IMPLEMENTATION OF A DIGITAL PID CONTROLLER USING THE 8751 MICROCONTROLLER CHIP

Abdul Rasool Jabar Al-Zubaidi and Osman Ibrahim Al-Agha

Department of Electronics Engineering, Sudan University of Science & Technology.

Received Jan. 2009, accepted after revision March. 2009

ABSTRACT

The complexity of the controller needed to implement a control law is a function of the plant and the stringency of the control requirements. The cost of an analog controller rises steeply with increasing control function complexity. In fact, implementing a complex control function may even become technically infeasible if one is restricted to use only analog elements. A digital controller, in which either a special purpose or a general purpose computer forms the heart is usually an ideal choice for complex control system. In this paper we will study the implementation of the PID (Proportional, Integral, Derivative) controller on the Intel 8751 microcontroller chip. The PID controller was chosen as being the most famous general purpose controller widely used in the industrial applications. Our final goal is to develop a versatile 8751 microcontroller based data acquisition and control system using PID control algorithm.

Keywords: PID (Proportional, Integral, Derivative) controller, programmable controller, Microcontroller, Digital Controller.

1. INTRODUCTION

Control system is a device or set of devices to manage, command, direct or regulate the behaviour of other devices or systems. There are two common classes of control systems, with many variations and combinations: logic or sequential controls, and feedback or linear controls. There is also fuzzy logic, which attempts to combine some of the design simplicity of logic with the utility of linear control. An automatic sequential control system may trigger a series of mechanical actuators in the correct sequence to perform a task. For example, various electric and pneumatic transducers may fold and glue a cardboard box, fill it with product and then seal it in an automatic packaging machine [3]. Figure 1 shows a simple digital control scheme.

Conversion systems can be used to transform analog signal into digital form for digital-computer processing. In some applications, such as digital control system, digital signals are also transformed to analog signals for controlling the continuous plant. Some analog signals
IMPLEMENTATION OF A DIGITAL PID CONTROLLER

Digital processing is usually employed prior to analog-to-digital (A/D) conversion (an anti-aliasing filter). Figure 2 represents an overall block diagram for a general conversion system. A transducer transforms a physical parameter such as pressure, temperature, velocity, or position into an analog signal (current, voltage frequency, etc.). Further analog processing is accomplished. The analog signal is then converted into digital form (binary-coded) for transfer into the digital processor. After appropriate transfer digital processing, digital signals are transferred to a memory device (register, computer memory, etc.) for storage and then converted to an analog signal for control purposes. The sequencing of the A/D process is controlled by various signals that are generated by the computer interface under software and/or hardware control. The D/A process is usually simpler to control since the output transfer is synchronized to the computer cycle period, and thus data are transferred to the output without any interface “handshaking”. [1], [2].

Figure 1: A Simple Digital Controller

Figure 2: General conversion-system organization for control
2. THE PID CONTROLLER

A feedback controller is designed to generate an output that causes some corrective effort to be applied to a process so as to drive a measurable process variable, \( y(t) \), towards a desired value, known as the set-point or reference, here noted \( r(t) \). The concept is based, as shown in figure 3, on the re-input of the system’s own output according to certain laws (hence the name ‘feedback’). It is desired for the systems output to follow the reference \( r(t) \). All feedback controllers determine their output by observing the difference, called error, here noted \( e(t) \), between the step point and the actual process variable measurement. This theory is valid for a wide class of systems, which include, but is not restricted to, linear systems [5].

For the class of the linear systems, a very widely used controller is the PID (Proportional Integral Derivative). The PID looks at (a) the current value of the error, (b) the integral of the error over a recent time interval, and (c) the current derivative of the error signal to determine not only how much of a correction to apply, but for how long. Each of those three quantities are multiplied by a ‘tuning constant’ and added together. Thus the PID output, \( y(t) \), is a weighted sum as shown in Figure 4.

Depending on the application one may want a faster convergence speed or a lower overshoot. By adjusting the weighting constants, \( K_p \), \( K_i \) and \( K_d \), the PID is set to give the most desired performance [6].

3. PROGRAMMABLE CONTROLLER.

The programmable controller (PC) is a computer-based system to provide a control of discrete-state system. All data input and control output are described as a two state, either ON or OFF.

Because the PC already has an internal computer, it is a logical extension to building the software and hardware required to provide continuous mode controller function. Thus, many PC also have the capability of providing controller actions in the proportional, integral, and derivative modes. For these applications, they have the facility to input and output or store signals such as the common 4-20 mA current.

Figure 3: Feedback Control System

Figure 4: PID Controller
Normal computers, generally, are not used for actual controller operation. They are more suitable for the supervision of an entire plant. The mainframe is the overseer of plant operation. It is also used for engineering studies of plant operation, inventory control, and many other management activities. Figure 5, suggests how a fully integrated computer-based process-control installation might be configured [4].

4. THE DIGITAL PID CONTROLLER

As explained so far we considered time continuous main and analog variables. Today, digital controllers are being used in many large and small-scale control systems, replacing the analog controllers.

It is now a common practice to implement PID controllers in its digital version, which means that they operate in discrete time domain and deal with analog signals quantized in a limited number of levels.

The trend toward digital rather than analog controls is mainly due to the availability of low-cost digital computers [8]. A digital version of the PID controller is shown in Figure 6.

In its digital version, the integral becomes a sum and the differential becomes a difference. The continuous time signal (\(t\)) is sampled in fixed time intervals that are equal to the pre-determined sample period, here called \(T_c\) (in figure 6, \(T_c = 1\)). An A/D (analog to digital) converter interfaces the input and a D/A (digital to analog) converter interfaces the output. This sampled and digitalized input, called \(e_D[j]\), exists only in time instants \(t = kT_c\) for all \(k \geq 0\). It is assumed that these digital values are processed instantly and the result is posted immediately, which obviously is not true. Even if it is possible to deliver the results faster from time to time it is most desirable to maintain a fixed and rigid sample period.

Then it is desirable for the controller to have the sample period \(T_c\) as small as possible and to have as many levels of quantization as possible. A lower bound for the sample period is the computing time of the whole cycle of the digital PID (which includes the A/D and D/A conversion). In most practical situations Noise filtering may imply another lower bound for the sampling period. The number of levels of quantization of the input and output analog variables will depend on the resolution of the A/D and D/A converters respectively [7].

4.1 Proportional Control.

The proportional mode controller action is defined by a term that is directly proportional to the error. The equation is

\[ P = K_p e_p + P_o \]  

(1)

where:

- \(K_p\) = proportional gain
- \(e_p\) = error
- \(P_o\) = controller output with no error.

The gain is expressed as percent controller output per percent error. The concept of proportional band (PB) is defined as \(1/K_p\) and represents the percentage error that will cause a 100% change in controller output. This mode is easily implemented by the computer in the form of an algorithm that simply calculates Equation (1) directly [9].

The proportional mode is provided through the software by an equation that is entirely like the analog equation. Because we are expressing the error as a fraction of the range of the output, what is calculated is a fraction of the maximum output.

\[ P = P_o + K_p e_p \]  

(2)

\[ P_out = P * R_out \]  

(3)

where:

- \(P_o\) = fraction of output with no error
- \(K_p\) = proportional gain (\%/\%)
- \(P\) = fraction of output with error.
- \(R_out\) = maximum output
- \(P_out\) = output
- \(e_p\) = error

4.2 Integral Mode.

The integral or reset mode calculates a controller output that depends on the history of the controlled variable error. In a
in a mathematical sense, history is measured by an integral of the error.

\[
P = K_i \int_0^t K e_p \, dt + p(0)
\]  \hspace{0.5cm} (4)

Where;

- \( K_i \) = integral gain in percent controller output per percent-second error (or, more commonly, per minute).

To use this mode in computer control, we need a way of evaluating the integral of error. There are many algorithms that have been developed to do this, all of which are only approximate, as only samples of the error in time are available. The simplest is called rectangular and is often accurate enough for use in process control. To see how this works, we should note that the integral in Equation (5) merely the net area of the \( e_p \) curve from 0 to \( t \). This is shown in Figure 7.

**Figure 5: Mainframes are used manage SBC process controllers.**

**Figure 6: Digital PID Controller**
IMPLEMENTATION OF A DIGITAL PID CONTROLLER

a) Exact integral = net area = (area above) - (area below)

b) Approximate integral = sum of rectangle areas

Figure 7: Rectangular integration algorithm construction.

\[ \int_{0}^{t} ep \, dt = \text{Net Area} = (\text{area of } ep > 0) - (\text{area of } ep < 0) \quad \ldots \quad (5) \]

\[ \int_{0}^{t} ep \, dt = (S + epi) \Delta t \quad \ldots \quad (6) \]

In rectangular integration, we simply use the periodic samples of \( ep \) to construct a series of height equal to the sample error and approximately equal to the time between samples. The integral (or area) is shown in figure 8.17 a. In an equation, rectangular, rectangular integration specifies that,

Where,

\( \Delta t \) = time between samples

\( S = e_{p1} + e_{p2} + \ldots + \) (sum of errors calculated from previous variable samples)

\( e_{pi} = \) last sample taken at time \( t \) specified in the integral.
It should be clear that the smaller the time between samples, the more closely the approximate answer will approach the actual integral[9].

The issue of sampling time becomes important for the integral mode. If the time between samples is too large, the error and control will be compromised. If the criteria of $t_s = 10t_{max}$ is satisfied, then the integral term also will be of sufficient accuracy using the rectangular algorithm.

Implementation of this mode in software involves the following basic equations:

$$\text{SUM} = S \times D_T$$
$$P_t = K_I \times \text{SUM} + P_0$$

Where:

- $\text{SUM}$ = a running sum of errors
- $K_I$ = the integral gain
- $D_T$ = time between samples
- $P_t$ = fraction of maximum output with error.
- $P_0$ = fraction of maximum output with no error.

4.3 Derivative Mode

The derivative controller mode, also called rate, derives a controller, output that depends on the instantaneous rate of change of the error.

$$P = K_D \frac{de_{pi}}{dt}$$

Where;

- $K_D$ = derivative gain (% per % error) $\frac{de_{pi}}{dt}$
- $\frac{de_{pi}}{dt}$ = rate of error change in percent per second (or minute)

The gain expresses the percent controller output for each percent/second change in error. This mode is implemented in computer control by calculating an approximate derivative of the error from the data samples. A derivative is defined as the rate at which a quantity is changing at an instant in time. We can calculate only the rate at which it is changing over the sample period $t$, which is therefore only an approximation. In terms of an equation, this is:

$$e_{pi} = \text{present error sample}$$
$$\Delta t = \text{time between samples}$$
$$\text{DEO} = \text{Change in Output Signal}.$$  
$$\text{DE} = \text{Change in Error Signal}.$$  

The set of equations for the derivative output can be developed directly from the definition. It is found:

$$e_{pi2} = e_{pi1} - \text{DEO} \quad \text{........... (10)}$$
$$\text{DE} = \text{DEO} \quad \text{........... (11)}$$
$$\text{PD} = K_D \times \text{DDE} / \text{DT} \quad \text{(12)}$$

As in the case of the integral mode, it is very important that the units of $KD$ and $DT$ agree.

5. HARDWARE ORGANIZATION

Our digital PID controller is essentially composed of the following sections.

5.1 Data acquisition section.

The data acquisition section consists of ADC0804 and CD4051B integrated circuit. The ADC0804 is an 8-bit microprocessor compatible, analog to digital converter. And the CD4051B is an 8-channel multiplexer. This section has the ability to transfer analog signals corresponding to the process variable and the set point, into digital form. The analog signal for the process variable can be the signal from the transducers and conditioning circuits or it can be directly a voltage signal varying from 0 to 5v. The analog signal for the set point is obtained from 5v power supply and potentiometer circuit, having output from 0 to 5 V[9].

5.2 Single board computer (SBC).

The signal board computer (SBC) is based on Intel 8751 micro-controller. This microcontroller provides 4k bytes of ultraviolet erasable programmable read only memory (EPROM) for program development. It also has 128 bytes data memory on chip, 32bidirectional I/O lines organized as four 8 bit I/O ports, multiple mode high speed programmable serial port, two multiple mode, 16 bit timer/counters, two level prioritized interrupt structure and one microsecond instruction cycle with 12 MHz crystal[9].
All the software of the controller can be written in the assembly language or C language of the microcontroller 8751 and stored in the embedded EPROM of the 8751. All the 8bit devices (i.e. ADC, DAC display driver, buffers, latches etc.) are connected to one port of the 8751 (say port 0), and the remaining port lines provide the enable signals for the above devices connected to the microcontroller 8751.

5.3 Keyboard section.

This section consists of 0 to 9 numeric keys and two function keys on a 4×3 key pad and eight more separate function keys. The 4×3 key pad is used for first entering and then scrolling the constants K_p, K_d and K_i of the PID controller, while the 8 separate function keys are used to display output, process variable, set point in percentages and sampling time in seconds one by one, to increase or decrease the output by pressing the key again and again for manual control [9].

5.4 Display section.

This section consists of ICM7218B (microprocessor compatible 8 digits, 7 segment display driver), 8 digits, 7 segment display, 3 bar graphs each consisting of 8- LED’s. Each bar graph is connected to an 8 bit latch for holding the value to be displayed constantly. The bar graphs are used to display the process variable, set point and the output. While the 8 digits 7-segment display is used for displaying the output, process variable, set point in percentages and the constants K_p, K_d and K_i during editing mode (tuning) of the PID controller[9].

5.5 Analog output section.

This section consists of digital to analog converter (MC1408) and voltage to current converter using op-amp LM324. The MC1408 is an 8-bit multiplying digital to analog converter, having current output. This current output is converted into standard 0 to 5V voltage output using op-amp LM324 and then this voltage output is converted into standard 4 to 20mA current output with a voltage to current converter. This standard voltage output and current output are used to drive the final control element in the process loop [9].

6. EXPERIMENTAL RESULTS

Table 1 below summarizes the comparison between the conventional analog PID controller and our digital PID controller based on the results attained from the paper.

<table>
<thead>
<tr>
<th>Digital PID Controllers base on 8751 Chip</th>
<th>Analog PID Controllers</th>
</tr>
</thead>
<tbody>
<tr>
<td>More economical because of cheap components and the simple design algorithm.</td>
<td>Comparatively expensive due to the complexity of the design algorithm.</td>
</tr>
<tr>
<td>Fully integrated and compact.</td>
<td>A large number of operational amplifiers and other components are needed.</td>
</tr>
<tr>
<td>High noise immunity.</td>
<td>Noise susceptibility is high.</td>
</tr>
<tr>
<td>More flexibility because of the ability to program and reprogram our chip.</td>
<td>Redesigning is required for any change in the system parameters.</td>
</tr>
<tr>
<td>High accuracy with faster processing and low power consumption.</td>
<td>Less accurate with more processing time and power consumption is higher.</td>
</tr>
</tbody>
</table>
7. SUMMARY AND CONCLUSION

Digital control permits the use of sensitive control elements with relatively low-energy signals. The advantages of using a digital transducer is the relative immunity of its digital signals to distortion by noise and nonlinearities and its high accuracy and resolution as compared to analog transducers. The employment of discrete digital signals provides for the design and development of complex and sophisticated control systems. For some control-system applications, improved system performance can be achieved by a sampled-data control system design over a continuous-data control-system design because of better noise filtering.

References


