

## Títol del Treball Final

# Supervisor de temperatura via Bluetooth

**Joan Clotet Santanach**

Grau de tecnologies de la telecomunicació.  
Disseny de sistemes electrònics.

**Carlos Gonzalo Moreno Soriano**  
**Germán Cobo Rodriguez**

15 de juny de 2020



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

## FITXA DEL TREBALL FINAL

|                                      |  |
|--------------------------------------|--|
| <b>Títol del treball:</b>            | <i>Supervisor de temperatura via Bluetooth</i>   |
| <b>Nom de l'autor:</b>               | <i>Joan Clotet Santanach</i>                     |
| <b>Nom del consultor/a:</b>          | <i>Carlos Gonzalo Moreno Soriano</i>             |
| <b>Nom del PRA:</b>                  | <i>Germán Cobo Rodríguez</i>                     |
| <b>Data de lliurament (mm/aaaa):</b> | <i>06/2020</i>                                   |
| <b>Titulació o programa:</b>         | <i>Grau de tecnologies de la telecomunicació</i> |
| <b>Àrea del Treball Final:</b>       | <i>Disseny de sistemes electrònics</i>           |
| <b>Idioma del treball:</b>           | <i>Català</i>                                    |
| <b>Paraules clau</b>                 | <i>Supervisor, temperatura i Bluetooth</i>       |

**Resum del Treball (màxim 250 paraules):** *Amb la finalitat, context d'aplicació, metodologia, resultats i conclusions del treball*

En l'actualitat, degut a què la tecnologia és a l'abast de gairebé tothom, la gent vol tenir control i comunicació permanent amb tots els elements que l'envolten, per això utilitza les xarxes de telefonia mòbil, Wifi, Wimax o qualsevol altre mètode que permeti connexió a internet.

Aquesta tecnologia de comunicacions també s'aplica al món de la indústria per tal d'obtenir dades que, un cop analitzades, permetin millorar la qualitat i la quantitat del treball.

En aquest treball de final de grau de tecnologies de la telecomunicació es desenvolupa una aplicació d'aquestes tecnologies per controlar la temperatura d'un sensor mitjançant una comunicació Bluetooth.

Per portar-ho a terme es busca un sensor de temperatura, un mòdul Bluetooth i un microcontrolador adients per aquest fi. Mitjançant un circuit elèctric obtindrem la temperatura del sensor i el microcontrolador la interpretarà i l'enviarà al mòdul Bluetooth perquè aquest el comuniqui a qui s'hi connecti, ja sigui mitjançant mòbil, ordinador o un altre aparell que disposi d'aquesta tecnologia.

**Abstract (in English, 250 words or less):**

Today, technology is available for everyone, people want control and permanently communication whit all things around them, that's way they use nets of mobile phones, Wifi, Wimax or another method that allows Internet connection.

These telecommunications technologies are also applied to the world of industry in order to obtain data, that, once analysed, can improve the quality and quantity of work.

In this final degree project developed an application of these technologies to control the temperature of a sensor via Bluetooth communication. To do this, a temperature sensor, a Bluetooth module and a microcontroller suitable for this purpose are sought.

Whit electrical circuit we obtain the temperature of the sensor and the microcontroller interprets and sends it to the Bluetooth module so that it communicates to the person who connects, by mobile phone, computer or other device that has this technology.

# Índex

|  |     |
|--|-----|
| Resum.....   | ii  |
| Abstract.....  | iii |
| Llista de figures.....                                 | iv  |
| Llista de taules.....                                  | v   |
| 1. Introducció.....                                    | 1   |
| 1.1 Context i objectius.....                           | 1   |
| 1.2 Enfocament i mètode.....                           | 1   |
| 1.3 Planificació.....                                  | 2   |
| 1.4 Incidències i riscos.....                          | 4   |
| 2. Bluetooth.....                                      | 4   |
| 2.1 Història.....                                      | 5   |
| 2.2 Característiques.....                              | 5   |
| 2.3 Cercar i escollir mòdul.....                       | 6   |
| 3. Sensor de temperatura.....                          | 7   |
| 3.1 Descripció.....                                    | 7   |
| 3.2 Justificació.....                                  | 8   |
| 4. Disseny del circuit elèctric d'adaptació.....       | 9   |
| 4.1 Circuit lector de resistència.....                 | 9   |
| 4.1.1 Pont de Wheatstone.....                          | 9   |
| 4.1.2 Lectura de la resistència.....                   | 10  |
| 5. Circuit elèctric.....                               | 13  |
| 5.1 Font d'alimentació de 5 V positius i negatius..... | 13  |
| 5.2 Font d'alimentació 3.5 V.....                      | 14  |
| 5.3 Disseny del circuit.....                           | 14  |
| 5.4 Disseny complet.....                               | 16  |
| 6. Microcontrolador.....                               | 18  |
| 6.1 Característiques.....                              | 18  |
| 6.2 Programació.....                                   | 19  |
| 6.3 Selecció del microcontrolador.....                 | 19  |
| 7. Programació del processador.....                    | 20  |
| 7.1 Introducció.....                                   | 20  |
| 7.2 Device_config.....                                 | 20  |
| 7.3 Pin_manager.....                                   | 21  |
| 7.4 Adc.....   | 22  |
| 7.5 Eusart.....  | 23  |
| 7.6 Mcc.....   | 24  |
| 7.7 Main.c.....  | 26  |
| 7.8 Simulació.....                                     | 27  |
| 8. Pressupost.....                                     | 29  |
| 9. Conclusions.....                                    | 31  |
| 10. Glossari.....                                      | 32  |
| 11. Bibliografia.....                                  | 33  |
| 11.1 Referències.....                                  | 33  |
| 11.2 Datasheets.....                                   | 33  |
| 11.3 Consulta.....                                     | 33  |

## Llista de figures

- Figura 1. Diagrama de blocs
- Figura 2. Diagrama de Gantt
- Figura 3: BT730-Sx module pin-out (Top View)
- Figura 4. Sensor
- Figura 5. Resposta de la resistència a la temperatura i dades del fabricant
- Figura 6. Pont Wheatstone
- Figura 7. Pont Wheatstone del circuit.
- Figura 8. Pont Wheatstone sense regulació d'intensitat i amb regulació d'intensitat
- Figura 9. Adaptador d'impedàncies
- Figura 10. Lectura de tensió d'entrada a l'amplificador
- Figura 11. Amplificador
- Figura 12. Valors de tensió per l'1
- Figura 13. Tensió 0 lògic
- Figura 14. Tensió 1 Lògic
- Figura 15. Font d'alimentació  $\pm 5$  V.
- Figura 16. Font d'alimentació 3.5 V
- Figura 17. Tensions de sortida quan  $T=20$  °C.
- Figura 18. Tensions de sortida quan  $T=120$  °C
- Figura 19. Simulació del sensor
- Figura 20. Esquema complet
- Figura 21. Placa PCB components
- Figura 22. Pistes cara de sota
- Figura 23. Pistes cara components
- Figura 25. Distribució dels PIN
- Figura 26. device\_config.c
- Figura 27. Definició ADC
- Figura 28. Definició UART
- Figura 29. Inicialització de l'ADC
- Figura 30. Selecció de canal i conversió ADC
- Figura 31. Control de EUSART
- Figura 32. Càlcul període conversió ADC
- Figura 33. Freqüències de treball de micro
- Figura 34. Codi mcc
- Figura 35. Conversió voltatge ADC en temperatura
- Figura 36. Comprovació mòdul Bluetooth
- Figura 37. Inici del mòdul Bluetooth
- Figura 38. Compilació
- Figura 39. Entrada de 45 °C
- Figura 40. Entrada de 120 °C
- Figura 41. Caixa exterior
- Figura 42. Antena
- Figura 43. Connector
- Figura 44. Caixa amb antena

## **Llista de taules**

Taula 1. Planificació

Taula 2. Riscos

Taula 3. Accions

Taula 4. Potències de transmissió

Taula 5. Principals característiques dels mòduls escollits

Taula 6. Sensors i principals característiques

Taula 7. Arxius que componen el programa

Taula 8. Pressupost

# 1. Introducció

## 1.1 Context i objectius

Vivim envoltats de dades i necessitem tenir control i comunicació permanent amb tots els elements que ens envolten, per això utilitzem les xarxes de telefonia mòbil, Wifi, Wimax o qualsevol altre mètode que ens permeti la connexió a Internet.

Tota aquesta tecnologia també s'aplica al món de la indústria per tal de poder obtenir dades que permetin millorar la quantitat i la qualitat del treball un cop analitzades.

Entre aquestes aplicacions industrials hi ha el disseny per supervisar un sensor de temperatura via Bluetooth objecte d'aquest treball. Això permetrà que qualsevol persona amb un dispositiu Bluetooth rebí informació del sensor.

Els coneixements adquirits durant els estudis de Grau de Tecnologies de la Comunicació ens permetran fer aquest disseny en el que haurem de:

- Seleccionar els components adients per el projecte, mitjançant la documentació tècnica de cada component.
- Dissenyar la font d'alimentació per els diferents blocs del circuit a partir dels 24 V<sub>cc</sub> subministrats.
- Dissenyar els circuits d'adaptació entre les entrades i les sortides dels diferents mòduls que componen el circuit.

## 1.2 Enfocament i mètode

Per construir el supervisor de temperatura via Bluetooth, s'han de dissenyar i construir les diferents etapes del següent diagrama de blocs, obtenint així el funcionament desitjat:

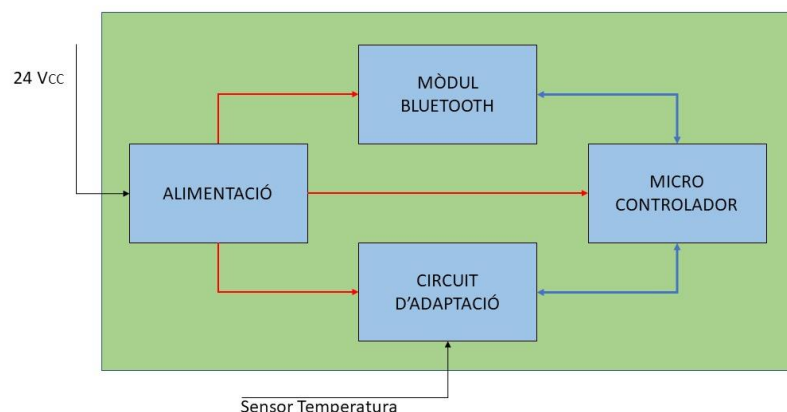


Figura 1. Diagrama de blocs

Les principals etapes per construir aquest model són:

- Cercar i escollir el sensor de temperatura més adient.



- Dissenyar el circuit d'adaptació del senyal del sensor per tal de poder llegir els valors que dóna.
- Dissenyar un equip que controli tant els senyals que es reben del sensor (circuit d'adaptació) com les dades que es volen enviar al Bluetooth.
- Cercar i escollir un equipament Bluetooth que permeti transmetre les dades que es volen llegir en el receptor.
- Dissenyar un circuit d'alimentació per fer funcionar cada un dels elements a partir dels 24 V subministrats.
- Disseny de la placa PCB on s'ha de muntar cada un dels mòduls descrits.

### **1.3 Planificació**

Per portar a terme aquest treball es divideix en diferents tasques que s'aniran desenvolupant en el temps:

- 1) Elecció del treball:
  - a. Planificació.
  - b. Elaboració de documents.
  - c. Instal·lació del programari.
- 2) Escollir el mòdul Bluetooth:
  - a. Introducció a aquesta tecnologia.
  - b. Estudi d'opcions dels mòduls existents.
  - c. Justificació del mòdul escollit
- 3) Escollir el sensor de temperatura.
- 4) Disseny del circuit elèctric per adaptar el senyal analògic obtingut al microcontrolador.
- 5) Escollir el microprocessador per treballar amb les dades obtingudes i la comunicació amb el mòdul Bluetooth.
- 6) Disseny de la font d'alimentació que permetrà fer funcionar tots els elements del circuit.
- 7) Disseny de la placa PCB que integri tots els elements de les etapes anteriors.
- 8) Finalització del projecte:
  - a. Simulació del circuit complert.
  - b. Elaboració del pressupost.
  - c. Muntatge de l'audiovisual.
  - d. Conclusions. Problemes sorgits i solucions aportades.

El diagrama de Gantt queda així:

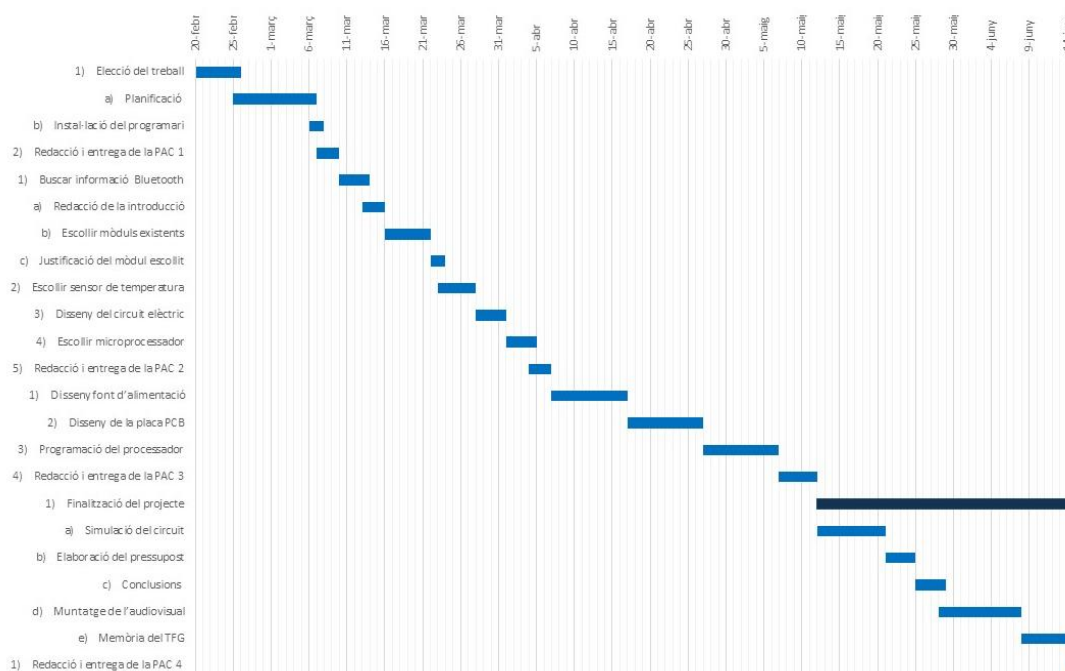


Figura 2. Diagrama de Gantt

| TASCA  | INICI           | FÍ              | HORES     |
|--|-----------------|-----------------|-----------|
| <b>PAC 1</b>                                       | <b>19/02/20</b> | <b>09/03/20</b> | <b>24</b> |
| 1) Elecció del treball                             | 20/02/20        | 25/02/20        | 3         |
| a) Planificació                                    | 25/02/20        | 06/03/20        | 12        |
| b) Instal·lació del programari                     | 06/03/20        | 07/03/20        | 1         |
| 2) Redacció i entrega de la PAC                    | 07/09/20        | 09/03/20        | 5         |
| <b>PAC 2</b>                                       | <b>10/03/20</b> | <b>06/04/20</b> | <b>33</b> |
| 1) Buscar informació sobre la tecnologia Bluetooth | 10/03/20        | 13/03/20        | 6         |
| a) Redacció de la introducció                      | 13/03/20        | 15/03/20        | 2         |
| b) Cercar i escollir mòduls existents              | 16/03/20        | 21/03/20        | 6         |
| c) Justificació del mòdul escollit                 | 22/03/20        | 23/03/20        | 2         |
| 2) Cercar i escollir sensor de temperatura         | 23/03/20        | 27/03/20        | 6         |
| 3) Disseny del circuit elèctric d'adaptació        | 28/03/20        | 31/03/20        | 6         |
| 4) Cercar i escollir microprocessador              | 01/04/20        | 04/04/20        | 6         |
| 5) Redacció i entrega de la PAC                    | 04/04/20        | 06/04/20        | 5         |
| <b>PAC 3</b>                                       | <b>07/04/20</b> | <b>11/05/20</b> | <b>35</b> |
| 1) Disseny de la font d'alimentació                | 07/04/20        | 16/04/20        | 10        |
| 2) Disseny de la placa PCB                         | 17/04/20        | 26/04/20        | 10        |
| 3) Programació del processador                     | 27/04/20        | 06/05/20        | 10        |
| 4) Redacció i entrega de la PAC                    | 07/05/20        | 11/05/20        | 5         |
| <b>PAC 4</b>                                       | <b>12/05/20</b> | <b>15/06/20</b> | <b>50</b> |
| 1) Finalització del projecte                       | 12/05/20        | 15/06/20        | 45        |
| a) Simulació del circuit complet                   | 12/05/20        | 20/05/20        | 10        |
| b) Elaboració del pressupost                       | 21/05/20        | 24/05/20        | 5         |
| c) Conclusions                                     | 25/08/20        | 28/05/20        | 5         |
| d) Muntatge de l'audiovisual                       | 28/05/20        | 07/06/20        | 15        |
| e) Memòria del TFG                                 | 08/06/20        | 13/06/20        | 10        |
| 2) Redacció i entrega de la PAC                    | 14/06/20        | 15/06/20        | 5         |

Taula 1. Planificació

## 1.4 Incidències i riscos

El projecte no s'escapa de tenir uns riscos a l'hora de portar-lo a terme, a les dues taules següents podem veure els riscos previstos (taula 2) i les possibles accions (taula 3) per disminuir o impedir l'impacte sobre el treball.

| Codi | Nom                                 | Conseqüència                                  | Impacte   | Probabilitat |
|------|-------------------------------------|---|-----------|--------------|
| R01  | Errada de la xarxa elèctrica        | Impedeix avançar en el treball                | Alt       | Molt baixa   |
| R02  | Ordinador no funciona               | Impossible tirar endavant amb el treball      | Molt alt  | Baixa        |
| R03  | Errada del programari               | No es podrà treballar amb el programa afectat | Alt       | Baixa        |
| R04  | Errada de connexió a Internet       | No podrem connectar per obtenir dades         | Mig       | Molt baixa   |
| R05  | Pèrdua de la feina feta             | Haurem de tornar a començar amb els treballs  | Alt       | Baixa        |
| R06  | Retras en les dates de planificació | No complirem els objectius previstos          | Baix      | Mig          |
| R07  | Imprevistos familiars               | Retard en el objectius previstos              | Alt       | Mig          |
| R08  | Malaltia                            | Retard en els objectius previstos             | Mig – Alt | Baixa        |

Taula 2. Riscos

| Codi   | Acció  | Tipus      | Risc residual |
|--------|--|------------|---------------|
| A01R01 | Canviar l'horari de dedicació  | Correctora | Baix          |
| A02R01 | Recuperar el temps perdut augmentant la càrrega de treball en altres dates | Correctora | Baix          |
| A01R02 | Fer còpies de seguretat per poder treballar des d'un altre ordinador       | Mitigadora | Mitjà         |
| A01R03 | Reinstal·lació del programa que falla                                      | Mitigadora | Mitjà         |
| A01R04 | Buscar connexió alternativa en el mòbil o un altre encaminador             | Correctora | Baix          |
| A01R05 | Tenir còpies de seguretat  | Mitigadora | Mitjà         |
| A01R06 | Augmentar les hores de treball dedicades                                   | Correctora | Baix          |
| A01R07 | Recuperar les hores perdudes amb més dedicació la resta de dies.           | Correctora | Baix          |
| A01R08 | Recuperar les hores perdudes amb més dedicació la resta de dies.           | Correctora | Baix          |

Taula 3. Accions

## 2. Bluetooth

Bluetooth és una tecnologia sense fils utilitzada per intercanviar dades entre dispositius en distàncies curtes (màxim 100 metres). Utilitza una amplada de banda ISM (Industrial, Scientific, Medical) de 2.4 a 2.485 GHz, per tant, està lliure d'obtenir llicència en tractar-se d'una banda lliure de l'espectre electromagnètic.

El Bluetooth és un dels tipus industrials que permeten construir xarxes PAN (Personal Area Network) sense fils, pensades per substituir a la comunicació mitjançant cable de dades RS-232.

## 2.1 Història

La tecnologia de ràdio de curta distància, que després s'anomenaria Bluetooth, la va començar Ericsson l'any 1989 per desenvolupar un auricular sense fils.

El 1997 Ericsson i IBM van acordar integrar aquesta tecnologia en un ordinador (IBM) i un telèfon (Ericsson) per poder-los connectar.

El 1998 després de mantenir contactes amb Intel, Toshiba i Nokia van crear Bluetooth SIG (Special Interest Group). Aquest grup s'encarrega de dictar els estàndards d'aquesta tecnologia i totes les seves especificacions. Amb el pas del temps aquest grup ha anat creixent fins a més de 30.000 companyies que en formen part en l'actualitat.

Bluetooth és un sistema de compatibilitat descendent, és a dir, l'estàndard més recent és compatible amb totes les versions anteriors.

## 2.2 Característiques

La tecnologia Bluetooth funciona entre 2.40 i 2.485 GHz amb bandes de protecció de 2 MHz a l'extrem inferior i 3.5 MHz a la part superior. Per la transmissió utilitza la tècnica anomenada salts de freqüència d'espectre ample (SSFH: Spread Spectrum Frequency Hopping) on cada equip de la xarxa utilitza de manera aleatòria 79 freqüències del mateix rang per enviar els paquets de dades protegint-se així de les interferències i transposicions.

La freqüència canvia de manera constant 1600 vegades per segon, així cada equip de la xarxa pot aprofitar l'espectre sencer.

L'arquitectura del Bluetooth és mestre / esclau. Un mestre pot comunicar-se amb fins a set esclaus dins un piconet (xarxa informàtica on els nodes estan connectats mitjançant Bluetooth). Tots els esclaus estan sincronitzats pel rellotge del mestre de la seva xarxa.

El protocol està basat en paquets de dades que modula en GFSK (modulació per desplaçament gaussià de freqüència) o PSK (modulació per desplaçament de fase).

Segons la potència de transmissió tenim quatre classes de dispositius:

| Classe | Potència |     | Rang (m) |
|--------|----------|-----|----------|
|        | mW       | dBm |          |
| 1      | 100      | 20  | 100      |
| 2      | 2.5      | 4   | 20       |
| 3      | 1        | 0   | 1        |
| 4      | 0.5      | -3  | 0.5      |

Taula 4. Potències de transmissió

La més utilitzada en aparells mòbils és la classe 2 i en ús industrial la classe 1.

Segons el protocol utilitzat tenim cinc versions, algunes amb actualitzacions:

- Bluetooth 1.0.- amb molts problemes a l'inici per connectar dispositius va aconseguir solucionar-ho amb dues actualitzacions (1.1 i 1.2). La taxa de transmissió era de 721 Kbps.






- Bluetooth 2.0.- van millorar amb l'actualització 2.1 + ERD (Enhanced Data Rate: taxa de dades millorada) la connectivitat i potència amb taxes de 2 Mbps.
- Bluetooth 3.0.- va millorar la velocitat amb l'anomenat HS (High Speed) que permet transmetre paquets més llargs. La taxa de transmissió arriba a 24 Mbps.
- Bluetooth 4.0.- té dues actualitzacions (4.1 i 4.2) i millora el consum d'energia amb l'aparició del BLE (Bluetooth Low Energy) i la connexió amb dispositius que funcionin amb IoT (Internet of Things). La seva taxa va de 25 Mbps fins a 32 Mbps.
- Bluetooth 5.0.- és la versió més recent però ja compte amb dues actualitzacions (5.1 i 5.2) dirigida principalment a la IoT amb una doble taxa de transferència i la possibilitat de rebre flux de dades de diferents dispositius alhora.

### 2.3 Cercar i escollir mòdul

Al mercat ja trobem diferents mòduls que ja contenen l'antena tant per connectar mitjançant cables com per soldar al circuit imprès on ha de funcionar.

Hi ha molts fabricants: STMicroelectronics, Texas Instruments, Qualcomm, Panasonic, etc...

Després d'estudiar diferents models, tots amb un abast d'almenys 100 m (classe 1) per què s'adapti millor a l'aplicació industrial que es necessita, amb antena integrada i tot encapsulat per disminuir el número d'interferències s'opta per reduir la llista a aquests cinc:

| Model           | BT 730-SA   | BT 740-SA   | RN 41 XVC   | RN 41-DS  | WT 41 U   |
|-----------------|---|---|---|---|---|
| Foto            |  |  |  |  |  |
| Fabricant       | Laird Connectivity  |   | Microchip Technology  |   | Silicon Laboratories  |
| Tipus           | V2.0  | V2.1  | V2.1+EDR  | V2.1+EDR  | 2.1+EDR   |
| Interface       | UART, USB   | UART  | UART, USB   |   |   |
| Alimentació     | 3.5V  |   | 3.3V  |   |   |
| Consum          | 35mA  |   | 45mA  | 30mA  | 44mA  |
| Sensibilitat Rx | -87 dBm   |   | -80dBm  |   | -94dBm  |
| Potència Tx     | 18dBm   |   | 16dBm   | 15dBm   | 17dBm   |
| Mides (mm)      | 15.29x28.71   |   | 24.4x29.9   | 13.4x25.8   | 14x35.3   |
| Range           | >1000m  |   | 100m  |   | 650m  |
| Preu            | 21.29€  | 21.15€  | 26.06€  | 20.41€  | 17.96€  |

Taula 5. Principals característiques dels mòduls escollits

D'aquests cinc que tenen unes característiques similars es tria el BT730-SA que té un consum en mode transferència baix, és econòmic i té molta potència de transmissió amb una cobertura més que suficient pel disseny si s'implementa en una nau industrial.

Les característiques més destacades són:

- Versió 2.0
- Classe 1 amb 18 dBm de potència de transmissió i -87 dBm de sensibilitat en recepció.

- El consum va de 1.25 mA quan esta inactiu fins els 35 mA en màxima transferència de dades passant pels 6 mA quan està connectat sense transferència de dades.
- El rang de freqüències que utilitza va de 2.402 a 2.480 GHz.
- Té tres tipus d'Interface: UART, GPIO i ADC.
- La tensió d'alimentació va dels 3.3 als 5 V.
- La cobertura és superior als 1000 metres.

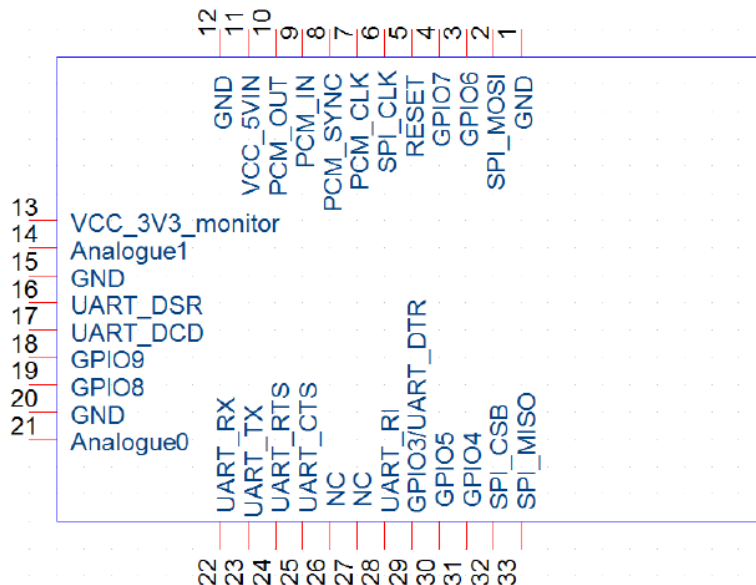


Figura 3: BT730-Sx module pin-out (Top View)

A la figura 3 es poden veure tots els pins del mòdul i les seves funcions. Destacar els pins 11 i 12 que són l'alimentació i els 23 i 24 que són la recepció i la transmissió en el sistema UART que s'utilitzarà pel seu funcionament. Altre és el 5 Reset.

### 3. Sensor de temperatura

#### 3.1 Descripció

És un dispositiu capaç de mesurar la temperatura del mitjà on és troba i comunicar-la a un altre dispositiu transmetent un senyal de manera analògica o digital.

Els sensors es poden dividir en:

- Passius: necessiten alimentació externa.
  - Resistius: varien la seva resistència segons varia la temperatura.
  - Capacitatius: varien la capacitat d'un condensador.
  - Inductius: varien la inductància d'una bobina.
- Actius: generen la seva pròpia energia.

Les principals característiques són:

- El rang de mesura: domini que pot mesurar.

- Precisió: error de mesura màxim.
- Sensibilitat: relació entre la variació de temperatura real i la temperatura mesurada.
- Resolució: variació mínima que veiem a la sortida.

Els tipus que trobem són:

- RTD (Resistance Temperature Detectors) són resistències que varien amb la temperatura, es fonamenten en el fet que la resistència dels metalls varia amb la temperatura.
- Termistor NTC (Negative Temperature Coefficient) és una resistència variable que decreix en augmentar la temperatura.
- Termistor PTC (Positive Temperature Coefficient) és una resistència variable que va augmentant en augmentar la temperatura.
- Bimetall: compost per dos metalls diferents, per tant amb diferent dilatació, que produeix variacions en la seva curvatura en variar la temperatura. Són digitals, llavors ja queden descartats per el treball on s'especifica que han de ser analògics.

### 3.2 Justificació

Es busca entre els altres tipus el nostre sensor, que ha de tenir un rang de +20° a +120°C i una precisió d'1°C.

Aquests rang de temperatures és típic en mesures de temperatures de líquids canalitzats, per tant, se'n busca un que s'adapti com els de S+S Realtechnik.

| SENSOR | CONSUM | RANG          | SENSIBILITAT                                   | PREU   |
|--------|--------|---------------|--|--------|
| Ni1000 | <0.3mA | -50 a +150 °C | 0.4+0.007 t                                    | 19.38€ |
| Pt1000 | <0.6mA | -50 a +150 °C | Class A: 0.15+0.002 t <br>Class B: 0.3+0.005 t | 18.16€ |

Taula 6. Sensors i principals característiques



Figura 4. Sensor

El Ni1000 es descarta per què la sensibilitat supera 1°C que és la precisió que ens demana el treball. Així, l'escollit és el S+S Realtechnik ALTF1 Pt1000 classe B, que utilitza la norma DIN EU 60751.

Per calcular la resistència de sortida ( $R_t$ ) entre 0°C i 850°C la fórmula que demana aquesta norma val:

$$R_t = R_0(1 + At + Bt^2) \quad (1)$$

on:

- $R_0$  és el valor de la resistència a 0°C (1000  $\Omega$ )
- $t$  el valor de la temperatura que mesurem i les constants A ( $3.9083 \cdot 10^{-3}$ ) i B ( $-5.775 \cdot 10^{-7}$ ).

Fent els càlculs pel rang desitjat s'obté la gràfica de la figura 5 on es poden comparar amb els valors donats pel fabricant i observar que la variació és completament lineal, fet que facilita la feina.

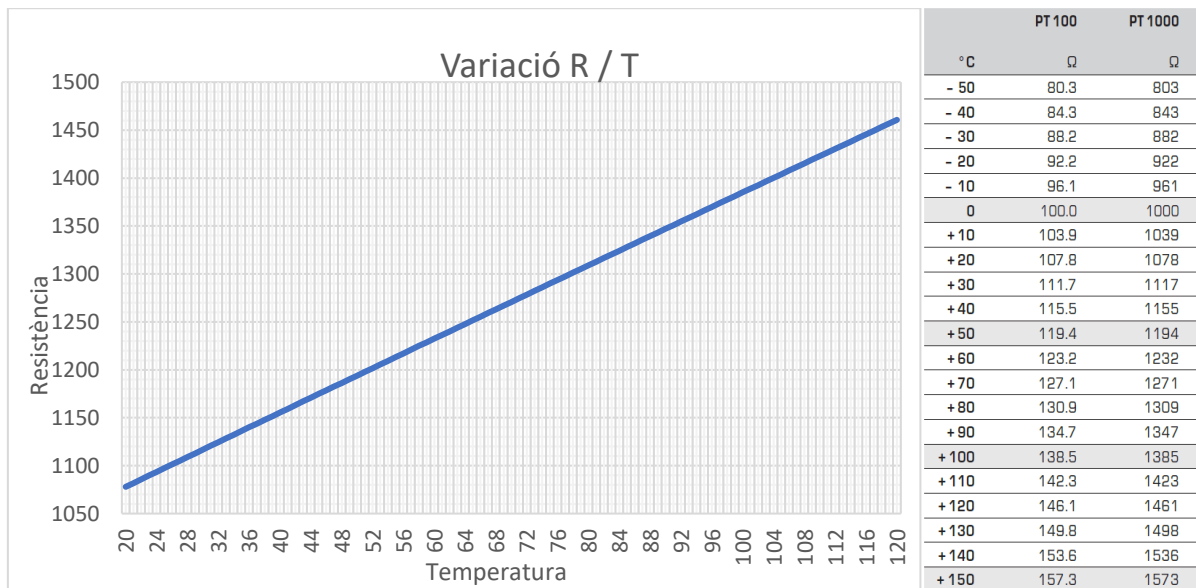


Figura 5. Resposta de la resistència a la temperatura i dades del fabricant

## 4. Disseny del circuit elèctric d'adaptació

Per què el sensor envii les dades al mòdul Bluetooth s'ha de dissenyar un circuit electrònic capaç de transformar la variació de la resistència del sensor en un voltatge per entregar-lo a un microprocessador i aquest processar les dades i enviar-les pel Bluetooth.

### 4.1 Circuit lector de resistència

#### 4.1.1 Pont de Wheatstone

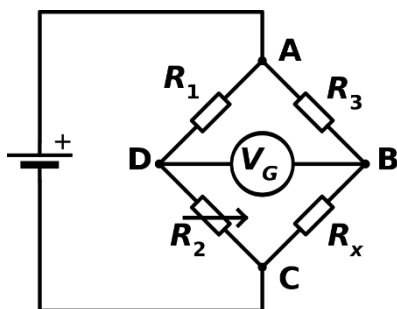


Figura 6. Pont Wheatstone

El pont de Wheatstone consisteix en un circuit tancat format per quatre resistències en un perfecte equilibri entre els braços del pont.

Quan una de les quatre resistències varia, el sensor per exemple, provoca una variació del voltatge entre els punts D i B poden saber el nou valor fàcilment aplicant Kirchoff entre els dos punts divisors i restant-los.

$$V_G = \left( \frac{R_2}{R_1 + R_2} - \frac{R_x}{R_x + R_3} \right) V_S \quad (2)$$

Així si  $R_x$  és el sensor capturant la tensió entre els punts DB es podrà fer un circuit que detecti les variacions de la resistència del sensor.



### 4.1.2 Lectura de la resistència

Per llegir els valors de resistència del sensor es fa servir un pont de Wheatstone, on s'ajusta el valor de les resistències perquè quan la temperatura val 20° C, que és la mínima que es vol llegir, s'obtingui 0 V a la sortida.

Per calcular la resistència s'utilitza la fórmula donada per la norma EN60751 que detalla com influeix l'increment de temperatura sobre el sensor.

$$R_t = R_0(1 + AT + BT^2) \quad (3)$$

On

$R_T$  és el valor de la resistència a la temperatura llegida.

$R_0$  és el valor de la resistència quan  $T = 0^\circ \text{C}$  (1000  $\Omega$ )

A i B constants del tipus de sensor, l'utilitzat és de platí i valen:

$$A = 3.9083 \cdot 10^{-3}$$

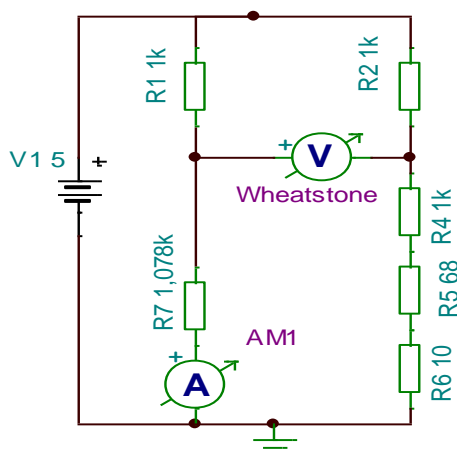
$$B = -5.775 \cdot 10^{-7}$$

T és la temperatura real que estem mesurant.

Així:

$$R_{20} = 1000(1 + 3.9083 \cdot 10^{-3} * 20 - 5.775 \cdot 10^{-7} * 20^2) = 1078$$

$$R_{120} = 1000(1 + 3.9083 \cdot 10^{-3} * 120 - 5.775 \cdot 10^{-7} * 120^2) = 1469$$



Amb aquests valors es munta el pont de Wheatstone de la figura 7.

$$R1=R2=1 \text{ K}\Omega$$

$$R4=1 \text{ K}\Omega$$

$$R5=68 \Omega$$

$$R6=10 \Omega$$

$$R7=1078 \Omega \text{ a } 20^\circ \text{C (és el sensor)}$$

Figura 7. Pont Wheatstone del circuit.

La combinació de R4, R5, R6 dóna 1078, igual que el sensor, per tal que quan hi hagi 20°C la sortida valgui 0V.

Un problema que sorgeix és que el pas del corrent pel sensor és molt gran (2.407mA) el que fa que la resistència variï donant errors de lectura, per solucionar-ho s'hi posa un regulador d'intensitat amb l'operacional OPA 333 que s'alimenta a 5V igual que la resta del circuit.

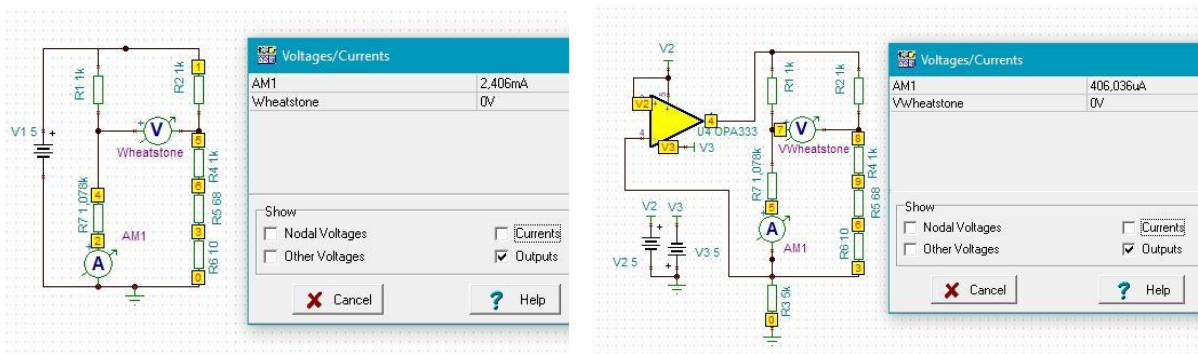


Figura 8. Pont Wheatstone sense regulació d'intensitat i amb regulació d'intensitat

Com es veu als esquemes amb la introducció de l'operacional aconseguim que quan la intensitat és màxima caigui dels 2.407 mA fins els 406.036  $\mu$ A, molt per sota dels 6 mA de tolerància del sensor. I la tensió és manté a 0 V.

Per trobar els valors de la resistència que regula la intensitat s'aplica la fórmula

$$G = 1 + \frac{R_{Wheat}}{R3} \quad (4)$$

on aïllem R3 i queda  $R3 = \frac{R_{Wheat}}{G-1} = \frac{1038.5}{1.20-1} = 5192.5 \Omega \approx 5 K\Omega$ .

Un altre problema que sorgeix és que les tensions són molt petites per poder llegir-les correctament al microcontrolador, per solucionar-ho s'utilitza un amplificador amb el mateix model d'operacional. Perquè funcioni correctament sense que la resistència del pont influeixi sobre la impedància del circuit hi ha d'haver un circuit adaptador d'impedàncies. Com el del dibuix:

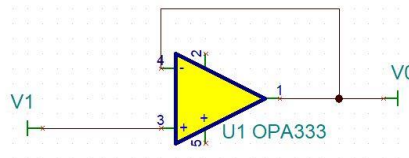


Figura 9. Adaptador d'impedàncies

I després es pot afegir l'amplificador per obtenir els valors desitjats. Per calcular el guany que necessari per obtenir 3.5V a la sortida del circuit quan el sensor té la resistència màxima (1461  $\Omega$ ) es representa el circuit amb el programa TINA-TI i es pot llegir la tensió al voltímetre de sortida dels operacionals.

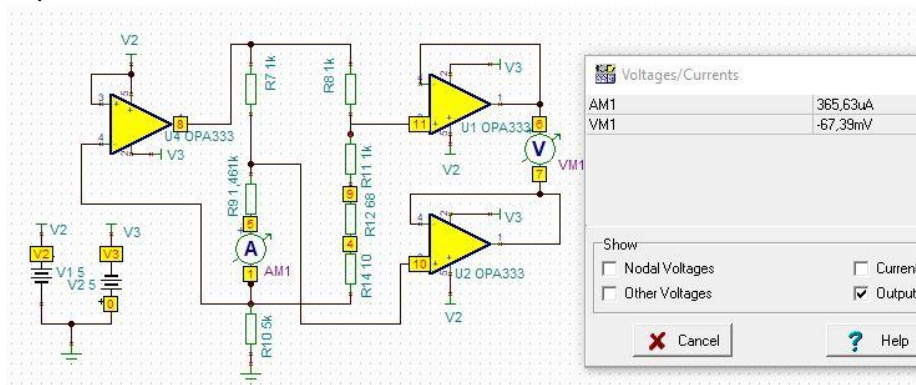


Figura 10. Lectura de tensió d'entrada a l'amplificador

$$\text{Així } \Delta V = \frac{V_0}{V_i} = \frac{3.5}{67,39 \cdot 10^{-3}} = 51.936$$

La tensió de sortida és 3.5V perquè és el valor que valdrà l'1 lògic a l'entrada del microprocessador.

El guany dependrà de les resistències. Es calcula el valor:

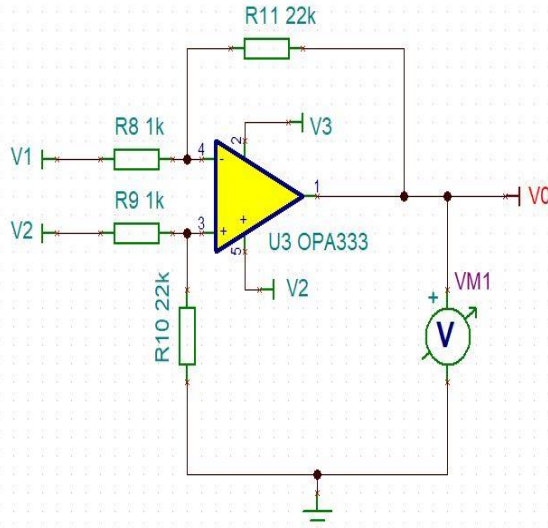


Figura 11. Amplificador

- Entrada inversora

$$\frac{V_1 - V_i}{R_8} = \frac{V_i - V_0}{R_{11}}$$

$$R_{11}V_1 - R_{11}V_i = R_8V_i - R_8V_0$$

$$R_{11}V_1 + R_8V_0 = R_8V_i + R_{11}V_i$$

$$V_i(R_8 + R_{11}) = R_{11}V_1 + R_8V_0$$

$$V_i = \frac{R_{11}V_1 + R_8V_0}{R_8 + R_{11}}$$

- Entrada no inversora

$$V_i = \frac{R_{10}}{R_9 + R_{10}}V_2$$

Per facilitar els càlculs  $R_8=R_9$  i  $R_{10}=R_{11}$  i s'iguala

$$\frac{R_{10}V_1 + R_8V_0}{R_8 + R_{10}} = \frac{R_{10}}{R_8 + R_{10}}V_2 \Rightarrow R_{10}V_1 + R_8V_0 = R_{10}V_2 \Rightarrow R_8V_0 = R_{10}V_2 - R_{10}V_1 \Rightarrow R_8V_0 = R_{10}(V_2 - V_1)$$

$$\frac{V_0}{(V_2 - V_1)} = \frac{R_{10}}{R_8} \text{ com que sabem que } V_i = (V_2 - V_1) \text{ llavors } \Delta V = \frac{R_{10}}{R_8}$$

Es dona el valor d'1KΩ a R8 (R9) i trobem que  $R_{10} (R_{11}) = 1K * 51,936 = 51.936 \approx 56K\Omega$  que és un valor estàndard de mercat i dona una mica més de guany que si arrodoneixo per sota a 51K.

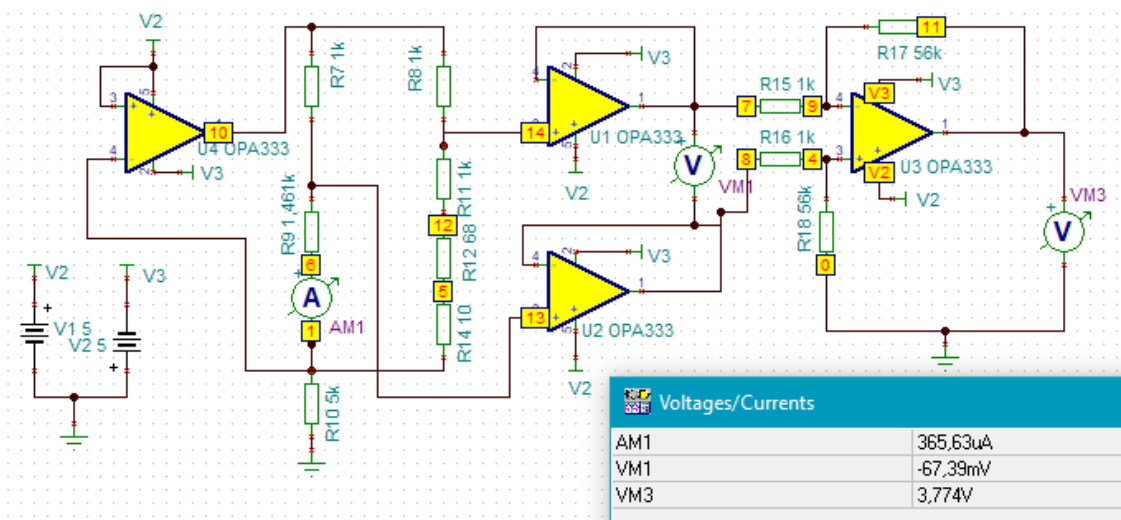


Figura 12. Valors de tensió per l'1

Un cop tot el circuit està muntat es poden llegir els valors de tensió a la sortida per la temperatura mínima i màxima.

$T(20^{\circ}) = 337,56 \mu\text{V}$  que serà el 0 lògic.

$T(120^{\circ}) = 3,774 \text{ V}$  que serà l'1 lògic.

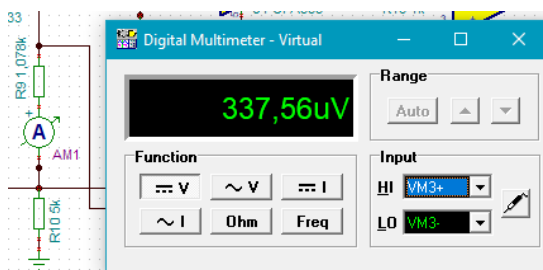


Figura 13. Tensió 0 lògic

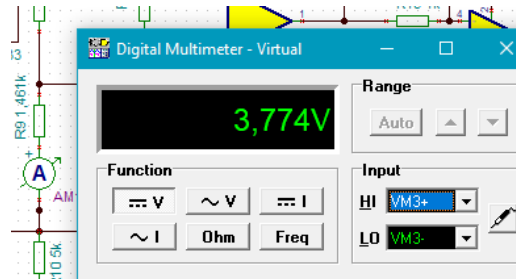


Figura 14. Tensió 1 Lògic

## 5. Circuit elèctric

La font d'alimentació subministrada és de  $24 \text{ V}_{\text{CC}}$  els quals s'adapten a les tres sortides necessàries per l'alimentació dels dispositius del circuit.

Cada una de les sortides esta equipada amb condensadors per filtrar les altes freqüències que puguin sortir dels operacionals, i per motiu d'aquests operacionals i el seu guany en laç obert tenim una impedància de sortida molt baixa facilitant l'adaptació amb les etapes següents del circuit.

### 5.1 Font d'alimentació de 5 V positius i negatius

A partir dels  $24 \text{ V}$  subministrats es fa una reducció a  $11 \text{ V}$  (Figura 16) mitjançant un díode Zener ( $12 \text{ V}$ ) i un transistor ( $2\text{N}2222$ ). Aquesta tensió serveix d'entrada a la font muntada amb l'amplificador operacional LM318, trobada en un vídeo de youtube<sup>1</sup> que ens serveix per reduir la tensió a la meitat i obtenir la sortida positiva i negativa de  $5 \text{ V}$  necessàries per alimentar els operacionals OPA333 utilitzats en la lectura del pont de Wheatstone.

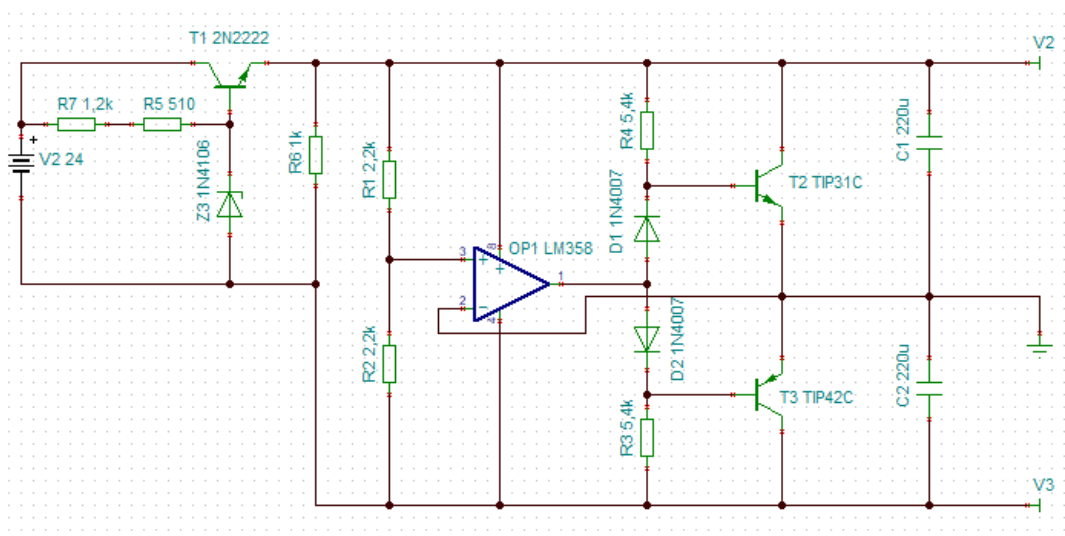


Figura 15. Font d'alimentació  $\pm 5 \text{ V}$ .

La regulació de tensió mitjançant l'operacional es controla amb un divisor de dues resistències i la tensió de sortida mitjançant dos transistors, un TIP31C tipus NPN per controlar la branca de sortida positiva i un TIP42C tipus PNP per la branca negativa.

El funcionament de les dues branques dels voltatges de sortida és similar, si el voltatge de sortida cau, el voltatge del terminal inversor també, fent que augmenti la sortida de l'operacional que provoca una major conducció dels transistors establint així el circuit.

A la sortida V2 hi ha els 5 V positius i a la sortida V3 els 5 V negatius, i com es veu en l'esquema tenim el punt de referència o terra entre els dos emissors dels transistors.

### 5.2 Font d'alimentació 3.5 V

La tensió es regula amb un amplificador operacional  $\mu A741$  (Figura 3) on es controla la tensió de referència mitjançant un díode Zener (1N3823) a 3.9 V i la tensió de sortida per un transistor 2N3440. El funcionament és molt similar a la font de  $\pm 5$  V. Prenent d'alimentació d'entrada els +5 V de l'altra font.

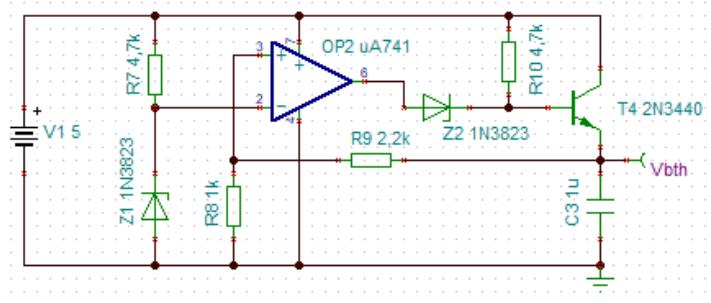


Figura 16. Font d'alimentació 3.5

L'entrada inversora es manté constant amb el díode Zener Z1 (1N3823) i la no inversora a través de la resistència R9 detecta les variacions de la sortida, així si la tensió de sortida cau també cau la sortida de l'amplificador operacional augmentant la conducció del transistor 2N3440, per tant, la tensió a la sortida i establint així el circuit. El díode Zener Z2 ens proporciona estabilitat a la sortida fins el punt de saturació del transistor.

### 5.3 Disseny del circuit

En l'apartat 4 s'hi troba el circuit de lectura de temperatura mitjançant el pont Wheatstone i els valors de tensió obtinguts a la sortida segons la temperatura llegida pel sensor Pt1000, ara s'introdueix aquest circuit amb el de les fonts d'alimentació per poder veure els resultats i que aquests no afecten a l'alimentació encara que variï el valor de la temperatura.

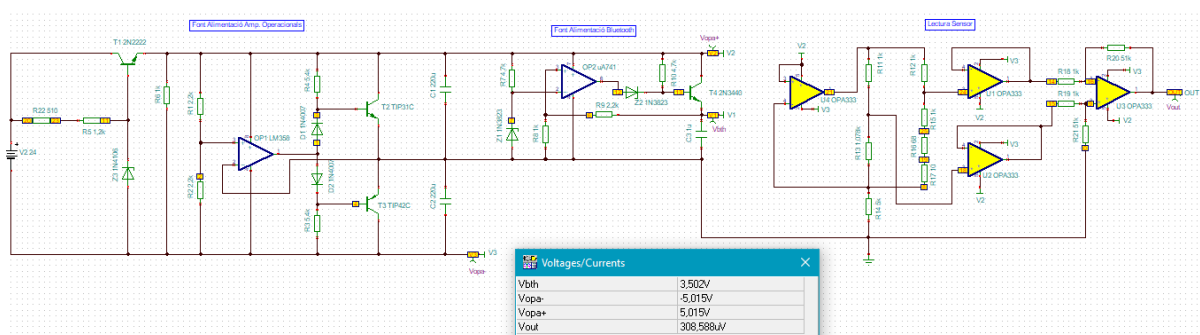


Figura 17. Tensions de sortida quan  $T=20$  °C

A la figura 18 veiem que la sortida positiva d'alimentació dels operacionals OPA333 val 5.015 V i la negativa -5.015 V (identificades com  $V_{opa+}$  i  $V_{opa-}$  respectivament) la d'alimentació del Bluetooth ( $V_{Bth}$ ) és de 3.502 V i la tensió de sortida que llegirà el convertidor ADC,  $V_{out}$ , 308.588  $\mu$ V per la temperatura mínima de 20° C.

Veiem a la Figura 18 que aquests valors d'alimentació no varien però la sortida cap el convertidor ara val  $V_{out} = 3.447$  V ja que el valor de la temperatura és de 120° C.

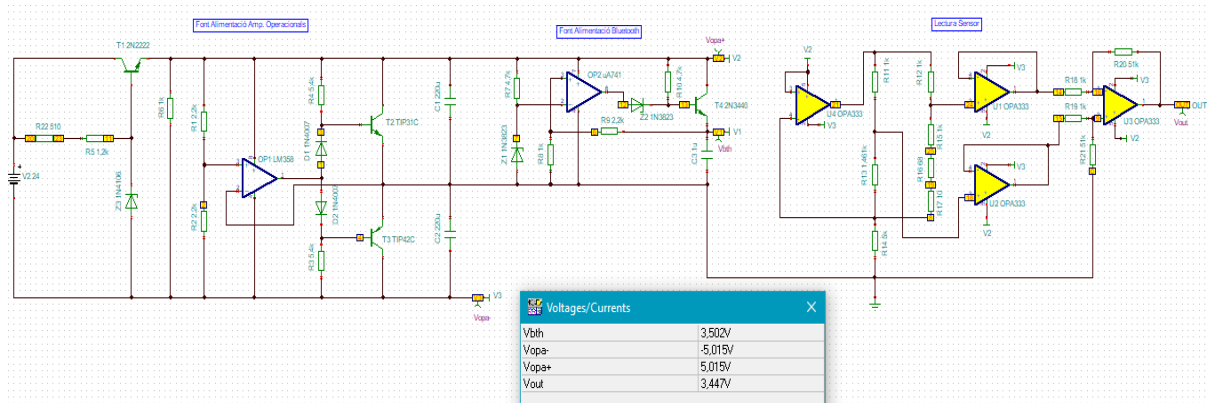


Figura 18. Tensions de sortida quan  $T=120$  °C

Es pot veure a la figura 19 la gràfica obtinguda en fer la simulació del funcionament del circuit quan la resistència de sortida va variant talment com si la temperatura augmentés gradualment.

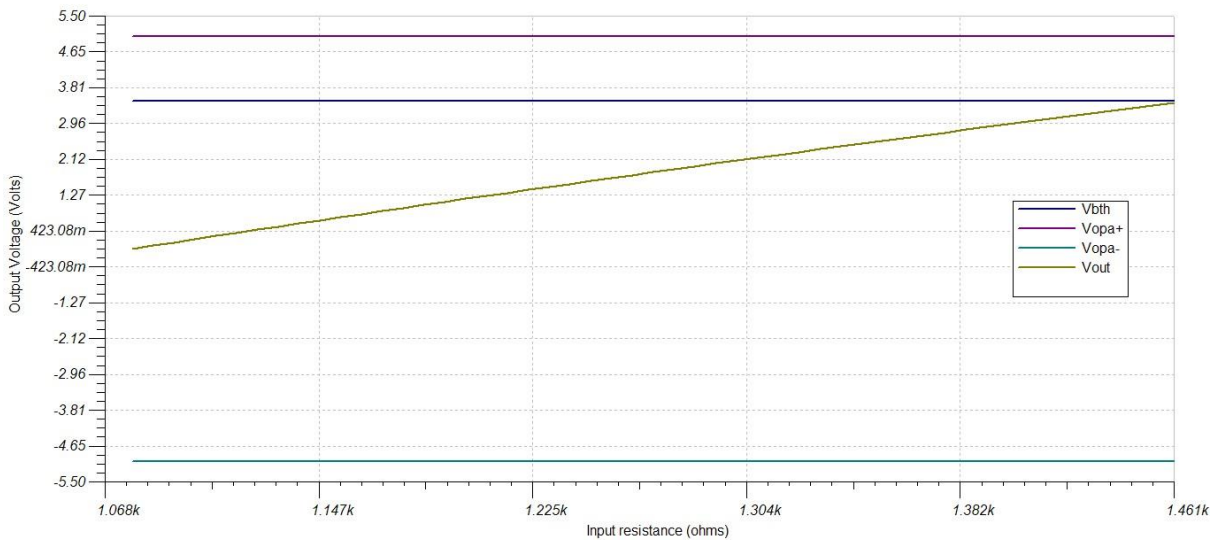


Figura 19. Simulació del sensor

L'aplicació TINA-TI té un control anomenat "select control object" que permet programar els valors inicial i final del component escollit, en aquest cas la resistència R13 que representa el sensor de temperatura. Així és veu com les alimentacions són estables i com la tensió de la sortida varia amb el augment de la resistència a l'entrada.

### 5.4 Disseny complet

Un cop estabilitzats el circuit a les tensions necessàries a la sortida s'afegeix el processador i el mòdul Bluetooth al circuit elèctric (figura 20) mitjançant l'aplicació Eagle, que també servirà per obtenir la placa PCB.

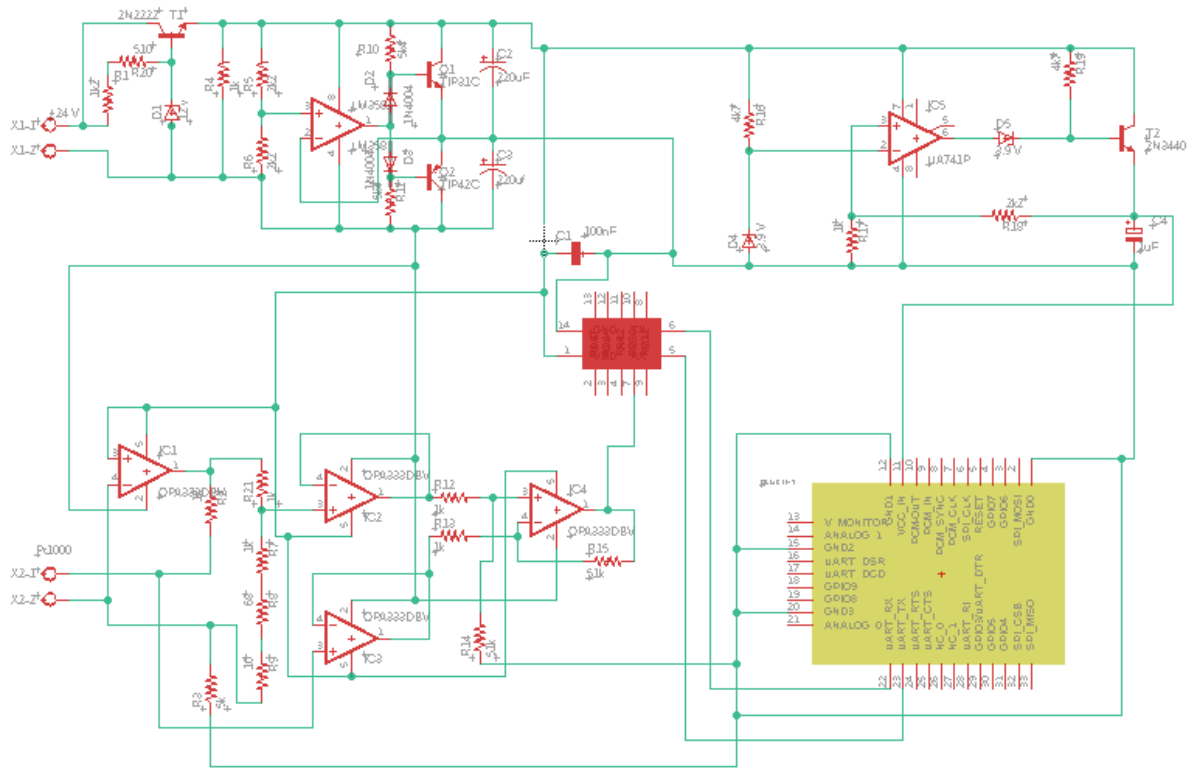


Figura 20. Esquema complet

Es pot veure que del processador només s'utilitzen els pins d'alimentació, un pel convertidor ADC i els dos pins necessaris per la transmissió i la recepció UART. Del mòdul Bluetooth, tot hi poder veure més connexions, es pot observar que la majoria són a terra i llavors hi ha l'entrada d'alimentació positiva i els dos pins necessaris per la transmissió i la recepció.

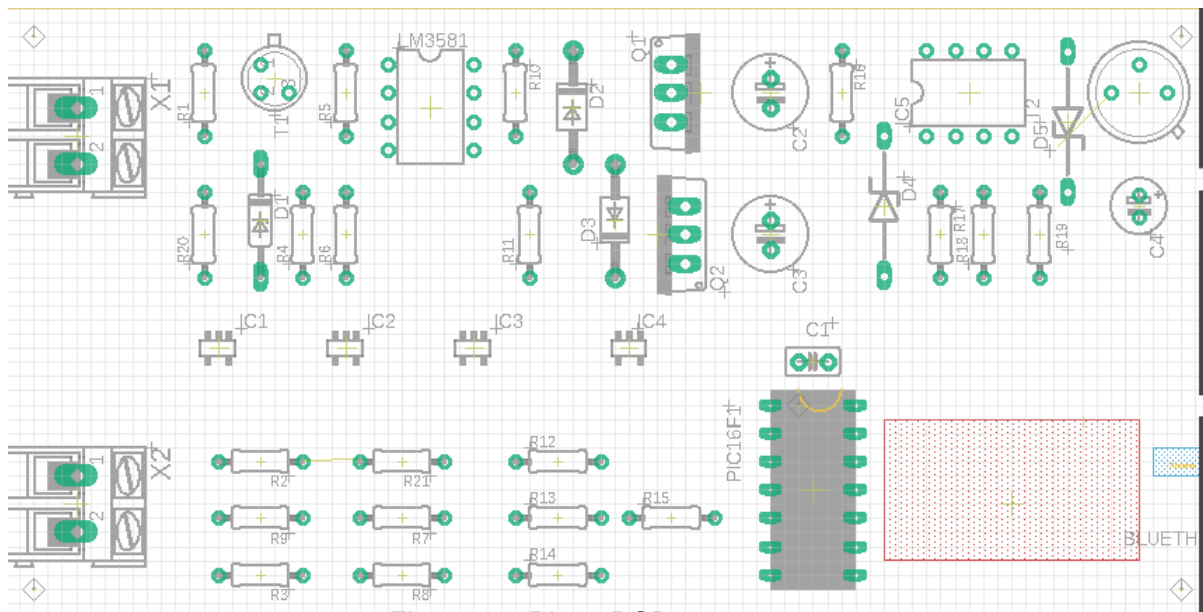


Figura 21. Placa PCB components

A la figura 21 es veu com queda la placa PCB muntada, amb els components i el Bluetooth en una cantonada tal com marquen els datasheets de l'element. La placa tindrà pistes per els dos costats per evitar ponts i perquè tan el mòdul Bluetooth com els operacionals OPA333 tenen encapsulat SOT-23 que va soldat per la cara superior ja que els pins no travessen la placa.

A la figura 22 es veuen les pistes de sota i a la figura 23 les pistes del cantó components.

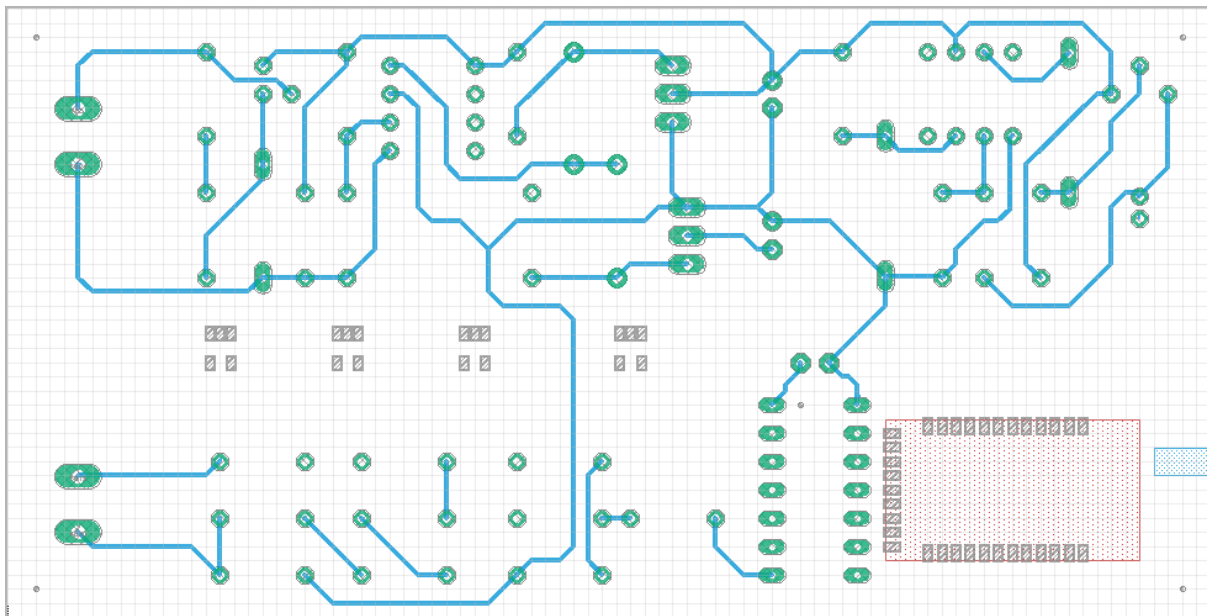


Figura 22. Pistes cara de sota

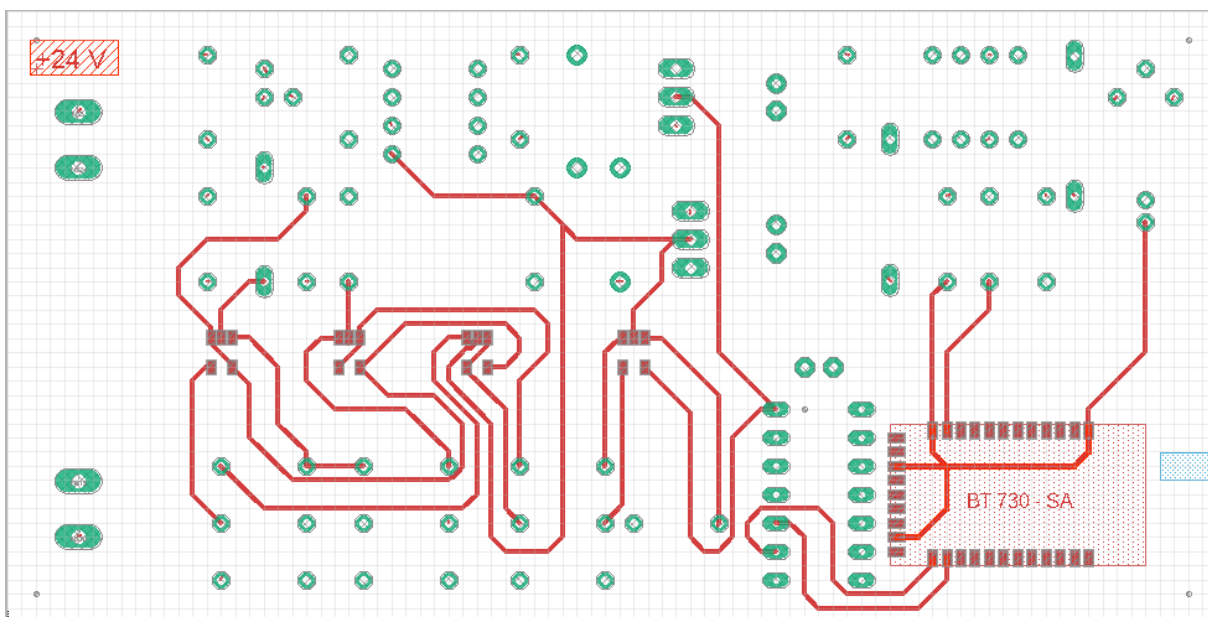


Figura 23. Pistes cara components



I el circuit complet es veu com a la figura 24.

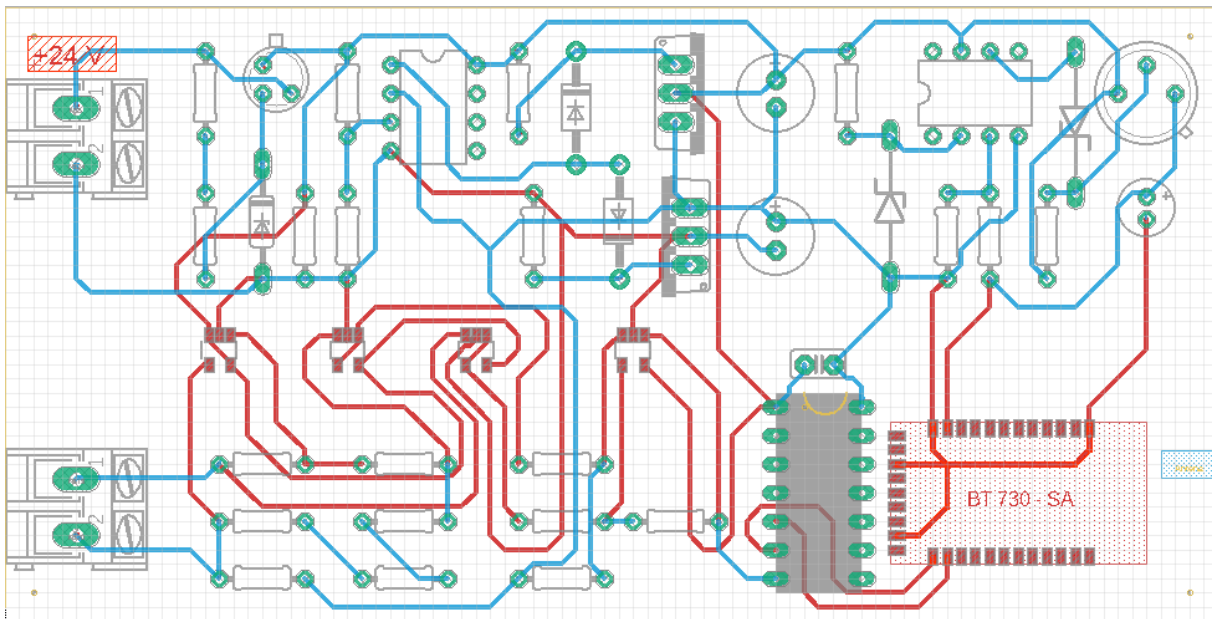


Figura 24. Placa PCB completa

## 6. Microcontrolador

El microcontrolador és un sistema encarregat de controlar el funcionament d'un dispositiu, per exemple controlar des de l'adquisició de temperatura del sensor fins el mòdul Bluetooth. Del microcontrolador dependrà el correcte funcionament de tot el circuit.

### 6.1 Característiques

Un microcontrolador és un sol circuit integrat especialitzat en controlar equips electrònics. Es considera un computador dedicat.

Aquest integrat inclou una CPU (unitat central del procés), una memòria, unitats d'entrada, unitats de sortida i una altre memòria no volàtil per guardar el programa, tipus ROM, EPROM, FRAM, etc...

El propòsit general del microcontrolador és llegir i executar els programes que l'usuari escriu prèviament en la seva memòria. Un mateix circuit pot servir per diferents tasques només canviant el programa, això simplifica el disseny de circuits electrònics.

Treballen a velocitats molt baixes comparats amb els microprocessadors i això els permet uns consums molt baixos, moltes vegades tenen el rellotge integrat que els facilita la sincronització amb els perifèrics que pugui controlar.

Els tipus de construccions o arquitectures més populars són la RISC (Reduced Instruction Set Computer) i la CISC (Complex Instruction Set Computer). La principal diferència és la complexitat de les instruccions que utilitzen.

## 6.2 Programació

Els microcontroladors estan dissenyats per processar dades en forma binària, per facilitar la programació hi ha compiladors de llenguatges C o BASIC al llenguatge màquina.

Un altre llenguatge molt utilitzat tot i que especialitzat és l'assemblador, que és una llista d'instruccions de paraules abreujades.

## 6.3 Selecció del microcontrolador

Hi ha molts fabricants de microcontroladors, com Texas Instruments, Motorola, Microchip (ATMEL) Intel, STMicroelectronics, etc...que ofereixen més o menys les mateixes característiques.

Per triar quin s'utilitza en el treball es té en compte el següent:

- Disponibilitat en el mercat, que sigui molt utilitzat perquè en cas de problemes hi hagi recanvi.
- Disponibilitat d'informació i eines de desenvolupament.
- Preu.
- L'aplicació on s'ha d'utilitzar, tipus d'entrades / sortides necessàries, alimentació, etc...

Per tot això s'escull un model de Microchip del tipus PIC, que són molt universals i econòmics, tenen molta informació a la xarxa i el seu programa Mplab disposa de compiladors de llenguatge C que en facilita la tasca.

S'utilitza el PIC16F18325, que encara s'està produint i val 1,01 € si se'n compra només un.

- Consta d'una arquitectura tipus RISC amb compilador optimitzat de C, té memòria SRAM d'1 KB tipus Flash i 256 Bytes de ROM tipus EEPROM.
- Pot treballar entre 1.8 i 5.5 V que el fa compatible amb les dues alimentacions que hi ha al circuit.
- La seva resolució ADC (Analog-to-Digital Converter) és molt gran (10 bits) i el rellotge intern té una freqüència màxima de 32 MHz, també consta d'un sistema Watchdog per vigilar les errades de programari i provocar el reinici.

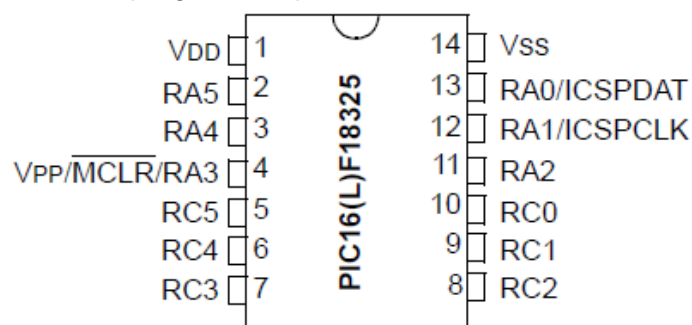


Figura 25. Distribució dels PIN

A la placa si munta un sòcol i així s'instal·la el microcontrolador un cop programat i s'evita el perill de manipulació, encara que es pot evitar per programació.

## 7. Programació del processador

### 7.1 Introducció

Per la programació del processador s'ha utilitzat el programa MPLAB de Microchip. S'han de configurar els pins de recepció i transmissió de l'UART, l'entrada de l'ADC (convertidor analògic a digital) que llegirà el valor de tensió provocat per la variació de temperatura, els reinicis en cas de caiguda de tensió i el rellotge intern, per al sincronisme de les operacions, la velocitat de transmissió que s'utilitza per comunicar amb el mòdul Bluetooth i si la comunicació és síncrona o asíncrona.

Per la programació i entrada de totes aquestes dades s'utilitza la eina MCC (MPLAB Code Configurator) del propi MPLAB i es fan les variacions necessàries en l'arxiu main. Així, un cop acabada la configuració obtenim els arxius de la taula 1.

| Fitxers desenvolupament | Fitxers definicions | Descripció  |
|-------------------------|---------------------|---|
| device_config.c         | device_config.h     | Configuració general del microcontrolador (rellotge, interrupcions, etc...) |
| pin_manager.c           | pin_manager.h       | Configuració dels pins (IO, ADC, UART, etc...)                              |
| adc.c                   | adc.h               | Configuració del convertidor analògic digital                               |
| eusart.c                | eusart.h            | Configuració de la comunicació UART.  |
| mcc.c                   | mcc.h               | Inicialització del sistema  |
| main.c                  |                     | Control de les operacions i dades del programa                              |

Taula 7. Arxius que componen el programa

Els arxius amb el mateix nom fan les mateixes funcions però amb el treball repartit per tal que el codi quedi més endreçat i clar. Normalment el fitxers header (.h) fan la definició de les funcions, estructures, macros i dades necessàries pel desenvolupament mentre els del mateix nom tipus c fan el desenvolupament del que s'ha presentat en els anteriors.

### 7.2 Device\_config

En aquests arxius s'hi troben les definicions generals del microcontrolador, com el tipus de rellotge i la freqüència en que treballa, interrupcions, programació del bit MCLR que permet l'entrada de dades en memòria, activació del bit de reinici per errada de voltatge (quan sigui menor de 2.45 V) reinici per desbordament de pila.

```
// CONFIG1
#pragma config FEXTOSC = OFF    // FEXTOSC External Oscillator mode Selection bits->Oscillator not enabled
#pragma config RSTOSC = HFINT32 // Power-up default value for COSC bits->HFINTOSC with 2x PLL (32MHz)
#pragma config CLKOUTEN = OFF    // Clock Out Enable bit->CLKOUT function is disabled; I/O or oscillator function on OSC
#pragma config CSWEN = ON       // Clock Switch Enable bit->Writing to NOSC and NDIW is allowed
#pragma config FCMEN = OFF     // Fail-Safe Clock Monitor Enable->Fail-Safe Clock Monitor is disabled

// CONFIG2
#pragma config MCLR = ON       // Master Clear Enable bit->MCLR/VPP pin function is MCLR; Weak pull-up enabled
#pragma config FWRT = OFF      // Power-up Timer Enable bit->PWRT disabled
#pragma config WDTE = OFF      // Watchdog Timer Enable bits->WDT enabled while running and disabled in SLEEP/IDLE; SWDTEN is
#pragma config LPBREN = OFF    // Low-power BOR enable bit->ULPBOR disabled
#pragma config BOREN = ON      // Brown-out Reset Enable bits->Brown-out Reset enabled, SBOREN bit ignored
#pragma config BORV = LOW      // Brown-out Reset Voltage selection bit->Brown-out voltage (Vbor) set to 2.45V
#pragma config PPS1WAY = ON    // PPSLOCK bit One-Way Set Enable bit->The PPSLOCK bit can be cleared and set only once; PPS
#pragma config STVREN = ON     // Stack Overflow/Underflow Reset Enable bit->Stack Overflow or Underflow will cause a Reset
#pragma config DEBUG = OFF     // Debugger enable bit->Background debugger disabled

// CONFIG3
#pragma config WRT = OFF       // User NVM self-write protection bits->Write protection off
#pragma config LVP = OFF       // Low Voltage Programming Enable bit->High Voltage on MCLR/VPP must be used for programming.

// CONFIG4
#pragma config CP = OFF        // User NVM Program Memory Code Protection bit->User NVM code protection disabled
#pragma config CPD = OFF       // Data NVM Memory Code Protection bit->Data NVM code protection disabled
```

Figura 26. device\_config.c

### 7.3 Pin\_manager

En aquests arxius es defineixen les entrades i sortides que el microcontrolador utilitza i del tipus que són (analògiques o digitals).

En la figura 27 es pot veure l'habilitació del pin 7 com entrada de l'ADC.

```
// get/set channel_ANC3 aliases
#define channel_ANC3_TRIS          TRISCbits.TRISC3
#define channel_ANC3_LAT          LATCbits.LATC3
#define channel_ANC3_PORT        PORTCbits.RC3
#define channel_ANC3_WPU         WPUCbits.WPUC3
#define channel_ANC3_OD          ODCONCbits.ODCC3
#define channel_ANC3_ANS         ANSELCbits.ANSC3
#define channel_ANC3_SetHigh()    do { LATCbits.LATC3 = 1; } while(0)
#define channel_ANC3_SetLow()    do { LATCbits.LATC3 = 0; } while(0)
#define channel_ANC3_Toggle()    do { LATCbits.LATC3 = ~LATCbits.LATC3; } while(0)
#define channel_ANC3_GetValue()  PORTCbits.RC3
#define channel_ANC3_SetDigitalInput() do { TRISCbits.TRISC3 = 1; } while(0)
#define channel_ANC3_SetDigitalOutput() do { TRISCbits.TRISC3 = 0; } while(0)
#define channel_ANC3_SetPullup() do { WPUCbits.WPUC3 = 1; } while(0)
#define channel_ANC3_ResetPullup() do { WPUCbits.WPUC3 = 0; } while(0)
#define channel_ANC3_SetPushPull() do { ODCONCbits.ODCC3 = 0; } while(0)
#define channel_ANC3_SetOpenDrain() do { ODCONCbits.ODCC3 = 1; } while(0)
#define channel_ANC3_SetAnalogMode() do { ANSELCbits.ANSC3 = 1; } while(0)
#define channel_ANC3_SetDigitalMode() do { ANSELCbits.ANSC3 = 0; } while(0)
```

Figura 27. Definició ADC

Destacar la línia “channel\_ANC3\_SetAnalogMode() ANSELCbits.ANSC3 = 1” el valor 1 indica que és una entrada analògica.

També trobem en aquest mateix fitxer les definicions de l'UART (figura 28). En el pins 5, RC5, la transmissió i en el pin 6, RC4, la recepció.

```
// get/set RC4 procedures
#define RC4_SetHigh()            do { LATCbits.LATC4 = 1; } while(0)
#define RC4_SetLow()            do { LATCbits.LATC4 = 0; } while(0)
#define RC4_Toggle()            do { LATCbits.LATC4 = ~LATCbits.LATC4; } while(0)
#define RC4_GetValue()          PORTCbits.RC4
#define RC4_SetDigitalInput()   do { TRISCbits.TRISC4 = 1; } while(0)
#define RC4_SetDigitalOutput() do { TRISCbits.TRISC4 = 0; } while(0)
#define RC4_SetPullup()        do { WPUCbits.WPUC4 = 1; } while(0)
#define RC4_ResetPullup()      do { WPUCbits.WPUC4 = 0; } while(0)
#define RC4_SetAnalogMode()    do { ANSELCbits.ANSC4 = 1; } while(0)
#define RC4_SetDigitalMode()   do { ANSELCbits.ANSC4 = 0; } while(0)

// get/set RC5 procedures
#define RC5_SetHigh()            do { LATCbits.LATC5 = 1; } while(0)
#define RC5_SetLow()            do { LATCbits.LATC5 = 0; } while(0)
#define RC5_Toggle()            do { LATCbits.LATC5 = ~LATCbits.LATC5; } while(0)
#define RC5_GetValue()          PORTCbits.RC5
#define RC5_SetDigitalInput()   do { TRISCbits.TRISC5 = 1; } while(0)
#define RC5_SetDigitalOutput() do { TRISCbits.TRISC5 = 0; } while(0)
#define RC5_SetPullup()        do { WPUCbits.WPUC5 = 1; } while(0)
#define RC5_ResetPullup()      do { WPUCbits.WPUC5 = 0; } while(0)
#define RC5_SetAnalogMode()    do { ANSELCbits.ANSC5 = 1; } while(0)
#define RC5_SetDigitalMode()   do { ANSELCbits.ANSC5 = 0; } while(0)
```

Figura 28. Definició UART

Destacar que defineix l'entrada i la sortida com analògica

```
"RC4_SetAnalogMode() ANSELCbits.ANSC4 = 1"
```

```
"RC5_SetAnalogMode() ANSELCbits.ANSC5 = 1"
```

#### 7.4 Adc

En els fitxers ADC, un cop definits els pins en els arxius pin\_manager, aquests defineixen les adreces de memòria i les funcions per tal d'analitzar si el canal està preparat, que comenci la conversió i que la guardi quan finalitzi. També és molt important que aquí es defineix la tensió de referència que s'utilitza per la conversió, en aquest cas és la d'alimentació.

```
void ADC_Initialize(void)
{
    // set the ADC to the options selected in the User Interface

    ADCON0 = 0x01;    // ADGO stop; ADON enabled; CHS ANA0;

    ADCON1 = 0xC0;    // ADFM right; ADNREF VSS; ADPREF VDD; ADCS FOSC/4;

    ADOCFR = 0x00;    // ADOCFR no_auto_trigger;

    ADRESL = 0x00;    // ADRESL 0;

    ADRESH = 0x00;    // ADRESH 0;
```

Figura 29. Inicialització de l'ADC

A la figura 29 es veu a quin canal està connectada (ADCON0) la tensió de referència (ADCON1) i les adreces de memòria activades per guardar la conversió (ADRESH, ADRESL).

```
void ADC_SelectChannel(adc_channel_t channel)
{
    ADCON0bits.CHS = channel;    // select the A/D channel

    ADCON0bits.ADON = 1;        // Turn on the ADC module
}

void ADC_StartConversion(void)
{
    ADCON0bits.ADGO = 1;        // Start the conversion
}

bool ADC_IsConversionDone(void)
{
    return ((bool) (!ADCON0bits.ADGO));    // Start the conversion
}

adc_result_t ADC_GetConversionResult(void)
{
    return ((adc_result_t) ((ADRESH << 8) + ADRESL));    // Conversion finished, return the result
}
```

Figura 30. Selecció de canal i conversió ADC

A la figura 30 es veuen les funcions que trien el canal ADC, en aquest cas només n'hi ha un, l'inici i seguiment de la conversió i com la guarda a memòria fent un desplaçament cap a la memòria superior per tal de tenir els 10 bits de definició.

## 7.5 Eusart

El microcontrolador porta habilitada una entrada com a recepció de l'EUSART en el pin 5 i es pot escollir el pin de transmissió entre els disponibles, s'escull el pin 6. En aquest arxiu s'indica que la comunicació és de 8 bits asíncrona i la velocitat 9600 bauds, que és la velocitat per defecte però es podria variar.

També s'hi troben les funcions per saber si el canal està disponible, per iniciar i controlar l'estat de la transmissió, control d'errors i interrupcions.

```

void (*EUSART_FramingErrorHandler)(void);
void (*EUSART_OverrunErrorHandler)(void);
void (*EUSART_ErrorHandler)(void);

void EUSART_DefaultFramingErrorHandler(void);
void EUSART_DefaultOverrunErrorHandler(void);
void EUSART_DefaultErrorHandler(void);

void EUSART_Initialize(void)
{
    BAUD1CON = 0x08; // ABD0VF no_overflow; SCKP Non-Inverted; BRG16 16bit_generator; WUE disabled;
    RC1STA = 0x90; // SPEN enabled; RX9 8-bit; CREN enabled; ADDEN disabled; SREN disabled;
    TX1STA = 0x24; // TX9 8-bit; TX9D 0; SENDB sync_break_complete; TXEN enabled; SYNC asynchronous;
    SPIBRGL = 0x67; // SPIBRGL 103;
    SPIBRGH = 0x00; // SPIBRGH 0;

    EUSART_SetFramingErrorHandler(EUSART_DefaultFramingErrorHandler);
    EUSART_SetOverrunErrorHandler(EUSART_DefaultOverrunErrorHandler);
    EUSART_SetErrorHandler(EUSART_DefaultErrorHandler);

    eusartRxLastError.status = 0;
}

bool EUSART_is_tx_ready(void)
{
    return (bool)(PIR1bits.TXIF && TX1STAbits.TXEN);
}

bool EUSART_is_rx_ready(void)
{
    return (bool)(PIR1bits.RCIF);
}

bool EUSART_is_tx_done(void)
{
    return TX1STAbits.TRMT;
}

eusart_status_t EUSART_get_last_status(void)
{
    return eusartRxLastError;
}

uint8_t EUSART_Read(void)
{
    while(!PIR1bits.RCIF)
    {
        CLRWDI();
    }
    eusartRxLastError.status = 0;
    if(1 == RC1STAbits.OERR) // EUSART error - restart
    {
        RC1STAbits.CREN = 0;
        RC1STAbits.CREN = 1;
    }
    return RC1REG;
}

```

```

void EUSART_Write(uint8_t txData)
{
    while(0 == PIR1bits.TXIF)
    {
        CLRWDI();
    }
    TX1REG = txData;    // Write the data byte to the USART.
}

char getch(void)
{
    return EUSART_Read();
}

void putch(char txData)
{
    EUSART_Write(txData);
}

void EUSART_DefaultFramingErrorHandler(void) {}
void EUSART_DefaultOverrunErrorHandler(void) { // EUSART error - restart
    RC1STAbits.CREN = 0;
    RC1STAbits.CREN = 1;
}
void EUSART_DefaultErrorHandler(void) {
}

```

Figura 31. Control de EUSART

## 7.6 Mcc

En aquest fitxer s'hi troben la inicialització de tot el sistema (Figura 34) principalment les definicions de l'oscil·lador intern i les divisions per adequar-se a les freqüències de treball, per exemple en la conversió analògica digital on per la conversió en els 10 bits necessitem 11.5 TAD (1 TAD és el temps de conversió d'un bit) on si s'utilitzés la velocitat normal de treball no acabaria de fer mai la conversió provocant que no hi hagués sortida. La divisió dels períodes de rellotge dona més temps a aquesta conversió.

El Mplab ho calcula automàticament però el datasheet del microcontrolador ens dona la fórmula per trobar la solució (Figura 32) i una taula amb recomanacions per l'aprofitament del rellotge (Figura 33)

*Assumptions: Temperature = 50°C and external impedance of 10kΩ 5.0V VDD*

$$\begin{aligned}
 TACQ &= \text{Amplifier Settling Time} + \text{Hold Capacitor Charging Time} + \text{Temperature Coefficient} \\
 &= TAMP + TC + TCOFF \\
 &= 2\mu s + TC + [(Temperature - 25^\circ C)(0.05\mu s/^\circ C)]
 \end{aligned}$$

*The value for TC can be approximated with the following equations:*

$$V_{APPLIED} \left( 1 - \frac{1}{(2^{n+1}) - 1} \right) = V_{CHOLD} \quad ;[1] \text{ } V_{CHOLD} \text{ charged to within } 1/2 \text{ lsb}$$

$$V_{APPLIED} \left( 1 - e^{-\frac{TC}{RC}} \right) = V_{CHOLD} \quad ;[2] \text{ } V_{CHOLD} \text{ charge response to } V_{APPLIED}$$

$$V_{APPLIED} \left( 1 - e^{-\frac{TC}{RC}} \right) = V_{APPLIED} \left( 1 - \frac{1}{(2^{n+1}) - 1} \right) \quad ;\text{combining [1] and [2]}$$

*Note: Where n = number of bits of the ADC.*

*Solving for TC:*

$$\begin{aligned}
 TC &= -CHOLD(RIC + RSS + RS) \ln(1/2047) \\
 &= -10pF(1k\Omega + 7k\Omega + 10k\Omega) \ln(0.0004885) \\
 &= 1.37\mu s
 \end{aligned}$$

*Therefore:*

$$\begin{aligned}
 TACQ &= 2\mu s + 892ns + [(50^\circ C - 25^\circ C)(0.05\mu s/^\circ C)] \\
 &= 4.62\mu s
 \end{aligned}$$

Figura 32. Càlcul període conversió ADC

| ADC Clock Period (TAD) |           | Device Frequency (Fosc)     |                             |                             |                             |                             |                             |
|------------------------|-----------|-----------------------------|-----------------------------|-----------------------------|-----------------------------|-----------------------------|-----------------------------|
| ADC Clock Source       | ADCS<2:0> | 32 MHz                      | 20 MHz                      | 16 MHz                      | 8 MHz                       | 4 MHz                       | 1 MHz                       |
| Fosc/2                 | 000       | 62.5ns <sup>(2)</sup>       | 100 ns <sup>(2)</sup>       | 125 ns <sup>(2)</sup>       | 250 ns <sup>(2)</sup>       | 500 ns <sup>(2)</sup>       | 2.0 µs                      |
| Fosc/4                 | 100       | 125 ns <sup>(2)</sup>       | 200 ns <sup>(2)</sup>       | 250 ns <sup>(2)</sup>       | 500 ns <sup>(2)</sup>       | 1.0 µs                      | 4.0 µs                      |
| Fosc/8                 | 001       | 0.5 µs <sup>(2)</sup>       | 400 ns <sup>(2)</sup>       | 0.5 µs <sup>(2)</sup>       | 1.0 µs                      | 2.0 µs                      | 8.0 µs <sup>(3)</sup>       |
| Fosc/16                | 101       | 800 ns                      | 800 ns                      | 1.0 µs                      | 2.0 µs                      | 4.0 µs                      | 16.0 µs <sup>(3)</sup>      |
| Fosc/32                | 010       | 1.0 µs                      | 1.6 µs                      | 2.0 µs                      | 4.0 µs                      | 8.0 µs <sup>(3)</sup>       | 32.0 µs <sup>(2)</sup>      |
| Fosc/64                | 110       | 2.0 µs                      | 3.2 µs                      | 4.0 µs                      | 8.0 µs <sup>(3)</sup>       | 16.0 µs <sup>(2)</sup>      | 64.0 µs <sup>(2)</sup>      |
| ADCRC                  | x11       | 1.0-6.0 µs <sup>(1,4)</sup> | 1.0-6.0 µs <sup>(1,4)</sup> | 1.0-6.0 µs <sup>(1,4)</sup> | 1.0-6.0 µs <sup>(1,4)</sup> | 1.0-6.0 µs <sup>(1,4)</sup> | 1.0-6.0 µs <sup>(1,4)</sup> |

Figura 33. Freqüències de treball de micro

```

void SYSTEM_Initialize(void)
{
    PMD_Initialize();
    PIN_MANAGER_Initialize();
    OSCILLATOR_Initialize();
    WDT_Initialize();
    ADC_Initialize();
    EUSART_Initialize();
}

void OSCILLATOR_Initialize(void)
{
    // NOSC HFINTOSC; NDIV 1;
    OSCCON1 = 0x60;
    // CSW HOLD may proceed; SOSCFWR Low power; SOSCBE crystal oscillator;
    OSCCON3 = 0x00;
    // LFOEN disabled; ADOEN disabled; SOSCEEN disabled; EXTOEN disabled; HFOEN disabled;
    OSCEN = 0x00;
    // HFFRQ 4_MHz;
    OSCFRQ = 0x03;
    // HFTUN 0;
    OSTUNE = 0x00;
}

void WDT_Initialize(void)
{
    // WDTPS 1:65536; SWDTEN OFF;
    WDTCON = 0x16;
}

void PMD_Initialize(void)
{
    // CLKRMD CLKR enabled; SYSCMD SYSCLK enabled; FVRMD FVR enabled; IOCMD IOC enabled; NVM
    PMD0 = 0x00;
    // TMR0MD TMR0 enabled; TMR1MD TMR1 enabled; TMR4MD TMR4 enabled; TMR5MD TMR5 enabled; T
    PMD1 = 0x00;
    // DACMD DAC enabled; CMP1MD CMP1 enabled; ADCMD ADC enabled; CMP2MD CMP2 enabled;
    PMD2 = 0x00;
    // CCP2MD CCP2 enabled; CCP1MD CCP1 enabled; CCP4MD CCP4 enabled; CCP3MD CCP3 enabled; I
    PMD3 = 0x00;
    // MSSP1MD MSSP1 enabled; UART1MD EUSART enabled; MSSP2MD MSSP2 enabled;
    PMD4 = 0x00;
    // DSMMD DSM enabled; CLC3MD CLC3 enabled; CLC4MD CLC4 enabled; CLC1MD CLC1 enabled; CLC
    PMD5 = 0x00;
}

```

Figura 34. Codi mcc



## 7.7 Main.c

S'arriba així al fitxer main.c, que és on un cop definit i iniciat tot el sistema es fan tots els càlculs i definicions necessàries per obtenir el propòsit desitjat.

Així, en aquest fitxer s'executa la fórmula (5) que permet trobar la temperatura que enviarà al mòdul Bluetooth (codi de la figura 35).

$$temp = \left( \frac{Sensor * (\Delta T)}{Q} \right) + T_i \quad (5)$$

On:

- Sensor és el voltatge llegit a l'entrada del canal ADC.
- $\Delta T$  és el rang de mesura de temperatura (120 °C – 20 °C)
- Q és el nivell de quantificació ( $2^{10} - 1$ )
- $T_i$  és la temperatura on comença la mesura (20 °C)

```
float sensor = ADC_GetConversion(channel_ANC3); //llegeix el valor de l'ADC
float temp = ((sensor*100)/1023)+20;           /* transforma els volts en temperatura
                                              *1023 és el nivell de quantificació amb 10 bits = (2^10)-1
                                              * 100 és el rang de mesura 120-20
                                              * suma 20 perquè és l'inici de mesura*/
putch (temp);                                 //escriu a la sortida cap el Bluetooth
__delay_ms (5000);                           // Espera 5 s abans de fer una nova lectura
```

Figura 35. Conversió voltatge ADC en temperatura

Un cop feta la operació l'envia al pin de transmissió de l'UART i espera 5 segons per fer una nova lectura.

En aquest arxiu també hi ha les comprovacions del mòdul Bluetooth (Bth\_Ready figura 36) per saber si esta preparat.

```
bool Bth_Ready(void) { // Comproba que el mòdul Bluetooth està preparat
    uint8_t Tx = (uint8_t)atoi("AT"); // Envia codi de comprovació de canal
    EUSART_Write(Tx);
    if(PIR1bits.RCIF==1) { // Comproba que obté resposta
        return getch();
    }
    else{
        Bth_Initialize(); // Si no hi ha resposta configura en mòdul
        return getch();
    }
}
```

Figura 36. Comprovació mòdul Bluetooth

Per saber si està preparat s'envia senyal de petició (AT) i si obté resposta es dona per preparat i si no l'inicia mitjançant la funció Bth\_Initialize (Figura 37). En aquesta es fa primer una trucada (ATA) i si contesta s'envia la configuració, ja que si s'havia perdut la connexió probablement haurà reiniciat i tindrà la configuració predeterminada.

```

void Bth_Initialize(void){
    uint8_t Tx;
    Tx = (uint8_t)atoi("ATA");           //Envia senyal de trucada
    __delay_ms(100);                       // Temps d'espera de resposta
    if (PIR1bits.RCIF==1){                 // Si obté resposta
        EUSART_Write(Tx);
        uint8_t Tx = (uint8_t)atoi("AT+BTGY"); //Configura mode esclau
        EUSART_Write(Tx);
        Tx = (uint8_t)atoi("AT+BTK=<>");    // Sense pin d'identificació
        EUSART_Write(Tx);
        Tx = (uint8_t)atoi("AT+BTN=<Sensor>"); // Que el veiem amb el nom de Sensor
        EUSART_Write(Tx);
    }
    else{
        __delay_ms(100);                    // Si falla la comunicació espera 100 ms i torna a començar el cicle
    }
    return;
}

```

Figura 37. Inici del mòdul Bluetooth

De la configuració predeterminada ja interessa que la comunicació sèrie sigui a 9600 bauds, 8 bits i asíncrona però ens interessa que sigui esclau, per tal de que no busqui connexions i sempre sigui visible, així se li envia la seqüència “AT+BTGY”. Per facilitar la connexió es treu la contrasenya, enviant “AT+BTK=<>” i que s’identifiqui amb el nom de sensor mitjançant “AT+BTN=<Sensor>”

En cas de no rebre resposta al senyal de trucada, espera 100 ms i torna a començar el cicle.

## 7.8 Simulació

L’Mplab també permet fer una simulació del funcionament del programa, tant en la compilació sense errors (figura 38) com mitjançant la seva aplicació stimulus que permet forçar entrades de dades, en aquest cas asíncrones i de valor de voltatge per simular el que li arribaria a l’entrada del pin de l’ADC i veure com les transforma a digitals i les guarda a les memòries ADRESH i ADRESL.

```

CLEAN SUCCESSFUL (total time: 92ms)
make -f nbproject/Makefile-default.mk SUBPROJECTS= .build-conf
make[1]: Entering directory 'C:/Users/Joan/Documents/UOC/UOC/TFG/PACS/PAC3/Micro/TFG_PAC3.X'
make -f nbproject/Makefile-default.mk dist/default/production/TFG_PAC3.X.production.hex
make[2]: Entering directory 'C:/Users/Joan/Documents/UOC/UOC/TFG/PACS/PAC3/Micro/TFG_PAC3.X'
"C:\Program Files (x86)\Microchip\xc8\v2.10\bin\xc8-cc.exe" -mcpu=16F18325 -c -fno-short-double -fno-short-float -O0 -fasmfile -max
"C:\Program Files (x86)\Microchip\xc8\v2.10\bin\xc8-cc.exe" -mcpu=16F18325 -c -fno-short-double -fno-short-float -O0 -fasmfile -max
"C:\Program Files (x86)\Microchip\xc8\v2.10\bin\xc8-cc.exe" -mcpu=16F18325 -c -fno-short-double -fno-short-float -O0 -fasmfile -max
"C:\Program Files (x86)\Microchip\xc8\v2.10\bin\xc8-cc.exe" -mcpu=16F18325 -c -fno-short-double -fno-short-float -O0 -fasmfile -max
"C:\Program Files (x86)\Microchip\xc8\v2.10\bin\xc8-cc.exe" -mcpu=16F18325 -c -fno-short-double -fno-short-float -O0 -fasmfile -max
"C:\Program Files (x86)\Microchip\xc8\v2.10\bin\xc8-cc.exe" -mcpu=16F18325 -c -fno-short-double -fno-short-float -O0 -fasmfile -max
"C:\Program Files (x86)\Microchip\xc8\v2.10\bin\xc8-cc.exe" -mcpu=16F18325 -WL,-Map=dist/default/production/TFG_PAC3.X.production.map

Memory Summary:
Program space      used  ADFh ( 2783) of 2000h words ( 34.0%)
Data space         used  68h ( 104) of 400h bytes ( 10.2%)
EEPROM space       used   0h (  0) of 100h bytes (  0.0%)
Data stack space   used   0h (  0) of 398h bytes (  0.0%)
Configuration bits used   4h (  4) of  4h words (100.0%)
ID Location space  used   0h (  0) of  4h bytes (  0.0%)

make[2]: Leaving directory 'C:/Users/Joan/Documents/UOC/UOC/TFG/PACS/PAC3/Micro/TFG_PAC3.X'
make[1]: Leaving directory 'C:/Users/Joan/Documents/UOC/UOC/TFG/PACS/PAC3/Micro/TFG_PAC3.X'

BUILD SUCCESSFUL (total time: 5s)
Loading code from C:/Users/Joan/Documents/UOC/UOC/TFG/PACS/PAC3/Micro/TFG_PAC3.X/dist/default/production/TFG_PAC3.X.production.hex...
Program loaded with pack, PIC16F18325_DFP, 1.3.117, Microchip
Loading completed
|

```

Figura 38. Compilació

En activar l’aplicació stimulus del Mplab podem forçar les entrades que ens interessin i per veure com reacciona el microcontrolador activem la finestra watches on podem observar les

dues adreces de memòria (ADRESH i ADRESL) com varien. A la figura 39 es pot veure com treballa l'ADC en entrar-li els valors del voltatge corresponents a 45 °C (991 mV).

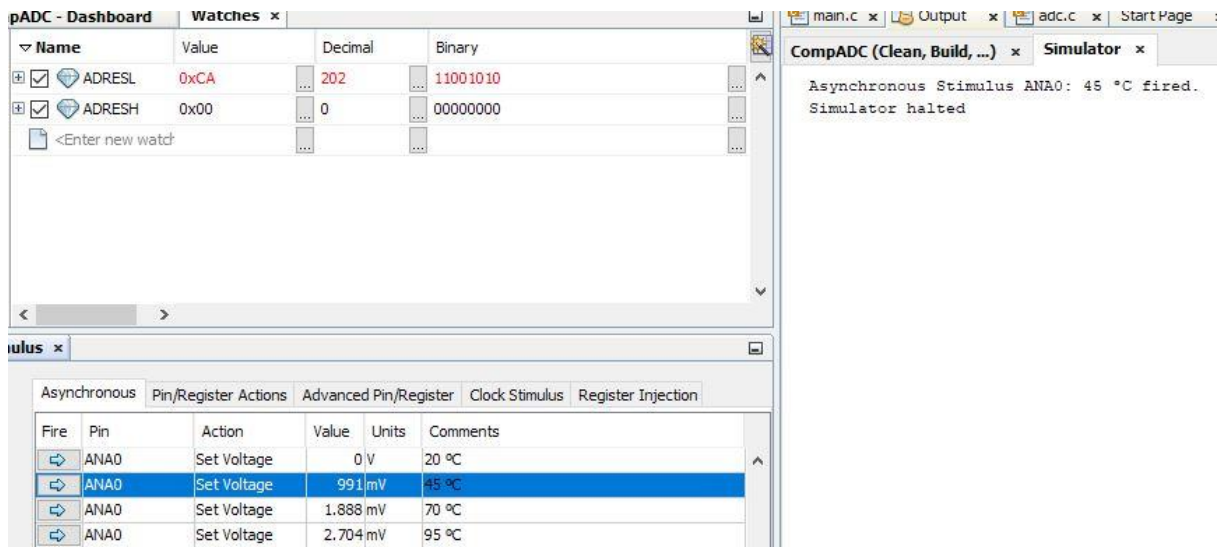


Figura 39. Entrada de 45 °C

I a la figura 40 l'entrada màxima de 120 °C (3.447 V).

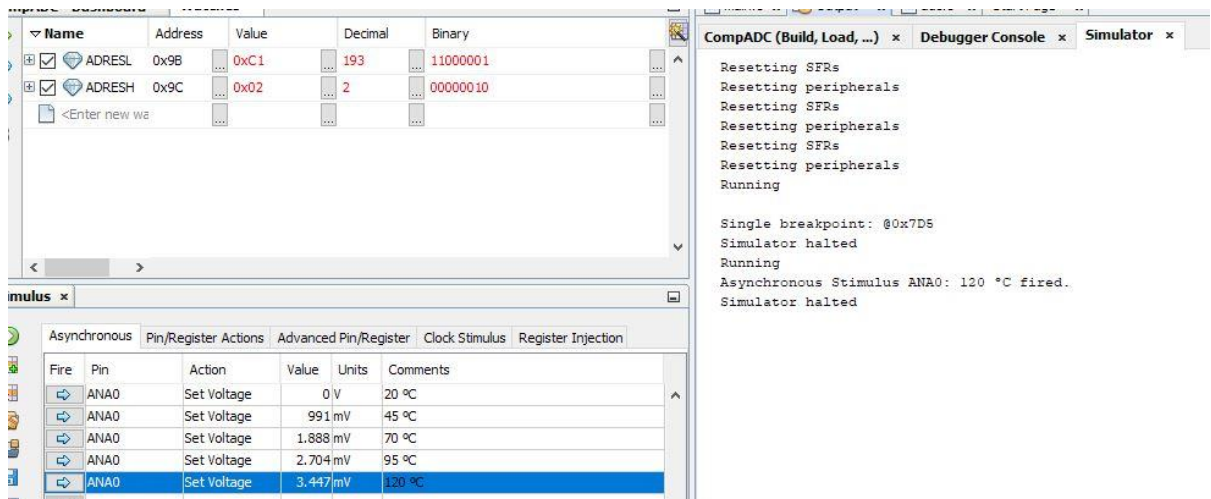


Figura 40. Entrada de 120 °C

A la figura 39 es pot veure que el valor de la memòria alta (ADRESH) queda a zero mentre que a la figura 40 veiem que té el valor màxim 2 (10 en binari).

## 8. Pressupost

| Quantitat | Component                           | Preu unitat | Preu total |
|-----------|-------------------------------------|-------------|------------|
| 1         | Mòdul Bluetooth BT-730-SA           | 21.29       | 21.29      |
| 2         | Condensadors 220 $\mu$ F            | 0.70        | 1.40       |
| 1         | Condensador 1 $\mu$ F               | 0.01        | 0.01       |
| 1         | Condensador 100 nF                  | 0.07        | 0.07       |
| 2         | Díodes 1N4007                       | 0.05        | 0.10       |
| 1         | Amplificador operacional LM358      | 0.30        | 0.30       |
| 1         | Amplificador operacional $\mu$ A741 | 0.45        | 0.45       |
| 4         | Amplificador operacional OPA333     | 2.56        | 10.24      |
| 21        | Resistències                        | 0.03        | 0.63       |
| 2         | Transistors 2N                      | 0.75        | 1.50       |
| 2         | Transistors TIP                     | 0.50        | 1.00       |
| 2         | Díodes Zener 1N3823                 | 7.79        | 15.58      |
| 1         | Díode Zener 1N4106                  | 3.28        | 3.28       |
| 1         | Microcontrolador PIC 16F18325       | 0.88        | 0.88       |
| 1         | Sòcol 14 pins                       | 0.12        | 0.12       |
| 1         | Placa PCB i soldadura               | 55.20       | 55.20      |
| 1         | Caixa paret Y716                    | 4.38        | 4.38       |
| 1         | Sensor Pt1000                       | 24.02       | 24.02      |
| 24        | Preparació del projecte             | 40.00       | 960.00     |
| 33        | Cerca d'informació                  | 40.00       | 1320.00    |
| 35        | Disseny                             | 40.00       | 1400.00    |
|           | TOTAL                               |             | 3820.45    |

Taula 8. Pressupost

Els components s'han tret de diferents subministradors, per exemple el microcontrolador i el mòdul Bluetooth del web de Mouser electronics<sup>2</sup>, la resta de components electrònics de Todoelectronica<sup>3</sup> i el sensor d'spluss<sup>4</sup>. La placa PCB i el muntatge dels components s'encarregaria a l'empresa PCBWay<sup>5</sup>.

El preu pot semblar exagerat però es tracta de la construcció d'una sola unitat, si el producte entres en cadena de muntatge els costos del disseny es podrien dividir per la quantitat en producció. Per exemple, si l'amortització fos en 100 unitats el cost de l'estudi seria de 36.80 €/unitat fent que el preu total baixi fins a 154.23 €/unitat.



Figura 41. Caixa exterior

La presentació final seria la de la figura 41, però també hi hauria la possibilitat de muntar l'antena exterior amb el Bluetooth BT-730-SC que és idèntic en preu i característiques al de l'estudi, per la diferència del preu de l'antena (figura 42), els connectors (figura 43) i el cable de connexió que puja a 6.81 €. Es pot veure l'acabat final en aquesta configuració a la figura 44.



Figura 42. Antena



Figura 43. Connector



Figura 44. Caixa amb antena

Aquest acabat seria necessari si entre el mòdul de Bluetooth i el receptor hi haguessin obstacles que produïssin molta atenuació del senyal.

## 9. Conclusions

En aquest treball s'han portat a terme un seguit d'accions que impliquen l'aplicació dels programes de diferents assignatures dels estudis de grau de tecnologies de la telecomunicació.

S'han desenvolupat les diferents fases d'un projecte des del plantejament fins el fi, comprovant que la divisió indicada en l'enunciat en diferents tasques ajuda molt a l'hora de planificar-lo i permet un millor seguiment del treball.

La planificació temporal no sempre coincideix el temps proposat amb el realment utilitzat però com que unes vegades és més temps del programat i altres menys s'ha anat compensant sense pràcticament diferències en el temps global.

Tot i tenir una sèrie de riscos planificats amb les seves solucions no s'han hagut de tenir en compte ja que tot ha funcionat correctament aportant tranquil·litat i millor gestió del temps a l'hora de treballar. En general, el temps marcat a l'inici ha estat suficient. També s'han desenvolupat mecanismes procedents de diferents fonts pels càlculs i programació del circuit electrònic.

En la pràctica hi ha hagut dificultats a l'hora d'utilitzar les eines informàtiques, pel muntatge dels circuits electrònics, com el TINA-TI o EAGLE, ja que no tenen en les seves llibreries tots els components utilitzats i s'han de fer importacions o noves creacions que s'aparten de la intuïtivitat que tenen aquests programes i obliguen a fer cerques o lectures del manual més acurades.

Mplab, de Microchip, és el programa utilitzat per la programació del microcontrolador i disposa d'eines molt intuïtives tant per la programació de la configuració (MCC) com per realitzar proves i compte amb l'avantatge de poder programar en llenguatge C, molt més intuïtiu i conegut que l'assemblador que realment utilitza el micro.

## 10. Glossari

**ADC**→ Analog to Digital Converter.

**BLE** → Bluetooth Low Energy.

**Bluetooth SIG** → Bluetooth Special Interest Group.

**CISC**→ Complex Instruction Set Computer.

**CPU**→ Central Processing Unit.

**EEPROM**→ Electrically-Erasable Programmable Read-Only Memory.

**EPROM**→ Erasable Programmable Read Only Memory.

**ERD** → Enhanced Data Rate.

**EUSART**→ Enhanced Universal Asynchronous Receiver Transceiver.

**FRAM**→ Ferroelectric Random Access Memory.

**GFSK** → Gaussian Frequency Shift Keying.

**GPIO** → General Purpose Input Output.

**HS** → High Speed.

**IoT** → Internet of Things.

**ISM** → Industrial, Scientific, Medical.

**MCC**→ MPLAB Code Configurator.

**NTC** → Negative Temperature Coefficient.

**PAN** → Personal Area Network.

**PTC** → Positive Temperature Coefficient

**PSK** → Phase Shift Keying.

**RISC**→ Reduced Instruction Set Computer.

**ROM**→ Read Only Memory.

**RTD** → Resistance Temperature Detectors.

**SSFH** → Spread Spectrum Frequency Hopping.

**SRAM**→ Static Random Access Memory.

**UART**→ Universal Asynchronous Receiver Transmitter.

**WIFI** → Wireless Fidelity.

**WIMAX** → Worldwide Interoperability for Microwave Acces.

## 11. Bibliografia

### 11.1 Referències

1. Classeamida (2018). Fuente simètrica o dual [en línia]. Google LLC Youtube. [Data de la consulta: abril de 2020]. <https://www.youtube.com/watch?v=3CFaW9FpRUM>
2. Mouser Electronics, Inc (2020). [Data de la consulta: maig 2020]. <https://www.mouser.es/>
3. TodoElectronica (2020). [Data de la consulta: maig 2020]. <https://www.todoelectronica.com/es/>
4. S+S Regeltechnik GmbH (2020). [Data de la consulta: maig 2020]. <https://spluss.de/es/>
5. PCBWay (2019). ). [Data de la consulta: maig 2020]. <https://www.pcbway.es/>

### 11.2 Datasheets

1. Microcontrolador 16F18325 <https://www.microchip.com/wwwproducts/en/PIC16F18325>
2. LM358 <https://www.ti.com/product/LM358>
3.  $\mu$ A741 <https://www.ti.com/product/UA741>
4. OPA333 <https://www.ti.com/product/OPA333>
5. Mòdul Bluetooth: <https://www.lairdconnect.com/wireless-modules/bluetooth-modules/bluetooth-20-21-and-30-modules/bt730-series-bluetooth-module#documentation>
6. Sensor ALTF-1: <https://spluss.de/es/produkte-es/temperatura/temperatura-pasiva/sensores-por-contacto/1101-6020-5211-120-altf1-pt1000-silikon-1-5m/>

### 11.3 Consulta

1. Wikipedia (2020). Sensor [en línia]. Wikimedia Foundation, Inc. [data de la consulta: març de 2020]. <https://ca.wikipedia.org/wiki/Sensor>
2. Wikipedia (2020). Bluetooth [En línia]. Wikimedia Foundation, Inc. [data de la consulta: març de 2020]. <https://en.wikipedia.org/wiki/Bluetooth>
3. Sayothan Sinha (2016). *Circuito de Fuente de 12 V con 741 y Zener*. Electrónica Unicrom. [data de la consulta: abril de 2020]. <https://unicrom.com/fuente-de-12-v-con-741-y-zener/>
4. Wikipedia (2019). *Microcontrolador*. Wikimedia Foundation, Inc. [data de la consulta: abril de 2020]. <https://ca.wikipedia.org/wiki/Microcontrolador>
5. Electrónica Chile (2018). Tipos y aplicaciones de microcontroladores. Electrónica Chile. [data de la consulta: abril de 2020]. <https://www.electronicachile.com/microcontroladores/tipos-y-aplicaciones-de-microcontroladores/>
6. Mundo Altavoces (2018). Bluetooth clases y versiones, ¿En qué se diferencian? [En línia]. Mundo Altavoces. [data de la consulta: març de 2020]. <https://mundoaltavoces.com/bluetooth-todas-las-clases-y-versiones-en-que-se-diferencian/>
7. Tecnologia Informatica (2020). Bluetooth: Que es? Para que sirve? Como usar Bluetooth? Problemas de conexión.[En línia]. Tecnologia-informatica. [data de la consulta: març de 2020]. <https://www.tecnologia-informatica.com/bluetooth/>
8. Bluetooth (2020). Bluetooth.[En línia]. Bluetooth SIG. [data de la consulta: març de 2020]. <https://www.bluetooth.com/>
9. Doja (2019). Evolución del Bluetooth. [En línia]. Balinvest trade. [data de la consulta: març de 2020]. <https://www.doja.es/2019/02/evolucion-del-bluetooth/>



10. Enrique Gomez (2018). Cómo conectar una PT100 ¿Qué es una RTD? [en línea]. Rincón ingenieril. [data de la consulta: març de 2020]. <https://www.rinconingenieril.es/como-conectar-una-pt100/>