PROGRAMMABLE LOGIC CONTROL

James Vernon: control systems principles.co.uk

ABSTRACT: This is one of a series of white papers on systems modelling, analysis and control, prepared by Control Systems Principles.co.uk to give insights into important principles and processes in control. In control systems there are a number of generic systems and methods which are encountered in all areas of industry and technology. These white papers aim to explain these important systems and methods in straightforward terms. The white papers describe what makes a particular type of system/method important, how it works and then demonstrates how to control it. The control demonstrations are performed using models of real systems designed by our founder - Peter Wellstead, and developed for manufacture by TQ Education and Training Ltd in their CE range of equipment. Where possible, results from the real system are shown. This white paper is about the ‘work horse’ of industrial automation – the programmable logic controller.

1. What is Programmable Logic Control?

Programmable Logic Control, or PLC as it is universally called, is the ‘work horse’ of industrial automation. My boss also calls it the ‘Cinderella of Control’, because it is not well treated in colleges and universities, but when control engineers use PLCs in industry they wonder how they lived without them, because they are so important and so wide spread. We think that there are a lot of three term controllers in the world, but there are many times more PLCs!

Programmable Logic Control is important because all production processes go through a fixed repetitive sequence of operations that involve logical steps and decisions. A PLC is used to control, time and regulate the sequence. Examples of production processes that are controlled using PLCs are – metal machining sequences, product assembly lines and batch chemical processes. Logical control has been used to control sequences of actions in automatic manufacturing systems for many years. Originally, a logical control system was ‘hard wired’ using electronic relays, timers and logical units. These systems were inflexible. Once a ‘hard wired’ logic system had been constructed, then if the machining schedule was altered for another type of product, the logic control system had to be manually rewired for the new application. This was inflexible and time consuming, it restricted the production scheduling of a factory and made changing of products difficult.

This situation began to change in the 1970’s when mini-computers, became available. It is easy to perform switching, time delays and logic in computer programmes. A computer programme is also easy to change when the control problem changes and allows many more logical functions than are possible with hardwired logic control. When reliable microprocessors became available in the 1980’s, with cheap memory and flexible input-output features, then the modern generation of PLCs began to emerge. Today the microcomputer based PLC is a robust, reliable instrument with many functions and features. Even cheap PLC’s are able to control a medium scale automatic machining station or chemical batch reactor. Large PLC systems are capable of running an entire factory automation system. PLC’s have the basic structure, shown in Figure 1.

From the figure, the PLC has four main units:

1. The Programme Memory –the instructions for the logical control sequence are stored here
2. The Data Memory – the status of switches, interlocks, past values of data and other working data is stored here
3. The output devices – these are hardware/software drivers for the industrial process actuators, such as solenoid switches, motors and valves
4. The input devices – these are hardware/software drivers for the industrial process sensors such as switch status sensors, proximity detectors, interlock settings and so on.

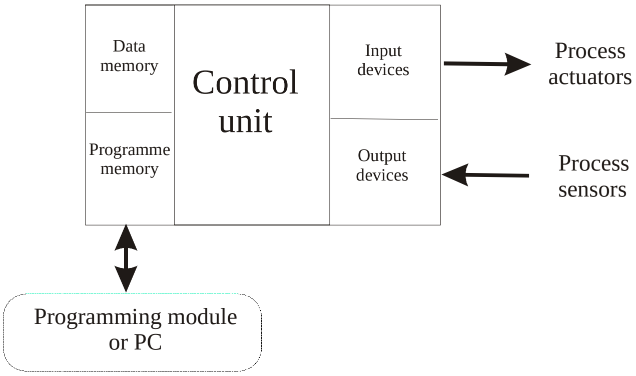

Figure 1. Schematic of a PLC

In addition to this the PLC has a programming unit. This is either a special module, a PC connected to the PLC by the serial link, or both. The programming unit is used to build, test and edit the logical sequence that the PLC will execute. At a basic level PLCs are programmed in a simple form of assembly code. Each manufacturer has their own standards and definitions for these codes. There are other programming languages, including the IEC 61131-3 standard, Sequential Function Chart, Function Block Diagrams. However, a long established standard programming language for programmable logic, called 'Ladder Logic', is universally understood by PLC programmers.

In recent years PLC's have become more sophisticated. As a result it is sometimes difficult to put all their features into the ladder logic framework, and so the alternative programming languages are becoming more widely used. At this time however, ladder logic is the standard way to describe a PLC programme.

2. Engineering Motivation – A typical batch process

Figure 2: The CE111 PLC Process

When a logic sequence has been programmed and downloaded to the Programme Memory unit, then the programme can be started, paused, single stepped and stopped from the PLC unit itself. The normal structure of a PLC sequence is a repetitive chain of elementary actions that are determined by the status of the input devices from the industrial process. The following paragraphs describe a typical PLC sequence with reference to a chemical engineering automation process.

All industrial production processes follow a fixed sequence of actions that are determined by the identified steps in the production process and the known reactions that must be made during the process. Industrial production processes can be the manufacture of mechanical or electrical products in a machining and assembly line, processing of raw materials, the formation of chemical or pharmaceutical products in a chemical process, plus many more. In this section I will describe a simple liquid processing sequence that is typical of many liquid processes sequences that are found in the process industries, [1].

To keep the illustration simple, a two-tank batch process from the chemical industry (as shown in Figure 2) is used. The process consists of two tanks, one mounted above the other. There is a pump that can be switched on or off, to pump liquid to the upper tank (Tank 1). Tank 1 has a solenoid operated valve in the base. This valve (Valve 3) is normally closed but it can be switched open so as to release fluid from Tank 1 into the lower tank (Tank 2). Tank 2 has a solenoid-operated valve (Valve 4) in its base so that Tank 2 can be emptied into a reservoir. The pump and valve inputs to the process are binary input signals that actuate the process by switching the pump on or off, or opening or closing the valves. The sensors are also binary; they tell the PLC whether a switch is open or closed. The exception is a flow meter, which gives a pulsed signal, which can be processed to find pump flow rates. The pulsed input for the flow meter is an example of how the scope of PLCs is expanding beyond pure logical signals.

All the inputs and outputs for the process in Figure 1 are listed in the table below:

Variable	**Process Input or Output**	Function
Fluid level, L1	Output	Tank 2 LOW level Switch
Fluid level, L2	Output	Tank 2 HIGH level Switch
Fluid level, L3	Output	Tank 1 LOW level Switch
Fluid level, L4	Output	Tank 1 HIGH level Switch
Fluid level, L5	Output	Reservoir LOW level Float Switch
Valve,V1	Input	Pump to Reservoir Valve (by-pass)
Valve,V2	Input	Tank 1 Feed Valve
Valve,V3	Input	Tank 1 Drain Valve
Valve,V4	Input	Tank 2 Drain Valve
Pump, P1	Input	Pump ON/OFF
Flow meter, F1	Output	Flow meter, pulse related to flow rate

Table 1. The variables for the PLC system

A simple batch sequence for this system might be as follows:

1. Open valve V2, close V3, and set pump P1 on (this starts filling Tank 1)
2. Wait until L4 is set and set pump P1 off (this stops filling Tank 1 when it is full)
3. Wait for 10 seconds (this could be to allow fluid mixing, reactions or settling to take place)
4. Open valve V3, close V4 (this starts emptying the contents of Tank 1 into Tank 2).
5. Wait until L3 is set and close V3 (this waits until all of Tank 1 is emptied into Tank 2)
6. Go to step 1 to repeat the sequence indefinitely

More complex fluid processing sequences can be set up with the process system in Figure 1, but I think that the key elements of a batch sequencing system are shown here. In the next section I will explain the elements of a PLC ladder logic programme.

3. Elements of Ladder Logic

The basic components in a ladder logic programme are the contact and the coil. The contact is the name given to a general input device – in can set by an external switch, an internally set logic or timer function. The coil is the name given to a general output device – and is used to drive motors, solenoids and other process actuators. These two basic devices are shown in Figure 3.

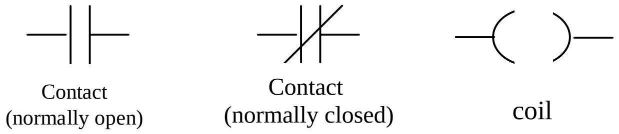

Figure 3. Basic Ladder Logic Components

Contacts

Contacts are the input devices in a ladder diagram. They are opened and closed by either an externally applied signal (usually represented by X), internal timer(labelled T), counter (labelled C), or internal logical flag (M and S). As shown in the table contacts are available in two forms - normally open and normally closed.

Coils

Coils are the output devices in a ladder diagram. They are used to operate external devices, and internal timers, counters and flags. Some manufacturers include coils that allow special operations to be performed which extend the capability of the PLC beyond that which can be obtained with the simple coils and contacts. The following instructions are a selection of the extra features that can be operated using the special coil feature from one manufacturer (Mitsubusi).

Master Control (MC) and Master Reset (MCR)

A master control block is a set of ladder programmes that are executed when an input device is activated. The end of a master control block is marked by the master reset instruction. It is the ladder logic equivalent of a subroutine or procedure in a conventional programming language.

Set (SET) and Reset (RST) Devices

These instructions are used to permanently set or reset an output device when a designated input is set. The output device holds the given value even if the input state is changed.

Reset (RST) Timer or Counter

This instruction resets a designated counter or timer. Any output device set by the timer will also be reset.

Pulse Rising Edge (PLS) and Pulse Falling Edge (PFE)

This instruction sets a designated output device for one cycle when either a rising (PLS) or falling edge (PFE) is detected from an input device.

Using the basic elements described here it is possible to build very complex logical sequences. In the next section I will show some actually examples of programmes written for the PLC process.

5. PLC Systems.

There are many, many companies offering PLC systems. The choice is large and confusing, and so it is best restrict the selection by first defining the features that are required in a particular application, (e.g. number of inputs, number of outputs, etc). This will reduce the number of options and there are several web sites that will help refine the choice beyond this. To motivate this, consider the application of controlling the PLC process described in Section 2. The PLC process has 6 outputs (level, flow rate and detectors) and 5 inputs (valve solenoids and motor). This gives the minimum I/O requirement of the PLC system needed to drive it. In addition for an application the system should be able to achieve a certain scan time, allow sufficient instructions for the most complex programme planned, and have a large enough programme and data memory. There will also be system requirements for input and output drive option, programming tools and communications protocols.

In a training and prototyping situation, we have found the CE123 PLC Trainer shown in Figure 4 to be excellent. It incorporates a 14 I/O channel industrial PLC system. The important features of the CE132 system is the packaging with buffered input and output connectors, manual input output switches and status indicators.

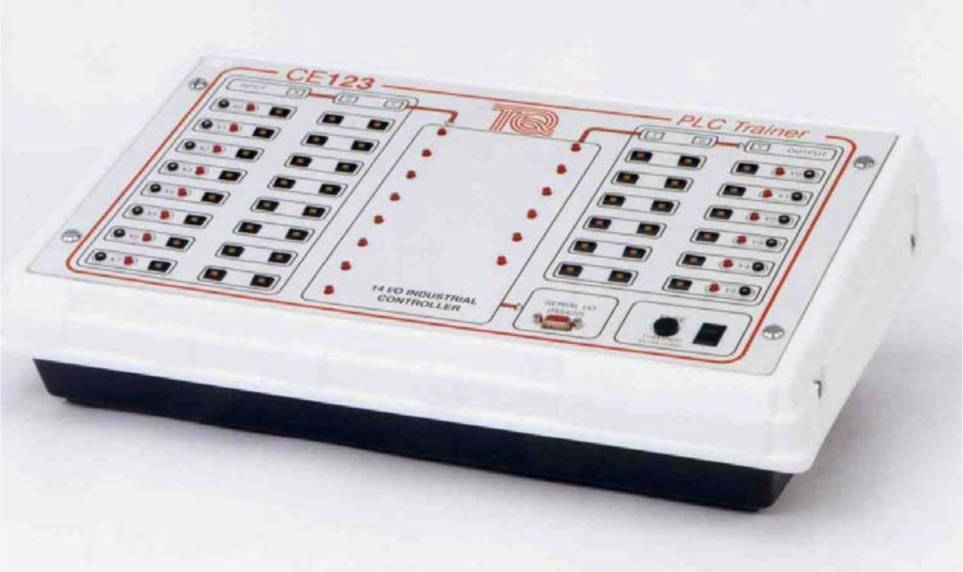

Figure 4. The CE123 PLC System

As I have remarked in Section 1, PLCs have a programming module that allows the user to enter the instructions that make up the logic sequence to be followed. There are a number of languages possible, but the most widely understood is the ladder logic scheme. Ladder logic instructions are entered into a ladder diagram that describes the entire logical sequence of a particular process. As an example, the Figure 5 shows a screen shot of a simple ladder diagram to be executed in the CE123 PLC system for the control of the PLC process.

The aim of this logic controller circuit is to fill Tank 1 and hold the level in the tank at the level of the ‘Tank 1 HIGH level switch’ (see Table 1). The programme is started by setting the input X6 to ‘1’ (this would be done with one of the manual switches on the CE123. The programme operates a level control cycle whereby Valves V2 and V3 are opened and closed, according to the state of the L4 sensor in Tank 1. The objective is to maintain the tank level in Tank 1 around the high mark. Going through the ladder diagram rung by rung:

Rung 1 starts the pump when X6 is high and if the reservoir level is high, it also opens the by-pass valve if the feed to Tank 1 is closed.

Rung 2 when the pump is high (or the reservoir level is too low) the Tank 2 valve is opened. (This is to drain water back to the reservoir and is not part of the control cycle.)

Rung 3 when the Tank 1 high level indicator is low and the input X6 is high then the Tank 1 feed is valve is opened.

Rung 4 when the Tank 1 high level indicator is high the drain valve to Tank 1 is opened.

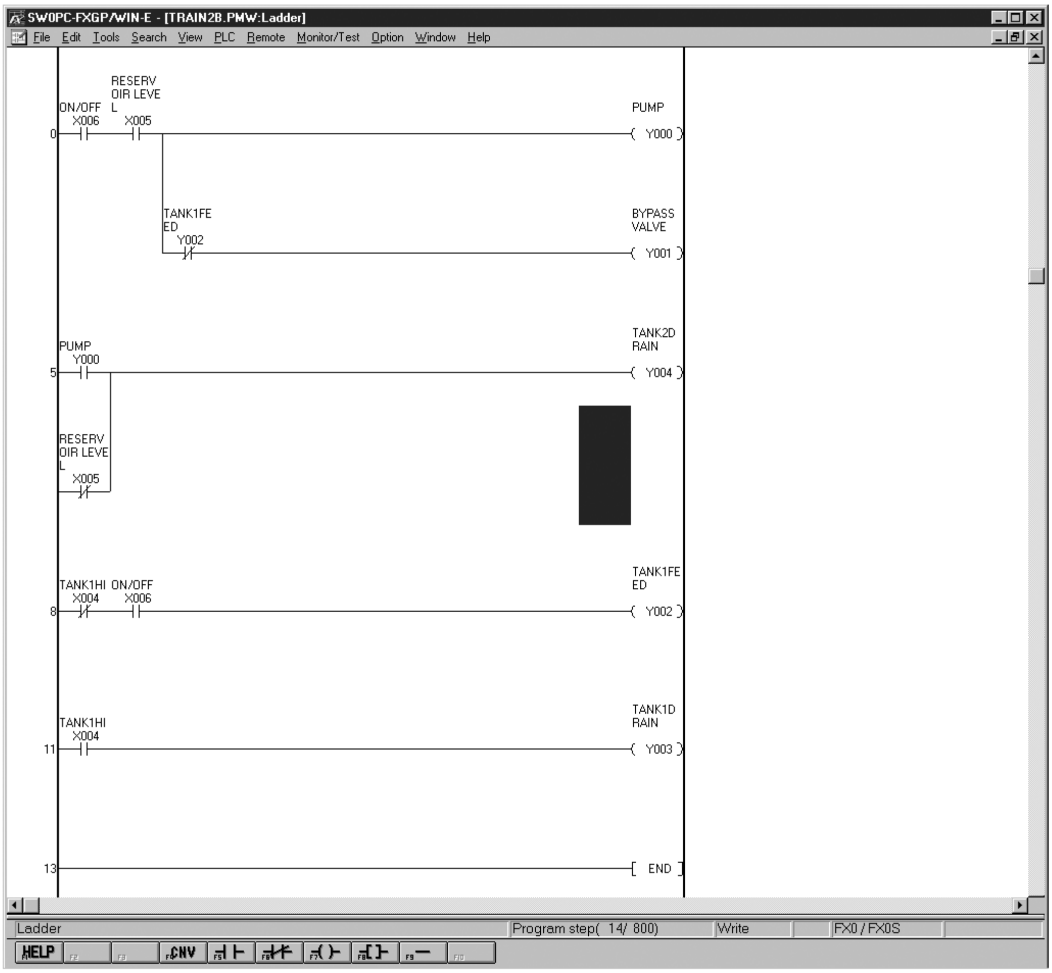

Figure 5. A Ladder Diagram for PLC Process Control

4. A Final Word

I am desolated to reveal to you that it is not possible to answer questions about our white papers, unless we have a contract with your organisation. For more information about PLC control and the CE123 PLC System go to the TQ Education and Training web site using the links on our web site www.control-systems-principles.co.uk or use the email info@tq.com. There are many good books on PLC and industrial automation. For example, [2] is clear and helpful, and [3] is a good introduction the new IEC standard. Also there are many manufacturer’s tutorials on PLCs which are available via the world-wide-web.

7. References

1. D.E. Seborg, T. F. Edgar, D. A. Melichamp, *Process Dynamics and Control*, Wiley, 1989.
2. G. Warnock, *Programmable Controllers: Operation and Application*, Prentice Hall, 1988.
3. R. W. Lewis, Programming industrial control systems using IEC 1131-1, IEE Press, 1998