

ECE 3651 Final Documentation

Andrey Kurenkov

April 27, 2015

Parts used in this project

- The Pololu 3π robot
- The arduino ADK
- Nexus 7 Android Tablet
- Galaxy S Phone with Android 4.0
- Misc. electrical connectors

All the developed components include the following: an Android Client app for the tablet, the 'server' app that controls the robot, the ADK code that forwards messages from the phone, and the Pololu code itself. All of these things can be found at my public GitHub repo. None of the parts are very complicated, but getting them to work together was a little difficult. I ended up using a set of pins with SerialSoftware on the pololu bot, which communicated to the Serial1 port on the ADK. The ADK then used the `AndroidAccessory` class to receive bytes from the phone, which itself was connected to the tablet via a standard socket connection. I actually wrote a generic handler-based class for the receiving side, which exclusively listened for the string 'driveCommand' but could actually handle many types of commands. Beyond having this fancy command-listener, the actual contents of the messages are simple 2-byte messages for direction and speed.

I was able to get a standard camera stream working on the phone, and also to get photos taken, but any attempts to stream video failed. Naturally I tried the suggested approach from StackOverflow first, but it failed. I also got a solution that took photos on a timer working, but it was unstable and seemed like it would fail. When I have more time I will explore this very promising post in Russian, which appears to promise an approach that I should be able to use.

A surprising aspect is that writing the control flow for the robot turned out to be more difficult than expected. I think in the future I will have both the joystick-esque controls as well as two progress bars to control the speed of both motors directly. The control logic is as follows:

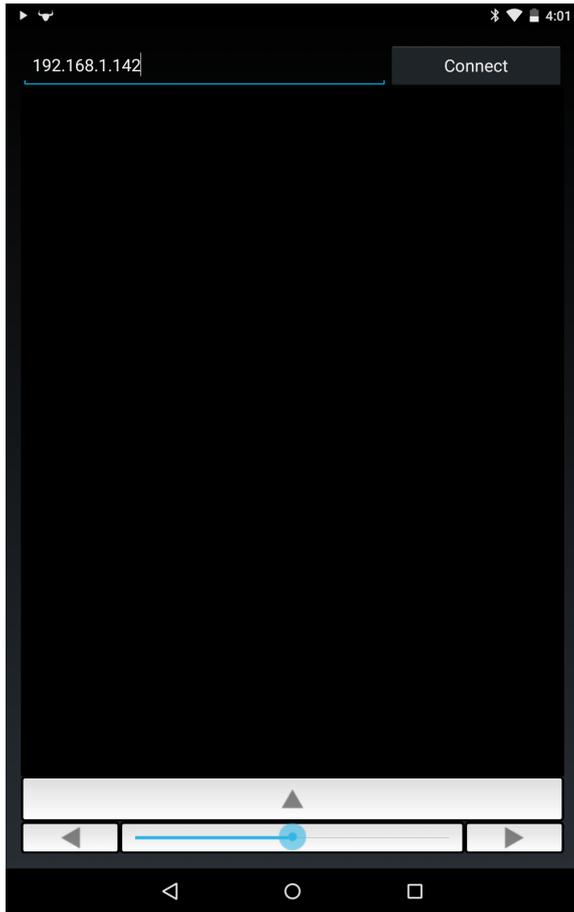
```
char mode = 'S';
```

```
// The main function.
```

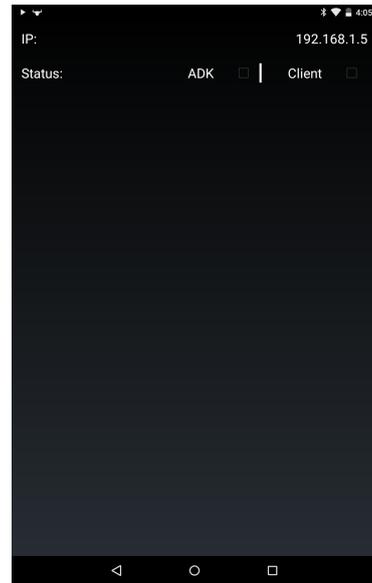
```

void loop()
{
  if (smcSerial.available()>2){
    char command = (char) readByte();
    byte speedByte = readByte();
    if((command=='S' && speedByte==0) || (mode=='S' && command!='S')){
      mode = command;
      if(speedByte==0)
        OrangutanMotors::setSpeeds(0,0);
    }
    if(mode=='F'){
      OrangutanLCD::clear();
      OrangutanLCD::print("Forward!");
      OrangutanMotors::setSpeeds(speedByte,speedByte);
    }
    if(mode=='R'){
      OrangutanLCD::clear();
      OrangutanLCD::print("Right!");
      OrangutanMotors::setSpeeds(-speedByte,speedByte);
    }
    if(mode=='L'){
      OrangutanLCD::clear();
      OrangutanLCD::print("Left!");
      OrangutanMotors::setSpeeds(speedByte,-speedByte);
    }
    if(mode=='B'){
      OrangutanLCD::clear();
      OrangutanLCD::print("Back!");
      OrangutanMotors::setSpeeds(-speedByte,-speedByte);
    }
  }
}

```



(a) Client Application - connect and controls work, but video not yet displayed.



(b) The phone server app - display IP and current connections.

Figure 1: Images of Android apps - both taken on Nexus, though the second one will run on the Galaxy S.