Microcontroller Application in Industrial Control & Monitoring Systems

Mir Munawar Ali[#1] Syed Habeebullah Hussaini[#2] Mohammed Zaheer Uddin Abbas[#3]
Mohammed Zain Ul Abedin[#4] Mohd Habeebullah Khan[#5] Mohd Omair[#5]
Mansoor Mohammed Khan[#6] Arutla Thirumala Chaya[#7] Challa.Thirupathi Reddy[#7]

[#1]B.Tech (EEE), M.Tech (Electrical Power Systems) Electrical Engineering Department, JNTU, Hyderabad, INDIA.
[#2] Department of Electronics and Communication Engineering, VIF College of Engineering & Technology, Hyderabad, INDIA.
[#3] Department of Electrical and Electronics Engineering, TKR College of Engineering & Technology, Hyderabad, INDIA.
[#4] Department of Electronics and Communication Engineering, Nizam Institute of Engineering & Technology, Hyderabad, INDIA.
[#5] Department of Electronics and Communication Engineering, Shadan College of Engineering & Technology, Hyderabad,INDIA.
[#6] Department of Electronics and Communication Engineering, Shaaz College of Engineering & Technology, Hyderabad, INDIA.
[#7]Department of Electrical & communication Engineering, Sreyas Institute of Engineering & Technology, Hyderabad, INDIA.

Abstract: **The design of the system can be used for industrial control and monitoring. This system is equipped with the necessary hardware so as analog and digital inputs and outputs can be interfaced with it. This paper also emphasizes the design to protect electrical circuitry by operating a Relay. This Relay gets activated whenever the electrical parameters exceed the predefined values. The Relay can be used to operate a Circuit Breaker to switch off the main electrical supply. Microcontrollers are used in automatically controlled products and devices, such as automobile engine control systems, implantable medical devices, remote controls, office machines, appliances, power tools. Digital and Analog I/P & O/P (inputs and outputs) are set as per industry standard. For ease and flexibility for employing this system, the software interface is provided by programming various functional blocks in the system.**

Keyword: **PLC (Programmable Logic Controllers), DCS (Distributed Control Systems), PID Control Timers and Device Drive Logics.**

I.INTRODUCTION

In Process Instrumentation/Batch Applications there are different systems available for monitoring and control in industry. Typical Systems used in industry are PLC (Programmable Logic Controllers) and DCS (Distributed Control Systems). These systems are very complex and costly, so they are not available for small scale industries where control is through relay based system. So, a system is proposed here that gives us a cost effective solution for small scale industrial automation. This system employs digital and analog interfacing circuitry in one controller giving a cost effective solution.
Digital inputs typically are pressure switches, level switches, liquid level switches, limit switches etc.
Analog input devices are potentiometer, Thermocouples, Temperature sensors, flow sensors, optical and magnetic sensors etc.
Digital inputs operate at 12v or 24v. Analog inputs and outputs are standard 7-24mA.Analog and digital inputs and outputs interfacing must be provided in the system.
Digital or on/off outputs are solenoids, solenoid actuator, relays and indicators. Analog output devices are 4-24mA indicators, stepper motor, servo motor etc.
For using this system in industrial applications, the software interface should be provided in the system. Various functional blocks are required for the logic design, mathematical operations on the data that is obtained through sensors, controlling the operations of various devices, Device driving mechanisms etc.
These functional blocks are enlisted below:

- Digital logic blocks:
- AND Logic
- OR Logic
- Device Drive Logics
- Analog functional Blocks:
- High/Low Select
- High/Low Compare
- Analog Transfer
- PID Control Timers

These functional blocks provide the interface between hardware and industrial applications. When this system is used for industrial control, analog and digital inputs and outputs that are required need to be interfaced with system and using functional blocks described earlier, the logic for industrial control is designed and implemented.

PLCs are used in many industries. PLCs are designed for multiple analogue and digital inputs and output arrangements, extended temperature ranges, immunity to electrical noise, and resistance to vibration and impact.

This is in contrast to non-distributed systems, which use a single controller at a central location. In a DCS, a hierarchy of controllers is connected by communications networks for command and monitoring.

Input and output modules form component parts of the DCS. The processor receives information from input modules and sends information to output modules.

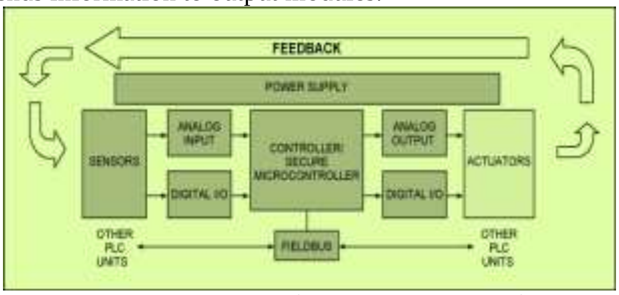

Figure 1: Block Diagram of the PLC & DCS control field

II. HARDWARE

Block Diagram

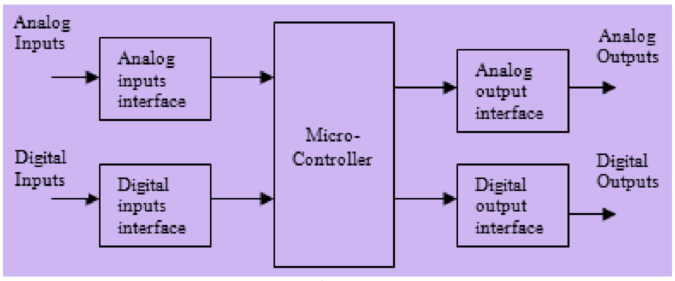

Figure 2: Block Diagram of the digital & analog system.

Figure 2 shows the basic block diagram of the system. In hardware, analog and digital inputs and outputs are interfaced to microcontroller via interface as shown in figure.

A. Digital Input Interface

Digital inputs typically are pressure switches, level switches, liquid level switches, limit switches etc. These digital inputs normally operate at 12V or 24V, whereas microcontroller supply voltage is 5V.

Also, if any fault occurs due to short or open circuit, microcontroller can be damaged permanently. To prevent this to occur, digital inputs are normally given to microcontroller after providing isolation using opto-isolators. These opto-isolators are available in IC, each IC contains two, four or eight opto-isolators.

B. Analog Input Interface

Analog input devices are potentiometer, Thermocouples, Temperature sensors, flow sensors, optical and magnetic sensors etc.

Analog inputs used in industry are typically 4-20mA. ADC input is analog voltage signal, while 4-20mA signal is analog current signal, so this 4-20mA signal can be converted to 1-5 volts using 250ohm resistor. Normally, 250 Ω resistor is not available as a standard resistor, so 220 Ω or 270Ω resistor can be used.

These inputs can be interfaced to the microcontroller through ADC (If microcontroller has inbuilt ADC, as in many case, the inputs can be given directly to the microcontroller port).

C. Digital Output Interface

Digital outputs are solenoids, solenoid actuator, relays and indicators. Microcontroller ports provide low source or sink current (For more information, refer to microcontroller's datasheet).

This current is not enough for driving digital outputs like Solenoid, Relays and Indicators etc. So, Suitable driver needs to be implemented for driving these devices. Depending upon the current requirement, suitable driver IC is selected.

So, all these interfacing circuitry has to be designed. This depends upon the number of analog and digital inputs and outputs selected for system implementation.

Also, Microcontroller may not have sufficient number of ports available. In that case, Port Expander IC is used. Serial Port expanders are more used these days. SPI/I^2C Serial Protocols can be used for serial port expanders.

D. Analog Output Interface

Analog output devices are, stepper motor, servo motors 4-20mA indicators. Analog outputs 4-20mA are standard in industry. The output of the microcontroller port is given to the DAC (Digital to Analog Converter). If the analog output of DAC is voltage, then it has to be converted to current by using op-amp as voltage to current converter. Suitable Op-amp should be selected in order to avoid any loading problem.

III. FUNCTION WORKING BLOCKS

DIGITAL:

Different digital logic blocks are used for manipulating the Boolean variables in the system. Most common among them are AND logic and OR logic. They are described below:

A. OR LOGIC

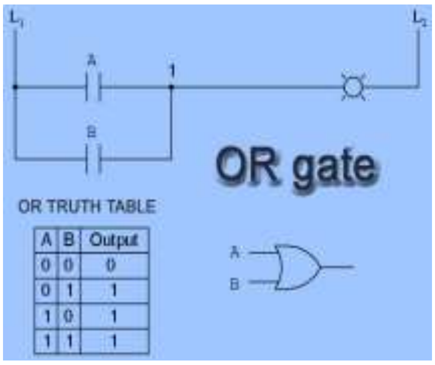

Figure 3: OR GATE Circuit implement

OR logic block gives high output if any of the digital inputs is high and low if all digital inputs are low.

B. AND LOGIC

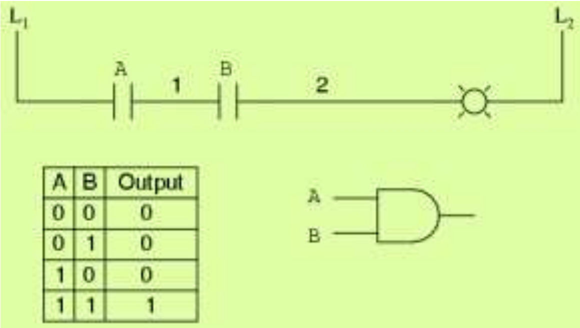

Figure 4: AND GATE Circuit implement

AND logic block gives high output only if all digital inputs to the block are high, else output is low in other case.

DEVICE DRIVE LOGIC BLOCKS:

C. UNIDIRECTIONAL DRIVE

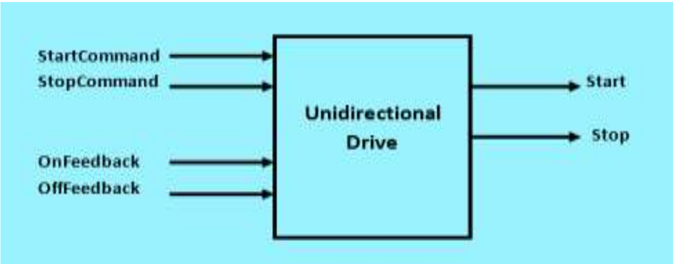

Figure 5: Unidirectional Drive

Unidirectional drive is used to control the unidirectional motor (for example, fan or pump). Motors are used for a wide range of industrial automation applications. Two commands for driving the motor - motor start command and motor stop command, because there are two separate contacts for on and off.

So, when the command is given to the unidirectional drive, the relay contact is picked up and output device is operated. The feedback for the status of the output device is given back to the block and is monitored.

If the feedback signal doesn't arrive within timeout value, the alarm is generated.

Algorithm for this drive can be derived as shown:

```
if(On Command == HIGH){ On Output = HIGH; Off Output = LOW;}
else if(Off Command == HIGH){ On Output = LOW: Off Output = HIGH;}
 Start Timer (Timeout Value);
 While (Timer_is_ON)
{if((On Feedback == On Command) || (Off Feedback == Off Command))
{ Stop Timer ();
Feedback Received = 1; Break;:
}
}
If(Feedback Received == 1)
{ No alarm ();}
else
{
Generate alarm ();
}
```

D. BIDIRECTIONAL DRIVE

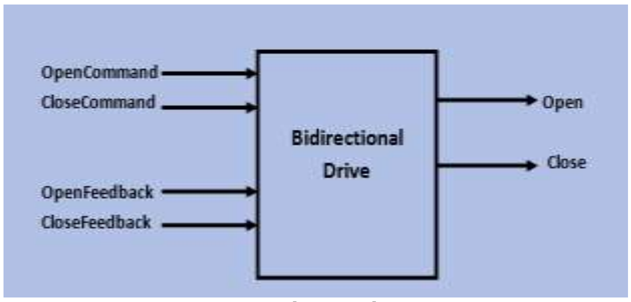

Figure 6: Bidirectional Drive

In bidirectional drive, there are two commands for driving the device in forward and reverse directions. When the open or close command is given to the bidirectional drive, the relay contact is picked up and output device is operated accordingly. The feedback for the status of the output device is given back to the block and the status of the feedback is monitored. If the feedback signal doesn't arrive within timeout value, the alarm is generated.

E. SOLENOID DRIVE

Figure 7: Solenoid Drive

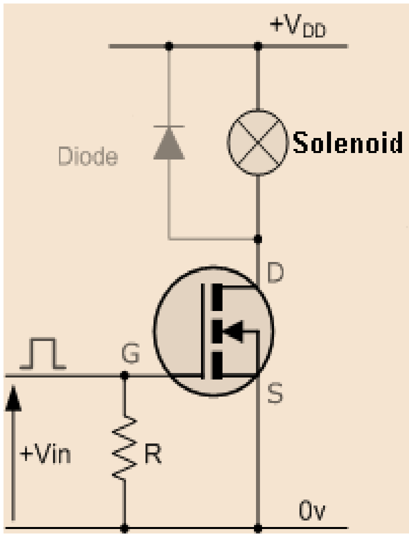

Figure 8: Solenoid drive circuit

In industries, solenoid valves are the most frequently used control elements in fluidics. Their tasks are to shut off, release, dose, distribute or mix fluids. So, to operate the solenoid valve, solenoid drive can be used. In which, command is given to the block and the output of the block causes relay pick up which in turn make valve open or close. Feedback is taken from the valve so as to confirm the status of the valve.

ANALOG:

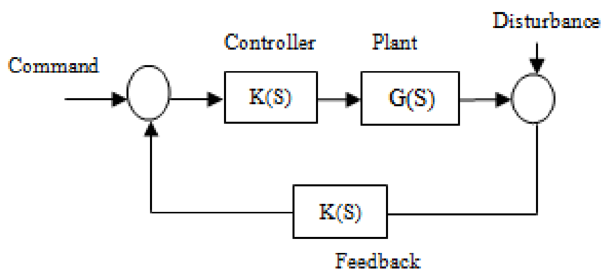

Figure 9: Basic Control System Block Diagram

A.PID

The input to the system is the set-point. A feedback is given back to the system and compared with set point. Set Point is the desired value of the output. The difference between set point and feedback is given to the controller input which can be PID Controller.

The Operation of the PID control block is dependent upon the values of parameters, Kp, Ki, Kd and set point. These parameters need to be tuned for a particular system for the proper operation of this algorithm. The output of the PID Block is used to drive the actuator.

Quantities like flow, temperature and voltage are not discrete signals but continuous ones. However, digital computers are used to manipulate sampled data values. So the digital PID control algorithm is implemented.

The algorithm for PID Control is as shown below which can be implemented on digital computer.

1. Take error as input to PID Algorithm.
2. Calculate Proportional Term: Kp * error
3. Calculate Proportional Term: Kd* (error-prevError)
4. Limit the value of Error to avoid integral wind-up
5. Calculate Integration Term: Ki
6. Accumulate error
7. Add Proportional, Integration and Derivative Terms.

B. FUNCTION GENERATOR

Function generator is used to approximate nonlinear relationship between input and output. The input range is divided into four sections and linear output to input relationship is set for each of the four sections. Function generator block computes analog output related to input according to the linear relationship of four sections.

Algorithm for Function Generator can be shown as follows:

Let, (x_1,y_1), (x_2,y_2), (x_3,y_3) and (x_4,y_4) are 4 points on graph of piecewise linear approximation of the characteristics.

Let, x be the input to the function generator block.

if($x < x_1$) then $y = x * y_1 / x_1$; // First Region

else if($x < x_2$) then $y = y_1 + (x – x_2) * (y_2 - y_1) / (x_2 - x_1)$; //Second Region

else if($x < x_3$) then $y = y_2 + (x – x_3) * (y_3 – y_2) / (x_3 – x_2)$; //Third Region

else if($x < x_4$) then $y = y_1 + (x – x_4) * (y_4 – y_3) / (x_4 – x_3)$; //Fourth Region

else $y = y_4$ //Does not fall within any region. Limit the value of output.

Function generator can be used in shaping algorithm. It can be used as noise filter. It is used to convert analog input (4mA to 20mA) into physical data such as temperature or pressure for simplification of further processing. This block is used to realize the nonlinear response of sensors and actuators.

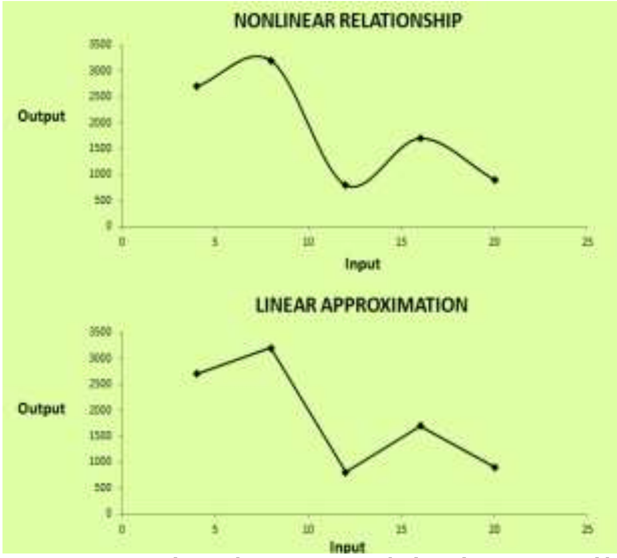

Figure 10: How non-linear characteristics can be linearly approximated by using function generator

C. HIGH SELECT

This functional block has 2 analog inputs (analog input1 and analog input2) and one analog output. Analog output will be the highest of the two analog inputs.

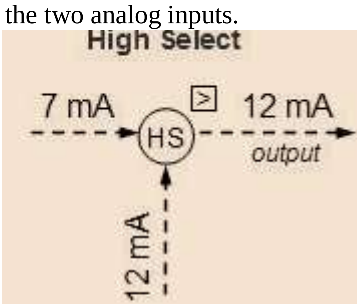

Figure 11: Implement of High stage function

D. LOW SELECT

This functional block has 2 analog inputs (analog input1 and analog input2) and one analog output. Analog output will be the lowest of the two analog inputs.

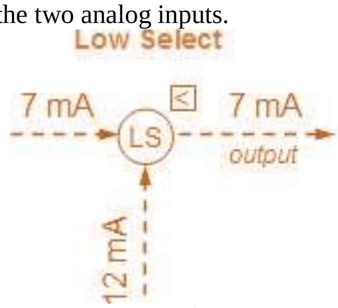

Figure 12: Implement of Low stage function

E. HIGH LOW COMPARE

This functional block has one analog input and two digital outputs (high output and low output). If input analog signal has value greater than high limit, high output will be high and low output will be low. If analog signal has value less than low limit then high output will be low and low output will be high.

If input is between high and low limit then both high and low outputs will be zero. These outputs can be used as an alarm if input goes below low limit or above high limit.
Algorithm for High Low Compare is given as shown below:
If (Input >= HIGH_LIMIT) then High Output = 1; Low Output = 0;
else if(Input <= LOW_LIMIT) then High Output = 0; Low Output =1;
else High Output = 0; Low Output = 0; End if

F. HIGH LOW LIMITER

Figure 13: High Low Limiter Function

This block limits the input signal between two specified high limit and low limit. The analog output equals analog input. when the analog input is between the high limit and the low limit. The analog output equals the high limit when the analog input is higher than the high limit and the analog output equals low limit when the analog input is lower than the low limit.
Algorithm for this block can be given as:
if(Input >= HIGH_LIMIT) then Output = HIGH_LIMIT;
else if(Input <= LOW_LIMIT) then Output = LOW_LIMIT;
else Output = Input; end if

G. AUTO MANUAL STATION

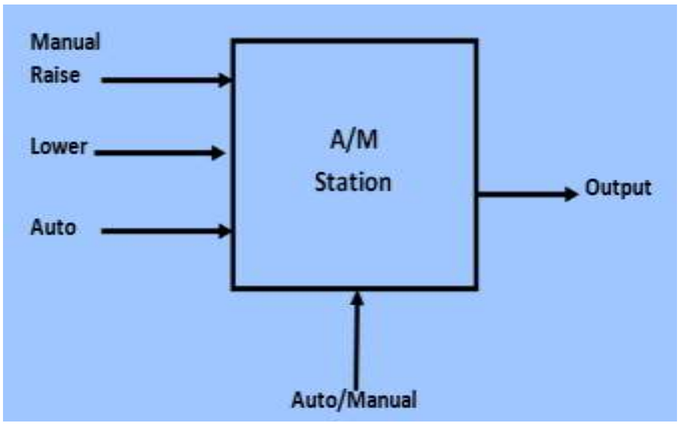

Figure 14: Auto/Manual Station Function

Auto manual station is used with PID Algorithm block. Auto/Manual select signal will decide whether the program is running in the automatic mode or manual mode. In manual mode the signals at the raise will cause the output to rise by some value, whereas the signal at the lower will cause the output to go down by some value. If the Program is in the automatic mode then the output of the PID is directly given as the output. It is useful to control the process manually if there is failure.

H. ANALOG TRANSFER

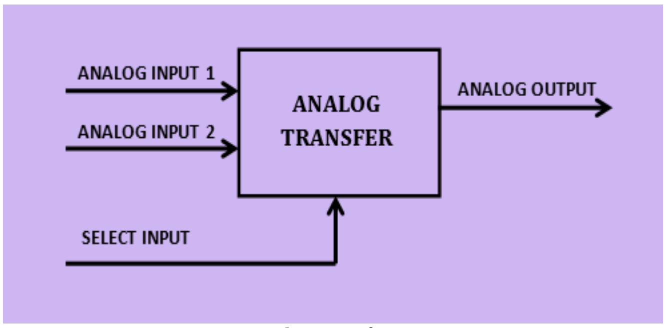

Figure 15: Analog Transfer Function

This block has two analog inputs (analog input1 and analog input2) one digital input (select input) and one analog output. If the select input is at logic 1 then analog output equals analog input1 and if the select pin is at logic 0 the analog output equals analog input2.

IV. SOFTWARE FUNCTIONING PROCESS

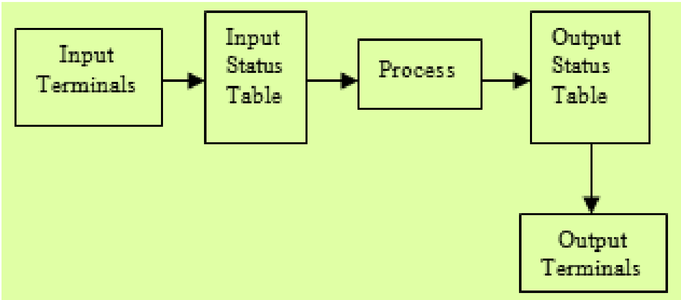

Figure 16: Flow Chart

Flow chart of the software is shown in figure16 . First of all, digital and analog inputs are scanned from the terminals; values of inputs are stored in input status table. Then process block is executed. Process cycle is the actual processing as per the requirement. It calls different function blocks, processes the inputs from input status table and output status table is updated accordingly. Then data is sent to output terminals from output status table. Various digital and analog outputs are connected to the output terminals.

V.CONCLUSION

The above paper clearly explains the microcontroller usage in the control process and the relevant monitoring of Input and Output values. The practical aspects of hardware and necessary interfacing circuits were studied and discussed in detail. Various functional blocks for control systems are described in accordance. Also, Algorithms of various functional blocks are prepared.

REFERENCES

[1] Muhammad Ali Mazidi; *8051 microcontroller and embedded systems*; 2nd Edition, 2011.
[2] Jacqueline Wilkie, Michael Johnson and Reza Katebi; *Control Engineering : An Introductory Course*; Palgrave Macmillan,2001
[3] Gayakwad. R.A, *Op-amps and linear integrated circuits*;4th Edition, 2010

[4] John.A.Shaw ; *PID Control Algorithm How it works, How to tune it and how to use it*; 2nd Edition,2003

[5] Richard Barnett, Larry O'Cull and Sarah Cox**;** *Embedded C Programming and the Atmel AVR*; Cengage Learning, 2006

[6] Christian Diedrich, Francesco Russo, Ludwig Winkel ,Terry Blevins; *Function Block Application in Control System Based on IEC 61804;*

[7] R.W. Lewis; Programming industrial control systems using IEC 1131-3; Revised Edition

[8] Michael Barr. "Embedded Systems Glossary". Neutrino Technical Library. Retrieved 2007-04-21.

[9] Heath, Steve (2003). Embedded systems design. EDN series for design engineers (2 ed.). Newnes. p. 2. ISBN 978-0-7506-5546-0. "An embedded system is a microprocessor based system that is built to control a function or a range of functions."

[10] Michael Barr; Anthony J. Massa (2006). "Introduction". Programming embedded systems: with C and GNU development tools. O'Reilly. pp. 1–2. ISBN 978-0-596-00983-0.

[11] Giovino, Bill. "Micro controller.com – Embedded Systems supersite".

[12] Kinner, Russell H., P.E. Designing Programable Controller Application Programs Using More than One Designer. 14th Annual International Programmable Controllers Conference Proceedings, 1985, p. 97-110.

[13] W. Bolton, Programmable Logic Controllers, Fifth Edition, Newnes, 2009 ISBN 978-1-85617-751-1, Chapter 1.

[14] "Introduction to Industrial Control Networks". IEEE Communications Surveys and Tutorials. 2012.

[15] Stout, T. M. and Williams, T. J. (1995). "Pioneering Work in the Field of Computer Process Control". IEEE Annals of the History of Computing.