

Alarmino, alarma per la llar

Marcos Puente Ruiz

Enginyeria Tècnica en Informàtica de Gestió
Sistemes encastrats

Jordi Bécares Ferrés

Pere Tuset Peiró

Xavi Vilajosana Guillen

06/07/2016



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 Peiro 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 <(màxim 250 paraules): <i>Amb la finalitat, context d'aplicació, metodologia, resultats i conclusions del treball></i> | |
| <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'avís és GSM. Alarmino 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 hardware y software lliures, com són les micro-controladores <i>Arduino</i> i el seu entorn de programació <i>Arduino IDE 1.6.8 Release</i>.</p> | |
| Abstract <(in English, 250 words or less)>: | |
| | |

Índex

| | | |
|------------|------------------------------|----------|
| 1. | Tasques Realitzades | 1 |
| 1.1 | Objectius assolits | 1 |
| 1.2 | Objectius no assolits | 1 |
| 1.3 | Diagrama de blocs | 1 |
| 1.4 | Diagrama de blocs | 2 |
| 1.5 | Sistemes de test | 4 |
| 2. | Autoavaluació | 4 |
| 3. | Conclusions | 5 |
| 4. | Guía d'instal·lació | 5 |

Llista de figures

Figura 1. Diagrama de blocs 2

Figura 2. Esquema comunicació 4

1. Tasques Realitzades

A continuació es detallen els objectius assolits, els no assolits i les solucions triades per a pal·liar els problemes trobats durant el desenvolupament del projecte.

1.1 Objectius assolits

Tots els objectius principals i secundaris del projecte han estat assolits amb èxit. També s'ha incorporat al codi final un dels objectius marcats com a extraordinaris a la planificació, que és el control del menú per comandament a distància IR.

Objectius principals

- Interfície d'usuari (HMI) basada en una LCD Keypad de 16x2 que realitza la tasca de mòdul principal, gestionant els diferents blocs del sistema i el menú d'opcions, a més de la comunicació amb el mòdul GSM.
- Sistema de comunicació per RF (NRF24L01) entre Arduinos per tal de comunicar els sensors independents amb el mòdul principal, i amb el mòdul GSM.
- Sistema d'alarma d'incendis, gestionada per un mòdul independent i dotat del sistema de comunicació per RF..
- Sistema de detecció de moviment, gestionat per un mòdul independent i dotat del sistema de comunicació per RF.
- Sistema de control de pas de línia, gestionat per un mòdul independent i dotat del sistema de comunicació per RF.
- Sistema de control d'accés a un punt concret localitzat, gestionat per un mòdul independent i dotat del sistema de comunicació per RF.

Objectius secundaris

- Sistema de comunicació entre el mòdul principal i el mòdul GSM per tal de generar les alertes entre el sistema i l'usuari (unidireccional), via SMS, quan es produeixi un avís d'alarma.
- Control d'accés, activació i desactivació de l'alarma a per clau de seguretat.

Objectius extraordinaris

- Control del menú per comandament a distància (IR).

1.2 Objectius no assolits

Objectius extraordinaris

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

1.3 Diagrama de blocs

El sistema es compost per tres blocs funcionals:

- Bloc de procés y control : Arduino Mega + LCD Keypad + NRF24L01

- Bloc de comunicació : Arduino UNO + Mòdul GSM + NRF24L01
- Bloc de detecció (4 unitats) : Arduino NANO + Sensor associat + NRF24L01

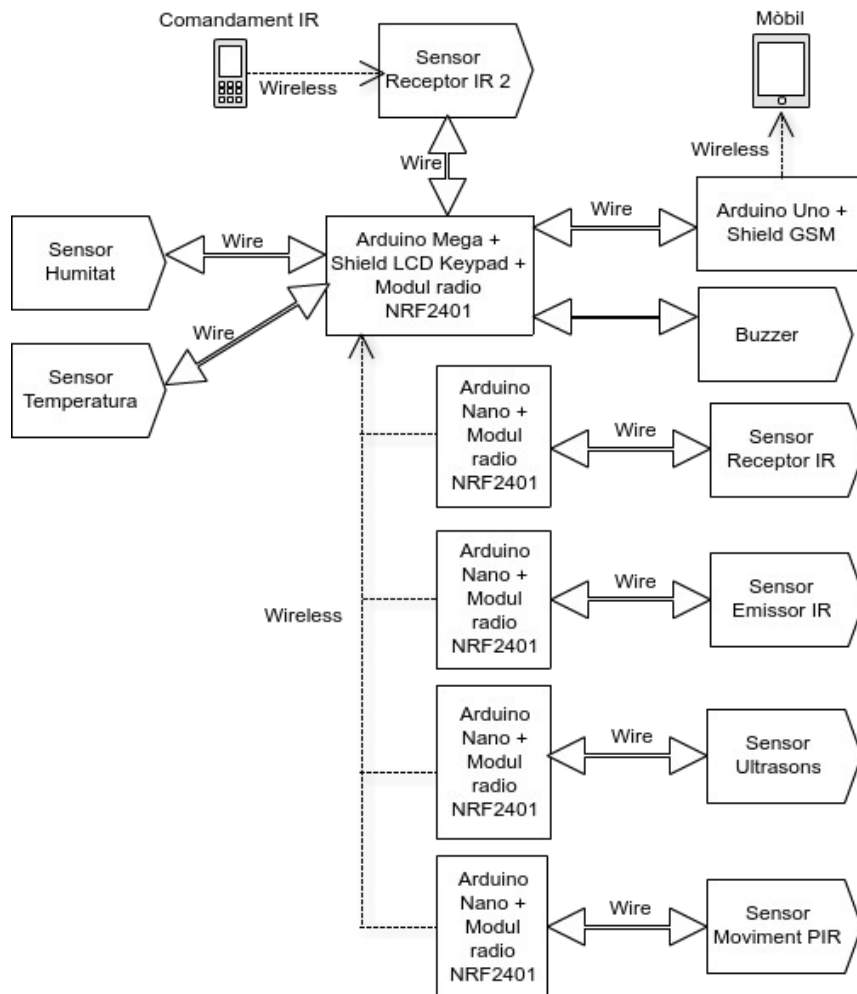


Figura 1. Diagrama de blocs

Comandament IR i el Mòbil es poden considerar elements externs al sistema, malgrat que poden interactuar programant el mode de procés i rebent informació respectivament.

1.4 Esquema de comunicacions

Definició dels camps del paquet de comunicació entre mòduls:

// Estructura de dades per a comunicar per RF entre Arduinos

// Recomenat tamany <= 32 Bytes

byte **cap**; // Marca inici = 123 // Longitud 1 byte acumulada = 1 Byte

byte **sensor**; // Codi numeric sensor // Longitud 1 byte acumulada = 2 Byte

// 1 - sensor de clima

// 2 - sensor d'infrarojos

// 3 - sensor d'Ultrasons

// 4 - sensor de Moviment

// 5 - detector tallCorrent Transformador + alimentador adaptat al efecte

// 0 - aquest valor indica que es tracta d'un registre de sincronisme

// de data i hora que ve informat al camp 'epoch'

byte **serie**; // N° seqüència sensor // Longitud 1 byte acumulada = 3 Byte

| | | |
|------------------------------|-------------------------|--|
| int valor1 ; | // Sensor temperatura | // Longitud 4 byte acumulada = 7 Byte |
| int valor2 ; | // Sensor infrarojos | // Longitud 4 byte acumulada = 11 Byte |
| int valor3 ; | // Sensor moviment | // Longitud 4 byte acumulada = 15 Byte |
| int valor4 ; | // Sensor ultrasons | // Longitud 2 byte acumulada = 17 Byte |
| byte valor5 ; | // Sensor de corrent | // Longitud 1 byte acumulada = 18 Byte |
| byte interrupcio ; | // Flag Polling = 111 | // Longitud 1 byte acumulada = 19 Byte |
| | // Flag Interrup = 222 | |
| byte alarma ; | // Codi d'alarma | // Longitud 1 byte acumulada = 20 Byte |
| unsigned long epoch ; | // Hora UNIX | // Longitud 4 byte acumulada = 24 Byte |
| byte periferic ; | // Arduino que contesta | // Longitud 1 byte acumulada = 25 Byte |
| byte menuActiu ; | // Menú seleccionat | // Longitud 1 byte acumulada = 26 Byte |
| byte submenuActiu ; | // Submenú seleccionat | // Longitud 1 byte acumulada = 27 Byte |
| byte reservat1 ; | // Reservat ús futur | // Longitud 1 byte acumulada = 28 Byte |
| byte reservat2 ; | // Reservat ús futur | // Longitud 1 byte acumulada = 29 Byte |
| byte cua ; | // Marca final = 234 | // Longitud 1 byte acumulada = 30 Byte |

Explicació dels camps del paquet de comunicació per RF:

- **cap i cua (byte y byte)** marquen els límits del missatge i serveixen per a la comprovació de recepció íntegra del mateix.
- **sensor (byte)** el número assignat a cada tipus de sensor (1 → 255).
- **serie (byte)** el número de serie del sensor (per si hi ha més d'un del mateix tipus (p.e. tres de moviment o dos d'infrarojos).
- **valor(x) (float)** camp per que el programa sensor xy posi el valor llegit.
- **alarma (byte)** pot ser informat pel mòdul sensor amb el valor 1 en cas d'alarma o 0 en el seu defecte.
- **interrupcio (byte)** Informa el codi d'interrupció de l'alarma (111 o 222).
- **unsigned long (4 bytes)** s'utilitza per allotjar la data UNIX que s'envia als mòduls sensor (perifèrics) en fer la sincronització.
- **periferic (byte)** el número de serie del mòdul sensor perifèric.
- **menuActiu (byte)** per informar els mòduls perifèrics quin és el menú actiu de l'aplicació.
- **submenuActiu (byte)** per informar els mòduls perifèrics quin és el submenú actiu de l'aplicació.
- **reservat1 (byte)** reservat per a ús futur
- **reservat2 (byte)** reservat per a ús futur

Esquema gràfic de les comunicacions per RF:

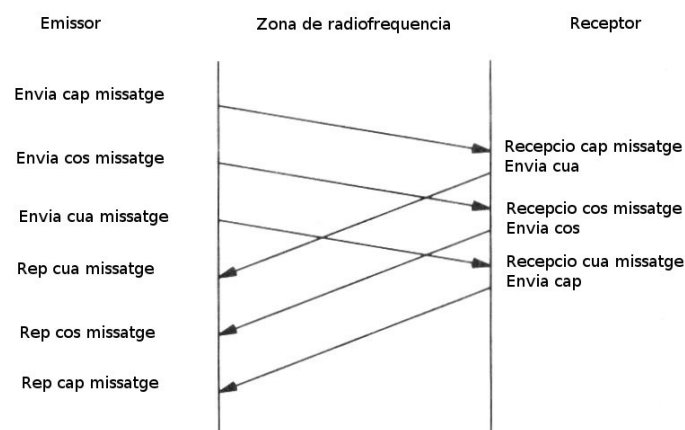


Figura 2.

Esquema comunicació

1.5 Sistemes de test

Els sistemes de test utilitzats en la realització d'aquest projecte han sigut els següents:

- **Proves estàtiques:** Són proves realitzades sense executar el codi de l'aplicació. S'ha anat seguint els documents de diagrames de blocs i de fluxos per tal de garantir el correcte desenvolupament de l'aplicació.
- **Proves dinàmiques:** Engloba les proves realitzades a partir de l'execució del codi font.
- **Proves de regressió:** S'avalua el correcte funcionament del programari desenvolupat davant evolucions o canvis funcionals. El propòsit d'aquestes proves és assegurar que els casos de prova que ja havien estat provats i van ser reeixits romanguin així. Aquest punt s'ha gestionat mitjançant diferents versions del codi desenvolupat.
- **Proves d'integració:** És el nivell de proves posterior a les proves modulars dels components d'un sistema. Se centra principalment en provar la comunicació entre els components d'un mateix sistema, comunicació entre sistemes o entre maquinari i programari.

2. Autoavaluació

La topologia de 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).

La connexió física és per tant independent de la distribució geogràfica. L'abast dels mòduls NRF24L01 amb sòcol (inclou capacitàncies) i antena incorporada és molt superior a les necessitats d'un habitatge mitjana 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 channel 1 Fins channel 108 per sobre de la freqüència de connexió internet) sense diferència apreciable.

La topologia actual, amb comunicacions per RF, físicament és una topologia de xarxa Cel·lular ja que tots els nodes estan a l'abast de tots els altres.

No obstant la lògica de les comunicacions establertes com standard és com si haguessin dues xarxes diferents.

1) Una xarxa en anell:

Un node ("control") donant torn (polling) a uns nodes "sensor" recollint informació un per un i serialitzant així la comunicació.

2) Una xarxa en estrella:

Un node ("Comunicació") rebent peticions de missatgeria directament d'altres nodes "sensor" que no es comuniquen entre ells.

3) 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 nodes "en anell" al mòdul "comunicació" per al seu processament.

3. Conclusions

Per tant, la meua sensació és la d'haver redisenyat un projecte que en sí mateix era poc robust.

4. 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 compil·lació i execucio del sistema autònom.

1- Instal·lar l'IDE Arduino 1.6.8. Vvisitar 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.

Per tal d'iniciar l'aplicació cal introduir la clau '111' al menú del sistema. Un cop a dins del menú es poden trobar les quatre opcions disponibles i activar l'alarma, entre d'altres.