

MTRX2700 Progress Report

William Thorp – 510548722

Daniel Monteiro – 500446063

Lemuel Fernandez – 510433668

Duncan Chen – 500459742

Henry Ryan – 500501085

Description of the project and aims:

The aim of our project is to map the location of a “trolley” (the trolley will be modelled by a microcontroller on a wheely chair) within a “supermarket” (modelled by the Tron Lab). A user will be able to search for an item or category of items within the supermarket, and the trolley will then display:

- Trolley current location
- The aisle of the item location
- The location of that aisle

The trolley will then point in the direction of that aisle (within the constraints of the servo motor’s range) using the lidar to guide the user to their item.

The trolley will complete mapping using the following method:

- Magnet markers are placed on the floor of the supermarket at equal distances. Markers will alternate between horizontal and vertical.
- The controller will calculate the direction of movement for the microcontroller and both the current and previous magnet state.
- By knowing the direction and magnet readings, we can determine the location of the trolley using the following logic:
 - If current magnet X == HIGH and previous magnet Y == HIGH and velocity > 0:
 - We are at location of next magnet.
 - Else if current magnet X == HIGH and previous magnet Y == HIGH and velocity < 0:
 - We are at location of previous magnet.
 - Else if current magnet X == HIGH and previous magnet X == HIGH:
 - We have moved back to the current magnet.
 - The above logic also applies if X and Y are swapped.
 - If both magnets == LOW and velocity > 0:
 - We are at the location between current magnet and next magnet.
 - Else if both magnets == LOW and velocity < 0:
 - We are at the location between current magnet and previous magnet.

| Magnet high current | Magnet high previous | Velocity | Response |
|---------------------|----------------------|----------|------------------------------|
| X | Y | + | Next magnet |
| X | Y | - | Previous magnet |
| X | X | n/a | Current magnet |
| Both low | Both low | + | Between current and next |
| Both low | Both low | - | Between current and previous |

Modules needed for project:

HCS12 modules:

- Magnetometer
 - This module should read in the raw data from the magnetometer. It should then return a binary response for both the x and y axes of the sensor – 1 if saturated (or above a certain threshold) and 0 otherwise. It should keep a record of which axis is currently high and which was previously high, as well as if neither are high.
 - Duncan is responsible for this module (with some help from Daniel). Progress as of week 12 – partially complete.
- Accelerometer
 - This module should read in the raw data from the accelerometers, and also take in the values from the magnetometer module. Some basic filtering should be completed to remove high frequency noise. The signals should be integrated once to determine velocity in x and y. The velocities are returned. These velocities should be reset every time the magnet reading is high to reduce integration drift.
 - Daniel is responsible for this module (some small assistance from Will). Progress as of week 12 – partially complete (filtering the signals to avoid drift error in the integration has been challenging).
- Location module
 - This module should take in the binary responses from the magnetometer and the velocities from the accelerometer. It should then return the location of the trolley with respect to a 1-dimensional grid. The grid resolution is double that of the number of magnet markers. The location is determined using the logic defined above (in the description and aims section).
 - Henry is responsible for this module (with some small help from Will). Progress as of week 12 – small (pseudocode only).
- Servo
 - This module should take in an angle and rotate the servo/lidar to point at that angle relative to the trolley orientation.
 - This module has mostly been prewritten by Stewart. Only small modifications required to allow input to be an angle in degrees. Daniel is responsible for this module. Progress as of week 12 – complete.
- Gyro
 - Get in raw data from gyro and apply some basic filtering. Integrate once to return the angle rotated (by the servo and trolley combined).
 - Daniel and Will are responsible for this module. Progress as of week 12 – partially complete (filtering signals to avoid drift error in the integration has been challenging).
- Orientation module
 - This module takes in the angle rotated by the servo and the measured angle of rotation from the gyro and calculates the rotation of the trolley.
 - Will is responsible for this module. Progress as of week 12 – very little (pseudocode only).
- Serialisation module

- Takes the position in the grid and the orientation of both the trolley and servo with respect to the trolley, then sends it to the python script. Receives the angle that the servo should be rotated to point towards the searched item.
- Lemuel is responsible for this module. Progress as of week 12 – partially complete.

7-SEG: takes aisle number determined by magnet module and displays on 7SEG

LCD: Displays directions to destination. E.g. continue forward for 2 aisles then turn left.

Direction generation will take current aisle and destination aisle and calculate directions to send to LCD.

Python modules:

- Serialisation module
 - This module unpacks the serialised data sent from the microcontroller to be used for updating the display. It sends the angle of rotation required by the servo.
 - Lemuel is responsible for this module. Progress as of week 12 – partially complete.
- Search module
 - This module allows the user to search for items or categories in the supermarket. This requires creation of a supermarket csv (or similar set of data), a search function with error handling (i.e. correcting misspelled searches or identifying when items are not in the supermarket), and returns the aisle the item is in.
 - Will is responsible for this module. Progress as of week 12 – complete.
- Mapping module
 - This module takes in the location data from the serialisation (trolley location) and search (item location) modules. It then displays these locations on a basic map of the Tron Lab using pygame. The map should also display the orientation of the trolley and orientation of the lidar. This module should also calculate the angle of rotation required by the servo.
 - Will and Lemuel are responsible for this module. Will is responsible for the pygame display, sprite animation and angle calculation, Lemuel is responsible for integration of data. Progress as of week 12 – partially complete with map, serialisation and search function integrated together (courtesy of Lemuel).