

## Direct Digital Control

Om Prakash Bharti

Electrical Engineering Department Indian Institute of Technology IIT(BHU) VARANASI-221005 Varanasi, U.P, India

---

**Abstract:** *With the advent of digital computers the engineers felt the need of utilizing computer advantages in the industrial process through the centralized control system. This thought led to the introduction of direct digital control (DDC) philosophy. In this paper the author has presented a brief review of the structure of DDC as well as its working with illustration. The two types of algorithm used for programming as PID controller are discussed. The author has also pointed out the merits of DDC over analog control. Besides a few practical application has also been described.*

**Keywords:** *digital control, structure, algorithm, application*

---

### I. Introduction

With the introduction of computers, engineers felt the need of utilizing computer advantages in instrumentation field through centralized control system. These advantages were optimization, alarm and logging function, historic analysis, trending, sequence control and self diagnostic techniques. This gave birth to Direct Digital Control philosophy in instrumentation with a back up of analog control system. In 1962, Imperial Chemical Industries (ICI) in England replaced complete analog instrumentation for process control by a Ferranti Argus computer. The computer considered 224 variables and controlled 129 valves directly. The name direct digital control (DDC) was coined to emphasize that the computer directly controlled the processes. DDC i.e. Direct Digital Control is a method of process control in which computer is an integral part of the loop. It essentially consists of microprocessor / pc / computer based controllers and the control logic is performed by software. It is also known as 'Loop Control' as the computer which provides the functions of comparator, controller, limiting and other safeguarding operations, is connected in the feedback loop.

### II. Benefits

Analog control refers to the design and implementation of controllers in continuous domain. Analog controller is restricted by the available hardware elements to take on simple, prescribed forms such as P, PI, PD and PID mode.

Digital controller works around computers and on discrete signals. It offers versatility and flexibility in the sense that complex advanced algorithms can be easily implemented by changing the program. Thus it need not be limited to a three – term (PID) control action only. Direct Digital Control because of computational and networking ability of computers offers advantages over past control technologies. Three distinct benefits of DDC are:

1. **Improved Effectiveness:** As the control loop logic is embedded with the software, this logic can be readily tweaked according to the requirements of the process. Based upon continuous monitoring of the process, more complex control schemes, energy and optimization strategies can be implemented.
2. **Improved operational efficiency:** DDC has the capabilities like offering visualization and storage of data in various formats, trend analysis of data for fault diagnosis and preventive maintenance schedules. They increase the operational efficiency of the plant. Communication capabilities permit remote monitoring and control and helps designers to visualize, diagnose and troubleshoot a problem.
3. **Increased Energy Efficiency:** Energy efficiency routines can be programmed easily. Further, monitoring of energy consumption patterns by each unit permits change of various set points, resulting in efficient utilization of energy.
4. **Economy:** When a process gets more complex and elaborate, one can easily check that a direct digital control may even be economically more viable considering initial equipment cost, operational cost and savings from performance improvement and modifications.

### III. Structure

The DDC directly interfaces to the process for data acquisition and control purpose. Because of the nature of digital devices, signals from the plant have to be converted into a suitable form before they can be transferred for processing by computer. Similarly, signal generated by the computer must be presented in a form compatible with the plant. The Fig.1 shows the various functional blocks of a DDC system.

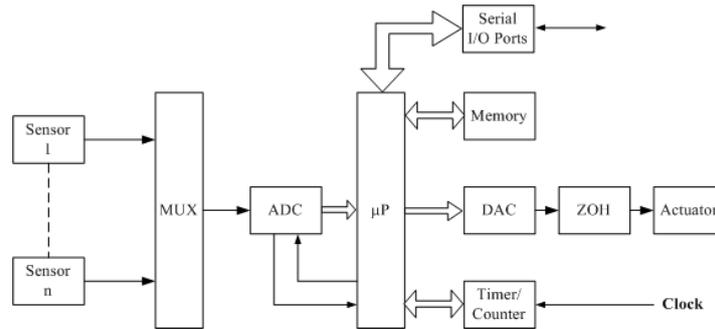


Fig.1

The multiplexer acts like a switch under microprocessor control. It switches and presents at its output the analog signal from a sensor / transmitter.

ADC transforms analog values obtained from sensors into digital signal so that a digital processing unit can work on them. The output of digital control logic is linked with final control elements through DAC. If pneumatic or hydraulic action is required, it is accomplished with electronic-to-pneumatic or electronic-to-hydraulic converters.

The microprocessor performs the following tasks

1. It reads the various process variables from different transmitters through multiplexer and ADC.
2. It determines the error for each control loop and executes control strategy for each loop.
3. It outputs the correction value to control valve through DAC.

Other important pieces of hardware that form part of a digital control loop are

Sampler:

The sampler is essentially a switch, operating usually at fixed intervals of time. When the switch closes, it grabs or samples the signal at that instant of time. Thus, if the source signal is continuous, the output of the sampler is a series of pulses, and the magnitude of each pulse is equal to the magnitude of the continuous signal at the instant of sampling.

Sample hold devices:

The computer output of DAC is a train of pulses. If this is a control signal, the process will be driven by pulses. This is not acceptable driving the valve at certain opening when pulse is present and closing the valve in between the pulses. To overcome this problem Sample hold device is used which holds the signals from DAC. The most common type is the Zero Order hold (ZOH), where each pulse is held until the next pulse comes along.

#### IV. Working

An illustration:

The Fig.2 depicts a DDC flow control loop

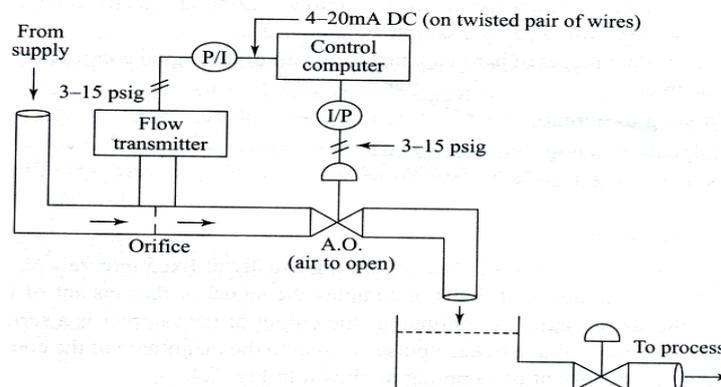


Fig.2

The output of the flow transmitter is converted into an electrical signal using P/I (Pressure-to-Current) converter. The resulting 4-20 mA current signal is converted into a voltage signal and supplied to the computer. The ADC output is subsequently acted upon by the discretized version of controller stored in the computer. The discrete output from the computer is converted back into continuous signal (voltage) by using a DAC, which is further converted to 4-20 mA signal. This current signal is connected to I/P (Current-to-Pressure) converter, producing a pneumatic signal in 3-15 psig range, which operates the valve.

The Fig.3 shows the block diagram of a process control loop with digital controller. Step-like continuous signals are represented as  $\hat{y}(t), \hat{u}(t), \hat{r}(t)$  and sampled signals are represented as  $y(k), u(k), r(k)$ . The controlled variable is measured and the continuous electrical signal is fed to the ADC, where it is sampled at a predetermined frequency. The value of the discrete signal thus produced from the sampler is then compared with the digitized value of the set point stored in the computer.

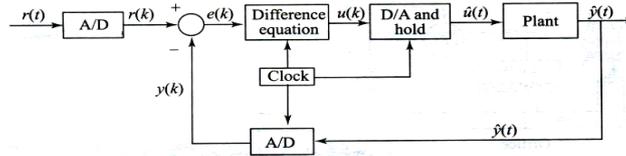


Fig.3

The sampling period, which is the time between the successive samples, is usually constant and is decided by the type of process and the disturbance. An appropriate computer program representing the controller (called the control algorithm) is executed which yields a discrete controller output. This discrete signal is then converted into a continuous electrical signal by means of a DAC followed by a sample hold device.

### V. Algorithm and flow chart

A digital controller accepts discrete sequence of error signals  $e(k)$ , and computes discrete sequence of control action  $u(k)$  according to a discrete control law that may be represented in time domain or in Z-domain. There are two commonly used algorithms for programming analog controllers like P, PI and PID.

- Position algorithm
- Velocity algorithm

Governing equation of a PID control action in continuous domain is given by:

$$m = K_p e + \frac{K_p}{T_i} \int e dt + K_p T_d \frac{de}{dt} + m_i \quad (m_i \text{ is the initial valve position})$$

1. In each sampling period (T), a sampled value of the process output enters the computer. Let  $y(nT)$  be the sampled value at the  $n^{\text{th}}$  sampling instant.  $Y(nT)$  is compared with set point at the same instant which yields the value of discrete-time error,  $e(nT)$  expressed as  $e(nT) = Y_{sp}(nT) - y(nT)$ . In terms of error at sampling instants,  $e(nT)$  may be written as  $e_n$ .
2. Then the discrete time control action produced by the proportional mode based on error present at  $e(nT)$  is  $K_p e(nT)$  or  $K_p e_n$ .
3. The control action produced by the integral mode is based on the integration (accumulation) of errors over a time period. Since error values are available on a discrete-time basis, the integral  $\int e(t) dt$  can be approximated by summation.

$$\int_0^t e(t) dt \approx T \sum_{k=0}^n e(k)$$

Therefore, the integral mode control action is given by:

$$m_n = \frac{T}{T_i} \sum_{k=0}^n e(k) = m_{n-1} + \frac{T}{T_i} e_n$$

4. For the derivative mode action, we need a numerical evaluation of the derivative term  $de/dt$ . Therefore,

$$T_d \frac{de}{dt} \cong \frac{T_d}{T} [e(nT) - e(n-1)T] = \frac{T_d}{T} [e_n - e_{n-1}]$$

Consequently, the control action of a digital PID controller is:

$$m_n = K_p \left[ e_n + \frac{T}{T_i} \sum_{k=0}^n e(k) + \frac{T_d}{T} (e_n - e_{n-1}) \right] + m_i \quad [1]$$

This is called **position algorithm** because the value calculated is the actual manipulated variable, which in many cases represents an actuator (valve, piston, etc.) position. The necessity to initialize the position algorithm (i.e., to somehow supply the initial actuator position,  $m_i$ , to the computer) can be a disadvantage for the controller. Knowledge of  $m_i$  is necessary to ensure a smooth transition (*bumpless transfer*) from manual or no control to automatic control. Additionally, if the computer fails to respond, then the resulting zero output could imply zero position; to which the actuator would respond, perhaps, with undesirable consequences.

With similar analogy, the control valve position at the  $(n-1)^{\text{th}}$  instant can be written as below.

$$m_{n-1} = K_p \left[ e_{n-1} + \frac{T}{T_i} \sum_{k=0}^{n-1} e_k + \frac{T_d}{T} (e_{n-1} - e_{n-2}) \right] + m_i \quad [2]$$

If Eq.[2] is subtracted from Eq.[1], the velocity form of the PID algorithm is obtained. The change in control valve position is better because there is no need to compute the initial position of valve.

$$\begin{aligned}
 m_n - m_{n-1} &= K_p \left[ (e_n - e_{n-1}) + \frac{T}{T_i} e_n + \frac{T_d}{T} (e_n - 2e_{n-1} + e_{n-2}) \right] \\
 &= K_p \left[ e_n - e_{n-1} + \frac{T}{T_i} e_n + \frac{T_d}{T} e_n - 2 \frac{T_d}{T} e_{n-1} + \frac{T_d}{T} e_{n-2} \right] \\
 &= K_p \left[ \left( 1 + \frac{T}{T_i} + \frac{T_d}{T} \right) e_n - \left( 1 + 2 \frac{T_d}{T} \right) e_{n-1} + \frac{T_d}{T} e_{n-2} \right] \\
 &= \left( K_p + \frac{TK_p}{T_i} + \frac{K_p T_d}{T} \right) e_n - \left( K_p + \frac{2T_d K_p}{T} \right) e_{n-1} + \frac{K_p T_d}{T} e_{n-2} \\
 &= A e_n + B e_{n-1} + C e_{n-2}
 \end{aligned}$$

[3]

In Eq.[4],  $A = \left( K_p + \frac{TK_p}{T_i} + \frac{K_p T_d}{T} \right)$ ,  $B = - \left( K_p + \frac{2T_d K_p}{T} \right)$ , and  $C = \frac{K_p T_d}{T}$

The constants A, B and C depend on controller parameters and sampling period. The equations of position algorithm and velocity algorithm may be observed. The term  $m_i$  is absent in Eq.[4]. A zero output from controller implies that the actuator should hold the previous position, which under many circumstances, would represent a safe (“fail-safe”) response.

The integration term actually represents area under the curve which is the collective summation of all the areas at every sampling instant. There are three ways to arrive at this sum viz., (i) forward rectangular rule, (ii) backward rectangular rule, and (iii) trapezoidal rule.

The integral control term expressions for the three rules are given as

Forward rectangular rule:  $u_k = u_{k-1} + K_i T e_{k-1}$  [5]

Backward rectangular rule:  $u_k = u_{k-1} + K_i T e_k$  [6]

Trapezoidal rectangular rule:  $u_k = u_{k-1} + \frac{K_i T}{2} (e_k + e_{k-1})$  [7]

The flow chart for implementing a DDC loop is shown in Fig.4.

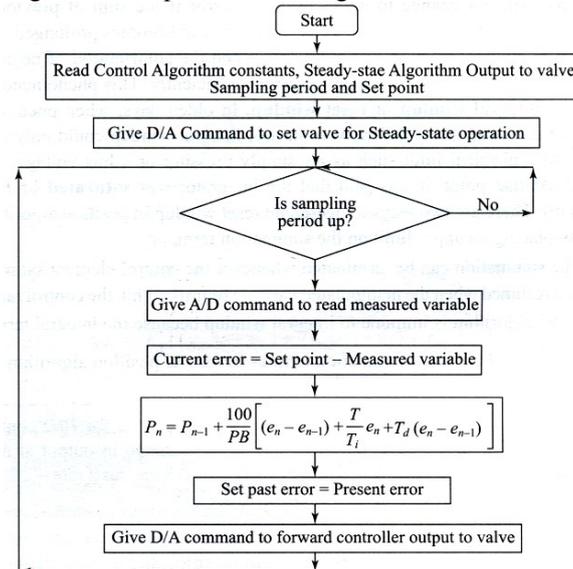


Fig.4

Position Vs Velocity Algorithm:

1. Reference Position:

The major advantage of position algorithm is the reference position. In its equation itself, the reference position of control valve is maintained at  $m_i$ . This is however not there in velocity algorithm. Thus whenever there is disruption due to shut down, communication failure or any other reason, the median position of valve will be known in the position algorithm and eventually valve will catch up without any synchronization problems. In velocity algorithm external device like stepper motor should store the last position of valve.

2. Reset wind up:

When the DDC Control loop is being tested then sensor inputs are connected to see whether the DDC accepts the signals. But control valves in process are not connected so that process is not disrupted. In position algorithm, the integral term increases to its limit because when the process is not connected, process reactions are same and for computer error is persisting. In this case when the DDC system is finally connected, the correction drives control valve to an extreme, causing process disruption. This problem does not arise in velocity algorithm

3. Auto manual switching

The DDC control loop with position algorithm bumps at the time of manual to auto switching. At the time of switching the algorithm must be initiated to zero error situation. Depending upon the process dynamics, operator selects the median position of the valve and corresponding  $m_i$  value. The DDC algorithm has no knowledge of this precise  $m_i$  value and thus switching causes bumps. Velocity algorithm is not susceptible to this problem since  $m_i$  value does not exist in the algorithm.

Thus velocity algorithm with corrections incorporated is best suited for DDC control loop.

### Applications

1. Automation strategy (In steel industry):

Steel industry is one of the most complex industries in terms of process, size and heterogeneity of operations. The control of such an industry is therefore a gigantic task. At the lowest level DDC finds wide applications such as

- (i) Monitoring and controlling the weight, size and quality of various materials.
- (ii) Calculating the exact proportion needed for correct quality charge for coke oven, sintering plant and blast furnace.
- (iii) Sequencing logic for equipments at yards, blast furnace, coke oven and sintering plant.
- (iv) Sequencing logic for positioning of charging conveyors.

2. Kiln automation (In cement industry):

The heart of any cement plant is the kiln. It has to be operated properly for good quality of clinker as well as smooth operation of plant. DDC can control fuel feed rate and keep burning zone temperature within desired limits. Kiln Speed is also monitored and controlled by DDC and thereby clinker burning is controlled. Besides, DDC can be used to control the temperature of cooler discharge and to improve the cooler efficiency by keeping the temperature of secondary air constant.

3. Water treatment plant:

The individual units in water treatment plant can be monitored and controlled by DDC. Following variables are monitored at pre-chlorination stage.

Raw water turbidity, raw water pit, raw water flow, liquid chlorine flow.

Depending upon raw water turbidity, pH and flow of raw water, the flow of liquid chlorine is decided on the basis of ratio control algorithm.

4. Water distribution control:

Depending upon water demands in various localities the water distribution is controlled to satisfy the needs of all. The water demands of various localities are inputs to computer which performs optimization on the basis of present flow in each water main, level in clear water reservoir etc. Based on the result of optimization the computer directs the DDC to start/stop various pumps.

### References

- [1]. C.D JohnsJon, "Process Control Instrumentation Technology", 8<sup>th</sup> Edition, PHI, 2011.
- [2]. D. Patronobis "Principles of Industrial Instrumentation", 3<sup>rd</sup> Edition, TMH, 2010.
- [3]. S.K Singh "Process Control Concepts, Dynamics and Applications", PHI, 2008
- [4]. S. Bhanot "Process Control Principles and Applications", OUP, 2008.
- [5]. Krishna Kant "Computer Based Industrial Control", 2<sup>nd</sup> Edition, PHI, 2012.
- [6]. O.N Pandey, "Instrumentation and Process Control", Kataria & Sons, 2008.



OM PRAKASH BHARTI  
Research Scholar IIT(BHU), Varanasi-221005, Varanasi, U.P INDIA