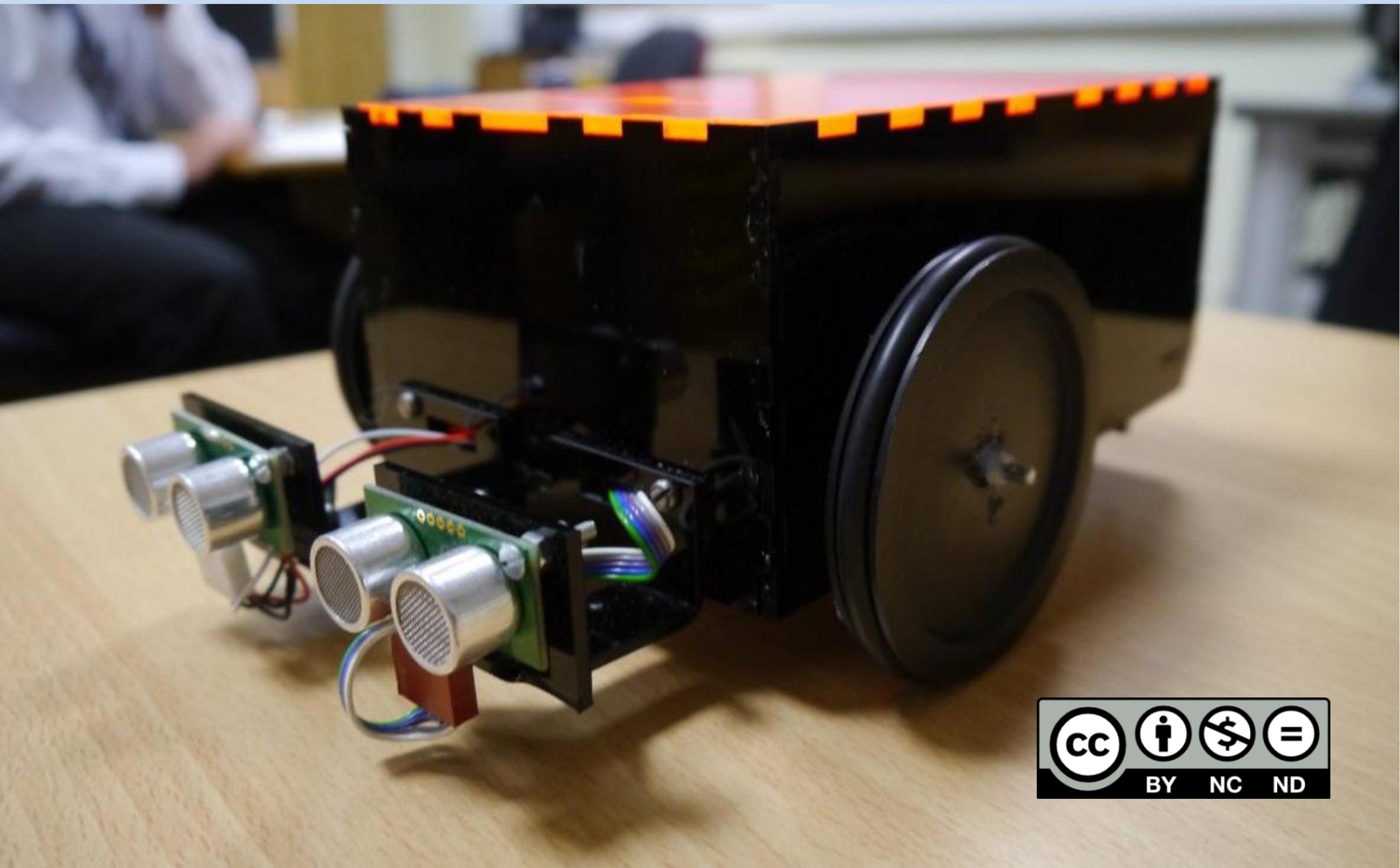
Theodore Heymann, David Shah, Nicolas Weninger

Overseen by: Cíaran Malik and Oliver Rokison

St Paul's School

Toyota Technology Challenge 2011-2012

{_}PIC.hacks



Contents

Licence

1 – General Information

1.1 – Introduction

We were informed of the Toyota Technology Challenge (TTC) in December. However, we were only informed of the deadline in mid-January. This left us six weeks to do the project. Where shortcuts have been taken, it is usually because we were somewhat short of time and were working to a very tight timescale.

1.2 – Team Identity

1.2a – Registration Information

Team Name	PIC.hacks
Team Members	Theodore Heymann
	David Shah
	Nicolas Weninger
Supervising Teachers	Cíaran Malik
	Oliver Rokison
School	St Paul's School, London

1.2b – Members

Name	**Year**	**Age**	**Team Role**
Nicolas Weninger	10	15	Team Manager and Head of Realisation
David Shah	10	14	Chief Systems Engineer
Theodore Heymann	10	15	Head of Design

1.2c – Ethos

With only 6 weeks to complete the project folder, it was necessary to work hard and efficiently to complete the folder on time. Although we did allocate tasks among ourselves, we worked very collaboratively, usually checking and improving upon each other's work. All key decisions were taken collaboratively.

The team consisted of a variety of different personalities, offering an advantage in terms of discussion and rationale of decisions throughout the project duration. Work was always spread out across the three of us, overlapping at times. The friendships within the group allowed this collaborative work to take place without any major problems.

1.2d – Name Choice

Initial brainstorming led us to the name *Elite Hacksaw*, reflecting a general enjoyment in tinkering with computers and electronics. We considered replacing the characters by other letters and symbols, but decided that this cluttered up the name too much.

Then the name PIChacks was suggested, a pun on the name PICAXE. We replaced this as our name. We tried various arrangements of this, such as PIChax, PIC•hacks and PicHacks. We settled with PIC.hacks as there is a clear separation between PIC and hacks in order to avoid mispronunciation and hacks is a real word.

1.2e – Logo Choice

We wanted the logo to be simple and to reflect our name or ethos. When our proposed name was *Elite Hackswaw*, a draft logo was created with a serrated blade on the bottom (*Image 1.1*). However, we dropped that logo in favour of curly braces enclosing an underscore in the fixed-width font 'Courier New' (*Image 1.2*).

This is a simple yet attractive logo that suggests programming and, perhaps, hacking. We also considered a logo where our name was written on a simplified microcontroller graphic, but thought it was too complicated (*Image 1.3*).

Image 1.1: Initial logo

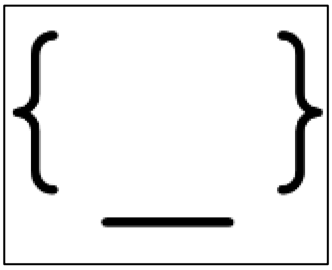


Image 1.2: Final logo

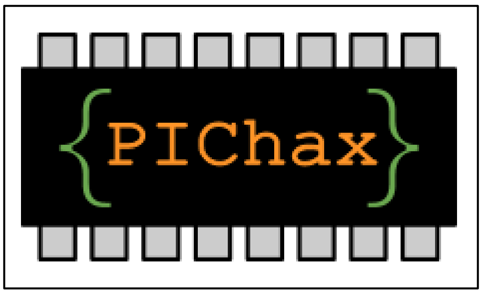


Image 1.3: A potential logo with name on chip

1.3 – Safety Concerns

1.3a – Using electronic components

- We only used components under adult supervision.
- We were careful not to short circuit batteries.
- We only used voltages up to 6V so as not to pose a health risk.
- We did not use a mains power supply.

1.3a – Using machinery and tools

- We always wore safety glasses when in the workshop.
- We were careful to tuck away loose clothing, such as ties, when working with power tools.
- We only used potentially hazardous machinery and tools under adult supervision.
- We did not use dangerous tools; we asked an adult to use them for us.
- We wore aprons when deemed necessary.

1.4 – Design Brief

To research, design, prototype and manufacture an environmentally-friendly intelligent vehicle able to navigate a random course using obstacle detection components without any external interference [remote control, etc.], controlled by a PICAXE microcontroller. The design must incorporate recycled and recyclable materials and be as efficient as possible.

1.5 – Necessary Specifications

Maximum Size: 300mm(l) x 190mm(w) x 150(h)

- Must have at least 3 wheels touching the ground at all times.
- Must have an on-off switch easily accessible from outside.
- Must use an environmentally friendly power source, preferably no more than 6V.
- Must be controlled by a PICAXE microcontroller.

- All electronic components must be sourced from Rapid Online.
- Must demonstrate eco-friendly features.

1.5a – Additional Constraints

The track walls are 50mm high. The obstacles are 150mm. All sensors must thus be able to detect obstacles at heights of less than 50mm from the floor.

1.6 – Final Specifications

Final Dimensions: 249mm(l) x 175mm(w) x 103mm(h)

- Two motorised wheels using 2-in-1 gearbox and a ball bearing third wheel.
- Powered by 5 rechargeable NiMH batteries or 4 alkaline batteries at 6V.
- On-off switch located on back.
- 2 Diagnostic LEDs and diagnostic buzzer.
- Two SRF05 ultrasonic range finders on custom mounts.
- A stacked PCB consisting of two modules.
- Logic powered by PICAXE 20X2 microcontroller.
- L293D motor driver.
- Acrylic case and MDF baseboard.

2 – Initial Research and Analysis

2.1 – Videos of Buggies from Previous Years

We watched two of the provided videos, Toyota Technology Challenge 2010-11[1] and Toyota Technology Challenge 2010-11 - Nick Freeman of Toyota & Chris Calver of Rapid Education [2], and one additional video, Toyota Technology Challenge 2009[3], to determine aspects of our vehicle that we would particularly need to consider when designing it.

2.1a – Conclusions

- Bumper switches were slow in detecting the correct way to turn, and the buggy must reverse before continuing, leading to a considerably slower completion time.
- Ultra-sonic range finders were slow when only one was used as the buggy has to turn twice before deciding the path of least resistance.
- The buggies appeared in general to be unable to correct their rotation in relation to the end after turning to avoid an obstacle. This lead to buggies moving the wrong way through the course and spending time travelling in circles.

2.2 – Human Modelling

We dragged several chairs into the middle of a computer room and spent half an hour pretending we were buggies detecting and navigating around obstacles. The aim of this was twofold: To experiment in an easy to comprehend way manner the combinations of sensors we might use, and to visually demonstrate potential algorithms to each other.

---

[1] rapidonline, 2011. Toyota Technology Challenge 2010-11. [video online] Available at: http://www.youtube.com/watch?feature=player_embedded&v=v6zXk-YHwYY [accessed 23/02/2012]
2rapidonline, 2011. Toyota Technology Challenge 2010-11 - Nick Freeman of Toyota & Chris Calver of Rapid Education. [video online] Available at: http://www.youtube.com/watch?feature=player_embedded&v=i9ajvf2RP_4 [accessed 23/02/2012]
[3] DBBuzzkiller, 2009. Toyota Technology Challenge 2009, [video online] Available at: http://www.youtube.com/watch?feature=player_detailpage&v=WHxPghURDgI [accessed 23/02/2012]

2.2a – Conclusions

- The buggies have a strong potential to end up facing the wrong way without careful programming.
- Travelling at 90 degree angles makes it easier to maintain device orientation and control. It could potentially be difficult to do this without using stepper motors or a built in recalibration system due to inaccurate motor spinning.

2.3 – Analysis of Motors

2.3a – Teaching Vehicle.

We used a buggy built for teaching Systems and Control (*Image 2.1*). With a 4.5V battery pack, the motors (though to be Rapid Online Order Number 70-2220, but not confirmed) turned very slowly. They did perform noticeably better with a 6V battery pack, and provided high torque, but were not fast enough for application in this project.

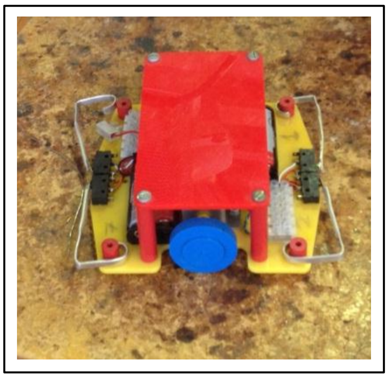


Image 2.1 – Test Buggy

2.3b – 2-in-1 Gearbox

Rapid Online Order Number: 13-1020

Using a 1F capacitor, we tested the efficiency of this motor and the motor described in *Section 2.3c*. *Image 2.2* shows the motor choices side by side. This motor and gearbox proved to be the most efficient, and was thus chosen for use in the final project. However, we were unable to carry out a full test as there was not enough charge held in the capacitor, so our results may not have been accurate. We were unable to measure current using a DC ammeter on a cheap multi-meter because readings were very erratic due to the brushed nature of the motor. It would be possible to drive the motors from a 6V power supply, so this would be the best voltage to use if possible to obtain the greatest speed and torque.

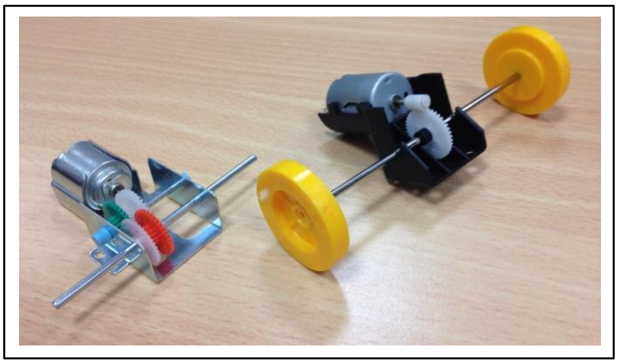


Image 2.2 – 2-in-1 Gearbox (left) and Worm Drive Gearbox (right)

2.3c – Worm Drive Gearbox

Rapid Online Order Number: 37-0310

We tested this motor and gearbox combination using the same method as in *Section 2.3b*. It proved to have fairly low torque and to be inefficient in relation to the 2-in-1 gearbox. The combination is also bigger, and so it might have been a challenge to find space in the buggy. This is the motor that came with the kit.

2.4 – PICAXE buggy with Optical Sensor

A PICAXE buggy equipped with an optical sensor was tested to locate some potential pitfalls. It was found that:

- Accurate turning was very difficult, particularly on carpeted surfaces. Calibration is needed to perform turns as precise as possible.
- Infrared distance sensors do not work very well. The distance measured varies depending on material, the voltage produced is not linearly proportional to the distance detected and they can appear to randomly detect non-existent objects, particularly in bright sunlight.

2.5 – TTC Kit

We assembled the supplied kit without any sensors. It appeared to move quite slowly, although we had no third wheel on this model, so this might have been inaccurate.

3 – Comparisons of Components, Design and Materials

3.1 – Sensor Types

3.1a – Micro-switches

Rapid Online Order Number: 78-2470

This is a sort of switch that operates with little pressure applied. It would be possible to detect obstacles in this way if the switches were placed on the front of the buggy, or use on the back to avoid crashing while reversing. See *Table 3.1* of advantages and disadvantages.

Table 3.1 – Advantages and Disadvantages of Micro-switches

Advantages	**Disadvantages**
Inexpensive (£0.58)	Requires a collision to detect object
Easy to program	Limited lifespan due to wear and tear
Use no power when not activated	Might break if car travels too fast
Require only one I/O pin	Very short range of detection

3.1b – Optical Distance Sensors

Rapid Online Order Number: 58-0982

This is a sensor that works by sending out an infrared beam that is reflected and timed to determine the distance from the object. They are quite expensive (£12.29), and it would be hard to justify the cost based on the disadvantages outline in *Table 3.2*. Testing proved the sensors almost useless for this application (*Section 2.4*).

Table 3.2 - Advantages and Disadvantages of Optical Distance Sensors

Advantages	Disadvantages
Can detect objects up to 80cm away.	Quite expensive (£12.29)
Easy to program	Detection varies significantly with material
Simple digital output (high if object>25cm)	Cannot find out exact distance to object
Require only one I/O pin	Hard to adjust

3.1c – Ultrasonic Range Finders

Rapid Online Order Number: 78-1085

A transducer send out a pulse which is reflected (or not) off an object and received by another transducer. The distance is worked out by measuring the length of time between sending and receiving the pulse. Testing on a breadboard showed this would be appropriate for this project (*Image 3.1*). The device tested was model SRF05 (*Image 3.1*). See the *Table 3.3*.

Table 3.3 - Advantages and Disadvantages of Ultrasonic Range Finders

Advantages	Disadvantages
Long range: about 4 metres	Expensive (£17.21)
Can sense a distance quite precisely	Might be hard to program from scratch
Can be configured to use 1 I/O pin	Ultrasound can affect animals that hear higher pitched noises than humans
Wide range of example code available	

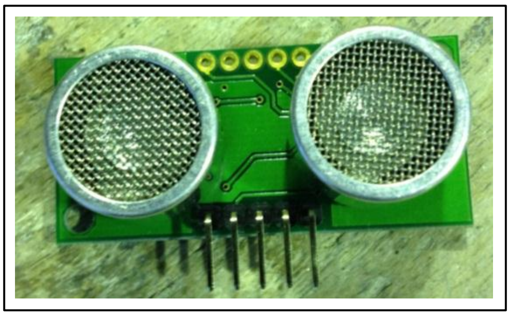


Image 3.1 – SRF05 Ultrasonic Range Finder

3.1c – High Performance Range Finders

Rapid Online Order Number: 78-1086

This is a newer, more advanced model of the ultrasonic range founder outlined in *Section 3.1c*. It has more features, but is also more expensive. We decided it was too expensive to use, and that if we did use it, we could only use one due to budget constraints. See *Table 3.4*.

Table 3.4 - Advantages and Disadvantages of High Performance Range Finders

Advantages	Disadvantages
Very long range: about 6m	Very expensive (£33.19)
Very accurate	Not RoHS compliant; contains a chemical that could harm the environment
Uses I2C bus, so would be easy to code	Too expensive to use two, limiting use
Can see multiple objects	Ultrasound can affect animals that hear higher pitched noises than humans
Variable gain enables precise calibration	
Built-in conversion to unit of choice	

3.1d – Digital Compass

Rapid Online Order Number: 78-1088

Although not able to detect obstacles in itself, it would be possible to accurately maintain 90 degree turning angles and always know which way the end of the course was. However, with calibration it would be possible to maintain a variable in the code with the orientation of the buggy, and a system was devised to automatically recalibrate while navigating the maze. Whilst there were some significant advantages to using the digital compass, the very high cost (£30.32) was not worth it.

3.1e – Conclusion

Of all the sensors we considered, the only two that were considered viable and useful to the project were Ultrasonic Range Finders and Micro-switches. Optical Distance Sensors were found to be not suitable for the buggy, and the Digital Compass and High Performance Range Finders were considered too expensive.

3.2 – Sensor Combinations

Having evaluated potential sensors (*Section 3.1*), it was necessary to decide what combination of sensors we should use. Only the combinations that were seriously considered are listed here; although other combinations might have been possible, we could not see any reason for choosing them at all.

3.2a – Two Micro-Switches

This would be an easy to code, easy to build and inexpensive option. Both switches would be located at the front, one at each side (*Image 3.2*). If the right switch is tripped, the buggy would reverse and turn left, and if the left switch is tripped, the buggy would reverse and turn right. This combination might allow the buggy to become stuck; in this case a randomised sequence could be used to free it. It would not be able to recalibrate on the fly. As physical contact is necessary every time the switch is activated, the switches may occasionally need to be replaced; they are prone to wear and tear, especially if the car travels at high speed. This is true of all combinations with micro-switches.

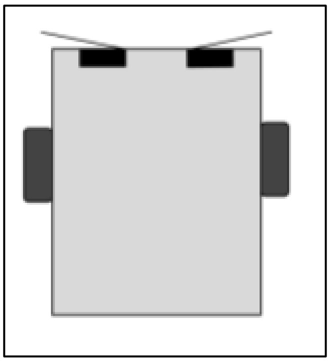


Image 3.2 - Two Micro-Switches

3.2b – Three Micro-Switches Option 1

This combination would work as with two micro-switches (*Section 3.2a*), but with a button micro-switch in the centre on the front. If the button micro-switch is tripped, the buggy turns a random direction.

3.2c – Three Micro-Switches Option 2

This combination would also work as with two micro-switches (*Section 3.2a*), but would have a micro-switch in the centre on the back. It would act as protection against reversing two much in small spaces, preventing damage to the case and gearbox system and reducing the chance of getting stuck.

3.2d – One Ultra Sonic Range Finder

Mounted on the front of the buggy, the range finder would be able to detect objects before hitting them, so no reversing would be necessary, and the vehicle needs to travel less far in total. However, if 90 degree turning angles are used, it would be necessary to turn each direction after detecting an obstacle to determine the route with least obstacles. This is the technique used by the buggy in the video analysed in *Section 2.1* entitled Toyota Technology Challenge 2009, and can easily result in a very slow completion time. This combination might work if the buggy does not travel at 90 degree angles, but it would be very hard to enable the buggy to remember the direction of the end.

3.2e – One Micro-Switch and Two Ultrasonic Range Finders

A micro-switch would be mounted on the front of the buggy and two ultrasonic range finders on the sides (*Image 3.3*). If the buggy bumps into an obstacle, the range finders on the side determine the direction of least resistance and the buggy turns that direction. The range finder pointing in the direction of the finish then constantly scans that direction while the buggy moves across the course until an opening is detected for the buggy to drive through. Using two sensors, it would be possible to perform calibration while navigating the course by using triangulation.

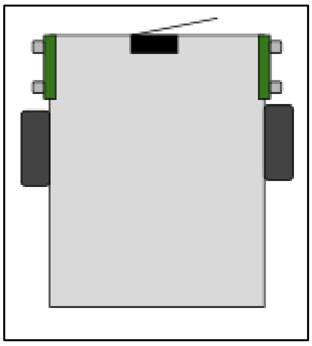


Image 3.4 - One Micro-Switch and Two Ultrasonic Range Finders

3.2f – Two Ultrasonic Range Finders

Mounted on swivel mounts on the front of the buggy, this is quite a versatile option. It would be possible to sensibly use several different algorithms as discussed in xxx, and automatic calibration would be possible using triangulation. The price would not be too excessive, making this a very sensible choice. If necessary, it would be possible to add a micro-switch on the back to prevent reversing into obstacles.

3.2g – Three Ultrasonic Range Finders

Using three sensors would enable quite complicated algorithms to be run efficiently. There are many different possible arrangements, but due to the difficulty in coding for three sensors and the high price, this option was not selected or fully explored.

3.2h – Conclusion

All the options, apart from the last one (three ultrasonic range finders), were economically viable. All were codeable. However, the option described in *Section 3.2f*, two ultrasonic range finders, was selected as it was deemed to be the most versatile option, capable of handling both very advanced and very simple algorithms.

3.3 – Motor and Wheel Combinations

As described in *Section 2.3b*, the 2-in-1 gearbox was chosen for the motors for the buggy. Testing showed it was efficient, it was not too expensive and it was readily available at school. We also considered using stepper motors, as these would have enabled precise turning, but they were too big and too expensive.

The specifications indicate that at least three wheels must be touching the ground at all times. Testing had previously shown that having parts of the vehicle dragging along the ground greatly reduces speed and efficiency anyway. The following section shows potential turning methods and wheels evaluated, with our final choice of layout.

Using four motors with four wheels would have been a good solution, but would have used double the power and would have required two L293D motor driver chips to handle the required current. This option is not discussed here as it was not deemed necessary or efficient. Likewise, linking two axles on each side was considered inefficient.

3.3a – Castor wheels

This is a sort of wheel that can rotate freely around its centre and is also able to rotate 360 degrees around the point of attachment to the underside of the buggy. Although this could be a good choice as it would avoid most problems of wheels skewing the buggy while turning, we decided against using a castor wheel as they can have a tendency to get stuck in a particular direction temporarily. This would significantly reduce the reliability, accuracy and efficiency of the buggy.

3.3b – Large Ball Bearing in Housing

It was brought to our attention that if the rules considered it as a wheel, an omni-directional ball bearing would be a good solution to the requirement for a third wheel that facilitates accurate and efficient movement and turning. It was confirmed that it would be allowed, and so this was used as the third wheel.

The ball bearing with housing was about the same height as the height of the chassis from the ground caused by the driving wheels, allowing the chassis to sit almost level on the ground. In order to not have any parts of the buggy dragging along the ground, it was decided to place the ball bearing towards the back of the vehicle as shown in *Image 3.5*.

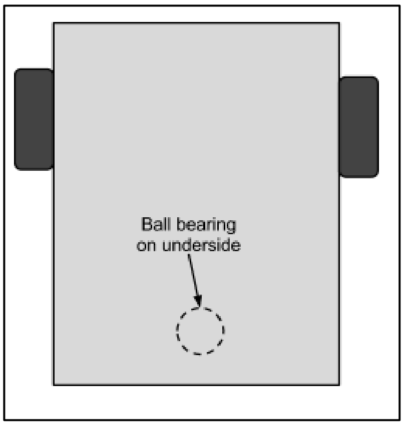


Image 3.5 - Final Arrangement of Wheels

3.3c – Turning Method

There were two main potential turning methods to choose from; reversing one wheel and keeping the other wheel turning forwards, or halting one wheel to act as a pivot point and turning the other wheel. As the buggy would not stop before turning, and would travel with considerable velocity and momentum, we decided that the latter option would work best. Wear and tear on the wheels, reducers, motors and gearboxes would thus be reduced.

3.3d – Using Pulse-Width Modulation to Control Motor Speed

As motors are usually slightly different, it might be necessary to limit the speed of one of them to make it uniform to make the vehicle travel straight and turn accurately. This is demonstrated in *Image 3.6*.

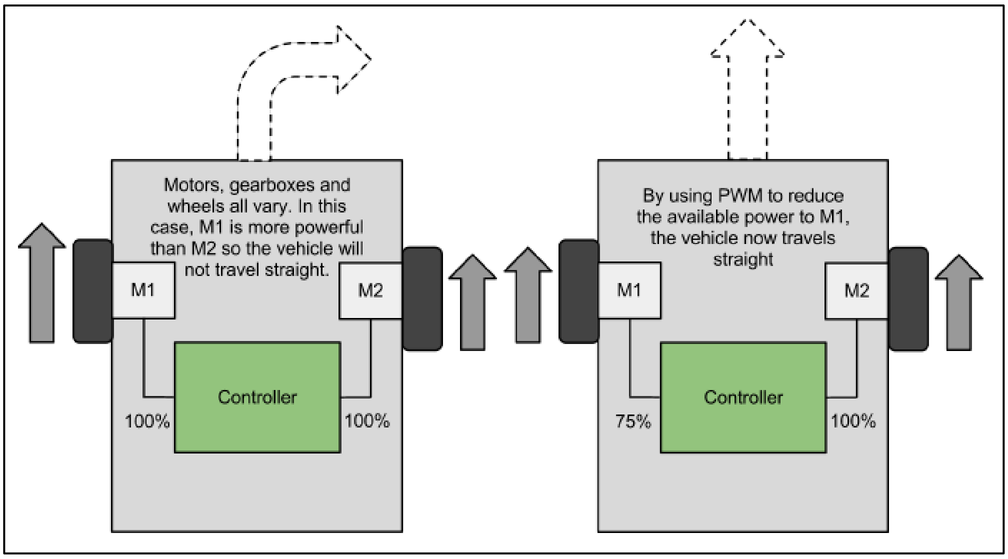


Image 3.6 - Using PWM to Modulate Motor Speed

3.4 – Power Source

3.4a – Battery Size

As the car would require a lot of power, and thus current, we decided to go for AA size batteries. AAA batteries are smaller, but have a correspondingly smaller capacity, and PP3 batteries have too high a voltage and are not designed for this sort of application. C or D cells have a greater capacity, but are too big and heavy for feasible use. Rechargeable C or D cells are expensive, and most battery chargers do not fit them.

3.4b – Rechargeable Batteries

Rechargeable NiMH batteries are more expensive than alkaline, but can be recharged and reused when run down. This means it is not necessary to buy new batteries when they have run down, making them more economical over time and more environmentally-friendly. It would in theory be possible to charge these from a solar charger, making powering the vehicle almost completely green, although we did not attempt this as we did not have such a charger available and thought that it would take too long to charge.

In high current situations, such as a buggy, NiMH batteries tend to last longer than alkaline batteries. Testing showed that the buggy ran noticeably better from NiMH rechargeable batteries. NiMH batteries also retain their voltage (1.2V) until they have almost run out of charge (See *Image 3.7*; the red flat line is for an NiMH battery discharge graph, the other line is for an alkaline battery. Batteries become unsuitable for use once their voltage drops below 1V). This means that it is much easier to calibrate turning; the buggy would turn faster or slower at different voltages.

However, due to the lower voltage than alkaline batteries (1.2V instead of 1.5V) it would be necessary to use 5 NiMH batteries instead of four alkaline batteries to obtain about 6V (See *Section 2.3b*). In order to facilitate changing the type of batteries to alkaline if absolutely necessary, a system was devised using an SPDT switch to easily switch between 4 alkaline batteries or 5 NiMH batteries (*Image 3.8*).

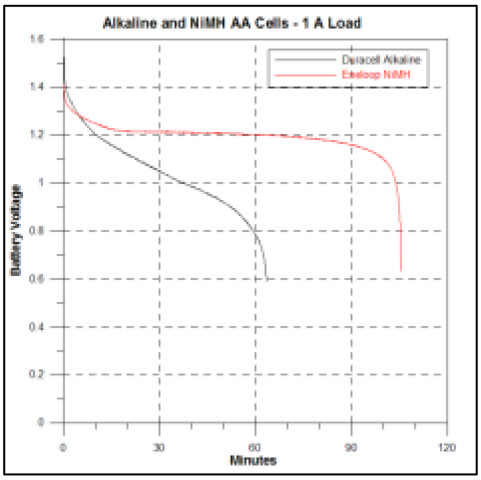


Image 3.7 - Comparison of Battery Discharge Curves

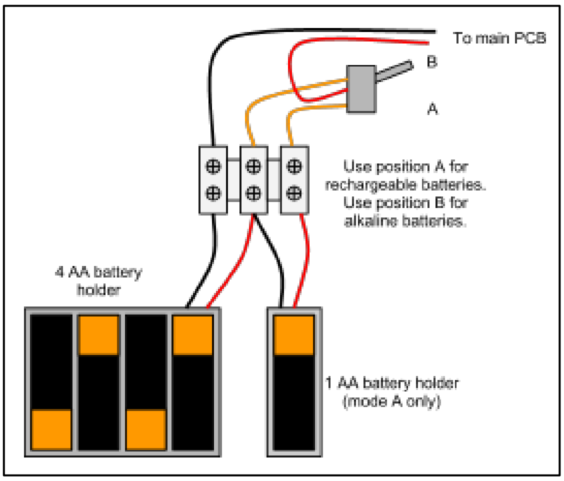


Image 3.8 - Battery Type Switch

3.4c – Alkaline Batteries

Alkaline batteries quickly run out of charge in high current situations, and do not have a discharge curve suited for accurate turning (*Section 3.4b*). They are very wasteful and bad for the environment. However, they are more easily available at school than rechargeable batteries. A system was designed that enabled alkaline batteries to be safely used on the buggy if necessary. We would not use alkaline batteries if not absolutely necessary.

3.4d – Use of a Power-Pack while Testing

We considered using a power-pack for testing to reduce battery wastage. However, as we had decided to use rechargeable batteries, this was not an issue. The leads from a power-pack would add simulated drag that would not allow us to accurately assess and test the vehicle. This idea was thus not put into practice.

3.5 – Algorithms

Algorithms are discussed in *Section 5.2*.

3.6 – Circuit Construction

As we decided to use two ultrasonic sensors, we were unable to use the TTC PCB which only supported one. There were two other options as outlined here.

3.6a – Breadboard on Vehicle

We considered placing a breadboard on the vehicle. This would mean all components, wires and the breadboard could be easily disassembled and reused if necessary, so would be fairly environmentally sound. It could also easily be reconfigured if necessary.

However, it would also be vulnerable to loose connections that might break. It would be heavy and completely impractical for any normal application, so this option was not selected.

3.6b – Own PCB

We had the possibility of our own PCB being made at school. As we were unable to use the TTC PCB (we needed space for an extra ultrasonic range finder and had chosen different motors), this was basically the only viable option.

As the PCB would need to very complex to accommodate all of the chosen components, it was decided to create a 'stacked' PCB; two PCBs placed on top of each other; the L293D was placed on the bottom layer, and most other components were placed on the top layer (not including switches, LEDs and ultrasonic range finders which were soldered to the board using wires and connectors). For most off-the-board components such as the ultrasonic range finder and motors, we used connectors instead of soldering the components directly to the board so components could be easily reused or replaced.

3.7 – Microcontroller

It was necessary to use a PICAXE microcontroller to control the buggy, ruling out other controllers such as an Arduino or a Rasberry Pi. The 18M2 was ruled out as it had too few I/O pins for this application. This left the 20M2 or 20X2, as it would be excessive to use a larger chip than necessary.

The 20M2 is cheaper than the 20X2 by about £1. However, the 20X2 has more memory, a faster processor and supports trigonometric functions. As there was a possibility of needing trigonometric functions to calibrate the vehicle, the 20X2 was selected.

3.8 – Material Evaluation

To build the car, we wanted a material that could be quickly and easily cut, preferably using the laser cutter, was aesthetically pleasing, light weight, and recycled or recyclable. To save time, we would want to avoid painting unless absolutely necessary.

3.9a – Acrylic

Table 3.5 - Advantages and Disadvantages of Acrylic

Advantages	Disadvantages
Aesthetically appealing	Production method involves oil and is bad for environment
Can be recycled or used as scrap	Quite heavy
Easily cut using laser cutter	Can scratch easily without protection, degrading appearance
Can be bent with strip heater	Needs a lot of energy for recycling
Could use scraps from various older projects	

3.8b – Polystyrene

Table 3.7 - Advantages and Disadvantages of Balsa Wood

Advantages	Disadvantages
Easily recycled	Probably have to vacuum form so would need wooden mould
Light	Potentially structurally unsound
Aesthetically pleasing	

3.8c – Balsa Wood

Table 3.8 - Advantages and Disadvantages of Balsa Wood

Advantages	Disadvantages
Easy to work with: can be cut manually or with the laser cutter	Less sturdy and strong than other materials, so possibility of damage
Very sustainable, if sourced correctly	Can look ugly when painted
Easily recyclable as scrap or to be made into sawdust	Painting it would waste time and it could chip, which would not look good
Extremely light (often used in airplane models)	
Can absorb impact well	

3.8d – Medium Density Fibreboard

Table 3.9 - Advantages and Disadvantages of MDF

Advantages	Disadvantages
Plenty at school, including unused scraps	Slightly more difficult to recycle, although can be turned into sawdust
Cheap	Sometimes considered to contain carcinogens, although not a problem in small amounts
Moderately strong	
Can be cut with laser cutter	

3.8e – Papier Mache

Table 3.10 - Advantages and Disadvantages of MDF

Advantages	Disadvantages
Recycled from old bits of paper	Inaccurate construction method

Fairly easy to construct	Slow construction; must be hand made
Very light	Hard to modify and make adjustments
	Ugly without loads of paint

3.8d – Old PCBs

Table 3.11 - Advantages and Disadvantages of MDF

Advantages	**Disadvantages**
David has many of these at home	Could be hazardous while cutting and drilling
Non-recyclable, so no further use	Hard to cut to specification
Components could be stripped and recycled.	Heavy

3.9 – Arrangement of Components

The three main bulky parts that needed to be fitted onto the chassis were the motors, the PCB and the batteries. These could not be laid out flat in the case because of length and width restrictions, so it would be necessary to stack two of them.

As the stacked PCB and the motor were both quite tall, it would not be possible to stack them in the case and stick within the height limit (150mm). As the PCB was delicate and so could not have anything on top of it, and the batteries needed to be accessible, the batteries were placed on a raised platform above the motors. The final arrangement is shown in the simplified *Image 3.9*. The battery platform is above the motors on spacers, but is not shown in the diagram.

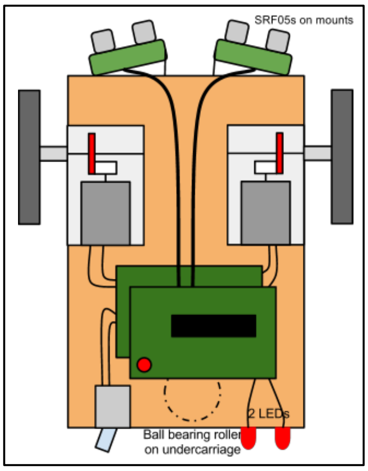


Image 3.9 - Component Layout

3.10 – Case Design

We choose black and orange as the principal colours for the case as they seemed fairly sleek and modern.

3.10a – Simple Cuboid

Our initial idea was a simple cuboid (*Image 3.8*). This could be easily assembled within the time limit out of a wide variety of materials. It would also comfortably fit all the components and would allow all components to be replaced relatively easily.

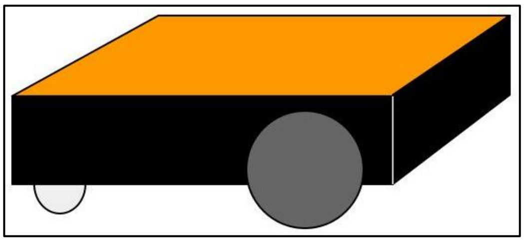


Image 3.8 – The cuboid

3.10b – Sloped Surfaces

We considered using sloped surfaces, particularly on the front of the buggy (*Image 3.9*). This design looks better than a simple cuboid (*Section 3.10a*) and is much more aerodynamic, but manufacturing this would require the use of a strip heater or very well made and sanded down teeth joints, limiting the materials that could be used. We could have had problems mounting the ultrasonic range finders on the front.

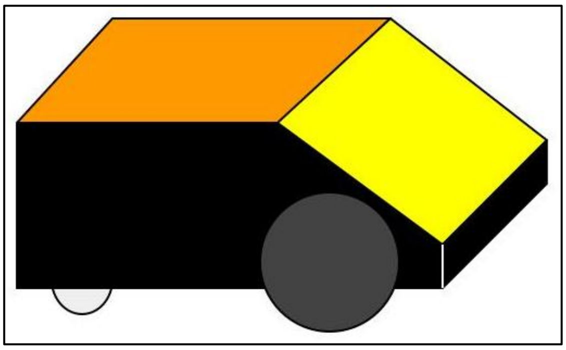


Image 3.9 – Sloped Surfaces

3.10c – Curved Surfaces

A curved front surface would be attractive but impractical. It might be hard to get at interior components, and could be challenging to mount the ultrasonic range finders. It would be necessary to vacuum form, injection mould, bend wood with steam or 3D print a curved surface, all options potentially available to us but none as practical as using the laser cutter.

3.10d – No Case

This could be seen as having a 'futuristic' look, but it would leave the components and interior workings very vulnerable to damage and destruction. It does, however, reduce the weight of the vehicle substantially.

3.10e – Conclusion

Although other options may be more attractive, the most practical option was to use the simple cuboid design outlined in *Section 3.10a*.

3.13 – Affixing the Case to the Chassis

This was an issue that caused some debate. It would be necessary to access the inside of the vehicle within about 10 seconds so components could easily be moved or replaced if necessary. This ruled out fastenings using bolts. As several components would be attached to the outside of the vehicle, if the case were to be fully removed it would have at least ten wires leading back into the heart of the vehicle. The problem could potentially be worked around by using plug-in connectors, so all components could be easily detached and reattached to the PCB. Other ideas centred on using metal pins, hinges or Velcro to secure the case. None of them were deemed practical in the time available.

However, another solution was proposed: the sides of the case (to which components would be attached) could be fixed to the base with bolts, and there could be a lift-off lid for simple maintenance. This idea was carried through to production.

4 – Construction and Manufacturing

4.1 – Printed Circuit Board

David was placed in charge of designing the PCB on PCB Wizard 3, a PCB design software on the school computers.

In order to allow the PCB to be as compact as possible in order to save space on the final design, the PCB was designed in two modules that would be stacked on top of each other using screws and spacers. 2.54mm headers were used to provide electrical power and connections between the two modules.

To facilitate the use of additional peripheral hardware, several pins were mapped out, including pins for additional buttons and the I2C bus, allowing for the potential use of a digital compass, an LCD display or external EEPROM storage device, should these devices be needed at a later date. A diode was used to reduce the voltage provided to the PICAXE and sensors.

The PCB was designed to include a 6V DC input, PICAXE-20X2 (also pin compatible with 20M2 if cost reduction is necessary), L293D motor driver with all driver pins connected, spare pads for 2 additional micro-switches and 2 SRF05 ultrasonic range finders running in single pin mode. In total, only two air wires were used.

To develop the PCB, an artwork view needed to be printed off on acetate and placed above an undeveloped PCB to be exposed to UV light. Sodium Hydroxide (NaOH) was to develop it followed by a bath in Sodium Persulfate to finalize the PCB. We were able to get a 24 hour turn around with this process, with the assistance of a teacher at all times.

4.2 – Ultrasonic Range Finder Mounts

We initially planned to purchase a mount from a vender; however, the ones we looked at were either the wrong shape or far too expensive, so we ended up designing our own. At this stage, the final case shape was not chosen, so we designed it to be able to swivel up, down, left and right, as to allow for the most flexibility later. We drew up an initial plan (*Image 4.2*).

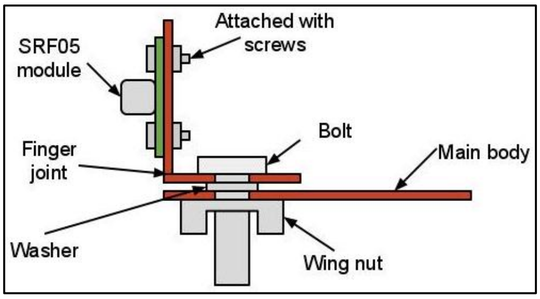


Image 4.2 - SRF05 Initial Mount Design

We chose acrylic for the material as the school had a large stock of scrap, it could be laser cut to specification and bent using a strip heater. *Image 4.3* and *Image 4.4* show our first prototype. M3 bolts and wing nuts held these individual parts together.

Image 4.3 - SRF05 Mount Prototype

Image 4.4 - SRF05 Mount Prototype (Constructed)

This needed to be sanded down and manually re-drilled using the pillar drill, as the mount was not able to swivel properly, prompting a change in the design file on Techsoft 2D Design.

Following further testing with the SRF05 mounted, the wire holes were redesigned and incorporated into the case design file. A second prototype was cut out on scrap, but a mistake lead to an engraved guide line being cut through (*Image 4.5*). This was fixed temporarily using spare bits of acrylic, to avid cutting out another copy (*Image 4.6*).

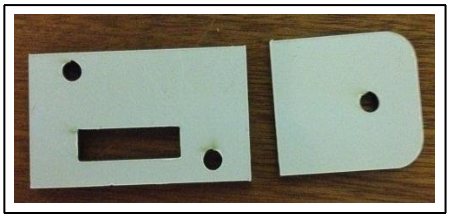


Image 4.5 - Split SRF05 Mount

Image 4.6 – Temporarily Fixed Mount

During mounting, it was realized that the holes were too small to affix the SRF05, so Nicolas tried to enlarge them using a pillar drill, breaking them. They were still usable for testing (*Image 4.7*).

The final version was designed when we knew the case design. As a result, the mount had to need to swing up and down anymore, simplifying the design immensely. These were cut out and mounted on the case (*Image 4.8*).

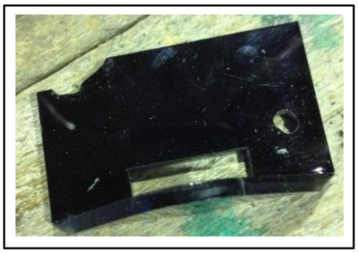


Image 4.7 - Broken Mount

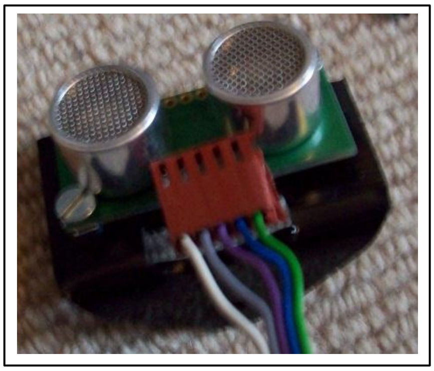
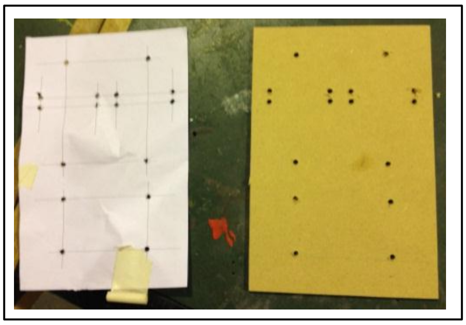


Image 4.8 – Final SRF05 Mount Design

4.3 – Chassis

We decided to use MDF to create the baseboard: it is easy to cut and drill and is lighter than acrylic. GCSE students were using the laser cutter when we wanted to make the chassis, so the basic shape was cut out on the band saw by a teacher.

Following this, we drilled out the holes using the design file as a template (fig.d+e). Holes were drilled for the motors, battery assembly and PCB assembly.



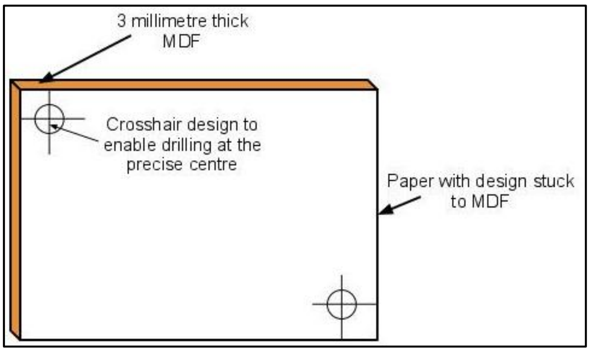


Image 4.9 - Drilled Chassis Holes

Image 4.10 – Chassis Drilling Method

We had initially planned to base it on the golden ratio for aesthetic reasons, but this would have made the base longer or wider than necessary, wasting material.

Throughout the manufacturing process, the holes needed to be made larger or repositioned using the drill. As we had chosen to use MDF, this prevented us from having to cut out many versions. Only one copy was ever cut.

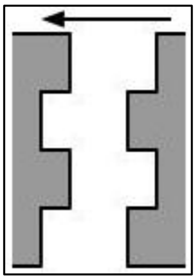
4.4 – Case

The standard cuboidal design was chosen for this with the proposed colour scheme as depicted, as we would easily be able to manufacture this within the time constraints and it would provide us with easy customizable options. It was also decided it should be made out of scrap acrylic wherever possible.

We designed the case around the following criteria: the interior should be easily accessible; the download socket should be reachable without the use of tools; the SRF05 sensor mounts should be mounted on the front with swivel mounts; there should be space for at least one LED and an SPST switch; it must not exceed the specified dimensions.

The case was designed on Techsoft 2D design using teeth joints to connect the sides (*Image 4.11*). We decided the case should include a removable lid and that the case could be removed within the 10-minute practice time with no difficulty.

The design was then modified, as we decided to have a slight overhang under the baseboard for aesthetic purposes and to lower the ultrasonic range finder under 50mm, but due to imperfections in the initial design, it was cut out incorrectly; heights varied by 1mm and one tooth did not fit together as planned. So as not to waste material, the longer tooth was filled down to specification and we disregarded the height difference as being unnoticeable (*Image 4.12*).



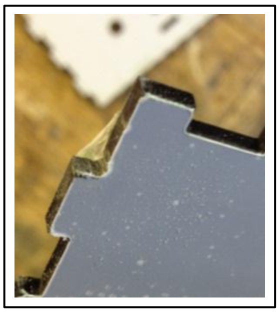


Image 4.11 - Tooth Joint

Image 4.12 – Filed Down Tooth

While we were checking to see that everything was working, it turned out that the holes for the switch and LED we had specified were too small. These were then enlarged using the pillar drill and the correct drill bit.

Following this, we attempted to mount the case onto the baseboard, using the method described in *Section 4.5*. It became apparent that the lid was too low for the internal components and the length of the sides were not long enough to incorporate the under hang, as the original case had not been designed to accommodate this feature. The case needed to be cut again on scrap acrylic.

The GCSE students, scrambling to finish their final projects, delayed progress and the eventual crash of the computing component of the laser cutter did not help either. Eventually, we had the modified design cut out with the correct dimensions and hole sizes. The only remaining issue was the hole for the download cable was positioned incorrectly. This was solved through the use of the pillar drill to enlarge it (*Image 4.13*).

Image 4.12 - Enlarged Download Socket Hole

4.5 – Assembly

The buggy was quicly assembled according to the plan in *Image 3.9*. It was built using M3 bolts for the battery layer above the motors & the PCB assembly and M2.5 bolts for the gearboxes. The case was to be affixed onto the baseboard through a series of 'tabs', which were attached to the case by solvent cement and to the base by M3 bolts. Unfortunately the solvent cement did not hold, so we resorted to using a hot glue gun during the testing phase. This method was later adopted, but we made sure that the glue was not visable from the outside.

In order not to be stranded by a broken component, we had to assemble the buggy in such a way that all key components could be easily removed and replaced without soldering or heavy tools. We connected the ultrasonic range finders and motors to the PCB through terminal blocks and the battery rig was completely removable. This way, should we be so unlucky as to have something break down, like a gear in the gearbox, we would be able to replace it quickly using spare parts.

4.6 – Equipment Used

Owing to the multi-faceted nature of the project, it was necessary to use a very wide range of equipment. Here is a list of large workshop machinery used during the course of the project. Small items of equipment such as screwdrivers and pencils are not included.

4.6a – Laser Cutter

Use: Cutting acrylic case and ultrasonic range finder mounts.

4.6b – Pillar Drill

Use: Drilling holes in chassis, case, ultrasonic range finder mounts and other modifications.

4.6c – Hot Wire Bender

Use: Bending ultrasonic range finder mounts

4.6d – Soldering Iron

Use: Soldering components

4.6e – Band Saw

Use (by teachers): Cutting edge of MDF chassis

5 – Product Testing and Program Development

5.1 – Vehicle Testing

5.1a – Speed of vehicle

The device performed noticeably better with rechargeable batteries in terms of maintaining a constant speed and direction as a result of the NiMH discharge curve compared to a standard alkaline battery.

Although the car may have been fast at initial sight, bringing up the possibility of instability on the track, during testing it appeared to go at a reasonable speed, and as we are using the 20X2 PICAXE chip, we can process information coming from the SRF05 sensors faster, allowing for a faster reaction time.

During early stages of testing, the vehicle reversed into a wooden cabinet, causing minor cosmetic damage. This mistake was due to a slight error in the program and was quickly corrected. (*Image 5.1*)

The option of adding a micro-switch to the rear of the vehicle was considered, as we had allowed for such a possibility on the PCB, however, it was rejected for fear of putting us behind schedule and lowering the aesthetic appeal of the vehicle. Such a switch would have made little difference anyway, as the reaction time would not be fast enough to avert the danger. We decided to stick with the initial plan.

Image 5.1 - Chip on Back of Vehicle

5.1b – Turning and Orientation

We managed to get a relatively accurate 90-degree turning circle on a polished wood floor with limited traction. This required minor calibration, which would be necessary on every different track surface, but easily done with the use of a variable in the program.

A straight direction of travel was achieved with the use of PWM. This was necessary as every DC motor will vary due to the manufacturing process and materials used to build it. This will require no calibration, we believe, as the differences in motor power output will remain the same.

5.1c – Sensors

We concluded that using one SRF05 during testing was not enough. This was done just in case the other SRF05 broke down and we were unable to replace it within the allocated time (*Image 5.2*).

Image 5.2 - Testing with One Ultrasonic Range Finder

We continued testing when the second SRF05 arrived from Rapid and found that when combined, they could easily negotiate a simple object using various methods, as described in *Section 5.2*.

5.2 – Algorithm Research and Development

5.2a – Buggy Angle Relative to Obstruction

In order to negotiate an object, it is very useful to find the angle of the device relative to the object. Some basic trigonometry is required.

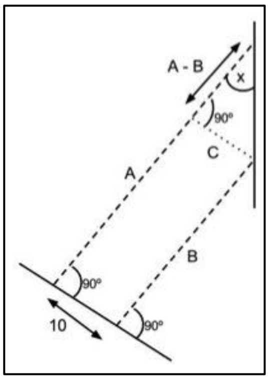


Image 5.3 - Trigonometry Required for Measuring Buggy Angle

As illustrated in *Image 5.3*, the two ultrasonic sensors are mounted at 90º from the base, and separated by 10, though this could be any number, and the distances they report form two lines, A and B. Another line is constructed, at the endpoint of B and perpendicular to B. We call this line C. C also has the length 10, because A and B are parallel two each other and originally separated by 10. This forms a right-angle triangle. We know that two of the sides are C (therefore 10) and A - B.

Therefore, we can calculate x:

$$x = \tan^{-1}(\frac{10}{a-b})$$

This requires the use of an 'inverse tan' function. The PICAXE 20X2 includes a function called "`arctan`" which does exactly this. As the PICAXE cannot handle decimals - and the input to inverse tan must be between 0 and 1 - the input must be multiplied by 100.

A code was written to show that it would be possible to provide different responses to an object depending on the angle of incidence, or for several other purposes including automatically 'straightening' so the device can attempt to travel parallel with an object, such as the course walls. However the PICAXE arctan function only works for angles up to 45º, but it would be easy to work around this. In order to prevent floating point or excessively large numbers, all these alternatives work with an input value 50x the original value and are intended to work up to 80º.

Attempting to re-implement the inverse tan function could do this. The first step would be to use a polynomial, generated through Excel's trend line function: A polynomial of order 2 diverges significantly from the required graph. The turning point of the polynomial would cause problems, so the next option would be to use an order 3 polynomial (*Image 5.4*).

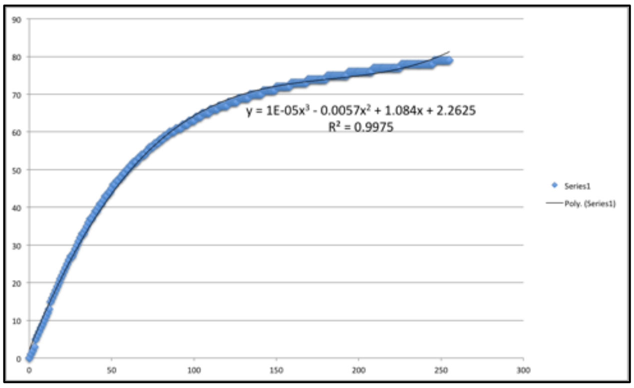


Image 5.4 - Order 3 Polynomial Graph

This would work very well, as shown by the r^2 value of 0.9975 (where 1 is a perfect fit.)

The equation it provides is:

$$\theta = \frac{x^3}{10^5} - 0.0057x^2 + 1.084x + 2.2625$$

Unfortunately the prevalence of decimals and powers render this equation unsuitable for the PICAXE microcontroller.

The other method, which shall be used in the project, is producing a look-up table. The values are pre-generated in Microsoft Excel and then stored in the PICAXE's 256-byte table memory using the TABLE directive. They can be accessed using the `readtable` command. Another advantage of the TABLE directive is that it also works on the PICAXE-20M2 (in fact, it is twice as large on the 20M2).

After generating the values in Excel, a simple VBA program was created to export the data from the spreadsheet into PICAXE table format.

The previous code, which used the atan function, can then be converted to use the `readtable` command (the actual lookup table has not been included in the sample code for clarity, though the above code would have to be included.

5.2b –- Simple Algorithm

The simplest sensible algorithm would, once one or more sensors detected an object, turn an arbitrary angle in whichever direction the object was furthest away. If the object were an equal distance to both sensors, it would either turn in a random direction or a hard-coded direction.

This method relies heavily on luck, is likely to get the buggy stuck and could be slow. It has the advantage of being very simple and requiring little processing power or code space.

This could be improved in several ways: The code as described above could be used to detect the angle of the obstacle and turn based on that angle. Catches would have to be used in several circumstances - for example if the angle is too great or if only one sensor detects the obstacle. Some method would have to be implemented if a corner is mistakenly detected - though in most cases this would not cause significant difficulty.

5.2c – Using a Variable to Record Angle

A variable could be used to keep track of the vehicle's current angle (approximately only) and then turn in such a way to make sure the vehicle attempts to head straight towards the target. This would be done by working out how long the motors have to run for to turn 1 degree, which can then be multiplied to produce the desired turn.

5.2d – Checking for an Obstruction

The vehicle could turn at random intervals towards the target if it is not pointing straight, and check if the path is free, as shown in *Image 5.4*.

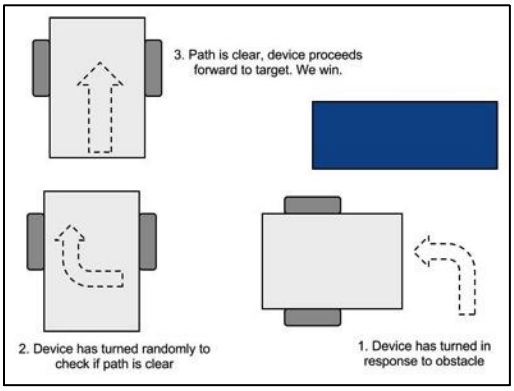


Image 5.4 - Checking for Obstacles

5.2e – Overall Algorithm Plan

A counter keeps track of the device's internal angle and is updated every time the device turns. Check if either sensor detects an object close by: If only one sensor detects an object (or the other gives an irrelevantly far reading), turn a random amount of degrees until neither sensor detect an object. If both detect a relatively close object, calculate the angle that the buggy is approaching the object, then turn that angle (*Image 5.5*).

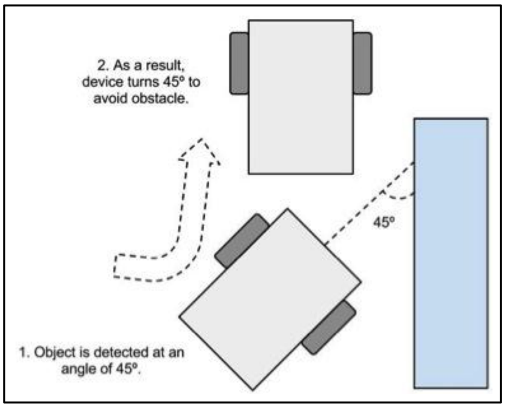


Image 5.5 - Turning at Obstacle

As we can assume all obstacles in the course are at 90º, 'snap' the internal angle variable to the nearest multiple of 90º. If the sensors are still obstructed after turning (for example, if the vehicle is stuck in a corner), keep turning until no obstruction is detected. If the device's angle is not equal to 0º (going straight ahead) after a random amount of time attempt to turn in such a way to get closer to 0º (up to a maximum of 90º each time) and check if the path is free with the method above. If it is free, take it.

6 – Critical Evaluation of Product

6.1 – Motors

The motors on the vehicle have a decent amount of torque and run at a high speed with a reasonable voltage supply. Mounting these to the baseboard was challenging, as the holes on the gearbox were very small, so we had to use small nuts and bolts. This might result in them becoming loose; however, this could be easily fixed within a few minutes.

6.2 – Case and Aesthetics

The case and baseboard were well designed to incorporate any possible future additions, like a micro switch; however, although it is structurally sound, the 'tabs' connecting the acrylic case to the MDF baseboard often became loose, meaning we needed to use a different bonding method many times. There is a risk of the same happening again, but this is minimal and would only occur upon heavy impact.

The overall aesthetics are good, with a modern sense that might not appeal to everyone. We tried to make the design fairly simple to avoid problems constructing it. However, the first version of the case had numerous flaws, and it was necessary to slightly redesign it. The second version worked.

6.3 – SRF05 Mounts

The SRF05 ultrasonic rangefinders were mounted on pieces of laser cut and strip-heater bended acrylic, offering an easy assembly and mounting method onto the main case. These could break if not handled correctly or on accidental impact. We plan to have spare parts to avoid this though. Initial ideas for the mount also intended for it to be able to move up-down as well; however, this became an unnecessary feature when we discovered the chassis would be flat anyway, and due to space limitations, this was not possible.

6.4 – Weight

The vehicle was quite heavy, lowering its efficiency and overall speed somewhat. Much of this weight is due to the large ball-bearing mount. As this is mounted on the rear of the vehicle, it provides stability, and allows the vehicle to be bottom heavy to avoid toppling over. Whilst it was a logical decision to use a ball bearing, we would probably try to find a different turning mechanism if we were to build another buggy.

6.5 – PCBs

The PCB consists of 2 layers: the control PCB, housing the PICAXE, and the motor driver PCB, housing the L293Ds. This saved space in the design while offering an easier job when wiring the components together. There is a problem with the stability of the top PCB, being only supported by two sets of headers and not additional spacers. If not handled delicately, this could result in a breakdown of communication between the control and driver PCBs, rendering the vehicle unusable. However, this should be easy to fix. Additional pads were added to the PCB, to break out the I2C bus should a digital compass or other advanced device need to be added.

6.6 – Dimensions

The vehicle is within the given size limit; the final dimension are 249mm(l) x 175mm(w) x 103mm(h). There was no problem in terms of height and length; however, the width proved to be more challenging; we had to cut down the axles and move wheels to comply with the constraint of 190mm. It ended up about 180mm wide.

6.7 – Assembly and Build Quality

Given the limited time we had, many aspects of the build quality were below expectations, although the vehicle is still rugged enough to survive a small collision. The positioning of loose wires, connecting the motor, SRF05s, LEDs, switches and battery unit, was not planned from the start, resulting in a slightly messy-looking interior and possible weaknesses between connections; however, most of these problems were solved later, when we stuck as many wires down as possible, especially around the gearboxes, and connections received an extra layer of solder to strengthen them. We propose to further fix this problem following folder submission.

6.8 – Eco-Friendliness

The case and chassis are all made out of recyclable materials, assembled with scrap material when possible, although the actual production process of acrylic is not the most environmentally friendly. Rechargeable AA cells are currently being used to power the vehicle, reducing waste and offering a more consistent voltage than traditional alkaline batteries.

6.9 – Technical Innovation

Throughout the project, we have encountered and solved many problems, such as those with the third wheel, the turning method, and the power supply. For example, we wanted the vehicle to be compatible with standard alkaline and NiMH rechargeable batteries, even though they run at different voltages. The solution was to create a method of switching between the configurations using a SPDT switch (*Image 3.8*). We believe that the vehicle and its algorithm have been built innovatively, and we this shows in the final product.

7 – Conclusion

This project was highly demanding, challenging and very good fun. Only having 6 weeks from start to finish provided an interesting goal for us, similar to the ones we will have to achieve next year for our Systems & Control GCSE course; however, having seen a completed product made to a high standard, we established it was certainly possible to create a high quality piece of work within a short time space. If the team is motivated and innovative, it can be done.

We plan to continue working on the vehicle and algorithm following folder submission to fine-tune various aspects of the vehicle in preparation for a much hoped for invite to the regional finals. We could, for example, work on a "calibration module," which would consist of a removable PCB with a digital compass in order to allow the device to automatically calibrate itself, saving time on the test track.

Along the way, we learned new skills, such as how to operate certain machinery, and improved our knowledge of the subject, such as differences and uses various material types, electronic theory and new BASIC commands, all of which will be of use in future years to come.

8 – Apendix (Photos of Finished Buggy)