Date: ____________________ Name: ____

LABORATORY EXERCISE 1
PROCESS DYNAMIC CHARACTERISTICS

OBJECTIVE: To become familiar with various forms of process dynamic characteristics, and to learn a method of constructing a simple process model from step test data. **Optional:** To become familiar with obtaining data from frequency response tests.

PREREQUISITE: Completion of PC-ControLAB tutorial (under **Help | Tutorial**) or an equivalent amount of familiarity with the program operation.

BACKGROUND: All process have both steady state and dynamic characteristics. From a process control standpoint, the most important characteristic is the process gain. That is, how much does the process variable (PV) change for a change in controller output. If both the PV and the controller output are expressed as normalized variables (i.e., 0 - 100%), then the process gain is a dimensionless number.

The two most important dynamic characteristics of a process are the amount of dead time in the process and its time constant. Real processes rarely exhibit a response of a pure first order lag (time constant) and dead time, but can often be approximated as a first order lag and dead time.

This exercise tests for the process gain, dead time and time constant for both a "pure" process (can be exactly represented as first order lag plus dead time) and for a more realistic process which can only be approximated as a first order lag plus dead time.

1. RUNNING THE PROGRAM

Start **Windows**.

Run **PC-ControLAB.**

2. FIRST ORDER LAG PLUS DEAD TIME PROCESS

Click on **Process | Select Model**.

Highlight "Folpdt.mdl" (First Order Lag Plus Dead Time) and press **Open**.

Press **Zoom** and change the PV scale range to 50-75. (Note that the PV scale has already been converted to 0 – 100% of measurement span.)

With the controller in MANUAL, press **Out.** Note the initial values:

Present PV: ________

Present Controller Output: ________

Key in a new output value of 45.0.

(Hint: After the numerical value is keyed in, wait until a vertical green line on the grid is just crossing the grid boundary before pressing ENTER. This will make it easier to estimate subsequent times.)

After the PV has stabilized at a new value, press **PAUSE**.

Final value of PV: ________

What type of process response does this appear to be? ________

How much did the PV change? ________

How much did you change the controller output: ________

Process Gain: $K_p = \dfrac{\Delta\, PV}{\Delta\, Output}$ ________

How long after the controller output change before the PV started changing?

Dead Time (T_d) ________

Calculate 63.2% of PV change: ________

Actual value of PV at 63.2% of change: ________

How long after the PV started changing (i.e, at the end of the dead time) before the PV crossed the 63.2% point?

Time Constant (τ) ________

Suppose we had used the 2/3 point, rather than 63.2%, to estimate the time constant.

2/3 of PV change: ________

Actual value of PV at 2/3 of change: ________

How long after the end of the dead time before the PV crossed the 2/3 point?

Time Constant (approx) (τ_{est}) ________

What would be the percent error if we used the 2/3 point to estimate the time constant, rather than the true 63.2%?

Percent error: $\dfrac{\tau_{est} - \tau}{\tau} \times 100$ ________

Select **Process | Change Parameters**. Observe the values listed for Dead Time and Time Constant. Do these parameter values agree with what you observed?

____

Select **Process | Initialize.**

Select **Process | Change Parameters.**

Select “Dead Time” and change its value to 2.0 (minutes).

Select “Process Gain” and change its value to 1.0.

Select “Time Constant” and change its value to 3.0 (minutes).

Press **CLEAR**.

Change the controller output from 35.0 to 45.0. Observe the response. Is this what you would expect? ________

You have just observed the response of a pure first order lag and dead time process. Very few, if any, processes are this “clean.” We will now look at a process with unknown dynamics, but we will attempt to approximate it with a first order lag plus dead time model.

3. UNKNOWN PROCESS

Click on **Process | Select Model**. Highlight “Generic” and press **Open.**

Notice that the PV scale is now in Engineering Units, rather than in percent.
(If not, then select **View | Display Range | Engineering Units**)

Upper end of scale (corresponds to high end of transmitter range) ________

Low end of scale (corresponds to low end of transmitter range) ________

Span of PV ________

Initial value of PV (in engineering units) ________

Initial value of PV (in percent of span) ________

Controller Output: ________

Change the controller output from 35.0 to 45.0.

When the PV reaches (apparent) equilibrium, press **Pause**. Does this look like a true first order lag plus dead time process? ________

Does it look “approximately” like a first order lag plus dead time process? ________

What is the final value of the PV (to the nearest whole number)? ________

How much did the PV change, in engineering units? ________

How much did the PV change, in percent of span? ________

Estimate the process gain. K_p ________

To estimate the dead time, draw (or visualize) a tangent to the PV curve, drawn at the point of steepest rise. From the time of controller output change to the intersection of this tangent with the initial steady state value is the apparent dead time.

Apparent dead time: T_d ________

Different observers might estimate anywhere between 1½ to 2 minutes. For the purpose of calculating controller tuning parameters, it is better to take the longer value where there is any uncertainty, since that will produce more conservative controller tuning.

The apparent time constant is the time from the end of the dead time to 63.2% of the process rise (or approximately the time from the end of the dead time to 2/3 of the process rise.)

Apparent time constant: τ ________

Rather than estimating the time to 2/3 rise point, you can calculate the value of the PV at 63.2% of the ultimate PV change, then use the scroll bar to determine the time precisely between the (estimated) end of the dead time and the 63.2% rise point. This gives a more accurate estimate of the time constant, but takes longer.

NOTE: One of the uses that can be made of the estimates of process gain, dead time and time constant is to calculate controller tuning parameters. (See Laboratory Exercise 9, PID Tuning from Open Loop Tests.) Since dead time is more difficult to control than a first order lag, then if you estimate dead time too short, you are estimating that the process is easier to control than it really is. This will result in controller tuning parameters that cause the loop to be overly aggressive. Similarly, if you estimate the time constant too long, you are estimating that the process is easier to control than it really is, and again the resulting controller tuning parameters will cause the loop to be overly aggressive. On the premise that is one is to make an error, it is better to err in the conservative direction than in the aggressive direction, then the following pragmatic guidelines can be given:

If there is any uncertainty in estimating the process parameters,

estimate the dead time on the long side, and
estimate the time constant on the short side.

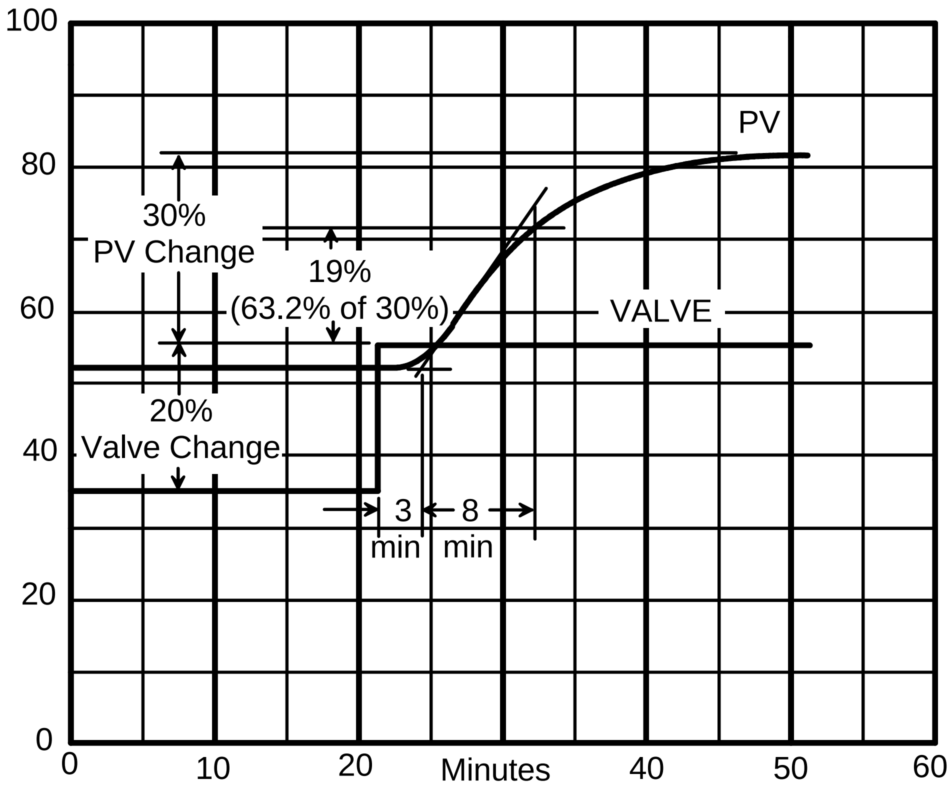


Figure 1
EXAMPLE OF ESTIMATING PROCESS MODEL PARAMETERS

4.0 OTHER FORMS OF STEP RESPONSE

This section will explore other forms of step response.

4.1 Negative Process Gain

Select **Process | Select Model.** Highlight “generic3.mdl" and press **Open.**

Record the initial values:

PV: ___________**DegF** __________**%**

Controller output: __________**%**

Increase the controller output by 10%. When the PV reaches equilibrium, record the following:

Final value of PV: __________**DegF** __________**%**

Final value of controller output: __________**%**

Process Gain:

$$K_P = \frac{\Delta PV(\%)}{\Delta \text{ Controller Output}(\%)} = \frac{PV_{final}(\%) - PV_{initial}(\%)}{\text{Cont Output}_{final}(\%) - \text{Cont Output}_{initial}(\%)} = \_\_\_\_\_\_\_\_$$

4.2 Integrating Process

*An **integrating process** is one which will not achieve a natural equilibrium. The control of liquid level, where either the vessel inflow or vessel outflow is fixed by an external set point, is a common example of an integrating process.*

Select **Process | Select Model**. Highlight "level2.mdl" and press **Open**.

This process model simulates a level control application in which the tank outflow is an independent process disturbance (load), and the controller controls a *valve in the inflow line.*

This process model has some elements of "realism" that are not needed in this exercise; These features will be used in subsequent laboratory exercises. For the present, we will eliminate these "realism" elements.

Select **Process | Change Parameters**.

Highlight the parameter labeled "Valve Pos: 0 = No; 1 =Yes." Change this parameter value to I.0.

Select the parameter labeled "Levl Noise: 0=No; I =Yes". (Use the scroll bar at the left hand side of the parameter list, if this parameter is not visible initially.) Change this parameter value to 0.0.

Press **Clear.**

Change the controller output from 35.0 to 38.0. This simulates an increase in inflow to the tank.

What does the PV (tank level) do? ______________________________

Does it appear that the PV is going to reach an equilibrium? ________

Before the PV reaches a limit (0 or I 00), press the button **StepIncr** above the controller faceplate. This simulates an increase in outflow from the tank. Note the increase in the trace labeled 'LOAD'.

If the PV is now falling, then the rate of outflow must exceed the inflow. Slightly increase the controller output (to increase the tank inflow) to attempt to stabilize the level. If the PV is rising, decrease the controller output. Can you eventually (approximately) stabilize the level by manual control? ________

4.3 Inverse Process Response

Occasionally the process response to a step change in controller output, or to a *load change, is initially in the opposite direction to that expected from "first principles." This is normally due to some underlying, second order* effect. *Once the second order effect disappears, then the process responds in the expected manner. A response such as this is called an "inverse response."*

An example of inverse response is the "shrink-and-swell" effect of steam boiler drum level. If the steam draw-off and steam rates are in equilibrium, then the drum level will be constant. On an increase in steam draw without a corresponding increase in feedwater rate, a decrease in drum level would be expected. Initially, however, due to the reduction in pressure and consequent flashing of water into steam, the level rises; if the excess steam draw is maintained, the level eventually begins to drop. This is qualitatively demonstrated in this exercise.

Select **Process | Select Model** and recall the Level2 process.

Select **Process | Change Parameters**.

Select the parameter labeled "Levl Noise: 0=No: 1=Yes". Change this parameter value to 0.0.

Select the parameter labeled 'BLK 57 LEAD TIME". Change the value of this parameter from 1.5 to -3.0.

Press **Clear.**

Press **StepIncr**. This simulates an increase in steam flow from the boiler drum.

Observe the PV (drum level). What does it initially do? ________________

(This is the drum level "swell" effect. If the steam flow had have been decreased, rather than increased, we would have seen the opposite effect, or the "shrink".)

What does the level eventually do? ________________

Suppose a feedback level controller had been in Automatic and controlling the feedwater rate (drum input). Upon sensing the initial change In level, would the level controller Increase or decrease the feedwater rate?

Would this change be in the proper direction for long-term correction?

Laboratory Exercise 23, Drum Level Control, will demonstrate a control technique widely used in steam generation applications for overcoming this problem.

5.0 FREQUENCY RESPONSE (Optional)

One means of characterizing a process is by its response to a sinusoidal (sine wave) input at various frequencies. At each frequency, the relevant data is:

the ***ratio of the amplitudes*** of the *input and output sine waves;*
the ***phase shift*** *between input and output signals.*

After a number of data points are taken, a **Bode plot** *of the data can be constructed.*

This portion of this laboratory exercise will determine a few data points for a Bode plot.

Through the **Process | Select Model** menu, read in “Generic1” process model. Be sure you are using the FEEDBACK control strategy (read the right hand side of the display title bar).

Check to see that the controller is in Manual.

Select **Control | Control Options**.

Scroll down until you see “Enable Sinusoidal Output.” Click on "YES."

Set the following parameter values:

Amplitude, Peak-to-Peak:	10
Period, minutes:	60

Press **Clear** then press **On** on the controller to initiate sinusoidal testing of the process.

Record the following:

Period, controller output and measurement:	60 minutes
Frequency, controller output and measurement,	0.0167 cycles/min
Controller Output Amplitude, Peak-to-Peak:	10%
Process Variable Amplitude, Peak-to-Peak (in % of full scale)	__________
Amplitude Ratio, PV/Controller Output:	__________
Time lag between controller output and PV (minutes):	__________
Phase shift, degrees:	

$$\text{Phase Shift} = \frac{\text{Time lag, minutes}}{\text{Period, minutes}} \times 360^0 \qquad \_\_\_\_\_\_\_\_\_\_$$

Repeat this test for periods of 30, 15, 7.5 and 3.75 minutes. (You may have to increase the amplitude of the input signal at the shorter periods. This is alright, since it is only the ratio of the output and input amplitudes that you are seeking.)

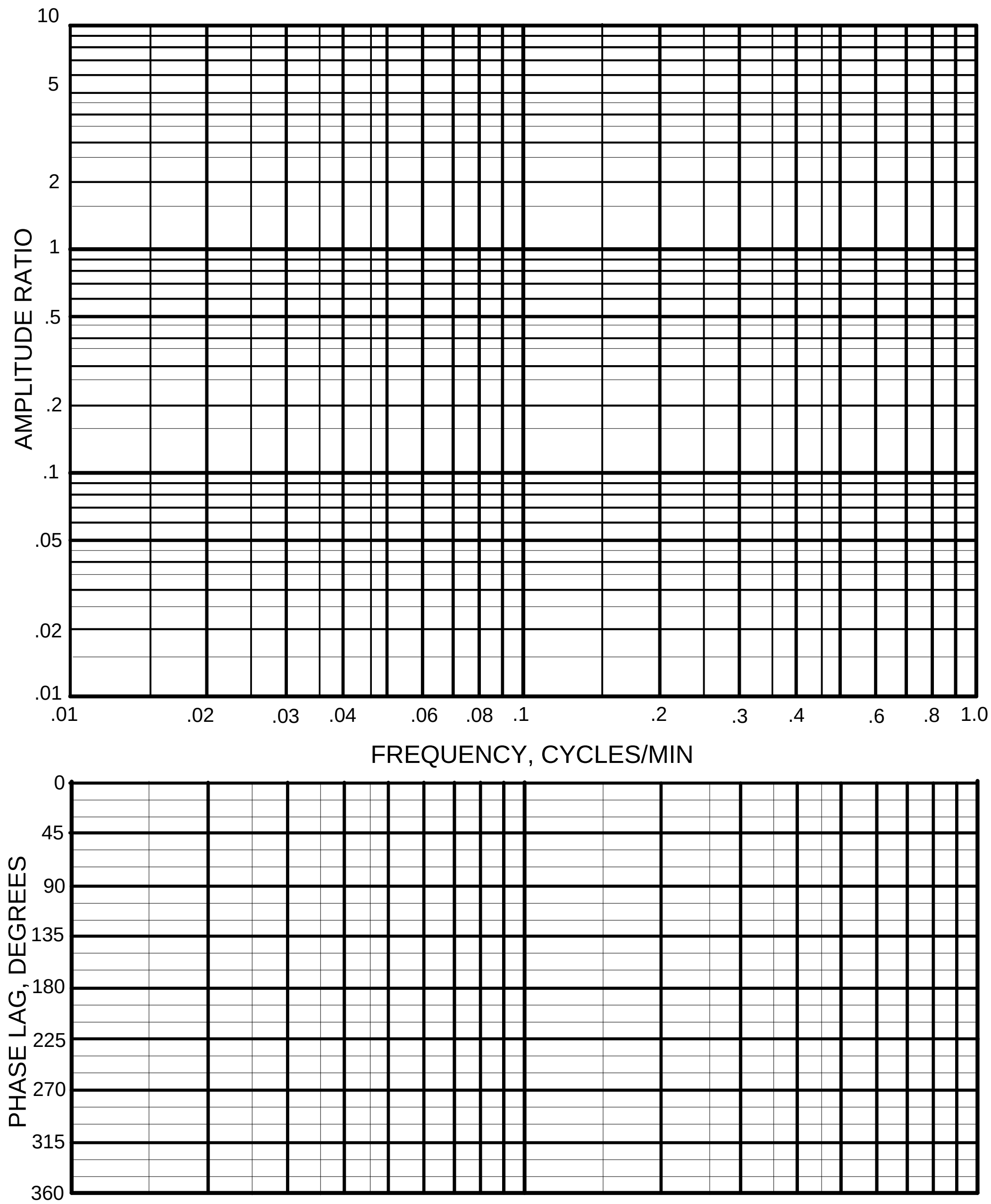
Period (Minutes)	Frequency (cyc/min)	Cont Output Amplitude	PV Amplitude	Amplitude Ratio	Time Lag	Phase Shift Degrees
60	0.0167	10.0				
30	0.0333	10.0				
15	0.0667	20.0				
7.5	0.1333	20.0				
3.75	0.2667	20.0				

Plot these values on the graphs on the following page.

(Note: On many Bode plots, **decibels**, rather than amplitude ratio, is plotted on a linear scale. The conversion equation from amplitude ratio to decibels is:

$$\text{Decibels} = 20\,\text{Log}_{10}(\text{Amplitude Ratio})$$

Also, frequency is usually plotted on a logarithmic scale reading in **radians per time unit**, rather than cycles per time unit. A complete Bode plot is beyond the scope of this laboratory exercise, however.)


AMPLITUDE RATIO
10
5
2
1
.5
.2
.1
.05
.02
.01
.01 .02 .03 .04 .06 .08 .1 .2 .3 .4 .6 .8 1.0
FREQUENCY, CYCLES/MIN
PHASE LAG, DEGREES
0
45
90
135
180
225
270
315
360