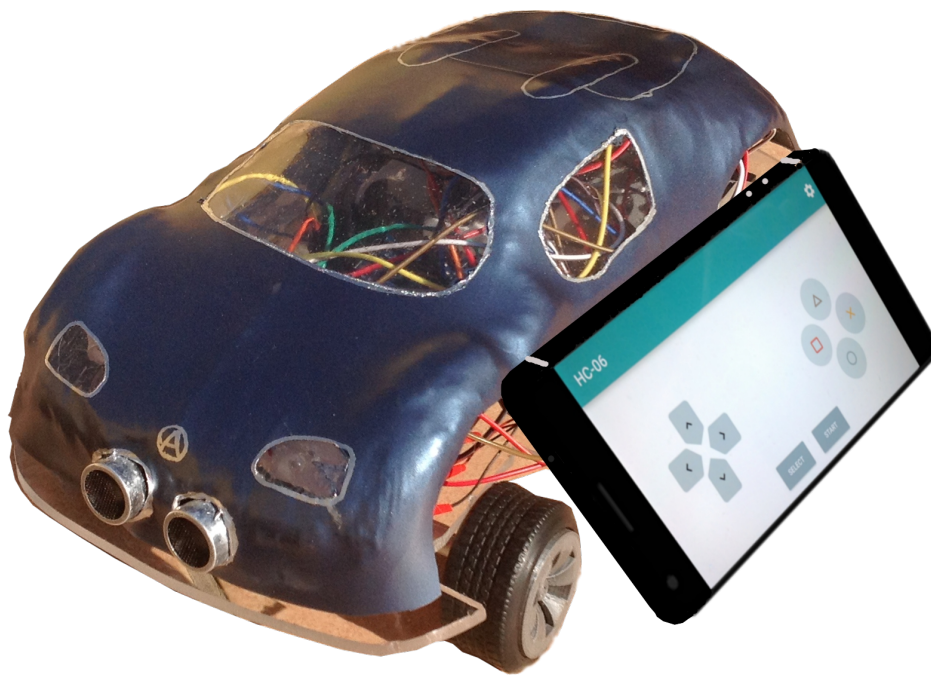


CREACIÓ I PROGRAMACIÓ D'UN COTXE TELEDIRIGIT AMB EL MÒBIL



Arnau Pararols i Colom
INS Narcís Monturiol (Figueres)
2n Batxillerat A
Tutor: Ivan Gonzalez
Data de presentació: 03/11/17

AGRAÏMENTS

Primer de tot, m'agradaria donar les gràcies a la meva família, ja que m'han donat suport i motivació durant tot el Treball de recerca, i m'han ajudat en tot el que han pogut. També els vull agrair el fet que m'hagin portat on necessitava per tal de poder completar el Treball.

Vull agrair a tres persones en concret perquè han fet que aquest treball funcioni. Són l'Arnau Caritg i el seu pare Joan Caritg, del taller Caritg, que m'han imprès totes les peces que he dissenyat amb la impressora 3D i totes les variacions a les peces que he anat fent. I agrair també a Joan Pellicer, exprofessor de tecnologia de l'Institut Cendrassos, que m'ha ajudat molt en la resolució dels problemes que he tingut al final del treball aportant-me solucions o bé ensenyant-me com deduir-les i arreglar els problemes. També vull agrair-los tot l'interès mostrat pel meu treball i tot el suport que m'han donat i com s'han prestat en ajudar-me.

Finalment m'agradaria agrair a totes les persones que m'han donat suport, sobretot quan tenia problemes amb el cotxe, agrair a les que m'han donat consells sobre el marc teòric i, en conjunt, agrair a tots aquells i aquelles que han fet possible el Treball.

ÍNDIX

1	Introducció.....	6
2	Cotxes de radiocontrol.....	7
2.1	Història	7
2.2	Tipus de cotxes de radiocontrol	8
3	Materials.....	11
3.1	Components elèctrics.....	11
3.1.1	Placa Arduino	11
3.1.1.1	Tipus de plaques Arduino.....	12
3.1.2	Mòdul Bluetooth.....	13
3.1.2.1	Història del Bluetooth.....	13
3.1.2.2	Logo del Bluetooth	13
3.1.2.3	El mòdul Bluetooth.....	14
3.1.3	Sensor de proximitat.....	14
3.1.4	Leds.....	16
3.1.5	Cables.....	17
3.1.6	Motors.....	17
3.1.7	Resistències.....	18
3.1.8	Brunzidor.....	20
3.2	Components de muntatge.....	20
3.2.1	Metacrilat.....	20
3.2.2	Cargols, femelles i bagues tancades.....	21
3.2.3	Planxa de plàstic.....	21
3.2.4	Pintura i promotor d'adherència.....	22
3.3	Pressupost.....	22
4	Explicació procés.....	24
5	Conclusions	36
6	Fonts d'informació.....	37
6.1	Webgrafia.....	37
6.2	Fonts d'il·lustracions.....	38
7	Annexos.....	41
7.1	Evolució del programari.....	41
7.1.1	Primera prova de control dels motors amb tecles del teclat.....	41
7.1.2	Prova del programa per girar amb les tecles.....	41
7.1.3	Introducció de leds al programa anterior.....	42
7.1.4	Programa anterior amb intermitència als leds.....	44

7.1.5 Programa d'iniciació per entendre el Bluetooth.....	45
7.1.6 Aplicació del mòdul Bluetooth i llums frontals en el programa fet anteriorment	46
7.1.7 Programa de la cançó "Back in Black" d'ACDC pel clàxon.....	48
7.1.8 Programa de la cançó inicial dels Simpsons pel clàxon.....	50
7.1.9 Programa d'iniciació per entendre el sensor de proximitat.....	51
7.1.10 Aplicació del sensor de proximitat i música al programa anterior.....	52
7.1.11 Programa d'introducció als servomotors.....	56
7.1.12 Programa final amb servomotors.....	57
7.1.13 Programa final.....	61
7.2 Disseny de peces amb SolidWorks.....	65
7.2.1 Disseny de les llantes.....	65
7.2.2 Disseny suports servomotors.....	65
7.2.2.1 Disseny suport servomotor dret.....	65
7.2.2.2 Disseny suport servomotor esquerre.....	66

1 INTRODUCCIÓ

Quan ens van començar a parlar del Treball de Recerca a principi de curs jo veia que em quedava molt lluny, però parlant amb amics més grans que ja l'havien fet i els companys de classe vaig veure que quan abans m'hi poses millor perquè podien sortir molts imprevistos que són més o menys fàcils de solucionar si tens temps, però si vas amb el temps just segurament, per més petits que siguin, no els pots resoldre com voldries.

Al principi no sabia quin tema volia tractar en el treball de recerca, només sabia que volia fer-lo relacionat amb la tecnologia. Com que l'any anterior havia tingut a l'Ivan Gonzalez com a professor de tecnologia i ens enteníem molt bé vaig demanar per fer el treball amb ell. Vaig portar dues idees principals sobre temes que m'agradaria fer que un era *les impressores 3D de menjar* i l'altre *la substitució de guix mèdic per motlles 3D*. Li vaig presentar aquests dos temes perquè veiés cap on anava orientat però jo estava obert a propostes. Em va proposar fer un cotxe teledirigit amb el mòbil i em va semblar una molt bona idea, ja que no seria un treball tan concentrat només en redactar i fer entrevistes sinó més orientat en el disseny, la programació i la creació.

Pel treball volia assolir diferents objectius. El primer objectiu era que el cotxe funcionés. El segon, poder introduir-hi leds a l'hora de girar i anar marxa enrere, i leds frontals per tal de fer-lo més realista. El següent objectiu era que el pogués conduir amb el mòbil via Bluetooth. Un altre que volia assolir, era el que distingia el meu cotxe de la gran majoria, volia posar-hi un sensor de proximitat perquè no xoqués. I per últim, volia que el cotxe fos el més econòmic possible però que quedés bé estèticament.

Pel cotxe volia agafar un cotxe de joguina i adaptar-lo, però per problemes de capacitat i estètica vaig acabar fent un cotxe teledirigit des de zero, creant-lo i dissenyant algunes peces jo mateix.

2 COTXES DE RADIOCONTROL

2.1 Història

La primera empresa a comercialitzar cotxes de ràdio control va ser la italiana EI-Gi (Electronica Giocattoli) i ho va fer l'any 1966 amb un primer model del Ferrari 250LM a escala 1:12, aquest va ser seguit pel Ferrari P4 d'escala 1:10 de la mateixa empresa. Aquest segon va ser presentat a la Fira de Juguines de Milà del 1968.

El 1973, l'empresa Jerobee, va crear un cotxe propulsat per nitro d'escala 1:12 utilitzant el motor Cox 049. D'aquest model se'n varen vendre cap a unes 50.000 unitats. Llavors l'empresa va ser comprada per Rocket Research Corp. la qual va millorar el cotxe incloent dipòsits amb més capacitats, dissipadors de calor i carcasses de policarbonat.

El 1976 l'empresa Associated Electrics va crear el RC12E, un dels primers cotxes de radiocontrol elèctrics. Després l'empresa Jerobee va crear el seu cotxe elèctric anomenat Lighting 2000 el qual va guanyar la ROAR National Championship¹ els anys 1981 i 1982.

A mitjans dels anys 70 es van inventar les bateries recarregables, la qual cosa va suposar un gran avenç, ja que anteriorment estaven fetes d'un àcid car d'aconseguir.

L'any 1977, la marca japonesa Tamiya va començar a crear carcasses de plàstic, la qual cosa permetia fer més models de cotxes i amb un nivell de detall molt alt. També van ser els pioners en crear buggies de radiocontrol amb sistemes d'amortització reals. El fet de crear sistemes d'amortització va fer que el hobby dels cotxes de radiocontrol pogués sortir dels terrenys planers i poder començar a anar per terrenys irregulars i rocosos. Els primers models tot terreny van ser el Sand Scorcher i el Rough Rider, els dos creats el 1979 i basats en dissenys de buggies reals. L'empresa Tamiya va seguir fent cotxes tot terreny millorant les suspensions, els motors, pneumàtics de cautxú i diferents carcasses estilitzades. També varen començar a fer cotxes més robustos com el Toyota HiLux Pickup. Els models de Tamiya eren realistes, duradors i fàcils de muntar, modificar i reparar.

¹ **ROAR National Championship:** competició de cotxes de radiocontrol a Estats Units i Canadà.

Dins dels models més coneguts de la marca Tamiya hi ha el Grasshopper i el Hornet com a buggies i com a monster trucks hi ha el Blackfoot i el Clodbuster.

L'any 1980, la marca Schumacher Racing² va desenvolupar un diferencial de bola ajustable, el qual permetia infinitats de modificacions per les diferents condicions de la pista.

El 1986, l'empresa Shumacher Racing Productes va crear el cotxe CAT (Competition All Terrain), el qual va ser considerat el millor buggy tot terreny de carreres d'aquells temps. El CAT va guanyar el campionat mundial tot terreny de l'any 1987.

Actualment, la tecnologia dels cotxes de radiocontrol ha avançat molt i tots els cotxes de competició porten sensors i detalls tècnics per obtenir més bon rendiment del cotxe i més bons resultats i els cotxes més coneguts són els Mini-Z de la casa Kyosho.

Anys 60, inici dels cotxes de radiocontrol.	Anys 70, era del nitro, cotxes elèctrics i modalitat off-road.	Anys 80, carreres de cotxes ràpids i off-road 4WD.	Anys 90, era dels cotxes de Touring.	Actualitat, gran creixament del Rockcrawling i el Shot Course.
---	--	--	--------------------------------------	--

Il·lustració 1: Eix cronològic dels cotxes de radiocontrol.

2.2 Tipus de cotxes de radiocontrol

Els cotxes de radiocontrol es poden classificar segons l'escala del cotxe, la tracció, les categories i el tipus de motor.

· Escala:

Quan parlem d'escala en temes de mida ens referim a la relació de la mida del producte amb la de la realitat. Per exemple, quan diguem que està a escala 1/8, vol dir que per una unitat al producte n'hi ha 8 al real. Això ho podem resumir amb la fórmula $1/n$, que el producte és n vegades més petit que el real.

Les escales principals i més utilitzades en el món dels automòbils de ràdio control són:

- Escales d'1/5 i 1/6. Són cotxes d'1 metre aproximadament, la qual cosa els fa més espectaculars, amb més detalls d'acabats i més voluminosos a l'hora de guardar-los. Al ser de dimensions considerades, són cotxes únicament d'exterior, ja que necessiten més espai per córrer. La majoria porten motors de gasolina i amb un

² **Shumacher Racing:** productora anglesa de cotxes de radiocontrol i accessoris destinats a les competicions.

motor de 125 cc poden arribar a estar rodant durant mitja hora sense parar.

- D'1/8 fins a 1/16. Són les escales més utilitzades perquè tenen una mida que no requereix un gran espai per emmagatzemar-los i es poden utilitzar tant en interiors com exteriors. Com que són els més utilitzats, no et serà difícil trobar gent amb qui compartir el hobby o competicions d'aquestes escales. També són els que presten més varietat de modificacions sigui de carrosseries o accessoris.
- Escales fins a 1/18. Els cotxes amb aquestes escales, necessiten molt poc espai per guardar-los gràcies a les seves petites dimensions i la gran majoria són de tipus RTR (Ready to Road = preparat per córrer), és a dir, que no requereixen *muntatge*. Estan destinats a ser utilitzats en interiors i en terrenys molt uniformes a causa de la seva altura amb el terra. La majoria de cotxes d'aquestes escales són elèctrics i estan destinats a la iniciació. Com que aquestes escales són una minoria, és difícil trobar modificacions pel cotxe i no existeixen gaires competicions.

·Tracció:

Es classifica segons el nombre de rodes motrius:

- Tracció normal. Quan hi ha dues rodes motrius, tant sigui a davant com a darrere. També s'anomena 2WD, sigles de l'anglès 2 Wheels Driving.
- Tracció integral. Quan les quatre rodes són motrius. Anomenat també 4WD, de l'anglès 4 Wheels Driving.

·Categories:

Les denominacions utilitzades pels cotxes segons la finalitat per la qual han estat dissenyats són:

- Off road. Fa referència a vehicles creats per anar per pistes de terra, és a dir, cotxes tot terreny.
- On road. Són vehicles que agafen velocitat a pistes d'asfalt.
- Monster. Principalment eren camionetes modificades amb grans suspensions i grans pneumàtics per donar espectacle amb salts, tot i que últimament s'utilitzen tot tipus de cotxes.
- Drifting. Cotxes de pista destinats a l'exhibició amb derrapatges.
- Rockcrawling. És una versió més extrema de l'Off road.
- Shot Course. Cotxes modificats en un circuit tancat de terra amb poca longitud.

- Touring. Cotxes destinats a donar voltes a un circuit ovalat amb les corbes aixecades per agafar més velocitat.

·Tipus de motor:

Quan classifiquem els cotxes de radiocontrol pel motor els podem diferenciar en dues classes:

- Motor de combustió. Són els primers cotxes de radiocontrol que van sortir i van propulsats per la combustió d'òxid de nitrogen (1), més conegut com a nitro. Aquests motors tenen un avantatge envers els elèctrics i és que al ser de combustió, creen un ambient més realista amb el soroll en córrer i l'olor de benzina, tot i que contaminen més i siguin més cars de mantenir.
- Motor elèctric. Són cotxes alimentats per bateries, la qual cosa té l'inconvenient que tenen un temps límit de funcionament. Els cotxes elèctrics tenen un factor que es pot considerar un avantatge o un inconvenient segons per qui, i és que no fan gaire soroll ni contaminen, tot i que hi ha gent que diu que perden l'essència dels cotxes. Un avantatge que tenen els cotxes elèctrics és que no necessiten tant de manteniment i són més econòmics.

3 MATERIALS

En aquest treball vaig utilitzar molts materials diferents i els he classificat en dues branques, els components elèctrics per fer funcionar el cotxe i components de muntatge per subjectar els components elèctrics o per decoració del treball.

3.1 Components elèctrics

3.1.1 Placa Arduino

La placa Arduino és una placa de circuit imprès simple. Està basada en el microcontrolador de codi obert Wiring³.

L'objectiu de les plaques Arduino és facilitar el disseny de circuits electrònics. Les plaques tenen un conjunt de pins els quals n'hi ha de digitals i d'analògics, igual que n'hi ha d'entrada i de sortida d'informació.

El projecte d'Arduino va començar al Interaction Design Institute Ivrea (IDII) a Ivrea, Itàlia. Els alumnes utilitzaven el microcontrolador anomenat BASIC Stamp, que en aquells temps valia uns 100 \$, que era molt car. L'any 2003 Hernando Barragán va crear el Wiring com a treball de final de màster. La idea del treball era crear eines barates i simples de manejar per tal que la gent inexperta pogués crear projectes digitals. El Wiring utilitzava un microcontrolador anomenat Atmega168, el qual l'any 2003, un grup d'alumnes de la mateixa Universitat d'Ivrea, el van substituir per un de més econòmic anomenat Atmega8. El grup d'estudiants va decidir canviar el nom de Wiring a Arduino.

Els integrants d'aquest grup van ser Massimo Banzi, David Cuartielles, Tom Igoe, Ginaluca Martino i David Mellis, i, encara que l'inici del projecte fos obra d'Hernando Barragán, no va ser convidat a formar part d'Arduino. El nom d'Arduino provenia del nom d'un bar d'Ivrea on els integrants del grup solien quedar. El bar va posar aquest nom en honor a Arduin d'Ivrea, que va ser marquès d'aquesta mateixa ciutat i rei d'Itàlia del 1002 dC al 1014 dC.

3 **Wiring:** és una plataforma de programari obert composta per llenguatge de programació, una interfície de desenvolupament integrada i una placa base amb un microcontrolador. Va ser desenvolupada per Hernando Barragán l'any 2003.

L'empresa Adafruit Industries és la proveïdora de les plaques d'Arduino. A mitjans del 2011 l'empresa havia distribuït unes 300.000 plaques oficials d'Arduino, nombre que dos anys després, el 2013, duplicà amb més de 700.000 plaques venudes.

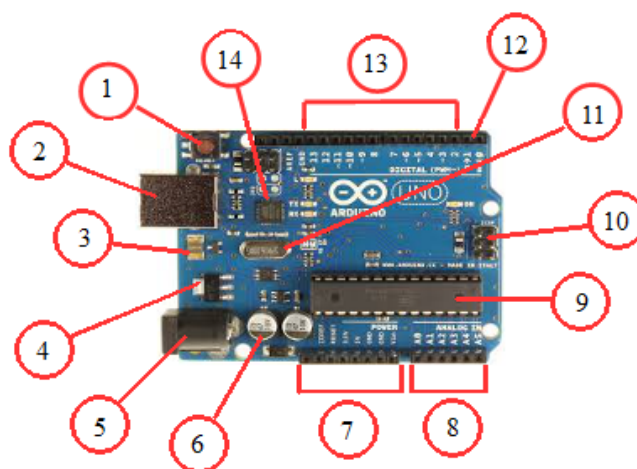
3.1.1.1 Tipus de plaques Arduino

Al mercat actual hi ha una gran varietat de plaques Arduino segons la seva finalitat. Les plaques més utilitzades són l'Arduino Micro, l'Arduino Nano, l'Arduino Mega, l'Arduino Yún i l'Arduino UNO entre altres. Per fer el treball jo vaig utilitzar la placa Arduino UNO, la qual es pot adquirir per uns 20 € a la botiga oficial d'Arduino.

Està formada per 14 pins digitals que són tan de sortida d'informació com d'entrada, 6 pins analògics d'entrada, un cristall oscil·lador a 16 MHz, una connexió USB, un connector d'alimentació, un encapçalament d'ICSP (Programació Serial en Circuit) i un botó de reinicialització.

Els elements que componen la placa són:

1. Botó de reinicialització.
2. Connector USB.
3. Fusible⁴.
4. Regulador de voltatge de 5 V.
5. Alimentació de 5 V a 12 V.
6. Condensadors⁵ de filtre.
7. Pins d'alimentació.
8. Entrades analògiques.
9. Microcontrolador ATmega 328.
10. ICSP.
11. Cristall oscil·lador a 16 MHz.
12. Pins de connexió serial.
13. Entrades i sortides digitals.
14. Xip de comunicació.



Il·lustració 2: Placa Arduino amb les parts senyalades.

4 **Fusible:** element de protecció del circuit.

5 **Condensadors:** dispositiu que emmagatzema energia.

3.1.2 Mòdul Bluetooth

3.1.2.1 Història del Bluetooth

El nom de Bluetooth prové del rei danès i noruec Harald Blåtand, el qual solia menjar nabius i li quedaven les dents blaves. La gent li deia Harald dents blaves, que traduït a l'anglès és Harald Blue tooth. Harald va ser conegut per fer que les tribus noruegues i suecodaneses s'unissin i es convertissin al cristianisme. Jim Kardach va posar el nom de Bluetooth al seu sistema que permetia als telèfons mòbils comunicar-se amb els ordinadors afavorint la connexió sense cables, ja que era com unificar un grup d'aparells.



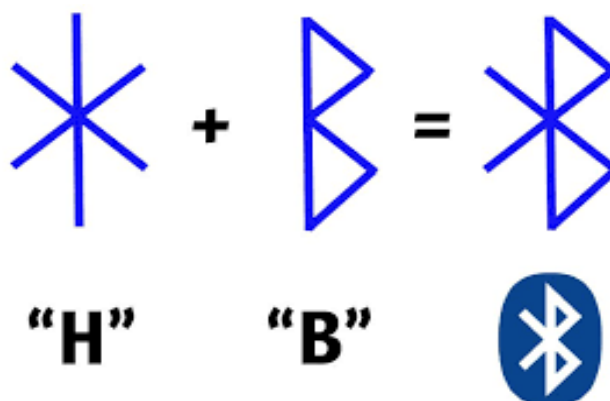
Il·lustració 3: Harald Blåtand



Il·lustració 4: Jim Kardach

3.1.2.2 Logo del Bluetooth

El logo de Bluetooth prové de la combinació de les runes⁶ Hagal i Berkana, que corresponen a les inicials de Harald Blåtand respectivament.



Il·lustració 5: Lletres Hagal i Berkana de l'alfabet rúnic i logo de Bluetooth.

⁶ **Runes:** conjunt de signes que formen l'alfabet rúnic, utilitzat en l'antiguitat a Escandinàvia i les illes Britàniques per escriure llengües germàniques.

3.1.2.3 El mòdul Bluetooth

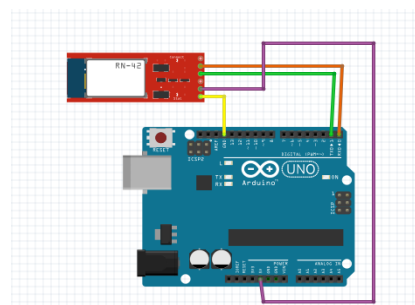
El mòdul Bluetooth és un aparell que permet la transmissió de veu i informació entre dos dispositius mitjançant una radiofreqüència. Permet que dos dispositius es puguin comunicar quan aquests es troben a l'abast, és a dir, fins a un límit de 100 metres entre ells depèn de la classe de potència del Bluetooth que s'estigui utilitzant. Les classes es poden diferenciar en tres diferents i són:

- Classe 1: té una potència de 100 mW i un límit de 100 metres de separació entre els dos dispositius.
- Classe 2: té una potència de 2,5 mW i permet que entre els dispositius hi hagi una distància de 10 metres. Els dispositius de classe 2 són els que s'utilitzen més freqüentment.
- Classe 3: és el dispositiu que menys s'utilitza, ja que la distància de separació va de 10 centímetres a 1 metre.

El mòdul que he fet servir jo pel treball s'anomena HC-06 que és un mòdul creat per les connexions de la classe 2, el qual només pot enviar o rebre informació però no les dues a la vegada. Llavors hi ha el mòdul HC-05 que aquest sí que et permet enviar i rebre informació a la vegada però com que jo només necessitava que el Bluetooth rebés informació, amb el mòdul HC-06 ja en vaig fer prou. Aquest mòdul es pot trobar al mercat per uns 8 € aproximadament.



Il·lustració 6: Mòdul Bluetooth HC-06.



Il·lustració 7: Esquema de la connexió del mòdul Bluetooth amb la placa Arduino.

3.1.3 Sensor de proximitat

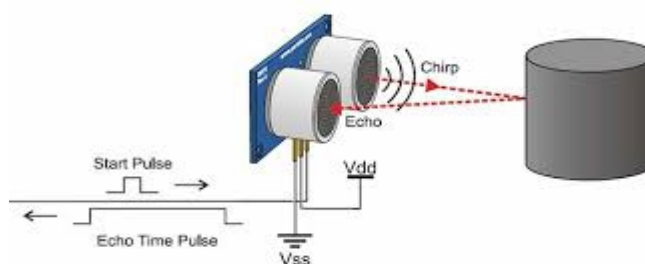
El sensor de proximitat és un sensor que detecta la presència d'objectes sense la

necessitat de tocar-los físicament. Ho fan mitjançant radiació lumínica, camp electromagnètic o radiació electromagnètica i mesurant la diferència de temps entre l'anada del senyal i la tornada.

Principalment, el sensor de proximitat va ser dissenyat per calcular distàncies entre el detector i l'objecte en qüestió, però sovint s'utilitzen per detectar la presència d'objectes. En detectar presències tenen moltes utilitats, com per exemple, els assecadors de mans, llums sense necessitat de cap polsador o fins i tot processos industrials, amb el conegut "fi de cursa" o "final de carrera"⁷.

Hi ha diferents tipus de sensors segons l'objecte que es vol detectar. Els sensors més comuns són:

- Fotoelèctrics. Utilitzen emissió de radiació lluminosa, la qual emet un raig de llum que sol ser amb llum infraroja perquè les llums d'altres fonts no hi puguin interferir, i reflecteix en els objectes desitjats.
- Magnètics. Són utilitzats per detectar objectes de materials ferromagnètics. Es poden detectar gràcies al fet que en ser ferromagnètics alteren el camp magnètic emès pel sensor i això fa que es modifiqui la freqüència de treball.
- Inductius. Detecten objectes metàl·lics però en aquest cas no ferromagnètics. Aquests metalls provoquen corrents que són induïdes pel camp magnètic del sensor i causen variació de l'amplitud del senyal.
- Per ultrasons. Hi ha un emissor que envia ultrasons al receptor i quan un objecte creua la línia imaginària que creen els ultrasons el receptor no pot rebre el senyal.



Il·lustració 8: Funcionament del sensor de proximitat d'ultrasons.

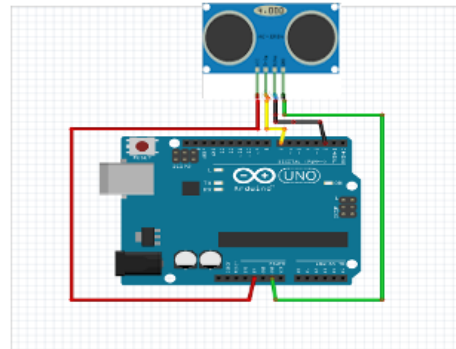
Per fer el meu cotxe de radiocontrol he utilitzat el sensor de proximitat anomenat HC-

⁷ **Fi de cursa o final de carrera:** dispositius electrònics, pneumàtics o mecànics que es posen al final del recorregut o d'un element mòbil per tal d'enviar senyals que puguin modificar l'estat d'un circuit.

SR04, el qual utilitza el mètode d'ultrasons per detectar els objectes. El seu espai de treball és de 2 cm a 400 cm i un angle de detecció de 15°. El sensor HC-SR04 té un preu de 5 € aproximadament.



Il·lustració 9: Sensor de proximitat HC-SR04.



Il·lustració 10: Esquema de la connexió del sensor de proximitat amb la placa Arduino.

3.1.4 Leds

El LED, de les sigles angleses de Light-Emitting Diode és una font de llum de dimensions molt reduïdes. El led funciona proporcionant-li tensió determinada als terminals del dispositiu semiconductor i això provoca que s'emeti energia en forma de fotons que són els encarregats de fer la llum. El dispositiu semiconductor està cobert per un plàstic que normalment està pintat però és simplement una qüestió estètica, ja que el color de la llum que emeten els leds depèn de l'energia que emeten els fotons. Els leds es connecten mitjançant dos borns, el llarg és l'ànode⁸, que es connecta al pol positiu, i el curt és el càtode⁹, que es connecta al pol negatiu.

Per començar amb la història dels leds ens hem de remuntar el 1907 quan el britànic H. J. Round va descobrir l'electroluminiscència com a efecte. Llavors l'any 1927 l'inventor rus Oleg Losev va inventar el primer led, però no li van trobar cap utilitat. Al cap d'unes dècades van trobar la utilitat com a llum de baix consum i no va ser fins al 1968 que l'empresa Monsanto va començar a produir en massa el led.

Actualment els leds són utilitzats per il·luminació domèstica, elements de senyalització i panells d'informació, llums dels cotxes, entre altres.

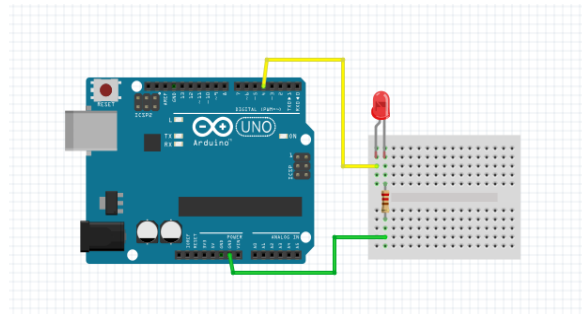
8 **Ànode:** elèctrode que rep càrregues negatives i emet càrregues positives.

9 **Càtode:** elèctrode que ep càrregues positives i emet càrregues negatives.

He utilitzat un total de 8 leds de 5 mm de diàmetre de diferents colors per tal que s'assemblés al màxim als colors de la il·luminació d'un cotxe real. He fet servir dos leds de color blanc per imitar les llums de davant, dos leds de color vermell per les llums de fre i marxa enrere i els quatre leds restants els he posat de groc per fer els intermitents. El total de tots els leds m'ha costat uns 6 €.



Il·lustració 11: Leds de color verd, vermell i groc.



Il·lustració 12: Esquema de la connexió del led amb la placa Arduino.

3.1.5 Cables

Per fer el treball he utilitzat cables fets expressament per les connexions amb plaques Arduino. He utilitzat un conjunt de 40 cables, tant de connexió mascle-mascle com mascle-femella. Un paquet de 40 cables de connexió mascle-mascle val aproximadament 5 €, igual que un paquet de 40 cables de connexió mascle-femella.

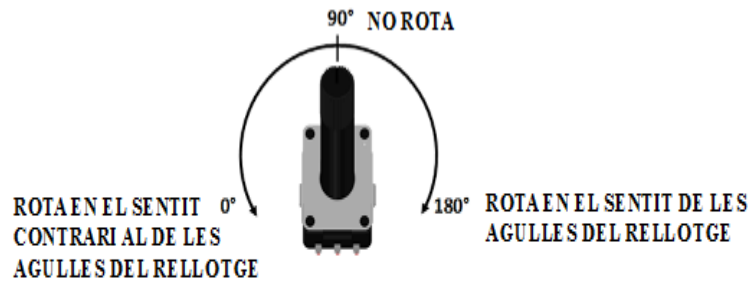


Il·lustració 13: Conjunt de cables.

3.1.6 Motors

Hi ha molts tipus de motors al mercat, segons si es busca potència, velocitat, motors de corrent continu, etc.

Els servomotors estan dissenyats per poder controlar la posició i la velocitat en un eix mitjançant els graus que gira. També es poden quedar parats en un punt desitjat.



Il·lustració 14: Sistema de rotació del servomotor segons els graus.

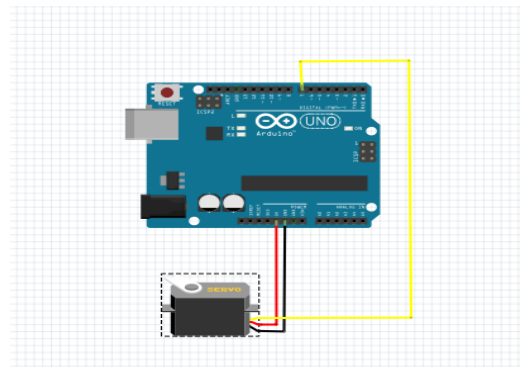
Tot i que la teoria diu que quan el servomotor treballa a 90° està parat, a la pràctica has de buscar el grau exacte segons el servomotor, perquè tenen un marge d'error però està al voltant dels 90°.

Hi ha dos tipus de servomotors els quals es classifiquen segons el seu gir:

- Servomotors de gir limitat. Són servos que poden girar fins a 180°, és a dir, que no poden fer una volta completa.
- Servomotors de rotació contínua. Són els que poden donar una volta sencera, i el que els diferencia de la resta de motors de rotació contínua és que es pot controlar la velocitat i el sentit mitjançant els graus. Jo vaig utilitzar dos servomotors de rotació contínua que s'anomenen MG90S, que tenen un cost de 8 € cada un.



Il·lustració 15: Servomotor.



Il·lustració 16: Esquema de la connexió del servomotor amb la placa Arduino.

3.1.7 Resistències













Les resistències o resistors són dispositius electrònics que serveixen per regular el

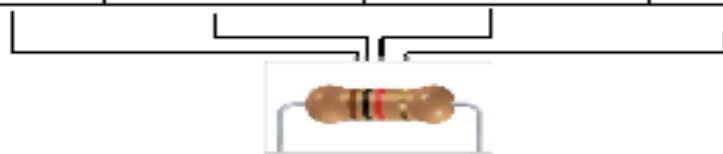
corrent elèctric d'un circuit elèctric.

Hi ha molts tipus de resistències, ja sigui per mides com per diferent valor d'Ohms (Ω). El valor es calcula mitjançant la Llei d'Ohm, la qual relaciona la resistència amb el voltatge i la intensitat.

$$\text{Resistència (R)} = \frac{\text{Voltatge (V)}}{\text{Intensitat (I)}}$$

En les resistències es pot saber el seu valor observant el color de les línies que hi ha pintades a la seva carcassa. Hi ha 4 línies, tres d'elles a un costat i l'última a l'altre costat. Les dues primeres són dues xifres significatives, la tercera és el multiplicador i la quarta és la tolerància. Els colors es classifiquen de la següent manera:

Color	1ª línia	2ª línia	Multiplicador	Tolerància
	0	0	x1	-
	1	1	x10	1,00%
	2	2	x100	2,00%
	3	3	x1000	-
	4	4	x10000	-
	5	5	x100000	-
	6	6	x1000000	-
	7	7	x10000000	-
	8	8	x100000000	-
	9	9	x1000000000	-
	-	-	x0.1	5,00%
	-	-	x0.01	10,00%



Il·lustració 17: Taula de la classificació de les resistències amb colors.

En el meu cas vaig utilitzar resistències pels leds i les vaig posar de 220 Ω i una tolerància del 5,00%, que amb la taula de dalt podem observar que la primera línia era de

color vermell, la segona també, la tercera de color marró i la quarta era de color daurat. Vaig fer servir quatre resistències les quals valen 0,01 € cada una.



Il·lustració 18: Resistència de 220 Ω .

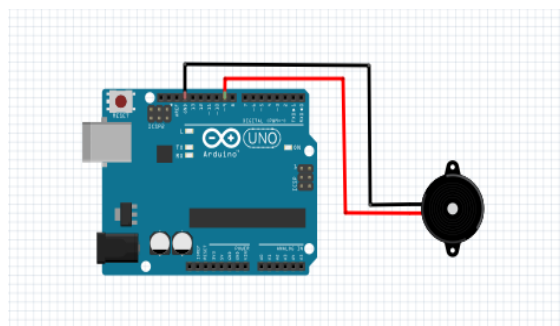
3.1.8 Brunzidor

El brunzidor és un dispositiu electrònic que emet un so en passar corrent entre els seus borns. El so que emet es pot variar canviant la freqüència la qual emet i això fa que soni un to diferent. Jo el vaig utilitzar per fer el clàxon del cotxe i, com que vaig decidir posar música com a clàxon, vaig haver de posar diferents freqüències i temps per tal que sonessin els tons que jo volia i així sonés com la melodia que havia triat, que en aquest cas era la melodia inicial de la sèrie de televisió Els Simpsons.

El brunzidor que vaig utilitzar jo, té un cost de 3 € tot i que n'hi ha de diferents preus segons el volum que poden emetre.



Il·lustració 19: Brunzidor.



Il·lustració 20: Esquema de la connexió del brunzidor amb la placa Arduino.

3.2 Components de muntatge

3.2.1 Metacrilat

El metacrilat, també conegut com a polimetacrilat, és un plàstic acrílic, per tant, és polímer¹⁰. És resistent, amb un alt grau de transparència i resistent a les condicions

¹⁰ **Polímer:** és una combinació de les paraules gregues *poly* (molts) i *mers* (part), per tant, és un material format per moltes parts, en aquest cas molècules unides.

exteriors. El metacrilat es pot tallar, perforar, acolorir i modelar, les quals coses el fan molt útil per la fabricació de claraboies i protectors de llums d'automòbils.

Jo el vaig utilitzar per fer la base del cotxe, així poder veure els components des de sota i poder clavar-los fent forats. Vaig fer servir una planxa la qual vaig dibuixar la forma que volia que tingues i la vaig tallar amb una serra. Un cop tallada la vaig llimar.

Una planxa DIN-A 4 de 4 mm de grossor val aproximadament 4 €. El preu de les planxes de metacrilat va en funció de la seva mida i el seu gruix.



Il·lustració 21: Planxa de metacrilat.

3.2.2 Cargols, femelles i bagues tancades

He utilitzat un conjunt de deu cargols de diferents mides i quatre femelles per poder falcar els diferents elements com la placa Arduino, la carcassa, els adaptadors dels servomotors i els suports pels servomotors. També vaig fer servir dues bagues tancades per subjectar l'eix de les rodes devanteres. Un paquet de 40 cargols de 2'50x10 mm em va costar 2 € igual que un paquet de 10 bagues tancades de 16x6 mm.

3.2.3 Planxa de plàstic

Per fer la carcassa del cotxe vaig fer servir una planxa de plàstic a la qual vàrem donar forma amb una màquina termo fusible i un motlle de guix. Un cop feta la forma de la carcassa la vaig tallar i llimar les vores.



Il·lustració 22: Màquina termo fusible.



Il·lustració 23: Mottle utilitzat en la màquina termo fusible.

Les planxes de plàstic es venen en paquets de 100 unitats i valen 20 €.

3.2.4 Pintura i promotor d'adherència

Per pintar la carcassa perquè no quedés transparent, vaig utilitzar esmalt acrílic en esprai. Abans d'aplicar la pintura s'ha de fregar la superfície que es vol pintar amb un paper de vidre, després s'ha de cobrir amb promotor d'adherència per tal que posteriorment la pintura s'enganxi i no salti. Un cop sec, ja es pot aplicar la pintura. Vaig escollir un esmalt acrílic per carrosseries en aerosol de color blau tempesta.

El promotor d'adherència val uns 10 € i l'esmalt acrílic em va costar 9 € aproximadament.

3.3 Pressupost

Material	Unitats	Preu unitat	Preu total
Placa Arduino UNO	1	20 €	20 €
Mòdul Bluetooth	1	7'86 €	7'86 €
Sensor de proximitat	1	4'50 €	4'50 €
Leds	8	0'80 €	6'40 €
Cables	40	0'13 €	5 €
Servomotors	2	7'99 €	15'98 €
Resistències	4	0'10 €	0'40 €

Brunzidor	1	2'70 €	2'70 €
Planxa de metacrilat	1	3'90 €	3'90 €
Cargols	10	0'05 €	0'50 €
Femelles	4	0'10 €	0'40 €
Bagues tancades	2	0'20 €	0'40 €
Planxa de plàstic	1	0'20 €	0'20 €
Promotor d'adherència	1	9'93 €	9'93 €
Esmalt acrílic	1	8'50 €	8'50 €
		Preu total	86'67 €

* A tots els preus se'ls ha aplicat el 21% d'IVA.

4 EXPLICACIÓ PROCÉS

Per començar, vaig decidir posar-me primer en la programació i després en l'adaptació del cotxe perquè creia que programar i fer que tot funcioni em portaria més temps.

Començant a posar-me amb la programació, em vaig trobar amb un problema a l'hora de connectar la placa Arduino UNO amb l'ordinador, ja que no la detectava a causa d'un problema al *software* de l'ordinador. Com que no sabia com solucionar aquest problema vaig agafar una placa que en aquest cas no era l'original d'Arduino sinó una de “marca blanca”, amb el seu *software* corresponent. Quan vaig voler carregar la placa per veure si era compatible amb l'ordinador, em va sortir un problema de sortida anomenat exit status 5, que buscant pel fòrum d'Arduino el vaig poder solucionar afegint una llibreria al meu programa.

Com que ja tenia uns coneixements previs de programació vaig decidir anar programant per parts i després ajuntar-ho tot, perquè és més fàcil fer diferents programes per cada funció o element i així pots veure si el programa funciona i, si té errors, són més fàcils de detectar quan tens un programa petit que en un de llarg. Fer-ho per parts també t'ajuda a poder fer proves amb un element determinat i que no influeixi als altres elements.

Vaig començar per programar els motors, ja que és la part principal d'un cotxe, i vaig començar programant-los per temps, és a dir, cada x temps anaven endavant, cada x temps enrere, cada x temps girava un perquè anés cap a la dreta i cada x temps l'altre per anar cap a l'esquerra. Vaig fer això al principi per aprendre el funcionament dels motors encara que al final no m'hagi servit de gaire pel Treball, ja que si fas anar un cotxe per temps, no el pots controlar perquè segueix la mateixa rutina en bucle.

Un cop vaig tenir això controlat vaig començar a fer subrutines, que són un mètode per poder canviar d'ordre mentre n'està executant una altra d'anterior. També vaig afegir ordres per poder controlar els motors amb tecles del teclat. Per començar vaig fer servir només les tecles *w* i *s* per anar endavant i enrere respectivament, com es pot observar en l'annex 7.1.1, i llavors vaig afegir les tecles *a* i *d* per girar a la dreta i a l'esquerra respectivament, que es pot veure en l'annex 7.1.2. Vaig triar aquests controls, ja que són els típics controls de moviment dels jocs d'ordinador. Quan vaig afegir aquests dos últims

vaig tenir un error molt bàsic, el qual no em deixava sortir quan entrava a les subrutines que estaven establertes perquè el cotxe girés cap als dos costats. El problema que vaig tenir és que no vaig anul·lar les subrutines anteriors, és a dir, que quan estava a una subrutina no li havia dit que deixes l'altre. Un cop solucionat això vaig afegir la tecla x per aturar-ho tot.

Després vaig decidir afegir leds quan girava i quan anava marxa enrere. El programa on afegeixo els leds es pot observar en l'annex 7.1.3. Un cop vaig tenir això fet vaig decidir apartar-ho i posar-me amb la connexió Bluetooth perquè no l'havia fet servir mai i no sabia com funcionava.

Per aprendre com funcionava em vaig descarregar un programa d'Internet per encendre un led a través del Bluetooth del mòbil, que es troba en l'annex 7.1.5. Un cop vaig aprendre el funcionament del Bluetooth mirant-me el programa, em vaig descarregar una aplicació pel mòbil per poder controlar la placa Arduino amb el mòbil i vaig pujar el programa a la placa per veure si funcionava. En pujar-ho i veure que no funcionava i que el mòdul Bluetooth estava calent, que no és normal, el vaig desconnectar perquè estava preocupat. Llavors vaig veure que el voltatge que havia posat al mòdul Bluetooth era superior al que podia aguantar, i per això s'havia sobrecarregat i no havia funcionat. Vaig canviar la sortida de la placa cap al mòdul del Bluetooth i ja ho vaig tenir solucionat.

Basant-me en el programa d'Internet que havia utilitzat per encendre un led amb el mòbil, vaig començar a fer el programa per poder moure els motors amb el mòbil. Per fer-ho vaig posar les tecles de l'ordinador que havia utilitzat a l'aplicació per tal de convertir-les en les fletxes per poder-los controlar i fer els comandaments més visuals.

Un cop feta la connexió mòbil-placa Arduino, vaig afegir intermitència als leds a l'hora de girar a dreta i a esquerra, llums quan feia marxa enrere i llums davanters independents al moviment, és a dir, que pots decidir quan encendre'ls i quan apagar-los i no afecta l'ordre que estiguis fent anteriorment. Aquest programa el trobem en els annexos 7.1.4 i 7.1.6.

Vaig decidir posar un clàxon amb un bronzidor i, per no fer un clàxon amb un sol soroll vaig decidir posar un tall d'alguna cançó que pogués servir de clàxon. Després de pensar bastant la cançó, vaig agafar la cançó de "Back in Black" d'AC/DC. Per poder-la reproduir

amb el brunzidor vaig buscar la partitura, buscar les freqüències de les notes i llavors posar les freqüències en l'ordre de les notes de la partitura. Un cop posades per ordre vaig posar la durada de les notes i dels intervals de pausa entre elles a oïda, per tal que s'assemblés el màxim a la cançó. Aquest procés, es pot veure en l'annex 7.1.7. La cançó s'assemblava però no m'acabava de convèncer i vaig decidir canviar-la. La primera opció per substituir-la va ser la de "Another Day of Sun" de la pel·lícula musical *La la land*. Un cop la vaig tenir feta amb les freqüències, les notes corresponents i els intervals, no em va acabar d'agradar com perquè fos un clàxon i la vaig canviar. Finalment vaig posar la cançó inicial dels Simpsons, ja que sóc molt fan d'aquesta sèrie i el resultat final del clàxon s'assemblava bastant a la cançó, programa del qual es troba en l'annex 7.1.7. Un cop ajuntat el programa del clàxon amb el programa general vaig afegir-hi una tecla per poder activar-lo des del mòbil.

Per fer el treball diferent i que no fos com tots els cotxes teledirigits vaig pensar que podia afegir-hi un sensor de proximitat a la part frontal del cotxe per tal que si et despistaves i es dirigia cap a una paret, s'aturaria a una certa distància, giraria 90 graus i seguiria recte sense que tu haguessis de controlar-ho, per tal d'evitar xocs.

Per començar a programar amb el sensor de proximitat vaig buscar informació a una pàgina web per saber com funcionava el sensor i els seus comandaments, ja que no l'havia utilitzat mai, i vaig fer proves amb un led. Quan el sensor detectava un obstacle a 50 centímetres el led s'encenia. El programa que vaig treure d'Internet es pot trobar en l'annex 7.1.9. Un cop controlat el programa del sensor de proximitat amb el led, vaig fer el programa perquè quan detectés un obstacle a 50 centímetres els motors es paressin, anessin enrere durant 1 segon i gressin per tal d'encarar-lo fora de l'obstacle. Un cop vaig tenir el moviment amb el sensor sense problemes vaig ajuntar-lo amb el programa general, en l'annex 7.1.10, fent del sensor la part principal perquè així estigués fent l'ordre que estigués fent, si trobava un obstacle a 50 centímetres, faria l'ordre del sensor de proximitat per si sol, sense necessitat de donar-li l'ordre.

Un cop vaig tenir el programa bastant encaminat vaig començar a posar-me amb l'organització i modificació del cotxe per tal de poder-hi posar els motors, mòduls i sensors. El problema que vaig tenir és que hi cabien molt just i no quedava bé estèticament, i vaig decidir fer un cotxe des de zero, fent la carcassa, rodes, base...

Per fer la carcassa del meu cotxe vaig decidir agafar la carcassa d'un cotxe teledirigit que tenia a casa i copiar-la fent un motlle amb guix. El primer problema que vaig tenir va ser que la carcassa per dins no feia la forma del cotxe sinó que tenia tots els encaixos i suports pels cargols. Jo pretenia emplenar la carcassa de guix i així ja tenir la forma però vaig haver de fer el motlle per sobre la carcassa i llavors fer-ne un altre per dins d'aquest motlle.



Il·lustració 24: Cotxe original.



Il·lustració 25: Interior de la carcassa original.

Per fer el motlle exterior vaig cobrir tota la carcassa amb una barreja de guix de tepar esquerdes i aigua. Un cop va estar sec ho vaig començar a treure però, en ser una capa tan fina, a l'hora de treure la part que envoltava els retrovisors se'm va trencar en bastants talls i, al ser guix, no vaig poder ajuntar-los perquè molts es van engrunar.



Il·lustració 26: Primer motlle trencat.

El segon cop que ho vaig provar ho vaig fer amb guix mèdic, el qual al ser amb tires era molt més fàcil de donar la forma de la carcassa i s'adaptava millor i també vaig treure els retrovisors. En poder adaptar millor el guix a la forma de la carcassa vaig poder treure'l

sense problemes.

Un cop vaig tenir el motlle de la part de dalt, el vaig forrar amb paper film perquè no s'ajuntés amb la massa de guix de tapar forats i aigua, la qual la vaig posar dins del motlle per fer el motlle interior. Quan va estar sec, el vaig poder treure amb facilitat mantenint els dos motlles intactes.



Il·lustració 27: Motlles interior i exterior.

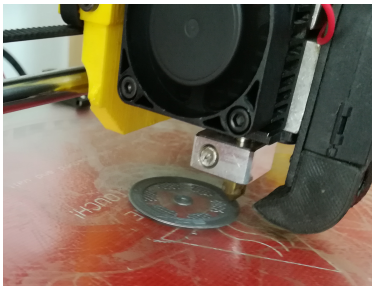
Amb el motlle interior vaig poder fer la carcassa del cotxe amb plàstic a la màquina termo fusible de l'institut, que és una màquina que utilitza escalfor per a donar forma amb un motlle al plàstic. Al principi no sabíem com anava la màquina i vam haver de fer quatre proves per tal d'aconseguir el resultat que volíem. A la primera ens va sortir bé però era massa baix i vam decidir fer-ho un altre cop, aquest amb una base de fusta a sota per elevar-ho, el vam deixar escalfar massa i a l'hora de pujar el motlle van quedar arrugues al plàstic. Després vàrem provar altre cop però sense deixar escalfar tant i en pujar el motlle es va moure i va sortir més elevat per un costat que per l'altre. Finalment vam aconseguir que sortís bé i vam fer dos motlles iguals per poder treballar posteriorment amb un i tenir l'altre de recanvi per si l'altre es malmetia.



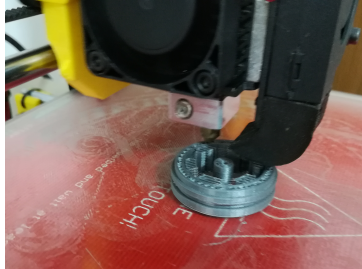
Il·lustració 28: Seguit de proves de la carcassa.

Un cop vaig tenir la carcassa feta vaig posar-me a fer la base del cotxe. Per fer-la vaig agafar una planxa de metacrilat i vaig dibuixar la forma de la carcassa i llavors ho vaig tallar amb una serra i ho vaig polir. Un cop tallada vaig marcar on anirien totes les peces.

Un cop vaig tenir la carcassa i la base bastant encaminats, vaig decidir posar-me amb les rodes. Primerament volia agafar les rodes d'algun cotxe de joguina que tingués per casa però cap encaixava amb els motors. Vaig pensar de dissenyar una petita peça i imprimir-la amb una impressora 3D i així poder adaptar les rodes als motors. Quan vaig pensar això em va venir al cap que podia crear-me jo mateix les rodes dissenyant-les i imprimint-les i així poder fer les mides que més em convinguessin. Per dissenyar-les vaig fer servir el programa SolidWorks, que és un software CAD de les sigles angleses Computer-aided desing (disseny assistit per ordinador). Al principi volia dissenyar les llantes i imprimir-les amb un plàstic dur i després dissenyar els pneumàtics i imprimir-los amb un plàstic més tou i flexible, però després vaig pensar que podia agafar els pneumàtics d'algun cotxe i així només havia de fer les llantes. Per fer les llantes vaig agafar els pneumàtics del mateix cotxe que havia agafat la carcassa per fer el motlle i vaig prendre les mides per poder començar a dissenyar les llantes en 3D. Pensava que dissenyar les llantes seria la part de les rodes que em portaria més feina, ja que has de treballar amb diferents estructures i plans i diferents operacions que poden portar problemes, però ho vaig trobar bastant fàcil. El que em va costar més va ser la mida del forat per encaixar-hi el motor, ja que a part de ser una mesura molt petita i complicada d'agafar exacte, hi ha impressores 3D que tenen un petit error de mesura el qual fa que les mesures amb les quals tu dissenyes es poden veure una mica alterades i les has de contrastar. Vaig haver d'imprimir i modificar les llantes 4 cops, ja que al primer cop la llanta no encaixava amb el pneumàtic. Al segon cop, que sí que vaig aconseguir que encaixes amb el pneumàtic, vaig fer el forat per encaixar el motor massa gran i entrava i sortia molt fàcilment. El tercer cop vaig voler reduir el diàmetre del forat i aquest cop em vaig passar de petit i no entrava i al quart, vaig fer una mesura intermèdia entre les dues anteriors i va entrar amb dificultat però no sortia. El disseny i les vistes d'aquesta última es roben en l'annex 7.2.1. Un cop impreses les quatre llantes vaig acoblar-les als pneumàtics i llavors vaig posar dues d'aquestes als motors.



Il·lustració 29: Primeres capes de la impressió de les llantes.



Il·lustració 30: Impressió de les llantes més avançada.



Il·lustració 31: Llanta impresa.

Fetes les rodes em vaig posar en el disseny i pintura de la carcassa. Vaig posar cinta de pintor a la carcassa als llocs on volia que es quedés transparent per després pintar-ho i que es veiés l'interior en unes determinades zones, com les finestres i els llums i intermitents.

Un cop vaig tenir les zones encintades vaig fer els forats per poder treure el sensor de proximitat.

Per fer-los vaig agafar un misto encès i el vaig pressionar contra el plàstic, fent que es formés un forat. Amb el mateix misto vaig anar fent més gran el forat i donant-li forma amb la mateixa escalfor.



Il·lustració 32: Carcassa amb cinta adhesiva a les zones transparents.



Il·lustració 33: Disseny sobre la carcassa.

Quan vaig tenir els forats fets vaig disposar-me a pintar la carcassa. Primer vaig llimar tota la superfície amb un paper de vidre per tar que la pintura no llisqués. Després vaig posar promotor d'adherència perquè la pintura es pogués adherir i no saltar. Un cop sec el promotor em vaig posar a pintar la carcassa amb un esmalt acrílic en esprai de color

blau tempesta. Vaig fer diverses passades deixant que s'assequés entremig. Un cop pintada, vaig fer els detalls amb un retolador de color platejat.



Il·lustració 34: Carcassa pintada.



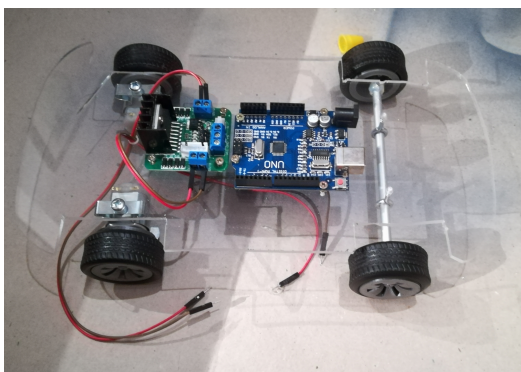
Il·lustració 35: Carcassa pintada amb zones transparents.



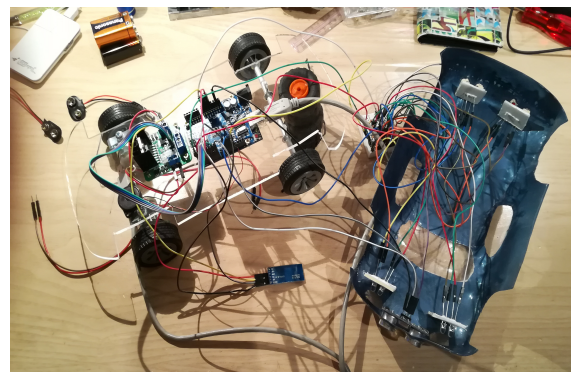
Il·lustració 36: Carcassa pintada amb detalls.

Pintat i decorat el cotxe, vaig començar a muntar tots els components en el cotxe. Vaig començar clavant la placa Arduino al metacrilat amb quatre cargols. Llavors vaig posar les abraçadores i els motors al seu lloc i les vaig collar amb cargols i femelles. També vaig clavar unes bagues tancades que subjectaven l'eix de les rodes davanteres i vaig acoblar les rodes a l'eix.

Amb la placa fixada, els cargols posats i tots els forats fets, vaig començar a posar els cables i a connectar els motors, el sensor de proximitat i el mòdul Bluetooth.



Il·lustració 37: Peces fixades al metacrilat.



Il·lustració 38: Connexió de tots els components.

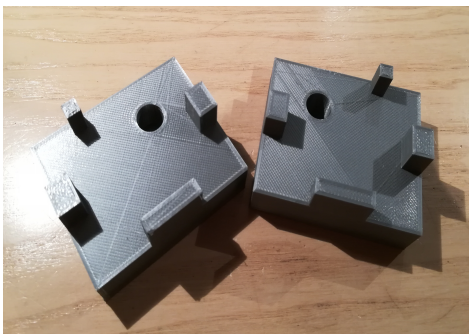
Per fer les connexions i les proves tenia el cotxe elevat, que les rodes dels motors no toquessin a terra, perquè no correués i així podia veure bé el seu funcionament i si fallava alguna cosa.

Quan vaig tenir-ho tot connectat vaig donar-li corrent al cotxe i va ser aquí on va començar un problema bastant greu perquè vàrem tardar molt de temps i moltes proves a trobar perquè fallava. El problema que em va passar va ser que quan donava ordres d'anar endavant o enrere o girar, els motors no em responien però en canvi sí que ho feien els leds. El més curiós va ser que si feia que detectés un obstacle amb el sensor, els motors sí que feien el que el programa deia.

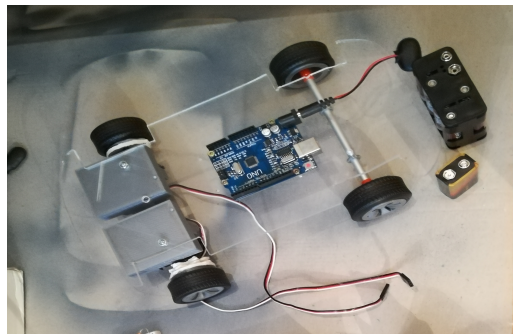
Després de mirar les connexions i tot vàrem veure que el problema estava en el programa i, és que no havia tancat la subrutina del sensor i el programa llegia sempre que si no detectava un obstacle, els motors estaven parats. Solucionat això, vaig voler provar el cotxe però els motors no varen tenir prou potència per moure'l i vaig haver de canviar-los.

Amb el professor vam pensar que posar servomotors solucionaria el problema de la potència, tot i que el cotxe aniria més lent. En voler posar servomotors vaig haver de fer uns suports per poder encaixar-los, ja que les abraçadores que utilitzava pels motors anteriors no em servien. Per dissenyar-los vaig fer servir el programa que també havia utilitzat en el disseny de les llantes, el SolidWorks però, en aquest cas, no tenia cap suport de servos de referència. Vaig agafar un servomotor i prenent mides vaig decidir que faria un suport més gran que el servo i hi posaria unes potes perquè quedés fixat. Vaig dissenyar un suport per la dreta i un per l'esquerra perquè els servos estaven orientats diferent en els dos costats i els suports havien d'estar girats. Podem torbar les vistes i el disseny dels dos suports dels servomotors en l'annex 7.2.2.

Un cop vaig tenir els suports dissenyats i impresos, els vaig collar al metacrilat amb un cargol i una femella.



Il·lustració 39: Suports pels servomotors.

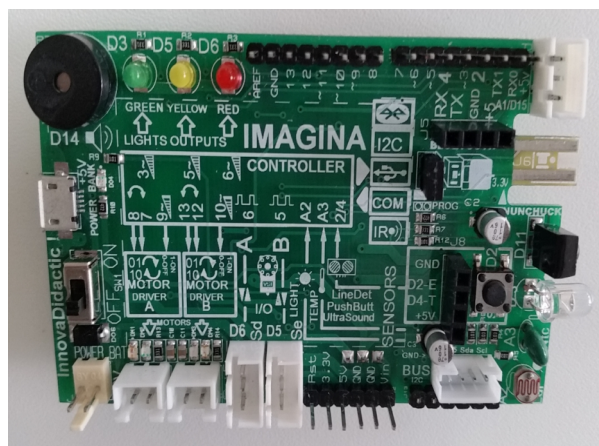


Il·lustració 40: Suports fixats al metacrilat.

Com que havia posat servos vaig haver de canviar el programa perquè tenen un funcionament i també una programació diferent dels motors que havia utilitzat anteriorment. Per fer el programa dels motors vaig recuperar una pràctica que havíem fet l'any anterior a l'assignatura de tecnologia i així vaig poder adaptar-lo al programa que ja tenia fet. Una cosa dels servos que em va costar molt de programar, va ser el punt en què estan aturats, ja que són motors que funcionen mitjançant graus i poden girar a dreta, a esquerra o estar parats. Se sol utilitzar 0° i 180° per girar i 90° per parar, però en cada motor hi pot haver una variació dels graus a causa de la fabricació. Vaig trobar que el punt que paraven els dos era a 91° , val a dir que vaig tenir sort que els dos paressin en els mateixos graus, ja que és complicat que coincideixin.

Quan vaig haver trobat tots els graus que necessitava, vaig connectar els servos a la placa i amb el programa que havia fet, però resulta que els motors no em funcionaven. Vaig repassar el programa per veure si trobava algun error però no el vaig saber trobar.

Per poder resoldre el problema vàrem decidir alimentar els servomotors amb alimentació individual i així fer que l'alimentació dels servomotors fos independent a l'alimentació de la resta de parts electròniques. Vam fer un error molt elemental i va ser que vàrem posar un voltatge superior al que poden aguantar els servos, i això va fer que creméssim les dues sortides que utilitzàvem pels servomotors i com a conseqüència, els servos quedessin malmesos. Per arreglar-ho, vam canviar la placa i els servos i vàrem decidir posar una placa *Imagina Arduino* sobre la placa Arduino. Això en permetia reduir el nombre de cables i fer que el mòdul Bluetooth i els servomotors estiguessin connectats directament a la placa i així poder alimentar-ho tot amb una sola alimentació externa sense cremar-ho.

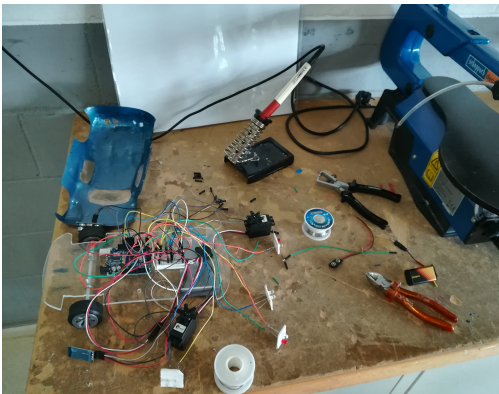


Il·lustració 41: Placa Imagina Arduino.

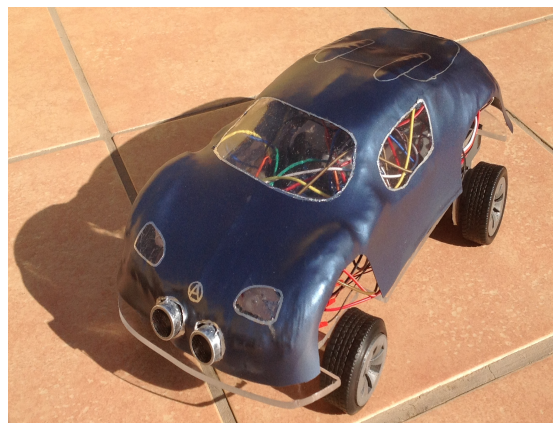
Fet el canvi, ens vam adonar que el Bluetooth funcionava intermitentment, i això feia que no es pogués controlar el cotxe perquè tant estava apagat com encès el mòdul Bluetooth. Vàrem decidir connectar la placa amb l'alimentació de l'ordinador i funcionava bé i feia tot el que havia de fer, però quan vam posar les piles, el Bluetooth es connectava i desconnectava molt sovint i així no es podia controlar el cotxe.

Finalment decidíem modificar el programa, que es pot trobar en l'annex 7.1.13, de manera que quan la placa fes la lectura no detectés en primer lloc el sensor de proximitat i amb això vam poder solucionar un problema que ens havíem trobat que era que hi havia un retard després de donar-li una ordre.

Per finalitzar el treball, vaig soldar tots els cables amb els leds i vaig protegir les soldadures que havia fet. Un cop fet això, vaig muntar el cotxe i el vaig cargolar.



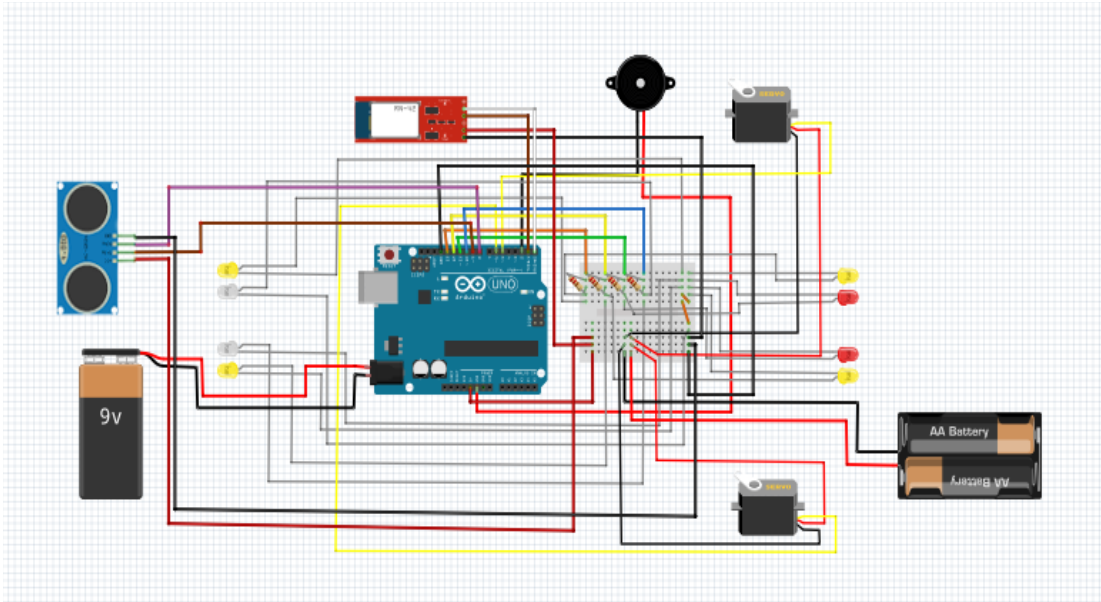
Il·lustració 42: Soldació dels cables.



Il·lustració 43: Cotxe final.

Un cop vaig tenir tot el muntatge fet i el cotxe acabat vaig decidir recrear el circuit que havia utilitzat en el cotxe a l'ordinador. Ho vaig fer amb un programa que es diu Fritzing,

que es un programa per dissenyar circuits elèctrics amb plaques Arduino. És el mateix que havia utilitzat anteriorment en els esquemes de com es connectaven les parts electròniques que he utilitzat.



Il·lustració 44: Esquema del circuit final.

5 CONCLUSIONS

Al principi del treball vaig haver de plantejar-me uns objectius i intentar assolir-los al final del treball. Jo em vaig plantejar cinc objectius dels quals només n'he pogut completar quatre, però estic satisfet, ja que són els quatre més importants i el que no he complert era secundari.

L'objectiu que no he pogut assolir és el de fer un cotxe econòmic i estètic. No l'he pogut assolir perquè el conjunt de tot el material que he utilitzat, que pot ser sí que es podria reduir el cost en alguns materials però tampoc baixaria gaire, costa aproximadament uns 85 €. Això és degut al fet que el cotxe té molts components elèctrics que individualment no costen gaire però junts sí. En el preu final cal afegir les hores invertides en la programació, disseny i muntatge de tots els components, per tant, he arribat a la conclusió que en aquest cas és millor la producció en cadena de les fàbriques de cotxes teledirigits perquè redueixen temps i costos.

Els quatre objectius que he aconseguit assolir m'han portat molts més problemes dels que em pensava que em portarien. M'han portat problemes de programació que abans de fer aquest treball no hagués sabut com resoldre'ls i ara, gràcies al fet que he buscat solucions i a l'ajuda de gent entesa en el tema, he aconseguit resoldre'ls i aprendre d'ells.

Durant tot el trajecte del treball m'he frustrat molts cops en veure que no funcionava i hi havia cops que podien passar diversos dies i no trobava la manera de solucionar alguns problemes. Però a part de tot el que après d'aquest món de la robòtica, em quedo amb la satisfacció de poder aconseguir allò que em vaig proposar a principi de curs i que molts cops m'he plantejat que no funcionaria i ha acabat funcionant.

6 FONTS D'INFORMACIÓ

6.1 Webgrafia

- <http://www.cochesrc.com> <<D'aquesta pàgina web he tret informació sobre les diferents escales dels cotxes teledirigits.>>
Data de consulta: 29/08/2017
- <https://www.prometec.net/indice-tutoriales/>
<<En aquesta he pogut trobar exemples d'alguns programes en els quals m'he basat per fer el meu programa.>>
Data de consulta: 26/04/2017
- <https://www.arduino.cc/>
<<És la pàgina oficial d'Arduino i he trobat informació sobre la placa Arduino UNO i solucions a alguns programes gràcies al seu fòrum.>>
Data de consulta: 05/02/2017
- <http://www.superrobotica.com/servomotores.htm>
<<En aquesta web he obtingut informació sobre els servomotors i altres materials i també m'ha servit per comparar preus.>>
Data de consulta: 18/09/2017
- <http://rcroundup.com/articles/history-of-rc-cars/>
<<He trobat informació general sobre els cotxes de radiocontrol i la seva història.>>
Data de consulta: 27/07/2017
- <https://sites.google.com/site/temasdedisenoymanufactura/arduino>
<<D'aquesta pàgina web he trobat les diferents parts de la placa Arduino.>>
Data de consulta: 21/03/2017
- <https://cdn.sparkfun.com/datasheets/Sensors/Proximity/HCSR04.pdf>
<<Aquest PDF m'ha servit per trobar informació sobre el sensor de proximitat.>>
Data de consulta: 13/05/2017
- <https://programarfacil.com/tutoriales/fragmentos/servomotor-con-arduino/>
<<En aquesta web he pogut trobar informació sobre els servomotors i la seva programació.>>
Data de consulta: 18/09/2017

- <https://www.motorpasion.com/otros/coches-rc-de-gasolina-tu-primera-guia>
<<La informació sobre els cotxes teledirigits de gasolina és tret a d'aquesta pàgina.>>
Data de consulta: 07/03/2017
- <http://www.mwmaterialsworld.com>
<<En aquesta pàgina web he trobat informació sobre el metacrilat i també el seu preu.>>
Data de consulta: 19/10/2017

6.2 Fonts d'il·lustracions

- *Il·lustració 1*: Eix cronològic dels cotxes de radiocontrol. – elaboració pròpia.
- *Il·lustració 2*: Placa Arduino amb les parts senyalades. – <https://www.google.es> (imatges) i elaboració de la senyalització pròpia.
- *Il·lustració 3*: Harald Blåtand. – <https://www.google.es> (imatges).
- *Il·lustració 4*: Jim Kardach. – <https://www.google.es> (imatges).
- *Il·lustració 5*: Lletres Hagal i Berkana de l'alfabet rúnic i logo de Bluetooth. – <https://www.google.es> (imatges).
- *Il·lustració 6*: Mòdul Bluetooth HC-06. – <https://www.google.es> (imatges).
- *Il·lustració 7*: Esquema de la connexió del mòdul Bluetooth amb la placa Arduino. – elaboració pròpia amb Fritzing.
- *Il·lustració 8*: Funcionament del sensor de proximitat d'ultrasons. – <https://www.google.es> (imatges).
- *Il·lustració 9*: Sensor de proximitat HC-SR04. – <https://www.google.es> (imatges).
- *Il·lustració 10*: Esquema de la connexió del sensor de proximitat amb la placa Arduino. – elaboració pròpia amb Fritzing.
- *Il·lustració 11*: Leds de color verd, vermell i groc. – <https://www.google.es> (imatges).
- *Il·lustració 12*: Esquema de la connexió del led amb la placa Arduino. – elaboració pròpia amb Fritzing.
- *Il·lustració 13*: Conjunt de cables. – <https://www.google.es> (imatges).
- *Il·lustració 14*: Sistema de rotació del servomotor segons els graus. – <https://www.google.es> (imatges) i elaboració pròpia de les explicacions.

- *Il·lustració 15:* Servomotor. – <https://www.google.es> (imatges).
- *Il·lustració 16:* Esquema de la connexió del servomotor amb la placa Arduino. – elaboració pròpia amb Fritzing.
- *Il·lustració 17:* Taula de la classificació de les resistències amb colors. – elaboració pròpia amb Fritzing.
- *Il·lustració 18:* Resistència de 220 Ω . – <https://www.google.es> (imatges).
- *Il·lustració 19:* Brunzidor. – <https://www.google.es> (imatges).
- *Il·lustració 20:* Esquema de la connexió del brunzidor amb la placa Arduino. – elaboració pròpia amb Fritzing.
- *Il·lustració 21:* Planxa de metacrilat. – <https://www.google.es> (imatges).
- *Il·lustració 22:* Màquina termo fusible. – elaboració pròpia.
- *Il·lustració 23:* Motlle utilitzat en la màquina termo fusible.
- *Il·lustració 24:* Cotxe original. – elaboració pròpia.
- *Il·lustració 25:* Interior de la carcassa original. – elaboració pròpia.
- *Il·lustració 26:* Primer motlle trencat. – elaboració pròpia.
- *Il·lustració 27:* Motlles interior i exterior. – elaboració pròpia.
- *Il·lustració 28:* Seguit de proves de la carcassa. – elaboració pròpia.
- *Il·lustració 29:* Primeres capes de la impressió de les llantes. – elaboració pròpia.
- *Il·lustració 30:* Impressió de les llantes més avançada. – elaboració pròpia.
- *Il·lustració 31:* Llanta impresa. – elaboració pròpia.
- *Il·lustració 32:* Carcassa amb cinta adhesiva a les zones transparents. – elaboració pròpia.
- *Il·lustració 33:* Disseny sobre la carcassa. – elaboració pròpia.
- *Il·lustració 34:* Carcassa pintada. – elaboració pròpia.
- *Il·lustració 35:* Carcassa pintada amb zones transparents. – elaboració pròpia.
- *Il·lustració 36:* Carcassa pintada amb detalls. – elaboració pròpia.
- *Il·lustració 37:* Peces fixades al metacrilat. – elaboració pròpia.
- *Il·lustració 38:* Connexió de tots els components. – elaboració pròpia.
- *Il·lustració 39:* Suports pels servomotors. – elaboració pròpia.
- *Il·lustració 40:* Suports fixats al metacrilat. – elaboració pròpia.
- *Il·lustració 41:* Placa Imagina Arduino.– <https://www.google.es> (imatges).
- *Il·lustració 42:* Soldació dels cables. – elaboració pròpia.

- *Il·lustració 43*: Cotxe final. – elaboració pròpia.
- *Il·lustració 44*: Esquema del circuit final. – elaboració pròpia amb Fritzing.
- *Il·lustració 46*: Vistes de la llanta. – elaboració pròpia amb SolidWorks.
- *Il·lustració 46*: Llanta amb 3D. – elaboració pròpia amb SolidWorks.
- *Il·lustració 47*: Vistes del suport pel servomotor dret. – elaboració pròpia amb SolidWorks.
- *Il·lustració 48*: Suport del servomotor dret amb 3D. – elaboració pròpia amb SolidWorks.
- *Il·lustració 49*: Vistes del suport pel servomotor esquerre. – elaboració pròpia amb SolidWorks.
- *Il·lustració 50*: Suport del servomotor esquerre en 3D. – elaboració pròpia amb SolidWorks.

7 ANNEXOS

7.1 Evolució del programari

7.1.1 Primera prova de control dels motors amb tecles del teclat

```

int motor=10; //introduïm el positiu del motor de la dreta en el pin 10.
int a=11; //introduïm el negatiu del motor de la dreta en el pin 11.
int motor2=5; //introduïm el positiu del motor de l'esquerra en el pin 5.
int b=6; //introduïm el negatiu del motor de l'esquerra en el pin 6.
int tecla; //introduïm la variable "tecla" per poder controlar els motors amb les tecles.
void setup() {
  // put your setup code here, to run once:
  pinMode(motor,OUTPUT);
  pinMode(a,OUTPUT);
  pinMode(motor2,OUTPUT);
  pinMode(b,OUTPUT);
  Serial.begin(9600);
}

void loop() {
  // put your main code here, to run repeatedly:
  digitalWrite(a,LOW); //el negatiu del motor de la dreta està apagat.
  digitalWrite(b,LOW); //el negatiu del motor de l'esquerra està apagat.
  tecla=Serial.read();
  if (tecla=='w'){
    digitalWrite(motor,HIGH);
    digitalWrite(motor2,HIGH);
  }
  if (tecla=='s'){
    digitalWrite(motor,LOW);
    digitalWrite(motor2,LOW);
  }
  if (tecla=='d'){
    digitalWrite(motor,LOW);
    digitalWrite(motor2,HIGH);
  }
  if (tecla=='a'){
    digitalWrite(motor,HIGH);
    digitalWrite(motor2,LOW);
  }
}

```

7.1.2 Prova del programa per girar amb les tecles

```

int motor=10; //introduïm el positiu del motor de la dreta en el pin 10.
int a=11; //introduïm el negatiu del motor de la dreta en el pin 11.
int motor2=5; //introduïm el positiu del motor de l'esquerra en el pin 5.

```

```

int b=6; //introduïm el negatiu del motor de l'esquerra en el pin 6.
int tecla; //introduïm la variable "tecla" per poder controlar els motors amb les tecles.
void setup() {
  // put your setup code here, to run once:
  pinMode(motor,OUTPUT);
  pinMode(a,OUTPUT);
  pinMode(motor2,OUTPUT);
  pinMode(b,OUTPUT);
  Serial.begin(9600);
}

void loop() {
  // put your main code here, to run repeatedly:
  digitalWrite(a,LOW); //el negatiu del motor de la dreta està apagat.
  digitalWrite(b,LOW); //el negatiu del motor de l'esquerra està apagat.
  tecla=Serial.read();
  if (tecla=='w'){
    digitalWrite(motor,HIGH);
    digitalWrite(motor2,HIGH);
  }
  if (tecla=='s'){
    digitalWrite(motor,LOW);
    digitalWrite(motor2,LOW);
  }
  if (tecla=='d'){
    digitalWrite(motor,LOW);
    digitalWrite(motor2,HIGH);
  }
  if (tecla=='a'){
    digitalWrite(motor,HIGH);
    digitalWrite(motor2,LOW);
  }
}

```

7.1.3 Introducció de leds al programa anterior

```

int motor=10;
int a=11;
int motor2=5;
int b=6;
int i;
int tecla;
int led1= 13;
int led2=12;
void setup() {
  // put your setup code here, to run once:
  pinMode(motor,OUTPUT);
  pinMode(a,OUTPUT);
  pinMode(motor2,OUTPUT);
  pinMode(b,OUTPUT);
  pinMode(led1, OUTPUT);
  pinMode(led2, OUTPUT);
}

```

```
Serial.begin(9600);
}

void loop() {
  // put your main code here, to run repeatedly:
  tecla=Serial.read();

  if (tecla=='w'){
    digitalWrite(motor,HIGH);
    digitalWrite(motor2,HIGH);
    digitalWrite(a,LOW);
    digitalWrite(b,LOW);
    digitalWrite(led1,LOW);
    digitalWrite(led2,LOW);
  }
  if (tecla=='x'){
    digitalWrite(motor,LOW);
    digitalWrite(motor2,LOW);
    digitalWrite(a,LOW);
    digitalWrite(b,LOW);
    digitalWrite(led1,LOW);
    digitalWrite(led2,LOW);
  }
  if (tecla=='d'){
    digitalWrite(motor,LOW);
    digitalWrite(motor2,HIGH);
    digitalWrite(a,LOW);
    digitalWrite(b,LOW);
    digitalWrite(led1,HIGH);
    digitalWrite(led2,LOW);
  }
  if (tecla=='a'){
    digitalWrite(motor,HIGH);
    digitalWrite(motor2,LOW);
    digitalWrite(a,LOW);
    digitalWrite(b,LOW);
    digitalWrite(led1,LOW);
    digitalWrite(led2,HIGH);
  }
  if (tecla=='s'){
    digitalWrite(motor,LOW);
    digitalWrite(motor2,LOW);
    digitalWrite(a,HIGH);
    digitalWrite(b,HIGH);
    digitalWrite(led1,HIGH);
    digitalWrite(led2,HIGH);
  }
}
```

7.1.4 Programa anterior amb intermitència als leds

```
int motor=10;
int a=11;
int motor2=5;
int b=6;
int tecla;
int led1= 13;
int led2=12;
void setup() {
  // put your setup code here, to run once:
  pinMode(motor,OUTPUT);
  pinMode(a,OUTPUT);
  pinMode(motor2,OUTPUT);
  pinMode(b,OUTPUT);
  pinMode(led1, OUTPUT);
  pinMode(led2, OUTPUT);
  Serial.begin(9600);
}

void loop() {
  // put your main code here, to run repeatedly:
  tecla=Serial.read();

  if (tecla=='w'){
    digitalWrite(motor,HIGH);
    digitalWrite(motor2,HIGH);
    digitalWrite(a,LOW);
    digitalWrite(b,LOW);
    digitalWrite(led1,LOW);
    digitalWrite(led2,LOW);
  }
  if (tecla=='x'){
    digitalWrite(motor,LOW);
    digitalWrite(motor2,LOW);
    digitalWrite(a,LOW);
    digitalWrite(b,LOW);
    digitalWrite(led1,LOW);
    digitalWrite(led2,LOW);
  }
  if (tecla=='d'){
    digitalWrite(motor,LOW);
    digitalWrite(motor2,HIGH);
    digitalWrite(a,LOW);
    digitalWrite(b,LOW);
    interdret();
  }
  if (tecla=='a'){
    digitalWrite(motor,HIGH);
    digitalWrite(motor2,LOW);
```

```

digitalWrite(a,LOW);
digitalWrite(b,LOW);
digitalWrite(led1,LOW);
digitalWrite(led2,HIGH);
}
if (tecla=='s'){
  digitalWrite(motor,LOW);
  digitalWrite(motor2,LOW);
  digitalWrite(a,HIGH);
  digitalWrite(b,HIGH);
  digitalWrite(led1,HIGH);
  digitalWrite(led2,HIGH);
}
}
}
void interdret(){
while(tecla=='d'){
  digitalWrite(led1,LOW);
  digitalWrite(led2,LOW);
  delay(500);
  digitalWrite(led1,HIGH);
  delay (500);
  tecla=Serial.read();
}
}
}

```

7.1.5 Programa d'iniciació per entendre el Bluetooth

```

#include <SoftwareSerial.h>
SoftwareSerial BT(10, 11);
// creates a "virtual" serial port/UART
// connect BT module TX to D10
// connect BT module RX to D11
// connect BT Vcc to 5V, GND to GND
void setup()
{
  // set digital pin to control as an output
  pinMode(13, OUTPUT);
  BT.begin(9600);
  BT.println("Hello from Arduino");
}
char a;
void loop()
{
  if (BT.available())
  // if text arrived in from BT serial...
  {
    a=(BT.read());
    if (a=='1')
    {
      digitalWrite(13, HIGH);
    }
  }
}

```

```

    BT.println("LED on");
  }
  if (a=='2')
  {
    digitalWrite(13, LOW);
    BT.println("LED off");
  }
  if (a=='?')
  {
    BT.println("Send '1' to turn LED on");
    BT.println("Send '2' to turn LED on");
  }
}
}
}

```

7.1.6 Aplicació del mòdul Bluetooth i llums frontals en el programa fet anteriorment

```

#include <SoftwareSerial.h>
SoftwareSerial BT(10, 11);
// creates a "virtual" serial port/UART
// connect BT module TX to D10
// connect BT module RX to D11
// connect BT Vcc to 5V, GND to GND
int motor=3;
int b=4;
int motor2=5;
int c=6;
int led1=13;
int led2=12;
int led3=8;
int led4=9;

void setup()
{
  // set digital pin to control as an output
  pinMode(motor,OUTPUT);
  pinMode(c,OUTPUT);
  pinMode(motor2,OUTPUT);
  pinMode(b,OUTPUT);
  pinMode(led1, OUTPUT);
  pinMode(led2, OUTPUT);
  pinMode(led3, OUTPUT);
  pinMode(led4, OUTPUT);
  // set the data rate for the SoftwareSerial port
  BT.begin(9600);
  // Send test message to other device
  BT.println("Hello from Arduino");
}
char a; // stores incoming character from other device
void loop()

```

```
{
  if (BT.available())
  // if text arrived in from BT serial...
  {
    a=(BT.read());
  }
  if (a=='w'){
    digitalWrite(motor,HIGH);
    digitalWrite(motor2,HIGH);
    digitalWrite(b,LOW);
    digitalWrite(c,LOW);
    digitalWrite(led1,LOW);
    digitalWrite(led2,LOW);
    digitalWrite(led3,LOW);
  }
  if (a=='x'){
    digitalWrite(motor,LOW);
    digitalWrite(motor2,LOW);
    digitalWrite(b,LOW);
    digitalWrite(c,LOW);
    digitalWrite(led1,LOW);
    digitalWrite(led2,LOW);
    digitalWrite(led3,LOW);
  }
  if (a=='d'){
    digitalWrite(led1,HIGH);
    digitalWrite(led2,LOW);
    digitalWrite(led3,LOW);
    delay(500);
    digitalWrite(led1,LOW);
    digitalWrite(led2,LOW);
    digitalWrite(led3,LOW);
    delay(500);
    digitalWrite(motor,HIGH);
    digitalWrite(motor2,LOW);
    digitalWrite(b,LOW);
    digitalWrite(c,LOW);
  }
  if (a=='a'){
    digitalWrite(led1,LOW);
    digitalWrite(led2,HIGH);
    digitalWrite(led3,LOW);
    delay(500);
    digitalWrite(led1,LOW);
    digitalWrite(led2,LOW);
    digitalWrite(led3,LOW);
    delay(500);
    digitalWrite(motor,LOW);
    digitalWrite(motor2,HIGH);
    digitalWrite(b,LOW);
    digitalWrite(c,LOW);
  }
}
```

```

}
if (a=='s'){
  digitalWrite(motor,LOW);
  digitalWrite(motor2,LOW);
  digitalWrite(b,HIGH);
  digitalWrite(c,HIGH);
  digitalWrite(led1,LOW);
  digitalWrite(led2,LOW);
  digitalWrite(led3,HIGH);
}
if (a=='p'){
  digitalWrite(led4,HIGH);
}
if (a=='l'){
  digitalWrite(led4,LOW);
}
}
}

```

7.1.7 Programa de la cançó “Back in Black” d'ACDC pel clàxon

```

int so=9;
void setup(){
  pinMode(so, OUTPUT);
}
//do 1046.5
//do# 1108.7
//re 1174.7
//re# 1244.5
//mi 1318.5
//fa 1396.9
//fa# 1480
//sol 1568
//sol# 1661.2
//la 1760
//la# 1864.7
//si 1975.5
//do' 2093
//do#' 2217.5
//re' 2349.3
//re#' 2489
//mi' 2637
//fa' 2793.8
//fa#' 2960
//sol' 3136
//sol#' 3322.4
//la' 3520
//la#' 3729.3
//si' 3951.1

void loop(){

```



```
tone (so,2637,300); //mi'  
  delay(1500);  
tone (so,2349.3,200); //re'  
  delay(200);  
tone (so,2349.3,200); //re'  
  delay(200);  
tone (so,2349.3,200); //re'  
  delay(1000);  
tone (so,1760,200); //la  
  delay(200);  
tone (so,1760,200); //la  
  delay(200);  
tone (so,1760,200); //la  
  delay(1000);  
tone (so,3136,200); //sol'  
  delay(200);  
tone (so,2637,200); //mi'  
  delay(200);  
tone (so,2349.3,200); //re'  
  delay(200);  
tone (so,1975.5,200); //si  
  delay(200);  
tone (so,1975.5,200); //si  
  delay(200);  
tone (so,1760,200); //do#  
  delay(200);  
tone (so,1568,200); //re  
  delay(200);  
tone (so,2637,300); //mi'  
  delay(1000);  
tone (so,2349.3,200); //re'  
  delay(200);  
tone (so,2349.3,200); //re'  
  delay(200);  
tone (so,2349.3,200); //re'  
  delay(800);  
tone (so,1760,200); //la  
  delay(200);  
tone (so,1760,200); //la  
  delay(200);  
tone (so,1760,200); //la  
  delay(700);  
tone (so,1975.5,250); //si  
  delay(250);  
tone (so,1661.2,400); //sol#  
  delay(500);  
tone (so,1975.5,250); //si  
  delay(250);  
tone (so,1760,400); //la  
  delay(500);  
tone (so,1975.5,250); //si
```

```

    delay(250);
tone (so,1864.7,400); //la#
    delay(500);
tone (so,1975.5,250); //si
    delay(200);
tone (so,1975.5,400); //si
    delay(500);
}

```

7.1.8 Programa de la cançó inicial dels Simpsons pel clàxon

```

int so=9;
void setup(){
  pinMode(so, OUTPUT);
}
//do 1046.5
//do# 1108.7
//re 1174.7
//re# 1244.5
//mi 1318.5
//fa 1396.9
//fa# 1480
//sol 1568
//sol# 1661.2
//la 1760
//la# 1864.7
//si 1975.5
//do' 2093
//do#' 2217.5
//re' 2349.3
//re#' 2489
//mi' 2637
//fa' 2793.8
//fa#' 2960
//sol' 3136
//sol#' 3322.4
//la' 3520
//la#' 3729.3
//si' 3951.1

void loop(){
//do mi fa# la sol mi do la fa# fa# fa# sol fa# fa# fa# so sib

tone (so,2093,600); //do'
  delay(550);
tone (so,2637,350); //mi'
  delay(400);
tone (so,2960,300); //fa#'
  delay(400);
tone (so,3520,300); //la'

```

```
    delay(300);
tone (so,3136,500); //sol'
    delay(600);
tone (so,2637,400); //mi'
    delay(400);
tone (so,2093,450); //do'
    delay(500);
tone (so,1760,400); //la'
    delay(500);
tone (so,1480,300); //fa#
    delay(300);
tone (so,1480,300); //fa#
    delay(300);
tone (so,1480,300); //fa#
    delay(300);
tone (so,1568,300); //sol
    delay(1500);
tone (so,1480,300); //fa#
    delay(300);
tone (so,1480,300); //fa#
    delay(300);
tone (so,1480,300); //fa#
    delay(300);
tone (so,1568,400); //sol
    delay(400);
tone (so,1864.7,500); //sib
    delay(500);
tone (so,2093,300); //do'
    delay(300);
tone (so,2093,300); //do'
    delay(300);
tone (so,2093,300); //do'
    delay(1500);
```

7.1.9 Programa d'iniciació per entendre el sensor de proximitat

```
int trigsensor=13;
int echosensor=12;
int led=2;

void setup()
{
  Serial.begin (9600);
  pinMode(trigsensor, OUTPUT);
  pinMode(echosensor, INPUT);
  pinMode(led, OUTPUT);
}

void loop()
{
  long duracion, distancia ;
  digitalWrite(trigsensor, LOW);
```

```

delayMicroseconds(2);
digitalWrite(trigsensor, HIGH);
delayMicroseconds(10);
digitalWrite(trigsensor, LOW);
duracion = pulseIn(echosensor, HIGH) ;
distancia = duracion / 2 / 29.1 ;
Serial.println(String(distancia) + " cm.");
int Limit = 50 ;
if ( distancia < Limit)
  digitalWrite ( led , HIGH) ;
else
  digitalWrite( led , LOW) ;
  delay (500) ;
}

```

7.1.10 Aplicació del sensor de proximitat i música al programa anterior

```

#include <SoftwareSerial.h>
SoftwareSerial BT(10, 11);
//#include <EducationShield.h>
//#include <pitches.h>
// creates a "virtual" serial port/UART
// connect BT module TX to D10
// connect BT module RX to D11
// connect BT Vcc to 5V, GND to GND
int motor=3;
int b=4;
int motor2=5;
int c=6;
int led1=13; // int. dreta
int led2=12; //int. esquerre
int led3=0; //darrere
int led4=1; //lums
int so=8; //(negre)
int trigsensor=2;
int echosensor=7;

void setup()
{
  // set digital pin to control as an output
  pinMode(motor,OUTPUT);
  pinMode(c,OUTPUT);
  pinMode(motor2,OUTPUT);
  pinMode(b,OUTPUT);
  pinMode(led1, OUTPUT);
  pinMode(led2, OUTPUT);
  pinMode(led3, OUTPUT);
  pinMode(led4, OUTPUT);
  pinMode(trigsensor, OUTPUT);
  pinMode(echosensor, INPUT);
  pinMode(so, OUTPUT);
}

```

```
// set the data rate for the SoftwareSerial port
BT.begin(9600);
// Send test message to other device
BT.println("Hello from Arduino");
}
char a;
void loop()
{
if (BT.available())
{
a=(BT.read());
}
long duracion, distancia ;
digitalWrite(trigsensor, LOW);
delayMicroseconds(2);
digitalWrite(trigsensor, HIGH);
delayMicroseconds(10);
digitalWrite(trigsensor, LOW);
duracion = pulseIn(echosensor, HIGH) ;
distancia = duracion / 2 / 29.1 ;
Serial.println(String(distancia) + " cm." );
int Limit = 50 ;
if ( distancia < Limit){
digitalWrite (motor, LOW);
digitalWrite (motor2, LOW);
digitalWrite (b, LOW);
digitalWrite (c, LOW);
delay(1000);
digitalWrite (motor, LOW);
digitalWrite (motor2, LOW);
digitalWrite (b, HIGH);
digitalWrite (c, HIGH);
delay(1000);
digitalWrite (motor, HIGH);
digitalWrite (motor2, LOW);
digitalWrite (c, HIGH);
digitalWrite (b, LOW);
delay(1000);
}
else
digitalWrite(motor,HIGH);
digitalWrite(motor2,HIGH);
digitalWrite(b,LOW);
digitalWrite(c,LOW);
digitalWrite(led1,LOW);
digitalWrite(led2,LOW);
digitalWrite(led3,LOW);
if (a=='w'){
digitalWrite(motor,HIGH);
digitalWrite(motor2,HIGH);
digitalWrite(b,LOW);
}
```

```
digitalWrite(c,LOW);
digitalWrite(led1,LOW);
digitalWrite(led2,LOW);
digitalWrite(led3,LOW);
}
if (a=='x'){
digitalWrite(motor,LOW);
digitalWrite(motor2,LOW);
digitalWrite(b,LOW);
digitalWrite(c,LOW);
digitalWrite(led1,LOW);
digitalWrite(led2,LOW);
digitalWrite(led3,LOW);
}
if (a=='d'){
digitalWrite(led1,HIGH);
digitalWrite(led2,LOW);
digitalWrite(led3,LOW);
delay(500);
digitalWrite(led1,LOW);
digitalWrite(led2,LOW);
digitalWrite(led3,LOW);
delay(500);
digitalWrite(motor,HIGH);
digitalWrite(motor2,LOW);
digitalWrite(b,LOW);
digitalWrite(c,LOW);
}
if (a=='a'){
digitalWrite(led1,LOW);
digitalWrite(led2,HIGH);
digitalWrite(led3,LOW);
delay(500);
digitalWrite(led1,LOW);
digitalWrite(led2,LOW);
digitalWrite(led3,LOW);
delay(500);
digitalWrite(motor,LOW);
digitalWrite(motor2,HIGH);
digitalWrite(b,LOW);
digitalWrite(c,LOW);
}
if (a=='s'){
digitalWrite(motor,LOW);
digitalWrite(motor2,LOW);
digitalWrite(b,HIGH);
digitalWrite(c,HIGH);
digitalWrite(led1,LOW);
digitalWrite(led2,LOW);
digitalWrite(led3,HIGH);
}
}
```

```
if (a=='p'){
  digitalWrite(led4,HIGH);
}
if (a=='l'){
  digitalWrite(led4,LOW);
}
if ( distancia < Limit){
  digitalWrite (motor, LOW);
  digitalWrite (motor2, LOW);
  digitalWrite (b, LOW);
  digitalWrite (c, LOW);
  delay(1000);
  digitalWrite (motor, LOW);
  digitalWrite (motor2, LOW);
  digitalWrite (b, HIGH);
  digitalWrite (c, HIGH);
  delay(1000);
  digitalWrite (motor, HIGH);
  digitalWrite (motor2, LOW);
  digitalWrite (c, HIGH);
  digitalWrite (b, LOW);
  delay(1000);
}
if (a=='u'){
  //do 1046.5
  //do# 1108.7
  //re 1174.7
  //re# 1244.5
  //mi 1318.5
  //fa 1396.9
  //fa# 1480
  //sol 1568
  //sol# 1661.2
  //la 1760
  //la# 1864.7
  //si 1975.5
  //do' 2093
  //do#' 2217.5
  //re' 2349.3
  //re#' 2489
  //mi' 2637
  //fa' 2793.8
  //fa#' 2960
  //sol' 3136
  //sol#' 3322.4
  //la' 3520
  //la#' 3729.3
  //si' 3951.1

  tone (so,2093,600); //do'
  delay(550);
```

```
tone (so,2637,350); //mi'  
  delay(400);  
tone (so,2960,300); //fa#  
  delay(400);  
tone (so,3520,300); //la'  
  delay(300);  
tone (so,3136,500); //sol'  
  delay(600);  
tone (so,2637,400); //mi'  
  delay(400);  
tone (so,2093,450); //do'  
  delay(500);  
tone (so,1760,400); //la'  
  delay(500);  
tone (so,1480,300); //fa#  
  delay(300);  
tone (so,1480,300); //fa#  
  delay(300);  
tone (so,1480,300); //fa#  
  delay(300);  
tone (so,1568,300); //sol  
  delay(1500);  
tone (so,1480,300); //fa#  
  delay(300);  
tone (so,1480,300); //fa#  
  delay(300);  
tone (so,1480,300); //fa#  
  delay(300);  
tone (so,1568,400); //sol  
  delay(400);  
tone (so,1864.7,500); //sib  
  delay(500);  
tone (so,2093,300); //do'  
  delay(300);  
tone (so,2093,300); //do'  
  delay(300);  
tone (so,2093,300); //do'  
  delay(1500);  
}  
}
```

7.1.11 Programa d'introducció als servomotors

```
#include <Servo.h>  
Servo md;  
Servo me;  
  
void setup() {  
  md.attach(6);  
  me.attach(7);  
}
```



```

void loop() {
  md.write(0);
  me.write(0);
  delay(3000);
  md.write(91);
  me.write(91);
  delay(3000);
  md.write(180);
  me.write(180);
  delay(3000);
  md.write(270);
  me.write(270);
  delay(3000);
}
//md davant=180
//me davant=0
//md enrere=0
//me enrere=180

```

7.1.12 Programa final amb servomotors

```

#include <SoftwareSerial.h>
#include <Servo.h>
SoftwareSerial BT(10, 11);
//#include <EducationShield.h>
//#include <pitches.h>
Servo md;
Servo me;
// creates a "virtual" serial port/UART
// connect BT module TX to D10
// connect BT module RX to D11
// connect BT Vcc to 5V, GND to GND
int led1=13; // int. dreta
int led2=12; //int. esquerra
int led3=0; //darrera
int led4=1; //lums
int so=8; //(vermell)
int trigsensor=2;
int echosensor=7;

void setup()
{
  // set digital pin to control as an output
  md.attach(5);
  me.attach(6);
  pinMode(led1, OUTPUT);
  pinMode(led2, OUTPUT);
  pinMode(led3, OUTPUT);
  pinMode(led4, OUTPUT);
}

```

```

pinMode(trigsensor, OUTPUT);
pinMode(echosensor, INPUT);
pinMode(so, OUTPUT);
// set the data rate for the SoftwareSerial port
BT.begin(9600);
// Send test message to other device
BT.println("Hello from Arduino");
}
char a; // stores incoming character from other device
void loop()
{
if (BT.available())
{
a=(BT.read());
}
long duracion, distancia ;
digitalWrite(trigsensor, LOW);
delayMicroseconds(2);
digitalWrite(trigsensor, HIGH);
delayMicroseconds(10);
digitalWrite(trigsensor, LOW);
duracion = pulseIn(echosensor, HIGH) ;
distancia = duracion / 2 / 29.1 ;
Serial.println(String(distancia) + " cm." );
int Limit = 50 ;
if ( distancia < Limit){
me.write(91);
md.write(91);
delay (1000); //els dos frenen
me.write(180);
md.write(0);
delay (1000); //van marxa enrere
me.write(0);
md.write(91);
delay (1000); //un gira i l'altre no
}
else
{
me.write(0);
md.write(180); //els dos van endevan
digitalWrite(led1,LOW);
digitalWrite(led2,LOW);
digitalWrite(led3,LOW);
}
if (a=='w'){
me.write(0);
md.write(180); //els dos van endevan
digitalWrite(led1,LOW);
digitalWrite(led2,LOW);
digitalWrite(led3,LOW);
}
}

```

```
if (a=='x'){
  me.write(91);
  md.write(91);//els dos parats
  digitalWrite(led1,LOW);
  digitalWrite(led2,LOW);
  digitalWrite(led3,LOW);
}
if (a=='d'){
  digitalWrite(led1,HIGH);
  digitalWrite(led2,LOW);
  digitalWrite(led3,LOW);
  delay(500);
  digitalWrite(led1,LOW);
  digitalWrite(led2,LOW);
  digitalWrite(led3,LOW);
  delay(500);
  me.write(91);
  md.write(180);//el de la dreta roda l'altre no
}
if (a=='a'){
  digitalWrite(led1,LOW);
  digitalWrite(led2,HIGH);
  digitalWrite(led3,LOW);
  delay(500);
  digitalWrite(led1,LOW);
  digitalWrite(led2,LOW);
  digitalWrite(led3,LOW);
  delay(500);
  me.write(0);
  md.write(91);//el de l'esquerra gira l'altre no
}
if (a=='s'){
  me.write(180);
  md.write(0);//els dos van enrere
  digitalWrite(led1,LOW);
  digitalWrite(led2,LOW);
  digitalWrite(led3,HIGH);
}
if (a=='p'){
  digitalWrite(led4,HIGH);
}
if (a=='l'){
  digitalWrite(led4,LOW);
}
if (a=='u'){
  //do 1046.5
  //do# 1108.7
  //re 1174.7
  //re# 1244.5
  //mi 1318.5
  //fa 1396.9
```

```
//fa# 1480
//sol 1568
//sol# 1661.2
//la 1760
//la# 1864.7
//si 1975.5
//do' 2093
//do#' 2217.5
//re' 2349.3
//re#' 2489
//mi' 2637
//fa' 2793.8
//fa#' 2960
//sol' 3136
//sol#' 3322.4
//la' 3520
//la#' 3729.3
//si' 3951.1
```

```
tone (so,2093,600); //do'
  delay(550);
tone (so,2637,350); //mi'
  delay(400);
tone (so,2960,300); //fa#'
  delay(400);
tone (so,3520,300); //la'
  delay(300);
tone (so,3136,500); //sol'
  delay(600);
tone (so,2637,400); //mi'
  delay(400);
tone (so,2093,450); //do'
  delay(500);
tone (so,1760,400); //la'
  delay(500);
tone (so,1480,300); //fa#
  delay(300);
tone (so,1480,300); //fa#
  delay(300);
tone (so,1480,300); //fa#
  delay(300);
tone (so,1568,300); //sol
  delay(1500);
tone (so,1480,300); //fa#
  delay(300);
tone (so,1480,300); //fa#
  delay(300);
tone (so,1480,300); //fa#
  delay(300);
tone (so,1568,400); //sol
  delay(400);
```

```

tone (so,1864.7,500); //sib
  delay(500);
tone (so,2093,300); //do'
  delay(300);
tone (so,2093,300); //do'
  delay(300);
tone (so,2093,300); //do'
  delay(1500);
}
}
//md davant=180
//me davant=0
//md enrere=0
//me enrere=180

```

7.1.13 Programa final

```

#include <Servo.h>
#include <SoftwareSerial.h>

SoftwareSerial BT(0,1);
// creates a "virtual" serial port/UART
// connect BT module TX to D10
// connect BT module RX to D11
// connect BT Vcc to 5V, GND to GND

Servo md;
Servo me;
int led1=13; // int. dreta
int led2=12; //int. esquerre
int led3=11; //darrere
int led4=10; //lums
int so=2; //clàxon
long TP_init(int trigger_pin, int echo_pin);
long Distance(int trigger_pin, int echo_pin);
int distancia;

void setup()
{
  // set digital pin to control as an output
  BT.begin(9600);
  md.attach(5);
  me.attach(6);
  pinMode(led1, OUTPUT);
  pinMode(led2, OUTPUT);
  pinMode(led3, OUTPUT);
  pinMode(led4, OUTPUT);
  pinMode(8, INPUT);
  pinMode(9, OUTPUT);
  pinMode(so, OUTPUT);
}

```

```
}  
  
char a; // stores incoming character from other device  
void loop()  
{  
do  
{  
  distancia=Distance(9,8);  
  Serial.println(distancia);  
  delay(300);  
  if (BT.available())  
  {  
    a=(BT.read());  
    if (a=='w') //endavant  
    {  
      me.write(0);  
      md.write(180);  
      digitalWrite(led1,LOW);  
      digitalWrite(led2,LOW);  
      digitalWrite(led3,LOW);  
    }  
    if (a=='x') //parat  
    {  
      me.write(91);  
      md.write(91);  
      digitalWrite(led1,LOW);  
      digitalWrite(led2,LOW);  
      digitalWrite(led3,LOW);  
    }  
    if (a=='d') //gir a l'esquerra  
    {  
      digitalWrite(led1,HIGH);  
      digitalWrite(led2,LOW);  
      digitalWrite(led3,LOW);  
      delay(500);  
      digitalWrite(led1,LOW);  
      digitalWrite(led2,LOW);  
      digitalWrite(led3,LOW);  
      delay(500);  
      me.write(91);  
      md.write(180);  
    }  
    if (a=='a') //gir a la dreta  
    {  
      digitalWrite(led1,LOW);  
      digitalWrite(led2,HIGH);  
      digitalWrite(led3,LOW);  
      delay(500);  
      digitalWrite(led1,LOW);  
      digitalWrite(led2,LOW);  
      digitalWrite(led3,LOW);  
    }  
  }  
}
```

```
    delay(500);
    me.write(0);
    md.write(91);
  }
  if (a=='s') //enrere
  {
    me.write(180);
    md.write(0);
    digitalWrite(led1,LOW);
    digitalWrite(led2,LOW);
    digitalWrite(led3,HIGH);
  }
  if (a=='p') //encendre llums curtes
  {
    digitalWrite(led4,HIGH);
  }
  if (a=='l') //apagar llums curtes
  {
    digitalWrite(led4,LOW);
  }
  if (a=='u') //so
  {
    tone (so,2093,600); //do'
    delay(550);
    tone (so,2637,350); //mi'
    delay(400);
    tone (so,2960,300); //fa#
    delay(400);
    tone (so,3520,300); //la'
    delay(300);
    tone (so,3136,500); //sol'
    delay(600);
    tone (so,2637,400); //mi'
    delay(400);
    tone (so,2093,450); //do'
    delay(500);
    tone (so,1760,400); //la'
    delay(500);
    tone (so,1480,300); //fa#
    delay(300);
    tone (so,1480,300); //fa#
    delay(300);
    tone (so,1480,300); //fa#
    delay(300);
    tone (so,1568,300); //sol
    delay(1500);
    tone (so,1480,300); //fa#
    delay(300);
    tone (so,1480,300); //fa#
    delay(300);
    tone (so,1480,300); //fa#
```

```

    delay(300);
tone (so,1568,400); //sol
    delay(400);
tone (so,1864.7,500); //sib
    delay(500);
tone (so,2093,300); //do'
    delay(300);
tone (so,2093,300); //do'
    delay(300);
tone (so,2093,300); //do'
    delay(1500);
}
}
}
while (distancia>40);
    me.write(91); //parar motors
    md.write(91);
    digitalWrite(led1,LOW);
    digitalWrite(led2,LOW);
    digitalWrite(led3,LOW);
    me.write(180); //anar enrere
    md.write(0);
    delay(2000);
    me.write(91); //parar motors
    md.write(91);
}

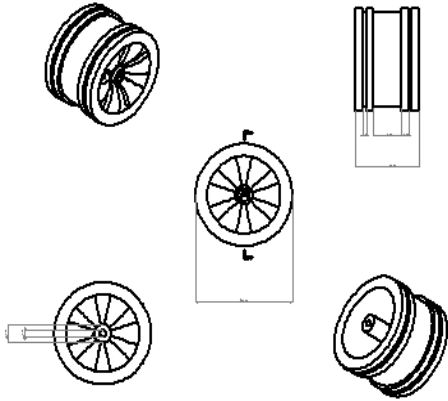
long TP_init(int trigger_pin, int echo_pin)
{
    digitalWrite(trigger_pin, LOW);
    delayMicroseconds(2);
    digitalWrite(trigger_pin, HIGH);
    delayMicroseconds(10);
    digitalWrite(trigger_pin, LOW);
    long microseconds = pulseIn(echo_pin ,HIGH);
    return microseconds;
}

long Distance(int trigger_pin, int echo_pin)
{
    long microseconds = TP_init(trigger_pin, echo_pin);
    long distance; distance = microseconds/29/2;
    if (distance == 0)
    {
        distance = 999;
    }
    return distance;
}

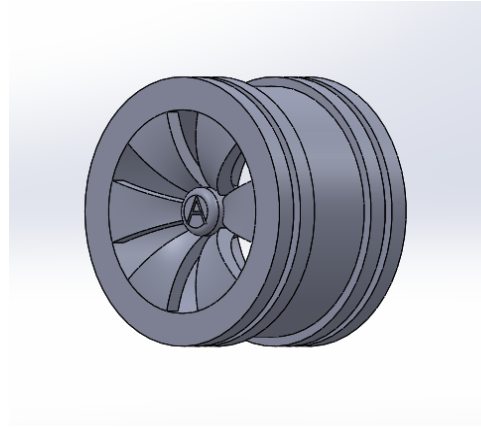
```


7.2 Disseny de peces amb SolidWorks

7.2.1 Disseny de les llantes



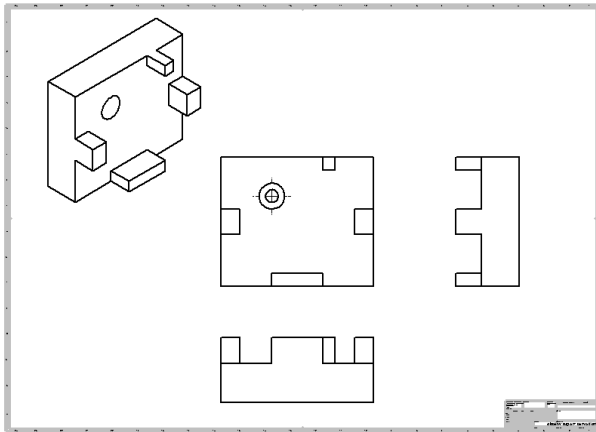
Il·lustració 45: Vistes de la llanta.



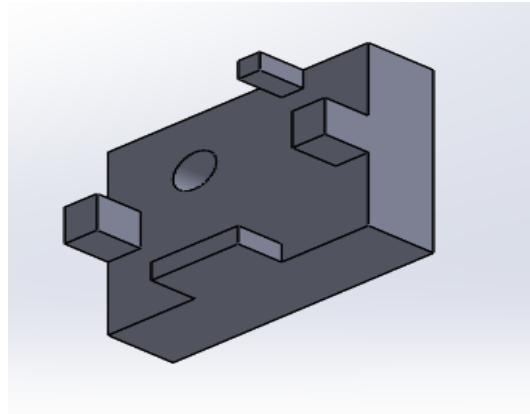
Il·lustració 46: Llanta amb 3D.

7.2.2 Disseny suports servomotors

7.2.2.1 Disseny suport servomotor dret

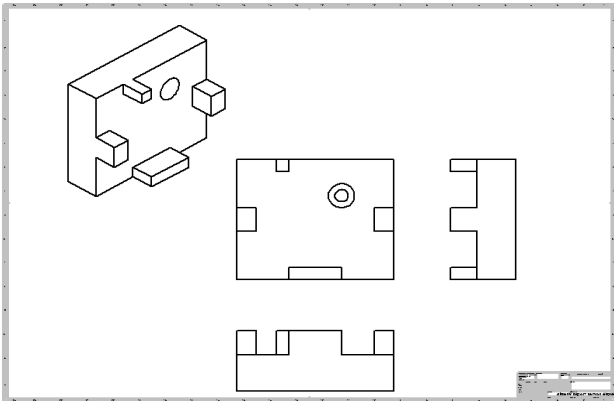


Il·lustració 47: Vistes del suport pel servomotor dret.

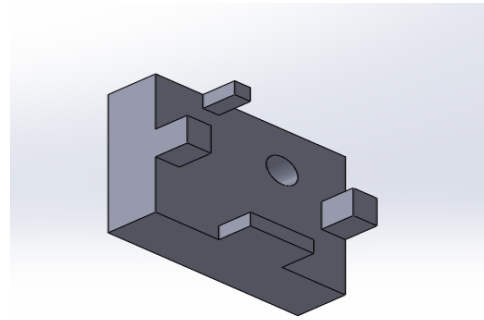


Il·lustració 48: Suport del servomotor dret amb 3D.

7.2.2.2 Disseny suport servomotor esquerre



Il·lustració 49: Vistes del suport pel servomotor esquerre.



Il·lustració 50: Suport del servomotor esquerre en 3D.