SENSORS AND MOTORS LAB

Lekha Walajapet Mohan

Team D: HARP

Teammates: Alex Brinkman, Rick Shanor,Abhishek Bhatia, Feroze Naina

ILR01

Oct. 14, 2015

I. Individual Progress:

For the Sensors and Motors Lab, I was involved in executing the Graphical User Interface. My teammate Feroze Naina, initiated the GUI and myself along with Alex Brinkman worked on design, implementation and facilitation of the serial communication.

For the GUI, I utilized the PyQt and PySerial packages. PyQt has a dataware and graphical user interface widgets. We have QtCore modules that contain non-GUI classes, the GUI classes are available in QtGui module , which has design templates and structure for the layout of the interface. UIC module defines the GUI by generating the python .py file . We communicated the data serially and sent data from arduino over to serial pc. We controlled dc motor with infrared laser finder, servo motor with a rotary encoder and stepper motor with ultrasonic range finder. To manipulate the values, we integrated the call back function that would vary the values manually or as per the given input and call the changes global.

Algorithm:

Step 0: Start
Step 1: Get input of sensor value, motor value and state
Step 2: Set the control mode, display text and control the slider value
Step 3: Set time periodically and initiate call back
Step 4: Start timer – replaces initial call to periodic call
Step 5: Set thread to do asynchronous input and output
Step 6: End

Alex was working on dc speed control, Rick was implementing dc position control, Abhishek was working on stepper motor and Feroze was working on servo control(Fig 3). Alex and Rick together implemented the encoder library. I was working on design and implementation of GUI (Fig 1). along with Alex and Feroze.

The picture below depicts our GUI layout. The motor selection option allows the user to switch between various types of motors and the control mode implies whether the motors are controlled manually or by changing the values in the slider. The current values display the latest global values of the running motor. During manual mode, the value of the slider is changed by the input user and GUI constantly links with the arduino for updating the new data. For example, stepper motor can be given or slided to values ranging from -180 degrees to 180 degrees.

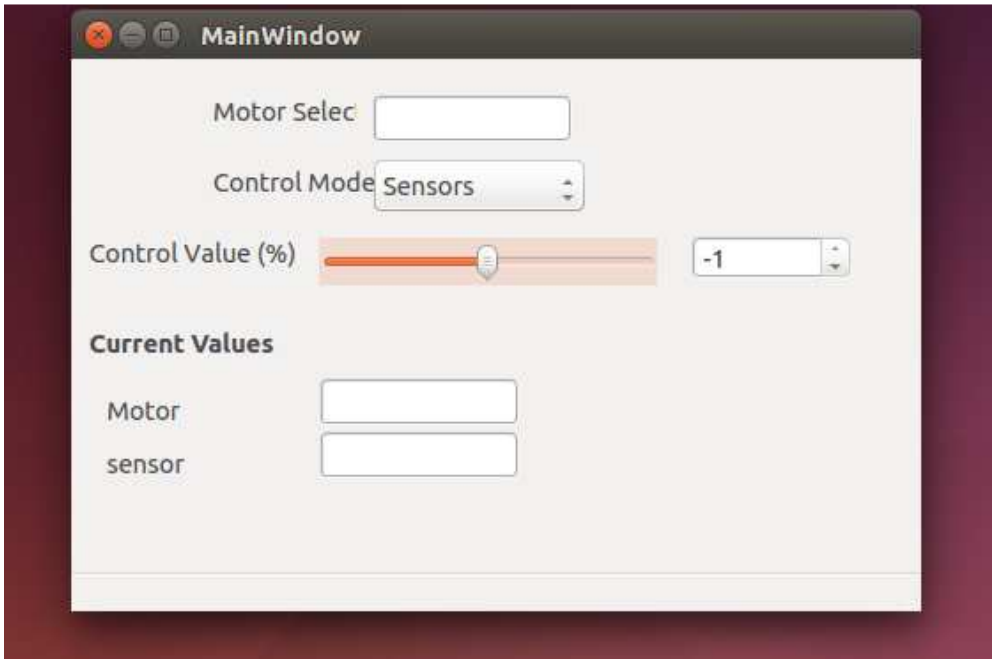

Fig 1 GUI Layout

On every loop, arduino checks for an input from the GUI, where the new values are updated from the arduino .

II **Challenges**

The biggest challenge was to integrate the code. We were able to get the GUI working separately, but integrating was a huge problem. Moreover, it worked on few of our laptops and refused to give any output in a couple of our laptops. This meant presence of other members of the team, who weren't assigned that particular task. Later, we realized that the serial port was not closed causing the function to freeze.

I feel that we should have approached TA's for help, so that we could have saved a lot time and focus more on testing the circuit. My take away from this assignment is that we need to prioritize and make the most of our time. Instead of spending too much of time on debugging, we could have asked for help. When it comes to my project, this lesson has taught me a valuable lesson. I am now connecting to people with expertise in my area of focus (perception). I'm getting great insights and warnings as to where my subsystem would fail, so that I have adequate measures to encounter them.

III **Team Work**

We used pulse command to read sonar sensor values (Fig 2). and analog digital communication for potentiometer and IR sensor. As a team we got started with Alex working on dc speed control, Rick was working on dc position control. Abhishek worked on stepper motor control and Feroze on servo control. Alex and Rick together implemented the encoder library. Myself and Feroze got started with the QT GUI and Alex joined with me in designing and implementing GUI. He later worked on serial communication between PySerial and Arduino, that would enable passing of stream of data between Arduino and GUI. Abhishek compiled the segments of Arduino together.

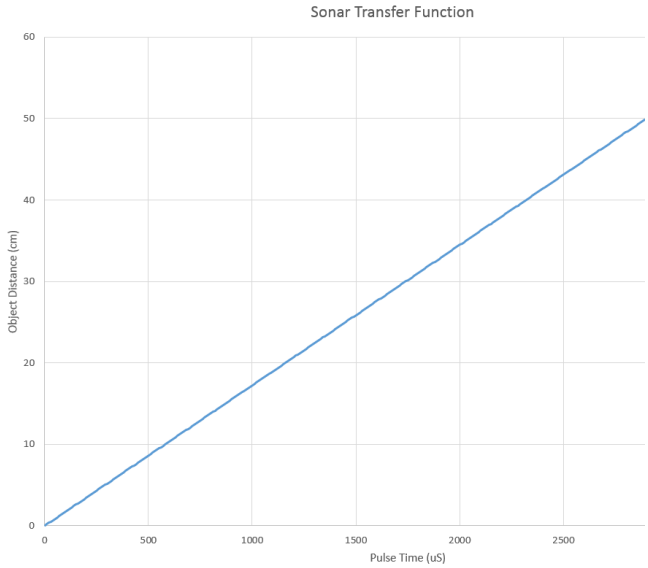

Fig 2 Sonar Transfer Function

IV Future Plans

With respect to the project, I feel that we are making a good progress in our respective areas. We were able to secure PR2 for testing purposes. Currently, I have started with the perception system, where I am training the PR2 with data set release by Amazon Picking Challenge. With the guidance of experts in computer vision, I am able to scrutinize various algorithms available for object detection and matching. By two weeks, I plan to have a trained algorithm that has atleast 85% accuracy of recognizing objects.
Alex and Feroze will be working on ROS integration, Abhishek will be working on state machine, simulation and electronics. Rick specializes in mechanical design and arm control and hence he will be focusing on that specifically. By the next project review, we will have ROS set up, a complete trained perception system and test results and analysis of vacuum suction gripper system and electrical design.

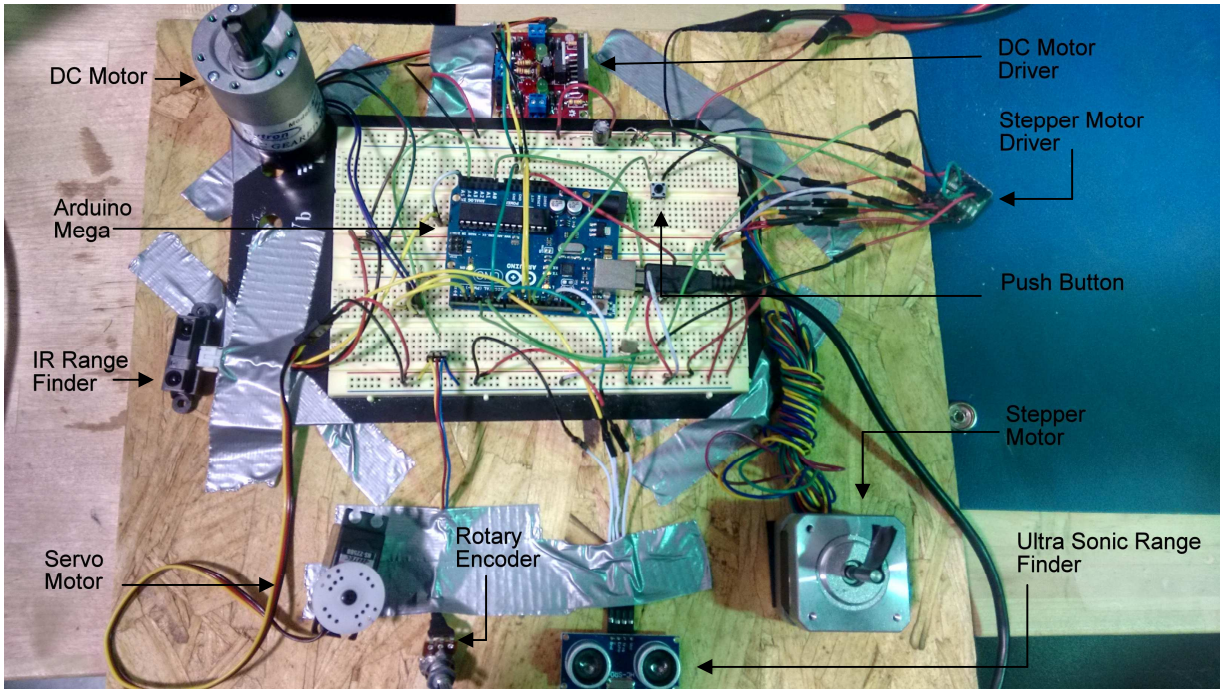

Fig 3 Breadboard connection circuit