**Hands-on Lab 3:**

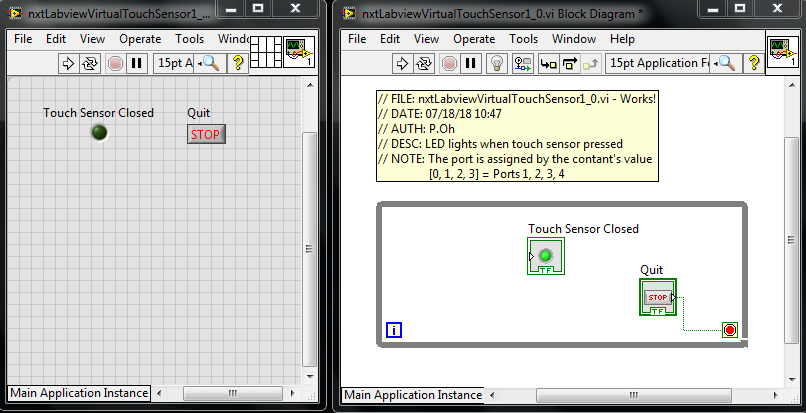
**LabVIEW – Virtual Touch Sensor and Voltmeter**

Previously LabVIEW was shown to generate voltages. By connecting the NXT Brick’s motor output ports, one could generate voltages. Here, the reverse is also possible. One connects to the Brick’s sensor ports. This lab demonstrates that the LEGO NXT can detect events (like a pressed button) and voltages.

# **Concept 1:** **Virtual Touch Sensor** (Analog Voltage Output)

**Step 1:** Create Front Panel

Open LabVIEW, File – Save All with nxtLabviewVirtualTouchSensor1.0.vi. In your front panel, add an LED and STOP button as seen in **Figure 1A (left)**. Create the associated block diagram: add a while-loop structure and wire the STOP button to terminate the loop. It’s also good practice to comment your code as seen in **Figure 1A (right)**.

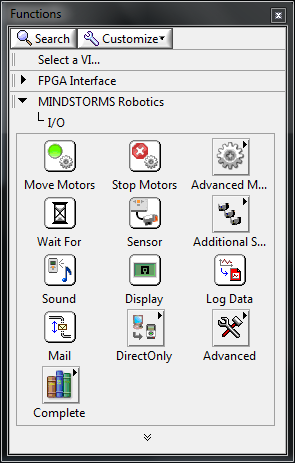
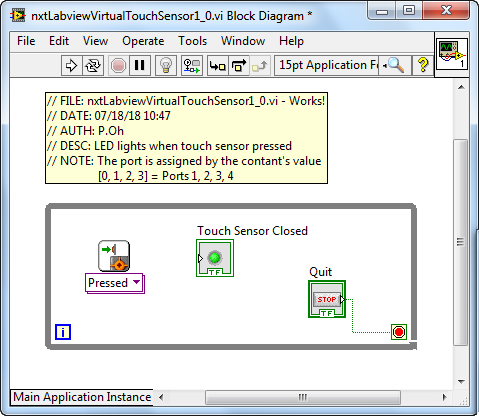


**Figure 1A:** LED and STOP Boolean controls in Front panel (left). While-loop structure in block diagram

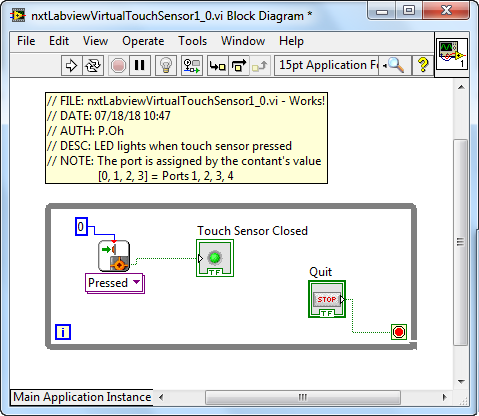
**Step 2:** Add Touch Sensor to Block Diagram

In the block diagram, go to the LEGO sensors control window as shown in **Figure 2A (left)**. Recall, this is done by selecting View – Functions Palette – MINDSTORMS Robotics – I/O. Click and drag the Sensor control into the block diagram as shown in **Figure 2A (right)**.

**Figure 2A:** MINDSTORMS Robotics – I/O dialog box shows the Sensor control (left). Click and drag the Sensor control into the block diagram (left).



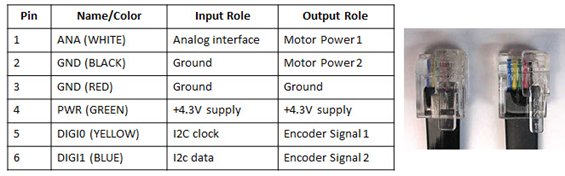
Next, add a Numeric Constant to the block diagram. Recall, this is done by from the Functions palette where one selects MINDSTORMS Robotics – Programming – Numeric. Wire this numeric constant to the Sensor control’s port input. One can view a control’s terminals by selecting your wire tool and hovering over areas of the control. By default this numeric control’s value is 0. Lastly, wire the Sensor control’s Yes/No output port to the LED control’s input. Your block diagram should look like **Figure 2B**.



**Figure 2B:** Final Block Diagram – a numeric constant (default value is 0) is added and wired to the Touch sensor block.

**Step 3:** Breadboard the Touch Sensor

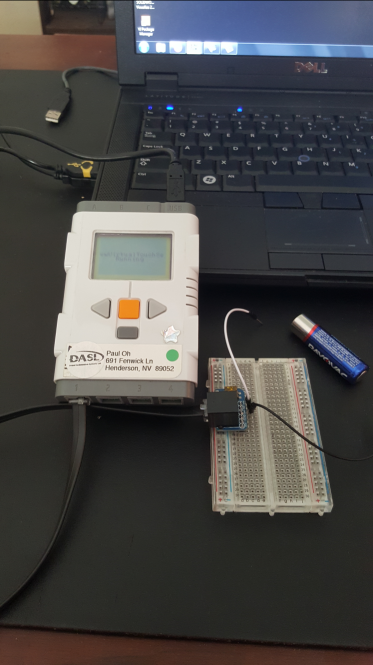
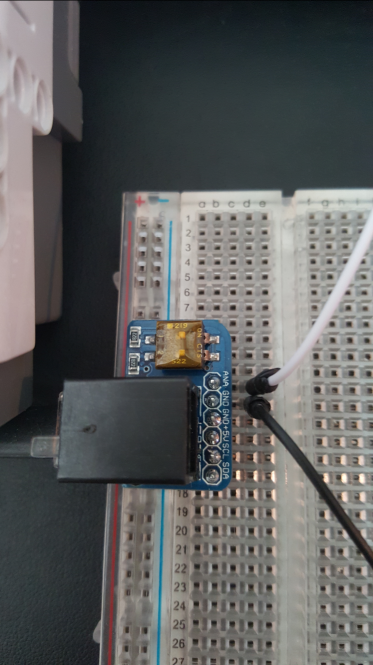
Recall in the previous lab, an NXT breadboard adapter facilitates access to ports (see figure below for reference).



A standard NXT cable consists of 6 colored wires. When this cable is plugged into the brick’s output port, and then each wire has the function given in the right-most column.

Plug a NXT cable to (sensor) Port 1 (see **Figure 3 left**). Next, add hookup wires to the NXT breadboard adapter’s ANA and GND pins as in **Figure 3 right**.

**Figure 1N:** Breadboard connections (left) and NXT adapter close up (right)



From the Front Panel, run your program. When the hookup wires are shorted then the LED should glow green. When the wires are disconnected, then the LED should stop glowing. Click the STOP button to terminate. Essentially, you’ve created a virtual touch sensor.

**Exercise 1:** In LabVIEW create programs for the following:

* 1. Connect the NXT cable to Port 2.Change nxtLabview-virtualTouchSensor1\_0.vi so that the LED glows green when the ANA and GND wires are shorted
  2. Write a new program which displays a count of how many times the ANA and GND wires are shorted

**Concept 2: Virtual Voltmeter** (analog voltage input)

**Step 1:** Create the front panel seen in **Figure 1A left**

The Meter control can be found from View – Controls Palette – MINDSTORMS Robotics – Numeric. The Meter control resembles an analog display. Right click on the meter and select Visible – Digital Display. This will allow one to view the meter’s numeric value. Next add a numeric indicator control and label it as “Digital Voltmeter”. This will just provide another display of any voltages read by the NXT.

Figure 3: DAQ Assistant configuration screen for a virtual voltmeter

**Step 2:** Create the Block Diagram

Create a while-loop around the controls in the block diagram (**Figure 1A right**) and wire the STOP button to terminate the loop.

Next, from MINSTORMS Robotics – I/O – Advanced, find the Setup Sensor and Read Generic Sensor controls (**Figure 2A**). Click and drag the Setup Sensor control to the left of the while loop. Click and drag the Generic Sensor control into the while loop. These two controls are used to configure and initialize the sensor. Wire these two controls via their NXT/EV3 terminals as seen in **Figure 2B**.

Figure 0A: Virtual voltmeter front panel

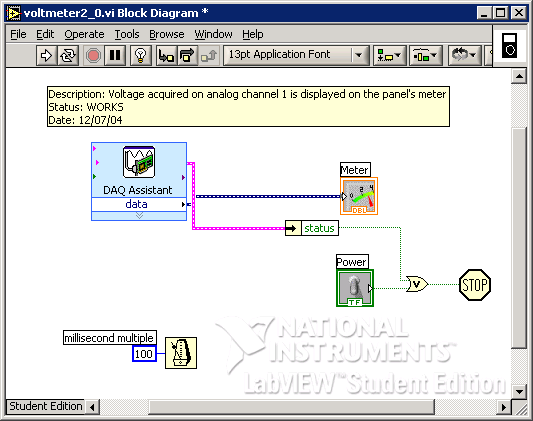


Figure 2B: Virtual voltmeter block diagram

Next, create 2 numeric constant controls. Wire each constant control to the Port terminal of the Setup Sensor and Read Generic Sensor controls (see **Figure 2C**).

To configure the sensor hover your mouse of the Setup Sensor control. Right click and select Create – Constant. By default, the constant has 2 components: NO\_SENSOR and RAWMODE. Hover your mouse over the NO\_SENSOR, right click and choose Properties. A pop up box will display. Select the Edit Items tab. Here, one sees various possible sensor types. Click SWITCH to highlight it, click the Move Up button and then OK. One will observe that the Setup Sensor control now displays SWITCH and RAWMODE components.

The NXT’s input Ports contain a 10-bit analog-to-digital converter (ADC). Thus, when the port receives a voltage input, it is converted into a digital value from 0 to 1023. Furthermore, each input port is configured as a voltage divider with a 5 Volt source. Thus, when a voltage enters the port, it is scaled by 5 Volts. Add a Multiply and Divide controls to your block diagram and wire with numeric constants as seen in Figure XX.

In the lower left corner of the block diagram is a timer control. Create this as follows: Right Click – All Functions – Time & Dialog – “Wait Until Next ms Multiple”. The constant “100” sets the block to execute every 100 milliseconds.

**Step 2:** Insert the DAQ Assistant control (Right mouse click – Input – DAQ Assist). Double-clicking the DAQ Assistant control in the block diagram will invoke the configuration screen. Choose Analog Input and select Voltage. This will invoke the screen in Figure 3.

Under the Terminal Configuration pull-down menu, select “Differential”. The minimum and maximum voltages should be 0 and +10 Volts respectively. Again, the NI-DAQ PCI-6025E card listed under “My Physical Channels”. Choose a1 (analog channel 1).

Choose the “Acquire Continuously” option with Samples to Read set at 100 and Rate of 1000 Hz. The clock rate should read Internal.

Click OK to complete the configuration.

Figure 3: DAQ Assistant configuration screen for a virtual voltmeter

**Step 3:** Connect a real adjustable power supply’s positive lead (red) to screw terminal screw terminal 5 (AI 1) and the negative lead (black) to screw terminal 6 (AI 9). Run the VI and observe the meter’s needle display the voltage.

**Exercise 2:**

* 1. On one computer run the virtual power supply (from Concept 1). Wire the analog output voltage to your computer that runs the virtual voltmeter.

**Concept 3:** **Virtual Data Recorder** (File input/output)

The data acquired can be stored to a hard disk file. As such, the virtual voltmeter in Concept 2 can be adapted into a virtual data recorder.

**Step 1:** A very simple front panel is needed because all acquired data will be recorded into a file on your hard drive. This file will be read and plotted with Excel. Create the front panel shown in Figure 4A. Use a while-loop control to create the block diagram (Figure 4B).

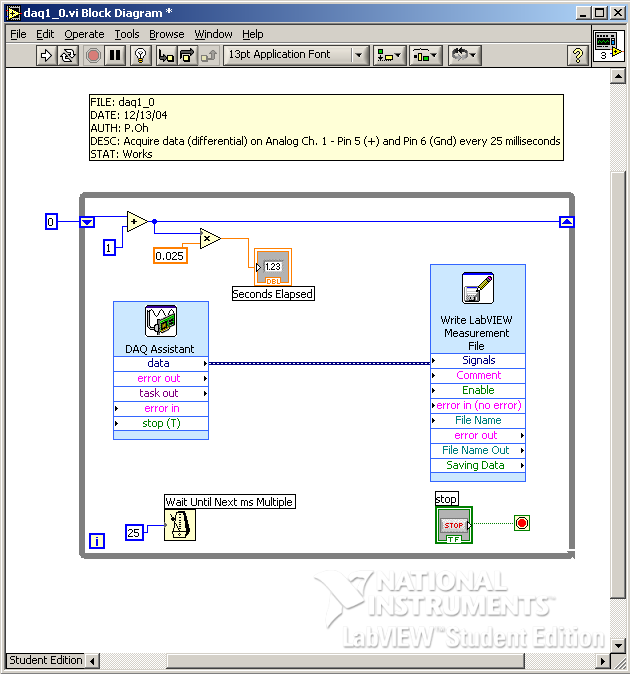


Figure 4B: Data Recorder block diagram

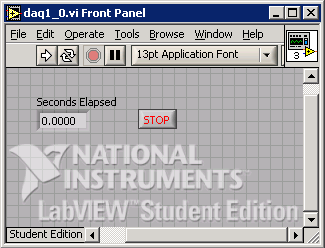


Figure 4A: Data Recorder front panel

The constant “25” associated with the timer sets the while-loop to execute every 25 milliseconds. A shift register variable that increments by 1 every iteration, is multiplied by 0.025 and is displayed in the front panel indicator labeled “Seconds Elapsed”.

**Step 2:** Insert the DAQ Assistant control (Right mouse click – Input – DAQ Assist). Double-clicking the DAQ Assistant control in the block diagram will invoke the configuration screen. This time choose Analog Input and select Voltage. This will invoke the screen in Figure 4. Under the Terminal Configuration pull-down menu, select “Differential”. The minimum and maximum voltages should be –10 and +10 Volts respectively. Again, one should see the NI-DAQ PCI-6025E card listed under “My Physical Channels”. Choose a1 (analog channel 1). Click OK to complete the configuration.

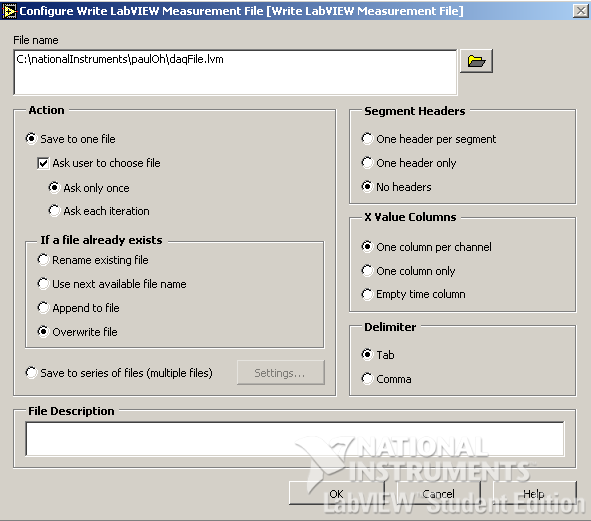


Figure 5: LabVIEW Measurement file (LVM) screen

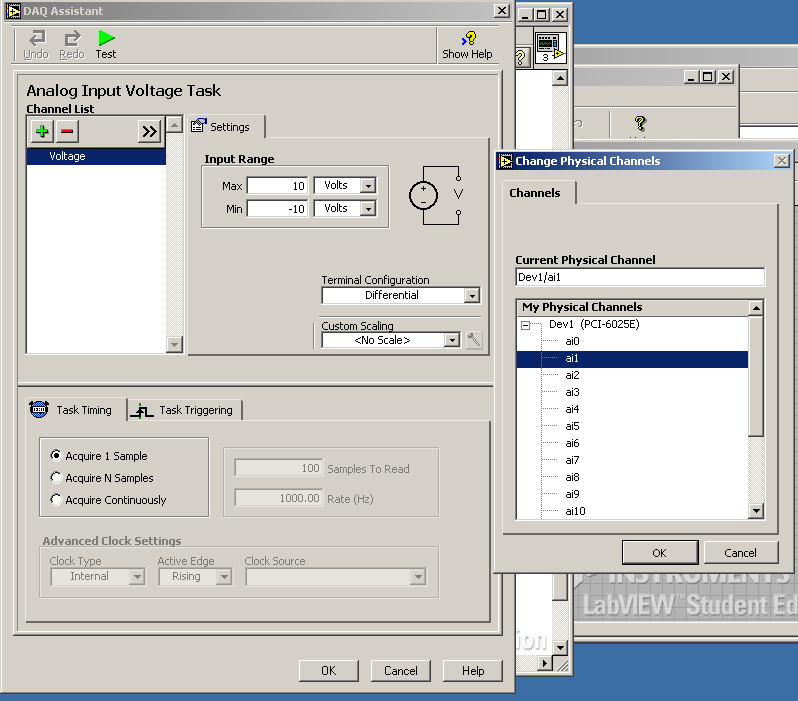


Figure 4: Choose a1 under My Physical Channels

**Step 3:** Acquired data is saved to a file on the hard disk. In the block diagram: Right-click – Output – Write LVM. Set your configuration screen to match the one given in Figure 5.

Chose a directory and file name (e.g. daqFile.lvm) to save acquired data. Choose the “Overwrite File” option and make sure the “Tab” is the delimiter.

**Step 4:** Attach a sinusoidal function generator to screw terminals 5 (AI 1) and 6 (AI 9). Set the function generator at 200 Hz with a +5 Volt amplitude. Run your program for about 5 seconds.

The file daqFile.lvm should be in the directory you choose in Step 3. Open the data file with Excel. There are two columns, the leftmost is the elapsed time at every 0.025 seconds and the rightmost is the acquired voltage value. Plot the data to see the sine wave.

**Exercise 3:**

* 1. What are the differences in the timer configurations used in Concepts 2 and 3?
  2. Eyeball the period and frequency of the Excel data plot. What are there values?
  3. Set the frequency to 80 Hz, acquire and plot the data. Explain the results.
  4. Set the frequency to 60 Hz, acquire and plot the data. Explain the results.
  5. In LabVIEW write a program that can correctly read a signal with a 40 Hz period