**Analog and Digital
Motor Control**

Curriculum Manual CA06

Issue: ME1219/E
Lesson Module: 17.06/0

Chapter	**Contents**	**Pages**

Introduction

Introduction

This curriculum text is designed to introduce analog and digital control methods used to control the MS15 DC Motor Control Module.

The work is presented in a largely non-mathematical form with the aim of clearly establishing the principles involved prior to the mathematical approach that has to be accepted beyond this level.

This curriculum makes use of Real-time Windows^{TM} based Virtual Control Laboratory software and a Control Laboratory Input/Output (CLIO) interface module which enables the student PC workstation to:

- perform as a function generator to supply (if required) the command (reference) input signal in various forms, for example, step or sinewave inputs.
- supply a wide range of different adjustable controller configurations, for example, open-loop or PID.
- perform as an eight-channel oscilloscope or voltmeter to display various control signals, for example, command input, position output.

Because this curriculum is concerned with the control of a servomechanism (the MS15 DC Motor Module) the terminology used is that related to servo systems rather than to process systems but the basic concepts are of course applicable to all types of control.

Throughout industry today, microcomputers are increasingly being used to control electromechanical servomechanisms with applications that can vary from the driving and controlling of elevators to robotic drive and control systems. This text looks at this important area of systems control and explains in detail the techniques employed when using a microcomputer to carry out these various control tasks.

The digital control material presented in the later chapters of this curriculum text is split into two sections in order to cover this subject in its broadest possible context. The first explains the techniques associated with analog interfacing and the second with digital interfacing. Numerous hands-on exercises are included covering both approaches.

WARNING

All DC motors have a limited life. When not performing an experiment, switch the unit off or remove the signal applied to the motor input (click disable on the control software).

The motor should be prevented from entering a state of high frequency oscillation for more than a few seconds. This manifests itself as either an audible ‘buzz’ or rough running of the motor. If this condition is not removed the motor may fail.

The experiments in this manual ensure that this condition should not arise. If it does, remove the input to stop the oscillation and check that the instructions have been followed correctly.

The Module Power Supplies

The System Power 90 Power Supply (or equivalent) contains all of the power supplies needed to make all the modules operate. You can switch these power supplies ON and OFF with the switch located on the front panel.

Making Circuit Connections

During each Practical Exercise in this manual, you will be asked to make circuit connections using the 4 mm Patching Cords. Whenever you make (or change) circuit connections, it is good practice to always do so with the Power Supplies switch in the OFF position. You should switch the Power Supplies ON only after you have made, and checked, your connections.

Remember that the Power Supplies switch must be ON in order for you to be able to make the observations and measurements required in the Exercise.

At the end of each Exercise, you should return the Power Supply switch to the 'OFF' position *before* you dismantle your circuit connections.

Your Workstation

Depending on the laboratory environment in which you are working, your workstation may, or may not, be computer managed. This will affect the way that you use this curriculum manual.

If you are in any doubt about whether your workstation is computer managed, you should consult your instructor.

Using this Manual at a Computer Managed Workstation

In order to use this curriculum manual at a computer managed workstation you will require a personal computer (PC) that has been installed with computer managed student workstation software.

If you are working in a computer managed environment for the first time, you should first read the operating information that has been provided with your computer managed workstation. This tell you how to:

- Log onto the management system and request work.
- Make responses to questions in a computer managed environment.
- Hand in your work when completed.
- Log off at the end of your work session.

Whenever you see the symbol  in the left-hand margin of this Curriculum Manual, you are required to respond to questions using your computer managed workstation. You should also record your responses in your student workbook so that you can review them at any time in the future.

The following D3000 Lesson Module is available for use with this Curriculum Manual:

D3000 Lesson Module 17.06

Using this Manual at a Workstation that is *not* Computer Managed

Whenever you see the symbol  in the left-hand margin of this Curriculum Manual, you are required to answer a question. If your workstation is *not* computer managed, you should record your answer in your student workbook so that it can be subsequently marked by your instructor.

Good luck with your Studies.

Chapter 1
Analog Control - Equipment Overview

Objectives of this Chapter

Having completed this chapter you will be able to:

- Locate the analog features of the DC Motor, Input Potentiometer and Interface Modules
- Connect together the modules for analog control exercises
- Operate the software controls to drive the DC Motor and observe its behavior

Equipment Required for this Chapter

- MS15 DC Motor Module
- AS3 Command Potentiometer
- CLIO Interface Module with PC Connection Lead
- System Power 90 Power Supply (or equivalent)
- 4 mm Connecting Leads
- PC running VCL Virtual Control Laboratory Software
- Trim Tool

1.1 Introduction

Practical exercises are an integral part of the course and waveforms displayed on the PC are used to illustrate parts of the course.

A PC with the Virtual Control Laboratory software and CLIO Control Laboratory Input/Output interface is used to replace a number of different traditional instruments, eliminating the need for separate signal generator, oscilloscope, multimeter or controller.

In this first chapter, you will refresh your knowledge of the DC Motor which is used as the 'Plant' and familiarize yourself with the interface board and the operation of the software.

1.2 The MS15 DC Motor Module

The 'Plant' is the MS15 DC Motor Module (Fig 1.1). This is an extremely versatile unit which provides a sufficient variety of instrumentation on board to enable a whole range of analog and digital investigation to be carried out. A schematic of the system is shown in Fig 1.2.

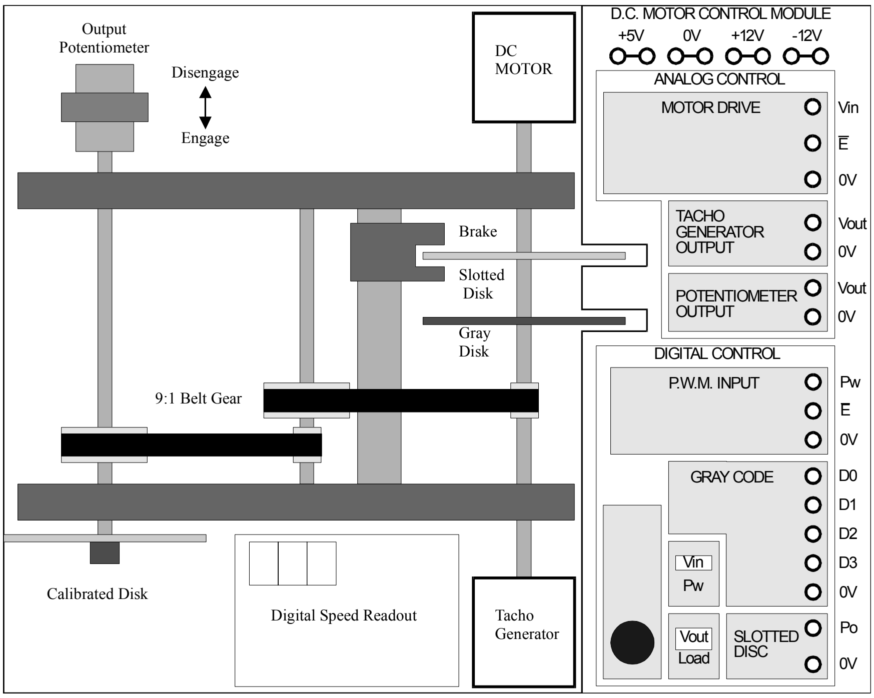


Fig 1.1 The MS15 DC Motor Module

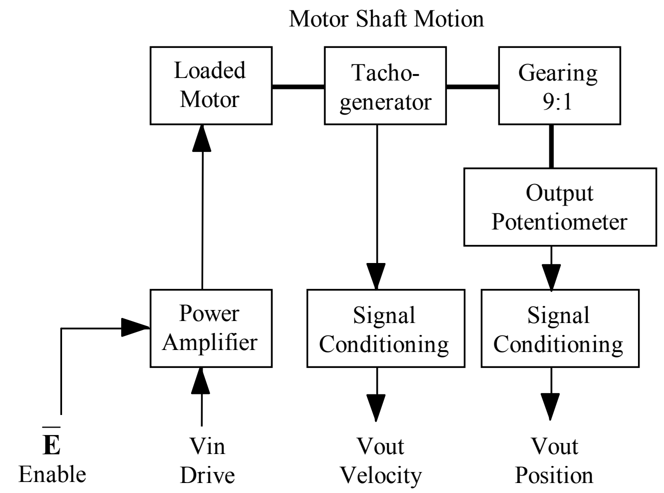


Fig 1.2 Schematic of MS15 DC Motor Module

The DC Motor system

A constant voltage applied to the DC motor produces a constant torque which, with a constant load on the motor, produces a constant motor speed. This applied voltage is the plant input.

The motor speed (or more correctly velocity) is measured using a tachogenerator mounted on the same shaft as the motor. A tachogenerator is just another motor connected in a different way and produces a voltage proportional to the motor speed. The voltage from the tachogenerator is used as the plant output in motor speed control experiments.

The motor drives an output shaft via a 9:1 speed reduction two stage belt drive. This means that the output shaft rotates at 1/9th the speed of the motor shaft. The position of the output shaft can be seen on the calibrated dial on the front of the unit. A voltage proportional to output shaft position is produced by a potentiometer mounted on the rear of the output shaft. The voltage from this potentiometer is used in position control experiments.

The plant input

The 12V motor requires significantly more current than can be supplied by the drive signal therefore the MS15 module contains a power amplifier which allows the low power drive signal to modulate the high power supply to the motor. An input of +5 volts will produce maximum speed in the counter-clockwise direction.

The 4 mm socket for this input is in the MOTOR DRIVE INPUT section of the module and is marked $\mathbf{V_{IN}}$.

The analog drive input is only available when the Not Enable ($\bar{\mathbf{E}}$) line is grounded and the MOTOR DRIVE switch (upper switch at center bottom of the circuit board) is in the $\mathbf{V_{IN}}$ position.

Velocity (or speed) output

The tachogenerator is connected to the output shaft and produces (after signal conditioning) a DC voltage in the range ±5 volts proportional to the rotational velocity of the motor. This output provides the primary feedback of information for closed loop speed control applications and performs a secondary role in many positional applications. A positive voltage is produced with the output shaft rotating in a counter-clockwise direction.

The 4 mm socket for this output is in the TACHOGENERATOR OUTPUT section of the module and is marked $\mathbf{V_{OUT}}$.

This velocity output signal is only available when the TACHOGENERATOR switch (lower switch at center bottom of the circuit board) is in the $\mathbf{V_{OUT}}$ position. With the switch in the **LOAD** position, the tachogenerator is connected to a variable resistor load which can be used to load the motor.

Position output

The output shaft carries a calibrated disc (degrees) and is coupled to a precision potentiometer which, via appropriate signal conditioning, provides a DC voltage in the range ±5 volts according to the position of the output shaft. This potentiometer output provides the primary feedback of information for closed loop position control over about 340° of output rotation. The output voltage is 0 when the shaft is at 180°, positive for angles above this and negative for angles below this.

The 4 mm socket for this output is in the POTENTIOMETER OUTPUT section of the module and is marked $\mathbf{V_{OUT}}$.

1.3 The AS3 Command Potentiometer

The AS3 Command Potentiometer (Fig 1.3) is used to provide a manually generated input signal. It is calibrated in degrees to correspond with the calibrated disc on the output shaft of the MS15 DC Motor Module. The unit gives a nominal output of ±5V.

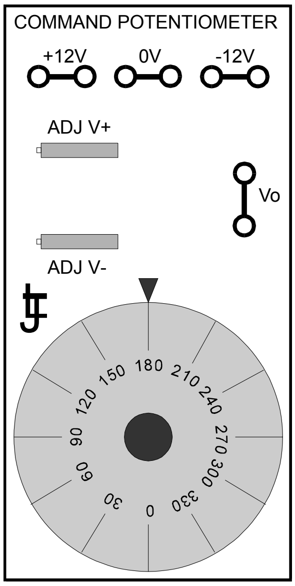


Fig 1.3 Command Potentiometer

1.4 Virtual Instrumentation

Virtual Instrumentation means that a personal computer (PC) is being used in place of a number of conventional instruments such as a Signal Generator, Oscilloscope, Panel Meter and a number of different types of controller.

Fig 1.4 overleaf shows a schematic of the Virtual Control Laboratory used in this course.

In this course we are concerned with understanding the performance of the closed loop systems rather than how you would build a controller. To simplify the experiments, all the controls are on the screen. This means that the control is always being implemented digitally but, as long as the conversion to digital and the conversion back to analog are performed fast enough, the system can represent an analog controller.

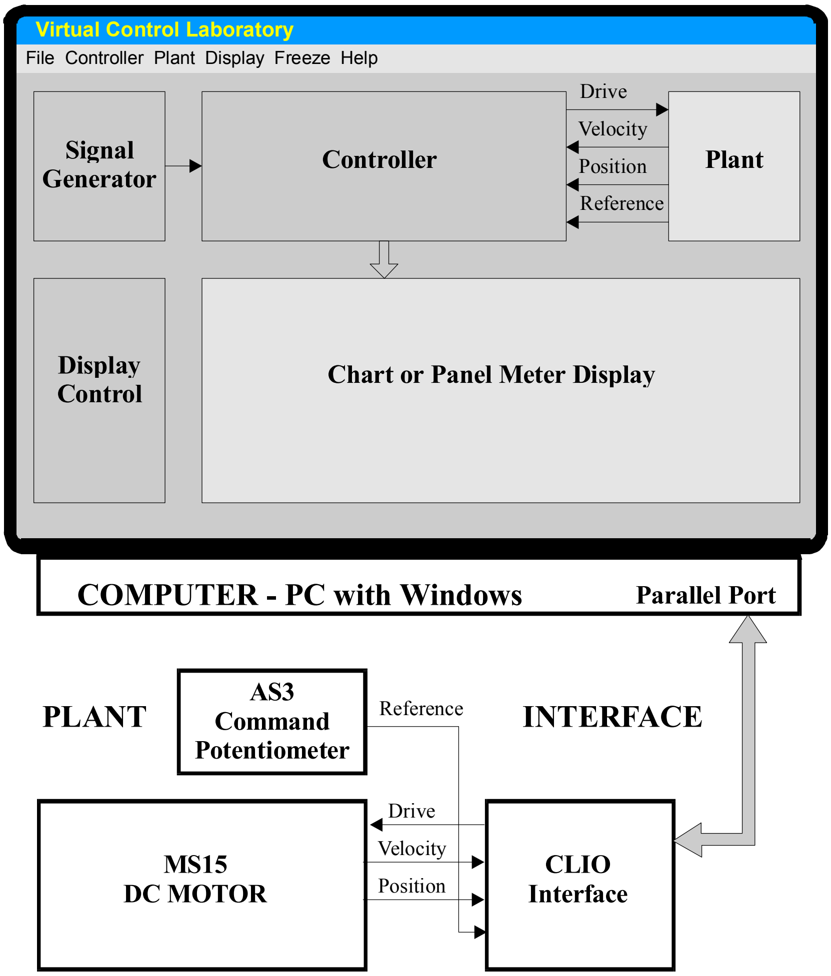


Fig 1.4 Schematic of the Virtual Control Laboratory System

The digital computer is connected to the MS15 DC Motor module via the CLIO Interface unit. This contains a fast Digital to Analog Converter (DAC) which provides a continuous analog drive to the motor and a 3-input Analog to Digital Converter (ADC) to convert the analog outputs from the motor (tachogenerator and potentiometer) and the command potentiometer into digital form for the computer.

1.5 The CLIO Interface Module

The Control Laboratory Input/Output (CLIO) module is the interface between the analog voltages of the motor system and the digital numbers of the computer system.

Fig 1.5 is a layout diagram of the interface module.

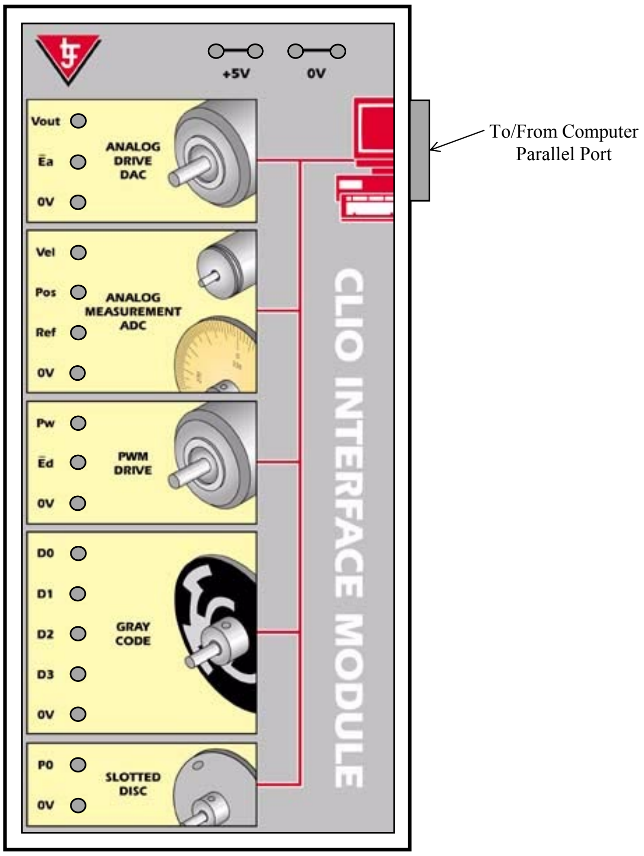


Fig 1.5 CLIO Interface Module

The input/output connections described below should be identified on the actual unit.

Analog drives - Motor module inputs

An analog signal is required to drive the DC motor. The Analog Drive provides a ±5 volt signal. This is obtained from the PC via a fast DAC (Digital-to-Analog Converter).

The 4 mm socket for this output is in the ANALOG DRIVE section and is marked $\mathbf{V_{OUT}}$.

The drive to the motor should be disabled when the motor is not in use. This can be controlled from the computer. The disable signal appears on the $\mathbf{\overline{E}_a}$ socket which should be wired to the $\mathbf{\overline{E}}$ socket on the motor board. The 4 mm socket for this output is in the ANALOG DRIVE section of the CLIO interface and is marked $\mathbf{\overline{E}_a}$.

Analog Measurements - Motor module outputs

To control the motor it is necessary to know what the motor is doing. This requires measurement. Three signals (each ±5 volt) can be measured by the Analog Measurement ADC (Analog-to-Digital Converter) and transmitted to the PC. The three signals measured are :-

Position

The output shaft position provided by the MS15 Potentiometer output voltage Vout.

The 4 mm socket for this input is in the ANALOG MEASUREMENT section of the CLIO interface and is marked **Pos**.

Velocity

The motor shaft velocity voltage provided by the MS15 Tachogenerator output voltage Vout.

The 4 mm socket for this input is in the ANALOG MEASUREMENT section of the CLIO interface and is marked **Vel**.

Reference input

The required (or reference) position or velocity as provided by the AS3 Command Potentiometer $\mathbf{V_O}$.

The 4 mm socket for this input is in the ANALOG MEASUREMENT section of the CLIO interface and is marked **Ref**.

1.5a **The purpose of the MS15 Potentiometer is to measure:**

[a] output position. [b] output speed.

[c] output torque.

1.5b **The purpose of the MS15 Tachogenerator is to measure:**

[a] output position. [b] output speed.

[c] output torque.

1.5c **The purpose of the AS3 Command Potentiometer is to produce an input for:**

[a] position control. [b] speed control.

[c] either depending on the application.

1.5d **The measurement range of the input signals to the CLIO module is:**

[a] ±2V [b] ±5V [c] ±10V [d] ±15V

1.6 Operating the Hardware System

Engaging and disengaging the output potentiometer

For position experiments, the output potentiometer is required. The output potentiometer is on the output shaft at the opposite end from the calibrated dial. The output potentiometer should be disengaged when not required, such as during speed control experiments, as unnecessary use will wear the unit. Practice engaging and disengaging the output potentiometer.

- To disengage the output potentiometer, push the output potentiometer assembly backwards, away from the calibrated dial.

- To engage the output potentiometer, align the flat on the output shaft with the flat on the output potentiometer boss, then pull the output potentiometer assembly forward towards the calibrated dial. It should move about 7 mm.

Leave the potentiometer in the engaged position.

Applying the brake

The eddy current brake is used to put a load onto the motor. In the upright position (position 0) there is no braking action. There are 2 braking positions marked 1 and 2. Position 2 gives heavier braking than position 1.

Wiring the system

All the experiments in the Analog section of this course are physically wired in the same way. Different configurations are achieved using data signal paths in the computer. Fig 1.6 below shows the wiring between the MS15 DC Motor module, the CLIO Interface Module and the AS3 Command Potentiometer.

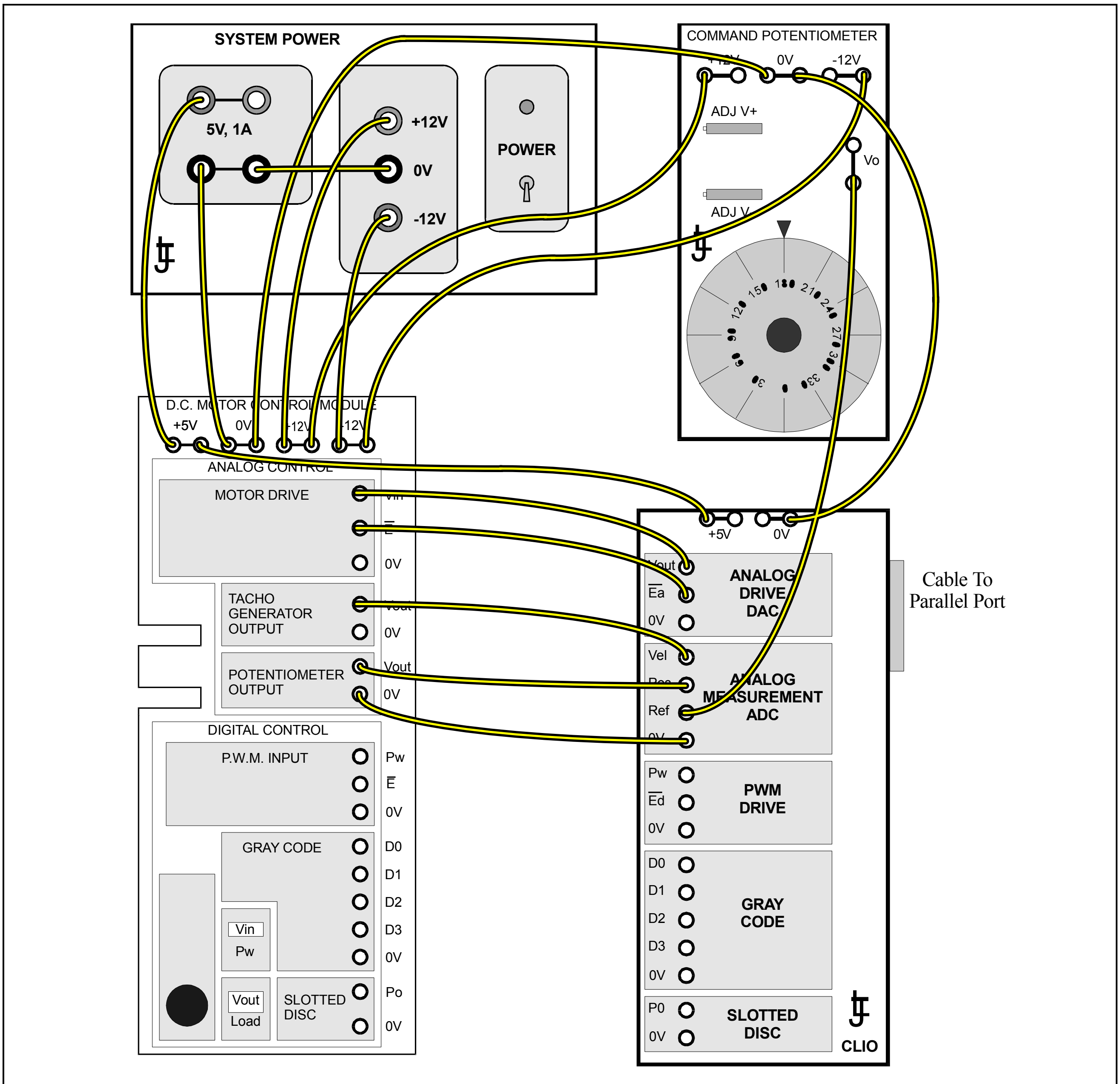


Fig 1.6 Wiring Diagram - Analog Control

These connections should be used unless instructed differently. If possible, the circuit should be left connected between experiments. Whether left connected or not, these connections should be checked at the beginning of each practical exercise. Wire up the system as in Fig 1.6 but, for the moment, leave the switch on the System Power 90 power supply in the OFF position.

1.7 The Virtual Control Laboratory Software

Much of the instruction in this course involves observing and understanding the responses of the plant under different operating conditions. Most of the observations and changes will be made using the PC-based VCL software and the mouse.

Loading the software

Start the Virtual Control Laboratory software by selecting the VCL Start menu item in the LJ Create group, or if running through a ClassAct managed workstation, click ‘Launch CAI/App’ and enter VCL as an Application Launch Code. Some time will now be spent introducing the elements of the package.

Loading the setup for the current experiment

Load the initial setup of Exercise 1 by selecting **File** from the Tool Bar then **Load setup** from the drop down menu. From the Load Dialog, select file **CA06PE01.ini** and click the **Open** button. [Note: If the CA06PE01.ini file is not in the list of files, navigate to the Program installation folder – usually C:\Program Files\LJ Create\VCL. If the file still cannot be found, ask your instructor to 'reset' the configuration files.]

In subsequent instructions, such a command sequence will be written:

File | Load setup | CA06PE01 | Open or just **Load setup | CA06PE01**.

The control system elements across the top of the PC screen

The Signal Generator

This is at the left of the screen. It is used to generate a variety of input signals. This source is used as the reference when *Internal* or *Int+Ext* is selected as the reference signal to the control system.

The Controller

This is the top center area of the screen. The controller is currently set to **Open Loop**. Different controllers can be selected via the **Controller** option on the Tool Bar.

The Plant

This is to the right of the screen. For this section of the course, the **Plant** is normally set to **MS15 Analog** indicating that the software is assumed to be interfaced to the analog control section of the MS15 DC Motor Control Module.

The Display

This takes up the lower two-thirds of the screen. Up to 8 channels can be displayed, normally in graphical form. This allows the relationships between various signals to be examined. The channels are color coded with the measurement point, channel number and trace color shown on the Controller mimic diagram. To access the controls of a particular channel, point the mouse and click at the channel number in the channel select area. Only the channels which have a meaning with the selected control method are available.

There are three other display options selectable via the **Display** option on the Tool Bar. These are:- Bar, Meter and List. Select **Display | Meter** and you will see a numerical display of the signals being measured. Go back to Graph using **Display | Graph**.

Adjusting on-screen controls

All on-screen controls are set by pointing and clicking.

'Flip' Controls

These have a control bar above and below the legend or number. Clicking the upper bar flips the legend to the next in the sequence or increments the digit. Clicking the lower bar flips the legend to the previous one in the sequence or decrements the number. Try it and see - you will not harm anything. Click above Signal Generator, *Signal*, **Step** and see the legend change to **Random**. Click below **Random** and see the legend return to **Step**. Note that the click area actually extends up or down from the center of the legend. Now click above and below the Signal Generator *Level* numbers. When you are sure how this works, return the number to 0%.

'Select' Controls

These look like push buttons. Click on the option you require. To practice, point and click at 2 in the Channel Select area and see the Scale change to show the scale for channel 2. The *ON/OFF*, *Magnify* and *Shift* controls for channel 2 are now available. These behave rather like oscilloscope controls. Change them to see their effects. Return them to *Magnify = 1*, *Shift = 0.*

1.8 Getting to Know the Equipment

Working with VCL Software and MS15 Motor Control Module

Wire up the system or check the wiring as shown in Fig 1.6.

In this exercise you will learn your way around the MS15 hardware and the VCL Software.

If you do not have the software loaded, start it now by selecting the VCL Start menu item in the LJ Create group or, if running through a ClassAct managed workstation, click ‘Launch CAI/App’ and enter VCL as an Application Launch Code. Load the initial setup for this experiment by selecting:

File | Load setup | CA06PE01 | Open

The table below is the summary of the setup required for this experiment. The values shown have been set by loading setup file CA06PE01. Any of the values may be changed by accessing the appropriate menu from the tool bar or by clicking the on-screen controls. The items on the last two rows are test rig settings and must be set manually.

File	**Controller**	**Plant**	**Display**
CA06PE01	Open-loop	MS15 Analog	Graph
Signal Generator		**Graph**	
Signal	DC-Level	1 Input	ON
Level	50%	2 Position	ON
Offset	0%		
Rate	10 msec	4 Velocity	ON
Reference	External		
DC Motor		**Output Potentiometer**	Engage
Brake	0	**Command Potentiometer**	180°

This table format will be used throughout this manual to indicate the initial settings for an experiment and for changes that need to be made to existing settings.

Set the command potentiometer to 180° and switch power ON. Nothing much should happen. Enable the motor by clicking in the *Disable* box

Changing speed using the Command Potentiometer

Rotate the Command Potentiometer towards 360°. The motor will start to rotate counter-clockwise. On the screen you will see the traces move.

- **Channel 1/Dark Blue/Input** follows the command potentiometer movement. With the angle above 180°, the drive voltage is positive and the shaft rotates counter-clockwise.

- **Channel 2/Blue/Position** shows the position of the output shaft. As the output shaft position moves towards 360° on the calibrated dial, the trace moves up the screen towards +5 volts. Near the top, the output potentiometer reaches the end of its range then jumps to the bottom of the screen as it picks up the negative voltage equivalent to 0° (approximately -5V).

- **Channel 4/Purple/Velocity** follows the command input as it is changed but lags behind any movement. This is the measurement of speed of rotation from the Tachogenerator. It can be seen that this signal tends not to show a constant velocity even when the command potentiometer is not being moved. This is a characteristic of mechanical tachogenerators.

Vary the position of the command potentiometer and observe how the traces change. As the command position goes below 180°, the motor rotates in the opposite direction and the output position ramps down before shooting up to the top of the screen.

Stop the motor by setting the command potentiometer to 180°.

Changing Speed using the Signal Generator

In the **Controller** area, set *Reference* to **Internal** by clicking on the bar below **External**. This selects the internal signal generator as the reference input signal. Select *Signal* **DC Level** and increase the *Offset* to 50%. This has the same effect on the motor speed (and the display) as increasing the angle of the Command Potentiometer. Make the *Offset* negative and observe that the motor rotates in the opposite direction. Return the *Offset* to +50%.

Applying the Brake

The eddy current brake is located forward of the DC Motor. This can be varied through three positions:- 0 = Off, 1 = Half and 2 = Full. Set the brake to each position and note the effect of the braking action upon the rotational speed of the motor.

Calibrating the Command Potentiometer

For correct operation, the command potentiometer and the output potentiometer should generate the same voltage when at the same angular position. It may be necessary to adjust the trim pots on the command potentiometer to ensure that the voltages are the same.

- Select **Display | Meter**
- Set *Reference* to **External**. This selects the command potentiometer as the input to the system.
- Disable the motor drive by clicking *Enable* in the controller area.
- Manually set both potentiometer dials to 300°. Using the trim tool, adjust ADJ V+ on the Command Potentiometer module until the Input voltage (channel 1) equals the Position voltage (channel 2).
- Set both potentiometer dials to 20°. Adjust ADJ V- on the Command Potentiometer until again the Input voltage equals the Position voltage.

If a large adjustment has been made, the last two steps should be repeated to ensure correct calibration. It is difficult to repeat the exact degree settings on the dials so do not waste time repeating the steps more than once.

Relationship between degrees and volts

All the measurements are made in volts so we need to know the relationship between the measured position voltage and the position in degrees.

Rotate the command potentiometer until the measured input voltage on channel 1 is 0V. Note the degrees in column 1 of Table 1.1 in your workbook.

Rotate the potentiometer until the output reads 1V. Enter this in column 2.

Calculate K_d by subtracting Degrees at 0V from Degrees at 1V.

Offset - Degrees at 0V	Degrees at 1V	Gain K_d - degrees/volt

Table 1.1 Potentiometer calibration

The relationship between degrees and voltage is then:

Degrees = K_d x V_{pos} + Degree Offset = [x Vout +]

Finishing with the equipment

If not continuing immediately with Chapter 2, switch power OFF, exit from the software and dismantle the wiring if instructed to do so by the laboratory supervisor.

1.8a **In speed control, with External input, the motor is stationary when the command potentiometer is at:**

a 0° b 90° c 180° d 270°

1.8b **Rotating the command potentiometer towards 360° causes the motor to rotate:**

a clockwise. b counter-clockwise.

1.8c **Setting the Signal Generator Offset positive causes the motor to rotate:**

a clockwise. b counter-clockwise.

1.8d **Setting the brake to position 2 causes the motor to:**

a speed up. b slow down.

c stay at the same speed.

Student Assessment 1

1. **For a constant input voltage, the DC Motor in the MS15 produces:**
   - a constant speed.
   - b a fixed position.
   - c constant acceleration.

2. **The Command Potentiometer provides:**
   - a a measure of the motor speed.
   - b a measure of the motor position.
   - c the external reference input to the controller.
   - d the setting for the motor position.

3. **The DAC in the CLIO interface converts:**
   - a digital numbers in the computer to analog motor drive voltages.
   - b analog voltages from the MS15 to digital numbers in the computer.
   - c digital signals in the MS15 to digital numbers in the computer.
   - d digital signals in the MS15 to analog signals in the computer.

4. **The ADC in the CLIO interface converts:**
   - a digital numbers in the computer to analog motor drive voltages.
   - b analog voltages from the MS15 to digital numbers in the computer.
   - c digital signals in the MS15 to digital numbers in the computer.
   - d digital signals in the MS15 to analog signals in the computer.

5. **The Virtual Control Laboratory software provides:**
   - a chart display of signals in the DC Motor and controller.
   - b a signal generator to provide the reference signal.
   - c various types of controller action.
   - d all of the above.

Chapter 2
Introduction to Control Systems

Objectives of this Chapter

Having completed this chapter you will be able to:

- Outline the objectives of control
- Indicate the width of application of control systems
- Describe what is meant by the 'Plant Model'

Equipment Required for this Chapter

- MS15 DC Motor Module
- AS3 Command Potentiometer
- CLIO Interface Module with PC Connection Lead
- System Power 90 Power Supply (or equivalent)
- 4 mm Connecting Leads
- PC running VCL Virtual Control Laboratory Software

2.1 The Objective of Control Engineering

Control systems are all around us although they are not always obvious. In the domestic kitchen there are control systems in the washing machine (water temperature, water level, drum speed, spin speed), refrigerator (freezer and fridge temperatures), oven temperature... . Elsewhere in the house there can be three control loops in a central heating system, tape and head speed controls in a video tape recorder, platter speed and head position in CD-ROM, hard and floppy disk drives in a computer. There are also a number of purely electronic control systems in radios, televisions and sound systems. Then there are the control systems in cars, ships, aircraft... .

The objectives of control engineering will be demonstrated by some examples using the DC motor as a speed control system - the sort of system used to control tape and head rotation speed in a Video Recorder.

Wire up the system as in Fig 1.6 shown in Chapter 1.

With the motor off or disabled, disengage the MS15 Output Potentiometer from the output shaft to reduce wear.

Start VCL software and **Load setup | CA06PE02**. This sets up the system as:

File	**Controller**	**Plant**	**Display**
CA06PE02	Open-loop	MS15 Analog	Graph
Signal Generator		**Graph**	
Signal	DC-Level	1 Input	ON
Level	50%	2 Position	OFF
Offset	0%		
Rate	10 msec	4 Velocity	ON
Reference	Internal		
DC Motor		**Output Potentiometer**	Disengage
Brake	0	**Command Potentiometer**	180°

The reference input is set to *Internal* which uses the internal signal generator to drive the system.

Switch ON and enable the motor. Nothing should happen.

The Controller is set to **Open-loop**. This means that there is no feedback control.

Reaching the desired output

We would like the motor to run at half speed. For this the tachogenerator output should be 2.5V (50% of 5V full scale value). What input voltage is then required? Find out by constructing an input/output table. Measure the input voltage (channel 1) and tachogenerator output voltage (channel 4) over the range of *Offset* values as shown in Table 2.1 and enter the values in your workbook.

Offset %	Input Voltage	Tacho Output Voltage
0		
10		
20		
30		
40		
50		
60		
70		
80		
90		
100		

Table 2.1 Relationship between drive input and velocity output

Disable the motor then plot your results in the graph in Fig 2.1 in your workbook. You will get a graph as shown overleaf in Fig 2.1.

It can be seen that there is not a linear relationship between input voltage and speed. To run the motor at half speed, the input has to be set to approximately 56% of its range.

With open loop, there is no guarantee that the actual speed will equal the set speed.

Many systems do work in open loop but it does not lead to good control.

The first objective of a control system is that the controlled output reaches the value desired of it.

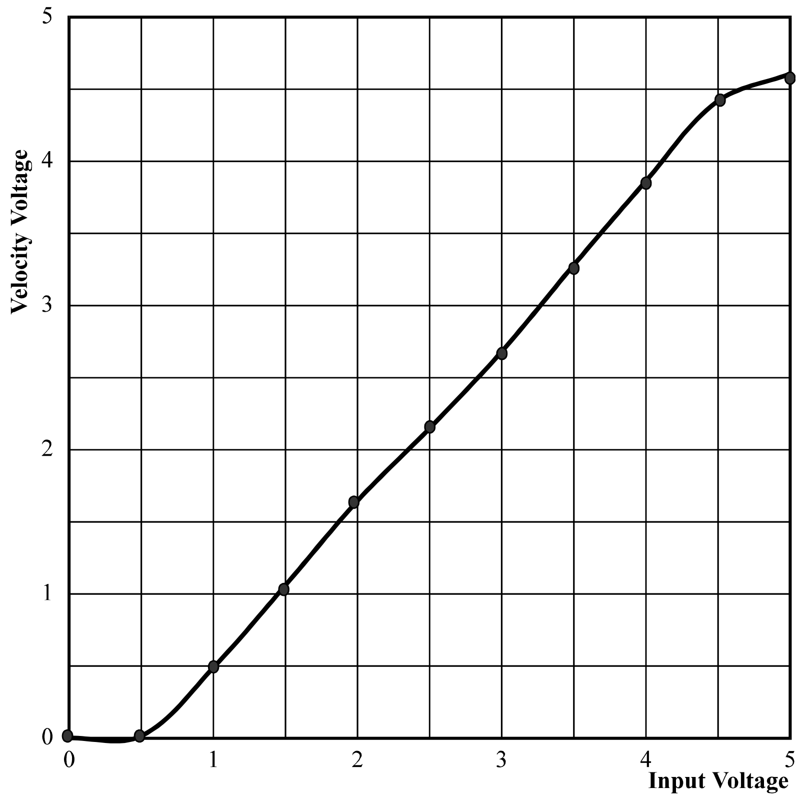


Fig 2.1 Drive vs speed under no load conditions

Reacting to Load Changes

Enable the motor and set the *Offset* to 50%. The brake is Off so this is the no load condition. Make a note of the tacho voltage in Table 2.2 in your workbook.

Now move the eddy current brake to position 1. You will see on the screen and on the motor rig that the speed drops as the motor is loaded. This is the half load condition. Again note the tacho speed in Table 2.2

Repeat for full loading with the brake in position 2. Then disable the motor.

Loading	Tacho voltage
No load	
Half Load	
Full Load	

Table 2.2 Change of speed with load

Not only is there a nonlinear relationship between input voltage and speed but the speed will change depending on the load.

If the control objective is to maintain a constant velocity under changing load conditions, then we cannot use open-loop control. A tape recorder operates under changing load as the tape is transferred from one spool to the other. So an open-loop control cannot be used to control a system subject to a changing load.

The second objective of a control system is that it maintains its output under changing load conditions.

Minimizing Transients

The third problem with a dynamic system is the time it takes to reach the required value from rest. We are all familiar with the comparison between cars based on the time to go from 0 to 60 mph (or 0 to 100 kph).

Put the brake back to 0, set the *Offset* to 0% and the *Signal* to **Step**. Adjust the *Level* until the velocity output reaches 50% (2.5V) at the end of the step period.

You are now asking the motor to reverse and seeing how long it takes the velocity to reach its new value. Measure the time from when the input changes until the output reaches 2V. The Time expansion controls may help you with this measurement.

Open-loop no load transient time to 2.5V = [secs]

Owing to the inertia of the motor and other factors, it takes some time for the motor to react to a change in demand. The time it takes to make a change is called the Transient Time.

The third objective of a control system is to reduce the transient time to as short as possible.

Fulfilling the objectives using feedback

Reaching the required level

From the tool bar, select **Controller | PID**. You can see the changes this has made from the controller mimic. The velocity output is now being fed back and compared with the input. This is a **Feedback Control system** or **Closed Loop Control system**. Do not be concerned with the controller detail - we will come back to this later.

After the transient period. the velocity and input voltages are now the same. You can see that the velocity trace (purple) now reaches the input trace (dark blue). The first objective has been met - the controlled value reaches the steady state value required of it.

Speeding up the system

Set the *Level* to 50%.

Now measure the transient time. Enter the value into your workbook and compare this closed loop transient time with the open-loop transient time measured earlier.

Open-loop no load transient time =	
Closed loop no load transient time =	

Adding the feedback controller has met the third objective - reducing the transient time of any change.

Reacting to load changes

Set *Signal* to **DC Level** and *Offset* to 50%.

Again examine the effect of changing the load by applying the brake. As the load is added there is a small decrease in the velocity then it recovers to its demanded level. When the load is removed, the motor speeds up then again recovers. The second objective has been met - the controlled value maintains its steady state value in spite of load changes.

Disable the motor.

You have seen that feedback control does work. Eventually you will understand how it works but before you can do that you must learn how to describe the behavior of the plant you are trying to control.

2.1a **Is the relationship between input voltage and rotational speed:**

[a] linear. [b] nonlinear.

2.1b **In a realistic control system, when the input is changed, the output should:**

[a] reach the demanded output immediately.

[b] reach the demanded output as soon as practical.

[c] never reach the demanded output.

2.1c **In a realistic control system, when the load is changed, the output should:**

[a] not change.

[b] return to its previous steady state value after a short transient time.

[c] never return to its previous steady state value.

2.1d **Introducing feedback control to a system:**

[a] increases the transient time.

[b] makes no change to the transient time.

[c] reduces the transient time.

2.1e **Which of these features of a home generally does not contain a closed loop control system:**

[a] washing machine. [b] oven.

[c] refrigerator. [d] toaster.

2.2 Plant Model

There becomes too many different entities described by the word 'system'. In this manual the word 'Plant' is used to describe the unit that has to be controlled. System will be reserved for the complete unit - Plant plus Controller.

All linear systems follow the same rules

In the illustrations above, you saw that the motor was characterized by how it responded to a Step input - **the Transient Response** - and how it behaved after the transient had died out - **the Steady State Response**.

The transient was of the motor speed in response to a change in drive voltage. If you applied the same step signal to a resistor/capacitor network as shown in Fig 2.2 below, you would get similar curves.

You also get similar curves if you measure the temperature after switching on an oven or the room heating of your home. The water level in a cistern after if has been flushed also follows this curve.

These are examples from mechanical engineering, electrical engineering, thermodynamics and hydraulics. All linear dynamic systems obey the same rules. Although the DC Motor is used as an example, the techniques being taught have applications in many other disciplines such as financial systems, management systems, learning systems and any dynamic system which can be described by the same differential equations which describe the behavior of engineering systems.

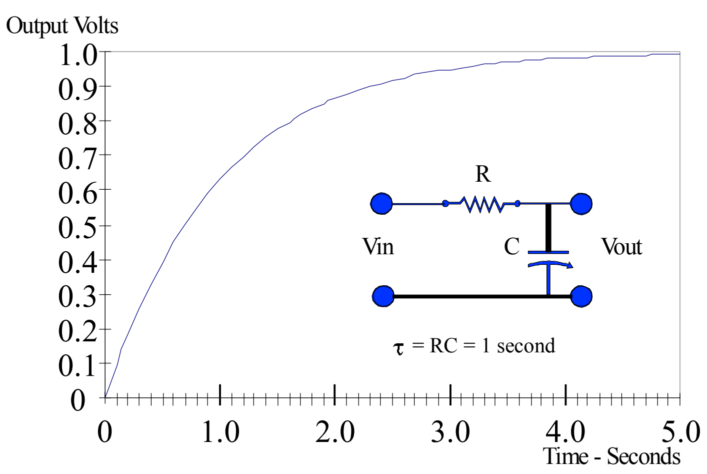


Fig 2.2 Step Response of a Simple RC Network

Characterizing a system by its response in time

Many systems are subjected to a step change in demand. Controlling the head position of the hard disk drive in the PC you are using is such an example. For these systems, the response in time is important.

There are other systems where the demand input is a ramp, not a step. Examples of these systems are Satellite Tracking Aerials and Scanning Radar systems. Here we are concerned with how the system tracks the steadily moving input in time.

For Step and Ramp inputs, the control engineer works with the Time Model of the system, which is how the system behaves in time after the signal is applied to it.

Characterizing a system by its frequency response

There are other systems where the demand input is much more complex. Consider the situation where a battle tank is trying to destroy a target while moving at high speed across rough terrain. The barrel must remain steady in space. The demand input in this case is a changing signal with no pattern to it - a Random Signal. The actual signal cannot be specified but the frequency range of the signal can be found and the system designed to cope with such a range of frequencies.

For Random or complex periodic signals, the control engineer works with the Frequency Model of the system, which is how the system behaves to a range of sine waves of different frequencies.

Developing the plant model

A control engineer needs to know how to determine both the Time and Frequency Models of systems and how to design controllers using these models.

The usual starting point is to investigate the behavior of the basic plant and, on the basis of the results, add whatever controller is required to improve the performance to a satisfactory level.

Transducers and signal conditioning devices have to be added to monitor events and produce signals suitable for manipulation. These cannot be 'perfect' and we must always seek to minimize errors that are introduced by our instrumentation devices but generally the dynamics of the monitoring equipment are inconsequential in the overall system.

Only a certain level of performance can be achieved. There are limits to the current that power supplies can deliver, the torque that motors can deliver, and so on.

Representing the system mathematically will only allow such effects to be included if they are accounted for in the analysis and, in all cases, involve a certain degree of compromise. During the course we will see instances where the actual behavior of the system does not agree with that predicted because of such effects. Initially such 'nonlinear' behavior will be ignored and only the **Linear Small Signal Model** will be considered.

The Linear Small Signal Model is the behavioral description of the system when small changes are made which do not take the system into nonlinear operating areas.

The plant model is developed in two ways - analytically or experimentally.

The analytical approach analyses the plant and from this develops a mathematical model. For the DC motor, we could calculate the inertia of all the rotating parts, assess the viscous friction of the bearing, calculate the current/torque relationship, determine the inductance and resistance of the field coils and, from all this, develop a model.

The experimental approach treats the plant as a 'Black Box', such that it has inputs and outputs but nothing is known, or needs to be known, of what actually happens inside. All our knowledge of the plant is gained from changing the inputs and observing the resulting outputs.

In this course the experimental approach is favored but what knowledge we have of the plant is used to our advantage.

A Plant Model is a description of how a system behaves. It is stated in terms which allow the determination of the steady state and transient performances of the plant. This allows a control engineer to formulate a control scheme which will result in the satisfactory performance of the overall system.

2.2a **A linear dynamic system model applies to:**

- a electrical systems only.
- b mechanical systems only.
- c any linear dynamic system.

2.2b **All linear dynamic systems obey the same rules. This means that a control engineer:**

- a must know the detailed engineering of many types of system.
- b only needs to know how a system behaves, not how it works.
- c does not need to know anything of the plant to be controlled.
- d needs to know how the system behaves but is helped by knowing how it works.

2.2c **The linear small signal model is valid:**

- a under all conditions.
- b for small changes only.
- c for large changes.

2.2d **If you wished to determine the transient response of a plant, you would use:**

- a the time model.
- b the frequency model.

2.2e **If you wished to determine the random response of a plant, you would use:**

- a the time model.
- b the frequency model.

Student Assessment 2

1. Which of the following is not an objective of a control system:

- a the output should reach its demanded value.
- b the output should react to input changes in a minimum time.
- c the output should recover to its steady state value if the load changes.
- d the output must remain steady in spite of load changes.

2. Control theory is applicable to:

- a engineering systems.
- b financial systems.
- c management systems.
- d all of the above and any system represented by linear differential equations.

3. Which of the following is not required in plant model for control purposes:

- a the transient behavior.
- b the steady state behavior.
- c the details of the plant operation.
- d power consumption of the plant.

Chapter 3
Time Response

Objectives of this Chapter

Having completed this chapter you will be able to:

- Measure the parameters of a plant using step tests
- Describe the characteristics of a first order lag
- State the time model of the DC motor

Equipment Required for this Chapter

- MS15 DC Motor Module
- AS3 Command Potentiometer
- CLIO Interface Module with PC Connection Lead
- System Power 90 Power Supply (or equivalent)
- 4mm Connecting Leads
- PC running VCL Virtual Control Laboratory Software

3.1 Step Response

We wish to determine a model which describes the time behavior of the plant using the 'Black Box' approach. To do this, you will ask the motor to change speed and infer the relationship between input voltage and output speed from the way in which the motor responds. You will be measuring the **Step Response** of the motor.

Fig 3.1 shows the block diagram of the motor with the parts used for a speed control system included within the shaded region.

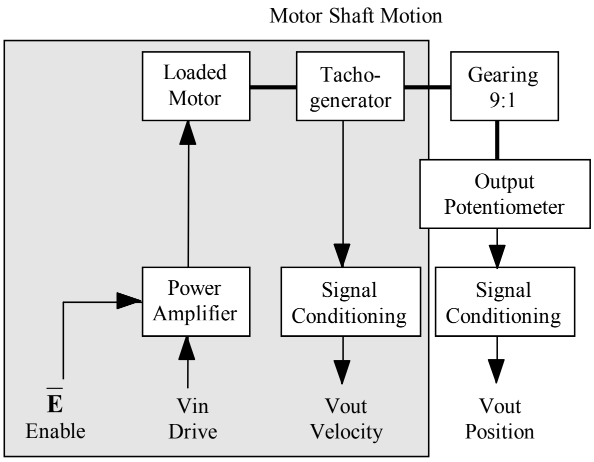


Fig 3.1 Block Diagram of the DC Motor

Behavior of the Plant in Time

The system should be wired with the standard analog system connections as shown in Fig 1.6 of Chapter 1.

Start the VCL software and **Load setup | CA06PE03**.

File	**Controller**	**Plant**		**Display**
CA06PE03	Open-loop	MS15 Analog		Graph
Signal Generator		**Graph**		
Signal	Step	1 Input	ON	
Level	60%	2 Position	OFF	
Offset	0%			
Rate	20 msec	4 Velocity	ON	
Reference	Internal			
DC Motor		**Output Potentiometer**		Disengage
Brake	0	**Command Potentiometer**		180°

Disengage the output potentiometer then switch power ON and **Enable** the motor.

The output velocity trace (purple) on the PC shows what is called the Step Response (what happens when there is a step change in the input).

There are two parts to any output time response when there is a change in input:

- A **Transient** period which occurs immediately the input changes and during which the system seems to be dominated by something other than the input.

- A **Steady State** condition which is reached after the transient has died out. The system seems to have settled down to the influence of the input.

The transient situation is produced by elements within the plant which cannot respond instantly. Mass in a mechanical system and capacitance in an electrical system both store energy so it takes time to change the velocity of a mass or to change the voltage across a capacitor.

In the DC motor, it is the mass of the motor armature and all the disks and dials connected to the motor shaft which require energy to get them moving or stop them moving. Actually it is the inertia of these elements, not mass, since we are dealing with rotating bodies.

The purple trace is the Step Response of motor speed. Observe that the speed does become constant after a time but initially lags behind the input. Expand the time scale by decreasing the *Rate* to 10msec and click the x2 time multiplier. Click **Freeze | Freeze**. This freezes the display at the end of the current cycle. The 'Frozen' control box appears when the cycle ends. The motor can now be disabled and measurements made from the screen.

You are going to measure the **Gain** and **Time Constant** which characterize the motor. Fig 3.2 overleaf shows the measurements to be made. The measurement facility is activated from the *Frozen* control box by clicking *Time ON*. The measurement lines and value boxes appear on the graph.

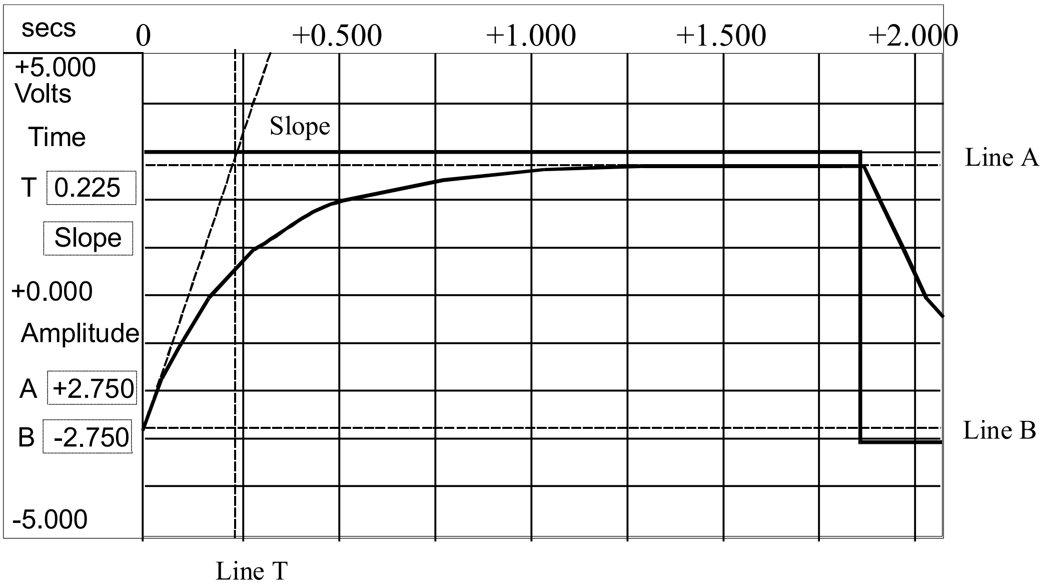


Fig 3.2 Sketch of output showing measurements to be made

Steady State Response

Input Span

Input span is the amount by which the input changes.

- Select channel 1/Input/Dark Blue. The scale will show the input channel scale.
- Select Line A by clicking within the A box. The box and the line will change color.
- Move the mouse until the pointer is pointing at the upper dark blue trace in the graph area. Click the left button and line A will move to where you are pointing. You can click again if you did not position the line exactly the first time. The A box indicates the level of the line.
- Click in the B box and, in the same way, position line B over the lower part of the dark blue trace.

The difference between A and B is the **Input Span**.

Input Span = A1 - B1 = ______

Enter the results in your workbook.

Output Span

Output span is the amount by which the output changes in response to the input changes.

Change to channel 4/Velocity/purple and repeat the measurements on the purple trace. Line B should be positioned where the trace can be seen starting at the left of the graph.

Output Span = A4 - B4 = ☐

Enter these results in your workbook.

Gain

Gain, or **Magnitude Ratio** or **Amplitude Ratio**, is the ratio between input and output when they have reached a steady state. The spans have been measured when the output has reached a steady state so:

$$\text{Gain} = \frac{\text{Output Span}}{\text{Input Span}} = \frac{A4 - B4}{A1 - B1} = \boxed{}$$

The steady state relationship between input and output is characterized by the Gain of the plant.

Transient Response

There are a number of ways to characterize the transient response. These come under the general heading of 'Rise Time' but there are many different definitions of Rise Time. You will measure three different times then we will see how these are related.

Initial Slope Method

- Make sure that lines A and B are the final and initial values of trace 4 respectively.
- Click in the Slope box. The line from the beginning of the transient sloping up to the right has changed to blue. This allows you to measure the initial slope of the velocity trace. The slope of the line can be changed by clicking in the graph area. The top of the line will move to the time at which you clicked.

- Move the slope line until its slope is the same as that of the initial part of the transient, such that the blue line covers the initial part of the purple velocity trace line.
- Click in the Time box. The vertical time line is highlighted.
- Click where the slope line crosses line A. The time shown is the **Time Constant** measured by the initial slope method.

Time Constant t_1 = | seconds

Settling time method

The time constant can also be calculated from the time it takes the transient to reach the final value.

- Move the Time line to the time at which the velocity trace first reaches its final value (when the purple trace reaches line A).

The time shown is 5 time constants from the start of the transient

Time Constant t_2 = | seconds

63% Method

Another time measurement is the time it takes for the transient to change by 63%.

From above, the output span = A4 - B4. The 63% level is then:

B4 + 0.63 (A4 - B4) = | volts

Use the values you have measured to calculate the 63% level for your experiment.

- Click the A box to highlight Line A and move it to the 63% level. You may not be able to set the line exactly owing to the screen resolution. Expanding the scale using the *Magnify* and *Shift* controls may help. The traces require to be redrawn using **Freeze | Redraw** option after *Magnify* or *Shift* are changes.
- Now click the Time box and move the time line to the time at which the velocity trace reaches its 63% level. The time shown is the **Time Constant** measured by the 63% method.

Time Constant t_3 = | seconds

t_1	t_2	t_3	Input Span	Output Span	Gain
ms	ms	ms	volts	volts	

Table 3.1 Step Response Results

Experience has shown us that the 63% measurement is more accurate than the other two techniques so use t_3 as the time constant in your model.

Enter your model gain and time constant into Table 3.2 in your workbook

Plant Gain K_p	
Time Constant τ	msec

Table 3.2 Motor model parameters

3.1a **The transient response is that part of the response curve which:**

[a] is determined by something other than the input signal.

[b] is determined by the input signal.

3.1b **The steady state is that part of the response curve which:**

[a] is determined by something other than the input signal.

[b] is determined by the input signal.

3.1c **The span of a signal is:**

[a] the value of the signal.

[b] the difference between the initial and final values of the signal.

[c] the ratio of the output value to the input value.

3.1d **If the input signal goes from 1 to 2 volts and the output signal goes from 5 to 10 volts, what is the system gain?**

3.1e **Which of the following cannot be used to measure the time constant?**

[a] Initial slope of the transient.

[b] Final slope of the transient.

[c] Time to reach the final value of the transient.

[d] Time to reach 63% of the output span.

3.2 The First Order Lag

The step response obtained is characteristic of a **First Order Time Lag**. A first order lag produces an exponential rise to a step input. Mathematically this is of the form:

$$\text{Change in Output} = \text{Change in Input x Gain x} \left[1 - e^{-t/\tau}\right] \qquad \text{Eqn 3.1}$$

where e is the exponential (or natural) number 2.7183... . τ is called the **Time Constant** and, with the **Gain**, fully characterizes a first order lag. t_1, t_2 and t_3 are measured estimates of this time constant.

To check that the measured step response is an exponential, or close to it, the computer can plot the response of an ideal curve over the measured one.

Click on **Plant | Servo**. The Plant has changed to a simulation of a servomotor such as the MS15. In the top box of the plant area, set K_p equal to the gain you have measured. In the next lower box enter the value of time constant measured (in milliseconds) then click in the *Overlay* box alongside the plant area. Note that for the overlay feature to be of use the magnify feature must be set to x1.

This will show you the measured response of the motor (purple) and the response of an exponential (light green) having the Gain and Time Constant you have measured. The values of gain and time constant can be changed and the graph redrawn until you have a good fit between the experimental data and the response of the theoretical model. The values of Gain and Time Constant set are the parameters which can be used to model the plant.

The two traces will not be an exact match owing to nonlinearities in the electronics and mechanics (such as deadband in the drive amplifier, or static bearing friction) but should be close enough for you to see that the response of the motor speed to a step input can be represented by an exponential function.

> The two parameters that define the model are
> **Gain** and **Time Constant**.
>
> **Gain** (K) is the Steady State relationship between input and output.
>
> **Time Constant** (τ) defines the Transient Time.

From Table 3.2, the control model parameters for the motor, under no load conditions are:

Gain (K) = **Time Constant (τ) =** **seconds**

3.2a **Is there an adequate match between the measured motor response and the response of the theoretical model?**

Yes or No

3.2b **Which of the following would not contribute to the discrepancies between actual and theoretical results?**

a Static friction in the bearings.

b Viscous friction in the bearings.

c Error in measuring gain and time constant.

d Deadband in the drive amplifier.

3.3 Characteristics of the First Order Exponential Lag

You now know what the exponential curve looks like but how does knowing the time constant give you a picture of the time behavior of the transient curve?

Value after 1 or more time constants

Use the Windows calculator on the computer to calculate the value of the curve. Do this at multiples of the time constant.

Using the keying sequence given below, calculate the values of $\left[1-e^{-t/\tau}\right]$ when t/τ = 0, 1, 2, 3, 4 and 5 and enter the values in Table 3.3.

An exponential is the inverse of the Natural Logarithm ln (ln $\equiv \log_e$). To calculate the value the curve has reached after 1 time constant, t = τ or t/τ = 1, use the Windows calculator key sequence:

1 +/- Inv Ln +/- + 1 = This calculates $\left[1-e^{-1}\right]$

This gives the answer 0.632... . This is where the 63% figure used earlier came from. After 1 time constant, an exponential response to a step has covered 63.2% of its total span.

The same keying sequence, but beginning with the other time ratios shown (0, 2, 3, 4 and 5), can be used to calculate the exponential values after 0 and 2 to 5 time constants.

t/τ	0	1	2	3	4	5
$\left[1-e^{-t/\tau}\right]$						

Table 3.3 Step Response of an exponential lag

From these figures, you can see that a step response will be at a value which is 32.8% (100 - 63.2) of its span away from its final value after 1 time constant, and 0.7% of its span away from its final value after 5 time constants. It can therefore be assumed that the transient has died out and the response has reached its **Steady State** value after 5 time constants.

There is an easier keying sequence. The first result gave us the value at the first time constant as 0.632 = 1 - 0.368. The value at the nth time constant is $1-0.368^n$.

Initial Slope

One of the characteristics of an exponential is that a line drawn at the initial slope crosses the final value of curve after one time constant. This was used as one of the methods of measuring the time constant of the motor. It can also be used to sketch an exponential curve without calculating lots of points along the curve.

Sketching an exponential

We want to show the shape of a response which has the form $\mathbf{Y = A\left[1-e^{-\frac{t}{\tau}}\right]}$

A is an amplitude multiplier. If it is assumed that A = 1 then the amplitude scale need only be multiplied by the actual value of A.

$\frac{t}{\tau}$ is the time multiplier so it is assumed that $\tau = 1$ and the time scale is multiplied by the time constant to give the actual time curve. This process is called normalization.

Fig 3.3 shows a sketch of the normalized exponential. Using the following procedure, you can sketch the exponential in your workbook.

- Draw a line from the starting point normalized amplitude value = 0 (at normalized time t = 0) to value = 1 (at t = 1). This is the initial slope of the normalized exponential curve.
- Mark the 63% point (value = 0.63) at t = 1 (the curve has covered 63% of its span, and is therefore 0.37 away from its normalized final value of 1). The curve will pass through this point.
- Draw a straight line from the 63% point (at t = 1) to normalized amplitude value = 1 (at t = 2). This line is the final slope of the first section of the curve (from t = 0 to t = 1) and the initial slope of the second section (from t = 1 to t = 2).
- During this second section of the curve the curve will again cover 63% of the distance to its final value (which, with an initial value at t = 1 of 0.37 away from its final value, gives the value at t = 2 of 0.37 x 0.37 = 0.14 away from its final value (or normalized amplitude value of 1 - 0.14 = 0.86).
- Draw a straight line from 14% at t = 2 to 1 at t = 3. This is the final slope of the second section and the initial slope of the third section. Each section can be treated as if it is the first section of a new exponential. By the fifth section the changes are too small to be graphed.

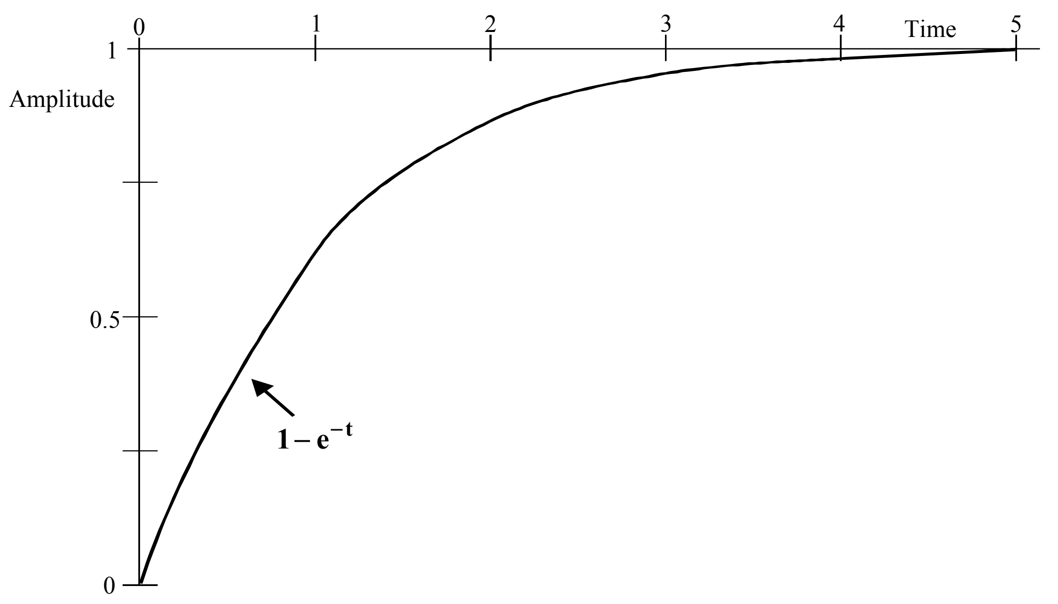


Fig 3.3 Normalized sketch of the exponential curve

3.3a **A plant has a gain of 0.8 and time constant of 3 seconds. Using the normalized sketch of a step response, determine the output response (in volts) to a 2 volt step input after 1.5 seconds.**

3.4 Steady State and Transient Response

If you multiply out Eqn 3.1, you see that:

Change in Output = (Change in Input x Gain) - (Change in Input x Gain x $e^{-t/\tau}$).

Now $e^{-t/\tau}$ tends towards 0 as t increases so the above term containing $e^{-t/\tau}$ represents the transient response, and the other term represents the steady state response. The Gain determines the Steady State Response. Our equation can therefore be re-written as:

Change in Output = Steady State Response – Transient Response

3.4a **The steady state output is determined by:**

a the plant gain only.

b the plant time constant only.

c both the plant gain and time constant.

3.4b **The transient response is determined by:**

a the plant gain only.

b the plant time constant only.

c both the plant gain and time constant.

3.5 What Contributes to the Time Constant

In the case of the DC motor, there are two factors controlling the time constant - the inertia of the rotating parts and the viscous friction of the bearings.

Step Response with a different load

The brake has the effect of increasing the friction. Unfreeze the display by selecting **Freeze | Start** then select **Plant | MS 15 Analog**. Set the eddy current brake to position 2 and repeat the gain and time constant measurements.

Unloaded gain =		Unloaded time constant	=	ms
Loaded gain =		Loaded time constant	=	ms

Your results should show that both the gain and time constant are changed. Both changes are due to increased frictional losses.

3.5a **Applying the eddy current brake:**

[a] increases the gain. [b] decreases the gain.

[c] has no effect on the gain.

3.5b **Applying the eddy current brake:**

[a] increases the time constant. [b] decreases the time constant.

[c] has no effect on the time constant.

3.6 Position Response

You may have noticed that, so far in this chapter, it is the speed that has been measured, not position. There is a good reason for this.

With the motor disabled, engage the output potentiometer. Unfreeze then enable the motor and click channel 2/Position/ON. This shows the position output and does not make much sense. There are a number of difficulties in measuring the position of the servo shaft while driving speed.

- The output potentiometer only measures position over 1 revolution so, as the dial goes through 360°, the trace jumps from top to bottom (or vice versa) of the graph.
- During the transient period there may be odder looking curves as the transient goes over the dead spot in the potentiometer.

The reason for this is that position is the integral of velocity so that, with a constant velocity, the position keeps changing. It can be difficult to measure something that is constantly moving which is why velocity was used to identify the plant time constant.

Although it can make identification difficult, you will see later that the integral effect makes servo control easier.

To be able to model the plant it is necessary to know the relationship between the velocity measured by the tachometer voltage and rate of change of position measured by the potentiometer voltage This relationship is called the **Integral Gain K_i**.

With the motor disabled, change the settings to those shown below.

File CA06PE03	**Controller** Open-loop	**Plant** MS15 Analog	**Display** Graph
Signal Generator *Signal* *Level* *Offset* *Rate*	 DC Level 60% 40% 10 msec	**Graph** 1 Input 2 Position 4 Velocity	 ON ON ON
Reference	Internal		
DC Motor **Brake**	 0	**Output Potentiometer** **Command Potentiometer**	Engage 180°

Enable the motor. The motor will run at a constant speed with the potentiometer output ramping up then returning to the bottom of the screen.

Make a note in your workbook of the velocity as shown on the red LED display.

Velocity = | **rpm** |

Set the timebase multiplier to x2, freeze the picture and switch the time markers ON. The graph will look like that shown in Fig 3.4.

The rate of change of position is measured by measuring the time it takes for the position trace (blue) to go from the bottom of the screen to the top.

- Select channel 2 then set Line A to +4.000 volts and Line B to -4.000 volts. Using the Time line, measure the times at which the output ramp crosses the two voltage markers. Enter these values into Table 3.4. of your workbook.
- $Slope = \frac{A - B}{T2 - T1}$ volts/second.

- Select channel 4 and use a voltage line to measure the tachometer voltage V_{vel}. (purple trace)
- Calculate the integrator gain K_i = Slope/V_{vel} volts per second per volt.

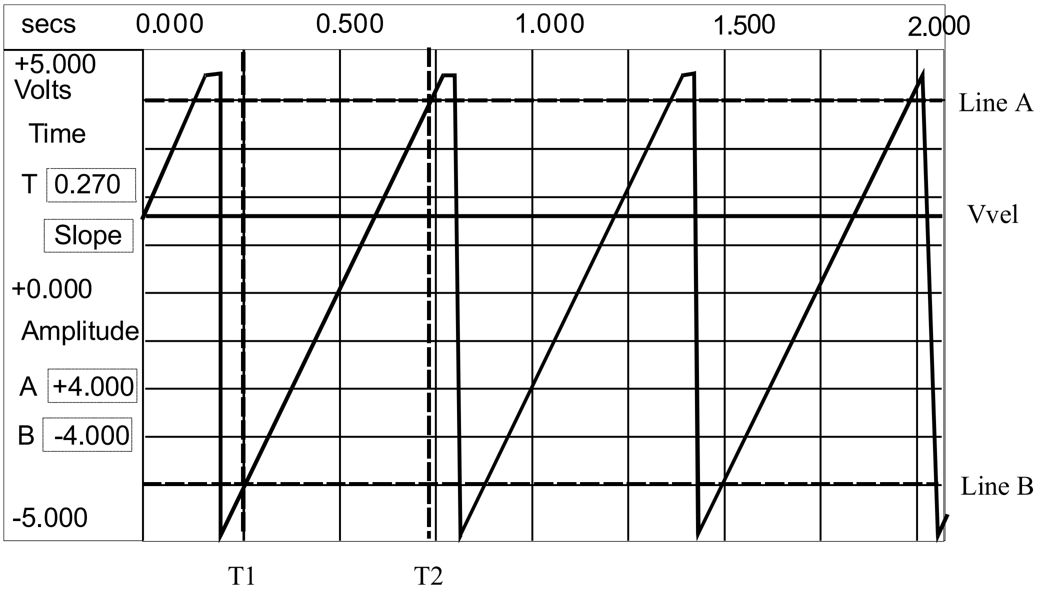


Fig 3.4 Determining the Relationship between Velocity and Position.

A Volts	B Volts	T1 secs	T2 secs	Slope volts/sec	V_{vel} volts	K_i
+4.000	-4.000					

Table 3.4 Relationship between velocity and position

A tachometer output of V_{vel} volts will produce a rate of change of position of K_i x V_{vel} volts/second.

3.6a **Position is the:**

[a] rate of change (or derivative with respect to time) of velocity.

[b] integral (with respect to time) of velocity.

3.6b **Which of the following is not true - Position is not used for identification of a servomotor because:**

[a] the limited range of the output potentiometer makes it difficult to measure the transient.

[b] it is difficult to measure the transient on top of the output ramp especially when it crosses the dead spot of the output potentiometer.

[c] noise on the output makes it difficult to see the transient.

3.7 Relationship between voltages and the parameters they represent

The model developed is based on the voltages being measured and it is this model on which all the calculations are based but it is also necessary to know the relationship between the voltages being measured and what is really happening in the plant. That is, what is the relationship between tachometer output volts and the rotational speed of the output in Revolutions per Minute (RPM) or Degrees per Second?

This can be found from other measurements on the graph already on the screen. You already know the tachometer output voltage so now you need to measure the actual rotational velocity this measures. Fig 3.5 below shows the measurements to be taken.

T1 secs	T2 secs	V_{vel} volts	T2-T1 secs/rev	revs/sec	revs/min	RPM	K_r

Table 3.5 Rotational Velocity

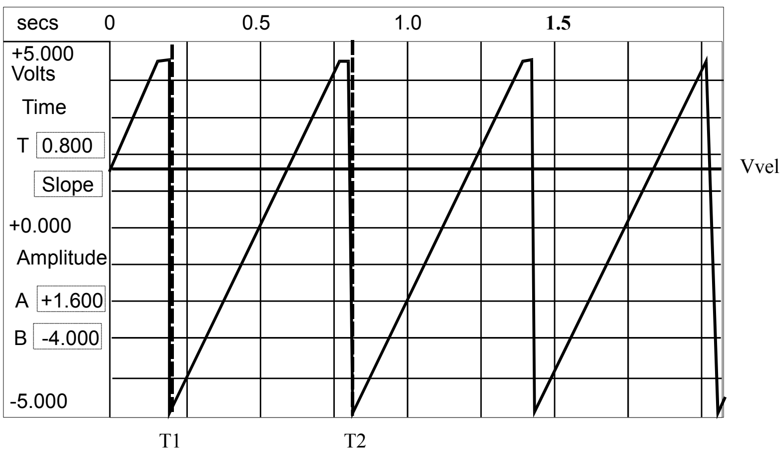


Fig 3.5 Measuring actual rotational velocity.

Revolutions/second = 1 / (T2 - T1). Revs/min = 60 / (T2 - T1). RPM is the display measurement made earlier and should agree with the revs/min measurement just made. K_r is the conversion ration between V_{vel} and RPM. $K_r = RPM/V_{vel}$ so that:

RPM = K_r x V_{vel} = [x V_{vel}]

In Chapter 1, you have already obtained the relationship between output voltage and degrees.

Degrees = (K_d x V_{pos}) + Degrees Offset = [ ]

3.7a **Determining the velocity of the output shaft by measuring the time for one cycle gave a result which agreed with the tachogenerator output on the motor rig.**

Yes or No

3.8 The Plant Model In Time

Fig 3.1 can now be redrawn as Fig 3.6 to show the model that has been identified. Fill in the figures for your measurements in your workbook version of this diagram. Your figures for your model should be close to those shown below.

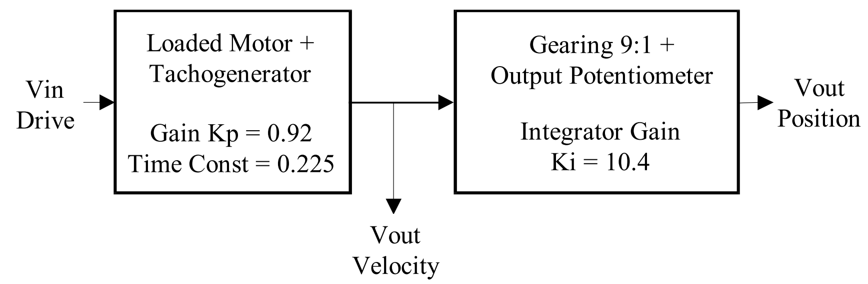


Fig 3.6 Measured Time Model of the DC Motor

3.8a **A system is characterized by a 1st order lag with a gain of 5 and a time constant of 2 seconds. If the input steps from 0 to 1 volt, what is the output (in volts) after 1 second?**

3.8b **In a test, the tachogenerator produced a voltage of 3.5 volts and the integral gain K_i was measured to be 10. What would be the expected rate of change of the output position (in volts/second)?**

3.8c **In the same test as question 3.8b, what would be the output RPM if the conversion ratio $K_r = 50$?**

3.9 Other Time Signals and Methods of Identification

There are two other standard time signals used in the study of control systems. Neither is very useful in system identification but they are used, as will be shown later, when studying system behavior. The two signals are **Impulse** and **Ramp**.

Impulse

A short sharp jab to the system. Theoretically the Impulse Response tells the same information as the Step Response but there is so little energy in the impulse that the plant hardly moves enough to see any response.

Unfreeze and Enable the motor. Select *Signal* - **Pulse**, *Level* - **100%** and *Offset* - ***0%***. Observe that the velocity does not change much and the response is very inconsistent. It would be difficult to measure the gain and time constant from these traces, especially if there was additional plant noise on them.

Ramp

Many systems have to be able to follow a ramp but the open loop ramp response is not easy to analyze. Motors tend to be nonlinear at low speed and this distorts the measured signal.

Select *Signal* - **Ramp**, *Level* - **60%**. A triangular shaped waveform is generated which ramps up at a constant rate then reverses to ramp down at the same rate. The velocity signal tries to follow the input ramp but lags behind. There is a considerable kink in the speed curve as the motor stops then reverses. This is due to nonlinearities - static friction in the motor and deadband in the drive amplifier.

Noise in the System

You may have noticed that the velocity signal is not exactly constant when the motor is asked to run at a constant rate. The signal is said to be noisy. As far as plant signals go, the motor signals are very quiet. Real plant is often much noisier than this. One of the problems can be to extract meaningful data out of the noise. There are mathematical techniques that can be used for this.

If the input signal itself is normally noisy, this noise signal can be used to identify the plant but again this involves considerable calculation.

Sometimes putting step changes into a plant is not acceptable to the plant operators. In these cases, low level artificial noise can be introduced which will not interfere with the plant operation but which will allow mathematical techniques to be used to identify the plant. This technique, called Pseudo Random Testing, is often used in the identification of slow plant where the time constant is measured in minutes, hours or even days.

Student Assessment 3

1. **The lag in the motor step response in caused by:**
   - [a] the inertia of the moving parts only.
   - [b] the viscous friction of the bearings only.
   - [c] both inertia and viscous friction.
   - [d] neither inertia nor viscous friction.

2. **Which of the three time constant measurements would seem to be the most accurate?**
   - [a] initial slope.
   - [b] 63%.
   - [c] time to final value.
   - [d] 10%-90%.

3. **After how many time constants can the transient period be said to be over:**
   - [a] 3.
   - [b] 4.
   - [c] 5.
   - [d] 6.

4. **Which of these signals is also used to determine the time model of a plant:**
   - [a] impulse.
   - [b] ramp.
   - [c] noise.
   - [d] sinusoid.

Chapter 4
Frequency Response

Objectives of this Chapter

Having completed this chapter you will be able to:

- Measure the parameters of a plant using frequency tests
- Describe the frequency characteristics of a first order lag
- Explain why Bode Plots are used in preference to other frequency plots

Equipment Required for this Chapter

- MS15 DC Motor Module
- AS3 Command Potentiometer
- CLIO Interface Module with PC Connection Lead
- System Power 90 Power Supply (or equivalent)
- 4 mm Connection Leads
- PC running VCL Virtual Control Laboratory Software

4.1 Frequency Analysis in Control Engineering

Control Engineering is the amalgamation of three different strands of development.

Process Engineering is concerned mainly with running plant at steady conditions so is concerned with maintaining an output against load fluctuations and other disturbances.

Mechanical Engineers have been concerned with the dynamics of vehicles and their suspension systems so have focused mainly on the time performance of systems.

Electrical Engineering grew out of telephony. Speech is a complex waveform but the human ear has a limited frequency range (around 15 kHz). Electrical Engineering developed a strong frequency bias and the branch of Control Engineering called Servomechanisms was developed from a frequency perspective. There are a number of analysis and design techniques which are based solely on the Frequency Response of the plant. These are graphical techniques and require only a knowledge of the measured frequency response of the plant to be controlled.

4.1a **Frequency analysis of control systems has been developed from:**

[a] mechanical engineering. [b] electrical engineering.

[c] process engineering.

4.2 Frequency Response

Sine waves are naturally occurring phenomena. Pluck a guitar string and it vibrates sinusoidally. A 'pure' musical tone is a sinusoid. Middle C is a sinusoid vibrating at 261.63 cycles per second. The S.I. unit is the Hertz (Hz = cycles per second = revolutions per second) after the German physicist who first described the concept.

It can be shown that any signal can be made up from a series of sinusoids of different frequencies and amplitudes so that there is a definite mathematical relationship between the frequency composition of a signal and its shape in time.

For plant identification purposes, the nice thing about a sine wave is that if you put a sine wave into a linear 'black box', you get out a sine wave of the same frequency but changed in amplitude and phase. Knowing the output frequency allows the signal to be extracted from noise using tuned filters or digital filtering techniques.

Fig 4.1 shows typical input and output sinusoids to and from a black box plant.

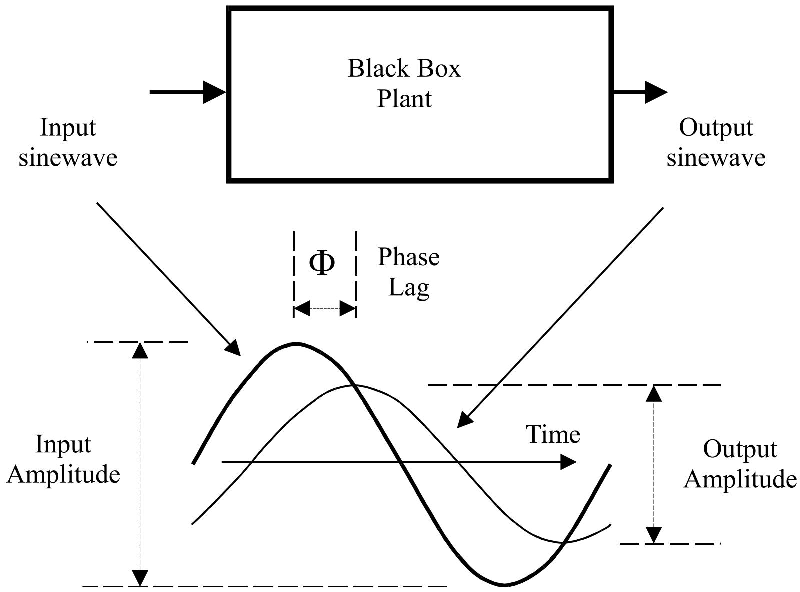


Fig 4.1 Relationships between input and output sinusoids

The output has a different amplitude from the input and the ratio:

$$\frac{\textbf{Output Amplitude}}{\textbf{Input Amplitude}}$$ is called the **Amplitude Ratio**.

From the diagram, it can be seen that the output sinusoid lags in time behind the input. This lag is measured in Degrees and is called the **Phase Lag ϕ**.

Phase Lag can be calculated from the time difference between the peak of the two sinusoids. The Frequency of a signal (in Hertz, Hz) is the number of cycles in one second. The Period is the time to complete 1 cycle so is the inverse of frequency:

$$\textbf{Period (seconds)} = \frac{\textbf{1}}{\textbf{Frequency (Hz)}}$$

Since there are 360° in one cycle period:

$$\textbf{Phase lag } \phi = \frac{\textbf{360} \times \textbf{Time Lag}}{\textbf{Period}} = \textbf{360} \times \textbf{Time Lag} \times \textbf{Frequency}$$

Frequency Response of the DC Motor

You are going to measure the frequency response of the motor and from this determine the frequency model of the plant.

Load setup | CA06PE04. The setup is given below. Note that the *Rate* has been replaced by *Frequency* and that the horizontal scale of the graph is now is now in Degrees.

File CA06PE04	**Controller** Open-loop	**Plant** MS15 Analog	**Display** Graph
Signal Generator		**Graph**	
Signal	Sine	1 Input	ON
Level	60%	2 Position	OFF
Offset	0%		
Freq	100 mHz	4 Velocity	ON
Reference	Internal		
DC Motor		**Output Potentiometer**	Disengage
Brake	0	**Command Potentiometer**	180°

Disengage the output potentiometer, switch ON and Enable the plant.

Following the procedure below, measure the amplitude and phase change of the speed signal over the range of frequencies shown in Table 4.1. Enter the measured values into the table in your workbook then calculate the amplitude ratio. Disable the motor after all frequencies have been measured.

Start at 100 mHz then go down the table then come back to 50 mHz and complete the low frequency tests.

At each frequency:

- Freeze the display using the **Freeze | Freeze** option. Remember that the graph is not frozen until it has completed drawing across the screen. At low frequency, do not initiate the freeze until the trace has began a cycle at the left of the graph area.
- When the **Frozen** message appears, click *Frequency ON*. Measurement boxes appear in the scale area.
- Measure the peak-peak voltage by placing lines A and B on the maximum and minimum velocity values. The scale must be set to channel 4/velocity. The input peak-peak can be found by measuring the input/dark blue trace. The input need only be measured at one frequency provided the signal generator level is not changed. The peak-peak value is A - B. At high frequencies the output amplitude will drop. Extra accuracy of measurement can be obtained by using the *Magnify* and *Shift* controls.

- Select Phase D by clicking in the D box. Move the vertical line until it intersects with the peak value. The reading in the box is the phase shift relative to 0° of the input trace. As you are measuring the phase shift at the peak, the phase lag will be the measured value minus 90°.

Input Amplitude = 6.0 Volts peak-to-peak

Frequency	Output (Volts p-p)	Phase Lag (Degrees)	Amplitude Ratio A	Amplitude Ratio $20\log_{10}A$ (dB)
10 mHz				
20 mHz				
50 mHz				
100 mHz				
200 mHz				
500 mHz				
1 Hz				
2 Hz				
5 Hz				
10 Hz				

Table 4.1 Frequency Response Test

The Amplitude Ratio is the ratio of Input to Output volts at any particular frequency.

These results may be plotted in a number of different ways but, for our purposes, the most useful is the **BODE PLOT** where the amplitude ratio and phase are plotted separately against log frequency. For reasons that will be explained later in this section, it is the logarithm of the amplitude ratio which is used, not the amplitude ratio itself. The amplitude ratio is converted into decibels (dB) by the formula:

$$\text{Amplitude Ratio (dB)} = 20\log_{10}\left[\frac{V_{out}}{V_{in}}\right]$$

Calculate the Amplitude Ratio in dB and enter this into your table. This table can now be used to draw the Bode Plot. Fig 4.2 is the Bode plot of the results in Table 4.1.

Preprinted Log/Linear graph paper can be obtained from technical stationers. A blank graph sheet is provided in your workbook for you to plot your results.

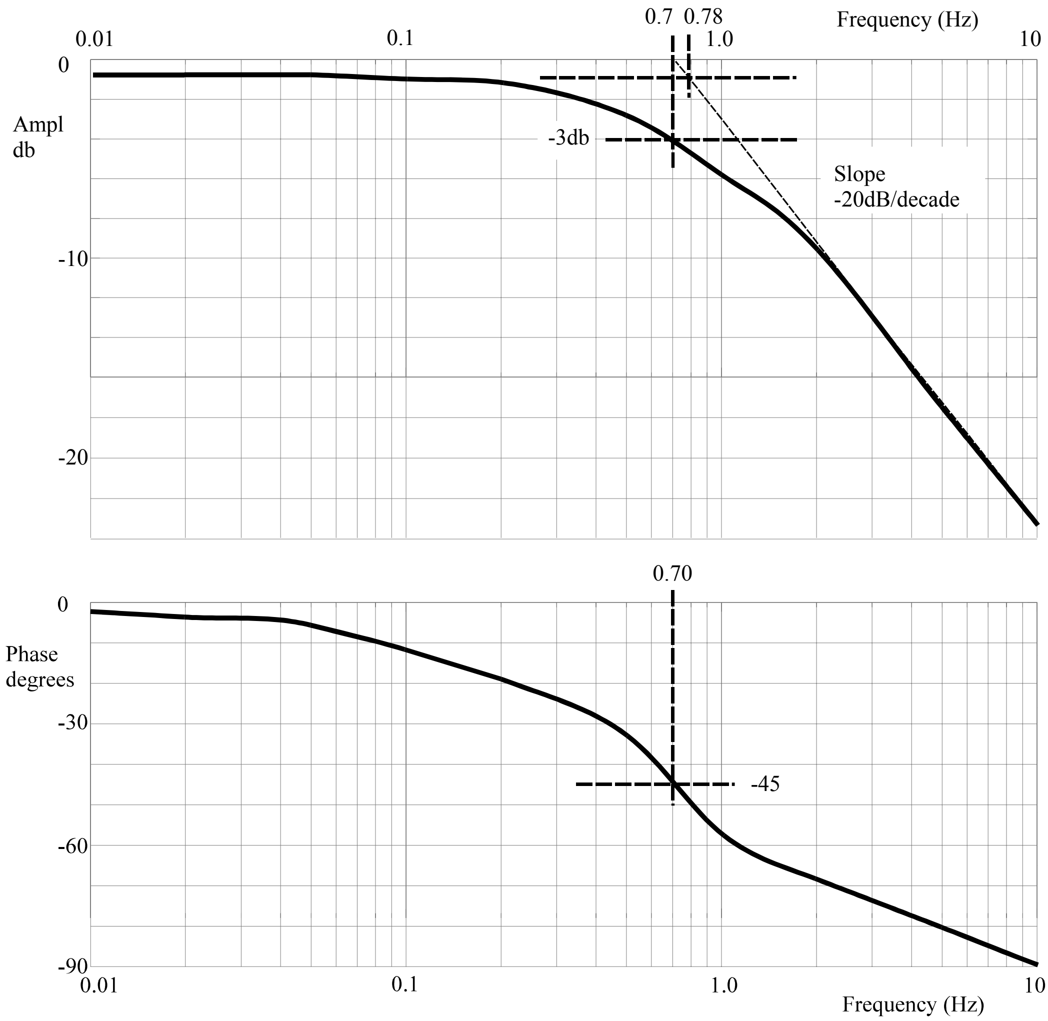


Fig 4.2 Bode Plot of the Frequency Response of the DC Motor

Amplitude Ratio

At low frequency, the Amplitude Ratio is the Gain (K_p) of the plant. At high frequency, the Amplitude Ratio drops linearly on the graph. The straight line actually drops at a rate of -20 dB/decade, such that every time the frequency increases by a factor of 10, the amplitude drops by 20 dB. On your graph, draw a straight line with this slope as a best fit to the high frequency section of the measured curve. This line is shown dashed in Fig 4.2. It is only on a Bode plot that the curve at high frequency becomes a straight line making it easy to determine the slope.

Also, draw a horizontal line to extend the low frequency part of the curve to higher frequencies. The frequency at which these two lines meet is called the **Cut-off Frequency** or **Break Frequency**. At the break frequency, the actual curve should be 3 dB below the low frequency level.

A first order lag is characterized, in frequency, by its low frequency gain and its cut-off frequency.

Phase Response

At low frequency, there is no phase shift. At high frequency, the Phase ϕ tends towards -90° (90° lag).

At the Break Frequency the phase ϕ is at -45°.

Finding the Break Frequency

- Draw a horizontal line through the low frequency amplitude points.
- Draw a line at slope –20 dB/decade through the high frequency points.
- The frequency at which these two lines meet is the break frequency.
- Adjust the lines if necessary to ensure that the actual amplitude curve is at –3 dB and the phase shift is -45° at the break frequency.

From the Bode plot in Fig 4.2, the break frequency has been determined by the three techniques. Enter your values into Table 4.2 in your workbook and calculate the average of the three readings.

From slope	From -3 dB point	From -45°	Average
Hz	Hz	Hz	Hz

Table 4.2 Cut-off Frequency

Radian Frequency

So far the frequency has been stated in Hz or Cycles/second. There is another measure of frequency called Radian Frequency. Radians are an angular measurement and there are 2π radians per cycle, such that 2π radians are equivalent to 360°. Radian Frequency has the symbol ω, where $\omega = 2\pi f$, and has dimensions radians/second.

Relationship between break frequency and time constant

Why use radian frequency? It can be shown that the Break Frequency, ω_c, in radians/second, is the inverse of the time constant, τ, in seconds, for example

Break Frequency $\omega_c = \mathbf{2\pi f_c} = \frac{\mathbf{1}}{\tau}$ or Time Constant $\tau = \frac{\mathbf{1}}{\omega_c} = \frac{\mathbf{1}}{\mathbf{2\pi f_c}}$

From Table 4.2, $f_c = 0.73$ Hz so $\omega_c = 2\pi \times 0.73 = 4.59$ rad/sec

$\therefore\ \tau = 1 / (4.49) = 0.22$ seconds

Gain K_p = Amplitude Ratio at low frequency = 0.92

Table 4.3 shows these results can be compared with those obtained from the time tests in Chapter 3.

Enter your results into the table in your workbook and compare the results obtained.

Test	Gain K_p	Time Constant τ
Time		
Frequency		

Table 4.3 Comparison of Time and Frequency Tests

4.2a **If you put a sinewave into a linear system you get out:**

- [a] a sinusoid of a different frequency.
- [b] a number of sinusoids at different frequencies.
- [c] a sinusoid of the same frequency but with different amplitude and phase.
- [d] a sinusoid of the same frequency, amplitude and phase.

4.2b **A 10 Hz sinusoid of 2 volt peak-peak amplitude is applied to a plant. The output amplitude is measured at 2.4 volts peak-peak. What is the Amplitude Ratio of the plant at this frequency?**

4.2c **A 10 Hz sinusoid of 1 volt peak-peak amplitude is applied to a plant. The time lag between input peak and output peak is 12 ms (0.012 seconds). What is the phase lag of the plant at this frequency (in degrees)?**

4.2d **Which of the following is not true. The cut-off frequency of a 1st order lag is the frequency at which:**

[a] the amplitude ratio is 3 dB below its low frequency value.

[b] the phase lags by 45°.

[c] the gain is 1.

[d] the high frequency slope of the amplitude ratio intersects the low frequency value.

4.2e **Do the frequency and time models obtained agree reasonably with each other?**

[Yes] or [No]

4.3 Accuracy of Frequency Tests

More accurate results could be obtained if more frequency points were tested but the three measures of cut-off frequency are consistent and agree with the time tests.

To show that the DC motor can be represented by a first order lag, Fig 4.3 shows the theoretical Bode plot of a first order lag having the gain and cut-off frequency measured. The measured curve is also shown. It can be seen that the first order lag is a good frequency model for the DC motor.

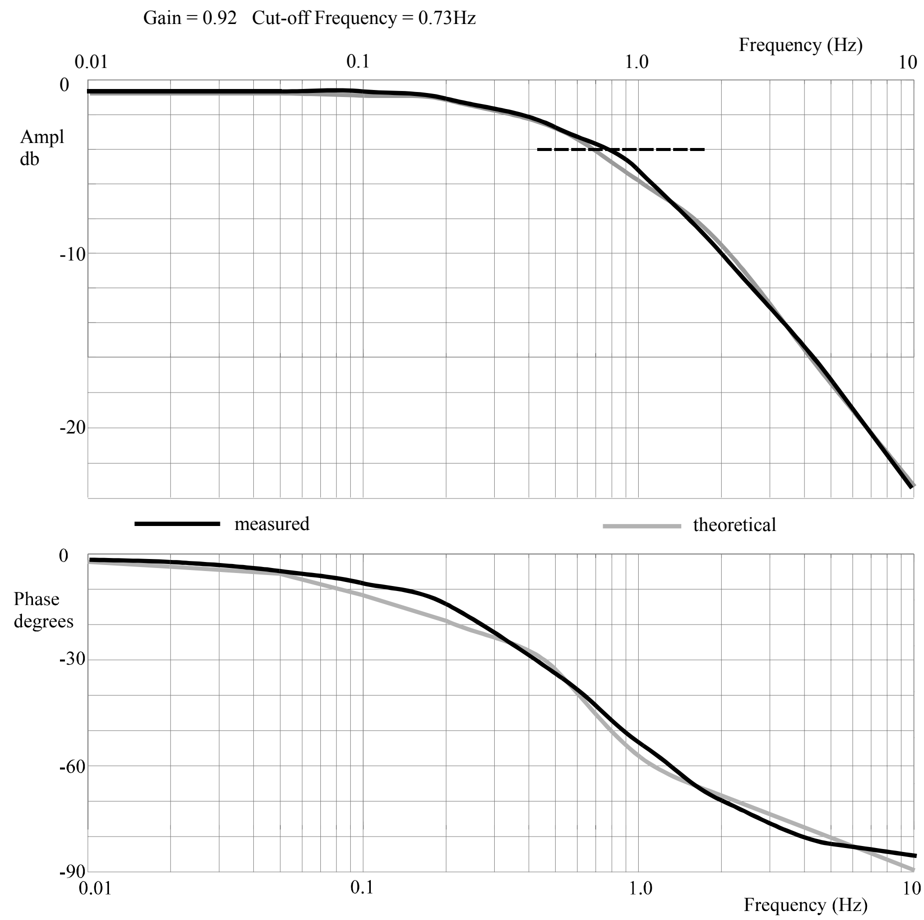


Fig 4.3 Bode plot of a 1st order lag

For the mathematically minded, the frequency response curves are described by the equations:

$$\textbf{Amplitude Ratio} = \frac{\text{Gain}}{\sqrt{1+\left(\frac{\omega}{\omega_c}\right)^2}} \text{ and } \textbf{Phase} = -\arctan\left(\frac{\omega}{\omega_c}\right)$$

where ω is the radian frequency, and ω_c is the cutoff radian frequency.
[arctan ≡ $\tan^{-1}$ or Inv tan on the Windows calculator]

4.3a **Can the DC motor be adequately modeled by a first order lag:**

Yes or No

4.4 Usefulness of Bode Plots - Building Complex Responses

It is a characteristic of logarithms that the log of two numbers multiplied together is the sum of the log of the two individual numbers, for example:

$$\log(A \times B) = \log A + \log B$$

If a plant consists of two first order lags one after the other (in series) then the output is the multiple of the lags. Since Bode is a logarithmic plot, the Bode plot of two lags is the sum of the two individual Bode plots. This is true for both amplitude and phase.

Parts of a plant can then be measured separately and the overall Bode frequency response is the sum of individual responses.

Having built up an overall Bode Plot in this way, it is then quite easy to translate the information into any of the other graphical forms used in control engineering design.

4.4a The frequency response of a complex plant can be determined from the frequency responses of individual parts by:

[a] adding the individual Bode amplitude plots and the phase plots.

[b] multiplying the individual Bode amplitude plots and the phase plots.

[c] adding the individual Bode amplitude plots and multiplying the phase plots.

[d] multiplying the individual Bode amplitude plots and adding the phase plots.

4.5 Usefulness of Bode Plots - Identification of Complex Responses

Just as the response of a complex plant can be built up from the knowledge of the responses of the component parts of the plant, so a complex plant can be identified from knowledge of the rules of Bode Plots. Each lag adds -90° to the phase lag and a slope of -20 dB/decade to the amplitude response at high frequency. This can be used to determine the different time constants for higher order lags. Bode plots are the only form of the frequency response which can be used in this way to identify the parameters.

Frequency response identification is useful in electronic circuits but tends to be difficult in industrial plant where the sine waves get swamped by noise. Electronic instruments called *Transfer Function Analyzers* can be purchased which can perform frequency response tests very quickly. These are useful for electro-mechanisms where the break frequency is greater than 1 Hz but generally are not available for slow processes where the time constant is measured in minutes or hours. You can also appreciate that it would take a long time to frequency test such slow plants. Time testing using noise analysis techniques is generally used for such slow processes.

Student Assessment 4

1. **If a plant has a measured cut-off frequency of 2 Hz, what is the time constant τ of this plant (in seconds)?**

a 0.04	b 0.06	c 0.08	d 0.10

2. **The input signal is 5 volts p/p and the output is 0.5 volts p/p. What is the amplitude ratio in decibels?**

a 10 dB	b 20 dB	c -10 dB	d -20 dB

3. **The slope of the amplitude response of a first order lag at high frequency is:**

a 10 dB/decade	b 20 dB/decade	c -10 dB/decade	d -20 dB/decade

4. **1 radian is equivalent to:**

a 45°	b 57.3°	c 60°	d 90°

Chapter 5
Principles of Feedback

Objectives of this Chapter

Having completed this chapter you will be able to:

- Identify the significant parts of a feedback control system and manipulate transfer function blocks
- Derive the closed loop and error transfer functions from the forward and open loop transfer functions
- Describe the effect of closing the loop on the steady state performance, stiffness, response time and frequency response

Equipment Required for this Chapter

- MS15 DC Motor Module
- AS3 Command Potentiometer
- CLIO Interface Module with PC Connection Lead
- System Power 90 Power Supply (or equivalent)
- 4 mm Connection Leads
- PC running VCL Virtual Control Laboratory Software

5.1 Introduction to Feedback

So far you have been measuring the characteristics of the plant without control applied. Now we shall see the effect of adding feedback.

Fig 5.1 is the block diagram of a general control system.

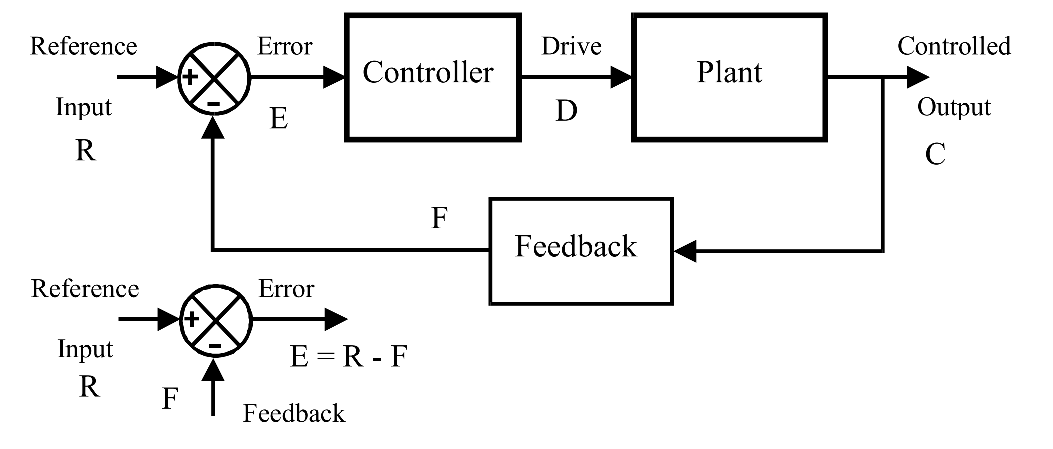


Fig 5.1 General Control System

There is a variable whose value is to be controlled (**Output** or **Controlled Variable - C**) and a variable which represents the value of the output required (**Input** or **Reference Variable - R**). This output is passed back via a feedback block to be subtracted from the input to form the **Error - E**. The error is manipulated by the **Controller** to provide the **Drive - D** which is the signal which tells the plant what to do.

By measuring what the output is doing and feeding this back to be compared with the input, a **Closed Loop Feedback Control System** has been created.

This chapter describes the performance of a simple plant (first order lag) with a simple proportional controller. You will see that this does not give very good control. Later chapters will show how controllers can be designed to give better control of simple and complex systems.

5.1a **The plant drive is created by passing the error signal through the:**

[a] feedback block. [b] controller block.

[c] plant block.

5.1b **The error signal is created by comparing the input signal with the:**

[a] feedback signal. [b] drive signal.

[c] output signal.

5.2 Transfer Function

To be able to analyze such a system, it is necessary to know the characteristics of each of the elements in the system. You have already seen how to measure the characteristics of the plant and the job of the control engineer is to decide on the characteristics of the controller and the feedback. A formal method of describing these elements and manipulating them is required.

The term **Transfer Function** is used to describe the relationship between the input and the output of a block and is usually denoted by the symbol **G**.

$$\textbf{Transfer Function G} = \frac{\textbf{Output}}{\textbf{Input}} \qquad \text{or} \qquad \textbf{Output} = \textbf{G} \times \textbf{Input}$$

Transfer Functions can be equations in time or frequency but all that is of concern at the moment is that the output can be calculated if the transfer function and input are known.

Blocks described by transfer functions can be connected in series (one after the other) as shown in Fig 5.2. The output of the first block is input to the second block, creating a series connection of the two blocks.

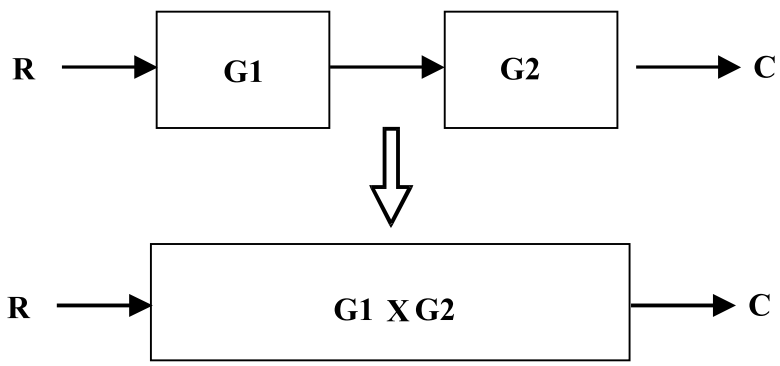


Fig 5.2 Transfer Function Blocks in Series

Transfer Functions of blocks in series are multiplied together.

If two blocks are in parallel, such that both have the same input and their outputs are summed together, then a parallel connection results, as shown in Fig 5.3.

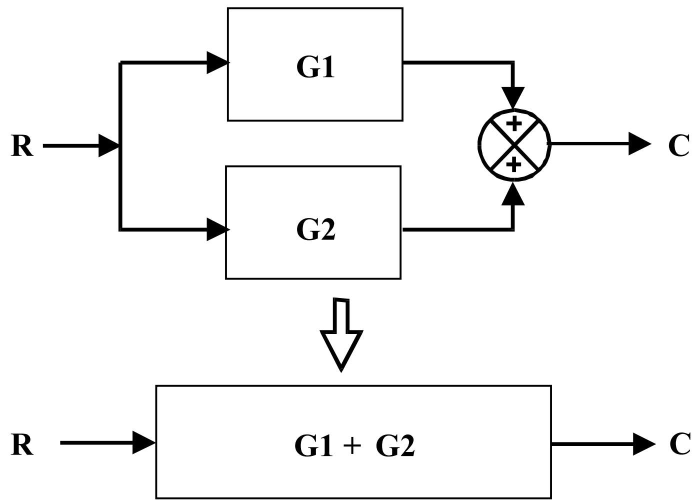


Fig 5.3 Transfer Function Blocks in Parallel

Transfer Functions of blocks in parallel are added together.

Remember that these relationships will only give the correct results if the dimensions throughout the system are consistent. For example (in a series connection), if the output of block 1 is a measurement in RPM then the transfer function of block 2 must have an input of RPM (and not, for example, Hertz) even though both are measures of rotational speed - there is a 60 times difference between the two.

A special term is used when the output is directly proportional to the input, such that there are no time constants involved. In this case the transfer function is termed the **Gain** which is given the symbol **K**.

Note: the gain can be less than 1 in which case it is sometimes termed **Attenuation**.

5.2a **If two blocks G1 and G2 are in series, the overall block is represented by:**

a G1 + G2 b G1 x G2 c G1 - G2 d G1 / G2

5.2b **If two blocks G1 and G2 are in parallel, the overall block is represented by:**

a G1 + G2 b G1 x G2 c G1 - G2 d G1 / G2

5.3 Closed Loop Transfer Function

Fig 5.1 has been redrawn slightly to give Fig 5.4. Each block has now been given a transfer function symbol. K is the transfer function of the controller, G is the transfer function of the plant and H is the transfer function of the feedback.

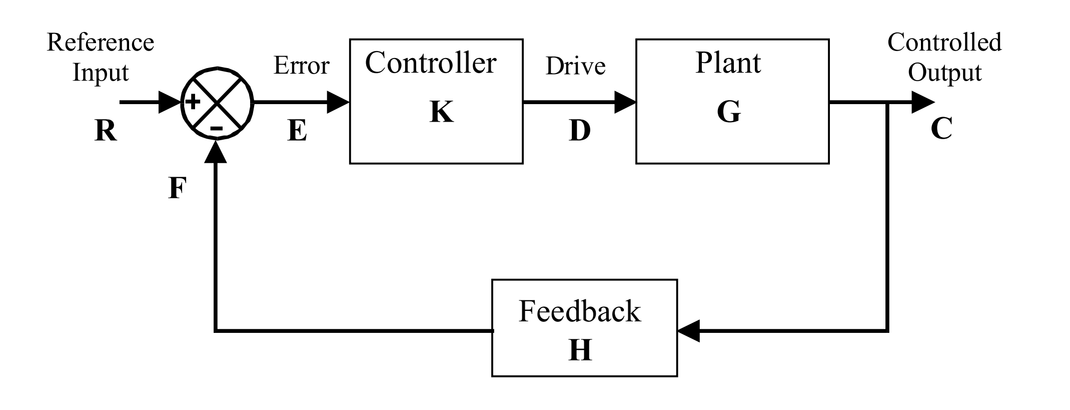


Fig 5.4 Block Diagram of a General Control System

Studying Fig 5.4 and applying the rules of block combination, it can be seen that:

$$\mathbf{Output(C) = K \times G \times Error(E) \quad \Rightarrow \quad C = K.G.E} \qquad \text{Eqn 5.1}$$

and that:

$$\mathbf{Error(E) = Input(R) - H \times Output(C) \quad \Rightarrow \quad E = R - (H \times C)} \qquad \text{Eqn 5.2}$$

Substituting Eqn 5.2 for Error into Eqn 5.1 gives:

$$\mathbf{C = K \times G \times E = K \times G \times [R - (H \times C)] = K.G.R - K.G.H.C}$$

Manipulate this to get the Output(C) on one side and Input(R) on the other:

$$\mathbf{C + K.G.H.C = K.G.R \quad \Rightarrow C(1 + K.G.H) = K.G.R \quad \Rightarrow C = \frac{K.G}{1 + K.G.H} R}$$

Bring the input over to the left to give the Closed Loop Transfer Function (CLTF):

$$\mathbf{Closed\,Loop\,Transfer\,Function} = \frac{\mathbf{Output}}{\mathbf{Input}} = \frac{\mathbf{C}}{\mathbf{R}} = \frac{\mathbf{K.G}}{\mathbf{1 + K.G.H}} \qquad \text{Eqn 5.3}$$

K.G is called the **Forward Loop Transfer Function (FLTF)** and **K.G.H** is called the **Open Loop Transfer Function (OLTF**). The Closed Loop Transfer Function can then be written as:

$\mathbf{CLTF} = \dfrac{\mathbf{FLTF}}{\mathbf{1 + OLTF}}$, or, in words:

$$\begin{matrix}\mathbf{Closed\,Loop}\\ \mathbf{Transfer\,Function}\end{matrix} = \frac{\begin{matrix}\mathbf{Forward\,Loop}\\ \mathbf{Transfer\,Function}\end{matrix}}{\mathbf{1} + \begin{matrix}\mathbf{Open\,Loop}\\ \mathbf{Transfer\,Function}\end{matrix}}$$

Knowing values for the plant, feedback and controller transfer functions, this equation allows us to predict how the closed loop control system will behave.

5.4 Error Transfer Function

It is sometimes useful to know the amount of error there is between the input and the output, relative to the input. The equation for the amount of error (relative to the input) can also be derived from equations 5.1 and 5.2.

$$\mathbf{E = R - H.C = R - H.K.G.E}$$

$$\mathbf{E + K.G.H.E = R \quad \Rightarrow E(1 + K.G.H) = R \quad \Rightarrow \frac{E}{R} = \frac{1}{1 + K.G.H}} \qquad \text{Eqn 5.4}$$

5.5 Closed Loop Performance - Steady State

The closed loop performance is described by the closed loop transfer function. To design a controller, we need to know what influences the closed loop transfer function. Restate the closed loop transfer function equation:

$$\mathbf{CLTF} = \frac{\mathbf{C}}{\mathbf{R}} = \frac{\mathbf{K.G}}{\mathbf{1 + K.G.H}}$$

For Steady State performance we do not need to consider the dynamic effects so each of the transfer functions can be represented by its gain. Transient performance will be considered in Section 5.7. Dividing top and bottom by KG gives:

$$\frac{\mathbf{C}}{\mathbf{R}} = \frac{\mathbf{1}}{\frac{\mathbf{1}}{\mathbf{K.G}} + \mathbf{H}} \qquad \text{Eqn 5.5}$$

In most systems there is direct feedback so H = 1. This is called **Unity Feedback**.

K is under our control so we could make K.G large. 1/(K.G) would then be much less than 1 in which case this term can be ignored and the output equals the input.

$$\frac{\mathbf{C}}{\mathbf{R}} = \frac{\mathbf{1}}{\mathbf{0 + 1}} = \mathbf{1} \Rightarrow \therefore \mathbf{C = R} \qquad \text{or} \qquad \textbf{Output = Input}$$

If only it were so easy. There are two difficulties to just increasing the gain. One is the transient effects which will be considered later and the other is noise in the measurement. As you may have noticed, the velocity measurement available from the tachometer tends to be noisy. If the gain is high, this noise is amplified and affects the performance of the motor.

Lets observe this in practice.

Steady State Velocity Error with Proportional Gain

For this exercise, the system is set to control the motor speed. The block diagram of the system is shown in Fig 5.5.

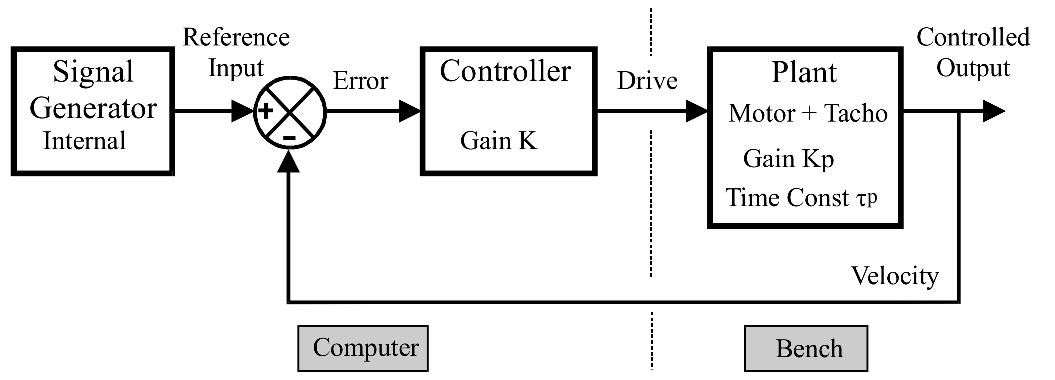


Fig 5.5 Diagram of the Speed Control System

The circuit is arranged so that the voltage from the tachometer (measuring motor velocity) is compared with the demanded velocity from the signal generator. The difference between these two signals, the error, is amplified and used to drive the motor.

Note: so far we have used speed and velocity as being interchangeable. To be exact, speed is a scalar quantity with magnitude only (such as 100 rpm), whereas velocity is a vector quantity with magnitude and direction (such as 100 rpm clockwise).

Start VCL software and **Load setup | CA06PE05**. The on-screen mimic should be compared with Fig 5.5. The Input signal is trace 1 (dark blue), the output velocity is trace 4 (purple), Error is trace 3 (dark green) and Drive is trace 5 (brown). The mimic shows where these measurements are made.

File CA06PE05	**Controller** Proportional	**Plant** MS15 Analog		**Display** Graph	
Signal Generator		**Graph**			
Signal	DC-Level	1 Input	ON	5 Drive	ON
Level	0%				
Offset	50%	3 Error	ON		
Rate	10 msec	4 Velocity	ON		
Reference	Internal				
DC Motor		**Output Potentiometer**		Disengage	
Brake	0	**Command Potentiometer**		180°	

In Chapter 3, the gain K_p of the motor was found. Substituting K_p for G gives the forward loop gain K.G = $K.K_p$. The velocity voltage is fed back directly so H = 1. In this case, the Closed Loop Transfer Function can be written as:

$$\mathbf{CLTF} = \frac{\mathbf{C}}{\mathbf{R}} = \frac{\mathbf{K.K_p}}{\mathbf{1 + K.K_p}} \qquad \text{Eqn 5.6}$$

and the Error Response, from Eqn 5.4, is:

$$\frac{\mathbf{E}}{\mathbf{R}} = \frac{\mathbf{1}}{\mathbf{1 + K.K_p}} \qquad \text{Eqn 5.7}$$

Check that the gain K in the gain box on the on-screen mimic is set to 1 and switch ON the motor. Investigate how the error decreases as the gain is increased. With the Reference Velocity set to 2.5 volts (50%), measure the Output Velocity and Error as the gain is increased. Make these measurements for the values of gain shown in Table 5.1 and enter the results in your workbook. If the motor starts to make a lot of noise at high gain, abandon the high gain settings as this could damage the motor.

Input Voltage, R = 2.5 volts, Plant gain K_p = ☐

Gain K	Velocity (Volts)	C/R	C/R Theory	Error (Volts)	E/R	E/R Theory	Comment
1							
2							
5							
10							
20							
50							
100							

Table 5.1 Accuracy as gain is increased.

Disable the motor.

In line with predictions, as the gain increases, C/R → 1 and E/R → 0. However, with high gain the output became noisy and the motor started making peculiar noises.

Steady State Error decreases as loop gain is increased.

Observation of the Drive signal should show that this signal gets noisier as the gain increases - all of the tacho measurement noise is being amplified. This noise is not good for the motor.

5.5a **Unity feedback means that the feedback transfer function H equals:**

[a] 0 [b] 1 [c] 2 [d] 10

5.6 Closed Loop Performance - Sensitivity to Load Disturbances

In Chapter 3, you saw that changing the load by adding eddy current braking greatly changed both the gain and time constant of the motor. Let us see what happens now to the speed when the additional load is added to the closed loop system.

Set the gain K = 10 and enable the motor. Add the braking load and note that there is relatively little change to the output speed whereas, in Chapter 3, there was a large change in the output. The improvement is the result of the feedback.

Disable the motor.

Closing the loop decreases the sensitivity to changes within the plant. The system is said to be stiffer.

Speed control was the problem which was solved by the first control device. James Watt was using his new steam engines to power factories. As the belt driven machines were engaged or disengaged from the engine, the speed changed resulting in poor quality product. In 1788 he developed the 'Flying Ball' Governor to regulate the speed of the engines. As the speed dropped, the steam valve was opened. As the speed increased the valve was closed. The Flying Ball Governor is a mechanical proportional controller and models are still in use today.

5.6a **When the eddy brake is applied, did the velocity:**

[a] fall by a relatively large amount. [b] fall by a relatively small amount.

[c] not change. [d] increase.

5.7 Closed Loop Performance - Transient

If τ_p is the plant time constant, as measured in Chapter 3, and τ_{cl} is the time constant of the closed loop system, it can be shown, for the motor speed control system we have, that:

$$\mathbf{t_{cl} = \frac{t_p}{1 + K.K_p}} \quad \text{Eqn 5.8}$$

As the controller gain is increased, the time constant falls leading to a speeding up of the system. Why should this be so since the inertia and friction of the motor have not changed? Consider Fig 5.6:

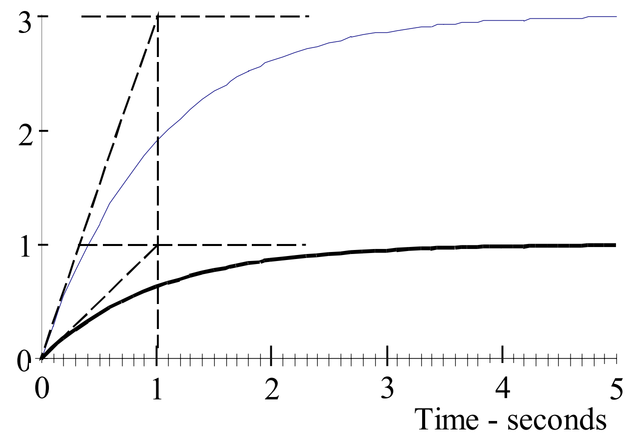


Fig 5.6 Exponential Curves With The Same Time Constant But Different Amplitudes

The two exponential curves shown have the same time constant but are of different amplitude. There is a 3:1 ratio in amplitude. Since the larger curve has to cover 3 times the distance in the same time, initially it must be moving 3 times faster than the smaller curve.

In a feedback system, it is the error amplified which provides the drive. Initially the error is the size of the step change and this is amplified so there is a large signal to drive the motor pumping more energy into the system. That is the theory. In practice it does not quite work out that way.

Change the setup as indicated by the Table below.

File CA06PE05	**Controller** Proportional	**Plant** MS15 Analog		**Display** Graph	
Signal Generator		**Graph**			
Signal	Step	1 Input	ON	5 Drive	ON
Level	20%				
Offset	0%	3 Error	ON		
Rate	10 msec	4 Velocity	ON		
Reference	Internal				
DC Motor		**Output Potentiometer**		Disengage	
Brake	0	**Command Potentiometer**		180°	

Using your measured value for gain K_p and time constant τ_p from Chapter 3, calculate the expected time constant τ_{cl} at each of the gains in Table 5.2 using Equation 5.5. Measure the actual time constant using the same technique that was used in Chapter 3 to measure time constants and enter the theoretical and experimental values into your workbook.

To help, the *Time* control should be set to x8 and the *Magnify* control to 5 on channel 4. Owing to the static friction, it is not easy to measure the time constant with K = 1 so, for this measurement only, set the *Level* to 40%.

K_p = [ ] τ_p = [ ] seconds

Gain K	Theoretical Time Constant	Measured Time Constant
1		
2		
3		
4		
5		
10		

Table 5.2 Decreasing Time Constant with Increasing Gain

For K = 1 → 3, the value for time constant follows the theory, i.e. increasing the gain decreases the time constant. However, above K = 3, the theory falls down as the time constant does not decrease in proportion to the increase in gain. The time constant with K = 10 should be approximately 1/5th of that with K = 1. Why is it not?

Set K = 10 and look at the Drive signal - the brown trace. When the step occurs, the initial error is 2V. The drive, theoretically, should be K x Error = 20V. But the maximum drive signal available is only 5V so the system saturates. Much less power than expected is being fed to the motor so the transient time does not decrease as expected. Once the drive saturates, the rise time of the transient becomes almost constant.

This is one of the practical nonlinear effects which was mentioned earlier.

> **The speed of response to a transient should increase with increasing gain but the initial slope is limited by the maximum drive signal.**

5.7a **As the controller gain is increased, would you expect the steady state error to:**

[a] increase. [b] stay the same.

[c] decrease.

5.7b **Did the measured steady state error behave as expected?**

[Yes] or [No]

5.7c **The increase in gain was limited by:**

[a] linearities. [b] drive saturation.

[c] noise in the system.

5.7d **As the controller gain is increased, would you expect the closed loop time constant to:**

[a] increase. [b] stay the same.

[c] decrease.

5.7e **Did the measured closed loop time constant behave as expected (decreasing steadily until saturation is reached) over the range of gains measured?**

[Yes] or [No]

5.7f **Using your measured values for plant gain K_p and time constant τ_p, calculate the gain K necessary for a closed loop time constant of 75 ms.**

5.8 Closed Loop Performance - Frequency Response

Since the time constant is reduced by increasing the gain, it would be expected that the break frequency would increase since break frequency is the inverse of time constant. Measuring the frequency response is not, as you have seen, an easy task but the effect of gain on frequency response can be illustrated by observing the phase change.

Change the setup as indicated by the Table below.

File CA06PE05	**Controller** Proportional	**Plant** MS15 Analog	**Display** Graph
Signal Generator		**Graph**	
Signal	Sine	1 Input ON	5 Drive ON
Level	20%		
Offset	0%	3 Error ON	
Freq	1 Hz	4 Velocity ON	
Reference	Internal		
DC Motor		**Output Potentiometer**	Disengage
Brake	0	**Command Potentiometer**	180°

Measure the phase lag of the output at the three gain settings shown in Table 5.3 and enter the values your workbook. Compare this figure with the open loop phase lag at this frequency as measured in Chapter 4.

Open loop phase lag at 1 Hz	

Closed loop phase lag at 1 Hz	
K = 1	
K = 5	
K = 10	

Table 5.3 Closed Loop Phase Lag

The phase lag has reduced indicating that the break frequency does increase with increasing gain. This time the increase continues as the gain increases. If you observe the drive signal you will see that, even at a gain of 10, it is not saturating. Because the input signal does not have steps in it, the error is never too great and the system stays within the linear region.

5.8a **Were the measured phase lags as expected?**

Yes or No

5.9 Summary of Proportional Feedback Control

Simple proportional control does improve the performance of a system. As gain is increased, the steady state error is reduced, the system gets stiffer and speeds up. However, gain cannot be increased indefinitely as system noise is amplified by the high gain. Also, power restrictions limit the amount by which the system speeds up.

Since there is only one control variable K, the control engineer can design for a specified steady state error or for a specified time constant. You cannot meet two specifications with one control variable.

Student Assessment 5

1. A closed loop system is created when input is compared with the:

- [a] output.
- [b] drive.
- [c] error.
- [d] rate of change of output.

2. A plant consists of two blocks in series. They have gains of 10 and 2. Is the overall gain of the plant:

- [a] 2
- [b] 10
- [c] 12
- [d] 20

3. A plant consisting of a single lag with a gain of 12 has closed loop control with unity feedback and a controller gain of 1.5. If the input is 2 volts, is the output voltage:

- [a] 1.8
- [b] 1.9
- [c] 2.0
- [d] 25

4. Which of the following causes the closed loop time constant not to decrease as the gain increases:

- [a] experimental error.
- [b] drive saturation.
- [c] amplifier deadband.
- [d] slow computer.

5. As the gain is increased, the break frequency:

- [a] increases.
- [b] decreases.
- [c] remains the same.
- [d] is unpredictable.

Chapter 6
Proportional Position Control

Objectives of this Chapter

Having completed this chapter you will be able to:

- Account for the excellent steady state performance of a proportional servo position system
- Account for the poor transient performance of a proportional servo position system

Equipment Required for this Chapter

- MS15 DC Motor Module
- AS3 Command Potentiometer
- CLIO Interface Module with PC Connection Lead
- System Power 90 Power Supply (or equivalent)
- 4 mm Connection Leads
- PC running VCL Virtual Control Laboratory Software

6.1 Introduction to Servomechanisms

Position control using a DC motor is one of the major applications of control. Many of the control solutions were developed for this type of problem - mainly to do with the aiming of artillery weapons by the military. There are many more peaceful applications for this type of control. The DC motor is also called a Servomotor and this branch of control is called **Servomechanisms**.

Fig 6.1 shows a sketch of the test system we have. Fig 6.2 is the block diagram of the system.

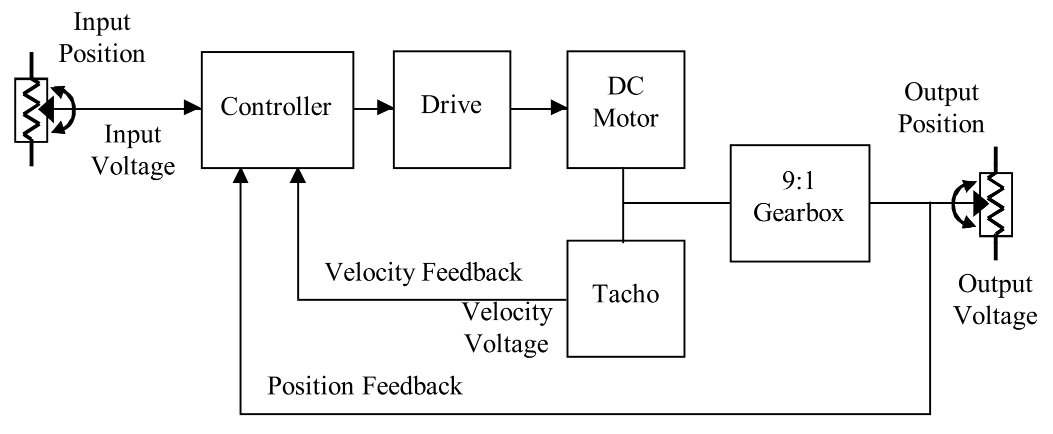


Fig 6.1 Sketch of a Simple Servo Controller

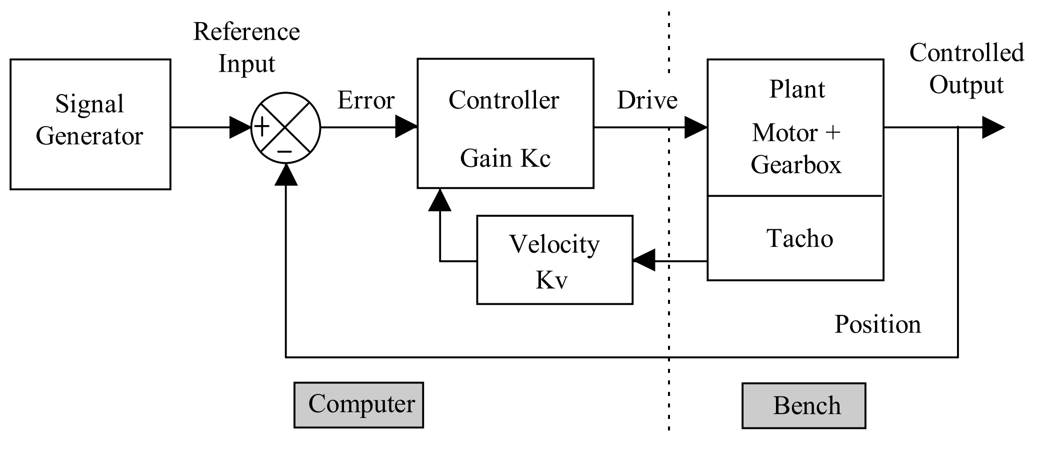


Fig 6.2 Block Diagram of a Servo System

In its simplest type of application, the artillery operator turns one dial to rotate the weapon barrel to point in the required horizontal direction, and another dial to elevate the barrel to point at the required vertical angle to the horizon. When the barrel is positioned correctly, the artillery weapon is fired. It important to get the weapon on target as quickly as possible.

Proportional Position Control

We will investigate the behavior of a position control system with proportional control.

Start VCL software and **Load setup | CA06PE06**.

File CA06PE06	**Controller** Servo	**Plant** MS15 Analog		**Display** Graph	
Signal Generator		**Graph**			
Signal	Step	1 Input	ON	5 Drive	OFF
Level	20%	2 Position	ON		
Offset	0%	3 Error	OFF		
Rate	10 msec	4 Velocity	OFF		
Reference	External				
DC Motor		**Output Potentiometer**		Engage	
Brake	0	**Command Potentiometer**		180°	

Check that controller gain K_c is set to 0.8 and VFB (Velocity Feedback) is Off. The additional feedback control box labeled VFB is not active during this investigation. Make sure that the output potentiometer is engaged then switch power ON.

To illustrate that this is a position servo, rotate the Command Potentiometer between 30° and 300° and watch the output dial follow, albeit rather sluggishly. Increase K_c to 5 and turn the input dial again. This time the output dial moves much faster but wobbles (oscillates) around before it settles to a steady value. Somewhere between these two gains there may be an optimum setting.

You can see the behavior of the system on the screen. Trace 1 (dark blue) is the input position and trace 2 (blue) is the output position. The other 3 traces, Error, Velocity and Drive, are available for other parts of the exercise but are currently Off.

Change the *Reference* Input to **Internal**. The signal generator should be set to **Step 20%**. You will see the step response of the closed loop position system on the blue trace. With $K_c = 5$, this oscillatory response is different to anything observed with the proportional speed control system examined in the last chapter.

6.1a **When the gain is low, which of the following best describes the movement of the output potentiometer as it tries to follow the input disk?**

[a] Sluggish.
[b] Swift with no oscillations.
[c] Swift with a lot of oscillation.

6.1b **When the gain is high, which of the following best describes the movement of the output potentiometer as it tries to follow the input disk?**

[a] Sluggish.
[b] Swift with no oscillations.
[c] Swift with a lot of oscillation.

6.2 Steady State Behavior

Set $K_c = 2.5$ and compare the input and output traces once the transient has died away. You will see that they are indistinguishable. Even with this low gain, the steady state conditions have been met.

In Chapter 5 equation 5.5, it was established that the steady state transfer function for a unity feedback system would be:

$$\frac{\mathbf{C}}{\mathbf{R}} = \frac{\mathbf{K.G}}{\mathbf{1 + K.G}} = \frac{\mathbf{1}}{\frac{\mathbf{1}}{\mathbf{K.G}} + \mathbf{1}}$$

A high gain is required to give a closed loop transfer function of 1. But, from the observations just made, it would appear that C/R = 1 with a low value of gain K_c.

Why is this so? The answer lies in the integration effect between velocity and position. At any steady velocity, after an infinite time you will have traveled an infinite distance so we can say that the steady state gain of an integrator is infinite.

The effect of the integral can be looked at in another way. Any position error will drive the motor. The feedback ensures that the motor will be driven to reduce that error. The motor will stop turning when there is no error.

This can be explained diagramatically. Examine Fig 6.3. When the error, and hence velocity, is a steady value the position will be a ramp - ignoring the transient lag effects of the motor. Position is the integral of velocity, or in other words the sum of all the velocities over a time.

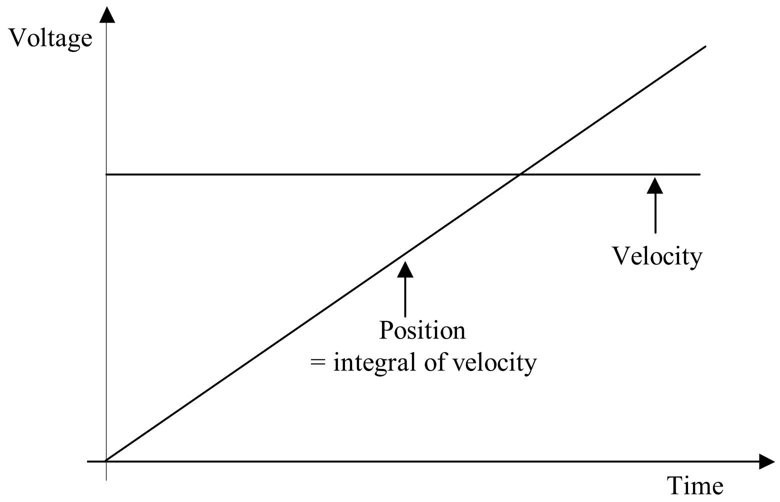


Fig 6.3 Signals Within the System

You can see what happens to the motor if you look at little time slices of the signal. This is shown in Fig 6.4. When the step is applied, there is a large error so the motor runs at high speed and the position ramps up quickly. At the end of the first period, the output has moved towards the input so the error is reduced and the motor now runs more slowly. The position also changes more slowly. At the end of each period, the motor is running more slowly and the position is changing more slowly but it will eventually get to where we cannot distinguish the output position from the input position.

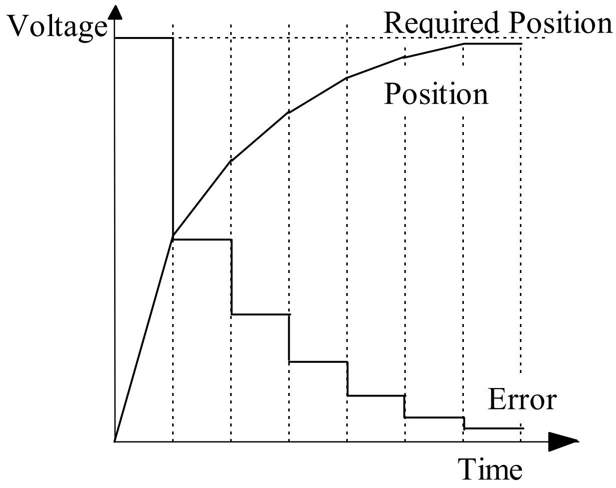


Fig 6.4 Position Response to a Step

Set K_c = 0.8 and you will see the position (channel 2) and error (channel 3) behave in the way described although the traces are much smoother.

For speed control, the motor could be described by its gain and time constant. Because position is the integral of velocity, in position control systems the motor is described by its gain, its time constant and an integration effect.

A position type control system (one with an integration effect in the forward path) will have no steady state error to a steady position input.

At low gain, the steady state error may not have been zero. This is due to practical considerations. If you look back at Fig 2.1 in your workbook, you will see that a small drive voltage is required before the motor will move. This, and static friction in the bearings, can cause a small steady state error when the gain is low.

6.2a **Except when the gain is very low, did the output always reach the steady state value set?**

Yes or No

6.3 Transient Behavior

Compared with the speed control system, having the additional integration effect between velocity and position eliminates the steady state error. However, the integration does create problems with the transient behavior.

Increase K_c to 3 and you will see the problem. Although the steady state value is reached, eventually, there is a lot of trouble getting there. If you were on a lift and it oscillated like this, you would not be very pleased.

We could just leave the gain at 0.8 but this would not give the speediest response.

In Fig 6.3, the effects of the motor lag were ignored. This is adequate at low gain but not when the gain is increased. Including the lag effect gives signals as in Fig 6.5.

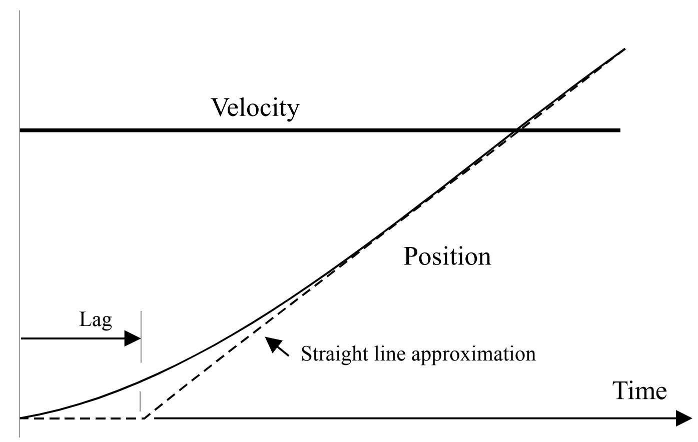


Fig 6.5 Signals within the System including the effect of the motor lag

When the step is applied, there is a delay before the motor reaches the speed demanded. This causes the position to lag behind the ideal signal. If this lag is added to the linear approximation, the response shown in Fig 6.6 results.

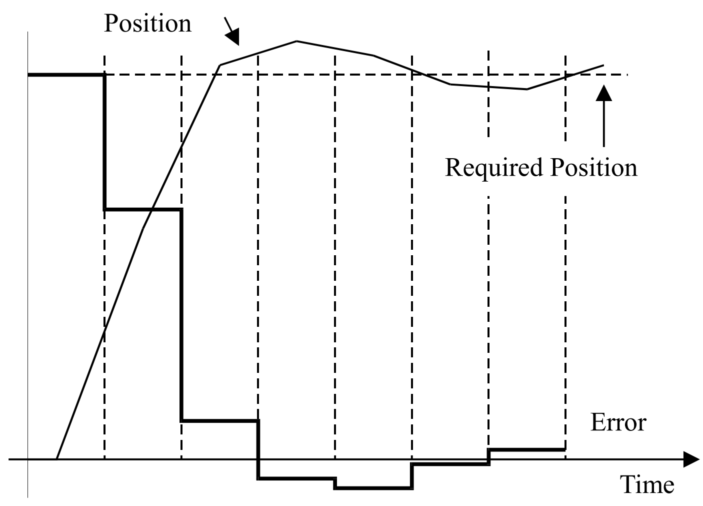


Fig 6.6 Position Response to a Step including Lag Effect

Within each time slice, the change in position is lagging behind the signal driving the velocity.

When the error signal reaches zero telling the motor to stop, the motor runs on beyond the required position owing to its inertial lag. The motor then has to be reversed to allow the position potentiometer to reach the required position. The lag tends to make the system less stable.

The servo system with only proportional control can give an adequate response but, with only one control, a system can be designed for a particular speed of response or for a particular degree of oscillation.

6.3a **Which of the following contributes least to the transient oscillations:**

[a] The high gain driving the motor at high speed.

[b] The inertial lag of the motor.

[c] The size of the input step.

6.4 Sensitivity to Load Disturbances

Set K_c to 2.0. The steady state position is at its demanded level. Apply the brake, setting it to position 2. The steady state position is still at its demanded level. Because of the integration effect, the output will always reach its demanded steady state level irrespective of load - provided the motor is adequate enough.

But there is a transient problem introduced by changing loads.

With the brake at position 2, adjust K_c to give a response that just does not oscillate when the step is applied. Now take the brake off and observe what happens to the transient. The output overshoots before coming to the steady value. The load has an effect on the transient response. So setting the gain to give a good transient response under one load condition can result in a poor response with a different load - a crowded lift would be sluggish reaching a floor, a lightly loaded lift would bounce around.

6.4a **Is it the case that increasing the load by applying the brake has no affect on the steady state performance:**

Yes or No

6.4b **Increasing the load by applying the brake makes the transient response:**

- a more sluggish.
- b more oscillatory.
- c remain the same.

6.5 Summary

Position control adds an integration effect into the system. This gives excellent steady state performance but the gain setting can give a variety of transient responses - ranging from the sluggish to the oscillatory - depending on the load.

An understanding of the oscillatory performance is required before ways of increasing the speed of response without causing oscillations can be introduced.

Student Assessment 6

1. Which of the following is not true. In a position control system, the steady state position error is zero because:

[a] the low frequency gain of an integrator is high.

[b] the error is integrated until there is no error.

[c] input position is compared with output position to produce an error signal.

[d] the output voltage is added to the input voltage to create the motor drive.

2. As the proportional gain is increased, the transient response:

[a] slows down.

[b] does not change.

[c] speeds up but is not oscillatory.

[d] speeds up but is oscillatory.

3. Compared to an open loop system, closing the loop under changing load conditions:

[a] has no affect.

[b] eliminates change in steady state error.

[c] eliminates steady state error.

[d] increases the steady state error.

Chapter 7
Behavior of Second Order Systems

Objectives of this Chapter

Having completed this chapter you will be able to:

- Distinguish between underdamped, critically damped and overdamped systems
- Relate the overshoot and damped frequency to damping factor and natural frequency
- Describe the form of the step response and the frequency response of a second order system

Equipment Required for this Chapter

- MS15 DC Motor Module
- AS3 Command Potentiometer
- CLIO Interface Module with PC Connection Lead
- System Power 90 Power Supply (or equivalent)
- 4 mm Connection Leads
- PC running VCL Virtual Control Laboratory Software

7.1 Second Order Systems

In the speed control system, the plant was characterized by its time constant, which is determined by the inertia of the rotor and the viscous friction. This arrangement can be described mathematically by a first order differential equation.

The position control system has an integration effect between velocity and position. This makes the position control servomechanism into a second order system.

As a first order system is characterized by its time constant, it could be expected that a second order system would be characterized by two time constants. Although some second order systems can be described in this way, most of the systems dealt with in closed loop control cannot be described so simply.

In Chapter 6, it was observed that, as the gain increased, the position control system became more and more oscillatory. Fig 7.1 shows how the step response changes as the proportional gain is increased in a typical second order system.

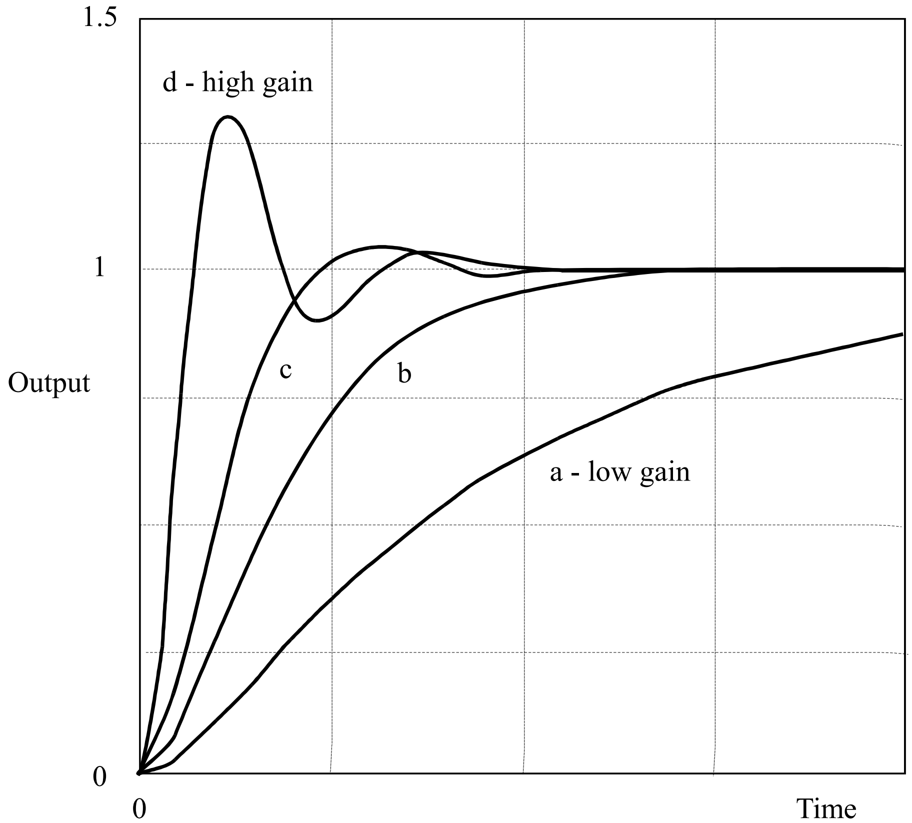


Fig 7.1 Step Response of a Typical Second Order System

When the gain is low, the response is sluggish and is said to be **Overdamped**. An overdamped response is characterized by two separate time constants.

Curve b shows the fastest response this system can have without any oscillation. This response is said to be **Critically Damped**. A critically damped response is characterized by two time constants both of the same value.

With higher gain, the response overshoots and oscillates. This type of response is said to be **Underdamped**. An underdamped response cannot be characterized by time constants. Mathematically it is described by a decaying sinusoid.

Examining Fig 7.1, the "best" response would appear to be somewhere between curves b and c. Before we can predict the gain necessary to give a specified response, we need to know how to describe the behavior of a second order system.

Underdamped systems are often described by the amount the response overshoots and by the frequency at which it oscillates.

There are two other parameters used to describe second order systems - **Damping Factor** and **Natural Frequency of Oscillation**.

To be able to predict the overshoot and frequency of oscillation of a closed loop system, we must develop how the different parameters are related to the gain and time constant of the plant being controlled.

7.1a **A position control system can be described as a:**

[a] 1st order system. [b] 2nd order system.

[c] 3rd order system. [d] 4th order system.

7.1b **Position control systems are designed to be:**

[a] overdamped.

[b] critically damped.

[c] slightly underdamped.

[d] very underdamped.

7.1c **The characteristic of a slightly underdamped system is that it:**

[a] approaches the steady state slowly.

[b] does not overshoot.

[c] may have a small overshoot.

[d] overshoots and has decaying oscillations.

7.1d **Overshoot increases as the gain increases:**

[Yes] or [No]

7.2 Overshoot and Damping Factor

Overshoot

Overshoot is the amount by which a response goes beyond the steady state value before settling down. Fig 7.2 shows the response of a typical underdamped system.

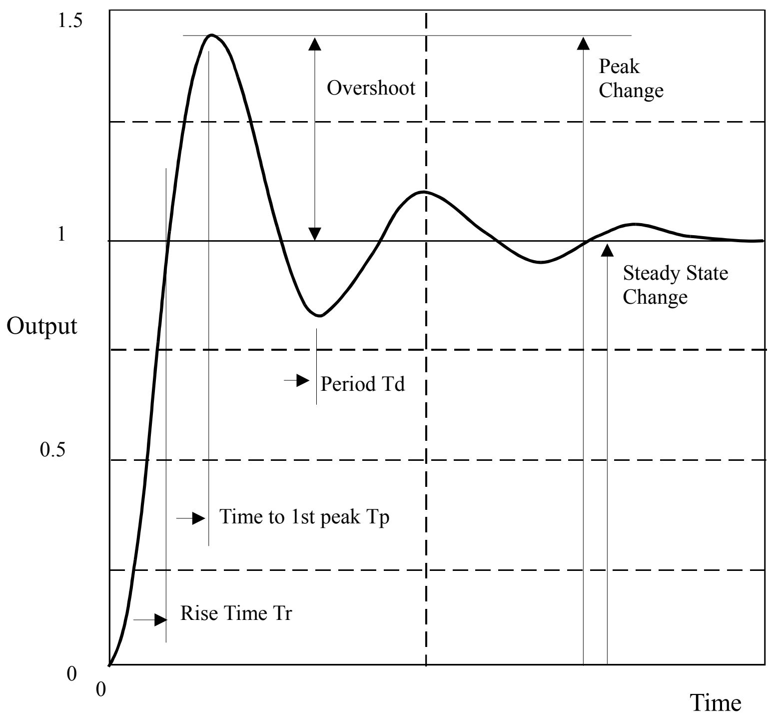


Fig 7.2 Step Response of an Underdamped System

Overshoot can be measured from the step response. It is the ratio:

$$\frac{\text{Peak Output Change - Steady State Output Change}}{\text{Steady State Output Change}}$$

Overshoot is usually stated as a percentage, which is the above ratio multiplied by 100.

Damping Factor

In the equations describing system behavior, overshoot is not an easy parameter to handle. Another parameter called **Damping Factor** is used and this gives an indication of the amount of overshoot in a system. Damping Factor has the symbol ζ (Zeta). ζ has a value of 1 when the system is critically damped, less than 1 when underdamped and greater than 1 when overdamped.

$\zeta < 1$ Underdamped - decaying oscillations
$\zeta = 1$ Critically Damped - just no overshoot
$\zeta > 1$ Overdamped - system sluggish.

The objective of a control system design is often to achieve a fast response without any overshoot or with just a little overshoot. Systems are usually designed for ζ in the range 0.7 to 1.

The amount of overshoot is wholly dependent on the Damping Factor. Measuring overshoot allows the damping factor to be calculated and knowing ζ allows the overshoot to be calculated. They are linked by the equations:

$$\zeta = \sqrt{\frac{1}{1+\left[\frac{\pi}{\ln(\text{overshoot ratio})}\right]^2}} \qquad \text{Eqn 7.1a}$$

$$\text{overshoot ratio} = e^{-\pi\zeta/\sqrt{1-\zeta^2}} \quad \Rightarrow \quad \%\ \text{overshoot} = 100 \times e^{-\pi\zeta/\sqrt{1-\zeta^2}} \qquad \text{Eqn 7.1b}$$

The natural logarithm ln (= $\log_e$) is the **ln** key on the Windows calculator.

To save lots of calculation, this relationship has been graphed in Fig 7.3. Fig 7.3b depicts the region of most interest to control engineers.

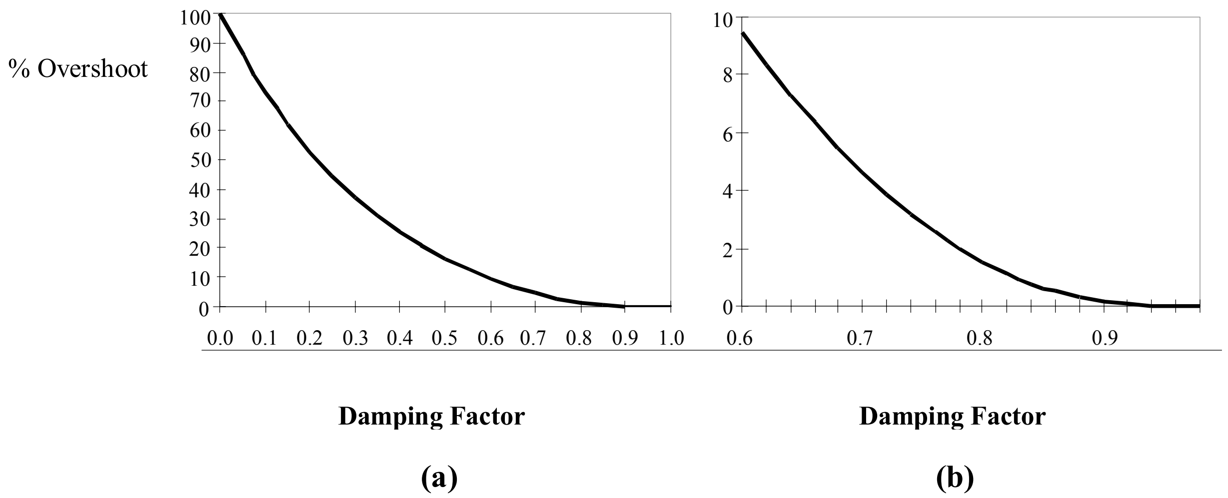


Fig 7.3 Damping Ratio vs Overshoot

7.2a **Damping Factor can be obtained by measuring overshoot:**

Yes or No

7.2b **The best response is obtained when:**

a $\zeta > 1$ b $\zeta = 1$

c $0.7 < \zeta < 1$ d $\zeta < 0.7$

7.2c **A response has an overshoot of 4%. What is its damping factor?**

7.2d **A response has a damping factor of 0.8. What is its percentage overshoot?**

7.3 Damped and Natural Frequencies of Oscillation

Damped Frequency

The frequency at which an underdamped system oscillates is called the **Damped Frequency**, $\boldsymbol{\omega_d}$. This can be determined by measuring the time between successive positive peaks if, as shown in Fig 7.2, there is more than one cycle.

The inverse of the period of a cycle is its frequency in Hertz:

$f_d = \frac{1}{T_d} \quad \Rightarrow \quad \omega_d = 2\pi f_d = \frac{2\pi}{T_d}$ where T_d is the period of the oscillation.

The time to the first peak, T_p, is half the period. The damped frequency can then be found by measuring the time to the first peak:

$$\omega_d = \frac{\pi}{T_p}$$

Natural Frequency

If there was no damping at all ($\zeta = 0$), the system would continuously oscillate at a frequency which is called the **Natural Frequency** of the system. This is given the symbol ω_n. The relationship between Natural and Damped frequencies is:

$$\omega_n = \frac{\omega_d}{\sqrt{1-\zeta^2}} \qquad \text{Eqn 7.2a}$$

$$\omega_d = \omega_n\sqrt{1-\zeta^2} \qquad \text{Eqn 7.2b}$$

It can be seen from Fig 7.1 that ω_d increases as the gain increases, which means that increasing the gain makes the system work faster but at the expense of increasing the overshoot.

7.3a **The natural frequency is the frequency at which the system will oscillate when:**

a $\zeta > 1$	b $\zeta = 1$
c $\zeta < 1$	d $\zeta = 0$

7.4 Relating ζ and ω_n to Open Loop Parameters

Second order systems are characterized by their Damping Factor ζ and Natural Frequency ω_n from which the overshoot and damped frequency can be obtained. To be able to design a system, we need to know how, in a closed loop system, ζ and ω_n relate to the plant parameters K_p, K_i and τ which were measured in Chapters 3 and 4.

The model shown in Fig 7.4 was developed.

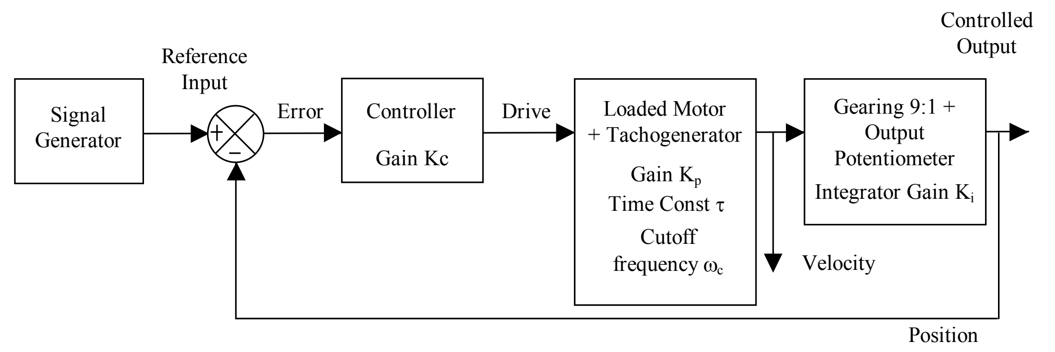


Fig 6.2 Block Diagram of a Servo System

The two parameters describing the system are Gain (K_p x K_i) and Time Constant τ. The time constant can also be stated as its inverse, the cut-off frequency ω_c. The proportional gain in the controller must also be considered so the forward loop gain of the servo system is $K = K_c$ x K_p x K_i.

Damping Factor

Analysis of the system equations show that:

Damping Factor $\zeta = \sqrt{\dfrac{\omega_c}{4K}}$ Eqn 7.3

Critical Damping is when $\zeta = 1$. From equation 7.3, this occurs at $K = \frac{\omega_c}{4}$.

When $K > \frac{\omega_c}{4}$, $\zeta < 1$ and the system is underdamped.

With $K < \frac{\omega_c}{4}$, $\zeta < 1$ and the system is overdamped.

For the best response, such that $0.7 < \zeta < 1$, set K to between $\frac{\omega_c}{2}$ and $\frac{\omega_c}{4}$.

> **Example -** A plant has a gain K_p of 1 and an integrator gain K_i of 10. If the break frequency is 5 rad/sec, what gain is required to give an overshoot of 5%?
>
> Look up Fig 7.3b. 5% overshoot required a damping factor of 0.7. From equation 7.3:
>
> $$0.7 = \sqrt{\frac{5}{4 \times K_c \times 1 \times 10}}$$
>
> $$0.49 = \frac{5}{40 \times K_c}$$
>
> $$K_c = \frac{1}{8 \times 0.49} = 0.255$$

Knowing the model of the plant, the controller gain can be used to set the overshoot of the closed loop system.

Natural Frequency

It can also be shown that the Natural Frequency $\omega_n = \sqrt{K\omega_c}$ Eqn 7.4

From equation 7.2b, this gives the Damped Frequency as:

$$\omega_d = \sqrt{K\omega_c(1-\zeta^2)} \qquad \text{Eqn 7.5}$$

or, substituting equation 7.3 for ζ into 7.5:

$$\omega_d = \sqrt{K\omega_c - \frac{\omega_c^2}{4}}$$

Remember, with $\zeta \geq 1$, there is no ω_d as the system is represented by two lag terms.

> **Example -** Using the same system as in the previous example, what is the time to first peak?
>
> $$\omega_d = \sqrt{0.255 \times 1 \times 10 \times 5 - \frac{25}{4}} = \sqrt{12.75 - 6.25} = \sqrt{6.5} = 2.53 \text{ rad/sec}$$
>
> $$T_p = \frac{\pi}{w_d} = \frac{\pi}{2.53} = 1.24\text{secs}$$

Since ω_c is fixed by the motor and the other system mechanics, the gain K_c is the only control variable in proportional control. Varying K_c affects both ω_d and ζ.

Using only proportional gain control, we can design for a particular overshoot or a particular time to first peak but not both. Additional techniques will be introduced in the next chapter which will allow both ω_d and ζ to be controlled.

7.4a **The gain for best operation is:**

a $K < \frac{\omega_c}{4}$ b $K = \frac{\omega_c}{4}$ c $\frac{\omega_c}{2} > K > \frac{\omega_c}{4}$ d $K < \frac{\omega_c}{2}$

7.4b **A plant has a gain K_p of 2 and an integrator gain K_i of 2. If the break frequency is 10 rad/sec, what gain is required to give a damped frequency of 13 rad/sec?**

7.5 Experimental Check of these Relationships

We have seen that increasing gain should increase the damped frequency and reduce the damping factor. But natural frequency and damping factor are not directly measurable so we shall measure overshoot and time to first peak. From these measurements we will calculate ω_n and ζ using the previous measurements of K_p, K_i and ω_c.

Start VCL software and **Load setup | CA06PE07**.

File CA06PE07	**Controller** Servo	**Plant** MS15 Analog		**Display** Graph	
Signal Generator		**Graph**			
Signal	Step	1 Input	ON	5 Drive	ON
Level	30%	2 Position	ON		
Offset	0%	3 Error	OFF		
Rate	10 msec	4 Velocity	OFF		
Reference	Internal				
DC Motor		**Output Potentiometer**		Engage	
Brake	0	**Command Potentiometer**		180°	

Set the gain K_c to 1.5. and K_v to 0 (VFB is not used in this exercise).

Switch On and, when a complete graph has been drawn, freeze the traces and enable the Time markers.

- Using the markers, measure the steady state output change and the peak output change. This allows the overshoot to be calculated.
- Obtain the damping factor ζ from Fig 7.3.
- Measure the time to first peak.
- Calculate the damped frequency ω_d and then the natural frequency ω_n.

Enter these values in Table 7.1 in your workbook

Gain K_c	Overshoot	ζ	T_p seconds	ω_d rad/sec	ω_n rad/sec
1.5					

Table 7.1 Measurement of Damping Factor and Natural Frequency

Using the model obtained in Chapters 3 and 4, calculate the expected values for damping factor and natural frequency with K_c = 1.5. Enter the values into Table 7.2.

K_c	K_p	K_i	ω_c rad/sec	ζ	ω_n rad/sec
1.5					

Table 7.2 Calculated values for Damping Factor and Natural Frequency

These results are in good agreement considering the nonlinearities within the drive motor and experimental error in this experiment and those used to determine the plant parameters.

7.5a **Compared with the calculated values, were the experimental values for damping factor and natural frequency:**

- a almost exactly correct.
- b reasonably close.
- c nowhere near.

7.5b **Which of the following would not contribute to errors between calculated and measured values for damping factor and natural frequency:**

- a nonlinearities in the drive amplifier.
- b errors in measuring open loop gain and time constant.
- c errors in measuring overshoot and damped frequency.
- d computer error.

7.6 Second Order Step Response

Rise Time

Fig 7.2 shows the step response of a second order system with the time to first peak and the overshoot marked.

So far the time response has been described by T_p, the time to first peak. With small overshoots this time can be difficult to measure so **Rise Time**, $\mathbf{T_r}$, is used instead. In this context, Rise Time is the time it takes for the step response to reach its final value for the first time. Obviously this only has meaning if there is an overshoot. Rise time is dependent on both damped frequency and damping factor. For a given damping factor, rise time is proportional to the inverse of damped frequency, which gives:

$$T_r \alpha \frac{1}{\omega_d}$$

Decay Time

Another characteristic of the response which has not been discussed is the time it takes for the oscillation to die away. This is called the **Decay Time**. This is an exponential with time constant $\tau_d = \zeta\omega_n$.

Fig 7.5 shows a number of step responses with the same ω_n and different values of ζ. These curves can be used to estimate a step response when the parameters are known.

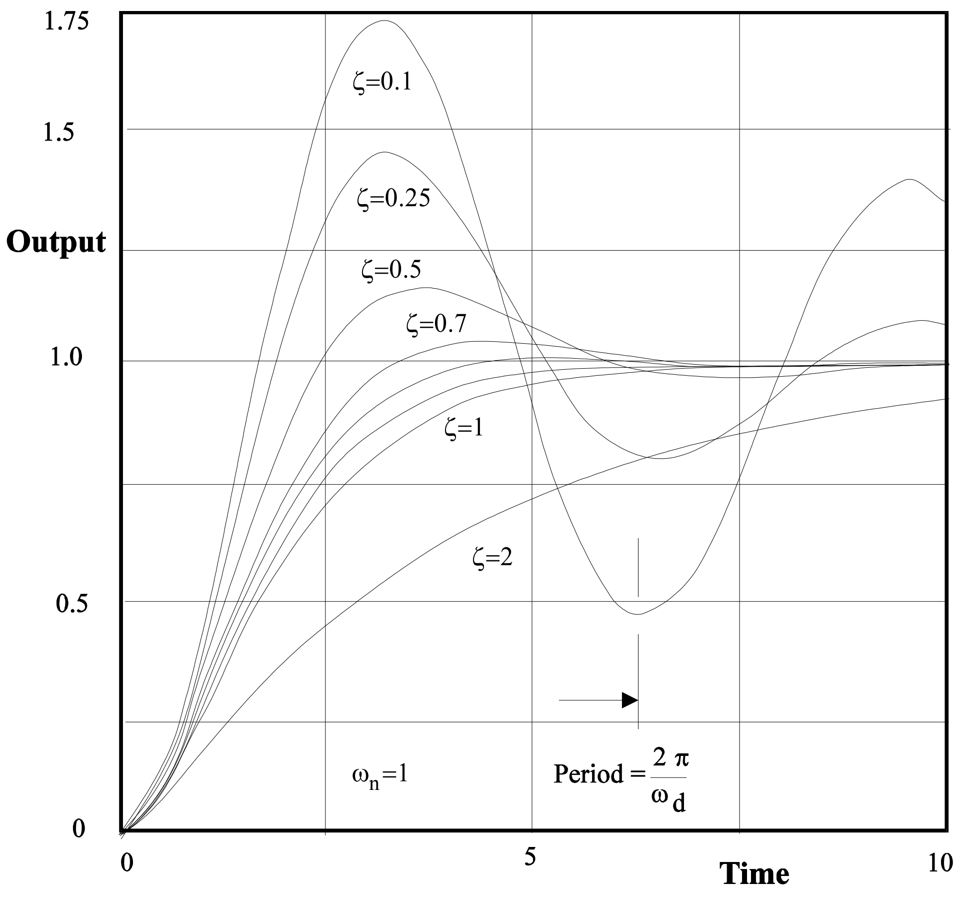


Fig 7.5 Second Order Step Response for Various Damping Factors

7.7 Second Order Frequency Response

From the step response it is not obvious why the Natural Frequency is important but it does become obvious when the frequency response is plotted.

Fig 7.6 shows the Bode Plot of the frequency responses of a number of second order systems all with the same natural frequency but with different damping factors.

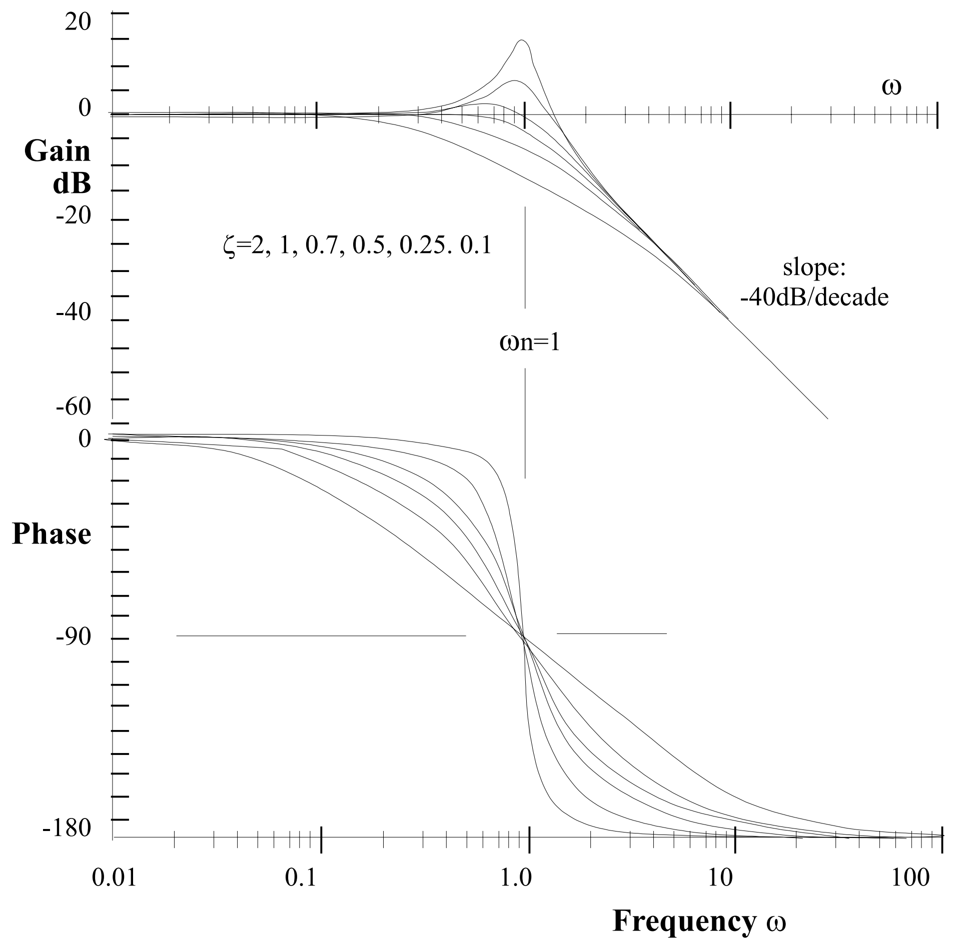


Fig 7.6 Second Order frequency Response

The main features of all the curves are:

1. At high frequencies the amplitude drops at -40 dB/decade along a line which crosses the 0 dB axis at ω_n.

2. The phase change when $\omega = \omega_n$ is -90° and tends to -180° at high frequency.

3. As the damping is reduced, the amplitude response peaks. The frequency at the peak is called the Resonant Frequency ω_r and is slightly lower than ω_n. This frequency is not important in control as it does not show until $\zeta < 0.7$ and most control systems are designed to have a higher damping factor than that.

It can be seen from these graphs that knowing ω_n and ζ allows the closed loop frequency response to be drawn.

7.7a The slope of the 2nd order frequency response at high frequencies cuts the 0 dB line at the:

[a] cutoff frequency ω_c. [b] damped frequency ω_d.

[c] natural frequency ω_n. [d] resonant frequency ω_r.

7.8 Summary

Second order servo systems can be described in terms of their Damping Factor ζ and Natural Frequency ω_n but specified values of both cannot be achieved with just proportional control.

Student Assessment 7

1. **The frequency of oscillation of a 2nd order response to a step input is called the:**
   - [a] cutoff frequency ω_c.
   - [b] damped frequency ω_d.
   - [c] natural frequency ω_n.
   - [d] resonant frequency ω_r.

2. **The time to first peak is a measure of:**
   - [a] cutoff frequency ω_c.
   - [b] damped frequency ω_d.
   - [c] natural frequency ω_n.
   - [d] resonant frequency ω_r.

3. **If the time to first peak is 0.5 minutes, is the damped frequency:**
   - [a] 0.21 rad/min.
   - [b] 0.105 rad/min.
   - [c] 0.105 rad/sec.
   - [d] 0.42 rad/sec.

4. **The high frequency slope of the 2nd order frequency response is:**
   - [a] -10 dB/decade.
   - [b] -20 dB/decade.
   - [c] -30 dB/decade.
   - [d] -40 dB/decade.

Chapter 8
Position Control with Velocity Feedback

Objectives of this Chapter

Having completed this chapter you will be able to:

- Set up a servo system to respond to step inputs using proportional gain and velocity feedback.
- Set up a servo system to respond to ramp inputs using proportional gain and transient velocity feedback

Equipment Required for this Chapter

- MS15 DC Motor Module
- AS3 Command Potentiometer
- CLIO Interface Module with PC Connection Lead
- System Power 90 Power Supply (or equivalent)
- 4 mm Connection Leads
- PC running VCL Virtual Control Laboratory Software

8.1 Velocity Feedback

In Chapter 7 it was shown that a system can be characterized by its Natural Frequency ω_n and its Damping Factor ζ.

The natural frequency ω_n is a measure of the speed of response of the system, the time to first peak being dependent mainly on ω_n. The damping factor ζ is an indication of how oscillatory the system is. It was also shown that, in a closed loop system, the proportional gain affected both ω_n and ζ. Increasing gain makes the system respond faster but also makes it more oscillatory. This results in greater overshoot and longer settling time.

What is required is a scheme whereby the proportional gain can be increased to speed up the system and another control introduced to increase the damping.

The additional control added is called **Velocity Feedback** (VFB). In the early days of electromechanical servo systems, it was easy to introduce velocity feedback as a voltage proportional to rotational velocity can be obtained from the signals associated with an electric motor. Nowadays a separate tachogenerator is more likely to be used to generate the velocity signal.

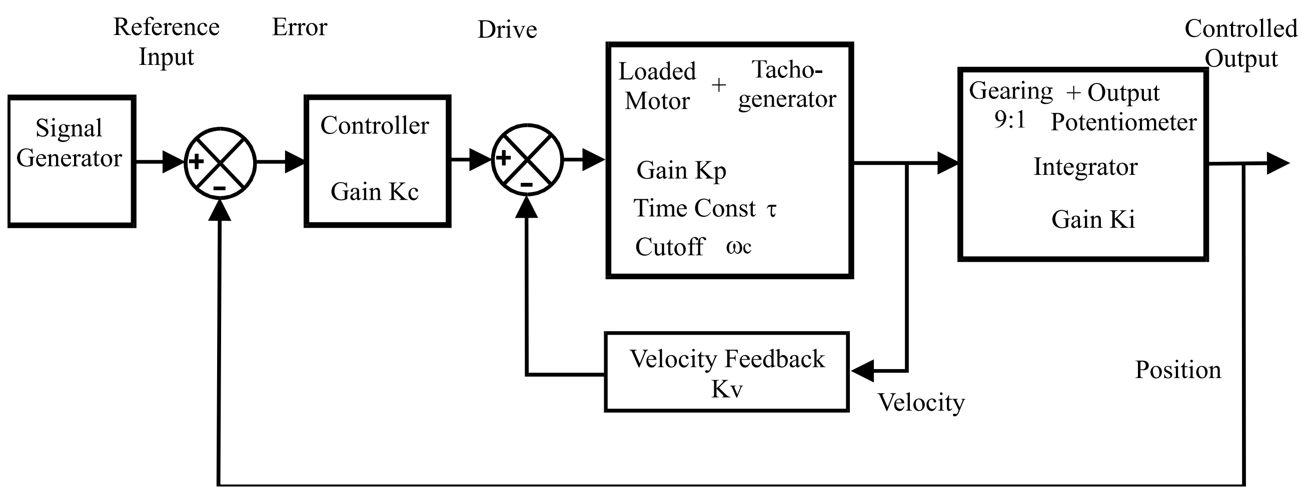


Fig 8.1 Servo System with Velocity Feedback

To influence the damping factor, the apparent time constant of the motor must be changed. In section 5.7 it was found that increasing the gain of the speed controller did reduce the time constant. Velocity Feedback uses this fact to produce an inner control loop with K_v being a time constant adjustment independent of the proportional gain. The block diagram of this is shown in Figure 8.1.

Start VCL and **load CA06PE08**. The setup will be:

File CA06PE08	**Controller** Servo	**Plant** MS15 Analog		**Display** Graph	
Signal Generator		**Graph**			
Signal	Step	1 Input	ON	5 Drive	ON
Level	10%	2 Position	ON		
Offset	0%	3 Error	OFF		
Rate	10 msec	4 Velocity	ON		
Reference	Internal				
DC Motor		**Output Potentiometer**		Engage	
Brake	0	**Command Potentiometer**		180°	

Switch on the system. The Gain K_c should be set to 2.5 and the Velocity Feedback K_v to 0 (Off). The *Magnify* control of the input and position traces have been set to 5.

The system is very underdamped. Observe that the drive signal (brown) does not go negative to brake movement until the position (blue) has gone above the input (dark blue). Selecting *Time* x4 will make this more obvious.

Now add Velocity Feedback by setting K_v to 1.00. The oscillations have been reduced. Observe that the drive now goes negative to brake the motor well before the output reaches its final value.

Increase K_v to 2.00. Braking now starts even earlier. Velocity Feedback has stabilized the system.

Adjust K_v until there is just a small overshoot and note the value in your workbook. This is usually considered the 'best' or 'optimal' setting for a particular gain.

Velocity Feedback for optimum setting K_v = ☐

8.1a **The best setting for velocity feedback gives:**

a a large overshoot.

b a small overshoot.

c no overshoot whatsoever.

8.2 Optimizing the Settings

Can the gain and velocity feedback now be adjusted to give the required performance for the system?

Increasing the gain should increase the speed of response. With a small overshoot, the speed of response can be measured by taking the time for the output response to reach its final value for the first time, i.e. when it crosses the input trace assuming both are on the same scale. This is called the Rise Time T_r. Since ω_n is proportional to $\sqrt{K}$ (from eqn 7.4) rise time will be proportional to $1/\sqrt{K}$, i.e. if K is doubled, rise time will fall by 0.7.

Using the x2, x4 or x8 time expansion controls will allow the rise times to be measured more accurately.

Set the gain to each of the values shown in Table 8.1. For each gain adjust K_v to give the same overshoot. Measure the rise time, calculate the expected rise time (using the Kc = 2.5 values and $1/\sqrt{K}$ proportionality) and enter the values into your workbook.

Gain K_c	**2.5**	**5**	**10**
VFB K_v			
Rise Time T_r seconds			
Expected Rise Time			

Table 8.1 Rise Time vs Gain

For a gain of 5, the Rise Time should agree with that expected but with a gain of 10 it may be higher than expected owing to drive saturation.

With velocity feedback to dampen the oscillations:

- Increasing the gain makes the system work faster - provided the drive signal does not saturate. Once the drive saturates the response time tends to remain constant.

- Increasing gain reduces steady state errors due to static friction and backlash.

However, owing to drive saturation, high gain will make the response to large changes differ from that to small changes. The gain and velocity feedback have to be optimized for the most important type of signal.

Having two controls allows two parameters to be controlled. Gain controls the speed of response and velocity feedback controls the amount of damping.

Load Sensitivity

With K_c = 10 and K_V at the setting to give optimum response, set the eddy current brake to position 2. Observe that there is little change to the overall response even though the extra load changes the gain and time constant of the plant. The response is now dominated by the loop gain and the velocity feedback.

8.2a **Experimentally, did the rise time agree with theory as the gain was increased:**

a over the full range of gain.

b at gains which do not cause drive saturation.

c at high gains only.

d at low gains only.

8.2b **Did adding the load:**

- [a] speed up the system.
- [b] have no affect.
- [c] slow the system down slightly.
- [d] slow the system down a lot.

8.3 Calculating K_c & K_v from Rise Time and Overshoot Specification

Often the specification for a system is given in terms of a required maximum %Overshoot and a maximum Rise Time.

Damping Factor can be obtained from overshoot by the formula:

$$\zeta = \sqrt{\frac{1}{1+\left[\frac{\pi}{\ln(\text{overshoot})}\right]^2}}$$

or from the curves in Figure 7.3.

It can be shown that the controller proportional gain, K_c, required for the specified time to first peak and damping factor is given by:

$$K_c = \left[\frac{\pi}{T_p}\right]^2 . \frac{1}{(1-\zeta^2)} . \frac{1}{K_p K_i \omega_c}$$

K_p, K_i and ω_c were measured in chapters 3 and 4.

The velocity feedback required is given by:

$$K_v = 2\zeta\sqrt{\frac{K_c K_i}{K_p \omega_c}} - \frac{1}{K_p}$$

Since K_V is dependent on K_C, K_C must be calculated first.

Design Exercise

Determine the gain and velocity feedback necessary for your servo system to have an overshoot of 10% and a time to first peak of 300 ms.

K_c = 　　　　　　K_V =

Enter these values into the controller and check measure the resulting overshoot and time to first peak with an input *Level* of 12%.

Plant Overshoot = 　　　　　　T_p = 　　seconds

Change the plant to **Plant | Servo**. Enter your model values into the plant gain, time constant and integral gain boxes. Measure the overshoot and time to first peak of the model.

Model Overshoot = 　　　　　　T_p = 　　seconds

The model values should be close to the specification but those for the plant may be a bit out. There are a number of reasons for this which would include errors in determining the model values and nonlinearities in the motor set.

The design calculations get you close. This is as good as you can expect.

8.3a **A servo system has a time constant of 500 ms, a gain K_p of 1 and an integrator gain K_i of 4. Determine the controller gain necessary for this servo system to have an overshoot of 5% and a time to first peak of 400 ms.**

K_c =

8.3b **Determine the velocity feedback required for the system in 8.3a.**

K_v =

8.3c **In the motor experiment, did the calculated values of K_c and K_v give:**

- [a] the response expected.
- [b] reasonably close to the response expected.
- [c] nowhere near the response expected.

8.3d **When the computer model was used, did the calculated values of K_c and K_v give:**

- [a] the response expected.
- [b] reasonably close to the response expected.
- [c] nowhere near the response expected.

8.4 Velocity Feedback with Ramp Inputs

Velocity Feedback is good for step inputs but is it good for ramp inputs?

Set the *Signal* to **Ramp** and *Level* = **50%**. Readjust the scale magnification of the traces 1 and 2 to *Magnify* = 1.

Set the gain K_c to 10 and the velocity feedback K_v to the value for the best step as recorded in Table 8.1. You should see that the position output follows the ramp but lags behind it.

Reduce K_v in steps of 1 and observe the output. The steady state lag should improve but the transient oscillation will get worse.

Velocity Feedback improves step responses but is not so helpful with ramp inputs where increase in VFB increases the following lag.

More complex arrangements are required if a system is required to follow a ramp.

8.4a **With the ramp input, what affect did increasing velocity feedback have on the steady state response?**

- [a] Increased the following error.
- [b] No effect.
- [c] Decreased the following error.

8.4b **With the ramp input, what affect did increasing velocity feedback have on the transient response?**

- [a] Increased the oscillations.
- [b] No effect.
- [c] Decreased the oscillations.

8.5 Summary

Velocity Feedback is an effective way of stabilizing servo systems which have to respond to step inputs. Increasing gain gives better response times and increases the stiffness. Increasing Velocity Feedback increases the damping thus reducing the oscillations.

Student Assessment 8

1. Adding velocity feedback:

[a] allows the system to be speeded up but causes oscillations.

[b] allows the system to be speeded up and removes oscillations.

[c] slows the system down.

[d] increases steady state error.

2. If the gain is increased by 3, the time to first peak will decrease by a factor of:

[a] 0.333 [b] 0.58 [c] 0.7 [d] 1

3. Which of the following would not cause errors in calculating values for K_c and K_v?

[a] Nonlinearities in the drive amplifier.

[b] Static friction in the motor bearings.

[c] Computer error.

[d] Experimental error in determining the model parameters.

Chapter 9
Three-Term or PID Control

Objectives of this Chapter

Having completed this chapter you will be able to:

- Define the purpose of Three-Term Control
- Explain the effects of Proportional Band, Integral Action and Derivative Action
- Outline how a PID controller is tuned

Equipment Required for this Chapter

- MS15 DC Motor Module
- AS3 Command Potentiometer
- CLIO Interface Module with PC Connection Lead
- System Power 90 Power Supply (or equivalent)
- 4 mm Connection Leads
- PC running VCL Virtual Control Laboratory Software

9.1 Three-Term Control

Servo control grew out of the need to control the targeting of artillery. Three-term control is a few decades older and was developed to control petrochemical plant. The original three-term controller (the Brown Pneumatic Controller/Recorder) was an intrinsically safe controller which could be used in hazardous locations. Examples of the modern version (manufactured by Honeywell) are still in use today. However, most three-term controllers are now electronic in operation but the terminology remains. Although the underlying mathematics are the same for three-term and servo control, sometimes different names are used for the same concept.

First we must look at the problems three-term control is trying to solve. Look back at Chapter 5 where speed control of the motor was discussed. The plant (the DC motor with speed output) is represented by its time constant and gain. There is no integral action in this type of plant.

It was seen that, with proportional control, there was always an error in the output. This error decreased as the gain increased but never disappeared. High gain also speeded up the transient response but could introduce unwanted overshoot and oscillations.

So, how is the steady state error eliminated without encountering oscillation and other stability problems? That is the problem which was solved by the original pneumatic three-term controller.

In Chapter 6 it was shown the integral effect between velocity and position eliminated the steady state error and in Chapter 8 velocity feedback was used to stabilize the system. Velocity is the rate of change (or the derivative) of position. So, having an integral effect in the forward loop is good for steady state error and having derivative action is good for stability.

The way in which the pneumatic controller worked placed the integral and derivative actions in parallel with the proportional element resulting in a three-term Proportional, Integral and Derivative (or PID) controller in the forward path of the control loop. Fig 9.1 opposite shows how this is configured.

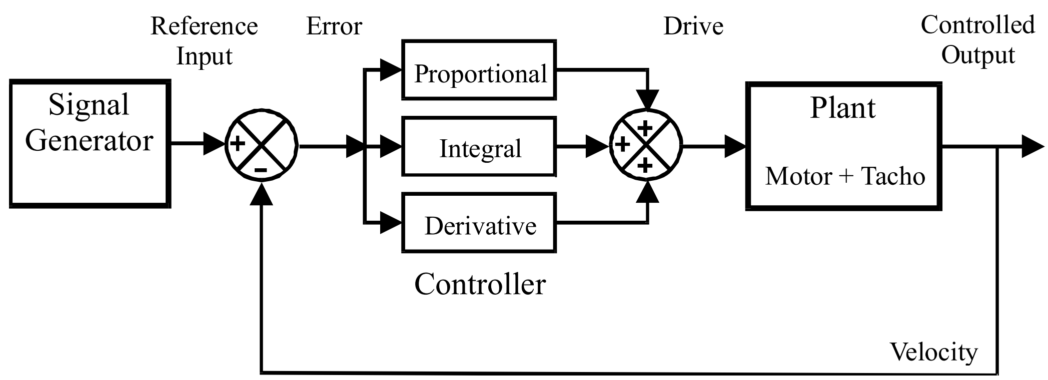


Fig 9.1 Block Diagram of a System with Three-Term Control

9.1a **PID control is introduced when:**

[a] there is natural integration in the plant.

[b] there is no natural integration in the plant.

[c] rate of change of output cannot be measured.

9.1b **In a system, increasing the gain:**

[a] reduces the response time.

[b] increases the response time.

[c] has no affect on the response time.

9.1c **Introducing an integrator in parallel with the proportional band:**

[a] increases the error.

[b] has no affect on error.

[c] decreases the error.

[d] eliminates the error.

9.1d **Introducing a differentiator in parallel with the proportional band:**

[a] increases oscillations.

[b] dampens oscillations.

[c] has no affect on the transient.

9.2 Proportional Control

Start VCL and **load CA06PE09**.

File	**Controller**	**Plant**		**Display**	
CA06PE09	PID	MS15 Analog		Graph	
Signal Generator		**Graph**			
Signal	Step	1 Input	ON	5 Drive	OFF
Level	50%			6 PB	OFF
Offset	0%	3 Error	ON	7 Integral	OFF
Rate	10 msec	4 Velocity	ON	8 Deriv	OFF
Reference	Internal				
DC Motor		**Output Potentiometer**		Disengage	
Brake	0	**Command Potentiometer**		180°	

Set Integral time constant (Itc) Off and Derivative time constant (Dtc) to 0. The controller is now proportional only.

The proportional control is marked PB. This stands for **Proportional Band**. Proportional Band is the inverse of gain. When expressed as a percent, $\%PB = \frac{100}{K_c}$. This nomenclature is a result of the origins of PID control.

Set PB to 100% ($K_C = 1$). Switch on the system. A considerable error can be seen between the input (ch1 dark blue) and the velocity (ch4 purple). Going back to Chapter 5, Eqn 5.6, the output will be K/(1+K). Since $K = K_c \times K_p$, and K_P is close to 1, the error will be around 50%.

Increase the controller gain to 2.5 by setting PB to 40%. Observe that the error has been reduced but is still considerable. You can also observe that the response time has decreased indicating that the system has speeded up.

Decrease the PB to 4% (controller gain = 25). As the gain is increased, the error reduces but the drive signal and the output get increasingly noisy.

The shortcomings of proportional control on a system without an integration effect were examined in Chapter 5.

Proportional Band on its own does not give adequate control. There is always an error and increasing the gain to reduce the error causes the plant to run rough.

9.2a **The principle use of Proportional Band in a PID controller is to:**

[a] reduce the transient response time.

[b] eliminate steady state error.

[c] dampen oscillations.

9.2b **If a controller has a proportional band setting of 20%, what is the value of the controller gain K_c?**

9.3 Integral Control

Position control of the servo did not have a steady state error to a step input because of the natural integration between velocity and position. It therefore seems that the way to achieve zero steady state error is to have an integrator in the forward loop.

The pneumatic controller placed the integrator in parallel with the proportional band. It was found that this configuration gave a more stable response than having the integrator in series with the proportional element.

- Decrease the input *Level* to 30% and set the PB to 40%. Now set the Integral time constant (Itc) to 1 second and click the On/Off box to bring in the Integral controller. Click channel 7 ON. Observe that, during the positive step, the output of the integral term (channel 7 yellow) now ramps upward with time as the error is integrated. This pushes the output closer to the required value.

- Decrease the Integral time constant until a good response is obtained. This should occur when the Integral time constant is of the order of 0.1s. The Integral action has removed the steady state error.

If the integral goes off scale and the system will not respond, click the Itc to Off then back to On. This resets the integrator.

The Integral time constant (or Integral Time) sets the response speed of the integrator. In some texts and installations, Integral Gain is used instead of Integral Time. Integral Gain is the inverse of Integral Time - the shorter the Integral Time, the higher the Integral Gain.

Although adding an integrator in this way does make the system more prone to oscillation, the closed loop system does not oscillate as readily as did the position system since the integrator in parallel with the proportional gain creates a dampening effect rather like that introduced by velocity feedback. As a result, many plants can be controlled by just the two terms, P+I.

9.3a **The principle use of Integral action in a PID controller is to:**

[a] reduce the transient response time.

[b] eliminate steady state error.

[c] dampen oscillations.

9.3b **Decreasing integral time will:**

[a] make the system more oscillatory.

[b] have no effect on the speed of response.

[c] make the system less oscillatory.

9.4 Derivative Control

Velocity Feedback was introduced to stabilize the position servo system. Since velocity is the rate of change (or the derivative) of position, velocity feedback can also be termed derivative feedback so derivative action may be required to stabilize the system.

The DC Motor is rather a simple system as it can be represented by a first order lag. As a result, the test rig being used does not exhibit the characteristics that makes it necessary to use derivative action. The effect of derivative action can be seen on a simulated system.

Make a note in your workbook of the PI settings you have determined for the DC Motor system.

PB = [%] Itc = [S]

Change the *Plant* to **Process**. The computer is now simulating a more complicated plant. Set PB = 30% and Itc = 0.24s. The Output should be showing an oscillatory transient. Increase the Derivative time constant (Dtc) until only a small overshoot can be seen. Derivative action works in a way similar to Velocity Feedback acting to stabilize the system - in theory. The plant being simulated is more like real process than the simple DC motor.

Make a note in your workbook of the derivative setting you have determined.

Dtc = [S]

Derivative Time Constant and Derivative Gain are directly related.

A problem with Derivative action that does not appear in the simulation is that high frequencies are amplified so that high frequency noise in the output can lead to a deterioration in performance.

Change **Plant** back to **Analog**, restore the PB and Itc settings to those recorded earlier and set Dtc to Off (0s). Reduce Itc by 0.02s. This should introduce some overshoot. Increase Dtc and observe the effect on the output and drive signals. With Dtc above 0.20s, the noise will dominate making the derivative action ineffective. Reset Dtc to Off (0s).

Do not allow the motor to run like this for long as it will damage the rig.

Measures can be taken to improve derivative action. It is actually impractical to build a pure differentiator - that would require infinite energy. Also, due to the lags in the plant, the plant does not change fast so the action of the differentiator can be limited to the expected rate of change of the output. This can be done in two ways. The differentiator is replaced with a high pass (or lead) filter which acts as a differentiator at low frequencies but not at high. Also, a low pass filter can be inserted into the derivative control path to filter out the high frequency noise from the tachogenerator.

9.4a **The principle use of derivative action in a PID controller is to:**

[a] reduce the transient response time.

[b] eliminate steady state error.

[c] dampen oscillations.

9.4b **Derivative action can cause a problem when:**

a the input is a random noise signal.

b the output measurement contains unwanted high frequency noise.

c the input does not change.

9.5 Tuning a PID Controller

If a mathematical analysis of a plant with three-term control is made, it is not at all obvious how the controller values can be calculated for a particular performance criterion.

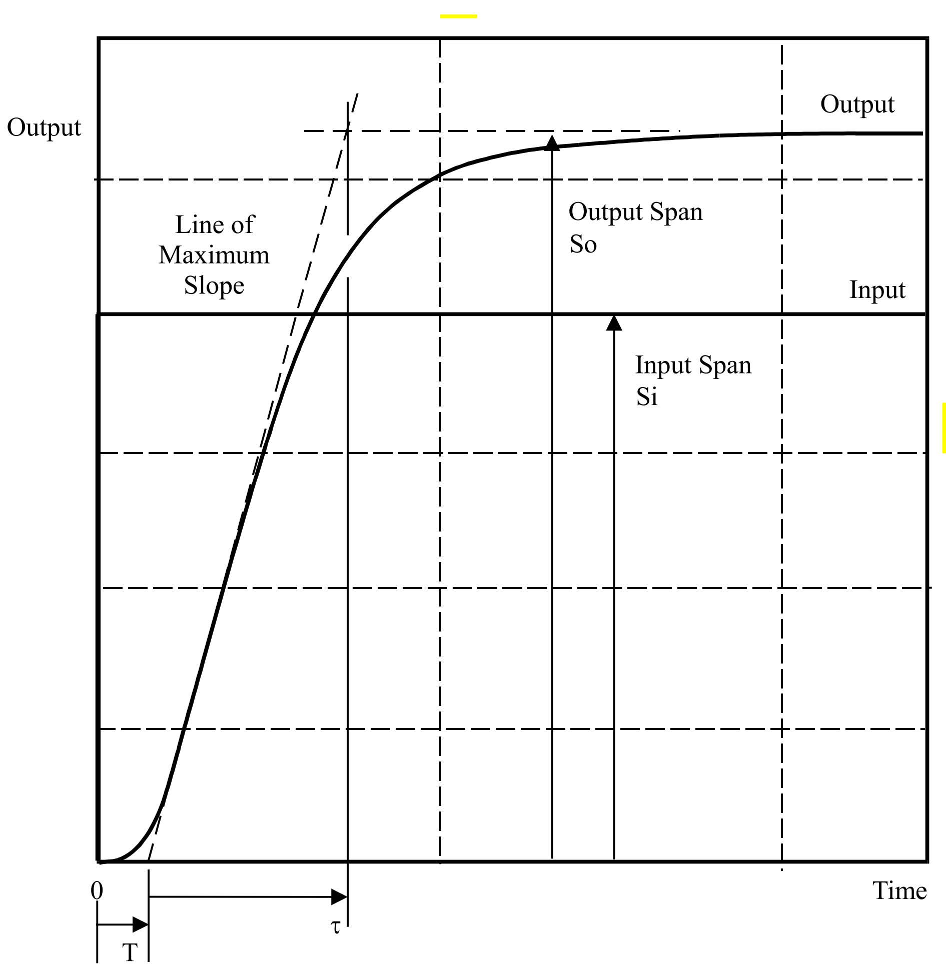
Setting the values for PB, Itc and Dtc for a plant can be done in a number of ways and this is called **Tuning the Controller**.

Many modern controllers are implemented using fast microprocessors. The more expensive models can use the normal operation of the plant to detect the error and adjust the parameters to suit the operating conditions.

Many less expensive models, used where the operating conditions of a plant are not changing, can have a computer (PC) plugged in via an RS232 serial communication link. When the plant is being commissioned, and from time to time during operation, the PC is plugged in to monitor and reset the settings. In a SCADA scheme where there is a supervising computer, the supervisor can initiate and perform the tuning operation from the control room.

Older plant may require the parameters to be set manually. This can be done by trial and error which is fine when the time constant is short, as in the experiments you are performing, but is tedious or impractical when the time constant is measured in minutes or hours. There is a simple rule-of-thumb procedure called **Ziegler - Nichols** but care must be taken with this as plant managers do not like engineers opening the loop and putting signals onto the plant.

To determine the PID settings using the Ziegler - Nichols tuning procedure, start by taking an open-loop step response of the system. (this can be dangerous). This will have the form shown in Fig 9.2 opposite.



So is the output span, Si is the input span
T is the time delay in seconds. It is the time at which the line of maximum slope crosses the initial value line.
τ is the time constant, in seconds, obtained from the line of maximum slope

Fig 9.2 Typical open-loop step response of a process.

Select **Controller | Open Loop** and **Plant | Process**. The signal generator should be set to *Signal* | Step, *Level* | 30% and *Rate* | 10 msec. Once a step had been drawn, freeze the picture and make the measurement - as detailed overleaf - required for the tuning calculation. Enter each of the measurements into your workbook.

Output Span - S_o

Measure the output span - the difference between the initial value and steady state value of the output.

Output span S_o = [Volts]

Input Span - S_i

Measure the input span - the difference between the initial value and steady state value of the input.

Input span S_i = [Volts]

Line of Maximum Slope - Time Constant τ

Using Line B and Slope controls, determine the line of maximum slope. It might be helpful to change channel 4 *Magnify* to 1/2 and *Time* to x2.

The time constant τ is the time the line of maximum slope takes to travel from the initial output value to the final output value.

τ = [seconds]

Time Delay T

The time delay T is the time at which the line of maximum slope crosses the initial value line.

T = [Seconds]

Controller Settings

The controller settings are then calculated from the table below.

Controller Type	Gain K_c	Integral T_I	Derivative T_D
P only	$\frac{S_i}{S_o} \times \frac{\tau}{T}$		
P + I	$0.9 \times \frac{S_i}{S_o} \times \frac{\tau}{T}$	$\frac{3.3 \times T}{K_c}$	
P+I+D	$1.2 \times \frac{S_i}{S_o} \times \frac{\tau}{T}$	$\frac{2 \times T}{K_c}$	$0.5 \times T \times K_C$

Table 9.1 Ziegler-Nichols Settings

For our PID example, calculate K_C (and hence PB%), T_I and T_D and enter the values in Table 9.2.

Controller Type	Gain K_C	PB%	Integral T_I	Derivative T_D
P+I+D				

Table 9.2 Controller Settings from Ziegler-Nichols Test

Unfreeze the display, restore the controller to PID and use the values calculated as the controller settings.

Do these settings give a good transient response? No, but Ziegler-Nichols was not designed to minimize the transient with a step input. However, the values calculated do get you into the correct region.

With PID, Ziegler-Nichols usually leads to a system which is too underdamped. Usually both integral and derivative time constants need to be increased.

Try setting T_I to 0.5τ and T_D to $0.5T_I$, leaving PB as calculated. This gives a much less oscillatory response.

In practice, techniques like this are used during plant commissioning to get approximate settings for the controller. Thereafter, the operators adjust the settings for what they consider to be best performance under their operating conditions.

In 90% of PID installations, the Derivative action is switched off. As has been shown, where there is a lot of noise in a system, derivative action tends to amplify the noise causing excessive wear on the actuators.

9.5a Did the Ziegler-Nichols values for PB, T_I and T_D lead to a system which was:

a underdamped.

b gave a good response.

c overdamped.

9.6 An Alternative form of the PID Controller

In describing PID, the parallel form of the controller has been used. This configuration separates the action of the three elements and makes it easier to understand what is taking place. However, there is an alternative configuration that was, in the days of pneumatic and analog controllers, easier to build. This serial configuration is shown in Fig 9.3.

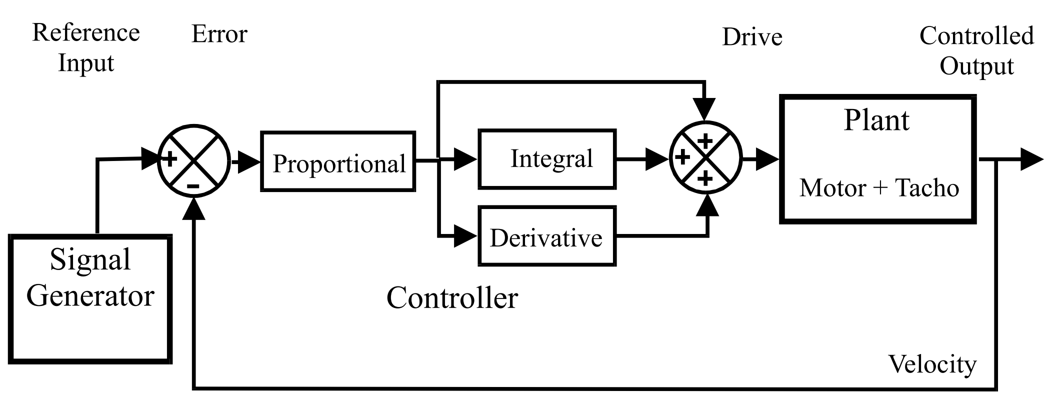


Fig 9.3 Serial form of PID controller

In this configuration, the gain precedes the integral and derivative terms. Most texts use the serial configuration, often without stating that they do so.

There is a simple relationship between the integral and derivative settings in the two different configurations.

Serial	**Parallel**		**Parallel**	**Serial**
t_i	$T_I = t_i/K_c$		T_I	$t_i = T_I \times K_c$
t_d	$T_D = t_d \times K_c$		T_D	$t_d = T_D/K_c$

Table 9.3 Serial And Parallel Integral And Derivative Times

For the serial configuration, the Ziegler-Nichols settings are shown in Table 9.4.

Controller Type	Gain K_c	Integral T_I	Derivative T_D
P only	$\frac{S_i}{S_o} \times \frac{\tau}{T}$		
P + I	$0.9 \times \frac{S_i}{S_o} \times \frac{\tau}{T}$	$3.3 \times T$	
P+I+D	$1.2 \times \frac{S_i}{S_o} \times \frac{\tau}{T}$	$2 \times T$	$0.5 \times T$

Table 9.4 Ziegler-Nichols Settings - Serial Configuration

The alternative settings for the serial configuration then becomes:

$t_i = 0.5\tau \times K_c$ and $t_d = 0.25\tau$

9.6a **There are two forms of PID controller. Provided that the settings are adjusted properly, do both give the same control action?**

Yes or No

9.7 Summary

Three-term or PID control is widely used in the process industries. Proportional Band defines the gain of the loop, Integral action eliminates steady state error after a change in input or load and Derivation action can stabilize the system.

The controller may be tuned manually or, with modern microprocessor based controllers, by the controller itself or by an external computer.

Student Assessment 9

1. A PID controller is introduced to:

- a increase speed of response by decreasing Proportional Band.
- b eliminate steady state error by introducing Integral action.
- c increase stability by introducing Derivative action.
- d all of the above.

2. A parallel PID controller has the settings - PB = 30%, T_I = 0.5 minutes, T_D = 0.24 minutes. The equivalent settings for a serial PID controller would be?

- a PB = 30%, t_i = 0.167 minutes, t_d = 0.72 minutes.
- b PB = 30%, t_i = 1.67 minutes, t_d = 0.07 minutes.
- c PB = 90%, t_i = 1.67 minutes, t_d = 0.72 minutes.
- d PB = 30%, t_i = 0.167 minutes, t_d = 0.07 minutes.

3. Modern microprocessor based PID controllers are tuned using:

- a PC linked via RS232 serial lines.
- b SCADA supervisory computer.
- c internal routines.
- d any one of the above.

Chapter 10
Stability

Objectives of this Chapter

Having completed this chapter you will be able to:

- Show how positive feedback can occur in a negative feedback system
- Explain Gain Margin and Phase Margin
- Show that transport lag adds phase and destabilizes a system

Equipment Required for this Chapter

- MS15 DC Motor Module
- AS3 Command Potentiometer
- CLIO Interface Module with PC Connection Lead
- System Power 90 Power Supply (or equivalent)
- 4 mm Connection Leads
- PC running VCL Virtual Control Laboratory Software

10.1 Positive Feedback

So far we have examined the effects of negative feedback. What happens if the system is connected so that the feedback is positive? Intuitively, if the output is added to the input then the error is increased, not decreased, and the system runs away (is unstable). This can happen but it is not as simple as that.

The block diagram of a system with positive feedback is shown in Fig 10.1. The forward loop gain is K and the time constant is, as before, τ.

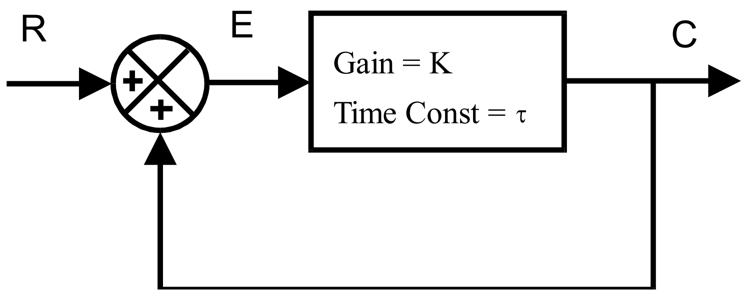


Fig 10.1 System with positive feedback

We will build such a system and see what happens.

Start VCL and **Load CA06PE10**.

File CA06PE10	**Controller** Proportional	**Plant** MS15 Analog	**Display** Graph
Signal Generator		**Graph**	
Signal	Step	1 Input ON	5 Drive OFF
Level	25%		
Offset	0%	3 Error ON	
Rate	100 msec	4 Velocity ON	
Reference	Internal		
DC Motor		**Output Potentiometer**	Disengage
Brake	0	**Command Potentiometer**	180°

The motor is now configured as a proportional speed control system. Set the Gain K to 0.6. The system will not run well with these settings.

Now change to positive feedback by clicking the summing circle. The - will change to + giving positive feedback.

Observe that the output speed is stable even though there is positive feedback.

Increase the gain to 0.7 then to 0.8 observing that, as the gain is increased, the motor speeds up. This is not a good control system but it is stable.

To keep the drive within the screen limits, reduce the step level to 15% then increase the gain to 0.9. The output is still stable.

Now increase the gain to 1.0. After the step change, the drive and speed can be seen to be increasing with time. The system is now unstable. Increase the gain to 1.1 and the instability will be more obvious.

Disable the motor.

10.1a In the stability experiment, instability was observed when:

a the feedback was negative and the control gain was greater than 1.

b the feedback was negative and the control gain was less than 1.

c the feedback was positive and the control gain was greater than 1.

d the feedback was positive and the control gain was less than 1.

10.1b Positive feedback gives a good control system:

Yes or No

10.2 Stability

Obviously positive feedback is not a good idea so why is it being introduced?

From earlier chapters, you will know that a plant can be represented by a number of lags. In Chapter 4 it was shown that a lag consists of an amplitude response and a phase response. At high frequencies, the phase response of a single lag approaches -90°, such that a high frequency sine wave would lag by 90°. A second order plant could lag by up to 180° and a third order system by up to 270°, each additional lag contributing -90°.

What happens to a sine wave which is phase shifted by -180°? It is inverted since $\sin(\theta - 180) = -\sin\theta$.

So, if the plant can phase shift a signal by -180°, negative feedback has been turned into positive feedback and there may be a stability problem.

The experiment showed that a positive feedback system could be stable provided that the gain is low. Mathematical analysis can show that a system will be stable provided that the amplitude response is less than 1 at the frequency when the phase lag is 180°.

Stability - A negative feedback system will be stable if, when the open loop phase lag is 180°, the open loop amplitude ratio (or gain) is less than 1.

From this, it is obvious that plant with only one or two lags cannot be unstable as the phase lag never exceeds 180°.

There are a number of ways of assessing the stability of a plant, some mathematical and some graphical. The most useful for our purposes is the Bode Diagram which was introduced in Chapter 4.

10.2a **A negative feedback system will be unstable when:**

a the open loop gain is less than 1 when the phase lag is 180°.

b the open loop gain is 1 or greater when the phase lag is 180°.

c the phase lag is less than 180° when the open loop gain is greater than 1.

10.2b **Is it true that a process characterized by two lags can never be unstable:**

Yes or No

10.3 Gain Margin and Phase Margin

Fig 10.2 opposite is the Bode Diagram of a third order plant.

From the graph, when the phase is -180°, the gain is -8 dB. The **Gain Margin** is thus 8 dB - the amount by which the gain can be increased before the closed loop system will go unstable.

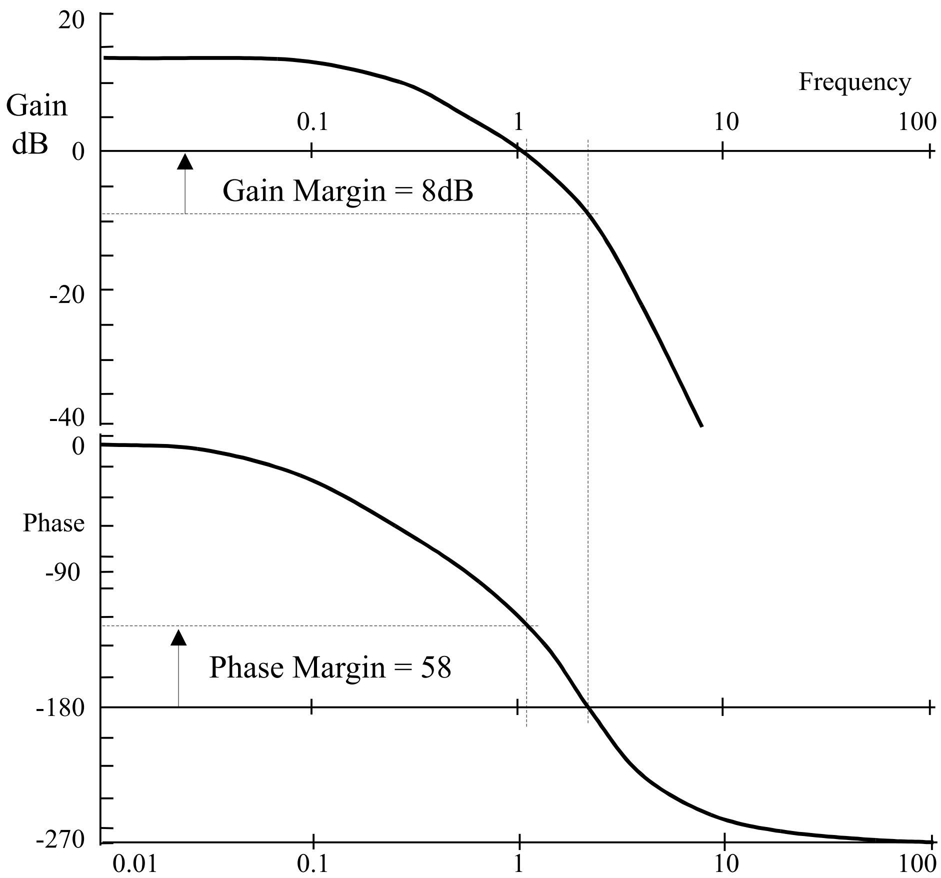


Fig 10.2 Bode Diagram showing Gain and Phase Margin

Gain Margin is the amount by which the gain is less than 1 when the phase is -180°

When the gain is 1 (0 dB), the phase is -122°. The **Phase Margin** is then -122°-(-180°) = 58°.

Phase Margin is the amount by which the phase is greater than -180° when the gain is 1 (0 dB).

The gain of a system is under our control so it is useful to know the Gain Margin. The use of the Phase Margin is not so obvious.

10.4 Transport Lag

In many systems there can be a transport lag. This can happen when the measurement of the output is delayed. Consider trying to control the amount of chocolate being added to a biscuit (or cookie). The biscuits are weighed on the conveyor belt before the chocolate is added and again after the chocolate has been poured but this second weighing has to wait until the biscuits have cleared the chocolate pouring area. There is thus a delay between adding the chocolate and weighing the amount added. The delay time, or transport lag, depends on the speed of the conveyor belt and the distance between the adding and weighing operations.

If you consider that a delay of 1 period is the equivalent of a phase shift of 360° then the delay θ can be written in terms of radian frequency as:

$\theta = -\frac{180\omega T}{\pi}$ degrees where T is the time delay in seconds.

The phase of a transport lag can be added to the Bode Plot and from this the minimum speed of the conveyor belt can be determined to ensure that the control system is stable.

Since transport lag effectively adds phase lag to a system, it makes it easier for a phase lag of 180° to be reached while the gain is greater than 1. Transport lag thus has the tendency to destabilize systems.

This can be seen using the simulated plant. Change **Controller | PID**, **Plant | Process**, *Level* = **25%** and *Rate* = **10 msec**. Set the controller settings to the Ziegler-Nichols values which were determined in Chapter 9 to give a reasonable response.

In the Plant area, the *Sample Delay* control adds transport lag. Increase the delay and observe the effect on the step response. At what delay does the system become unstable owing to the extra phase lag introduced by the transport delay?

System becomes unstable with transport lag delay = [ ]

Set the delay to 7 and vary the controller settings to give good control of this plant. To start with, reduce the gain a bit (increase PB).

Note the controller settings.

PB = [%] Itc = [S] Dtc = [S]

Compare these settings with those used in Chapter 9 for the plant with delay 2. The most noticeable change is that the gain has had to be reduced. This is in line with the stability theory. Increasing transport lag decreases gain margin while reducing the controller gain increases the gain margin.

10.4a **Is it true that transport lag adds phase lag:**

[Yes] or [No]

10.4b **To maintain stability with increased transport lag, the gain should be:**

[a] increased.

[b] left unchanged.

[c] decreased.

10.4c **At what delay did the experimental system go unstable?**

Student Assessment 10

1. Instability occurs when:

- a there is any positive feedback.
- b the open loop gain is greater than 1 when the phase lag is 180°.
- c the open loop gain is less than 1 when the phase lag is 180°.
- d when the open loop gain is greater than 1.

2. Gain Margin is:

- a the amount by which the gain has to be increased to cause instability.
- b the difference between the low frequency amplitude ratio and the amplitude ratio when the phase lag is 180°.
- c the difference between the amplitude ratio at 90° lag and 180° lag.
- d the difference between the amplitude ratio at 90° lag and 270° lag.

3. Phase Margin is:

- a the difference between the low frequency phase lag and the lag when the amplitude ratio is 1.
- b the amount by which the phase lag can be increased before instability occurs.
- c the difference between 180° lag and the high frequency lag.
- d the difference between the lag when the amplitude ratio is 1 and the lag at high frequency.

4. Transport lag:

- a increases the open loop gain.
- b decreases the open loop gain.
- c increases the phase lag.
- d decreases the phase lag.

Chapter 11
The Use of Computers for Control

Objectives of this Chapter

Having completed this chapter you will be able to:

- Describe the advantages and disadvantages of using computers in real-time control
- Define what is meant by Direct Digital Control, Multi-loop systems and SCADA

11.1 Computers in Control

Computers of all types are used extensively for control purposes. The applications range from using a network of workstations controlling a large process plant to using a simple single chip microprocessor in a washing machine.

Advantages

Whatever the scale of operation, the prime advantages of using computers are:

- Control actions are not limited to the behavior of electronic circuits.
- The computer system can often service more than one item of plant by time sharing.
- The computer can be programmed to respond to changing conditions as and when they arise.
- Information can be stored or transmitted for performance analyses, quality control or statutory archive purposes.
- System tuning can be performed on-line. This is called **Adaptive Control**.
- **Knowledge Based** techniques can be employed where a data bank of information about the plant is built up and control decisions are made based on this knowledge. This is sometimes called **Artificial Intelligence** or **AI**.
- Control decisions can be made on incomplete analysis of the situation. This is called **Fuzzy Logic**.
- Built in self diagnostic programs can identify hardware and software faults. This becomes increasingly important as systems get more complicated and maintenance and testing become a major problem.

Disadvantages

Despite these advantages, there are several potential disadvantages which must be considered:

- Development costs, particularly in software, can be high.
- The computers, power supplies and interfaces must be built to a high standard of reliability. In critical cases, multiple systems with majority voting must be used.

- Where a computer is handling several tasks, failure to perform certain tasks within a certain (usually quite short) time frame can make the system unstable. This requirement makes the real time operating systems required for control different from the operating systems used in commercial operations.

Applications

Despite these problems, the computer represents the only way of obtaining the sophistication required for many modern control schemes. Computers themselves are extremely reliable and relatively inexpensive. It is the increase in computing power and speed and the fall in cost which has led to the escalation of the use of computers in control systems and other real-time applications.

There are, mainly, three levels at which computers are involved with control. These are:

- Direct Digital Control
- Multi-loop Control
- Supervisory Control

These three schemes are detailed below but there are many other ways in which digital control schemes can be organized.

11.1a Which of the following is not an advantage of digital control?

[a] control actions are not limited to the behavior of electronic systems.

[b] information can be stored for performance analyses.

[c] there is a high software development cost.

[d] system tuning can be performed on-line.

11.1b Which of the following is a disadvantage of digital control?

[a] the computer can be programmed to respond to changing conditions.

[b] unreliability of complex computers.

[c] knowledge based techniques can be employed.

[d] built in self diagnostic programs can identify faults.

11.2 Direct Digital Control

This is the simplest level at which computers are involved. Here a small computer, often a single chip microprocessor, replaces the electronics of an analog controller and, as far as the control action is concerned, behaves like an analog controller. Fig 11.1 shows the schematic of such a system.

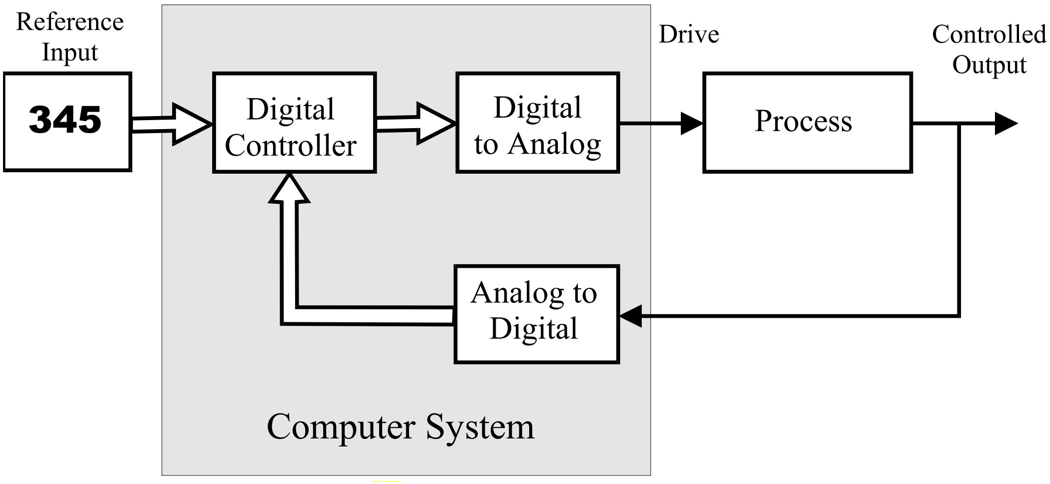


Fig 11.1 Schematic of a Direct Digital Control System

The plant signals are measured using analog or digital techniques. The drive signal is calculated by the microprocessor and used to drive the plant.

This type of system and the interfacing techniques involved are described in later chapters of this manual.

The advantages of such controllers are that the same hardware can be used to deliver a wide range of control schemes, the differences between applications being in the software, not the hardware. It is also much easier to provide digital readouts and user friendly setup controls when a microprocessor is used.

The digital controller may also perform tuning functions and, with a suitable network interface, report operations to a central supervisory computer. It may also receive setup instructions from the supervisory computer. Reporting may consist of regular logs (status reports) and exception reports when the controller variable is in an alarm condition.

11.3 Multi-loop Digital Control

In the early days of computers in control, multi-loop control was popular since a company could not afford to dedicate an expensive and physically large computer to a single loop. Fig 11.2 shows a schematic of a multi-loop system.

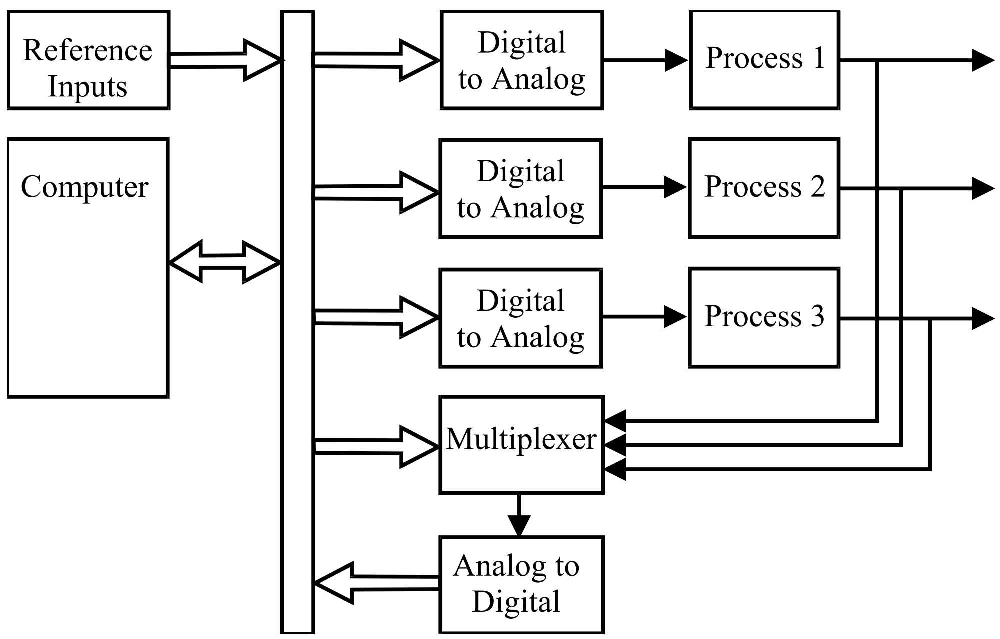


Fig 11.2 Schematic of a Multi-loop Digital Control System

The outputs of a number of processes within a plant were monitored in sequence using multiplexing techniques. The control drive necessary was calculated and output to the appropriate actuator. With the availability of inexpensive microprocessors, this use of computers has died out except where a number of loops interact and must be controlled together.

Interactive multi-loop schemes can require powerful computing facilities but such systems are outside the scope of this course.

11.4 Supervisory Control - SCADA

SCADA stands for Supervisory Control And Data Acquisition. Most modern process plants are controlled using a SCADA system as outlined in Fig 11.3.

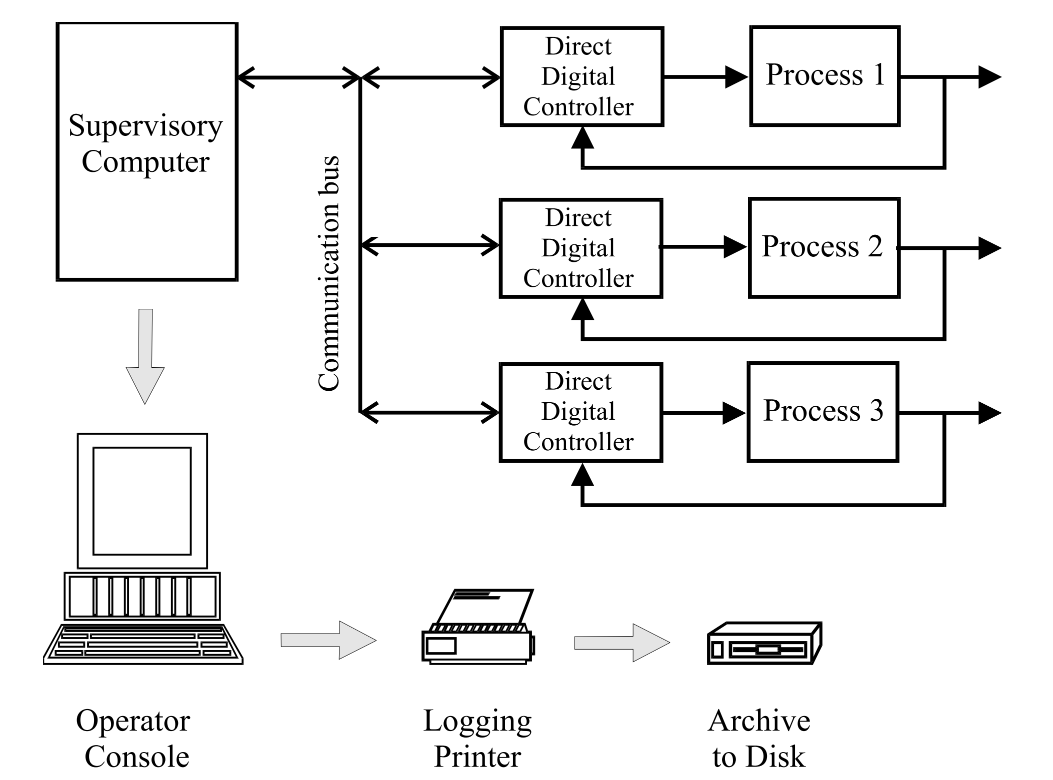


Fig 11.3 Outline of a SCADA System

In a SCADA scheme, a supervisory computer oversees the operation of a number of direct digital controllers. Operators in a central control room view the plant operation on computer displays, initiating and responding to plant conditions via the mouse, keyboard and touch screens but the direct digital controllers perform the actual control actions.

SCADA schemes can range from a single workstation supervising a small number of individual controllers to large multi-processor, multi-workstation networked systems controlling a large process such as a refinery or power station.

11.4a **Direct digital control replaces an analog controller with a microprocessor based system?**

Yes or No

11.4b **Which of the following can be an extra advantage of using direct digital control?**

a more user friendly user interface.

b internal or external digital tuning.

c performance logging to supervisory system.

d all of the above.

11.4c **Multi-loop digital control systems allows a computer to be shared between a number of control loops?**

Yes or No

11.4d **SCADA systems are used in large processes to supervise a number of direct digital controllers?**

Yes or No

11.5 Computer Control in this Course

Supervisory systems and multi-loop systems are beyond the scope of this course. The main emphasis in the following chapters will be on Direct Digital Control but much of the discussion on interfacing techniques and programming considerations does have relevance to all aspects of computers in control.

You may have noticed that you have been using a computer for control throughout this course although the subject under discussion has been linear analog control. If a computer is fast enough and accurate enough it can be made to behave as if it were an analog controller and that is how the PC has been used so far.

In the rest of the course we shall introduce the digital techniques used in single loop control. We shall cover:

- **Analog Interfacing** how real-time analog voltages are read and generated by a computer.
- **Control Computation** how the control signals are calculated.
- **Digital Interfacing** how digital techniques can be used in measurement and actuation.

Student Assessment 11

1. **Which of the following statements is not true? Using digital computers in control systems:**
   - [a] gives greater flexibility of control strategy.
   - [b] allows easier supervision of the overall process.
   - [c] allows changes to be made as conditions change.
   - [d] freezes the control strategy when the process is commissioned.

2. **A process with a few loops is most likely to use:**
   - [a] analog controllers.
   - [b] direct digital control.
   - [c] multi-loop control.
   - [d] SCADA system.

Chapter 12
Analog Interfacing

Objectives of this Chapter

Having completed this chapter you will be able to:

- Describe how computers handle numbers
- Explain the workings of a Digital to Analog Converter
- Explain the operation of a Successive Approximation Analog to Digital Converter
- Describe the effects of amplitude and time quantization of signals

Equipment Required for this Chapter

- CLIO Interface Module with PC Connection Lead
- System Power 90 Power Supply (or equivalent)
- 4 mm Connection Leads
- PC running VCL Virtual Control Laboratory Software
- 4-digit Digital Voltmeter (optional)

12.1 Analog Interfacing

Most of the signals being measured are continuous in nature. The velocity of the motor does not change in steps nor does the shaft position. These signals are analog in nature.

A computer deals in digits. Collections of on/off levels are used to represent discrete numbers.

Analog Interfacing is the technology of converting continuous analog signals to and from discrete digital numbers.

12.2 Computer Representation of a Number

Before looking at the conversion process, we need to know how numbers are represented within a computer.

In the decimal system we are all used to, there are 10 digits labeled 0-9. To create a number larger than 9, the digits are arranged in columns with each column to the left representing a number 10 times that to its right. Thus, in decimal, the number 128 is a shorthand for (1x100)+(2x10)+(8x1). This can also be written as $(1 \times 10^2)+(2 \times 10^1)+(8 \times 10^0)$.

Since a logic gate or memory cell has two levels - On/Off, 5v/0v, 3v/0v etc - the binary system of number representation is used by digital computers. The two logic states represent the numbers 0 and 1 and each column to the left is twice that to its right. The binary number 1101 can be written as (1x8)+(1x4)+(0x2)+(1x1) or $(1 \times 2^3)+(1 \times 2^2)+(0 \times 2^1)+(1 \times 2^0)$ so represents the same value as 13 decimal.

Each digit (0 or 1) is called a **bit**. Within the computer, bits are stored and manipulated in groups of 8 bits. A group of 8 bits is termed a **byte**. The number of bits normally handled together by the computer is called a **word**. The word size is used as a measure of the power of a computer. The PC has a word size of 32 bits and is thus called a 32-bit computer. Single chip microprocessors such as the PIC and MSC51 ranges have 8-bit words. There are other microprocessors and microcomputers which have 16-bit words.

Hardware and software routines in the computers blur this distinction. An 8-bit machine will have double word instructions and 32-bit machines have byte and half word instructions. The computer that is chosen for a particular application is a compromise between speed, power, space and cost.

Table 12.1 shows the range of decimal numbers which can be handled by each of the word lengths. Negative numbers can be represented in binary form by a technique called 2's complement which allows easy addition and subtraction of binary numbers.

Word length	Unsigned	Signed
8	0 to 255	-128 to 127
16	0 to 65536	-32768 to 32767
32	0 to 4294967296	- 2147483648 to 2147483647

Table 12.1 Numbers represented by computer word lengths

An 8 bit number has a resolution slightly better than 0.4% such that 1 bit represents 1/256 or 0.004 (0.4%) of the total range. This is accurate enough for many systems such as oven temperature control or washing machine water level control but most industrial systems require better accuracy than this. For most industrial direct digital control systems, a 16 bit word is adequate. Generally an overall accuracy of 0.1% is acceptable. This is equivalent to a 10-bit (1/1024) resolution.

12.2a What is the decimal value of the binary number 10011100?

12.2b How many Bits are represented by a Byte?

a 1 b 8

c 16 d 32

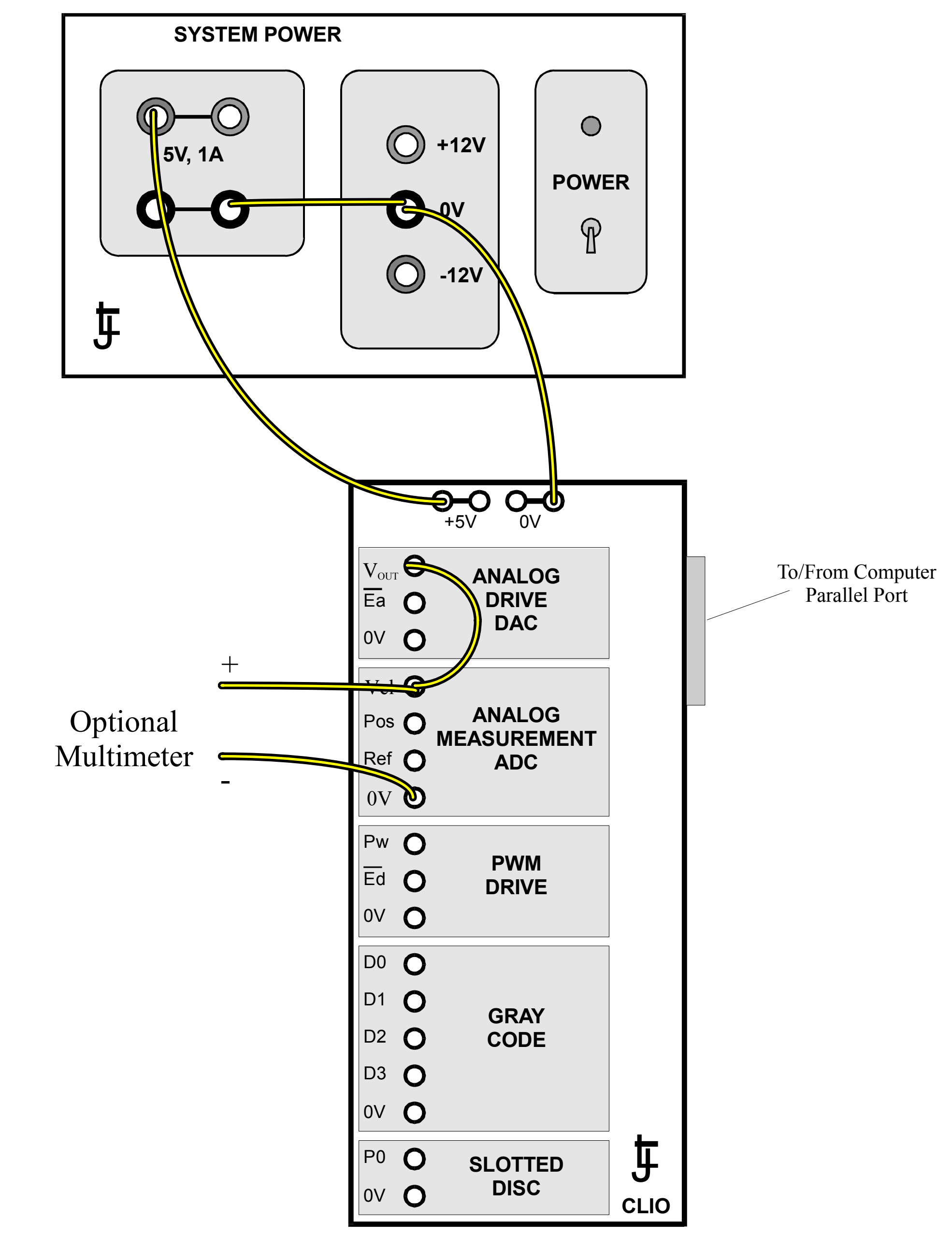


Fig 12.1 Connections for conversion tests

12.3 Digital to Analog Conversion

As some of the analog to digital techniques rely on having a digital to analog converter as part of the conversion technique, digital to analog conversion is described first.

Wire the system as shown in Fig 12.1. Although the DC Motor is not used in this chapter, the power wiring may be left connected for later experiments.

Start VCL and **Load CA06PE12**.

File CA06PE12	**Controller** DAC	**Plant** MS15 Analog	**Display** Meter
Signal Generator		**Graph**	
Signal	DC-Level		
Level	0%		
Offset	0%		
Rate	100 msec	4 Velocity ON	
Reference			
DC Motor		**Output Potentiometer**	Disengage
Brake	0	**Command Potentiometer**	180°

On the CLIO interface board, **Vout** is connected to the **Vel** input so channel 4 of the panel meter displays the voltage that the DAC is outputting to the Vout socket. If you wish to monitor the output directly, a 4 digit digital voltmeter can be connected to the Vout pin with the range selected so that the least significant digit displays 1 mV.

The mimic shows a 6-bit Digital to Analog Converter (DAC) register. The box beneath the DAC register shows the voltage (millivolts) expected to be outputted by the DAC.

The task is to take a binary digital word and from it create a voltage proportional to the value of the word. In this case, the least significant digit (lsd) corresponds to 80 mV.

Make sure all digits are at 0 - clicking within a digit box toggles the bit between 0 and 1. The output should be 0 volts but, on the on-screen digital display, there may be a small offset owing to errors in the measurement system. Click the lsd - the box on the right - and the output should change to 80 mV.

Click the other bits in turn as shown in Table 12.2 to determine the voltages generated by each of the other bits. Enter the results in Table 12.2 in your workbook.

bit 5	bit 4	bit 3	bit 2	bit 1	bit 0 80mV	Decimal Value	Expected Output	Voltage Reading
0	0	0	0	0	0	0	0	
0	0	0	0	0	1	1	80mV	
0	0	0	0	1	0	2		
0	0	0	1	0	0			
0	0	1	0	0	0			
0	1	0	0	0	0			
1	0	0	0	0	0			
1	0	0	0	0	1			
1	1	1	1	1	1	63	5.04V	

Table 12.2 DAC outputs

Other patterns may be tried to confirm that the output voltage is the sum of the weighting of each of the bits set.

This example is for a unipolar 6-bit DAC with a reference voltage of 5.12 volts. Because there are $2^n = 2^6 = 64$ steps, the **Resolution** is:

$$\text{Resolution} = \frac{\text{Reference}}{\text{Steps}} = \frac{5.12}{64} = 80\text{mV}$$

The **Range** is 0 to $[(2^n - 1)\text{x}resolution]$ = 0 to [63*80] mV = 0 to 5.04 volts.

Weighted Resistors

DAC's were originally built using a weighted resistor chain. This is shown in Fig 12.2 opposite.

Each resistor multiplies the reference by a different amount and each adds to the DAC output. The problem with this scheme is the wide range of resistor values required and their accuracy. Producing such a range of matched resistors is an expensive process and not well suited to semiconductor production techniques.

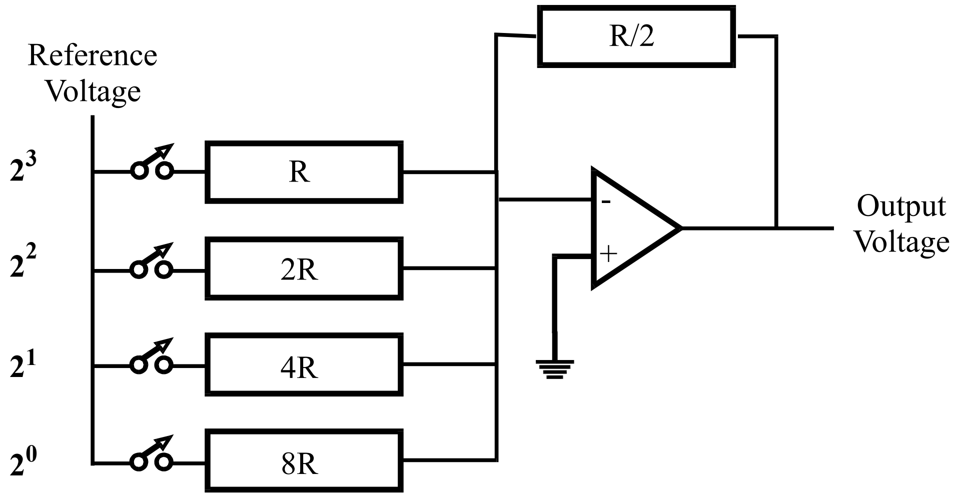


Fig 12.2 Weighted resistor DAC

R-2R Ladder

Another technique used is the R-2R ladder network. This is shown in Fig 12.3.

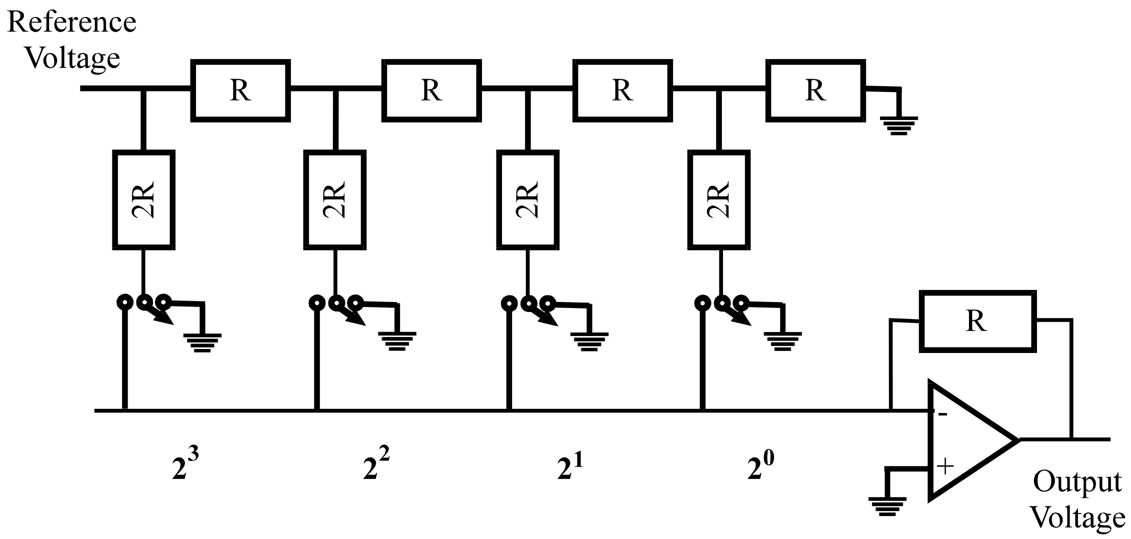


Fig 12.3 R-2R DAC

With this configuration, the current flowing in each rung of the ladder is half that of the rung to the right. The currents in rungs set to 1 are summed in the operational amplifier to form the output voltage. Since only two resistor values are used, this configuration is much easier to produce especially on a semiconductor chip.

The DAC used in the CLIO interface unit is a 12 bit R-2R unit. The reference voltage is 4.096 volts. The output is amplified by a factor of 2.5 and offset by a negative DC voltage (-5.12V) to give a bipolar output (an output capable of both positive and negative values). This gives a resolution of (4.096x2.5)/4096 = 2.5 mV. The range is from -5.120V to +5.1175V.

12.3a **If a DAC has a resolution of 2.5 mV, what voltage (in millivolts) will be generated if the DAC input is set to 1010011?**

12.3b **If this is a 12 bit DAC, what (in volts) is the full scale output of the DAC?**

a 10.2300 V b 10.2325 V c 10.2350 V d 10.2375 V

12.3c **Originally weighted resistor DACs were used. Which of the following contributed to the unpopularity of this technique?**

a difficult to match the wide range of resistor values required.

b difficult to trim the resistors to the accuracy required.

c difficult to lay down the range of resistors on integrated circuits.

d all of these.

12.4 Analog to Digital Converter

Converting from analog to digital is a bit more involved. The most used technique is called Successive Approximation. The schematic of a 4 bit Successive Approximation ADC is shown in Fig 12.4 opposite.

Click on **Controller|ADC**. The mimic now shows a 6 bit Successive Approximation Register. You can change the values of this register so, to explain the workings of this type of ADC, you become the decision maker in the process.

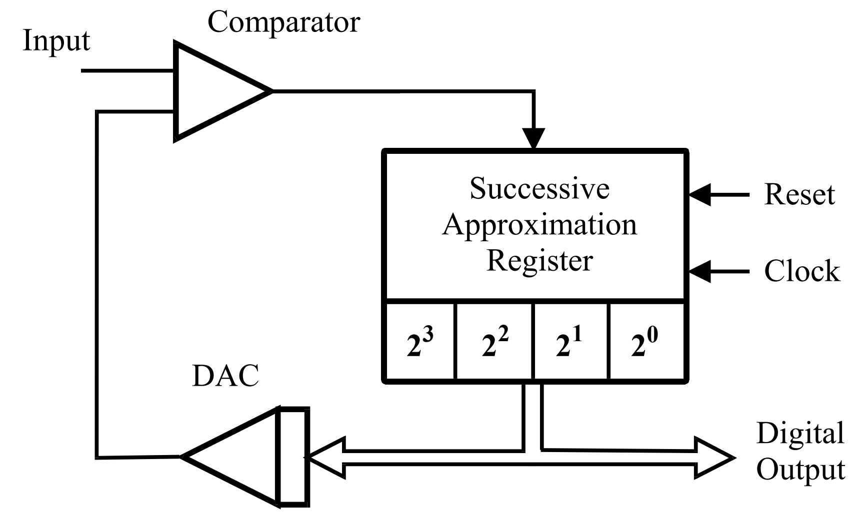


Fig 12.4 A Successive Approximation ADC system

Set the signal generator *Signal* = **DC Level**, *Offse*t = **72%**. This should generate a voltage of 3.600V on the Vout socket. This is used as the input to the converter and can be measured by the voltmeter if it is connected into the circuit.

Make sure that all the digits in the word to be determined are 0. Digits are toggled by clicking in the digit box.

In Fig 12.4, it can be seen that the output of the digital word is connected to a DAC. The output of this DAC is compared with the signal to be measured. The objective is to set the output of the DAC to be the same as the voltage being measured. The digital input to the DAC is then the digital value of the input signal.

Select **Display|Graph**. Channel 4 (purple) is the voltage to be measured and this voltage can be seen in the scale area when ch4 is selected. Channel 2 (blue) is the output of the measurement DAC and again the voltage can be read in the scale area when ch2 is selected.

Each digit is tested in turn, starting with the most significant digit (msd) on the left.

Set the msd to 1 by clicking its box. Is the DAC output (blue) greater than the input (purple)? If it is, your guess at the value of the msd is wrong so set it back to 0. If the DAC output is less, your guess is correct so leave the bit set to 1. Enter the value obtained in the msd box in Table 12.3 of your workbook.

Now test each of the digits down the chain in the same way. Set the bit to 1 and, if the DAC output is greater than the voltage to be measured, reset the bit to 0 otherwise accept the value of 1. As you enter each binary digit, you will see the decimal of the measured value approach that of the input value. By testing each digit in turn, you have successively approximated the binary value to give a digital representation of the input analog value.

Offset %	Input Voltage	msd					lsd 80mV	Measured Voltage
72%	3.600	1	0	1	1	0	1	3.600
50%								
30%								

Table 12.3 Successive Approximation Values

Set the signal generator offset to the other values shown in Table 12.3 and find the digital values of these offsets.

In practice, the successive approximation register, DAC and comparator are all built onto a single chip. This type of ADC works reasonable quickly - in the order of microseconds. Each bit in the conversion required 1 clock pulse so increasing the resolution does not greatly add to the conversion time. The accuracy depends on the accuracy of the DAC and, like the stand alone DAC discussed above, 12 bit accuracy is now common place at relatively low cost.

Sample/Hold

With Successive Approximation, the digital value is being converted back to analog and compared with the actual signal being measured. There can be a problem if the actual signal changes during the conversion process. For this reason, a **Sample/Hold** circuit is always placed before the converter. Before conversion starts, this samples the value to be measured by charging a capacitor then holds the analog value steady on the capacitor during the conversion process.

Multiplexing

ADC's are not inexpensive products. Often it is too expensive to allocate a separate ADC to each variable that has to be measured. Inputs are brought to an array of switches. In turn, each input is connected to the ADC and the conversion made. Such an arrangement is called **Multiplexing** and the device containing the switches is called a **Multiplexer**.

Fig 12.5 shows the complete schematic of the Successive Approximation ADC used on the CLIO board.

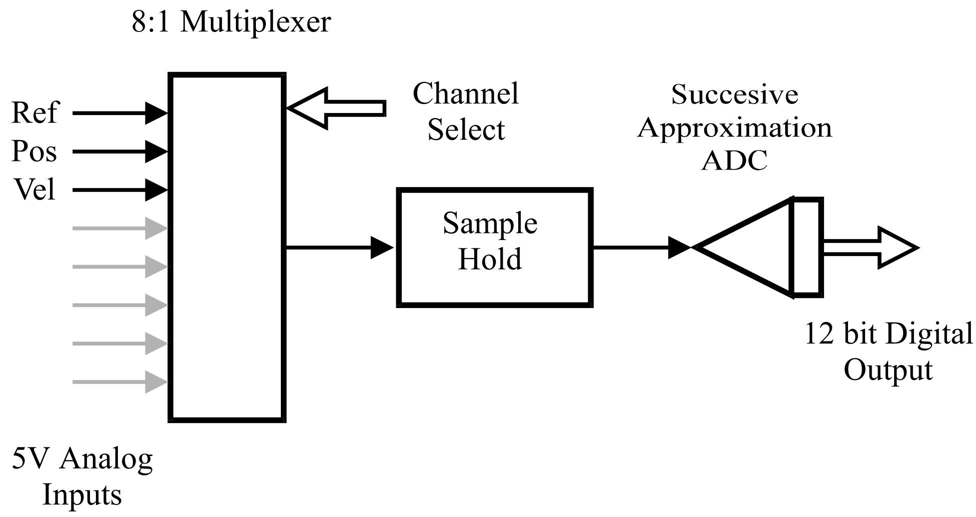


Fig 12.5 ADC System used in CLIO Interface

The CLIO ADC is an 8 channel 12-bit bipolar ADC converting signals in the range -5.12 to 5.1175 volts with a resolution of 2.5 mV. Only three of the channels are actually used.

Like DACs, an ADC is characterized by its range and resolution. In addition, conversion speed is also factor.

Other Conversion Techniques

There are a number of other ADC techniques in use.

Ramp Converter An up/down counter connected to a DAC is used. If the DAC output is less than the input, the counter counts up. If it is greater than the input, the counter counts down. This is an inexpensive way of providing individual ADCs per channel provided the input never changes fast.

Flash Converter This is the type of converter used for digitizing video signals. It is fast, expensive and low resolution.

Dual Slope Integrator This is the type of converter used in most digital voltmeters. It can be very accurate but is slow - 20 conversions per second. If the integration period is linked to the mains frequency, inaccuracies owing to mains interference can be eliminated. Large values of capacitance are required which are physically bulky so dual slope is not suited to inclusion on semiconductor chips.

Sigma Delta This converter is replacing dual slope in many applications. It can have the same noise rejection characteristics of the dual slope, can be highly accurate (up to 24-bits (1 part in 16,777,216 resolution)) and can be fully integrated onto a chip. Accuracy and speed can be traded under computer control making this an ideal technique for use in process industries where speed of conversion is not critical. With mains rejection, Sigma Delta converts at the same rate as dual slope (20 conversions per second).

12.4a **Sample/Hold is used with a Successive Approximation ADC to hold the input voltage steady during the conversion process:**

[Yes] or [No]

12.4b **A multiplexer allows a number of inputs to be converted by the same ADC:**

[Yes] or [No]

12.4c **The fastest type of ADC is:**

- [a] Successive Approximation.
- [b] Dual Slope.
- [c] Sigma Delta.
- [d] Flash.

12.4d **The most accurate type of ADC suitable for fabrication on an IC is:**

- [a] Successive Approximation.
- [b] Dual Slope.
- [c] Sigma Delta.
- [d] Flash.

12.5 Problems with ADCs and DACs

Whereas the analog signal is continuous in both time and amplitude, the digitized signal is discrete in both time and amplitude, such that the amplitude can only have values at multiples of the resolution and the value is only known at sample times.

Amplitude Quantization

Select **Controller | Quanta** and set the *Amplitude* to **5-bits 320mV** and *Time* to **1**. The ADC now has a range of ±5V and a resolution of 320mV. Set the signal generator to *Signal* = **Sine**, *Level* =**50%,** *Offset* = **0%** and *Rate* = **1mS (10mS for NT operating system)**. Channel 1 (dark blue) shows the sine wave being output (at full 12 bit resolution). Channel 4 (purple) shows what the computer thinks the input look likes as it samples with reduced resolution.

The input having only discrete values means that the controller can never control closer than 1 bit. This is called **Quantization Error**. At the moment this is set to 320mV as only 5 bits are being used.

Increase the number of bits by clicking the *Amplitude* box and see the quantization error decrease. At the 12-bit resolution of the CLIO interface, the resolution is 2.5mV which represents slightly less than 1/10th of a degree of the servo output shaft. This is adequate for a small servo but would not be sufficient for the position control of an astronomical telescope.

The resolution of the ADC and DAC system must be chosen to give an acceptable quantization error.

Time Quantization

The discrete nature of the time sampling also creates difficulties. On the display, the signal generator trace is being sampled every 1 degree, but the input trace (purple) can be being sampled slower than this. The number of degrees between samples can be set in the *Time* box. Set *Time* = **10**. The effect of the time quantization can be seen as the signal is flat between samples but the purple trace is at the correct (dark blue) value at the sample time. At this sampling rate, the input is still recognizable as a sine wave. As the sample interval increases, the sampled signal gets less and less like a sine wave.

The sample rate must be faster than the frequencies being measured.

Vary *Amplitude* and *Time* to see the measurement problems that arise when the sampling is too slow and the resolution is not good enough.

Aliasing

Set *Time* back to **1** and *Amplitude* to **12-bits**. Set *Rate to* **700mS (7secs for NT operating system)**. You are now sampling 700 times slower than previously. The sine wave is being sampled just slightly faster than once every two cycles. It will take a few minutes to build up the picture, especially when using NT.

The measured input now looks like a low frequency sine wave. This is not good. You are trying to measure a high frequency but, because of inappropriate sampling, the computer is seeing a much lower frequency. This is called **Aliasing** and occurs when a frequency signal is sampled less than twice per period. Instead of the correct frequency, the computer sees a low frequency alias. This can create a lot of problems in a control system if there is high frequency noise on the measurement. The controller will see the low frequency alias as an error and try to eliminate a signal which does not really exist.

The sampling rate of the ADC and DAC system should be chosen to be fast enough to respond to the fastest inputs or disturbances and the signals should be filtered to eliminate unwanted high frequencies which might cause aliases.

Student Assessment 12

1. A 10 bit binary number has a resolution of?

a 0.4%	b 0.1%	c 0.01%	d 0.001%

2. The ratio of resistor values required for a 10 bit R-2R DAC is:

a 2048:1	b 1024:1	c 512:1	d 2:1

3. Which of the following is not a feature of Successive Approximation ADCs?

a Is reasonably fast.

b Can be integrated on an IC.

c Performs with reasonable accuracy.

d Is extremely expensive.

4. A small steady state error in a digital position control system can be caused by:

a using too slow an ADC.

b amplitude quantization errors.

c time quantization errors.

d aliasing.

Chapter 13
Direct Digital Control

Objectives of this Chapter

Having completed this chapter you will be able to:

- Describe how numbers are handled within a computer
- Relate the problems associated with digital integration
- Explain how digital differentiation is performed
- Outline the problems arising from low resolution and slow sampling

Equipment Required for this Chapter

- MS15 DC Motor Module
- CLIO Interface Module with PC Connection Lead
- System Power 90 Power Supply (or equivalent)
- 4 mm Connection Leads
- PC running VCL Virtual Control Laboratory Software

13.1 Introduction to Direct Digital Control

This is not a course on computers and programming but, to understand direct digital control and some of the problems in implementing it, some aspects of computing are introduced in this chapter.

You will see how positive and negative numbers are handled, how they are added and subtracted to form the error signal, multiplied to form the gain and how integration and differentiation are performed to create a PID controller.

Small microprocessors or microcomputers are used in direct digital control. These do not normally have floating point processors so all computations are performed with fixed length arithmetic.

The problems associated with processor word length, conversion resolution and conversion timing will be addressed.

13.2 Number Representation

In the last chapter we saw that modern ADCs and DACs are available with resolutions of 8, 10 or 12 bits. Even with 8 bit conversions, working with an 8 bit word would cause rounding errors so 16 bit words will be assumed. However, for clarity, many of the examples will use shorter word lengths than this.

The signals to be dealt with can be both positive and negative so how are negative numbers represented in a computer? The easiest way would be to use the most significant digit as a sign bit but does this work when a positive and a negative number are added, e.g.

2	⇒	0010		
-2	⇒	1010		msd used as sign bit

+	⇒	1100	⇒ -4	Obviously this does not work.

2's Compliment Numbers

A technique called 2's complement is usually used to represent negative numbers. To negate a number, it is logically inverted then 1 is added to the inverted number.

2	$\Rightarrow$	0010		
Invert	$\Rightarrow$	1101		
1	$\Rightarrow$	0001		

+	$\Rightarrow$	1110	$\equiv$	-2 in 2's complement form
4	$\Rightarrow$	0100		
-2	$\Rightarrow$	1110		

+	$\Rightarrow$	0010	$\Rightarrow$ 2	

This gives the correct answer, the carry bit being ignored.

For a 4 bit word, the full range of numbers from -8 to +7 are shown in Table 13.1

2's Complement Binary	Decimal
0111	7
0110	6
0101	5
0100	4
0011	3
0010	2
0001	1
0000	0
1111	-1
1110	-2
1101	-3
1100	-4
1011	-5
1010	-6
1001	-7
1000	-8

Table 13.1 2's complement numbers (4-bit word)

The binary mode of the windows calculator can be used to practice 2's complement binary arithmetic but remember that the calculator works with a 32 bit word.

13.3 Range and Normalization

If the ADC and DAC are each 10 bits, which 10 bits of the 16 bit word should they occupy? The lower 10 bits would seem the obvious ones to use but doing this looses accuracy especially when integrating small differences - see below in the section on Integration.

Usually the numbers are normalized to be fractions of full scale, i.e. they occupy the top bits. The binary number (2’s complement form) 1000000000000000 would represent the most negative value (say representing -5.12 volts). The binary number 0111111111000000 would represent the most positive value of 5.10 volts (note that the resolution is 0.02V). So variable values are normalized to the interface range.

13.4 Arithmetic

The main arithmetic operations required are subtraction to calculate the error signal and multiplication to perform the gain. Addition and subtraction are straightforward operations in a microprocessor.

Multiplication to perform gain calculations present more of a difficulty. When two 16 bit numbers are multiplied, the result is a 32 bit number. Also, how is the gain represented within the computer? A gain of 5 might be represented by 0000000000000101 but how is a gain of 5.5 represented? If the gain can be set to 3 decimal places, gain calculations must be done in two parts. 5.500 is stored as 5500 (1010101111100). The variable is multiplied by this amount forming a 32 bit word. This 32 bit number is then divided by 1000 (1111101000) to form the 16 bit result. With a gain of 5.5, any error signal greater than 5.12/5.5 volts will cause the result to be bigger than full scale. The multiplication routine must ensure that results greater than the range are limited to the maximum or minimum of the range before the higher 16 bits of the 32 bit result are discarded.

13.4a **What is the binary result of the binary subtraction 01011100 - 00101101?**

a 00100101

b 00101010

c 00101111

d 11010000

13.4b **If the ADC and DAC in a system both have 12 bit resolution and the computer has a word length of 16 bits, which of the 16 bits would the conversion words occupy?**

- a top 12.
- b bottom 12.
- c middle 12.
- d start 1 down from the top.

13.4c **A proportional control system has a range from -2.048 to + 2.047 volts. The input reading is 1 volt and the output reading is 0 volts and the gain is 4. What is the drive voltage?**

- a 4 V
- b 2.047 V
- c -2.048 V
- d -4 V

13.5 Integration

In PID control there is a requirement to integrate the error. How is this done? With an analog signal, integration is the area under the curve as shown in Fig 13.1.

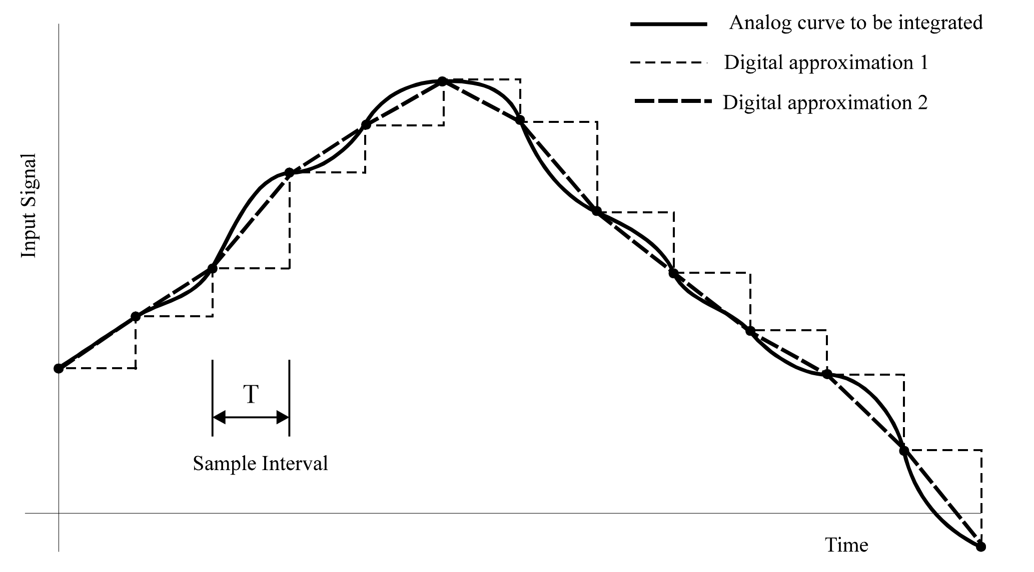


Fig 13.1 Analog integration and digital summation

There are a number of mathematical rules for calculating the area under the curve, the easiest being to keep a running total of all the incoming signal samples. However, the sampling interval must be taken into account, especially if it can be changed.

So, it would appear that the integral is:

$$I_n = (S_n \times T) + I_{n-1} \qquad \text{Eqn 13.1}$$

where S_n is the n^{th} sample, T is the interval between samples, I_n is the new integral value and I_{n-1} is the previous integral value.

Two things to be considered with this approach. From Fig 13.1, it can be seen that assuming the signal is constant during a sample interval does give an error. A more correct integration will result if it is assumed that the signal is a straight line between one sample and the next. The calculation would then be:

$$I_n = (S_{n-1} \times T) + \left(\frac{S_n - S_{n-1}}{2} \times T\right) + I_{n-1} \qquad \text{Eqn 13.2}$$

The first term is the rectangular area and the second is the triangular area. In a computer it is easy to remember the last sample so there is no difficulty in performing this calculation. This equation can be simplified to:

$$I_n = (S_{n-1} \times T) - \left(\frac{S_{n-1}}{2} \times T\right) + \left(\frac{S_n}{2} \times T\right) + I_{n-1} = \left(\frac{S_n + S_{n-1}}{2} \times T\right) + I_{n-1}$$

Eqn 13.3

The second problem with this calculation is not so obvious. Take the case where we are using a 16 bit word, the ADC has a 12 bit resolution (each bit representing 2.5 mV) and the sampling period is 1/64th of a second. A 1 bit error would then be stored as 0000000000010000. Multiplying this by 1/64 shifts the word right by 6 bits giving the result 0000000000000000. Nothing is integrated so the integrator cannot correct for a 2.5 mV error. It takes an error 4 times greater than this, i.e. 10 mV, before anything is added to the integrator total. The integral action would not then be doing its proper job of eliminating steady state error.

The solution to this is to store the sum of the samples and do the multiplication after the addition i.e.:

$$I_n = A_n \times T \quad \text{where} \quad A_n = \frac{S_n + S_{n-1}}{2} + A_{n-1} \qquad \text{Eqn 13.4}$$

Example

The following series of samples have been taken at 10 ms intervals:

0, 50, 100, 200, 250, 300, 300, 260, 220, 150, 75, 30

13.5a **The integral of these samples calculated using equation 13.1 is?**

13.5b **Using equation 13.3 is?**

13.5c **Using equation 13.4 is?**

Enter these values into your workbook. There will be a small difference between answers a and b. Answers b and c should be the same.

To illustrate the effects of word length, calculations can be made with integers only, i.e. the fractions of each calculation are dropped. This is the decimal equivalent to working with a fixed word length. To maintain the integer approach, multiplying by T = 0.01 can be regarded as dividing by 100.

13.5d **Using equation 13.3 working with integers only gives an integral value of?**

13.5e **Using equation 13.4 working with integers only gives an integral value of?**

Enter these values into your workbook. When working with restricted word lengths, the order of computation can be important. Equations 13.3 and 13.4 give the same answer when there is no word length restriction but equation 13.4 is much more accurate when the word length is restricted.

When writing computer programs for real time control using fixed word lengths, care must be taken with the word size and the order in which calculations are performed.

Another problem with the digital integral element in PID control occurs if the integral goes off-scale. The integral value must be held at the maximum positive or negative values and not allowed to 'wrap round'.

13.6 Differentiation

An approximation to differentiation is also easily performed but again care must be taken with the computation. Fig 13.2 shows a signal for which the digital differentiation is required. Differentiation is the rate of change of the signal so can be approximated by the difference between samples.

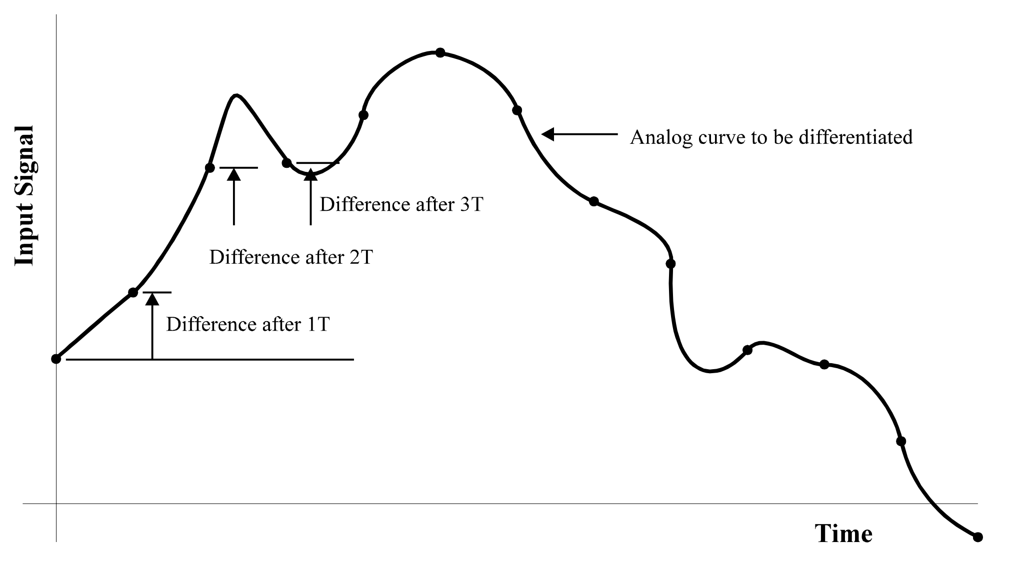


Fig 13.2 Digital differentiation

However, from Fig 13.2, it can be seen that the difference depends on the sample rate. The difference must be divided by the sample interval.

This gives the difference as:

$$D_n = \frac{S_n - S_{n-1}}{T}$$

Again there are problems with digital differentiation.

If the sampling interval is too short, there is only a small change between samples so only a few of the least significant digits are used. When corrected for the sample interval, there may be only a few values of slope produced by the differentiator resulting in crude control by the differential action.

If the sampling interval is too long, the difference values are greater but significant fast changes can be missed. The sampling interval should be chosen with regard to the system time constant. Around 10 to 50 samples during one time constant are required.

Care also has to be taken with the scaling of the division by the time interval T.

The derivative is a measure of rate of change so high frequency signals are amplified. To prevent this, the measured signals should be filtered to eliminate unwanted high frequencies.

13.6a **If the sample rate of a digital differentiator is too fast, the result will be:**

- [a] the difference between successive measurements will be too small to be significant.
- [b] significant change in signal may be missed.
- [c] large changes will result in out of range values.
- [d] only a few levels of slope produced.

13.6b **If the sample rate of a digital differentiator is too slow, the result will be:**

- a the difference between successive measurements will be too small to be significant.
- b significant change in signal may be missed.
- c large changes will result in out of range values.
- d only a few levels of slope produced.

13.7 Order of Calculation

You have probably been taught at some time that the order of calculation in arithmetic does not matter i.e.

$$(K \times A) + (K \times B) = K \times (A + B) = (A + B) \times K$$

However, when working with fixed word lengths, the order can matter. Fig 13.3 shows two versions of the block diagram of a PI controller. Under the rules of block diagrams and of normal arithmetic, both diagrams should give the same output. However, if the proportional gain is greater than 1, the output of the proportional element can easily reach the limits of the range which results in errors in the integral and differential calculations. If the gain is less that 1, resolution is lost when calculating the integral and derivative. So, with fixed word length, the lower configuration is usually used in digital implementation of PID.

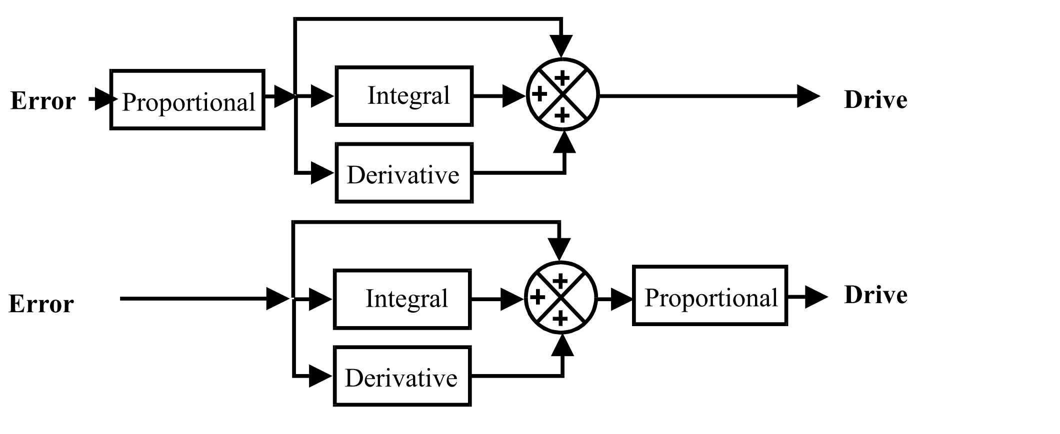


Fig 13.3 Different PID order of calculation

13.8 Effect of Sample Resolution

Provided the calculations are ordered properly, the main effect on performance accuracy is the resolution of the ADC and DAC.

Rewire the system as shown in Figure 1.6, chapter 1.

Start VCL and **load CA06PE13**.

File	**Controller**	**Plant**		**Display**	
CA06PE13	PID	MS15 Analog		Graph	
Signal Generator		**Graph**			
Signal	Step	1 Input	ON	5 Drive	Off
Level	50%			6 Proportional	Off
Offset	0%	3 Error	ON	7 Integral	Off
Rate	10 msec	4 Velocity	ON	8 Derivative	Off
Reference	Internal				
DC Motor		**Output Potentiometer**		Disengage	
Brake	0	**Command Potentiometer**		180°	

If the controller parameters set do not give a satisfactory response, go back to chapter 9 and use the settings you determined for your DC Motor.

You should have a step response with just no overshoot. Now change the bit setting from 12-bit to 5-bit and observe the changes to the response.

There will be two different effects:

1. The transient responses in little steps, each 1 least significant bit, i.e. 320 mV.

2. The steady state does not remain steady. It changes up and down by 1 bit (320 mV). The control system is trying to maintain a steady value between the two nearest levels it can measure so is oscillating between the two levels.

13.9 Effect of Sample Rate

All the calculations within the PC implementation of the controller use the floating point capability of the PC to give accurate integration and differentiation as the sample rate changes so that a wide range of plant can be controlled. However, the effects of sample rate can be seen by slowing the rate at which the ADC and DACs operate.

Set the resolution back to 12 bits. Now decrease the signal generator *Rate* which controls the rate at which the signal generator, display and the conversions are sampled.

Observe the system behavior at the rates shown in Table 13.2 and opposite each rate, in your workbook, note the change in stability of the system. Initially this will show as a change of overshoot.

Rate msecs	Stability
10	
20	
50	
100	
200	
500	

Table 13.2 Effect of sampling rate on stability

As the time between samples increases, the system damping decreases evident by increasing overshoot. Eventually the sampling rate is too slow to maintain control of the system and it begins to oscillate.

13.9a **Reduction of the system resolution will:**

a speed up the system response.

b slow down system response.

c limit the steady state accuracy to the least significant digit.

d improve the steady state accuracy.

13.9b **Reduction of the system sampling rate will:**

- [a] speed up the system response.
- [b] slow down system response.
- [c] make the system more stable.
- [d] make the system less stable.

13.10 Summary

Microcomputers with analog interfacing are now fast enough and inexpensive enough to be used as controllers without worrying too much about the effects of digitization except in very fast or very accurate situations.

The use of a microcomputer system gives a number of other advantages owing to the ease with which the processor can communicate with other processors. This allows the implementation of SCADA systems and effective tuning using more powerful processors.

Digital systems also allow more ergonomic displays and setup procedures.

Student Assessment 13

1. A proportional control system has a range from -2.048 to + 2.047 volts. The input reading is 1 volt, the output reading is 1.5 volts and the gain is 4. What is the drive voltage?

- [a] 2.047
- [b] 2
- [c] -2
- [d] -2.048

2. When working with a fixed word length, which equation would be used to calculate the integral of error?

- [a] eqn 13.1.
- [b] eqn 13.2.
- [c] eqn 13.3.
- [d] eqn 13.4.

3. An error signal is sampled at 50 ms intervals. Two successive readings are 1.500 and 1.800 volts. The slope of the signal in volts/second is:

- [a] 0.015
- [b] 0.133
- [c] 6
- [d] 66.67

4. The resolution of a DAC causes:

- [a] time quantization errors.
- [b] amplitude quantization errors.
- [c] aliasing.
- [d] no errors at all.

5. Too slow a sampling rate will cause:

- [a] slowing of the response
- [b] increased steady state error.
- [c] instability.
- [d] reduction of overshoot.

Chapter 14
Digital Interfacing

Objectives of this Chapter

Having completed this chapter you will be able to:

- Describe Pulse Width Modulation and other digital drive techniques
- Describe Gray code position measurement and other absolute and relative digital position measurement techniques
- Indicate the problems associated with digital speed measurements
- Compare the behavior of analog and digital control systems

Equipment Required for this Chapter

- MS15 DC Motor Module
- CLIO Interface Module with PC Connection Lead
- System Power 90 Power Supply (or equivalent)
- 4 mm Connection Leads
- PC running VCL Virtual Control Laboratory Software

14.1 Introduction to Digital Interfacing

When discussing digital control, we have looked at how digital controllers have been used as direct replacement for analog controllers. This involved using ADCs and DACs to convert between analog and digital forms.

As has been shown, the conversion process adds costs and can create problems. It would be easier if the computer could drive using logic signals rather than having to convert to analog form and if the measurements were also available in digital form.

The MS15 DC Motor system has a number of digital interfaces which will be used to illustrate digital interfacing

- **Pulse Width Modulation** (PWM) provides a digital drive to the motor. To select the PWM input, move the Motor Drive switch to the PWM position.

- The **Slotted Disc** can be used to measure the speed of the motor.

- The **Gray Code** coded disc measures the angular position of the motor shaft.

The system wiring using the digital interface is shown in Figure 14.1. The CLIO interface for the digital interface consists of a logic PWM output labeled Pw, a logic input for the slotted disc labeled P0 and 4 logic inputs for the Gray Code measurement labeled D0-D3.

Wire up the system as shown in Figure 14.1 but do not switch on yet.

IMPORTANT: PWM Calibration
If this is the first time that your PC has been used to control the DC Motor system using PWM, it is recommended that the PWM function in the VCL software is calibrated before continuing.

If you are not sure whether PWM calibration has already been carried out, please check with your instructor. Full instructions on how to do this are provided in Chapter 14, Section 14.3 of the CA06 Instructor's Manual.

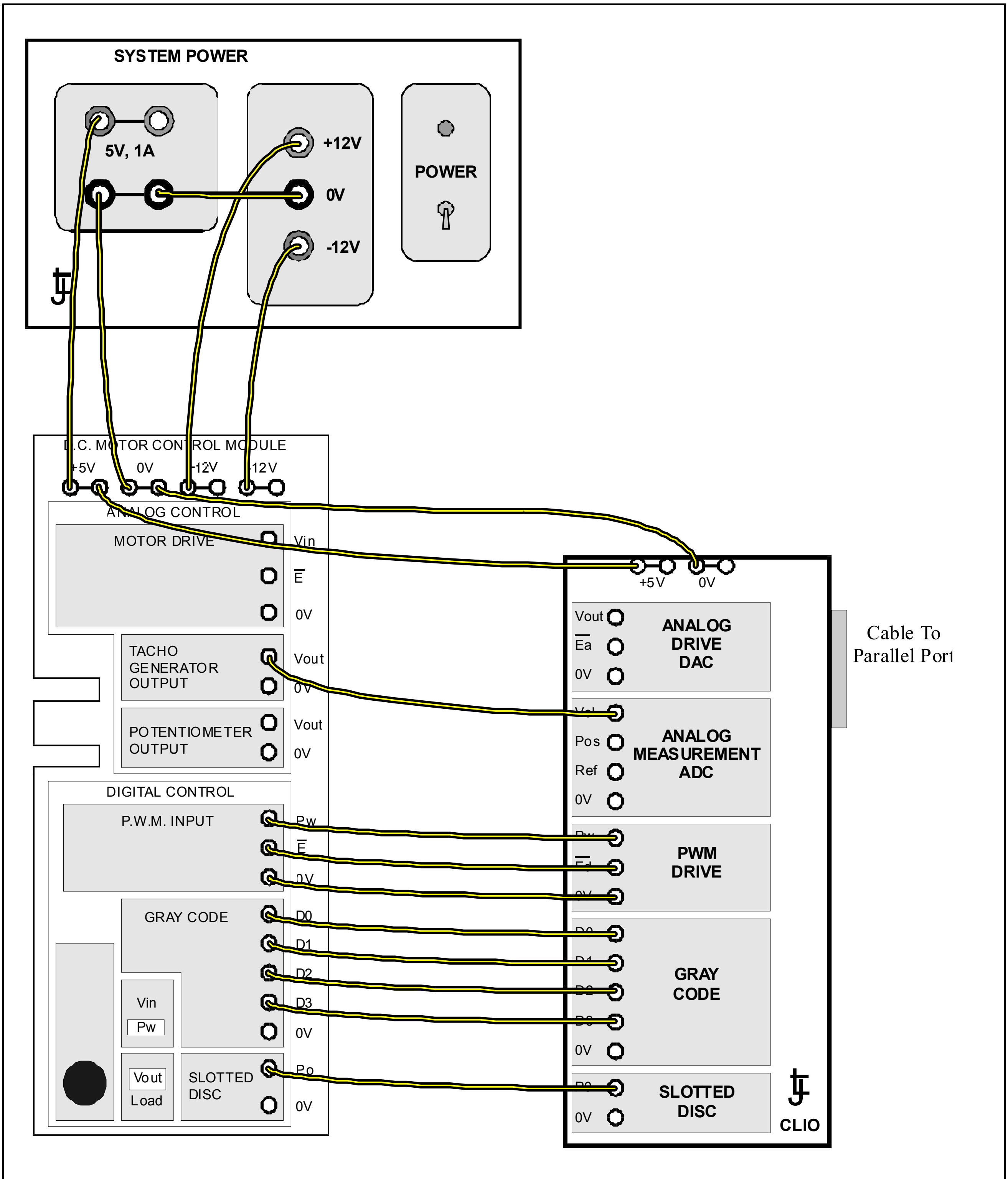


Fig 14.1 Wiring for digital interfacing

14.2 Digital Drive

In earlier chapters the drive to the plant had to be converted to an analog signal to power the DC Motor. As has been shown, digital to analog converters are quite complex devices. There are many instances where a control signal can be provided directly eliminating the need for the DAC.

Bang-bang

The simplest type of digital drive is called a **Bang-bang** system. Oven temperature and refrigerator temperatures are controlled in this way. The oven heater is ON until the required temperature is reached then is switched OFF. When the temperature drops, the heater is switched ON again.

Using thyristor switches, a low level digital signal can switch on and off vast amounts of electrical power with little power loss.

There is unlikely to be a computer in a domestic oven but many heating applications in industry do use microprocessor control of electric heating, especially when the temperature has to be controlled to a profile in time. Plastic molding machines require this type of temperature control.

Fig 14.2 shows the mains voltage required for various power levels in a temperature control system. The power is switched as required. If half power is required to maintain a temperature, power would be ON for half the time and OFF for half the time. The power can be controlled by a single digital line from a computer.

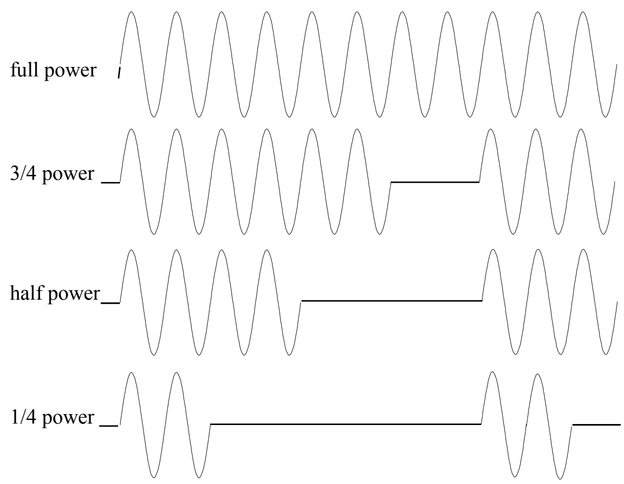


Fig 14.2 Thyristor ON/OFF control

Thyristors used in this way normally contain circuitry to ensure that power is always switched when the voltage is zero. This minimizes the electrical noise produced by the switching action.

In bang-bang control, the application decides the length of the ON period and the length of the OFF period. However, it was realized from this that the power to a plant could be controlled by varying the ON/OFF ratio of a signal. This has led to the use of Pulse Width Modulation.

Pulse Width Modulation - PWM

The average value of a periodic two level signal depends on its ON/OFF ratio as shown in Fig 14.3. When the pulse train is averaged using an analog filter or the filter action of the plant itself, the average value goes from full ON to OFF (or full ON in negative direction) as the ON time is reduced. For many processes, such as the DC Motor or an oven, such a pulse train will be averaged by the process itself so that the power drive can be an efficient logic power switch rather than an inefficient analog power amplifier.

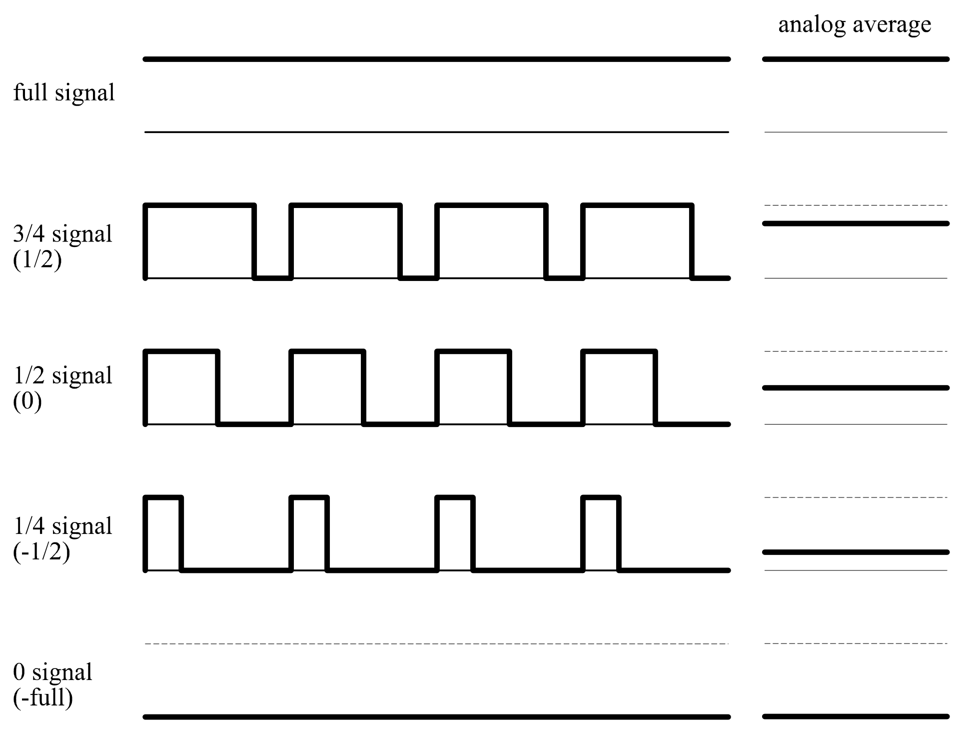


Fig 14.3 Pulse Width Modulation

This technique works well provided the processor can provide the timing accuracy. Many processors have special registers which will provide PWM outputs with accurate timing.

For the DC Motor plant being used, an update rate of 10 milliseconds (0.01 seconds) has been used. To obtain the equivalence of 12 bit resolution, the ON and OFF times have to be controlled to 0.01/4000 seconds, i.e. 2.5 millionths of a second 2.5μs). This is well within the capabilities of many modern processors.

PWM outputs are available on some microprocessors and integrated circuits are available to drive DC motors. These take a low power logic signal and convert it to a high power PWM drive to the motor coils.

Stepper Motors

A stepper motor is a variant of the DC motor designed in such a way that the armature is stable at a number of fixed angles. A voltage pulse on the drive coils moves the armature to the next stable position, i.e. it steps to the next position. The drive coils can be arranged so that movement can be in either direction. Stepper motors are normally designed to move by 7.5° or 15°.

A computer can easily drive a stepper motor - special power drive chips are available. One logic line specifies the direction of rotation and another is pulsed to rotate the rotor by one step. Since the power drive is a switch, this arrangement is power efficient.

Stepper motors are used when a mechanism has to be positioned without the use of a feedback system. Such a system requires that the starting position (zero datum) is known. Positioning the print head in a laser printer is a typical example. Many machine tools use stepper motors for positioning the tool and work-piece.

14.3 Modified PWM used on the MS15 DC Motor system

The digital drive to the DC Motor uses a modified form of PWM which makes it easier to meet the timing considerations. In this implementation, only the length of the ON signal is used. The PWM signal is shown in Fig 14.4.

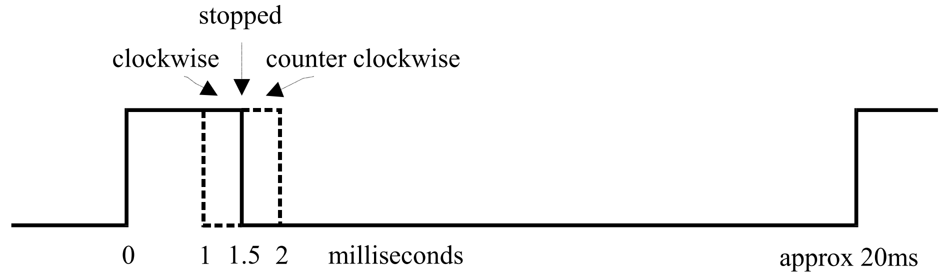


Fig 14.4 PWM used with MS15

A 1 ms pulse will cause the output shaft to rotate at full speed clockwise. A 2 ms pulse will cause full speed counter-clockwise rotation and a 1.5 ms pulse is the stopped condition. This pulse should be repeated at intervals.

The timing of the pulse width gives a resolution about half that of the DAC, i.e. the PWM resolution is approximately 1 in 2000.

Start the VCL software and **load CA06PE14**

File CA06PE14	**Controller** Open-Loop	**Plant** MS15 Digital	**Display** Event
Signal Generator		**Logic Display**	
Signal	DC Level	PWM	
Level	0%		
Offset	0%		
Rate	10 msec		
Reference	Internal		
DC Motor		**Output Potentiometer**	Disengage
Brake	0	**Command Potentiometer**	180°

Notice that the **Plant** is now *MS15-Digital* indicating that the digital sections of the DC Motor controls and the CLIO interface are being used.

The PWM signal appears on the Pw socket in the PWM Drive area of the CLIO module and should be connected to the Pw input on the MS15 module. To select PWM drive rather than analog drive, the Motor Drive selector switch at the bottom of the MS15 unit must be in the PWM position.

PWM Drive

The PWM mode of the event display should be selected. This shows the waveform being sent to the motor drive circuit. The width of the pulse (in milliseconds), and the speed of the motor (in volts) as measured by the tachometer, are shown as numerical values.

With the input *Offset* = **0%**, the display should show a 1.5 ms pulse and the motor should not be rotating.

Increase the *Offset* in **10%** steps and note that the motor rotates counter clockwise. At 100% Offset, the pulse width should be 2 ms and the motor is running at full speed. Enter the results in Table 14.1 of your workbook and plot the resultant graph in Fig 14.5.

Offset %	Pulse Interval	Tacho Output Voltage
0	1.5	
10		
20		
30		
40		
50		
60		
70		
80		
90		
100		

Table 14.1 Relationship between PWM drive input and velocity output

Compare this curve with the one obtained in chapter 2, Fig 2.1. As shown in Fig 14.5, they should be very similar. Pulse width modulation, requiring only a single logic output from the computer, can be used to generate the drive for the DC motor.

Disable the drive.

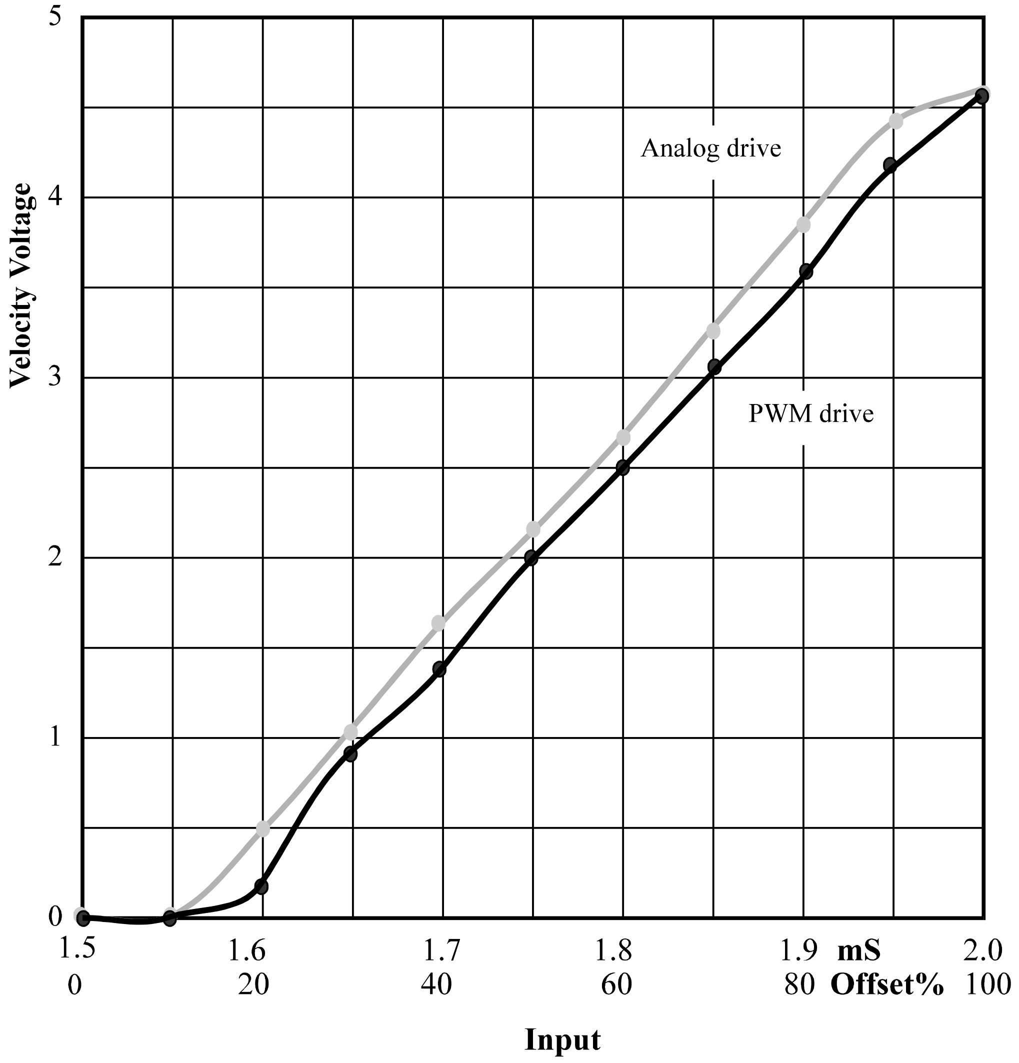


Fig 14.5 PWM Drive vs speed under no load conditions

14.3a **A Bang-bang system switches on power when the output is less than the set point and switches it off when the output is more than the set point.**

[Yes] or [No]

14.3b **Pulse Width Modulation controls power to drive the system by?**

[a] varying the amplitude of a square wave signal

[b] varying the pulse width of a repetitive pulse

[c] varying the frequency of a square wave

14.3c **Was the measured PWM drive curve close to that obtained for analog drive?**

[Yes] or [No]

14.4 Digital Position Measurement

Position measurements can either be absolute or relative. Within the boundaries of the system, an absolute measurement can tell, at any time, where it is within the system.

The analog potentiometer is an absolute measurement as it tells the angular position of the system.

A relative system can only tell you how far it has moved from the start of the measurement sequence. Unless told in absolute terms where it started, the relative system does not know where it is.

Many digital measurement systems are relative systems. The difference between absolute and relative is illustrated in Fig 14.6.

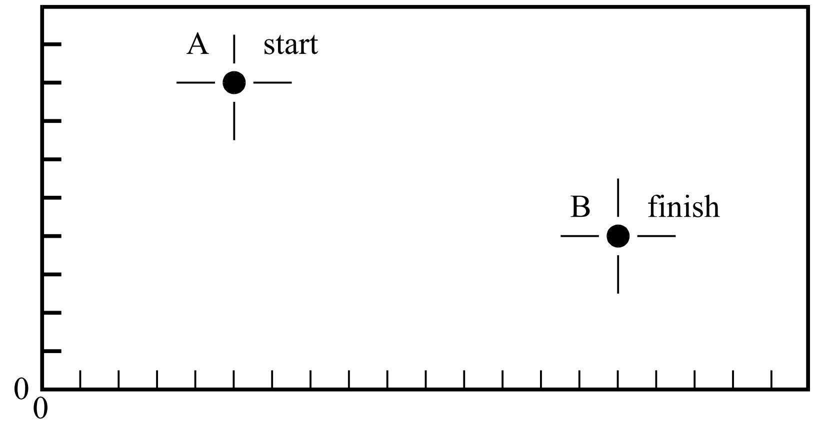


Fig 14.6 Absolute and relative measurement

On a worktable, a machine tool head starts at position A and moves to position B. Within the measurement space, the absolute position of A is x=5, y=8 and of B is x=15, y=4. A relative measurement starting at A and finishing at B would be x=10, y=-5. Unless the absolute position of A is known, the position of B is only known relative to the start position A.

Relative Position Measurements

Digitally, relative measurements are easier than absolute ones. On a machine tool, it is easy to move the tool head to a zero datum position then make all measurements relative to this datum position.

Fig 14.7 overleaf shows one way of measuring relative movement. A cog wheel is mounted on the shaft whose position has to be measured - this may be derived from a linear movement via a gearing system. Every time a tooth moves past the detector it creates a pulse which is counted. The total number of pulses is the distance moved. Two detectors are used to detect the direction of movement - if B is high when A goes high, the cog is rotating counter clockwise and if B is low when A goes high the cog is rotating clockwise.

A number of technologies can be used to detect cog movement. Interrupting a light beam is a useful technique but is susceptible to dirt. Inductive or capacitance proximity detectors can be used. Magnetic Hall effect sensors are also used but this requires the teeth to be a series of magnets.

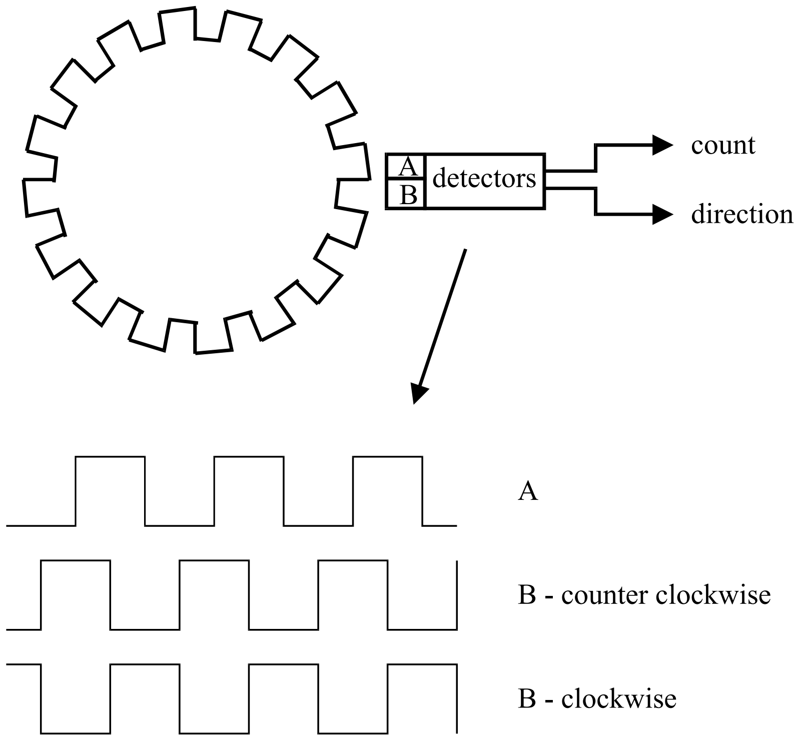


Fig 14.7 Measuring change of position using a cog wheel

This is a useful technique but there is a limit to the number of notches which can be detected on a cog. Gearing can overcome this to some extent but mechanical techniques like this do not give very accurate measurement.

Optical techniques are used for accurate position measurement in machine tool control and metrology. Fig 14.8 shows how this works.

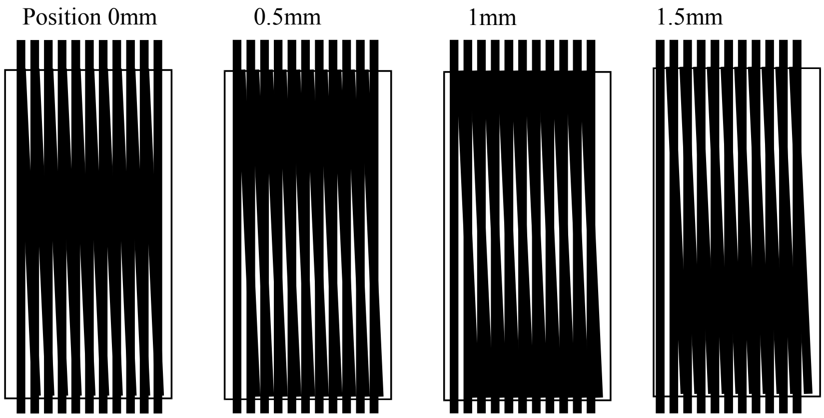


Fig 14.8 Moiré Fringe for position measurement

Two optical gratings are produced photographically. One has a series of vertical lines uniformly spaced. The other has the same set of lines but these are angled slightly. In Fig 14.8, the lines are 1 mm thick and 2 mm apart. The second set is angled so that the bottom of the line is 2 mm to the right of the top of the line. When the two gratings are placed on top of each other, a fringe pattern can be seen. This is called a Moiré Fringe.

One long grating is fixed to the bed of the machine and another short grating is fixed to the moving platform. As the platform moves, the black band moves up or down. One band will move past the detect head each time the platform moves by the pitch of the grating. The bands are counted to determine the distance moved.

Since very fine gratings can be produced, at a price, this technique is used for very accurate measurement. Small movements of the platform are magnified into large movements of the black bar. Machine tool inspection stations generally use this technique.

Absolute Position Measurements

Absolute position measurement is based on coded discs as shown in Fig 14.9. A digital pattern is screened onto a clear disc so that a digital pattern dependent on the angle of rotation can be picked up by the photo-detectors.

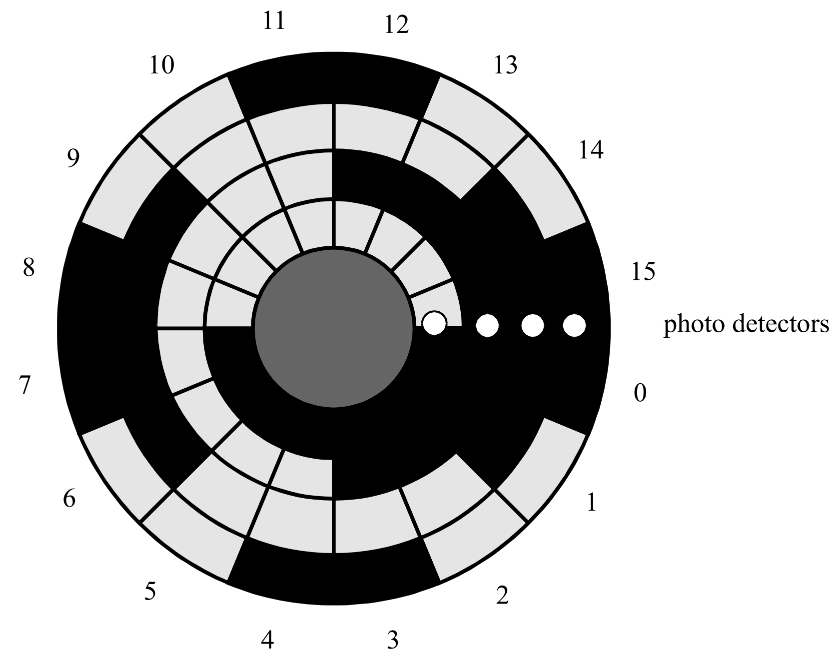


Fig 14.9 Gray coded disc

Looking at the disc in Fig 14.9, you will see that it is not a binary code that is used. The code used, called a Gray code, is given in Table 14.2. This is the disc used on the MS15 module.

Position	Angle	D3	D2	D1	D0	Change
0	0 → 22.5	0	0	0	0	D3
1	22.5 → 45	0	0	0	1	D0
2	45 → 67.5	0	0	1	1	D1
3	67.5 → 90	0	0	1	0	D0
4	90 → 112.5	0	1	1	0	D2
5	112.5 → 135	0	1	1	1	D0
6	135 → 155.5	0	1	0	1	D1
7	155.5 → 180	0	1	0	0	D0
8	180 → 202.5	1	1	0	0	D3
9	202.5 → 225	1	1	0	1	D0
10	225 → 245.5	1	1	1	1	D1
11	245.5 → 270	1	1	1	0	D0
12	270 → 295.5	1	0	1	0	D2
13	295.5 → 315	1	0	1	1	D0
14	315 → 337.5	1	0	0	1	D1
15	337.5 → 360	1	0	0	0	D0

Table 14.2 4-bit gray code

The main reason for using the Gray code is that, at every change of code, only one digit changes. This makes the alignment of the optical sensors much less critical. If a binary code was used, on the change from 7 to 8, all four digits change. If D3 were mis-aligned so that it changed before the others, the computer would detect a change of position from 7 to 15 then to 8. Use of the Gray code eliminates such glitches. Gray code is also used because there is a logical relationship between Gray code and binary code which makes program writing easier.

To detect the code using LED's, the tracks must be relatively wide. This makes it difficult to make high resolution discs especially if they have to rotate at high speed. Electronics or computer technology can be used to overcome some of these deficiencies but high resolution coded discs are expensive to produce.

14.5 Digital Position Measurement used on the MS15 DC Motor system

The MS15 module is fitted with a 4-bit Gray code disc mounted on the motor shaft.

Change the *Logic Display* from **PWM** to **Gray**. The display now shows the output of the Gray code disc.

With the output *Disabled*, rotate the output shaft until the pointer is somewhere between 0 and 40 degrees. Click in the *Sync* box and the *Degrees* will read the measured angle. The Gray code has been synchronized to the output shaft position. This is necessary since the Gray disc is mounted on the motor shaft which rotates 9 times for every rotation of the output shaft. The Gray disc can only measure within 360/9 = 40° segments of the output shaft. Synchronization tells the computer that the output shaft is in the zero sector. The computer then keeps track of which sector it is in by detecting the change from code 15 to code 0.

The 4-bit Gray code then resolves the output position to 40/16 = 2.5°. This is roughly equivalent to using the potentiometer connected to a 7 bit ADC.

Slowly rotate the output shaft counter clockwise and you will see the code output go through the sequence shown in Table 14.2. The display shows the logic waveforms being measured. Note that only one of the four digits changes at any one time.

Enable the motor and set the input *level* to **50%**. The Gray code is impossible to read by eye. Change to **Display | Graph**. The position trace (blue) now shows the position of the output shaft.

Disable the drive.

14.5a Is it true that a relative position measurement measures the change in position from an arbitrary starting position.

Yes or No

14.5b **Moiré Fringe measurement is used in inspection machines because:**

a it can detect very small changes in position.

b it is a relative position measurement technique.

c it is inexpensive.

14.5c **A Gray coded disc returns the binary pattern 0101. What range of angles of the shaft does this represent?**

a 0 to 40° b 0 to 22.5° c 135 to 155.5° d 270 to 295.5°

14.5d **If the disc is rotating clockwise, what will be the next code returned?**

14.6 Digital Speed Measurement

Speed is not easy to measure digitally in a control context. For the DC motor, it is a number representing the revolutions per minute that is required. The obvious way to obtain this is to count the number of times the shaft revolves in one minute. A slotted disc is provided for this purpose on the MS15.

Enable the motor and set the input *Offset* to **+100%**. This drives the motor at maximum speed. Read the speed shown on the red LED display. This is the speed of the output shaft in RPM.

Maximum output shaft speed = **RPM**

There is a 9:1 gearing between the motor and the output shaft so the speed of the motor shaft is 9 times this figure.

Maximum motor shaft speed = **RPM**

Disable the drive and make a note of these figures in your workbook.

Frequency Measurement

The motor should be rotating at around 2000 RPM. The resolution obtained from the Tacho + ADC system was 1 in 2000. With one pulse per revolution as provided, the pulses would have to be counted for one minute before the measurement could be used. But, to control speed or position, the measurements are required every 10 milliseconds.

We could put more slots on the disc.

2000RPM is the same as 2000/60 = 33.3 revolutions per second

Therefore one revolution takes 1/33.3 seconds = 30 milliseconds.

In 10 ms the shaft revolves by 1/3 of a revolution.

To measure 2000 counts in 10 milliseconds would require a wheel with 6000 slots.

Period Measurement

Rather than measure the number of pulses in a given time, the time between pulses could be measured. At maximum speed, only 3 slots are required for one to come past every 10 ms but at a speed of 1 RPM, 2000 x 3 = 6000 slots would be required for a slot to pass every 10 ms.

Also, since the period is the inverse of frequency, the period measurement requires to be inverted before being used. This requires a division operation within the processor which can take up time.

Practical Systems

Many applications do not require the rotational speed, the accuracy or the update rate of the servo system. In many applications, a multi-slot disc, possibly speeded up via gearing, can be used.

A 30 slot disc geared to run 10 times faster than the motor shaft would give a 1% resolution when sampled every 10 ms or 1 in 2000 resolution if sampled every 200 ms. Problems with this are that, at maximum speed, the disc would be rotating at 20000RPM. This is fast for a mechanical system. Also the inertia of the disc would be reflected to the motor by the gear ratio, i.e. 10 times, so would add significantly to the motor inertia.

As with the position measurement by slotted disc, the direction can be determined by using two sensors placed 1/4 of a cycle apart.

Slotted discs using optical, magnetic or other proximity detection techniques are used in many applications but not in those where fast sampling is required. ABS braking is one such example.

Measurement by Computer

Microprocessor timer inputs can be configured for either frequency or period measurement. Frequency is easier to implement as the results do not depend upon the clock frequency of the timer used in the period measurement. Also, frequency is proportional to speed whereas period is inversely proportional which necessitates an extra division operation.

For fast sampling of speed as required by the servomotor example, a tachometer with analog to digital conversion is the most suitable combination. Many microprocessors have built in ADCs but, if not, it is now easy to interface one to a microprocessor - see section 14.7 below.

14.6a **Digital speed measurement using a toothed wheel is not useful for:**

[a] systems requiring infrequent measurements.

[b] systems requiring accurate measurement in a short time.

[c] systems requiring rough measurements.

14.6b **The technique used on the test rig to obtain a digital measurement of speed is:**

[a] frequency measurement from toothed wheel sensors.

[b] period measurement from toothed wheel sensors.

[c] analog to digital conversion from the tachogenerator.

14.7 Other Digital Measurements

Since many control systems are now being implemented as direct digital controllers, both microprocessors and peripherals are being designed to make the task easier.

Microprocessor Features

The timers available on microprocessors can be configured as either counters for frequency measurement or as timers for frequency measurement. The processor only requires to set up the configuration and read the data at the correct time.

Many microprocessors now have PWM registers. Again the computer sets up the timing required and the hardware then produces the signal required without further processor intervention until the mark space ratio has to be changed.

A number of analog input channels (ADCs) are also provided on many microprocessors.

There are also special registers for interfacing with peripherals.

Interfacing

Initially peripherals such as ADCs and DACs were treated as part of the memory space. That meant that the peripherals had to be close to the processor. Now a number of different interfacing techniques are provided which allow the peripheral to be mounted at a distance from the processor. These interfacing techniques generally involve transmitting the digital data as a serial data stream.

There are three popular schemes: - CAN, SPI (or Microwire) and I^2C. CAN is used in harsh environments and is popular on robots and in cars. I^2C is more useful in computing peripherals such as real time clocks. SPI is the scheme most used with control type peripherals and most microprocessors now include an SPI interface.

Fig 14.10 shows a system, similar to the CLIO interface, using an SPI interface. SPI uses 3 lines common to all peripherals and a select line for each individual peripheral. When the select line is active (low), data is clocked into or out of the selected chip. There is no standard protocol, each device having its own requirements. This lack of formality makes SPI very flexible. In the CLIO interface used with the PC, it takes between 40 and 80 microseconds to send or receive data from a device. A dedicated microprocessor could drive the devices faster than this.

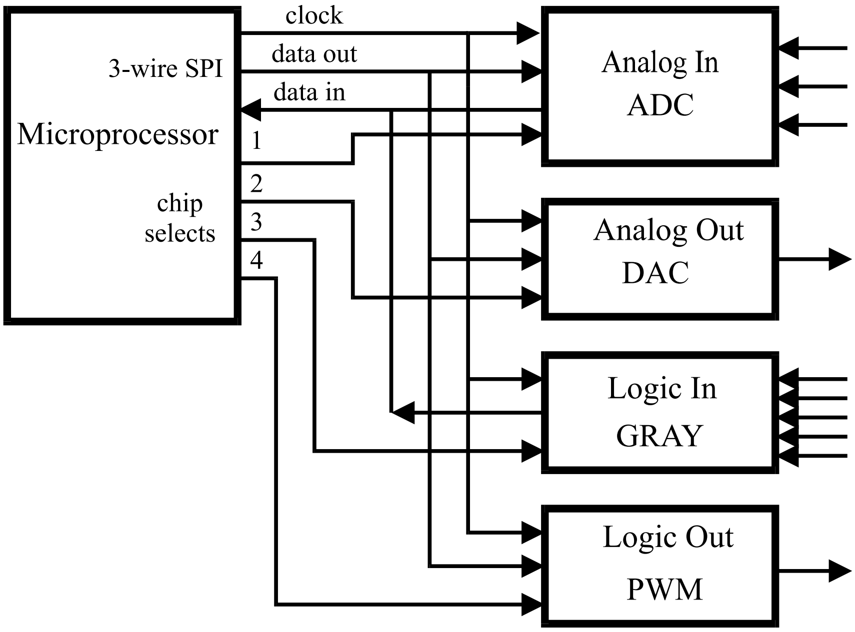


Fig 14.10 SPI or Microwire interface example

Sensors with built in conversion

With the increase in computing power of integrated circuits and their reduction in size and power requirements, measurement devices are now being produced with conditioning and conversion electronics built into the device. Digital temperature sensors are available from a number of suppliers.

14.8 Controller using Digital Measurement

Disable the drive, select **Display | Event | Gray**, turn the output dial to the 1st sector (0-40°) and Sync the Gray code.

Select **Display | Graph, Controller | Servo** and set *Signal* | **Step**, *Level* | **50**% and *Offset* | **0**%.

The system is now set as a position servo using digital position measurement, analog velocity measurement and PWM drive.

Enable the drive.

Set the proportional gain K_c to 5 and adjust the velocity feedback K_v to give a system with just a little overshoot. Enter the velocity feedback used into your workbook.

K_v for optimal response = ☐

Compare this with the value obtained in chapter 8 with analog measurement and drive. There should not be much difference between the two figures.

Change to **Display | Event | Gray**. This shows the Gray code of the position being measured. Change to **PWM** and the pulse width of the drive is shown.

Digital measurement and drive has not made any significant difference to the performance of the system although the lack of resolution in the position measurement is evident.

In some instances, like PWM, digital techniques offer an advantage, in others, analog sensors followed by analog to digital conversion give better performance. Most modern controllers are now microprocessor based using a combination of analog and digital interfacing techniques.

The PC as a digital controller

The PC does not have the special PWM, timer or Microwire registers that can be found on some microprocessors. As a result, the experiments using the digital peripherals may have given poorer performance than would have been obtained using a dedicated microprocessor. The PC performs better with the analog peripherals. The PC, however, does excel in providing the graphic display and easy to use operator interface.

In industry, the process control function and the plant operating function would normally be provided by separate computers, the fast closed loop control being provided by a microprocessor based system and the system display and control provided by a machine with the interactive graphic power of the PC.

14.8a **Compared with analog interfacing, did the system with digital interfacing perform:**

[a] better. [b] as good as.

[c] poorer but adequately. [d] poorer and not adequately.

14.8b **PWM digital output is used because:**

[a] it can be much more power efficient.

[b] it uses only 1 output pin on the microprocessor.

[c] it can be as accurate as analog drive.

[d] all of the above.

Student Assessment 14

1. A PWM system as used on the MS15 requires to be updated every 50 milliseconds. 10-bit resolution is required. What timing resolution is required from the microprocessor producing the signal?

   [a] 5 milliseconds [b] 500 microseconds

   [c] 50 microseconds [d] 5 microseconds

2. Gray coded discs are preferred to binary coded discs because:

   [a] they measure absolute position.

   [b] only one digit changes for each change in code.

   [c] they produce a code which can be used directly by the computer.

   [d] they can be fitted onto smaller discs.

3. The most common digital interfacing technique for the input and output of signals in real time control measurements is:

   [a] connection as memory. [b] CAN.

   [c] I^2C. [d] Microwire.

4. The 'real' world is predominately analog in nature. Are the control systems used now:

   [a] all analog. [b] predominately analog with some digital.

   [c] predominately digital with some analog. [d] all digital.