UNIVERSITY OF CAMBRIDGE
Department of Engineering

TAIPEI 101 (M101)

Final Report

Name	Crsid	Lab group
Laura S. Clapham	lsc43	43EM
Ben L.C. Silva	blcs2	
Caleb C.C. Chan	cccc4	44EM
Tom Hill	th621	
Derek J. Dong	jd976	45EM
James C-C. Lecomte	jccl4	

I. Outline

II. Design details

1. Overall Design Approach

Our approach for the robot was to navigate to the collection zone using the line sensor, find the block using either the infrared distance sensor or ultrasound, clamp it in motorised arms, and then detect metal using a metal detector with a fixed position relative to the arms. We chose to use the line sensors as we thought computer vision was too difficult to implement in the time scale. This proved to be a good decision as one of our software team members fell ill and was forced to leave Cambridge, which would have made computer vision significantly harder to integrate.

We chose to use sensors to locate the block as we thought it would be more reliable than a mechanical system that attempted to sweep the entire collection zone at once. However, there was a lot of noise in the sensor signals which proved difficult to resolve and delayed the creation of a detection algorithm. This ultimately led to a redesign of our scanning process which was not fully implemented by the time of the competition. It is possible that by choosing a purely mechanical solution, we could have reduced the time needed to test and integrate, and therefore have been ready by the competition deadline. Our choice of arms was successfully implemented however, which provided good clearance over the ramp by elevating the block.

We decided to use a metal detector vs picking up the block with a magnet as we were unaware of the magnet's availability during the initial planning stage, and when it became available, we did not stop to reconsider our previous choice. To ease integration with other components, we decided on a state-based program that was entirely modular. This eliminated the risk of a change effecting other parts of the program and was the right decision to make for a large project. However, it did make it slightly difficult for team members who had not encountered a state-based system to help check the code in the later stages of the project.

[**Appendix D1: Whole system block diagram**]

2. Mechanics

Initial design of front section

At first glance, we thought of hooping and dragging the block from the collection area back to the deliver area, as a less complicated mechanism would be required. However, when we inspected the competition table, we discovered that there is a small discontinuity going from the table surface to the ramp which might prevent the block being dragged over. [**appendix M1**] Therefore, we decided to lift the block to provide clearance above the ramp. To do this, we need a front section composing of the grabbing components which can be raised and lowered by a servo motor. We need a servo motor for this purpose because it can lock its position in order to keep the front section lifted. A spool of wire was constructed out of steel sheets and mdf, with the steel filed down to be roughly circular. We decided water jetting a steel circle was unnecessary and too slow.

Mounted on the front section is an 18 RPM motor, two grabber arms and a worm gear. After some prototyping with cardboard, we discovered turning one notch on the arm gears would require one full worm rotation. Turning 6 notches to close the arms would take 20 seconds with the 18 RPM motor which we decided is too slow. As a result, the worm gear is replaced with two helical gears which opens and closes within 1 second.

The main platform of the front section is 4mm MDF board laser cut to 200mm x 8mm. The brackets for the helical gears are made from steel sheets, and the front section is connected to the main section by a steel plate spot-welded to a studding and bolted to the front section. The arms were prototyped in bent steel, and later changed to 3D printed.

Initial design of main chassis [**appendix M2: CAD of Robot**] [**appendix M3: Picture of Robot**]

As the arms will continuously grab the block and be lifted until the robot reaches the collection area, there is no need for a place on the chassis to store the block. Therefore, in order to keep manufacturing process as simple as possible and high accessibility to components, we went with a flat platform design. The chassis is made of 4mm MDF sheet laser cut to 200mm x 200mm.

We discovered it would be difficult to steer the robot using 4 wheels. Therefore, we decided to use a 3-wheel system, with two motorised wheels and a ball caster as the third support. This allows us to steer the robot by just giving the motors different angular velocity and let the ball caster swing left and right. The 4" wheels are chosen for two reasons: a) to maximise speed (40RPM x 4" x π = 12.8 m/min and b) to keep the chassis high off the ground enough so it does not get stuck on the edge of the ramp and the bridge. Although this raises the centre of mass, which could affect stability, we decided to go ahead as we believe our wheelbase is long enough to prevent the robot from tipping over. The ball caster is mounted onto the chassis with a long arm in order to keep the chassis levelled.

The servo motor is mounted on the main chassis, and it would pull a string attached to the front section, which would be pivoted against the front of the main chassis. Shelves are placed at the rear of the chassis to place the Arduino boards and circuit boards. The battery is mounted towards the rear of the chassis on the underside to offset the moment created by the front section and maximise space for electronics. Most of the mounts will be fabricated by bending aluminium sheets into the right shapes. The motor mount was originally 3D printed. However, given the wire heat-shrinked onto the motor casing gives it a lot of irregularities, the 3D printed mount could not grip the motor properly. So, we switched to using a MDF board cut-out which is mounted to the screw plate of the motor and the main chassis though an aluminium angle.

Problems encountered and changes made after integrating

When we first assembled the robot and tested the line following algorithm, we discovered some major problems.

First, due to wear and tear of the motors from previous years, the power and torque output by the 40RPM motors are not balanced, meaning 1 side of the robot has more driving force acting on it than the other side, causing the robot to always drift. The bigger problem is that the side with less power could not climb up the ramp, causing the robot to rotate up the ramp. To solve this, we switched out the 40RPM motors for 18RPM motors which are in slightly better condition and capable of producing more torque. As expected, the change turns out to be the right one.

With the 18RPM motor switched out, we had to use the 40RPM motor to drive the grabber arms, which caused the second problem. The 40RPM motor does not produce enough torque for the grabber arm to grip the block firm enough. As a result, we had to put something on the surface of the grabber arm to increase friction. We have tried sandpaper, heat shrink and rubber bands, at the end we went with spreading an even layer of hot-melt adhesive on the surface. Although it is not an elegant solution, it gives us just enough grip to transport the block.

As we were testing the line following algorithm, it was discovered that the robot was slightly understeering. This caused issues in line detection and pathing, as the algorithms worked off the assumption that the point of rotation lied on the driving wheel axes. To fix this we shorten the front section to bring the centre of mass towards the rear to allow easier steering.

3. Electronics

[Appendix E1: Overall schematic diagram] & [Appendix E2: Overall layout]

Approach to design and testing

There were three main tasks for the electronics team: metal detection, line following and block searching. Originally, we split tasks among the team and worked on them separately. This approach had its advantages, like easily being able to work on the project at home and getting two tasks done at once, but eventually we realised that it was more effective to work together on the same task. This meant we could use different perspectives and ensure that each other moved on when something wasn't working instead of obsessing over the same strategy.

The three main tasks remained almost completely isolated from each other throughout the project meaning we could easily ensure that each aspect was working properly before we used them all together. LT Spice was used to model behaviour of systems and to optimise values of components and the output signals were measured using a PicoScope and the Arduino.

The electronics team had to work with the mechanics and software teams closely. For example, the design of the metal detector affected its location on the robot and whether it would have to move up and down using a servo. The output of the metal detector had to be interpreted and processed by the software team. A rolling average algorithm was used to clean the signal and the result taken by the metal detector would have to change the state of the robot.

Metal detection

The metal detector uses an AD22151 magnetic field sensor that contains a hall effect sensor and amplifier circuit. It gives out a high voltage when the magnetic field strength is over a certain value and otherwise a low voltage. A magnet is placed above the sensor at a distance that gives a low value when there is no metal in between it and the sensor but a high value when the block is placed on top of the sensor. The circuit was soldered onto its own board and placed in between the grabber arms. As the robot drove up to the block, the block would be pushed onto the breakout board containing a hall effect sensor.

The magnetic field sensor was a late revelation as the solution for the vast majority of the project was to use a coil. The circuit would have used the fact that the inductance of a coil increases when in the presence of ferrous metal. The inductor was used in a tank circuit with a capacitor. When the inductance increases the charge collected on the capacitor, and so the voltage over the capacitor, increased and could be measured. The reason this was changed is explained in the reflections section.

Pathing (Line follower)

For pathing, we wanted to allow the robot to be aware of its position on the table, and to do so we want line sensors to detect lines. We want the robot to be able to follow lines and detect perpendicular lines it comes across, so we designed a system to satisfy both criteria. We put two line-sensors on the front of the robot since it is the minimum amount needed to gather enough information.

The line follower circuit includes an OPB704 line sensor which uses an LED, emitting IR light, and a phototransistor. When on the line a low voltage is output and when off the line a high voltage is output. This signal enters a SN7414N Hex Schmitt inverter before being input to the Arduino for use in the line following program.

The accurate position of two sensors were carefully considered. The main problem was that the distance between sensors and the ground changed when the robot going up or down on the ramp, and line sensors only work in a certain range of distance. We managed to find out that the rear of the front wheel is a good position, and specific distance was determined by several experiments.

Block searching

The final plan was to mount the ultrasonic sensor in front of the robot facing downwards and for the robot to scan the entire box by rotating to scan the entire width then moving forward and scanning the next width and so on. The failure of this aspect is explored further in the reflections section.

4. Software

[Appendix S1: Flowchart of software]

The basic of our software design is state-driven: we used Finite State Machine to determine robot behaviour s. All sensor inputs are converted to a rolling average to decrease the noise.

Pathing algorithm is based on 2 line sensors. Data from the line sensor is used to correct the direction of the robot by changing the velocity of the left and right wheel motors separately in a negative feedback loop. When one sensor circuit detected a white line the robot would know that its path was deviating too far to the other side and so it would turn in the opposite direction until this was corrected.

The program could remember when a horizontal line was crossed (detected when Arduino receives high signals from both sensor circuits). This allowed the robot to know which state to enter at which time, for example it would know when it had reached the collection box and so would know when to start the grabbing cycle.

III. Reflections

1. Team Structure Review

Our team have one team leader and 3 sub-teams (ME, Elec, Software). We used Slack to communicate, GitHub to share progress and Gantt chart to track deadlines.

3 sub-teams structure can make sure that every member has a clear role, which is quite manageable and normally increases team productivity. However, in a small project like IDP, some drawbacks are shown. First, this structure reduces communication between sub-teams, which means lack of information flow. For example, some of our team members did not know much about software designs. Before the first competition, all the members who knew how the software works happened to be absent until 2 minutes before the competition, and the three remaining members couldn't even start the robots. This resulted in us not having time to test the robot before the competition and thus failing in it. Second, it also leads to some under-utilisation of human resources (e.g. the software cannot be tested until the overall robot has been built). Potential improvement includes having more team meeting to help communicate and educating other sub-teams on how stuff works to allow them to know the overall graph.

Having a clear team leader can lead to a high team productivity and can ensure the decisiveness of the team in case of disagreement. However, the advantages highly depend on the leader: When the leader is not good at managing or cannot devote sufficient effort to managing, the team productivity decreases dramatically (this happened in the project).

2. Team Efficiency Review

From the team leader's point of view, the team M101 has good productivity, but it can still be improved by several measures:

- Avoid schedule mistakes (e.g. Dyson centre opening hours)

- Set specific tasks for each lab session instead of setting a vague goal of "do this by that time". For each phase rather than talking about what to do in general (e.g. set a deadline as "implement line-following before 16 Oct" "finish final chassis cut and start to build final circuit on it before 2 Nov" instead of "we should finish all designs before 10 Oct" "We should combine all parts to get a complete robot before 15 Oct". They can be good guidelines but not suitable deadlines.

- Consider team focus carefully. Vital tasks should be figured out and focused on in advance so that team productivity can be fully used to gain the most benefits.

- Report progress to teach other sub teams about how to get your finished work integrated so when some member becomes unavailable during the project (which happened to us), other sub team members can substitute and help with balancing workload.

3. Pathing & Block Searching Mechanism Review

Computer vision & line-following

Computer vision was seriously considered along with line-following as path-designing method at the beginning of the project. We considered it as a powerful tool but decided that this would not be necessary, since using it would be an over-engineered solution that has many failure points and would take too long to develop and is likely to be unreliable. It is too dependent on factors like the brightness level of the room and shadows cast on the table, etc.

Block searching

Implementing a block searching mechanism was the most challenging aspect of the project; there were many issues with the distance sensors that were not apparent at the start. When the block surface is not directly normal to the sensor the readings given are inaccurate and inconsistent. Therefore, the original design where an ultrasonic sensor would be placed at the front of the robot facing horizontally was problematic. Mounting the ultrasonic sensor on an antenna-like structure so that it points down solved this issue. This unfortunately was only sorted late into the project so it was difficult for the software team, who were often unable to attend, to code a way for the robot to scan the box. It became apparent that it was going to be almost impossible to finish the block finding system in time for the final competition so we had to make the decision to leave it out and focus on other things so we could at least secure 30 points.

Metal detection system review

One of the biggest major changes in the design of the robot was the switch from a inductor coil-based system to detect metal to a hall effect sensor at 3 weeks into the project. This was the correct decision to make as previous detectors using the coil were extremely noisy and unreliable, however it was a difficult decision to make as considerable amounts of time had been spent trying to make them work. With hindsight the decision should have been made a long time before but the small advances with the detector coils made it seem as though a breakthrough was always near. The main problems with using the coils were that they were insensitive and so struggled to detect the small amount of steel within the block. It was also time consuming to make different coils to test and optimize the components used in the circuit.

Competition's performance review

- Failure of 1st competition

Our expectation before the 1st competition was to reach the other side but not to collect or sort block. The robot had been capable of achieving this for several days, but the day before the competition we integrated the grabbing section onto the main chassis. We found in testing that this made the robot too heavy to climb up the ramp, so we knew that the front section needed to be removed for the competition. This meant it was impossible to pick up a block, but possible to cross the ramp.

However, during our testing of the robot, it seems that the left motor axle had come loose from the casing. This was likely due to a combination of historic wear and tear, and stressing the motor by having too heavy a weight for its torque limit. As a result, during the competition, in a configuration that had previously worked, the robot was unable to climb the ramp. It veered to the left and either ran off the ramp or toppled over. We had not been able to test the robot immediately before the competition, as only 3 members had arrived in time to test it, and none of them knew how the software operated. This revealed a flaw in our team management that we sought to improve before the final competition.

Coming out of the competition we had several priorities: to reduce the weight to torque ratio to an acceptable level, and to integrate the metal detector and detection algorithm. We solved the weight problem by swapping the motors as detailed in the mechanical section. We also ensured for the final competition that more people had a basic understanding of the software, and that we stopped working on the robot in plenty of time before the competition, in a state that we could test and see work.

- Performance in final competition

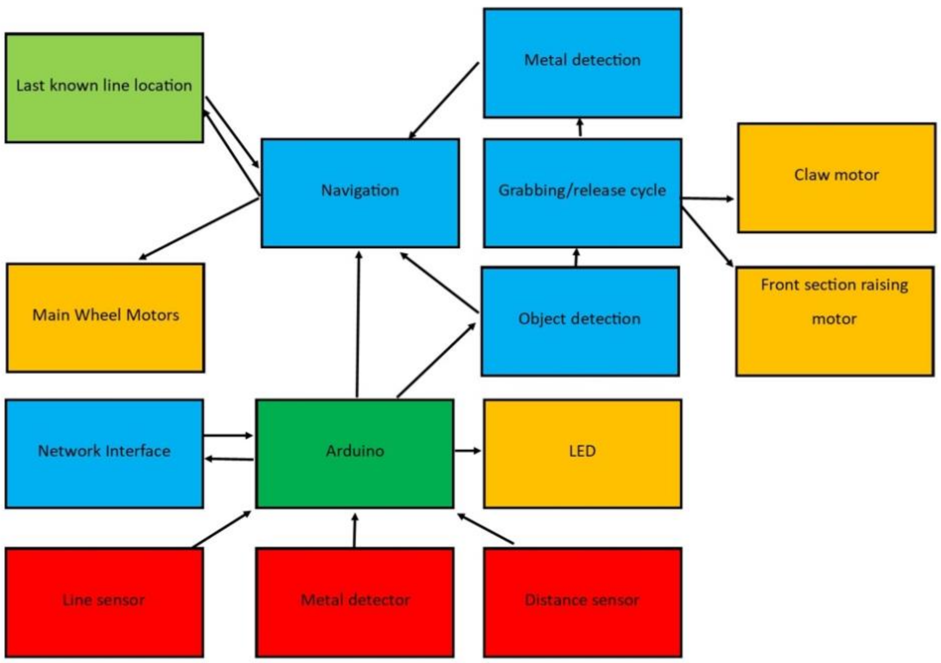
The day before the competition we had tested all the mechanical and electrical systems working together and all that was left to do was to integrate the software to create a scanning cycle to locate blocks. Our minimum expectation was that we could pick up the first block, as it would be placed in the same place every time, and we were hoping to be able to locate other blocks using the cycle.

However, on the day of the competition, our software team was not able to attend until an hour and a half before the competition. As more of the team had knowledge of the software vs the first competition, we were still able to test the robot and debug the code, but we were unable to create the new software needed for the scanning cycle. Therefore, we were unable to achieve more than our minimum expectation.

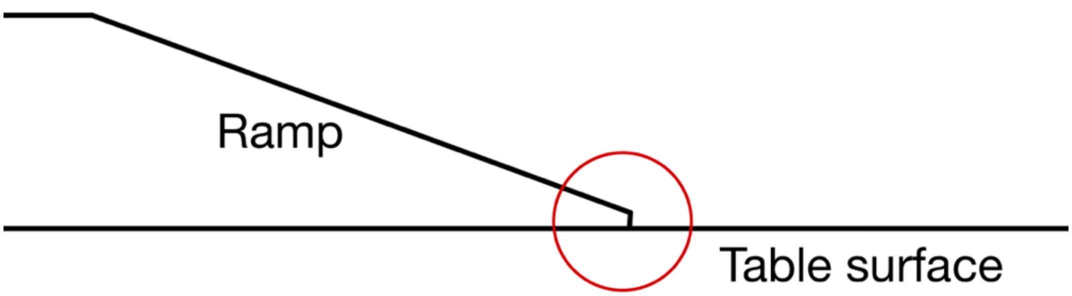
We did quite well in the final competition and the performance was exactly what we expected before competition: our robot managed to get the first block and correctly told the difference between metal and plastic. Compared with the first competition, our strategies changed: We still did R&D on the morning before the final competition, but each progress made was archived and ensured that it could be easily replicated. With an hour to go, we stopped development to concentrate on the competition and ensure a consistent, predictable performance in the competition.

Appendix

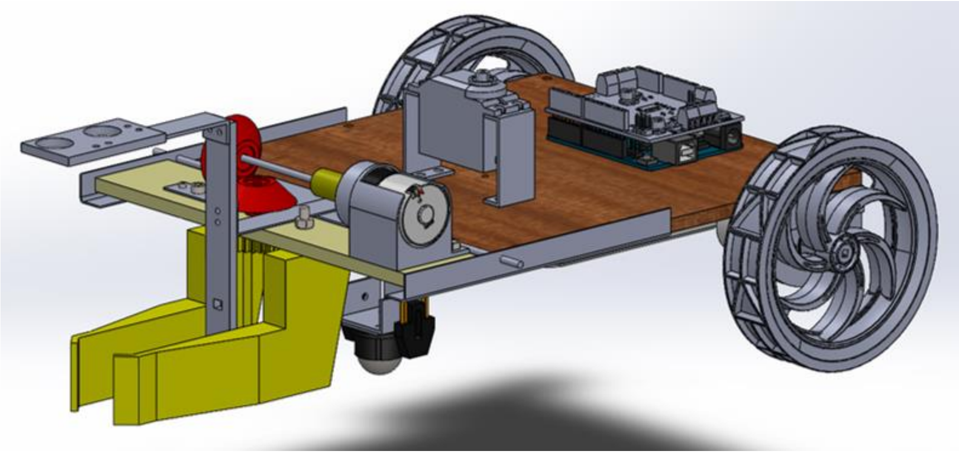
[D1: Whole system block diagram]



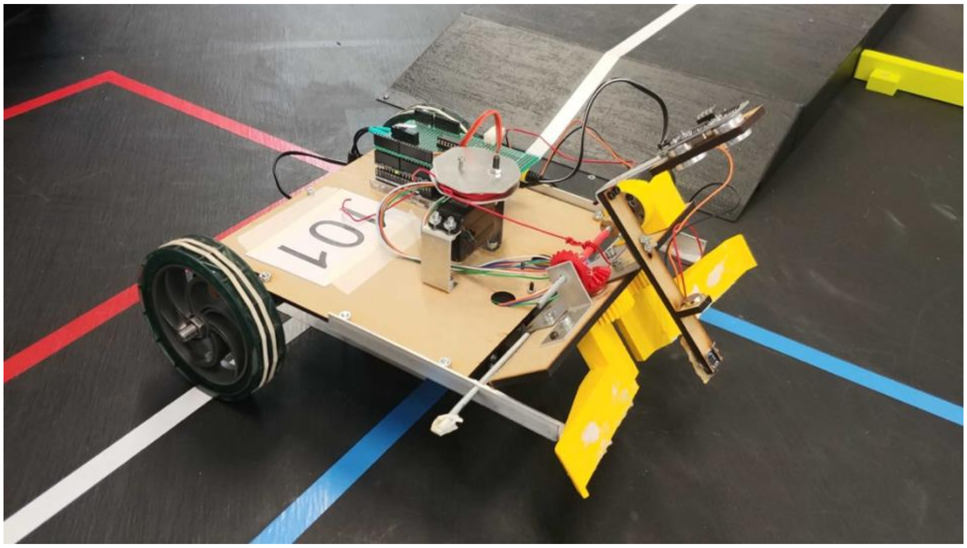
[M1: Illustration of discontinuity of table surface up the ramp]



[M2: CAD of robot]

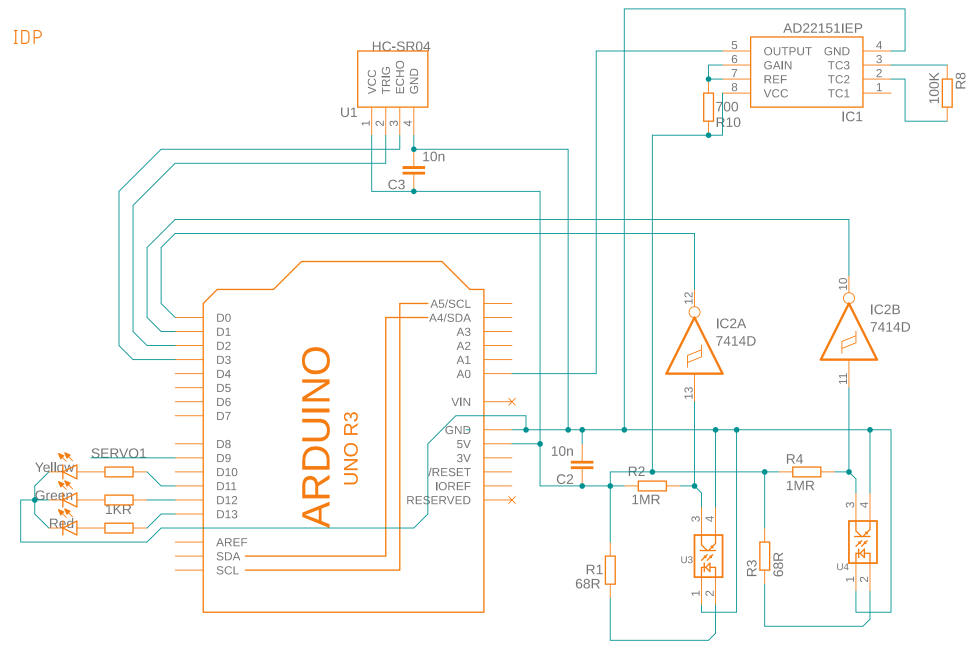


[M3: Assembled Robot]

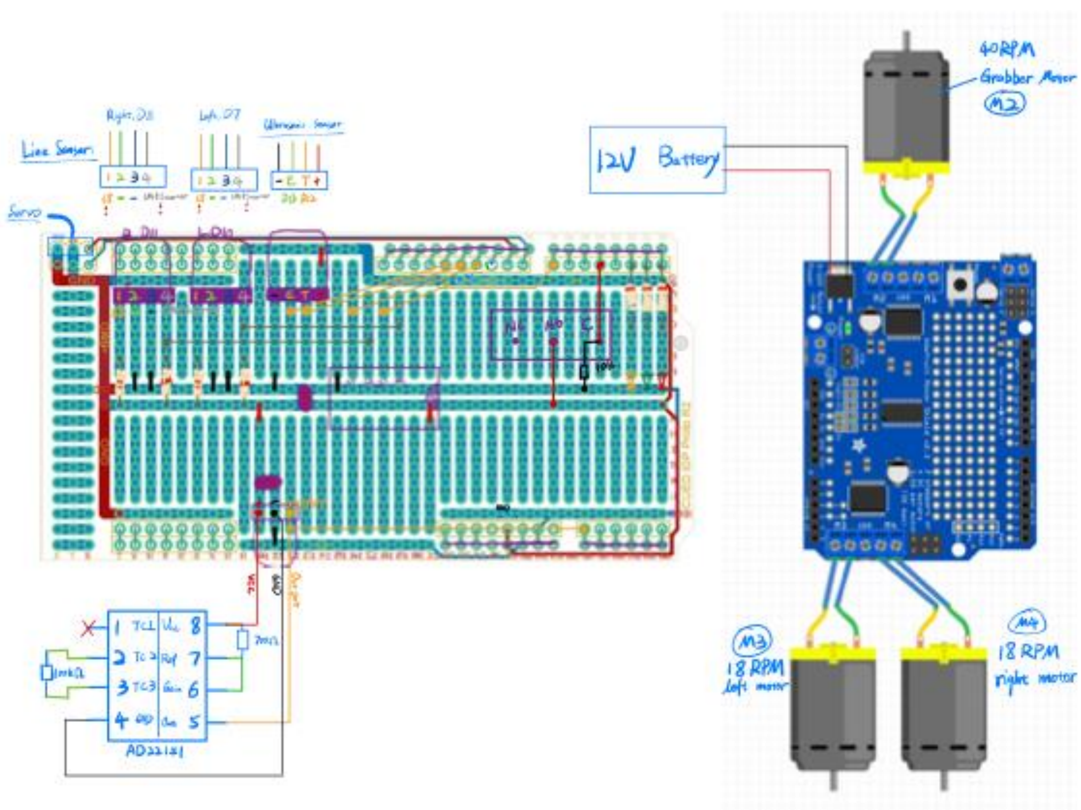


[M4: Torque

[E1: Overall schematic diagram]



[E2: Overall layout]



[S1: Flowchart of software]