1. SCADA SYSTEMS HARDWARE (AND FIRMWARE)

1.1. Introduction

This chapter introduces the fundamental concepts of DCS systems. The terms Supervisory Control and Data Acquisition (SCADA), Distributed Control System (DCS), Programmable Logic Controller (PLC), and Smart Instrument are defined and placed in the context used in this manual. The chapter is split into the following sections:

- Definitions of the terms SCADA, DCS, PLC and smart instrument
- Remote terminal unit (RTU) structure
- PLCs used as RTUs
- System reliability and availability
- Communication architectures and philosophies
- Typical considerations in configuration of a master station

1.1.1. Control Systems

A control system is a collection of components working together under the direction of some machine intelligence. In most of the control systems, electronic circuits provide the intelligence, and electromechanical devices such as sensors, actuators and motors provide the interface to the physical process. For example, in an automobile, various sensors supply the on-board computer with information about the engine's condition. The computer then calculates the precise amount of fuel to be injected into the engine and adjusts the ignition timing. The mechanical components of the system include the engine, transmission, steering, wheels, etc. To design, diagnose or repair such sophisticated systems, one must understand the electronics, machines and control system principles.
In a modem control system, electronic intelligence controls some physical process. Control systems are 'automatic' in such things as automatic pilot and automatic washer.

Because the machine itself is making the routine decisions, the human operator is free to do other things. In many cases, machine intelligence is better than direct human control as it can react faster or slower, keep track of long-term slow changes, respond more precisely, and maintain an accurate log of the system's performance.
The subject of control systems involves multiple disciplines like - electronics, power control devices, sensors, motors, mechanics, and control system theory, which tie together all these concepts.

1.1.2. History of control systems

Since the beginning of life, natural control systems have existed with human beings. Let us consider how the human body regulates temperature. If the body needs to heat itself, food calories are converted to produce heat; on the other hand, evaporation causes cooling. Because evaporation is less effective, especially in humid climates it is not surprising that our body temperature is set at 98.6°F near the high end of earth's temperature spectrum to reduce demand on the cooling system. If temperature sensors in the body notice a drop in temperature, they signal the body to burn more fuel. If the sensors indicate too high a temperature, they signal the body to sweat.

Man-made control systems have existed in some form since the time of the ancient Greeks. One interesting device described in literature, is a pool of water that could never be emptied. The pool had a concealed float-ball and valve arrangement. When the water level lowers, the float drops and opens a valve that admits more water.

In the twentieth century, electrical control systems came into existence when, electromechanical relays were developed and used for remote control of motors and devices. Relays and switches were also used as simple logic gates to implement some intelligence. Using vacuum-tube technology, significant development in control systems was made during World War II. Dynamic position control systems or servomechanisms were developed for aircraft applications, gun turrets, and torpedoes. Today, position control systems are used in machine tools, industrial processes, robots, cars and office machines, to name a few.

Developments and advents in the field of electronics had an impact on control system design. Solid-state devices started to replace the power relays in motor control circuits. Transistors and integrated circuit operational amplifiers became an integral part of analog controllers. Digital integrated circuits replaced bulky relay logic. During this period, the processes or so-called automatic machines were controlled either by analog electronic circuits, or circuits using switches, relays, and timers. After the advent of the microprocessor, the control devices and systems have undergone a lot of redesign to incorporate a microprocessor controller. The advantage of the increased processing power that comes with the microprocessor has added sophistication and new features to the controllers.

Finally, and perhaps most significantly, the microprocessor allowed for the creation of digital controllers that are inexpensive, reliable, able to control complex processes, and adaptable.

1.2. Comparison of the terms SCADA, DCS, PLC and Smart instrument

1.2.1. SCADA system

A SCADA (Supervisory control and data acquisition) system means a system consisting of a number of remote terminal units (RTUs) collecting field data, connected back to a master station via a communication system. The master station displays the acquired data and allows the operator to perform remote control tasks.

The accurate and timely data (normally real-time) allows optimization of the operation of the plant and process. Further benefits include more efficient, reliable and most importantly, safer operations resulting in lower cost of operations compared to earlier non-automated systems.

There is a fair degree of confusion between the definition of SCADA and process control systems. SCADA has the connotation of remote or distant operation. The inevitable question is how far 'remote' is - typically this means over a distance such that the distance between the controlling location and the controlled location is such that direct-wire control is impractical (i.e. a communication link is a critical component of the system).

A successful SCADA installation depends on utilizing proven and reliable technology, with adequate and comprehensive training of all personnel in the operation of the system.

There is a history of unsuccessful SCADA systems and the contributing factors include inadequate integration of the various components of the system, unnecessary complexity in the system, unreliable hardware and unproven software. Today hardware reliability is less of a problem; but the increasing software complexity is producing new challenges. It should be noted that many operators judge a SCADA system not only by the smooth performance of the RTUs, communication links and the master station (all falling under the umbrella of SCADA system), but the field devices also (both transducers and control devices). The field devices however fall outside the scope of SCADA in this manual and will not be discussed further. A diagram of a typical SCADA system is given in Figure 1.1.

On a more complex SCADA system, there are essentially five levels or hierarchies:

. Field level instrumentation and control devices
. Marshalling terminals and RTUs
. Communications system
. The master station(s)
. The commercial data processing system

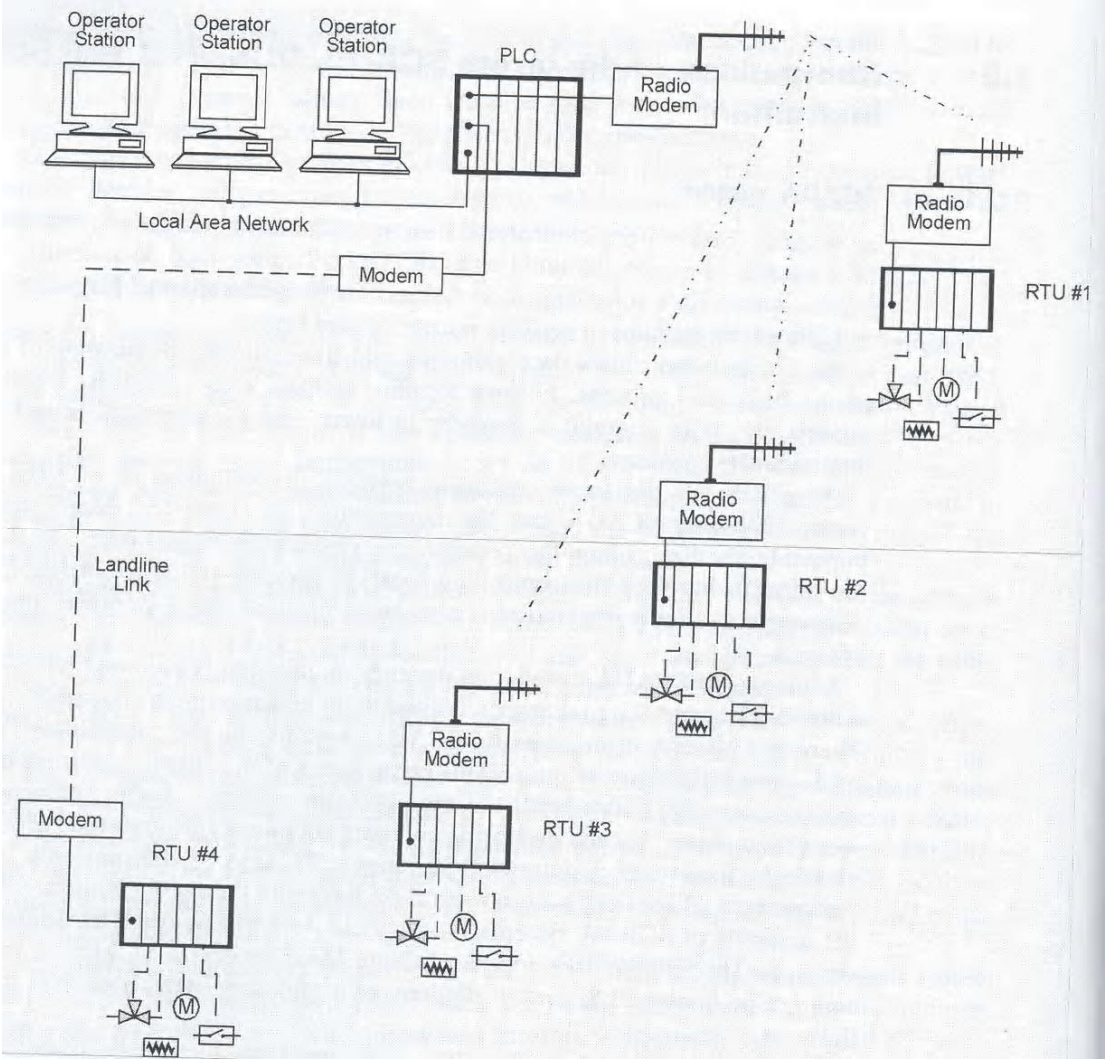

Figure 1.1. Diagram of a typical SCADA system

The RTU provides an interface to the field analogue and digital signals situated at each remote site.

The communications system provides the pathway for communications between the master station and the remote sites. This communication system can be radio, telephone line, microwave and possibly even satellite. Specific protocols and error detection philosophies are used for efficient and optimum transfer of data.

The master station (and sub masters) gathers data from the various RTUs and generally provides an operator interface for display of information and control of remote sites. In large telemetry systems, sub master sites gather information from remote sites and act as a relay back to the control master station.

SCADA technology has existed since the early sixties and there are now two other competing approaches possible - Distributed control system (DCS) and Programmable logic controller (PLC). In addition, there has been a growing trend to use smart instruments as a key component in all these systems. Of course, in the real, the designer mixes and matches the four approaches to produce an effective system matching his/her application (see Figure 1.2)

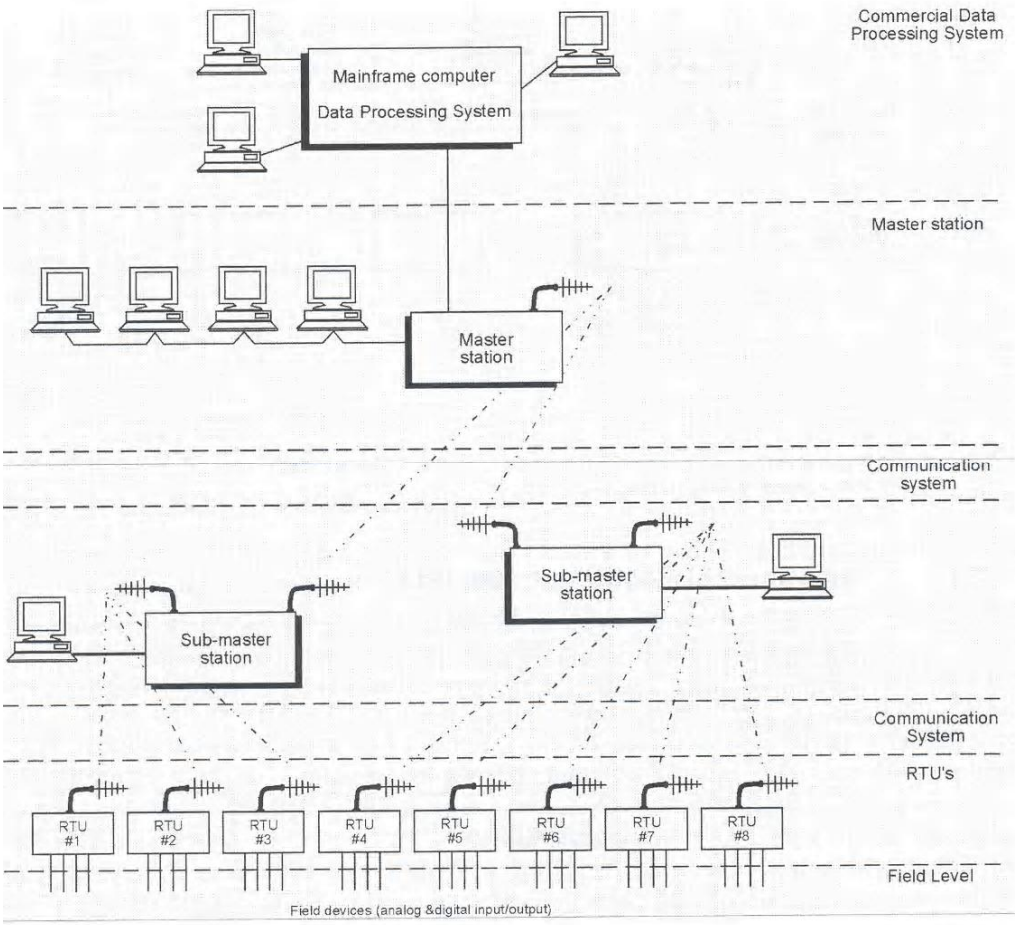

Figure 1.2. SCADA System

1.2.2. Distributed control system (DCS)

In a DCS, the data acquisition and control functions are performed by a number of distributed microprocessor-based units, situated near to the devices being controlled or, the instrument from which data is being gathered. DCS systems have evolved into providing very sophisticated analogue (e.g. loop) control capability. A closely integrated set of operator interfaces (or man machine interfaces) is provided to allow for easy system configurations and operator control. The data highway is normally capable of high speeds-typically 1 Mbps up to 10 Mbps (see Figure 1.3).

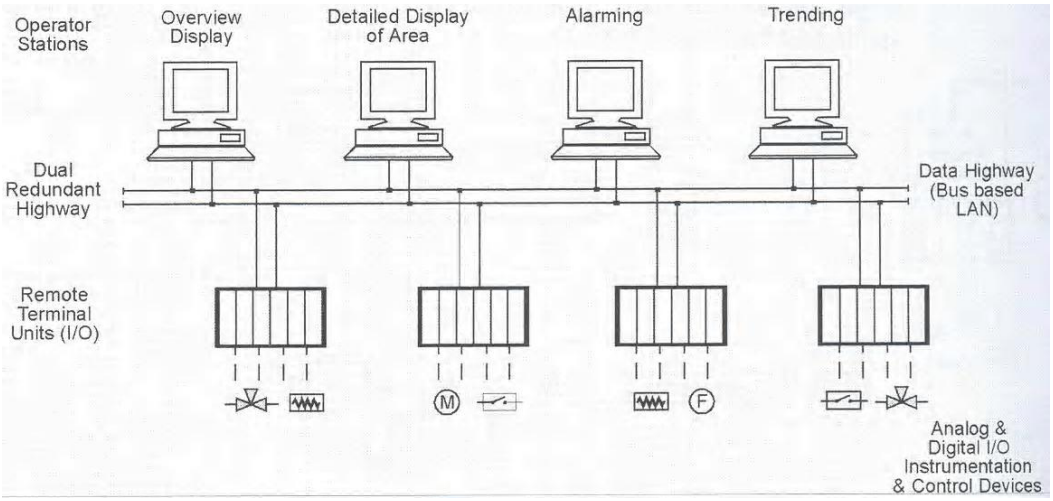

Figure 1.3. Distributed Control System (DCS)

1.2.3. Programmable logic controller (PLC)

Since the late 1970s, PLCs have replaced hardwired relays with a combination of ladder logic software and solid-state electronic input and output modules. They are often used in the implementation of a SCADA RTU as they offer standard hardware solution, which is very economically priced (see Figure 1.4).

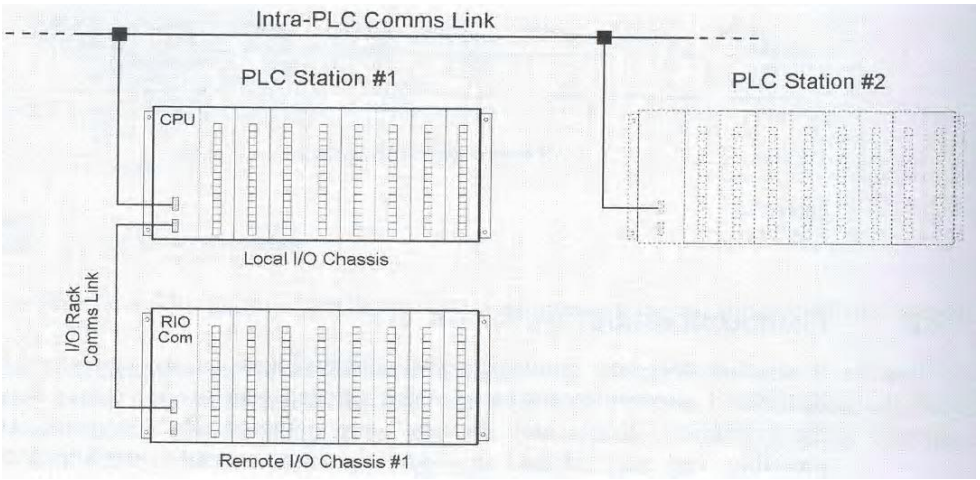

Figure 1.4. Programmable logic controller (PIC) system

1.2.4. Smart instrument

Although this term is sometimes misused, it typically means an intelligent (microprocessor based) digital measuring sensor (such as a flow meter) with digital data communications provided to some diagnostic panel or computer based system (see Figure 1.5).

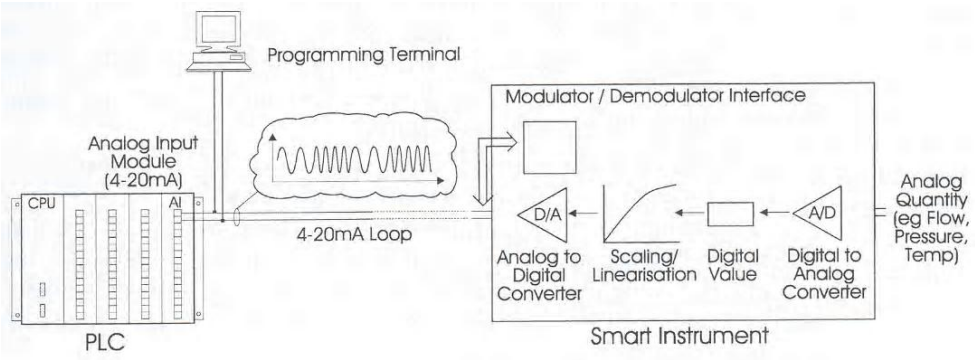

Figure 1.5. Typical Example of a Smart Instrument

1.2.5. DCS versus SCADA terminology

In this manual, we will treat the terms SCADA systems and DCS interchangeably; it is worth considering the differences.

The goals of a DCS (distributed control system) and SCADA (supervisory control and data acquisition system) can be quite different. It is possible for a single system to be capable of performing both DCS and SCADA functions, but few have been designed with this in mind, and therefore they usually fall short somewhere. It has become common for DCS vendors to think they can achieve SCADA functionality, because the system specifications seem so similar, but a few requirements such as data availability, event logging and update processing separates a viable SCADA system from a DCS system.

A DCS is a process-oriented system and it treats the control of the process (the chemical plant, refinery or whatever) as its main task, and it presents data to operators as part of its job. On other hand, a SCADA system is data gathering oriented; and the control center and operators are its focus. Interestingly enough, the remote equipment is merely there to collect the data - though it may also do some very complex process control.

A DCS operator station is intimately connected with its input/output signals (I/O) through local wiring, communication buses (e.g. FieldBus, networks) etc. When the DCS operator wants to see information he/she usually makes a request directly to the field I/O and gets a response. Field events can directly interrupt the system and advise the operator.

A SCADA system must continue to operate when field communications have failed. The 'quality' of data shown to the operator is an important facet of SCADA system operation. SCADA systems often provide special 'event' processing mechanisms to handle conditions that occur between data acquisition periods.

There are many other differences, but they tend to involve a lot of detail. The underlying points are:

- A SCADA system needs to transfer secure data and control signals over a potentially slow, unreliable communications medium, and needs to maintain a database of 'last known good values' for prompt operator display. It frequently needs to do event processing and data quality validation. Redundancy is usually handled in a distributed manner.
- A DCS is always connected to its data source, so it does not need to maintain a database of 'current values'. Redundancy is usually handled by parallel equipment, not by diffusion of information around a distributed database.

These underlying differences prompt a series of design decisions that require a great deal of more complexity in a SCADA system database and data-gathering system than is usually found in a DCS. DCS systems typically have correspondingly more complexity in their process-control functionality.

Some industries use both DCS and SCADA products. The operator stations for each product line can use the same UNIX workstations. The systems share data (and thus form a composite SCADA/DCS system), but the SCADA database architecture is significantly different from the DCS data architecture, to the extent that the SCADA master station database looks to the DCS operators very much as if it were directly connected DCS I/O. The DCS people are (of course) keen to simplify this to cut costs. However, they do not yet have a viable alternative for the mechanisms required in SCADA systems to have communications redundancy and data redundancy to provide the sort of SCADA system reliability that the users expect.

The requirement for either a SCADA or DCS system depends on the customer's system requirement specifications. A careful analysis of the data collection and data quality requirements will indicate which type of system i.e. SCADA or DCS is most appropriate. However, the more features a system provides the more it will cost, so if the customer does not need SCADA-type data gathering facilities it will usually be more economical to use a DCS-type system. Briefly, DCS and SCADA are still two different systems, and depend on what the customer specifies, as to which is appropriate for a particular installation.
Another major difference in DCS/SCADA is their alarm and events philosophies. To put it in very simplistic terms, a SCADA system is event driven, while a DCS is process state driven. A DCS is primarily interested in process trends; a SCADA system in process events.
A SCADA master station or HMI system generally considers changes of state (both status points and analogue changes leading to alarms) as the main criteria driving the data gathering and presentation system. Any undetected changes of state simply cannot be missed. This is reflected firstly in the field devices, which are biased to the rapid scanning of digital inputs, and secondly at the protocol level where transmission of changes of state (COSs) and sequence of events (SOEs) are generally given a higher priority than analogue scans.
SCADA software is event driven. A change of state will cause the system to generate all alarms; events, database updates and any special processing required relating to that change directly from the recognition of that change (including any analogue alarms).

Event and alarm lists are of major importance to the operator, sometimes more than data screens. Filtering of these lists is often quite complex, allowing displays sorted by plant/system area and alarm/event category/importance. Configuring alarms and events for points is relatively easy; as such, attributes are usually added by default when a point is added to a SCADA system database. On the down side, system manufacturers can neglect display of process data. It can be difficult to draw and configure system displays, and graphics can be disappointing, although modern operating systems with off the shelf display packages are overcoming this.

Conversely, DCS systems and process control system based SCADA HMIs are state based and consider the process variable's present and past states to be the main criteria driving the DCS. SCADA protocols are generally register scanning based, with no specific change of state processing provided. Should a point toggle between scans, it will not be seen by the DCS. If any change of states is critical (as some would be for a DCS used for SCADA applications), a point must be latched on until it is confirmed it has been scanned, which can be difficult and non-deterministic. Field devices do not scan points rapidly, but may be able to present them to the DCS in an overall faster time.

DCS software tasks are generally run sequentially, rather than event driven. Therefore, alarms or events are not generated when a point changes state, but when that particular process is run. Events and alarm lists are secondary in importance to the process displays, and filtering may not be as complex and flexible. Configuring points is a separate task, with points requiring alarms and events needing to be configured in a separate action. On the up side, the generation and display of data, especially analogue trends and standard process blocks, is far more user friendly and easier for both operator and engineer. But, there are many exceptions to these generalities, and many DCS manufacturers have produced systems to deal with changes of state (COSs) (both by producing event driven base systems and 'special' COS description alarming), and similarly there are SCADA systems with greater data acquisition and process control capability.

The technical difference between DCS & SCADA is in their field I/O or control unit systems, which can often be seen and this will be explained in a later section.
Note: This manual will henceforth consider DCS, PLC and smart instruments as variations or components of the basic SCADA concept.

1.3. Considerations and benefits of SCADA system

Typical considerations when putting a SCADA system together are:

- Overall control requirements
- Sequence logic
- Analogue loop control
- Ratio and number of analogue to digital points
- Speed of control and data acquisition
- Master/operator control stations
- Type of displays required
- Historical archiving requirements

- System consideration
- Reliability/availability
- Speed of communications/update time/system scans rates
- System redundancy
- Expansion capability
- Application software and modeling

Obviously a SCADA system's initial cost has to be justified. A few typical reasons for implementing a SCADA system are:

- Improved operation of the plant or process resulting in savings due to optimization of the system.
- Increased productivity of the personnel.
- Improved safety of the system due to better information and improved control.
- Protection of the plant equipment.
- Safeguarding the environment from a failure of the system.
- Improved energy savings due to optimization of the plant.
- Improved and quicker receipt of data so that clients can be invoiced more quickly and accurately.
- Government regulations for safety and metering of gas (for royalties & tax etc).

1.4. Remote terminal units

An RTU (sometimes referred to as a Remote telemetry unit) as its title implies is a standalone data acquisition and control unit, generally microprocessor based, which monitors and controls equipment at some remote location from the central station. RTU is often the term used by DCS vendors, for the device that gathers data from the field input/outputs. Its primary task is to control and acquire data from process equipment at the remote location and to transfer this data back to a central station. It generally also has the facility for having its configuration and control programs dynamically downloaded from some central station. There is also a facility to be configured locally by some RTU programming unit. Although traditionally the RTU communicates back to some central station, it is also possible to communicate on a peer-to-peer basis with other RTUs. The RTU can also act as a relay station (sometimes referred to as a store and forward station) to another RTU, which may not be accessible from the central station.

Small sized RTUs generally have less than 10 to 20 analogue and digital signals; medium sized RTUs have 100 digital and 30 to 40 analogue inputs. RTUs having a capacity greater than this can be classified as large. A typical RTU configuration is shown in Figure 1.6.

A short discussion follows on the individual hardware components. Typical RTU hardware modules include:

- Control processor and associated memory
- Analogue inputs
- Analogue outputs
- Counter inputs
- Digital inputs
- Digital outputs

- Communication interface(s). Power supply
- RTU rack and enclosure

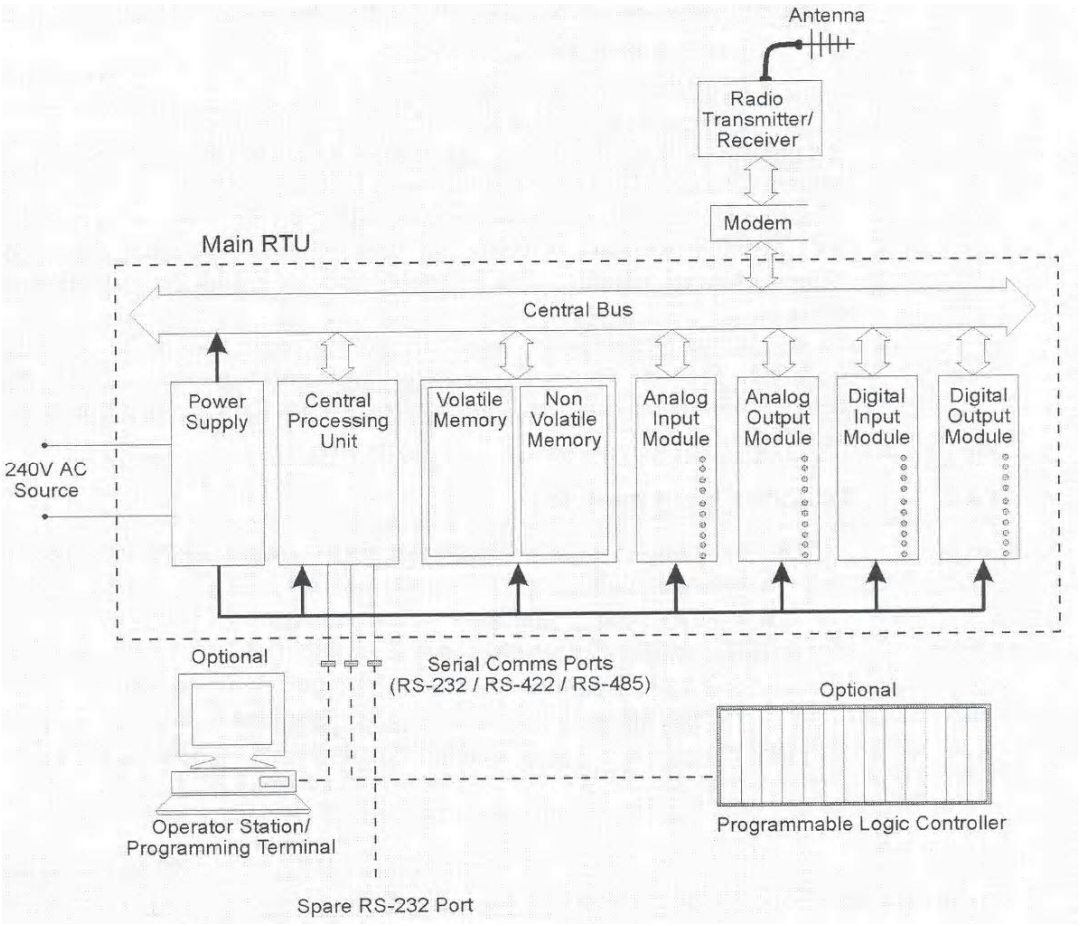

Figure 1.6. Typical RTU Architecture

1.4.1. Control processor (or CPU)

This is generally microprocessor based (16 or 32 bit) e.g. 68302 or 80386. Total memory capacity of 256 kByte (expandable to 4 Mbyte) broken into three types:

EPROM (or battery backed RAM)	256 kByte
RAM	640 kByte
Electrically erasable memory (Flash or EEPROM)	128 kByte

A mathematical processor is a useful addition for any complex mathematical calculation. This is sometimes referred to as a coprocessor.
Communication ports - typically two or three ports either RS-232/RS-422/RS-485 for:

- Interface to diagnostics terminal
- Interface to operator station
- Communications link to central site (e.g. by modem)

Diagnostic LEDs provided on the control unit ease troubleshooting and diagnosis of problems (such as CPU failure/failure of I/O module etc).
Another component, which is provided with varying levels of accuracy, is a real-time clock with full calendar (including leap year support). The clock should be updated even during power off periods. The real-time clock is useful for accurate time stamping of events.
A watchdog timer is also required to provide a check that the RTU program is regularly executing. The RTU program regularly resets the watchdog time. If this is not done within a certain time-out period the watchdog timer flags an error condition (and can reset the CPU).

1.4.2. Analogue input modules

There are five main components making up an analogue input module. They are:

- The input multiplexer
- The input signal amplifier
- The sample and hold circuit
- The A/D converter
- The bus interface and board timing system

A block diagram of a typical analogue input module is shown in Figure 1.7.

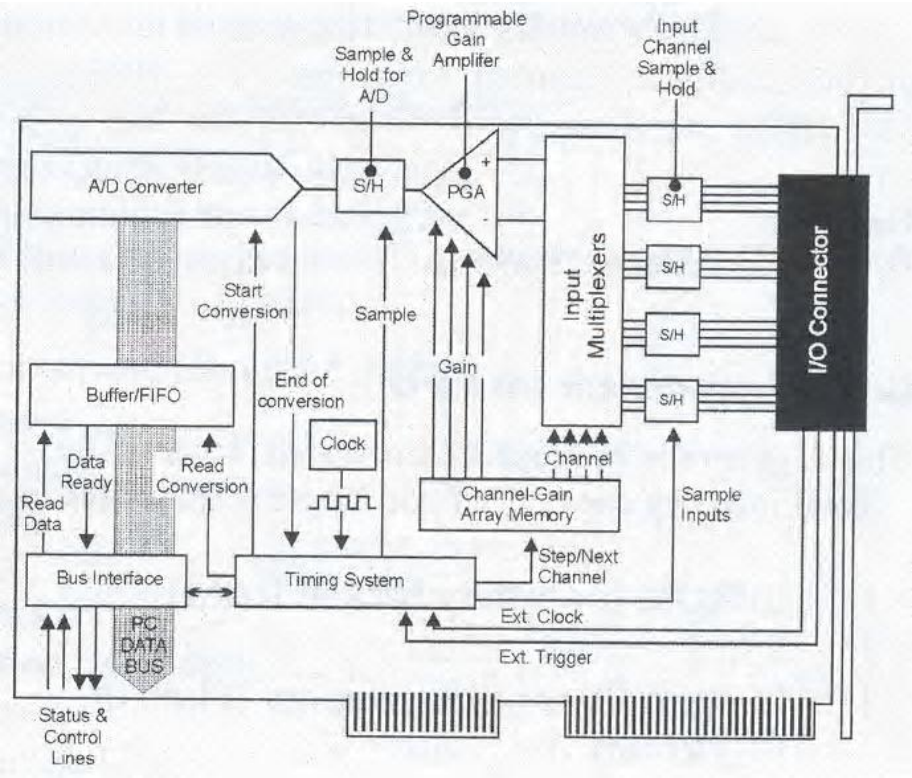

Figure 1.7. Block Diagram of a Typical Analogue Input Module

Each of the individual components will be considered in the following sections.

Multiplexers
A multiplexer is a device that samples several (usually 16) analogue inputs in turn, and switches each to the output in sequence. The output generally goes to an A/D converter, eliminating the need for a converter on each input channel. This can result in considerable savings.

Connection methods
There are two methods of connecting signal sources to the data acquisition board: *Single - ended and differential;* they are shown below. In general, differential inputs should be used for maximum immunity. Single-ended inputs should only be used where it is impossible to use either of the other two methods.
In the descriptions that follow, these points apply:

- All signals are measured relative to the board's analogue ground point, AGND, which is 0 V.
- HI and LO refer to the outputs of a signal source, with LO (sometimes called the *signal return)* being the source' reference point and HI being the signal value. The signal values (VHIn – VLOn), are represented by Esn in the diagrams, where *n* is the signal's channel number.
- AMP LO is the reference input of the board's differential amplifier. It is *not* the same as AGND but it may be referenced to it.
- Because of lead resistance, etc, the remote signal reference point (or ground) is at a different potential to AGND. This is called the *common mode voltage* V_{CM}. In the ideal situation V_{CM} would be 0 V, but in real-world systems V_{CM} is not 0V. The voltage at the board's inputs is therefore *Esn* + V_{CM}.

Single-ended inputs
Boards that accept single-ended inputs have a single input wire for each signal, the source's HI side. All the LO sides of the sources are common and connected to the analogue ground AGND pin. This input type suffers from loss of common mode rejection and is very sensitive to noise. It is not recommended for long leads (longer than 11 m), or for high gains (greater than 5x). The advantage of this method is that it allows the maximum number of inputs, is simple to connect (only one common or ground lead necessary), and it allows for simpler A/D front-end circuitry. We can see from Figure 1.8 that because the amplifier LO (negative) terminal is connected to AGND, what is amplified is the difference between *Esn* + V_{CM} and AGND, and this introduces the common mode offset as an error into the reading. Some boards do not have an amplifier, and the multiplexer output is fed straight to the A/D. Single-ended inputs must be used with these types of boards.

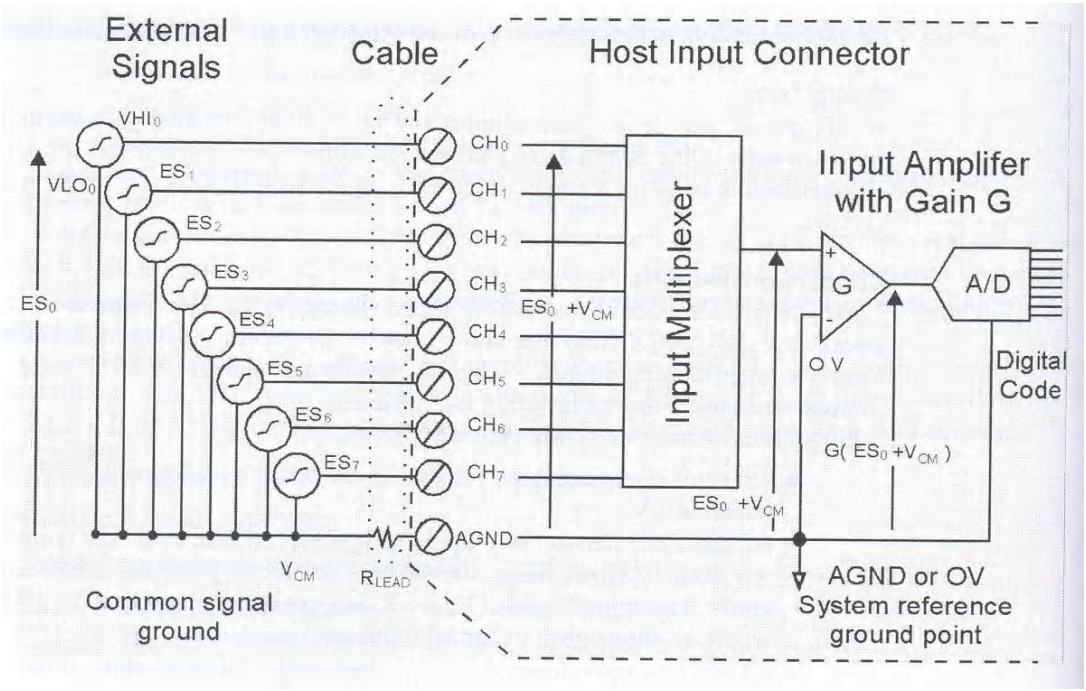

Figure 1.8. Eight Single-ended Inputs

Differential inputs

True differential inputs provide the maximum noise immunity. This method must also be used where the signal sources have different ground points and cannot be connected together. Referring to Figure 1.9, we see that each channel's individual common mode voltage is fed to the amplifier's negative terminal; the individual V_{CMn} voltages are thus subtracted on each reading.

Note that two input multiplexers are needed, and for the same number of input terminals as single-ended and pseudo-differential inputs, only half the number of input channels is available in differential mode. Also, bias resistors may be required to refer each input channel to ground. This depends on the board's specifications (the manual will explain the exact requirements), but it normally consists of one large resistor connected between each signal's LO side and AGND (at the signal end of the cable) and sometimes it requires another resistor of the same value between the HI side and AGND.

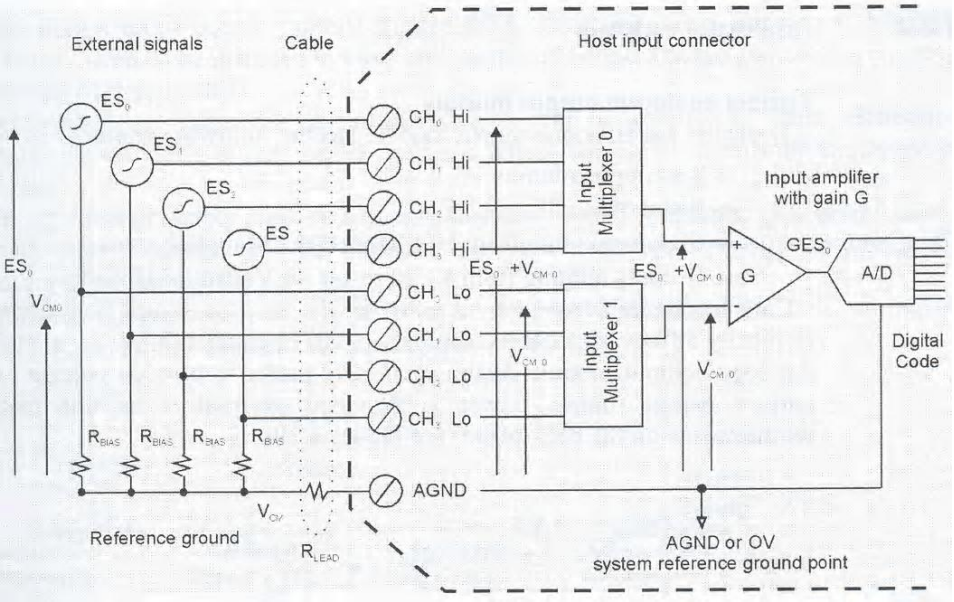

Figure 1.9. Four Differential Inputs

Note that V_{CM} and V_{CMn} voltages may be made up of a DC part and possibly a time varying AC part. This AC part is called noise, but we can see that using differential inputs, the noise part will also tend to be cancelled (rejected), because it is present on both inputs of the input amplifier.

1.4.3. Typical analogue input modules

These have various numbers of inputs. Typically there are:

- 8 or 16 analogue inputs
- Resolution of 8 or 12 bits
- Range of 4-20 mA (other possibilities are 0-20 mA/± l0 Volts/0-l0 Volts)
- Input resistance typically 240 kOhm to 1 MOhm
- Conversion rates typically 10 microseconds to 30 milliseconds
- Inputs are generally single ended (but also differential modes provided)

For reasons of cost and minimization of data transfer over a radio link, a common configuration is eight single ended 8 bit points reading 0-10 Volts with a conversion rate of 30 milliseconds per analogue point.

An important but often neglected issue with analogue input boards is the need for sampling of a signal at the correct frequency. The Nyquist criterion states that a signal must be sampled at a minimum of two times its highest component frequency. Hence, the analogue to digital system must be capable of sampling at a sufficiently high rate to be well outside the maximum frequency of the input signal. Otherwise, filtering must be employed to reduce the input frequency components to an acceptable level. This issue is often neglected due to the increased cost of installing filtering with erroneous results in the measured values. It should be realized the software filtering is NOT a substitute for an inadequate hardware filtering or sampling rate. This may smoothen the signal but it does not reproduce the analogue signal faithfully in a digital format.

1.4.4. Analogue outputs

Typical analogue output module

Typically the analogue output module has the following features:

- 8 analogue outputs
- Resolution of 8 or 12 bits
- Conversion rate from 10 μ seconds to 30 milliseconds
- Outputs ranging from 4 - 20 mA/ ±10 Volts/0 to 10 Volts

Care has to be taken here on ensuring the load resistance is not lower than specified (typically 50 kOhm), or the voltage drop will be excessive.

Analogue output module designs generally prefer to provide voltage outputs rather than current output (unless power is provided externally), as this places lower power requirements on the back plane (see Figure 1.10).

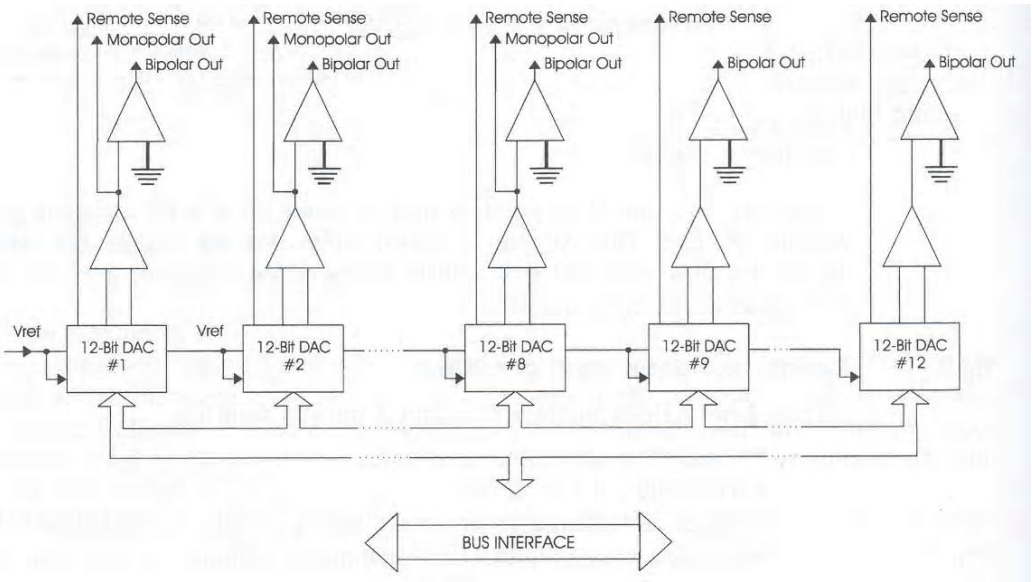

Figure 1.10. Typical analogue output module

1.4.5. Digital inputs

These are used to indicate such items as status and alarm signals. Status signals from a valve could comprise two limit switches with a contact closed indicating valve - open status and the other contact closed indicating valve - closed status. When both open and closed status contacts aren't closed, this could indicate the valve is in transit. (There would be a problem if both status switches indicate closed conditions). A high level switch indicates an alarm condition.

It is important with alarm logic that the RTU should be able to distinguish the first alarm that occurred from the subsequent spurious alarms that occur.

Most digital input boards provide groups of 8, 16 or 32 inputs per board. Multiple boards may need to be installed to cope with numerous digital points (where the count of a given board is exceeded).

The standard normally open or closed converter may be used for an alarm. Generally, normally closed alarm digital inputs are used where the circuit is to indicate an alarm condition.

The input power supply must be appropriately rated for the particular convention used - normally open or closed. For the normally open convention, it is possible to de-rate the digital input power supply.

Optical isolation is a good idea to cope with surges induced in the field wiring. A typical circuit and its operation are indicated in Figure 1.11.

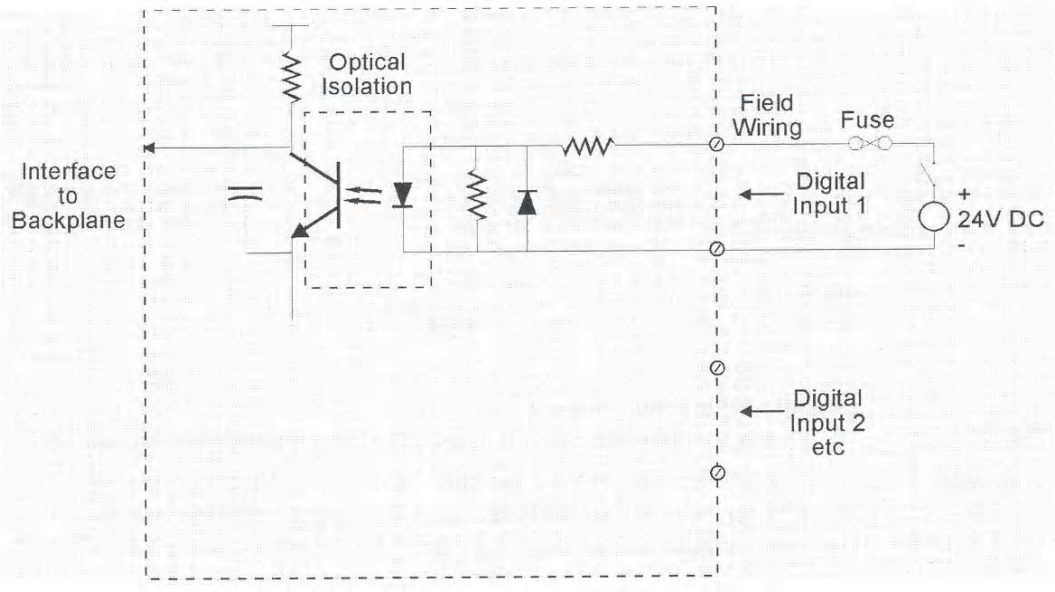

Figure 1.11. Digital input circuit with flow chart of operation

The two main approaches of setting the input module up as a sink or source module are as indicated in Figure 1.12.

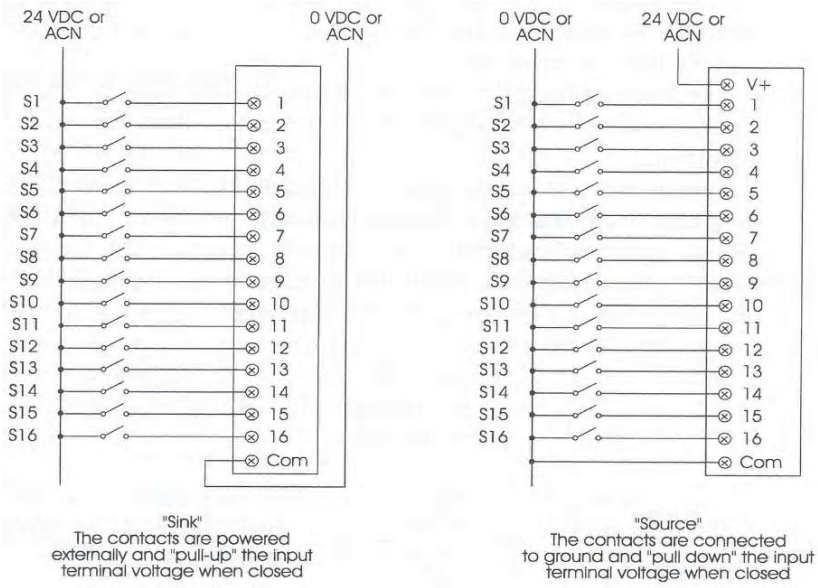

Figure 1.12. Configuring Input Modules as Sink or Source

Typical digital input module

Typically the following would be expected of a digital input module:

- 16 digital inputs per module
- Associated LED indicator for each input to indicate current states
- Digital input voltages vary from 110/240 V AC and 12/24/48 VDC
- Optical isolation provided for each digital input

1.4.6. Counter or accumulator digital inputs

There are many applications where a pulse input module is required - for example from a metering panel. This can be a contact closure signal or if the pulse frequency is high enough, solid state relay signals.

Pulse input signals are normally 'dry contacts' (i.e. the power is provided from the RTU power supply rather than the actual pulse source).

Figure 1.13 gives the diagram of the counter digital input system. Optical isolation is useful to minimize the effect of externally generated noise. The size of the accumulator is important when considering the number of pulses that will be counted before transferring the data to another memory location. For example, a l2-bit register has the capacity for 4096 counts. 16 bits gives 65536 pulses. If these limits are ignored, the classical problem of the accumulator cycling through zero when full could occur.

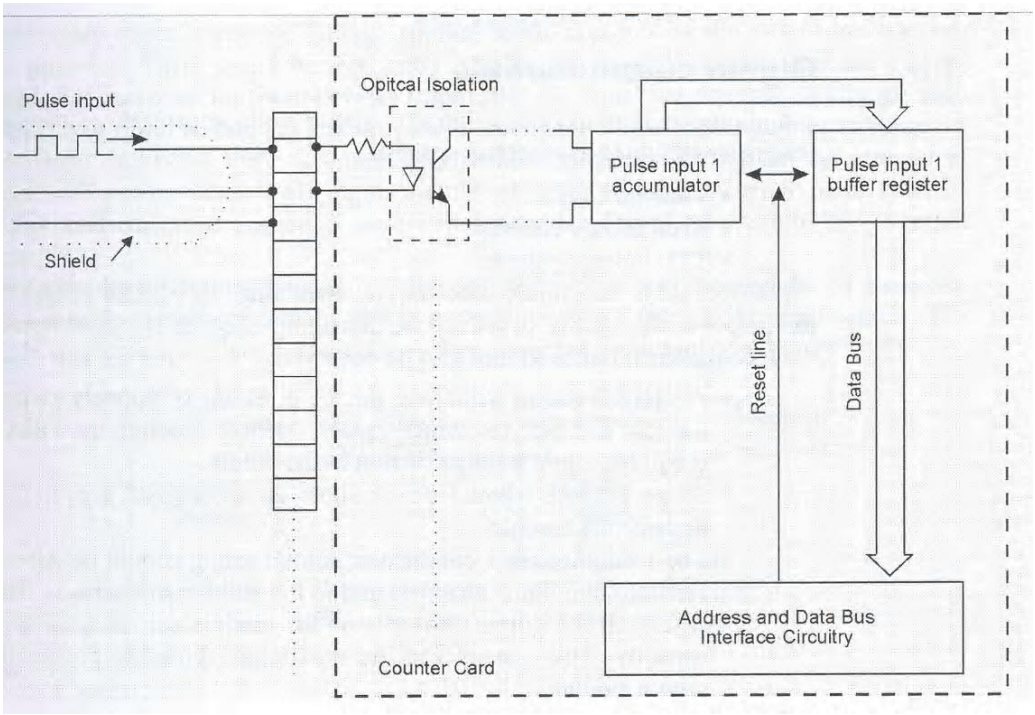

Figure 1.13. Pulse Input Module

Two approaches are possible:

- The accumulator contents can be transferred to RAM memory at regular intervals, where the old and current value differences can be stored in a register.
- The second approach is where a detailed and accurate accounting needs to be made of liquids flowing into and out of a specific area. A freeze accumulator command is broadcast instantaneously to all appropriate RTUs. The pulse accumulator will then freeze the values at this time and transfer to a memory location, and reset the accumulator so that counting can be resumed again.

Typical counter specifications

The typical specifications here are:

- 4 counter inputs
- Four 16 bit counters (65536 counts per counter input)
- Count frequency up to 20 kHz range
- Duty cycle preferably 50% (ratio of mark to space) for the upper count frequency limits

Note that the duty rating is important, as the counter input needs finite time to switch on (and then off). If the on pulse is too short, it may be missed, although the count frequency is within the specified limits.

An Schmitt trigger gives the preferred input conditioning although a resistor capacitor combination across the counter input can be a cheap way to spread the pulses out.

1.5. Digital output module

A digital output module drives an output voltage at each of the appropriate output channels with three approaches possible:

- Triac switching
- Reed relay switching
- TTL voltage outputs

The TRIAC is commonly used for AC switching. A resistor is often connected across the output of the TRIAC to reduce the damaging effect of electrical transients. Three practical issues should also be observed:

- A TRIAC output switching device does not completely switch on and off but has low and high resistance values. Hence, although the TRIAC is switched off it still has some leakage current at the output.
- Surge currents should be of short duration (half a cycle). Any longer will damage the module.
- The manufacturer's continuous current rating should be adhered to. This often refers to individual channels and to the number of channels. There are situations where all the output channels of the module can be used at full rated current capacity. This can exceed the maximum allowable power dissipation for the whole module.

Typical digital output module include:

- 8 digital outputs
- 240 V AC/24 VDC (0.5 Amp to 2.0 Amp) outputs
- Associated LED indicator for each output to indicate current status
- Optical isolation or dry relay contact for each output (see Figure 1.14)

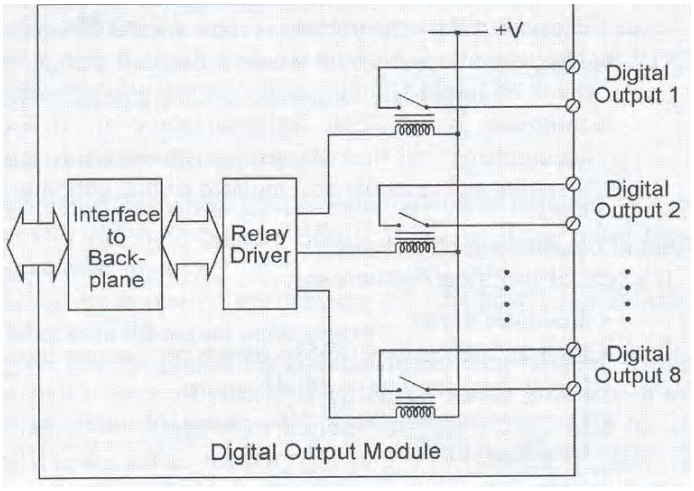

Figure 1.14. Digital output module

'Dry' relay contacts (i.e. no voltage applied to the contacts by the output module) are often provided. These could be reed relay outputs for example. Ensure that the current rating is not exceeded for these devices (especially the inductive current). Although each digital output could be rated at 2 Amps; the module as a whole cannot supply 16 Amps (8 by 2 Amps each) and there is normally a maximum current rating for the module of typically 60% of the number of outputs multiplied by the maximum current per output. If this total current is exceeded there will be overheating of the module and eventual failure.
Note also the difference between sinking and sourcing of an I/O module. If a module sinks a specified current, it means that it draws this current from an external source. If a module sources a specific current it drives this current as an output (see Figure 1.15).

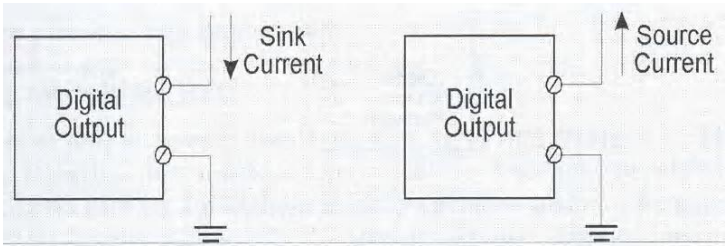

Figure 1.15. Source and sink of current

When connecting to inductive loads it is a good suggestion to put a flywheel diode across the relay for DC systems and a capacitor/resistor combination for AC systems as indicated in Figure 1.16. This minimizes the back EMF effect for DC voltages with consequent voltage spikes when the devices are switched off.

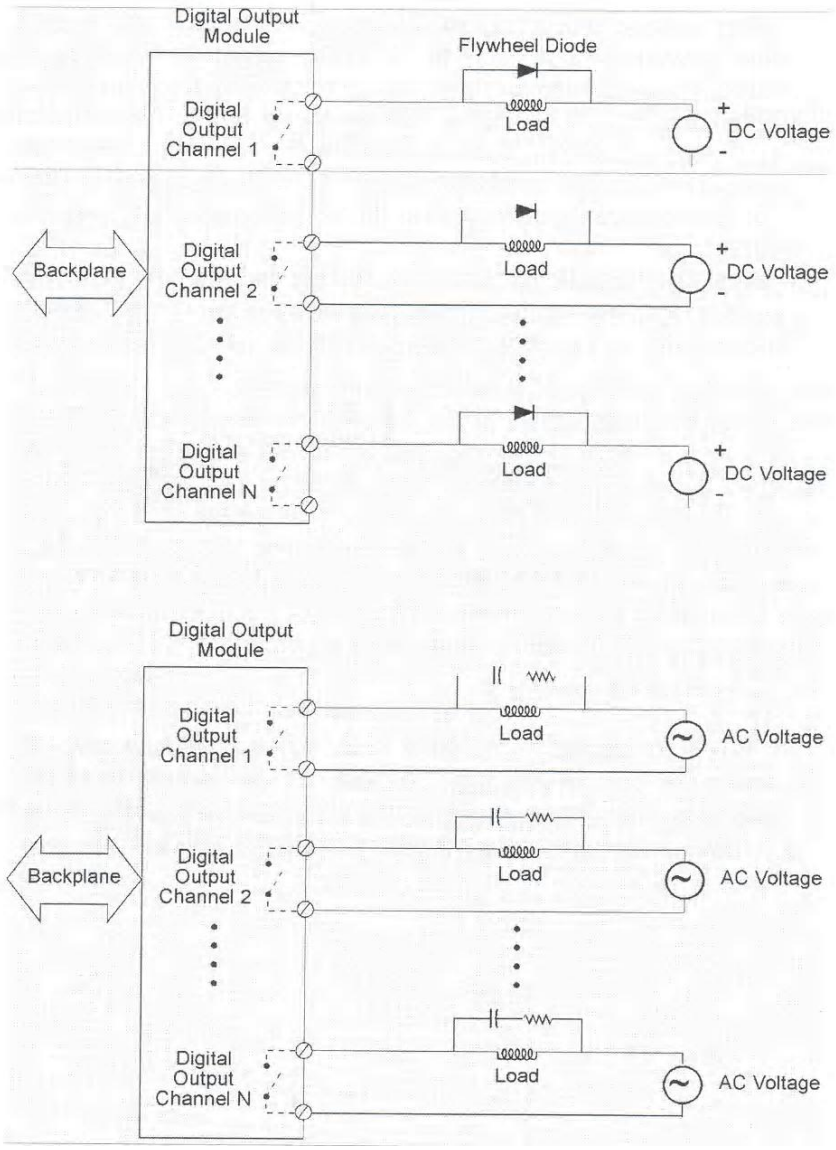

Figure 1.16. Flywheel Diodes or RC Circuits for Digital Outputs

1.5.1. Mixed analogue and digital modules

As many RTUs have only modest requirements as for as the analogue and digital signals are concerned, a typical solution would be to use a mixed analogue and digital module. This would typically have:

- 4 analogue inputs (8 bit resolution)
- 2 digital inputs
- 1 digital output
- 2 analogue output (8 bit resolution)

1.5.2. Communication interfaces

The modern RTU should be flexible enough to handle multiple communication media such as:

- RS-232/RS-442/RS-485
- Dialup telephone lines/dedicated landlines
- Microwave/MUX
- Satellite
- X.25 packet protocols
- Radio via trunked/VHF /UHF /900 MHz

Interestingly enough, the more challenging design for RTUs is the radio communication interface. The landline interface is considered to be an easier design problem. These standards will be discussed in a later section.

1.5.3. Power supply module for RTU

The RTU should be able to operate from 110/240 VAC ±10% 50 Hz or 12/24/48 VDC ± 10% typically. Batteries that should be provided are lead acid or nickel cadmium. Typical requirements here are for 20-hour standby operation and a recharging time of 12 hours for a fully discharged battery at 25°C. Associated power supplies such as solar (e.g. 12 VDC supply) and generators will be discussed separately in 'Practical radio telemetry'.
The power supply, battery and associated charger are normally contained in the RTU housing. Other important monitoring parameters, which should be transmitted back to the central site/master station, are:

- Analogue battery reading
- Alarm for battery voltage outside normal range

Cabinets for batteries are normally rated to **IP** 52 for internal mounting and **IP** 56 for external mounting.

1.5.4. RTU environmental enclosures

Typically, the printed circuit boards are plugged into a back plane in the RTU cabinet. The RTU cabinet usually accommodates inside an environmental enclosure which protects it from extremes of temperature / weather etc.
Typical considerations in the installations are:

- *Circulating air fans and filters* should be installed at the base of the RTU enclosure to avoid heat build-up. Hot spot areas on the electronic circuit should be avoided by uniform air circulation. It is important to have a heat soak test .
- *Hazardous areas* RTUs must be installed in explosion proof enclosures (e.g. oil and gas environment)

Typical operating temperatures of RTU's are too variable when the RTU is located outside the building in a weather proof enclosure. These temperature specifications can be relaxed if the RTU is situated inside a building where the temperature variations are not as extreme (provided of course consideration is given to the situation where there may be failure of the ventilators or air-conditioning systems).
Typical humidity ranges are 10-95%. Ensure at the high humidity level that there is no possibility of condensation or the circuit boards or there may be contact corrosion or short-circuiting. Lacquering of the printed circuit boards may be an option in these cases.

Be aware of the other extreme where, low humidity air (5%) can generate static electricity on the circuit boards due to stray capacitance. CMOS based electronics is particularly susceptible to problems in these circumstances. Screening and grounding the affected electronic areas can only reduce static voltages. All maintenance personnel should wear a ground strap on the wrist.
If excessive electromagnetic interference (EMI) and radio frequency interference (RFI) is anticipated in the vicinity of the RTU, special screening and earthing should be used.
Some manufacturers warn against using handheld transceivers in the neighborhood of their RTUs.
Continuous vibration from the vibrating plant and equipment can also have an unfavorable impact on a RTU, in some cases. Vibration shock mounts should be specified for such RTUs.
Other areas, which should be considered with RTUs are lightning (or protection from electrical surges) and earthquakes (which is equivalent to vibrations at frequencies of 0.1 to 10 Hz).

1.5.5. Testing and maintenance

Many manufacturers provide a test box to test the communications between the RTU and master stations; and also to simulate a master station or RTU in the system. The three typical configurations are indicated below in Figure 1.17.

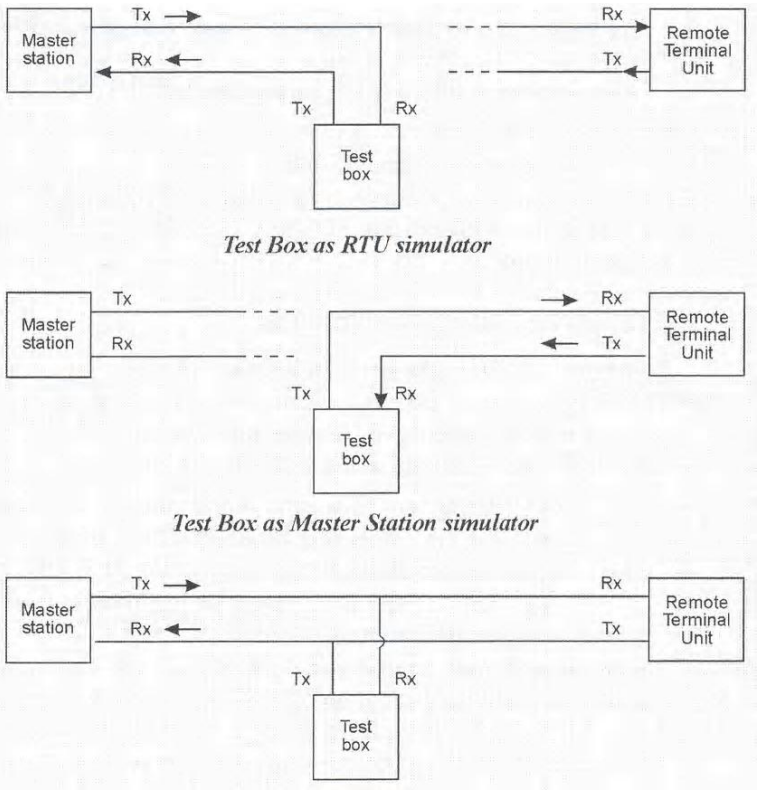

Figure 1.17. SCADA test box operating mode

The typical function provided on a test box is:

- *Message switches:* The simulated messages that the user wants to send to the RTU or master station are input here.
- *Message indicators:* Display of transmit and receiver data.
- *Mode of operation:* The user selects one of the three modes of operation. Test box in eavesdropping mode between RTU and master station; test box to RTU; test box to master station. An additional self-test mode is often provided.

There are other features provided such as continuous transmissions of preset messages. Often the test box is interfaced to a PC for easier display and control of actions.

1.5.6. Typical requirements for an RTU system

In the writing of a specification, the following issues should be considered:

Hardware

Individual RTU expandability (typically up to 200 analogue and digital points)

- Off the shelf modules
- Maximum number of RTU sites in a system shall be expandable to 255.
- Modular system - no particular order or position in installation (of modules in a rack)
- Robust operation - failure of one module will not affect the performance of other modules
- Minimization of power consumption (CMOS can be an advantage)
- Heat generation minimized
- Rugged and of robust physical construction
- Maximization of noise immunity (due to harsh environment)
- Temperature of -10 to 65°C (operational conditions)
- Relative humidity up to 90%
- Clear indication of diagnostics
- Visible status LEDs
- Local fault diagnosis possible
- Remote fault diagnostics option
- Status of each I/O module and channel (program running/ failed /communications OK / failed)
- All modules connected to one common bus
- Physical interconnection of modules to the bus shall be robust and suitable for use in harsh environments
- Ease of installation of field wiring
- Ease of module replacement
- Removable screw terminals for disconnection and reconnection of wiring

Environmental considerations

The RTU is normally installed in a remote location with fairly harsh environmental conditions. It typically is specified for the following conditions:

- Ambient temperature range of 0 to +60°C (but specifications of -30°C to 60°C are not uncommon)
- Storage temperature range of -20°C to +70°C
- Relative humidity of 0 to 95% non condensing
- Surge withstand capability to withstand power surges typically 2.5 kV, 1 MHz, for 2 seconds with 150 Ohm source impedance
- Static discharge test where 1.5 cm sparks are discharged at a distance of 30 cm from the unit
- Other requirements include dust, vibration, rain, salt and fog protection.

Software (and firmware)

- Compatibility checks of software configuration on hardware against actual hardware available.
- Log kept of all errors that occur in the system both from external events and internal faults
- Remote access of all error logs and status registers
- Software operates continuously despite power on or off, due to loss of power supply or other faults
- Hardware filtering provided on all analogue input channels
- Application program resides in non volatile RAM
- Configuration and diagnostic tools for:
    - System setup
    - Hardware and software setup
    - Application code development/management/operation
    - Error logs
    - Remote and local operation

Each module should have an internal software continuously testing the systems I/O and hardware. Diagnostic LEDs should also be provided to identify any faults or to diagnose failure of components. It is important that all these conditions are communicated back to the central station for indication to the operator.

1.5.7. Application programs

Many applications, which were previously performed at the master station, can now be performed at the RTU, due to improved processing power and memory/dish storage facilities available. Many RTUs also have a local operator interface provided.

Typical application programs that can run in the RTU are:

- Analogue loop control (e.g. PID)
- Meter proving
- (Gas) flow measurement
- Compressor surge control

1.6. PLCs used as RTUs

A PLC or Programmable logic controller is a computer based solid-state device that controls industrial equipment and processes. It was initially designed to perform the logic functions executed by relays, drum switches and mechanical timer/counters. Analogue control is now a standard part of the PLC operation as well.
The advantage of PLC over RTU offerings from the various manufacturers is that, it can be used in a general-purpose role and can easily be set up for a variety of different functions.
The actual construction of a PLC can vary widely and does not necessarily differ much from generalizing on the discussion of the standard RTU.
PLCs are popular for the following reasons:

- Economic solution: PLCs are greatly more economic solutions than a hardwired relay solution and a short run manufactured RTU's.
- Versatility and flexibility: PLCs can easily have their logic or hardware modified to cope with modified requirements for control.
- Ease of design and installation: PLCs have made the design and installation of SCADA systems easier because of the emphasis on software.
- More reliable: When correctly installed, PLCs are a far more reliable solution than a traditional hardwired relay solution or short run manufactured RTUs.
- Sophisticated control: PLCs allow for far more sophisticated control (mainly due to the software capability) than RTUs.
- Physically compact: PLCs take up far less space than alternative solutions.
- Easier troubleshooting and diagnostics: Software and clear cut reporting of problems allows easy and swift diagnosis of hardware/firmware/software problems on the system as well as identifying problems with the process and automation system.

A diagram of a PLC and its means of operation using standard ladder logic are discussed in the following section.

1.6.1. PLC software

The ladder logic approach to programming is popular because of its perceived similarity to standard electrical circuits. Ladder logic is one component of the IEC 61131-3 programming standard. Two vertical lines supplying power are drawn at each of the sides of the diagram with the lines of logic drawn in horizontal lines.
Figure 1.18 shows the 'real world' circuit with PLC acting as the control device and the internal ladder logic within the PLC.

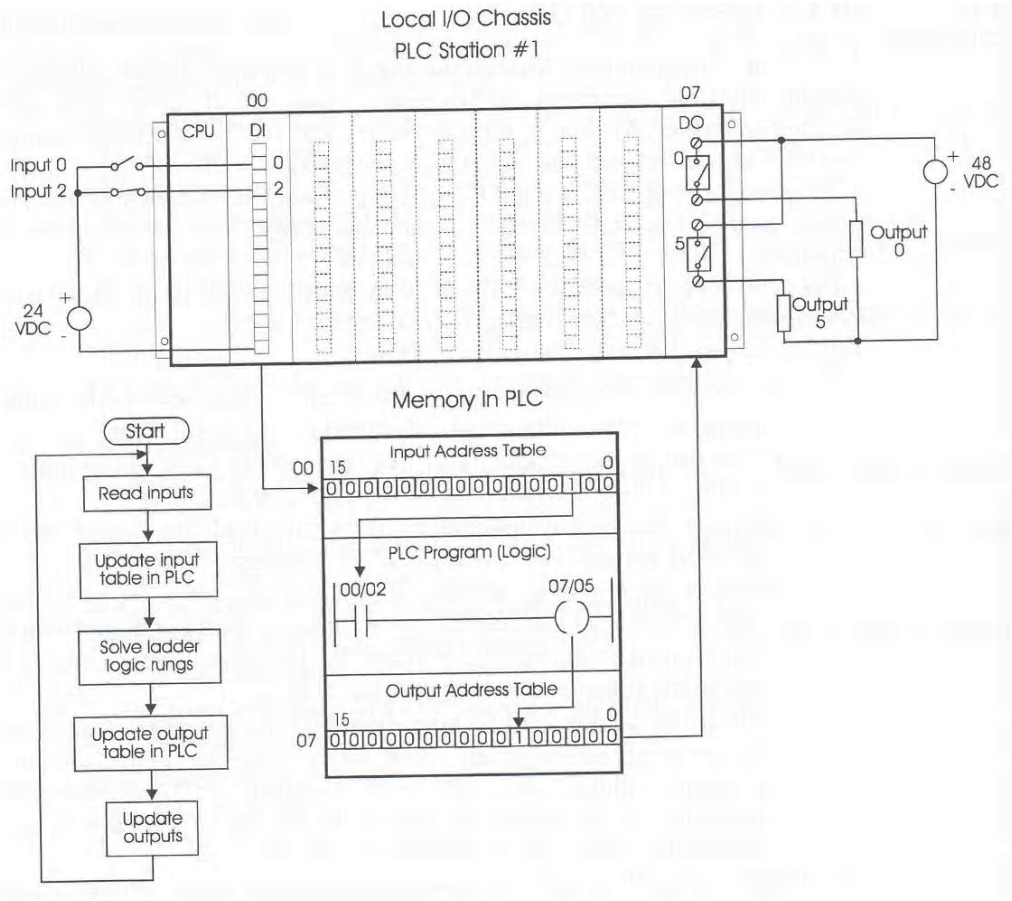

Figure 1.18. The concept of PLC ladder logic

Basic rules of ladder logic

The basic rules of ladder logic are stated as follows:

- The vertical lines indicate the power supply for the control system (12 VDC to 240 V AC). The 'power flow' is visualized to move from left to right.
- Read the ladder diagram from left to right and top to bottom (as in the normal Western convention of reading a book).
- Electrical devices are normally indicated in their normal de-energized condition. This can sometimes be confusing and special care needs to be taken to ensure consistency.
- The contacts associated with coils, timers, counters and other instructions have the same numbering convention as their control device.
- Devices that indicate a start operation for a particular item are normally wired in parallel (so that any of them can start or switch the particular item on). See Figure 1.19 for an example of this.

- Devices that indicate a stop operation for a particular item are normally wired in series (so that any of them can stop or switch the particular items off). See Figure 1.20 for an example of this.

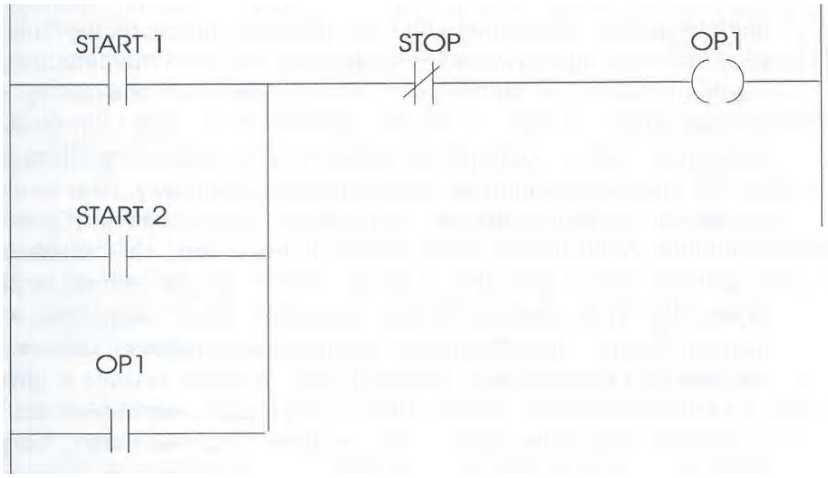

Figure 1.19. Ladder logic start operation (& logic diagram)

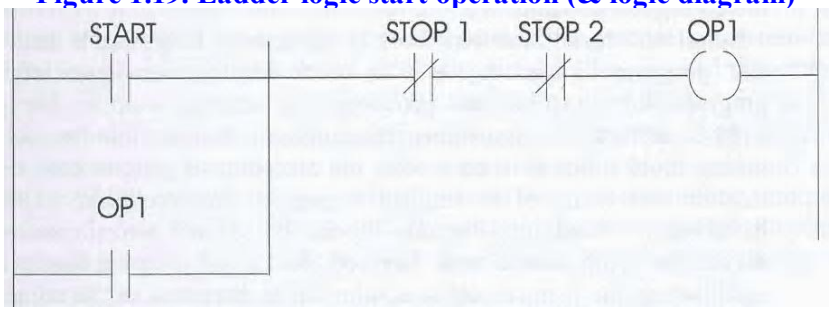

Figure 1.20. Ladder logic stop operation (& logic diagram)

- Latching operations are used where a momentary start input signal latches the start signal into the on condition; so that when the start input goes into the OFF condition, the start signal remains energized ON. The latching operation is also referred to as holding or maintaining a sealing contact. See the previous two diagrams for examples of latching.
- Interactive logic. Ladder logic rungs that appear later in the program often interact with the earlier ladder logic rungs. This useful feed back mechanism can be used to provide feedback on successful completion of a sequence of operations (or protect the overall system due to failure of some aspect).

1.6.2. Implementing IEC 1131-3 in a comprehensive process control strategy

IEC 1131-3, ISA SP-88, describes the open architecture, distributed control system, programmable logic controller, hierarchical design, and batch process control.

In the past, the development of a process control strategy has often involved implementation of a proprietary or advanced programming language. With no prior knowledge of the programming product, rigorous training and familiarization was required to become proficient in its use. The time and money associated with such training often led to reliance on specific

individuals for future enhancements and maintenance of the configuration software. The inefficiency that results from this practice can be witnessed throughout the process control industry.

Another possible area of inefficiency stems from the need to combine both continuous control and discrete control in one system. The centralized, continuous control of a distributed control system (DCS) is often merged with the localized, discrete control offered by the programmable logic controller (PLC). Sometimes a simple integration is difficult due to the difference in configuration languages and communications protocol between the two systems.

The implementation of IEC 1131-3 in process control provides, an extremely simple and efficient solution to the two scenarios described above, while providing an open architecture for future software development considerations at the same time. As with any standard, IEC 1131 establishes a common foundation on which to build by providing a comprehensive set of programming tools, techniques and terminology. The most prominent component of the standard is part 3, which defines the programming languages of the standard. Three languages are graphical in nature, while the remaining two languages are presented in a textual format. The languages are described below.
They empower the programmer to mix and match languages as deemed necessary by the configuration task.

Function block: Function block is a graphical language. It includes standard, derived, and program blocks. A function block has one or more inputs acted upon by a programmed algorithm and produces one or more outputs. For standard blocks, IEC 1131-3 defines the algorithms. Examples of standard blocks include math, logic, and timing blocks. For derived blocks, the algorithm is programmed by the user and can be implemented in any of the languages specified. In some instances, derived blocks may not have inputs or outputs. Program blocks that do not have inputs or outputs are used to divide the configuration task. Derived blocks and program blocks are the tools used for establishing the hierarchical structure that is discussed in the section designing a control strategy.
Instruction lists: Instruction list is a low-level language similar to assembly language. It is used for fast and efficient applications. One operation may take a group of instruction lists statements.
Ladder logic: Ladder logic is a graphical language. It is identical to the type of logic implemented by most programmable logic controllers. It is composed of a set of standard relay symbols that simulate traditional relay logic. Ladder logic is a key ingredient to the seamless integration of continuous and discrete control in one controller.
Sequential function chart: The sequential function chart (SFC) language is a graphical language. It is a flow chart method developed for batch processors and automated start-up and shutdown procedures. Its format provides easy implementation of the batch process control standards set forth in ISA SP-88.

Four configuration elements are specified. They are:

- Step: A point specifying execution of an action.
- Transition: The condition following each step that controls progression of the chart.
- Action: The operation performed when its associated step is active. Actions can be programmed in any of the specified languages. It also should be noted that an action could be driven by any Boolean output of a function block.
- Branch: Allows for divergence and convergence of a chart to proceed along parallel paths; also provides looping capability.

Structured text: Structured text is a high-level language very similar to the PASCAL programming language. It provides standard comparison statements and iterative looping. The structured text language is useful for performing complex mathematical computations that would be cumbersome and confusing if implemented using standard math function blocks. When combined with the sequential function chart language, it provides an ideal tool for communicating information to an operator.

Designing a control strategy with program and derived function blocks: Control schemes are divided into logical areas of functionality. A hierarchy of operation can be implemented that not only simplifies the configuration process, but also provides convenience for future additions or maintenance. The hierarchy is established by the method in which the blocks are added to the configuration. The configuration editor has a format similar to a spreadsheet. There are labeled columns and rows, which define cells on the spreadsheet. When a function block is placed on the spreadsheet, it occupies a number of cells. When the function block is a program or derived block, a connection is made to a new layer, or sheet, which is at a lower level in the hierarchical structure.

Determining the appropriate configuration tools: Program blocks organize the process into logical divisions. Each program block's contents, start in one of the specified programming languages. Ladder logic could be used if the process is discrete-intensive. For most DCS applications, the function block diagram language is used to define process loops and motors. A continuous analogue PID control function block, with its inputs on the left side and outputs on the right, is defined as another function block diagram sheet. That sheet resides one level below the PID control function block and will contain a 'SETPOINT,' 'PID,' 'ALARM,' and 'AUTO_MANUAL' function block, unique to that loop. A motor or on/off valve function block can be defined as another function block diagram or ladder logic diagram sheet. That sheet would layout the necessary contacts and timers for proper motor or on/off valve control.

Implementing the control strategy: For implementing the control strategy, piping and instrument drawings were the main design documents. The first step was to break the process down into separate areas. Those areas were established as individual program blocks by the conditions set in the ISA SP-88 standard. A programming language was then selected for each program task. Function block language was used to define all motors and loops on the project. Analogue inputs, alarms, and discrete alarms were also on the function block sheets. Structured text language was used for automatic/ manual control and output to control valves. It was also used as the initialization step for sequential function charts. Operators were notified through structured text configuration. Ladder logic language was used for motor and on/off valve control. It was also used to implement vendor supplied ladder logic, which was provided to control their trash compactor equipment. This resulted in saving engineering time, because the control logic supplied had already been tested and implemented at other sites. Sequential function charts were used for batch sequencing control, start-up, and shutdown logic. The start-up and/or shutdown logic was started by operator control or process emergency conditions.

IEC 1131-3 software standard provides one integrated platform for discrete and regulatory control. Control of the process is optimized to the appropriate programming language. Vendor's software can be included into the configuration with minimal debugging and checkout. Standard guidelines for configuration documentation will result in easier software modifications in the future.

1.7. System reliability and availability

The individual component of the SCADA system contributed to the overall reliability of the system. As the master station is a strategic part of the entire SCADA system, it is important that the system reliability and availability is carefully considered. Loss of a single RTU, although unpleasant, should still allow the system to continue to function as before.
Master station components that are critical are:

- Control processing unit (CPU)
- Main memory and buffer reprinters
- Dish drive and associated controller card
- Communications interface and channel

1.7.1. Redundant master station configuration

There are various redundant configurations possible. Two approaches possible are shown in Figure 1.21. The simplest approach is to have a cold standby changeover, where a switch is generated to change over from primary to secondary.

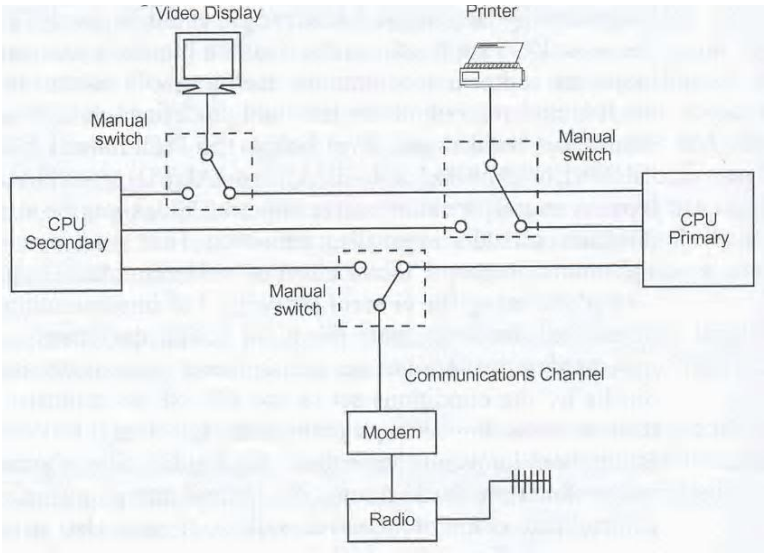

Figure 1.21. Cold standby SCADA system

An example of hot standby configuration is given in Figure 1.22. Here a watchdog timer (WDT) is activated if the primary CPU does not update or reset it within a given time period. Once the WDT is activated, a changeover is affected from the primary to the secondary CPU system. Due to the use of continuous high speed memory updating of the secondary CPU's memory, the secondary or backup CPU contains all the latest status data (until the WDT activated the changeover).

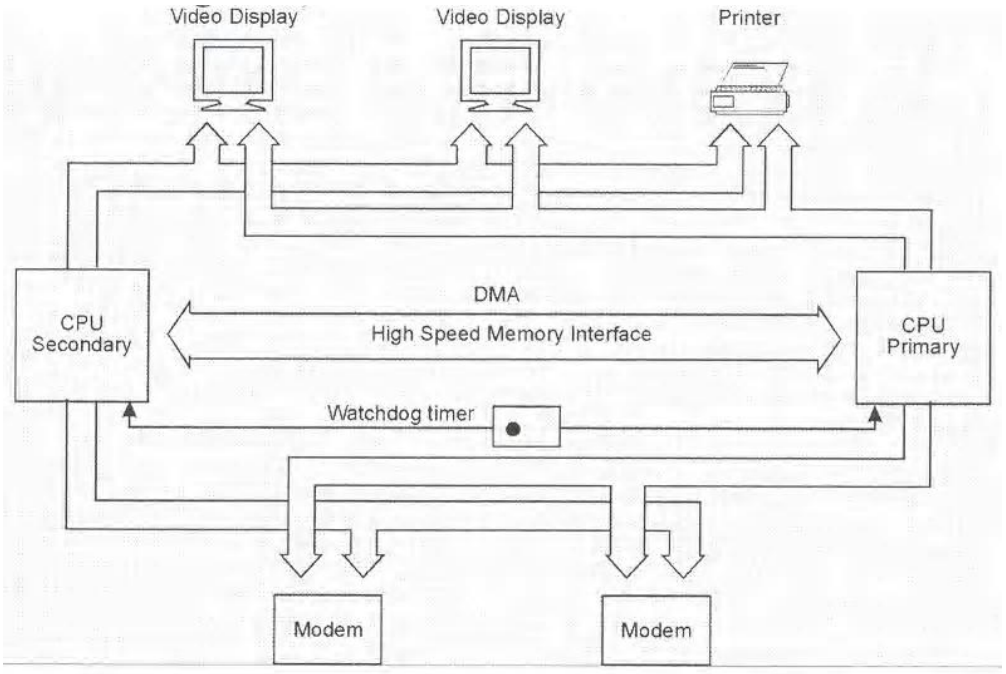

Figure 1.22. Dual Ported Peripheral