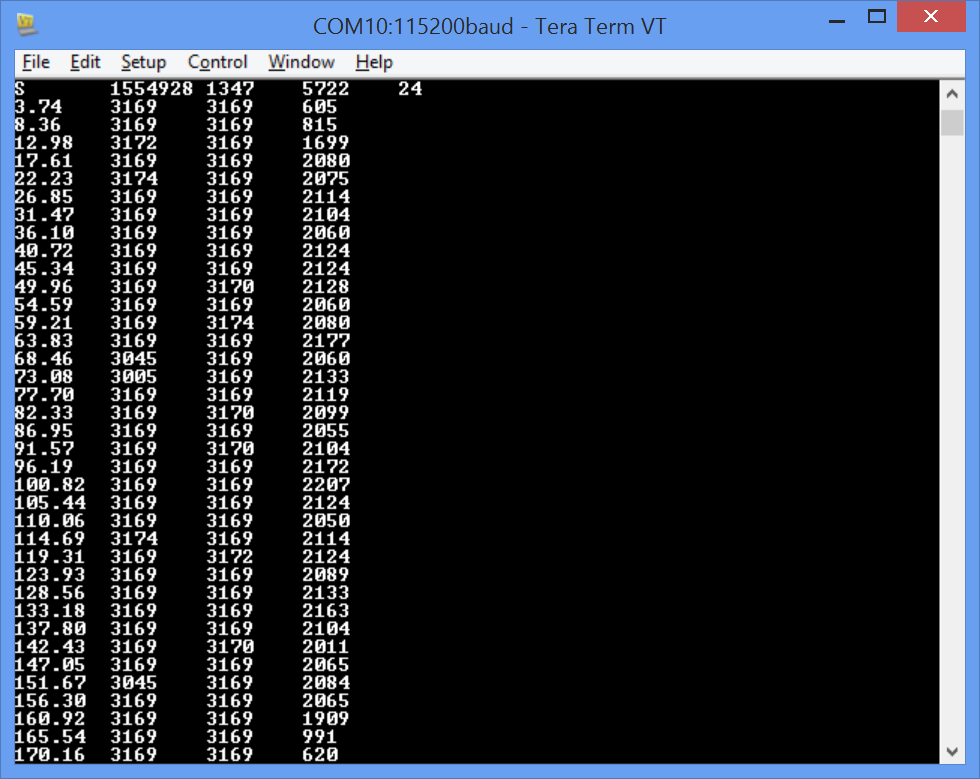
**Invisible Robotics Bumper (iRoboBumper)  
System Demonstration**

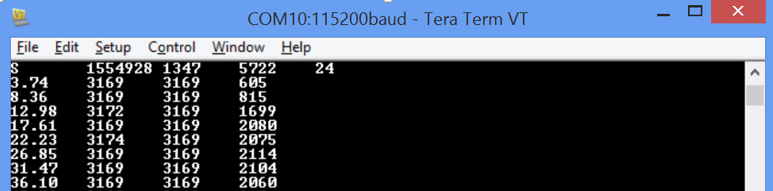
The Arduino Sketch outputs all of the data from the iRoboBumper system as ASCII text. This was intended so that a PC with a second XBee module connected to it can be used to see the output of the system.

Figure 58 below shows a screen shot of the system's data captured from a serial terminal application on a PC. Remember that the system is taking a reading from each PING)))sensor 50 times a second and outputting it via the serial port, so data scrolls down at a decent pace. The capture in the figure shows 37 sets of data which represents approximately 0.75 seconds in time.

  
Figure 58 - Serial Terminal Data Capture

The Arduino code outputs two different types of data and all data is tab delimited. The first type of data is a status line that starts with capital 'S'. This type of line is output once, every time the "home" alignment hole is detected. The 'S' is followed by four values that represent:

* The time it took the system to rotate the previous 360° (i.e., pan time) in microseconds
* Present commanded servo PWM time in microseconds
* Present battery voltage in millivolts
* Present temperature in degrees Celsius.



**Figure 58b – Top Portion of Serial Terminal Data Capture from Figure 58**

Thus, in the Figure 58 (partially shown again in the above as 58b) we see the following line at the top:

S 1554928 1347 5722 24

This tells us that it took a little over 1.5 seconds to pan 360 degrees, the servo's PWM signal is high for 1347 microseconds, the battery voltage is 5.722 V, and the temperature is approximately 24 °C.

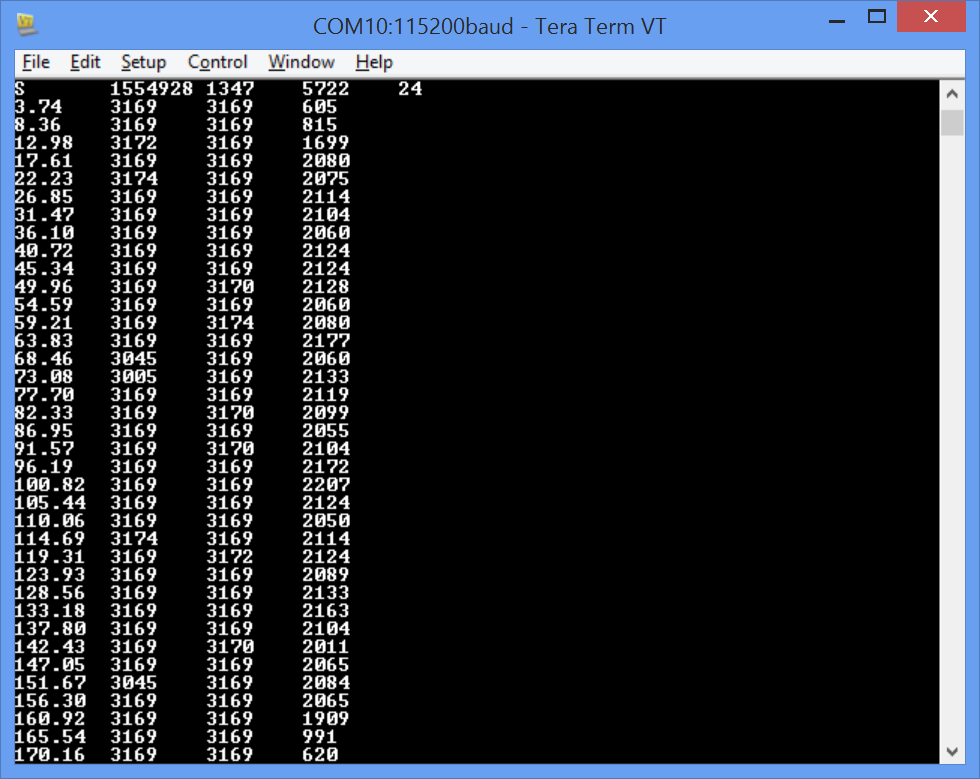
The second type of line contains the sensor data and is output roughly 50 times a second. These lines represent, from left to right:

* + The approximate angle that the system has rotated since the last time the "home" alignment hole was detected in degrees
  + The reading from the first PING)))sensor in millimeters
  + The reading from the second PING)))sensor in millimeters
  + The reading of the proximity sensor in millivolts.

We also see the following line immediately after the status line:

3.74 3169 3169 605

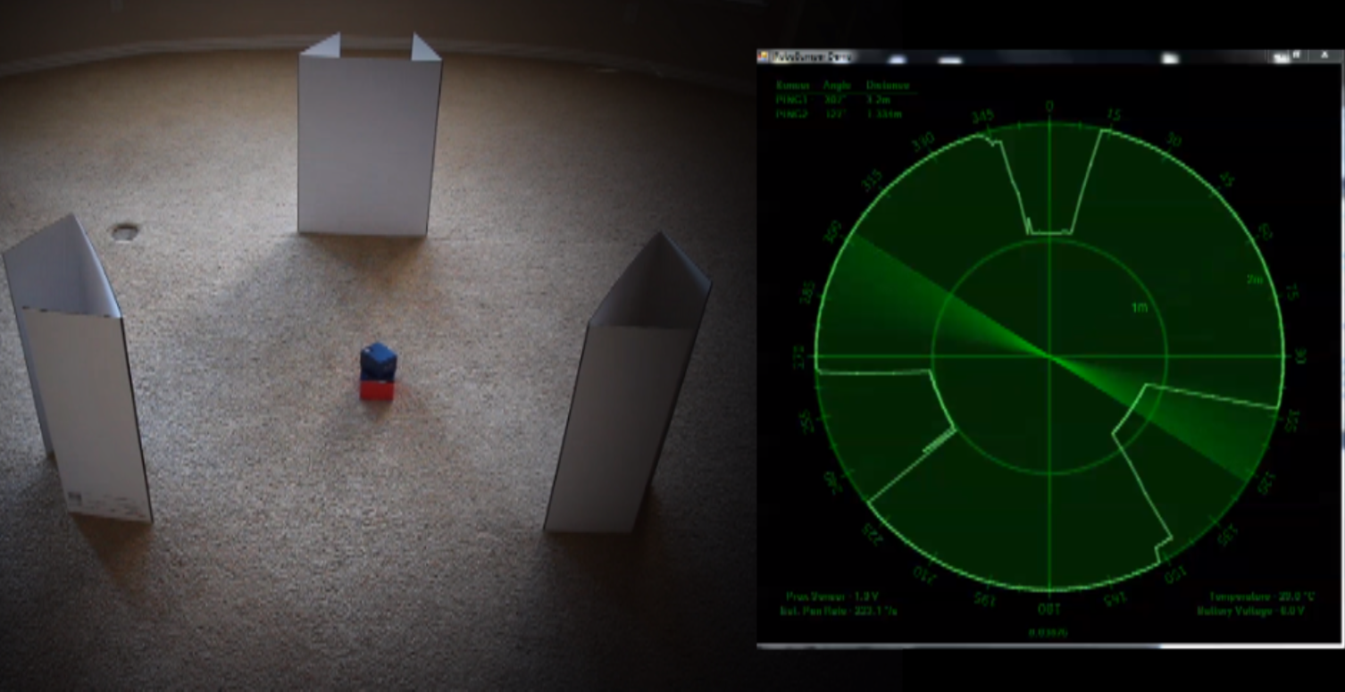
This indicates that the system has rotated ~3.74 degrees from "home", both PING)))sensors reported a range of 3.1m (i.e. this is typical when no return was received and thus indicates there is nothing closer than 3 meters), and finally the voltage from the proximity sensor was 0.605 V.

  
Figure 58 - Serial Terminal Data Capture

Notice that the proximity sensor voltage (the last number in the line) is relatively low here, but rises to approximately 2.1 V at about the 6th line, where it stays until it starts falling at the bottom of the capture in Figure 58. The drop in the proximity sensor voltage is due to the proximity sensor detecting the second alignment hole that is 180 degrees from the "home" alignment hole.

In this run, at approximately 170 degrees we have fallen to 0.6 V and appear to still be dropping. This is expected as the data represents approximately 0.75 seconds and our pan rate was 1.55 seconds on the previous pan. Thus, we should be nearing the half way point of this pan (i.e. 180 degrees).

Obviously, this is not the optimal way for us to view the data, especially when it is streaming out at 50 times a second. To help us visualize the data, a Microsoft Windows application was created to receive, parse, and display this data. Figure 59 below is a screen shot from the demonstration video and shows the iRoboBumper system with obstacles on the left and the demonstration application on the right.

Figure 59 - Demonstration Application Example < Insert video here>

The application visual shows the most recent distance data for the full 360 degree area around the system. In addition, it show the angle that each PING)))sensor is pointing at the current time. The outer ring of the circle represents 2 meters and the inner ring represent 1 meter. Thus, each of these targets are around the 1 meter range, but the bottom right target is slightly closer. It is hard to read in the image above, but the current pan rate, battery voltage, temperature, and proximity sensor data are all displayed in the area surrounding the distance data. The demonstration video for this project shows the system in operation while the demonstration application is running, with different sets of obstacles including obstacles that are moving towards the system as well as obstacles being removed.

The demonstration video can be viewed at http://

The demonstration application can be downloaded at http://

# Back to Top