The Programmable Logic Controller: its prehistory, emergence and application

Mark John Walker
BSc, MSc

A thesis submitted for the degree of
Doctor of Philosophy

28 September 2012

Department of Communication and Systems
Faculty of Mathematics, Computing and Technology
The Open University

DATE OF SUBMISSION: 2? SEPTEMBER 2012
DATE OF AWARD: 21 ?????? 2012

Abstract

Programmable Logic Controllers (PLCs) are widely used devices controlling industrial machines and processes and many other diverse applications, requiring primarily, combinatorial logic and sequential control. The PLC is a *hidden* technology, little known by the general public and overlooked in academic historical studies of technology. The research reported in this thesis aims to address this lack of awareness.

The thesis explores the development of sequential and combinatorial logic control technologies, the emergence of the PLC, its subsequent development and its industrial applications. Patents and first-hand accounts and experiences from senior industrial engineers in a number of diverse manufacturing industries have been used as the primary research sources since, as a hidden technology, academic historical accounts are sparse. This approach illustrates, through using the PLC as an example, a potential method of studying other, unrelated hidden technologies.

The research has revealed the influence of geography, industrial settings and earlier engineering practices on the design, selection and application of PLC control technologies, and comments on the how these influences define specific *communities of practice*.

Acknowledgements

First and foremost I would like to express my sincere thanks to my supervisors, Professor Chris Bissell and Professor John Monk. Their support, expert guidance, infinite wisdom and incredible patience have made this thesis possible.

My thanks also go to David Young, David Leeming, Peter Bruce, Alan Morris, Tony Daley, Percy Hammond and John Pittwood for their invaluable contribution to the research, it is much appreciated. Also, my thanks to Tim Metcalfe for allowing me to use the diagnostic media plant example.

Last but certainly not least, I would like to say a special thank you to my wife, Philippa, for her love and support (moral and financial), encouragement, and unlimited levels of patience. I really couldn't have done it without her!

Contents

List of Figures

Glossary

5TI	-	Texas Instruments PLC
A/D	-	Analogue to digital converter
AB	-	Allen Bradley Company
Combinatorial Control	-	Control that performs logical operations on plant parameters that have binary states.
CPU	-	Central Processing Unit
D/A	-	Digital to analogue converter
DCS	-	Distributed Control System
DDC	-	Direct Digital Control
FBD	-	Function Block Diagram
Fieldbus	-	Published Industrial network standard
I/O	-	Inputs and Outputs (analogue or digital)
IL	-	Instruction List (IEC 61131 programming Language)
Image	-	Memory reserved for inputs and outputs
Input Image	-	Memory reserved for inputs
Ladder Logic	-	Notation used to represent relay circuits for programming purposes
LD	-	Ladder Diagram (IEC 61131 programming Language)
Mainframe	-	Main rack-based computer
MES	-	Manufacturing Execution System
Mimic Panel	-	Hard-wired lamps and switches used to represent a process plant
Modbus	-	Modicon digital communications protocol
Modicon	-	PLC manufacturer (from **Mod**ular **Di**gital **Con**troller)
MPC 80	-	Microprocessor Process Controller '80'
Output Image	-	Memory reserved for outputs
Patent: Claims	-	Features of the invention that are "claimed" to be novel.
Patent: Filing Date	-	Date the patent was first filed
Patent: Priority Number	-	Number representing the patent and filing date, used to give a priority to the patent in other countries.
PC	-	Programmable Controller
PLC	-	Programmable Logic Controller
Process Industry	-	Description used to refer to continuous process control applications
Profibus	-	Process Fieldbus
Relay Ladder Logic	-	Notation used to represent relay circuits for programming purposes
SCADA	-	Supervisory Control and Data Acquisition
Sequential Control	-	Control that determines the order in which actions follow one another
Sequential controller	-	Device that controls the order in which actions follow one another
SFC	-	Sequential Function Chart (IEC 61131 programming Language based on GRAFCET)
ST	-	Structured Text (IEC 61131 programming Language)
TCS	-	Turnbull Control Systems

Chapter 1 – Introduction

1.1. Introduction

Control, in an engineering context, can be interpreted simply as the means to force or influence the behaviour of a system so that it acts in a desirable way. The requirements for the systems to be controlled can be simply the automatic operation of a machine or process, the maintenance of the consistency of a product or a physical state, or the achievement of desirable responses to events and situations. In order to meet these varied requirements, different control techniques are employed.

Many control systems monitor the state of the system that they attempt to control and then provide responses or control actions in order to change or maintain the system's state in the desired way. This type of control feeds back the system's status to the controller and is generally known as 'feedback' or 'closed loop' control. Other control techniques, on the other hand, rely on inferences from prior knowledge of the system about the outcome of control actions to apply, for example, a specific control action for a specific time; this does not involve feedback and such techniques are often referred to as 'open-loop'.

One type of control requires the system's state to be altered only when particular combinations of conditions or events occur. Occurrence of these conditions can usually be represented as transitions in 'true' or 'false' states and might be referred to as a 'combinatorial logic' control system. Security door access systems, for example, employ combinatorial logic to open and close doors dependent on a number of criteria (valid security pass, push-button depressed, and so on).

The control of sequences is also a requirement for many systems. Sequences require control actions to be carried out in a particular, pre-defined order, with individual steps triggered by expiry of a time interval or the occurrence of a specific event on the plant. Lift controls, for example, require sequencing because actions have to be carried out in a particular order (the door must be closed before moving the lift to a particular floor; it must be level with the floor before opening the door for a set period of time, and so on).

Process control employs both open- and closed-loop control, but closed loop, employing feedback from measuring devices (instruments and sensors) on process plant, is particularly important. The objective of the control system is often to monitor and maintain the steady state of a process, subject to raw material or environmental variations, or changes in demand for the product or degradation in plant equipment. Such control is sometimes referred to as 'continuous control'. A coal-fired power station boiler, for example, requires continuous pressure monitoring and control of fine adjustments to fuel flows in order to maintain a specific pressure while accommodating fluctuations in steam demand and fuel quality (Waddington and Maples, 1987).

This thesis is about a particular technology that performs a variety of control tasks and is known as the Programmable Logic Controller (PLC). The PLC started by performing the task that electro-mechanical relays used to do and ultimately replaced them.

Relays and the PLC

Relays are electro-mechanical devices consisting of an electro-magnet, a sprung armature and one or more contacts. The armature is drawn to the electromagnet when energised and the mechanical action opens or closes the switch contact. Relays can perform two functions: 1) to act as a power amplifier by using light currents and low voltages to switch higher voltages and currents in separate electrical circuits; and 2) to perform simple logic operations when switches are used in combination[1]. Relays have been used widely for performing combinatorial logic control and, in conjunction with timing devices, sequential control.

Originally known as a 'Programmable Controller' or 'PC' (the term was later usurped for the personal computer), the PLC is widely used to provide industrial control solutions throughout the world. PLCs have been used to automate machines and processes throughout the manufacturing and process industries since their introduction to the automotive industry in 1969 by Modicon (Erickson, 1996, Clare et al., 2005). Initially developed as a programmable relay-replacer, the early PLC was only able to perform on-off control using programmable sequential and combinatorial logic (Erickson, 1996). In contrast to the fixed-wired relay control systems used for sequential control, the PLC's sequential logic program can be altered and

[1] Examples of simple logic constructs are the logical 'AND', consisting of two different relay contacts wired in series. A simple logic 'OR' function can be achieved with two different relay contacts connected in parallel. Complex logic constructs can be built from these basic building blocks using multiple relays and interconnecting their contacts.

stored without the need for extensive and time consuming re-wiring, thus improving changeover times for production lines (Fauci, 1997).

The design and implementation of logical and sequential control of processes, perhaps a less problematic than process control but a necessary aspect of control, has been left to the plant engineer and technician to implement – engineers who work in factories, processes and plant which the PLC is designed to control. There has been little interest by the academic community in the history and development of these technologies (some of the reasons for which will be discussed in Chapter 3), and one aim of this thesis is to redress the balance.

1.1.1. Studying a Hidden Technology

PLCs form a family of technologies — a technology genre. They are manufactured by many organisations and commonly customised to suit a particular plant's operations but nevertheless have similar attributes. Because of these shared characteristics, the thesis uses the term 'PLC' as a generic reference.

The Programmable Logic Controller is an example of a widely used technology. In 2010, PLCs had a 47.5% share of the world-wide control system market worth $10,528.8 million (Datamonitor, 2011a). In the UK, the PLC has a 49.2% share of the control system market worth $457.1 million (Datamonitor, 2011b). These figures demonstrate that the PLC is an industrially highly significant technology.

The Datamonitor report (2011a) gives no figure detailing the application split of PLCs for manufacturing and non-manufacturing use due to the control system's "fragmented market". However, the report surmises that "The key buyers will be taken as companies in the manufacturing sector", suggesting that manufacturing industry is the major purchaser of PLC systems. This claim also suggests a study of PLCs should emphasise industrial applications.

Although the PLC is in widespread use in manufacturing industry, little is known about the technology outside a small band of engineers and technicians.

Edgerton (2006) argued that technology is often seen in terms of invention and innovation, and defined by accounts of emerging technologies focused on dates. This "innovation centric" approach ignores the technologies that are widely used and, like the PLC, are largely unknown both the general public and academia. Such technologies, despite having widespread and significant uses, are overlooked and "hidden" from the limelight of historical accounts.

A particular problem with studying these "hidden technologies" that do not attract great public or academic attention, is the resulting dearth of formal historical records and accounts. The question is, therefore: how do you study them? With little documentary evidence in the form that is commonly referenced by historical academic studies, different methods and sources need to be identified. Alternative investigatory techniques together with traditional forms of research need to be employed thus a broad aim of this thesis is to address this task, and explore methods of studying hidden technologies.

1.1.2. What is a PLC?

A Programmable Logic Controller (PLC) is a highly reliable and robust industrial device employed on many industrial applications that include the control of manufacturing assembly lines, chemical processes, machine control and also non-industrial applications such as baggage handling, the control of lock gates and even fairground rides.

The modern PLC (Figure 1.1) is a specialised computer designed to interface with processes and machines via instruments, sensors and actuators. PLCs generally do not have a built-in human machine interface (HMI) or specific display other than indicators to reflect the status of inputs and outputs[2]; instead, displays and operator input devices are connected separately to the PLC. The PLC provides additional communications facilities that allow the connection of separate devices such as displays, operator input systems (e.g keyboards), complex sensors and instruments, other PLCs and computers using digital communication techniques.

[2] There are recent exceptions where PLCs and HMIs are combined in a single package, but they are treated as separate devices for programming purposes.

Figure 1.1 - Programmable Logic Controller

PLCs predominantly provide combinatorial logic and sequential control for processes or machines. Combinatorial logic performs logical operations on plant parameters that have binary states and operate devices with binary states. An example is an interlock that requires certain conditions to be fulfilled on the plant before the plant can be made operational; for instance a machine tool can only be started when the machine is turned on AND a guard is in position. Effectively the combinatorial logic forms a predicate from binary data on the state of the plant and its current evaluation determines the response of the controller that operates the plant.

Sequential control determines the order in which actions follow one another. Sequence control requires the controller to respond to the state of the product; which might be obtained by direct measurement (for example, detecting a billet of steel reaching an end of travel limit at a Steel Mill), or, where a meaningful measurement is not available, derived from the passage of time.

Real-time Control

Systems that control machinery and industrial plant generally require 'real-time' operation and responses. A simple definition of a 'real-time system' is 'Any system in which the time at which output is produced is significant'. The reaction time of the PLC or 'lag' between an input and its effect on an output is a key parameter of real-time control and this definition implies that it must be sufficiently small to be acceptable. 'Significant' really depends on the requirements of the process and Bennett provides an alternative definition that states:

> "A real-time system reads inputs from the plant and sends control signals to the plant at times determined by plant operational considerations" (Bennett, 1994)

In terms of software, Bennett (1994) defines a "real-time" program as:

> "A program for which the correctness of operation depends both on the logical results of the computation and the time at which the results are produced" (Bennett, 1994)

A real-time control system, therefore, is a system that responds to an input event by setting the output in an appropriately timely manner that is determined by the requirements of the process or machine under control. Processes can differ widely in their response requirements, for example: temperature control for a brewery may require real-time responses in minutes or even hours; high speed bottling lines or packaging systems may require responses in milliseconds. The time limitations of the computer system are set by the input and output processing times plus the control-program execution time.

The PLC is a real-time control system, but its processing and execution time limits performance, which therefore restricts the domains of operation; designs and implementations of PLCs that result in faster response times will lead to timely responses and correct operation for a wider range of control tasks and a wider market for the PLC manufacturer. One strand of the development of PLCs has therefore been to exploit techniques and components that produce quicker responses.

1.1.3. PLC applications

PLCs are acceptable for the control of process plant and machinery in the manufacturing industries because of their rugged design, straightforward interfacing for inputs and outputs with plant equipment and easily understood and programmed by manufacturing and electrical engineers and technicians (Bryan and Bryan, 1997, Clare et al., 2005, Bolton, 2006).

PLCs are used widely throughout manufacturing and this forms the focus of the study in this thesis. The application of PLCs in the manufacturing industries is discussed in Chapter 6.

Wider Application of PLCs

PLCs are used in a wide variety of situations requiring combinatorial and sequential control, including non-manufacturing applications that are perhaps not immediately obvious. Many airport baggage handling systems for instance, use PLCs extensively for airport conveyor systems control (Vickers and Chinn, 1998, Huiqun and Chunmei, 2010). Particular examples are found at Hong Kong airport, which in 1998 reported the

"World's largest integrated baggage handling system" utilising 68 PLCs (Forger, 1998) and more recently, Heathrow Airport's Terminal 5, using more than 200 PLCs for its baggage transport system (Derksen et al., 2007).

PLCs are also employed on other large commercial and utility applications: the Panama Canal lock gate control system, upgraded and operational in August 2005, incorporates over 800 networked PLCs connected to 80,000-90,000 input and output (I/O) points (James, 2006); Ship's engine and steering systems control (Krapf and Garschagen, 2005); the Channel Tunnel (Eurotunnel) which utilises more than 450 PLCs for the "control and monitoring of systems for distributing electricity to the catenary power lines and for lighting, air conditioning and ventilation, drainage" (Anon, 1994); fire detection and fire fighting systems (Pearse, 1995); radioactive waste handling systems (Rosli et al., 2011); and in the railway industry for the electronic interlocks used to control signals, points, line crossovers and level crossings (Pavlovic and Ehrich, 2010).

Less prosaic uses of the PLC can also be found in more unusual quarters. Fairground rides for example utilise PLCs for the control of roller coasters (Anon, 1996, Grogan, 2012). Perhaps even more unusual is the provision of the PLC control system for a "Power Nap Pod" found in America's Empire State Building, reported in the Times in 2004 (Allen, 2004).

1.2. Manufacturing and the Requirements for Control

The Datamonitor report (2011a) showed that PLCs are widely used in the manufacturing industries. Manufacturing however, is diverse. Many people have tried to devise ways to categorise manufacturing (Bessant and Haywood, 1988, McCarthy, 1995, Porter et al., 1999, Schonberger, 1999, McCarthy and Ridgway, 2000, Brandl, 2007) and the results are inconsistent.

Groupings and classifications of manufacturing are frequently defined by the industrial sector producing the manufactured products or materials (McCarthy and Ridgway, 2000). Examples include the oil and gas industry, pharmaceutical industry, white goods, automotive and so on. McCarthy (1995) suggests an alternative way of classifying manufacturing is by the operational characteristics of the movement, logistics and control of resources required for production. There are a number of manufacturing paradigms to be considered, but at the highest level, there are two categories – process (continuous) manufacturing and discrete manufacturing (Porter et al., 1999, Schonberger, 1999).

Process manufacturing is concerned with the continuous transformation of raw materials into the final product by chemical or physical means. Characterised by constant uninterrupted inputs of raw materials and outputs, this method of manufacturing is often defined as continuous production (Brandl, 2007). This requires control that stabilises the processing conditions, for example, by adjusting for raw material variations or variable external conditions. The control should also be adjustable to respond to changes in the rate of production. Discrete manufacturing is categorised by the production of unique or individual items by means of fabricating, machining or assembling the finished products or articles from raw materials.

A clear characteristic for differentiating discrete from process manufacturing is the fact that in discrete manufacturing, the final product can be broken down or disassembled to its constituent parts (Hitomi, 1996), whereas, a process manufactured item cannot be returned to its original raw material state (Brandl, 2007).

Process and Discrete manufacturing cover a wide range of industries but in practice, many manufacturing activities fall somewhere in between. The manufacture of some products require the processing of raw materials and then some form of packaging, this combines both process and discrete methods of manufacturing. For example beer is produced from the process of brewing, the output of which is divided and stored into discrete packages (kegs, bottles and cans). This highlights the need for a third category often called "Hybrid Manufacturing" (Kowalewski, 2002).

The PLC in Action

To illustrate how the application of combinatorial logic and sequential control can be applied to industrial process plant, an example of a reactor vessel is considered. Figure 1.2 shows a diagram of a PLC controlled reactor vessel which is one part of a hybrid manufacturing process.

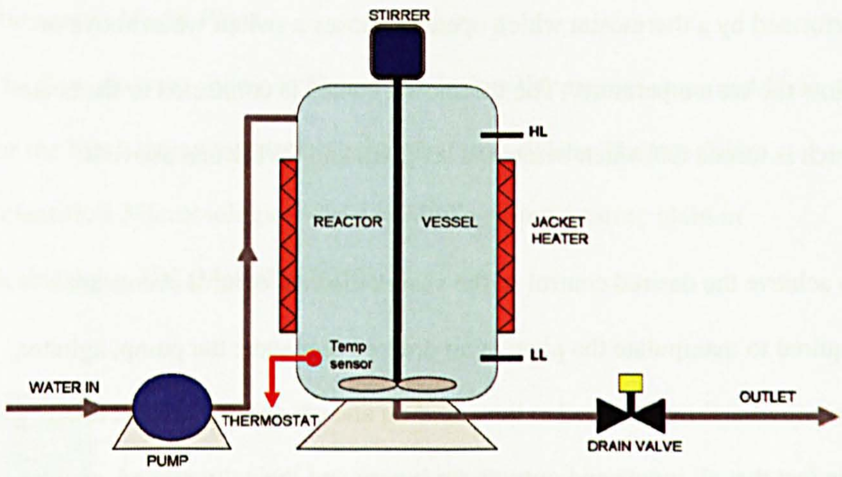


Figure 1.2 – Reactor Vessel Process Plant

A mixture of sequence and continuous control is required to operate the plant to make the manufactured product. The continuous (process) control aspects of the reactor vessel system relate to the heating and mixing of the raw materials added to the vessel. In this case, when the raw materials are mixed and heated, they cannot be returned to their constituent components and so reflect the characteristics of a continuous process. However, the contents of the vessel are filled in batches, processing conditions are established and maintained for a period of time and on completion of the process, the vessel is emptied leaving the vessel available for the next batch, requiring sequence and combinatorial control.

Generally speaking, sequential and combinatorial control is provided by a PLC. Process control often requires, perhaps in addition, other types of controller, which will not be discussed here. However in the example shown in Figure 1.2 the PLC is used for all control aspects because all inputs to the PLC and outputs from the PLC have a digital or binary state (True or False, ON or OFF). Temperature control, for instance, is performed by a thermostat which opens or closes a switch when above or below the set temperature. The thermostat switch is connected to the heater which is turned ON when below the set point and OFF when above it.

To achieve the desired control of the vessel, discrete control events are required to manipulate the plant in an ordered sequence: the pump, agitator, heater and drain valve need to be turned on and off and in the right order. The fact that all inputs and outputs are binary and logically related, enables the reactor vessel to be controlled by combinatorial and sequential logic, provided by a PLC. In this example, the PLC provides the combined process and sequence control of the plant. The detail of how this control is achieved by the PLC is discussed in chapter 2 using the same reactor vessel model (Figure 1.2).

Diagnostic Media Plant

The flow diagram shown in Figure 1.3 depicts the process steps and flows for the batch manufacture of biochemical materials at Thermo fisher Scientific's Microbiological Culture Media manufacturing plant in Basingstoke, UK (Metcalfe, 2012).

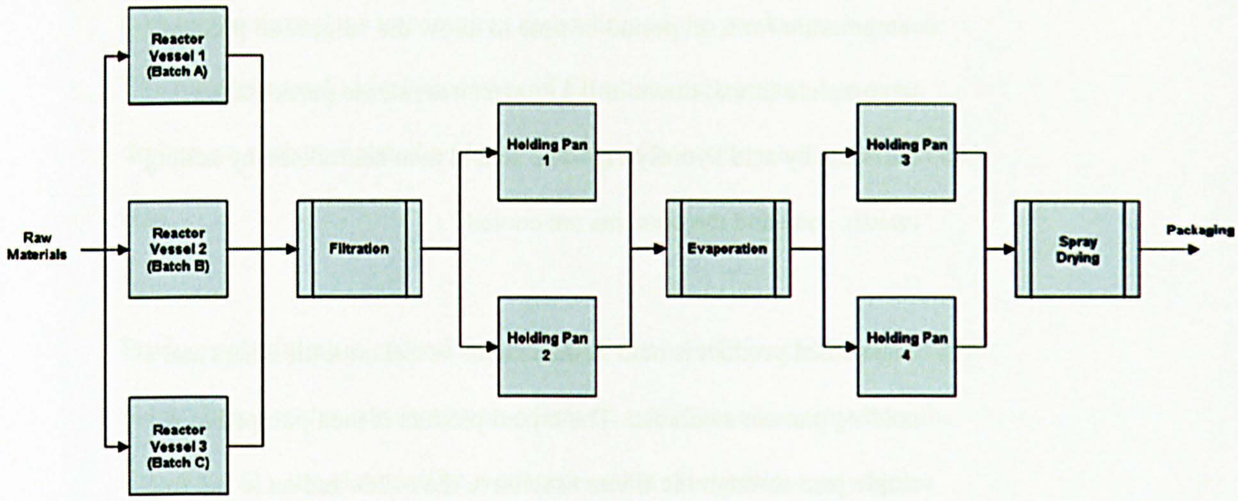


Figure 1.3 – Microbiological Diagnostic Media Manufacturing Plant process overview

The manufacture of biochemical materials such as microbiological diagnostic culture media is inherently a batch manufacturing operation[3]. Each batch shown in the reactor vessels (A, B or C) is potentially a different product in terms of the type and quantities of the raw materials used, and the required process parameters such as temperature and time. Each product is produced in a reactor and is then filtered, concentrated in an evaporator and then spray-dried before being packaged.

Batches are produced by the same plant by sharing resources such as the Filter Press, Evaporator and Spray Dryer, which are large and expensive

[3] Diagnostic culture media is designed to rapidly grow microbiological organisms (bacteria, yeasts etc), so has a relatively short shelf-life. Small batches of the media are manufactured to avoid stock expiry wastage.

plant. Figure 1.3 depicts the process flow of a plant that has four separate stages:

1) The raw materials for a batch of product are mixed and heated in a reactor vessel and are treated in a series of timed and measured operations. The contents of the vessel are held at a pre-determined temperature for a set period of time to allow the "digestion process" to complete (break-down of the enzymes to release particular nutrients by acid hydrolysis). The acid is then neutralised by adding caustic soda and the contents are cooled.

2) The cooled product is held in the reactor vessel until the filters and a holding pan are available. The mixed product is then pumped in a single pass through the filters to remove the solids and on to the free holding pan.

3) The next stage of the process is to remove some of the water held in the filtered liquid product so that it is reduced down to a thicker syrup consistency ready for spray drying. This is achieved in the evaporation plant once it and a holding pan are free.

4) The "syrup" is then spray dried to a powder and passed on for final weighing and packaging once the drier becomes available.

Each reactor vessel load is a distinctive batch of product consisting of raw materials and requiring a recipe (stored in the PLC) characteristic of the culture media being produced. Between each batch, the individual process plant (filter, evaporator and spray dryer) are cleaned down, as is each holding vessel to avoid cross-contamination of the product. Additionally, when a batch of product has left the reactor vessel, the vessel is cleaned and prepared for a new batch of product. Figure 1.3 shows a configuration for three possible batches of product that are initiated in each of the reactor vessels. The reactor vessel performs the longest duration process and can take one to three days to complete depending on the product being made. The remaining individual processes of filtration, evaporation and spray drying are each completed within a number of hours depending on the batch size.

Product scheduling is critical for the effective and efficient utilisation of the plant. Because the subsequent processes (2 and 3) take less time than the first stage (1), the plant can be managed so that it is continuously in use (with the exception of clean-downs). Invariably, all reactor vessels are in use, either processing or holding a batch of product or being cleaned.

The Diagnostic Media plant illustrates that the simple classification of this (or any) process as 'continuous' or 'discrete' is difficult. The actual process described is a 'hybrid' or mixture of sequence and continuous control:

a) The control system for the reactor vessel manages and controls the initial sequence of the process consisting of: filling, heating and mixing; maintaining the temperature; cooling; neutralising the acid; and coordinating the emptying of the vessel with other reactors, filtration equipment and the holding pans. Also for a period it has to precisely control the temperature.

b) The filter sequence synchronises the operation of the pumps and appropriate valves required for the product transition. Filter pressure is monitored and imposes a constraint on the rate of delivery from the pumps.

c) The evaporator control system continuously passes the product through the evaporation process to achieve a target viscosity in continuous closed-loop control of pressure (vacuum) and temperature. The control sequence also requires that an input is available (holding pans 1 or 2) and an output destination (holding pans 3 or 4) is available.

d) Finally, a single pass of the product from either holding pan 3 or 4 through the spray dryer is completed. The spray dryer requires continuous temperature and pump-speed control to ensure product consistency. The spray dryer sequence is governed by the level in the holding pans and whether an outlet for the spray dryer is available.

The cleaning cycles too require the control of pumps and valves in a suitable sequence.

The control of discrete events that include the opening and closing of valves, stopping and starting pumps and so on, are those that the original developments of the PLC aimed to manage.

Discrete control is combined with the 'continuous' temperature control of the reactor vessel using a PLC together with an electronic process controller. When the product in the reactor vessel has been controlled to maintain constant temperature and agitated for a set period of time (determined in the product batch recipe) it is transferred to the filtration process to either Holding Pan 1 or 2 according to which is available. The overall process for the batch of product then moves from a continuous mode of operation, where the temperature is maintained for a period, to a sequential, essentially discrete one (transfer line valves opened and the transfer pump started).

The final packaging stage is a discrete process where the spray dried culture media, now a powder is dispensed by weight to a number of individual containers which can be counted. The dispensing and packaging systems require combinatorial and sequence control (no continuous control required), provided by PLCs.

Temporal and Spatial Considerations

Processes can be organised either spatially, temporally or a combination of both.

Spatial organisation refers to processes that are divided into stages performed by different items of plant which feed one another in turn with all stages operational at any one time. The chemical process industry for example has spatially distributed processes, linked by a pipeline. There is no need for sequence control in a plant where things are spatially distributed and there is continuous availability such as chemical plant. The processing sequence is built into the arrangement of physical plant linked by pipelines.

Temporal organisation on the other hand has processes that are related in or by time because a machine or process can perform different tasks at different times. Flexible manufacturing systems have this attribute when multiple function machines are used to carry out the manufacturing process (Shewchuk and Moodie, 1998). Temporal organisation requires a flexible manufacturing system so that the same production equipment can change function.

Many manufacturing systems typically contain a variety of spatially and temporally distributed processes. The mix of spatial and temporal processes is particularly apparent with hybrid manufacturing systems. Machines and their control systems are programmed to carry out different tasks and functions to suit the manufacture of different products. The diagnostic media plant described in Figure 1.3 for example, uses the same reactor vessel but different times, temperatures and agitation parameters for each different product batch.

The difference between the spatial and temporal layouts dictates the function of the control system employed. In process manufacturing plant, there are requirements to control and monitor many analogue parameters such as temperature, pressure and flow, for large installations there are often hundreds of control loops to manage. Typically, mainframe computers or Distributed Control Systems (DCS) are used because they are inherently designed to perform continuous control and the process is spatially organised. Programmable Logic Controllers (PLCs) are popular control systems found in discrete manufacturing plant. Discrete processes require sequential and on/off control which is time and event based and therefore temporally organised.

The distinction between the traditional continuous process control systems and manufacturing systems employing sequential control (PLCs) is fading (Davis, 1992). Many manufacturing processes need to control sequences of operations in addition to maintaining process parameters as with continuous control. Hybrid manufacturing requires both continuous and sequential control attributes. Closed-loop control algorithms (e.g. PID control) have been developed for PLCs together with associated programming languages, allowing the PLC to perform continuous process control as well as handling logic expressions and sequence control. Similarly, process controllers have evolved so they incorporate the capability to provide sequence control. The selection of the appropriate technology is influenced by the dominant manufacturing category, be it continuous or discrete.

1.3. Engineering Cultures

There are many different disciplines represented in engineering. The major disciplines are mechanical, electrical, civil and chemical engineers. Furthermore, the engineering community is not just populated by professional engineers working alone, the space occupied by each of the disciplines is also shared with technicians and craftsmen, often organised in a hierarchy and all performing crucial and relevant roles.

Engineering disciplines tend to form their own cultures under a broad engineering umbrella; each discipline has its own methods, technical language and codes of practice. These cultural groupings form 'communities of practice'. Wenger et al explain that...

> "Communities of practice are groups of people who share a concern, a set of problems, or a passion about a topic, and who deepen their knowledge and expertise in this area by interacting on an ongoing basis... Over time, they develop a unique perspective on their topic as well as a body of common knowledge, practices, and approaches... They may even develop a common sense of identity. They become a community of practice." (Wenger et al., 2002)

There are many other influences that form different, distinctive but related sub-cultures. Maintenance engineers and technicians for example, can be drawn from any of the engineering disciplines although electrical and mechanical engineers and technicians dominate. There are also influences from local working organisational structures at the factory or installation level, to the wider influences from the particular industry sectors that engineers are employed such as the steel, petrochemical and automotive industries. "Every organization and industry has its own history of practice-based communities, whether formally recognized or not." (Wenger et al., 2002).

Some technologies have been adopted by specific communities of practice over long periods of time. Sequential control has hitherto been the preserve of production and electrical engineers and technicians and the PLC has been adopted by these communities. Although a specialised computer, the PLC has remained outside and remote from the prevalent computing and IT culture. For instance, the dominant programming language employed in PLCs was based on the 'ladder' wiring diagram notation used for representing configurations of relays familiar to the practising electrical and production engineer (Rullán, 1997) and adopted virtually wholesale as 'ladder logic' (discussed in Chapter 2).

The use of the ladder logic programming notation facilitated a rapid take-up of the PLC without requiring specific knowledge of computer programming techniques. Maintenance engineers and technicians were able to understand PLC technology and its programming language, ensuring that the new technology could be directly supported on the production line using existing knowledge and expertise (Wareham et al., 1988). Indeed, this was a key requirement, allowing the PLC to be employed as a direct replacement for the relay-based control cabinet without requiring the expertise of expensive computer programming personnel (Fauci, 1997).

1.4. Research Questions, Aims and Methodology

This broad question this research endeavours to answer is:

> **How did logic control, sequential control, and PLC technologies, develop in the 20th Century and why?**

And as a result of asking that question, it was necessary to study what proved to have the characteristics of Edgerton's "hidden technologies" and this leads to the second research question:

> **How do you study a hidden technology?**

The more specific aims of the research are therefore:

To explore the emergence and subsequent development of the PLC and to determine what factors influenced the development of the PLC.

To explore how sequential control technology developed in the 20th Century prior to the application of the PLC.

To investigate technological factors influencing the use and deployment of PLC technologies

To demonstrate and justify a method of studying a hidden technology.

Methodology

The research methods used to conduct the research have been influenced by the lack of academic material on the history of the PLC (see Chapter 3). In essence, academic focus has been directed to the history of control, in particular feedback and the development of computers and software. The history of sequential control and the technologies used to achieve it has not been at the forefront of academic research. This thesis attempts to address this shortfall in knowledge by using alternative sources of information, notably that found in patent material and first-hand interviews of PLC users. It is an approach to studying "hidden technologies" that could be applied to other 'hidden' technologies.

Patents

Patents were chosen as an information resource because they provide a method of tracing technological developments; they are particularly useful in identifying the incremental improvements of a technology. Although patents are primarily commercial and legal documents, they are assessed by external examination bodies, adding a certain level of independence and reliability, although the focus is on legal protection. Another strength of using patent material is that they record the invention at the time it was introduced, so are not retrospective in nature. The links back to "prior art" recorded in earlier patents and published material, provide a useful method of identifying and tracing incremental technological progress.

At the time of writing this thesis, PLCs have been in use for over 40 years and earlier sequential control technologies considerably longer. Patents provide a rich source of information that cover, in some detail, the period of time that concerns the development of PLCs and its influences. They do, however, have certain limitations due to their commercial nature and the fact that they refer to *potential* concepts and not necessarily *viable* innovations. Hence in order to support the patent findings, interviews were used to corroborate the commercial use of the patented technology.

A detailed examination of the patent search and analysis methodology used in the research is presented in Appendix A.

Interviews

Interviews were conducted to investigate the impact of cultural diversity on the development and use of the PLC in a number of manufacturing related environments.

The interviews recorded recollections from a number of senior engineers and employees associated with using PLCs and automation technologies from the 1960s onwards. The purpose of this approach was to establish the influences and aspirations of the end users, and ascertain what their experiences were regarding the development of sequential control technologies. A set of broad questions were prepared for guidance (see Appendix B).

However, open ended interviews extract a richer viewpoint from interviewees (Yin, 1994, p84), so the interviews were allowed to develop as the interviewees recollected anecdotes from their working past experiences. The interview process was designed to reveal information that may not have been considered by the researcher and as such provide a wider insight to the development and use of PLC and associated technology.

A specific aim for the interviews was to gain an understanding of the industrial and professional cultures that exist in different industrial sectors to determine industry specific preferences and motivations.

1.5. Overview of the Thesis

Chapter 1 – Introduction (This Chapter)

Chapter 1 has introduced the PLC as a technology: what it is, and where it is used. The broad classification of manufacturing types was discussed and it was claimed that attempting to classifying manufacturing types is a difficult and inexact process ultimately of debatable value. Many manufacturing methods fall into a mixture of continuous and discrete manufacturing and are often applied as a hybrid of the two. The Diagnostic Culture Media plant was used as an example to show how the manufacturing processes contain the continuous and discrete elements along with the temporal and spatial considerations. Industrial and engineering cultures were introduced and it was explained that these could have an influencing effect on the development and use of the PLC.

The research questions, aims and the justification for the research were stated. In essence, there is a gap in knowledge relating to the historical development of combinatorial and sequential control, and the PLC. The research aims are to address this gap. One broad aim of the thesis is about studying a hidden technology, exemplified by the PLC. The methodology used for constructing the patent search and analysis and the interview techniques used in the research were outlined.

Chapter 2 – PLC Technology

Chapter 2 discusses the technology of the PLC and explains its unique features and attributes that differentiate it from other computers and control technologies. The fundamental concepts behind the PLC's architecture and operation are explained. A detailed examination of the programming concepts and languages are presented and examples are used to explain how the PLC processes and interprets program instructions. The last section of this chapter discusses the manufacturers of PLCs and the impact of, and requirements for standards.

Chapter 3 – Literature Review

This chapter reviews the academic literature concerning the history and development of sequential control and the PLC. Literature pertaining to the history of control and the history of computers is reviewed and the gaps in knowledge are identified and discussed. Alternative sources of literature such as the commercial and trade press are also examined to reveal the current state of knowledge regarding the history of sequential control and the history of the PLC. Finally, the social impact and influences of automation technologies are considered.

Chapter 4 – Developments in Sequential Control (1900-1969)

Chapter 4 examines the development of sequential control technologies through the examination of patent literature. The chapter covers the development of electro-mechanical sequential control systems through to the incorporation of electronic devices and finally the application of computers to solve sequential control problems.

Chapter 5 – Emergence and Development of the PLC

Patents are again used as a source of information to review the emergence and subsequent development of the PLC beginning with the first "Programmable Controller". The chapter also reviews the use and application of general-purpose computer technologies and the development of purpose built controllers. Technologies that were incorporated into PLC design are identified and discussed.

Chapter 6 – Industry and Engineering Practice

This chapter provides an analysis of the interviews conducted throughout the period of research with practising engineers. Chapter 6 discusses the engineering and industry cultures found in the world of the control engineer. The correlation between the interview material and the patent findings are identified, discussed and used to corroborate key aspects in the development of the PLC.

Chapter 7 – Conclusion

The final chapter of the thesis draws conclusions from the results and findings of the research. The research questions are considered and the limitations and shortcomings of the thesis are identified and discussed with regard to how these can be addressed in future research. Finally comments are made on the contribution of the research to the history of technology.

Chapter 2 – PLC Technology

2.1. Introduction

The PLC has been introduced as a robust and reliable device that is ubiquitous and employed to control many disparate processes and machines. The aim of this chapter is to describe the technology of the PLC in greater detail in order to draw out its distinguishing features and describe how the PLC functions.

The first section examines the technical attributes of the PLC that enable it to exert real-time control, in sometimes, harsh industrial environments. In order to control these processes and machines, the PLC interacts with devices that include plant instrumentation, sensors and actuators and the mechanisms to achieve this are explained. The second section discusses the programming options available for the PLC. Engineers need to program or configure PLCs in order to perform their desired control actions and this is achieved in a unique way. Programming notations that are designed to be familiar to practising engineers and technicians are introduced and discussed using a simple control example. The final section of this chapter considers PLC manufacturers and the requirements for standardisation.

2.2. PLC Architecture

The PLC provides timing, combinatorial logic and sequential logic control for the automation for industrial (and non-industrial) machines and processes. Unlike a general-purpose computer, the PLC is a robust device, able to withstand harsh industrial environments and designed to work on the factory floor. Although programmable, it is designed to run a single, distinct, dedicated procedure, specific to the application process and specialised to perform control related functions.

The PLC is designed to facilitate the connection of the plant's sensors, instruments, and actuators through its I/O system. A modular approach is adopted where inputs and outputs are grouped into exchangeable units known as "cards" or "modules". For most PLC systems, inputs and outputs (I/O) are separated onto input only and output only cards[4]. The I/O cards are inserted or "slotted" into a "rack" (backplane) that facilitates the connection to the PLCs modular central processor unit (CPU) for addressing and data connectivity.

Figure 2.1 shows a block diagram of an early PLC, developed and adapted from the diagram format presented in Nakao et al's (1976) "Programmable sequence controller" patent. The diagram is concerned only with binary inputs and outputs and depicts the primary functions and components of a conceptual PLC.

---

[4] Some PLC manufacturers provide cards or modules providing mixed I/O on the same device, generally found on smaller PLCs.

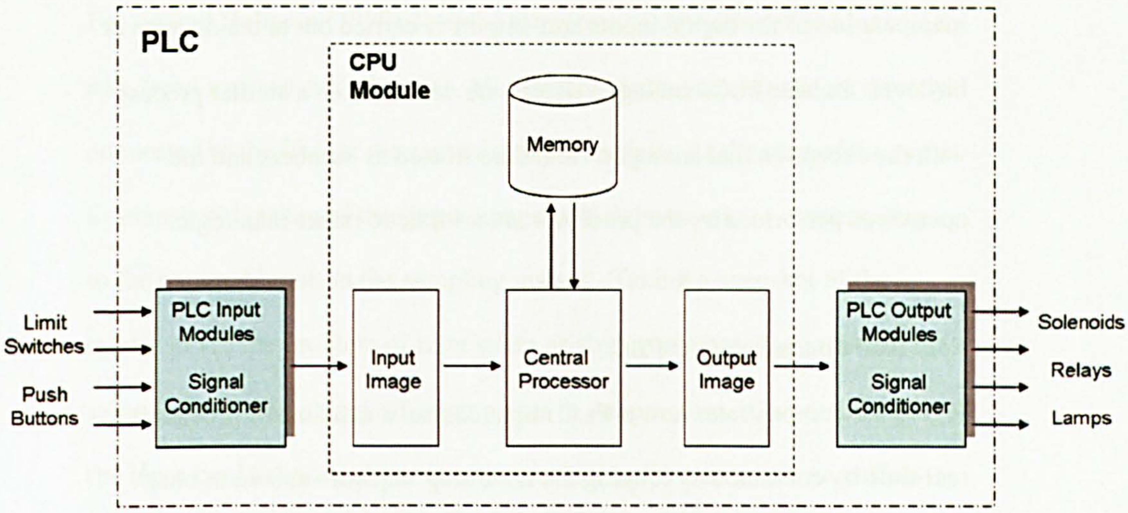


Figure 2.1– Conceptual diagram of a Programmable Logic Controller

Input signals are received by the PLC via the input modules which are conditioned and converted to digital data values compatible with the Central Processor (CPU) module. All digital inputs and outputs are buffered to protect the central processor from external signal transients and to allow the processor to work independently of the I/O read-write and signal conditioning time constraints. Digital inputs form an "input image" which is stored in the buffer.

The memory holds a fixed operating system which incorporates an interpreter that interprets the "control program" written by the user. The control program determines the logical operation performed on the state of the inputs, outputs and any counters or timers maintaining a record of the progress of the process under control. The input data and control program determine how the central processor is to set the outputs. The output results are stored in a buffer and referred to as the "output image". The output modules, convert the output image into electrical control signals that operate valves, motors or other devices on the plant or machine under control. Data manipulation of the digital inputs and outputs is carried out at the discrete or bit-level. In later PLCs analogue signals are dealt with in a similar process with the exception that analogue values are treated as numbers and the operations performed by the processor are arithmetic rather than logic.

Real-time Control and the PLC

Figure 2.2 demonstrates how a PLC might control a machine or process in real-time by continuously looping the three step sequence shown in blue[5].

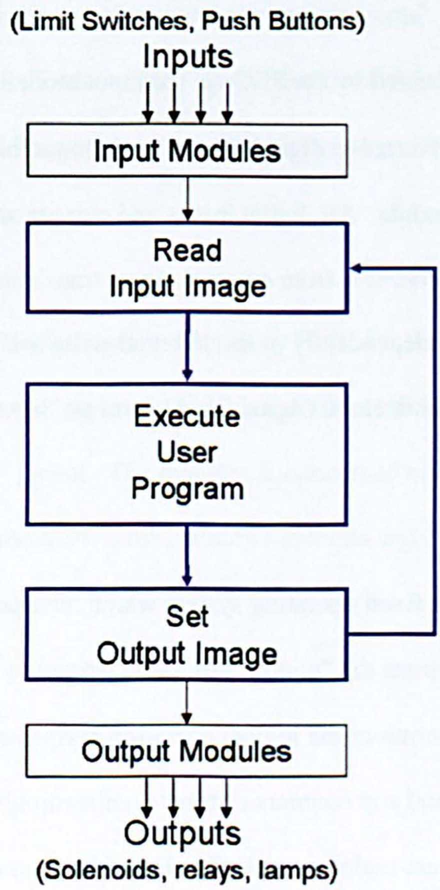


Figure 2.2 – PLC Program Cycle

[5] Some systems may be interrupt driven and would not be modelled in this way. However, the early PLC patents do propose this cycle.

Fabian and Hellgren (1998) describe the process demonstrated in Figure 2.2 as a "read-execute-write" cycle. All external inputs and outputs are directly connected to the PLC at the same time (in parallel). At the start of the cycle a 'snapshot' is taken of the 'input image' which has a one-to-one mapping to the external inputs at the sampling instant. Taking a snapshot of the inputs solves the problem of false states arising from changing inputs. If the same input is read at two different points in the program, it is possible that the input could change state between processing the two points and affect the output. The snapshot ensures that all inputs are consistent throughout the course of the cycle.

The control program is then executed in its entirety from start to finish using the data obtained from the input image. The results obtained by the executed program are saved as an 'output image', having a one-to-one mapping to the external outputs. Finally, the output image is written to the PLCs physical output interfaces. The whole process is then repeated in a continuous cycle known as the "scan cycle" (Fabian and Hellgren, 1998).

Strictly speaking, the cyclic program conforms to Bennett's (1994) definition of "classical sequential programming" where "actions are strictly ordered as a time sequence". The PLC processes the program in a strict unaltered sequence and all actions are carried out: the PLC reads the inputs and applies the entire fixed length program, then sets the outputs accordingly.

All of these actions are fixed in terms of duration and determine the 'scan-time' of the PLC. Although the scan time is a function of the program length, and assuming that no conditional jumps are used in the logic, its duration remains fixed and therefore so does the response time, making the PLC a 'real-time' control device. It is this method of processing that allows a sequentially processing digital computer, to emulate the parallel processing of the original relay circuit it was designed to replace.

In a practical application, there may be some exceptions. The program may represent an elaborate logic function, which in the processor, is specified and evaluated in a series of simple steps. Thus the time to fully evaluate the function will depend on the number of steps and the speed at which the processor can execute each step. There is therefore the possibility that the time to evaluate the logic or analogue functions will be excessive. The consequences of this are that real-time control cannot be achieved according to Bennett's definition because the response time is limited by the computer system and will be too slow for the process or machine under control. In order to circumvent this situation, the cycle time has to be speeded up (faster processor) or some functions executed outside of the cycle time by using interrupts.

Robustness and Security

The PLC is a single-function computer, dedicated to the control task, designed to work in harsh environments unsuited to general-purpose computers (Walker et al., 2010). Located in environments that are potentially hot, dusty, wet and contain large electrical machines (motors), for example steel works and chemical processing plant, the PLC is resilient to wide temperature fluctuations, vibration and electro-magnetic interference (Clare et al., 2005, Sudhir and Sujata, 2011). Frequently located within robust cabinets or 'enclosures' (see Figure 2.3), or applied directly on the plant or to the machine under control, the PLC is often hidden from general view. Although out of sight, the PLC is vital for running the plant and is a key component in many controlled industrial systems.

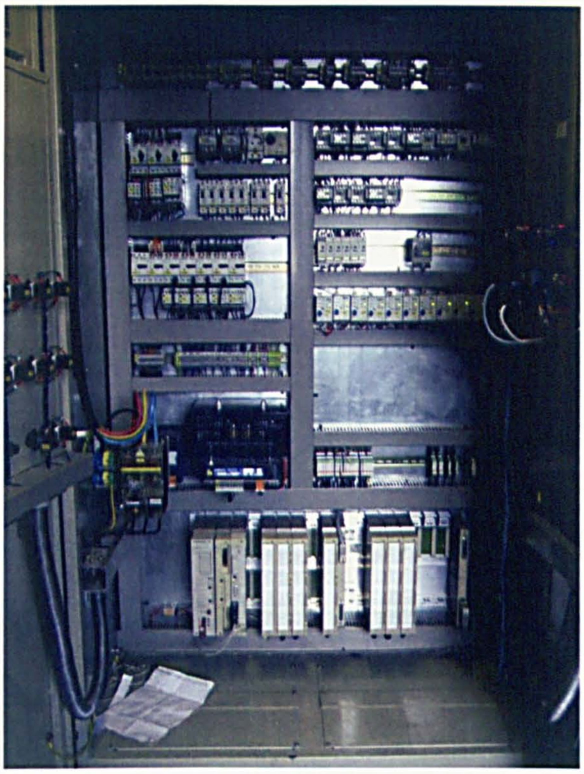


Figure 2.3 - PLC (bottom) located within an enclosure.

Recent "cyber" security threats such as the "Stuxnet" virus have targeted PLC automation systems (see Constantine, 2011, Gross and Karlsson, 2012) but traditional control system security measures have been more mundane. According to McKay (2012) ...

> "Security was accomplished by locks on the plant doors and guards at the gates. Even if someone were to gain physical access to the system, the tools, methods, and procedures used were so specialized that no-one outside the control engineering community would be able to make much sense of what they were seeing. Security was accomplished by physical perimeter security, air-gapping, and 'security by obscurity'." (McKay, 2012)

A common solution used to prevent unauthorised physical access is to house the PLC in a locked enclosure or cabinet. Additional security is achieved because the PLC commonly does not have the ready means to alter the program in situ; a separate programming device or tool has to be used.

2.3. Programming and Logic Notations

Control programs are commonly written for the PLC on an external and detachable programming device or "tool" and are downloaded to the PLC's memory. Modern PLCs, connected to local networks and the internet, also allow the remote upload and download of programs. Control programs for the PLC can be written in a number of different programming languages specified in the standard IEC 61131-3 (BSI, 2003b), although not all manufacturers support all of the programming languages listed there.

There are two textual languages: "Instruction List" (IL) – a low-level language similar to Assembly Language; and "Structured Text" (ST) – a high-level programming language, similar to the Pascal and BASIC programming languages. There are also two graphical languages: "Function Block Diagram" (FBD) – based on electronic logic-gate symbols; and "Ladder Diagram" (LD), sometimes referred to as "Ladder Logic" – based on electrical relay wiring diagrams. A fifth language, although not specifically a programming language, is "Sequential Function Chart" (SFC) which is a graphical method of structuring the internal organisation of PLC programs based on GRAFCET (BSI, 2003b).

Ladder Diagrams

Ladder logic was the original programming notation used with PLCs. The notation is still widely used to date with an estimated seventy percent of PLC programs written in the ladder language in 2007 (Molina et al.). Ladder logic notation is derived from the "ladder diagrams" used to represent electrical wiring circuits containing electro-mechanical relays. Wareham et al (1988), Adamski and Monteiro (1996) and Rameback (2003) noted that the resemblance to electrical relay circuits and wiring has meant that the PLC is used predominantly by electrical engineers and technicians.

Figure 2.4 shows a simple electrical circuit for a D.C. motor control. The motor is started by the spring-loaded 'Start' push-button which closes the initial circuit for the motor relay. One switch contact of the motor relay (RL1a) turns the power on to the motor; a second contactor on the motor relay (RL1b) is used as a 'latch' to hold the motor relay on when the start button is released. To stop the motor, the 'Stop' push-button, a normally closed switch, is depressed breaking the circuit and de-energising the motor relay, also opening the latch circuit. The relay RL1 in this example is used as a power amplifier to switch a motor on or off within a high-voltage DC circuit.

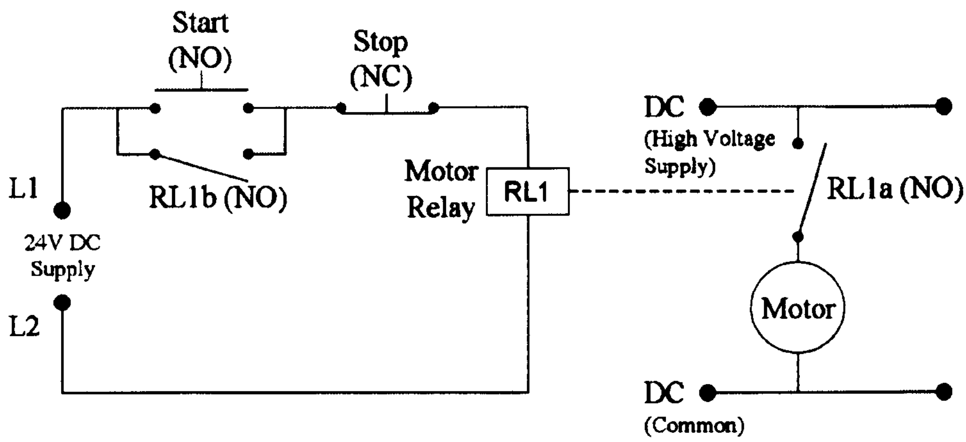


Figure 2.4 – Simple motor latch circuit

The ladder diagram equivalent to the circuit diagram is shown in Figure 2.5.

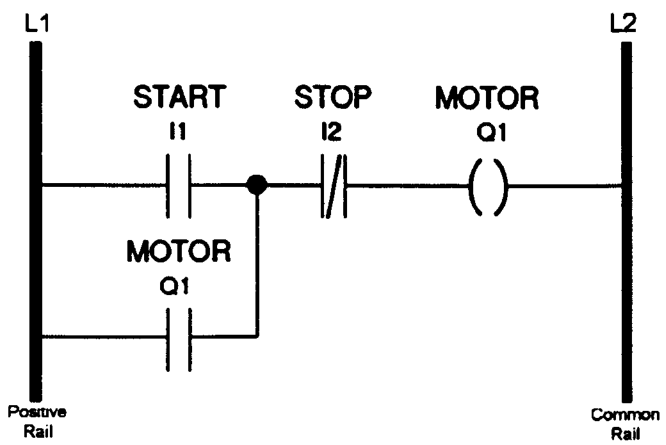


Figure 2.5 – Ladder diagram equivalent of the motor latch circuit

The symbol for a normally open (NO) switch or relay contact is shown by the '-I I-' symbol and a normally closed (NC) is shown by '-I/I-'. The ladder diagram performs exactly the same function as the electrical circuit but the power rails, Line 1 (L1) and Line 2 (L2) are shown vertically, forming the uprights of the ladder. The switches and relay contacts are connected from left to right on the diagram and the circuit is made when there is a clear path from the left-hand rail to the right. This is known as a 'rung'. A program with many rungs gives the appearance of a ladder (see Figure 2.7) and so this notation has become known as *Ladder Logic* or *Ladder Diagrams*. The ladder diagram representation of electrical relay circuits has enabled the PLC to be used directly by electrical engineers and technicians without requiring computer programming skills or knowledge.

The PLC generally performs the logic functionality represented in the ladder diagram. Unlike the electromechanical relay equivalent, it does not generally provide power amplification. In this example, the output Q1 would be used to energise another electromechanical relay to provide the necessary signal amplification in order to switch the motor on or off.

To demonstrate how sequence control is described using ladder logic, the plant shown in Figure 2.6 (a repeat of Figure 1.2) is used.

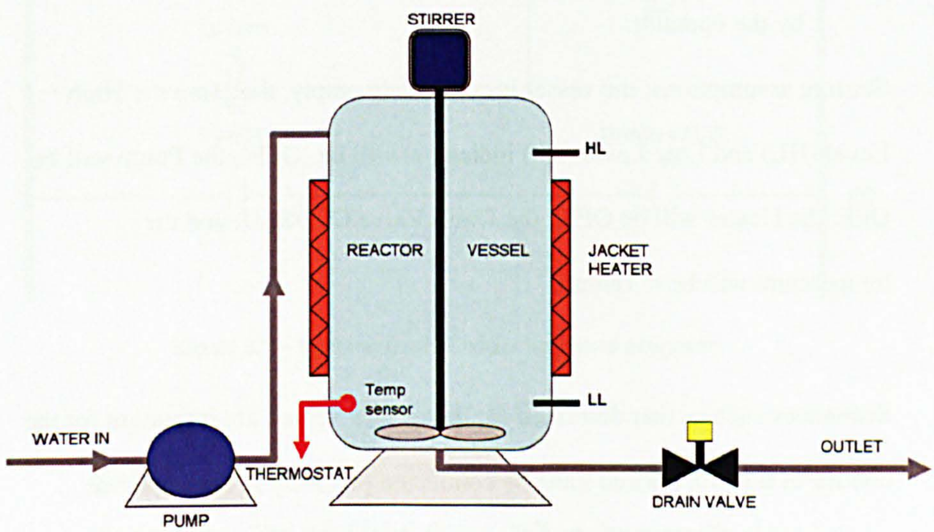


Figure 2.6 – Reactor vessel process diagram

The reactor vessel is required to fill, heat and mix its contents. After holding the temperature for a set time, the control system is required to automatically drain its contents and reset the vessel ready for the next batch. Active components of the vessel include: a water inlet, driven by an electric motor driven pump; a motor driven stirrer; a wrap-around electric jacket heater; a valve controlled drain to empty the vessel; a temperature sensor and two level sensors (high and low).

The process is completed in five steps:

(1) Fill the vessel with water and reagents;

(2) Heat and mix the contents of the vessel to a set temperature (Temp1);

(3) Hold the contents of the vessel at a fixed temperature for a set length of time (T1);

(4) Empty the vessel switching off the stirrer and heater when empty.

(5) The vessel should not automatically refill the vessel unless requested by the operator.

Starting assumptions: the vessel is completely empty, therefore the High Level (HL) and Low Level (LL) indicators will be 'OFF'; the Pump will be OFF; the Heater will be OFF; the Drain Valve CLOSED; and the temperature will be < Temp1.

Sequences such as that described for the reactor vessel, are important for the control of the process and must be conducted precisely in the sequence described in steps (1) to (5) and to avoid potentially damaging conditions. For example, running the stirrer whilst the vessel is empty could cause irreparable damage the stirrer bearings. Similarly, heating an empty vessel could burn out the heating elements because the liquid contents prevent the elements obtaining excessive temperatures. In practice, additional safety systems would be used in conjunction with, and alongside a PLC, to prevent unsafe conditions from arising. A ladder diagram used to achieve the control requirements is shown in Figure 2.7.

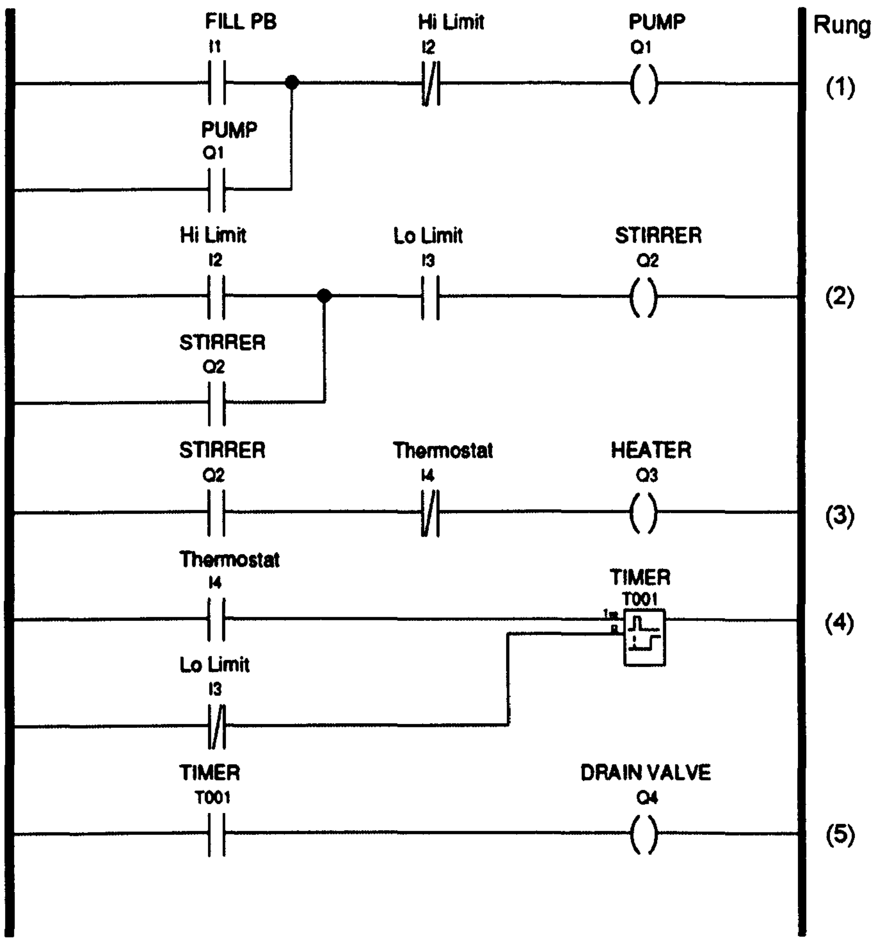


Figure 2.7 – Reactor vessel Ladder sequence program

Rung (1) – External 'Fill' pushbutton (I1) is momentarily pressed by the operator. The 'Hi Limit' (I2) is OFF so the normally closed (NC) contact completes the circuit made to the pump relay (Q1) which switches ON. The Q1 contact latches the pump circuit when the 'Fill' pushbutton is released. Water fills the vessel, first closing 'Lo Limit' (I3) until the level rises to the 'Hi Limit' which opens (NC) I2 and switches the pump OFF.

Rung (2) – Stirrer (Q2) starts when the normally open (NO) contact 'Hi Limit' (I2) is reached (Lo Limit already closed as per previous statement). Q2 provides the circuit latch for the stirrer. The Stirrer is switched OFF if the level falls below the Lo Limit (I3 OPEN).

Rung (3) – The Heater (Q3) is enabled by the stirrer (Q2) running and the Thermostat (I4) is < Temp1 (I4 closed). Temperature is maintained by the Thermostat which opens and switches off the heater (via I4) when Temp1 is reached.

Rung (4) – When the temperature has been reached for the *first time*, the timer is started (single event). The Timer is RESET when the vessel Lo Limit (I3) level indicator (NC) is enabled.

Rung (5) – At the end of the set time period (hold temperature time), the output of the timer OPENS the Drain Valve (Q4) to empty the contents of the vessel. When the level passes the low limit switch (I3 OPEN), the timer is RESET (Rung 4), the stirrer is stopped (Rung 2), the Heater is disabled (Rung 3) and output T001 closes the drain valve.

The system is reactivated again by the "Fill" pushbutton. As the vessel fills the Low Limit switch is made (ON) etc.

Ladder diagrams are read from left to right, for the elements in each rung, and rungs are read from top to bottom as shown in Figure 2.8. All rungs in the complete ladder diagram are processed sequentially by the PLC in the same way and this method enables PLCs to be used for sequence control applications. Sequential control is achieved by the ladder program because the completion of a previous event enables or disables a succeeding ladder rung. The program is processed sequentially and is continuously repeated or 'cycled'. In this way, the PLC can emulate the equivalent relay circuit which 'processes' all relay contacts at the same time, or 'in parallel'.

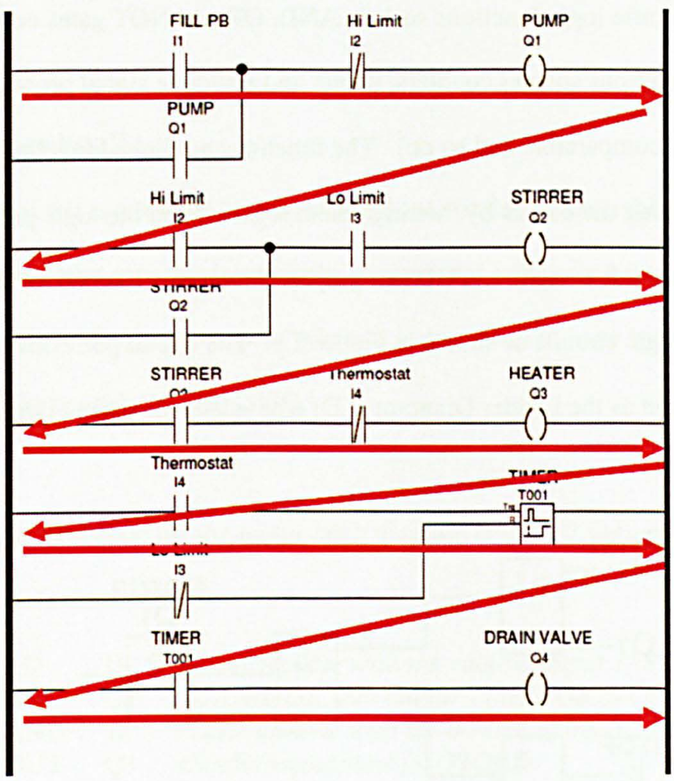


Figure 2.8 – Ladder program sequence

The cyclic process of read and store inputs, process the ladder program and store and write outputs typically takes 10 to 50 ms per cycle (Bolton, 2006) and is repeated continuously. The cycle time depends on the program length and the PLCs processor speed. The cyclic processing method enables the PLC to emulate the relay circuit's parallel process because the cycle time is faster than, or at least comparable to the response time of the relay.

Function Block diagrams (FBD)

According to Lewis (2004) "FBD is widely used to represent continuous control activity and depicts a control program as a network of connected function blocks". Function blocks are organised units of code, written in any of the other languages defined in IEC 61131 that are interconnected to perform specific logic functions such as AND, OR and NOT gates or more complex functions such as counters, timers and analogue signal processing (PID loops, comparators and so on). The functions are linked together by interconnecting the blocks by "wiring" them together and the logic process are read from left to right. . FBD is a graphical language likened to electronic logic circuits as shown in Figure 2.9. The circuit performs the same function as the Ladder Diagram (LD) equivalent shown in Figure 2.5.

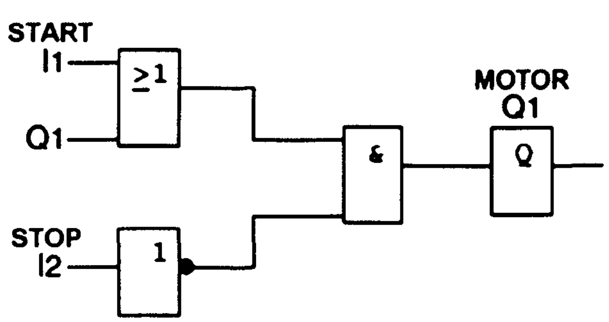


Figure 2.9– FBD program representation of the motor latch circuit

(Key to symbols: '≥1' = OR, '&' = AND, '1•' = NOT and 'Q' = OUTPUT)

In a letter to 'Electronics and Power', Barrow (1976) observed that the "the Europeans (including UK) and Scandinavians have entered the programmable controller suppliers league; the majority, if not all, of whom have adopted a Boolean and/or flowchart [FBD] programming language". However, the Ladder Diagram notation was still popular and used extensively in the UK. The Ladder and "flowchart" programming languages each had their advantages explained Barrow, but "flowcharts are understood by mechanical and process engineers as well as electrical engineers". Lewis (2004) agrees with this notion stating that "FBD is provided by most of the European PLC manufacturers. Hardware designers and control engineers prefer it as it fits well with a data flow view of a system".

Instruction list (IL)

Instruction list is a textual low-level language similar to Assembly Language where mnemonics are used to represent processor commands. This low-level language is used "where it is necessary to have compact, time-critical code" (Huuck, 2005).

The equivalent code for the motor latch diagram (Figure 2.5) is represented in IL as follows:

```
LD    I1   <Load register with the value of input 1 (START)>
OR    Q1   <Logical OR with output 1 (MOTOR latch contact)>
ANDN  I2   <Logical AND with the inverse of input 2 (STOP)>
OUT   Q1   <Set/Reset output 1 (MOTOR)>
END
```

The efficiency of the interpretation of the code written in Instruction List (IL) has made it a useful candidate for writing specialised functions. Molina et al (2007) observed that "It is better suited to solve problems that deal with mathematical algorithms, or to process data intensively" As a consequence IL "is very popular amongst programmers accustomed to low level languages like embedded systems developers". Functions are reusable blocks of code that are used to perform specific tasks such as motion control, PID loop control, panel displays and so on. All of the PLC programming languages (LD, FBD, IL, ST and SFC) can call functions in their execution.

Structured text (ST)

Structured Text (ST) is a high-level programming language that "strongly resembles Pascal" (Bolton, 2006). IEC 61131-1 (BSI, 2003a) defines Structured Text as "A textual programming language using assignment, sub-programme control, selection and iteration statements to represent the application programme for a PLC-system". The equivalent motor latch circuit program is represented in ST below:

```
IF (I1 = 1 OR Q1 = 1) AND I2 = 0 THEN
        Q1 = 1
ELSE
        Q1 = 0
End_IF
```

As with many high-level languages, the construction of an ST program consists of declarations, expressions and statements and according to Molina et al. (2007) is "attractive to computer science programmers, and it is better suited to solve math or algorithmic problems".

Sequential Function Charts (SFC)

IEC 61131:1 (BSI, 2003a) defines Sequential Function Charts as "A graphical and textual notation for the use of steps and transitions to represent the structure of a program organization unit (program or function block) for a PLC-System". In essence, SFC elements are used for structuring the sequences and organization of programmable controller programs and function blocks. A Sequential Function chart is not a specific programming language but is a "pictorial representation of a system's operation to show the sequence of events involved in its operation" (Bolton, 2006). Figure 2.10 shows a partial SFC representation for the sequential control of the Reactor Vessel (see Figure 2.6).

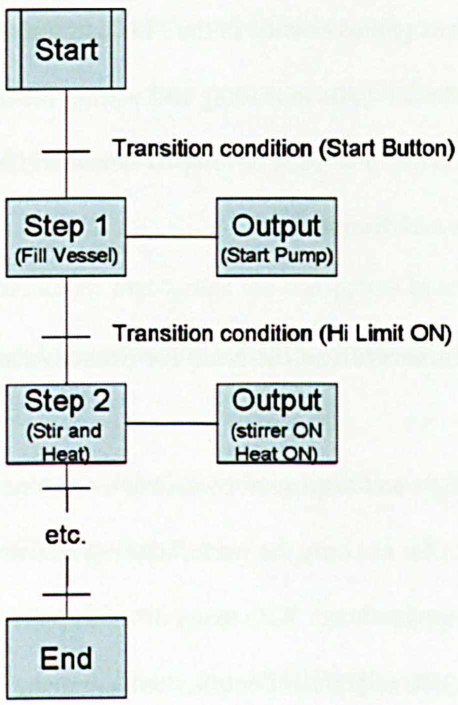


Figure 2.10 – SFC example

Sequential Function Charts are derived from GRAFCET (Adamski and Monteiro, 1996, Lewis, 2004) and are used to control the sequences of functions written in the LD, FBD, IL and ST programming languages[6].

The different programming languages reveal some important cultural aspects relating to PLC technology. Electricians for example have a tendency to use the ladder logic notation because of its resemblance to electrical and relay circuits. Computer programmers on the other hand may prefer to use the computer language based Structured Text (ST). Electronic engineers may gravitate to the Assembly Language equivalent Instruction List (IL) and process engineers Function Block Diagrams (FBD). The cultural divide in using the different PLC programming languages appears to relate to experiences gained outside of the PLC community. Each notation requires different skills in reading and writing the programming language, and these skills have been developed elsewhere, in different engineering cultures and disciplines.

2.4. PLC Manufacturers and the Need for Standardisation

The combinatorial logic and sequential control of a machine or process, is a common requirement for not only the manufacturing industries but also non-manufacturing applications. With many diverse applications for technologies that require sequential control, there are many end-users of PLCs (Datamonitor, 2011a). It is not surprising therefore that a large number of PLC manufacturers exist in the market.

[6] Although not discussed further in this thesis, a more in-depth explanation of Structured Text and Sequential Function Charts used with PLCs is given in BS EN 61131-3:2003 Programmable controllers — Part 3: Programming Languages .

The wide number of applications for PLCs has attracted control system manufacturers and suppliers from diverse backgrounds, arguably because of market opportunities. Many of the recent PLC manufacturers developed from the following control-related industries:

1) Manufacturers who produce control equipment without computers. These include electrical components (for example relays and switches) and electronic control systems (logic gates and circuits, analogue process controllers). Examples include Allen-Bradley and Siemens.

2) Organisations that produce ruggedised computers (including mainframe computers, mini-computers and industrial PCs), for example Modicon and DEC.

3) Software producers that turned the computers to sequence controllers (Modicon and National Instruments).

4) Companies that produced the whole control system (for example PLC and DCS manufacturers)

Some companies such as Siemens and Allen-Bradley were already manufacturing electrical switchgear and components used for control and automation purposes. The adoption and development of PLC technology was a natural development because they were aware of and able to further exploit the existing market for sequential control technologies. Modicon, on the other hand were more opportunistic, developing a rugged computer control system without pre-existing knowledge of the wider control and automation market and targeting particular applications in the automotive industry. At the time of writing, Modicon, now a brand of Schneider Electric (an electrical systems manufacturer), and Allen-Bradley (now part of Rockwell Automation) are still significant PLC and control system manufacturers.

The resulting diversity in the automation market and the PLC manufacturers has resulted in a lack of standardisation in PLC technology. It was difficult to mix different manufacturer's PLC systems on plant because of incompatibilities between hardware and software, therefore many users became locked in to single supplier PLC systems (Thoma, 2009). For example I/O devices used different standards and could only be used with a single manufacturer's system. Similarly, programs could not be transferred between different manufacturer's PLCs.

The need for standards was recognised by the users of PLCs to overcome these difficulties and two primary standards emerged: IEC 61131 (BSI, 2003a), introduced in 1993, for Programmable Controllers which primarily related to the standardisation of the software languages used; and IEC 61158 (BSI, 2004), introduced in 1999, relating to control system communications – Fieldbus. The impact of these two standards is discussed in Chapter 6.

One standard that has met with some acceptance across industry, defines analogue signals for process control systems. Continuous input signals, corresponding to analogue process values, are represented as thermocouple, resistance and current signals (e.g. 4-20mA); output signals are generally represented as 4-20mA current signals (Agashe and Agashe, 2011). The 4-20mA signal format is commonly used to represent analogue input and output values and has been widely adopted, enabling PLCs to connect to any instrument or sensor complying with the standard (later published as IEC 60381-1:1982).

2.5. Chapter Summary

This chapter has discussed the technical features of the PLC that enable it to function as a real-time control system. The PLC is a modular device designed to accommodate input and output signals from sensors, instruments and actuators. The general concepts of the hardware architecture were explained detailing the input and output images, and the general flow of data and processing through the architecture of the PLC. Bennett's definition of real-time control was developed by discussing how this is achieved with the PLC. The security and environmental aspects of PLC technology were considered.

Aspects of the PLC's Programming and logic notations were reviewed in some depth. Particular emphasis was given to the ladder logic notation using a simple motor latch circuit example. The same motor latch example was used to explain alternative programming notations based on the IEC 61131 standard. A reactor vessel process example was used to develop the ladder notation and explain how the PLC achieves sequence control.

The final section of this chapter explained that the market for PLC technology was large and diverse and gave an account of the effect of this market on PLC manufacturers. The resulting diversity in PLC manufacturers, each with their own version of the PLC, has prompted the requirement for standards. This user led requirement has met with limited success, with the exception of the analogue signal representation.

Chapter 3 – Literature review

3.1. Introduction

This chapter reviews the academic and technical literature associated with sequential and combinatorial control and its related technologies. Edgerton's (2006) "use-centric" approach is discussed with regard to the development of the programmable logic controller (PLC). In contrast to Edgerton, the technological deterministic view of the PLC as a "disruptive" technology, first proposed by Christensen (1997) is considered.

Evidence is presented to demonstrate that there has been little emphasis or interest in academic literature concerning the history and development of sequential control technologies. A lack of substantive literature pertaining to industrial (or otherwise) sequence control has directed the research toward alternative sources of literature, namely the trade press (magazines such as Automation, Control Engineering and the like) and in particular, the use of patents.

3.2. Use-based and Disruptive technologies

Jones and Bissell (2011) observed that since the 1970s, many science and technology studies have "turn[ed] away from technological determinism and towards various forms of 'social shaping of technology'". One particular view has taken the form of "the social construction of technology" (SCOT) which argues that technology is shaped by human interaction and activity (see Bijker, 1995, Bijker et al., 1987, Bijker and Law, 1992, MacKenzie and

Wajcman, 1999). However, the view, as held by Christensen (1997), that technology is disruptive and generates change because of its very existence, still persists. Jones and Bissell maintain that in historical studies, technological determinism has been largely sidelined by the social constructivists view. However technological determinism "is still often found in the media, politics and business" and receives continued support from economists and business academics.

3.2.1. "Hidden" Technologies

Edgerton (2006) tells us that the term 'technology' is linked to novelty and the future. Technology is often related to 'invention' and 'innovation' and it is this perception that influences many historical accounts. Edgerton calls this approach "innovation centric", which is frequently defined by key emergent technological accounts that are often reduced to fixed dates. Some technologies are widely used but either unrecognised or regarded unremarkable by both academia and the general public. Edgerton remarks that many studies in the history of technology are innovation centric, concentrating on the first-use and diffusion of new "attention grabbing" technologies. These technologies are indeed important but many existing and developing technologies, significant through their widespread use, are "hidden" from the limelight, taken for granted and are frequently ignored.

According to Edgerton (2006), there is an alternative "use-centred" or "use-based" approach to the history of technology, the "history of technology-in-use". Edgerton describes many existing older technologies that are well diffused and still in widespread persistent use. These older, mature technologies may not have found favour with the academic community because they lack the excitement and interest of leading edge "state-of-the-art" developments but they are still significant. Typical examples given by Edgerton include: the "common" bicycle; corrugated iron – a building material still very much in demand throughout the world; and the washing machine – a device that transformed home life in the 1940s and 50s and still going strong. What these technologies have in common is that they do not particularly stimulate interest either with the general public as a whole, or academia in general, and have become "hidden".

Edgerton points out that:

> "...most writing on the history of technology and on the relations of technology and society is concerned with innovation, with the emergence of new technologies. It fails to distinguish this from the study of technology in widespread use, which is necessarily old, and is often seen as out-of-date, obsolete, and merely persisting" (Edgerton, 1999)

As a result of this apparent lack of interest, academic attention has been largely devoted to understanding and commenting on emergent technologies and innovation. The history of the PLC, discussed in this thesis is also a technology-in-use; it too has largely been overlooked in history of technology studies.

Did the "invention" of the PLC push itself onto the market and create demand or did it develop and arise from market requirements or "pull"? Edgerton explains that...

> "Invention and innovation rarely lead to use, but use often leads to invention and innovation...The very fact of adoption, leads to development effort being concentrated on these technologies" (Edgerton, 1999)

The impetus behind this development is explained by "reverse salients" (Hughes, 1983), parts of a technological system, that due to inadequate or deficient development, impedes the development of the whole system. It is the existence of these "reverse salients" or bottlenecks in a technology in-use that provide the impetus for incremental and radical innovative activity (Edgerton, 1999).

Edgerton claims that the bias in literature is toward the "study of scientists and technologists employed in research" and not the roles associated with other forms of work. He remarks that "Just as we should not confuse innovation with technology-in use, we should not confuse changes in knowledge with knowledge in-use" (Edgerton, 1999).

On the impact a technology has on communities and society in general, Edgerton comments:

> "'Technological determinism' is the thesis that a society is determined by the technologies in use. Nevertheless it is usually defined and attacked as the absurd thesis that technical innovation determines social change" (Edgerton, 1999).

In the context of the PLC, it is the engineers and technicians using the technology that form the "society". Within this society, the technological terms and language forms a distinctive culture, and as Monk (2005) noted "Those that become users of a dominant terminology come to form the rump of a professional group identified by the language that they used".

Jones and Bissell observe that:

> "Distinctive features of the SCOT [Social Construction of Technology] framework include the notion of 'relevant social groups'. Such groups are defined as 'those groups who share a meaning in an artifact' and can include designers or users" (Jones and Bissell, 2011).

The sub-culture associated with the PLC was not purely defined by the PLC itself, rather, it existed beforehand. The PLC technological system, including its cultural and linguistic aspects was absorbed into that culture.

Social Construction of Technology (SCOT)

The social constructivist, or SCOT view, emerged from the seminal workshop at the Twente University of Technology in 1984, to discuss developments in the Social Studies of Technology. This was the workshop where Pinch and Bijker (1984) presented their landmark paper "The Social Construction of Facts and Artefacts: or How the Sociology of Science and the Sociology of Technology might Benefit Each Other". Law (1984), reviewing this workshop, commented "Perhaps, in retrospect, we may look back to it as the place where the social study of technology first became a recognisable field rather than a set of individuals". Papers from the workshop were reproduced by the widely regarded book edited by Bijker, Hughes and Pinch (1987) entitled "The social construction of technological systems : new directions in the sociology and history of technology".

The basis of SCOT theory draws from the work of historians and sociologists and is counter to the view that technology is deterministic. The SCOT view proposes that technological systems are continually changing and are the product of human intervention and interaction; therefore, they cannot be fully comprehended without gaining an understanding of the political, social, economic and technical elements associated with a technology. Prell (2009) argues that "Key among the strengths of this approach is SCOT's aim to open up the "black box" of technology, thus revealing the multiple social forces that influence and shape the life-course of a technology".

Pinch and Bijker (1984) argue that technological development progresses because of the user's requirement for variation in a technology, for example, the different requirements for the early bicycle relating to speed and safety. Variations arise because people perceive problems with a technology differently. Hughes (1983) interprets these problems as the "reverse salients" that direct technical effort to resolve them.

SCOT, however, does have its critics, most notably from Langdon Winner (1993). Winner argues that the social constructivists approach, reflected in the methodology proposed by proponents of SCOT, is narrow in its perspective. There is an "almost total disregard for the social consequences of technical choice" which explains the emergence of a technology but ignores its future effects and consequences. SCOT assumes which social groups are relevant, ignoring and excluding some social groups that "...have no voice but that, nevertheless, will be affected by the results of technological change". It is superficial because of its focus only on the "immediate needs, interests, problems, and solutions of specific groups" and disregards the possibility that there may be other "dynamics evident in technological change". Finally, Winner argues that SCOT only "notice[s] that some technological projects succeed and others fail" and offers no deeper moral judgement.

Theoretical Approach

The most useful theoretical background for this study is Edgerton's work on technology in use.

Edgerton (1999) provides some guidance for researching the history of technology and distinguishes innovation from use. Researching a technology-in-use "opens up new areas for investigation, including the history of maintenance, repair and remodelling, as well as further developing accounts of innovation based on use" These themes seem particularly resonant for the history of the PLC 'system' because it is a technology in use. The PLC system includes a small but well defined set of users and it is not the fact that engineers in general are familiar with the technology but a well defined subset of the engineering community, the electrical and control engineer.

3.2.2. The "Hidden" PLC

The Programmable Logic Controller or 'PLC' appears to meet Edgerton's criteria for a "hidden" technology. It is a specialised computer control system, used to control widely disparate machines and processes in many different and widely dispersed applications. Applications range from industrial sequential and process control for manufacturing (conveying and assembly systems); to utility and non-manufacturing applications (airport baggage handling systems, lock gates and lifts); and the more obscure applications of fairground rides (Anon, 1996) and the automated nap chair (Allen, 2004).

The machines and processes under the control of a PLC are often highly visible to the general public and industrial users. The PLC itself is never seen in public; it is frequently housed in protective enclosures and buildings and hidden from public view. PLCs were ubiquitous, hidden from view and taken for granted but essential to the operation of many industrial processes. An article in Vanity Fair (Gross and Karlsson, 2012), regarding the vulnerability of PLCs to a computer virus called 'Stuxnet', found it necessary to explain what a PLC is and what it's used for:

> "programmable-logic controllers—tiny computers about the size of a pack of crayons, which regulate the machinery in factories, power plants, and construction and engineering projects. These controllers, or P.L.C.'s, perform the critical scut work of modern life. They open and shut valves in water pipes, speed and slow the spinning of uranium centrifuges, mete out the dollop of cream in each Oreo cookie, and time the change of traffic lights from red to green... Although controllers are ubiquitous, knowledge of them is so rare that many top government officials did not even know they existed until that week in July." (Gross and Karlsson, 2012)

The PLC is not an attention grabbing technology for the unfamiliar observer. A PLC doesn't actually look like a computer that most people are familiar with, it is just a box connected to many electrical wires and cables with no discernable information display of interest to the passer by. These apparently bland details and appearance of the PLC contribute to the general lack of public awareness and even knowledge of its existence and accord with Edgerton's view of a hidden technology.

There are other aspects of the PLC that align it with Edgerton's differentiation of a novel technology and a "technology-in-use". The PLC was developed from the innovative pre-existing leading edge technologies in electronics and computing. It was a user of existing technology and followed development rather than pushed the technological frontiers. Historians have shown little interest in the PLC because its development has been relatively unremarkable, and have instead concentrated on electronic and computing innovation. Academic literature therefore appears to be limited on PLC development.

The PLC is a technology that was developed from the techniques employed with pre-existing technologies and practices. It required the understanding of engineers and technicians familiar with the skills and techniques prevalent at the time and point of use, namely, electrical engineering and relay-based control systems (for combinatorial and sequential control systems). Computers for control purposes had been available and employed since the 1950s, but were expensive, frequently unreliable, and required specialist skills that were not readily available. The requirement for automation systems was that electrical, plant and process engineers could use and maintain them, without these specialist skills.

3.2.3. Disruptive Technologies

Christensen (1997), in his widely influential work "The Innovator's Dilemma" coins the term "Disruptive Technology" to describe a technology that superimposes itself onto an existing technology and replaces it. Although, in general, the context of Christensen's book looks at the commercial view of innovation, the concepts can be applied to technological aspects.

In order to define what a disruptive technology is, it is perhaps useful to first define what a disruptive technology is not. Christensen initially makes a distinction between two new technology groups: "sustaining technologies" and "disruptive technologies". Christensen explains that "These concepts are very different from the incremental-versus-radical distinction". A sustaining technology is a new technology that improves the performance of the incumbent technology and can be incremental, radical or discontinuous in nature. Christensen observes that most technological advances are sustaining in nature (Christensen, 1997).

A disruptive technology on the other hand, is a radical alternative to an existing incumbent or dominant technology and has the following characteristics:

- A new disruptive technology initially underperforms the dominant (incumbent) one.
- The disruptive technology provides additional features to that of the dominant technology that a limited number of "fringe" customers value.
- Products based on disruptive technologies are typically cheaper, simpler, smaller or more convenient than those established on the dominant technology.
- The leading firms' mainstream customers generally do not want or initially cannot use products based on disruptive technologies.
- Disruptive technologies are first commercialized in emerging or insignificant markets.

- The new disruptive technology steadily improves in performance until it displaces the dominant one.

(Christensen, 1997, Tellis, 2006)

The following studies provide examples of disruptive technologies that meet Christensen's criteria: The transistor versus the vacuum tube (Christensen, 1997); The digital camera versus traditional film cameras (Danneels, 2004, Lucas Jr and Goh, 2009, Utterback and Acee, 2003) and the electronic calculator versus the slide-rule (Utterback and Acee, 2003). The common theme is that the disruptive technologies (transistor, digital camera and the calculator) eventually replaced the technologies that dominated the market prior to their arrival and ultimate acceptance.

Christensen's criteria for determining a disruptive technology does have its critics however. There are still some differences in opinion of what exactly a disruptive technology is and when it has occurred.

Danneels (2004), Markides (2006) and Tellis (2006) claim that Christensen is lacking a common criterion with which to establish different classifications of technology and that inconsistent terminology is used to prevent a clear definition of "disruptive technology". This lack of a clear definition can lead to incorrect interpretations.

Danneels (2004) and Linton (2009) argue that a technology's 'disruptiveness' is based on perspective, particularly relating to that of a company subjected to the disruptive technology. Therefore a definition of a disruptive technology can differ from company to company; some companies may view the same technology as a "sustaining" rather than a "disruptive technology.

There are also questions raised concerning the point at which a technology is deemed to be disruptive - "is a technology disruptive only once it displaces incumbents that built their business on the prior technology?" (Danneels, 2004). Adner (2002) however, supports Christensen's view and defines a "technology disruption" occurs "when the new technology displaces the mainstream technology from the mainstream market".

3.2.4. Is the PLC a Disruptive Technology?

In "*The Innovators Dilemma*", Christensen (1997) describes a disruptive technology relating to motor controls. This short study centres on "electronic programmable motor controls" and briefly relates the story of the development of motor controls by Modicon and Allen-Bradley.

> "In 1968, a startup company, Modicon, began selling electronic programmable motor controls – a disruptive technology from the point of view of mainstream users of electromechanical controls." (Christensen, 1997)

The 'electronic programmable motor controls' referred to by Christensen relates to the technology that later becomes known as the Programmable Logic Controller (PLC). Christensen (1997) describes the motor controls industry as one that supplies sophisticated heavy-duty switches to stop and start large electric motors and their electrical protection systems (overloads and current surges). These switches were often linked through systems of electromechanical relay systems to provide control for conditional and sequential control. The study goes on to describe how Allen Bradley, a leading supplier of heavy-duty motor controls, entered the market of electronic motor controls shortly after Modicon's entry.

Christensen (1997) claims that the "electronic motor controller" (PLC) is a disruptive technology because it initially underperforms the incumbent technology, the electro-mechanical relay. He also argues that disruptive technologies are started by small start-up companies (for example Modicon) because the leading firms either do not want, or cannot use, the new technology and that their additional benefits are only of interest to a limited number of customers. Christensen explains that suppliers of the emerging electronic motor controller technology were forced to "cultivate an emerging market for programmable controllers: the market for factory automation" and that customers for this technology were not the equipment manufacturers but the equipment users.

The arguments put forward by Edgerton and Christensen appears to be in conflict with regard to the PLC. It does not seem possible that the PLC can be a regarded as a "use-based", "hidden technology" (Edgerton, 2006) and be "disruptive" (Christensen, 1997) at the same time. This thesis reviews the history and development of the PLC technological system and attempts to determine how and where the PLC sits within these two viewpoints.

3.3. Academic Literature Review

The Programmable Controller (PC), perhaps more widely known as the Programmable Logic Controller (PLC) has been used for control and automation applications within the manufacturing and process industries since the late 1960s. Originally designed to replace electro-mechanical relay control panels (Brown, 1984, Clare et al., 1995, Erickson, 1996), it has developed to provide sophisticated control and automation solutions well beyond its original remit. According to some proponents, the development of the PLC had resulted in "a consequential revolution of control engineering" (Bolton, 2006) and PLCs have become the "workhorses of factory automation" (Erickson, 1996).

Despite the PLCs impact on the industrial world of process automation, historians and academics have concentrated on the development of automatic control theories relating to feedback. Computer scientists and historians have also shown little interest in the PLC as a technology and have concentrated on the development of electronic and computer technologies. As a result, the historical development of the PLC appears to have been of little interest to the academic mainstream. Searches conducted through academic and professional institutional sources such as: Science Direct, Society for the History of Technology (SHOT), IEEE History Centre, British Library, IET, InstMC and the Newcomen Society has revealed little information on the development of the PLC.

3.3.1. Feedback and Classical Control Theory

Historians of automatic control (see Mayr, 1970, Bennett, 1979, 1993a, 2002, Bissell, 1991, 1996, 2001, 2009, Mindell, 2002), have primarily concentrated on feedback or classical control. However, sequential and combinational (logic) control is barely mentioned in these works, and certainly not discussed in any great depth.

Bissell provides an excellent resource for identifying the main contributions to the field of the history of control up to the mid 1990s in his "Secondary sources for the history of control engineering: an annotated bibliography" (Bissell, 1991) and its update in 1995 (Bissell, 1996). Bissell explains that the bibliography is not a "detailed critical assessment of citations" but it outlines the scope of the most important contributions, summarised under a set of headings. The bibliography demonstrates that there are few historical studies in modern control and sequential control is not included in the bibliography. Indeed, Bissell notes that there is an "...overwhelming preponderance of publications dealing with the development of classical control engineering or with earlier periods; there appears to be little historical assessment of modern control" (Bissell, 1991).

Bennett's "A History of Control Engineering 1800 – 1930" (1979) and his second volume covering the period 1930 – 1955 (Bennett, 1993b) provide a detailed examination of automatic control and traces the technological developments of feedback and the development of classical and modern control theory. Sequence control and PLCs form no significant part in these studies, nor does Bennett's "A brief history of automatic control" (1996), however he provided a clue to this perceived indifference and attitude to sequential control when he wrote,

> "At the beginning of the 1920s 'control' was, with a few exceptions, thought of as the switching on, or off, of devices – motors, pumps, valves – either directly or through relays". (Bennett, 1993b)

The control applications Bennett describes are relatively simple applications using well established methods and technology. The more challenging and mathematical aspects of feedback control was where the academic interest lay. This perspective is shared by other historians and is reflected in the works of Bissell, Mayr and Mindell.

The importance of feedback control cannot be understated, however, the relevance of sequential control is not completely overlooked by the historians. Bissell (2009) mentions the development of the PLC as an important but underrepresented technology amongst the application of computers in process control.

The technology to achieve combinational logic and sequential control was well established and proven, requiring little modelling or specialised design techniques. It was not until digital techniques became available from the development of electronics and computing, that academic interest was stimulated. Spurred on by developments for military applications such as gun, radar and missile control (Mindell, 2002) and the space race (James, 1981, Johnson, 2008, Mindell, 2008), the application of mainstream computers were largely applied to the analysis and control of analogue parameters for industrial process control and the control of servos rather than the on-off 'logic' control of plant and processes.

Bennett turned his attention to sequence control in the field of real-time computer control (Bennett, 1988) and stated that "[a]lthough sequence control will occur in some part of most systems it often predominates in batch systems..." He continued by segregating pure sequence control technologies from mainstream process control computers because...

> "Most sequence control systems are much simpler and have no loop control. They are systems which in the past would have been controlled by relays, discrete logic or hard-wired, integrated circuit logic units... Special computer systems known as 'programmable controllers' have been developed for these simple sequence systems together with special program languages" (Bennett, 1988)

Bennett draws a technological distinction between sequence control systems and process control computers in terms of the relative simplicity of the control task. Sequence control was the simpler task that could be accomplished by using the specialised programmable controller. Process control on the other hand was regarded as the more complex task requiring real-time process control computers to perform closed-loop feedback control using direct digital control (DDC) techniques.

Bollinger and Duffie (1988) explain that for years "analog devices were controlled by analog controllers" and that "there was no need to be concerned with discrete events and the logical intelligence for machines and processes, commonly referred to as sequential control". Their explanation for this is:

> "On the one hand, feedback control theory formed a significant body of knowledge on which much research has been and continues to be done. It was not uncommon for a control engineer to work on only this aspect of a problem. On the other hand, sequence control formed another body of knowledge that was originally implemented with little theory (originally in the form of relay or pneumatic logic) and was generally implemented by the practitioner of logic systems". (Bollinger and Duffie, 1988)

Pneumatic logic did not really develop for industrial systems until the mid 1960s and interest soon waned with the developments in electronics and the introduction and use of the integrated circuit (Scrupski, 2001). However relays were used for industrial logic and sequential control systems and Bollinger and Duffie indicate that prior to alternatives for relay control systems, such as PLCs, sequence control was a separate, non-academic discipline from feedback control; it was regarded as a practical skill that required little theoretical knowledge. They allude to the fact that such systems were implemented by practical engineers and technicians.

Ashley and Pugh (1968) endorse Bollinger and Duffie's view and state that there is evidence to support "that little analysis is used in the design of control systems for sequential machines", and claim that "It is true to say that virtually all control systems for sequential mechanisms are designed on an intuitive basis".

Following the establishment of commercial programmable controller technology in the 1970s, the view that there was no formal design method for programming sequential control systems was sustained. This view was endorsed by Rahman and Woodward:

> "Previous work in this area has been of a rather specialised nature. A generalised model for p.s.c.s [programmable sequence control systems] is not available, and, consequently, the designs of such systems are based on intuition" (Rahman and Woodward, 1975)

Erickson (2001) provides further evidence that sequential control was not seen as a serious academic discipline and that the teaching of PLC technology was marginalised in universities. Erickson explained that in the early 1980s "the typical control engineering graduate (ChE, E,or ME) had a course in feedback control theory and those interested in a career in control secured a position in the aerospace or chemical industries".

Over the next 20 years (1980 to 2000), an increasing emphasis on manufacturing automation involving PLCs had developed. Erickson (2001) observed that in spite of this, "The typical college or university has been slow to recognize this trend" and that many universities had devoted only "a portion of a course or laboratory to PLCs". In Erickson's opinion, "every university that teaches control system courses should have at least one course devoted to PLC programming". This statement appears to support the notion suggested by Bollinger and Duffie (1988) that sequence control was regarded as a non-academic, practical skill that could be implemented with little theory and therefore required little training.

The PLC, it appears, is the poor relation as an automation technology. A study conducted by Collins et al. (1996) looked at the adoption of programmable automation technologies and discussed the uptake of the key automation technologies and the organisations that used them. The study focused...

> "...on a myriad of programmable automation technologies that gained a foothold in 1970s and became more widely adopted in the 1980s. Programmable automation (PA) is based on computer control systems, instead of mechanical or electro-mechanical control systems found in conventional, "hard" automation technologies". (Collins et al., 1996)

The technologies Collins et al. concentrate on are "Discrete manifestations of PA" which are computer numerically controlled machines, robots, and automated transfer devices. Flexible manufacturing and/or computer integrated manufacturing are examples of PA integrated into entire systems. What is interesting, is that the PLC is one such technology, widely used at the time of the study, but not mentioned in the paper.

3.3.2. The Emergence of Digital Techniques (Drums, Cams and Relays)

By the first half of the 20th century, electro-mechanical technologies for achieving sequential control consisted of drums, cams and relays. Rotating drums used protrusions that opened or closed switch circuits as the drum rotated. Sequential control was achieved by the relative position of the switches on the drum exterior. Timer control was dependent on the speed of rotation of the drum and its switch position. The use of cams extended the capability of the drum sequencing by allowing some level of alteration of the control sequence by the removal, addition or adjustment of the cams.

The limitations of performing sequence control by drums and cams were the constraints of fixed speed, physical size and the fact that only open loop control could be achieved. Drum and cam control systems were limited to the fixed speed of the motor and gearing arrangements that were permanent once set. The physical size of the electro-mechanical drum or cam arrangement limited the number of control switches and hence constrained the level of complexity in the program of the sequence. Because the drum or cam system was fixed it could not readily respond to external conditions and could only follow a fixed sequence (Walker et al., 2010).

A relay is an electrically operated switch. It consists of one or more sets of contacts that are opened or closed mechanically by an electromagnet or solenoid. Relays were particularly versatile because of their flexibility and reliability. In order to achieve sequential control, relays were interconnected through their wiring to create combinational or sequential logic; the wiring effectively held the logic program. The flexibility that the wiring gave was the ability to alter the "logic program" on a semi-permanent basis by changing the interconnections and allowed more complex control schemes to be developed. The wiring partially overcame some of the fixed spatial constraints associated with rotating drums and made possible the ability of the control system to provide a limited response to control situations. Unfortunately, like the mechanical drums and cam systems, altering the program (changing the wiring) was time consuming and prone to errors (see chapter 4).

In essence, drums and cams timed things and were only able to perform open loop sequential control. Drums and cams were unable to monitor and respond to conditions on the plant or perform logic because control events were strictly related to the fixed pattern on the drum. Relays on the other hand, were interconnected in such a way as to give logic or combinatorial control that was able to react to conditions on the plant, but had no timing capability; without timing, relays were unable to perform sequential control. Neither drums nor relays alone could do both sequential and combinatorial logic.

To use relays for sequential control, a timing device of some sort is required. Limited timing could be achieved using time-delay relays employing bimetallic strips for the timing device but sequential control was not easy to accomplish. As Chapter 4 will show, it was possible to develop more comprehensive sequential control systems from a combination of drums and relays.

Complex relay-based control systems were still relatively large due to the physical size needed to accommodate the relays and the wiring looms. It was common practice to house the resulting relay logic in enclosures to manage the wiring and configuration of the control system, however, the size and power requirements of the enclosure was proportional to the complexity of the control application. Consequently reliability became an issue when large numbers of relays were interconnected for complex control schemes. Relays had a finite life in terms of number of operations and alterations to the 'program' were a time consuming and complex task, "With respect to electrical equipment only, the most frequent single source of down-time is relay failures and malfunctions" (Macoy, 1966). It was thus desirable to keep the logic as simple as possible.

Bollinger and Duffie (1988) mention pneumatic control as an alternative to relays but this was not really a contemporary of the relay until the 1960s. Up until the 1950s, pneumatic control was used primarily for analogue process control applications. However, by the late 1950s and early 1960s, a "second stage of development" in pneumatic theory saw wider applications that encompassed communications and pneumatic logic (McHutchison, 1966).

The use of pneumatic and hydraulic means to conduct logic control, collectively known as "Fluidics", was initially developed for the aerospace industry and owed much of its development to the electronics industry (Parker, 1969). They were further developed into the mid 1960s and were deployed in some industrial systems but according to Scrupski (2001) "interest faded as digital ICs came online". Although interest may have faded, pneumatic logic and control are used where electrical and electronic control may not be appropriate, for example explosive environments, and are frequently used in conjunction with other control systems such as computer control and PLCs (Wojtecki, 1999).

3.3.3. The Versatile Relay

Relays were used extensively and played an important role in the development and operation of telephone exchange systems. In the late 1930s Claude Shannon (1938) realised that relays could be wired in such a way that Boolean logic problems could be represented and resolved using the digital on-off states of the relay. Shannon was concerned with relating relay logic to Boolean algebra thereby simplifying the design and analysis of such relay circuits. The advantage of this approach was the use of fewer relays and hence a reduction (or optimisation) in size could be accomplished. Shannon's influential Masters thesis (1940) prepared the way for the development of digital circuits and computers using binary devices that initially incorporated relays.

Many developments were subsequently transferred to other applications such as early computers and industrial control. For example, computer pioneer Konrad Zuse, used telephone relays to build his Z2 and Z3 computers in 1941 (Rojas and Hashagen, 2000).

Keister (1949) talks about the combinational logic that can be achieved with relays using the telephone exchange as an example. Keister's paper focuses on simple combinational logic derived from telephone relays and concepts.

He also refers to "higher order circuits" by which he means sequential control applications thereby differentiating between the two forms of logic-based control (combinational and sequential). Ritchie (1949) in a related paper, describes the use of relay logic to achieve this sequential control.

Relays were not simply confined to perform digital switching and combinational logic. From the discipline of control, Tsypkin published his classical book "Relay Control Systems" (Tsypkin, 1984), originally published in Russian in 1955. Tsypkin's work reflected the versatility and complexity of relays used in feedback control systems. Tsypkin developed what is known as the "Tsypkin locus" from his research based on delay and sampled data systems. This work was further scrutinised by Bergen (1962) and Gelb and Henrikson (1963).

3.3.4. Computers and Information Technologies

By the 1950s, computers had advanced from using electro-mechanical relays to electronic components, initially using vacuum tubes or valves. Electronic computers, for example "Colossus" developed by Tommy Flowers (McKay, 2010) and ENIAC (Wilkes, 2006) used vacuum tubes or valves to conduct the switching at considerably higher speeds than relays. Valves were to be later replaced by transistors increasing the computational power, reliability and eventually reducing physical size and cost.

In 1953, Claude Shannon commented on the recent developments in computing and automata in the journal of the Institute of Radio Engineers (IRE) and in particular, the use of computers (Shannon, 1953). Shannon explained in the paper that the "bread-and-butter work" of large-scale computers, largely involved solving numerical problems. Shannon also stated that "To many of us, however, the most exciting potentialities of computers lie in their ability to perform non-numerical operations". These non-numerical operations included "work with logic, translate languages, design circuits, play games, co-ordinate sensory and manipulative devices".

Shannon recognised some key applications of computers other than the number crunching they were originally tasked with. Of note are the applications of "circuit design", logic, and the coordination of "sensory and manipulative devices" which are very much engineering activities, in particular, the latter two are closely related to control. Shannon develops this argument and states:

> "The largest and most reliable current information processing machine is still the automatic telephone system. Our factories are filled with ingenious and unsung devices performing almost incredible feats of sensing, processing and transporting materials in all shapes and forms. Railway and power systems have elaborate control and protective networks against accidents and human errors. These, however, are all special-purpose automata. A significant new concept in non-numerical computation is the idea of a general-purpose programmed computer - a device capable of carrying out a long sequence of elementary orders analogous to those of a numerical computer. The elementary orders, however, will relate not to operations on numbers but to physical motions, operations with words, equations, incoming sensory data, or almost any physical or conceptual entities." (Shannon, 1953)

The relay-based telephone system of 1953 was still dominated by the electro-mechanical relay and Shannon draws attention to the "unsung" technologies employed in the factories, the combinational and sequential control systems. However, he describes these systems as "special-purpose automata", which to all intents and purposes they were. Many industrial sequential control systems were based on fixed-wired relay logic which was not readily alterable and therefore fixed in function in the same way as automata. The "significant new concept in non-numerical computation" (the computer control of devices) was the "general-purpose programmed computer" (Shannon, 1953). Within 20 years, this new concept would also be available to the factory in the form of the Programmable Logic Controller.

3.3.5. History of Computing

Computer historians have looked in detail at the development of the computer in terms of their design (hardware) and the software that programs them. Leading historians such as William Aspray and Martin Campell-Kelly have written prolifically on the history, policy, and social study of computers and information technologies (see Aspray et al., 1990, Campbell-Kelly and Aspray, 2004, Campbell-Kelly, 2003, 2007, 2009). Industrial applications, and in particular real-time computing, have concentrated on the informational aspects of computer technology. Campbell-Kelly and Aspray (2004) for example, jointly discuss real-time computing in terms of the developments derived from "Project Whirlwind" (a computer aircraft simulator developed during World War II) and the real-time computer, focusing on applications providing: real-time response facilities for its users; real-time information applications including air defence systems – 'SAGE' (Semi-Automatic Ground Environment); airline booking systems; and barcodes. The application of real-time computers in manufacturing for automation and process control are not discussed in detail (with the exception of manufacturing accounting systems).

The history of the modern digital computer, software development combined with its social, business and economic implications are dealt with in great detail, yet there are still many facets of the computer technological system to come under the scrutiny of the historians, and as Aspray concedes, "The history of information science is a young field" (Aspray, 2011).

3.3.6. Process Control Computers

Other computer historians have taken an application perspective on the industrial use of the computer for manufacturing and process control. Jonathan Aylen (2010) highlights the difference between industrial and commercial computers…"Process-control computers were technically significant because they had different design objectives to general-purpose computers for scientific and commercial use". Process-control computers required real-time responses in order to "sample data from a variety of sensors, calculate time-based derivates, and distribute signals meant strict timing control". These industrial requirements brought about the development of interrupts, an emphasis on analogue to digital conversion and the early adoption of transistor and ferrite core memories and visual display units.

Initially, in the 1950s, computers were employed in an "advisory" capacity providing data logging and alarm facilities (Aylen, 2004). One such system, General Electric's GE 312 GARDE (Gathers Alarms Records Displays Evaluates) was "explicitly designed not to implement control actions", thus the role of the computer was consigned to a supervisory one. A reason given by Aylen (2004) was that "In general, sensing devices were not reliable enough to allow direct online process control at the ironmaking and steelmaking stage". However, the 1960s saw the control loop close due to the increased reliability in sensors and the introduction of Direct Digital Control.

According to Aylen, the late 1950s saw the application of computers to control and monitor industrial processes. E. J. Otis, writing on Industrial Digital Systems in 1958, discusses the application of digital computers to process control.

> "An examination of any industrial process today involves the consideration of a complex problem encompassing many variables to be measured and to be controlled... It also emphasises the need for a system sufficiently adaptable and flexible to satisfy the many and diverse input-output combinations encountered in a process." (Otis, 1958)

Otis (1958) highlights the requirement to provide process data as inputs to the computer. Figure 3.1 provides a snapshot of the functional components of the process-control computer. Analogue process readings are individually connected to a single "scanner" and analogue-to-digital (A/D) converter. A digital switch is used to select either digital data or the current analogue value output from the converter. Prior to visual displays, operator input was by the typewriter or tape reader for loading a pre-written program. The information output to the operator was also via typewriters. The diagram also shows only digital outputs directly connected to the process rather than analogue outputs.

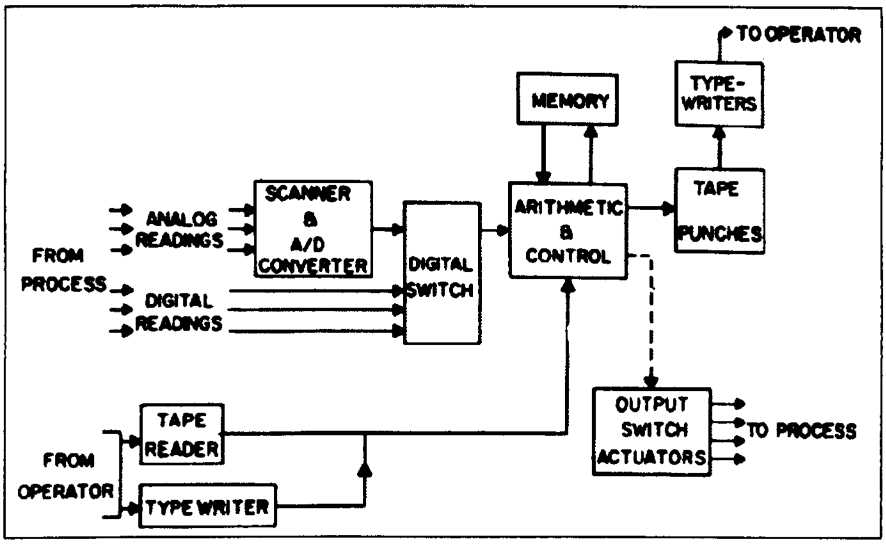


Figure 3.1 – Typical digital control computer Otis (1958)

Developments in electronics, from the transistor to integrated circuits (ICs) reduced the size and cost of computers and their use became more prevalent in the steel industry. The 1970s and 1980s saw the introduction of PLCs – Aylen observes that:

> "As the price of hardware fell, minicomputers or programmable logic controllers (PLCs) became feasible for localised control of processes. These could be bolted on in an almost ad hoc way leading to quick upgrades and the progressive spread of computerisation across an increasing range of process control tasks" (Aylen, 2004).

PLCs were often employed within a hierarchical distributed architecture under the control of a centralised computer. However, this was not problem free, the difficulty encountered with PLCs, observed Aylen (2004) "was that PLCs were becoming more powerful and widely used, but still did not achieve the very fast response times required for process control operations".

The process-control computer became a specialised application, targeted toward industrial processes, and differentiated itself from the commercial and informational computers and software. Rameback (2003) summarises the development of the process-control computer. In the 1960s, computers developed from a process monitoring and supervisory role to that of performing closed loop control and absorbed the analogue electronic loop controller (known as Direct Digital Control or DDC). With the advent and use of the microprocessor in the 1970s (Monk, 1980), the process-control computer became known as a Distributed Control System or 'DCS', specialising in analogue process control. Rameback (2003) makes the point that at the same time, PLCs were entering the industrial process control market to provide sequential control for machines and processes. With regard to both technologies, he concludes:

> "In practice the DCS and the PLC was used in the same industries, but because of the different user groups the systems were developing separately over the years. The DCS users were Instrument and Process Engineers while the PLC users were the Electrical Engineers." (Rameback, 2003)

Rameback points to the fact that electrical engineers used PLCs and process engineers used a different technology, the DCS. The different technologies in use emphasises the cultural divide between the two user groups. The PLC was a technology that developed from methods using electromechanical relay control logic techniques (relay ladder logic, as discussed in Chapter 2) that was intuitively programmed by end users (Ashley and Pugh, 1968, Rahman and Woodward, 1975); these end users were the electrical engineers and technicians. Process and instrument engineers were largely concerned with control loops and feedback where the sophisticated use of computers and the application and understanding of control theory was dominant. As Rameback (2003) explains, although in practice the two groups worked together, their technology remained separate.

3.3.7. Recent Literature (1990 – 2011)

The literature thus far has been discussed by way of a historical narrative. The thesis reviews the development of the PLC up to the end of the 1980s and it is useful to have an understanding of more recent literature concerning the development of PLC technology. This section considers the key themes reported in the recent literature of the last two decades relating to the development of the PLC.

Over the last two decades (1990-2010), literature concerning the PLC has concentrated on the themes of integration, software development and applications. Development of core PLC technologies has slowed and the PLC has become a mature technology. Academic interest has shifted to systems oriented studies (Bradley, 2006, Rehg et al., 1999, Russell, 1990) and has concentrated on how the PLC can be used in larger and wider control and information technologies.

Integration has therefore been an important developmental aspect of PLC technology, achieved through the adoption of industrial and commercial communications technologies (Adsett, 1990). The use of digital communication techniques enabled the PLC to exchange data with other computer systems providing connectivity with computer-based Human Machine Interfaces (HMIs) that replaced the early special purpose 'mimic' panels[7]. Further benefits allowed the PLC-based control system to communicate to higher level computer systems such as 'Supervisory Control and Data Acquisition' (SCADA) systems[7], providing the means to distribute not only inputs and outputs but also control via early 'fieldbus' technologies[7] (Rao et al., 1995).

The early digital or 'industrial' communication systems were often proprietary and limited to single PLC manufacturer systems. Attempts were made to standardise the industrial communications systems and protocols in the guise of 'fieldbus' and resulting in the IEC 61158 standard that was only partially successful (Felser and Sauter, 2002). The adoption of Ethernet and industrial Ethernet for control purposes has now become the preferred communications protocol. With the availability of Ethernet-based communications, the emphasis has concentrated on connectivity to external higher-level business software systems such as Enterprise Resource Planning (ERP) and Manufacturing Execution Systems (MES) (Bradley, 2006).

Software development has included the use and adoption of high-level computer programming languages such as "structured text". The development of sequential programming techniques has also featured in recent literature, particularly the development of GRAFCET, also referred to as Sequential Function Chart (SFC) (Honchell and Robertson, 1996, Lewis, 2004). Standardisation of the programming languages has also been a concern of industry and the introduction of the IEC 61131 standard (BSI,

---

[7] 'Mimic panels', 'SCADA' and 'Fieldbus' are discussed in the Glossary and Chapters 5 and 6.

2003b) has played a large part in how programming notations are presented to the user. The ladder diagram notation still appears to be the dominant programming language preferred by PLC users (Payne, 2007, Sudhir and Sujata, 2011).

Another significant area of recent literature, involving sequential control and the PLC, has been devoted to industrial control system applications. Examples include: the implementation of a distributed PLC-based control system for a wastewater treatment plant for the Port Alice pulp mill, Canada (Stively and Keiver, 1996); the integration of PLCs and SCADA for automated manufacturing processes within the pharmaceutical industry (Alata et al., 2008); and a recent PLC application for the control of induction steel heating furnaces (Unver et al., 2005). Unver et al justify the use of a PLC for their application because "...modern PLCs are modular, affordable, have large memory capacities, and also have special functions such as pulse width modulation (PWM) and data communication". Industrial applications related to software developments and IT systems integration dominates the literature in this area.

Other literature discusses the diminishing differences between Distributed Control Systems (DCS) and PLC technologies. Davis (1992) argues that the distinction between the two technologies is fading; software tools have been developed for the PLC enabling it to perform process control and DCS have adopted the sequence control techniques used with PLCs, including the adoption of the ladder logic notation. Fauci (1997) expands this argument and explains that PC-based SCADA enhanced the capability of PLC control systems enabling them to compete directly with DCS. Finally, Rameback

(2003) supports Fauci's argument and comments that "The borderlines between PLC's and DCS's are disappearing for the process industries and more DCS functionality is moved into the systems that are used by the Manufacturing Industries (traditional PLC users)".

Some "traditional" PLC functions are also being moved into the DCS, the interfaces to switchgear and motor drive equipment are examples. Rameback concludes that "there are no fundamental differences between a DCS and a PLC" and that they effectively do the same job. Summing Davis, Fauci and Rameback's arguments, the competing technologies of the DCS and PLC have developed to such a degree that they can perform the same functions. The choice of technology arguably is one of preference and "At the end of the day, overall reliability is the main requirement for a process control system" (Rameback, 2003).

3.4. Alternative Literature Sources

3.4.1. Handbooks

There is a large body of practical and applied technical literature relating to the PLC and its technology. Several books were written about PLCs with a practical bias, discussing PLC technology and its implementation using example industrial applications. Engineering handbooks, for example Geng (2004) and Sudhir and Sujata (2011) and Bhuyan (2006) provide very succinct overviews of programmable technologies. Clare et al (2005) provide a more detailed description of PLC technology, programming languages and applications. Other books, such as Ackermann (1993), Batten (1994), Bolton (2006), Bryan and Bryan (1997) and Crispin (1990) describe the technical features in some detail and in particular programming techniques with a heavy bias toward the end user – the practising engineer and technician.

There are few accounts of the development of the PLC and most books contain a brief introduction that largely centres on the Modicon story where the first commercially recognisable PLC was developed and installed for General Motors Hydramatic Division in Detroit.

Morley, often described as the "father of the PLC", features significantly in many accounts such as Andrew and Hugh (2007), Hendricks (1996) and Ball (2008) and in Morley's own books "Out of the Barn" (Morley, 2003) and "The technology machine: how manufacturing will work in the year 2020" (Moody and Morley, 1999). Morley, originally working for Bedford Associates, oversaw the development of the Modicon 084 (**MO**dular **DI**gital **CON**troller), with Modicon becoming a company in its own right.

As with many stories of invention, the PLC has its own "creation myth", although many have a grain of truth in them. Myths relating the Morley story regarding the New Years Day "memo" and GM Hydramatic are included in the accounts of: Hendricks (1996), Bryan and Bryan (1997), Christensen (1997), Clare et al (2006), and Ball (2008). Morley is more humble about the stories reported in the trade press and states that he was just a member of the team at Modicon. "He sees himself as the father of the PLC, rather than the inventor" says Dunn (2008).

Ball (2008) also notes the “origination disputes” of the PLC:

> “In later years, a competitor started action to challenge Modicon's claim that Dick Morley was the "father" of the PLC. Attorneys could not find an early PLC patent listing Dick Morley”

The reason, according to Ball, was that Morley’s patent was entitled “Digital Computer” (see Chapter 5) and continued legal action was not pursued.

3.4.2. Corporate Literature

Corporate journals can offer an alternative insight into the development of a technology. The Fuji Electric Review for instance reveals some developments in the use of electronic devices for sequence control purposes. Topics covered include: the initial use of germanium transistors for “control function and reliability” (in comparison to magnetic relays) and the adoption of silicon transistors (Sakuragi et al., 1968); the automation and “centralisation” of sequence control (Akiyoshi, 1969); the “PROGIC” “IC-circuited sequencer”, programmed by “dioded-pins” (Hayashibe, 1972); to the development of the Universal Sequence Controller the “USC-4000” (Shizima and Sakuragi, 1973). Corporate Journals such as the Fuji Electric Review can demonstrate the practical application and adoption of technologies by companies historically.

3.4.3. The Press and Trade Magazines

The general public were largely unaware of “programmable controllers” or PLCs until the mid 1980s because of a lack of reporting in the general press (see Gross and Karlsson, 2012). One of the earliest articles uncovered that was associated with "Programmable Controllers" appears in the Times newspaper in 1984. The article entitled "A giant step for automation?" (Brown, 1984) is a report on General Electric's "Workmaster Programmable Control Information Centre", a suitcase-based programming device to "program quickly programmable controllers and robots". Brown briefly describes the programmable controllers as "compact microchip-based units which, because of their versatility and high reliability, are replacing relay-based electro-mechanical systems" and explains that "the Workmaster is easy to use because its programming language is based on the relay logic traditionally used by control engineers". This particular article appears nearly 15 years after the first commercially developed programmable controllers and PLCs were introduced.

Another article in The Times, written in 1985, refers to the application of an Allen-Bradley PLC which is used to automate an automotive production line for building engine cylinder heads sub-assemblies at "Perkins, the Peterborough diesel manufacturer" in the UK (Anon, 1985). This very brief mention of the PLC is largely concerned with automating a factory production line and its consequences rather than the technology used to conduct the control process. A later article, written in 2004, looks an intriguing application of PLCs for an automated power-nap chair! "When time is up [after the nap] a programmable logic controller orchestrates a gentle waking" (Allen, 2004).

The three press articles illustrate the fact that PLC technology remained relatively unreported until the mid 1980s, and little thereafter other than interesting or amusing applications. The lack of reporting in the national press reveals that the general public were generally unaware of PLCs and sequential control technology.

The trade press, scientific and engineering magazines were an important source of information regarding sequential controllers that were targeted at practising engineers and technicians. There were occasional appearances in the form of advertising material such as the "Programmable Matrix Controller" (Allen-Bradley, 1972) in Scientific American (see Figure 3.2). Allen-Bradley, a supplier of electrical and electronic components, placed a full page advert in Scientific American for a modular potentiometer together with a new control system known as a Programmable Matrix Controller (PMC). The PMC was Allen-Bradley's first commercial "stored program" controller and a forerunner of the modern Programmable Logic Controller (PLC). The system was developed for Allen -Bradley by William Kiffmeyer in 1971 and the concept is embedded in his patent US3810118 (Kiffmeyer, 1974).

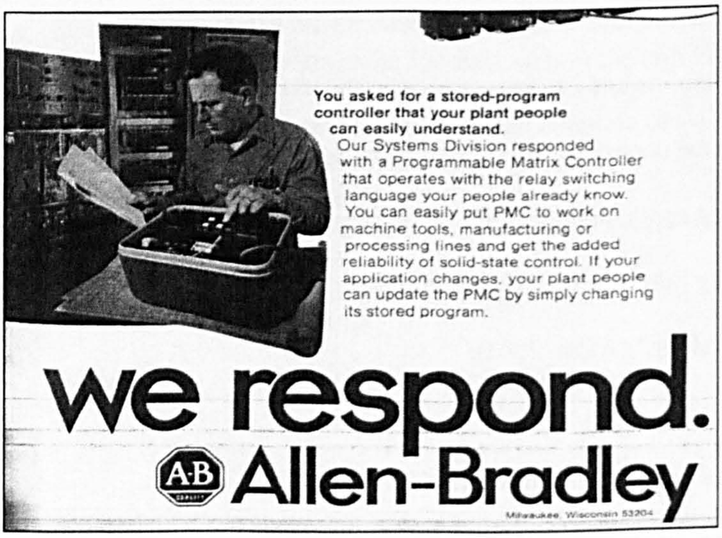


Figure 3.2 – PMC Advert - Scientific American 1972

A number of articles were published in *Control Engineering* as the PLC technology was emerging. Bairstow (1969) reports on the development of "[a] simple programmable controller that may spell the end for relay panels", the PDP-14 from Digital Equipment Corporation (DEC). Bairstow explained that "[s]ince July of last year [1968], when a major auto manufacturer asked Digital Equipment Corp. to propose a standard machine controller, DEC's designers have been hard at work on an answer". Bairstow extolled the virtues of the controller and stated that:

> "[The DEC PDP-14] will do anything a relay panel can do and more. Intended for the control of production lines, transfer machines, and so on, the controller is also superior to a relay panel in that its logic can be changed quickly and at a low cost" (Bairstow, 1969).

It is reasonable to assume that the "major auto manufacturer" was GM Hydramatic, also referred to by Moody and Morley (1999), Clare et al (1995) and others in their accounts of the history of the PLC dating the GM specification to July 1968. The price of the PDP-14 in 1969 was expected to be under $4,500.

Control Engineering published "A special report" on Programmable Logic Controllers nearly a year after the PDP-14 article. The article titled "Programmable Logic Controllers - Painless Programming to Replace the Relay Bank" (Lapidus, 1971) introduces the PLC, the significance of the PLC for industrial control and discusses the systems available at the time of publishing (1971). Lapidus, in dramatic fashion, heralds the arrival of the PLC:

> "A new breed of specialized computer has quietly emerged that may revolutionize one of the few remaining bastions of complex electromechanical control – the relay sequencer...With these

> machines, it is no longer necessary to learn an unrelated technology (computer programming) to perform the control job at hand. At the same time they provide a bumpless transfer for the control engineer who wishes to ease into computer control and the world of solid-state circuits." (Lapidus, 1971)

The use of the expression "bumpless transfer" relates to a term used by control engineers for a smooth changeover for analogue controllers with integral action. The prose used by Lapidus appears to be aimed at the experienced and practising control engineer.

Lapidus' report appears to be influential and is referenced in later patents relating to the development of PLCs (see chapter 5), notably Allen Bradley's Programmable Matrix Controller (Kiffmeyer, 1974, Ricketts Jr. et al., 1973).

In April 1972 Lapidus published a follow-up article in Control Engineering (Lapidus, 1972). The article is based on a survey of PLC users who were asked to compare logic control components (Relays, solid-state logic and PLCs). The findings in the report gives a snapshot view of what industrial sectors were using PLCs and sectors that were beginning to adopt PLCs.

A further article published in Control Engineering (Andreiev, 1972), gives a brief update on PLC technology develepments, use, and discusses suppliers of control systems in general terms. Andreiev notices that there is a move toward the use of semi-conductor memory and improved immunity from electrical interference. Furthermore, Andreiev comments on trends in developments relating to increased sophistication and capability - "some units in this group may begin to approach minicomputers in the sense they can do more than their primary job of Boolean logic (ladder diagram) processing" and the development of smaller PLCs designed as a simple and inexpensive relay-replacer on "single station mass-production machines".

3.5. Patent Literature

Academic literature relating to the history and development of the PLC is sparse. Industry however, recognised the need for the control of sequences and invented their own way of doing things. As a consequence, alternative sources of literature have been consulted; these include commercial (trade) literature and patents. Patents provide a rich source of information that can give a historical insight into the development of a technology and in this case, have been a particularly useful tool in researching the history and development of the PLC.

The history and development of the PLC discovered through the analysis of patents is discussed in-depth in Chapters 4 and 5. A full bibliography of the significant patents reviewed in the course of the research is contained in Appendix C.

3.6. Chapter Summary

This chapter has reviewed the academic and technical literature associated with sequential and combinatorial control and its related technologies. The literature review has shown that literature concerning the development of sequential control and the history and development of the PLC is thin if non-existent. Historians of control have concentrated on the traditional areas of control and in particular feedback. Similarly, computer historians have concerted their efforts on the technical and commercial aspects of computer development.

The lack of academic material has led to a study of alternative sources of literature that include engineering handbooks, trade magazines, manuals, press reports and patents. This literature provided a useful insight into industrial applications, user experiences and identifying technologies and markets for the PLC at the time of publication.

The concepts of a use-centric history and disruptive technologies were discussed highlighting some of the cultural aspects of manufacturing, control and the engineer. It was determined that the PLC as a technology-in-use was a useful framework to review the development of sequential control and in particular the PLC. The notion of disruptive technologies was discussed in some detail and forms the framework for a discussion in Chapter 7 with regard to the PLC being a disruptive technology (or not).

Finally, patents were identified as an alternative source of literature, providing a rich but distinctive source of information in the spheres of technical literature and a means to apply a historical context.

Chapter 4 – Developments in Sequential Control (1900-1969)

4.1. Introduction

Prior to the emergence of the PLC, alternative technologies were used to achieve combinatorial and sequential logic control. This chapter reviews the technological developments in sequential control revealed in patents from 1900 to 1969. The emergence and development of the PLC itself is discussed in Chapter 5.

Patents provide the primary research resource in this chapter. Within the patent material reviewed in this thesis, two distinct themes occur: technological change and functional (or systems) change. Technological change is concerned with the use of new technologies, for example the introduction of the transistor or the use of microprocessors. Functional change is represented in the way the technology is used or employed, often perceived as a systems change. An example of functional or systems change would be how the technology could be used to monitor conditions on the plant under control. Both of these themes are identified within the patent material used in this chapter.

4.2. Developments in Industrial Sequential Control

The textile industry in the 18^{th} Century arguably saw the first appearance of programmable industrial automation with the programmable looms of Jacques de Vaucanson (1709–1782) using a drum with punched paper tape around. The programmable loom was later perfected in 1804 by Joseph

Charles Marie Jacquard (1752–1834) replacing the punched paper tape with punched cards (Koetsier, 2001). Electrification and in particular the electric motor revolutionised manufacturing where power requirements could be obtained over long distances and meet intermittent demand (Roberts, 1989). The ensuing development and application of electrical devices and instruments led to further requirements to automate the control of machines as they grew in complexity.

4.2.1. Sequence Control 1900-1920

At the turn of the 20th Century, many innovations centred on the rapidly evolving telegraph and telephone technologies. As these 'telecommunications' technologies increased in capacity and complexity, requirements for automation emerged in the form of complex telephone switch exchanges. One goal was that of reliably automating the switching of telephone circuits through exchanges. McBerty's (1910) patent "Automatic Selective Switching Apparatus suitable for use in Telephone Exchanges" demonstrates where early inventors were concentrating their efforts.

It was realised that the technology used to automate telephone switch exchanges could be applied to industrial processes to achieve the automatic sequential control of electrical and mechanical machines. The "Sequence Switch" (Reynolds and Baldwin, 1915) was designed for use in automatic telephone systems but also "useful for controlling electrical apparatus of many kinds". The improvement Reynolds and Baldwin proposed used insulated and conducting disks. The insulated disk shown in figure 4.1 was cut away to reveal an electrical circuit. Connections were made via brushes to the conducting disk thus effecting different combinations of switching sequences with multiple disks located on the shaft. The shaft was rotated via successive "step by step movements" achieved by a continuously rotating motor that rotated the shaft through two friction discs, one fixed to the motor and the other fixed to the shaft. The friction discs (clutch plates) were engaged via the action of a solenoid which was energised and de-energised at "certain points of its revolution" relating to particular time intervals.

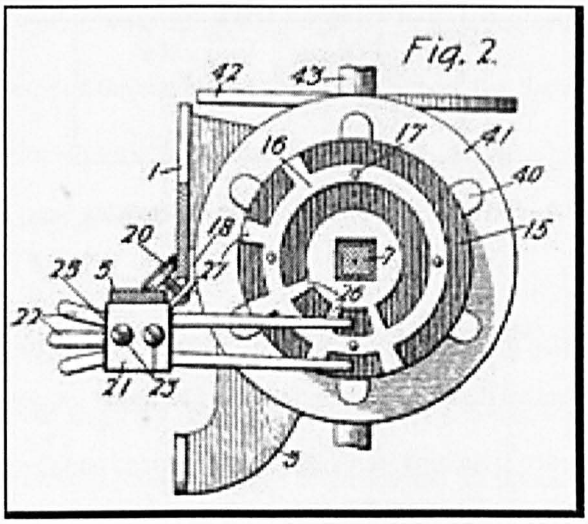


Figure 4.1 – "Sequence Switch"

(Reynolds and Baldwin, 1915)

John Kingsbury (1910) representing the Western Electric Company in the United States, described the requirement for operating multiple circuits for automatic telephone systems. The patent proposed an improvement over the purely combinational logic application of the switch and introduced the means to sequentially control it. The patent stated that cams rotating about an axis to operate switches were not new, but what the Western Electric

Company claimed to be unique, was an electro-mechanically operated switch to operate multiple circuits in different combinations (Figure 4.2).

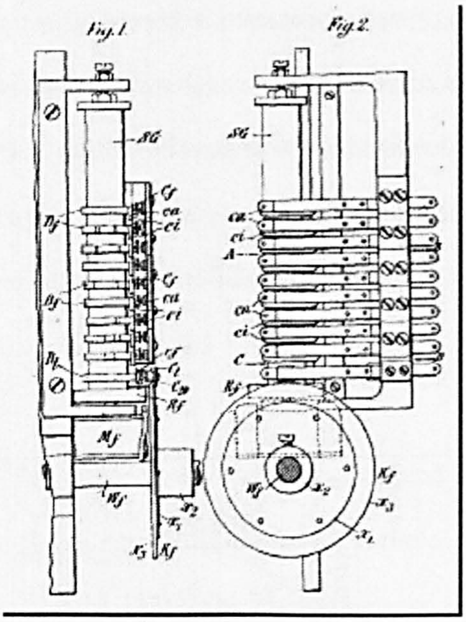


Figure 4.2 – Electro-mechanically operated switch (Kingsbury, 1910)

All three patents (McBerty, 1910, Reynolds and Baldwin, 1915, Kingsbury, 1910) were technical developments relating to functional improvements in telephone sequence switching; reliability was enhanced and the provision more robust means of sequential control. However, an important step for industrial sequential control was the realisation that telephone switching technology had the potential for wider applications.

4.2.2. 1920s

In Chapter 3 of the thesis it was noted that prior to the 1920s, most control applications were thought of as the "switching on, or off, of devices" (Bennett, 1993b). The switching of devices such as solenoids, pumps and valves was accomplished either directly by the operator or by the use of cam-activated switches and electro-mechanical relays.

An example of the direct manual control of sequences, was given by Firth and Kennedy (1914), for traffic management signals... "Signals automatically regulated by a person in charge of the same who will be placed upon an elevated seat, platform or cabin in such a position as to command a view of the traffic". The signals, they explained, were "operated mechanically by any suitable well known method such as electric, manual, hydraulic or pneumatic power". A novel aspect of Firth and Kennedy's invention was that the "signals may be automatically operated by the man in charge of the same as aforesaid but in addition thereto by any automatic device such as road contacts operated by the vehicles, either by the wheels or any adjunct or device thereto".

The automation of traffic signals was clearly a goal and an early improvement was suggested by Miles (1926). Miles was concerned with operating the traffic signals in a timed sequence and his invention provided the "...means for varying or resetting the control so that change in the signals is effected at longer or shorter intervals whilst their sequence of operation is unaffected". He achieved this by using a rotating drum driven by "an electric motor, pneumatic motor, or clockwork motor". A similar improvement was proposed by Vennin (1929) using "rotary drum switches operated by a clock actuated mechanism".

For the control of a machine by a prescribed sequence (sequential control), relays were predominantly used. Electric motors for example, required sometimes complex start-up sequences, ably demonstrated in Cutler-Hammer's (1926) patent:

> "Controllers for synchronous motors are commonly provided with electromagnetically operated switches for selectively establishing low voltage starting connections and high voltage running connections for the motor armature and a switch for connecting the field of the motor to a direct current source." (Cutler-Hammer, 1926)

Cutler-Hammer (1926) described the means to alter the sequence, an arrangement that comprised a "relay that is adjustable for varying the said sequence" and is "adjustable by virtue of it having adjustable contacts". Although it was possible to alter the sequence, modification of the fixed wired connections was still required. Cutler-Hammer's patent demonstrates the complexity of the sequence switching arrangement for a single electric motor (see Figure 4.3) and in particular, highlights the early requirements to provide a solution for automated sequence control, albeit for a fixed function application with predictable demands.

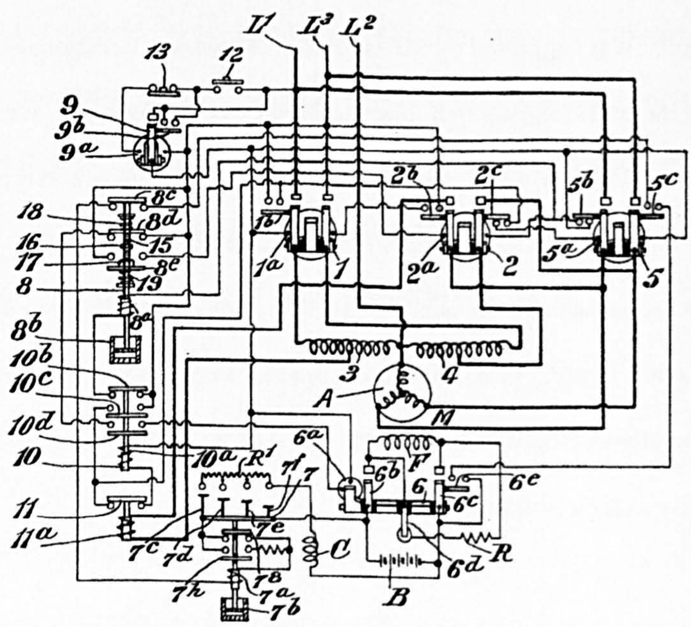


Figure 4.3 – GB251559 "Starting Controller for Synchronous Motors"(Cutler-Hammer, 1926)

A subsequent development was to allow the sequence to be altered by the operator without making extensive changes to the wiring. Hodgkins (1926), for Westinghouse Electric and Manufacturing Company, described the application of an alterable "Sequence Switch" to an industrial process. Hodgkins' invention itself was primarily a technological development, but his patent also reflected a functional change. The patent described the expanding requirements and applications of relay technology beyond telecommunications, and in this instance, the embodiment (tangible representation of the invention) was centred on the sequential control of a blast furnace.

Further functional developments in the form of industrial applications appeared in a patent for the "Automatic sequence control for valves" (Kennedy, 1927), proposing a device that was used to control the operation of 'water gas' manufacture equipment. Water gas was a manufactured gas, produced by passing steam over a hot carbon-based fuel such as coal or coke (Speight, 2007). Kennedy described in the embodiment of his invention, hydraulically operated valves controlled by cam activated electrical switches. The electrical switches were selected and changed according to the positions of the cam sleeves. The principle that the sequence could be altered by changing the cam arrangements arguably produced an early programmable (alterable) sequence controller. Kennedy had identified the need, and provided the means to alter sequences to match process requirements.

A further and significant system development was also noted by Kennedy, in an earlier patent filed in 1921 (Kennedy, 1924). Kennedy identified the requirement for continuously monitored safety systems using 'interlocks' in sequential control, and proposed techniques that are still in current use. Safety interlocks are pre-conditional safety circuits designed to prevent unsafe or unwanted plant conditions occurring.

Kennedy's interlock switches were connected in the 'normally closed' condition (switch closed with the relay de-energised) to represent the 'off' or inactive position instead of normally open circuits (switch closes when the relay is energised). The purpose of this arrangement is to detect breaks in the cabling and safety circuit. In normal operation, the safety circuit is closed, if a fault or unwanted condition is detected, the switch is opened and the control system responds accordingly (for example the system is shutdown or stopped). The same condition is achieved if the connecting cable is severed or broken; the signal is disconnected ensuring that the plant remains in a safe condition. The conventions of using safety interlock circuits and normally closed switch arrangements were used to achieve an automatic "safe shutdown" employing continuous monitoring at 15 second intervals should an erroneous condition occur (Kennedy, 1924, 1927).

Kennedy's control system used electrical switches connected to mechanical valves for status and monitoring. Up to this point, sequential control devices such as the sequence switch had hitherto switched devices according to some single start event (for example a motor start) and conducted a time derived control sequence. Control systems could not respond to changing plant conditions and so inherently employed open loop control. Kennedy had introduced feedback to the sequential control system via the valves and safety systems, reflecting status and condition, thereby closing the control loop.

4.2.3. 1930s

By the 1930s, the means to achieve automated sequential control was effectively based on rotating cams, shafts and drums to open and close electrical switches and relays, however applications were becoming more complex. Combinatorial logic was provided by the switching arrangement and the timing and sequencing provided by electro-mechanical means. Such an arrangement is demonstrated in a patent submitted by Jenkins (1931), detailing an increasingly complex switching circuit driven by a crank-shaft.

Manual Control and Automation

Neuman (1933) described one aspect for the perceived requirement to automate processes in the 1930s where he refers to the problems encountered with manual control methods. The objective of Neuman's invention, automatic sequence control, was...

> "...to provide means whereby the frequently disastrous results of careless or inattentive manual control and operation of processing apparatus of various kinds may be obviated, and excessive waste and expense in the practice of various industrial processes eliminated." (Neuman, 1933)

This perhaps harsh indictment of manual control methods reflects the frustration experienced by manufacturers with the limitations of manually operating processes, in this case the sugar processing industry. Issues with manual control methods were not just restricted to the sugar industry and this view is supported by Bennett (1991), who reports that the adoption of automation and control technologies was increasing during the 1920s and 1930s. With regard to the process industries, the instrument manufacturers argued that "...automatic control reduces "labor, manufacturing and power costs, increase[s] quality and output, and prevent[s] spoilage of work"..." (quoted in Bennett, 1991).

Neuman's solution was to provide multiple electrical control switch units, each unit "being relatively and independently adjustable" thereby automatically controlling the process cycle and duration of several phases of a process (Neuman, 1933). Processes were perceived as integrated but separate steps, each controlled by separate time-linked electrical switches. The automatic sequential control system at this stage was still very much electro-mechanical in nature, primarily based on strategically placed switch contacts, relays and timer mechanisms.

4.2.4. 1940s

By the 1940s, electro-mechanical systems still dominated methods of achieving sequential control. Webb (1948) described a "mechanically driven timing and limit switch mechanism" for soot blowers fitted to boilers in 1944. Webb used "cam-operated switches" on his device that was "made to rotate automatically a predetermined number of revolutions in a forward or a reverse direction." (Webb, 1948) Automatic sequence control was performed on a number of Webb's devices by "the provision of an additional cam and switch incorporated in the switch unit for each device". A feature that Webb described as "secondary" was the wiring and switch arrangement, designed to "afford automatic operation of a number of devices under a variety of conditions".

Supervisory Control

The concept of supervisory control (the high-level management and control of many sub-ordinate controllers) also appeared in patents in the 1940s. Appel (1941) proposed a solution to manage multiple controllers to provide "independent groups of controls for separately effecting recurring operations of a process". Appel referred to another patent, proposed by the same company (assignee), Houdry Process Corp, that provided the subordinate individual valve controllers dedicated to a single valve (Thomas et al., 1940). Appel provided "an electrical timing mechanism having isolated control circuits for operating different mechanisms simultaneously and non-interferingly". The objectives stated by Appel were to provide a system capable of controlling processes having different cycle times and number of operating steps and to provide independent "groups of controls" for separately "effecting recurring operations of a process". Appel's "successive switching arrangements" provided supervisory controls to the individual "Valve Cycle Timer Apparatus" (controlling the valves) detailed by Thomas et al. (1940).

The significance of Appel's patent is a functional development rather than a technological advance in sequential control. Existing technology was organised in line with the concept of a higher-level control system overseeing the operation of independent fixed-function controllers. The technology employed is still electro-mechanical utilising a "moveable contact arm" for the electrical switching, driven by a motor with variable gearing for the timing.

Introduction of Electronic Control

Electronic devices such as the vacuum tube (valve) began to appear in control system patents in the mid 1930s, this technological development was demonstrated in a 1935 patent by Alfred Stark (1940). Stark proposed a circuit switching arrangement using vacuum tubes for multiple current consumers, for example street lamps. However, the use of electronic devices in sequential and logic-based control systems appeared a little later in the early 1940s for sequential controller related patents.

William Hills' (1949) patent for General Electric gives a good account of the limitations of relays and electro-mechanical devices when it was filed in 1948. The embodiment of Hills' invention is applied to resistance welding, an application he claimed, that required accurate timing and high switching speeds. Hills suggested that pure electro-mechanical relay switching was either inaccurate in terms of timing, or not fast enough to control the process adequately.

Hills used the electronic valve-based "Timing Apparatus" invented the previous year by Maurice Bivens (1949). Maurice Bivens was a prolific inventor, filing 44 patents for General Electric[8]. Hills' patent demonstrated that technological developments in electronic engineering were now being employed in automation control systems.

[8] Source Esp@cenet citation search 16/01/12.

4.2.5. 1950s

Modularity and Maintenance

Standardisation and modularity were (and are) important concepts in industrial control systems because they aided the maintenance of equipment and provided potential improvements in reliability. Maintenance was supported by the use of modular, interchangeable spare parts that could be easily exchanged by the engineer. Although this was not a particularly a new concept for industry (it was one of the drivers of mass production) and certainly not a development in applying new technology, it was a functional improvement to sequential control systems. A modular system design meant that a control system could be built from a standardised set of components. The same control system design could not only be used for different applications, its functionality could also be expanded to meet future requirements.

Aston Electrical Products Pty (Anon, 1952) provided an example of this approach with their patent filed in 1950 concerning a development to the control and automation of the resistance welding machine. Aston proposed an Interchangeable time-delay unit within a standard "basic" relay panel (shown in Figure 4.4).

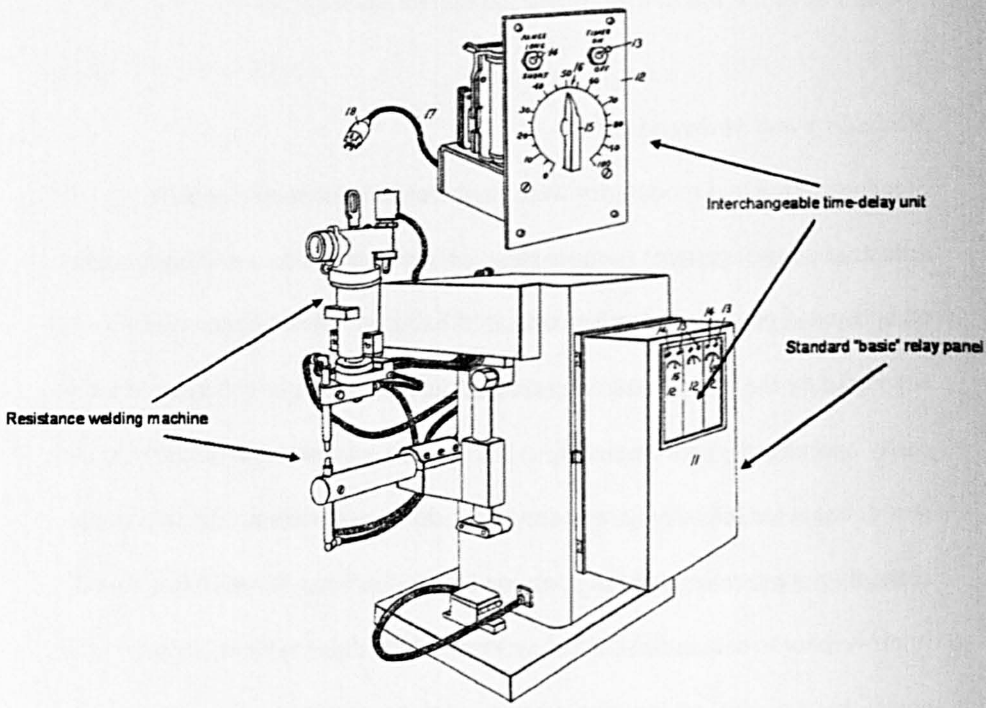


Figure 4.4 – GB672381 Multiple sequence electric control for use with resistance welders. (Anon, 1952)

The essence of this patent was to use modular, interchangeable units (a concept that was adopted in more recent PLCs as separate input/output and function modules). The patent recognised the benefits of standardising control modules to perform specific functions. The relay panel itself managed the overall logic control (combinatorial) through relays and contactors. Sequences were developed from the timer modules.

In detail, Aston Electrical's patent proposed:

> The "Contactor" panel is of a standard design ("main frame") and supplied as a "basic unit". It has:
>
> 1) Control cabinet case.
>
> 2) Two relay panels… "…one basic to the control, and the other an auxiliary and termed the interlock panel"

> The interlock panel is a complete unit and can be blanked off when not required for a particular application.
>
> Two different contactors (mechanical switch or electronic) can be fitted and readily exchanged for each type if required. Electronic is an "Ignitron" contactor.
>
> Inter-panel plug connection through keyed octal plugs and sockets to connect the contactor and relay panel (combinatorial logic).
>
> "The [keyed] plugs, apart from making such a change a simple matter, allow it to be done in [a] foolproof manner"
>
> (Anon, 1952)

To summarise, Aston Electrical's patent proposed a complete automation system comprising: a basic and standard control panel design (similar in concept to a modern PLC system); interchangeable modules for the contactor and timer units designed for general control but embodied in a resistance welding application; and the keyed octal plugs eliminated potential connection errors by maintenance personnel. This standardised approach and in particular keyed octal plugs enhanced the maintainability of the control system.

Requirements for Automatic Control

William Roberts (1956) of Brookhurst Switchgear Ltd gave an account of the general requirements for the automatic control of manufacturing plant perceived in 1953. The patent essentially centred on motor control requirements for industrial plant but also applied to material handling equipment.

> "...in the automatic handling of materials, the routing of the material will vary according to the requirements through the processing machinery or to storage into hoppers, etc. Similarly, installations using steam, air or fluids, for many purposes require electrically-operated valves which can be conveniently cracked, opened or shut in a sequential manner". (Roberts, 1956)

What makes Roberts' patent interesting is not his detailed description of the invention, which is largely based on improvements to existing technology, but his description of the key attributes he identifies for automatic control.

> "Important factors in the controls for such sequence schemes are that:-
>
> (a) They are conveniently grouped together;
>
> (b) They occupy relatively a small space, but at the same time provide for full access for connecting up, inspection and maintenance;
>
> (c) They can easily be adjusted to control variable sequences;
>
> (d) They indicate the sequence order of operations;
>
> (e) They allow the motor control gear grouped or on individual starting panels, to be placed to suit the installation, and with the controls either combined into the grouped starting panels or separately mounted;
>
> (f) They allow the number and grouping of the connecting wires from the controls to the motor starting panels to be the economical minimum.
>
> An object of the present invention is to provide a control system and control gear which meets some or all of the above requirements" (Roberts, 1956)

Roberts clearly demonstrated that his objective was aimed at improving the automation system. More particularly, the points he made, related to the maintenance capability of the system. Of particular interest is: ease of access for inspection and maintenance (b); ease of adjustment (c); and the indication of the operation sequence (d).

The factors stated by Roberts (1956) were indicative of the industrial automation requirements perceived in 1953 originating from the end users of automated industrial processes. Roberts, together with Aston Electrical Products (Anon, 1952), highlighted the goals of introducing standardised equipment and for providing systems that could be maintained at the point of use.

Roberts' (1956) patent shows the developing functional systems used with automated control technologies. Sequential control systems employed with industrial processes were becoming modular with more emphasis placed on the maintainability of such systems.

Cams & Electronic Devices

Prior to semi-conductors, the Times (Anon, 1947) reported that thermionic vacuum valves were used for the control of production lines, or more specifically for "the coordination of a long line of electrically driven machines operated in sequence on the product". Electronic components were used for manufacturing purposes, particularly for the electric welding machines that were "extensively employed in motor-car and aircraft factories". Electric welding machines developed "...a large demand for electronic control devices for timing, current adjustment, interruption of current at regular intervals, and heat control".

In the 1950s, sequence control was still largely achieved by electro-mechanical means and further evidence of this appears in Roessler's patent (1958), where sequential plant control was achieved by the traditional technologies of relays and cams. Roessler's development proposed a "Sequence Program Control" highlighting the requirement for the control system to be alterable or programmable. Roessler achieved this by altering cams on multiple "cheaply made units".

Apart from material handling applications, Roessler also confirmed the list of typical industrial applications requiring sequence control:

> "Typical illustrative but definitely not limitative applications are in[:] material handling operations; electric, hydraulic and pneumatic program control; injection molding procedures; material forming press procedures; and successive mechanical and chemical operations on work" (Roessler Jr., 1958)

Roessler also stated the further requirement that the control system must be robust and immune from the vibration and shock of heavy machinery. Clearly, this was perceived as an important requirement and one that could preclude the direct local application of computer control in the mid 1950s due to the fragility of the early computers. Cam-based switching such as that proposed in Roessler's patent, shown in Figure 4.5, was a robust technology and could be placed directly in the area of application at the point of use, meeting the requirements proposed by Roberts (1956).

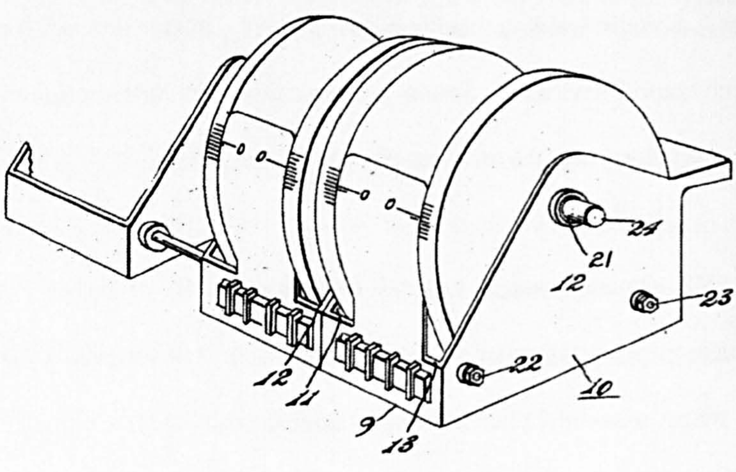


Figure 4.5 – Cam-based switching arrangement (Roessler Jr., 1958)

Motor driven cams however did have their disadvantages, notably the physical size and relative complexity when multiple cams were used. Pierz (1961) proposed an electronic solution for the programming and sequence control of fuel burners, "... which will incorporate an electronic timing circuit for timing the several events, thereby eliminating the use of motor driven shafts, cams and electrically heated bi-metal switches for the timing operation." Different timings were achieved by altering the resistance values to the timing circuit. However, the fuel burner controller was a fixed-function device for a standard application and effectively it was only the timing intervals that could be altered. Pierz's timing circuit was nonetheless an important development supplanting the motor-driven cam applications proposed in earlier patents.

It was not simply the fact that Pierz's timer was an important development, timing itself was a problem. Prior to electronic timers, various ways have been used to derive process timings that included clockwork devices, electric motor driven cams or drums and heated bi-metallic strips. In the end, the technological development of digital electronics helped to solve the timing problem. With digital electronics and computers with their ability to provide fast and accurate counters, timing (and timers) became more sophisticated and were employed in the control of increasingly complex sequences.

4.3. The use of computers for Sequential Control

Hugh Millis Jr (1965) filed a patent for a "General purpose parallel sequencing computer" in 1959; it was not published until 1965, taking nearly six years. Millis gave an account of the application of digital computers for industrial automation and remarked that in the field of automation, digital computers have been generally applied to large continuous flow processes such as oil refining and "other related essentially chemical plants". Millis however, described what he regarded in 1959, as a developing and different application for digital computers:

> "Another and basically quite different area in automation, but one having considerably wider application, may be defined as sequence processing or sequential control in which individual, discrete actions of interrelated mechanisms are required on a relative time ordered or time arranged basis to provide a unified, final operational result" (Millis Jr., 1965).

The essence of this "different" approach to automation was based on the premise that "Sequencing as a basic process, however, is essentially digital in nature..." (Millis Jr., 1965). Millis further explained that

> "It is this digital form of sequencing which has evolved more recently as the second basic approach to the mechanization of sequencing problems, owing primarily to the inflexibility and the inherent limitations as to the types and number of elements which can be simultaneously handled by the analogue or ingenious mechanism type of sequencing."

The description of the application of digital computers to industrial processes in Millis' patent gave an account of how computer technology was applied to seemingly disparate industrial automation applications. The patent itself was concerned with a technological development, the use of digital computers to achieve sequential automation, but according to Millis, this was a recently evolved form of digital sequencing. Millis also referred to the inflexibility of pre-existing technologies used to achieve sequential control... presumably Millis' "ingenious mechanism" referred to the complex cam and relay based systems discussed earlier!

4.3.1. Cyclic Program

In addition to Millis' proposal that digital computers can be applied to sequential control applications, other developments in digital computing were emerging. IBM, through Theodore Cox (1965), detailed some important technological concepts relating to computer diagnostics and test. Cox's patent not only described and provided a useful tool to aid the diagnostic capability for computer control systems, but also influenced the notion of a 'cyclic program'.

Although Cox's patent related to digital computer diagnostics rather than sequential control, the concept of a cyclic program is an integral feature of the PLC. In essence, the cycle of a PLC is:

1) Read inputs status and store (in input memory or image).
2) Apply the sequential logic program to the input image values and set resulting output values in the output memory or image.
3) Write the output image to set the outputs.

This sequence of events is repeated or cycled continuously as shown in Figure 4.6.

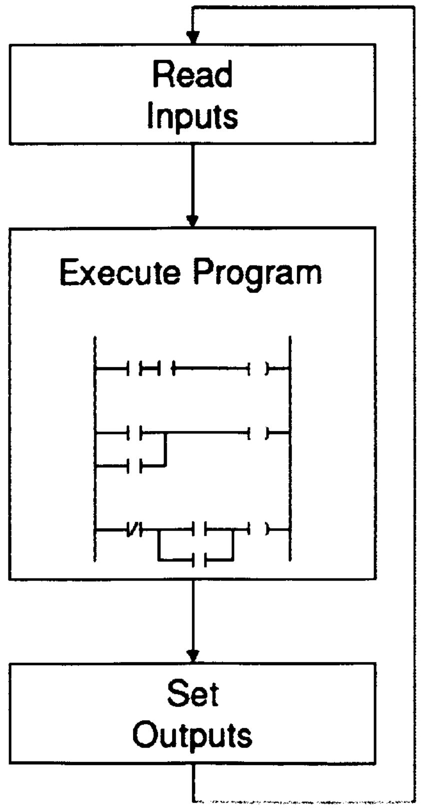


Figure 4.6 – PLC Cyclic Program Operation

4.3.2. Sequence Controllers

Although by the 1960s digital computers were being applied to sequential control problems, there were relatively few applications in industry. Computers were, on the whole, only applied to the larger process industries such as chemical and steel. Computers were also expensive and required programming expertise not readily available or necessarily affordable to many small to mid-size organisations. Edward Yetter (1966a) provided a highly reliable low cost controller that was "...adjustable in programming to accommodate a great variety of control conditions".

Yetter stated two important objectives relating to the intended users of his invention. Firstly, the controller provided an arrangement so that the "...circuitry can be modified without difficulty by relatively unskilled personnel to accommodate changes in process control pattern". This statement clearly indicated the requirement to allow non-specialists access to the control system, albeit that they were "relatively unskilled"! Presumably, Yetter was referring to the plant engineers and technicians without computer programming knowledge.

Yetter's second point described the circuit layout which "...lends itself to a corresponding graphic depiction of the process itself" (Yetter, 1966a). This was an important step for the programming of sequential controllers and it allowed the engineer to program the controller in a familiar format that related to the physical plant under control. The benefits of this approach were significant: it gave the engineer the ability to program the controller in a manner that was directly related to the plant and sequence under control; it also improved the maintainability of the control system because faults could be diagnosed without the need to rely on the abstract world of computer languages.

In terms of PLC development, Yetter also stated a further objective previously highlighted by Roessler, that was to become an important, if not essential feature of the programmable controller - **robustness**. The controller was to be "...compact in space requirements and rugged enough to withstand successfully adverse plant environmental conditions", showing clearly that the controller was to be located directly within the industrial environment itself.

Yetter also developed a more sophisticated "Programmed Batch Sequence Controller" (Yetter, 1966b). Both of Yetter's controller patents were closely related and contained the same objectives of accessibility to the plant engineer (functional improvement) and ruggedness in the industrial environment (technological development).

4.3.3. Alarm and Status Monitoring

An important aspect of monitoring plant status is the notification and display of events and alarms. A common method employed with relay-based control, used active hard-wired displays commonly known as "mimic panels". The displays utilised lamps and buzzers to represent the plant and process under control, which were used to indicate when an erroneous condition on the plant has occurred. An error condition was detected by setting limits on instruments or signalling a status when undesired conditions were met; effectively these alarms were the result of built-in diagnostics.

With the relay panel alarm signals hard-wired, they are difficult to change without considerable plant downtime. Alarms are also generally limited in number due to the size and complexity restrictions imposed by the wiring cabinets; additional relay logic takes up valuable space.

Macarthur (1967) provided an interesting insight into one potential solution for detecting alarm conditions and displaying alarm messages. In essence, a computer was used to control a slide projector by activating a slide to display an alarm message. The slide is selected by comparing a status or data value using a 'look-up table' held in computer memory and accessible to the program, containing the relevant combination of conditions pertaining to an alarm situation. The benefit of using a computer was that the alarm criteria could be altered in software by changing the program.

The idea of using a computer controlled slide projector was quite novel. It provided considerable flexibility and allowed the development of more complex diagnostics through system status relationships defined within the software. Through using slides, relevant alarm conditions could be communicated to the operator or engineer without ambiguity. At the time of filing his patent in 1966, Macarthur observed that many computer monitoring systems relied on computer printouts which were deemed by Macarthur to be "limited".

4.3.4. The Programming Device

A defining feature of a PLC is that it is programmed by a separate and detachable programming device. Two of the main advantages are:

1) One programmer unit could be used to program and monitor many controllers. It was less expensive to purchase a single programming device that could be used for all systems rather than one for each controller.

2) Removing the means to alter the program on running/operational controllers made the system more secure from both accidental

and intentional (sabotage) disruption. Only key personnel would have access to the programming tool so use could be strictly managed.

The main disadvantage however, was the lack of a fixed access point for the engineer (although in a modern system, a programming device can connect over a network). In order to diagnose faults or modify the program, the programming device needed to be taken to the controller on site and physically plugged in.

Clayton et al. (1968) described the concept of an external programming device in 1965 with the rather confusingly named "Remote Calculator". Clayton et al noted that due to the development of "high speed digital computers" in the 1960s, communication between the computer and the user had become so complex and demanding, that it would require the user to be a highly skilled programmer.

It had become apparent to Clayton et al that for engineers and scientists to access digital computers, improved techniques of "user-to-computer communication" be sought. Their device, they explained, provided a means to utilise a computer for problem solving "without having a special knowledge of programming". It related to remote input and output apparatus cooperating with a computer so as to "appear to the user to be autonomous calculating devices" (Clayton et al., 1968). The Remote Calculator is shown in Figure 4.7 below.

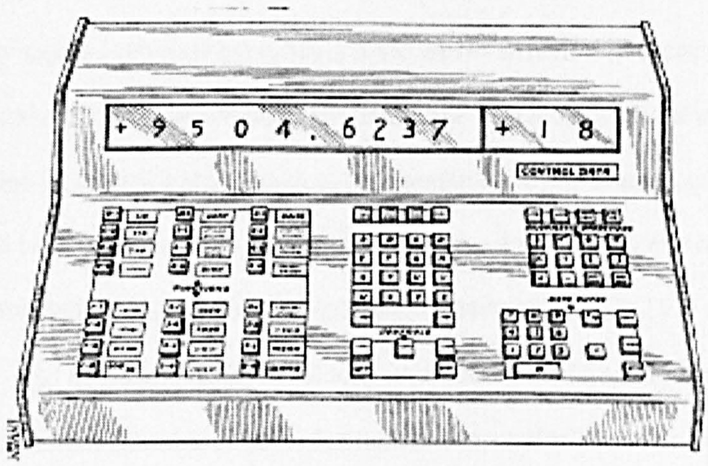


Figure 4.7 – Remote Calculator (Clayton et al., 1968)

The concept of using a remote device to provide a user friendly interface to a computer had the advantage that the interface could be tailored to the intended audience. The remote device simplified the otherwise complex interaction of programming the computer directly. This feature enabled non-computer specialists to program computers.

The programming devices for industrial controllers developed using this concept, enabling engineers to program and interrogate control systems in an understandable way. The benefits for maintenance staff meant that modifications and fault diagnosis could be carried out immediately on the plant, without having to import the necessary and expensive programming expertise required for complex computer systems.

4.3.5. Development of the Programmable Controller

The number of applications using computers for process and sequential control expanded with the decreasing cost and increasing reliability of the digital computer. However, problems were encountered when controlling multiple processes. Inefficiencies were highlighted with the way the computer handled and coped with inputs and outputs. George Smeallie of the Bailey Meter Company attempted to provide a solution to this problem.

Smeallie (1972) first described the way existing computers, available in 1969, handled the control of multiple processes. Each process to be controlled requires "at least one execution" of the command program and multiple processes are controlled sequentially. The computer controls each process by performing a series of tests on the process "output variables", appearing as inputs to the computer. The collective tests for each process under control are generally known as a "test and branch program". An executive program is used to select the appropriate "test and branch program" to control each respective process. The control of several processes is achieved in a "sequential fashion according to a predetermined priority schedule".

According to Smeallie, the problem with accomplishing the digital control of multiple processes using the above method was that "multiple executions of the test and branch programs are required" so the frequency of the "control service" to each individual process depended on a number of factors: the number of processes to be controlled; the complexity of the control task; the variability (size) of the programs; and the finite limit of the processor speed.

To paraphrase Smeallie - because the programs were executed sequentially, all of these factors combined, could affect the frequency the control task was serviced by the computer. Put another way, the digital computer could not service the control programs in an even manner and it was possible that this could affect the real-time control of the processes. A further issue identified by Smeallie was that of memory – "the memory requirements are inflated because of the need for memory locations for storing the data and programs required by this type of sequential data processing".

The solution proposed by Smeallie was to provide a process control system employing a central processor with "an apparatus for and method of performing simultaneous control operations on a plurality of processes". This was achieved by providing an input interface for "simultaneously interrogating a plurality of sets of process signal variables" (input data) and assembling and organising the input data for use by the central processor. Smeallie also treated the outputs to the processes under control in a similar fashion in an "output interface" for connection to the external processes.

Effectively Smeallie had devised a method that employed the management and control of input and output data as separate and independent entities (or images), freeing up the processor to resolve the control logic and mathematical functions. This method was a technological development that was to be used in a number of future PLC patents and was cited 27 times including Dummermuth's (1976) "Programmable Logic Controller" patent.

4.3.6. Some Problems with Digital Computers

The transition from the traditional cam or relay based electro-mechanical control system to digital computer control was not entirely problem free.

Lawson et al. (1970) identified the problem of mismatched input hardware and its corresponding memory status in digital systems. The problem related to the change in input device status (for example 'ON' to 'OFF' or 'OPEN' to 'CLOSE') becoming out of step with the value held in computer memory. This "disagreement" between the operational state of the hardware and the status in memory "may lead to a rapid degeneration of the control capability of the processor" and the effect of this disagreement could be so substantial that "...further control of processes is impossible" (Lawson et al., 1970).

The solution proposed by Lawson et al (1970) was to provide a method of identifying potential errors (discrepancies) between the hardware and memory status by keeping an independent record in memory. If an error was detected "steps may be taken to re-initialize certain data on the basis of algorithms and data stored in a secure or permanent memory". The objective was to "increase the reliability of a program controlled process control system" and the patent describes a method of error detection and correction by applying an algorithm.

A further problem with digital computer control was identified by Leroy Wirsing (1971). Wirsing explained that in 1970, sequential control techniques were limited to two principal types: wired logic devices such as electronic or relay circuits and stored program computers. According to Wirsing, "wired logic" had a fixed logical arrangement which must be physically altered if a different arrangement was required, which can be both difficult to achieve and expensive.

The stored-program computer on the other hand "requires extremely high-speed and complex logic elements with associated packaging problems. Thus substantial quantities of logic are usually needed to implement the 'stored-program' system". The main difficulty, Wirsing explained, was that the "wired-logic designer" had little or no knowledge of computer programming so a "computer programmer in addition to the logic designer is needed to implement a particular program" (Wirsing, 1971).

Wirsing's solution was to provide a control system that is "readily implemented without the need for sophisticated programming or complex fixed-wiring structures" (Wirsing, 1971). Wirsing proposed a system whereby the logic elements were constructs in software rather than the physically hard-wired alternative or produced as a result of an overly sophisticated program. In essence, the approach was to use software representations of logic elements (virtual logic elements) that could be built up to represent electronic logic components (for example "AND" or "OR" gates). After some initial training, the logic engineer could then build his own combinatorial logic circuits.

The solutions proposed by Lawson et al (1970) Wirsing (1971) were to form integral features of what was later to become known as the Programmable Logic Controller.

4.4. Chapter Summary

This chapter reviewed the key developments appearing in patent literature of sequential control technologies leading to the emergence of the

Programmable Logic Controller (PLC). Patents were used throughout the chapter as a source to highlight the developing technologies and systems relating to sequential control technologies. The patent material covered the 20th Century up to the emergence of the PLC in the late 1960s and early 1970s.

Patents are a rich source of material that not only describe new technologies, ideas and systems but also details why these inventions were deemed necessary and important. The issue of a patent is no guarantee that an invention turns into an innovation and becomes a commercial success, however when reviewing historical patents, you do have the benefit of hindsight! Patents also provide an account of the developmental state of a technology and indeed its technological system (who and what uses the technology and why) when they were filed, providing a useful historical account. As such, the patents detailed in this chapter have provided an insight into the development of sequential control and identified key steps and reasons for its advancement.

This chapter has revealed and discussed the development of sequence control from simple switching to elaborate electro-mechanical systems. The chronological order of the patents demonstrates the increasing complexity of applications (machines and processes) requiring sequence control and how the solutions were developed to respond to that demand. The adaptability of the control system was important and in order to meet this requirement, the control system was developed to make it "adjustable" and "alterable".

Maintenance was another significant aspect that was important not just to the manufacturers of the control system, but also to the end user. Solutions were developed for ease of maintenance that included concepts of modularity and standardisation. Additionally, some control systems were used in harsh industrial environments so the sequential control system needed to be robust.

When it was attempted to use computers for sequential control, it was identified that programming was a rare and specialised skill, a skill not readily available to the factory or held by the maintenance engineer. Giving the "ordinary engineer" the ability to program, monitor and test these control systems was an important objective, and this review of patents demonstrates that practising engineers (patent authors and end users) had influenced and shaped the development of this control technology.

This study of patents, has in particular, demonstrated the application of old in-use technologies (electro-mechanical cams and relays based control systems) and their integration with new emerging technologies such as electronic devices and computers. The next chapter, chapter 5, discusses the further integration of old and new technologies with the emergence and development of the PLC.

Chapter 5 – The Emergence and Development of Programmable Controllers

5.1. Introduction

This chapter continues with the development of sequential control systems with the emergence and development of electronic systems and computer-based programmable logic controllers (PLCs). Again patents have been the primary research resource in identifying the principal historical, technological and functional (system) developments. Further developments are investigated using the patent literature which established the PLC as an identifiable and distinguishable technology. Following the establishment of the PLC, further advances to the PLCs technological system are identified and reviewed.

5.2. Introduction of Programmable Controllers

5.2.1. Modicon

The first patent to describe the common features now recognisable as the PLC was the "Digital Computer-Industrial Controller System And Apparatus" (Fletcher and Rosseau, 1972) filed in 1969 by Modicon. The patent defines the key characteristics of the first commercially recognised form of the PLC. This was an important patent in the development of the PLC and it was cited in 72 subsequent patents as prior art (Espacenet, 2012a).

Fletcher and Rosseau discussed the application of digital computers in industry and observed:

> "Ever since the large scale application of digital computers to business and scientific problems in the late 1950's, the application of general purpose digital computers to industrial process control has been considered a desirable goal. Except for rather specialized situations the goal has largely not been reached." (Fletcher and Rosseau, 1972)

The reason, they argued, was the available commercial computers being too large and expensive except for the "most complex of processes" and that the "programming of such computers for a particular process according to prior art methods is a nearly Herculean task". Moreover, many hours and much money were expended making the computer work in the industrial environment, only to be repeated again if any large changes were subsequently required.

According to Fletcher and Rosseau the reliability and maintainability of industrial computers was deemed to be poor in industrial environments, due in part to frequent breakdowns and the requirement for the technical support of external specialist programmers. Despite their comments on digital computers, the embodiment of the patent incorporated a DEC PDP-8/L as the "basic element" of the "industrial controller"; the PDP-8/L was a relatively small but reliable general-purpose digital computer.

Although the PDP-8/L computer required programming in the conventional manner, in order to make the system accessible to the "ordinary engineer", the system was pre-programmed with an "executive program", the purpose of which was to provide the means to interface to, and run, programs developed externally to the computer. The engineer wrote a "special-purpose program" by using a "programming panel"; this program was in turn managed by the executive program that was fixed and inaccessible to the engineer. Although Fletcher and Rosseau's "industrial controller" was based on a general-purpose computer, the device became specialised; it was still programmable but in a limited way.

The purpose of the industrial controller was to perform the control actions achieved at that time by relays, counters and timers. The controller was programmed by a separate programming tool or "programming panel" (Figure 5.1) that the "ordinary industrial engineer with no computer programming experience can use" (Fletcher and Rosseau, 1972). Programming was achieved via thumbwheels and push-buttons. This enabled the operator to relate a "ladder diagram" to the controller by directly transferring the ladder diagram notation, familiar to the engineer, line by line, via the panel. In Figure 5.1 the left-most thumbwheel was used to select the ladder diagram line number and the control elements (contacts, counters and timers) were selected by the function push-buttons. Values for the counters and timers were entered by the "Reference Number Function" thumbwheel (bottom).

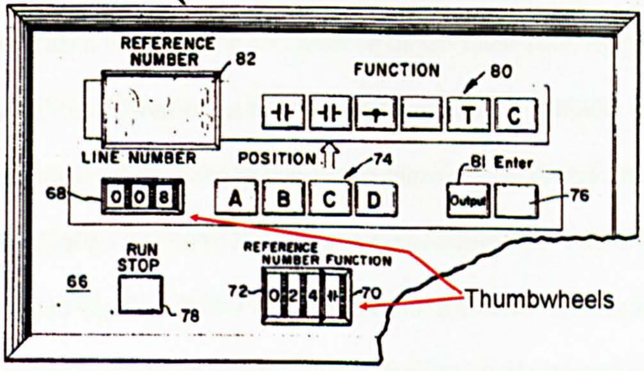


Figure 5.1 – Programming Panel (Fletcher and Rosseau, 1972)

Cyclic Program

A feature of the controller was the embodiment of a "cyclic program" (see Chapters 2 and 4), under the control of the executive program. When a ladder program was completed by the user (engineer) and the controller set to "run", the executive program would repeatedly run or cycle the ladder program. This gave the controller the ability to control the process at regular intervals governed by the time to complete a full cycle of the program. The program length set the cycle time but as Fletcher and Rosseau (1972) stated, the "speed of direct response by the computer is immaterial because the cycle is fast enough to update the outputs in real time". Of course, this was in comparison to the relay and timer circuits used in 1969 and the computer cycle time to perform the sequencing function, appeared faster in operation than the equivalent relay circuit.

Typically, relay switching times were around 10-20ms and Keller (1962) gives "often less than 1ms" for "glass-enclosed contacts" (reed relays). The "magnetic core memory" cycle time for the DEC PDP-8/L was typically 1.6μs, performing simple addition in 3.2μs (DEC, 1970). However, relay circuits immediately act upon their inputs and in direct response, set outputs, effectively processing the logic at the same time, or in 'parallel'. The computer on the other hand has to process one command at a time in sequence, or 'serially'. Timing is therefore an important issue. Although a number of processor cycles would be needed to perform the sequencing task, the DEC PDP-8/L minicomputer appeared to be fast enough to compete with relay switching times because it was seemingly able to emulate the physical characteristics of the parallel processing of the relay circuit.

Telephone Support

The provision of remote technical support via a standard telephone line is another aspect of the controller stated in Fletcher and Rosseau's patent. The program could be transferred over standard telephone lines to a "central station" for diagnostic and maintenance purposes. The provision of telephone support was a feature that was already in use supporting the larger computer systems and applications for business, scientific and larger process control purposes. It was however, quite novel for the smaller sequential control applications. Fletcher and Rosseau (1972) explained that if the computer failed (apparently not an uncommon concern with early digital computers) the program could be quickly restored over the telephone lines

> "Another problem of the prior art is that the prior art computer industrial controllers are subject to breakdowns.... In case of loss of memory the programs may be supplied by telephone on short notice." (Fletcher and Rosseau, 1972)

Reliability and maintenance were deemed to be very important for the industrial control application.

Interrupts

On 2nd July 1970, Modicon filed two further patents for a "Digital Computer", (Greenberg et al., 1973, Morley, 1973). The patents are almost identical and clearly relate to the same device. A schematic diagram of the "Digital Computer" is depicted in Figure 5.2. However, although the two patents are related, the claims do differ: "Dick" Morley's patent (1973), referred to the design and operation of a new "digital computer" for industrial and scientific "real time" applications; the second patent by

Greenberg, Fletcher, and Morley (1973) highlighted a method of interrupting a program on the digital computer. Morley appears as the inventor in both patents and was clearly central to the development of Modicon's "Digital Computer".

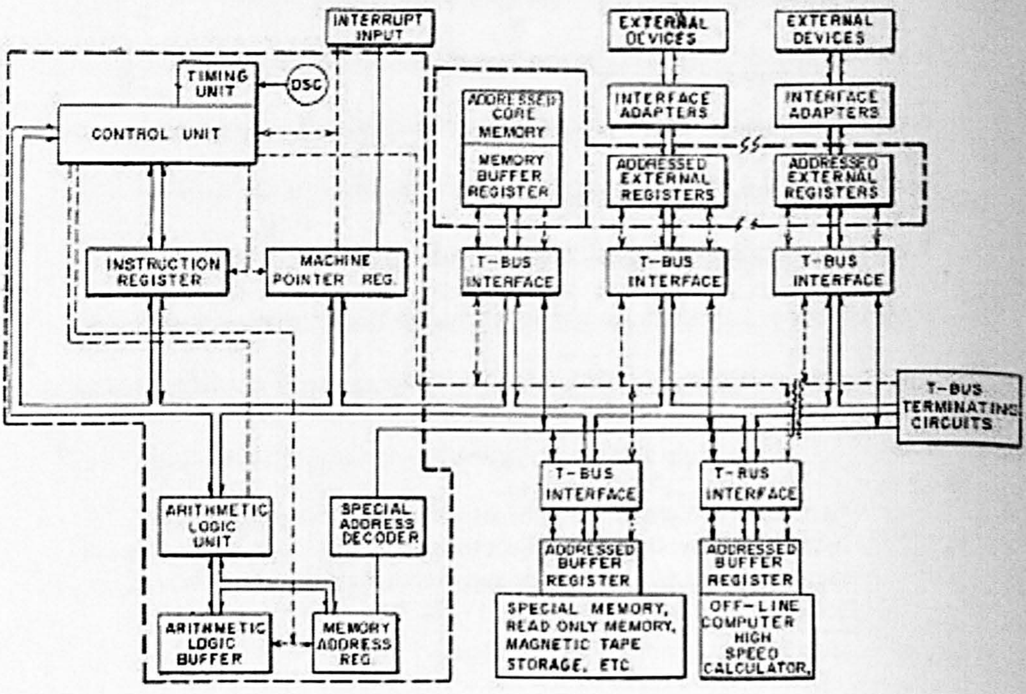


Figure 5.2 – Block Diagram of the Modicon Digital Computer (Morley, 1973, Greenberg et al., 1973)

Both patents stated that digital computers of the "prior art" were "originally conceived as batch processors of data" and "not adapted to process continually updated data, that is they were not capable of operating in real time" (Greenberg et al., 1973, Morley, 1973). This was a general problem for real-time computing, but in the context of programmable controllers, they described two problems with prior art computers:

1) The data transfer to and from external devices.
2) Handling the interrupt and sub-routines servicing the external devices.

The first problem, noted Morley (1973) was that "In the prior art all data must pass through an input/output register and be placed in addressed locations in the main memory before processing". This process involved the movement of data "from the external device to the input/output register, and then to the memory location". The transfer, under the control of the central processor, took up valuable processing time. This difficulty, explained Morley, was "completely overcome by assigning to each external device a register or portion thereof which has a memory address and is in fact addressed according to the common scheme and part of the main memory of the computer". Thus the data held in the directly accessible register locations was always accessible to the central processing unit (CPU).

The second problem was a general real-time computing issue that a number of computer manufacturers, including DEC, were attempting to solve. The problem related to the fact that the external devices (inputs and outputs) required servicing by a "special subroutine" and that "the central processing unit must interrupt its current subroutine at an appropriate place and then start the subroutine and go through it" (Greenberg et al., 1973). The "central processing unit must return to its original place in the previously current subroutine or start the program over at some arbitrary point" they explained. The problem arose if the program started from an arbitrary point when situations could occur "in which certain parts of the program are not performed often enough" leading to potentially long response times. Alternatively, if after the subroutine was completed, the program was returned to the original place, "rather elaborate program provisions have to be made" because a program may potentially have "interrupts of interrupts of interrupts". Many permanent registers would be required for storing the "accumulator, program counter, and program counter save" register information (Greenberg et al., 1973).

The solution proposed for the 'Digital Controller' for the second problem, used a "nested interrupt handling mechanism" (similar in concept to that used on the DEC PDP-11 mini-computer, described by Bell et al., 1970 (Bell et al., 1970)). Greenberg et al (1973) used main memory locations to perform the functions of the accumulator and program counter stored in "save registers" that were "specified by a single unaddressed machine pointer hardware register". Each subroutine would then have its own set of registers, undisturbed in the main memory and specified by the pointer becoming a "dedicated machine". "The machine enters a catatonic state, and can be reactivated at the same point in its programmed operation by merely replacing the address of the program counter in the machine pointer".

A further and powerful consequence of this new scheme was that "another computer can be connected to an addressed external register and the second computer is thereby directly addressable according to the common addressing scheme of the first computer". Their invention had provided an "architectural scheme" to enable the connection of many computers "possible only with machines initially designed as large scale" (Morley, 1973, Greenberg et al., 1973).

In summing up both patents, Greenberg et al (1973) and Morley (1973) remarked that their invention, the "Digital Computer" was:

> "particularly adapted to scientific and industrial applications requiring continual input and output of data in "real time" to and from the computer... The machine is thus particularly adapted for real time processing of continually updated information rather than to the batched processing commonly employed by commercial business oriented computers." (Greenberg et al., 1973, Morley, 1973)

Finally, looking to the future, Greenberg et al (1973) and Morley (1973) predicted the impact of the developing electronic integrated circuit technologies, noting "Those skilled in the art will realize that the computer's main memory at the state of present technology is usually a core memory. Integrated circuit memories may soon come into greater use." Integrated circuits and the development of the microprocessor were to have a major impact on industrial control systems.

5.2.2. Digital Equipment Corp (DEC)

In September 1972, one month following the publication of Fletcher and Rosseau's (1972) patent, DEC published a patent in the United Kingdom (UK) for a new "Control System" (Ricketts Jr. et al., 1972). The priority date on the patent was stated as 7th Jan 1969, over eleven months earlier than the filing date of Fletcher and Rosseau's patent. At the time of filing, it was known that General Motors (GM) Hydra-matic division had issued a specification for a sequence controller for their production line at their Detroit manufacturing plant (Clare et al., 2005, Hendricks, 1996, Erickson, 1996).

Patent records show that Ricketts et al (1972) published their patent in a number of countries in addition to the United Kingdom, including Germany, Japan and France using an original priority number raised in the United States (US), but did not actually publish the patent there. This is known from a later patent published by Ricketts Jr. et al (1973) for a "Programmable Machine Controller", providing the following partial explanation… "This is a continuation of an application Ser. No. 789,585 filed Jan. 7, 1969 for a Control System now abandoned".

Why this patent was abandoned in the US is unclear but one could speculate that it coincided with the time that the GM controller requirements specification was issued. Modicon were also bidding for the same contract. Perhaps it is reasonable to assume that there were technical deficiencies or issues in the patent that Modicon were able to exploit, leading to DEC updating the patent in the later version. In any case, Ricketts et al US patent (US3753243) was published in 1973 and other than minor alterations, closely resembles the earlier UK patent where the diagrams and much of the descriptive material are identical.

Ricketts et al (1972, 1973) described the "prior art" (background) to their "Programmable Machine Controller" by explaining what they saw as the two dominant control technologies associated with machine and sequential control: numerical control (NC) and combined relay & sensor control. They concluded that numerical control was expensive and that a cheaper way to achieve machine control was to employ bi-state sensors controlled by the program logic held within the interconnections of relay circuits. However, the simple relay and sensor combination was itself expensive because "Whenever the design of the finished part is [substantially] changed, this network is frequently discarded and an entirely new one wired into place" (Ricketts Jr. et al., 1973).

The objectives of Ricketts et al's solution, stated in the patents, were to:

1) "Provide an electrical machine control system that reduces this relatively high changeover cost"

2) Provide a highly reliable machine controller that was "characterized by relatively low maintenance costs".

3) Provide a system that allowed the modification and testing of "the logic arrangement prior to the actual wiring of the system".

(Ricketts Jr. et al., 1973)

Ricketts et al (1973) proposed a "controller for initiating and terminating the operations of a machine" by using a special-purpose "data processor" to read the status of connected bi-state sensors and set outputs (e.g. actuators) using a stored program. The processor compared the values of the sensors to the conditions and criteria contained in an "expression" held within the program and set an output signal accordingly. These expressions were known as "governing functions" and could be represented in the form of a Boolean logic statement. Effectively, these were functionally equivalent to the ladder notation but from a different notational tradition. The processor sequentially compared the sensor inputs to the governing functions "step by step" in order to produce the desired control effect.

The processor of the "Programmable Machine Controller" proposed by Ricketts et al (1973) was "continuously cycling through all governing functions", exploiting the cyclic program model previously proposed by Cox (1965) and Fletcher and Rosseau (1972). In order to reaffirm that the system was fast enough for machine control purposes in terms of processing speed, they stated that with "present-day equipment, this step-by-step comparison can be accomplished so fast that the time involved is insignificant insofar as machine tool operation is concerned" (Ricketts Jr. et al., 1973).

Ricketts et al (1973) comment further on the application of digital computer technology as a replacement for traditional relay circuits. The advantages, they remarked were:

> "...the system can be changed much more easily to accomplish a different set of machine operations... [because] one need merely modify the governing functions stored in the processor by altering the instructions" (Ricketts Jr. et al., 1973)

Also, a reduction in wiring requirements was achieved, saving costs, because there was "no need to alter the connections from the machine sensors".

The program for the new controller was to be developed on an external "conventional data processor" having a "magnetic core" read/write memory (Ricketts Jr. et al., 1972, Ricketts Jr. et al., 1973). This gave the advantage that a program could be developed and tested prior to loading to the controller's internal memory.

The controller program was held on a "wired memory", in modern terminology, Read Only Memory or 'ROM'. When Ricketts Jr. et al filed their patent in 1972, there was a gap in semi-conductor technology. The reason semi-conductor memory had not been used for programmable controllers at that time, was that there were no read-only memories available that were re-programmable. Non-volatile ROM itself was robust and the reason the controller itself did not use volatile read/write memory (e.g. RAM) directly was because the loss of program could have serious consequences (in terms of safety and cost) if control of the plant was lost.

When a program had been successfully developed, it was transferred from the external data processor to punched cards or punched tape which was used to "control a machine that [lays] down the wire in a wired memory" (Ricketts Jr. et al., 1973). The wired memory used a wire threaded through ferrite cores that stored the binary value permanently and was unalterable. Wired memory was used because it was cheaper and provided an "economical and reliable method of providing the memories for the controllers". If the governing function (fixed program) of the controller subsequently needed to be altered, the "entire memory must be replaced". But, according to Ricketts et al, it was still less expensive than the alternative to read/write core memory and replacement was facilitated by the use of "pluggable units".

Developing the programs using a "general-purpose data processor" provided flexibility "when required for trouble-shooting or program development and testing". However, the controller had the "low cost of a fixed-memory special purpose processor when operating in the internal mode to provide the machine-controlling function for which it is designed" (Ricketts Jr. et al., 1973). In some respects, similar to the concept proposed by Fletcher and Rosseau (a single separate programming tool for multiple controllers), one general-purpose data processor was used to program many controllers.

Ricketts et al. justified this by saying "even the cost of this device is relatively low when amortized among the individual controllers".

It was not just the fixed memory (ROM) that improved reliability and reduced costs. A further feature and advantage claimed by Ricketts et al (1973) was the "uncomplicated structure of the controller" (shown in Figure 5.3). "As compared with most digital computers, it has relatively few elements" and this "coupled with the small instruction set and the fixed memory" produced a low cost, highly reliable controller. The attribute of high reliability, they claimed, was of "particular importance in machine tool applications where lost time can be expensive".

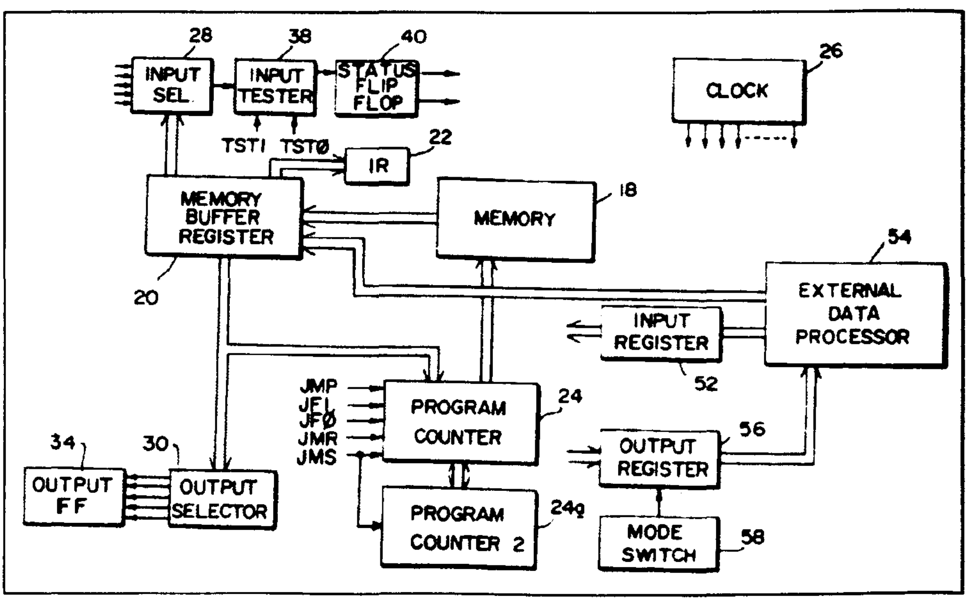


Figure 5.3 – Block diagram of the "Programmable Machine Controller"

5.2.3. Allen Bradley

The Programmable Matrix Controller

The "Programmable Matrix Controller" (PMC) filed in 1971 by the Allen-Bradley Company (Kiffmeyer, 1974), records the next significant step in the development of the PLC. The PMC differed from Fletcher and Rosseau's system in one significant way: it did not employ a general-purpose computer such as the DEC PDP8 in its construction. Also, it did not use a general-purpose computer (or data processor) to program the controller in the manner proposed by Ricketts et al. The PMC consisted of a smaller logic controller developed by Allen-Bradley using circuit arrangements to resolve logical constructs. Although the PMC can be regarded as a type of computer, emphasis was placed on interfacing with a large number of input and output devices rather than providing "extended computational capabilities" (Kiffmeyer and Baron, 1974).

The program was stored on a "matrix" that was held in Programmable Read Only Memory (PROM). The technology of ROM had developed and was now available in an alterable (programmable) form. The commercial adaptation of the matrix itself was based on fusible-link PROM chips, programmed or "burned" off-line. Each PROM was 256 bytes and a maximum of 6 PROMs could be addressed, therefore the largest possible program would consist of 1536 lines of instructions (Dummermuth, 2002). Having only four instruction codes, the PMC was capable of being programmed by non specialist "programmers" that used an external programming device very similar to the Modicon system described by Fletcher and Rosseau, again using the ladder diagram notation.

The relatively small size, low cost and high reliability of the PMC gave it an advantage over its contemporary computer-based alternative. Although limited in computational capability, it was suited to replace the unsophisticated relay-control panels used widely for sequential and combinatorial logic control applications.

Appendix D contains a detailed examination of the Programmable Matrix Controller patent.

Input and Output Expansion

Allen Bradley's "Programmable Controller Expansion Circuit" (Kiffmeyer and Baron, 1974) is a continuation of the Programmable Matrix Controller patent and was filed three months later in July 1971. Kiffmeyer and Baron expanded the "input-output capacity of programmable logic controllers" (the PMC) by modifying the existing address decoding circuitry. The "expansion circuit" used a gating circuit to activate alternative banks of input and output addresses.

Kiffmeyer and Baron's aim was to maximise the potential held within the limited word length employed in the Matrix Controller. The alternative, they presumed, was to increase the capacity of the memory matrix "so that increased word sizes can be accommodated and a larger number of attached input-output devices can be addressed" (Kiffmeyer and Baron, 1974). However, they claimed, this approach was "expensive, both in terms of the cost of the memory matrix, and the cost of the additional hardware needed to decode each instruction".

Kiffmeyer and Baron demonstrated that it was important to make every effort to keep costs to a minimum and to maximise the efficiency of the designs used for the programmable controllers. Also, although not explicitly mentioned within the patent, the desire to expand the number of addressable input and output devices connected to the controller was an important factor. This would enable larger and more complex systems to be controlled by this new technology.

A Definition of the Programmable Controller

Kiffmeyer and Baron's patent (1974) allowed clear distinction to be made between the programmable controller technology and that of the general-purpose computer. The purpose of this technology was to provide connections to external devices (sensors and actuators) so that 'real-time' control can be exerted on machines and processes. Computers on the other hand were designed to process data rather than external devices. They give a succinct definition of the programmable controller:

> "Programmable controllers accept input signals that indicate the condition of various input devices such as limit switches, push buttons, solenoids and photoelectric cells, compare these input conditions to the conditions specified in a stored program, and energize or deenergize output devices in accordance with the instructions in the program." (Kiffmeyer and Baron, 1974)

With regard to the application of the PMC and its inputs and outputs, Kiffmeyer and Baron state that:

> "The various input devices are attached to machine tools, or other industrial apparatus, and each device is connected to a specific input circuit in the controller. Likewise, the various output devices are attached to the machine too, or other controlled industrial apparatus, and each is connected for actuation by a specific output circuit in the controller."

With the introduction of specialised sequence controllers from Modicon, DEC and Allen Bradley, the new programmable devices were developing as a distinct technology.

5.2.4. Alternative Sequence Controllers

Using a simplified controller such as the Programmable Matrix Controller, independent of the standard general-purpose computer, proved to be a popular concept. Otsuka et al (1974) from the Tokyo Shibaura Electric Company, now known as Toshiba, filed their patent on 19th July 1972 for a "Sequence Controller". The patent was originally filed a year earlier on 22nd July 1971 in Japan under Priority number JP19710054159, a mere three months after Kiffmeyer's PMC patent was filed by Allen Bradley.

In accordance with previous patents, Otsuka et al (1974) explained that prior to their invention "Sequence controllers have heretofore been constituted by relays or contactless relays" (electronic logic circuits) that required complicated wiring and interconnection schemes. This is the reason, they concluded, that it was "accordingly impossible to mass-produce the sequence controllers". Moreover, "the prior-art devices have been extremely troublesome" because of the need to change the wiring "thereby rendering it difficult to modify the sequence." With the advances in electronics, they suggested, the sequences previously held within the wiring could be stored in "memories such as magnetic cores."

A diagram of the proposed invention described in Otsuka et al's patent is shown in Figure 5.4:

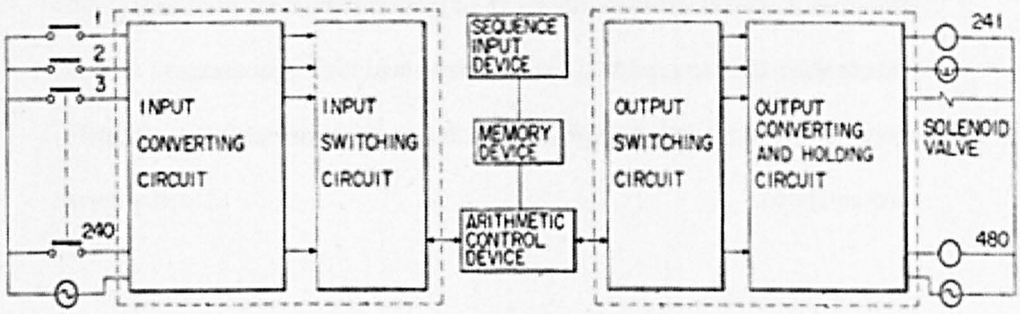


Figure 5.4 - Block diagram of the Sequence Controller (Otsuka et al., 1974)

Otsuka et al's (1974) system comprised a "sequence input device" to which a programming tool or console is connected and allows the instruction sequences to be entered "without specially programming it" into a "memory device". The instructions held in memory are passed sequentially to the "Arithmetic Control Device" which performs logical and arithmetic operations on the input and output information. The result is transmitted to the output circuit which in turn, controls the connected output devices (e.g. solenoid). The connection to the external input devices is carried out by an input interface (input converting circuit) which converts external signals into "logical values". The output unit accepts and holds output states from the output switching unit and provides control outputs.

The controller is programmed by a device, similar in concept, to Allen Bradley's programming tool for the Programmable Matrix Controller. An external console is connected to the "Sequence Input Device" containing elements representing logic functions represented as push-buttons on a keyboard. However, Otsuka et al's invention differs from Allen Bradley "Matrix" programming tool in that the sequence to be executed by the controller is expressed and represented as "contactless diagram symbols" (Otsuka et al., 1974). Instead of the ladder diagram notation, logic gate symbols were used to represent the combinational logic sequences. Figure 5.5 below shows equivalent "contactless diagram" and relay circuit representations.

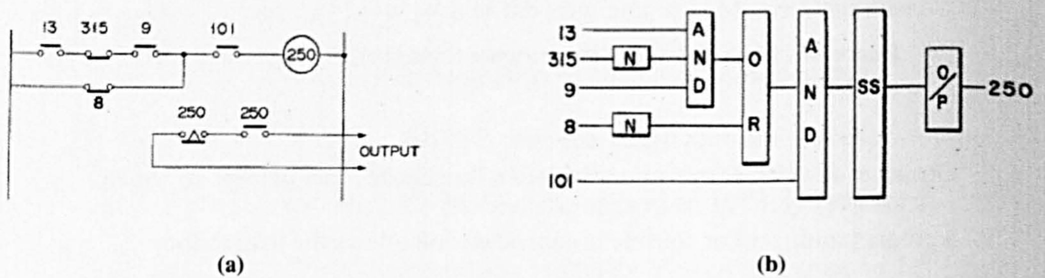


Figure 5.5 - Diagram (a) traditional relay circuit and Diagram (b) depicting the equivalent "contactless diagram" (Otsuka et al., 1974)

It was possible to program the Sequence Controller with six different logic instructions with a maximum of six instructions per sequence. This was deemed adequate by Otsuka et al (1974) who remarked that "Even a very large sequence can be expressed by six operating instructions". Each step in the sequence could be regarded as the equivalent to a rung on a ladder diagram and the maximum number of sequences that could be executed depended on the "capacity of the memory device". In line with its contemporary controllers of the early 1970s, the sequences held in memory were continuously cycled, approximately every 20ms.

Otsuka et al (1974) explained that the advantage of expressing the sequence by the "contactless diagram symbols" was the ability to "simply and clearly set, modify and check the sequence". Further, "a sequence can accordingly be attained by mere manipulation of push buttons without any particular knowledge of programming". A key objective was to enable the practising engineer to have the means to program a digital controller without requiring specific programming skills, an objective similarly expressed by Modicon (Fletcher and Rosseau, 1972) and Allen Bradley (Kiffmeyer, 1974) in their earlier patents.

Koyanagi et al (1976) draws attention to the fact that by 1973, the requirements for sequence control were becoming more sophisticated and complex. The "pre-controller technology" (electro-mechanical and relay-based control systems) were not up to the task because of reliability issues and the inconvenience when the inevitable modifications were required. Sequence control, performed by programmable controllers, had become an established technology and used widely throughout industry. Koyanagi et al were reinforcing the requirements for applying adapted computer control to solving sequential logic problems in line with other programmable controller developments. The "Digital Logical Sequence Controller" proposed by Koyanagi et al is shown in Figure 5.6 below.

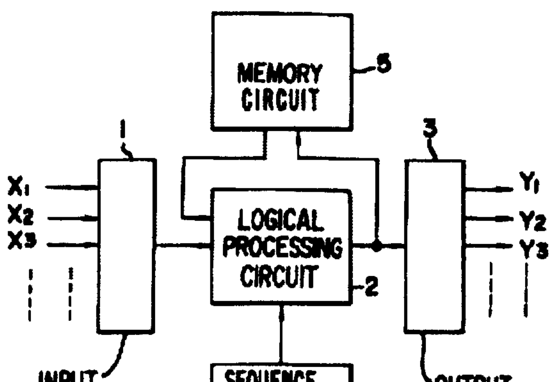


Figure 5.6 - Sequence Program Storage Circuit (Koyanagi et al., 1976)

The sequence controller shown in Figure 5.6 was remarkably similar to the classic programmable controller design using a logic processor (2) to process inputs (1) and outputs (3). The sequence program or "pattern of sequence control operation" was stored in a "Sequence Program Storage Circuit" (4) which was a core memory. Intermediate logic results were stored in the memory circuit (5). The sequence program was entered via a keyboard.

5.3. Programmable Controller Developments

Improved Diagnostics

By the early 1970s, the new sequence controllers had established themselves as a viable alternative to the larger and more cumbersome relay control panels. However there were still practical problems to be overcome, for instance, input fault detection. Chace et al provide a useful example of the impact and effects of problematic inputs with machine tool control they observed in 1971:

> "...[A] majority of machine tool malfunctions still exist, largely due to limit switch and controllable device failures. These malfunctions may be complete and cause complete machine tool shutdown. They may also be intermittent" (Chace et al., 1972).

The diagnosis of these "malfunctions is extremely difficult" in comparison to the "more reliable diagnostic procedures for relay-control systems" where "an experienced operator listens to relay clatter for changes in the relay noise pattern" (Chace et al., 1972). Although automatic diagnostic methods did exist with data processors (computers), the problem was adapting them to asynchronous sequence controllers. Clearly, a new approach was required.

The solution proposed by Chace et al. was to employ a separate "Analyzer for Sequencer Controller". The "analyzer" employed a "transition monitor" device, a data processor working under the supervision of the sequence controller to synchronise the two systems. The transition monitor recorded the input transition sequence and compared the result with the pre-existing expected result stored in a look-up table. Malfunctions were detected if any discrepancies between the actual transition sequence and the predicted transition sequence existed. This approach is similar to the idea previously proposed by Lawson et al. (1970).

A further enhancement to aid the maintenance and diagnostics of sequential control systems was filed in 1973 by William Seipp (1974). Seipp proposed the idea of providing "indicating circuits for indicating the condition of the several input circuits and output circuits of the controller". The indicating circuits included Light Emitting Diodes (LEDs) "controlled by the logic seen by or directed from the controller". Although, this was a relatively simple idea, it has become a standard feature on modern PLCs and as Seipp concluded, "Maintenance of the programmable controller is quite easily performed, since the source of the difficulty can be more quickly located".

Programming Tool and Memory Developments

Although one of the most advantageous features of programmable controllers for the engineer was the "simplicity of their language", an important objective conveyed by Struger and Radtke (1974) was to...

“simplify and speed up the process of transferring solutions to control problems, as expressed by the control engineer, into the controller”.

Allen Bradley continued to develop the Programmable Matrix Controller (PMC) ‘system’ by making further improvements to the programming device. Struger and Radtke (1974) described a “Controller Programmer”, a development of the programming tool for the earlier PMC. Their proposal was an improved means to load a program onto the PMC using a programmer with read/write memory. This new feature ran the control program from the programmer memory rather than the read-only memory of the controller. The benefit was that program corrections could be made directly on the installed controller at the point of use. This was necessary according to Struger and Radtke because “in practice, changes in the program are often needed after the controller is operated on the job site in order to overcome unforeseen problems or programming errors”.

When a program was tested and deemed to be correct, the modified program held on the programmer was transferred to an external read only memory that was “suitable for installation in the controller”. Note that although the controller could run from the external programmer’s read/write memory, it was not possible to directly alter the installed read-only memory on the controller itself. Replacement memory was programmed externally by the programmer and then physically installed on the controller. In terms of function, this concept was very similar to the “general-purpose data processor” suggested by Ricketts Jr. et al (1973).

To ensure that the new program was successfully conveyed to the read-only memory the "transfer is made in synchronism with a ten-phase clock and a check is made by a comparator to verify the proper loading of each instruction in the read-only memory" (Struger and Radtke, 1974).

Struger and Radtke (1974) explained that read-only memories were used on the PMC controllers because they were more reliable and less expensive than read/write memories in 1971. A "plated wire" memory was used as the read/write memory in the embodiment of their "Controller Programmer" and although this type of memory was expensive, the program could be "easily loaded, changed and erased". However, as with other programmer units, the "programmer circuit is detached from the controller and its read-write memory is erased to make it available for use with other controllers", amortising the costs if used for multiple controllers.

A further and significant feature of the Controller Programmer was claimed by Struger and Radtle. "[The] input unit of the programmer circuit not only includes push buttons and thumbwheels for manually loading the read-write memory in the programmer circuit, but it is also adapted to load the read-write memory in response to data on punched paper tape". The thumbwheels and push-buttons were not only aimed at the practising control engineer in the field, but also so that programs could be developed by the engineer offline, and then automatically produce the punched tape by means of a "digital computer which is programmed to translate logic diagrams, or Boolean equations directly into controller language".

5.4. The Programmable Logic Controller

Allen-Bradley continued its development of programmable controller technology resulting in patent US3942158 (Dummermuth, 1976). In this patent, Dummermuth first uses the device description "Programmable Logic Controller" (although the term was also used in Kiffmeyer's patent to describe the function of the PMC). According to Ball (2008), Allen Bradley registered the term "Programmable Logic Controller", presumably relating to Dummermuth's patent submission, however Matsushita, now Panasonic, filed their patent titled "Programmable Logic Controller" (Hamano, 1974) on 17th March 1971, predating Dummermuth's PLC patent by over two years (in terms of priority (filing) dates). The Matsushita PLC was a "programmable logic controller which performs sequential operations in accordance with the programmed instructions by scanning the input, comparing the input with the conditions specified in the program, and finally by energizing or deenergizing the outputs" (Hamano, 1974).

Dummermuth's PLC differed from both Hamano's PLC and Kiffmeyer's PMC in two principle ways: It used a memory buffer to store the input and output (I/O) values; and the executive program and I/O scan rates were decoupled from one another. The Dummermuth (1976) patent is particularly important and was a key technological development in the history of the PLC; it was cited in 61 other PLC related patents (Espacenet, 2012b).

Background

The historical background in Dummermuth's (1976) patent stated that programmable controllers were typically connected to industrial equipment such as assembly lines and machine tools to "sequentially operate the system in accordance with a stored control program". The field of operation (at the time of filing) was limited to digital systems, and there was no mention of analogue-based process control applications. Dummermuth explained that in prior systems like the PMC, program instructions are stored in memory and executed "in rapid sequence" to examine the status of inputs and to "energize or deenergize" selected output devices. The output devices are "contingent upon the status of one or more of the examined input devices" according to the control program instructions.

The sequentially operated programs of a controller like the PMC referred to by Dummermuth, were executed one line at a time in a continuous loop. According to Dummermuth (1976), the method of processing the control program line-by line has limitations. The response time of reacting to changing input conditions is directly related to the time it takes to scan the entire control program. The program loops from start to finish sequentially, so the greater the number of lines of program, the longer the time taken to re-run the same program instruction and this effect slows the response of the controller. This in turn limits or constrains the complexity of the system because there will be a point at which the response time will be too slow for the equipment connected to the programmable controller.

The objectives of the Dummermuth's patent were to provide a programmable controller that overcame the limitations of long control programs, slow program execution rates and to improve the noise immunity of its inputs.

The noise immunising circuitry Dummermuth referred to was required to smooth out and remove unwanted electrical signals such as switch bounce and interference[9]. The unfortunate consequence of using such noise immunity circuits was the introduction of a propagation time delay caused by the input and immunity circuits. This in turn restricted the maximum scan rate in order to accurately detect the presence of an input signal.

Objectives of the PLC

Dummermuth's (1976) first objective was to "optimize the rate at which the input and output circuits on a programmable controller are scanned". The problem he identified was the switching speed of the input circuits. The rate at which input information is received by the processor, processed and generates the output is "limited by propagation times and the time delays which are associated with the noise immunity circuitry" and in "prior programmable controllers a compromise is made between these conflicting objectives with the result that the program instruction execution rate is reduced and the noise immunizing circuits are kept to a minimum".

Dummermuth's solution was to use read/write memory buffers (RAM) to store the values of the inputs and outputs (I/O) and make the buffers accessible to the processor rather than the I/O directly. Dummermuth refers to these buffers as the "input image" and "output image".

[9] Noise immunity circuits are designed to couple the unwanted signals (generally high frequency signals) to a common potential (e.g. earth). A simple example could be a low-pass filter consisting of a capacitor and resistor network on the electrical input to the controller.

The advantage of using the buffered I/O approach is that the I/O scan rate can be decoupled from the scan rate of the processor. According to Dummermuth (1976) "The rate at which the scanner circuit interrupts the processor to couple status data is substantially less than the rate at which the control program is executed under the direction of the processor". If the I/O scan rate is decoupled from the controller scan rate then "...the input/output speed of the programmable controller can be optimized to meet the requirements of the input and output interface circuitry." This enabled the input circuits to be optimised without compromising noise immunity.

A second objective was to improve (increase) the processing speed of the executive program. A further advantage of using buffered I/O meant that the logic processing cycle running the "executive" program was also decoupled from the I/O serving cycle. This enabled the execution times of the processor to be speeded up because it did not have to read in each input/output. The execution program had permanent and direct access to the input and output image. According to Dummermuth:

> "High speed logic circuits are used in the processor and random access memory and these are suitably shielded from external noise sources. State of the art data processing speeds can thus be accomplished to execute the control program and manipulate the status data in the input and output image tables." (Dummermuth, 1976)

A third objective for Dummermuth's PLC was to "provide the means for interfacing a programmable controller with external systems... other circuits may be connected to interrupt the processor and couple data between an external system and the random access memory" (Dummermuth, 1976). The objective was to provide communications connectivity to external devices such as external computers, programming devices, printers and displays. The new PLC was introduced in 1974 and the Dummermuth patent forms the basis of the archetypal modern PLC.

Dummermuth's final objective provided an insight into future developments of the PLC. The invention was to:

> "...provide means for interfacing a programmable controller with external systems. In addition to the scanner circuit, other circuits may be connected to interrupt the processor and couple data between an external system and the rando[m] access memory. Such systems may include, for example, a computer, an arithmetic unit, or a control program loader..." (Dummermuth, 1976).

The means to connect to external digital systems such as computers was to become an important feature of PLC technology.

5.5. Programming Methods

5.5.1. Ladder Diagrams

The ability to program the controllers from directly translating a ladder diagram to program the controller was a popular concept. Nakao et al (1974) filed a patent to provide "A general purpose sequence controller wherein a schematic electric circuit diagram comprising a ladder network of circuit lines disposed between two vertical bus lines is changeable and simulated by a special purpose control program...". The patent itself was originally filed in Japan on 31 July 1972. A continuation of this patent (Nakao et al., 1977) was filed on 13 June 1974 in the United States (23 July 1973 in Japan) with additional claims but the essence of the invention remained unchanged.

One objective of the invention was to "provide a new and improved unique general purpose sequence controller wherein a schematic electric circuit diagram can be easily simulated by a simple special purpose control program". The idea was comparable with the recent developments in programmable controllers to enable a control engineer to program the controller directly from a standard electrical ladder diagram without having prior computer programming knowledge.

A second objective was to provide a "general sequence controller having a logic operation circuit capable of continuously examining an external input signal individually with logical 'AND' and 'OR' functions". This statement demonstrated that the ladder notation was fairly simplistic in terms of functionality. The proposed controller used "examine commands" to read in external input signals and applied "logical 'AND' and 'OR' functions" combined with a secondary memory to store the intermediate results. Although only 'AND' and 'OR' logic could be represented in the program, the controller could resolve multiple inputs.

Nakao et al. (1974) described the prior art in terms of contemporary programmable sequence controllers and their limitations, thereby emphasising the advantages of their invention. In "conventional programmable sequence controllers" a logic operation on an input condition was performed by a "logic operation circuit". The circuit was provided with only one memory element to memorise the result of the operation. In order to process multiple inputs "a series of examine commands had to be provided and unified in the form of logical AND functions for continuously examining the input conditions". The disadvantage of this approach, they claimed, was developing the "executive program" to implement the sequence control - "troublesome logic operations were required and a [skilful] programmer is needed for developing the programming operations". Additionally, because the programming became complicated it "decreased the effective utilization capacity of the memory device" and "then the executive program was itself made long".

5.5.2. Alternatives to Ladder – Boolean Algebra

In section 5.2.4, Otsuka et al. (1974), on behalf of Tokyo Shibaura Electric Co (now Toshiba) had proposed the use of "contactless diagram symbols" (logic gates) as a logic representation for programming controllers. In 1973 another Japanese company, Mitsubishi, also proposed an alternative to the ladder diagram programming notation, a "sequence controller with computer-like control functions adapted to sequence controls" (Koyanagi et al., 1976).

Sequence controls, Koyanagi et al (1976) explained, were "indispensable in the present-day industrialized society" and being generally relay-based, this "sequence control technique is widely used in the field of industrial process control". The typical industrial applications employing sequential control referred to in the patent were for "power plant and substation control, conveyor system control, machine tool control, assembly line control in the automotive plant, and rolling line control". The drawback of this pre-controller technology was that it was "inconvenient", and a "degradation of reliability" was experienced when these applications required modification.

They further noted that "the controlled objectives have become much more sophisticated which has necessitated the use of an increasing number of relays, with the result that the logical design has become intricate and the approach has become more difficult."

Koyanagi et al's proposed solution operated "in general on flow-chart system or Boolean algebraic system (conversion system) through programming". The relay sequence program was entered via a keyboard "expressed in terms of Boolean algebra" and stored in the core memory of the controller. This "Boolean algebraic system offers high processing efficiency" claimed Koyanagi et al (1976).

A Boolean algebraic programming representation was also proposed by Allen R Holecek (1976) for the Babcock and Wilcox Company. The "application of data or information processing techniques to control the operation of specialized machines [such as machine tools]... has a well established industrial position". Control instructions to "one of these "automated" tools" are generally programmed by a "machinist in a "program language" of which, for instance, the "Fortran" language is perhaps best known." The specialised machine's electrical system, translates the Fortran instructions into a "machine language".

The disadvantages were that "language translation circuits" were needed and two sets of instructions had to be stored in the memory of the machine: the program language (e.g. Fortran) that the machinist understands; and the "machine language to which the tool responds". This was not only inefficient in terms of memory requirements, but program mistakes, were often "very difficult to identify and correct" and may "exceed the reasonably anticipated programming abilities of the average machinist" leading to "frequent expensive work stoppages" claimed Holecek (1976).

Holecek's (1976) solution was to provide a control system for a "machine tool" that allowed the operator or "machinist" to generate a program in the form of a Boolean expression. This was achieved by "instructions that are coded in a numerical form and entered into the system in the form of Boolean algebraic expressions". The expressions correspond to a machine tool "status enquiry", reflecting a specific machine function and its binary condition ('ON' or 'OFF'; 'YES' or 'NO').

The individual terms of the expression are related to each other "by means of conventional mathematical symbols, e.g. addition sign (+), parentheses, and the like" but, to avoid confusion, Holecek was keen to point out that "These symbols, however, may not have the same effect that they have in conventional arithmetic processes. Thus, the addition sign usually means the functional equivalent of the word "or" rather than the usual "added to" meaning" (Holecek, 1976). Thus the control system used the standard Boolean algebraic constructs '.' and '+' to represent 'AND' and 'OR' respectively.

The embodiment of Holecek's patent described a control system that was applied to a machine tool. Sequential control was achieved by using basic logic constructs, similar in function to the existing relay logic, and as Holecek stated, their invention could be applied to any "specialized machine".

Further Boolean Developments

In Germany, Siemens also proposed an alternative to ladder diagrams, preferring Boolean constructs. The patent was first published in German (Schmidt et al., 1973) and later in English (Schmidt et al., 1975). Schmidt et al. related logic circuits "for performing Boolean logic operations" used in a number of industrial control applications. As with Koyanagi et al and Holecek, Schmidt et al's device performed simple logic equations using 'AND' and 'OR' Boolean equations, represented in the program software. Prior to their invention, Schmidt et al claimed that this was achieved either by software using a digital computer or in a "hard wired form" (using relays or electronic logic circuits) and "whether it is done in software or hardware, the amount of hardware required rises as the quantities of signals to be processed increases".

Schmidt et al's objective was to provide a control system that could process an "arbitrarily large number of logic combinations with a defined minimum number of building elements" (Schmidt et al., 1975). Their control system was to provide the means to solve large and complex Boolean equations quickly within the limitations of available memory and a reducing the complexity. For a computer using many inputs and outputs, the "expenditure for obtaining a function of the variables associated with the program is the most essential aspect when establishing the technique of the computer"... "In known universal computers" the variables associated with a control function are "obtained successively in a predetermined sequence" which requires a considerable amount of time and a large number of control commands, which make it an "unfavourable" technique (Schmidt et al., 1975).

The large computer memory requirements Schmidt et al noted, were due to excessively long "jump spans" when many variables are used. 'Jumps' are the requirement to jump or skip a number of lines of program code. A jump is similar to an 'IF, THEN, ELSE' statement in a modern high-level programming language. It relates to a test on a particular condition (e.g. Start Button depressed):

> **IF** "Start Button depressed" = **'TRUE'**
>
> **THEN** "Motor Output" = **'ON'**.
>
> **ELSE 'JUMP TO'** *(location of next test or function in memory).*

The 'Jump Span' is the maximum number of lines of the program can jump (or skip).

Schmidt et al (1975) noticed that large and complex programs sometimes required long jump spans and in order to accommodate this requirement, larger memory sizes were necessary. Their solution to this problem was to provide a "Flow Chart" method of programming functions that used a "sequence of controlled jumps having limited jump spans" (see Figure 5.7). The "requirement of storage space is small when the jump span is relatively staged" and was based on the idea that if shorter "jump spans" are used, the memory requirement is lower because fewer instructions need to reside in the fast access memory (RAM).

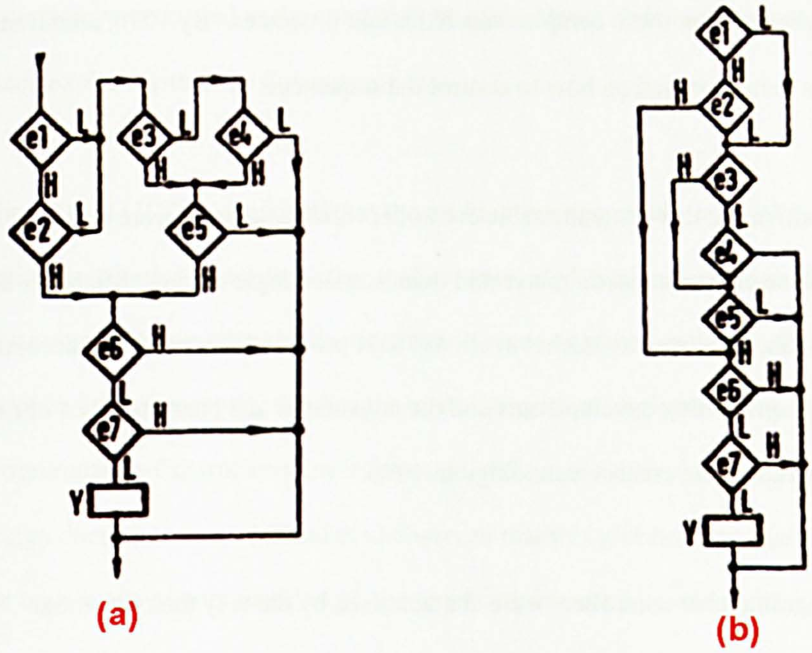


Figure 5.7 - Flowchart representation of a program sequence (a) rearranged to illustrate the length of the jump instruction (b). (Schmidt et al., 1975)

To implement the "flow chart" solution the jump spans are restricted to a maximum of six. Schmidt, et al justified this because "A statistical evaluation of adjusting controls and positional controls of machine tools and the automatic control electronics, has shown that with jump spans of up to 6, more than 95% of all conditional jumps during the obtaining of a function may be carried out" (Schmidt et al., 1975). In the 5% of cases where a large or "special jump" is still required, Schmidt et al explained that "the immediately following storage location is accessed for reading the address of the storage location to which the jump has to be made".

5.5.3. Sequence Programming Techniques

The ability to represent multiple relay circuits within the controller's program and memory meant that programmable controllers were now being applied to ever more complex machines and processes. By 1976, attention was being focused on how to control the sequences.

Reaffirming that programmable controllers and sequencers were designed to replace electromagnetic relays and transistorised logic circuits, Michel et al (1978b, translated as Michel et al., 1978a) , provided a summarised account of programming developments and the advantages and characteristics of the programmable control technology in 1976.

Programmable controllers were characterised by the way their electrical connections were envisaged, for example ladder logic and ladder diagrams, and one advantage was that the technology could be understood and used by technicians. A second advantage was the flexibility of the system; the technology was adaptable to suit and closely fit the particular task (machine or process) they were applied to (Michel et al., 1978a).

The disadvantage of programmable controller technology, according to Michel et al (1978a), was that the circuits were fixed (for example limited to 4-element rungs on a ladder diagram) and the existing technology had difficulty in performing the more complex functions of counting and comparisons. Further, remarked Michel et al, the engineering practices associated with the technology were flawed. Initially, for a project or installation involving the use and application of a programmable controller, a technical specification was developed. From the specification, the sequential logic was designed and finally, the system was built. However, the sequential aspect of the program itself was not clearly documented; the sequences were buried in the complexities of the direct code and were not made explicit within the documentation.

Michel et al (1978a) explained that the problem was generally realised during installation or when further maintenance or alteration was required. The assumption was that the documentation was completed at the design stage and that frequently, the designs were changed on implementation. Programmable Control systems it appeared, did not aid the upkeep of the design documentation. Michel et al observed that this problem was due to the high levels of complexity with the logic diagrams and because of this, maintenance technicians were less inclined or did not have the ability to keep the documentation updated. According to Michel et al, the technician, when carrying out maintenance tasks, looked at the relevant sections of the code only and did not relate the individual logic runs to the overall sequence.

Michel et al's solution was that sequences should be simply represented and facilitated by the use of mathematical tools or representations that were currently known. Graphical representations such as Petri nets, step diagrams and other flow charts techniques were suggested to symbolise the sequences (Michel et al., 1978a). Further developments in representing sequences resulted in the use of GRAFCET[10] leading to the present day embodiment in PLCs as Sequential Function Charts (SFC) which are used to represent and indeed program large or complex sequences in a standard format according to IEC 61131 (David, 1995).

[10] GRAFCET is a sequential control modelling language based on Petri-nets introduced in 1977 in France.

Drum Sequencers

Michel et al remarked that programming sequences was complicated using the limited ladder diagram notation; the more complex the program, the more difficult it was to follow the sequence. PLC developers began to look at alternative techniques of programming sequences. One surprising solution was presented by Morley et al (1980) of Modicon, who in 1977 developed a "Programmable sequence controller with drum emulation and improved power-down power-up circuitry". Morley et al had drawn on the ideas of an earlier technology for controlling sequences and implanted them in the "digital computer" of the PLC. As the descriptive title stated, the objective was to "emulate" the functions of an electro-mechanical drum sequencer.

The drum emulation was achieved as follows: "The controller emulates mechanical sequence drums so that at any one time each of the simulated drums within the controller executes one of the addressable drum lines programmed within the drum" (Morley et al., 1980). Individual sequences were represented on a "simulated drum" and called in turn to output the sequence. Each simulated drum could be programmed to energise any output or "any memory bit utilized by the controller in order to provide communication between drums". The concept of using internal memory bits meant that simulated drums could communicate with and control other simulated drums so in effect, the complete sequence of a machine could be represented by cascaded simulated drums.

Morley et al (1980) also explained how they implemented jump instructions to the programmed drum sequence. "Each drum was also given "one or two sets of exit conditions" which caused the drum to rotate to a specific "drum line". Older electro-mechanical drums were fixed in sequence because the drums were permanently rotated and the sequence could not be skipped. Morley et al's technique overcame this limitation. The ability to exit a drum sequence enabled the controller to react to major events and emergency situations – "The controller can also sense emergency conditions and cause any or all of the drums to rotate to a specified line regardless of the drum line then being executed by the controller for each of the drums" (Morley et al., 1980). The advantage was that the controller could manage an "ordered and complete shutdown of the controller" if any critical conditions existed, including a power failure.

A key additional feature Morley et al proposed for handling a power outage was the implementation of a non-volatile sequence memory. The controller "maintains memory validity for all types of shutdown situations" including "momentary losses of any supply voltage". This was achieved by using a "clock-calendar" that was capable of "continued operation during periods of extended power outages". The "clock-calendar" was not just reserved for handling the effects of power failures and the continuation of the sequence on power restoration, it was integral for controlling time and date dependent sequences. The "clock-calendar can be utilized in any drum line to form part of the control scheme" (Morley et al., 1980).

In keeping with the programming devices previously developed by Modicon, programming the sequence controller drum lines was achieved by an external "data communication device". The device utilised a "simple user-oriented language, with monitoring and diagnostic capability to facilitate debugging" (Morley et al., 1980). Modicon were clearly targeting the end user of their sequence controller system, the practising control engineer.

Alternative Drum Sequencer Techniques

Satoshi Yano (1986) for Omron Tateisi Electronics Co, also proposed a drum sequencer to effect sequential control, filing the patent in Japan in 1982. The principle approach described in the patent uses a different method to that of Morley's patent (filed in 1977), a counter instead of a timer.

Yano (1986) described in some detail the background and reasons for using an emulated drum sequencer. Before programmable controllers were developed, simple, mechanical sequence controllers known as "drum type sequencer and a rotary cam type sequencer" had been widely used. Effectively, "rotary cam" sequence control was achieved by a mounting cams fixed to a rotating shaft that operated micro switches, or in the case of a drum sequencer, a metal projection representing the same function as a cam. These types of sequence controllers had the advantage that "their construction is simple and the sequence can be changed readily". However, stated Yano, their parts could mechanically wear and, more significantly, they could only be applied to "sequence control which is extremely small in scale and simple", and were rarely used (at the time of filing, 1982).

Programmable controllers on the other hand, the most general being the "relay ladder type", served large-scale, complicated applications with "high-degree control functions" (Yano, 1986). They provided "small-scale and simple sequence controls... [expressed] in the form of a circuit diagram using a relay to provide programming". "Generally, a shift register command or stepping switch command is used or counter command and data comparison command are combined to program the same sequence control as that handled by the drum type sequencer".

According to Yano (1986), programming sequential control using a conventional programmable controller was not simple and an extremely lengthy program has to be set up which was cumbersome, prone to errors and very difficult to read. Consequently "the characteristic of the drum type sequencer, wherein the sequence change may be accomplished readily has not been introduced into conventional programmable controllers". Presumably Yano was not familiar with the work conducted by Modicon and Morley et al.

Yano's solution was to provide a "[software] function of a drum type sequencer can be made subject to programming" based on using a counter or "drum counter". The drum counter could be stepped one position at a time or "inched" as per a mechanical drum counter and the position of the "cam" corresponded to "data in the numeric range" held on the counter. "In this manner, programming can be accomplished by the almost same simple logic as that of the drum type sequencer" explained Yano (1986). To summarise, Yano used a cyclic counter that held a number of numerical values at each position. The "counter" (or register) could be stepped to one position at a time (rather like a stack pointer in a computer). The "numerical" value, when converted to single bits, held in each position of the counter, corresponded to the switch values of a normal cam or drum sequencer.

5.5.4. Ladder diagrams and the introduction of the microprocessor

Programming or converting a ladder diagram into a program input for the controllers were achieved in a number of different ways. For example, Fletcher and Rosseau's solution was to select each rung of the ladder by line number using a thumbwheel. The rung was restricted to four elements, each selected by a position number and element type entered by push-buttons (Fletcher and Rosseau, 1972). A similar method was used by Kiffmeyer (1974) for Allen Bradley's Programmable Matrix Controller. The programming format was mechanically fixed rather than the flexible text-based software code used in modem programming languages. The important thing at the time (early 1970s), was to give the engineer the ability to translate the electrical ladder diagram into an input sequence entered directly onto the controller via the programming device.

Improvements and developments with electronic devices such as integrated circuits and microprocessors both decreased the physical size and increased the functionality and sophistication of programmable controllers. However, the historical reasons to engage the control engineer by means of familiarity with the ladder diagram notation persisted. Struthers-Dunn Inc filed a patent in 1977 with the rather lengthy but descriptive title "Process Control System that Controls its Outputs According to the Results of Successive Analysis of the Vertical Input Columns of a Hypothetical Ladder Diagram" (Henry et al., 1980).

Henry et al proposed the use of a microprocessor to resolve a program entered in the ladder notation. The benefits, they claimed, when compared to the available programmable controller technologies in 1977 that did not use a microprocessor, were: lower in cost; more reliable; simpler in construction and "simpler to put into use by persons unskilled in the technology involved".

Memory Improvements

Memory technology was still developing and expensive. However the decreasing cost of semi-conductors, and in particular the developments in integrated circuits, helped to circumvent this problem.

Kintner (1980) provided improved and enhanced integrated circuit based memory that automatically stored the on/off status of all inputs, outputs and the "wire node" (intermediate logic results). The ability to store and make accessible this data to the processor simplified programming in that it made possible "a particularly simple unidirectional-logic programming mode because the programmer does not have to keep track of which logic operations must be temporarily stored".

For the maintenance and control engineer, Kintner (1980) noted that the benefits were two-fold: 1) data pertaining to logic results stored in memory could be directly monitored to display status aiding fault-finding; and 2) individual "wire nodes" could be manually forced "on" or "off" for maintenance purposes or the like" (Kintner, 1980).

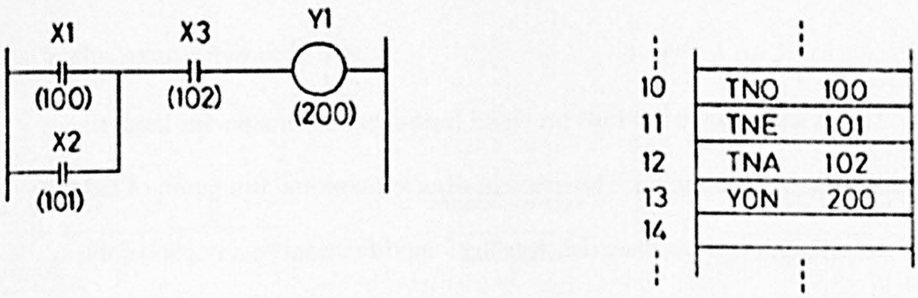
"Microprograms"

Struger and Dummermuth (1981) from Allen Bradley proposed a microprocessor based programmable controller that was compatible with prior controller programming devices. The microprocessor-based system system introduced the ability to provide improved timers of a higher resolution (10ms) previously not available with the solid-state programmable controllers. The microprocessor was adapted to the programmable controller by being "microprogrammed" where the "microprocessor is converted to programmable controller instructions by using "macroinstructions". Struger and Dummermuth provided an insight into the efforts required to adapt a microprocessor to the programmable controller.

Further developments of employing microprocessors in programmable controllers were revealed by Allen Bradley's "Dual language programmable controller" (Struger et al., 1981). Prior to Struger et al's patent, programmable controllers employing microprocessors, converted ladder diagram instructions to machine code instructions for execution that were not accessible to the control engineer (user). The objective of this patent was to enable the programmable controller to run both "control instructions" (ladder logic) and "machine language routines". Machine language routines were the machine code pertaining to the microprocessor instruction list. Struger et al's development was to make this instruction list accessible so that programs could be written in machine language and ladder logic within the confines of one control program.

Struger et al. (1981) explained that the control engineer or "user" was thus able to write programs "outside the scope of the fixed programmable controller instruction set" in order to carry out additional functions not available in the ladder logic notation. Machine code programs were executed in accordance with the control (ladder) program and both were stored in the main memory.

Similar in concept to Struger et al's Dual language programmable controller, Yomogida et al (1985) proposed the idea of storing a "microprogram" in memory in order to carry out complex functions, that were either not possible or difficult using the ladder program representation.



Command Word	Operation Code	Meaning
TNA	0001	Test whether a designated I/O element is ON and obtain a test result ANDed with a previous test result.
TNO	0010	Test whether a designated I/O element is ON and obtain a test result ORed with a previous test result.
TNE	0011	Test whether a designated I/O element is ON, obtain a test result ORed with a previous test result, and incorporate the test result into a previous AND test result.
YON	0111	Energize a designated output element if the test flag indicates satisfaction, or de-energize the designated output element if the test flag indicates non-satisfaction.

Figure 5.8 – Ladder sequence with microprogram equivalent (Yomogida et al., 1985)

Figure 5.8 shows a simple ladder diagram with the microprogram equivalent instructions. Also, note the addressing for each variable is numeric representing the input and output addresses.

5.5.5. Ladder Programming Improvements

Developments in programming were not just centred in opening up the enhanced instruction set of the microprocessor to the PLC; the ladder notation itself was being developed. Toshiba Machine Co Ltd submitted a patent proposing a method of expanding the number of columns in a ladder logic program (Jiyunichi and Yoshihiko, 1982). Toshiba's patent demonstrates the continuing relevance of ladder diagram notation to Japanese manufacturers of programmable controllers.

Taylor and Vaniglia (1984) provided further enhancements for ladder diagram programming. The inherent disadvantage and limitation of ladder programming, they observed, was that "modifications to any part of the program require that all processing be inhibited, and consequently that the operation of the controlled mechanism temporarily be suspended". In the manufacturing environment, this was a common requirement and the shutdown of a production line was very undesirable.

Taylor and Vaniglia proposed a system whereby the ladder program could be modified on-line without interrupting the execution of the main control program. Arguments as to whether this is a desirable feature are beyond the scope of this thesis, however, this is an endemic feature still found in modern PLCs. Additionally, Taylor and Vaniglia proposed a further objective that related "variable names corresponding to control devices and variable values". Many PLC systems of the 1970s and early 80s referred to device addresses rather than variable names. This feature aided the readability of the ladder program to the programmer.

Programming Sequences

Hitachi Ltd filed their "Sequence Controller" patent (Takaki, 1984) on 7th May 1983, later published in the UK (Takaki, 1986) followed by the US as "Sequence control method and apparatus" (Takaki, 1987). Takaki proposed a method and system for programming sequences that differed from the continually cycling program of prior programmable controllers, where the sequence was embedded within one main program. Takaki's solution was to represent the process sequence as "stage programs" that were called when particular criteria had been met.

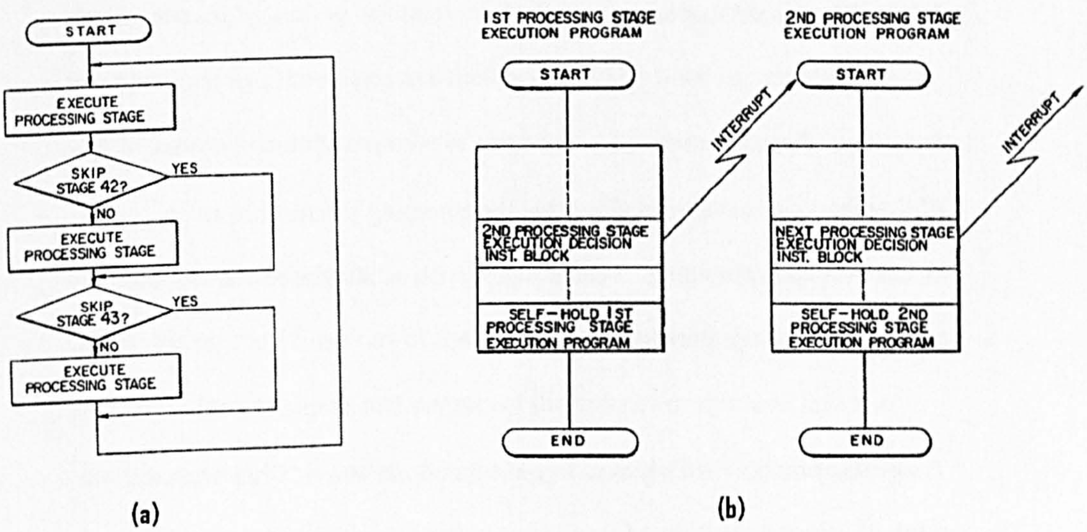


Figure 5.9 – Flow Chart and Stage Program (Takaki, 1987)

The flow diagram depicted in Figure 5.9(a) demonstrates how Takaki interpreted how programmable sequence controllers executed sequence programs. All program instructions are held within the one cycled program and sequence steps in the program are represented by jump or "skip" instructions. Figure 5.9(b) shows each sequence as a separate block of programming code. A "main program" is cycled in the conventional manner and controls the execution of the individual sequence blocks.

Among several advantages claimed by Takaki (1986), sequences were "easy to grasp" by the programmer and allowed easier modification and improved fault-finding. The "stage program" method of representing sequence processing was an early forerunner of the "Sequential Function Chart" programming technique adopted by the BS EN 61131 Programmable Controller standard (BSI, 2003b).

5.6. Electronic Improvements

Integrated Circuits

PLC technology developed alongside the emerging innovations in electronics and computing. Technologies such as integrated circuits were adopted and incorporated into the controller

Texas Instruments filed a patent in 1974 that described a "Programmable Logic Controller implemented in semiconductor integrated circuits" (Burkett and Henry, 1976). The use of integrated circuits (ICs) allowed the development of smaller and more complex PLCs. Burkett and Henry explained that "Existing programmable controllers constrain programmers to only a few parallel paths in each line or rung of the relay ladder diagram logic". This constraint was "unduly limiting" because of the complexity of the logic applications, programmable controllers were normally applied to. It was desirable remove this constraint by providing an "unlimited number of parallel paths in each line of a relay ladder diagram". Burkett and Henry achieved this by the development of a "one bit word width push down stack" for storing partial solutions when processing the ladder program.

Further considerations for memory constraints were published by Burkett and Henry (1977) in a later patent. The fixed size memory in prior art controllers was a limitation requiring the controller processor model to be exchanged should the memory capacity be exceeded due to program alteration. This was expensive and frequently resulted in over specified processors with greater memory capacity to be used. Burkett and Henry proposed a "variable module memory" where memory capacity could be simply altered by adding or removing memory modules. The advantage of this approach was that expensive memory could be sized accurately to the task in hand without the expense of large unused memory capacity.

Input and Output Modules

One of the primary functions of the Programmable controller is to provide the connectivity to inputs and outputs of the process or machine under control. A modular approach was taken by Seipp (1977) involving the interconnection of specific and separate input and output modules designed to accommodate the signal types of the analogue and digital instruments and actuators connected to the controller. Seipp explained that with prior controllers, a decoder network was required for "assembling and addressing the various input/output (I/O) modules".

To overcome this difficulty, Seipp proposed self-addressing modules so that a "decoder network is not needed in the module and each module can be used universally in various positions in the programmable controller system". This invention proposed by Seipp allowed for the development of universal and interchangeable input and output modules enabling the programmable control system to be easily expanded.

5.6.1. Microprocessors

The microprocessor was introduced in 1971 by Intel (Noyce and Hoff, 1981) and were applied to programmable controllers from the mid 1970s onwards. In 1976, William Seipp of Gulf & Western Industries Inc submitted a patent for the aptly titled "Programmable controller using microprocessor" (Seipp, 1978). The patent provides a brief account of previous attempts to apply microprocessors to programmable controllers and explains the associated technical issues encountered during the early 1970s.

Seipp (1978) stated that "In the last few years, a very advanced data processing device has become commercially available. This device, known as a microprocessor, is sold by various companies". The available "standard" microprocessors were the "integrated circuit technology of the PMOS and NMOS type". Seipp observed that although a number of attempts had been made to utilise the microprocessor for programmable controllers, these attempts had "...resulted in complex software requirements". The explanation for this, noted Seipp, was that the microprocessor was limited in capability for performing simple logic operations because of the following limitations:

> 1) "Since a standard microprocessor includes four or eight data terminals, it is extremely difficult to process single bit information"
>
> 2) "...the internal processing by the microprocessor is somewhat limited so that a substantial number of software steps or program steps must be generated to perform even somewhat simple logic operations."
>
> 3) It was "...difficult to debug a system employing a microprocessor because there is a distinct inability to stop the microprocessor at a selected position and then read the internal condition of the internal registers and modify these registers without complicated software."
>
> 4) "[T]he ability to interrupt the microprocessor for jumping to a subroutine by external stimuli is limited by the circuitry or locations available for this purpose"
>
> (Seipp, 1978)

Seipp explained that using a microprocessor for processing "relatively simple logic conditions" such as AND, OR and INVERT, is "somewhat complicated" because processing the bit information for logic requires "extensive software for masking of data lines and for shifting data between lines".

The primary objective of the invention was to overcome the technical limitations identified by Seipp for application in a programmable controller which can "employ a standard microprocessor without requiring the complex software generally associated with microprocessors" (Seipp, 1978). Seipp achieved his objective by adapting "external circuitry for reducing the software requirements of the microprocessor in controlling a machine or system". The embodiment of Seipp's patent applied a "standard microprocessor", the Intel 8080 to his programmable controller. Further work conducted by Siepp (1979, 1980) entailed the development of further improvements in adapting microprocessors to the programmable controller (PLC).

Allen Bradley also contributed to the development and use of microprocessors at the heart of the PLC. Dummermuth et al (1979) explained that in prior programmable controllers, hardwired logic circuits were required to decode machine instructions; with the use of microprocessors, it was possible to replace the logic circuits with program instructions. Dummermuth's patent described a "Boolean processor" providing hardware assistance in executing bit-oriented instructions. The patent demonstrated the difficulties experienced in resolving single-bit logic equations in the early use and application of microprocessors in programmable controllers.

Brown et al (1981), also from Allen Bradley, further develop the use of a microprocessor-based PLC and extol the virtue that size, complexity and cost can be reduced. The patent particularly emphasises the requirement to provide compatibility with "prior peripheral equipment" (from Allen Bradley) and in particular with earlier programming devices. Further emphasis was placed on memory error checking and detection. Brown et al also demonstrated that Allen Bradley was looking to expand the application of PLC technology in the numerical control system market. The objective stated for the "Mini-programmable controller", was to "provide a small, low-cost programmable controller that can be incorporated in a numerical control system as a programmable interface".

Scan Time

Schultz et al (1987) submitted a further patent from Allen Bradley that described the increasing complexity of the tasks that programmable controllers were applied to. Schultz et al stated that

> "[S]ome of the more complex control program instructions, such as timers, arithmetic, and counters require considerable time. In some applications, therefore, they must forego the use of more powerful instructions for the sake of a faster scan time".

This scan-time restriction clearly limited the application of PLCs for applications that required high speed scan times and hence responses to external conditions.

Schultz et al (1987) proposed a solution to overcome the potential problem of slow scan times by using real-time interrupts (based on using a real-time clock) to service the more complex processor functions (such as timers, arithmetic, and counters). The interrupts are managed by a second processor that dynamically adjusts the interrupt time interval to ensure that the real-time control program is not hindered.

The "Micro-PLC"

The early 1980s saw PLC manufacturers concentrating on size and cost reduction. Masahiro (1983) filed a patent in Japan for a "Sequence Controller" in 1981 providing an example of a single device PLC. This was achieved by connecting a control unit, input output terminals and a power module packaged in a single unit. This type of PLC would become generally referred to as a "Micro PLC" and used for small sequential control applications of typically less than 100 inputs and outputs. The patent is in Japanese with no available translation other than a brief abstract. However, it is worth including because it was developed by Japanese company Sharp. Sharp's PLC marks the start of low-cost micro-PLCs used to automate very simple sequential control applications.

5.7. Visual Displays

CRT Displays

The programming devices developed by Modicon and Allen Bradley started out as simple thumbwheel and switch-based interfaces to program the PLC. Displays were limited to lamps and numerical indicators such as that described by Fletcher and Rosseau (1972) and Kiffmeyer (1974). Kiffmeyer's programming device can be seen in the advert for the PMC in Scientific American (Figure 5.10). The first programming devices were limited in their ability to display dynamic process information and could only show component information such as a single line (rung) number, element position and status (e.g. ON or OFF). Fault-finding was thus a slow and arduous task.

Figure 5.10 - Programmable Matrix Controller Programming Tool (Allen-Bradley, 1972)

Cathode Ray Tube (CRT) displays were not used on or with programming devices until the late 1970s. Hill et al (1981), from Modicon, demonstrated the early application of CRT displays for programming units in 1978. The use of CRTs was enabled with programmable controllers employing integrated circuits and microprocessors, devices capable of driving standard video-based information displays. Figure 5.11 shows Hill et al's programming device with its built-in CRT display alongside the input and output terminals connected to a PLC.

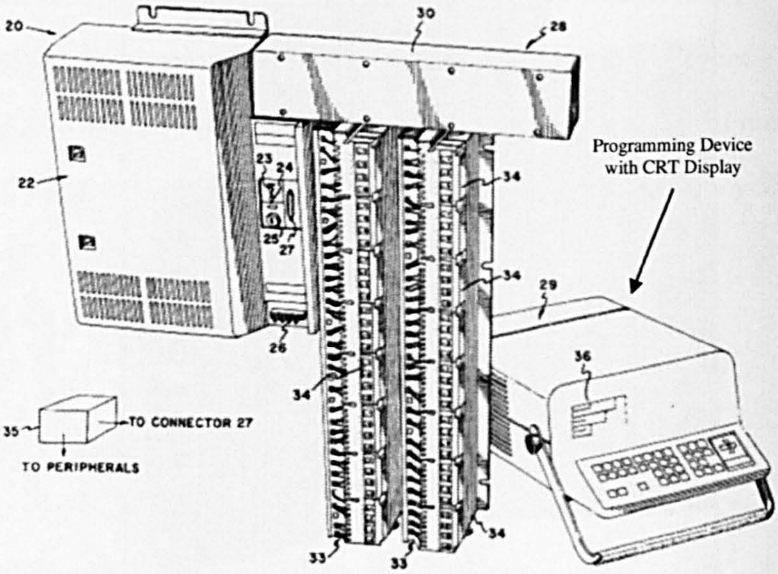


Figure 5.11 – Programming Device (Hill et al., 1981)

Hill et al. explained that the CRT type display enabled the dynamic visualisation of the ladder network and allowed "the user to monitor the real-time power flow at any particular point in the displayed ladder diagram network". This was achieved by allowing "…the user to move a cursor to any node in the network with an associated light-emitting diode (LED) on the programming panel indicating the real-time power status of that node". The control engineer was now able to see complete sections of the ladder rung in real-time and monitor the logical status or "power flow" of the network by displaying the combined elements status. The term "power flow" relates to the combined connection status of the relay circuit used in the electrical ladder notation.

The use of a CRT display for the visualisation of ladder networks also permitted further enhancements with the programming device… "The present invention also provides a programmable controller with improvements not found in prior art programmable controllers, such as the capability of inserting one or more networks between two existing networks" (Hill et al., 1981).

The ability to insert networks into an existing program was an important innovation that considerably eased the modification and "debugging" of control programs. In previous programming systems, at the point of insertion of a new section or rung of ladder logic code, all subsequent lines or rungs would require renumbering. Hill et al enabled new lines of code simply to be inserted at any point the programmer desired.

The basis of Hill et al's patent was an improved programming tool for the control engineer that enhanced the programming, monitoring and de-bugging facilities for a control program. The programming device was specifically designed to perform the task of programming PLCs and was not a general-purpose display. Later programming devices began to take on the appearance of the PC using larger CRT screens such as that used by Schultz et al (1987) shown in Figure 5.12. By the 1990s and beyond, PLC programming devices developed into software applications and were run on personal computers and in particular laptop computers for portability (Clare et al., 2005, Zankl, 2006).

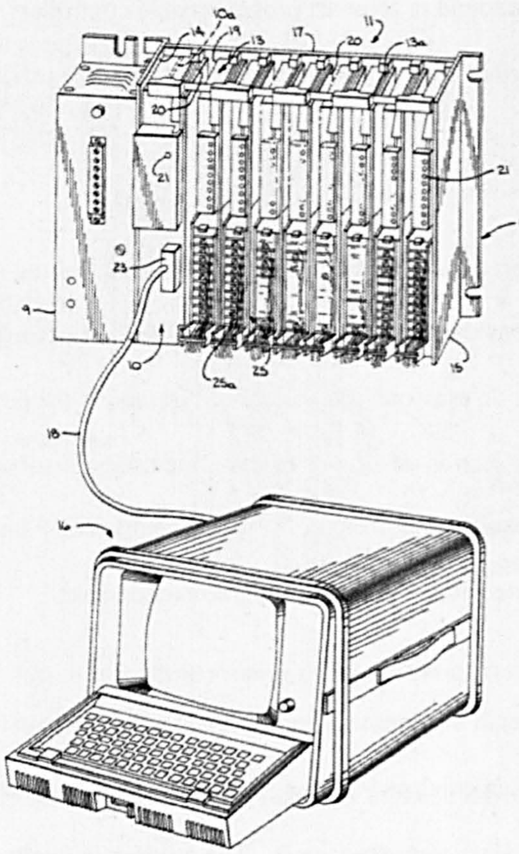


Figure 5.12 – CRT Programming Tool (Schultz et al., 1987)

Process visualisation and the Mimic Panel

Process visualisation had hitherto been accomplished by the use of lamps and buzzers that were driven directly by the PLCs outputs (see Chapter 4). These indicator or "mimic" panels could be located nearby to the machine or plant under control or remotely. Novel solutions such as that proposed by (Macarthur, 1967), using a computer to drive a slide projector were not suitable for most industrial environments.

Saito et al (1986) from Omron Tateisi Electronics Co confirms the fact that although CRT type displays were available, they were not always employed in PLC controlled systems to display information to the operator. Mimic panels were, and still are used to provide an interface to the operator of the plant. Control was performed by switches at the inputs and display information was driven directly from the outputs. Saito et al. explained that driving mimic panel alarms and status indicators directly from the PLC required complex ladder sequences to be directly added to the ladder program. The additional alarm and status sequences in the ladder program increased the complexity and length of the program and increased the scan time of the PLC.

To reduce the status and alarm processing burden, Saito et al. (1986) proposed a solution based on using pattern recognition from the I/O as a separate program not interfering with the cycle time. When a pattern was matched, a particular number is displayed corresponding to a fault or status condition is set which in turn directly drives a display device. Saito et al's patent demonstrated that even by the mid 1980s, memory availability was still an issue for PLC manufacturers and the patent proposed an alternative method for freeing up valuable memory for driving mimics. The additional advantage was to keep program length and hence scan times to a minimum.

Dummermuth (1976) demonstrated that the PLC could communicate with external computers by developing its communications capability. Computers enabled the graphical depiction of plant and processes and led to the introduction of PC-based Human Machine Interfaces (HMIs) for PLC systems. Networked communications capability also led to the connectivity with Supervisory Control and Data Acquisition (SCADA) systems, providing full plant visualisation.

5.8. Analogue Inputs and Outputs

In 1971, Halsall and Murrell (1973) provided an account of the development of analogue codes used to represent analogue input values in a digital form. Their patent is largely concerned with the digital computer control of large chemical plant employing Direct Digital Control (DDC) techniques. Although not directly related to early programmable controllers, the techniques and status of reading analogue inputs as well as the cost implications are discussed. Halsall and Murrell provide background information that relates to the lack of analogue inputs and outputs found with early programmable controllers.

According to Halsall and Murrell, there were two types of computer control available in 1971, supervisory control and Direct Digital Control (DDC). Where supervisory control is employed "use is made of individual analogue process controllers with a digital computer setting the desired controller values and, in some cases, control parameters", whereas "In known systems of Direct Digital Control a digital computer is programmed to carry out all the functions of the individual analogue process controllers in addition to calculating the optimizing data." (Halsall and Murrell, 1973).

The advantage of DDC, Halsall and Murrell (1973) noted, was that

> "...the control of, for example, a complex chemical plant involves the rapid calculation of a considerable amount of process data at frequent intervals and that, digital computers having the characteristic property of high computational speed and specifiable accuracy are particularly suitable for this purpose.
>
> Direct Digital Control systems can generate the control functions by digital means with greater precision than the analogue controllers used in supervisory systems".

On the other hand, "supervisory systems are still preferred in applications to complex plant in those cases where interruption of production due to computer failures would have serious economic consequences". Having established the basic differences in the control of analogue variables, they conclude that "Whichever form of computer control is adopted the problems of feeding analogue process measurements to the digital computer and operating analogue control devices from the digital computer remain."

Halsall and Murrell (1973) go on to explain that special purpose equipment is required to sample multiple analogue inputs and provide analogue to digital conversion for the computer. They concluded that "The cost and complexity of this specialized plant interface equipment is usually several times greater than the cost and complexity of the computer." Additionally, if using a "common signal multiplexer and analogue to digital converter... the failure of which would be as serious as a failure of the computer." Costs would also be incurred when updating existing plant already equiped with conventional analogue control equipment because it would require "considerable modification to the existing instrumentation".

The background related by Halsall and Murrell demonstrated that by 1971, analogue inputs and outputs for computer control purposes was an expensive and complex affair. There were still issues with reliability and clearly concern with entrusting full plant control to the computer. The result was that the computer control of large process plant, dominated by analogue variables, was left to larger specialised computer applications employing DDC. The programmable controllers introduced by Modicon and Allen

Bradley of the early 1970s were designed to provide simple sequential control solutions in a cost-effective manner. The supervisory control model described by Halsall and Murrell was also applicable to PLCs which could control sequences and logic only, leaving analogue parameters (process control) to the established process controllers and instrumentation.

Struger and Grants (1979) provide a useful distinction between the programmable controller and computer process control:

> "The processor in a programmable controller is designed to rapidly execute programmable controller type instructions which call for the manipulation of single-bit input data and the control of single-bit output data. Indeed, programmable controllers are distinguishable in this respect from process controllers which employ general purpose digital computers to control industrial equipment and processes." (Struger and Grants, 1979)

The primary function of a "process controller" is to monitor analogue signals and control analogue devices whereas the PLC "monitors large numbers of single-bit digital devices" and "control a large number of single-bit operating devices". The processor of a PLC "is typically less complex, less powerful and less costly than the processor of a general purpose computer" largely because it "processes single bits rather than words and because it is not called upon to perform substantial arithmetic functions".

Developments in electronic technologies would see the cost of handling analogue inputs and outputs reduce, together with the size of the equipment required to achieve it; microprocessors enabled programmable controllers to add analogue inputs to its repertoire.

In 1978 Gulf & Western Industries (Fauchier et al., 1980) proposed a system for converting analogue signals to multiplexed digital data for programmable controllers. Fauchier et al observed that "In systems to be controlled by programmable controller, it is often necessary to input a large number of analog signals, such as thermocouple voltages, speed analog signals and position analog signals", recognising the requirement for PLCs to read in analogue values. They achieved this aim by inputting analogue signals "into a programmable controller using a microprocessor, with its inherent limitations... using only a single analog to digital conversion circuit". Previous attempts to input multiple analogue signals to programmable controllers, they explained, were based on converting analogue signals individually making the cost of multiple analogue inputs prohibitive. The limitation of the microprocessor was the processing of analogue data and this was overcome by multiplexing multiple analogue signals to a "single conversion circuit" processed by the CPU of the microprocessor.

Further work on processing analogue values was demonstrated in a patent issued by Siemens (Hinsken, 1988b, translated as Hinsken, 1988a). Hinsken used two processors in the controller: a bit processor for binary data processing and a data-"word processor" for digitised analogue data processing. The patent was only published in German but provides a good example of the novel techniques that were employed to adapt microprocessors to process sequential control digital and analogue data.

By the mid 1980s, programmable controllers were able to handle analogue inputs but there is little mention in the patents regarding analogue outputs.

The controller responses to analogue processing appeared to be primarily binary in nature (i.e. digital output responses). Struger and Grants (1979) patent of Allen Bradley provides a useful background to how an analogue input parameter from a position encoder was handled in 1977. The essence of their technique is related to the comparison of the analogue input value to pre-determined limit values (i.e. comparator), setting a digital output accordingly.

5.9. Networking

With their increasing popularity among control engineers, programmable controllers were applied to larger and more complex machines and processes. Some PLC applications provided control over large distances, requiring considerable cable lengths to connect the actuators and sensors. Lengthy cable runs were used to directly connect sensors and actuators to a single controller. An alternative to was to link up remotely located input and output modules connected by a single cable carrying multiple digital connections in parallel to the controller. However, even with direct digital communications, difficulties were experienced with timing and the synchronisation of the remote devices.

Soulsby and Seipp (1978) observed that:

> "One of the difficulties is that the programmable controller operates at a relatively high speed and some of the external locations can not function as rapidly as desired. In addition, if the locations are spaced substantially from the system, the communication time between the various external locations can greatly exceed the normal machine cycle time of the programmable controller. Thus, the programmable controller must wait for external communications".

Their solution used "external addressable terminals or locations serving as input or output units" and, based on using a programmable controller employing a microprocessor, a "system for monitoring the logic conditions at such external addressable locations". Soulsby and Seipp were keen to point out that "the invention has somewhat broader applications and may be used in various programmable controllers".

'Dick' Morley (1979) developed Modicon's (now owned by Gould Inc) programmable controller networking capability further. Morley recognised the potential for data errors and response time due to time constraints in transmitting the data over long distances. He proposed a system that employed a "high speed data bus (N-bus)" upon which remote input/output modules and CPUs operated autonomously under the control of a "master controller". Morley's approach to interconnecting several programmable controllers and external I/O modules, under the control of a master controller, enabled complex distributed control strategies using programmable controllers to be devised.

Two further "Programmable Controller" patents (Provanzano et al., 1981, 1984) filed in 1979 and 1981 respectively by Modicon, demonstrated further developments in networking PLCs. These closely related patents described the use of two databases, one 16-bit and the other 24-bit: The 16-bit database was used for the main logic and analogue processing and the 24-bit database was used to accommodate the communications between external I/O and "closely-coupled multiple CPU's and other peripherals" (Provanzano et al., 1981). Provanzano et al (1984) used a "Fi[r]st-in, First-out" (FIFO) buffer memory for the receipt of "user node data". Provanzano et al (1981) further developed the programmable controller network interface for additional nodes. Both patents described the programmable controller features with particular reference to expansion in terms of local and external I/O and peripheral systems, including the parallel processing of additional controller CPUs (to allow PLCs to communicate with other PLCs).

There were a number of benefits in using Provanzano et al's improvements:

> "The basic system can be modularly upgraded to provide such additional functions as: (1) additional memory with a user space of at least 32,000 words, (2) the ability to directly address up to 64,000 references, (3) a remote I/O up to 5,000 feet from the mainframe central processing unit, and (4) the ability to support computer oriented functions including interrupt driven I/O, process control oriented user languages, mass storage, closely-coupled multiple CPU's and other peripherals" (Provanzano et al., 1981, 1984).

Digital communications enabled PLC systems to employ remote inputs and outputs and take advantage of distributed processing. Modicon had developed an early PLC communications system ('Modbus') that would develop and become part of the fieldbus technology developed in the 1980s (Sauter, 2010).

Digital communication also enabled the PLC to exchange data with other computer systems and peripherals including Human Machine Interfaces (HMIs) that replaced the early special purpose "mimic" panels. Further benefits allowed the PLC-based control system to communicate to higher level computer systems such as "Supervisory Control and Data Acquisition" (SCADA) systems, providing the means to distribute not only inputs and outputs but also control via early "fieldbus" technologies (Rao et al., 1995).

5.10. Chapter Summary

Programmable logic controllers (PLCs) were originally developed to control industrial sequence and combinational logic control and performed the control actions of relays, counters and timers. A key requirement was that they were accessible to the ordinary industrial engineer without the knowledge and experience of computer programming. The preferred "language" of the emerging programmable controllers was based on emulating ladder circuit diagrams, used by electrical and control engineers. PLCs were programmed by a purpose built, detachable and separate programming device or console that enabled the engineer to enter ladder diagram program constructs directly without conversion.

The emergence of the programmable controller (later called the PLC) was an incremental process based on the combined developments in sequential control and computing and the patents published from companies such as Modicon, Allen Bradley, Gulf & Western Industries and DEC demonstrate that fact. Concepts for real-time control included the development of the cyclic program which provided the controller with a deterministic response. The PLC had to compete with existing relay circuit based control systems and the fast cycle time of the controller enabled a response that was at least as fast, if not faster than its relay equivalent. These developments formed the basis of the PLCs ability to perform real-time control and were successively developed by the leading controller manufacturers.

Advances in semi-conductor electronics appear to have significantly influenced the PLC with the advent of integrated circuits and most notably the microprocessor. An early issue was the cost and availability of memory and read-only memories (wired memory and PROMs) were initially employed. The decreasing cost and increasing reliability of semi-conductors enabled the use of read/write memories. More sophisticated programming techniques could be used and the PLC expanded using microprogramming techniques enabling the PLC to use the microprocessor instruction set alongside the ladder notation. The use of smaller and more reliable electronic devices not only reduced the physical size of the PLC but also facilitated other programming features and techniques.

Interrupts enabled more sophisticated programming constructs, for example arithmetic operations, to be implemented without affecting the real-time response. The PLC developed the ability to cost-effectively process analogue inputs and outputs, and digital communications enabled PLC or control networks to be developed, expanding the scope and size of the overall control system. With this additional functionality, the PLC could effectively operate beyond purely sequential control applications and compete directly with larger computer control systems.

Chapter 6 – Industry and Engineering Practice

6.1. Introduction

Manufacturing forms the focus of the applications of the PLC explored in this thesis. However, manufacturing covers a wide-range of industry sectors, each with its own basket of raw materials and consequent hazards and variations in handling and processing. There is therefore a variety of industrial practices that have resulted in different approaches to the control of processes. While the PLC brought advantages to some industries, PLCs also introduced new problems. To gather data on industrial practices and how "reverse salients" (Hughes, 1983) influenced the development of the PLC, interviews were carried out to gather the personal experiences of engineers who commissioned or designed control systems employing PLCs.

The viewpoints expressed in the interviews highlight some of the cultural differences within the engineering profession, and between industrial sectors and even the geographical regions where control systems, particularly PLCs, are used. In industries that are mostly concerned with process control, which are therefore characterised by handling large numbers of analogue inputs and outputs, the process and instrument engineer dominates. Where the application of combinational logic and sequence control is prevalent, engineers were more inclined to have an electrical engineering background and bias (Rameback, 2003).

Different industries manufacture different products (some liquid and some solid, for example) using different techniques and processes. It is not clear that all industries have the same issues, problems and challenges in controlling their manufacturing processes. For this reason interviewees were sought from a variety of different industries. The engineers who were interviewed represented industries that produced steel, agrochemicals, petrochemicals, water, sugar and special purpose cable manufacturing machinery. Additionally, a PLC manufacturer and a process control system developer offered their views on the development of their respective technologies.

Presenting the wide range of primary source material gained from these interviews has proved problematic. The solution adopted in this thesis is as follows:

1. The following three Sections give an overview of the major conclusions that emerged from the interviews, synthesising the information obtained.

2. Important, and quite detailed, accounts of *all* the interviews are presented as the rather lengthy Section 6.5, following the earlier synopsis.

3. Two *full* interview transcripts (Young, 2007, Leeming, 2008) are given as Appendix E to the thesis as a whole, in order to provide exemplars of the approach employed.

6.2. Industry: Influence and Practice

Senior engineers were asked about their experiences and recollections of the control systems they encountered during their careers. Predominantly the engineers interviewed were working in the UK so their accounts and recollections of developments in sequential control, and the introduction of PLC technology, reflect control system developments in the UK. However, the views of engineers in some industries, notably petrochemical and power generation (see Morris, 2007b, Daley, 2010) were indicative of international industrial practice.

6.2.1. Control Strategy and the Choice of Technology

The chemical industry, Leeming (2008) observed, used discrete electrical and pneumatic PID controllers for process control prior to the application of mainframe computers. This industry sector primarily required the control of analogue process parameters. Morris (2007a, b) confirmed that petrochemical refineries utilised mainframe computer control technology.

Leeming and Morris disclosed that the chemical industries were slow to adopt new technologies and were "conservative", preferring to use (or reuse) proven technologies. In contrast, Young (2007) observed that "[t]he steel industry has always been fairly progressive in terms of control...", and this appears to be borne out by the steel industry's early adoption of the PLC. Young described the application of PLCs in the steel industry from the early 1970s; PLCs were then used extensively for sequential control. Leeming recalled that PLCs provided sequential control in the chemical industry when they controlled ancillary equipment and were not a part of the primary control system in a chemical plant. PLCs were therefore restricted to the control of safety shutdown systems and remote plant such as pumping stations.

The prevailing control system chosen by the process industry therefore was the mainframe computer-based control system which was preferred because of the requirement to monitor and control large number of analogue process parameters. The mainframe systems could also provide sequential control and combinational logic but were large, expensive systems that were designed primarily to carry out process control.

Leeming, Morris and Daley (water industry) were working predominantly in process related industries. Hammond (2007) (whilst working on a batch process control project for British Sugar) recognised the need to combine analogue loop control and digital sequential control for sugar processing by developing the microprocessor-based MPC 80. Leeming used process control computers in the guise of mainframes, and later smaller microprocessor-based controllers such as the MPC 80, to control large chemical process plant. Morris and Daley (2010) on the other hand, were more used to applying distributed PLC-based control systems over large geographical areas.

The differences in approach appear to have been influenced by the nature of the process under control (batch or continuous) and the wider social and safety aspects of control. The processes for which Leeming (agro-chemicals) was providing control solutions were batch oriented and affected by variations in the properties of raw materials and therefore dominated by analogue variables. Morris, however, was implementing robust, semi-autonomous sequential and continuous control solutions at remote pumping stations which were responsive to fluctuating consumer demand. There were no deviations in material specifications, such as those experienced by Leeming. Morris also provided supervisory control and alarms for remote controllers distributed over a wide area whereas Leeming's plant was on a single site.

In the steel industry PLCs were used for control because they were fast and environmentally robust and could be placed close to the process under control. "I remember it being very ruggedly built, built like a battleship" recalled Young. The benefits of distributed control were realised because of the requirement for the high processing speed for the fast moving and responsive control required for the steel rolling plant.

For the Original Equipment Manufacturers (OEMs), the decision to select the appropriate control technology appears to be based on economics. OEMs build and maintain one-off bespoke systems designed for a single stage of the manufacturing process (for example, wire-drawing machines or pumps). According to Bruce (2008), it is important for the bought-in control technology to be inexpensive, relatively easy to use and customise. PLCs were chosen as the preferred control technology because a single PLC model was relatively inexpensive, could be applied to different cable machines, did not require external computing resources and were reliable and easy to maintain.

6.2.2. Geo-political Influences

Engineers responsible for sequence control of industrial plant and equipment form "communities of practice" (Wenger and Snyder, 2000, Wenger et al., 2002, Aylen, 2012). Leeming and Young, for example, demonstrated clear differences in the adoption of particular "preferred" technologies within the Steel (PLC) and process industries, reflecting the difference between electrical and process engineers identified by Rameback (2003).

Communities of practice are also shaped by political and geographical influences. Pittwood (2008b) observed, in the UK, larger user organisations tend to use PLC systems manufactured by companies that also provide technical support such as training and technical help. For example water and power generation utilities, are influenced in their choice of PLC by the availability of local product support (See Daley and Morris, section 6.6).

Both the water industry and parts of the petrochemical industry use remote unmanned control systems employing PLCs, linked back to a supervisory control system (SCADA) for overall monitoring and high-level control purposes. Despite this similarity, Morris suggests that the Petrochemical industry ignored national and political boundaries and focused on a small number of suppliers who, for many petrochemical installations did not have a local presence.

There are also cultural differences reflected in the choice of PLC programming methods which, in some industry sectors, differ from one geographic or linguistic setting to another. Pittwood suggests that mainland Europeans have a preference for programming in the electronic logic-gate based notation (Function Block Diagram) and the low-level microprocessor instruction set "Instruction List" (similar to assembly language) whereas the English speaking countries, particularly North America and the UK, have a preference for the electrical ladder diagram notation.

6.2.3. Equipment Standardisation

All of the engineers interviewed attempted to standardise on a single PLC manufacturer, but all reckoned they had limited success.

Peter Bruce provides a good example of a specialised equipment supplier, standardising on a single PLC system and supplying "packages" (component parts of plant that incorporated their own controllers) to larger enterprises. OEMs provided the specialised machinery and packages used in the chemical, water and steel industries. For Bruce, the ideal solution was to standardise on a single PLC manufacturer in order to reduce development and training costs. Development time could be saved because programming code could be re-used on new projects for similar equipment. Familiarity with the equipment from one supplier also curtailed training costs because new equipment from the same supplier was likely to have familiar features. Familiarity also facilitated efficient technical support. The attempt to standardise on a single PLC resulted in Bruce using PLCs manufactured and supplied by Square-D for 15 years. However the "Hi-Draw" company still had to be flexible and Bruce remarked that "we standardised where could unless the customer insisted on a different PLC".

The OEM's choice of control system was determined largely on economic grounds and provides an explanation as to why the standards imposed by the larger organisations could not always be met without implications of increasing costs and time. Leeming confirmed that standardisation was not always feasible when purchasing plant equipment based on OEM "packages" also different sites within a single company had their own preferred control systems. According to Leeming, there was a trade-off between standardisation across the industry and not having "all your eggs in one basket" so that if there was a problem with one system on one site, the same issue would be unlikely to appear on another installation. A mix of technology was also desirable at the plant level so that the risk of a "common mode failure" was reduced or even eliminated.

Standardisation was also limited in the steel and water industries. Young explained that designers of an individual item of plant within a steel works attempted to adopt a single control system as their standard controller and "Whoever got in first to a plant or an area generally put the spike into the ground", giving a monopoly to that supplier. The monopoly was not industry wide though. Daley's experience in the water industry was similar to that of Leeming. Attempts were made to standardise on specific PLCs (GEM 80 and Texas 5TIs), but on occasions, some equipment was supplied with the original equipment manufacturer's (OEM) preferred PLC, differing from the water industry's standard.

The petrochemical industry however imposed its own preferences and avoided deviations set by the OEMs. Morris explained that a global, industry-wide approach is taken to PLC standardisation, crossing national and international boundaries. He observed that the oil and gas industry was very conservative and "it wants to use equipment that has been in use for a long time [and] ...using familiar equipment cuts down on the [installation and commissioning] time". Generally, remarked Morris, "the industry has standardised on Allen Bradley PLC systems".

6.3. Technological Change and Developments

Up to the 1960s, sequential and combinational logic control was generally performed by electro-mechanical relays and rotating drum arrangements. Young (2007) noted that the steel industry had processes that were "very much relay logic based" and that many relays were used for complex sequencing systems such as conveyor and blast-furnace control. These "relay intensive" systems required "extensive maintenance" because of the requirements for frequent design changes to the process and its control and the unreliability of large number of relays despite the reliability of the individual relays. Daley experienced similar reliability problems with relay-based sequential control systems in the water industry. Auxiliary relays expanded the capacity of the control system, however, as the number of relays increased, the reliability of the control system inevitably decreased.

From around 1966, Bruce along with Morris, and later Young in 1970, utilised modular electronic systems to perform logic functions. Devices with names such as the "Bistat", "Norlogs" and Mullard's "Norbit 2" formed logic circuits based on 'AND' and 'OR' gates. However, wiring changes were still required to make any changes to the logic or sequence of control, leading to the inevitable and undesirable lengthy plant shutdowns.

Morris recalled that programmable sequence control systems such as the "Mangol" were starting to appear about 1969; they used "plug-in diode Matrix cards" making it possible to prepare a program offline without interrupting the plant operation. Once programmed the newly programmed matrix card could be exchanged quickly with the card on the plant. While relay-panel changeovers could take up to two weeks (Pittwood, 2008a, b), the use of matrix cards reduced the changeover time so that plant shutdowns were measured in minutes. The disadvantage was the program size and hence the extent of the logical function was restricted to the fixed size of the diode matrix.

The first PLC available in the UK, according to Young and Pittwood, was the Modicon 184 in 1973. Young recalled that his earliest working experience with the PLC was in fact with a "Memocon 184", a "re-badged" Modicon 184 for the Japanese market. In the petrochemical industry, Morris noted the arrival of the PLC and recollected, the advantage of this new technology was that "a new program could be prepared off line, tested and then implemented without requiring a total plant shutdown". Further technological improvements and developments to the PLC followed. The initial developments included faster processors, larger program memory sizes and larger capacity of the digital input and output (I/O) cards. Young observed,

> "it was a very rapid rate of evolution but the technology was still very simple. They were all based on a common design; all had a

central processor with a central stack of I/O and a central power supply to run the lot". (Young, 2007)

6.3.1. Analogue Capability

Following the introduction of the PLC, efforts were directed to developing the I/O capability. Digital I/O expanded from 8 to 16 then up to a maximum of 32 inputs per card. Young explained that "a 64 input card was too large because of the difficulty with connecting external cabling". Pittwood also recalled that improvements in electrical noise immunity were made, leading to the determination of "switching thresholds" and a "time-based window" to counter switch bounce.

With the expansion and development of digital I/O, attention turned to developing analogue I/O and processing for PLCs. Improvements enabled the PLC to process digitised analogue data and set a digital output based on the results of some "analogue computation" explained Young. Young recalled that "[to] start with, analogue measurement was very tentative and slow to develop". Initially only analogue inputs were available for the measurement and display of plant parameters and there was no closed-loop control capability.

The requirement to process analogue inputs and data was recognised by Percy Hammond at Warren Spring Laboratory (WSL). Hammond and his team were tasked with investigating the application of the "new microprocessors" for combined logical and closed loop control, leading to the collaborative development of the MPC 80 controller by WSL, Negretti and Zambra and British Sugar. The MPC 80 was destined to control 'hybrid' manufacturing processes that required both sequencing and process control and was subsequently taken up by the chemical processing industry as indicated by Leeming.

Other controllers were developed on similar lines to the MPC 80 including Turnbull Control Systems (TCS) process controller. However these were regarded as a 'digital controller' (DCS) rather than a PLC and were originally targeted at small-scale analogue control. PLCs developed the capability to perform closed loop analogue control and were also used on some process plant, but not a great deal according to Leeming. The selection was based on whether the control required was more on/off (PLC) or more analogue (DCS).

One of the significant advances in PLC analogue control was not directly related to PLC development, it was the widespread adoption of the long established 4-20mA signal format finally covered by IEC 60381-1 (BSI, 1982) to represent analogue values. Daley, with regard to applications in the water industry, observed that the 4-20mA analogue signal representation enabled instrument manufacturers to concentrate on developing accurate instruments that could output standard signals that were compatible with the PLC. Additionally, Pittwood observed, later instrumentation developments allowed the "intelligence" (analogue computations) to reside in the instrument rather than load the PLCs processor.

Daley remarked that the adoption of the 4-20mA analogue signal standard enabled PLCs to directly compete with DCS. However, not all were convinced that the PLC was a potential replacement. In the opinion of

Leeming, PLC manufacturers would be "dead in the water or they would have a very limited market" if they didn't develop analogue capability but "they were always very weak [in comparison to DCS] in that market".

6.3.2. Ladder Logic and Programming

Pittwood observed that PLCs were predominantly programmed in the "relay ladder logic" notation in the UK. Morris pointed out that before PLCs, sequence control was performed by electro-magnetic relays "designed by electrical engineers on ladder diagrams". Young added that ladder logic...

> "...emulated the ladder network which [electrical engineers] were very familiar with for years. So why change it by using something else if that actually did the business". (Young, 2007)

According to Bruce, people who had an electrical background found ladder logic easy and when handing over a new system to the customer, it was "much easier on-site when trying to explain it to the maintenance guys". Bruce revealed that he was self-taught with regard to PLCs and undertook little formal PLC training throughout his career...

> "I don't think I ever went on a course on any of the PLCs, certainly not until recently. We just learnt it as we went along...The PLC was really designed for people that were familiar with electrical circuits [that] could pick it up without the need for extensive programming skills". (Bruce, 2008)

Leeming and Hammond argued that the process industry, on the other hand, had become familiar with computers, using mainframes and also the small "digital controllers" such as the TCS and MPC 80. Hammond explained that the MPC 80 controller, on its introduction, differed from its contemporary PLCs by including analogue control and in not using ladder diagrams.

> "We were looking at batch processes where you often encounter logical control and sequence control but also some closed loops... We developed a language called "Senztrol" for the MPC80. We kicked off with what was called "Control Basic", really a version of Basic which had instructions that were relevant to control and "Senztrol" grew out of that." (Hammond, 2007)

Hammond confirmed that programming the MPC 80s was still carried out by instrument and process engineers ... "In fact it [the software] was developed by engineers rather than software people" he recalled.

Programming Devices

The early PLC programs were developed in a "suitcase" programmer recalled Young. Pittwood also remembered the early programming device which largely consisted of thumbwheels and pushbuttons to represent elements of the ladder diagram. It was only possible to view one rung of the ladder at a time.

Young explained that the easiest method of reviewing or inspecting a ladder program was to print it out:

> "[Y]ou used to have reams and reams of paper for a large program, churned out from a simple dot-matrix printer... in order to try and find a particular rung, you had to plough through the whole print-out, locate the desired rung and then use the keypad to find the rung on the programming device". (Young, 2007)

Ladder diagram printouts were used to review and even backup PLC programs, particularly for the smaller PLCs. The ladder diagram printouts were retained because "the logic was usually very simple and they could be easily and quickly reproduced" explained Leeming.

Following the "suitcase" programming devices for the PLCs of the early 1970s, hand-held programming devices were introduced to program the smaller PLCs such as the Texas 5TI. Bruce recalled that in 1979, "PLC programming was a bit painful in those days.... the PLC was programmed on big hand-held programmer units in a combination of Ladder Logic and Statement List". Leeming also experienced the Texas 5TI PLC, and described the programming device as a "calculator that plugged directly into the PLC". He also mentioned a later development of the hand-held programmer, which enabled the storage of the control program on a floppy disk in addition to the printed copy.

The further development of PLC programming devices was, according to Pittwood, the addition of CRT displays and keyboard. Bruce recalled the introduction of CRT-based programming devices was in 1984/5. The advantage of this development was that multiple lines of code could be viewed as opposed to a single line at a time as with the older "suitcase" programming devices. Pittwood and Bruce agreed that software packages to program the PLC, which ran on the newly introduced personal computers (PCs), became prevalent. PCs with large CRT displays were not strictly portable and could not be taken on to the shop floor. The emergence of the laptop computer enabled the programming device to be taken out onto the plant or machine and directly plugged in to the PLC.

6.3.3. Structural Developments: Remote I/O and Distributed Control

Young observed that by the first half of the 1980s, the PLC system was still limited to one processor and one set of local I/O, "if you had another process alongside you would need to repeat the whole thing again...We thought it would be nice to get them to talk together" and early attempts to coordinate PLCs required linking one PLCs I/O to another; that is the PLCs connections that were intended to operate plant equipment and read sensor data were appropriated to provide communication links between PLCs. By about the mid 1980s, Allen Bradley came up with a concept called "Remote I/O" that connected PLCs by a digital "communications rail" (network connection). "[T]his revolutionised the physical installation" remarked Young.

Young noted two significant advantages in using remote I/O for the steel industry: it enabled displays to be positioned in convenient places separate from the controls, which were positioned close to the machines and plant; and since digital communication required only one connection, it also reduced the cabling. Prior to remote I/O the PLC positioning was a compromise and cables for each device operated or read by the PLC had to be run back individually to the PLC. Bruce also recognised the benefits of reduced cabling costs using remote I/O for the long wire-drawing machines and noted "Although there was no significant software design savings, remote I/O saved the customer money through reduced wiring". There were no software design savings because each I/O point had to be addressed in the same way as if it were directly connected to the PLC.

Remote I/O only partially solved some technical issues. Young observed that remote I/O introduced a new problem, that of response speed caused by communications latency. The PLC's scan time increased because of more I/O together with a slower transmission rate due to the longer distances involved. Scan time then became a fair proportion of the response time, the impact of this, was that PLCs started to introduce unacceptable delays and consequent errors in plant and process control. The solution to this problem was to give the remote I/O stations their own capability or "intelligence" to control their local processes independently of the master PLC.

Distributing the control "intelligence" throughout the plant opened up opportunities for PLCs to be used in other industries. The water and petrochemical industries for example both had remote processes that extended beyond the boundaries of a single plant. Oil pipeline pumping stations extended many miles between stages and water and sewerage pumping stations could be located miles away from the main control centre. By the late 1980s, local controls were being interlinked to form larger, better coordinated systems, observed Daley. Remotely located stations worked autonomously but under the direction of a central control station.

The concept of distributed control was not overlooked by the process industry. Hammond's earlier work with British Sugar and the development of the MPC 80 controller initially started out using a controller that was dedicated to a single process. Subsequent alterations were made to the control system architecture in order to extend the control system – multiple MPC 80 controllers were connected to and supervised by a master controller, also an MPC 80. "This was called distributed control" remarked Hammond (this work was arguably earlier than that used with PLCs).

The distributed control model was not universally accepted by all industries. Leeming explained that the chemical industry was more reluctant to adopt the networked controller approach, instead preferring to retain a single mainframe computer that connected all instrumentation and actuators (I/O) via a single "marshalling" room or cabinet.

Leeming explained that there were disadvantages when using PLCs on large processing plant with hazardous areas. The purpose of the "marshalling room" was to provide an intrinsically safe connection area for all external signals. The distributed PLC architecture was not suitable because PLCs could not be located directly in the hazardous area. The mainframe model was preferred because on a hazardous plant, it was beneficial to have central control and all connections for external signals located in a safe area.

6.3.4. Robustness and System Resilience

Industrial process plant can be very large and at times dangerous; a failure of the plant or control of the plant could cause grievous harm to personnel and considerable damage to equipment. The control devices and systems, consequently need to be robust and reliable.

Young recalled that in the steel industry as PLC systems became larger and more complex, attention focused on reliability and the consequences of a system failure. In order to improve the reliability of critical systems the concept of "standby" or "redundancy" was introduced using an additional PLC connected to the same I/O as the controlling PLC. This "standby" PLC monitored the performance of the controlling PLC and in the event of detecting a failure, the standby PLC would take over the control of the plant.

The reliability of the control system for chemical and process plant was also of concern. Daley recounts the use of standby PLCs in the water industry. Redundant systems were also used in the chemical industry to back up critical plant, recalled Leeming; this was to alleviate the risk of a "single point failure". The general objective was to safely maintain production in the event of a failure.

Morris related a different approach taken by the electrical power generation and petrochemical industries, where PLCs were used for monitoring and shutdown systems. As with other industries, safety is a high priority but the approach taken on petrochemical plant was not to provide redundant control systems that could continue operation in the event of a fault but to provide PLCs that shut the plant down safely when a fault was detected. Morris explained that in the oil industry, for reasons of safety, if something goes wrong you stop.

> "[I]f the PLC doesn't work you can just pull the plug out, switch it off all together and it will stop everything... The oil industry doesn't mind about that, it can always start up and get going again". (Morris, 2007b)

In electricity generation a higher premium is placed on continued operation.

6.3.5. Documentation

One of the key attributes of the PLC was that it was easy to program by the engineer and technician. Pittwood observed that as a consequence "Many PLCs were programmed on the plant itself during the commissioning phase and as a result, documentation tends to be poor". Additionally, PLC programs could be modified in situ and bore little resemblance to the original design. Pittwood also disclosed that individual engineers and technicians often had their own personal, and frequently different, copy of a program that they loaded in the event of a problem.

The introduction of CRT displays and the PC not only allowed programs to be easily copied and backed up but also enabled comments and notes to be added to the program; "this was an important step in documenting programs" explained Bruce. Other benefits of documenting systems were expressed by Daley, who remarked that for the water industry, systems logged all interactions and changes to the plant and identified who made them. The benefit of this documentation system was that it increased security and helped to prevent personnel making unauthorised changes to the control system on the plant.

Documentation, its accuracy or the lack of it, appears to have been an industry-wide concern. The chemical industry for instance, tackled the issue of documentation in a formal way. Leeming explained that "Applied Version Control" rules were applied to the mainframe systems and the PLC systems (although the rules were less stringent). Leeming conceded that the software was harder to control for PLCs.

Pittwood concluded that PLC manufacturers are now developing tools to assist the documentation process and introducing a systems approach to development. However, these tools are not the full answer, – "A disciplined approach is required throughout industry and this is a role that the engineering professions are now taking a lead in".

6.3.6. User-led development and SCADA

Engineers and plant operators play a role in the selection, deployment and configuration of PLCs. PLCs provided the means to alter the program in situ and this was often carried out by the plant and maintenance engineers. The PLC made it easy to change and modify the program "and that gave the plant operators tremendous flexibility to be able to modify the process, on the fly in many cases" (Young, 2007). The use of PLCs enabled "user led designs" where engineers were able to implement solutions requested by the users (managers and operators) during the commissioning phase which was a considerable advantage (Daley, 2010).

One component of operator acceptance of automation technology was providing plant visualisation via Human Machine Interfaces (HMIs) and Supervisory Control and Data Acquisition (SCADA) systems. Young explained that it was desirable "to let the operator have sight of the plant". Initially this was achieved with hard-wired "mimic panels" and later by pre-designed remote operator panels.

A particular technology that provided connectivity to PLCs and enabled plant-wide visualisation was SCADA. Computer-based SCADA systems were connected to controllers on industrial plant and provided graphical representations of the plant's state and supervisory level controls from a PC terminal. SCADA systems were brought into the PLC market and duly adopted by the steel industry. "Once SCADA was on the scene, it really started to revolutionise the PLC market at that time" remarked Young. The water industry also used SCADA technology to provide an interface for remote PLC controlled applications. Daley however noted there were "Spin-off" effects, associated with the automation of the water industry and the application of distributed control and SCADA systems. In particular the developments reduced the number of people required to operate the plant.

Industries that used large numbers of PLCs had a tendency to use SCADA, for example steel and water. This was not the case for smaller machine suppliers or OEMs making bespoke automation systems. Bruce pointed out that SCADA was too expensive for their wire-drawing machine application and that they were not involved in factory-wide systems, instead developing and writing their own PC-based visualisation application. However, hardware and software interfaces were provided to enable their system to communicate with external SCADA systems.

The chemical industry, as Leeming stated, preferred mainframe systems over PLCs for their core process automation and control technology and did not use SCADA extensively. The mainframe computer-based control provided process visualisation through graphic displays and so SCADA was not required in many cases. "SCADA almost passed us by to be honest. We were either in mainframes or sometimes we added SCADA onto PLC systems but not a lot. It didn't really fit with the strategy at the time" remarked Leeming.

6.3.7. Technical Standards

IEC 61131 and in particular "Part 3: Programming languages" made some improvements on standardising the programming notation. According to

Pittwood, prior to the introduction of IEC 61131, manufacturers produced PLCs with their own versions of programming languages, for example the Ladder Diagram notation. Pittwood has observed that now, most manufacturers have followed the conventions used in the standard but the programs are still not portable between the different manufacturers PLCs.

Leeming commented that Fieldbus, the standard for communication between control and instrumentation equipment, had too many variations and no standardised version, "a similar argument to Betamax Vs VHS". Although no single Fieldbus standard emerged, the multiple communications standards were published and became "open standards". Daley noted that PLC manufacturing companies were forced to adopt open systems so that the end-users were not locked into proprietary standards. This meant that components from different suppliers had a degree of compatibility.

6.4. Engineering Practices

Bennett (1996) and Bissell (1999), in researching the history of automatic control, show that control engineering has exploited results and attracted the attention of workers from diverse disciplines. Control engineering's applications in a variety of disciplines was reflected in university teaching toward the end of the 20th Century. Bissell (1999) commented that "In the U.K., there are few specialist first-degree programs in control engineering; control is more commonly studied as mandatory or optional modules in electrical, electronic, chemical, and mechanical engineering degree programs". Bernstein (1999) takes a similar view, explaining that "control engineering tends to be the least tangible of all subjects in the engineering curriculum [which]...depends on technology, and this technology is highly interdisciplinary".

At the same time each discipline introduces its own cultural variations. The interview data shows that within engineering there are cultural differences between the electrical "automation" engineer and the process or "instrumentation" engineer. The steel, chemical and sugar industries provide examples exemplified by Young, Leeming and Hammond's approach to sequence control.

The steel industry has many processes requiring robust sequential control (although the control of analogue parameters such as temperature, speed and position are still important) and has a significantly high proportion of digital to analogue inputs and outputs. Many of the processes require the control of large electrical machines (motors, actuators and so on), so many plant engineers have an electrical engineering background. Young, an electrical engineer, demonstrated that the steel industry was a progressive and enthusiastic user of PLC technology, actively involved in developing and implementing innovations such as remote I/O and distributed control.

The chemical industry on the other hand was a leading proponent of the application of mainframe computers shaped in part by the nature of the materials and processes under control. The presence of aggressive materials and explosive hazards also led to a more conservative approach to new technologies. As Leeming explained, the chemical industry is largely engaged in continuous or large batch process control, and as such is dominated by the control and monitoring of analogue process parameters. There is clearly a requirement for the sequential control of processes but this is minimal when compared to the number of analogue parameters associated with large chemical plant. Process and instrumentation engineers feature significantly in the design and application of control systems in the chemical industry, and PLCs are primarily purchased as components of fixed "packages", invariably manufactured by OEMs.

The water industry operates continuous processes that clearly require the monitoring and control of analogue parameters in purifying water and treating sewerage (pressures, flows and water quality). The petrochemical industry also has a need for process control in its refineries. In addition, both industries are involved in distribution over wide geographic areas that which necessarily introduces remotely located processes requiring repetitive sequential control of, for instance, pumps and safety shutdown systems.

Daley recalled that there was a certain "differentiation" between the instrumentation engineers (process bias) and the "automation" engineers (electrical bias) in the early days. However certain technological developments eased the gap between the two: PLCs gained the ability to process analogue input signals; and accept the standard 4-20mA signal familiar to the instrument engineers. Daley observed that the compatibility of the PLC with a standard instrument analogue signal representation meant that the "automation and instrument engineers were now working together... This led to integrated systems where engineers thought more about systems rather than their own direct discipline".

Pittwood and Hammond highlight the distinction between Software engineers and maintenance engineers. Pittwood explained that the term "configurable" was initially used in relation to PLCs rather than the term "programmable", in order to make the PLC more acceptable to the electrically biased shop-floor engineers and technicians. Hammond suggested that in the 1980s, software people were very different from engineers, "software engineers were concerned with quasi theoretical concepts and mathematical algorithms and so on".

The interviews indicate that the application of PLCs and sequential control is undertaken by disparate industries which are typically "interdisciplinary" in nature. There appear to be some national norms, such as the near universal use of ladder logic in the UK, and the fact that PLCs intended to be used by electrically biased engineers. There are also industry specific influences, which include different engineering disciplines and their associated cultures that exert influence over what technology is used to perform the control task at hand.

6.5. Detailed Interview Accounts

This rather lengthy section provides the results, presented as distilled accounts, of the interviews with senior engineers in industry. All of the engineers were highly experienced in the early and developing use of PLCs. This section gives an account of each interview rather than a full transcript, for reasons of clarity and brevity (but two example full transcripts are given in Appendix E).

6.5.1. Steel Industry

Interview with Eur Ing David Young (2007) and further correspondence (Young, 2009). The transcript of the interview is contained in Appendix E.

Eur Ing David Young is a Chartered Engineer and Fellow of the Institution of Engineering and Technology (IET) and has worked as the Operations Electrical Engineer and Manager of Project Engineering within the steel industry for over 41 years. Having retired in 2001, he has worked as a consultant on power systems (calculation, protection and coordination), safety, regulation and procedural type activities; he also acts as an expert witness.

Precursors to the PLC

Young's experience dates back to the 1960s prior to the introduction and use of PLCs. According to Young (2007) "The steel industry has always been fairly progressive in terms of control and a lot of the processes which were in the steel industry were very much relay logic based". Complex sequencing systems containing many conventional electro-magnetic relays were used for conveyors, blast furnaces and for steel plants. The "relay intensive" sequencing systems were housed in hard-wired panels that consumed power and were difficult to modify and change. Although they were very reliable, they did still fail and so required extensive maintenance. The steel industry, like a lot of process control industries, was very much subject to change because the plant was designed and installed to do a certain series of functions. However it was not uncommon that soon after installation somebody wanted it to do something different.

In the early 1970s Young experienced solid-state logic gate systems in the steel industry based on AND, OR, NAND, NOR and XOR plug-in cards. There were a number of those on the market including: "Bistat" made by "Brookhirst Igranic", later absorbed by Cutler Hammer (TNA:PRO-CR420, 2012); a modular system by Square D; and a further system utilising "Norlogs". Young also remembered a German system, as he remembers called "Controlmatic", which had a different approach but again it was using electro-magnetic devices "Unfortunately they were quite difficult to work with and the schematic diagrams were not the easiest to follow" (Young, 2007).

First PLCs

The first PLCs appeared in the steel industry in the early 1970s and were American based "…one of the early ones that I saw in the early 1970s was a Modicon 184, marketed in the UK by [electrical switchgear manufacturer] MTE". It was promoted in Japan as a "Memocon" by Yaskawa and had a central processing package with a separate power pack (see Figure 6.5). "I remember it being very ruggedly built, built like a battleship; it had separate I/O as most PLC systems even to this day have. The Modicon was one of the earliest processors that I had any dealings with but it was typical of the type of processor unit we had" stated Young (2007).

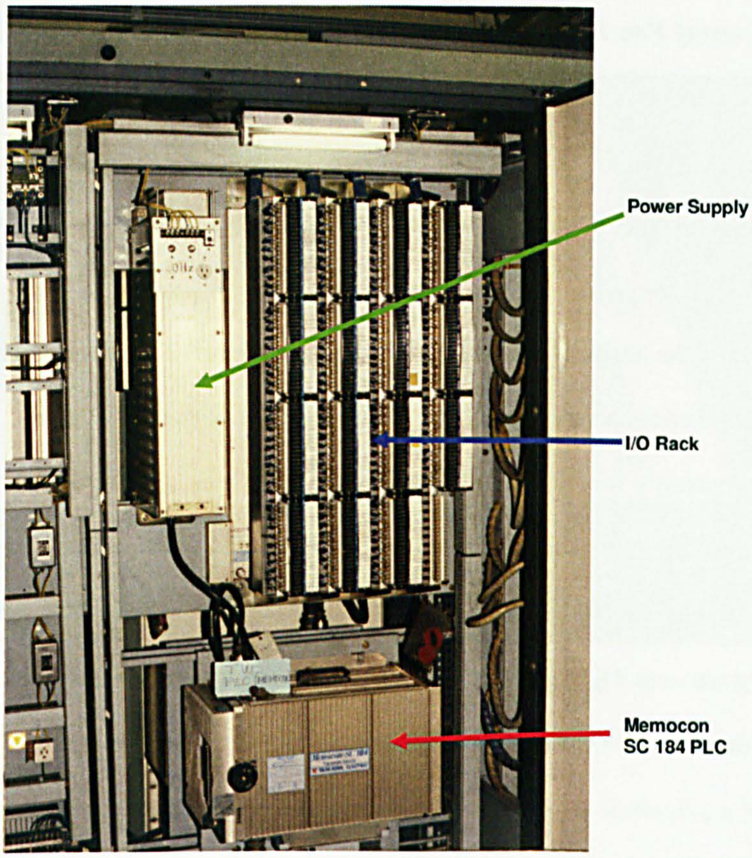


Figure 6.1 - Japanese Memocon SC 184 (Modicon 184)

Other PLCs available in the early 1970s were Allen Bradley's "2" series and the Texas TI range, a smaller type of PLC, with about 200 I/O in total. In the late 1970s further PLCs emerged such as GEC's GEM 80, produced in 1979 (Anon, 2007) and the "shoebox" PLCs by Mitsubishi and Toshiba having just 16 I/O for placing on a dedicated machine tool or a piece of control system. Young explained that "There were many of these PLCs and the development was very rapid at that time, something new coming out every week". PLC manufacturers and suppliers were expanding memory size and the number of I/O cards "it was a very rapid rate of evolution but the technology was still very simple. They were all based on a common design: all had a central processor with a central stack of I/O and a central power supply to run the lot" (Young, 2007).

Programming Panel, Ladder Logic and Memory

The PLC itself was all programmed in ladder logic and was designed and configured to suit electrical people who were familiar with using relay logic on ladder networks. It "emulated the ladder network which they were very familiar with for years. So why change it by using something else if that actually did the business" (Young, 2007). The very early PLCs had about four input devices per rung with an output at the end of the rung. The program memory was just an integrated part of the processor and invisible as far as the programmer was concerned. It was there so you could put timers, latches, flip-flops, delays and inversions into the program. The memory size was 4K and 8K in the early PLCs which were often restricted to approx 100 or 200 I/O.

A dedicated programming panel was used to program the PLC. Each PLC manufacturer had their own purpose built and designed programming panel that "looked like a suitcase". It had rows of little switches and lights to show the programming sequence as you went through it. Well many had push-buttons but it was dedicated hard type components [switches] for programming the PLC, so this all went into "magnetic-based" memory. The memory that stored the program on the PLC was limited "and the ladder network could only be played back line-by-line... you could only interrogate it rung-by-rung". Figure 6.6 shows a more advanced programming device and has "virtually" a QWERTY keyboard. It used LEDs to display the register contents, so you were looking directly at the rungs, line by line.

Figure 6.2 - "Suitcase" Programming Device

Young explained that the easiest method of reviewing or inspecting a ladder program was to print it out "you used to have reams and reams of paper for a large program, churned out from a simple dot-matrix printer". Young goes on to explain that the PLC ladder program was not easy to interrogate and "in order to try and find a particular rung, you had to plough through the whole print-out, locate the desired rung and then use the keypad to find the rung on the programming device". Program annotation was unknown in the early days of the PLC.

According to Young, one of the great advantages of the PLC system was that changes could be made very easily. "Anyone who could find a way through the rungs of the program could very quickly put another function, another set of contacts or another dependency". The PLC made it easy to change and modify the program "and that gave the operators tremendous flexibility to be able to modify the process, on the fly in many cases". Prior to that, it was resolved by changing the wiring [hard-wired changes]. "And of course it [the PLC] was reliable" concluded Young. In order to retain a copy of the program held in PLC memory, programs were stored on a special purpose manufacturer's magnetic cassette tape via the programmer. The program could be recorded and theoretically be reloaded back into the system if it crashed.

Analogue Inputs

As the PLC evolved, it first increased the amount of I/O and "then of course analogues were the next development" remarked Young. "In the steel industry, we wanted to measure things like position for instance, which was a very common requirement... Position was one of the earliest ones that I came across, mainly from potentiometers giving a position, driven by a motor shaft driving a gearbox which in turn was driving a roll or similar".

Other analogue quantities common to the steel industry included temperature, weight and flow. "To start with, analogue measurement was very tentative and slow to develop" explained Young. Initially, the requirement was to know the physical position of that in the PLC and display that on the ladder network. Closed loop control was later, explained Young, so initially it was analogue inputs only... "We were then able to take analogue signals directly into the PLC and do some [form of] computation... so analogue computation came into being, because you could then manipulate the analogue signals".

PLC outputs relating to analogue inputs such as weight derived from load cells, tended to be digital. "The weighing process always ended up with an output which meant that it needed a signal to drive an actuator somewhere to open a gate or a vibrating feeder, to control the feed of material onto a conveyor belt" (Young, 2007).

Remote I/O and Distributed Processing

The late 1970s and early 1980s saw developments relating to PLC memory and Inputs and Outputs. CPU memory started to expand to 32k, 64k and so on. I/O got bigger starting with 8 inputs per card, then 16 then 32. A 64 input card was too large because of the difficulty with connecting external cabling. Cabling was a big problem. There were an enormous number of multi-core cables across the plant to bring the inputs and outputs to the central processor as signals were derived from sources often 100m plus from the processor – this was very expensive.

The PLC system was still limited to one processor and one set of local I/O, "if you had another process alongside you would need to repeat the whole thing again" explained Young. An early attempt to get PLCs to talk to one another was by virtue of digital or analogue inputs and outputs, linking one PLCs I/O to another. "There was no communication capability at machine code level at this stage. We thought it would be nice to get them to talk together".

By the mid 1980s, the breakthrough came "I think it was Alan Bradley who introduced 'remote I/O' – what a breakthrough that was". From the PLCs central processing unit (CPU) you could have a 'communications rail' (network) that connected inputs and outputs at remote locations.

> "We used to drop an I/O rack, out in the middle of the plant somewhere, run one communication cable to it and a power supply and then connect the I/O to the CPU as if it were local... What I can say is that this revolutionised the physical installation" (Young, 2007).

This also meant that more things in terms of displays, control and information could be put on the system. "Plant operators wanted indicator lamps, switches and all sorts of things to be driven in remote locations by remote IO". Unfortunately it was not all good news, explained Young, because this introduced the problems of scan time and associated delays in system response.

One of the drawbacks of the early PLCs was that all sensors and actuators had to be connected directly to the input and output (I/O) channels of the Central Processing Unit (CPU). Locating a PLC centrally resulted in long and complex cable runs prone to pick up electrical noise from the heavy electrically driven machinery, common in the steel industry. The introduction of remote I/O placed near the individual units under control and communicating digitally with the PLCs CPU via a single cable reduced problems of interference. However, remote I/O introduced another problem, response speed caused by communications latency. For each cycle of the program, the CPU had to read the inputs and write to the outputs so scan time then became a fair proportion of the processing time. The impact of this was that PLCs started to introduce delays and hence errors in plant and process control applications.

Young explained this problem by describing an early experience of a PLC control system using distributed inputs for rolling mill positioning control.

The problem was identified as the time taken to transmit position measurement data back to the PLC, process the signal and then send a response to set an actuator. In essence, the scan time of the remote I/O was found to be too slow to allow a timely control action to stop a 10 ton steel sheet at the right point without overshooting its target. A solution was proposed that reduced the communication latency by locating an additional small PLC close to the machine that could receive data and control the process directly without the need to wait for a response from the central PLC. This main controlling PLC was then just transmitting and receiving time independent data and not directly controlling the process. This 'distributed processing' approach increased the opportunities to exploit PLCs in large plants, particularly those covering physically large areas (Walker et al., 2010).

In the steel industry, passive and intelligent remote I/O, allowed the connection of remote sensors and PLCs to a single master PLC via a digital connection, which reliably transmitted the signals without degradation over large distances. Digital communications enabled PLCs to interface with existing control and instrumentation systems and co-ordinate independently controlled smaller areas of plant or clusters of machinery (Wareham et al., 1988).

Programming Unit

Along with the developments in remote I/O and networking in the mid-1980s, there were also advances made to the programming devices.

> "Clearly if you have remote I/O, you have to be able to address those remote racks"... we needed to have the capability of having proper programming facilities to sit with the PLC itself" (Young, 2007).

This came from the use of early [personal] computers and was purely a terminal to interrogate the CPU. The very early ones were DOS based systems and a tremendous step forward from the earlier technology.

> "You would plug the programmer PC into the CPU rack then you could look at the program displayed on the screen rung by rung. This enabled you to program the PLC, address the remote racks and address the I/O all from a PC based programming tool which was a revolution at the time" (Young, 2007).

Hot Standby Configuration

PLC systems were becoming larger and more complex so attention began to be focused on reliability and the consequences of a system failure. "People at that time didn't have that much trust in computer type technology, early computer systems were prone to crashes for reasons that were seldom explained". Often, the only way to resolve the situation was to reboot the system which took a long time using tape systems with no guarantee of resolving the problem. In order to improve the reliability of critical systems the concept of "standby" or "redundancy" was introduced. An additional 'standby' PLC would be connected to the same I/O as the controlling PLC and monitor its performance. In the event of detecting a failure, the standby PLC would take over the control of the plant. Various manufacturers such as Allen Bradley (now part of Rockwell Automation), Modicon and GEC (GEM 80 PLCs) developed this concept and came out with a capability called "Hot Standby Configuration".

HMIs & SCADA

There was a need in the steel industry for remote operator panels out on the plant. Invariably these would be made up of push-buttons, switches and lights (mimic panels), driven by the remote I/O. In order to minimise the effort required in designing and building a unique panel for every application, Rockwell came out with a pre-built operator panel called "Ready Panel". This was a small pre-configured piece of I/O with blocks of keys set out in a 8x8 matrix which could be used as switches or lights. It connected to the PLC as remote I/O and was configured in software; the lights and switches could be physically labelled on the plant by the engineers and operators. This innovation not only saved engineering time but also reduced the cabling requirement on the plant, "It was easy to do and I've used dozens of those on various installations".

As the control requirements for steel plants became more complex and sophisticated, the industry moved away from "simple control" and operator panels. It was desirable "to let the operator have sight of the plant" or put another way, have a graphical representation of the whole plant. Supervisory Control and Data Acquisition (SCADA) systems were brought into the PLC market and duly adopted by the steel industry. Computer-based SCADA systems linked to the PLCs controlling the plant and provided graphical representations and supervisory level controls from a PC terminal. "Once SCADA was on the scene, it really started to revolutionise the PLC market at that time" said Young.

A further advantage of SCADA is that alarm systems are easily incorporated and alarm management (prioritisation) is inbuilt. Hard wired alarm annunciators are not required and precise alarm timing and history is readily available. Additionally, it enabled the trending of numerous plant parameters for analysing plant performance.

Standardisation

Young recalled that although remote I/O through digital networking was achieved, there was no universal network. Companies like GEC, Modicon, Allen Bradley (AB) and Telemecanique (TX series) all had their own systems but they wouldn't talk to each other. In the steel industry, this led to plant standardisation on a single PLC control system supplier for individual plant within the works. "Whoever got in first to a plant or an area generally put the spike into the ground" giving a monopoly to that supplier. "This is exactly what happened" explained Young, "within the works the Blast furnaces for instance were very much Modicon but the Steel Plant were AB and the Rolling Mills were very much GEM 80"

Even within the same works there was no commonality explained Young, "[if] someone would buy a dedicated piece of equipment from Germany, it came with a Siemens Simatic PLC of course". The end result was that the steel works ended up with hybrid control systems, with each plant using a different PLC system depending on the function and purpose. Dedicated equipment that was purchased as a whole often came with its own control system that had to be linked in to the plant control system.

6.5.2. Agro-chemical (Process) industry

Interview with David Leeming (2008). The full transcript of this interview is contained in Appendix E.

David Leeming has worked in the chemical and process industry since 1968. Starting as a Control and Electrical Tradesman, he was subsequently sponsored to study for a degree and graduated in 1975 working as a professional maintenance engineer. Leeming has extensive experience in the maintenance of process plant and the manufacture of chemicals, agro-chemicals and hazardous materials (e.g. Nitro-Cellulose). He has served in a number of senior engineering roles throughout his career including the Syngenta Site Engineer (Huddersfield) and latterly Global Head of Compliance Engineering for Syngenta.

Industry Background

The agro-Chemical Industry consists of large-scale "active Ingredients plants" with high volume throughputs and often consisting of multi-stage batch operations and occasionally continuous process plants. 90-95% plants are controlled by Distributed Control Systems (DCS) that use mainframe computers to ensure the reliability and continuity of the product. Quality and consistency are important factors to ensure that there are no differences between batches in order to produce the "same product, batch after batch after batch". A DCS was used to control process reliably by removing the variation you can get from different shifts - "5 shifts and therefore 5 different ways of operation".

Plants manufacturing hazardous materials, such as the Nobel Nitro-Cellulose plant, are physically smaller because the explosive hazard is "much more intense". These plants tend to use mid-range PLCs. In particular, the Negretti MPC 80 type systems were used because they were able to handle analogue measurements (the Negretti MPC 80 systems were really small process controllers or "digital controllers" with analogue and digital capability). In the late 1970s, normal PLCs were limited or unable to handle analogue signals. It was rare to see mainframe computer control systems (DCS) used on Nobel's plant except on the larger Pentaerythritol tetranitrate (PETN) plant "because that was beginning to get to the scale of the kind of plant used at Syngenta, Huddersfield. Most plant used discrete instrumentation or PLCs on operating units to give a "localised level of automation".

The control technology used was determined by the scale of plant. On large chemical plants PLCs were used mainly on "packages" such as air conditioning units and compressors. You would frequently find PLCs on formulation and packing plant which were of a standardised and repeatable design. PLCs were also found on older assets (plant) controlling more critical units.

Early Automation Technologies

In the Early 1970s agro-chemical processing plant used discrete PID controllers and instruments to control the process. This also included some pneumatic control because electronic technology came in later, between 1975-1980. From 1974 the Huddersfield plant became fully computerised with mainframe computers using Ferranti Argus and Kent Systems. These were applied to large scale plants of 1200 I/O or more, having analogue inputs such as temperature and pressure. The analogue outputs driven by the computers were typically for valves. The control strategy for smaller plant, prior to PLCs, would be discrete controllers. Supervisory Control and Data Acquisition (SCADA) systems were not used much.

> "SCADA almost passed us by to be honest. We were either in mainframes or sometimes we added SCADA onto PLC systems but

> not a lot. It didn't really fit with the strategy at the time". (Leeming, 2008)

Early PLC experience

PLCs were introduced and used from circa 1980 to replace relay sequencing units. However, there were early issues with using PLCs because they can "fail very differently to relay systems". It was soon realised that the "health" of the PLC had to be monitored via "watchdogs" (error checking software or hardware) for example. There were also some instances where PLCs had "reversals of outputs" (a logic '1' represents 'OFF' rather than 'ON') when compared to other control technologies. A serious problem related to the potential for a "common mode failure"[11] when using shared PLC operating software. The "common mode failure" was eliminated by using different PLCs (hardware and operating systems) therefore removing the common mode.

The chemical industry was good at collecting information, reporting and learning from the facts gathered. Eventually it came up with a shared view on how to implement and use PLCs "that probably said... be a bit more conservative, fine for running sequences but don't put your safety systems on them". PLCs were then generally used to replace timers and relay sequencers.

The first PLCs Leeming used were Texas 5TIs which were purely digital and fairly basic. The Texas 5TIs were "re-badged by Siemens as the S7-200". Some Allen Bradley PLCs were also used because they had analogue

[11] In this instance, Leeming is referring to a single fault condition arising in a particular manufacturers PLC (hardware or operating system), which can affect multiple plants that employ the same type of PLC (i.e. the same fault appears on all).

and control functionality plus controllers from Turnbull Control Systems (TCS); the TCS was really a digital controller (small DCS) rather than a PLC and was originally targeted at small-scale analogue control. The selection was based on whether the control required was more on/off (PLC) or more analogue (TCS).

Reliability and Backups

Memory for the smaller controllers, such as the Negretti controllers was held on Erasable Programmable Read Only Memories (EPROMs). The settings for the controller were extracted and stored in a database together with a backup set of the EPROMs. No electronic backup was kept for the Texas 5TI PLCs, only the ladder diagram printouts were retained because "the logic was usually very simple and they could be easily and quickly reproduced".

In terms of PLC software Control, the same "Applied Version Control" rules were applied as for DCS: systems were reviewed monthly; necessary housekeeping was performed for backups and modifications; and backups were refreshed where necessary (on some older systems, memory could dissipate over time). Things were more relaxed with PLCs and housekeeping was carried out approx once every 6 months or on a shutdown or replacing equipment. However the software was harder to control.

For the TCS, the technicians had access to the programming unit and could change the settings. They were encouraged and trained to understand the technology (TCS) to realise the impact and implications of any changes. Attempts were made to make certain settings unalterable, but to a point it didn't matter, everything safety related was not on that kind of system. Access to the programs was restricted and was kept with the professional engineers, the technicians were provided with program printouts to fault-find. The Texas 5TIs were like "black boxes" and the programs couldn't be altered, printouts were provided for fault-finding as per the TCS.

Standardisation

Attempts were made to standardise the control equipment used on the plant, however this was met with limited success. It was achieved to some extent with the smaller digital controllers such as the Negretti MPC 80, 84 and 85. Standardisation was also attempted with the PLCs but this had ended up with some "creep" depending on the supplier of the equipment and their preferred embodiment. It was very expensive to change a supplier's preferred system so you accepted what came with the equipment on "packages".

Standardisation was achieved with the larger mainframe computer based DCS. The Huddersfield site used Kent systems which had Batch control, tracking and handling features. At the time, the Kent DCS was "miles ahead of the competition" because competitors were just set up for continuous control (e.g. refining). The Kent batch mechanisms and in particular understanding of the process requirements was very good. The Huddersfield site had two generations of Kent, the K70 and K90. After the K90, ICI developed an in-house designed system. The latest plant at Huddersfield, the "Paraquat" plant is moving to the Siemens PCS (Process Control System) which uses the same I/O cards for the S7 PLC range. Other sites had their own preferences and some chose the Delta-V DCS for their applications.

There were some benefits in having diversity in suppliers for different sites because you were protected from a business risk perspective (i.e. if one supplier folds).

DCS vs PLC

PLCs were considered as an alternative control technology to the DCS but there were some disadvantages when using on large processing plant with hazardous areas. The distributed PLC architecture was not suitable because PLCs could not be located locally to the hazardous area. The preference is to connect all electrical systems in a safe are known as a "Marshalling Room" with all external signals between the marshalling room and the plant made intrinsically safe via a barrier. With all plant signals concentrated in a single area it made sense to use the mainframe model based on the DCS. There were other significant benefits from having a control system centred in one location:

1) Operators are located in a safe area.

2) Central control of plant, work and permits. "The more distributed… the less control you've got".

3) The control room became a "hub of knowledge" in one place or area. The single point failure risk of the DCS was alleviated by employing redundant systems.

There were further advantages the DCS has over the PLC. Some Chemical processes require greater supervision and intervention because the materials specification can change. Materials can be out of, or at the extremes of the specification and that might affect the reaction. On-site technologists constantly monitor the processes and need to understand and change the key parameters to make the correct product.

Programming Devices and Displays

The Texas 5TIs had a small programming device that looked like a calculator that plugged directly into the PLC. The device had a "program store" so that a control program could be downloaded to the PLC from a computer (PC) or uploaded from the PLC to the computer. Files were saved on a floppy disk for backup. The Negretti MPCs had a dedicated "Engineers terminal" which required an access code (password) to gain access and make changes. Password access was the same as Kent systems. The DCS also had dedicated engineer's terminals that were fixed (not portable) because of the centralised control. The early DCS systems in the 1970s also provided rudimentary CRT mono-displays for the operator. Both the engineers and operator terminals were text-based with displays consisting of limited variables and simple graphics. By the 1980s DCS developed more sophisticated graphics or "mimics" to show greater detail on the screen.

It was not possible to extract much information from the PLC for display on the DCS so PLC information was displayed via directly connected hard-wired mimic panels. SCADA didn't appear until the late 1980s when they could offer more functionality. However it was possible to look at the I/O status of the PLC directly from its input and output cards (LED status displays).

Control System Developments

In 1979 the underlying technology at the Stevenston plant was DCS which was coming in at that time. DCS were expensive systems so you needed a large plant for them to be cost-effective. However, there was also a lot of installed pneumatic control technology on the site. Two developments came in at that time that allowed improvements to be made to the pneumatic systems. One was the PLC which allowed the plant to move away from huge relay and timer panels, which were complicated, expensive and in some places dangerous. Then there were developments in the process control side which came up very quickly. The biggest development was with the digital control systems like the TCS. They had more functionality than the pneumatics or even the analogue electrical controllers.

As an example a pH control system, which was essentially was pneumatic other than the pH measurements was converted to TCS control. The pneumatic system had 21 components and parts to it. After conversion to the TCS the system ended up with only about four components. The conversion from pneumatic to full electronic control dramatically increased reliability and gave other improvements in control. This was significant, it brought precise digital control to the discrete level, and you didn't have to have it at a major plant level. After the introduction of control systems like the TCS, it was a case of deciding what technology (DCS, TCS or PLC) fits the unit size operation.

The next significant development was the introduction of PLCs with analogue I/O and control algorithms. In the opinion of Leeming, PLC manufacturers would be "dead in the water or they would have a very limited market" if they didn't develop analogue capability but "they were always very weak [in comparison to DCS] in that market". PLC manufacturers had their strength in machinery control, guarding machinery and production lines. "There are no production lines in a chemical factory". It was much more likely to see PLCs on "package deals": compressors, fridge units and air handling units – all fairly standard equipment. Alternatively PLCs were used on small units where there is digital and analogue with the right functionality to suit a PLC.

The next development with PLCs was Remote I/O & Fieldbus. ICI didn't go in that direction because the selection of I/O was hindered by flammable atmospheres; remote I/O has to be located in a safe area or inside something safe (enclosure). This approach introduced safety issues, when working on the equipment, for example maintenance, where a "Certificate of Safety to Work" would then need to be issued to work on something that was not certified for the hazardous area. The best option was to locate the I/O and PLC in a central area.

Fieldbus also had too many variations and no standardised version, a similar argument to Betamax Vs VHS (see also Felser and Sauter, 2002, Thomesse, 2005, Sauter, 2010). In Leeming's experience, when Fieldbus emerged (1990s), there was never much confidence by process engineers in Fieldbus but there was confidence about the existing design of plant (centralised I/O and marshalling). With other emerging technologies such as RF networking, the process industry was never the first adopter of this type of technology. It would try it after it became established & becomes the norm, "with handling dangerous chemicals, you can't afford to take chances".

Leeming stated other shortcomings of PLC technology applied to the chemical industries. It was limited in I/O size, so was not applicable to the large process plants. The PLC manufacturers approach was from the digital on/off control end and the chemical industry approach was more about measurements. Although the industry uses a lot of digital I/O (press switches, valve position sensors) there is more analogue and DCS are built around that kind of data processing. However, recent developments have shown that some PLCs and PLC components are used with DCS, for example the Siemens Process Control System (PCS).

6.5.3. Discrete Manufacturing

Interview with Peter Bruce (2008).

Peter Bruce started working for the Atomic Energy Authority at Harwell in 1966, straight from leaving school. Whilst at Harwell, he had the opportunity of working with computers, a subject that he took up at university a few years later. There was no exposure to PLCs at university but he did carry out some significant work with Transistor–Transistor Logic (TTL) and Boolean logic. In 1978 Peter Bruce started working for a company called 'Hi-Draw' based in Romsey that made machinery for the wire and cable industry. He was particularly involved with designing and building the control systems for wire-drawing machines. From 2004 Peter Bruce started his own business working in the same industry, designing and supporting wire-drawing machine control systems.

PLC Projects

One particular project, in 1979 involved linking a computer with a PLC, which was one of the first of its kind recalls Bruce. Hi-Draw made a wire-drawing machine for a company up in Warrington that was the first one that used a PLC and a computer on the same machine. In the late 1960s and early 1970s most computers were in buildings using punch cards. "Certainly, when PLCs first came along, computers were still relatively large devices and didn't really lend themselves to machine control". They were programmed in high-level languages, for example Basic and Fortran and were used for business applications like accounting. The first PLC to market recalls Bruce, was the Modicon PLC in the early 70s and later in the 1970s and early 1980s was the GEM 80. The Warrington project wasn't that successful, concludes Bruce "not because of any wrong principles but because the hardware wasn't reliable". The crossover between computers and PLCs was very tricky in those days.

The PLC used on the project was a "Reliance Automate". It was one of the very early PLCs, a huge American built machine, and "we used a DEC PDP 11 computer". The essential problem was the communications between the PLC and the computer, it wasn't reliable. The computer was used for monitoring the status and faults of the wire-drawing machine and setting parameters such as speeds. Basically the communications kept breaking down and the information on the screen would disappear. The PLC itself was OK in terms of controlling the machine, because it was effectively used as a relay-replacer. Programming the PLC "was a bit painful in those days", there were no computers to program the PLC in 1979, it was programmed on big hand-held programmer units in a combination of Ladder and Statement List. After that project, we used much smaller PLCs made by people like Telemcanique for very simple relay replacement.

Square D PLCs

We had an order for a cable machine for another project in 1982, and were asked to build a system to monitor PLC diagnostics, speed etc. Hi-Draw started working with a company called 'STC' which had started to develop PLCs. Unfortunately, half-way through the project they decided to withdraw the PLC so left us a bit "high and dry". We responded to this predicament and had a quick look at the market and decided to use a "Square D" PLC.

In 1982, Square D seemed to have "leap-frogged the other PLC manufacturers and had come up with a PLC that seemed pretty good" and "ahead of the field". Square D had a plug-in "computer module" that could be programmed in Basic. It only had a small memory, about 9K, but was compatible with Square D equipment and could talk directly to the PLC. In fact it connected directly into the same rack as the PLC and could be plugged straight into a monitor for display and a printer. Square D was the only company at the time that had this technology which was useful for connecting to a display and printer that could be configured by them. This system was a successful combination and Hi-Draw used Square D PLCs and equipment for about 15 years. They were "very good, reliable and we hardly had any hardware problems … so we standardised where we could unless the customer insisted on a different PLC".

Alternative PLCs

In 1993 Hi-Draw was asked by an American company to build a machine to manufacture hydraulic hose who insisted on using Allen Bradley PLCs. By the early 1990s Square D seemed to have slowed or reduced their development of their PLC range so the company used Allen Bradley PLCs until 2003, when it closed down.

Other alternatives were considered such as Siemens PLCs, however they were found to be difficult to use and not user friendly, "you couldn't do anything without having to go on a course". Part of the problem was that the Siemens' PLCs "were too flexible, it had too many functions and was difficult too understand", especially if you had a ladder diagram background (Siemens used Function Block Diagrams (FBD) and Statement List). If you came from an electrical background, ladder was much easier to understand. So they were difficult to use for electricians and were "much easier on-site when trying to explain it [the system] to the maintenance guys". Ladder provided a definite advantage in industries that were not used to computers.

The initial learning-curve was very painful from some of the European manufacturers and there were cultural difference between Europe and America. "The Americans [Allen Bradley] tried to make things 'idiot proof' and their software was very user friendly". It must be said that Siemens are much friendlier now.

Future application of PLCs (direction of PLC technology)

PLCs are very rugged devices and Peter Bruce expects them to be around for a long time – however, he concedes that this is "possibly coloured by my own experience". There are now smaller PLCs on the market with their own programming environments (e.g. Siemens LoGo) for controlling machines with just tens of I/O. PLCs will still be used for control systems consisting of up to a few hundred I/O. For larger systems, consisting of thousands of I/O computer-based control systems such as the "Soft PLC" or DCS are likely to be used.

Human Machine Interfaces (HMIs)

HMIs have improved dramatically over the last few years. The hard-wired mimic panel didn't suit the wire-drawing application and displays needed text information (status and alarms) and data displays (e.g. speed), so computers were ideal. Hi-Draw started to use IBM Personal Computers (PCs) because they were reasonably priced when compared to the larger mini or mainframe computers. They were also OK on the factory floor because the dust from the wire drawing process was not conductive and (at the time of interview, 2008) early PCs are still working in some plants.

However there are some problems with using PCs:

1) Support for PCs is now difficult because they become obsolete after 3-4 years with no continuing support.
2) Original software won't run on new PC operating systems.
3) PC input cards don't fit and run on modern PCs due to changes in PC architecture.
4) You can't replace the original disk in the original PC because it will be incompatible with the old PC architecture.

The industrial HMI has now been chosen to overcome these problems explained Bruce. There are no moving parts and reliability much greater than with PCs plus the support better and more long-term from the HMI manufacturer.

Hi-Draw looked at SCADA software in the early 1990s but didn't adopt this technology for their systems. The company was not involved in factory wide systems because they were more machine builders. SCADA was also very expensive for their application. To enable their system to communicate with external SCADA systems, they provided hardware and software interfaces such as compatible communications ports and "data exchange" registers. Hi-Draw decided to develop and write their own bespoke program and this software was used until the company until it closed in 2003.

Remote I/O and Fieldbus

The physical size of the wire-drawing machines lent themselves to the application of remote I/O and Fieldbus and in particular, Allen Bradley's "Devicenet" was used. Each wire-drawing machine has 7 or 8 blocks with one central control station housing the PLC. Remote I/O was used on each block and connected back to the PLC. "Although there was no significant software design savings, remote I/O saved the customer money through reduced wiring" said Bruce.

Hi-Draw used sub-contractors to build machines so the company was really a systems integrator that carried out the design and implementation of the completed machine. All software was implemented in-house making it easier to commission and support the installed systems on the customer's premises. This made sense because they would have to produce a detailed specification to hand over to a third party, so Hi-Draw "might as well have written it in the first place".

Programming Device

Initially, Square D had a specialised programming device, "which was large, heavy & painful to use". The PLC was programmed in Ladder. The smaller PLCs, Telemechanique and Toshiba, were programmed by small handheld programmers using fluorescent or Liquid Crystal Displays. They were generally programmed in Statement List (Assembly Language) because of the small display. It was not possible to put notes on to the program using the handheld programmers, which was a disadvantage.

Larger programming devices were developed around 1984/5 using CRT displays and a full keyboard. The advantage was that programs could easily be copied and backed up. Another advantage was that they enabled comments and notes to be added to the program; this was an important step in documenting programs. With laptop computers from the mid-80s onwards, PLC programming software packages became more prevalent. Laptops could be taken out onto the plant or machine & directly plugged in to the PLC.

6.5.4. Petrochemical

E-mail correspondence and Interview with Alan Morris (2007a, b).

Alan Morris graduated with a BSc in Electrical Engineering from the University of Manchester Institute of Science and Technology (UMIST) in 1966. After gaining his degree he worked for Fairey Engineering at Dungeness "B" Nuclear Power Station on safety interlocking of irradiated nuclear fuel dismantling. In 1968 Morris moved to Simon Carves as an Instrument Engineer working on large chemical process control projects including ICI's Terylene plant in Northern Ireland and Russia. From 1975 Alan Morris has held senior instrumentation and control system engineering roles in the Petro-chemical and power generating industries on a wide variety of projects, retiring in 2006.

History

Morris recalls that prior to 1968, plant monitoring and control mainly used pneumatic instrumentation; electronics had started to make an impact and was only just making its present felt in the production industries. Sequence control was performed by electro-magnetic relays "designed by electrical engineers on ladder diagrams". The early 1970s saw the application of computers for process control and electronic devices such as Mullard's "NORBIT 2" were beginning to be used for some combinational and sequential control applications in the process industries. The NORBIT 2 devices were silicon-based electronic logic circuits built from discrete components and encapsulated in a container. The device looked like a large 'Integrated Circuit' (IC) because it had two parallel rows of connecting pins

(Anon, 1962, Dean, 1973). Morris remembers that the main difficulty with these independent logic circuits and relays "was that any change in the sequence required rewiring, taking time and needing a plant shutdown while it was done".

1969-1970 saw the arrival of the 'Mangol' control system, designed by ICI and automation products from INSTEM Computer Systems Ltd. Both systems were programmed on a plug-in diode matrix card, which made it possible to prepare a new program while the plant was still operating. A plant shutdown was still required to implement the change but this was now reduced to a "matter of minutes to swap [the] cards". The INSTEM product was used on a terylene plant, which was also a success.

By 1972 the 'Honeywell TDC 2000' Distributed Control System (DCS) was announced, providing integrated monitoring, process and sequence control and at about the same time, the first PLCs arrived. The PLCs with their software programming was a significant development because "a new program could be prepared off line, tested and then implemented without requiring a total plant shutdown".

Morris recollects that "there were some initial technical issues with the early PLCs, notably with the PLC outputs". The output units "were based on industrial computer practice", where relays or solenoids were connected directly to the power supply and switched on or off by earth switching or 'earthing' the other side through the PLC. It was noticed that if there was an earth fault on the return cabling to the PLC, the relay or solenoid would be permanently energised, regardless of the output state of the PLC. This was notified to the PLC manufacturers who "changed the output unit design to one where the switching was done on the supply side". In the event of a cable fault the relay would now safely de-energise.

From 1979 to 1986 Alan Morris was involved with PLC based monitoring and safety shutdown systems for the oil industry in the Middle East and noticed a shift in preferred PLC suppliers. Initially, the majority of the early PLC systems in the oil and gas industry used Modicon PLCs until "[Modicon] decided to concentrate on the US automobile industry" leaving other PLC manufacturers like Allen Bradley to take over. The majority of systems used in the oil and gas industry now are Allen Bradley. The oil industry is very conservative, "it wants to use equipment that has been in use for a long time" and what people are familiar with. "It is very difficult to introduce something new and untried into the oil industry...using familiar equipment cuts down on the time".

Industry Differences

Around the late 1970s and early 1980s, PLCs began to be used for the monitoring of Fire & Gas detection systems. Legislative developments in the oil industry became even more concerned about the safety and reliability of software systems with "predicted availability figures of better than 99.9%" being required. In response to this demanding target, the concept of using "an internal periodic test program" to confirm correct operation was put forward. According to Morris, "It was more than a 'watchdog'; it was checking that the entire operation was working correctly". This concept allowed the availability figure to be upgraded "due to the certainty that the

PLC was operating correctly". This concept was later incorporated in the Pepperl & Fuchs and subsequent "fail-safe" PLCs.

One of the distinguishing aspects between the oil industry and the power station industry is the reaction to a fault. In the oil industry, for reasons of safety, if something goes wrong you stop… "if the PLC doesn't work you can just pull the plug out, switch it off all together and it will stop everything… The oil industry doesn't mind about that, it can always start up and get going again". A power station is different; you don't necessarily want to stop and disrupt the supply unless it is a safety issue. If something goes wrong, you want to keep going and give people time to try and sort things out. So there is a difference in attitude between the two industries.

Power station people are very worried about spurious shutdowns. The process can be operating perfectly correctly but you could get a false input signal caused by a blocked pipe or something and the PLC would shutdown the plant. This would be a major disaster for a power station; they need to keep going whenever possible, even through these spurious shutdowns. The 'internal periodic test program', used in the oil industry led to more spurious shutdowns when it was applied to the power industry; it was less reliable than it would have been without that program. "But without that program, nobody would use if for safety systems so something done with the best of intentions ended up making the situation worse" explains Morris.

According to Morris, the oil industry as a whole tends to follow American practice so at first Modicon PLCs were dominant then Allen Bradley PLC systems and very few others. "Power stations were slightly different; they tended to be more nationalistic" says Morris. "A power station is built near where people live and is generally surrounded by all the support industries that it requires". In a country where most of the equipment has to be imported, the choice of control system used comes from, or is imposed by, the equipment supplier. "You tend to find that if for example ABB build a power station, the equipment comes from Germany, where if the French company Alstom build it, the equipment supplied is all from France".

6.5.5. Water Industry

Interview with Anthony Daley (2010).

Tony Daley started his career as an electrical power systems engineer in 1961. After working for a period as a power station engineer in Canada, he returned to the UK to work for the Liverpool Water Authority, later known as North West Water. On his return to the UK, North West Water was investing in their engineering infrastructure and in particular, their control systems. Tony Daley's experience as a senior electrical engineer in the water industry spans over 40 years where he has been involved in the introduction and application of sequential and process control technologies.

Historical Developments

By the late 1960s and early 1970s most control in North West (NW) Water was achieved manually by direct electrical switching (pumps, valves etc), sequence control was accomplished by using relays. Problems were encountered when sequences needed to be changed because often complex relay circuits needed to be rewired. Further complications occurred because the number of contacts on an individual relay was limited so auxiliary relays had to be used to expand the capacity of the control system (auxiliary relays are additional relays connected to and activated by the main relay to provide more contacts). Using large numbers of relays in a control system reduces system reliability because of the combined failure rates of the individual relays. Reliability problems were further exacerbated as a result of the increasing circuit complexity because of the potential for mistakes in the wiring diagrams, generated when systems were modified.

In the late 1970s, control was becoming increasingly important and relevant to the water industries. In 1979 Tony Daley attended a number of trade shows that raised his awareness of "computer-based control systems" or PLCs, such as the Texas 5TI. The PLC could be used as a direct alternative to the existing relay-based sequential control panels and they began to be employed widely within NW Water. PLCs provided the flexibility that was not possible with hard-wired relay panels because the program could be altered without changing the wiring. The use of PLCs enabled "user led designs" where engineers were able to implement solutions requested by the users (managers and operators) during the commissioning phase which was a considerable advantage.

Culture

Daley remembers that there was a certain "differentiation" between the Instrumentation engineers (process bias) and the "automation" engineers (electrical bias) in the early days. However certain technological developments eased the gap between the two cultures: PLCs gained the ability to process analogue input signals; and the evolution of the standard

4-20mA signal format to represent analogue values. The 4-20mA analogue signal was an important step which signifies when the instrument output is lost: 4mA equates to the lower instrument range limit, if the signal was less than 4mA or non-existent (0mA), then the signal can be assumed to be corrupted or lost.

With the 4-20mA analogue signal representation, instrument manufacturers were able to concentrate on developing accurate instruments that could output standard signals, which were then processed directly by the PLC. The compatibility of the PLC with a standard instrument analogue signal representation, meant that the "automation and instrument engineers were now working together". This led to integrated systems where engineers thought more about systems rather than their own direct discipline.

PLC Developments

Advances in electronics and speed provided a gateway to automation and control in the water industry. PLC control systems grew larger because of their developing ability to be connected together. Prior to Fieldbus communications technology, this was usually achieved by connecting PLCs to a "master PLC" to control the communications. By the late 1980s systems were being developed from the ground-up and local control was beginning to be interconnected to larger systems. An example of the design and implementation of such a system is the implementation of the "Control and automation of the Manchester to Liverpool sludge Pipeline" (Daley, 1989a, b). Things changed with the emergence of Fieldbus systems. PLC manufacturing companies were forced to adopt open systems so that the end-users were not locked into proprietary standards. This meant that different components could be used from different suppliers.

Systems were being built with reliability aspects in mind. PLCs were more reliable than relays and redundant systems (standby PLCs) and Uninterruptible Power Supply (UPS) systems were used to improve reliability still further. Control system standardisation made sense and NW Water tried to standardise on PLCs but this was not always possible. The PLCs selected were Texas 5TIs and GEM80s but some plant and equipment was supplied with the equipment manufacturers preferred PLC on occasions. Ladder Logic was the preferred programming language, which now is much more functional & complex.

There were "Spin-off" effects, associated with the automation of the water industry. Automation required less people to operate the plant, especially with distributed control and SCADA systems. It also highlighted and hence stopped personnel making unauthorised changes to the control system on the plant. Systems were able to log all interactions with the plant which would identify the changes and who made them, enabling the appropriate corrective action to be taken.

6.5.6. British Sugar

Interview with Percy Hammond (2007).

Percy Hammond is a chartered engineer whose interest in control engineering stemmed from his work as a project leader for the National

Physics Laboratory (NPL) Ergonomics Division at the Warren Spring Laboratory (WSL). Hammond's interest related to 'control concepts in biological structures' and 'bioengineering' and applying these concepts to industrial control applications (see Hammond, 1967). In 1978 Percy Hammond set up the Control Engineering Division at WSL (TNA:PRO-AY, 2012), initially researching areas of application for microprocessors. Percy Hammond left WSL in 1981 and took up the position of Director of the DTI's *Computer Aided Design Centre* in Cambridge.

History

The National Physical Laboratory (NPL) and WSL had been working on computer control for a number of years using minicomputers. These were "all the rage" and there was a lot of minicomputer control going on at the time. When microprocessors came along, a colleague called David Williams recognised that some of the minicomputer control requirements could be tackled by the new microprocessors because they were powerful enough to do the job.

> "The Intel 8080 microprocessor in particular, there was also the Motorola 6800, [they] captured the imagination of what could be done with this little chip, which was quite amazing." (Hammond, 2007)

The questions that had to be answered were: how do we do it? and what are the functions that we need to apply? It was rapidly discovered that both logical control and closed loop controls were needed, "so we had to try and combine both of them". This idea was really a natural progression from what was going on in the use of minicomputers.

As part of a research programme, WSL conducted a large survey looking at a number of industries including the food industry to assess possible application areas for PLC control and also closed loop control. Quite a lot of development work was initially undertaken with Heinz in West London, Park Royal, with things like the canning process for heating and cooling, which was quite an obvious candidate for microprocessor control. We had to switch between one temperature and another in a sequential timed-control way. The analogue signals were handled individually and the signal conditioning was carried out using A/D converters.

Having surveyed the field, the team decided that British Sugar was the firm to work with. WSL were a government organisation so not industrially oriented but saw their job as introducing industry to new ideas. The idea was to choose a production or manufacturing company to support and provide WSL with experimental facilities and to find an instrument company that would develop any instrumentation that would come out of it, Negretti and Zambra were chosen.

The collaboration between WSL, British Sugar and Negretti and Zambra led to the development of the "Microprocessor Controller" using the Intel 8080 microprocessor or "MPC 80", which not only had programmable logic control but also closed loop control. This differed from other PLC suppliers such as Allen Bradley at the time who was concentrating on relay replacement systems and WSL thought it was essential to combine logic control and closed loop control. Along with Percy Hammond, Peter King did a lot of the original thinking and development on the MPC 80 project and the history and implications of their work is well documented (Anon, 1978, Hammond, 1980b, a, Hammond and King, 1980).

Programming the MPC 80 at British Sugar

The MPC 80 controller differed from its contemporary PLCs by including analogue control and also in another significant way, the programming language. In order to combine the features of sequential control and process control, WSL chose to develop a new programming language based on 'BASIC'.

> "We were looking at batch processes where you often encounter logical control and sequence control but also some closed loops... We developed a language called "Senztrol" for the MPC80. We kicked off with what was called "Control Basic", really a version of basic which had instructions that were relevant to control and "Senztrol" grew out of that." (Hammond, 2007)

The programming of the MPC 80s was purely carried out by engineers. "In fact it was developed by engineers rather than software people" recalls Hammond. In the 1980s software people were very different from engineers; "software engineers were concerned with quasi theoretical concepts and mathematical algorithms and so on". The technicians and operators at British Sugar were involved in using and maintaining the control systems.

The first phase at British Sugar was based on using controllers that were dedicated to control a single process; the process was to control an evaporator in the production of sugar crystals (to evaporate the water content). Subsequent developments were made to the MPC 80 control system architecture. The concept was to move away from the dedicated controller model and move to a larger control system with the dedicated controllers supervised by a master controller. The master was another MPC80 and so was identical in terms of hardware. It was called distributed control.

Introduction of Automation Technology

Percy Hammond recalls the impact that automation had on the workforce. "We only encountered the unions once throughout this work and that was at Cadbury's Chocolate bean factory". Union members approached WSL to do a study in their factory. They obtained permission from Cadbury's management because they wanted to know how microprocessors might improve the efficiency of the operation. The process was cocoa beans to paste for chocolate manufacture and moisture content was critical. The unions paid for the work and this was the only time that the unions were involved. "We never had any problems with the operators in those days; they were all prepared to embrace new technology" claims Hammond. "The essential thing was to talk to them, to tell them exactly what we were going to do and how it would help them, not put them out of work and that sort of thing".

6.5.7. PLC Manufacturer

John N Pittwood - Newcomen Society presentation (2008a) and interview (2008b).

John Pittwood, I Eng, started his career with an engineering apprenticeship and then as an electrician in the steel industry. Following his apprenticeship he has held positions as a contract Electrical Engineer in diverse industries such as fire detection, boiler making and mechanical handling. He joined Merlin Gerin in 1976 as an Applications Engineer for their PLC range and has stayed with the company ever since. Merlin Gerin merged with Telemecanique in 1981 and was later acquired by Schneider Electric in 1988, owning the Modicon, Square D and Merlin Gerin brands. John Pittwood has held a number of different positions in the company including Sales, technical management and project management. At the time of interview in 2008, he was the UK Marketing Manager for Energy Infrastructure Systems and Solutions.

Introduction of the PLC

In the 1950s, factories had little automation and manual labour was invariably used for assembly tasks. Where there was automation, it was mainly simple relay logic. By the 1960s, the automotive industry identified the requirement for automation and relay-based control panels were built to conduct sequential control on the production lines. The sequential control system could be highly complex and control panels became considerably large and could be up to 30m long. With the use of large, complex relay panels, production model changes required significant alterations to the relay panels. Changeovers could take up to two weeks in order to carry out the design, re-wiring and subsequent time spent on fault finding. Factories lost two weeks of production during this process which was deemed too expensive (Pittwood, 2008a).

Attempts were made to apply early computer control on some production lines as an alternative to the relay-based control. However, the computers used were not industrial (unable to withstand the harsh environments) and required skilled programmers, not generally found on factory floors. As a consequence, a specification was issued by General Motors describing the US automotive requirements for a digital sequence controller to replace the relay-based control:

1) It should be "quick" [fast in operation].
2) Scalable.
3) Smaller [than the relay panels].
4) Industrialised [environmentally robust] – able to withstand electrical noise, vibration, wide temperature range.
5) Configurable by [electrical] engineers and technicians.
6) Easy to fault find.

In response to this specification, Dick Morley in 1969 developed the Modicon 084 (84th project from Bedford Associates). The cost of Modicon's device was £3,500 and the Modicon 084 is now an exhibit in the Smithsonian museum. The next model by Modicon, the '184' was developed and released in the UK in 1973. (Pittwood, 2008a)

Pittwood (2008a) described how the original Modicon 084 "programmable controller" met the key requirements stated by General Motors. The Modicon 084 was a specialised computer for controlling a machine rather than processing data. It was more robust than its contemporary computers because it used a ferrite core memory rather than electro-mechanical memories such as magnetic drums. The Modicon 084 was able to withstand the vibration and dust issues associated with the shop floor as well as limiting the potential for human errors; there was no ON/OFF switch so the

084 could not be switched off unintentionally or otherwise. The issue of electrical noise was tackled by determining switching thresholds. 0-24Vdc digital signals were used and >18V represented a 'True' and <5V represented a 'False' logical values. A "time-based window" was used to counter switch bounce.

The term "configurable" was initially used rather than "programmable" to avoid confusion with computer programmers and to make the new controller more acceptable to the shop-floor engineers and technicians. The controller was programmed in a format that was based on the electrical diagram conventions based on American 'JIS' drawing formats using ladder diagrams, familiar to the plant engineers and technicians. The new "configurable" device also allowed the reuse of code. Or put another way, one programmed solution could be re-used again for a different application or motor control for example.

John Pittwood (2008b) was first exposed to the new "Programmable Controllers" in 1973 at a conference held by electrical component manufacturer MTE, who were promoting solid state relays (SSR) and control technologies. His objective for attending the conference was to investigate using SSRs for sequence control at a Rover car production line. Pittwood recalls that the afternoon presentation was about a new product called a "programmable controller", the Modicon 184 and he recognised that this was a viable alterative to SSR and traditional relay technology.

At the time of the conference in 1973, the programmable controller was rejected by Rover but two years later Rover began using PLCs for controlling their manufacturing plant. Customers were initially reluctant to try the new PLC technology because they were very expensive, due to the fact that PLCs were built to very high engineering standards using very robust but expensive devices such as core stores (Pittwood, 2008b).

Programming Device and Programming

The first programming device used thumbwheels and push-buttons to represent the ladder diagram and it was only possible to view one line of code (rung) at a time. The program was entered line by line using the following sequence:

1) Line Number
2) Line type
3) Element
4) Element type
5) Element Reference

This sequence was repeated for all lines required in the program. Each line contained four input elements and one "output coil" only. Code requiring more complex instructions had to be built up from multiple four-element blocks. The result of each line provided an output that could also be used as an input for later blocks of code.

One of the key developments made to the programming tool, was the use of CRT displays" remarked Pittwood. With the early "switch-based" programming tool, only one line of code at a time could be monitored. When CRT-based devices were introduced, multiple lines of code could be viewed. Other developments saw the hardware programming device move to a software application when PCs and laptop computers came into use.

The first programming devices were generally quite expensive and it was difficult for users of small PLC-based control systems to justify the cost. For some of the smaller PLCs, particularly the cheaper Japanese versions, the software was issued free of charge. However, technical support for these systems was very limited and these systems were typically used to automate small systems. The larger PLC manufacturers charged for their programming devices and software but provided ongoing technical support. For customers or end-users with significant automation projects or controlling large expensive plant or processes, this was an important feature.

Programming Notation

Some geographical and political preferences in the method of programming PLCs have emerged over the years. Relay Ladder Logic was the programming convention used in the early PLCs which is still particularly popular in North America and the UK. European (excluding the UK) PLC users and manufacturers tend to prefer the "logic gate representation" (Function Block Diagram) and the low level "machine code" ('Instruction List', similar to Assembly Language) form of programming. Other forms or programming using high-level computer programming languages and in particular sequence control languages such as GRAFCET have also been introduced for PLCs.

One feature that has persisted with PLCs is the fact that all of the programming languages are specific to the PLC manufacturer's model and are non-transferable between different makes of PLC, even if programmed in the same “type” of language. For example the same ladder program from a Modicon PLC could not be used in an Allen Bradley PLC. The introduction of IEC 61131 in 1993 (BSI, 2003) has made some improvements in terms of standardising the notation and most PLC manufacturers have followed the conventions laid down in the standard. The push for standardisation, came from the industry end-users rather than from the PLC manufacturers themselves and has had limited success. The programs are still not portable between different manufacturers PLCs but they are at least understandable by most control engineers. Later PLCs were capable of mixing programming languages within the same overall program. This enables the engineer to use the most appropriate programming language for any given task.

PLC Communications

Many PLC manufacturers developed their own PLC communications systems in the early 1980s. All of these systems were proprietary and were independent of one another which mean that end-users were restricted to using one specific PLC manufacturer's system on an installation. In an attempt to standardise the many different protocols appearing on the market, it was again the customers (end-users) that forced the issue. The main drive came from General Motors by introducing the Manufacturing Automation Protocol (MAP) to provide a standard communications protocol across different PLC and automation vendors. MAP was only partially successful and was succeeded by the development of Fieldbus standards, but again, different manufacturers adopted their own preferred Fieldbus technology.

Latterly, Ethernet, and particularly "Industrial Ethernet", is becoming the most commonly used communications technology for automation systems.

Despite the lack of industry-wide standardisation, the development of automation networks impacted the size and scope of PLC applications. Digital networks enabled remote I/O modules to be connected directly to the PLC so that larger plant could be controlled by a single controller. Later, PLCs themselves could be connected together via the network enabling distributed control systems.

The mid 1970s saw the PLC manufacturers develop the input and output modules leading to the introduction of analogue I/O modules alongside high-speed counters and timers. Pittwood believes that PLCs have now really developed the main I/O capability to meet almost all industrial applications and the distributed intelligence has now moved out to the instrumentation. Complex instruments, such as flow meters and analytical instruments can now pass multiple parameters back to the PLC digitally via the network and do not need to have direct connections to the PLC through its I/O modules. Control is still performed in the PLC but the complex mathematical functions can be carried out in the instrument itself without burdening the PLC's processor.

The Legacy of the PLC

When the PLC was introduced, one of its key attributes was that it was easy to program by the engineer and technician. "Many PLCs were programmed on the plant itself during the commissioning phase and as a result, documentation tends to be poor" remarks Pittwood. Additionally, as time passes, PLC programs have been modified in situ and bear little resemblance to the originally installed program. To make matters worse, it has been known that individual engineers and technicians have their own personal copy of a program. In the event of a problem, the engineer uploads his or her own copy of the program because as far as they are concerned, that program worked the last time they used it.

PLC manufacturers are now developing tools to assist the documentation process and introducing a systems approach to development. Features such as automated change control, backups and security are all potential issues, particularly with remote support and the internet access of control systems. These tools are still in development, but they are not the full answer. "A disciplined approach is required throughout industry and this is a role that the engineering professions are now taking a lead in" says Pittwood (2008b).

Chapter 7 – Conclusion

7.1. Introduction

This chapter reviews the research aims and presents a summary of the subsequent findings and conclusions from the research. A critical appraisal of the research approach is given in terms of the scope and the methodology used in the thesis. The contribution of the thesis to knowledge is considered in terms of the research questions. Finally, possible further work and future research is identified.

7.2. Research Aims and Findings

7.2.1. Sequential control technology Development

Research aim: *To explore how sequential control technology developed in the 20th Century prior to the application of the PLC.*

Developments in sequential control, shown in the patent material investigated in Chapter 4, are clearly incremental. Electro-mechanical systems were initially used to provide sequential and combinatorial control in the guise of relays and drums. The introduction of the electronic timer was of particular significance. Timing was already a feature in rotated drum and cam controlled systems, with the timing achieved by motors and gears. Relays were predominantly used for combinatorial logic (e.g. motor switching control) and the introduction of timing mechanisms followed by electronic timers allowed relay systems to be used in more complex control applications.

In the 1950s computers were beginning to be used for large process control applications but were too expensive for smaller sequence control applications which were largely set in industries in which specialist computer programming skills were not available on the factory floor. As a consequence, relays continued to be used for sequential and combinatorial logic control systems. As the demands for control increased in complexity, relay control systems became large and unwieldy; suffering problems with reliability and requiring re-installation of the wiring when changes were demanded.

More easily alterable sequential control systems were introduced to overcome the re-wiring problems and to reduce production down-time during control changes, but these systems still had limited capabilities. Electro-mechanical systems were developed with interchangeable cams, and later relay circuits were replaced in some instances with electronic logic gates. However, plant shutdowns were still required when changes to the control system were deemed necessary. The introduction of solid-state electronic control systems using detachable "plug-board" based programming techniques was a development that provided a means to configure a logic system independently and remotely from the process under control. This solution was limited to the fixed size of the plug-board matrix and could not configure larger sequential control systems.

Many industrial environments were inhospitable to mainframe computers, which required clean, temperature stable and vibration free locations. The introduction of the minicomputer was an improvement but they were still restricted by the need for specialist programming staff. To alleviate this, programming interfaces were developed so that engineers could configure the control computer without having prior knowledge of the specific programming environment. These developments contributed to the emergence of the PLC as a sequential and combinatorial control technology.

7.2.2. The emergence and subsequent development of the PLC

Research aim: *To explore the emergence and subsequent development of the PLC and to determine what factors influenced the development of the PLC.*

The PLC has its own "creation myth" and many accounts of the emergence of the PLC, largely promoted in the trade press and engineering handbooks, relate to the "Father of the PLC" story, detailing how "Dick" Morley started Modicon and oversaw the invention and introduction of the PLC (see Hendricks, 1996, Moody and Morley, 1999, Morley, 2003, Clare et al., 2006, Ball, 2008, Dunn, 2008). As with many stories of invention, there is more than a grain of truth in it, but it fails to demonstrate the complexity of the emergence of the PLC. Fletcher and Rosseau's (1972) patent, filed by Modicon, is certainly a significant patent outlining the key features of the PLC, although based on a modified DEC PDP/8 minicomputer. The patent is particularly significant because it marks the point in time when the PLC becomes a recognisable and distinct device. However, the patent research of this thesis shows that the PLC resulted from a number of incremental developments and evolved, or rather emerged, from those developments.

The PLC resulted from developments that were driven by solving the "reverse salients" (Hughes, 1998) encountered in applying sequential and combinatorial logic control technologies in industry. PLC Ladder Logic programming notation was developed utilising a notation that was familiar to electrical engineers and overcame the programming induced "reverse salient". Additional "reverse salients" prompted the evolution of the PLC's capabilities. Developments included: expanding I/O, increasing memory, analogue I/O, remote I/O, use of microprocessors and visual displays (revealing multiple ladder rungs).

The application of computers in harsh environments was another problem. Industrial environments could be wet or dusty, subject to high and fluctuating temperatures, and prone to mechanical vibration and electrical interference. The PLC was designed to be a rugged control device that could withstand these adverse environmental conditions and be positioned directly on the machine or plant under control.

To overcome the expense associated with computer technology, the PLC moved away from the general-purpose computer architecture and developed its own logic solving circuitry, as demonstrated by Kiffmeyer's (1974) Programmable Matrix Controller.

The introduction of the microprocessor further enhanced the PLC's potential. Microprocessors improved the processing capability of the PLC and process control technologies alike. Hammond (2007) described the application of the microprocessor with the MPC 80, and was an early proponent of applying the technology for process control applications. Microprocessors could process numbers, enabling the calculation and manipulation of analogue instrument data. A particular problem with microprocessor technologies for the PLC, was related to "bit processing", essential for solving logic constructs. Seipp (1978) and Dummermuth et al (1979) largely circumvented this problem with the "Boolean processor". The PLC was also able to take advantage of the microprocessor to process analogue data and competed with the process control technology. Ultimately, the choice between the use of a process controller and a PLC was guided by the preferences and familiarity of the engineers designing and commissioning control systems, preferences that characterised prevalent practices in different industry sectors.

7.2.3. Technological factors

Research aim: *To investigate technological factors influencing the use and deployment of PLC technologies*

One of the striking aspects of the PLC is that it plays a part in the work of a very limited set of people. The use and application of PLC technology is largely confined to a narrow group of engineers, many of whom have an electrical engineering background. The knowledge held and the practices performed by this distinctive group of engineers shapes a "community of practice", manifest in the UK by the unique ladder logic programming language. Outside the bounds of this "PLC community", little is known about PLC technology, and beyond the engineering disciplines, there is hardly any knowledge of the PLC's existence.

The focus of this research has been manufacturing which was shown to be difficult to categorise and diverse in its approach to applying control technologies. Many industries have sequential control, combined with continuous (process) control requirements, which I have described as "hybrid control" in earlier chapters. The choice of control technology is determined by the dominant control requirements; PLCs are used when sequential and logic control is the main requirement. However, it is simplistic to strictly relate this to specific industry segments. Water treatment, for example, requires continuous control but also has a distribution network commonly spread over a wide geographical area. The control of water distribution networks require the semi-autonomous control of remotely operated pumps and valves; PLCs operating within in a distributed control architecture provide a robust control solution. The Petrochemical industry follows a similar control model for its distribution networks for pumping stations, but uses process control technologies at the refining plant.

Some industries, such as the chemical industry, operate centralised control systems based on process control technologies. However, PLCs are still employed on fixed function plant, supplied by original equipment manufacturers (OEMs), and referred to as "packages". Where PLCs are used, the OEM chooses the control system on economic grounds, keeping costs to a minimum by reducing training and development overheads by using familiar technologies. Bespoke machinery manufacturers and suppliers also prefer to stick to a known technology to reduce development and ongoing maintenance costs. An added advantage is that the customer's in-house engineers and technicians are familiar with these technologies, aiding and improving system acceptance by reducing the training requirements for the customer.

The research revealed that some industries (e.g. the chemical industries) were slow to adopt new technologies and were "conservative", preferring to use (or reuse) proven technologies. In contrast, other industries (e.g. steel) were more progressive. This reflected the attitudes and differences in the adoption of particular "preferred" technologies demonstrating the difference between electrical and process engineers. Where PLCs were used, process engineers showed a preference for the "Function Block Diagram" (FBD) notation because it represented process flows. Electrical engineers, on the other hand, preferred Ladder Logic.

PLC programming methods and language adoption were also influenced by geo-political factors, particularly with industry sectors that differ from one geographic or linguistic setting to another. In English speaking countries, the US & UK for example, Ladder Logic was the preferred programming notation; European engineers, who were not native English speakers, it was reported, preferred the Function Block Diagram notation and the "Instruction List" programming methods.

Geo-political differences also influenced the choice of technology. In the UK larger user organisations tend to use PLC systems manufactured by companies that also provide technical support such as training and technical help. On the other hand the Petrochemical industry ignores national and political boundaries and focuses on a small number of suppliers who, for many petrochemical installations will not have a local presence.

Surprisingly, there is little effective standardisation for PLC technologies. Unlike the personal computer, software and peripherals are not interchangeable. One reason for this *may* be that PLC manufacturers have appeared to steer away from standardisation because it was in their commercial interest to perpetuate a proprietary control architecture. The end users have been the greatest proponents of standardisation, but have been met with limited success. The Fieldbus standard (IEC 61158) failed to gain acceptance because it had multiple, incompatible communications standards. The PLC standard, IEC 61131, has only been partially successful, with general agreement achieved on the notation only; programs are still not portable between different manufacturer's PLCs. The only standard that has been successfully adopted is the analogue signal representation IEC 60381.

7.2.4. Studying a hidden technology

Research aim: *To demonstrate and justify a method of studying a hidden technology*

Edgerton argues that some common place, use-centred technologies are frequently overlooked and become "hidden" from historical accounts. The identification of the PLC as a hidden technology has been confirmed by extensive literature searches which have yielded few useful results. Brief historical accounts have appeared in the trade press and some engineering handbooks (see Clare et al., 2006, Andrew and Hugh, 2007, Ball, 2008, Dunn, 2008), but, to date, no in-depth accounts of the PLC's development have been located. The outcome of this literature search appears to suggest that the PLC does conform to Edgerton's notion of a "use-centred" or "use-based" technology because of its pervasiveness and the lack of academic historical studies.

Studying the development of a "hidden technology" presents the researcher with a number of problems. With a lack of underpinning academic material, alternative sources of literature and other means to obtain evidence pertaining to the development of sequential control and PLC technology, had to be identified and researched. One of the more obvious routes to bridging this gap in knowledge was to obtain first-hand accounts from people who have direct experience in developing or using the technology. However, there is a danger that this leads to an over reliance on the accuracy of the memory of the interviewees. An additional method of studying a hidden technology, which proved particularly fruitful for the PLC, was to review the patents associated with PLC technology.

The research presented in this thesis has demonstrated that the use of patent data, corroborated by first-hand accounts, does provide a viable method of studying a hidden technology. Appendix F shows the correlation between the patent and interview data.

7.2.5. Thesis conclusions

Before the PLC, sequential control technology (relays, wiring, enclosures and so on), was considered an inseparable part of the plant, or at least part of the instrumentation. As soon as the relay control was mimicked by electronic devices (using low currents and new logic levels), it called upon new skills in the manufacture of the sequencing device. These skills and techniques were not commonly available amongst staff running production plant and this opened up the possibility for those with the skills, to make and sell electronic devices to plant installers and operators, especially because the new devices offered advantages (primarily easy programmability) to the plant operator. In particular, it provided a new avenue for existing manufacturers of instrumentation and control equipment, who had already gained skills in electronics, and had access to the buyers in plant manufacturing and operating companies.

The programmability (especially when linked with notations familiar to plant installers and maintainers), allowed a separation between the designers and manufacturers of PLCs, who perhaps knew little about the plant being controlled but who could make good PLCs, and the plant maintainers and operators who knew little about the new technologies of sequencing but did know about the plant and could describe the sequencing and logic operations in ways that could be handled by the PLC. Thus the PLC became an identifiable device with its boundary set by the distinctive technologies of plant and light current electronics and maintained by the distinctive skills associated with the different technologies... The boundary was marked by the programming notation.

Use-based and Disruptive Technologies

The PLC is, in many respects, a "use-based" technology rather than an innovative technology because it substitutes for the electro-mechanical relay and relay-panel and exploited an established notation, features of the dominant sequential control technology of the first half of the 20th Century. The PLC, although a radical development from the relay itself, is designed to emulate the pre-existing relay circuits and complement existing relay and electrical switch systems for industrial control purposes. PLC technology therefore perpetuated the practices of the electrical engineer and technician.

The PLC is also a "sustaining technology" rather than a "disruptive technology" (Christensen, 1997). Christensen uses the Modicon story as an example of a disruptive technology because it was associated only with a narrow application, motor control, rather than wider sequential control applications used in other industries. However, according to Christensen's definition, it could be described as a "radical" and "discontinuous" *sustaining* technology because it perpetuated the technology of the relay through the ladder logic notation.

7.3. Contribution

This thesis has contributed to knowledge by answering the two research questions stated in Chapter 1. Research question one:

How did logic control, sequential control, and PLC technologies, develop in the 20th Century and why?

The research has presented a detailed analysis in the developments of combinatorial logic and sequential control technologies in the 20th Century. Explanations of how and why these technologies developed have been offered, through the study of relevant patents and interview material. Chapter 4 relates the development of sequential control up to the emergence of the PLC and demonstrates that development was incremental and invariably led by technology users. Chapters 5 and 6 detail the emergence and subsequent development of the PLC, which was an incremental development from the application of electronic logic devices and computers. The thesis discusses the technical developments of PLC technology through patents and through the eye-witness accounts of the engineers interviewed in the research.

The research offers new knowledge on the development of automated sequential control, and the Programmable Logic Controller in particular. The development of the PLC is a novel area of research which is concerned with an important and widely used technology.

The research revealed that the PLC was, by Edgerton's (2006) definition, a "hidden technology" prompting the second research question:

How do you study a hidden technology?

The research has confirmed that there is little academic material available detailing the history of logic and sequential control technologies, including the PLC; consequently an alternative approach and methodology had to be used. Patents provided a rich and detailed source of information regarding the developments of these technologies, exposing the requirements for, and incremental improvements of, the technology. The material obtained from the patented information revealed a steady incremental development pattern since later patents consistently cited earlier key patents.

Patent derived data, particularly for the PLC, was supported by the accounts of engineers, familiar with PLC and control technologies from a number of industries. The patents, although they provided examples of innovation, were consistently linked to the existing requirements of the engineers. This could only be revealed by setting the patents against the engineers' accounts (see Appendix F). The interviews also provided insight into how and why the technology was used.

It is proposed that the research method used in this thesis, could also be applied to other technologies, and that this method can be a viable approach to studying the development of a hidden technology.

7.4. Critical Review of the Research Approach

Scope

The research was limited in scope, by time and the availability of suitable interviewees. More interviewees would reveal a wider picture covering a greater number of manufacturing industries. Notably, it was not possible to get input to the research from the automotive industry. However, the spread of industries covered by the interview material was felt to be illustrative. Further research into non-manufacturing examples would have broadened application areas of PLCs, however, the main PLC users were recognised to be in the manufacturing industries.

Interviews with PLC manufacturers were difficult to obtain, with the exception of Schneider Electric through John Pittwood (2008b). Attempts were made to contact other PLC manufacturers but, possibly due to the many takeovers prevalent in the industry, no responses were forthcoming.

There are a great number of published patents under a wide range of similar sounding categories, detailing inventions which could have an influence on the development of PLCs. Inevitably the search was restricted.

Limitations of Methodology

The earliest patent data (pre-PLC) could not be corroborated as a "use-based" technology because it pre-dated the living memory of the interviewees. Unfortunately there was no other supporting literature covering this period. However, an advantage of using patents as a research resource is that they are produced in conjunction with an independent and professional examiner, adding strength to the reliability of the data and information contained within. A further advantage is that they provide a vast resource of accessible information.

Care must be taken when using patent data. Patents are essentially legal (commercial) documents and the patented idea or technology may not have come to fruition. The trade press articles, also used in the research, can be similarly criticised because they are commercially biased documents, which may promote concepts that do not find a market. However, the trade press can provide an insight into current industrial trends and sources. Supported by commercial interests and without a high level of academic rigour the trade press can also reinforce myths about technologies and their value. The "creation myths" associated with the development of the PLC are perpetuated in this medium. (This is not to say that 'academic' historical accounts do not also need to be treated with caution!)

Personal Influences

The author has been associated with the use and application of PLCs for a number of years in the context of a manufacturing environment employing process and discrete control. In the role of an engineering project manager, PLCs have been used to provide control for a wide range of projects from simple relay-replacing upgrades to the implementation of new processing plant. The resulting experience has been gained by implementing control systems that have been dominated by the hybrid manufacturing model; the author has rarely found pure continuous control that hasn't required some form of sequencing.

The impact that this research has had on the author is to realise that the PLC is an "engineer's computer". It appears to be well known within a small group of suppliers and users, particularly manufacturing electrical and control engineers, but is virtually unknown outside this "community of practice". Even amongst the more mechanically-biased designers, the PLC is regarded as a "black-box" that the control engineer has dominion over.

The research has shown that this is a view that reaches beyond the organisational boundary.

The history and development of the PLC is an interesting, and until now, untold story. It is important because PLCs provide the control for many applications requiring automation that pervade the world we live in. The story of the PLCs development is about preserving well known, familiar conventions such as those surrounding relay logic. It is also about adapting and using new technologies for old functions, basically an engineered product for the engineer.

7.5. Further Work and Future Research

Aspects such as security are outside the scope of the thesis and have only been covered at a superficial level. The general concept of "air gap" security (the use of lockable enclosures, limited and controlled access to programming tools, and limited operator switches on the PLC) has contributed to the early security measures taken with PLCs. The advent of networking and the subsequent risks to PLC security from the internet have not been discussed. This includes "cyber security threats" such as "Stuxnet" (Gross and Karlsson, 2012).

This thesis covers the development of sequential control and the PLC up to 1990 and does not include the more recent PLC developments. A further opportunity is therefore to research the development of PLCs from 1990 onwards. Subjects covered in this period include Holonic systems such as that proposed by (Black and Vyatkin, 2010) and (Dai and Vyatkin, 2012).

In addition to this, a study of the effect that standards such as IEC 61131 has had on PLCs and control technologies. As Dai and Vyatkin (2012) point out "... the traditional PLC's programming paradigm of the IEC 61131-3 standard... is becoming a serious bottleneck.".

The research on technological cultures in this thesis could be further expanded. A more in-depth analysis of geo-political influences would reveal different country and industry perspectives. This research should also look into the viewpoints, influences and preferences of different professional organisations.

Finally, a perhaps more obvious theme for future research, is to apply the methodology presented in this thesis to another hidden technology. From a control engineering perspective, reasonable candidates are process control systems and the Supervisory Control and Data Acquisition (SCADA) system. However, there are, no doubt, many other technologies, in many different disciplines, which are still in widespread use, that have also held little academic interest; their histories are waiting to be written.

References

Ackermann, R. (1993) *Programmable logic controllers : book of exercises with solutions : basic level,* Festo Didactic, Esslingen.

Adamski, M. A. and Monteiro, J. L. (1996) 'Declarative specification of system independent logic controller programs', *Proceedings of the IEEE International Symposium on Industrial Electronics, ISIE '96.,*1996. pp. 305-310 vol.1.

Adner, R. (2002) 'When are technologies disruptive? A demand-based view of the emergence of competition', *Strategic Management Journal,* 23 (8), pp. 667-688.

Adsett, D. (1990) 'Logic controllers are easy to use', *Control and Instrumentation,* 22 (1), pp. 42.

Agashe, S. and Agashe, S. D. (2011) 'Programmable Logic Controllers'. In Lipták, B. (Ed.) *Instrument Engineers' Handbook,* CRC Press, pp. 90-100.

Akiyoshi, T. (1969) 'Measurement and Control Equipment for Waterworks', *Fuji Electric Review (Japan),* 15 (5), pp. 184-172.

Alata, M., El-khasawneh, B. and Al-Hourani, S. (2008) 'Mechatronics in pharmaceutical engineering: Advanced control to upgrade a tablet coating machine', *Mechatronics and Its Applications, 2008. ISMA 2008. 5th International Symposium on*2008. pp. 1-6.

Alford, C. (1973) 'Design Aspects of Computer Control in Discrete Manufacturing', *Manufacturing Technology, IEEE Transactions on,* 2 (2), pp. 26-36.

Allen-Bradley (1972) 'Programmable Matrix Controller', *Scientific American,* Sept 1972, p. 129.

Allen, D. (2004) 'Will workers pay to take forty winks?', *The Times,* London, England, 21 October 2004, Feature Articles (aka Opinion), p. 10.

Andreiev, N. (1972) 'Programmable Logic Controllers - An Update', *Control Engineering,* Sept 1972, pp. 45-47.

Andrew, S. and Hugh, J. (2007) 'Control with Embedded Computers and Programmable Logic Controllers'. *Mechatronic System Control, Logic, and Data Acquisition,* CRC Press, pp. 1-13.

Anon (1947) 'Factory Electrical Equipment', *The Times,* London, England, 01 Jan 1947, p. 24.

Anon (1952), Aston Electrical Products Pty *Improved multiple sequence electric control means for use with resistance welders, electrical circuits and process control apparatus*, GB672381, United Kingdom

Anon (1962) *Norbit Sub-Assemblies YL6000 Series*, Mullard Equipment Limited. Available from: http://www.electrojumble.org/DATA/Norbits.pdf [Accessed 23/02 2012].

Anon (1978) 'Government lab works on tomorrow's factory', *New Scientist*, 80 (1127), pp. 359.

Anon (1985) 'A pioneer emerges', *The Times*, London, 13 May 1985, p. 22.

Anon (1994) 'Inside the Channel Tunnel', *Railway Age*, 01 August, p. 42(2).

Anon (1996) 'Riding high is safer than driving a car', *The Times*, London, England, 07 August 1996, Feature Articles (aka Opinion), p. 9.

Anon (2007) *GEM 80 Brochure*, Converteam. Available from: http://www.gem80.com/documentation/GEM80brochure.pdf [Accessed 13 Feb 2012].

Appel, H. J. (1941), Houdry Process Corp, *Successive switching arrangement*, US2250453, United States

Ashley, J. R. and Pugh, A. (1968) 'LOGICAL DESIGN OF CONTROL SYSTEMS FOR SEQUENTIAL MECHANISMS', *International Journal of Production Research*, 6 (4), pp. 291.

Aspray, W. (2011) 'The History of Information Science and Other Traditional Information Domains: Models for Future Research', *Libraries & the Cultural Record*, 46 (2), pp. 230-248.

Aspray, W., Bromley, A. G., Campbell-Kelly, M., Ceruzzi, P. E. and Williams, M. R. (1990) *Computing before computers*, Iowa State University Press, Ames.

Aylen, J. (2004) 'Megabytes for metals: development of computer applications in the iron and steel industry', *Ironmaking and Steelmaking*, 31 (6), pp. 465-478.

Aylen, J. (2010) 'Promoting the Prosaic: The Case for Process-Control Computers', *Annals of the History of Computing, IEEE*, 32 (3), pp. 96-95.

Aylen, J. (2012) 'Bloodhound on my Trail: Building the Ferranti Argus Process Control Computer', *International Journal for the History of Engineering & Technology,* 82 (1), pp. 1-36.

Bairstow, J. N. (1969) 'Machine Control: Solid-State Logic Challenges Relays', *Control Engineering,* 16 (3), pp. 53.

Ball, K. (2008) In *Control Engineering,* Vol. 55 Reed Business Information, pp. 66-70.

Barrow, A. (1976) 'Programmable controllers', *Electronics and Power,* 22 (5), pp. 310-311.

Batten, G. L. (1994) *Programmable controllers : hardware, software, and applications,* (2nd ed. Edn), McGraw-Hill, New York ; London.

Bell, G., Cady, R., McFarland, H., Delagi, B., O'Laughlin, J., Noonan, R. and Wulf, W. (1970) In *Proceedings of the May 5-7, 1970, spring joint computer conference*ACM, Atlantic City, New Jersey.

Bennett, S. (1979) *A history of control engineering, 1800-1930,* Peregrinus on behalf of the Institution of Electrical Engineers, 1986, London.

Bennett, S. (1988) *Real-time computer control,* Prentice Hall, New York ; London.

Bennett, S. (1991) '"The Industrial Instrument-Master of Industry, Servant of Management": Automatic Control in the Process Industries, 1900-1940', *Technology and Culture,* 32 (1), pp. 69-81.

Bennett, S. (1993a) 'Development of the PID controller', *Control Systems, IEEE,* 13 (6), pp. 58-62, 64-5.

Bennett, S. (1993b) *A History of Control Engineering 1930 - 1955,* Peter Peregrinus Ltd.

Bennett, S. (1994) *Real-time computer control : an introduction,* (2nd ed. Edn), Prentice Hall, New York ; London.

Bennett, S. (1996) 'A brief history of automatic control', *Control Systems Magazine, IEEE,* 16 (3), pp. 17-25.

Bennett, S. (2002) 'OTTO MAYR: Contributions to the History of Feedback Control', *IEEE Control Systems,* 22 (2), pp. 29.

Bergen, A. (1962) 'A note on Tsypkin's locus', *Automatic Control, IRE Transactions on,* 7 (3), pp. 78-80.

Bernstein, D. S. (1999) 'Enhancing undergraduate control education', *Control Systems Magazine, IEEE,* 19 (5), pp. 40-43.

Bessant, J. and Haywood, B. (1988) 'Islands, archipelagoes and continents: Progress on the road to computer-integrated manufacturing', *Research Policy,* 17 (6), pp. 349-362.

Bhuyan, M. (2006) 'Controllers and Indicators'. *Measurement and Control in Food Processing,* CRC Press, pp. 243-302.

Bijker, W. E. (1995) *Of bicycles, bakelites, and bulbs : toward a theory of sociotechnical change,* MIT Press, Cambridge, Mass. ; London.

Bijker, W. E., Hughes, T. P. and Pinch, T. J. (1987) *The social construction of technological systems : new directions in the sociology and history of technology : Workshop : Papers,* MIT Press, Cambridge, Mass.

Bijker, W. E. and Law, J. (1992) *Shaping technology/building society : studies in sociotechnical change,* MIT Press, Cambridge, Mass. ; London.

Bissell, C. (2009) 'A history of automatic control', *Springer Handbook of Automation,* pp. 53-69.

Bissell, C. C. (1991) 'Secondary sources for the history of control engineering: an annotated bibliography', *International Journal of Control,* 54 (3), pp. 517-528.

Bissell, C. C. (1996) 'Bibliography of secondary sources for the history of control engineering: Update 1995', *International Journal of Control,* 63 (5), pp. 995-1002.

Bissell, C. C. (1999) 'Control education: time for radical change?', *Control Systems Magazine, IEEE,* 19 (5), pp. 44-49.

Bissell, C. C. (2001) 'The Role of A. A. Andronov in the Development of Russian Control Engineering', *Automation and Remote Control,* 62 (6), pp. 863-874.

Bivens, M. E. (1949), GEN ELECTRIC, *Electric timer,* US2473237, United States

Black, G. and Vyatkin, V. (2010) 'Intelligent Component-Based Automation of Baggage Handling Systems With IEC 61499', *Automation Science and Engineering, IEEE Transactions on,* 7 (2), pp. 337-351.

Bollinger, J. G. and Duffie, N. A. (1988) *Computer control of machines and processes,* Addison-Wesley, Reading, Mass.

Bolton, W. (2006) *Programmable logic controllers,* (4th ed. Edn), Amsterdam ; London : Elsevier, 2003 (2004 [printing]).

Bradley, R. (2006) 'Progressive PLCs', *Works Management*, 59 (9), pp. 35-39.

Brandl, D. (2007) *Design patterns for flexible manufacturing*, ISA. Available from: http://libezproxy.open.ac.uk/login?url=http://www.knovel.com/knovel2/Toc.jsp?BookID=1821 [Accessed 11 January 2010].

Brown, F. (1984) 'A giant step for automation?', *The Times*, London, 02 Oct 1984, p. 18.

Brown, R. A., Hu, S. C. and Struger, O. J. (1981), Allen-Bradley Company (Milwaukee, WI), *Mini-programmable controller*, US4282584, United States

Bruce, P. (2008) *Personal Interview*. Mark Walker (author), 18 Jan 2008.

Bryan, L. A. and Bryan, E. A. (1997) *Programmable Controllers*, (2nd Edn), Industrial Text Company, Marietta, Georgia, United States.

BSI (1982) In *Analogue signals for process control systems. Specification for direct current signals* Vol. BS 5863-1:1984, IEC 60381-1:1982 BSI.

BSI (2003a) In *Programmable Controllers*, Vol. BS EN 61131-1:2003 BSI, pp. 8.

BSI (2003b) In *BS EN 61131-3:2003 Programmable controllers - Part 3: Programming Languages*, Vol. BS EN 61131-3:2003 BSI.

BSI (2004) In *IEC 61158-1:2004 Digital data communications for measurement and control - Fieldbus for use in industrial control systems*, Vol. IEC 61158-1:2004 BSI.

Burkett, B. G. and Henry, R. W. (1976), Texas Instruments Incorporated (Dallas, TX), *Programmable logic controller with push down stack*, US3953834, United States

Burkett, B. G. and Henry, R. W. (1977), Texas Instruments Incorporated, *Variable module memory*, US4030080, United States

Campbell-Kelly, M. (2003) *From airline reservations to Sonic the Hedgehog : A history of the software industry*, MIT Press, Cambridge, Mass.

Campbell-Kelly, M. (2007) 'The history of the history of software', *IEEE Annals of the History of Computing*, 29 (4), pp. 40-51.

Campbell-Kelly, M. (2009) 'Origin of Computing. (Cover story)', *Scientific American*, 301 (3), pp. 62-69.

Campbell-Kelly, M. and Aspray, W. (2004) *Computer : a history of the information machine,* (2nd ed. Edn), Westview, Boulder, Colo. ; Oxford.

Chace, D. E., Melvin Jr., J. A., Ricketts Jr., A. W., Steinberger, E. P. and Symmes, D. T. (1972), Digital Equipment Corporation (Maynard, MA), *Analyzer for Sequencer Controller,* US3701113, United States

Christensen, C. M. (1997) *The innovator's dilemma : when new technologies cause great firms to fail,* Harvard Business School Press, Boston, Mass.

Clare, W. N., Kaplan, G. T., Sadlon, D. R., Wiktorowicz, A. C. and Gilbert, R. A. (1995) 'PLC - Programmable Logic Controllers'. In Liptak, B. G. (Ed.) *Instrument engineers' handbook: Process Control,* (3rd Edn), Oxford : Butterworth-Heinemann, pp. 722-723.

Clare, W. N., Kaplan, G. T., Sadlon, D. R., Wiktorowicz, A. C., Gilbert, R. A. and Wendt, C. W. (2005) 'PLCs: Programmable Logic Controllers'. In Liptak, B. G. (Ed.) *Instrument Engineers' Handbook: Process Control and Optimization.*

Clare, W. N., Kaplan, G. T., Sadlon, D. R., Wiktorowicz, A. C., Gilbert, R. A. and Wendt, C. W. (2006) 'PLCs: Programmable Logic Controllers'. In Liptak, B. G. (Ed.) *Instrument Engineers' Handbook: Process Control and Optimization.*

Clayton, B. B., Fagen, R. E., Marshman, W. W. and Wolbrecht, H. M. (1968), CONTROL DATA CORP, *Remote calculator,* US3380031, United States

Collins, P. D., Hull, F. M. and Hage, J. (1996) 'Profiles of leaders, followers, and laggards in programmable automation adoption', *IEEE Transactions on Engineering Management,* 43 (3), pp. 285-296.

Constantine, L. (2011) 'Crossing the Line: Terrorism in Cyberspace and Targets in Real-Space', *Cyberworlds (CW), 2011 International Conference on,* 4-6 Oct. 2011, 2011. pp. 1-4.

Cox, T. C. (1965), IBM, *Computer control apparatus,* US3226684, United States

Crispin, A. J. (1990) *Programmable logic controllers and their engineering applications,* McGraw-Hill, London.

Cutler-Hammer (1926), Cutler-Hammer MFG Co, *Improvements in or relating to electric motor controllers,* GB251559, United Kingdom

Dai, W. and Vyatkin, V. (2012) 'Redesign Distributed PLC Control Systems Using IEC 61499 Function Blocks', *IEEE Transactions on Automation Science and Engineering,* 9 (2), pp. 390-401.

Daley, A. (2010) *Interview 21/07/2010.* Mark Walker (author), 21 July.

Danneels, E. (2004) 'Disruptive technology reconsidered: A critique and research agenda', *Journal of Product Innovation Management,* 21 (4), pp. 246-258.

Datamonitor (2011a) *Control Systems Industry Profile: Global,* Datamonitor Plc. Available from: http://libezproxy.open.ac.uk/login?url=http://search.ebscohost.com/login.aspx?direct=true&db=bth&AN=65433973&site=eds-live&scope=site [Accessed 29 March 2012].

Datamonitor (2011b) *Control Systems Industry Profile: United Kingdom,* Datamonitor Plc. Available from: http://libezproxy.open.ac.uk/login?url=http://search.ebscohost.com/login.aspx?direct=true&db=bth&AN=65433970&site=eds-live&scope=site [Accessed 19 April 2012].

David, R. (1995) 'Grafcet: a powerful tool for specification of logic controllers', *Control Systems Technology, IEEE Transactions on,* 3 (3), pp. 253-268.

Davis, R. (1992) 'PLC vs DCS in process control--The distinction is fading', *Textile, Fiber and Film Industry Technical Conference, 1992., IEEE 1992 Annual,* Piscataway, NJ, USA1992. Publ by IEEE, p. 5.

Dean, K. J. (1973) 'Trends in semiconductor digital circuits', *Radio and Electronic Engineer,* 43 (1.2), pp. 67-74.

Derksen, R., Wouden, H. v. d. and Heath, P. (2007) *Testing the Heathrow Terminal 5 Baggage Handling System – Before It Is Built* Available from: http://newsweaver.ie/qualtech/e_article000776781.cfm?x=b11,0,w [Accessed 17 April 2012].

Dummermuth, E. (1976), Allen-Bradley Company, *Programmable Logic Controller,* US3942158, United States

Dummermuth, E. (2002) *A Bit Twiddlers History of the PLC,* Putman Media Inc. Available from: http://www.ab.com/abjournal/nov2002/pdfs/ABJ0211.pdf [Accessed 27 May 2006].

Dummermuth, E., Grudowski, R. A., Grants, V. and Schmidt, O. (1979), Allen-Bradley Company (Milwaukee, WI), *Digital control system with Boolean processor,* US4165534, United States

Dunn, A. (2008) *The father of invention: Dick Morley looks back on the 40th anniversary of the PLC.* Available from: http://www.automationmag.com/features/the-father-of-invention-dick-morley-looks-back-on-the-40th-anniversary-of-the-plc.html [Accessed September 12, 2008].

Edgerton, D. (1999) 'From Innovation to Use: Ten Eclectic Theses on the historiography of Technology', *History and Technology,* 16 (2), pp. 111-136.

Edgerton, D. (2006) *The shock of the old : technology and global history since 1900,* Profile, London.

EPO (2011) *European Classification (ECLA),* Eurppean Patenty Office. Available from: http://worldwide.espacenet.com/help?topic=ecla&locale=en_GB&method=handleHelpTopic [Accessed 09 Dec 2011].

Erickson, K. T. (1996) 'Programmable logic controllers', *Potentials, IEEE,* 15 (1), pp. 14-17.

Erickson, K. T. (2001) 'Factory automation: a controls course for every university', *Proceedings of the American Control Conference,* Arlington, VA 2001. pp. 1167-1172 vol.2.

Espacenet (2012a) *List of Citing Documents: US3686639 (A) — 1972-08-22* Available from: http://worldwide.espacenet.com/publicationDetails/citingDocuments?page=0&KC=A&ND=3&date=19720822&NR=3686639A&DB=EPODOC&locale=en_EP&CC=US&FT=D [Accessed 15 June 2012].

Espacenet (2012b) *List of Citing Documents: US3942158 (A) — 1976-03-02* Available from: http://worldwide.espacenet.com/publicationDetails/citingDocuments?CC=US&NR=3942158A&KC=A&FT=D&ND=3&date=19760302&DB=EPODOC&locale=en_EP [Accessed 07 June 2012].

Fabian, M. and Hellgren, A. (1998) 'PLC-based implementation of supervisory control for discrete event systems', *Proceedings of the 37th IEEE Conference on Decision and Control,* 1998. pp. 3305-3310 vol.3.

Fauchier, J. F., Seipp, W. H. and Whiteside, S. E. (1980), Gulf & Western Industries, Inc. (New York, NY), *System for converting analog signals to multiplexed digital data,* US4188617, United States

Fauci, J. L. (1997) 'PLC or DCS: selection and trends', *ISA Transactions,* 36 (1), pp. 21-28.

Felser, M. and Sauter, T. (2002) 'The fieldbus war: history or short break between battles?', *Factory Communication Systems, 2002. 4th IEEE International Workshop on*2002. pp. 73-80.

Firth, T. W. S. and kennedy, J. N. C. (1914), *Improvements for Signalling, Regulating and Controlling Street Traffic and in the Method of its Application for such Traffic.*, GB191317337, United Kingdom

Fletcher, W. E. and Rosseau, L. B. (1972), Modicon Corporation, *Digital Computer - Industrial Controller System and Apparatus*, US3686639, United States

Forger, G. (1998) *World's largest integrated baggage handling system opens in Hong Kong*, Nexis UK. Available from: http://www.lexisnexis.com/uk/nexis/results/docview/docview.do?docLinkInd=true&risb=21_T14487224905&format=GNBFI&sort=BOOLEAN&startDocNo=1&resultsUrlKey=29_T14487224909&cisb=22_T14487224908&treeMax=true&treeWidth=0&csi=169864&docNo=2 [Accessed 17 May 2008].

Gelb, A. and Henrikson, L. (1963) 'On the analytical application of Tsypkin's method', *IEEE Transactions on Automatic Control,* 8 (1), pp. 69-70.

Geng, H. (2004) *Manufacturing engineering handbook*, McGraw-Hill, New York ; London.

Google (2011) *Google Translate*. Available from: http://translate.google.com/#auto|en|Einrichtung%20Zur%20Durchfuehrung%20Boolescher%20Verknuepfungen [Accessed 20 Oct 2011].

Gordon, T. T. and Cookfair, A. S. (2000) *Patent Fundamentals for Scientists and Engineers* CRC Press.

Greenberg, M. P., Fletcher, W. E. and Morley, R. E. (1973), Modicon Corporation (Bedford, MA), *Digital Computer*, US3740722, United States

Grogan, A. (2012) 'The golden age of amusement parks', *Engineering and Technology Magazine,* 20 February 2012.

Gross, M. J. and Karlsson, J. F. (2012) 'A Declaration of Cyber-War', *Vanity Fair,* 53 (No. 4), pp. 152.

Halsall, J. R. and Murrell, A. P. C. (1973), *Control Systems*, US3731279, United States

Hamano, G. (1974), Matsushita Electrical Industrial Co., Ltd. (Osaka, JA), *Programmable Logic Controller*, United States

Hammond, P. H. (1967) 'Living control systems', *Electronics and Power,* 13 (9), pp. 338-342.

Hammond, P. H. (1980a) 'IEE Computing & Control Division: Chairman's address. The transfer of computing and control technology into manufacturing industry', *Physical Science, Measurement and Instrumentation, Management and Education - Reviews, IEE Proceedings A*, 127 (1), pp. 27-37.

Hammond, P. H. (1980b) 'The transfer of computing and control technology into manufacturing industry', *IEE Proceedings*, 127, Pt A (No. 1), pp. 27-37.

Hammond, P. H. (2007) *Personal Interview.* Mark Walker (author), InstMC, London. dated 08 Jan 2007.

Hammond, P. H. and King, P. J. (1980) 'A prospect for industrial control', *Electronics and Power*, 26 (1), pp. 38-45.

Hayashibe, S. (1972) 'Fuji Programmable Sequencer "PROGIC"', *Fuji Electric Review (Japan)*, 18 (1), pp. 44-46.

Hendricks, H. (1996) *The History of the PLC as told to Howard Hendricks by Dick Morley* Available from: http://www.barn.org/FILES/historyofplc.html [Accessed 06 Oct 2011].

Henry, D. E., Murrell, T. A., Schilt, M. H. and Weaver, A. C. (1980), Struthers-Dunn, Inc. (Pitman, NJ), *Process control system that controls its outputs according to the results of successive analysis of the vertical input columns of a hypothetical ladder diagram*, US4217658, United States

Hill, L. W., Stoodley III, T. J. and Malcolm, R. (1981), Modicon Div. Gould Inc. (Andover, MA), *Programmable controller*, US4292666, United States

Hills, W. B. (1949), GEN ELECTRIC, *High-speed sequence control*, US2492749, United States

Hinsken, G. (1988a) *Computer With Bit Processor And Word Processor* EPO Patent Translate. Available from: http://translationportal.epo.org/emtp/translate/?ACTION=description-retrieval&COUNTRY=DE&FORMAT=docdb&KIND=A1&LOCALE=en_EP&NUMBER=3720006&OPS=ops.epo.org&ENGINE=google&SRCLANG=de&TRGLANG=en [Accessed 07 June 2012].

Hinsken, G. (1988b), Siemens AG, *Computer With Bit Processor And Word Processor* DE3720006, Germany

Hitomi, K. (1996) *Manufacturing Systems Engineering: A Unified Approach to Manufacturing Technology, Production Management and Industrial Economics*, CRC.

Hodgkins, C. H. (1926), Westinghouse Electric and Manufacturing Company, *Sequence switch*, US1606785, United States

Holecek, A. R. (1976), Babcock & Wilcox Company, *Apparatus for Processing Data in the Form of a Boolean Expression*, GB1444791, United Kingdom

Honchell, J. W. and Robertson, T. L. (1996) 'Is the role of applied programming languages changing?', *FIE '96. 26th Annual Conference., Proceedings of Frontiers in Education Conference,* 1996. pp. 791-794 vol.2.

Hughes, T. P. (1983) *Networks of power : electrification in Western society, 1880-1930,* Johns Hopkins University Press, Baltimore.

Hughes, T. P. (1998) 'Designing, Developing, and Reforming Systems', *Daedalus,* 127 (4), pp. 215-232.

Huiqun, Y. and Chunmei, X. (2010) 'Design and Implementation of the Key Technologies for Baggage Handling Control System', *Measuring Technology and Mechatronics Automation (ICMTMA), 2010 International Conference on,* 13-14 March 2010, 2010. pp. 958-961.

Huuck, R. (2005) 'Semantics and Analysis of Instruction List Programs', *Electronic Notes in Theoretical Computer Science,* 115, pp. 3-18.

James, S. E. (1981) 'Evolution of Real-Time Computer Systems for Manned Spaceflight', *IBM Journal of Research and Development,* 25 (5), pp. 417-428.

James, T. (2006) 'All locked up [Panama Canal]', *Computing & Control Engineering Journal,* 17 (4), pp. 16-21.

Jenkins, H. C. (1931), Henry Charles Jenkins, *Improvements in the automatic electric control of cyclic operations in mechanisms andother equipment*, GB348659, United Kingdom

Jiyunichi, S. and Yoshihiko, O. (1982), TOSHIBA MACHINE CO LTD, *Operating Method of Ladder Circuit Input Part on Column Cycle System* JP57000705, Japan

Johnson, C. (2008) 'Analogue Apollo: Cybernetics and the Space Age', *Paragraph,* 31 (3), pp. 304-326.

Jones, A. and Bissell, C. (2011) 'The social construction of educational technology through the use of authentic software tools', *Research in Learning Technology,* 19 (3), pp. 285-297.

Keister, W. (1949) 'The Logic of Relay Circuits', *American Institute of Electrical Engineers, Transactions of the,* 68 (1), pp. 571-576.

Keller, A. C. (1962) 'Relays and Switches', *Proceedings of the IRE*, 50 (5), pp. 932-934.

Kennedy, J. S. (1924), Kennedy, James S., *Automatically-controlled water-gas set*, US1498174, United States

Kennedy, J. S. (1927), Bartlett Hayward Company, *Automatic sequence control for valves*, US1634327, United States

Kiffmeyer, W. W. (1974), Allen-Bradley Company, *Programmable Matrix Controller*, US3810118, United States

Kiffmeyer, W. W. and Baron, L. G. (1974), Allen-Bradley Company (Milwaukee, WI), *Programmable Controller Expansion Circuit*, US3806877, United States

Kingsbury, J. E. (1910), Kingsbury, John Edward, *An Automatic Sequence Switching Device and its Control*, GB190920840, United Kingdom

Kintner, P. M. (1980), Eaton Corporation (Cleveland, OH), *Integrated circuit controller programmable with unidirectional-logic instructions representative of sequential wire nodes and circuit elements of a ladder diagram*, US4227247, United States

Koetsier, T. (2001) 'On the prehistory of programmable machines: musical automata, looms, calculators', *Mechanism and Machine Theory*, 36 (5), pp. 589-603.

Kowalewski, S. (2002) 'Hybrid systems in process control: Challenges, methods and limits', *AICHE Symposium Series*, New York 2002. American Institute of Chemical Engineers, pp. 121-135.

Koyanagi, H., Takahashi, M., Yamauchi, N., Matsumura, M. and Morimoto, K. (1976), Mitsubishi Denki Kabushiki Kaisha (Tokyo, JA),Nissan Motor Company, Limited (BOTH OF, JA), *Digital logical sequence controller*, US3944987, United States

Krapf, D. and Garschagen, W. (2005) 'Big apple boats: the first of three new ferries for New York City enters service', *Workboat*, 01 April, p. 42(4).

Lapidus, G. (1971) 'Programmable Logic Controllers - Painless Programming to Replace the Relay Bank', *Control Engineering*, 18 (4), pp. 49-60.

Lapidus, G. (1972) 'Programmable Logic Controllers - User Experience', *Control Engineering*, 19 (4), pp. 35-39.

Law, J. (1984) 'International Workshop on New Developments in the Social Studies of Technology Twente University of Technology Enschede, The Netherlands', *4S Review*, 2 (4), pp. 9-13.

Lawson, D. A., Peterson, R. W. and Stockert, A. A. (1970), Bell Telephone Labor Inc, *Process Control System Including Hardware Element Status Map In Memory*, US3495220, United States

Leeming, D. (2008) *Personal Interview.* Mark Walker (author), 30 May 08.

Lewis, B. (2004) 'Developments in industrial control languages - where next?', *Open Control Systems - The Importance of Industrial Standards* 2004. pp. 3/1-316.

Linton, J. D. (2009) 'De-babelizing the language of innovation', *Technovation,* 29 (11), pp. 729-737.

Lucas Jr, H. C. and Goh, J. M. (2009) 'Disruptive technology: How Kodak missed the digital photography revolution', *The Journal of Strategic Information Systems,* 18 (1), pp. 46-55.

Macarthur, J. G. (1967), Bunker, Ramo, *Monitoring apparatus,* US3324458, United States

MacKenzie, D. A. and Wajcman, J. (1999) *The social shaping of technology,* (2nd ed. Edn), Open University Press, Milton Keynes, [Eng.] ; Philadelphia, Pa.

Macoy, E. W. (1966) 'Experience with Semiconductor Logic on Package Fabricating Machines', *Industry and General Applications, IEEE Transactions on,* IGA-2 (3), pp. 182-190.

Markides, C. (2006) 'Disruptive Innovation: In Need of Better Theory*', *Journal of Product Innovation Management,* 23 (1), pp. 19-25.

Masahiro, I. (1983), Sharp KK, *Sequence Controller,* JP58002906, Japan

Mayr, O. (1970) *The origins of feedback control,* MIT Press, Cambridge, Mass. ; London.

McBerty, F. R. (1910), Frank R McBerty, *Automatic Selective Switching Apparatus suitable for use in Telephone Exchanges,* GB190920839, United Kingdom

McCarthy, I. (1995) 'Manufacturing classification: Lessons from organizational systematics and biological taxonomy', *Integrated Manufacturing Systems,* 6 (6), pp. 37 - 48.

McCarthy, I. and Ridgway, K. (2000) 'Cladistics: a taxonomy for manufacturing organizations', *Integrated Manufacturing Systems,* 11 (1), pp. 16 - 29.

McHutchison, W. (1966) 'Pneumatic Control Design', *Nature,* 210 (5038), pp. 771.

McKay, M. (2012) 'Best practices in automation security', *Cement Industry Technical Conference, 2012 IEEE-IAS/PCA 53rd*, 14-17 May 2012, 2012. pp. 1-15.

McKay, S. (2010) *The secret life of Bletchley Park : the history of the wartime codebreaking centre and the men and women who were there,* Aurum, London.

Metcalfe, T. (2012) *Personal email.* Mark Walker (author), Basingstoke. dated 16 April 2012.

Michel, B., Vincent, F., Jean-Claude, C. and Guy, T. (1978a) *Automate Programmable Pour La Commande Du Deroulement D'un Cycle De Fonctionnement D'installation*, EPO Patent Translate Available from: http://translationportal.epo.org/emtp/translate/?ACTION=description-retrieval&COUNTRY=FR&FORMAT=docdb&KIND=A1&LOCALE=en_EP&NUMBER=2361689&OPS=ops.epo.org&ENGINE=google&SRCLANG=fr&TRGLANG=en [Accessed 07 June 2012].

Michel, B., Vincent, F., Jean-Claude, C. and Guy, T. (1978b), ANVAR [FR], *Automate Programmable Pour La Commande Du Deroulement D'un Cycle De Fonctionnement D'installation,* FR2361689,

Miles, L. P. (1926), *Improvements in apparatus for controlling road traffic*, GB262271, United Kingdom

Millis Jr., H. L. (1965), Millis Jr., Hugh L., *General purpose parallel sequencing computer*, US3204087, United States

Mindell, D. A. (2002) *Between human and machine: feedback, control, and computing before cybernetics,* The John Hopkins University Press.

Mindell, D. A. (2008) *Digital Apollo: human and machine in spaceflight,* Mit Pr.

Molina, F. J., Barbancho, J., Leon, C., Molina, A. and Gomez, A. (2007) 'Using industrial standards on PLC programming learning', *Control & Automation, 2007. MED '07. Mediterranean Conference on*, 27-29 June 2007, 2007. pp. 1-6.

Monk, J. (1980) 'Developments in microelectronics technology and the economics of the semiconductor industry', *International Journal of Social Economics,* 7 (1), pp. 13-23.

Monk, J. (2005) In *International Conference on Engineering Education and Research* Tainan, Taiwan.

Moody, P. E. and Morley, R. E. (1999) *The technology machine : how manufacturing will work in the year 2020,* Free Press, New York, N.Y. ; [Great Britain].

Morley, R. (1973), Modicon Corporation (Bedford, MA), *DIGITAL COMPUTER*, US3761893, United States

Morley, R. E. (1979), Gould Inc, Modicon Div (Rolling Meadows, IL), *Digital input/output system and method,* US4162536, United States

Morley, R. E. (2003) *Out of the Barn,* The Instrumentation, Systems and Auromation Society.

Morley, R. E., Bromberg, M. A. and Taylor, W. A. (1980), Andover Controls Corporation (Andover, MA), *Programmable sequence controller with drum emulation and improved power-down power-up circuitry*, US4213174, United States

Morris, A. (2007a) *email.* Mark Walker (author), 04 Oct 2007.

Morris, A. (2007b) *Personal Interview.* Mark Walker (author), 30 Nov 07.

Nakao, H., Naruse, K., Hasegawa, K., Kawade, S., Tokura, Y. and Matsuno, K. (1974), Toyoda Koki Kabushiki Kaisha (Aichi-ken, JA), *General purpose sequence controller*, US3832696, United States

Nakao, H., Naruse, K., Hasegawa, K., Kawade, S., Tokura, Y. and Matsuno, K. (1977), Toyoda Koki Kabushiki Kaisha (JA), *General purpose sequence controller*, US4025902, United States

Nakao, H., Tokura, Y., Toshihiko, Y. and Matsuura, K. (1976), Toyoda Koki Kabushiki Kaisha,Toyoda Jidosha Kogyo Kabushiki Kaisha, *Programmable sequence controller*, US3996565, United States

Neuman, J. J. (1933), *Automatic sequence control means*, US1939183, United States

Noyce, R. N. and Hoff, M. E. (1981) 'A History of Microprocessor Development at Intel', *Micro, IEEE,* 1 (1), pp. 8-21.

Otis, E. J. (1958) 'Industrial Digital Systems', *Industrial Electronics, IRE Transactions on,* PGIE-6, pp. 73-79.

Otsuka, K., Nakagawa, T. and Shimokawa, Y. (1974), Tokyo Shibaura Electric Co., Ltd. (Kawasaki-shi, JA), *Sequence Controller*, US3806714, United States

Parker, G. A. (1969) 'Fluidic Systems', *Nature,* 221 (5178), pp. 393.

Pavlovic, O. and Ehrich, H. D. (2010) 'Model Checking PLC Software Written in Function Block Diagram', *Software Testing, Verification and Validation (ICST), 2010 Third International Conference on*, 6-10 April 2010, 2010. pp. 439-448.

Payne, J. (2007) 'PLCs - Paste Present and Future. (Cover story)', *Industrial Maintenance & Plant Operation*, 68 (1), pp. 33-37.

Pearse, G. (1995) 'Fire underground.; fires in road and rail traffic tunnels; fire detection and fire fighting equipment; hydraulic fluids', *World Tunnelling and Subsurface Excavation*, 01 March, pp. 73-76.

Pierz, R. J. (1961), Pierz, Richard J., *Programming and sequence control for fuel burners*, US3008517, United States

Pinch, T. J. and Bijker, W. E. (1984) 'The Social Construction of Facts and Artefacts: or How the Sociology of Science and the Sociology of Technology might Benefit Each Other', *Social Studies of Science (Sage Publications, Ltd.)*, 14 (3), pp. 399-441.

Pittwood, J. (2008a) Presentation "*Control Freaks and how they do it - Programmable Logic Controllers*" Newcomen Society, London, 09 Jan 2008.

Pittwood, J. (2008b) *Personal Interview*. Mark Walker (author), 16 Oct 2008.

Porter, K., Little, D., Peck, M. and Rollins, R. (1999) 'Manufacturing classifications: relationships with production control systems', *Integrated Manufacturing Systems*, 10 (4), pp. 189-198.

Prell, C. (2009) 'Rethinking the Social Construction of Technology Through 'Following the Actors': A Reappraisal of Technological Frames', *Sociological Research Online*, 14 (2), pp. 4.

Provanzano, S. R., Aldrich, W. H., D'angelo, R. A., Drottar, E. P., Finnegan Jr., J. J., Heom, J., Hill, L. W., Malcolm, R. D., Nollet, M. C., Perlman, B. S., Tressler, M. B., Van Schalkwyk, J. E. and Webb, K. N. (1984), Gould Inc. (Rolling Meadows, IL), *Programmable controller*, US4484303, United States

Provanzano, S. R., Aldrich, W. H., D'angelo, R. A., Drottar, E. P., Finnegan Jr., J. J., Heom, J., Hill, L. W., Malcolm, R. D., Nollet, M. C., Perlman, B. S., Tressler, M. B., Vanschalkwyk, J. E. and Webb, K. N. (1981), Gould Inc. (Rolling Meadows, IL), *Programmable controller*, US4307447, United States

Rahman, M. L. and Woodward, M. E. (1975) 'Generalised model for programmable sequence controllers', *Electronics Letters*, 11 (4), pp. 86-87.

Rameback, C. (2003) 'Process automation systems-history and future', *Proceedings. ETFA '03. IEEE Conference Emerging Technologies and Factory Automation* 2003. pp. 3-4 vol.1.

Rao, N., Kumar, V. S., Khera, S. K. and Suryanarayana, G. (1995) 'Programmable controllers for special control applications in rolling mills', *Industrial Automation and Control, 1995 (I A & C'95), IEEE/IAS International Conference on (Cat. No.95TH8005)*1995. pp. 173-177.

Rehg, J. A., Swain, W. H., Yangula, B. P. and Wheatman, S. (1999) 'Fieldbus in the process control laboratory-its time has come', *Frontiers in Education Conference, 1999. FIE '99. 29th Annual*1999. pp. 13B4/12-13B4/17 vol.3.

Reynolds and Baldwin (1915), Western Electric Company, *Sequence Switch*, US1127808, United States

Ricketts Jr., A. W., Devault, A. T., Doane, R. C., Dumser Jr., J. M. and Holzer Jr., J. M. (1973), Digital Equipment Corporation (Maynard, MA), *Programmable Machine Controller*, US3753243, United States

Ricketts Jr., A. W., Devault, A. T., Doane, R. C., Dumser Jr., J. M. and M, H. J. J. (1972), Digital Equipment Corp (US), *Control System*, GB1290651, United Kingsdom

Ritchie, A. E. (1949) 'Sequential Aspects of Relay Circuits', *American Institute of Electrical Engineers, Transactions of the*, 68 (1), pp. 577-581.

Roberts, G. K. (1989) 'Science, technology and everyday life, 1870-1950'. In Chant, C. (Ed.) Routledge in association with the Open University, London, pp. 85-87.

Roberts, W. E. (1956), Brookhirst Switchgear LTD, *Improvements in or relating to sequence-operating control systems for electric motors and control switches for use in such systems*, GB746601, United Kingdom

Roessler Jr., C. E. (1958), Roessler Jr., Charles E., *Sequence program control*, US2846892, United States

Rojas, R. and Hashagen, U. (2000) *First Computers : History and Architectures*, MIT Press. Available from: http://libezproxy.open.ac.uk/login?url=http://search.ebscohost.com/login.aspx?direct=true&db=nlebk&AN=49188&site=eds-live&scope=site [Accessed 13 Dec 2011].

Rosli, D., Shalina, S. M., Anwar, A. R., Zaid, H. M. and Ashhar, K. M. (2011) 'Development of an automatic handling system for radioactive waste', *Mechatronics (ICOM), 2011 4th International Conference On*, 17-19 May 2011, 2011. pp. 1-5.

Rullán, A. (1997) 'Programmable logic controllers versus personal computers for process control', *Computers & Industrial Engineering*, 33 (1-2), pp. 421-424.

Russell, D. W. (1990) 'Integration of PLCs and databases for factory information systems', *Systems Integration, 1990. Systems Integration '90., Proceedings of the First International Conference on* 1990. pp. 730-737.

Saito, Y., Doi, T. and Hirase, K. (1986), Omron Tateisi Electronics Co. (Kyoto, JP), *Programmable controller*, US4608628, United States

Sakuragi, T., Takemura, T. and Kaneda, M. (1968) 'F-Matic N Series (Fuji Transistorised Digital Control Elements)', *Fuji Electric Review (Japan)*, 14 (6), pp. 204-214.

Sauter, T. (2010) 'The Three Generations of Field-Level Networks - Evolution and Compatibility Issues', *Industrial Electronics, IEEE Transactions on*, 57 (11), pp. 3585-3595.

Schmidt, R., Meier, W., Wietzeg, R. and Schutz, H. (1973), SIEMENS AG, *Einrichtung Zur Durchfuehrung Boolescher Verknuepfungen* DE2321200, Germany

Schmidt, R., Meier, W., Wietzeg, R. and Schutz, H. (1975), Siemens Aktiengesellschaft (Munich, DT), *Serial programmable combinational switching function generator*, US3902050, United States

Schonberger, R. J. (1999) 'Types of Manufacturing'. In Dorf, R. C. (Ed.) *The technology management handbook*, CRC Press, Boca Raton ; [Great Britain], pp. 1 v. (various pagings).

Schultz, R. E., Schmidt, O. and Johnston, D. A. (1987), Allen-Bradley Company, Inc. (Milwaukee, WI), *Programmable controller with dynamically altered programmable real time interrupt interval*, US4638452, United States

Scrupski, S. (2001) 'Pneumatic Logic Devices Pushed', *Electronic Design*, 49 (11), pp. 50.

Seipp, W. (1974), Gulf & Western Industries, Inc. (New York, NY), *Programmable Controller Using a Random Access Memory*, US3827030, United States

Seipp, W. H. (1977), Gulf & Western Industries, Inc. (New York, NY), *Self-addressing modules for programmable controller*, US4050098, United States

Seipp, W. H. (1978), Gulf & Western Industries, Inc. (New York, NY), *Programmable controller using microprocessor*, US4107785, United States

Seipp, W. H. (1979), Gulf & Western Industries, Inc. (New York, NY), *Programmable controller using microprocessor*, US4180862, United States

Seipp, W. H. (1980), Gulf & Western Industries, Inc. (New York, NY), *Programmable controller using microprocessor*, US4200916, United States

Shannon, C. E. (1938) 'A Symbolic Analysis of Relay and Switching Circuits', *American Institute of Electrical Engineers, Transactions of the*, 57 (12), pp. 713-723.

Shannon, C. E. (1940) *A symbolic analysis of relay and switching circuits*. Thesis M.S. --Massachusetts Institute of Technology Dept. of Electrical Engineering 1940.

Shannon, C. E. (1953) 'Computers and Automata', *Proceedings of the IRE*, 41 (10), pp. 1234-1241.

Shewchuk, J. P. and Moodie, C. L. (1998) 'Definition and Classification of Manufacturing Flexibility Types and Measures', *International Journal of Flexible Manufacturing Systems*, 10 (4), pp. 325-349.

Shizima, M. and Sakuragi, T. (1973) 'Universal sequence controller (USC-4000)', *Fuji Electric Review (Japan)*, 19 (1), pp. 15-22.

Smeallíe, G. R. (1972), *Multiple Process Control System*, US3651484, United States

Soulsby, D. R. and Seipp, W. H. (1978), Gulf & Western Industries, Inc. (New York, NY), *Programmable controller having a system for monitoring the logic conditions at external locations*, US4078259, United States

Speight, J. G. (2007) 'CHAPTER 5: History of Gas Processing'. *Natural Gas: A Basic Handbook*, Euromoney Institutional Investor PLC / Gulf Publishing Company, pp. 113-130.

Stark, A. (1940), *Schaltungsanordnung zur wahlweise nacheinander ueber Fernleitungen erfolgenden Inbetriebsetzung einer Vielzahl von Stromverbrauchern*, DE692228, Germany

Stively, K. H. and Keiver, L. A. (1996) 'Low-budget, high-performance distributed control system', *Industry Applications Conference, 1996. Thirty-First IAS Annual Meeting, IAS '96., Conference Record of the 1996 IEEE* 1996. pp. 2421-2427 vol.4.

Struger, O. J. and Dummermuth, E. H. (1981), Allen-Bradley Company (Milwaukee, WI), *Microprogrammed programmable controller*, US4266281, United States

Struger, O. J. and Grants, V. (1979), Allen-Bradley Company (Milwaukee, WI), *Programmable controller with limit detection*, US4158226, United States

Struger, O. J. and Radtke, J. D. (1974), Allen-Bradly Company (Milwaukee, WI), *Controller Programmer*, US3798612, United States

Struger, O. J., Schultz, R. E. and Sammons, B. E. (1981), Allen-Bradley Company (Milwaukee, WI), *Dual language programmable controller*, US4302820, United States

Sudhir, A. and Sujata, A. (2011) 'Programmable Logic Controllers'. *Instrument Engineers' Handbook*, CRC Press, pp. 90-100.

Takaki, M. (1984), Hitachi Ltd; Hitachi Control Systems, *Sequence Controller*, JP59205605 Japan

Takaki, M. (1986), Hitachi Ltd; Hitachi Control Systems, *Sequence control method and apparatus* GB2175112, UK

Takaki, M. (1987), Hitachi, Ltd. (Tokyo, JP),Hitachi Control Systems, Inc. (Ibaraki, JP), *Sequence control method and apparatus*, US4683549, United States

Taylor Jr., R. C. and Vaniglia, C. L. (1984), Cincinnati Milacron Inc. (Cincinnati, OH), *Programmable control apparatus and method*, US4486830, United States

Tellis, G. J. (2006) 'Disruptive Technology or Visionary Leadership?*', *Journal of Product Innovation Management*, 23 (1), pp. 34-38.

Thoma, G. (2009) 'Striving for a large market: evidence from a general purpose technology in action', *Industrial & Corporate Change*, 18 (1), pp. 107-138.

Thomas, H., Pearson, J. M. and Nopper, E. J. (1940), Houdry Process Corp, *Valve Cycle Timer Apparatus*, CA390832, Canada

Thomesse, J. P. (2005) 'Fieldbus technology in industrial automation', *Proceedings of the IEEE*, 93 (6), pp. 1073-1101.

TNA:PRO-AY (2012) *Records created by the Warren Spring Laboratory and successors* The National Archives of the UK (TNA): Public Record Office (PRO) AY. Available from: http://www.nationalarchives.gov.uk/catalogue/displaycataloguedetails.asp?CATID=647&CATLN=2&accessmethod=5 [Accessed 01 March 2012].

TNA:PRO-CR420 (2012) *Brookhirst Switchgear Ltd*, The National Archives of the UK (TNA): Public Record Office (PRO) CR 420. Available from: http://www.nationalarchives.gov.uk/a2a/records.aspx?cat=017-cr420&cid=0#0 [Accessed 24 September 2012].

Tsypkin, Y. Z. (1984) *Relay control systems,* CUP.

Unver, H. M., Aydemir, M. T. and Celik, V. (2005) 'Control freaks [induction furnace logic controller]', *Power Engineer (see also Power Engineering Journal),* 19 (3), pp. 36-39.

Utterback, J. and Acee, H. (2003) In *Keith Pavitt Conference* Vol. 14 University of Sussex, UK

Vennin, C. F. D. (1929), Westinghouse Brake & Signal, *Improvements relating to signalling systems for road traffic,* GB317927, United Kingdom

Vickers, K. and Chinn, R. W. (1998) 'Passenger terminal baggage handling systems', *Systems Engineering of Aerospace Projects (Digest No. 1998/249), IEE Colloquium on,* 28 Apr 1998, 1998. pp. 6/1-6/7.

Waddington, J. and Maples, G. C. (1987) 'The control of large coal- and oil-fired generating units', *Power Engineering Journal,* 1 (1), pp. 25-36.

Walker, M. J., Bissell, C. C. and Monk, J. (2010) 'The PLC: a logical development', *Measurement + Control,* 43 (9), pp. 280-284.

Wareham, R., Extrusions, A. and Barrie, O. (1988) 'Ladder diagram and sequential function chart languages in programmable controllers', *Fourth Annual Canadian Programmable Control and Automation Technology Conference and Exhibition, 1988,* Toronto, Ont1988. IEEE, pp. 12A-14/1-4.

Webb, E. F. D. (1948), IVOR POWER SPECIALTY COMPANY LIMITED, *Improvements in or Relating to Mechanically Driven Timing and Limit Switch Mechanism,* GB601665, United Kingdom

Wenger, E., McDermott, R. A. and Snyder, W. (2002) *Cultivating Communities of Practice : A Guide to Managing Knowledge,* Harvard Business School Press.

Wenger, E. C. and Snyder, W. M. (2000) 'Communities of Practice: The Organizational Frontier', *Harvard Business Review,* 78 (1), pp. 139-145.

Wilkes, M. (2006) 'What I remember of the ENIAC', *Annals of the History of Computing, IEEE,* 28 (2), pp. 30-31.

Winner, L. (1993) 'Upon Opening the Black Box and Finding It Empty: Social Constructivism and the Philosophy of Technology', *Science, Technology, & Human Values*, 18 (3), pp. 362-378.

WIPO (2011) *International Patent Classification*, World Intellectual Property Organization (WIPO). Available from: http://www.wipo.int/ipcpub/ [Accessed 09 Dec 2011].

Wirsing, H. L. (1971), GTE Automatic Electric Laboratories, Incorporated (Northlake, IL), *Stored-Logic Real Time Monitoring And Control System*, US3624611, United States

Wojtecki, R. G. (1999) *Air logic control for automated systems*, Boca Raton, Fla. CRC Press.

Yano, S. H., JP) (1986), Omron Tateisi Electronics Co. (Kyoto, JP), *Programmable controller having a drum type sequencer function subject to programming*, US4564898, United States

Yetter, E. W. (1966a), Du Pont, *Process Sequence Controller*, CA736102, Canada

Yetter, E. W. (1966b), Pont DU., *Programmed batch sequence controller*, US3275988, United States

Yin, R. K. (1994) *Case study research : design and methods*, (2nd Edn), Sage Publications, Thousand Oaks, Calif.

Yomogida, T., Suzuki, Y. and Ito, K. (1985), Toyoda Machine Works Ltd, *Programmable sequence controller*, US4510580, United States

Young, D. (2007) *Personal Interview with David Young*. Mark Walker (author), IET, Savoy Place, London. dated 19 Sep 2007.

Young, D. (2009) *e-mail correspondence*. Mark Walker (author), 25 April 2009.

Zankl, A. (2006) *Milestones in automation : from the transistor to the digital factory*, Publicis Corp. Publ., Erlangen.

Appendices

Appendix A

Patent Search and Analysis Methods

The patent search was conducted in four stages, each stage employing a search method. Each stage is summarised below and the advantages and disadvantages are explained

1) Key word search (e.g. "Programmable Controller", PLC etc). This stage identified some interesting patents but at random, however this search method provided some key companies and inventors. The disadvantage was that patent titles seldom reflect or accurately describe the technology in a recognisable form resulting in the retrieval of many unrelated patents. Additionally, the patent described a new technology that was versed in the terms of pre-existing technologies.

2) Cited patent search. From the uncovered patents, cited patents (prior art) were traced backwards and forwards in time. This method revealed further patents from known both known inventors and identified new inventors. The limitation of using this method was that it was possible to miss some patents due to a narrow field of companies.

3) The patent data obtained from the key word and cited patent searches was reviewed in its entirety and this identified the "European Classification (ECLA) symbol, based on the International Patent Classification (IPC) classes and sub-classes (WIPO, 2011), pertaining to the general area of programmable controllers. The dominant classification obtained from this method was 'G05B19/05' where the IPC number:

   G - Physics;
   05 - Controlling; Regulating;
   B - Control or regulating systems in general; functional elements of such systems; monitoring or testing arrangements for such systems or elements;
   19 - Programme-control systems
   /05 - Programmable logic controllers, e.g. simulating logic interconnections of signals according to ladder diagrams or function charts

   (WIPO, 2011)

   IPC number explanation: "The European classification (ECLA) symbol is made up of a letter denoting an existing IPC class (International Patent Classification), followed by a number (two digits) denoting the IPC section level (e.g. B65). followed by a sequence of a letter (e.g. B65D) denoting the IPC subclass level, a number (variable, 1-3 digits, e.g. B65D81) denoting the IPC group level, a forward slash "/" and a number denoting the IPC subgroup or full classification." (EPO, 2011)

The full IPC subgroup level was not used on patents published prior 1950 because subgroup '/05'did not exist (although has been applied retrospectively). The IPC classification 'G05B19' was used as a search term and encompassed all subgroups (below the '/') providing all data in the classification for PLCs and importantly associated technologies (such as relay, drum and cam based) to effect sequential and logic control.

4) The final search level was carried out under the IPC classification G05B19/05 and searched by year range in decades (e.g. G05B19/05 1950:1960). This revealed all invented technologies classified under Programmable controllers.

The search systems however do have a potential flaw, all search methods have a human element and are prone to error: Incorrect key words may have been used (Method 1); cited patents may not be directly relevant to the PLC (method 2); the classification may not be ideal and themselves were determined by human beings. Gordon and Cookfair provide a timely warning with relying on the classification system:

> "Remember, the classification system of the PTO is not an exact science; it is influenced by humans. One may have differences of opinion as to classification, but the cross-reference system will generally eliminate the errors." (Gordon and Cookfair, 2000)

The use of classification G05B19/05 however, is supported by the first two methods of patent searching because they identified the classification by reviewing the technology described in the patent literature itself, supporting the use of the classification system.

It is possible to overlook some PLC technology patents because they are classified differently. The practical implication of this is that it precluded a more in-depth analysis of alternative classifications because of the sheer volume of published patents. This is perhaps an area for further research and is beyond the scope of this thesis.

Analysing Patent Literature

The database chosen to conduct the research was the European Patent Office database "Esp@cenet". Esp@cenet is a useful patent web-site because it enables access over 70 million patents world-wide from 1836 onwards. Patent bibliographical data could be exported to external computer applications (e.g. database and spreadsheet applications) and patents downloaded in an accessible electronic document form (PDF). Although the search tools were provided were not extensive, advanced searches could be conducted on the bibliographical data and word searches on the title and abstract where permissible. A further feature and advantage of Esp@cenet was the ability to identify the patent family, those that were associated with the priority number, so it was possible to identify the first patent filing and the country of origin.

Other accessible patent sites such as Google Patents, Free patents Online and Science Direct (patents) were examined but provided only limited patent access, largely centred on American patents.

A unique Microsoft Access database was constructed to contain the patent data. Using a bespoke database enabled additional data to be included in the record, including diagrams and quotations. Links were provided to patent files (PDF) and detailed notes in electronic form. Patents were individually categorised in a simple weighting system as to their perceived relevance and importance to the dev elopement of the PLC. The main advantages of using a purpose designed database was to provide facilities to store patent information from selected and examined patents (277 in total) and provide the ability to conduct specialised search queries on the data obtained therein. An example of a representative database record is shown below, at the end of this appendix.

Difficulties in Translation

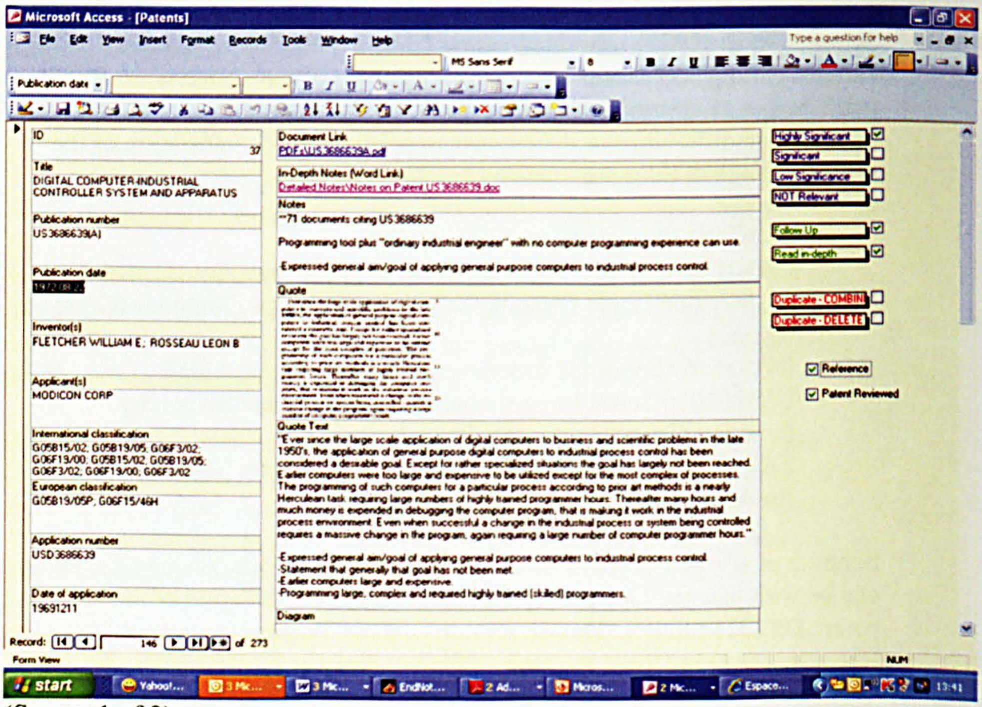
Schmidt, et al. (1973) submitted patent DE2321200 "Einrichtung zur Durchfuehrung boolescher Verknuepfungen" on behalf of Siemens. It was published in 18 countries, strangely under a number of different titles which demonstrate difficulties or differences in opinion on the translation of the title. The English titles are repeated below:

AU6749374 Logic Circuit (Australia)
CA1017418 Device for Performing Logic Operations (Canada)
GB1466466 Logic Circuits for Solving Boolean Functions (Great Britain)
IN138676 Circuit for Processing Binary Signals (India)
US3902050 Serial Programmable Combinational Switching Function Generator (United States)
ZA7402154 Logic Circuits for Processing Binary Signals to Solve Boolean Functions (South Africa)

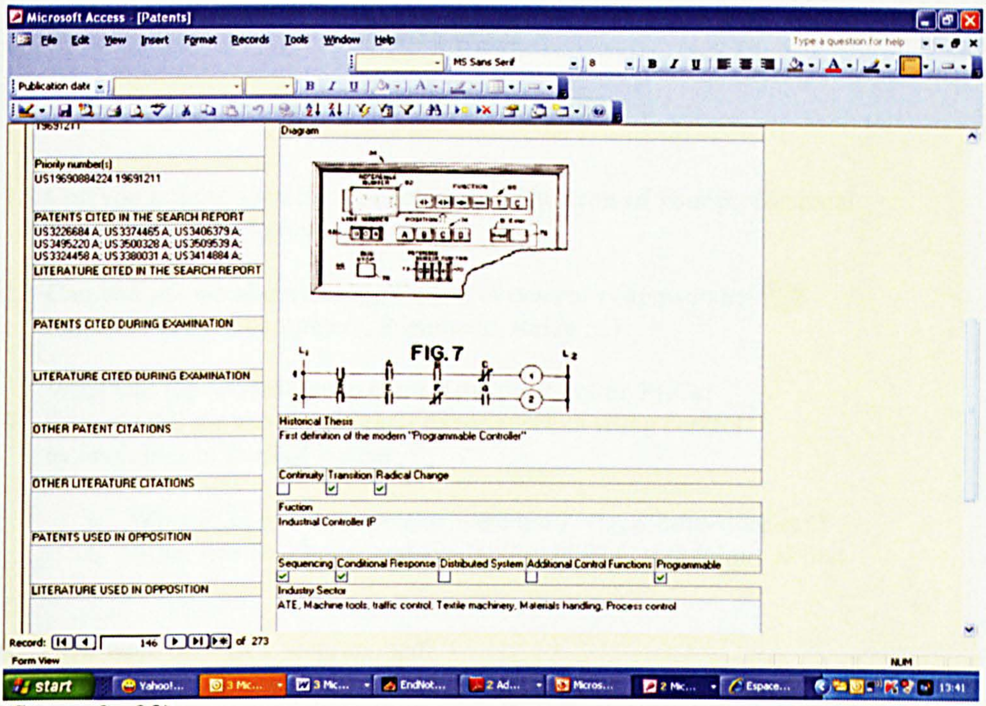
Schmidt et al's patent provides an example of how unrelaible name searches can be with patents. One example of an automated translation of the title for patent DE2321200 is a "Device for carrying out Boolean operations" (Google, 2011) but there are many different variations. US patent versions have been used where possible in the thesis in order to maintain a level of consistency in the document structure and translation. US3902050 "Serial Programmable Combinational Switching Function Generator" (Schmidt et al., 1975) is used to describe the contents of patent DE2321200 in this particular instance.

Patent Database Record

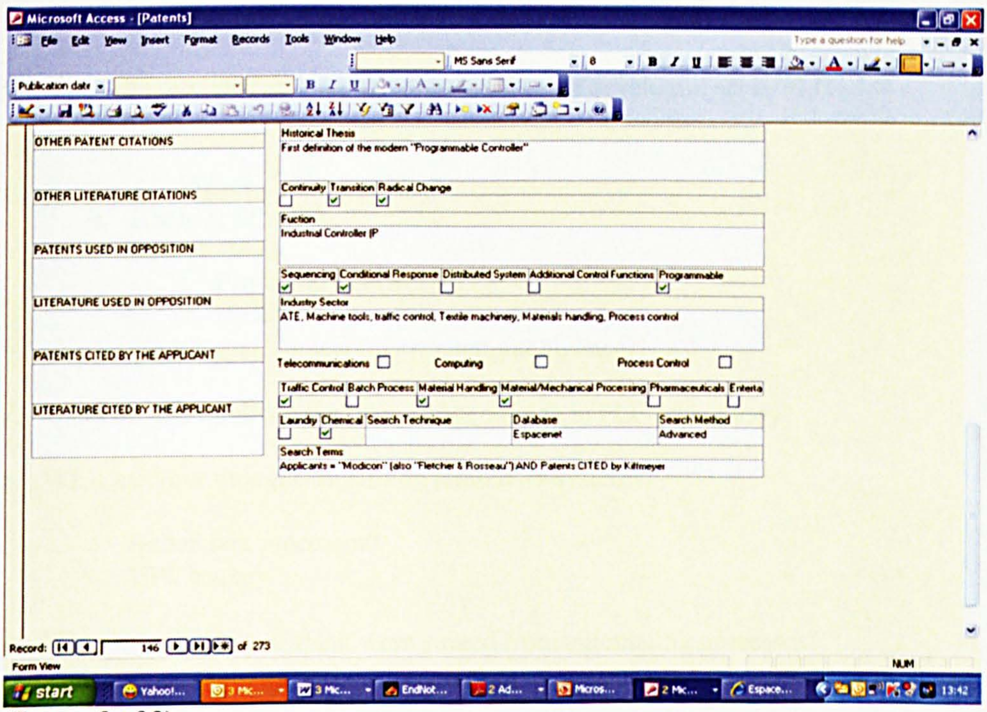
Fletcher and Rosseau (1972) patent US3686639



(Screen 1 of 3)



(Screen 2 of 3)



(Screen 3 of 3)

Appendix B

Sample Interview Questions

1. Can you tell me something about yourself in terms of your professional and industrial background?

2. Can you tell me about the application of control systems in the **???** industry? (PLCs, computers, Pneumatic, Relay …)

3. What was the technology to control the plant before PLCs?
4. Can you tell me about your early experiences of using control technologies in the **???** industry?

    a. What PLC memory systems were used? (tape, core-store etc.)
    b. What was the choice and availability of PLC technology at that time?

5. How were the PLCs programmed?

    a. Did they use dedicated programming devices?
    b. If so, how did they work or how were they used?
    c. Who controlled the device?

6. What in your opinion were the most significant developments in PLC technology?

    a. Analogues
    b. HMIs & SCADA
    c. Remote I/O?
        i. Proprietary networks?
        ii. Fieldbus?
    d. Distributed Control (distributed intelligence)?

7. What do you think are the main shortcomings in PLC technology?

8. What are your thoughts regarding redundant systems?

    a. Redundant processor?
    b. UPS backup

9. What benefits do you think were gained from automating processes?

10. Do you have a preferred technology for performing control and automation?

    a. If yes, what is it and why?
    b. If no, why?

11. What (other) applications would you use a PLC/computer for?

12. What models of PLC did you use?

13. Were the PLCs linked to other systems (e.g. SCADA) in any way? Did this involve using fieldbus or other networking technologies?

14. Software control and management

    a. Do you have a preferred programming language? *What and why?*
    b. What was/is the process for choosing the control technology and specifying or designing the application?
    c. Was/is system security an issue?
        i. PLC programming key
        ii. Location and access of master copies
        iii. Version control

15. What strategy do you use for safety systems on PLC controlled plant and equipment?

    a. Mechanical and hard-wired (e.g. safety valves and directly wired E-stops)
    b. What do you think about new safety systems like Profisafe and safety PLCs?

16. Was computer control employed anywhere else in the industry?

17. What is you opinion of the new PC-based control systems such as "soft PLCs"?

18. What would you use for new control applications? (Soft PLCs, Programmable Automation Controllers (PACs), DCS...)

19. What major issues did you have using PLCs?

20. What do you see for the future of control systems in the **???** industry?

    a. What direction do you think it will take?
    b. Is there in your opinion a future for PLCs?

21. Is there a topic you feel that I should have asked you about, but did not, or anything else you want to add?

Appendix C

Bibliography of Significant Patents

Title	Publication Number	Publication date	Inventor(s)	Applicant
Automatic Selective Switching Apparatus suitable for use in Telephone Exchanges.	GB190920839 (A)	1910-01-13	MCBERTY FRANK R	FRANK R MCBERTY
An Automatic Sequence Switching Device and its Control.	GB190920840 (A)	1910-06-30	KINGSBURY JOHN EDWARD	KINGSBURY JOHN EDWARD
Improvements for Signalling, Regulating and Controlling Street Traffic and in the Method of its Application for such Traffic.	GB191317337 (A)	1914-07-23	FIRTH THOMAS WILLIAMS STAPLEE; KENNNEDY JOHN NASSAU CHAMBERS	FIRTH THOMAS WILLIAMS STAPLEE; KENNNEDY JOHN NASSAU CHAMBERS
SEQUENCE-SWITCH	US1127808 (A)	1915-02-09	REYNOLDS; BALDWIN	WESTERN ELECTRIC COMPANY
Improvements in the Manufacture of Glass Bottles and in Glass Shaping and Blowing Machines therefor.	GB111868 (A)	1917-12-11	WILZIN ARTHUR [FR]	WILZIN ARTHUR [FR]
Signal system for controlling street traffic	US1328269 (A)	1920-01-20	DAVIS BENJAMIN W	
Improvements in ionic relays	GB148582 (A)	1920-08-05		WILLIAM HENRY ECCLES; FRANK WILFRED JORDAN
Automatically-controlled water-gas set	US1498174 (A)	1924-06-17	KENNEDY JAMES S	
Sequence switch	US1504283 (A)	1924-08-12	TAYLOR HERBERT B	WESTERN ELECTRIC CO
Flat paper cup forming machine	GB237037 (A)	1925-07-23		VORTEX MFG CO
Improvements in or relating to electric motor controllers	GB251559 (A)	1926-05-06		CUTLER HAMMER MFG CO
Sequence switch	US1606785 (A)	1926-11-16	HODGKINS CHARLES H	WESTINGHOUS E ELECTRIC & MFG CO
Improvements in apparatus for controlling road traffic	GB262271 (A)	1926-12-09		LOUIS PERCIVAL MILES
Improvements in machines for making articles of wood pulp	GB268483 (A)	1927-04-07		JOHN JAMES HENRY STURMEY
Automatic sequence control for valves	US1634327 (A)	1927-07-05	KENNEDY JAMES S	BARTLETT HAYWARD CO
Sequence interlock for automatic valves	US1648052 (A)	1927-11-08	KENNEDY JAMES S	BARTLETT HAYWARD CO
Improvements relating to signalling systems for road traffic	GB317927 (A)	1929-08-29		CHARLES FRANCIS DICKSON VENNIN; WESTINGHOUS E BRAKE & SIGNAL
Improvements in the automatic electric control of cyclic operations in mechanisms andother equipment	GB348659 (A)	1931-05-11		HENRY CHARLES JENKINS
Improvements in traffic control systems	GB360471 (A)	1931-11-06		AUTOMATIC SIGNAL CORP
Automatic sequence control means	US1939183 (A)	1933-12-12	NEUMAN JACOB J	
Schaltungsanordnung zur wahlweise nacheinander ueber Fernleitungen erfolgenden Inbetriebsetzung einer Vielzahl von Stromverbrauchern	DE692228 (C)	1940-06-15		ALFRED STARK
VALVE CYCLE TIMER APPARATUS	CA390832 (A)	1940-08-20	THOMAS HENRY; et al.	HOUDRY PROCESS CORP
Successive switching arrangement	US2250453 (A)	1941-07-29	APPEL HENRY J	HOUDRY PROCESS CORP

Improvements in or Relating to Mechanically Driven Timing and Limit Switch Mechanism	GB601665 (A)	1948-05-11	EDWARD FRANCIS DANIEL WEBB	IVOR POWER SPECIALTY COMPANY LIMITED
Multiple furnace control	US2451518 (A)	1948-10-19	STRICKLAND JR HAROLD A	OHIO CRANKSHAFT CO
High-speed sequence control	US2492749 (A)	1949-12-27	HILLS WILLIAM B	GEN ELECTRIC
Improvements in or relating to automatic control means for gas making and similar cyclically operated apparatus	GB633759 (A)	1949-12-30		POWER GAS LTD; GEORGE WRIGHT; FRANK RICHMOND HOLMES
Improved multiple sequence electric control means for use with resistance welders, electrical circuits and process control apparatus	GB672381 (A)	1952-05-21		ASTON ELECTRICAL PRODUCTS PTY
Binary automatic computer	GB736144 (A)	1955-09-07		REMINGTON RAND INC
Multicoordinate digital information storage device	US2736880 (A)	1956-02-28	FORRESTER JAY W	RESEARCH CORP
Improvements in or relating to sequence-operating control systems for electric motors and control switches for use in such systems	GB746601 (A)	1956-03-14	WILLIAM ERIC ROBERTS	BROOKHIRST SWITCHGEAR LTD; WILLIAM ERIC ROBERTS
Sequence program control	US2846892 (A)	1958-08-12	ROESSLER JR CHARLES E	
Improvements in or relating to automatic traffic signalling systems	GB824781 (A)	1959-12-02	SHAND GEORGE; ATKINSON GEORGE GEOFFREY	VICKERS ELECTRICAL CO LTD
Programming and sequence control for fuel burners	US3008517 (A)	1961-11-14	PIERZ RICHARD J	
General purpose parallel sequencing computer	US3204087 (A)	1965-08-31	MILLIS JR HUGH L	
Electro-mechanical plugboard sequencing apparatus	US3205369 (A)	1965-09-07	EACHUS JOSEPH J; HARPER SAMUEL D	HONEYWELL INC
Traffic control system	US3206721 (A)	1965-09-14	RUDDEN JAMES B; GERLOUGH DANIEL L	BUNKER RAMO
Computer control apparatus	US3226684 (A)	1965-12-28	COX THEODORE C	IBM
Stored logic computer	US3246303 (A)	1966-04-12	AMDAHL LOWELL D; et al.	THOMPSON RAMO WOOLDRIDGE INC
SEQUENCE CONTROL LAYOUT FOR AUTOMATION SYSTEMS	CA733654 (A)	1966-05-03	JENSEN BORGE H F	JENSEN BORGE H F
PROCESS SEQUENCE CONTROLLER	CA736102 (A)	1966-06-07	YETTER EDWARD W	DU PONT
PROCESS FOR MANUFACTURE OF DIAROYLAMINOANTHROQ UINONE PIGMENTS	CA737961 (A)	1966-07-05	EHRICH FELIX F; JAFFE EDWARD E	DU PONT
Sequence controller	US3264612 (A)	1966-08-02	YETTER EDWARD W	DU PONT
PROGRAMMED BATCH SEQUENCE CONTROLLER	CA742364 (A)	1966-09-06	YETTER EDWARD W	DU PONT
Programmed batch sequence controller	US3275988 (A)	1966-09-27	YETTER EDWARD W	DU PONT
Monitoring apparatus	US3324458 (A)	1967-06-06	GERARD MACARTHUR JOHN	BUNKER RAMO
Sequence control arrangements for a computer	GB1110463 (A)	1968-04-18		JEUMONT SCHNEIDER
Remote calculator	US3380031 (A)	1968-04-23	BURKE CLAYTON BILLY; et al.	CONTROL DATA CORP
Improvements relating to automation of working machines	GB1126891 (A)	1968-09-11		HERMAN BA RGE FUNK JENSEN
Digital data processing system	US3406379 (A)	1968-10-15	MAX PALEVSKY; et al.	SCIENT DATA SYSTEMS INC

Electronic process control devices	US3414884 (A)	1968-12-03	FUNCK JENSEN HERMANN BORGE	HERMANN BORGE FUNCK JENSEN
PROCESS CONTROL SYSTEM INCLUDING HARDWARE ELEMENT STATUS MAP IN MEMORY	US3495220 (A)	1970-02-10	LAWSON DAVID A; PETERSON RALPH W; STOCKERT ALFRED A	BELL TELEPHONE LABOR INC
DATA SYSTEM MICROPROGRAMMING CONTROL	US3500328 (A)	1970-03-10	WALLIS DONALD E	IBM
STORED-LOGIC REAL TIME MONITORING AND CONTROL SYSTEM	US3624611 (A)	1971-11-30	WIRSING HOWARD LEROY	GTE AUTOMATIC ELECTRIC LAB INC
MULTIPLE PROCESS CONTROL SYSTEM	US3651484 (A)	1972-03-21	SMEALLIE GEORGE R	BAILEY METER CO
DIGITAL COMPUTER-INDUSTRIAL CONTROLLER SYSTEM AND APPARATUS	US3686639(A)	1972-08-22	FLETCHER WILLIAM E; ROSSEAU LEON B	MODICON CORP
[CONTROL SYSTEM]	GB1290651 (A)	1972-09-27	RICKETTS JR A W; et al.	DIGITAL EQUIPMENT CORP [US]
ANALYZER FOR SEQUENCER CONTROLLER	US3701113 (A)	1972-10-24	CHACE DONALD E; et al.	DIGITAL EQUIPMENT CORP
CONTROL SYSTEMS	US3731279 (A)	1973-05-01	HALSALL J; MURRELL A	HALSALL J; MURRELL A
DIGITAL COMPUTER	US3740722 (A)	1973-06-19	GREENBERG M; FLETCHER W; MORLEY R	MODICON CORP
PROGRAMMABLE MACHINE CONTROLLER	US3753243 (A)	1973-08-14	RICKETTS A; DEVAULT A; DOANE R; DUMSER J; HOLZER J	DIGITAL EQUIPMENT CORP
DIGITAL COMPUTER	US3761893 (A)	1973-09-25	MORLEY R	MODICON CORP
PROCESS CONTROL COMPUTER	US3761882 (A)	1973-09-25	BARTLETT P; HENRY D; MURRELL T	STRUTHERS DUNN
CONTROLLER PROGRAMMER	US3798612 (A)	1974-03-19	STRUGER O; RADTKE J	ALLEN BRADLY CO
PROGRAMMABLE CONTROLLER EXPANSION CIRCUIT	US3806877 (A)	1974-04-23	KIFFMEYER W; BARON L	ALLEN BRADLEY CO
SEQUENCE CONTROLLER	US3806714 (A)	1974-04-23	OTSUKA K; NAKAGAWA T; SHIMOKAWA Y	TOKYO SHIBAURA ELECTRIC CO
PROGRAMMABLE MATRIX CONTROLLER	US3810118 (A)	1974-05-07	KIFFMEYER W	ALLEN BRADLEY CO
PROGRAMMABLE CONTROLLER USING A RANDOM ACCESS MEMORY	US3827030 (A)	1974-07-30	SEIPP W	GULF & WESTERN INDUSTRIES
GENERAL PURPOSE SEQUENCE CONTROLLER	US3832696 (A)	1974-08-27	NAKAO H; NARUSE K; TOKURA Y; MATSUNO K; HASEGAWA K; KAWADE S	TOYODA MACHINE WORKS LTD
EINRICHTUNG ZUR DURCHFUEHRUNG BOOLESCHER VERKNUEPFUNGEN	DE2321200 (A1)	1974-11-07	SCHMIDT RUDOLF DIPL ING; et al.	SIEMENS AG
PROGRAMMABLE LOGIC CONTROLLER	US3849765 (A)	1974-11-19	HAMANO G	MATSUSHITA ELECTRIC IND CO LTD
PROGRAMMABLE MATRIX CONTROLLER	CA959144 (A1)	1974-12-10	KIFFMEYER WILLIAM W	ALLEN BRADLEY CO
PROCESS CONTROL COMPUTER	US3881172 (A)	1975-04-29	BARTLETT PETER G; HENRY DONALD E	STRUTHERS DUNN
PROGRAMMABLE CONTROL APPARATUS	US3886528 (A)	1975-05-27	IRANI JAMSHED; END EDUARD	SPRECHER & SCHUH AG
SERIAL PROGRAMMABLE COMBINATIONAL SWITCHING FUNCTION GENERATOR	US3902050 (A)	1975-08-26	SCHMIDT RUDOLF; MEIER WERNER; WIETZEG	SIEMENS AG

			RAINER; SCHUTZ HARTMUT	
Data transfer and manipulation apparatus for industrial computer controllers	US3930233 (A)	1975-12-30	MORLEY RICHARD E; SCHELBERG JR CHARLES C	MORLEY RICHARD E; SCHELBERG JR CHARLES C
Programmable logic controller	US3942158 (A)	1976-03-02	DUMMERMUTH ERNST	ALLEN BRADLEY CO
Digital logical sequence controller	US3944987 (A)	1976-03-16	KOYANAGI HARUO; et al.	MITSUBISHI ELECTRIC CORP
Programmable logic controller with push down stack	US3953834 (A)	1976-04-27	BURKETT BOBBY G; HENRY RAYMOND W	TEXAS INSTRUMENTS INC
APPARATUS FOR PROCESSING DATA IN THE FORM OF A BOOLEAN EXPRESSION	GB1444791 (A)	1976-08-04	Allen R Holecek	BABCOCK & WILCOX
Sequential control system	US3995257 (A)	1976-11-30	IKI SHUNICHI	NISSAN MOTOR
Programmable sequence controller	US3996565 (A)	1976-12-07	NAKAO HISAJI; et al.	TOYODA MACHINE WORKS LTD; TOYOTA MOTOR CO LTD
METHOD FOR CARRYING OUT LOGICAL OPERATIONS IN A CONTROL COMPUTER	GB1473710 (A)	1977-05-18		HEINE K
General purpose sequence controller	US4025902 (A)	1977-05-24	NAKAO HISAJI; et al.	TOYODA MACHINE WORKS LTD
Variable module memory	US4030080 (A)	1977-06-14	BURKETT BOBBY G; HENRY RAYMOND W	TEXAS INSTRUMENTS INC
Self-addressing modules for programmable controller	US4050098 (A)	1977-09-20	SEIPP WILLIAM H	GULF & WESTERN INDUSTRIES
PROGRAMMABLE LOGIC CONTROL SYSTEM WITH MEMORY FOR TEMPORARY STORAGE	GB1490550 (A)	1977-11-02		TEXAS INSTRUMENTS INC
PROGRAMMABLE LOGIC CONTROL SYSTEM WITH MEMORY FOR TEMPORARY STORAGE	GB1490548 (A)	1977-11-02		TEXAS INSTRUMENTS INC
PROGRAMMABLE SEQUENCE CONTROLLER	GB1493319 (A)	1977-11-30		TOYODA MACHINE WORKS LTD
PROGRAMMIERBARER PROZESSRECHNER ODER AUTOMAT	DE2735874 (A1)	1978-02-23	THUILLIER GUY [FR]; et al.	TELEMECANIQ UE ELECTRIQUE
Programmable controller having a system for monitoring the logic conditions at external locations	US4078259 (A)	1978-03-07	SOULSBY DONALD R; SEIPP WILLIAM H	GULF & WESTERN INDUSTRIES
AUTOMATE PROGRAMMABLE POUR LA COMMANDE DU DEROULEMENT D'UN CYCLE DE FONCTIONNEMENT D'INSTALLATION	FR2361689 (A1)	1978-03-10	BLANCHARD MICHEL; et al.	ANVAR [FR]
Programmable controller using microprocessor	US4107785 (A)	1978-08-15	SEIPP WILLIAM H	GULF & WESTERN INDUSTRIES
APPARATUS CONTROL CIRCUIT	GB1529325 (A)	1978-10-18		BABCOCK & WILCOX CO
Programmable controller with limit detection	US4158226 (A)	1979-06-12	GRANTS VALDIS [US]; STRUGER ODO J [US]	ALLEN BRADLEY CO
Digital input/output system and method	US4162536 (A)	1979-07-24	MORLEY RICHARD E [US]	GOULD INC [US]
Digital control system with Boolean processor	US4165534 (A)	1979-08-21	DUMMERMUTH ERNST; et al.	ALLEN BRADLEY CO

Programmable controller using microprocessor	US4180862 (A)	1979-12-25	SEIPP WILLIAM H [US]	GULF & WESTERN INDUSTRIES [US]
System for converting analog signals to multiplexed digital data	US4188617 (A)	1980-02-12	FAUCHIER JESS F [US]; SEIPP WILLIAM H [US]; WHITESIDE STEPHEN E [US]	GULF & WESTERN INDUSTRIES [US]
MICROCOMPUTER CONTROLLER	JP55028199 (A)	1980-02-28	HAWAADO SHII MOTSUKU; KAARU WATOSON CHIE	XEROX CORP
Programmable controller using microprocessor	US4200916 (A)	1980-04-29	SEIPP WILLIAM H [US]	GULF & WESTERN INDUSTRIES [US]
Programmable sequence controller with drum emulation and improved power-down power-up circuitry	US4213174 (A)	1980-07-15	BROMBERG MICHAEL A [US]; MORLEY RICHARD E [US]; TAYLOR WILLIAM A [US]	ANDOVER CONTROLS CORP
Process control system that controls its outputs according to the results of successive analysis of the vertical input columns of a hypothetical ladder diagram	US4217658 (A)	1980-08-12	HENRY DONALD E [US]; et al.	STRUTHERS DUNN
Integrated circuit controller programmable with unidirectional-logic instructions representative of sequential wire nodes and circuit elements of a ladder diagram	US4227247 (A)	1980-10-07	KINTNER PAUL M	EATON CORP
Microprogrammed programmable controller	US4266281 (A)	1981-05-05	STRUGER ODO J; DUMMERMUTH ERNST H	ALLEN BRADLEY CO
SYSTEM FOR CONVERTING ANALOGUE SIGNALS TO MULTIPLEXED DIGITAL DATA	GB1592405 (A)	1981-07-08		GULF & WESTERN INDUSTRIES
Mini-programmable controller	US4282584 (A)	1981-08-04	BROWN RONALD A; HU SUNG C; STRUGER ODO J	ALLEN BRADLEY CO
Programmable controller	US4292666 (A)	1981-09-29	HILL LAWRENCE W; et al.	GOULD INC MODICON DIV
Dual language programmable controller	US4302820 (A)	1981-11-24	STRUGER ODO J; et al.	ALLEN BRADLEY CO
Programmable controller using microprocessor	US4303990 (A)	1981-12-01	SEIPP WILLIAM H	GULF & WESTERN INDUSTRIES
Programmable controller	US4307447 (A)	1981-12-22	PROVANZANO SALVATORE R; et al.	GOULD INC
OPERATING METHOD OF LADDER CIRCUIT INPUT PART ON COLUMN CYCLE SYSTEM	JP57000705 (A)	1982-01-05	SASAKI JIYUNICHI; OKAYAMA YOSHIHIKO	TOSHIBA MACHINE CO LTD
UNIT FOR CONTROLLING REMOTE INPUT AND OUTPUT IN PROGRAMMABLE LOGICAL CONTROLLER	JP57098007 (A)	1982-06-18	HOTSUTA MASAAKI	OMRON TATEISI ELECTRONICS CO
SEQUENCE CONTROLLER	JP58002906 (A)	1983-01-08	IZUMIDA MASAHIRO	SHARP KK
SEQUENCE CONTROLLER	JP59103105 (A)	1984-06-14	DAIGO HIROKI	KOYO ELECTRONICS IND CO
Programmable controller	US4484303 (A)	1984-11-20	PROVANZANO SALVATORE R [US]; et al.	GOULD INC [US]
Programmable control apparatus and method	US4486830 (A)	1984-12-04	TAYLOR JR RALPH C [US]; VANIGLIA CHRISTOPHER L [US]	CINCINNATI MILACRON INC [US]

Programmable sequence controller	US4510580 (A)	1985-04-09	YOMOGIDA TOSHIHIKO [JP]; SUZUKI YASUO [JP]; ITO KYOJI [JP]	TOYODA MACHINE WORKS LTD [JP]
Programmable controller having a drum type sequencer function subject to programming	US4564898 (A)	1986-01-14	YANO SATOSHI [JP]	OMRON TATEISI ELECTRONICS CO [JP]
CONTROLLER WITH CRT DISPLAY	JP61101809 (A)	1986-05-20	MASAI KOICHIRO; et al.	MATSUSHITA ELECTRIC IND CO LTD
Programmable controller	US4608628 (A)	1986-08-26	SAITO YASHITANE [JP]; et al.	OMRON TATEISI ELECTRONICS CO [JP]
Programmable controller with dynamically altered programmable real time interrupt interval	US4638452 (A)	1987-01-20	SCHULTZ RONALD E [US]; et al.	ALLEN BRADLEY CO [US]
Programmable controller with function chart interpreter	US4742443 (A)	1988-05-03	ROHN DAVID R [US]; et al.	ALLEN BRADLEY CO [US]
Multiprocessor device for interconnection between computers, programmable logic controllers and peripheral terminals	FR2612661 (A1)	1988-09-23	ROMIEUX ALAIN	REIGA [FR]
COMPUTER WITH BIT PROCESSOR AND WORD PROCESSOR	DE3720006 (A1)	1988-12-29	HINSKEN GERHARD PROF DR ING [DE]	SIEMENS AG [DE]
Stored-program control system	DE3808135 (A1)	1989-09-28	ABENDROTH PETER DIPL ING [DE]; SASSENBACH HELMUT DIPL ING [DE]	KLOECKNER MOELLER ELEKTRIZIT [DE]
LADDER SEQUENCE CONTROLLER	WO8909952 (A1)	1989-10-19	WATT KIM J [US]; et al.	SQUARE D CO [US]

Appendix D

Notes on patent US3810118 - Programmable Matrix Controller

Kiffmeyer (1974) filed patent US3810118 on behalf of Allen-Bradley for the PMC in 1971; its operation is briefly described below:

The Programmable Matrix Controller (PMC) reads the status of a defined set of input devices and operates a defined set of output devices according to a set of instructions stored within a memory. Inputs are defined as 2-state devices to give on/off or open/closed status (digital) and examples include limit switches, push buttons and thermostats. Similarly, outputs are also defined as 2-state devices and include actuators, solenoids, control valves and indicator lights. Although the PMC can be regarded as a type of computer, emphasis was placed on interfacing with a large number [for the time] of input and output devices rather than providing "extended computational capabilities".

The purpose of the PMC was to provide a controller that would automatically control and sequence the output devices depending on the conditions of a particular set of input devices. It was designed to be easy to use by control engineers who according to Kiffmeyer "determine the sequence of operations ... by the use of ladder diagrams". The control engineers Kiffmeyer refers to, were generally not trained to transfer these "ladder diagram" solutions directly into a computer language and so the PMC was designed to allow them to enter instructions directly from the sequence depicted in the diagram; the rationale was to minimise the skill and training necessary to program the controller.

A further objective of the PMC was to allow the stored program to be easily and inexpensively changed for equipment modifications, production improvements and so on. Hard-wired relay panels and controllers had to be rewired to effect these changes which could be time consuming and expensive. Computer control, apart from the expensive initial investment in hardware and the subsequent programming costs were also expensive to upgrade if there was no in-house programming resources.

Programming the PMC

In order to illustrate how the PMC was programmed a simple motor control circuit shown in Figure 2 is used. The input devices are labelled as follows: start button (I 44), stop button (normally closed - I 45), motor auxiliary latch relay contacts (I 46). The Output device is an electric motor 'M' with address output 12 (O 12). The labels I 44, I45, I46 and O 12 correspond to the input and output address of each device on the PMC.

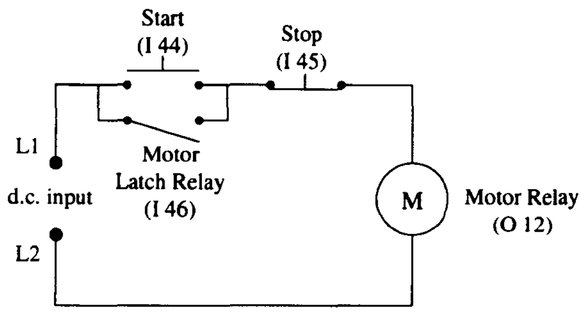


Figure 1. Motor control circuit

The circuit functions by allowing the operator to start the motor (O 12) if the Start button (I 44) is closed AND the stop button is NOT pressed. If this condition is met the motor starts and the auxiliary relay (I 46) closes, holding the motor on when the start push button is released. The motor is stopped when the stop button (I 45) is pressed breaking the power supply line to the motor.

The ladder diagram equivalent to the circuit depicted in figure 2 is shown below:

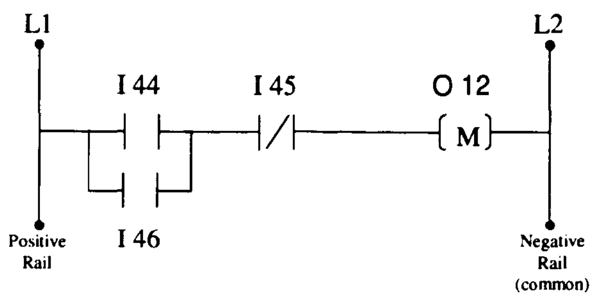


Figure 2. Motor control Ladder Diagram

An alternative way of reading this diagram would be to take each possible path in turn. There are two possible paths through this diagram which can be described as follows:

Path 1: **IF** start button (I 44) is closed **AND** stop button (I 45) is **NOT** closed (i.e. open)

OR

Path 2: **IF** motor latch (I 46) is closed **AND** stop button (I 45) is **NOT** closed (i.e. open)

THEN

Activate motor (**SET** O 12)

The programming method for the PMC was based on four distinct operations:

1. Read input and test if closed (Examine if Closed - **XIC**)
2. Read input and test if open (Examine if Open – **XIO**)
3. Save the result and branch to new line (Branch to new line – **BRT**)
4. Set (activate) the output (**SET**)

This set of instructions can be applied directly to the ladder diagram depicted in Figure 3:

XIC I 44	(Examine if start button closed)
XIO I 45	(Examine if stop button open)
BRT	(save result and start next path)
XIC I 46	(Examine if motor latch closed)
XIO I 45	(Examine if stop button open)
SET O 12	(SET (activate) the motor output)

It can be seen from this example that programs loaded into the matrix can be directly taken from the ladder diagram equivalent. The program above would occupy six consecutive locations in the memory matrix. The four operations XIC, XIO, BRT and SET are the two most significant bits of the 8-bit word stored in one of the locations in the memory matrix; they are represented as 00, 01, 10 and 11 respectively. The 6 least significant bits of the word form the address of the input or output. For example the start push button is located at address 44 which by its circuit design is an input.

The full 6-bit input-output address range of the PMC is thus $000\ 000_2$ to $111\ 111_2$ giving a total of 64 individually addressable locations. Because there are 6 bits in the address range, it is convenient to reference the address in the base 8 or octal number system.
Therefore the actual start push button address should be read as 44_7 which equates to binary $100\ 100_2$. From this it can be seen that one instruction can be held as a single 8-bit word within the memory matrix. For example the full instruction XIC I 44 is thus represented as $00\ 100\ 100_2$ and XIO I 45 as $01\ 100\ 101_2$ and so on.

Table 1 shows the motor control program in mnemonic and binary formats. The BRT instruction has no address associated with it therefore the address value is indeterminate and has no effect; this is represented as an 'x' in the table.

Memory Address	PMC Mnemonic	Binary Equivalent
00	XIC I 44_7	$00\ 100\ 100_2$
01	XIO I 45_7	$01\ 100\ 101_2$
02	BRT	$10\ xxx\ xxx_2$
03	XIC I 46_7	$00\ 100\ 110_2$
04	XIO I 45_7	$01\ 100\ 101_2$
-05	SET O 12_7	$11\ 001\ 010_2$

Table 1. Motor Control Program

Program Address Range

Similarly, the address range of the memory matrix is 64 consecutive words, the program detailed above would require 6 locations in memory. The input output device range should not be confused with the memory address range. Kiffmeyer uses a 64- word diode memory matrix in his patent but also states that alternative memory devices may be used such as a core store; a larger memory size will allow longer and more sophisticated programs.

Dummermuth (1976) supports this and states that fusible-link Programmable Read Only Memory (PROM) chips were used for the memory matrix on the commercial adaptation of the PMC.

Input/Output (I/O) Address Range

The designation of inputs 'I' and outputs 'O' is only used in this instance to clarify the example above; the PMC has a fixed number of 64 input/output addresses limited by the 6-bit address. Physical Inputs and outputs were organised as printed circuit boards (PCBs) with four of each type per PCB. It was possible to arrange combinations in groups of 4 (e.g. 4-inputs or 4-outputs) to suit a particular application. The logic operations are applied to any particular address regardless of type and the programmer would need to know which addresses were configured as outputs so that the correct operation was applied. The status of any particular output can also be read and regarded as an input within in the logic itself. Inputs however, could not be written to via the SET instruction due to their electronic design.

Operation of The PMC

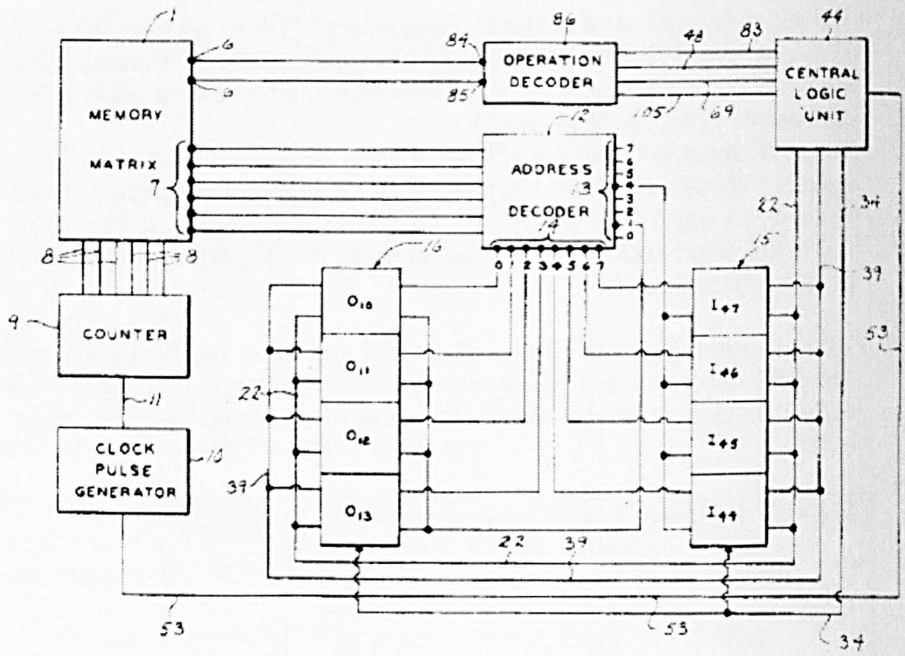


Figure 3. PMC Block Diagram

The block diagram depicted in Figure 3 shows the key features of the PMC. In summary, the PMC operates as follows:

1. The Clock Pulse Generator provides timing and synchronisation for the PMC and increments the Counter.
2. The Counter reads each line of a program held in the memory matrix in turn. Each line of the program consists of a control code and a memory address for the selection of inputs or outputs.
3. The control code (2-bits) is decoded by the Operation Decoder which instructs the Central Logic Unit to perform one of four

operations on the memory location addressed (input or output device).

4. The individual input or output device determined by the memory address is activated by the Address Decoder.

The description above is perhaps a little over simplified. In order to examine the operation of the PMC, it is worth looking at each component in turn.

Clock Pulse Generator and Counter

This Clock Pulse Generator circuit is simply a 100 KHz square wave generator. Its purpose is to provide timing pulses to the Counter and the Central Logic Unit. The Counter on receiving a clock pulse generates 6-bit digital signals to address a unique location in the Memory Matrix. The memory matrix shown has 64 unique locations and each one is addressed sequentially. After the last location has been addressed, the counter repeats the sequence thereby re-reading the program continuously.

Memory Matrix

The term "matrix" refers to a programmable memory array, each line of the array consisting of an 8-bit value or word; the embodiment referred to in Kiffmeyers patent uses a diode matrix which is a read only memory having a capacity of 64 eight-bit words. However, Kiffmeyer states that larger 8-bit memories such as core stores could be used or additional diode memory matrices could be added for longer programs.

Each 8-bit word consisted of a 2-bit operation code (op-code) and a 6-bit address. The 6-bit address enabled a combined total of 64 discrete inputs and outputs to be individually addressed via decoding circuitry. Inputs and outputs were determined by their circuitry based on printed circuits designed in groups of four inputs or four outputs, this enabled combinations that matched the application to be determined. For example, an application may have 16 outputs and 48 inputs. The operation code consisted of the two most significant bits of the word from which were derived the four instructions XIC, XIO, BRT and SET.

Address Decoder

The input to this circuit is the 6-bit Input/Output (I/O) bits connected directly from the memory matrix to the address decoder. Figure 3 shows the bottom or "least significant" lines are connected to four input circuits and four output circuits (e.g. 'I_{47}' and 'O_{10}'), these circuits would be repeated for the number of inputs and outputs connected to the PMC, only one set is shown for clarity. The outputs to the right-hand side of the address decoder are used to select these sets of inputs and outputs.

Operation Decoder

The Operation Decoder circuit takes the two most significant bits from the program word instruction and decodes the appropriate logic for the particular operation code required. The circuit consists of a simple arrangement of logic gates to derive four different outputs from the two binary coded digits. Although four lines are shown only three lines are actually used. The 'XIC' code (00_2) produces logic '0' or 'low' on each line therefore providing a determinable state. All other codes produce logic '1' or 'high' one of the remaining three lines thus enabling the logic circuitry for the appropriate function.

Central Logic Unit

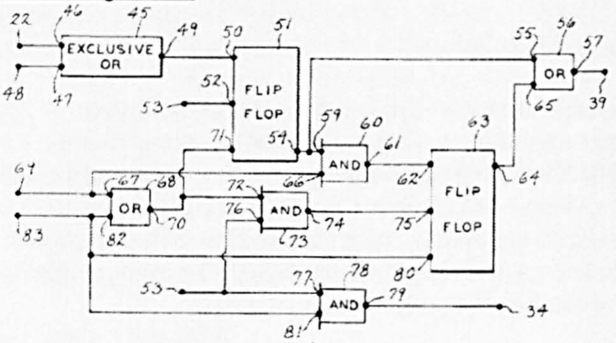


Figure 5. Central Logic Unit

The output of the decoder directly drives the Central Logic Unit (CLU) and enables the appropriate logic to perform the functions XIC, XIO, BRT and SET. In order to explain the principle of the CLU, it is necessary to simplify. The two "Flip Flops" are used as memory devices. Input line (22) is a single bit bus to which all inputs are connected. The initial conditions are both Flip Flops reset.

Using the motor control program as an example, the first path is examined, each instruction at a time. The purpose of the first Flip Flop (55) is to record a condition that is **NOT** met, in other words its purpose is to record a failure. For example if the start button is not pressed (XIC is false) or the stop button is open (XIO is False) the Flip Flop would be set; a successful pass leaves the Flip Flop in its original state. Because there is another path in the program that could be logically true, the successful result of the first Flip Flop must be stored; this is achieved with the second Flip Flop (63) and is the result of a BRT instruction. If the path was unsuccessful, the second Flip Flop remains in its unset or reset state. The BRT instruction now resets the first Flip Flop ready to test the second path.

I order to clarify this explanation the first Flip Flop only sets on a failure and is reset after each path by the BRT function. The BRT function sets the second Flip Flop when a pass has been detected via the First Flip Flop. A path is effectively testing a series circuit and any break in a series circuit would cause a failure. The second Flip Flop is used in a parallel circuit configuration and only one path is required to be successful for the circuit to work. This enables any number of paths to be examined because we only need to record one pass. The CLU effectively tests each path or series circuit in turn. And the first path that is successful is stored in the second Flip Flop. The circuit continues to examine all subsequent paths until a SET instruction is received. This causes the result to be stored in the output circuit pointed to in the SET instruction address.

Conclusions

The PMC was effectively a programmable logic device that could directly replace relay circuits. Its advantages were that it could represent complex AND and OR relay logic circuitry and could be programmed without the requirement for complex and expensive programming tools. The commercial adaptation of the matrix itself was based on fusible-link Programmable Read Only Memory (PROM) chips, programmed or "burned" off-line. Each PROM was 256 bytes and a maximum of 6 PROMs could be addressed, therefore the largest possible program would consist of 1536 lines of instructions (Dummermuth, 2002).

Having only four instruction codes, the PMC was capable of being programmed by non specialist "programmers" such as control and plant engineers without in-depth programming skills or knowledge. One of the main benefits was that the PMC could be directly programmed from a ladder diagram which was familiar to many engineers and technicians.

The PMC differed from the more complex computer based solutions such as that provided by Modicon because it's functionality was fixed. It was a totally sequential machine and could not perform program jumps or sub-routines; it could only perform simple AND/OR logic and program length was limited to the number of memory chips that could be addressed. The main advantages were: it was comparatively inexpensive; it was easily understandable and simple to program; and benefited from the reliability of solid state circuitry in comparison the traditional hard-wired relay based circuits, commonly in use at that time.

Appendix E

Example Interview Transcripts

David Young	**IET, Savoy Place, London**	**19/09/2007**	**pp 353-374**
David Leeming	**Syngenta, Huddersfield**	**30/05/2008**	**pp 375-390**

David Young Interview – 19 Sept 2007

Introduction

Eur Ing David Young is a Chartered Engineer and Fellow of the Institution of Engineering and Technology and has worked as an operations electrical engineer and manager of project engineering within the steel industry for over 41 years. David has many years' experience with the design and implementation of control systems and safety technologies. He is now retired and works as a consultant on power systems (fault calculations and protection coordination), safety, risk assessment, regulation, procedures project management and also acts as an Expert Witness

The interview was carried out at the Institution of Engineering and Technology (IET) at Savoy Place in London on 19th September 2007.

Q. Can you tell me about your professional and engineering background?

David Young: My experience prior to retirement has predominantly been, almost completely in the steel industry. I originally served a steel industry apprenticeship and then went on to study for a degree after that. I had numerous jobs in an engineering capacity most of which have had an electrical involvement in one shape or another. I have been involved in all aspects of electrical and mechanical engineering in the steel industry and I've always worked in the steel industry apart from university vacations when I worked on steam turbine control and instrumentation and design of a.c. commutator motors I've worked in the steel industry until I retired in 2001, that's 41 years uninterrupted, ending up as Manager of Project Engineering and Design for Teesside and Scunthorpe which were two of the main Corus works. I was also Project Manager for Year 2000 compliance of all control and instrumentation equipment in the Teesside and Scunthorpe businesses

I've covered all aspects of engineering ranging from PLCs, control systems, instrumentation right through to power systems and generation [including] distribution and high voltage, covering the whole spectrum throughout my career. I've had a great interest in control and applications though not being directly involved in too much exact detail but from the specification and application [of control systems], that's where I've done a lot. I retired in 2001 and have worked for myself since that time as a consultant on power systems (calculation, protection and coordination), safety, regulation and procedural type activities.
I am also very active in the IET with degree accreditation, membership and Local network activities

Q. Can you tell me something about your early experiences of using PLC and control technologies including computer control?

David Young: Well I can go back prior to PLCs. The steel industry has always been fairly progressive in terms of control and a lot of the processes which were in the steel industry were very much relay logic based. We used to have fairly complex sequencing systems for conveyors, blast furnaces and for steel plants based on relay technology. These systems contained hundreds of conventional electro-magnetic relays, all pre-wired of course and incorporated into control systems to give the functionality as required for whatever the process was.

The big problem was that they were hard-wired panels which were very relay intensive; they were difficult to modify and change. They were very reliable, however they were electro-magnetic components required extensive maintenance, consumed power and sometimes they would fail but so do PLCs: I/O cards fail, processors fail and they are not immune. The big problem was making any changes. The steel industry, like a lot of process control industries was very much subject to change because the plant was designed and installed to do a certain series of functions. However it was not uncommon that soon after installation somebody wanted it to do something different.

One of the first applications I had experience of, which was in the early 1970s was a solid-state logic system based on logic gates. They were based on AND, OR, NAND, NOR and XOR plug-in cards and there were a number of those on the market. **Bistat** was one such system made by Brookhirst Igranic (Cutler Hammer) and there was a modular system by Square D. and a further system utilising "Norlogs" They really incorporated resistors and diodes because they were passive type circuits. Sometimes the more ambitious ones would have a transistor or two on-board to do various bits of functionality but by and large they were passive type devices. Sometimes you would go a little bit further and have a flip-flop or monostable on-board sequence which meant that you had to put some functionality into the system by using a transistor or other device of that type.

Q. Was this like a memory, so you could record the state?

David Young: Rather to change from this state to that state for instance. I came across a lot of those systems and indeed specified, installed, worked and lived with them for some years.

There was another German system that I remember called **"Controlmatic".** They had a different approach but again it was using electro-magnetic devices but it had more solid-state technology on-board the cards; it was a hybrid system and used devices like "polarised relays" for example. Unfortunately they were quite difficult to work with and the schematic diagrams were not the easiest to follow. We used to have a complete Sinter Plant at Redcar, a complex plant which was controlled by the Controlmatic system that interfaced with an analogue weighing system by "Schenck" of Germany.

So that was the forerunner, then the PLC appeared on the market. The very early ones were American based and one of the early ones that I saw in the early 1970s was a Modicon 184. Marketed in the UK by MTE. And in Japan the machine was marketed as (Memocon) It had a central processing package with a separate power pack and I remember it being very ruggedly built, built like a battleship; it had separate I/O as most PLC systems even to this day have.

The input cards in these days were generally 110VAC because that suited the steel industry. 110VAC was the standard control voltage in those days, in the steel industry and still is to this day despite the safety recommendations of having much lower voltages like 48V or 24V but I have seen a shift towards that. 110v ac cards provided an easy interface to the plant

I still know "steelworks engineers that still traditionally prefer 110VAC because that's stood the test of time. 110V is safe: it's 110V with two rails derived from a double wound transformer and the key thing is that one rail (the common) is earthed and all of the coils are connected to that common earthed rail. There is a very good reason for that from a safety aspect. You never ever put contacts between the coils and the common earth circuit because if you get two earth faults (one earth fault can remain undetected), you can short the contacts out.

In the early PLC days we used to take 110V signals directly into PLC input cards, usually 8 opto-isolated inputs in the early days. For outputs we had to interface from the output card, normally a solid-state device, to drive something like a relay or contactor. The early PLCs of course didn't have the power to drive a contactor device directly so we used the PLC output to drive an interposing relay and the contacts of the interposing relay was used to drive the plant. The interposing relays were usually equipped with pilot neons to indicate the relay status to facilitate fault detection.
Sometimes the PLC was required to drive large contactors with 230V DC operating coils. As the coils were highly inductive interposing relays were absolutely essential

110V gave a definite advantage on the output. The logic voltage within the PLC was 12V or 16V [to drive the 110V outputs via the interposing interface relays]. From the safety point of view it gave excellent isolation. One of my favourite ways of specifying equipment like this was to have terminal rails with plug-in type relays actually on the terminal rail itself. This formed a boundary: on one side control (PLC) and the other side connected to the plant; it was a very good interface and is still used today. Those small plug-in type relays were very reliable and it was a very good interface mechanism.

That was the basic configuration and all digital by the way, we haven't got to the analogues yet! We had digital inputs directly into the PLC via optical isolation, outputs generally by relays so we have got segregation between inputs and outputs in the PLC.

The PLC itself was all programmed in ladder logic. It was fairly traditional and probably in the very earliest ones that I saw, about four input devices per rung with an output at the end of rung. These input devices could be anything: normally open [contacts], normally closed [contacts], a latch ... all sorts of functions. You used to build up a ladder network so all the logic built into the PLC device [program] used inputs via the input cards and drove outputs via the output card.

Q Presumably you had memory so that you could store a ladder diagram and record or remember a state, build your own latch for example?

David Young: The memory was sat within the processor itself, it was invisible as far as the program was concerned, no one ever worried about that, it was just an integrated part of the processor. But memory was there so you could put timers in, latches, flip-flops, delays and inversions, even in the early PLCs. But this required a dedicated programming panel, long before the days of having a computer interface. Each PLC manufacturer had their own purpose built and designed "suitcase", well it looked like a little suitcase, a programming panel. It had rows of little switches and lights to show the programming sequence as you went through it.

Q So everything was set up on switches, there were no keyboards?

David Young: Well many had push-buttons but it was dedicated hard type components [switches] for programming the PLC, so this all went into memory. The memory size was about 8K for the first ones I had any association with. You can't put a lot of logic into 8K, even with some of the simple things we are talking about; I think 4K and 8K were about the first I ever saw. The early PLC's were often restricted to approx 100 or 200 I/O

Q This sounds very early on in the development of the PLC.

David Young: Well they were very early, and then they gradually progressed to 16, 32, 64K and so on. I am not sure of the technology associated with the memory, it certainly wasn't flash memory, you used to have magnetic based memory in those days. It had limited memory which stored the program and the ladder network could only be played back line-by-line. You couldn't bring it up to view on a nice PC screen for example and interrogate it over a network. You could only interrogate it rung-by-rung.

The result was that you used to have a print-out of the ladder network, rung-by-rung so you could look at it as a piece of paper; you used to have reams and reams of paper for a large program, churned out from a simple dot-matrix printer. In order to try and find a particular rung, you had to plough through the whole print-out, locate the desired rung and then use the keypad to find the rung on the programming device. So it wasn't easy to interrogate by any means. Program annotation was unknown in these early examples

Q It appears to be biased toward the electrical engineer. Would that be the only person that would program the PLC?

David Young: You've hit the nail on the head. It was designed and configured to suit electrical people who were familiar with using relay logic on ladder networks. So it emulated the ladder network which they were very familiar with for years. So why change it by using something else if that actually "did the business".

It needed some form of storage to retain the program. My first experience of this was a magnetic tape, a cassette tape but not a standard cassette tape, it was a special purpose manufacturer's cassette. So the program was stored on magnetic medium, via the programmer. Of course magnetic storage often presented problems when operating in the steel industry due to fine ferrous particles in the atmosphere !!!

The program could be recorded and theoretically be reloaded back into the system if it crashed, and they did crash. They could crash for a multiplicity of reasons, sometimes when they crashed, the only way to recover the system was to dump the whole lot and hopefully you had the latest program on the tape. If someone had been in on the night shift and put in a modification and hadn't updated the master, it was gone for ever. So it was only as good as was the master store of the data; but what's new, it's still the same to this day!

Even in modern industry, sometimes the latest program for the PLC is in the engineer's top drawer on a disk somewhere. If anything happened to that or if someone during the night shift modifies a program and they don't have software and change control systems, which is a key management technique now, the same situation applies, it's gone forever! We have got a lot better now but in the early days there was no such system. A lot of on-the-spot changes.

But this of course was one of the great advantages of the PLC system. Now you can make changes very easily. Anyone who could find a way through the rungs of the program could very quickly put another function, another set of contacts or another dependency. It was very easy to change and modify and that gave the operators tremendous flexibility to be able to modify the process, on the fly in many cases because prior to that, it was resolved by changing the wiring [hard-wired changes]. And of course it was reliable.

I'll introduce one or two pictures ...

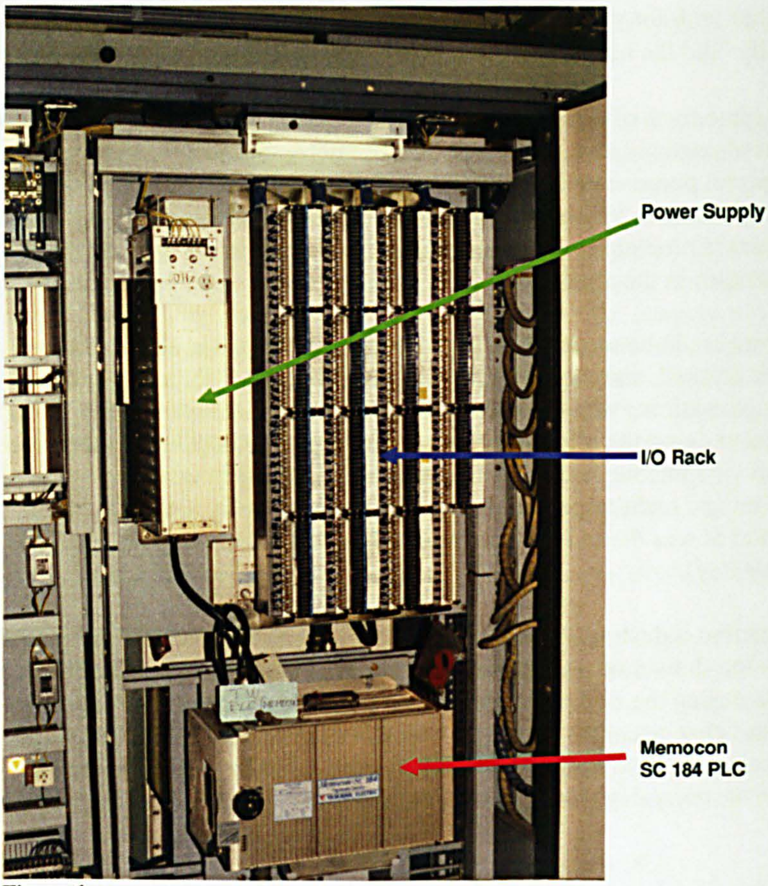


Figure 1

Picture of a Japanese Memocon SC 184 (Modicon 184)

The picture above (Fig.1) shows one of the first PLCs I ever saw, called "Memocon" an *SC*-184 by Yaskawa, but that is a thinly disguised Modicon which was of American origin, it was most likely copied and they put their own name on. The Modicon was one of the earliest processors that I had any dealings with but it was typical of the type of processor unit we had.

We had the rows of I/O as you can see and then we had the interfaces and so-on.
The image [Figure 2] shows the 3 position master switch "Run/Program/Store" and the processor was key locked. This was an important thing because who had the key! These were the real things that caused problems in those days.

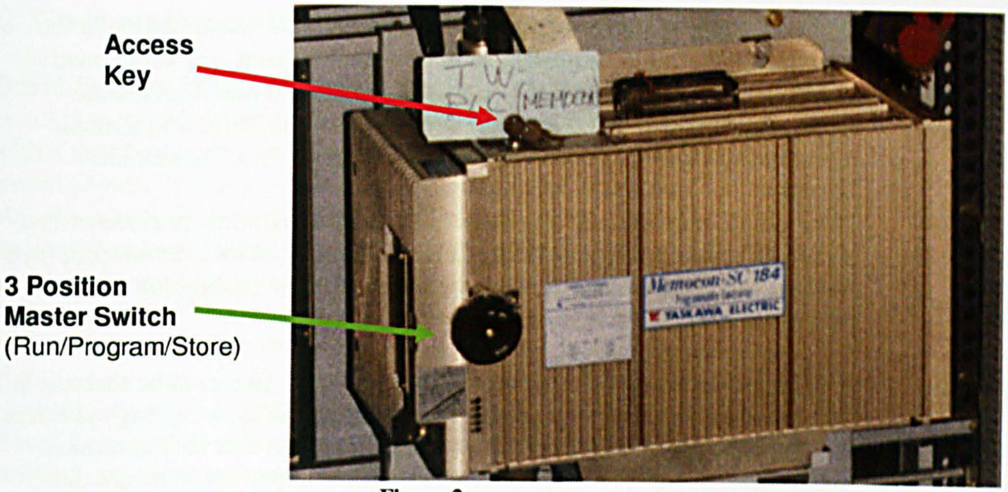


Figure 2

There was one particular engineer, a very clever guy, a graduate who had a talent for this sort of thing and he designed and built all of this virtually under his own steam on a complete blast furnace control system. He had very little support because no one knew anything about it but he was very bright and he designed it, built it and it worked beautifully; this was one of the first PLCs I ever came across.

There were other PLCs at that time, Allen Bradley had the "2" series and then there was the Texas TI range, a smaller type of PLC, about 200 I/O in total that this sort of system would accommodate with the memory sizes that we're talking about. GEC also produced the GEM80s, although they were slightly later. Then there were the tiny "shoebox" PLCs by Mitsibushi and Toshiba having just 16 I/O for placing on a dedicated machine tool or a piece of control system. It was a dedicated little PLC with its own little program box that replaced 16 relays and push-buttons for instance.

There were many of these PLCs and the development was very rapid at that time, something new coming out every week. The PLC manufacturers and suppliers were expanding memory size and the number of I/O cards, it was a very rapid rate of evolution but the technology was still very simple. They were all based on a common design: all had a central processor with a central stack of I/O and a central power supply to run the lot.

Analogues

Q. What was the next stage of PLC development?

David Young: As the PLC evolved, it got bigger and bigger, providing more and more I/O. Then of course analogues were the next development, people wanted to measure analogue quantities. In the steel industry, we wanted to measure things like position for instance, which was a very common requirement. For example a measurement obtained from a variable resistor or potentiometer which was driven by a gearbox shaft for example to give a signal representing position. Other analogue quantities would be temperature, flow or any other type of parameter that was common in the steel industry. But to start with, analogue measurement was very tentative and slow to develop. Position was one of the earliest ones that I came across, mainly from potentiometers giving a position, driven by a motor shaft driving a gearbox which in turn was driving a "roll" or similar

The requirement was that you wanted to know the physical position of that in the PLC and display that on the ladder network. As time developed, then maybe to control position, this is now the concept of closed loop control which was later. **So it was analogue inputs only**.

Weighing was another very common [analogue] parameter to be measured. I came across this in the steel industry during the 1980s, weighing materials via load cells for instance, which was something that was very common. The weighing process always ended up with an output which meant that it needed a signal to drive an actuator somewhere to open a gate or a vibrating feeder, to control the feed of material onto a conveyor belt.

But the first thing was to weigh. Basically it used a load cell in a Wheatstone bridge arrangement with two dummies and two active components, the strain gauges. The output is obtained by exciting across the arm with a fixed ac or dc voltage giving a low level modulated signal. There is nothing in the PLC that can accommodate this low level signal. In the early PLCs it was just the standard measurement signals such as 0-10V and 4-20mA which are the traditional signals. The output from the bridge was in mV so the question was: how do you handle that? Well the answer is that the load-cell would input to what we used to call a "front-end amplifier" or conditioning amplifier which was effectively a signal converter. It was a purely analogue device that from an input of say 0-5mV from the load-cell bridge, gave an output of 0-10V which could be fed into a standard input card for the PLC. We were then able to take analogue signals directly into the PLC and do some [form of] computation within the PLC **so analogue computation came into being then** because you could then manipulate the analogue signals. So there was always signal conditioning in the front end inputting to the PLC and that would be **external to the PLC**. Low signal level thermocouples for temperature measurement and control were also used

We are now talking about signal conversion, so whatever the transducer was at the front end, it would then feed via the conditioning amplifier to the standard input module. Davy United, a weighing system manufacturer for instance used to have a specialism in this and we used to refer to this as the "Davy Front end Amplifier".

Q. Would you have a different design for each application?

David Young: That was a typical weighing application. If it was a position feedback of a drive chain for example, where you have a DC motor with a gear box operating a variable gate device for position, the output would physically drive the slide of a potentiometer. We would connect 0 to 10 volts across the potentiometer and the output from the slide would also be 0 to 10 volts as it drove the output of the potentiometer up. We would now have a representation of the actual position. We could put a command signal [set-point] into that, which used a standard signal from another potentiometer and then do a comparison between the two signals which gives you an error. The output of which can be used to actuate a control system to drive the motor. We are beginning to develop now the concept of closed loop control within the PLC (other means of closed loop control utilising analogue systems have been very common in the steel industry for over half a century)

Previously it was achieved in hard wired format, now we are able to do some of the processing within the PLC. We can put a command signal in the PLC and it let it do the comparison. I remember a blast furnace charging system at Redcar; control of the furnace was critically dependant on where the material was deposited in the furnace itself via a tilting and rotating chute. The control for the chute was via an electric motor driving a gear box and that gear box drove the chute itself. The chute position was determined by a self synchronising system or "selsyn"; a wound rotor three phase machine. The chute drove the selsyn and the problem was that the sensing position was 200 meters away from the where the measurement was taken. In those days low level analogue signals were not very good at being transmitted over long distances so this was a problem. So what we did was use a selsyn transmitter and a selsyn receiver located in the control panel with a three phase jumper between them, the selsyn receiver in turn drove a linear potentiometer. This was a very rugged and accurate link. Selsyn stands for "self synchronisation" and it was a military development from "Magslips" which were used for tracking guns during WW2 and the technology was adapted for industrial control after the war .We used this application to drive the potentiometer which was fed straight into the PLC as an actual position. More sophisticated systems utilised coarse and fine control selsyns with perhaps 10:1 gearing between the streams to provide greater accuracy for small angular movements. The gearing was equipped with mechanical anti backlash correction

We used to have all these electro-mechanical devices long before the days of encoders. Now you would just use an incremental or absolute encoder and feed it straight into a PLC encoder input card. There was no such technology in those days and we were finding ways around these things by using such techniques like this. They were reasonably reliable – there were a lot of small couplings and it only needed the coupling to slip where the grub screw hadn't been tightened then it wasn't given its correct position. People always blamed the PLC although it was usually the mechanical drive components that were wrong.

In the case of temperature measurement – you have a thermocouple where its output voltage is extremely low, certainly at that time not suitable to feed straight into the PLC. We therefore installed a conditioning amplifier which conditions the output signal from the thermocouple to 4 - 20 mA which is a standard signal level used with instrumentation and was fed into the PLC which would accept that signal type.

Q. Were these the main methods of connecting all non-standard external devices to the PLC?

David Young: Yes, in the early days certainly but later we then saw the introduction of devices which would accept the signal directly. You could get an input card which would accept some of these signals directly, you could obtain a thermocouple input card for instance which would allow the direct connection of a thermocouple. Jumping forward, today, you can get an encoder input card for example which allows you to put the digital encoder directly into the PLC. At that time however you had to use an external conditioner.

Q. How was the output signal handled?

David Young: Output was the same, if it was a 0 – 10 volt or 4 to 20 mA output the PLC could handle it. Sometimes you needed a much bigger output to drive hundreds of mA. Servo Hydraulic systems for example would need a servo-valve power amplifier driven by the PLC directly on a separate card that fits in the PLC rack with a separate conditioning card used to convert the PLC output level directly to 0- 250 mA in order to drive a servo-valve for instance, this is again signal conditioning. Now this is where the interface starts to develop with the hydraulic manufacturers eg Oilgear Towler, Rexroth, Vickers and Moog who had a lot of expertise in Hydraulics. They developed hydraulic valves with interface requirements so that they would interface with the PLCs. Actual valve position transducers were also incorporated into the valves using transducers such as LVDT's

The main thing up to this point in time was seeing the development in different Inputs and Outputs. But we were still limited to one processor, one set of IO and that was local IO only and then if you had another process alongside you would need to repeat the whole thing again.

Q. All unconnected and independent controllers?

David Young: Yes correct. In the early days the only way you could get them to talk together is IO to IO by virtue of digital or analogue inputs and outputs. There was no communication capability at machine code level at this stage. We thought it would be nice to get them to talk together.

Cabling was a big problem; there were an enormous number of multi-core cables across the plant to bring the inputs and outputs to the central processor as signals were derived from sources often 100m plus from the processor This was a big cost.

CPUs started to expand [memory] 32k, 64k and so on. I/O got bigger and we started off with 8 inputs per card, then 16 then 32. 64 was too big because of [the difficulty with connecting] external cabling. Things were beginning to be easier at this time but the cabling issue was the big thing and was extremely costly for large remote I/O. counts

The PLC was developing at a rate of knots. Then the breakthrough came. I think it was Alan Bradley who introduced 'remote I/O' – what a breakthrough that was. From the CPU you could have a communications rail that connected inputs and outputs at remote locations. We used to drop an IO rack, out in the middle of the plant somewhere, run one communication cable to it and a power supply and then connect the IO to the CPU as if it were local – Wonderful; a miracle as far as cabling is concerned.

Q When would this be?

David Young: I think it was about the mid 1980s when I first saw remote IO. What I can say is that this revolutionised the physical installation. This also meant we could put more things in terms of displays, control and information. Plant operators wanted indicator lamps, switches and all sorts of things to be driven in remote locations by remote IO. Unfortunately it was not all good news, however, as this introduced the problems of scan time and associated delays in system response

<u>**Programming Unit**</u>

There was a development in the programming at this time because clearly to have remote IO you have to be able to address those remote racks.

This photograph [Figure 3] shows the briefcase or suitcase set:

Figure 3

This is a bit more advanced and has virtually a QWERTY keyboard, the earlier ones were more primitive and just had dedicated push buttons. It used LEDs to display the register contents, so you were looking directly at the rungs, line by line. There was no PC type display.

We needed to have the capability of having proper programming facilities to sit within the PLC itself. This came from the use of early computers and was purely a terminal to interrogate the CPU. The very early ones were DOS based systems and a tremendous step forward from the earlier technology. You would plug the programmer PC into the CPU rack then you could look at the program displayed on the screen rung by rung. This enabled you to program the PLC, address the remote racks and address the IO all from a PC based programming tool which was a revolution at the time.

This was very popular with the engineers at the time. It was also possible to plug the programmer into the remote I/O rack; you could interrogate the system at the plant itself.

Then of course the analogue signal requirements grew and we used to be able to do full process control of plants by the means of the I/O. We had the CPU running I/O racks with more and more inputs and outputs, then things started to happen. Things were now not quite as good; we began to have problems with the speed and response time of the PLC.

At this time, CPUs were not clocking very fast and the scan time[12] was the key thing. You may wish to cancel my earlier comments on this topic? Each cycle of the program, the CPU had to read the inputs and write to the outputs so scan time then became a fair proportion of the processing time. The impact of this was that PLCs started to introduce delays and errors in plant and process control applications. By the time the CPU looked at the I/O and gave a control action, when the action was completed it was round to its next scan, if it was a fast response loop it was too late and the process had gone out of control. This took a lot of cracking to be honest.

Q How was this resolved? Did you take this problem to the PLC manufacturers directly or just wait for the PLC to be improved?

David Young: What you're introducing now is how to manage PLC projects. I started off doing all the design and implementation in-house, well not me personally but my teams if you like. We did a lot of the design and implementation on the fly against not a very good if not poor specification in some cases. It was very much along the lines of basic sketch then modifying the design as we went along.

At that time, we didn't know anything about scan times and it wasn't important in the early days so we got away with it 9 times out of 10. But once we started to increase requirements, fast analogues in particular because they take a lot of memory especially with sampling and I/O scans,

[12] Scan time is the time taken for the CPU to read the inputs and store the input image, apply the program to the input image and the output image and then write the output image to set the actual outputs.

we then started to run into problems. We realised that doing it ourselves as a small team within a department was not the way to do it.

The way to do the job was to write out a specification, a User Requirements Specification (URS) then send this to a series of contractors for them to develop proposals based on that. In turn, the contractors would give us a proposal of hardware and software configuration to satisfy the URS; they would bid for a turnkey project based on that. The turnkey project would be to provide a detailed Functional Design Specification (FDS) which was developed from the URS.

The FDS would provide the detail for the hardware design, the software design, the configuration of the panels and control system including the cable schedules and the price to do all the software to satisfy that functionality **and most important hardware and software testing and simulation** We would use one organisation, a **Systems House** as we used to call them to do the work for us. That, I still believe to this day is the best way to do that, provided you pick the software house correctly with the correct level of expertise, knowledge and capability to handle these things.

Q. PLCs seemed to have developed in order to meet more advanced user requirements, do you think that you exerted this influence directly or was it via the systems houses?

David Young: Well the URS was the key and this was given to the Systems House. What does the user want the plant and process to do and let the systems house give the configuration of how he thinks he can do it. There are no two standard solutions of course. We used to get back a series of proposals all with different system configurations and weighty manuals on bids of how they propose to do the job. It was my job to assess these tenders and proposals and review their solutions. The systems houses had obviously consulted the manufacturers for their advice on how to handle particular applications. Other systems house proposals had gone to other PLC manufacturers. We had to unscramble all of these proposals and select a final solution after having discussed them with the systems houses of course to determine which way to move forward.

We would then select "Mr B's" tender for example, with his technology and configuration then install and work on it. But there would be a glitch or two and maybe it didn't work. This is where I first came to be familiar with the problem of scan time; it mainly appeared when trying to sense something.

For example we had a measuring wheel that was sensing a moving steel plate. The steel plate position was measured by means of a wheel that was raised and lowered down onto the plate and that wheel was driving an encoder which was fed into remote I/O and used to drive an output derived from the CPU. This output was used to drive a motor which propelled the roller table. This system sensed the position of the plate and gave a slow-down position when it was a certain distance from the required position and then to gradually move the plate in until it met its actual required position then issue a stop command to apply the brake. That was a very common thing in the steel industry and still is, known as a "plate positioner". We are often required to position steel, plates or sections to an accuracy of several millimetres for a 30 m plate length

My first experience of this was a dismal failure because the encoder was connected into remote I/O and the motor and brake command was from the CPU and it was over or under shooting. We could never get the accuracy of the positioning and it was purely because of the random nature of where it hit the position during the scan.

We didn't know any of this at the time and the manufacturers weren't sure also. The reason was, particularly with this configuration of remote IO connected to a central, single CPU, that IO speed via the data network was relatively slow. I remember that with our very first application of this problem, we scratched our heads and went on for some weeks before we cracked it. Eventually someone had a brainwave that this problem was scan-speed dependent. The answer was very simple: we put the encoder directly into a CPU rather than remote IO to improve the scan time.

That is a simplified version of many things that used to happen in that era. It was all a question of development, the system houses didn't know the answer to this, they were feeling their way and they learnt more than we did. They then knew for the next time, for the next "culprit" that came along with a job to be specified!

Q. So what you were doing was using a local CPU with the I/O and distributing the processing on the plant?

David Young: Yes, distributed processing. So if we needed some fast response down at IO level we would put a supplementary processor with the remote IO, that is, put some intelligence down at the remote IO level, let that run the IO but still have the communications over the IO network back to the central CPU for orchestrating the whole thing. In other words put intelligence and fast processing down on the remote IO. So speed was a big thing at that time.

Hot Standby Configuration

The next thing that came in PLC evolution (maybe not in the right sequence but there abouts) was in answer to the following question: we are building these enormous systems now, what happens if it crashes? What happens if the whole thing goes down?

People at that time didn't have that much trust in computer type technology. Early computer systems were prone to crashes for reasons that were seldom explained. Things would die, crash or lockout, the only way to solve things was to boot them up again and sometimes that didn't work. Put the tape in (it was before disks), it used to take half hour to load the tape and it still didn't work! So we asked the question, can't we do something about this on our high reliability systems? That led to another concept of **standby** or **redundancy**. At that time various manufacturers: Allen Bradley (now

Rockwell), Modicon and GEM all came out with a capability called "**Hot Standby Configuration**"

It was hot standby as far as the CPU was concerned so you would have CPU 1 and CPU 2 still feeding the same IO rack. The idea was that both were running, communicating with each other and shadowing each other. In theory, if there was a fault detected for example if CPU 1 crashed, then automatically, in milliseconds or even microseconds, it would change over from CPU 1 to CPU 2. Then CPU 2 would run the IO and keep the process going. ie theoretically "bump less transfer"

In practice this was only partially successful. I put numerous systems in like that and that configuration in its own right caused more problems, concerns and failures than ever it did by having a single CPU. Invariably it wasn't the CPU that was the cause of the problem, it was the power pack, the data communications card or defective IO that allowed the CPU to communicate with the IO and so on and it was anything but the CPU because it was very reliable.

In the early days concern was all about the CPU people would say "there's complexity in there so we need to have a hot standby for that". Dreadful, I will never go for that and we had some dismal failures. We threw the things [hot standby system] out in many cases and went back to single processors or at least distributed processors. The fact that to have all your eggs in one basket by using two CPUs with auto hot changeover, as it were called is a totally misaligned concept.

Similarly it was like having a UPS for the 110V supply to the power pack. Putting a UPS creates more trouble than it solves, I learnt that the hard way as well! On a mains failure, the UPS would fail. It was far better just to have the raw 110V supply. This was apparent particularly with the early ones which didn't hold the supply

So that was the hot standby configuration, I remember it well with the blast furnaces that we put this in and it failed; often unexplained failures. I have been in touch with the suppliers to discuss this and asked "have you found the cause?" and "Why did that CPU fail & not take over?" I have not really been able to get to the bottom of the problem. Meanwhile the works manager is asking "what are you doing about the reliability of my plant?" Who would want to be an operational engineer!

HMIs & SCADA

So then the next thing started to come along.

There was a need in the steel industry for remote operator panels out on the plant somewhere. Invariably these would be made up of push-buttons, switches and lights that are fed in through the remote IO. Rockwell came out with a solution this problem called a "Ready Panel". This was a small pre-configured piece of IO and on-board was an 8x8 matrix of keys in blocks and each of these could be switches, lights, selectors and so-on. It worked like an IO rack, it just sat on the remote IO with a 110V power supply and a communications cable and you could configure it in software and physically label it for the lights and switches. This meant a further reduction in cabling all you had to do was sit this on the operator desk and you configured all of the little push-buttons and switches. It was easy to do and I've used dozens of those on various installations.

Then of course plants became more and more complex and we moved away from just the simple control, people wanted something more sophisticated than that. The problem was how do we get graphics and "proper" operator interfaces which you could let the operator have sight of the plant. I think these came in about the early to mid 1980s and we brought in the concept of the Supervisory Control and Data Acquisition or SCADA system. Once SCADA was on the scene, it really started to revolutionise the PLC market at that time.

SCADA is a PC that runs a dedicated software package that you can configure the screens to show all sorts of graphics, for example switches, pumps, conveyors flow rates, change the status of valves and so on, you can have many screens. It's all screen [display] based development, which is carried out off-line then downloaded into its own PC running on a 286 PC for instance. Later more advanced PCs were used such as the 386, 486, Pentium etc, I can remember the evolution of all of that lot. The first SCADA I saw was DOS based and run on the 286 and later the 386 . The graphics were rather crude with low resolution when compared to today but they were there and that was the concept that was laid down.

This brought some intelligence into all this PLC control system and it all plugged into the PLC communications.

The other great advantage of SCADA systems are

- Alarm systems are easily incorporated and alarm management (prioritisation) is inbuilt .This means that hard wired alarm annunciators are not required and precise alarm timing and history is readily available
- Trending of numerous parameters is easily obtained which is very useful for analysing plant performance

Q. How did the HMI and SCADA actually communicate with the PLCs?

David Young: On the early PLCs, IO used to run on a dedicated IO highway connected to the CPU, purely for IO. If we take the Allen Bradley system, they introduced their own highway system called "Data Highway" or the later "Data Highway Plus". That was a high-speed intelligent link between CPUs and SCADA. You could drop off all sorts of things: SCADA system, PCs and all sorts of components. That was the Bradley system, another system was ModBus, developed by Modicon. That was before the days of the universal bus or Fieldbus, for example Profibus.

It was a fast highway and each one had a dedicated communications card in the CPU that enabled the highway to run. I can't remember the actual speeds but we are talking MHz running on the data highway, even the very early ones. This revolutionised communications. That was my introduction to this kind of technology on the dedicated Bradley Data highways or Modicon's Modbus. Each manufacturer had their own. Profibus was supposedly a universal bus, it was anything but! GEC with the GEMs had their own data highway and of course none of these things would talk to each other. There was no universal network so if you had a GEC, Modicon, Allen Bradley (AB) and a Telemecanique TX series, they all had their own systems and they wouldn't talk to each other, there was no universal network to the whole thing.

Q. Did that make you concentrate on a single supplier?

David Young: Yes of course it did and it was very much that, it depended on how good the salesmen were. Whoever got in first to a plant or an area generally put the spike into the ground. So if Allen Bradley were established in a plant, the next time there was a development to do the reaction from the customer was "we are an AB site here, we don't want to know anything about Modicon or Siemens" AB then had a monopoly situation.

This is exactly what happened. Within the works the Blast furnaces for instance were very much Modicon but the Steel Plant were AB and the Rolling Mills were very much GEM80. Even within the same works there was no commonality.

Then someone would buy a dedicated piece of equipment from Germany, it came with a Siemens Simatic PLC of course. Siemens used the symbolic notation and Function Block Programming rather than Ladder Logic. Once you get used to it, it was marvellous of course but for the ordinary electrician and Electrical Engineers on the shop floor, they didn't know anything about it; it was a bit heavy and complex for them, and yet it was a very good system of course and this is what happened.

There were no common themes at all. It was very much driven by the engineer specifying "I want AB because I've always had this" and the number of fights I've had with manufacturers. I was doing this job with some German equipment and the customers on the plant, the production people that I was working for, insisted that it was AB equipment that came with the German welding equipment. I remember being in Germany with the welding equipment manufacturer and the manufacturer said that "we don't know anything about AB" and "we are only Siemens". This was quite a fight that I eventually won and they finally went in with AB. Afterwards they complemented me on this and stated that "Mr Young, you have opened up our horizons by insisting on this, we can now offer for sale our equipment based on Siemens or AB!" Because they then had to learn AB techniques and they then had the two strings which they could then offer to future customers.

That's how black and white it was... and still is. I still come across this attitude, "we must have AB" and "we must have Siemens" and why? "because we have all this equipment in the plant". Even if it was redundant equipment, as old as the hills, the operating system is completely different and I state that they'll have to learn it all again. Ultimately they would agree but this was a hassle, but that's what it drove us to.

Now another very interesting plant that I was associated with, this was in the Coke Ovens in the mid 1970s and it was a brand new plant built by Bristol Babcock. It was used for pre-heating coal and feeding to the coke ovens and it was a proprietary plant, never been built in this country before. We have been pioneers in the Steel industry! There was a lot of process control in that plant: temperature control, flow and weighing. Bristol Babcock designed and specified the plant and their recommendation was that the control system should run on their DCS (Distributed Control System). It was PC based and I'm talking about an early PC system here called the Abacus. The system included full PID (Proportional, Integral, and Derivative) for all the analogue loops When it came, we found that it was very much a PC based system [in that it was not related to a PLC] that had an instruction set which was all through DOS. It had to run in an as horrible environment as you could wish for [coal dust and grimy]. [The control system] was not only running the analogue loops, controlling temperatures but also the digital logic as well all within the PC; it was much ahead of its time to be honest. There was a lot of apprehension about it especially because of the filthy horrible environment that it was in, yet it worked, and it worked a dream. I've never seen computer cards covered in so much filth and grime and continue to work. How on earth that thing ever worked heaven only knows! Occasionally it would crash but invariably we would boot the thing up again, put the tape in, with half an hour to reload and off it would go; we never knew how and thank God it would!

That was an Abacus, and was proprietary system because Babcock were the process engineers who designed the plant and put their own control system in which was purpose designed for their needs. There was some merit in that. What was the alternative? For us at British Steel so say "no, we don't want that, we want all the digital stuff controlled by Modicon and the process control by Kent or by TCS controllers for all the analogues" you end up with a hybrid system but then the responsibility transfers from them to us, Corus. You end up giving them grace to go on and do it, it's [now] their responsibility. It becomes a **Turnkey project, Turnkey technology, influenced to a point but not very much.**

And that was a perfect example but that's one of the ones that worked and it worked very well, very well indeed; I remember the pain and hardship with the thing but it worked. But that was a change completely from the tradition that we knew at that time which is what we've talked about here. It wasn't very easy to access, going through a DOS based screen system and getting into that by people that weren't used to doing it.

Q. Was this was an early move to the computing environment?

David Young: Yes that's right. We had SCADA systems you see, they were all PC based and we went from DOS on the early ones through to Windows at various stages from the early 1990s. I remember that having SCADA systems which were there and running on pretty early Windows, it was relatively user friendly in comparison with the DOS system. People used to like it, the engineers and even the electricians used to think that it was not so bad because it was relatively easy to get in. It was supplied with industrial type enclosures which were ruggedised and indeed a ruggedised mouse, a desk-mounted roller type. We would get some wonderful plants in very rough environments using that type of technology.

The big thing was, carrying the operators on. They had been used to all of the early panels where you had rows and rows of switches, big handles and big levers so they could go with big gloves on and work all this stuff coming straight off the plant, they didn't have to take their gloves off. We must also remember that another new concept for the operators was the insertion of numerical data from the keypad rather than the earlier thumbwheel switches

In converting the operators to work with a screen, a qwerty keypad and a mouse, the only way, which is something that I learnt very early in my career in dealing with this, was to carry the operators with us from a very early stage; form a project team of the engineers but have the operators on board in that team and you let the operators carry it if we're talking about the desk layout for instance. If you let the operator assist with the screen design and panel layout by asking him what he thinks and explain what he wants then immediately he is on your side. This is exactly what it came to when configuring the graphic pages, the operators then design the screens. Once you've let the operator design the screens he's on your side, then he can't then back off. ie delegation of ownership

That's one of the most important things in this sort of configuration is to carry the operator, the guy that's driving the plant, and I don't mean his boss, I mean the guy that's actually running the plant. By and large, if you've got the right and enthusiastic operators on your side, they generally run with this new technology.

If people say "don't give them something like that, they can't..." absolutely wrong, absolute rubbish! I think that we've totally underestimated the capability of operators and people on the shop floor. Give them a chance, the number of things and number of times they've shone and been an enormous help in the design, installation ,testing and, commissioning particularly. If you got operators who are familiar with what is happening out there, you let them design the screens and design the desk layout. Whereas if you come along and say "I've given you that desk and you'll do that now" the operators will just let them mess up!

It works wonders, that's something you really do learn the hard way and it's easy. We've had operators who've wanted [or have been able] to reprogram SCADA systems. We don't let them do it because this is where the concern of software control and change control comes in. We've made big strides in that but we're not home and dry yet. When I left this type of work, we still have a long way to go as far as control of change is concerned and documentation management.

Safety and Security

Q. Would you only let someone who has a certain role or authorisation make changes?

Yes it was password protected and only allows the people that had an appropriate password; we had three or four levels. But if there's anything which is affecting something which could have a safety implication, then that's the highest level of all of course. That is something which needs to go through very rigorous change control procedures. That is before it's done, it's not done on-the-fly. It's done with very careful thought and consideration before any change is put into place for very obvious reasons if there's a safety implication.

I've never accepted that fact that PLCs could be relied upon for safety. So **always, but always,** emergency stops and final safety devices such as final limit switches, ultimate l temperature switches, whatever were hard-wired into high integrity relay systems. Safety systems were subject to Risk Assessment and configured to provide the resultant required Safety Integrity Level in accordance with appropriate British and International standards The safety systems were completely separate from the PLC but probably monitored by the PLC. Each individual emergency stop probably had an auxiliary set of contacts taken into the PLC so that you can see on the SCADA system what the status is of that but the actual stopping was hard-wired. And that's still a rule to this day.

Safety Systems

Q. What do you think about new safety systems like Profisafe and safety PLCs? Are you not convinced by that?

I've looked into it and provided that it's engineered and specified correctly it's OK. If you're familiar with the standard IEC 61508 and its sector derivatives and you go into the SIL Safety Integrity Levels, level 1 and level 2 are alright but once you go into level 3, you've got to be very careful. They are available and it can be done, I mean Pilz have a complete solid-state PLC system a complete safety PLC for instance. It comprises 3 independent separate processors by 3 separate manufacturers. So they've got an Intel processor, a Fujitsu processor and a Motorola processor for example so there is no common mode failure in there. Each with their own instruction set inside and arranged in a 2-out-of-3 voting system with redundancy. ie A Triplex System as used on the Airbus "fly by wire" system That's approved by the Health & Safety Executive for safety purposes. Personally, I haven't used one.

Q. You're still happier using hard-wired systems?

David Young: My days in using this equipment, when at Corus dates back to 2001, if I was to be involved now, I would look very carefully at this to see what the market had to offer. Because I have done some work with Pilz since I retired, I learned a lot about this system and I've got a different view now compared to what I used to have. So those systems are available and they are safe and you don't [necessarily] need hard-wired emergency stops. You can still have duplex systems if you like and they do have the safety integrity levels which you need to have, namely SIL 2 and SIL 3. I wouldn't go so far as to say SIL 4, but the only two real applications that I know of which are SIL4 are in the railway industry for signalling the nuclear industry and perhaps high risk petrochemical plants I certainly haven't come across SIL4 in the steel industry.

So, in my book, emergency stops, final stops and final devices are best kept hard-wired. Maybe that's an old fashioned sort of [point] of view but this philosophy has stood the test of time!!! [*tape unclear – coffee being poured!*]

Software

It must also be remembered that the all programmable devices require **software** as well as hardware for their operation. Preparation of high integrity software is notoriously difficult and despite extensive testing and simulation, "hidden failures" can remain undetected. Solution of this difficult problem requires rigorous procedures incorporating Validation and Verification by independent persons/organisations (Depending on the SIL level) the requirements are prescribed in detail in IEC 61508 which is "The Bible" for all programmable systems

In my current IET Degree Accreditation activities I always make a point of raising these important concepts during accreditation visits to Universities. The response is frequently rather embarrassing to say the least !!! **ie the importance of teaching software design for safety applications**

Q. I suppose you know the circuitry is based on proven principles, is it a big leap to trust that functionality to a processor?

David Young: Well it is and it's verifying [it] you see, part of the high integrity system is verifying that it works when it's called upon to work. So I've introduced in the past functional testing of emergency stops and other safety systems for instance on a periodic basis. A physical functional test is done and I don't mean a test by just looking at it, I mean physically hitting the emergency stop and making sure that it actuates the final stop element, i.e. the contactor or relay and that's a physical check. OK the next time it operates, it might fail but there are ways and means around that. Are you familiar with cyclic monitoring of emergency stop circuits? It automatically runs through a cycle every time it works and if it doesn't go through its cycle, it locks out. That to me puts a higher level of integrity in.

Safety technologies still have some way to go and is still in its infancy really. Mechanical safety systems are of prime importance and that any electrical/control safety systems are in addition to and must not replace mechanical devices.

Addendum

The rest of the interview was based on notes and briefly covered the following topics:

Variable Speed Drives – PLCs are not fast enough and do not incorporate the required functionality for direct control [Drives incorporating dedicated microprossessors are invariably used]. PLC interfacing for input and output commands, however, is invariably provided

MMIs (HMIs) – provided simplified panels [in terms of design and build?]

Soft PLCs and PC control – good future and very flexible. People are now much more familiar with Windows systems and programs.

Future of PLCs? – Days are limited; there will be more PC control. Systems can talk to drives etc via Fieldbus. There will be more PC control and young engineers are more computer aware and literate. Interfacing to high power output devices is still required

MW – David, thanks for your time!

David Leeming Interview 30 May 2008

Location: Syngenta Huddersfield manufacturing Plant

MW: Could you tell me something about your background?

DL: I started in 1968 as a Control and Electrical Tradesman. The company operated a scheme to sponsor people to study for a degree if you had the aptitude and I was lucky enough to get sponsored. I graduated in 1975 and started working directly on the plant, moving into design and then projects. I then moved onto the start-up and commissioning of effluent treatment plant at Huddersfield. From there I moved to Stevenston, Ayrshire and worked as Control and Electrical Maintenance Engineer for the Acids and Nitro-Cellulose Department.

That was a Sulphuric acid and weak and strong nitric acid plants. We made nitrocellulose, dehydrated it and rendered it is safe in an industrial plant; we also make granulated nitrocellulose as well. I spent about five years on the maintenance side doing plant based projects to improve performance and that sort of thing. That was just that time that digital electronic technology was coming in. I then moved onto projects and became the site project manager, and also did projects at other Nobel's industry explosive facilities around the country. We did one in Wigan, *Penryndeudeath* in Wales, one in Nakhorn Sawan Thailand, and we also did some work for the Royal Ordnance in Kidderminster and Bishopton. We actually designed a computer system for the nitro-glycerine plant at Bishopton, Glasgow.

After projects, between 1989 and 1993 I moved back down to Manchester to move on to bigger projects working for ICI engineering. I worked at Grangemouth, Teeside, Huddersfield, Macclesfield, various assets across the country. I was still attached to explosives until about 1992 as well, so I kept the major project link to there and I also looked after the people side of things too.

In 1993 I came back to Huddersfield and looked after an area of the site (Colours) and also as the site electrical engineer. So I did two roles really: the responsible electrical engineer and looking after all of the people and the performance management of 50% of the Engineering staff. I gradually took on extra bits and eventually looked after the whole of the factory and the project group. The company was split from Zeneca to Syngenta and **Avecia** and at that point I became the Syngenta site engineer. I have operated as the site engineer for the last seven years or so and now am the **Global Head of Compliance Engineering**.

The sort of thing that runs through pretty much from the Nobel's explosive days CV, is a line that talks about restructuring and reorganising. If you look at one of the main skills I have, it is probably about restructuring and reorganising people and systems to meet the site requirements.

MW: So you have obviously seen a lot change and you've introduced a lot of change then?

DL: Yes, I'm pretty comfortable with change really. I've introduced a lot, in fact it gets a bit boring when there isn't a lot of change about!

MW: Can you tell me about the application of control systems in the agro-chemical industry.

DL: Well in the agro-chemical industry the scale of the plants are large. Particularly in what we call the active ingredient plants, the high volume throughput plants, multi-stage batch operations generally. Occasionally there is the odd continuous plant. 90 to 95% of the plants are DCS controlled. They are mainframe computer operated to get that reliability and continuity to make sure it's the same product batch after batch after batch. The DCS was used to remove the variation you can get with 5 shifts and therefore 5 different ways of operation.

Looking back at my Nobel Explosive experience, the plants tend to be smaller, sometimes very much physically smaller because of the hazard, which is so much more intense. Those tended to have the mid-range PLC. They would have had the Negretti type systems which were able to handle analogue measurements than normal PLCs at that time. As I remember, we only had one mainframe computer and that was on a PETN plant. That used to have a mainframe computer because that was beginning to get to the scale of the kind of plant used at Syngenta, Huddersfield. Everything else was either manual control room type things with discrete instrumentation or PLCs on different small operating units just to give a localised level of automation.

MW: Is it largely the scale of plant that would determine the selection of the control technology used?

DL: You tend to find PLCs on packages, on small units and where we've got older assets you might find PLCs controlling the more critical units. If I had taken you to a formulation and packing plant, that we don't have any in the UK now, you would have seen a lot of PLCs because they were a fairly standard design.

MW: What control technologies were used prior to these automation technologies? You mentioned pneumatics?

DL: Yes this goes right back to the early 1970s. We built plants here at Huddersfield around 1974 and they were fully computerised with Ferranti Argus and Kent systems. I do remember my standard check [question] with students was "guess the size of the memory?" it was 24K! Smaller than the first PCs almost!

So we were on mainframe computers from about 1974 but that was again with the scale-up, the scale of some of these plants was huge, 1200 Input/Output (I/O), valves analogue inputs and that kind of thing. On a smaller plant, prior to PLCs the control strategy would have been control with discrete instruments. A bit of SCADA, I think SCADA almost passed us by to be honest. We were either in mainframes or sometimes we added SCADA onto PLC systems but not a lot. It didn't really fit with the strategy at the time.

MW: What are your earliest experiences of using PLCs?

DL: *We probably introduced PLCs around about 1980 and we were replacing relay sequencing units and that kind of thing, we learnt very quickly that these things can fail very differently to relay systems. We* realised that we had to monitor the health of the PLC using for example watchdogs. Also, that you didn't put your final safety integrity trips inside them because you didn't know whether they would operate as they should operate. For example, an ideal installation for a PLC at that time might have been a lift, but you would have to make sure that you had separate integrity systems for the emergency brakes and things like that. I believe now that there are some high integrity PLCs on lifts.

MW: Was safety was a big factor in your choice of technology?

DL: We did have some incidents with some of the PLCs where we had reversals of all the outputs; what should have been on was off and what should have been off was on! Some very obscure faults came up. We were very good at reporting in ICI, we very quickly learnt and collected the information together and came up with a view on how to implement and use PLCs. This view probably said "be a bit more conservative about it, yes it's fine for running sequences but don't put your safety systems on them".

I think that that was where we were and we used them a lot where we had timers and relay sequence type things and replaced them with PLCs, the first ones being the Texas 5TI. I think we also had some Allen Bradley PLCs because they had a bit more analogue and control functionality, the 5TI was purely digital "Ins and Outs" and fairly basic too to be honest, but they did the job.

MW: I think Siemens brought the Texas Instruments PLCs and re-badged them as the S7 200 range.

DL: Yes, that's right

MW: There have been attempts to make the PLC more robust such as the Safety PLC, also other techniques such as redundancy, using redundant processors. What are your thoughts on those?

DL: Well we've done that once or twice in my career. I avoided them like the plague where ever I could. I can remember the triple redundancy things but frankly if it's the same machine, it could have the same problems with its operating software, so we didn't. We built a high integrity control and protection system for some propellant presses for Thailand and we used two types of PLC, one was an Allen Bradley and the other was a TCS (Turnbull Control Systems) which was really a digital controller. We had them communicating with each other with external watchdogs on them. The beauty of that was that they both had different software and different operating systems, so we were not going to get a common mode failure if they had both been the same. So we avoided Safety PLCs to be honest. That was one of the more complicated systems we've put in.

MW: What about plant signals, you have tended to standardise on 24Vdc. One topic that has arisen for improving the performance and reliability of PLC signals has been the use of Uninterruptible Power Supply (UPS) systems to back up the 24V supply.

DL: Yes, you may have noticed in Paraquat we saw the UPSs. The power supplies are all backed up, even the 24Vdc, everything is backed up. You will have noticed all the dc power supplies that there are at least two and they can operate with one on all the racks. We will have had hazard studies which we now call PRAs (Process Risk Analysis) not just on the equipment and the hardware on the plant but on things like the racks and I/O allocations to make sure that you don't create, a situation if a card loses all its power forcing a range of valves to close or open.

MW: Back to early memory systems such as the Ferranti Argus core-stores, what memory and back-up systems did the early PLCs and control systems use?

DL: I can think of two types, both different. The Negretti tended to use E^2PROM or EPROM. What we tended to do was blow the memory, once we had programmed the machine, burn an EPROM, take it out then burn a second so we had a backup set of EPROMs then. The second generation of those was the E^2PROM which we basically under a UV light and erased them. With the devices [PLCs] like the Texas 5TI it was very simple, basically they were backed up by simply keeping the relay logic diagrams.

MW: Just the print outs?

DL: Yes, about as simple as that. The logic was usually very simple and they could be easily and quickly reproduced. On the TCSs, we would extract the settings and store them in a database.

MW: How did you control the software?

DL: We tried to do the same things that we did with the Distributed Control Systems (DCS) which included version control. We would look at the systems once a month and do any necessary housekeeping, backups, modifications and so on but also refresh the backup as well [reload the backup soft/firmware]. So the routine housekeeping would be to back up the tape, refresh the E^2PROMs and that sort of thing.

We were a bit more relaxed with the PLCs, about once every 6 months, on a shutdown or when equipment needed replacing. Harder to control, particularly the TCS because the technicians to could get hold of the programming unit and actually change the settings. We tried to encourage them to understand these things [systems] and we tried to make sure that certain settings couldn't be modified. Because none of these settings were safety related, to a point it didn't matter, everything safety related was not on that kind of system.

MW: So people did have access to the programming units and to the software and were capable of changing the software themselves but they weren't the critical systems.

DL: Yes, that was more on the control side. On the digital in and digital out we tended to keep the programs within the professional engineers. Then it's a case of providing the equivalent ladder logic, they would fault find using that. If they struggled with it they would have to come to us [professional engineer] if there was any problem, which wasn't often.

MW: There are mechanisms on some of the control systems that stop people from changing the programs but they could monitor. Did you use any of those?

DL: Yes, on the Texas 5TIs they couldn't alter anything, they were like "black boxes".

MW: You've mentioned the Negretti MPC 80s, Allen Bradley and the Texas 5TIs, did you look at other PLC types?

DL: Yes the Negretti MPC 80s, 84s and 85s, we had a full range of them but like most companies, we tried to standardise a bit. However you ended up with creep [unplanned diversification], because if you dealt with a French company for example, you would find a Telemacanique PLC installed in the equipment that you really couldn't change. Sometimes because their design was around that particular system and if you asked them to put something else in, it would cost you an arm and a leg. The maintenance of the equipment by somebody else would just make it more expensive and difficult. So on packages we accepted what came in with it. You get a lot of simple packages that use the simple PLCs such as the Omron, which had ten I/O controlling the switch-over of air dryers for example.

We tried to control it, for the sake of the site to keep to a particular standard. So we started with the Texas 5TIs, then when we wanted some analogue and digital we would have gone to an Allen Bradley. But at the same time the TCS family was increasing its capability from process control to having digital processing as well, so we were able to go both ways. It would depend on whether it was more [process] control or more on/off [digital control]. If it was more on/off you would go the PLC route or more control you would go the TCS route.

MW: Did you apply the same approach to DCS as well?

DL: Yes we did standardise with DCS. This site, Huddersfield has always used Kent. An excellent system when it came to batch control, batch tracking, batch handling, it was miles ahead of the competition. The competition was designed for continuous plant such as refining and [the Kent] batch mechanisms and understanding was really very good. We had two generations of Kent: the K70 and K90. We then went to an in-house designed system by ICI which we are now replacing with a Siemens system [Siemens PCS].

This site chose Siemens but another site chose Delta-V. There are some benefits in having diversity in suppliers; if one goes bust you are protected from a business risk perspective.

MW: Did you look at the Siemens PLCs?

DL: Yes we did, but we thought that the plant was too big for that, in view of the work that the plant had to do.

MW: PLCs tried to break into the process control market. Did you consider PLCs to control the plant?

DL: They did indeed. You can picture a plant like Paraquat with a PLC on every floor connected to a master PLC in a control room. However it's a hazardous area, when you think about plant design it [the distributed PLC architecture] becomes quite difficult and what you really want is everything in a safe area. If you are going to put it all in a safe area, you might just as well have the mainframe computer [DCS]. So we never really went down the track of using any big or massive PLC systems.

MW: A consequence of that is that you have to use a centralised control model where everything is wired back through isolation in a marshalling cabinet/area.

DL: Yes but you've not just got that, you have got all the control as you say but you also have the operators in a safe area, everybody has to go to that area to get the permits, you control all the work that goes on the plant from that area, you padlock everything off from that area. It means that there is a "hub of knowledge" about everything that goes on with the plant. The more distributed everything is around the plant, the less control you've got.

MW: That's an interesting concept. In some other industries, they try to distribute the control throughout the plant. Whereas you are trying to bring all control back to one point or area.

DL: Yes absolutely.

MW: How do you alleviate the risk of a single point control system failure?

DL: We use redundant processors and systems, so if one of the processors fail, the other one just kicks-in.

Programming Units

MW: With regard to PLCs, what programming tools have you used?

DL: I remember the Texas 5TI, you used to get something that looked like a very large calculator that you used to plug in and program [the PLC] from that. You could store the program and download it to a computer, just a normal PC, a very early PC like a Commadore so you would save the programs on floppy disks or something like that.

With the Negretti, you had an engineer's terminal. You had to enter an access code and once you had access, you could make any changes that you wanted to the software. This was the same with the Kent systems, you had some sort of access code, and unless you've got that code you can't make any changes.

MW: Was the engineer's terminal fixed or portable?

DL: It would have been a fixed terminal that would have given some rudimentary and fairly crude displays to the operator. The first ones were black and white, well green and white. Some of the things you would set up for something like a process with six stages, you would have some sort of text display (e.g. Stage 1 .. 6) with some comments saying "complete" so you could see where you were. You didn't have the kind of mimics that you have now.

MW: I suppose the fact that you had centralized control meant that you could have fixed displays.

DL: That was the difference really. In the same age, the early 1980s, you had sophisticated mimics that gave you the status of every valve and so on but with the PLCs and Negretti's you couldn't extract much information from them. You could look at the PLC input and output LEDs and connect hard-wired mimic panels using lamps and switches but it wasn't until the late 1980s when they started bolting a SCADA system together with the PLCs and offering one package that could do a bit more.

MW: Your processes appear to require an amount of supervision and intervention. Is the availability of process information a key element in controlling your industrial processes?

DL: Well things can change, we test every material that comes into the factory but there's a [material] specification; some things could come in slightly weaker but still within the specification but that might affect the reaction. So you have to understand the key parameters, you have to monitor and watch them so that you make the right product.

MW: So there's a lot analytical interpretation of the process that people have to make.

DL: We have a lot of technologists on-site that keep their eyes on the processes and make sure that the processes remain stable. Because for example, contaminated raw materials and things like that could have a big impact on what comes out. We now have processes that are mature, robust, well understood and the people know how to operate them so it's not too bad now.

MW: One of the interesting differences between this industry and others is that there is a much greater involvement from operations staff, in terms of monitoring the process because of variability's. Does it subsequently require direct human interaction into controlling the process?

DL: Well you do everything you can into making it as non-variable as possible. For example on the Paraquat plant, you have things like "measure vessels" so you know that every batch gets the same quantity of material. But of course if something happened in the previous stages which is not quite right, that will carry on into the next stage and it does need a good analytical eye on it.

As you've seen, we've implemented the Active Factory and we can view that in the office, just as easily as you can on the plant. It tells us everything that we need and all the key parameters are delivered to it from anywhere on the site.

PLC Technology Developments

MW: What do you think are the most significant developments in PLC technology?

DL: If you go back to 1979, the underlying technology was the mainframe [DCS] which was coming in at that time. They were pretty expensive so you had to have a big plant to do it. We were installing them Stevenston but the major installed technology was pneumatic. Two things came in, one was the PLC which allowed us to get away from huge relay and timer panels which were complicated, expensive and in some places dangerous. Then you got the [process] control side which came up very quickly then. I think the biggest development that came in was when we got the digital control systems, the small ones like the TCSs. They came in with much more functionality than the pneumatics or even the analogue electrical controllers. They allowed us to do a great deal more than you could with anything else at the time.

I can give you a good example: We converted a pH control system which essentially other than the pH measurements, everything was pneumatic. This had 21 components and parts to it. We changed this system to a TCS; we could even put a pH curve into the TCS. We ended up with about four components after the conversion. We took 21 components and pneumatics to full electronics so you can imagine the reliability benefits and improvements. That to me was pretty significant, it brought precise digital control to the discrete level, and you didn't have to have it at a major plant level. After that, it was a case of deciding what technology fits the unit size operation.

The next thing after that was, I believe, the PLC manufacturers realised that if they didn't introduce analogue and control algorithms into the PLC they would either be dead in the water or they would have a very limited market. PLC manufactures then started to come into that area but I think they were always weak in the analogue and control side and they always had their strength in machinery, guarding machinery, production lines that kind of thing. In an environment like a chemical factory, there are no production lines, you're pumping stuff from one place to another and mixing it with something else. You probably would always pick something that was more able to handle measurements rather than just digital.

You are much more likely in this environment to see PLCs on package deals: compressors, fridge units, air handling units, all fairly standard equipment. Alternatively, small units where there is digital and analogue and there is the right functionality to suit a PLC.

MW: What are your general thoughts or opinions about remote I/O and Fieldbus?

DL: We just never went that way. The selection of remote I/O was probably hindered by flammable atmospheres. If you use remote I/O, you have to put into a safe area or you would have to find somewhere safe to put it inside, a flammable enclosure for example but if it went inside a flammable enclosure, you could not work on it. You would have to write a certificate of safety to work on it and you would be working on something that was not certified for the hazardous area that you would be working in then. I think that we were always disposed to locate it in a central area; keeps it dry, keeps it clean, keeps it well maintained. Fieldbus was just in a too difficult box because there were too many variations of it and we never saw a standardised version even though the company I was with (ICI) was part of the development of Fieldbus.

There were just too many arguments about it, like the Betamax and VHS video formats, we were just never really confident about it. We were confident about the way we designed plant and we knew what we were doing so we stuck with what we knew about.

MW: so I imagine that technologies like RF networking would be a no-go area?

DL: We would never be a fist adopter on something like that – no way! I think that once something was established and became a norm, we would try it then. However, when you are handling chemicals and particularly the types of chemicals as we do, you cannot afford something to go wrong.

You try hard to standardise the equipment that you purchase. There will always be somebody's flow meter that you need to buy as opposed to the standard supplier then suddenly you're into some form of conversion, you have to use "4-20mA" standard signal on that because it won't talk. So we just never went that way.

MW: What do you think were the main shortcomings in PLC technology?

DL: Well I think it was really that they were limited in size and the PLC manufacturers approached it from the digital end. In our industry it's much more about measurements and less so about digitals. We do a lot of valve position sensors and checks such as pressure switches so we still use a lot of digital I/O but there's probably more analogue, much more analogue and these machines [DCS etc] are built around that kind of data processing.

MW: Some of the DCS systems use a PLC or a subset of a PLC type of control within them.

DL: Yes and the output cards would be a PLC output card.

MW: What do you think are the main benefits you have realised by automating the process?

DL: Well you get consistency without a doubt. It allows our technical people to run trials relatively quickly. For example "Do we really need to stir this for 10 minutes after the mix or could it be 9? Let's just go in and try that". So we frequently run trials runs, it's dead easy to come in and make those kinds of changes.

Every computer project that we've ever put forward has always been on the basis of batch on batch consistency. It can be consistently bad as it can be consistently good if you've got it wrong. You can optimise the process then from that point and we frequently do that and we frequently get more out of the plant than it was ever designed for.

MW: Do you think that the automation technology itself has helped you with your Manufacturing excellence programs? Has it been a contributing factor?

DL: Well, we've talked about the things that have moved things forward... The change from pneumatics to PLCs and TCS technologies was a major step forward and mainframe computers. I think the next main step forward was Active Factory [MES] because it puts information to everybody. It wasn't just the plant controllers, this information could be anywhere round the world now. You can literally look at anything, any piece of data you want. You can correlate any data that you want and for the technical people in offices and labs that support the processes, it just opened up a completely different mind-set.

On a particular plant suffering production problems, we used to have a weekly meeting. The technical people on the plant after a few weeks eventually started bringing in best-of-the-best data, classic lean manufacturing concepts. "Well this batch, the best we've done is 3:30 hours so let's keep it at that every week. Then we started getting best-of-the-best from each process step and this was from extracting data from the system which we hadn't seen before. We couldn't see it before because people were filling manual Process Instruction (PI) sheets and to go through those to get that kind of data would take an army of people. So you really started to get good quality information. You turned your data into information and then you started to act on it.

This was a huge benefit; it has completely changed the way we deal with information. We've created "dash-boards" for every plant. We call them dash-boards because they show the key parameter information, like a car dash-board. Not everything is displayed but the temperature, pressure and the things that are useful. It's not usually expensive either, if you look at what we've paid to do it, it's trivial compared to the size of the plant. The information has always been in the DCS, sat in the back of the controllers but they weren't "wired up". But now we've "squirted" it into the IT systems, what we can do with the information is amazing.

MW: What applications would you use select current PLC or DCS control systems for?

DL: It depends on the scale of the plant, but here would be the big systems but on the smaller plants, that maybe had 10/20 vessels then your probably into something like PLC but again, something capable of handling the analogues signals and could communicate to the IT systems.

I think whenever you need process control, that's when the big system comes into its own. Steel works, refineries that sort of thing, big complicated processing units. I see PLCs as more the formulation and packing facilities where you have lots of machinery guarding with many inputs and essentially a few outputs to run. PLCs come with the machine, whereas, we have to make and program the DCS system to run a machine which is considerably bigger.

MW: What systems do you have in place for <u>software control and management?</u>

DL: We have pretty good controls now. Whether it was Negretti or these kind of systems, any time we make a change we will log it, give it a version control, take a back up and retain the back up in a locked place. The backup would normally be a fire-proof safe on-site. They will probably do the housekeeping at least once a month, I can't imagine they will do it any less than that.

MW: so it's very rigorous in that respect

DL: Yes.

MW: In terms of project management what are your procedures and processes for selecting control systems?

DL: We write a fairly detailed URS ourselves. We write a URS that helps get the architecture right and gets the input/output configuration right subject to our hazard studies. What we will do is our guys will take the old flow charts and go through that with the operators and the plant guys and come up with the flow charts we require which will hopefully be similar to what we had originally. We try to encourage no change when we are doing a control system change-over. If it's new, we would go through the same sort of process and develop some charts. We then work together with the supplier to write the software, the bespoke software for us and then we would sit down and commission it with them.

We have a simulator for the Siemens system so we can operate the simulator and use that to train the operators as well. We can test all the sequences on that simulator too.

MW: Are you heavily involved in developing and writing the Functional Design specification (FDS) developed from the URS or do you leave that to the supplier or systems integrator?

DL: Yes we do write the URS and the FDS because we don't feel that they understand the chemical side of the plant enough. No matter how detailed you write the specifications you can't get that bit over. So the final coding for the sequences in the software has to be a joint effort. We bring the knowledge of what we want the plant to do, they bring the knowledge of what the software can do and the two come together.

MW: So you have to have a very close working relationship with your suppliers?

DL: Yes, it has to be. Siemens are based in Manchester and we have people going backward and forwards from both organisations between these places.

MW: So you have complete control and understanding of the [software] development process.

DL: Yes absolutely, and we are doing the code changes for modifications ourselves, once we have the machines they're ours. We have a full modification control system for authorisation, approvals, sign-off, coded, added, tested and closed off as a modification. So we have a good control of that as well.

MW: I can see that the modification process is influenced by professional and technical good practice but Is that influenced by any external bodies?

DL: Yes, from regulation. The regulators have high expectation from top tier COMAH sites. So we have to prove that we are well in control of the plant and processes, which we are.

Security

MW: PLCs have mechanisms to restrict access to the means to alter programs, how do you manage this?

DL: We tend to have a route to an engineer who can do something, other than that the operators can't do a thing. They can't change alarm settings, control settings or anything like that, because in this environment, if you allowed that it's essentially a modification. In theory, if you allowed complete access like that, they might turn a set-point or something to above a vessels temperature capability, so we don't allow it.

Safety Systems

MW: I noticed on the plant you have alarm annunciator panels which are separate from the process control systems. Why is that?

DL: We will always keep our safety systems separate to computers. But they do feed into the computers to tell it that a safety system has tripped. What that does is take the sequence to a "sequence hold", shuts or opens valves dependent on what the safe state is. It [effectively] takes the sequence to "abort" basically and then it would have to start-up again. But it's the hardware that drives the software then, the hardware tells it to do something because it's tripped.

We do now use PLC trip and interlock systems, the ones that are approved, "TUV" approved and people like that so they have a high reliability.

MW: The so-called "Safety PLC"?

DL: Yes, but they are more safety modules as opposed to PLCs. They almost look like blocks or relays that you connect together and we've used those.

MW: You also use mechanical safety systems on a vessel, what about the emergency stops?

DL: Yes, mechanical systems for pressure relief but things like emergency stops are not used through safety PLC systems and go straight to contactors; they cut the power straight off a contactor. Even if it went through to a computer, you would still put the emergency stop straight to the contactor. Even if the drive was computer controlled, we tend to always do that. We put everything so that it physically breaks the power to the contactor regardless of the state of the electronic or computer device. For example if the card failed on to a computer it wouldn't matter.

MW: Would you ever consider using the new Safety PLC or safety module systems for your emergency-stop systems?

DL: We went through a period where we were looking at types of relays and some motor controls. There was some legislation that came out about 10 years ago, and some of the ways that we were running our computer plant we felt went against that. We had to put in some memory relays to make sure that we had the right integrity. It was about the computer connection to drives if I remember and we changed the way we did it to comply with the legislation.

In terms of using a safety PLC system for emergency stops though we would always try and break the master line and have it as an electrical direct-in-line circuit break.

MW: What about using Fieldbus based safety systems such as "Profisafe"?

DL: I think we would be about the last to use that! What we are handling, the processes and the hazard studies that we do would say that there are so many options of failure here. We have got the data for the systems we use and we have the data from the nuclear industry so that we can prove that the relays and devices have the reliability. For example this will fail once every hundred or a thousand years, we have the data to back that up. But you can't for some of these other things such as safety PLCs because they don't have that historical data.

MW: What are your thoughts on emerging technologies like "Soft PLCs"?

DL: I suspect they will end up in the packages, package deals and we won't even know it's appearing.

MW: Do you see it as something could nudge its way into the DCS fields?

DL: I'm not sure to be honest. I think that the DCS field, what it does and what it does it on; it does it on a big piece of hardware and the software needs to match each other exactly and the scale is fixed, I just can't see it creeping in to be honest.

I can only see it appearing on things when we don't know it's there. It was like when I showed you the "Fridge" this morning. We know it gives us certain indicators [information], if you open the door it has a PLC but nobody would know or bother about it, it's just there.

MW: What about Process Automation Controllers (PACs)?

DL: I suppose that's the TCS end of the technology, the next level up from them. You don't really get much more functionality with them also.

I can't see it changing a great deal from where we are now. We had some massive changes in the 70s and in the last decade with data and information, those are the two big changes as I see it.

MW: What do you think that future is of PLC technology?

DL: I think it will continue quite strongly in package deals. Cigarette making machines, formulation and packaging machines and stretch wrapping machines, where somebody buys something fairly complicated with inputs and outputs but it does the same thing time and time again. *The person that programmes it is the equipment supplier and not the user.* The user simply says that "I want one of them". I think that you'll find that they will end up in all those kind of things. They will probably sit in fire alarm systems annunciators and so-on.

MW: What do you think the future is for DCS technology?

DL: I think that if you have a plant of a certain scale then it's obvious that you need to marry it up with a DCS. Now, you are going to also plug that DCS into the IT system. We kept trying that in the 90s and we did all sorts of things and we did do it, but it wasn't easy. I can remember one bridge we built, to get data from one system to another and it completely backup up and shut the computers down because it couldn't transfer data and the whole thing got clogged. What we have now, "Active Factory" seems pretty robust, and I can't see anybody buying any system without that. I can see people buying PLCs without it. You can see what's happening on a piece of kit about the size of my office but if it's the size of the plants we have talked about here then you need that data.

It's like most things it [technology] can't get much smaller because you need to get wires on them, they can get more functionality, but actually if they get more functionality, they are only moving into areas where other things have got it anyway. So it's quite a difficult area now and the number of suppliers have reduced considerably I think as they've all taken each other over because the market is only a certain size.

A classic area where we did end up with PLCs, I think it was Stevenston, and they were pretty good in this, we had about five compressors. They were a very inefficient operation because they were all operating independently. And we put a PLC "brain" on each. It was pre-programmed and each of them fed some data into a master PLC which then controlled the loading and unloading of those five compressors.

You can see in that application because if that's in the Power House or somewhere like that, large process plant like Paraquat doesn't care about that, it's the sort of thing that can operate blindly. Perhaps someone is checking the health of it now and again.

MW: I suppose it's a sequence really, as long as it's repetitive and there's minimal quality control.

DL: Yes that's right.

MW: Is there a topic you feel that I should have asked you but haven't? Have I missed something?

DL: No I don't think so. I think we've sort of meandered through the process control and SCADA development over the last 20 to 30 years. From blind control to pretty basic terminals that just gave basic process information. You might have 2 or 3 pages that say check the pressures, temperatures through to the sophisticated control and monitoring systems that you see here today. And you've got all the things in between.

The only thing I would say about PLCs is that it depends on the scale of the plant that will decide which technology you go to. But you're not going to put one of these systems on a fill and packing line but neither are you going to put the biggest PLC you can on it. But when you really think of it, it probably has the Siemens I/O that a PLC has so it's really about connecting it to a better brain. I am sure that somebody has said to me that the cards are intrinsically safe so they [Siemens] have recognised that the big market is chemicals otherwise PLC manufacturers wouldn't bother going through all the certification process. But if they are intrinsically safe that saves an awful lot marshalling room.

If you're a milk or wine bottling company or something like that you're going to have PLCs all over the place but the scale of architecture on the sites that we're on, it's the big stuff.

MW – David, thank you very much indeed.

Appendix F

Patent and Interview Correlation

Table A1 shows the main developments in PLC technology from selected patents and shows the links to the interviews containing evidence supporting the patent. The patent list is not exhaustive but gives examples describing the technology in use; the corresponding interview references, confirm the date range and commercial presence of the innovation described in the patent.

Time Period	Key Development	Patent Source (example)	Secondary Source (example)
Pre- PLC (1969)	Sequence control performed by relays, cams and Drums	GB251559 (Cutler-Hammer, 1926) US1939183 (Neuman, 1933) US2846892 (Roessler Jr., 1958) US3624611 (Wirsing, 1971) US3944987 (Koyanagi et al., 1976)	(Young, 2007) (Morris, 2007b)
1950s	Computers for process control	US3204087 (Millis Jr., 1965) US3226684 (Cox, 1965) US3275988 (Yetter, 1966b) US3686639 (Fletcher and Rosseau, 1972) US3740722 (Greenberg et al., 1973)	(Bruce, 2008) (Young, 2007) (Morris, 2007b)
1953	Reliability & Robustness	GB746601 (Roberts, 1956) CA736102 (Yetter, 1966a) US3686639 (Fletcher and Rosseau, 1972) US3753243 (Ricketts Jr. et al., 1973) US3810118 (Kiffmeyer, 1974)	(Young, 2007) (Morris, 2007b) (Bruce, 2008)
1960s	Electronic logic gates	US3624611 (Wirsing, 1971)	(Morris, 2007b) "Norbit 2" (Young, 2007) "Bistat"
1969	Programming Tool Development	US3686639 (Fletcher and Rosseau, 1972) US3810118 (Kiffmeyer, 1974)	(Daley, 2010) (Young, 2007) (Bruce, 2008)
1969	Ladder programming language	US3686639 (Fletcher and Rosseau, 1972) US3810118 (Kiffmeyer, 1974) US3832696 (Nakao et al., 1974) FR2361689 (Michel et al., 1978b)	(Bruce, 2008) (Young, 2007) (Morris, 2007b) (Pittwood, 2008b) (Leeming, 2008) (Daley, 2010)
1969	Memory: core store	US3740722 (Greenberg et al., 1973) GB1290651/US3753243 (Ricketts Jr. et al., 1972, 1973) US3806714 (Otsuka et al., 1974) US3944987 (Koyanagi et al., 1976)	(Young, 2007) (Pittwood, 2008b)
1971	Analogue Inputs & Outputs	US3731279 (Halsall and Murrell, 1973) US4188617 (Fauchier et al., 1980) US4158226 (Struger and Grants, 1979) DE3720006 (Hinsken, 1988b)	(Young, 2007) (Leeming, 2008) (Daley, 2010) (Hammond, 2007) (Pittwood, 2008b)
1971	Memory: Solid-state	US3810118 (Kiffmeyer, 1974) US3798612 (Struger and Radtke, 1974) US4227247 (Kintner, 1980)	(Young, 2007) (Leeming, 2008) (Pittwood, 2008b)
1976	Remote I/O and Distributed Control	US4078259 (Soulsby and Seipp, 1978) US4162536 (Morley, 1979) US4307447/ US4484303 (Provanzano et al., 1981, 1984)	(Bruce, 2008) (Daley, 2010) (Young, 2007) (Morris, 2007b) (Pittwood, 2008b) (Hammond, 2007)
1978+	Microprocessors	US4107785 (Seipp, 1978) US4165534 (Dummermuth et al., 1979) US4217658 (Henry et al., 1980) US4266281 (Struger and Dummermuth, 1981) US4302820 (Struger et al., 1981) US4282584 (Brown et al., 1981) US4188617 (Fauchier et al., 1980)	(Hammond, 2007)

Table A1 - Patent and interview correlation

The purpose of table A1is to demonstrate the correlation between the developments described in the patent material with the experiences and recollections of the engineers interviewed as part of the research. Not all innovations are described in the table although some are explicit precursors to later technological developments. Alternative supporting literature has been excluded from the table for reasons of clarity but is no less important. For example, Aylen (2004, 2010) provides an account of the development and application of digital computers for process control in the 1950s and 60s for the steel industry that supports the time frame suggested in the patent material and the interviewee's recollections.

Correlation of Interviews with Patent Material

Chapters 4 and 5 reviewed patents relevant to the development of sequential control and PLC technologies. Patents are a useful source of information because they describe the "background" to the invention and current "state-of the art" at the time of filing, providing a useful and valid historical record. The accuracy of the patent itself is checked externally by third-party patent examiners regarding "prior art" to ensure that the idea being patented is indeed novel.

What a patent cannot tell you is what happened after it was filed. There is no exact or direct record in the patent literature that demonstrates whether the invention became a successful innovation. In order to verify that a patent is an accurate historical record, it is useful to correlate the patent information with other independent sources. This section attempts to confirm the main technological developments stated from the patent research from the recollections of the interviewees.

The issue relating to robustness and reliability is raised by Young, Morris and Bruce who relate the importance of the reliability of PLC technology and its ability to function in the harsh environments. These issues were initially identified in the patents in the 1950s, prompted by the application of electronic control systems and the early application of computers to process control. Perhaps surprisingly this does not feature in earlier patents but one explanation could be that the existing control technologies (relays, cams and drums) were already suited to the industrial environment. Reliability and robustness was a significant feature of the PLC and was explicitly stated in the early patents (Fletcher and Rosseau, 1972, Ricketts Jr. et al., 1972, Kiffmeyer, 1974) defining one of the key characteristics of the new emerging technology.

The ladder diagram programming language was another defining feature of the PLC and is mentioned in many patents. All of the interviews revealed that the ladder notation was used in the early PLCs and is still used as a popular PLC programming language to the present day.

Developments in memory are only regarded by the end users of the PLC in terms of the size of the program. Young, for example, recalled that usable program memory was expanding considerably in the 1970s. The technology relating to the memory appears not to be an important facet of the PLC from the user's perspective. The technical aspects of the memory appear only to be of importance to the PLC designers and these innovations are recorded in the patents.

The use of microprocessors presents a similar story. All interviewees were aware of the importance of using microprocessors and the benefits gained from them for the PLC, but none explicitly stated this fact with the exception of Hammond. The PLC was largely regarded as an of-the-shelf device and the relative importance was based primarily on its function, not necessarily the underpinning technology. Hammond was directly involved in the development of a microprocessor-based controller (MPC 80) and as such was acutely aware of the technical challenges imposed by implementing this technology.

The key advances for the PLC in the 1970s and early 80s noted by the interviewees was the development of analogue I/O and digital networking. Analogue inputs enabled the PLC to expand the potential number of applications by having the ability to monitor process variables and driving digital outputs in response to programmed conditions. Analogue outputs further improved the PLCs capability to control processes, for example proportional valve control, that enabled the PLC to compete with process-control computers and DCS, albeit on a smaller scale. Networking provided another significant step in the development of the PLC system. The introduction of remote I/O enabled physically larger processes and equipment to be controlled by the PLC. Following remote I/O was the ability to distribute the processing over the plant or larger distances (see Daley, 2010, Morris, 2007b, Young, 2007).