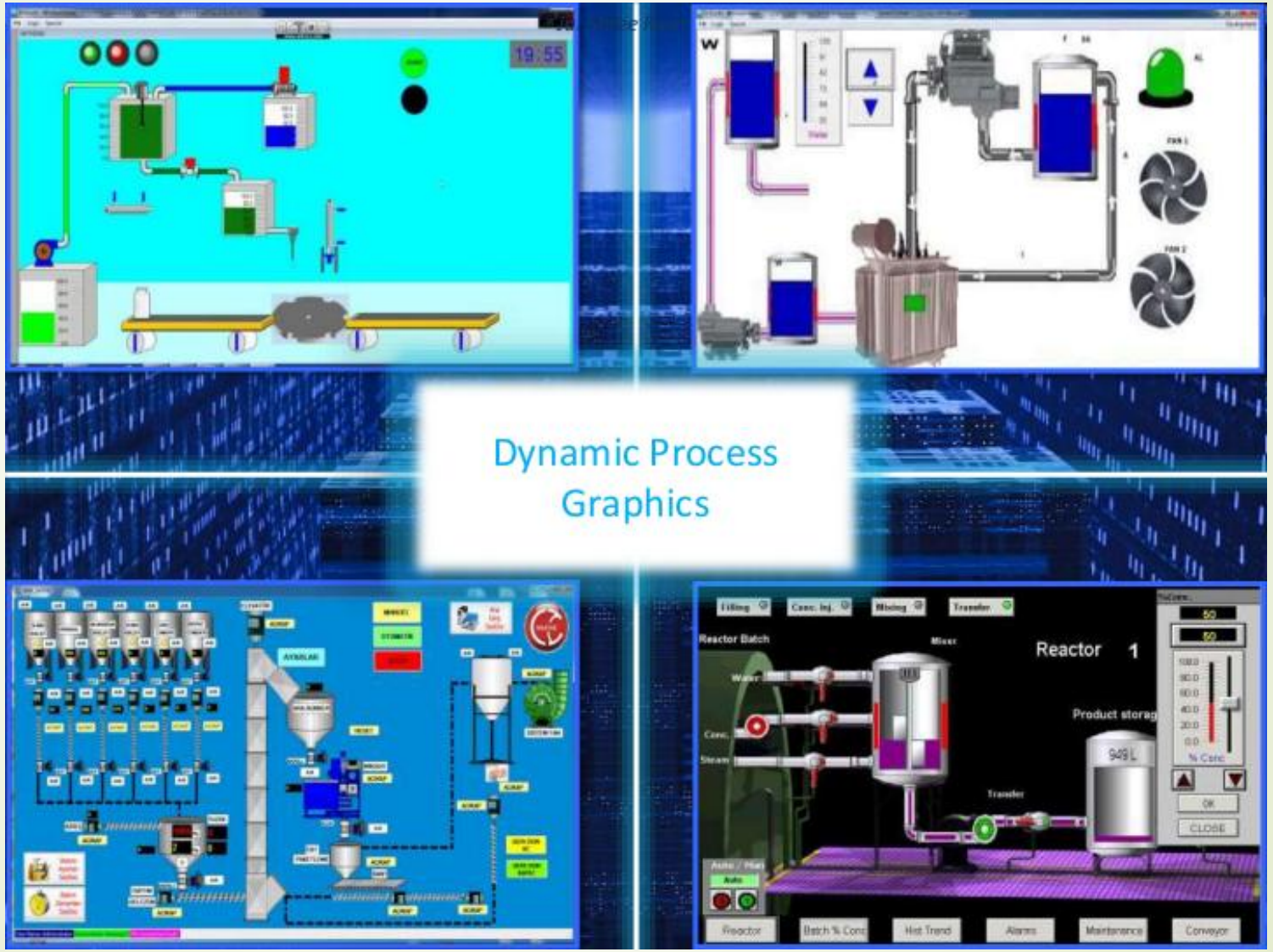
FEATURES OF SCADA

- Dynamic process graphic
- Real-time and Historical trending
- Alarms
- Recipe Management
- Security
- Device connectivity
- Script for logic development
- Database connectivity.

1. DYNAMIC PROCESS GRAPHICS

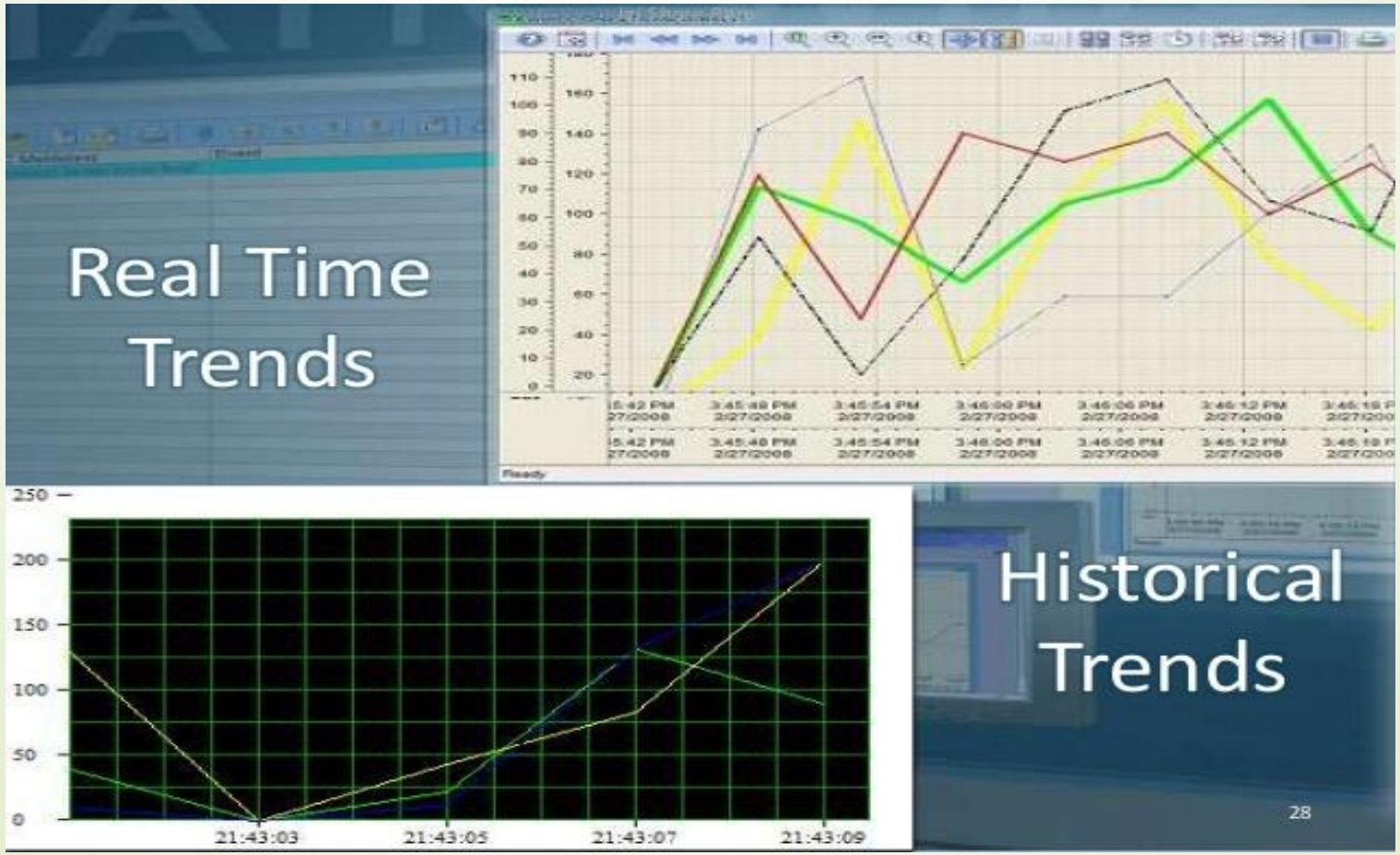
- Using this feature, one can develop graphics which can resemble the plant. The graphics can include reactors, valves, pumps, agitators, conveyors as well as other equipment and machinery used in the plant.
- The status of the equipment running/stopped can be shown using different colors/animation.




Dynamic Process
Graphics
Reactor Batch
Mixer
Reactor 1
Product storag
Reactor
Batch % Conc
Heat Trend
Alarms
Maintenance
Conveyor

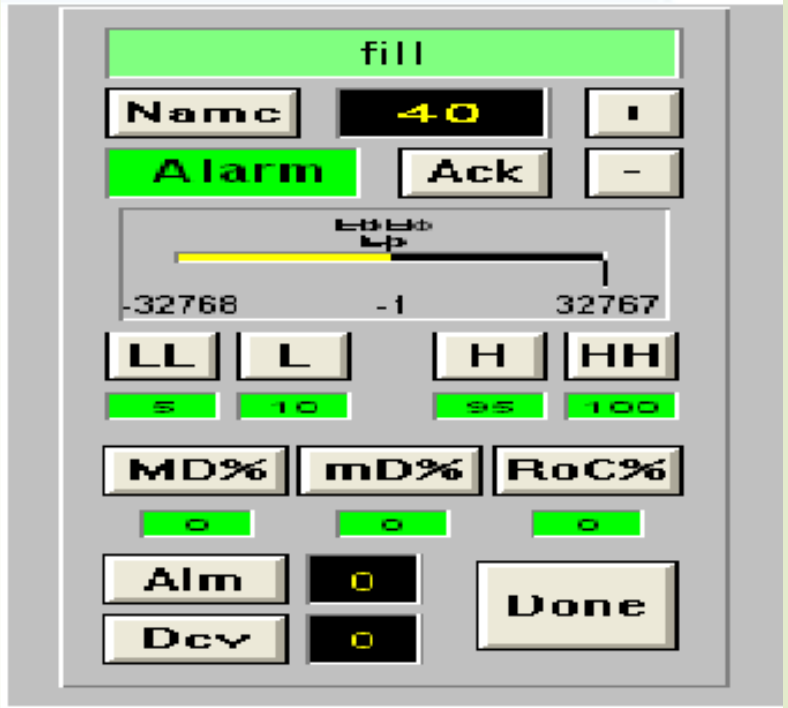
2. REAL-TIME & HISTORICAL TRENDS

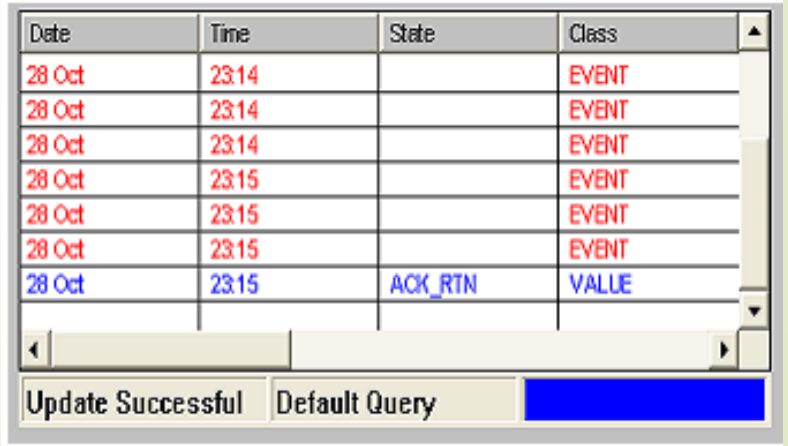
- This facility is used for representing the data in graphical form
- Real-time data will plot the real-time value for fixed period of time while historical data will have historical data stored value which can be viewed on demand.
- Depending upon the storing capacity of the hard-disk one can specify the no of days the data can be stored



3. ALARMS

- Every plant needs proper monitoring and control of the process parameters.
- Alarms represent warnings of process conditions that could cause problems, and require an operator response.
- Generally alarms are implemented by using the lamps or hooters in field but in SCADA it can be represented using animation.
- In many SCADA software, four type of alarm limits are used ie HI, HIHI, LOW, LOW LOW.

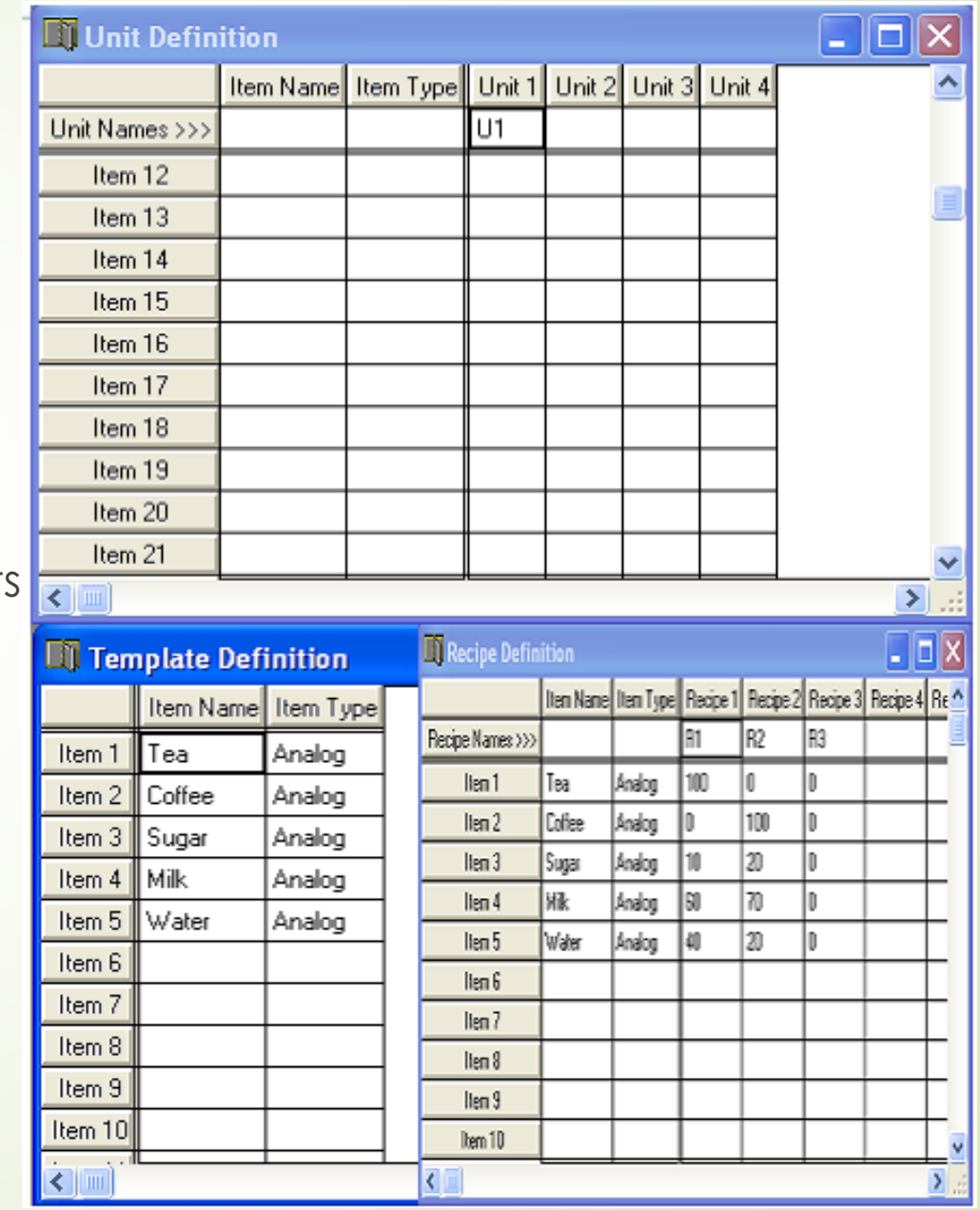




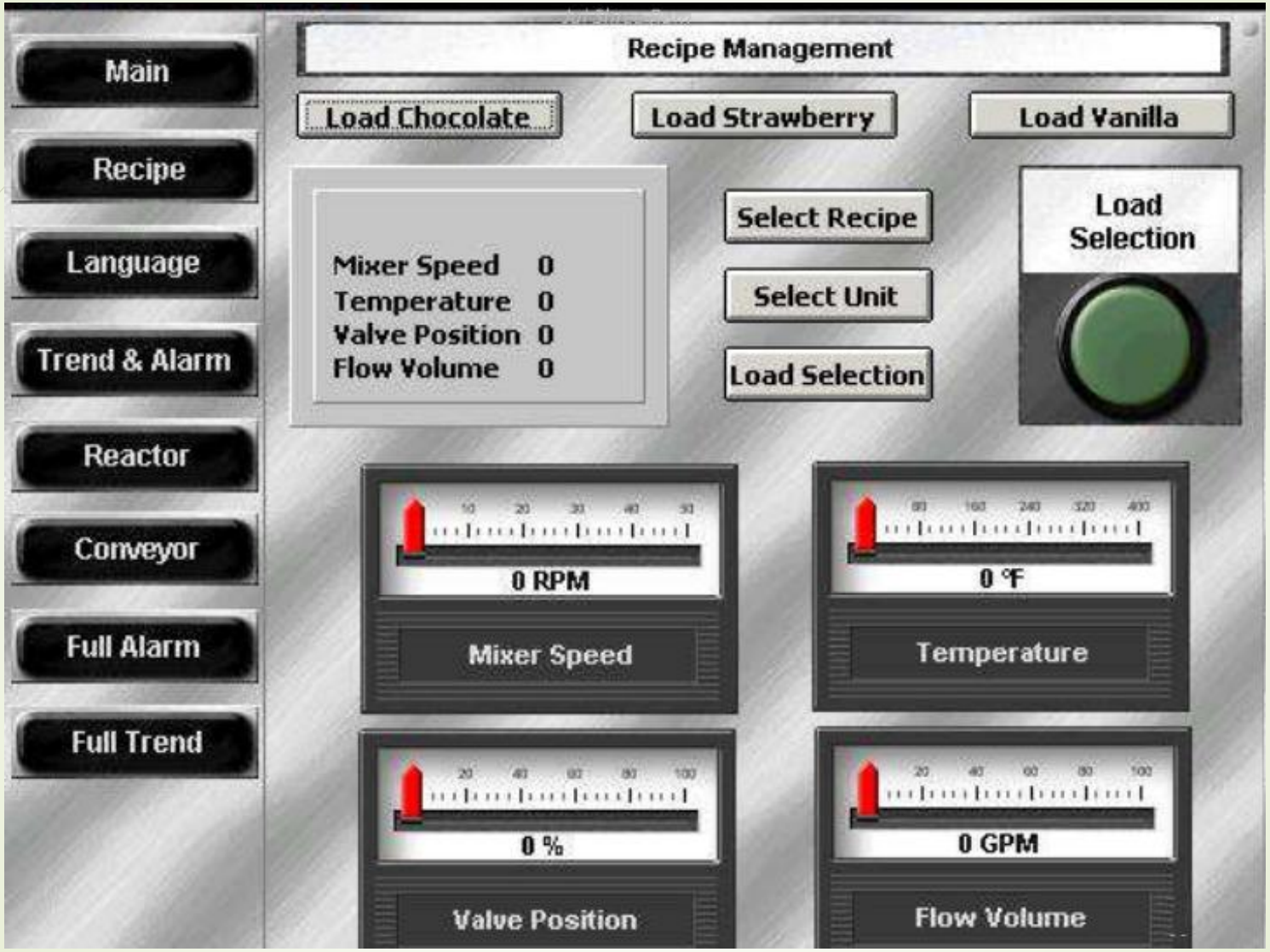

ALARMS

4. RECIPE MANAGEMENT

- In many cases we use the same plant for manufacturing different product range. For example an oil blending plant can manufacture power oil, transformer oil, automobile oil.
- The recipe management is facility used for maintain various recipes of different products and implement it on the process.
- The recipe can be stored in a single server and it can be fetched by any client server from any area to run the process

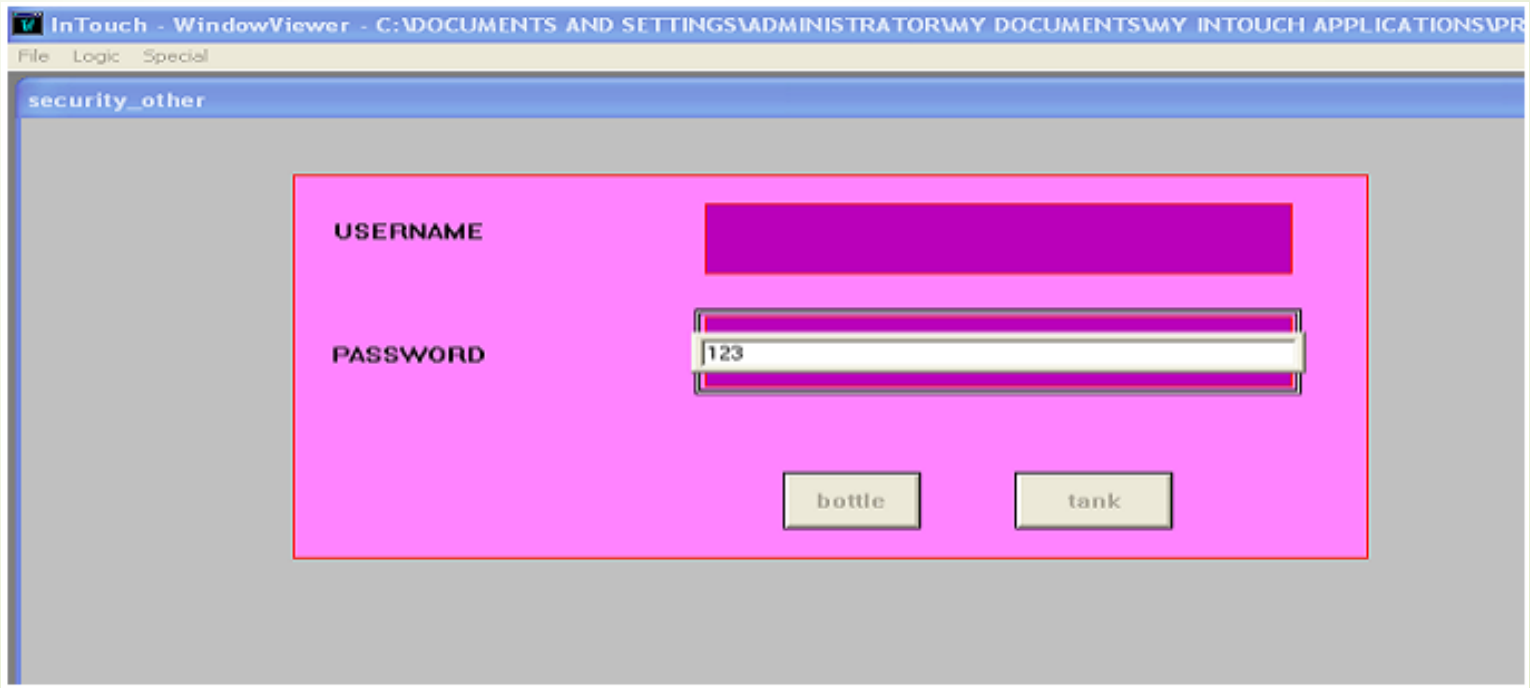



Recipe
Management
Cakes Finished: 0/6
Incorrect Cakes: 0/2


Main
Recipe
Language
Trend & Alarm
Reactor
Conveyor
Full Alarm
Full Trend
Recipe Management
Load Chocolate
Load Strawberry
Load Vanilla
Mixer Speed 0
Temperature 0
Valve Position 0
Flow Volume 0
Select Recipe
Select Unit
Load Selection
Load Selection
0 RPM
Mixer Speed
0 °F
Temperature
0 %
Valve Position
0 GPM
Flow Volume

5. SECURITY

- Every SCADA software has various levels of security for securing the application by avoiding unauthorized access.
- Depending upon the access level given the operator/engineers is allowed to do the task. In most of the cases, operators are allowed only to operate the plant while maintenance engineers can do the application modifications.
- The security can be given for individual as well as for groups



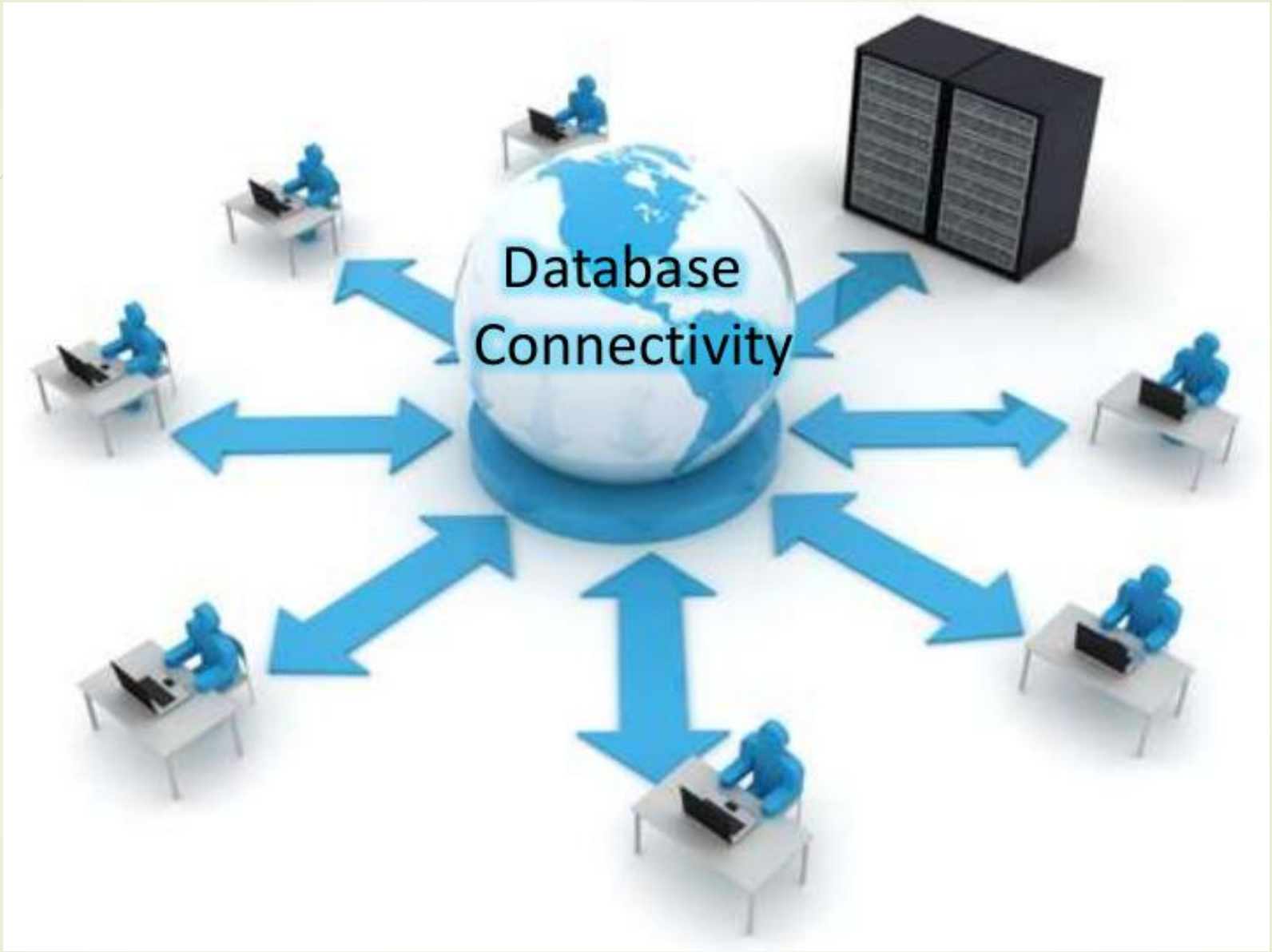

SECURITY
PASSWORD

6. DEVICE CONNECTIVITY

- Every control hardware has its own communication protocol for communicating with different hardware/software. Some of the leading communication protocol include Ethernet, DH485.
- The SCADA software needs device driver software for communication with PLC or other control hardware.
- More the driver software available better is the device connectivity. Most of the SCADA software used in the industry have connectivity with most of the leading control system.

7. DATABASE CONNECTIVIT

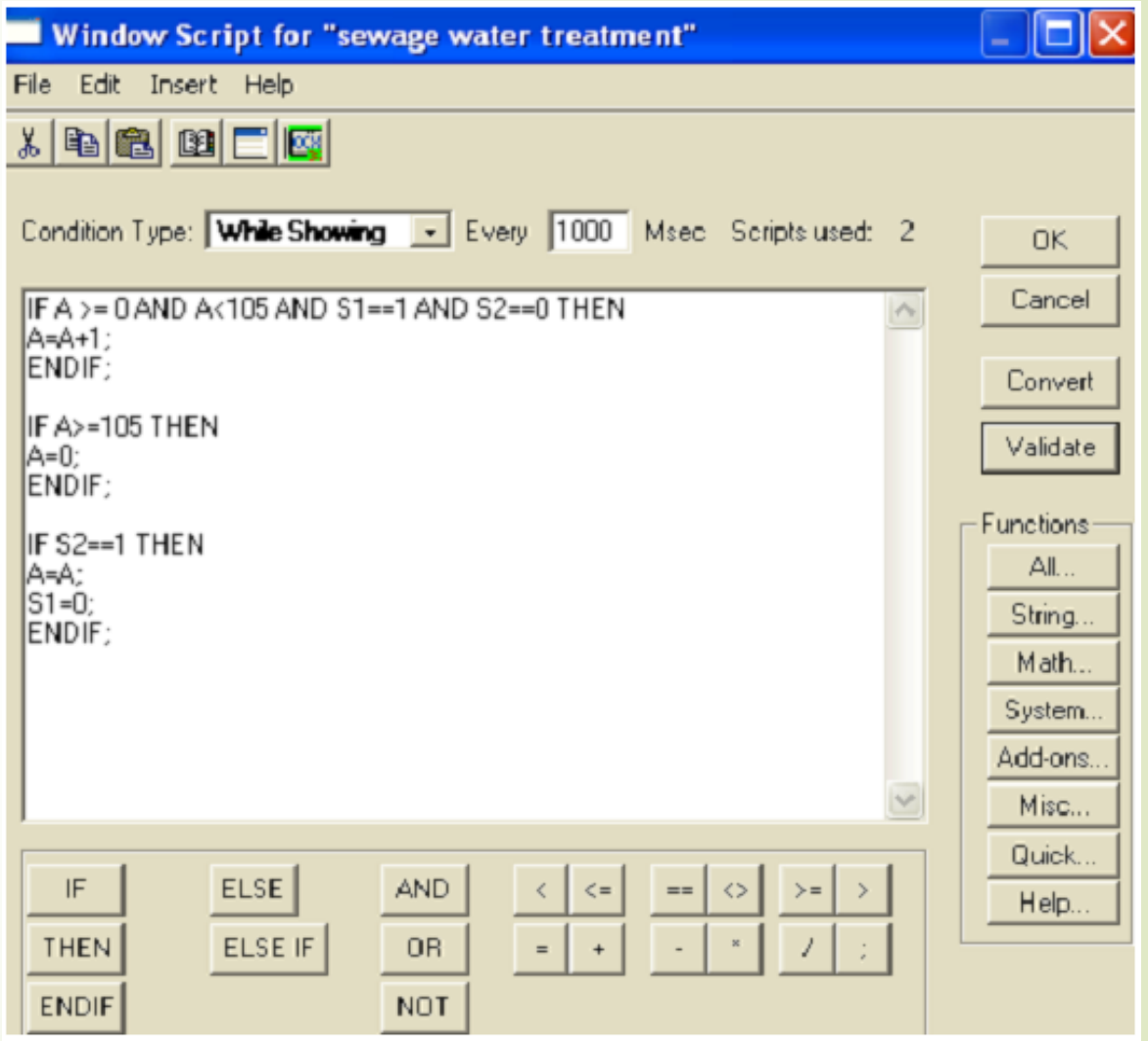
- In many plants, it is important to download the real-time information. In this case the database connectivity is must.
- Many SCADA software don't have their own database. Hence for storage and reporting they use third party database like MS Access or SQL.
- Wonderware InTouch provides a single integrated view of all your controls and information resources. Intouch enables engineers, supervisors, operators and managers to view and interact with the working of entire operation through graphical representations of their production processes.


Database
Connectivity

8. SCRIPTS

- Script Script is a way of writing logic in SCADA software. Every SCADA software has its own instructions and way of writing program.

- Using scripts, one can develop complex applications.




SILO
PANEL DE CONTROL
START
STOP
START
STOP
EMBAZADO
BANDA
LLENADO
PILOTOS
RUN
FULL
FILL
PARO
SCRIPT

Applications of SCADA

- Electric Power G.T.D.
- Water Supply Management
- Mining
- Irrigation
- Manufacturing Industries
- Traffic Signal Control
- Regional Electricity Monitoring

Advantages of Automation

- Human efforts
- Less manufacturing time
- Quality production
- Accuracy increases
- Simplified complex tasks
- Rise in efficiency
- Optimized output
- Demand of skilled labour
- Reliable processing

Disadvantages of Automation

- Dependency on machines.
- Increased development cost.
- Unemployment of unskilled labour.
- Rise in power demand.
- Requirement of skilled labour.

Applications of Automation

- Food industry
- Mining industry
- Manufacturing
- Washing stations
- Waste management
- Home automation