

PREFACE Copyright

This work is released under the *Creative Commons Attribution-NoDerivs 3.0 license*

AS Systems & Control

Unmanned Automated Vehicle

Ben Goldsworthy

Candidate Number: 9056

Centre Number: 26202

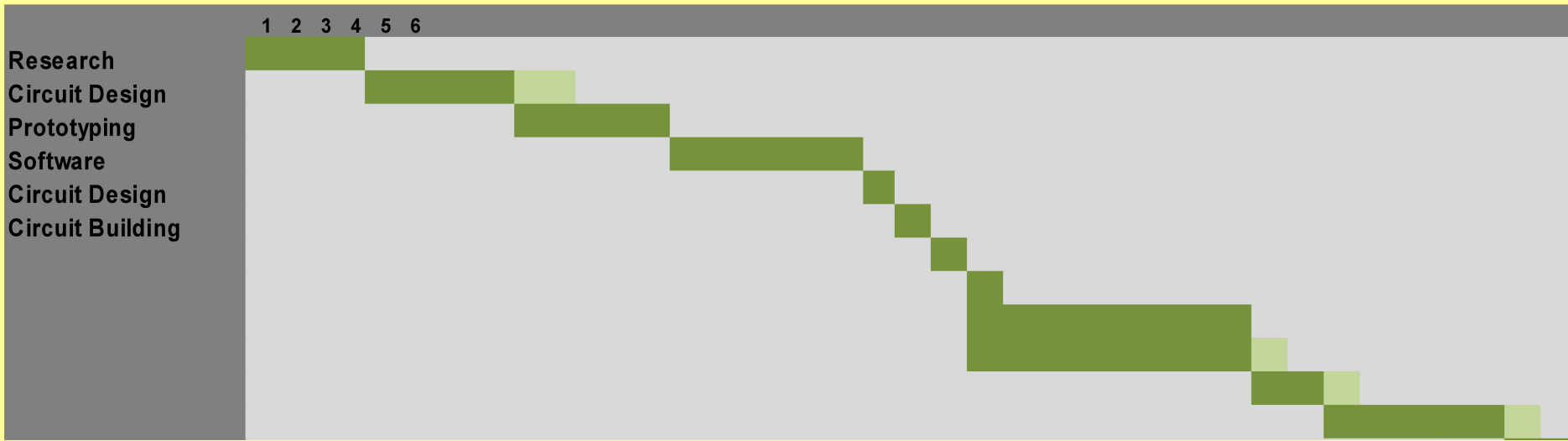
Bourne Grammar School

2013-2014

PROJECT Brief

A Bourne Grammar School teacher has asked for a car that will avoid obstacles without human interference. If I have time, a controllable mode would also be a good addition.

PROJECT Gantt Chart



This is a Gantt chart showing how I plan to divide up my time in this project.

As you can see, I expect the research, software programming and housing design stages to take substantially longer than the other stages. This is because I am planning to create a complex product that I predict will require a lot of research and a comprehensive programming solution. Housing design has always taken me a long time even on simple projects, so I imagine this shall be no different.

RESEARCH Project Ideas

Web controlled car w/ webcam feed

Idea 1

Web Controlled Car w/ Webcam Feed

This project consists of a car that can be controlled by issuing instructions from a website, and which would also have a webcam on board, the feed from which would be displayed on the website. This proof-of-concept project could then be used to explore dangerous areas or for surveillance.

This project would be complex and take a lot of work. Whilst it would be impressive to make, it doesn't seem particularly feasible. For one, I would need to host a website for controlling the car and find some way of getting a web server on the car, perhaps by using a *Raspberry Pi*, but that would make the project overly expensive.

The car will need a couple motors to drive it and an on board processor to run them, and a webcam unit. The website will need buttons with which to input instructions for movement and such as well as a video feed for the webcam.

The web-control and webcam aspects, then, are somewhat far-fetched, but I think I can do something more realistic with the remote controlled car idea...

Idea 2

Remote-Controlled Car

This product consists of a remote controlled car again, but this time controlled with a separate remote control device. This is a much more realistic project and clearly doable, as the prevalence of remote controlled cars in toy stores would suggest.

However, I feel this project is too simple and easy. It is also not particularly interesting, as remote controlled cars are a dime a dozen. It is certainly more doable than the previous idea, but ideally there will be some middle ground between the two.

This project will need the same car components as the previous idea, but the input method will be entirely different. It will need two 2-bit inputs for forward/backward and left/right motion, as well as PTM buttons for any cosmetic details such as horns and a slide or rocker switch for something like headlights.

Idea 3

Remote-Controlled Car w/ Unattended Mode

This project is an extension of the previous one, incorporating an alternative mode in which the car can be left to drive about on its own, avoiding hazards and lasting for as long as the batteries do.

I think I shall go through with this project. It's more feasible than the web controlled car, but also more complex and interesting than the vanilla remote controlled car. The unattended mode means that the car could be left to its own devices when showing off a school S&C department on open day, for example, or for trundling along in an environment with sheer drops aplenty.

As far as resources, this project will need same as the regular remote control car, although it may possibly need a controller with more inputs and outputs to handle the extra sensory data that it will need in order to drive unattended.

RESEARCH Project Interview

Before I started coming up with ideas for my project, I needed to sit down with Mr Brown, the client, in order to get a comprehensive product brief and ensure that he and I were on the same page.

ME: Hello, Mr Brown. Could you please give a brief overview of your request.
MR BROWN: Of course. I would like a small car that can be set to both an attended and an unattended mode, the former controlled via a separate remote control device and the latter driving around at random and being able to react to oncoming obstacles and drops.
ME: Are there any cosmetic touches you can think of?
MR BROWN: These would be entirely optional, but little things like headlights or a horn might be amusing.
ME: How are you imagining that unattended mode might work?
MR BROWN: If there was some way of detecting oncoming obstacles, the car could be programmed to stop, reverse a small amount, turn a random number of degrees and then drive on. This might lead to issues if the car was to drive into a corner, however, but I'll leave you to work that out.
ME: And how would the attended mode remote control work?
MR BROWN: I could press a button or push a lever on the remote control device and the car would move forward, until I released the button.
ME: Backwards too?
MR BROWN: If forward motion was achieved, backwards motion would be a mere act of reverse-engineering.
ME: And if I added any of the aforementioned cosmetic touches, place buttons on the remote control?
MR BROWN: Yes, particularly the horn, although if you were to implement LED headlights, having them turn on automatically based on external luminosity would be nice.
ME: I see. Thanks for your time.

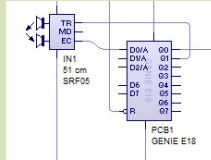
RESEARCH Plan

Area	Why?	Conclusions
Inputs	Inputs are devices that send data to the project, which can then be acted upon. Possible inputs stretch from switches to sensors and can detect all manner of both digital and analogue data. My car project will need a number of inputs, from the remote control device to the sensors that will keep the car from driving into obstacles.	I will need some sensors, such as an SRF05 ultrasonic distance sensor and some LDRs, in order for the unattended mode to work. Switches are also necessary for a master on/off and for the remote control. I will also need an infra-red or radio wave receiver on the car.
Processes	Processes are the aspects of the project that make the decisions using data gleaned from inputs. These include logic gates and ICs. For my project, I will need a number of processes for controlling the car and remote control.	A couple PIC microcontrollers, potentially C18s or C20s for their input and output quantities, should be suitable for both the car and the remote control. A 555 timer may be necessary for the horn and I may need transistors or n-channel MOSFETS to power larger components.
Outputs	Outputs are components that give out data, usually as a result of a process. Outputs can give off analogue and digital data such as light and sound. My car will need a number of outputs, most importantly the motors to drive it, and some larger components may need a transistor to power them.	Motors are an obvious component for the car, and LEDs and some form of sound production such as a loudspeaker or piezo siren for the more cosmetic features. I will also need an infra-red or radio wave transmitter on the remote control.
Materials	The chassis will need to be fairly durable, in case the car runs into something or falls in the course of its adventures, however it must also be light or else it might weight down the car too much.	There is a wide variety of materials from which I can make both the movement system and the chassis. I will use laser-cut wheels as they provide me with more flexibility in design and are cheap, and I will use MDF for the chassis, as this is a prototype proof-of-concept piece, and acrylic seems presumptuous.
Pre-existing Products	Looking at pre-existing products is important because it allows you to benefit from the discoveries of people who have previously developed similar products. There is a wide range of pre-existing products that I could look at, from remote-controlled toy cars for children to the Mars Exploration Rover and even AI programming.	I looked at pre-existing remote-control cars and the Mars rover. From this I gleaned that a lightweight shell and long-lasting power source were necessary, and a couple of ideas of how instructions could be transmitted and received.

RESEARCH Inputs



An **SRF05 ultrasonic range finder** could be used to detect obstacles and stop the car before it hits them. It works via a process similar to echolocation, shooting out ultrasonic pulses and measuring how long it takes for them to bounce back.



Advantages	Disadvantages
These sensors are vital for the car's ability to detect obstacles and such, feeding the car spatial data of its surroundings. Avoiding crashes will both lengthen the product's lifespan and allow for an unattended mode.	The SRF05 is possibly the most expensive component I will be looking at, running around £15 per item. It is also a bulky component, which will increase the car's weight.



A **mini-LDR** detects ambient light. It is a non-RoHS compliant component due to the presence of cadmium. The LDRs could be used for both light-dependent headlights and placed on stalks to detect if the car is in a corner.

Advantages	Disadvantages
An LDR deals with analogue data and is easy to implement, especially if I use PIC microcontrollers later on which support analogue inputs natively.	The component's non-RoHS compliance could impede sales in the EU.



A **thermistor** is a type of resistor where the resistance changes based on temperature.

Advantages	Disadvantages
Thermistors are cheap components, and can be used to automatically change, say, a potential divider based on temperature.	Limited scope for use in my project.

Sensors and Switches

A **slide switch** detects linear motion and could be used on the car as a master on/off switch. The SPST switch can be in either an on or off position, and some models have other positions between the two.



Advantages	Disadvantages
Having a master on/off switch would save battery power when the car isn't in use, as well as keeping it from turning headlights on whenever it gets placed in a cupboard. A multi-position slide switch could also be used for multiple speed settings.	A slide switch is perhaps easier to catch on something and accidentally activate/deactivate than other switch designs. It is also a large component.

A **rocker switch** is another option for a master on/off switch. The SPST switch can be in either an on or off position.



Advantages	Disadvantages
The advantages are much the same as the slide switch, although without the benefit of multiple positions.	Same as the slide switch but potentially less likely to be accidentally flipped.

An **infra-red receiver** is required to receive signals in the infra-red wavelength sent from an infra-red transmitter.



Advantages	Disadvantages
The infra-red receiver allows instructions to be sent over large distances without the need for physical connections.	The receiver and transmitter need to be lined up, and the instructions can only be so complex.

RESEARCH Processes



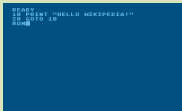
Python is high-level programming languages that can be well-suited for use as a scripting language.

Advantages

The language is powerful and the *Raspberry Pi* supports it natively. I am also somewhat familiar with it.

Disadvantages

Python is much too advanced for simple PIC controllers to deal with.



BASIC is a very simple programming language, and the programming language used to program PIC microcontrollers. GenieBASIC is the variant used by *Circuit Wizard*.

Advantages

BASIC is the native language for PIC microcontrollers and is a very easy language to pick up and start using (the BBC used it for their 80s *Computer Literacy Project*).

Disadvantages

The language is quite basic, and I don't know if the *Raspberry Pi* would support it.



Assembly language is about as low-level as a language can be without being pure binary. It is controlled using opcodes and is suitable for low-level projects as it is basic and resource inexpensive.

Advantages

Projects programmed in assembly language are very fast and resource inexpensive. If I learned to use assembly language, that would be a valuable skill.

Disadvantages

Assembly language is very difficult to use and I have no experience with it. The amount of time it would take to learn would leave me with little to spend on the rest of the project.

Instructions & Chips

A **555 timer** is an integrated circuit component that can be set to either monostable or astable mode; the former will activate something for an amount of time, the latter will turn something on and off repeatedly.



Advantage

Useful for using the loudspeaker, which I may use for a horn, although some PIC microcontrollers can do this without a 555 timer.

Disadvantages

If I don't even get around to implementing the horn feature, the 555 timer is entirely pointless.

PIC microcontrollers are a series of simple CPUs commonly used in schools and hobbies. They can take analogue inputs natively as well as being able control servo motors from the get-go. They are also programmed in BASIC.



Advantages

PIC microcontrollers are cheap as PIC chips and there is a wide variety of models with different properties. They can take analogue input natively.

Disadvantages

PIC microcontrollers are very low-level and can have limited numbers of inputs and outputs, as well as analogue support.

A **thyristor** is a switch that remains open even when voltage is removed, creating a latching circuit. This allows it to be used in alarm setups.



Advantage

Useful for latching circuits; perhaps I could implement an immobiliser into my project in case of remote control car theft?

Disadvantages

Limited scope to use a thyristor in my project.

RESEARCH Outputs



A pair of **motors** with a DPDT switch would allow for both wheels to move forwards and in reverse, which also allows for turning.

Advantages	Disadvantages
Motors are a vital component for a car, as without them it couldn't possibly drive.	Motors are bulky components and require a transistor to draw enough power.

A **piezo siren** is another possibility for producing sound. It uses a sheet piezoelectric material to generate noise.



Advantages	Disadvantages
The piezo siren also produces sound and requires little power.	The production of sound is not overly important, and if I used a piezo siren, I would be limited to a maximum of three irritating tones.



An **LED** displays a bright light when current passes through it, and they are available in many colours and sizes. LEDs are bright and last for a long time.

Advantages	Disadvantages
LEDs are very simple to implement, need just one 330Ω resistor and take very little charge to run. They last for a very long time.	Nothing involving LEDs is vital in my project, they are all extra details.

A **loudspeaker** oscillates when current is passed through, creating sound; either a tone or a sound file.



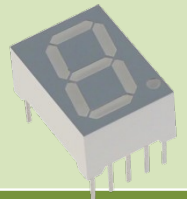
Advantages	Disadvantages
Allows me to play a sound for the fake engine sound and/or the horn.	Potentially even less vital than the LEDs.



A **servomotor** is a motor that can be set to a precise angles and kept there. I could perhaps use servo motors for a contracting 'storage mode'.

Advantages	Disadvantages
Natively supported by some PIC microcontrollers. I could use servomotors for any components I need to move about to precise positions.	Limited scope for servomotors in my project.

A **seven-segment display** consists of a number of LEDs under a screen, where each can illuminate one of eight segments. This allows for the printing of the numbers 0-9, and a decimal point.



Advantages	Disadvantages
Using a 4026B decade counter, I can easily implement a seven-segment display without having to figure out which PIC outputs go to which display pins.	Limited scope for seven segment displays in my project. If there is no 4026B handy, it will be a hassle to figure out which pins to use.

RESEARCH Materials



Laser-cut wheels, consisting of a lot of flat boards assembled to create a more 3D wheel shape, are another option. I could also use **3D printed wheels**, which have much the same pros and cons.

Advantages	Disadvantages
The nature of laser cutting/3D printing means that these wheels could be of potentially any design, and be cheap to produce.	I would likely need to run through multiple wheel designs, which would take time. Also, the wheels may be more fragile.



Caterpillar tracks are another choice, continuous loops of material over a series of wheels. These would work well if I went for a tank-like design.

Advantages	Disadvantages
These tracks are very all-terrain, which may come in handy depending on what the user wants to use the car for. They also give it a unique look, and the likely slow speed of the car means that the slow nature of caterpillar tracks isn't a problem.	Tracks can slip off, which can be a hassle. I also doubt I could get hold of any of these.



Blockboard is a manmade board that consists of strips of softwood pressed together and a veneer sheet of another wood over the top and bottom.

Advantages	Disadvantages
Blockboard is strong and lightweight, which would make it suitable for my project.	Blockboard is perhaps not quite lightweight enough for the small motors I expect to be using to be able to work with.



Purpose-made **wheels** would likely have softer rubber coatings with some degree of tread, as well as suiting the car-like image of the product better.

Advantages	Disadvantages
The rubber tire will lead to a softer ride, which could insulate the internal components. They also look more like real tires, which is a cosmetic victory, and the tread could potentially give the car more grip.	These might not be available for me.

Legs are a very advanced movement device that would make use of numerous servomotors to mimic the natural movement of legs.



Advantages	Disadvantages
If I managed to make a leg-based movement system, that would be a project in and of itself. The product would also be able to move off-road terrain.	Obviously incredibly complicated, hard to get stable even on flat ground.

Paper is without doubt the cheapest possible material, and is made of masses of pulped wood, usually bleached.



Advantages	Disadvantages
Paper is incredibly cheap and easy to source, and it would be easy to draw designs onto it.	Paper is just not strong enough to give the vehicle or control any degree of structural integrity.

RESEARCH Materials



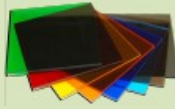
Corrugated fibreboard is a cheap option, consisting of corrugated flutes contained within two flat sheets of card

Advantages	Disadvantages
Corrugated fibreboard is incredibly cheap, making it an ideal material for prototyping if nothing else.	Fibreboard is also not strong at all, and looks cheap and unfinished. If I'm going for any degree of looks, fibreboard's off the cards.



MDF (medium-density fibreboard) consists of very densely packed wooden fibres

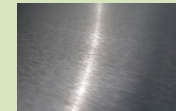
Advantages	Disadvantages
MDF is a strong material, whilst only being slightly more expensive than fibreboard. It also has a charmingly rustic aesthetic, which might be what I'm after.	Like fibreboard, MDF can make a product look unfinished. It's not colourful my any means, and it's illegal in the US and elsewhere due to presence of formaldehyde, which could be an issue.



Another option is **acrylic**, a clear plastic that comes in a variety of colours

Advantages	Disadvantages
Acrylic is also a strong material, whilst also having the advantage of being colourful and attractive. Acrylics also includes thermoplastics.	Acrylic is more expensive than the previous options.

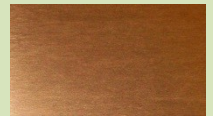
Steel is a more expensive metal option, and the tools to using it may not be present, but it is far stronger than any of the other materials.



Shell

Advantages	Disadvantages
Metal is very strong, and includes a wide variety of metals, all of which have different properties.	Metal is far more expensive than the other options, it may well be too heavy for the car, and it would need specialist equipment to work with that I don't have access to.

Copper is another metal choice. It is a softer metal, and so easier to work with without all of the fancy machinery (although it would still need some), but it is still expensive and potentially heavy.



Advantages	Disadvantages
Copper has the same advantages as steel, although may be cheaper and is easier to mould.	Same as steel

ABS & PLA are two plastics used in 3D printers. These devices have the advantage of being able to produce complex 3D shapes, which more complexity than a laser cutter.



Advantages	Disadvantages
With the 3D printer, I could build the entire frame & chassis in one piece, and it could be as complex as I like.	3D printing is mainly used for prototyping, as the products it produces are somewhat rough around the edges.

RESEARCH Pre-existing Products



Looking at pre-existing remote-control cars is useful for learning about the car design, as these products exist in the real marketplace and so have to be at least somewhat well-designed.

For the chassis design, the general theme seems to be lightweight, car-shaped (perhaps with a cosmetic shell to make it resemble a real car) and about 300mm long. They tend to be fairly durable and can withstand crashes, although this may be a result of not having any way of avoiding impacts. They often have bumpers on the front to alleviate some damage, and this may be a good place to hide the ultrasonic distance sensor. They are invariably made out of lightweight plastic of some sort. The remote control instructions are sent to the car via radio waves or occasionally infra-red LEDs,

and there will invariably be a small chip in the car to deal with the instructions. These require that both the transmitter and receiver are uncovered and working. Motors drive the car forwards and backwards and smaller servo motors turn the wheels left and right. Many contain lithium batteries, which give them far better battery life but are beyond my price range.

The remote controls tend to consist of two 2-bit inputs in the form of forwards/backwards and left/right knobs. This simple and intuitive control scheme allows the target market of young children to quickly get to grips with the product. I can see no way of improving this scheme, and will probably follow it, although a limitation is the linear steering control, rather than a more natural potentiometer-based method.

Parking sensors use ultrasonic transmitters/receivers which ping signals off of potential obstructions and glean from the time taken to receive how far away the obstacle is. If it falls under a certain threshold, an alarm will sound. I can use this system, practically unchanged, in my project, although rather than an alarm sounding, the car may just stop, and certainly will if in unattended mode.



An idea of how to use the distance sensors and such to avoid the robot ending up in trouble can be taken from the Mars rover, which contains a vast array of distance sensors, altimeters, thermometers etc. These are mostly housed in extruding stalks to give the Rover ample warning of any oncoming obstacles. Also from the Rover comes the idea of lightweight yet strong materials, in the Rover's case aluminium, and a long-lasting power source. In the Rover's case, this is a nuclear power generator, which I believe is slightly outside the scope of my project.

One other thing to be learned from the Mars rover is that no matter how complex the sensor array is, there is still the chance it may get stuck on a rock and die and I will have to anticipate my project not working perfectly.

RESEARCH Pre-existing Products

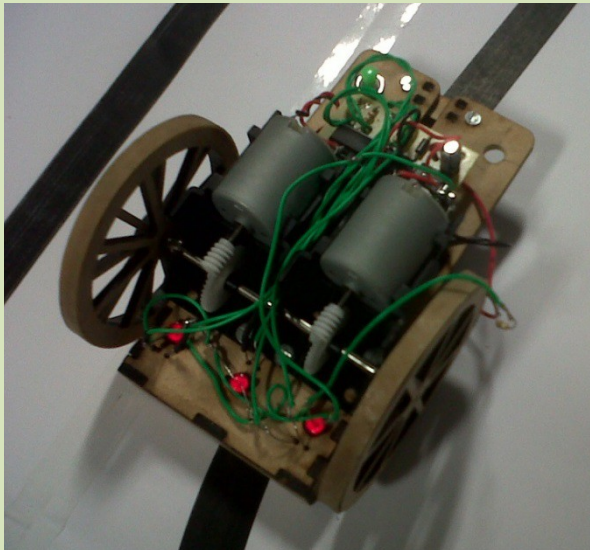
	RC cars	Parking sensors	Martian rover
Inputs	The control unit will contain switches for forward/backward and left/right movement. The car has to contain a radio/IR receiver somewhere on it, exposed to the outside.	The parking sensors use ultrasonic distance sensors to detect distance.	There are a vast array of inputs, from sensors to scanners. I obviously don't need even a fraction of what the Martian rover has on board, but there are a couple that could be relevant, such as ultrasonic distance sensors.
Processes	The cars contain small, basic chips to handle instructions. They can be entirely analogue, but this is rare.	Parking sensors have slightly more advanced chips as they have to deal with more information.	There is a very smart computer on the Martian rover, and it seems presumptuous to think I'll need it for my project.
Outputs	The remote control will have a radio/IR transmitter on it to send instructions to the car. There are always motors to drive the wheels forwards/backwards and often smaller servomotors for steering. There are often cosmetic details such as LEDs.	The outputs include the ultrasonic distance sensor, which acts as both an input and an output, LCD displays to output warning messages and distances and piezo sirens or loudspeakers to emit a warning noise if the car gets too close to an obstacle.	The outputs on the Martian rover are wheels and motors, which are relevant to my project, and servomotors and lasers, which aren't.
Materials	The cars and remote controls are generally made of durable yet lightweight plastic so that they can still move, but also withstand crashes.	More durable but heavier plastic (presumably ABS) is often used, as the device doesn't need to move but is more likely to be dropped or trod on.	The Martian rover is constructed from lightweight but very strong aluminium, which may be too expensive for my project, but does demonstrate the need for a lightweight shell.

RESEARCH Investigation



I decided to experiment with battery life in order to find out how long the product could run for and how much juice it would need to run it.

I first tried a 9V PP3 battery, as they have a very convenient shape for fitting snugly into a boxy design, and 9V is more than enough volts. However, the alkaline PP3 batteries only have a capacity of 565mAh, which is nowhere near enough to power my project. The motors alone draw around 300mA, under load, each, which would drain the battery in around fifty minutes ($565\text{mAh} / 600\text{mA} = 0.94\text{h}$), and the PIC will need some current too, as will a 7805, LEDs, sensors, etc.



I got hold of a line-following buggy (run off of 3xAA batteries) that was laying around the lab and tested batteries with it. The three cheap batteries made it twitch and whir, but were unable to move the wheels. The high quality batteries (pictured right), however, gave it the power it needed to drive.

I timed how long the car could drive for, which gave me a rough estimate for my somewhat-similar project. The car drove around, getting caught up occasionally, for 9m45s. This is fine for a proof-of-concept project.



AA batteries are far better. Whilst cheap zinc-carbon models, such as those pictured left, tend to be around 400-500mAh, which is again not enough to power my project, alkaline batteries (pictured below) are around 1800-2600mAh, plenty for my project's needs (for the motors alone, $1800\text{mAh} / 600\text{mA} = 3\text{h}$). Three 1.5V AA batteries also gives my project 4.5V, which should again be fine; 9V would have been massively excessive.



If performance was the goal, I could try to get hold of some lithium polymer batteries (output ~13,500mAh), but that seems excessive for this project.

RESEARCH Questionnaire

In order to gather some idea of what specification I should design my project along to, I crafted a questionnaire and gave it around to various peers of mine. The results of one such person are presented below, along with conclusions I could draw from the collected results.

Q: Do you think you would use the unattended mode feature?
A: Yes, it would be nice to leave it to do its own thing whilst I do other things, and it could amuse children.

Q: How much do you expect to be off-roading with the car?
A: The floors of the school are generally flat but the doorways could be a problem.

Q: What sort of speeds are you after?
A: Speed is not important, as this is more of a proof-of-concept project.

Q: Would you like headlights?
A: That would be useful if they came on in a darkened area, for example if I leave the car in a cupboard and forget where it is.

Q: How long would you need the battery life to last?
A: Again, this isn't important for this proof-of-concept.

Q: Does it need a way of interacting with the environment?
A: No, merely watching and waiting.

Q: Does it need to make an engine noise like a real car?
A: If at all possible, that would be a nice addition.

Q: What price would you expect the product to be?
A: I think around £20 sounds reasonable.

Q: Do you think you would use the attended or the unattended mode more?
A: I imagine I would end up using the unattended mode more, as it would be like a real-world screensaver (as long as the batteries held out)

Q: Do you need the car to work in light or dark conditions?
A: It would be nice if it would work in the dark.

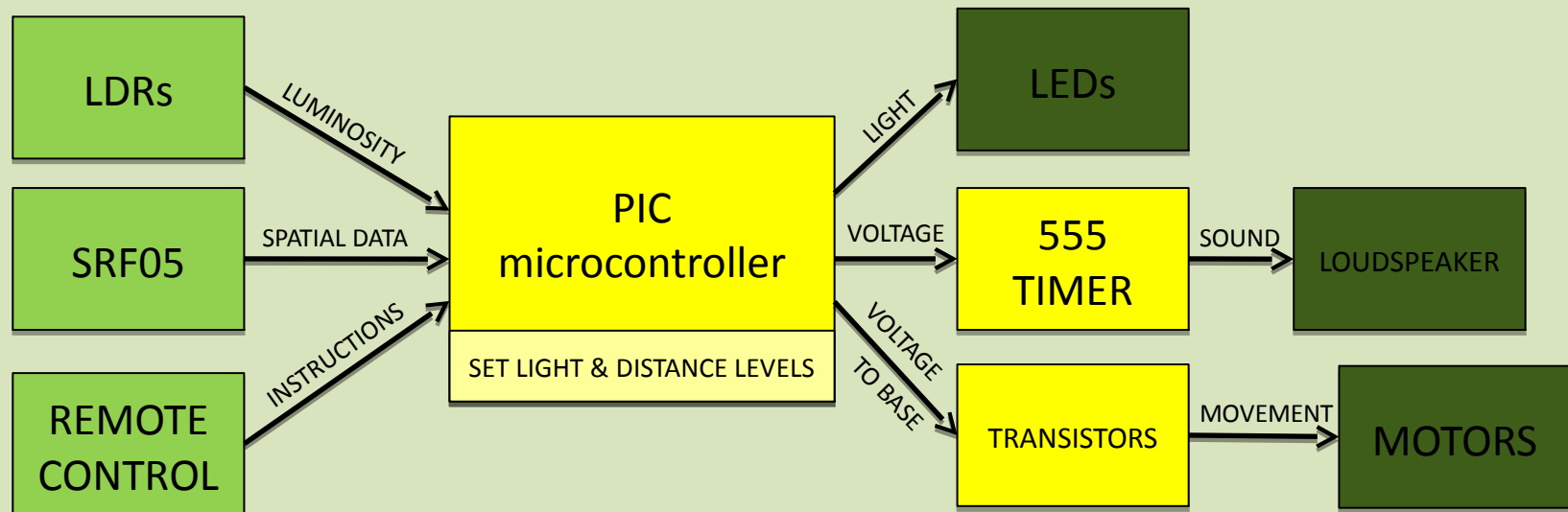
Q: Would you want a more advanced project that requires more maintenance or a simpler one that doesn't?
A: I feel a simple project that requires the minimum of maintenance would be better.

Q: Petrol, diesel or electric?
A: Electric, although maybe a hybrid? No, I'm sorry to even suggest it.

Q: Finally, would you be likely to buy the product for yourself or for a child you associate with?
A: I would likely buy it for myself under the guise of buying it for a child, to be honest.

From this I can design a comprehensive specification for the design of my project. 66% of respondents specified a size of around about that of a textbook, 75.2% said that they would not be using the vehicle off-road and a whopping 99% said that the battery life was a minor concern on account of the project being a proof-of concept.

RESEARCH System Block Diagram



This system block diagram is a hugely abstracted depiction of how the system will work and the constituent parts interlink. It shows inputs in light green, processes in yellow and outputs in dark green.

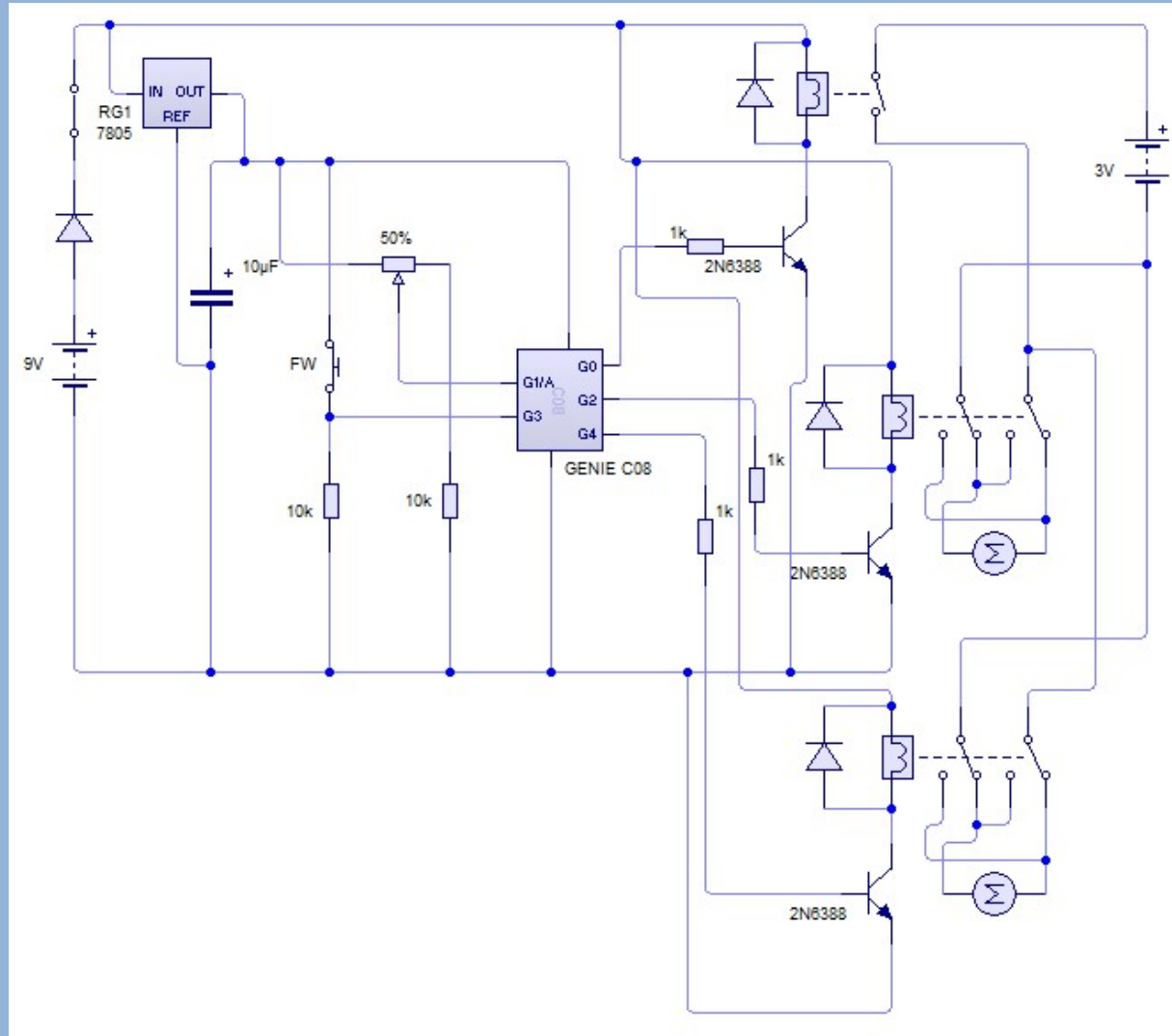
The LDR and SRF05 feed data to the C20, which compares the former's contributions to set levels to determine whether or not to turn on headlights. The remote control also feeds instructions into the C20, which leads to motor movement and horns blaring.

RESEARCH Specification

- Size** The car must be 200mm $\pm$ 10mm from bumper to bumper, approximately the size of a netbook. The controller must be 150mm $\pm$ 10mm in order to fit in the hands easily.
- Ergonomics** The remote control must weigh between 120-480g and the controls must all be reachable from a holding position, without any grip movement necessary.
- Aesthetics** Function is more important, although a rough car shape would be a goal. If a stranger can be asked to describe what the product resembles and replies with “a car”, this shall be considered fulfilled.
- Power** The product must be able to run for 10mins $\pm$ 2mins on 3xAA premium-quality batteries.
- Components** The school will cover the first £10 of the project; after that I’m on my own, so the components must not come to much more than £15 $\pm$ £2, and they must collectively not weigh so much as to make the car unable to move.
- Programming** The program must be compact enough to be stored on the PIC microcontroller without needing any extra EEPROM memory, and an unfamiliar user must be able to understand the code via provided comments.
- Materials** The car must be durable enough to survive a collision and light enough to move when told.
- Product Testing** When I have completed the project, I shall create a test plan to test both the hardware and software of the product.
- Maintenance** Beyond battery replacements, there must be no tasks that require the user (e.g. changing oil).
- Product Cost** This is a proof of concept, but a price of £20 $\pm$ £5 to cover the predicted cost of £15 $\pm$ £2 would be reasonable in a retail version.
- Product Life Span** Notwithstanding crashes and falls, the product could conceivably work for years if given a constant stream batteries.
- Unattended Mode** The car must be able to left to its own devices for 5 minutes without getting trapped.

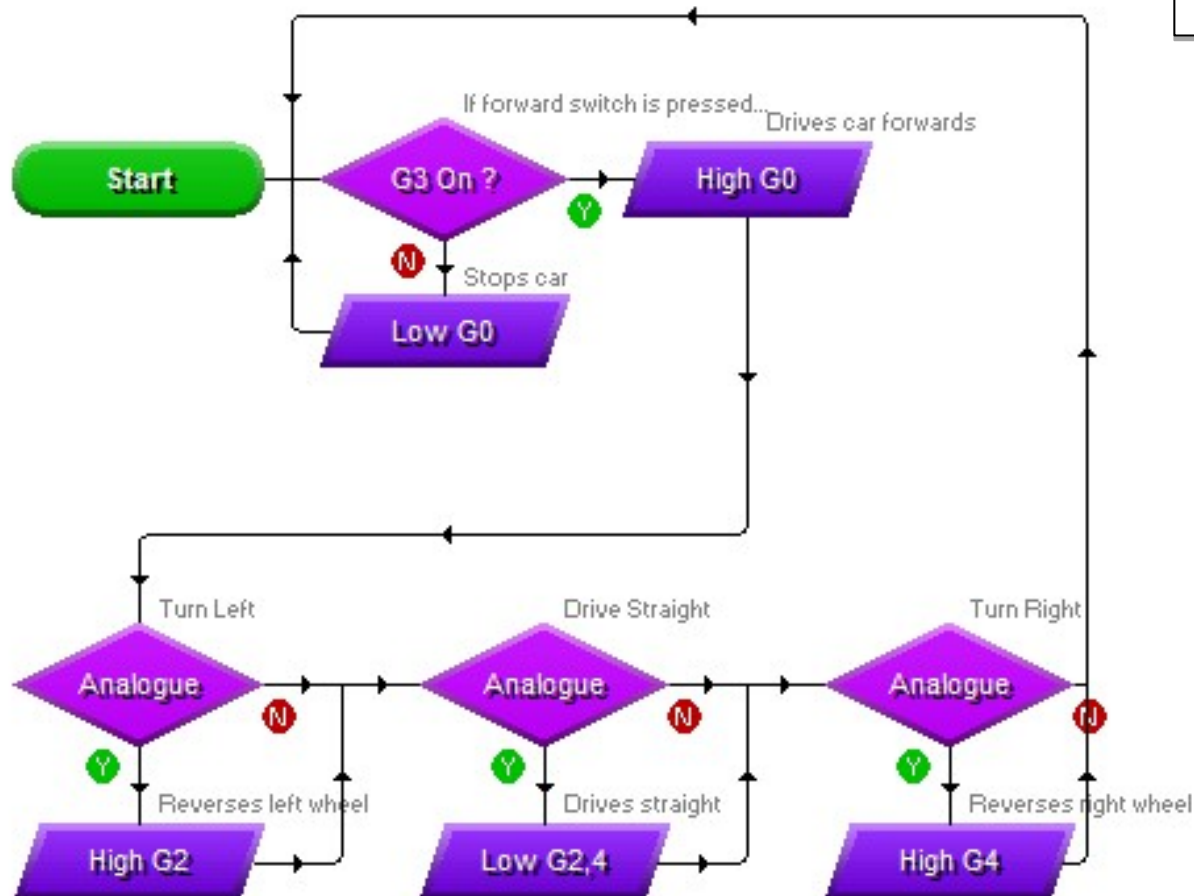
DESIGN Circuit 1

Overview



DESIGN Circuit 1

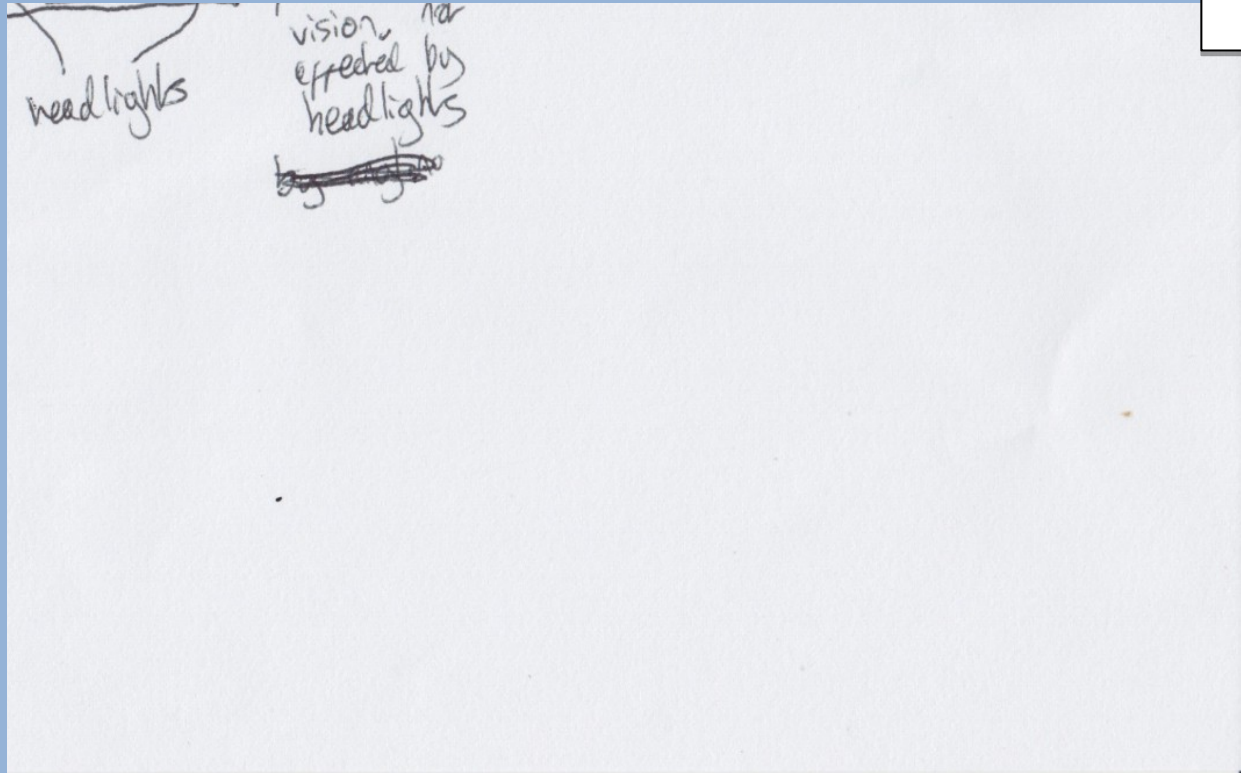
Software



The program for this circuit, here planned out with a flowchart, checks to see if the forward movement button is pressed. If so, it moves on to checking which way the steering potentiometer is turned, before returning to check the forward button again. If not, it stops the car and checks the forward button again.

DESIGN Circuit 1

Sketch

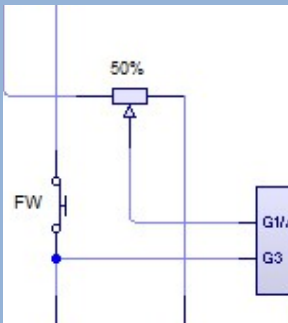


DESIGN Circuit 1

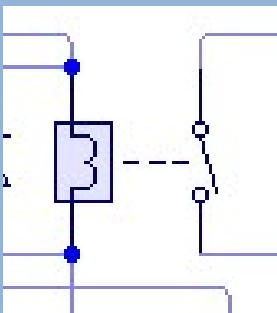
Details

Size	6
Ergonomics	8
Aesthetics	7
Power	5
Components	7
Programming	9
Materials	n/a
Product Testing	n/a
Maintenance	8
Product Cost	8
Product Life Span	7
Unattended Mode	6

My first circuit design uses DPDT relay switches, above which is a SPST relay switch, to control two motors, and a potentiometer and PTM switch combination to determine direction. The advantage of this one is that the potentiometer makes for a unique steering method and it is a fairly simple circuit. The disadvantage is that the relays will drain power quickly and can be expensive.

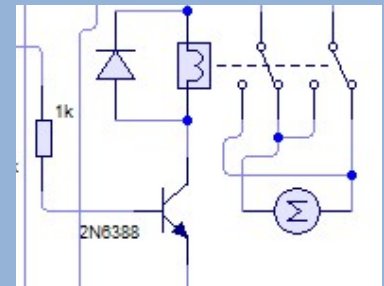


An issue with this design is that a potentiometer is not the best method of controlling the car. It can't do quick turns very well, although perhaps a steering wheel shape could be laser cut and placed on top in order to make it easier to use.

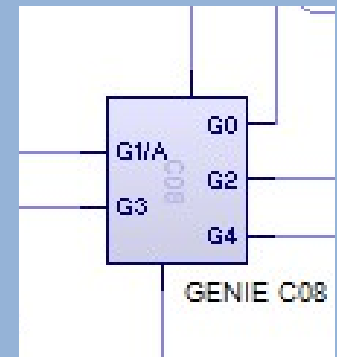


This design requires a lot of relay components. These use an electromagnetic coil to magnetically open and close switches on a separate circuit. If I wanted to add reversing to this method, it would require a few more relays and some complex wiring.

I have had to use NPN transistors for each of the relays, as the amount of current output by a PIC is not enough to power them and they must be connected directly to the 9V rail. I could also use n-channel MOSFETs, which are similar devices for powering greedier components.

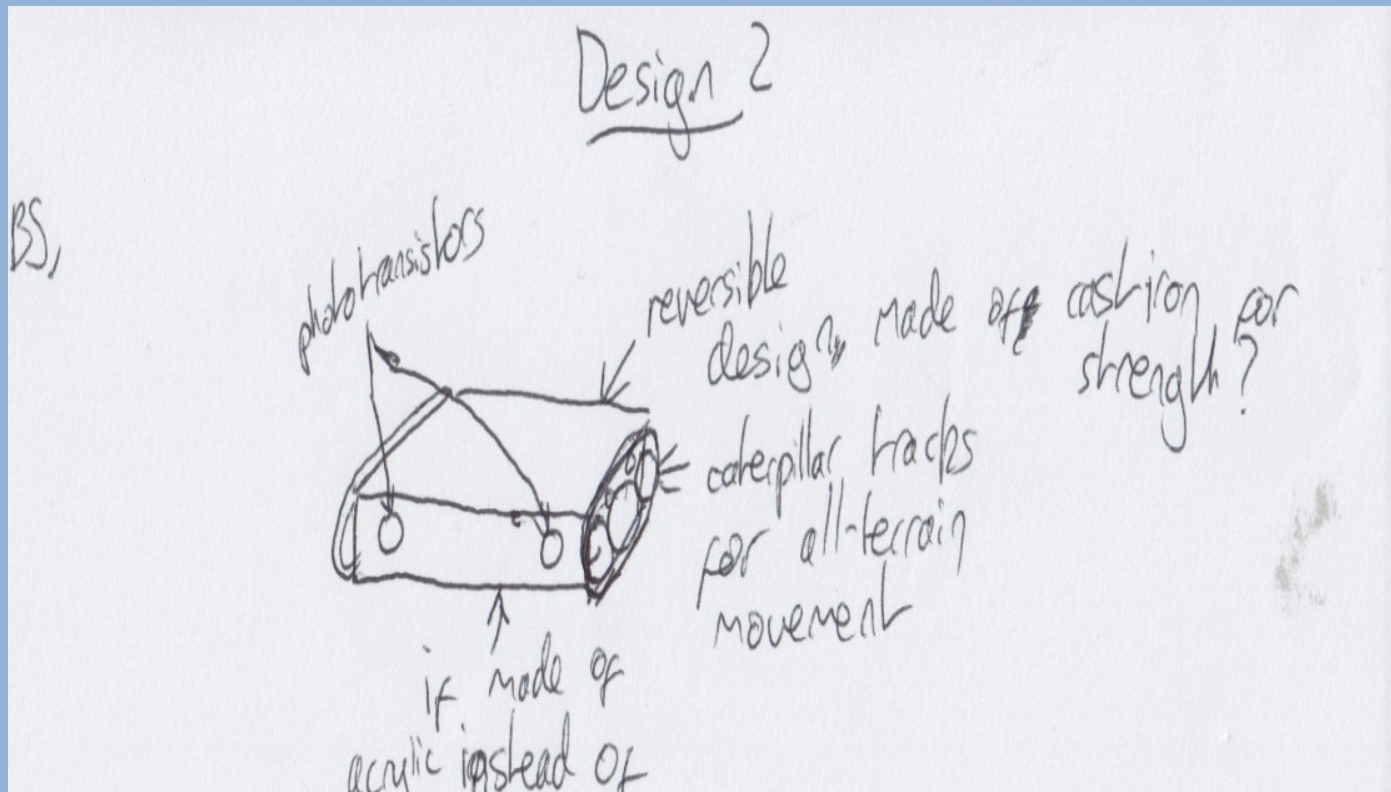


The Genie C08 has only two inputs (one analogue) and three outputs. This is fine for the circuit I have made, but if I was to add reversing as an option, I would need a larger chip.



DESIGN Circuit 2

Sketch

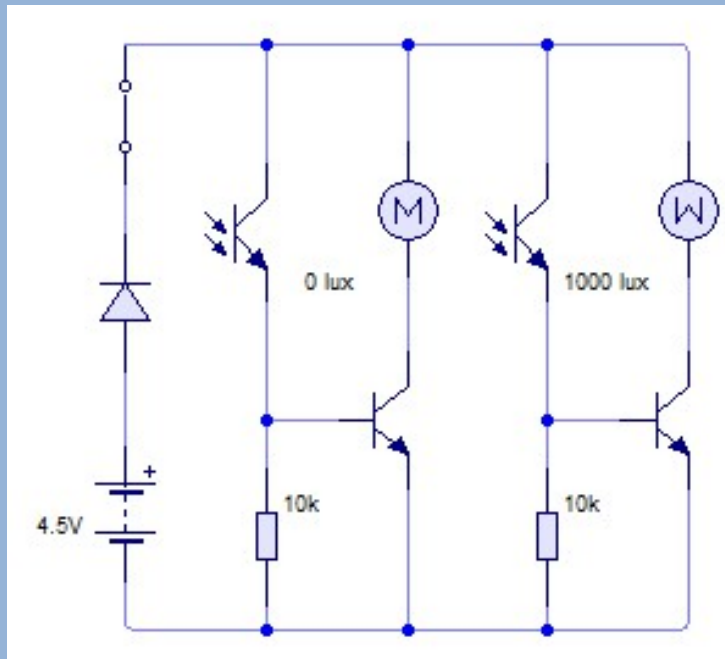


DESIGN Circuit 2

Overview & Details

Size	6
Ergonomics	n/a
Aesthetics	5
Power	9
Components	6
Programming	n/a
Materials	n/a
Product Testing	n/a
Maintenance	5
Product Cost	9
Product Life Span	9
Unattended Mode	4

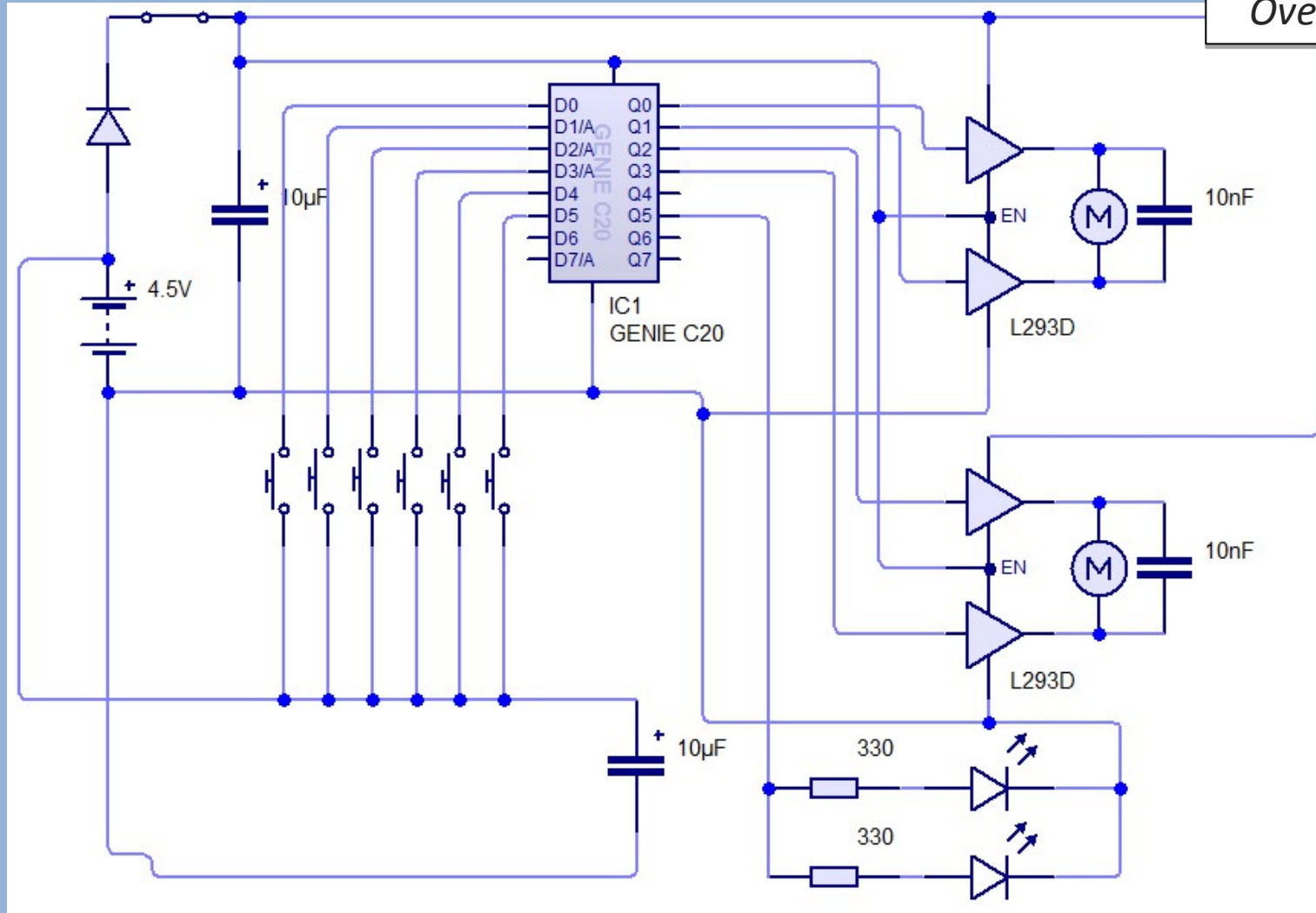
This second circuit uses two photo-transistors to detect light and power two motors. For example, if the leftmost photo-transistor detected a lot of light and the right one didn't, right motor would stop and the car would turn to the right to avoid the block. The advantage of this one is that it is incredibly simple and potentially powerful; the disadvantage is that it is limited in its capabilities and the most important components are unreliable.



This is an incredibly simple circuit, with no need for software and hardly any components. However, it is very limited in what it can do, and photo-transistors are apparently difficult to get working. However, I could keep in mind the general theory behind this circuit when I come to implement the unattended mode into the finished product, as this system can be left do its own devices very easily, and it's not difficult to put in.

DESIGN Circuit 3

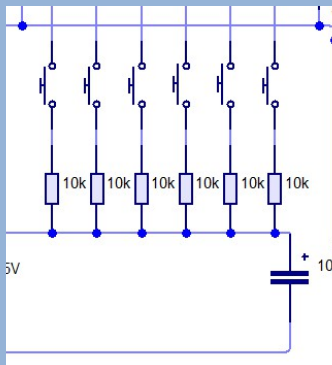
Overview



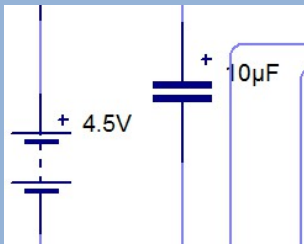
DESIGN Circuit 3

Details

This final circuit design lays out the basic components that I will need for a separate remote control-controlled setup, from the motors to the C20 PIC microcontroller. The circuit simulates in *Circuit Wizard* without exploding, which is a good first step. The advantage of this circuit is that it can perform every task that I need it to, the disadvantage is that it is a more expensive and potentially more complex, and therefore harder to produce and maintain. Before I make my decision about which circuit to do, I feel this one needs a bit more development.

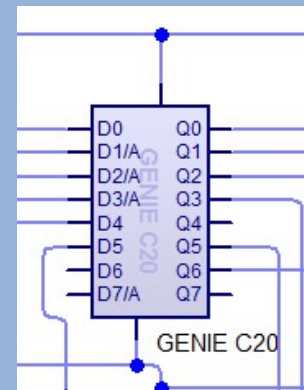


The collection of PTM input switches simulate the remote control device for now, although I will try to make a proper remote control circuit in a later revision.

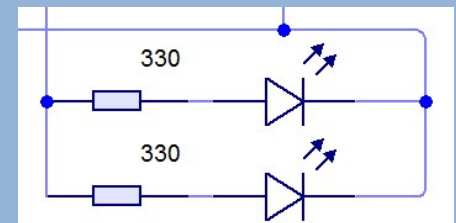


3 1.5V AA batteries will hold charge for far longer than a single 9V PP3 battery (see Research – Investigation)

The C20 PIC microcontroller controls the IO of the project. The C20 comes with 7 inputs, including 4 that can receive analogue data, and 7 outputs. It must also be wired up to both the voltage rail (although it can only take 5V before exploding) and the 0V rail. I considered using a C18, but it lacked an analogue input that I needed.

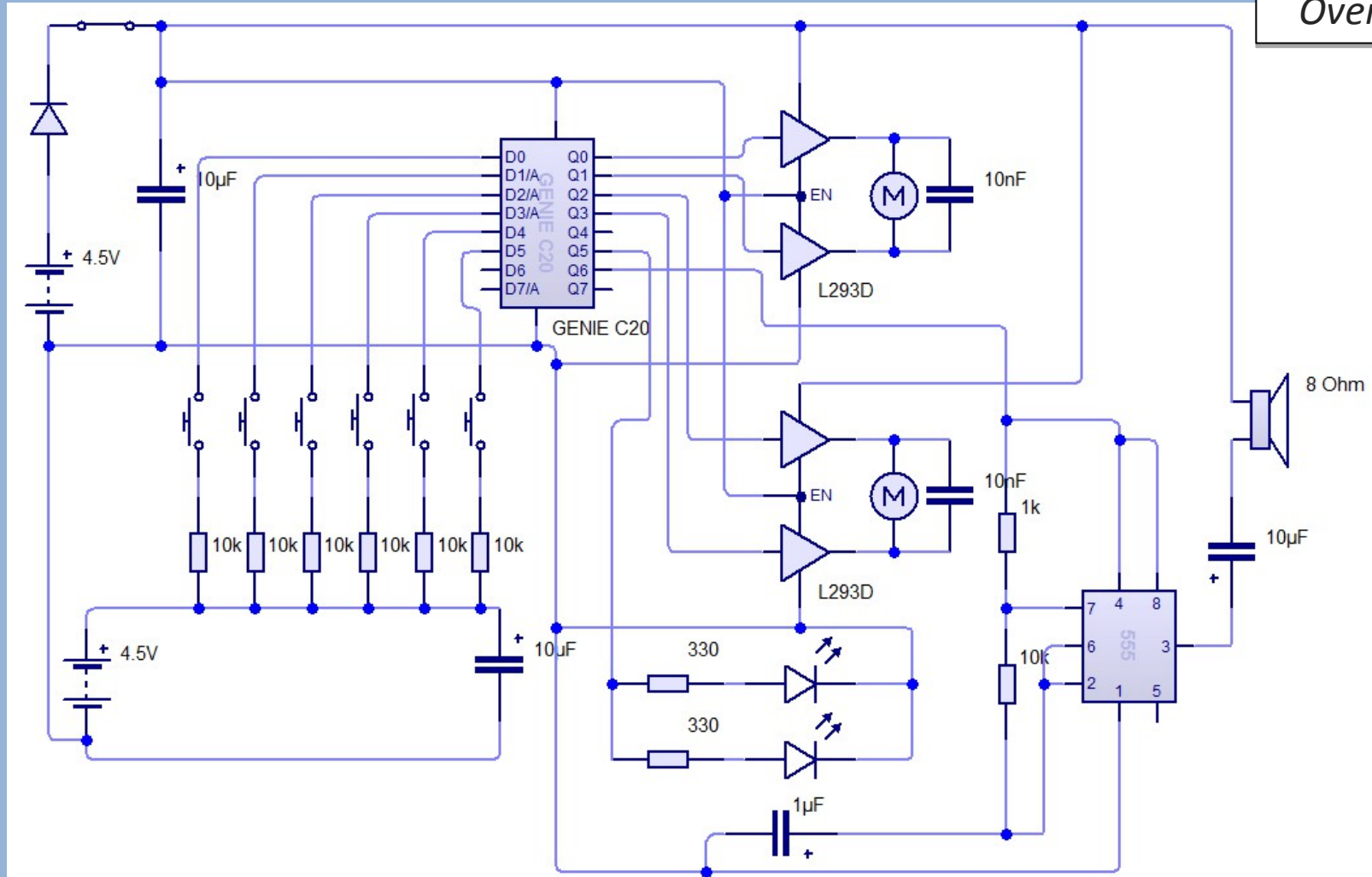


This part of the circuit represents the headlights, two LEDs that don't need to ever be on independently of each and so can be placed on a parallel circuit.



DESIGN Circuit 3 rev. 1

Overview



DESIGN Circuit 3 rev. 1

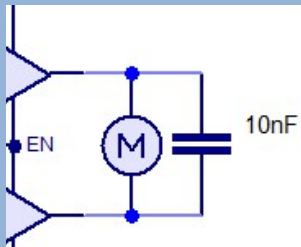
Details

Size	n/a
Ergonomics	n/a
Aesthetics	n/a
Power	7
Components	8
Programming	7
Materials	n/a
Product Testing	n/a
Maintenance	6
Product Cost	7
Product Life Span	8
Unattended Mode	10

Rev. 1 of my circuit adds the loudspeaker that I plan to use for a horn and/or fake engine noise when the car is running. The addition also meant adding a 555 timer and a $10\mu\text{F}$ electrolytic capacitor. The addition of the 555 timer also means adding a $10\text{k}\Omega$ and a $1\text{k}\Omega$ resistor.

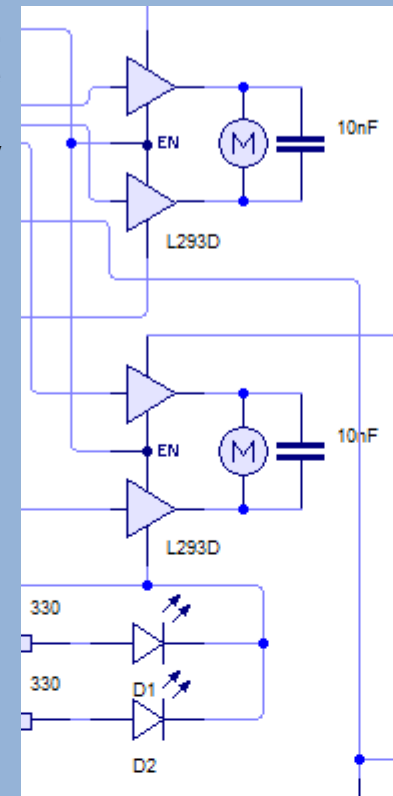
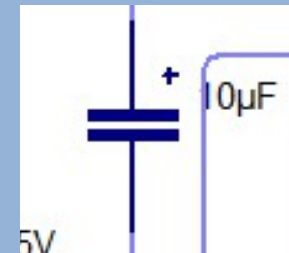
The capacitors alongside the motors are there to give the motors the inertia needed to begin spinning. Flyback diodes to save the motors from being destroyed by feedback when they are deactivated are included in the h-bridges.

I decided to use h-bridges for the two motors, which allows them to change direction easily and without adding some DPDT switches into an already cluttered circuit design.



The capacitor at the beginning of the circuit serves as a smoothing capacitor by keeping the circuit going through minute losses of power.

This part of the circuit consists of a 555 timer and a loudspeaker and represents the car's horn. I had some trouble with this initially, as the loudspeaker would explode every time the circuit was simulated, but I believe this to be a glitch in the program and not what will happen in real life.



DESIGN Circuits Analysis

Circuit 1

Size	6
Ergonomics	8
Aesthetics	7
Power	5
Components	7
Programming	9
Materials	n/a
Product Testing	n/a
Maintenance	8
Product Cost	8
Product Life Span	7
Unattended Mode	6

Circuit 2

Size	6
Ergonomics	n/a
Aesthetics	5
Power	9
Components	6
Programming	n/a
Materials	n/a
Product Testing	n/a
Maintenance	5
Product Cost	9
Product Life Span	9
Unattended Mode	4

Circuit 3 rev. 2

Size	n/a
Ergonomics	n/a
Aesthetics	n/a
Power	7
Components	8
Programming	7
Materials	n/a
Product Testing	n/a
Maintenance	6
Product Cost	7
Product Life Span	8
Unattended Mode	10

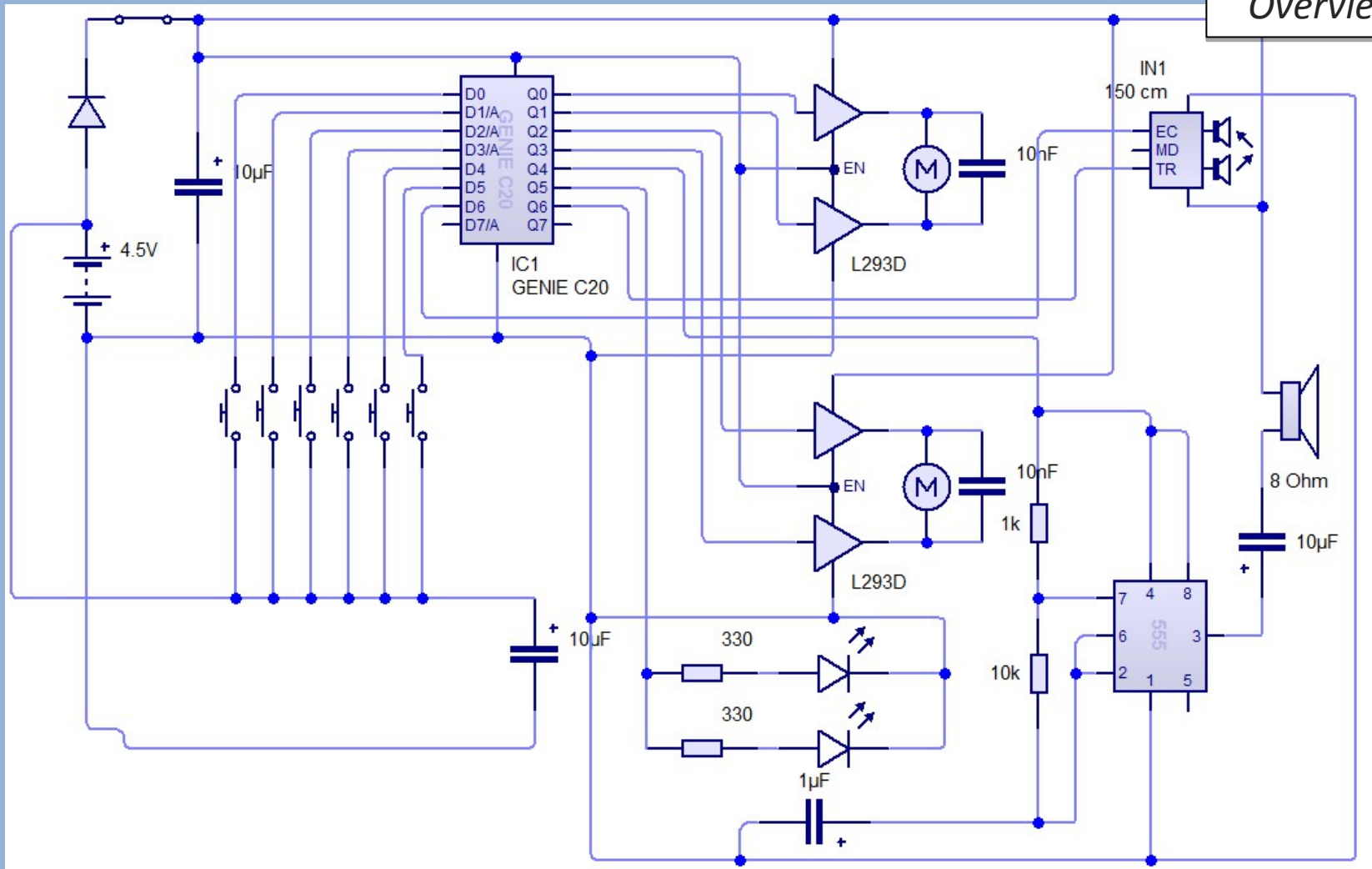
Circuit 1 has the advantage of the unique steering method via potentiometer and the ease of use, but it is likely the most expensive of the choices with the two relays. The program is simple, however, and it doesn't seem like it would be too hard to build. For example, there's only two inputs from the remote control to the car.

Circuit 2 is easily the simplest of the circuits, having no program or human inputs and running solely on analogue data. The biggest issue, however, is that I have found phototransistors to be unreliable at best and it would take some effort to set them up in a working configuration. Whilst its simplicity is the biggest advantage of Circuit 2, I also feel that it may count against me in an AS project.

Circuit 3 rev. 2, however, seems to be on the right track. I developed it beyond its original idea (e.g. added the ultrasonic distance sensor necessary for the unattended mode) and feel that it is complex enough for an AS project, cheap enough to be reasonable and fulfils the brief. For these reasons, I shall develop this circuit further and use it in my final project.

DESIGN Circuit 3 rev. 2

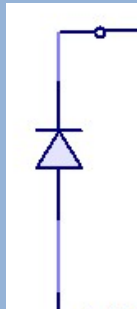
Overview



DESIGN Circuit 3 rev. 2

Details

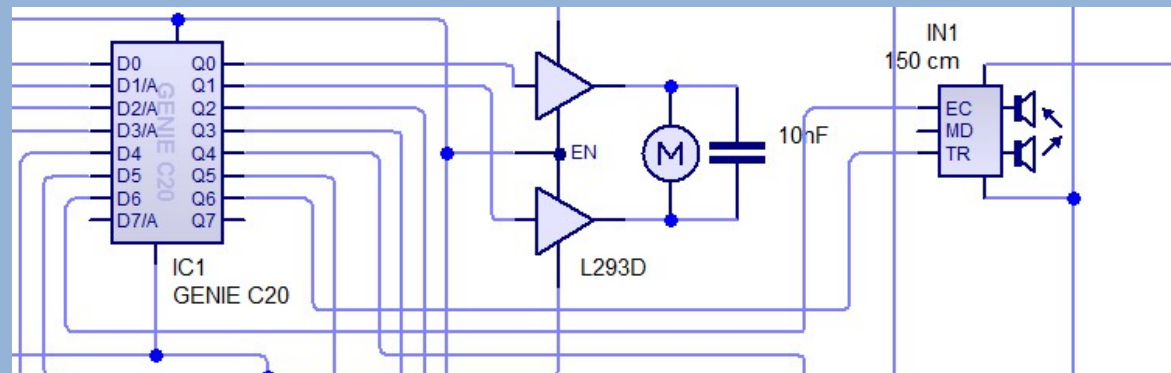
Rev. 2 of my circuit adds the SRF05 ultrasonic distance sensor that I plan to use for the unattended mode. However, the SRF05 is a very expensive component and the price of my project has just shot up.



This diode, placed above the battery, means that if the batteries are put in the wrong way round, the current will be blocked and the components won't get broken. However, it takes up 0.7V.



This SPST switch simulates the rocker or slide switch that I will use as the master on/off switch.



The SRF05 ultrasonic distance sensor is a unique component in that it connects both to an input and an output pin on the C20. The chip uses the output pin to get the SRF05 to give off a sonic pulse. If the pulse is bounced back by an object, it arrives and is fed into the C20 as an input. The code to the right demonstrates how this is programmed.

```
start
do
  ' Send a sonic pulse and measure the echo. A value of
  ' 255 is returned when there is no echo.
  ultra A, 1, 0

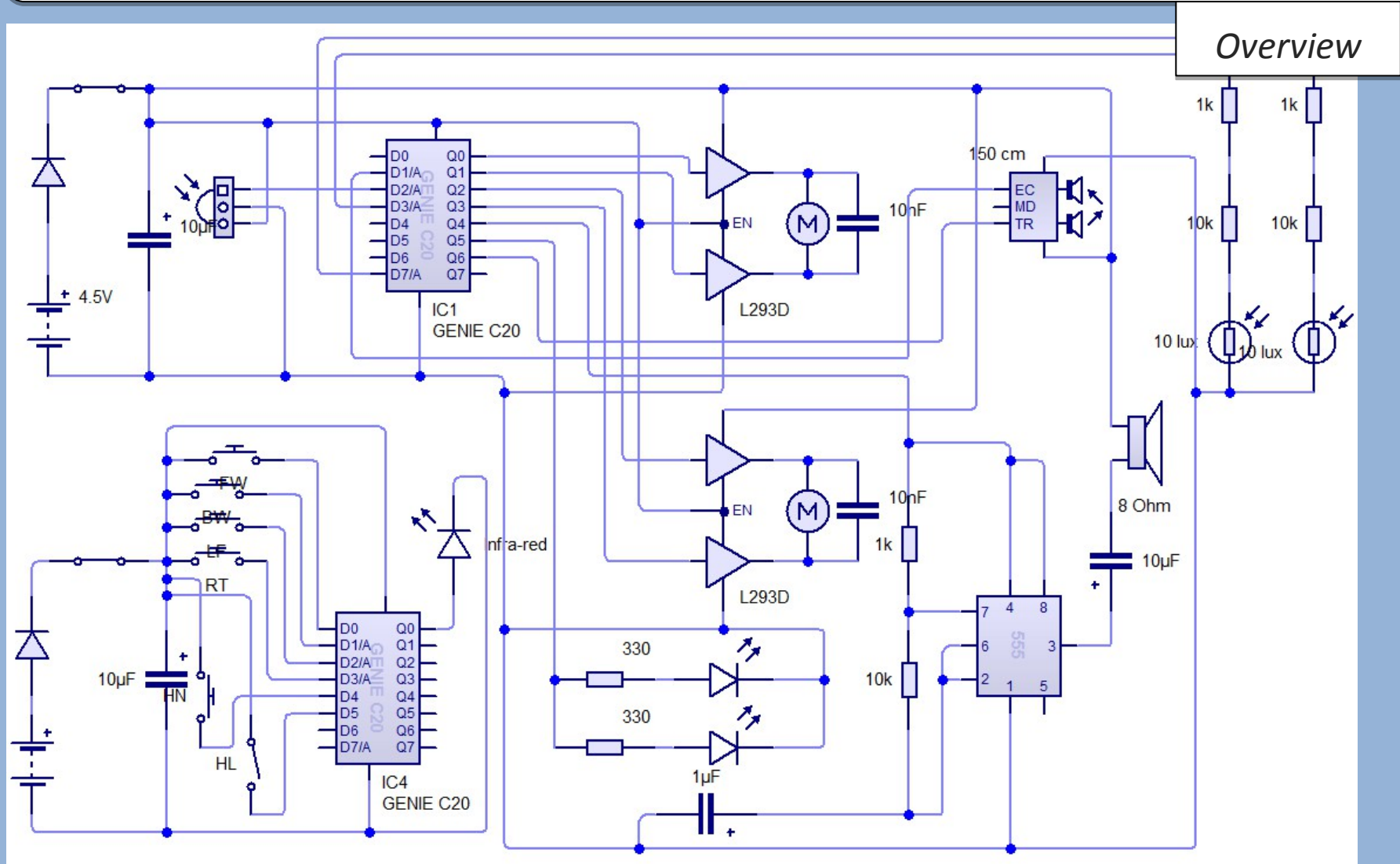
  if A = 255 then
    ' No object found within range of the sensor.
    lcd 0, "[line 1]Ultrasonic Range[line 2]Nothing Detected"

  else
    ' Display the distance, in cm, on a connected liquid
    ' crystal display (LCD).
    lcd 0, "[line 1]Ultrasonic Range[line 2]Distance: [A] cm "

  end if

  ' Wait for message to be displayed.
  wait 2
loop
```

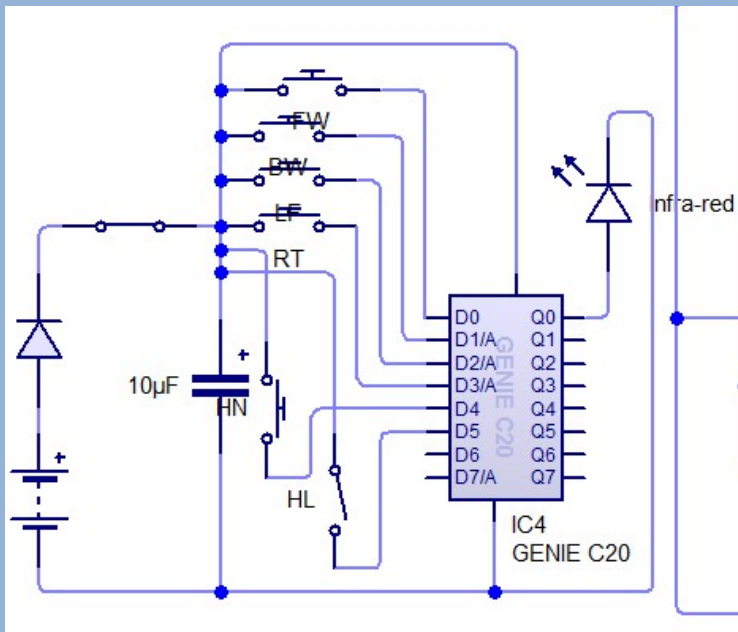
DESIGN Circuit 3 rev. 3



DESIGN Circuit 3 rev. 3

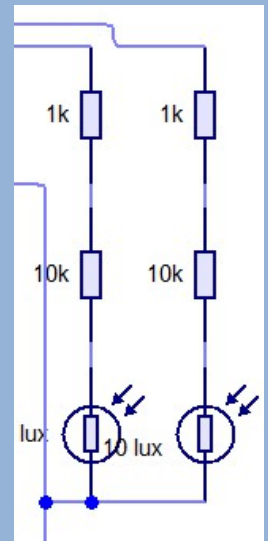
Details

Rev. 3, the final revision of my circuit, adds the separate infra-red remote control circuit rather than the simulation with buttons I was using before, as well as the LDRs for a more advanced unattended mode. The price is getting climbing rapidly now, although the school can get components for cheaper than I have seen.

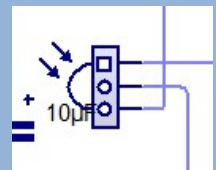


This is the temporary hideous remote control circuit. The labels are obscured, there's no pull-down resistors and the remote control should really be its own file. However, the general gist is there.

This section of the circuit contains the two LDRs used to activate the headlights automatically in dark surroundings. As they are inputs, they probably shouldn't be over on the right of the circuit diagram. I will sort this out later.

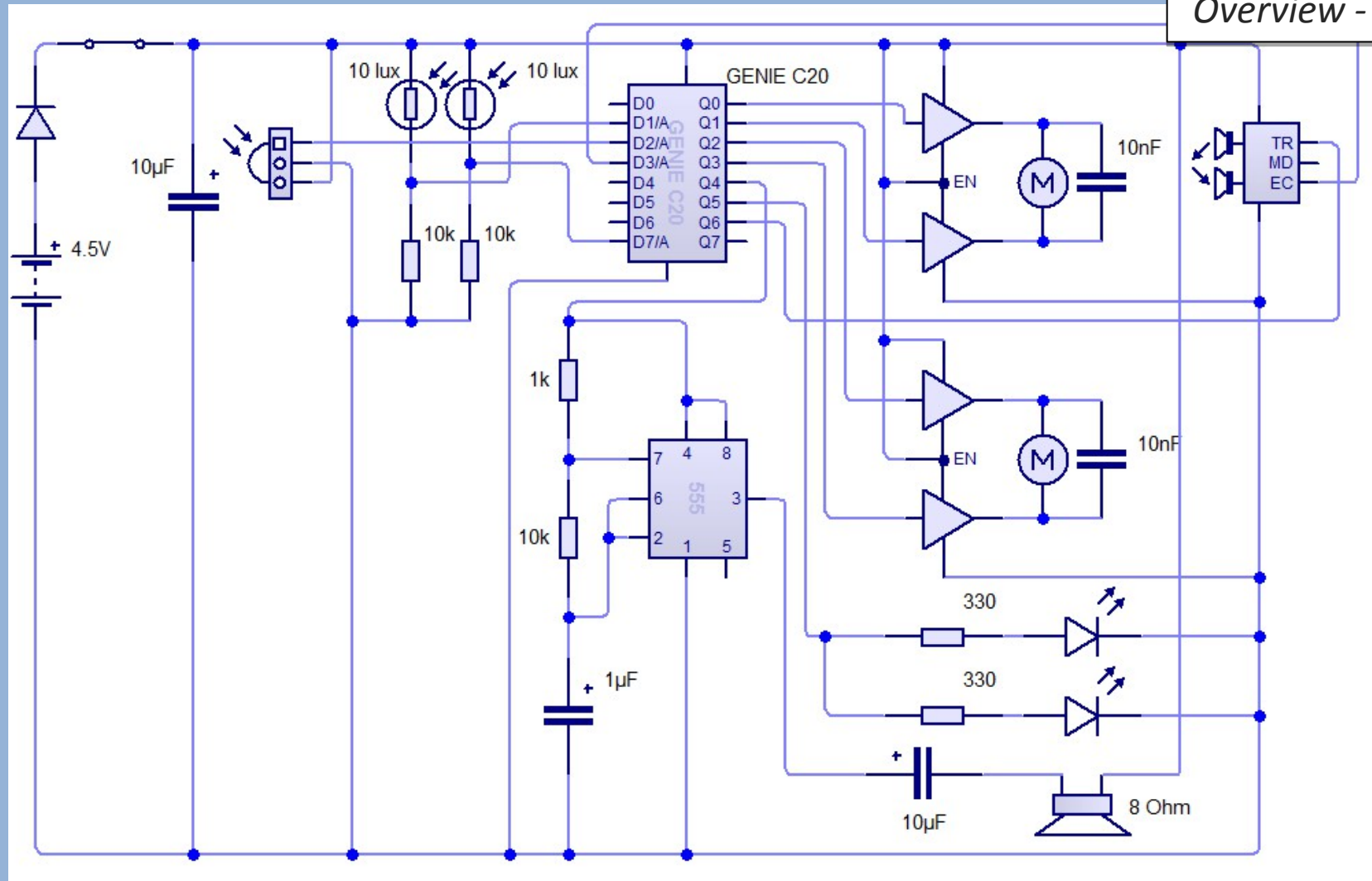


This is an IR receiver, and receives the signals from the IR transmitter in the remote control circuit.



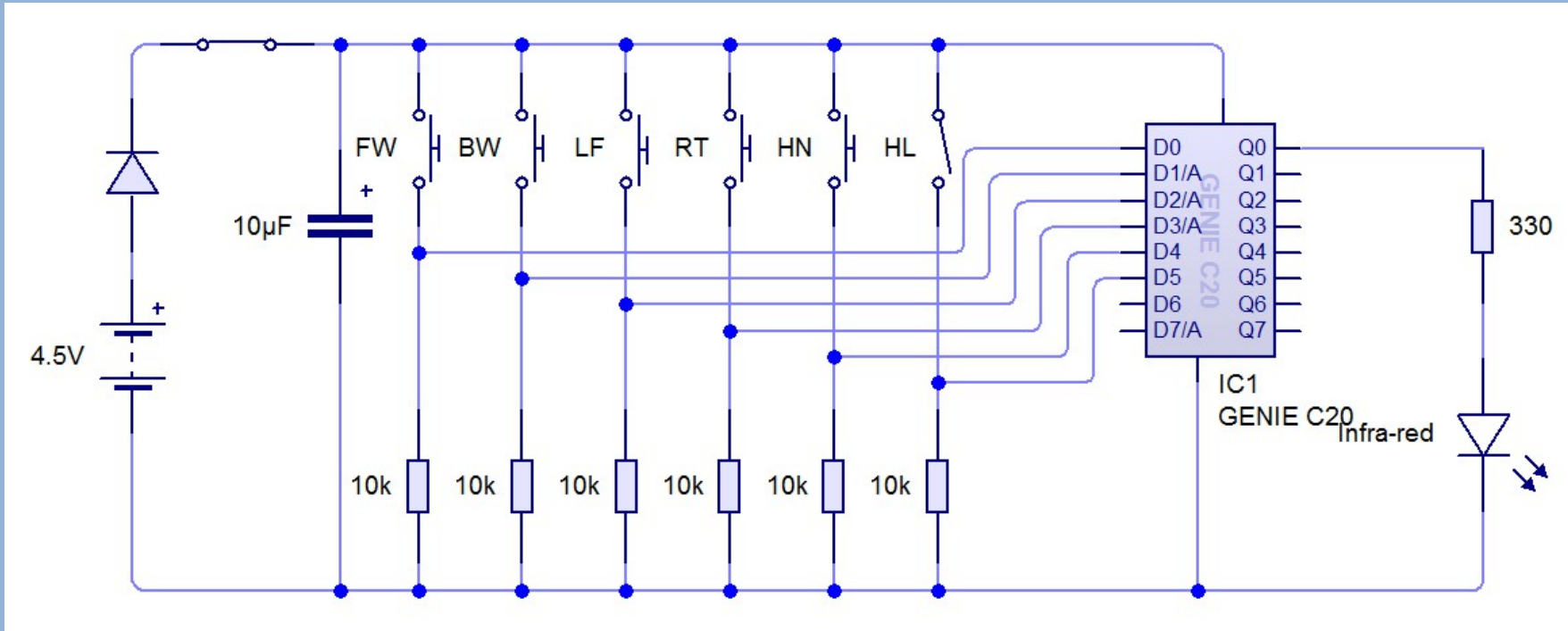
DESIGN Circuit 3 rev. 4

Overview - Car



DESIGN Circuit 3 rev. 4

Overview – Remote Control



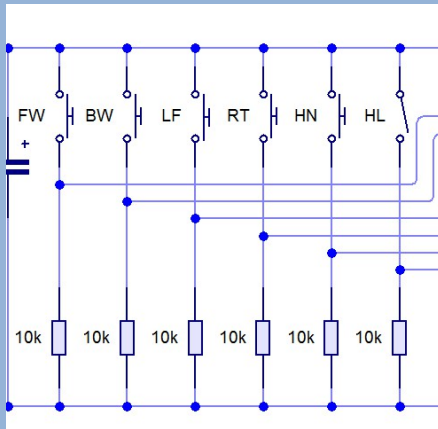
DESIGN Circuit 3 rev. 4

Sketch

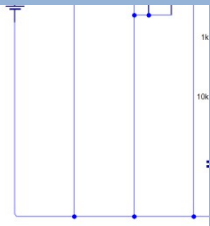
DESIGN Circuit 3 rev. 4

Details

This final revision rearranges the circuit diagram to be easily readable, as well as relegating the remote control circuit to its own file.



As a result of the rearranging, the labels explaining what each of the remote control's buttons do are now visible. I also added the 10kΩ pull-down resistors that are required to keep the PIC inputs from floating.

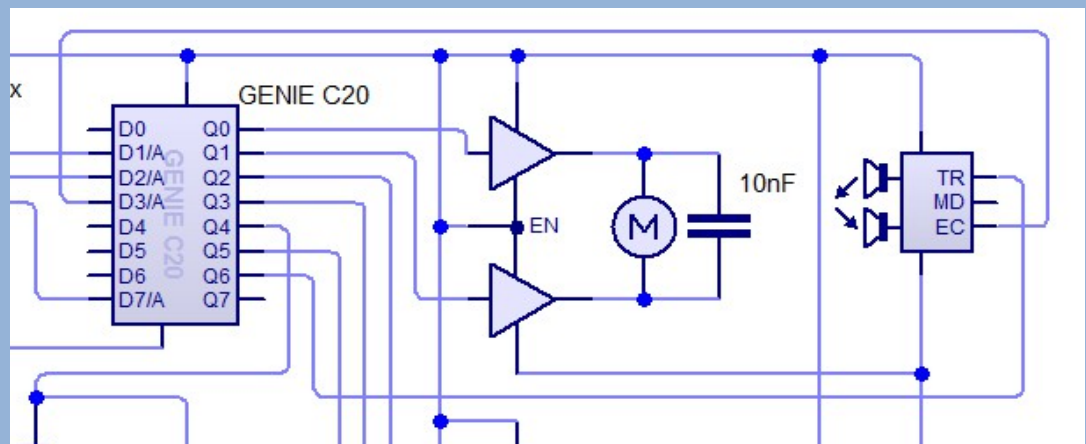


I dislike this large blank area of the circuit, but it was necessary in order to make the power and 0V rails along the top and the bottom.



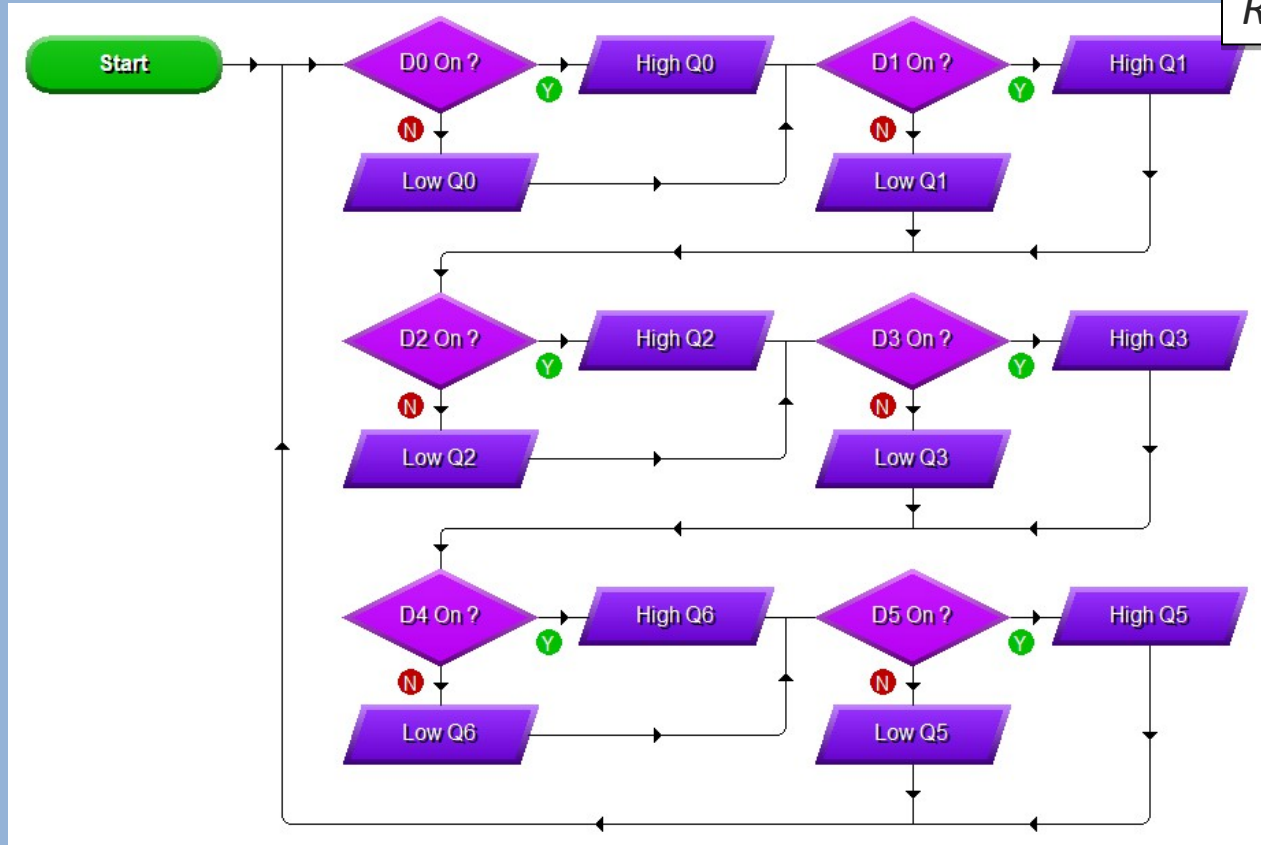
The inspiration for the case sketch is the robot Tornado from the TV series *Robot Wars*. The design is simple and boxy, and allowed Tornado to be flipped and remain operational. I will probably forgo the spikes and flywheel, however.

The problem with using the SRF05 ultrasonic distance sensor in my circuit is that, as a result of it having both an input and an output, it is difficult to place in a sensible place in the circuit.



DESIGN Software

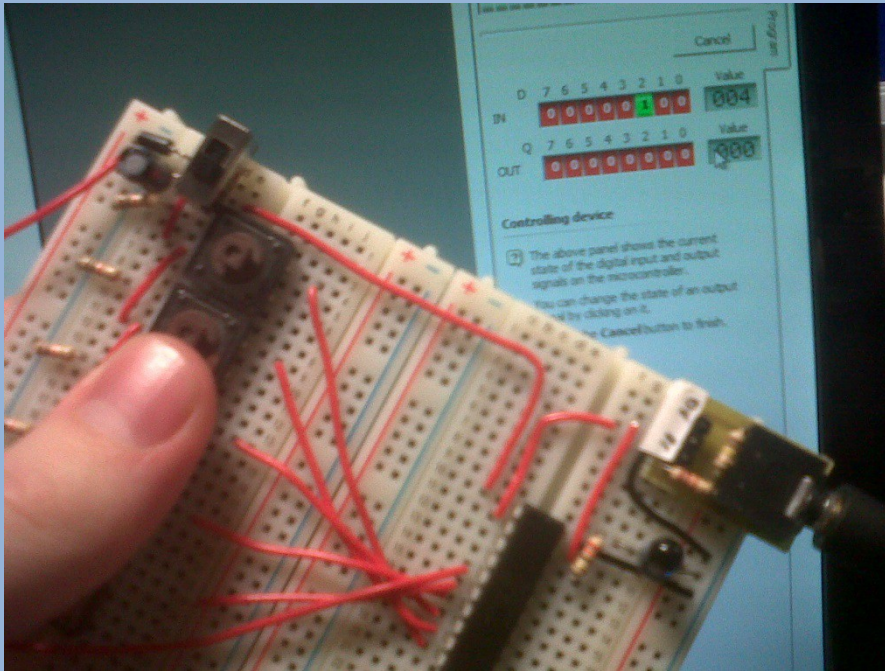
Remote Control



Using *Circuit Wizard*'s built-in flowchart function, I created a quick flowchart to show how the program logic would eventually work. The program deals with inputs (buttons in the circuit diagrams) being queried and, if on, their respective outputs being activated in turn. The flowchart is also recursive, so that it will repeat for as long as the circuit has power.

DESIGN Prototyping

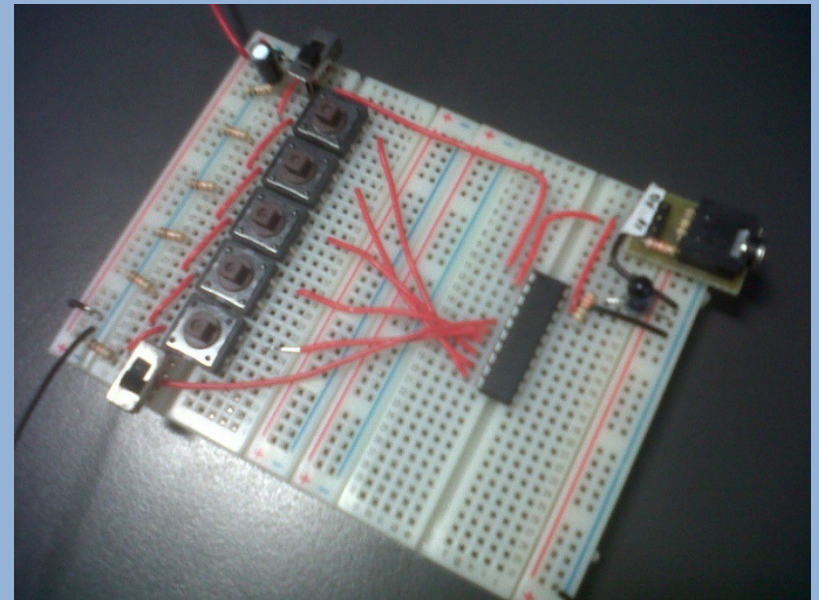
Remote Control



Once it was finished, however, it became clear that sending signals between the remote control and the car would be a hassle, as the IR receivers we have in stock aren't sensitive enough to receive anything from a range of more than 10cm. I decided to remove the remote control aspect of the project, leaving just an autonomous robot that can drive about on its own once as long as it receives power. However, this breadboard shows that the remote control works in theory and would be easy to implement at a later date.

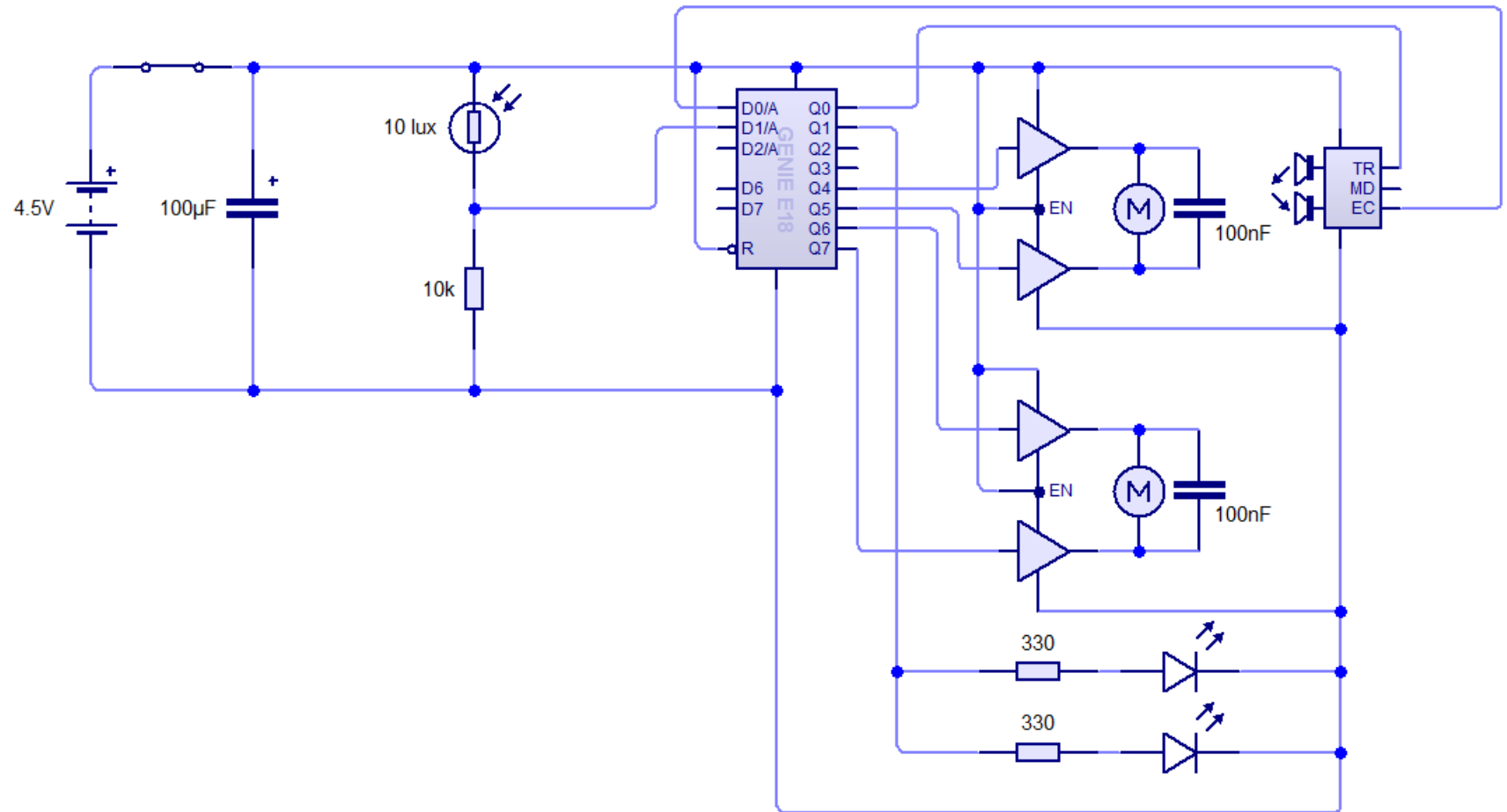
With the circuit and program finished, I was ready to breadboard an prototype of the circuit. Breadboarding has an advantage over circuit simulation in that it shows whether the circuit will work when introduced to the real world. A disadvantage, however, is that it takes time to do, but I have plenty so this isn't a problem.

When I finished the prototype I plugged it into the PC and tested the inputs. As shown to the left, when I pressed a button, the corresponding input lit up in *Circuit Wizard*. This shows that the circuit definitely works.



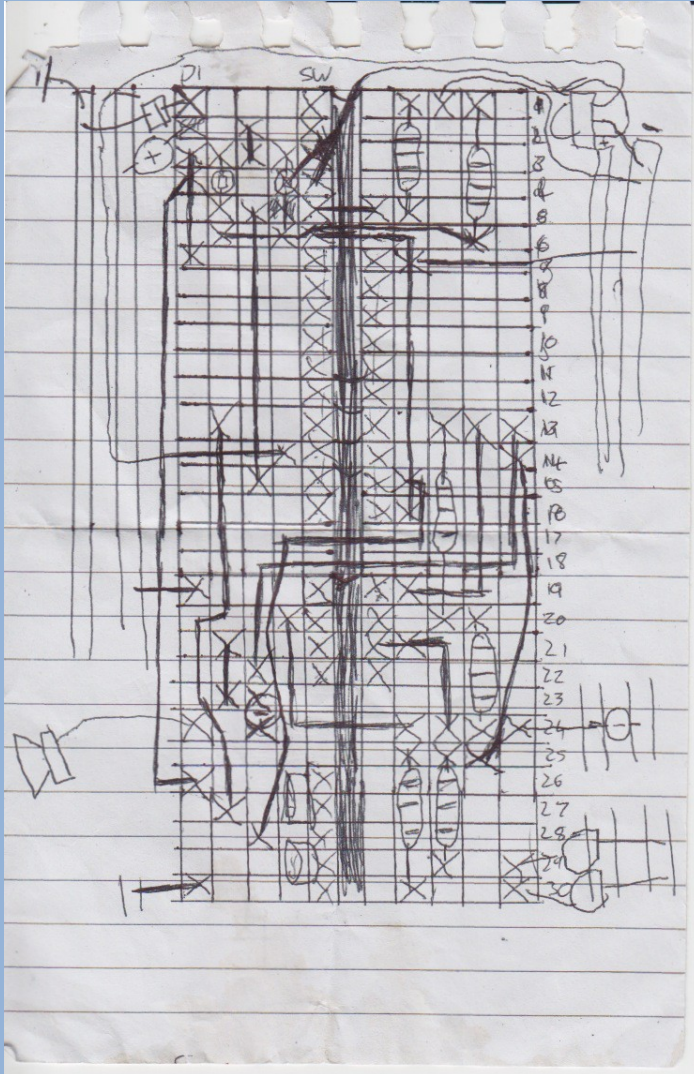
DESIGN Circuit 3 rev. 5

Overview



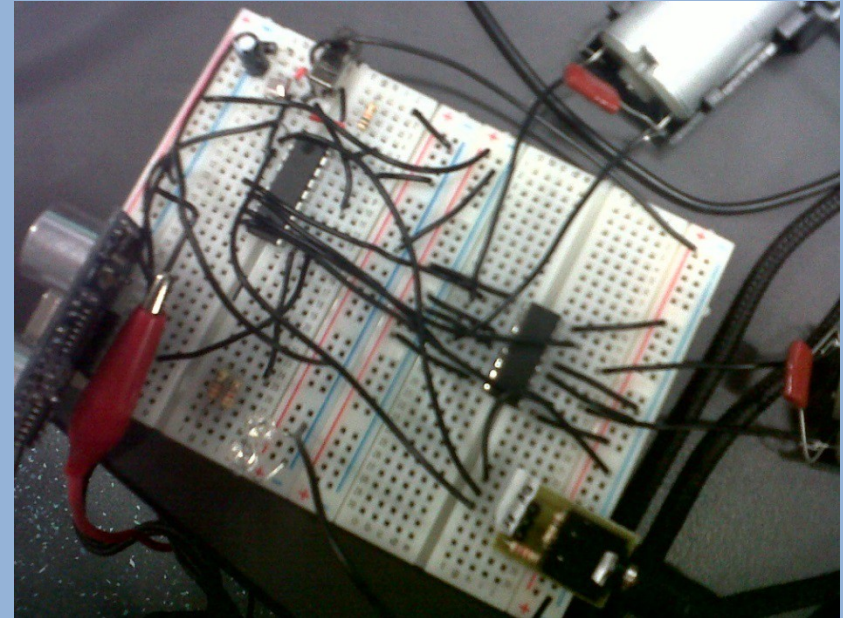
DESIGN Prototyping

Car



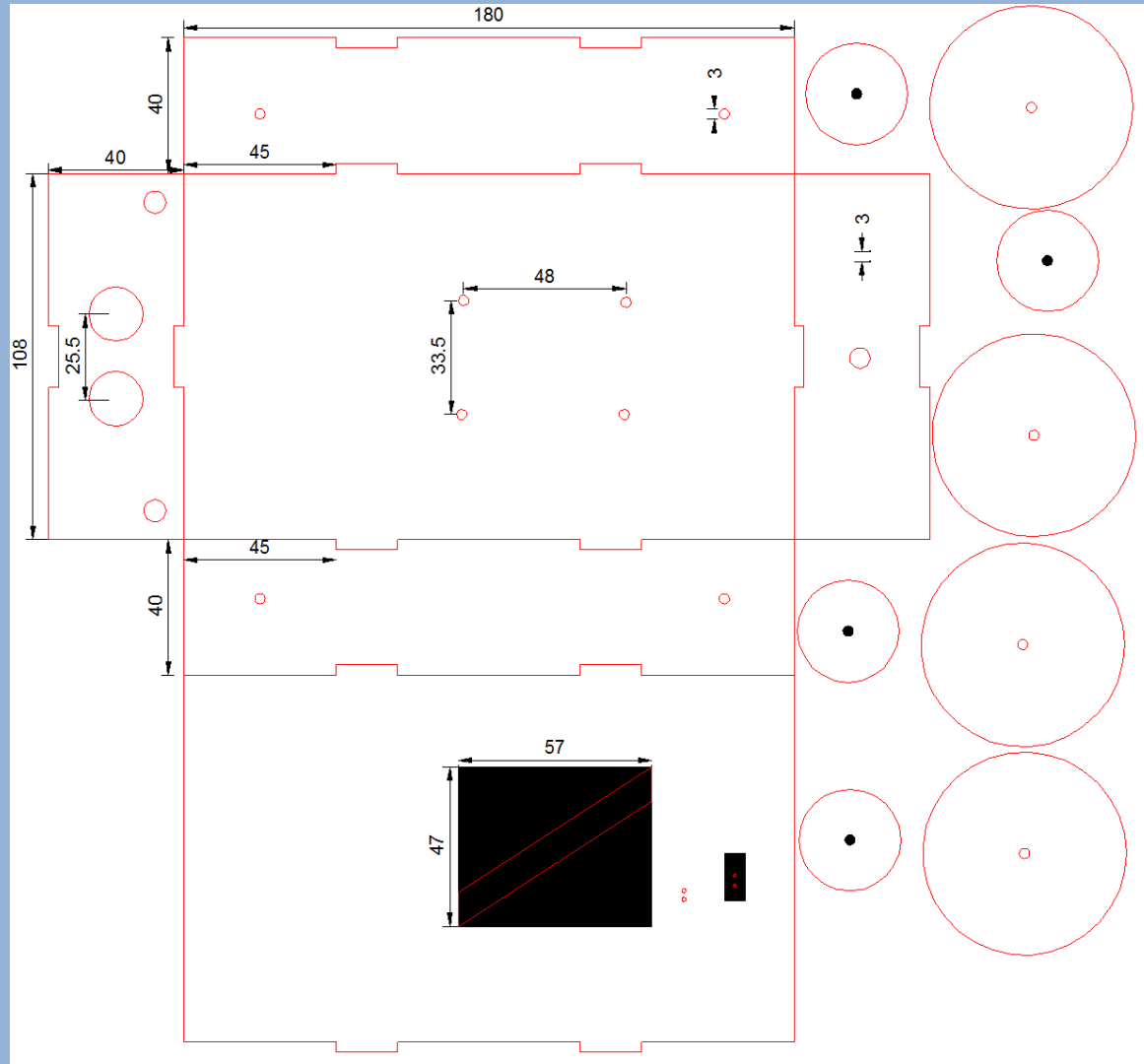
With the car circuit slightly redesigned I set about breadboarding it. I started with what you can see to the right; a sketch showing the wiring layout of the eventual breadboard. I made this as I had no access to wires and wanted to get on with things.

The finished breadboard (below) is certainly no looker, but it all functions correctly; the LEDs turn on when the LDR is covered and the motors spin correctly depending on the distance of a hand from the SRF05. With the circuit design vindicated, I was ready to move on to producing the PCB.



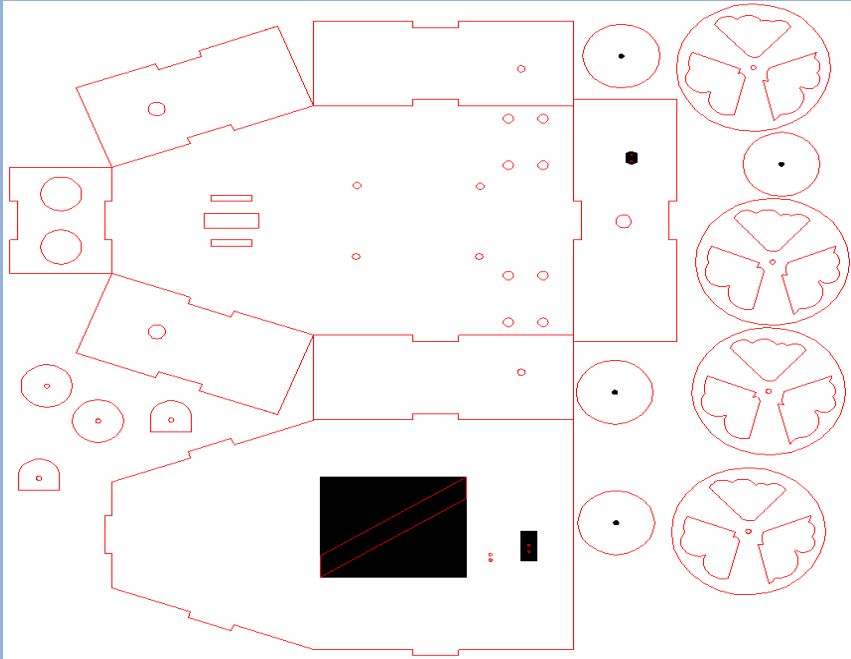
DESIGN Housing

Overview



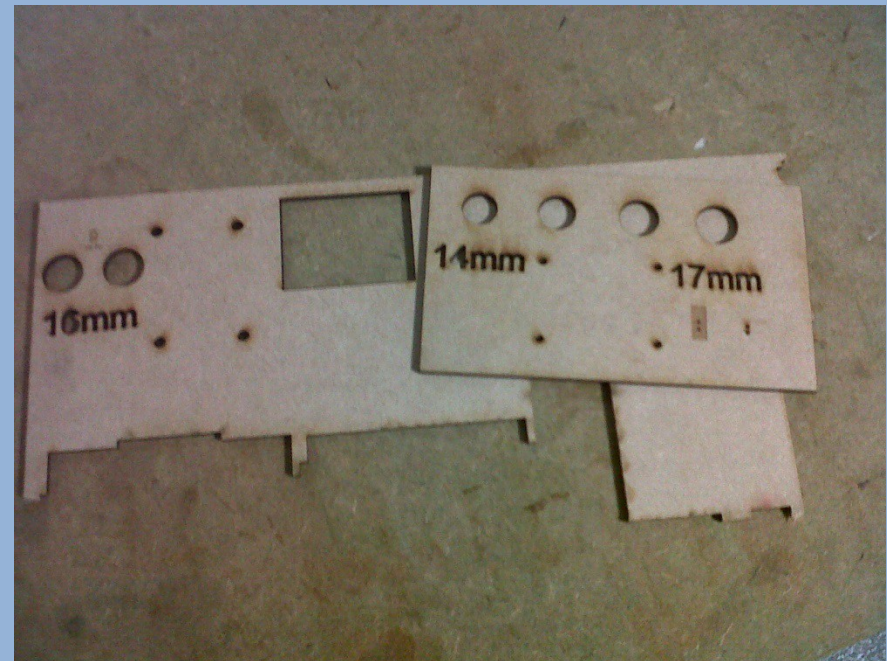
DESIGN Housing

Details

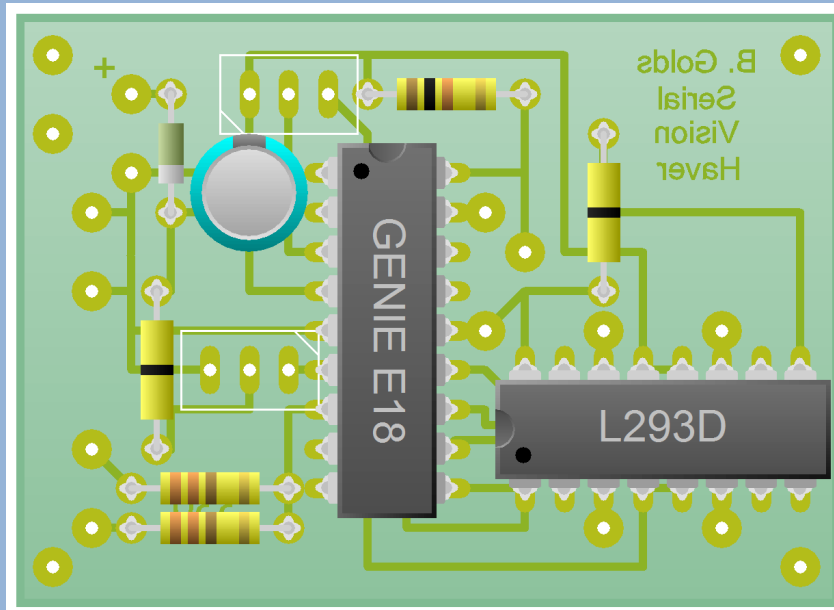


After this, I drew on the rest of the drawing. The holes offset from the centre of the second-from-top rectangle are for mounting the PCB onto, and these were the most finicky to get accurate, but obviously important. To the left is the front panel, with holes for the headlight LEDs and SRF05, and to the right is the back, with holes for the LDR and download socket. Above and below have holes for front and rear wheels (the smaller wheels with black circles in the centres are hubcaps) and the bottom rectangle has a sunken area for the battery pack (with a cut-out diagonally across for the wires on the back to nestle in) and for the power switch.

The first step with designing my housing was to make sure the hole sizes and alignments were all correct. My design contains a number of holes, and if any were incorrect none of the others would be able to fit. To ensure my holes were correct, I laser cut a number of hole guides, which were quicker to cut and used less material than printing my entire design each time, and allowed me to tweak the various holes little by little until everything fit together.

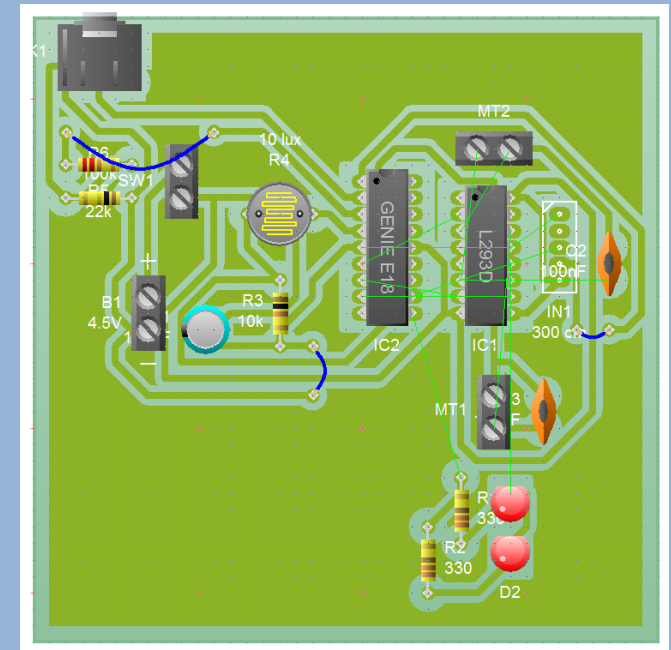


DESIGN PCB



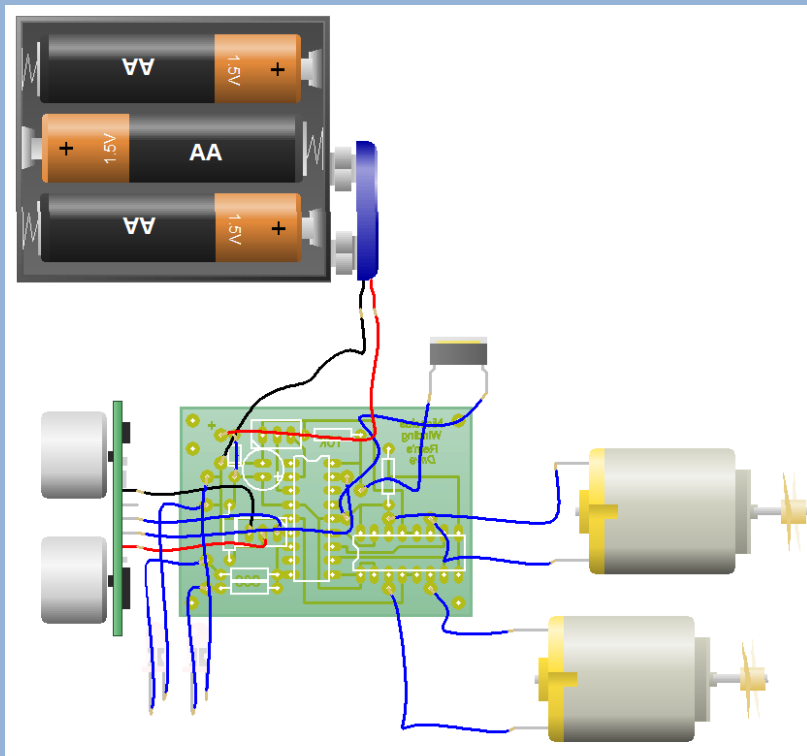
After this I put on the largest components; the ICs. However, my first design ended up using a lot of link wires due to the difficulty of wiring up the L293D half-H bridge driver (it requires all four corner pins to be connected to power, and all four middle pins to be connected to 0V). In the second attempt, I disabled the snap to grid function and aligned the tracks coming out of the top (or the left, in the case of the above image) of the chip just close enough to all fit through but just far enough apart to etch properly. After this I added the resistors, pads, etc. and was left with a PCB that: measured only 53x38mm, almost half the size of the automatically-generated PCB; contains no link wires, although two 0Ω resistors were unavoidable; and has only one unavoidable area of wastage in the top right, which I filled with a label with my name for style.

I made the mistake of using *Circuit Wizard's* built-in automatic PCB generator at first, which produced what you see below; it measures 95x95mm, has three link wires and is about a 50/50 split of components to wasted space. Obviously, something had to be done. Starting from scratch and assembling the PCB by hand, I started with the initial components of the circuit design (e.g. the battery, the diode, the capacitor and the download socket) as these form the crux of much of the subsequent placement of components.



DESIGN Testing

Using *Circuit Wizard*, I added the external components that the final product would have and simulated the circuit. This doesn't give me a completely accurate view of whether it will work or not, as *Circuit Wizard* is nothing if not temperamental, but it should demonstrate whether it all works in theory, and the current view setting can give a broader insight into how current is (or should be) flowing than using a multimeter on each length of track.



The **drive test** involves turning the circuit on and ensuring that the motors both turn to drive the car forward.

Expected Result	Actual Result
The motors turn simultaneously.	The motors turned simultaneously.

PASS

The **H-bridge test** involves turning the circuit on and testing the voltage pins on the L293D half H-bridge driver.

Expected Result	Actual Result
The four pins receive voltage.	The pins are receiving voltage.

PASS

The **download test** involves attaching a download cable and pushing a program from the computer onto the PIC.

Expected Result	Actual Result
The program is pushed successfully.	The program pushed successfully.

PASS

The **ultrasonic test** involves changing the value coming in from the SRF05 ultrasonic distance sensor and seeing what the result, if any, is on the circuit.

Expected Result	Actual Result
One wheel stops, turning the car, for a random amount of time.	The car turned for a random amount of time.

PASS

The **headlights test** involves changing the value coming in from the LDR and seeing what the result, if any, is on the circuit.

Expected Result	Actual Result
The two LEDs turn on when the LDR value goes below a certain threshold.	I lowered the LDR value and the LEDs turned on.

PASS

DESIGN Making Plan

Step	Process	Amplification	H&S/QA	Time
1	Etching PCB	Print the artwork, place it over the PCB material in the lightbox, place in developer tank, wash, place in etch tank, wash, trim PCB, drill holes, strip PCB, wash, dry.	QA: Check tank temperature, ensure no breaks in track with both eyes and multimeter, ensure fresh drill bits. H&S: Wear apron, gloves and goggles when etching, goggles when drilling.	1 hour
2	Soldering PCB components	Solder SIL and DIL pin sockets, resistors, diode, capacitor, and external flying components.	QA: Ensure soldering iron is clean, check orientation of components. H&S: Wear apron and goggles, don't leave the soldering iron turned on.	2 hours
3	Laser cutting housing	Send .dtd file to laser cutter computer, print to laser cutter, ensure settings set to appropriate material, place material in laser cutter bed, cut.	QA: Check mirrors and cutting head are correctly aligned/mounted tightly, test finished result with calliper. H&S: Don't look into beam, ensure extractor is turned on, don't breath fumes if using MDF.	2 hours
4	Assembling product	Stick housing together with glue, mount PCB and motors to housing with M3 and M4 screws, place external flying components in their respective holes.	QA: Twist wires together for neatness, use heat shrink tubing on exposed wires. H&S: Don't trip.	2 hours
5	Downloading program	Place E18 and half-H bridge driver chips onto PCB, plug download socket into computer, connect to power supply, open Circuit Wizard, download program.	QA: Test all inputs and outputs with Circuit Wizard. H&S: Don't place finger into power supply socket.	30 minutes

Total time: 7:30h

BUILDING PCB

Etching

With the circuit designed and the program written, it was time to move on to creating the actual PCB. The first thing I had to do was print off the circuit diagram onto transparent paper before placing this over some uncovered PCB material (cut to size with a three-in-one machine) and exposing it to a lightbox for three minutes. After this was done, I took the PCB and submerged it in an etching solution for a few seconds, removing it to see if it was etched. It wasn't, so I dunked it back in, and repeated the process until it was done. At one point I was distracted and it was left in for a bit too long, but it luckily emerged unscathed.

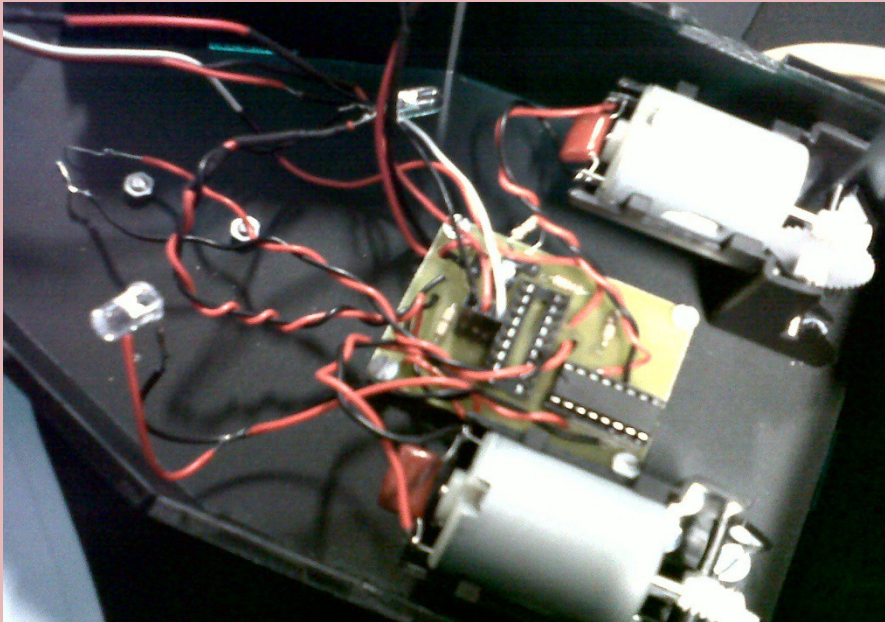


After this, I washed in in water to remove the etching solution and then placed it in a bubble tank for ~15mins. Once this was done, I took it out and drilled the hole, using a 1mm drillbit for the component holes and pads and a 3mm drillbit for the mounting holes in the corners. Finally, I got an etch resist stripper and stripped the etch resist from the board. Looking at the PCB, it seemed it had all etched well with no broken sections of tracks, which was remarkable considering how much I had manhandled it during the etching. With this all done, it was ready to have the components soldered.



BUILDING PCB

Soldering



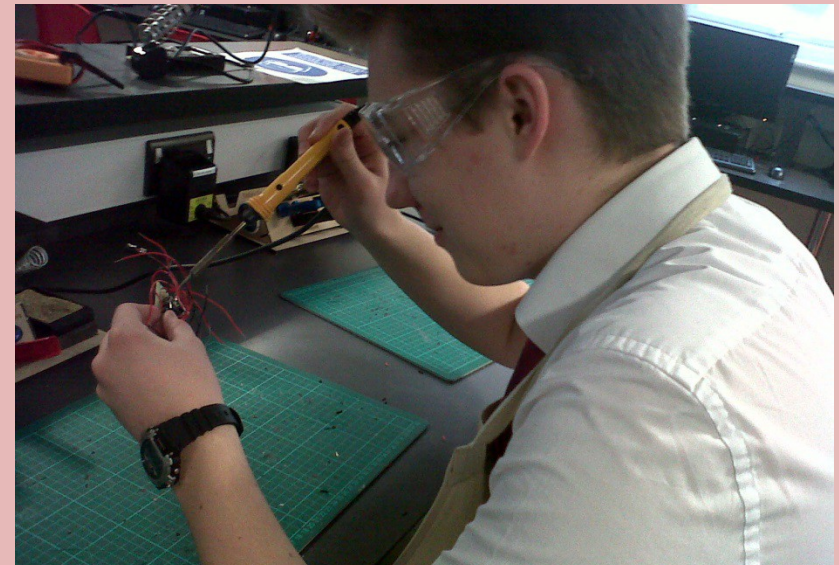
Next up was the smattering of small, two-pin components. These were the resistors, the capacitor and the diode. These easy components went in fine.

Now I had to solder the various wires trailing off to the various externally mounted components (e.g. the motors, the LDR, etc.). These soldered easily, although some of components that I used multi-strand wire for were very fiddly.

Finally, I had to solder on the external components. The LEDs and LDR went on simply enough, but the motors had to be soldered on with a 100nF capacitor going through both of their connectors, but this was simple enough.

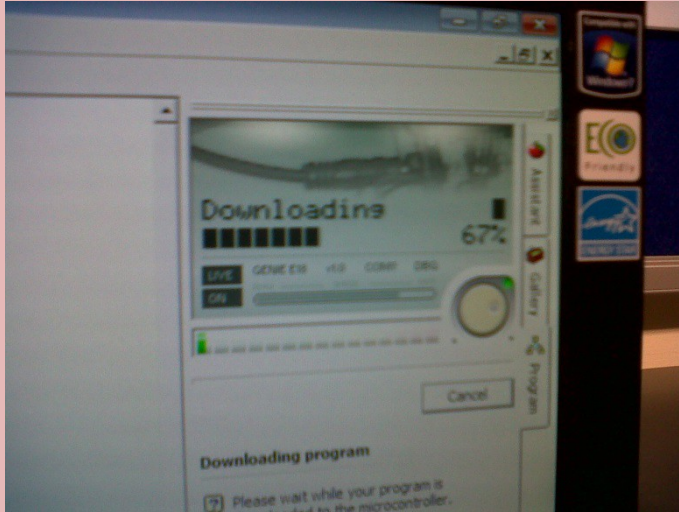
I started by soldering the most fiddly components, fiddly by virtue of the number of pins they have; the ICs. Because ICs cannot handle high temperatures well, they cannot be mounted directly to the PCB. Instead, Dual-In-Line (DIL) sockets must be used. These allow the IC to be mounted at a later, post-soldering date. These went in fine.

I then moved on to the Single-In-Line (SIL) sockets for the download socket and SRF05 ultrasonic sensor. These went in seemingly fine, and I followed them up by soldering the SRF05 and the download socket to them.



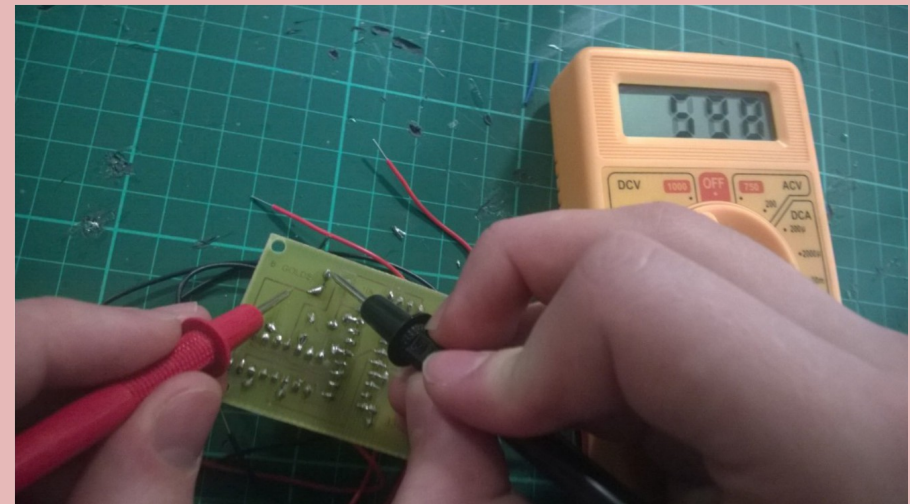
BUILDING PCB

Testing



I then plugged the circuit into the computer and downloaded the program. It didn't go particularly well. On the more minor side of things, due to a peculiar property of a PIC chip and some track brushing against an unused output pin, two output pins had to be turned on to turn on the LEDs. As the extra pin was unused, however, this was not much of an issue. More worrying was that the output to the h-bridge would only work on two of the four outputs. After replacing both the PIC and the h-bridge, as well as just about every other solution under the sun, it still didn't work. Eventually, the cause was found to have been me somehow managing to solder two pins of the DIL socket together between the socket and the PCB, so that the error couldn't be seen without removing the socket. With that out of the way, however, the circuit worked fine.

With all of the components soldered, I continuity tested various points using a multimeter. Most was fine, but one stretch of track was giving off nothing, so I investigated more closely and found a tiny hairline fracture on the track, presumably a result of my clumsy etching. This was promptly fixed by soldering a length of wire, taken from an off-cut of one of the resistors I had used, over the gap. I had to be careful not to have the solder jump across multiple tracks and make a mess, as there was a label very close to the track. I pulled it off, however, and then went over the rest of the PCB with a fine-toothed comb and found no more faults. I then went over it with a multimeter and again found no faults.



BUILDING Software

With the remote control excised, the program (which had also not included any functionality for the car up until this point) had to be redesigned. I decided to forgo the flowchart and program it directly in GenieBASIC, which is much more powerful and faster to code, especially as I have previous experience with VB.NET and no previous experience with Scratch, which the flowchart workflow more closely resembles.

```
start
do
  'This sends an ultrasonic ping out in front of the car.
  ultra B, 0, 0
  'If the distance detected is more than practically in front
  'of the car, the way is clear and the wheels can move.
  if B >= 7 then
    'Motors forward
    out %0101xxxx
  elseif B < 7 and J < 3 then
    J = J + 1
  elseif J = 3
    out %1010xxxx
    wait 1
    out %0x0xxxxx
    gosub sub_Turning
    J = 0
  endif

  gosub sub_Headlights
loop

sub_Turning:
do
  ultra C, 0, 0
  if C >= 7 then
    return
  else
    out %0001xxxx
    wait 1
  endif
loop

sub_Headlights:
'If the detected light is too low, we have to turn the headlights on.
if A1 > 160 then
  out %xxxx0x0x
  return
else
  out %xxxx1x1x
  return
endif
```

These comments do not affect the code, but allow a human to better understand what is going on

This section loops indefinitely, or until the power is removed

The use of the variable J is an attempt to avoid false positives from the ultrasonic sensor

This part reverses for 1000ms to pull the robot away from any danger

This subroutine runs when called in the loop, executes its code, then returns to where it was called

Turns until the obstacle is no longer detected

BUILDING Software rev. 1

Unfortunately, along with a slew of other issues, the previous program was too complex for its own good. Given more time, I could probably have found a way of working it out, but the end of the project was rapidly approaching and time was of the essence.

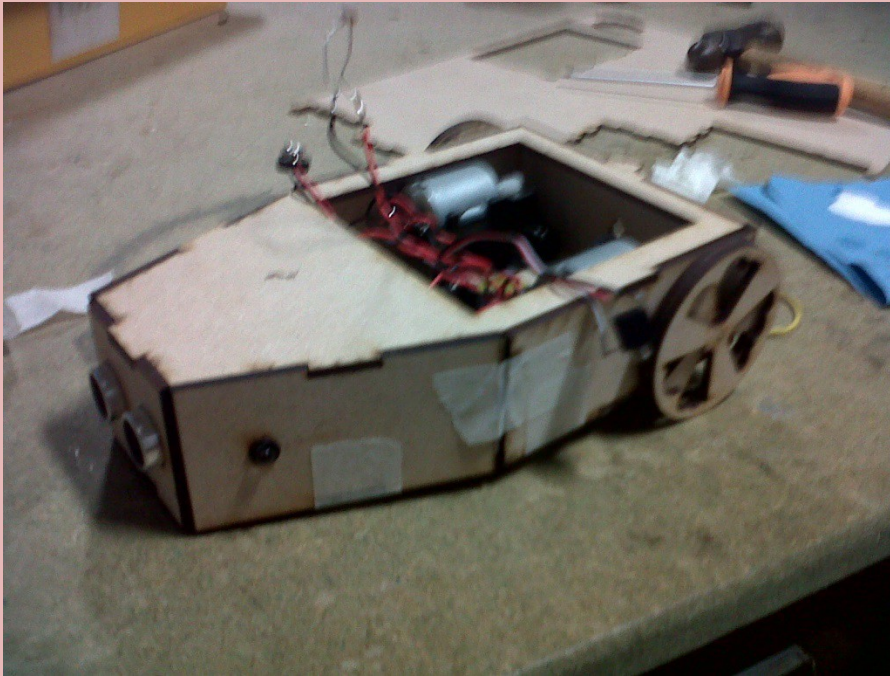
```
start
  'Turn on the LEDs to show that the car is switched on.
  out %xxxxxx1x
  'This loops until power is removed.
  do
    'Drives the car forwards
    out %0101xxxx
    'Waits 100ms
    pause 100
    'Stops the car
    out %0000xxxx
    'Waits 100ms again
    pause 100
    'Sends an ultrasonic ping out in front of the car.
    ultra B, 0, 0
    'If the distance detected is less than 10cm, there is an obstacle.
    if B < 10
      'This jumps the code to the turning subroutine below.
      gosub sub_Turning
    endif
  loop

sub_Turning:
  'This loops until an obstacle is no longer detected.
  do
    'Sends out another ultrasonic pulse.
    ultra C, 0, 0
    'If the distance detected is more than or equal to 10cm...
    if C >= 10 then
      '...the code returns to where it jumped out of the previous block,
      'at the end of the loop.
      return
    'However, if the distance detected is less than 10cm still...
    else
      'The car makes a sharp turn, with both wheels spinning in
      'opposite directions.
      out %1001xxxx
      'It continues to turn for 1000ms, then runs through this sub again.
      pause 1000
    endif
  loop
```

To rule out the possibility of movement confusing the ultrasonic sensor, I changed the program so that the car would stop before taking a reading. This happens very quickly and gives the robot a jittery motion.

Glancing blows confused the ultrasonic sensor, so I added a 1000ms pause to ensure that the robot definitely did turn some amount. A pause of 4000ms would be a 90° turn, which is worth keeping in mind if it came to designing obstacle courses for the robot that it could cope with.

BUILDING Housing



I assembled the housing using PVA glue, with masking tape to hold the pieces together whilst it dried. When it was done, I realised a number of further issues: the small pilot wheel at the front was far too small for the car, and would need to be replaced; the ultrasonic distance sensor, pointed straight forwards as it was, would periodically register the floor as an obstacle and the robot would freak out; and there were some more minor issues to do with hole placement. However, all was not to go swimmingly...

My first attempt to laser cut the housing was thwarted by the screws on the laser print head coming loose and leading to an unusable mess. Attempt #2 was more successful, producing a net as per my specifications. However, when I assembled the net, I found that a couple of my measurements were off and that it didn't quite fit together as planned. I made the necessary corrections and tried again, and this time it all came out perfectly.

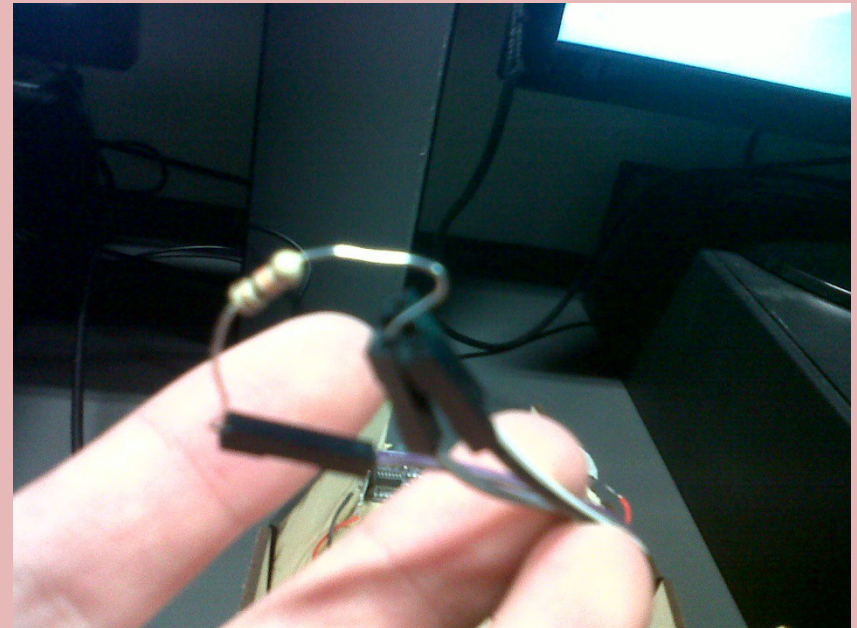


BUILDING The Download Cable Issue



Accepting that I had managed to break physics, I soldered the 10kΩ resistor to the PCB and the robot continued to work fine. Perhaps a smarter man than I would be able to posit an explanation for this mystery, but for now the resistor hack seemed to satisfy the unknowable laws of the universe so I could move on with the rest of the project.

For reasons unknowable, the robot would work fine just as long as the download cable was plugged in. Much time was spent trying to work out if it was the program, the PIC chip or something else. Eventually, as a last-ditch attempt to remedy the problem, I tried the robot with the download cable plugged into the download socket, but the other end not plugged into the computer. Inexplicably, this worked. Essentially, the download cable was now functioning as a pointless 10kΩ resistor that nonetheless was required for the robot to work properly.



BUILDING Housing



I fiddled with the laser cutter settings and it worked the second time around. Assembling the housing meant using polymer cement, which is injected around joints and is then sucked in via a capillary action. The pieces must then be held in position for some time to allow the glue to work its magic, and then you are left with two nigh-inseparable pieces of acrylic. The glue worked a treat, giving me absolutely no issues; a welcome respite after the download cable debacle.

Having confirmed that the housing net was largely correct in MDF, I laser cut it out in acrylic. Some changes were made, such as replacing the pilot wheel with a 3D-printed ball socket and repositioning some mounting holes, and as I expected this to be the final version I also added some cosmetic touches, more for my own amusement than any other reason. My initial attempt went poorly, as the etched areas came out as garbled messes, some parts didn't cut completely out and one piece warped with the heat and was unusable.



BUILDING Final Product



After this, I plugged in the batteries to ensure that everything was working. The motors turned, and when I placed my hand in front of the sensors the car reversed and then turned until I moved my hand away, whereupon it returned to driving forwards. Satisfied, I put it down on the desk and watched as it shuddered up to a wall, stopped, then turned to the right and drove off. Success!

With the housing designed and assembled in acrylic, all that was left was to mount the PCB and external components within it and seal it all up. I started by mounting the PCB onto the holes on the bottom of the car. With this done, I wired up the external components such as the LDR and the SRF05. With this done, I positioned them in their respective holes. The motors needed their poles trimmed to fix in the car beside each other, so I clamped them to a desk and cut them to size with a hacksaw. They were then affixed to the chassis with M4 screws.



BUILDING Issues

Issue	Problem	Diagnosis	Solution	Time
1	Broken track on PCB	Multimeter gave no continuity reading	Piece of wire soldered over gap. An easy fix, and luckily the only one I had to make despite poor form with regards to the etching	15 mins
2	LED needing additional output to turn on	LEDs would only turn on with inputs Q0 & Q2 turned on, despite only being wired up to Q0	Initially just rewrote code, then later used a craft knife to scrape away track connecting LEDs to Q2. Either would have been fine, but I thought I would need Q2 for a LCD screen to troubleshoot the download cable issue so had to free it	5 mins
3	Laser cutter botched the job	Pieces excessively burnt, holes uneven, lines inconsistent with 2d drawing	Re-tightened laser cutter print head screws. A small nuisance, but not overly taxing	20 mins
4	Pilot wheel too small	Assembled case, observed wheel size insufficiency	Replaced wheel with 3D-printed ball socket. Another easy fix, and in retrospect the initial wheel would have been terrible when the car tried to turn	5 mins
5	The download cable issue	The robot worked fine until the download cable was removed	Place resistor of equal resistance to the download cable over the download socket pins, hope. I have no idea what happened, but I don't want to question it.	2 hours
6	Laser cutter botched the job again	Pieces warped, not cut through, etches a mess	Cleaned the laser cutter mirrors and played with the settings. Again, a small nuisance	25 mins
7	Car stopped working when placed on the floor	Car worked fine when elevated	Angled ultrasonic face 20° inwards. This didn't work, but reprogramming the robot did.	3 hours

Total time: 6:10h

BUILDING Specification

Size	The car must be 200mm $\pm$ 10mm from bumper to bumper, approximately the size of a netbook. The controller must be 150mm $\pm$ 10mm in order to fit in the hands easily.	FULFILLED
Ergonomics N/A	The car measures 210mm from bumper to bumper. There is no controller. The remote control must weigh between 120-480g and the controls must all be reachable from a holding position, without any grip movement necessary. There is no remote control any more.	
Aesthetics	Function is more important, although a rough car shape would be a goal. If a stranger can be asked to describe what the product resembles and replies with “a car”, this shall be considered fulfilled. The robot looks like a car and has had people comment to this effect.	FULFILLED
Power FULFILLED	The product must be able to run for 10mins $\pm$ 2mins on 3xAA premium-quality batteries. I never had a chance to test the robot with batteries, but the calculations suggest this works.	
Components FULFILLED	The school will cover the first £10 of the project; after that I’m on my own, so the components must not come to much more than £15 $\pm$ £2, and they must collectively not weigh so much as to make the car unable to move. The car can move, and I estimate the cost to be around £12.	
Programming FULFILLED	The program must be compact enough to be stored on the PIC microcontroller without needing any extra EEPROM memory, and an unfamiliar user must be able to understand the code via provided comments. People commented on the code being relatively understandable, and no extra memory was needed.	
Materials FULFILLED	The car must be durable enough to survive a collision and light enough to move when told. Four collisions and the robot was still standing, and more importantly moving.	
Product Testing FULFILLED	The tests are in the following slide. The robot passed all but the battery test, as explained above.	
Maintenance FULFILLED	Beyond battery replacements, there must be no tasks that require the user (e.g. changing oil). This is satisfied.	
Product Cost	This is a proof of concept, but a price of £20 $\pm$ £5 to cover the predicted cost of £15 $\pm$ £2 would be reasonable in a retail version. I feel I would be comfortable charging this, or perhaps £15.	FULFILLED
Product Life Span FULFILLED	Notwithstanding crashes and falls, the product could conceivably work for years if given a constant stream of batteries. This seems likely.	
Unattended Mode FULFILLED	The car must be able to left to its own devices for 5 minutes without getting trapped. I left the robot for 5 minutes and it was still going by the end of it.	

EVALUATION Testing

The **collision test** will involve driving the car into a wall four times and then testing it to see if it still works.

Expected Result	Actual Result
-----------------	---------------

The car survives the crashes and continues to work afterwards.	
--	--

The car survived the crashes and continued to work afterwards.
--

The **battery life test** will involve wiring up the car to fresh batteries and timing how long it takes for it to run out of power. As the car takes no human input, it can be simply left alone for this period.

Expected Result	Actual Result
-----------------	---------------

The car continues to run for about 10mins $\pm$ 2mins on 3xAA batteries.	
--	--

I couldn't get my hands on any non-drained AA batteries, but the product testing earlier would suggest they would work.

The **polarity test** will involve wiring the power supply up to the wrong nodes and seeing if the circuit copes.

Expected Result	Actual Result
-----------------	---------------

The diode will restrict the flow of current.	
--	--

The circuit survived..

The **traction test** will involve putting the robot on various surfaces to see if it can still drive.]

Expected Result	Actual Result
-----------------	---------------

The car manages most surfaces it would normal encounter.	
--	--

Thanks to the rubber bands on the wheels, the car coped with everything.
--

The **aesthetics test** will involve asking an impartial stranger about their reaction to the product, with an aim for recognition of its carlike quality.

Expected Result	Actual Result
-----------------	---------------

The stranger finds the robot appealing to look at, and recognises it as carlike.	
--	--

An interviewed stranger remarked that "it looks like a car".
--

The **price test** will involve tallying up the prices for all of the components used (using the cheaper prices that the school can get) to ensure that the product is reasonable to produce.

Expected Result	Actual Result
-----------------	---------------

The price will be within $\pounds 15 \pm \pounds 2$.	
---	--

The price came to $\sim \pounds 15$.

The **function test** will involve leaving the car to its own devices for five minutes (or less, pending the results of the **battery life test**) in an obstacle-ridden (although level, due to limitations in the product) environment.

Expected Result	Actual Result
-----------------	---------------

After the allotted time, the car has not found itself caught on anything and continues to drive around.	
---	--

The car drove around for five minutes and didn't get caught.
--

The **size test** will involve measuring the dimensions of the finished product. It will also incorporate the **weight test**, whereby whether the weight of the product is suitable shall be tested by seeing if the car can move.

Expected Result	Actual Result
-----------------	---------------

The product is within the boundaries set in the specification and is light enough to move.	
--	--

The product measures 210x140x90mm and is light enough to move.
--

EVALUATION User Feedback

I asked a number of people unfamiliar with the project to observe and comment on it. Collected here are their views, from which a more detached, neutral view on the successes and failures of the project can be obtained, as well as an indication of any potential marketability.

Lucy Sluice

Age: 18
Job: Student

Miss Sluice has recently finished her A-Levels. As such, she now finds herself with more free time than she is used to. She inquired about the robot after hearing about it through a friend. She was appreciative of the fact that she could leave the robot running in her room and occasionally pay attention to it, like other executive toys, e.g. Newton's cradles, drinking birds, etc.



"I enjoy the robot; it's a nice little distraction in the background. In between planning what I want to do at university and hanging out with friends I like to unwind by sprawling out on my bed and watching it avoid obstacles or setting my Newton's cradle off.

It isn't perfect, however. It runs out of charge after about ten minutes, whereas I would prefer it to run for a few hours or something. However it is quick enough to replace the batteries, helpfully located on the outside of the robot, and I have a job that helps to ensure a steady stream of them.

I could see buying this for my little brother to play with, he's four."

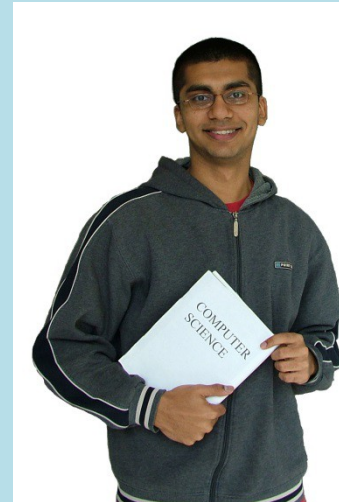
Suggested Improvements

"Deal with the power supply, and then I'll be fine."

Zimran Achmaev

Age: 20
Job: Student

Mr Achmaev is currently at Teesside University studying Computer Science. This gives him a more technical appreciation of the robot and allows him to better suggest feasible modifications. Mr Achmaev was more critical than Miss Sluice, but that is to be expected from someone with his knowledge. He was, however, broadly fond of the project, and I have taken his suggestions on board.



"It's an okay project for someone just starting out in robotics, as you are. I can see from reading your project that you encountered quite a few issues, and whilst I'm as baffled by the download cable issue as the next man, I would say it seems like it has been a rewarding learning experience.

As for the robot itself, I don't know if it would really be worth selling at the moment. It doesn't do an awful lot, and voraciously drains batteries doing the little it does. More functionality or less resource cost would make it more worthwhile.

Suggested Improvements

"Make it do more things, at the moment it is the bare minimum of a robot. Very good job, though."

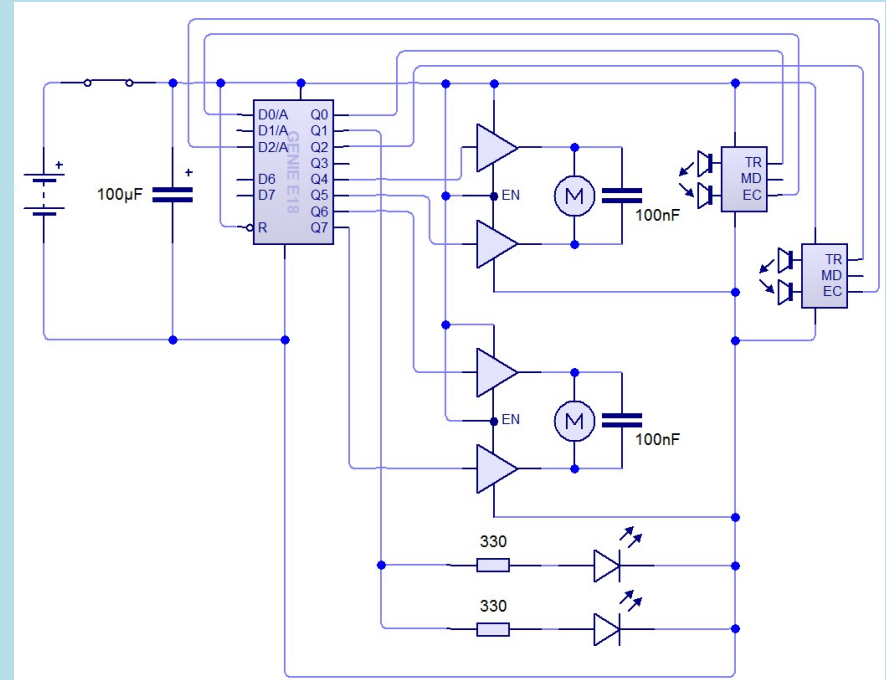
EVALUATION Modifications

Correct Ultrasonic Angle/LDR Hack/Second Ultrasonic Detector

The project is clearly not perfect. When placed on the ground, the car will drive forwards and turn when obstacles are encountered, but there is still progress to be made. For one, obstacles at anything other than perpendicular angles tend to baffle the robot, which would obviously impede its ability to exist in the real world. Another issue is that the robot has no way of detecting drops, arguably a bigger threat to it than obstacles. I could repurpose the LDR I implemented for the LEDs and point it straight down, which would give the car no ability to detect obstacles but would leave it able to recognise (by sudden increases in luminosity) drops and avoid them. This wouldn't be an entirely satisfying fix, however, but it may be the best I can do. If I had enough time for a more comprehensive redesign, I could replace the LDR with a second ultrasonic detector, which would allow the car to drive over luminescent surfaces and hills, which would be the best case scenario.

Program

The program could also be made far more complex. For example, some form of machine learning could be implemented that would allow the robot to learn on its own the best way to avoid obstacles via trial-and-error, and then perhaps teach another copy of the same robot what it has learned. This idea comes from an experiment by Warwick, et al., but there is a lot of scope for expanding it; machine intelligence is a fascinating field.



EVALUATION Modifications

Manipulators

Taking a page from the Mars rover's book, further modifications would be possible that allow the robot to interact with the world. The most fundamental of these could be some sort of servomotor-based articulated arm system, possibly with some sort of pincer on the end for manipulation of the environment. The arm could be replaced by a fire extinguisher, for example, and added heat sensors could allow the robot to be sent into dangerous environments in place of humans to extinguish fires.

Powering

In response to Miss Sluice's comments, I set about thinking about alternative powering methods. There are a number of options, from the conventional and therefore uninteresting ones like solar panels to one that I think I would implement if given the time. Taking a page from the dodgem book, with the addition of a conductive wire to the top and bottom of the car, it could be placed between positively- and negatively-charged plates and power itself indefinitely. Combining this with the Pin Art executive toy, perhaps a box could be made where obstacles could be pushed through the bottom plate and the car would reliably trundle along.

First Project

Another avenue of development, and a more drastic one at that, would be to return to the first idea from the beginning of the project for a web-controlled car. I learnt a bit of Python in anticipation of it, and combined with the manipulators would increase its ability to be used in dangerous environments, a la bomb disposal robots.



EVALUATION Evaluation

Project Evaluation

As you will have seen earlier, the finished product satisfies all of the criteria set out in the specification (with the exception of the ergonomics category, which became redundant after the remote control was removed from the equation). However, the product is by no means perfect. I have gone over these improvements in the previous two slides in detail.

It was a bumpy road to finishing it, and I almost didn't manage to, but I'm glad now that I have. As far as hurdles to overcome, they started off reasonable, with the IR sensor being too intermittent to make the remote control feasible, and slowly built to ridiculous heights. Both times I laser cut the chassis, some error in the laser cutter's alignment ruined the entire batch, as though it had some personal vendetta against my convenience. Then came the solder trapped under the PIC's DIL socket, which took the better part of two days to finally remedy, more by luck than anything. As I'd clearly not had enough masochism for one project, it all culminated in the download cable issue. When a problem is so unexplainable and arcane that it needs an entire slide devoted to its mysteries, you know something is horribly wrong. The finest minds in the S&C department were all at a loss to explain what was happening, but the resistor hack seemed to work so we all joined in an unspoken agreement never to comment on the matter, or observe it any closer, lest the waveform should collapse.

After all that, the robot still refused to work when placed on the ground, but would happily oblige if I'd lift its back about 1mm off of the ground, if that. Thinking it might be due to the angle of the front panel, I moved it around using Blu-Tack, but to no avail. I was on the verge of giving up when I decided to try changing the program, and then it finally decided to work. I don't know what issue it was having, but I have little to no interest in looking into it any more.

The benefit of all these issues is that I'm very proud and relieved to have it all finished and working.

Self Evaluation

Perhaps, for my first ever experience of Systems & Control, I should have chosen something simpler. Having very much not done that, however, and thrown myself in at the deep end, I feel I've learned to swim very quickly. Combined with my background in A-Level Computing, I think this knowledge will come in very handy in the future, especially as I wish to pursue a career in Computer Science. Even beyond that, knowing how circuitry works is a great skill; I could fix things in the future, rather than replace them, or spend my free time tinkering and whipping up all manner of things to save me time and effort, a la *Wallace and Gromit*. Also, despite the multitude of problems I encountered, I (most of the time) enjoyed troubleshooting them, trying to find out just what the cause was this time.

BIBLIOGRAPHY Image Sources

Image of SRF05 ultrasonic distance sensor, slide 7

http://www.dfrobot.com/index.php?route=product/product&product_id=333

Image of mini-LDR, slide 7

<http://laserbots.blogspot.co.uk/2012/12/laser-setup-and-detection.html>

Image of thermistor, slide 7

<http://www.digikey.com/product-detail/en/PT222J2/615-1007-ND/1014535>

Image of slide switch, slide 7

http://cnmat.berkeley.edu/sensor_module/mini_slide_switch

Image of rocker switch, slide 7

<http://web.cecs.pdx.edu/~eas199/B/equipment/>

Image of IR receiver, slide 7

<http://uk.mouser.com/new/vishay/vishayirreceiver/>

Python logo, slide 8

<http://www.python.org/community/logos/>

Image of BASIC, slide 8

<http://en.wikipedia.org/wiki/BASIC>

Image of Motorola MC6800 Assembly Language, slide 8

http://en.wikipedia.org/wiki/Assembly_language

Image of 555 timer, slide 8

<http://www.circuitstoday.com/555-timer>

Image of PIC 16F54 microcontroller, slide 8

<http://www.tandyonline.co.uk/components/semiconductors/microcontrollers.html>

Image of thyristor, slide 8

<http://www.circuitstoday.com/thyristor-vs-thyratrons>

Image of 12V to 24V -150W small motor, slide 9

http://www.everything-ev.com/index.php?main_page=product_info&products_id=236

Image of 5mm green LED, slide 9

<http://www.tandyonline.co.uk/5mm-flashing-green-led.html>

BIBLIOGRAPHY Image Sources

Image of servomotor, slide 9

<http://www.engineersgarage.com/articles/servo-motor>

Image of piezo siren, slide 9

<http://www.bitsbox.co.uk/sensors.php>

Image of loudspeaker, slide 9

<https://simple.wikipedia.org/wiki/Loudspeaker>

Image of 7-segment display, slide 9

http://www.analogiccomponents.com/product_info.php?products_id=56

Image of laser-cut wheels, slide 10

http://www.etsy.com/market/wooden_wheels

Image of caterpillar tracks, slide 10

<http://www.hindleys.com/index.php/craft/wheels-axles-mounting-plates/caterpillar-tracks.html>

Image of blockboard, slide 10

<http://www.rembrand.co/products.php?product=18-mm-x-2440-x-1220-Indonesian-Blockboard-BB%7B47%7DCC>

Image of wheels, slide 10

<http://www.smart-stuff.co.uk/winter-tyres-on-steel-wheels---451-fortwo-3888-p.asp>

Image of BigDog, slide 10

<http://random.mytko.org/2008/03/boston-dynamics-military-robot-dog.html>

Image of paper, slide 10

<http://ze-r-o.deviantart.com/art/Hi-Res-Lined-Paper-Texture-129381292>

Image of corrugated fibreboard, slide 11

<http://visualmattersservices.com/art-storage-guidelines-and-corrugated-cardboard/>

Image of MDF, slide 11

<http://alamedaelder.com/mdf-kitchen/8/painting-mdf-kitchen-cabinet-6-view-kitchen-cabinet-pa-kitchen/>

Image of acrylic, slide 11

http://www.tapplastics.com/product/plastics/cut_to_size_plastic/acrylic_sheets_transparent_colors/519

BIBLIOGRAPHY Image Sources

Image of steel, slide 11

<http://buildipedia.com/at-home/kitchen/countertops-101?print=1&tmpl=component>

Image of copper, slide 11

<http://metalloffcuts.co.uk/shop/copper-sheet-metal-classic-natural-finish.html>

Image of 3D printer, slide 11

http://www.theregister.co.uk/2013/02/04/ten_3d_printers/

Image of remote-controlled car, slide 12

<http://familycourtchronicles.com/philosophy/dissonance/>

Image of parking sensor, slide 12

<http://www.safestchina.com/wholesalers-reverse-parking-sensor/>

Image of Mars rover, slide 12

http://en.wikipedia.org/wiki/Mars_Exploration_Rover

Image of PP3 battery, slide 14

http://en.wikipedia.org/wiki/Nine-volt_battery

Image of Tornado, slide 36

<http://www.teamstorm.com/storm2/grandfinals.htm>

Image of Kevin Warwick, slide 59

<http://www.kevinwarwick.com/>

Image of robotic arm, slide 60

http://www.iccrobotics.com/robot_arm.html

Image of dodgem cars, slide 60

<http://www.irvinleisure.co.uk/rides-for-hire/dodgem-cars/>