

Alarmino: Alarma per la llar

Marcos Puente Ruiz

Enginyeria Tècnica en Informàtica de Gestió
Sistemes encastats

Jordi Bécares Ferrés

Pere Tuset Peiró

Xavi Vilajosana Guillen

13/06/2016



Aquesta obra està subjecta a una llicència de [Reconeixement-NoComercial-SenseObraDerivada 3.0 Espanya de Creative Commons](https://creativecommons.org/licenses/by-nc-nd/3.0/es/)

FITXA DEL TREBALL FINAL

Títol del treball:	<i>Alarmino: Alarma per la llar</i>
Nom de l'autor:	<i>Marcos Puente Ruiz</i>
Nom del consultor/a:	<i>Jordi Bécares Ferrés</i>
Nom del PRA:	<i>Pere Tuset Peiró Xavi Vilajosana Guillen</i>
Data de lliurament (mm/aaaa):	<i>07/2016</i>
Titulació o programa:	<i>Enginyeria Tècnica en Informàtica de Gestió</i>
Àrea del Treball Final:	<i>Sistemes encastats</i>
Idioma del treball:	<i>Català</i>
Paraules clau	<i>Arduino, radiofreqüència, autònom</i>
Resum del Treball	
<p>Alarma per a la llar basada en microcontroladors <i>Arduino</i> i amb comunicació per radiofreqüència i sistema GSM. El projecte s'engloba en l'àrea dels sistemes encastats.</p> <p>L'alarma està dotada de sis mòduls independents que s'encarreguen de gestionar els sis sensors diferents de que consta el sistema, a més del mòdul principal encarregat de comunicar-se amb aquests. El mòdul central està equipat amb una pantalla LCD de 16x2 i quatre botons per controlar el menú d'usuari de l'aplicació.</p> <p>El menú disposa de quatre opcions, entre les que es troben el canvi de clau, l'activació de dos modes diferents i desactivació de l'alarma.</p> <p>El sistema d'avisos és GSM. <i>Alarmino</i> envia missatges de text al telèfon mòbil de l'usuari si es detecta qualsevol alarma.</p> <p>El resultat final és un sistema de seguretat autònom, capaç de detectar moviment, presència per proximitat, pas per línia per infrarojos, talls de llum, risc d'incendi i risc d'inundació.</p> <p>Cal destacar també el baix cost que aquest projecte ha suposat, degut a la utilització de maquinari y programari lliures, com són els micro-controladors <i>Arduino</i> i el seu entorn de programació <i>Arduino IDE 1.6.8 Release</i>.</p>	
Abstract	

Índex

1.	Introducció	1
1.1	Context i justificació del treball	1
1.2	Descripció del treball	2
1.3	Objectius del TFC	3
1.4	Enfocament i mètode seguit	4
1.5	Planificació del treball	5
1.6	Recursos emprats	7
1.7	Productes obtinguts	9
2.	Antecedents	9
2.1	Estat de l'art	9
2.2	Estudi de mercat	10
3.	Descripció funcional	11
3.1	Alarmino: Alarma per la llar	13
a)	Diagrama de blocs de l'aplicació	13
b)	Xarxa de comunicacions	14
c)	Com interactuen els diferents objectes al sistema	16
3.2	Alarmino HMI	17
a)	Diagrama de blocs i/o esquema	18
4.	Descripció detallada	18
4.1	Hardware del sistema	21
4.2	Software del sistema	22
4.3	Programació	23
4.4	Problemes	24
5.	Viabilitat tècnica	25
6.	Valoració econòmica	27
7.	Conclusions	28
7.1	Autoevaluació	28

7.2	Línies de futur	28
8.	Glossari	30
8.	Bibliografia	31
9.	Annexos	33
Annex 1.	Geografia	33
Annex 2.	Esquema elèctric bloc control	34
Annex 3.	Esquema elèctric bloc comunicació	35
Annex 4.	Esquema elèctric blocs detecció	35
Annex 5.	Execució i compilació	36
Annex 6.	Guía d'instal·lació	38

Llista de figures

Figura 1.	Diagrama de Gantt	6
Figura 2.	Maquinari emprat pel prototip	8
Figura 3.	Diagrama de blocs	13
Figura 4.	Topologia de xarxa cel·lular	14
Figura 5.	Taula de pins d'interrupció d'Arduino	15
Figura 6.	Topologia de xarxa en estrella	16
Figura 7.	Estructura paquet de comunicacions	19
Figura 8.	Protocol de comunicacions	20
Figura 9.	Taula cost material prototip	27

1. Introducció

L'alarma de seguretat proposada com a TFC consisteix en un sistema autònom gestionat per diversos micro-controladors i compost a més de diversos sensors, un sistema de radiofreqüència (RF) y de tecnologia GSM. Tot això amb l'objectiu de dissenyar i construir un sistema de seguretat totalment funcional. L'alarma proposada com a TFC és capaç de detectar risc d'incendis, detectar moviment i proximitat en una zona delimitada, mitjançant sensors d'ultrasons [4] o infrarojos [5], entre d'altres, i està dotada amb un sistema GSM per avisar els usuaris enviant-los un SMS al telèfon en cas de detectar algun perill.

El projecte s'ha desenvolupat en base a la micro-controladora *Arduino* i el seu entorn de programació *Arduino IDE 1.6.8* i per tant basat en programari i maquinari lliures. Tenint en compte aquest aspecte i el fet que el prototip desenvolupat és totalment autònom s'ha escollit com a nom del TFC: ***Alarmino: Alarma per la llar.***

Alarmino està equipat amb tot el necessari per poder treballar de forma totalment autònoma, és a dir que no necessita estar connectat a un ordinador per funcionar, ni tampoc necessita estar connectat a la corrent elèctrica de la vivenda. A més al estar basat en uns micro-controladors i un llenguatge de programació ambdós amb llicència lliure, cal destacar el baix cost que ha suposat la realització del prototip. El sistema de seguretat compta amb totes les funcions necessàries per assegurar una zona determinada, com pot ser una vivenda o un local.

Compta amb comandament a distància, possibilitat d'instal·lar cada un dels sensors de forma independent sense necessitat de cablejat, i gràcies als mòduls RF que incorpora, i compta també amb un sistema d'avisos silenciosos, discret i efectiu, ja que notifica les alarmes enviant un SMS a l'usuari per mitjà de la tecnologia GSM.

1.1 Context i justificació del treball

La realització d'aquest projecte ha estat motivada pel baix nombre de sistemes encastats autònoms i basats en maquinari lliure *Arduino* capaços de funcionar sense equips informàtics. Analitzant diferents projectes es va poder comprovar que la gran majoria dels nodes receptors d'informació requerien d'un ordinador per poder interpretar les dades recaptades, per aquesta raó es va plantejar la possibilitat de desenvolupar un sistema autònom, el qual comptés amb capacitat per a gestionar les dades i alertar d'una situació anòmla.

El sistema compta amb la possibilitat de variar la quantitat de sensors amb la qual es realitzen les captures de valors en relació a la temperatura, ultrasons, moviment, etc. Aquestes

característiques fan que l'equip pugui ser modular i en condicions on no es disposi d'una infraestructura per albergar equips més potents, el sistema resulta veritablement pràctic. A més ofereix la possibilitat de la seva utilització en diferents usos, tant domèstic o professional, ja que variant els intervals de mesurament podem demanar més valors, obtenint resultats més precisos.

Ha estat d'especial importància la possibilitat que el maquinari lliure brinda a l'hora de ser modificat per treballar com un equip totalment diferent a la seva funció principal, de manera que en aquests moments la capacitat de l'alarma per a la llar pot ser ampliada pel que a funcions es refereix: afegir sensors, nous mòduls Shield o canviar totalment la seva utilització. Característica avalada pels seus pins d'entrades i sortides analògiques i digitals de què *Arduino* disposa, sense necessitat d'utilització de circuits preimpresos ni d'equips elèctrics com soldadors, etc.

1.2 Descripció del treball

El projecte consisteix a desenvolupar una alarma per a la llar basada la tecnologia ja esmentada, composta per un node coordinador capaç d'interactuar amb l'usuari i els nodes independents encarregar de detectar i enviar els avisos de les incidències. Tots nodes o mòduls compten amb la peculiaritat de ser autònoms, és a dir, no necessiten cap ordinador per a rebre, interpretar les dades i alertar.

El mòdul principal o de control compta amb una interfície d'usuari (HMI) i una pantalla LCD dotada de cinc botons per al control del menú de l'aplicació, per tal que l'usuari pugui configurar, activar o desactivar el sistema de seguretat.

Alarmino està pensat per ser un guàrdia silenciós, atès que alerta directament enviant un SMS al telèfon mòbil de l'usuari o fent una trucada si s'escau, però no està pensada per repel·lir les possibles invasions mitjançant sirenes o alertes lluminoses.

Una altra peculiaritat de la versatilitat del sistema és la facilitat d'instal·lació que suposa, atès que cada mòdul sensor és independent del demés, és a dir que no estan connectats físicament ja que es comuniquen amb el mòdul central i el node GSM per radiofreqüència. Això dóna l'avantatge que els mòduls sensor poden ser col·locats a la vivenda a qualsevol indret sense necessitat de cablejat.

Si a més li sumem l'opció de controlar el mòdul principal per mitjà d'un comandament a distància, tenim una alarma per la llar totalment usable i funcional.

1.3 Objectius del TFC

Tots els objectius proposats a la planificació del TFC, considerats necessaris per a obtenir una aplicació robusta i complerta, han estat assolits amb èxit. A més s'ha decidit incorporar un dels tres objectius extraordinaris proposats a l'inici.

A continuació es detalla la totalitat de les fites proposades i assolides del projecte *Alarmino: Alarma per la llar*:

- **Interfície d'usuari (HMI)** per configurar y controlar les funcions d'usuari d' *Alarmino* a través d'una pantalla LCD de 16x2 i un keypad d'entrada.
- **Sistema de comunicacions per RF** per establir el protocol de comunicació entre els components independents del sistema.
- **Disseny dels mòduls 'sensor' autònoms** que s'encarreguen de gestionar cada un dels sensors i comunicar-se amb el sistema central per RF.
 - Hi ha quatre mòduls d'aquest tipus, un per cada sensor de l'alarma. (temperatura, moviment, infrarojos i ultrasons). Són els següents:
 - **Mòdul de temperatura**, que s'encarrega de controlar el risc d'incendi.
 - **Mòdul de moviment**, que s'ocupa de vigilar si hi ha algú en una zona sense persones.
 - **Mòdul d'IR**, encarregat també de la seguretat d'un estai protegit, que aquest cas controla un pas per línia IR.
 - **Mòdul d'ultrasons**, que té la funció del control de l'accés a un punt concret localitzat.
- **Sistema d'avís d'alarmes via mòbil (SMS)** a l'usuari, per mitjà de la tecnologia GSM.
- **Sistema de seguretat per PIN** per tal de tenir un control contra intrusos de les funcions del sistema i de la seva configuració per mitjà del teclat.
- **Comandament a distància (IR)** per al control del menú (objectiu extraordinari).

Els objectius extraordinaris que han quedat reservats per a una futura actualització són:

- Dotar el sistema d'un sensor d'aigua per afegir la funcionalitat d'alarma d'inundació.
- Control bidireccional entre el sistema i l'usuari, via SMS. Una possible funcionalitat podrà ser la d'engegar i apagar el sistema a través d'un SMS o una trucada per part de l'usuari.

1.4 Enfocament i mètode seguit

L'enfocament del projecte ha estat des dels seus començaments una idea basada en l'aprenentatge i estudi de la tecnologia *Arduino*. Per poder dur a terme el mateix, s'ha intentat seguir en la totalitat del procés tots els recursos tals com: llibreries, exemples, codis o idees existents dins de la plataforma arduino.cc/playground [1].

Atès que el sistema encastat consta d'aspectes que han resultat bastant dificultosos, com ara l'enviament de variables mitjançant el serial, conversions de caràcters a cadenes, comunicació, la pàgina oficial d'*Arduino* ha resultat de vital importància per a dur a terme el projecte.

S'emmagatzema un repositori per consultar el funcionament de les fonts disponibles, llibreries, i exemples de desenvolupament. Ha estat un pilar fonamental per a la construcció del projecte ja que està dotada d'una font d'informació molt gran per a la plataforma.

La metodologia de desenvolupament i implementació ha estat durant tot el procés un esquema basat en etapes amb el seguiment del diagrama de Gantt desenvolupat en planificació del projecte.

Per a la constitució de la memòria ha resultat de gran utilitat fer un recull de totes les fonts utilitzades durant el procés de construcció i desenvolupament del TFC.

No s'ha contemplat el desenvolupament de classes o llibreries específiques ja que el codi existent en cada funció no ho requeria.

1.5 Planificació del treball

En aquest apartat es detallen breument les tasques necessàries que s'han hagut de realitzar per completar el prototip *Alarmino*. Les fases de la planificació i el seu detall es mostren de forma cronològica:

Fase 1

- Dissenyar i implementar Menú Principal (HMI): Recopilar informació respecte la *Shield LCD Keypad* i codificar un menú bàsic amb dues opcions:
 - 1. Mode IN: Aquesta opció activarà un sensor i posarà l'alarma en funcionament.
 - 3. Desactivar: Aquesta opció desactiva l'alarma (apaga el Mode IN).
- Estudi de les diferents llibreries existents sobre el mòdul de ràdio NRF4201 i codificació del sistema de comunicació entre dos o més *Arduino*.
- Implementar el codi necessari per interpretar les dades generades per un sensor (moviment) gestionat per un *Arduino*.

Fase 2

- Implementar el codi necessari per interpretar les dades generades per la resta del sensors (ultrasons, infrarojos, temperatura).
- Realitzar el muntatge dels *Arduino Nano* (amb el mòduls de ràdio instal·lats en la Fase 1) amb un sensor per cada un.
- Modificació del Menú Principal: Afegir una nova opció al menú bàsic (desenvolupat a la Fase 1)
 - 2. Mode OUT: Aquesta opció activa tots els sensors disponibles i posa l'alarma en funcionament.
- Codificar el sistema de missatges entre la *Shield GSM* i el telèfon mòbil de l'usuari final.
- Codificar el sistema de comunicació per ràdio entre la *Shield GSM* i *Arduino* per gestionar les diferents incidències rebudes respecte els sensors.

Fase 3

- Investigar i estudiar les diferents llibreries de la *Shield GSM* per tal de realitzar trucades i enviar SMS.
- Implementar el codi necessari per interpretar les dades generades pel sensor d'humitat.
- Afegir sol·licitud de clau de seguretat a l'inici del sistema (necessària per a accedir al menú principal), sense aquesta clau de seguretat no es podrà accedir a l'aplicació.
- Afegir nova opció (4. Canvi de clau) pel control de la clau de seguretat de l'alarma. Aquesta opció permetrà canviar la clau de seguretat del sistema. Es demanarà la clau antiga i un cop validada (3 intents), demanarà introduir la nova clau.

- Codificar les ordres necessàries per tal d'emmagatzemar la clau de seguretat de l'aplicació a la memòria EEPROM d'Arduino.
- Codificar el control del menú per comandament a distància (IR).

En el següent diagrama, es realitza la representació temporal de les tasques exposades en l'apartat

anterior. Com podem observar, algunes de les tasques són solapades ja que poden ser executades a

mateix temps, la qual cosa genera un aprofitament del temps. De la mateixa manera algunes de les tasques

necessiten de treballs o tasques prèvies per ser iniciades.

Les dates i fites estimats en la programació inicial han tingut un concurrència temporal correcta.

No s'han detectat grans retards en cap de les parts.

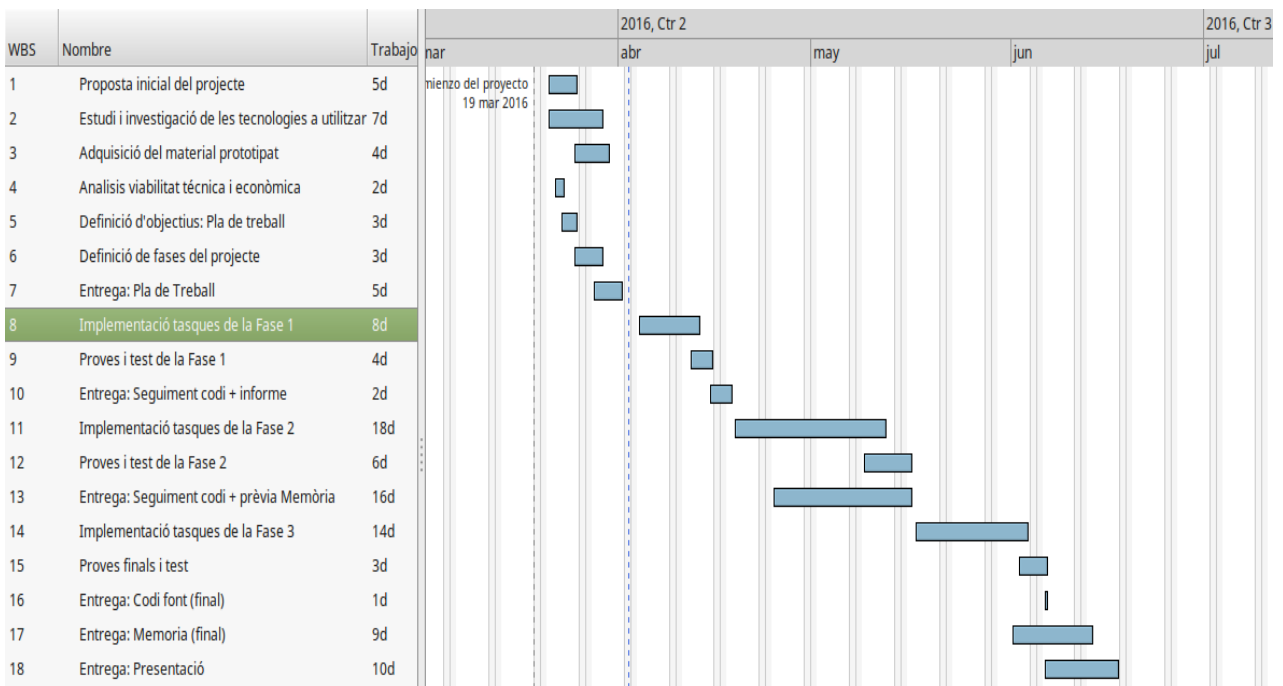


Figura 1. Diagrama de Gantt

1.6 Recursos emprats

Per a la realització del prototip s'han emprat els següents recursos:

- Sèrie de micro-controladors, el prototip requereix d'un *Arduino Mega 2560 REV3*, un *Arduino UNO REV3* i quatre *Arduino Nano*.
- Quatre mòduls de ràdio NRF2401 per la comunicació sense fils entre els diferents mòduls.
- Sensors d'humitat, de temperatura, d'ultrasons, de moviment i infrarojos.
- Cabejat, resistències, protoboard per tal de fer la connexió de tot el conjunt.
- Ordinador portàtil *Toshiba Satellite A300*, necessari per programar i carregar el codi font als micro-controladors.
- Entorn de programació *IDE Arduino 1.6.8* (llenguatge Wiring & Processing), el qual ha permès desenvolupar, comprovar, compilar i carregar els programes o Sketch als equips *Arduino* mitjançant el port USB. Així com la comunicació serial i accés a la configuració directa del mòdul LCD i al mòdul GSM.
- Llibreries necessàries per a controlar alguns dels sensors, les shield LCD i GSM són les següents:
 - Time, tractament integral de dates i hores en formats clàssics i unix.
 - EEPROM, la qual ens dona accés a una memòria ROM programable i esborrable elèctricament.
 - LiquidCrystal, la qual dota la plataforma d'una facilitat per a poder realitzar funcions d'escriptura, esborrat de la pantalla, encesa i totes les funcionalitats necessàries d'una manera molt intuïtiva.
 - DHT11, la qual ens facilita el control i gestió del sensor de temperatura que porta el mateix nom.
 - IRremote, que ens permet establir comunicació entre el comandament a distància i l'aplicació.
 - RF24, encarregada de gestionar els mòduls de radiofreqüència NRF2401.

<p>Arduino UNO, Mega i Nano</p> 	<p>Shield LCD Keypad 16x2 i modul de ràdio NRF2401</p> 
<p>Regulador d'alimentació i Shield GSM/GPRS</p> 	<p>Sensors utilitzats; ultrasons, PIR, infrarojos, temperatura i mòdul de ràdio NRF2401</p> 
<p>Components electrònics variis; Rellotge RTC, relé i Protoboard</p> 	<p>Control remot IR i alimentació 9V</p> 

Figura 2. Maquinari emprat pel prototip

Tots els equips estan muntats en protoboards amb un regulador de corrent que lliura 5 volts [3] i/o 3,3 volts per alimentar tant a l'Arduino com als sensors per evitar sobrecàrregues de consum.

Per alimentar el regulador s'han utilitzat bateries de 9V, que sense ser la millor opció possible, són suficients i són el més còmode.

L'excepció és el mòdul GSM/GPRS que necessita una major intensitat i està alimentat per una bateria de 6 piles de 1,5Volts que encara que també donen una tensió de 9Volts com les altres, lliuren 900mA contra els 400mA que lliura la pila de 9Volts .

1.7 Productes obtinguts

El Producte obtingut és un prototip d'alarma autònom, econòmic i amb potencial per a créixer en el futur donada la facilitat d'adaptació de components que brinda el maquinari lliure.

2. Antecedents

En els dos capítols següents es realitza una introducció a la tecnologia *Arduino* en l'àmbit artístic caracteritzat com una potent eina de desenvolupament cultural, instructiu i formatiu. També es realitzarà una valoració de l'estudi de mercat de la tecnologia utilitzada, així com un breu comentari sobre les diferents tecnologies existents.

2.1 Estat de l'art

Arduino és una eina utilitzada per artistes audiovisuals i electrònics (Art mitjana). Existeixen multitud de projectes entorn a aquesta plataforma des de satèl·lits, robots, equips audiovisuals, instruments musicals controlats a través d'*Arduino*. Atès que és una plataforma econòmicament assequible, el programari i maquinari lliures permeten als artistes tenir una eina de treball de baix preu (10 euros aproximadament), la qual compta amb fonts d'informació en comunitats destinades al desenvolupament. Nombroses instal·lacions artístiques estan dotades avui d'aquesta tecnologia, remarcant peces que permeten als visitants interactuar o intervenir en el funcionament de les obres.

També s'ha de destacar que *Arduino* està basat en un llenguatge de programació anomenat Processing / Wiring, derivat del llenguatge C, resultant així que l'aprenentatge no requereixi de coneixements d'enginyeria industrial o electrònica per a la programació. Totes aquestes

característiques fan que aquesta plataforma estigui sent utilitzada en centres d'art per a la realització de workshops d'electrònica i instal·lacions artístiques.

L'ensenyament és un camp que s'està proveint de les característiques del dispositiu, centres d'educació secundària compten amb plaques *Arduino* per instruir els alumnes en introducció a l'electrònica.

Existeixen en el mercat multitud d'alternatives avui en dia a *Arduino* sobretot després de l'aparició d'aquest, van des de la construcció pròpia d'una plataforma casolana amb un PIC

AVR fins a sistemes com Pingüí el qual compta amb PIC de microxip. PowerJaguar també és una plataforma de aprenentatge que com Pingüí aquesta composta de PIC de Microchip en lloc de AVR. Encara existeixen diverses, aquestes dues últimes són pràcticament clons d'*Arduino* pel que fa a disseny o estructura es refereix. Són utilitzades de manera habitual per Artistes ja que poden resultar fins i tot més barates que la plataforma escollida per a la realització d'aquest projecte, o com no la plataforma utilitzada a UOC TinyOS la qual ja és avalada per una llarga trajectòria de projectes.

2.2 Estudi de mercat

El dubte més important a l'hora decidir la plataforma del TFC per abordar el projecte ha estat la versatilitat que brinda *Arduino*. Aquest micro-controlador va ser creat per dues persones, destacant a David Cuartielles.

En un primer moment *Arduino* era una placa amb un xip Atmel el qual no comptava amb connexió USB, però que ja tenia entrades i sortides analògiques i digitals. Posteriorment se li va dotar de connectivitat USB amb un xip FDDI, i en l'actualitat es poden trobar equips que compten amb xips de ràdio Xbee integrats en el pròpia placa, com és el cas de *Arduino Nano*.

Una altra característica fonamental a l'hora de valorar la presa de decisió d'escollir la plataforma de treball ha estat la consolidació d'aquesta en el mercat. Pingüino o PowerJaguar són alternatives que estan resultant d'un potencial creixent però que encara no tenen un calat suficient en les comunitats de desenvolupament. Ambdues són maquinari i programari lliure però les seves fonts de consulta no poden ser comparades a les fonts existents per *Arduino* així com totes les llibreries o accessoris de mercat.

Podria resultar avantatjós utilitzar les dues plataformes citades en el cas de reaprofitar codi escrit per PICS els quals podrien ser utilitzats si d'aquesta elecció es tractés.

3. Descripció funcional

L'aplicació *Alarmino* ha evolucionat des d'un disseny anterior mixt entre bus i estrella (Tots connectats al central, o individualment o a través d'un bus I2C) cap al disseny actual sense fil i comunicat completament per radiofreqüència (RF).

Els principals motius per canviar el disseny primitiu de cablejat per l'actual comunicat amb radiofreqüència en la seva totalitat han estat:

En primer lloc l'enorme dificultat que presenta un cablejat eficient i pràctic en un pis o casa no pensat en principi per això.

En segon lloc l'autonomia modular i la flexibilitat que representa la comunicació sense fils a l'hora de canviar de lloc un sensor o d'afegir/suprimir sensors.

S'han assajat dos tipus diferents de mòduls de RF. Els primers utilitzats van ser els anomenats 433Mhz que són molt senzills, tenen un mòdul emissor i un altre receptor i com el seu nom indica treballen a 433 megahertz. Més tard es va prendre la decisió d'utilitzar els transceptors NRF24L01 [1] que treballen a 2,4GH i tenen un major abast i que a més de ser més moderns, tenen desenvolupades diverses llibreries d'ajuda a la programació [2].

La connexió física és ara per tant, independent de la distribució geogràfica. L'abast dels mòduls NRF24L01 + PA + LNA 2.4 GHz amb Socket adapter (petita shield que inclou els elements electrònics necessaris per a un bon funcionament) i amb antena incorporada, és molt superior a les necessitats d'un habitatge mitjà i fins i tot gran.

S'ha provat fins a una distància aproximada de 30 o 40 metres (La màxima que s'ha pogut, tot el pis més un petit jardí i un carrer) i s'ha comunicat correctament en diverses freqüències diferents (des del canal 1 fins al canal 108 per sobre de la freqüència de connexió internet) sense diferència apreciable.

Finalment s'ha utilitzat el canal 106 per entendre que es tracta d'una freqüència cap al final de la utilitzada per WiFi i que no ha donat cap problema en les proves.

Descripció dels blocs funcionals

- Bloc de procés i control: És el bloc compost per *Arduino MEGA + Shield LCD Keypad + Mòdul Radio NRF2401*. Rep la informació dels sensors que l'envien els blocs independents del sistema, mitjançant ones de ràdio. D'altra banda aquest bloc s'encarrega d'avaluar la incidència detectada i basat en el mode de procés actual, decidir si cal comunicar-se amb el bloc de comunicacions *Arduino UNO + Shield GSM* per tal de enviar el missatge SMS corresponent al telèfon de l'usuari o usuaris configurats en el sistema.

(Es pot consultar l'esquema elèctric a l'annex '9.2. Esquema elèctric bloc control')

- Bloc de comunicació: És el bloc compost per l'*Arduino UNO + Shield GSM*. Compleix la funció de generar i enviar els avisos d'alerta per SMS que li demana el *Arduino MEGA* del bloc de procés i control. Les alertes són enviades a l'usuari a través del telèfon mòbil. Aquest mòdul es comunica amb el mòdul principal per RF (NRF24L01).

(Es pot consultar l'esquema elèctric a l'annex '9.3. Esquema elèctric bloc comunicació')

- Bloc de detecció: És el conjunt de totes les unitats *Arduino Nano + mòdul ràdio NRF2401*. Cadascuna de aquestes unitats pot controlar un o mes sensors ubicats al mateix lloc. Aquesta independència dels sensors permet una gran mobilitat geogràfica. Cada *Arduino Nano* albergarà, a part de la gestió d'un o mes sensors, la gestió d'un mòdul de ràdio NRF2401, necessari en cada cas per a comunicar-se amb el mòdul principal.

(Es pot consultar l'esquema elèctric a l'annex '9.4. Esquema elèctric bloc detecció')

3.1 Alarmino: Alarma per la llar

a) Diagrama de blocs de l'aplicació

El sistema es compost per tres blocs funcionals:

1. Bloc de procés y control : Arduino Mega + LCD Keypad + NRF24L01
2. Bloc de comunicació : Arduino UNO + Mòdul GSM + NRF24L01
3. Bloc de detecció (4 unitats) : Arduino NANO + Sensor associat + NRF24L01

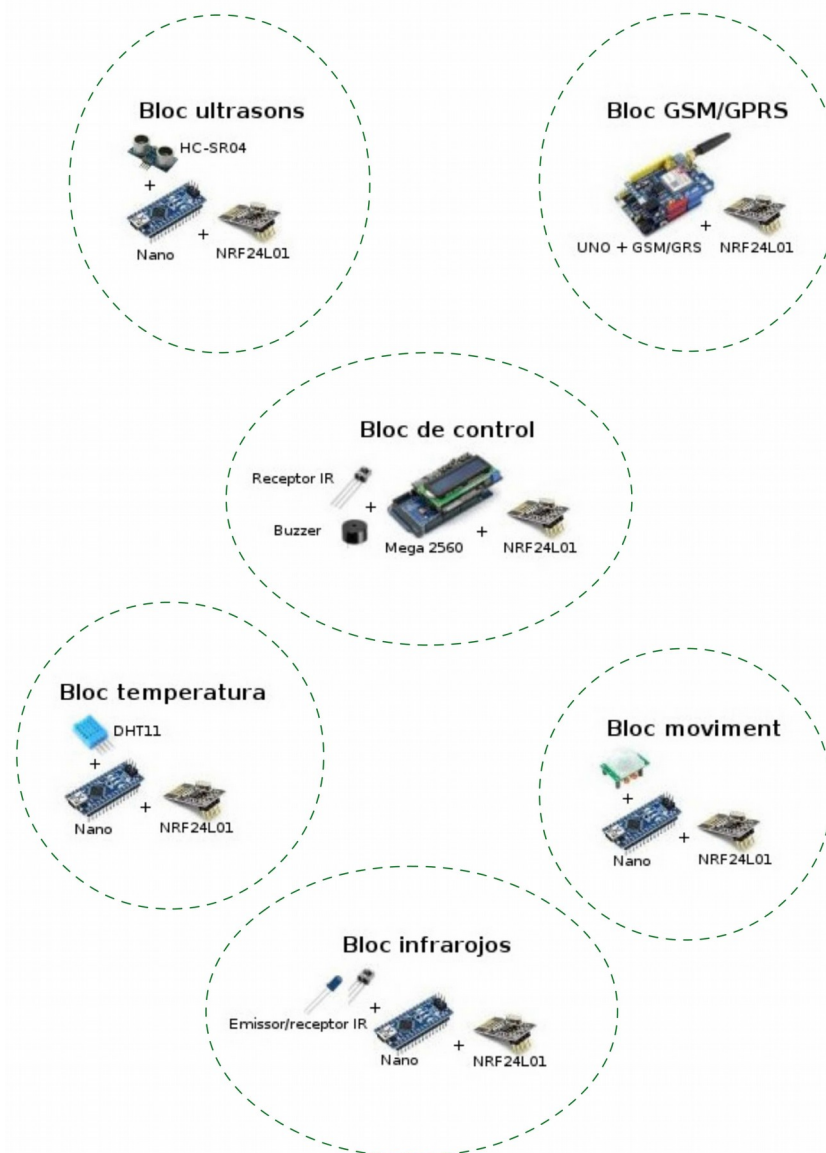


Figura 3. Diagrama de blocs

El comandament a distància IR i el Mòbil es poden considerar elements externs al sistema, malgrat que poden interactuar programant el mode de procés i rebent informació respectivament, per tant no s'han inclòs al diagrama.

b) Xarxa de comunicacions

La xarxa actual, amb comunicacions per RF, físicament és una topologia de xarxa cel·lular [5] ja que tots els nodes estan a l'abast de tots els altres, cosa interessant a tenir en compte per a futura evolució del sistema.

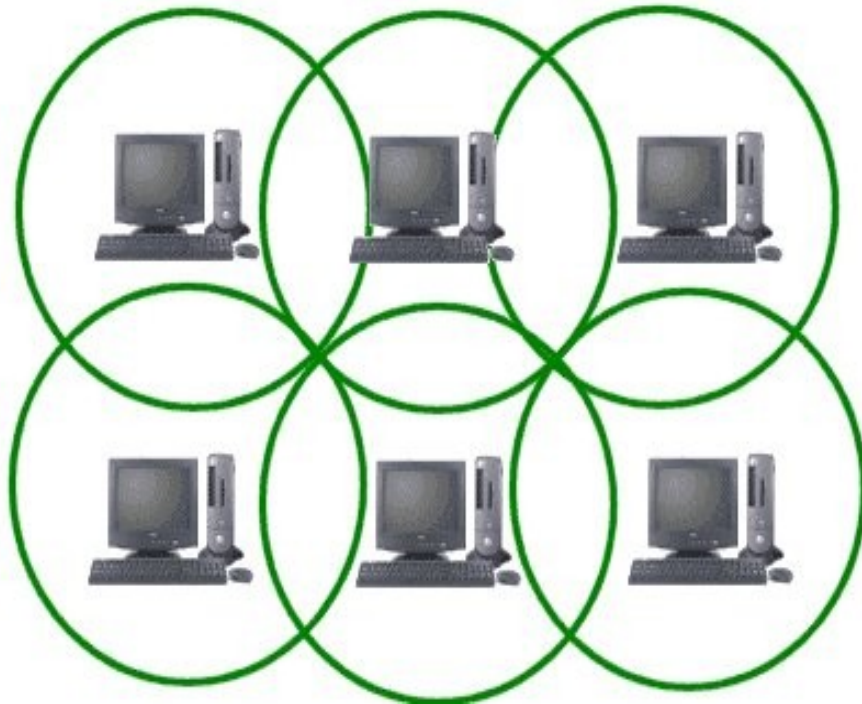


Figura 4. Topologia de xarxa cel·lular

No obstant això, a causa de la lògica de les comunicacions establertes per l'aplicació el comportament general és com si haguessin dues xarxes en estrella [6] diferents, una d'elles penjant de l'altra.

1. Una xarxa en estrella però amb tractament per torns. Hi ha un node central (control) donant torn (polling) a uns nodes "sensor" per recollir informació un per un i serialitzant així la comunicació.

Aquests nodes no es comuniquen entre ells. Tots escolten per un "pipe" [1] (identificador usat per NRF24L01, que suporta fins a sis diferents) i transmeten per un altre diferent. "Control" utilitza els mateixos "pipes" al revés.

2. Una xarxa en estrella monodireccional. En aquesta no hi ha conversa sinó només peticions dels sensors al node central en una sola direcció.

Consta d'un node central ("comunicació") rebent peticions de missatgeria directament d'altres nodes "sensor" que treballen per rutines d'interrupció i que no es comuniquen entre ells.

* Els tipus d'Interrupcions que suporten els diferents models *Arduino* son els indicats en la tabla següent.

Model Arduino	Int 0	Int 1	Int 2	Int 0	Int 0	Int 0
Mega 2560	Pin 2	Pin 3	Pin 21	Pin 20	Pin 19	Pin 18
UNO	Pin 2	Pin 3	-	-	-	-
Nano	Pin 2	Pin 3	-	-	-	-

Figura 5. Taula de pins d'interrupció d'Arduino

Aquests sensors emeten pel mateix "pipe" [3] que el programa "control" per evitar al GSM/GPRS haver d'escoltar per dos canals diferents.

Existeix a més una connexió lògica dels dos nodes ("control" i "comunicació") fruit de les peticions de missatgeria del mòdul "control" que centralitza totes les dels sensors tractats i en cas de petició d'alarma, els envia al mòdul " comunicació "per al seu processament.

Aquesta comunicació és també unidireccional de "control" a "comunicació" i pel mateix "pipe" que utilitza "control" per al "polling" amb els sensors sense interrupció. L'efecte resultant per "comunicació" és com si "control" i tots els seus sensors de "polling" fossin un més dels seus nodes peticionaris en estrella.

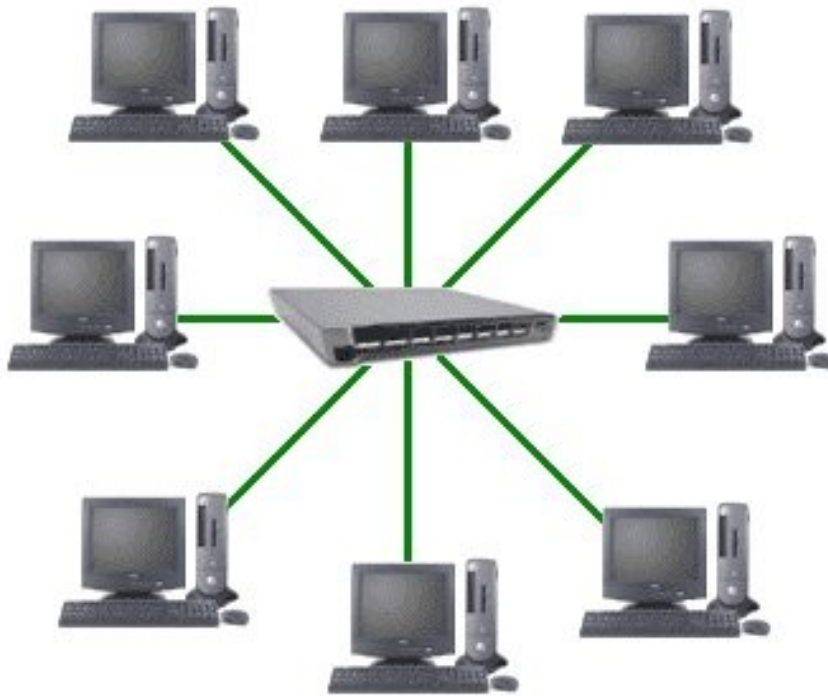


Figura 6. Topologia de xarxa cel·lular

c) Com interactuen els diferents objectes en el sistema

A partir de la topologia lògica detallada en l'apartat anterior, dues estructures en estrella i una d'elles actuant com un node més de l'altra gairebé es pot deduir que hi ha un estricte protocol de comunicació. És el següent:

1. El node "control" envia a l'arrencar i cada canvi de minut un registre de sincronisme amb l'hora exacta en format UNIX i el menú / submenú seleccionat. Tots els nodes tracten aquests registres per sincronitzar el rellotge (la primera vegada que tracten un registre d'aquest tipus) i conèixer la manera de treball del sistema (menú / submenú).
2. El node "control" envia peticions d'informació als nodes "sensor", per rigorós torn, aquests informen les dades sol·licitades i re-envien el registre, que és llegit per "control" i pren una decisió de procés en funció de la informació rebuda en la resposta.

3. En el cas d'haver-hi una resposta que arribi als valors previstos per a donar una alarma, "control" emetrà un registre amb que haurà de ser tractat per "comunicació" qui al seu torn emetrà l'alarma en forma de SMS o trucada perduda als cel·lulars previstos.

En tots els registres de emesos per "control" s'inclou el menú / submenú actiu a causa que aquest pot ser canviat dinàmicament.

4. Els nodes "sensor" que no estan en el "polling" de "control", és a dir que funcionen amb rutines d'interrupció via "attach", envien a "comunicació" la petició d'alarma sense mes.
5. El node "comunicació" gestiona les peticions de alarma. Per cada missatge d'una alarma particular, "comunicació" enviarà la 1^a vegada un SMS. la 2a i 3a vegada una trucada perduda per cridar l'atenció i a partir de la tercera petició idèntica la ignorarà.

3.2 Alarmino HMI

Atès que la filosofia seguida durant el desenvolupament del projecte consisteix a crear un sistema no vinculat a un PC no es disposa de programari o aplicació funcional en un ordinador. La interfície d'usuari desenvolupada està relacionada amb el control de l'aplicació per part de l'usuari.

Les opcions i possibilitats que ofereix el menú de l'aplicació mostrat per pantalla són les següents:

- **Inici** de l'aplicació mitjançant la introducció de contrasenya de 3 números.
- **Opció 1** anomenada '*Alarmino* OUT' i que és l'encarregada d'activar el mode d'alarma total, és a dir que s'activen tots els sensors. Aquesta opció està pensada per a ser activada quan no hi ha ningú a dins de la vivenda. Per activar aquest mode es sol·licita l'entrada de la clau de seguretat per part de l'usuari.
- **Opció 2** anomenada '*Alarmino* IN' i que és l'encarregada d'activar el mode d'alarma parcial, és a dir que s'activen només una part dels sensors. Aquesta opció està pensada per a ser activada quan l'usuari és a dins de la vivenda, per exemple mentre

dorm. Els sensors activats són tots excepte el sensor d'ultrasons. Per activar aquest mode es sol·licita l'entrada de la clau de seguretat per part de l'usuari.

- **Opció 3** anomenada 'Desactivar' i que és l'encarregada de desactivar el mode d'alarma actiu. Per executar aquesta acció cal introduir la clau de seguretat.
- **Opció 4** anomenada 'Canvi clau' i que és l'encarregada de canviar la clau de seguretat de l'aplicació.

4. Descripció detallada

El punt 3.1 apartat c) detalla suficientment el flux de missatges entre nodes, però no deixa clara l'estructura que permet això.

Pel que fa a l'estructura del registre per enviar i rebre els missatges, s'ha recorregut al disseny d'un registre de longitud fixa i menor o igual a 32 Bytes, que conté tota la informació necessària per a cada cas i suficient en tots els casos contemplats de l'aplicació.

Estructura de dades per a comunicar per RF entre Arduinos (tamany <= 32 bytes)

Format i nom	Comentari	Longitud	Acumulat
byte cap	Marca inici missatge	1 byte	1 byte
byte sensor	Codi numéric	1 byte	2 bytes
byte serie	Nº de seqüència sensor	1 byte	3 bytes
float sensor1	Sensor DHT11 temperatura	4 byte	7 bytes
float sensor2	Sensor SC-HR04 ultrasons	4 byte	11 bytes

float sensor3	Sensor IR infrarojos	4 byte	15 bytes
float sensor4	Sensor PIR moviment	2 byte	17 bytes
byte sensor5	Sensor tall corrent	1 byte	18 bytes
Byte interrupcio	Tipus d'interrupció	1 byte	19 bytes
byte alarma	Codi d'alarma	1 byte	20 bytes
unsigned long epoch	Hora UNIX	4 byte	24 bytes
byte periferic	Arduino que respon o emet	1 byte	25 bytes
byte menuactivo	Menú actiu	1 byte	26 bytes
byte submenuactivo	Submenu actiu	1 byte	27 bytes
byte reservado1	Reservat us futur 1	1 byte	28 bytes
byte reservado2	Reservat us futur 2	1 byte	29 bytes
byte cua	Marca fi del missatge	1 byte	30 bytes

Figura 7. Estructura paquet de comunicacions

La mida del registre ha estat especialment cuidat codificant els nodes (perifèrics) i sensors i utilitzant els camps necessaris exclusivament per evitar que la mida excedís els 32 Bytes que és la mida màxima d'un bloc de transmissió (payload) utilitzat pels mòduls NRF24L01.

Naturalment és possible utilitzar missatges de més d'un bloc i però s'ha de desenvolupar un protocol propi per rebre aquests missatges en trossos de 32 Bytes o menors.

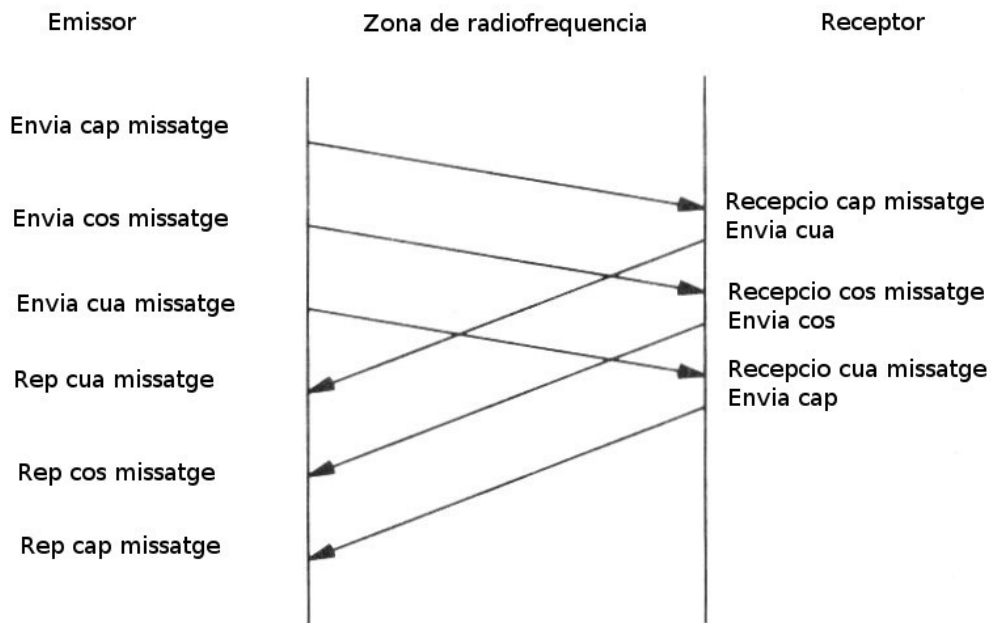


Figura 8. Flux de comunicacions

Resum de l'ordenació de RF

Registres Emesos per "modulControl":

- Registres de format fix amb identificació del "sensor" i "sèrie" del sensor al qual van dirigits.
- Registres amb "sensor = 0" per sincronitzar tots els Arduinos. El tracten tots els "sensorXX_n" i també "modulGPRS".
- Registres de petició d'alarma per sensors sense interrupcions porten un codi "111" al camp "interrupció" per indicar que són per a "modeloGPRS" i els altres ho han d'ignorar.

Registres Emesos per "sensorXX_n":

- Sempre són registres de petició d'alarma per interrupcions porten un codi "222" al camp "interrupció" per indicar que són per a "modeloGPRS" i els altres ho han d'ignorar.

* Ni els registres de sincronisme ni els d'alarma (111 i 222) es contesten. Només s'executen.

4.1 Hardware del sistema

El maquinari d'*Alarmino* és relativament senzill.

El Bloc de control i procés està format actualment per un Arduino MEGA amb una pantalla LCD i un sensor IR per al comandament a distància. Podria ser gairebé qualsevol Arduino, però per a futures funcions, (un servidor web per exemple) convé posar un Arduino potent.

El Bloc de Comunicació consta d'un Arduino UN amb una shield GSM / GPRS que incorpora un xip SIM900 i una targeta SIM de telefonia.

El Bloc Sensor està format per diversos (actualment 6) petits Arduinos NANO, cada un amb el seu sensor particular.

Tots els equips incorporen un mòdul transceptor de ràdio NRF24L01 + PA + LNA 2.4 GHz amb Socket adapter i antena incorporada.

Tots els equips estan muntats en unes protoboards amb un regulador de corrent que lliura 5Volts i / o 3,3Volts per alimentar tant a l'Arduino com als sensors i Shields evitant així sobrecarregar el Arduino.

Per alimentar el regulador s'han utilitzat bateries de 9V, que sense ser la millor opció possible, són suficients i són el més còmode.

L'excepció és el mòdul "comunicació" que necessita una major intensitat i està alimentat per una bateria de 6 piles alcalines de 1,5Volts que encara que també donen una tensió de 9Volts, lliuren entre 900 i 1.200 mA contra els 400mA que lliura la pila de 9Volts.

En les proves realitzades, utilitzant piles de 9Volts 400mA sol funcionar però hem detectat que algunes vegades el GSM no aconsegueix donar-se d'alta a la xarxa de telefonia, mentre que amb les 6 piles alcalines de 1,5 V en bateria (9Volts 900mA) no es ha detectat ni un sol error.

4.2 Software del sistema

El Programari del sistema ha estat desenvolupat amb la *IDE Arduino 1.6.8* i amb el suport dels tutorials que hi ha en quantitat a Internet, encara que són de destacar per la seva concreció i acurada presentació els de la sèrie de tutorials de prometec [7].

L'aplicació *Alarmino* s'ha distribuït en tres grans blocs que es corresponen amb els tres blocs de programació.

Així el programa (sketch) "control" es correspon amb el bloc físic "control", el "comunicació" amb el bloc "comunicació" i el tercer programa "modulSensor" és realment un programa model que pot tractar qualsevol dels sensors instal·lats i que s'hagi de parametritzar perquè tracti a un d'ells en particular.

Aquest disseny té l'avantatge de simplificar el manteniment de la programació ja que s'evita la proliferació de molts programes la estructura és similar però amb tractaments diferent de vegades molt similars i de vegades molt diversos, el que porta sovint a un caos de versions.

Els programes que s'executaran els sensors no s'han de modificar mai, sinó només el "modulSensor" que és el que l'ha generat.

Crec que no hem d'entrar aquí en un disseny orgànic que està ja inclòs en les capçaleres dels programes que es van lliurar (codi).

A la depuració i optimització dels programes ha estat de bona ajuda un article sobre això que recollim a [8].

4.3 Programació

La programació sempre es un punt clau en tot projecte informàtic.

Malauradament no hi ha una programació estàndar i automàtica per fer la mateixa cosa, la qual cosa porta a cometre errors que s'han de localitzar i depurar per a poder dir que un programa és acabat.

Des d'altre punt de vista, aquesta flexibilitat de poder fer la mateixa cosa de mes de una manera es la que dona gracia i versatilitat als llenguatjes de programació.

La programació del projecte *Alarmino* no es una excepció i crec que val a dir que en un TFC d'informàtica de gestió com aquest es mes rellevant que no pas la part el·lectrónica, que finalment es un suport físic per a la nostra tasca que es el Software.

El Software desenvolupat per a *Alarmino* ja ha estat comentat en apartats anteriors des del punt de vista funcional però cal també destacar algunas particularitats de lo que es propiament programació.

El projecte *Alarmino* ha estat programat pensant no només en la aplicació actual sino també en la facilitat de manteniment i creixement futur.

Los tres programes desenvolupats están escrits per ser entesos per las personas i no només per las máquinas. Es a dir, els programes no son compostats per llargues rutinas que ho fan tot en seqüencia sino que la estructura principal es simple degut a l'us de rutinas mes simples (divide et impera) todas les rutines están força documentades a dintre mateix del codi del programa per tal que las personas que tinguin que modificar una rutina, afeixir codi, o simplement llegir el programa per comprendre lo que fa, lo puguin fer facilment.

En el cas del programa "modeloSensor" (el mes original de tots tres) la novetat consisteix en que no és realment un programa sino que mes aviat podem dir que es on proto-programa ja que el programa es pot compilar per comprovar que no hi ha errors de sintaxis però no es pot executar tal i com está ja que les diferents rutines dedicades a los diferents sensors que tenim (i el mateix pasará si n'afeixim d'altres) fan servir els mateixos recursos dels Arduinos NANO que incorporarán els diferents sensors.

L'única desavantatge es el creixement del programa, però com ja hem asenyalat la modularització i els comentaris insertats amb el codi neutralitzen aquest efecte, i en canvi l'aventatge que presenta la concentració del manteniment ho compensa sobradament.

Recordem que una copia d'aquest programa parametrizada (amb només quatre punts) per tal de seleccionar el sensor concret a tractar, genera un programa per a un perifèric concret (aquest programa si que s'executa).

Un altre punt interessant per senyalar es, al meu criteri, que degut a la naturalesa dels sensors que hem anomenat d'interrupció, (v.g. una línia d'infrarojos tallada o una incidència puntual en un sensor d'ultrasons o de moviment, pot no ser detectada si el codi utilitza la instrucció "delay()" que fa "dormir" l'Arduino) per tant en les rutines que tracten aquest tipus de sensors no se pot fer servir aquesta instrucció [12].

Degut aquest efecte de "dormir" l'Arduino que te la instrucció delay(), per si hi ha necessitat d'esperar un temps, hem desenvolupar una rutina que es cridada "espera(n)" i que a efectes de esperar n milisegons fa el mateix que la instrucció delay(n) sense "dormir l'Arduino" de tal manera que la rutina del "attach interrupt" pot prendre control, que en un delay(n) no pot.

4.4 Problemes

No hi ha programació que no presenti cap problema i si existeix tal cosa no es divertit programarla.

Per suposat la programació d'*Alarmino* ha presentat problemes diversos, però son coses que el programador he de tenir sempre en compe. Més encara *quan* es tracta d'un llenguatge de programació en el que no tenim gaire experiència com ara el Processing.

Son coses asumibles i ja esperades en un desenvolupament de software qualsevol.

En aquest projecte hi ha agut sobre tot un problema que ha estat a punt de espatllar'ho tot degut a que no era facil de aïllar i per altra banda no era sempre igual. Estem parlant dels problemes amb les transmissions / recepcions per radio amb els mòduls NRF24L01.

Una vegada probats amb els exemples que porten les llibreries i alguns altres trobats als forus d'internet va començar la festa. Primerament no funcionava correctament emetent un missatge de text char*, la recepció era un buffer plé de caràcters estranys.

A continuació fallava al intentar emetre una estructura de dades formada por diversos camps. Tot aixó no era mes que el començament, un anticip del patiment que vindria mes tard I va anar millorant segons el coneixament de les característiques de mòdul s'incrementava.

Finalment i amb el ajut del documents d'internet [9] i d'altres semblava que tot estava ben programat i va començar la prova amb dades en estructura i treient les dades per el Serial.bus per tal de veure el resultat de les emissions i lo que es rebia a l'altra banda. *quan* ja el funcionament semblava satisfactori i despres de afeixir algunes instruccions que en teoria no tenian cap relació amb la ràdio, va deixar de funcionar (tot donave Timeout i no es rebia el ack *quan* s'emetia un missatge, i lo mes greu era que a vegades no donava ack però el missatge Si s'emetia correctament perquè l'altre conversador lo rebia. Aquest problema va resultar un bug de les llibreries utilitzades [10] però va estar a punt de invalidar'ho tot i sense temps material de canviar de mòduls o cap altre solució.

Finalment probant llibreries amb noms similars però provinents de diferents llocs (i de tamany també bastant diferents, que ja demostra que no son ni de lluny iguals) es va trobar la que estem fer servir que només falla molt de tant en tant. S'ha de dir que aquestes llibreries ja estan fora de manteniment i estan substituïdes per una de nova RadioHead [12] que engloba el support als mòduls de ràdio en general i als NRF24L01 en particular.

El estat actual es estable ja que tenim els errors controlats i no es el moment de canvis importants, però en un futur está fortament recomenat que la llibreria RadioHead substitueixi a la nrf24 actual. Així ho farem.

L'exit obtingut finalment amb aquesta eina tan poderosa i delicada com ràpida i difícil de monitoritzar (Tenir en conta que la monitorització per si mateixa pot desvirtuar i fins i tot invalidar fàcilment els resultats obtinguts) amb una cosa tan veloç que uns microsegons de mes o de menys tornan un ack/nack en Timeout, no perquè el missatge estigui be o malament o no hagi arribat sino perquè hem arribat tard a la medicació. Una vegada entés aixó tot es mes facil.

5. Viabilitat tècnica

El desenvolupament del projecte ha donat lloc a un equip amb alt nivell d'adaptabilitat, la capacitat que *Arduino* ofereix a l'hora de reconduir els projectes brinda la possibilitat al sistema de ser utilitzat no només per a la funció esdevinguda si no per a múltiples problemàtiques.

Aquesta justificació corrobora que la inversió en el maquinari pugui ser més elevada que la d'un equip dissenyat només per treballar en mode d'alarma per a la llar però a diferència d'aquest cas, el node dissenyat no té una funció tancada. Aquesta premissa es compleix també a nivell de software ja que el disseny de la comunicació entre els equips permet que es pugui variar amb certa facilitat la tasca dels nodes.

Per això és possible afirmar que la viabilitat del projecte es pot considerar com a positiva quan el producte requerit necessiti tenir una capacitat d'adaptació, millora o canvi de funció.

Es pot destacar l'alta informació existent a Internet sobre els equips utilitzats en el desenvolupament del projecte, al contrari que els equips maquinari comercials, els quals compten amb un sistema totalment tancat i que en molt poques ocasions permeten intervenir en el seu funcionament per millorar o adaptar la seva capacitat operativa formalment.

L'enviament de dades fora de la xarxa a través de la targeta GSM/GPRS capacitaria el sistema d'un avantatge de poder per enviar dades remotament d'una manera senzilla. Totes aquestes característiques poden justificar el desenvolupament de nous nodes emissors que facin un sistema complet i que puguin operar en un entorn real, on la seva operativitat conjunta proporcioni més sentit al projecte.

D'altra banda la viabilitat del producte en un entorn comercial requereix d'un alt nivell de proves, on els equips deuen ser sotmesos a esforços tant climàtics, operativitat, funcionament a llarg termini, etc. Deuen complir amb una normativa de qualitat a més de ser un sistema que pugui garantir que poden treballar en condicions de seguretat real.

Per a un treball sotmès a una seguretat real, la senyal d'alarma hauria de ser gestionada per una central receptora d'alarmes, fet que dona un plus de seguretat en estar disponibles 24 hores-365 dies a l'any, sense dependre de la cobertura mòbil o que ens haguem deixat el mòbil a casa.

6. Valoració econòmica

A continuació es realitza una valoració del cost dels materials emprats, així com una valoració estimada de la instal·lació, la mà d'obra i manteniment del sistema per un any.

Cost dels materials del prototip

Descripció	Quantitat	Preu Unitari	Preu Total
Arduino Mega 2560	1	10,51€	10,51€
Arduino Uno R3	1	5,00€	5,00€
Shield GSM/GPRS	1	16,18€	16,18€
LCD Keypad Shield	1	3,22€	3,22€
Arduino Nano	4	3,00€	12,00€
Cables	3	1,63€	3,26€
Protoboard	7	3,45€	24,15€
Sensor Temperatura	1	1,00€	1,00€
Sensor Corrent	1	3,96€	3,96€
Sensor Ultrasons	1	1,29€	1,29€
Sensor Moviment	1	2,00€	2,00€
Emissor/Receptor IR	3	1,40€	4,20€
Buzzer [2]	1	1,00€	1,00€
Mòdul de ràdio NRF24L01	6	2,80€	16,08€
		Total	103,85€

Figura 9. Taula cost material prototip

7. Conclusions

Els objectius del projecte han estat fixats per les particularitats descrites al principi d'aquest document, que era crear un sistema d'alarma autònom basat en *Arduino* i lliure de llicències, tant de Software com de Hardware.

Totes les fites principals han estat complertes durant aquest treball.

Cal destacar que s'ha implementat per fi i després d'alguns intents fracassats per causa de les necessitats del hardware, un sistema GSM/GPR. Aquesta funció permet enviar SMS i fer trucades a telèfons mòbils i per ell moment de desenvolupament actual esdevé la mes espectacular particularitat des treball.

L'aprenentatge i coneixements adquirits han estat els desitjats, s'ha aconseguit introduir-se no només a la plataforma *Arduino* sinó també en tot el seu entorn de desenvolupament com ara: investigació d'articles i recursos en arduino.cc o arduino playground.

Les hores de recerca han estat molt superiors a les de desenvolupament, però s'ha aconseguit una base per seguir creant projectes dins d'aquesta plataforma o millorant aspectes d'aquest projecte a més de la possibilitat de abastar altres de nova índole.

La experiència amb la Ràdio Freqüència ha estat amb diferència la mes dura però també la mes enriquidora de totes tot i que (o potser degut a que) ha estat el mes gran problema a resoldre en tot el desenvolupament del programari.

7.1 Autoavaluació

Naturalment jo no puc ser totalment imparcial tot i que voldria ser-ho, però el que tinc molt clar es que l'esforç efectuat ha estat gran i la dedicació ha estat tot el temps lliure que queda quan, com en el meu cas, he començat a treballar després d'anys de no tenir feina, fora de la meva ciutat i a dos mesos d'entregar aquest treball.

La tria de materials ha sigut el mes fàcil gracies a la versatilitat del maquinari *Arduino*.

El disseny, la programació i també la documentació per formar-se un mateix tant o segurament mes que per documentar el treball, com sempre ha donat bastant treball i algunes situacions desconcertants d'errors en principi incomprensibles però ¡La máquina sempre acaba tenint raó! com tot informàtic ha de tenir sempre present.

Jo personalment em sento satisfet del treball realitzat i de les fites aconseguides i sobre tot de l'esforç realitzat amb il·lusió i ganes de fer-ho bé en un temps tan curt.

7.2 Línies de futur

El disseny actual d'*Alarmino* preveu una ampliació de funcionalitats sobretot basades en la utilització de Radiofreqüència (RF) que allibera de servituds de cablejats enutjosos i complicats.

A partir d'aquí, un relé pot ser activat per un *Arduino NANO* en qualsevol lloc d'una casa i efectuar qualsevol operació desitjada que depengui d'obrir o tancar un circuit elèctric, rebent una ordre per Ràdio Freqüència.

Això porta a pensar en el control remot per internet, que sens dubte és el futur més atractiu del projecte, tot i que passa per resoldre alguns problemes tècnics de la web.

La idea bàsica seria muntar un servidor web local on es publicarien les dades recollides constantment dels sensors instal·lats, sobretot de temperatura i humitat (tot i que també les d'alarmes) que actualment es reben però no s'exploten i es podrien consultar per internet.

Això és clarament possible amb una shield ethernet i una adreça IP fixa (que té poc cost i en qualsevol cas pot simular-se amb una direcció indirecta).

En el següent pas, es podria a més des d'un ordinador remot, interactuar amb el sistema ordenant una acció concreta, només amb marcar una opció o una altra en la pàgina web. per exemple encendre l'aire climatitzat o apagar-lo, obrir una electrovàlvula per a reg etc ...

8. Glossari

Arduino: Plataforma de maquinari lliure basada en un microcontrolador, la qual compta amb un entorn de desenvolupament propi. Desenvolupada per facilitar la creació de projectes electrònics.

Art Mitjana: Art lligat a les noves tecnologies, com ara electrònica, audiovisual, videojocs, etc.

AVR: Microcontrolador de la família Atmel dissenyat per a l'execució del llenguatge C.

GSM: El sistema global per a les comunicacions mòbils (de l'anglès Global System for Mobile communications, i originàriament del francès groupe spécial mobile) és un sistema estàndard, lliure de regalies, de telefonia mòbil digital.

Hardware lliure: Maquinari de tipus obert del qual posa a disposició totes les seves característiques, permetent a més la seva modificació amb total llibertat.

LCD: Pantalla de cristall líquid.

Llibreria: En informàtica, una biblioteca (de l'anglès library) és un conjunt d'implementacions funcionals, codificades en un llenguatge de programació, que ofereix una interfície ben definida per a la funcionalitat que s'invoca.

Processing: Llenguatge de programació de codi obert basat en Java.

Software lliure: programari de tipus obert, el qual pot ser consultat o modificat sense restricció.
Sensor: Dispositiu capaç de detectar magnituds físiques o químiques, anomenades variables d'instrumentació, i transformar-les en variables elèctriques.

Shield: Interfícies disponibles per connexió directa Arduino.

Sketch: Nom de programes dissenyats per a plataforma Arduino.

Wiring: Llenguatge de programació utilitzat en Arduino basat en C.

Workshop: Tallers amb temàtiques d'electrònica, audiovisual, art media, etc.

9. Bibliografia

NRF24L01 Característiques

[1] <http://wiki.makespacemadrid.org/index.php?title=NRF24L01>

RF24-Master Library

[2] <http://wiki.makespacemadrid.org/index.php?title=NRF24L01>

NRF24L01 “pipes”

[3] http://www.cs.mun.ca/~paul/cs4723/material/nordic/nrf24l01_tutorial_3.pdf

NRF24 payload size

{4} <https://devzone.nordicsemi.com/question/4114/packet-size-in-nrf24l01/>

Topología de Red Celular

[5] <https://sites.google.com/site/topologiasdered708/home/topologia-celular>

Topología de Red En Estrella

[6] https://es.wikipedia.org/wiki/Red_en_estrella

Tutoriales Arduino prometec

[7] <http://www.prometec.net/>

Optimización del código de programación para Arduino

[8], <http://booleanbite.com/web/optimizacion-de-memoria-de-nuestro-codigo-de-arduino/>

Arduino, pàgina oficial, (18/10/2015). Informació tècnica

[9], <http://playground.arduino.cc/>

Buzzer, informació tècnica, (18/10/2015). Informació sobre característiques i exemple

[10] <http://www.prometec.net/buzzers/>

Arduino, informació tècnica, (20/10/2015). Informació riesgos elèctricos

[11] <http://www.trastejant.es/blog/10-formas-de-destruir-un-arduino/>

Ultrasons HC-SR04, (21/10/2015). Tutorial

[12] <http://elcajondeardu.blogspot.com.es/2014/03/tutorial-sensor-ultrasonidos-hc-sr04.html>

Utilització del comandament a distància per infrarojos, (25/10/2015). Lectura del codi i exemple

[13] <http://elcajondeardu.blogspot.com.es/2014/02/utilizacion-mando-infrarrojos-i-lectura.html>

Sensor VS1318B, (28/10/2015). Connexió i exemple

[14] <http://arduino.stackexchange.com/questions/3926/using-vs1838b-with-arduino>

Fòrum pàgina oficial Arduino, (29/10/2015). Arduino Forum

[15] <http://forum.arduino.cc/index.php?topic=147680.0>

LCD Keypad Shield, (29/10/2015). Tutorial

[16] <http://www.prometec.net/lcd-keypad-shield/>

LCD 16x2, (30/11/2015). Conexió i exemple

[17] <http://librearduino.blogspot.com.es/2014/01/display-lcd-16x2-tutorial-demostracion-arduino-caracteres-personalizados.html>

Sensor DHT11, (01/11/2015). Manual connexió i exemple

[18] <http://www.instructables.com/id/Sensor-de-Temperatura-y-Humedad-DHT11-y-Arduino/>

Guia del Arduimaniaco, (01/11/2015). Comparativa entre Arduino i LPC1769

[19] <http://www.xataka.com/especiales/guia-del-arduinomaniaco-todo-lo-que-necesitas-saber-sobre-arduino>

Interrupcions Arduino, (12/11/2015). Conexió i exemple

[20] <http://www.educachip.com/como-y-por-que-usar-las-interrupciones-en-arduino/>

Programació, (18/11/2015). Resetejar Array en Arduino

[21] <http://arduino.stackexchange.com/questions/3167/clear-existing-array-when-getting-new-serial-command>

Shield GSM/GPRS. Tutorial SIM900

[22] <http://www.simpasture.com/36023.html>

10. Annexos

Annex 1. Geografia

Per a la ubicació dels diferents elements del projecte, hem de tenir en compte el paper que juga cada sensor.

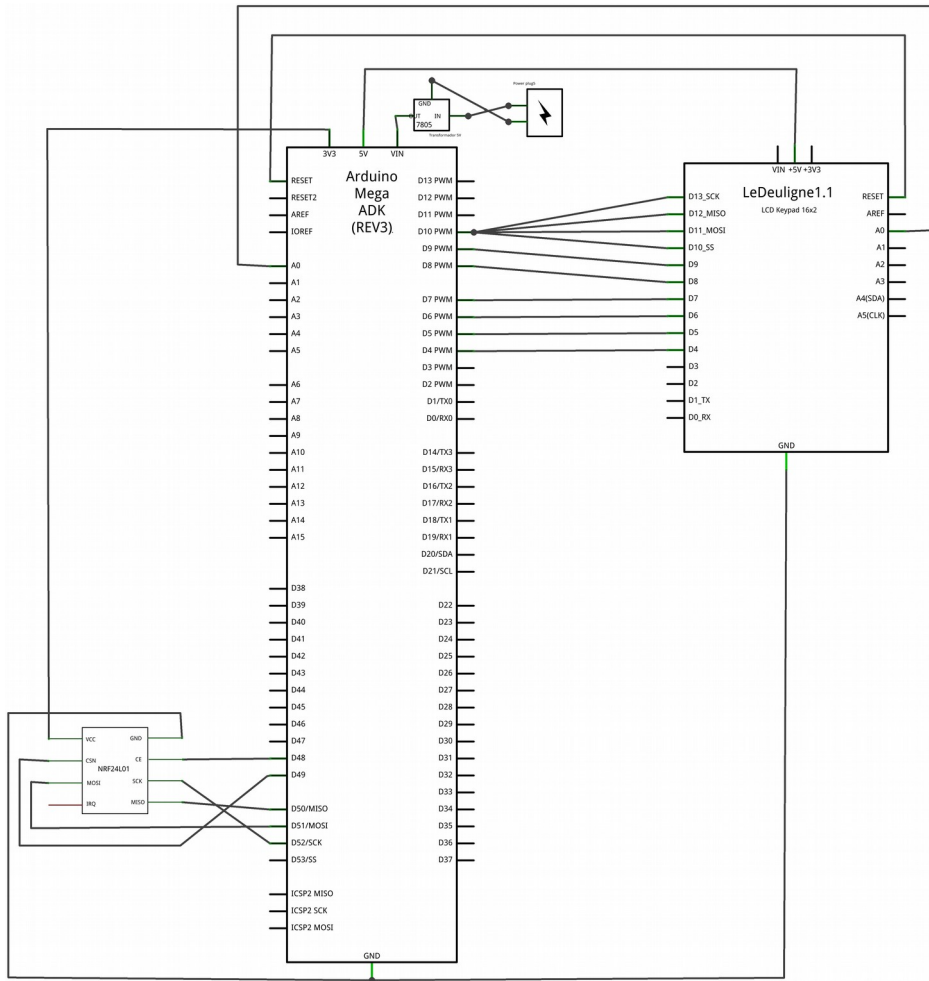
Per als sensors que detecten invasió domiciliària, en fonamental posar-los en punts concrets dissimulats perquè no siguin evidents i camuflar-los amb flors, gerros, llums o posar-los sota les taules.

Per als sensors "tranquils" com Clima o Cort de corrent elèctric gairebé qualsevol lloc pot ser correcte, clima en un balcó o safareig o qualsevol altre lloc que se'ns ocorri.

El sensor de tall de corrent es l'únic que necessita un endoll i no per la seva alimentació, que també és amb bateries, sinó per detectar si hi ha corrent elèctric. Per aquest cas s'ha habilitat una presa de corrent en un armari de la cuina.

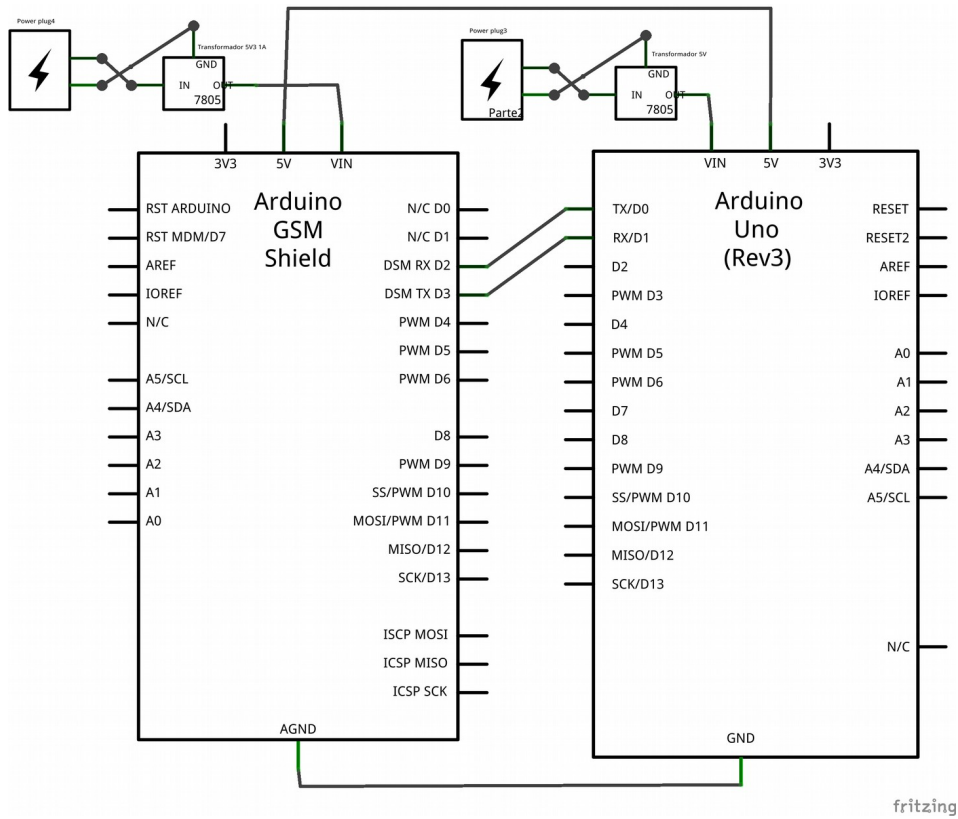
Finalment, els mòduls "control" i "comunicació" poden estar tranquil·lament en un armari, cosa convenient sobretot per "comunicació" per l'ocasional so de la connexió amb la telefonia mòbil (lleuger brunzit).

Annex 2. Esquema elèctric bloc control

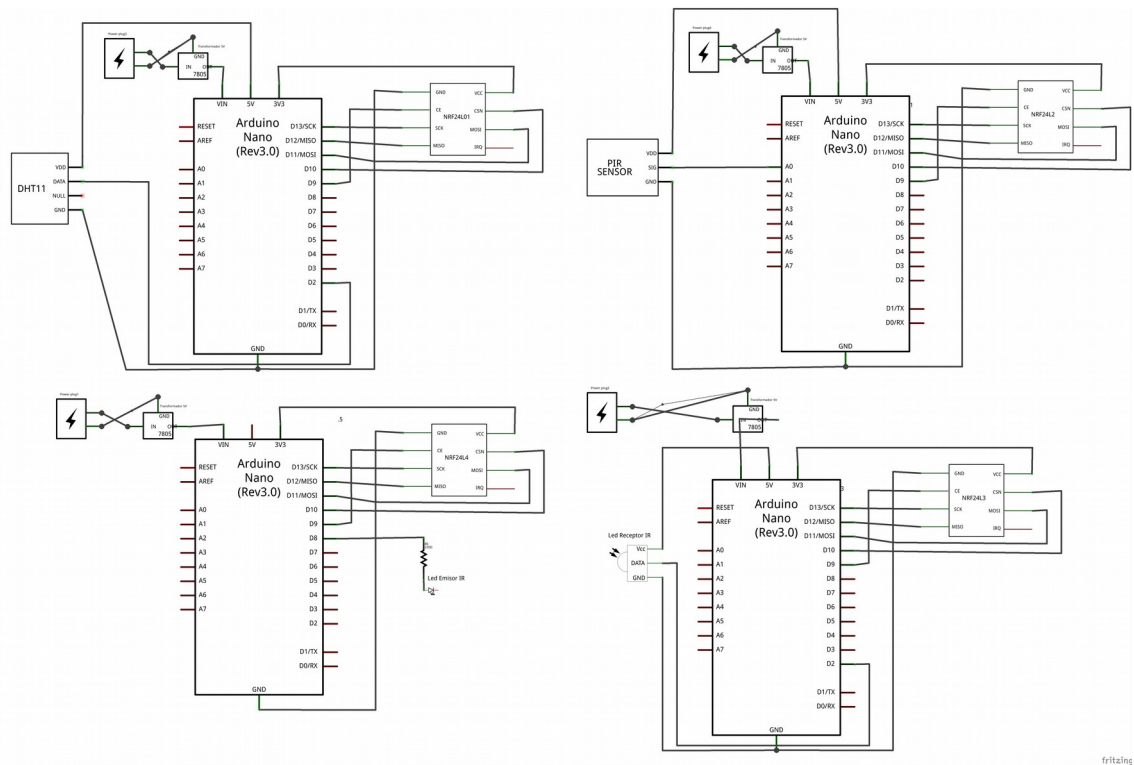


fritzing

Annex 3. Esquema elèctric bloc comunicació



Annex 4. Esquema elèctric blocs detecció



Annex 5. Guia d'instal·lació

Per tal d'instal·lar i executar el sistema *Alarmino* s'han de realitzar els següents passos:

Comprovar el contingut del fitxer `mpuenter_CodiFinal.zip`, que ha d'incloure:

- Una carpeta anomenada '*Alarmino2v0.1*', que ha de contenir les subcarpetes amb tot el programari i una subcarpeta amb les llibreries necessàries:
 - carpeta `modulClima`, que conté el codi font executable del modul
 - `modulControl`, que conté el codi font executable del modul
 - `modulGPRS`, que conté el codi font executable del modul
 - `modulIR`, que conté el codi font executable del modul
 - `modulMoviment`, que conté el codi font executable del modul
 - `modulUltrasons`, que conté el codi font executable del modul
 - llibreries, conté les llibreries necessàries per a la compilació i execució del sistema autònom.
1. Instal·lar l'*IDE Arduino 1.6.8*. Visitar la pàgina de descàrrega d'arduino i descarregar i descomprimir l'IDE adequat al seu sistema operatiu.

<https://www.arduino.cc/en/Main/OldSoftwareReleases#previous>

2. Afegir les llibreries necessàries a l'arduino IDE 1.6.8, de la següent manera:
 - Desde el menú 'Programa → Incloure llibreria → Afegir llibreria .ZIP', seleccionar les llibreries presentades en l'arxiu comprimit del TFC.
3. Obrir els sis mòduls de l'aplicació desde l'IDE i compilar-los. Un cop fet això pujar cada mòdul a un arduino determinat, com es descriu a continuació:
 - `MòdulControl`: Es carrega a un arduino Mega equipat amb una shield LCD Keypad i un mòdul de ràdio NRF2401.
 - `MòdulSensor(x)`: Pujar cada un dels mòduls sensor a un arduino nano independent amb un mòdul de ràdio NRF2401 a cada un d'ells.
 - `MòdulGPRS`: Es carrega en un arduino UNO dotat d'una shiels GSM/GPRS i alimentat amb una bateria de 9 volts.

Es defineix el mode d'operació per la posada en marxa de l'equip, en la següent figura s'indica mitjançant una numeració de les diferents parts dels equips:

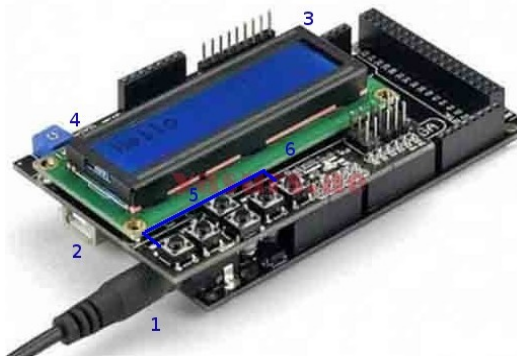


Figura 10. Conexions mòdul de control

1. Entrada d'alimentació per a bateria de 9V.
2. Port alimentació USB.
3. LCD Keypad 16x2 per al control del menú.
4. Potenciòmetre control lluminositat LCD.
5. Cinc botons per al control del menú (selecció, esquerre, pujar, baixar i dreta).
6. Reset Arduino.
7. Font alimentació 5V (per al control de talls de subministrament elèctric).

Mode d'operació:

1. Alimentar *Arduino* amb la bateria de 9V
2. Connectar un transformador des de la corrent elèctrica a la font d'alimentació 3V (per al control dels talls de llum).
3. Immediatament apareixerà en la pantalla LCD la sol·licitud de la clau de seguretat, de 3 números, que per defecte és '111'. Introduir-la i prémer la tecla 'Select'.
4. Durant uns dos segons veurem per LCD el missatge de benvinguda d'*Alarmino* mentre sonen les 4 primeres notes de la melodia d'*Star Wars*.
5. Després d'un segon se'ns mostrarà per pantalla el menú de l'aplicació.

Resolució de problemes:

1. Funcionament anòmal de l'alarma:
 - Comprovi que l'equip no està patint danys externs, calor excessiu, etc.
 - Comprovi l'estat dels sensors.

Annex 6. Execució i compilació

Arduino IDE 1.6.8

Arduino posseeix un entorn de programació propi, el qual permet edició, depuració, compilació i posterior pujada dels programes o Sketch a la placa *Arduino*. A més de les característiques tractades anteriorment, l'entorn permet la comunicació amb el port sèrie tant de lectura com d'escriptura i monitorització. Els programes dissenyats són denominats Sketch i com s'ha comentat anteriorment poden ser verificats amb el programari propietari.

A continuació una imatge de l'editor esmentat:

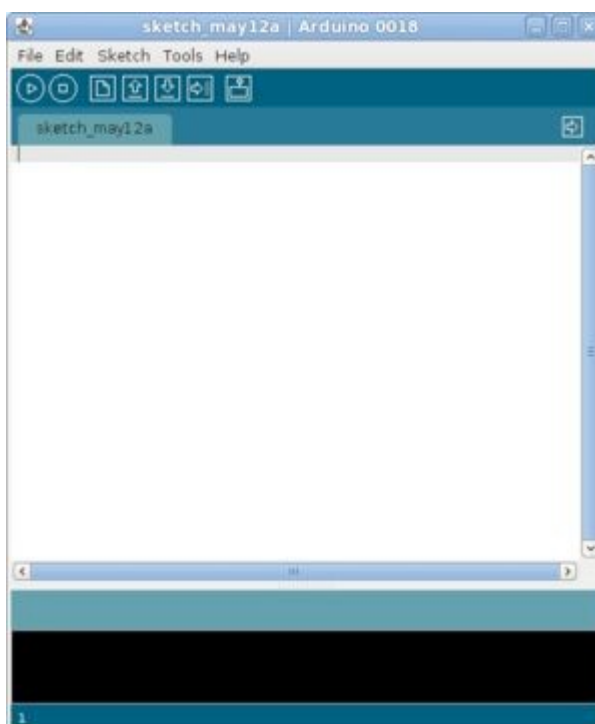


Figura 16, entorn de programació Arduino

El mateix editor conté les llibreries comunament utilitzades en *Arduino*, a més d'exemples tipus; poder comprovar l'electrònica o realitzar proves funcionals. També s'ha de destacar que l'editor està proveït d'un depurador d'errors. Passos a seguir per poder realitzar una programació, verificació, compilació i posterior càrrega del codi al micro-controlador:

- 1er Obrir el programa Arduino, si no es disposa pot aconseguir-se en el següent enllaç: <http://arduino.cc/en/Main/Software>.

- 2on En el propi editor es pot copiar el codi lliurat o pot editar-se.

- 3er Un cop editat o copiat es pressiona a la tecla Verify i es realitza la depuració.

- 4rt Connectar el dispositiu al port USB, de vegades cal establir en quin port es troba l'USB i el model de placa Arduino de què disposem. Per a això només caldrà anar a preferències i seleccionar els valors correctes.

- 5é Si la verificació ha tingut èxit, vam passar al procés de compilació i pujada del codi a la placa, per a això pressionem a Upload to I/O Board

- 6é Un cop el programa ha estat pujat si s'estan transmetent dades via Serial podríem comprovar-les pressionant en Serial, en el nostre cas a 9600 bauds.

Nota : En cas d'estar utilitzant GSM/GPRS Shield hem de retirar els 2 jumper que es troben a la part superior de la Shield, per poder realitzar la pujada del firmware.