# Programming Things Report

Moving

To start things of, moving the Zumo was really straight forward when just working with the W,A,S,D keys because there was little that I had to expect (it was a pre-set instruction that I could simply code in) but when exploring other methods of trying to move the Zumo based on the environment then that got a little tricky. The first task was to get the Zumo moving with W,A,S,D but the second was moving the Zumo without any interaction from the user but to start it. This is where I found the resource from pololu <https://www.pololu.com/docs/0J19/all> very useful. This specific link, is to a document that has a lot of information about the reflectance sensory array. This allowed me to have the Zumo keep moving forward and stopping when reacting to the reflectance sensors.

Turning

When I started to implement the turning mechanic, I tried setting the speed to the motors but what I noticed is that that the Zumo would turn in a constant circle without stopping. To fix the issue I implemented a delay after the speed was set which allowed me to turn with more precision.

Xbees

Communicating over Xbees was very difficult at first because of little things that caused a lapse of communication (mainly flipping the switch on the Xbee shield). Overtime this became muscle memory and making the adjustments were effortless.

Stopping at the end of a corridor and adjusting course

This is where the reflectance sensor array comes into play. The way that I’ve coded my sensors is that the sensor furthest to the left and the furthest to the right must hit the wall at the same time to stop the Zumo completely. If one sensor is triggered but not the other, then the Zumo will adjust its path to continue with the designated route.

Display

The display on the GUI is responsible for receiving messages from the Zumo. Every single time an instruction has been executed then a message gets sent to the gui. The biggest obstacle was to get a customer message sent to the GUI (something that gets progressive i.e. sending the gui what room your in) but because of the nature of Arduino and processing it was really straight forward to right to the serial with a the relevant message.