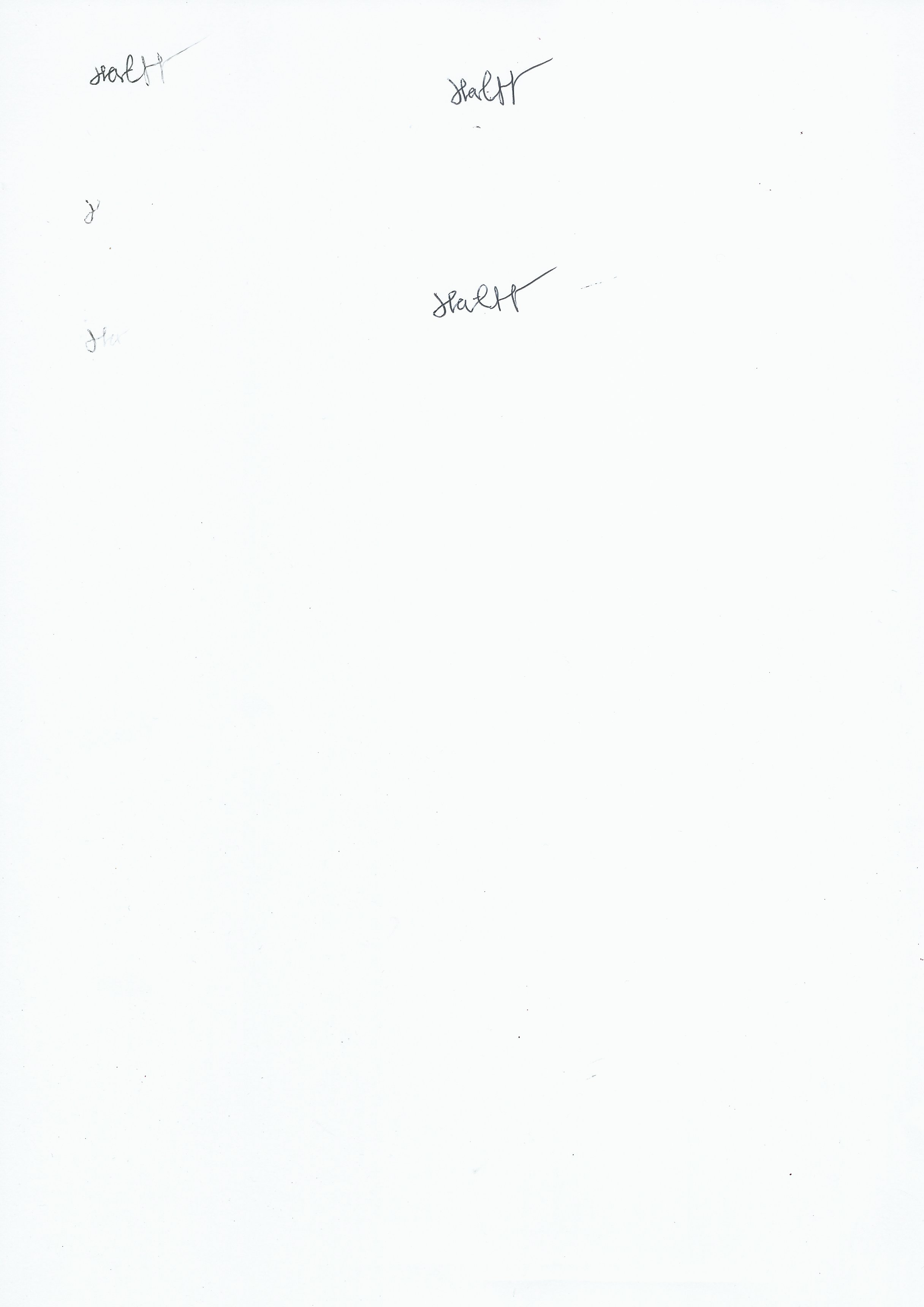
**Integrated Design Project**

**Final Report**

**Michaelmas Term 2017, Team 105**

By Karthik Suresh

Wolfson College, LG56

*I confirm that this work is my own. *

**The Problem**

The task called for a robot to be manufactured that could:

* First, pick-up and identify six golf balls of assorted types.
* Next, deliver all six golf balls to their correct locations, corresponding to their weight and colour.
* And finally, return to the start.

The robot was required to complete all these tasks autonomously, within 5 minutes.

**Result**

Team M105’s robot, ‘Tiger Woods’ Ball Handler’, started by successfully picking up and identifying the first ball. It then picked up the next ball successfully but failed to identify it correctly. The third ball got stuck in the entrance of the storage ramp and was not only identified incorrectly, but also prevented the fourth and fifth balls from being able to be identified. The fifth ball was dropped while being picked up but the robot managed to store the first four balls successfully. The robot was not programmed to attempt to pick up the sixth ball.

The robot then successfully travelled along the lines to the start node of its delivery sequence. It calculated the correct drop-off path and dropped the first ball in the correct bin. The robot’s drop-off actuator’s poor implementation caused the robot to unintentionally drop the second ball while dropping the first ball. The robot then travelled to the right node to deliver its next ball, but failed to stop correctly. The robot’s operation was terminated here as it mounted the heavy ball drop-off bin. The robot scored 25 points for its effort.

**Initial Design - Key Considerations, Difficulties**

Time optimisation was the first criterion which was prioritised when considering designs. With only 5 minutes to pick up and deliver all 6 balls, it was decided that the robot needed to be able to pick up and store all six balls, so that only one trip needed to be made between the pick-up and drop-off areas. A two-tier ramp was designed so that gravitational potential energy could move balls into a compact array, so that ‘pick-up’ and ‘drop-off’ could be conducted from the same side of the robot. It was believed that having the robot reverse would be troublesome and a significant source of error.

Simplicity was another key criterion that was prioritized during design formulation. It was thought that the simplest systems would lead to the smallest number of mechanical errors. Steps were taken to ensure that multi-arm or rotating systems were avoided. Thus, during the design stage, a rotating release mechanism system to drop the balls was abandoned in favour of a simpler ‘push’ actuator.

In retrospect, the most difficult part of the initial design was being able to foresee the ‘right’ compatibility issues. For example, the two-tier ramp system was difficult to assemble precisely. Several stop-gap measures had to be taken to rectify issues that restricted balls from travelling into, down and off the ramp reliably. Another example of this could be seen with our dropping mechanism. While the ‘push’ actuator was a simple solution, it was not infallible. A more intricate mechanism like a rotating flap would have been more successful.

A key misconception during the initial design phase was the idea that a simple and accurate system did not require in built redundancies, or the incorporation of flexible constraints. To manufacture the robot exactly to design specifications, or to write a perfect line following algorithm, proved to be unfeasible.

**Project Schedule**

The initial project schedule estimated that all production would be completed 10 days prior to the competition date. Only electrical production was completed on schedule. Mechanical production was completed 4 days prior to the competition date and software production continued until the evening before the competition date. Problems with computer assisted drawings, compatibility issues between waterjet cut parts and additional mechanical requirements were the main reasons for a delay in mechanical production. Fixation on perfect line following and a failure to predict the complexity of code associated with sub-systems other than line following were the main reasons for the delay in software production. Furthermore, the effects of minor delays were amplified by the increasing lack of table time in the last week as congestion in the lab increased.

With greater foresight, the project schedule could have been adhered to more closely. Greater concurrent interaction between all three sub-teams to maintain and continuously develop a mental picture of the robot would have enabled the software team to complete their fundamental tasks within the projected timeframe. Even with the delay in mechanical production, the project could have been more successful if in the few testing days at the end of the project only calibration issues were left to be resolved. However, within the little table time that was secured, in addition to calibration tests, line following issues and pathfinding issues had to be dealt with. Inefficiency in testing was also exacerbated by the inability to diagnose the source of issues quickly. For example, on the last evening time was wasted because a software solution was being sought after for a rectifiable mechanical issue.

**Key Software Considerations**

The software design was carefully thought out and built with a large degree of modularity so that changes could be made easily without altering code for other sub-routines. The main issues faced by the software team revolved around line following.

In early discussions, the need to place IR sensors on the drive axle of the robot was prioritised as it was thought that this was the only way the robot could stop on a junction accurately, and thus turn accurately. It was also thought that the robot needed to be have ‘front wheel drive’ as ‘rear wheel drive’ would lead to inaccuracies because of resistance to motion in a front castor wheel. These assumptions later proved to be the source of bigger problems.

Placing critical IR sensors close to the drive axle resulted in the robot veering in and out while line following, as while on the axle the IR sensors would only feedback a deviation once it was significant. It was also thought that sensors on the axle would be enough to allow the robot to turn accurately, however it was found that the IR sensors were quite sensitive and at the width the sensors needed to be placed to follow a line on a straight path, only a partial turn was completed by the robot. Eventually, a decision was made to add a fourth sensor, along the robot’s centreline but far from the drive axle, to aid the robot with turning. The benefits of placing line following sensors far ahead of the drive axle could be seen from the performance of other robots during testing in the last week.

Despite the poor decisions made during the design phase, a function based on two different levels of veering was written that enabled the robot to follow lines reasonably well. The function was robust enough to follow a line accurately, but broke down when a robot started from a junction with the edge of its front sensors just off the intended line of travel. This problem only occurred occasionally, when the robot started to travel from the last ‘pick-up’ node to the first ‘delivery-node’. This was only discovered late in the last week of the project and instead of trying to rectify the issue by writing more code, it was thought more efficient to just reduce the chance of this happening by not trying to pick up the sixth ball.

The critical reason for the robot’s failure is unknown. Simulations done after the test coupled with analysis of the test video reveal that the robot calculated the path correctly, and that the IR sensors on the robot registered the junction it was meant to stop at.

**Key Mechanical Considerations**

Mechanical production was made challenging because of inaccuracies in the parts cut with the water jet. Extra time was spent re-drilling misaligned holes, and hand folding misaligned sheets. Furthermore, certain late changes in design required additional brackets and components to be manufactured. Ensuring these parts were compatible required time consuming trial and error.

An accumulation of inaccuracies led to systematic issues that compromised some of the robot’s functions. After assembly, the sweepers and its gear box had a lower clearance than what was stipulated in the design. This meant the motor that drove the gear assembly was repositioned to sit under the chassis and the sweepers had to be cut to fit over the wall on which the golf balls were placed. This led to ‘play’ within the gearbox which caused the pick-up process to be temperamental. Repositioning the sweeper motor under the chassis meant that the robot could no longer travel over the light ball white drop-off bins, to accurately drop balls into the cups at the back of these bins.

Certain sub-assemblies, such as the weight detection micro switch and drop-off actuator were not robust enough to perform their tasks reliably. Balls would often get stuck on the weight micro switch and the drop-off actuator would often fail to push soft balls out or push too many balls at once. These sub-systems could have performed better had designs been more thoughtful and more time been available to calibrate them accurately.

**Key Electrical Considerations**

The electrical circuits of the robot were produced perfectly. All sensor outputs and inputs could be read and written, reliably and independently. The only thing that could have been improved was the implementation of the LDR. More meticulous testing could have been conducted to understand how the LDR could produce distinct results and how the LDR results could have been made less susceptible to external conditions.

**Prototype Costing**

For a production run of a 100-1000 units, material costs, and electrical and mechanical production overhead costs would scale with the number of units produced. The cost of man hours and resources spent during the design phase would only be incurred once. Software production costs would also only be incurred once as software can be copied and distributed at very low cost (i.e. increase in number of units produced would not lead to a significant change in this cost). Testing costs are high for the initial few units, but once processes are refined they should eventually tend to a fixed value per unit.