

Sudan University of Science & Technology



Collage of Graduate Studies

PLC Design Based On Monitoring Crude Oil Vessel

تصميم المتحكمات القابلة للبرمجة اعتمادا علي أوعية مراقبة النفط الخام

A dissertation

Submitted in Partial Fulfillment of the Requirements

For the Degree of MSc in Mechatronics Engineering

By:

Salim Makki Ahmed Awadallah

Supervisor:

Dr. A.Rasoul Elzubaidi

September 2015

DEDICATION

To my late mother,

She taught me to persevere and prepared me to face challenges with faith and humility. She was a constant source of inspiration to my life. Although she is not here to give me strength and support I always feel her presence, that used to urge me to strive to achieve my goals in life.

ACKNOWLEDGEMENT

So many people have encouraged and supported me throughout the writing of this research. I would like to acknowledge their contribution .

My thanks and gratitude to Dr. A.Rasoul Elzubaidi for all support and guidance. Special thanks to our best friends without specification for their advice, support and endless patience.

Thanks are extended to our colleagues for their great effort and encouragement work without them would not have been possible.

TABLE OF CONTENTS

Dedication	I
Acknowledgment	II
Table of contents	III
List of Figures	VI
List of tables	IX
Abbreviations	X
Abstract	XIV
المستخلص	XV
Chapter One :Introduction	
1.1 Background.....	1
1.2 Problem Statement.....	3
1.3 Objective.....	3
1.4 Scope of Work.....	3
1.5 Methodology.....	4
1.6 Research Outline	6
Chapter Two :Literature Review	
2.1 Control System Overview.....	7
2.1.1 Process Overview.....	7
2.1.2 Process Control Elements.....	9
2.1.3 Process Control Variables.....	10
2.1.4 Manual Control Operation.....	10
2.1.5 Automated System Building Blocks.....	13
2.2 PLC Overview.....	15

2.2.1 PLC Architecture.....	21
2.2.2 Program Scan.....	28
2.2.3 PLC Programming Languages.....	29
2.3 Previous Studies	40
Chapter Three : Simulation Hardware Devices	
3.1 The Central Processing Unit (CPU)	44
3.2 The I/O Section.....	46
3.3 Analog I/O Modules	48
3.4 Freelance 800F Architecture	52
3.4.1 Operator station	52
3.4.2 Engineering Station	52
3.4.3 Process Station – AC800F	53
3.5 Field devices	60
3.5.1 Basic Measurement System	60
3.5.2 Level Measurement	62
3.5.3 Temperature Measurement	63
3.5.4 Pressure Measurement	64
3.5.5 Flow Measurement	65
3.5.6 Analyzer	66
3.5.7 Pump	68
3.5.8 Motor Operated Valve	69
3.5.9 Emergency Shutdown Valve	70

Chapter Four: Software

4.1 Control Builder F	71
4.2 Digivis	71
4.3 System Configuration	72
4.3.1 Project Manager	73
4.3.2 Project Tree	73
4.3.3 Hardware Structure	80
4.3.4 Commissioning	86
4.4 Process Description	89
4.4 .1 Level monitoring	89
4.4 .2 Temperature monitoring	91
4.4 .3 Pressure monitoring	92
4.4 .4 Flow monitoring	93
4.4 .4 Water cut monitoring	94
4.5 Simulation	95

Chapter Five: Result & Discussion

5.1 Result	97
5.2 Discussion	99

Chapter Six: Conclusion&Recommendations

6.1 Conclusion	103
6.2 Recommendations	104
References	106
Appendices.....	108

LIST OF FIGURES

Figure 1.1 Flow Chart of Research Methodology.....	5
Figure1.2 System block diagram	6
Figure 2.1 Control systems functional view.....	8
Figure 2.2 Water flow tank process	9
Figure2.3 Manual control systems	12
Figure 2.4 Level manual controls	14
Figure 2.5 Closed-loop controls.....	14
Figure 2.6 Programmable logic controller.....	17
Figure 2.7 Relay- and PLC-based control panels.. ..	18
Figure 2.8 All the logic is contained in the PLC's memory	19
Figure 2.9 Relationships between the inputs and outputs	19
Figure 2.10 PLC communication module.....	20
Figure 2.11 High-speed counting.....	21
Figure 2.12 Control program	21
Figure 2.13 Typical parts of a programmable logic controller	23
Figure 2.14 Fixed I/O configurations.....	24
Figure 2.15 Modular I/O configurations	25
Figure 2.16 The power supply supplies DC power to other modules	26
Figure 2.17 Typical PLC processor modules	26
Figure 2.18 Typical PLC scans cycle.....	27
Figure 2.19 Typical PLC input/output (I/O) system connections	28
Figure 2.20 PLC program scan cycle	29
Figure 2.21 Standards IEC 61131 languages associated with PLC programming ..	31

Figure 2.22 Comparison of ladder diagram and instruction list programming...	32
Figure 2.23 Function block diagram equivalents to ladder	33
Figure 2.24 Block diagram parts	34
Figure 2.25 Scale Change block diagram	35
Figure 2.26 Individual Drive Function for Bi-directional Units block diagram...	36
Figure 2.27 Analog Monitoring block diagram	38
Figure 2.28 Timer, Switch-on Delay	39
Figure 2.29 parameter definition masks	40
Figure 3.1 Sections of a PLC processor module	45
Figure 3.2 PLC power supply	46
Figure 3.3 Typical processor modules.....	47
Figure 3.4 Rack-based I/O section	48
Figure (3.5) 16, 32, and 64 point I/O modules.....	49
Figure 3.6 Analog input and output to a PLC	50
Figure(3.7) 4-channel analog thermocouple input module	51
Figure 3.8 Sensor and analog module supplied power	52
Figure 3.9 Typical analog I/O control system	53
Figure 3.10 AC 800F	55
Figure 3.11 Base unit with CPU board – PM802F	55
Figure 3.12 Power supply module SA 801F	56
Figure 3-13 Ethernet modules EI 803F	57
Figure 3.14 Profibus module FI 830F	58
Figure 3.15 Serial Interface module FI 820F	59
Figure 3.16 CAN module FI 810	60
Figure 3.17 S800 I/O AI810	61
Figure 3.18 Instrument block diagram	62

Figure 3.19	Level Transmitter	64
Figure 3.20	Temperature Transmitter and Electrical Heater	65
Figure 3.21	Pressure Transmitter	66
Figure 3.22	Turbine Flow Meter	67
Figure 3.23	Water Cut Analyzer	68
Figure 3.24	Pump	69
Figure 3.25	Motor Operated Valves	70
Figure 3.26	Emergency Shutdown valve	71
Figure 4.1	project tree user interface	75
Figure 4.2	System view	85
Figure 4.3	Station view	86
Figure 4.4	Freelance 800F system	89
Figure 4.5	Block Diagram of Level Transmitter	90
Figure 4.6	Block Diagram of Pump	91
Figure 4.7	Block Diagram of Temperature Transmitter	92
Figure 4.8	Block Diagram of Heater	93
Figure 4.9	Block Diagram of ESD Pressure Transmitter	93
Figure 4.10	Block Diagram of Emergency Shutdown Valve	94
Figure 4.11	Block Diagram of Flow Transmitter	95
Figure 4.12	Block Diagram of Analyzer	95
Figure 4.13	Block Diagram of Motor Operated Valve.....	96
Figure 4.14	Vessel Process Monitoring by Using Freelance 800F simulator	97

LIST OF TABLES

Table 2.1 Method Operation of Inhibition inputs 37

Table 5.1 Inputs 98

Table 5.2 Outputs 98

Table 5.3 Output Alarming 99

ABBREVIATIONS

AC	Advant Controller.
A/D	Analog to Digital Converter
AI	Analog Input
AO	Analog Output
API	American Petroleum Institute
bar	barometer unit of atmospheric pressure measurement
Bps	Bits per second
CAN	Control Area Network or Control and Automation Network
CIP	Common Industrial Protocol
CPU	Central Processing Unit
CS	Configuration Server
DCS	Distributed Control System
D/A	Digital to Analog Converter
DC or dc	Direct current
DCP	Data Collection Platform
DI	Digital Input
DO	Digital Output
DP	Distributed I/O

ESDV	Emergency Shutdown Valve
FBD	Function Block Diagram
GS	Gateway Station
HMI	Human Machine Interface
IEC	International Electrotechnical Commission
IEEE	Institute of Electrical and Electronic Engineers (US)
IDF	Individual Drive Function
IL	Instruction List
IP	Internet Protocol
ISO	International Organization for Standardization
IT	Information Technology (as in IT manager or IT department)
KPa	kilo-Pascal
LAD	Ladder Diagram
LED	Light Emitting Diode
mA or ma	milliamperes
MB	mega-Byte
Mbs, mbps	Mega Bits per Second
MMI	Man-Machine Interface

MOV	Motor Operated Valve
MSB	Most Significant Bit
MV	Manipulated variable
NIR	Near Infrared
PC	Personal Computer
PID	Proportional Integral and Derivative
PL	Program list
PLC	Programmable Logic Controller
PS	Power Supply
PS	Process Station
psi or PSI	Pounds Per square Inch
<i>PTC</i>	Positive Temperature Coefficient
PV	Process variable
OPC	Open Platform Communications
OS	Operator Station
OWD	On-Line Water Determination
RAM	Random access memory
RED	Redundant
ROM	Read only memory

RTD	Resistance temperature device
SCADA	Supervisory Control And Data Acquisition
SFC	Sequential Function Chart
SP	Set Point (reference)
ST	Structured Text
TCP/IP	Transmission control Protocol / Internet Protocol
TON	Timer On Delay
UTP	Unshielded Twisted Pair

ABSTRACT

A PLC is a digital computer used to automate electromechanical processes. PLC is frequently used when implementing automated control, which is an important part of many modern industries.

This research is based on monitoring Level Temperature Pressure Flow and water cut of crude oil in the vessel by using ABB Programmable Logic Controller .

Automatic control of crude oil in vessel can work continuously and can provide accurate quantity of crude in less time. In such process there is no need of labor so there is no human error. Without human error, the quality of product is better and the cost of production would definitely decrease .

The purpose of the research is to replace the manual system being used in the industry, compare the time, and manpower requirement for both the existing system with the proposed automated system.

المستخلص

المتحكمات المنطقية القابلة للبرمجة عبارة عن حاسوب يستخدم للتحكم في العمليات الكهروميكانيكية . تستخدم المتحكمات المنطقية القابلة للبرمجة في تصميم التحكم الأوتوماتيكي وهذا يشكل دور مهم في جزء كبير من الصناعات الحديثة.

هذا البحث مبني علي مراقبة المستوي ودرجة الحرارة والضغط و التدفق وكمية الماء للنفط الخام الموجود في وعاء باستخدام متحكمات قابلة للبرمجة من شركة أي بي بي .

التحكم الأتوماتيكي في النفط الخام الموجود في وعاء يمكن ان يحدث باستمرار ويمكن ان نحصل علي كمية دقيقة من البترول في أقل زمن .

في هذه العمليات لاتوجد الحاجة للعمال ولهذا لا يوجد خطأ بشري . مع عدم وجود الخطأ البشري جودة المنتجات سوف تكون جيدة وتكلفة هذه المنتجات بكل تأكيد سوف تقل.

الغرض من هذا البحث استبدال النظام اليدوي الذي يستخدم في المصانع بالمقارنة مع النظام الأتوماتيكي من حيث الزمن والحاجة الي الأيدي العاملة.

Chapter One

INTRODUCTION

1.1 Background

PLCs were first introduced in the 1960's. The primary reason for designing such a device was eliminating the large cost involved in replacing the complicated relay based machine control systems. Bedford Associates (Bedford, MA) proposed something called a Modular Digital Controller (MODICON) to a major US car manufacturer. The MODICON 084 brought the world's first PLC into commercial production.

When production requirements changed so did the control system. This becomes very expensive when the change is frequent. Since relays are mechanical devices they also have a limited lifetime because of the multitude of moving parts. This also required strict adherence to maintenance schedules. Troubleshooting was also quite tedious when so many relays are involved. Now picture a machine control panel that included many, possibly hundreds or thousands, of individual relays. The size could be mind boggling not to mention the complicated initial wiring of so many individual devices. These relays would be individually wired together in a manner that would yield the desired outcome.

The problems for maintenance and installation were horrendous.

These new controllers also had to be easily programmed by maintenance and plant engineers. The lifetime had to be long and programming changes easily performed.

They also had to survive the harsh industrial environment. The answers were to use a programming technique most people were already familiar with and replace mechanical parts with solid-state ones which have no moving parts. Communications abilities began to appear in approximately 1973.

The first such system was Modicon's Modbus. The PLC could now talk to other PLCs and they could be far away from the actual machine they were controlling. They could also now be used to send and receive varying voltages to allow them to use analog signals, meaning that they

Were now applicable to many more control systems in the world. Unfortunately, the lack of standardization coupled with continually changing technology has made PLC communications a nightmare of incompatible protocols and physical networks.

Nowadays computer control and information system technology is applied widely in most of the process industry, because it may produce significant technical and economic benefits. Process control information systems assist operating personnel in producing the required output of products with minimum quality variations, least consumption of the raw material and energy, and maximum efficiency.

Different control techniques have been proposed for vessel crude oil controlling. It is common to use relays to make simple logical control decisions. The relays allow power to be switched on and off without a mechanical switch. The development of low cost computer has brought the most recent revolution, the Programmable Logic Controller (PLC).

With the advent of the PLC; it has become the most common choice for manufacturing controls. A PLC is a digital operating electronic apparatus which uses a programmable memory for internal storage of instruction for implementing specific function such as logic, sequencing, timing, counting and arithmetic to control through analog or digital input/output modules various types of machines or process.

1.2 Problem Statement

This research addresses the problem of various programmable control manufacturers using different nomenclature and program forms by describing the principles involved and illustrating them with examples from a range of manufacturers.

Although there are several methods of control process such as PID controller that is widely used in industry field but the system model cannot be accurate enough since many of PID loops in operational today are continual need of monitoring and adjustment since they can easily become in properly tuned (e.g. this due to process parameter variations or operating condition changes).

1.3 Objective

The objective of this research is to develop a simple a process plant that control process in vessel using programmable logic controller PLC Freelance 800F . The main objective of this research is to be able to construct Functional Block Diagram that can control the desired system by entering mnemonic code in to the programming unit.

1.4 Scope of Work

The work done during this project can be divided in to several parts.

First of all the programmable logic controller and its applications have to be studied focusing more on Freelance 800F. Next the function block diagram has to be constructed in to mnemonic code before key in to the programming console then the system need to be tested and modify the function block diagram to suit the hardware.

Finally we will be used suitable PLC simulator to control the system.

1.5 Methodology

There are a few methods that must be following to complete this Research. The method is as below:

- 1- The Research had started with study the literature of the Research. It's including the control system and process control elements.
- 2- Then study about the Programmable logic controller and create the Functional Block Diagram. At the same time studies about the component such as valve and sensor. Next, try to construct the hardware.
- 3- Test the functionality of the Functional Block Diagram and the hardware. Then integrate the hardware and the software. If it can't function, then try to construct the Functional Block Diagram and the hardware again.
- 4- Then if the hardware and software is successfully done, then prepare for the final report.

The design of this project has three major parts. First is the interfacing of the input devices to the controller. Second is the interfacing of all the loads with the controller and the third is to make it stand alone system and provide it with necessary and different levels of voltage supplies.

1.6 Research Outline

This research will be divided in to six chapters to provide the understanding of the whole research.

Chapter One

Introduction to programmable logic controller. This chapter also explains about research objectives and scopes of work and discuss about problem statement.

Chapter Two

Describe about the literature review that has been studied to get information to complete the research. This study is focused especially on the concepts of process control and automation.

PLC Architecture and languages also explained in this chapter.

Chapter Three

Covers the simulation hardware devices and Freelance 800F Architecture Chapter Four

Covers up all the research software and system configurations.

Chapter Five

Explains the results and discussions of the research .

Chapter Six

Conclusion and recommendation of the research for the future development or system modification.

Chapter Two

LITERATURE REVIEW

2.1 Control System Overview

A control system is a device or set of structures designed to manage, command, direct, or regulate the behavior of other devices or systems. The entire control system can be viewed as a multivariable process that has a number of inputs and outputs that can affect the behavior of the process. Figure 2.1 shows this functional view of control systems. This section is intended as a brief introduction to control systems.

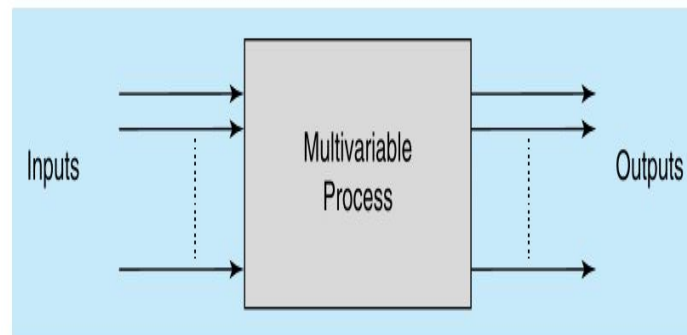


Figure 2.1 Control systems functional view.

2.1.1 Process Overview

In the industrial world, the word process refers to an interacting set of operations that lead to the manufacture or development of some product. In the chemical industry, process means the operations necessary to take an assemblage of raw materials and cause them to react in some prescribed fashion to produce a desired end product, such as gasoline. In the food industry, process means to take raw materials and operate on them in such a manner that an edible high-quality product results. In each use, and in all other cases in the process industries, the end product must have certain specified properties that depend on the conditions of the

reactions and operations that produce them. The word control is used to describe the steps necessary to ensure that the regulated conditions produce the correct properties in the product.

To produce a product with the specified properties, some or all the process variables must be maintained at specific values in real time. Figure 2.2 shows free water flow through a tank, similar to rain flow in a home gutter system. The tank acts in a way to slow the flow rate through the piping structure. The output flow rate is proportional to the water head in the tank. Water level inside the tank will rise as the input flow rate increases. At the same time, output flow rate will increase with a noticeable increase in the tank water level. Assuming a large enough tank, level stability will be reached when the flow in is equal to the flow out. This simple process has three primary variables: FLOW IN, FLOW OUT, and the tank level. All three variables can be measured and, if desired, also can be controlled. The tank level is said to be a self-regulated variable.

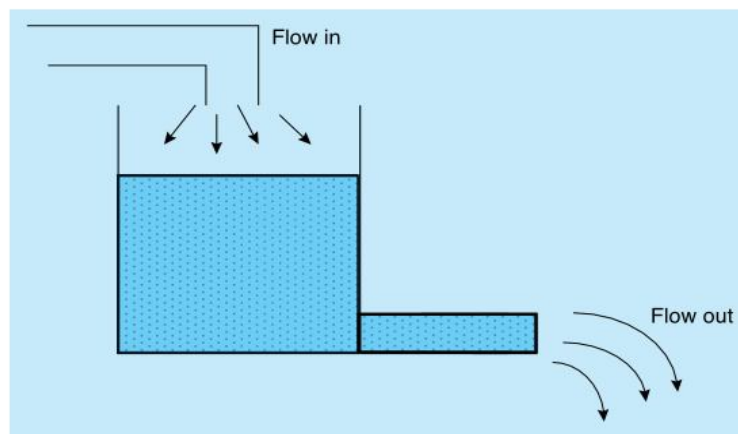


Figure 2.2 Water flow tank process.

Some of the variables in a process may exhibit the property of self-regulation, whereby they will naturally maintain a certain value under normal conditions. Small disturbances will not affect the tank level stability because of the self-

regulation characteristic. A small increase in tank inflow will cause a slight increase in the water level. An increase in water level will cause an increase in tank outflow, which eventually will produce a new stable tank level. Large disturbances in tank input flow may force undesired changes in the tank level. Control of variables is necessary to maintain the properties of the product, the tank level in our example, within specification. In general, the value of a variable v actually depends on many other variables in the process as well as on time.

2.1.2 Process Control Elements

A simple process-control loop consists of three elements: the measurement, the controller, and the final control element. Measurement is one of the most important elements in any process-control plant. Decisions made by the controller are based on the real-time measurements information received. Regardless of system type, all controller decisions are similarly based on measurements, control strategy, and the desired process response/performance. Final control elements can refer to actuators such as control valves, heaters, variable-speed drives, solenoids, and dampers.

In most process plants, a final control element is often a control valve. In manufacturing assembly lines, final control elements mostly will include variable-speed drives, solenoids, and dampers. Final control elements receive command signals from the controller/PLC in real time to bring about the desired changes in the controlled process. As with sensors/measurement elements, final control devices interface with the PLC output modules in a similar way. The PLC digital-signal outputs are transformed to the actuator required digital- or analog-signal format, which might require a D/A conversion or coupling isolation.

2.1.3 Process Control Variables

Process-control variables that are commonly either measured by sensors or regulated through actuators (final control elements) include temperature, pressure, speed, flow rate, force, movement, velocity, acceleration, stress, strain, level, depth, mass, weight, density, size, volume, and acidity. Sensors may operate simple ON/OFF switches to indicate certain events or detect objects (proximity switch), empty or full (level switch), hot or cold (thermostat), high or low pressure (pressure switch), and other overload conditions.

The final or correcting control element is the part of the control system that acts to physically change the process behavior. In most processes, the final control element is a valve used to restrict or cut off fluid flow, pump motors, louvers used to regulate airflow, solenoids, or other devices. Final control elements are typically used to increase or decrease fluid flow. For example, a final control element may regulate the flow of fuel to a burner to control temperature, the flow of a catalyst into a reactor to control a chemical reaction, or the flow of air into a boiler to control boiler combustion. In any control loop, the speed with which a final control element reacts to correct a variable that is out of set point is very important. Many of the technological improvements in final control elements are related to improving their response time.

2.1.4 Manual Control Operation

In a manual control system, humans are involved in monitoring the process and carrying out necessary decisions to bring about desired changes in the process. Still, computers and advanced digital technologies may be used to automate a wide variety of process operation, status, command, and decision-support functions. Sensors and measurement instruments are used to monitor the status of different process variables conditions, whereas final control elements or actuators are used

to force changes in the process. As shown in Fig. 2.3, humans close the control loop and establish the connection between measured values, desired conditions, and the needed activation of the final control elements.

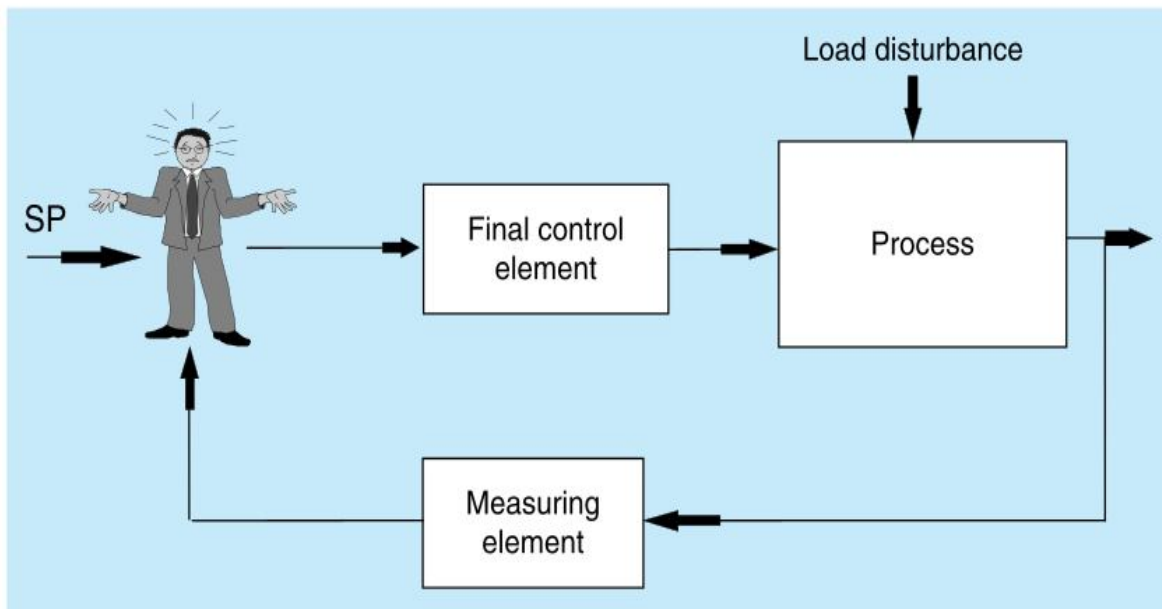


Figure2.3 Manual control systems.

Manual control is widely available and can be effective for simple and small applications. The initial cost of such systems might be relatively smaller than that of automated ones, but the long-term cost is typically much higher. It is difficult for operators to achieve the same control/quality because of varying levels of domain expertise as well as unexpected changes in the process. The cost of operation and training also can become a burden unless certain functions are automated. Most systems start by using manual control or existed previously with manual operation. System owners acquire and accumulate process-control experience over time and use this knowledge eventually to make process improvements and eventually automate the control system. The introduction of digital computers into the control loop has allowed the development of more flexible control systems, including higher-level functions and advanced algorithms.

Furthermore, most current complex control systems could not be implemented without the application of digital hardware. However, the simple sequence of sensing, control, and actuation for the classic feedback control becomes more complex as well.

A real-time system is one in which the correctness of a result depends not only on the logical correctness of the calculation but also on the time at which the different tasks are executed. Time is one of the most important entities of the system, and there are timing constraints associated with system tasks. Such tasks normally have to control or react to events that take place in the outside world that are happening in real time. Thus a real-time task must be able to keep up with external events with which it is concerned. Figure 2.4 shows a simple manual control system. The level in the tank shown varies as a function of the flow rate through the input valve and the flow rate through the output valve.

The level is the control or controlled variable, which can be measured and regulated through valve control and adjustment at the input or output flow or both. The two valves can be motorized and activated from an easy-to-use operator interface. Valve position variations are achieved through an operator input based on observed real-time process conditions. We will see next that the operator can easily be eliminated.

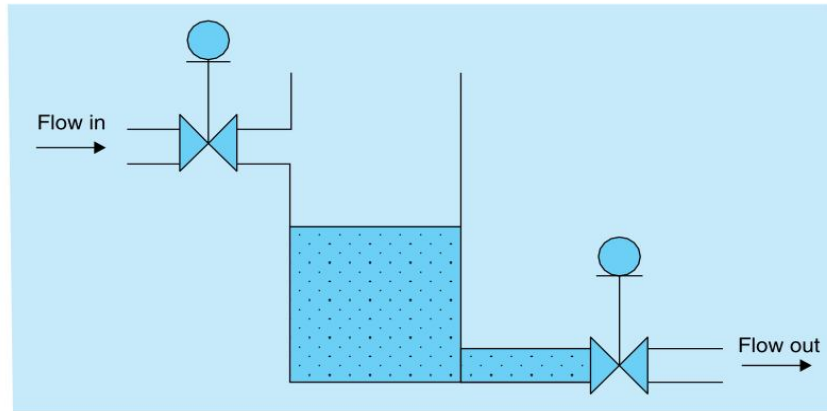


Figure 2.4 Level manual controls.

2.1.5 Automated System Building Blocks

The closed control loop shown in Fig. 2.5 consists of the following five blocks:

- Process
- Measurement
- Error detector
- Controller
- Control element

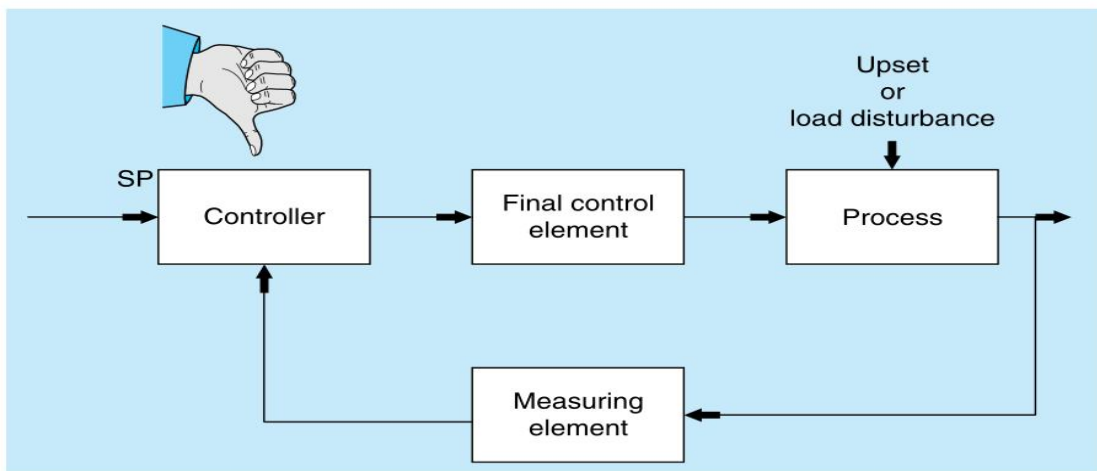


Figure 2.5 Closed-loop controls.

In manual control, the operator is expected to perform the task of error detection and control. Observations and actions taken by operators can lack both consistency and reliability. The limitations of manual control can be eliminated through the implementation of closed-loop systems and the associated process-control strategies. Figure 1.5 shows a block diagram of a single-variable closed-loop control. The controller can be implemented using variety of technologies, including hardwired relay circuits, digital computers, and more often PLC systems. It is impossible to achieve perfect control, but in the real world, it is not needed. We can always live with small errors within our acceptable quality range.

Errors in real time are used to judge the quality of the system design and its associated controller. The errors can be measured in three ways, as explained by the following definitions:

Absolute error = set point – measured value

Error as percent of set point = absolute error/set point \times 100

Error as percent of range = absolute error/range \times 100

Range = maximum value – minimum value

Errors are commonly expressed as a percentage of range and occasionally as a percentage of set point but rarely as an absolute value. Also, most process variables are also commonly quantified as percentage of the defined range. This quantification allows for universal input-output PLC computer interfaces regardless of the physical nature of the sensory and actuating devices. A PLC analog input module having several input slots can accommodate and process temperature, pressure, motor speed, viscosity, and many other measurements in exactly the same way. Later chapters will detail the PLC hardware and software as applied to real-world industrial control applications.

2.2 PLC Overview

Programmable logic controllers (Figure 2.6) are now the most widely used industrial process control technology.

A programmable logic controller (PLC) is an industrial grade computer that is capable of being programmed to perform control functions. The programmable controller has eliminated much of the hardwiring associated with conventional relay control circuits. Other benefits are including easy programming and installation, high control speed, network compatibility, troubleshooting and testing convenience, and high reliability. The programmable logic controller is designed for multiple input and output arrangements, extended temperature ranges, immunity to electrical noise, and resistance to vibration and impact. Programs for the control and operation of manufacturing process equipment and machinery are typically stored in battery-backed or nonvolatile memory. A PLC is an example of a real-time system since the output of the system controlled by the PLC depends on the input conditions.

The programmable logic controller is, then, basically a digital computer designed for use in machine control. Unlike a personal computer, it has been designed to operate in the industrial environment and is equipped with Special input/output interfaces and a control programming language. The common abbreviation used in industry for these devices, PC, can be confusing because it is also the abbreviation for “personal computer.” Therefore, most manufacturers refer to their programmable controller as a PLC, which stands for “programmable logic controller.” Initially the PLC was used to replace relay logic, but its ever-increasing range of functions means that it is found in many and more complex applications. Because the structure of a PLC is based on the same principles as those employed in computer architecture, it is capable not only of performing relay

switching tasks but also of performing other applications such as timing, counting, calculating, comparing, and the processing of analog signals.



Figure 2.6 Programmable controllers.

Programmable controllers offer several advantages over a conventional relay type of control. Relays have to be hardwired to perform a specific function. When the system requirements change, the relay wiring has to be changed or modified. In extreme cases, such as in the auto Industry, complete control panels had to be replaced since it was not economically feasible to rewire the old panels with each model changeover.

The programmable controller has eliminated much of the hardwiring associated with conventional Relay control circuits (Figure 2.7).



(a)



(b)

Figure 2.7 Relay- and PLC-based control panels. (a) Relay-based control panel. (b) PLC-based control panel. Source:

It is small and inexpensive compared to equivalent relay-based process control systems. Modern control systems still include relays, but these are rarely used for logic.

In addition to cost savings, PLCs provide many other benefits including:

- **Increased Reliability.** Once a program has been written and tested, it can be easily own loaded to other PLCs. Since all the logic is contained in the PLC's memory, there is no chance of making a logic wiring error (Figure 2.8). The program takes the place of much of the external wiring that would normally be required for control of a process.

Hardwiring, though still required to connect field devices, is less intensive. PLCs also offer the reliability associated with solid-state components.

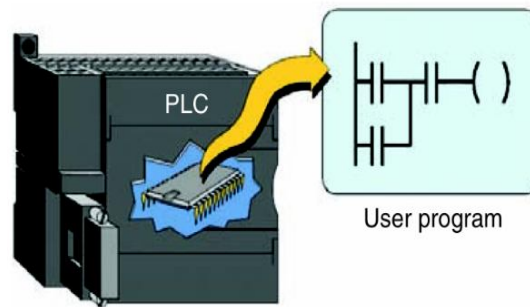


Figure 2.8 All the logic is contained in the PLC's memory.

- **More Flexibility.** It is easier to create and change a program in a PLC than to wire and rewire a circuit. With a PLC the relationships between the inputs and outputs are determined by the user program instead of the manner in which they are interconnected (Figure 2.9).

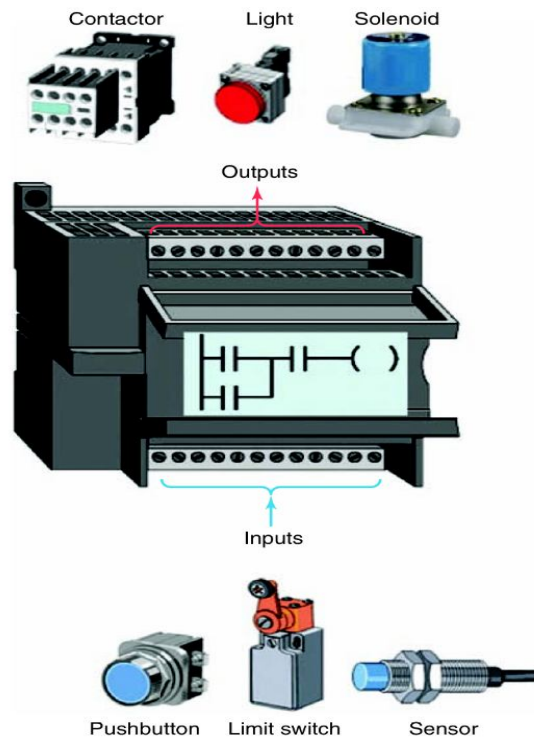


Figure 2.9 Relationships between the inputs and outputs are determined by the user program.

Original equipment manufacturers can provide system updates by simply sending out a new program. End users can modify the program in the field, or if desired, security can be provided by hardware features such as key locks and by software passwords.

- **Lower Cost.** PLCs were originally designed to replace relay control logic, and the cost savings have been so significant that relay control is becoming obsolete except for power applications. Generally, if an application has more than about a half-dozen control relays, it will probably be less expensive to install a PLC.
- **Communications Capability.** A PLC can communicate with other controllers or computer equipment to perform such functions as supervisory control, data gathering, monitoring devices and process parameters, and download and upload of programs (Figure 2.10).

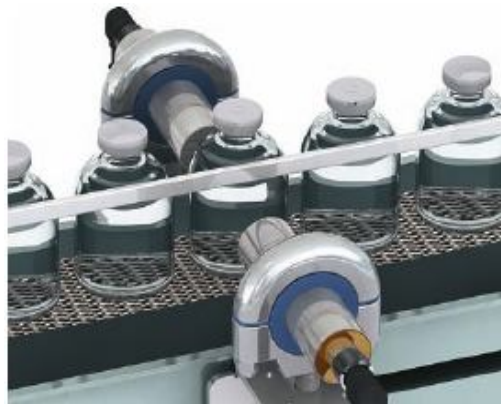


Figure 2.10 PLC communication module.

- **Faster Response Time.** PLCs are designed for high-speed and real-time applications (Figure 2.11). The programmable controller operates in real time, which means that an event taking place in the field will result in the execution of an operation or output. Machines that process thousands of items per second and

objects that spend only a fraction of a second in front of a sensor require the PLC's quick-response capability.



Figure 2.11 High-speed counting.

- Easier to Troubleshoot. PLCs have resident diagnostics and override functions that allow users to easily trace and correct software and hardware problems.

To find and fix problems, users can display the control program on a monitor and watch it in real time as it executes (Figure 2.12).

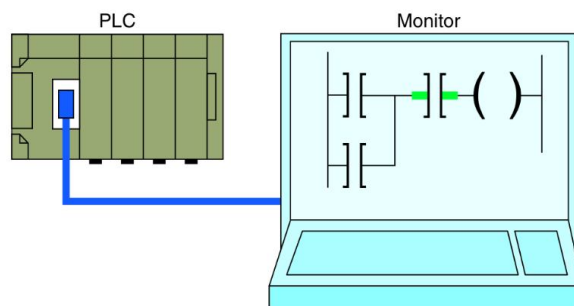


Figure 2.12 Control program can be displayed on a monitor in real time.

2.2.1 PLC Architecture

A typical PLC can be divided into parts, as illustrated in Figure 2.13. These are the central processing unit (CPU), the input/output (I/O) section the power supply, and the programming device. The term architecture can refer to PLC hardware, to PLC software, or to a combination of both. An open architecture design allows the system to be connected easily to devices and programs made by other manufacturers. Open architectures use off-the-shelf components that conform to approved standards. A system with a closed architecture is one whose design is proprietary, making it more difficult to connect to other systems.

Most PLC systems are in fact proprietary, so you must be sure that any generic hardware or software you may use is compatible with your particular PLC. Also, although the principal concepts are the same in all methods of programming, there might be slight differences in addressing, memory allocation, retrieval, and data handling for different models. Consequently, PLC programs cannot be interchanged among different PLC manufacturers.

There are two ways in which I/Os (Inputs/Outputs) are incorporated into the PLC: fixed and modular. Fixed I/O (Figure 2-14) is typical of small PLCs that come in one package with no separate, removable units. The processor and I/O are packaged together, and the I/O terminals will have a fixed number of connections built in for inputs and outputs. The main advantage of this type of packaging is lower cost. The number of available I/O points varies and usually can be expanded by buying additional units of fixed I/O. One disadvantage of fixed I/O is its lack of flexibility; you are limited in what you can get in the quantities and types dictated by the packaging. Also for some models, if any part in the unit fails, the whole unit has to be replaced.

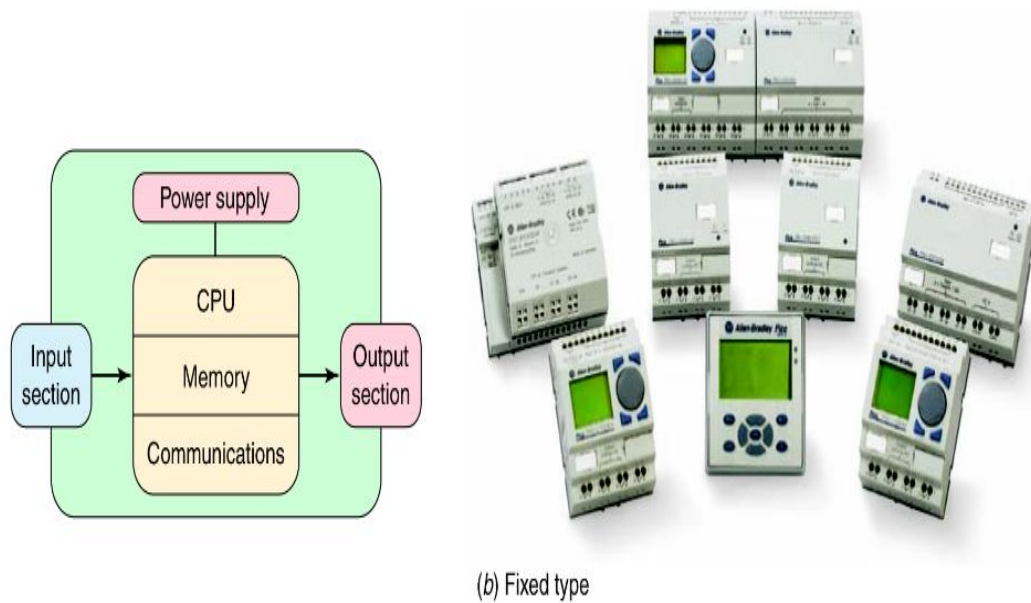
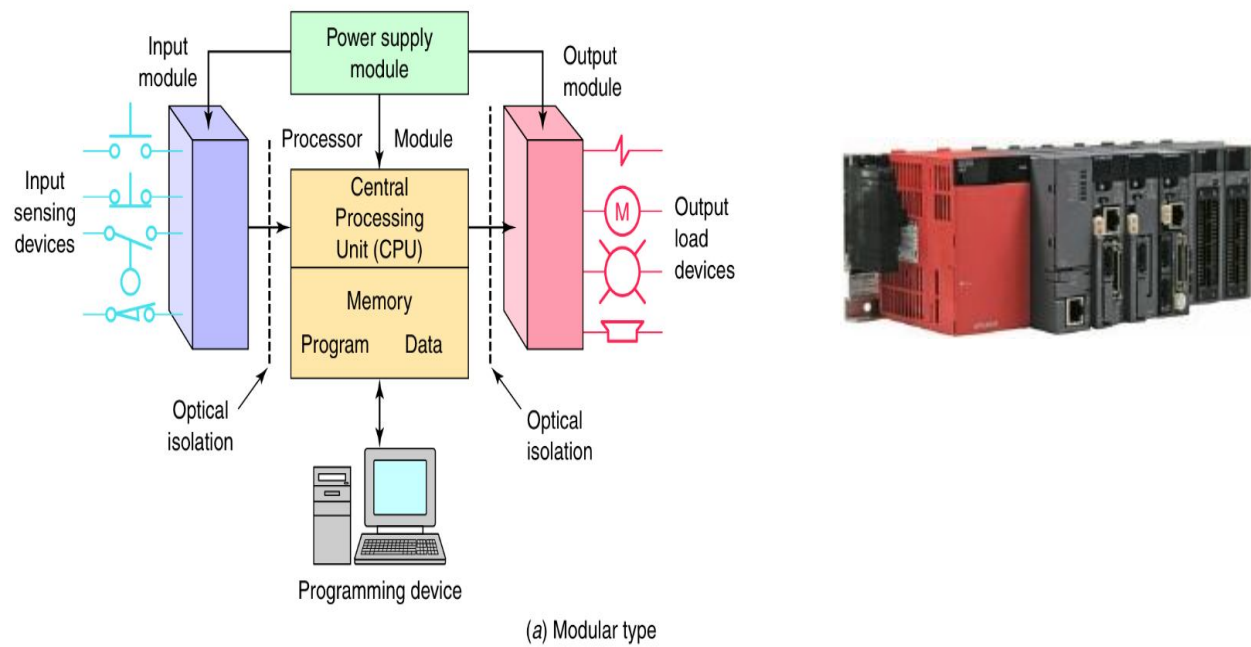


Figure 2.13 Typical parts of a programmable logic controller.

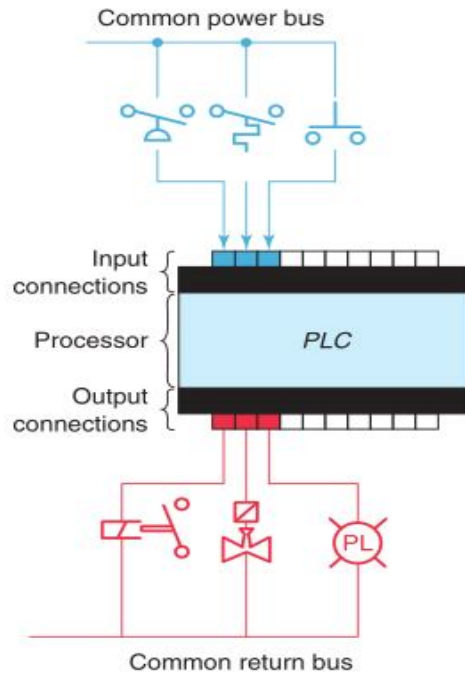


Figure 2.14 Fixed I/O configurations.

Modular I/O (Figure 2.15) is divided by compartments into which separate modules can be plugged. This feature greatly increases your options and the unit's flexibility.

You can choose from the modules available from the manufacturer and mix them any way you desire. The basic modular controller consists of a rack, power supply, processor module (CPU), input/output (I/O modules), and an operator interface for programming and monitoring. The modules plug into a rack. When a module is slid into the rack, it makes an electrical connection with a series of contacts called the backplane, located at the rear of the rack.

The PLC processor is also connected to the backplane and can communicate with all the modules in the rack.

The power supply supplies DC power to other modules that plug into the rack (Figure 2.16). For large PLC systems, this power supply does not normally supply power to the field devices. With larger systems, power to field devices is provided

by external alternating current (AC) or direct current (DC) supplies. For some small micro PLC systems, the power supply may be used to power field devices.

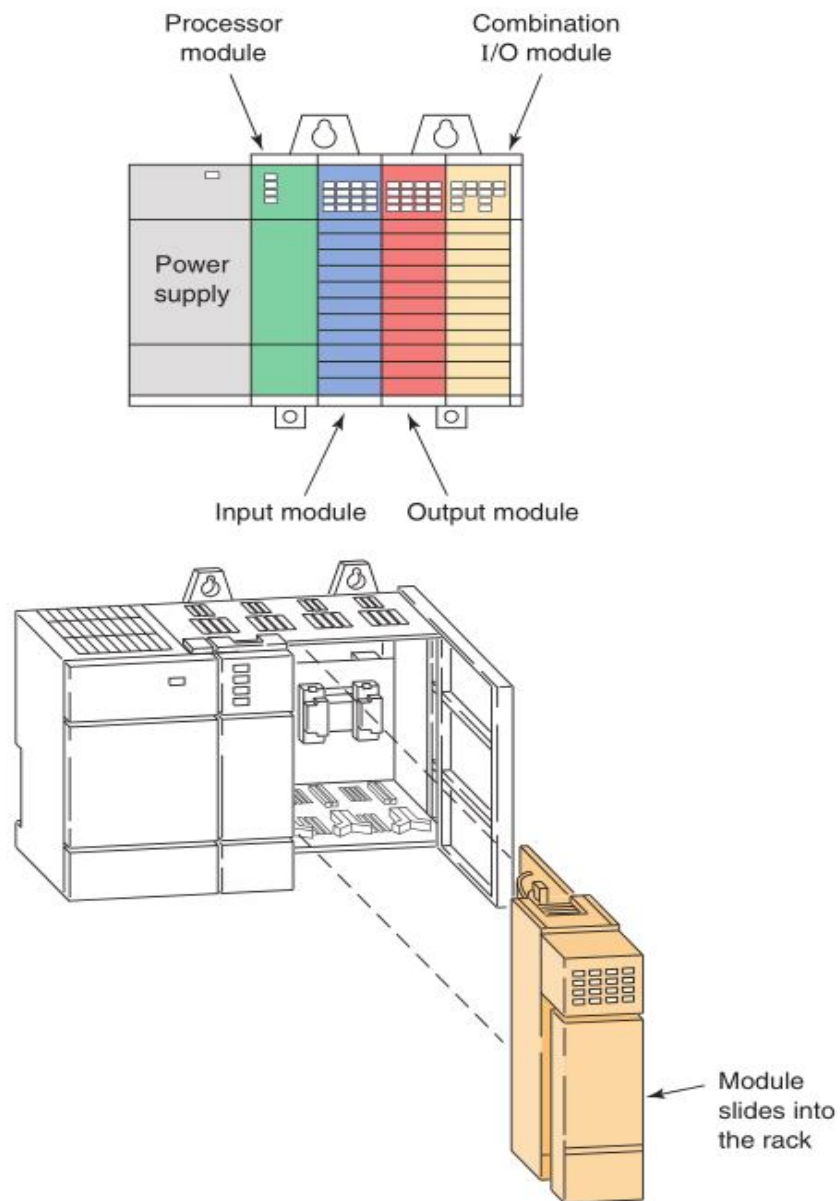


Figure 2.15 Modular I/O configurations



Figure 2.16 The power supply supplies DC power to other modules that plug into the rack.

The processor (CPU) is the “brain” of the PLC. A typical processor (Figure 2.17) usually consists of a microprocessor for implementing the logic and controlling the communications among the modules. The processor requires memory for storing the results of the logical operations performed by the microprocessor. Memory is also required for the program EPROM or EEPROM plus RAM.



Figure 2.17 Typical PLC processor modules.

The CPU controls all PLC activity and is designed so that the user can enter the desired program in relay ladder logic. The PLC program is executed as part of a repetitive process referred to as a scan (Figure 2.18). A typical PLC scan starts with the CPU reading the status of inputs.

Then, the application program is executed. Once the program execution is completed, the CPU performs internal diagnostic and communication tasks. Next, the status of all outputs is updated. This process is repeated continuously as long as the PLC is in the run mode.

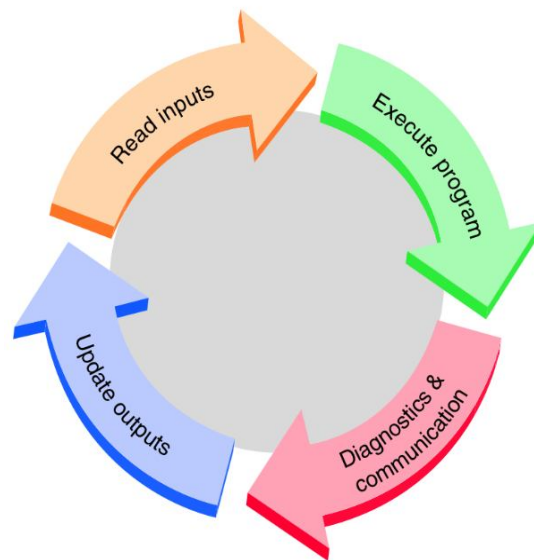


Figure 2.18 Typical PLC scans cycle.

The I/O system forms the interface by which field devices are connected to the controller (Figure 2.19). The purpose of this interface is to condition the various signals received from or sent to external field devices. Input devices such as pushbuttons, limit switches, and sensors are hardwired to the input terminals.

Output devices such as small motors, motor starters, solenoid valves, and indicator lights are hardwired to the output terminals. To electrically isolate the internal components from the input and output terminals, PLCs commonly employ an optical isolator, which uses light to couple the circuits together.

The external devices are also referred to as “field” or “real-world” inputs and outputs. The terms field or real world are used to distinguish actual external devices that exist and must be physically wired from the internal user program that duplicates the function of relays, timers, and counters.

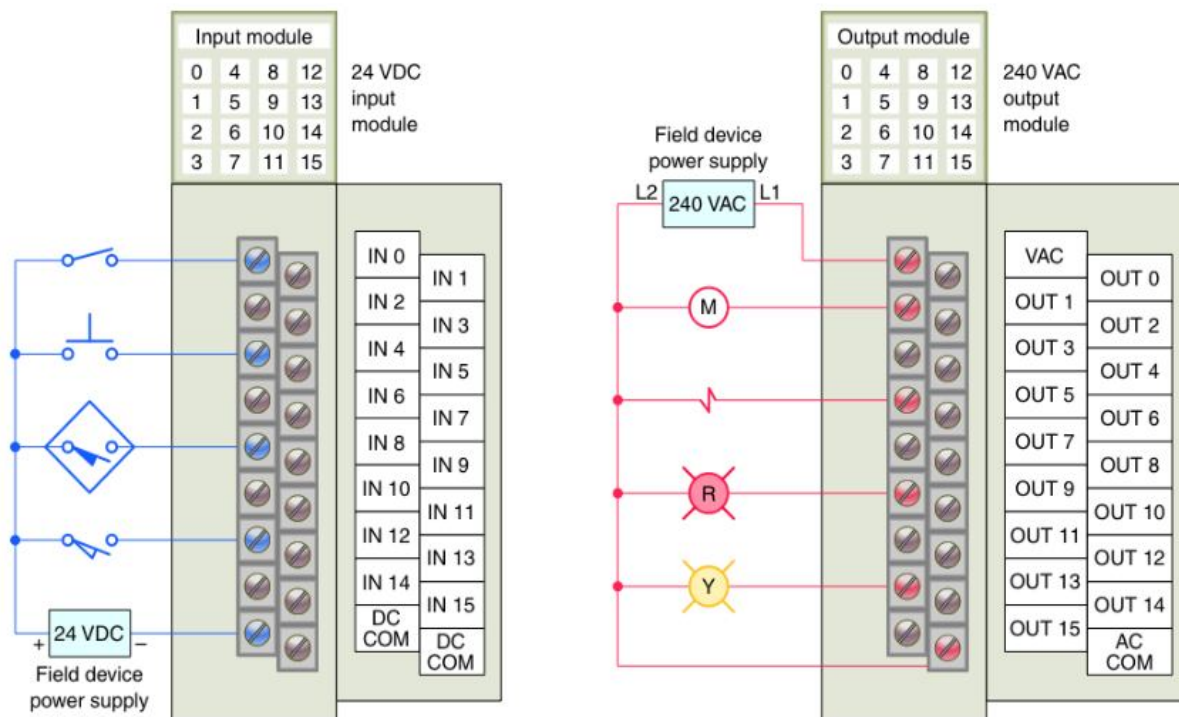


Figure 2.19 Typical PLC input/output (I/O) system connections.

2.2.2 Program Scan

When a PLC executes a program, it must know in real time when external devices controlling a process are changing. During each operating cycle, the processor reads all the inputs, takes these values, and energizes or de-energizes the outputs according to the user program.

This process is known as a program scan cycle. Figure 2.20 illustrates a single PLC operating cycle consisting of the input scan, program scan, output scan, and housekeeping duties. Because the inputs can change at any time, it constantly repeats this cycle as long as the PLC is in the RUN mode.

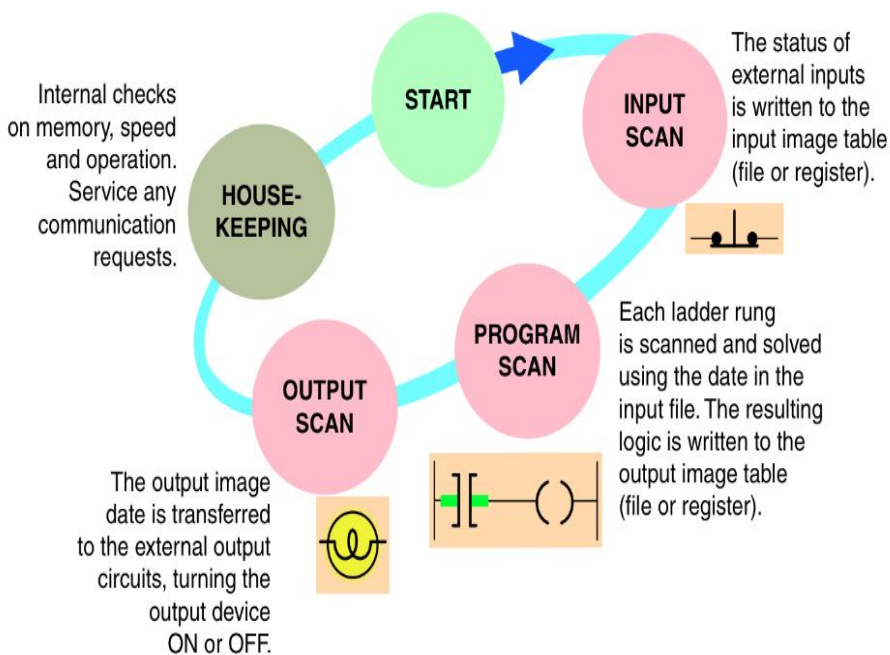


Figure 2.20 PLC program scan cycle.

The time it takes to complete a scan cycle is called the scan cycle time and indicates how fast the controller can react to changes in inputs. The time required to make a single scan can vary from about 1 millisecond to 20 milliseconds.

If a controller has to react to an input signal that changes states twice during the scan time, it is possible that the PLC will never be able to detect this change. For example, if it takes 8 ms for the CPU to scan a program, and an input contact is opening and closing every 4 ms, the program may not respond to the contact changing state. The CPU will detect a change if it occurs during the update of the input image table file, but the CPU will not respond to every change. The scan time is a function of the following:

- The speed of the processor module
- The length of the ladder program
- The type of instructions executed
- The actual ladder true/false conditions

The actual scan time is calculated and stored in the PLC's memory. The PLC computes the scan time each time the END instruction is executed. Scan time data can be monitored via the PLC programming. Typical scan time data include the maximum scan time and the last scan time.

2.2.3 PLC Programming Languages

The term PLC programming language refers to the method by which the user communicates information to the PLC. The standard IEC 61131 (Figure 2.21) was established to standardize the multiple languages associated with PLC programming by defining the following five standard languages:

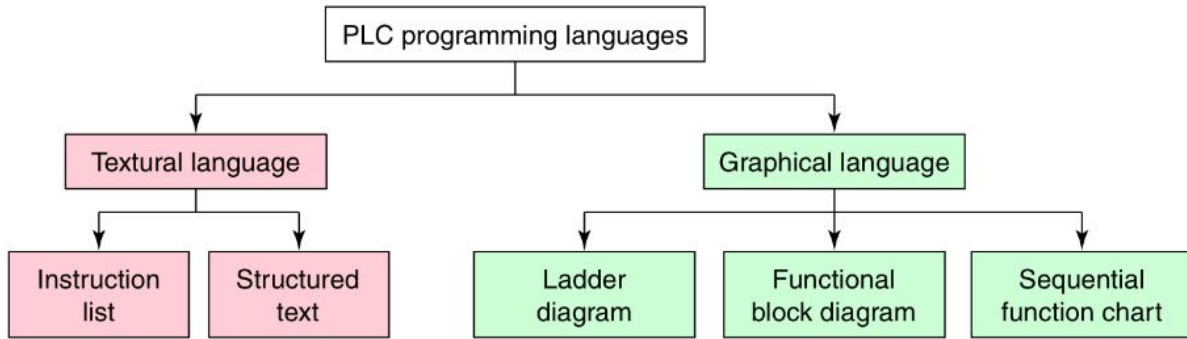


Figure 2.21 Standards IEC 61131 languages associated with PLC programming.

- Ladder Diagram (LD): a graphical depiction of a process with rungs of logic, similar to the relay ladder logic schemes that were replaced by PLCs.
- Function Block Diagram (FBD): a graphical depiction of process flow using simple and complex interconnecting blocks.
- Sequential Function Chart (SFC): a graphical depiction of interconnecting steps, actions, and transitions.
- Instruction List (IL): a low-level, text-based language that uses mnemonic instructions.
- a high-level, text-based language such as BASIC, C, or PASCAL specifically developed for industrial control applications.

Ladder diagram language is the most commonly used PLC language and is designed to mimic relay logic. The ladder diagram is popular for those who prefer to define control actions in terms of relay contacts and coils, and other functions as block instructions. Figure 2.22 shows a comparison of ladder diagram programming and instruction list programming. Figure 2.22a shows the original relay hardwired control circuit. Figure 2.22b shows the equivalent logic ladder

diagram programmed into a controller. Note how closely the ladder diagram program closely resembles the hardwired relay circuit.

The input/output addressing is generally different for each PLC manufacturer. Figure 2.22c show how the original hardwired circuit could be programmed using the instruction list programming language. Note that the instructional list consists of a series of instructions that refer to the basic AND, OR, and NOT logic gate functions.

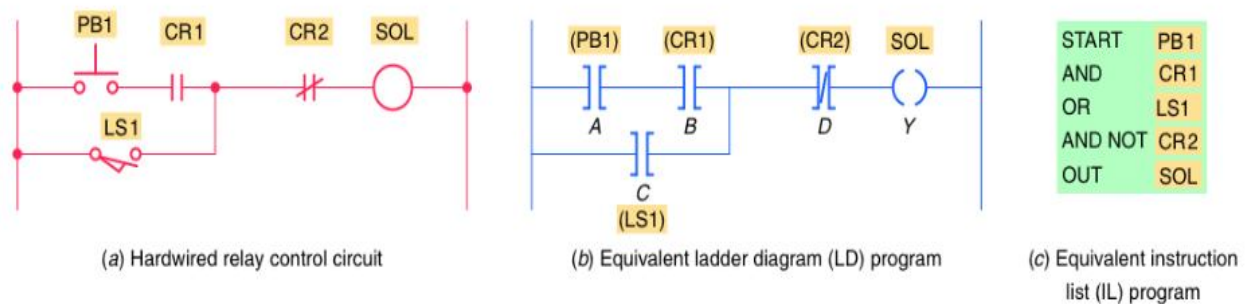


Figure 2.22 Comparison of ladder diagram and instruction list programming.

• Function Block Diagram (FBD)

Functional block diagram programming uses instructions that are programmed as blocks wired together on screen to accomplish certain functions. Typical types of function blocks include logic, timers, and counters. Functional block diagrams are similar in layout to electrical/electronic block diagrams used to simplify complex systems by showing blocks of functionality. The primary concept behind a functional block diagram is data flow.

Function blocks are linked together to complete a circuit that satisfies a control requirement. Data flow on a path from inputs, through function blocks or instructions, and then to outputs.

The use of function blocks for programming of programmable logic controllers (PLCs) is gaining wider acceptance. Rather than the classic contact and coil representation of ladder diagram or relay ladder logic programming, function blocks present a graphical image to the programmer with underlying algorithms already defined.

The programmer simply completes needed information within the block to complete that phase of the program. Figure 2.23 shows function block diagram equivalents to ladder logic contacts.

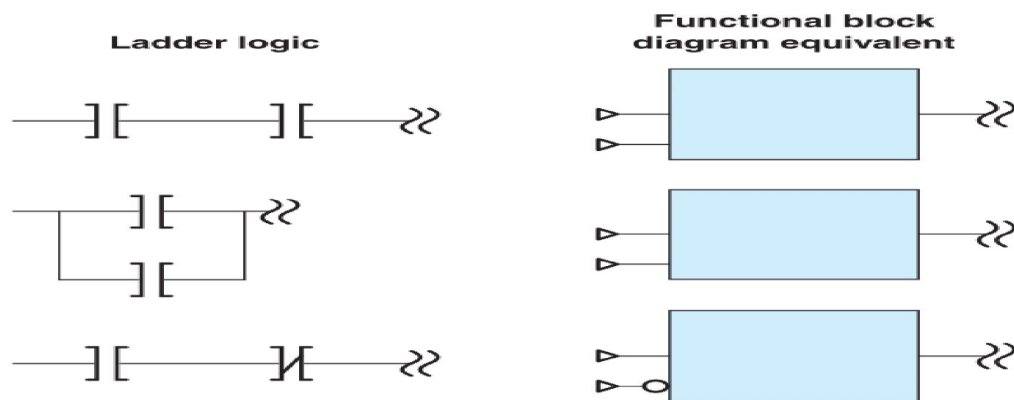


Figure 2.23 Function block diagram equivalents to ladder

All parameters of the function blocks are defined in the function block diagram. Clearly structured and easy to understand parameter dialogs, in which all block-specific entries can be made, are available.

Once completed, the function block diagram can be verified using a plausibility check for errors or syntactic accuracy. Any errors or warnings are displayed in a list, and it is possible to navigate directly to the source of the error by simply clicking on the relevant line in that list.

- **Blocks**

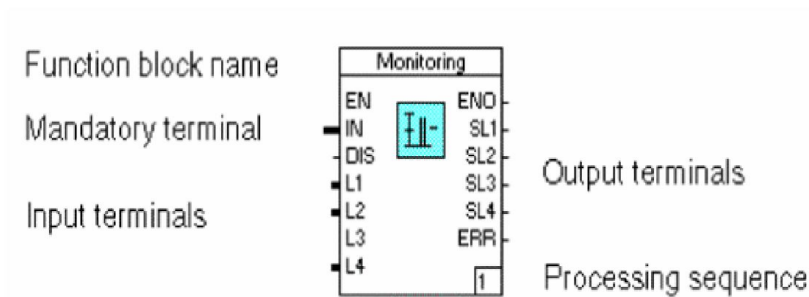


Figure 2.24 Block diagram parts

Block in Figure 2.24 consist of following :

- **Frame:** The block frame limits the selector area of the block. From its color one can establish whether the block has been selected or parameterized in any incorrect manner.
- **Function block name :** Unlike the functions, all function blocks are displayed with a tag name (max. 16 characters). All block names can be found in the system wide tag list.
- **Icons :** An icon is used to symbolize the block type of a function block - a function abbreviation that of a function.
- **Terminals :** Here a distinction must be made between inputs and outputs. Corresponding to the signal flow, inputs are always displayed on the left and outputs on the right. Just as in the case of the signal flow lines, the color and line width of the terminals reveal information on the data type needed/set.
- **Mandatory/ Optional parameters :** Mandatory terminals call for data supply via the signal flow line in order to enable the block to operate correctly, while this does not apply for optional connections. Optional terminals are displayed shorter

for the purpose of differentiation. Some optional terminals disappear altogether due to the parameter definition of fixed values.

- Terminal designation: An abbreviation next to each function block terminal denotes the terminal's function, for example EN for enable.
- Processing sequence: The code on the lower right of the block indicates the processing sequence within the program.
- **Scale Change, SCAL Block Diagram**

The scaling of an analog signal can be changed with this function block Figure 2.25. This is needed for adjusting the scales of several analog values which must be compared or calculated. Signals lying outside the new Measuring range due to adjustment of scaling, are limited between 0 and 100% of the new measuring range.

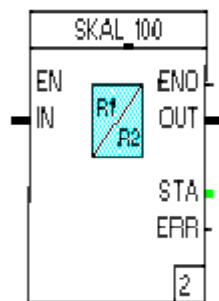


Figure 2.25 Scale Change block diagram

- **Individual Drive Function for Bi-directional Units Block Diagram**

The dual-channel individual drive function Figure 2.26. relays control commands to technical control equipment (e.g. electric motor with two directions of rotation; valve with stop position) in the process. The field device connected can be configured.

The function has two separate inputs (IN0, IN1) and outputs (OU0, OU1) for the command outputs 0 and 1.

If both inputs are deleted or are on logic-1 signal, this will be considered a stop command. The control commands can come from a higher-ranking control (automatic operating mode) or can take place manually (manual operating mode).

The operating mode inputs MM and MA have priority over operating mode selection, i.e. the selection of the manual/automatic operating modes is enabled only if both inputs are on logic-0 signal.

If both inputs are on logic-1 signal, the manual operating mode has priority over automatic.

The output of control commands can be made dependent on safety, inhibit, local intervention and fault signals.

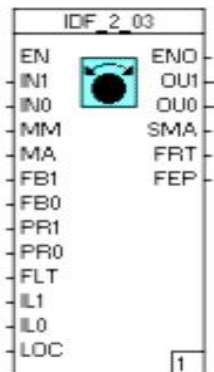


Figure 2.26 Individual Drive Function for Bi-directional Units block diagram

After the control command has changed, the time is monitored as run time until feedback occurs (end position reached). The feedback can be external (feedback variable available) or internal. If the run time is exceeded, an error message appears.

Arrival at the end positions of the two outputs is signaled to inputs FB0 and FB1. A dead time can be configured for the feedback inputs FB0 and FB1 (default = 0 ms). This can prevent an end-position being triggered by delayed feedback in the event of a STOP near the end position. The dead time is included in the configured run time.

- Inhibition inputs : Changes in the control command for commands 0 and 1 can be prevented separately using inputs IL0 and IL1 for the output OU0 and OU1. The inhibition applies in manual and automatic operating modes.

Safety intervention .

The inputs PR0 and PR1 act directly on the control command.

The following table 2.1 shows the method of operation:

Table 2.1 Method Operation of Inhibition inputs

<i>Input PR0</i>	<i>Input PR1</i>	<i>Control command</i>
0	0	No safety
0	1	Safety OU1
1	0	Safety OU0
1	1	Safety stop

- **Analog Monitoring Block Diagram**

Representation of an analog input signal and monitoring of this signal for up to 4 limit values Figure 2.27. With the input DIS (logic-1 signal) the monitoring of the limit values can be removed from the processing. The limit values can be fixed values or analog signals.

For each limit value various types of limit value processing can be selected (messages). In the event that a limit value is violated, the configured message text is outputted. A colored envelope characterizes the violation of the limit value in the faceplate.

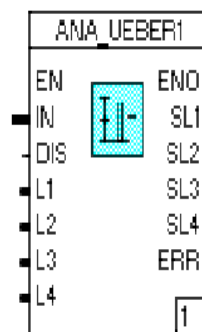


Figure 2.27 Analog Monitoring Block Diagram

- **Timer, Switch-on Delay, TON Block Diagram**

This block Figure 2.28 is used for time control and monitoring of certain operating states. It delays switch-on TON (Timer on Delay) by a configurable time. Switch-off is relayed un delayed.

A premature reset of the delay time is possible with the RESET-signal RES. The delay time is reactivated after resetting the reset signal. Activation of positive or negative edge can be configured.

The set delay time can be interrogated at output TS and the elapsed time can be interrogated at output TC in the TIME format.

A message can be generated at logic 1 signal of output OUT.

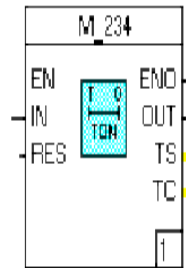


Figure 2.28 Timer, Switch-on Delay (TON)

- **Handling the parameter definition masks**

By virtue of the different parameters governing the various function blocks, there is no uniform parameter definition mask. However, certain sections are used similarly in all or in some parameter definition masks. Besides, there are several parameter definition masks for large blocks and they can be edited in any order desired.

Using the parameter definition masks of the function block "Continuous ratio controller C_CR" Figure 2.29 the basic features are outlined below:

- General data: Name, short designation of block, if necessary number of parameter definition mask currently in use.
- Group: Some parameters are classified in groups e.g. the message values. The parameters are placed in a frame and a group name portrays the parameter function in the upper frame corner.
- Input field color :
 - Red background: Mandatory parameters
 - Blue background: Marked for overwrite

- Data field : For example for entering parameters such as measuring range start and measuring range end. In the case of parameters that can also be specified externally, data can only be entered if no signal flow line is connected to the respective terminal. Conversely, the terminal disappears from the block display if a parameter has been entered. Please consult the block description for the parameters to which this applies.

Parameters: Continuous controller, ratio C_CR (1/4)

General data

Name: FIC022 Short text: Processing: ☒

Long text: Sequence: 5

Range start: 0.0 Range end: 100.0 Dim:

Ratio SCS: 0.5 Ratio SCE: 10.0 Dim:

Int. set point: 0.0 ☒ Access

Output: 0.0 ☒ Access ☐ Lambda control

Ratio: 1.0 ☒ Access Spec. ratio: 1.0

Bias: 0.0 ☒ Access

Messages

No.	Type	Value	Access	Hyst.	Prio.	Hint	Message text
1			<input type="checkbox"/>	0.3			
2			<input type="checkbox"/>	0.3			
3			<input type="checkbox"/>	0.3			
4			<input type="checkbox"/>	0.3			

OK Cancel Save Reset Check Help << >>

Figure 2.29 parameter definition masks

- List : there are lists where only the preset list entry is visible. The invisible list section can be opened out (arrow). The desired entry is taken over by clicking the input field.

Some lists have an input field that can be freely edited. In these cases, the arrow points away a little from the input field. The entered text is taken over into the list and is available also in all subsequent parameter definition masks of blocks of the same type.

2.3 Previous Studies

- **PLC System for Oil Refinery Control**

C. M. Davidson

Conf. Industrial Electronics, Control and Instrumentation, vol. ,2007

- Programmable Logic Control (PLC) system is always used to control small industries like water treatment stations; electric power stations and irrigation systems. Oil and gas refineries generally rely on a Distributed Control System (DCS) to provide all process and equipment control functionality. In this paper PLC system is used to control a whole oil refinery instead of the conventional control through DCS.

- **Adopting Distributed Control System To Control Desalting Process in Petroleum Refinery**

Rajesh Kumbara S. K1, Dr. D. N. Shivappa

International Journal of Emerging Technology and Advanced Engineering

Volume 3, Issue 10, October 2013

- This work on ‘Adopting Distributed Control System to Control Desalting Process in Petroleum Refinery’ is carried out at Yokogawa India Ltd, Bangalore. Earlier the desalting process in petroleum refinery was controlled using Programmable Logic Controller (PLC).

Since it has got some problems like control system flexibility and redundancy etc, company has decided to implement Distributed Control System (DCS) to automate the desalting process. Mined crude oil is the raw material to petroleum refinery industry which contains water and dissolved salts principally chlorides of sodium, calcium, and magnesium.

- **PLC Based Industrial Automation & Monitoring**

Prof. Burali Y. N.

International Journal of Engineering and Science

Vol. 1, Issue 3 (Sept 2012), PP 01-04

- In process industries, field buses have dominated the connection establishment between sensors, actuators and controllers. In large scale industries the number of sensors and actuators are greater in number where wiring of such components to the controller involves higher installation cost.

- **PLC-based SCADA System for Oil Storage and Application**

Wang Xibin

Proceedings of the 2012 Second International Conference on Electric Information and Control Engineering - Volume 02; 04/2012

- In the paper, configuration software and PLC are applied to construct SCADA System of the petroleum transportation. A petroleum storage and transportation database is created, which provide a large amount of experimental data to the research of petroleum storage and transportation and provide the basis for the promotion of new methods and new technologies.

- **A monitoring System For PLC Controlled Manufacturing System**

Liping Guo

Canadian Journal on Electrical and Electronics Engineering Vol. 4, No. 2 April 2013

- In this paper, we propose a monitoring system for Programmable Logic Controllers (PLCs) controlled manufacturing system based on CC-Link fieldbus in the process monitoring and control for film plating. The PLCs are mainly utilized for collecting data as well as realizing auto-tuning PID and sequence control

strategies. In addition, Mitsubishi GT Designer software with Human-Machine Interface (HMI) is used to monitor the dynamical process.

- **Temperature Monitoring System based on PLC**

Shoucheng Ding*, Wenhui L

College of Electrical and Information Engineering, Lanzhou University of Technology, Lanzhou 730050, Gansu, China

- The programmable logic controller (PLC) is an industrial control computer; it is the new automatic device inherited computer, automatic control technology and communication technology.

System temperature signal detected by the temperature sensor. The temperature transmitter will be the temperature value converted into a voltage signal of 0-10V into PLC. PLC voltage signal setting compared to the temperature deviation after PID operation; the system will issue a temperature control signal to reach the electric heater voltage control. So that it implements a continuous monitoring and control of the temperature. The temperature monitoring system in the industrial field has a certain value.

Chapter Three

SIMULATION HARDWARE DEVICES

The system hardware comprises of Operator station and Engineering Station and Process Station AC800F and field devices.

The AC 800F controller has a modular structure. The CPU is designed as a backplane to which various modules power supply units, Ethernet or fieldbus modules can be attached in line with the application.

On the fieldbus side, modules for PROFIBUS-DPV1,MODBUS (master/slave,RTU or ASCII), IEC 60870-5-101 and CAN for Freelance Rack I/O are available.

The fieldbus line and the connected field devices are entirely configured and parameterized using the engineering tool Control Builder F. No further external tools are needed for configuration. Fieldbus and device configuration can be performed offline without connection to the field devices.

Even a single AC 800F controller can be connected to both buses, Profibus and Foundation Fieldbus at the same time.

Furthermore Freelance Rack I/O can be connected to the AC 800F, in which case the CAN module is used.

This allows you to operate five I/O racks, typically up to 1000 I/Os, for each AC 800F. The I/O racks are equipped with a link module and up to nine I/O modules and can be mounted separately at a distance of up to 400 m from the AC 800F.

each of the hardware components is discussed in detail in this chapter.

3.1 The Central Processing Unit (CPU)

The central processing unit (CPU) is built into single unit fixed PLCs while modular rack types typically use a plug-in module. CPU, controller, and processor are all terms used by different manufacturers to denote the same module that performs basically the same functions. Processors vary in processing speed and memory options. A processor module can be divided into two sections: the CPU section and the memory section (Figure 3.1). The CPU section executes the program and makes the decisions needed by the PLC to operate and communicate with other modules. The memory section electronically stores the PLC program along with other retrievable digital information.

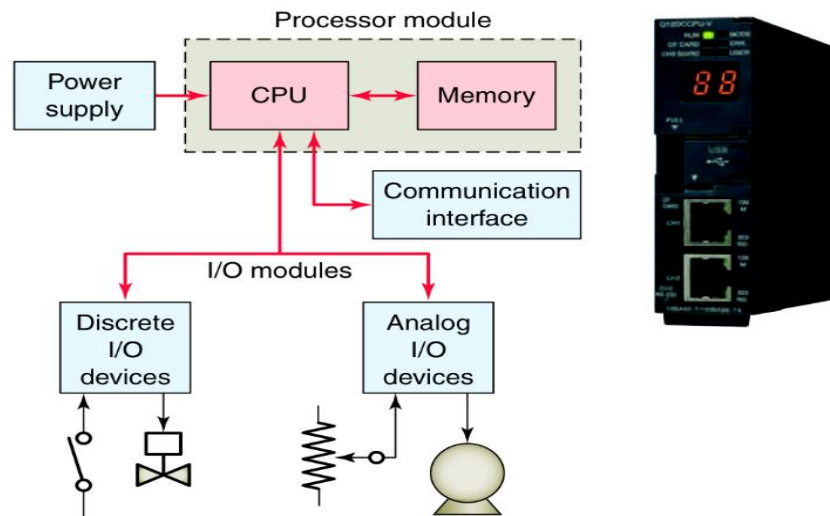


Figure 3.1 Sections of a PLC processor module.

The PLC power supply provides the necessary power (typically 5 VDC) to the processor and I/O modules plugged into the backplane of the rack (Figure 3.2). Power supplies are available for most voltage sources encountered. The power supply converts 115 VAC or 230 VAC into the usable DC voltage required by the

CPU, memory, and I/O electronic circuitry. PLC power supplies are normally designed to withstand momentary losses of power without affecting the operation of the PLC. Holdup time, which is the length of time a PLC can tolerate a power loss, typically ranges from 10 milliseconds to 3 seconds.

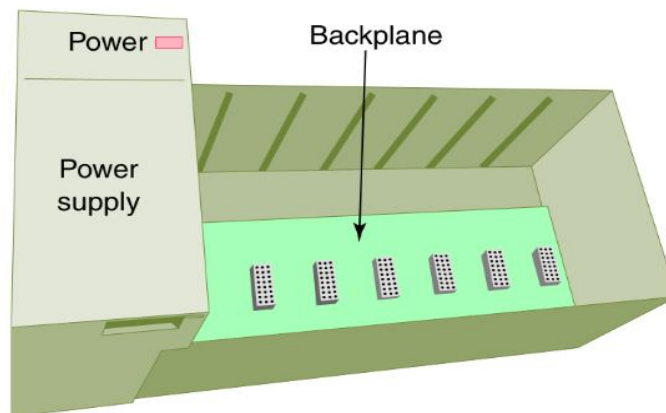


Figure 3.2 PLC power supply.

The CPU contains the similar type of microprocessor found in a personal computer. The difference is that the program used with the microprocessor is designed to facilitate industrial control rather than provide general purpose computing. The CPU executes the operating system, manages memory, monitors inputs, evaluates the user logic, and turns on the appropriate outputs.

Associated with the processor unit will be a number of status LED indicators to provide system diagnostic information to the operator (Figure 3.3). Also, a key switch may be provided that allows you to select one of the following three modes of operation: RUN, PROG, and REM.

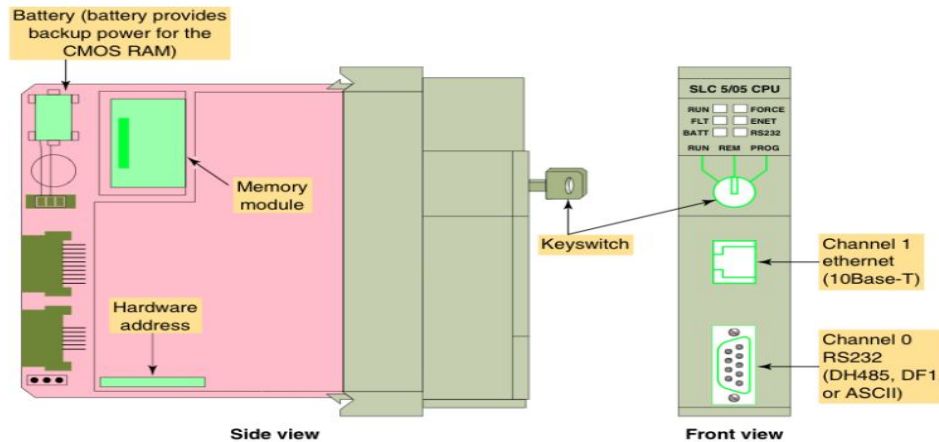


Figure 3.3 Typical processor modules.

The processor module also contains circuitry to communicate with the programming device. Somewhere on the module you will find a connector that allows the PLC to be connected to an external programming device. The decision-making capabilities of PLC processors go far beyond simple logic processing. The processor performs other functions such as timing, counting, latching, comparing, motion control and complex math functions.

3.2 The I/O Section

The input/output (I/O) section of a PLC is the section to which all field devices are connected and provides the interface between them and the CPU. Input/output arrangements are built into a fixed PLC while modular types use external I/O modules that plug into the PLC.

Figure 3.4 illustrates a rack-based I/O section made up of individual I/O modules. Input interface modules accept signals from the machine or process devices and convert them into signals that can be used by the controller.

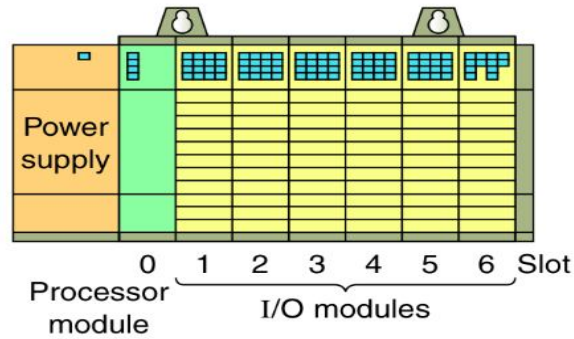


Figure 3.4 Rack-based I/O section.

Output interface modules convert controller signals into external signals used to control the machine or process. A typical PLC has room for several I/O modules, allowing it to be customized for a particular application by selecting the appropriate modules. Each slot in the rack is capable of accommodating any type of I/O module.

The I/O system provides an interface between the hardwired components in the field and the CPU. The input interface allows status information regarding processes to be communicated to the CPU, and thus allows the CPU to communicate operating signals through the output interface to the process devices under its control.

Input and output modules can be placed anywhere in a rack, but they are normally grouped together for ease of wiring. I/O modules can be 8, 16, 32, or 64 point cards (Figure 3.5). The number refers to the number of inputs or outputs available. The standard I/O module has eight inputs or outputs. A high-density module may have up to 64 inputs or outputs. The advantage with the high-density module is that it is possible to install up to 64 inputs or outputs in one slot for greater space savings. The only disadvantage is that the high-density output modules cannot handle as much current per output.



Figure (3.5) 16, 32, and 64 point I/O modules.

3.3 Analog I/O Modules

Earlier PLCs were limited to discrete or digital I/O interfaces, which allowed only on/off-type devices to be connected. This limitation meant that the PLC could have only partial control of many process applications. Today, however, a complete range of both discrete and analog interfaces are available that will allow controllers to be applied to practically any type of control process.

Discrete devices are inputs and outputs that have only two states: on and off. In comparison, analog devices represent physical quantities that can have an infinite number of values. Typical analog inputs and outputs vary from 0 to 20 milliamps, 4 to 20 milliamps, or 0 to 10 volts.

Figure 3.6 illustrates how PLC analog input and output modules are used in measuring and displaying the level of fluid in a tank. The analog input interface module contains the circuitry necessary to accept an analog voltage or current signal from the level transmitter field device. This input is converted from an analog to a digital value for use by the processor. The circuitry of the analog output module accepts the digital value from the processor and converts it back to an analog signal that drives the field tank level meter.

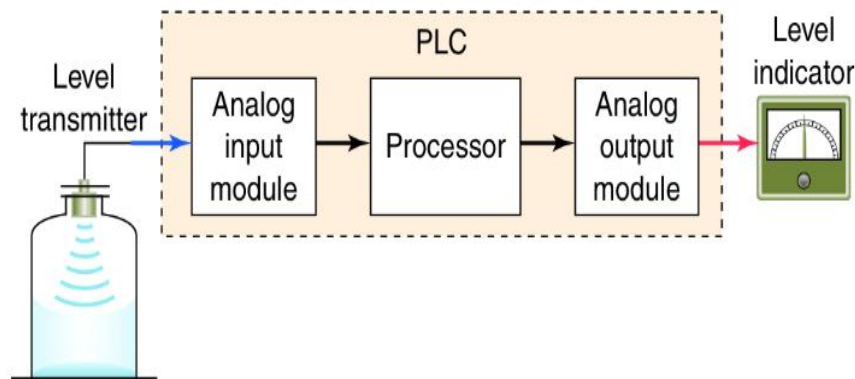
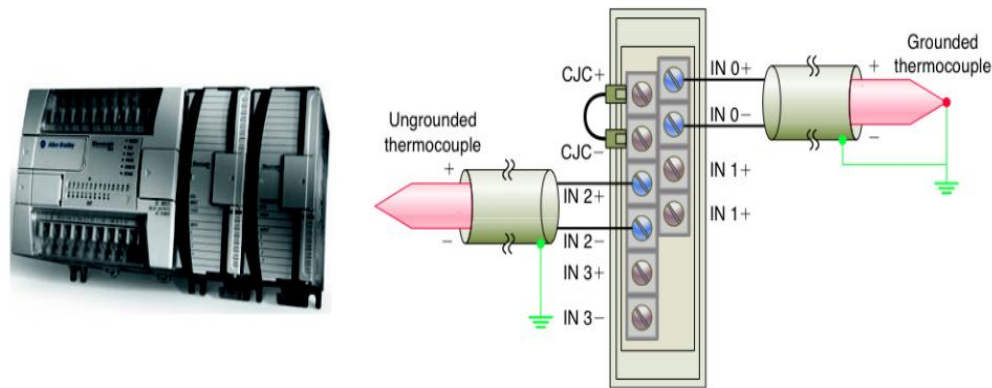


Figure 3.6 Analog input and output to a PLC.

Analog input modules normally have multiple input channels that allow 4, 8, or 16 devices to be interface to the PLC. The two basic types of analog input modules are voltage sensing and current sensing. Analog sensors measure a varying physical quantity over a specific range and generate a corresponding voltage or current signal. Common physical quantities measured by a PLC analog module include temperature, speed, level, flow, weight, pressure, and position. For example, a sensor may measure temperature over a range of 0 to 500°C, and output a corresponding voltage signal that varies between 0 and 50 mV.

Figure 3.7 illustrates an example of a voltage sensing input analog module used to measure temperature. A varying DC voltage in the low mill volt range, proportional to the temperature being monitored, is produced by the thermocouple. This voltage is amplified and digitized by the analog input module and then sent to the processor on command from a program instruction. Because of the low voltage level of the input signal, a twisted shielded pair cable is used in wiring the circuit to reduce unwanted electrical noise signals that can be induced in the conductors from other wiring. When using an ungrounded thermocouple, the shield must be connected to ground at the module end. To obtain accurate readings from each of

the channels, the temperature between the thermocouple wire and the input channel must be compensated for. A cold junction compensating (CJC) thermistor is integrated in the terminal block for this purpose.



Figure(3.7) 4-channel analog thermocouple input module.

When connecting voltage sensing inputs, close adherence to specified requirements regarding wire length is important to minimize signal degrading and the effects of electromagnetic noise interference induced along the connecting conductors. Current input signals, which are not as sensitive to noise as voltage signals, are typically not distance limited. Current sensing input modules typically accept analog data over the range of 4 mA to 20 mA, but can accommodate signal ranges of -20 mA to 120 mA. The loop power may be supplied by the sensor or may be provided by the analog output module as illustrated in Figure 3.8 . Shielded twisted pair cable is normally recommended for connecting any type analog input signal.

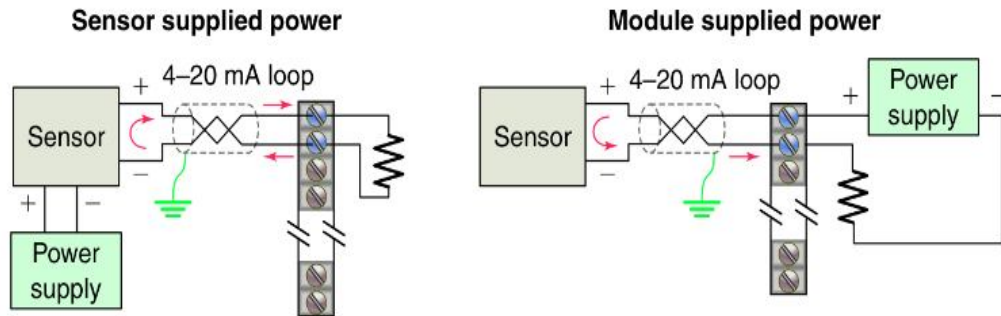


Figure 3.8 Sensor and analog module supplied power.

The analog output interface module receives from the processor digital data, which are converted into a proportional voltage or current to control an analog field device.

The transition of a digital signal to analog values is accomplished by a digital-to-analog (D/A) converter, the main element of the analog output module. An analog output signal is a continuous and changing signal that is varied under the control of the PLC program. Common devices controlled by a PLC analog output module include instruments, control valves, chart recorder, electronic drives, and other types of control devices that respond to analog signals.

Figure 3.9 illustrates the use of analog I/O modules in a typical PLC control system. In this application the PLC controls the amount of fluid placed in a holding tank by adjusting the percentage of the valve opening. The analog output from the PLC is used to control the flow by controlling the amount of the valve opening. The valve is initially open 100 percent. As the fluid level in the tank approaches the preset point, the processor modifies the output, which adjusts the valve to maintain a set point.

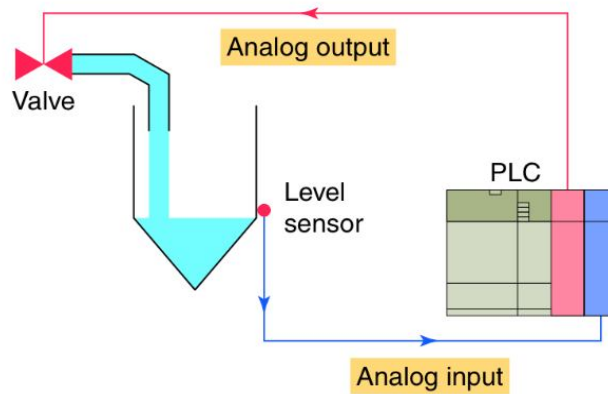


Figure 3.9 Typical analog I/O control system.

3.4 Freelance 800F Architecture

Freelance 800F provides an operator station and a process station. The operator station contains the functions for operation and observation, archives and logs, trends and alarms. Open-loop and closed-loop control functions are processed in the controllers which communicate with actors and sensors in the field.

The operator station PCs can also be used as engineering station. A permanent connection to the engineering system is not necessary.

3.4.1 Operator Station

The Operator station is based on a PC with Microsoft Windows 2000 Operating System and the Digivis Operator station Software.

Control Builder F and Digivis can be used simultaneously on the same PC.

3.4.2 Engineering Station

The Engineering Station is based on a PC with the Microsoft windows 2000 Operating system and the Control Builder F Engineering Software.

It is used by the operator for the system configuration, commissioning and Documentation. After these tasks have been completed, the Engineering station can be disconnected and used for other purposes.

3.4.3 Process Station – AC800F

AC 800F opens up the flexibility of Fieldbus technology of the user. The AC 800F collects and processes diagnostic and process data from four Fieldbus lines which may be different types .

It does this in addition to the task of conventional process station .Up to four different Fieldbus modules can be plugged in to the AC 800F .

The communication with the controllers runs via Ethernet.

AC 800F Figure 3.10 can be equipped with a set of fieldbus modules, covering all major fieldbuses used in process automation. With AC 800F you have the option to run these process stations either redundantly (CPU redundancy, fieldbus module redundancy) or without redundancy. Modular plug-in input/output modules are used in accordance with the type and quantity of process signals. With AC 800F,fieldbus-compliant components such as remote I/O, field devices, and network components can be used. ABB offers equipment for applications covering standard and hazardous areas.

The AC 800F consists of:

- The housing with CPU board and module slots
- The power supply module,
- At least one Ethernet module
- A maximum of four fieldbus modules

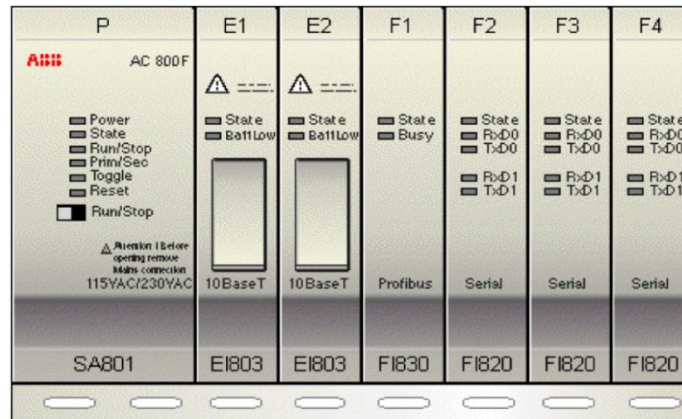


Figure 3.10 AC 800F

- **Base Unit with CPU Board – PM802F**

Base unit with CPU board – PM802F Figure 3.11 is the basic unit cyclically scan signals from the fieldbus sensors via the corresponding fieldbus modules ,processes these signals according the application , programs installed by the user and sends appropriate signals to the field bus actuators via the fieldsbus modules.

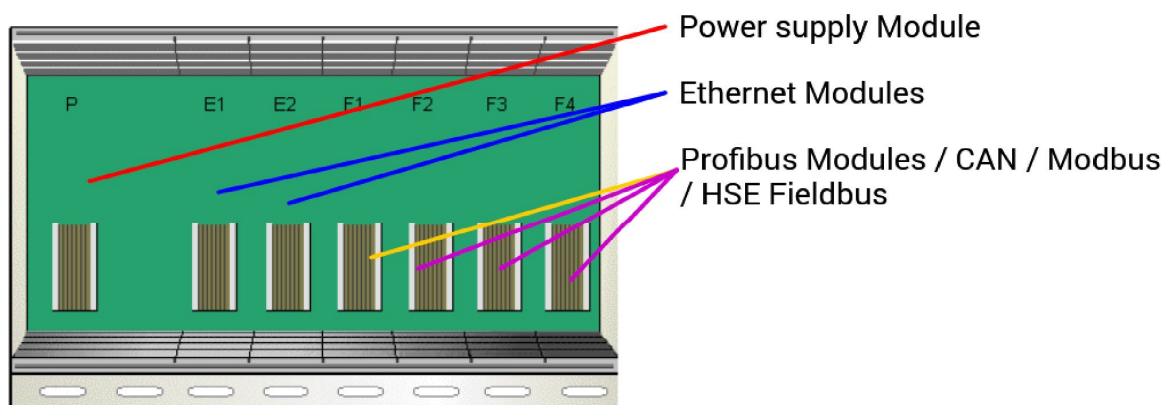
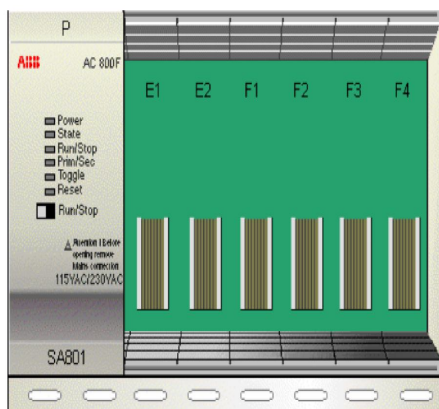


Figure 3.11 Base unit with CPU board – PM802F

- **Power Supply Module – SA 801F**

The AC800F modules are supplied with 5VDC/5A and 3.3VDC/5A auxiliary power supply Figure 3.12 . The power supply has open circuit overloaded and sustained short circuit protection . The electronically controlled output voltage provides high stability and low residual ripple

In case of power loss 20ms the power supply module generates power fail signal this signal is used by the CPU module to shut down operations and enter to a safe state .This required for a controlled restart of the system and the user application when power is restored . The output voltage remains within its tolerance limits for at least 15 ms.



SA 801F:
Input voltages from 115 ... 230 V AC



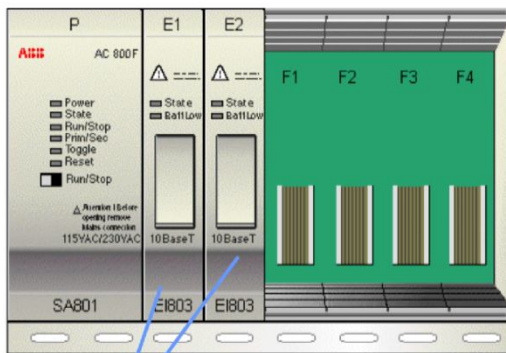
Figure 3.12 Power supply module SA 801F

- **Ethernet Modules – EI 803F**

These communication modules Figure 3.13 provide Ethernet communication to the system bus compliant with IEEE802.1 standard .

Ethernet Protocol is an open communications protocol based on the Common Industrial Protocol (CIP) layer used in both DeviceNet and ControlNet. It allows users to link information seamlessly between devices running the EtherNet protocol without custom hardware.

Communication module ,compliant with 10Base T shielded Twisted Pair.



EI 803F:
10BaseT module for Shielded Twisted Pair cables for PM 802F



Figure 3-13 Ethernet modules EI 803F

- **Profibus Module – FI 830F**

The FI830F module Figure 3.14 interfaces to the Profibus fieldbus . It provides functionality according to the PROFIBUS-DP V1 standard .

Profibus is a field bus standard for applications in the manufacturing industry, process automation and building automation. The Profibus is defined in the

standard EN 50 170. This standard is supplemented by technical guidelines published by Profibus Nutzerorganisation e.V. (PNO-Profibus User Organization). The module is the master on the profibus line in and allows connecting up to 126 profibus slaves configuration and parameterization is carried out completely with Line redundancy can be achieved using external configuration tools are required.

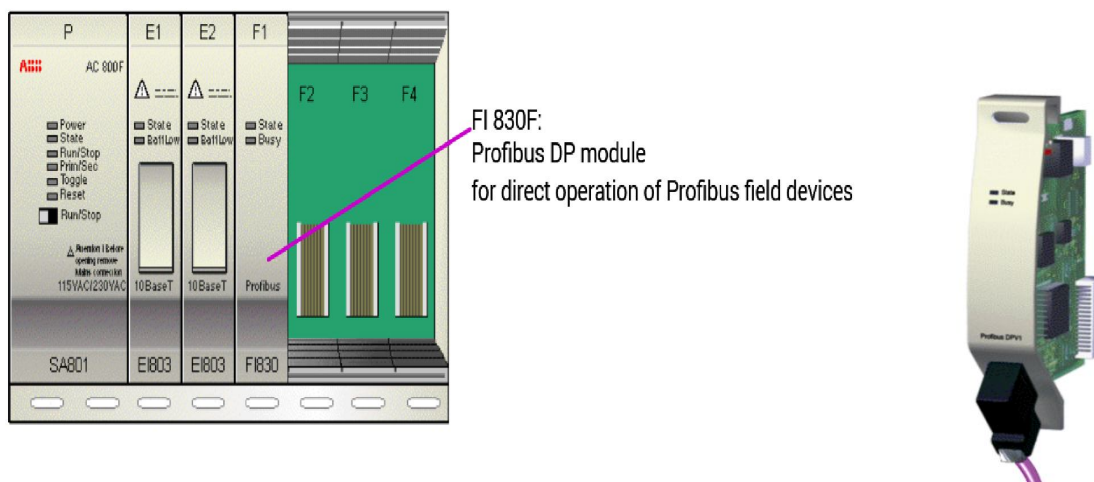


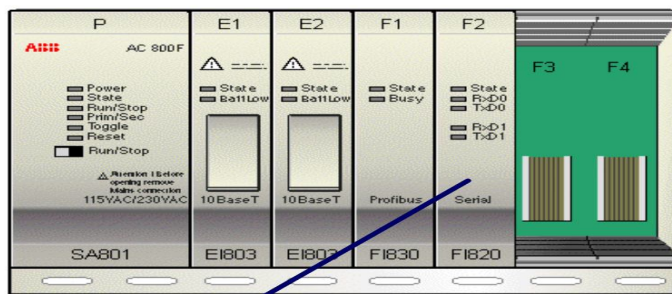
Figure 3.14 Profibus module FI 830F

- **Serial Interface Module – FI 820F**

The FI820F module Figure 3.15 provides connectivity to a variety of serial fieldbus and serial Protocols.

Standard protocol is MODBUS by using different connection cables the physical interface can easily be selected RS485(half duplex) RS422(full duplex) or RS232. MODBUS Protocol is a messaging structure, widely used to establish master-slave communication between intelligent devices.

Modbus is a serial communication protocol originally developed by Modicon for use with its PLCs. Basically, it is a method used for transmitting information over serial lines between electronic devices. The device requesting the information is called the Modbus Master and the devices supplying information are Modbus Slaves. Modbus is an open protocol, meaning that it's free for manufacturers to build into their equipment without having to pay royalties.



FI820 F – SERIAL COMMUNICATION MODULE

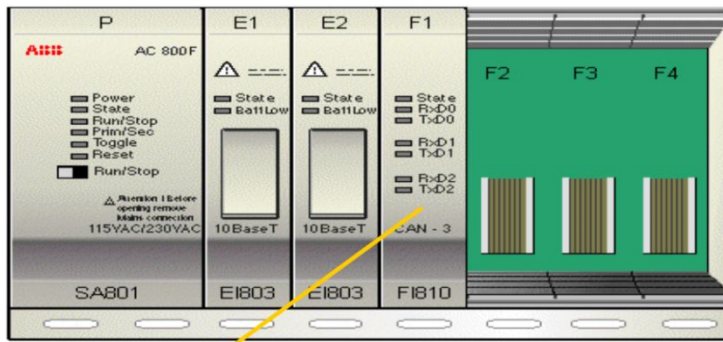
Figure 3.15 Serial Interface module FI 820F

- **CAN Module – FI 810**

The CAN FI810 module provides connectivity to the freelance 2000 rack I/O. It provides according CAN 2.0 specification .

The CAN bus Module as in Figure 3.16 connects the central unit to the I/O units. It is used to transmit the input/output values from the AC 800F to the I/O modules. For this purpose, the AC 800F is equipped with CAN module FI 810F.

All interfaces are electrically isolated and support redundant operation in conjunction with second AC 800F.



FI810 CANbus module

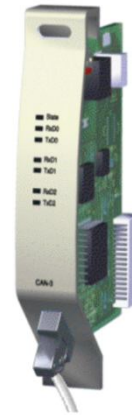


Figure 3.16 CAN module FI 810

- **S800 I/O – AI810**

The analog input module Figure 3.17 has 8 channels each channel can be either a voltage or current input. The current input is able to handle a short circuit to transmitter supply at least 30Vd c without damage current limiting is performed with PTC resistor the input resistance of current input is 250 ohms PTC included. The module distributes the external transmitter power to each channel. All 8 channels are isolated from Module Bus in one group power to the input stages is converted from 24 V on the Module Bus.

By permitting installation in the field, close to sensors and actuators, S800 I/O greatly reduces the installation cost by reducing the cost of cabling. It is possible to exchange modules and reconfigure the system during operation. Redundancy options allow a high degree of availability.

I/O Module -S800 I/O consists of:-

- Digital output module
- Digital input module
- Analog input module
- Analog output module



Figure 3.17 S800 I/O AI810

3.5 Field devices

3.5 .1 Basic Measurement System

A basic instrument/measurement system consists of three elements:

- Transducer/sensor. The transducer is the part of the measurement system that initially converts the controlled variable into another form suitable for the next stage. In most cases, conversion will be from the actual variable into some form of electrical signal, although there is often an intermediate form, such as pneumatic.
- Signal conditioning. In computer process control, signal conditioning is used to adjust the measurement signal to interface properly with the A/D conversion system.
- Transmitter. The transmitter has the function of propagating measurement information from the site of measurement to the control room where the control function is to occur. Usually pneumatic or electronic signals are used.

A simplified block diagram of a basic measurement system is shown in Figure 3.18.



Figure 3.18 Instrument block diagram.

Most modern analog instruments use the following standard signal ranges:

- Electric current of 4 to 20 mA
- Electric voltage of 0 to 10 V
- Pneumatic pressure of 0.2 to 1.0 bars (The bar is a unit of pressure equal to 100 kPa and roughly equal to the atmospheric pressure on earth at sea level.)
- Digital with a built-in binary digital encoder so as to provide a binary digital Output. Having a standard instrument range or using digital signals greatly contributed to the advancement of digital process control and the evolution of modern PLCs.

The following are a few of the primary advantages of such instruments:

- All instruments can be easily calibrated.
- The signal produced is independent of the physical measurement. For example the minimum signal (e.g., temperature, speed, force, pressure, and many other measurements) is represented by 4 mA or 0.2 bars, and the maximum signal is represented by 20 mA or 1.0 bar.
- The same PLC hardware-interface modules are used for all measurements.
- Users can select instruments from a large number of competing vendors; all must comply with universal standards.

3.5.2 Level Measurement

Level sensors detect the level of liquids and other fluids and fluidized solids. The substance to be measured can be inside a container or can be in its natural form. The level measurement can be either continuous or point values. Continuous level sensors measure level within a specified range and determine the exact amount of substance in a certain place, while point-level sensors only indicate whether the substance is above or below the sensing point. Generally the latter detect levels that are excessively high or low.

There are many physical and application variables that affect the selection of the optimal level monitoring method for industrial and commercial processes. The selection criteria include the physical: phase (liquid, solid), temperature, pressure or, dielectric constant of medium, density (specific gravity) of medium, agitation (action), acoustical or electrical noise, vibration, mechanical shock, tank and shape. Also important are the application constraints: price, accuracy, appearance, response rate, ease of calibration or programming, physical size and mounting of the instrument, monitoring or control of continuous or discrete (point) levels. In short, level sensors are one of the very important sensors and play very important role in variety of consumer/ industrial applications. As with other type of sensors, Level Transmitter Figure 3.19 are available or can be designed using variety of sensing principles. Selection of an appropriate type of sensor suiting to the application requirement is very important.



Figure 3.19 Level Transmitter

3.5 .3 Temperature Measurement

A temperature transmitter Figure 3.20 is an electrical instrument that interfaces temperature sensor (e.g. thermocouple , RTD or thermostat) to a measurement or control device .

The thermocouple is the most widely used temperature sensor. Thermocouples operate on the principle that when two dissimilar metals are joined, a predictable DC voltage will be generated that relates to the difference in temperature between the hot junction and the cold junction. The hot junction (measuring junction) is the joined end of a thermocouple that is exposed to the process where the temperature measurement is desired.

The cold junction (reference junction) is the end of a thermocouple that is kept at a constant temperature to provide a reference point. For example, a K-type thermocouple, when heated to a temperature of 300°C at the hot junction, will produce 12.2 mV at the cold junction. Because of their ruggedness and wide temperature range, thermocouples are used in industry to monitor and control oven and furnace temperatures.

The electrical heater are used to maintain the temperature of the oil in the vessel.



Figure 3.20 Temperature Transmitter and Electrical Heater

3.5.4 Pressure Measurement

A pressure sensor measures pressure, typically of gases or liquids. Pressure is an expression of the force required to stop a fluid from expanding, and is usually stated in terms of force per unit area. A pressure sensor usually acts as a transducer; it generates a signal as a function of the pressure imposed. Pressure sensors are used for control and monitoring in thousands of everyday applications. Pressure sensors can also be used to indirectly measure other variables such as fluid/gas flow, water level. Pressure sensors can alternatively be called pressure transducers, pressure transmitters Figure 3.21



Figure 3.21 Pressure Transmitter

3.5 .5 Flow Measurement

Flow can be measured in a variety of ways. Positive-displacement flow meters accumulate a fixed volume of fluid and then count the number of times the volume is filled to measure flow. Other flow measurement methods rely on forces produced by the flowing stream as it overcomes a known constriction, to indirectly calculate flow. Flow may be measured by measuring the velocity of fluid over a known area. Many industrial processes depend on accurate measurement of fluid flow. Although there are several ways to measure fluid flow, the usual approach is to convert the kinetic energy that the fluid has into some other measurable form.

Turbine type flow meters are a popular means of measurement and control of liquid products in industrial, chemical, and petroleum operations.

Turbine flow meters, like windmills, utilize their angular velocity (rotation speed) to indicate the flow velocity. The operation of a turbine flow meter is illustrated in Figure 3.22. Its basic construction consists of a bladed turbine rotor

installed in a flow tube. The bladed rotor rotates on its axis in proportion to the rate of the liquid flow through the tube. A magnetic pickup sensor is positioned as close to the rotor as practical. Fluid passing through the flow tube causes the rotor to rotate, which generates pulses in the pickup coil. The frequency of the pulses is then transmitted to readout electronics and displayed as gallons per minute.



Figure 3.22 Turbine Flow Meter

3.5.6 Analyzer

A water cut Analyzer Figure 3.23 measures the water content (cut) of crude oil and hydrocarbons as they flow through a pipeline. While the title "Water cut" has been traditionally used, the current API naming is OWD or On-Line Water Determination. The API and ISO committees have not yet come out with an international standard for these devices but there are however standards in place for fiscal automatic sampling of crude oil namely API 8.2 and ISO 3171.

Water cut Analyzer are typically used in the petroleum industry to measure the water cut of oil flowing from a well, produced oil from a separator, crude oil transfer in pipelines and in loading tankers. There are several technologies used. The main technologies are dielectric measurements using radio or microwave frequency and NIR measurements and less common are gamma ray based instruments.

The water cut is the ratio of water produced compared to the volume of total liquids produced from an oil well. The water cut in water drive reservoirs can reach very high values.



Figure 3.23 Water Cut Analyzer

3.5.7 Pump

A pump Figure 3.24 is a device that moves fluids (liquids or gases), or sometimes slurries, by mechanical action.

Pumps can be classified into three major groups according to the method they use to move the fluid: direct lift, displacement, and gravity pumps.

Pumps operate by some mechanism (typically reciprocating or rotary), and consume energy to perform mechanical work by moving the fluid. Pumps operate via many energy sources, including manual operation, electricity, engines, or wind power, come in many sizes, from microscopic for use in medical applications to large industrial pumps.

Mechanical pumps may be submerged in the fluid they are pumping or be placed external to the fluid.



Figure 3.24 Pump

3.5 .8 Motor Operated Valve

Motor Operated Valve (MOV) Figure 3.25 are often called as On-Off valves as the motors serve the purpose of fully opening or fully closing valves in pipelines. For example, cooling water lines, process pipelines where controlling of fluid is not required, motor operated valves can be used to fully allow or fully stop the fluid flow. These valves are not used for throttling purposes as they serve mainly On-Off service application.

Motor operated valves can be of various types e.g. Gate/ Ball/ Butterfly etc. with actuator control. Design of Motors and valves can be different. An electric motor is mounted on the valve and geared to the valve stem so that when the motor operates the valve will open or close. For this MOV, motor operated with actuator control from local panel or, from control room is required. There is a requirement of co-ordination among Piping-Electrical-Instrumentation-Process engineers and vendor for design and procurement of such motor operated valves.



Figure 3.25 Motor Operated Valves

3.5.9 Emergency Shutdown Valve

The emergency shutdown function is performed by an emergency shutdown valve. An emergency shutdown valve (also referred to as SDV or Emergency shutdown valve, ESV, ESD, or ESDV) is an actuated valve designed to stop the flow of a hazardous fluid upon the detection of a dangerous event. This provides protection against possible harm to people, equipment or the environment. Shutdown valves form part of a Safety instrumented system.

The process of providing automated safety protection upon the detection of a hazardous event is called Functional Safety. Shutdown valves Figure 3.26 are primarily associated with the petroleum industry although other industries may also require this type of protection system.



Figure 3.26 Emergency shutdown valve

Chapter Four

SOFTWARE

ABB system use two software Control Builder F Engineering Software and DigiVis Operator Station Software.

4.1 Control Builder F

Control Builder F is the engineering tool of Freelance 800F. It is used for configuration and commissioning of all automation functions and displays in a Freelance system. Control Builder F supports the following functions:-

- Configuration and commissioning of user programs.
- Tree view of the programs for convenient program overview and selection, IEC 61131-3 programming.
- Graphical hardware configuration.
- Integral Fieldbus configuration for Profibus and Foundation Fieldbus.
- Project-wide variables and function blocks lists.
- On-line cross reference.
- Efficient plausibility check.
- Extensive Online help and more.

4.2 Digivis

The DigiVis software, based on Microsoft Windows as a graphical user interface, enhances the ease of use and the performance of process operation. In addition, you can also use any PC peripherals such as monitors, printers, and keyboards that are available on the market for Windows-compliant PCs.

The operation and observation (DigiVis) and configuration (Control Builder F) functions can also be performed together on one PC. The DigiVis operator interface offers the following features:-

- Transparent and rapid operation due to a clearly structured information hierarchy
- User-specific function key assignment for fast display selection
- A large number of pre-engineered displays.
- Rapid selection of the correct measuring points in case of process alarms.
- Logging of all operator actions, including name and time stamp.
- System diagnostics, even down to the field device, allowing extended field device error diagnostics.
- Uniform process alarm and message concept and clearly arranged display of messages and operator hints.
- Configurable voice output on the PC for process alarms.
- Dual-monitor operation on a single PC, with one mouse and one keyboard.

4.3 System Configuration

System configuration of Freelance 2000 system consists of the following:-

- Project Manager
- Project Tree
- Hardware Structure
- Commissioning and Documentation

4.3.1 Project Manager

From the Project Manager function you can Start and Edit the project.

Editing of Projects consists of:-

- Creating New Projects
- Open/Close Project
- Importing/Exporting of Project
- Saving/Save as Project
- Setting up Online connection (Configuration to Commissioning)
- Releasing Online connection (Commissioning to Configuration)
- Exiting of Control Builder F

4.3.2 Project Tree

Project Tree provides an Overview of the functions in a project. The Individual elements or objects generally know as Project Objects.

The upper most Object in the project Tree is the configuration CONF, which is the sum total of all the project objects in an AC800F/Freelance system.

The first structural level below CONF is formed by the following Resources:-

- D-PS (Process Station) Resource
- D-OS (Operator Station) Resource
- D-GS (Gateway Station) Resource and
- Operate IT CS (Operate IT Configuration Server)

There is also additional elements namely the Pool of User Defined function blocks and the Global Display.

Figure 4.1 show project tree user interface

menu line

project tree
with nodes

state line

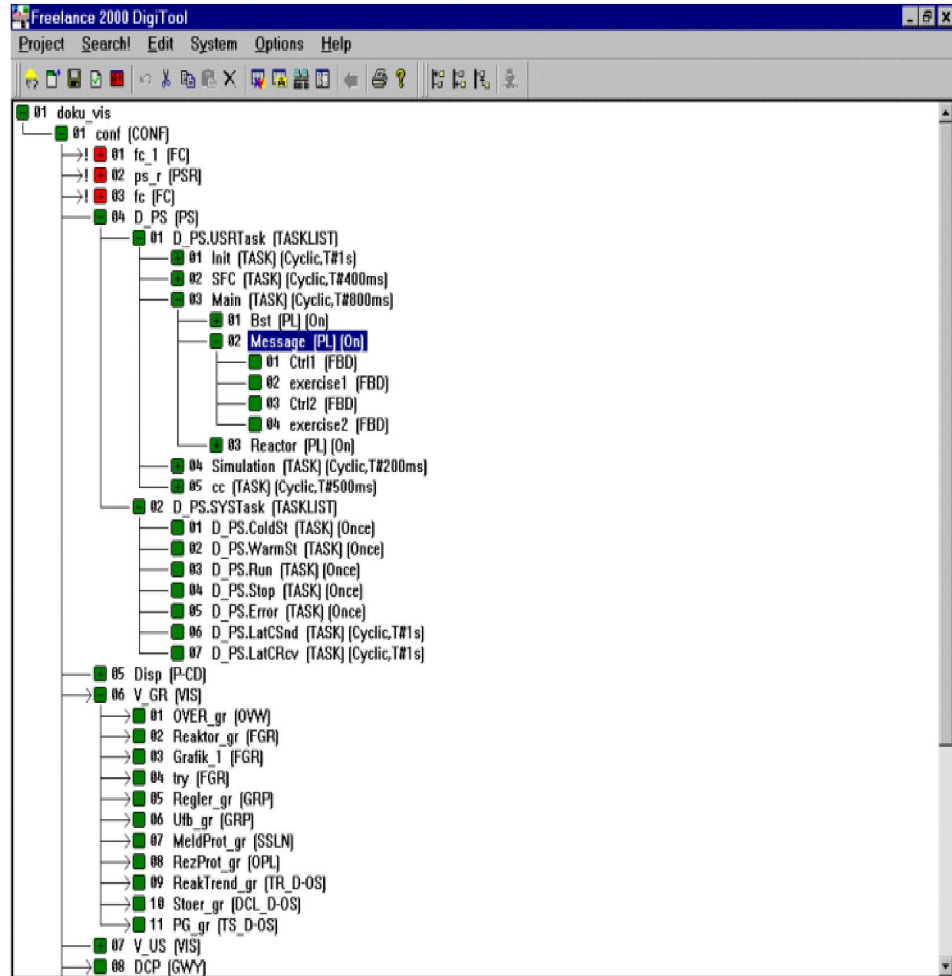


Figure 4.1 project tree user interface

• Project Tree General Process objects

First line **The assigned project name appears here.**

(CONF)	Configuration	The configuration level grouping all resources and permitting data transfer
(D-PS or PS,FC, AC800F)	Process station	The process station contains the CPU module which processes the programs configured under the resource. The type of the process station is defined in the hardware structure by allocating the resource to the hardware object. The short label D-PS indicates that a station has not yet been assigned in hardware manager to a physical station. Following assignation, the assigned station type is displayed: PS for a

rack system, FC or AC800F for a Field Controller.

(D-OS or VIS)	Operator station	An operator station is a resource permitting operation and observation on a PC using the DigiVis program package. The code D-OS indicates that the station has not yet been assigned to a physical station. After it has been assigned, the station type VIS is displayed.
(D-PS/RED or PSR , AC 800 FR)	Redundant Process station	<p>A redundant process station contains redundant CPU modules. The rack system uses two type DCP 10 CPU modules. A redundant Field Controller consists of two AC 800F. The two controllers are connected by means of a redundant link and appear the same as a process station to the user. When the process station is loaded one of the two CPU modules becomes the primary CPU and the other becomes the secondary CPU.</p> <p>The short label D-PS/RED that a station has not yet been assigned in hardware manager to a physical station. Following assignation, the assigned station type is displayed: PSR for a rack system or AC800F for a Field Controller.</p>
Pool		"Memory" of incorrect project objects or those no longer required for processing which you may want to return to the process.

• Resources

The first structural level below configuration is formed by the resources, which represent the various different stations in a project. For the processing of the process itself there are the D-PS (process station) resources, for operation and observation of the process there are the D-OS (operator station) resources and Maestro CS (Maestro NT configuration server), for interfacing to external systems there are the D-GS (gateway station) resources, and for integrating data from external systems there is the OPC server resource.

Data exchange between the process stations is performed by means of cross communication. Each process station is equipped with 10 connections for data

exchange with operator stations and gateway stations. If more stations are configured, then the effective communication links of all the subscribers in a Freelance system are specified in a separate parameter definition dialog.

- **Process station**

Process stations are added with resource types D-PS and D-PS/Red (non-redundant and redundant stations respectively). Correlation to the physical stations is carried out in the hardware manager. Here, process stations (PS) can be selected for rack systems, and Field Controllers (FC and AC800F) can be chosen for connecting Fieldbus modules. The network addresses and resource IDs of the stations are also defined within the hardware manager, and in commissioning mode the configured program modules are loaded from the project tree into the relevant stations.

The execution of user programs within a process station is controlled by tasks. Within a task the sequence of user programs is defined either by program lists or by a structure that is configured using the sequential function chart. While the programs in a program list are run cyclically, sequential function chart programs are run only for specified periods of time according to their structure.

The user programs in the process station, both under a program list and under a sequence flow, are created using the IEC 6113-3 programming languages function block diagram FBD, ladder diagram LD or instruction list IL.

- **Operator station D-OS**

The D-OS resource is provided for operation and observation of the process by an operator station using DigiVis software. Assignment to a physical station is indicated by the code VIS.

Standard operating facilities, e.g. faceplates, are provided for all known tags and variables on the operator stations. In addition, graphic displays, trend displays, SFC displays and logs can be configured and structured for the DigiVis operator stations.

- **Project objects of an operator station D-OS**

(TASKLIST)	Task list	Object for separating the system tasks and the user tasks.
(TASK)	Task	Object which controls the processing of the subordinate program lists and sequential controllers within the resource. A distinction is made between cyclical tasks and those which are processed only once for specific events. In addition, a default task can be configured for each resource. This task is always executed when none of the other tasks is being executed (cyclically or once only).
(TASK/RED)	Redundant task	All subsidiary program lists and sequential function charts within this task are executed redundantly. All tasks can be in redundant format. All the variables in a redundant task must be written through the process image mode.
(PL)	Program list	List of FBD, IL and LD programs which are processed according to the consecutive number in the project tree. Processing of the PLs can be switched ON or OFF.
(FBD)	FBD program	Program which was generated using the Function Block Diagram language (FBD).
(LD)	LD program	Program which was generated using the Ladder Diagram (LD) Language
(IL)	IL program	Program which was generated using the Instruction List (IL) language.
(FGR)	Graphic display	Display of freely grouped static and dynamic display objects generated by the graphics editor.

- **Gateway station**

The gateway stations D-GS are used to make data from the Freelance system available to other systems. In principle, all the data from the Freelance system can be read and written via a gateway station. In addition to each gateway station in a Freelance system, the relevant server software from the add-on packages DigiDDE, DigiOPC or DigiCSO must also be installed on the network. (If, for example, a gateway station of type OPC gateway is configured in a Freelance system, then the Freelance OPC server software must be installed on a PC that is linked on the network with the Freelance process stations).

- **Task TASK and redundant task TASK/RED**

The task object comprises all the subordinate program lists and SFC programs. The tasks determine how quickly the programs are processed on the resource. A distinction is made between user and system tasks.

In user tasks, the programs are processed cyclically; in system tasks, certain events in the resource or commissioner actions determine whether a certain system task is executed. The programs within a task are controlled by a program list or by an SFC program.

The number of user tasks is limited to 9 tasks per Process Station/Field Controller (Field Controller default setting: 3 tasks) . A maximum of 8 cyclic tasks and one default task can be configured.

The default task is always executed if none of the other tasks is executed (either cyclically or once only).

- **Predefined System Tasks**

All system tasks are grouped in the project tree under *SYSTask node. The predefined Tasks let you start and stop processing sections of a user program as a reaction to system state changes. The event which triggers the task, e.g. a certain state transition of the resource, or an error in the user program, is always defined. All predefined system tasks are created in the resource after initial program loading. After the resource is initialized, the system tasks are automatically started, but do not compute until the event assigned to them occurs.

The following tasks are predefined:

- * ColdSt [Task] [Once]
- * WarmSt [Task] [Once]
- * Run [Task] [Once]
- * Stop [Task] [Once]
- * Error [Task] [Once]
- * LatCSnd[Task] [Cyclic,T#1s]
- * LatCRcv[Task] [Cyclic,T#1s]
- * RedSt [Task/Red][Once] (only for redundant process stations)

- **Program List**

The Program List object contains all subordinate programs and controls the processing sequence of these programs. The programs are processed depending on their consecutive number in the project tree. The higher-order task determines how fast the programs are processed on the CPU module of the process station. The programs are written in the Function Block Diagram (FBD), the Ladder Diagram (LD) or Instruction List (IL) language.

4.3.3 Hardware Structure

Within the hardware structure the resources defined in the project tree are allocated to the hardware actually required. A system essentially consists of the process, operator and Gateway stations. These stations are allocated to resources according to IEC 61131-3. The resources serve as structuring elements in the project tree for allocation of the application program parts and displays to the hardware actually required.

The D-PS resources which are configured in the project tree are assigned to the process stations in the hardware structure. A process station can have a rack-based set-up, that is, conventional with input and output modules assembled in module assembly frames for process control, or by the Field Controller through one or more field buses with intelligent field devices and/or remote I/O.

In the conventional process station, the I/O modules are mounted in modules supports (Racks). The CPU module processes all programs of this resource and is simultaneously the communication module to the system and the I/O modules. The Process station consists of the central unit and 4 I/O units max. The central unit is always the station with the CPU module.

The I/O units do not need an own CPU module: they can accept other I/O modules so that their numbers can be expanded (from 8 for the central unit) to a maximum of 44 for a process station. All units need a link module for feeding in the power supply and an identical rack for holding the modules.

Depending on the Field Controller, the process station is now provided in two versions: the FC and the redundant Industrial IT Controller AC 800F. The Field Controller takes the fieldbus modules and makes it possible to connect various fieldbusses.

The Field Controller basic unit consists of the case and the main board, which together form a unit which can be equipped with various modules. The module for the power supply and an Ethernet module for connection to the system bus are absolutely necessary. Both modules are available in various designs. A Field Controller can be equipped with a maximum of 4 fieldbus modules selected from CAN, Profibus and serial modules.

The CAN module allows the connection of a maximum of 5 I/O units and thus the connection of 45 I/O modules in the way in which they are also used in the conventional process station.

Each Profibus module allows the connection of a Profibus line, i.e. the connection of a maximum of 125 slaves. Each of these slaves can also be modular, i.e. contain a maximum of 64 modules.

The serial module has 2 interfaces which can be occupied at option with the Modbus master interface protocol, the Modbus slave interface protocol, the telecontrol interface protocol, the Protronic interface protocol or the Sartorius scale interface protocol.

The number of process and operator stations is not limited. The information given in the boot parameters of the process station determines how many operator stations or gateways will be supported by the process station.

The Operator stations are commercial PCs in which the software program DigiVis has been installed for process visualization. All displays and logs are configured with the DigiTool software program and loaded into the operator station.

D-ES resource stands for an Engineering station, i.e. the configuration tool DigiTool. It is also displayed in the hardware structure; this display is only for documentation purposes.

The hardware can be configured in two different view areas. In the tree view area the complete system can be configured in a tree structure.

In the graphic view area the hardware can be configured in several display levels:

The system view displays the complete hardware structure. In the system view the individual stations like the process station, Field Controller or operator station are activated and allocated to the project tree resources.

The station view displays the complement of a process station or a Field Controller. In the station view the modules or Field Controller modules are allocated to slots.

The detail view displays further information on the modules of a process station or the modules of the Field Controller so that they can be identified more easily. The detail view of the Profibus master displays a bus line with the slaves configured on the bus. The slaves are represented using bitmaps which are specified in the device database file. If no bitmap files are specified in the device database file of the respective vendor, standard bitmaps are displayed.

However, it is possible subsequently to assign vendor-specific bitmaps. The detail view of the Profibus slave displays a device view in the form of an individual bitmap file which can be assigned in the parameter dialog of the slave.

The variables which are to be linked with the process via the I/O modules are entered in the I/O editor. A number of I/O components is provided by each module or slave according to the number of channels. Via these I/O components it is possible to directly use the information in programs and graphic displays.

If the I/O component is also to appear in the variable list, a variable name must be indicated. The I/O component name is composed of the object name (16 characters) and the component name (16 characters), thus making available a total of 32 characters.

Diagnosis components are available in addition to the I/O components. Through the network configuration, communication addresses, so-called resource Ids (previously station numbers), are assigned to the resources. IP addresses of the units assigned to the resources are also indicated in the network setting. When the project objects are loaded into the stations, the corresponding application program parts (as allocated in the project tree of the individual resources) are loaded into the resources. The resources or individual parts of the application program are loaded with DigiTool commissioning.

- **Hardware Structure User Interface**
- **Tree View**

The hardware structure tree view displays all hardware structure objects. Beginning with the system object, other hierarchical levels can be entered down to the device level.

The stations are entered via a position number showing the respective station in the graphic view display. Position query is made every time a station or a gateway is inserted.

Each object features object parameters such as name, short text and long text. The parameters relating to the objects are described in the Engineering manuals of the process stations. See Engineering Manual, Process station.

Individual object names are allocated automatically so the user does not need to allocate names. Names can be subsequently configured.

- **Graphic View**

The hardware structure graphic view features several views with different information within their detailed levels.

- **System View :** Figure 4.2 Rapid overview over the global structure, e.g. number of operator and process stations, number of FieldControllers, number of gateways.

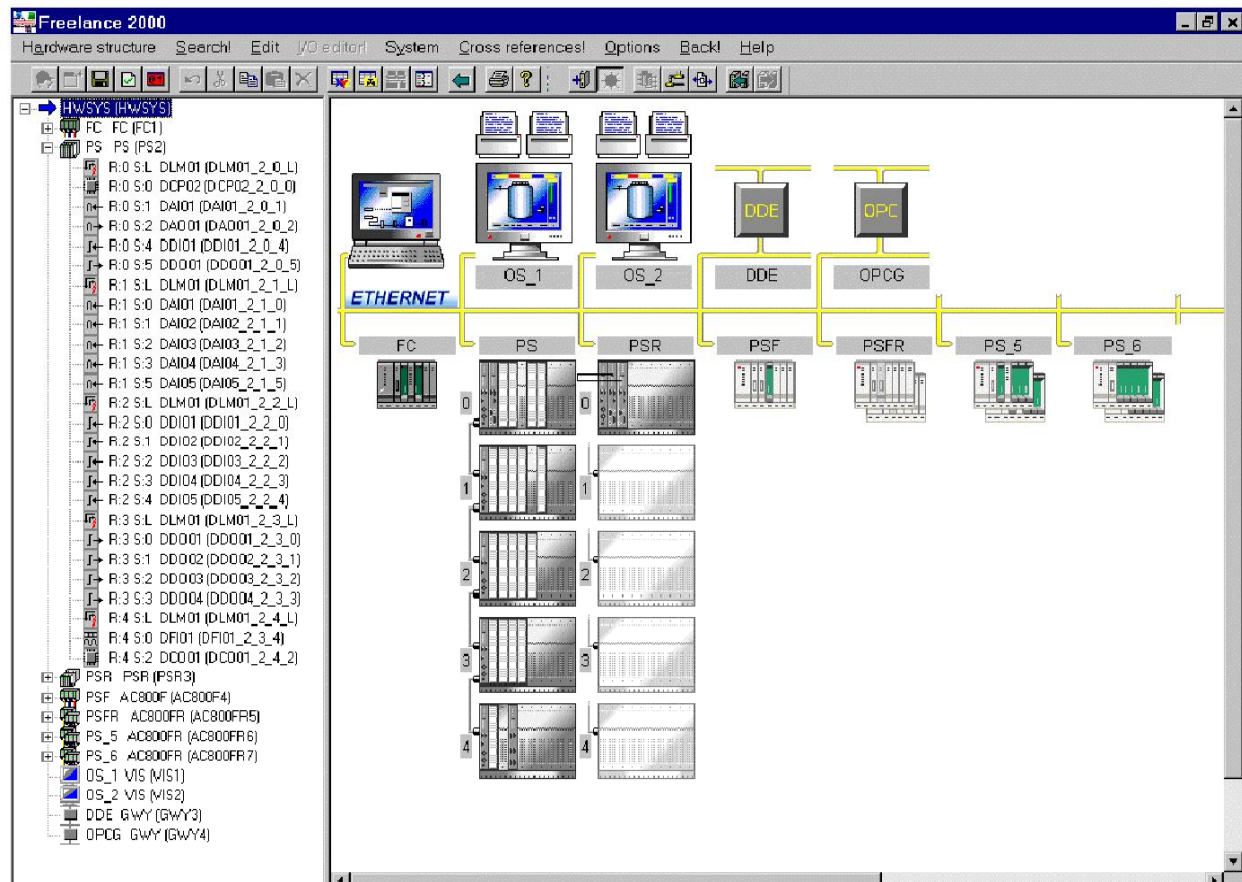


Figure 4.2 System view

- **Station View:** Figure 4.3 Immediate information concerning equipped/free slots and plugged-in module types. Intuitive equipment by simple double click.

The station view of the process station shows the central unit with the CPU module and the I/O units.

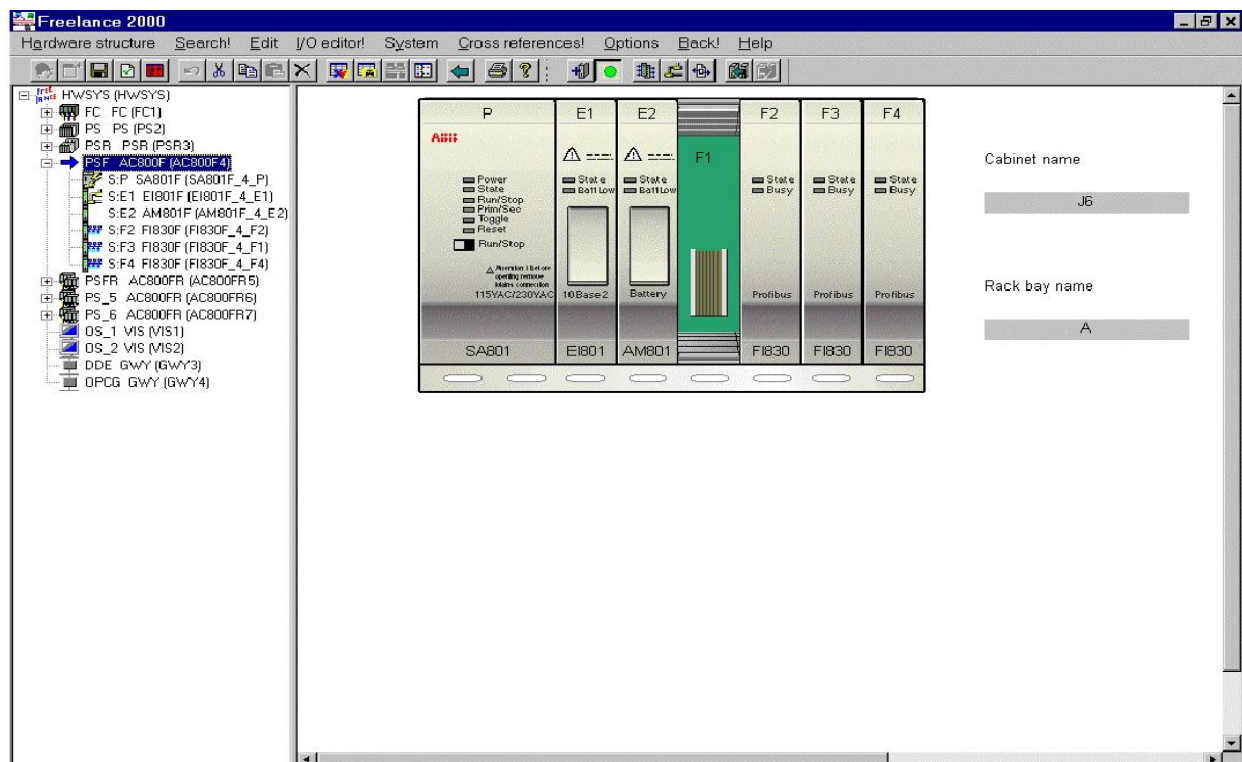


Figure 4.3 Station view

- Detail View:** The detail view differs depending on the object selected and displays a detailed display of the object. Displayed are the module type (status information only online), the slaves connected, their vendor, model name and bus address (status information and diagnostic information only online) or only the slave connected in its device view with some information such as vendor, model name and bus address (status information and diagnostic information only online).

4.3.4 Commissioning

Commissioning is an operating mode of DigiTool which offers a range of other functions in addition to the loading of project objects. However, in contrast to the configuration operating mode, the user programs cannot be changed structurally. Only when the user programs have been configured and checked for plausibility, can they be loaded into the process station or the Field Controller and started as part of the commissioning process.

When a project is commissioned for the first time, the complete project must be loaded into the respective station. Later, only changes need to be loaded. Changed project objects are loaded and started in accordance with selections made in the project tree. The selection of single or multiple user program objects is made in the project tree, as already described under configuration.

As well as starting, loading, stopping or initializing project objects, such as a resource, a task or a program list, it is also possible to edit individual function blocks of programs. This enables the operator to change operating modes, switch to a specified operating state and set parameters for function blocks without having to reload the modified program. There are two methods of making these changes known to the system:

- Write loads the changes without saving them in the project file.
- Correct saves the changes in the project file, as well as loading them.

After loading a project with Load whole station, additional configuration changes may be loaded incrementally. A configuration change consists of the creation of, deletion of or a change to a project object. A configuration change which effects other project elements is said to have side effects.

During loading, the existence of side effects will cause the changed project object and any objects affected by it to be stopped. For this reason, changes with side effects should be loaded during operations only with the greatest of caution. The user is made aware of the occurrence of side effects by the way the object nodes concerned are displayed in the project tree.

In FBD and LD programs, binary values are displayed directly with their logical state of 1 or 0.

The state of the binary signal is recognized by a different line type. In the IL programs, the current contents of the accumulator are displayed in their own column.

Freelance 800F provides an operator level and a process level. The operator level contains the functions for operation and observation, archives and logs, and alarms. Open-loop and closed-loop control functions are processed in the controllers which communicate with actors and sensors in the field.

The whole system Figure 4.4 consists of the following components:

- Control Builder F – Engineering Tool
- DigiVis – Operator software
- AC 800F Controllers
- Configuring and commissioning hardware and software
- Fieldbus and device management
- Cross-references
- Graphics editor for DigiVis

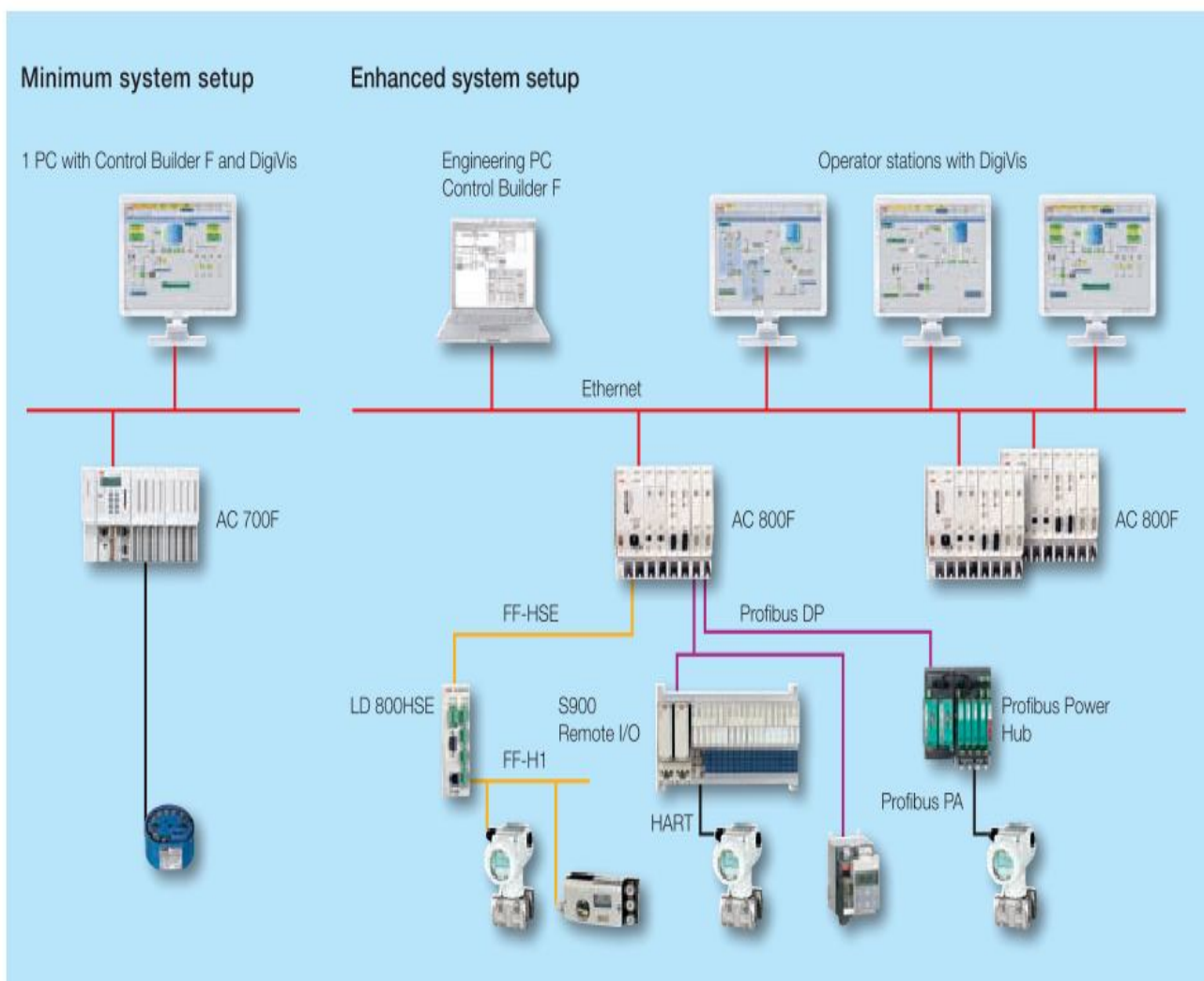


Figure 4.4 Freelance 800F system

4.4 Process Description

4.4 .1 Level Monitoring

Monitoring the level of crude oil can be accomplished through the use of a level transmitter by turning a pump ON or OFF we can control the level.

Figure 4.5 shows a block diagram of level transmitter that can be used to implement monitoring the crude oil level in the vessel.

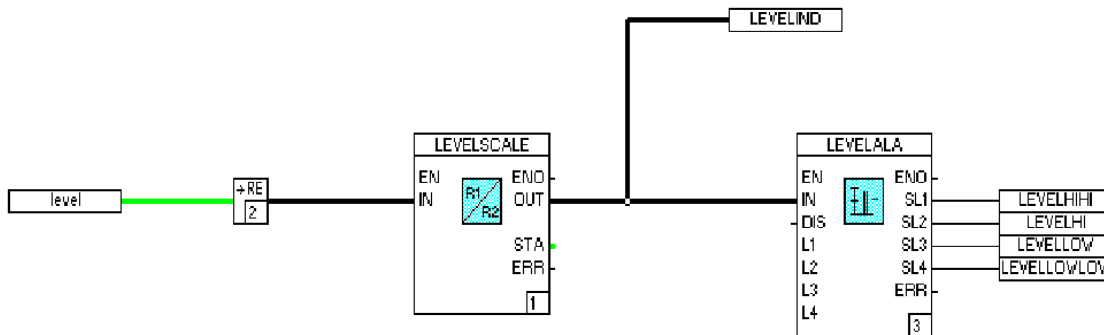


Figure 4.5 Block Diagram of Level Transmitter

The mode operation of pump can be programmed as follows:

In manual Mode the pump will start if the crude oil in the vessel is at any level except low.

In automatic Mode if the level of oil in the vessel reaches a high point, the pump will start Automatically so that crude oil can be removed from the vessel, thus lowering the level.

When the crude oil level reaches a low point, the pump will stop .

On the discharge side of the pump pressure is monitored by pressure indicator.

to ensure pumps are operating within the normal process operating range, and also to ensure they are not operated outside their allowable envelope.

The pressure measuring device will initiate ; high-high pressure alarm and shut down of pumps.

High pressure set points should be provided to initiate an alarm condition; high-high pressure set points will initiate pump shut down.

Figure 4.6 shows a block diagram of pump

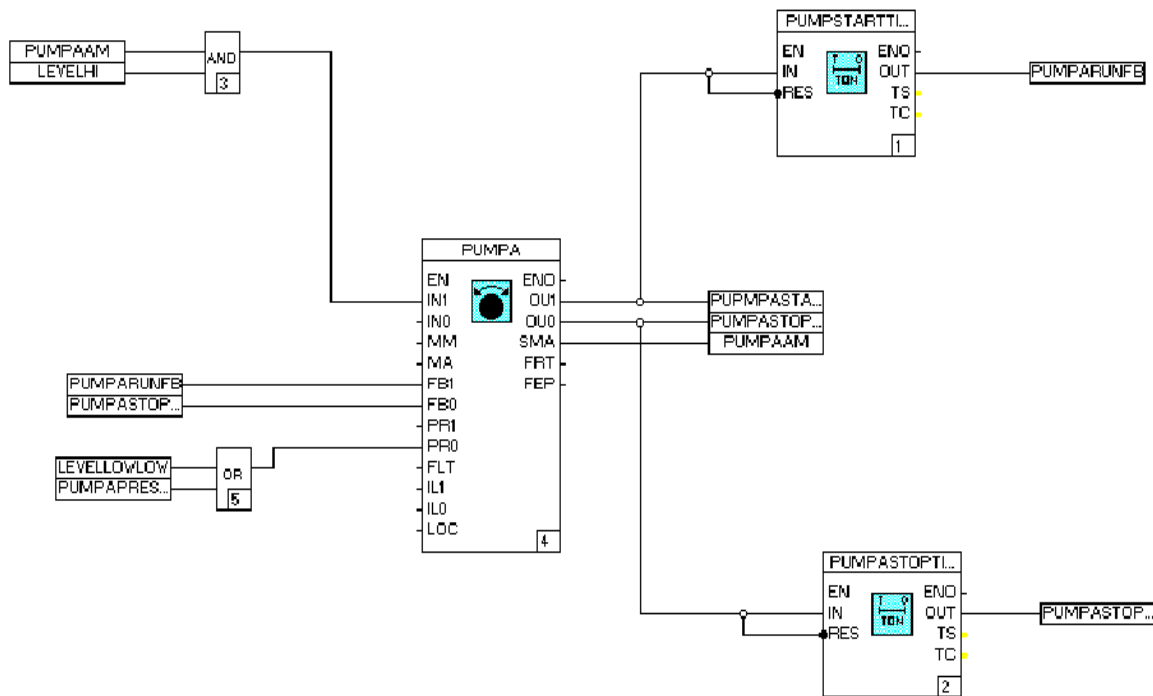


Figure 4.6 Block Diagram of Pump

4.4 .2 Temperature Monitoring

The Temperature transmitter used to monitor the current temperature of the crude oil. Figure 4.7 shows block diagram of Temperature Transmitter that can be used to implement monitoring the crude oil temperature in the vessel.

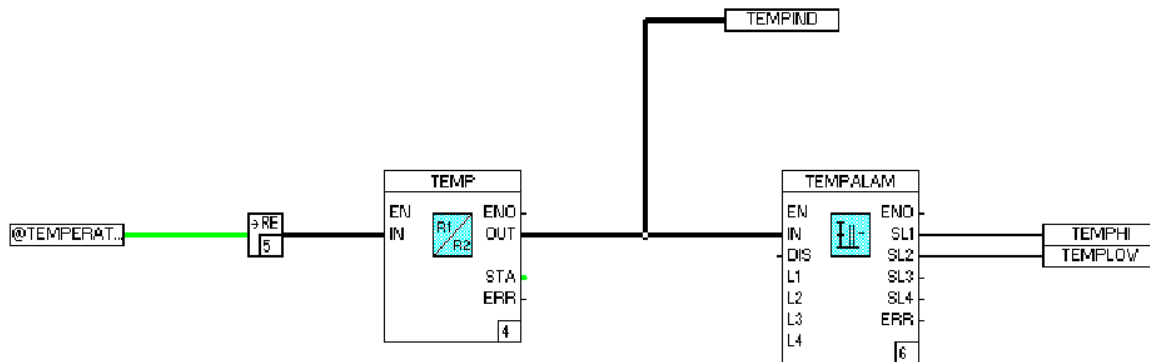


Figure 4.7 Block Diagram of Temperature Transmitter

The electrical heater are used to maintain the temperature of the oil in the vessel. The heaters will auto cut in when the temperature drops below the set point. .Figure 4.8 shows block diagram of the heater can be used to used to maintain the temperature of the crude oil temperature in the vessel.

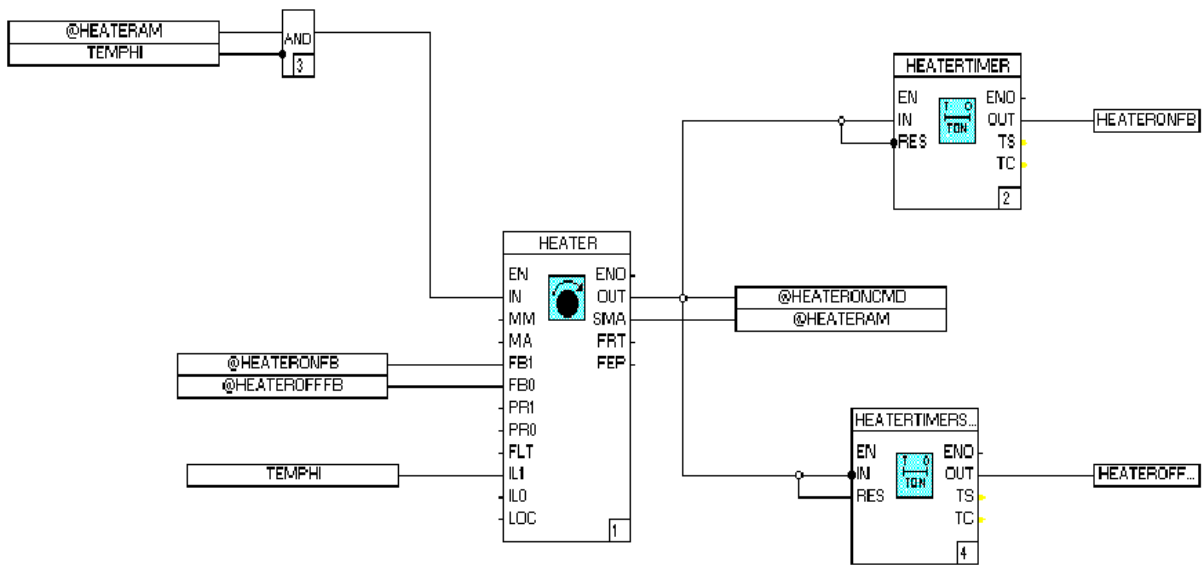


Figure 4.8 Block Diagram of Heater

4.4 .3 Pressure Monitoring

The pressure transmitter used to monitor the pressure in the vessel.

Figure 4.9 shows a block diagram of pressure transmitter that can be used to implement monitoring the crude oil level in the vessel.

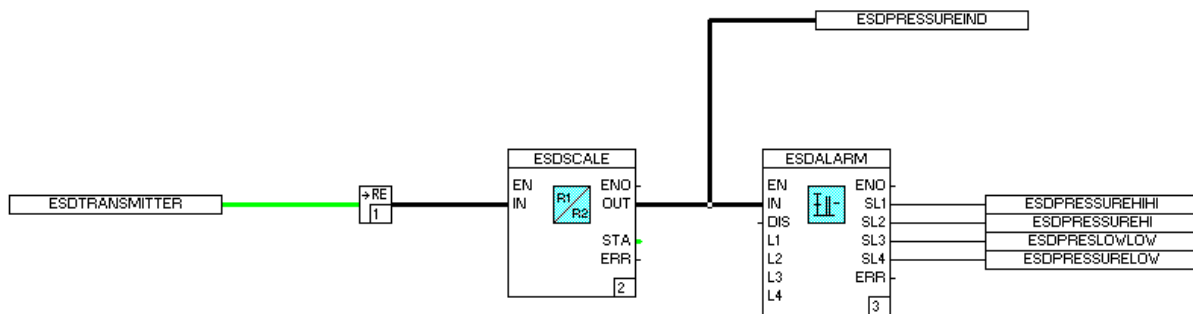


Figure 4.9 Block Diagram of ESD Pressure Transmitter

when the pressure in the vessel reach high-high the emergency shutdown valve (ESD) automatically close Figure 4.10 shows a block diagram of (ESD) Valve.

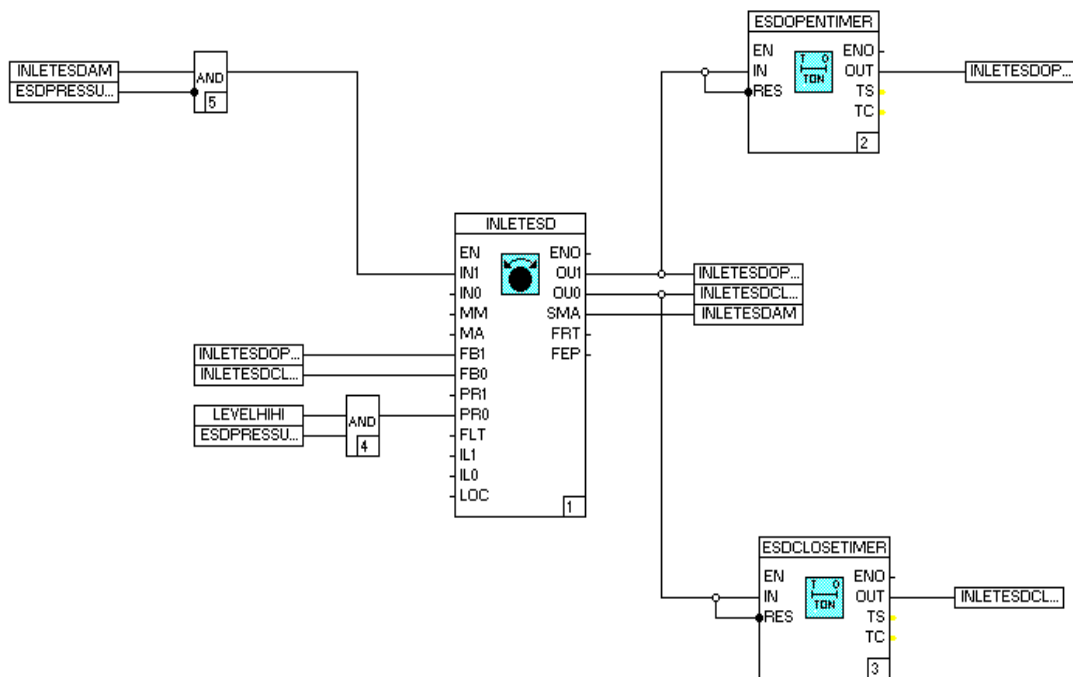


Figure 4.10 Block Diagram of Emergency Shutdown Valve

4.4 .4 Flow Monitoring

Flow monitoring provided on the discharge side of the pump. Flow monitoring should indicate flow rate, and accumulated flow volume.

Figure 4.11 shows a block diagram of flow transmitter that can be used to implement monitoring the crude oil flow on the discharge side of the pump.

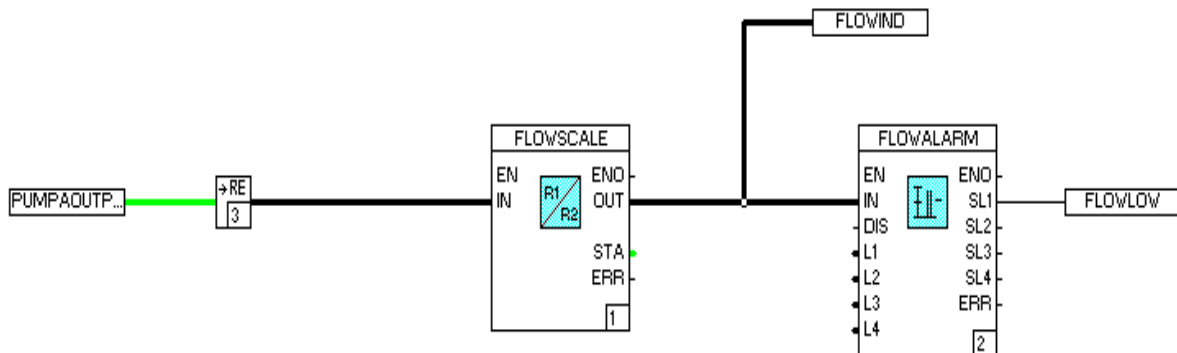


Figure 4.11 Block Diagram of Flow Transmitter

4.4 .4 Water Cut Monitoring

Analyzer is used to monitor the water cut in the crude oil to ensure the crude meet the specifications . Figure 4.12 shows a block diagram of analyzer .

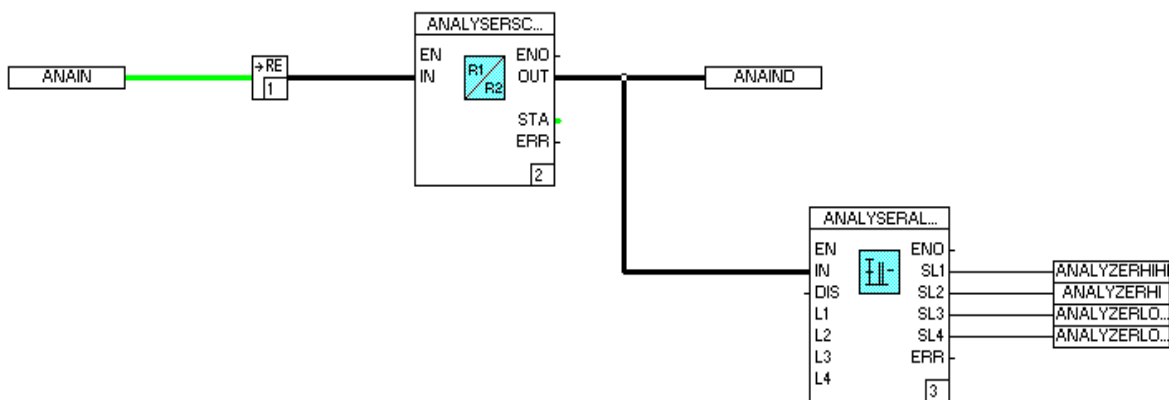


Figure 4.12 Block Diagram of Analyzer

When the water cut reach high-high the motor operated valve (MOV) automatically open. Figure 4.13 shows a block diagram of motor operated valve .

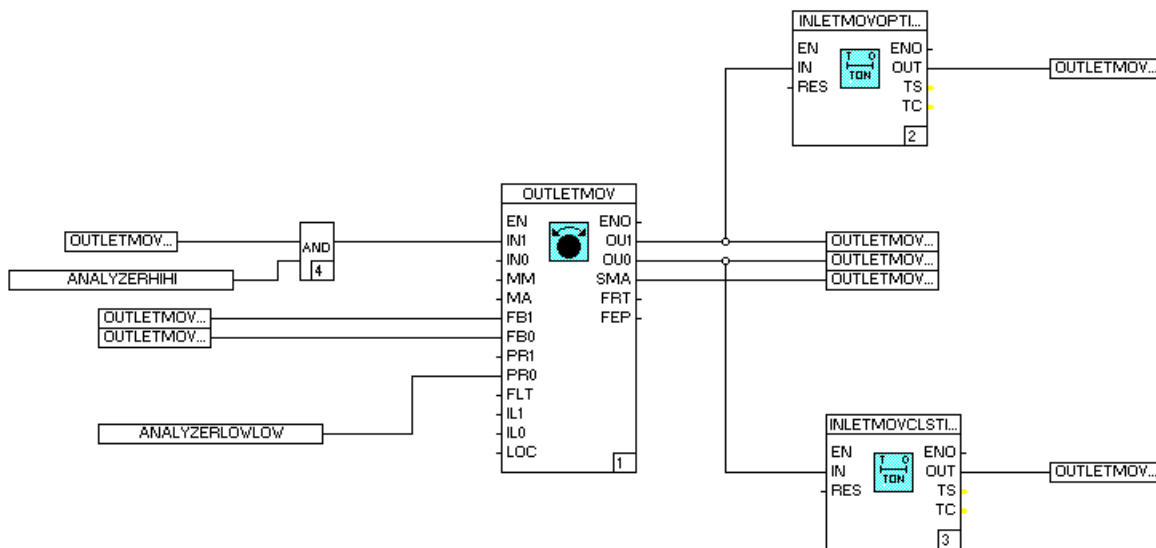


Figure 4.13 Block Diagram of Motor Operated Valve

4.5 Simulation

System checkout and troubleshooting are critical tasks before the final deployment and commissioning of any process-control/automation system. These tasks must be performed according to well-established rules and documented procedures.

These rules/procedures are designed carefully to prevent any potential problems that might cause personnel injuries or damage to resources or compromise any of the predefined process/end-product qualities. Simulation and emulation techniques are used throughout the implementation phases of the project prior to the final checkout. Simulation tools allow designers to examine system-performance

situations under varying scenarios before or aside from the actual implementation. These steps also can be used to efficiently verify design concepts prior to the commitment of resources. Emulation phases are accommodated in PLC software-development tools that allow the user to emulate I/O hardware, HMIs, control logic, and communication facilities. The initial phase of emulation checks only the control logic, whereas the second phase includes actual and simulated hardware/interfaces. The final emulation phase uses actual system hardware, interfaces, and communication resources.

Simulation of the program Figure 4.14 using the Freelance 800F simulator . the program is running according to the process description.

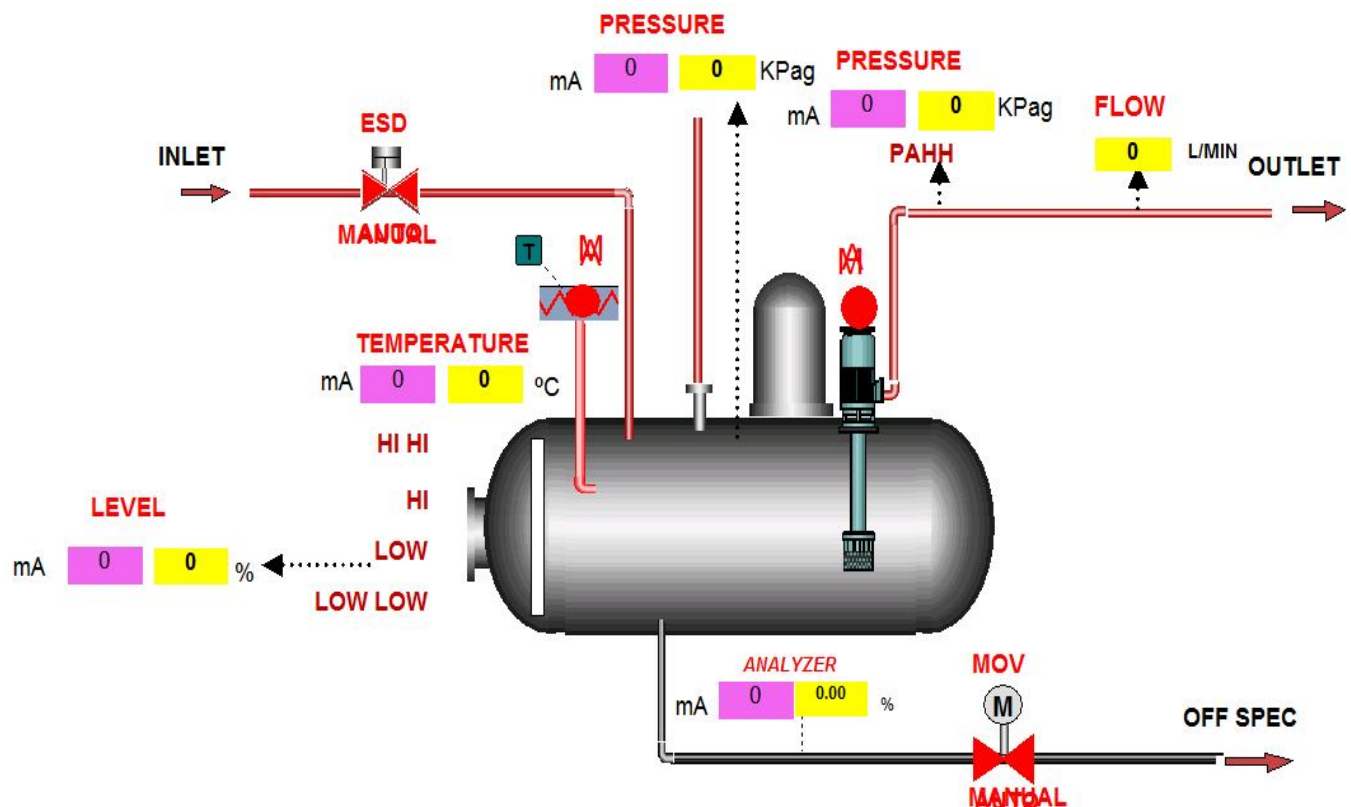


Figure 4.14 Monitoring Process in Vessel by Using Freelance 800F simulator

Chapter Five

RESULT&DISCUSSION

5.1 Result

The various results obtained in the research are presented here. Using PLC various sensors are connected in the developed system. The following tables indicates the number of inputs and outputs .

Table 5.1 Inputs

Input	Data type
level	INTEGER
TEMPERATURE	INTEGER
PUMPAOUTPRESS	INTEGER
ANAIN	INTEGER
ESDTRANSMITTER	INTEGER
INLETESDAM	BOOLEAN
HEATERAM	BOOLEAN
PUMPAAM	BOOLEAN
OUTLETMOVAM	BOOLEAN

Table 5.2 Outputs

Output	Data Type
LEVELIND	BOOLEAN
TEMPIND	BOOLEAN
PUMPAPREIND	BOOLEAN
FLOWIND	BOOLEAN
ANAIND	BOOLEAN
ESDPRESSUREIND	BOOLEAN
OUTLETMOVOPENCMD	BOOLEAN
OUTLETMOVCLSECMD	BOOLEAN
OUTLETMOVOPENFB	BOOLEAN
OUTLETMOVCLOSEFB	BOOLEAN
INLETESDOPENCMD	BOOLEAN
INLETESDCLOSCMD	BOOLEAN

INLETESDOPENFB	BOOLEAN
INLETESDCLOSEFB	BOOLEAN
PUMPASTARTCMD	BOOLEAN
PUMPASTOPCMD	BOOLEAN
PUMPARUNFB	BOOLEAN
PUMPASTOPFB	BOOLEAN
HEATERONCMD	BOOLEAN
HEATERONFB	BOOLEAN
HEATEROFFFB	BOOLEAN

Table 5.3 Output Alarming

Output	Data Type
LEVELHIHI	BOOLEAN
LEVELHI	BOOLEAN
LEVELLOW	BOOLEAN
LEVELLOWLOW	BOOLEAN
TEMPHI	BOOLEAN
FLOWLOW	BOOLEAN
TEMPLOW	BOOLEAN
PUMPAPRESHIHI	BOOLEAN
PUMPAPRESHI	BOOLEAN
ESDPRESSUREHIHI	BOOLEAN
ANALYZERHIHI	BOOLEAN
ANALYZERHI	BOOLEAN
ANALYZERLOW	BOOLEAN
ANALYZERLOWLOW	BOOLEAN

5.2 Discussion

All inputs and outputs associated with the process-control system must be listed in the I/O table along with the associated instrumentation.

On commissioning the FBD language the program is displayed in the same way as in configuration except that in commissioning mode the program cannot be modified structurally.

Individual function blocks can be selected and parameters set for them. Operating modes can also be called up and modified from commissioning mode. Thereafter, certain program test functions are available to whoever is commissioning the system.

Integer value in table 5.1 are initially displayed directly with their Integer number

Boolean values (binary values) are initially displayed directly with their logical state of 1 or 0.

logical 1 true

logical 0 false

Monitoring and control are the core of the PLCs function. PLC can be connected through adequate network and real-time communication facilities.

.Most PLC system faults occur in the field wiring and devices. The wiring between the field devices and the terminals of the I/O modules is a likely place for problems to occur. Faulty wiring and mechanical connection problems can interrupt or short the signals sent to and from the I/O modules.

The sensors and actuators connected to the I/O of the process can also fail. Mechanical switches can wear out or be damaged during normal operation. Motors, heaters, lights, and sensors can also fail. Input and output field devices must be compatible with the I/O module to ensure proper operation.

When a problem occurs, the best way to proceed is to try to logically identify the devices or connections that could be causing the problem rather than arbitrarily checking every connection, switch, motor, sensor, I/O module, and so on. First, observe the system in operation and try to describe the problem. Using these observations and the description of the control system, you should identify the possible sources of trouble. Compare the logic status of the hardwired inputs and outputs to their actual state. Any disagreements indicate malfunctions as well as their approximate location.

Some of your troubleshooting can be accomplished by interpreting the status indicators on the I/O modules. The key is to know whether the status indicators are telling you that there is a fault or that the system is normal. Often PLC manufacturers supply a troubleshooting guide, map, or tree that presents a list of observed problems and their possible sources.

In the case of a manual system, all equipment is started and stopped by the operator, all process operations are controlled by the operator. This requires that the plant be manned continuously while in operation, perhaps with more than one operator.

The type of control system should be identified with its specific components, if decided; otherwise, general requirements of the system must be given. Include any special requirements such as interfaces to vendor or customer systems, operator panels, field stations, control-room displays, and communication with other control systems. Define the basic system architecture, identifying and explaining any deviations from standard models.

The following is a sample of typical control-system components:

- Complete listing of control-system inputs and associated sensors
- Complete listing of control-system outputs and associated final control elements/actuators

- Basic system control structure and alarming parts
- Safety control-system requirements
- Process information system and data acquisition
- Procedural control recipes, coordination, and sequencing
- Communication and networking interfaces

In the case of an automatic system, all equipment is started and stopped by the control system adjusted automatically to maintain the system levels, discharge pressures, etc. This may allow unattended plant operation or operation with a single operator, but requires a more complex and expensive control system, with associated maintenance.

All alarms must be latched until the Operator has acknowledged them. If the alarm is indicated by a lamp, it must flash until acknowledged then remain steady until the alarm clears. If it is indicated on a computer screen, an appropriate color code or symbol must be used to indicate for each alarm whether it has been acknowledged. Automated systems should log the time at which the alarm occurred, the time it was acknowledged and the time it cleared. Logs may be printed on paper or recorded electronically.

Valve and equipment status should use a consistent method of symbols and colors, whether the status is indicated through lamps or on a color computer screen. The color-coding scheme should be consistent with any existing equipment displays elsewhere in the plant.

The range of the transmitter should be chosen so that it will normally operate at one-half to two-thirds of scale at normal design pressure; the transmitter should not be operated full-time near the top end of the scale. This will provide some safety margin on over-pressure as well as prolonging the life of the sense element in the transmitter.

Monitoring of the system discharge pressure can be useful in identifying possible problems in the discharge piping or force main and in monitoring pump performance. The pressure-metering device should be connected to either the control system or to a recording device or both.

The design should have provision for local control systems where parts of the plant may be operated or controlled from a remote location. The local control stations should include provision for preventing the operation of equipment from remote locations.

The flow metering element should be selected carefully to ensure that there are no obstructions. Provision should be made so that the flow-metering element can be bypassed or isolated for routine maintenance activities. The flow-metering device should be connected to either the control system or to a recording and totalizing device or both. This provides for a record of flows out of the lift station. It can also be used to help identify possible problems in the discharge piping or force main.

When making decisions relating to instrumentation and control, the following factors should be considered:

1. Plant size and complexity
2. Regulatory requirements
3. Hours of attended operation
5. Primary element reliability
6. Primary element location
7. Whether controls should be manual or automatic.
8. The data storage and recording requirements and whether data acquisition should be central or distributed .

Chapter Six

CONCLUSION&RECOMMENDATIONS

6.1 CONCLUSION

Programmable logic controllers (PLCs) have made it possible to precisely control large process machines and driven equipment with less physical wiring and lower installation costs than is required.

Monitoring and control systems can be a conventional system with recorders, indicators, switches, push buttons, indicating lights, control panels, etc. or it can be a computerized control system that utilizes various configurations of hardware and software to provide the control required. Computerized systems can be separated into two groups, PLC (Programmable Logic Controller) Systems and Distributed Control Systems.

The PLC based system is a multipurpose system with extensive scope for modification. The plant status, alarms, motor starters, meters and analyzers are all wired into input/output cards located in what are called racks. The racks may be mounted separately or placed in specific plant areas to reduce wiring costs. The input/output racks are associated with controllers that are programmed to perform the required process control functions. Changes can generally be made relatively easily by modification of or addition to the PLC controller programs.

PLC control systems have been designed to be easily installed and maintained. Troubleshooting is simplified by the use of fault indicators and messaging displayed on the programmer screen. Input/output modules for connecting the field devices are easily connected and replaced.

Several factors should be considered when developing a plan for the instrumentation and controls. A monitoring requirements vary depending on the type of facility being considered and its location; this will impact on the selection and type of instrumentation being considered. Instrumentation and control requirements will also depend on the size of the plant, and as each process has its own set of conditions to be monitored and controlled there will be different technical requirements to be met. In general, instrumentation and control should provide efficient and safe automatic and manual operation of all plant systems with a minimum of operator effort. Automatic systems should also be provided with manual back-up systems.

6.2 RECOMMENDATIONS

Programmable logic controllers (PLCs) are the result of evolution of hardwired analog control systems.

Automatic systems should use either Programmable Logic Controllers (PLC's) or a Distributed Control System (DCS). The operator interface may be in the form of traditional control panels (i.e.: lights, gauges and switches), electronic control panels (with text and/or graphics) and computers.

Digital communication between components of the control system must be reliable and self-monitoring. The communication protocol must meet the following requirements:

1. It must include error checking and reporting, to ensure that data is correctly transferred from one component to another;
2. The components of the system must detect the failure of the communication system (either between individual components of the system or between the system and the operator); and

3. It must be compatible with a variety of manufacturer's instruments and equipment, in order to allow for expansion of the system.

The operator interface may consist of a local hard wired control panel or mimic, character based input/output panel, personal computer or workstation depending on system size, process complexity, control system functions and operator interface manufacturer. Where personal computers or workstations are used, select the hardware based on reliability, software compatibility, vendor support and suitability for continuous operation in the plant environment. The operator interface software may provide the operator with interactive control and monitoring of the plant, handle and annunciate alarms, log and trend events and process variables and generate the required reports. Process control and logic should be performed by PLC and not the operator interface computer or workstation.

Complete design documents should be prepared to ensure that construction can be completed correctly and also to properly record the system for future reference. The following are required in the design documents:

1. Design and construction standards, specifications and installation details
2. Panel sizing and general arrangement
3. Control system functional requirements
4. Control component and instrument data sheets
5. Operator interface and control hardware and software specifications including input and output (I/O) lists
6. Control system programming and packaged system configuration standards, structure and scope.

REFERENCES

- [1] Khaled kamel, Eman Kamel, "Programmable Logic Controllers Industrial Control" Cenveo® Publisher Services Version 1.0 (2013).
- [2] ABB Automation "Engineering Manual Process Station" - AC 800F Version 7".
- [3] Frank D. Petruzella "Programmable Logic Controllers", McGraw-Hill 4th editions (2011).
- [4] Terry R. Borden, Richard A. Cox. " Technician's Guide to Programmable Controllers", Delmar, Cengage Learning 6th editions (2012).
- [5] L.A. Bryan, E.A. Bryan" Programmable controllers: theory and implementation " , Industrial Text Company 2nd editions (1997).
- [6] W. Bolton "Programmable Logic Controller" , Newnes is an imprint of Elsevier 5th Edition (2009).
- [7] Hugh Jack "Automated manufacturing system with PLC" version 5.1(2008)
- [8] E.A. Parr "Programmable Controllers An engineer's guide", Newnes is an imprint of Elsevier 3th Edition (2003).
- [9] Béla G. Lipták, " Instrument engineers' handbook : process software and digital networks" , CRC PRESS 3th editions(2002).
- [10] Frank Petruzella. Electric motors and control systems, McGraw-Hill (2010)
- [11] W. Bolton "Instrumentation and Control System ", Elsevier Science & Technology Books August (2004).
- [12] Dominique Placko "Fundamentals of Instrumentation and Measurement",Hermès Science (2007).
- [13] Robert B. Northrop "Introduction to Instrumentation and Measurement

Taylor & Francis Group 2th Edition (2005).

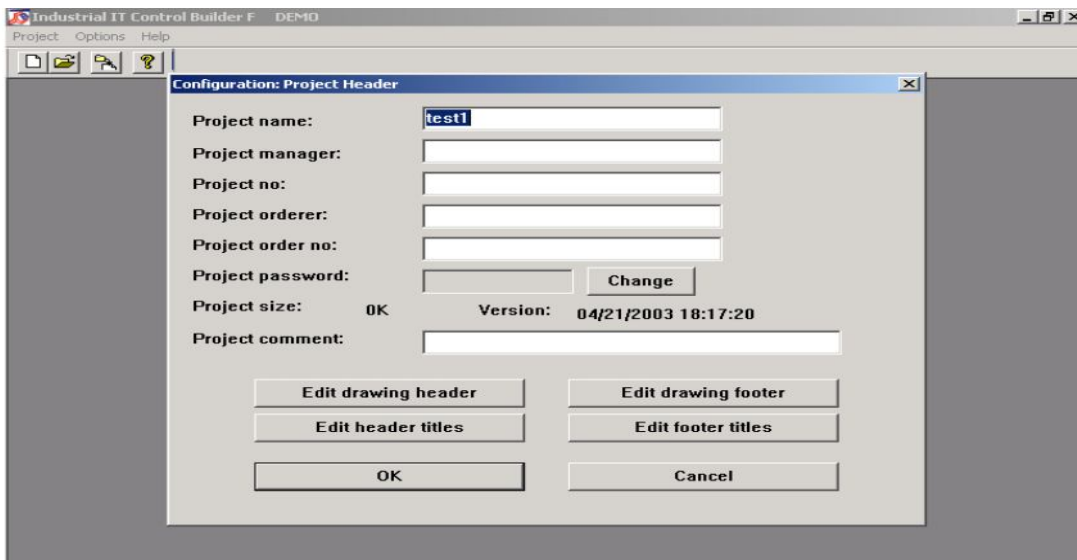
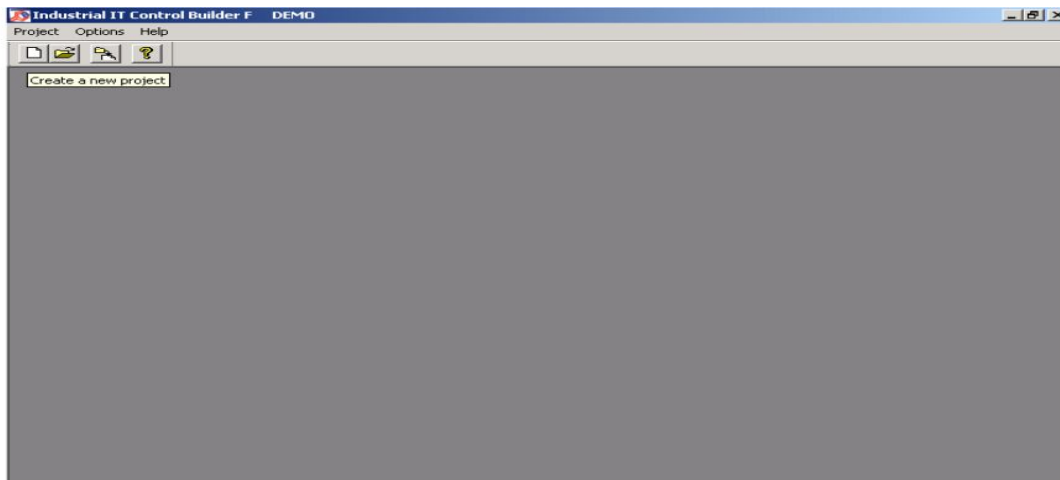
- [14] Albert F. Cutter, Sr "Electrician's Guide to Control and Monitoring Systems" McGraw-Hill.
- [15] Jonathan Love "Process Automation Handbook", Springer-Verlag London Limited (2007).
- [16] ABB Automation "Freelance 2000 DigiVis Operators Manual-Operator Station".

APPENDICES

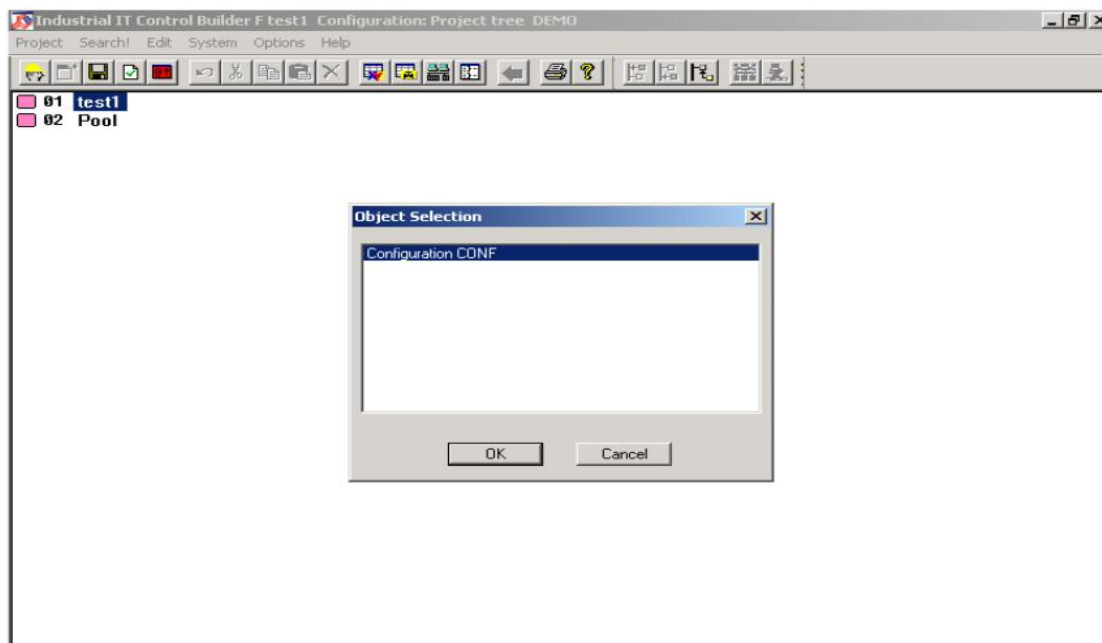
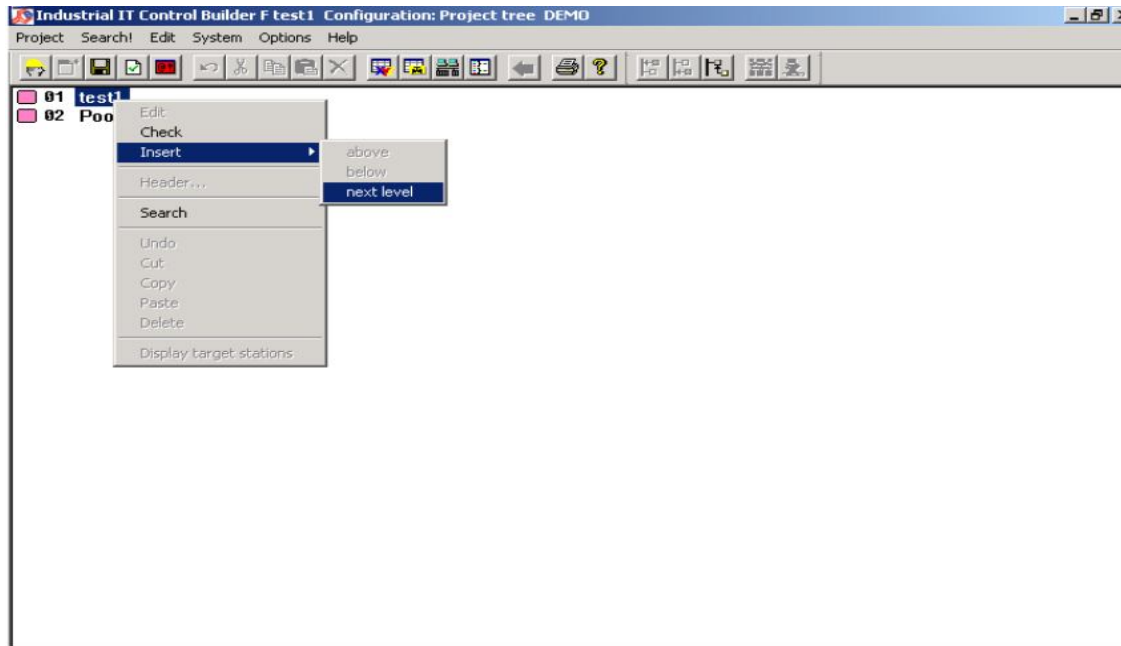
This appendices details the steps needed to create a project and simulate it using the Freelance 2000 and development software.

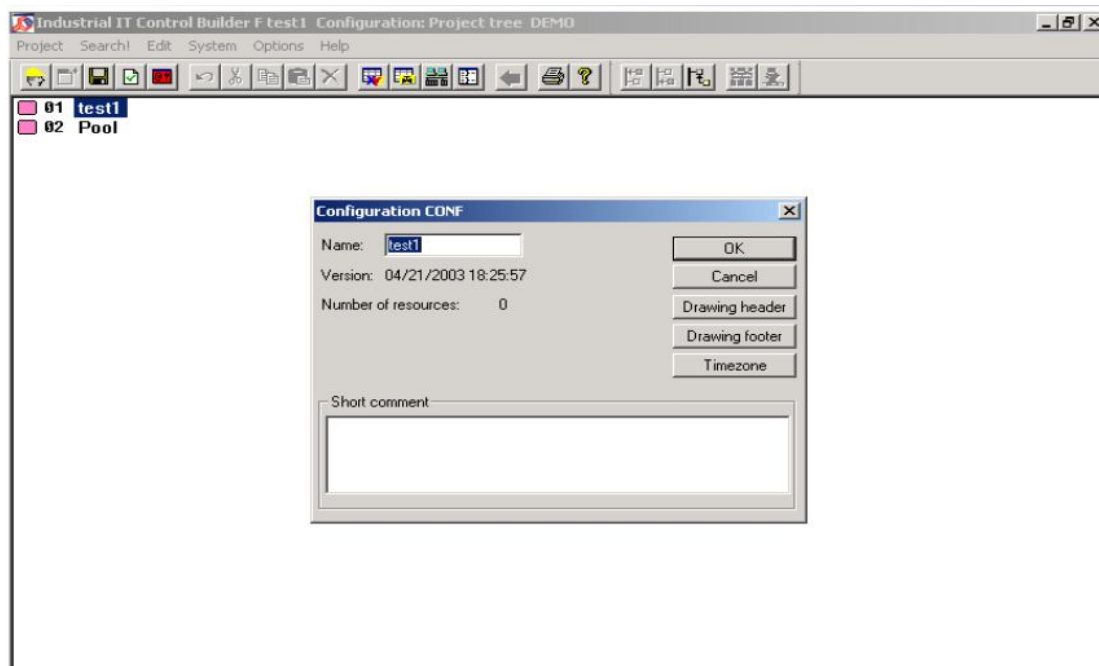
It can help through the detailed tasks involved in the implementation, simulation, and testing of typical industrial automation tasks. The focus here is on the graphical user interface, including both command and status.

Appendix A: Creating New Projects

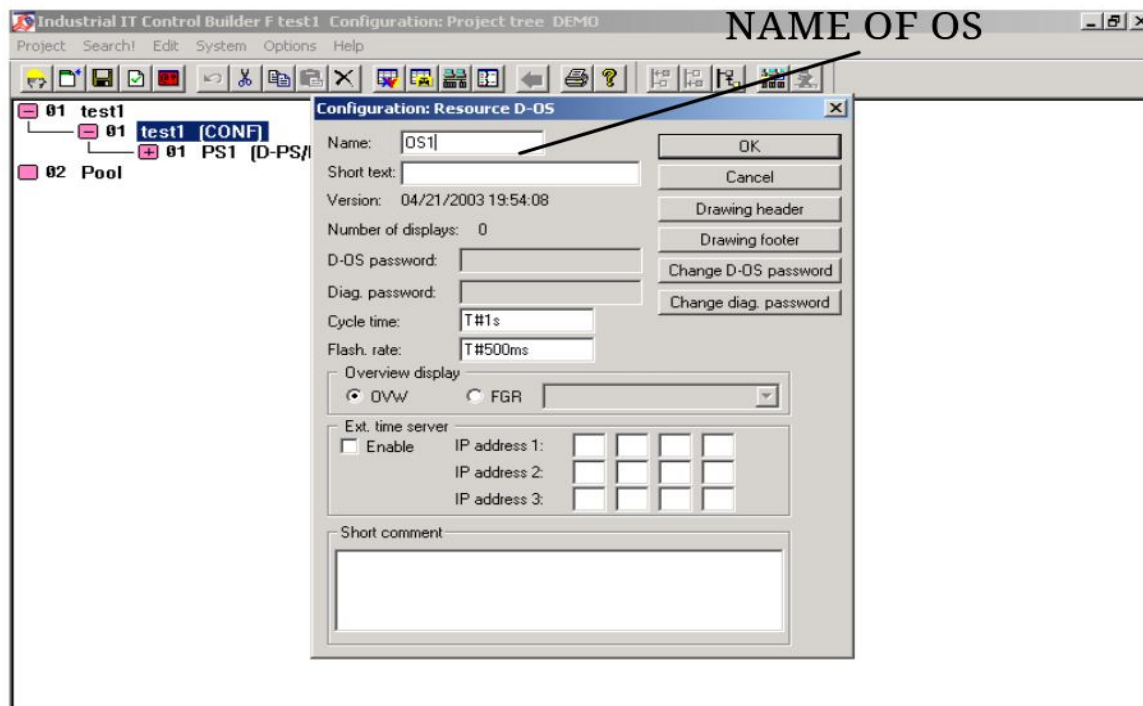
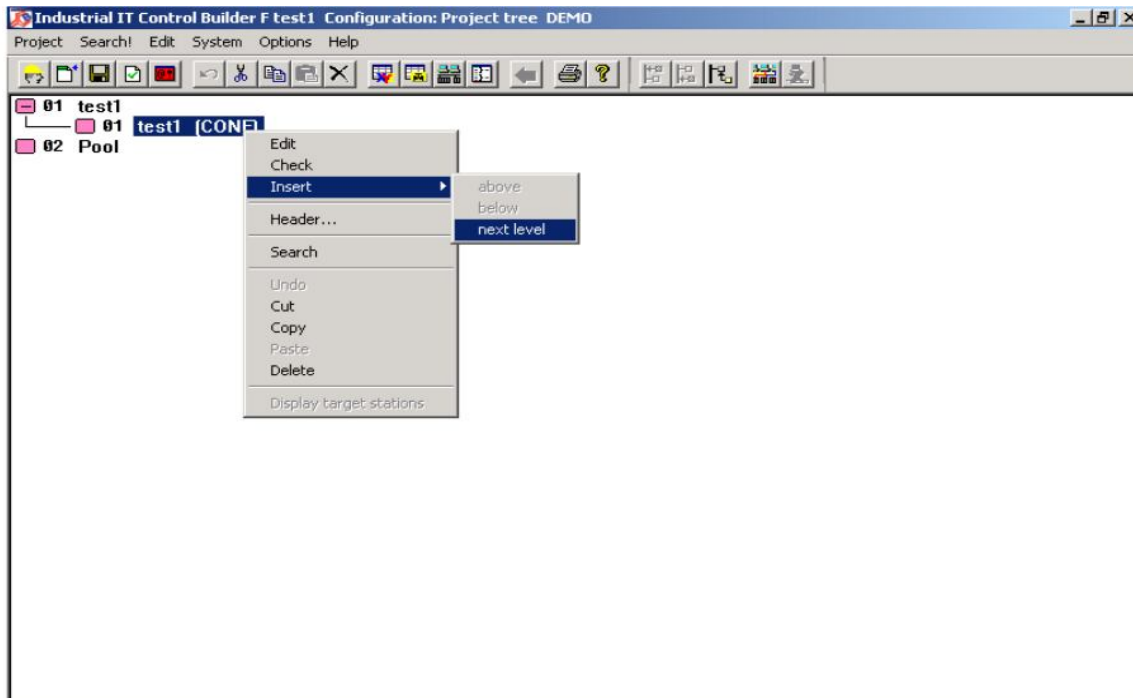


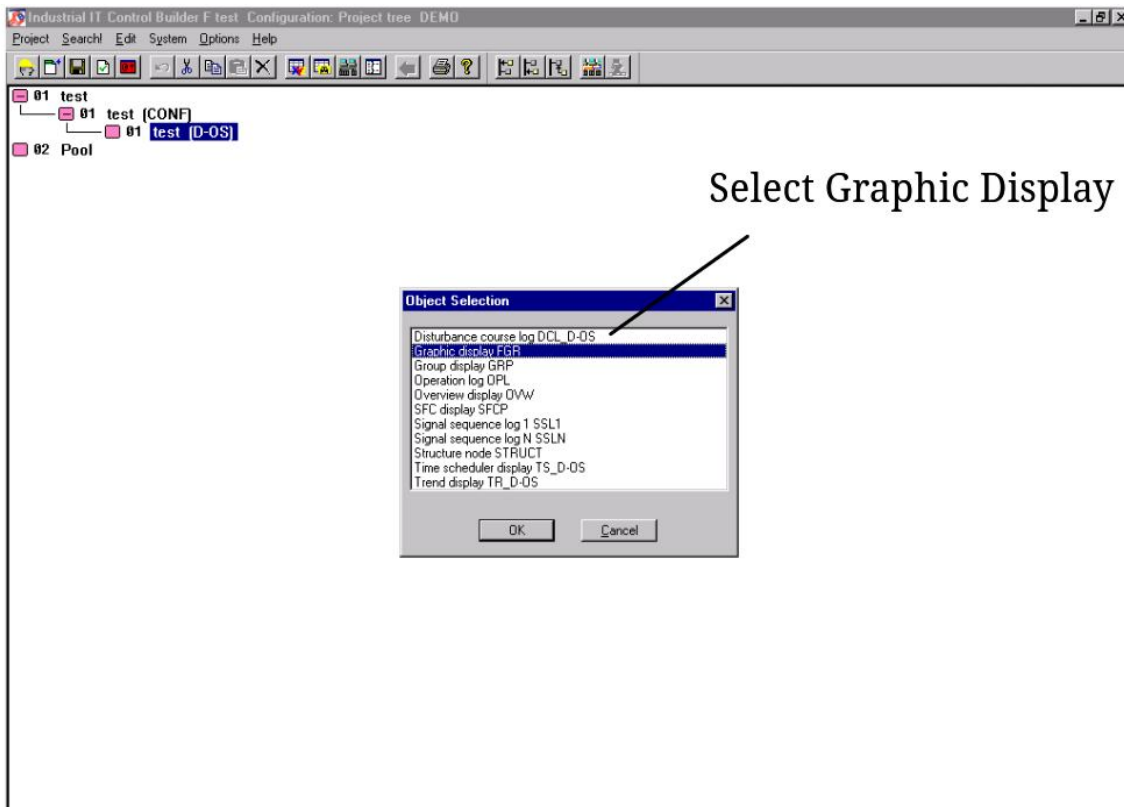
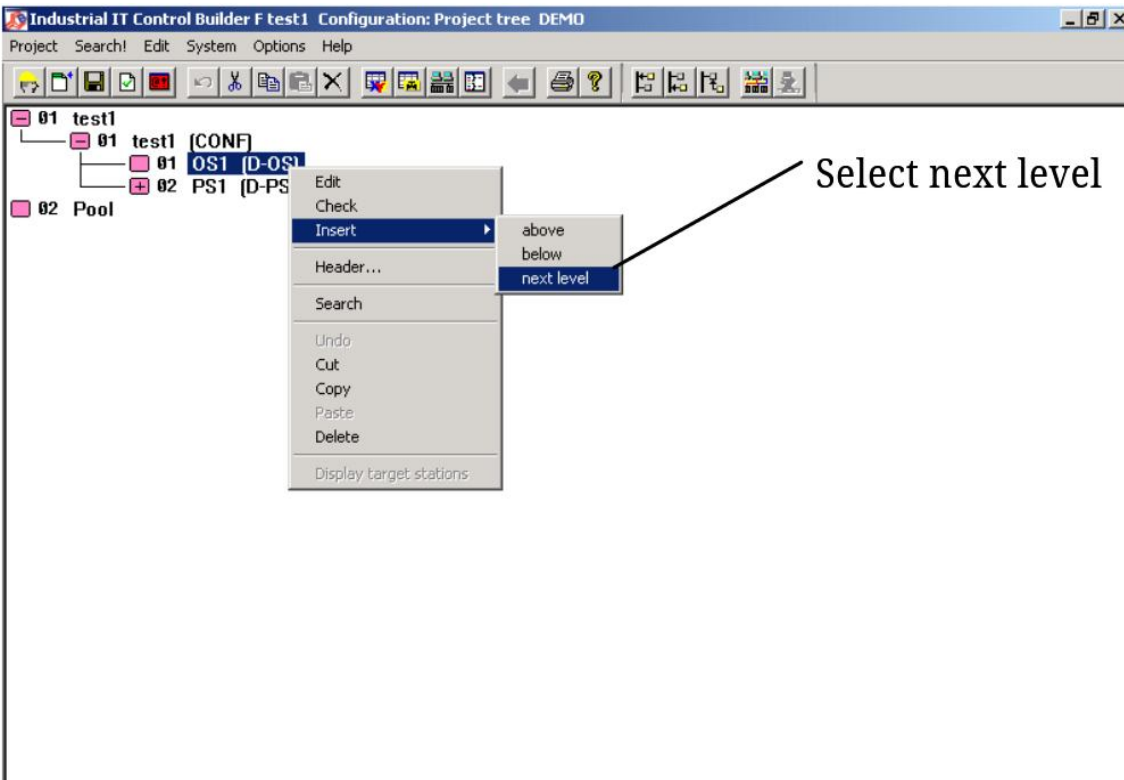
Appendix B: Starting The Project Tree

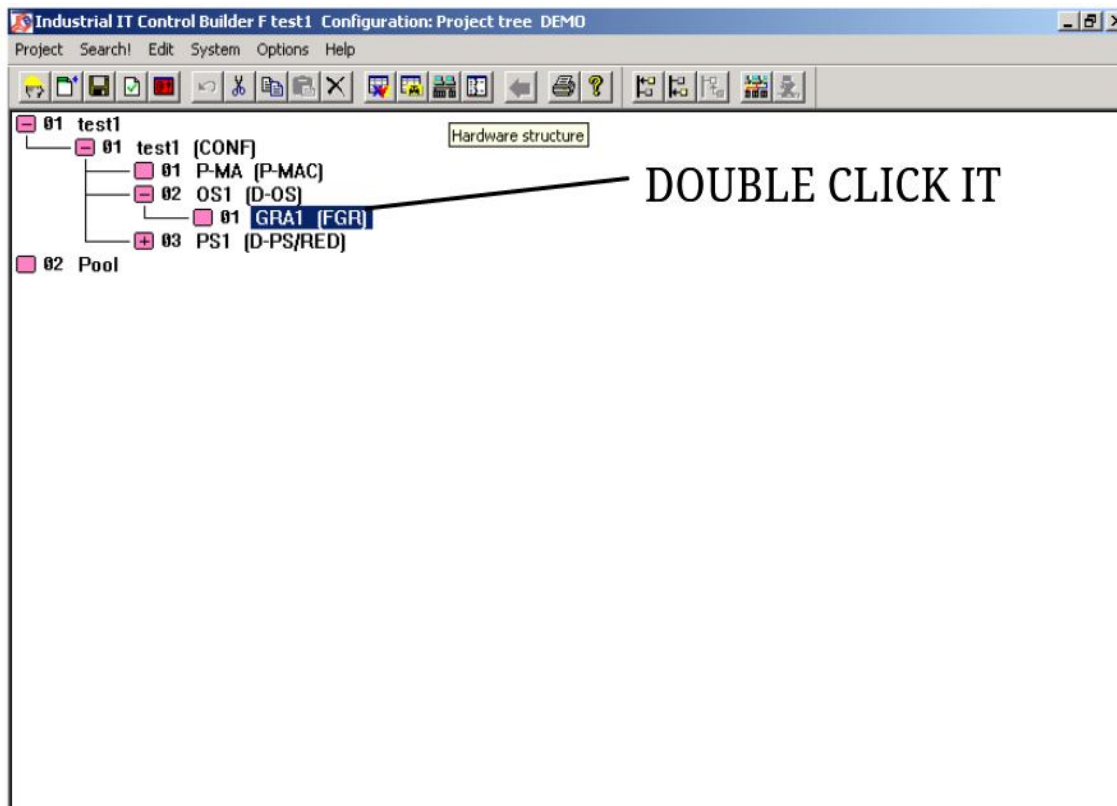
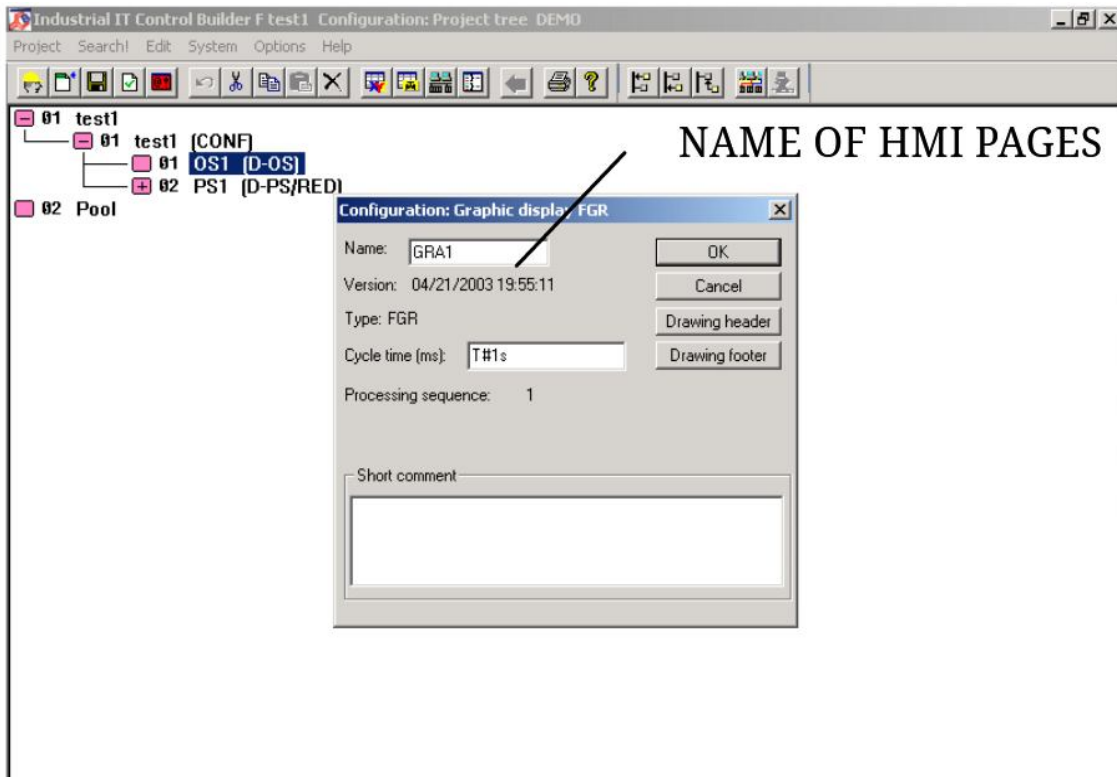




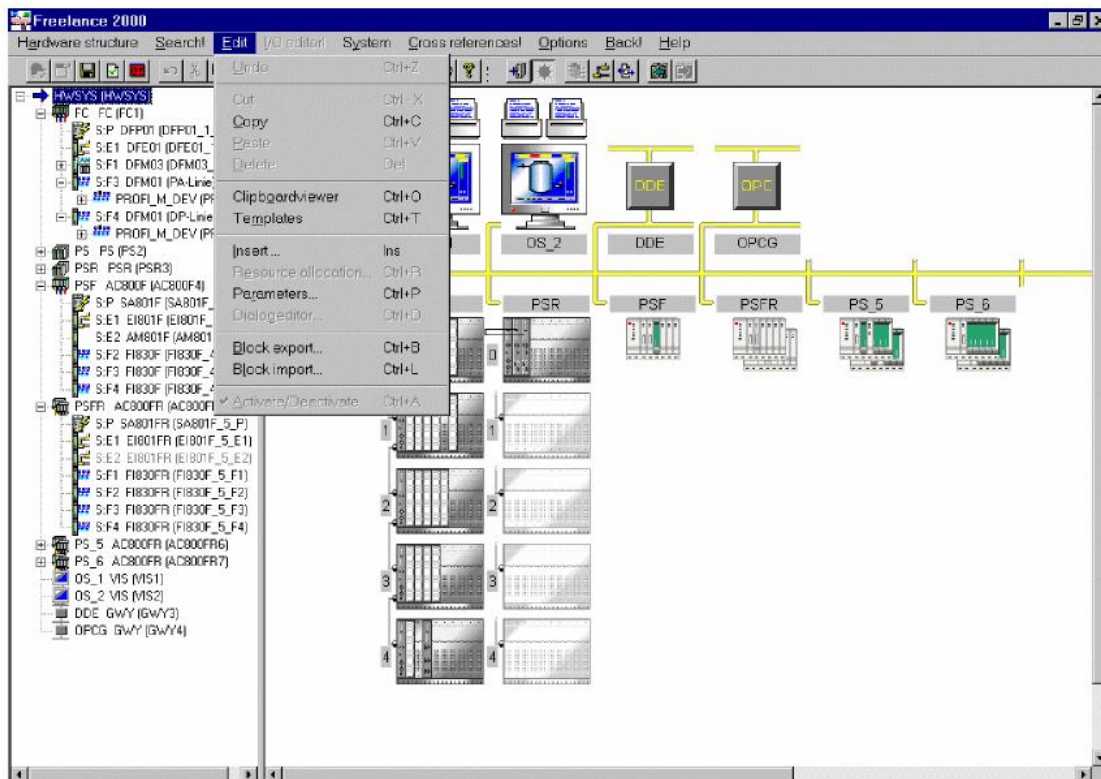
Appendix C: Create Operator Station



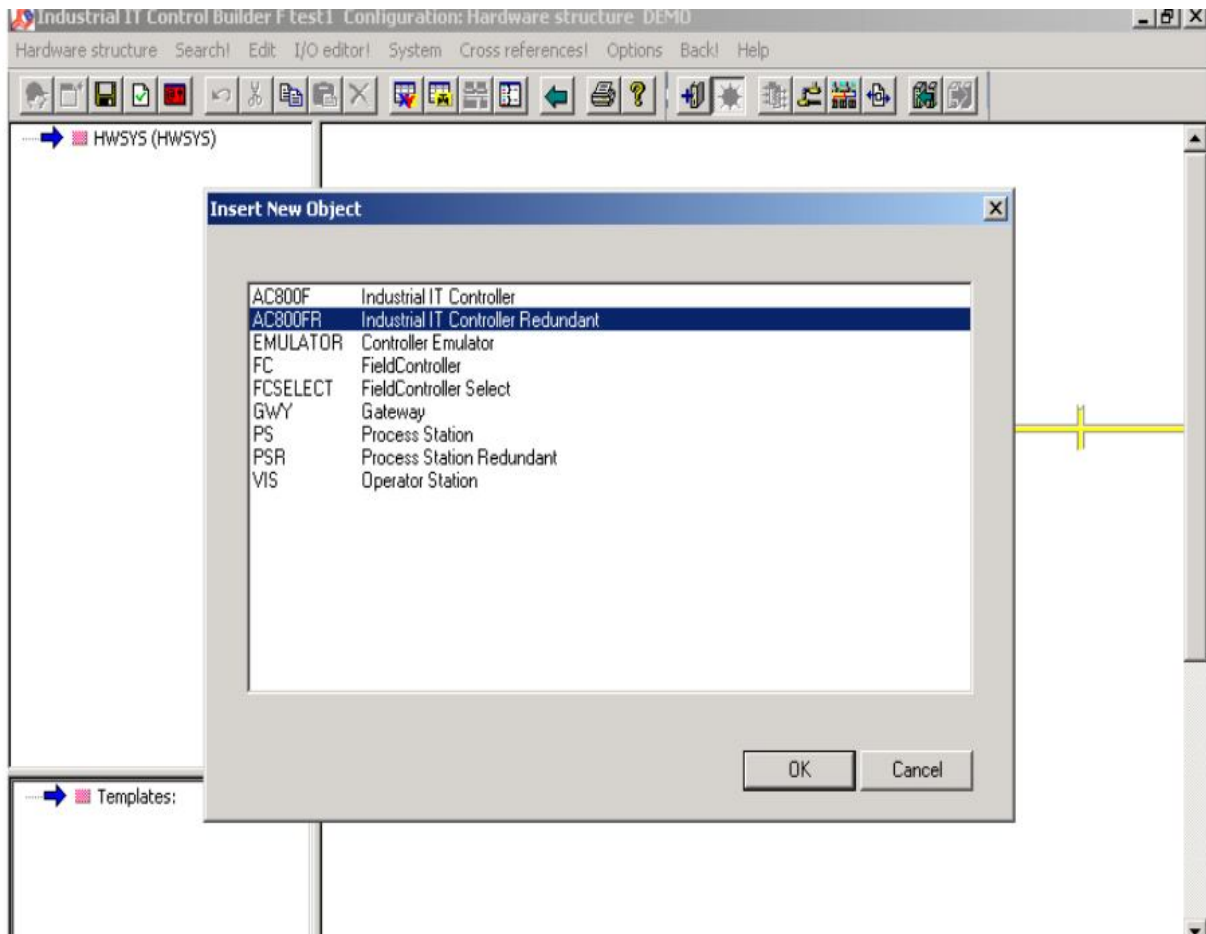




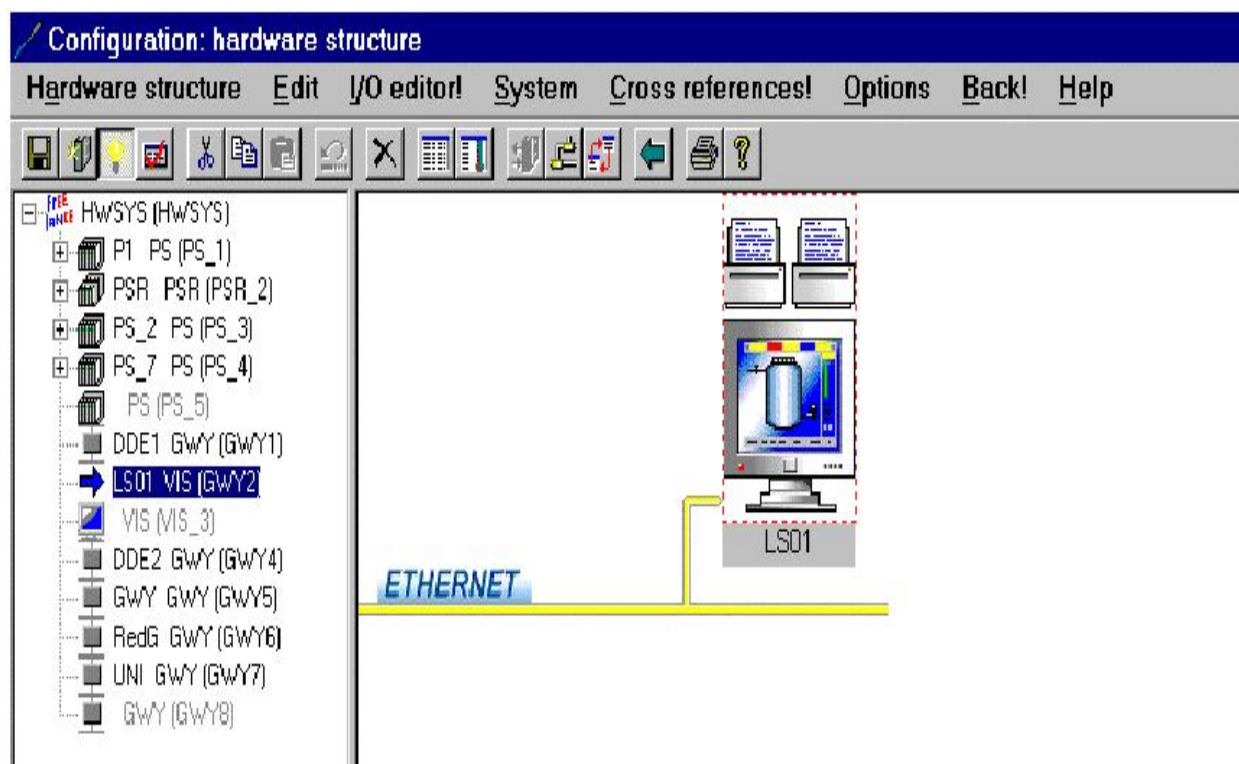
Appendix D: Editing The Hardware Structure



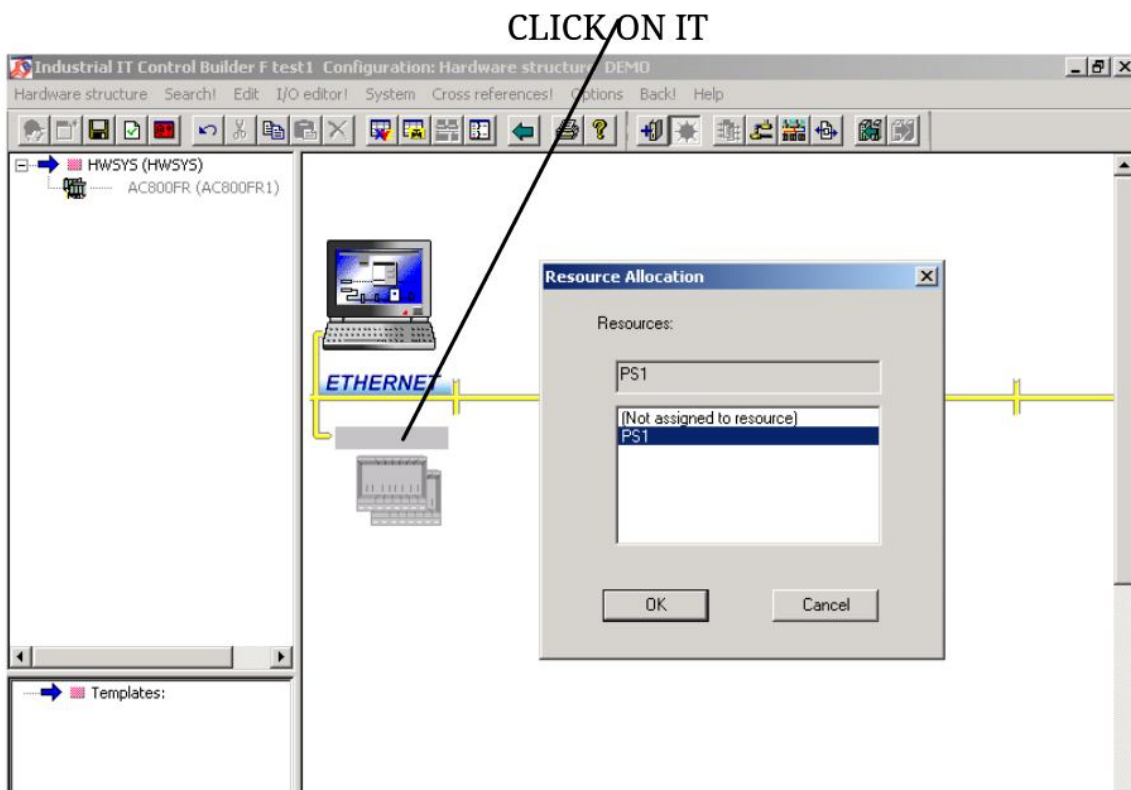
Appendix E: Insert an AC 800F



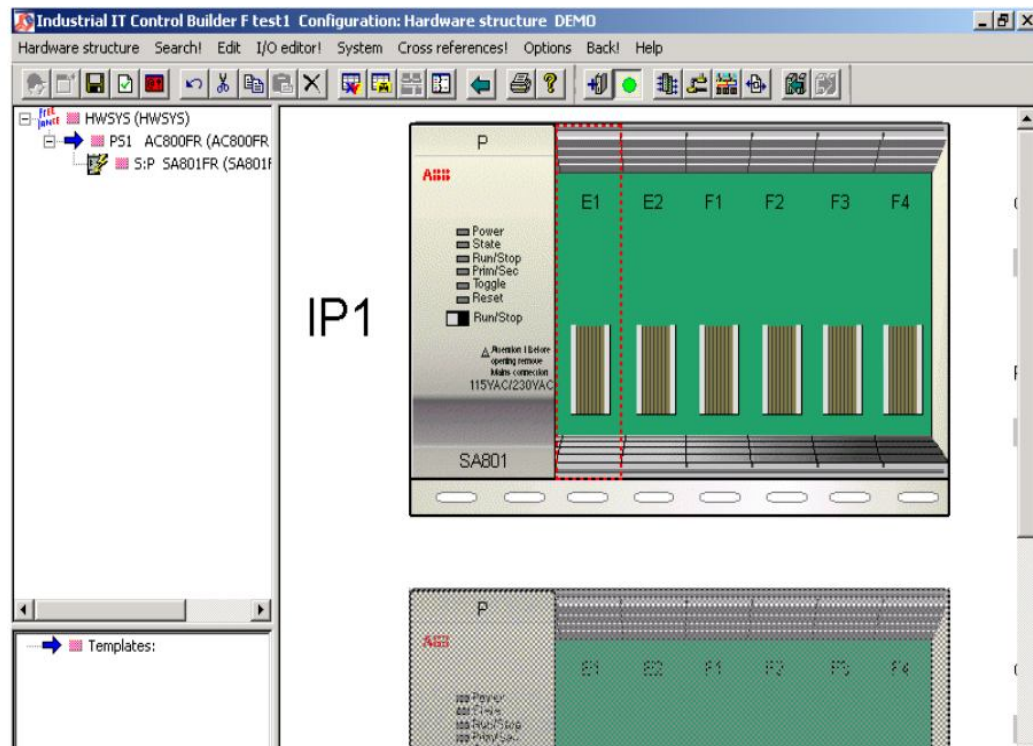
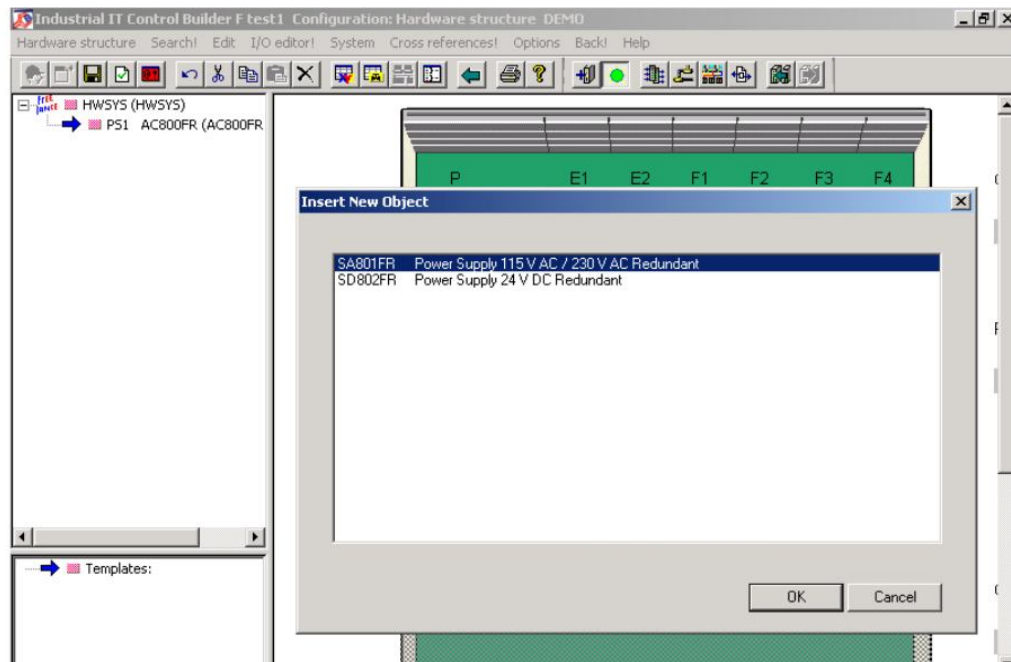
Appendix F: Insert PC Components in The Operator Station

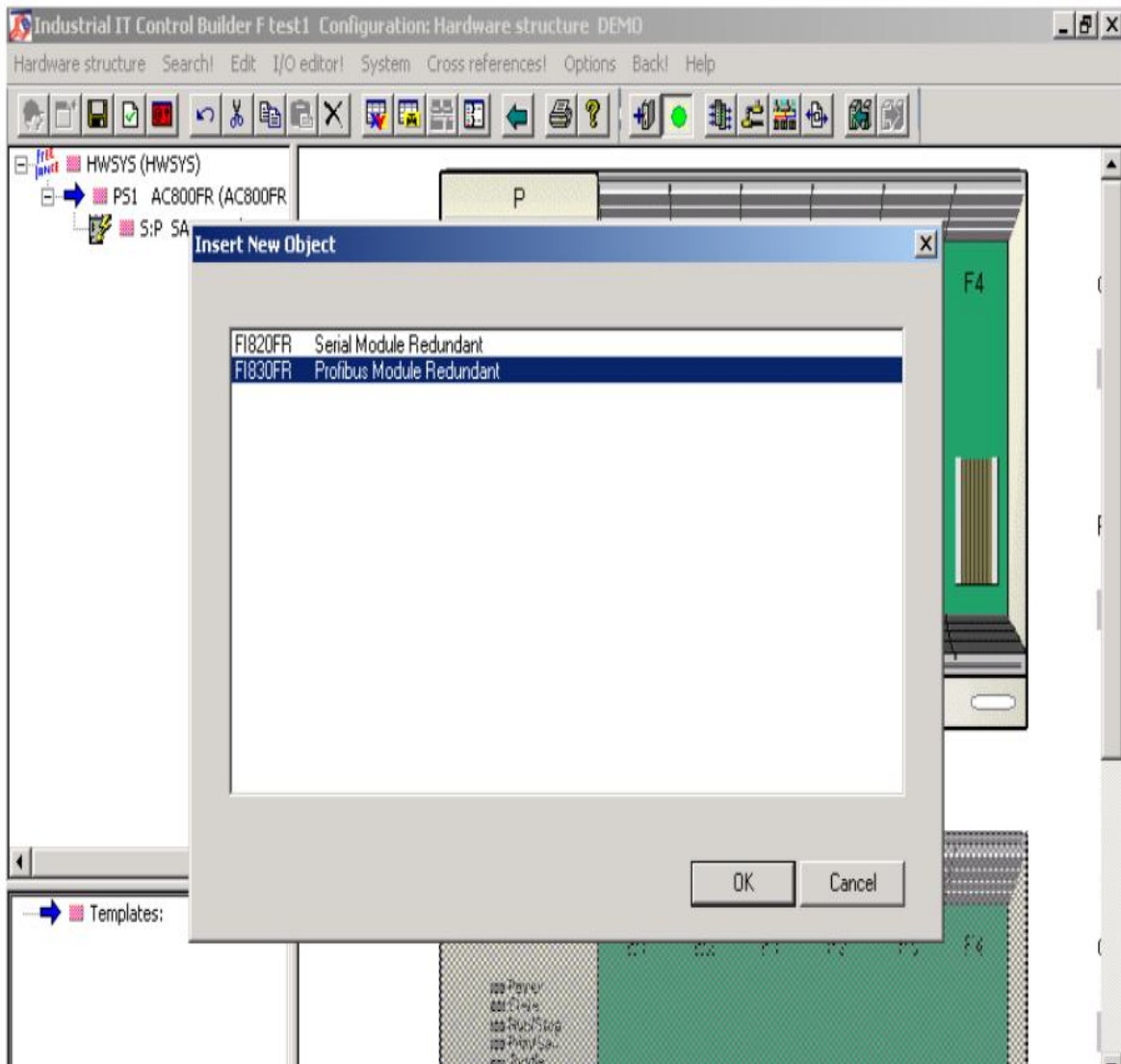


Appendix G: Insert Operator Station

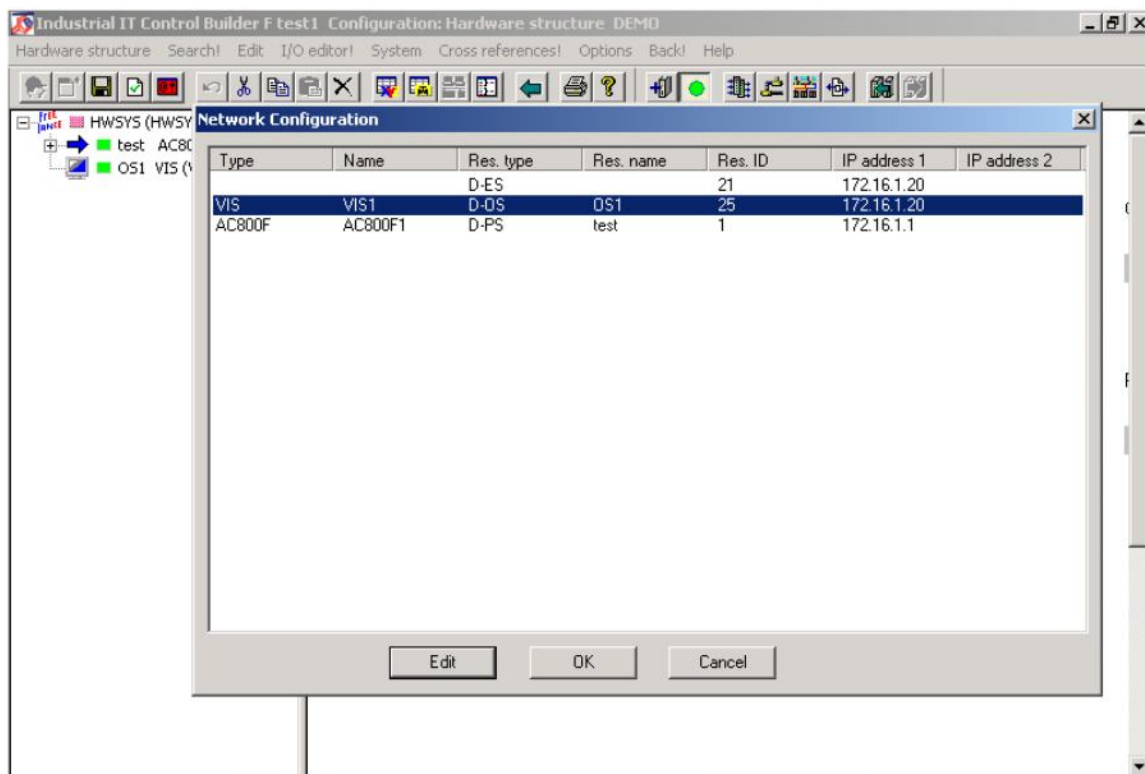


Appendix H: Insert Power Supply, Ethernet and Profibus Master Modules

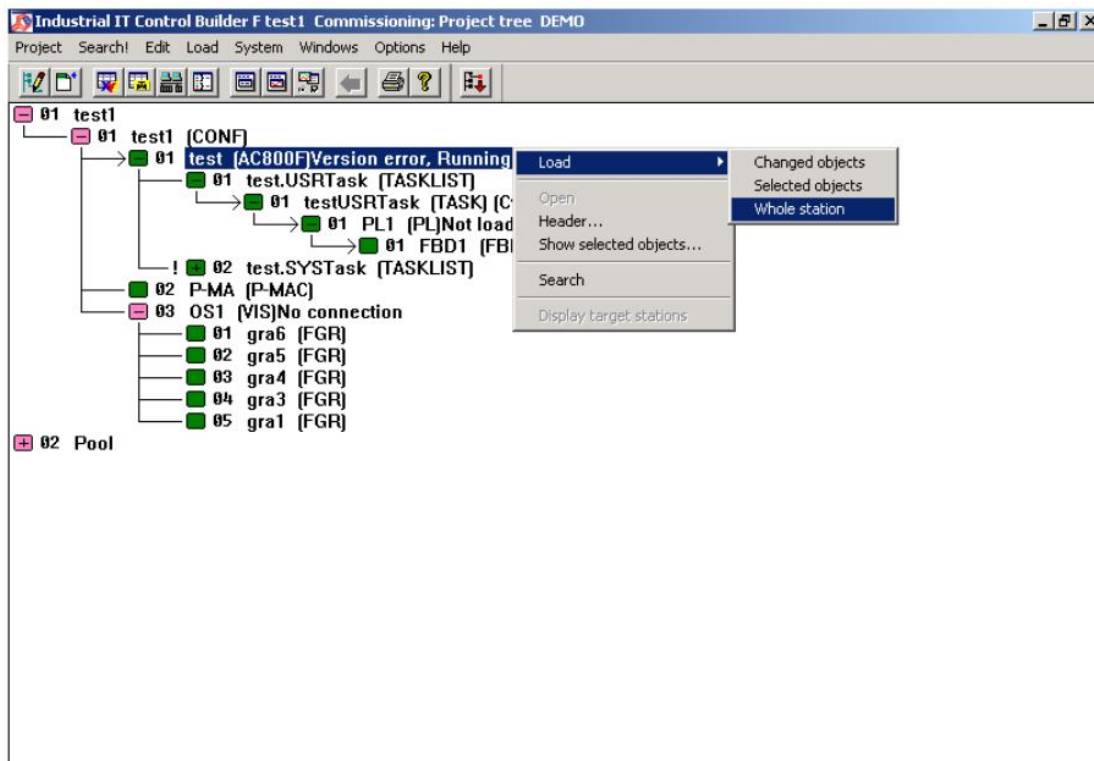
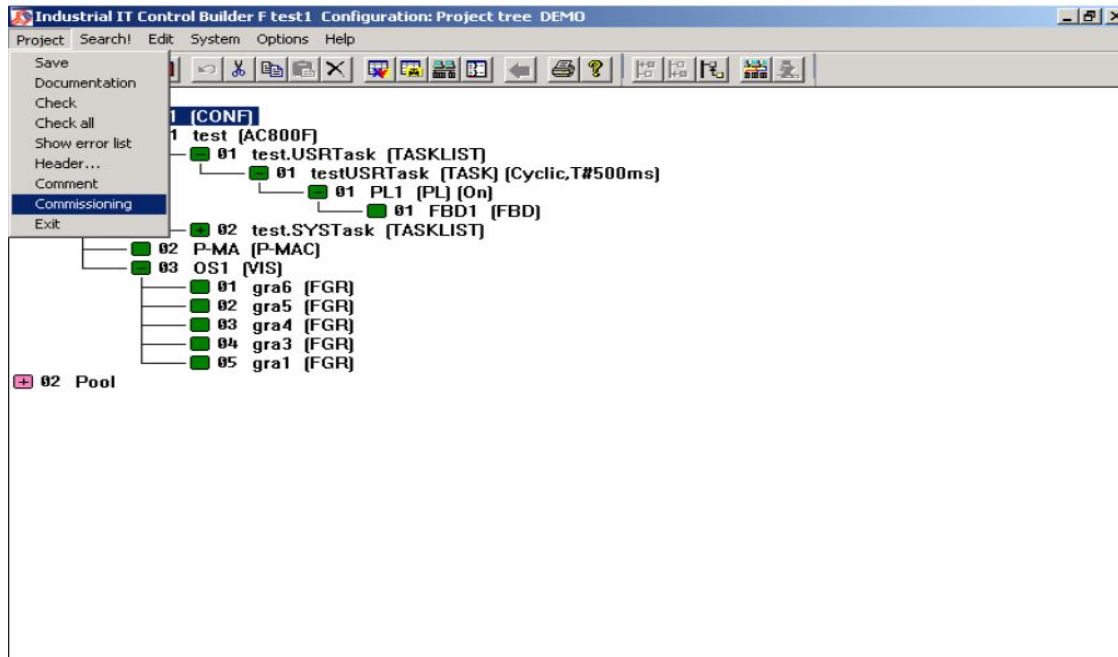












Appendix I: Network Configuration

























Appendix J: Commissioning



Appendix K:Toolbar-Buttons

	Change to configuration mode
	Change to commissioning mode
	Check selected object with all accompanying sub objects
	List all check messages of selected object and the accompanying sub objects
	Save project or project part currently working on
	Create a new project
	Open existing project
	Import a project file
	Export of the current project
	Call online help system
	Edit the general data (header data) of the selected object
	Call up the variable list
	Call up the tag list
	Call up the hardware structure
	Call up the structured data types
	Back to the program from which the current program has been called
	Hardcopy the monitor content
	Show value window
	Show trend window
	Define content for value and/or trend window
	Insert a new object above the selected object
	Insert a new object below the selected object
	Insert a new object in next hierarchical level
	Close all communication links

	Load changed objects into selected station
	Call up dialog editor
	Insert following line
	Show all cross references of selected variable or tag
	Show interface declaration of current user defined function block
	Show parameter mask of the accompanying task
	Toggle display of drawing grid
	Insert a column into a SFC program
	Insert a row into a SFC program
	Delete a column of a SFC program
	Delete a row of a SFC program
	Call up the operation dialog of on SFC program
	Select all graphic objects of a graphic display
	Return to the parameter mask of the graphic object
	Change between graphic display and graphic pool
	Call up the library functions for graphic macros
	Edit a graphic macro
	Activate or deactivate a station or a module
	Show previous hardware object
	Show next hardware object
	Call up the I/O editor
	Call up network editor