MANUAL FOR RECTILINEAR AND TORSIONAL POSITION CONTROL SYSTEM

Prof. R.A. de Callafon, Dept. of MAE, UCSD, version 3.1415

ECP HARDWARE & SOFTWARE

Turning on Hardware

Turn on the ECP (model 205 or 210) control box by pushing the **BLACK ON** button. Make sure the mechanical system (weights/springs) is located in the correct position before you start your experiments. The ECP control box also has a **RED OFF** button. Use the **RED OFF** button to turn of the hardware in case of emergency.

In Case of Emergency

In case you experience a wildly moving mechanical system or when the mechanical system is vibrating strongly, make sure to push the **RED OFF** button on the front of the ECP control box. Turning of the ECP control box avoids damage to the experiment in case you specified the wrong control algorithm (unstable) and or reference signals that are too large. At all times, keep away from any moving parts. Make sure to push the **RED OFF** button also when someone is too close (less than 4 inches) to the experiment.

Start ECP software

Log in with your username and password and start the ECP software via the ECP icon on the desktop. Once ECP starts up, you will see a window similar to Figure 1.

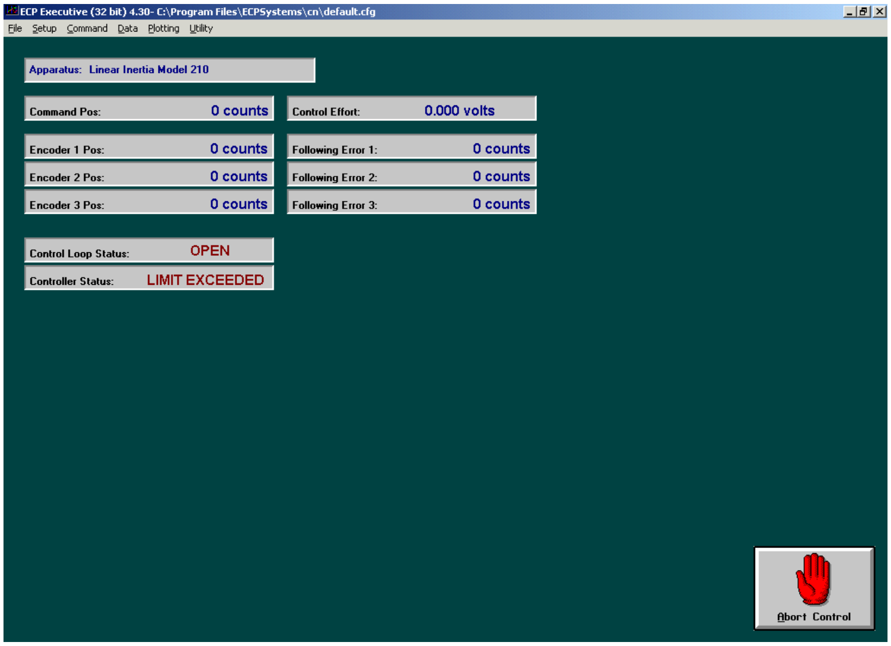


Figure 1: ECP main window

Open-loop versus Closed-loop experiments

Open-loop experiments (no feedback controller implemented) are used to study the dynamics of the uncontrolled mechanical system. Closed-loop experiments require the *design of a feedback control algorithm* that continuously monitors the encoder positions and computes control signals for the Servo Motor to control the forces applied to the mechanical system. In summary:

- Open-loop experiments consist of applying *input signals to the Servo Motor* and *measuring the outputs* (encoder position of the different rigid bodies) of the mechanical system via Encoder 1 and/or 2. These experiments are done typically during your 1st and 2nd week of your lab.
- Closed-loop experiments consist of applying *reference signals to the Servo Motor control loop* and *measuring the outputs* (encoder position of the different rigid bodies) of the mechanical system. These experiments are done typically during your 2nd and 3rd week of your lab.

OPEN-LOOP EXPERIMENTS (1st and 2nd week of lab)

An open-loop experiment for the rectilinear or torsional system only requires the specification of a open-loop trajectory. For the specification of an open-loop trajectory, please follow these steps:

Setup of open-loop trajectory

1. First turn of any closed-loop control algorithm by clicking the large **Abort Control** button in the main ECP window (see Figure 1).
2. Select **Command - Trajectory** from the main menu bar (see Figure 1) to specify the test signal (called trajectory) for the servo Motor. A window similar to Figure 2 will open, allowing you to specify various test signals (trajectories). Typically we will use Step and Sinusoidal inputs.

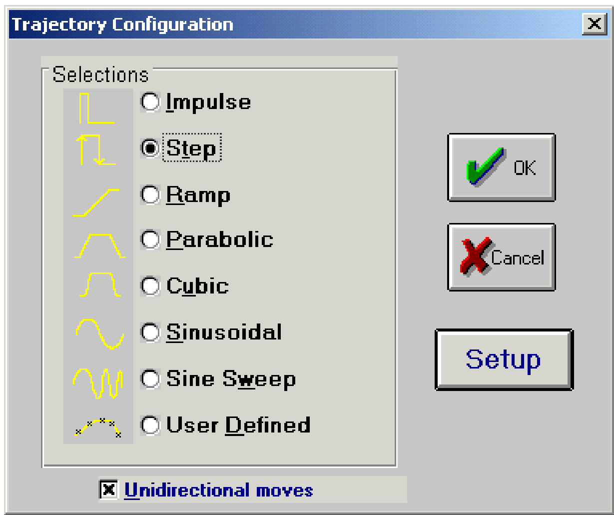


Figure 2: Trajectory Configuration Window

3. Select **Step** and click on **Setup** button to set up the step signal. A window similar to Figure 3 will open, allowing the specification of the Step Size, Dwell Time (how long is the step) and the number of repetitions.

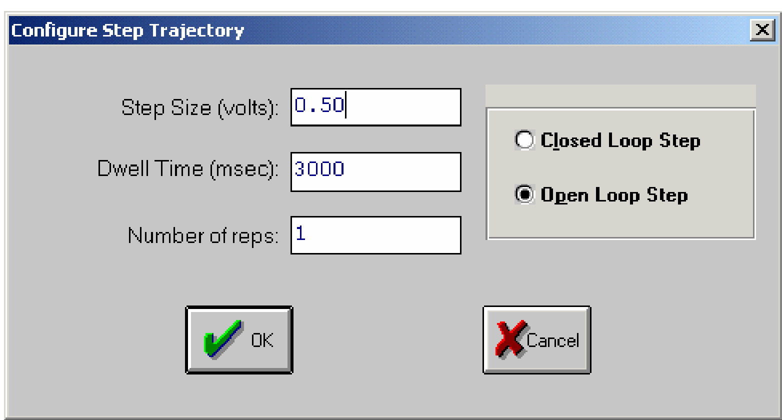


Figure 3: Configure Step Trajectory Window for an open-loop step

For an open-loop experiment, make sure to select **Open-Loop Step**. This causes the **Step Size** to be expressed in units of Volts, indicating an input signal on the Servo Motor in Volts.

NOTE: open-loop step sizes should be limited to a maximum of 0.75 Volt to avoid violent movements and damage to the mechanical system.

4. Close the Configure Step Trajectory window (Figure 3) by a click on the **OK** button and close the Trajectory Configuration window (Figure 2) also by a click on the **OK** button

EXPERIMENTS & DATA ACQUISITION

Performing experiments (open- and/or closed-loop) requires setting up data acquisition parameters to indicate which signal should be measured during the experiment. Subsequently the experiment must be run to upload the experimental data. These steps are described in the following.

Set up data acquisition

Make sure you have set up the correct (open-loop or closed-loop) experiment as described earlier.

1. Select **Data - Setup Data Acquisition** from the main menu bar (See Figure 1) to specify which signals to measure during your open-loop experiment. A window similar to Figure 7 should appear.

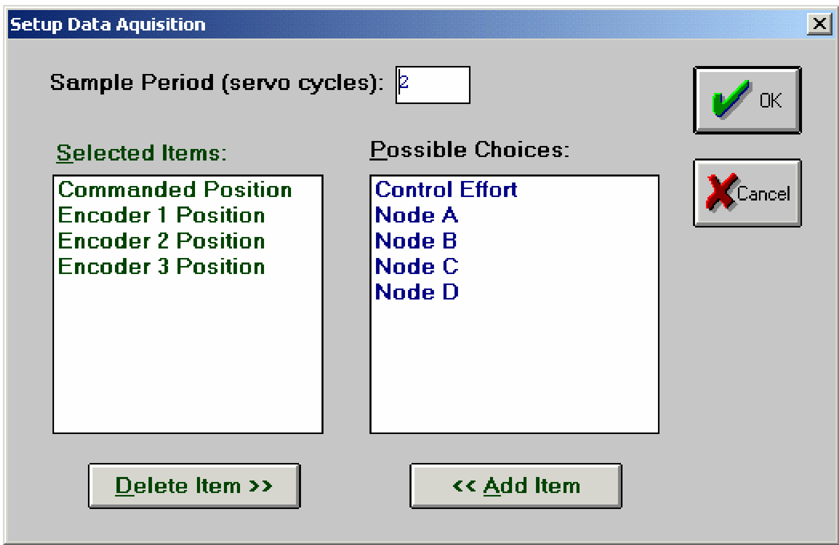


Figure 7: Setup Data Acquisition Window

2. Select the data you would like to measure by clicking on the following names:

Encoder 1 Position	=	Output Position of Mass/Inertia 1
Encoder 2 Position	=	Output Position of Mass/Inertia 2
Encoder 3 Position	=	Output Position of Mass/Inertia 3 (typically NOT used)
Control Effort	=	Input Signal to Servo Motor
Commanded Position	=	Reference signal specified under Setup Trajectory (see above)

   Click on the **Delete Item** or **Add Item** buttons to respectively exclude or include that variable.

3. Think which data you would like to measure when you do your experiment. If you do an *open-loop experiment*, you probably only want to measure **Control Effort**, **Encoder 1 Position** and/or **Encoder 2 Position**. For **closed-loop experiments** it is worthwhile to also measure the **Commanded Position** (reference) signal to inspect the steady-state error. Click on **OK** button to close the Setup Data Acquisition Window.

Perform Experiment for Data Acquisition

Make sure your have set up the data acquisition according to the steps described above.

1. Select **Utility - Zero Position** from the main menu (see Figure 1) to reset all encoder values to 0. This gives nice plots that will start at 0. You might have to do this several times in case of closed-loop experiments.
2. Select **Command - Execute** from the main menu to execute your experiment and a window similar to Figure 8 will open.

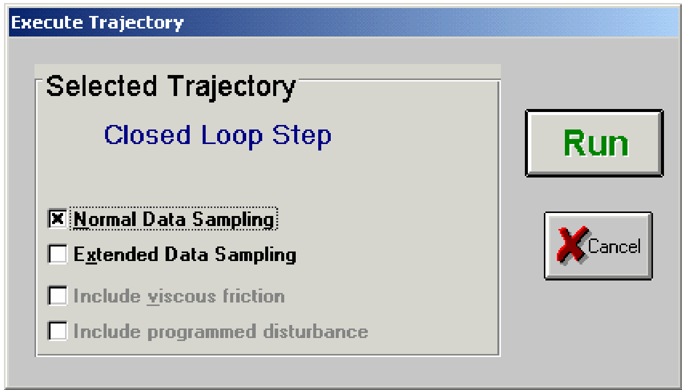


Figure 8: Execute Trajectory Window

Click on **RUN** button to start your experiment and data logging.

NOTE: have someone keep his/her your finger close to the RED OFF button on the front of the ECP control box. In case the algorithm or the trajectory results in a violently moving mechanical system, push the RED OFF button on the front of the ECP control box.

3. After the experiment ran successfully (no violent movement of mechanical system), an Upload Successful window similar to Figure 9 appears.

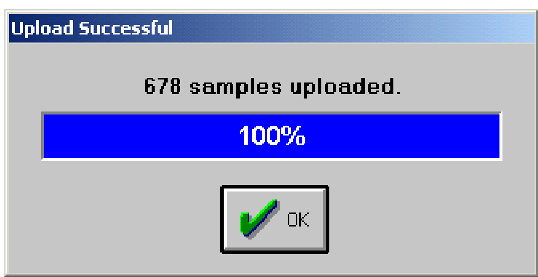


Figure 9: Upload Successful window after experiments completed

Click on **O.K.** to finish your experiment. If the Upload Successful window appears very quickly, the mechanical system might have hit the safety switches, causing the experiment to be terminated abruptly. In addition the control algorithm will be shut off and requires re-implementation. Try to reduce your step size or adjust the (PID) control parameters in case of closed-loop control to avoid this error message.

PLOTTING & SAVING DATA

Plotting Data

Make sure you were able to perform a succesfull experiment (no violent movement of the mechanical system) and that you were able to upload the data according to the steps described above.

1. Select **Plotting - Setup Plot** from the main menu (see Figure 1) to plot and examine the data from your experiment and a window similar to Figure 10 will appear.

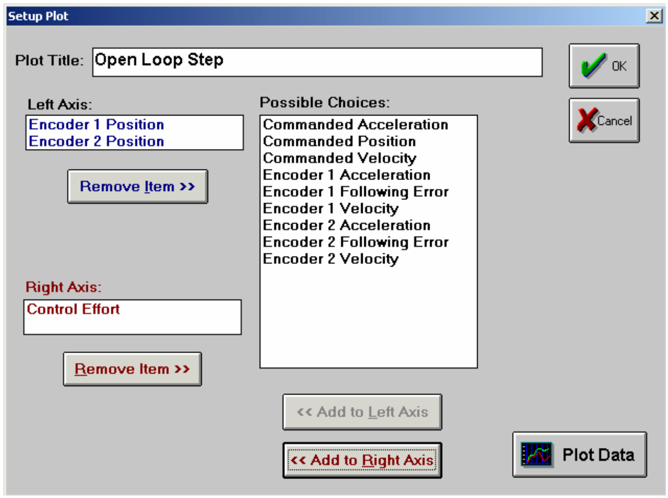


Figure 10: Setup Plot window

2. Select the data you would like to plot by clicking on the names and click on the **Remove Item** or **Add to Left Axis** or **Add to Right Axis** buttons to respectively exclude or include the plot of that variable. You can plot a maximum of 2 variables per axis and depending what you selected under Setup Data Acquisition, the following variables can be plotted.

   Encoder 1 Position = Position of Mass/Inertia 1
   Encoder 2 Position = Position of Mass/Inertia 2
   Encoder 3 Position = Position of Mass/Inertia 3
   Control Effort = Input Signal to Servo Motor
   Commanded Position = Trajectory specified

   Note that Velocity and Acceleration measurements are found by numerically differentiating the measurements and tend to be noisy!
3. For the open-loop experiments, it is best to plot the output (**Encoder 1 and/or 2 Position**) on the **Left Axis** and the input (**Control Effort**) on the **Right Axis**. For closed-loop experiments, it is best to plot the output (**Encoder 1 or 2 Position**) and the **Commanded Position** on the same axis to inspect overshoot and steady state errors.
4. Click on **Plot Data** button and a plot of the data will appear in a new window.

Saving Data

Make sure you have performed a successful experiment and were able to plot the data as described in the steps above. Whenever you run a new experiment, the data of the previous experiment will be overwritten in memory. So if you like your measurement and you would like to save the data for your report for plotting purposes, follow these steps:

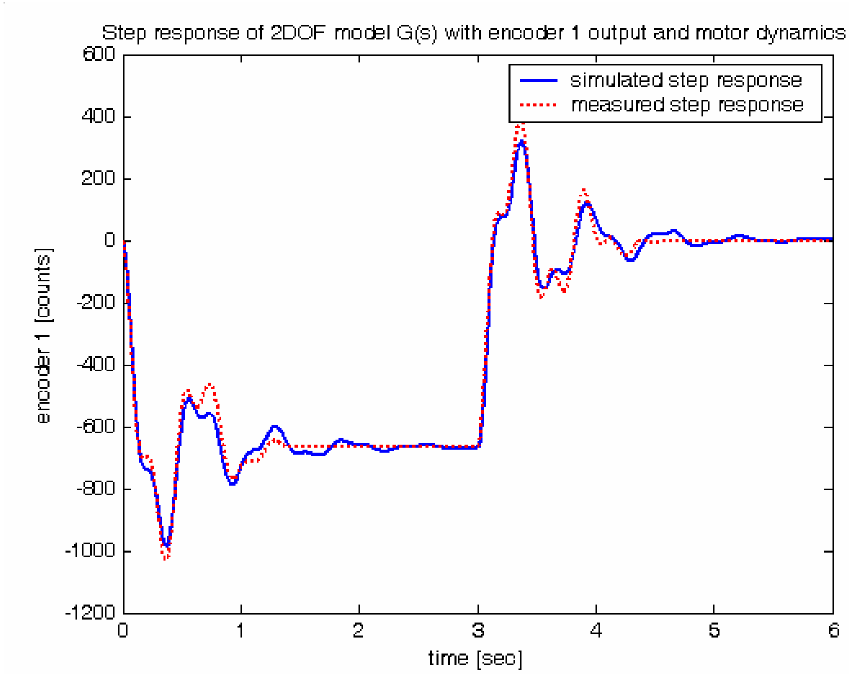
1. Select **Data - Export Raw Data** from the main menu (See Figure 1) to save your data before you start a new experiment.
2. When saving the raw data, make sure you save it in your directory under **C:\labcourse**\
3. The saved data will be text file where the data is stored column-wise and can be opened with Notepad and/or Excel. The data file can directly be read by the Matlab program **ecpread** available in your directory. The Matlab program **ecpread** is also used by the Matlab script **maelab** available in your directory to plot your simulation and experimental results for Model Validation and Controller validation (see next page).

4. You can also modify the text file into an m-file so Matlab can read the data. Details to convert the text file into an m-file can also be found in your lab handout and requires the following editing steps:
   a. First line in text file: Comment out the first line with **%**
   b. Second line in text file: Enter **dummy=** before the opening bracket **[**.
   c. Last line in text file: put a semicolon **;** behind the closing bracket **]**.
   d. After last line in text file: define time vector **t**, input vector **u** and output **y** by **selecting the appropriate columns from the dummy variable**. For example, if you have selected to save the control effort (input **u**) and the encoder 1 position (output **y**), this can be done by adding the following lines to the end of the text file:
      t=dummy(:,2);
      y=dummy(:,3);
      u=dummy(:,4);
      clear dummy;
5. Save the raw text file as a file with the extension **.m**. Result is a Matlab script that can be run to read in your measurements.
6. Final note: make sure you use the variables **t**, **u** and **y** to define respectively the time vector, the input vector and the output vector. This allows the data file to also be read by the script file **maelab.m** to validate your models.

VALIDATION OF MODELS

Comparing experimental data with a simulation can validate models of the ECP rectilinear and torsional system. Matlab can handle experimental data and simulations and you are provided with a script file called **maelab** to perform all the necessary simulation, validation and control steps. To use the **maelab** script file, follow the following steps:

1. Start Matlab
2. In the Matlab command window, type in **pwd** and verify that you are indeed in your working directory under C:\labcourse\
3. Edit the file **parameters.m** by typing in **edit parameters** to specify the parameters of your model. Make sure to save the **parameters.m** file before you continue.
4. Run **maelab** script file by typing in **maelab** in the Matlab command window.
5. Specify the encoder output (1 or 2) you are interested
6. Specify the name of the filename that contains your model parameters (default **parameters.m**)
7. Specify degrees of freedom you would like to simulate. Specifying 1 will simulate a 1 Degree of Freedom (DOF) mass/damper/spring system, using only the model parameters m1, d1 and k1. Specifying 2 will simulate a 2DOF mass/spring/damper system using m1, d1, k1 and m2, d2, k2.
8. Specify the sign of the encoder (1 or -1). This is necessary when a **positive step** on the system results in a **negative reading** (use -1) of the encoder output.
9. Use the menu option to "simulate open loop step response" or "simulate open loop sinusoidal response" and enter the values for the simulation. Typically the values should be the same as done during the experiment to be able to compare simulations with experiments.
10. NOTE: when asked for a filename, make sure to put the filename between quotes, e.g. **`myfile'**
11. For validation purposes, by the end of the 2nd week of your 171b lab, you should have a simulation and an experiment (based on step or sinusoidal excitation) that show close resemblance, similar to the figures below.

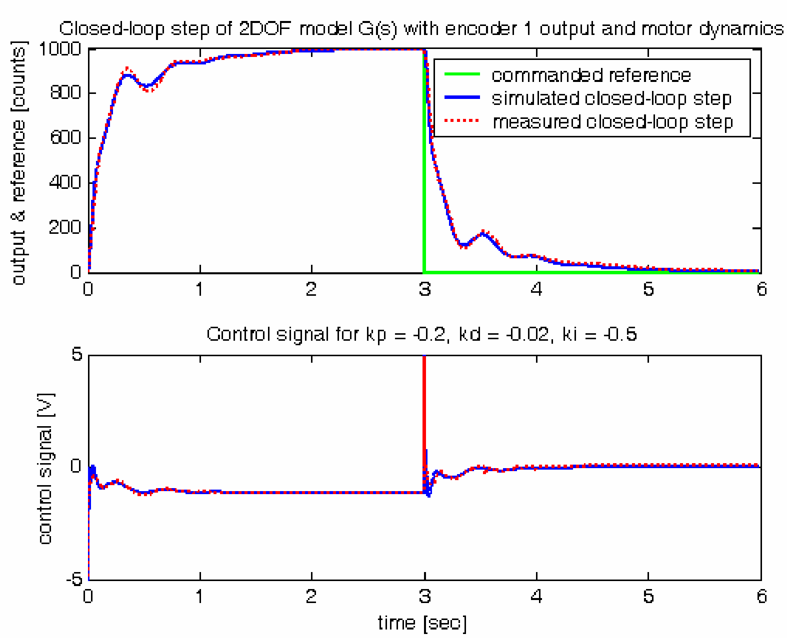


Example of figure for open-loop step-based model validation at the end of week 2

VALIDATION OF CONTROL

Before implementing a (new) control algorithm, first verify the performance of your proposed P-, PD- or PID-control algorithm with Matlab by running a closed-loop simulation with the Matlab script file called **maelab**. Based on the model that you have validated (as indicated above) It allows you to verify whether your control algorithm will be stable on the actual system and to **verify whether the control signals stay within bounds and are not subjected to excessive oscillations** when making a step on the reference signal. The procedure to **validate the control algorithm before implementation** is as follows:

1. Start Matlab
2. In the Matlab command window, type in **pwd** and verify that you are indeed in your working directory under C:\labcourse\
3. Edit the file **parameters.m** by typing in **edit parameters** to specify the parameters of your model. Make sure to save the **parameters.m** file before you continue.
4. Run **maelab** script file by typing in **maelab** in the Matlab command window.
5. Specify the encoder output (1 or 2) you are interested
6. Specify the name of the filename that contains your model parameters (default **parameters.m**)
7. Specify degrees of freedom you would like to simulate. For validation of the control algorithm you must **specifying 2 to simulate the full 2DOF** mass/spring/damper system using m1, d1, k1 and m2, d2, k2 parameters.
8. Specify the sign of the encoder (1 or -1). This is necessary when a **positive step** on the system results in a **negative reading** (use -1) of the encoder output.
9. Use the menu option to "Design/evaluate feedback controller" and enter the numerical values for the **kp (proportional gain)**, **kd (derivative gain)** and **ki (integral gain)**. Keep in mind the bounds on the gains to avoid excessive control signals: **|kp| < 1, |kd| < 0.02 and |ki| < 1**.
10. Make sure to motivate and argument the choice of your kp, kd and ki values in your lab report based on the figures being created by **maelab** script file.
11. Use the menu option to "Simulate closed loop step response" and enter the values for the step response simulation. The values should be the same as done during an experiment for comparison purposes. Typically a closed-loop step should be in the order of 1000 counts.
12. NOTE: when asked for a filename, make sure to put the filename between quotes, e.g. **`myfile'.** Initially, for control validation purposes, no experimental data is required.
13. For final validation purposes at the end of the 3rd week, you should have a simulation and an experiment (based on a closed-loop step response) that show close resemblance, similar to the figure below.



Example of figure for validation of control algorithm based on closed-loop step data and simulation

CLOSED-LOOP EXPERIMENTS (2nd and 3rd week of lab)

A closed-loop experiment for the rectilinear or torsional system requires the specification of both a control algorithm and a closed-loop trajectory. For the specification of the control algorithms and the closed-loop trajectory, please follow these steps:

Setup of Control Algorithm

1. Select **Setup - Control Algorithm** from the main menu bar (see Figure 1) to specify the servo control algorithm. A window similar to Figure 4 will open.

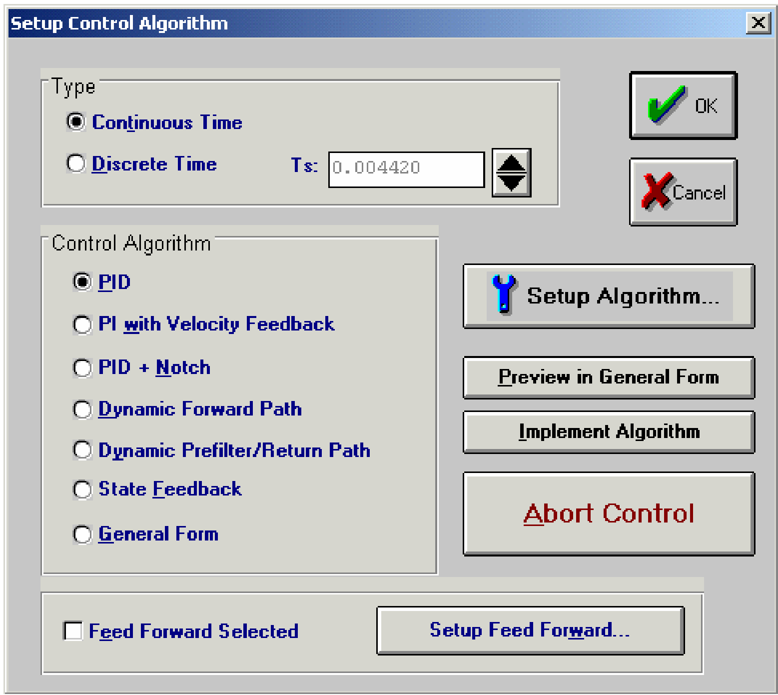


Figure 4: Setup Control Algorithm Window

2. Make sure *Type* is set to **Continuous Time** and the *Control Algorithm* is set to *PID* to be able to specify a PID controller.
3. Click on the **Setup Algorith**m button to set up the PID control algorithm and a window similar to Figure 5 will appear.

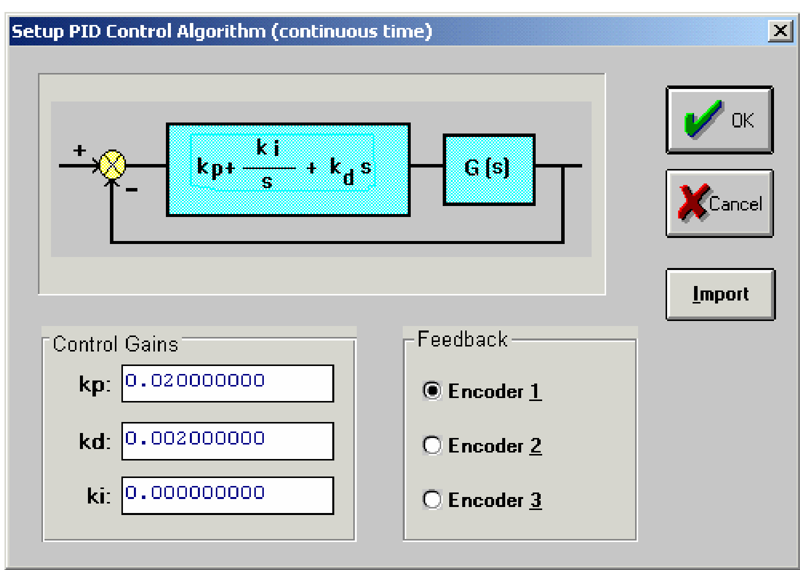


Figure 5: Setup PID Control Algorithm (continuous time) window

4. Make sure you select the correct **Encoder 1, 2 or 3** under *Feedback* and specify the value of the **Kp** (proportional gain), **Kd** (derivative gain) and the *Ki* (integral gain) in this window. Click on **OK** button to close the Setup PID Control Algorithm (continuous time) window.

> **NOTE: make sure Kp, Kd and Ki satisfy the following bounds to avoid excessive control signals: 0<Kp<1, 0<Kd<0.02 and 0<Ki<1. Especially the bound on Kd of 0.02 is important as large derivative gains cause strong vibrations in the mechanical system that should be avoided at all times!**

5. In the Setup Control Algorithm (see Figure 4) make sure to click the **Implement Algorithm** button to actually implement the control algorithm.

> **NOTE: keep the mouse hovering over the Abort Control button and give a small tap to the mechanical system. If this results in excessive movement of vibrations of the mechanical system your control algorithm may be destabilizing and a direct click on the Abort Control button shuts of the control algorithm.**

6. If the implementation of the control algorithm does not result in excessive movement of vibrations of the mechanical system click the **OK** button to close the Setup Control Algorithm (see Figure 4) window. You should see the control is being active by the **Control Loop Status: CLOSED** in the main window of ECP (see Figure 1).

Setup of closed-loop trajectory

1. Select **Command - Trajectory** from the main menu bar (see Figure 1) to specify the test signal (called trajectory) for the servo Motor. A window similar to Figure 2 will open, allowing you to specify various test signals (trajectories). Typically we will use Step and Sinusoidal inputs.
2. Select Step and click on Setup button to set up the step signal. A window similar to Figure 6 will open, allowing the specification of the Step Size, Dwell Time (how long is the step) and the number of repetitions.

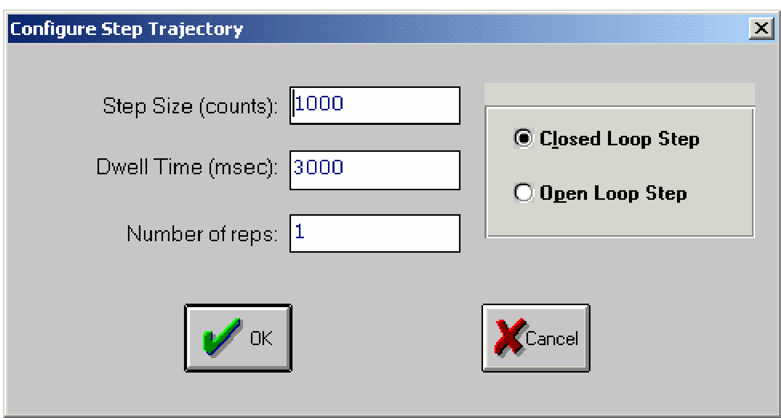


Figure 6: Configure Step Trajectory window for a closed-loop step

For a closed-loop experiment, make sure to select **Closed-Loop Step**. This causes the Step Size to be expressed in units of counts, indicating a reference signal on the Servo Motor control loop in Encoder counts.

> **NOTE: closed-loop step size should be limited to a maximum of 4000 counts to avoid violent movements and damage to the mechanical system.**

3. Close the Configure Step window (Figure 6) by a click on the **OK** button and close the Trajectory Configuration window (Figure 2) also by a click on the **OK** button.