

Pau Añó Martínez

Disseny i confecció d'un metrònom digital enfocat a instruments de percussió basat en un Microcontrolador de baix cost

TREBALL DE FINAL DE GRAU

dirigit per Jose Luis Ramírez Falo

Grau d'Enginyeria Electrònica Industrial i Automàtica



UNIVERSITAT ROVIRA I VIRGILI

Tarragona

2021

Agraïments

A José Luis Ramirez Falo per la tutorització del Projecte,
al José Rodríguez Tena per l'ajuda a l'hora d'imprimir la capsa del prototip

i

a la família per donar-me suport en tot el procés.

0 Índex

0 Índex	4
1 Introducció	6
1.1 <i>Objectiu</i>	6
1.2 <i>Motivació</i>	6
2 “State of the art”	7
2.1 <i>Producte a desenvolupar</i>	7
2.2 <i>Productes existents</i>	7
2.3 <i>Funcionalitats a afegir</i>	8
2.4 <i>Funcionalitats del producte</i>	9
3 Mètodes i càlculs	10
3.1 <i>Prototip</i>	10
3.1.1 <i>Producte final</i>	10
3.1.2 <i>Transductor</i>	12
3.1.3 <i>Alimentació del prototip i avaluació consums</i>	12
3.2 <i>Microcontrolador</i>	13
3.2.1 <i>Consideracions en la tria</i>	13
3.2.2 <i>Tria del microcontroladors</i>	14
3.2.3 <i>Placa de desenvolupament XPlained ATMEGA328P</i>	14
3.3 <i>Hardware addicional</i>	16
3.4 <i>PCB</i>	18
3.4.1 <i>Circuit electrònic</i>	18
3.4.2 <i>Traçat de pistes</i>	20
3.4.3 <i>Manufactura i soldadura de la PCB</i>	22
3.5 <i>Programació</i>	25
3.5.1 <i>Terminologia</i>	26
3.5.2 <i>Estructura del programa</i>	26
3.5.3 <i>Menús</i>	28
3.5.4 <i>Generació de la consigna</i>	30
3.5.5 <i>Ús EEPROM</i>	33
3.5.6 <i>Ús llibreries externes</i>	34
3.6 <i>Caixa prototip</i>	35
3.6.1 <i>Disseny</i>	35
3.6.2 <i>Impressió i resultat final</i>	38
3.7 <i>Cost prototip</i>	40
3.7.1 <i>Materials</i>	40
3.7.2 <i>Cost ma d’obra</i>	41
3.7.3 <i>Cost d’enginyeria</i>	41
3.7.4 <i>Cost final</i>	42
3.8 <i>Estudi econòmic prototip</i>	42
4 Conclusions	45

5	Annexos	47
5.1	<i>Circuit elèctric</i>	47
5.2	<i>Codi font</i>	48
5.2.1	<i>Codi complet</i>	48
5.2.2	<i>Llibreria</i>	85
5.3	<i>Tria del microcontrolador</i>	87
5.3.1	<i>Fitxa tècnica: STM32L100DISCOVERY</i>	91
5.3.2	<i>Fitxa tècnica: ATTINY416 Xplained Nano.....</i>	93
5.3.3	<i>Fitxa tècnica: ATMEGA328P-XMINI</i>	95
5.4	<i>Fitxes tècniques components.....</i>	97
5.4.1	<i>TLC5628CN1</i>	97
5.5	<i>Avaluació consums.....</i>	100
5.5.1	<i>Botons.....</i>	100
5.5.2	<i>Buzzer.....</i>	100
5.5.3	<i>LED's</i>	102
5.5.4	<i>LCD</i>	103
5.5.5	<i>Integrats</i>	104
5.5.6	<i>Microcontrolador i Xplained</i>	104
5.5.7	<i>Còmput total</i>	104
5.6	<i>Costs</i>	105
5.6.1	<i>Càlcul cost 200 prototips</i>	105
5.6.2	<i>Càlcul cost SMT assembly.....</i>	107
5.7	<i>Manual d'usuari</i>	109
6	Referències	115
6.1	<i>Webgrafia.....</i>	115
6.2	<i>Webs de proveïdors.....</i>	116
6.3	<i>Referències imatges</i>	117

1 Introducció

El projecte consisteix en l'elaboració d'un metrònom digital enfocat l'ús en instruments de percussió. El mitjà pel qual es vol pretendre l'elaboració d'aquest, és mitjançant un sistema basat en un microcontrolador.

1.1 Objectiu

L'objectiu és el de crear un metrònom digital plenament funcional a nivell de software i hardware. Addicionalment es pretén elaborar un projecte complet i que el resultat final del projecte sigui un producte plenament funcional i “*plug and play*” per a l'usuari.

Finalment, es pretén que aquest sigui competitiu en el mercat, és a dir, que el cost sigui baix però incorpori el màxim de funcionalitats interessants per a l'usuari.

1.2 Motivació

Aquest projecte apareix de la fusió de dos interessos personals, un d'ells acadèmic i l'altre relacionat amb hobbies.

Fa uns deu anys vaig començar en la pràctica d'instruments de percussió, en concret la bateria. Amb l'aprenentatge de l'instrument he pogut veure moltes connexions entre la música i temes tècnics.

Al segon quadrimestre del tercer curs del grau vaig poder comprovar gràcies a la assignatura de microcontroladors el potencial que tenen aquests dispositius.

2 “State of the art”

2.1 Producte a desenvolupar

El producte a desenvolupar és un metrònom digital. Aquests són freqüentment usats per músics que vulguin perfeccionar la practica de l'instrument.

Si féssim una analogia tècnica, un metrònom es un element de sincronisme. Quan es vol realitzar la comunicació entre dos nodes, apareix el problema que en trames llargues per culpa de les toleràncies entre els rellotges del node receptor i emissor apareguin problemes de sincronisme. Per això s'usen mètodes per tal d'evitar aquest efecte, com codificacions que incloguin flancs de pujada o baixada que permetin re-calibrar els rellotges.

En un grup de música, el metrònom acaba sent l'element de sincronisme entre tots els integrants. En actuacions en directe és habitual que s'usi per tal de donar solidesa al so global d'una banda. També es habitual que el control d'aquests elements el tinguin els elements de percussió de un grup musical, donat que son els encarregats de crear la base rítmica d'una cançó.

2.2 Productes existents

Primerament s'observaran els productes que hi ha ara al mercat i quines son les seves funcionalitats. A partir d'aquestes funcionalitats es determinarà quines son les funcions que es volen implementar al prototip.



Figura 1. Tama RW200.

Amb un preu de 85 € el Tama RW200 es un dels metrònoms més venuts i més complets del mercat. Donat que Tama es un fabricant que en gran part fabrica instruments relacionats amb la percussió, ha enfocat el producte als músics d'aquest camp.

Taula 1 Funcionalitats del Tama RW200

Funcionalitat	Hardware associat	Descripció funcionalitat
TAP	<i>Pad</i> de goma a la part superior	Es clica la tecla TAP TEMPO i mitjançant el golpeig al <i>pad</i> es genera la consigna. El rang de treball de les consignes és de 30 a 250 BPM.
Modificar consigna	Rodeta situada a la part inferior dreta	Augmenta o disminueix la consigna en funció de cap a on es realitzi el gir.
Diversos modes	Botó mode	El metrònom té dos modes de funcionament: <i>Practice</i> : Mode per defecte. El metrònom funciona de forma lliure. Es poden modificar els sons que ofereix el metrònom i parar/encendre el metrònom. <i>Stage</i> : Enfocat a l'ús en directe. Permet la reproducció d'uns programes configurats prèviament.
Guardar programes	Botons <i>program</i> i botó <i>store</i>	Permet navegar entre els diversos programes guardats i realitzar canvis en aquests.
Executar programes	Botons <i>program</i> i capacitat d'utilitzar un pedal	Un cop en el mode <i>Stage</i> s'executen els programes de forma ordenada. Els programes s'inicien o bé per la pulsació del botó START/STOP o bé amb l'ús de un pedal extern
Modificar el so de sortida	Botons superiors al display	Mitjançant una sèrie de botons es permet modificar el so que ofereix el metrònom al funcionar. Es pot modificar la subdivisió i la presència o no d'un <i>beat</i> (cop) dominant.
Visualització de la informació	Display LCD	Mitjançant un display es pot veure tota la informació del metrònom. Permet la navegació clara entre menús.

2.3 Funcionalitats a afegir

A part d'algunes de les funcions comentades en l'apartat 2.2, es pretén que el metrònom inclogui una funció addicional.

La funció en qüestió es que el “*tap*” es pugi fer tant mitjançant els botons com picant un element en concret del instrument. Aquest fet permetrà una interacció home-màquina que encara no es veu en aquest tipus de dispositius.

Per tal d'implantar aquesta funció, s'instal·larà algun tipus d'element sensor a l'element al qual el músic vulgui assignar-li aquesta funció. L'element sensor ha de ser capaç de detectar quan l'element es glopejat i fer el mateix tractament que es faria en el cas que el “*tap*” es donés a través dels botons.

2.4 Funcionalitats del producte

A partir del comentat en els apartats 2.3 i 2.2, es determina que aquestes són les funcionalitats que ha de tindre el metrònom digital a desenvolupar.

1. Start/Stop del funcionament del metrònom.
2. Capacitat d'assignar consigna de forma manual.
3. Capacitat de veure el tempo al que esta actualment el metrònom.
4. Funció tap amb un element de l'instrument.
5. Funció tap amb botons.
6. Varis menús de cara a l'usuari.
7. Sortida a auriculars.
8. Capacitat de guardar una sèrie de programes amb consignes configurades prèviament.
9. Capacitat de reproduir els programes prèviament guardats.

3 Mètodes i càlculs

En aquest apartat es descriu el prototip realitzat. A continuació es comenten elements tant de software com de hardware que conformen el prototip.

3.1 Prototip

En aquest apartat es presenta el prototip final.

3.1.1 Producte final



Figura 2. Prototip final (vista superior)

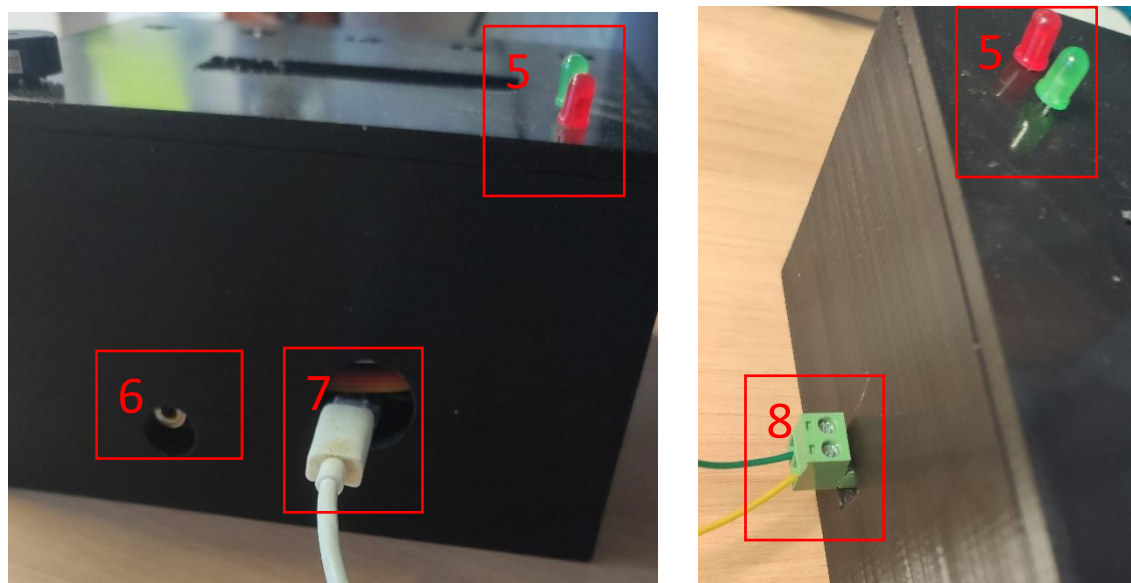


Figura 3. Esquerra: vista posterior del prototip. Dreta: vista lateral del prototip.

En la Figura 2 i en la Figura 3 es pot veure el prototip final. Els elements numerats són d'accés a l'usuari. En la Taula 2 es descriu cadascun dels elements enumerats.

Taula 2. Descripció dels elements enumerats a la Figura 2 i Figura 3

Numero	Descripció
0	Botó 0
1	Botó 1
2	Botó 2
3	Botó 3
4	Botó 4
5	LED per la visualització de la consigna
6	Sortida d'àudio jack 3.6 mm
7	Alimentació del prototip a través de un cable micro USB.
8	Bornes per la connexió per part del usuari del transductor
9	Display LCD

Pel que fa a la mida del prototip, mesura 150 mm d'amplada, 150 mm de longitud i 80 mm d'altura. El pes és de 476 g.

3.1.2 Transductor

El transductor que s'usa en conjunció amb el prototip és el *TE Connectivity* 1002794. És un transductor fabricat per *TE Connectivity*. Gràcies al material piezoelèctric que conté, ofereix tensió als pins de sortida quan està subjecte a vibracions.



Figura 4. Transductor TE 1002794.

El transductor va connectat al prototip mitjançant els connectors de la Figura 3. Es preveu que sigui l'usuari el que realitzi la instal·lació amb l'element terminal. Conjuntament amb el prototip es preveu que estigui inclòs un transductor i cable per tal que el usuari realitzi el posicionament. En la Figura 5 es pot veure un exemple de la instal·lació del transductor en el element terminal.

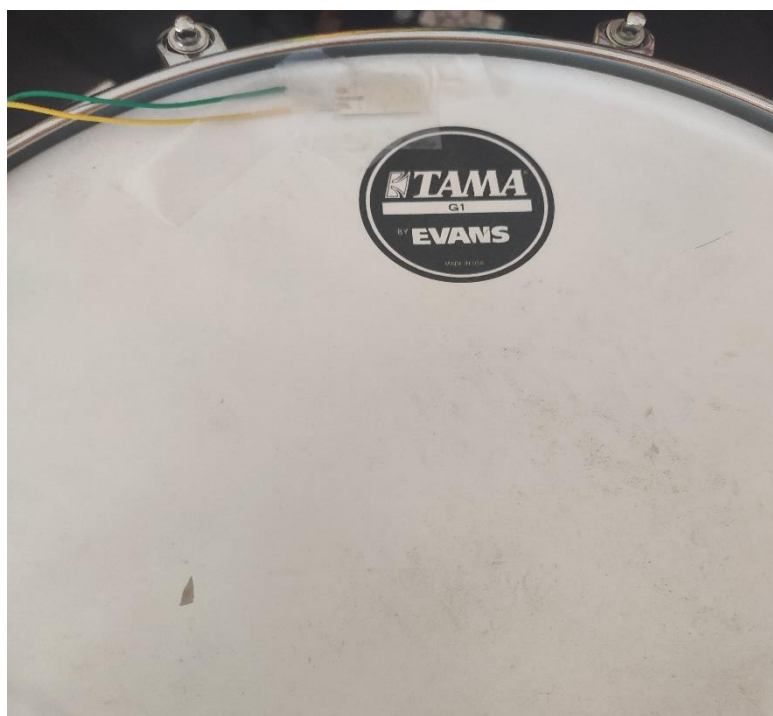


Figura 5. Exemple de instal·lació del transductor en el element terminal

3.1.3 Alimentació del prototip i avaluació consums

El prototip necessita una alimentació de 5 V de continua a través del connector USB present a la part posterior del prototip (Figura 3 element 7).

L'alimentació vindria donada a través d'un carregador de mòbil convencional, d'una *powerbank* o bé des d'un ordinador. Es deixa a l'usuari l'elecció del mètode que prefereixi.

És d'interès veure quins és el consum del prototip a través del anàlisi dels circuits d'aquest. En el annex 5.5 es desglossa el consum de cadascun dels elements i es realitza el còmput total de consum.

Es determina que el consum del prototip es:

Tensió:	5 V
Consum corrent:	41.57 mA
Potència:	207,85 mW

3.2 Microcontrolador

El prototip esta basat en un sistema amb un microcontrolador. En aquest apartat es comenten els criteris generals per a la tria del microcontrolador i s'introdueix la plataforma de desenvolupament que s'ha utilitzat per a dur a terme el projecte.

3.2.1 Consideracions en la tria

Els microcontroladors tenen una mida molt reduïda i sovint es presenten en encapsulats que s'han de soldar mitjançant SMT. S'ha considerat buscar un kit de desenvolupament en el qual s'inclogui el microcontrolador, els mecanismes necessaris per alimentar-lo via USB i els elements auxiliars que aquest necessita.

S'ha valorat positivament que el kit de desenvolupament estengui els pins del microcontrolador a una mida a la qual es puguin soldar pins. També s'ha valorat que el kit de desenvolupament disposi de una sortida que pugui alimentar correctament tot el hardware que es decideixi implementar a una tensió de 5V.

Per la tria del microcontrolador s'han tingut en compte els següents paràmetres: (l'ordre no implica més o menys prioritat)

- Capacitat de poder-se programar mitjançant C i ASM.
- Convertidor Analògic-Digital.
- Sortida PWM
- Els pins anteriorment esmentats
- Cost inferior a 10€
- Que disposi d'un IDE funcional i gratuït amb el suport del fabricant.

- Suficient documentació per a realitzar consulta: “*Datasheets, Application Notes, Programming Reference Manuals...*”
- Capacitat d’abastiment ràpid a través d’RS Components.
- Memòria no volàtil per tal de guardar certes dades.

3.2.2 Tria del microcontroladors

En el procés de selecció del microcontrolador i el kit de desenvolupament associat, se’n trien un total de tres:

- STM32L100C
- ATTINY416
- ATMEGA328P

A partir de les fitxes tècniques d’aquests i d’altres consideracions es determina el següent:

- STM32L100C: Es descarta l’ús del microcontrolador STM32L100RC per aspectes relacionats amb l’entorn de desenvolupament associat aquest. No pas per característiques tècniques.
- ATTINY416: Es descarta l’ús del microcontrolador ATTINY416 per aspectes tècnics. Es consideren insuficients el nombre de pins usables en entrades i sortides, hi hauran limitacions a l’hora d’implementar les diferents funcions durant el transcurs del treball.
- ATMEGA328P: El microprocessador escollit per a l’implementació del projecte.

En l’annex 5.3 es pot veure el procés complert de la tria del microcontrolador. Addicionalment s’inclouen les fitxes tècniques dels tres anteriorment esmentats.

3.2.3 Placa de desenvolupament XPlained ATMEGA328P

El microcontrolador ATMEGA328P esdevé l’escollit juntament amb la placa de desenvolupament XPlained en la que va integrat.

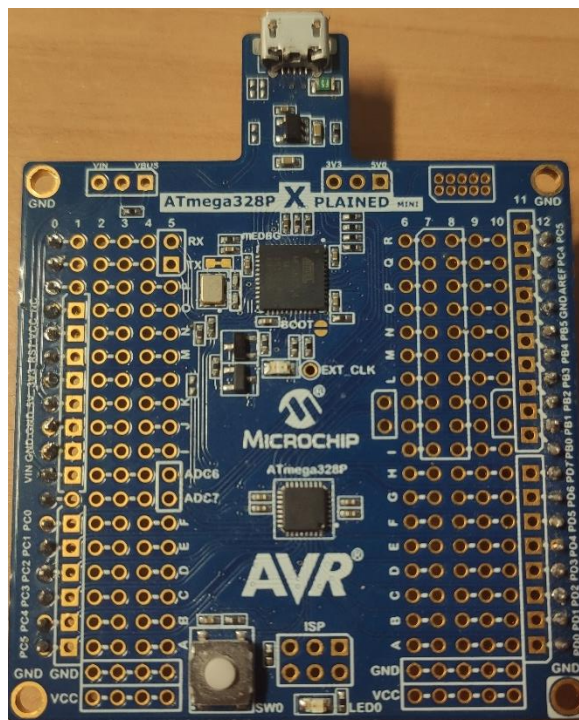


Figura 6. Xplained ATMEGA328P

Aquesta combinació de placa-microcontrolador, permet la programació del microcontrolador a través de USB. Addicionalment amb la connexió USB s'alimenta la placa. La tensió de funcionament dels diversos elements d'aquesta és de 5V.

Si es realitzés un canvi en el Jumper J301 de la placa (veure Figura 7), el microcontrolador passaria a funcionar a una tensió de 3.3V. Ara bé amb la disminució de tensió, hi hauria una disminució en la freqüència de rellotge del microcontrolador. A 5 V treballa a 16 Mhz mentre que en el cas de 3.3 V treballa a una freqüència de 8 Mhz. Aquesta disminució es **incompatible** amb el correcte funcionament prototip.

A més de dotar al microcontrolador dels elements necessaris per a funcionar de forma correcta, la Xplained permet estendre els pins de entrades i sortides del microcontrolador a una mida a la qual es permet soldar pins de 2,54 mm.

Taula 3. Hardware addicional necessari respecte als punt de l'apartat 2.4

Punt	Elements necessaris i comentari
1. “Start/stop” metrònom	Element necessari: Un botó. Comentari: S’afegeix una resistència de Pull up i un condensador per tal d’evitar falses commutacions.
2. Modificar consigna manualment	Elements necessaris: Dos botons. Comentari: El mateix que en el punt 1.
3. Visualització de la consigna	Element necessari: Ús d’un display LCD 2*16 Comentari: S’usa un display basat en el controlador Hitachi HD44780. Circuiteria externa per tal de dotar el display dels elements que necessita.
4. Funció tap amb element del instrument	Element necessari: Element piezoelèctric o transductor. Comentari: Enrutat del senyal de tensió del transductor fins al canal 1 del convertidor Analògic Digital.
5. Funció tap amb botons	Elements necessaris: Un botó. Comentari: El mateix que al punt 1.
6. Varis menús	Elements necessaris: Un botó. Comentari: Enrutat del valor de la tensió al botó cap al canal 0 del ADC.
7. Sortida a auriculars	Elements necessaris: Conversor DAC extern TLC5628CN1 ¹ . Comentari: Enrutat de totes les senyals necessàries.
8. Capacitat de guardar els programes	Elements necessaris: Memòria no volàtil. Comentari: Ús de la EEPROM pròpia del microcontrolador. No requereix hardware addicional.
9. Capacitat de reproduir els programes	Elements necessaris: Botons. Comentari: No requereix hardware addicional. Ús dels botons ja presents.

¹ Veure Annex 5.4.1 per a més informació sobre el component en qüestió.

3.4 PCB

S'integra el microcontrolador i els elements externs en una PCB per tal d'encapsular al màxim el prototip i reduir les interferències en els senyals. Es té cura del senyal de sortida a auriculars ja que serà la única sortida de tipus analògic i tindrà contacte directe amb l'usuari final.

3.4.1 Circuit electrònic

Al Annex 5.1 es mostra el circuit elèctric implementat en la PCB. El circuit ha estat dissenyat mitjançant el programa Capture Cis Lite.

La connexió senyalada pel requadre verd al Annex 5.1 es una consideració de disseny. Amb això i una correcta disposició de vies a l'hora de soldar permet discriminar entre l'ús o no de l'última etapa amplificadora del amplificador operacional TLC2242.

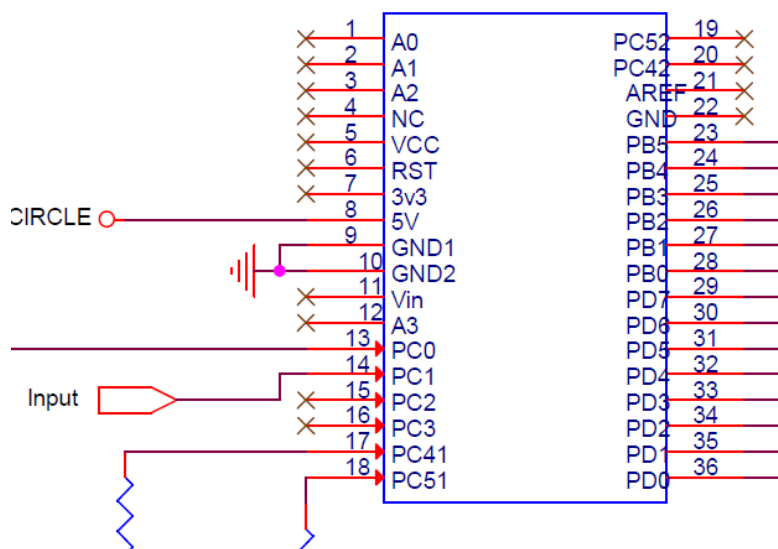


Figura 8. Símbol de la Xplained en el esquemàtic del circuit electrònic.

A la Figura 8 es pot veure el símbol que representa la Xplained a l'esquema. La sortida de 5V (pin 8) serà l'encarregada d'alimentar tots els circuits elèctrics de la PCB que ho requereixin. La massa serà la referència per a tot el circuit. Es pot comprovar com hi han 15 pins que no estan connectats, queden assenyalats amb una "X".

Dels diferents pins del microcontrolador es tracen pistes a tots els altres elements del circuit. A la Taula 4 es detalla l'ús de cada pin² al esquemàtic de forma ordenada.

² Els noms dels pins corresponents als que apareixen en la placa X-plained. Veure referència 2 de la webgrafia.

Taula 4. Ús de cada pin de la Xplained.

Branca esquerra		Branca dreta	
Pin	Ús	Pin	Ús
A0	Sense ús	PC52	Sense ús
A1	Sense ús	PC42	Sense ús
A2	Sense ús	AREF	Sense ús
NC	Sense ús	GND	Sense ús
Vcc	Sense ús	PB5	CLK out SPI, necessari per al DAC
Rst	Sense ús	PB4	Sortida PWM, dedicat a l'activació del buzzer
3v3	Sense ús	PB3	Camp de dades SPI, necessari per al DAC
5 V	Tensió de referència per a tots els elements del circuit extern.	PB2	"Enable" de la actualització del valor del DAC.
GND 1	Massa de tots els elements del circuit extern.	PB1	Display LCD
GND 2	Ídem a GND1	PB0	Display LCD
Vin	Sense ús	PD7	Display LCD
A3	Sense ús	PD6	Display LCD
PC0	Entrada ADC, ús en el botó 0	PD5	Display LCD
PC1	Entrada ADC, entrada de la senyal del sensor de vibració	PD4	Display LCD
PC2	Sense ús	PD3	Botó 1
PC3	Sense ús	PD2	Botó 2
PC41	Sortida al LED vermell	PD1	Botó 3
PC51	Sortida al LED verd	PD0	Botó 4

Com es pot comprovar s'estan usant quasi totes les entrades i sortides del microcontrolador. El botó 0, esta connectat al ADC per tal d'estalviar un pin el qual pot ser usat per una sortida

digital adicional. Encara es podria disposar de dos elements additionals del tipus entrades/sortides digitals en el cas que passéssim dos botons als pins PC2 i PC3. Aquests pins són entrades del ADC.

El fet d'afegir dos botons més dificultaria la programació i no es veu una funcionalitat extra rellevant per al dispositiu.

3.4.2 Traçat de pistes

Mitjançant l'ús del *PCB editor Lite* de *Cadence* es realitza el traçat de les pistes i la ubicació dels elements en la PCB. Es decideix realitzar una PCB de dos capes.

Es descarta usar una PCB de més capes pel sobre cost i el increment de complexitat a l'hora de soldar components i verificar el correcte traçat de les pistes un cop es manufactura la PCB.

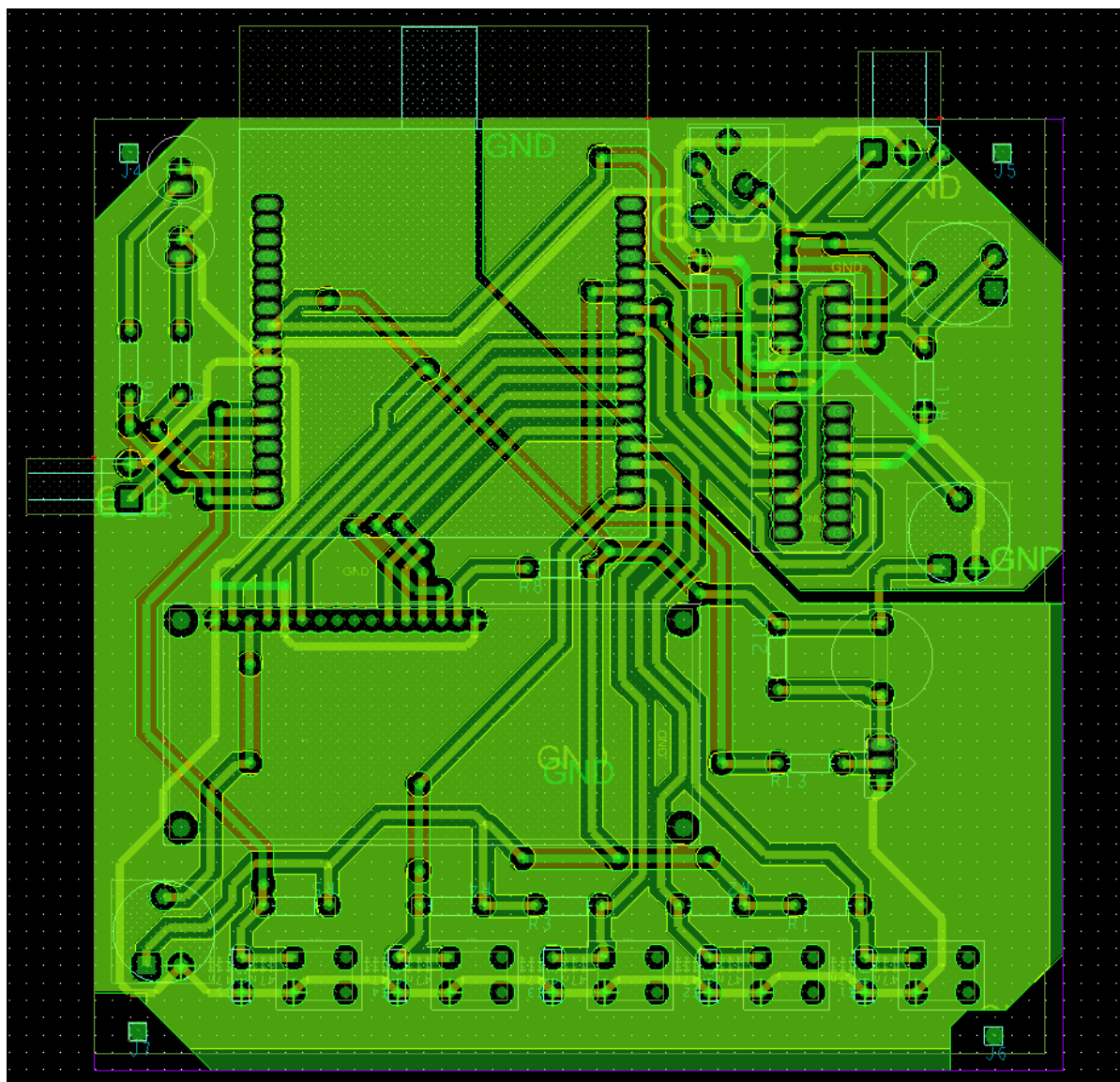


Figura 9. Traçat de les pistes i zones de coure de la PCB

En la Figura 9 es veuen superposades les dos capes de la placa, en verd la cara superior o “top” i en groc la cara inferior o “bottom”. Es pot comprovar com el verd es el color que impera i no permet veure el traçat de la cara inferior. A la Figura 10 veu el traçat de les pistes per la cara inferior.

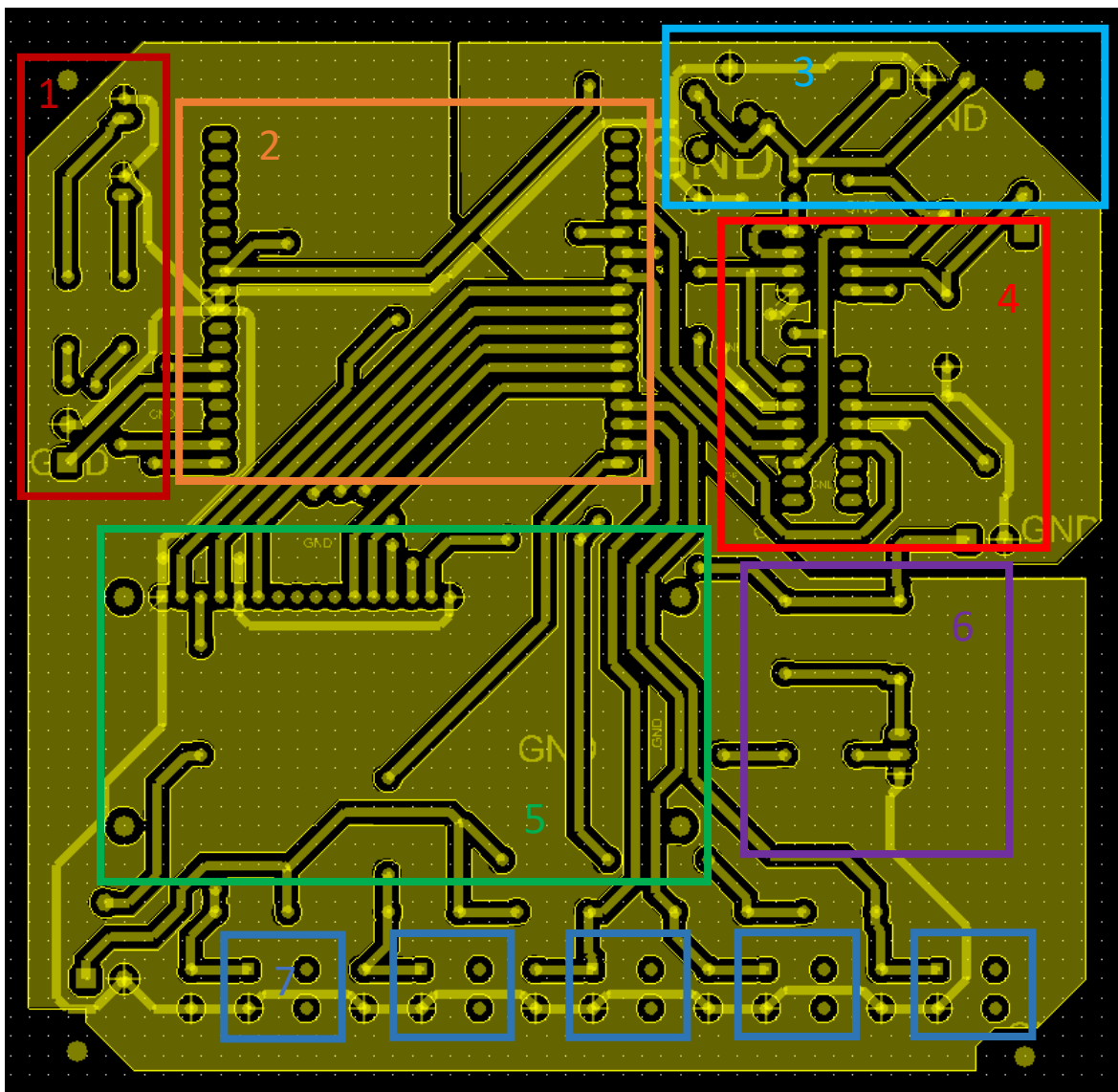


Figura 10. Traçat de les pistes per la cara inferior o “bottom”

En la Figura 10 podem situar els diferents components que conformen la placa.

Taula 5. Descripció dels elements enumerats en la Figura 10.

Bloc	Descripció
1	Bloc amb els LED verd i vermell. Hi ha una resistència prèvia als LEDs
2	Bloc amb el Atmega 328P i les connexions als pins d'aquest.

3	Bloc amb les dues sortides d'àudio, hi ha l'opció de connexió directa a auriculars via connector jack de 3.6 mm i l'opció a connectar directament dos cables a través d'un connector. Aquesta opció és present pel procés de disseny del producte.
4	Bloc amb el DAC (integrat de 16 pins inferior) i el amplificador (integrat de 8 pins superior). Hi ha presència dels elements necessaris per tal de dotar aquests elements de les tensions d'alimentació necessàries.
5	Bloc on esta ubicat el display LCD de 2*16 segments.
6	Bloc on està ubicat el "buzzer" i tots els elements necessaris per al seu accionament.
7	Ubicació dels botons. Hi ha presencia de condensadors de Hold up.

En la ubicació de cadascun dels components s'ha buscat que el traçat de la pista associada a les senyals sigui reduïda al màxim.

Les pistes són de 50 mils (1,27 mm) d'ample, aquest valor ofereix una resistivitat baixa mentre que no és una mida excessivament ampla que dificulta el traçat de les pistes.

Adicionalment, s'estableixen zones de coure en la cara superior i inferior de pla de massa per a que no apareguin punts on hi hagi una diferència de potencial entre masses. Amb això s'afavoreix el retorn dels corrents i es redueix la mida dels bucles inductius. Adicionalment, es realitza una separació entre les masses de la zona de sortida d'àudio i les masses generals.

3.4.3 Manufactura i soldadura de la PCB

Un cop realitzat el disseny, es generen els fitxers Gerber a partir de la PCB dissenyada. Un cop generats s'envien els fitxers a una empresa per tal de que manufacturi la PCB.

El proveïdor escollit per tal que realitzi la placa es JLCPCB. El cost i la ràpida manufactura fan que aquest sigui el proveïdor escollit.

Una de les dificultats de treballar amb aquest proveïdor es causada pel fet de estar en fusos horaris molt diferenciats. Previ a la manufactura de la PCB es van realitzar una sèrie de modificacions seguint les indicacions de JLCPCB. Un cop realitzades la placa va tardar un total de 10 dies naturals a arribar. Cal comentar que es va produir un endarreriment en la recepció de la PCB pel fet que aquesta va haver de passar per el despatx de duanes.

Un total de 5 PCB han estat manufacturades, el fabricant no permetia la realització de menys unitats.

A la Taula 6 es veuen els paràmetres constructius de les plaques realitzades per JLCPCB.

Taula 6. Paràmetres de les PCB realitzades.

Cares	2	Gruix coure	1 oz
Color	Verd	Recobrint	HASL (amb plom)
Material dielèctric	FR4 Standard TG	Gruix dielèctric	1.6 mm
Altura	139.7	Amplada	137.2 mm

A la Figura 11 i Figura 12 es pot veure la cara superior i inferior de les PCB a la seva recepció.

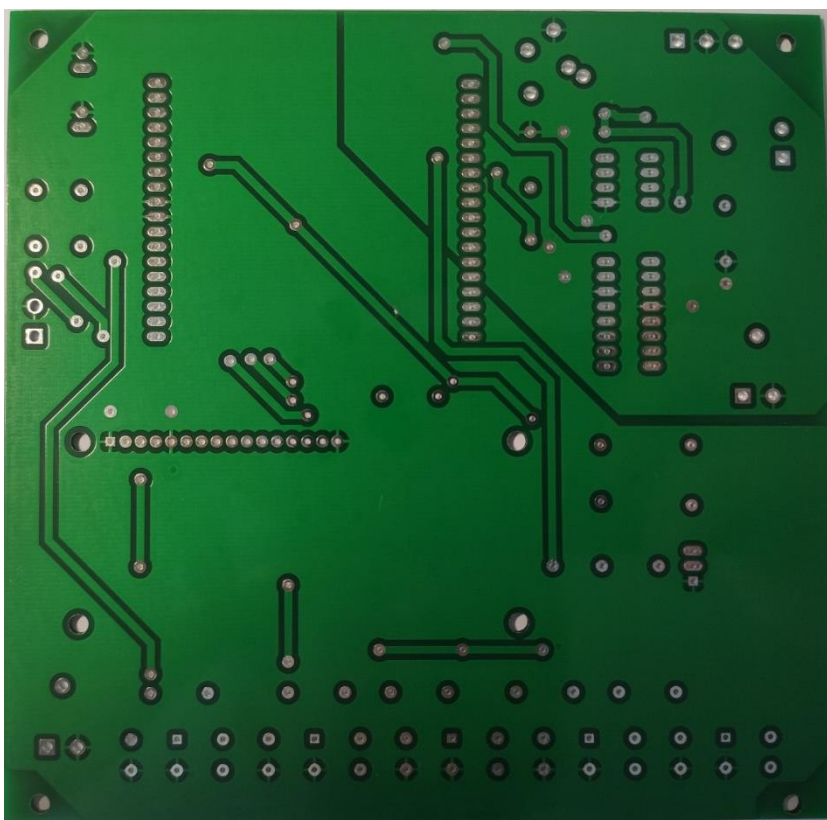


Figura 11. Cara superior de la PCB.

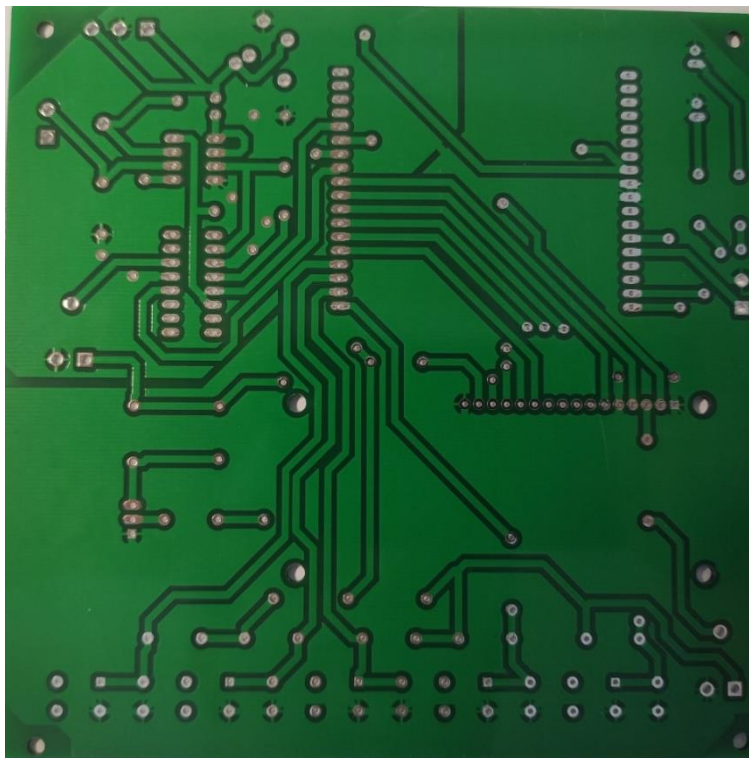


Figura 12. Cara inferior de la PCB.

Un cop es rep la placa i després de realitzar una comprovació de continuïtats es procedeix a la soldadura dels diferents components de la placa.

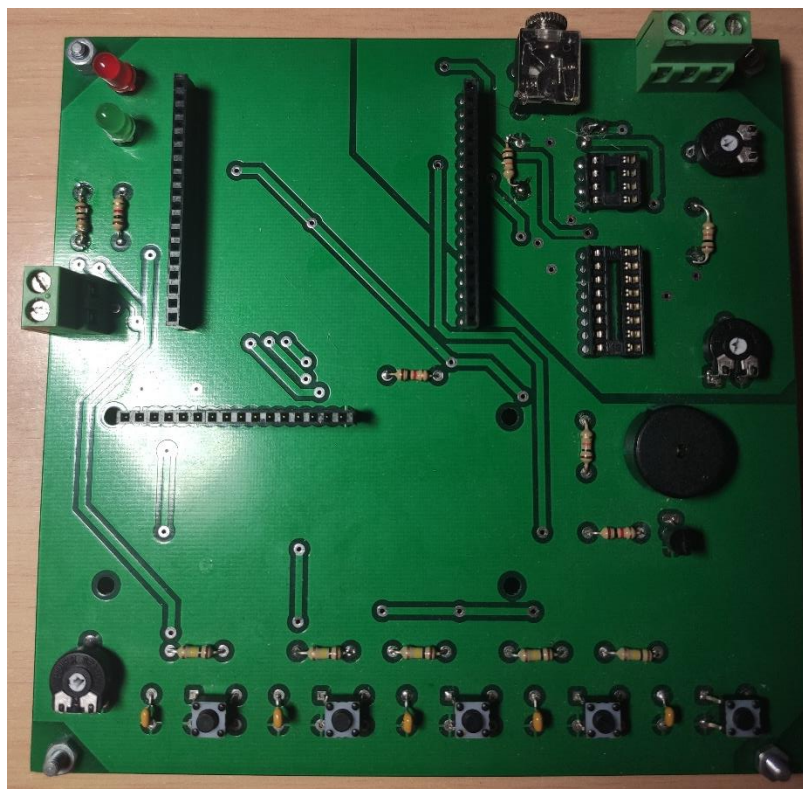


Figura 13. Estat de la cara superior un cop s'ha realitzat el procés de soldadura.

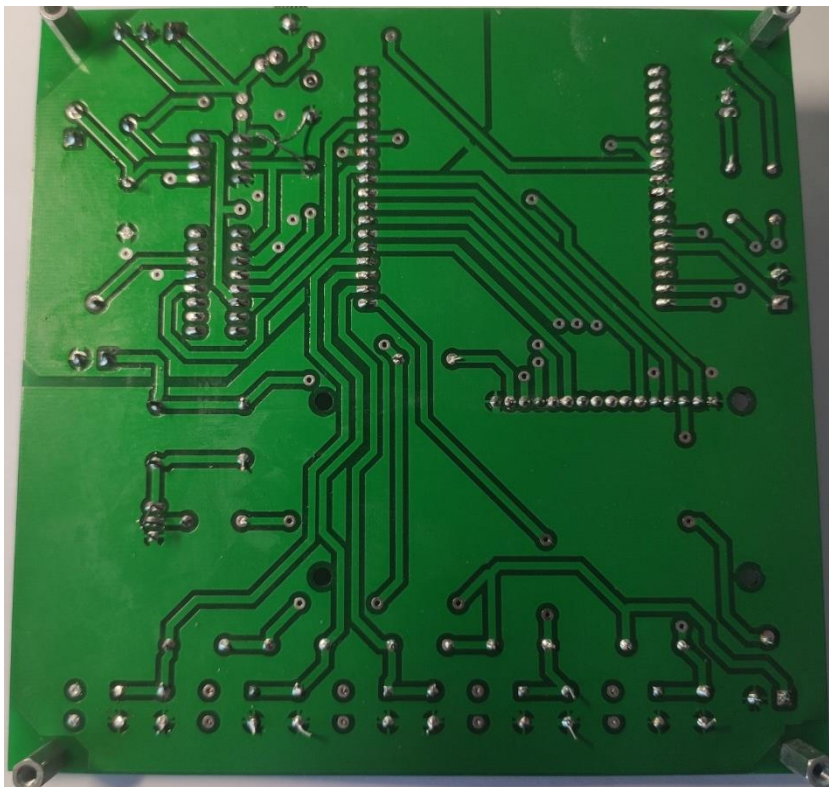


Figura 14. Estat de la cara inferior un cop realitzat el procés de soldadura.

En la Figura 13 es pot comprovar com hi han 3 tires de pins. Aquestes serveixen per la connexió del display LCD i el microcontrolador. S'opta per usar pins donat que seria més fàcil reaprofitar el LCD i el microcontrolador en el cas que la placa tingues algun defecte intern. També s'evita que tant el LCD com el display estiguin subjectes a una temperatura massa alta durant el procés de soldadura.

Anàlogament a l'ús dels pins, s'han usat sòcols per tal d'instal·lar els circuits integrats a la placa per les mateixes raons anteriorment comentades.

El procés de soldadura s'ha vist simplificat en gran part per l'elecció dels "pads" a l'hora de realitzar la placa. El haver imposat uns diàmetres més grans ha facilitat la soldadura. Addicionalment, els pads ja venien estanyats per part del fabricant.

Finalment les vies presentaven continuïtat a diferència del que es pensava fruit del procés de fabricació.

3.5 Programació

A continuació es descriu el codi del metrònom a grans trets. Es d'interès entendre'n el funcionament donat que va relacionat amb el disseny del hardware.

Primerament es comenta una terminologia específica per aquest apartat del treball, a continuació es comenta l'estructura del programa a grans trets per després entrar en quins menús disposa el dispositiu. Finalment es comenten els aparats que son tècnicament més complexos.

El codi complert del prototip es troba en el annex 5.2.1.

3.5.1 Terminologia

- **Cop:** Cada vegada que es pitja el botó 4 quan s'està creant una nova consigna o bé quan el ADC considera que la lectura es suficientment alta per considerar que s'ha produït un cop al element terminal.
- **Consigna:** Valor en BPM o "*beats per minute*" (cops per minut) que te el metrònom assignat.
- **Encendre el metrònom:** Els leds vermell i verd fan un display d'acord al valor de consigna. La sortida en àudio reproduceix un so d'acord a la consigna. Si esta habilitat el "*buzzer*" emet un so d'acord a la consigna.
- **Parar el metrònom:** Els LED, la sortida en àudio i el buzzer (si està habilitat) deixen de funcionar.
- **Creació nova consigna:** A través del còmput total del temps que ha passat entre la primera vegada que s'ha detectat cop i la quarta es genera una nova consigna.

3.5.2 Estructura del programa

En la Figura 15 es pot observar el diagrama del bucle principal d'execució.

En el bucle principal d'execució es comprova si es necessari actualitzar el display LCD. En el cas que sigui necessari, es realitzen els canvis necessaris.

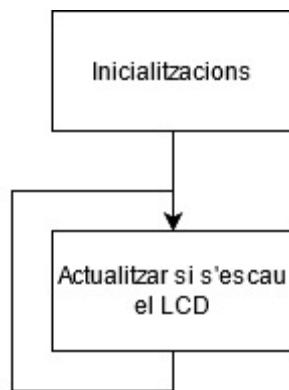


Figura 15. Diagrama del bucle principal d'execució

El programa compta amb un total de 6 interrupcions que de forma síncrona o asíncrona executen el codi pertinent. A la Taula 7 es descriuen les interrupcions utilitzades, quina es la seva funcionalitat en el prototip i quin esdeveniment les dispara.

Taula 7. Llistat de interrupcions. Estan ordenades de mes a menys prioritària. Les prioritats al ATmega328P són fixes.

Vector Interrupció	Disparo	Descripció del ús
PCINT2	Event	La interrupció es dispara cada vegada que hi ha un canvi en els pins connectats als botons 1, 2, 3 i 4.
TIMER2_COMPA	Temps T = 1 ms	La interrupció es dispara cada vegada que el compte intern del temporitzador assoleix un valor establert. Esta configurat que es dispari a una freqüència de 1000 Hz (cada 1 ms). Aquesta interrupció s'usa per la generació de la consigna.
TIMER1_COMPA	Temps T = 1 ms	La interrupció es dispara cada vegada que el compte intern del temporitzador assoleix un valor establert. Esta configurat que es dispari a una freqüència de 1000 Hz (cada 1 ms). Aquesta interrupció s'usa per a la visualització de la consigna a través dels leds, la sortida de audio i el buzzer.
TIMER0_OVF	Temps T = 20 ms	La interrupció es dispara cada vegada que el compte intern del temporitzador assoleix el valor màxim. Esta configurat que es dispari a una freqüència de 50 Hz (cada 20 ms) Aquesta interrupció s'usa per a realitzar peticions de conversió al convertidor Analògic Digital. S'usa tant per al botó 0 com per a realitzar lectures del transductor.
SPI_STC	Event	La interrupció es dispara cada vegada que ha acabat la transmissió de 1 byte a través de la SPI
ADC	Event	La interrupció es dispara cada vegada que el convertidor Analògic/Digital ha acabat la conversió.

3.5.3 Menús

Per tal d'entendre el funcionament del codi es necessari veure els diversos menús que hi ha en el dispositiu i a quines funcionalitats donen accés cadascun.

En el prototip es disposa de tres menús. Per a realitzar el canvi entre els diferents menús, el botó que ha de pitjar l'usuari és el 0 (veure Figura 2).

Menú 1

Menú principal, per defecte el prototip s'inicia en aquest menú. En la Figura 16 es pot veure la visualització del display en aquest cas.

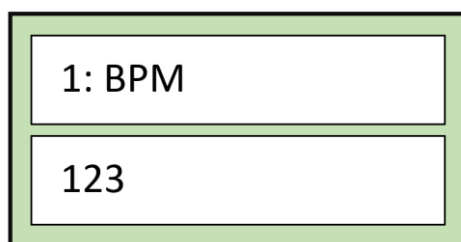


Figura 16. Visualització del display al menú 1.

A la Taula 8 es descriu l'ús dels botons en aquest mode. L'ús dels botons pot repercutir en una actualització dels valors presents al display.

Taula 8. Ús dels botons en el Menú 1.

Botó	Funció
1	Baixa 1 BPM la consigna
2	Puja 1 BPM la consigna
3	Para/Encén el metrònom
4	Creació d'una nova consigna i s'encén el metrònom
Funcionalitat extra	Si abans de pitjar el botó 4, es pitja el botó 3 i es manté premut durant els quatre cops necessaris per a generar la consigna, s'actualitza el valor de consigna però no s'encén el metrònom.

Menú 2

Menú que permet navegar entre els diversos programes guardats en el dispositiu. En la Figura 17 es pot veure la visualització del display en aquest cas.

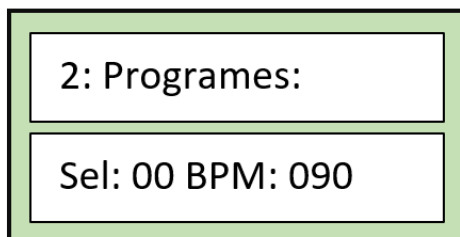


Figura 17. Visualització del display al menú 2

A la Taula 9. es descriu l'ús dels botons en aquest mode. L'ús dels botons pot repercutir en una actualització dels valors presents al display.

Taula 9. Ús dels botons en el Menú 2.

Botó	Funció
1	Baixa 1 el programa seleccionat
2	Puja 1 el programa seleccionat
3	Para/Encén el metrònom en la consigna que hi ha en el programa seleccionat.
4	Al pitjar tres cops seguits es guarda al programa seleccionat el valor de consigna que tingui el metrònom

Menú 3

Menú destinat a la configuració del metrònom per part de l'usuari. En la Figura 18. es pot veure la visualització del display en aquest cas.

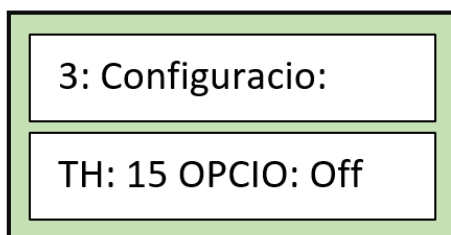


Figura 18. Visualització del display al menú 3

A la Taula 10 es descriu l'ús dels botons en aquest mode. L'ús dels botons pot repercutir en una actualització dels valors presents al display.

Taula 10. Ús dels botons en el Menú 3

Botó	Funció
1	Baixa 1 el llindar o “Threshold” del ADC
2	Puja 1 el llindar o “Threshold” del ADC
3	“Toggle” entre on i off. S’habilita o no que el “buzzer” emeti sons seguint la consigna.
4	Es guarda la configuració

3.5.4 Generació de la consigna

La generació de la consigna és un dels apartats més importants pel que fa a la programació del metrònom. L’usuari ha de ser capaç de generar una nova consigna mitjançant o bé la funció tap del botó 4 o l’ús del transductor connectat a l’instrument.

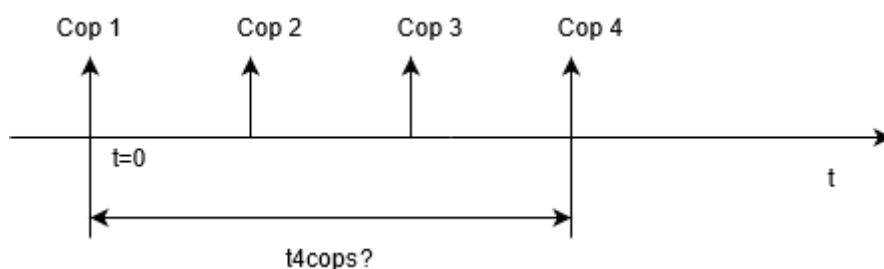


Figura 19. Temps necessari per a la generació de la consigna

Tal i com es veu en la Figura 19, per tal d’extreure la consigna es necessari veure el temps que passa entre la realització de quatre cops. Un cop es tingui el temps es pot extreure el valor de la nova consigna. En els metrònoms la consigna que s’usa es BPM, acrònim de *beats per minute* (cops dividit per minut). Un cop es té el temps que passa entre els quatre cops, la consigna es pot extreure seguint les formules de l’Equació 1 i l’Equació 2.

Equació 1. Càlcul dels cops per segon

$$f_{cop} = \frac{cops}{s} = \left(\frac{t_{4cops}(s)}{3} \right)^{-1}$$

Equació 2. Càlcul dels BPM a partir de la freqüència de cop

$$bpm = \frac{cops}{minut} = f_{cop} * 60$$

Cal destacar que el microcontrolador en sí no compta temps com a tal, sinó que s’ha d’usar un mecanisme via *Timers* (temporitzadors) que es comenta més endavant.

Detecció de cops

Aquest bloc es el primer dels blocs responsables de generar la consigna.

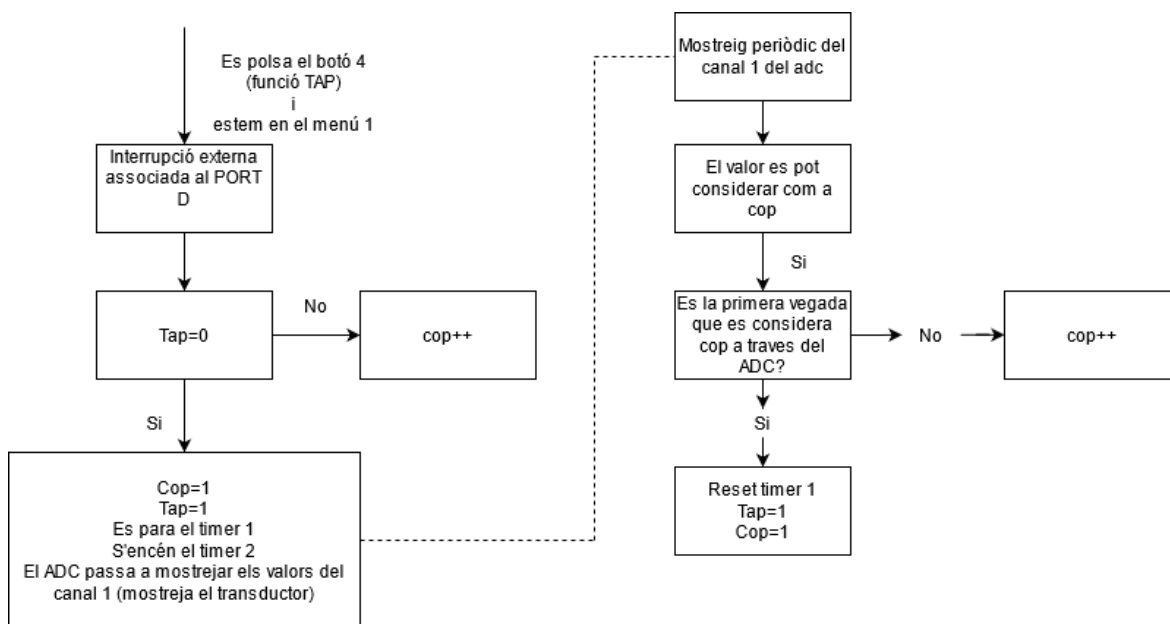


Figura 20. Diagrama de la actualització de "cop".

La generació de la consigna comença prement el botó 4 sempre que s'estigui al menú 1. La primera vegada que es premi el botó s'iniciarà el Timer 2 i es parerà el Timer 1. El ADC passa a mostrejar el canal 1 fent lectures del valor que entrega el transductor. El fet de passar al canal 1 fa que el canvi de menú no sigui possible mentre s'estigui realitzant el canvi de consigna.

En el cas que la consigna es vulgui generar únicament via botons, cada cop que es premi el botó s'incrementarà el valor de cop.

Ara bé si via el ADC es detecta que s'ha donat un cop, es reinicia el temps que esta comptant el Timer 1 i es fica cop a 1 sigui quin sigui el cas. A partir d'aquest instant es pot introduir el cop via el transductor que esta connectat al element físic com a través del botó 4.

Timer 2

Aquest es el segon bloc encarregat de la generació de la consigna juntament amb la detecció de cop.

El Timer 2 es un dels elements que ha requerit mes consideracions de disseny. L'objectiu es establir un sistema que doni la suficient resolució per tal de que el sistema tingui la màxima capacitat de detecció de consigna.

L'actualització del valor de cop ve a partir del bloc *Detecció de cops*. El Timer 2 mirarà si cop es igual a 4 cada cert temps. En el cas que sigui així, elaborarà la nova consigna.

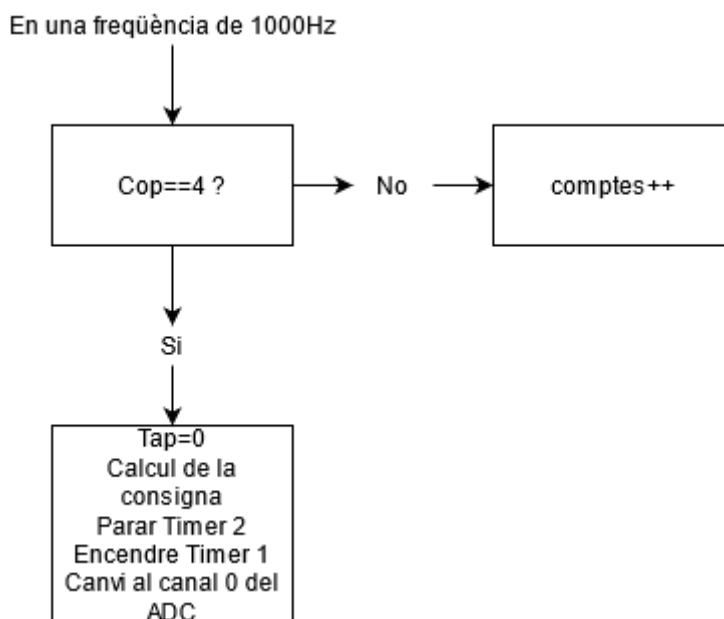


Figura 21. Diagrama del Timer 2

Tal i com s'ha comentat amb anterioritat, el microcontrolador no es capaç de comptar temps com a tal. En aquest cas s'implementa un temporitzador via software. És a dir, amb una certa freqüència s'entrarà a una RSI (Rutina de Servei a la Interrupció) i dins d'aquesta hi haurà una variable que comptarà les vegades que s'ha realitzat el pas per la rutina.

Per tant hi ha un paràmetre de disseny que s'ha d'escollir, i es la freqüència en la que s'ha d'entrar a la RSI. Aquesta freqüència s'ha d'escollir tenint en compte que s'ha de dotar al metrònom de la capacitat de detecció suficientment ràpida per a tindre prou precisió en tots els rangs. En la Taula 11. s'il·lustra la problemàtica.

Taula 11. Càlcul dels temps entre 4 cops i diferències de temps entre dos consignes en rangs alts i baixos de BPM.

BPM	t_{4cops} (s)	Δt (s)	BPM	t_{4cops} (s)	Δt (s)
Part alta de BPM			Part Baixa de BPM		
300	0,6		40	4,5	
299	0,60200669	0,00200669	39	4,61538462	0,11538462
298	0,60402685	0,00202016	38	4,73684211	0,12145749
297	0,60606061	0,00203376	37	4,86486486	0,12802276
296	0,60810811	0,0020475	36	5	0,13513514
295	0,61016949	0,00206138	35	5,14285714	0,14285714
294	0,6122449	0,00207541	34	5,29411765	0,1512605
293	0,61433447	0,00208957	33	5,45454545	0,16042781
292	0,61643836	0,00210389	32	5,625	0,17045455
291	0,6185567	0,00211834	31	5,80645161	0,18145161
290	0,62068966	0,00213295	30	6	0,19354839

Es pot veure que per a valors alts de consigna la diferència de t_{4cops} entre 300 BPM i 299 BPM es de 0.002 s en mentre pel que fa a valors baixos de BPM la diferència es de la ordre de 0.2s.

Equació 3. Càlcul de la freqüència a la qual correspon un període de 0.002 s

$$f = \frac{1}{T} \rightarrow \frac{1}{0.002} = 500 \text{ Hz}$$

Si s'aplica el teorema de *Nyquist*, es veu que la freqüència a la qual el microcontrolador ha d'entrar a la RSI és de 1000 Hz. Amb aquesta freqüència dotarem el sistema de suficient precisió a l'hora de treballar en consignes altes. Donat que la diferència es més alta en valors baixos de BPM també es tindrà precisió en aquest rang.

Per tal d'aconseguir que el Timer 2 entri dins de la RSI en una freqüència de 1000 Hz s'han configurat els següents paràmetres:

- "Prescaler" de la freqüència de rellotje del timer és 64.
- Registre ORCA1= 249.

El *Prescaler* es el factor de divisió del rellotje que usa el timer respecte la freqüència de rellotje del microprocessador.

Com que la freqüència a la que treballa al microcontrolador és 16 MHz, els valors anteriors es justifiquen de la següent forma³:

Equació 4. "Prescaler" emprat.

$$f = \frac{f_{clock}}{prescaler * 2 * (1 + ORCA1)} = \frac{16 * 10^6}{64 * 250} = 1000 \text{ hz}$$

3.5.5 Ús EEPROM

Donat que es pretén implementar un prototip el qual pugui guardar una sèrie de consignes configurades per l'usuari, es necessari establir un mecanisme pel qual aquestes dades es guardin en memòria no volàtil.

El microcontrolador disposa de dos tipus de memòria no volàtil, flash i EEPROM. En la primera és on es guarda el codi de programa que el microcontrolador executa cada cop que està alimentat elèctricament.

Es considera utilitzar la memòria EEPROM per tal de guardar les dades en memòria no volàtil. S'escull aquesta memòria pel fet que la memòria EEPROM disposa de 100.000 cicles de lectura/escriptura segons el *datasheet*.

³ Formula extreta del de la referència 1 de la webgrafia. DS470002061-B, pagina 132. El dos es eliminat donat que no s'està implementant la funció "toggle" a cada pas per el OC1A.

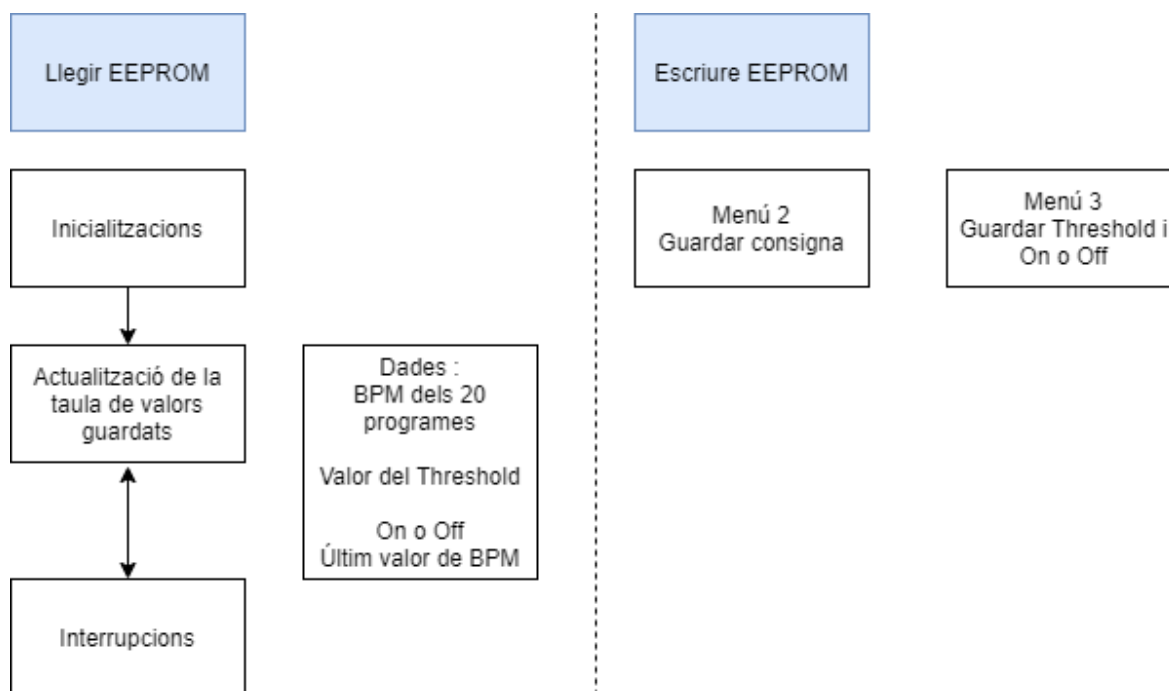


Figura 22. Diagrama sobre l'ús de la EEPROM.

Les dades que són escrites/llegides són 20 consignes, el valor del *Threshold* i si el buzzer esta a On o Off. Les 20 consignes son escrites/llegides de la EEPROM com a 2 bytes donat la consigna es una variable de tipus *short* (16 bits). Pel que fa al *Threshold* i si el buzzer esta a on o off, són variables de tipus *char* (8 bits).

A l'inicialitzar el prototip es realitza una lectura de la memòria EEPROM. A partir dels valors llegits, s'actualitza una taula de valors. Les diferents interrupcions llegeixen els valors provinents de la EEPROM a través de la taula de valors.

Pel que fa a l'escriptura de la EEPROM, aquesta es realitza des de dos llocs diferents. El primer es des del **Menú 2**, on al clicar tres cops el botó 4 es guarda al programa indicat la consigna que té el metrònom en aquell moment.

El segon lloc es des de el Menú 3, on al clicar el botó 4 es guarda el valor que hi ha configurat de *Threshold* i si esta habilitat o no el buzzer.

3.5.6 Ús llibreries externes

En el codi del prototip s'ha utilitzat una llibreria externa. Aquesta s'ha utilitzat per a realitzar la escriptura de cadenes de caràcters al display LCD de 2*16 segments.

Gran part dels displays LCD de 2*16 segments venen controlats per el mateix driver, el HD44780 de HITACHI. Aquest va ser desenvolupat a la dècada dels 80 i ha estat l'estàndard en aquest camp.

Això fa que hi hagi gran disponibilitat de llibreries dedicades als diferents microcontroladors del mercat, i que hi hagin alguns microcontroladors que portin incorporat el driver per així poder controlar amb més facilitat la pantalla LCD.

En el cas del projecte s'ha usat la següent llibreria.

Autor: ElectroSaurio

Enllaç descarrega:

<https://drive.google.com/drive/folders/1c8i6nc60kUsVsqpWJ0A01crZLzVu5bw>

A l'annex 5.2.2 es pot veure el codi complert de la llibreria. Un cop s'ha entès el funcionament de la llibreria, aquesta ha estat reduïda després de comprovar que hi havia línies de codi redundants.

En la comunicació entre el microcontrolador i el driver és molt rellevant la seqüència en el enviament dels diferents caràcters i que es respectin una sèrie de temps per tal que el driver pugui actualitzar els camps internament.

Es per això que hi ha presència de rutines de *delay*. La crida a les rutines associades al canvi de la visualització en el display LCD es realitzen des del bucle principal d'execució i no des de les rutines de interrupció per evitar retards en les prestacions.

3.6 Caixa prototip

Per tal d'encapsular el prototip final, es realitza una caixa mitjançant impressió 3D.

En el disseny de la caixa s'ha tingut amb compte la senzillesa del muntatge. Com que es preveu que el prototip s'usi en entorns com podria ser un concert, o una sala d'assaig, s'ha de considerar que la caixa tingui una bona resistència mecànica als cops.

3.6.1 Disseny

Mitjançant Nanocad 3D es dissenya la caixa del prototip. Aquesta la formaran dos peces, la tapa i la part inferior.

Tapa

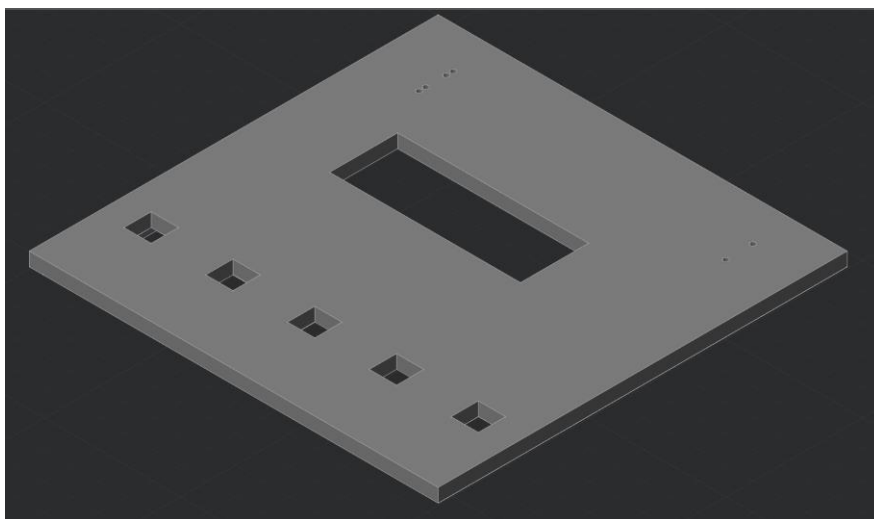


Figura 23. Perspectiva superior de la Tapa

La tapa és un rectangle de 150*150 mm amb cinc orificis per a situar els polsadors, dos parells d'orificis per a situar els LED i un parell de forats per a situar al Buzzer. Té un gruix de 5 mm.

Adicionalment es pot veure un forat de grans dimensions que es per on es podrà veure el display LCD de 2*16 segments.

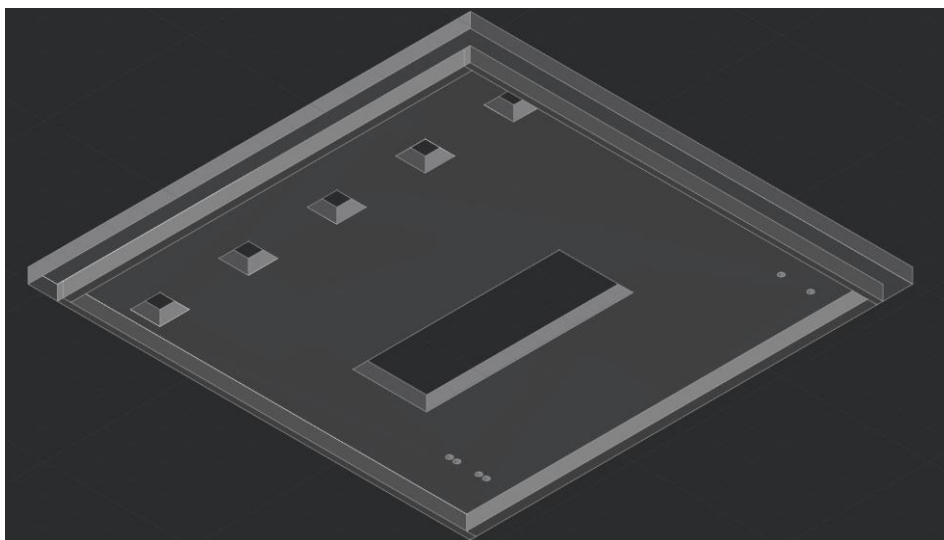


Figura 24. Perspectiva inferior dreta de la tapa.

En la Figura 24 es pot veure com per la part inferior de la tapa hi ha un detall el qual permetrà que la tapa es situï de forma correcta a la part inferior de la caixa.

Part inferior

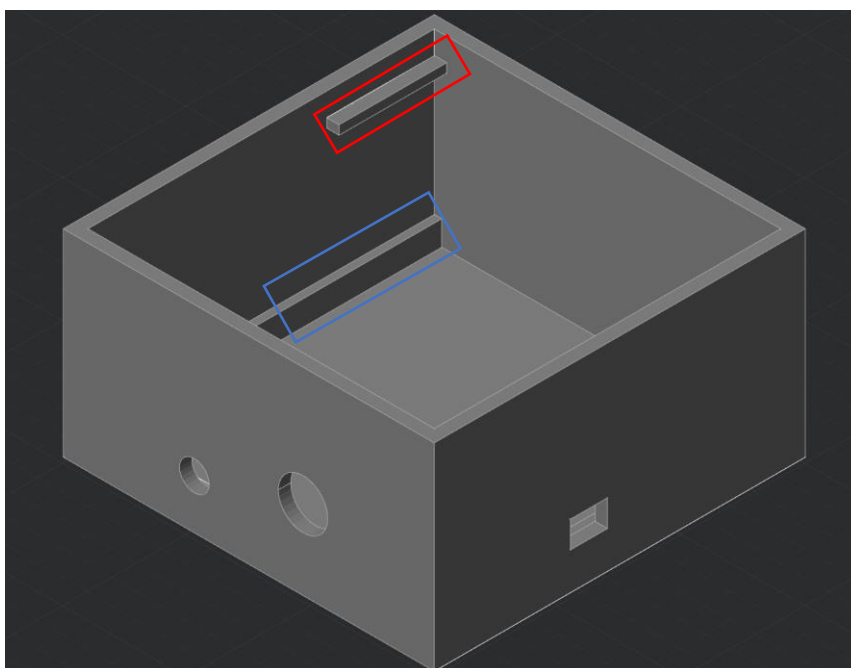


Figura 25. Perspectiva de la part inferior de la caixa.

La part inferior de la capsua mesura 150 * 150 mm amb una altura de 80 mm. Té un espai interior de 140 * 140 mm fruit dels 5 mm que té d'espessor a les parets. Es poden veure un total de 3 orificis. Els tres son necessaris donat que el prototip té interacció amb elements exteriors.

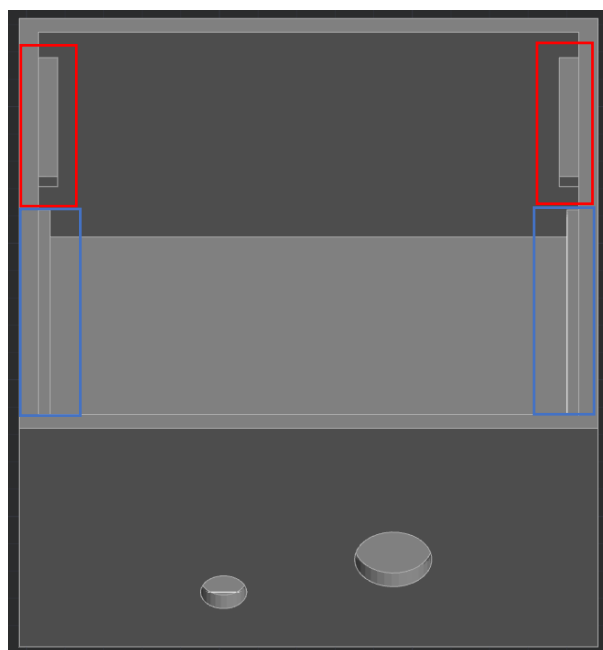


Figura 26. Perspectiva superior de la part inferior.

En la part inferior d'aquest element es situarà la placa PCB. Per això s'introdueix als dos costats un petit llit (element en blau) per tal de situar-hi la placa a sobre.

En la part superior es poden trobar unes petites pestanyes situades per tal de poder aguantar una placa perforada que contingui els botons (element en vermell).

3.6.2 Impressió i resultat final

La caixa va ser realitzada gràcies a la impressora 3D del DEEEA (Figura 27). El model en qüestió es una BCN 3D Sigma R19.

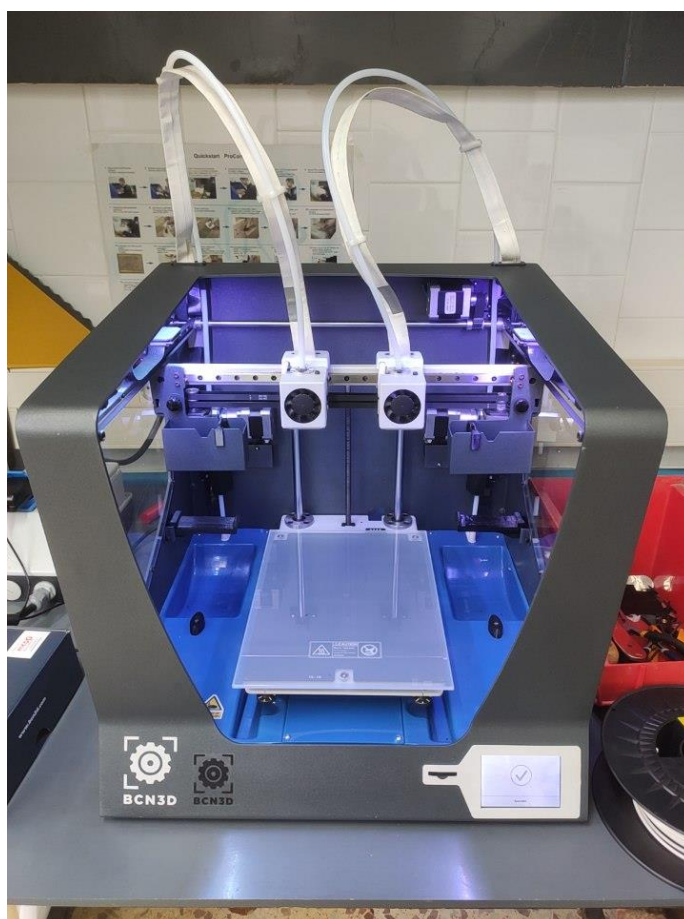


Figura 27. Impressora 3D del DEEEA

Aquesta impressora usa el mètode d'impressió *Fused filament fabrication*. El mètode es basa en fondre material plàstic per posteriorment situar-lo en la posició X,Y, Z determinada.

El material emprat en la impressió ha estat el PLA. A la Taula 12 es veuen els temps de fabricació de les peces i el pes un cop fabricades.

Taula 12. Temps de fabricació i pes de les peces

Peça	Temps de fabricació	Pes
Tapa	8 h	69 g
Part inferior	1 dia 18 h	239 g

A la Figura 28 i Figura 29, es poden veure els resultats de la tapa i la peça inferior. El fet que la tapa tingui un rectangle interior de 140 *140 mm fa que no sigui necessari l'ús d'elements externs per a que la caixa quedi tancada de forma correcta.

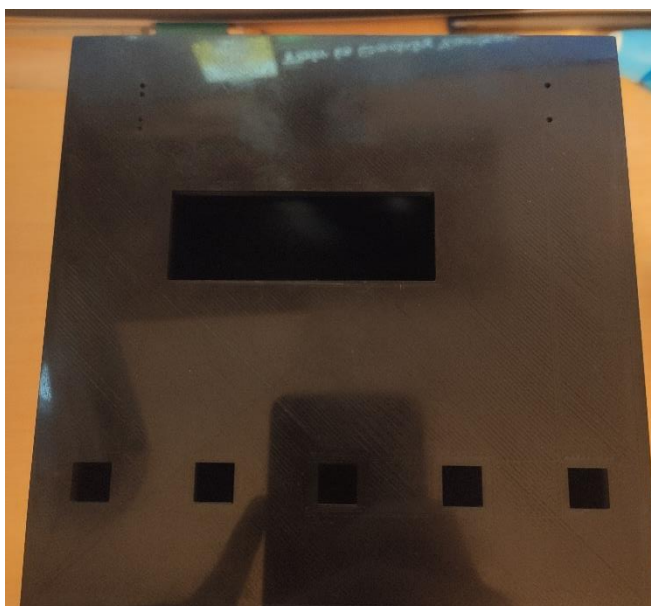


Figura 28. Tapa un cop impresa enquestada en la part inferior



Figura 29. Part inferior un cop impresa

3.7 Cost prototip

En aquest apartat es descriuen els costos materials i de desenvolupament del prototipus.

3.7.1 Materials

Taula 13. Llistat de materials del prototip. Es trien components subministrables per poques unitats i que tinguin muntatge THT.

Component	Codi Rs	Preu	Quantitat	Total
PCB	JLC PCB	5	1	5
Atmega 328P-Xplained	130-6148	8,47	1	8,47
Transductor	893-7086	4,32	1	4,32
Buzzer	457-011	0,564	1	0,564
TLC5628CN	200-6640	6,49	1	6,49
TLC2272AC	331-2176	3,95	1	3,95
Display Lcd	Amazon	6,35	1	6,35
Transistor BJT	124-1315	0,042	1	0,042
Botons	283-9951	0,38	5	1,9
Resistor 100 kΩ	707-7824	0,142	5	0,71
Resistor 10 kΩ	707-7745	0,125	2	0,25
Resistor 1 kΩ	707-7666	0,125	4	0,5
Resistor 100 Ω	707-7587	0,141	1	0,141
Potenciòmetre 10 kΩ	Farnell (3128459)	0,621	3	1,863
Condensador 100 nF	181-6522	0,113	5	0,565
LED Vermell	228-5988	0,3	1	0,3
LED Verd	228-6004	0,418	1	0,418
Conjunt connector aeri i PCB 90°	Amazon	0,5	1	0,5

Pins femella	Amazon	0,5	1	0,5
Pins mascle	Amazon	0,5	2	1
Sòcol 8 potes	185-9851	0,184	1	0,184
Sòcol 16 potes	185-9864	0,153	1	0,153
Sortida Jack 3.6 mm	913-1030	2,038	1	2,038
Cable connexió		0,5	2	1
Encapsulat mecànic		1	7(orientatiu)	7
Suma				56,246
Iva 21%				11,81166
Total				68,05766

El preu dels materials es veuria reduït en el cas que es decidís realitzar una quantitat més àmplia de prototips. En el annex 5.6.1 es veu el cost dels mateixos materials en el cas que es realitzessin 200 metrònoms.

En realitzar una quantitat més gran de metrònoms, s'hauria d'optar per components SMD. Això permetria reduir el encapsulat del prototip i al seu torn reduiria els costos dels components electrònics. Ara bé, en el cas que s'optés per soldar els components de forma manual implicaria una inversió important pel que fa a equips de soldadura.

3.7.2 Cost mà d'obra

Es considera un total de 2 hores/ metrònom per tal de realitzar les tasques necessàries per a al muntatge del prototip. Es considera la soldadura manual de components THT.

3.7.3 Cost d'enginyeria

En el disseny s'ha requerit un total de 150 hores de enginyeria per tal de desenvolupar el prototip. En aquest còmput d'hores es tant la creació del software i hardware del metrònom com per la validació del funcionament d'aquest.

En els costos d'enginyeria també s'han de considerar els costos derivats de les proves realitzades en hardware previ al definitiu.

En el desenvolupament del producte primerament es van realitzar proves en una placa *protoboard*. Un cop es validat el primer disseny, es realitza una "*placa de topos*" amb un disseny més avançat i més complert pel que fa a components.

A partir d'aquest disseny es realitza la PCB final.

3.7.4 Cost final

A partir del comentat en els apartats anteriors es determinen un seguit de costos i es calcula el cost del desenvolupament del prototip.

Concepte	Preu (€)	Quantitat	Total (€)
Material	66,33	1 unitat	66,33
Ma d'obra per unitat	15 €/h	2 h	30
Enginyeria	20 €/h	150 h	3000
Costs addicionals enginyeria	200	1	200
		Total	3296 €

3.8 Estudi econòmic prototip

Es realitzaran dos estudis econòmics del prototip considerant una sèrie de paràmetres.

Primer estudi econòmic

Aquest estudi econòmic es realitzarà a partir del mateix disseny realitzat sense cap tipus de modificació. Es consideren els següents paràmetres.

- Manufactura manual del metrònom.
- Soldadura dels components THT.
- Considerant un total de 70 € de cost per unitat. Es considerarà que a partir de les 200 unitats el cost per unitat sigui de 15 € menor. Això es gràcies a la bonificació que realitzen els proveïdors quan compres quantitats àmplies tant de components electrònics com de plaques PCB. Veure annex 5.6.1.
- Considerant un preu de venda de 120€.

Es vol comprovar quina haurà de ser la quantitat de prototips realitzats per tal de amortitzar la inversió realitzada en enginyeria.

En la Figura 30. es pot comprovar com a partir de les 160 unitats es comencen a veure els beneficis. Fins llavors encara s'estaran amortitzant els costos d'enginyeria. Cal comentar que s'ha considerat un preu de venda de 120€ per a il·lustrar quan es tardaria a amortitzar la inversió considerant un benefici de 20 € per unitat.

El preu de venda de 120 € esta dins del marge on es podria situar el producte al mercat. Ara bé, un preu més competitiu faria el producte mes atractiu.

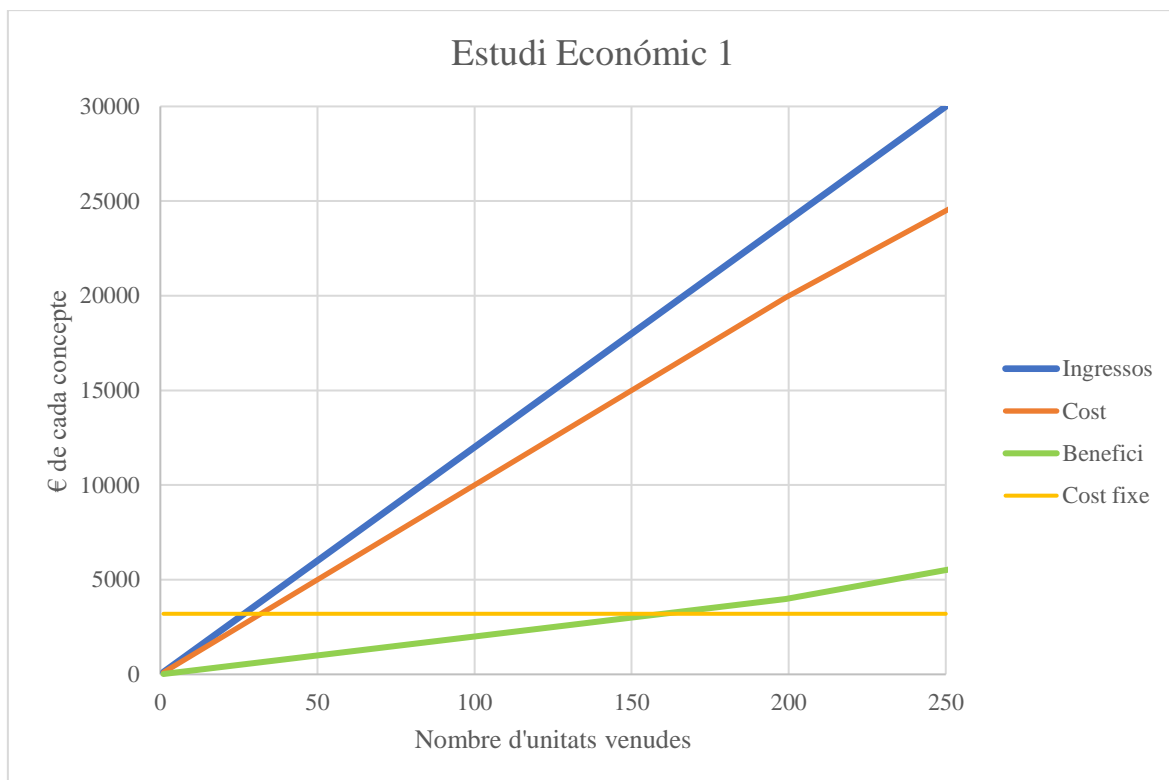


Figura 30. Primer estudi econòmic.

Segon estudi econòmic

En aquest estudi es consideraran els següents paràmetres:

- Components soldats via SMD per part del fabricant de les Plaques PCB. Es redueix el cost dels components i la mà d'obra. Obliga a realitzar un treball més ampli d'enginyeria.
- Es considera un sobre-cost de 1000 € pel que fa a enginyeria.
- Es calculen 45 € de cost per unitat. Al annex 5.6.2 es pot veure el preu dels diferents components en aquest cas.
- El cost de la mà d'obra es redueix a una quarta part donada la menor càrrega de treball.
- Es considera un preu de venda de 85€, el mateix que el *Tama Rythm Watch* vist en l'apartat 2.2.

Es vol comprovar quina haurà de ser la quantitat de prototips realitzats per tal de amortitzar la inversió realitzada en enginyeria.

En la Figura 31 es pot veure com a partir de les 130 unitats venudes ja s'ha amortitzat la inversió d'enginyeria. Es comprova que el benefici per unitat es major, de 32,5 € per unitat.

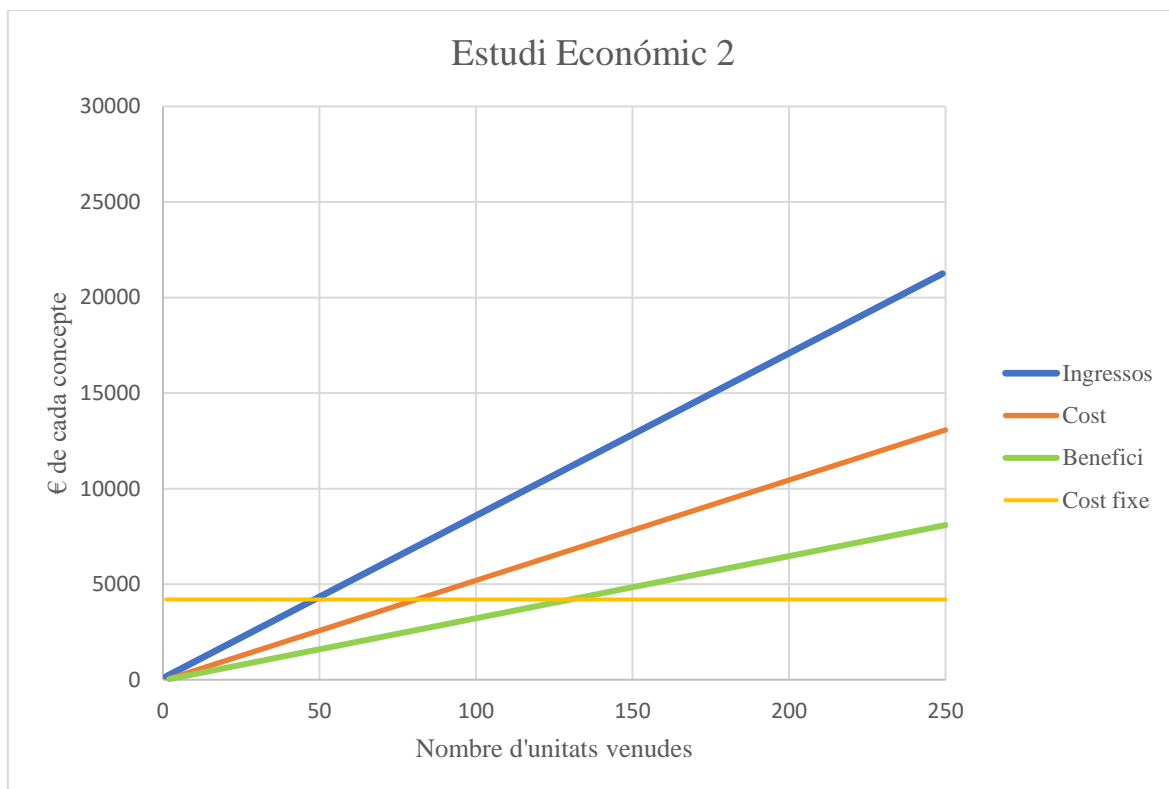


Figura 31. Segon estudi econòmic.

Conclusió estudi econòmic

Es necessari invertir recursos per a reduir el cost del producte. Caldria una inversió de uns 1000 € en enginyeria. Aquesta aniria dirigida a realitzar un disseny de PCB el qual inclogués components SMT.

La inversió permetria reduir els costos en la adquisició dels components electrònics de la placa i alhora, com la soldadura la realitzaria el fabricant de les PCB's, reduiria la mà d'obra que s'hauria d'emprar per al muntatge del metrònom.

Al tindre un producte més econòmic de manufacturar, es podria situar en un rang de preus més competitiu. Es considera el PVP de 85 € per a emparellar el metrònom al *Tama Rythm Watch*. Tot i reduir el PVP el benefici per prototip seria més ampli respecte al del primer estudi econòmic.

La viabilitat econòmica del producte va lligada a aquesta inversió, sinó el marge que es té per unitat es massa limitat. Es podria considerar un PVP més elevat, ara bé, posaria al prototip en un "price point" el qual seria menys atractiu per a un potencial consumidor.

4 Conclusions

A continuació es realitzen una sèrie de conclusions sobre la realització del prototip. També es comenta el grau d'assoliment dels objectius marcats en l'inici del projecte.

- S'ha assolit l'objectiu de implementar un metrònom digital plenament funcional. Totes les funcionalitats que s'han considerat interessants per a un potencial usuari s'han pogut integrar incorporant elements a nivell software i/o hardware.
- A part d'incorporar funcionalitats ja presents en models comercials, el metrònom digital incorpora una funcionalitat pionera en el mercat: La detecció de consigna a través de la realització de cops en un element del instrument. Un dels punts febles de la implementació d'aquesta funcionalitat és el fet que es necessari el correcte muntatge del element terminal per part de l'usuari. Si bé a nivell software s'ha intentat reduir al mínim aquest efecte, es un punt feble a considerar.
- La interacció a través dels botons i pantalla dota a l'usuari la capacitat de interaccionar amb el metrònom de forma intuïtiva. S'ha aconseguit dotar al prototipus de una robustesa suficient en quant al funcionament d'aquesta interacció pel que fa al software i l'electrònica del prototip.

Amb el prototip comercial, vindria inclòs un manual d'ús recollit al annex 5.7 el qual explica el correcte ús del prototip. Ara bé, és provable que de forma voluntària o involuntària es realitzin males praxis les quals mitjançant software s'intenta evitar que causin efecte en la dinàmica del programa del metrònom.

- El circuit electrònic dissenyat per al metrònom es plenament funcional.

La metodologia de disseny del hardware electrònic ha demostrat ser la correcta a l'hora d'implementar un prototip el qual inclogui un microcontrolador. Els passos previs al disseny i manufactura de la PCB han permès que en tot el procés de disseny només hagi fet falta la manufactura de una sola PCB.

Ara bé, amb les lliçons apreses en el procés de disseny, optaria per realitzar un nou disseny de PCB. Aquest aniria enfocat a complir les següents especificacions:

- a) Reduir la mida de la PCB per a així poder reduir la caixa del metrònom. Així es reduiria el cost i el temps de fabricació de la caixa.
- b) Realitzar un disseny incorporant components SMT, i fer que la empresa JLCPCB realitzi el muntatge i soldadura dels components electrònics. Amb aquest pas es reduiria el temps que es destina a realitzar la soldadura dels diferents components electrònics necessaris per al metrònom.
- c) Substituir dos dels potenciòmetres per resistències fixes o un altre circuit. Per al display LCD i pel convertidor DAC s'usen dos potenciòmetres per tal de generar

referències de tensió. Es va decidir utilitzar aquest mètode per tal de poder realitzar modificacions en el procés de disseny del prototip.

Per a substituir el potenciòmetre del display LCD, primerament s'hauria d'establir un proveïdor regular de displays LCD. En funció de l'LCD utilitzat, la tensió de referència que es necessita per a ajustar el contrast es diferent. Abans d'establir un valor fix de tensió s'hauria de verificar el correcte comportament dels displays LCD.

- He pogut comprovar com la versió del entorn del ORCAD de la qual disposo per haver cursat Equips Electronics, resulta incòmoda a l'hora de treballar en el desenvolupament d'un prototip.

En el cas que hagués de tornar a realitzar aquest projecte o bé per a la realització de una nova PCB seguint el comentat a la conclusió anterior, optaria per l'ús de un software professional.

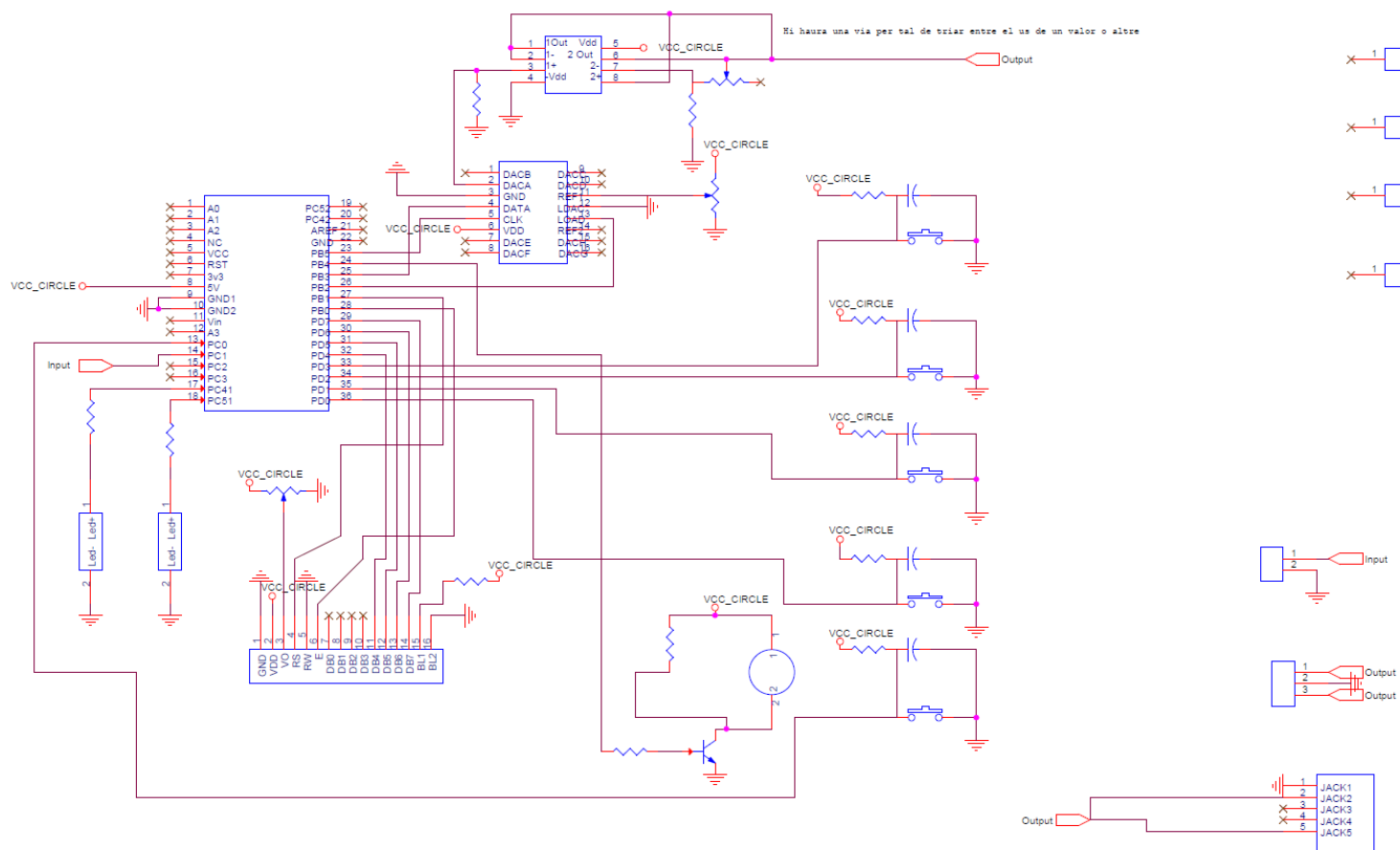
Els beneficis de utilitzar un software professional sobrepassen el cost monetari que pot suposar la compra d'una llicència. En el cas que tornés a realitzar el projecte optaria per comprar una llicència d'un mes de durada.

- Considero que la integració de la PCB a la caixa no es del tot correcta. El disseny de la capsa es adequat per a la ubicació i el conexionat amb l'exterior dels elements de la PCB. Ara bé considero deficient la integració el display LCD i els botons. Per culpa d'aquesta mancança en el posicionament d'aquests components apareixen deficiències en el comportament del prototip.
- S'ha aconseguit generar una senyal d'àudio a partir de un DAC via comunicació mitjançant sèrie. Això demostra que tot i que un microcontrolador no tingui aquest element, mitjançant altres sistemes que tingui incorporats com pot ser la SPI, es pot usar un element extern que compleixi amb els requeriments que es necessitin.
- Considero que seria econòmicament inviable desenvolupar a nivell comercial el prototipus amb l'estat actual en que es troba. Si bé del primer estudi econòmic (apartat 3.8) se'n podria despendre el contrari, considero insuficient el marge de benefici que es podria arribar a extreure del prototip en el cas que es volgués comercialitzar.

Seguint amb el comentat en l'estudi econòmic, considero que la feina de enginyeria per tal de desenvolupar una placa PCB la qual integri components SMT muntats pel fabricant, es indispensable per tal d'implementar el prototip de forma comercial. Mitjançant aquesta tècnica es redueixen molt els treballs de mà d'obra que s'han de realitzar. A més el cost dels components electrònics es veu altament reduït.

5 Annexos

5.1 Circuit elèctric



5.2 Codi font

En aquest annex s'inclou el codi corresponent al prototip. En l'apartat 5.2.1 s'inclou tot el codi font, incloent la llibreria externa. A continuació, al apartat 5.2.2 s'inclou únicament el codi corresponent a la llibreria externa.

5.2.1 Codi complet

```
/*  
  
 * File:    Principal.c  
  
 * Author:  Pau Añó Martínez  
  
 *  
  
 * Created on 8 de febrero de 2021, 11:26  
  
 */  
  
// LLIBRERIES  
  
#define F_CPU 16000000UL  
  
#include <avr/io.h>  
  
#include <avr/interrupt.h>  
  
#include <xc.h>  
  
#include <util/delay.h>  
  
// DEFINICIONS LLIBRERIA EXTERNA  
  
#define LCD_INST 0  
  
#define LCD_DATA 1  
  
#define LINEA1      0x80 // Set DDRAM at specified location fica la  
DDRAM apuntant a la linia 1  
  
#define LINEA2      0xC0 // Set DDRAM at specified location fica la  
DDRAM apuntant a la linia 2 P  
  
#define CLEAR      0x01 // Limpia el display  
  
// DEFINICIONS PROPIES  
  
#define TRAM_ZERO 600  
  
#define PRIMER_TRAM 621  
  
#define PENDENT_1 0.48  
  
#define PENDENT_1m 48  
  
#define ORDENADA_1 300
```

```
#define SEGON_TRAM 643

#define PENDENT_2 0.45

#define PENDENT_2m 45

#define ORDENADA_2 290

#define TERCER_TRAM 667

#define PENDENT_3 0.42

#define PENDENT_3m 42

#define ORDENADA_3 280

#define QUART_TRAM 692

#define PENDENT_4 0.39

#define PENDENT_4m 39

#define ORDENADA_4 270

#define CINQUE_TRAM 720

#define PENDENT_5 0.36

#define PENDENT_5m 36

#define ORDENADA_5 260

#define SISE_TRAM 750

#define PENDENT_6 0.33

#define PENDENT_6m 33

#define ORDENADA_6 250

#define SETE_TRAM 783

#define PENDENT_7 0.30

#define PENDENT_7m 30

#define ORDENADA_7 240

#define VUIITE_TRAM 818

#define PENDENT_8 0.28

#define PENDENT_8m 28

#define ORDENADA_8 230

#define NOVE_TRAM 857

#define PENDENT_9 0.25
```

```
#define PENDENT_9m 25
#define ORDENADA_9 220
#define DESE_TRAM 900
#define PENDENT_10 0.233
#define PENDENT_10m 23
#define ORDENADA_10 210
#define ONZE_TRAM 947
#define PENDENT_11 0.21
#define PENDENT_11m 21
#define ORDENADA_11 200
#define DOTZE_TRAM 1000
#define PENDENT_12 0.19
#define PENDENT_12m 19
#define ORDENADA_12 190
#define TRETZE_TRAM 1059
#define PENDENT_13 0.17
#define PENDENT_13m 17
#define ORDENADA_13 180
#define CATORZE_TRAM 1125
#define PENDENT_14 0.15
#define PENDENT_14m 15
#define ORDENADA_14 170
#define QUINZE_TRAM 1200
#define PENDENT_15 0.13
#define PENDENT_15m 13
#define ORDENADA_15 160
#define SETZE_TRAM 1286
#define PENDENT_16 0.11
#define PENDENT_16m 11
#define ORDENADA_16 150
```

```
#define DISSET_TRAM 1385

#define PENDENT_17 0.10

#define PENDENT_17m 10

#define ORDENADA_17 140

#define DIVUIT_TRAM 1500

#define PENDENT_18 0.086

#define PENDENT_18m 8

#define ORDENADA_18 130

#define DINOVE_TRAM 1636

#define PENDENT_19 0.07

#define PENDENT_19m 7

#define ORDENADA_19 120

#define VINTE_TRAM 1800

#define PENDENT_20 0.06

#define PENDENT_20m 6

#define ORDENADA_20 110

#define VINTIUNE_TRAM 2000

#define PENDENT_21 0.05

#define PENDENT_21m 5

#define ORDENADA_21 100

#define VINTIDOSE_TRAM 2250

#define PENDENT_22 0.04

#define PENDENT_22m 4

#define ORDENADA_22 90

#define VINTITRESE_TRAM 2571

#define PENDENT_23 0.03

#define PENDENT_23m 3

#define ORDENADA_23 80

#define VINTIQUATRE_TRAM 3000

#define PENDENT_24 0.02
```

```
#define PENDENT_24m 2

#define ORDENADA_24 70

#define VINTICINQUE_TRAM 3600

#define PENDENT_25 0.016

#define PENDENT_25m 1.6

#define ORDENADA_25 60

#define VINTISISE_TRAM 4500

#define PENDENT_26 0.011

#define PENDENT_26m 1.1

#define ORDENADA_26 50

#define VINTISETE_TRAM 6000

#define PENDENT_27 0.33

#define PENDENT_27m 0.6

#define ORDENADA_27 40

// Encendre SPI

#define PORT_ON_SPI PORTB|=(1<<PORTB2)

#define PORT_OFF_SPI PORTB &=~(1<<PORTB2)

// DEFINICIONS DE PROGRAMA

#define LED_ONR PORTC|=(1<<PORTC4)

#define LED_ONV PORTC|=(1<<PORTC5)

#define LED_OFFR PORTC&=~(1<<PORTC4)

#define LED_OFFV PORTC&=~(1<<PORTC5)

#define BUZZER_ON PORTB|=(1<<PORTB4)

#define BUZZER_OFF PORTB&=~(1<<PORTB4)

// VARIABLES GLOBALES

char vegades=0, valorsuperat=0;

char tap=0, cop=0;

char consignaH=0, consignaL=0;

char lectura=0;

char y=0;
```

```
char esperar=0;

short bpm=80;

short comptes=0;

short comptest1=0;

char buscant=0;

short delay=0;

char fase_delay=0;

char fase_led=0;

char consigna_anterior=0;

char ledv=0,ledr=0;

short comptes_ha_tardat=0;

short ndelay=0,a=250;

char start=0;

char canvis_disp=0;

short intermig=0;

char comprovacio=30;

char menu=0;

char premut=0;

char start_men_2=0;

// spi

char SPI1=0x00;

char dins_de_spi=0;

short val_audio=0;

// adc

char canvi_canal=0;

char canal_actual=0;

char vh_old=0;

char tap_adc=0;
```

```
char ignorar_boto=0;

char val_mul=0;

short promig;

char range_promig=50;

char promig_i=0;

char promigs[50];

char encendre=0;

// Missatges Pantalla

char presentacio1[16]="Benvinguts";

char presentacio2[16]="Metronom TFG PAU";

char menu1[16]="1: BPM";

char menu2[16]="2: Programes:";

char menu2_2[16]="Sel:   BPM:   ";

char menu3[16]="3: Configuracio";

char menu3_2[16]="TH:   OPCIO:  ";

char cadenabpm[16]="";

// valors guardats en EEPROM

short bpms[20];

char threshold;

char cad_lectura[44];

char escriure_canvi=0;

//dinàmica ampliada del programa

char programa=0;

char on=1;

// buzzer

char beep=0;

char ciclo_r=0,ciclo_v=0;
```

```
// botons dinàmica augmentada

char anterior0=0,anterior1=0,anterior2=0,anterior3=0;

char no_encendre=0;

// cadena àudio

char cad_dac[80]={

-3,  0,  0, -6,  0,  0, -6,  0, // 0-7

 2, -6,  0,  2, -6,  0,  2, -9, // 8-15

 0,  2, -9, -6,  0, -15, -6,  0, // 16-23

-21, -9,  5, -30, -18, 11, -18, -33, // 24-31

 5, -9, -33, -6, -3, -24, -9, -15, // 32-39

-18,  0, -18, -27,  8, -12, -36,  0, // 40-47

-3, -36,  2,  5, -30,  8, 11, -21, // 48-55

14, 11, -24, 11,  8, -30, 14, 20, // 56-63

-39,  8, 50, -33,  2, 56, -15, -12, // 64-71

68, 11, -42, 44, 44, -15,  5, 44}; // 72-79
```

// FUNCIONS DE INICIALITZACIÓ

```
void inicialitzacionstimers(){

//timer 0, el encarregat per overflow de fer el autotrigger del ADC quan
esta enabled.

TCCR0A=0x00;// pagina 113 del manual

TCCR0B=0x00; // se manté apagat, ficant el prescaler s'encen

TIMSK0= TIMSK0|0x01; // s'habilita interrupcio

//timer 1 el encarregat de generar senyal de sortida

TCNT1=0; // registre que conte el contador de cops/ recorda estem en
comparacio

OCR1A=249; // el registre que contindra el valor a comparar

TCCR1A=TCCR1A|0b00000000; //Control

TCCR1B=TCCR1B|0b00001000; // Control

TCCR1C=TCCR1C|0x80; // Control

TIMSK1=TIMSK1|0x02; // Interrupció habilitada
```

```
// habilita les interrupcions toca el SREG

//timer 2 el encarregat de gestionar el comptatge dels botons pa la
funció tap

TCNT2=0; // registre que conte el comptador de cops/ recorda que anem en
comparacio

OCR2A=249; // el registre que contindrà el valor a comparar

TCCR2B=TCCR2B&0xF8; //timer sopped

// quan toqui arrancar TCCR2B=TCCR2B|0x04;

TIMSK2=TIMSK2|0x02; // s'habilita interrupció

}

void inicialitzacionsADC(){

// preparació dels registres determinats per tal que nomes habilitant el
enable des de la interrupció de boto es pugui començar a llegir els
valors que entrega la piezo

ADCSRA|=(1<<ADIE); // habilitem interrupció

ADMUX|=(1<<REFS0); // elecció de avcc com a referencia 5v

ADMUX|=(1<<ADLAR); // left ajust, perdem els dos bits de menys pes

}

void inicialitzacionsbotons(){

// boto de la placa

DDRB|=(1<<DDB5); // led

//PORT D

DDRD&=~(1<<DDD0); // pd0 com a entrada

DDRD&=~(1<<DDD1); // pd1 com a entrada

DDRD&=~(1<<DDD2); // pd2 com a entrada

DDRD&=~(1<<DDD3); // pd3 com a entrada

// com que no pujem el valor del PORTD no habilitem el pull-up intern

PCMSK2|=(1<<PCINT16); // habilito la interrupció dels botons del D

PCMSK2|=(1<<PCINT17);

PCMSK2|=(1<<PCINT18);

PCMSK2|=(1<<PCINT19);
```

```
PCICR|=(1<<PCIE2);

// LEDS PORT C

DDRC|=(1<<DDC4);

DDRC|=(1<<DDC5);

//BUZZER PORT B

DDRB|=(1<<DDB4);

}

void ini_interrupcions(){

    EICRA=EICRA|0x01;// any logical change on int 0 generates an
    input request

}

void spi(){

    SPCR|=0b11011001; // control

    DDRB|=(1<<DDB3)|(1<<DDB5)|(1<<DDB2); // configuració dels pins com
    a sortida

    PORT_ON_SPI; // encesa del pin

}

// FUNCIONS DE CALCUL DE PARAMETRES

short cbpm(short comptes){

// funció per passar dels comptes generats pel timer a una consigna en
BPM

if (comptes<TRAM_ZERO){

    return 300;

}

if (comptes<PRIMER_TRAM){

    return (-PENDENT_1*(comptes-TRAM_ZERO)+ORDENADA_1);

}

if (comptes<SEGON_TRAM){

    return (-PENDENT_2*(comptes-PRIMER_TRAM)+ORDENADA_2);

}

if (comptes<TERCER_TRAM){

    return (-PENDENT_3*(comptes-SEGON_TRAM)+ORDENADA_3);

}
```

```
}  
  
    if (comptes<QUART_TRAM) {  
        return (-PENDENT_4*(comptes-TERCER_TRAM)+ORDENADA_4);  
    }  
  
    if (comptes<CINQUE_TRAM) {  
        return (-PENDENT_5*(comptes-QUART_TRAM)+ORDENADA_5);  
    }  
  
    if (comptes<SISE_TRAM) {  
        return (-PENDENT_6*(comptes-CINQUE_TRAM)+ORDENADA_6);  
    }  
  
    if (comptes<SETE_TRAM) {  
        return (-PENDENT_7*(comptes-SISE_TRAM)+ORDENADA_7);  
    }  
  
    if (comptes<VUITE_TRAM) {  
        return (-PENDENT_8*(comptes-SETE_TRAM)+ORDENADA_8);  
    }  
  
    if (comptes<NOVE_TRAM) {  
        return (-PENDENT_9*(comptes-VUITE_TRAM)+ORDENADA_9);  
    }  
  
        if (comptes<DESE_TRAM) {  
            return (-PENDENT_10*(comptes-NOVE_TRAM)+ORDENADA_10);  
        }  
  
        if (comptes<ONZE_TRAM) {  
            return (-PENDENT_11*(comptes-DESE_TRAM)+ORDENADA_11);  
        }  
  
        if (comptes<DOTZE_TRAM) {  
            return (-PENDENT_12*(comptes-ONZE_TRAM)+ORDENADA_12);  
        }  
  
        if (comptes<TRETZE_TRAM) {  
            return (-PENDENT_13*(comptes-DOTZE_TRAM)+ORDENADA_13);  
        }
```

```
}  
  
    if (comptes<CATORZE_TRAM) {  
return (-PENDENT_14*(comptes-TRETZE_TRAM)+ORDENADA_14);  
    }  
  
    if (comptes<QUINZE_TRAM) {  
return (-PENDENT_15*(comptes-CATORZE_TRAM)+ORDENADA_15);  
    }  
  
    if (comptes<SETZE_TRAM) {  
return (-PENDENT_16*(comptes-QUINZE_TRAM)+ORDENADA_16);  
    }  
  
    if (comptes<DISSET_TRAM) {  
return (-PENDENT_17*(comptes-SETZE_TRAM)+ORDENADA_17);  
    }  
  
    if (comptes<DIVUIT_TRAM) {  
return (-PENDENT_18*(comptes-DISSET_TRAM)+ORDENADA_18);  
    }  
  
    if (comptes<DINOVE_TRAM) {  
return (-PENDENT_19*(comptes-DIVUIT_TRAM)+ORDENADA_19);  
    }  
  
    if (comptes<VINTE_TRAM) {  
return (-PENDENT_20*(comptes-DINOVE_TRAM)+ORDENADA_20);  
    }  
  
    if (comptes<VINTEIUNE_TRAM) {  
return (-PENDENT_21*(comptes-VINTE_TRAM)+ORDENADA_21);  
    }  
  
    if (comptes<VINTIDOSE_TRAM) {  
return (-PENDENT_22*(comptes-VINTEIUNE_TRAM)+ORDENADA_22);  
    }  
  
    if (comptes<VINTITRESE_TRAM) {  
return (-PENDENT_23*(comptes-VINTIDOSE_TRAM)+ORDENADA_23);  
    }
```

```
}  
  
    if (comptes<VINTIQUATRE_TRAM) {  
        return (-PENDENT_24*(comptes-VINTITRESE_TRAM)+ORDENADA_24);  
    }  
  
    if (comptes<VINTICINQUE_TRAM) {  
        return (-PENDENT_25*(comptes-VINTIQUATRE_TRAM)+ORDENADA_25);  
    }  
  
    if (comptes<VINTISISE_TRAM) {  
        return (-PENDENT_26*(comptes-VINTICINQUE_TRAM)+ORDENADA_26);  
    }  
  
    if (comptes<VINTISETETE_TRAM) {  
        return (-PENDENT_27*(comptes-VINTISISE_TRAM)+ORDENADA_27);  
    }  
  
    if (comptes>=6000) {  
        return 30;  
    }  
  
    return 0;  
}  
  
short bpmcomptes(short bpm) {  
    // Càlcul dels comptes per al timer 2 a partir d'una consigna  
    if (bpm>ORDENADA_2) {  
        return (PRIMER_TRAM-100*(bpm-ORDENADA_2)/(PENDENT_1m));  
    }  
    if (bpm>ORDENADA_3) {  
        return (SEGON_TRAM-100*(bpm-ORDENADA_3)/(PENDENT_2m));  
    }  
    if (bpm>ORDENADA_4) {  
        return (TERCER_TRAM-100*(bpm-ORDENADA_4)/(PENDENT_3m));  
    }  
    if (bpm>ORDENADA_5) {  
        return (QUART_TRAM-100*(bpm-ORDENADA_5)/(PENDENT_4m));  
    }  
    if (bpm>ORDENADA_6) {
```

```
return (CINQUE_TRAM-100*(bpm-ORDENADA_6)/(PENDENT_5m));  
  
} if (bpm>ORDENADA_7){  
  
return (SISE_TRAM-100*(bpm-ORDENADA_7)/(PENDENT_6m));  
  
} if (bpm>ORDENADA_8){  
  
return (SETE_TRAM-100*(bpm-ORDENADA_8)/(PENDENT_7m));  
  
} if (bpm>ORDENADA_9){  
  
return (VUITE_TRAM-100*(bpm-ORDENADA_9)/(PENDENT_8m));  
  
} if (bpm>ORDENADA_10){  
  
return (NOVE_TRAM-100*(bpm-ORDENADA_10)/(PENDENT_9m));  
  
} if (bpm>ORDENADA_11){  
  
return (DESE_TRAM-100*(bpm-ORDENADA_11)/(PENDENT_10m));  
  
} if (bpm>ORDENADA_12){  
  
return (ONZE_TRAM-100*(bpm-ORDENADA_12)/(PENDENT_11m));  
  
} if (bpm>ORDENADA_13){  
  
return (DOTZE_TRAM-100*(bpm-ORDENADA_13)/(PENDENT_12m));  
  
} if (bpm>ORDENADA_14){  
  
return (TRETZE_TRAM-100*(bpm-ORDENADA_14)/(PENDENT_13m));  
  
} if (bpm>ORDENADA_15){  
  
return (CATORZE_TRAM-100*(bpm-ORDENADA_15)/(PENDENT_14m));  
  
} if (bpm>ORDENADA_16){  
  
return (QUINZE_TRAM-100*(bpm-ORDENADA_16)/(PENDENT_15m));  
  
} if (bpm>ORDENADA_17){  
  
return (SETZE_TRAM-(bpm-ORDENADA_17)/(PENDENT_16m));  
  
} if (bpm>ORDENADA_18){  
  
return (DISSET_TRAM-100*(bpm-ORDENADA_18)/(PENDENT_17m));  
  
} if (bpm>ORDENADA_19){  
  
return (DIVUIT_TRAM-100*(bpm-ORDENADA_19)/(PENDENT_18m));  
  
} if (bpm>ORDENADA_20){  
  
return (DINOVE_TRAM-100*(bpm-ORDENADA_20)/(PENDENT_19m));  
  
} if (bpm>ORDENADA_21){
```

```
    return (VINTE_TRAM-100*(bpm-ORDENADA_21)/(PENDENT_20m));
} if (bpm>ORDENADA_22){
    return (VINTIUNE_TRAM-100*(bpm-ORDENADA_22)/(PENDENT_21m));
} if (bpm>ORDENADA_23){
    return (VINTIDOSE_TRAM-100*(bpm-ORDENADA_23)/(PENDENT_22m));
} if (bpm>ORDENADA_24){
    return (VINTITRESE_TRAM-100*(bpm-ORDENADA_24)/(PENDENT_23m));
} if (bpm>ORDENADA_25){
    return (VINTIQUATRE_TRAM-100*(bpm-ORDENADA_25)/(PENDENT_24m));
} if (bpm>ORDENADA_26){
    return (VINTICINQUE_TRAM-100*(bpm-ORDENADA_26)/(PENDENT_25m));
} if (bpm>ORDENADA_27){
    return (VINTISISE_TRAM-100*(bpm-ORDENADA_27)/(PENDENT_26m));
} if (bpm>30)
    return (VINTISETE_TRAM-100*(bpm-30)/PENDENT_27m);
if (bpm==30){
    return (6000);
}
return 0;
}

void bpmconv(short bpm,char cad[],char index){
// passa una consigna a una taula en el índex que s'indica en codificació
//ASCII

    char centenes,desenes,unitats;

    centenes = bpm/100;

    desenes = (bpm-centenes*100)/10;

    unitats = (bpm-centenes*100-desenes*10);

    cad[index]=centenes+0x30;

    cad[index+1]=desenes+0x30;

    cad[index+2]=unitats+0x30;
```



```
    SPDR=cad_dac[val_audio]; // camps de generació de so
    val_audio++;
}
else{
    dins_de_spi=0;
    PORT_OFF_SPI; // fem el canvi d'estat necessari per al DAC
    _delay_ms(1); // Valor requerit pel DS
    PORT_ON_SPI;
    if (val_audio!=50){ //Hem d'enviar mes caràcters de dades?
        SPDR=SPI1; // Començem la trama de 2 bytes
    }
}
}
ISR (ADC_vect){ // Es dispara cada cop que ha acabat la conversió del ADC
    char vh,vl;
    vl=ADCL;
    vh=ADCH;
    if (canal_actual==1){
        if (vegades<range_promig-2){// si no tenim un mínim de valors no
            podem generar el promig
            promigs[vegades]=vh;
            vegades++;
        }
    }
    else{
        calculpromig();

        if ((vh>promig+threshold)&&(esperar==0)){ // es el at multiplicat
per mil, les peticions del adc van a 60 HZ
            esperar=1;// farà que les deu següents s'ignorin
            if (tap_adc==0){
                tap_adc=1;
            }
        }
    }
}
```

```
        pararTimer2(); // fem un reset al timer 1, considerem el
primer cop com a partida des de la qual generarem la consigna

        encendreTimer2(); //

        tap=1;

        cop=1;

    }

    else{

        cop++;

    }

}

renovarpromig(vh);

}

}

else{ // si el canal es el zero estarem mirant la senyal que ens
vingui del botó

    if (vh==0){

        if (ignorar_boto==0){

            premut=0;

            menu=(menu+1)%3; // canvi del valor del menú

            ignorar_boto=1; // farà que s'ignorin les següents
            canvis_disp=1;

        }

    }

    else{

        ignorar_boto=0;

    }

}

}

ISR(PCINT2_vect){ // Vector d'interrupció quan hi ha un canvi en els
botons 1 a 4

    if(!(PIND&(1<<PIND3))){ // Boto 4

        anterior3=1;
```

```
}  
  
else{  
  
    if (anterior3==1){  
        anterior3=0;  
  
    if ((menu==0)&&(fase_delay==0)){ // cas menú=0  
        if (tap==0){ // iniciar el tap  
            pararTimer1();  
            encendreTimer2();  
  
            tap=1;  
            cop=1;  
            canvi_canal=1;  
            vegades=0;  
  
            if (!(PIND & (1 << PIND2))){  
                no_encendre=1;  
            }  
            else{  
                no_encendre=0;  
            }  
        }  
        else{  
            LED_ONV;  
            LED_ONR;  
            cop++; // augmentar el valor de COP  
        }  
    }  
  
    if (menu==1){ // cas menu=1  
        // save consigna  
        premut++;  
  
        if (premut==3){ // s'ha de prémer tres vegades per a guardar en  
EEPROM
```

```
        premut=0;

        bpms [programa]=bpm;

        escriptura_EEPROM(0, bpm, programa, 0);

        canvis_disp=1;

    }

}

if (menu==2){ // cas menu=2

    escriptura_EEPROM(0, threshold, 40, 1);

    escriptura_EEPROM(0, on, 41, 1);

}

}

}

if (!(PIND & (1 << PIND2))) { // boto 3

    anterior2=1;

}

else{

    if ((anterior2==1) && (no_encendre==0)) {

        anterior2=0;

        // start stop estant en menu 0 i 2

        if ((menu==0)) { // menu=0

            if (start==0) {

                start=1;

                encendreTimer1();

            }

            else{

                start=0;

                pararTimer1();

            }

        }

    }

    if (menu==1) { // encendre el metronom en la consigna del programa
```

```
    if ((start==0) || (start_men_2==0)) {  
        start_men_2=1;  
        bpm=bpms[programa];  
        ndelay=bpmcomptes(bpm)/3;  
        start=1;  
        encendreTimer1();  
    }  
    else{  
        start=0;  
        pararTimer1();  
    }  
}  
  
if (menu==2){ // cas menu 2  
    // opcio buzzer  
    canvis_disp=1;  
    on=(on+1)%2;  
}  
}  
  
else {  
    no_encendre=0;  
    anterior2=0;  
}  
}  
  
if (!(PIND & (1 << PIND1))) { boto 2  
anterior1=1;  
}  
else{  
if ((anterior1==1) && (premut==0)) {  
anterior1=0;  
if (menu==0) { //cas menu=0
```

```
// pujem la consigna una unitat
    if (bpm!=300){
        bpm++;
        ndelay=bpmcomptes (bpm) /3;
        escriptura_EEPROM(0,bpm,21,0);
        canvis_disp=1;
    }
}
if (menu==1){ //cas menu=1
// programa+1
    start_men_2=0;
    programa=(programa+1)%20;
    canvis_disp=1;
}
if (menu==2){ // cas menu=2
    threshold=(threshold+1)%25;
    canvis_disp=1;
}
}
}
if (!(PIND & (1<<PIND0))) { // boto 1
    anterior0=1;
}
else{
    if (anterior0==1){
        anterior0=0;
        premut=0;
        if (menu==0){ // cas menu=0
            // baixem la consigna una vegada
            if (bpm!=30){
```

```
bpm--;

ndelay=bpmcomptes (bpm) /3;

escriptura_EEPROM(0,bpm,21,0);

canvis_disp=1;
}
}

if (menu==1){ // cas menu=1

start_men_2=0;

if (programa==0){

programa=19;

}

else{

programa--;

}

canvis_disp=1;

}

if (menu==2){ //cas menu=2

// threshold-1

if (threshold==0){

threshold=25;

}

else{

threshold--;

}

canvis_disp=1;

}

}

}
```

```
ISR (TIMER0_OVF_vect){ // interrupció que salta per overflow del TIMER0

    if ((canvi_canal==1)&&(~ADCSRA&0x40)){ //si no esta fent conversió i
    se vol fer un canvi de canal

        canvi_canal=0;

        if (canal_actual==0){

            canal_actual=1;

            canviaracanallADC();

        }

        else{

            canal_actual=0;

            canviaracanal0ADC();

        }

    }

    if (~ADCSRA&0x40){ // si no esta fent conversio

        ADCSRA|=(1<<ADSC); // peticio de adc

        if (esperar>0){

            esperar=(esperar+1)%15;// ficarà esperar a 0 després de varies

            //necessari per a la RSI del ADC

        }

    }

}
```

```
ISR (TIMER2_COMPA_vect){ // Es dispara quan el comptador arriba al valor
assignat

    TCNT2=0;

    comptes++;

    if ((tap==1)&&(fase_delay==0)){

        if (cop==4){ // si cop es quatre analitzem el temps que ha passat

            consigna_anterior=1;

            delay=comptes/3; // comptes per al display actualitzat

            comptes_ha_tardat=comptes;

        }

    }

}
```

```
tap_adc=0;

bpm=cbpm(comptes); // càlcul consigna

escriure_canvi=1;

canvis_disp=1;

canvi_canal=1;

vh_old=0;

LED_OFFV;

LED_OFFR;

if (no_encendre==0){ // mirem si s'ha d'encendre o no el metrònom
start=1;

fase_delay=1;

}

else

{

start=0;

tap=0;

}

}

if (comptes==6000){ // cas que comptes sigui 30 BPM

bpm=cbpm(comptes);

canvis_disp=1;

delay=comptes/3;

buscant=0;

comptes_ha_tardat=comptes;

canvi_canal=1;

vh_old=0;

LED_OFFV;

LED_OFFR;

if (no_encendre==0){

start=1;
```

```
        fase_delay=1;
    }

    else{
        start=0;
        tap=0;
    }
}

}

if (fase_delay==1){ //fase prèvia a la encesa del metrònom
    // s'espera el temps corresponent a un altre cop
    if (comptes==delay+comptes_ha_tardat){
        menu=0;
        comptes=0;
        fase_delay=0;
        ndelay=delay;
        pararTimer2 ();
        tap=0;
        if (start==1){
            encendreTimer1(); // s'encén el metrònom en si
        }
    }
}
}
```

```
ISR (TIMER1_COMPA_vect){ // Es dispara cada cop que COMPA del Timer1
arriba al valor establert

//Aquí es genera la senyal de sortida, d'àudio, leds i buzzer

    comptest1++;

    if
((comptest1==ndelay) || (comptest1==(ndelay+ndelay)) || (comptest1==(ndelay+n
delay+ndelay))){

        ledv=0; //activem ledv mes avall
    }

    if (comptest1==(ndelay+ndelay+ndelay+ndelay)){

        comptest1=0;

        ledr=0; // activem ledr mes avall
    }

    if (on==1){

        if (ledv==0){

            ciclo_v=1;

            beep=0;

        }

        if (ledr==0){

            ciclo_r=1;

        }

    }

    if (ciclo_v==1) // paràmetres per al so del verd
    {

        if (beep<3){ // 3 es una cota marcada

            BUZZER_ON;

            beep++;

        }

        else{

            beep=(beep+1)%4; // 4 es una cota marcada

            BUZZER_OFF;
```

```
    }

    //Amb els valors de 3 i 4 Estem establint un PWM de D=0.75.
    La freqüència es de 250 hz

}

if (ciclo_r==1) //paràmetres per al so del vermell
{
    if (beep<1){
        BUZZER_ON;
        beep++;
    }
    else{
        beep=(beep+1)%2;
        BUZZER_OFF;
    }

    //Amb els valors de 1 i 2 Estem establint un PWM de D=0.5. La
    freqüència es de 500 hz

}

}

else {
    BUZZER_OFF;
}

// Zona display LED i envio del primer element de la trama de la
SPI

if (ledr==0){
    spi();
    SPDR=SPI1;
    val_audio=0;
    LED_ONR;
}

if (ledr<25){
    ledr++;
}

}
```

```
        else{
            LED_OFFR;

            ciclo_r=0;
        }
    if (ledv==0){
        spi();

        SPDR=SPI1;

        val_audio=0;

        LED_ONV;
    }

    if (ledv<25){
        ledv++;
    }

    else{
        LED_OFFV;

        ciclo_v=0;
    }
}

// FUNCIONS AUXILIARS RSI
void canviaracanal1ADC(){
    ADMUX|=0x01;
}

void canviaracanal0ADC(){
    ADMUX&=~0x01;

    vegades=0;
}

void encendreTimer0(){
    // hem d'encendre el timer, habilitar la petició de interrupció per
    part del ADC

    TCNT0=0x00; // Counter a 0
}
```

```
TCCR0B|=0x05; // Prescaler del timer, l'estem encenent

ADCSRA|=(1<<ADEN); // Enable ADC

ADCSRA|=(1<<ADSC); // petició de conversió del ADC

}

void pararTimer0() {

    // hem de parar el timer, deshabilitar que el ADC ens pugui causar
    interrupcio per conversió feta

    TCNT0=0x00; // Counter del timer a 0

    TCCR0B&=~0x05;

    ADCSRA&=~(1<<ADEN); // Turning off ADC

}

void encendreTimer1() {

    ledr=0; // encenem LED vermell

    ledv=1;

    beep=0;

    comptest1=0; // pas per la interrupció

    TCNT1=0x0000; // registre que conte el comptador del Timer

    TCCR1B=TCCR1B|0x03; // Prescaler a 64 interrupció a 1 kHz

}

void pararTimer1() {

    comptest1=0;

    TCNT1=0; // registre que conte el comptador del Timer

    TCCR1B=TCCR1B&0xF8; //parem el source de clock

    ledv=0;

    ledr=0;

    LED_OFFV;

    LED_OFFR;

}

void encendreTimer2() {

    comptes=0;

    TCNT2=0x00; // registre que conte els comptes del timer
```

```
TCCR2B=TCCR2B|0x04; // Prescaler a 64 interrupcio a 1 kHz

LED_ONV;

LED_ONR;

}

void pararTimer2 () {

    comptes=0;

    TCNT2=0;

    TCCR2B=TCCR2B&0xF8; //parem el source de clock

}

//

// PART LLIBRERIA EXTERNA

void lcd_start()

{ // pols de 0 1 al enable, PB0 HIGH to low fa el enable

    PORTB &=~ (1<<PORTB0);

    _delay_us(1); // mínim de 1 us per a cada E cicle

    PORTB |= (1<<PORTB0);

}

void lcd_write(uint8_t type, uint8_t inst_data)

{ // type o be es LCD_INST= 0; o bé LCD_DATA=1

    // data es per transmetre coses a escriure mentre que inst es per a

    situar la linea o limpijar la pantalla

    if (type){

        // LCD_DATA

        PORTB |= (1<<PORTB1); // a 1 el portb 1, es el RS WRITE DATA

TO RAM

    }

    else{

        //LCD_INST DO NOT WRITE DATA TO RAM

        PORTB &=~ (1<<PORTB1);

    }

    PORTD = (PIND & 0x0F) | (inst_data & 0xF0); //transmissio dels

datos de mes pes
```

```
    lcd_start(); // enable carreguem data

    PORTD = (PIND & 0x0F) | ((inst_data<<4) & 0xF0); // transmissio
dels datos de menys pes

    lcd_start(); //enable carreguem data

    _delay_ms(2);

//RAM es la que fa el display a freqüència de oscil·lació
}

void lcd_init()
{
    DDRB |= (1<<DDB0) | (1<<DDB1);

    DDRD |= 0xF0;

    PORTB &=~ (1<<PORTB0);

    PORTD = (PIND & 0x0F) | 0x20; //0X2X SET interface data length4bit,
1 linia, tipo F

    // crida a enable

    lcd_start();

    _delay_us(100);

// parametres de inicialitzacio, en cada write es fiquen els valors i
s'activa el enable

    lcd_write(LCD_INST, 0x2C); // DDRAM address set. Valor C

    lcd_write(LCD_INST, 0x08); // set display, cursor and blinking off

    lcd_write(LCD_INST, 0x01); // clear display

    lcd_write(LCD_INST, 0x06); //assign cursor mmoving directiont
and make shift of entire display enable

    lcd_write(LCD_INST, 0x0C); // set cursor but blinking off

}

void lcd_print(char *s)
{
    while(*s)
    {
```

```
        lcd_write(LCD_DATA, *s);

        s++;

    }

}

// EEPROM

void escriptura_EEPROM(char adreca_inicial, short data, char index, char
tipo){

    char data1,data0;

    char adreca;

    if (tipo==0){ //si hem de guardar un valor de bpm de programa

        data1=((data&0xFF00)>>8)&0xFF;

        data0= data&0x00FF;

        adreca=adreca_inicial+(index<<1);

        // deshabilitar interrupcions

        while (EECR &(1<<EEPE)){

        }

        cli();// se deshabiliten les interrupcions

        EEARH=0;

        EEARL=adreca;

        EEDR=data1;

        // escriure un 1 a EEMPE

        EECR|=(1<<EEMPE);

        // comença la escriptura

        EECR|=(1<<EEPE);

        // Procedim a escriure el segon Byte

        while (EECR &(1<<EEPE)){ // s'espera que s'hagi escrit correctament

        }

        EEARL=adreca+1;

        EEDR=data0;

        // escriure un 1 a eempe
```

```
EECR|=(1<<EEMPE);

// comença la escriptura

EECR|=(1<<EEPE);

while (EECR &(1<<EEPE)){

}

// ja estaria la escriptura de tipo 0

}

if (tipo==1){ // el ON/OFF del buzzer i el valor del threshold

cli();

EEARH=0;

EEARL=adreca_inicial+index;

EEDR=(char)data&0x00FF;

EECR|=(1<<EEMPE);

// comença la escriptura

EECR|=(1<<EEPE);

while (EECR &(1<<EEPE)){ // s'espera que s'hagi escrit correctament

}

}

sei();// s'habiliten interrupcions. Es rellevant que no es
produeixin interrupcions durant la escriptura a la EEPROM

}

void lectura_EEPROM(char adreca_inicial){

char i=0;

cli();

EEARH=0;

while (i<44){ // 20 consignes de 2 bytes, mes 1 byte del on/off, mes 1
byte del threshold, mes 2 bytes de la última consigna en la que s'ha
treballat

while (EECR & (1<<EEPE)){

}

}
```

```
// adreça inicial

EEARL=adreca_inicial+i;

EECR|=(1<<EERE);

cad_lectura[i]=EEDR;

i++;

}

Sei();

}

void actualitzar_data(){

// actualitzem les dades a partir de la lectura de la EEPROM

char i=0;

char y=0;

while(y<20){

bpm[y]=((cad_lectura[i]<<8)&0xFF00)+(cad_lectura[i+1]&0x00FF);

y++;

i=i+2;

}

threshold=cad_lectura[40];

on=cad_lectura[41];

if ((cad_lectura[43]==0xFF)&&(cad_lectura[42]==0xFF)){

bpm=80; // valor de seguretat en el cas que hagi hagut un problema

en guardar

}

else{

bpm=((cad_lectura[42]<<8)&0xFF00)+cad_lectura[43];

}

}

// MAIN

int main(){
```

```
lectura_EEPROM(0);

actualitzar_data();

inicialitzacionstimers();

inicialitzacionsADC();

inicialitzacionsbotons();

spi();

ndelay=bpmcomptes(bpm)/3;

asm("Sei"); // activem globalment les interrupcions

lcd_init(); // Missatges de presentació del display

lcd_write(LCD_INST, CLEAR);

lcd_write(LCD_INST, LINEA1);

lcd_print(presentacio1);

lcd_write(LCD_INST, LINEA2);

lcd_print(presentacio2);

_delay_ms(3000);

encendreTimer0(); // comença a detectar-se els canvis en boto 0

canvis_disp=1;

while (1){

    if (canvis_disp==1){

        canvis_disp=0;

        if (menu==0){

            lcd_write(LCD_INST, CLEAR);

            lcd_write(LCD_INST, LINEA1);

            lcd_print(menu1);

            lcd_write(LCD_INST, LINEA2);

            bpmconv(bpm, cadenabpm, 0);

            lcd_print(cadenabpm);

            if (escriure_canvi){

                escriptura_EEPROM(0,bpm,21,0); // es situa aquí per a no
                causar retard en el timer

                escriure_canvi=0;

            }

        }

    }

}
```

```
    }  
    }  
    if (menu==1){  
        lcd_write(LCD_INST, CLEAR);  
        lcd_write(LCD_INST, LINEA1);  
        lcd_print(menu2);  
        lcd_write(LCD_INST, LINEA2);  
        bpmconv(bpms[programa],menu2_2,13);  
        char_caracter(programa,menu2_2,5);  
        lcd_print(menu2_2);  
    }  
    if (menu==2){  
        lcd_write(LCD_INST, CLEAR);  
        lcd_write(LCD_INST, LINEA1);  
        lcd_print(menu3);  
        lcd_write(LCD_INST, LINEA2);  
        char_caracter(threshold,menu3_2,3);  
        if (on==1){  
            menu3_2[13]=' ';  
            menu3_2[14]='0';  
            menu3_2[15]='\n';  
        }  
        else{ // on = 0  
            menu3_2[13]='0';  
            menu3_2[14]='f';  
            menu3_2[15]='f';  
        }  
        lcd_print(menu3_2);  
    }  
}
```

```
        else{  
            asm("NOP");  
        }  
    }  
}
```

5.2.2 Llibreria

En aquest apartat s'inclou **únicament** el codi corresponent a la llibreria emprada.

```
void lcd_start()  
{ // pols de 0 1 al enable PB0 HIGH to low fa el enable  
    PORTB &=~ (1<<PORTB0);  
    _delay_us(1); // minim de 1 us per a cada E cicle  
    PORTB |= (1<<PORTB0);  
}  
  
void lcd_write(uint8_t type, uint8_t inst_data)  
{  
    if (type){  
        PORTB |= (1<<PORTB1);  
    }  
    else{  
        PORTB &=~ (1<<PORTB1);  
    }  
    PORTD = (PIND & 0x0F) | (inst_data & 0xF0);  
    lcd_start(); // enable carreguem data  
    PORTD = (PIND & 0x0F) | ((inst_data<<4) & 0xF0);  
    lcd_start();  
    _delay_ms(2);  
}  
  
void lcd_init()
```

```
{  
  
    DDRB |= (1<<DDB0) | (1<<DDB1);  
  
    DDRD |= 0xF0;  
  
    PORTB &=~ (1<<PORTB0);  
  
    PORTD = (PIND & 0x0F) | 0x30;  
  
    _delay_ms(100);  
  
    lcd_start();  
  
    _delay_ms(5);  
  
    lcd_start();  
  
    _delay_us(100);  
  
    lcd_start();  
  
    _delay_us(100);  
  
    PORTD = (PIND & 0x0F) | 0x20;  
  
    lcd_start();  
  
    _delay_us(100);  
  
    lcd_write(LCD_INST, 0x2C);  
  
    lcd_write(LCD_INST, 0x08);  
  
    lcd_write(LCD_INST, 0x01);  
  
    lcd_write(LCD_INST, 0x06);  
  
    lcd_write(LCD_INST, 0x0C);  
  
}  
  
void lcd_print(char *s)  
{  
  
    while(*s)  
    {  
        lcd_write(LCD_DATA, *s);  
  
        s++;  
  
    }  
  
}
```

5.3 Tria del microcontrolador

Després de realitzar una cerca a través dels diversos fabricants en considero dos per a la tria del microcontrolador:

- Microchip
- StMicroElectronics

Aquest dos esdevenen els escollits per la documentació que ofereixen i la gran quantitat de kits de desenvolupament que ofereixen amb diversos perifèrics integrats. El cost també es un fet important, ofereixen plaques amb el microcontrolador enquestat per menys de 10 €.

Primera Tria

En la primera tria es considera el microcontrolador STM32L100RCT6 fabricat per StMicroelectronics, el qual esta enquestat en un kit de desenvolupament DISCOVERY. A l'apartat 5.3.1 hi ha la fitxa tècnica completa del kit de desenvolupament.

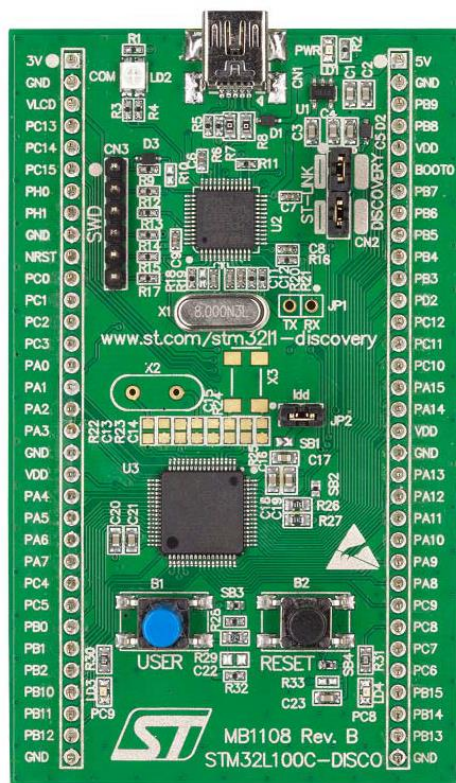


Figura 32 Kit de desenvolupament Discovery

Les característiques que fan que el consideri són les següents:

- Arquitectura de 32 bits.
- Cost reduït.
- Driver dedicat a displays LCD.

- Gran nombre de timers.
- Presencia de DAC.
- Memòries no volàtils que permetrien la implementació del codi amb molt marge.

Previ a la realització de la compra es decideix realitzar un apropament al entorn de desenvolupament, en concret instal·lo el STCubeIde en el ordinador personal. El programa no es mostra funcional. Decideixo passar a buscar altres IDE's compatibles amb el dispositiu amb qüestió.

El segon IDE instal·lat es el Kiel μ Vsion. Al familiaritzar-me amb el programa decideixo realitzar un canvi en la tria del microcontrolador. Prefereixo realitzar el programari en un entorn que sàpiga que es totalment funcional.

Conclusió

Es descarta l'ús del microcontrolador STM32L100RC per aspectes relacionats amb el entorn de desenvolupament associat aquest, no pas per característiques tècniques.

Segona Tria

Un cop descartat el microcontrolador STM32L100RC, es considera el ATTINY416. Veure annex 5.3.2 per a veure la fitxa tècnica.

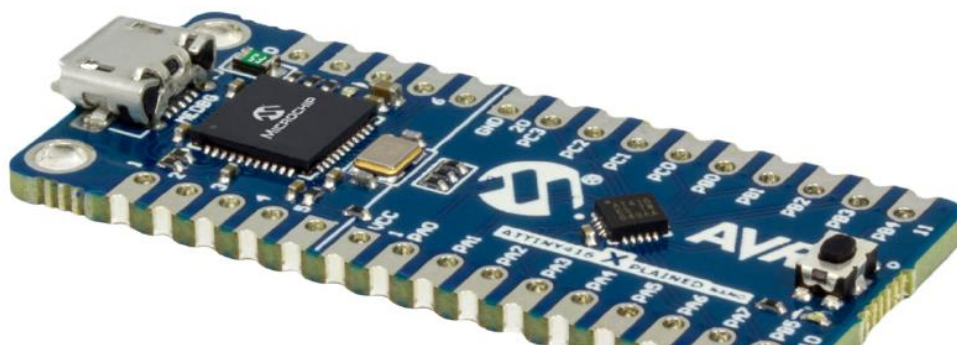


Figura 33: ATtiny416 Xplained Nano

Les característiques que fan que es consideri son les següents:

- Preu reduït.
- Mida del kit molt reduïda.

- Fabricat per Microchip, el entorn de desenvolupament pot ser el MPLABX.

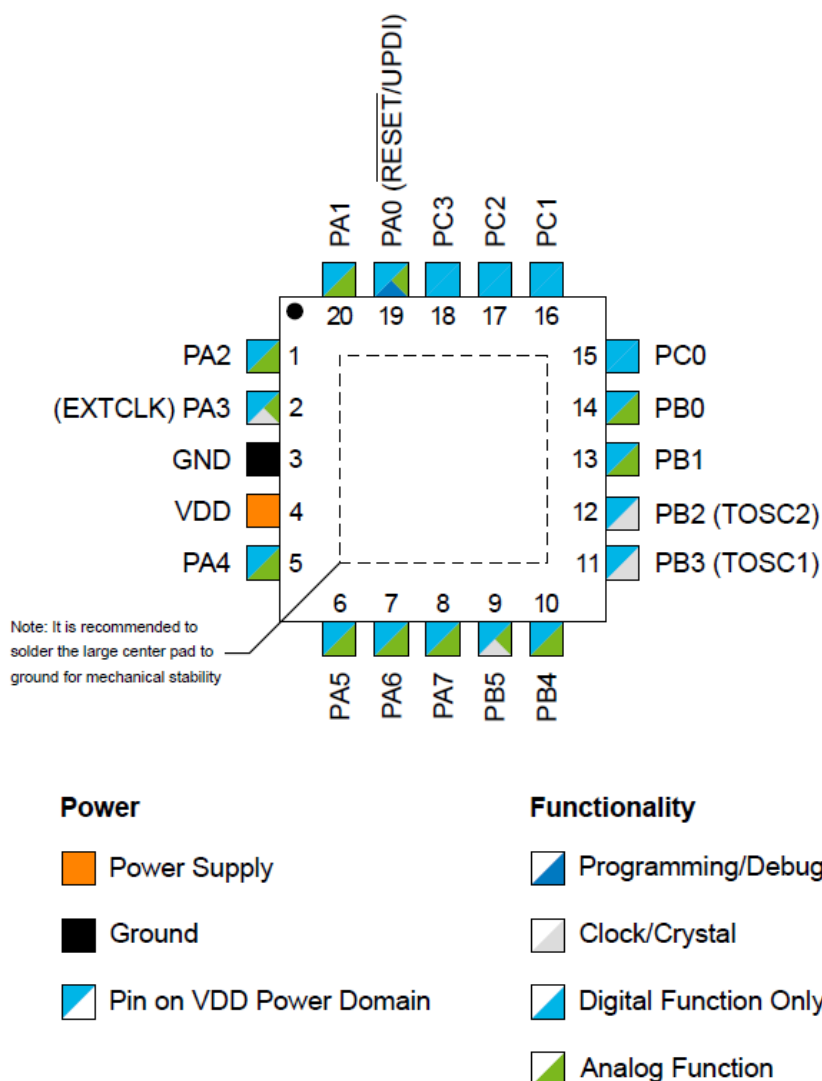


Figura 34. "Pinout" del ATtiny416

Tal i com es pot veure en la figura anterior, extreta del manual de Microchip el encapsulat presenta una quantitat baixa de pins d'entrades i sortides. Revisant les especificacions que es vol tindre en el producte, es descarta aquesta opció.

Conclusió

Es descarta l'ús del microcontrolador ATTINY416 per aspectes tècnics. Considero insuficient el nombre de pins usables en entrades i sortides, hi hauran limitacions a l'hora d'implementar tant les funcions que es desitgen tindre com d'altres que es vulguin incorporar durant el transcurs de la realització del treball.

Tercera tria – i definitiva

Seguint amb la línia de la segona tria, es continua buscant un microcontrolador compatible amb les característiques del projecte que formi part de la família de Microchip.

S'escull el ATmega328P incrustat dins de un kit de desenvolupament Xplained. Veure apartat 5.3.3 per veure'n la fitxa tècnica.

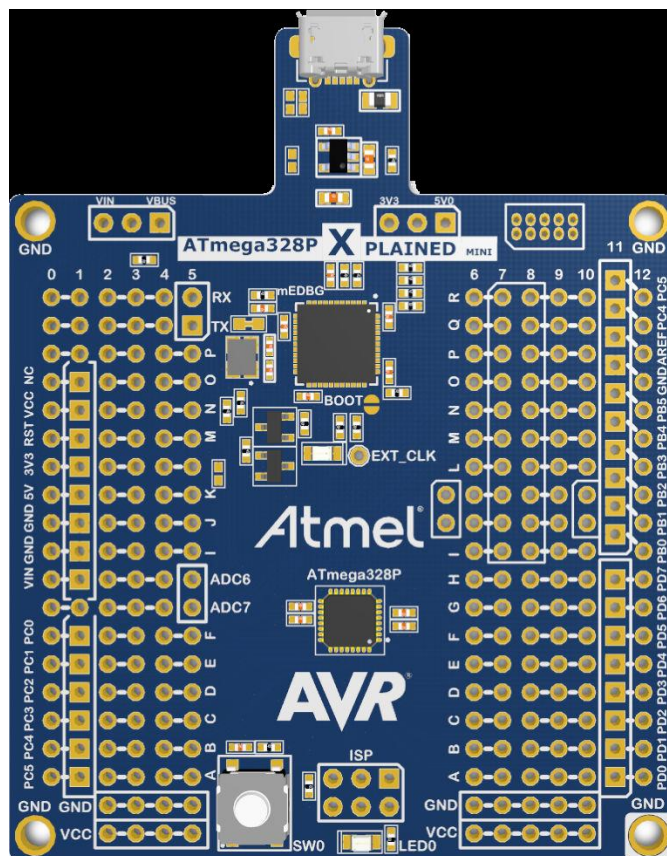


Figura 35. Atmega 328P-Xplained Mini

Les característiques que fan que es consideri son les següents:

- Preu reduït.
- Mida del kit continguda.
- Fabricat per Microchip. L'entorn de desenvolupament pot ser el MPLABX, programa amb el qual es disposa d'experiència de cursos passats.
- Suficients pins per tal de desenvolupar les funcions del projecte.

Es decideix avançar en la compra del kit de desenvolupament Xplained amb el microcontrolador ATmega328P.

5.3.1 Fitxa tècnica: STM32L100DISCOVERY

Fitxa tècnica: STM32L100DISCOVERY			
Aspectes generals			
Fabricant	ST Microelectronics		
Microcontrolador	STM32L100RCT6 (Nucli ARM-M3)		
Kit de desenvolupament	https://es.rs-online.com/web/p/kits-de-desarrollo-de-microcontroladores/7932803/ Consulta 4/03/2021		
Direcció web del fabricant del microcontrolador	https://www.st.com/en/microcontrollers-microprocessors/stm32l100rc.html Consulta 4/03/2021		
Direcció Web amb documentació proporcionada del fabricant	https://www.st.com/en/microcontrollers-microprocessors/stm32l100rc.html#documentation Consulta 4/03/2021		
IDE fabricant	STM32CubeIDE		
Web IDE	https://www.st.com/en/development-tools/stm32cubeide.html Consulta 4/03/2021		
Alternatives IDE	Arm Kiel uVision https://www2.keil.com/mdk5/uvision/ Consulta 4/03/2021		
Aspectes del Microcontrolador i Placa de desenvolupament			
Freqüència de rellotge	32 MHz	MIPS:	33.3
Arquitectura	RISC	Tipologia processador	32 bits
ADC	12 bits, 24 canals	DAC	2 canals 12 bits
SPI	Si	USART	3
“Timers”	10 (8 de 16 bits+ 2 watchdogs)	Driver LCD	Si
Memòries			

Flash	256 kBtye	RAM	16 kbyte
EEPROM	4 kbyte		
Preu			8,57 €

5.3.2 Fitxa tècnica: ATTINY416 Xplained Nano

Fitxa tècnica: ATTINY416 Xplained Nano			
Aspectes generals			
Fabricant	Microchip		
Microcontrolador	ATtiny416		
Kit de desenvolupament	https://es.rs-online.com/web/p/kits-de-desarrollo-de-microcontroladores/1655135/ Consulta 4/03/2021		
Direcció web del fabricant del microcontrolador	https://www.microchip.com/wwwproducts/en/ATTINY416 Consulta 4/03/2021		
Direcció Web amb documentació proporcionada del fabricant	https://www.microchip.com/wwwproducts/en/ATTINY416#datasheet-toggle Consulta 4/03/2021		
IDE fabricant	MPLABX		
Web IDE	MPLABX https://www.microchip.com/en-us/development-tools-tools-and-software/mplab-x-ide Consulta 4/03/2021		
Aspectes del Microcontrolador i Placa de desenvolupament			
Freqüència de rellotge	20 MHz	MIPS:	20
Arquitectura	RISC	Tipologia processador	32 bits
ADC	10 bits, 1 canal	DAC	1 canal 8 bits
SPI	Si	USART	1
“Timers”	3 (2 de 16 bits i 1 de 12 bits)	Driver LCD	No
Memòries			

Flash	4 kbyte	RAM	256 kbyte
EEPROM	128 kbyte		
Preu			10,58 €

5.3.3 Fitxa tècnica: ATMEGA328P-XMINI

Fitxa tècnica: ATMEGA328P-XMINI			
Aspectes generals			
Fabricant	Microchip		
Microcontrolador	Atmega328P		
Kit de desenvolupament	Consulta 4/03/2021		
Direcció web del fabricant del microcontrolador	https://www.microchip.com/wwwproducts/en/ATmega328P#datasheet-toggle Consulta 4/03/2021		
Direcció Web amb documentació proporcionada del fabricant	https://www.microchip.com/wwwproducts/en/ATmega328P#datasheet-toggle Consulta 4/03/2021		
IDE fabricant	MPLABX		
Web IDE	MPLABX https://www.microchip.com/en-us/development-tools-tools-and-software/mplab-x-ide Consulta 4/03/2021		
Aspectes del Microcontrolador i Placa de desenvolupament			
Freqüència de rellotge	16 MHz	MIPS:	16
Arquitectura	RISC	Tipologia processador	8 bits
ADC	10 bits, 6 canals	DAC	No
SPI	Si	USART	3
Timers	2 de 8 bits 1 de 16 bits	Driver LCD	No

Memòries			
Flash	32 kbyte	RAM	2 kbyte
EEPROM	1 kbyte		
Preu			10,25 €

5.4 Fitxes tècniques components

5.4.1 TLC5628CN1

Descripció: Convertidor digital-analògic (DAC) de 8 bits amb comunicació sèrie. Es va escollir aquest component suposant que la trama seria de 8 bits.

Justificació compra: Presència de interfase sèrie (comunicació amb 3 fils) i es suposava que treballaria en trames de 8 bits. Es va descartar treballar amb DAC's de més bits donat que disposem de un microcontrolador de 8 bits. El preu va ser un factor important també, aquests integrats cars (ordre de 6 € els més senzills).

Pinout:

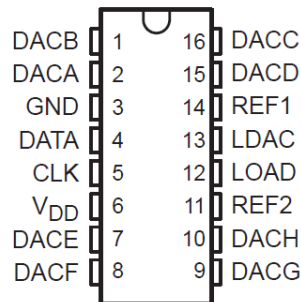


Figura 36. "Pinout" del TLC5628

Format de trama:

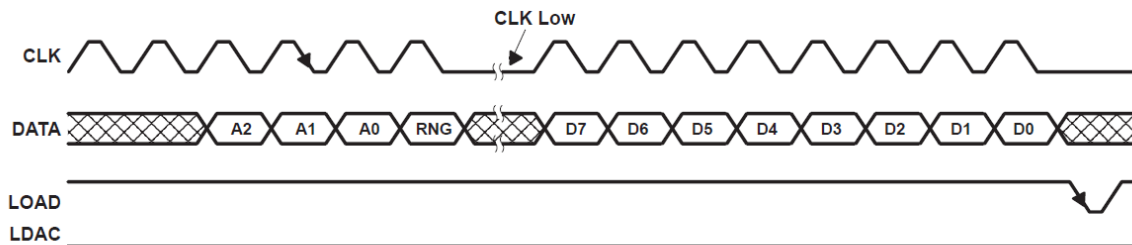


Figure 3. Load-Controlled Update Using 8-Bit Serial Word (LDAC = Low)

Figura 37. Format de trama TLC5628

Es pot comprovar com el DAC treballa en una trama de 12 bits. Els camps D0-D7 corresponen als bits de dades mentre que A2-A0 serveix per triar a quina sortida es vol que el camp de dades actualitzi el valor en sortida. RNG correspon al rang de sortida en que es desitja que treballi el DAC.

Dins del Datasheet⁴ es pot veure com el DAC pot treballar en diversos formats de trama. S'escull el de la Figura 37.

⁴ Veure referència 4 de la Webgrafia

Table 2. Serial Input Decode

A2	A1	A0	DAC UPDATED
0	0	0	DACA
0	0	1	DACB
0	1	0	DACC
0	1	1	DACD
1	0	0	DACE
1	0	1	DACF
1	1	0	DACG
1	1	1	DACH

Figura 38. Camp A2, A1, A0

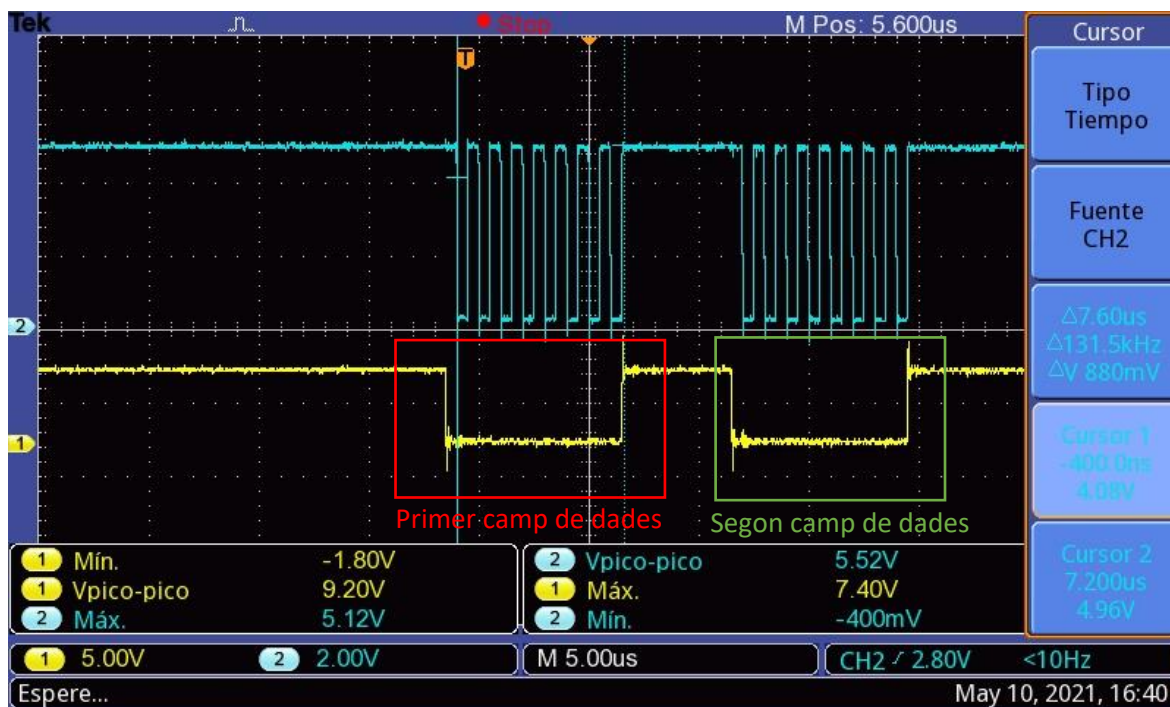


Figura 39. Captura oscil·loscopi: Blau PB5.Groc PB3,

En la Figura 39 es pot veure l'activitat dels Pins PB5 i PB3. PB5 és la sortida de rellotge (*clk*) de la comunicació mitjançant SPI entre el microcontrolador i el DAC.

Es poden veure un total de vuit flancs de baixada en els qual s'envien els bits dels diversos camps. Això està d'acord configurat en el registre SPCR.

Es pot comprovar com la freqüència es de aproximadament 1 MHz. Això està d'acord al configurat en el registre SPCR. S'està dividint per 16 la freqüència de rellotge del microcontrolador.

Pel que fa al pin PB3, el qual conte les dades de la trama, es pot veure com hi ha dos camps. El primer per defecte en totes les transmissions es 0x00, els últims quatre bits de menys pes són A2,A1,A0 i RNG. El segon camp conté el camp de dades, amb els bits de B7 a B0. En aquest cas es pot comprovar que les dades enviades correspondrien a un 0x00 de dades. En el camp de dades de *cad_dac* hi ha valors que son 0 així que es considera que ens trobem davant d'aquest cas.

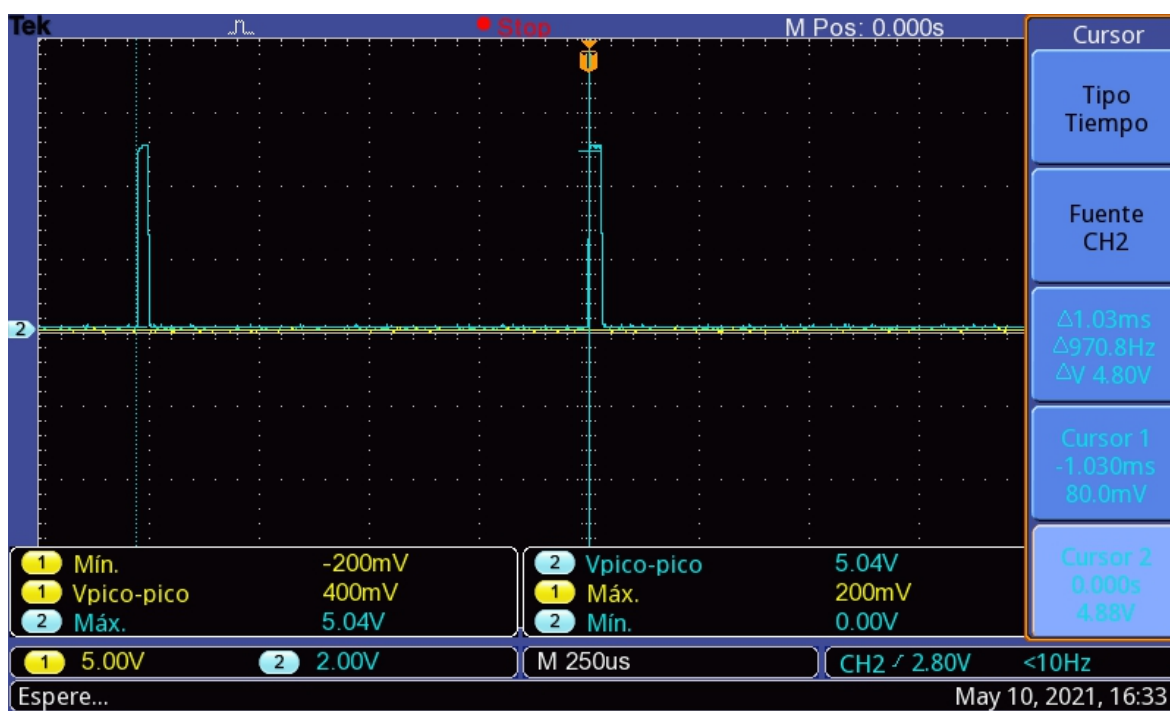


Figura 40. Pin PB2.

El pin PB2 realitza la funció del LOAD de la Figura 37. En el moment en que s'està enviant una trama esta a 1. Un cop ha acabat la transmissió de les dos trames de 8 bits, realitza un flanc de baixada. Al realitzar aquest flanc de baixada el valor de la sortida del DAC s'actualitza amb el valor introduït. Un cop es realitza el flanc de baixada, es manté en valor baix durant 1 ms. Aquest valor s'ha escollit a partir del Datasheet del component.

5.5 Avaluació consums

5.5.1 Botons

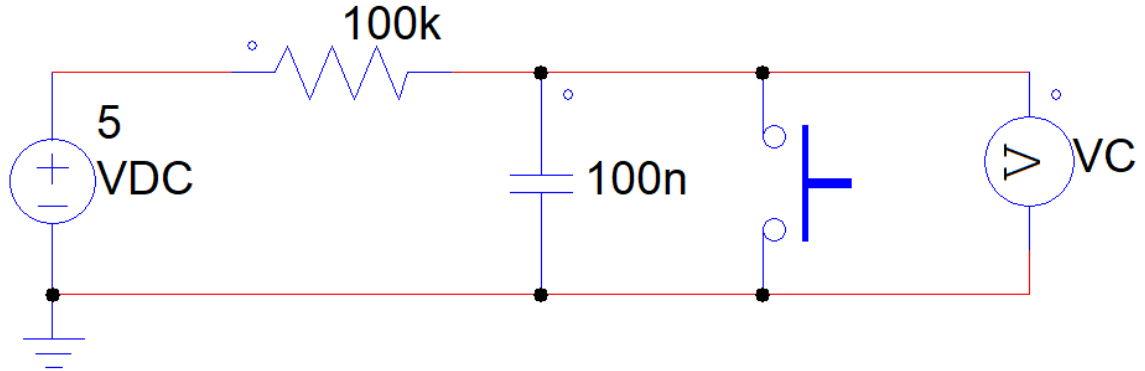


Figura 41. Circuit equivalent botons

Si s'aplica teoria de circuits es veu quin serà el consum quan el contacte estigui obert i tancat.

Equació 5. Càlcul dels consums dels botons

$$I_{obert} = \frac{V_{dc} - V_c}{R} = \frac{5V - 5V}{100k\Omega} = 0A$$

$$I_{tancat} = \frac{V_{dc} - V_c}{R} = \frac{5V - 0V}{100k\Omega} = 50\mu A$$

Si es té en compte que hi han 5 botons, el cas més desfavorable implicaria que els botons tenen un consum de **250 μA** .

5.5.2 Buzzer

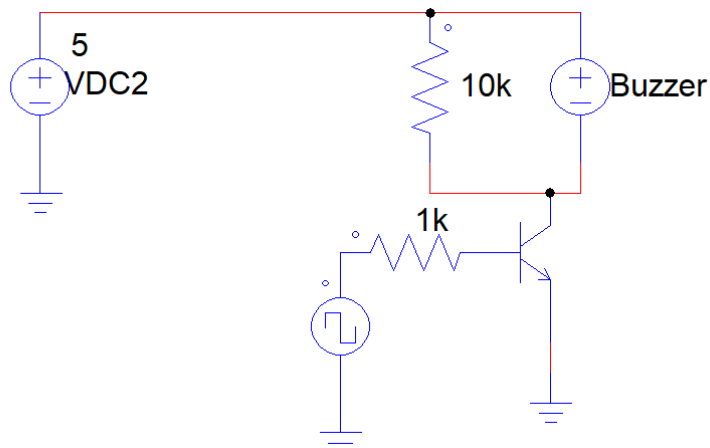


Figura 42. Circuit d'accionament del "Buzzer".

Per poder avaluar el consum i també entendre el funcionament del circuit d'accionament es interessant veure el model equivalent del "Buzzer".

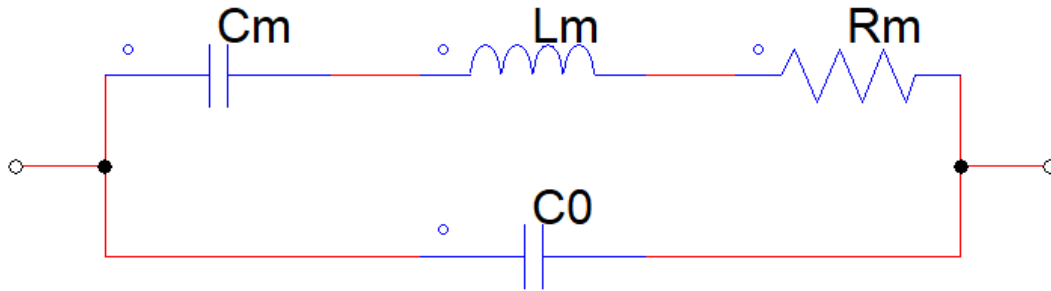


Figura 43. Model equivalent del Buzzer.

On els elements de la Figura 43 són:

- Cm: Capacitància que modelitza el encapsulat mecànic.
- Lm: Inductància del circuit mecànic.
- Rm: Resistència causada per pèrdues mecàniques.
- Co: Capacitància del element ceràmic per sota de la freqüència de ressonància.

Es fàcil veure que el model equivalent en continua seria un circuit obert, es per això que aquest element requereix un accionament mitjançant PWM a certa freqüència.

Adicionalment es necessari situar un resistor de 10 kΩ (veure Figura 42) en paral·lel al Buzzer per tal de descarregar els condensadors en el cas que no estigui accionat.

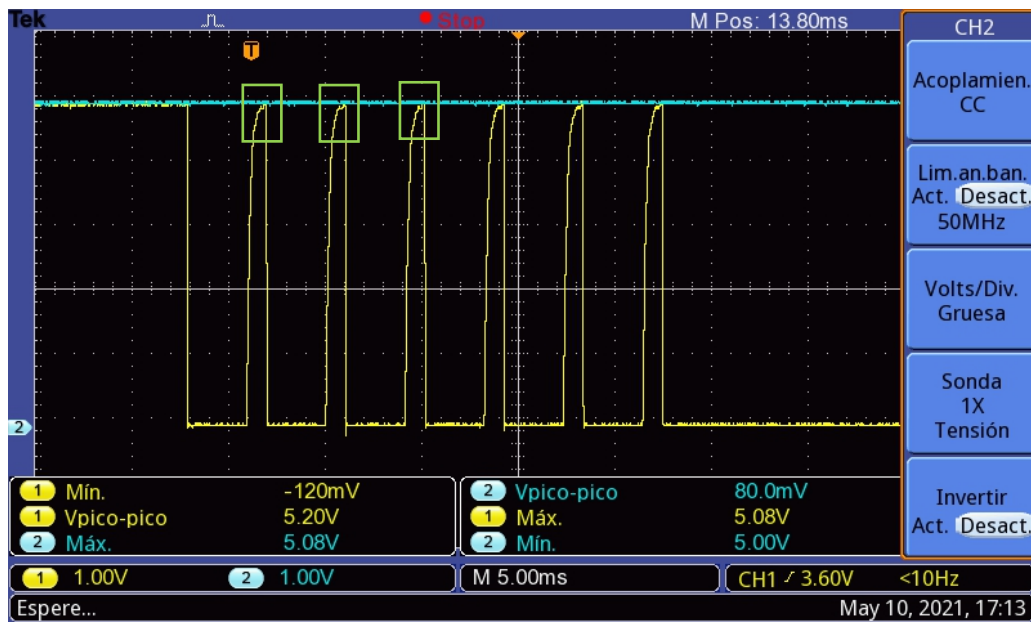


Figura 44. Comportament de la tensió al col·lector del transistor BJT respecte GND. Consigna de 300 BPM's.

En la Figura 44 es demostra que el model de la Figura 43 es vàlid. La corba en la part superior es pròpia de sistemes on hi ha la presència d'un condensador.

En el moment en el qual s'actua sobre la base del transistor, la tensió del col·lector passa a ser pròxima a 0 V. Un cop es deixa d'actuar sobre la base, la tensió creix fins als 5 V (veure Figura 42).

Quan actuem sobre la base del transistor, la tensió del Buzzer entre bornes passa a ser 5 V. En el moment en que es deixa d'actuar, els condensadors del model es descarreguen a través de la resistència de 10 kΩ fins que la tensió dels condensadors es 0 V. En verd a la Figura 44 es pot veure el final de la descarrega. Correspon clarament a un sistema de primer ordre amb un circuit RC.

Si a partir de la figura considerem que la constant de temps es 1 ms:

Equació 6. Modelització del Buzzer com a un únic condensador. Càlcul aproximat de la capacitat equivalent.

$$t = 1 \text{ ms}; t = R * C \rightarrow 1 * 10^{-3} = 10 * 10^3 * c \rightarrow 100 \text{ nF}$$

Cal destacar que el fet de mesurar a la zona del Buzzer introdueix molt soroll i dificulta la feina del *trigger* del oscil·loscopi. Com que la pinça de GND de la sonda del oscil·loscopi està a una distància considerable de la punta de la sonda, s'està formant una antena que recull tota classe de interferències irradiades pels components de la PCB.

5.5.3 LED's

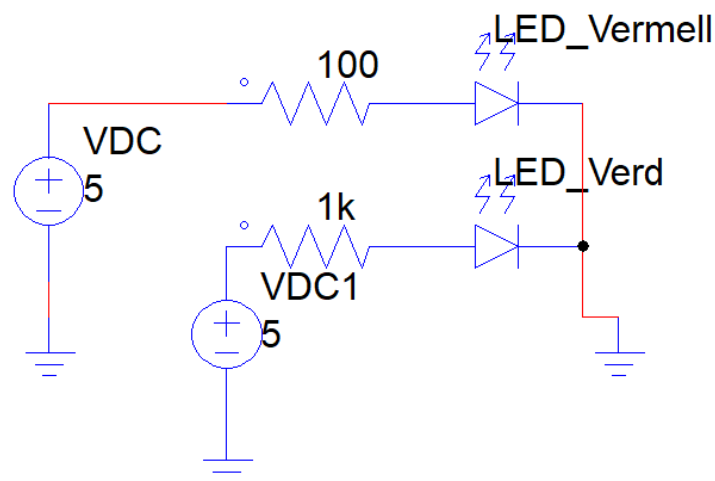


Figura 45. Circuit equivalent quan els LED's estan a ON.

Els dos LED's es poden accionar de forma independent. En el pitjor dels casos, els dos estan encesos alhora.

Equació 7. Càlcul del consum del Led Vermell a partir d'una mesura realitzada amb el multímetre.

$$I_{onVermell} = \frac{V_{R100}}{100 \Omega} = \frac{2.38 \text{ V}}{100 \Omega} = 23.8 \text{ mA}$$

Equació 8. Càlcul del consum del Led Verd a partir d'una mesura realitzada amb el multímetre.

$$I_{onVerd} = \frac{V_{R\ 1k}}{1000\ \Omega} = \frac{2.47\ V}{1000\ \Omega} = 2.47\ mA$$

Donat que els LED es poden accionar alhora, el consum per part dels LED en el pitjor cas serà de $26.27\ mA$. Cal destacar que quan s'acciona el led vermell, la tensió al pin de sortida es de $4.77\ V$. El consum de corrent fa que la regulació de tensió de sortida no sigui perfecta.

5.5.4 LCD

En aquest apartat s'avaluarà el consum dels elements auxiliars que alimenten al display LCD i el consum de la lògica d'aquest.

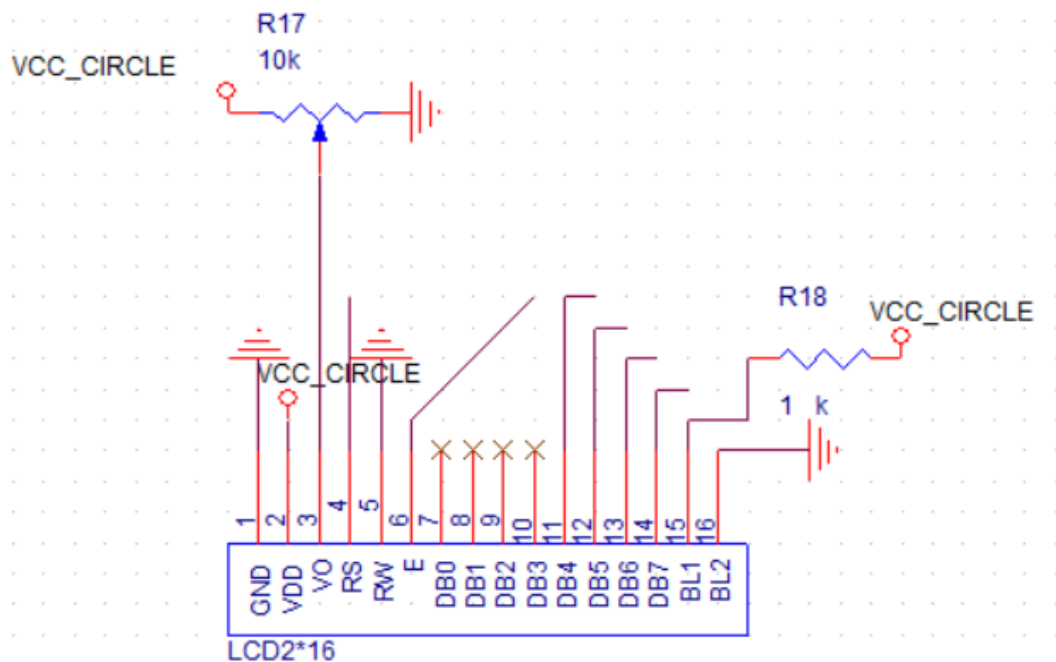


Figura 46. Pinout del display LCD.

A partir del “pinout”, del datasheet⁵ i de mesures amb el multímetre es determinen els següents consums:

Equació 8. Càlcul dels consums del display LCD.

$$i_{R18} = \frac{V_{R18}}{1000\ \Omega} = \frac{2.8\ V}{1000\ \Omega} = 2.8\ mA$$

$$i_{R17} = \frac{V}{10\ k\Omega} = \frac{5\ V}{10\ k\Omega} = 500\ \mu A$$

$$i_{lògica} = 1.25\ mA$$

⁵ Veure referència 7 de la Webgrafia

Per tant el consum total del bloc és:

Equació 9. Consum total del bloc LCD.

$$i_{LCD} = 2.8 \text{ mA} + 500 \mu\text{A} + 1.25 \text{ mA} = 4.55 \text{ mA}$$

5.5.5 Integrats

A partir del datasheet⁶ del TLC 5562 es determina el següent consum.

Equació 10. Consum del integrat TLC 5562.

$$i_{DD} = 4 \text{ mA}$$

Consum de corrent derivat de la creació de la referència de tensió.

Equació 11. Consum de la referència de tensió creada per a alimentar el TLC5562.

$$i_{Vref} = \frac{V}{10 \text{ k}\Omega} = \frac{5 \text{ V}}{10 \text{ k}\Omega} = 500 \mu\text{A}$$

5.5.6 Microcontrolador i Xplained

Segons el DS del fabricant⁷, el microcontrolador tindrà un consum de 6 mA.

A través de la Xplained es pot passar tot el consum de corrent que hi hagi en la PCB. Els diferents elements de la PCB estan alimentats a través del Pin 5 V de la Xplained.

Si es consulta el DS de la Xplained⁸ es pot comprovar que el consum pot ascendir fins a un **màxim** de 500 mA. A partir d'aquest corrent es dispararia un fusible "rearmable" present en la placa.

5.5.7 Càlcul total

A partir de la suma dels consums de cadascun dels elements es determina el consum total del prototip.

Equació 12. Càlcul del consum total del prototip.

$$I_{total} = 0.25 \text{ mA} + 26.27 \text{ mA} + 4.55 \text{ mA} + 4 \text{ mA} + 0.5 \text{ mA} + 6 \text{ mA} = 41.57 \text{ mA}$$

Com es pot veure hi ha molt de marge respecte els 500 mA que es el màxim que podria arribar a proporcionar la Xplained.

Aquest mètode analític per a calcular el consum del prototip es il·lustratiu. És a dir, serveix per a donar l'ordre de magnitud del consum del microcontrolador. S'assumeix que pugin haver-hi variacions respecte al funcionament real del prototip tot i haver utilitzat mesures derivades del circuit real en alguns casos. El mètode més efectiu per avaluar el consum real del prototip es via una resistència shunt veure quina es la caiguda de tensió que es produeix.

⁶ Veure referència 4 de la Webgrafia.

⁷ Veure referència 1 de la Webgrafia

⁸ Veure referència 2 de la Webgrafia

5.6 Costs

5.6.1 Càlcul cost 200 prototips

Taula 14. Càlcul del cost per prototip si es realitza una comanda de 200 prototips.

Component	Codi Rs	Preu (€)	Quantitat	Total (€)
PCB	JLC PCB	1,45	1	1,45
Atmega 328P-Xplained	130-6148	8,47	1	8,47
Transductor	893-7086	4,32	1	4,32
Buzzer	457-011	0,424	1	0,424
TLC5628CN	200-6640	4,38	1	4,38
TLC2272AC	331-2176	3,67	1	3,67
Display Lcd	Amazon	6,35	1	6,35
Transistor BJT	124-1315	0,042	1	0,042
Botons	283-9951	0,38	5	1,9
Resistor 100 kΩ	707-7824	0,142	5	0,71
Resistor 10 kΩ	707-7745	0,125	2	0,25
Resistor 1 kΩ	707-7666	0,125	4	0,5
Resistor 100 Ω	707-7587	0,141	1	0,141
Potenciòmetre 10 kΩ	Farnell (3128459)	0,405	3	1,215
Condensador 100 nF	181-6522	0,113	5	0,565
LED Vermell	228-5988	0,284	1	0,284
LED Verd	228-6004	0,418	1	0,418
Conjunt connector aeri i PCB 90°	Amazon	0,5	1	0,5
Pins femella	766-6607	0,5	1	0,5

Pins mascle	547-3166	0,5	2	1
Sòcol 8 potes	185-9851	0,152	1	0,152
Sòcol 16 potes	185-9864	0,126	1	0,126
Sortida Jack 3.6 mm	913-1030	1,63	1	1,63
Cable connexió		0,5	2	1
Caixa del dispositiu		1	7	7
Suma (€)				48,627
Iva 21%				10,21167
Total (€)				58,83867

Es miren els costos dels diferents components en el cas que es realitzi una comanda de 200 prototips.

5.6.2 Càlcul cost SMT *assembly*

Taula 15. Preu dels components mitjançant la STM assembly de JLC PCB.

Component	Codi Rs	Preu (€)	Quantitat	Total (€)
PCB	JLC PCB	5	1	5
Atmega 328P	JLC PCB	4	1	4
Transductor	893-7086	4,32	1	4,32
Buzzer	457-011	0,564	1	0,564
TLC5628CN	JLC PCB	1,99	1	1,99
TLC2272AC	JLC PCB	2,4754	1	2,4754
Display Lcd	Amazon	6,35	1	6,35
Transistor BJT	JLC PCB	0,0145	1	0,0145
Botons	283-9951	0,38	5	1,9
Resistor 100 kΩ	JLC PCB	0,006	5	0,03
Resistor 10 kΩ	JLC PCB	0,006	2	0,012
Resistor 1 kΩ	JLC PCB	0,006	4	0,024
Resistor 100 Ω	JLC PCB	0,006	1	0,006
Potenciòmetre 10 kΩ	JLC PCB	0,3878	3	1,1634
Condensador 100 nF	JLC PCB	0,01	5	0,05
LED Vermell	228-5988	0,3	1	0,3
LED Verd	228-6004	0,418	1	0,418
Conjunt connector aeri i PCB 90°	JLC PCB	0,5	1	0,5
Pins femella	JLC PCB	0,05	1	0,05

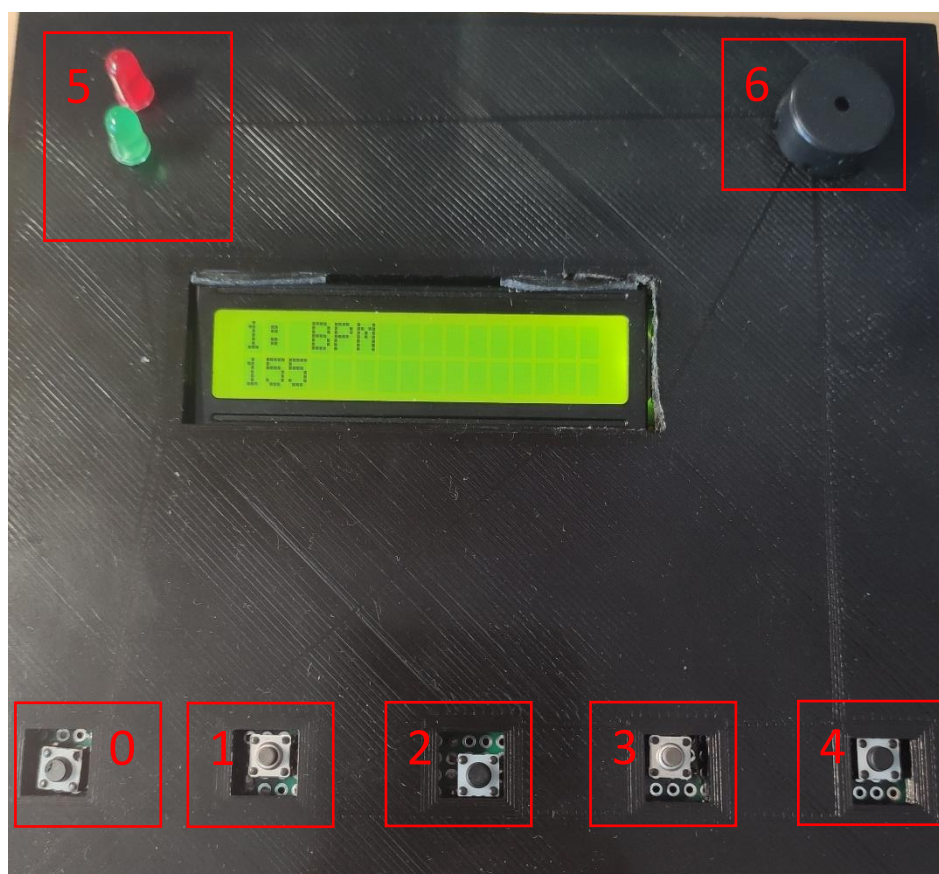
Pins mascle	JLC PCB	0,05	2	0,1
Sortida Jack 3.6 mm	JLC PCB	0,6	1	0,6
Cable conexió		0,5	2	1
Caixa del dispositiu		1	6	6
Suma				37,4673
Iva 21%				7,868133
Total				45,335433

El càlcul dels preus dels diferents components es realitza mitjançant el web de JLCPCB dedicat a mostrar el cost dels diferents components en el cas que s'usi la SMT Assembly.⁹

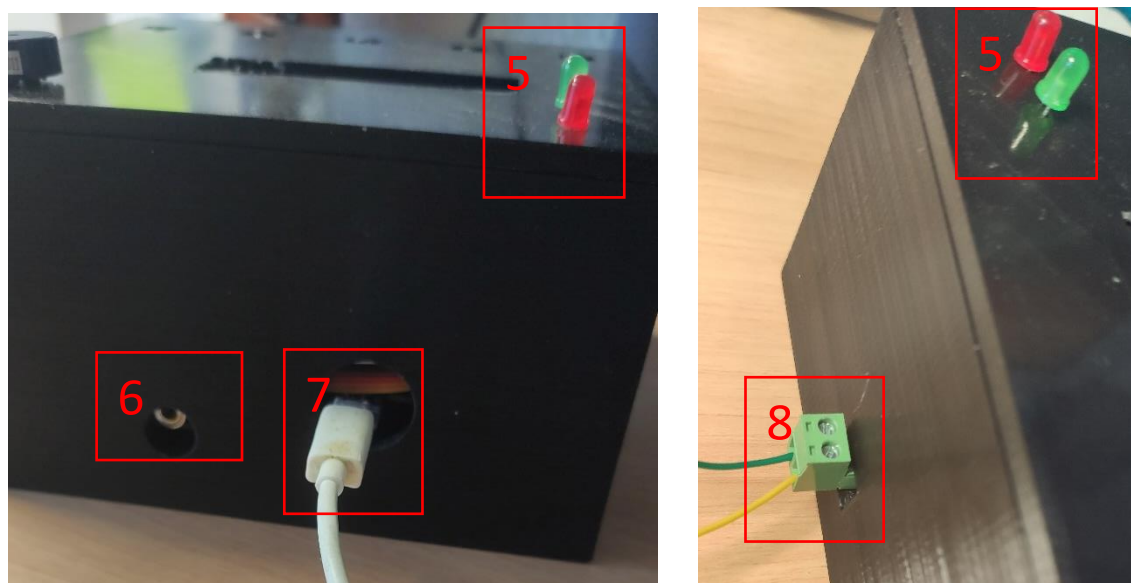
⁹ Veure referència 4 del annex 6.2

5.7 Manual d'usuari

Gràcies per adquirir el Metrònom Digital. A continuació es descriu el funcionament d'aquest.



1 Vista superior del metrònom.



2 Vista Posterior i vista lateral del dispositiu

Element assenyalat	Descripció
0	Botó 0
1	Botó 1
2	Boto 2
3	Boto 3
4	Botó 4
5	Leds per a la visualització de la consigna
6	Sortida d'àudio jack 3.6 mm
7	Alimentació del prototip a través de un cable micro USB.
8	Bornes per la connexió per part del usuari del transductor
9	Display LCD

Primers passos:

Per alimentar al prototip convé alimentar-lo via cable USB (no inclòs). Es pot alimentar mitjançant un carregador de mòbil, una “*powerbank*” o connectant el cable directament al port USB d'un ordinador.

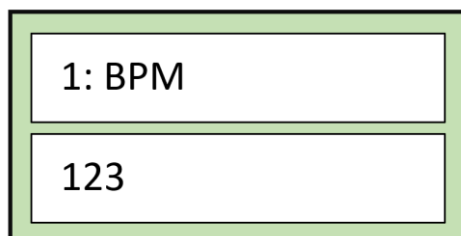
El cable s'ha de connectar al connector USB enumerat amb el nombre 7.

Navegació a través dels menús i funcionalitats:

El metrònom disposa de tres menús, per navegar a través d'ells s'ha de pitjar el **botó 0**.

Menú 1

Menú principal, per defecte el metrònom s'inicia en aquest menú. A continuació es veu la visualització



Funcionalitats:

Botó	Funció
1	Baixa 1 BPM la consigna
2	Puja 1 BPM la consigna
3	Parar/Encendre del metrònom amb la consigna de la pantalla
4	Creació d'una nova consigna i s'encén el metrònom
Funcionalitat extra	Si abans de pitjar el botó 4, es pitja el botó 3 i es manté premut durant els quatre cops necessaris per a generar la consigna, s'actualitza el valor de consigna però no s'encén el metrònom.

Per generar la consigna s'ha de prémer 4 cops el botó 4. El primer cop que es premi el botó s'encendran els LED verd i vermells superiors. Aquests indiquen que el metrònom està buscant la nova consigna. Si passa massa temps, la consigna que es generarà serà de 30 BPM.

Es recomana no pitjar cap altre botó mentre s'està generant la consigna. En el cas que es vulgui generar una nova consigna **però no encendre el metrònom**, abans de prémer el botó 4 per primer cop s'ha de prémer el botó 3. Per tal que no s'encengui el botó 3 ha d'estar premut fins al final de la generació de consigna.

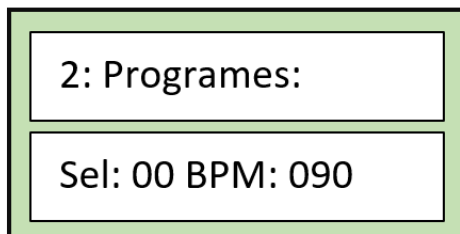
Els botons 1 i 2 serveixen per a pujar/ baixar individualment la consigna.

Una vegada el metrònom funciona seguint una consigna, aquesta es reproduïx de tres formes:

- A través dels LED.

- A través del Buzzer (cal que estigui habilitat, veure Menú 3).
- A través de la sortida d'àudio. El plug jack de 3.6 mm d'auriculars s'ha de connectar al element enumerat amb el nombre 6.

Menú 2



Funcionalitats:

Botó	Funció
1	Baixa 1 el programa seleccionat
2	Puja 1 el programa seleccionat
3	Para/Encén el metrònom en la consigna que hi ha en el programa seleccionat.
4	Al pitjar tres cops seguits es guarda al programa seleccionat el valor de consigna que tingui el metrònom
Botó	Funció

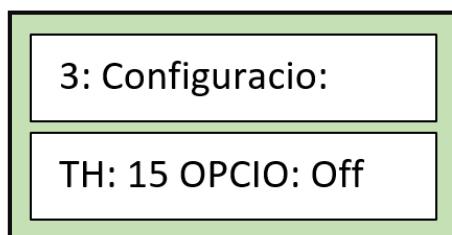
Menú per guardar 20 consignes. Destinat al ús en concerts, pots guardar tot el repertori i canviar de cançó en dos clics. Hauràs de prémer el boto 2 i posteriorment el 3. El metrònom passarà a funcionar amb la nova consigna.

Per tal de guardar una consigna a un programa es molt senzill:

- Genera la consigna que vulguis al menú 1.
- Canvia al menú 2.
- Ves al programa on vulguis guardar la consigna que acabes de generar
- Posa tres cops el botó 4.

Menú 3

Menú destinat a la configuració del metrònom.



Botó	Funció
1	Baixa 1 el llindar o “ <i>Threshold</i> ” del ADC
2	Puja 1 el llindar o “ <i>Threshold</i> ” del ADC
3	“ <i>Toggle</i> ” entre on i off. S’habilita o no que el “ <i>buzzer</i> ” emeti sons seguint la consigna.
4	Es guarda la configuració

En aquest menú s’ajusta el valor del llindar de detecció de cop del element connectat al element terminal. Aquest paràmetre s’ajusta amb el botó 1 i botó 2.

Amb el botó 3 s’habilita i s’inhabilita l’ús del Buzzer. Mitjançant el botó 4 es guarda tota la configuració.

Element de detecció:

La funcionalitat més rellevant del metrònom és la capacitat de detectar la generació de consignes a través de un element sensor connectat a un element terminal.

Instal·lació:



Exemple d’instal·lació del element sensor en una caixa. S’ha de situar pròxim al aro evitant el cop directe. Mitjançant cinta adhesiva s’ha de situar el element damunt de la superfície de glopeig. Es recomana traçar els cables d’una forma recta.

El connector l'altre extrem del element sensor s'ha de connectar al connector enumerat amb un 8.

Configuració:

Cada cop que l'element sensor s'instal·li en una ubicació diferent convé ajustar el nivell de *threshold*. Per tal de configurar-lo s'ha de seguir el següent procés:

- Al menú 1 prémer el botó 4 per començar el procés de generació de consigna.
- Realitzar un cop a l'element al qual el element sensor esta connectat.
- En el cas que no es detecti cop baixar el nivell de *threshold*. En el cas que es detectin més cops dels que toca, pujar el nivell de *threshold*.
- Repetir el procés fins que els cops a l'element corresponguin amb la generació de consigna final.

Generació de consigna a través de l'element sensor:

Per tal de generar la consigna a través del element sensor s'ha de seguir la següent seqüència:

- Prémer el botó 4 estant al menú 1.
- Realitzar un cop a l'element sensor. A partir d'aquest instant es començarà a comptar per a la generació de la consigna. Per tal de produir els tres cops següents es pot realitzar un cop el element sensor o bé usar el botó 4.

6 Referències

6.1 Webgrafia

- 1) Microchip. megaAVR® Data Sheet DS40002061B. Microchip 2020. [16/05/2021]
Disponible a:
<
<https://www.google.com/url?sa=t&rct=j&q=&esrc=s&source=web&cd=&ved=2ahUKEwj61siDh87wAhWWhf0HHV2XCxgQFjAAegQIAhAD&url=https%3A%2F%2Fww1.microchip.com%2Fdownloads%2Fen%2FDeviceDoc%2FATmega48A-PA-88A-PA-168A-PA-328-P-DS-DS40002061B.pdf&usg=AOvVaw0AeyDbLKnsAW56eAs8jbK5>
>
- 2) Microchip. Atmega328P Xplained Mini DS50002659B. Microchip 2020 [16/05/2021]. Disponible a:
<
<https://www.google.com/url?sa=t&rct=j&q=&esrc=s&source=web&cd=&cad=rja&uact=8&ved=2ahUKEwiXiJrFqM7wAhVLXhoKHZygDRwQFjAAegQIBhAD&url=http%3A%2F%2Fww1.microchip.com%2Fdownloads%2Fen%2FDeviceDoc%2FATmega328P-Xplained-Mini-UG-DS50002659B.pdf&usg=AOvVaw2ZAN3XZmb2YGtH-mMj7rmm>
>
- 3) Texas Instruments. TLC227x, TLC227xA: Advanced LinCMOS Rail-to-Rail Operational Amplifiers. Texas instruments 2016. [16/05/2021]. Disponible a:
<<https://www.ti.com/product/TLC2272?qgpn=tlc2272> >
- 4) Texas Instruments. TLC5628C, TLC5628I OCTAL 8-BIT DIGITAL-TO-ANALOG CONVERTERS. Texas instruments 1997. Disponible a:
<
https://www.google.com/url?sa=t&rct=j&q=&esrc=s&source=web&cd=&cad=rja&uact=8&ved=2ahUKEwido7Grqs7wAhVKyYUKHb3NAXsQFjAAegQIBBAD&url=https%3A%2F%2Fwww.ti.com%2Flit%2Fds%2Fsymlink%2Ftlc5628.pdf&usg=AOvVaw3lWQTxs_C3AdRSZIP28MYT >
- 5) St Microelectronics. Discovery kit for STM32L100 Value Line. St Microelectronics. St Microelectronics 2014.[16/05/2021] Disponible a:
<
https://www.google.com/url?sa=t&rct=j&q=&esrc=s&source=web&cd=&ved=2ahUKEwj_7Z3Gsc7wAhVGTBoKHWvvDQ8QFjAMegQIBRAD&url=https%3A%2F%2Fdocs.rs-online.com%2F2516%2F0900766b812ddcf7.pdf&usg=AOvVaw3B8L3Tp6oYa8UsGH1LDjBs >
- 6) Microchip. AVR® Microcontroller with Core Independent Peripherals

and picoPower® Technology. Microchip 2017 [19/05/2021] Disponible a:

<

<https://www.google.com/url?sa=t&rct=j&q=&esrc=s&source=web&cd=&ved=2ahUKEwjn1c-ss7wAhUO4YUKHTKaAR8QFjABegQIARAD&url=http%3A%2F%2Fww1.microchip.com%2Fdownloads%2Fen%2FDeviceDoc%2F40001913A.pdf&usg=AOvVaw24mO7hHhroFRUOMeheqJoD> >

7) BEJING Qingyun Hi-Tech Development Co. LTD. Standard Character LCM. BEJING Qingyun Hi-Tech Development Co. LTD [19/05/2021]. Disponible a : <<https://datasheetspdf.com/pdf/1031442/QINGYUNHI-TECH/LCM1602C/1> >

8)

http://web.alfredstate.edu/faculty/weimandn/lcd/lcd_initialization/lcd_initialization_index.html

6.2 Webs de proveïdors

- 1) JLCPCB. [16/05/2021]. < <https://jlcpcb.com/> >
- 2) Rscomponents. [16/05/2021]. < <https://es.rs-online.com/web/> >
- 3) Farnell. [16/05/2021] < <https://es.farnell.com/> >
- 4) JLCPCB SMT Assembly Prices [16/05/2021]. < <https://jlcpcb.com/parts> >

6.3 Referències imatges

Llistat imatges:

Figura 1. Tama RW200.

Figura 2. Prototip final (vista superior)

Figura 3. Esquerra: vista posterior del prototip. Dreta: vista lateral del prototip.

Figura 4. Transductor TE 1002794.

Figura 5. Exemple de instal·lació del transductor en el element terminal

Figura 6. Xplained ATMEGA328P

Figura 7. Pinout de la Xplained.

Figura 8. Símbol de la Xplained en el esquemàtic del circuit electrònic.

Figura 9. Traçat de les pistes i zones de coure de la PCB

Figura 10 Traçat de les pistes per la cara inferior o "bottom"

Figura 11. Cara superior de la PCB.

Figura 12. Cara inferior de la PCB.

Figura 13. Estat de la cara superior un cop s'ha realitzat el procés de soldadura.

Figura 14. Estat de la cara inferior un cop realitzat el procés de soldadura.

Figura 15. Diagrama del bucle principal d'execució

Figura 16. Visualització del display al menú 1.

Figura 17. Visualització del display al menú 2

Figura 18. Visualització del display al menú 3

Figura 19. Temps necessari per a la generació de la consigna

Figura 20. Diagrama de la actualització de "cop".

Figura 21. Diagrama del Timer 2

Figura 22. Diagrama sobre l'ús de la EEPROM

Figura 23. Perspectiva superior de la Tapa

Figura 24. Perspectiva inferior dreta de la tapa.

Figura 25. Perspectiva de la part inferior de la caixa.

Figura 26. Perspectiva superior de la part inferior.

Figura 27. Impressora 3D del DEEEA

Figura 28. Tapa un cop impresa enquestada en la part inferior

Figura 29. Part inferior un cop impresa

Figura 30. Primer estudi econòmic.

Figura 31. Segon estudi econòmic.

Figura 32 Kit de desenvolupament Discovery

Figura 33: ATtiny416 Xplained Nano

Figura 34. Pinout ATtiny416

Figura 35. Atmega 328P-Xplained Mini

Figura 36. Pinout del TLC5628

Figura 37. Format de trama TLC5628

Figura 38. Camp A2, A1, A0

Figura 39. Captura oscil·loscopi: Blau PB5.Groc PB3,

Figura 40. Pin PB2.

Figura 41. Circuit equivalent botons

Figura 42. Circuit d'accionament del "Buzzer".

Figura 43. Model equivalent del Buzzer.

Figura 44. Comportament de la tensió al col·lector del transistor bjt respecte GND. Consigna de 300 BPM's.

Figura 45. Circuit equivalent quan els LED's estan a ON.

Figura 46. Pinout del display LCD.

	Autoria
Figura 1	Tama. Extreta de: https://www.tama.com/usa/products/detail/rw200.html
Figura 2	Pròpia
Figura 3	Pròpia
Figura 4	Pròpia
Figura 5	Pròpia
Figura 6	Pròpia
Figura 7	Microchip. Extreta de: < https://www.google.com/url?sa=t&rct=j&q=&esrc=s&source=web&cd=&cad=rja&uact=8&ved=2ahUKEwiXiJrFqM7wAhVLXhoKHZygDRwQFjAAegQIBhAD&url=http%3A%2F%2Fww1.microchip.com%2Fdownloads%2Fen%2FdeviceDoc%2FATmega328P-Xplained-Mini-UG-DS50002659B.pdf&usg=AOvVaw2ZAN3XZmb2YGtH-mMj7rmm >
Figura 8	Pròpia
Figura 9	Pròpia
Figura 10	Pròpia
Figura 11	Pròpia
Figura 12	Pròpia
Figura 13	Pròpia
Figura 14	Pròpia
Figura 15	Pròpia
Figura 16	Pròpia
Figura 17	Pròpia
Figura 18	Pròpia

Figura 19	Pròpia
Figura 20	Pròpia
Figura 21	Pròpia
Figura 22	Pròpia
Figura 23	Pròpia
Figura 24	Pròpia
Figura 25	Pròpia
Figura 26	Pròpia
Figura 27	Pròpia
Figura 28	Pròpia
Figura 29	Pròpia
Figura 30	Pròpia
Figura 31	Pròpia
Figura 32	St Microelectronics. Extreta de: https://www.google.com/url?sa=t&rct=j&q=&esrc=s&source=web&cd=&ved=2ahUKEwj_7Z3Gsc7wAhVGTBoKHWvvDQ8QFjAMegQIBRAD&url=https%3A%2F%2Fdocs.rs-online.com%2F2516%2F0900766b812ddcf7.pdf&usg=AOvVaw3B8L3Tp6oYa8UsGH1LDjBs >
Figura 33	
Figura 34	Microchip. Extreta de: https://www.google.com/url?sa=t&rct=j&q=&esrc=s&source=web&cd=&ved=2ahUKEwjn1c-ss7wAhUO4YUKHTKaAR8QFjABegQIARAD&url=http%3A%2F%2Fww1.microchip.com%2Fdownloads%2Fen%2FDeviceDoc%2F40001913A.pdf&usg=AOvVaw24mO7hHhroFRUOMeheqJoD >
Figura 35	Microchip. Extreta de: < https://www.google.com/url?sa=t&rct=j&q=&esrc=s&source=web&cd=&cad=rja&uact=8&ved=2ahUKEwiXiJrFqM7wAhVLXhoKHZygDRwQFjAAegQIBhAD&url=http%3A%2F%2Fww1.microchip.com%2Fdo

	wnloads%2Fen%2FDeviceDoc%2FATmega328P-Xplained-Mini-UG-DS50002659B.pdf&usg=AOvVaw2ZAN3XZmb2YGtH-mMj7rmm >
Figura 36	Texas instruments. Disponible a: < https://www.google.com/url?sa=t&rct=j&q=&esrc=s&source=web&cd=&cad=rja&uact=8&ved=2ahUKEwido7Grqs7wAhVKyYUKHb3NAXsQFjAAegQIBBAD&url=https%3A%2F%2Fwww.ti.com%2Flit%2Fds%2Fsymlink%2Ftlc5628.pdf&usg=AOvVaw31WQTxs_C3AdRSZIP28MYT >
Figura 37	Ídem a la figura 36
Figura 38	Ídem a la figura 37
Figura 39	Pròpia
Figura 40	Pròpia
Figura 41	Pròpia
Figura 42	Pròpia
Figura 43	Pròpia
Figura 44	Pròpia
Figura 45	Pròpia
Figura 46	Pròpia