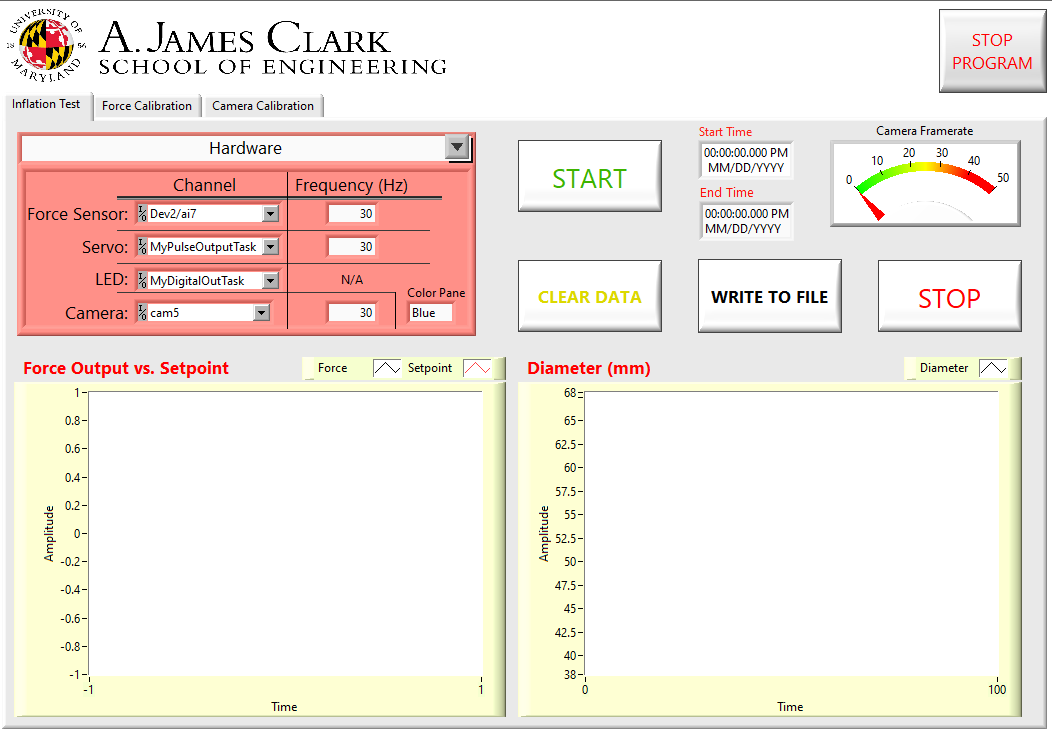
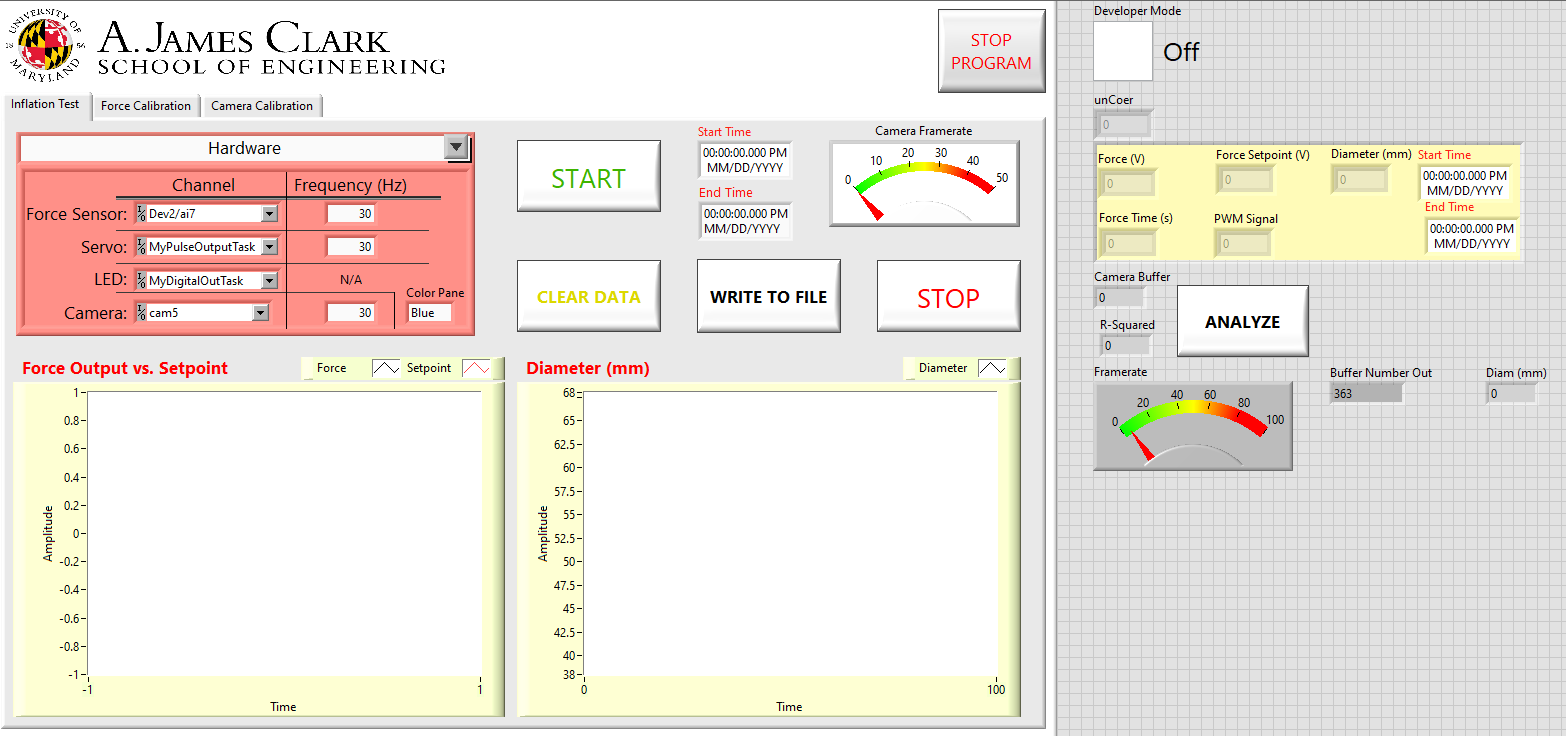
Inflation Apparatus

Running the Code

1. To open the code, first open the LabVIEW project “InflationApparatus.lvproj” located in the top level of the “Inflation Apparatus” folder. Always open the project first to ensure the correct subVI’s are referenced in the main VI.
2. In the project, open the top level VI “InflationApparatus.vi” The following front panel will open:



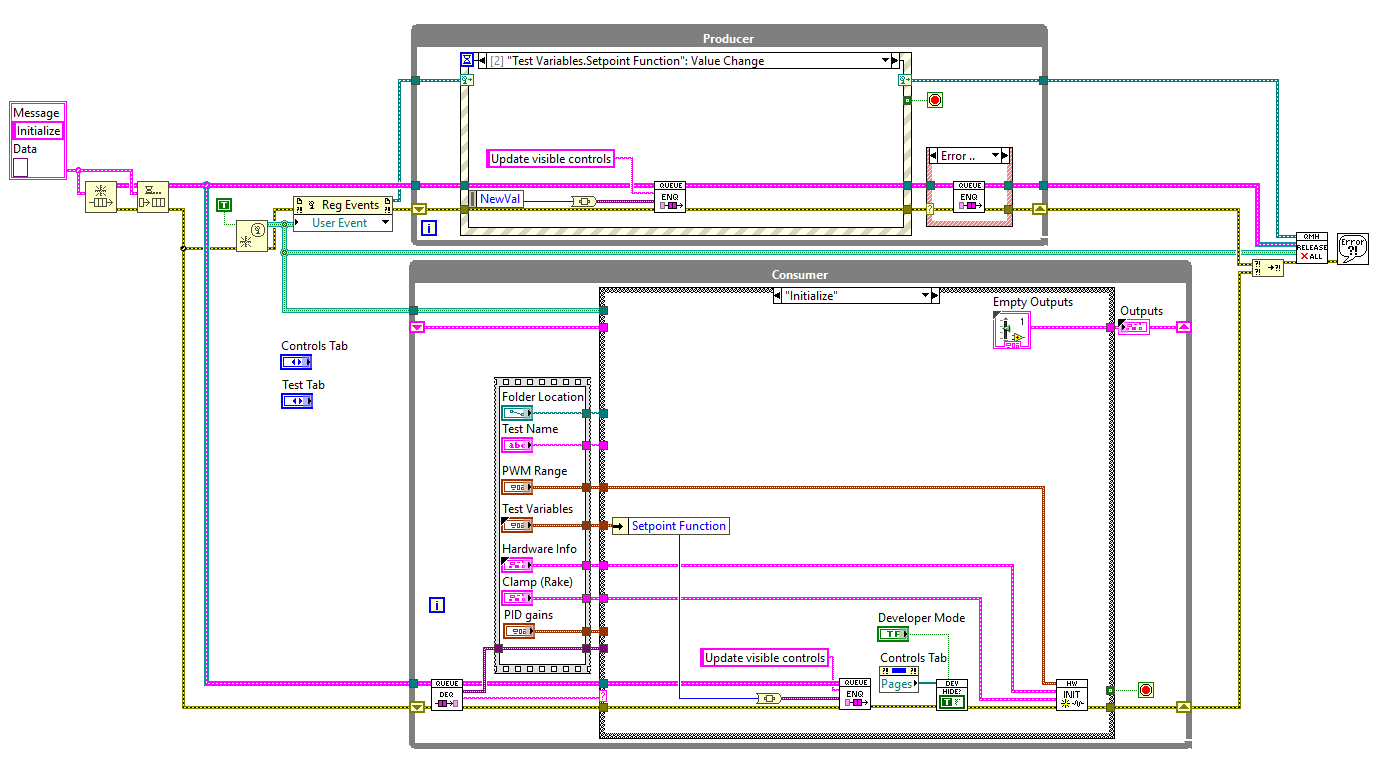
1. Pressing the run arrow on the taskbar will no longer start a test. To start a test, press the run button while the VI is running. During a test, the stop button can be pressed to stop the test. Pressing the stop button will not abort the program.
2. To abort the program, the Stop Program button in the top right can be pressed. Do not abort the program with the abort button (red stop sign) in the taskbar unless it’s absolutely necessary. Aborting in this way prevents hardware from shutting down correctly.
3. Test controls can be edited in the red tab selector located above the force v. setpoint graph. There are options for Hardware, Test Variables, Report Generation, and Developer options.
4. Developer options will only be visible when developer mode is on. The developer mode selector is located to the right of the front panel and is only accessible with LabVIEW development systems:



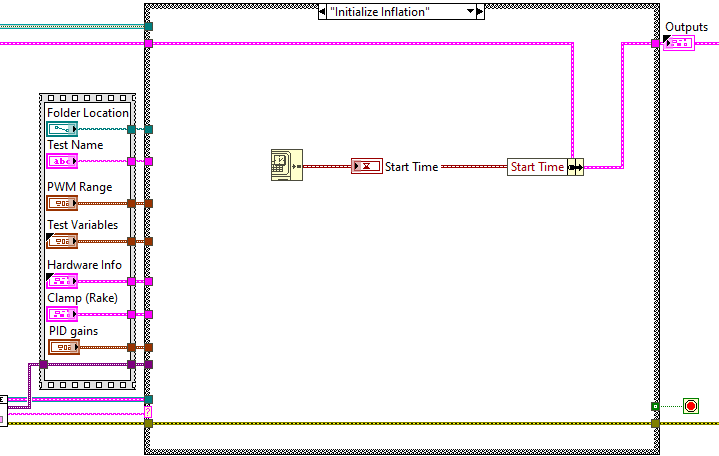
1. Charts can be cleared by pressing the clear data button
2. Data files can be created by pressing the write to file button. The file generated will be saved to the specified folder. The file name will contain the specified name followed by the date. File names will automatically alphabetically increment.
3. Force/camera calibration tabs are currently in place; however, they are not in a runnable state.

Block Diagram

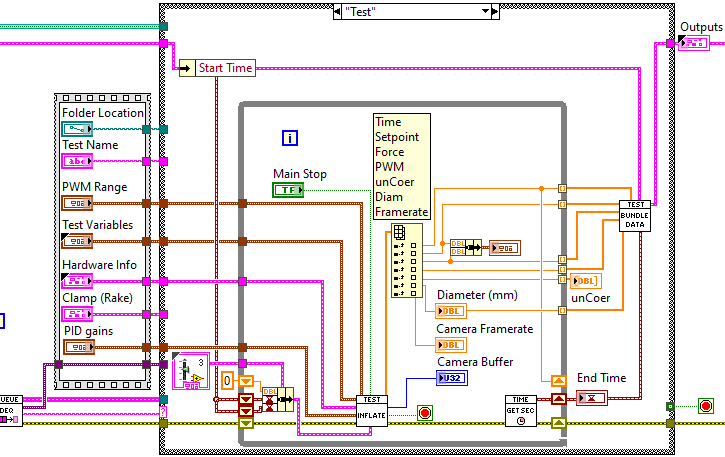
1. The code uses a parallel processing structure to deal with user events (ie. button presses) and code processes. The structure is a producer-consumer based approach where the top loop handles user events and queues states for the bottom loop to run. Ex) When the run button is pressed, the top loop queues the “Main Start” enum. When this is queued the “dequeue element” VI in the bottom loop reads the queued enum as “Main Start” and runs that state. More on producer-consumer loops (also called Queued Message Handlers) here: <https://www.ni.com/en-us/support/documentation/supplemental/21/using-a-queued-message-handler-in-labview.html>.
2. The figure below shows the block diagram for reference:



1. When the VI is run, the initialize case will be run in the producer loop. This case updates the visibility of displays and starts all hardware. \*Note: this is different from previous versions. By initializing HW when the VI is run, equipment does not require synchronization to start tests. When tests are started, the equipment is already initialized and synchronized.
2. When the start button is pressed, the producer loop goes to the Initialize inflation state and saves the start time:



1. Next, the Test case is run. In this case, all test data and variables are sent to the Inflation Test VI for data acquisition and control. This VI outputs an array of force and diameter data. At the end of the loop execution, the time is record for the next loops start time:



1. In Inflation Test VI, the current test time is calculated by using difference between the current loop execution absolute time and the last loop execution absolute time (calculate time VI). The time value is sent to the setpoint VI to get the current setpoint value based on the relative time. Meanwhile, force sensor data is acquired from the USB-6211. The setpoint, force data, and other test variables are sent to the control VI. The control VI outputs a PWM percentage that gets sent to the servo VI. The servo VI sends the PWM signal directly to the servo. While all of this is going on, camera data is acquired and processed from the camera. All dbl data values are sent to the data array: 