Session 2
# Op-amp, A/D-D/A converters and Compensator Emulation

Purpose

The objectives of this session are
- To learn the basic amplifier circuits for typical phase-lead and phase-lag compensators and design a typical analog phase-lead and phase-lag compensators.
- To investigate how signals are passed through the real-time control hardware/software system using D/A and A/D converters.
- To replace the analog compensators with digital compensators that emulates the physical compensators. Build the WinCon application, implement and test the compensators in the real-time

Introduction

1 Operational Amplifier
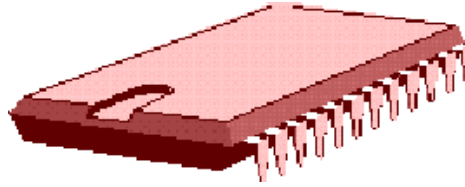
A typical control system includes a filter or compensator also known as *controller* whose role is to modify or shape the reference command input in ways that improve the system response. Analog controllers are commonly realized with passive circuit elements and operational amplifier.

The circuit symbol for an op-amp with its five essential terminals and its voltage transfer characteristic is shown in Figure 2.1.
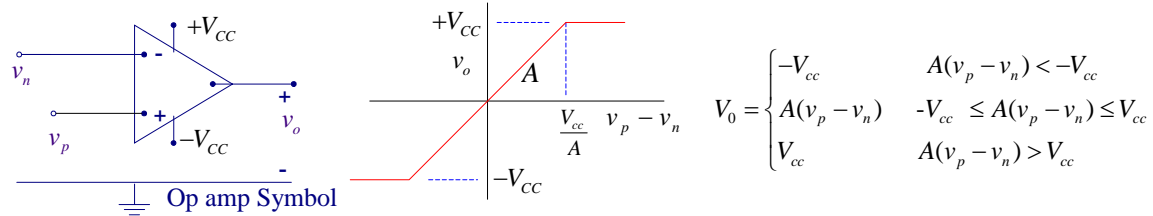

$$V_0 = \begin{cases} -V_{cc} & A(v_p - v_n) < -V_{cc} \\ A(v_p - v_n) & -V_{cc} \le A(v_p - v_n) \le V_{cc} \\ V_{cc} & A(v_p - v_n) > V_{cc} \end{cases}$$

Figure 2.1 Op-amp symbol and the input/output characteristic

When $|V_p - V_n|$ is small the op-amp behaves as a linear device. Outside this linear range the op-amp saturates and behaves as a nonlinear device. For operation in the linear range the net input voltage is constrained to the voltage V_{cc} applied to the power terminals. For a typical op-amp operating on its linear region, the voltage difference between the two inputs is very small, less than 2 mV, i.e., $v_x = v_p - v_n \simeq 0$. Also the input resistance is

very large, and the output resistance is negligibly small. For example, the common 741 op-amp has $A \simeq 10^5$, $R_{in} \simeq 2 \times 10^6 \Omega$, and $R_o \simeq 40\Omega$.

Because of the high input resistance, negligibly small current flows into the two op-amp inputs, usually on the order of μA. In most practical applications, the circuit analysis can be simplified with good accuracy by assuming the op-amp to be ideal, i.e.,

- $A = \infty$, so that $v_n = v_p$
- $R_{in} = \infty$, so that $i_n = i_p = 0$
- $R_o = 0$

1.1 Phase-lead and phase-lag compensators

The implementation of a transfer function is most commonly done using inverting op-amp circuit. Since the op-amp is a very high gain device, feedback must be added to the amplifier in order to stabilize it. The feedback circuit is connected from the output terminal to the inverting input terminal. This connection results in negative feedback. An example of a practical op-amp circuit in s-domain is shown in Figure 2.2.
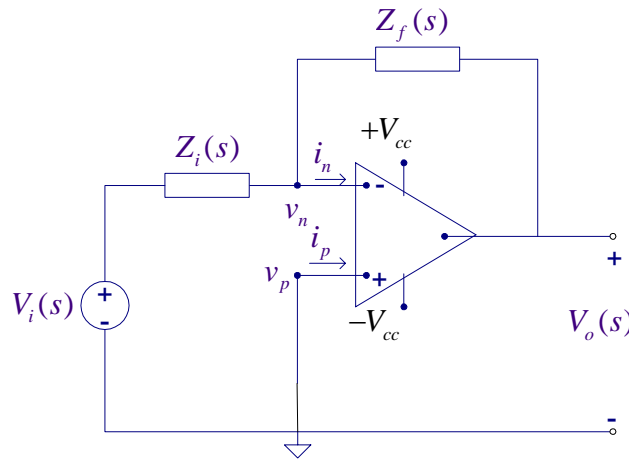

Figure 2.2 General inverting operational Amplifier

Since $i_n \simeq i_p \simeq 0$ and $v_n \simeq v_p = 0$, KCL will result in

$$\frac{V_i(s)}{Z_i(s)} + \frac{V_o(s)}{Z_f(s)} = 0$$

or

$$\frac{V_o(s)}{V_i(s)} = -\frac{Z_f(s)}{Z_i(s)} \qquad (2.1)$$

By proper choice of impedances we can realize phase-lead compensator (High-pass filter), phase–lag compensator (low-pass filter), PI, PD, or PID compensators that will be used to improve the performance of the control system. The transfer function in (2.1) contains a negative sign. To obtain a positive signal we can add another inverting op-amp with two identical input and feedback resistors, i.e., $R_{2f} = R_{2i}$

Consider the circuit of Figure 2.3 with impedances consisting of parallel RC circuits.
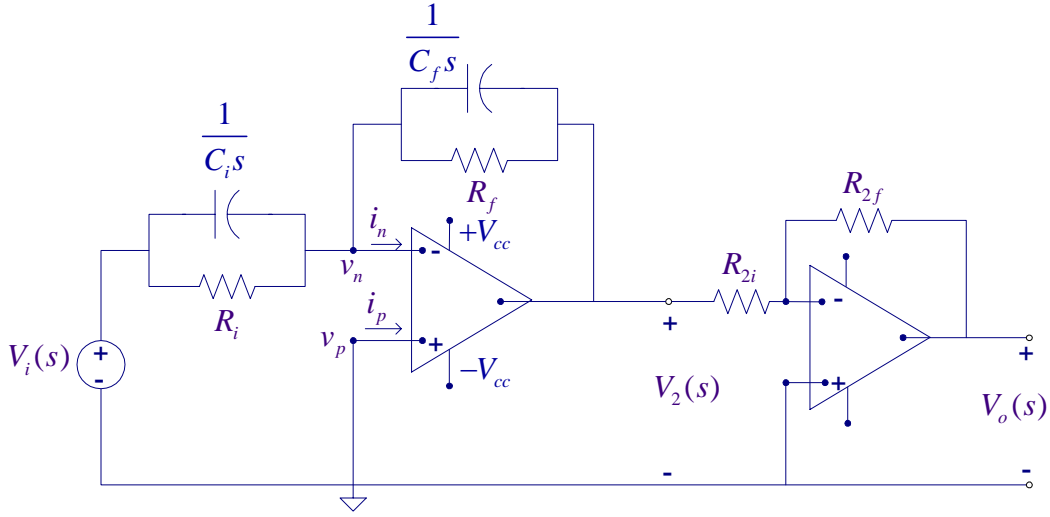
Figure 2.3 Basic compensator circuit

The parallel branch impedances are

$$Z_i(s) = \frac{R_i / C_i s}{R_i + 1/C_i s} = \frac{R_i}{R_i C_i s + 1} \text{ and } Z_f(s) = \frac{R_f / C_f s}{R_f + 1/C_f s} = \frac{R_f}{R_f C_f s + 1}$$

Therefore,

$$\frac{V_2(s)}{V_i(s)} = -\frac{Z_f(s)}{Z_i(s)} = -\frac{R_f(R_i C_i s + 1)}{R_i(R_f C_f s + 1)}$$

If another inverting op-amp with equal forward and feedback resistors are used the signal would be inverted and the compensator transfer function becomes

$$G_c(s) = \frac{V_o(s)}{V_i(s)} = \frac{C_i}{C_f} \frac{(s + 1/R_i C_i)}{(s + 1/R_f C_f)} = \frac{K_c(s + a)}{(s + b)} \tag{2.2}$$

Where

$$K_c = \frac{C_i}{C_f}, \quad a = \frac{1}{R_i C_i}, \quad \text{and } b = \frac{1}{R_f C_f} \tag{2.3}$$

For a realistic compensator a and b are greater than zero and the pole and zero are located in the left half s-plane as shown in Figure 2.4.
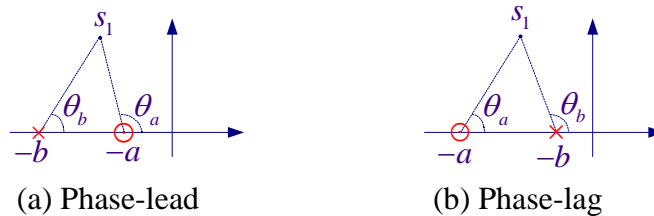

(a) Phase-lead   (b) Phase-lag
Figure 2.4 Compensator phase angle contribution

For a given $s_1 = \sigma_1 + j\omega_1$, the transfer function angle given by $\theta_c = \theta_a - \theta_b$ is positive if $a < b$ as shown in Figure 2.4 (a), and the compensator is known as the *phase-lead*

controller. On the other hand if $a > b$ as shown in Figure 2.4 (b), the compensator angle $\theta_c = \theta_a - \theta_b$ is negative, and the compensator is known as the *phase-lag controller*

The relations in (2.3) can be used to realize the phase-lead or phase-lag compensators, and since there are three equations and four unknowns, a suitable value for one of the parameters say R_i is chosen and the other elements are calculated from (2.3).

If C_f is removed $Z_f(s) = R_f$, and then we have

$$\frac{V_2(s)}{V_i(s)} = -\frac{Z_f(s)}{Z_i(s)} = -\frac{R_f(R_iC_is+1)}{R_i} = -\left(\frac{R_f}{R_i} + R_fC_is\right)$$

The compensator transfer function known as *Proportional Derivative* or *PD controller* becomes

$$G_c(s) = \frac{V_o(s)}{V_i(s)} = \frac{R_f}{R_i} + R_fC_is \tag{2.4}$$

or

$$G_c(s) = K_P + K_Ds \tag{2.5}$$

where

$$K_P = \frac{R_f}{R_i}, \text{ and } K_D = R_fC_i \tag{2.6}$$

If R_f is removed $Z_f(s) = 1/C_fs$, and then we have

$$\frac{V_2(s)}{V_i(s)} = -\frac{Z_f(s)}{Z_i(s)} = -\frac{R_iC_is+1}{R_iC_fs} = -\left(\frac{C_i}{C_f} + \frac{1/R_iC_f}{s}\right)$$

The compensator transfer function known as *Proportional Integral* or *PI controller* becomes

$$G_c(s) = \frac{V_o(s)}{V_i(s)} = \frac{C_i}{C_f} + \frac{1/R_iC_f}{s} \tag{2.7}$$

or

$$G_c(s) = K_P + \frac{K_I}{s} \tag{2.8}$$

where

$$K_P = \frac{C_i}{C_f}, \text{ and } K_I = \frac{1}{R_iC_f} \tag{2.9}$$

1.3 Phase-lead and phase lag Frequency Response

The phase-lead compensator transfer function is

$$G_c(s) = \frac{K_c(s+a)}{(s+b)}, \text{ where } a < b$$

The frequency response transfer function is

$$G_c(j\omega) = \frac{K_c(j\omega + a)}{(j\omega + b)} = \frac{K_c a}{b}\left(\frac{1 + j\omega/\omega_o}{1 + j\omega/\omega_p}\right)$$

Where $\omega_o = a$, and $\omega_p = b$ are the two corner frequencies.

Assuming a dc gain of unity for the compensator, (i.e., $\dfrac{K_c a}{b} = 1$) the transfer function is

$$G_c(j\omega) = \frac{1 + j\omega/\omega_o}{1 + j\omega/\omega_p} \text{ , with } \omega_0 < \omega_p$$

The Bode plot is as shown in Figure 2.5. It is apparent that the phase-lead compensator is a high-pass filter.
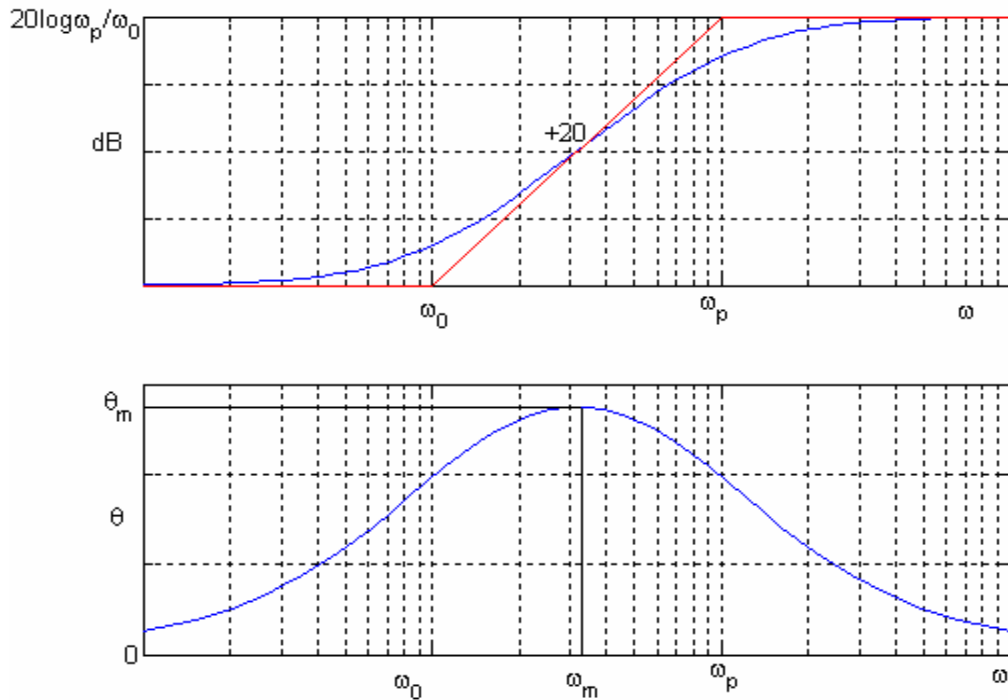

Figure 2.5 Bode diagram for phase-lead compensator

For the phase-lag compensator $\omega_0 > \omega_p$ and the compensator is a low-pass filter as shown in Figure 2.6.

The maximum positive angle in the case of the phase-lead compensator or maximum negative angle in the case of the phase-lag compensator occurs at geometric mean values of ω_o and ω_p or the frequency at maximum phase lead or phase lag is given by

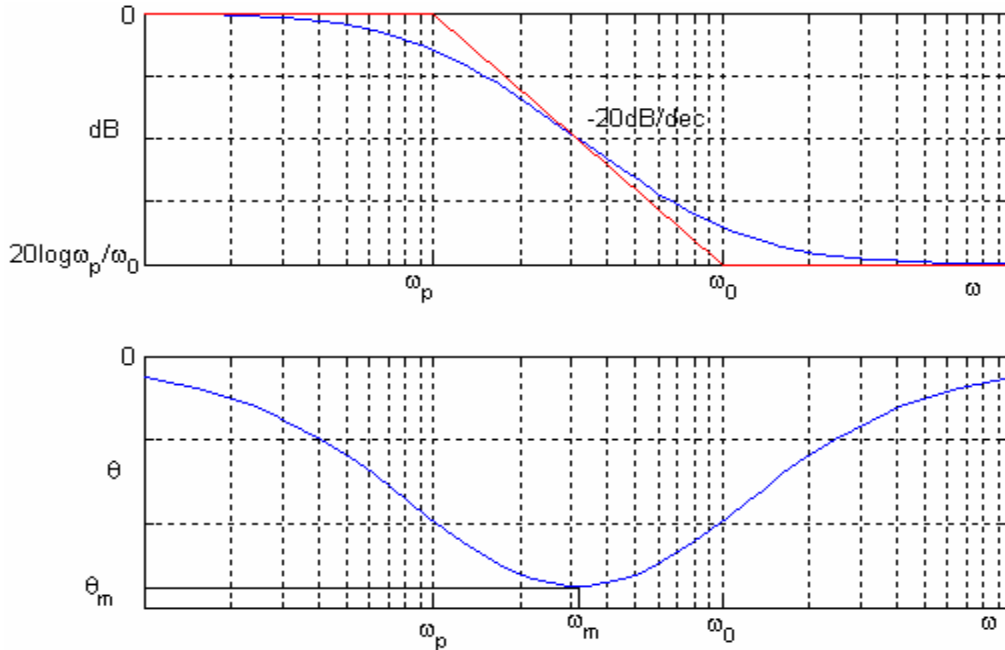$$\omega_m = \sqrt{\omega_o \omega_p} \tag{2.10}$$

2.5

Figure 2.6 Bode diagram for phase-lag compensator

2 Pre-laboratory Assignment
2.1 Phase-lead compensator realization

(a) Design an op-amp circuit to realize a phase lead compensator with the following transfer function.

$$G_c(s) = \frac{8(s+20)}{(s+160)}$$

For practical consideration, use resistors in the range of $1 K\Omega$ to $10\ M\Omega$ and capacitors less that $10\ \mu F$. You may use a $2\ \mu F$ capacitor for C_i in (2.3).

(b) Find the frequency response transfer function $G_c(j\omega)$ and evaluate the magnitude and phase angle of $G_c(j\omega)$ at the frequency of $f = 10$ Hz.

(c) Use the **bode** function in MATLAB to obtain the frequency response.

2.2 Phase-lag compensator realization

Repeat (a)-(c) for a phase-lag compensator with the following transfer function

$$G_c(s) = \frac{0.5(s+16)}{(s+2)}$$

2.3 PID Compensator
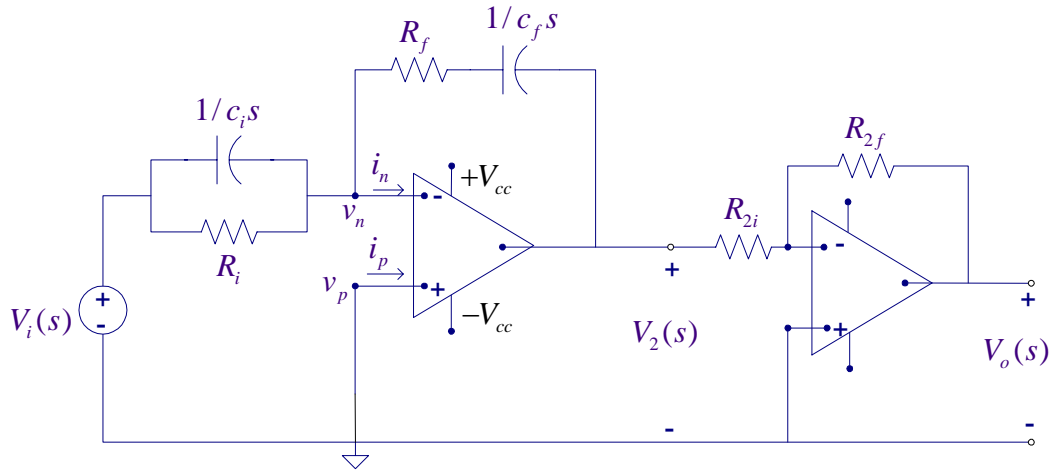
Figure 2.7 shows an arrangement for the PID compensator.

Figure 2.7 A circuit for the PID compensator

Show that the compensator transfer function is given by

$$G_c(s) = K_P + \frac{K_I}{s} + K_D s \tag{2.11}$$

where

$$K_P = \frac{R_f}{R_i} + \frac{C_i}{C_f}$$

$$K_I = \frac{1}{R_i C_f} \tag{2.12}$$

$$K_D = R_f C_i$$

2.3 Digital Emulation of compensators

In many modern industrial and commercial control systems analog compensators are replaced with a digital computer that performs calculations that emulate the physical compensators. A digital computer in the loop with sample-and-hold units can replace numerous analog compensators with a subsequent reduction in cost. Any changes or modifications that are required in the future can be implemented with simple software changes rather than expensive hardware modifications.

In the real-time control system the input signal is converted from analog form to digital form by means of the analog-to-digital converter. After passing the input through compensator developed in Simulink, the C code is generated automatically by Matlab RTW, which is then converted to analog form by the digital-to-analog converter. In this lab the phase-lead and phase-lag compensators introduced in section 2 will be emulated in Simulink and the resulting analog signal will be observed and measured on the oscilloscope.

3 Laboratory Procedure
3.1 Observation of a signal through A/D and D/A converters

Attach a T connector to the 33120A function generator, connect a BNC cable to one end of the T connector and connect the other end of the cable via a BNC to RCA adapter to Channel 0 of the analog input. Connect a BNC cable from the other end of the T connector to Channel 2 of the oscilloscope. Connect a BNC cable with a BNC to RCA adapter from analog Output # 0 to Channel 1 of the oscilloscope as shown in Figure 2.7.
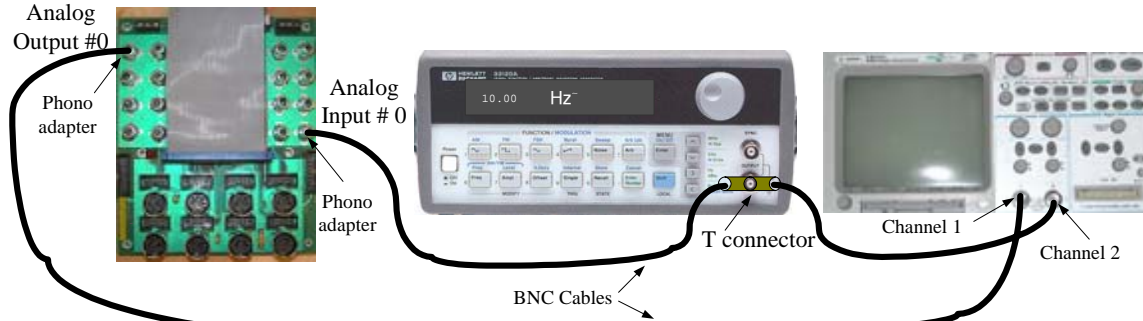

Figure 2.7 Wiring diagram

Turn on the function generator and adjust for a sine wave of 10 Hz with 2 V peak-to-peaks. If the function generator is not set to the high impedance mode, a reading of 1 V on the function generator represents an output of 2 V peak-to-peaks. Turn on the oscilloscope and reset it by the sequence **Save/Recall**, **Default Set-Up** (softkey), **Autoscale**. Set the horizontal channel to 10 msec/div and both vertical channels to 500 mV/div, and set the Center Adjust to zero mV on both channels.

Start a Simulink model. From Quanser Toolbox/Quanser Consulting MQ3 Series get a Quanser analog input block and set the Channel to Use to 0. Get a Quanser analog output and set the Channel to use to 0. Connect the two blocks together and attach a digital Scope as shown in Figure 2.8.
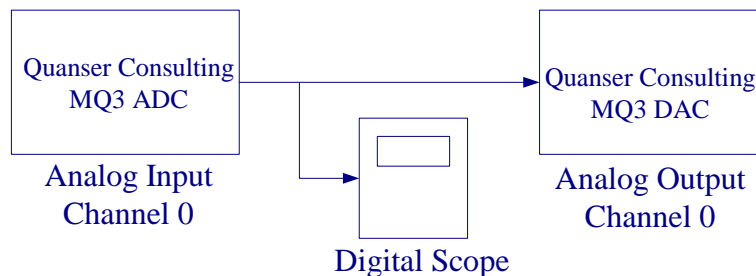

Figure 2.8 Implementation model

Save the model (say Lab2A.mdl). Set the Simulation parameters (select the Solver tab and set the sampling rate on the compensator to 0.001 seconds under "Fixed Step Size". Select ode4 Rung Kutta for integration method). In the Simulation drop down menu set the model to External. Start the WinCon **Server** on your laptop and then use **Client**

Connect, in the dialog box type the proper Client workstation IP address. Click WinCon/Build, this will generate the code and download it to the client. Click on the **Start** button on the WinCon Server to run the model. Click on Plot/New/Scope in WinCon Server and in the Select variable to display dialog box, click on Digital Scope (The name you assigned to the Simulink Scope) to select the variable you want to plot and press OK. This opens a real-time plot and displays the signal as read by A/D. Since the signal frequency is 10 Hz (T=0.1 sec), in the digital Scope menu click **Update/Buffer** and for the Buffer size type 0.2 (second) to display two cycles. You also should have the recovered analog signal on the oscilloscope. Press 'Single' on the Agilent Scope and capture a single trace of data.

You should have the Agilent IntuiLink software for the 54600 oscilloscopes, and the USB port NI-488.2 driver installed on your laptop. The IntuiLink can be found by opening Microsoft Excel, and then looking to see if the "Agilent 54600" toolbar appears just below the top row of Excel pull-down menus. If it does not appear, then you need to install the software from a CD-ROM (available at the MSOE EECS TSC, room S-350).

Capturing the Oscilloscope plot and plotting data
Connect the GPIB for the oscilloscope to the USB port on your laptop. Start Excel and press the Get Waveform Data icon to capture the data. Delete the first row containing texts. Highlight the data (no texts) in the first three columns to the last data row and copy the data to the clipboard. In MATLAB open a new M-File and paste the data copied to the clipboard. Save the data with a file name having extension **dat.** (e.g., **VinVout.dat**). Set the current directory to where the file VinVout.dat is stored and type the following statements at the MATLB prompt to obtain a MATLAB Figure plot.

>> load VinVout.dat
>> plot(VinVout(**:**, 1), VinVout(**:**, 2), 'r', VinVout(**:**, 1), VinVout(**:**, 3), 'b'), grid
>> legend('Recovered analog signal', 'Original analog signal')

Click on the **STOP** button to stop running the model.

3.2 Phase-lead Compensator emulation and implementation
Save your implementation model (Lab2A.mdl) under a new name say (Lab2B.mdl). Get a Transfer Fcn block from the SIMULINK Continuous library, and insert it in the new model as shown in Figure 2.9. Note that the signal goes through the A/D converter via the compensator to the D/A converter. Double click on the Transfer Fcn block to open its dialog box, enter [8 160] and [1 160] for the numerator and denominator respectively.
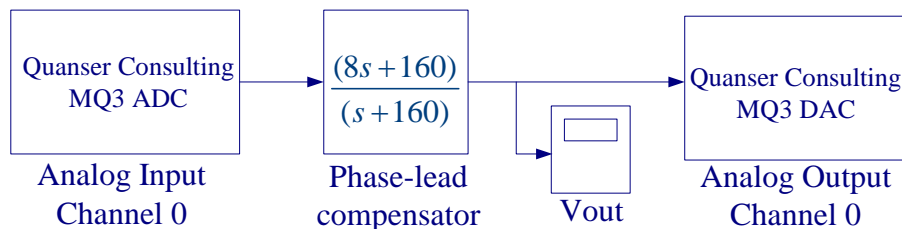
Quanser Consulting MQ3 ADC	$\dfrac{(8s+160)}{(s+160)}$	Vout	Quanser Consulting MQ3 DAC
Analog Input Channel 0	Phase-lead compensator		Analog Output Channel 0

Figure 2.9 Simulink implementation diagram for phase lead compensator.

Turn on the function generator and adjust for sine wave of 10 Hz with 2 V peak-to-peaks. In the WinCon server click **File/New** and use **Client Connect**, in the dialog box type the proper Client workstation IP address. Generate the real-time code corresponding to your diagram by selecting the "**Build**" option of the WinCon menu from the Simulink window. Click on the **Start** button on the WinCon Server to run the model. From the WinCon Server Plot/New/Scope select Vout to display the signal as read by A/D. In **Update/Buffer** set the Buffer size type 0.2 (second) to display two cycles.

Set the horizontal channel to 20 msec/div and the vertical channels to suitable mV/div, set the Center Adjust to zero mV on both channels. You should now see the output signal on the oscilloscope. Capture the data in Excel, copy the data (no text) to the clipboard paste in a new page in MATLAB editor and save as a **.dat** file (say PhaseLead.dat). Set the current directory to where the file PhaseLead.dat is stored and type the following statements at the MATLB prompt to obtain a MATLAB Figure plot.

>> load PhaseLead.dat
>> plot(PhaseLead(:,1), PhaseLead(:, 2), 'r', PhaseLead (:, 1), PhaseLead(:, 3), 'b'), grid
>> legend('Output signal', 'Input signal')

Change the signal frequency from 1 Hz to about 50 Hz and just observe how the output amplitude and phase angle changes and draw a conclusion regarding the filter characteristics. Also note that as the frequency increases the recovered signal become less accurate why?

Click on the **STOP** button to stop running the model.

Find the peak-to-peak values of the two waveforms and determine the voltage gain. Also use the Cursor to measure the compensator phase shift θ_c.

3.3 Phase-lag Compensator emulation and implementation

Save your implementation model (Lab2B.mdl) under a new name say (Lab2C.mdl). Set the transfer function parameters to the given values as shown in Figure 2.10.
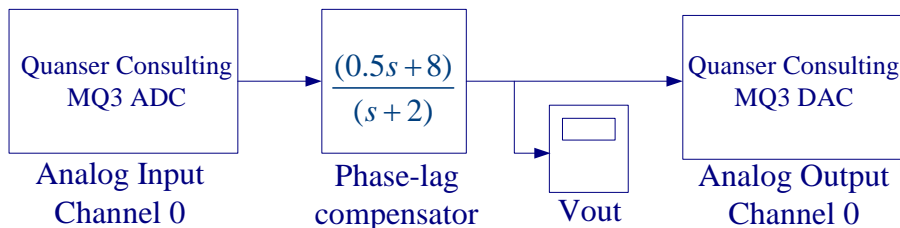

Quanser Consulting MQ3 ADC	$\dfrac{(0.5s+8)}{(s+2)}$		Quanser Consulting MQ3 DAC
Analog Input Channel 0	Phase-lag compensator	Vout	Analog Output Channel 0

Figure 2.10 Simulink implementation diagram for phase-lag compensator.

Turn on the function generator and adjust for sine wave of 10 Hz with 2 V peak-to-peaks.

In the WinCon server click **File/New** and use **Client Connect**, in the dialog box type the proper Client workstation IP address. Build and run the model. You should now see the output signal on the oscilloscope.

Capture the data in Excel, copy the data (no text) to the clipboard paste in a new page in MATLAB editor and save as a **.dat** file (say PhaseLag.dat). Set the current directory to where the file PhaseLag.dat is stored and type the following statements at the MATLB prompt to obtain a MATLAB Figure plot.

>> load PhaseLag.dat
>> plot(PhaseLag(**:**, 1), PhaseLag(**:**, 2), 'r', PhaseLag (**:**, 1), PhaseLag(**:**, 3), 'b'), grid
>> legend('Output signal', 'Input signal')

Change the signal frequency from 1 Hz to about 50 Hz and just observe how the output amplitude and phase angle changes and draw a conclusion regarding the filter characteristics.

Click on the **STOP** button to stop running the model.

Find the peak-to-peak values of the two waveforms and determine the voltage gain. Also use the Curser to measure the compensator phase shift θ_c.

4. Report

- Based on your data in part 3.2 estimate the update rate of the D/A in seconds. Refer to the analog-to-digital Converter (ADC) and digital-to-analog Converter (DAC) discussion under the Introduction section of the first laboratory session.

- Summarize the parameters of the analog phase-lead and phase-lag compensators realization of your pre-laboratory work.

- Compare the measured gain and phase values of the two digital compensators with the theoretical values found at 10 Hz.

- State the observation made in part 3.2 and 3.3 regarding the frequency response characteristics of each filter.

- State why the recovered signal accuracy is depleted as the signal frequency is increased.

- State advantages and disadvantages of digital compensators compared to analog compensators.