UNIT-V
ALARMS

An alarm is defined as an indication of an abnormal process condition. An alarm annunciator is a device which signals the presence of abnormal process conditions using a visual display usually supplemented with an audible warning such as buzzer or siren.

* **Alarm** is needed to issue warning when some process variable exceeds a critical value.

* **Annunciator** is a device or group of devices that call attention to changes in process condition that have occurred.

* **Interlock** is a feature that makes the state of two mechanisms or functions mutually dependent. It may be any electrical, electronic, or mechanical devices or systems. In most applications, an interlock is used to help to prevent a machine from harming its operator or damaging itself by preventing one element from changing state. It is a method of ensuring safety in industrial environments by forcing the operator through a predetermined sequence using a defined selection of keys, locks and switches.

Special demands of an alarm system.

1) Problems that need operator attention
2) Process changes that require corrective action
3) Unsafe operating conditions before Emergency Shut-down of the plant
4) Hazardous conditions
5) Deviations from desired/normal conditions

Single Variable Alarms:

The simplest alarms are those involving only a single variable. For this type of alarm a simple implementation is to convert the measurement signal to a voltage and use a digital comparator. In general we want to use a comparator with hysteresis where the comparator output switches between 1and 0 repeatedly during transition. This protects against noise on the signals and nonlinear effects in the comparator itself.

Multivariable Alarms:

In many cases an alarm is dependent not only upon the value of a single variable but also on the combined values of one or more variables. Thus, for example, a tank with high pressure and high level may represent some unpleasant condition for which an alarm should be issued. In these cases we can use comparators combined with appropriate logic circuits to produce an alarm. In general, hysteresis should again be used to protect against noise and nonlinear effects.

Problem:1

Design an alarm that provides a logic high of 5 V when a liquid level exceeds 4.2 meters. The level has been linearly converted to a 0–10 volt signal for a 0–5 meter level. Hysteresis should be 0.1 volts.

<u>Solution</u>

For this type of alarm a simple implementation is to convert the measurement signal to a voltage and use a digital comparator. In general we want to use a comparator with hysteresis wherein the comparator output switches between 1and 0 repeatedly during transition.

Input level range = 0 to 5m; Output voltage change = 0 to 10V

The level L and voltage output V are related by the relation , V = 2L.

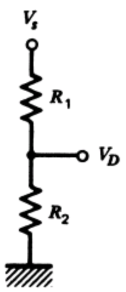
Thus, for L = 4.2 m the output is V=2 x 4.2m = 8.4 V.

The level of 4.2 meters corresponds to 8.4 V

Hysteresis = 0.1 V ; Hysteresis = $(R/R_f)\ V_o$

$0.1 = (R/R_f)\ 5$; $R/R_f = 0.1/5 = 0.02$; Let R= 1 kΩ; $R_f = 50$ kΩ

For providing 8.4 V reference, We can use a voltage divider circuit



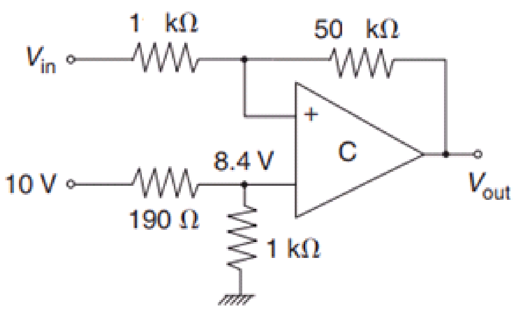
$$V_D = \frac{R_2 V_s}{R_1 + R_2}$$

where V_s = supply voltage
R_1, R_2 = divider resistors

If V_s= 10V, V_D = 8.4V, V_D/V_S= 0.84 = $R_2/(R_1+R_2)$

Let R_2 = 1 kΩ, 0.84 (R_1+1000) = 1000, 0.84 R_1 + 840 = 1000

0.84R_1=1000-840 = 160, R_1=160/0.84 = 190.47Ω

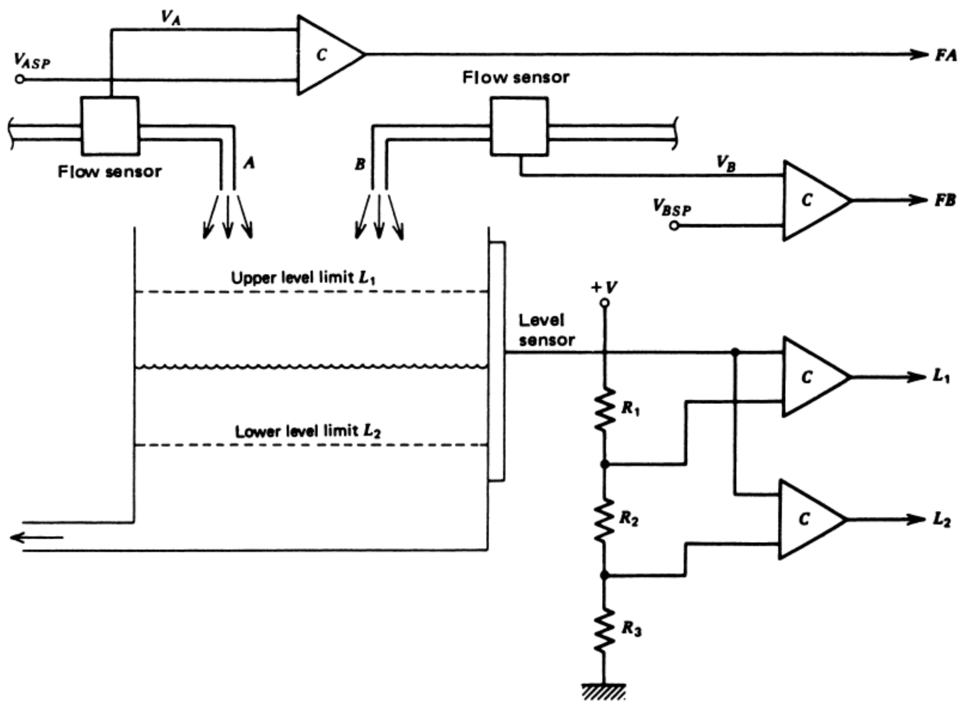


Problem:2

Consider the following figure, a holding tank for which liquid level, inflow A, and inflow B are monitored. These measurements are converted to voltages and then, with comparators, to digital signals that are HIGH when some limit is exceeded. The flow variables FA and FB will be 0 for low flow and 1 for high flow. The level variables are such that L_1 is 1 if the level exceeds the lower limit and L_2 will be 1 if the level exceeds the upper limit. The alarm will be triggered if either of the following conditions occurs:

1. LOW and neither FA nor FB HIGH
2. HIGH and FA or FB or both HIGH

Implement this problem with digital logic circuits.



<u>Solution</u>

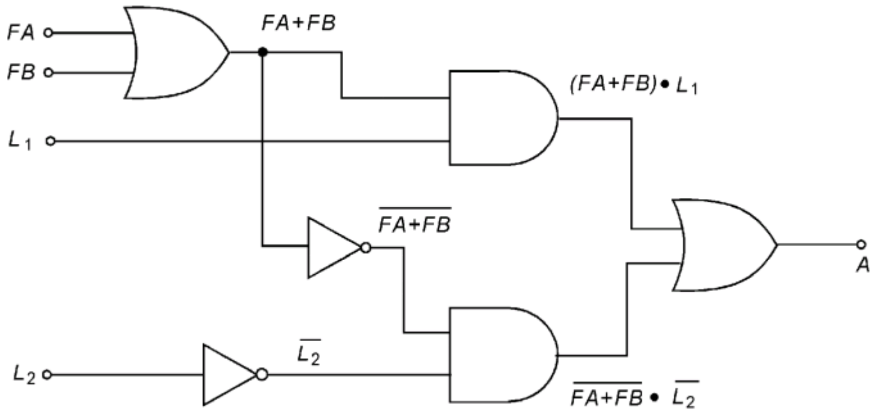
The variables FA, FB L_1 and L_2 are Boolean in that they have values of logic 0 or 1. We can write Boolean equations giving an alarm output for the given two conditions. This can be done directly as

1. $A = \overline{L_2} \cdot (\overline{FA + FB})$

2. $A = L_1 \cdot (FA + FB)$

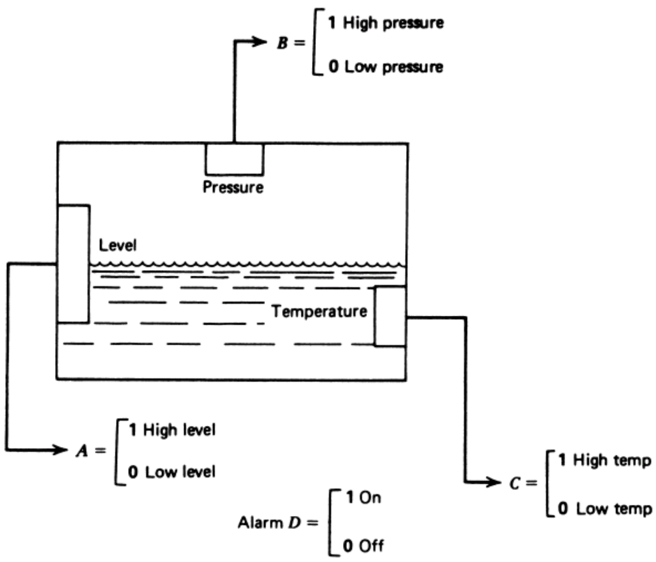
Now either of these conditions is provided by an OR operation:

Logic gates that can be used to directly implement this equation are shown in Figure.



Problem:3

Consider a mixing tank shown below, for which there are three variables of interest: liquid level, pressure, and temperature. Design an alarm when the following combinations of conditions occur among these variables.



<u>Solution:</u>

Referring to Figure, we denote level by A, pressure by B, and temperature by C, and assume that setpoint values have been assigned for each variable so that the Boolean variables are either 1 or 0 as the physical quantities are above or below the setpoint values. The alarm will be triggered when the Boolean variable D goes to the logic true state.

The alarm conditions are,

1. Low level with high pressure
2. High level with high temperature
3. High level with low temperature and high pressure

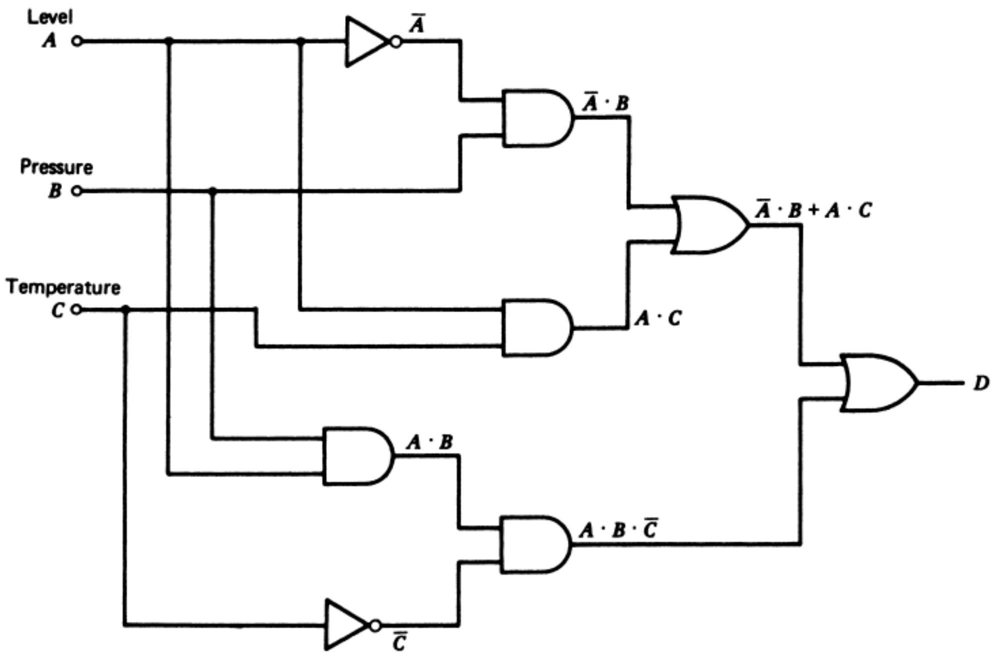
We now define a Boolean expression with AND operations for each condition:

1. $D = \overline{A} \cdot B$ will give D=1 for condition 1
2. $D = A \cdot C$ will give D=1 for condition 2
3. $D = A \cdot \overline{C} \cdot B$ will give D=1 for condition 3

The final logic equation results from combining all three conditions so that if any is true, the alarm will sound . This is accomplished with the OR operation

$D = \overline{A} \cdot B + A \cdot C + A \cdot \overline{C}.B$

The implementation of this equation using AND/OR gates is shown in Figure



Problem:4

A process-control system specifies that temperature should never exceed 160^{o}C if the pressure also exceeds 10 kPa. Design an alarm system to detect this condition, using temperature and pressure transducers with transfer functions of 2.2mV/^{o}C and 0.2 V/kPa, respectively.

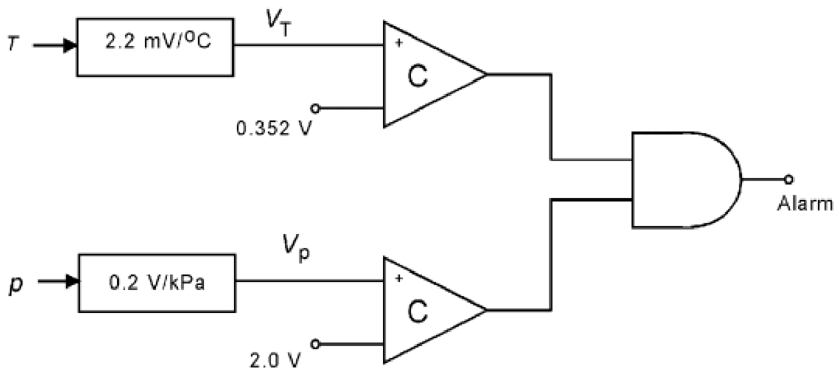
Solution:

The alarm conditions will be

A temperature signal of (2.2 mV°C) (160°C) = 0.352 V

and a pressure signal of (0.2 VkPa) (10 kPa) = 2 V

The circuit is shown in the Figure



Problem: 5

Test for an alarm condition for a process which involves moving speed and rate of loading in a conveyer system.

(i) Speed is low, both weight and loading rate are high

(ii) Speed is high and weight and loading rate are low.

Find the Boolean Equation describing the required alarm output and draw the logic diagram.

Solution:

Let us assume S= speed

W= weight

R = Loading rate

The alarm will be triggered if either of the following conditions occur.

(i) S is low, both W and R are high

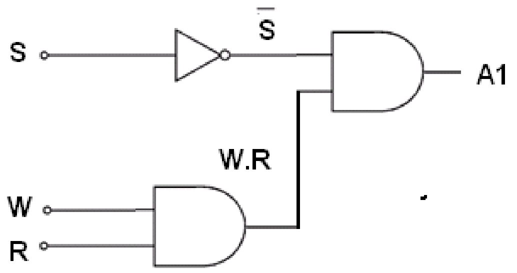
(ii) S is high and W and R rate are low.

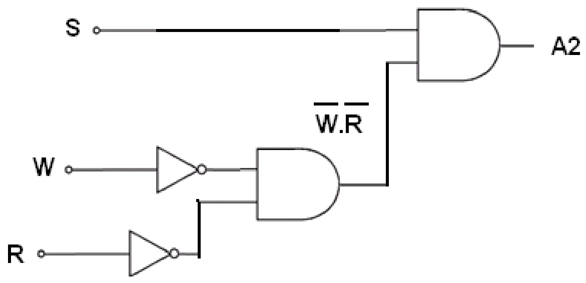
The Boolean equations giving an alarm output A_1= 1 and A_2 =1 for the given two conditions. This can be done directly as

$$A_1 = \overline{S}.(W.R)$$

$$A_2 = S.(\overline{W}.\overline{R})$$

The logic gate solution:



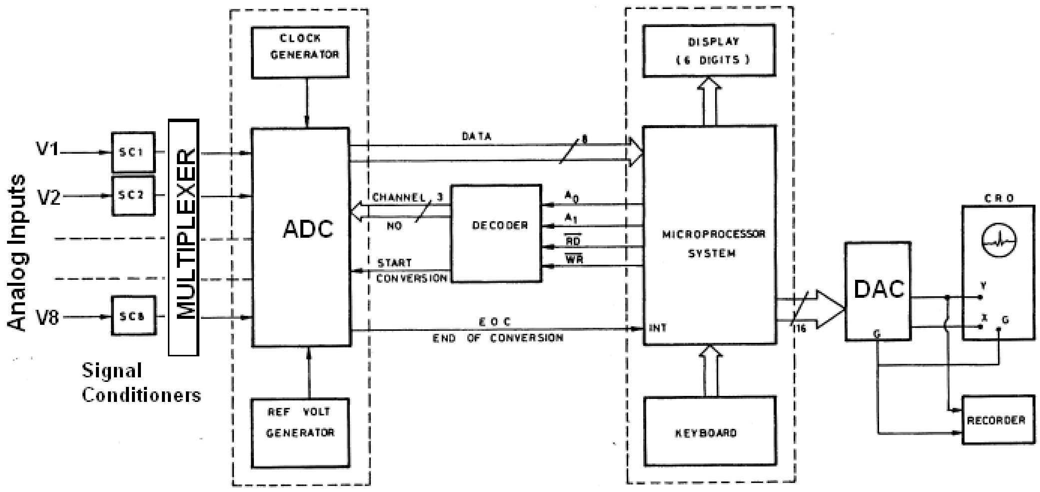


MICROPROCESSOR BASED DATA ACQUISITION SYSTEM (DAS)

Microprocessor based equipments generally offer benefits like higher reliability, more versatility, lesser hardware cost and lower power consumption. The microprocessor based DAS consists of a multiplexer, an analog to digital converter and a microprocessor system. The microprocessor technology has made it possible to realise stand-alone instruments capable of performing real time analysis of signals during the data acquisition time itself.

THE DAS HARDWARE:

The block diagram representation of the DAS is shown in Figure.



A Typical Data Acquisition System

The major blocks of a Data Acquisition System are

(i) Sensor (which give analog inputs, V1,V2,...)
(ii) Signal conditioner
(iii) Multiplexer
(iv) Analog to Digital Converter (ADC)
(v) Microprocessor
(vi) Dogital to Analog converter
(vii) Display devices

The analog input is sensed by the sensors converted in to require voltage signals using signal conditioning circuits and given to multiplexer.

The multiplexer (MUX) is a switch that sequentially scans numerous input signal channels and directs them in a pre programmed manner to the ADC for digitizing. This saves the cost of using an ADC for each channel.

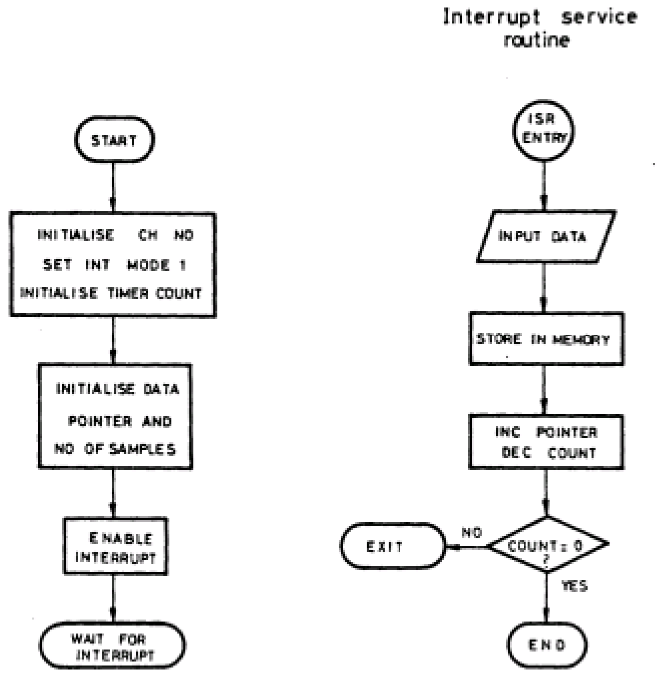
The input signals are amplified and filtered and fed to a mutichannel Analog to Digital Converter (ADC) built around the single chip ADC. The conversion time of this ADC chip may vary between 10 to 1000 microseconds which is fast enough to acquire signals. The external clock required for the counter can be generated by a voltage controlled oscillator IC. The voltage regulator is used to provide the reference voltage required for the ADC. The tri-state buffered output of the ADC enables direct transfer of the converted data to the data bus of the microprocessor system. The ADC requires a start conversion (SC) command for initiating the conversion process. A pulse is generated at the end of each conversion, which is connected to the interrupt input of the microprocessor.

The system operates with a clock and the memory consists of ROM and RAM. A decoder is used to select the channel and generate the start conversion command. Other facilities incorporated in the system include a keyboard, display controller, a programmable interval timer and a programmable peripheral interface.

The DAS periodically acquires the input data at a sampling rate specified through the software. Provision can be made to vary the sampling frequency and the acquisition time.

Data Acquisition Flowchart:

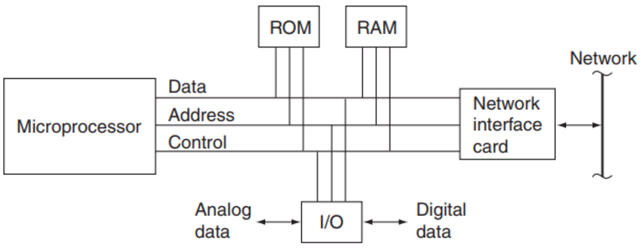
The flow chart for the data acquisition function is shown in Figure. The channel number and the number of samples are initialised along with the starting address of data. The interrupt mode is set and then the interrupt is enabled. In the interrupt service routine, date available from the ADC is moved on to the memory after checking the end of conversion pulse. Then the memory address is incremented and the sample count is decremented. After one sample interval the interrupt is enabled for storing the next sample. The process is continued until the sample count becomes zero.



Flow chart for the data acquisition

Any microprocessor based data acquisition system has a standard array of devices and are available as small integrated circuits that can be mounted on a relatively small printed circuit board (PCB). The ROM stands for read-only-memory, non volatile memory that holds the programs that the processor executes. Typically the ROM consists in part of electrically erasable memory so that programming updates can be down loaded over the communication network. The RAM stands for read write memory and is used to hold the transient results of calculations and other results of data processing. The data I/O typically consists of ADCs and DACs as well as digital I/O channels.

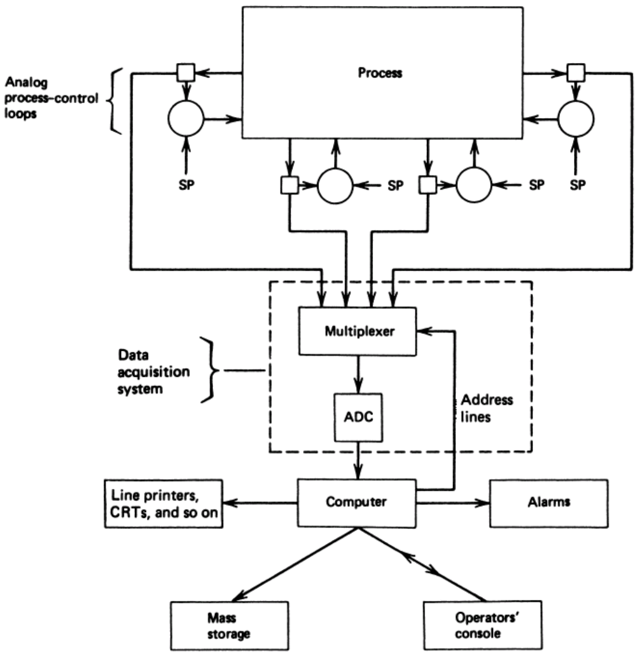
Microprocessor based Controller:



Basic structure of a microprocessor-based Control

The use of a microprocessor as the controller means it must solve the control equations. If microprocessor control is implemented, external commands can be used to select the desired mode (PI, PID, etc.) and the gains for each mode. The microprocessor accepts an input value from an ADC or over the bus from the sensor. The value will have been encoded as a binary number. In describing the algorithms, the measurement range of the controlled variable is known. For the purposes of algorithm description, the variable has to be converted from a binary encoding to its actual value as a floating-point variable (pressure, temperature, etc.) in the control program. This may entail linearization as well as scaling. In the program the error will be used as a fractional quantity rather than a percent.

The general features of a computer data logging system are shown in Figure.



General features of a data-logging system using a computer.

Computers are used for many purposes in a process control industries. They are also for engineering design, financial analysis and plant operations managements. With the development of high speed digital computers with mass digital storage, it became possible to record data continuously and automatically, display the data on command, and perform calculations on the data.

Problem:

Analyse the given equation Volage V=(3.1[p+10]$^{1/2}$-9.8) , where p is the pressure, develop the equations used to find the pressure from the binary input and also the error. The pressure range is 0-30 psi and set point is 15psi. This voltage is provided as input to an 8 bit unipolar ADC with 10V reference. Contrast the actual error with the computed error for a pressure of 17.3 psi

Solution:

Let us call the sample from the ADC as N_i, which is the base 10 equivalent of the binary output of the ADC. We can then find the voltage corresponding to this sample (within the V of the ADC) as,

$$V_i=(10/256)\ N_i$$

Now the pressure sample can be determined by using the given equation relating pressure and voltage as

$$p_i = \left(\frac{V_i + 9.8}{3.1}\right)^2 - 10$$

Then, the error as a fraction of range is $e_i = \left(\frac{p_i + 15}{30}\right)$

To find the actual error for 17.3 psi we should use the previous equation with p = 17.3 psi,

$$e = (17.3 - 15)/30 = 0.077$$

To find the sample error we must take into account the loss in information due to the ADC. Thus, we calculate just as the computer will. The voltage of the sensor is:

$$V_{17.3} = 3.1(17.3 + 10)^{1/2} - 9.8 = 6.397\text{ V.}$$

The output of the ADC will be,

$$N_i = \text{int}\left[\frac{6.397}{10}(256)\right] = 163$$

Problem:

Interpret the maximum sampling rate, when a data logging system monitors 12 loops. A microprocessor require 4μs per instruction and 100 instructions to address a multiplexer line. The multiplexer requires to select 20 μs and capture the value of an input line. The ADC conversion time is 30 μs per conversion.

The 100 instructions require a time of 4 μs x 100 = 400 μs

This must be done for 12 loops.

Thus, the total instruction time is 12 x 400 μs= 4800 μs

The ADC conversion time is 30 μs per conversion

For 12 conversions we have 30 μs x 12 = 360 μs,

The total time spent in multiplexer switching is 240 μs.

Adding 4800+360+240= 5400 μs

The maximum sampling rate is the reciprocal of 5400 μs

= 1/5400 μs = 185 samples / sec

MICROPROCESSOR-BASED PID CONTROLLER

There are two approaches to PID control systems:

(i) The analog approach

(ii) The digital approach.

Conventional analog controllers use primarily RC circuits, transistors and operational amplifiers as computing elements. The hardware is simple, inexpensive and reliable; it is easy to operate and easy to service. However, the applications of such controllers are limited to the control of relatively simple loops. With analog controllers, flexibility and expandability are also lacking.

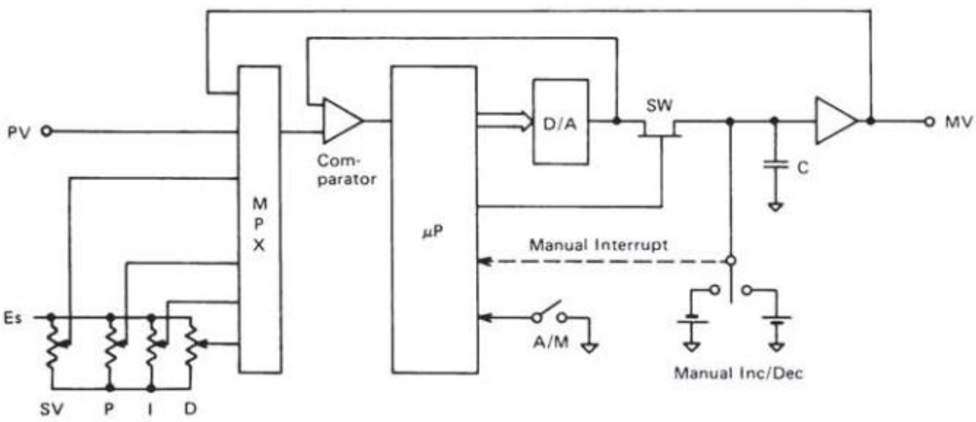
Digital controllers use large computer to supervise many loops using microprocessor based controllers which can handle eight or sixteen loops. The advent of one-chip microprocessors has made single loop controllers possible. This chapter describes the design and applications of microprocessor based PID controllers.

A combined A/D, D/A converter reduces the cost. The converter needs high resolution and moderate speed. The microprocessor incorporates PID control algorithms. They are easy to use. The controllers are easy to interface to supervisory computer systems.

HARDWARE

Typical microprocessor-based systems acquire analog inputs, convert them to digital signals using A/D converters, process the digital signals in the microprocessor, and finally output the results via D/A converters. Inputs are not continuous, but it depends upon the sampling rate and the output is held constant between samples by using sample and hold circuits. If more inputs and outputs are required A/D, D/A converters are multiplexed.

A single IC (DAQ) can be used for both A/D and D/A conversion. A comparator compares the output from the DAQ with an input from the field. The DAQ output is updated periodically; sample and hold circuitry is used so that the output appears continuous. The controller include the setpoint (SP) and the PID controller parameters. The setpoint and parameters are set using potentiometers as in conventional analog controllers. The controller outputs the manipulated variable (MV). The controller reads the process variable and the setpoint and performs PID control. Process Variable (PV) and SP must be read accurately. The PID computation is performed by microprocessor software. In this controller, characteristics of PID action can be easily realized using software. Auto/Manual transfer can also be provided. When the controller is switched from auto to manual mode, The microprocessor then reads the input value and periodically refreshes it via the DAQ. When the controller is switched from manual mode to auto, the PID controller is initialized so that the preceeding manual output is its initial output. Thus, even if there is a deviation between process variable and set point when the controller is transferred from manual to auto mode, transfer is bumpless. The setpoint and PID parameters are set by potentiometers, so the settings are retained even if power fails. The output MV is held by a sample and hold circuit, so if power to the circuit is lost, the output is held



A Microprocessor based PID controller

SOFTWARE

The analog PID control equation cane represented as,

$$u = K_P \left(e + \frac{1}{T_I} \int_0^t e \mathrm{d}t + T_D \frac{\mathrm{d}e}{\mathrm{d}t} \right) + u_0$$

Where, u= controller output

K_p= Proportional gain,

e=Error

T_I= Integral time,

T_D= Derivative time,

u_0= Bias

To get digital PID control algorithm, the integration is replaced by summation and differentiation is replace by backward difference.

$$\int_o^t e(t)\mathrm{d}t \approx T \sum_{j=0}^{k} e_j$$

$$\frac{\mathrm{d}e(t)}{\mathrm{d}t} \approx \frac{e_k - e_{k-1}}{T}$$

The digital PID equation can be represented as

$$u_k = K_P [e_k + \frac{T}{T_I} \sum_{j=0}^{k} e_j + \frac{T_D}{T}(e_k - e_{k-1})] + u_0$$

Where, k=Number of samples,

e_k= Present sampling instant,

e_{k-1}= Previous sampling instant

The outputs of conventional controllers are normally restricted to the range 4 to 20 mA. The manufacturer can provide programs in accordance with the user's specifications. However, if changes are required, the manufacturer must revise the programs, which can be costly. Hence, controllers use programmable ROMs for program storage and can be programmed by the user using a simple dedicated or general-purpose language.

The following technologies are incorporated in these microprocessor-based PID controllers. (1) Multiplexing ADC & DAC (2) High-speed and high-resolution conversion can be achieved using a combination of successive approximation and dual slope conversion techniques. (3) PID algorithm is implemented using software. (4) Controller variations are implemented by adding/changing ROMs. (5) For ease of operation, analog indicator can be used.