

Treball de Fi de Grau
Grau en Enginyeria de Tecnologies Industrials

**Sistema de supervisió de temperatures amb
indicació de l'estat en un telèfon mòbil**

MEMÒRIA

Autor: Edgar González Quevedo

Director: Emili Lupon Rosés

Convocatòria: Juliol 2015



Escola Tècnica Superior
d'Enginyeria Industrial de Barcelona



Resum

Aquest projecte consisteix en el desenvolupament d'una aplicació basada en un microcontrolador que supervisa la temperatura de diferents punts captada amb uns sensors adients i lliura la informació sobre el seu estat a un telèfon mòbil. El sistema ha de detectar si la temperatura d'aquests punts està o no dins d'un interval de temperatures concret i ha de comunicar un missatge a un telèfon mòbil amb sistema operatiu Android informant de les temperatures que estan fora del rang desitjat, visualitzant-se aquesta informació en la pantalla del telèfon mòbil.

Per a fer aquest projecte el primer pas ha estat la definició d'especificacions, com el número de sensors i la interfície amb l'usuari. Després s'ha fet el disseny de l'estructura del sistema i, tot seguit, s'ha escollit el hardware del mateix. L'elecció dels components emprats s'ha basat en el seu grau de disponibilitat a la universitat. La comunicació entre el mòbil i el microcontrolador s'ha fet a través d'un mòdul de Bluetooth. Un cop definit el hardware, el següent pas ha estat la programació del microcontrolador, que s'ha fet en llenguatge C. El programa s'ha dissenyat de manera que controli de forma endreçada, sense que hi hagi interferències ni conflictes elèctrics, tots els elements que estan connectats al microcontrolador. Finalment, s'ha dissenyat una aplicació per al telèfon mòbil amb Android amb ajuda del software de disseny del MIT anomenat "MIT App Inventor".

El sistema final permet a qualsevol mòbil amb Android i Bluetooth connectar-se al microcontrolador i rebre dades de temperatures instantànies, màximes i mínimes. També incorpora un sistema d'alarma que salta quan un sensor se surt d'un dels seus límits superior o inferior, tot indicant de quin sensor es tracta, i també permet configurar els valors d'aquestes temperatures límits per a cada sensor. A més, es pot visualitzar la temperatura instantània de cada sensor a través d'una pantalla LCD situada al costat del node central.

Aquest projecte ha assolit satisfactòriament els objectius proposats. La comunicació per Bluetooth entre el telèfon mòbil i el microcontrolador funciona fins a una distància de 100 metres aproximadament.

Índex

RESUM	3
ÍNDEX	5
1. GLOSSARI	7
2. PREFACI	9
2.1. Motivació	9
3. INTRODUCCIÓ	11
3.1. Objectius del projecte	11
3.2. Abast del projecte.....	11
4. ESPECIFICACIONS DEL SISTEMA	13
4.1. Nombre de sensors	13
4.2. Obtenció de la temperatura.....	13
4.3. Visualització de la temperatura	13
4.4. Aplicació del telèfon mòbil.....	14
4.5. Cicle de funcionament.....	15
5. ARQUITECTURA DEL SISTEMA	17
5.1. Esquema de l'aplicació	17
5.2. Selecció de components	17
6. CARACTERÍSTIQUES DELS COMPONENTS	21
6.1. Microcontrolador PIC18F4520	21
6.1.1. Característiques generals.....	21
6.1.2. Ports paral·lels.....	22
6.1.3. Timers	23
6.1.4. USART	24
6.1.5. Placa de desenvolupament Open18F4520	25
6.1.6. Eines per a la programació.....	26
6.2. Sensor de temperatura DS18B20	27
6.2.1. Característiques generals.....	27
6.2.2. Bus de comunicació 1-Wire.....	30
6.3. Pantalla LCD1602	31
6.4. Transceptor Bluetooth Click Mikroe-958	33
6.4.1. Característiques generals.....	33
6.4.2. Mòdul RN41	34

7. METODOLOGIA	37
8. CONFIGURACIÓ DEL HARDWARE	39
8.1. Connexió del Bluetooth Click	39
8.2. Connexió de la LCD1602.....	40
8.3. Connexió dels sensors de temperatura	42
9. COMUNICACIÓ ENTRE PIC18F4520, BLUETOOTH CLICK I TELÈFON MÒBIL	45
9.1. Configuració del <i>baud rate</i> del microcontrolador i del Bluetooth Click.....	45
9.2. Format de les dades	46
10. DESCRIPCIÓ DELS PROGRAMES DEL MICROCONTROLADOR	49
10.1. Programa principal.....	49
10.1.1. Funció.....	49
10.1.2. Estructura i funcionament.....	50
10.1.3. Organització del codi	55
10.2. Programa per configurar el Bluetooth Click	56
10.2.1. Funció.....	56
10.2.2. Estructura i funcionament.....	56
10.2.3. Organització del codi	57
11. DESCRIPCIÓ DE L'APLICACIÓ DEL TELÈFON MÒBIL	59
11.1. MIT App Inventor 2	59
11.1.1. Designer	59
11.1.2. Blocks.....	61
11.2. Descripció de l'aplicació.....	62
11.2.1. Funcionament.....	62
11.2.2. Manual d'usuari	63
12. PRESSUPOST	67
13. IMPACTE AMBIENTAL	69
CONCLUSIONS	71
BIBLIOGRAFIA	73
Bibliografia complementària	73

1. Glossari

1-Wire: És un protocol de comunicacions en sèrie dissenyat per *Dallas Semiconductor*. Està basat en un bus amb un únic mestre i diversos esclaus que es comuniquen a través d'un únic fil de dades.

ALU (de l'anglès *Arithmetic Logic Unit*): És un circuit digital que realitza operacions aritmètiques i operacions lògiques entre dos operands.

ASCII (de l'anglès *American Standard Code for Information Interchange*): És un codi de caràcters basat en l'alfabet llatí, tal com s'usa a l'anglès modern.

Baud rate: És el nombre d'unitats de senyal transmeses per segon. Un *baud* pot contenir diversos bits.

Bit de paritat: És un dígit binari que indica si el nombre de bits amb valor 1 en un conjunt de bits es parell o imparell. Els bits de paritat conformen el mètode de detecció d'errors més simple.

CMOS (de l'anglès *Complementary metal-oxide-semiconductor*): És una de les famílies lògiques utilitzades en la fabricació de circuits integrats. La seva principal característica consisteix en l'ús de transistors de tipus pMOS i nMOS configurats tals que, en estat de repòs, el consum d'energia és degut únicament als corrents paràsits.

CPU (de l'anglès *Central Processing Unit*): És el hardware dins un computador o altres dispositius programables, que interpreta les instruccions d'un programa informàtic mitjançant la realització de les operacions bàsiques aritmètiques, lògiques i d'entrada/sortida del sistema.

EEPROM (de l'anglès *Electrically Erasable Programmable Read Only Memory*): És un tipus de memòria de només lectura que pot ser programada, esborrada i reprogramada elèctricament.

IDE (de l'anglès *Integrated Development Environment*): És una aplicació de software que proporciona serveis integrals per facilitar el desenvolupament de software al programador del microcontrolador.

LCD (de l'anglès *Liquid Crystal Display*): És una pantalla prima i plana formada un nombre de píxels en color o monocroms situats davant una font de llum o reflectora.

Master-Slave: És un model de comunicació en el que un dispositiu o procés té control

unidireccional sobre un o més dispositius. En alguns sistemes el *master* o mestre és escollit d'un grup de dispositius elegibles, amb els altres dispositius actuant de *slaves* o esclaus.

Microcomputador: És un computador la CPU del qual és un microprocessador.

Microcontrolador: És un microcomputador materialitzat en un únic chip integrat (conté la CPU, la memòria de programa, la memòria de dades i els ports d'entrada/sortida), destinat a la implementació d'aplicacions relativament senzilles.

Microprocessador: És un processador o CPU materialitzat en un únic circuit integrat o xip. A vegades es requereixen dos circuits integrats per materialitzar microprocessadors molt potents, rebent el conjunt el nom de *chipset*.

PLL (de l'anglès *Phase-Locked Loop*): És un tipus de circuit electrònic i sistema de control que genera un senyal de sortida la fase del qual està relacionada amb la fase del senyal d'entrada.

Pull-up: En electrònica, és l'acció d'eleva la tensió de sortida d'un circuit lògic.

RAM (de l'anglès *Random Access Memory*): La memòria d'accés aleatori s'utilitza com a memòria de treball de computadores per al sistema operatiu, els programes i la major part del software. Materialitza la memòria de dades i, en ordinadors en els que els programes executats canvien al llarg del temps, la memòria de programa.

RISC (de l'anglès *Reduced Instruction Set Computer*): En arquitectura computacional, és un tipus de disseny de CPU utilitzat en microprocessadors que es caracteritza per tenir instruccions de mida fixa i presentades en un nombre reduït de formats, i perquè només les instruccions de càrrega i emmagatzematge accedeixen a la memòria de dades.

UART (de l'anglès *Universal Asynchronous Receiver-Transmitter*): És un mòdul digital que implementa la interfície d'algunes comunicacions en sèrie asíncrones.

USART (de l'anglès *Universal Synchronous Asynchronous Receiver Transmitter*): És un mòdul digital que implementa la interfície d'algunes comunicacions en sèrie síncrones o asíncrones.

2. Prefaci

2.1. Motivació

La finalitat principal del projecte és la realització del treball de fi de grau del GETI (Grau d'Enginyeria en Tecnologies Industrials) de l'Escola Tècnica Superior d'Enginyeria Industrial de Barcelona. El treball està definit en la memòria dels títols de grau de l'Escola com un exercici original a realitzar de manera individual, que cal presentar i defensar davant d'un tribunal.

Amb aquest projecte també es pretén ampliar els coneixements adquirits a l'assignatura d'Electrònica, assignatura troncal del GETI. En aquesta assignatura, s'ensenyen els coneixements més bàsics de les diferents branques de l'electrònica. La branca en la que es vol aprofundir en aquest projecte és la dels microcontroladors.

Per poder realitzar aquest projecte, ha estat de gran utilitat cursar l'assignatura optativa del GETI anomenada "Desenvolupament d'aplicacions basades en microcontroladors". Aquesta assignatura reforça l'aprenentatge d'aquesta branca de l'electrònica ja que ha estat necessari entendre el món dels microcontroladors i aprendre a programar-los.

3. Introducció

3.1. Objectius del projecte

El projecte consisteix en el disseny i realització física d'una aplicació basada en un microcontrolador. L'aplicació haurà de permetre la supervisió de les temperatures d'un conjunt de punts amb un telèfon mòbil. En cada un d'aquests punts hi haurà un sensor de temperatura que es comunicarà amb el microcontrolador.

L'usuari haurà de poder veure les temperatures des de la pantalla del telèfon mòbil, i rebre algun tipus de missatge en cas que alguna d'elles se surti d'uns límits, superior o inferior, configurables per l'usuari. El sistema també disposarà d'un visualitzador situat al costat del microcontrolador per poder veure les temperatures.

La comunicació entre el telèfon mòbil i el microcontrolador serà bidireccional i per radiofreqüència.

També es desenvoluparà el software per al microcontrolador, i es farà el desenvolupament d'una aplicació per al telèfon mòbil.

Un cop assolits els objectius principals, s'intentarà millorar l'aplicació del mòbil per tal de que l'usuari pugui visualitzar altres tipus de temperatures relatives a cada punt, com les temperatures màximes o mínimes, i que pugui comprovar quins valors límits de temperatura té establerts per a cada sensor. El sistema es concebrà de forma que el microcontrolador sigui qui faci tots els càlculs necessaris de la manera més eficient possible, reduint les tasques del telèfon mòbil a la interfície amb l'usuari.

3.2. Abast del projecte

En aquest apartat es descriuran les limitacions que s'han establert abans de començar el projecte i que afecten al desenvolupament i al resultat del mateix.

En primer lloc, s'ha decidit fer servir components que estiguin disponibles a la Universitat, concretament al Departament d'Enginyeria Electrònica. També s'ha escollit l'Android com el sistema operatiu que ha de tenir el telèfon mòbil, ja que permet el desenvolupament d'aplicacions de forma gratuïta.

D'altra banda, s'ha intentat que s'hagi d'aprendre només un llenguatge de programació nou. Donat que l'aplicació per al mòbil es programa en llenguatge Java i el microcontrolador en llenguatge C, s'ha hagut de simplificar la programació de l'aplicació per al telèfon mòbil. Per

fer-ho, s'ha fixat a Bluetooth el tipus de comunicació entre el mòbil i el microcontrolador. D'aquesta manera, es poden aprofitar les rutines existents per al control de Bluetooth en el telèfon mòbil, resultant més senzilla la programació del mateix.

Per últim, degut a la darrera limitació, és necessària la utilització d'un transceptor Bluetooth que permeti al microcontrolador comunicar-se amb el mòbil a través d'una de les interfícies disponibles al microcontrolador.

4. Especificacions del sistema

En aquest punt és definiran amb detall quines són les característiques que ha de presentar el sistema final, com per exemple, el format en què s'ha d'obtenir la temperatura, on s'ha de mostrar, com ha de ser l'aplicació per al telèfon mòbil, amb quina freqüència s'ha d'actualitzar la temperatura, etc.

4.1. Nombre de sensors

És necessari considerar de quants llocs de diferents de l'espai es vol supervisar la temperatura. Per a aquest projecte s'ha decidit que el nombre màxim de sensors dels quals es podrà obtenir la temperatura simultàniament és igual a 8. Aquest nombre pot variar, tot i que a l'hora de fer les proves es va fixar a 8 arbitràriament.

4.2. Obtenció de la temperatura

El microcontrolador haurà d'obtenir la temperatura de cada sensor en graus Celsius i amb una precisió de 0.5°C. Les temperatures podran oscil·lar entre -10°C i +80°C. A mesura que el microcontrolador vagi obtenint les temperatures, haurà d'anar actualitzant el valor màxim i mínim de la temperatura de cada sensor. A més, haurà de comprovar si les temperatures dels sensors superen el seu límit superior o baixen del seu límit inferior, ambdós independents per a cada sensor. En cas que això passi, el microcontrolador haurà de saber quin dels dos límits s'ha superat, a quin sensor es correspon i comunicar-ho al telèfon mòbil.

4.3. Visualització de la temperatura

La visualització de les temperatures es farà a través de dos visualitzadors diferents: a través d'una pantalla situada al costat del microcontrolador, i a través de la pantalla del telèfon mòbil.

En el primer cas, es visualitzarà la temperatura instantània dels sensors amb la resolució d'una dècima de grau. A cada línia de la pantalla es representarà la temperatura d'un sensor. El format serà el següent: s'escriurà la lletra "S", seguida del número del sensor i de dos punts. Després es posarà la temperatura, que pot ser negativa o no, amb un punt per separar la part entera de la part decimal. Finalment s'escriurà el símbol del grau (°) i una "C" de graus Celsius. A la Taula 4.1 es poden veure alguns exemples que il·lustren com és aquest format. En cas que la pantalla escollida no tingui suficients línies per poder representar les temperatures de tots els sensors a la vegada, s'haurà de disposar de dos

botons, a prop del microcontrolador, que permetin escollir quin conjunt de temperatures es visualitza en cada instant.

Temperatura instantània del sensor 1	S1: 23.7°C
Temperatura instantània del sensor 3	S3: -7.0°C
Temperatura instantània del sensor 7	S7: 57.5°C

Taula 4.1: Exemples de representació de la temperatura en la pantalla de la placa

La visualització de les temperatures en l'aplicació del telèfon mòbil es definirà dins l'apartat 4.4 de la memòria.

4.4. Aplicació del telèfon mòbil

La funció principal de l'aplicació del telèfon mòbil és fer d'interfície amb l'usuari. Donat que es tracta de supervisar la temperatura de diversos punts, l'usuari haurà de poder establir per a cada sensor uns límits superior i inferior, de manera que s'haurà d'enviar una alarma en cas que algun sensor se surti d'aquests límits. L'alarma que es rebrà al mòbil serà una vibració acompanyada amb una notificació amb text a la pantalla indicant de quin sensor es tracta i quin dels seus dos límits s'ha superat. Aquesta alarma saltarà un cop cada 5 segons i desapareixerà en el moment que la temperatura del sensor en qüestió hagi tornat a la normalitat.

L'usuari també haurà de poder visualitzar, per a qualsevol sensor de forma independent, la seva temperatura instantània i les seves temperatures màxima i mínima. Per tal de poder comprovar que la comunicació entre el mòbil i el microcontrolador funciona correctament, hi haurà l'opció de visualitzar els límits superiors i inferiors de la temperatura de cada sensor. D'aquesta manera, es podrà comprovar si els límits que es volen configurar s'estan enviant correctament del mòbil al microcontrolador.

Per a cada sensor es podrà visualitzar un tipus de temperatura a la vegada, és a dir, la instantània, la màxima, la mínima, el límit superior o el límit inferior. Cada temperatura es representarà amb una línia que contindrà una "T", el número del sensor, una abreviació del tipus de temperatura que s'està visualitzant, dos punts, la temperatura amb una resolució d'un grau, amb un punt per separar la part entera de la part decimal, i el símbol del grau seguit d'una "C". A la Taula 4.2 s'il·lustren alguns exemples de representacions de temperatures.

És important destacar que l'aplicació no haurà de fer cap càlcul matemàtic amb les

temperatures. Els càlculs de temperatures màximes i mínimes, així com la comparació de les temperatures límits amb la temperatura instantània, els farà el microcontrolador. Així doncs, les notificacions d'alarma les rebrà el mòbil directament del microcontrolador.

Temperatura instantània del sensor 2	T2inst: 23.7°C
Temperatura mínima del sensor 8	T8min: -7.0°C
Límit superior de la temperatura del sensor 7	T7high: 57.5°C
Temperatura màxima del sensor 4	T4max: 40.0°C
Límit inferior de la temperatura del sensor 1	T1low: 2.2°C

Taula 4.2: Exemples de representació de la temperatura en el telèfon mòbil

4.5. Cicle de funcionament

El microcontrolador haurà de llegir la temperatura de tots els sensors un cop cada segon. Després de la lectura, haurà d'actualitzar els valors màxims i mínims de la temperatura de cada sensor i haurà de comprovar que cada temperatura estigui dins dels seus límits.

Amb aquesta mateixa freqüència, s'haurà d'enviar al mòbil el tipus de temperatura corresponent a cada sensor, una indicació de si hi ha alarma i de quin sensor es tracta. Per defecte, quan s'inicialitzi el microcontrolador, s'enviaran les temperatures instantànies, i els límits superior i inferior per a tots els sensors seran de 50.0°C i -10.0°C, respectivament. En qualsevol moment, l'usuari que estigui connectat amb el seu mòbil podrà canviar aquests paràmetres. Tot i així, aquests canvis es desfaran en el moment que el microcontrolador es desconnecti de l'alimentació.

5. Arquitectura del sistema

5.1. Esquema de l'aplicació

Amb el contingut present en la introducció i les especificacions es pot traçar un esquema funcional de l'aplicació. Els principals elements del sistema són un telèfon mòbil, un microcontrolador, un element visualitzador i un conjunt de sensors de temperatura. Tal i com s'ha fixat en l'apartat 3.2, el telèfon mòbil haurà de tenir sistema operatiu Android i Bluetooth. La comunicació entre el mòbil i el microcontrolador serà bidireccional i per Bluetooth, fet que introdueix un element transceptor al sistema. A més, el transceptor es comunicarà amb el microcontrolador a través d'un port sèrie del mateix. A la Figura 5.1 es mostra l'esquema de l'aplicació.

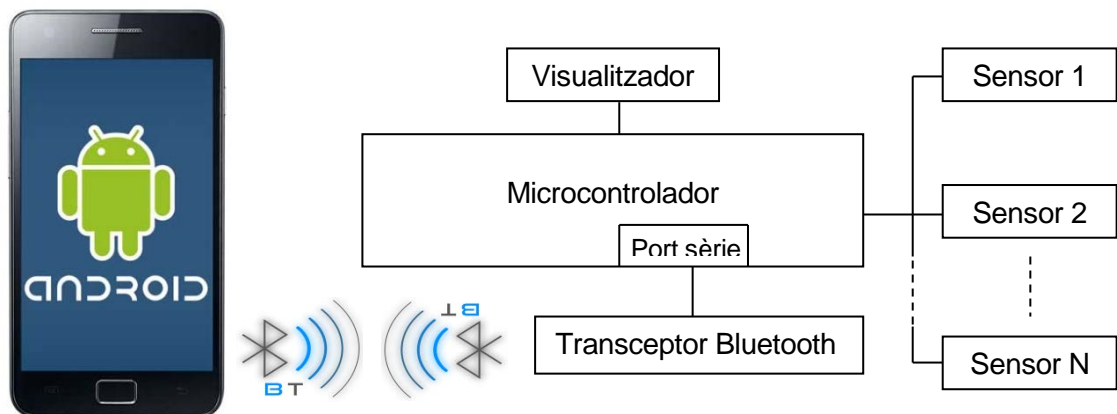


Figura 5.1: Esquema funcional de l'aplicació

5.2. Selecció de components

Un cop definida l'arquitectura funcional del sistema, s'ha de decidir quins components es faran servir per portar a terme cada funció. En el cas del telèfon mòbil, per a fer les proves, s'ha fet servir un mòbil de la marca *HTC*, ja que es disposava del mateix amb anterioritat a la realització del projecte. Un cop finalitzada l'aplicació, qualsevol telèfon amb Android i Bluetooth seria suficient per connectar-s'hi.

Quant als altres components, a l'apartat 3.2 s'indica que s'han d'emprar aquells que estiguin disponibles a la universitat. Aquests components es fan servir a l'assignatura optativa "Desenvolupament d'Aplicacions Basades en Microcontroladors" del GETI i també a l'assignatura troncal del màster "Ampliació d'Electrònica".

Com a microcontrolador es farà servir el model PIC18F4520 de *Microchip Technology Inc.* Un exemplar del mateix es mostra a la Figura 5.2. Juntament amb aquest microcontrolador, es disposa d'una placa de desenvolupament. Es tracta d'una placa de circuit imprès que serveix a l'enginyer com a eina per familiaritzar-se amb el microcontrolador i per aprendre a programar-lo. També serveix als usuaris per a crear prototips en el desenvolupament d'aplicacions. La placa de desenvolupament que es farà servir és l'anomenada *Open18F4520 Development Board*.



Figura 5.2: Microcontrolador PIC18F4520 de Microchip

L'element visualitzador serà una pantalla de cristall líquid o LCD (acrònim anglès de *Liquid Crystal Display*). En concret, es farà servir el model LCD1602 de *Waveshare Electronics*. Aquest model incorpora un controlador LCD de *Hitachi*, el HD44780. A la Figura 5.3 es mostra una imatge de la pantalla. Aquesta pantalla té dues línies de 8 caràcters, de forma que serà necessari incorporar dos botons per escollir quina parella de temperatures es visualitza d'acord amb les especificacions.

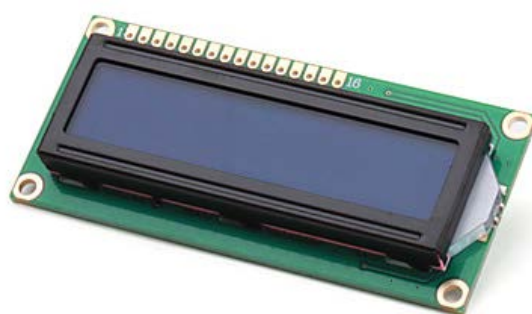


Figura 5.3: LCD1602 de Waveshare Electronics

El sensor de temperatura del qual es disposa és el model DS18B20 de *Maxim Integrated*. Es tracta d'un termòmetre digital. A la Figura 5.4 es pot veure un sensor DS18B20.



Figura 5.4: Termòmetre digital DS18B20 de Maxim Integrated

Per a la comunicació entre el mòbil i el microcontrolador, es disposa de l'accessori *Bluetooth Click Mikroe-958* de *Mikroelectronica*. Aquest model incorpora el transceptor *RN41* de *Roving Networks*. A la *Figura 5.5* es pot veure una imatge del conjunt.



Figura 5.5: Accessori Bluetooth Click Mikroe-958 de Mikroelectronica

6. Característiques dels components

En aquest capítol s'explicaran els detalls més importants dels components escollits per a fer l'aplicació, com ara les seves prestacions, els seus esquemes i com funcionen. Les característiques dels components són molt importants ja que condicionaran la programació de l'aplicació i també el seu funcionament.

6.1. Microcontrolador PIC18F4520

6.1.1. Característiques generals

El microcontrolador que es fa servir en el projecte és el PIC18F4520 de la família PIC de Microchip. Els microcontroladors d'aquesta família es caracteritzen per ser tipus RISC (de l'anglès *Reduced Instruction Set Computer*), de forma que tenen un joc d'instruccions reduït i senzill, són petits i tenen un consum, cost i temps de disseny reduïts. Els microcontroladors de Microchip realitzen una instrucció cada 4 cicles de rellotge.

El model PIC18F4520 té una memòria dades RAM (de l'anglès *Random Access Memory*) de 1536 bytes i una memòria de programa EEPROM (de l'anglès *Electrically Erasable Programmable Read Only Memory*) de 32768 bytes. La memòria RAM és la que fa servir el processador per a emmagatzemar els resultats dels càlculs matemàtics realitzats, mentre que la memòria de programa emmagatzema les instruccions que executarà quan estigui alimentat. És un microcontrolador de 8 bits, amb un comptador de programa de 21 bits, les instruccions estan codificades amb 16 bits, i la seva ALU (de l'anglès *Arithmetic Logic Unit*), amb la que fa les operacions, és de 8 bits. La seva tensió d'alimentació de funcionament pot estar compresa entre 2,0V i 5,5V.

Tots els microcontroladors necessiten d'un senyal de rellotge per poder funcionar. Un senyal de rellotge és un senyal digital binari, habitualment periòdic, que serveix per coordinar les accions de diversos circuits. Aquest senyal pot ser intern o extern al microcontrolador. Durant el procés de programació, uns bits de configuració permeten seleccionar el tipus de rellotge. La característica més important d'aquest senyal és la seva freqüència, donat que indica el ritme amb el que es fan les operacions internes. Les possibilitats existents de senyal de rellotge en el cas del PIC18F4520 són:

- Rellotge intern d'alta freqüència: 8MHz amb possibilitat de divisió per qualsevol nombre parell comprès entre 2 i 64.
- Rellotge intern de baixa freqüència: 31KHz.
- Rellotge extern: 4MHz amb possibilitat de multiplicació per 4 gràcies a un llaç de

seguiment de fase o PLL (de l'anglès *Phase-Locked Loop*), sistema de control que genera un senyal de sortida la fase del qual està relacionada amb la fase del senyal d'entrada.

El microcontrolador incorpora un conjunt de perifèrics, que per definició permeten la integració d'un processador amb el món exterior. Els perifèrics més destacats del microcontrolador, i que s'utilitzen en aquest projecte, són els ports paral·lels d'entrada sortida, els *timers* i un port sèrie de transmissió i recepció de dades: l'USART. Als apartats 6.1.2, 6.1.3 i 6.1.4, respectivament, es fa una descripció detallada d'aquests perifèrics.

A la Figura 6.1 es mostra el diagrama de pins d'aquest microcontrolador. Molts d'aquests pins poden fer diverses funcions. Per software es pot configurar quina funció fa en cada cas.

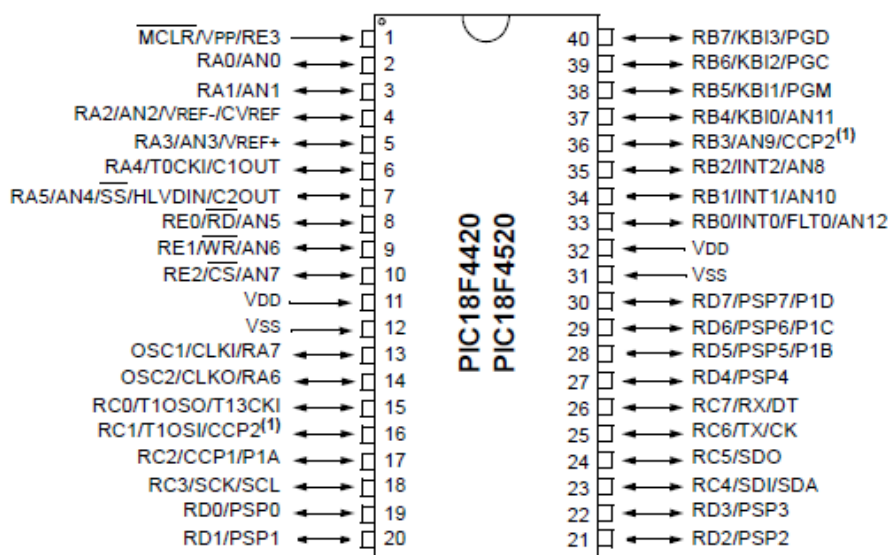


Figura 6.1: Diagrama de pins del PIC18F4520

Juntament amb aquest microcontrolador es disposa també de la seva corresponent placa de desenvolupament. Aquesta placa incorpora hardware que permet la connexió d'altres elements amb el microcontrolador d'una manera ràpida i senzilla i facilita el desenvolupament d'aplicacions. A l'apartat 6.1.5 es fa una descripció d'aquesta placa.

6.1.2. Ports paral·lels

Tot microcontrolador disposa de 1 o més ports paral·lels d'entrada/sortida. Aquests ports permeten llegir dades del món exterior i escriure dades o enviar dades cap al món exterior i

estan associats a un conjunt de pins del microcontrolador.

En el cas del PIC18F4520, es disposa de 5 ports d'entrada/sortida: el PORTA, PORTB, PORTC, PORTD i PORTE, de 8 bits cadascun.

Els ports poden ser de 3 tipus segons el tipus de dades que intercanvien:

- Digitals. Intercanvi de dades digitals.
- Analògics. Intercanvi de dades analògiques. En aquests casos és necessari un conversor d'analògic a digital intern al microcontrolador.
- Mixtos. Es poden configurar bit a bit per software per establir un intercanvi de dades digital o analògic.

Els ports també es poden classificar segons la direcció del flux de dades:

- Entrada. El microcontrolador llegeix dades de l'exterior.
- Sortida. El microcontrolador envia dades cap al món exterior.
- Bidireccionals. La direcció del flux de les dades es pot configurar bit a bit per software.

Els 5 ports del microcontrolador del projecte són bidireccionals, essent el PORTA mixt analògic digital, i la resta digitals.

6.1.3. Timers

Un *timer* és un perifèric que té associat un valor numèric codificat en binari natural amb un conjunt de bits (normalment amb 8 o 16). La seva funció es comptar esdeveniments, incrementant el seu valor numèric en 1 cada cop que un esdeveniment succeeix. Aquests esdeveniments poden ocórrer cada cert temps (periòdics), o poden estar generats per un senyal extern al microcontrolador.

En el projecte només es fa servir un timer del microcontrolador, i per comptar temps. En aquesta modalitat, l'objectiu és obtenir un temporitzador que, a partir del senyal de rellotge de la CPU, mesuri el pas del temps i generi una interrupció periòdica cada cop que arriba al seu valor màxim, posant-se a zero aleshores.

Una interrupció és la indicació d'un esdeveniment aparegut en el hardware o software d'un microcontrolador, que provoca una ruptura de la seqüència d'instruccions que aquest realitza, de forma que el processador passa a executar una rutina especial que pot dependre de l'esdeveniment que hagi generat la interrupció. Després d'executar la rutina

especial, el programa torna al punt on havia estat interromput.

Mitjançant una interrupció periòdica, es pot aconseguir que el microcontrolador executi un conjunt d'instruccions de forma periòdica.

Per configurar el període de les interrupcions, es pot dividir la freqüència del senyal d'entrada al *timer* i es pot establir el valor inicial del mateix, fent que comenci a comptar a partir d'un cert número. A la Figura 6.2 es mostra una fórmula per calcular el període de comptatge del temporitzador.

$$T_{\text{comptatge}} = T_{\text{CLKi}} * N * (\text{MAX} - \text{MIN} + 1)$$

on

TCLKi: Període del senyal de rellotge del microcontrolador

N: Valor pel qual es divideix la freqüència del senyal d'entrada al timer

MAX: valor màxim del timer en decimal ($= 2^n - 1$, on *n* són els bits del timer)

MIN: valor inicial del timer en decimal

Figura 6.2: Fórmula pel càlcul del període de comptatge d'un timer

6.1.4. USART

L'USART (de l'anglès *Universal Asynchronous Receiver Transmitter*) és un perifèric que facilita la transmissió en sèrie de dades, síncrona o asíncrona. Està format per dos blocs independents: un emissor i un receptor, amb els pins TX i RX del microcontrolador, respectivament. Aquests pins també es corresponen a 2 pins del PORTC del PIC18F4520. Permet enviar i rebre dades de forma simultània, tenint els blocs emissor i receptor la mateixa velocitat. La velocitat està generada per un circuit intern a la USART, anomenat generador de *baud rate*. El *baud rate* és el nombre d'unitats de senyal transmeses per segon (en aquest cas cada unitat de senyal és un bit).

En aquest projecte l'USART s'utilitzarà en mode asíncron, de forma que els dos sistemes que es comuniquen entre ells tenen un *baud rate* generat per circuits independents. És necessari, per tal que la comunicació sigui correcta, que les velocitats de transmissió i recepció dels dos sistemes siguin molt semblants, amb un error relatiu màxim d'un 0,5%.

El *baud rate*, en mode asíncron i per al PIC18F4520, es pot configurar per software i depèn del valor decimal emmagatzemat a un registre especial del microcontrolador, el SPBRG (en binari natural), d'un bit que s'anomena BRGH, emmagatzemat a un registre, i de la

freqüència del senyal de rellotge del microcontrolador . Hi ha diferents fórmules per calcular el *baud rate* en funció del valor del bit BRGH i de la mida del registre SPBRG (Taula 6.1).

Bit BRGH	#bits registre SPBRG	Fórmula <i>baud rate</i> (bits/s)
0	8	$F_{osc}/[64 (n+1)]$
1	8	$F_{osc}/[16 (n+1)]$
0	16	
1	16	$F_{osc}/[4 (n+1)]$

F_{osc} =Freqüència del senyal de rellotge (Hz)
n=valor en decimal del registre SPBRGH

Taula 6.1: Fórmules per al càlcul del *baud rate* de l'USART

Per fer els càlculs, primer s'escull un *baud rate* i un senyal de rellotge. Amb aquests valors s'aïlla d'una de les fórmules el valor de n associat. Si el valor de n és enter, es podrà establir sense errors el *baud rate* escollit. Si aquest valor té xifres decimals diferents de 0, s'haurà d'arrodonir al nombre enter més proper, cometent-se un error sobre el *baud rate* desitjat. Per minimitzar l'error, es pot fer aquest càlcul amb les altres fórmules i quedar-se amb aquella que presenta un error més petit; o bé escollir un senyal de rellotge amb una freqüència diferent.

Un altre aspecte important de l'USART és el format en què s'envien les dades. Les dades s'envien en grups de 8 bits o de 9 bits, segons es configuri per software. La tramesa de dades comença amb un bit d'inici, amb valor lògic 0, i continua amb els 8 o 9 bits de dades. Finalment, s'acaba amb un o dos bits de parada, amb nivell lògic 1.

6.1.5. Placa de desenvolupament Open18F4520

La placa de desenvolupament d'aplicacions per al PIC18F4520, anomenada Open18F4520, disposa de connectors per a un conjunt de perifèrics externs, com ara la LCD1602 escollida per al projecte, o els sensors de temperatura DS18B20, entre d'altres. Un dels connectors serveix per connectar un dispositiu que permet programar el microcontrolador. S'ha de tenir

en compte, per a cada perifèric, com està connectat a la placa i a quins ports d'entrada/sortida del microcontrolador està associat. Aquest és un detall molt important, ja que és possible que hi hagi conflicte elèctric entre dos perifèrics diferents que fan servir un mateix port del microcontrolador. A l'apartat 8 s'explicarà com s'han connectat els components del projecte a la placa.

La placa també incorpora dos polsadors. Aquests dos polsadors s'utilitzaran per permetre, davant la limitació que la LCD només permet representar dues temperatures a la vegada, a l'usuari escollir quina parella de temperatures s'han de visualitzar en cada moment.

La placa també conté uns elements, anomenats *jumpers*, que permeten connectar diferents sortides de la placa. Aquests poden fer un simple pont, o bé permeten fer una configuració, com ara la selecció de la tensió a 5V o a 3,3V. En aquest projecte, s'ha treballat a 3,3V degut a les tensions de treball dels components que s'han fet servir. A la Figura 6.3 es pot veure una imatge de la placa.

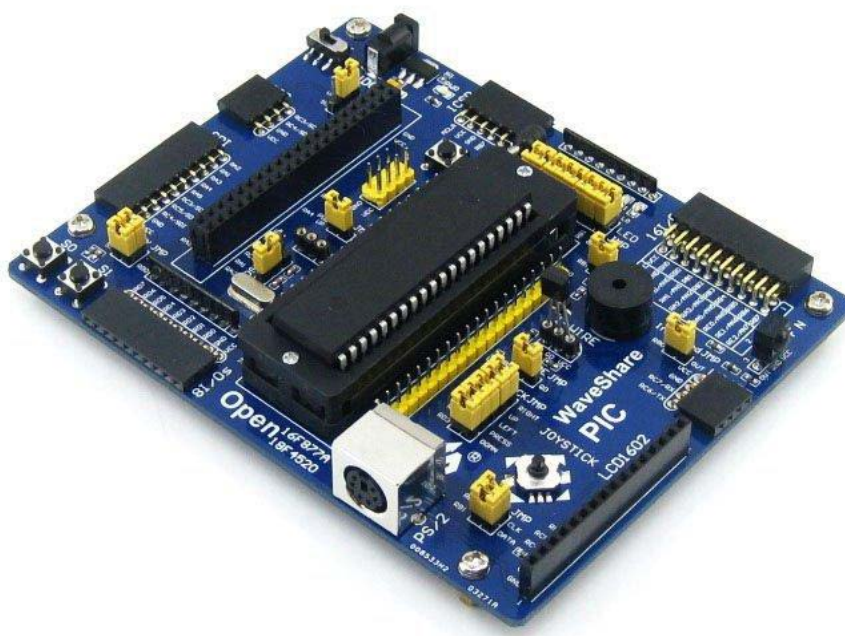


Figura 6.3: Placa de desenvolupament Open18F4520

6.1.6. Eines per a la programació

Al mercat existeixen un conjunt d'eines que permeten desenvolupar el programa i carregar-lo al microcontrolador.

En primer lloc, es disposa del programa MPLAB, destinat a productes de la marca Microchip. Es tracta d'un editor IDE (de l'anglès *Integrated Development Environment*) ja que

proporciona serveis integrals per facilitar el desenvolupament del software al programador, i el seu entorn conté editor de text, simulador, depurador, ensamblador, editor d'enllaços i compilador. L'editor de text és el que es fa servir per a escriure el programa en llenguatge de programació d'alt nivell, en aquest cas en llenguatge C. El simulador i depurador permeten provar el programa, detectar errors i corregir-los. La resta d'eines serveixen per convertir el programa escrit en llenguatge C en un arxiu que conté zeros i uns i que pot ser carregat al microcontrolador amb ajuda d'un programador.

Per carregar el programa al microcontrolador, s'ha fet servir el programador MPLAB ICD 2. Aquest model és connecta a l'ordinador per via USB i a la placa de desenvolupament del PIC18F4520 mitjançant un connector especial. Després de fer aquesta connexió, es pot programar el microcontrolador des del programa MPLAB IDE. A la Figura 6.4 es mostra una imatge del programador.



Figura 6.4: Programador MPLAB ICD 2 de Microchip

6.2. Sensor de temperatura DS18B20

6.2.1. Característiques generals

El sensor de temperatura DS18B20 de *Maxim Integrated* proporciona mesures de temperatura en graus Celsius en el rang comprès entre -55°C i $+125^{\circ}\text{C}$ i només presenta tres pins: VCC, GND i DQ. A la Figura 6.5 es mostra el diagrama de pins del sensor. La seva tensió d'alimentació pot oscil·lar entre 3,0V i 5,5V i es pot establir entre els seus pins GND i VCC, o bé es pot alimentar des de la mateixa línia de dades DQ.

La precisió de la temperatura és de mig grau i es pot garantir per a mesures fetes entre -10°C i $+85^{\circ}\text{C}$, complint així les especificacions del projecte. La resolució és seleccionable per l'usuari i va de 9 a 12 bits. A la Taula 6.2 es pot veure el temps màxim de conversió de la temperatura per a cada resolució.

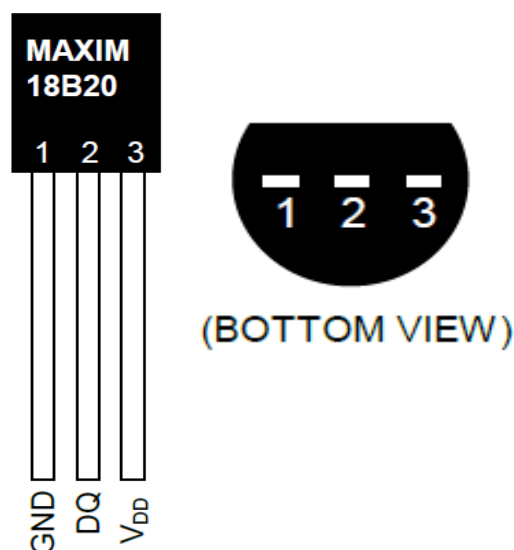


Figura 6.5: Diagrama de pins del sensor

Resolució (bits)	Resolució ($^{\circ}\text{C}$)	R1	R0	Temps màxim de conversió (ms)
9	0,5000	0	0	93,75
10	0,2500	0	1	187,5
11	0,1250	1	0	375,0
12	0,0625	1	1	750,0

Taula 6.2 Temps de conversió i bits R1 i R0 del registre de configuració per a les diferents resolucions

El sensor conté memòria volàtil del tipus RAM estàtica, anomenada *scratchpad*, formada per 9 registres de 8 bits cadascun. Dos d'aquests bytes s'utilitzen per emmagatzemar la temperatura, que està codificada en complement a 2 en unitats de $1/16^{\circ}\text{C}$. A la Figura 6.6 es mostren aquests dos bytes, essent LS BYTE el byte menys significatiu i MS BYTE el més

significatiu. 5 dels bits del MS BYTE es fan servir per indicar el signe de la temperatura i la resta tenen associats els pes d'una potència de 2 de °C.

	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
LS BYTE	2^3	2^2	2^1	2^0	2^{-1}	2^{-2}	2^{-3}	2^{-4}
	BIT 15	BIT 14	BIT 13	BIT 12	BIT 11	BIT 10	BIT 9	BIT 8
MS BYTE	S	S	S	S	S	2^6	2^5	2^4

S = SIGN

Figura 6.6: Registres del sensor que contenen la temperatura

El DS18B20 també té una memòria EEPROM amb 3 bytes que serveixen per configurar paràmetres de funcionament del sensor i que es copien automàticament a 3 bytes del *scratchpad*. Per a aquest projecte, és demana tenir una resolució mínima d'una dècima de grau, així doncs serà necessari poder regular-la. Això es fa en un registre anomenat *Configuration Register*, els detalls del qual es mostren a la Figura 6.7. Els bits R1 i R0 són els que determinen, segons els seus valors, la resolució de la temperatura. A la Taula 6.1 es pot veure quina resolució està associada a cada combinació de valors de R0 i R1.

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
0	R1	R0	1	1	1	1	1

Figura 6.7: Registre de configuració de la resolució del sensor

Cada sensor té un codi ROM únic de 64 bits, que permet a una unitat central (en aquest cas el microcontrolador) distingir els diferents sensors i comunicar-se amb ells de forma independent.

Per controlar el funcionament dels sensors, es fa servir un bus de comunicació anomenat *1-Wire*. A l'apartat 6.2.2 es fa una descripció del bus *1-Wire*. Un bus de comunicació és un medi compartit de comunicació constituït per un conjunt de línies (conductors) que connecta diferents unitats en un sistema digital i permet la transferència de dades a través de senyals elèctrics. El bus *1-Wire* permetrà al microcontrolador enviar ordres als diferents sensors. Aquestes ordres estan codificades en 1 byte i tenen com a objectiu iniciar la conversió de la temperatura, llegir el *scratchpad* o escriure en el *Configuration Register*, entre d'altres.

6.2.2. Bus de comunicació 1-Wire

El bus *1-Wire* està basat en un bus, un *mestre* i diversos *esclaus* connectats a una única línia de dades. El mestre és el dispositiu que envia ordres i el que té el control del bus, mentre que els esclaus són els dispositius que actuen en funció del que el mestre mana. La línia de dades/alimentació treballa amb senyals en drenador obert que proporcionen el nivell 0V (0 lògic) i requereix una resistència de *pull-up* connectada a l'alimentació que permet elevar la tensió de la mateixa per subministrar el 1 lògic. Aquest bus permet operar entre 2,8V i 6V.

La transmissió de dades es realitza en sèrie, bit rere bit, per aquesta única línia de dades, a una taxa de transmissió d'un bit cada 70 microsegons. Totes les ranures de temps, tant de lectura com d'escriptura d'un bit, són de 70 microsegons.

El primer pas per iniciar una comunicació és reiniciar el bus. Per fer-ho, el mestre manté la línia de dades a 0V durant 480 microsegons. Amb això, es reinicien tots els dispositius connectats al bus. Els dispositius reiniciats indiquen la seva presència mantenint el senyal de dades a 0 entre 60 i 240 microsegons.

Per enviar un bit a 1 el mestre baixa a 0V la línia de dades durant 1-15 microsegons i, després, deixa anar el bus i espera el temps necessari per ocupar tota una ranura de 70 microsegons. Per enviar un bit a 0 el mestre baixa a 0V la línia de dades durant 60 microsegons i, després de deixar el bus, espera durant 10 microsegons. Els dispositius esclaus llegeixen el bit aproximadament als 30 microsegons després del flanc de baixada de cada bit.

Quan el mestre vol llegir les dades del dispositiu esclau, posa 0V durant 6 microsegons a la línia de dades i després deixa anar el bus, esperant durant 9 microsegons. Durant l'espera l'esclau pot transmetre un 0 o un 1 lògics. Si vol enviar un 1 lògic manté el bus a nivell alt de tensió i si vol enviar un 0 lògic manté el bus a nivell baix. En passar aquest temps, el mestre mostreja el bus per llegir el bit en qüestió i espera durant 55 microsegons. L'operativa per aquestes operacions es resumeix a la Taula 6.3.

Operació	Descripció	Implementació
Reset	Reinicia els esclaus del bus 1-Wire i els deixa llestos per a rebre ordres.	Portar el bus a 0, espera de 480 μ s Deixar anar el bus, espera de 70 μ s Mostrejar el bus: 0=dispositius presents 1=no hi ha dispositiu
Escriure un bit a 0	Envia un bit '0' als esclaus del 1-Wire	Portar el bus a 0, espera de 60 μ s Deixar anar el bus, espera de 10 μ s
Escriure un bit a 1	Envia un bit '1' als esclaus del 1-Wire	Portar el bus a 0, espera de 6 μ s Deixar anar el bus, espera de 64 μ s
Llegir un bit	Llegeix un bit dels esclaus del 1-Wire	Portar el bus a 0, espera de 6 μ s Deixar anar el bus, espera de 9 μ s Mostrejar el bus per a llegir el bit espera de 55 μ s

Taula 6.3: Operacions del bus 1-Wire

Degut a la manera en què s'implementen aquestes operacions, el mestre haurà de ser capaç de generar un retard de 1 microsegon, precís i repetitiu. A més, el seu port de comunicació haurà de ser bidireccional i amb sortida del tipus *Open-Drain* per poder posar la resistència de pull-up que es necessita.

Les dades s'envien o reben en grups de 8 bits (bytes). El protocol pot incloure detecció d'errors transmetent codis de detecció d'errors. Com en el bus poden haver-hi molts dispositius, el protocol inclou l'adreçament dels mateixos emprant un codi únic de 64 bits, dels quals el byte més significatiu indica el tipus de dispositiu, i l'últim és un codi de detecció d'errors de 8 bits. Les comandes que puguin interpretar els esclaus depenen d'aquests.

Microchip proporciona un arxiu en llenguatge C que conté un conjunt de funcions que realitzen les operacions del bus 1-Wire. Aquest arxiu existeix per a la versió dels microcontroladors tipus PIC18, i les funcions estan pensades per a una freqüència del senyal de rellotge del microcontrolador de 4MHz. Si es fes servir una freqüència diferent, s'hauran de modificar les funcions.

6.3. Pantalla LCD1602

La pantalla de la LCD1602 de Waveshare Electronics està formada per dues línies que permeten representar 16 caràcters cadascuna. El controlador que incorpora permet representar símbols alfanumèrics i altres tipus de símbols. El seu consum és reduït gràcies a

la tecnologia CMOS (de l'anglès Complementary Metal-Oxide-Semiconductor), un tipus de tecnologia de construcció de circuits integrats que assegura l'estalvi d'energia de la manera més eficient sense perdre prestacions. El rang de temperatures en funcionament és de 0 a +50°C, i la tensió d'alimentació ha de ser de 3,3V.

PIN NO.	SÍMBOL	DESCRIPCIÓ	FUNCIÓ
1	VSS	Terra	0V (GND)
2	VCC	Alimentació pel circuit lògic	+3,3V
3	V0	Ajust del contrast de la LCD	Es recomana potenciòmetre de 10K entre V0 i GND
4	RS	Selecció de registre de dades/instruccions	RS=0: Registre d'instruccions RS=1: Registre de dades
5	R/W	Selecció Lectura/Escriptura	R/W=0: Escriptura Registre R/W=1 Lectura Registre
6	E	Senyal d'habilitació	Actiu quan el nivell alt passa a nivell baix
7	D0	Línies d'entrada/sortida de dades	8 bits: D0-D7
8	D1		
9	D2		
10	D3		
11	D4		
12	D5		
13	D6		
14	D7		
15	A	Subministrament de tensió per LED+	+3,3V
16	K	Subministrament de tensió per LED-	0V (GND)

Taula 6.4: Característiques dels pins de la LCD1602

La LCD té 16 pins que permeten controlar-la. A la Taula 6.4 es mostren els detalls de cada un d'ells. Els pins D0 a D7 són les línies que permeten llegir o escriure dades en la LCD. Es poden fer servir els 8, essent més ràpida la transferència de dades, o bé es poden fer servir només 4. A part dels pins de l'alimentació, també té pins que permeten ajustar la nitidesa i el contrast de la pantalla. El pin E és un pin d'habilitació i serveix per controlar la transferència de dades. El pin R/W és el que indica si s'està llegint o escrivint i el pin RS indica si es tracta d'una dada o d'una instrucció.

Per controlar la LCD, existeix un arxiu en llenguatge C del professor de l'ETSEIB Manuel Moreno Eguilaz, que conté funcions que permeten inicialitzar la LCD, esborrar els caràcters que puguin estar representats a la pantalla, situar el cursor d'escriptura a un punt concret de la pantalla o escriure un caràcter a la LCD, entre d'altres. Aquesta llibreria és vàlida per al PIC18F4520 operant a una freqüència de senyal de rellotge de 4MHz. Si es fes servir una freqüència diferent, s'haurien de modificar les funcions.

6.4. Transceptor Bluetooth Click Mikroe-958

6.4.1. Característiques generals

El Bluetooth Click Mikroe-958 de Mikroelektronika és un accessori en forma de targeta que permet emetre i rebre dades via Bluetooth a través d'un port sèrie tipus UART, que és com un USART però només amb comunicació asíncrona. Està dissenyat per ser alimentat a una tensió de 3,3V. Admet un ritme de transmissió de dades superior a 3Mbps (Megabits per segon) i pot operar amb altres dispositius Bluetooth a distàncies superiors a 100m.

A la Figura 6.8 es mostra un diagrama del circuit d'aquest accessori, mentre que a la Figura 5.5 s'ha mostrat una vista superior del mateix. Els pins més importants són els +3,3V i GND per establir l'alimentació, i els pins TX i RX de transmissió i recepció de dades des del punt de vista del mòdul, respectivament. Al diagrama del circuit també es pot veure el mòdul de radiofreqüència RN-41 que incorpora. Per configurar el Bluetooth-Click s'ha de configurar el mòdul RN-41. A l'apartat 6.4.2 s'expliquen les característiques més importants d'aquest mòdul. També es pot apreciar que hi ha uns elements connectors o *jumpers* que estan inicialment sense connectar però que mitjançant una soldadura es podrien connectar. Es tracta de connectors de configuració que permeten posar un nivell alt de tensió en els pins GPI3, GPI4, GPI6, GPI7, RTS i CTS del mòdul RN41.

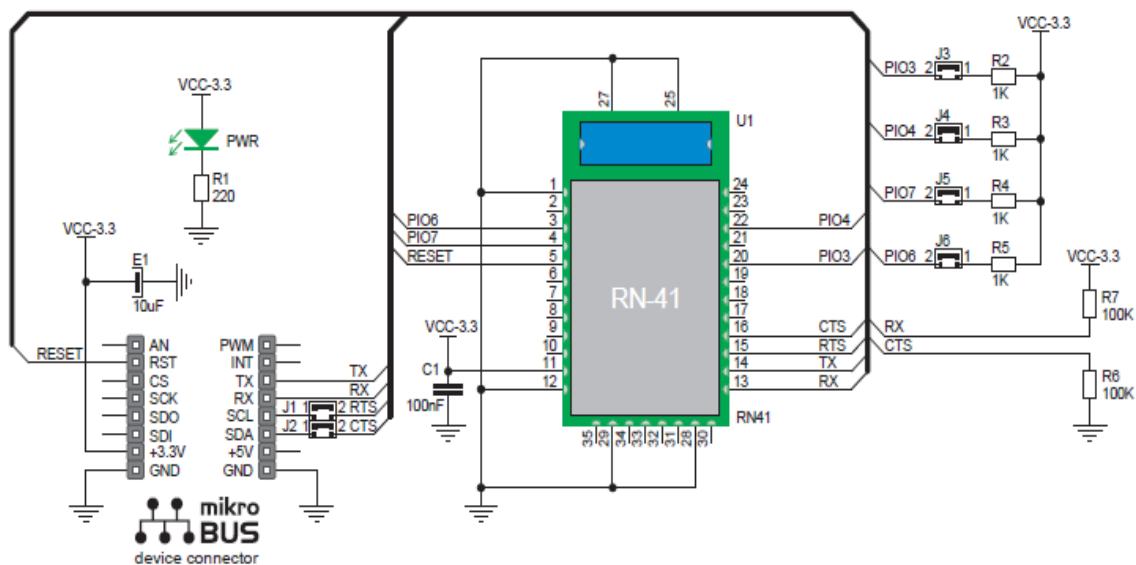


Figura 6.8: Diagrama del circuit del Bluetooth Click

6.4.2. Mòdul RN41

El dispositiu RN41 és un mòdul de Bluetooth de la marca Microchip que pot operar en dos modes: el mode de dades i el de comandes. El mode de comandes serveix per configurar el dispositiu a través del mòdul UART fent servir comandes en codi estàndard ASCII (de l'anglès *American Standard Code for Information Interchange*). En el mode de dades, quan el mòdul rep una dada via Bluetooth, la adequa i la passa al port UART, i quan rep una dada per l'UART, el mòdul construeix el paquet Bluetooth i el transmet (per Bluetooth). A la Figura 6.9 es mostra una imatge i l'esquemàtic del mateix.

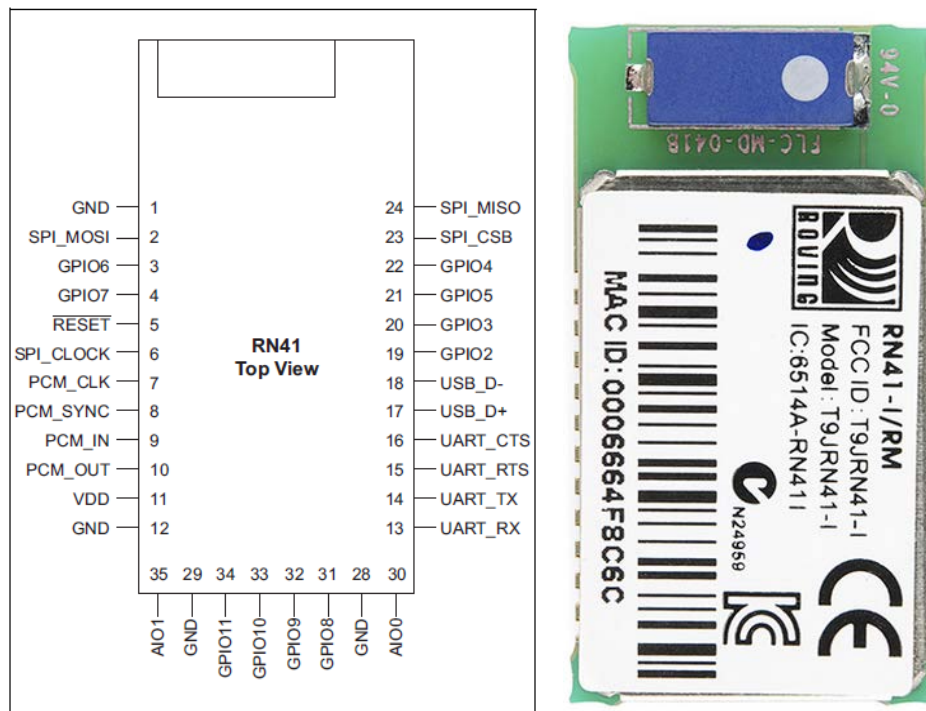


Figura 6.9: Diagrama de pins i imatge del mòdul RN41 de Microchip

La configuració de fàbrica del mòdul Bluetooth és:

- Mode esclau de Bluetooth.
- Codi pin d'aparellament 1234.
- Port sèrie (UART): *baud rate* de 115200 bits/s, paraules de 8 bits, sense bit de paritat per a la detecció d'errors, i amb 1 bit de parada.
- Control de flux del port sèrie desactivat.

És possible canviar aquests paràmetres de funcionament, de manera que els canvis aplicats constitueixin una configuració per defecte en el moment de l'encesa del dispositiu. Això es pot aconseguir desenvolupant un programa per al PIC18F4520 per tal que el microcontrolador comuniqui al mòdul de radiofreqüència amb quins paràmetres es vol que funcioni. Només és necessari executar el programa un cop. En aquest projecte s'ha de modificar la configuració del dispositiu, en concret canviar-ne el *baud rate* per tal d'adequar-lo al de l'USART, ja que la comunicació serà asíncrona. A l'apartat 9 es justificarà i s'explicarà el *baud rate* escollit.

L'intercanvi de dades entre el microcontrolador i el Bluetooth Click, en el programa de configuració, hauria de ser el següent: en primer lloc, el microcontrolador ha d'enviar "\$\$\$" en ASCII per l'UART entre els primers 60 segons després de l'encesa per tal de fer entrar al mòdul de Bluetooth en mode de comandes. Fet això, el mòdul retorna "CMD" si la connexió és correcta. Un cop en aquest mode, si s'envia una comanda, el mòdul retorna "AOK" si la comanda és vàlida, "ERR" si la comanda no és vàlida i "?" per a comandes no reconegudes. Les comandes de configuració no s'apliquen fins que es reinicia el mòdul i, un cop aplicades, es desen fins que són explícitament canviades o fins que les característiques de fàbrica són restaurades. Els paràmetres que es poden modificar són el nom del dispositiu, el baud rate, el mode mestre/esclau, el tipus d'autenticació per a la connexió i el codi pin del dispositiu, entre d'altres.

També es poden fer ajustos del mòdul per via hardware a través dels pins GPI3, GPI4, GPI6, GPI7. El pin GPI3, si es posa a "1" lògic, habilita la detecció automàtica. El GPI4 permet restablir les característiques de fàbrica. El GPI6 serveix per posar el mode mestre automàtic i el GPI7 ajusta el baud rate. Si es posa un "1" lògic, el baud rate quedarà fixat a 9600 bits/s independentment del que s'hagi establert per software. Si es posa un "0", El baud rate serà de 115.000 bits/s, o bé el que es configuri per software.

7. Metodologia

En aquest capítol s'explicaran els passos que s'han seguit per construir l'aplicació i les eines que s'han utilitzat.

El projecte s'ha desenvolupat per fases, de manera seqüencial. Per a cada component situat en la placa de desenvolupament, s'ha escrit un codi en llenguatge C dins l'entorn de desenvolupament MPLAB i s'ha provat el seu funcionament.

S'ha començat pel LCD1602, fent que representi un conjunt de temperatures arbitràries seguint les especificacions. Tot seguit s'ha implementat l'ús dels dos pulsadors de la placa per modificar la parella de temperatures que es representen a la pantalla. Després de desenvolupar-ne el codi corresponent, s'ha escrit el codi per al control dels sensors de temperatura. Primer s'ha escrit un codi per controlar un únic sensor, i posteriorment s'ha ampliat per a múltiples sensors. Acte seguit s'ha dissenyat la part del programa on es transmeten i es reben dades per l'USART. Per comprovar aquesta part, s'ha fet servir un dispositiu que permet veure les dades enviades per l'USART en un ordinador personal.

Després d'haver obtingut tot aquest bloc de codi, s'ha desenvolupat un programa a part que permet programar l'accessori de radiofreqüència Bluetooth Click en una configuració concreta.

L'últim component que s'ha programat és el telèfon mòbil. Per construir l'aplicació, s'ha utilitzat una plataforma de disseny d'aplicacions per telèfons mòbils amb Android anomenada MIT App Inventor. Aquesta plataforma conté blocs de codi que faciliten el disseny d'aplicacions i permet provar les aplicacions a mesura que es construeixen. A l'apartat 10.1 se'n fa una descripció més detallada.

Un cop dissenyades les parts de programa que controlen cada component, s'ha agrupat tot, fent les modificacions sobre software o hardware necessàries per tal que l'aplicació funcioni complint les especificacions del projecte.

8. Configuració del hardware

En aquest capítol s'explicarà com estan connectats els sensors de temperatura, la pantalla LCD i l'accessori Bluetooth Click a la placa de desenvolupament del PIC18F4520. Per a cada component, s'explicarà quins pins del microcontrolador s'han fet servir per al seu control i quins muntatges s'han fet per evitar conflictes elèctrics.

8.1. Connexió del Bluetooth Click

El mòdul Bluetooth Click converteix dades de Bluetooth a port sèrie USART i a l'inrevés. Així doncs, aquest ha d'anar connectat a l'USART del PIC18F4520.

La placa de desenvolupament té un connector per a l'USART (Figura 8.1) amb 4 contactes. Dos dels contactes són per donar alimentació al dispositiu que es connecti, i els altres dos corresponen als pins RC6/TX i RC7/RX de transmissió i recepció, respectivament, des del punt de vista del microcontrolador.

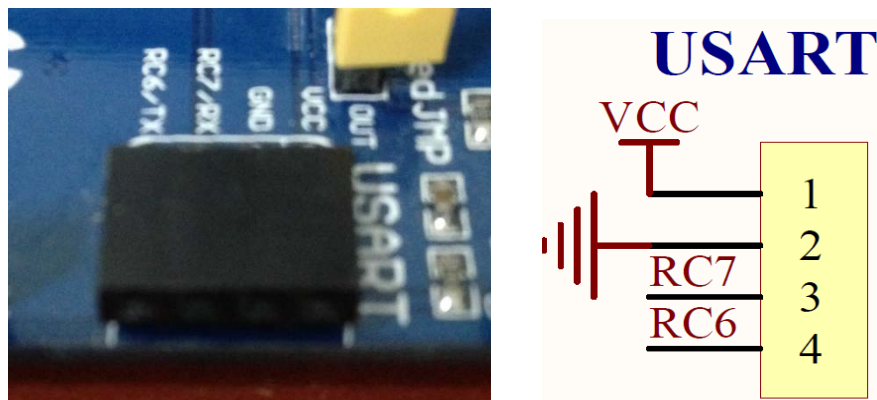


Figura 8.1: Connector per la USART de la placa Open18F4520

Les connexions que s'han de fer són les següents: el pin VCC (pol positiu alimentació) de l'accessori s'ha de connectar amb el contacte de VCC corresponent de la placa, i el mateix cal fer amb el pin GND (pol negatiu alimentació) per alimentar el dispositiu a 3,3V. El pin de transmissió TX del Bluetooth Click es connectarà al pin de recepció RC7/RX del PIC18F4520 i el pin de recepció RX al pin de transmissió RC6/TX del PIC18F4520. Això es fa perquè el que transmet el mòdul (cap al microcontrolador) és el que rep el microcontrolador i viceversa.

Per fer l'enllaç, s'ha construït una mànega que sigui compatible amb el connector de la placa

de desenvolupament i respecti les connexions anteriors. A la Figura 8.2 es mostra el detall del muntatge. Cada cable de la mànega té un color diferent per evitar confusions a l'hora de fer la connexió. El cable vermell és el de l'alimentació, el marró el de terra, el groc el corresponent al pin TX del Bluetooth Click i el taronja el corresponent al pin RX, també del mòdul de Bluetooth.

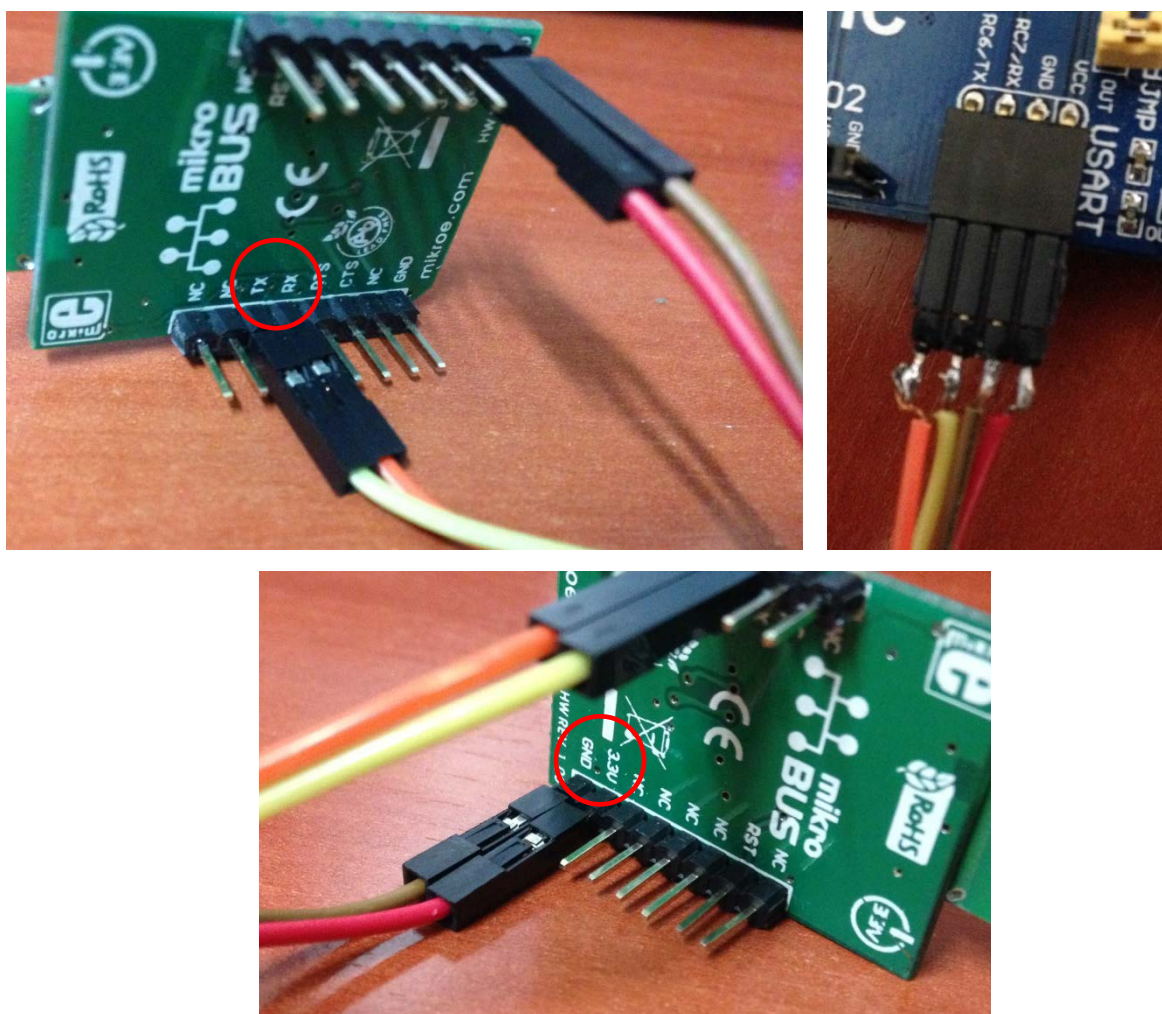


Figura 8.2: Muntatge per connectar el Bluetooth Click al connector USART

8.2. Connexió de la LCD1602

A la placa Open18F4520 hi ha un connector pensat específicament per al LCD1602. A la Figura 8.3 es mostra aquest connector i, per als pins del 4 fins el 14 de la LCD, a quin port d'entrada/sortida del microcontrolador estan associats. Com es pot observar, el pin RC6/TX del PIC18F4520 el fa servir el pin E de la LCD com a port d'entrada/sortida RC6, però també

el fa servir mòdul Bluetooth Click, actuant el pin del microcontrolador en aquest darrer cas com a TX de l'USART.

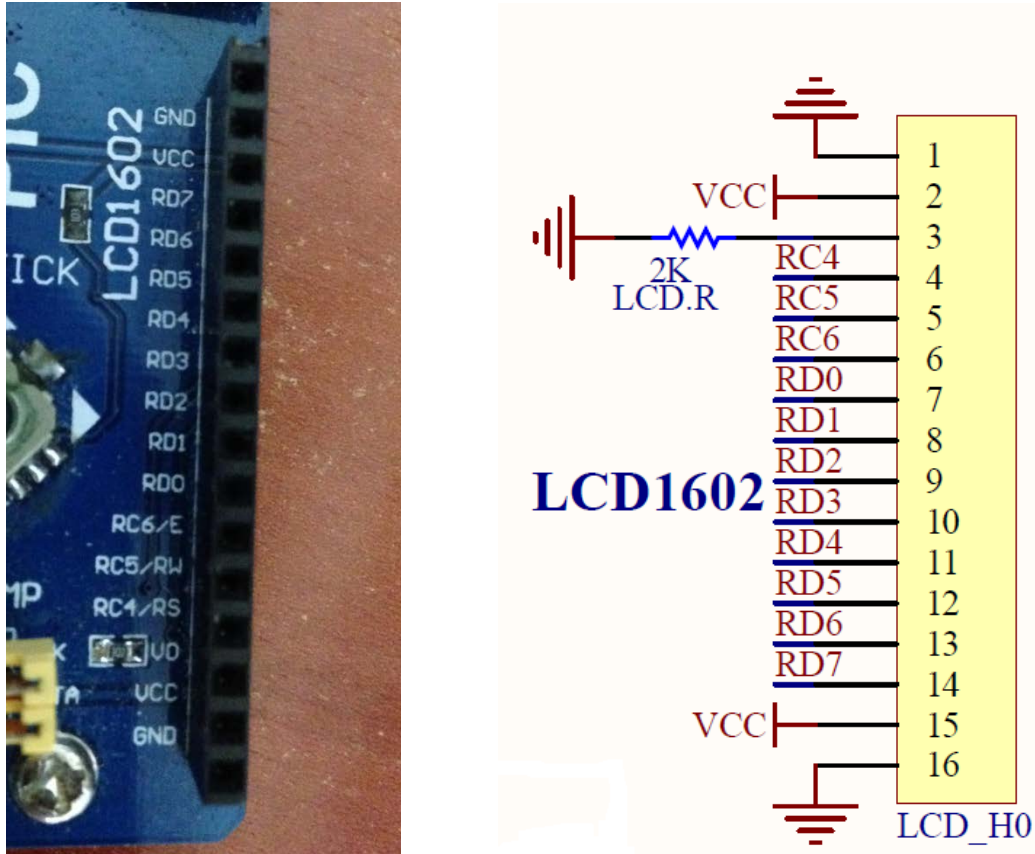


Figura 8.3: Connector per la LCD1602 de la placa Open18F4520

El mateix pin del microcontrolador no pot ser port d'entrada/sortida RC6 i port TX de l'USART a la vegada. Així doncs, s'ha de fer servir un altre pin per algun dels dos components.

L'accessori de Bluetooth no es pot connectar a un port diferent, ja que els pins TX i RX de l'USART estan associats a dos pins concrets del microcontrolador. Així doncs, s'ha d'escollir un port d'entrada/sortida diferent per al pin E del LCD.

Per solucionar el conflicte, s'ha decidit fer servir només els pins D7, D6, D5 i D4 de transmissió de dades del LCD, quedant lliures els ports corresponents als pins D3, D2, D1 i D0. El pin E aleshores s'ha desviat al port RD3 del PIC18F4520, on abans anava el pin D3. A la Figura 8.4 es mostra una imatge del muntatge final. Aquesta modificació fa que la transmissió de dades del microcontrolador a la LCD sigui més lenta, però a l'hora de representar les temperatures no es nota el canvi.

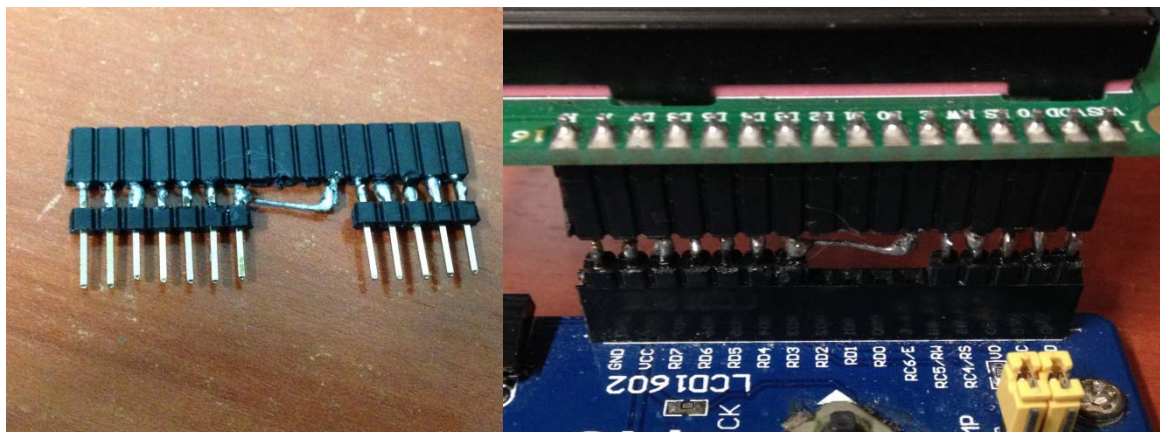


Figura 8.4: Muntatge per traslladar el pin E de la LCD al port RD3

Amb la modificació feta, dos dels ports que queden lliures, el RD0 i el RD1, també estan associats als dos pulsadors de la placa de desenvolupament. A la Figura 8.5 se'n mostra una imatge. Aquests pulsadors es podran fer servir, mentre el programa estigui en funcionament, per escollir quina parella de temperatures es vol veure per la pantalla en cada moment.

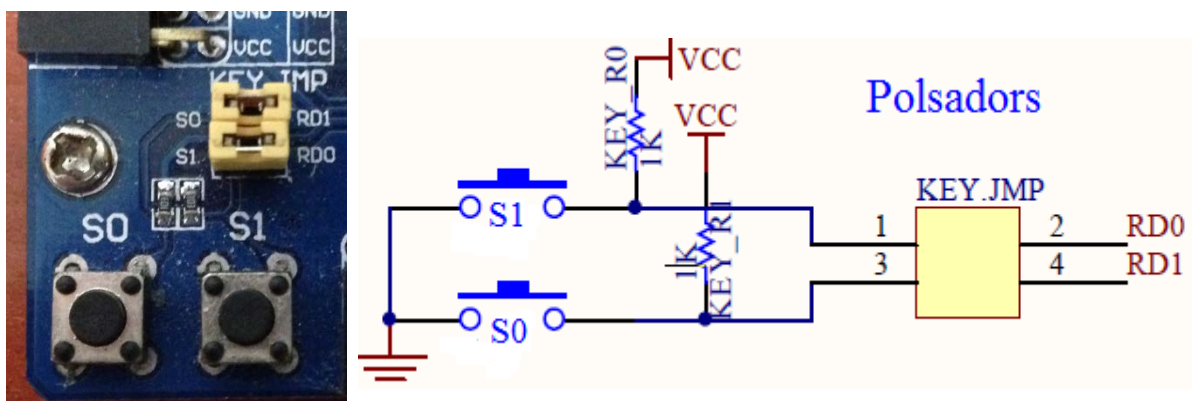


Figura 8.5: Pulsadors de la placa de desenvolupament

8.3. Connexió dels sensors de temperatura

La placa de desenvolupament té un connector de tres contactes que permet connectar un únic sensor de temperatura. Cada contacte està associat a un pin del sensor: VCC, GND i

DQ. La línia de dades DQ del sensor està associada al port RB0 del microcontrolador i no presenta conflictes elèctrics amb cap altre component que s'utilitzi en el projecte. El connector també incorpora la resistència de pull-up que el bus de comunicacions 1-Wire requereix.

Donades les especificacions, és necessari poder connectar més sensors (fins a 8). Per fer-ho, tenint en compte les característiques del bus 1-Wire, només cal fer un muntatge per tal que als contactes del connector es puguin connectar més sensors.

A l'hora de fer el muntatge, s'ha de tenir en compte la ubicació on es voldran situar els diferents sensors. Per fer les proves no hi ha hagut cap preferència i el muntatge que s'ha fet consisteix en dues plaques soldades per darrere, a cada una de les quals es poden situar fins a 8 sensors. Aquestes plaques estan connectades entre sí mitjançant 3 cables (VCC, GND i DQ) i també estan connectades al connector d'un sensor de la placa. A la Figura 8.6 es pot veure una imatge del muntatge.

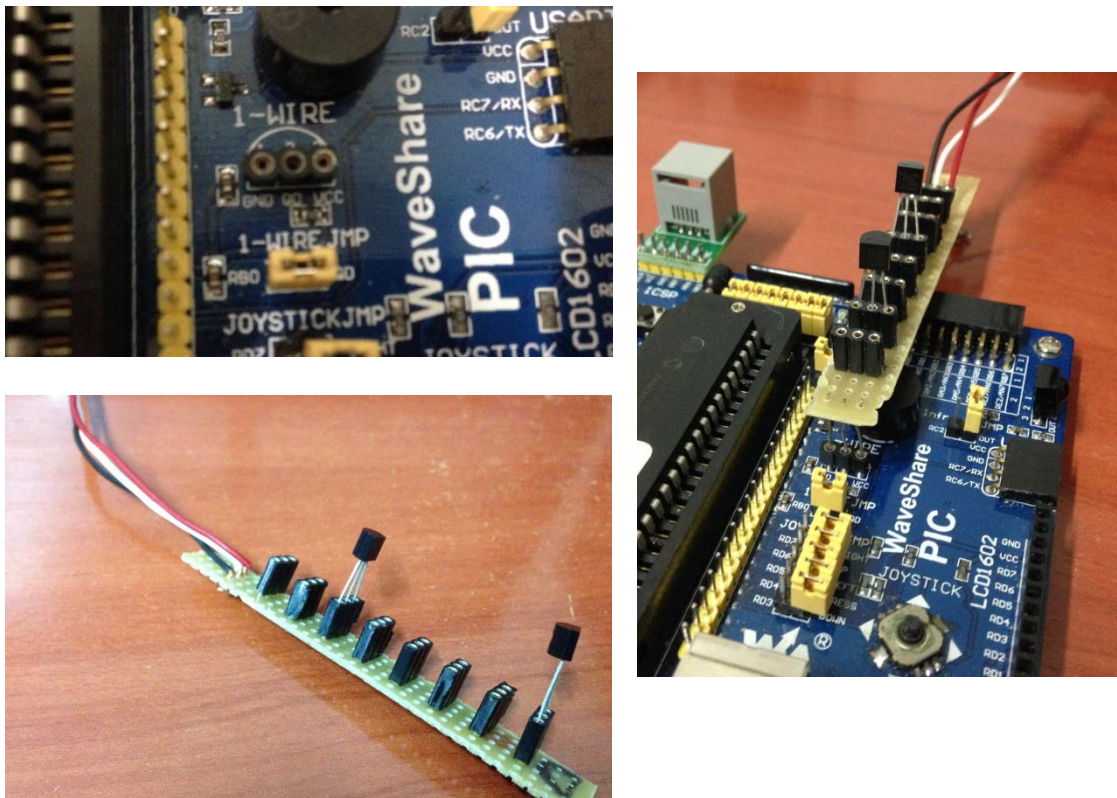


Figura 8.6: Muntatge per connectar més d'un sensor al bus 1-Wire

9. Comunicació entre PIC18F4520, Bluetooth Click i telèfon mòbil

En aquest apartat s'explicarà com s'ha organitzat la comunicació entre el microcontrolador i el telèfon mòbil, usant com a element transceptor el mòdul Bluetooth Click. Primer s'explicarà el càlcul del *baud rate*, i després el format dels missatges que s'envien bidireccionalment.

9.1. Configuració del *baud rate* del microcontrolador i del Bluetooth Click

El *baud rate* de l'USART del microcontrolador i el del Bluetooth Click s'ajusten independentment ja que la comunicació és asíncrona, i s'ha d'intentar que els dos siguin el més semblants possibles per al bon funcionament de l'aplicació.

El Bluetooth Click pot configurar-se amb un conjunt finit de *baud rates* (en bits/s): 921.600, 460.800, 230.400, 115.200, 57.600, 38.400, 19.200, 9.600, 4.800, 2.400 i 1.200. L'USART del PIC18F4520, en canvi, pot ajustar-se més acuradament amb el procediment descrit a l'apartat 6.1.4.

El que s'ha fet és intentar ajustar l'USART del microcontrolador a la velocitat per defecte del Bluetooth Click, 115.200 bits/s. A la Taula 9.1 es mostren els càlculs fets (basant-se en la Taula 6.1), tots ells amb el bit BRGH a 1 ja que és l'opció que dóna resultats més propers en aquest cas. Com es pot apreciar, no és possible tenir un bon funcionament a 115.200 ja que suposaria un error mínim d'un $0,83\% > 0,5\%$.

El pròxim pas és, doncs, provar amb altres *baud rates*. S'ha provat amb 57.600 i 38.400 bits/s. Els resultats també es mostren a la Taula 9.1. Hi ha dues configuracions, corresponents al *baud rate* de 38400 bits/s, que donen un error molt reduït ($0,16\% < 0,5\%$). S'ha escollit aquella que utilitza el registre SPBRG de 8 bits.

Això implica que el sistema funcioni amb el senyal de rellotge extern del microcontrolador i amb PLL, a 16MHz de freqüència, i que la comunicació entre el microcontrolador i el Bluetooth Click sigui a 38400 bits/s.

Paràmetres d'entrada				Resultats		
Baud rate desitjat	F _{OSC} (MHz)	Bit BRGH	#bits registre SPBRG	Valor obtingut de n	Valor a introduir en SPBRG	Error comés (%)
115.200	4	1	8	1,17	1	14,53
115.200	16	1	8	7,68	8	4,16
115.200	16	1	16	33,72	34	0,83
57.600	4	1	8	3,34	3	10,18
57.600	16	1	8	16,36	16	2,20
57.600	16	1	16	68,44	68,00	0,64
38.400	4	1	8	5,51	6	8,88
38.400	16	1	8	25,04	25	0,16
38.400	16	1	16	103,17	103	0,16

Taula 9.1: Càlculs fets per ajustar el baud rate de l'USART del PIC18F4520

9.2. Format de les dades

En aquest apartat es descriurà el format de les dades que el telèfon mòbil pot enviar al microcontrolador, i el de les dades que envia el microcontrolador al telèfon mòbil.

Els missatges que pot enviar el telèfon mòbil al microcontrolador, amb els camps que contenen i els possibles valors, es mostren a la Taula 9.2. Tots els camps es codifiquen amb un byte i en complement a 2. Això és degut a que el software per al disseny de l'aplicació del telèfon mòbil, el MIT App Inventor, no permet dividir un byte de dades en diversos bits, ni tampoc permet transmetre o llegir dades bit a bit.

En el cas de la transmissió de dades del microcontrolador al telèfon mòbil, només hi ha un tipus de missatge: la transmissió de la temperatura d'un sensor, sigui del tipus que sigui. A la Taula 9.3 es mostra el format d'aquest missatge. Tots els camps han estat codificats amb un

byte i en complement a 2 per la mateixa raó d'abans.

Missatge	Camps
Rebre la temperatura instantània per tots els sensors	Número especial [-99]
Escollir tipus de temperatura que es vol visualitzar al telèfon mòbil (un sensor)	Número del sensor [1...8]
	Tipus de missatge [2=escollir tipus de temperatura]
	Tipus de temperatura [0=instantània, 1=màxima, 2=mínima, 3=límit superior, 4=límit inferior]
Reiniciar temperatura màxima (un sensor)	Número del sensor [1...8]
	Tipus de missatge [3=reiniciar temperatura màxima]
Reiniciar temperatura mínima (un sensor)	Número del sensor [1...8]
	Tipus de missatge [4=reiniciar temperatura mínima]
Establir límit superior de la temperatura (un sensor)	Número del sensor [1...8]
	Tipus de missatge [0=establir límit superior]
	Part entera de la temperatura [-128...127]
	Part decimal de la temperatura [0...9]
Establir límit inferior de la temperatura (un sensor)	Número del sensor [1...8]
	Tipus de missatge [1=establir límit inferior]
	Part entera de la temperatura [-128...127]
	Part decimal de la temperatura [0...9]

Taula 9.2: Format dels missatges que envia el telèfon mòbil

Missatge	Camps
Temperatura d'un sensor	Número del sensor [1...8]
	Part entera de la temperatura [-128...127]
	Part decimal de la temperatura [0...9]
	Indicació d'alarma [0=no hi ha alarma, 1=la temperatura instantània del sensor ha superat el límit superior, -1=la temperatura instantània del sensor ha baixat del límit inferior]

Taula 9.3: Format dels missatges que envia el microcontrolador

10. Descripció dels programes del microcontrolador

En aquest apartat s'explicarà com funciona el programa del microcontrolador de l'aplicació final, i el programa fet servir per programar l'accessori Bluetooth Click. En cada cas es descriuran la funció i les instruccions que desenvolupa el microcontrolador i com està estructurat el programa, sense mostrar instruccions concretes de llenguatge de programació C. També s'explicarà com s'han organitzat els arxius que contenen el codi dels programes. A l'annex A s'inclou el codi amb comentaris del programa del microcontrolador de l'aplicació final. A l'annex B s'inclou el codi amb comentaris del programa fet servir per programar l'accessori Bluetooth Click.

10.1. Programa principal

10.1.1. Funció

El programa principal és aquell que executa el PIC18F4520 quan es vol utilitzar l'aplicació final. La seva funció és fer que el microcontrolador es comuniqui amb els sensors de temperatura, amb el Bluetooth Click, amb el LCD i amb els elements auxiliars de la placa de desenvolupament per tal de complir amb les especificacions del projecte.

Per una banda ha de fer un conjunt de tasques amb un període d'un segon: obtenir les temperatures de tots els sensors (cada un d'ells identificat amb un número), actualitzar-les a la pantalla LCD, actualitzar-ne els valors màxims i mínims per a cada sensor, comprovar si alguna temperatura està fora dels seus límits (alarma) i enviar les temperatures i indicacions d'alarma per l'USART.

També ha de poder detectar si un dels polsadors de la placa ha estat pitjat, i canviar la parella de temperatures que s'estan representant a la LCD en conseqüència.

D'altra banda, ha de gestionar les sol·licituds que puguin arribar per l'USART, originades des d'un telèfon mòbil. Els possibles missatges que poden arribar al microcontrolador es mostren a la Taula 9.2.

Per fer totes aquestes funcions, el microcontrolador ha de fer un conjunt de càlculs i operacions d'una manera estructurada en el temps. L'ordre o estructura d'aquestes operacions es descriu a l'apartat 10.1.2.

10.1.2. Estructura i funcionament

Per entendre el funcionament del programa, és convenient tenir en compte les variables que s'han utilitzat i la seva funció (Taula 10.1).

VARIABLE	FUNCIÓ
i	Variable auxiliar per a executar instruccions de forma repetitiva
last_value_S0	Recorda l'últim valor mostrejat en el polsador S0 de la placa
last_value_S1	Recorda l'últim valor mostrejat en el polsador S1 de la placa
current_value_S0	Emmagatzema el valor actual en el polsador S0 de la placa
current_value_S1	Emmagatzema el valor actual en el polsador S1 de la placa
MODE	Recorda quin tipus de temperatura (màxima, mínima, instantània, límit superior, límit inferior) ha d'enviar al mòbil per a cada sensor
data	Emmagatzema un byte rebut per la USART
receive_step	Variable auxiliar per interpretar i gestionar correctament les dades rebudes per la USART
format	Emmagatzema el tipus de missatge rebut per la USART. Aquest pot ser de reiniciar temperatures màximes o mínimes, de canviar el mode, de configurar nous límits superiors o inferiors, etc.
sensor_addressed	Emmagatzema el número del sensor sobre el qual es vol fer un canvi. Rebut per la USART
pe_receive	Emmagatzemen la temperatura, rebuda per la USART, que es vol fixar com a límit superior o límit inferior per un sensor
pd_receive	
alarm_flag	Indicar si la temperatura d'un dels sensors se surt d'un dels seus límits i de quin sensor i quin límit es tracta
TH_e	Emmagatzemen el límit superior de la temperatura per a cada sensor
TH_d	

TL_e	Emmagatzemen el límit inferior de la temperatura per a cada sensor
TL_d	
pe_max	Emmagatzemen la temperatura màxima de cada sensor
pd_max	
pe_min	Emmagatzemen la temperatura mínima de cada sensor
pd_min	
p_e	Emmagatzemen la temperatura instantània de cada sensor
p_d	
sp	Emmagatzema el <i>scratchpad</i> del sensor de temperatura del qual es vol llegir la temperatura
cs	Recorda quina parella de temperatures s'han de representar en la LCD
array_address	Emmagatzema els codis ROM de cada sensor
NUM_SENS	Recorda el nombre de sensors presents en el bus en l'encesa

Taula 10.1: Descripció de les variables del programa principal

En el moment d'iniciar-se, el programa comença fent una sèrie d'operacions:

- Inicialitza un conjunt de variables.
- Configura els ports RD0 i RD1, associats als pulsadors, com a entrada per tal de llegir els valors digitals de les tensions associades.
- Inicialitza i configura el Timer0 per tal que origini una interrupció cada 25ms, mentre les interrupcions estiguin habilitades.
- Inicialitza i configura el LCD.
- Executa un algoritme de detecció de sensors a partir del qual en llegeix el codi ROM i calcula quants sensors hi ha presents al bus.
- Configura la resolució de tots els sensors a 12 bits (0,0625°C).

- Configura l'USART en mode asíncron i amb un baud rate de 38400 bits/s. S'habiliten les interrupcions de recepció, que s'activen cada cop que es rep un byte.

Després d'aquestes operacions, el programa arriba a un bucle, on executa de forma repetitiva un conjunt d'instruccions. Les instruccions dins d'aquest bucle poden ser interrompudes per una interrupció. A l'apartat 6.1.3 s'explica el que és una interrupció.

A continuació es mostren les instruccions dins del bucle repetitiu:

- Ordena a tots els sensors fer una conversió de la temperatura.
- Habilita les interrupcions.
- Genera un retard d'un segon, durant el qual pot ser interromput.
- Desactiva les interrupcions.
- Llegeix les temperatures de tots els sensors, actualitza els valors màxims i mínims, i comprova si alguna temperatura està fora dels límits, actualitzant una variable anomenada `alarm_flag` en conseqüència.
- Envia per l'USART un tipus de temperatura per cada sensor (instantània, màxima, mínima, límit superior, límit inferior), segons el que s'ha especificat des del telèfon mòbil. Per defecte s'envia la temperatura instantània. El format en què s'envia la temperatura per cada sensor és el següent: primer envia el número del sensor, després la part entera de la temperatura corresponent, després la part decimal i, finalment, un número indicador d'alarma. Si aquest número és un 1, el sensor sobrepasa el límit superior, si és un -1 el límit inferior, i si és un 0 cap límit.
- Actualitza en el LCD els valors de les temperatures instantànies.

En aquest programa hi poden haver dues interrupcions diferents, sempre i quan estiguin habilitades (99,9% del temps). Una interrupció cada 25ms deguda al timer0, i una interrupció per l'USART cada cop que es rep un byte. En cada cas la seqüència d'instruccions que s'executa és diferent i no pot ser interrompuda per l'altre interrupció.

La seqüència d'instruccions quan la interrupció és originada pel timer0 és la següent:

- Llegeix el valor digital de la tensió en els polsadors S0 i S1.
- Compara per S0 i S1 el valor digital llegit amb el valor anterior del mateix per detectar un flanc de baixada, fet que indica que han estat polsats (els circuits dels polsadors tenen una resistència de pull-up). Si és el primer cop que entra a aquesta

seqüència, el valor anterior es considera igual a un 1 lògic.

- Quan hi ha un flanc de baixada, canvia la parella de temperatures que es mostren a la LCD. Si el polsador és el S1 es mostren les temperatures dels sensors amb un número identificador immediatament superior respecte dels que s'estaven representant inicialment. Si un dels sensors és el de número més gran, es torna al sensor 1. Amb el polsador S0 es fa el mateix però amb un número immediatament inferior, i si s'estava representant el sensor 1, es passa a representar el sensor de número més gran.
- Finalment, emmagatzema els valors digitals de S1 i S0 llegits a l'inici de la rutina d'interrupció en les variables anomenades `last_value_S1` i `last_value_S0`.

Quan la interrupció és deguda a una dada (byte) rebuda per l'USART, es fan les següents accions:

- S'emmagatzema el byte rebut en la variable `data`. Aquesta dada pot contenir diferent informació segons el pas de recepció (variable `receive_step`) en el que s'estigui. El pas inicial de recepció és el 0.
- Si la variable `receive_step` és 0, la dada rebuda podrà contenir un número referit a un sensor, sobre el qual es voldrà fer algun canvi; o bé un número especial: el -99. Si es un número referit a un sensor, aquest es guardarà en `device_addressed`. Aleshores `receive_step` passarà a ser 1 i s'acabarà la rutina d'interrupció. En cas que sigui -99, es canviarà la variable `MODE` per tal d'enviar la temperatura instantània per tots els sensors i s'acabarà la rutina d'interrupció.
- Si `receive_step` és 1, la dada rebuda és un número que indica quin tipus de modificació es vol fer sobre el sensor referenciat en `device_addressed`. Aquest número es guarda a la variable `format` i en funció del seu valor s'executen instruccions diferents (Taula 10.2). Quan es fan aquestes accions finalitza la rutina d'interrupció.
- Si `receive_step` és 2, la dada rebuda és un número que indica quin tipus de temperatura s'ha d'enviar per l'USART per al sensor amb el número emmagatzemat a `device_addressed`. El número 0 vol dir temperatura instantània, el 1 màxima, el 2 mínima, el 3 límit superior i el 4 límit inferior. Aquest número s'emmagatzema a la variable `MODE` per recordar-lo en el moment de la transmissió de dades per la USART. La variable `receive_step` passa a ser 0 i finalitza la rutina d'interrupció.
- Si `receive_step` és 3, la dada rebuda és la part entera d'una temperatura, la qual es vol establir com a límit superior o inferior (segons la variable `format`) de la

temperatura del sensor identificat per `device_addressed`. S'emmagatzema aquesta dada en la variable `pe_receive`, i `receive_step` passa a ser 4 per rebre'n la part decimal. Finalitza la rutina d'interrupció.

- Si `receive_step` és 4, es rep la part decimal de la temperatura explicada en l'apartat anterior i, juntament amb la part entera rebuda en el pas anterior, s'emmagatzemen en les variables que contenen els límits superiors o inferiors de les temperatures, segons correspongui. Finalment `receive_step` es posa a 0 i s'acaba la rutina d'interrupció.

Cal destacar que el microcontrolador també executa algunes instruccions que no han estat explicades però que serveixen per garantir un bon funcionament de l'aplicació. Aquestes instruccions es poden veure a l'annex A.

NÚMERO (format)	MODIFICACIÓ SOL-LICITADA	ACCIONS
0	Modificar el límit superior de la temperatura d'un sensor	Canvia el valor de <code>receive_step</code> a 3 per tal de rebre la temperatura en qüestió
1	Modificar el límit inferior de la temperatura d'un sensor	
2	Canviar el tipus de temperatura que es vol rebre d'un sensor	Canvia el valor de <code>receive_step</code> a 2 per tal de rebre un missatge indicant el tipus de temperatura es vol obtenir
3	Reiniciar la temperatura màxima d'un sensor	Modifica el valor de la temperatura màxima del sensor indicat per <code>device_addressed</code> . El nou màxim serà $-55,0^{\circ}\text{C}$ i <code>receive_step</code> torna a ser 0
4	Reiniciar la temperatura mínima d'un sensor	Modifica el valor de la temperatura mínima del sensor indicat per <code>device_addressed</code> . El nou mínim serà $125,0^{\circ}\text{C}$ i <code>receive_step</code> torna a ser 0

Taula 10.2: Accions en la rutina d'interrupció de l'USART quan la variable `receive_step` val 1, en funció del valor de la variable `format`

10.1.3. Organització del codi

El codi del programa principal està repartit en diferents arxius. Hi ha un arxiu que és l'arxiu principal. En aquest arxiu estan definides les rutines d'interrupcions i la funció principal, també anomenada *main*, que és la que executa el microcontrolador quan està en funcionament. La resta d'arxius contenen un conjunt de paràmetres de configuració i funcions que fa servir l'arxiu principal.

En aquest cas, hi ha 5 fitxers més apart de l'arxiu principal, anomenats *ConfigurationBits*, *1wire_PLL*, *auxiliar*, *DS18B20* i *LCD1602_PLL*. El primer de tots no té cap funció de llenguatge C, únicament serveix per configurar certs paràmetres del microcontrolador, com ara la freqüència del senyal de rellotge, configurada a 16MHz. Els altres 4 fitxers contenen funcions relacionades amb el control dels sensors, del LCD, de l'USART i dels polsadors. A la Taula 10.3 es fa una descripció d'aquests 4 fitxers.

NOM FITXER	DESCRIPCIÓ
1wire_PLL	Conté funcions per controlar el funcionament del bus 1-Wire. Aquestes funcions permeten posar un 0 o 1 lògics al bus, escriure o llegir un bit o un byte, donar un senyal de reinici o buscar dispositius. L'arxiu serveix quan el microcontrolador té un senyal de rellotge de 16MHz.
DS18B20	Funcions per ordenar als sensors una conversió de temperatures, per llegir el <i>scratchpad</i> dels sensors i per descodificar la temperatura del <i>scratchpad</i> .
LCD1602_PLL	Conté funcions per inicialitzar la LCD, per esborrar els caràcters que hi hagi a la pantalla, per situar el cursor i per escriure caràcters a la pantalla. L'arxiu serveix quan el microcontrolador té un senyal de rellotge de 16MHz.
auxiliar	Les funcions que conté fan tasques concretes, referents a diferents parts del programa: <ul style="list-style-type: none"> • Dues funcions per canviar la parella de temperatures que es representa en la LCD. • Una funció per actualitzar la LCD amb els nous valors de temperatura obtinguts. • Una funció que identifica els sensors presents al bus. • Una funció per enviar totes les temperatures i les indicacions d'alarma per la USART. • Una funció per calcular les alarmes. • Una funció per calcular les temperatures màximes i mínimes.

Taula 10.3: Descripció dels fitxers secundaris del programa principal

10.2. Programa per configurar el Bluetooth Click

Aquest programa es basa en un exemple de programa proporcionat per Mikroelectronika, pensat per un altre microcontrolador i que configura el dispositiu d'una manera diferent, a més de fer altres funcions que en aquest projecte no són necessàries. El que s'ha fet és adaptar el codi per al PIC18F4520 i modificar algunes seccions per configurar el dispositiu apropiadament.

En els següents apartats s'explicarà com funciona la part del programa que és d'utilitat per al projecte.

10.2.1. Funció

El programa desenvolupat per configurar el dispositiu Bluetooth Click ha de canviar característiques del funcionament del mòdul de radiofreqüència comunicant-se a través de l'USART. El programa pot canviar el nom del dispositiu, posar-ho en mode slave o configurar-lo amb un baud rate igual al que tindrà la USART del microcontrolador quan estigui funcionant en règim permanent (38400 bits/s). També permet configurar una contrasenya per a quan un telèfon mòbil intenti connectar-s'hi.

De la mateixa manera que en el programa principal, el microcontrolador també ha de fer un conjunt d'operacions de forma ordenada en el temps. A l'apartat 10.2.2 es fa una descripció d'aquestes operacions. Cal destacar que el programa executa altres instruccions que no són importants.

10.2.2. Estructura i funcionament

Per configurar el mòdul, el microcontrolador només ha de fer una sèrie d'operacions, durant les quals poden haver-hi interrupcions degut a la recepció de dades per l'USART. Un cop fetes aquestes operacions, el dispositiu ja està configurat i es pot desconnectar el microcontrolador de l'alimentació.

A continuació s'expliquen les instruccions que executa el microcontrolador:

- Inicialitza un conjunt de variables per gestionar la comunicació amb el Bluetooth Click.
- Inicialitza i configura l'USART amb un baud rate igual al que tingui el Bluetooth Click abans de configurar-se (per defecte 115200 bits/s), amb interrupcions per recepció.
- Escriu per l'USART "\$\$\$" en codi ASCII i espera fins que el mòdul Bluetooth comuniqui que ha entrat en mode de comandes.

- Configura el dispositiu enviant un conjunt de comandes. Després de cada comanda s'espera a rebre la senyal "AOK" en ASCII indicant que la comanda enviada és correcte. A la Taula 10.4 es mostren les comandes que s'envien i la funció que tenen.
- Envia la seqüència "---" en ASCII per tal de sortir del mode de comandes i rep "END" indicant que se n'ha sortit.

COMANDA (ASCII)	FUNCIO
SN,BT	Estableix el nom del dispositiu Bluetooth a "BT"
SU,38	Estableix el baud rate a 38400 bits/s
SM,0	Estableix el mode de funcionament esclau
SP,1234	Estableix el codi pin 1234 per connectar-se al mòdul

Taula 10.4: Comandes de configuració del mòdul Bluetooth Click

10.2.3. Organització del codi

El codi d'aquest programa està repartit en 3 arxius: un arxiu principal, un arxiu auxiliar amb funcions per gestionar la comunicació amb el dispositiu Bluetooth (*BT_routines*) i un arxiu de configuració del microcontrolador.

En l'arxiu principal és on està definida la rutina d'interrupció per rebre dades per l'USART i la funció principal o *main*, que és la que executa el microcontrolador. En aquesta funció es crida a una subrutina que envia totes les ordres necessàries per configurar el Bluetooth segons el procediment descrit a l'apartat 10.2.2. Aquesta subrutina es diu *BT_Configure* i està definida a l'arxiu auxiliar.

L'arxiu de configuració és el mateix que el del programa principal, el *ConfigurationBits*.

11. Descripció de l'aplicació del telèfon mòbil

En els següents apartats es farà una descripció del software utilitzat per programar l'aplicació pel telèfon mòbil i s'explicarà el seu funcionament. S'inclourà un manual d'usuari explicant com s'ha d'utilitzar el sistema format per la placa de desenvolupament amb tots els components i el telèfon mòbil.

11.1. MIT App Inventor 2

El MIT App Inventor és una eina de programació basada en blocs que permet a qualsevol persona, inclús si no té experiència, començar a programar i construir aplicacions funcionals per a dispositius amb sistema operatiu Android. Els més inexperts poden tenir la seva primera aplicació bàsica funcionant en un hora o menys, i poden programar aplicacions més complexes en un temps significativament més petit que amb altres llenguatges de programació basats en text.

El seu entorn de desenvolupament està format per un dissenyador (*Designer*) i per un àrea per blocs funcionals (Blocks).

11.1.1. Designer

En el dissenyador se situen i configuren un conjunt de components que tindran una certa funció en l'aplicació. Esta format per 5 seccions: el *Palette*, *Viewer*, *Components*, *Media* i *Properties*. El *Palette* és la zona on s'escullen els components que es faran servir, que poden ser visibles en la pantalla del telèfon mòbil o no visibles. El *Viewer* és un visualitzador que mostra com queden els components visibles distribuïts a la pantalla. *Components* mostra un esquema dels components que s'han incorporat a l'aplicació i permet canviar-los el nom o esborrar-los. *Media* permet carregar arxius a l'aplicació, com ara imatges o sons i *Properties* permet modificar alguns paràmetres dels components situats. En molts casos aquests paràmetres es poden modificar durant el funcionament de l'aplicació. A la Figura 11.1 es pot veure, com a exemple, les 5 seccions de l'entorn de disseny d'una aplicació.

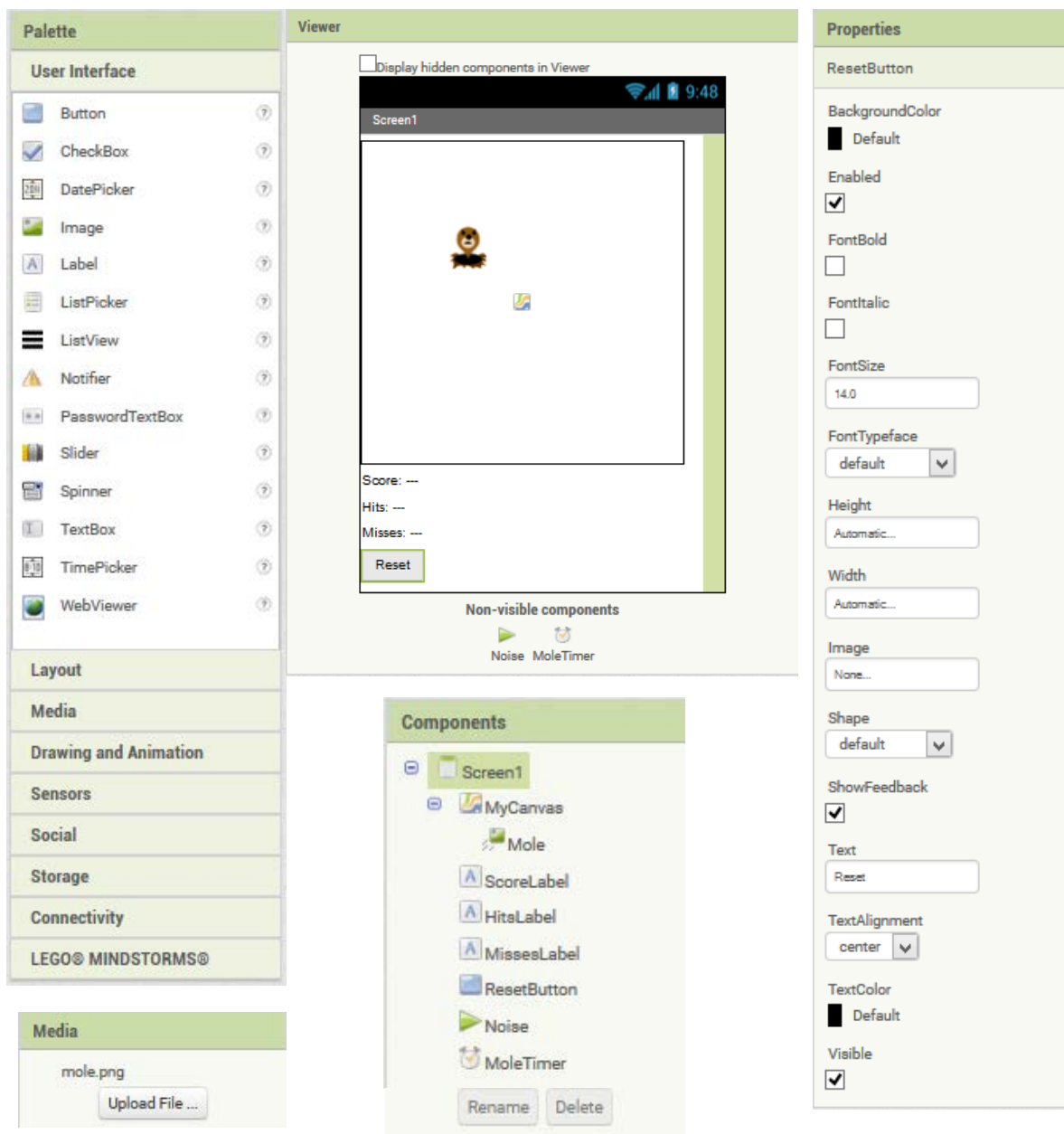


Figura 11.1: Exemple de les 5 seccions de l'entorn de disseny d'una aplicació

En l'aplicació feta per aquest projecte s'han fet servir, com a elements visibles, botons (*Button*), etiquetes de text (*Label*), un visualitzador de llistes (*ListView*) i seleccionadors d'elements d'una llista (*ListPicker*). Com a elements no visibles s'ha utilitzat un component per gestionar la comunicació per Bluetooth (*BluetoothClient*), temporitzadors per realitzar determinades tasques amb certa periodicitat (*Timer*), un notificador per mostrar missatges a la pantalla (*Notifier*), un reproductor per reproduir vibracions (*Player*) i un element que

permet recordar dades de l'últim cop que es va utilitzar l'aplicació (*TinyDB*).

11.1.2. Blocks

L'àrea de blocs és la secció on es programa el funcionament dels components situats al dissenyador i, en definitiva, de l'aplicació. Per fer l'aplicació, es seleccionen blocs que fan una tasca concreta i es connecten entre ells. Hi ha diferents tipus de blocs: alguns blocs estan associats a components que s'han situat al projecte i altres blocs venen incorporats pel MIT App Inventor i permeten fer operacions lògiques i aritmètiques, operar amb text (*strings*) i amb llistes, definir variables, o controlar l'ordre de les instruccions a executar pel mòbil, entre d'altres. A la Figura 11.2 es mostra l'àrea de blocs d'un exemple d'aplicació.

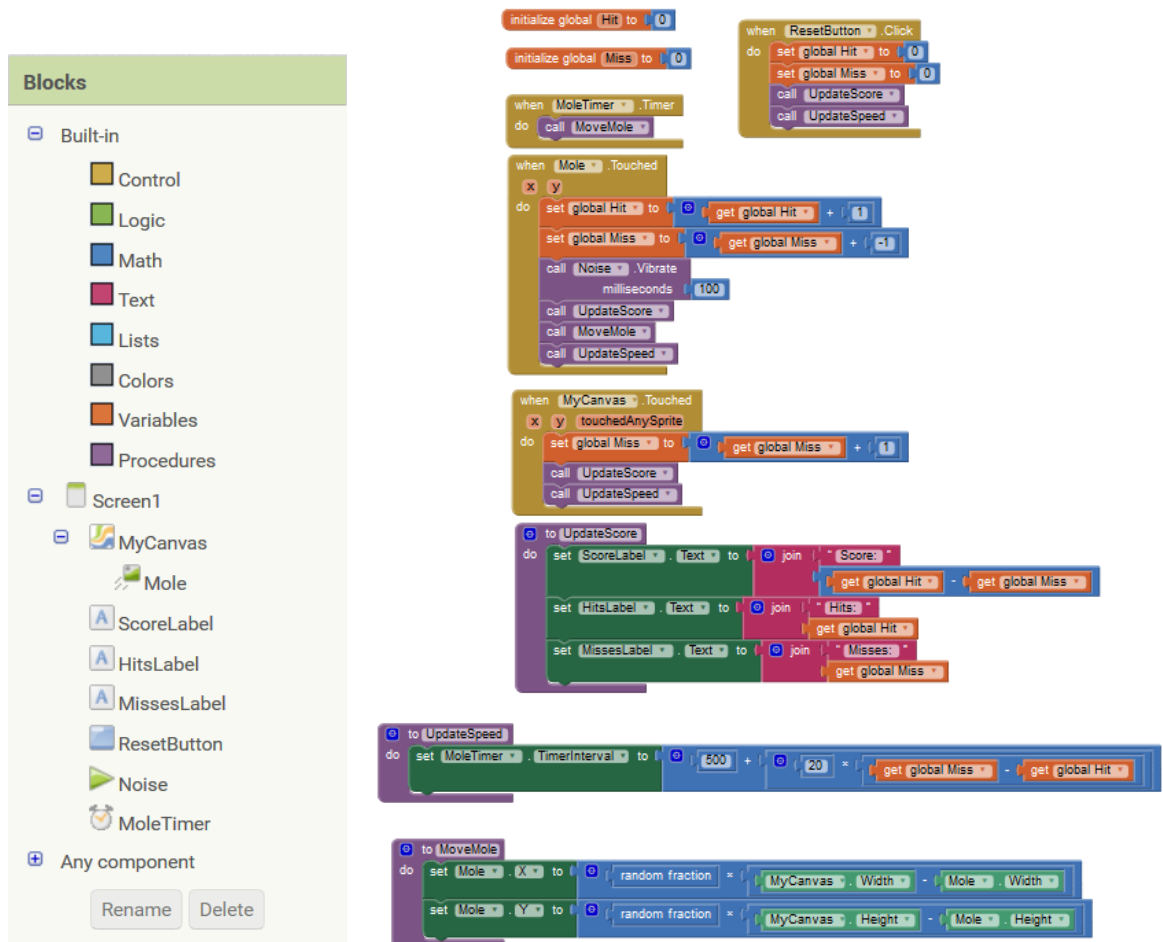


Figura 11.2: Exemple d'àrea de blocs d'una aplicació

11.2. Descripció de l'aplicació

11.2.1. Funcionament

L'aplicació pel telèfon mòbil fa d'interfície amb l'usuari operador del sistema de supervisió de temperatures. A la Figura 11.3 es mostra l'aspecte que té l'aplicació en el moment d'iniciar-se.

Primer de tot ha de poder connectar-se al Bluetooth-Click, prèviament emparellat amb el telèfon mòbil, per enviar sol·licituds i rebre dades de temperatures. Per defecte, després de connectar-s'hi, el mòbil envia un missatge al microcontrolador indicant que vol rebre el tipus de temperatura instantània.

Les dades que rep el mòbil són un tipus de temperatura per cada sensor (per defecte instantània), amb indicacions de si les temperatures instantànies han rebassat algun dels seus límits. El format en què estan aquestes dades està descrit a l'apartat 9.2.

Per fer el procés de recepció, es té en consideració que els missatges que envia el microcontrolador cap al mòbil són de 4 camps d'un byte cadascun. El mòbil comprova cada 40ms si li han arribat dades i, en cas afirmatiu, les llegeix en grups de 4 bytes (un grup=una temperatura), sabent què vol dir cada un dels 4 bytes. Després de la descodificació, el mòbil representa les temperatures rebudes en la pantalla seguint les especificacions del projecte. Si el microcontrolador informa que alguna de les temperatures està fora dels seus límits, el mòbil emet un missatge indicant el sensor i el límit rebassat, i vibra durant 1 segon.

L'aplicació també ha de gestionar totes les sol·licituds que l'usuari pugui fer a través dels botons presents en l'aplicació i donar les respostes que siguin necessàries en cada cas. Aquestes sol·licituds es poden fer quan l'aplicació està funcionant i s'ha fet la connexió amb el mòdul de radiofreqüència.

Quan l'usuari pitja algun dels botons: *Set TH*, *Set TL*, *Reset Tmax*, *Reset Tmin*, *Inst*, *Max*, *Min*, *TH* o *TL* (Figura 11.3), el mòbil ha de recordar que el botó ha estat pitjat i enviar la sol·licitud corresponent al microcontrolador en el moment adequat.

Donat que el mòdul de radiofreqüència no pot emetre i rebre dades a la vegada, l'aplicació s'ha d'assegurar que envia la sol·licitud quan el mòdul no està emetent. Tenint en compte que les dades de temperatures s'emeten amb un període d'un segon, s'estableix el moment adequat per enviar sol·licituds durant 500 ms després d'haver rebut un conjunt de temperatures.

La utilitat específica de cada un dels botons i elements de l'aplicació es descriu a l'apartat 11.2.2.

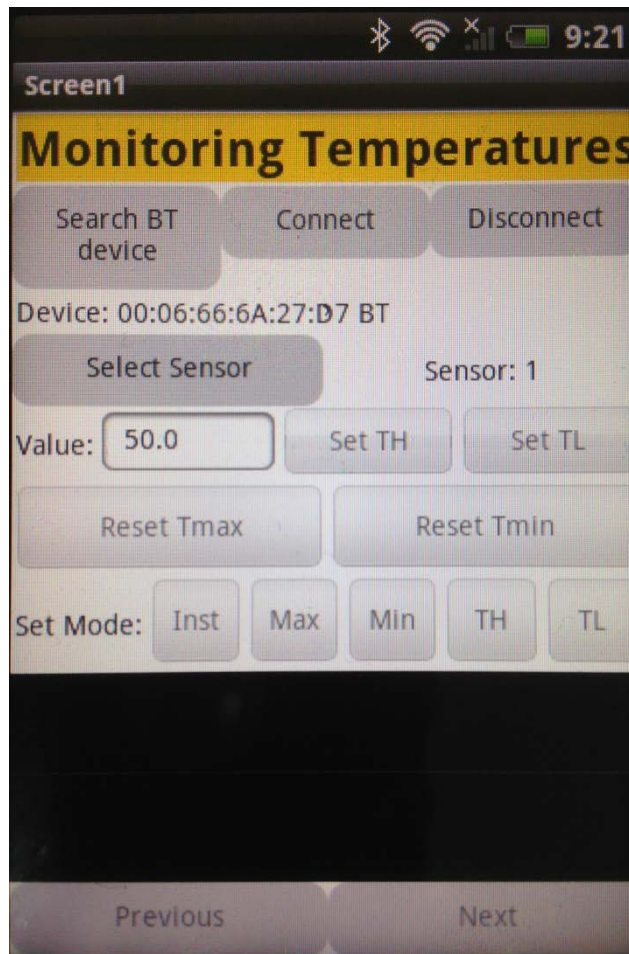


Figura 11.3: Aplicació pel telèfon mòbil

11.2.2. Manual d'usuari

Aquest manual explica com utilitzar el sistema format per la placa de desenvolupament amb els components i l'aplicació pel telèfon mòbil, fent referència a elements de la imatge de l'aplicació en la Figura 11.3.

En primer lloc, abans de poder connectar-se al Bluetooth Click, tots els elements de la placa han d'estar connectats tal i com es descriu a l'apartat 8, amb els sensors situats en els punts on es vulgui supervisar la temperatura i connectats al bus 1-wire.

Després d'haver fet la connexió del hardware, el pròxim pas consisteix en fer l'emparellament Bluetooth entre el telèfon mòbil i el mòdul de radiofreqüència de la placa. El mòdul de Bluetooth té com a nom BT i demana la contrasenya 1234 per fer l'emparellament. Acte seguit ja es pot iniciar l'aplicació.

Dintre de l'aplicació, si l'emparellament ha estat correcte, es pot fer la connexió. S'ha de buscar i seleccionar el dispositiu BT a través del botó *Search BT Device* i prémer el botó *Connect*. El dispositiu seleccionat apareix en una etiqueta de text just a sota del botó *Search BT device* i és recordat per l'aplicació per a la propera vegada que s'iniciï. Si hi ha un error en la connexió el telèfon vibrarà i mostrarà un missatge en la pantalla informant de l'error. Si s'ha connectat amb èxit, també apareixerà un missatge indicant-ho.

Al cap d'uns pocs segons d'haver-se fet la connexió, el mòbil començarà a rebre les temperatures instantànies de tots els sensors i mostrarà les dels sensors 1 i 2 per la pantalla. Els botons que permeten fer configuracions (inicialment inhabilitats) s'habilitaran.

Cal destacar que, per a la representació de les temperatures, s'utilitza un punt decimal en comptes d'una coma ja que l'aplicació s'ha decidit fer en anglès, per tal que pugui ser entesa internacionalment.

Per veure la temperatura d'una altra parella de sensors, s'ha de pitjar el botó *Next* o el botó *Previous*. Si el botó pitjat és *Next*, es visualitzaran les temperatures dels sensors amb un número identificador immediatament superior als que hi havia representats. Si un d'ells és el que té un número identificador més gran, es tornarà al sensor 1. Amb el botó *Previous* passa el mateix però amb un número identificador immediatament inferior, i si s'estava representat la temperatura del sensor 1, es representarà la temperatura del sensor amb número identificador més gran.

Per fer una modificació sobre un sensor, el primer pas és seleccionar el número identificador del sensor a través del botó *Select Sensor*. El sensor seleccionat apareixerà a la dreta d'aquest botó.

Amb el sensor seleccionat, hi ha 5 botons que són per indicar quin tipus de temperatura es vol rebre: el botó *Inst* per rebre la temperatura instantània, el *Max* per rebre la màxima, el *Min* per rebre la mínima, el *TH* per rebre el límit superior i el *TL* per rebre el límit inferior.

Per reiniciar les temperatures màximes o mínimes del sensor seleccionat, s'ha de polsar el botó *Reset Tmax* o *Reset Tmin*, respectivament.

Per modificar els límits superior o inferior de les temperatures per a un sensor (inicialment a 50,0°C i -10,0°C, respectivament), en primer lloc s'ha de seleccionar el nou límit que es vol establir, tant si es superior com inferior. Per fer-ho, s'ha d'introduir el valor de la temperatura en °C (entre -55,0 i 125,0) en la caixa de text precedida per *Value*: amb una xifra decimal separada per una coma o un punt. Un cop introduït el valor, es poden polsar els botons *Set TH* o *Set TL*, segons si el valor introduït es vol establir com a límit superior o límit inferior del sensor seleccionat, respectivament.

Durant el funcionament de l'aplicació, si la temperatura d'un sensor se surt dels seus límits, el telèfon vibrarà i apareixerà una notificació de text indicant de quin sensor es tracta i quin dels seus dos límits ha superat. Aquesta notificació apareixerà cíclicament cada 5 segons mentre la temperatura en qüestió segueixi estant fora dels límits. Si tenen lloc dues o més alarmes a la vegada, només es visualitzarà la del sensor amb número més gran.

Per afegir o treure sensors al sistema, s'ha de polsar el botó *Disconnect* per desconnectar el mòbil del Bluetooth-Click, desconnectar l'alimentació de la placa de desenvolupament i afegir o treure els sensors que siguin necessaris. Un cop fetes les modificacions, es pot tornar a alimentar la placa i a fer la connexió amb el mòdul de radiofreqüència. En aquest procés es restableix la configuració per defecte dels sensors.

Cal destacar que la numeració dels sensors pot variar en afegir o treure sensors. Així doncs caldria identificar el número associat a cada sensor en el moment de connectar-se.

En el cas que l'usuari vulgui visualitzar les temperatures instantànies a prop de la placa, ho pot fer a través del LCD, variant la parella de sensors representats amb els polsadors S0 i S1, anàlogament a com es fa amb els botons *Previous* i *Next* de l'aplicació.

Si un dels sensors fa mal contacte, la temperatura que es llegirà serà de 85,0°C, no afectant aquest valor a les alarmes ni al càlcul dels valors màxims ni mínims. El mateix passa si es desconnecta un sensor tot i que la temperatura llegida serà de -1,0.

12. Pressupost

A continuació es mostrarà el càlcul del cost de realització del projecte. Es detallarà el cost de cada component, de les eines utilitzades per desenvolupar l'aplicació, de l'amortització de l'ordinador utilitzat i de les hores dedicades.

MATERIAL			
Element	Quantitat	Preu Unitari (€)	Cost Total (€)
Placa Open18F4520	1	25,83	25,83
LCD1602	1	3,29	3,29
Sensors DS18B20	5	1,73	8,65
Bluetooth Click	1	25,83	25,83
ICD3	1	179,01	179,01
Total			242,61

DESPESES DE PERSONAL			
Tipus de feina	Hores	Cost Horari (€/h)	Cost Total (€)
Estudi previ	80	40,00	3.200,00
Programació	120	40,00	4.800,00
Muntatge	20	40,00	800,00
Proves	30	40,00	1.200,00
Redacció memòria	80	40,00	3.200,00
Total			13.200,00

CONSUM DE RECURSOS			
Tipus de recurs	Hores	Cost Horari (€/h)	Cost Total (€)
Consum elèctric del PC i plaques (100W)	270	0,14	37,80
Consum elèctric de l'espai de treball	270	0,14	37,80
Total			75,60

COST TOTAL DEL PROJECTE	
Part	Cost (€)
Material	242,61
Temps de treball	13.200,00
Consum de recursos	75,60
Total	13.518,21

13. Impacte ambiental

En aquest capítol es farà una anàlisi de l'impacte ambiental del projecte. És necessari tenir en compte dos aspectes clarament diferenciats: les possibles accions quan el producte ha arribat al final de la seva vida útil, i la influència de les transmissions per radiofreqüència en les persones i en altres elements electrònics propers als transmissors.

Les opcions a considerar quan acaba la vida útil del producte són el reciclatge o el desfet en gestors de residus especialitzats en residus electrònics.

Si es considera el reciclatge de les parts d'un exemplar de sistema, aquest resulta inviable ja que la reutilització implicaria un cos més elevat que l'adquisició de nous components. Per tant, la millor opció consisteix en la reutilització de cada conjunt per a altres funcions.

A més, tot i que es facin diferents fases de reciclatge, és necessari acabar desfent els components de forma correcta. Això es realitza en centres de gestió de residus d'aparells elèctrics i electrònics (RAEE).

Quant a les transmissions de radiofreqüència, aquestes són generades pel mòdul Bluetooth Click i pel telèfon mòbil. Només és necessari analitzar les del Bluetooth Click. Per fer els càlculs sobre la seva influència, és necessari conèixer la freqüència de treball, la distància a la que es trobaran els elements de les antenes i la potència de les mateixes.

La freqüència de treball del mòdul de radiofreqüència Bluetooth Click és d'entre 2,402 i 2,480 GHz, compresa dintre de la categoria Alta freqüència d'entre 3 MHz i 3 GHz. A altes potències, els espectres electromagnètics que se'n generen poden causar dolors, cremades, o pèrdues de visió en individus, entre d'altres. Per conèixer aquest impacte en les persones és necessari controlar la potència del camp electromagnètic aplicat i el temps d'exposició.

La potència màxima de sortida del Bluetooth Click és de 1 mW. Per determinar la incidència del camp generat pel dispositiu és té en consideració el paràmetre Relació d'Absorció Específic (SAR). Segons l'Organització Mundial de la Salut, entre les freqüències 1 MHz i 1 GHz es necessita un SAR mínim de 4 W/Kg per produir efectes perjudicials en la salut i, segons el Codi de Seguretat 6 del departament de sanitat de Canadà, per un llarg període d'exposició, no és admissible una densitat de potència més gran que 0,4 W/Kg per evitar lesions de qualsevol tipus. Suposant un pes elevat per una persona, com ara 75 Kg, i l'últim valor mencionat, es pot calcular la potència màxima que es pot suportar, que és de 30 W. Tenint en compte que la potència màxima del mòdul de radiofreqüència és de 1 mW, es pot concloure que l'ús del sistema no suposa cap dany a les persones.

Conclusions

La principal conclusió és que ha estat possible aconseguir que el sistema desenvolupat compleixi amb totes les especificacions del projecte, tot i tenir la limitació de fer servir components disponibles a la universitat.

Cal destacar que l'existència d'algunes llibreries per al control d'alguns components (LCD1602, DS18B20, 1-Wire...) ha facilitat el desenvolupament del software, com també ho ha fet la plataforma MIT App Inventor en el cas del desenvolupament de l'aplicació per al telèfon mòbil.

Una altra conclusió favorable és que el rang de temperatures mesurable pel sistema és, de fet, més ample que el demanat a les especificacions. Fora d'aquest rang, però, la precisió del sensor disminueix, essent diferent de 0,5 °C.

Quant a l'alimentació del sistema, s'ha aconseguit fer-la amb la mateixa tensió per a tots els components ja que els seus rangs d'operació tenien un interval en comú.

Tot i que a l'hora de fer les proves s'ha fixat el nombre de sensors, al sistema final es podrien situar més sensors fent modificacions sobre el hardware. Es podrien derivar més cables del bus 1-Wire per tal de situar tants sensors com fos necessari.

Finalment, segons les necessitats de l'usuari del sistema, es podria augmentar la distància màxima que limita la supervisió de les temperatures des del telèfon mòbil (actualment de 100 m). Per fer-ho, s'hauria de substituir el mòdul de radiofreqüència Bluetooth Click per un altre basat en Wifi, amb més abast. Aquesta substitució implicaria haver de programar noves rutines per al control de la comunicació, tant per al microcontrolador com per al telèfon mòbil.

Bibliografia

- Microchip Technology Inc. *PIC18F2420 / 2520 / 4420 / 4520 Data Sheet*. 2008.
- Maxim Integrated. *DS18B20 Programmable Resolution 1-Wire Digital Thermometer*. 2008.
- Manuel Moreno. *DS18B20 Lib User Manual*. UPC BarcelonaTech, 2014.
- SUNPLUS. *SPLC780D: 16COM/40SEG Controller/Driver Datasheet*. 2004.
- Manuel Moreno. *LCD1602 Lib User Manual*. UPC BarcelonaTech, 2014.
- Mikroelektronika. *Bluetooth Click Manual*. 2012.
- Microchip Technology Inc. *RN41 / RN41N Class 1 Bluetooth Module with EDR Support*. 2014
- Roving Networks. *Bluetooth Data Module Command Reference & Advances Information User's Guide*. 2013.
- Microchip Technology Inc. *MPLAB C18 C COMPILER LIBRARIES*. 2005.

Bibliografia complementària

- Dr. Farid Farahmand. *Programming the PIC18 Using C-Coding*. 2012.
- Microchip Technology Inc. *MPLAB C18 C COMPILER GETTING STARTED*. 2005.
- Armando Astarloa Cuéllar. *Sistemas de procesamiento digital*. Aitor Zuloaga Izaguirre. Delta Publicaciones, 2008.
- Dogan Ibrahim. *Advanced PIC Microcontroller Projects in C*. 2011.