PHYC 500: Introduction to LabView

M.P. Hasselbeck, University of New Mexico

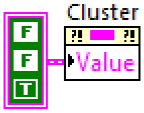
Exercise 16 (v 1.2) Controlling hardware with DAQ device

This exercise has two parts. First, simulate a traffic light circuit using a State Machine. Second, implement the traffic light with a DAQ device to control red, yellow, and green LED hardware.

Simulating a traffic light

Select a State Machine from the LabView templates and configure the Enum Type Def for three cases: Red, Yellow, and Green (R-Y-G). Download the custom traffic light indicator (light.ctl) from the class website and place it on the Front Panel of your VI (right click on the Front Panel to select a control). On the Block Diagram, this indicator takes a cluster input of three Booleans corresponding to the state (TRUE or FALSE) of the red, yellow, and green LEDs. This indicator is a Strict Type Def with a custom decoration depicting the traffic light structure. Only one light can be in the TRUE state – the other two must be FALSE.

There are several options for implementing this State Machine, but here the indicator is implemented *inside* the individual states. Create a Property Node for each state; the Green state is configured using the cluster constant (see previous exercise):



Insert Wait constants to define how long each light is illuminated (eg. 5s, 2s, and 5s for R-Y-G). The Boolean indicators will be wired to corresponding colored LEDs on an electronics breadboard via the USB-1208LS data acquisition device manufactured by Measurement Computing.

Parts needed

Measurement Computing USB-1208LS DAQ and interface cable
Circuit breadboard
Red, yellow, and green light-emitting diodes
3 resistors having values between 100—300 Ω (not critical)
Hookup wires and small blade screwdriver

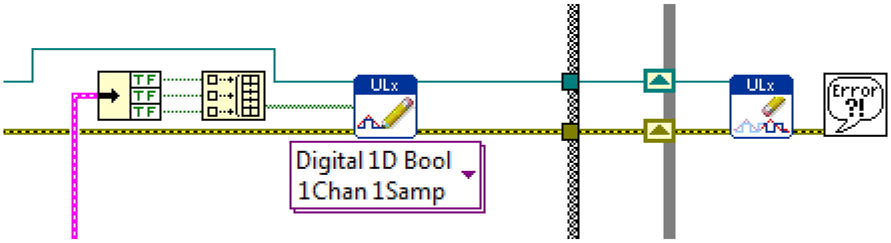
The USB-1208 is a lower performance, low cost alternative to NI DAQ hardware. It is not recognized by the NI-MAX program and must be setup with drivers and interfacing VIs supplied by Measurement Computing. This software should be already installed on the lab computers.

Configure the USB-1208LS

Close LabView. Plug the 1208LS into a computer USB port. Open the program Measurement Computing: InstaCal. The device should be immediately recognized. If not, right-click on the PC Board List icon and select Refresh. You can test basic communication with InstaCal by clicking on Configure and instructing the device to flash its green LED.

Modifying LabView VI for ULx

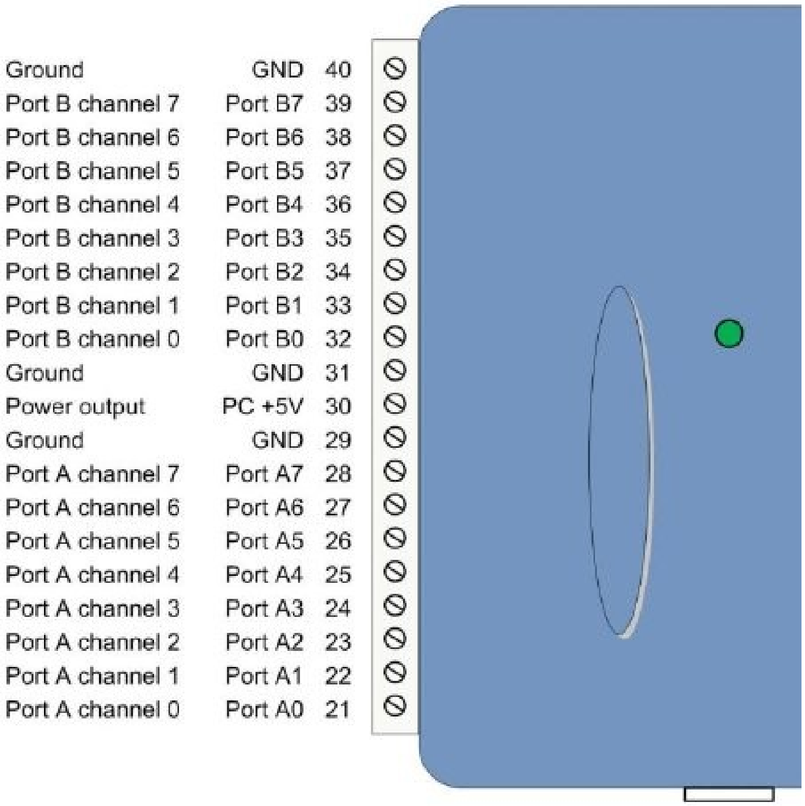
Go to the Block Diagram. The 1208LS can be configured in a variety of ways. The approach used here will be to place all three Boolean indicators in a 1-D array. Unbundle the cluster input wire and build the Boolean array. You can find these functions easily using Quick Drop. Click on the icon bottom edge of the Build Array icon and drag it down to provide 3 inputs. Refer to the two left icons in following Block Diagram.



Digital 1D Bool
1Chan 1Samp

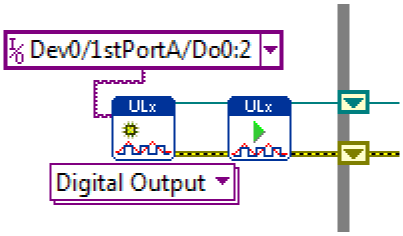
Connect wires from the individual Boolean terminals to the open inputs. Each element in the Boolean array will be assigned a unique digital output line on the 1208LS. Do not delete the Boolean indicators from the front panel as they will aid in troubleshooting. Leave the Build Array output unwired for now.

Right-click on an open space in the block diagram, locate User Libraries: ULx for NI LabView. Use the thumbtack in the upper left corner of the window to hold it open. It is also recommended that the Context Help (CTRL-H) of LabView be enabled to assist in wiring. Select the ULx Create Channel VI and place it on the left-outside of the While Loop that forms your State Machine. On the menu immediately below the icon, select Digital Output. Find the **lines** input connection on the left side of the icon; right-click and create a constant. Clicking on the blank constant should display a list of available digital output lines. Example: Dev0/1stPortA/Do0 configures only Port A0 (terminal connection 21 on the 1208LS device) as a digital output channel. Here we are interested in a three channel output, i.e. more than one line. Pins 21, 22, and 23 can be enabled as digital outputs by manually modifying the input string constant to read as follows: Dev0/1stPortA/Do0:2. This creates a ULx task interface constant for the corresponding three digital output terminals. The first array element could be wired to Do0 (Pin 21), the second element to Do1 (Pin 22), and the third to Do2 (Pin 23). The terminals available for digital I/O on the 1208LS are shown in the following diagram.



Ground	GND	40
Port B channel 7	Port B7	39
Port B channel 6	Port B6	38
Port B channel 5	Port B5	37
Port B channel 4	Port B4	36
Port B channel 3	Port B3	35
Port B channel 2	Port B2	34
Port B channel 1	Port B1	33
Port B channel 0	Port B0	32
Ground	GND	31
Power output	PC +5V	30
Ground	GND	29
Port A channel 7	Port A7	28
Port A channel 6	Port A6	27
Port A channel 5	Port A5	26
Port A channel 4	Port A4	25
Port A channel 3	Port A3	24
Port A channel 2	Port A2	23
Port A channel 1	Port A1	22
Port A channel 0	Port A0	21

LabView must be instructed to start the ULx interface you have just configured. To do this, go to the ULx palette and select the ULx Start Task VI. Place it immediately to the right of the Create Channel VI, but outside the While Loop. Connect the Task-Out and Task-In lines as well as the Error-Out and Error-In lines of the adjacent VIs. Create two Shift Registers on the While Loop. The task and error lines on the output of the Start Task VI will eventually get wired inside the State Machine and you will see something that looks like this:



Dev0/1stPortA/Do0:2

Digital Output

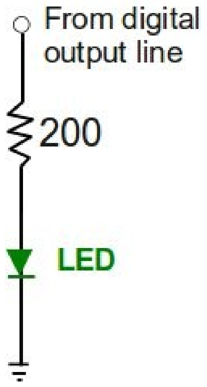
For now, leave the outputs unwired. When communicating with external hardware as done here, it is good programming practice to use the Error Handling line to assist in diagnosing problems.

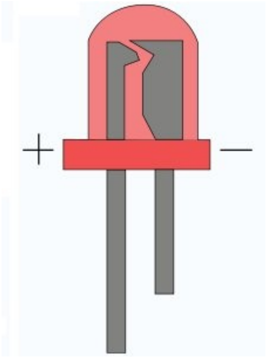
In the open palette, select the ULx Write VI. Place it inside the While Loop near the output of the Build Array icon. Configure it as Digital: Single Channel: Single Sample: 1-D Boolean (N lines). Now make the connections from the ULx Start Task VI output terminals (task out, error out) to the corresponding input connections on the Write VI. Route these wires around the Case Structure in the State Machine so they are not hidden. The Build Array output should appear as a thick green line (indicating a 1-D Boolean array) that is connected to the **data** input terminal of the ULx Write VI. Refer to the first figure of this exercise.

The ULx interface should be cleared after it executes. This is accomplished with the ULx Clear Task VI on the palette. Place it outside the While Loop and connect the task and error lines as shown above. If an error occurs, diagnostic information will appear on the user interface by connecting the error line to Dialog & User Interface: Simple Error Handler as shown.

Light-Emitting Diodes

Arrange R-Y-G LEDs in the pattern of a traffic light on the circuit breadboard. The current in each diode should be limited with a 100--300 Ω series resistor as shown in the diagram. Unlike a resistor, a diode cannot be placed in a circuit arbitrarily. The anode (+ terminal) should be connected to the resistor and the cathode (– terminal) to ground. Setup a common ground on the breadboard and use a jumper wire to connect it to terminal 29 on the 1208LS. Use three more jumper wires to connect the appropriate digital output lines. Do not operate any LED without a current limiting resistor.





You will need to follow the color connection scheme used to build the 1-D Boolean array inside your VI. Start the VI and verify proper LED operation. Demonstrate for the instructor.

Troubleshooting: If one or more LEDs fail to operate, check the polarity by referring to the above diagram. Verify proper terminal connections on the 1208LS.