

ANALYSING AN INDUSTRIAL AUTOMATION PYRAMID AND PROVIDING SERVICE ORIENTED ARCHITECTURE

Shivani Kumra^{#1}, Laxman sharma^{#2}, Yashvi Khanna^{#3}, Abhishek Chattri^{#4}

Abstract— Industrial automation platforms are experiencing a paradigm shift. New technologies are making their way in the area, including embedded real-time systems, standard local area networks like Ethernet, Wi-Fi and ZigBee, IP-based communication protocols, standard Service Oriented Architectures (SOAs) and Web Services. The introduction of these new technologies raises important problems that need to be properly solved, one of these being the need to support real-time and quality-of-service (caps) for real-time applications. This paper describes a SOA enhanced with real-time capabilities for Industrial Automation. The proposed architecture allows for negotiation of the caps requested by this paper describe the different stages of automation from field devices sensors to remote operator desk and also describe the architecture of a system .

Keywords— Industrial Pyramid, Soa , Qos , Sensors , Controller

I. INTRODUCTION

In present scenario automation industries is experiencing a paradigm shift. But, some technological barriers are preventing the advancement of such technologies in current industrial practice. A critical bottleneck in process efficiency and flexibility of current systems is represented by the networking infrastructure. Today, many standardized protocols like Modbus, Profibus, Ethernet variants are making inroads. In real-time visibility of the different assets of a company (equipment, inventory, machines, etc.), there is still a huge gap between the physical world and the high-level monitoring systems. The major functional areas encountered in many process control applications are as follows: data acquisition and conditioning, direct digital control, supervisory control and sequence control. Typical control system architecture for the industrial process uses separate buses for network and data acquisition. We use a combination of Switched Ethernet and Foundation Field bus as network standards. This paper overcome with the problem of remote monitoring of plant process. This paper discuss various field signal sensing device, programmable logic controller, scanner, different communication protocols, communication system between field device & high level monitoring & control software (SCADA)

II. INDUSTRIAL AUTOMATION PYRAMID

An Industrial Automation System consists of numerous elements that perform a variety of functions related to Instrumentation, Control, Supervision and Operations Management related to the industrial process. Below, we classify the major functional elements typically found in industrial automation systems and also describe the nature of technologies that are employed to realize the functions.

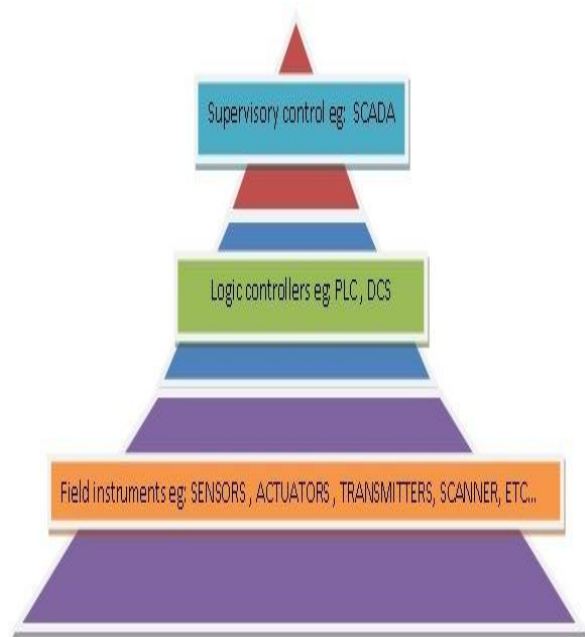


Fig.1.industrial automation system

Industrial automation systems are very complex having large number of devices with confluence of technologies working in synchronization Industrial automation systems are organized hierarchically as shown in the following figure 1

III. SENSORS

- A) **Proximity sensor:** Sensors are devices used to provide information on the presence or absence of an object. Sensors include limit switches, photoelectric, inductive, capacitive, and ultrasonic sensors. These products are

packaged in various configurations to meet virtually any requirement found in commercial and industrial applications.

Proximity Sensor	Advantages	Disadvantages	Applications
Limit Switch	High current capability Low cost Familiar "low-tech" sensing	Requires physical contact with target Very slow response contact bounce	Interlocking Basic end-of-travel sensing
Photoelectric	Senses all kinds of materials Longest Sensing range Very fast response time	Lens subject to contamination Sensing range affected by color and reflectivity of target	Packaging Material handling Parts detection
Inductive	Resistant to harsh environments Very predictable Long life Easy to install	Distance limitations	Industrial and Machines Machine tool Senses metal-only targets
Capacitive	Detects through some containers Can detect non-metallic targets	Very sensitive to extreme environmental changes	Level sensing
Ultrasonic	Senses all Materials	Resolution Repeatability Sensitive to temperature changes	Anti-collision Doors Web brake Level control

B) Pressure sensor: The object of pressure sensing is to produce a dial indication, control operation or a standard (4 - 20 mA) electronic signal that represents the pressure in a process. The most common pressure sensors or primary pressure elements are described below.

C) Bourdon tubes: Bourdon tubes are circular-shaped tubes with oval cross sections. The pressure of the medium acts on the inside of the tube. The outward pressure on the oval cross section forces it to become rounded. Because of the curvature of the tube ring, the bourdon tube then bends.

D) Bellows: Bellows type elements are constructed of tubular membranes that are convoluted around the circumference. The membrane is attached at one end to the source and at the other end to an indicating device or instrument. The bellows element can provide a long range of motion.

E) Diaphragms: A diaphragm is a circular-shaped convoluted membrane that is attached to the pressure fixture around the circumference. The pressure medium is on one side and the indication medium is on the other.

F) Differential pressure transmitters: A differential pressure capsule is mounted inside housing. One end of a force bar is connected to the capsule assembly so that the motion of the capsule can be transmitted to outside the housing. A sealing mechanism is used where the force bar penetrates the housing and also acts as the pivot point for the force bar. Provision is made in the housing for high-pressure fluid to be applied on one side of the capsule and low-pressure fluid on the other. Any difference in pressure will cause the capsule to deflect and create motion in the force bar.

G) Temperature sensor: Almost all industrial processes need accurately controlled temperatures. Physical parameters and chemical reactions are temperature dependent, and therefore temperature control is of major importance. Temperature is without doubt the most measured variable, and for accurate temperature control its precise measurement is required. The major types used are

- 1) Resistance temperature detector (RTD)
- 2) Thermocouples
- 3) Pyrometers

H) Flow measurement

1) Vortex flow meters: The vortex flow meter is also a velocity-type flow meter that places a flow element in the flow stream that is not streamline. When the flow stream strikes the flow element, a series of vortices is produced.

Hence this type of flow meter is sometimes called a vortex shedding flow meter.

- 2) **Mass flow meters:** These meters all had moving parts and complex mechanical designs. First developed for the measurement of aircraft fuel, some are still in use. However, because of their complex nature and high maintenance costs, they are gradually being replaced by more robust and less maintenance-demanding designs.

E) Electromagnetic flow meter : The most common flow meter apart from mechanical flow meters is the magnetic, commonly referred to as a "mag meter" or an "electromagnetic". A magnetic field is applied to the metering tube, which results in a potential difference proportional to the flow velocity perpendicular to the flux lines. The physical principle at work is Faraday's of electromagnetic induction.

IV. PROGRAMMABLE LOGIC CONTROLLERS

A Programmable controller is a solid state user programmable control system with functions to control logic, sequencing, timing, arithmetic data manipulation and counting capabilities. It can be viewed as an industrial computer that has a central processor unit, memory, input output interface and a programming device. The central processing unit provides the intelligence of the controller. It accepts data, status information from various sensing devices like limit switches, proximity switches, executes the user control program stored in the memory and gives appropriate output commands to devices such as solenoid valves, switches etc.

Input output interface is the communication link between field devices and the controllers. Through these interfaces the processor can sense and measure physical quantities regarding a machine or process, such as, proximity, position, motion, level, temperature, pressure, etc. Based on status sensed, the CPU issues command to output devices such as valves, motors, alarms, etc. The programmer unit provides the man machine interface. It is used to enter the application program, which often uses a simple user-friendly logic.

We can actually consider the PLC to be a box full of hundreds or thousands of separate relays, counters, timers and data storage locations. They don't physically exist but rather they are simulated and can be considered software counters, timers, etc. Each component of a PLC has a specific function

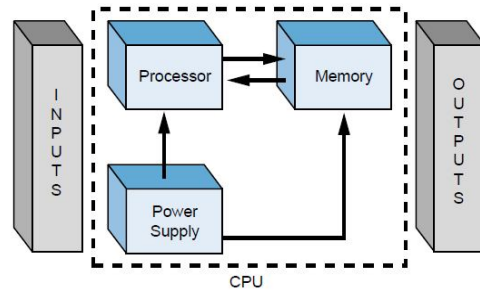


fig 2. PLC components

- A) **CPU:** The term CPU is often used interchangeably with the word processor; however, the CPU encompasses all of the necessary elements that form the intelligence of the system the processor plus the memory system and power supply.
- B) **The Processor:** The processor of a PLC holds and executes the user program. In order to carry out this job, the processor must store the most up-to-date input and output conditions.
- C) **Input image table:** The input conditions are stored in the input image table, which is a portion of the processor's memory. That is, every single input module in the I/O section has assigned to it a particular location within the input image table. That particular location is dedicated solely to the task of keeping track of the latest condition of its input terminal.
- D) **Output image table:** The output conditions are stored in the output image table, which is another portion of the processor's memory. The output image table bears the same relation to the output interface of the I/O section that while terminals are analog inputs. You can directly connect any analog input to the processor via these terminals. A modern PLC output module is optically isolated, and uses a triac, power transistor or relay as the series connected load controlling device. Terminal 1 to 8 are these type of O/P terminals whereas terminal D/A is Analog output terminal from processor. Each output device is wired to a particular output terminal on the I/O interface.
- E) **User program memory:** A particular portion of the processor's memory is used for storing the user program instructions. We will use the name user program memory to refer to this processor subsection.

Before a PLC can begin controlling an industrial system, a human user must enter the coded instructions that make up the user program. This procedure called programming the PLC.

As the user enters instructions, they are automatically stored at sequential locations within the user program memory. This

sequential placement of program instructions is self-regulated by the PLC, with no discretion needed by the human user.

The total number of instructions in the user program can range from a half dozen or so, for controlling a simple machine, to several thousand, for controlling a complex machine or process.

After the programming procedure is complete, the human user manually switches the PLC out to PROGRAM mode into RUN mode, which causes the CPU to start executing the program from beginning to end repeatedly.

- **Input/output Section:** This part of system is on separate board connected to processor via cable. It allows the processor to communicate with the outside world. It is also called Data Acquisition System (DAS). This part of system provides
 - digital inputs
 - digital outputs.
 - analog inputs

F) **PLC Operation:** A PLC works by continually scanning a program. We can think of this scan cycle as consisting of 3 important steps.

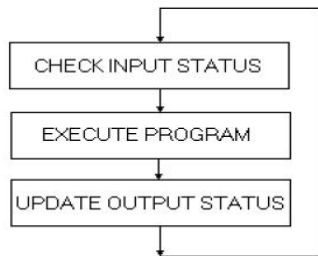


fig 3 PLC operation cycle

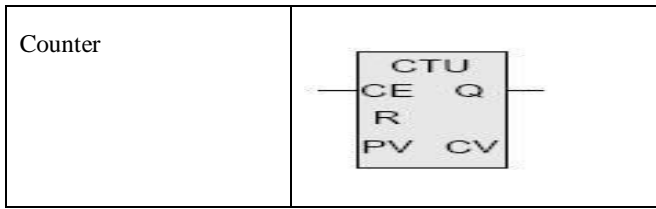
The first step is to check the input status. This step is therefore generally referred to as the “Check Input Status” stage. First the PLC takes a look at each input to determine if it is on or off. In other words, is the sensor connected to the first input on? How about the second input? How about the third? This goes on and on through the entire program. It records this data into its memory to be used during the next step. Next the PLC executes your program one instruction at a time, called the “Execute Program” stage. For example, if your program said that if the first input was on then it should turn on the first output. Since it already knows which inputs are on/off from the previous step it will be able to decide whether the first output should be turned on based on the state of the first input. It will store the execution results for use later during the next step.

Finally the PLC updates the status of the outputs. It updates the outputs based on which inputs were on during the first step and the results of executing your program during the second step. Based on the example in step 2 it would now turn on the first output because the first input was on and your program said to turn on the first output

G) **PLC programming:** Ladder logic or ladder diagrams are the most common programming language used to program a PLC. Ladder logic was one of the first programming approaches used in PLCs because it borrowed heavily from the relay diagrams that plant electricians already knew. The symbols used in relay ladder logic consist of a power rail to the left, a second power rail to the right, and individual circuits that connect the left power rail to the right.

Basic symbols such as a normally open contact, normally closed contact, normally open coil, normally closed coil, timer and counter most logical conditions can be represented.

LOGIC	SYMBOLS
Normally Open Contact	
Normally Open Coil	
Normally Closed Contact	
Normally Closed Coil	
AND “gate”	
OR gate	
Timer	



V. SCADA

Supervisory Control and Data Acquisition. SCADA systems are used to monitor and control a plant or equipment in industries such as telecommunications, water and waste control, energy, oil and gas refining and transportation. These systems encompass the transfer of data between a SCADA central host computer and a number of Remote Terminal Units (RTUs) and/or Programmable Logic Controllers (PLCs), and the central host and the operator terminals.

SCADA systems consist of: One or more field data interface devices, usually RTUs, or PLCs, which interface to field sensing devices and local control switchboxes and valve actuators. A communications system used to transfer data between field data interface devices and control units and the computers in the SCADA central host. The system can be radio, telephone, cable, satellite, etc., or any combination of these. A central host computer server or servers (sometimes called a SCADA Center, master station, or Master Terminal Unit (MTU). A collection of standard and/or custom software [sometimes called Human Machine Interface (HMI) software or Man Machine Interface (MMI) software] systems used to provide the SCADA central host and operator terminal application, support the communications system, and monitor and control remotely located field data interface devices

A) Field Data Interface Devices

A PLC is a device used to automate monitoring and control of industrial facilities. It can be used as a stand-alone or in conjunction with a SCADA or other system. PLCs connect directly to field data interface devices and incorporate programmed intelligence in the form of logical procedures that will be executed in the event of certain field conditions. PLCs have their origins in the automation industry and therefore are often used in manufacturing and process plant applications. The need for PLCs to connect to communication channels was not great in these applications, as they often were only required to replace traditional relay logic systems or pneumatic controllers. SCADA systems, on the other hand, have origins in early telemetry applications, where it was only necessary to know basic information from a remote source.

VI. SCADA ARCHITECTURES

SCADA systems have evolved in parallel with the growth and sophistication of modern computing technology. The following sections will provide a description of the following three generations of SCADA systems:

- First Generation – Monolithic
- Second Generation – Distributed
- Third Generation – Networked

A) Monolithic SCADA Systems: When SCADA systems were first developed, the concept of computing in general centred on “mainframe” systems. Networks were generally non-existent, and each centralized system stood alone. As a result, SCADA systems were standalone systems with virtually no connectivity to other systems.

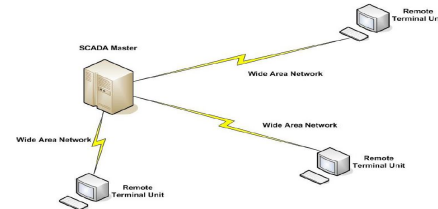


fig 4. First generation SCADA architecture

B) **Distributed SCADA Systems:** The next generation of SCADA systems took advantage of developments and improvement in system miniaturization and Local Area Networking (LAN) technology to distribute the processing across multiple systems. Multiple stations, each with a specific function, were connected to a LAN and shared information with each other in real-time. These stations were typically of the mini-computer class, smaller and less expensive than their first generation processors.

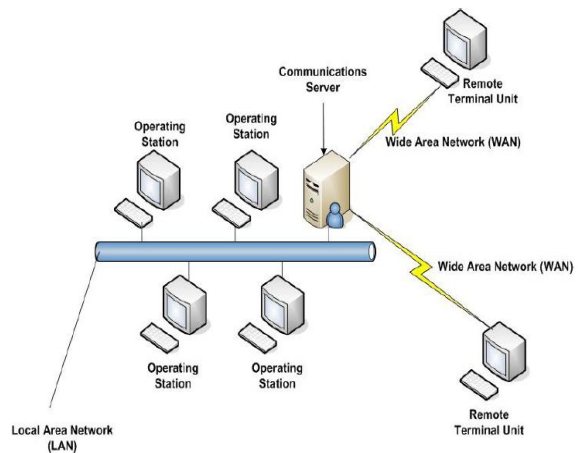


Fig 5. Second generation SCADA architecture

C) Networked SCADA Systems : The current generation of SCADA master station architecture is closely related to that of the second generation, with the primary difference being that of an open system

architecture rather than a vendor controlled, proprietary environment. There are still multiple networked systems, sharing master station functions. There are still RTUs utilizing protocols that are vendor-proprietary. The major improvement in the third generation is that of opening the system architecture, utilizing open standards and protocols and making it possible to distribute SCADA functionality across a WAN and not just a LAN.

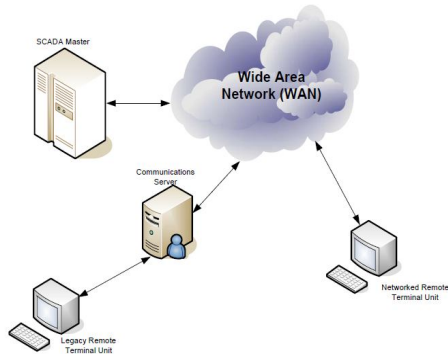


Fig 6 A networked SCADA system.

D) RS-485

RS485 or EIA (Electronic Industries Association) RS485 is a balanced line, half-duplex transmission system allowing transmission distances of up to 1.2 km. The following table summarizes the RS-485 Standard:

Parameter	
Mode of drivers	Differential
Maximum cable length	1200 meters
Number of drivers and receivers	32 driver and 32 receiver
Maximum common mode voltage	12 to -7 volts
Receiver sensitivity	+/-200mv

4.5.1 Connection diagram

A Belden 9841 (Single pair) or 9842 (Two pair) cable with a characteristic impedance of 120 ohms is recommended, the cable should be terminated at each end with a 120 ohm, quarter watt (or greater) resistor. There must be no more than two wires connected to each terminal, this ensures that a

“Daisy Chain or “straight line” configuration is used. A “Star” or a network with “Stubs (Tees)” is not recommended as reflections within the cable may result in data corruption.

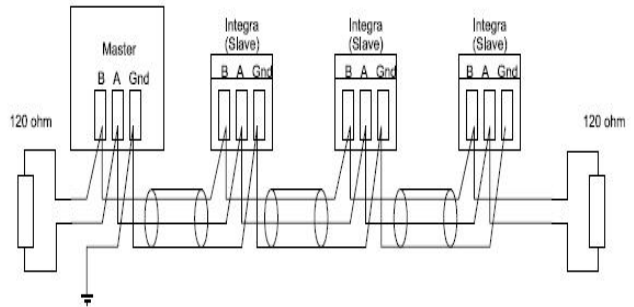


Fig7 A RS-485 Cable Connections



Fig 8 A daisy chain connection

VII. PLANT CONNECT ARCHITECTURE

Data acquisition server (das) connects and polls devices which are to be monitored. Das uses protocols like OPC, MODBUS etc..one das per site. Das sends the data to plant connect continuously (if server on intranet) or periodically (if server on internet). Plant connect server provides various functions to browser based clients

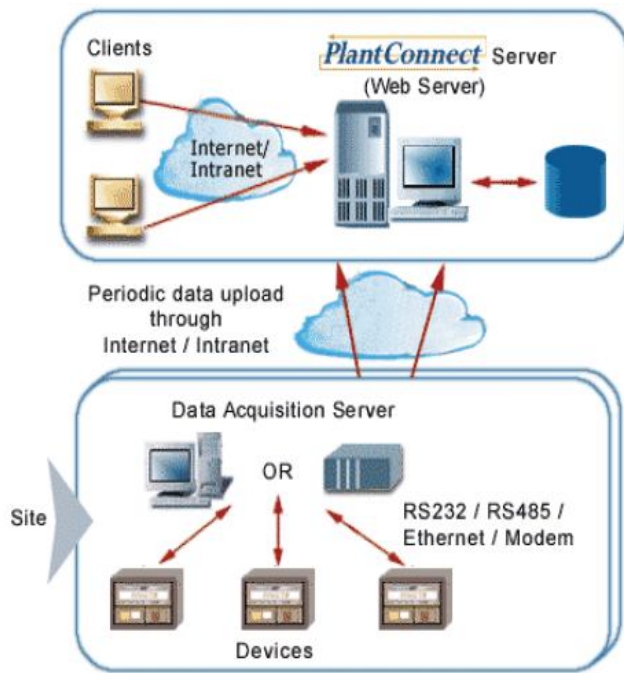


Fig 9 Plant Connect Architecture

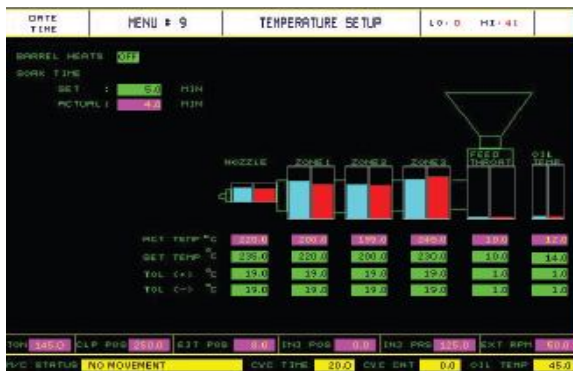


Fig 10 Web based system for monitoring & control of devices through intranet/internet.

A special version of Plant Connect A Plant Connectis created for monitoring ambient air quality. is Monitoring is done by acquiring data from various gas analysers for measuring CO, SO2, NOX, SPM etc. and from weather station. System allows uCO, user to ser continuously monitor data as well as perform analysis through vacontinuously various reports and rious charts. It supports various chartschartslike wind rose, poll rose, perspective bar, polar like chart etc.

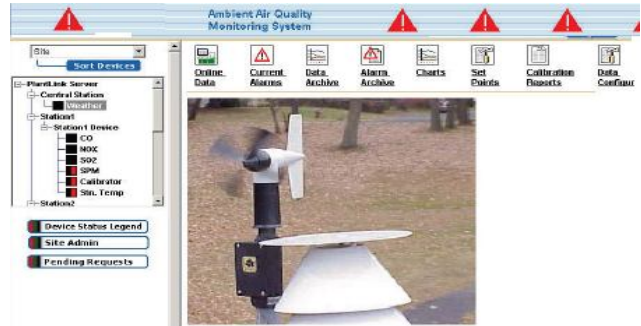


Fig 11 Plant Connect -AAQMS

VIII. CASE STUDY

A) PLC PROGRAMMING

There are many companies who manufacture PLC like Siemens , GE Fanuc , Schneider, Allen bradly etc. each one of above supply there ladder logic programming software along communication here ladder logic programming is done in Step 7 Microwin software used to program me all simens make PLC'S

As per user requirement the ladder logic program is done as

B) Notation

Inputs to PLC

Notation used in ladder logic	Description
I0.0	Relay output from scanner 1 when any LPG sensor reaches alarm condition 60% LPG to gas ratio
I0.1	Relay output from scanner 2 when any LPG sensor reaches alarm condition 60% LPG to gas ratio
I0.2	Relay output from scanner 3 when any LPG sensor reaches alarm condition 60% LPG to gas ratio
I0.3	Relay output from scanner 1 when any LPG sensor reaches alarm condition 80% LPG to gas ratio
I0.4	Relay output from scanner 2 when any LPG sensor reaches alarm condition 80% LPG to gas ratio
I0.5	Relay output from scanner 3 when any LPG sensor reaches alarm condition 80% LPG to gas ratio
I0.6	Relay output from scanner 1 when any Smoke sensor detects smoke
I0.7	Relay output from scanner 2 when any Smoke sensor detects smoke
I1.0	Relay output from scanner 3 when any Smoke sensor detects smoke
I1.1	Ultrasonic sensor relay output when level reaches 5mts from top
I1.2	Ultrasonic sensor relay output when

	level reaches 0.5mts from top
Output from PLC	
Notation used in ladder logic	Description
L0.1 (OUT_01)	Alarm in operator room LPG in air 60%
L0.2(OUT_02)	Hotter alarm blows in plant LPG in air 80%
L0.3(OUT_03)	Alarm smoke detection starts artificial rain
L0.4(OUT_04)	On/Off water tank filling pump

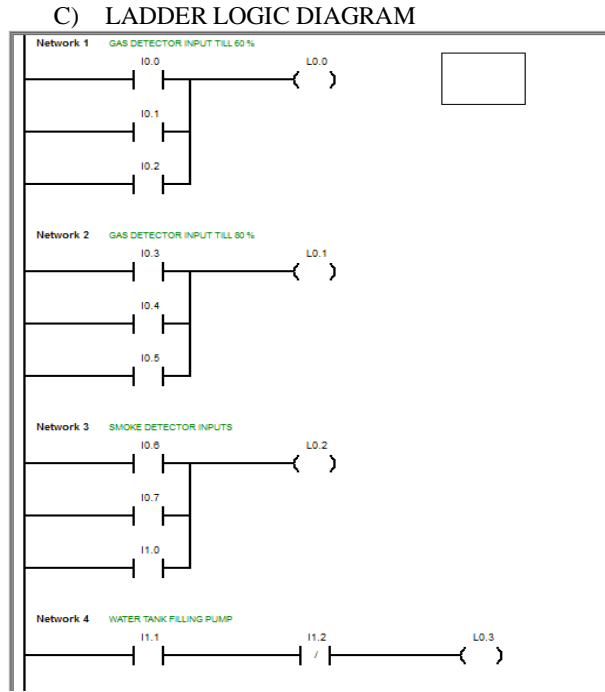


FIG 12 Ladder logic diagram

IX. SIMULATION AND RESSULT OF LADDER LOGIC

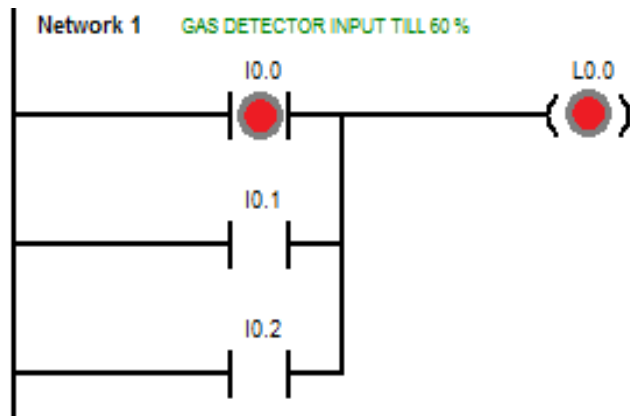


FIG 13 Alarm 1 from scanner 1

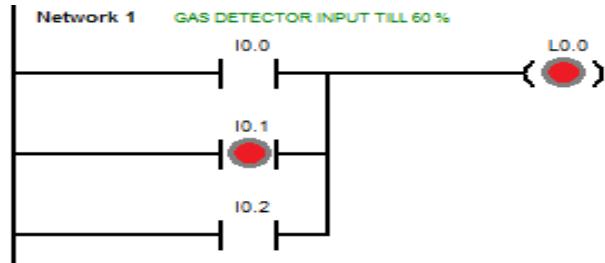


FIG 14 Alarm 1 from scanner 2

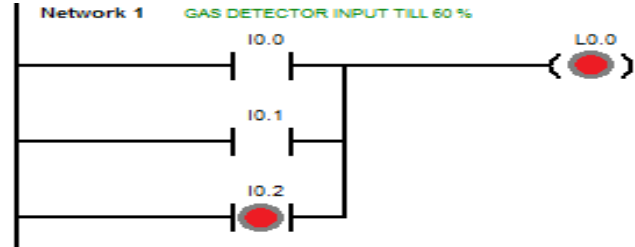


FIG 15 Alarm 1 from scanner 3

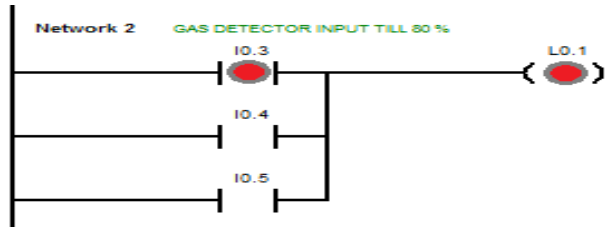


FIG 16 Alarm 2 from scanner 1

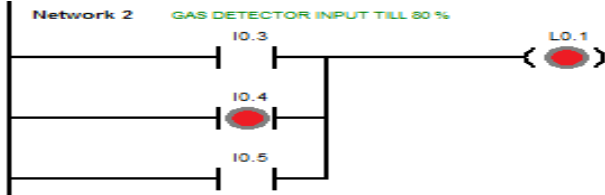


FIG 17 Alarm 2 from scanner 2

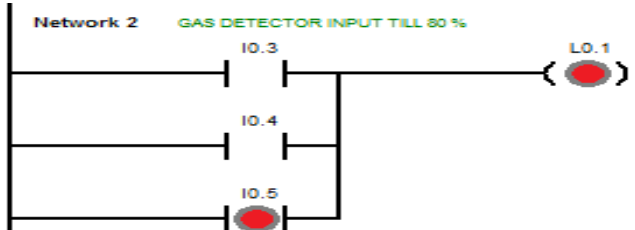


FIG 18 Alarm 2 from scanner 3

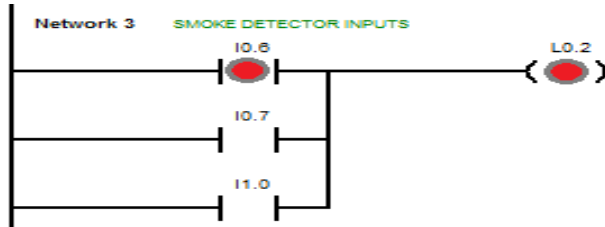


FIG 19 Smoke detection in scanner 1 location artificial rain starts

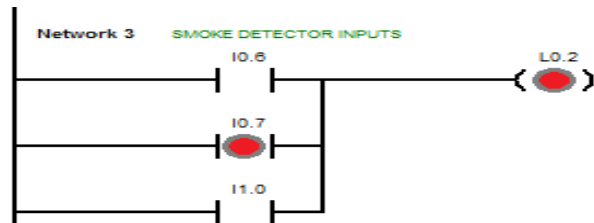


FIG 20 Smoke detection in scanner 2 location artificial rain starts

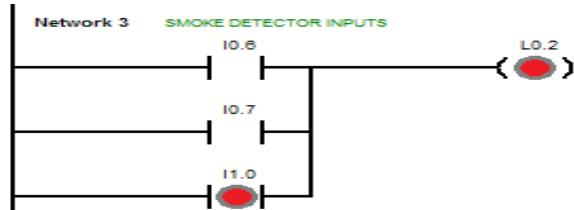


FIG 21 Smoke detection in scanner 1 location artificial rain starts

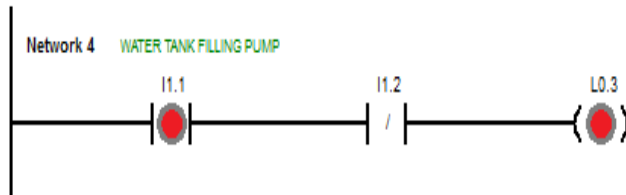


FIG 22 Water level below 5 meter from top filling pump ON

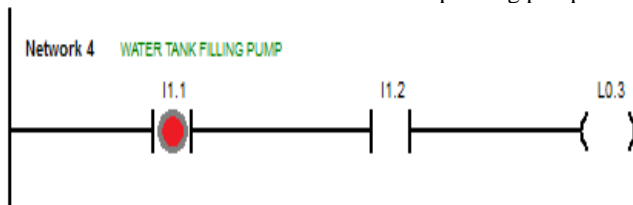


FIG 23 Water level at 0.5 meter from top filling pump OFF

X. CONCLUSIONS

This paper addressed some of the problems that arise in deploying a middleware layer for supporting SOAs in next-generation industrial automation platforms. In particular, real-time and QOS aspects have been addressed, giving an effective way to guarantee QOS in service provisioning through SOA. The architecture of the proposed framework has been described, and its effectiveness has been shown by means of extensive experimental evaluations, both quantitative and qualitative, highlighting that the framework provides significant and effective advantages over existing solutions.

REFERENCES

[1] Rachna Dhand ,Dr. Gareth Lee ,Dr. Graeme Cole ,School Of Engineering And Energy, Murdoch University, Murdoch 615,0, Applying Service-Oriented

Architecture To Industrial Control Systems Australia , 2010
 [2] Birgit Vogel-Heuser, Universität Kassel; Gunther Kegel, Pepperl+Fuchs ,Klaus Bender, Technische Universität München; Klaus Wucherer , Global Information Architecture For Industrial Automation, 2009
 [3] Tommaso Cucinotta, Member, Ieee, Antonio Mancina, Gaetano F. Anastasi, Giuseppe Lipari, Leonardo Mangeruca, Roberto Checco, And Fulvio Rusinà, A Real-Time Service-Oriented Architecture For Industrial Automation , 2009
 [4] Basic Of Sensors, Siemens,2010
 [5] Hanwei Sensors Mq-6, Wwww.Hwsensor.Com,2011
 [6] L.A. Bryan E.A. Bryan, Programmable Controllers, Chapter 1 ,2nd edition,1997
 [7] Programmable Logic Controllers: Programming Methods and Applications by John R. Hackworth and Frederick D. Hackworth, Jr. CHAPTER 1,2,Prentice hall , 2009
 [8] Keith Stouffer, Joe Falco, Karen Kent, Guide to Supervisory Control and Data Acquisition (SCADA) and Industrial Control Systems Security Recommendations of the National Institute of Standards and Technology 2006
 [9] RS485 & Modbus Protocol Guide, Tyco electronics
 [10] National Communications System , Technical Information Bulletin 04-1 , Supervisory Control And Data Acquisition (Scada) Systems,2004