

Vehicle WiFi

Estudiant: Francesc Pujol Contreras
Enginyeria Tècnica d'informàtica de Sistemes

Consultor: Jordi Bécares Ferrés

15 de gener de 2013

Agraïments:

Aquest projecte ha estat la culminació de un viatge molt enriquidor i m'omple de alegria haver assolit aquesta fita. Aquest camí però no hagués estat possible sense l'inestimable ajunt de tota la gent que ha estat al meu voltant durant aquest temps i al seu recolzament.

Gràcies en primer terme als tutors i consultors que ho han fet possible, especialment a Jordi Bécares per fer de guia en un projecte tant interessant.

Donar també les gràcies a tots els amics, per ser-hi, i sobretot el més profund agraïment a tota la família, per la paciència i comprensió demostrats durant aquest temps, i en especial, al meu estimat fill **Oriol**, que segur té més ganes que jo per que es conclougui aquesta empresa.

Moltes gràcies a tothom !

RESUM

El present projecte, dins l'àrea dels sistemes encastats té com a objectiu la construcció de un **vehicle a motor elèctric** amb connectivitat per xarxes sense fils **WiFi**, de forma que pugui ser controlat a distància. Amb aquesta finalitat ha estat desenvolupada una **aplicació per taules iPad de control remot del vehicle**.

Al cor del disseny del vehicle es troba un sistema microcontrolador LPC1769 basat en un processador ARM-Cortex M3, que amb una freqüència de rellotge de 120Mhz ens ofereix una capacitat de processament notable per un cost reduït, que destaca per oferir un elevat nombre de ports de connexió i comunicació.

En forma de perifèrics connectats al sistema microcontrolador, el projecte inclou el mòdul WiFi RN-XV de comunicacions sense fils per xarxes WiFi, que destaca per les serves reduïdes dimensions i un potent sistema de gestió. Per el control del desplaçament i la direcció, a banda del xassís amb motors de corrent continu, el sistema inclou un mòdul independent controlador de motors. El disseny el completa un compacte sistema sonar per la detecció d'obstacles.

El dispositiu s'ha programat amb el suport del sistema operatiu FreeRTOS, de forma que s'ha aconseguit un disseny modular, gràcies a la construcció de controladors per cada un dels dispositius utilitzats.

En base a aquests recursos, el vehicle desenvolupat, permet ser controlat remotament amb qualsevol equip de xarxa que disposi de un terminal **Telnet**. Per oferir una experiència de control remot més interessant, el sistema pot ser controlar amb una senzilla aplicació per taules tàctils.

Índex de continguts

| | |
|--|----|
| 1. Introducció..... | 6 |
| 1.1. Justificació..... | 6 |
| 1.2. Descripció del projecte..... | 7 |
| 1.3. Objectius del TFC..... | 7 |
| 1.3.1. Modelar un sistema microcontrolador per controlar un vehicle..... | 8 |
| 1.3.2. Dotar el sistema de connectivitat sense fils..... | 9 |
| 1.3.3. Disposar d'un sistema de depuració (log/debug)..... | 9 |
| 1.3.4. Implementar un vehicle amb sistema de control per motors..... | 9 |
| 1.3.5. Dirigir remotament el vehicle..... | 9 |
| 1.3.6. Oferir un sistema de control automàtic..... | 9 |
| 1.3.7. Control de direcció..... | 10 |
| 1.3.8. indicació visual del nivell de senyal de xarxa..... | 10 |
| 1.3.9. Detecció obstacles per mitjà de sensor de ultrasons..... | 10 |
| 1.3.10. Sistema de parada automàtica per obstacles..... | 10 |
| 1.3.11. Aplicació de control remot del vehicle amb tablet..... | 10 |
| 1.4. Enfocament i mètode seguit..... | 11 |
| 1.5. Planificació del projecte..... | 12 |
| 1.6. Recursos emprats..... | 15 |
| 1.6.1. Maquinari..... | 15 |
| 1.6.2. Programari..... | 17 |
| 1.7. Productes obtinguts..... | 19 |
| 1.8. Breu descripció dels altres capítols de la memòria..... | 20 |
| 2. Antecedents..... | 21 |
| 2.1. Estat de l'art..... | 21 |
| 2.2. Estudi de mercat..... | 27 |
| 3. Descripció funcional..... | 28 |
| 3.1. Sistema total..... | 28 |
| 3.1.1. Diagrama de blocs..... | 29 |
| 3.1.2. Com és la xarxa..... | 29 |
| 3.1.3. Com interactuen els diferents objectes en el sistema..... | 30 |
| Disseny de l'aplicació al sistema encastat..... | 31 |
| 3.3. Disseny de la interfície d'usuari..... | 34 |
| 4. Descripció detallada del sistema..... | 39 |
| 4.1. Disseny del vehicle..... | 39 |
| 4.1.1. El sistema LPC1769: disseny de controladors..... | 39 |
| 4.1.2. Disseny del prototip..... | 47 |
| 4.2. Aplicació de control remot..... | 49 |
| 5. Viabilitat tècnica..... | 51 |
| 6. Valoració econòmica..... | 52 |
| 7. Conclusions..... | 53 |
| 7.1. Conclusions..... | 53 |
| 7.2. Proposta de millores..... | 54 |
| 7.3. Autoavaluació..... | 54 |
| 8. Glossari..... | 55 |
| 9. Bibliografia..... | 57 |
| 10. Annexos..... | 58 |
| 10.1. Ports de connexió del LPC1769..... | 58 |

Índex de figures

| | |
|---|----|
| Figura 1: Escenari de treball..... | 7 |
| Figura 2: Planificació inicial..... | 13 |
| Figura 3: Planificació final..... | 14 |
| Figura 4: Vehicle electrònic | 19 |
| Figura 5: Aplicació per tauleta..... | 20 |
| Figura 6: L'Internet de les coses..... | 22 |
| Figura 7: Esquema del sistema Arduino..... | 23 |
| Figura 8: Esquema del sistema Raspberry Pi..... | 24 |
| Figura 9: Conjunt i plaques components del LPC1769..... | 25 |
| Figura 10: Diagrama de blocs, nodes servidor i client..... | 29 |
| Figura 11: Estructura de la placa LPC1769..... | 32 |
| Figura 12: Interconnexió dels dispositius..... | 33 |
| Figura 13: Inici de la sessió Telnet..... | 34 |
| Figura 14: Aplicació de control remot per tauleta tàctil..... | 35 |
| Figura 15: Client de port sèrie per depuració | 38 |
| Figura 16: Xassís previst inicialment..... | 41 |
| Figura 17: Xassís del vehicle fet servir finalment..... | 41 |
| Figura 18: Diagrama de blocs amb els controladors de dispositius..... | 43 |
| Figura 19: Diagrama de flux, sistema principal del vehicle electrònic amb microcontrolador..... | 44 |
| Figura 20: Interacció entre tasques i llibreries..... | 45 |
| Figura 21: Relació de comandes de control per el vehicle..... | 46 |
| Figura 22: Protoboard amb connexió dels diferents mòduls..... | 47 |
| Figura 23: Regulador de tensió..... | 47 |
| Figura 24: Placa final amb connectors i components..... | 48 |
| Figura 25: Configuració en Pont H..... | 48 |
| Figura 26: Diagrama UML, estructura de classes..... | 49 |
| Figura 27: Funció de transformació a la velocitat..... | 50 |

1. Introducció.

Els sistemes robòtics són molt presents a les nostres vides. A ningú li resulta estrany trobar a les llars robots de neteja i cada cop resulta més freqüent escoltar les innovacions que es van produint al respecte. De totes formes encara es pot considerar que és una tecnologia molt incipient, al menys respecte al àmbit domèstic, doncs al món industrial la robòtica ja fa molts anys que es troba consolidada encara que continua experimentant un fort creixement, per exemple, recentment l'empresa Foxconn¹ comunicava un pla per implantar un milió de robots a les seves plantes en tres anys.

Donat que els ordinadors de butxaca també s'han tornat realitat en pocs anys, mirem per exemple els smartphones, resulta molt interessant poder aprofundir en aquests sistemes, basats en un sol xip i que ofereixen unes prestacions sorprenents.

1.1. Justificació.

Si fem una mica de recerca a través d'Internet, veurem de seguida que existeix una gran quantitat de projectes i productes de robòtica en forma de vehicle. Com a tret diferenciador d'aquest projecte podem destacar el potencial que ofereixen tant els dispositius fets servir com les eines amb les que ens recolzem. A banda de poder dur a la pràctica coneixements adquirits durant els presents estudis d'enginyeria, aquest projecte obre una àmplia finestra per poder continuar explorant una arquitectura sobre el que s'intueix un elevat nombre de possibilitats.

L'arquitectura del sistema adoptat per aquest projecte, potser no és tant popular com altres productes, com Arduino o Raspberry Pi, però les seves característiques de potència i connectivitat el posicionen clarament com una bona opció a l'hora d'encarar un projecte d'aquest àmbit. L'entorn de desenvolupament ens permet programar amb llenguatge C, de forma que podrem construir aplicacions ben estructurades i eficients.

El fet de implementar les comunicacions sense fils amb l'estàndard WiFi ens permetrà igualment aconseguir un dispositiu flexible capaç de interaccionar amb un nombre molt elevat de dispositius.

¹ Foxconn http://news.xinhuanet.com/english2010/china/2011-07/30/c_131018764.htm

1.2. Descripció del projecte.

El projecte essencialment consisteix en afegir uns petits motors i rodes al sistema microcontrolador, per aconseguir que es pugui desplaçar. El sistema ha de ser capaç de oferir connectivitat per Internet, enviant i rebent dades a través del mòdul de comunicacions sense fils WiFly RN-XV. Amb aquest mètode de comunicació el sistema té que ser capaç de moure's per anar endavant i enredera. Un segon requeriment és que el vehicle disposi de un sistema de control automàtic, amb el qual buscarà el punt on té millor cobertura WiFi, cercant la posició on es troba el punt d'accés sense fils (Acces Point).

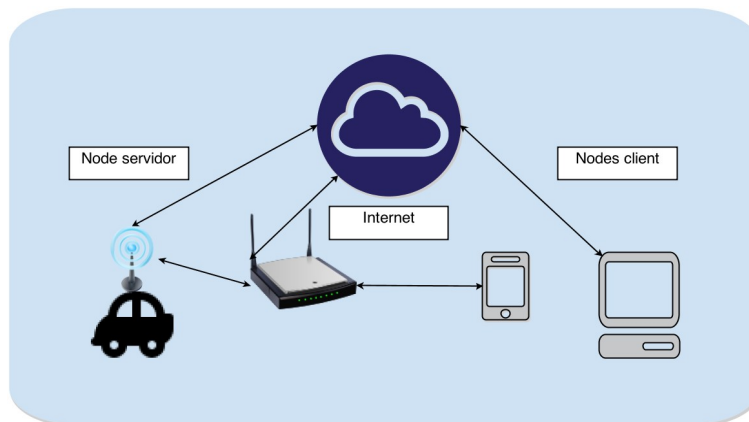


Figura 1: Escenari de treball

1.3. Objectius del TFC.

Els objectius que es persegueixen al present projecte són la **construcció de un vehicle des de la perspectiva de la robòtica: en base a un ordinador**, que permeti experimentar amb les diferents tecnologies disponibles als sistemes actuals, i així posar en pràctica els coneixements adquirits durant els presents estudis. Aquests objectius s'ha estructurat en dos grups:

Objectius principals:

- Modelar un sistema microcontrolador per controlar un vehicle.
- Dotar el sistema de connectivitat sense fils.
- Disposar d'un sistema de depuració (log/debug).
- Implementar un vehicle amb sistema de control per motors.
- Controlar la tracció amb motors.
- Dirigir el vehicle, a partir dels sistemes de connectivitat sense fils i de control de motors.
- Oferir un sistema de control automàtic del vehicle.
- Controlar automàticament el moviment de tracció del vehicle.

Objectius secundaris :

- Poder controlar la direcció del vehicle amb motors.
- Oferir una indicació visual del senyal de xarxa.
- Dotar el sistema de detecció obstacles per mitjà de sensor de ultrasons.
- Incorporar un sistema de parada automàtica al detectar obstacles propers.
- Aplicació de control remot del vehicle amb Tablet:
 - Connexió WiFi
 - Control de velocitat i direcció
 - Recepció de la distància dels obstacles

1.3.1. Modelar un sistema microcontrolador per controlar un vehicle.

Es durà a terme la programació del sistema encastat per controlar el conjunt de dispositius que conformaran el sistema complet. L'execució i depuració del codi es faran directament sobre el dispositiu connectat al ordinador de desenvolupament. Un cop completat tot el sistema, aquesta programació quedarà enregistrada al controlador i ens ha de permetre fer-ho servir de forma totalment autònoma.

1.3.2. Dotar el sistema de connectivitat sense fils.

Es farà servir un mòdul de comunicacions sense fils totalment independent del sistema microcontrolador. Aquest sistema de comunicacions és el que ens ha de permetre poder controlar el vehicle amb dispositius remots.

1.3.3. Disposar d'un sistema de depuració (log/debug).

Per desenvolupar el sistema en primera instància, i disposar posteriorment de un servei de diagnostic, el vehicle comptarà amb una connexió específica amb la que poder visualitzar el seu estat i poder realitzar les comprovacions pertinents.

1.3.4. Implementar un vehicle amb sistema de control per motors.

Com es construeix un vehicle a motor elèctric, s'incorporarà un sistema de control de motors, específic per aquesta tasca, independent del microcontrolador del sistema. Amb aquest sistema controlarem la tracció del vehicle.

1.3.5. Dirigir remotament el vehicle.

Un cop completats i validats els punts anteriors, es desenvoluparà el sistema de control, que té per missió permetre al vehicle rebre i interpretar les diferents ordres de control, procedents del dispositiu extern de control.

1.3.6. Oferir un sistema de control automàtic.

De forma alternativa al mecanisme de control remot, el sistema disposarà de un mode de funcionament automàtic segons el que, aquest, s'anirà desplaçant per cercar la zona on el nivell de recepció del senyal de xarxa sense fils resulti òptim.

Per fer-ho es controlarà automàticament el moviment de tracció del vehicle amb el sistema de control de motors.

1.3.7. Control de direcció.

Un com superats els objectius principals, incorporarem en el sistema de control del vehicle la direcció, mitjançant el mateix mòdul de control de motors per la tracció.

1.3.8. indicació visual del nivell de senyal de xarxa.

Un altre dels objectius secundaris, correspondrà a la visualització del nivell de senyal de la xarxa sense fils, l'anomenat RSSI, que es mostrarà segons el nivell de lluminositat de un díode LED, actualitzant-se de forma regular.

1.3.9. Detecció obstacles per mitjà de sensor de ultrasons.

També en aquest segon grup d'objectius, es dotarà al sistema de un sensor de obstacles per ultrasons, amb el que podrem detectar tant la presència com la distància dels possibles obstacles a la zona frontal del vehicle.

1.3.10. Sistema de parada automàtica per obstacles

Amb la informació recollida per el sensor del punt anterior, es programarà el sistema de detecció d'obstacles que farà aturar automàticament el vehicle si es detecta un obstacle a un mínim de distància.

1.3.11. Aplicació de control remot del vehicle amb tablet.

Finalment, per completar el projecte es confeccionarà una interfície gràfica per mitja d'una aplicació de control remot per tauleta. Aquesta aplicació establirà una connexió per xarxa sense fils amb el vehicle i mostrarà constantment la distància a la que es trobin els possibles obstacles. Disposarà també de un control amb el que es podrà controlar la velocitat i el sentit del moviment del vehicle, així com la direcció.

1.4. Enfocament i mètode seguit.

La metodologia aplicada per desenvolupar el projecte ha consistit en una primera etapa d'exploració dels diferents elements i eines a utilitzar en el desenvolupament del projecte. En un projecte nou, cal un procés previ d'aprenentatge sobre les eines i productes a fer servir, que ha de permetre fer-ne un us adequat i poder conèixer les possibilitats. Això ens ha ajudat poder valorar de forma més adequada la viabilitat dels posteriors objectius a assolir.

Un cop familiaritzats amb el sistema microcontrolador, els dispositius i l'entorn de desenvolupament, s'ha procedit a dissenyar el programari de controlador per els dispositius utilitzats, amb l'objectiu de oferir un sistema modular, on els diferents components puguin interactuar de forma clara i previsible, amb interfícies ben definides. El desenvolupament s'ha fet seguint el model d'enginyeria iteratiu i creixent, de forma que a cada entrega s'han anat identificant les possibles millores i correccions. Aquests cicles tancats i incrementals han permès anar madurant els mòduls, no s'ha començat a construir la aplicació final fins que no s'ha consolidat el desenvolupament de cada mòdul i testejat les seves funcionalitats. Això ha inclòs casos més generals encara que poguessin quedar fora dels requisits del projecte final, com ara la interacció amb servidors a Internet, però també ha permès anar aprofundint en el coneixement dels recursos fets servir, com ara el sistema operatiu per sistemes encastats .

El següents passos han estat formalitzar la proposta del projecte i definir el pla de treball, amb els que s'han concretat l'abast del projecte i la seva temporització, detallant les diferents tasques a dur a terme dins el projecte per poder assolir els objectius. Durant el procés de elaboració del projecte, de manera paral·lela, s'ha anat recollint informació i desenvolupant esquemes respecte al seu funcionament per poder enriquir posteriorment la documentació generada.

1.5. Planificació del projecte.

Per dur a terme aquest projecte hem disposat d'un període de temps de 4 mesos aproximadament. Els primers 2 mesos han estat dedicats a realitzar la exploració i el aprenentatge dels diferents productes i eines. Aquest procés ha estat pausat amb un seguit de proves de avaluació continua tal i com figura en la següent relació de tasques, on tot seguit es detalla la previsió de tasques per realitzar el projecte, objectiu d'aquesta signatura:

Tasques

| ID | NOM | Inici | Fi | Durada | Fet |
|-------|--|--------|--------|--------|------|
| 1 | PAC1 | Sep 19 | Oct 9 | 15d | 100% |
| 2 | PAC2 | Oct 10 | Nov 6 | 20d | 100% |
| 3 | PAC3 | Nov 7 | Nov 23 | 13d | 100% |
| 4 | PAC4 | Nov 19 | Nov 30 | 10d | 100% |
| 5 | PAC5 | Nov 28 | Jan 2 | 26d | 40% |
| 6 | Control de motors | Dec 3 | Dec 12 | 8d | 0% |
| 6.1 | Via app Windows | | | 1d | 0% |
| 6.2 | Val LPC1769 amb codi Hardcoded | | | 3d | 100% |
| 6.3 | Via UART-USB-CP2102 cap a LPC1769 | | | 2d | 100% |
| 6.4 | Rebre comandes via UART-WIFLY | | | 2d | 50% |
| 7 | Alimentació del circuit | Dec 6 | Dec 7 | 2d | 0% |
| 7.1 | Alimentació i placa final | | | 1d | 0% |
| 7.2 | Connectors i soldadures | | | 1d | 0% |
| 8 | Sistema amb control autònom | Dec 7 | Dec 13 | 5d | 0% |
| 8.1 | Recepció de senyal WiFi | | | 1d | 100% |
| 8.2 | Algorisme de moviment en un espai de 2 dimensions | | | 3d | 0% |
| 8.4 | Enviar les ordres cap al control de motors | | | 1d | 0% |
| 8.5 | Mecanisme per evitar col·lisions amb obstacles | | | 1d | 0% |
| 9 | Control de la direcció del vehicle | Dec 14 | Dec 28 | 10d | 0% |
| 9.1 | Control de direcció al microcontrolador LPC1769 | | | | |
| 9.1.1 | Crear connexió: funció WiFlyTelnetServer | | | 1d | 50% |
| 9.1.2 | Crear la funció de control amb les comandes | | | 2d | 50% |
| 9.1.3 | Enviar les ordres cap al control de motors | | | 2d | 0% |
| 9.1.4 | Enviar missatges de status | | | 1d | 0% |
| 9.1.5 | Mecanisme per evitar col·lisions amb obstacles | | | 2d | 0% |
| 9.2 | Control de direcció al ordinador | | | | |
| 9.2.1 | Crear connexió: Telnet amb terminal de Linux | | | 1d | 100% |
| 9.2.2 | Crear connexió: Aplicació client de Telnet | | | 2d | 80% |
| 9.2.3 | Enviar les comandes i rebre les confirmacions | | | 2d | 20% |
| 9.2.4 | Rebre els missatges de status | | | 1d | 0% |
| 10 | Detector per ultrasons de obstacles | Dec 21 | Dec 28 | 6d | 0% |
| 10.1 | Crear driver lectura ADC | | | 1d | 0% |
| 10.2 | Connectar mòdul, test, calcular distància | | | 1d | 0% |
| 10.3 | Enviar missatges de status | | | 1d | 0% |
| 11 | Aplicació de control per mòbil | Dec 18 | Jan 2 | 12d | 0% |
| 11.1 | Crear connexió Telenet | | | 3 | 0% |
| 11.2 | Crear comandes | | | 3 | 0% |
| 11.3 | Rebre status | | | 3 | 0% |
| 11.4 | Crear interfície usuari | | | 3 | 0% |
| 12 | PAC6 | Dec 20 | Jan 15 | 19d | 0% |
| 13 | PAC7 | Jan 16 | Jan 21 | 4d | 0% |

Planificació inicial, diagrama de Gantt:

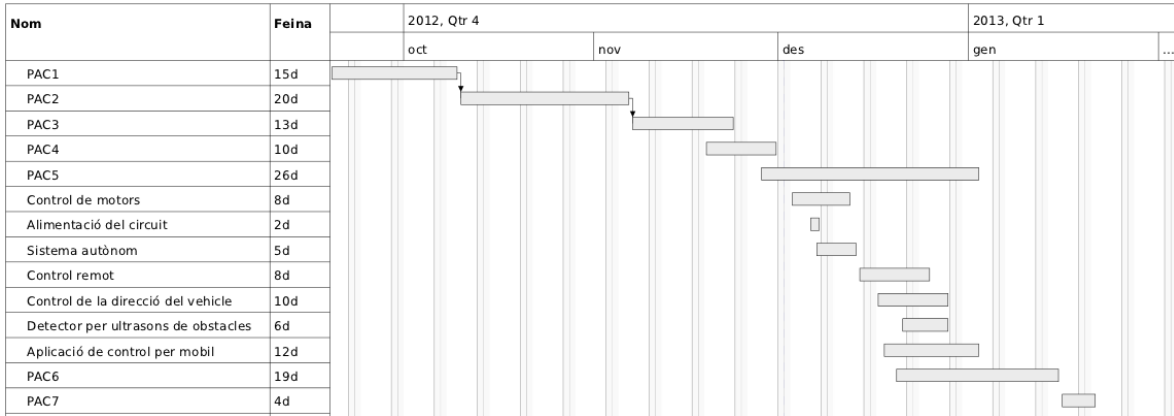


Figura 2: Planificació inicial

Un cop realitzades les tasques definides a la planificació inicial, es pot observar al següent diagrama de Gantt, que la temporització planificada no ha patit cap desviació important respecte a la planificació inicial a excepció del disseny i muntatge de la placa final amb circuit imprès, que ha requerit més feina de la prevista en un bon començament. Com es pot observar en els dos diagrames, s'ha mantingut un cert grau de paral·lelisme entre les diferents tasques realitzades. En una situació amb recursos humans suficients, això ens permetria optimitzar els recursos disponibles. En aquest cas en concret, ens ha ajudat a no introduir endarreriments a tasques crítiques com per exemple les corresponents al sistema de control remot.

Planificació final, diagrama de Gantt;

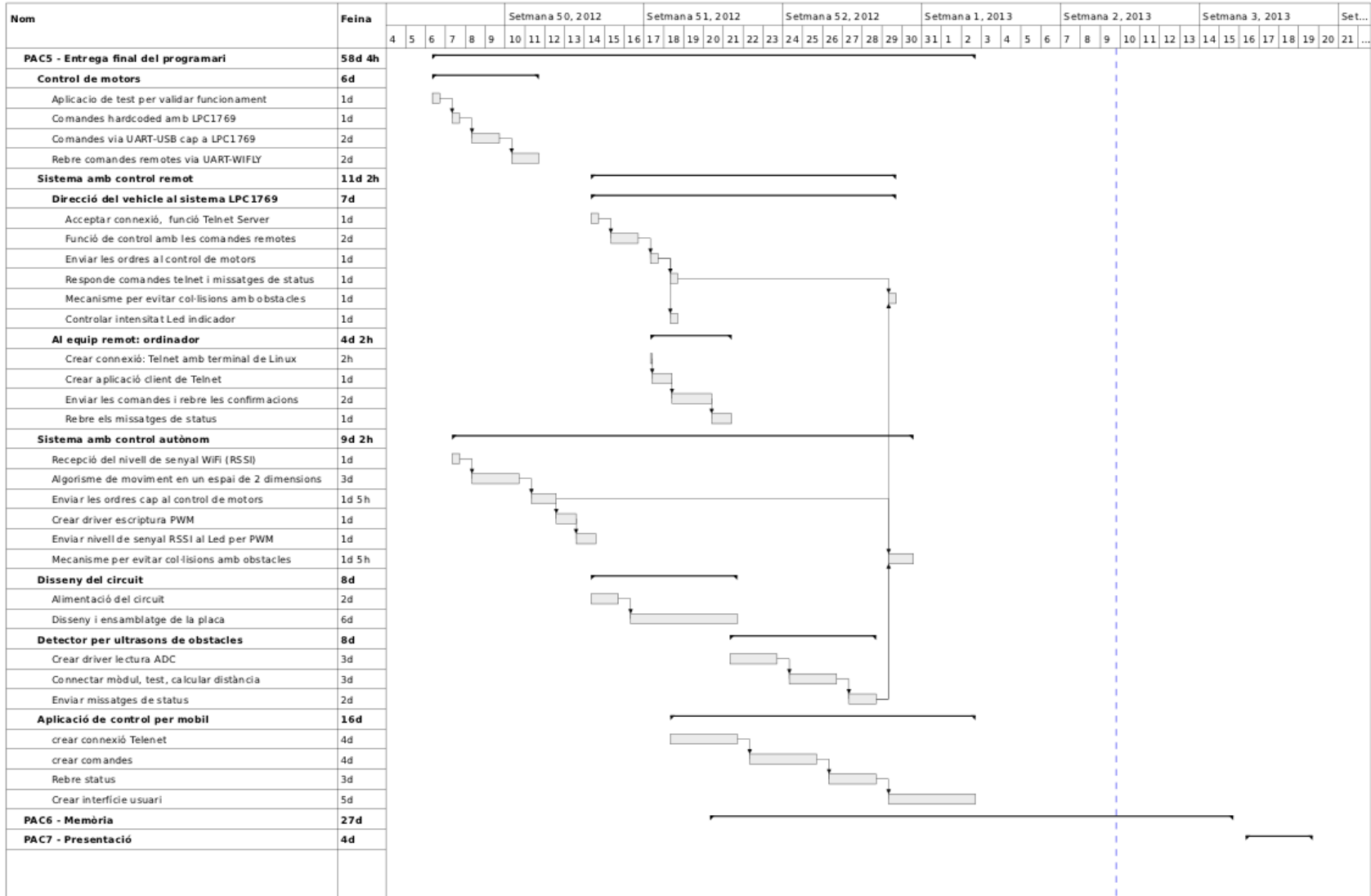


Figura 3: Planificació final

1.6. Recursos emprats.

1.6.1. Maquinari.

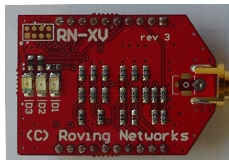
- **Sistema microcontrolador PC1769:**

Es tracta de una placa dissenyada com a sistema de avaluació per els circuits integrats del fabricant NXP. Aquesta placa pertany a una família de dispositius anomenada LPCXpresso que, desenvolupada conjuntament amb les empreses Code Red Technologies i Embedded Artists, s'han concebut per oferir als enginyers una serie de circuits impresos amb les que avaluar i poder realitzar dissenys basats en els seus microcontroladors. Aquests es basen en l'arquitectura de microprocessadors RISC de 32 bits ARM.

Les plaques de la família LPCXpresso es caracteritzen per poder ser dividides en dos unitats independents, una anomenada LPC-LINK creada per oferir connectivitat amb al sistema de desenvolupament i l'altre on es troba el propi microcontrolador per el disseny final.

- **Sistema de xarxa sense fils WiFly RN-XV**

Per disposar de connectivitat a xarxes sense fils WiFi farem servir l'adaptador WiFly RN-XV



del fabricant Roving Networks. Aquest mòdul té una distribució de pins que permet simplificar la migració de dispositius XBee amb arquitectura 802.15.4 cap a una plataforma amb connectivitat TCP/IP. Per aquest motiu, al nostre projecte s'ha

fet servir una placa adaptadora que ens proporciona una distribució física dels pins a 2,54mm que és la que fa servir amb els diferents connectors i plaques fetes servir al projecte.

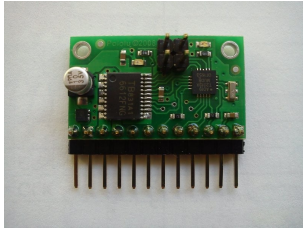
- **Adaptador USB-sèrie CP2102**

El mòdul CP210x USB a UART és un VCP (Virtual COM Port). Per fer servir aquest dispositiu en entorns Windows, caldrà un controlador o driver que permeti fer-lo servir com a port Com Virtual. En cas de fer servir GNU/Linux probablement no ens caldrà instal·lar

cap controlador addicional, dons es tracta d'un maquinari que està ben suportat per el sistema, que ja inclou el controlador.

- **Controlador de motors DC Qik 2s9v1**

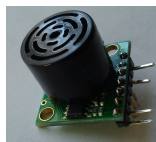
Aquest controlador està dissenyat per poder controlar 2 motors petits DC (de corrent continu). El mòdul permet que qualsevol



microcontrolador o ordinador amb un port sèrie (en el cas de ports RS-232 caldria un conversor de nivells de tensió) pugin controlar tant la direcció com la velocitat dels motors. La part electrònica de control pot ser alimentada amb tensions a partir de 2,7 de forma que és

compatible amb els microcontroladors que funcionen a 3.3V. La comunicació sèrie és bidireccional, i disposa de un sistema de autodetecció de la velocitat que pot arribar fins al 38400 bauds. Disposa també de un sistema opcional de detecció d'errors basat en el conegut sistema CRC (Cyclic redundancy check). Disposa de dos Leds de estat amb els que proporcion informació tant de la comunicació sèrie com del possible estat d'error del dispositiu. També incorpora dos jumpers per la configuració, que permet per exemple establir una velocitat fixa de comunicació sèrie.

- **LV-MaxSonar EZ3**



Dedicat a la detecció de obstacles, aquest reduït dispositiu permet identificar objectes a distàncies d'entre 0 i 6.45 metres amb una resolució de 2,54cm (1 polzada), proporcionant informació de la distància per objectes a partir de un punt mínim de 15,24cm.

Respecte a la interconnexió el dispositiu es pot comunicar tant per port sèrie com per sortida de voltatge analògica o amb polsos PWM.

- **Tauleta tàctil**

Es tracta d'un tipus de terminal portàtil, evolució de les PDA, que s'ha anat popularitzant als darrers temps després del boom dels ordinadors portàtils i la telefonia mòbil. Pràcticament la totalitat de la interacció es realitza amb una pantalla tàctil que pot estar desenvolupada

amb tecnologia capacitiva o resistiva. Aquestes pantalles es caracteritzen per oferir en general més de un punt de contacte simultani (multi-touch) amb el que es reconeixen els diferents gestos realitzats amb els dits. Normalment no compten amb teclat físic, i per l'entrada de dades en presenten un de virtual a la pantalla, encara que també solen admetre teclats externs sense fils. En aquest projecte es farà servir concretament la tauleta iPad d'Apple.

1.6.2. Programari.

- **LPCXpresso V4.3.0 basat en Eclipse V 3.6.2**

Es una plataforma de desenvolupament de NXP. Està dissenyada per microcontroladors LPC basats en l'arquitectura ARM. S'ha construït en base a una versió adaptada del IDE Eclipse IDE, amb la que desenvolupar sobre les plaques de baix cost d'aquest fabricant.

Es una eina que permet escriure el codi de l'aplicació, compilar i executar sobre la placa objectiu connectada al port USB, així com realitzar la seva depuració, amb control de l'execució, breakpoints (punts de aturada) i consulta de valors en temps d'execució.

- **FreeRTOS**

Es tracta d'un sistema operatiu en temps real per sistemes encastats, distribuït sota llicència GPL encara que inclou una excepció per permetre que els usuaris puguin mantenir tancat el codi propi, per permetre aplicacions propietàries. Està dissenyat per ser simple i petit. El nucli està format per uns pocs arxius en C. Per fer el codi llegible, fàcil de portar i fàcil de mantenir, està escrit bàsicament en C, però inclou algunes funcions en llenguatge ensamblador on es necessari (rutines específiques per cada arquitectura).

Proporciona mètodes per tasques o subprocessos, control de concurrència per exclusió mútua, cues, semàfors i temporitzadors. L'èmfasi el posa en oferir un sistema compacte amb una alta velocitat d'execució.

- **Aplicació de comunicacions serie CuteCom**

CuteCom és un terminal gràfic per comunicacions sèrie, com ara minicom o HyperTerminal de Windows encara que aquest s'executa en Linux, FreeBSD i Mac OS X.

Està dirigit principalment als desenvolupadors de maquinari o d'altres persones que necessiten un terminal per comunicar-se amb aquest tipus de dispositius. És programari lliure i es distribueix sota la llicència de codi lliure GPL 2.

- **MonoTouch for iOS**



Es tracta de la versió comercial per iOS del entorn de desenvolupament .NET. Permet als desenvolupadors crear aplicacions basades .NET amb el llenguatge C#, que s'executen en els dispositius iPad, iPhone i iPod d'Apple. Per una banda aprofiten la API d'iOS i per un altre canto permeten la reutilització de codi i biblioteques que s'han construït per .NET. El producte inclou l'IDE de desenvolupament MonoDevelop així com integració amb Xcode, l'editor de Apple.

A diferència de les aplicacions Mono habituals a d'altres entorns com GNU/Linux, MacOS o Windows, les aplicacions amb MonoTouch es compilen a codi màquina. Això és degut a que el nucli de iOS impedeix que facin servir compiladors JIT (Just-In-Time) al executar aplicacions sobre el dispositiu.

- **Distribució Ubuntu 11.04 del sistema operatiu GNU/ Linux**



Com a entorn de desenvolupament del projecte, s'utilitzarà Ubuntu, una popular distribució del sistema operatiu GNU/Linux basada en Debian. Aquest sistema operatiu d'escriptori està desenvolupat per l'empresa Canonical Ltd, que ofereix serveis de suport o infraestructura al núvol (cloud). En destaca un programa de actualitzacions de versions cada 6 mesos però que també ofereix versions més estables a llarg plaç cada 2 anys. La principal llicència que fa servir es la GNU GPL (GNU General Public License) encara que hi ha excepcions como ara drivers propietaris. Als darrers temps s'ha anat incorporant però més programari propietari accessible directament amb el seu gestor d'aplicacions.

1.7. Productes obtinguts.

Amb l'elaboració del present projecte s'ha desenvolupat un vehicle a motor elèctric que basa el seu funcionament en un sistema microcontrolador, disposa de connectivitat amb xarxes sense fils WiFi i incorpora un sistema de detecció d'obstacles per ultrasons. Gràcies a aquests elements el vehicle permet una comunicació simple i eficaç amb dispositius que disposin dels protocols estàndard de Internet, amb una simple sessió de Telnet.

- **Vehicle electrònic**

Potser l'aspecte més notable d'aquest producte és la flexibilitat. El cor del sistema, el microcontrolador LPC1769 ens presenta un sistema d'alt rendiment i amb elevat nivell de connectivitat, aspecte que s'ha aprofitat en el disseny del vehicle, fent servir diferents ports serie per integrar la majoria de mòduls. A més a més, al integrar l'IDE LPCXpresso, un entorn de desenvolupament que permet executar i depurar la programació sobre el dispositiu de forma ràpida i fàcil, podem afirmar que ens trobem davant d'un sistema sobre el que realitzar ampliacions o adaptacions, resulta poc complicat.

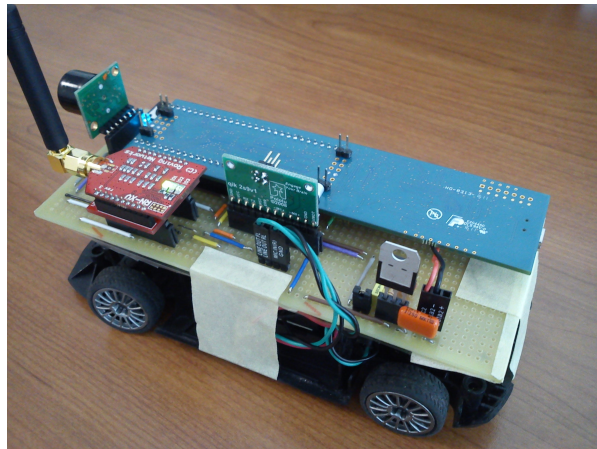


Figura 4: Vehicle electrònic

- **Aplicació per tauleta tàctil de control remot**

Amb la mateixa visió de simplicitat, s'ha creat una app de control remot per el vehicle anomenada WiFiPad, orientada a dispositius tàctils. Per el present projecte s'ha implementat sobre una tauleta tàctil iPad de Apple encara que amb l'entorn de desenvolupament adoptat, es pot executar sobre altres taules com les que fan servir el sistema Android, o ordinadors amb Linux, Windows o MacOs. Ens troben doncs, al igual que amb el vehicle, una solució flexible que és fàcilment adaptable i ampliable.



Figura 5: Aplicació per tauleta

1.8. Breu descripció dels altres capítols de la memòria.

Fins ara s'ha fet una introducció als diferents aspectes del projecte, justificant les decisions preses, planificant l'elaboració o detallant els productes fets servir i els resultats obtinguts. Als propers capítols situarem, el producte dissenyat, dins el context de les tecnologies disponibles actualment i tot seguit, aprofundirem en el disseny dels sistemes desenvolupats. Posteriorment farem una valoració tècnica i econòmica per estudiar la seva viabilitat. Per acabar dedicarem els darrers capítols a extreure un seguit de conclusions i a aportar informació de referència.

2. Antecedents.

Encara que els microcontroladors i els microprocessadors tenen un grau de presència respecte al temps semblants, no podem afirmar que gaudeixin del mateix grau de difusió, donada la característica específica dels sistemes dedicats on es fan servir el microcontroladors envers al plantejament de propòsit general que tenen el processadors.

Cal aclarir però, que de fet estem parlant de dispositius molt semblants, en molts casos trobem microcontroladors que tenen el mateix disseny que el microprocessador corresponent, com per exemple, el microprocessador Motorola 6800 var anar seguit dels microcontroladors 6801 i 6803 incorporant aquests darrers, aspectes característics dels sistemes encastats com ara memòries RAM i ROM o un port sèrie.

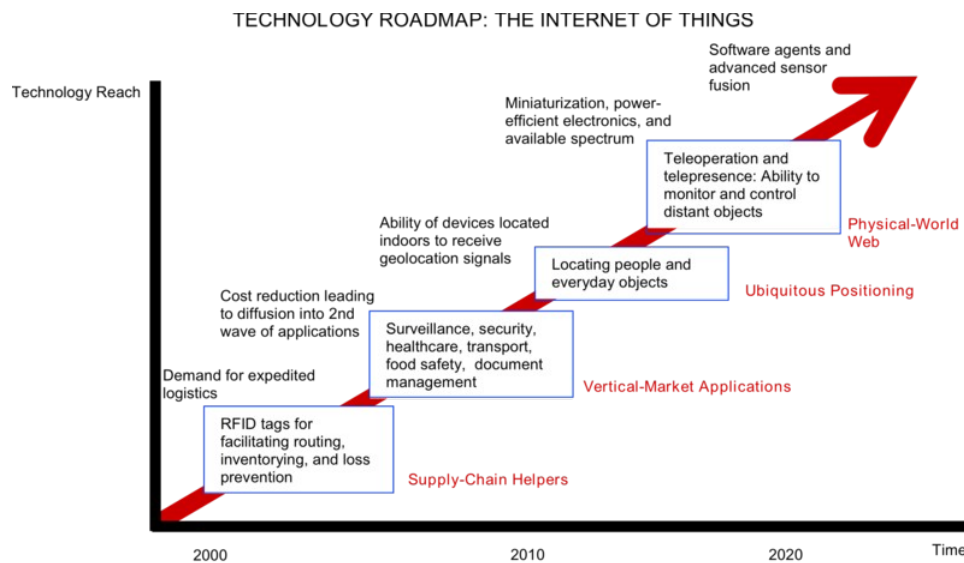
Aquesta distinció però s'ha anat tornat més difusa als darrers anys degut a la multiplicació de sistemes SoC (sistem-on-chip) on es construeixen en un sol circuit integrat la major part de mòduls que componen un ordinador o sistema informàtic. Aquest creixement es pot atribuir en bona mesura al alt ritme de popularització dels dispositius mòbils, que requereixen sistemes de dimensions i consums molt reduïts.

2.1. Estat de l'art.

El llançament al mercat recentment de dispositius com el Raspberry Pi, tot un ordinador de la mida de una targeta de crèdit per un preu de 35\$ la unitat, i la seva ràpida i àmplia adopció² juntament amb l'èxit que han demostrat altres dissenys cronològicament anteriors com, Arduino, ens mostren un escenari on els petits sistemes encastats guanyen cada dia més presència. En tots dos casos, ens trobem, al igual que amb el sistema microcontrolador LPC1769 adoptat en el present projecte, circuits accessibles a nivell electrònic, amb connectors genèrics que permeten poder afegir nou maquinari per adaptar les seves prestacions a les necessitats del usuari. A més a més, tots tres sistemes ofereixen una interfície USB amb la que accedir ràpida i còmodament al entorn de programació de cada dispositiu. I tot plegat per un cost molt moderat. No resulta estrany doncs que la popularització d'aquest tipus de dispositius hagi crescut amb rapidesa.

² En pocs mesos ja s'han comptabilitat més d'un milió de unitats venudes: <http://www.wired.com/design/2013/01/raspberry-pi-million-boards/>

Resulta clar veure que els esmentats sistemes segueixen un comportament semblant als dispositius mòbils en general, i és segurament aquesta una de les conseqüències que ha impulsat la creixent adopció dels sistemes SoC juntament amb d'altres fenòmens com ara la difusió dels sistemes oberts, en primera instància per programari i posteriorment en altres aspectes com el maquinari, passant per la literatura i la cultura en general, que han fet aparèixer un seguit de noves idees i possibilitats, plasmades en conceptes com ara el de l'Internet de les coses (Internet of Things)³ o la forta tendència de maquinari obert, amb múltiples exemples com ara els ordinadors portàtils del projecte OLPC (One Laptop Per Child), els telèfons mòbils OpenMoko, o la recent videoconsola OUYA.



Source: SRI Consulting Business Intelligence
 Figura 6: L'Internet de les coses

Fins i tot, conceptes com el de processadors oberts⁴ comencen a agafar força.

Tornant però al maquinari dels sistemes encastats, val la pena enumerar les principals característiques dels esmentats dispositius per tenir una idea més clara de la situació actual:

³ La 'Internet de les coses', el futur ja és aquí <http://www.escacc.cat/ca/contingut/la-internet-de-les-coses-el-futur-ja-es-aqui-3024.html>

⁴ The Free Software Foundation (FSF) announced today that it has published an initial set of criteria for endorsing computers and other devices. <http://www.fsf.org/news/endorsement-criteria>

Arduino Duemilanove, especificacions tècniques:

- Microprocessador Atmega168 a 16 Mhz
- Tensió operativa 5V
- Tensió de entrada (recomanada) 7-12V
- Tensió de entrada (límits) 6-20V
- Digital I/O Pins 14 (of which 6 provide PWM output)
- Entrades analògiques 6
- Memòria Flash (per codi) 16 KB (2 KB utilitzats per el bootloader)
- SRAM 1 KB
- EEPROM 512 bytes

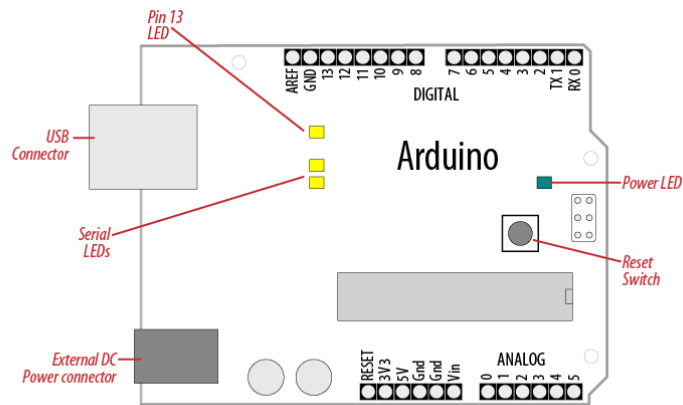


Figura 7: Esquema del sistema Arduino

Raspberry Pi Model B , especificacions tècniques:

- SoC: Broadcom BCM2835 (CPU, GPU, DSP, and SDRAM)
- Microprocessador: 700 MHz ARM1176JZF-S core (ARM11 family)
- GPU: Broadcom VideoCore IV, OpenGL ES 2.0, 1080p30 h.264/MPEG-4
- Memòria (SDRAM): 256 Megabytes (shared with GPU)
- Ports USB 2.0: 2 (via integrated USB hub)
- Adaptador xarxa: 10/100 Ethernet (RJ45)
- Low-level peripherals: 8 × GPIO, UART, I²C bus, SPI bus with two chip selects, +3.3 V, +5 V, Ground
- Onboard storage: SD, MMC, SDIO card slot
- Power ratings: 500 mA (2.5 W) 700 mA (3.5 W)
- Power source: 5 volt via MicroUSB or GPIO header
- Sortida vídeo: Composite RCA, HDMI
- Sortida audio: 3.5 mm jack, HDMI

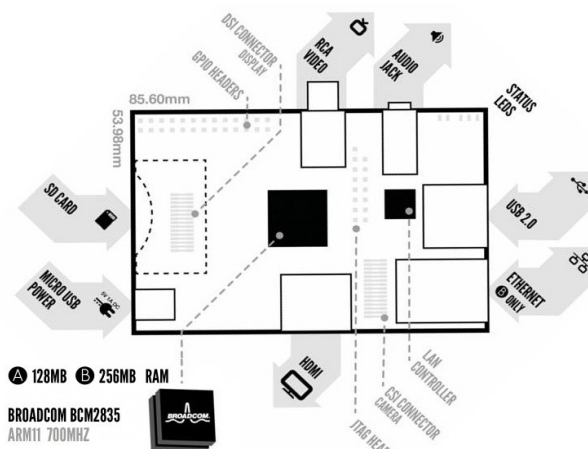


Figura 8: Esquema del sistema Raspberry Pi

NXP LPP1769, especificacions tècniques:

- Microprocessador ARM Cortex-M3 core with speed up to 120 Mhz
- Memòria Flash 512 kB
- Memòria RAM 64 kB
- Xarca Ethernet 10/100 Ethernet MAC
- Ports USB 2.0 1 USB 2.0 Host , 1 USB 2.0 Device , 1 USB 2.0 OTG
- Alimentació: 5 volts via MicroUSB o connector J6
- Low-level peripherals 2xCAN, 4xUART, 3xI2C, 2xSSP, 1xI2S
- Altres dispositius 4 32-bit counter/timers , 1 Watchdog timer , PWM/Motor, Motor control PWM with support for three-phase motor control, Quadrature encoder interface , 12-bit ADC multiplexing 8 pins , Single 10-bit DAC output , 3 power modes: Sleep, Deep-sleep, and Deep power-down

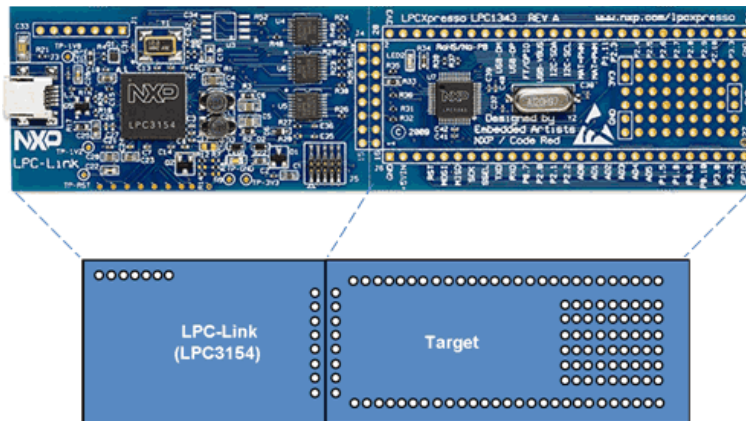


Figura 9: Conjunt i plaques components del LPC1769

Com es pot observar en les especificacions, a nivell de recursos com la CPU o la memòria podem apreciar diferències substancials entre els diferents dispositius, cal tenir present però, que per a molts dissenys amb sistemes encastats, aquestes diferències poden no constituir un factor determinant i si en canvi el nombre i tipus d'entrades i sortides o el consum energètic del sistema per posar dos exemples. No convé passar per alt, altres factors que poder acabar sent

determinants en els projectes com ara la capacitat de creixement del sistema dissenyat i el cost dels dispositius triats.

Respecte a les comunicacions, encara que dos dels sistemes LPC1769 i Raspberry incorporen un adaptador Ethernet per accedir a Internet, i en el cas de Arduino hi han disponibles diferents mòduls d'expansió anomenats Shields per afegir aquest tipus de connectivitat, cap dels tres incorpora un sistema de xarxa sense fils.

Aquest sembla un factor cada cop més important, doncs al igual que amb Internet, el potencial dels sistemes informàtics amb o sense connexió a la xarxa, canvia absolutament.

En aquest aspecte cal tenir en compte tres tecnologies d'àmplia difusió en les comunicacions sense fils. Parlem de les **WiFi**, **Bluetooth** i les **3G** per dades sobre infraestructura de telefonia mòbil.

En la següent comparativa poder veure quines són les deferències entre aquestes tecnologies:

| | Bluetooth | Wifi | 3G |
|-------------------------|------------------|-----------------|--|
| Bit-rate: | 2.1Mbps | 600Mbps | 3.1mbps |
| Abast: | 10 m | 100 m | Depèn de la infraestructura del operador |
| Latència: | 200ms | 150ms | 200 to 300 ms |
| Freqüència: | 2.4 GHz | 2.4, 3.6, 5 GHz | 1.8 – 2.5GHz |
| Ample de banda: | 800 Kbps | 11 Mbps | 384 kbit/s |
| estàndard: | IEEE802.15 | IEEE802.11 | CDMA 2000, UMTS, EDGE etc |
| Any de desenvolupament: | 1994 | 1991 | 2003 |
| Consum: | Baix | Alt | Baix |
| Seguretat: | Regular | Alta | Alta |
| Cost: | Baix | Alt | Alt |

Com es pot observar, les xarxes WiFi destaquen per la velocitat i com per l'abast però potser l'aspecte més importat el grau d'integració amb Internet.

2.2. Estudi de mercat.

Gràcies al èxit de projectes com Arduino, en aquests moments podem trobar una infinitat de vehicles i dispositius construïts amb aquest dispositiu com a base. També trobem tot tipus de kits educatius amb més o més grau de ser modificats amb uns preus molt raonables com per exemple un robot seguidor de línies anomenat LINE TRACKING MOUSE⁵ amb un preu de 25 Euros que ha de muntar el comprador. Això si, en aquest cas, i en molts d'altres el producte té unes característiques predefinides que normalment no podrem canviar. Es tracta de solucions tancades.

En canvi com dèiem, amb Arduino i la recent Raspberry Pi, la situació és força diferent. són productes dissenyats per ser modificats, per construir coses noves que requereixen la potència que proporciona la "intel·ligència" dels processadors. Podem agafar com a exemple el 3pi Robot⁶, que amb un cost més elevat que el exemple anterior, uns 100\$ ens obre una mica més les portes a poder modificar el producte.

Sembla però que els projectes més interessants i fins i tot en molts casos el més econòmics són els realitzats per aficionats a la robòtica que en base a sistemes encastats estan construint projectes realment interessants. Per exemple podem veure com es construeixen drons⁷ (petits helicòpters o avions).

5 Robot que segueix una línia sobre un fons blanc: http://www.diotronic.com/kits-modulos/kits-didacticos/kits-educacionales-varios/line-tracking-mouse_r_594_7307.aspx

6 Vehicle compatible Arduino: <https://www.sparkfun.com/products/9211>

7 Raspberry Pi Drone: http://www.theregister.co.uk/2012/06/12/raspberry_pi_drone/

3. Descripció funcional.

El conjunt del sistema ha estat desenvolupat al voltant del sistema microcontrolador LPC1769. Es tracta de un sistema prou complex com per encarar el disseny amb una estructura més propera a sistemes informàtics més grans, docs disposem d'un abundant nombre de recursos per gestionar. Per aquest motiu el sistema s'ha estructurat a partir de controladors per els diferents recursos de maquinari necessaris per completar els objectius del projecte. Amb aquests controladors s'ha creat una interfície tant per diferents ports de comunicació del microcontrolador com per als dispositius externs connectats al sistema com ara l'adaptador de xarxa sense fils o el sistema de detecció d'obstacles per ultrasons. Aquestes pautes ens permetrien poder substituir els dispositius per d'altres amb característiques semblants amb un baix impacte sobre la totalitat del sistema. Aquest tret també és present a l'aplicació client dissenyada per taules, portable entre un nombre molt divers de dispositius gràcies al framework adoptat.

3.1. Sistema total.

El sistema s'ha concebut com entorn on participen dos nodes: per un cantó el vehicle, un dispositiu autònom que es pot controlar de manera remota, accedint a un punt d'accés estàndard amb tecnologia sense fils WiFi. Un segon dispositiu connectat a aquesta mateixa xarxa, ja sigui amb fils o sense, localment o de forma remota serà capaç de controlar al vehicle.

El vehicle participa a la xarxa com un **node servidor**, en el sentit que és el que espera la connexió de xarxa, mitjançant el protocol de comunicacions Telnet. Al rebre una petició de connexió i ser acceptada, el dispositiu interpretarà els missatges rebuts i durà a terme les accions corresponents.

El **node client**, pot ser tant un ordinador amb un simple terminal de comandes com una aplicació Telnet basada en interfície gràfica o una aplicació específicament dissenyada, per transmetre les ordres específiques que accepta el sistema del vehicle. La comunicació entre els sistemes és bidireccional, aspecte especialment important en el cas de comandes amb les que s'obtingui informació del vehicle.

3.1.1. Diagrama de blocs.

En el següent diagrama, es mostra el conjunt del sistema on els nodes servidor i client interaccionen per mitjà de la xarxa, que pot ser tant distant com es necessiti, a través de Internet:

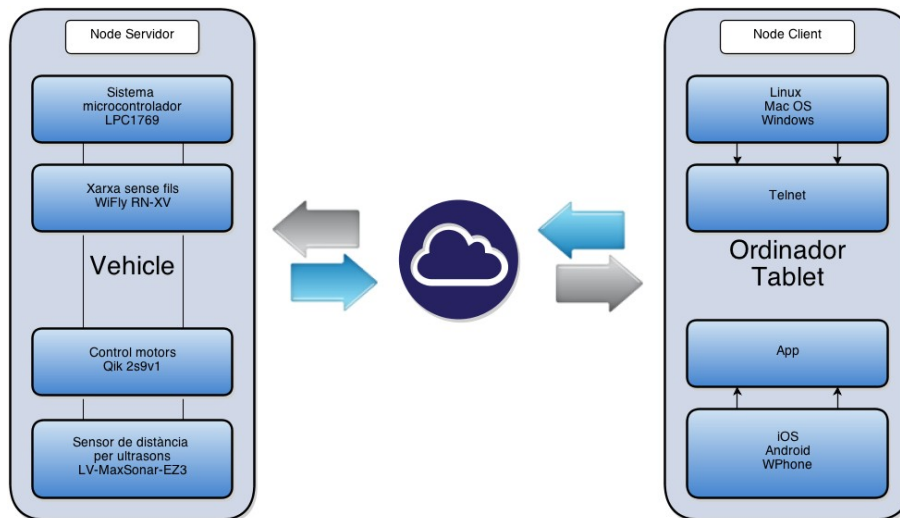


Figura 10: Diagrama de blocs, nodes servidor i client

3.1.2. Com és la xarxa

El vehicle dissenyat fa servir per comunicar-se un adaptador WiFi. Aquest adaptador sense fils, el WiFi RN-XV, està certificat per operar amb xarxes IEEE 802.11b/g a 2.4GHz. Els mètodes de accés a les xarxes sense fils que suporta inclouen tant xarxes obertes (open) com xarxes amb mètodes de autenticació WEP, WPA i WPA-PSK.

En primera instància, un cop s'ha posat en funcionament el vehicle, aquest realitza un procés de inicialització amb el que s'estableix la connexió al punt d'accés de la xarxa sense fils WiFi. Un cop establerta la connexió i obtinguda la configuració IP amb el servei DHCP (Dynamic Host Configuration Protocol) del punt d'accés, el node servidor ja estarà preparat per acceptar connexions.

Es en aquest punt, en funció de la configuració física que es faci del sistema per mitjà de un jumper, on definirem quin és mode de funcionament del vehicle, entre dos possibles: control automàtic o control remot. Tant en mode automàtic com en mode remot, el vehicle farà servir la xarxa sense fils WiFi, encara que de formes diferents. Al primer cas, al sistema li cal anar obtenint la informació respecte al nivell de senyal de la xarxa, mentre que en el segon, el accés a la xarxa constitueix requisit necessari per poder controlar el vehicle de forma remota.

El node client és el que iniciarà la comunicació, obrint una sessió Telnet contra la direcció IP del vehicle, que en el seu rol de servidor, un cop accepti la connexió, enviarà un missatge com a resposta per identificar la connexió establerta. En aquest punt, el servidor quedarà a la espera de rebre les comandes per part del client. Aquestes comandes es classifiquen en tres grups segons la seva finalitat: les de **configuració**, les que impliquen **accions** i les de **estat**.

Al primer grup, el de **configuració**, comptem amb comandes per modificar el comportament del sistema del vehicle, ens permeten indicar per exemple si el vehicle ens té que donar resposta a les properes comandes que rebí o per el contrari deixar de enviar-nos confirmació de les ordres rebudes.

El següent grup de ordres corresponen a les **accions** pròpies d'un vehicle, executant en darrera instància accions sobre el sistema de motors, amb les que el vehicle realitzarà per exemple desplaçaments o girs.

Finalment ens trobem amb el grup de ordres o comandes de **estat**, que ens permeten rebre informació del sistema servidor. Entre les més importants, tenim la comanda que ens indicarà a quina distància està detectant algun obstacle el sistema del sonar.

3.1.3. Com interactuen els diferents objectes en el sistema

Com s'ha indicat anteriorment, el sistema microcontrolador LPC1769 destaca entre d'altres aspectes, per oferir un elevat nombre de sistemes d'interconnexió amb d'altres dispositius. Entre ells en destaquen 4 ports UART (Universal Asynchronous Receiver-Transmitter), 3 ports I2C (Inter-Integrated Circuit), 1 port SPI (Serial Peripheral Interface Bus) i 70 ports GPIO (General Purpose Input/Output) així com ports USB i Ethernet .

També s'havia indicat, que un dels criteris que s'han seguit per la selecció dels dispositius és el seu voltatge de treball, doncs el controlador, encara que s'alimenta a 5V, treballar a una tensió de 3.3V.

Això vol dir que els dispositius que hi connectem als diferents ports també han de tenir aquest voltatge.

En aquest disseny, s'han fet servir per les comunicacions entre dispositius principalment els ports UART, doncs encara que es tracta de un dels protocols més antics de comunicació i en podem de trobar d'altres que ofereixen més velocitat i prestacions, constitueix un protocol molt conegut i és senzill de implementar i simplifica la confecció, depuració i manteniment del sistema.

Altres ports del sistema LPC1769 utilitzats en la construcció del vehicle han estat el ADC (Analog to Digital Converter) i el PWM (Pulse-Width modulation). El convertidor analògic a digital és el que connecta el sistema detector d'ultrasons al microcontrolador. Per la seva banda, el port PWM s'ha fet servir conjuntament amb un Led per oferir indicacions visuals del nivell de recepció de senyal a la xarxa WiFi en que estem connectats.

3.2. Disseny de l'aplicació al sistema encastat

S'ha dissenyat el vehicle a motor elèctric en base al sistema microcontrolador LPC1769 i equipat-lo amb un sistema de control sense fils via l'adaptador WiFly RN-XV. Mentre que el microcontrolador, amb nucli ARM a 120Mhz basat en l'arquitectura RISC ofereix un rendiment molt notable i alt grau de connectivitat, el mòdul Wifly RN-XV ens ofereix connectivitat a xarxes sense fils amb WiFi en un mòdul reduït, flexible i de baix consum. D'aquesta forma podrem controlar el vehicle amb qualsevol dispositiu que faci servir el protocol de Internet TCP/IP.

L'equip incorporar un controlador de motors, el mòdul Qik 2s9v1 que permet governar els dos motors que incorporar el xassís utilitzat. Finalment per el sistema de detecció d'obstacles per ultrasons, s'ha utilitzat el LV-Max-EZ3. El sistema també disposa de una interfície de depuració gràcies a una connexió de comunicacions sèrie. A aquesta interfície ens podrem connectar al ordinador amb l'adaptador per port sèrie UART-USB-CP2102.

Respecte a la connectivitat dels diferents dispositius, cal tenir en compte que el sistema microcontrolador s'alimenta amb una tensió externa de 4,5V a 5,5V però treballa a 3,3V en tots els seus ports, i és aquest, un dels motius que han condicional la tria de la resta de mòduls, que tinguin aquesta tensió de treball.

Per alimentar el vehicle, s'ha fet servir un conjunt de 5 piles AA de 1,5V. Com ja s'ha indicat amb anterioritat, la placa de la família LPCXpresso utilitzada es divideix en dos unitats independents, la que ofereix connectivitat amb al sistema de desenvolupament amb una connexió Mini-USB i la placa on es troben el microcontrolador i els circuits necessaris per el seu funcionament. Cal tenir en compte que amb la connexió USB, la placa de connectivitat proporciona tensió a tot el conjunt i per tant durant el desenvolupament no cal una font de alimentació externa.

La següents figures mostren a nivell estructural, l'organització de la placa LPC1769:

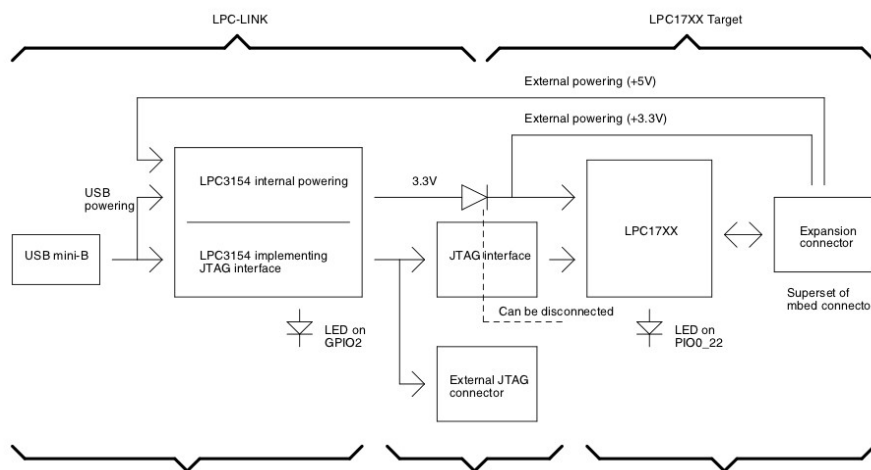


Figura 11: Estructura de la placa LPC1769

Per la construcció del sistema, s'ha incorporat també un detector de obstacles per ultrasons. Encara que el sonar triat ofereix altres mètodes de interconnexió com ara via port serie, s'ha triat un dels ports d'entrada ADC del LPC1769, a causa de la simplicitat que implica en el disseny i la possibilitat que ens ofereix per explorar diferents mètodes d'interacció entre els dispositius.

L'estructura de connexió adoptada per el vehicle es mostra al següent diagrama de blocs:

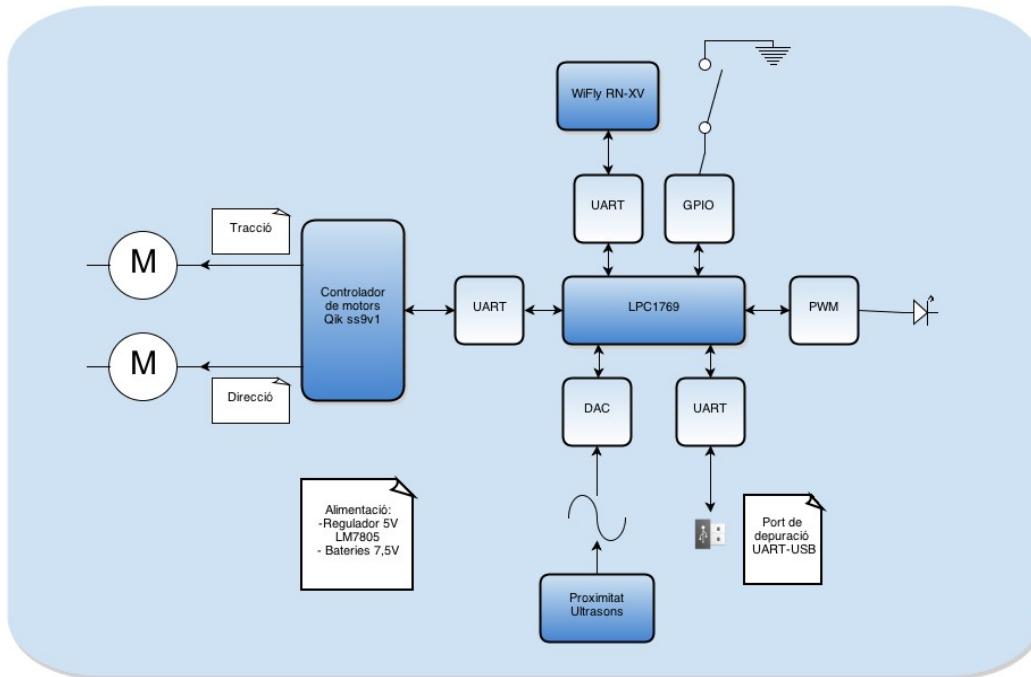


Figura 12: Interconnexió dels dispositius

Respecte al programari, disposem d'un firmware enregistrar el microcontrolador compondre per els diferents controladors i l'aplicació central s'encarreguen de:

- Fer la lectura inicial del jumper de configuració
- Inicialitzar els diferents dispositius connectats al sistema
- Establir la connexió a la xarxa sense fils WiFi
- En funció de la configuració:
 - Realitzar la lectura del nivell de senyal RSSI de la xarxa
 - Iniciar el procés de recepció de comandes com a servidor Telnet
- Realitzar lectures periòdiques del dispositiu de ultrasons
- Visualitzar amb el led de estat la informació corresponent, com ara el nivell RSSI
- Enviar amb el port depuració les notificacions de cada una de les operacions dutes a terme als dispositius.

Finalment, com a element central sobre el codi desenvolupat, s'ha fet servir el sistema operatiu en temps real per sistemes encastats FreeRTOS. Al tractar-se d'un sistema molt compacte amb nombre limitat i ben definit de mètodes per treballar amb tasques, semàfors i cues. Amb el seu us hem creat un codi que resulta fàcil d'interpretar i mantenir. Al treballar amb diferents controladors per als dispositius, tant la definició de **tasques** separades per realitzar determinades accions, com l'us de **semàfors** per controlar l'accés concurrent als ports de comunicació o les **cues** per intercanviar missatges entre tasques han resultat fonamentals, per aconseguir un sistema modular.

3.3. Disseny de la interfície d'usuari

Durant el desenvolupament del projecte s'ha seguit una filosofia de claredat. Per aquest motiu a la interfície de comunicació amb el dispositiu remot, s'ha optat per fer servir el protocol de comunicacions Telnet. Aquest és un dels protocols més veterans a Internet, dissenyat per crear sessions de comandes remotes o també anomenats terminals virtuals contra un sistema, normalment multi-usuari. Al fer servir aquest mecanisme, aconseguim d'una forma fàcil transmetre comandes de text i rebre les respostes entre els sistemes. Aquest protocol està implementat per sobre la capa TCP, de forma que el protocol ens garantirà la entrega dels missatges.

Els avantatges que ens aporta aquest mètode de comunicació són, per una banda, que resulten fàcils d'interpretar i fer servir per una persona i a l'hora simples d'automatitzar dins un programa.

Amb una sessió de terminal podem obrir amb l'ordinador una connexió Telnet cap al vehicle i enviar-hi comandes com es mostra tot seguit:

```
cesc@cesc-Aspire:~$ telnet 192.168.1.36 2000
Trying 192.168.1.36...
Connected to 192.168.1.36.
Escape character is '^]'.
*Ready*
$$sonar
cm:251.20
```

Figura 13: Inici de la sessió Telnet

Per que el sistema pugui identificar les comandes, aquestes van precedides dels caràcters \$\$. En cas de disposar de algun valor associat a la comanda, aquest s'especifica amb el símbol dos punts

seguit del valor. Per completar l'ordre caldrà un final de línia amb la tecla Enter (↵). Per exemple la comanda **\$\$led:30↵** estariem indicant al sistema de vehicle que el Led indicador tingui un lluminositat de valor 30.

Aquesta implementació del sistema de comunicació, ens ha permès desenvolupar posteriorment una interfície gràfica tàctil sobre tauleta, que fa servir aquest protocol juntament amb les comandes definides per realitzar les diferents accions amb el vehicle.

La següent figura mostra el disseny de l'aplicació de control per taula:



Figura 14: Aplicació de control remot per tauleta tàctil

De tot el grup de comandes implementades sobre el sistema del vehicle elèctric, a la interfície gràfica s'ha fet us de les següents instruccions:

Comanda \$\$init

Aquesta comanda indica al sistema remot que executi una inicialització dels motors, així com l'estat de les variables de configuració, que controlen per exemple el sistema de control d'obstacles.

Comanda \$\$f:nn

Per controlar el sistema de tracció, fent que el motor de tracció giri en sentit d'avanç per el vehicle. El paràmetre nn permet especificar la velocitat en que girarà el motor.

Comanda \$\$b:nn

Igual que la comanda anterior, per controlar el sistema de tracció, fent que el motor de tracció giri en sentit de retrocedir per el vehicle El paràmetre nn permet especificar la velocitat en que girarà el motor.

Comanda \$\$l:nn

De forma semblant a les comandes anteriors, per controlar el sistema de direcció, fent que el motor de direcció giri en el vehicle cap a l'esquerra. El paràmetre nn permet especificar la força amb que la girarà el motor.

Comanda \$\$l:nn

De forma semblant a les comandes anteriors, per controlar el sistema de direcció, fent que el motor de direcció giri en el vehicle cap a l'esquerra. El paràmetre nn permet especificar la força amb que la girarà el motor.

Comanda \$\$r:nn

Complementaria a l'anterior, per controlar el sistema de direcció, fent que el motor de direcció giri en aquest cas el vehicle cap a la dreta. El paràmetre nn permet especificar la força amb que la girarà el motor.

Comanda \$\$stop

Aquesta comanda actuarà tant sobre el motor de tracció com el direcció, fent que el vehicle s'aturi.

Comanda \$\$sonar

Gràcies al sistema de detecció per ultrasons, podem consultar amb aquesta comanda la distància en centímetres del obstacle més proper detectat per el dispositiu. Aquest té un abast mínim de detecció de uns 15,24 cm, per tant qualsevol obstacle que es trobi a una distància inferior, no es podrà determinar correctament quina és la seva distància real respecte al sensor.

Comanda \$\$led:nn

Per obtenir certa informació visual sobre el vehicle s'ha incorporat un led sobre el que podem controlar la intensitat amb que s'il·lumina. Amb aquesta comanda podem especificar amb el valor nn aquesta intensitat. A més a més, si el valor que pot oscil·lar entre 1 i 1200 el situem a zero, obtindrem com a resposta una fluctuació constant de la intensitat del Led.

Comanda \$\$obstacles

Aprofitant el sistema de detecció per ultrasons, amb aquesta comanda podrem indicar si el vehicle s'ha aturar automàticament en detectar un obstacle a una distància inferior al valor predeterminat de 25cm. Aquest sistema es troba actiu per defecte, de forma que al introduir per primera vegada la comanda deixa d'estar actiu, i si tornem a enviar la comanda una segona vegada aquest tornarà a estar actiu.

Respecte al sistema de depuració, s'ha optat per fer servir un dels ports UART de comunicació sèrie disponibles al microcontrolador, per rebre al ordinador la informació generada per el sistema encastat. Aquesta connexió s'ha realitzat físicament per mitja d'un adaptador SERIAL-USB.

La següent captura mostra els missatges rebuts durant un procés de inicialització realitzat amb èxit, rebuts per el port sèrie virtual ttyUSB0 amb una velocitat de transmissió de 115200 bauds:

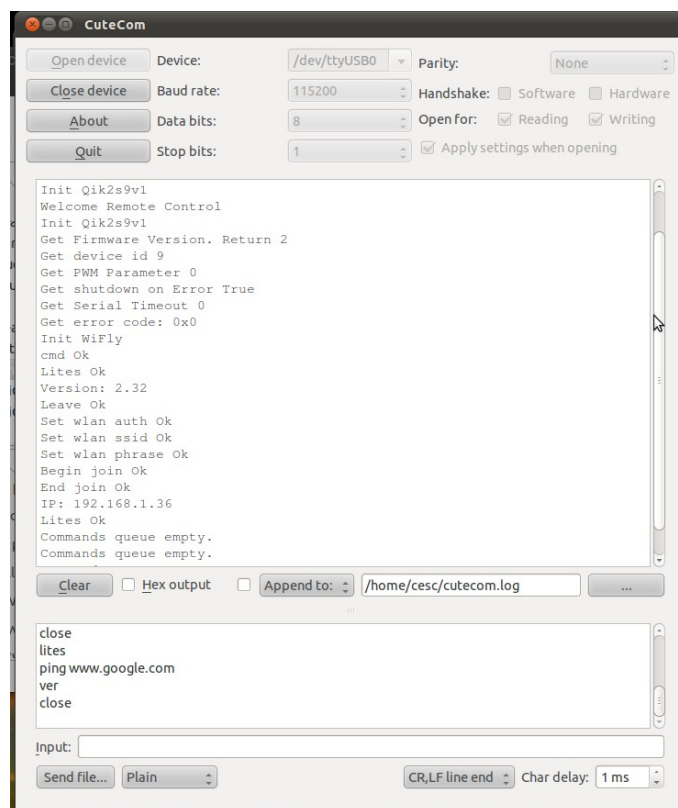


Figura 15: Client de port sèrie per depuració

4. Descripció detallada del sistema.

En aquest apartat explicarem els diferents apartats de l'apartat anterior de forma més detallada, aprofundint en els detalls tècnics i presentant diferents diagrames per mostrar de forma més clara possible com s'ha realitzat el disseny.

4.1. Disseny del vehicle

Per començar aprofundirem en els detalls relatius a l'estructura del vehicle. Veurem, per una banda el desenvolupament del programari que controla tot l'equip: el firmware. Tot seguit es mostrarà el disseny físic amb un primer prototip que ens portarà a concloure amb disseny de la placa final.

4.1.1. El sistema LPC1769: disseny de controladors:

Donat que el disseny ha estat centrat al voltant del sistema microcontrolador, els primers passos que s'han seguit, han estat dirigits a disposar d'una llibreria per controlar els diferents ports del sistema, segons els requeriments del sistema. Recordem que tant l'adaptador sense fils com el controlador de motors es connecten al sistema microcontrolador per mitjançant el ports sèrie UART. També farem servir aquest tipus de connexió al port de depuració.

Al crear un projecte amb l'entorn de desenvolupament **LPCXpresso**, sobre la placa LPC1769, l'entorn ens permet especificar unes llibreries anomenades CMSIS que defineixen per els sistemes ARM Cortex-M de forma estandarditzada, l'accés al registres hardware dels perifèrics. Així mateix també proporcionen un conjunt de fonts d'exemple amb els que es mostra la utilització dels diferents ports del microcontrolador. Aquests exemples han estat el punt de partida per confeccionar les llibreries per els ports **UART**, **PWM** i **ADC** utilitzats al sistema.

S'ha posant un èmfasi especial als ports UART, doncs són els que més s'han fet servir al projecte. A més a més, com que les velocitats de transmissió entre els diferents mòduls por ser força diferent, ha estat fonamental aconseguir que no es generes cap problema amb aquestes comunicacions. Amb aquestes finalitats, aquesta llibreria disposa de un conjunt de funcions que permeten realitzar lectures i escriptures a les diferents UART seguin el mateix estil de les funcions

printf i **scanf** de les biblioteques estàndard de C, adaptades això si a les peculiaritats d'aquest sistema de comunicacions. També s'ha aprofitat les característiques del sistema **FreeRTOS** per proporcionar control de concurrència a les funcions, de forma que es pugui treballar amb el sistema de tasques que proporciona el sistema FreeRTOS.

Per sobre d'aquesta llibreria per ports sèrie, s'ha creat una segona llibreria dedicada al controlador de comunicacions sense fils **WiFi RN-XV**. Aquest dispositiu ofereix un ampli conjunt de comandes en forma de sentències de text intel·ligibles per a una persona, agrupades en diferents categories segons el tipus d'accions que realitzen: configuració i consulta per , paràmetres de Adhoc, Broadcast, Comunicacions/Transferències, DNS, FTP, IP, Opcions, Sistema, Time, UART i WLAN. Encara que la llibreria no proporciona una API exhaustiva a totes les funcionalitats disponibles en el dispositiu, si que dona cobertura a les característiques més importants i proporciona un bon punt de referència per ampliar i completar el controlador. Al igual que en el controlador UART, s'ha tingut en compte el control de concurrència a les funcions, per permetre el correcte funcionament al treballar amb el sistema de tasques de FreeRTOS.

També, en contacte directe amb la llibreria per els ports UART, s'ha creat la tercera llibreria dedicada en aquest cas al controlador de motors **Qik 2s9v1**. A diferència del controlador anterior, aquest mòdul té un conjunt de comandes molt més reduït, i a més a més codificades de forma numèrica, situació que donada la quantitat de instruccions, no complica gaire la seva interpretació, No obstant, per millorar la claredat del codi, s'han di finit etiquetes per cada un dels codis amb el nom de la comanda corresponent.

En aquest dispositiu les comandes s'agrupen en dos categories: les de configuració/estat i les de operacions sobre els motors. A diferència dels controladors anteriors, en aquest cas no s'ha contemplat la possibilitat de treballar de forma concurrent amb el dispositiu, doncs no té gaire sentit que més d'un procés interactuï amb els motors de forma simultània. Un aspecte interessant a tenir en compte al fer servir les funcions d'aquest controlador és la distinció que cal fer entre aturar els motors fixant la seva velocitat a 0 o amb la funció `MotorCoast()`, on a diferència de l'anterior, el que fa és deixar el eix del motor lliure, sense que s'apliqui cap força per frenar l'eix.

La implementació d'aquest darrer mòdul ens porta a la utilització dels **motors** incorporats al **xassís**. Encara pugui semblar estrany, un dels elements que menys ha influït en el disseny del vehicle ha estat el xassís. A causa dels endarreriments en la recepció del producte, s'ha fet servir el xassís de una joguina en comptes del pressupostat. Tal i com es pot apreciar en les següents figures, són força diferents:



Figura 16: Xassís previst inicialment.

En el primer xassís, es fan servir els dos motors per la tracció, de forma que per controlar la direcció caldrà fer anar els motors de manera simultània a velocitats diferents. El vehicle girarà amb més o menys velocitat en funció de aquesta diferència de velocitat entre els motors.



Figura 17: Xassís del vehicle fet servir finalment.

En canvi amb el xassís utilitzat al projecte, s'ha fet servir un dels motors per la tracció i l'altre altre per controlar la direcció. Encara que les accions de control de direcció i tracció es tindran que programar de forma diferent, l'impacte sobre el disseny resulta molt petit.

El circuit de detecció d'obstacles per ultrasons **LV-MaxSonar-EZ3** és també força polivalent respecte als formes de connexió, Fer servir la comunicació sèrie va ser la primera intenció, permetent així reutilitzar el codi desenvolupat. En principi aquest és el dispositiu amb el que resulta més fàcil la interacció, dons no calen comandes per obtenir la informació del sonar. De forma periòdica va enviant missatges de text per el port sèrie amb la distància a la que detecta l'obstacle. Hi ha però un petit detall que passa fàcilment desapercbut, a les seves especificacions tècniques. La sortida del port sèrie proporciona el senyal invertit, de forma que cal una porta NOT o una NAND fent comunes les serves entrades per poder rebre el senyal sèrie a l'entrada del microcontrolador. Per no afegir més electrònica al disseny i per aprofitar i explorar altres mètodes d'entrada, s'ha optat per fer servir la sortida analògica proporcionada per aquest mòdul i fer una conversió cap a senyal digital amb ADC que incorpora el LPC1769. Novament s'ha implementat en forma de llibreria la part de programació que realitza les operacions sobre els registres de maquinari del ADC.

Al fer servir aquest mètode d'entrada, han estat necessari uns petits càlculs per determinar la distància de l'obstacle en funció de la lectura realitza del senyal.

La funció de conversió que proporciona el fabricant del detector per ultrasons és la següent:

$$V_{cc} / 512 = V_i$$

on **V_{cc}** correspon a la tensió d'alimentació del circuit que, en aquest cas són 3,3V i el terme **V_i** fa referència a la fracció de volts per polzada (Volts per Inch) .

Per altra banda el mostreig del senyal al ADC es fa amb una precisió de 12 bits, el que vol dir que la lectura estarà en un rang de $2^{12}=4096$ valors possibles, és a dir que per un senyal de 3,3V d'entrada el valor amb el que es representarà serà el màxim 4095.

Operant amb la expressió podem representar la funció de conversió com:

$$\text{distancia_cm} = ((V_i / 4095) * 512) / (1 / 2,54)$$

Finalment, tenim al circuit dos Led, un per indicar que el circuit té alimentació per mitja de les bateries i un segon Led que podrem controlar dins el sistema per proporcionar informació visual del mode de operació i en el cas del funcionament automàtic del vehicle, representar el nivell de senyal da xarxa RSSI. Com es tracta poder regular la intensitat amb que l'il·lumina el Led, podem fer servir el mateix mecanisme que s'aplica al control de motors: **PWM** aplicar pols a una freqüència prou

alta com per que no sigui perceptible i ajusta l'amplada d'aquest pols per tenir més o menys temps de tensió, i aconseguir en proporció a 'aquesta diferents nivells de intensitat lluminosa.

Aquesta darrera tasca el realitzarà novament amb una llibreria dedicada als ports PWM que ofereix el microcontrolador.

Un cop desenvolupats els diferents mòduls controladors, s'ha dut a terme la relativament senzilla tasca de implementar el codi central que gestiona el sistema. La interacció entre el sistema central i els diferents controladors està representat en la següent figura:

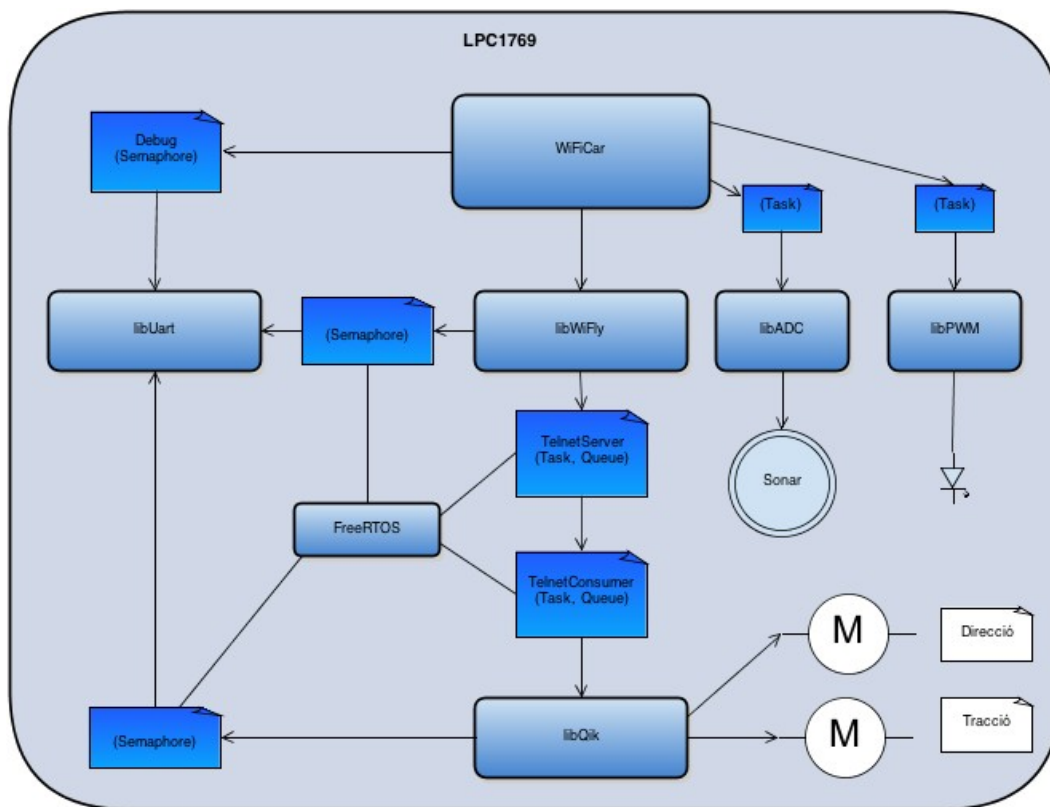


Figura 18: Diagrama de blocs amb els controladors de dispositius

Dins l'espai de treball que engloba els diferents projectes per al mòdul LPC1769, es defineix com a punt d'entrada, el projecte WiFiCar, on s'implementa el codi central del sistema. Aquest farà referència a la resta de mòduls creant les diferents tasques i cues de missatges per el funcionament del sistema.

Recordem que el vehicle disposa de dos modes de funcionament. En cas del mode de control remot, un cop completat el procés de inicialització, el sistema iniciarà el procés de servidor Telnet que ha de permetre controlar el vehicle. Ho veiem al diagrama de flux:

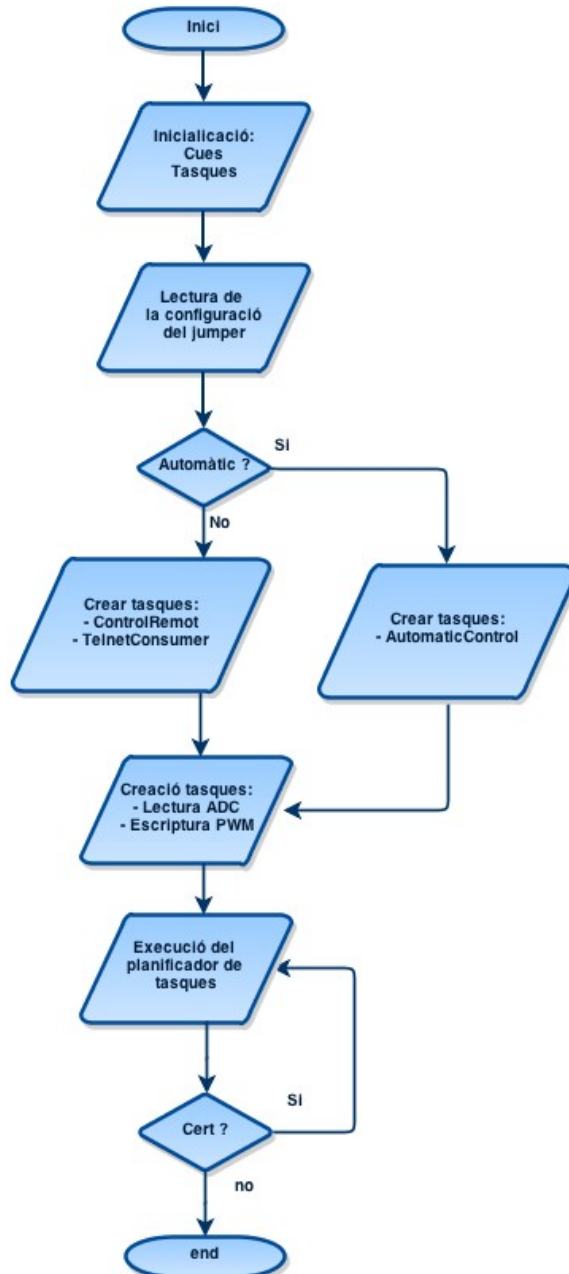


Figura 19: Diagrama de flux, sistema principal del vehicle electrònic amb microcontrolador.

Com mostra el diagrama anterior un cop realitzades les inicialitzacions, s'inicia el planificador de tasques de FreeRTOS i el funció principal queda retinguda en un bucle sense fi. A la següent figura es pot veure, respecte a les diferents llibreries de controladors desenvolupades, com amb l'ajuda de les funcions del sistema operatiu, les diferents tasques utilitzen les funcions de les llibreries segons el mode de funcionament triat al vehicle:

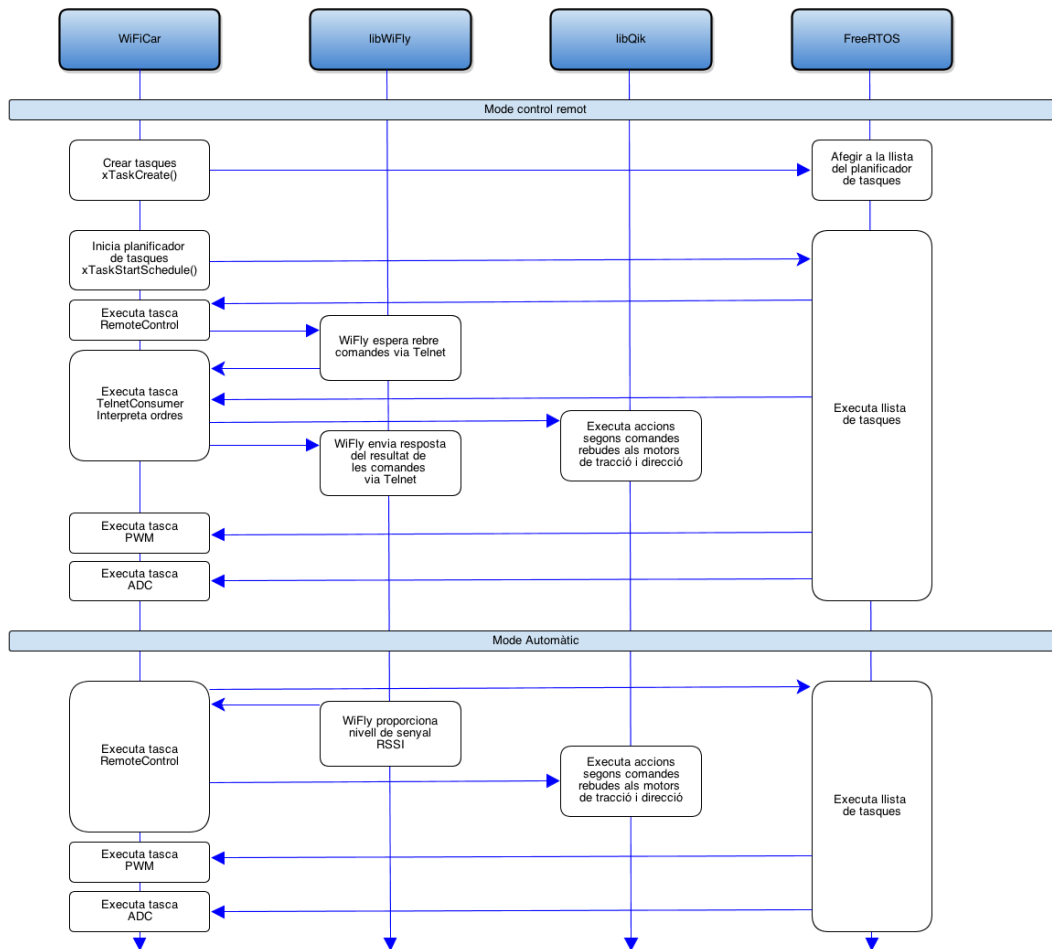


Figura 20: Interacció entre tasques i llibreries

Finalment cal veure les operacions que podem realitzar amb el sistema del vehicle, per mitjà de les comandes que es poden rebre gràcies a la comunicació Telnet entre els sistemes desenvolupats. El vehicle admet un conjunt de comandes que detallem a la següent taula:

| Comanda | Acció | Descripció |
|------------------|------------------------------|--|
| \$\$f:nn | Forward M0 | Tracció al motor 0 |
| \$\$b:nn | Backward M0 | Tracció al motor 0 |
| \$\$fd:nn | Forward M1 | Direcció al motor 1 |
| \$\$bd:nn | Backward M1 | Direcció al motor 1 |
| \$\$l:nn | Turn Left M1 | Gir amb el motor 1 |
| \$\$r:nn | Turn Right M1 | Gir amb el motor 1 |
| \$\$stop | Stop M0 and M1 | Atura els motors 0 i 1 |
| \$\$sonar | Get sonar distance | Lectura de sonar |
| \$\$obstacles | Toggle obstacles detection | Activa / desactiva la detecció d'obstacles |
| \$\$obstacles:nn | Obstacles detection distance | Defineix la distància en la detecció d'obstacles |
| \$\$status | Toggle status responses | Activa / desactiva la resposta a les comandes |
| \$\$led:nn | Set led power. | Especifica la intensitat de Led |
| \$\$init | Set default configuration | Inicialitza la configuració del sistema |
| \$\$exit | Close Telnet server | Finalitza el servidor de Telnet |

Figura 21: Relació de comandes de control per el vehicle.

En aquest projecte, amb el xassís utilitzat, on la tracció es controla amb el motor M0 i la direcció amb el motor M1, les comandes \$\$fd i \$\$bd no tenen aplicació pràctica, doncs estan concebudes per ser utilitzades amb el sistema on els dos motors s'encarreguen del moviment de tracció i s'ha de definir velocitats diferents entre els motors per fer girar el vehicle.

També cal fer notar que les diferents ordres de moviment prenen un límit en els valors de velocitat dels motors. Aquests límits s'han establert de forma empírica, al anar fent les proves de funcionament del vehicle un cop muntat.

Com a ajuda per la introducció de comandes al terminal, si fem Enter sense cap altre caràcter, es repetirà la darrera comanda introduïda.

També, amb finalitats per l'aprenentatge i la depuració, s'ha permès poder enviar comandes al propi adaptador WiFly RN-XV. Per fer-ho només cal enviar una primera comanda amb la seqüència amb els símbols \$\$\$ i tot seguit qualsevol comanda reconeguda per aquest dispositiu com ara la ordre ver que ens indicarà la versió del firmware del dispositiu. Encara que això s'ha fet amb finalitats didàctiques, cal tenir present que aquesta forma de modificar la configuració del adaptador podria esdevenir un risc de seguretat, segons la finalitat del sistema.

4.1.2. Disseny del prototip:

A mida que s'han anat creant els diferents controladors, ha estat necessari dur a terme les diferents connexions entre els mòduls electrònics. La següent figura mostra com s'ha fet el muntatge sobre un suport anomenat Protoboard, que està concebut per poder comprovar el funcionament dels dissenys:

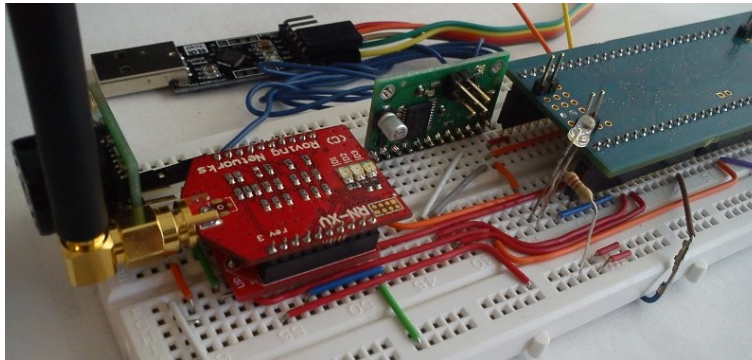


Figura 22: Protoboard amb connexió dels diferents mòduls.

Aquests primers muntatges ens han ajudat a trobar la distribució més adequada per la placa final. Cal remarcar que a banda de les connexions d'alimentació i entre els diferents ports, també s'ha seguit la consideració de connectar les línies de RESET tant del controlador WiFi com del mòdul de motors, cap a sortides del microcontrolador. Durant el procés de inicialització de cada mòdul, es fa us d'aquesta connexió per produir un procés de reinicialització a nivell de maquinari i garantir d'aquesta forma que el dispositiu es trobi operatiu i en un estat conegut abans de començar a realitzar les diferents operacions.

Per alimentar el vehicle s'ha optat per fer servir un regulador de tensió LM7805CV al que s'han connectat 5 piles LR6 (AA) de 1.5V. Amb els condensadors indicats per el fabricant, proporcionant una funció estabilitzadora de tensió:

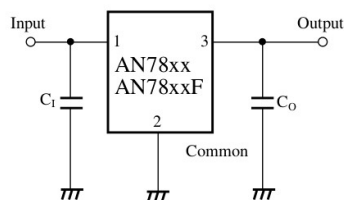


Figura 23: Regulador de tensió

Al disseny s'ha optat per fer servir connectors femella de 0,1 polzades per tots els mòduls utilitzats. Això permet que el circuit sigui més fàcil de mantenir i en simplifica les possibles modificacions o ampliacions.

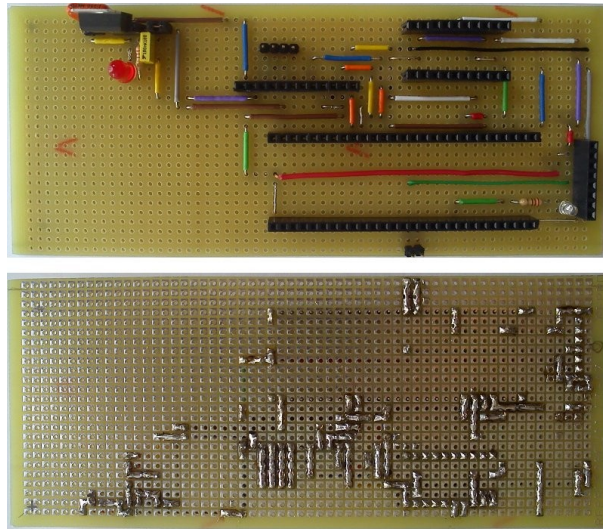


Figura 24: Placa final amb connectors i components

Les mateixes piles que alimenten la electrònica de control es fan servir per proporcionar tensió als motors mitjançant la placa Qik 2s9v1, però connectades abans del regulador de tensió amb l'objectiu de minimitzar els efectes de oscil·lació a la tensió de alimentació a la electrònica de control.

La placa de motors incorpora el circuit integrat TB6612FNG específicament dissenyat per control de petits motors DC (de corrent continu) de fins 1 ampere de corrent constant. Internament, el circuit fa servir una configuració anomenada Pont H, en referència a la representació típica del gràfic del circuit:

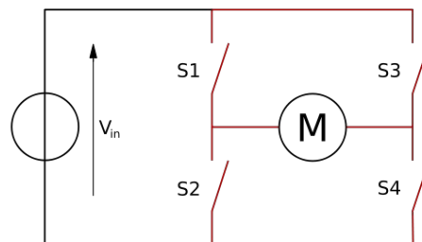


Figura 25: Configuració en Pont H

4.2. Aplicació de control remot

Per controlar el vehicle a banda de l'us directe amb un ordinador, fent servir un terminal de comandes de text per obrir una sessió amb client Telnet, també ha dissenyat una aplicació per interfície gràfica.

Aquesta ha estat desenvolupada amb el llenguatge de programació orientat a objectes C#. Encara que aquest llenguatge pertany a un grup de tecnologies de Microsoft anomenant .Net per entorns Windows, existeix una implementació del programari lliure anomena Mono. Això ens permetrà fer servir aquest llenguatge així com algunes biblioteques sobre plataformes que no són les de Microsoft, en el cas d'aquest projecte, sobre el sistema iOS, present a la tauleta iPad de Apple.

Per aquests dispositius, l'empresa Xamarin ofereix un entorn comercial que integra totes les eines necessàries per poder fer servir aquest framework.

Com es pot observar en el diagrama de classes, en el disseny s'ha aplicat el patró MVC (Model Vista Controlador) que separa dades, interfície i lògica d'aplicació en tres capes.

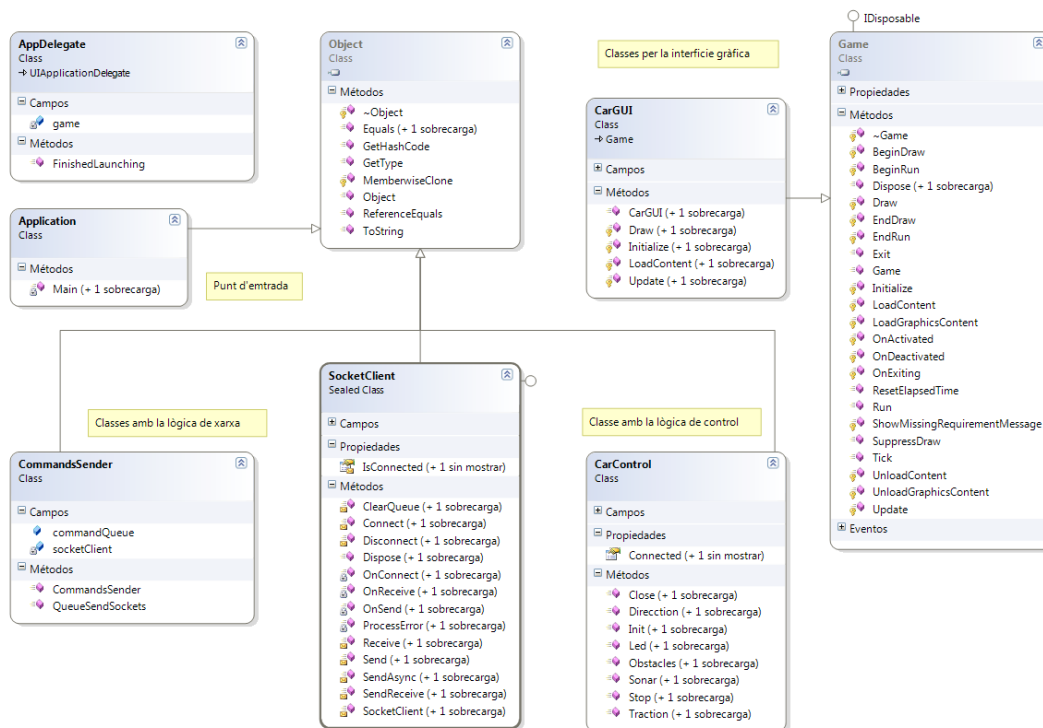


Figura 26: Diagrama UML, estructura de classes.

La classe CarGUI correspon a la interfície d'usuari. Aquesta deriva de la classe Game que proporciona la biblioteca MonoGame, una implementació Open source de la API XNA destinada a aplicacions gràfiques d'alt rendiment, especialment orientada als videojocs.

Aquesta classe es la responsable de recollir i interpretar les accions sobre la pantalla tàctil, així com la càrrega i presentació dels gràfics que donen representació als controls per pantalla.

La programació de la lògica dels controls es troba implementada a la classe CarControl, on disposem de mètodes per establir la comunicació i per efectuar les diferents accions com el control de direcció o la petició de informació sobre la distància dels obstacles.

Per millorar la interacció amb la interfície gràfica, s'ha implementat la classe CommandSender, que gestiona una cua amb les diferents comandes que genera la capa de lògica d'aplicació i permet, juntament amb la classe SocketClient⁸, donar una resposta fluida en la interacció amb la pantalla tàctil.

A la lògica de control, per millorar el temps de reacció del vehicle a l'acceleració inicial, s'ha fet servir una funció logarítmica per transformar l'entrada lineal de la velocitat. Això produir un ràpid creixement de la velocitat que un primer moment, que després va suavitzant tal i com mostra la figura:

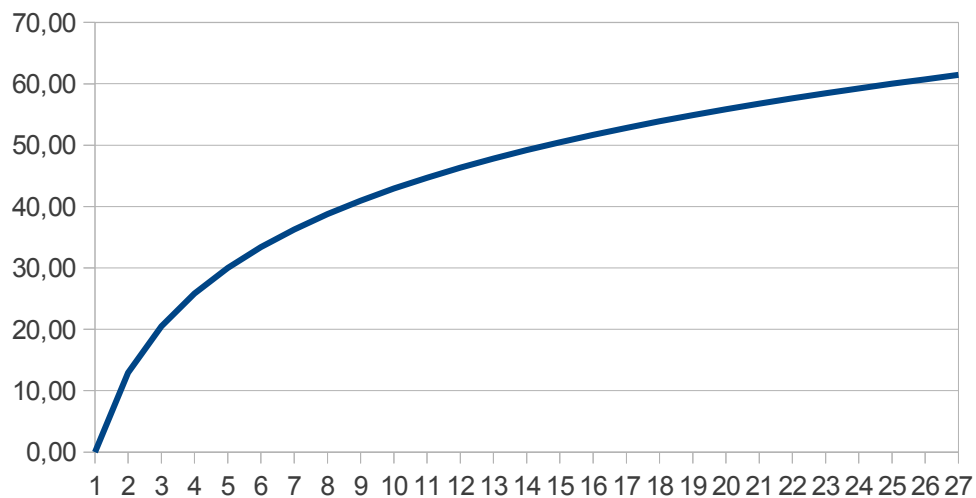


Figura 27: Funció de transformació a la velocitat

⁸ Aquesta classe és una versió adaptada procedent de un article a CodeProject sobre Sockets asíncrons de Marcos Hidalgo Nunes. <http://www.codeproject.com/Articles/22918/How-To-Use-the-SocketAsyncEventArgs-Class>

5. Viabilitat tècnica.

El resultat del projecte ens ha permès obtenir un vehicle que compleix els requisits i que ofereix un al grau de flexibilitat a l'hora de plantejar-ne modificacions, tant per la construcció modular que s'ha plantejar a nivell físic com a nivell de programació, on cada mòdul està gestionat per una llibreria independent. A més a més comptem amb un microcontrolador de una potència de comput suficient com per desenvolupar projectes més exigents. A tal d'exemple, entre les diferents llibreries disponibles al sistema, n'inclou una per realitzar funcions de DSP, destinada al tractament de senyals. Això ens permetria poder fer tractament tant a nivell de so com d'imatge, amb els que podríem realitzar processos de reconeixent.

En aquest projecte la diversitat de ports de comunicació del sistema ens ha permès triar amb total llibertat els mòduls a fer servir i ens dona la tranquil·litat de saber que disposem de un nombre encara molt ampli de connexions lliure per afegir noves funcions al sistema. Queda clar doncs que ens trobem davant d'un disseny molt escalable.

Respecte a les comunicacions, el disseny també ens dona via lliure per plantejar projectes més ambiciosos, donada la alta difusió de les xarxes sense fils WiFi, cada cop presents a més tipus de dispositius.

Cal fer notar però que, ni a nivell de microcontrolador ni respecte al adaptador de xarxa WiFi, hem dut a terme tasques intensives per mirar de identificar quins són els límits, aspecte que seria interessant explorar per tenir una noció més precisa del que és pot arribar a fer amb cada un dels dispositius.

6. Valoració econòmica.

Tot seguit es presenta el pressupost per la realització del projecte que inclou tant els costos dels materials, la ma d'obra i el cost de lloguer dels recursos utilitzats per no imputar al projecte el cost d'adquisició dels mateixos, doncs entenem que en aquest cas caldria un període de amortització d'aquests béns.

| Producte | Descripció | Uts. | Preu ut. | Preu IVA | Import |
|-----------------------|--|-------|----------|----------|----------------|
| LPC1769 | Sistema microcontrolador | 1 | 20,00 | 24,20 | 24,20 |
| WiFi RN-XV | Mòdul xarxa sense fils WiFi | 1 | 34,95 | 42,29 | 42,29 |
| Qik 2s9v1 | Controlador de motors serial | 1 | 19,90 | 24,08 | 24,08 |
| LV-MaxSonarEZ3 | Detector per ultrasons | 1 | 25,80 | 31,22 | 31,22 |
| LM7805CV | Regulador de tensió 5V | 1 | 1,65 | 2,00 | 2,00 |
| Magician Chassis | Xassis amb motors DC | 1 | 14,95 | 18,09 | 18,09 |
| Leds | Díodes Led vermell i blau | 2 | 0,50 | 0,61 | 1,21 |
| Resistències | Resistències 3300 oms | 2 | 0,30 | 0,36 | 0,73 |
| Sòcol xBee | Sòcol adaptador per WiFi 0,1" | 1 | 2,50 | 3,03 | 3,03 |
| Base bat AA x3 | Base per bateries AA x3 | 1 | 0,68 | 0,82 | 0,82 |
| Base bat AA x2 | Base per bateries AA x2 | 1 | 0,65 | 0,79 | 0,79 |
| Tira 40 pins f. | Tira pins femella PCB rectes | 2 | 0,99 | 1,20 | 2,40 |
| Placa topos | Placa fibra topos 100x160 2,54 | 1 | 8,16 | 9,87 | 9,87 |
| Cable kit | 1x 140 Pcs Solderless Breadboard Jumper Cable Wire Kit | 1 | 5,38 | 6,51 | 6,51 |
| Tira pins 10 m. | Tira pins mascle PCB rectes | 2 | 5,38 | 6,51 | 13,02 |
| Tramesa | Despeses de les trameses dels productes | 3 | 7,95 | 9,62 | 28,86 |
| Servei desenvolupador | Comput d'hores dedicades íntegrament al desenvolupament | 100 | 25,00 | 30,25 | 3025,00 |
| Lloguer ordinador | Ordinador portàtil, cost de lloguer per temps de un dia | 30 | 4,00 | 4,84 | 145,20 |
| Lloguer Monotouch | Llicència de entorn, cost de lloguer per temps de un dia | 30,00 | 2,00 | 2,42 | 72,60 |
| Lloguer instrumental | Material de treball: polímetre, soldador, protoboard, etc. | 30 | 0,50 | 0,61 | 18,15 |
| Total | | | | | 3385,41 |

Per el preu/hora corresponents als serveis del desenvolupador s'ha fet una estimació prenent com a referència els preus de empreses de consultoria informàtica, encara que caldria entrar en més detall respecte al procés de fabricació del sistema, tenint en compte que es tracta de un prototip.

7. Conclusions.

En aquest punt farem un breu valoració del aspectes més rellevants del projecte, així com una enumeració de les diferents millores que es podrien dur a terme. Finalment es fa una reflexió en termes d'autoavaluació per valorar quins aspectes s'haguessin pogut enfocar d'una forma diferent.

7.1. Conclusions

Un cop finalitzat el projecte podem concloure que la planificació s'ha ajustat de forma prou adequada als objectius del projecte i que aquests han estat assolits de forma satisfactòria.

El vehicle, encara que no s'ha pogut construir amb el xassís projectat en un inici, és perfectament funcional i ens ha permès assolir tots els objectius plantejats, tant els primaris com el secundaris. La excepció potser ha estat el sistema de control automàtic, en dos sentits. Per una banda ens hem trobat amb una dificultat en la comunicació amb l'adaptador WiFi. Durant el temps que un equip estableix una connexió per Telnet amb el dispositiu, ens hem trobat amb un comportament inesperat, al deixar aquest de rebre i enviar informació per el seu port sèrie. Això impedeix que puguem realitzar operacions sobre el mòdul WiFi amb el microcontrolador durant el temps que es troba aquesta connexió oberta. Aquest condicionant ens ha obligat a establir el mode de funcionament del vehicle per mitja de un jumper, encara que el desitjable hagués estat poder fer-ho amb el sistema de comandes implementat.

També es tindria que dir que, respecte al sistema de funcionament automàtic, la expectativa al inici era la de fer que el vehicle es mogués no tant sols en línia recta per cercar el punt de senyal WiFi optim, si no que, es desplaçés en diferents direccions, mirant d'establir diferents punts del pla amb els nivells de senyal associats per poder triangular la posició del punt d'accés. Finalment però aquesta idea es va abandonar donada la dificultat de determinar les diferents posicions relatives en les que es podria situar el vehicle i el grau de variabilitat del senyal RSSI en un mateix punt del plànol.

7.2. Proposta de millores

Un dels aspectes que estan força limitats en el vehicle és la configuració de la xarxa sense fils. Tots aquests paràmetres estan definits a nivell de codi en el firmware del dispositiu. Aquest és un aspecte que clarament podria millorar. Una segona opció en aquest sentit seria fer servir l'adaptador de xarxa en mode ad-hoc de forma que no es tingués que establir una connexió al punt d'accés, fent el propi vehicle de element de xarxa amb el que connectar. També es tindria que valorar quins paràmetres podríem configurar remotament com ara la pròpia configuració de xarxa.

El sistema de comandes per Telnet també es podria dissenyar de forma més entenedora, admetent per exemple comandes més intel·ligibles com per exemple que `$$r` i `$$right` fossin sinònims de la mateix acció.

L'aplicació de tauleta també admet millores clares. Per exemple no hi ha implementat cap apartat de configuració, de forma que no es pot modificar l'adreça IP del vehicle remot ni cap altre paràmetre configurable. També es podrien provar alternatives a la comunicació Telnet per mirar de obtenir una comunicació més fluida oferint, per exemple, mètodes alternatius a la sessió Telnet.

7.3. Autoavaluació

En general la feina realitzada ha estat satisfactòria, la sensació és que hi han molts aspectes on valdria la pena aprofundir més, però per el temps disponible crec que el projecte ha sigut prou profitós. Seria interessant poder provar el sistema de comunicació sense fins amb el protocol UDP en comptes de TCP, en especial per mirar de millorar la resposta amb l'aplicació tàctil.

En tot cas, l'objectiu principal, el de conèixer de més aprop el sistemes encastats esta clarament assolit.

8. Glossari.

Adhoc: A les xarxes de comunicació, especialment sense fils, on no hi ha un node central, tots els dispositius estan en igualtat de condicions.

API: (de l'anglès Application Programming Interface) és un conjunt de declaracions que defineix el contracte d'un component informàtic amb qui farà ús dels seus serveis.

Broadcast: Mètode per transferir un missatge a tot un conjunt de receptors, de forma simultània.

CRC: (de l'anglès Cyclic Redundancy Check) sistema detector d'errors aplicat als sistemes de comunicació digitals, que genera codis de verificació.

DNS: (de l'anglès Domain Name System) és un sistema de traducció de noms a adreces IP, que funciona sobre una base de dades distribuïda.

Firmware: Bloc d'instruccions de programa per a propòsits específics, gravats en una memòria de tipus no volàtil.

FTP: (de l'anglès File Transfer Protocol) és un protocol estandarditzat per enviar fitxers entre ordinadors amb qualsevol sistema operatiu.

Host: Terme per referir-se als equips connectats a la xarxa que proveeixen i fan servir serveis en aquesta.

JIT: compilació en temps d'execució (de l'anglès Just-In-Time) també coneguda com traducció dinàmica, és una tècnica per millorar el rendiment de sistemes de programació que compilen a bytecode.

Jumper: Mecanisme que es fa servir en equipaments electrònics per tancar la connexió entre els pins d'un circuit.

RSSI: (de l'anglès Received signal strength indication) es un terme que es fa servir per mesurar el nivell de potencia dels senyals rebuts a les xarxes sense fils.

SoC: (de l'anglès System-on-a-chip) descriu la tendència cada vegada més freqüent d'usar tecnologies de fabricació que integren tots o gran part dels mòduls components d'un ordinador o qualsevol altre sistema informàtic o electrònic en un únic circuit integrat o xip.

Sonar: (de l'anglès Sound Navigation And Ranging) és un sistema de navegació i localització similar al radar però que emet impulsos ultrasònics.

TCP: (de l'acrònim de Transmission Control Protocol) és un protocol orientat a la connexió dintre del nivell de transport del model OSI que permet l'entrega de paquets de manera fiable, en el cas de TCP anomenats segments.

UART: (de l'anglès Universal asynchronous receiver/transmitter) és una peça de maquinari que tradueix dades entre formes paral·leles i serials.

UDP: (de l'anglès User Datagram Protocol) és un protocol del nivell de transport del model OSI, basat en l'intercanvi de datagrames. UDP permet l'enviament de datagrames a través d'una xarxa sense que s'hagi establert prèviament una connexió.

WiFi: (de l'acrònim de Wireless Fidelity) és una marca per a un conjunt d'estàndards de compatibilitat per a comunicacions per a xarxes locals sense fils (WLAN).

WLAN: (de l'anglès Wireless Local Area Network) és un sistema de comunicació de dades sense fils flexible, molt utilitzat com a alternativa a les xarxes d'àrea local

Xbee: Nom comercial per a una família de mòduls de comunicació sense fils del fabricant Digi International.

Xcode: Entorn de desenvolupament per el sistema MacOS sobre el llenguatge Objective-C.

9. Bibliografia.

[1] **Microcontrolador Wiki**, (28/8/2012). Dedicat TFC com a introducció al sistema encastat utilitzat, url: <http://cv.uoc.edu/app/mediawiki14/wiki/IniciCortexM3>

[2] **knowledgebase NXP**, (20/9/2012). Fòrum del fabricant del microcontrolador LPC1769, amb exemples de codi, FAQ, etc. url: <http://knowledgebase.nxp.com>

[3] **FreeRTOS**, (16/9/2012). Pàgina oficial del sistema operatiu per sistemes encastats, llibreries, mòduls de maquinari, url <http://www.freertos.org>

[4] **WiFly Adaptador WiFi**, (14/12/2012) Pàgina del fabricant Roving Networks amb la descripció i detalls tècnics sobre el mòdul de xarxa sense fils, url: http://rovingnetworks.com/products/RN_XV

[5] **Sistemes encastats**, (23/9/2012) Informació general sobre els sistemes encastats, url: http://en.wikipedia.org/wiki/Embedded_system

[6] **Internet of Things** , (4/1/2013) Descripció detallada de conceptes, serveis, entorns de treball relacionats amb el concepte, url: http://en.wikipedia.org/wiki/Internet_of_Things

[7] **LV-MaxSonar-EZ**, (8/12/2012) Quick Start Guide. provided to serve as an easy to use set-up guide for the ultrasonic sensors. url: <http://www.maxbotix.com/articles/016.htm>

[8] **LPC1769 Pin function table**, (27/10/2012) proporciona una descripció detallada de la configuració de pins per al LPC1769, url: http://nanohome.be/nxp/LPC1769_Pins.php

10. Annexos.

10.1. Ports de connexió del LPC1769

El sistema microcotrolador LPC1769 de NXP i es caracteritza per tenir un nombre elevat de pins de connexió. A més a més cada un dels pins es pot configurar per dur a terme diferents funcions. Per oferir la màxima flexibilitat, poder fer servir un mateix port en diferents pins del circuit. Per aquest motiu s'han confeccionat en base als manuals proporcionats per el fabricant i amb la ajuda de informació procedent d'Internet les següents taules on es detallen les diferents funcions que poden realitzar els cada un dels terminals de la placa. La columna J6 és la que indica cada un dels terminals del connector mascle del circuit.

NXP LPC1769

| Descrip | Ref | J6 | PIN | FUN1 | FUN2 | FUN3 | FUN4 |
|--------------|-------------|--------------|------------|---------------------|-----------|-----------|--------|
| | GNDX | J6-1 | | GND | | | |
| | EXT_POWX | J6-2 | | VIN (4.5-5.5V) | | | |
| | VB | J6-3 | | VB (battery supply) | | | |
| | RESET_N | J6-4 | 17? | nRESET | | | |
| | P0[9] | J6-5 | 76 | P0.9 | I2STX-SDA | MOSI1 | MAT2.3 |
| | P0[8] | J6-6 | 77 | P0.8 | I2STX-WS | MISO1 | MAT2.2 |
| | P0[7] | J6-7 | 78 | P0.7 | I2STX-CLK | SCK1 | MAT2.1 |
| | P0[6] | J6-8 | 79 | P0.6 | I2SRX-SDA | SSEL1 | MAT2.0 |
| UART3 | P0[0] | J6-9 | 46 | P0.0 | RD1 | TXD3 | SDA1 |
| UART3 | P0[1] | J6-10 | 47 | P0.1 | TD1 | RXD3 | SCL1 |
| | P0[18] | J6-11 | 60 | P0.18 | DCD1 | MOSI0 | MOSI |
| | P0[17] | J6-12 | 61 | P0.17 | CTS1 | MISO0 | MISO |
| | P0[15] | J6-13 | 62 | P0.15 | TXD1 | SCK0 | SCK |
| | P0[16] | J6-14 | 63 | P0.16 | RXD1 | SSEL0 | SSEL |
| | P0[23] | J6-15 | 9 | P0.23 | AD0.0 | I2SRX-CLK | CAP3.0 |
| | P0[24] | J6-16 | 8 | P0.24 | AD0.1 | I2SRX-WS | CAP3.1 |
| | P0[25] | J6-17 | 7 | P0.25 | AD0.2 | I2SRX-SDA | TXD3 |
| | P0[26] | J6-18 | 6 | P0.26 | AD0.3 | AOUT | RXD3 |
| | P1[30] | J6-19 | 21 | P1.30 | VBUS | AD0.4 | |
| | P1[31] | J6-20 | 20 | P1.31 | SCK1 | AD0.5 | |
| UART0 | P0[2] | J6-21 | 98 | P0.2 | TXD0 | AD0.7 | |
| UART0 | P0[3] | J6-22 | 99 | P0.3 | RXD0 | AD0.6 | |
| | P0[21] | J6-23 | 57 | P0.21 | RI1 | RD1 | |
| | P0[22] | J6-24 | 56 | P0.22 | RTS1 | TD1 | |
| | P0[27] | J6-25 | 25 | P0.27 | SDA0 | USB-SDA1 | |
| | P0[28] | J6-26 | 24 | P0.28 | SCL0 | USB-SCL1 | |
| | P2[13] | J6-27 | 50 | P2.13 | EINT3-N | I2STX-SDA | |

| | | | | | | | |
|--|---------------|-------|----|-------|-------------|-------------|--------|
| | P1[18] | PAD1 | 32 | P1.18 | USB-UP-LED | PWM1.1 | CAP1.0 |
| | P1[19] | PAD2 | 33 | P1.19 | MCOA | USB-PPWR-N | CAP1.1 |
| | P1[20] | PAD3 | 34 | P1.20 | MCFB0 | PWM1.2 | SCK0 |
| | P1[21] | PAD4 | 35 | P1.21 | MCABORT | PWM1.3 | SSEL0 |
| | P1[22] | PAD5 | 36 | P1.22 | MCOB | USB-PWRD | MAT1.0 |
| | P1[23] | PAD6 | 37 | P1.23 | MCFB1 | PWM1.4 | MISO0 |
| | P1[24] | PAD7 | 38 | P1.24 | MCFB2 | PWM1.5 | MOSI0 |
| | P1[25] | PAD8 | 39 | P1.25 | MC1A | MAT1.1 | |
| | P1[26] | PAD9 | 40 | P1.26 | MC1B | PWM1.6 | CAP0.0 |
| | P1[27] | PAD10 | 43 | P1.27 | CLKOUT | USB-OVRCR-N | CAP0.1 |
| | P1[28] | PAD11 | 44 | P1.28 | MC2A | PCAP1.0 | MAT0.0 |
| | P1[29] | PAD12 | 45 | P1.29 | MC2B | PCAP1.1 | MAT0.1 |
| | P3[25] | PAD13 | 27 | P3.25 | MAT0.0 | PWM1.2 | |
| | P3[26] | PAD14 | 26 | P3.26 | STCLK | MAT0.1 | PWM1.3 |
| | P4[28] | PAD15 | 82 | P4.28 | RX-MCLK | MAT2.0 | TXD3 |
| | P4[29] | PAD16 | 85 | P4.29 | TX-MCLK | MAT2.1 | RXD3 |
| | P0[19] | PAD17 | 59 | P0.19 | DSR1 | SDA1 | |
| | P0[20] | PAD18 | 58 | P0.20 | DTR1 | SCL1 | |
| | P2[9] | PAD19 | 64 | P2.9 | USB-CONNECT | RXD2 | |

| Descrip | Ref | J6 | PIN | FUN1 | FUN2 | FUN3 | FUN4 |
|--------------|-----------------|--------------|-----|--|-----------|-----------|--------------|
| | VIO_3V3X | J6-28 | | VOUT (+3.3V out) if self powered, else +3.3V input | | | |
| | Not used | J6-29 | | | | | |
| | Not used | J6-30 | | | | | |
| | Not used | J6-31 | | | | | |
| (Ethernet) | RD- | J6-32 | | RD- | RD- | | |
| (Ethernet) | RD+ | J6-33 | | RD+ | RD+ | | |
| (Ethernet) | TD- | J6-34 | | TD- | TD- | | |
| (Ethernet) | TD+ | J6-35 | | TD+ | TD+ | | |
| | USB-D- | J6-36 | | USB-D- | | | |
| | USB-D+ | J6-37 | | USB-D+ | | | |
| | P0[4] | J6-38 | 81 | P0.4 | I2SRX-CLK | RD2 | CAP2.0 |
| | P0[5] | J6-39 | 80 | P0.5 | I2SRX-WS | TD2 | CAP2.1 |
| | P0[10] | J6-40 | 48 | P0.10 | TXD2 | SDA2 | MAT3.0 |
| | P0[11] | J6-41 | 49 | P0.11 | RXD2 | SCL2 | MAT3.1 |
| UART1 | P2[0] | J6-42 | 75 | P2.0 | PWM1.1 | TXD1 | |
| UART1 | P2[1] | J6-43 | 74 | P2.1 | PWM1.2 | RXD1 | |
| | P2[2] | J6-44 | 73 | P2.2 | PWM1.3 | CTS1 | TRACEDATA[3] |
| | P2[3] | J6-45 | 70 | P2.3 | PWM1.4 | DCD1 | TRACEDATA[2] |
| | P2[4] | J6-46 | 69 | P2.4 | PWM1.5 | DSR1 | TRACEDATA[1] |
| | P2[5] | J6-47 | 68 | P2.5 | PWM1.6 | DTR1 | TRACEDATA[0] |
| | P2[6] | J6-48 | 67 | P2.6 | PCAP1.0 | RI1 | TRACECLK |
| | P2[7] | J6-49 | 66 | P2.7 | RD2 | RTS1 | |
| | P2[8] | J6-50 | 65 | P2.8 | TD2 | TXD2 | |
| | P2[10] | J6-51 | 53 | P2.10 | EINT0-N | NMI | |
| | P2[11] | J6-52 | 52 | P2.11 | EINT1-N | I2STX-CLK | |
| | P2[12] | J6-53 | 51 | P2.12 | EINT2-N | I2STX-WS | |
| | GNDX | J6-54 | | | | | |

| | | | | | | | |
|--|------|--|----|-------|-----------|-----|--|
| | None | | 29 | P0.29 | USB-D+ | | |
| | None | | 30 | P0.30 | USB-D- | | |
| | None | | 86 | P1.17 | ENET-MDIO | | |
| | None | | 87 | P1.16 | ENET-MDC | | |
| | None | | 88 | P1.15 | ENET-REF | CLK | |
| | None | | 89 | P1.14 | ENET-RX | ER | |
| | None | | 90 | P1.10 | ENET-RXD1 | | |
| | None | | 91 | P1.9 | ENET-RXD0 | | |
| | None | | 92 | P1.8 | ENET-CRS | | |
| | None | | 93 | P1.4 | ENET-TX | EN | |
| | None | | 94 | P1.1 | ENET-TXD1 | | |

