RESEARCH ARTICLE OPEN ACCESS

# Control of an Industrial Mixer using PLC

*Dheerendra Yadav & **Keerti Sharma
*, **(Department Electronics and Instrumentation Engineering, Institute of Technology and Management, Gwalior, India)

**ABSTRACT**
This paper is a design and implementation of a monitoring and control system for industrial mixer using PLC .This mixer used for an effective, reliable and highly flexible method for the control of processes within an Industrial environment. The objective of this study is to simulate the control an industrial mixer and by doing this we can develop a program that can be used for the actual implementation of such a model. This mixer is controlled using a programmable logic controller and programmed using ladder logic language.

**Keywords-** PLC,SCADA,Sensors,Automation

## I. INTRODUCTION

In this paper we are using PLC to control the Industrial Mixer. Programmable Logic Controller (PLC) is a small computer used for automation of real-world processes, such as control of machinery on factory assembly lines. The PLC usually uses a microprocessor. The program can often control complex sequencing and is often written by engineers. The program is stored in battery-backed memory and/or EEPROMs. Unlike general-purpose computers, the PLC is packaged and designed for extended temperature ranges, dirty or dusty conditions, immunity to electrical noise, and is mechanically more rugged and resistant to vibration and impact.

By implementing this project we decreases man power, thus increase in production of the industry.

## II. DEFINITION OF PLC

A digital electronic device that uses a programmable memory to store instruction and to implement function such as logic, sequencing, timing, counting and arithmetic in order to control machines and processes.The term logic is use primarily concerned with implementing logic and switching operations .Input devices e.g. switches, and output devices e.g. motors, being controlled are connected to the PLC and then the controller monitors the inputs and outputs according t this program stored in the PLC by the operator and so controls the machine or process.Originally they were designed as a replacement for hard-wired relay and timer logic control systems.PLCs have the great advantage that it is possible to modify a control system without having to rewrite the connections to the input and output devices, the only requirement being that an operator has key in a different set of instruction. The result is a flexible system which can be used to control systems which vary quite widely in their nature and complexity.

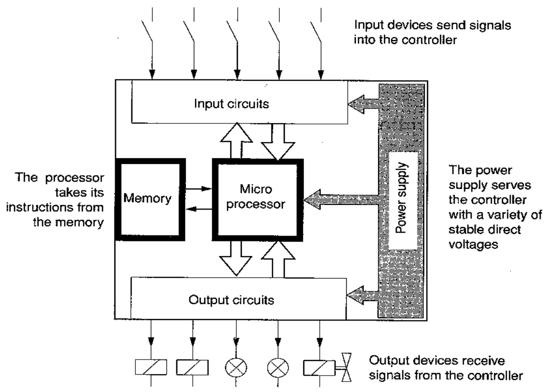
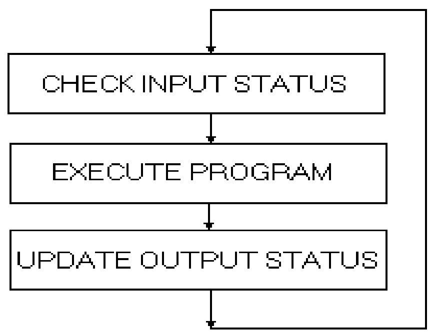



**Fig.(a) Structure of PLC**

## III. PLC OPERATION

A PLC works by continually **scanning** a program. We can think of this scan cycle as consisting of 3 important steps. There are typically more than 3 but we can focus on the important parts and not worry about the others. Typically the others are checking the system and updating the current internal counter and timer values.




**Step 1-CHECK INPUT STATUS**

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... It records this data into its memory to be used during the next step .

**Step 2-EXECUTE PROGRAM**

The PLC executes your program one instruction at a time. Maybe our 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 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 when this condition is true results for use later during the next step.

**Step 3-UPDATE OUTPUT STATUS**

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 when this condition is true.

## IV. SCOPE OF THE WORK

This paper uses the model of an Industrial mixer which has already been developed in Virtual Machine. The modelled components include pumps, thermostat, heater, level sensors, heater, the electric motor, the indicators and the liquids. A model of each of the component has been developed individually and then articulated. These components are to be controlled by the action of the programmable logic controllers which is to takes in field-inputs from these components and then give appropriate control signals as the field outputs. The software program which is used to perform the control is developed in ladder logic language using bottom -top approach (module by module) i.e. individual programs that will run the motor, the sensor, the heater will be developed individually and articulated. The ladder logic program is developed using Logix pro 500.The program is also simulated with Logix pro 500 and the test results obtained are displayed on a tabular form.

## V. BLOCK DIAGRAM AND DESCRIPTION

The main steps that were undertaken to implement this paper are:

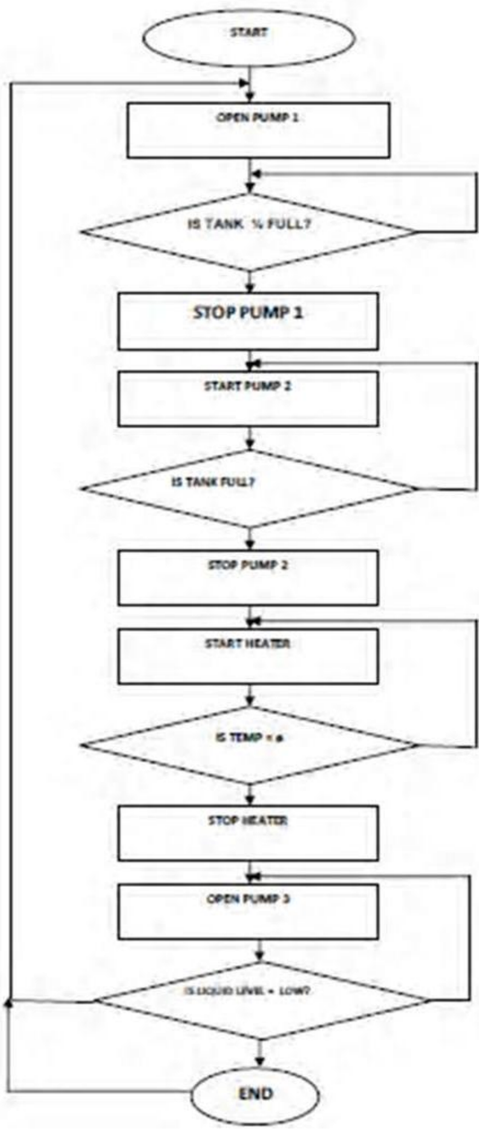
**1. Modelling the machine and its operation**: -

Here we had to decide the equipment whose properties we wish to control and it's method of functioning. For this particular paper we wish to control the action of the Industrial mixer whose operation involves valves, sensors, electric motors, and the liquids. These properties are modelled using Virtual Lab software which provides Library of Parameterized modules which represent the actual behaviour of the components of the Industrial Mixer.

**2. Assignment of the input and outputs**:- Here all the input and output devices which are intended to be connected to the PLC are identified . The input devices are assigned input addresses while the output devices are also assigned output addresses.

**3. Writing the Program**:-In this process a ladder diagram program which performs the control logic is written.

**4. Programming into Memory**:-Here first of all an I/O generation is done to prepare the system configuration of the programmable logic controller. Then the program is entered into memory either by a programming console or a computer -aided ladder software tool. It also involves debugging and checking for coding errors and the simulation of the whole operation in done in software.




***Software Programme***

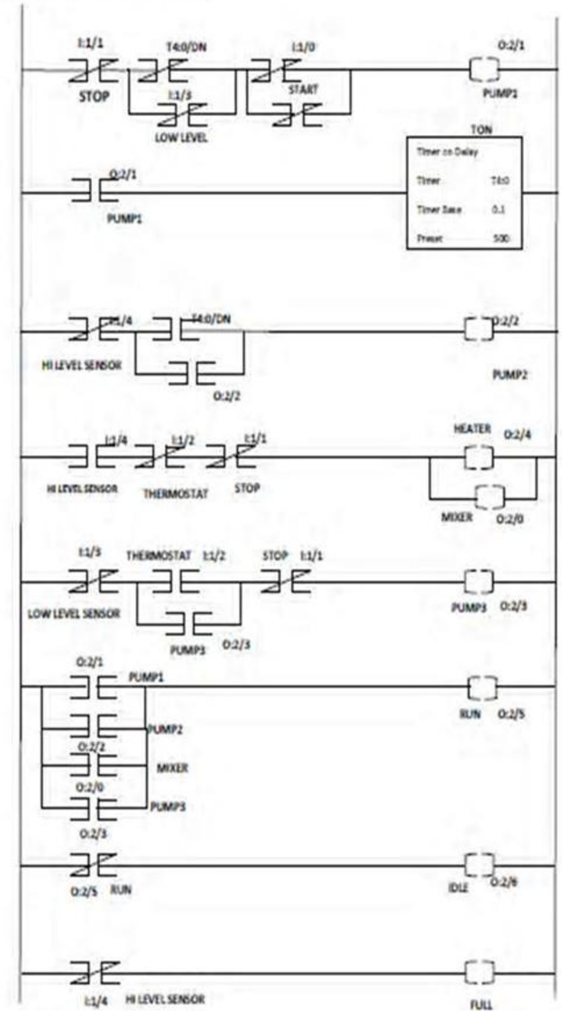



**Fig.(b)Design flow cycle**
**Fig.(c)Ladder Programming**

**5. Running and Testing**: - This involves simulating the action of the operation of the Programmable logic controller within the model Industrial mixer in software and obtaining the test results in a tabular form.

## VI. SOFTWARE SUBSYSTEM DESIGN

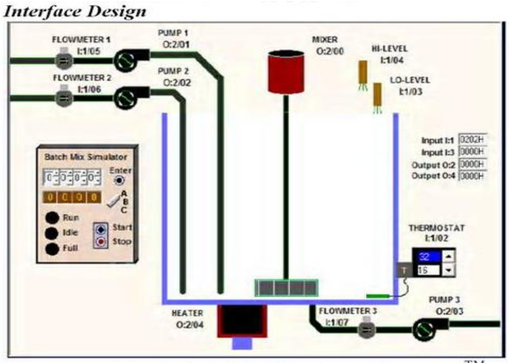
The actual software is to be written in Ladder Programming Language. The mixer is designed to take in two liquids from the input pumps, stir and heat them to a mixture before collecting the mixture at the output pumps. All operation instructions come from the programmable logic controller to either switch ON/OFF any component of the mixer subsystem depending on its state or the state of another component. The mixer operates by taking in two different liquids from the input pumps one after the

other and then performing a heating and stirring operation on this liquid within the vessel. The mixture collected from the output pump and the whole process restarts. The ladder diagram is the traditional way of representing electrical sequences of operations. These diagrams represent the interconnection of field devices in such a way that the activation, or turning ON, of one device will turn ON another device according to a predetermined sequence of events. An example of a ladder diagram is shown Ladder diagrams are easy to use and interpreted and are widely accepted in the industry. The programmable logic controller can also implement the ladder diagram so that control operations are performed more reliably. The control program used in the programmable logic controller is developed using ladder logic.
The interface design deals with how the input and output devices are assigned memory addresses according to input and output interfaces to which they are connected on the programmable logic controller. The interface design is determined by the particular programmable logic controller which is chosen for the control operation. For the control of the batch mixer we have chosen to use an 8-input discreet module with part number.

The table shows the different input and output device interface addresses and their description.

| INPUT ADDRESS | DESCRIPTION | OUTPUTADDRESS | DESCRIPTION |
|---|---|---|---|
| I:1/00 | START (NO,PB) | 0:2/00 | MIXER |
| I:1/01 | STOP (NC,PB) | 0:2/01 | PUMP1 |
| I:1/02 | THERMOSTAT | 0:2/02 | PUMP2 |
| I:1/03 | LOWLEVELSENSOR | 0:2/03 | PUMP3 |
| I:1/04 | HI LEVELSENSOR | 0:2/04 | HEATER |
| T4:0/DN | TIMER | 0:2/05 | RUN |
| C5:0/DN | COUNTER | 0:2/06 | IDLE |
| | | 0:2/07 | FULL |

*Interface Design*




ACTUAL Simulation of the Mixer in Logix pro 500TM

Fig.(d)Simulation and Interface

***Software Operation***

Figure3.shows the ladder logic application program for the control action. This code has 9 rungs which is executed within the CPU one after the other. It is seen from the ladder program that each device is assigned an address which is the memory address that holds the bit representing that particular device e.g. PUMP 1 is assigned the address O:2/01 which means that bit 1 of port 2 is an OUTPUT port which has been assigned to the device PUMP 1. This same mnemonic is applicable to all other memory addresses.

1. The first rung contains the instruction for opening PUMP1. It includes codes for the START & STOP instructions. It also includes code for the timer, [T4:0/DN], whose function will be explained in the second rung. There is an instruction here to use a counter to keep PUMP3 open when the mixer is suddenly stopped by pushing the STOP button on the control panel.

2. On the second rung, we see an instruction which uses the timer, [T4:0/DN], to stop the PUMP after a preset time. The time base is used to multiply the preset time to obtain the actual time. So for this particular application, the Timer is used to stop the PUMP1 after; (100 x 0.1)s i.e. 10s .

3. The function of the instructions on the third rung is to START PUMP2 immediately the timer deactivates PUMP1. It is seen that it also includes the code for the Timer, [T4:0/DN]. The PUMP2 is stopped when the tank is full by a TANKFULL signal.

4. structions on the fourth rung are used to start the action of the electric motor that rotates within the MIXER. This action is initiated when TANKFULL causes PUMP2 to STOP. The HEATER also starts heating the vessel at the same time so that both actions are able to effect the mixing action.

5. The fifth rung contains the instruction that is used to stop the MIXER and at the same time open PUMP3. The rung a code LOWLEVEL that stops PUMP3 when all the liquid mixture has been discharged.

6. The sixth rung is essentially used to program the indicator RUN to show us when the machine is in operation i.e. when PUMP1, PUMP2, HEATER OR PUMP3 is functioning.

7. The seventh rung also contains another instruction that is used for the indicator. Here it is used TO Indicate the IDLE status of the system which occurs when the system is NOT running.

8. The eight rung contains an instruction that is used to indicate FULL. This occurs when the full level sensor detects that the mixer is filled.

## VII. CONCLUSIONS

In conclusion, this paper has shown how programmable logic controllers are a flexible and easily adaptable to typical industrial equipment such as the industrial mixer. These controllers are programmed by using a programming language called Ladder logic language. This language is easy to understand and is used by engineers and technicians thus can be changed to suit any particular need. The Industrial mixer is one of the several applications of the programmable logic controller to the control of an industrial process. This helps to eliminate huge costs.

**References:**

[1]. Mr.Mohammed Bani Younis and Dr.Georg Frey.," A Formal Method Based Re-Implementation Concept for PLC Programs and Its Application". 1-4244-0681-1/06/2006 IEEE.

[2].Mr.LiLingWang1,HongYingwei2.,"DEVELOPMENTOFA DISTRIBUTEDCONTROL SYSTEM FORPLC-BASEDAPPLICATIONS", Proceedings of the Ninth International Conference on Machine Learning and Cybernetics, Qinumentagdao, 11-14 July 2010, 978-1-4244- 6527-9/10/©2010 IEEE.

[3]. Dr.Vijay Kumar Khatri, Ahsan Javed Ghangro, Jetandar Kumar and Syed Jaad UI Haque.,"Industrial Data Acquisition and Control System using two PLCs' Networked over MPI Network", 2009 IEEE symposium on Industrial Electronics and Application (ISIEA2009), October46,2009,KualaLumpur,Malaysia.,978-1-4244-4/09/@2009IEEE.

[4]. Mr. S.N.Rao,A senior Instrumentation Engineer at IOCL,Mathura.

[5]. Mr. D.B.V. Singh ,Head of Department in Electronics and Instrumentation department,ITM,Gwalior.

[6]. Mr. N.S.Rana,Mr.Shyam Babu and Mr. P.K. Pandey, Department in Electronics and Instrumentation ,ITM,Gwalior.