

Treball final de carrera.

***Sistemes de comunicació
per a Radio Comunicació .***

UOCaCAR

Consultor: Antoni Morell Pérez.

Alumne: Joan Roig Ferré

Semestre: Segon 2013-14.

Sistemes de comunicació per a Radio Control (RC) amb Arduino.**Índex.**

Índex.....	1
Introducció.	3
Objectius:.....	4
Objectius generals.....	4
Objectius específics.....	4
Enfoc del treball.....	5
El món del Radio Control.....	6
Antecedents històrics.....	6
Rang legal	7
Freqüències segons el medi.....	8
Terra.....	8
Aire	9
Mar.....	10
El món del Open HW.....	11
Introducció	11
Predecessors d'Arduino.....	11
Processing.....	11
Wiring.....	12
Arduino.....	13
Arduino UNO.....	15
Arduino Mega	15
Alternatives.....	16
Android.....	17
Estudi de RC amb Arduino.....	18
Disseny	18
Programació amb Arduino.....	18
Estructura del Codi.....	18
Errors típics.....	19
Part mecànica.....	20
Motors.....	20
Motors. Primer Prototip.....	21
Motors. Segon prototip.....	21
Motors. Proposta Final.....	22

Sistemes de comunicació per a Radio Control (RC) amb Arduino.

Alimentació.....	23
Suport Arduino.....	24
Control amb Arduino.	25
Moviments bàsics.	25
RC i Arduino.....	26
Arduino a 433MHz.	26
Prova de comunicació en 433MHz.....	26
Exemple de la matriu de polsadors.....	27
Matriu de Polsadors i 433MHz.....	28
Shield de Comandament i 433MHz.....	29
Arduino Bluetooth i Android	31
Control del Cotxe amb Arduino i Android.	32
Sensor d'humitat i temperatura.	32
Detector de monòxid de carboni.....	33
Possibles millores.....	34
Prototip Arduino – Android.....	35
Interfície Arduino.	35
Shield de control.	35
Codi Arduino.	36
Interfície en Android	37
Estructura Programa Android.....	37
SDK / Eclipse.	38
APP Inventor 2.	40
Pressupost.....	45
Possible comercialització.	46
Kits.....	46
Sensor mòbil.....	46
Consultoria.	46
Conclusions.....	47
Bibliografia.....	48

Introducció.

El món de la electrònica és relativament jove on cada generació té tendència a revolucionar tot lo prèviament existent canviant el món en el que vivim. La progressió tecnològica que han experimentat els seus components és espectacular: cada dia són més petits, simples, versàtils i barats. Cada cop l'electrònica es fa més fàcil d'interconnectar entre ella i a l'abast de tothom, aquest apropament al públic en general ha sorgit un gran moviment DIY (*Do It Yourself*) dintre d'aquest món. Aquesta tendència ha causat que en la xarxa es comparteixi cada cop més informació de com es poden fabricar des d'una impressora, connectar els objectes comuns de casa a Internet IOT (*Internet of Things*) o, com no, fabricar els teus propis vehicles a radio control. El món del Radio Control (RC) també ha experimentat tota aquesta evolució electrònica, reduint el preu dels seus components i augmentant-ne les prestacions. Passant a ser un món tancat dominat per grans companyies a quelcom obert i flexible a cada persona. En els sistemes professionals poder caracteritzar el vehicle i triar la freqüència és bastant comú; i en els sistemes DIY la llibertat és absoluta, cada dia són més comuns els manuals per fabricar-se el propi dispositiu, o robotitzar el mateix afegint-hi capacitat autònoma, capacitat per prendre decisions, ...

En aquest treball es demanava investigar el món del Radio control i fer una possible implementació amb Arduino o alguna altra plataforma Open Source. Arduino és una plataforma ideal per a la investigació i per fer prototips, cosa que permet la investigació sobre el terreny en casos pràctics. Es realitzarà un estudi genèric del món del RC, passant a un viatge d'aprenentatge i aplicació de la plataforma Open Source Arduino aplicat a diferents tecnologies al control de vehicles i la gestió de sensors. Com a part final es veurà com fer una aplicació amb Android, de forma ràpida, i la farem servir per una aplicació que combini bona part de les habilitats adquirides per obtenir un cotxe teledirigit que ens mostri telemetria.

Per tant tractarem una part teòrica per passar ràpidament a exemples pràctics que farem servir en un muntatge final. En aquest mateix apartat exposaré els objectius que s'esperen aconseguir, i quin serà el plantejament pràctic per aconseguir-ho.

Sistemes de comunicació per a Radio Control (RC) amb Arduino.

Objectius:

A continuació es detallen els objectius plantejats en aquest projecte, dividits entre els d'investigació o teòrics, i els més pràctics i concrets.

Objectius generals.

Hi ha una sèrie de coneixements teòrics que son recomanables per assumir una visió general del món de la Radiocomunicació i de l'Arduino. Es podria resumir aquest coneixements com:

- Estudi dels diferents sistemes de radio control.
- Estudi dels antecedents històrics del món Arduino:
 - Antecedents històrics.
 - Alternatives.
- Estudi tècnic dels diferents medis de RC:
 - Terra.
 - Aire.
 - Mar.
- Estudi pràctic de les solucions de comunicació radio per Arduino.
- Programació amb Arduino.
- Programació amb Android.
- Comunicació Arduino – Arduino.
- Comunicació Arduino – Android.

Objectius específics.

De forma més específica, hi ha una sèrie de maquinari i programari que hem de fer ús. Concretament el següent:

- Control de motors amb Arduino mitjançant transmissors.
- Programació de placa de drivers de motors per Arduino.
- Ús de la parella emissor / receptor sèrie a 433MHz.
- Ús d'un emissor / receptor de Bluetooth amb Arduino.
- Sensors amb Arduino:
 - Sensor de temperatura.
 - Sensor d'humitat.
 - Sensor de CO (escalable a tota la sèrie de sensors MQ).
- Disseny i ús de Shields en Arduino:
 - Shield "comandament a distància": amb polsadors i interfície radio.
 - Shield "sensors i Bluetooth": amb sensor variis i comunicació Bluetooth.
- Adquirir experiència en l'entorn de programació Arduino.
- Conèixer preliminarment els entorns de programació d'Android:
 - SDK / Eclipse.
 - MIT App Inventor 2.

Sistemes de comunicació per a Radio Control (RC) amb Arduino.

Enfoc del treball.

El treball constarà d'una part introductòria de teoria del món del RC, l'Open Hardware i un breu resum de la història del Arduino. En la part RC es farà un estudi tècnic diferenciant els diferents medis (terra, mar i aire), les principals característiques i diferències. S'hi inclourà també un petit resum històric i de la seva evolució.

Seguidament la part principal del projecte consistirà en un estudi de varies tecnologies aplicades al Arduino . Es començarà per fer un estudi dels motors elèctrics, les diferents possibilitats que tenim per controlar motors senzills DC, i com poden ser controlats per un Arduino. Després es faran proves d'intercomunicació entre dos dispositius Arduino, primer amb un mòdul de 433MHz fent-ne una aplicació pràctica senzilla, i seguidament amb Bluetooth, una vegada provat el Bluetooth es provarà amb un Android. Per acabar la part més de prova es connectaran una sèrie de sensors a un Arduino per obtenir telemetria.

Amb aquests coneixements es farà una part pràctica final dissenyant una Shield d'Arduino que ens permeti telemetria, control del cotxe i comunicació amb Bluetooth, per altre banda es programarà una aplicació amb Android per rebre la telemetria i fer de comandament remot.

Sistemes de comunicació per a Radio Control (RC) amb Arduino.**El món del Radio Control.**

El món del RC va sorgir com una curiositat per a nens però actualment està adquirint matisos molt més professionals, des de robots autònoms a cotxes amb IA. Per tant ha passat de ser una curiositat paral·lela al desenvolupament tecnològic a un camp d'interès per si mateix.

**Antecedents històrics.**

Fins pràcticament fa pocs anys els vehicles a radio control tan sols consistien en un element actiu, o comandament, que controlava el vehicle. Per tant la potència i característiques del remot es podien basar en poc més a que a controlar els motors.

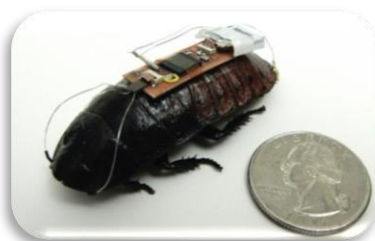
La primera generació de RC consistia en una etapa de radio AM connectada directament als motors de forma molt directe, molts d'ells amb relés. Com que el sector es va popularitzar va néixer la necessitat de tindre gran varietat de freqüències i es van començar a utilitzar filtres en la part radio.

Poc a poc tota la tecnologia va començar a funcionar a través d'ones o impulsos. Amb l'entrada en joc del FM aquestes ones varen començar a ser cada cop més "quadrades" o digitals. Empreses comercials com: Futaba, JR Multiples o Graupner; varen sorgir en aquest àmbit. Sempre s'han fet servir unes certes freqüències les més populars: 27Mhz, 35Mhz, 54Mhz i 72Mhz, aquesta última sobretot als Estats Units; on a demés teníem varis canals per tal d'evitar les interferències.

Hitec va ser un dels primers a fer el canvi als 2.4GHz, primer per tema de patents, ja que empreses com Futaba feia anys que operava en l'anterior marge freqüencial amb productes molt bons, flexibles i adaptables en freqüència i canal. Un dels altres motius era per donar accés WiFi amb el seu "Aurora 9", que es considera no tan sols un Emissor/Receptor de RC sinó un mòdem.

El fet de fer servir la banda lliure dels 2.4GHz va suposar un canvi d'enfocament bastant important, i la gent va començar a fer servir aplicacions en Bluetooth, wireless, ... fent que les opcions augmentessin alhora que en baixés el preu. Una de les característiques que en va sorgir és que la bidireccionalitat en les comunicacions ara era molt més senzilla, i per tant es podia enviar telemetria, incorporar sensors, ... cosa molt útil per controlar un vehicle aeri per exemple.

L'últim gran canvi ha estat la baixada del preu dels controladors, i el sorgiment d'alternatives Open Source, que permeten dotar de molta intel·ligència al vehicle, fent que prengui petites decisions, com l'estabilitat en aeronaus o evitar obstacles en cotxes. També ha sorgit tot un camp paral·lel i no pròpiament de Radio Control: els UAV (Unmanned Aerial Vehicle) o sistemes autònoms aeris, cotxes/robots que poden realitzar petites funcions sols (seguir una línia ...) o molts d'altres.



El futur és molt difícil de preveure, però amb la ràpida progressió i l'abaratiment de la tecnologia ningú es pot imaginar que es pot aconseguir en un futur no molt llunyà.

Sistemes de comunicació per a Radio Control (RC) amb Arduino.**Rang legal**

Legalment es permet fer us de les bandes lliures de freqüències recomanat les de : 433Mhz, 2'4Ghz i 5'8GHz ; no obstant i que les tradicionals son: 27Mhz, 35Mhz, 54Mhz i 72Mhz; El govern el que sí controla molt és el tema de les potències d'emissió, ja que obliga a treure's el permís de Radioaficionat si es vol excedir aquesta potència.

Per exemple el 15 de setembre del 2005 la "Secretaría de Estado de Telecomunicaciones y para la Sociedad de la Información" va publicar en el BOE la limitació per a 35MHz. Consistent en 18 canals de 10KHz, a una potència màxima de 500mW p.r.a i una potència de màxima de 100mW. També limita el seu ús a curt abast i a "aficionat".

**ANEXO**

Interfaz reglamentada en España

Descripción: Equipos de corto alcance (SRD) para control de aeromodelismo.

Control por radio en aplicaciones de Aeromodelismo.

Numero: 2.

Fecha: 14 de junio de 2004.

Parámetro Datos técnicos

1 Frecuencia/Banda de frecuencias.

35,025-35,205 MHz.

Canal 1: 35,030 MHz. Canal 10: 35,120 MHz.

Canal 2: 35,040 MHz. Canal 11: 35,130 MHz.

Canal 3: 35,050 MHz. Canal 12: 35,140 MHz.

Canal 4: 35,060 MHz. Canal 13: 35,150 MHz.

Canal 5: 35,070 MHz. Canal 14: 35,160 MHz.

Canal 6: 35,080 MHz. Canal 15: 35,170 MHz.

Canal 7: 35,090 MHz. Canal 16: 35,180 MHz.

Canal 8: 35,100 MHz. Canal 17: 35,190 MHz.

Canal 9: 35,110 MHz. Canal 18: 35,200 MHz.

2 Canalización/Anchura de banda. 10 kHz (18 canales).

3 Modulación. No especificada.

4 Separación dúplex. No hay.

5 Nivel de potencia. Potencia máxima de salida de equipo 500 mW p.r.a. maxima 100 mW.

6 Servicio radioeléctrico/tipo de dispositivo. Dispositivo de corto alcance.

Parámetros de información opcional

7 Licencia/uso. No se precisa licencia. Uso común.

8 Evaluación/notificación. Clase II.

9 Norma técnica de referencia. ETSI EN 300 220-3.

10 Otras observaciones. CNAF UN-10 frecuencias exclusivas para esta aplicación.

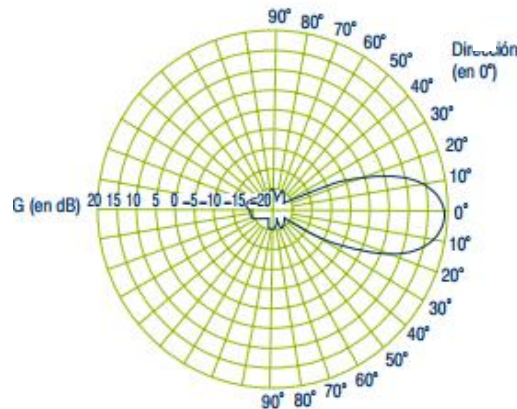
Parámetro Datos técnicos...

Sistemes de comunicació per a Radio Control (RC) amb Arduino.**Freqüències segons el medi.**

Cada medi té les seves pròpies característiques mesurades amb termes com: propagació, absorció, reflexió i penetració. Tots els medis tenen quelcom en comú com el tractament de les antenes habitualment s'aproxima a la longitud d'ona: ($\sim 300.000.000\text{m/seg}$) / freqüència.

$$\text{Per } 35\text{MHz} : 300.000.000 / 35.000.000 = 300 / 35 = 8,57\text{m.}$$

Si fem servir una freqüència 10 vegades més petita tindrem una antena 10 vegades més gran, per lo que les freqüències baixes demanen antenes enormes, també podem minimitzar aquesta longitud fent servir mitges ones o quarts. Aquesta limitació de potència limita molt la cobertura, de fet és el seu objectiu, per això en RC sempre s'ha fet servir les freqüències més baixes possibles; ja que les freqüències a MHz tenen més cobertura i menys atenuació.

**Terra.**

Es pot considerar que és el medi per defecte, encara a dia d'avui és en el que hi ha més equips i aficionats encara que amb l'abaratiment dels equips medis com l'aeri estan revolucionant el món del RC. El sistema terrestre té un gran avantatge respecte dels altres medis, si es perd la senyal de control.. no passa res, el vehicle simplement es deixa de moure, per lo que el fa un medi ideal per aquest projecte, sobretot en la part d'investigació.

Al ser un medi poc hostil ens permet simplificar al mínim la part de RC, ja que aquesta passa a ser menys crítica, i de fet bona part dels vehicles terrestres es caracteritzen més per la autonomia o potència dels seus motors. En aquest projecte ens permet no preocupar-nos del pes del dispositiu, la pèrdua de senyal o altres problemes més propis del control de robots propis d'altres disciplines de la enginyeria.

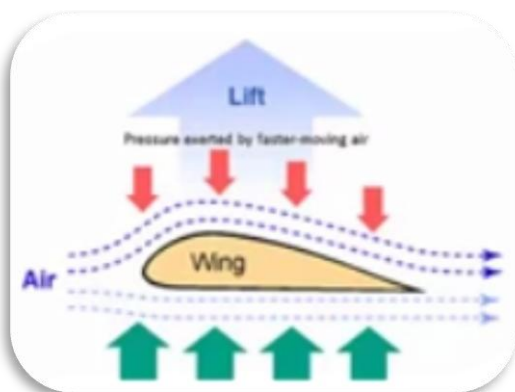


Sistemes de comunicació per a Radio Control (RC) amb Arduino.Aire

El medi aeri és molt semblant al terrestre amb algunes peculiaritats com el fet de disposar, habitualment, de visió directe, que les distàncies acostumin a ser més grans i que s'acostumi a no tenir obstacles en la visió. Alguns equips senzills de baix cost d'Aeromodelisme, de fet, fan servir el fet de que la visió directe per fer servir infrarojos, encara que aquests equips la pèrdua de control del equip és molt freqüent.

Actualment s'està començar a utilitzar molt els 2'4GHz ja que aquestes tenen molta més capacitat i està molt preparades per tal fer servir codificació digital i així evitar la interferència d'altres equips de la mateixa freqüència, degut justament a aquest perill d'interferència s'estan començant a fer servir molt en Aeromodelisme on perdre la senyal pot ser crític, tot i que la cobertura és teòricament inferior.

Els equips clàssics eren maquetes d'avions i funcionaven amb els mateixos principis dels avions clàssics utilitzats ja al 1903 pels germans Wright. Amb un motor o dos, gràcies a la configuració de les seves ales s'obté un vol més o menys estable i controlat per unes solapes en les ales que en permeten el control d'altura. El principal problema és que tan sols s'aconsegueix certa estabilitat si es troba en moviment.



Per solucionar aquest problema al 1924 es va intentar dissenyar un "quadcòpter" encara que sense els sistemes electrònics de control actual no es va poder fer funcionar. No va ser fins una dècada més tard, 1936, amb la invenció del helicòpter que es va aconseguir una solució eficient. No va ser fins als voltants de l'any 2000 que van començar a fer viables els quadcòpters gràcies als estabilitzadors electrònics d'altitud, aquests tenen quatre motors els quals 2 giren en un sentit i els altres dos en sentit contrari per evitar que es fiqui a donar voltes, i controlant la velocitat dels mateixos aquest canvien d'altitud, giren, ...

Quadrotor: Flying Principle TUM

Keep position:

- Thrust compensates for earth gravity
- Torques of all four rotors sum to zero

Jürgen Sturm Autonomous Navigation for Flying Robots 4

1:02 / 10:27 SPEED 1.0x



Sistemes de comunicació per a Radio Control (RC) amb Arduino.**Mar.**

L'aigua es un medi hostil per les ones de radio, últimament s'hi està investigant molt tan per fer funcionar els mòbils com per fer comunicacions subaquàtiques a llarga distància. Actualment, no obstant, es fa servir ultrasons o senyals molt baixa freqüència (VLF 3-30MHz) per les comunicacions sota del aigua, en freqüències superiors no es pot penetrar més que uns pocs metres; i s'acostumen a fer servir balises per fer la comunicació per aire i comunicar per cable amb el vehicle, de pas aconseguim un sistema pràctic per recuperar-l'ho en cas de pèrdua de les comunicacions.



Al ser un medi tant peculiar val la pena fer un petit incís tècnic sobre algunes característiques específiques del medi.

Conductivitat de l'aigua.

L'aigua destil·lada és un bon aïllant amb una atenuació molt alta, però en estat natural conté sals i altres materials, fent-la més conductora fet que fa que atenuï menys les senyals que passen a través d'ella, com més concentració més es dona aquest efecte, la temperatura també és un factor a tindre en compte. La conductivitat en 1m d'aigua de mar es considera de $0'25\Omega$, encara que es dobla en les regions fredes del àrtic i és la meitat en regions salines com el mar Roig. Una de les mostres d'aigua que es prenen de referència en aigua dolça és l'aigua d'Adelaida presa el 1983, amb una concentració de 300mg/litre obtenint una mostra de $0'0546\text{mhos per metre}$ (uns $18'31\Omega$).

Refracció de l'aigua.

Quant una ona electromagnètica canvia de medi existeix una pèrdua per refracció, en l'aigua de mar pot arribar a ser de 60dB per baixes freqüències. És molt difícil de combatre per lo que la solució més fàcil és la de simplement evitar-la ficant el transmissor sota l'aigua, o un repetidor que estigui en contacte amb l'aigua i l'aire a la vegada.

Sistemes de comunicació per a Radio Control (RC) amb Arduino.**El món del Open HW.**

Actualment estem vivint una petita revolució en el món del Hardware (HW), tal com ja havia passat amb el software cada cop tenim més alternatives lliures de tota mena de maquinari, que passen des de petites plaques de control a projectes tan engrescadors com Arduino o Raspberry Pi.

Introducció

L'origen de bona part de les tecnologies actuals s'ha creat o popularitzat en entorns de creació lliure ja des dels seus principis en universitats. Aquest entorn de cultiu ha creat una font de negoci tan gran que ha creat una divisió entre els que pensen que el coneixement ha de ser lliure per a tothom i els que defensen la propietat intel·lectual.



Un de les persones més radicals, a la vegada que més icòniques i productives del sector, és Richard Stallman. L'any 83, després de batallar a favor dels seus ideals de compartir codi de forma lliure va crear el projecte GNU per crear un Sistema Operatiu completament lliure, i l'any 85 va crear la fundació FreeSoftwareFoundation que promovia la llibertat absoluta en el camp del programari.

Com era d'esperar aquesta radicalitat, i el fet que la paraula Free en anglès vol dir llibertat i gratuït, va fer que alguna gent del sector, la majoria, incitats per grups empresarials, establissin unes noves normes del joc, a la vegada que l'hi canviaven el nom. Així va ser com en el 1998 es crees el moviment Open Source, liderat per Tim O'Reilly, en català s'acostuma a traduir com a Codi Obert.

Tot el moviment estava bastant basat i pensat en el Software, no obstant l'abaratiment de la electrònica i la popularització d'Internet va fer aplicar els mateixos principis de compartir coneixements amb el naixement del Open Hardware.

Predecessors d'Arduino.

Dintre del món del disseny va sortir una necessitat d'incorporar la tecnologia al disseny i a l'art; però per fer-ho s'ha de disposar d'una plataforma suficientment intuïtiva i fàcil de tractar per gent que no tenen un perfil tant tècnic. Dos dels màxims exponents d'aquesta tendència a apropar el món del HW i de la programació a tot el públic són els projectes Processing i Wiring.

Processing.

El Processing es un llenguatge de programació Open Source basat amb Java però amb la sintaxis molt simplificada i fent ús de models gràfics. Processing també acostuma a designar el entorn integrat de desenvolupament (IDE en anglès) orientat a crear obres artístiques i a el disseny. Neix amb la intenció d'ensenyar els fonaments de la programació en un entorn més visual, intuïtiu i immediat.

Sistemes de comunicació per a Radio Control (RC) amb Arduino.

En el 2001 el projecte va ser començat originalment per Casey Reas i per Benjamin Fry del MIT Media Lab, amb la clara intensió d'introduir en la programació a la gent aliena a ella, fent que de forma casi immediata podem tenir un resultat visualment satisfactori.

A part de ser un llenguatge molt famós per artistes gràfics per desenvolupar composicions orgàniques i com a base d'alguns videojocs, en aquesta introducció ens interessa perquè engendrà els projectes germans Wiring i Arduino.

Wiring.

Wiring s'acostuma a definir com una plataforma de codi obert per a fer prototips, integrant de forma intuïtiva un llenguatge de programació, un microcontrolador i una placa amb el connexionat (idealment integrada amb el microcontrolador). La idea principal és semblant a la filosofia del Processing però més orientada al món físic, permetent a qualsevol ser capaç d'encendre un led (l'equivalent al "Hello World") i anar fent evolucionar el codi; tot fent esbossos (Sketching) del part de la electrònica de projectes aparentment complexos per tal d'investigar, aïllar errors,...

El projecte va néixer de mans de Hernando Barragán el 2003 en l'Institut Ivrea de Interacció i Disseny (el mateix d'on sortirà Arduino com a evolució del Wiring), encara que actualment s'ha traslladat a l'Escola d'Arquitectura i Disseny de la Universitat de Los Andes en Bogotá. El disseny HW de Wiring es publica sota Creative Commons (Attribution Share-Alike 2.5) i el codi font de la interfície de treball i les llibreries HW sota GPLv2.



Wiring està basat en el Processing però incorpora algunes novetats a la seva interfície, la Wiring IDE, tot i que està escrita en Java i té una estructura semblant al Processing, ve per defecte amb una llibreria C/C++ anomenada "Wiring" que permet tractar les entrades i sortides forma molt senzilla i semblant a com es faria en aquests llenguatges. Es pot escriure els codis en C/C++, amb la diferència que tan sols es necessiten definir 2 funcions per fer funcionar el codi:

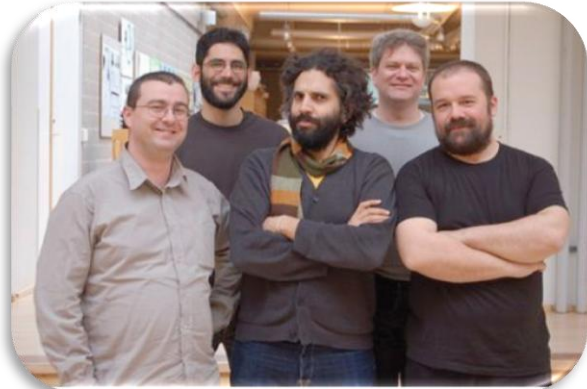
- Setup(): funció que s'executa al inici o es defineix l'entorn de treball.
- Loop(): bucle de codi que s'executa després del Setup() i mentre hi hagi alimentació.

Més tard el compilador s'encarrega d'adaptar el codi per ser carregat al processador i fins i tot ens ofereix la possibilitat d'afegir una funció main() per fer-l'ho un codi C++ vàlid.

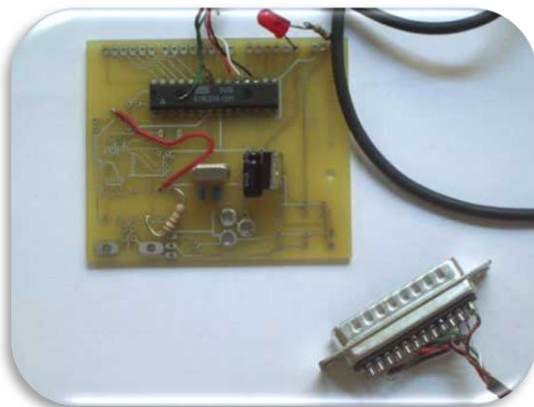
Sistemes de comunicació per a Radio Control (RC) amb Arduino.**Arduino**

Arduino I va néixer el 1995 a Pombia, fill de Dadone, era Marquès de Ivrea i es va autoproclamar rei d'Itàlia el 1002. Uns anys més tard l'hi varen dedicar un bar, el "Bar di Re Arduino" que freqüentava Massimo Banzi (en la foto ell i el seu equip: Massimo Banzi, David Cuartielles, Tom Igoe, Gianluca Martino, i David Mellis). Aquest professor de l' Institut d' Ivrea cansat del alt preu dels controladors que es feien servir va crear i batejar lo que actualment coneixem com Arduino.

Es tracta d'una placa de HW lliure que consisteix amb un microcontrolador i un entorn de desenvolupament amigable, pesada per facilitar projectes procedents de diferents disciplines.

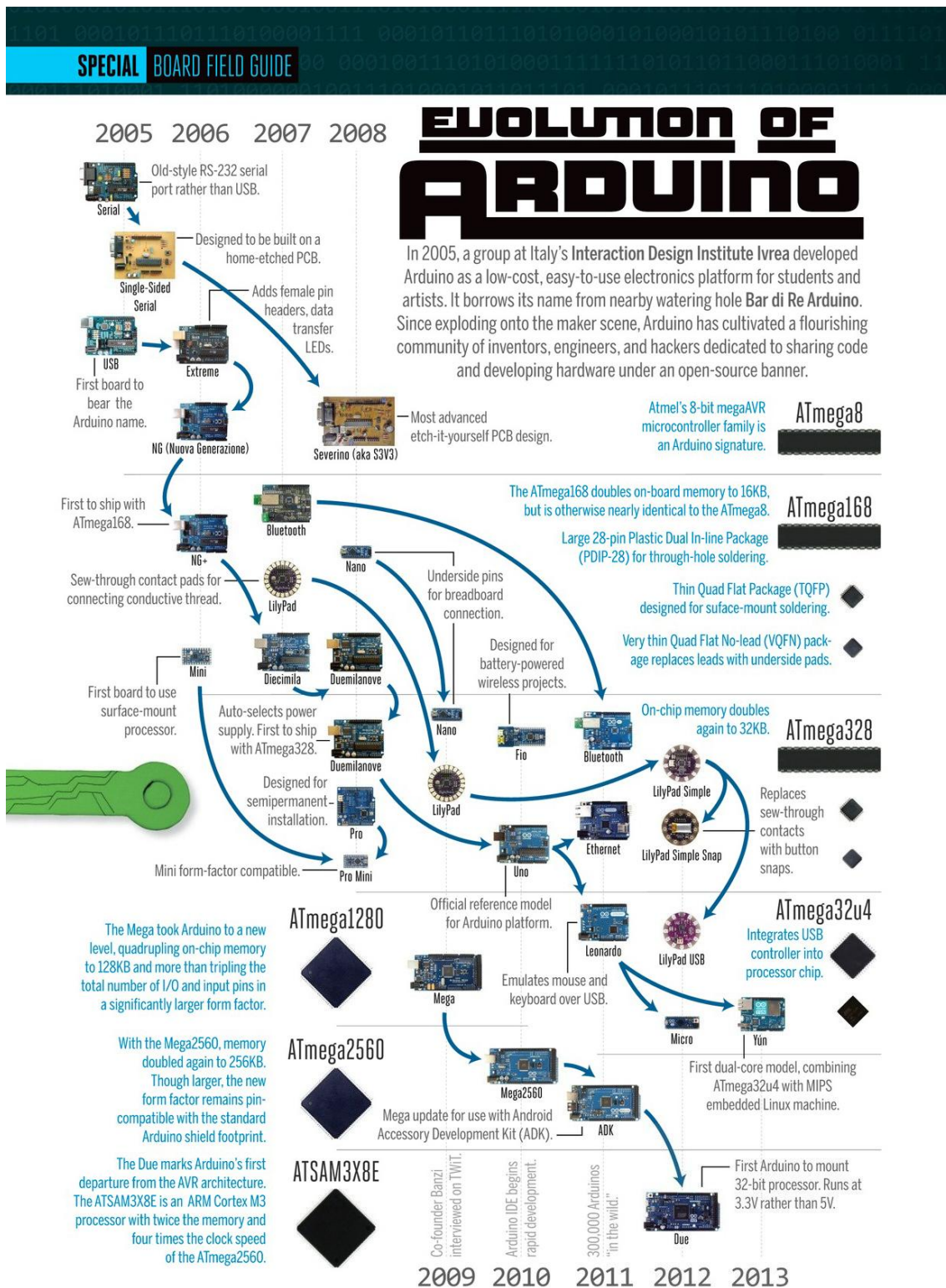


El projecte Arduino neix el 2005 però no neix de forma esporàdica, ja existien projectes semblants, de fet podríem dir que es germà de Wiring i fill del Processing. El seu baix cost, la flexibilitat de l'entorn de treball i el fet de ser Open Source va fer que s'hagi popularitzat moltíssim.



Actualment existeixen gran varietat de plaques Arduino, ja no tan sols les oficials sinó una infinitat de adaptacions particulars, cosa normal considerant que els seus dissenys són públics. Els primers models, i el més que popular Arduino Uno, estan basats en la família de xips ATmega. S'ha evolucionat molt des del seu primer disseny que venia amb un manual perquè el soldés el mateix usuari. I ara existeixen gran varietats de models, on no solament tenen una més que eficient protecció i gran varietat de ports, sinó solucions concretes a necessitats concretes.





Sistemes de comunicació per a Radio Control (RC) amb Arduino.**Arduino UNO.**

La placa que es farà servir com a controlador és la versàtil Arduino UNO. Hi ha moltes versions de la mateixa i de fet no és res més que la evolució de la Duemilenove. Té el xip en format clàssic enlloc del SMD, per lo que el podem substituir o el podem treure i fer servir un cop programat sense tota la circuiteria de control ni els ports extres.

Una de les característiques que el fa ideal per la investigació i el prototipatge són els sistemes de control, com per exemple la protecció dels ports d'entrada-sortida o el de l'alimentació, si s'aplica més de 500mA al USB el sistema simplement "talla" l'alimentació.

L'Arduino UNO R3 és una placa de microcontrolador amb el xip ATmega328. Té 14 entrades/sortides digitals (de les quals 6 es poden fer servir com sortides PWM), 6 sortides

analogies, un oscil·lador de cristall a 16MHz, una connexió USB, un botó de reinici i una capçalera ICSP.

Treballa per defecte a 5V, es recomana una font d'alimentació entre 7 i 12V (però suporta entre 6 i 20V). Pot subministrar 40mA per pin I/O, i 50mA per les sortides a 3'3V. És important perquè ens obliga a fer servir etapes de potència en la major part dels cassos. Té una memòria flash de 32KB, un sector d'inici de 0'5KB, una SRAM de 2KB i una EEPROM de 1KB.

El seu nom se l'hi va donar perquè és el primer model de l'actual generació de plaques, basades amb una entrada USB, i la majoria deriven o evolucionen d'aquesta. Tot i la diferència amb els models Micro i els actuals amb SO com la Galileo o la Tre.

Arduino Mega

En algunes fases del projecte es farà servir un segon controlador Arduino per tal de controlar el receptor, la placa que s'ha triat per aquest fi ha estat l'Arduino Mega 2560 aquest amb el xip en SMD. L'Arduino MEGA 2560 és una placa de microcontrolador que deu el seu nom al seu processador: el xip ATmega2560.

Té 54 entrades/sortides digitals (de les quals 15 es poden fer servir com sortides PWM), 16 sortides analogies, un oscil·lador de cristall a 16MHz, una connexió USB, un botó de reinici i una capçalera ICSP.

Treballa per defecte a 5V, es recomana una font d'alimentació entre 7 i 12V (però suporta entre 6 i 20V). Pot subministrar 40mA per pin I/O, i 50mA per les sortides a 3'3V. Té una memòria flash de 256KB, un sector d'inici de 8KB, una SRAM de 8KB i una EEPROM de 4KB.

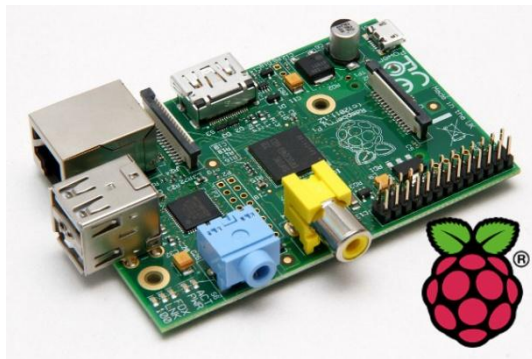
Sistemes de comunicació per a Radio Control (RC) amb Arduino.**Alternatives**

Hi ha gran varietat de plaques que tenen sortides analògiques i digitals que poden controlar des de LEDs a motors. Originalment es podien distingir dos alternatives clares les plaques que estaven molt orientades a la circuiteria i poc a la interfície amb l'usuari com poden ser l'Arduino, Nanode o l'espanyola, i molt orientada a sensors, Libelium Wasp mote.



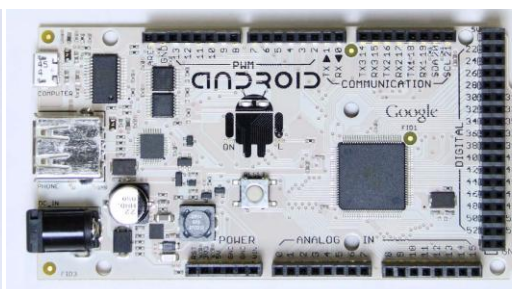
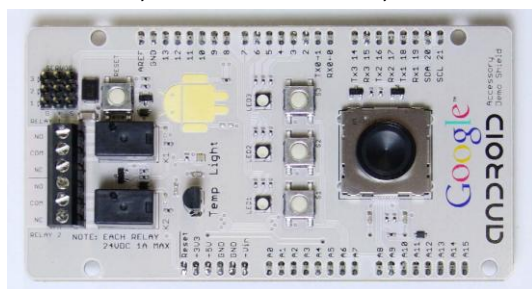
Ja s'ha fet referència al projecte predecessor a Arduino a l'Institut Ivrea: el Wiring. Presenta característiques, i fins i tot preus, molt semblants al Arduino, i és per tant una clar substitut en gairebé tots els sentits a aquest, encara que menys popular.

L'altre alternativa tecnològica són els ordenadors Micro basats en Linux com poden ser la Raspberry Pi, la BeagleBone o fins i tot els últims Arduino (Tre o Galileo), que no son més que petits ordenadors que tenen certa facilitat en controlar directament maquinari.



En l'any 2006 a Cambridge Eben Upton, Rob Mullins, Jack Lang i Alan Mycroft; varen dissenyar un petit equip, la RaspBerryPi, com a catalitzador per a noves generacions molt allunyades del món HW i tan sols sabien fer servir eines ofimàtiques o HTML com a molt. No obstant això la seva gran flexibilitat i la facilitat de fer sortides digitals ha fet que la seva popularitat entre els informàtics hagi estat

sense precedents, fent-se servir en tota mena de projectes. Algunes de les seves característiques principals dels models actuals son les sortides a 3'3V, tindre un port ethernet, sortida VGA i fer servir una targeta SD com a memòria. Una de les seves curiositats és que el seu processador principal és un processador pensat per el tractament d'imatge, per lo que el fa ideal en aquest entorn i hi ha multitud d'aplicacions fetes amb ella; com estacions multimèdia, control de càmeres, ...

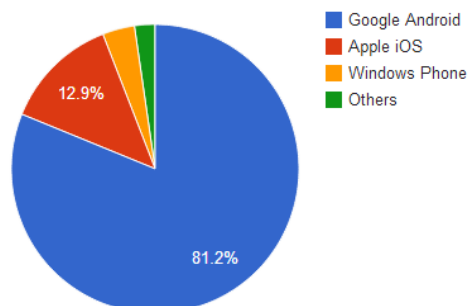


En el 2011 Google va llençar en la seva conferència anual de Maig un kid de desenvolupament anomenat Google ADK consistent amb una placa Arduino MEGA (encara que amb el seu logo) i una Shield que tenia els sistema operatiu Android OS i una sèrie de perifèrics per interactuar amb el món real.

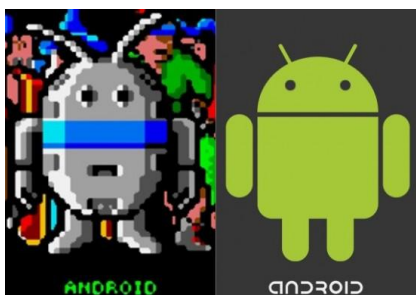
Sistemes de comunicació per a Radio Control (RC) amb Arduino.**Android.**

L'any 2005 el ja poderós Google va comprar una petita companyia Android.INC, tot feia pensar en l'aparició d'un telèfon de Google un "gPhone" o quelcom semblant però en el 2007 Google va sorprendre a tothom anunciant la creació de l'"Open Handset Alliance" i Android va passar a ser Open Source. El 2008 l'Android SDK 1.0 aparegué, com a plataforma de desenvolupament basada en Java, i HTC va llençar el seu primer mòbil Android amb T-Mobile USA. En 2009 va començar a estendre's i en 2010 tan sols era superat per Black Berry, que, curiosament, a finals d'aquest any ha anunciat que deixarà de fabricar terminals. I de llavors amb ara doncs, tal i com es pot veure al gràfic, la majoria de terminals nous han estat Android en 2013.

Mobile OS Market Share, Q3 2013



Tècnicament Android es basa en una versió modificada del nucli Linux, tal com la seva plataforma de desenvolupament SDK esta basada en Java. Tant la Open Handset Alliance com la Android Open Source Project (AOSP), tenen com objectiu mantenir i continuar desenvolupant l'Android per esdevindre la plataforma de desenvolupament de referència dels telèfons intel·ligents.



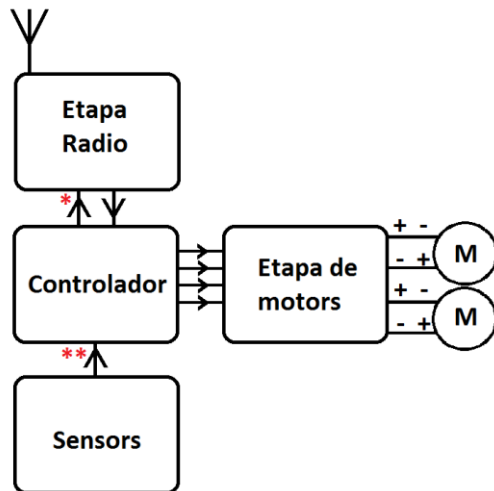
Com bona part del món Open Source els orígens de tots els seus detalls no sempre son clars, per exemple el seu logo, l'Andy, es basa en esbós fet per Irina Block en 5min i que tothom va rebutjar els intents posteriors per fer-l'ho més atractiu visualment, ja que la seva simplicitat el feia una icona internacional, evitant que conceptes culturals o religiosos s'interposessin enmig. No obstant i que s'ha la seva semblança amb el robot de "Guantleet: The Third Encounter" dels anys 90 és més que evident.

Altres font apunten al Best Seller : "Do Androids Dream of Electric Sheep.", de'n Philip K. Dick dels 1968 i que va esdevindre més tard la base per la pel·lícula Blade Runner. També cal destacar la obsessió pel menjar que demostren el nom de les seves versions:



Sistemes de comunicació per a Radio Control (RC) amb Arduino.**Estudi de RC amb Arduino.**

Aquesta secció del treball consistirà en una sèrie de proves amb gran multitud de tecnologies provant diferents solucions pels motors, sistemes radio, plataformes, ... tot buscant solucions diferents pel mateix fi. Tal com es demanava el controlador central serà el mateix, un Arduino, en aquest cas un Arduino Uno.

Disseny

S'estructurarà del disseny central del TFC de tal forma que consisteixi en una sèrie de mòduls intercanviables entre si fent que en qualsevol moment es pugui canviar un part sencera a una altre tecnologia sense que això afecti a la resta del projecte. Fen-t'ho d'aquesta manera ens podem permetre luxes com canviar el vehicle, la placa de control, el programa, ...

La primera divisió és la més obvia:

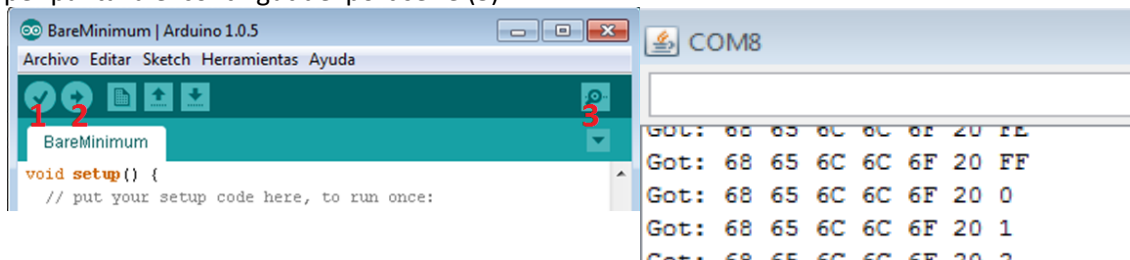
- Part mecànica: control físic del cotxe
- Recepció radio: part radio del cotxe, HW
- Emissió radio: des d'on controlem el cotxe, en principi un Android, però per fer proves pot anar bé fer-ho des d'un altre Arduino o des del PC.

(*) S'anirà fent evolucionar poc a poc el projecte amb complexitat, per exemple en 433MHz el sistema serà completament unidireccional mentre que en Bluetooth pot ser bidireccional.

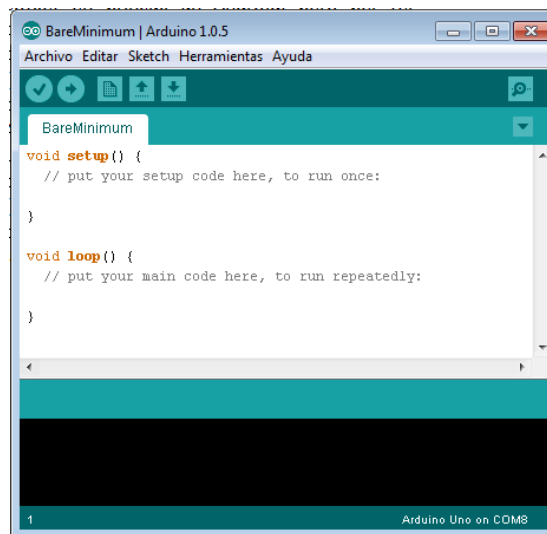
(**)Igualment la part de telemetria i sensors no te sentit si la unitat de control no rep dades de la unitat remota.

Programació amb Arduino.

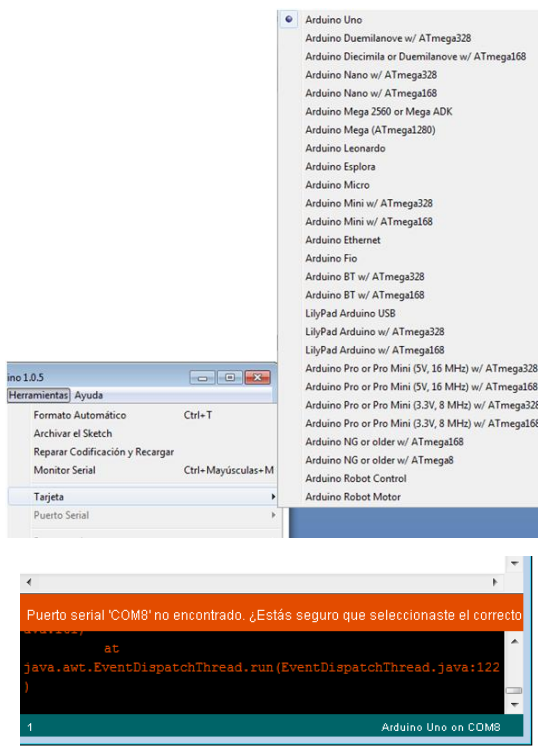
Tal com ja s'ha comentat l'Arduino s'alimenta dels projectes Wiring i Processing, molt propers per herència al codi C i amb una interfície de desenvolupament basat en Java. L'entorn de treball que ens ofereix el propi projecte Arduino ens deixa entre, d'altres opcions molt útils, a la vegada que fàcils de fer servir, comprovar el codi (1), carregar el codi a la placa (2), i veure per pantalla el contingut del port sèrie (3).

**Estructura del Codi.**

El codi d'Arduino es caracteritza per seguir un ordre d'execució concret, primerament s'executarà el codi del "void setup()", en ell ficarem tot el codi de configuració i inicialització de variables, definició de ports, ...; tot seguit s'executarà el bucle "loop()" en ell ficarem tot el codi que s'executarà de forma recurrent, o sigui que quan acabem d'executar totes les ordres de dintre del loop, tornarem a començar, el procés es repetirà mentre hi hagi alimentació. De fet el símbol d'Arduino reflecteix aquest procés un "+ -" que simbolitza la part electrònica dintre d'un símbol d'infinet indicant el bucle "∞".

Sistemes de comunicació per a Radio Control (RC) amb Arduino.

Evidentment per tal de facilitar la programació es poden definir constants, variable globals i funcions. Les constants i variables globals, s'acostumen a introduir abans del setup() i les funcions entre el setup() i el loop(); encara que al tractar-se d'un projecte pensat per a no programadors la veritat és que és molt poc sensible a estructures, pel que podem definir variables en quasi qualsevol punt del codi.

Erroros típicos.

La popularitat del Arduino també té els seus desavantatges a l'hora de carregar un codi nou a la placa ens hem d'assegurar de que hem seleccionat la placa amb la que estem treballant. Acostuma a estar seleccionada la placa del Arduino Uno per defecte, encara que si fem ús d'altres plaques, com nosaltres amb la MEGA, s'ha de tenir en compte.

La interfície de programació fa ús dels ports "COM" per a comunicar amb la placa, però, sobretot en Windows, la pèrdua de connectivitat és molt comú, sobretot si hi ha sobrealimentacions, connexions al port sèrie (que és el mateix que utilitza el USB, ...). De fet el port sèrie del propi ordenador es desactiva per seguretat si detecta un problema elèctric, cosa que ens obliga a reinicialitzar el port.

Part mecànica.

Per simplificar i abaratir la part mecànica es farà servir com a base un cotxe teledirigit comercial que s'adaptarà al projecte. He comprat un cotxe teledirigit per 4'5€, que té motor funcionat a 4'5V-5V, com que les sortides del Arduino son de 5V simplifica molt tot el procés.

Motors.

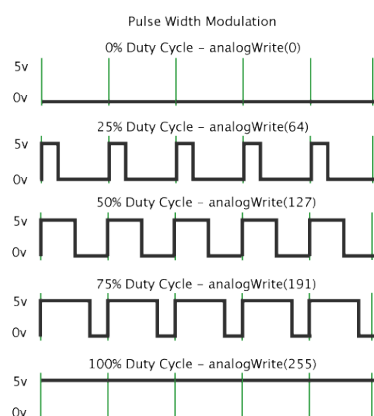
En electrònica digital resulta molt còmode fer servir motors de corrent contínua (DC) enlloc dels de corrent alterna. Aquests últims son molt millor en quant a rendiment i potència, però com que tenim que treballar amb motors petits no ens és necessari.

Els motors DC consisteixen amb una sèrie d'electroimants rodejats de bobines inductives que el fan girar. Es poden classificar de moltes maneres segons la tecnologia que fan servir per fer l'ho rodar encara que en el món de l'Arduino se sol utilitzar una classificació més orientada a com controlar els motors: simples, pas a pas i servo motors.

Els motors pas a pas son motors que se'ls pot indicar quan han de girar, habitualment tenen una sèrie de passos de volta, o sigui si se li fa moure 3 passos a la dreta farà un 3/100 de volta a la dreta; aquest tipus de motors es fan servir molt per processos mecanitzats com les impressores en 3D.

Els servomotors són motors extremadament precisos, i lents, que es fan servir molt en robòtica. Ens permeten controlar molt un moviment però habitualment ni tan sols permeten fer una volta completa. En RC es podrien fer servir pel control de direcció si no fossin de moviment lent.

En RC habitualment es fan servir motors de continua bidireccionals senzills. Tenen dos pols en els que per a un se l'hi introdueix VCC i per l'altre massa fent que es mogui, si intercanviem el pol que rep massa i el que rep l'alimentació aquest es mou en sentit contrari. El control de direcció de la majoria de cotxes teledirigits de baixa gamma que s'han desmuntat fan servir un motor d'aquests per avançar o retrocedir, i altre per girar en les dos direccions però delimitant físicament l'angle de gir.



http://arduino.cc/es/Tutorial/PWM#U0-OP_I_uZ4

L'Arduino té sis sortides digitals PWM que simulen ser analògiques fent que la sortida estigui a 5V més o menys temps durant un cicle. La sortida és purament digital però si hi connectem un element analògic com un led o un motor DC, aquest dona la sensació que brilla amb menys intensitat o que va diferents velocitats. Això pot resultar útil si es vol controlar el motor de forma que acceleri si porta molta estona anant endavant o si volem ficar l'opció de "turbo". S'escapa una mica del objectiu del TFC però de totes formes procuraré fer servir ports PWM i fixar la seva velocitat de rotació amb una variable fàcil de canviar si es vol.

Sistemes de comunicació per a Radio Control (RC) amb Arduino.**Motors. Primer Prototip.**

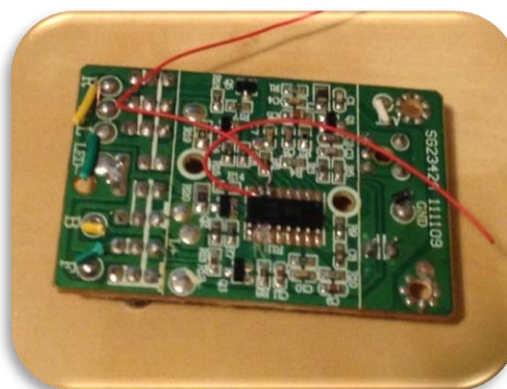
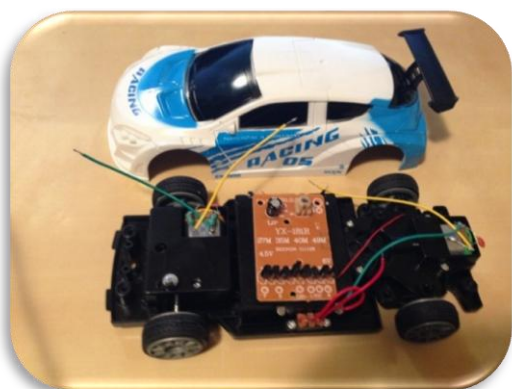
El primer intent consistia en fer servir la circuiteria existent canviant tan sols la part de RC, que era la part d'estudi del TFC. Aconseguint quelcom molt senzill i ràpid, cosa molt interessant per les primeres proves.

VO2	1	16	VI2
GND	2	15	VO1
SI	3	14	VI1
OSCI	4	13	V _{CC}
OSCO	5	12	Turbo
Right	6	11	Forward
Left	7	10	Backward
RD	8	9	LD

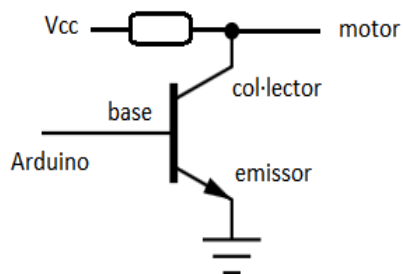
El procés era relativament fàcil, la majoria del vehicles comercials antics, i els actuals de baix cost, tenen un esquema semblant, amb un xip genèric RX2.

Aquest té 4 entrades que venen de la part RC de la placa i que nosaltres podem controlar des de Arduino si les hi desviem, incloent-hi també la GND per tindre una referència de voltatge comú.

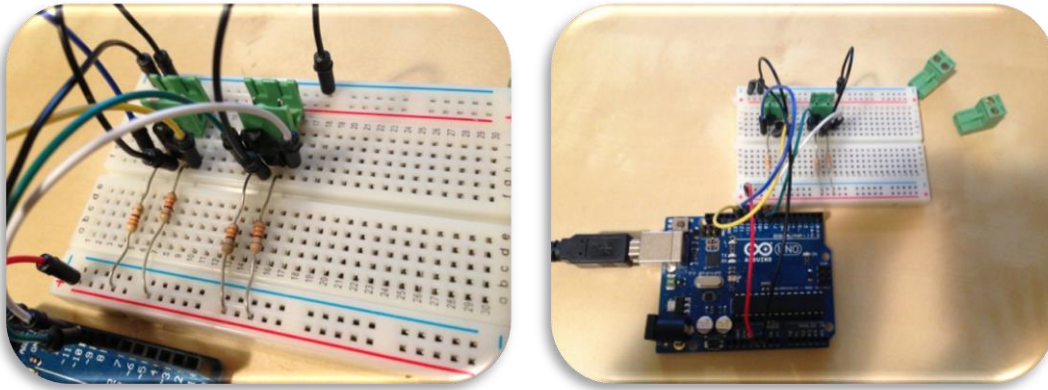
La proposta és fàcil d'implementar però presenta problemes, a part del estètic i legal de reutilitzar una placa que té un altre propòsit. Les connexions soldades a un xip de dimensions reduïdes donava problemes causant que alguns dels cables es desconnectessin amb la manipulació i que finalment es danyessin en el procés.

**Motors. Segon prototip.**

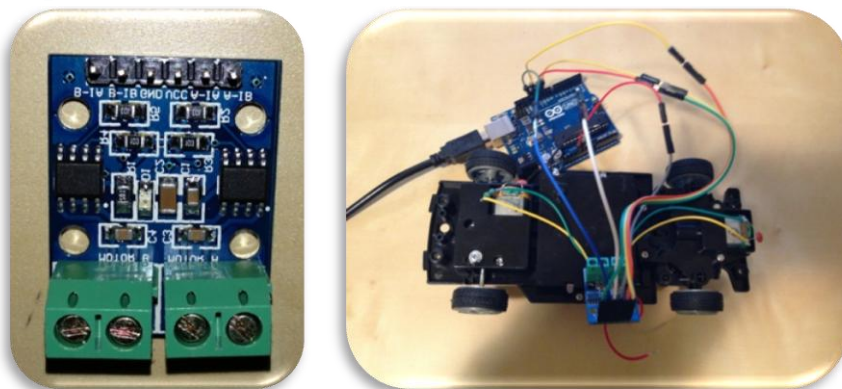
A continuació es va descartar el control de motors i s'optà per un disseny propi. Tan sols es tracta de fer servir l'Arduino per controlar un transistor, amb el que fem que l'entrada rebi o no l'alimentació de Vcc. Aquest disseny ens permet a més de fer que el motor rebi una alimentació molt diferent de la emet l'Arduino, lo que s'anomena etapa de potència, per exemple si els motors funcionessin a 12V com fan alguns models de cotxes de RC comercials.



El cotxe que vaig servir té tan sols 2 motors per lo que s'ha de connectar un circuit com l'anterior a cada extrem del motor. Llavors tenim que fer que l'Arduino envii tensió a un dels extrems però no a l'altre, fent que el cotxe giri a un costat o avanci, si volem l'efecte contrari tan sols intercanviant el valor de les dos sortides del Arduino que controlen el motor.

Sistemes de comunicació per a Radio Control (RC) amb Arduino.

El sistema funciona, i fins hi tot fent servir el propi Arduino com a sistema d'alimentació, però és poc escalable. Els motors, sobretot al canviar de direcció de funcionament, poden crear corrents inverses, ja que en el fons no són res més que bobines, i crear problemes en tota la circuiteria. La forma senzilla de solucionar-ho és ficant un diode per desviar les corrents inverses i recomanablement un condensador per suavitzar les transicions.

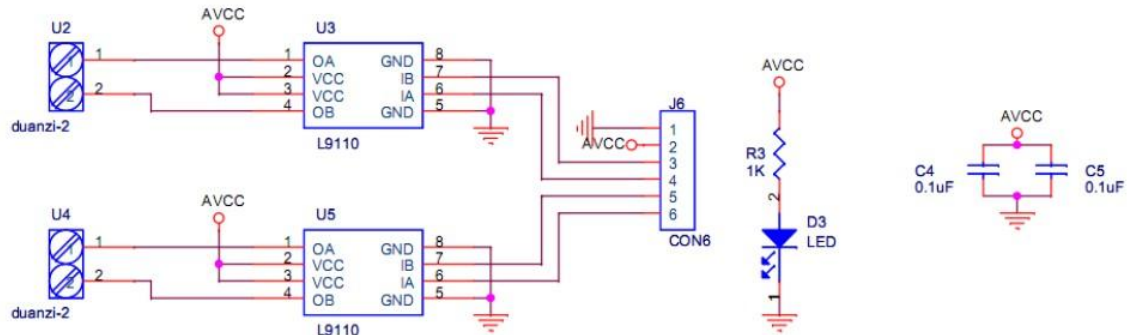
Motors. Proposta Final.

Finalment s'ha fet servir una etapa de control de motors ja implementada i especialment preparada per Arduino, se les anomena drivers, i no són res més que una sèrie de transistors de forma molt semblant a la de la solució anterior.



Sistemes de comunicació per a Radio Control (RC) amb Arduino.

Al fer servir una solució ja implementada dona al sistema la protecció necessària, a part de ser una solució molt barata i elegant. El seu funcionament és molt senzill, té 4 sortides per controlar 4 motors (en aquests cas els 2 motor bidireccionals) i 6 entrades. La part centrals de la entrada són l'alimentació, que no ha de perquè ser la mateixa que en l'Arduino, i la massa, que si que ha de estar en comú amb l'Arduino.

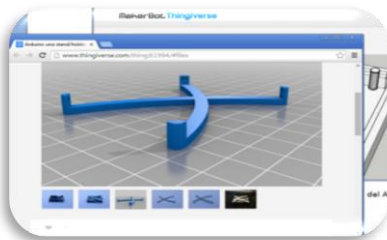


Alimentació.

El cotxe venia amb un carregador de 3 piles de 1'5V per lo que com a màxim ens pot subministrar 4'5V. L'alimentació recomanable del Arduino és entre 6V i 12V, per lo que hem d'augmentar la seva capacitat o ficar una font d'alimentació suplementària. He optat per connectar un cargador extern de 2 piles en sèrie per proporcionar l'energia necessària. L'altre opció era ignorar el sistema de cargador actual i fer servir una pila de 9v típica en Arduino. M'he decantat pel primer sistema per aprofitar l'interruptor que venia amb el cotxe i perquè les piles de 1'5V són molt més econòmiques i fàcils de trobar.

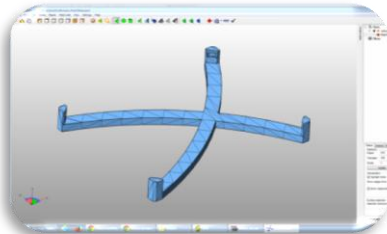


Teòricament així aconseguim un voltatge de 7'5V ($3 \times 1'5V + 2 \times 1'5V$). Curiosament si es comprova de forma empírica obtenim una voltatge lleugerament superior, però com que està dintre del marge de treball del Arduino, i aquest té nombrosos sistemes de protecció, és una tensió perfecte per a treballar. De fet habitualment la majoria de components electrònics tenen toleràncies de l'ordre del 10%.

Sistemes de comunicació per a Radio Control (RC) amb Arduino.**Suport Arduino.**

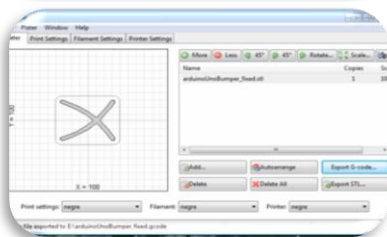
Per tal de mantenir la placa d'Arduino Uno fixa s'ha fet servir una impressora en 3D i per fer suport a mida en el xassís del cotxe. Primerament s'ha descarregat un disseny existent del suport:

<http://www.thingiverse.com/thing:61994>

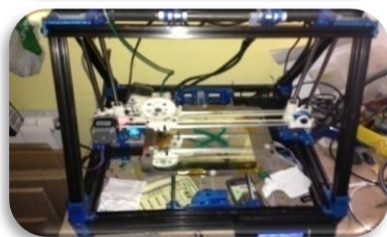


Per comprovar que no tingués cap error de disseny, cares obertes, ni problemes d'estructura s'ha testejat la forma amb el NetFabb Studio.

On es pot observar clarament la traducció del model a una estructura de triangles.



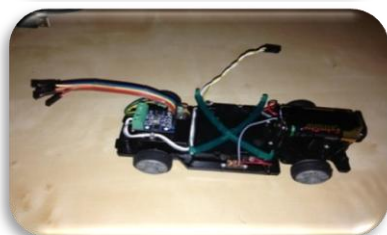
Fent servir el programa Slic3r he traduït el model tridimensional a ordres d'impressora o codi GCode, on s'inclouen ordres simples per controlar els 5 motors de la impressora, la temperatura del extrusor i la del "llit".



Seguidament he fet servir el codi GCode en una impressora 3D. En aquest cas un Mendel Max adaptada per la UPC, la anomenada RepRapBCN, concretament la segona versió comercial del 2013.



En la imatge podem veure la impressió finalitzada, encara que sense polir. El material és PLA, que no és res més un Bio-Plàstic obtingut amb elements com la tapioca. I en un color verd perquè destaquí en el prototip del cotxe, en producció seria recomanable el negre per integrar-se millor.



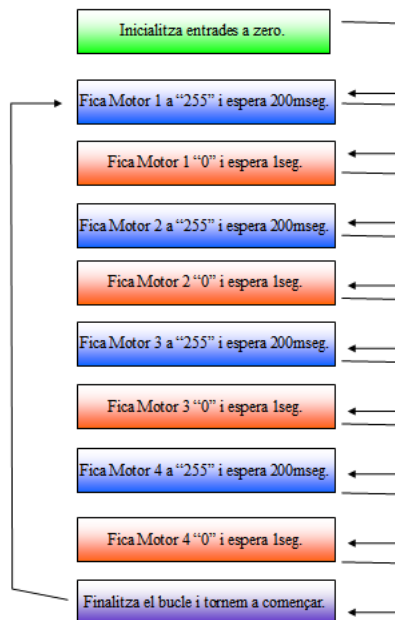
Per acabar, després d'eliminar els residus d'impressió, he enganxat amb adhesiu especial per a plàstic de modelisme la base al cotxe i he comprovat que encaixi.



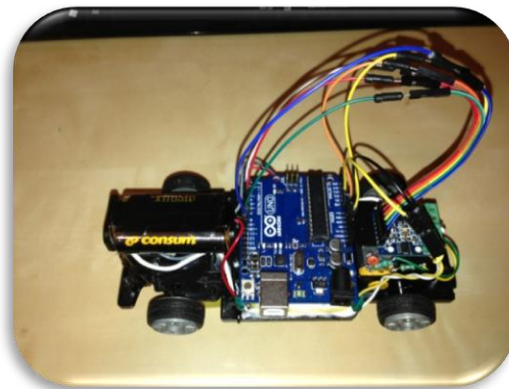
Aquest suport ens dona l'estabilitat que necessitem sense tindre que tenir una estructura complicada o tenir que recórrer a mètodes més permanents.

Sistemes de comunicació per a Radio Control (RC) amb Arduino.**Control amb Arduino.**

Gràcies a la placa de control de motors el control del cotxe des del Arduino és simplifica a 4 entrades una per motor. Pel que simplifica bastant el control ja que tan sols s'han de connectar als ports PWM.

Moviments bàsics.

Per comprovar els funcionament dels motors, la programació amb Arduino i el sistema d'alimentació autònom es farà una petita prova. Es connectarà les quatre entrades dels motors als ports PWM: 5,6,10 i 11; i la alimentació del cotxe als ports Vin de l'Arduino i al GND. El codi es troba al Annex 1. Part del Codi 1.

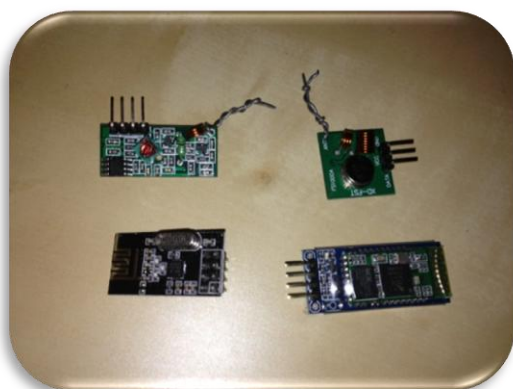


Al connectar-l'ho el cotxe mou un dels motors durant 200mseg, s'espera 1segon, i passa al següent motor, i així amb els 4. En el vídeo "motorsV0.MOV" de l'Annex se'n pot veure el funcionament. Gràcies a això ja hem comprovat que tot funciona i ens ha permès veure la posició real dels motors:

```

motorsv1
#define esquerra 5 // B-IA, cable marró
#define dreta 6 // B-IB, cable vermell
#define avant 10 // A-IA, cable verd
#define retro 11 // A-IB, cable blau
  
```



Sistemes de comunicació per a Radio Control (RC) amb Arduino.**RC i Arduino.**

El món del Arduino s'ha popularitzat tant que ens permet de fer servir petites plaques de gran qualitat a preus ridículament baixos. Per lo que són una bona opció.

En la primera fila de la imatge un receptor i un emissor de 433MHz (uns 3€ ambdós), el primer component de la segona fila un emissor/receptor genèric de 2'4GHz (menys d'1€) i el darrer un emissor/receptor de Bluetooth (2'4GHz) amb la codificació integrada (uns 6'5€).

Arduino a 433MHz.

Per tal de fer una prova de comunicació per radiofreqüència farem servir un emissor i un receptor comercials a 433MHz, com els de la imatge. Aquest son fàcils de fer servir, sobretot amb l'ús de la biblioteca VirtualWire.

En aquesta prova faré servir dos Arduinos per comunicar-se entre ells. El primer, un Arduino Uno, controlarà el cotxe. Tindrà el receptor i en tot moment estarà a l'espera del emissor.

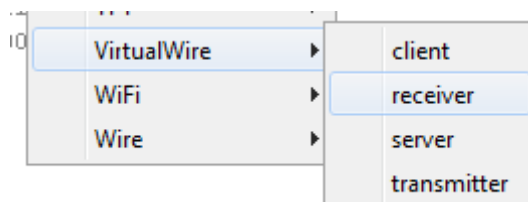


L'emissor serà un Arduino Mega amb una matriu de pulsadors de 4x4. En principi tan sols farà servir 4 dels botons (endavant, endarrere, dreta i esquerra), però podria fer servir els botons per fer alguna acció predeterminada, o tindre varies velocitats depenent de quin botó apremem.

Tant la emissora, la placa petita, com el receptor, la placa gran, tan sols tenen 3 pins VCC, GND i Dades; curiosament el receptor té con pins de dades però que contenen la mateixa informació.

Prova de comunicació en 433MHz.

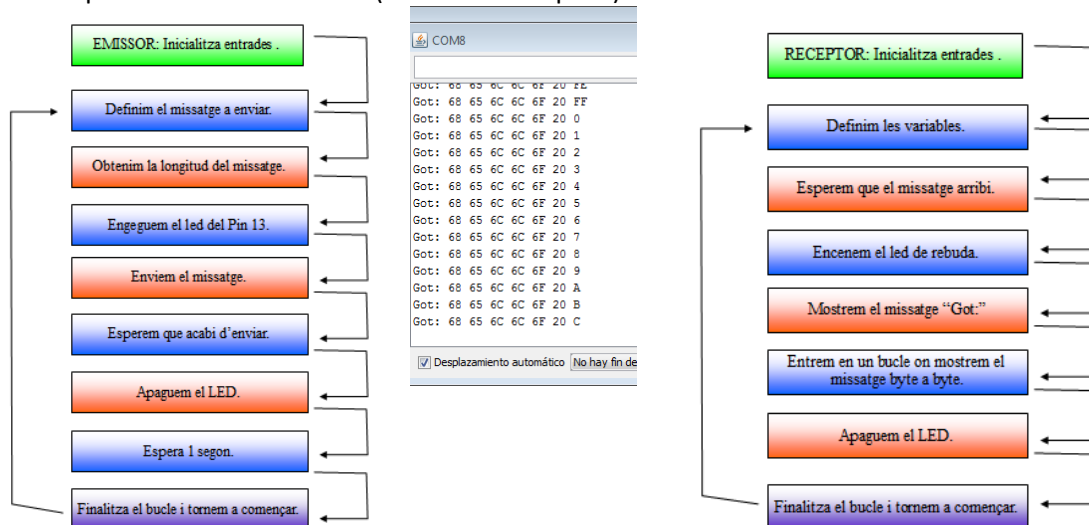
Amb un parell dels exemples d'ús de la web de VirtualWire (receiver and transmitter) comprovem que el dispositiu funciona, i faig una primera introducció al ús de la llibreria.



El codi és molt senzill i ens permet tan veure com en cada bucle s'envia el "hello" acompanyat d'un contador.

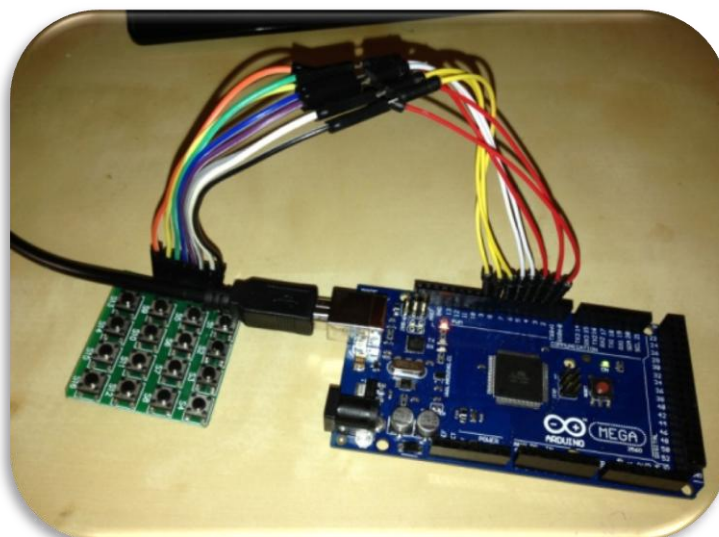
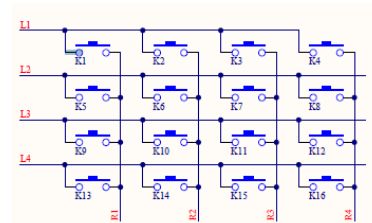
Sistemes de comunicació per a Radio Control (RC) amb Arduino.

El codi es pot trobar com a Codi 2 (emissor i receptor) en l'Annex1.

**Exemple de la matriu de pulsadors.**

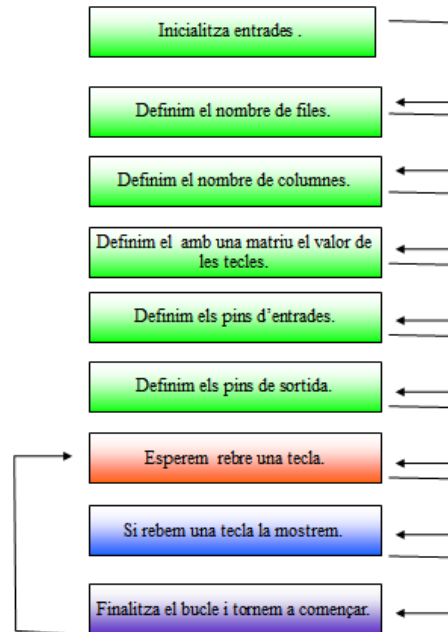
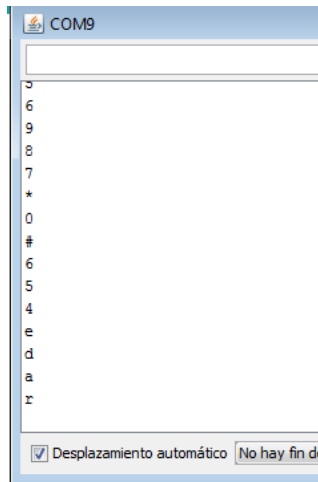
Per fer una primera prova es provarà de fer servir una matriu de pulsadors per tal d'obtenir una sèrie de valors d'entrada per després ser transmesos.

Es farà servir una matriu de pulsadors comercial, un Arduino MEGA i la llibreria Keypad.h d'Arduino per tal de provar el seu funcionament (<http://playground.arduino.cc/code/Keypad>).

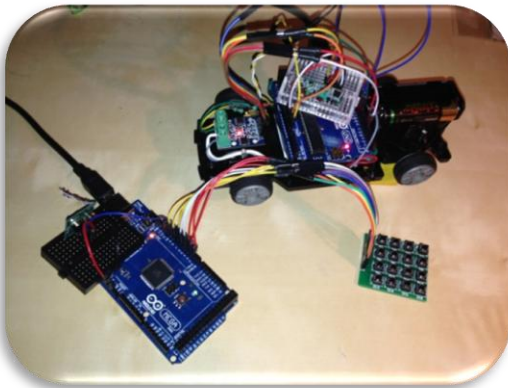


Sistemes de comunicació per a Radio Control (RC) amb Arduino.

El codi el podem trobar com a Codi 3 en l'Annex 1.



S'ha modificat lleugerament el codi, ja que el de la llibreria tracta una matriu de 4x3 o teclat mínim, s'han afegit els valors "e,d,a i r" per tal d'assignar valors a la fila buida i de pas donar un sentit a la prova, ja que seran els caràcters que es faran servir per controlar el cotxe.

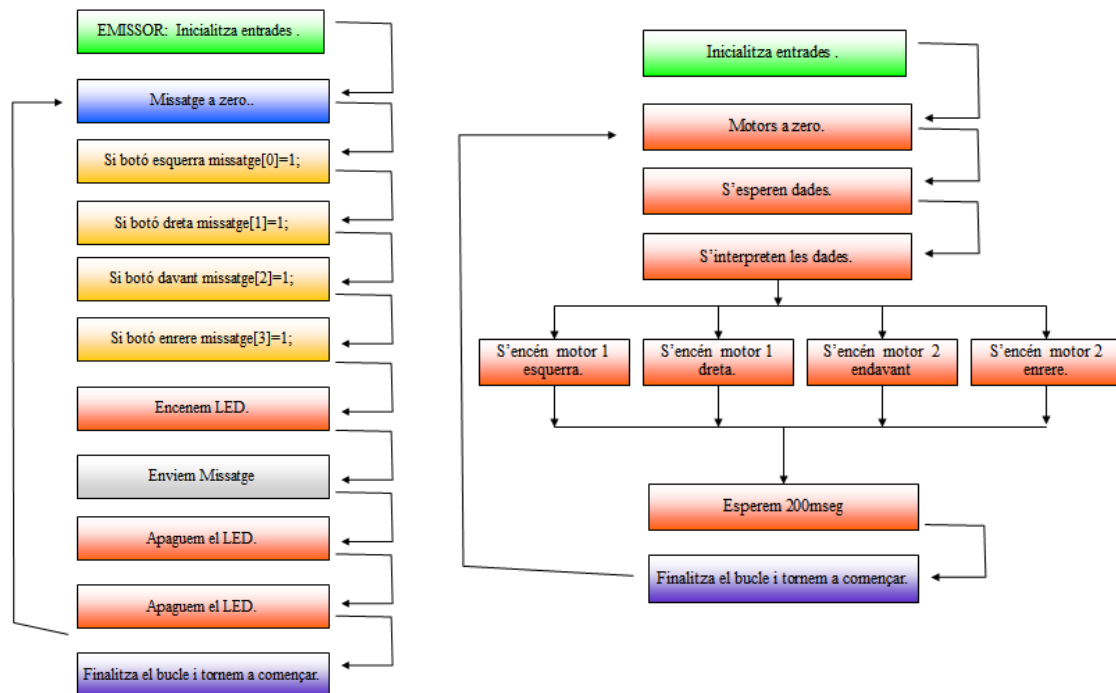
Matriu de Polsadors i 433MHz.

Després s'integren els dos exemples i es modifica el codi per controlar amb un Arduino MEGA i una matriu de polsadors, el cotxe teledirigit que estarà controlat per un Arduino UNO i el control de motors.

S'ha modificat l'array de caràcters del transmissor per incloure els 4 camps que ens interessen i un extra de control.

També s'ha modificat el receptor per incloure el control de motors prèviament testejat d'una forma senzilla amb un valor del array per direcció dels motors.

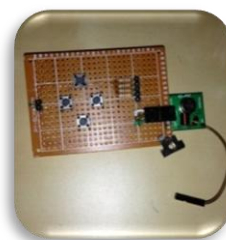
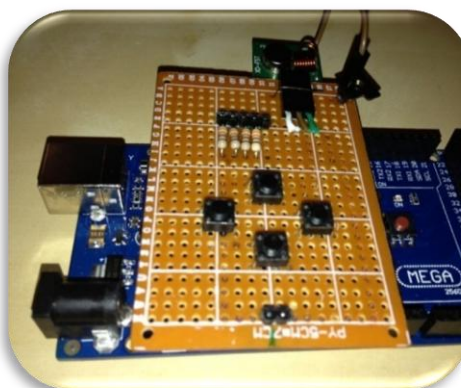
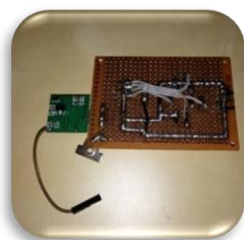
El codi el podem trobar com a Codi 4 (emissor i receptor) en l'Annex 1.

Sistemes de comunicació per a Radio Control (RC) amb Arduino.**Shield de Comandament i 433MHz.**

Farem una versió del apartat anterior però amb pulsadors i tot integrat en una placa o Shield que va directament connectada al Arduino MEGA sense necessitat de cap altre connexió.

S'aprofitarà també per connectar una antena una mica millor al emissor ja que fins ara es feia servir un tros de cable directament al anclatge.

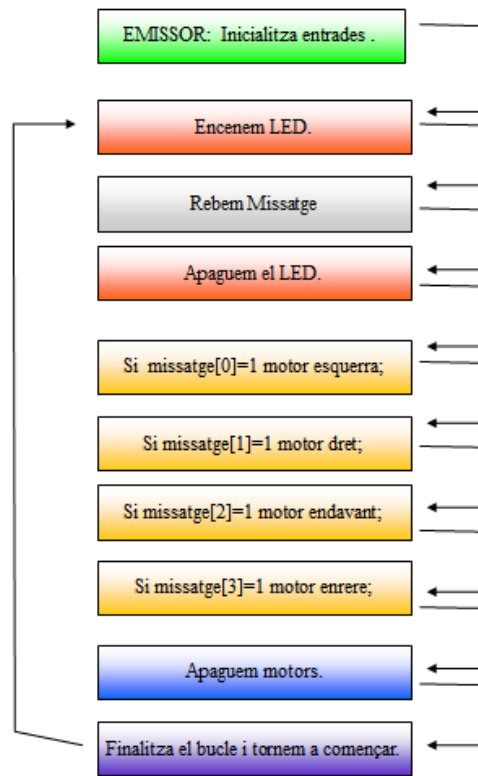
La circuiteria és molt senzilla tan sols 4 pulsadors connectats a VCC en un extrem i a l'altre tant al Arduino com a una resistència a massa. El mòdul de 433 va connectat a un altre pin i compartint l'alimentació.



Tal com podem observar en els vídeos (rc433V00.MOV,rc433V01.MOV) del Annex el seu funcionament és semblant al de la matriu de pulsadors, amb la diferència que ara es poden moure els dos motors a la vegada, o sigui girar a la vegada que es mou el vehicle.

Sistemes de comunicació per a Radio Control (RC) amb Arduino.

Podem trobar el codi com a Codi 5 en l'Annex 1, que és el codi de l'emissor, com a receptor es mantindrà el Codi 4 (receptor).



Sistemes de comunicació per a Radio Control (RC) amb Arduino.**Arduino Bluetooth i Android**

Ara es farà servir un mòdul comercial de Bluetooth per tal de comunicar l'Arduino amb el dispositiu Android, en aquest cas una Tablet.

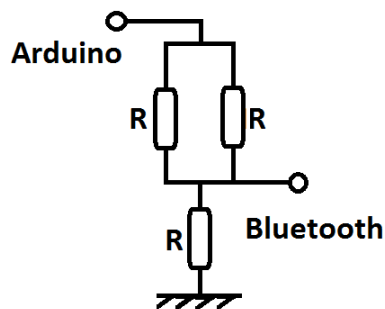
El mòdul Bluetooth té un funcionament sèrie, per lo que es podria connectar directament als



ports sèrie del Arduino, però aquest mòdul té la peculiaritat de funcionar a 3'3V per lo que s'ha d'adaptar la TX, es farà mitjançant un divisió de tensió senzill amb resistències i connectar-li-ho als ports 0 i 1.

Connecto la sortida de TX del Arduino a un parell de resistències de valor igual (330Ω) en paral·lel en l'altre extrem dels resistor puc connectar a la RX del dispositiu, i una resistència d'igual valor a massa; com que la sortida de l'Arduino serà o 5v o 0v a l'altre extrem tindrem 3'3V.

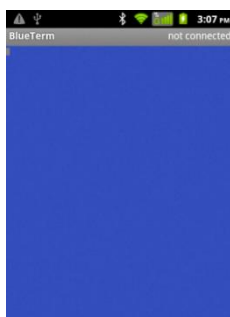
El procés en RX no serà necessari ja que l'Arduino ho fa de forma automàtica.



$$R_{\text{paral} \cdot \text{lel}} = \frac{R * R}{R + R} = R/2$$

$$V_{\text{bluetooth}} = I * R = (V_{\text{arduino}} / (R + R/2)) * R = (V_{\text{arduino}} / (3R/2)) * R = (2/3) V_{\text{arduino}}.$$

O sigui entre 0V i 3'3V.



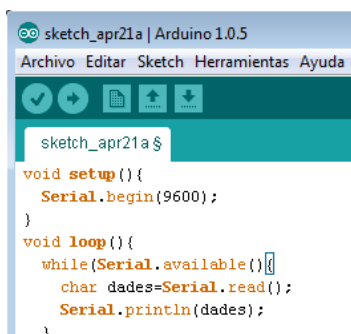
Per tal de comunicar l'Arduino amb l'Android faré servir un programa en l'Android anomenat BlueTerm, que no és més que un terminal sèrie que fa servir Bluetooth.

En el programa BlueTerm busco l'Arduino i el trobo amb aquestes dades:

Device: HC-06

MAC: 00:14:02:13:15:12

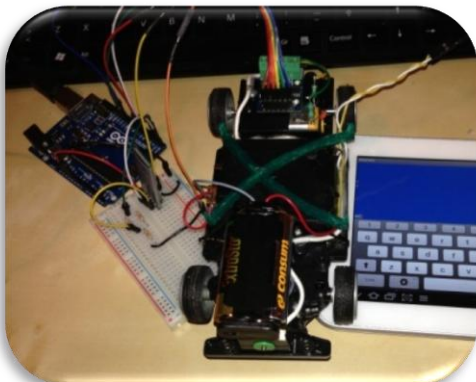
Pass: 1234



Seguidament s'ha escrit un petit codi en Arduino per comprovar la connectivitat. Com es pot observar no hi ha un control específic de la comunicació ni biblioteques addicionals, simplement es fa servir el port sèrie per defecte del Arduino. Per cert, aquest port també el fa servir per comunicar-se amb el USB per lo que s'ha de desconnectar el mòdul Bluetooth durant la transferència del codi al Arduino.

Sistemes de comunicació per a Radio Control (RC) amb Arduino.**Control del Cotxe amb Arduino i Android.**

Ara testejarem la part de control de motors per tal de fer una prova de control del vehicle des del dispositiu Android.

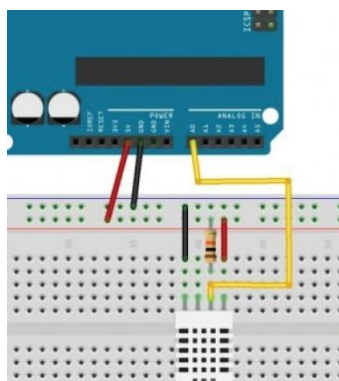
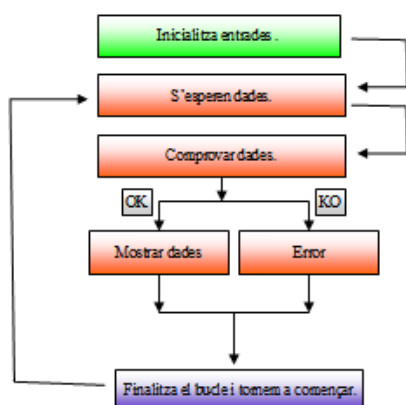


Des de la tablet s'envien els caràcters clau, com: e,d,a i r; per tal de moure el vehicle i el resultat és la que es pot observar en el vídeo del Annex: rcBluetoothV00.mp4.

Es farà servir un esquema idèntic al fet servir en la matriu de pulsadors a 433Mhz, no obstant i que el codi sigui diferent, on fem servir el port sèrie enlloc de configurar un software serial, el codi es pot consultar a l'Annex 1 amb el nom de Codi 6.

Sensor d'humitat i temperatura.

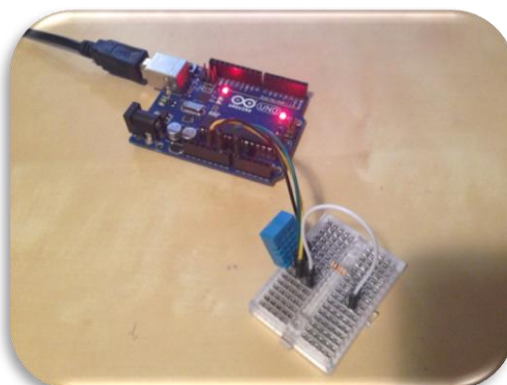
És un dels sensors bàsics que s'inclouen en casi tots els kits d'iniciació a l'Arduino i que per tant està molt documentat. S'ha fet servir com a base l'exemple del portal "practicando Arduino" modificant-l'ho per lleugerament. El codi es pot trobar com a Codi 7 en l'Annex 1.

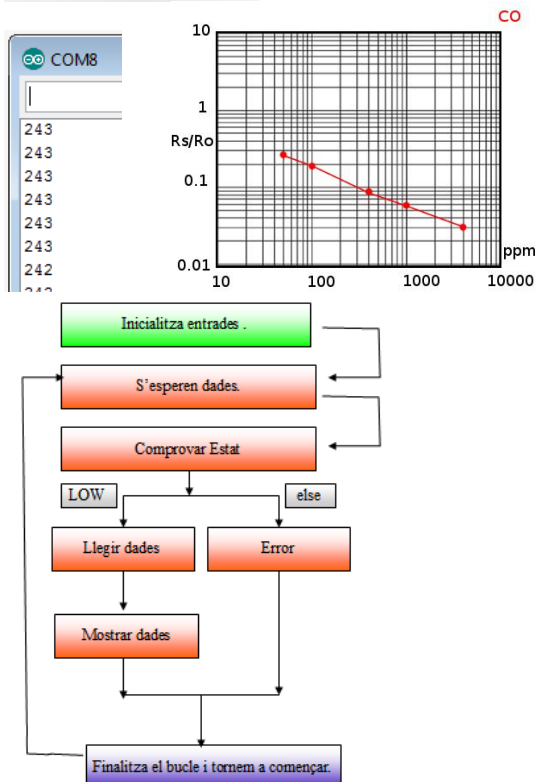
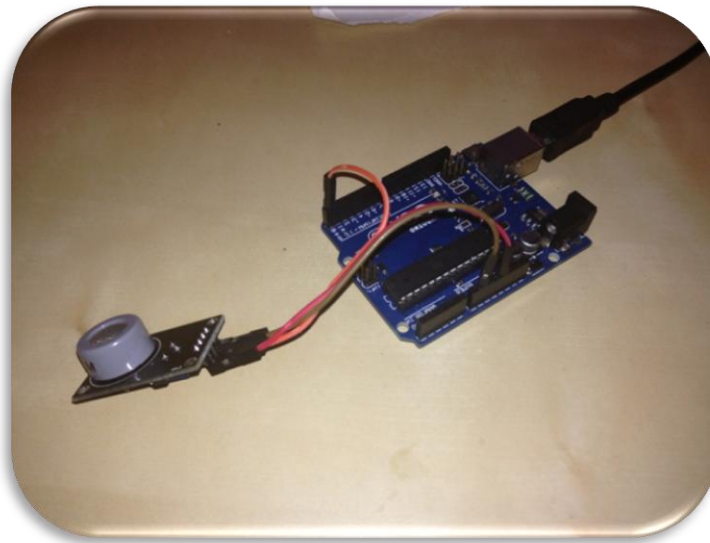


<http://www.practicandoarduino.com/en/author/practicandoarduino/>

```

COM8
temperature:25.00 humidity:80.00
temperature:25.00 humidity:80.00
temperature:25.00 humidity:80.00
temperature:25.00 humidity:80.00
temperature:25.00 humidity:80.00
temperature:25.00 humidity:80.00
temperature:25.00 humidity:80.00
temperature:25.00 humidity:80.00
temperature:25.00 humidity:80.00
temperature:25.00 humidity:80.00
temperature:25.00 humidity:80.00
temperature:25.00 humidity:80.00
  
```



Detector de monòxid de carboni.

El sensor MQ-7 és un resistor variable sensible a la quantitat de monòxid de carboni. El seu integrat ja inclou l'alimentació, per lo que no tenim cap circuiteria a ficar i es pot connectar directament a una de les entrades del Arduino.

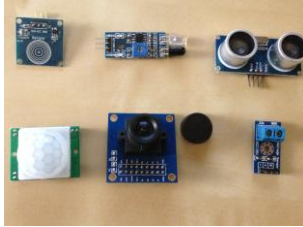
El monòxid de carboni, CO, es forma quan hi ha una combustió de carboni però en entorns amb insuficient nivell d'oxigen, com pot ser un motor o caldera en un entorn tancat. També es dona en grans quantitats per fenòmens naturals en llocs amb escassa ventilació: grutes, túnels, ... per lo que pot ser interessant tindre una mesura de la seva concentració per tal de preservar la salut si s tracta d'un lloc de treball.

Podem observar la gràfica on s'observa la pèrdua de resistència a mesura que augmenta el nivell de CO, el ambient actual segons la mesura, que hem realitzat, és de 243ppm,.

Es pot comprovar el codi en l'Annex 1 com a Codi 8, que consisteix en l'exemple d'ús de la llibreria del dispositiu.

Sistemes de comunicació per a Radio Control (RC) amb Arduino.**Possibles millores.**

Es podrien fer servir gran multitud de sensors, plaques de control alternatives o simplement aplicacions diferents, aquí se'n poden veure uns quants exemples.

**Sensors Varis:**

Se l'hi pot donar certa utilitat al projecte fent una petita millora que ens pot donar cert valor afegit al producte és la de introduir sensors al vehicle per a usos concrets. En la fotografia: sensor de contacte, de proximitat per infrarojos, de proximitat per ultrasons, de moviment, una càmera i un sensor de voltatge.

**Detectors de gasos:**

Fent servir la mateixa tecnologia que el MQ-7 existeixen tota una sèrie de sensors amb les mateixes característiques que podrien ser fets servir per multitud d'usos:

MQ-2: Sensible al metà, butà, LPG i fum.

MQ-3: Sensible al Alcohol, Etanol i fum

MQ-4: Sensible al Metà, i gas CNG

MQ-5: Sensible al gas natural i al LPG.

MQ-6: Sensible al gas butà i al LPG.

MQ-8: Sensible al hidrogen.

MQ-9: Sensible al monòxid de carboni i gasos inflamables.

i hi ha les sèries MQ1xx, MQ3xxA, MGxx, ...

**Ús de la RaspBerryPi:**

Es podria fer una sèrie de proves per suplir l'Arduino o l'equip Android per una RaspBerryPi, que ens dona una sèrie d'opcions noves.

**Quadcòpter.**

Hi ha molts projectes d'implementació de quadcòpters en Arduino, una opció seria la d'introduir els sensors de la part final per obtenir telemetria mitjançant el vehicle aeri.

Realitat Virtual.

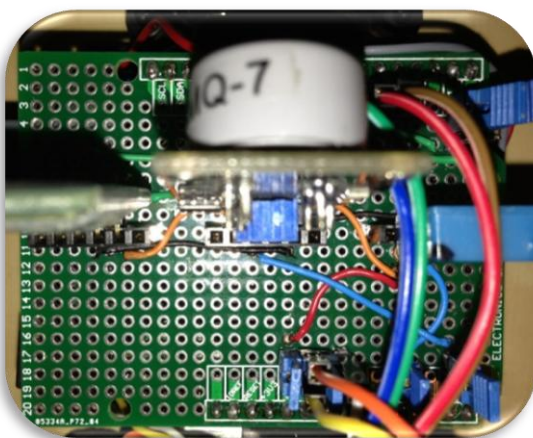
Ficant una càmera al Arduino podríem fer un videojoc amb Android on es controli el cotxe a la vegada que veiem per pantalla el mateix que el vehicle.

Prototip Arduino - Android.

En aquesta part aprofitarem part de les parts teòricopràctiques per implementar una possible aplicació real dels coneixements adquirits en les investigacions prèvies per fer un producte que es pugui comercialitzar.

Interfície Arduino.

Es desenvoluparà una placa de baix cost que ens permetrà que amb cost mínim poder subministrar un detector mòbil de temperatura, humitat i monòxid de carboni, que pot resultar útil per fer una primera exploració de mines, sales de calderes, ...

Shield de control.

Aprofitant els circuits de l'apartat anterior farem una placa que es pugui connectar al Arduino Uno i amb ella controlar els motors, obtenir telemetria dels sensors (humitat, temperatura i CO), a més de rebre i enviar mitjançant Bluetooth.

Per tal de facilitar el control s'ha posicionat una sèrie de "jumpers" per tal de poder desconnectar de forma individual parts senceres de circuit.

Alimentarem la placa mitjançant la Vin, entrada - sortida, però amb un jumper per poder-la desconnectar. Es col·locarà 2 jumpers en els ports 0 i 1, ja que són els ports sèrie que també es faran servir en la comunicació USB i que

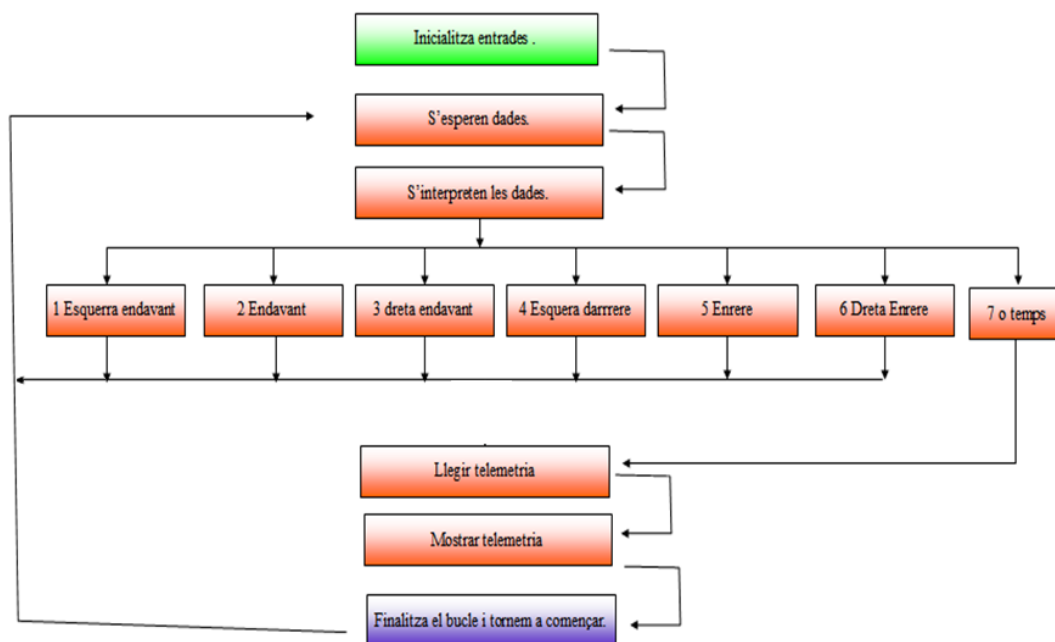
Sistemes de comunicació per a Radio Control (RC) amb Arduino.

si està connectada quan es fa la transmissió de dades dona problemes per lo que poder-les desconnectar fàcilment és molt útil.

Encara que al tindre la Vin és una mica redundant, també s'ha ficat un jumper a l'entrada de 5V amb la que alimentarem la majoria dels components.

Encara que no es pugui observar molt bé a la fotografia s'ha col·locat un resistor de 10K a la placa per el sensor DHT i un reductor de corrent amb 3 resistors per el MQ-7;

Gràcies als mòduls utilitzats en queda una circuiteria molt neta i amb pocs components afegits, per tant amb menys fonts de possibles errors de muntatge, és un dels grans avantatges de fer prototips fent servir tecnologia Arduino.

Codi Arduino.

En l'Annex 2 podem observar el codi complert. Esperarem dades o el missatge corresponent a que el temps màxim ha passat en aquest cas s'obtindrà la telemetria de les tres unitats de mesura i s'enviarà al dispositiu Android des de la placa Arduino.

Sistemes de comunicació per a Radio Control (RC) amb Arduino.**Interfície en Android**

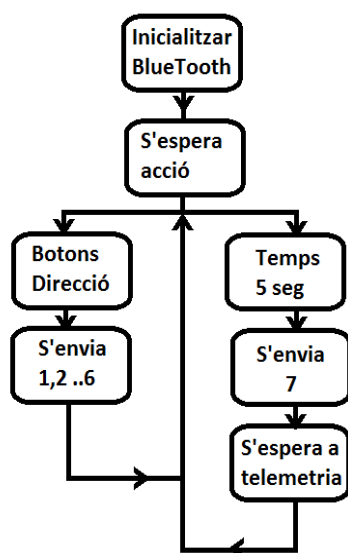
S'implementarà una solució Android per tal de rebre telemetria del dispositiu i controlar-ne els motors. En ella podrem veure a part de logos (com el dissenyat per aquesta aplicació de l'esquerra) i demés, la telemetria de temperatura, humitat i CO de la Shield del cotxe RC.

S'ha dissenyat un logo per la aplicació Android consistent amb un Andy del color corporatiu del Arduino, amb el seu logotip al pit i unes rodes als peus com si fossin patins per indicar que es tractarà del control d'un vehicle. Evidentment s'hauria d'assegurar la legalitat

d'ús, per tema llicència, abans de la seva comercialització.

Estructura Programa Android.

La cada vegada més popular plataforma Android ens ofereix varies eines per a programar aplicacions. En aquest cas s'ha provat la eina "oficial" Eclipse i l'entorn SDK, encara que la opció final s'ha programat fent servir el portal del MIT anomenat APP Inventor.

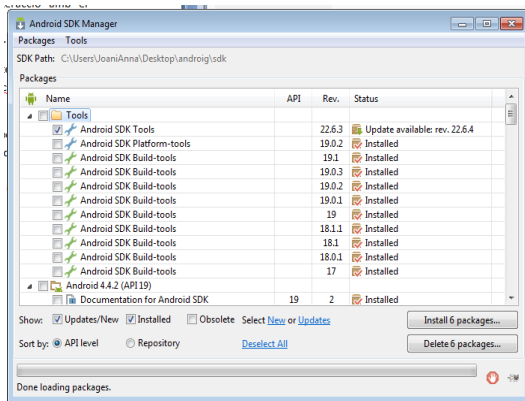


La aplicació tindrà tres parts clarament diferenciades. Per començar tindrem un botó que ens permetrà connectar-nos a bluetooth amb el vehicle a RC.

Si no hi ha hagut cap problema esperem una de les dos línies d'acció principals. La més clara son els sis botons de direcció (endavant-esquerra, endavant, endavant-dreta, enrere-esquerra, enrere, enrere-dreta), si s'apreta un d'ells l'Android enviarà al Arduino un condí numèric del 1 al 6 perquè aquest favi moure el cotxe.

1 Esquerra Endavant	2 Endavant	3 Dreta Endavant
4 Esquerra Enrere	5 Enrere	6 Dreta Enrere

La segona acció serà la de rebre telemetria, ho farem cada 5 segons, de forma que l'Android envi el codi "7" al Arduino, seguidament el programa s'esperarà a rebre les dades: humitat, temperatura i nivell de CO.

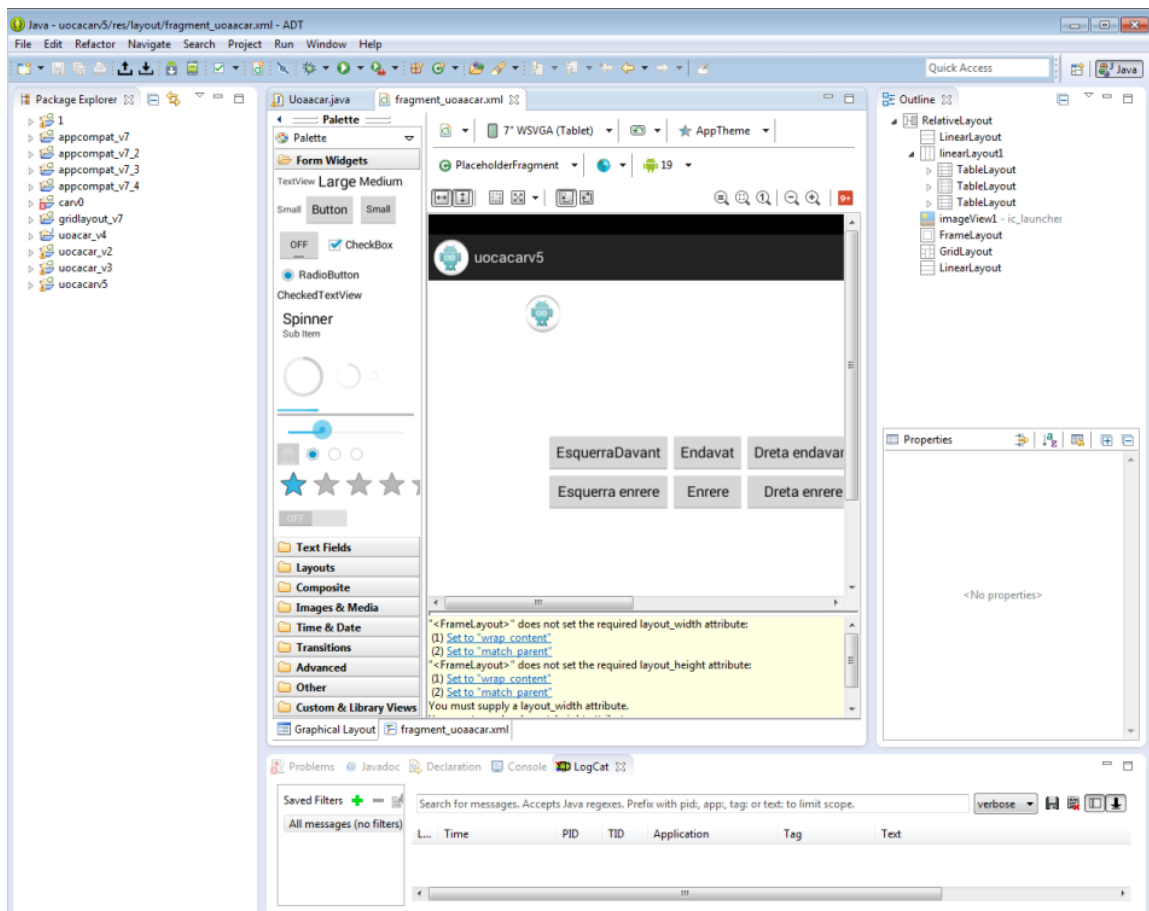
Sistemes de comunicació per a Radio Control (RC) amb Arduino.**SDK / Eclipse.**

Tot i que s'ha descartat la opció en aquest treball, és convenient, ja que es fa una aplicació amb Android de parlar de la seva plataforma de programació.

L'entorn consisteix amb un gestor de dispositius l'SDK, el qual permet la interacció amb el terminal real, en pot crear un de virtual, controla la versió dels drivers,....

L'altre part es l'entorn de programació WYSIWYG.

En el que podem fer un disseny ràpid ficant i traient elements, i la opció de passar directament al codi font. Per veure'n el seu funcionament s'ha fet un petit menú semblant al que s'ha de fer servir i es carrega a la tauleta. El codi font es troba en l'Annex 2, tot i que no es tracta del de la aplicació definitiva i és tan sols un primer esbós del programa.



Si passem a l'opció de codi ens permet programar manualment les aplicacions, per defecte es genera un codi en Java, tot i que es pot configurar perquè utilitzi com a llenguatge nadiu el C o el C++.

Sistemes de comunicació per a Radio Control (RC) amb Arduino.

```

Uoacarc.java  fragment_uoacarc.xml
...
super.onCreate(savedInstanceState);
setContentView(R.layout.activity_uoacarc);

if (savedInstanceState == null) {
    getSupportFragmentManager().beginTransaction()
        .add(R.id.container, new PlaceholderFragment()).commit();
}

@Override
public boolean onCreateOptionsMenu(Menu menu) {
    // Inflate the menu; this adds items to the action bar if it is present.
    getMenuInflater().inflate(R.menu.uoacarc, menu);
    return true;
}

@Override
public boolean onOptionsItemSelected(MenuItem item) {
    // Handle action bar item clicks here. The action bar will
    // automatically handle clicks on the Home/Up button, so long
    // as you specify a parent activity in AndroidManifest.xml.
    int id = item.getItemId();
    if (id == R.id.action_settings) {
        return true;
    }
    return super.onOptionsItemSelected(item);
}

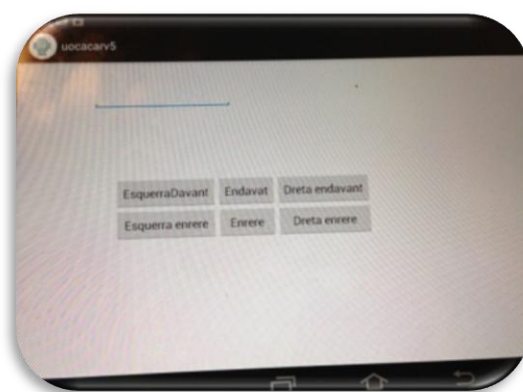
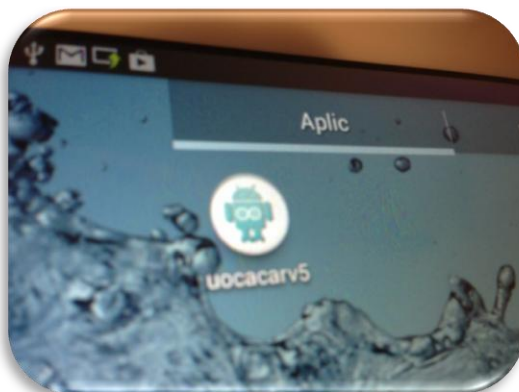
/**
 * A placeholder fragment containing a simple view.
 */
public static class PlaceholderFragment extends Fragment {

    public PlaceholderFragment() {
    }

    @Override
    public View onCreateView(LayoutInflater inflater, ViewGroup container,
        Bundle savedInstanceState) {
        View rootView = inflater.inflate(R.layout.fragment_uoacarc,
            container, false);
        return rootView;
    }
}

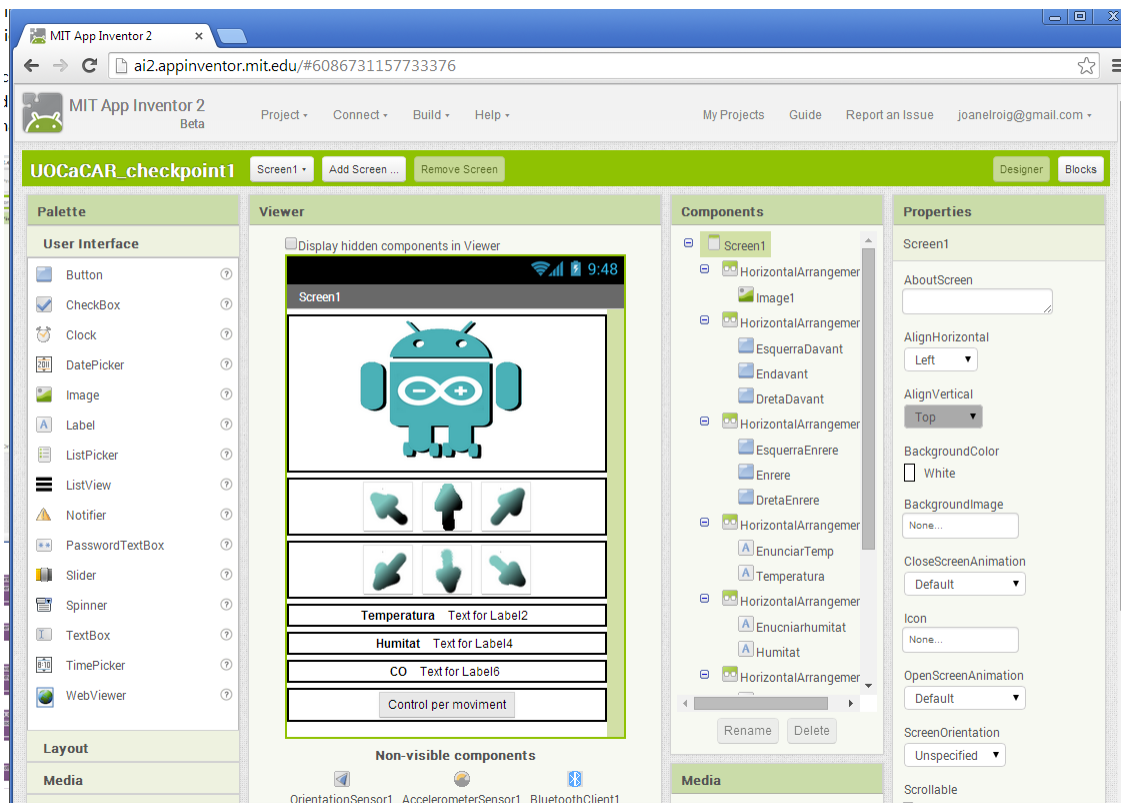
```

Podem observar com es veuria aquest petit exemple un cop carregat al terminal.



Sistemes de comunicació per a Radio Control (RC) amb Arduino.**APP Inventor 2.**

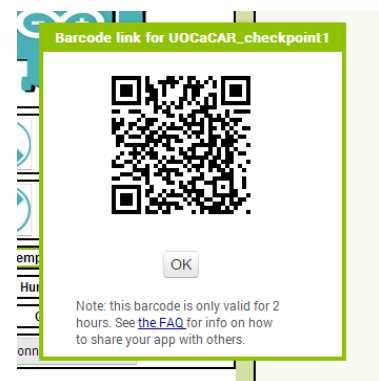
Per tal de simplificar la feina l'“App Inventor” que ens permet fer una aplicació funcional relativament ràpid i sense tocar codi font, ja que es programa de forma visual amb la utilització de blocs de programació. L'ha desenvolupat el MIT com a eina didàctica, i fins a cert punt orientada a introduir en el món de la programació dins l'escola. L'eina consta de dos mètodes de treball: mode Disseny i mode Blocs.

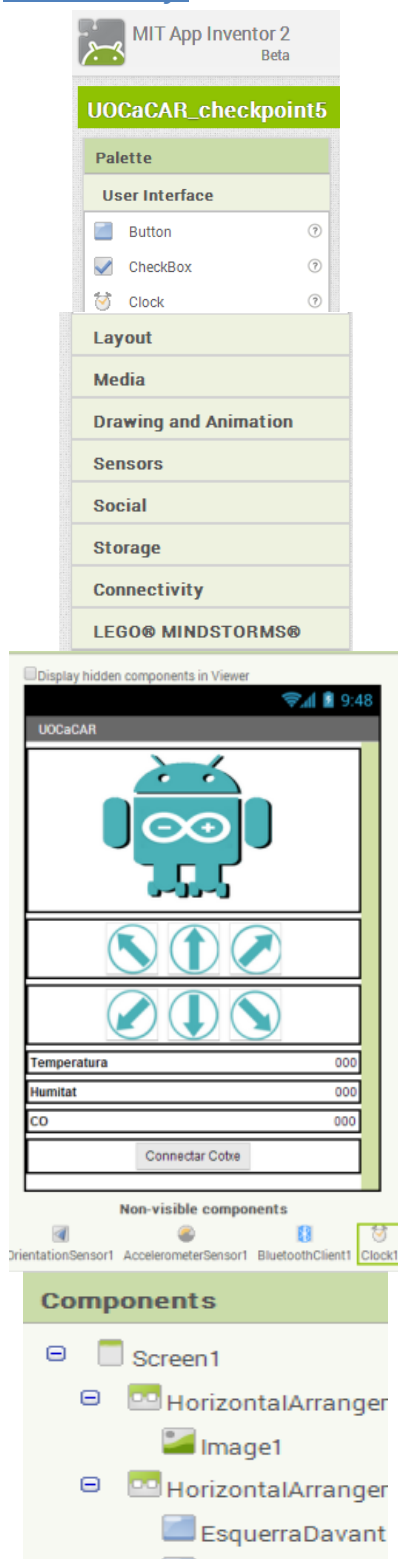


El programa es troba en la seva segona versió però en Beta com a APP Inventor 2, al APP Inventor 1 se l'hi ha canviat el nom a APP Inventor Clàssic. Tot i ser beta, i bastant nou, amb documentació abundant i molt clara.

Un dels problemes que he detectat ha estat la connexió USB, per tal d'interactuar amb la màquina, tot i ser un entorn web, s'han d'instal·lar una sèrie de controladors però que no contempen, encara, gaires models de mòbils. Per sort es disposa d'un altres mètodes de descarregar el programa. Un consisteix en instal·lar una aplicació en el dispositiu, descarregant-te la App, sigui generant un “apk” (com el de l'Annex 3) i copiant-lo a mà en el sistema d'arxius, o bé directament amb un codi QR que genera l'aplicació, però que tan sols té 2 hores de vida.

Tot i que és molt fàcil crear una aplicació, no ens ofereix un accés al codi font, la versió 1 permetia descarregar el codi font però era un codi propi que tant sols traduïa els blocs a llenguatge informàtic. Podem trobar el disseny de la App en l'Annex 4, en un format “.aia” propi del App Inventor.



Sistemes de comunicació per a Radio Control (RC) amb Arduino.**Mode Disseny.**

El mode disseny consisteix amb una pantalla WYSIWYG en la que es pot anar col·locant els elements que es troben a la part esquerra.

Hi tenim la majoria de funcions bàsiques se'ns permet dividir la pantalla i es disposa d'elements bàsics com botons, quadres text, quadres d'imatge, camp contrasenya, ...

També hi ha una sèrie de sensors, mètodes de comunicació, accés al rellotge, que no es veuen amb en el disseny però els s'han de requerir en aquesta pantalla. Com a curiositat es pot observar una etiqueta "LEGO® MINDSTORMS®" cosa que ens confirma el perfil d'eina de desenvolupament acadèmica orientat a aquest projecte per a nens.

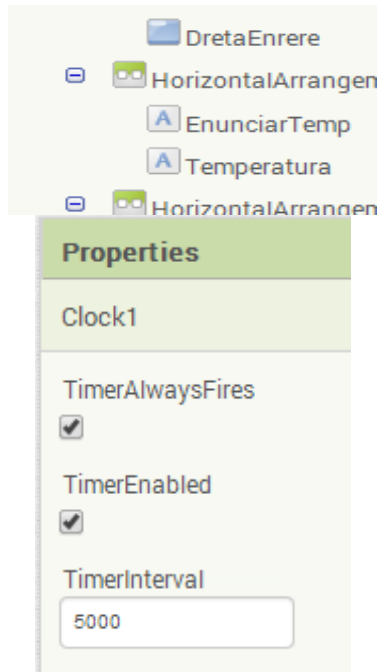
En la part central es pot veure tal com quedarà el disseny. He dissenyat una aplicació que ens permet fer els moviments bàsics mentre es rep la telemetria del cotxe.

Podem veure les diferents parts. Una part purament estètica com és el logotip. Dos fileres amb els botons de posició. I una fila per cada valor de la telemetria obtinguda.

En la part inferior s'ha col·locat el botó que ens permetrà d'iniciar la comunicació amb l'Arduino mitjançant Bluetooth.

Just a la dreta tenim un llistat de tots els components que s'han fet servir, és molt útil de canviar el nom de totes les etiquetes, ja que després ens serà molt més fàcil fer-les servir.

En aquest cas tan sols he ficat nom mnemotècnics als elements que es faran servir en la zona de blocs. O sigui els 7 botons (6 de moviment i el de connexió) i els camps de text on anirà a parar el resultat de la telemetria.

Sistemes de comunicació per a Radio Control (RC) amb Arduino.

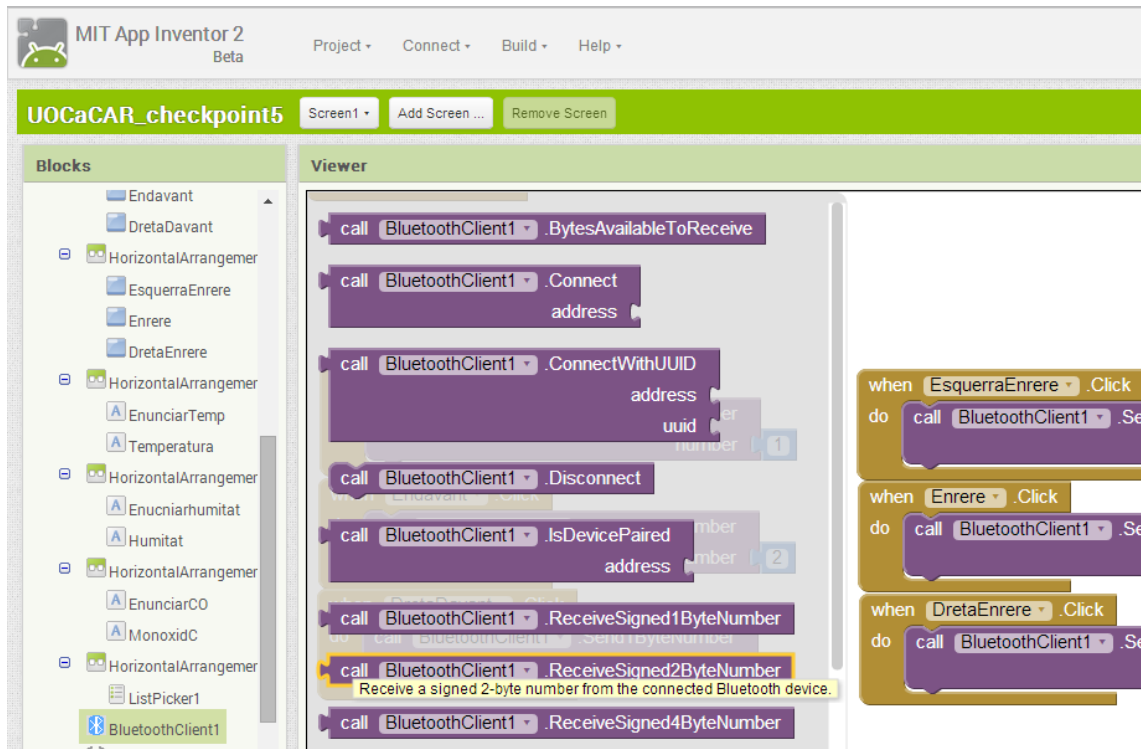
Per acabar amb la part de disseny a la dreta del tot es localitza la part que ens permet de personalitzar l'element que es te seleccionat en aquell moment, Ja sigui canviar la lletra o el contingut.

Per exemple he canviat el botons estàndards del control de moviment i els hi he fica com a fons, sense text, una imatge consistent amb una fletxa del mateix color que el logo manipulada prèviament amb el programa de retoc fotogràfic Gimp.

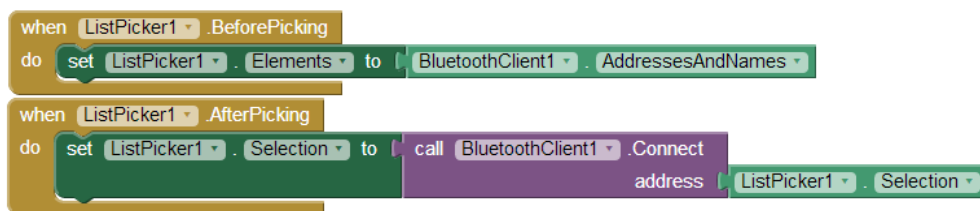
Una altre modificació ha estat la d'un element no visible com és el rellotge (com es pot veure a la foto) el qual té un camp en milisegons que ens indica quan hi haurà una interrupció, en aquest cas definida a 5seg.

Sistemes de comunicació per a Radio Control (RC) amb Arduino.**Mode Blocs.**

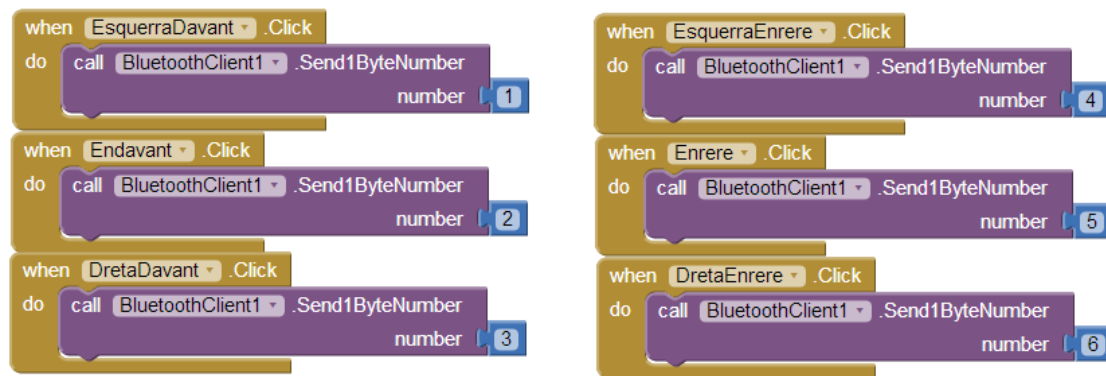
A continuació, mitjançant una sèrie de blocs predefinits, podem assignar funcionalitats als elements que hem ficat en el nostre disseny. L'entorn de treball és bastant intuïtiu, en la part esquerra disposem d'un llistat amb tots els elements prèviament creats, els elements ocults (rellotge, Bluetooth, ...) i una sèrie de variables i procediments bàsics.



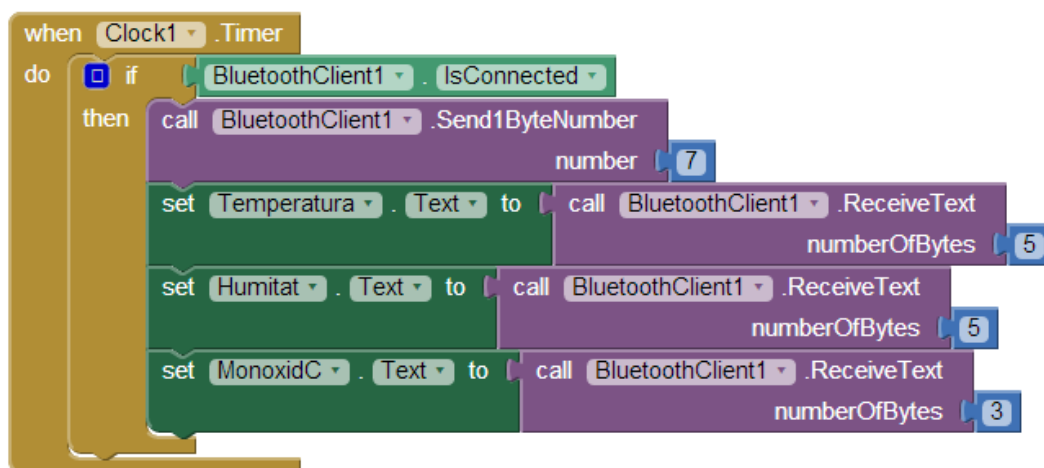
El codi respecta bastant el disseny conceptual. Primerament s'inicialitza el Bluetooth quan apremem el botó. Es pot observar que el codi consisteix dos parts abans i després del "picking", això vol dir que ens sortirà una llista amb totes les connexions Bluetooth disponibles i que se n'ha de triar una, a la qual ens connectarem com a clients.



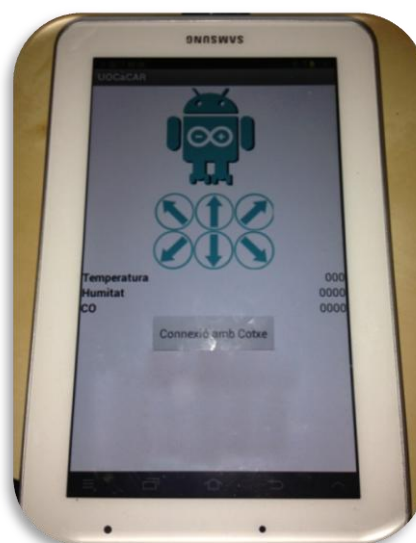
A continuació es configura la acció al pulsar les fletxes de control. Es pot observar que el procés és bastant directe. Quan apremem el botó s'envia mitjançant Bluetooth 1byte amb el valor de la acció a portar a terme.

Sistemes de comunicació per a Radio Control (RC) amb Arduino.

Per acabar, en el disseny hem definit que el rellotge fes un avís cada 5segons, en aquesta part es configura la part de telemetria. En la que si no hi ha cap problema, esperem rebre un text mitjançant Bluetooth d'una mida determinada, definida per un variable numèrica.



El resultat final un cop instal·lat en el dispositiu és de la imatge, i es pot veure un exemple del seu funcionament en el vídeo del annex [final.mp4](#).



Sistemes de comunicació per a Radio Control (RC) amb Arduino.**Pressupost**

ITEM	BOTIGA	UNITATS	PREU U	Divisa	PREU FINAL
Arduino Uno	EBay	1	9,47	\$	7,58 €
Arduino Mega	EBay – delight-digi	1	15,48	\$	12,38 €
Driver Motors	EBay	1	3,50	\$	2,80 €
Cables Test Mascle	EBay – jinbostore	1	1,90	\$	1,52 €
Cables Test Femella 40	EBay – sseariver2009	1	1,65	\$	1,32 €
Mòduls RX/TX 433Mhz	Ebay – gamesalor	1	1,05	\$	0,84 €
Mini Bareboards 9*17	Ebay – simitter01	2	2,02	\$	3,23 €
Bareboard 9*30 + 4	Ebay – heavenstores	1	0,99	\$	0,79 €
Mòdul Piles	Ona radio	1	1,20	€	1,20 €
Resistors	Ona radio	1	1,21	\$	1,21 €
Matriu de polsadors.	EBay-shoppingnewfrog	1	1,08	\$	0,86 €
Jumpers Mascle	Ebay – single2000	2	0,50	\$	0,80 €
Cotxe Teledit	Tot a cent	1	4,50	€	4,50 €
Mòdul Bluetooth	EBay – polida2008	2	7,80	\$	12,48 €
Placa de proves 5x7cm	Ebay – sk2store	1	0,23	\$	0,18 €
Control de temperatura i	Ebay – alice1101983	1	1,43	\$	1,14 €
Sensor de CO.	EBay -sinedya	1	3,65	\$	2,92 €
Shield de Prova Arduino	Diotronic	1	3,00	€	3,00 €
Presupost Material					58,77 €
Inversió Feta					24,23 €
Tablet Samsung Galaxy Tab	MediaMarkt	1	270,00	€	270,00 €
Hores	Estudiant Universitari	271	6,00	€	1.626,00 €
Presupost Total					1.954,77 €

En blau es poden observar les hores i el material que no s'ha tingut que adquirir perquè ja es disposava de projectes previs.

Sistemes de comunicació per a Radio Control (RC) amb Arduino.**Possible comercialització.**

La major part central del treball s'ha plantejat com un estudi pràctic del món del RC des del punt de vista del Arduino, no obstant i això, clarament se'n podria treure'n beneficis de la següent forma.

Kits.

El material fer servir, a excepció del dispositiu Android, tenen un valor tota proper als 60€ per lo que es podria comercialitzar un "kit" per aficionats i estudiants per un preu orientatiu de 100€. De fet hi ha molta varietat d'aquest productes i actualment estan proliferant a la vegada que el món del DIY orientat a la electrònica.

Preu de compra:	60€
Preu de venda:	100€
Benefici per unitat:	40€ (40%)

Sensor mòbil.

La part final més pràctica es tracta de dissenyar un cotxe teledirigit per obtenir telemetria, en aquest cas nivells de CO, temperatura i humitat; encara que solament en la família de sensors MQ podríem convertir el cotxe amb un detector de multitud de coses, i fins i tot sense necessitat de canviar la electrònica ja que tot la família de sensors MQ tenen la mateixa estructura (Vcc, massa, sortida digital).

Aquesta telemetria es podria comercialitzar en entorns hostils on la vida humana pot córrer perill, encara que crec més lògic, degut al seu preu, de presentar-ho com a projecte humanitari per a economies que no es puguin permetre una alternativa comercial.

Preu electrònica:	60€
Preu orientatiu de venda (amb consultoria):	1000€
Beneficis:	940€ (unes 2 setmanes de feina)

Consultoria.

Tot el món del Codi Lliure, té característiques comuns, una d'elles és la de no obtenir benefici directe del treball previ sinó del que s'inverteix per a cada projecte. Com hem vist en el càlcul de costos la majoria són deguts al càlcul d'hores, i això que s'han plantejat a 6€, un del benefici més lògic seria la de facturar hores per ajudar a muntar o adaptar el sensor mòbil o de cobrar per fer classes pràctiques de funcionament d'Arduino juntament amb els kits.

	6€ l'hora	~20€ l'hora
Classe (20hores / alumne):	120€ per un 1 alumne	360€ per cursos de mínim 3alumnes
Consultoria Sensor, 80hores:	480€ preu becari	1600€ preu mínim consultoria

Conclusions.

Vist en profunditat tot el Open HW dona una sèrie d'opcions interessants. El projecte Arduino prové d'una sèrie d'intents previs per apropar el món de la electrònica i la programació a persones sense una formació tècnica prèvia. Això permet que tothom tingui unes eines bàsiques i molt intuïtives per desenvolupar els seus projectes. Aquestes eines poden variar des del HW, més divers i assequible del que es pot pensar; a gran quantitat d'entorns de treball per a desenvolupar i gestionar aquests projectes. La premissa d'aquest treball era el estudi del món del RC aplicat des de la perspectiva del món Arduino i l'Arduino és un entorn de treball ideal per investigar diferents opcions i tecnologies, i ideal per fer prototips.

Hem observat com actualment el món del RC tendeix cada cop més a fer servir freqüències de 2'4GHz, o properes, que tot i ser menys eficients des del punt de vista radio, com més freqüència menys longitud d'ona, per tant menys abast i potències més limitades; però a la vegada ens ofereix la possibilitat de fer servir bona part dels protocols lliures propis d'aquesta banda, com el Wireless o el tractat aquí Bluetooth. Aquests protocols ens ofereixen la possibilitat de fer una integració fàcil amb la transmissió de dades i la bidireccionalitat; amb tot el ventall de possibilitats que estan desenvolupant gran multitud de projectes sobretot relacionats amb IOT (Internet of Things). El fet de fer servir aquestes espectre de freqüència també ens dóna accés a opcions més populars de HW per tant més econòmiques.

La facilitat amb que diferents tecnologies es poden integrar, la quantitat infinita d'informació respecte a cada pas, la varietat de materials, alternatives, ... i tot a preus assequibles i fins i tot amb manuals per fer tu mateix part del procés per estalviar costos... personalment estem a les portes d'una revolució on el HW no es important, ni tan sols el coneixement .. tan sols la capacitat de fer quelcom amb ells.

Bibliografia.

Schmidt, Maik (2011)

Arduino. A Quick-Start Guide.

Prgagmatic Programmers, LLC. Dallas, Texas, Estats Units d'Amèrica.

ISBN-10: 1-934356-66-2

ISBN-13: 978-1-93456-66-1

McRoberts, Michael. (2013).

Beginning Arduino.

Technology in Action. Springer Science+Business Media. New York. Estats Units d'Amèrica.

ISBN-13(pbk): 978-1-4302-5016-6

ISBN-13(elec): 978-1-4302-5017-3

Margolis, Michael. (2011)

Arduino Cookbook.

Imprès en Estats Units distribuït per O'Reilly Media, Inc., Sebastopol, Rússia.

ISBN: 978-0-596-80247-9

LSI: 1299267108

Ph. D.Jack Purdum (2012)

Beginning C for Arduino.

Technology in Action, Segona Edició. New York. Estats Units d'Amèrica.

ISBN-13 (pbk): 978-1-4302-4776-0

ISBN-13(elect): 978-1-4302-4777-7

Warren, John-David; Adams, Josh; Molle, Harald (2011)

Arduino Robotics.

Technology in Action. New York. Estats Units d'Amèrica.

ISBN-13(pbk): 978-1-4302-3183-7

ISBN-13(elec): 78-1-4302-3184-4

Karvinen,Tero; Karvinen, Kimmo (2011)

Make: Arduino Bots and Gadgets.

Imprès en Canadà distribuït per O'Reilly Media, Inc., Sebastopol, Rússia.

ISBN: 978-1-449-38971-0

Göransson, Andreas; Guartielles ,David. (2013).

Professional. Android Open Accessory. Programming with Arduino.

John Wiley & Sons. Indianapolis, Indiana, Estats Units d'Amèrica.

ISBN: 78-1-118-45476-3

ISBN(ebk): 78-1-118-45477-0

Gargenta, Marko (2011)

Learning Android. Building

Imprès en Estats Units distribuït per O'Reilly Media, Inc., Sebastopol, Rússia.

ISBN: 978-1-449-39050-1

Sistemes de comunicació per a Radio Control (RC) amb Arduino.

Cuadro de Atribución de Frecuencias. (consultat el 23 de març de 2014)

<http://www.minetur.gob.es/telecomunicaciones/Espectro/CNAF/cuadroAtribuciones2013.pdf>

Arduino Tool. Official Website. (consultat el 24 de març de 2014)

<http://www.arduino.cc/>

Proyecto ROV/UAV Subacuático. (consultat el 29 de març de 2014)

<https://sites.google.com/site/proyectorovsubacuatico/home/sistema-de-comunicaciones/comunicaciones-subacuaticas>

Processing 2. Official Website. (consultat el 30 de març de 2014)

<http://www.processing.org>

Wiring, get on board. Official Website. (consultat el 30 de març de 2014)

<http://wiring.org.co>

Raspberry Pi Official Website. (consultat el 6 d'abril de 2014)

<http://www.raspberrypi.org/>

Arduino Playground. (consultat el 15 d'abril de 2014)

<http://playground.arduino.cc/>

16.- DHT11 Sensor de Temperatura y Humedad . (consultat el 15 d'abril de 2014)

<http://www.practicandoarduino.com/en/author/practicandoarduino/>

Android SD. (consultat el 3 de maig de 2014)

<http://developer.android.com/sdk/index.html>

MIT:APP Inventor. (consultat el 11 de maig de 2014)

<http://appinventor.mit.edu/explore/c>