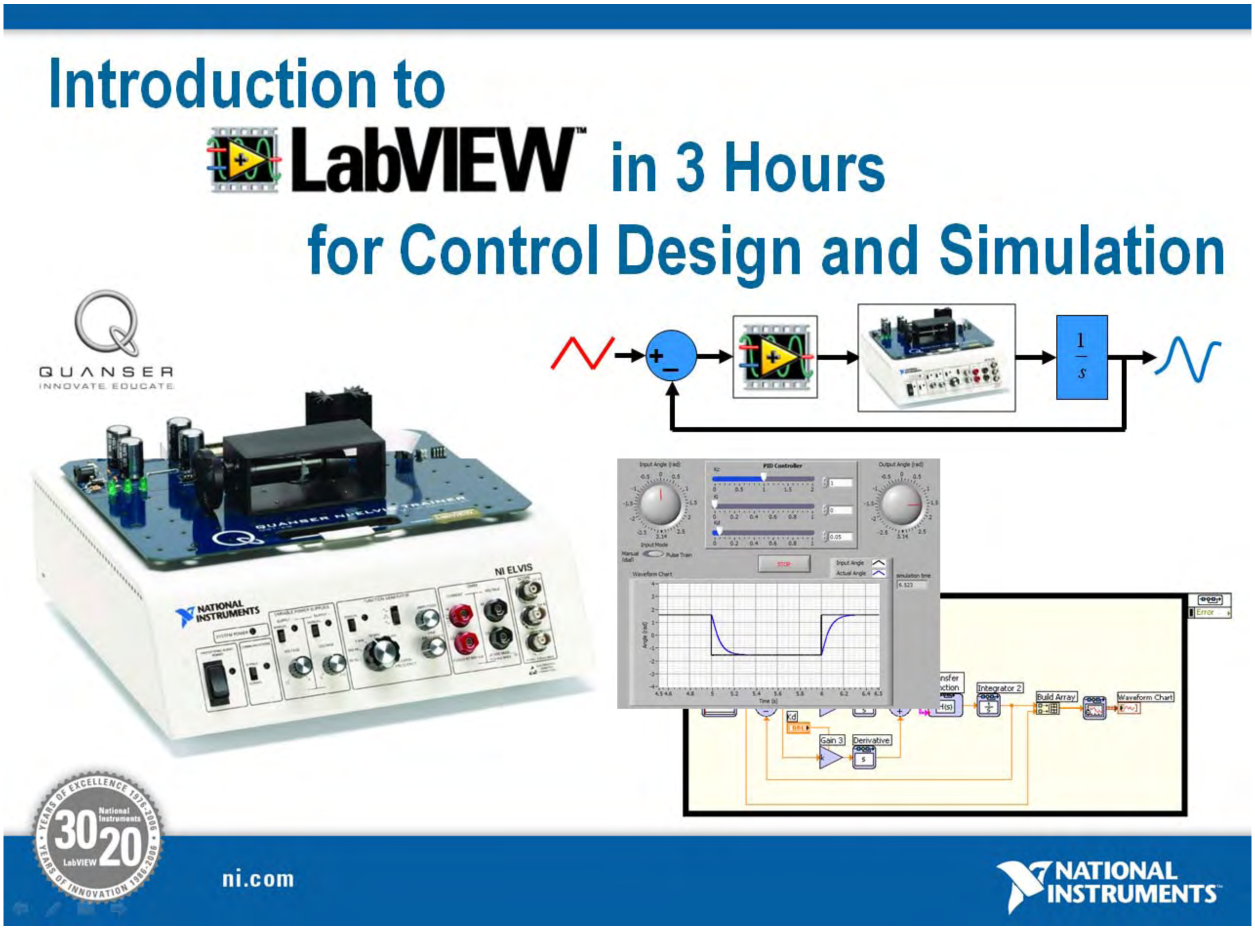
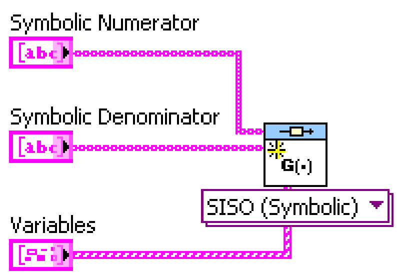
NATIONAL INSTRUMENTS

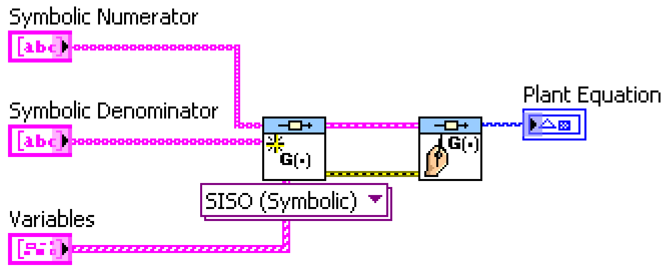


Exercise 1a:

1. Build the VI block diagram
   - ☐ Open a blank VI from the “Getting Started” screen. Save it as `Exercise_1a.vi`.
   - ☐ Switch to the block diagram of the VI (<Ctrl+E>).
   - ☐ Open the Functions Palette by right-clicking anywhere in the block diagram. Dock the palette by clicking the icon at the top-left corner of the palette window.
   - ☐ Place the `CD Construct Transfer Function Model.vi` on the block diagram. From the functions palette, navigate to **Control Design & Simulation » Control Design » Model Construction » CD Construct Transfer Function Model.vi**, and place it on the block diagram. From the drop-down menu select **Single-Input Single-Output (Symbolic)**.
   - ☐ Right-click the **Symbolic Numerator** input terminal of the `CD Construct Transfer Function Model.vi`, and select **Create » Control** from the shortcut menu.
   - ☐ Right-click the **Symbolic Denominator** input terminal of the `CD Construct Transfer Function Model.vi`, and select **Create » Control** from the shortcut menu.
   - ☐ Right-click the **Variables** input terminal of the `CD Construct Transfer Function Model.vi`, and select **Create » Control** from the shortcut menu.
   - ☐ The block diagram should look like this:



   - ☐ Place the `CD Draw Transfer Function Equation.vi` on the block diagram. From the functions palette, navigate to **Control Design & Simulation » Control Design » Model Construction » CD Draw Transfer Function Equation.vi**, and place it on the block diagram.
   - ☐ Wire the **Transfer Function Model** output terminal of the `CD Construct Transfer Function Model.vi` to the **Transfer Function Model** input terminal of the `CD Draw Transfer Function Equation.vi`.
   - ☐ Right-click the **Equation** output terminal of the `CD Draw Transfer Function Equation.vi`, and select **Create » Indicator** from the shortcut menu.
   - ☐ Triple-click the label of the picture indicator, and rename it `Plant Equation`.
   - ☐ The block diagram should look like this:



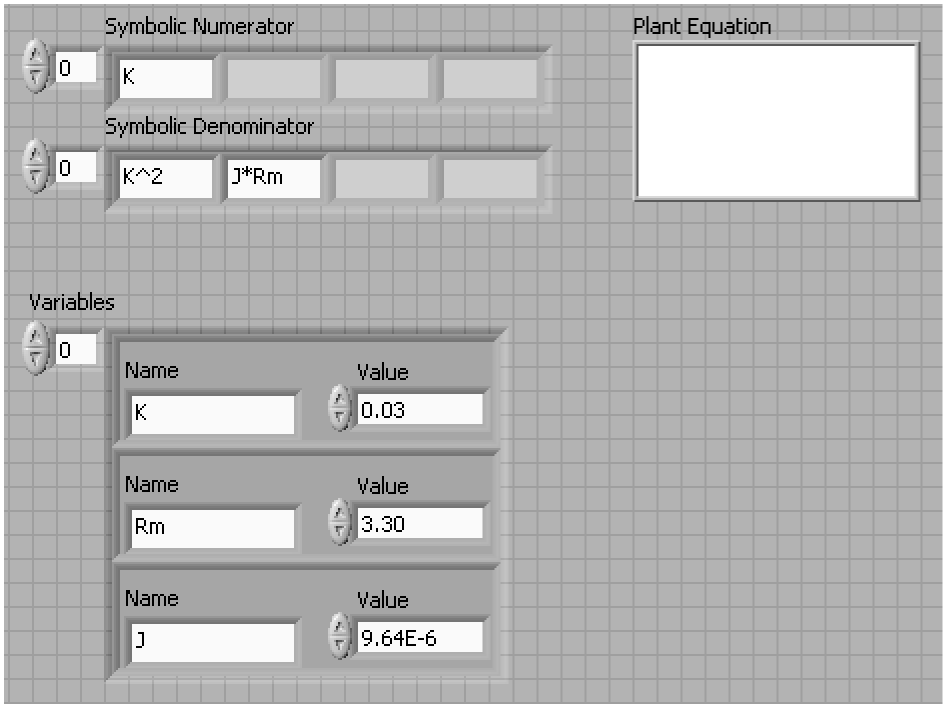
- ☐ Save the VI (<Ctrl+S>).

2. Run the VI
   - ☐ Switch to the front panel of the VI (<Ctrl+E>).
   - ☐ Right-click the `Value` numeric control inside the `Variables` control and select **Display Format…**, then select **Automatic Formatting**. This will allow the control to display small numbers properly.
   - ☐ On the front panel, set the values of the `Symbolic Numerator`, `Symbolic Denominator`, and `Variables` controls according to the following table:

     [redacted]

     Note: Symbolic variables are case sensitive, so make sure to use consistent capitalization.

   - ☐ Set the front panel control values to default. Select **Edit » Make Current Values Default**.
   - ☐ The front panel should look like this:



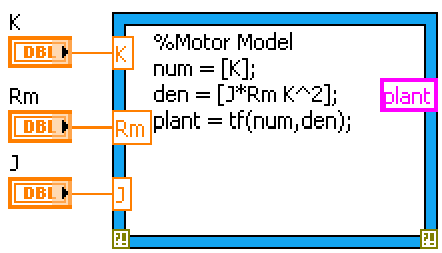
- ☐ Run the VI (<Ctrl+R>). Observe the transfer function drawn in the `Plant Equation` indicator. This transfer function will be used in future exercises.
- ☐ Save the VI (<Ctrl+S>).

Exercise 1b:

1. Build the VI block diagram
    - ☐ Open a blank VI from the "Getting Started" screen. Save it as `Exercise_1b.vi`.
    - ☐ Switch to the block diagram of the VI (<Ctrl+E>).
    - ☐ Open the Functions Palette by right-clicking anywhere in the block diagram. Dock the palette by clicking the icon at the top-left corner of the palette window.
    - ☐ Place a MathScript Node on the block diagram. From the functions palette, navigate to **Programming » Structures » MathScript Node**, and draw a rectangle on the block diagram to define the node.
    - ☐ Enter the following code inside the MathScript Node:

    ```
    %Motor Model
    num = [K];
    den = [J*Rm K^2];
    plant = tf(num,den);
    ```

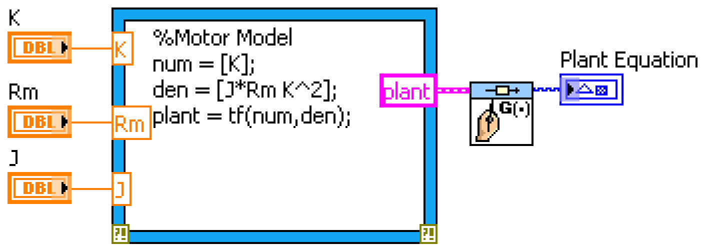
    - ☐ Right-click on the left border of the MathScript Node and select **Add Input**. Enter K as the input. Right-click on the input terminal of the K variable and select **Create » Control** from the shortcut menu.
    - ☐ Right-click on the left border of the MathScript Node and select **Add Input**. Enter `Rm` as the input. Right-click on the input terminal of the `Rm` variable and select **Create » Control** from the shortcut menu.
    - ☐ Right-click on the left border of the MathScript Node and select **Add Input**. Enter `J` as the input. Right-click on the input terminal of the `J` variable and select **Create » Control** from the shortcut menu.
    - ☐ Right-click on the right border of the MathScript Node and select **Add Output**. Enter `plant` as the output. Right-click on the output terminal of the `plant` variable and select **Choose Data Type » Add-ons » TF Object**.
    - ☐ The block diagram should look like this:



    - ☐ Place the `CD Draw Transfer Function Equation.vi` on the block diagram. From the functions palette, navigate to **Control Design & Simulation » Control Design » Model Construction » CD Draw Transfer Function Equation.vi**, and place it on the block diagram.
    - ☐ Wire the output terminal of the `plant` variable to **Transfer Function Model** input terminal of the `CD Draw Transfer Function Equation.vi`.
    - ☐ Right-click the **Equation** output terminal of the `CD Draw Transfer Function Equation.vi`, and select **Create » Indicator** from the shortcut menu.

☐ Double-click the label of the picture indicator, and rename it `Plant Equation`.

☐ The block diagram should look like this:



☐ Save the VI (<Ctrl+S>).

2. Run the VI

☐ Switch to the front panel of the VI (<Ctrl+E>).

☐ Set the values of the front panel controls according to the following table:

K	0.028
Rm	3.3
J	9.64E-06

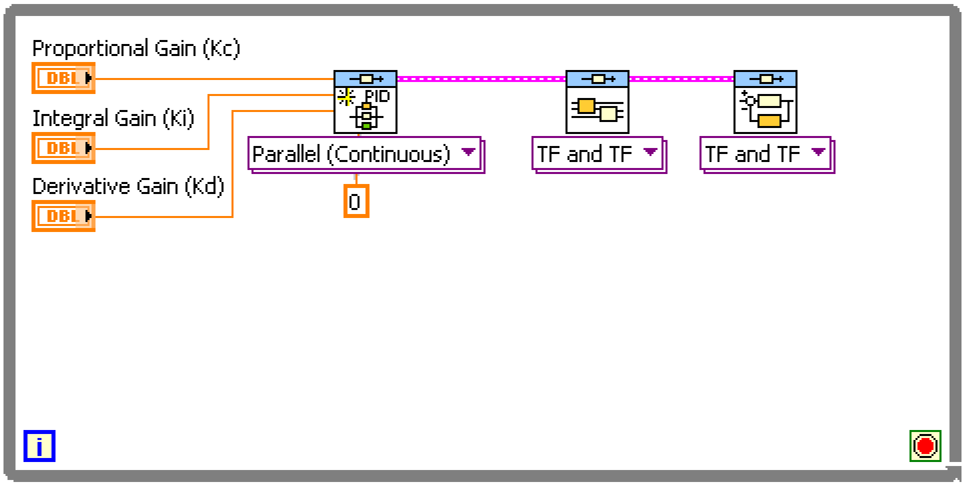
☐ Set the front panel control values to default. Select **Edit » Make Current Values Default**.

☐ Run the VI (<Ctrl+R>). Observe the transfer function drawn in the `Plant Equation` indicator. This transfer function will be used in future exercises.

☐ Save the VI (<Ctrl+S>).

Exercise 2:

1. Build the VI block diagram
   - ☐ Open a blank VI from the “Getting Started” screen. Save it as `Exercise_2.vi`.
   - ☐ Switch to the block diagram of the VI (<Ctrl+E>).
   - ☐ Open the Functions Palette by right-clicking anywhere in the block diagram. Dock the palette by clicking the icon at the top-left corner of the palette window.
   - ☐ Place a **While Loop** on the block diagram. From the functions palette, navigate to **Programming » Structures » While Loop**. Draw a rectangle on the block diagram to define the loop.
   - ☐ Place the **CD `Construct PID Model.vi`** on the block diagram. From the functions palette, navigate to **Control Design & Simulation » Control Design » Model Construction » CD Construct PID Model.vi**, and place it on the block diagram. From the drop-down menu select **PID Parallel**.
   - ☐ Right-click on the **Kc** input terminal of the CD `Construct PID Model.vi` and select **Create » Control** from the shortcut menu.
   - ☐ Right-click on the **Ki** input terminal of the CD `Construct PID Model.vi` and select **Create » Control** from the shortcut menu.
   - ☐ Right-click on the **Kd** input terminal of the CD `Construct PID Model.vi` and select **Create » Control** from the shortcut menu.
   - ☐ Right-click on the **High Frequency Time Constant [s] (Tf)** input terminal of the CD `Construct PID Model.vi` and select **Create » Constant** from the shortcut menu.
   - ☐ Place the **CD `Series.vi`** on the block diagram. From the functions palette, navigate to **Control Design & Simulation » Control Design » Model Interconnection » CD Series.vi**, and place it on the block diagram. From the drop-down menu select **Transfer Function and Transfer Function**.
   - ☐ Wire the **Transfer Function Model** output terminal of the CD `Construct PID Model.vi` to the **Model 1** Input terminal of the CD `Series.vi`.
   - ☐ Place the **CD `Feedback.vi`** on the block diagram. From the functions palette, navigate to **Control Design & Simulation » Control Design » Model Interconnection » CD Feedback.vi**, and place it on the block diagram. From the drop-down menu select **Transfer Function and Transfer Function**.
   - ☐ Wire the **Series Model** output terminal of the CD `Series.vi` to the **Model 1** input terminal of the CD `Feedback.vi`.
   - ☐ The block diagram should look like this:



☐ Place the **CD `Step Response.vi`** on the block diagram. From the functions palette, navigate to **Control Design & Simulation » Control Design » Time Response » CD Step Response.vi**, and place it on the block diagram. From the drop-down menu select **Transfer Function**.

☐ Wire the **Closed Loop Model** output terminal of the CD `Feedback.vi` to the **Transfer Function Model** input terminal of the CD `Step Response.vi`.

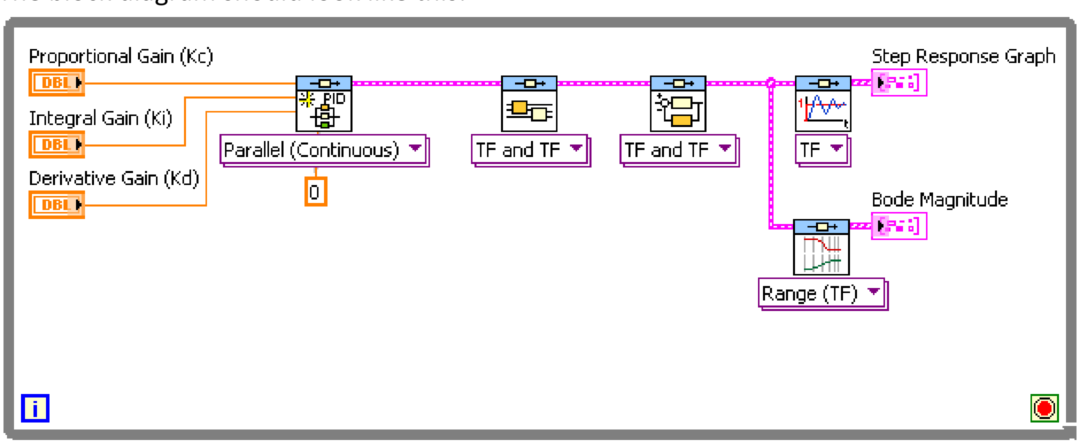
☐ Right-click the **Step Response Graph** output terminal of the CD `Step Response.vi` and select **Create » Indicator** from the shortcut menu.

☐ Place the **CD `Bode.vi`** on the block diagram. From the functions palette, navigate to **Control Design & Simulation » Control Design » Frequency Response » CD Bode.vi**, and place it on the block diagram. From the drop-down menu select **Frequency Range » Transfer Function**.

☐ Wire the **Closed Loop Model** output terminal of the CD `Feedback.vi` to the **Transfer Function Model** input terminal of the CD `Bode.vi`.

☐ Right-click the **Bode Magnitude** output terminal of the CD `Bode.vi` and select **Create » Indicator** from the shortcut menu.

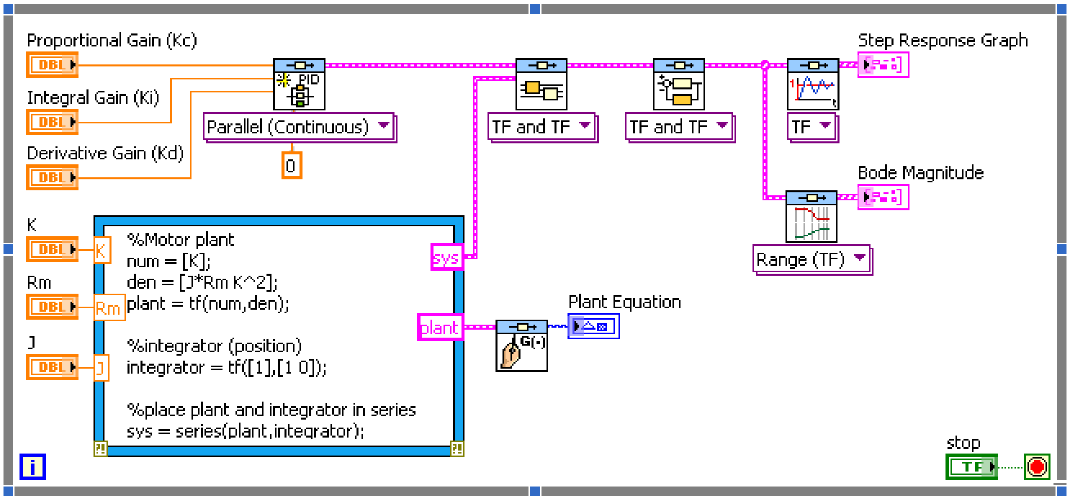
☐ The block diagram should look like this:



- ☐ Place a **MathScript Node** on the block diagram. From the functions palette, navigate to **Programming » Structures » MathScript Node**, and draw a rectangle on the block diagram to define the node.
- ☐ Enter the following code inside the MathScript Node:

```
%Motor plant
num = [K];
den = [J*Rm K^2];
plant = tf(num,den);
%integrator (position)
integrator = tf([1],[1 0]);
%place plant and integrator in series
sys = series(plant,integrator);
```

- ☐ Right-click on the left border of the MathScript Node and select **Add Input**. Enter K as the input. Right-click on the input terminal of the K variable and select **Create » Control** from the shortcut menu.
- ☐ Right-click on the left border of the MathScript Node and select **Add Input**. Enter `Rm` as the input. Right-click on the input terminal of the `Rm` variable and select **Create » Control** from the shortcut menu.
- ☐ Right-click on the left border of the MathScript Node and select **Add Input**. Enter `J` as the input. Right-click on the input terminal of the `J` variable and select **Create » Control** from the shortcut menu.
- ☐ Right-click on the right border of the MathScript Node and select **Add Output**. Enter `sys` as the output. Right-click on the output terminal of the `sys` variable and select **Choose Data Type » Add-ons » TF Object**.
- ☐ Wire the **sys** output terminal of the MathScript Node to the **Model 2** input terminal of the `CD Series.vi`.
- ☐ Right-click on the right border of the MathScript Node and select **Add Output**. Enter `plant` as the output. Right-click on the output terminal of the `plant` variable and select **Choose Data Type » Add-ons » TF Object**.
- ☐ Place the `CD Draw Transfer Function Equation.vi` on the block diagram. From the functions palette, navigate to **Control Design & Simulation » Control Design » Model Construction » CD Draw Transfer Function Equation.vi**, and place it on the block diagram.
- ☐ Wire the **plant** output terminal of the MathScript Node to the **Transfer Function Model** input terminal of the `CD Draw Transfer Function.vi`.
- ☐ Right-click the **Equation** output terminal of the `CD Draw Transfer Function Equation.vi`, and select **Create » Indicator** from the shortcut menu.
- ☐ Double-click the label of the picture indicator, and rename it `Plant Equation`.
- ☐ Create a control button for the **Stop** function of the While Loop.
- ☐ The block diagram should look like this:



☐ Save the VI (<Ctrl+S>).

2. Run the VI

☐ Switch to the front panel of the VI (<Ctrl+E>).

☐ Set the front panel controls according to the following table:

K	0.028
Rm	3.3
J	9.64E-06
Proportional Gain (Kc)	1
Integral Gain (Ki)	0
Derivative Gain (Kd)	0.05

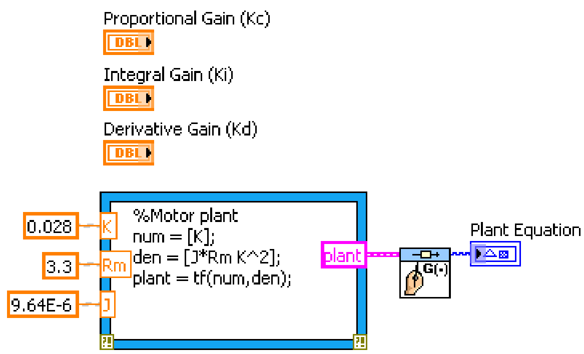
☐ Set the front panel control values to default. Select **Edit » Make Current Values Default**.

☐ Run the VI (<Ctrl+R>). Change the values of `Proportional Gain (Kc)`, `Integral Gain (Ki)`, and `Derivative Gain (Kd)`, and observe how the time and frequency response change.

☐ Save the VI.

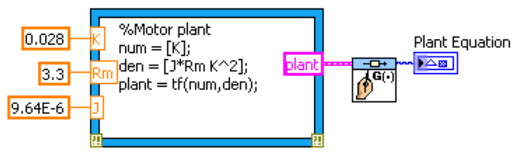
Exercise 3:

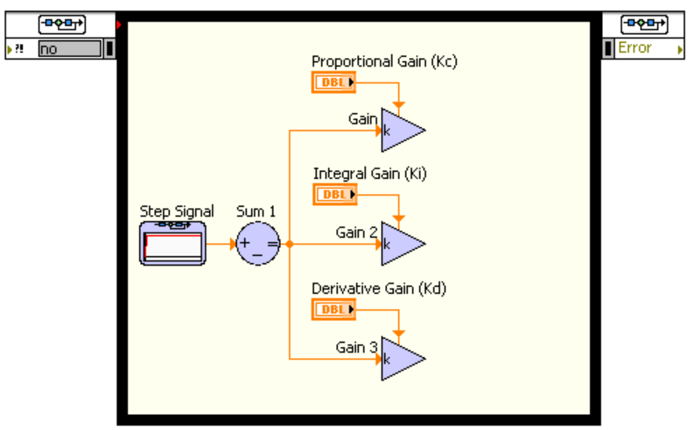
- ☐ Open `Exercise_2.vi`. Save the VI as `Exercise_3.vi`.
- ☐ Go to the block diagram. Remove the While Loop. Right-click the border of the loop and select **Remove While Loop** so that the code inside the loop is not deleted.
- ☐ Change the controls connected to the MathScript Node into constants.
  - o Right-click the K control and select **Change to Constant** from the shortcut menu. Set the value of the constant to 0.028.
  - o Right-click the Rm control and select **Change to Constant** from the shortcut menu. Set the value of the constant to 3.3.
  - o Right-click the J control and select **Change to Constant** from the shortcut menu. Set the value of the constant to 9.64E-6.
- ☐ Remove everything from the block diagram except for the MathScript Node, the constants wired to the MathScript Node, the PID controls (Proportional Gain (Kc), Integral Gain (Ki), and Derivative Gain (Kd)), and the `CD Draw Transfer Function Equation.vi`.
- ☐ Optional: Remove the `sys` output terminal from the MathScript Node, then remove lines 5-10 from the code inside the MathScript Node. (This portion of code creates the `sys` model, which consists of the `plant` model in series with an integrator. In this exercise, the integration will be performed in a Simulation Loop.)
- ☐ The block diagram should look like this:



- ☐ Place a **Simulation Loop** on the block diagram. From the functions palette, navigate to **Control Design & Simulation » Simulation » Simulation Loop**. Draw a rectangle below the MathScript Node on the block diagram to define the loop.
- ☐ Place the `Proportional Gain (Kc)`, `Integral Gain (Ki)`, and `Derivative Gain (Kd)` controls inside the Simulation Loop.
- ☐ Place a **Step Signal** block inside the Simulation Loop. From the functions palette, navigate to **Control Design & Simulation » Simulation » Signal Generation » Step Signal**, and place it in the loop. Double-click the Step Signal block to open the configuration dialog. Set the value of **step time** to 0.01. Click **OK**.
- ☐ Place a **Summation** block inside the Simulation Loop. From the functions palette, navigate to **Control Design & Simulation » Simulation » Signal Arithmetic » Summation**, and place it in the loop. Label the Summation block “Sum 1”.

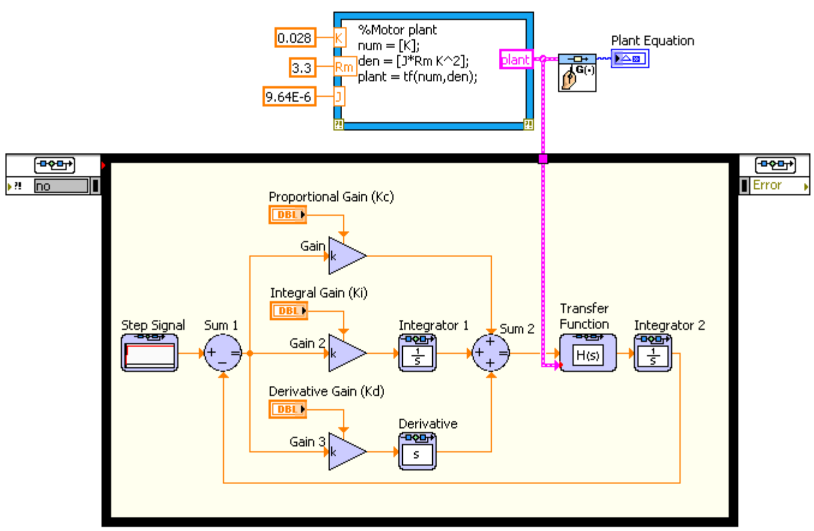
- ☐ Place three **Gain** blocks inside the Simulation Loop. From the functions palette, navigate to **Control Design & Simulation » Simulation » Signal Arithmetic » Gain**, and place each block in the loop. Do the following for each Gain block:
  - o Right-click the Gain block and select **Visible Items » Label** from the shortcut menu.
  - o Right-click the Gain block and select **Configuration** from the shortcut menu. Set **Parameter Source** to “Terminal”.
- ☐ Make the following wiring connections in the Simulation Loop:
  - o Connect the **output** terminal of the Step Signal block to the **Operand1** terminal of the Sum 1 block.
  - o Connect the **Result** terminal of the Sum 1 block to the **Input** terminal of each of the three Gain blocks.
  - o Connect the `Proportional Gain (Kc)` control to the **gain** terminal of the Gain block.
  - o Connect the `Integral Gain (Ki)` control to the **gain** terminal of the Gain 2 block.
  - o Connect the `Derivative Gain (Kc)` control to the **gain** terminal of the Gain 3 block.
- ☐ The block diagram should look like this:





- ☐ Place two **Integrator** blocks inside the Simulation Loop. From the functions palette, navigate to **Control Design & Simulation » Simulation » Continuous Linear Systems » Integrator**, and place each block in the loop. Label the two blocks “Integrator 1” and “Integrator 2”
- ☐ Place a **Derivative** block inside the Simulation Loop. From the functions palette, navigate to **Control Design & Simulation » Simulation » Continuous Linear Systems » Derivative**, and place each block in the loop.

- ☐ Place another **Summation** block inside the Simulation Loop. Label the Summation block "Sum2". Double-click the Sum2 block to open the configuration dialog. Click the icons on the diagram so that the block has three "Add" input terminals. Click **OK**.
- ☐ Place a **Transfer Function** block inside the Simulation Loop. From the functions palette, navigate to **Control Design & Simulation » Simulation » Continuous Linear Systems » Transfer Function**, and place it in the loop. Double-click the Transfer Function block to open the configuration dialog. Set the **Parameter source** option to "Terminal". Click **OK**.
- ☐ Make the following wiring connections in the Simulation Loop:
  - o Connect the **output** terminal of the Gain 2 block to the **input** terminal of the Integrator 1 block.
  - o Connect the **output** terminal of the Gain 3 block to the **input** terminal of the Derivative block.
  - o Connect the **output** terminals of the Gain, Integrator 1, and Derivative blocks to the **Operand1**, **Operand2**, and **Operand3** terminals of the Sum 2 block.
  - o Connect the **Result** terminal of the Sum 2 block to the **input** terminal of the Transfer Function block.
  - o Connect the **plant** output of the MathScript Node to the **Transfer Function** terminal of the Transfer Function block.
  - o Connect the **output y(k)** terminal of the Transfer Function block to the **input** terminal of the Integrator 2 block.
  - o Connect the **output** terminal of the Integrator 2 block to the **Operand2** terminal of the Sum 1 block. This will close the feedback loop.
- ☐ The block diagram should look like this:



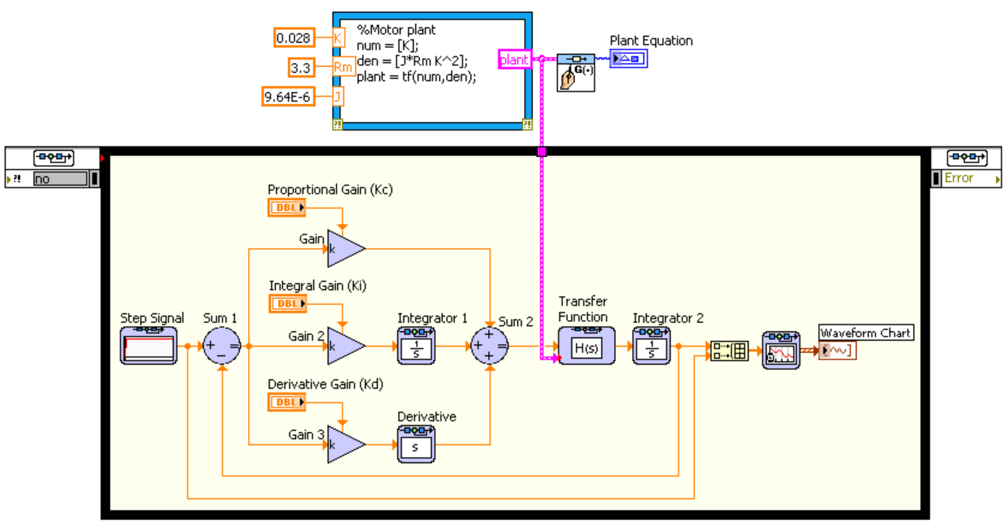
- ☐ Place a **Build Array** function inside the Simulation Loop. From the functions palette, navigate to **Programming » Array » Build Array**, and place it in the loop. Resize the Build Array node so that it has two inputs.

☐ Place a **SimTime Waveform** block inside the Simulation Loop. From the functions palette, navigate to **Control Design & Simulation » Simulation » Graph Utilities » SimTime Waveform**, and place it in the loop.

☐ Make the following wiring connections in the Simulation Loop:
  - Connect the **output** terminal of the Integrator 2 block to the first input terminal of the Build Array function.
  - Connect the **output** terminal of the Step Signal block to the second input terminal of the Build Array function.
  - Connect the **appended array** terminal of the Build Array function to the **Value** terminal of the SimTime Waveform block.

☐ Configure the simulation parameters of the Simulation Loop:
  - Right-click the border of the Simulation Loop, and select **Configure Simulation Parameters**.
  - Set the value of **Final Time (s)** to 1.
  - Set the value of **ODE Solver** to "Runge-Kutta 4".
  - Set the value of **Step Size (s)** to 0.001. This will set the simulation step size to one millisecond.
  - Click the **Timing Parameters** tab, and check the box labeled **"Synchronize Loop to Timing Source"**.
  - Set the value of **Period** to 1. This will set the loop to run once per millisecond (actual time).
  - Click **OK**.

☐ The block diagram should look like this:



☐ Optional: Configure the display options for the Waveform Chart.
  - Switch to the front panel of the VI (<Ctrl+E>).
  - Right-click the Waveform Chart, and select **Properties**.
  - In the **Display Format** tab, select "Floating Point" for the value of **Type**. Click **OK**.

- Double-click the rightmost numerical value on the x-axis of the Waveform Chart, and set the value to 1.

☐ Run the VI. Change the value of the PID Gains, and then run the VI again.

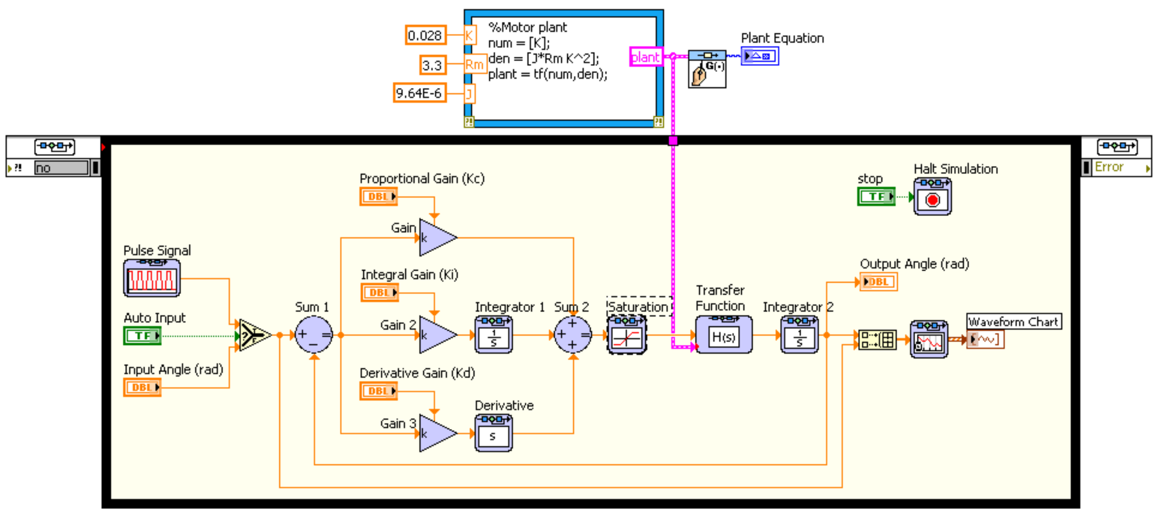
Exercise 2:

1. Build the VI block diagram
    - ☐ Open `Exercise_3.vi`. Save the VI as `Exercise_4.vi`.
    - ☐ Place a **Boolean** button on the front panel. From the controls palette, navigate to **Modern » Boolean » Push Button**, and place it on the front panel. Label the button "Auto Input".
    - ☐ Place a **Dial** on the front panel. From the controls palette, navigate to **Modern » Numeric » Dial**, and place it on the front panel. Label the dial "Input Angle (rad)".
    - ☐ Set the minimum and maximum values of the scale of the "Input Angle (rad)" dial to -3.14 and 3.14, respectively.
    - ☐ Make a copy of the "Input Angle (rad)" dial (<Ctrl+Drag>). Label the new dial "Output Angle (rad)".
    - ☐ Change the "Output Angle (rad)" dial from a control to an indicator. Right-click the dial and select **Change to Indicator** from the shortcut menu.
    - ☐ Place a **Stop Button** on the front panel. From the controls palette, navigate to **Modern » Boolean » Stop Button**, and place it on the front panel. This will allow you to terminate the simulation at will.
    - ☐ Take the following steps to customize the Waveform Chart display options:
        - o Right-click the chart and select **Y Scale » Autoscale Y** from the shortcut menu. Verify that the Autoscale Y option is no longer checked.
        - o Set the minimum and maximum values of the Y Axis to -4 and 4, respectively.
        - o Set the maximum value of the X Axis to 2.
    - ☐ Switch to the block diagram of the VI (<Ctrl+E>).
    - ☐ Right-click the border of the Simulation Loop, and select **Configure Simulation Parameters** from the shortcut menu. Set the value of **Final time (s)** to "inf". This will set the Simulation Loop to run until it is manually terminated. Click **OK**.
    - ☐ Make sure that the "Auto Input", "Input Angle (rad)", "Output Angle (rad)", and "stop" control terminals are all inside the Simulation Loop.
    - ☐ Right-click the **Step Signal** block, and select **Replace » Programming » Comparison » Select** from the shortcut menu. This replaces the Step Signal block with a **Select** function.
    - ☐ Place a **Pulse Signal** block inside the Simulation Loop. From the functions palette, navigate to **Control Design & Simulation » Simulation » Signal Generation » Pulse Signal**, and place it in the loop.
    - ☐ Double-click the Pulse Signal block to open the configuration dialog. Set the value of **amplitude** to 3.14. Click **OK**.
    - ☐ Place a **Halt Simulation** block inside the Simulation Loop. From the functions palette, navigate to **Control Design & Simulation » Simulation » Utilities » Halt Simulation**, and place it in the loop.
    - ☐ Make the following wiring connections in the Simulation Loop:

- Wire the **output** terminal of the Pulse Signal block to the **t** (top-left) terminal of the Select function.
- Wire the “Input Angle (rad)” control terminal to the **f** (bottom-left) terminal of the Select function.
- Wire the “Auto Input” control terminal to the **s** (center-left) terminal of the Select function.
- Wire the **output** terminal of the Integrator 2 block to the ”Output Angle (rad)” indicator terminal.
- Wire the “stop” control terminal to the **Halt?** terminal of the Halt Simulation block.

☐ Insert a **Saturation** block at the PID output. Right-click the wire connecting the **Result** terminal of the Sum 2 block to the **input u(k)** terminal of the Transfer Function block. Select **Insert » All Palettes » Control Design & Simulation » Simulation » Nonlinear » Saturation**. Double-click the Saturation block to open the configuration dialog. Set the values of **lower limit** and **upper limit** to -24 and 24, respectively (These values represent the maximum voltage inputs of the DC motor). Click **OK**.

☐ The block diagram should look like this:



☐ Save the VI (<Ctrl+S>).

2. Run the VI

☐ Switch to the front panel of the VI (<Ctrl+E>).

☐ Run the VI (<Ctrl+R>). Click the “Auto Input” button, and observe the output signal follow a pulse signal.

☐ Turn off the “Auto Input” button, and change the value of “Input Angle (rad)”. The output signal should respond.

☐ Change the values of the PID gains, and observe how the output signal responds to the input with different gain values.

☐ Press the “Stop” button to end the simulation.