UNIVERSITY OF CAPE TOWN

DEPARTMENT OF MECHANICAL ENGINEERING

RONDEBOSCH, CAPE TOWN, SOUTH AFRICA

Development of a Low-Cost, Mid-Sized, Tele-Operated, Wheeled Robot for Rescue Reconnaissance

Author: **Matthew Wilson**

Supervisor: **Mr Stephen Marais**

Date: **1 October 2013**

MEC4110W Individual Project

Project Number: **49**

Abstract:

A need arose, in recent years, for low-cost tele-operated robots for search and rescue operations in confined spaces such as collapsed buildings. This project was to design, build, and test a mid-sized, man-portable robot that fulfilled this need and could, when required, climb stairs.

To address this challenge, a robot was designed that made use of a passive obstacle climbing system. This offered a high degree of driving freedom while keeping the number of motors minimised and the overall profile of the robot very low for entering low voids.

The robot was equipped with four cameras operating in stereoscopic pairs, sending back “3D” live video footage via a single analogue transmitter. Control was achieved using a PC and a user interface that included a common game controller for joystick input. The PC and the robot's onboard microcontroller communicated wirelessly.

Unfortunately, complexities in the assembly of the robot caused delays which meant comprehensive testing could not be completed. However, the tests that were completed indicate the only overall shortcomings were a severe difficulty in turning and a lack of torque.

This report details the above-mentioned development process.

Declaration

1. I know that plagiarism is wrong. Plagiarism is to use anotherâs work and pretend that it is oneâs own.
2. Each significant contribution to, and quotation in, this assignment — "Development of a Mid-Sized, Wheeled, Stair-Climbing, Low-Cost Robot for Use in Rescue Reconnaissance" — from the work(s) of other people has been attributed, and has been cited and referenced.
3. This assignment is my own work.
4. I have not allowed, and will not allow anyone to copy my work with the intention of passing it off as his or her own work.

Matthew Wilson, WLSMAT004

Date: October 1, 2013

Acknowledgements

Whilst this is ostensibly an "individual project" there were a number of people who helped to such an extent as almost to be collaborators, and there were many more who also need thanks.

First and foremost, I thank God for the grace and inspiration He gave me to get through this and pull work hours I had never thought possible without my body's collapse. I am grateful He has brought me to this place and kept me creative and joyful in the midst of large amounts of linear thinking and stress.

Secondly, I owe a great deal of thanks to my supervisor, Mr Stephen Marais, not only for academic help in guiding me through this project, but also for practical help with equipment and facilities that I have felt very fortunate to have had put at my disposal.

Then a large thank you and a pat on the back to my fellow two low-cost robotics forth-years, James Stock and Adam Healy. I know none of us would have got as far as we did were it not for shared experience and support.

A special thank you to Thomas Mathew an Greig Knox for practical support sorting out electronics and programming issues, especially on the last day of testing.

I am also most appreciative of the camaraderie in the Robotics and Agents Research Lab — to Kirsty, Roger, Tim, Julian, Angus, Mike, Victor, and Max.

Lastly, a great big "Thank you so much!" to my mom and dad and Cath for moral, emotional, and all kinds of other support on the home front.

Executive Summary

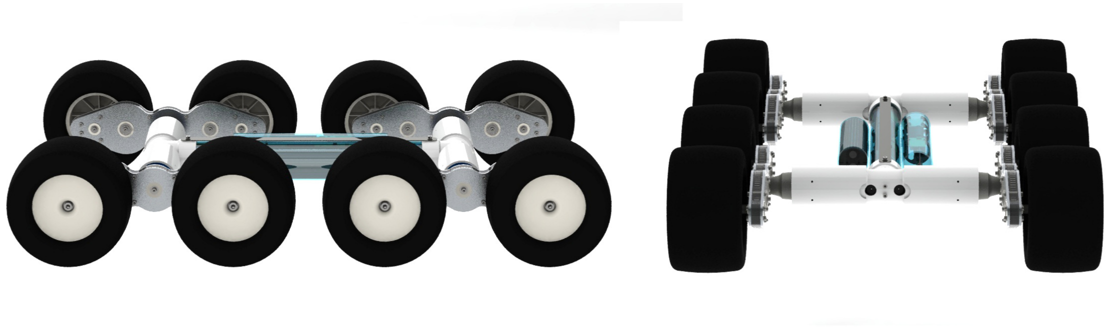


Figure 1: The Robot

A need arose, in recent years, for low-cost tele-operated robots for search and rescue operations in confined spaces such as collapsed buildings. This project was to design, build, and test a mid-sized, man-portable robot that fulfilled this need and could additionally climb stairs.

The principal intended use for the robot was in urban search and rescue (USAR) — the location and extraction of victims from urban disaster zones. The robot was intended for reconnaissance in such conditions, and so had to be rugged, man-portable, manoeuvrable, and remotely operated, while transmitting back a live video feed.

There are a number of robots on the market that can handle this job, but, starting at \$4875, these are by no means low-cost, and many rescue organisations cannot afford them.

To address this challenge, the robot in Figure 1 was designed that made use of a major-wheel-minor-wheel drive system, dubbed the "load-intuitive module" (LIM), that enabled passive obstacle climbing — that is to say, when the robot ran up against an obstacle and the wheels jammed, motor torque would be transferred to flipping pairs of wheels over each other to climb the obstacle. This offered a high degree of driving freedom while keeping the number of motors minimised and the overall profile of the robot very low for entering low voids.

The robot was equipped with four cameras operating in stereoscopic pairs, as in Figure 2, sending back "3D" live video footage via a single analogue transmitter. Control was achieved using a PC and a user interface that included a common game controller for joystick input. The PC and the robot's onboard microcontroller communicated wirelessly.

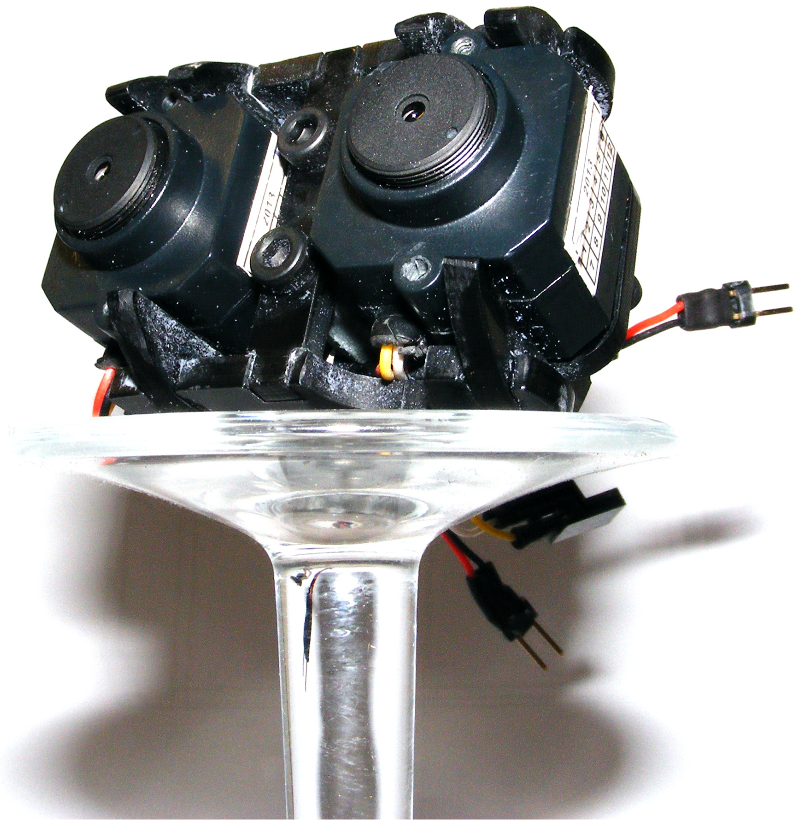


Figure 2: The stereo camera assembly.

Unfortunately, complexities in the assembly of the robot caused delays which meant comprehensive testing could not be completed. However, the tests that were completed indicate the only overall shortcomings were a severe difficulty in turning and a lack of torque. All electronics and programming worked as intended, and given enough current, portions of the drive train were made to climb stair-sized obstacles. The robot was also more expensive than hoped, but still far more cost-effective than even the lowest-cost commercial reconnaissance robot.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

This report details the design, building, and testing of a mid-sized, low-cost, wheeled, stair-climbing robot intended for use in rescue reconnaissance, primarily in urban search and rescue (USAR) operations. This project was under the umbrella of the Low Cost Robotics (LCR) project at the Robotics and Agents Research Laboratory (RARL) at the University of Cape Town (UCT).

Reasons for the Project

As part of the development of a previous robot at RARL, a group of students from the lab attended a conference in Turkey, in 2012, which brought together robot developers and first responders (the teams of people called on for rescue operations in disaster events) to facilitate communication between the two groups to advance the integration of robotics into disaster relief operations. One of the points made by first responders in their presentations was that many of the existing robots on the market were infeasible options for first-responders because they were either too big and awkward to deploy, or too expensive for first-response teams to acquire or risk loosing in structurally unstable buildings. Many were both cumbersome and expensive. What was wanted were small, man-portable, low-cost robots that, while perhaps not doing everything many of the existing robots could, would be easily acquired, deployed, and replaced if destroyed or lost.

Out of this conference, an on-going project was started at RARL in 2013 to develop LCR solutions for search and rescue operations. In general, technological development in South Africa is limited by a lack of research capital and the low availability of leading-edge technology and materials, so it might be said that this project is ideal for the location of this laboratory as there is already a good knowledge pool available regarding attaining design goals with severe resource and technology limitations.

The project detailed in this report was run in parallel with two other undergraduate projects under the LCR banner, each designing a robot to different size and mobility specifications. It was intended that out of this parallel development, much would be learned in reduced time about what might make a commercially viable product.

Figure 1.1: A tornado-devastated neighbourhood in Joplin, Missouri. [2]

Defining the Operating Context

To define the requirements on the robot, there must be an understanding of USAR operations. Typically, USAR operations take place in and around damaged, collapsed, or partially collapsed buildings, where forces of nature (human or otherwise) may have wrought havoc. First response teams and their equipment are be expected to be self-sufficient for up to 72 hours from deployment to a site. [1] Sometimes these sites are not all accessible by vehicle, as can be seen in Figure 1.1, and all rescue equipment (robots included) must be carried to the site by rescue personnel. If no stable opening presents itself, a standard 24" equilateral triangular opening will be cut into a wall or other stable blockage to allow passage of rescuers and their equipment (again, including robots). While it can be expected that there will be some open floor exposed, it is likely that there will be fallen objects and rearranged furniture to be navigated. In multi-story buildings, stairs can be expected, and any lifts[1] or escalators can be expected to be inoperable, if not collapsed. Safe to say, a standard remote-control car is unlikely to be able to get very far. A robot must be able to surmount at least the challenge of stairs — ascending, descending, or both. Also, many buildings will be deemed structurally unsafe for rescuers to enter unless they are certain there are people inside needing assistance. It is for forays into these buildings that robots are ideally suited, as robots can feed back video footage of the area but having a building collapse on a robot is no cost compared to having a building collapse on a rescuer.

So, the ideal rescue reconnaissance robot would have a long battery life, be light enough for one man to carry easily, small enough to fit at least through a 24" equilateral triangular opening (or smaller void), capable of navigating any obstacle in its path, cost-effective enough to be considered disposable in case of accidental destruction, able to feed back live video and other sensory information to the rescuers, and controllable easily and remotely to deep within a collapsed building from safely outside the danger zone.

Focusing the Project

Most existing robots used for urban search and rescue (USAR) have considered all of these traits but the low cost one. To such an end, this project was specified, initially, as the development of a mid-sized, 6-wheeled, stair-climbing, low-cost robot for USAR. The "6-wheeled" specification was treated loosely, however — a guideline

[1] elevators

for conceptualisation.

The LCR project as a whole is looking to develop a complete, low-cost solution, including robot and user interface (UI), but as this project was principally concerned with testing robot configurations, the UI was allowed to be a laptop computer for the time being, with support for joystick inputs. Thus, the complete project would deliver: a mid-sized, wheeled, stair-climbing, camera-and-sensors-equipped, wirelessly controlled robot; a graphical user interface (GUI) displaying the video feed and sensor data and accepting human interface device (HID) inputs that would control the robot; and a detailed report on the design, building, and testing that would help further the LCR project by providing lessons learned in, and recommendations for, the development of low-cost robots for USAR.

This project was a first-iteration project, and as such was always to be developed upon in other projects after its completion, so functionality was considered of much higher value than marketability. This project had to be executed on a tight schedule, beginning in March, 2013, and ending in October of the same year, with a six-week break in the middle, so indefinite refining of the robot was not feasible. However, where time was in hand, some issues were addressed before moving on to subsequent objectives. Cost-minimising was of primary importance in this project, and all design and building decisions were made based on this.

Objectives of this Report

This report will detail the design, building, and testing of the 6-wheeled, low-cost, rescue reconnaissance robot and provide recommendations to aid further LCR projects.

To give the fullest possible picture of the development, the report will start with a review of the information gathered in research for the development of the robot. It will then show the concept-generation phase of development, along with the design choices with their justifications, and then the main design phase will be detailed under sub-sections for mechanical, electrical, and software systems. Thereafter, an outline of the testing procedures will be given, preceding the details of the building and testing of the various systems and sub-systems. The report will conclude with a cost assessment, then a summary of lessons that may be learned from this project and recommendations for further LCR projects.

Chapter 2

Background and Theory

To truly understand the project and what was required, thorough research needed to be done to ensure the wheel was not re-invented several times over and that the robot that was developed would actually be useful and applicable to the intended field — urban search and rescue. The following is a summation and paraphrasing of relevant information that surfaced in that research.

2.1 A Short History of Rescue Robotics

As well as knowing what other reconnaissance robots are on the market and in development today, knowing why they have come to be the way they are is also important to the design of a new robot, and as such, it is good to look briefly at the history of these vehicle. One would do well to take note of any recurring themes therein.

2.1.1 Origins

The Very Beginning

The basis of all tele-operated ground vehicles — vehicles controlled by operators some distance away from them — is the need to get some device some place where the operator cannot go or does not wish to go. The concept of tele-operation started coming to the fore with the burgeoning of explorations around electricity and magnetism at the turn of the 19th century. Nikolai Tesla, among others, showed a particular interest in delivering power and information wirelessly over distances, and patented the designs for the wirelessly operated boat in Figure 2.1 for militaristic purposes (again, getting a device where an operator did not wish to go himself) in 1898 [3].

The 20th Century

In World War II, various countries made tele-operated vehicles, but they all had shortcomings, not least of which was the common lack of camera feeds. Germany's Goliath "tracked mine" also had to operate on a tether, which was easy to disrupt and too short for operator safety. Later in that century, with improved electronic technology, wireless control burgeoned, but only with the 9/11 attacks in New York City in 2001 did robotics development really start to gain momentum.

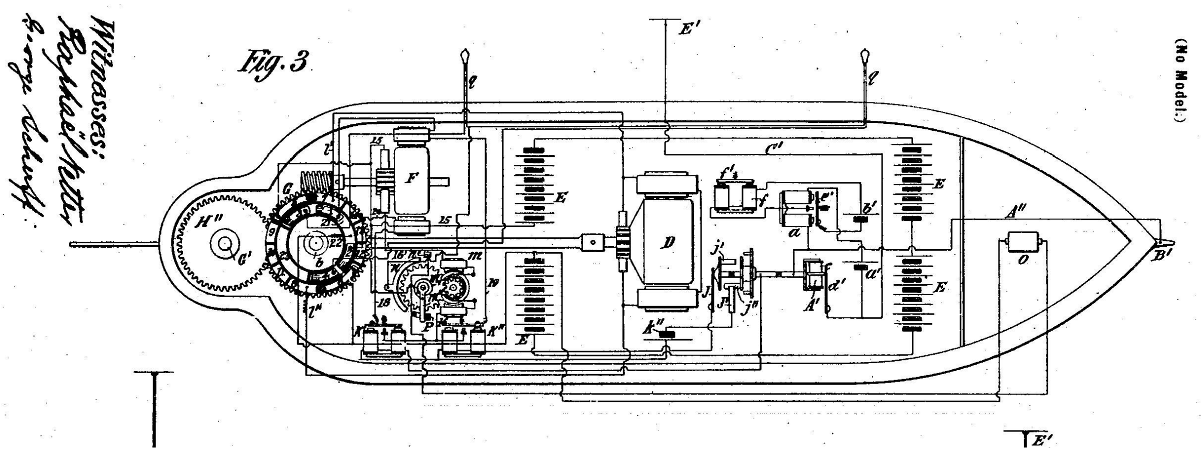


Figure 2.1: An electro-mechanical schematic from Tesla's 1898 wireless boat patent

Figure 2.2: Sept. 17, 2001 - FEMA Urban Search and Rescue teams work to clear rubble and search for survivors at the World Trade Center. [4]

9/11, 2001 With the attacks on the World Trade Centers, rescuers called for robots to go where rescuers could not — down vents and shafts and under piles of rubble like those in Figure 2.2 Even then, though, the robots were at such a stage of development that it was the robot suppliers — not the rescuers themselves — who had to operate the robots.

Subsequent wars and other disasters saw a boom in robotics development, and this made its way into urban search and rescue. Since this project is aimed at that field, it would be best to gain a more in-depth understanding it before continuing.

2.2 Urban Search and Rescue Operations

According to the Federal Emergency Management Agency (FEMA) in the United States,

> "Urban search-and-rescue (US&R) involves the location, rescue (extrication), and initial medical stabilization of victims trapped in confined spaces." [5]

They go on to say that victims are most often trapped due to structural collapse, but transport and mine accidents and collapsed trenches are also causes. These events can be

Figure 2.3: A rescuer exits a collapsed structure in Haiti after the earthquake in 2012. [6]

caused by anything from a natural disaster to a terrorist attack or hazardous materials release.

2.2.1 Operating Conditions

Outdoors USAR operations can take place over vast areas or in isolated buildings. In some cases, vehicular access can be limited, as in Figure 1.1, and first-responders must reach the site on foot, climbing over much debris to do so. Once at a site, buildings and vehicles are assessed externally by structural and safety experts. If a site is considered safe to be entered, it will be searched and victims extricated if possible. A system of markings is used to identify to rescue teams whether a site is being searched or has been searched, and what the results of the search were when finished. Thus multiple teams can work systematically across any affected area and not waste time and resources duplicating search efforts.

Indoors Victims usually seek refuge in structurally safe voids created in collapses. These voids can be in or under vehicles, in basements, under tables or baths, in cupboards, or under partially collapsed walls and roofing that may create impromptu leantos. These voids may be anywhere on a site and are often within buildings. If a building is safe to be entered, rescuers must enter and search for those voids that may hold victims. Often access for rescuers is restricted to crawling through tight holes as seen in Figure 2.3. If a building is deemed structurally unsafe to be entered, teams can call out to illicit responses from conscious victims inside or use visual scopes and listening devices to try to detect victims in the building. If a living victim is detected in a dangerous building, steps are taken to deal with hazmat dangers and shore up the structure to make it safe for rescuers to enter, then to find and extricate the victim.

2.2.2 Current Technology

Eyes, Ears, and Dogs For most searching, first-response teams rely on the rescuers' own eyes and ears and the skills of search dogs. The dogs are trained to sniff out victims and bark to alert their handlers when they find anyone. However, these dogs will not distinguish between alive and dead victims, and they only find victims; other necessary reconnaissance tasks such as identifying structural weaknesses and hazardous materials (hazmat) threats must be done by human rescuers. [7]

Tools and Electronics The current suite of tools available to first responders is limited. Regarding search equipment, specifically, all officially required options are restricted to hand-held scopes and listening devices of up to 70“ in length [8], so human rescuers must necessarily enter the danger zone to do a thorough search. To access these areas, if no stable entry presents itself, rescuers have two main options. They may drill a small hole into which they can insert a searchcam or fibre-optic probe, or they may cut an opening for a rescuer or dog to crawl through. A probe's reach and field of view are limited, so probing is often done using a series of holes at intervals along a structure. If a rescuer enters the danger zone, they may call out for responses from trapped victims; use headlamps to search visually for victims, structural weaknesses, and hazmat dangers; or use night vision goggles or thermal cameras to identify bodies in dark or murky voids.

2.2.3 The Need

The Early 2000s In USAR operations, the safety of the team members themselves is paramount, but most of the places they are required to go are very dangerous. Back in 2003, a workshop with North American USAR practitioners, following the World Trade Center rescue efforts, produced a list of high-priority needs in USAR technology. A few of the pertinent ones are quoted here:

- Improved real-time data access (data pertaining to site conditions, personnel accountability, medical information, etc.)

- The ability to accurately and non-invasively locate survivors following structural collapse — the ability to "see" through walls, smoke, debris, and obstacles
- The ability to communicate (transmit signals) through/around obstacles
- Improved monitoring systems (i.e., atmospheric, biomedical, personnel accountability, etc.) - real-time, portable, multi-function devices that expand on existing detection capabilities
- Integration/consolidation of functions found in multiple pieces of equipment into a single piece of equipment
- Reliable non-human, non-canine search and rescue systems - robust systems that combine enhanced canine/human search and rescue capabilities without existing weaknesses (i.e. robots) [9]

Development Response

With these needs expressed, robotics development over the next few years followed a generalised approach, with robots developed to meet as many needs as possible. Unfortunately, this brute-force approach to covering all bases has resulted in very expensive robots which were very difficult to develop, and which sometimes also required multiple personnel to manhandle them onto site. Other need-environments like law enforcement and military reconnaissance have turned up smaller, throwable robots like the iRobot® FirstLook and Recon Robotics Throwbot, but these are still top-of-the-range, costly items and are optimised for buildings that are still intact. When a rescue organisation spends several thousand dollars on a piece of equipment like a robot, it has to be treated as an investment, and must therefore be operated with great care — something that is not really feasible in many time-constrained USAR operations.

New Needs

With rescuers now having some experience with these robotics developments, a new need has emerged — one for robots that deal with most of the needs expressed in 2003, but are also cost-effective and as easy to deploy as any other one-man tool.

2.3 Current Global Lightweight and Low-Cost Robot Development

To avoid re-inventing the wheel, as ever, and to gain inspiration and ideas, some consideration should be made of contemporary development of lightweight and low-cost reconnaissance robots. One of the primary ways in which development is channelled and promoted in these fields is through competitions hosted by military or civil organisations and agencies. In the following sections, some of the robots and concepts that these competitions — and other development environments — have produced will be considered for what they may contribute to this project.

2.3.1 Military Competitions

In most robot-developing countries, military development is seldom as cash-strapped as civil development, but the need for *lightweight* reconnaissance robots for use in dismounted operations is very prevalent. Recently, the US military has sent four lightweight robots — Throwbot, FirstLook, Dragon Runner 10, and Armadillo — to Afghanistan for field testing, and on their home front has held a "Robot Rodeo" to operationally test a much larger selection [10]. From the results of these tests, a few high-performing robots were considered relevant to this project:

Figure 2.4: RHex [11]

RHex Boston Dynamics' RHex is a cross between a standard wheeled robot and a hexapod walker. As seen in Figure 2.4, it uses six semicircular legs hinged at one end instead of wheels pivoted around their centres. By driving its legs in tripod groups, it remains stable, with a motion similar to a six-wheeled robot, but has the advantage of being able to climb almost any obstacle — like a legged robot. RHex is a very expensive, sophisticated vehicle, with full waterproofing and leading-edge sensors among other features, but the mobility concept alone is worthy of note. [11]

Throwbot Recon Robotics' Throwbot is a prime example of how to get maximum functionality out of a very small package. Throwbot is a two-wheeled, tail-dragging robot, simply equipped with a small camera, but, as in Figure 2.5, it is all fitted into a hand-sized package that can be thrown 15m horizontally, through windows, fall 4.5m, and remain fully operational when it lands. To be as tough as it is, though, Throwbot uses very high-quality materials, like titanium, giving it a starting price of $4875. [13]

FirstLook iRobot®'s FirstLook is closer to the size this project is considering. At 229mm wide × 255mm long × 102mm high, with a flexible antenna on top and weighing only 2.45kg, the robot and its controls can be carried in a standard backpack. FirstLook, like the Throwbot, can be thrown significant distances and through windows, remaining fully operational upon landing. As can be seen in Figure 2.6, it has a tracked drive, with a pair of flippers on the one end extending back past its centre of gravity so that the entire robot can be flipped end over end. The flippers allow the robot to self-right, pitch its cameras upwards, and climb some obstacles — all things this project should attain to. Firstlook's impressive specifications and audio-visual equipment make it very expensive, though. [15]

Figure 2.5: The Throwbot LE kit [12].

Figure 2.6: FirstLook [14]

Figure 2.7: The ARA Pointman [16]

Figure 2.8: The ARA Pointman climbing stairs [17]

Pointman The Applied Research Associates (ARA) Pointman Tactical Robot is another moderately-sized (480mm wide × 330mm long × 133mm high, with a 457mm tail), lightweight (6.8kg) reconnaissance robot, like the FirstLook, but with mobility nearing that of RHex. It is a four-wheeled robot, shown in Figure 2.7, but with an actively-powered tail between the axle bases that makes the Pointman self-righting and can rotate those axle bases to help the robot traverse difficult terrain. This ”polymorphic locomotion system“ allows the Pointman to climb obstacles up to 280mm high, as in stair climbing in Figure 2.8, yet also drive into voids only 150mm high, something RHex cannot do. The Pointman has two cameras — one midway along the tail, providing good situational awareness for driving, and another at the base of the tail for use when the tail is lowered for obstacle negotiation. The upper camera can be swapped for thermal or visible illumination systems [17]. The Pointman is also meant to be “affordable” for law-enforcement agencies [7], but considering the general state of reconnaissance robotics, this is a very relative assessment, and cannot be considered affordable by the standards of this project.

2.3.2 Non-militaristic Competitions

Some civic organisations, such as the United States' National Institute of Standards and Technology (NIST) have also put resources into catalysing robotics development, especially for search and rescue. The NIST started the RoboCup Rescue league in 2005, with a set of standardised tests to assess robots' respective aptitudes in the USAR environment. All robots competing in the league have been tracked, wheeled, or a combination thereof, but it has only been the tracked robots that have fully completed the mobility challenges, so no positive inspiration for wheeled mobility systems can be gained from the robots in this competition. [18]

2.3.3 Low-Cost Robotics Development

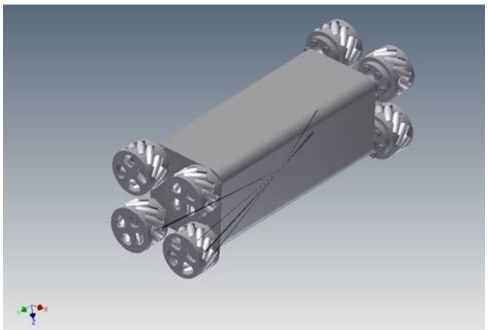


Figure 2.9: The SPYROB [19]

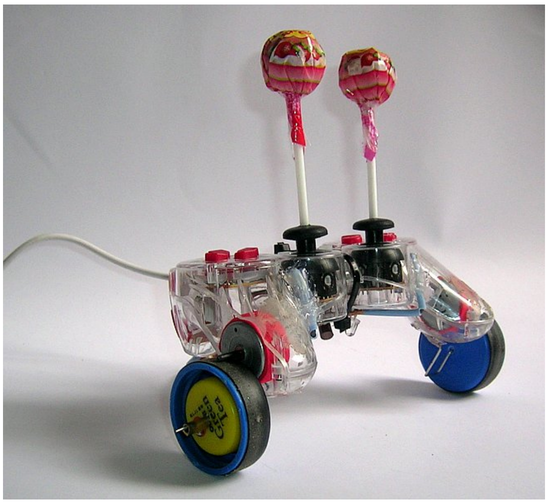
While there seems to be a push for low-cost robotics, what is deemed "low-cost" at the moment is laughable. The Throwbot LE is claimed (by its manufacturers) to be the "world's most affordable tactical robot" at $4875 for the kit [13].

SPYROB

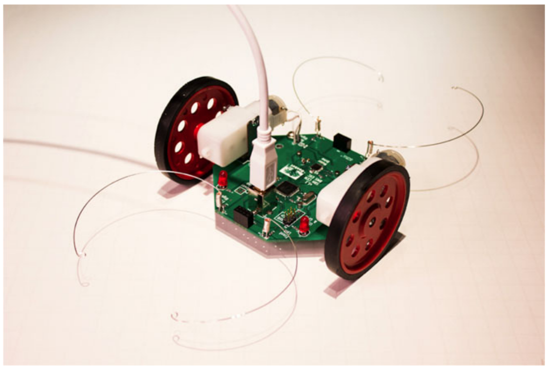
There are a few projects in progress trying to reconcile the robotics version of "low cost" with that of the public at large, like the SPYROB project by EUREKA which is pulling in multiple organisations and firms to develop a low-cost, throwable, "environmental reconnaissance" robot, as conceptualised in Figure 2.9, with a target price for the initial production model of €2500 [19, 20]. The SPYROB project, however, seems still to be waiting to get enough participants on board before it goes ahead.

AFRON $10 Robot Design Challenge

While there may thus not be any completed projects to be taken as full examples of truly low-cost reconnaissance robots, there are other, less closely-related projects from which to glean ideas and information. A good example is the $10 robot design challenge that the African Robotics Network (AFRON) ran in 2012. AFRON put out a challenge to help bring inexpensive robotics education to African classrooms, with a target product cost of $10. Interesting and inspiring entries included "Suckerbot" — a hacked PlayStation® controller with the vibration masses replaced by bottle top wheels and lollipops on each joystick to act as primitive accelerometers — and "Baobot" which used its circuit board as the chassis (probably not a good idea "as is" in USAR, but it should be kept in mind that circuit boards can *add to* structural integrity). [21]



(a) Suckerbot



(b) Baobot

Figure 2.10: Two submissions for the AFRON $10 challenge [21].

2.4 The Robot

With all the above information in hand, the requirements on this robot can be better refined and elaborated upon. This last section of the Background and Theory chapter will thus consider how to apply the above information to developing this robot.

2.4.1 Mobility Requirements and Drive Systems

From consideration of existing robots, as in the Current Global Lightweight and Low-Cost Robot Development section, it can be seen that the existing techniques for traversing obstacle-strewn terrain (without resorting to full walking) are limited to tracks with flippers as on the FirstLook, "polymorphic" systems like on the ARA Pointman, or unconventional wheels like RHex's legs.

The Environmental Challenge

The USAR terrain is the major consideration here. Mission scenarios can be dusty, wet, or even muddy, which adds another dimension to climbing obstacles, as was clearly illustrated when the ARA Pointman was tested in a mine for a National Geographic documentary [7]; the developers had to add galvanised rubber to the axle housings to stop them slipping on the dusty rubble — a scenario they had not had to deal with in the standard law enforcement scenarios the robot was designed for. Similarly, robots can beach on jutting obstacles if they have fixed geometry [18]. Whatever drive system is chosen, it should consider the need to adapt to shifting terrain containing a wide variety of obstacles — not just furniture and stairs.

Motors

As for actual drive, salvaging motors like in the AFRON challenge would be cheapest, but perhaps not reliable enough. Perhaps a better option would be purchasing low-cost hand tools and appropriating everything from them but their cases. As for *types* of motor, certainly standard brushed DC motors would be lower-cost than brushless ones for the same power metrics. Also, the fewer motors required, the lower the cost, unless removing motors requires the addition of an extremely complicated drive train.

2.4.2 Materials and Manufacturing

Materials and manufacturing techniques are intrinsically linked, so will be considered here together.

Cost-Reduction Cost reduction in this area can be done in three main ways: making use of inexpensive, easily available and machinable materials; reducing the processing and machining required; and minimising the amount of raw material required. In some cases there will be compromises, as inexpensive materials can be weak or overly dense where a more expenisve material could fulfill the same role using less material or weighing less, thereby proving to be the most cost effective solution in the long run. As well as considering man-portability, increased mass requires increased power from the motors — especially in climbing — which can increase total cost.

Unconventional Solutions Also, unlike a project where cost is of less concern, manufacturing the majority of parts from raw materials could be unwise. Unconventional solutions, such as using plumbing pipe for a chassis or repurposed electronics as in the "Suckerbot", may prove best in places. This would certainly reduce machining costs, and any parts made in bulk by a third party will be more ecconomically attained than if the same were manufactured once-off in-house.

Manufacturing Where manufacturing parts in-house cannot be avoided, the design must be such that the lowest-cost manufacturing can be used. For instance, RARL has access to a laser cutter, which can be used very quickly, affordably, accurately, and automatically to manufacture flat parts from certain polymers, woods, and composites. Where stronger materials like metal must be used, sheet metal should be considered because its manufacturing would be more economical than that of a similar part machined from a solid block.

Lastly, under manufacturing, the design must consider ease of both the assembly and disassembly of the robot; the robot must be easy to maintain, and if a part fails, it must be easy to replace. A low number and specialisation of required tools for assembly would also simplify assembly and reduce overall cost.

2.4.3 Video

A live video feed is perhaps the most fundamental sensor requirement for this project. Small, low-cost cameras like webcams can be had for as little as R150, and there are many ways to get the feed back to the operator. The positioning of cameras is also important — ideally, the view should encompass the entire robot, showing where its limits are, and how it is positioned relative to its environment. In practise, this is not entirely possible, but positioning a forward-facing camera up and back from the front edge of the robot is certainly helpful. Robots like the FirstLook have *four* cameras, giving them 360° situational awareness, but every camera is a further cost, and in USAR, unlike in the tactical situations FirstLook is designed for, a robot does not need to be aware of gunmen sneaking up on it. A rear-facing camera, though, may be helpful for reversing.

Figure 2.11: The LG D2342 passive 3D monitor with polarised glasses.

Stereoscopy With the recent boom in stereo 3D film-making and 3D consumer products, consumer monitors and head-mounted displays that can display stereo 3D, like the LG D2342 23” monitor in Figure 2.11, have become relatively common. With this in mind, it is worth considering stereo 3D camera options in case this may help operation of the robot.

Stereo video uses two cameras side-by-side, aimed parallel to each other. A clear, thorough explanation of how stereoscopy and binocular vision work is given gy [22] In today’s world of wireless technology, for live video streams from these cameras, either analogue or digital cameras can be used. Either way, the two images must either be transmitted in parallel, or interlaced somehow into one stream.

Parallel streaming requires two of every transmission component — two sets of cables, two transmitters, two receivers, and possibly two ports on the display device. This makes set-up relatively straightforward — a matter of setting the right channels on the transmitters and receivers. However, doubling every component doubles the price.

Single-channel streaming differs depending on whether digital or analogue video is used. For digital video, it is a case of compressing each stream enough, or resizing the feeds so each frame can be sent down the line one after the other. Some CCTV systems make use of this kind of system to display multiple images on one monitor.

For analogue video, there are different options. Some old 3D films used colour filters — for instance the classic red and cyan glasses in Figure 2.12 — to restrict the right and left eyes to certain colour spectra. This filtering can be done right at the cameras and the video signals simply summed, but the loss of colour for each eye is far from satisfactory.

Another option, and in fact an method proposed as a standard at the beginning of the recent stereo boom, is field-sequential 3D. This relies on certain properties of composite video. Composite video is the old standard analogue video format found

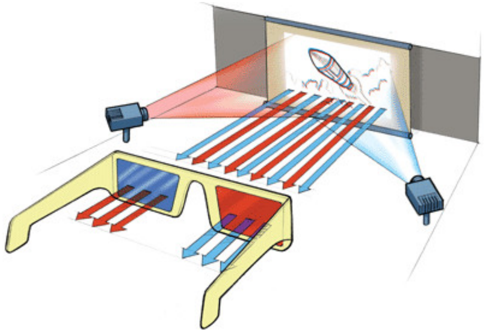


Figure 2.12: Anaglyph (red and cyan) display filtering.

in televisions and media systems around the world. There are various standards for composite video, the most common being PAL and NTSC, but for the purposes of field-sequential transmission, the differences between these standards are irrelevant.

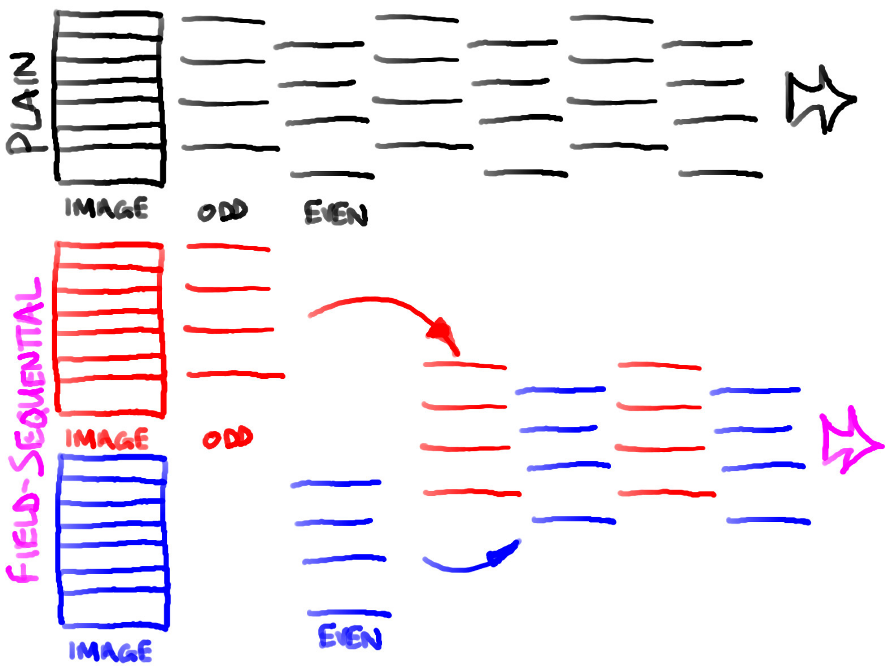


Figure 2.13: Field-sequential stereoscopic transmission.

Composite video is transmitted line by line, as in the top of Figure 2.13, but, to increase the rate at which the full screen area is updated, all the odd lines are sent first, followed by the even lines. Thus the signal "scans" down the screen twice for each frame. These two scans are called the odd and even phases [23]. Field-sequential 3D uses these phases to carry the two views, as in the lower section of Figure 2.13, with a halving of the vertical resolution of each feed, but a preserved aspect ratio

Field-sequential analogue transmission has a digital equivalent in "row-interlaced". The difference is only that the analogue signal is a time function where the digital information is stored in whole frame units.

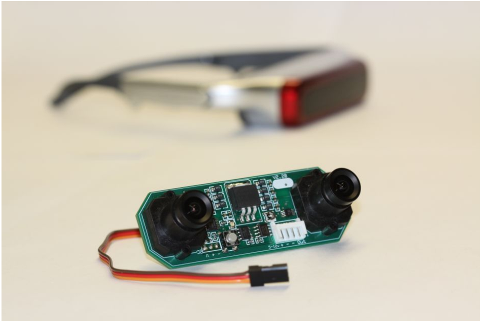


Figure 2.14: The New Generation Hobbies 3D FPV camera [24]

2.4.4 Sensors

Infrared and Thermal Imaging Beyond the standard visual-range camera, USAR operations have been known to be significantly aided by the use of thermal or infra-red (IR) imaging to identify victims amidst dirt and rubble or through thick smoke. Thermal imaging is best for this, but a thermal camera would be a large cost, would likely be larger than other cameras, and would require additional protection against scrapings and shock. An optional, modular thermal imaging package, as offered on the Pointman may make the best commercial sense in the end, offering the customer a lower starting price with the thermal imaging easily added, or even hired, at a later date.

All cameras can be aided by illumination, too, with headlights or infra-red lights.

Other sensors Other sensors can be helpful, such as a compass and inclinometer for navigation and situational awareness. In USAR operations, atmospheric monitoring may be considered helpful, too, though there are degrees of this, and it may be most cost effective to acquire an all-in-one sensor that simply registers the presence of gases rather than several sensors dedicated to specific gases. Less obvious sensors, such as rotary encoders (or similar devices) may be necessary for internal position control, even if their data is never overtly displayed for the operator. In fact, position control on the wheels and other moving parts is a major consideration for ensuring the robot is not too buffeted by its environment — a design like RHex would be impossible to implement without accurate position control over the drive motors.

2.4.5 Power

For the scale of this project, it is assumed only electric motors running off pre-charged batteries are applicable (i.e. no IC engines). Even so, there are a number of options for batteries. For repeated, economical use, rechargeable batteries are preferable, offering longer total life than their non-rechargeable counterparts. There are five major rechargeable battery chemistries: Nickel Cadmium (NiCd), Nickel Matal Hydride (NiMH), Lead Acid, Lithium Ion (Li-ion), and Lithium Ion Polymer (Li-polymer). They can be summarised as shown in Table 2.1.

While some of these (particularly the lithium batteries) tend to come in varying manufacturer-specified packages, there are also a number of international standard

Table 2.1: Battery chemistry strengths and weaknesses [25]

	Strengths	Weaknesses
NiCd	Lowest cost-per-cycle Very rugged High charge and discharge rates	Moderate capacity Needs monthly full-discharges to avoid "memory" build-up Needs careful disposal due to highly toxic constituents
NiMH	30-40% higher capacity than NiCd Less prone to "memory" Less harmful to the environment	Cannot supply the same currents as NiCd Needs more complex charging High levels of self-discharge (needs to be charged just prior to every use) Still requires full-discharges for maintenance
Lead Acid	Rugged, simple, and inexpensive Require no maintenance Can provide high current	Very heavy Provide limited charge cycles Environmentally unfriendly if not disposed of carefully
Li-ion	Highest energy density of this selection Can discharge at moderate currents Require no charge maintenance	Expensive Fragile Subject to ageing Require dedicated circuitry to control charge and discharge levels
Li-Poly	Similar to Li-ion, but can be made very small and lightweight	Similar to Li-ion, but lower energy densities Propensity to explode in use or recharge if physically damaged

battery shapes in each of which can be found most or all of the above-mentioned chemistries, and which will be available world-wide. These standard battery shapes are each associated with a certain approximate voltage, and thus, given a socket for one shape of battery, the end-user, rather than the designer, can determine which chemistry they wish to employ. A standard battery size also allows the implementation of non-rechargeable batteries if, for example, there has not been an opportunity to charge the regular batteries. The most ubiquitous standard battery form is AA (penlight), with capacities around 1000 - 3000mAh. Alternative common forms are C ("flashlight batteries") — 4000 - 8000mAh — and 9-volt — 120 - 1200mAh. [26]

2.4.6 Control and Communications

For this project it is assumed the majority of control will come directly from the operator; extensive autonomy would require more complex sensor arrays and more powerful processing, and is also a controversial topic in today's robotics. There are two main ways to get user commands to the robot — tethers and wireless communication.

Many early unmanned ground vehicles (even many not so early ones) were operated via tethers. Tethers allow power to be stored externally to the robot and make multiple communications lines relatively easy to implement. However, in USAR, the number of jagged edges and potential snag points in the operating environment makes tethers undesirable. Wireless control has limitations indoors, with structures — especially steel reinforcements — of buildings acting as electromagnetic barriers, but the improved mobility greatly offset this.

As for how the user issues the commands, contemporary small robots, like the Pointman, Throwbot, and FirstLook all have dedicated controllers, with some, like the Pointman, having additional options for a laptop interface if required. Since the robot is likely to be running code on a serial-equipped microprocessor, it makes sense to start with an RS-232 serial interface and develop any more dedicated control later; RS-232 wireless modules are available that allow wireless serial communication from any serial-equipped computer. A laptop can also fairly simply be made to show the video feed on its monitor.

With the proliferation of video games and standardisation of their controllers, companies like iRobot® have noticed drastically-reduced operator training times for their robots with the use of standard game controllers. This makes the initial use of a laptop all the more appealing, because game controllers can be integrated easily into the control interface.

Chapter 3

Concept Development

From the data in Chapter 2, a list of desired specifications for the robot were laid out. With these in place, concepts could be generated and evaluated. The basic specifications list is as follows:

Table 3.1: Specifications List

Cost	<R4000
Stairs	$\leq$ 80% inclination, $\leq$ 200mm height
Void	$\geq$ 200mm
Mass	Easily man-portable ($\approx$ 8kg)
Speed	$\geq$ 1m/s
Endurance	$\geq$ 1hour (mobile)
Range	$\geq$ 30m range inside steel-reinforced concrete structures (wireless range is significantly impaired indoors, so that is the limiting environment)
Sensors	Camera
Control	Wireless — laptop-based graphical user interface (GUI) — inputs via common HID joystick

With specifications in place, as outlined in Table 3.1, design concepts were explored and evaluated so as to meet those specifications. A total of four distinct concepts were evaluated and the best one chosen to be carried forward into the later phases of the project. While not every concept strictly adhered to the project description — some stretched the term "wheeled" to its limit and one had eight instead of the suggested six wheels — none was discarded on that basis, since the initial project specifications were treated as benchmarks and guidelines for creativity, rather than strict laws, within reason.

3.1 Concept No. 1

The first concept was for a flat, partially-tracked, six-"wheeled" platform, as seen in Figure 3.1 — a short pair of tracks was placed between pairs of wheels that could be raised and lowered (in unison) to allow the robot to reach up to and climb stairs and other obstacles.

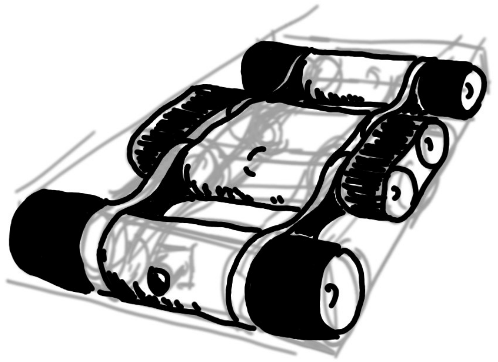


Figure 3.1: Concept No. 1: A partially tracked, actively-controlled-geometry concept

Conception For a robot to climb an obstacle taller than itself, it needs either to drive up the side of the obstacle, or to be able to reach up to the top of it. In a dusty environment, and with simple wheels, the first option is infeasible, but the second can be achieved with some form of shape-shifting on the robot's part. This first concept was a development of the most simple shape-shifting concept available for a six-wheeled platform — three evenly-spaced pairs of wheels with a body that hinges around the middle pair's axle. Unfortunately for that concept, it is almost always unstable in climbing; if the CoG is rearward it struggles to lift that onto obstacles, if forward then the rear wheels always lift instead of the front, and if centralised, then there is no reliable way to determine which wheels will actually rise. This could be solved by replacing the middle pair of wheels with a pair of tracks; the robot could then raise its front and rear wheels simultaneously while remaining stable on the base provided by the tracks — hence Concept No. 1.

3.1.1 Systems and Layout

The systems of this concept are shown in Figure 3.2. Without threading complex belt drives through the booms, this concept ends up using at least seven motors (blue) — one for each wheel, and one for the shape-shifting. Despite this, there is room for a large battery (green), circuitry between the motors, and an array or sensors (at least cameras) in each end pod (red).

3.1.2 Feasibility

Climbing 2D simulation of the concept indicated it could very easily navigate stairs, as shown in Figure 3.3. However, the stairs it could navigate were fairly shallow — their depth limited by half the length of the robot, and their height by the length of the booms.

Geometry Climbing aside, actively-controlled geometry is highly advantageous. This concept could become very low-profile when needed and could even be designed to be rela-

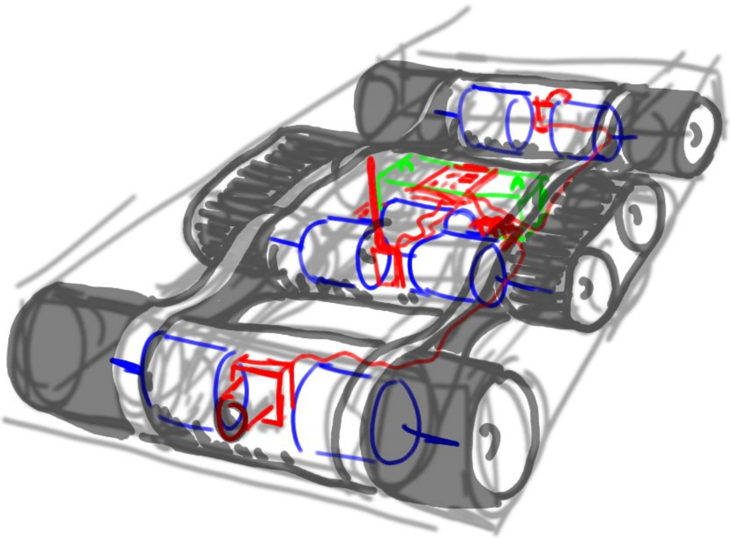


Figure 3.2: The systems layout for Concept No. 1

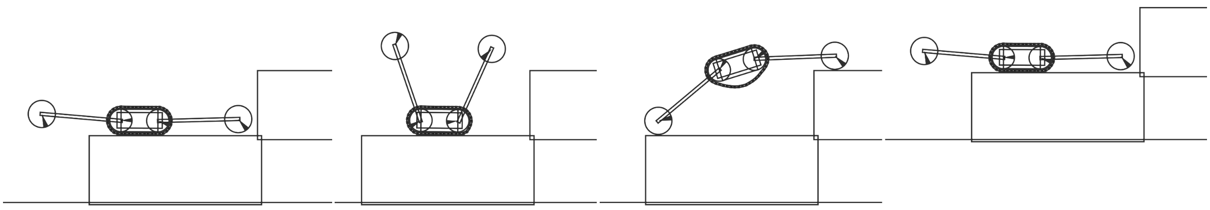


Figure 3.3: Concept No. 1 climbing stairs in 2D simulation

tively narrow, with the associated turning issues resolved by picking up the booms to reduce its wheelbase drastically when needed. This concept is also symmetric in three planes, so could be driven upside down or backwards just as easily as forwards and upright.

Systems As for the required components and systems, the major concern, besides the required number of motors, was that tracked locomotion is always tricky to execute when designing from scratch. However, Zodiac have recently released a track-driven automatic pool cleaner, and spare parts for those tracks are mass-produced, easily available [27], and have been proven to work; they would provide an easy way to avoid track design issues and, through their mass-production, reduce cost.

3.2 Concept No. 2

The second concept is for a low-cost version of the RHex robot developed by Boston Dynamics, as seen in Figure 3.5. The RHex concept has been openly developed by multiple universities [29] and, as such, is not proprietary to Boston Dynamics, but their

Figure 3.4: Zodiac's MX^{TM}8 tracked pool cleaner [28]

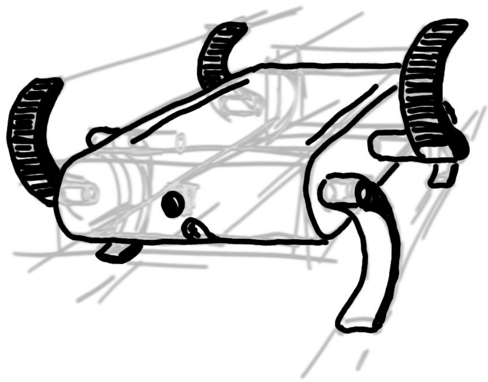


Figure 3.5: Concept No. 2: A low-cost implementation of RHex

product *is* the most fully-developed implementation of the concept on the market and as such shows what the concept is capable of in operation.

Conception In researching current robotic reconnaissance systems, each encountered robot was considered to determine whether that concept could fulfil this project's mobility specifications "as is". Boston Dynamics' RHex seemed to do so; the RHex concept provides many of the mobility-related advantages of legged systems while retaining most of the mechanical simplicity of static-geometry wheeled systems. In fact, the Rhex has been shown to be very effective on loose, unstable terrain — something that cannot be said of most legged robots, and something that would be very useful for this project. Rhex can also climb obstacles up to around $1\frac{1}{2}$ times its body height.

3.2.1 Systems and Layout

The systems of this concept are shown in Figure 3.6. At its most mechanically simple, the concept could be implemented with six motors — one for each leg — but it can be shown, by symmetry, that the forward and rear legs on each side always move in unison, and so the concept could, if need be, be implemented with only four motors, with a belt

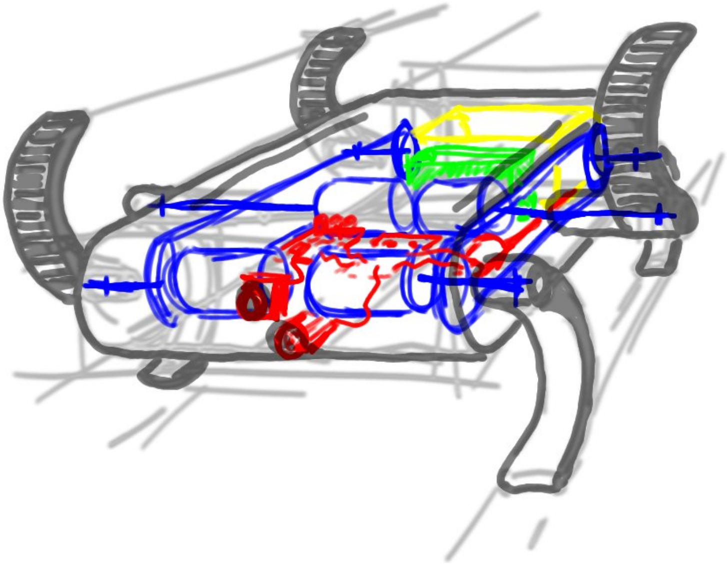


Figure 3.6: The systems layout for Concept No. 2

drive or similar system connecting the front and rear legs on each side (blue). Either way, there should be plenty of space for a large battery (green), all the electronics and sensors (red), and even extra space for some form of payload (yellow). Because it does not bend like Concept No. 1, wiring and mechanical connections in this concept would be much easier to install.

3.2.2 Feasibility

Climbing Any RHex-based platform should be able to climb onto surfaces around $1\frac{1}{2}$ times its axle height, especially if the CoG is toward the front of the platform. Rhex has shown this to be true [29], and the addition of nodules to the legs helps grip considerably. Also, because of the almost-walking nature of this concept, many obstacles shorter than the ground clearance would simply be stepped over, rather than climbed, and the body and camera(s) would thus remain very stable.

Geometry Unfortunately, because the legs have only one degree of freedom — rotation in a vertical plane — they must necessarily pass through their highest position with every stride. This means it would have height limits as shown in Figure 3.7. USAR operations would require the robot to enter low voids, and if this had to be a limiting factor on the robot's size, it would cripple its obstacle-climbing capabilities.

Systems As mentioned previously, this concept's body would have plenty of space for all necessary components, and its mechanical simplicity means the only moving parts are the legs, and perhaps the belts if a four-motor configuration is preferred. The challenge in choosing and designing this concept's systems would come in the choice of motors; because of the length of the legs, the motors would have to handle very

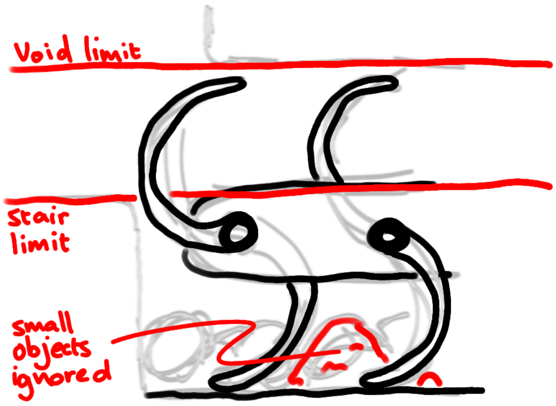


Figure 3.7: Height Limitations and Capabilities of Concept No. 2

high torques, and for effective walking, the motors would need very precise positional control — motors with these two characteristics tend only to be associated with low cost in that that they are *not*.

3.3 Concept No. 3

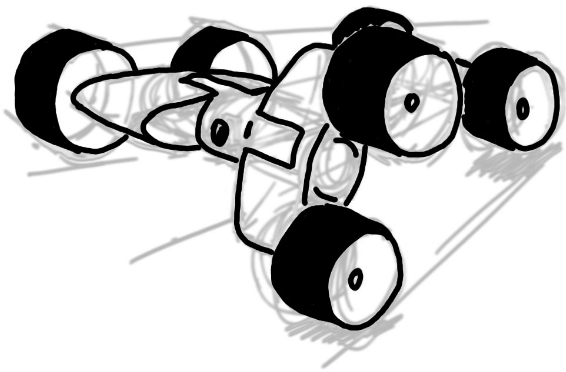


Figure 3.8: Concept No. 3: An adaptation of the "tri-star" drive train for a low-profile, passively-controlled-geometry concept

The third concept is for an automatically-shape-shifting, six-wheeled platform, as seen in Figure 3.8 — the front two wheels on each side are mounted on Load-Intuitive Modules (LIMs) that pivot around their centres and are geared to flip end-over-end when the wheels encounter a certain level or resistance, thus automatically trying to climb any obstacle the robot encounters.

Conception

This concept is based on a combination of the Pointman, referred to in Section 2.3, and the tri-star major-wheel/minor-wheel drive concept. A major-wheel/minor-wheel system has an array of "minor wheels" around a central hub such that the array itself can be rotated as a "major wheel". The tri-star system uses three minor

(a) The vehicle (b) The major wheel

Figure 3.9: The Lockheed Terrastar and a view of the gearing of its “wheel” [30, 31]

wheels geared to the central hub in such a way that on level ground the minor wheels drive the vehicle, but when the minor wheels encounter a certain level of resistance (an obstacle, for instance), the torque feeds back through the system and the major wheel is rotated, automatically climbing over the obstacle. Tri-star drives have been successfully implemented on vehicles like the Lockheed Terrastar amphibian, in Figure 3.9, and powered and un-powered versions of it are used on a variety of devices for assisting in moving loads up and down stairs. The problem with tri-star, for the purposes of this project, is that, like RHex, it can only climb obstacles that are a limited fraction of its height.

The Pointman can climb obstacles taller than its drive system, however, and was thus the inspiration for this concept — a combination of major-wheel/minor-wheel drive in a Pointman layout. Using only two minor wheels (“bi-star”), the gearing concept still works such that the major wheel rotates when the minor wheels hit obstructions, but the major wheel is only the height of one minor wheel when flat, yet can reach up to the tops of obstacles around $1\frac{1}{2}$ times higher than the height of a minor wheel — ideal for this project. However, if the major wheel beaches on an obstacle, the torque will revert to the minor wheels, and they will spin freely in the air. To avoid this, another set of wheels were added where the Pointman has its boom. These wheels push the robot forward, so even when the major wheels beach, they are helped by the rear wheels to keep rotating.

Constantly referring to “major wheels” and “minor wheels” is tedious and confusing, even more so because the major “wheels” are flat rather than round in any way. So, for the purposes of this project, the whole bi-star assembly will be referred to as a load-intuitive module, or “LIM”, referring to its automatic adaption of transmission under varying loads, and evoking associations with legged locomotion. Thus, this is a two “LIMed” concept.

For a more detailed explanation of the kinetics and kinematics of the bi-star arrangement, see Appendix A: Bi-Star Mathematics.

3.3.1 Systems and Layout

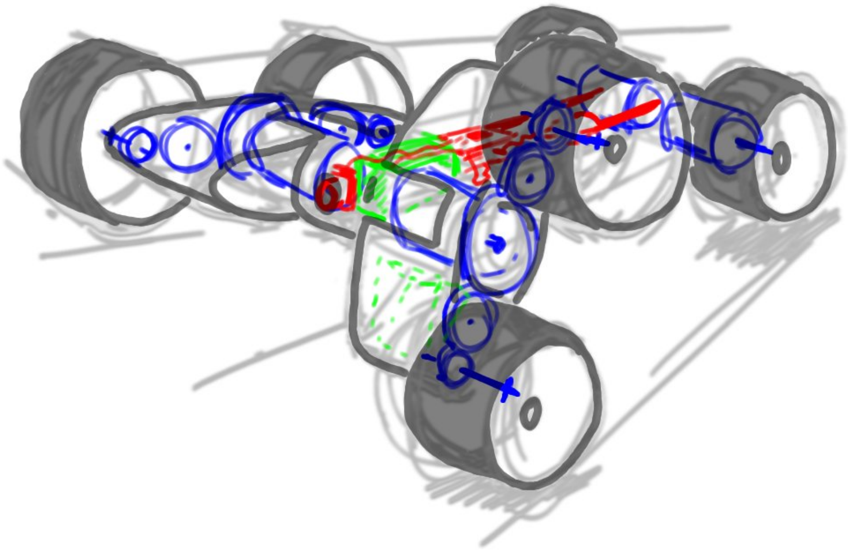


Figure 3.10: The systems layout for Concept No. 3

The systems of this concept are shown in Figure 3.10. The concept requires four motors (blue) for its six wheels — two for the rear wheels and two for the two LIMs on the front. To avoid beaching, and give the most space for the LIMs, the body is narrow and cylindrical, leaving little room for circuitry (red) and any decent sized battery (solid green). However, if a slip-ring connection system is used, batteries could be stored in the LIMs', freeing up space in the body and allowing for more, and larger, batteries (dashed green).

3.3.2 Feasibility

Climbing The Pointman, using directly-actuated axle hubs (very similar to LIMs) can climb stairs easily. However, the extra rear wheels required by a LIM-based derivative of the Pointman shift weight rearward, and make climbing stairs a little trickier for this concept. In fact, initial models and basic 2D simulation of this concept showed it would stop going up stairs once the rear wheels reached the first stair. In practice, one stair was fine but a flight was impossible. To avoid this Achilles' heel, a clutch mechanism would have to be introduced, like on the Lockheed Terrastar, to force LIM rotation rather than wheel rotation. This would be helpful under many more circumstances than just stairs, *but* would be an extra pair of actuators, adding extra cost and complication and possibly impairing control ergonomics if the operator had to toggle the clutch for every stair.

Geometry This concept is flat until it hits an obstacle, at which point it automatically "shape-shifts", as it were, to climb over it. Thus, if the wheels are the height-limiting components, it should be able to mount obstacles at least $1\frac{1}{2}$ times taller than its flat height (taller if the spacing between LIM wheels is increased) — easily satisfying this project's requirements, even if its obstacle height limit is a small fraction less in practice. It would have to be relatively broad if skid steering were employed, however — more so than Concept No. 1 which can reduce its wheelbase for tight turns.

Systems The narrow body of this concept leaves little room to play with. If the battery were incorporated in the body, it would need to be small, reducing endurance. Placing the batteries in the LIMs would vastly increase the potential size of batteries to be included. This would mean including some form of sliding electrical contact between body and LIM, since the LIMs must be unlimited in their rotation, and that is inherently problematic — some form of dust protection would have to be included, considering the operating context, and the complexity of the system would increase significantly. However, this could be mitigated by installing independent battery packs in each LIM, wired in parallel so that if one failed, the system would continue to run, be it with reduced endurance.

Machanically, gearing is always a little tricky, and the torques on these gears are also quite high, requiring careful design and installation.

3.4 Concept No. 4

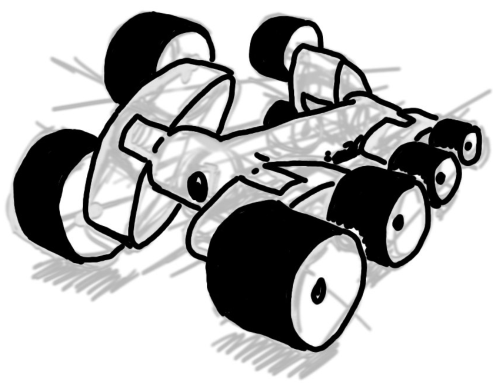


Figure 3.11: Concept No. 4: An adaption of Concept No. 3 to improve climbing mobility

The fourth concept is for a four-LIMed platform, as seen in Figure 3.11 — the rear pair of wheels of Concept No. 3 are replaced by another pair of LIMs to improve climbing mobility.

Conception Modelling and 2D simulation of Concept No. 3 showed that it would be incapable of climbing a full flight of stairs. Beyond the first stair, the rear wheels gave little, if no, support to the climbing — hence Concept No. 4. By replacing the rear wheels on Concept No. 3 with another pair of LIMs, the rear of the vehicle could be endowed with the same climbing capabilities as the front.

3.4.1 Systems and Layout

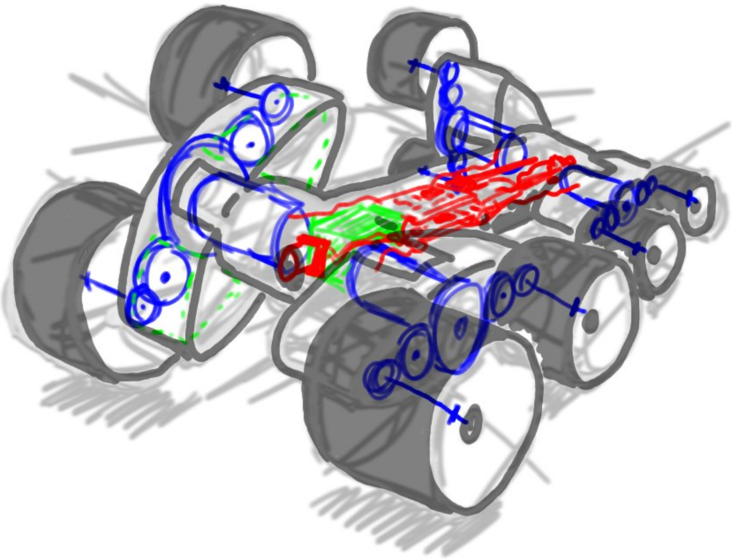


Figure 3.12: The systems layout for Concept No. 4

The systems of this concept are shown in Figure 3.12. As with Concept No. 3, this concept requires four motors (blue) — this time all four driving LIMs. Again, the body is narrow and cylindrical, leaving little room for circuitry (red) and any decent sized battery (solid green). Were slip-ring power connections implemented, this concept would provide even more space for batteries in its LIM voids (dashed green) than Concept No. 3.

3.4.2 Feasibility

Climbing This concept eradicates the climbing issues associated with Concept No. 3, allowing for full stair climbing with only the main drive motors actuated. The clutch mechanism mentioned in Concept No. 3 could still be added to allow even fuller control, but its incorporation would be more due to desire than need (at least as regards stair climbing).

Geometry This concept has all the same geometry advantages and disadvantages as Concept No. 3, with the added advantage that it is symmetrical in all three major planes, and therefore both invertible and reversible.

Systems The systems for this concept are the same as for Concept No. 3, with the added complexity of another two LIMs. However, these are repeated assemblies, so design complexity would be negligibly affected.

3.5 Concept Selection

The advantages and disadvantages of the concepts above can be summarised in Table 3.2.

Table 3.2: Concept Evaluation Summary

Partially Tracked	RHex	Bi-Star 6-Wheeled	Bi-Star 8-Wheeled
Pros	Pros	Pros	Pros
Profile control Internal space Easy to turn 3 planes of symmetry	Loose terrain Internal space Mechanically simple 3 planes of symmetry	Few motors Automatic climbing Low profile 2 planes of symmetry	Few motors Automatic climbing Low profile Unlikely to beach 3 planes of symmetry
Cons	Cons	Cons	Cons
Tracks Many motors Limited stair profile	Height Programming Motor precision Torque load	Beaches on stairs Idler gears or belts Camera stability Distinct front	Most mechanically complex Idler gears or belts Camera stability Not six-wheeled

The RHex concept, because of its height issue, would only have been feasible under this project's specifications if no better concept presented itself and the six-wheeled bi-star concept is superseded by the eight-wheeled version. Thus the decision was primarily made between the partially tracked and eight-wheeled bi-star concepts. Comparing those two, the tracked concept's issues with the large number of motors required and limited stair-climbing were considered to outweigh the mechanical complexities of the eight-wheeled concept.

Thus, the concept settled upon for this project was Concept No. 4 — the eight-wheeled, four-LIMed, bi-star concept.

Chapter 4

Design

The design of the robot was carried out in four parts — the mechanical design, the video system design, the design of the rest of the electronics, and the programming. As far as possible, the interfaces between these were kept very simple so that each would not affect the others too significantly. However, there was a definite hierarchy that determined the order of development — the mechanical design determined space allowed for electronics, and the electronics and mechanical design governed the specifics of the programming (if not the overarching algorithmic concepts). For this reason, the mechanical design was begun first, the video and other electronics were begun after the parts had been submitted for manufacture, and the programming was only begun in earnest once there was hardware to test it on.

The rest of this chapter is dedicated to each of these systems in roughly the order they were developed. In practice, each design section ran over into the next, and each was developed right up until this report was printed.

4.1 Mechanical Design

The mechanical design was broken up into sub-assemblies, since many parts of the robots were duplicates of others — for instance there were four LIM assemblies and two end assemblies. The whole concept of the robot revolved around the LIMs, so the robot was developed inward from there.

4.1.1 LIMS

Gear Ratio A feeling for the required gear ratio was first acquired through prototyping in LEGO and on a 2D computer simulation using Algodoo. A simplified kinetic model, shown in Figure 4.1 was then used to confirm and refine the numbers by balancing torque distribution between flipping and rolling.

In brief, the model goes as follows: flipping becomes easier as the LIM raises itself, since gravity has a shorter lever arm on which to act, so at the vertical position there is no need for flipping torque since gravity no longer restrains flipping. Therefore,

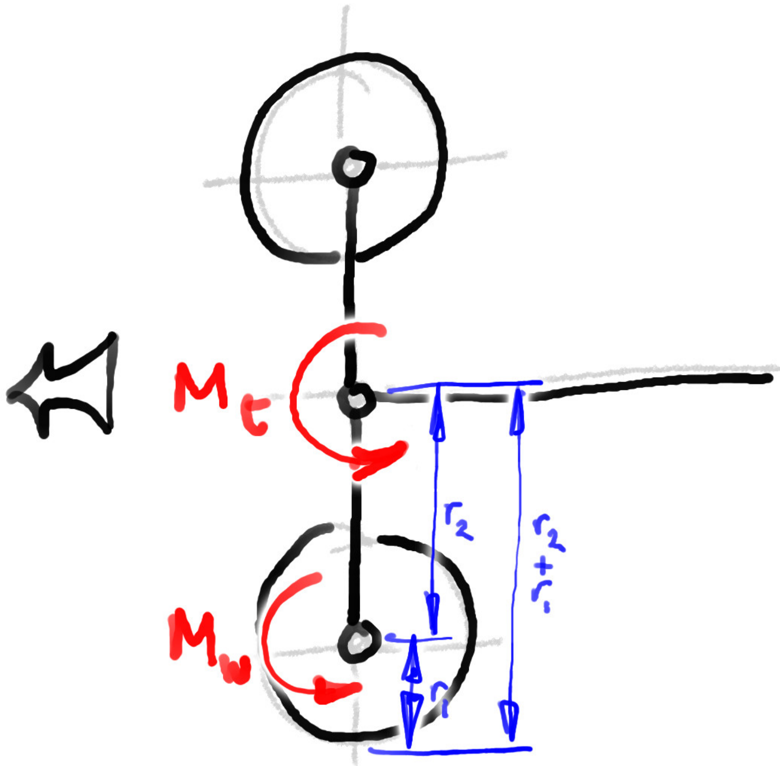


Figure 4.1: The LIM optimisation model

at the vertical position, the force exerted on the ground by the wheel torque and by the flipping torque should be the same, or very slightly in favour of the wheel torque.

$$F_t = F_w \tag{4.1}$$

$$\frac{M_t}{r_1 + r_2} = \frac{M_w}{r_1} \tag{4.2}$$

Since wheel flipping torque is simply motor torque and wheel torque is motor torque over the gear ratio:

$$N = \frac{M_t}{M_w} = \frac{r_1 + r_2}{r_1} \tag{4.3}$$

So, for this robot, the ideal N by this model was

$$N = 2.4 \tag{4.4}$$

Also, by increasing the size of the idler gears, their axles' distance out from the centre increased. This increased their leverage for flipping, reducing the overall inter-gear and axle forces.

Figure 4.2: A partially assembled LIM.

Structure For this prototype, the minimal material usages was wanted, so the full-sized "paddles" were replaced with minimal enclosures that were just thin enough to house the gears. However, this put large bending loads on the housing, so the paddle walls were thickened to 3mm and brace blocks were added as close to the pinions as possible to hold the shape of the gearbox, as seen in Figure 4.2.

Bearings Ball bearings were considered too expensive and bulky given the narrowness of the flippers, so what had been spacers were adapted to bushings that became journal and thrust bearings. This reduced the number of parts and simplified location. However, the material (Vesconite) and machining time required were not inexpensive.

Wheels It was decided to use pre-made RC buggy wheels, rather than make new ones, as these are available internationally in standard sizes and are mass-produced and therefore cost much less than it would cost to make a similar grade of wheel for only one robot. These wheels provide excellent grip and suspension.

Axles were designed to fit these wheels with a 14mm hexagon end that accommodated an M6 screw to hold the wheel on.

Motors Low cost cordless screwdrivers, shown disassembled in Figure **??**, were re-purposed to drive the LIMs with a few relatively simple adjustments: the shafts of these screwdrivers were replaced with custom-made shafts to fit the LIM drive gears; the casing was cut short and the original battery packs and mechanical switches dispensed with; to mount the motors, the pins that held the gearbox to the main housing were replaced with sections of M2.5 stainless steel threaded rod that extended out through the chassis of the robot and were secured with nuts on the outside of the robot; perspex bungs were used to secure the motor gearboxes in the

Figure 4.3: The screwdriver used for its motor and gearbox.

chassis, and the cut-off end of the motor housing fitted snugly into the PVC pipe junction used for the chassis.

4.1.2 Chassis

Materials For low cost but decent strength, standard PVC water piping was chosen to make up the chassis. The motors were accommodated in 50mm sections, while, for as much middle ground clearance as possible, 40mm section was used for the length of the body. Conveniently, 40mm plumbing T-junctions are designed to accommodate 40mm pipe on their insides and 50mm pipe on the outsides, so no extra adaptors were needed. The T-junctions would also accommodate any cameras on the robot.

Fastening and Reinforcing Initially, an additional aluminium spine was designed to fit down the length of the body pipe, but fitting electronics around it was troublesome, and a judgement was made that the PVC pipe alone would be sufficient for the load of the robot's own bodyweight.

The 50mm pipe was permanently fastened to the junction pieces with PVC weld, and the body piece, which had to be easily removable for maintenance, had holes added for locating pins that would be screwed into place to hold the two end assemblies onto the body pipe.

Because PVC is plastic and its standard joints are not flush on the outside, skid plates were added to the top and bottom of the body to facilitate and protect the body in climbing over obstacles — computer simulation indicated this would happen over most of the length of the body.

Electronics layout is described in more detail in Section 4.3.2

4.1.3 Mechanical System Design Summary

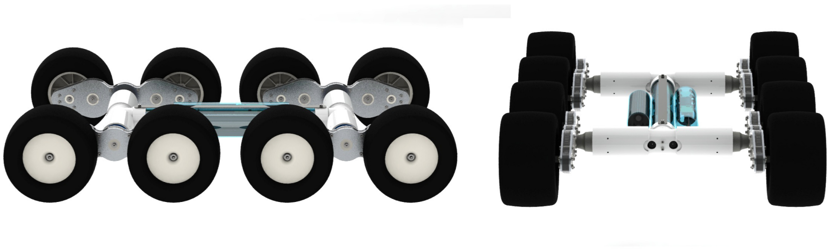


Figure 4.4: The mechanical structure.

The mechanical design, as seen in Figure 4.4, was done from the LIMs inward. Four LIMs with 2.4:1 gear ratios were attached to the protrusions of an "I"-shaped chassis constructed from standard PVC plumbing piping. The LIMs were driven by re-purposed chordless screwdrivers and the wheels were standard RC buggy wheels. The body was protected from scraping on the ground by skidplates.

4.2 Video System Design

4.2.1 Video Transmission Mode

The transmitters for this project were a given due to there already being a few in the lab. These were analogue PAL video transmitters, which meant the chosen cameras also had to be PAL.

Stereo Since this robot was to be a bigger robot than similar projects run in RARL at the time, it was decided to use it to ascertain the feasibility of low-cost stereoscopic video transmission. Since transmitters are relatively costly, it was decided to get stereo video through the one transmitter.

With this in mind, the initial concept was to quarter the screen, as on some CCTV monitors, and send all four camera feeds at once that way. It was quickly discovered that the only hardware available to do this quartering was very bulky, and expensive, as befitted permanent CCTV installations rather than small mobile robots.

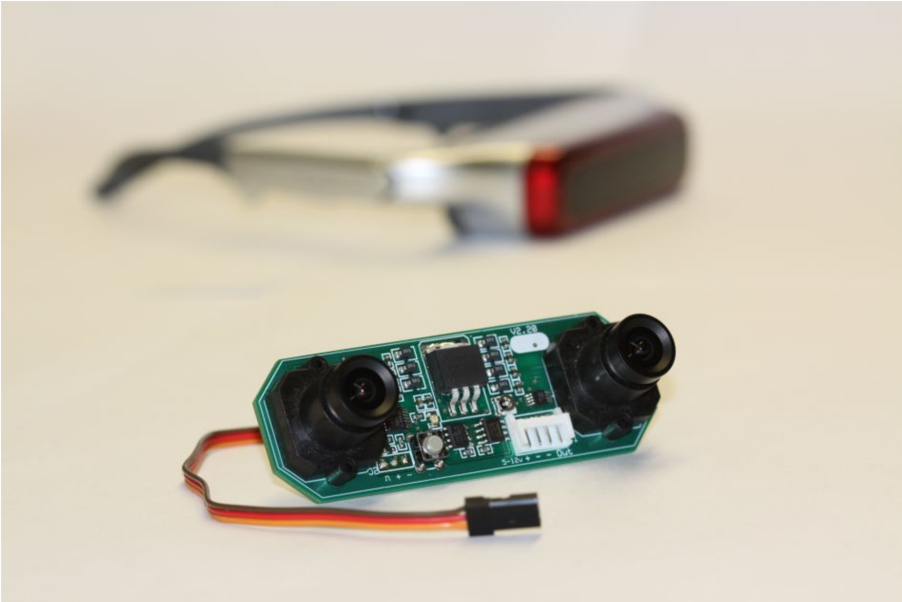


Figure 4.5: The New Generation Hobbies 3D FPV camera [24]

Field-Sequential 3D Turning to the first-person-view (FPV) hobby community, the $330 FPV stereo camera shown in Figure 4.5 was found [24] that used field-sequential stereo. This led to research into this form of stereo which led to the discovery that this system could be implemented with two integrated circuits: a high-speed video multiplexer (MUX) and a video "sync separator".

4.2.2 Stereo Circuitry

The sync separator provides signals corresponding to the analogue video encoding in PAL, and one of its pins goes high for the even field and low for the odd. This pin can therefore be used to drive a selection pin on the MUX. To this end, free samples of a sync separator and a four-channel video MUX were ordered from Texas Instruments.

Testing These circuits were set up on breakout boards and tested on a breadboard. Once the sync separator was correctly wired (the RSet pin was especially important), a perfect 25Hz square wave was extracted from the O/E pin that corresponded to the phases of the video. The MUX pin A0 was driven off this so that the MUX would switch between cameras 0 and 1 when the other select pin (A1) was low, or 2 and 3 when A1 was high.

The MUX output was sent to the video transmitter, received by a receiver at the PC, converted to digital video by a composite-to-USB converter, and displayed with a stereo video media player on the PC. Unfortunately, the USB converters already in the lab turned out to be dropping the second phase of the video — a process called "deinterlacing" sometimes used to stabilise bad signals. For a video stream that made use of both phases, this was unacceptable, so new, higher-quality USB converters were acquired.

It was noted that the new video converter pulled the output on the field-sequential circuitry too high, and therefore disrupted the sync separator operation, rendering a very distorted signal by the time it reached the PC. The radio transmission, when introduced, had the beneficial effect of acting as an large buffer that rectified these issues.

Simplification and Amplification It was then realised that front and rear cameras need never be on simultaneously, and could therefore be connected to the same two MUX inputs. This also simplified sync separation, as the input to that chip could be taken from the common "video 0" pin on the MUX. It also dispensed with the need for impedance matching on the receiving side of the MUX (PAL video uses a 75Ω line impedance) since the unused camera naturally matched the used camera's impedance.

Unfortunately, the video output under these conditions was rather dim, so the MUX amplification option was used to boost the output. It was noted that the MUX had frequency response characteristics based on the resistors chosen for this gain, so while the ratio of resistors determined the level of gain, the actual values of the resistors had to be chosen by consulting the MUX datasheet.

Audio was much simpler — audio superimposes without the need for synchronisation, so all audio lines could simply be connected directly to the transmitter at once. To simplify this further, audio was only taken from one camera from each end of the robot.

Physical Dimensions The physical dimensions of the board were set to accommodate the breakout boards used for testing since there were no extra samples to attach to a new board. Also, the front and rear video inputs were placed on opposite ends of the board to reduce wire bending in the assembly.

Power Supply It was discovered that the cameras that were bought and specified at 6-12V actually had 5V DC regulators built into their wiring and could, in fact, be operated at 3V, so the extra wiring and DC regulator were stripped off to save space and the cameras connected to the 3V rail.

Configurations For versatility, the sync signal between the sync separator and the MUX was interrupted with a jumper that could be moved to connect the MUX selection pin rather to a buffered I/O pin from the microcontroller if full resolution mono video were ever desired.

4.2.3 Video System Design Summary

In summary, front or rear camera selection was done by the switchboard (see Section 4.3); both left cameras directly connected and both right cameras were directly connected; only 2 channels were actually used on MUX; audio was all connected to one point; the video board ran off 12V and the cameras off 3V; the system can be manually switched between left and right inputs if the jumper is transferred to its alternate position.

4.3 Electrical Design

Space was limited, despite the robot's wheelbase, due to the very narrow body. Thus, it was chosen to have multiple electronic modules that were attached to the main body to provide the require payload volume.

4.3.1 Printed Circuit Board Design

GT16A Board The GT16A microcontroller board was designed to be a general-purpose board that could be used by the lab for multiple projects, and was derived from an existing board used by RARL. All connectors were replaced by pin headers and all resistors, etc, connected to the microcontroller's pins were removed to make the board as non-specific as possible. The board was, however, equipped with its own 3V regulator and pins were attached to take this rail out to other components. The controller board's actual layout was done by Ms Tracy Booysen.

Switch-board Because the controller board was so non-specific, a switch board was required for enabling power switching on the cameras and any future modules, like headlights, that the microcontroller would need to operate. The switchboard was initially designed to run four pairs of devices, duplicated front and rear of the robot, with a single "front/rear" switch pin. A second iteration, which was preferred in the end, dispensed with the pairing and front/rear pin, settling for eight separate "channels" — a much more versatile design. Without the logic gates required for the first option, this also made for a smaller, lower-cost board.

Video Video control was given its own board as outlined in Section 4.2.

RS-232 For RS-232 serial communication, a pair of modules were acquired that interrupt the serial lines in such a way that the two original boards will not "notice" the difference. This was very convenient, as it meant these wireless modules could be configured and then simply left on connected to the microcontroller board with no extra coding required.

Power Regulation For 12V regulated power for the cameras and transmitter, a self-contained buck-boost unit (the SCW05A-12) was acquired from Mean Well [32].

Motor Controllers Brushed DC motor controller (H-bridge) boards were acquired from Pololu.com. Initial motor testing indicated there would be large current draws when the robot hit obstacles — in the region of 10A and above. To accommodate this, the drivers that were chosen were models that could supply 12A continuous and 30A peaks. The boards contained all the additional circuitry required to interface directly with 3V microcontrollers, and even had LEDs to indicate output speed and direction when no motor was attached. These conveniences meant that no additional circuitry was required and the I/O pins of the GT16A board could be connected directly to the motor controllers.

Manu-facture The various printed circuit boards that were designed were outsourced to Beta Layout for printing.

4.3.2 Layout

The naïve initial plan was to fit 12 AA batteries and all the electronics in the central body pipe, with perhaps the option of putting the motor controllers in the end sections behind the motors. It very quickly became apparent that the batteries, at least, would not fit, especially when the chosen batteries were updated to the fatter CR123As. However, it was only when all the electronics were physically built that a practical layout could be achieved.

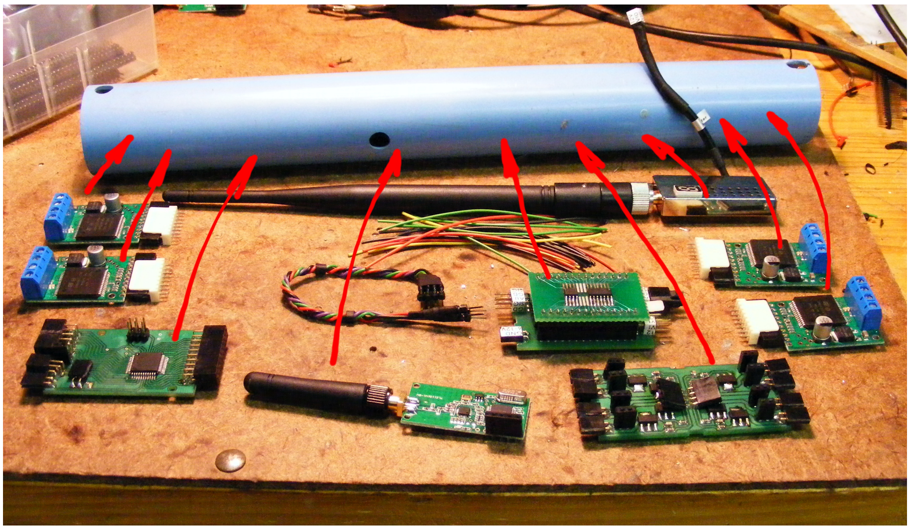


Figure 4.6: All the primary electronics with the insufficiently roomy chassis pipe.

When all the boards were in hand and could be handled with the central pipe (as in Figure 4.6, it was discovered that the central pipe alone was not room enough for the electronics — especially with the bulk of the video antenna — and the motor controllers would not fit in behind the motors. Unfortunately, by that time it was too late to change the body design; given more time, a flat box may have been a better solution (although the modular system is actually very beneficial as described below). As it was, it was decided to fit the batteries in one auxiliary pod, and the radio modules and 12V regulator in a "comms" pod, as roughly shown in Figure 4.7. The pods were made from the same PVC piping as the body, and capped with the battery caps from the screwdrivers that were used for drive.

The comms pod interfaced with the body via a 20-pin header with pins as shown in Figure (insert figure). On the body side, the BDM pins were connected to the micro-controller board while on the comms pod side, those pins were disconnected. This would allow programming and debugging of the robot in the field with only the removal of the comms pod rather than total removal of the electronics.

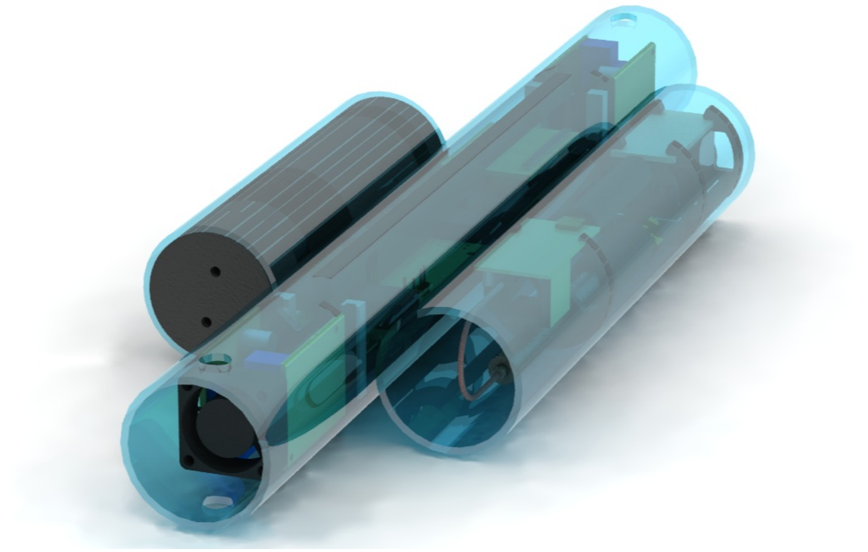


Figure 4.7: The electronics pods in an early layout. Left-to-right: battery pod, body pipe, comms pod.

4.3.3 Electronics System Design Summary

The electrical system was housed in the chassis pipe and two external modules. It comprised a number of small, specialised boards rather than large, combined boards. These were: a microcontroller board, a switchboard, a video control board, an RS-232 wireless module, a 12V buck-boost regulator, and four H-bridge boards.

4.4 Programming

With LabView being used for the PC software and C for the microcontroller, there are a variety of syntaxes that would make code-level explanation simply frustrating in this report, so algorithmic descriptions will be used where possible. In general, the PC ran thread-based code (independent, simultaneously executed instruction streams) while the microcontroller ran interrupt-based code (a main loop that is paused whenever an event calls for a special handling function and then returned to once those handlers are complete). The full set of software is shown in Figure 4.8

4.4.1 Controller Interpretation

While skid-steering drive systems would usually require only two axes of control (forward/reverse and turn, or one axis per motor), the Algodoo simulations of this robot, as shown in Figure 4.9 indicated that having independent control over the front and rear sections of the robot was necessary when dealing with steep obstacles, necessitating some extra layer of control. A few solutions to this were considered — completely independent joysticks for the front and rear sections as in Figure 4.10(a); a single joystick with an extra axis for proportionately balancing control between

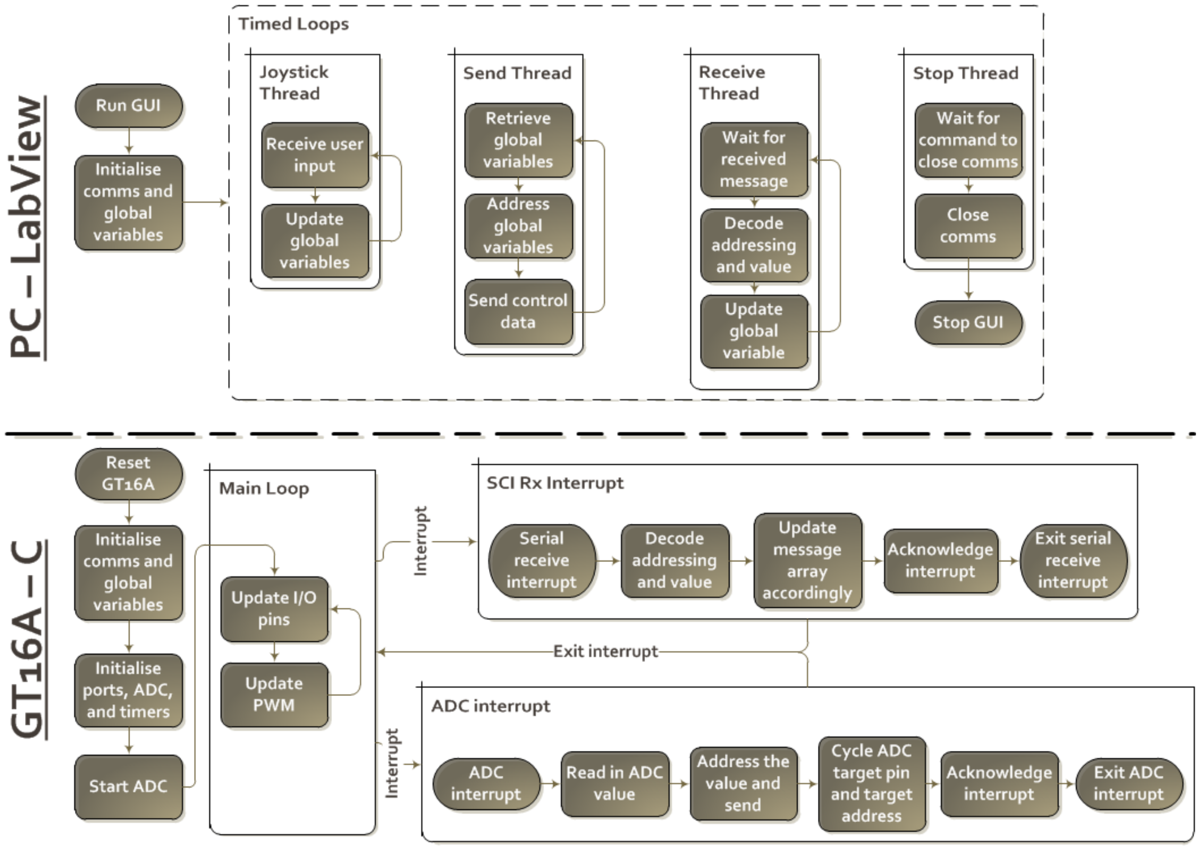


Figure 4.8: A flow diagram of the full software set. The thick dotted line separates PC and GT16A code.

the front and rear systems as in Figure 4.10(b); or a hybrid of those two. For convenience in development and for universality, it was decided to use a standard gamepad or XBox 360® controller for analogue user input. With these, it was decided to use the hydrid of the two solutions above, as described below and shown in Figure 4.10(c).

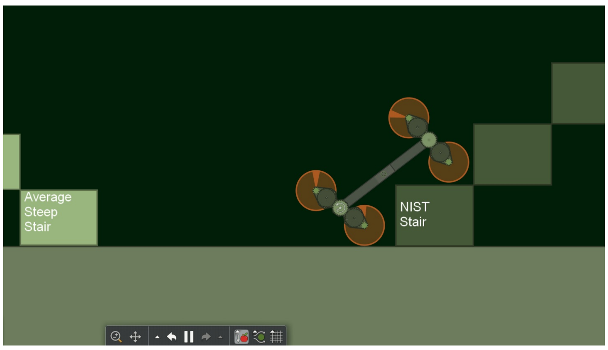
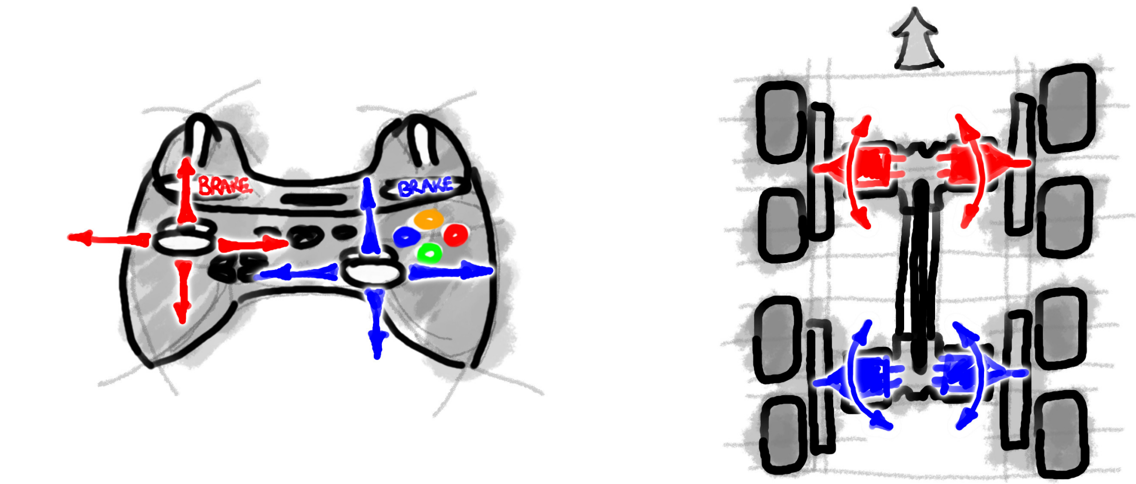
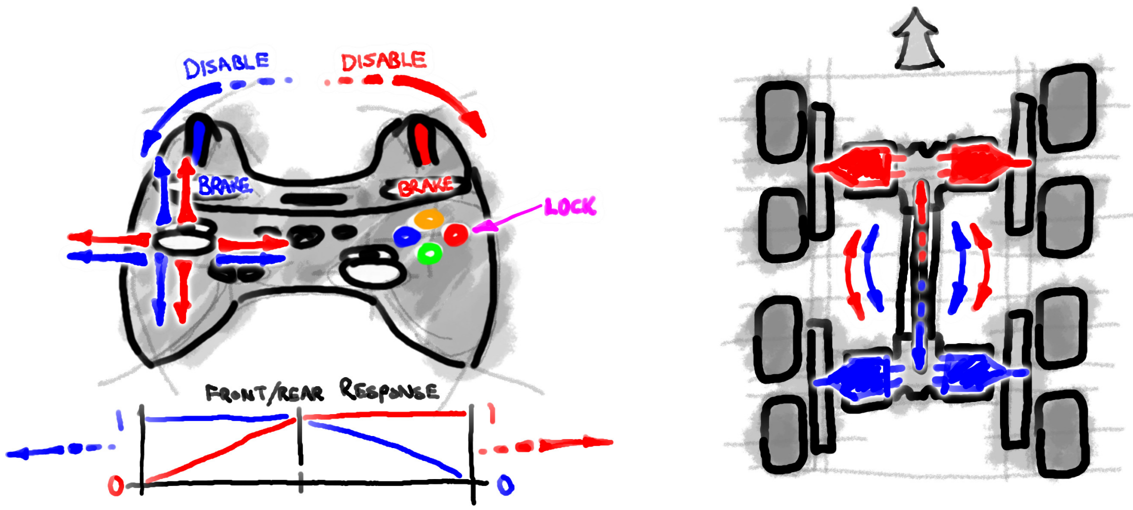


Figure 4.9: An Algodoo simulation, climbing a standard NIST stair.

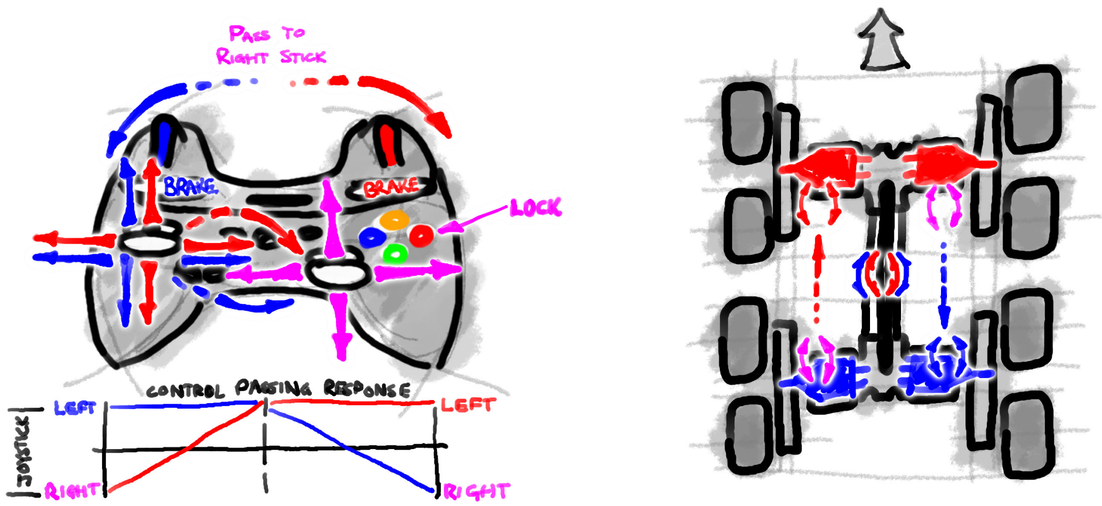
For driving over level ground, independent front and rear control is excessively complicated for the operator. The chosen control layout, therefore, uses one primary joystick (the left stick) for all motors unless explicitly told otherwise. For climbing



(a) Mode 1



(b) Mode 2



(c) Mode 3

Figure 4.10: Control mode concepts 1 to 3

obstacles, the front or rear motors can be "handed over" to the right stick by pulling the right and left triggers respectively. For instance, pulling the right trigger 60% means that the front motors receive 60% of their input from the right stick and 40% from the left stick and the rear motors continue to receive all their input from the left stick. On 4-axis gamepads, only 100% passing is possible, so pulling the right trigger means the right stick simply controls the front and the left stick simply controls the rear.

Each joystick is translated into motor power levels by adding the vertical axis of the relevant stick to both motors and by adding the horizontal axis to the left motor and subtracting it from the right. This means fully forward results in equal forward, fully left or right results in motors turning equally in opposite directions, and each 45° diagonal corresponds to only one motor being on.

4.4.2 Communication Protocol

This robot was designed to use a single serial wireless transceiver for control communication, and so this required a messaging system that could distribute and collate messages to and from a variety of modules of code. That is, as opposed to a system wherein each message (for instance each motor power level) has its own channel as in some analogue radio control applications.

The RS-232 serial communication system (the physical communications layer used for this project) sends 8-bit (1-byte) data units. These are usually used in one of two ways: pre-determined groups of bytes, called packets, that rely on synchronisation of the transmitter and receiver; or packets that contain start characters, stop characters, information regarding message length, and error checks. The first form is simple, but unreliable, since a loss of synchronisation results in a completely garbled message. The second form is far more complex, but once successfully implemented is far more reliable. There are many standardised forms of this character-based communication, since most digital communication, from video to e-mail, is done via serial, but many are very complex, designed for large quantities of data, as opposed to the very small volumes required for this project.

This project only required the messages listed in Table 4.1.

As noted in the table, some values are simply boolean true/false values, requiring only single bits. The integer values that are sent are either user commands or rough indications of the load on the motors, so can afford to be of slightly lower resolution than a full 8 bits would afford, especially if this improves communications.

With this in mind, a much simpler communication protocol was implemented for this robot wherein each byte contained 3 "address" bits. This limited the address to values 0-7, thus limiting the amount of information that could be sent to 8 5-bit numbers, but also added communications redundancy — if one byte was lost in the transmission, the other bytes would remain unaffected, unlike with character-based systems where errors invalidate entire messages. In theory, this also allows certain bytes to be sent at different frequencies to others (to update the motor speeds more frequently that the camera on/off, for instance), or for certain bytes to be left out if unchanged for some time.

Thus the full array of possible messages, translated from Table 4.1 into the addressed format, is shown in Table 4.3.

For the prototype, a simplified version was used, with no confirmation bits, and only one general "video enable" bit, rather than independent camera control.

With this system, motor speed could be given from 0-15 in either direction, and converted to a pulse-width-modulation value on the microcontroller, as explained in Microcontroller Code, below. Current levels, as simple magnitudes, could be given from 0-31 and interpreted by the UI to give an indication of motor load to the user, as described in Testing User Interface, below.

4.4.3 Microcontroller Code

The actual code on the GT16A microcontroller was fairly straightforward, since most of the complicated processing was done by the controller PC. Joystick interpretation and conversion of the current readings to amperes was done on the PC, and the only thing the microcontroller had to do to the video was turn it on or off.

At the most basic level, the microcontroller controlled the voltage across each motor, switched on and off the various circuits in the robot, and compiled and sent or received and decoded messages from the controller PC.

Communication As described under the Communication Protocol section, above, the microcontroller received bytes of data which contained their own address information, and similarly sent messages to the PC with addresses appended. Since any byte received would be one of eight possible messages, an array of eight bytes was created to receive these messages.

When a byte was received by the communication port on the GT16A, an interrupt was generated. In this interrupt, the received data was saved to a temporary variable which was then separated into its address and data. The data could then simply be placed in the array at the address position.

All actions to be taken on the information received were handled by the main control loop which ran whenever the controller was not in an interrupt.

Motor Control Since there were four motors to control and the GT16A does not have any built-in digital-to-analogue converter, never mind four, pulse width modulation (PWM) was used. The GT16A has two timers, each capable of generating two independent PWM signals, giving four in total — the exact number needed.

PWM generates a square wave as shown in Figure 4.11 where the width of the high portion relative to the low portion determines the average voltage across the motor. So,

$$V_{avg} \approx V_{max} \cdot \frac{TPMXCnV}{TPMXMOD} \tag{4.5}$$

This is done by the timer modules on the GT16A, and does not use CPU time unless a value is to be changed. For four motors, both timers were set to edge-triggered

Table 4.1: Full list of required comms for low-cost robot

	Variable	Type	Note
PC to Robot	Motor velocity 0	Signed integer	Can be low-resolution (4 or 5 bits)
	Motor velocity 1	Signed integer	
	Motor velocity 2	Signed integer	
	Motor velocity 3	Signed integer	
	Forward cameras On/Off	Boolean	Merged for prototype due to there being no rear cameras.
	Video circuits On/Off	Boolean	
	Rear cameras On/Off	Boolean	Not used in the prototype due to faulty rear camera and lack of mono-mode testing.
	Left/Right Camera Select	Boolean	
	Request confirmation	Boolean	Not used in prototype for lack of time
	Confirm received	Boolean	
Robot to PC	Motor current 0	Unsigned integer	Can be low-resolution (4 or 5 bits)
	Motor current 1	Unsigned integer	
	Motor current 2	Unsigned integer	
	Motor current 3	Unsigned integer	
	Request confirmation	Boolean	Not used in prototype for lack of time
	Confirm received	Boolean	

Table 4.3: Communications Message Map

	Address			**Data**				
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
PC to Robot	0	0	0	Dir. 0	Motor Velocity 0			
	0	0	1	Dir. 1	Motor Velocity 1			
	0	1	0	Dir. 2	Motor Velocity 2			
	0	1	1	Dir. 3	Motor Velocity 3			
	1	0	0	Unused	Front Cam	Rear Cam	Video	Left/ Right
	1	0	1	Unused				
	1	1	0					
	1	1	1	Unused	Unused	Unused	Request Conf.	Confirm Received
Robot to PC	0	0	0	Motor Current 0				
	0	0	1	Motor Current 1				
	0	1	0	Motor Current 2				
	0	1	1	Motor Current 3				
	1	0	0	Unused				
	1	0	1					
	1	1	0					
	1	1	1	Unused	Unused	Unused	Request Conf.	Confirm Received

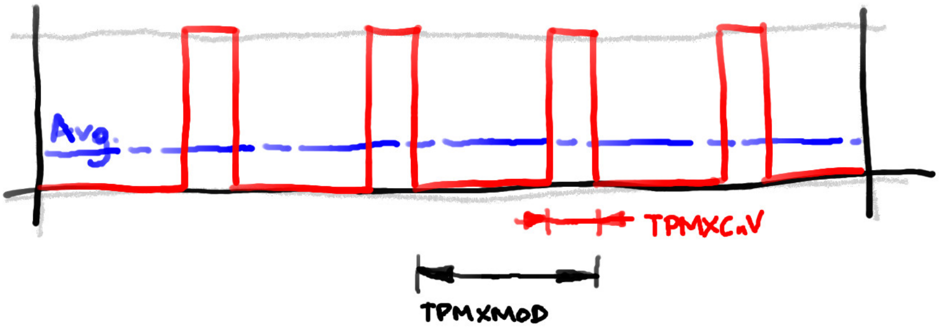


Figure 4.11: A PWM signal (red)

PWM mode at the same frequency (20kHz to be as close to outside the audible range as possible but within the capabilities of the motor control H-bridge circuits to handle), and each of the four channels was given a value corresponding to the motor control input.

The value for each channel was interpreted from the first four bytes of the array of received messages (see Communications, below). There were two main options for this interpretation — calculation or a look-up table. Since the range of possible values was relatively small, and the response curve was unlikely to be dynamically altered while in operation, it was decided to use look-up tables rather than calculation. This would also take less time than calculation, since retrieving the required two bytes from memory would only take around 4 clock cycles where each of the multiplication or division operations required in calculation would take at least that [33]. The look-up table values can be easily altered by the programmer to non-linear or even non-quadratic curves if needed to fine-tune the motor response.

The GT16A cannot handle PWM levels very close to either zero or full pulse, since this does not give the timer a chance to toggle. So, using Microsoft Excel to calculate the required table values, the look-up table was set up for a minimum of 8 and a maximum that was 8 shorter than the timer modulo value. Several curves were calculated and all included in the code, with the unused ones commented out to make them easier to access if a change were required.

With the look-up table, the value for the motor speed stored in the array of received messages was used as the address in the look-up table from which to pull the channel value. The direction of each motor was determined by two pins that had to be the inverse of each other to enable the motor in either direction. These pin values were taken from bit 5 of each motor's byte in the message array. This resulted in the three simple lines of code below for each motor:

```
//PWM set
  //Motor 0
TPM1C0V=Curve[RecArray[0] & 0x0F];   //Call power curve value
PTAD_PTAD0= (RecArray[0] >> 4) & 1; //  Set direction 0
PTAD_PTAD1 = ~PTAD_PTAD0;           //  Set direction 1 to opposite
```

Communications For each byte received, the address was extracted and used to write the received byte in its entirety to the respective byte of the message array. This was accomplished by bit shifting and masking as below:

```
address = ((data >> 5) & 7);  //Pull out the address
RecArray[address] = data & 0x1F;  //Put the value into the "received" array
                                   // at the corresponding address.
```

Current Sensing The "Current Sense" pin on the H-bridges provided an analogue voltage proportional to the current drawn by the motors at any one point. Thus, to be interpreted by the GT16A, the analogue-to-digital converter (ADC) on the GT16A had to be used. The ADC has up to eight inputs, but can only read one at a time. Thus, to read all four, the ADC had to cycle through the respective pins each time it completed reading one.

The ADC, in the setup used, gives a full 8-bit reading (0-255), which had to be scaled down to the 5 bits that could be sent. The ADC cycle number could also be used as the address to be appended, and then the reading immediately sent back to the PC. This was simply done in the two lines below, although the temporary variable could even have been excluded and the whole current sensing process dealt with in one line.

```
tempCS = (ATDRH / 8) + (CScycle * 32);  //Read ATD, shrink to 5 bits,
                                         // and append address.
                                         // CScycle is the cycle number.
SCI1D = tempCS;  //Send value back to PC (write to the SCI data register).
```

4.4.4 Testing User Interface

The user interface for the robot was designed in National Instruments' LabView, a graphical programming language that generates graphical user interfaces (GUIs) very quickly and simply.

A PC, as opposed to the microcontroller on the robot, can handle multiple "threads" — streams of code executing on independent timing — which is useful for user interfaces. Since joysticks' analogue inputs must be polled, joystick handling would use up too much CPU time on a single-threaded program. Similarly, waiting to received messages on the serial line would hold up everything else in a single-threaded program. For this robot's user interface, four of LabView's "timed loops" were used for the four threads — Joystick, Comms Read, Comms Send, and Comms Stop (as seen in Figure 4.12). Variables were shared between the threads so that, for instance, motor levels from the joystick thread could be sent by the comms send thread. Initialisations, such as opening the serial port and setting up the received message array, were done outside the loops so that they only executed once at the beginning of the program. Assigning the threads to available CPU cores was handled automatically by LabView.

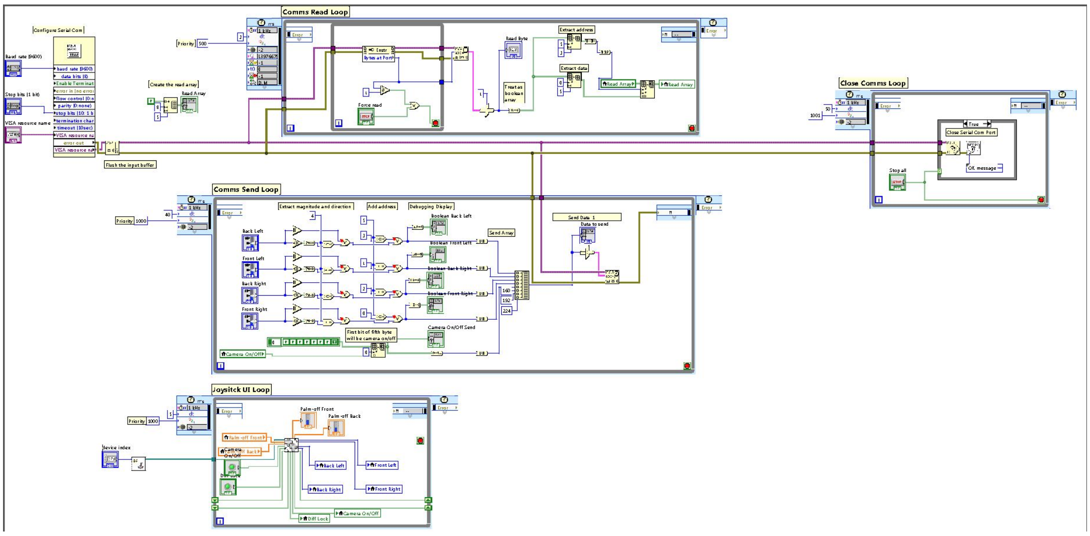


Figure 4.12: A shrunk view of the LabView code for the testing UI, showing the four thread loops, seen as four large boxes in the code.

The actual front-end of the UI — the GUI in Figure 4.13 — was kept very basic for testing, since it was necessary to see exactly what data was being sent and received, and where that data was going. Thus, while in a public release there may be indicator bars for motor levels and current readings, for the testing GUI the sent and received data was displayed as arrays of LEDs so that each bit could be individually examined.

For expediency of testing, LabView was not used for the stereoscopic video processing, however. Stereoscopic Player, a specialised third-party stereoscopic media player was used instead, since it already had all the required features — the ability to decode any stereo input, including field-sequential PAL video, and display it in any output, from mono to anaglyph (red and green), side-by-side (shown in Figure 4.14), or row-interlaced stereo as used to display on the LG D2342 3D monitor used for testing.

Both programs — the LabView VI and Stereoscopic Player — had to be run simultaneously for testing, as shown in Figure 4.15.

4.5 Design Summary

The robot was designed in four primary sections — mechanical, video, electrical, and programming. As much as possible, modular designs were used, from the four repeated LIMs to the separation of circuit boards to the multi-threading of the LabView program.

In short, the robot had eight wheels on four LIMs, driver by four re-purposed cordless screwdrivers. These were attached to a PVC chassis which also housed the cameras and other electronics. Two pod modules were used to expand the space available for electronics. The electronics comprised principally of power regulation and distribution, a microcontroller, a video system, four H-bridge boards, and wireless modules for data

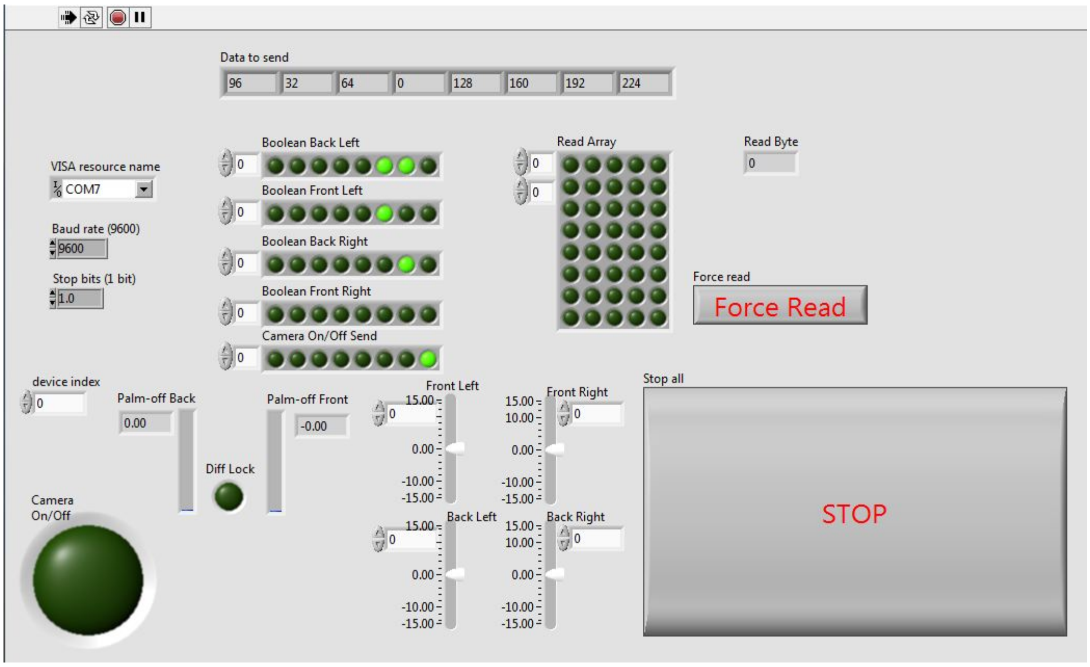


Figure 4.13: The LabView GUI used for testing. The lower left section relates to joystick control, the upper left to comms sending, the upper right to received information, and the large "STOP" button to closing the serial port.

Figure 4.14: Stereoscopic Player displaying right and left camera images respectively for cross-eye "glassesless" viewing as used in early testing

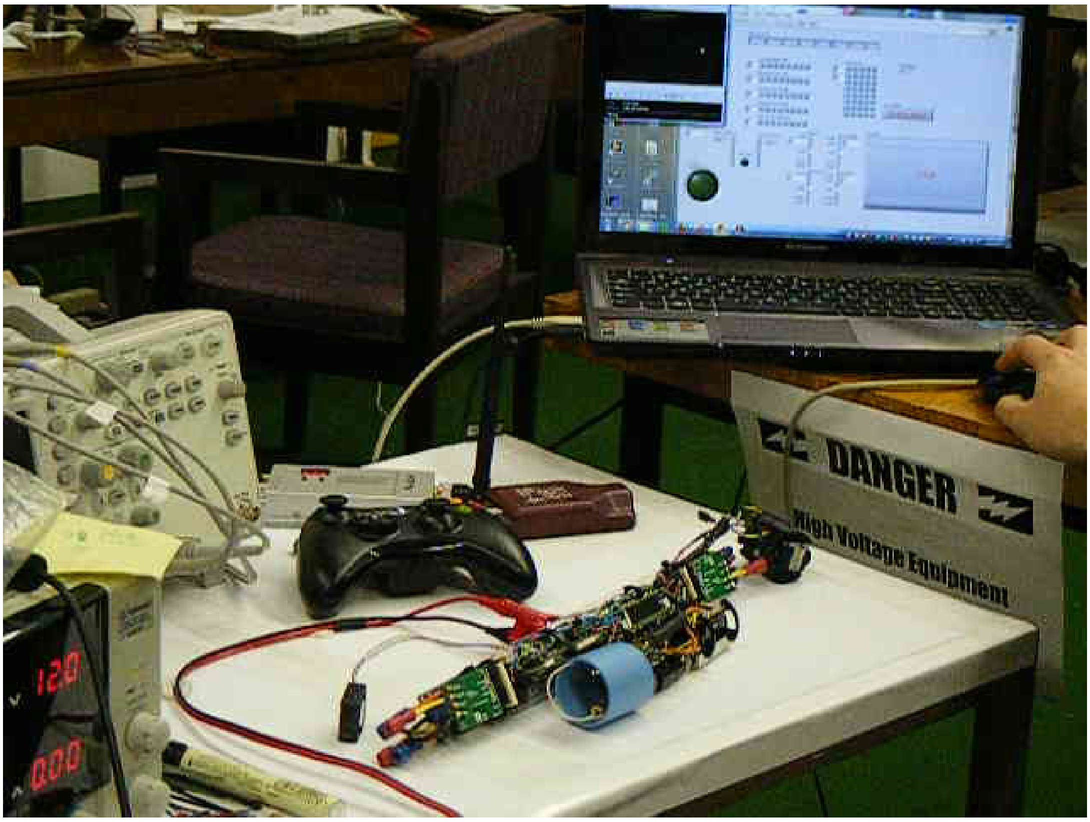


Figure 4.15: The complete testing UI: the XBox controller and PC, with both LabView and Stereoscopic Player open. This is a frame from a video of the electronics system testing.

and video. The UI on the operator PC was designed to take joystick input, translate it into usable commands for the microcontroller, send and receive data, and display the received data. The microcontroller was programmed to convert the received commands to applicable outputs, take in sensory data, and send that back to the operator PC.

Chapter 5

Testing

A variety of tests were designed and performed for this project. Some were to ascertain adherence to the specifications laid down in Chapter 3, and others were simply testing whether individual components worked at all.

5.1 Assembly Assessment

Alignment During assembly, note was taken of the occurrences of misalignment and what caused these — specifically which production processes — to ascertain requirements for refinement in future iterations of the project.

Access While every effort was made during the CAD stage to ensure the robot could be assembled and disassembled easily, it was only in the actual assembly and disassembly that the efficacy of those efforts could be assessed.

5.2 Subsystem Testing

5.2.1 LIMs

Motor Torque A motor was tested to determine the maximum torque it could supply before an electrical or mechanical limit was reached. A 600mm ruler was attached to the output shaft and a 500g mass suspended from the end of the ruler. A power supply was attached to the motor set to 12V. The current was then increased until the motor failed or could lift the mass.

5.2.2 Video

Positioning A pair of cameras was set at a variety of separations on an adjustable bracket to determine the optimal separation for viewing comfort, given the proximity of the cameras to the ground during operation of the robot. The cameras were gimballed to allow directional adjustment. The screws on the gimbals were adjusted until the images from the cameras were parallel and thus suitable for stereoscopy. (include figure of CCD misalignment) Once positions had been determined, fixed brackets could be made for the correct spacing and alignments.

5.2.3 Electronics

Switchboard Each circuit on the switchboard was connected to a 12V supply in turn and a 100kΩ resistor was connected across its output terminals. The voltage across the resistor was measured with an oscilloscope while a 3.3V trigger signal was applied to the "data" input for that circuit. The resistor and testing connections are shown in Figure 5.1.

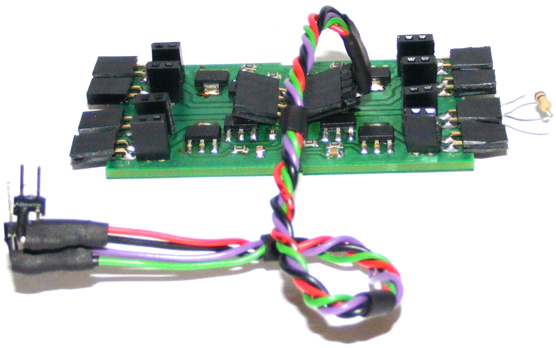


Figure 5.1: The switchboard with the 100kΩ resistor (right) used for basic connection testing.

Sequential Power Tests The complete system was powered up sequentially from the H-bridges through to the GT16A microcontroller board, with each stage tested for correct operation:

Each H-bridge board was supplied with voltages ranging from 5V to 18V, with the reverse-protected output connected to an oscilloscope to test whether the reverse-protected output was passing the voltage correctly. The input polarity was then reversed at 5V to test each board's reverse-voltage protection.

Once reverse voltage protection was confirmed, the next stage in the power sequence was connected — the 12V buck/boost regulator. The permanently connected fan was used as an indicator that power was supplied, and an oscilloscope was placed across the output to take accurate readings. The input voltage varied from 5 to 16.5V and the output noted at those levels, including the low voltage at which the supply could no longer provide 12V.

Once correct power regulation was confirmed, the RangeVideo transmitter was connected to confirm that it powered up from the regulator and to check what current it drew. It was then disconnected for the rest of the tests to lower the current draw while total current limits were unconfirmed.

The GT16A microcontroller board was then connected to the 12V regulator and the 3V output and indicator LED were checked for correct operation.

Once the microcontroller board was determined to be working correctly, including being flashed with code, the switchboard was connected to power and the microcontroller and each connected output was tested for the correct voltages when the GT16A toggled the respective data pins.

The video circuits were connected to the switchboard and transmitter and tested to confirm whether they continued to operate once actually installed in the robot.

Range The wireless serial modules were progressively separated while transmitting data until data was no longer received. A laptop PC and a battery-powered microcontroller were each connected to a transceiver, and a loopback program was run wherein any byte sent by the PC was incremented by the microcontroller and returned to the PC. A surveyor's wheel was used to measure distance while the microcontroller was separated from the PC until the loopback program no longer returned bytes.

5.2.4 Communication

Loopback The first test of communications was to get the GT16 to send back whatever it received immediately. This was done with the LabView test VI that addressed each byte to be sent and also sorted the received bytes by address, so this tested both the addressing system and whether there was communication at all.

5.3 System Testing

5.3.1 Basic Measurement

The robot's box dimensions were measured with a tape measure, and the robot was weighed on an electronic scale.

5.3.2 Performance

Running Off Power-Supply The motive characteristics of the robot were first tested by simply connecting the motors directly to power supplies, before the motor controllers were tested for functionality. Both a full assembly and a tail-dragging half-assembly were tested in this way. They were driven forwards and backwards, presented an obstacle, and made to attempt to skid steer by reversing the direction of one motor.

Turning Turning capability was evaluated by attempting turns with a the robot on a power tether.

Stair Climbing The robot was presented with a stair-sized obstacle and the voltage and current required to climb that obstacle were recorded.

Level Speed The robot was driven at various power levels over 3m of flat ground while being video recorded. The footage was then analysed to determine the robot's speed and acceleration characteristics.

5.4 Costing

As comprehensive an expenditure list as possible was generated to assess the cost of the prototype and area for cost-reduction in future iterations.

Chapter 6

Results and Recommendations

The results of the various tests give indications of everything from ergonomics to performance. Some results highlighted issues, and some tests suggested other ways in which the design might be improved. In each of the following sections, the results for each set of test is given, followed by any recommendations there may be to improve related aspects of the design.

6.1 Assembly

6.1.1 LIMs

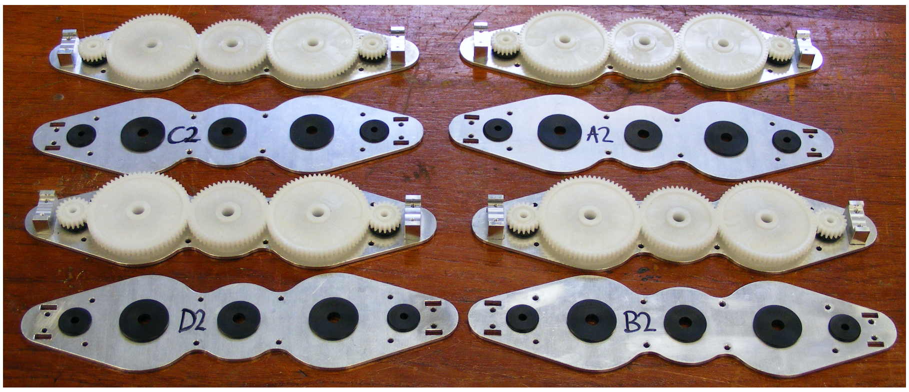


Figure 6.1: The side walls of LIMs with the gears, bushings, and brace blocks to fit them.

LIM Gearboxes A last-minute change from laser cutting to CNC milling of the side walls of the LIMs, shown in Figure 6.1, meant that the holes for the brace blocks were produced with rounded edges that had to be filed out, as shown in Figure **??**. This hand-filing led to slight misalignments of the side walls that aggravated other alignment issues with the gears. The primary issue with the smooth running of the gears, however, was slight variations in the gear moduli as a result of their moulding process — certain gears meshed too tightly. This primarily occurred with the largest gears.

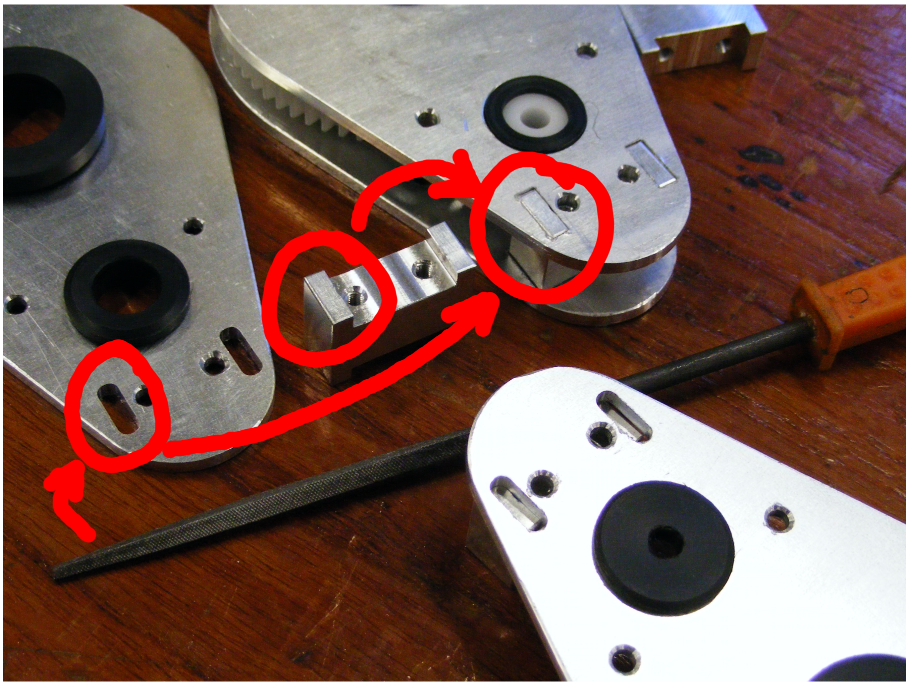


Figure 6.2: The brace blocks and the holes they were meant to fit. The small square file was used to square the edges for the fit shown on the right.

For the prototype, this was accounted for by reaming out the mounting holes with adjustable reamers until each ran just smoothly enough to be turned by hand from the input shaft. For further production, this laborious process would be infeasible, so it would be worth designing the inner three holes, of the five, to be freer running than they were for the prototype — perhaps by as much as 0.1 or 0.2 mm for the input gear. With that room, gears could be shuffled to match free-running and tight-running gears so that the input gear would naturally take up an offset to compensate, as occurred after the reaming of the prototype. (illustrative figure, perhaps also photo)

As for the brace blocks, the choice to change production to CNC milling from laser cutting was due to the required gear shaft alignment accuracy, which the CNC mill was better suited to achieve. Therefore the brace blocks and their mounting holes should be redesigned with CNC milling in mind.

Mounting LIMs

Mounting the LIMs to the end assemblies was another minor point noted in the assembly. Using M2.5 threaded rod to hold the motor gearboxes made the motors very secure, but there was no feature on the rod with which to twist it, as there would be on a standard screw. Using a double nut was one option that was tried, but this was frustrating due to the number of turns needed to insert each of the eight rods. In the end, it was discovered that the quickest and easiest way to insert the rod was with a hand tap drill as shown in Figure 6.3.

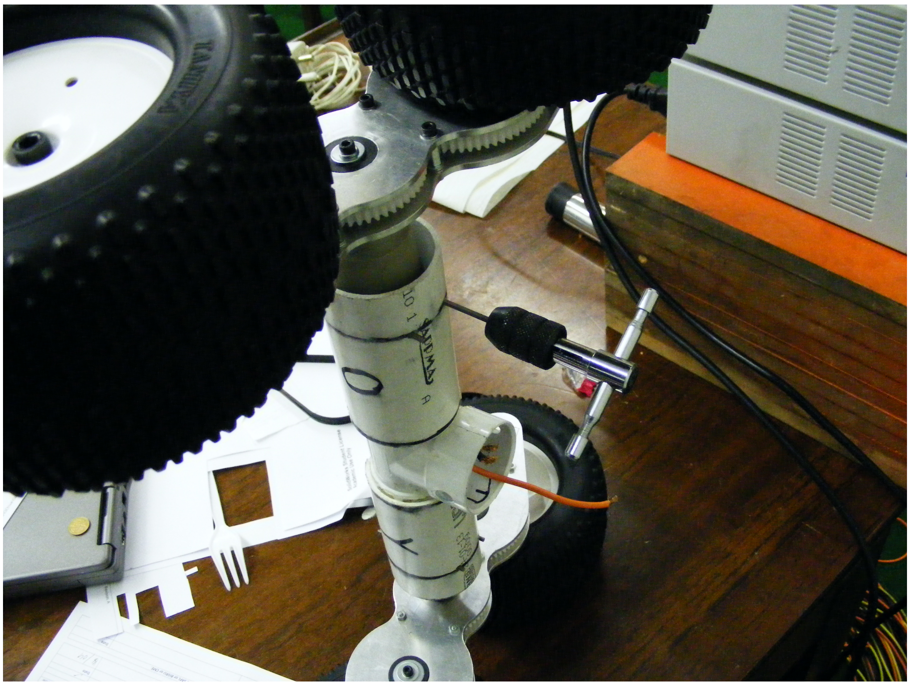


Figure 6.3: Inserting the threaded rod with a hand drill.

6.1.2 Electronics

While the electronics all worked satisfactorily, and could be taken apart easily to access or change components when not in their housing, there was a very large number of connectors and wires to be connected in the assembly. Given the very limited space for electronics, this was quite a challenge — not only to design but also to assemble and fix when there were issues.

Carefully planning the lengths of each wire on a 1:1 printout of the spine assembly drawing was helpful but imprecise, and some wires ended up very tight where others had to be coiled to take up slack. The diagram was also limited to two dimensions, and it was found in the actual assembly that one half, and one quarter in particular, of the tube's cross-section was far more packed with wiring than the rest. Some connections also had to be taken across boards simply because the connector had been arbitrarily placed on that side of the board during the design. Also, since the average board was so small in this project, wire thicknesses were a not-insignificant portion of any cross-section taken through the assembled tube.

In the end, the number of wires and the tightness of the fit of the brackets in the pipe added up to make the electronics fit too stiffly in the pipe. Such a hard push was required to get them in that lubrication was required to remove them again, and the comms pod connection and BDM pins were made inaccessible due to being too tightly wedged in.

The comms pod, on the other hand, was very easy to assemble and disassemble and all fitted very smoothly and freely, yet with a snug enough fit not to rattle around.

So, aesthetically, all the electronics fitted in the main pipe, but functionally there needed to be more room. The main problem was the cylindrical nature of the body, meaning flat components like circuit boards naturally congregated across the diameters. A simple solution to all these problems would be to use a rectangular-section body, even if only for the pods. This would give more room for more electronics to be moved into the pods and relieve the main pipe of some congestion. Making the whole body rectangular in cross-section would be even better, perhaps even making possible a central column of wiring around which circuit boards could be arrayed, making routing so much easier.

Square-section pods would also make it possible to use better batteries. While this project did not get so far as to run with batteries, it was noted (after buying the batteries) that the high current requirements — as much as 50A if all four motors were around full load — were too much for the CR123As, even in two parallel banks, since each could only supply 4A. Hobby-style lithium-ion packs can often go up to much higher currents, but even the smallest high C-value packs would not fit the cylindrical pods as they were for this project. A square section of the same essential dimensions would have accommodated these batteries easily, however.

6.2 Sub-Systems

6.2.1 LIMs

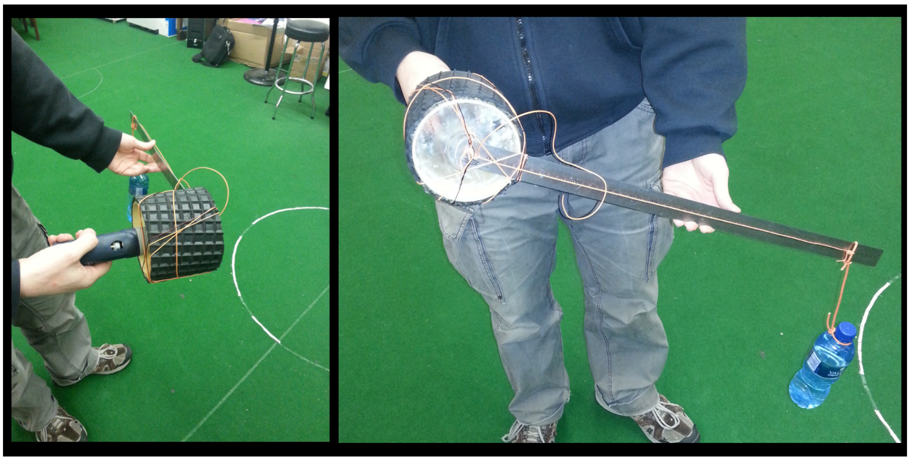


Figure 6.4: Testing a motor for torque characteristics. The 500ml bottle is filled with water.

Motor Torque One motor was crudely characterised by attaching a ruler with a 500g mass as shown in Figure 6.4. It was assumed that the only significant masses were those of the ruler and the 500ml of water on the end. At 12V and 10A current limiting, the motor gearbox slipped. The torque at that point was calculated as:

$$
\begin{aligned}
T &= \sum m_i \cdot d_{CG_i} && (6.1)\\
&= (m \cdot d_{CG})_{ruler} + (m \cdot d_{CG})_{water} && (6.2)\\
&= 0.2kg \cdot 0.32m + 0.5kg \cdot 0.56m && (6.3)\\
T &= 3.36Nm && (6.4)
\end{aligned}
$$

6.2.2 Video

The video circuit performed as hoped, proving more reliable than the breadboard-based prototype. There was a slow creep of the one video signal, due to the asynchronisation of the cameras and the timing being based off the signal from the other, but this was very small, and easily remedied by resetting the video circuit. Under artificial lighting in the evening, and given a poor antenna angle, the video transmission was too dim to be usable, given the original output gain of 2, so the circuit was adjusted to give a gain of just over 4. This made it easier to see under the artificial lighting, but it was noted in earlier tests on a breadboard that this level of gain made daylight conditions overly bright. USAR would seem to justify a bias toward artificial lighting conditions, however.

Using the adjustable bracket and gimbal system, it was found that the CCD sensors[1] in the cameras were approximately 12° horizontally misaligned and 5° vertically misaligned. The horizontal misalignment was compensated for by laser-cutting Perspex brackets that set the cameras at 12° relative to each other, as shown in Figure 6.5, but the vertical alignment was left to the gimbals since it was less drastic.

Testing various separations of the cameras, it was found that the standard human eye spacing of 60-64mm was too great for navigating in close proximity to the ground as the robot would have to do. When the cameras were placed in a position simulating the robot's ground clearance, it was found that a separation of 35mm was much more comfortable to view, but still gave a strong enough stereoscopic effect for distinguishing objects on a cluttered desk.

From misalignment to poor light-level adjustment, these cameras proved somewhat frustrating to use. Improving the cameras should help the alignment issues and make video gain easier to handle — properly aligned sensors would mean a standard bracket could be made with minimal gimbal adjustment required, and better light-level compensation on the camera's part should mean a lower gain could be selected and therefore there would be less of an issue with daylight conditions.

6.2.3 Electronics

Switchboard The switchboard worked as expected — each output terminal presented the 100kΩ resistor with the supply voltage when the data pin was supplied with 3.3V and disconnected the resistor when the data pin was left floating or low. Unfortunately, labelling of the switchboard terminals was done incorrectly on a subsequent occasion;

[1] The video photo-receptor components — analogous to celluloid in film cameras.

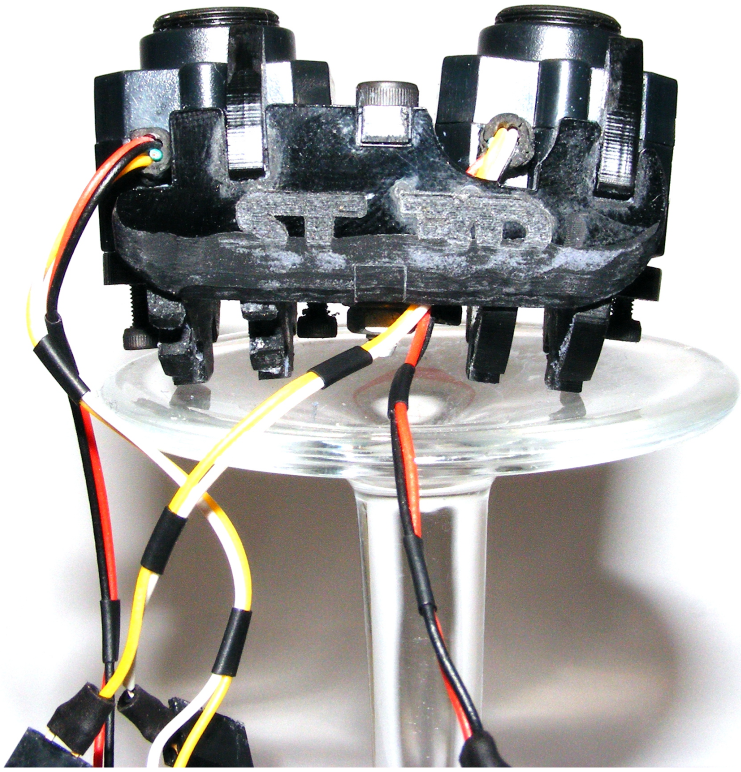


Figure 6.5: The stereo camera assembly showing the required offset angle to make the CCD sensors parallel.

when the switchboard was finally connected to the rest of the circuitry according to those labels, some of the video components were damaged by the applied reverse voltage before the error was detected.

Sequential Power Tests

It must be noted that during the power-up testing, the oscilloscope displayed a constant 600mV when disconnected, so all voltages below are given as 600mV less than displayed.

All H-bridge boards' power regulation worked as their datasheets professed: the V_{OUT} pin supplies the voltage connected across V_{IN} and GND terminals, but with reverse voltage protection; when the supply was connected the wrong way around at 5.5V, V_{OUT} went to a steady -1V as shown in Figure 6.6.

The buck/boost 12V regulator also worked as its datasheet stated: it supplied a steady 12V, as shown in Figure 6.7 with any input from 7.7V up to the tested 18V.

It was discovered that the RangeVideo radio video transmitter drew a wide range of currents and, in fact, caused the voltages on the bench supply to vary depending on the orientation of the antenna, and which antenna was connected. With this in mind, it was decided to test the transmitter in the actual configuration in which it

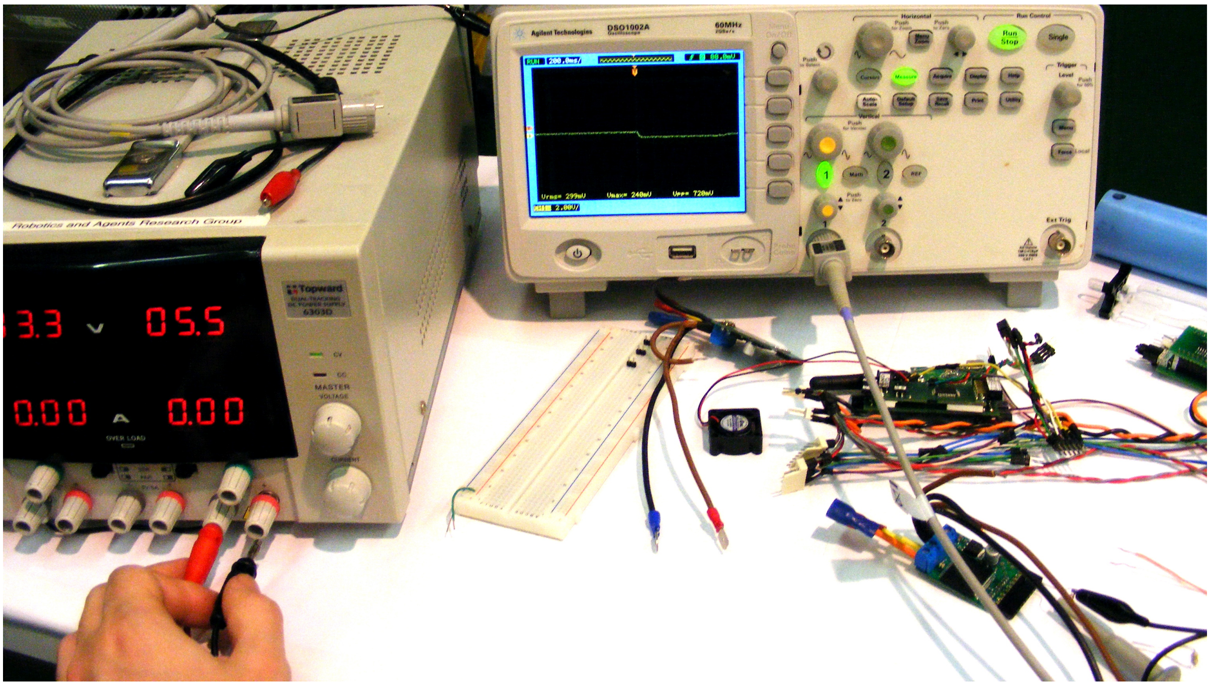


Figure 6.6: Applying a reverse voltage to an H-bridge board. Note the -1V output on V_{OUT} shown on the oscilloscope.

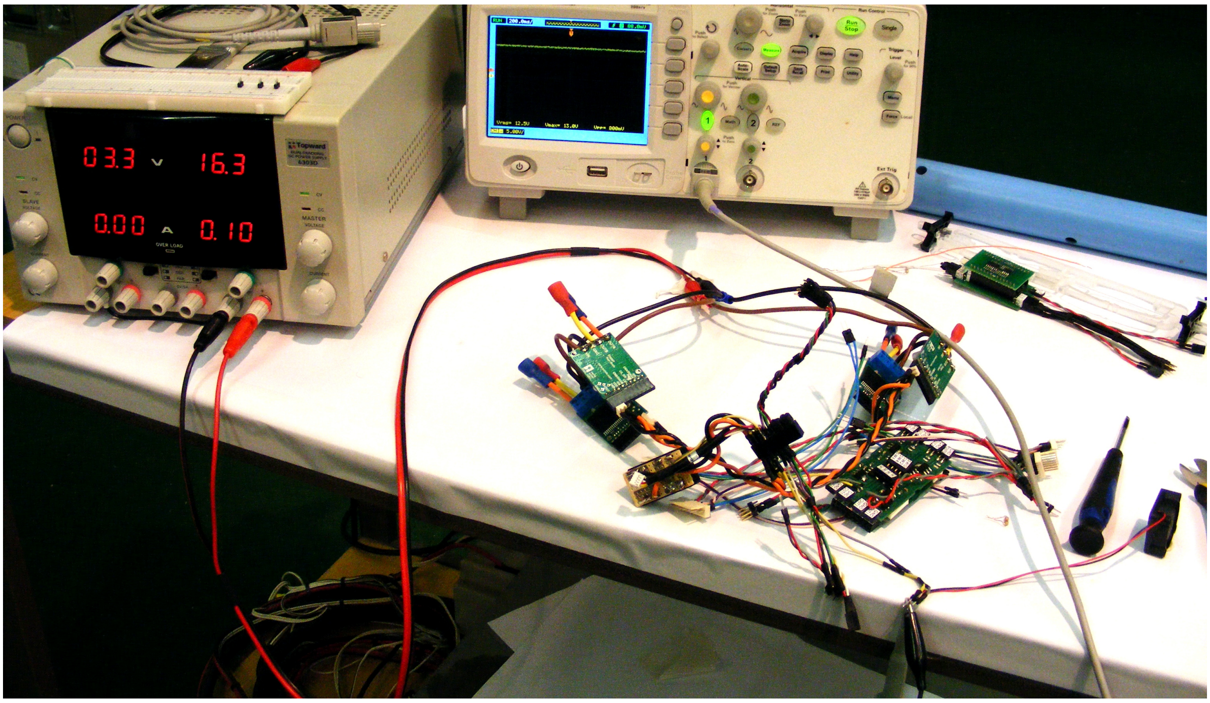


Figure 6.7: The 12V regulator, connected to the V_{OUT} pins on the H-bridge boards, supplying a steady 11.9V despite the 16.3V input

would be used, so the full comms pod was assembled with the antenna that would be used on the final assembly. In this configuration, the total current stabilised at under 0.5A, with the displayed bench supply voltages sitting at 12.1 and 3.5V on the 12.0 and 3.3V rails respectively.

While this setup worked, the phenomena described above could not be explained with the knowledge-base applied to this project. It is therefore recommended that further dedicated research is done into refining the video transmission system if, as is so, a robust and stable system is the desired final product.

The GT16A microcontroller board turned on as expected and its power regulator supplied a steady 3.1V. The indicator LED dimmed under low voltage supply and remained constant when the board was correctly powered. With the GT16A connected and the video transmitter disconnected, the system drew 0.14A.

Range The range testing of the RS-232 radio-frequency transceiver was carried out by Messrs James Stock and Adam Healy. The test was done using the 10dB power setting on the modules, and with a variety of antenna orientations. The results are summarised in Table 6.1, and the equipment shown in Figure 6.8

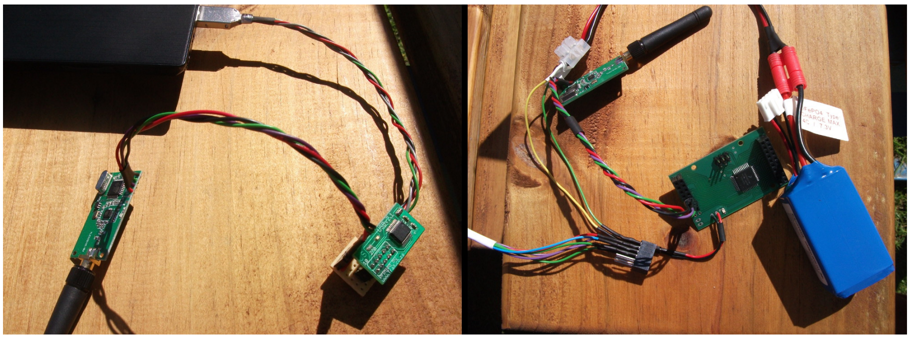


Figure 6.8: The range testing equipment — the computer with transceiver, and the GT16A with transceiver and battery.

These ranges were less than hoped for, although it is hard to translate line-of-sight, open-space figures into indoor, out-of-sight figures. It is recommended, at least, that further testing is done to determine indoor performance at a variety of transceiver dB levels. It may be worth further research into more powerful transceivers, or at least range extension options such as relay modules — another pair of transceivers which intercept the message and pass it on to the robot when out of single-unit range.

6.2.4 Program Execution

Communication The first test of program execution was to have to GT16A pass back exactly what it received. This was executed without error, as expected. The LabView program also correctly compiled messages for all four motors and camera enable/disable and decoded the received messages, placing them in their addressed locations in the "read array". Thus the read array mirrored the send array. There were occasional pauses in receiving, however, as well as occasional freezes of the entire LabView interface, each pause and freeze lasting only a few seconds.

These random pauses in communications, coupled with the results of the range test for the transceivers, indicate general unreliability in the wireless communication system as it stands. Taken together with the strange phenomena observed around the video transmitter, these results indicate that a much deeper understanding of radio-frequency communications and thereafter a much better communication system is necessary before developing a releasable product. This project used RF communications because certain RF parts were already available in RARL from the project's outset; other communications hardware may yet provide a better solution.

PWM When communication was running, all four PWM channels were updated smoothly and independently, with good square edges despite being at the maximum frequency the H-bridges as rated to. The 5-bit resolution appeared relatively smooth to the eye and was considered more than sufficient for user remote control. It was noted that LabView would not send a fifth byte in the message array unless the total message length was a multiple of two — that is to say, the messages were only fully sent and received when there were eight bytes in the send array, rather than five. The last three elements could not simply be left as zero, however, since they would then be interpreted as having address 000, corresponding to overwriting Motor Speed 0 with 0.

For a more flexible system, messages could be sent on a pre-byte basis, like on the GT16A itself, rather than an array basis. This should allow odd numbers of elements, although it would require an extra counter variable in LabView and calculations to take the place of the array handling.

Switching The camera toggle button on the XBox controller worked as intended, both in the software and the hardware. Bit 0 of Byte 4 was toggled by pressing the button, and this in turn toggled the four pins on the GT16A between 0 and 3V with no noticeable delay between button press and voltage change.

6.3 System

6.3.1 Basic Measurement

The robot was found to be 5.6kg without batteries, which even if they weighed a full kilogram, would take it up to 6.6kg — only 82% of the 8kg on the specifications list. It was interesting to note that the SolidWorks mass estimation was 5.78kg for the complete assembly with batteries.

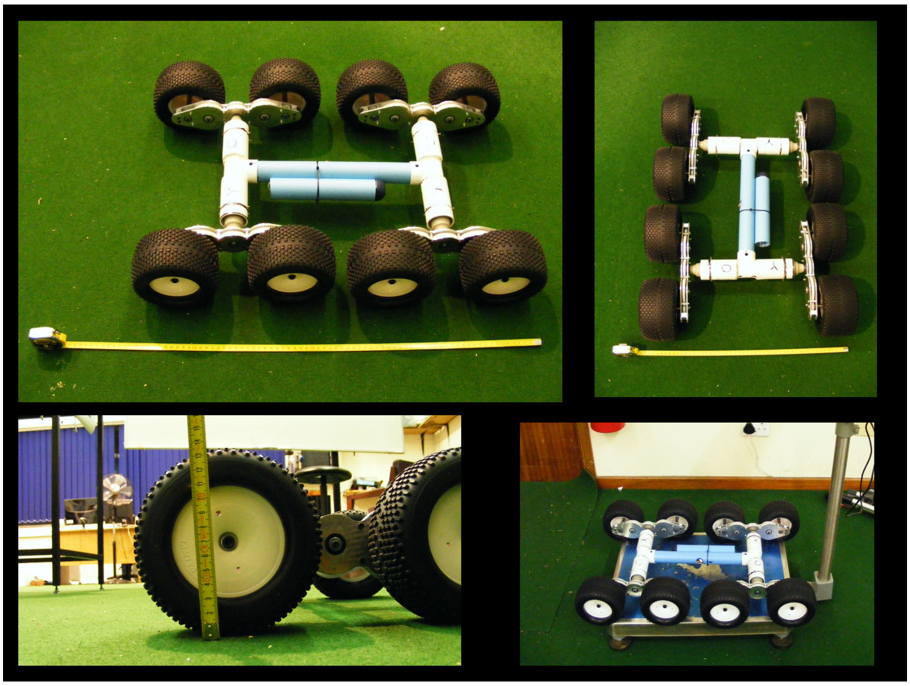


Figure 6.9: Taking basic measurements of the robot.

The fully-assembled robot came in at $720mm \times 510mm \times 140mm$, as shown in Figure 6.9. This agrees with the SolidWorks dimensions to two significant figures — an acceptable margin given the flexibility of the structure. This also automatically means that the robot can fit into a void less than 200mm high, as specified, since it is only 70% of that height.

6.3.2 Drive System

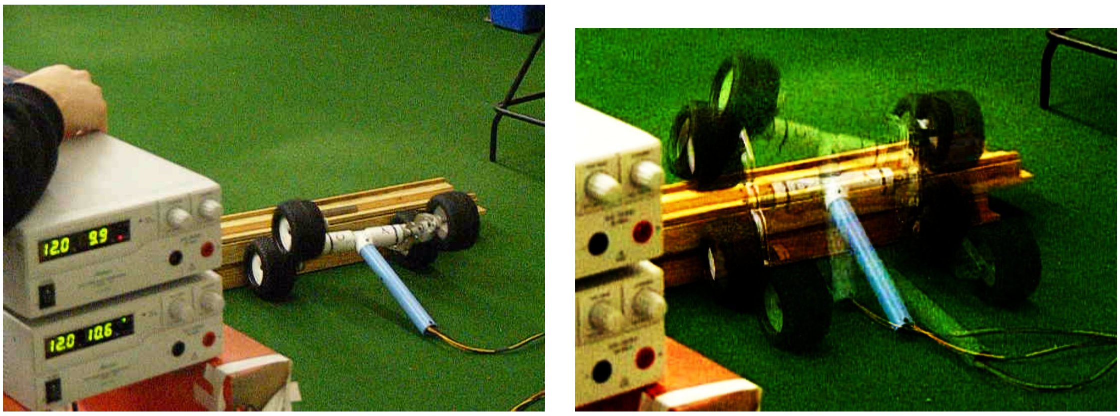
Running Off Power-Supply

The full assembly drove forwards and backwards with moderate power — 5A at 9V (45W) when full speed was reached. Given an obstacle or required to skid steer, however, the power supplies hit their current limits and the motors stalled.

The tail-dragging half-assembly proved more initially promising — with only one motor per power supply, each motor could be supplied with the required currents. In fact, initially it was over-powered and the output shafts from the screwdrivers would slip in the input gears. This was eventually resolved (for testing purposes only) with the ubiquitous cyanoacrylate (super glue) and some Epidermex (an epoxy adhesive). With these adjustments, the robot climbed a stair-sized obstacle, shown in Figure 6.10, with the power supply current-limiting at 10A at 12V for one motor (the other motor only drew around 8A after the initial shock of jamming against the obstacle).

Turning

Attempting to turn the full robot when all four motors were connected to two power supplies resulted in almost no noticable motion; there was a slight jerk in the desired



(a) Power supplies (b) Climbing action

Figure 6.10: One end of the robot climbing a stair-sized pile of planks. Note the upper power supply (right motor) is current limiting in (a).

direction before the motors stalled. Attempting a turn with the tail-dragging half-assembly was only an improvement in that it did not stall. On the contrary — the motors did not stall, instead stripping the left screwdriver's gearbox. In this case the wheels began to lift more obviously, coming slightly off the ground.

Some successful turning was inadvertently accomplished during other drive tests, however, when one motor was turned off slightly ahead of the other, once the robot was already running. This would seem to indicate that the system was on the cusp of turning on the failed occasions, but turning the motors in opposite directions was asking each motor to turn too sharply given too little an arc in which to do so — moving both motors in the same direction lengthened the arc of the turn, making it easier for the LIMs to flip. Put another way, the LIMs flipped more easily when the required ratio of lateral-to-forward movement of each grounded wheel was less — the wheels would roll easily, but would not slip sideways.

This would seem to indicate that the wheels had too much grip for the application, especially on the high-grip carpet of the lab. While moving into a dusty environment like a a collapsed building would probably help to a degree, it is recommended that lower-grip tyres and a wider wheelbase be used in the future if required turning torque is to be reduced. Alternatively, much stronger motors and gearboxes with a higher reduction ratio would also go a long way to helping the robot turn.

Level Speed

Since this project only got so far as to run the robot tethered in a small lab space, the full speed of the robot could not be determined. However, the robot covered its own length of 0.73m in 0.5s, putting its speed in these tests at 1.46m/s, almost 50% greater than specified in the project requirements. Since this was not even the maximum speed, there is definitely scope for reducing the speed of the robot in return for higher torque.

6.4 General

6.4.1 Costing

A full list of expenditures is given in Table 6.3.

A number of these items could have been purchased for less given more time or a larger quantity. Specifically, equivalent wheels can be bought from HobbyKing.com [34] for $11.39 per pair — approximately R500 without shipping and customs tax for all eight wheels. Buying these in large quantities should mitigate the shipping cost.

While R11500 is far in excess of the R4000 that was aimed for, $1130 is still far less than a quarter the cost of even the lowest-cost Throwbot model.

6.4.2 Further Testing

There were a few proposed tests that were not possible in the time for this project, or due to incomplete assembly.

The lack of batteries made it impossible to test endurance. Endurance is a critical aspect of a tele-operated robot for operations like USAR, and this information would have been very useful.

Incline testing would also be helpful for performance characterisation. Because of the LIMs, the robot is unlikely to behave like standard wheeled robots on an incline; there will be a point where rolling gives way to flipping and perhaps a point where flipping causes nett rearward motion, as well as the usual stall and slip points.

Table 6.1: TxRx Range Test Results

Antenna orientation relative to separation		Maximum Communication Distance	
Computer	**Microcontroller**	**(m)**	**Repeatability**
perpendicular	perpendicular	60 - 70	Inconsistent
towards	towards	80	Consistent
perpendicular	towards	82	Consistent
away	towards	67	Consistent

Table 6.3: Project expenditures, as accurately calculated as possible.

	Item	Source	Cost	Qty.	Total	Cumulative Total
1	PVC waste t-junction 40mm	Builders Express	ZAR 8.95	2	**ZAR 17.90**	ZAR 17.90
2	PVC pipe 50mm x 1m	Builders Express	ZAR 19.49	1	**ZAR 19.49**	ZAR 37.39
3	Camera	Light-in-the-Box	ZAR 141.86	4	**ZAR 567.44**	ZAR 604.83
4	Video Tx		ZAR 650.00	1	**ZAR 650.00**	ZAR 1254.83
5	Video Rx		ZAR 650.00	1	**ZAR 650.00**	ZAR 1904.83
6	GT16A Controller		ZAR 34.00	1	**ZAR 34.00**	ZAR 1938.83
7	PCB Manufacture	Beta Layout	ZAR 800.00	0.22	**ZAR 176.00**	ZAR 2114.83
8	PCB Manufacture	Beta Layout	ZAR 500.00	0.25	**ZAR 125.00**	ZAR 2239.83
9	Workshop hours		ZAR 300.00	10.5	**ZAR 3150.00**	ZAR 5389.83
10	Skid Strips	Vulcan Steel	ZAR 13.28	1	**ZAR 13.28**	ZAR 5403.11
11	Electrical components	RS Components	ZAR 150.00	1	**ZAR 150.00**	ZAR 5553.11
12	Spur Gears	RS Components	ZAR 1443.29	1	**ZAR 1443.29**	ZAR 6996.40
13	Tamiya SP-1302 wheel pairs	Hobby Warehouse	ZAR 400.00	4	**ZAR 1600.00**	ZAR 8596.40
14	Aluminium plates		ZAR 79.80	1	**ZAR 79.80**	ZAR 8676.20
15	12A Motor drivers	Pololu	ZAR 991.01	1	**ZAR 991.01**	ZAR 9667.21
16	Composite-to-USB video converter	HiFi Corp	ZAR 600.00	1	**ZAR 600.00**	ZAR 10267.21
17	Batteries	NiteCore	ZAR 95.00	8	**ZAR 760.00**	ZAR 11027.21
18	Cordless Screwdrivers		ZAR 120.00	4	**ZAR 480.00**	**ZAR 11507.21**
					Dollars:	**$1135.37**
					at 12:30am on 30th September 2013	

Chapter 7

Conclusions

In general, each individual component, sub-system, and system worked as expected. The only reason full testing could not be accomplished was because the assembly was quite so arduous due to the imperfections and unexpected intolerances that needed to be accommodated or worked around. It is reasonable to assume, based on the individual system test, that the complete system would have worked, given time, be it with a very heavy current draw for climbing stairs. Turning on the spot was not be possible and is unlikely to become possible with complete integration of all systems.

As regards the specifications, explicitly:

- Cost far exceeded the specified, wished for, R4000. However, it was still more than four times less expensive than the smallest, simplest reconnaissance robot on the market.
- It could not be determined whether the robot could climb stairs, but one half-assembly did overcome a single stair-sized obstacle and the robot can definitely fit into voids less than 200mm high.
- The mass was less than 82% of the specified 8kg, making it very easily man-portable.
- While not comprehensively tested, speed far exceeded the specified 1m/s and should be reduced to improve torque.
- Range tests were inconclusive as regards the specification for indoor range. However, the low performance with full line-of-sight does not bode well for communications through obstructions.
- The camera system exceeded the required single camera by providing stereoscopic footage both front and rear using only one transmitter.
- Control was successfully implemented as specified — a joystick was interpreted by a UI to control the robot wirelessly.

References

[1] FEMA. *Urban Search and Rescue Participants.* http://www.fema.gov/urban-search-rescue-participants, March 2013.

[2] Reuters. http://www.aljazeera.com/photo_galleries/americas/201152510754802696.html, March 2013.

[3] Nikolai Tesla. *Method of and apparatus for controlling mechanism of moving vessels or vehicles.* Patent 613809, United States Patent Office, November 1898.

[4] Andrea Booher/ FEMA News Photo. http://www.commercialappeal.com/photos/galleries/911-news-photos/42974/, March 2013.

[5] FEMA. *Urban Search and Rescue.* http://www.fema.gov/urban-search-rescue, March 2013.

[6] United States Army. http://www.army.mil/article/36534/usace-urban-search-and-rescue-cadre-returns-from-mission-in-haiti, March 2013.

[7] National Geographic Channel. *Collapse - Master of Disaster.* documentary video, 2013.

[8] FEMA. *Approved Task Force Equipment Cache List*, March 2011.

[9] James Wong and Cassandra Robinson. *Urban Search and Rescue Technology Needs: Identification of Needs*, November 2004.

[10] Paul McLeary. Little robots, big chance to shine. *Defense News International*, July 2012. http://www.defensenews.com/article/20120715/DEFFEAT01/307150005/Little-Robots-Big-Chance-Shine.

[11] Boston Dynamics. *RHex - Devours Rough Terrain.* http://www.bostondynamics.com/robot_rhex.html, March 2013.

[12] Recon Robotics. http://www.reconrobotics.com/products/LE_product_photos.cfm, March 2013.

[13] Recon Robotics. *Recon Scout Throwbot* LE *brochure*, 2011.

[14] iRobot. http://www.unmanned.co.uk/autonomous-unmanned-vehicles/ugv-data-specifications-fact-sheets/irobot-110-firstlook-unmanned-ground-vehicle-
July 2011.

[15] iRobot. *iRobot 110 FirstLookÂ® specifications brochure*, 2012.

[16] ARA. http://www.ara.com/robotics/PDFs/NTOA%20Assessment.pdf, May 2013.

[17] Applied Research Associates and Inc. *ARA LRV Brochure 1-4-06*, 2008.

[18] Eugene Dreyer. *50 Shades of Dissertation*. Masters dissertation, The University of Cape Town, November 2012.

[19] Sirris. *TR — Environmental reconnaissance robot*. http://www.sirris.be/defaultPage.aspx?id=11790&LangType=2067, April 2013.

[20] EUREKA. *EUREKA Project Form SPYROB*. http://www.sirris.be/uploadedFiles/SIRRIS/Members_Network/EUREKA%20FORM%20SPYROB.pdf, April 2013.

[21] Nathan Hurst. *These $10 Robots Will Change Robotics Education*. http://www.wired.com/design/2012/09/afron-winners/, September 2012.

[22] Matt Cowan. *Stereoscopic 3D - How it Works*. http://reald.com/Content/Files/Stereoscopic3DHowitWorks.pdf, September 2013.

[23] Tim Seifert. *Colour video signal generation and connections*. http://www.cameratim.com/info/colour-video, September 2013.

[24] Captain3D. *3D (stereoscopic) FPV flying and movies*. http://www.rcgroups.com/forums/showthread.php?t=1644240, May 2012.

[25] Battery University. *What's the Best Battery*. http://batteryuniversity.com/learn/article/whats_the_best_battery, May 2013.

[26] epec. *Battery Cell Types*. http://www.epectec.com/batteries/cell/, May 2013.

[27] Pool Supply Sales. *Zodiac Baracuda MX8 Parts*. http://www.poolsupplysales.com/parts/pool-cleaner/zodiac-baracuda/mx8.html, May 2013.

[28] Zodiac. http://www.zodiac.co.za/catalog/automatic-cleaners-pool-suction-mx-8#.UY5vDaJTDj4, May 2013.

[29] Uluc Saranli, Martin Buehler, and Daniel E. Koditschek. Rhex: A simple and highly mobile hexapod robot. *The International Journal of Robotics Research*, 20(7):616–631, 2001.

[30] Lockheed. http://www.unusuallocomotion.com/pages/locomotion/unusual-locomotion-wheels-tracks-walking-reptation.html, May 2013.

[31] Lockheed. http://www.amphibiousvehicle.net/amphi/Lb_Lz.html, May 2013.

[32] Mean Well. *5W DC-DC Regulated Single Output Converter Datasheet*. http://www.meanwell.com/search/scw05/SCW05-spec.pdf, September 2013.

[33] Freescale Semiconductor. *MC9S08GT16A/MC9S08GT8A Data Sheet*, July 2006.

[34] HobbyKing. *1/8 Buggy 3.4 (offset) Wheel/Tire 17mm Hex (2pcs/bag)*. http://www.hobbyking.com/hobbyking/store/__25070__1_8_Buggy_3_4_offset_Wheel_Tire_17mm_Hex_2pcs_bag_.html, September 2013.

Appendix A

Bi-Star Mathematics

The base point for understanding how a bi-star drive train works is understanding its kinematics (how each part is constrained to move with respect to the others). The actual kinetics of the systems are required to optimise torques. A simple kinetic model is described in Section 4.1.1 in the main body of this report.

A.0.3 Kinematics

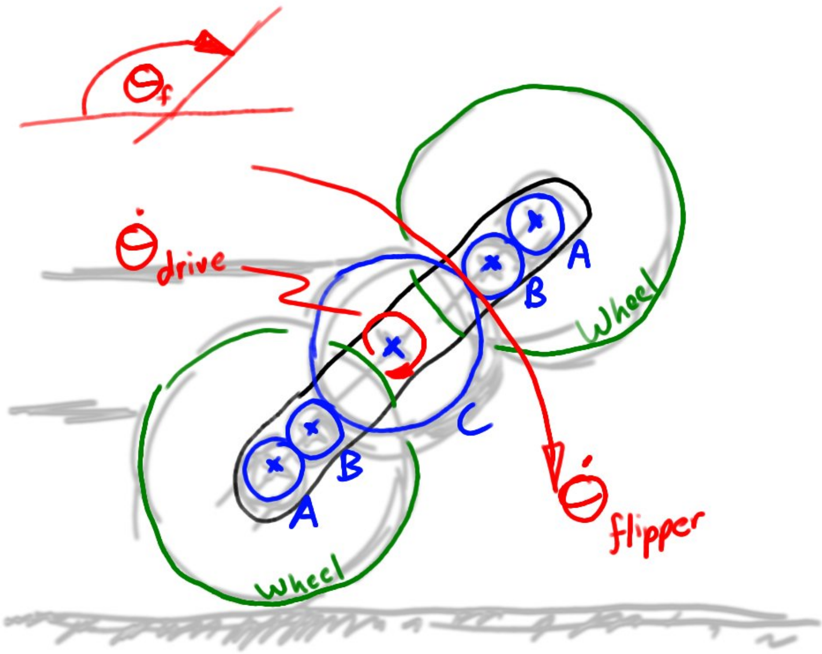


Figure A.1: A bi-star module schematic

The bi-star drive train can be analysed like an epicyclic[1] gear train. For the purposes of this analysis, a tabular approach will be used, but an instantaneous-centres approach will give the same results.

[1] planetary

Table A.1: Bi-Star Kinematics

	A	B	C	Carrier Arm
Motion Relative to Frame	$\dot{\theta}_{wheel}$	$-\frac{N_A}{N_B}\dot{\theta}_{wheel}$	$\frac{N_A}{N_C}\dot{\theta}_{wheel}$	$\dot{\theta}_{flipper}$
Motion Relative to Carrier Arm	$\dot{\theta}_{flipper}$	$\dot{\theta}_{flipper}$	$\dot{\theta}_{flipper}$	0
Absolute Motion	$\dot{\theta}_{flipper} + \dot{\theta}_{wheel}$	$\dot{\theta}_{flipper} - \frac{N_A}{N_B}\dot{\theta}_{wheel}$	$\dot{\theta}_{flipper} + \frac{N_A}{N_C}\dot{\theta}_{wheel}$	$\dot{\theta}_{flipper}$

From Table A.1, the driving speed (absolute $\dot{\theta}$ of C) can be expressed as:

$$\dot{\theta}_{drive} = \dot{\theta}_{flipper} + \frac{N_A}{N_C}\dot{\theta}_{wheel} \tag{A.1}$$

If CA is not rotating, $\dot{\theta}_{flipper} = 0$, so:

$$\dot{\theta}_{drive} = \frac{N_A}{N_C}\dot{\theta}_{wheel} \tag{A.2}$$

$$\dot{\theta}_{wheel} = \frac{N_C}{N_A}\dot{\theta}_{drive} \tag{A.3}$$

If wheels are not rotating, $\dot{\theta}_{flipper} + \dot{\theta}_{wheel} = 0$

$$\dot{\theta}_{drive} = \dot{\theta}_{flipper}\left(1 - \frac{N_A}{N_C}\right) \tag{A.4}$$

$$\dot{\theta}_{flipper} = \frac{\dot{\theta}_{drive}}{\left(1 - \frac{N_A}{N_C}\right)} \tag{A.5}$$

So, if the wheels get stuck, motion of the flipper is dependent on the gear ratio:

- If $\frac{N_A}{N_C} > 1$, the flipper rotates opposite to the drive, flipping backwards — good for smooth terrain following.
- If $\frac{N_A}{N_C} = 1$, the flipper cannot be driven by the drive. Consider the flipper rotating of its own accord — at a 1:1 gearing, neither the wheels nor drive rotate, so no work can be done by the drive in that motion.
- If $\frac{N_A}{N_C} > 1$, the flipper rotates with the drive, flipping forwards — better for hooking up onto ledges on obstacles.

Appendix B

Risk Assessments

Appendix C

LabView Code

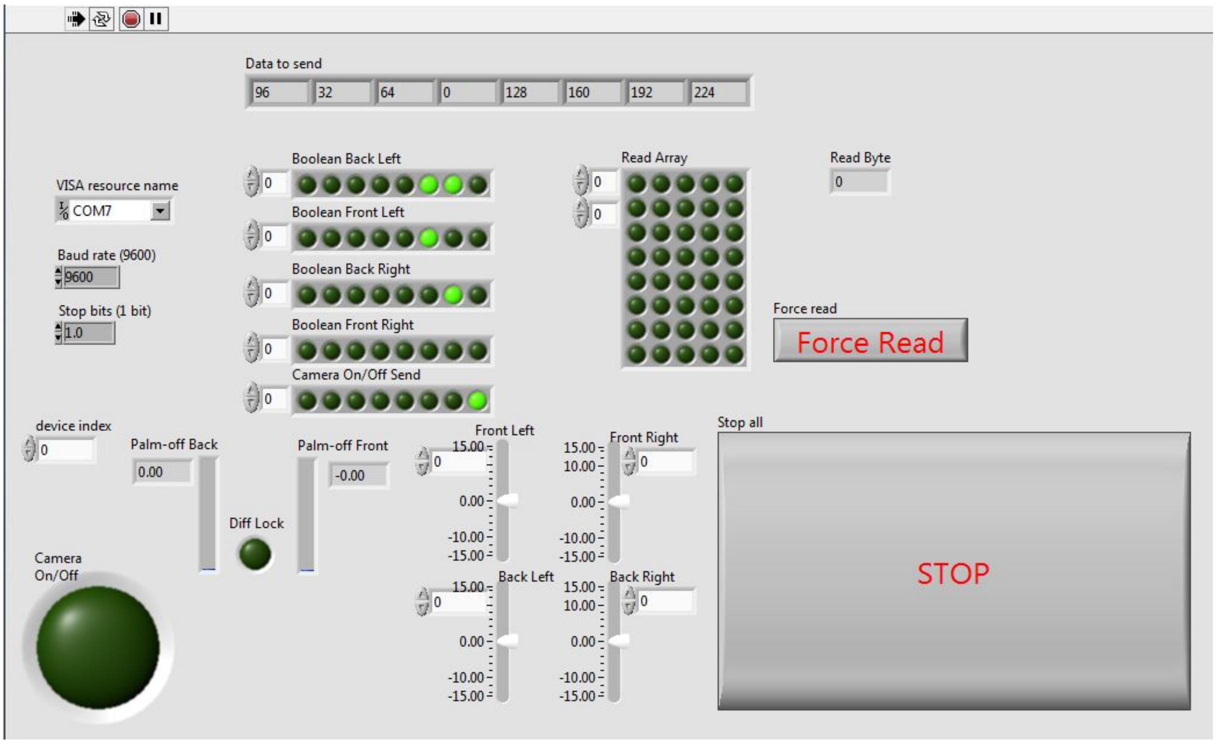


Figure C.1: The LabView graphical user interface (GUI)

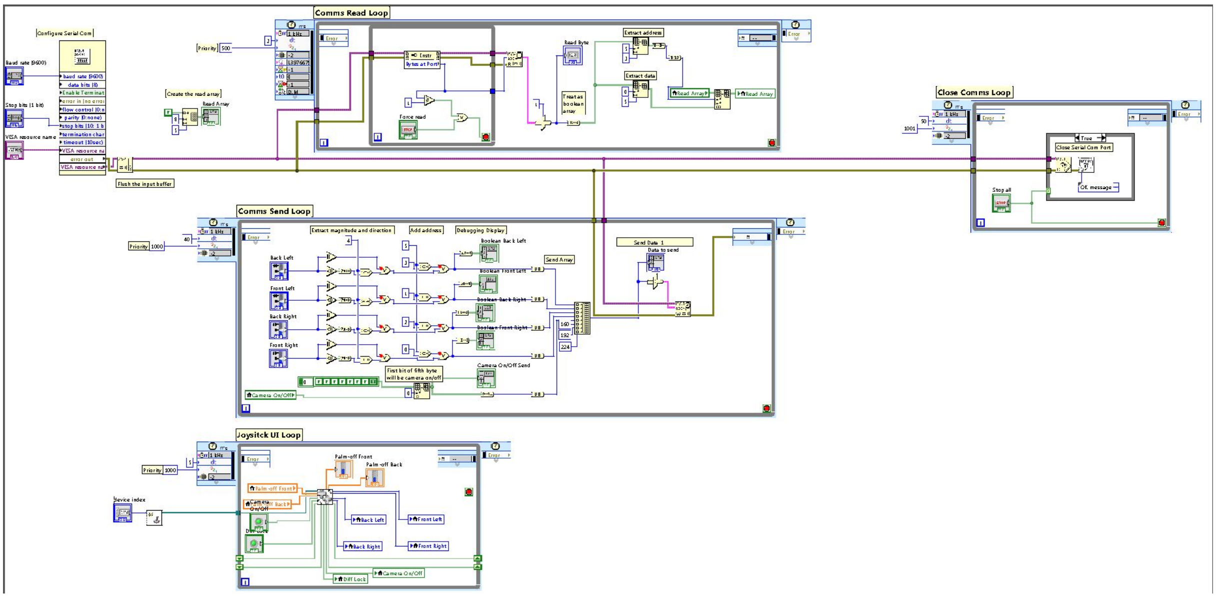


Figure C.2: The LabView block diagram ("code")

Appendix D

C Code

```
        /*
----------------
################
Project Testing

  Driving All
    Motors
      AND
Getting Current
    Feedback
################
 Matthew Wilson
   WLSMAT004
################
----------------
        */

//********************************************************//
// Takes in RS232 data, sorts it into a message array,    //
// sets motor PWM and switches on and off video power     //
// accordingly, reads current sensing levels via ADC, and //
// send those reading back to the user.                   //
//********************************************************//
//********************************************************//
//                                                        //
//       Pin-outs:                                        //
//                                                        //
//       Camera Enable/Disable                            //
//         PTC0-3                                         //
//       PWM                                              //
//         PTD0,1,3,4                                     //
//       Current sensing (ADC)                            //
//         PTB0-3                                         //
//       Direction Enables                                //
//         PTA 0,1                                        //
//         PTA 2,3                                        //
//         PTA 4,5                                        //
//         PTA 6,7                                        //
//                                                        //
//********************************************************//

#include <hidef.h>      /* for EnableInterrupts macro */
#include "derivative.h" // include peripheral declarations

  //initialise interupts and functions HERE
void int_sci(void);   //declare the Serial Comms interrupt function.
void sort_address(unsigned char dat);   //declare the address-sorting function
void adc_irq(void);        //declare the ADC interupt handler

  //any LCD extern stuff HERE
```

```
  //initialise variables HERE
unsigned char Data =  0; //place to store received data from SCI.
unsigned int  modu =  1; //save the modu value.
unsigned char CScycle = 0; //cycle number to select which ADC channel to read.

unsigned char RecArray [8] =
  {
    0,0,0,0,0,0,0,0
  }; //Array for received bytes once address has been processed.

unsigned int Curve [16] =
  {
  8,  70, 132,  194,  255,  317,  379,  441,  503,  565,  627,  689,  750,  812,  874,
    936 //  Linear
//  8,  12, 24, 45, 74, 111,  156,  210,  272,  342,  420,  507,  602,  705,  816,  936
    //  Squared
//  8,  248,  347,  423,  487,  544,  595,  642,  686,  727,  766,  803,  838,  872,
    905,  936 //  Square root
//  8,  8,  10, 15, 26, 42, 067,  102,  149,  208,  283,  374,  483,  612,  762,  936 //
      Cubed
//  8,  384,  482,  551,  605,  651,  692,  728,  761,  791,  819,  845,  869,  893,
    915,  936 //  Cube root
  };


//************************************************************

void main(void)

{

  SOPT=53;                  // Kill the dog.
  ICGC2=0x70;               //sets MFD divider
  ICGC1=0x38;               //32kHz -> 18.874368MHz bus rate
  while (ICGS1_LOCK==0);    //loop until FLL locks


  PTADD= 0xFF;     // Make PTA an output (motor direction selection)
  PTAD = 0xAA;     // Set all motors to clockwise.
  PTBDD &= ~0x0F; // Set first 3 PTB pins to inputs (current sensing ADC)
  PTCDD = 0x0F;    // Enable output on pins 0-3 of port C
  PTCD  = 0x0F;    // Pull all those pins high (enable cameras by default)
  PTDDD = 0x0F;    // Enable PWM pins


//**********************//
//** OTHER SETUPs HERE **//
//**********************//

/*********************************************************
************* ADC SETUP *******************************
*********************************************************/

    //We now need to set the ADC up. (standard setup)
    //Options used are as follows:
    //ATD powered up
    //DJM isn't used in 8 bit mode so any value, arbitrarily low
    //RES 8 high for 8 bit
    //SGN low because we want an unsigned output
    //Prescaler set to 4 for a division ratio of 10
ATDC=0xA4;
  //Set which pins are required as ADC inputs, rest of the port as IO
  // This enables the pull-up resistors for those pins.
ATDPE=0x0F;

      // !!! NOTE: if the LCD is enabled, can only use 1st 2 pins. !!!

  //Start the converter running in single shot mode, with interrupts
ATDSC=0x40;   //pin 1
//ATDSC=0x41;   //pin 2
//ATDSC=0x42;   //pin 3

/*********************************************************
```

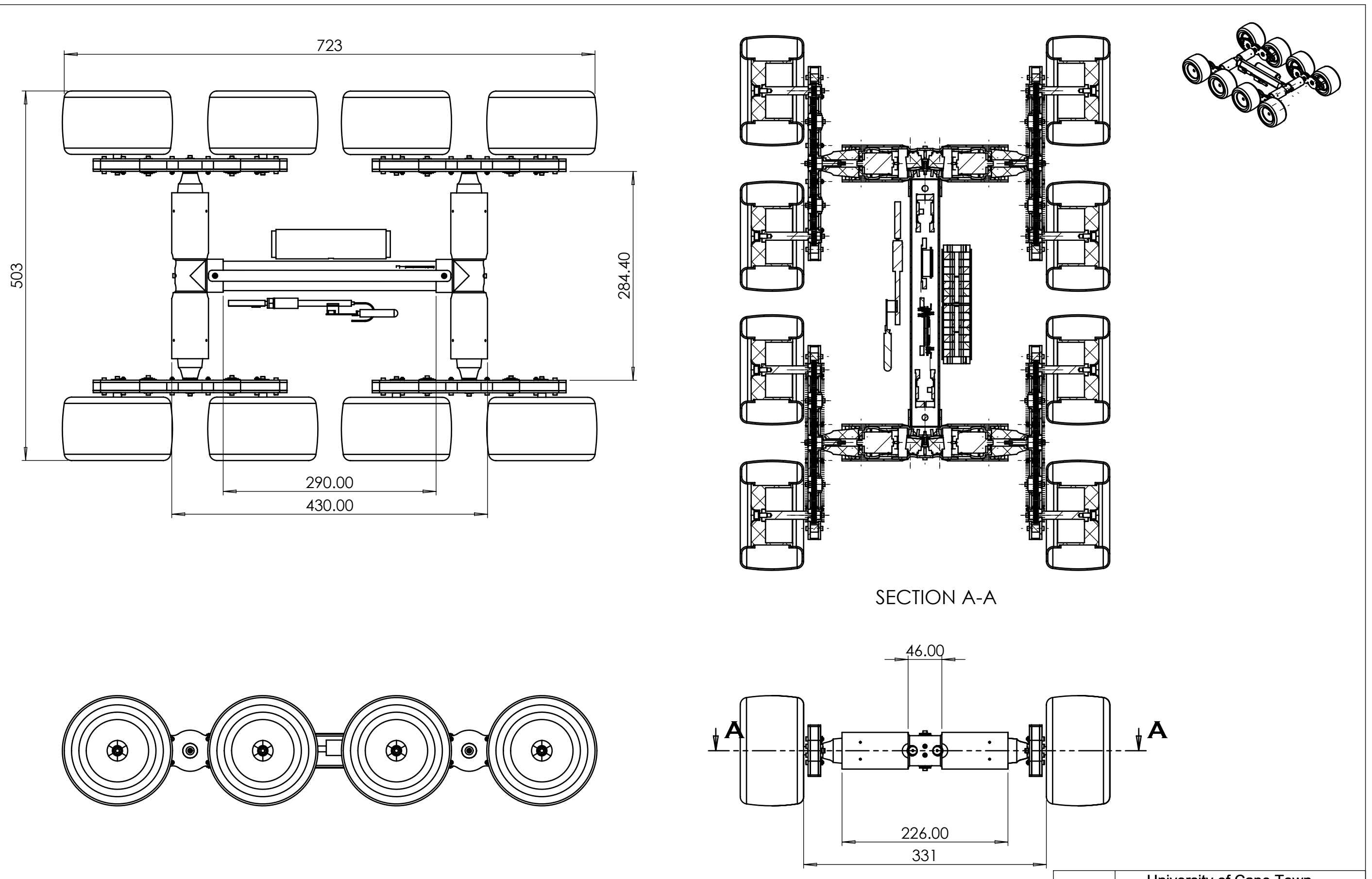
```
************** PWM SETUP ********************************
*******************************************************/

  //Produce a 1kHz 25% duty cycle out of the PTD4 (Timer 2 channel 1) pin

  //Set the modulo registers up to produce a 20kHz period
    // We now need to set the Timer up for 20kHz, interupt mode, BusClk driven
    // Register setup is as follows:
    // TOIE=0 interrupts disabled
    // CPWMS=0 for up counting only
    // CLKSB=0 Bus Clock driven
    // CLKSA=1 Bus Clock driven
    // PRE2=0 \
    // PRE1=0  }- Prescaler=1
    // PRE0=0 /
    // Modulo value=(18.874368*10^6)/(20000*prescaler) = 943.7184
modu   =944;     //no prescaler needed
TPM1MOD=modu;
TPM2MOD=modu;

  //Set the channel value registers to almost 0% (but NOT fully 0%)
TPM1C0V=0x0008;
TPM1C1V=0x0008;
TPM2C0V=0x0008;
TPM2C1V=0x0008;

//Setup the PWM:
// The channel pin is used as an output.
//7     6     5     4     3     2     1     0
//flag  ints? 1     0     Edge-Level    0     0
//                        0     0 - no PWM
//                        0     1 - output inverted
//                        1     0 - output non-inverted
//                        1     1 - output inverted
  //Set the timer channel status and control register up as follows:
  //Interrupts disabled
  //mode bits='10' for edge aligned PWM
  //Edge/level bits='10' for high-true outputs to give non-inverted PWM
TPM1C0SC=0x28;
TPM1C1SC=0x28;
TPM2C0SC=0x28;
TPM2C1SC=0x28;

  //Set the timer up for no interrupts, bus clock source, prescaler for 1kHz
TPM1SC=0x08;
TPM2SC=0x08;

/*******************************************************
************** SCI SETUP ********************************
*******************************************************/

  //Initialize clock generator and FLL
ICGC2=0x70;             //sets MFD divider
ICGC1=0x38;             //32kHz -> 18.874368MHz bus rate
while (ICGS1_LOCK==0);  //loop until FLL locks
SCI1BDH = 0x00;         //Serial Baud rate divider
SCI1BDL = 123;          //Serial Baud rate divider baud rate 9600
SCI1C1 = 0x00;          //Serial Register 1 Setup


//Setting up the SCI1C2 Register:
/*
  The channel register is as follows.
  7     6     5     4     3     2     1     0
  TIE   TCIE  RIE   ILIE  TE    RE    RWU   SBK
  |     |     |     |     |     |     0     0
  Trans interrupt enable  |     |
        |     |     |     |     |
        Trans COMPLETE interrupt|enable
              |     |     |     |
              Rec interrupt enable
                    |     |     |
                    Idle line interrupt enable
```

```
                          |     |
                          Trans enable
                                |
                                Rec enable
*/
// For example ---
  //We now need to set up our basic settings as follows
  //TIE = 0 Transmit interrupts disabled
  //TCIE = 0 Transmission complete interrupts disabled
  //RIE = 1 Receive interrupts enabled
  //ILIE = 0 Line idle interrupts disabled
  //TE = 1 Transmitter enabled
  //RE = 1 Receiver enabled
  //RWU = 0 Receiver wakeup control disabled
  //SBK = 0 Send break disabled
SCI1C2 = 0x2C;          //Serial Register 2 Setup


  //DON'T FORGET THIS NEXT BIT:
  asm ("cli");  //clear interupts
  for(;;)       //main loop
    {
      PTCD = 0x0F * RecArray[4];  //Enable/disable cameras


      //PWM set
        //Motor 0
      TPM1C0V=Curve[RecArray[0] & 0x0F];  //Call power curve value
      PTAD_PTAD0= (RecArray[0] >> 4) & 1;    // Set direction 0
      PTAD_PTAD1 = ~PTAD_PTAD0;           //  Set direction 1

        //Motor 1
      TPM1C1V=Curve[RecArray[1] & 0x0F];  //Call power curve value
      PTAD_PTAD2= (RecArray[1] >> 4) & 1;    // Set direction 0
      PTAD_PTAD3 = ~PTAD_PTAD2;           //  Set direction 1

        //Motor 2
      TPM2C0V=Curve[RecArray[2] & 0x0F];  //Call power curve value
      PTAD_PTAD4= (RecArray[2] >> 4) & 1;    // Set direction 0
      PTAD_PTAD5 = ~PTAD_PTAD4;           //  Set direction 1

        //Motor 3
      TPM2C1V=Curve[RecArray[3] & 0x0F];  //Call power curve value
      PTAD_PTAD6= (RecArray[3] >> 4) & 1;    // Set direction 0
      PTAD_PTAD7 = ~PTAD_PTAD6;           //  Set direction 1
    };
}

//===========================================================================
//===========================================================================

  //interupts HERE
interrupt 17
void int_sci(void)
  {
    unsigned char ackn;

    ackn = SCI1S1; // Acknowledge the interrupt
    Data = SCI1D; // Read in data
    sort_address(Data); // Put that data into the right place
//    SCI1D=Data; // Send data
  }


//===========================================================================

void sort_address(unsigned char dat)
  {
    unsigned char addr = 0;

    addr = ((dat >> 5) & 7);  //Pull out the address
    RecArray[addr] = dat & 0x1F;  //Put the value into the "received" array
                                  // at the corresponding address.
```

```
  }
//==========================================================================

interrupt 23              // ADC INTERUPT
void adc_irq(void)
{
  unsigned char tempCS = 0; //variable to process to get current value

  //ADC reading:
// read from ATDRH (8 bit)
//        or ATDR (10 bit)

tempCS = (ATDRH / 8) + (CScycle * 32);  //Read ATD, shrink to 5 bits,
                                        // and append address.

SCI1D = tempCS;  //send


//** Cycle CScycle **//
  if (CScycle < 3)
    {
      CScycle++;
    }else
      {
        CScycle = 0;
      };

ATDSC = 0x40 + CScycle; //reset the ADC to check the next channel
}
```

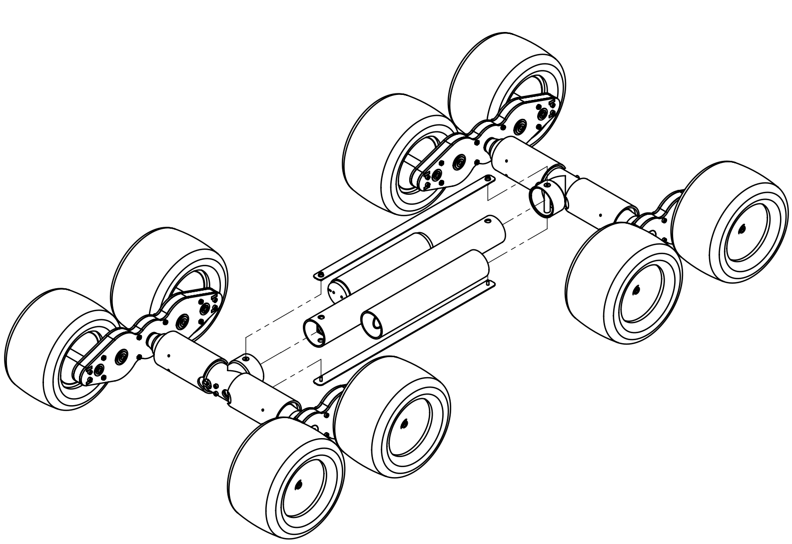
Appendix E

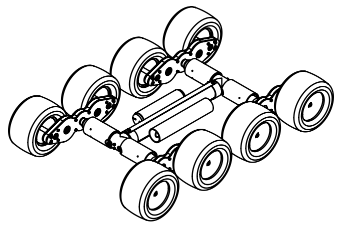
Drawing Pack



A3 Landscape	University of Cape Town Department of Mechanical Engineering			
	Title: Main Assembly			
Assembly Drawing	Scale: 1:5	Date: 2013/10/01	Sheet: 1	of 28
	Drawn By: Matthew Wilson, WLSMAT004		Drawing Number: 4110-49-01	

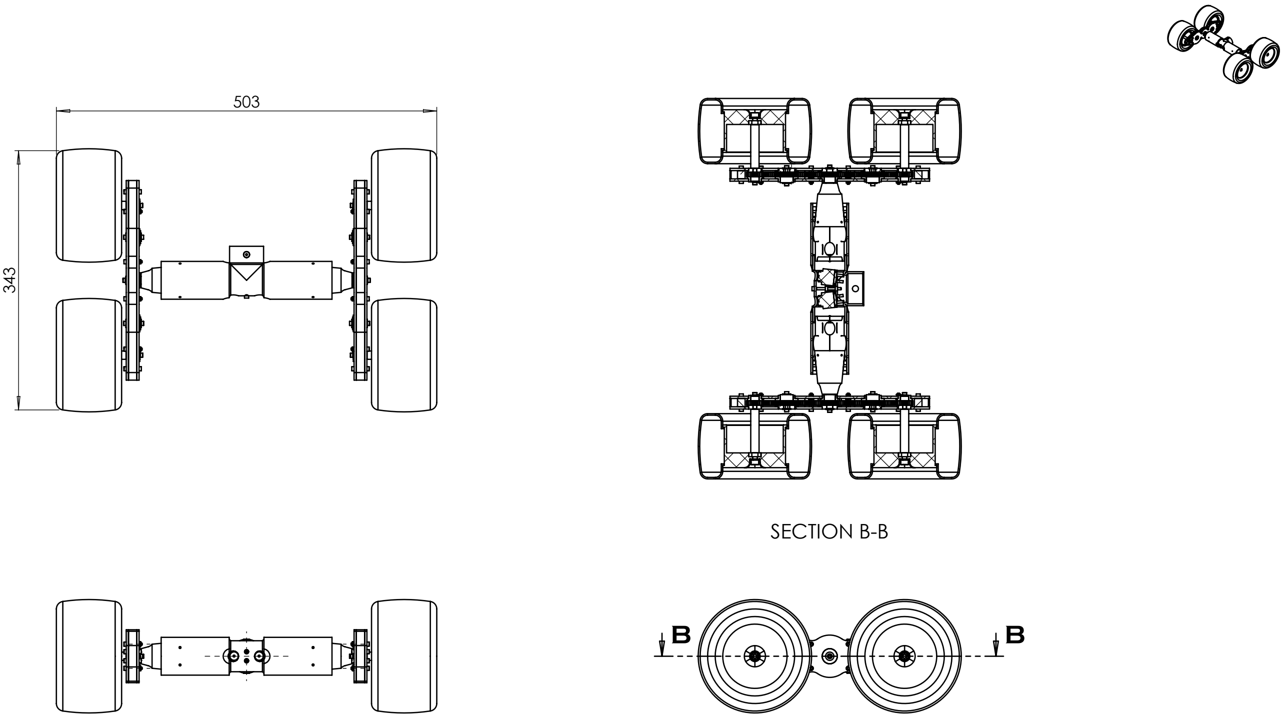
SolidWorks Student Edition.
For Academic Use Only.



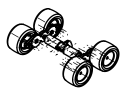


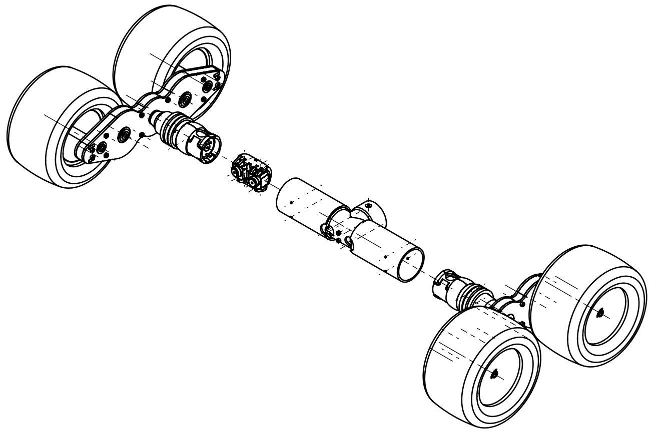
SolidWorks Student Edition.
For Academic Use Only.

A3 Landscape	University of Cape Town Department of Mechanical Engineering			
	Title: Main Assembly Exploded View			
Assembly Drawing	Scale: 1:5	Date: 2013/10/01	Sheet: 2	of 28
	Drawn By: Matthew Wilson, WLSMAT004		Drawing Number: 4110-49-02	



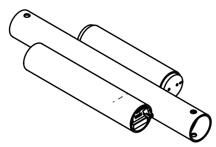
A3 Landscape	University of Cape Town Department of Mechanical Engineering		
	Title: End Assembly		
Assembly Drawing	Scale: 1:5	Date: 2013/10/01	Sheet: 2 of 28
	Drawn By: Matthew Wilson, WLSMAT004		Drawing Number: 4110-49-02



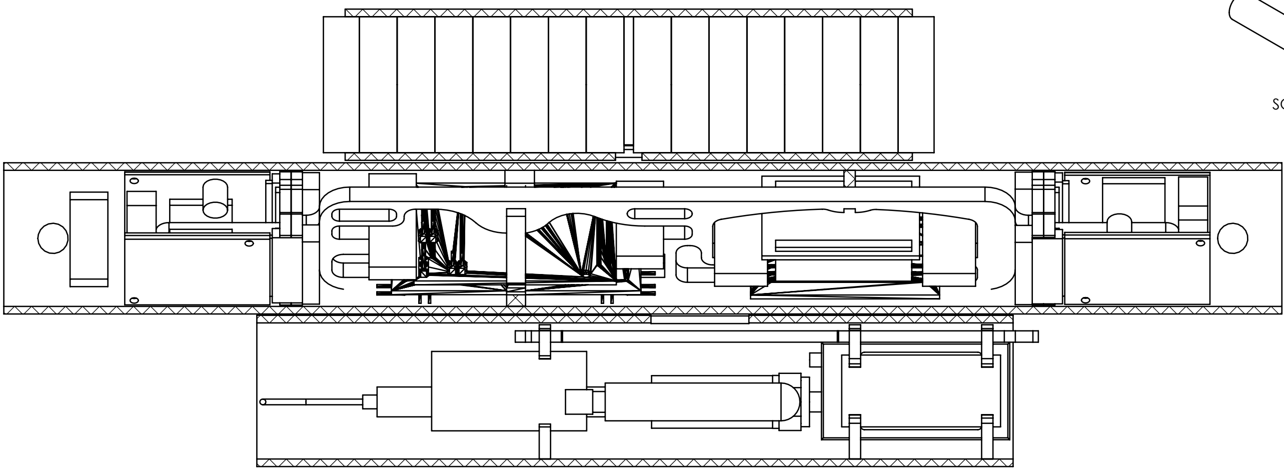


SolidWorks Student Edition.
For Academic Use Only.

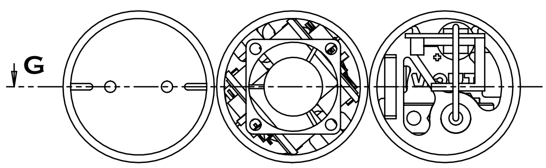
A3 Landscape	University of Cape Town Department of Mechanical Engineering			
	Title: End Assembly Exploded View			
Assembly Drawing	Scale: 1:5	Date: 2013/10/01	Sheet: 4	of 28
	Drawn By: Matthew Wilson, WLSMAT004		Drawing Number: 4110-49-04	



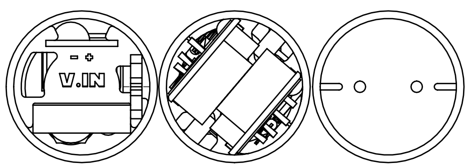
SCALE 1:5



SECTION G-G

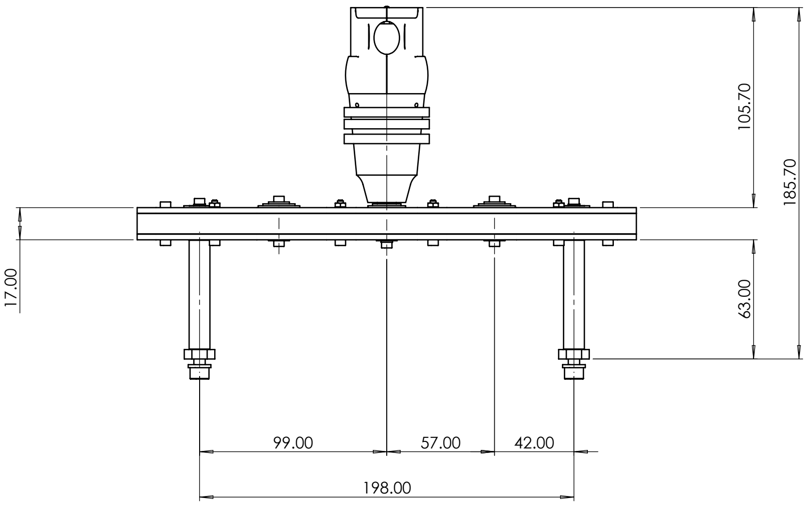


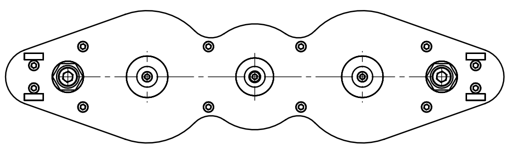
FRONT VIEW

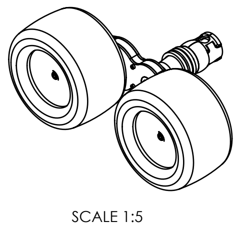


BACK VIEW

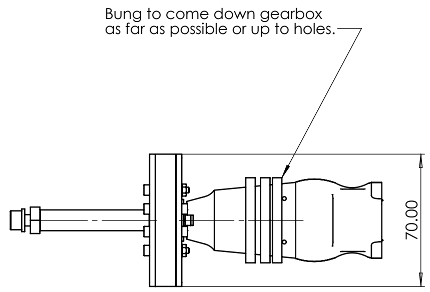
A3 Landscape	University of Cape Town Department of Mechanical Engineering			
	Title: Spine Assembly			
Assembly Drawing	Scale: 1:1	Date: 2013/10/01	Sheet: 3	of 28
	Drawn By: Matthew Wilson, WLSMAT004		Drawing Number: 4110-49-03	



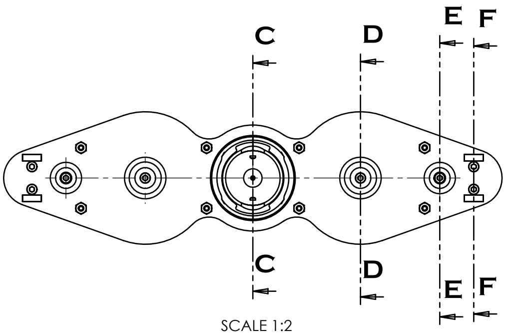




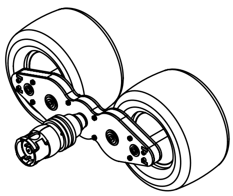
SCALE 1:5



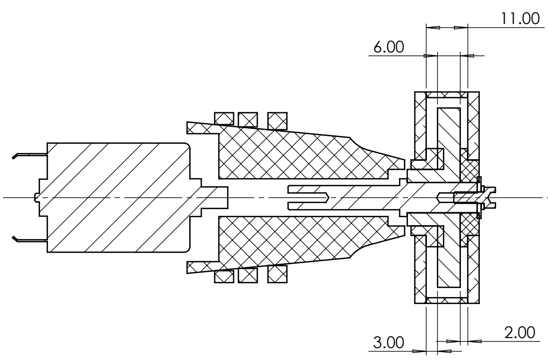
A3 Landscape	University of Cape Town Department of Mechanical Engineering			
	Title: LIM Assembly			
Assembly Drawing	Scale: 1:2	Date: 2013/10/01	Sheet: 3	of 28
	Drawn By: Matthew Wilson, WLSMAT004		Drawing Number: 4110-49-03	



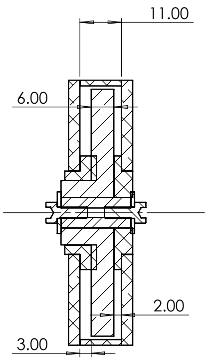
SCALE 1:2



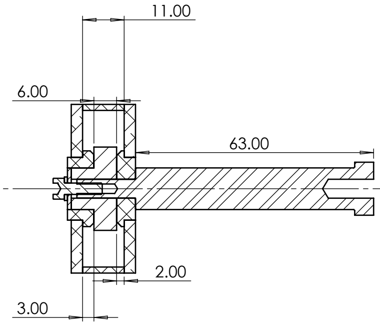
SCALE 1:5



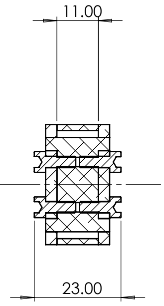
SECTION C-C



SECTION D-D

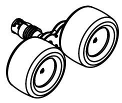


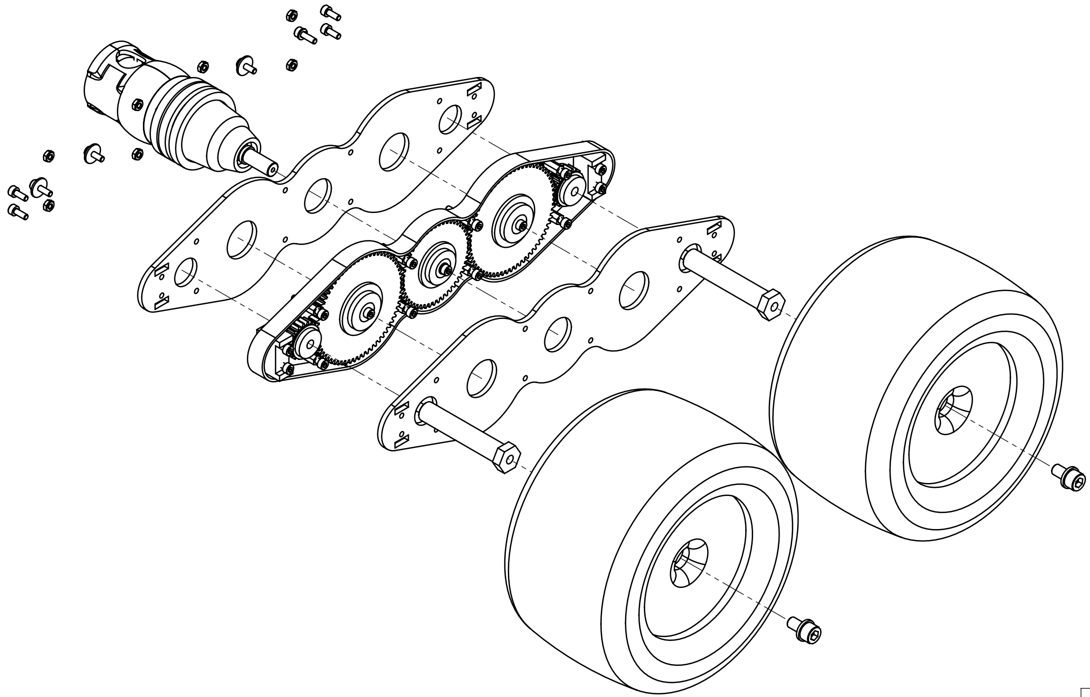
SECTION E-E



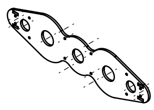
SECTION F-F

A3 Landscape	University of Cape Town Department of Mechanical Engineering			
	Title: LIM Assembly (Sections)			
Assembly Drawing	Scale: 1:1	Date: 2013/10/01	Sheet: 4	of 28
	Drawn By: Matthew Wilson, WLSMAT004		Drawing Number: 4110-49-04	

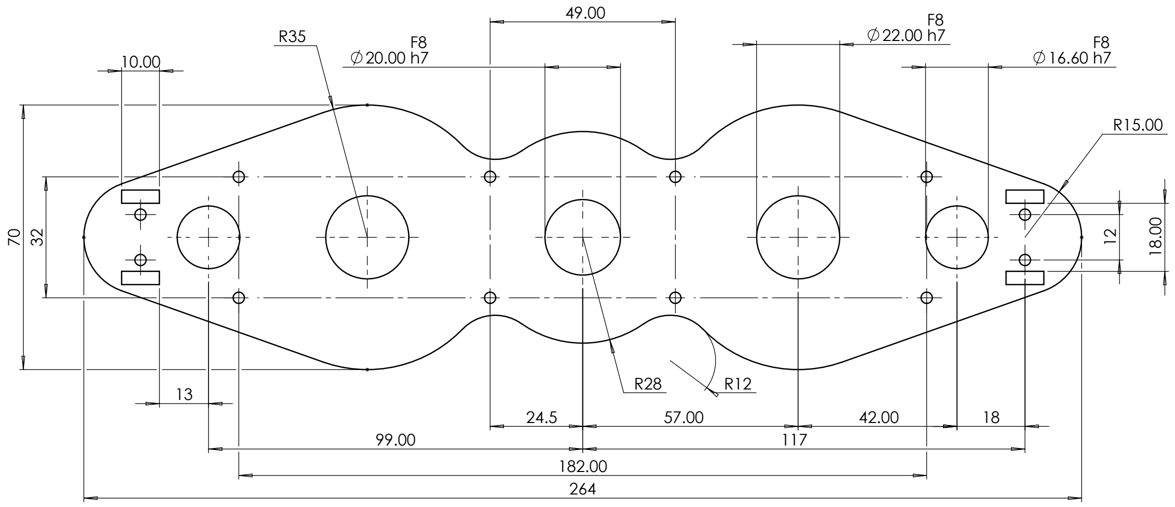




A3 Landscape	University of Cape Town Department of Mechanical Engineering			
	Title: <Lim Assembly Exploded View>			
Assembly Drawing	Scale: 1:2	Date: 2013/10/01	Sheet: 8	of 28
	Drawn By: Matthew Wilson, WLSMAT004		Drawing Number: 4110-49-8	

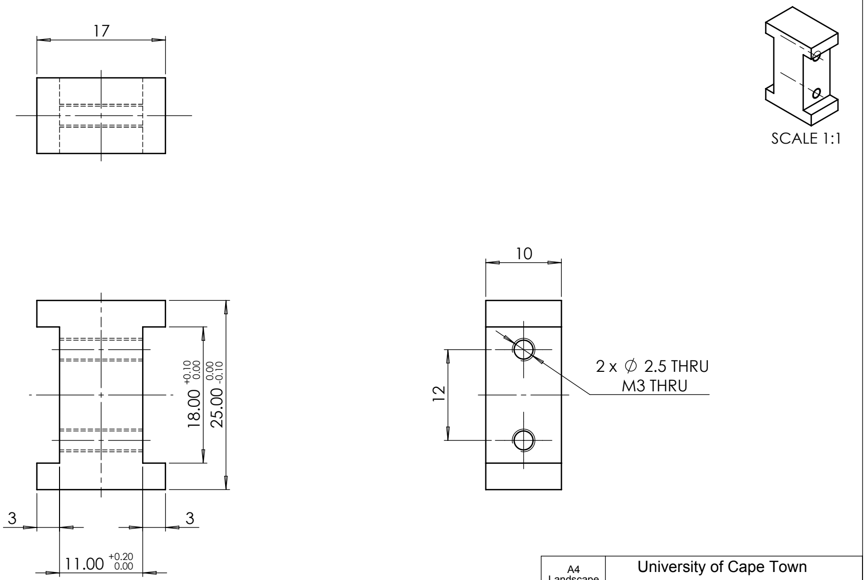


SCALE !:5

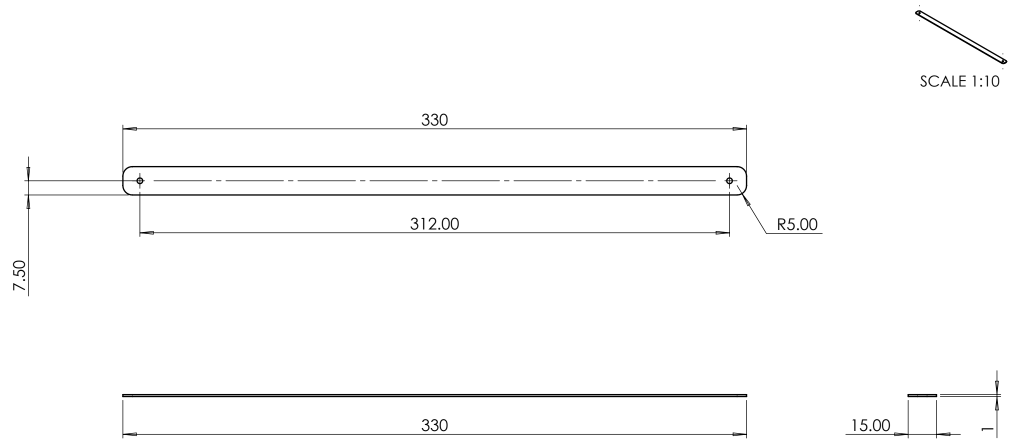


Profile to be laser cut
from 3mm aluminium sheet

A3 Landscape	University of Cape Town Department of Mechanical Engineering				
	Title: Flipper Wall				
Quantity: 8	Part Finish: As cut	Date: 2013/10/01	Scale: 1:1	Sheet: 5	of 28
Material: Aluminium	Drawn By: Matthew Wilson WLSMAT004		Drawing Number: 4110-49-05		

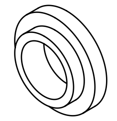


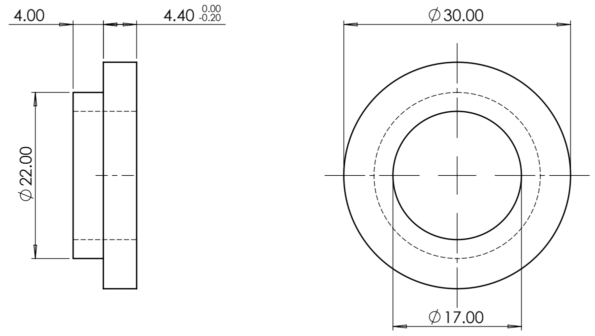
A4 Landscape	University of Cape Town Department of Mechanical Engineering				
	Title: Brace Block				
Quantity: 8	Part Finish: Fine	Date: 2013/10/01	Scale: 2:1	Sheet 6	of 28
Material: Aluminium	Drawn By: Matthew Wilson WLSMAT004		Drawing Number: 4110-49-06		



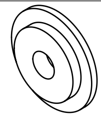
Note:
To be cut from 1mm
mild steel sheet.

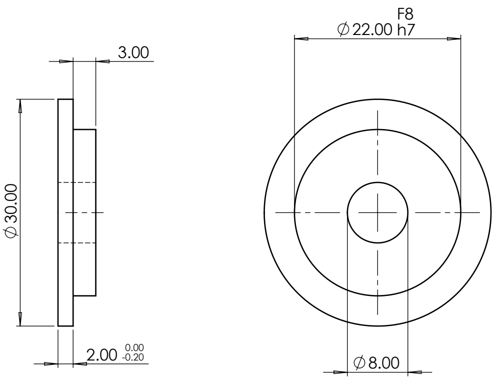
A4 Landscape	University of Cape Town Department of Mechanical Engineering				
	Title: Skid Strip				
Quantity: 2	Part Finish: As cut	Date: 2013/10/01	Scale: 1:2	Sheet 7	of 28
Material: Mild steel	Drawn By: Matthew Wilson WLSMAT004		Drawing Number: 4110-49-07		



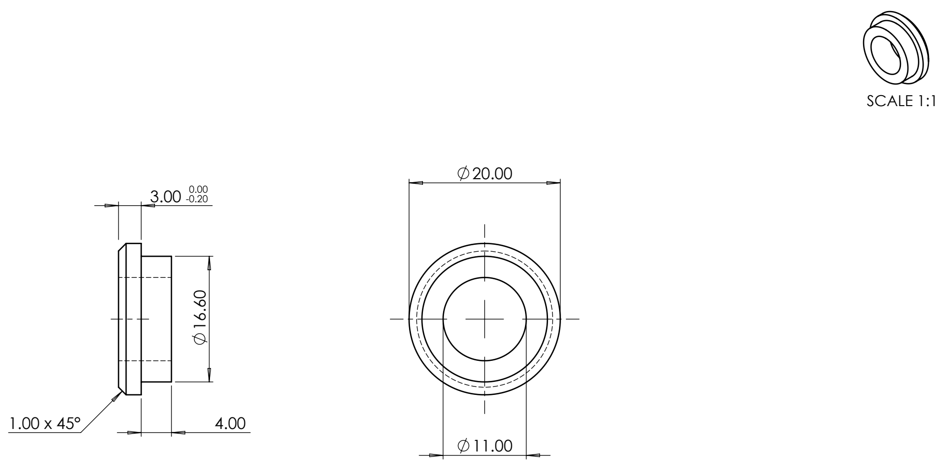


A4 Landscape	University of Cape Town Department of Mechanical Engineering				
	Title: Idler Bushing				
Quantity:	Part Finish:	Date:	Scale:	Sheet	of
8	Fine	2013/10/01	2:1	8	28
Material: Vesconite	Drawn By: Matthew Wilson WLSMAT004		Drawing Number: 4110-49-08		

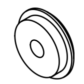


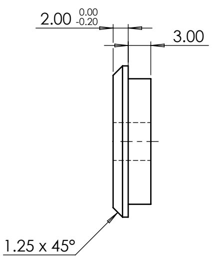


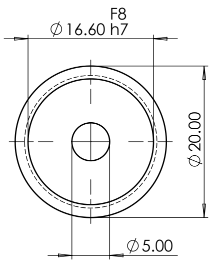
A4 Landscape	University of Cape Town Department of Mechanical Engineering					
	Title: Idler Outer Bushing					
Quantity:	Part Finish:	Date:	Scale:	Sheet	of	
8	Fine	2013/10/01	2:1	9	28	
Material: Vesconite	Drawn By: Matthew Wilson WLSMAT004		Drawing Number: 4110-49-09			



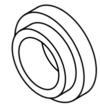
A4 Landscape	University of Cape Town Department of Mechanical Engineering				
	Title: Pinion Bushing				
Quantity: 8	Part Finish: Fine	Date: 2013/10/01	Scale: 2:1	Sheet 10	of 28
Material: Vesconite	Drawn By: Matthew Wilson WLSMAT004		Drawing Number: 4110-49-10		

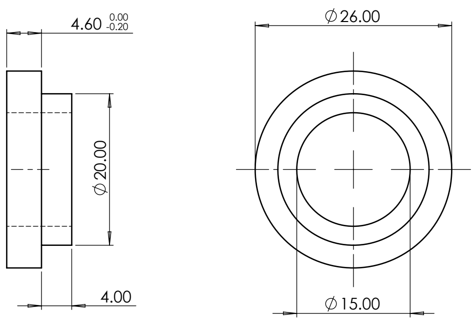




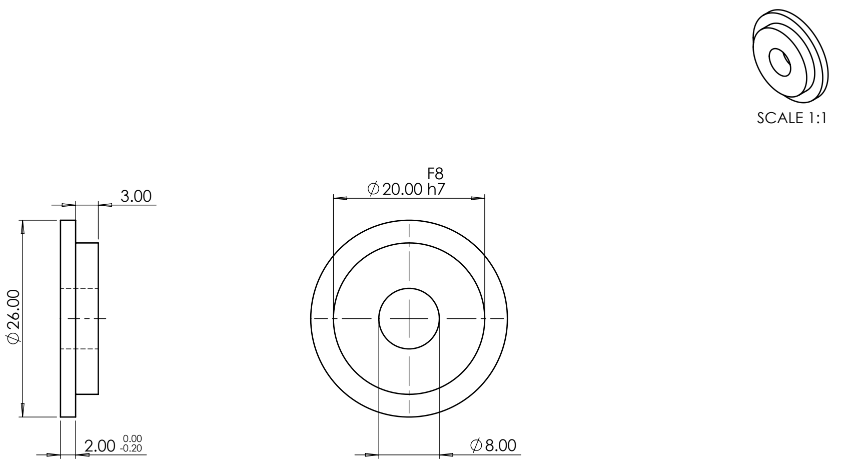


A4 Landscape	University of Cape Town Department of Mechanical Engineering					
	Title: Pinion Outer Bushin					
Quantity:	Part Finish:		Date:	Scale:	Sheet	of
8	Fine		2013/10/01	2:1	11	28
Material:		Drawn By:	Drawing Number:			
Vesconite		Matthew Wilson WLSMAT004	4110-49-11			



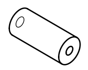


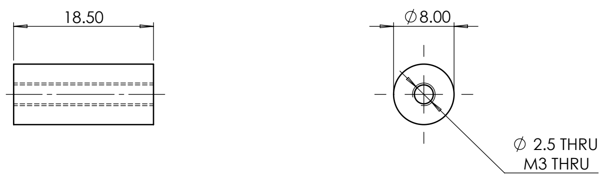
A4 Landscape	University of Cape Town Department of Mechanical Engineering				
	Title: Internal Bushing				
Quantity: 4	Part Finish: Fine	Date: 2013/10/01	Scale: 2:1	Sheet 12	of 28
Material: Vesconite	Drawn By: Matthew Wilson WLSMAT004		Drawing Number: 4110-49-12		



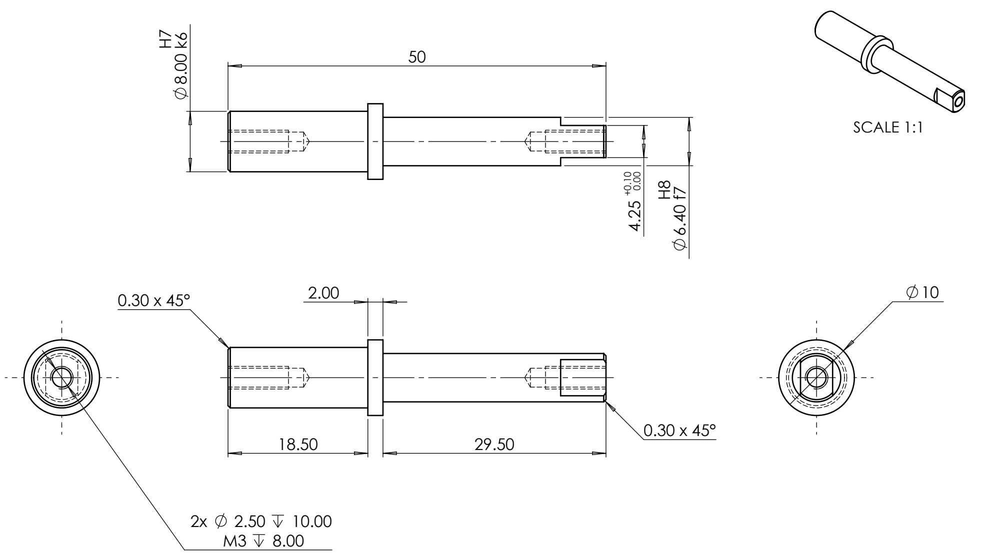
SolidWorks Student Edition.
For Academic Use Only.

A4 Landscape	University of Cape Town Department of Mechanical Engineering				
	Title: Outer Bushing				
Quantity:	Part Finish:	Date:	Scale:	Sheet	of
4	Fine	2013/10/01	2:1	13	28
Material: Vesconite	Drawn By: Matthew Wilson WLSMAT004		Drawing Number: 4110-49-13		

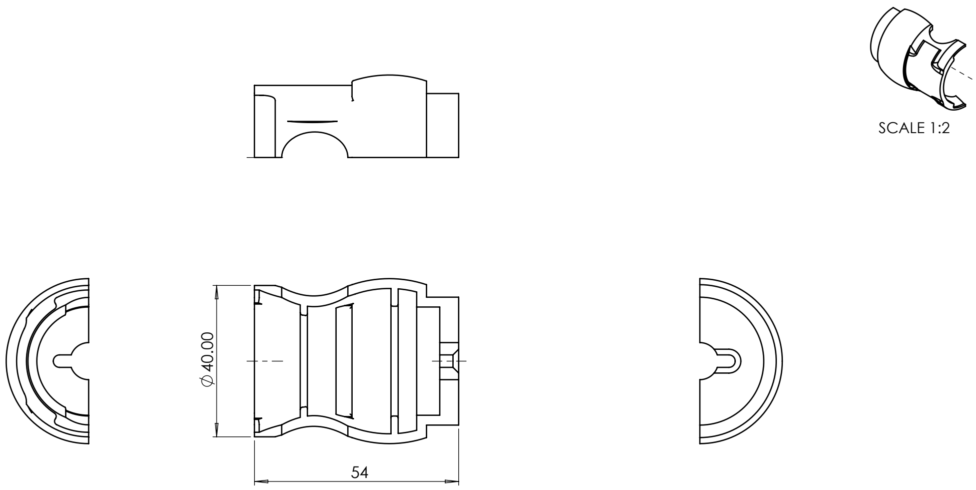




A4 Landscape	University of Cape Town Department of Mechanical Engineering				
	Title: Idler Shaft				
Quantity: 8	Part Finish: Medium	Date: 2013/10/01	Scale: 2:1	Sheet 14	of 28
Material: Mild steel	Drawn By: Matthew Wilson WLSMAT004		Drawing Number: 4110-49-14		



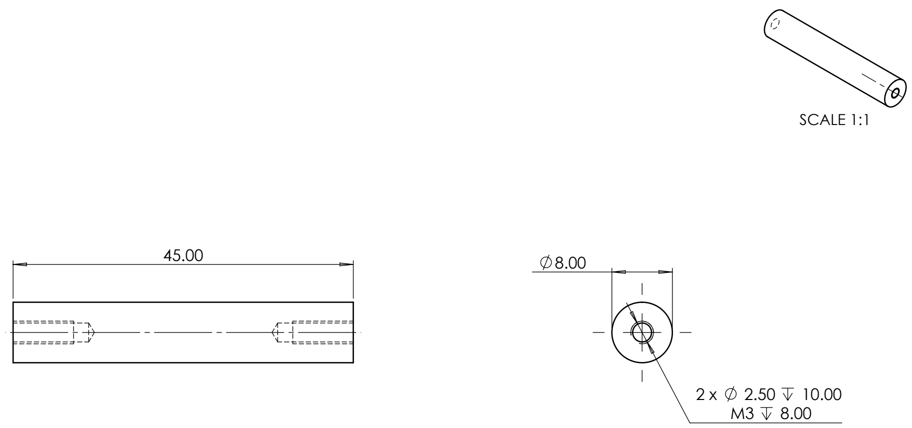
A4 Landscape	University of Cape Town Department of Mechanical Engineering				
	Title: Screwdriver Output Shaft				
Quantity:	Part Finish:	Date:	Scale:	Sheet	of
4	Fine	2013/10/01	2:1	15	28
Material: Mild Steel	Drawn By: Matthew Wilson WLSMAT004		Drawing Number: 4110-49-15		



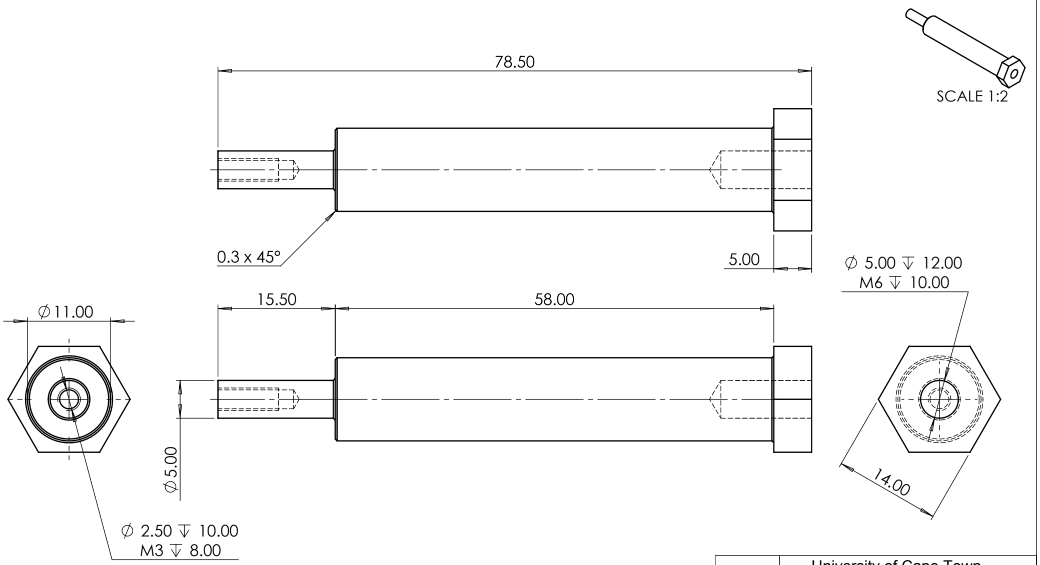
Note:
Dimensions given for modification
of standard part.

Do not manufacture from raw material.

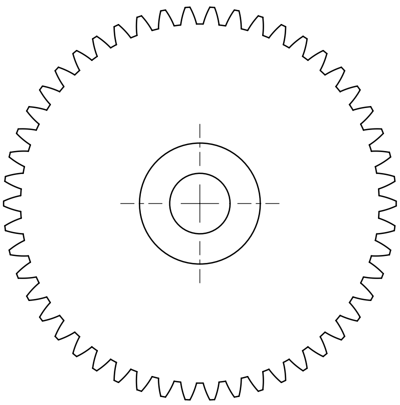
A4 Landscape	University of Cape Town Department of Mechanical Engineering				
	Title: Modified Motor Case				
Quantity: 8	Part Finish: As cut	Date: 2013/10/01	Scale: 1:1	Sheet 16	of 28
Material: Standard part	Drawn By: Matthew Wilson WLSMAT004		Drawing Number: 4110-49-16		

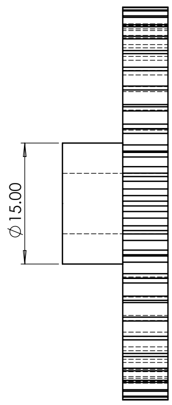


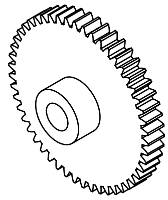
A4 Landscape	University of Cape Town Department of Mechanical Engineering				
	Title: Holding Pin				
Quantity:	Part Finish:	Date:	Scale:	Sheet	of
2	Medium	2013/10/01	2:1	17	28
Material: Mild steel	Drawn By: Matthew Wilson WLSMAT004		Drawing Number: 4110-49-17		



A4 Landscape	University of Cape Town Department of Mechanical Engineering				
	Title: Axle				
Quantity: 8	Part Finish: Medium	Date: 2013/10/01	Scale: 2:1	Sheet 18	of 28
Material: Mild Steel	Drawn By: Matthew Wilson WLSMAT004		Drawing Number: 4110-49-18		







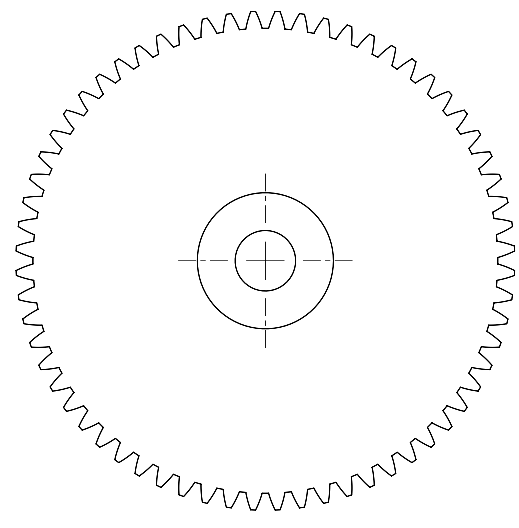
SCALE 1:1

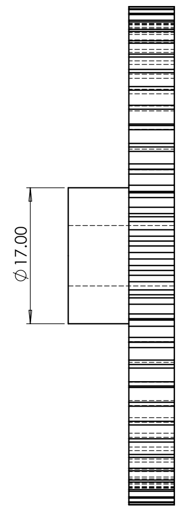
NOTE:

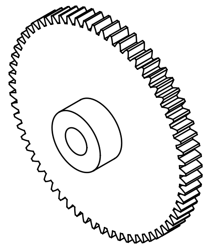
Modified from standard component only where dimensioned.

Do not manufacture from raw material

A4 Landscape	University of Cape Town Department of Mechanical Engineering				
	Title: Gear Modification				
Quantity: 4	Part Finish: Moderate	Date: 2013/10/01	Scale: 2:1	Sheet 19	of 28
Material: Standard part	Drawn By: Matthew Wilson WLSMAT004		Drawing Number: 4110-49-19		







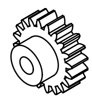
SCALE 1:1

NOTE:

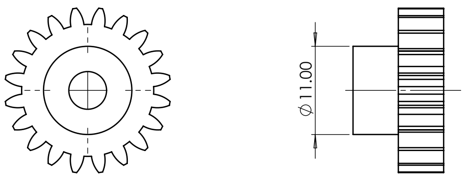
Modified from standard component only where dimensioned.

Do not manufacture from raw material

A4 Landscape	University of Cape Town Department of Mechanical Engineering				
	Title: Idler Modification				
Quantity: 8	Part Finish: Moderate	Date: 2013/10/01	Scale: 2:1	Sheet 20	of 28
Material: Standard part	Drawn By: Matthew Wilson WLSMAT004		Drawing Number: 4110-49-20		



SCALE 1:1

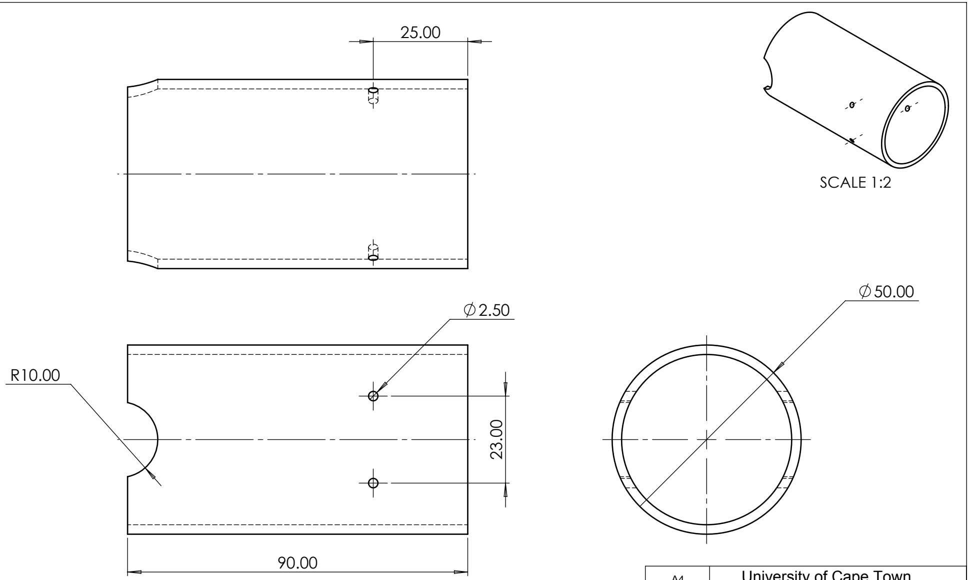


NOTE:

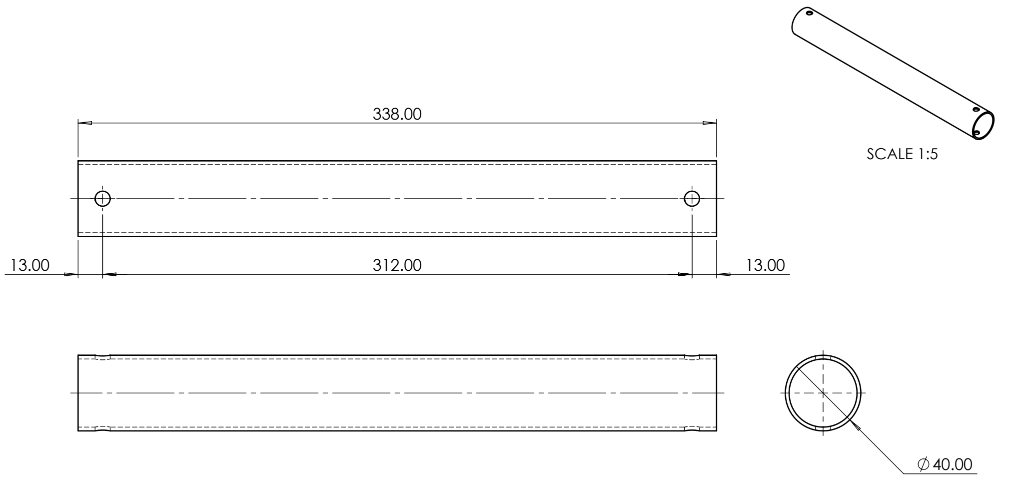
Modified from standard component only where dimensioned.

Do not manufacture from raw material

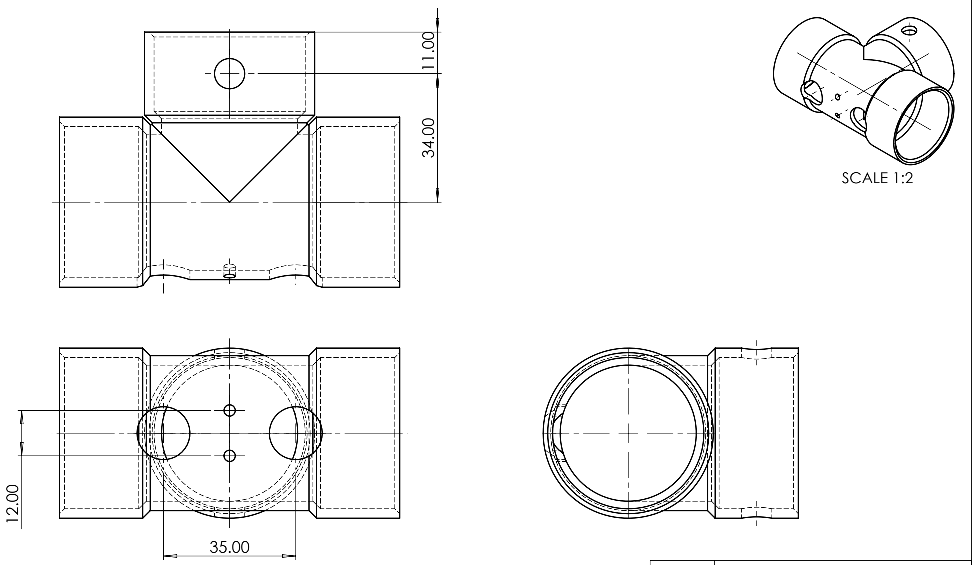
A4 Landscape	University of Cape Town Department of Mechanical Engineering				
	Title: Pinion Modification				
Quantity:	Part Finish:	Date:	Scale:	Sheet	of
8	Moderate	2013/10/01	2:1	21	28
Material: Standard Part	Drawn By: Matthew Wilson WLSMAT004	Drawing Number: 4110-49-21			



A4 Landscape	University of Cape Town Department of Mechanical Engineering				
	Title: 50mm PVC Cover				
Quantity: 4	Part Finish: Rough	Date: 2013/10/01	Scale: 1:1	Sheet 22	of 28
Material: PVC	Drawn By: Matthew Wilson WLSMAT004		Drawing Number: 4110-49-22		



A4 Landscape	University of Cape Town Department of Mechanical Engineering				
	Title: 40mm PVC Chassis Pipe				
Quantity: 1	Part Finish: As cut	Date: 2013/10/01	Scale: 1:2	Sheet 23	of 28
Material: PVC	Drawn By: Matthew Wilson WLSMAT004		Drawing Number: 4110-49-23		



A4 Landscape	University of Cape Town Department of Mechanical Engineering					
	Title: PVC T-Junction					
Quantity:	Part Finish:		Date:	Scale:	Sheet	of
2	As cut		2013/10/01	1:1	24	28
Material: PVC		Drawn By: Matthew Wilson WLSMAT004		Drawing Number: 4110-49-24		

Appendix F

Preliminary Report