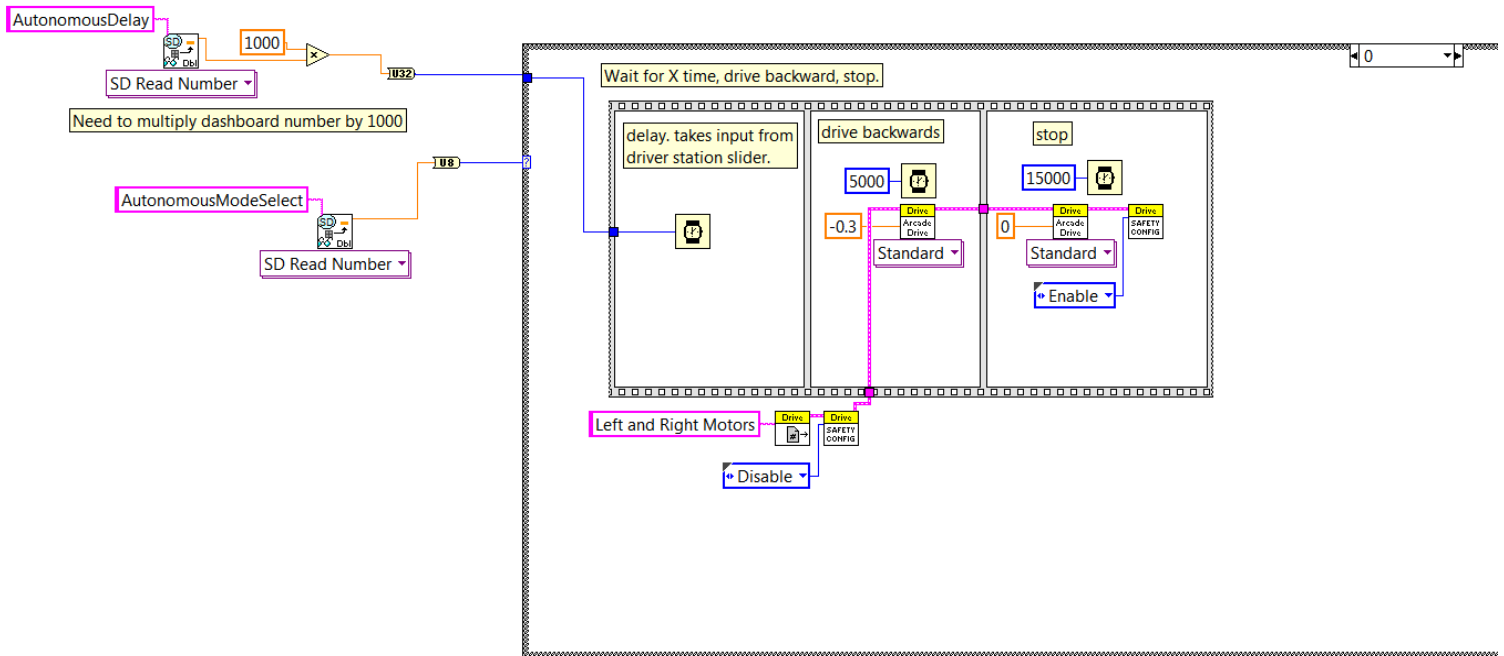


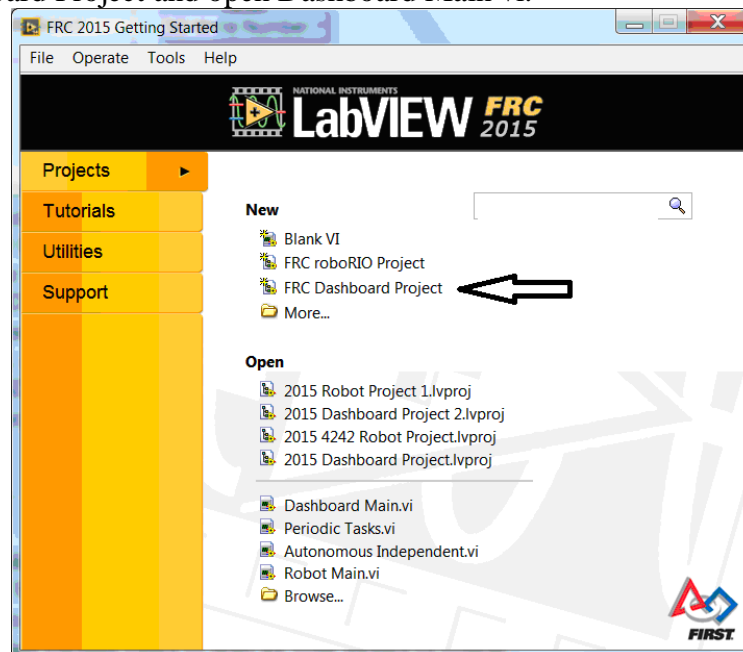
Part 1: Setting Up the Autonomous Code

1. In Autonomous Independent, create a case structure in which the autonomous modes will be placed (Programming > Structures > Case Structure).
2. Insert a SD Read Value vi (WPI Robotics Library > Dashboard > SD Read Value.vi). Change the SD Read Value type to SD Read Number by selecting it from the drop-down menu below the vi.
3. Wire the SD Read Number vi to the condition of the case structure. You may have to convert the output to an unsigned integer (Mathematics > Numeric > Conversion > To Unsigned Byte Integer.vi).
4. Right-click on the Name terminal on the SD Read Number and create a constant. Take note of what you name the SD Read Number. I named mine "AutonomousModeSelect".
5. Add as many cases to the case structure as you need.

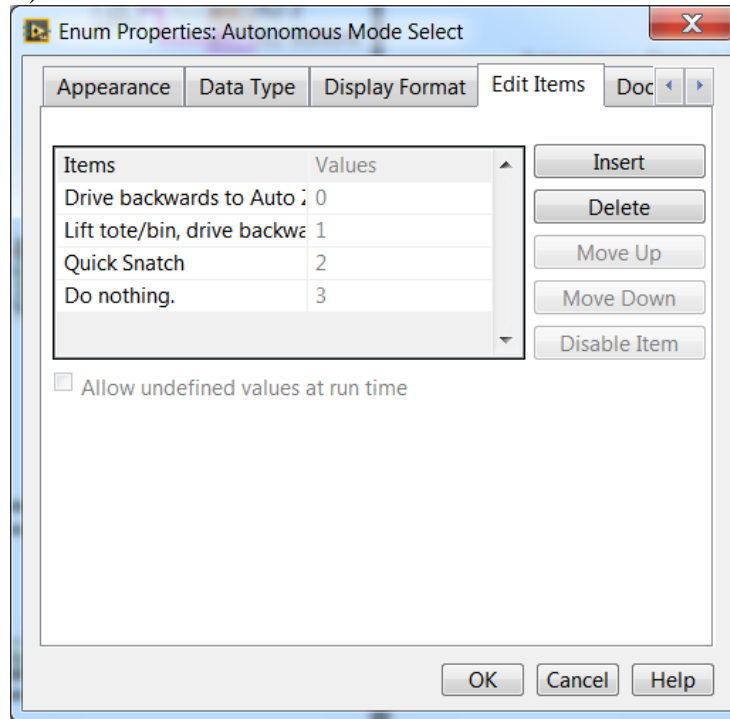


Part 2 Creating the Selection Controls in the Dashboard

1. Start a new FRC Dashboard Project and open Dashboard Main vi.



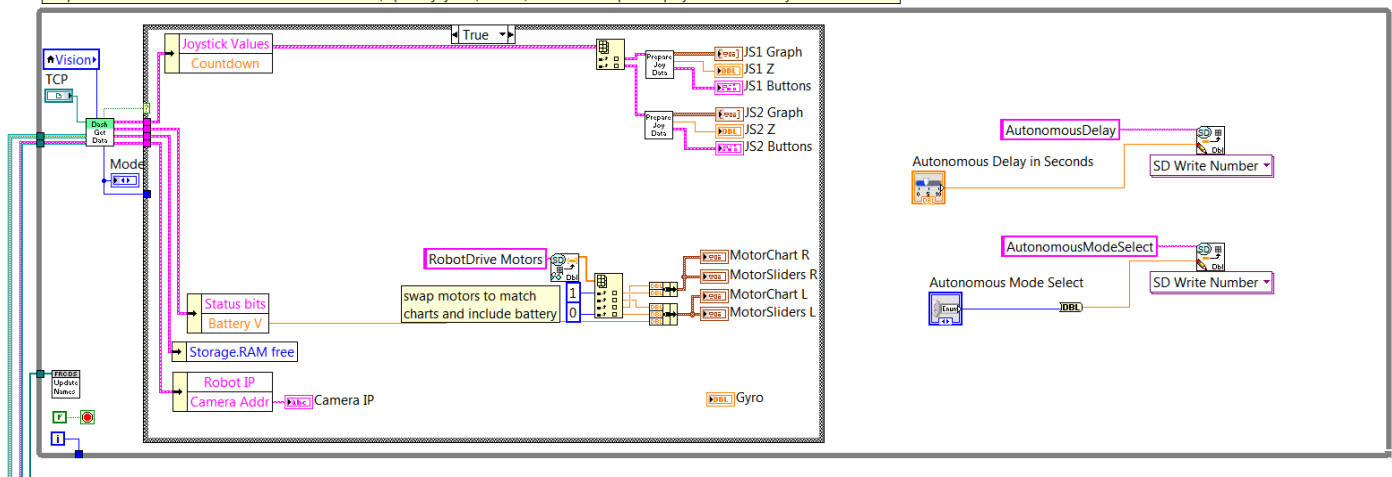
2. In the Custom tab on the Front Panel, insert an Enum vi (Modern > Ring & Enum > Enum.vi). The name here is just text on the Front Panel. I call mine “Autonomous Mode Select” but it is just for show and is not a variable name.
3. Press ctrl+E to access the block diagram. Expand Loop 1 to give you room to work. Next, find the Enum vi you added to the front panel. From what I can tell, it seems to place it randomly within the diagram. Once you’ve located it, drag it and only it into the expanded space within Loop 1.
4. Right-click on the Enum vi and change it to a control. This may have already been done automatically.
5. Right-click on the Enum vi, select Properties, and click on the Edit Items tab. Insert as many Items as you have cases in your autonomous code. Items here are simply the text that will show up in your autonomous selection drop down menu, you can name them whatever you want. The Values correspond to the cases in your autonomous code (0, 1, 2...). Click Ok when finished.



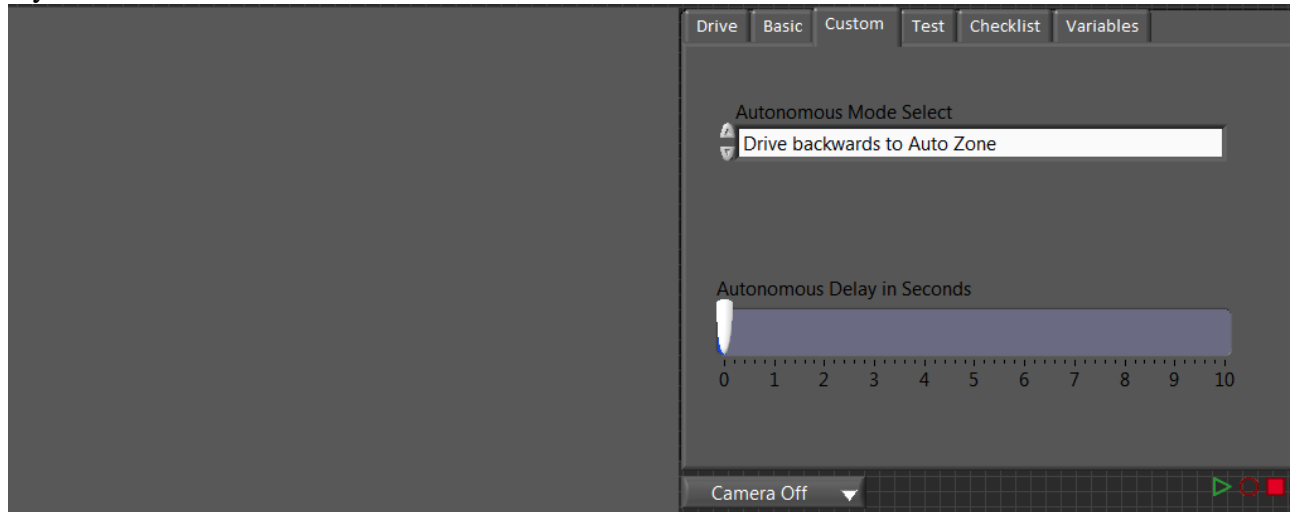
6. Insert a SD Write Value vi (WPI Robotics Library > Dashboard > SD Write Value.vi). Change the SD Write Value type to SD Write Number by selecting it from the drop-down menu below the vi.
7. Wire the output terminal of the Enum vi to the Numeric terminal of the SD Write Number vi. You may have to convert the output to a double precision float (Mathematics > Numeric > Conversion > To Double Precision Float.vi).
8. Right-click on the Name terminal on the SD Write Number and create a constant. Name it the same as the SD Read Number in your autonomous code. “AutonomousModeSelect” in my example.

If the display needs to process the data before display, loop one is an example of doing that for the joysticks and motors.

Loop 1. Receive various data streams from Driver Station, update joystick, motors, and more complex displays that don't easily bind to variables.



9. Finally, go back to the Dashboard Front Panel and arrange and size things as you want. You should see the Items you added to the Enum control.



10. Save everything.

Part 3 – Building the executable and telling the Driver Station to launch your custom Dashboard

From the LabVIEW tutorials:

After you customize the FRC Dashboard Project, you must rebuild the project so that the Driver Station launches the most recent version of the project.

Complete the following steps to rebuild the project and specify which project the Driver Station launches automatically.

1. In the **Project Explorer** window of the Dashboard Project, expand the list of Build Specifications to reveal the FRC_Dashboard application.
2. Right-click **FRC_Dashboard** and select **Properties** from the shortcut menu. If the **Application Builder Information** dialog box appears, click **OK**.
3. Verify that the **Target filename** is Dashboard.exe. The Driver Station reads this executable file as the file to launch.
4. Verify that the **Destination directory** is where you want to save the file. Click **Build** to build the new project.
5. Explore to where you saved the new executable file. Copy the new Dashboard.exe file over to the C:\Program Files\FRC Dashboard directory.

Note: In my example code above, you'll also notice an Autonomous Delay. This setup passes a value from the Dashboard to the autonomous before the match starts. I added a Horizontal Pointer Slider (Modern > Numeric > Horizontal Pointer Slider) to the Dashboard Front Panel and configured it to increment in 0.25 intervals (right-click > Properties > Data Entry, 0.25, Coerce to Nearest). Everything else is setup almost the same way as the Autonomous Mode Select. The only exceptions are 1) you don't need to convert to a double when wiring the Horizontal Pointer Slider to the SD Write Number in the Dashboard code and 2) you need to convert from double to an Unsigned Long Integer when wiring from the SD Read Number to the Wait vi in your autonomous code.