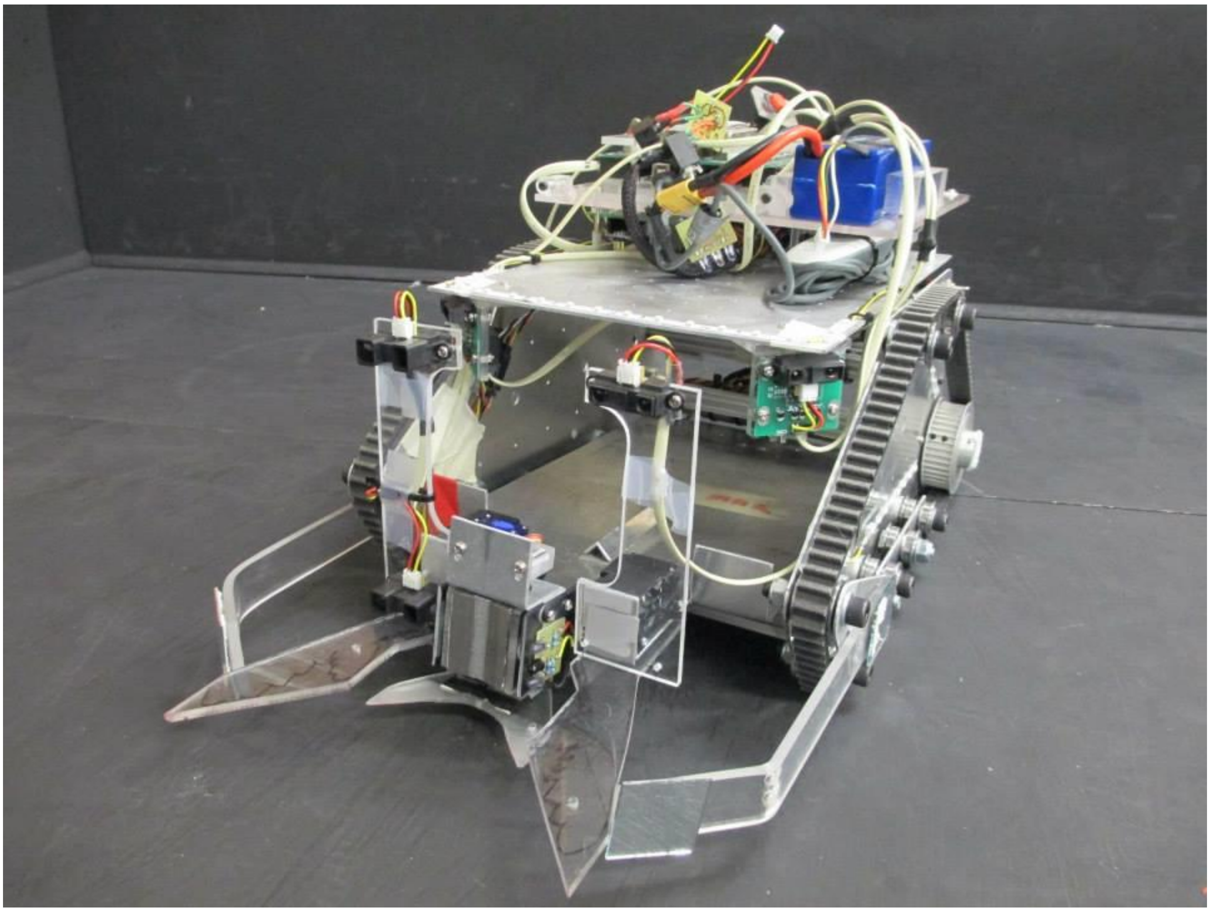
Robocup Design Evaluation Report

Group 8



Matthew Young [redacted]

Hamish Black [redacted]

Chris MacEwan [redacted]

1 Executive Summary

This document, the Evaluation Report (ER), details the progress of Group 8 in the 2014 Robocup. The goal of the Robocup was to develop a robot capable of autonomously navigating an arena to find and collect packages of varying mass. Robots competed with a robot from an opposing team to collect the most mass in 5 minutes. The preceding reports to this one; the Conceptual Design Report (CDR) and Detailed Design Report (DDR) describe the conception and development of a design based around a mechanically operated magnet.

The chosen design consists of 4 main mechanisms which fulfil dedicated tasks. At the most basic level, the robot must be capable of identifying, collecting and storing packages whilst navigating obstacles. The collection mechanism consists of the magnet, servos, package funnel and storage bay which guide packages towards the magnet, then collect and store them. The navigation mechanism is a sensor suite composed of a variety of IR distance sensors and a photodiode array. These sensors all work in tandem to detect walls, obstacles and packages. Specifically, a series of IR sensors are used to detect the presence of obstacles and the distance to them. The photodiode array is used to detect packages directly in front of the magnet. The locomotion mechanism is entirely comprised of the tracked chassis, upon which every other component of the robot is mounted. The chassis forms the base of the robot and allows it to move around the arena as well as navigate low obstacles. The final mechanism is encompassed by the Arduino and all the software loaded on it. The software coordinates all the various components of the robot, processing data from the navigation mechanism and using it guide the robot's movement as well as actuate the collection mechanism when a package is within reach.

This design proved to be extremely successful, coming second place in the Robocup competition. The success of the robot is attributed to a robust physical design, reliable package detection, good quality software and a collection mechanism requiring low precision to be successful. By comparison, the opposing robots encountered problems when they became stuck on obstacles, could not collect packages or failed to detect packages altogether.

This report provides a detailed evaluation of this robot, as well as a brief evaluation of some of the opposing robots. The evaluation discusses the design and qualities of each robot, and how effective each design was in the competition. A quantitative assessment of some of their similar properties such as speed and detection range has also been provided.

Contents

Table of Figures

Table of Tables

2 Introduction

Every year, 2nd Pro Mechatronics students at the University of Canterbury compete in the Canterbury RoboCup Competition. Students are split into teams of three, trying to design a robot to complete a specific challenge. This year the scenario was search and rescue mission in the wake of a zombie apocalypse. Two teams battle head to head within a "city" scattered with "food packages" over the course of 5 minutes. Each "food package" was represented by metal cylinders of varying mass between 0.5 and 1 kg. Once found, the robot had to collect the package and store it on-board or return it to their respective teams "HQ", represented by a coloured section of the arena. Once all packages were collected or the time limit was up, the robot that had collected the most cumulative mass won. Each team faced off against each other in a round-robin style competition, with the winners facing the winners of other rounds and the losers facing the losers of the other round.

The "city" was an arena 2.4 x 4.9m in size with 400mm high walls surrounding it. The arena contained a number of obstacles scattered throughout with the final location of each not being revealed until the day of the competition. The obstacles included <25mm high ground obstacles such as include ramps and speed bumps that the robot should be able to traverse, and red obstruction obstacles such as cylinders and walls which were >400mm high that the robot should manoeuver around. Each Team had its own "HQ" 300 x 400mm in size within the arena surrounded by a ~10mm high rim to prevent any deposited food packages from rolling out of the base.

The "food packages" all looked outwardly identical, a golden steel cylinder of diameter 50mm and height 70mm with an annular groove to facilitate gripping. Each package was either 0.5, 0.75 or 1.0kg in weight with the location of each package changing between rounds.

This report was preceded by the Conceptual Design Report (CDR) and the Detailed Design Report (DDR). The CDR outlined the specifications and performance requirements for the robot, as well as the initial design concepts. The DDR detailed the development of the robot design and compared its current performance the criteria in the CDR. This report continues on by describing and evaluating the final design. This report also evaluates four other teams' robots, and compares their performance with that of our robot (Group 8). The final design has been split into the defining functional mechanisms: collection, navigation, locomotion and software. The evaluation covers the features that contributed to the robots success in addition to any features that didn't work as intended or could be improved. It also compares the performance of the robot to the original requirements laid out in the CDR, and justifies any disparities between them. In addition to evaluating our robot, this report also evaluates four competing robots, and the various features made them successful or unsuccessful.

3 Design Description

3.1 Overall Description

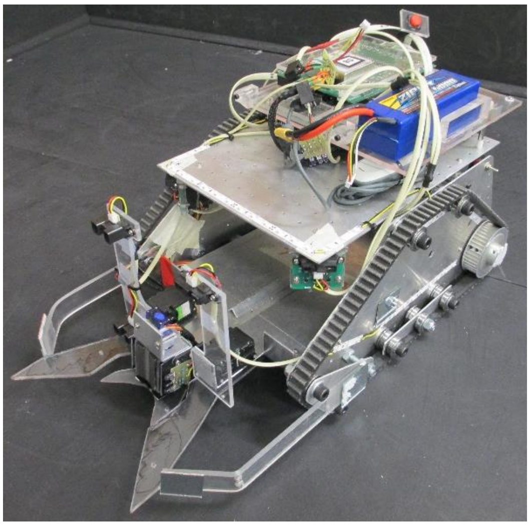


Figure 3.1: Final Robot Design

The robot's basic behaviour was to do a random walk around the arena, alternating between driving forwards and spinning on the spot until a package was detected. When an obstacle was detected using any of the IR distance sensors, the robot would turn away from it and continue wandering. If a package was detected, the robot would stop and drive forwards until the robot automatically initiated package collection. The method of collection involved funnelling packages towards a magnet mounted on the front of the robot. When a package was detected directly in front of the magnet, the magnet turned on, attached itself to the package and then rotated the package into the storage bay inside the body of the robot. Once a package had been collected, or an obstacle detected and avoided, the robot would continue to wander the arena. Refer to Appendix B for a labelled diagram of the robot.

The overall robot can be broken down into four mechanisms containing the following components:

1. **Collection:** magnet, perspex funnel, package sensor and storage bay.
2. **Navigation:** IR distance sensors.
3. **Locomotion:** chassis, track extensions and additional idlers.
4. **Software:** all software related to navigating the robot, detecting packages/obstacles and driving the magnet.

3.2 Collection Mechanism

The collection mechanism on the final design used the magnet and a minimised version IR array as outlined in the DDR. The magnet was kept in a slightly elevated position to avoid scraping the bottom of it on obstacles, the 'home' position. An IR sensor was mounted on the bottom of the magnet to detect packages directly in front of the magnet. When a package was detected the magnet was lowered and engaged while the robot drove forward slightly, thereby helping to push the package onto the magnet. After a short delay, the magnet would then rotate into an upright position just above the storage bay. The magnet was then disengaged, allowing the package to slide off into the storage bay. After another delay, the magnet was returned to its original position. This process is summarised in Figure 3.2, which was implemented in software to control the magnet.

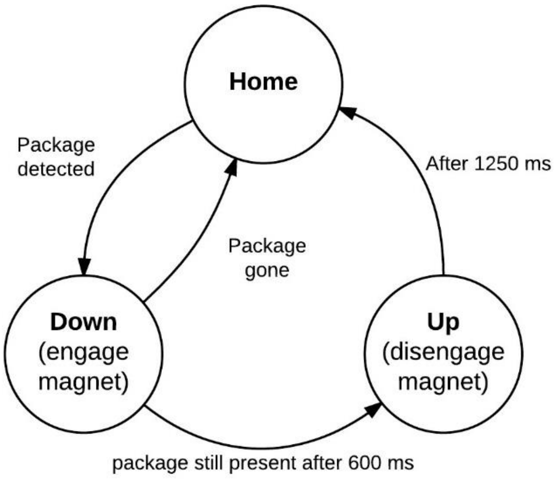


Figure 3.2: State Diagram for Magnet Software

The mechanical elements of the design such as the perspex funnel and the storage bay, did not interface with the Arduino at all and are passive elements of the overall design intended to increase the effectiveness of the pickup mechanism and reduce software complexity.

One addition to the robot not covered in the CDR or DDR was the addition of Perspex 'arms' on the front of the robot and smaller ones on the underside of the magnet. After the results of the second functionality assessment, it became evident that the computer software and navigation methods were not precise enough to guide the robot directly towards a package. As a result, when a package was detected, it was often off centre and the collection mechanism would fail to collect it. To correct this, the Perspex arms were added to funnel any packages in towards the magnet, removing the need for precision in navigation.

In addition it was found that the robot occasionally failed to detect a package, but still managed to move close enough to run it over, knocking it over in the process. This was considered undesirable as the robot was no longer capable of collecting packages on their side. The addition of more Perspex shields in front of the tracks allowed the robot to push any errant packages to the side rather than running them over. These arms were not part of the original design or the DDR.

The original plan in the DDR also included IR arrays mounted on the side of the magnet, as well as underneath it. The intention was to use these arrays to detect off centre packages and correct the robots movement accordingly. However this idea was abandoned after the introduction of the Perspex arms which were much more effective made them redundant. Instead, a single IR sensor is used in order to determine if a package is directly in front of the magnet assembly. This is termed the package sensor and is used to trigger the state transitions shown in Figure 3.2.

3.3 Navigation Mechanism

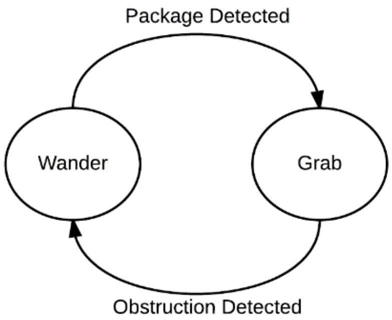


Figure 3.3: State Diagram for Navigation Software

The method of navigation used in the final design was a vastly simplified version of the original plan laid out in the DDR. This was in part because it was realised that the same effect could be achieved with a simpler design, and because of difficulties encountered with implementing navigation methods originally envisaged such as using the ArduCam for computer vision.

The navigation system in the final design was composed of five IR distance sensors. Two short range sensors were mounted on the sides and aimed at 45° from the front of the robot to detect any obstacles to the left or right. Two long range sensors were mounted on the front, with one sensor mounted at the same height as a package, and the other mounted 10 cm directly above the first. These sensors were used to detect any packages in front of the robot, as discussed in the section below. A final medium range IR sensor was also mounted on the other side of the front of the robot, purely to eliminate the blind spot in the robot's vision. When a sensor detected an object within a threshold distance, the robot would take an action described in Table 3.1 until the obstruction was no longer present.

Table 3.1: Actions Taken in Response to IR Distance Sensor Readings

Sensor Reading too close	**Action Taken**
Left and Right	Drive Back.
Left	Turn CW.
Right	Turn CCW.
Front *Only*	Sensor Closer on Left? Turn CW. Sensor Closer on Right? Turn CCW.

3.3.1 Package Detection

The package detection system is used to determine whether the robot is pointing directly at a package at any time during the round. The system utilises a pair of long-distance infra-red distance sensors mounted approximately 10cm above each other, both pointing in the same direction. The readings from both sensors are compared, and if an object appears much closer to the lower sensor, it can be inferred that the robot is pointing at a weight. This information is then passed to the navigation task which instructs it to transition from the *Wander* to the *Grab* state. Figure *3.4* illustrates the sensor pair detecting a weight.

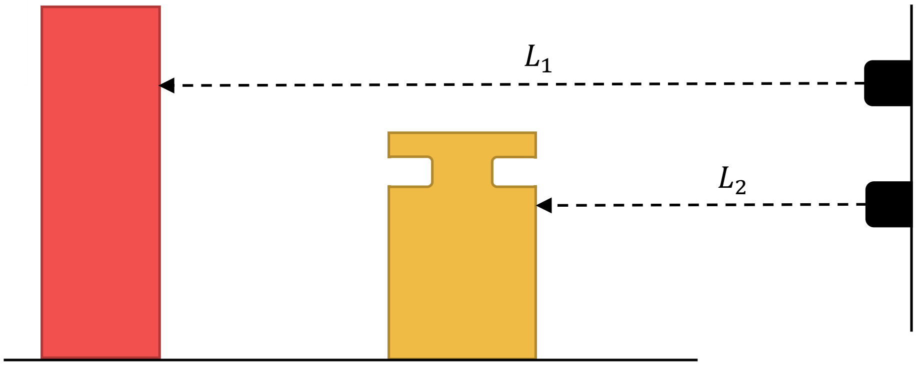


Figure 3.4: Detecting a Weight Using a Pair of IR Distance Sensors

3.4 Locomotion

The locomotion mechanism consists entirely of the chassis and two 28PA51G Low noise DC motors, which were used to move the robot and navigate obstacles.

To increase the amount area available to store the packages, the provided chassis was rearranged in an up-side down configuration with its long side on the bottom. This was a trade-off, as this had the effect of increasing friction between the tracks and the ground, reducing the ability of the robot to steer as the weight increased.

The original chassis configuration had two problems: the robot could not longer climb over ground obstacles that were up to 25mm high due the track sloping diagonally backwards from the front of the robot, and the cross members and side plates of the chassis would 'beach' the robot if driven over the ground obstacles due to the tracks only providing 3mm of clearance. To solve these issues, extensions were designed to add to the front of the robot that sloped the tracks diagonally forwards of the robot, rather than backwards allowing the robot to drive forwards over obstacles 25mm and greater. An extra roller was added between the two bottom middle rollers to ensure the tracks did not deflect upwards and lose the little clearance the tracks provided for the side plates. The location of the two bottom cross beams was changed to solve the beaching issue. The front beam now clears 25mm above the ground and has the storage plate and mount for the collection mechanism fastened to the top of it. The rear bottom cross beam was raised to 45mm above the ground, however to allow for the sloping of the storage tray towards the back the tray was fastened to the bottom of the cross beam at height of 25mm from the ground.

3.5 Software

The software for the robot uses an open source real-time operating system called ChibiOS. It supports a wide range of hardware platforms and is well supported on the Arduino Mega ADK. It provides a range of services which have made the software design for the robot much more simple and modular. The project makes heavy use of the built-in scheduler in order to ensure that all tasks run at the desired rate. This has been shown to work well in practice and all versions of the robot software have used this operating system without encountering any difficulties.

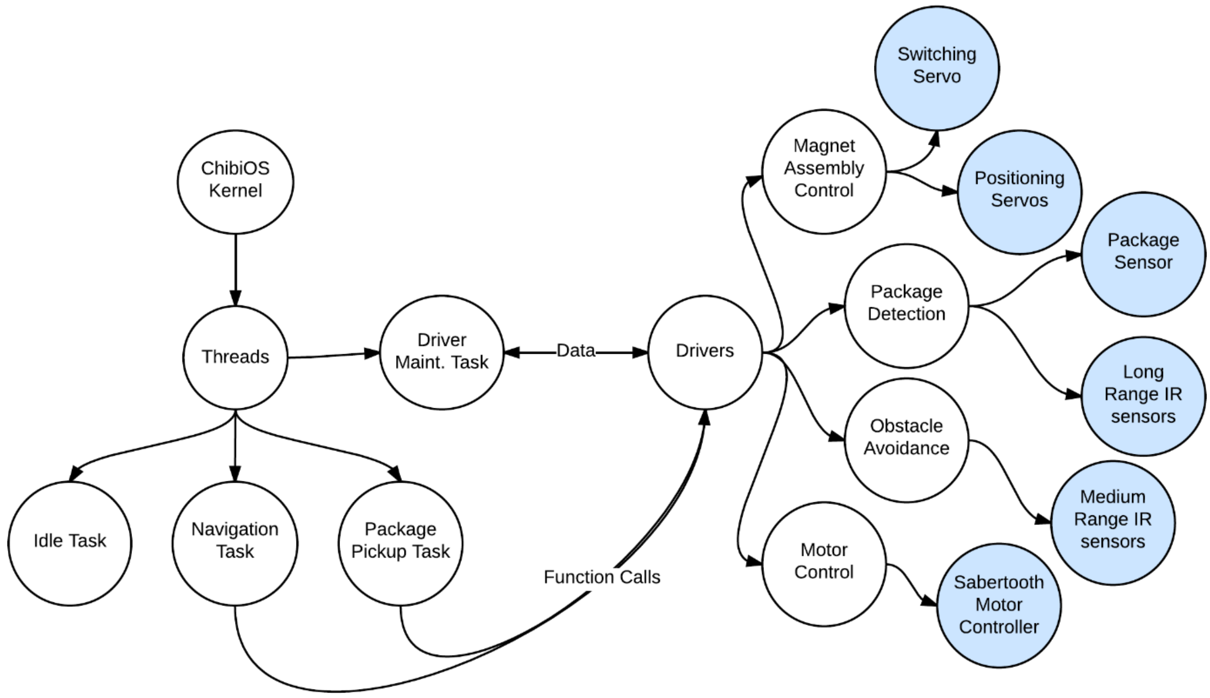


Figure 3.5: Robot Hardware and Software Architecture

The program is split in to three main types of subsystems.

3.5.1 Navigation and Package Pickup tasks.

These tasks are responsible for all high-level decision making the robot needs to make. It never interfaces with the hardware directly and instead controls the robot by making calls to the different drivers in order to read from sensors or control actuators. By processing this information, these tasks can determine the appropriate course of action and instruct the drivers to carry this out with minimal code.

The strategies used for navigation and package pickup are outlined in Sections 3.3 and 3.2, respectively.

3.5.2 Drivers and Maintenance Task.

Drivers form the Hardware Abstraction Layer (HAL) of the robot. They provide the interface between the hardware and the other tasks. They expose a simple interface and allow for the caller to read or set a value without needing to understand the underlying hardware. This is advantageous because it keeps all code for a specific sensor in one place, reducing the incidence of shared data problems and increasing the maintainability and simplicity of the overall program.

This approach was very helpful for controlling the smart servos used to move the magnet assembly. The magnet assembly was routinely moved to three different positions as part of the package pickup sequence and the smart servo requires an angle and movement time specified when commanding it to move. Both the home and down positions were very close to each other a movement time of 1 second would be too slow, while it would be the correct speed for moving from the down to up position for example. This meant that the movement time needed to be adapted based on the magnet assembly's current position. By abstracting away this logic, it could be hidden from the package pickup task and it only needed to command the drivers to move the magnet assembly to the desired position.

The driver maintenance task runs at approximately 100 Hz (the fastest rate at which the sensors update) and polls all ADCs for updated values, before storing it in memory. This means that should the navigation task need to know whether there is an obstruction present it can immediately determine and does not need to wait for the ADC read to finish (this can take some time). This improves the overall performance of the program.

4 Evaluation

4.1 Evaluation of our Robot

Table 4.1: Competition Performance of our Robot

	Opposing Group	No. packages collected		Mass (kg)	
Round		**Group 8**	**Opponent**	**Group 8**	**Opponent**
1	6	7	0	5	0
2	3	7	2	5	1.25
3	9	4	0	2.75	0
4	1	0	7	0	4.75
5	7	1	0	0.5	0
6	1	4	4	2.5	3
Total	-	23	13	15.75	9
Average	-	3.8	2.2	2.625	1.5

Table 4.2: Group 8 Robot Performance Before and During the Competition

Characteristic	During testing	During Competition
Maximum speed (unloaded)	0.27 ms^{-1}	0.20 ms^{-1}
Maximum speed (loaded)	0.26 ms^{-1}	0.19 ms^{-1}
Maximum rotation speed (unloaded)	70 degrees/s	42 degrees/s
Maximum rotation speed (loaded)	53 degrees/s	38 degrees/s
Average mass collection	0.68 Kg/min	0.53 Kg/min
Collection success rate	95%	90%
Battery life[1]	2 hrs	N/A
Maximum package detection range	30 cm	30 cm

4.1.1 Successes

In most cases when a package is detected, the robot has turned slightly past the package and is not pointing directly at it. This would lead to a failed pickup should the robot drive forward and attempt collection. Two measures were used in order to compensate for this. First, the robot will 'kick back' in the opposite direction to the turn in order to increase the likelihood the robot is pointing in the correct direction; and second, Perspex arms have been added to the front of the robot in order to funnel packages towards the pickup area. These two measures combined created a very effective package pickup system that was able to work over 90% of the time, meeting the original requirement laid out in the first report. Adding the arms was however a trade-off, as the robot was no longer able to pick up toppled packages or those far in corners.

The additions to the chassis that provided the improved track layout proved very successful before and during the competition. They allowed the robot to traverse every ground obstacle during the competition, only getting stuck once for ~5 seconds until it freed itself.

[1] Note that the battery life assumes continuous movement for the duration of the battery life, and that the battery life was not exceeded during the entire competition.

4.1.2 Failures

Adding the arms make it very difficult for the robot to pick up weights close to the moveable barriers in the arena. This wasn't an issue at first, however when the arena layout changed to one where most of the weights were kept close to the barriers, our robot completed a round without picking up *any* weights.

4.1.3 Unmet Requirements

There were many requirements outlined in the CDR that were not met in the final design for a variety of reasons. The requirements that were not implemented due to time constraints include:

> **THE ROBOT SHALL BE CAPABLE OF RESETTING ITSELF IN CASE OF A FAULT.**

During rounds 3, 4 and 5 the robot came across two different situations that it needed to reset itself. In round 3 the robot got caught on the opposing teams robot for ~40 seconds, then got stuck trying to pick up a cylindrical obstacle until the end of the round. In round 4 the robot tried to pick up a wall obstacle for ~30 seconds until it managed to drive out of it by force, not by evasion. In round 5 the robot got stuck again trying to pick up a cylinder obstacle until the end of the round.

> **THE ROBOT SHALL MAINTAIN KNOWLEDGE OF ITS POSITION IN THE ARENA AT ALL TIMES.**

As the project progressed, it became apparent that the navigation methods originally envisaged were not necessary. The robot did not need to plan routes throughout the arena, as mission objectives could be completed by doing a 'random walk'. The implementation of a random walk did come with issues however, as the robot would sometimes stay in an area that was depleted of accessible weights, something this requirement was originally intended to prevent. This issue could be fixed by changing the times the robot would spend driving forward vs turning to favour driving forward for longer periods of time.

> **THE ROBOT SHALL MAINTAIN A COUNT OF THE CUMULATIVE WEIGHT OF FOOD PACKAGES ABOARD THE ROBOT.**

This was not a major design factor in the end. The robot did not stand to gain anything useful by measuring total on-board weight, besides the detection of a successful weight collection – something which could be measured through other means. For all intents and purposes, a successful weight collection was defined in software as the robot completing the package pickup sequence.

> **THE ROBOT SHALL BE CAPABLE OF PICKING UP FOOD PACKAGES WHICH ARE PLACED AGAINST ARENA WALLS OR OBSTACLES. THIS INCLUDES FOOD PACKAGES IN THE CORNER.**

With the addition of the Perspex arms the robot could no longer collect any weights in the corners. There was limitations to the robots ability to collect weights close to the obstacles and walls but this was due to the threshold values of the IR sensors, rather than the arms. These threshold values could have been lowered slightly to allow a greater chance of weight collection in these instances but drawbacks may have been the robot hits the obstacles to frequently and damages itself.

4.2 Evaluation of Group 1 (Scott, Zane)

The basic design of this robot involved three sections of Perspex, cut so that the distance between them was approximately equal to the diameter of the annular groove cut into each package. Simply running over an upright package caused the package to move up and onto these Perspex rails, which then guided the package into the robot. Two 'flippers' on the front of the robot also helped push the packages onto the rails. Other notable features of the design include the large 3D printed wheels and the computer vision system used for navigation and package detection.

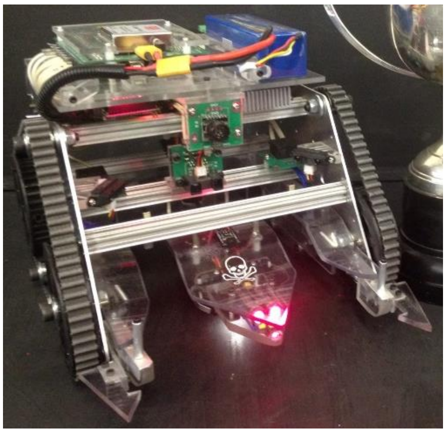


Figure 4.1: Group 1's robot

The robot navigated the arena by following walls when it was close to them, and turning 110° randomly left or right every 10 seconds. When an obstacle was detected, the robot moved in the opposite direction to avoid it. If a package was detected, the robot would drive straight at it, effectively running it over in order to collect it. Once collected, the robot would move a short distance past it to confirm that it had been collected. Once the package had been successfully collected the robot would resume wandering around the arena.

This design was very successful for a variety of reasons. The large wheels drastically increased the speed of the robot, allowing it to cover more area and reach more weights than any of the opposing robots. And because of the nature of the collection mechanism, the robot never had to stop moving when collecting packages, meaning that from the start of the round, the robot never stopped moving, giving it an additional speed advantage over the competition. The computer vision system was a very effective method of detecting packages because it could detect packages in a 30° arc in front of the robot, as opposed to IR sensors which could only detect objects in a narrow beam directly in front of them.

One major drawback to the design was that the robot could not pick up packages in any sideways orientation. However all of the packages in the competition were placed in the upright orientation, and packages were very rarely knocked over in any of the rounds so this issue never became a problem. As a result, this robot was able to move faster, and pick up packages more accurately than any opposing robot.

For a quantities breakdown of this robot's speed and accuracy, refer to Table 4.3.

4.3 Evaluation of Group 7 (Kieran)

The basic design of this robot involved a metal plate which was laser cut in order to create a rack which metal packages could be lifted off the ground and stored. In order to perform the package collection, the robot would first stop and then use the servo arm to push the packages on to the rack. This was in contrast to group 1's robot which would use the momentum of the robot moving over the package coupled with a bump from flippers in order to send the packages on to the storage rack without needing to stop.

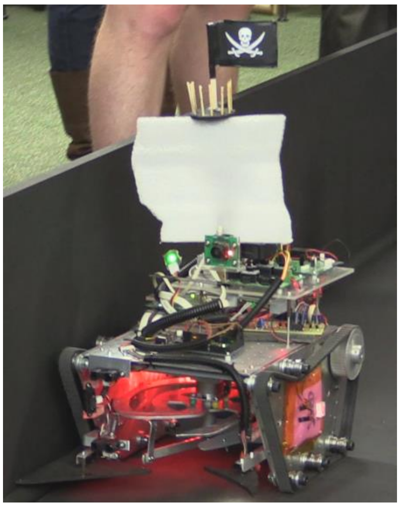


Figure 4.2: Group 7's robot

The robot navigated the arena by driving forwards and making turns at random times. This allowed it to cover a lot of ground while reducing the risk that the robot became stuck in any one part of the arena. Infra-red distance sensors and sonar were used to avoid obstacles simply by turning away from anything that got too close and package detection was performed in a similar manner to our robot, utilising a pair of long-distance IR sensors placed above each other and comparing the measurements.

This design was successful, managing to achieve 3rd place on the day. The pickup mechanism was able to pick packages up on the first attempt 90% of the time and was not materially affected by driving over toppled packages due to the large ground clearance unless the weights got caught under the tracks.

However this robot did suffer from a few issues that plagued it on the day. These were known to the team, but there was not enough time to address them all before the competition. The robot needed to stop for approximately 10 seconds in order to pick up packages. In a competitive environment, this was a huge amount of down time and a software update which would resume movement as soon as the weight was lifted off the ground may be worth investigating. This would limit down time to a few seconds at most.

The robot was liable to tip over backwards if a weight got caught under the tracks. This issue was fixed on the right hand side of the robot with a larger Perspex bumper, but not on the left. This scenario occurred when the robot was competing against our group, knocking it out of the competition.

4.4 Evaluation of Group 3 (Daniel, Danny, David)

This was one of two robots to attempt to pick the packages up from above using a crane design. The collection mechanism used an electromagnet mounted at the end of an arm actuated by a stepper motor. The robot would drive up until it was directly in front of a package it was attempting to collect, at which point it would lower the crane arm, activate the electromagnet and slide open a receiving tray and deposit the magnet.

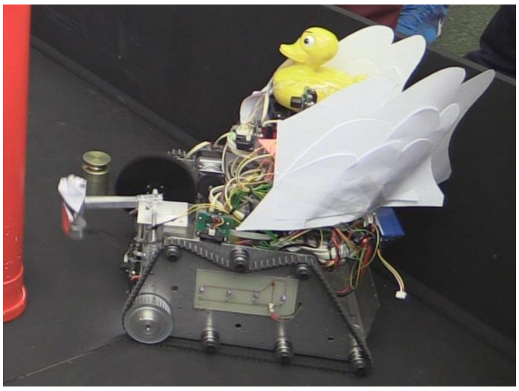


Figure 4.3: Group 3's robot

The navigation and package detection algorithms used were much the same as other groups.

This design was moderately successful, managing to collect 1.25kg in its first round though it was unable to win any. It was one of three robots which did not need to drive in to weights to pick them up, something which could have made the robot much more competitive depending on the weight placement.

A few improvements could have been made in order to make the robot much more effective. Much like our robot, the precision alignment the collection mechanism required was an issue. An attempt was made to mitigate it with extra sensors, though the simplest fix was to use 'arms' mounted on the front of the robot in order to shepherd packages towards the collection area. This is a passive design feature and would increase the collection success rate, while reducing the complexity of the required software. It would reduce the ability of the robot to pick up obstacles hard up against walls, though this may be considered an acceptable trade-off.

Another improvement would be to use long-range instead of medium-range sensors for package detection. This unnecessarily limited the range at which the robot could sense obstacles to approximately 50cm. Simply changing the sensors would double the range, ensuring that the robot could locate packages much more quickly.

4.5 Evaluation of Group 6 (Tim, Anthony, Jie Hong)

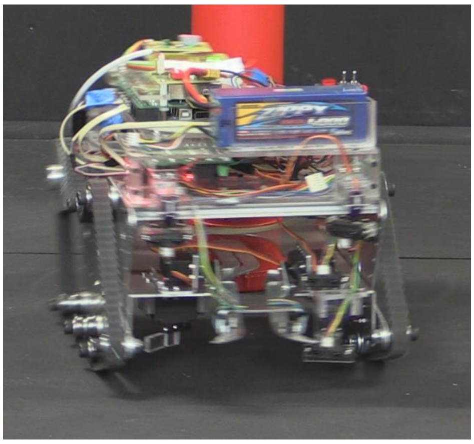


Figure 4.4: Group 6's robot

Like many other groups, this robot collected packages by engaging a set of forks with the annular groove cut into the packages, similar to a fork-lift. The forks then lifted the package up, making it slide onto a holding rack.

For package detection this group used two IR sensors, one mounted directly above the other, with the bottom sensor at the same level as packages, and the top sensor just above them. For navigation, more IR sensors were used to detect walls and other obstacles. In addition, all IR sensors on this robot were mounted on servos, allowing the sensors to sweep back and forth across the front of the robot, vastly expanding the robots field of vision. This robot used the default provided chassis, with a slight modification to extend the front tracks forward and upward.

This robot had the same basic behaviour as the other robots in the competition, wandering randomly around the area, and picking up packages when it spotted them. The major point of difference was that the sweeping sensors gave it a much wider field of vision, allowing it to detect packages not directly in front of the robot.

This effectiveness of this robot is hard to judge because the robot became physically stuck in a couple of rounds of the competition. So the physical design was not able to navigate obstacles in all instances. Another flaw in the physical design was that successful collection required the package to be directly in front of the forks on the front of the robot. And while the hardware and sensors were in place to detect the package, the software was not precise enough to guide the robot directly and accurately to the packages. Unlike our robot, there were no appendages on the front of the robot to help funnel packages towards the collection mechanism. This meant that although the robot could detect packages more successfully than many other robots, it struggled to successfully collect many packages.

4.6 Quantitative Evaluation of Competing Robot Designs

Although each robot had different designs, there are some basic qualities of each robot that can be measured and compared, which have been provided in Table 4.3.

Table 4.3: Performance Comparison of Evaluated Robots

Characteristic	Group 8	Group 1	Group 7	Group 3	Group 6
Maximum speed	0.24 ms^{-1}	0.62 ms^{-1}	0.41 ms^{-1}	0.22 ms^{-1}	0.31 ms^{-1}
Maximum rotation speed	42 degrees/s	81 degrees/s	63 degrees/s	53 degrees/s	60 degrees/s
Average mass collection	2.5 kg	3.7 kg	0.8 kg	0.75 kg	0.5 kg
Collection success rate[2]	90%	100%	90%	56%	25%
Max package detection range	60 cm	50 cm	1.2 m	30 cm	60 cm
Storage capacity	10	9	9	6	9

[2] Note that the collection success rate only considers the packages that were successfully or unsuccessfully collected once the respective robots collection mechanism had been deployed.

However, this assessment is not an accurate depiction of each robot's performance, as the quality of software was often the deciding factor that determined if a robot got stuck, malfunctioned or failed to collect a package. Since software quality is not a measureable quality of a robot it has been omitted from Table 4.3, but should still be considered a dominating factor in each robots performance.

5 Conclusions

With its current design the robot functions as intended, albeit with a simplified design when compared to what was originally envisaged. Any improvements to the current design would simply aim to increase the speed and precision of the navigation and collection mechanisms, perhaps using additional sensors to obviate the need for the Perspex arms and restore the ability for the robot to pick up weights close to obstacles or in the corner.

Our robot suffered significantly from blind spots, becoming stuck in an endless pickup loop at one point as it was unable to detect it was driving in to an obstacle. By mounting the sensors on servos and panning them back and forth like Group 10 did, blind spots could be eliminated and the robot would not have suffered from this issue. This would be worth considering implementing in the future.

One of the advantages of our development was that our robot had gone through four different iterations of the current design. Each iteration was tested, and each time a flaw or problem was discovered it influenced the next iteration of the design. This meant that our final design was very robust and efficient, and did not encounter any mechanical problems. By comparison, for many opposing groups, their final design was the first design and only design that they had implemented. This was mostly due to a lack of time and poor management, which were problems that we avoided by starting our mechanical design much earlier than most other groups. As a result, we were able to optimize our design to a much greater degree than many of the other groups, who often didn't have the time or resources to fix any problems they encountered in their design.

6 Contribution Statement

6.1 Matthew Young

Report writing	PCB design and production	Magnet mount design
Software development	Chassis design	Software debugging
Chassis modification	Sensor arrangement	Hardware re-arrangement
Software testing	Storage plate design	CAD modelling

6.2 Hamish Black

Report writing	PCB design and production	Magnet mount design
Software development	Chassis design	Software debugging
Chassis modification	Perspex funnel construction	Storage plate construction

6.3 Chris MacEwan

Report writing	PCB design and production	Light display development
Software development	Chassis design	GitHub setup
Software debugging	IDE environment setup	Software architecture
Software driver development	Package detection algorithms	Sensor calibration
IR circuit design	Component acquisition	

7 Appendices

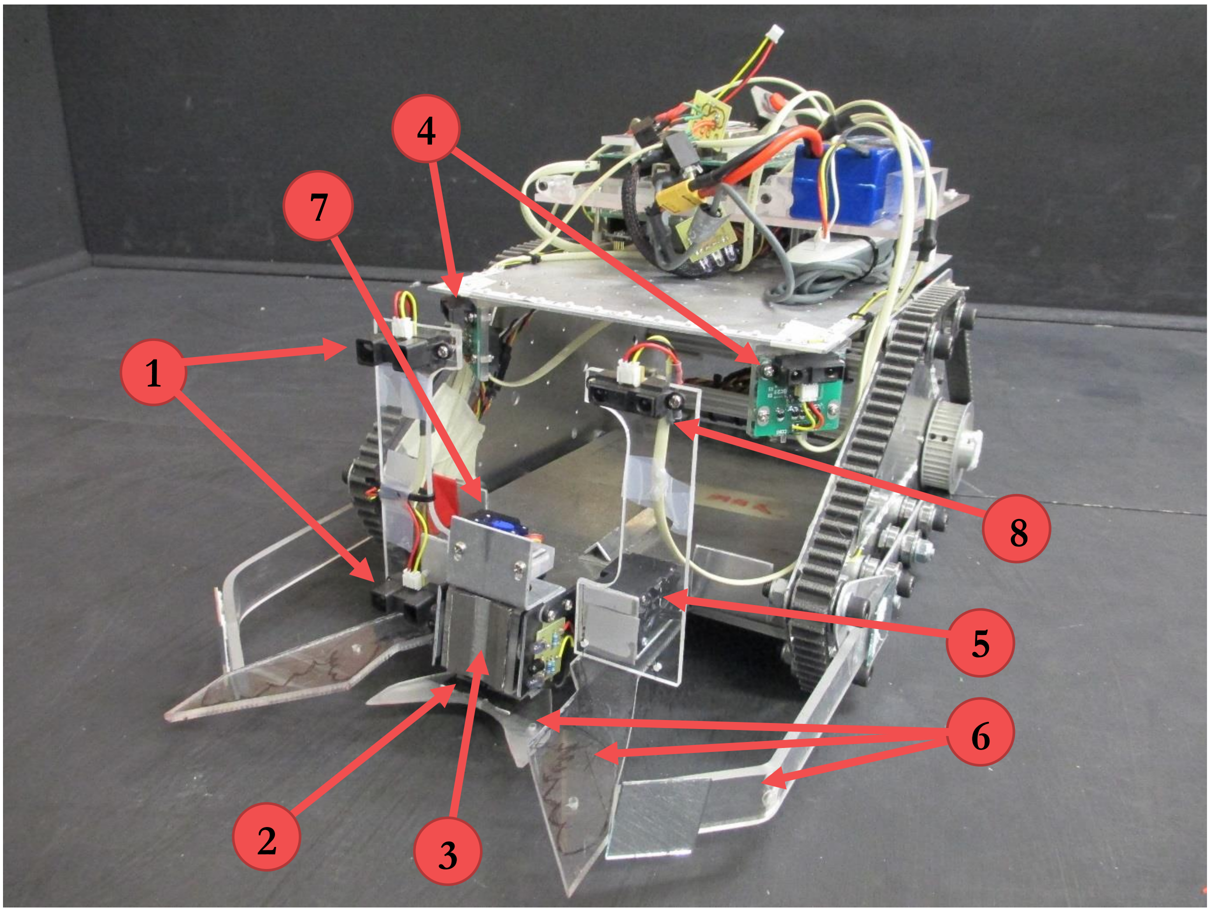
Appendix A: Bill of Materials

This list contains all the significant components currently present as part of the final design. All components are assumed to have been purchased using retail prices rather than wholesale prices. Where no price for aluminium parts has been provided, a price of 5c per gram has been applied to find the total price for the component.

Part description	Supplied	Manufacturer/ supplier	Part #	Qty	Cost	Total cost ($NZD)
Structural Components						
Chassis tank track	Y	Bestorq	880-8M	2	10.12	20.24
Chassis body	Y	Ullrich	N/A	2	18	36
Chassis cross beam	Y	Ullrich	N/A	5	3.25	16.25
Chassis extension inner plate	N		N/A	2	4.7	9.4
Chassis extension outer plate	N		N/A	2	1.75	3.5
Perspex 300*300*5	Y	Dotmar	N/A	2	6	12
Electronic Boards						
Arduino Mega ADK	Y	Arduino		2	120	240
I/O board	Y	UC		1	40	40
Dual 12A motor driver	Y	DFRobot	DRI0003	1	96	96
Power regulator	Y	UC		1	12.9	12.9
Battery protection	Y	UC		1	14	14
ArduCam	Y	Arduino		1	36	36
Sensors						
IR long range dist sensor	Y	Sharp	GP2Y0A02YK0F	2	17	34
IR medium range dist sensor	Y	Sharp	GP2D12	2	17	34
IR short range distance sensor	Y	Sharp	GP2D120	1	18	18
Servos						
Smart servo	Y	Herkulex	DRS-0101	2	50	100
Micro servo	Y	Herkulex	HXT900	1	4	4
12V low noise DC motor	Y	DFRobot	28PA51G	2	53	106
Standard servo	Y	hexTronik	HX12K	2	12	24
Fasteners						
M3 5 mm screw	Y	Mitre 10	N/A	4	0.3	1.2
M3 10 mm screw	Y	Mitre 10	N/A	26	0.3	7.8

M3 15 mm screw	Y	Mitre 10	N/A	12	0.3	3.6
M3 20 mm screw	Y	Mitre 10	N/A	8	0.3	2.4
M8 40 mm hex head bolts	Y	Mitre 10	N/A	14	0.725	10.2
M8 1.5 mm thick washer	Y	Mitre 10	N/A	60	0.25	15
M8 2 mm thick washer	Y	Mitre 10	N/A	20	0.28	5.6
M8 hex head nuts	Y	Mitre 10	N/A	20	0.12	2.4
Miscellaneous						
Magnet	N	Unknown	N/A	1	9	9
Smart servo arm	Y	Herkulex	N/A	2	0	0
M8 Bearings	Y	DFRobot	608RS	24	1	24
Drive wheel	Y	Unknown	N/A	2	10	20
Switch	Y	SonarPlus	ST0335	1.8	2	3.6
IR diode	N	RS Components	665-5322	2	0.295	0.59
IR Photodiode	N	RS Components	699-7600	1	3	3
					TOTAL	**965.98**

Appendix B: Diagram of the robot with labelled components



1. Long range IR sensors used for package detection.
2. IR array mounted on the underside of the magnet.
3. Magnet used for packaged collection.
4. Short range IR sensors for detecting obstacles.
5. Herkulex smart servo for rotating magnet.
6. Perspex arms for funnelling packages.
7. Micro servo for turning magnet on and off.
8. Medium range IR sensor for obstacle detection.