

Treball final de grau

Estudi: Grau en Enginyeria Electrònica Industrial i Automàtica

Títol: Pedal digital de guitarra programable

Document: 1. Memòria

Alumne: Marta González i Palomino

Tutor: Daniel Alexandre Macaya i Masferrer

Departament: EEEA

Àrea: ESA

Convocatòria (mes/any): setembre/2018

ÍNDEX

1. INTRODUCCIÓ	3
1.1. Antecedents	3
1.2. Objecte.....	3
1.3. Especificacions i abast	3
2. SENYAL DE LA GUITARRA I ESTRUCTURA	4
2.1. El senyal de la guitarra elèctrica.....	5
2.2. Estructura de disseny del pedal.....	8
3. SELECCIÓ DE COMPONENTS	10
3.1. Placa de desenvolupament NUCLEO-F429ZI	10
3.1.1. Comparació de dispositius.....	10
3.1.2. Descripció de la placa de desenvolupament.....	13
3.2. Còdec WM8734.....	15
3.2.1. Configuració I2C.....	17
3.2.2. Configuració I2S.....	19
3.3. Amplificador operacional OPA725.....	20
4. DISSENY I FUNCIONAMENT DEL CIRCUIT	23
4.1. Condicionament de l'entrada	24
4.2. Condicionament de la sortida	28
4.3. Connexions del còdec	30
4.4. Control i visualització.....	34
4.4.1. Polsador 4PDT	34
4.4.2. Selector d'efectes i potenciòmetres	35
4.4.3. Mòdul LCD	37
5. DISSENY D'EFECTES DIGITALS	39
5.1. Efectes de distorsió del senyal	39
5.2. Efectes de retard del senyal	41
5.3. Filtre pas-baix.....	45

6. PROGRAMACIÓ NUCLEO-F429ZI	47
6.1. Inicialització i configuració del còdec	48
6.2. Màquina d'estat	49
6.2.1 Efectes	52
6.3. Programació I2S.....	53
6.4. Interrupció i entrades analògiques.....	54
7. RESUM DEL PRESSUPOST	56
8. CONCLUSIONS	57
9. RELACIÓ DE DOCUMENTS	58
10. BIBLIOGRAFIA.....	59
11. GLOSSARI	61
A. CONFIGURACIÓ STM32 CUBEMX	63
A.1. Interfície del programa	63
A.2. Configuració de perifèrics	66
A.3. Codi generat per CUBEMX	68
B. PROGRAMA	70

1. INTRODUCCIÓ

1.1. Antecedents

Els pedals de guitarra digitals ofereixen múltiples efectes de so gràcies al processament digital del senyal. No obstant, actualment existeixen pocs pedals digitals els quals es pugui modificar la seva programació interna. Els xips que s'utilitzen són específics per a cada marca, com per exemple els processadors de les marques BOSS i DigiTech.

1.2. Objecte

L'objectiu d'aquest projecte és dissenyar un pedal digital de guitarra programable, el qual permeti a l'usuari programar diferents efectes existents i crear-ne de nous totalment personalitzats.

1.3. Especificacions i abast

Es dissenyaran alguns efectes de guitarra (com per exemple overdrive, "delay"...) per tal de poder implementar la programació i funcionament del pedal, i s'escollirà el tipus de tecnologia adient per aplicar els algorismes dels efectes.

Es seleccionaran els elements necessaris per adquirir i generar de nou el senyal (convertors A/D i D/A). També s'adequarà el senyal per la correcta conversió. S'utilitzaran elements per seleccionar i visualitzar l'efecte desitjat, com podrien ser LED's, switchos i pantalles LCD.

2. SENYAL DE LA GUITARRA I ESTRUCTURA

Els pedals de guitarra permeten modificar la forma d'ona de el senyal que prové de la guitarra. A partir de la seva modificació, es poden obtenir multitud d'efectes de so.

Alguns dels efectes més coneguts es troben associats a un gènere de música en particular; per exemple la distorsió per als gèneres de rock o metal. Aquests efectes es poden superposar, de manera que sonin diversos efectes alhora.

Actualment en el mercat dels pedals de guitarra, es poden trobar tant pedals analògics com digitals. Els pedals analògics es construeixen amb un o diversos circuits amb amplificadors operacionals, resistències i condensadors. Per contra, els pedals digitals adquireixen el senyal, a partir d'un algoritme es modifiquen les mostres i es torna a reconstruir el senyal.

Existeixen pedals digitals que només produeixen un tipus d'efecte, o es poden trobar pedaleres que incorporen multitud d'efectes. L'inconvenient d'aquests pedals és que no permeten la modificació de la seva programació interna.



Figura 1. Pedalera d'efectes BOSS ME-80

En aquest projecte es dissenyarà un pedal digital de guitarra programable, el que permeti a l'usuari programar l'efecte desitjat, el que permet l'aprenentatge del processament digital del senyal i a crear efectes totalment personalitzats.

Primer es farà una petita introducció al senyal de la guitarra, posteriorment s'explicarà l'estructura adoptada per tal de realitzar el pedal digital.

2.1. El senyal de la guitarra elèctrica

La guitarra elèctrica produeix senyal gràcies a la interacció entre els elements que la componen: les cordes i les pastilles.

Les pastilles de la guitarra elèctrica estan formades per sis imants permanents (un per a cada corda) i un bobinatge al voltant d'aquests. Aquests imants es solen fabricar d'alnico (un aliatge d'alumini, níquel i cobalt), mentre que les cordes són normalment d'acer. Entre aquests dos components es crea un camp magnètic, degut a l'atracció entre les cordes i els imants.



Figura 2. Pastilla simple amb bobinat de coure

Quan es toquen les cordes, les vibracions d'aquestes varien el flux del camp magnètic als imants, provocant la variació del voltatge de sortida, creant així el senyal resultant de la guitarra. El corrent canvia de sentit en funció del moviment d'oscil·lació de les cordes; cap a una banda és positiu i cap a l'altre, negatiu. Aquest canvi de sentit provoca que el senyal de la guitarra sigui alterna.

El senyal de sortida és una suma de tots els corrents induïts al tocar cadascuna de les cordes. Si es toca una sola corda, el senyal serà més nítid que no pas al tocar un acord.

A la sortida de les pastilles trobem els controls de to i de volum de la guitarra; el control de to pot actuar com a filtre pas alt o pas baix, i el control de volum regula la tensió de sortida de les pastilles.

També trobem un selector per tal de seleccionar la pastilla a utilitzar, o fer una combinació de dues; per exemple, una guitarra que té tres pastilles, pot arribar a tenir un selector de cinc posicions.

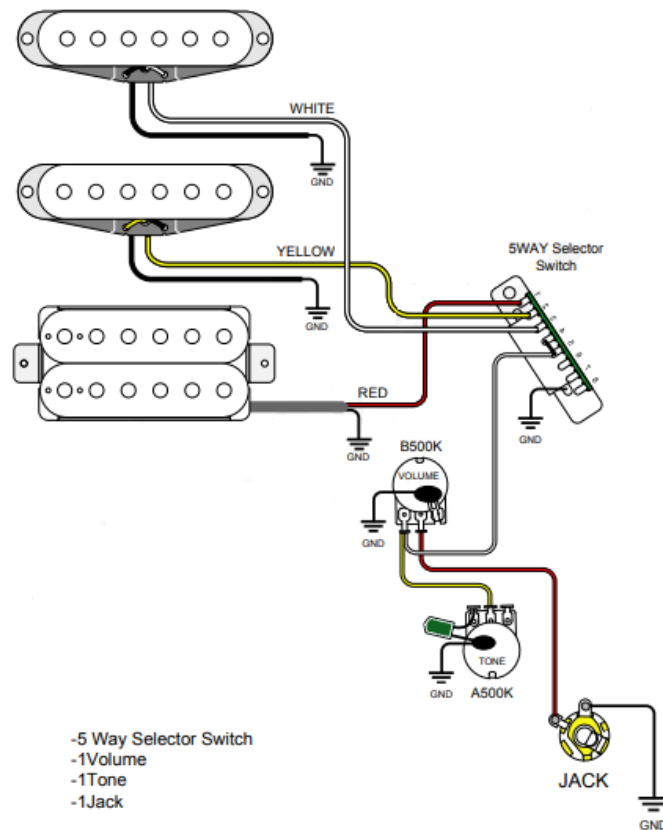


Figura 3. Connexions internes de la guitarra elèctrica

En el mercat existeixen molts tipus de pastilles; cadascuna, segons la seva construcció i materials creen el so característic dels tipus de guitarra existents. Els paràmetres que determinen el so d'una pastilla de guitarra són la tensió de sortida i la freqüència de ressonància. A nivell de tensió, les pastilles es poden classificar en simples o dobles, i en actives o passives.

Les pastilles dobles es troben formades per dues pastilles simples, que tenen polaritats oposades; el que s'aconsegueix amb aquesta configuració és la suma de les tensions de sortida provocada per les vibracions de les cordes, i la cancel·lació del soroll que es pugui introduir a les pastilles (soroll provinent d'altres aparells). Amb la suma de les tensions, les pastilles dobles tindran el doble de tensió de sortida que una pastilla simple.

L'altre classificació que s'ha de considerar és si les pastilles són actives o passives. Les pastilles es consideren passives quan no tenen cap circuit addicional que amplifiqui el senyal de sortida d'aquesta. Les pastilles actives porten un circuit d'amplificació amb un operacional i resistències. Les pastilles actives tindran una tensió de sortida més elevada que les pastilles passives.

S'han buscat alguns fabricants de pastilles per a guitarra per tal de comparar les tensions de sortida en funció del tipus de pastilla. Les següents pastilles pertanyen a la marca DiMarzio. Totes són pastilles de tipus passives.

Model pastilla	Tipus	Vout (corda)	Rout (k Ω)
HS-2	Simple	90	14,97
Super distortion S	Simple	325	13,18
X2N	Doble	510	15,83
Imperium Bridge	Doble	325	13,70

Taula 1. Característiques de sortida de pastilles passives

Les següents pastilles pertanyen a la marca EMG. Totes són pastilles de tipus actives.

Model pastilla	Tipus	Vout (corda)	Vout (acord)	Rout (k Ω)
H	Doble	1,0	3,0	10
HA	Doble	1,0	3,0	10
58	Doble	3,0	4,5	10

Taula 2. Característiques de sortida de pastilles actives

Totes les tensions de sortida anteriors són valors de pic a pic. Per aquest projecte, s'ha acotat només per a pastilles de tipus passives.

Un paràmetre a tenir en compte, a l'hora de definir el senyal de la guitarra, és l'amplada de banda d'aquest. L'amplada de banda anirà des de la sisena corda al aire (82,412Hz) , fins a l'últim traste de la primera corda (1.174,624Hz). Aquestes freqüències pertanyen l'afinació estàndard de la guitarra.

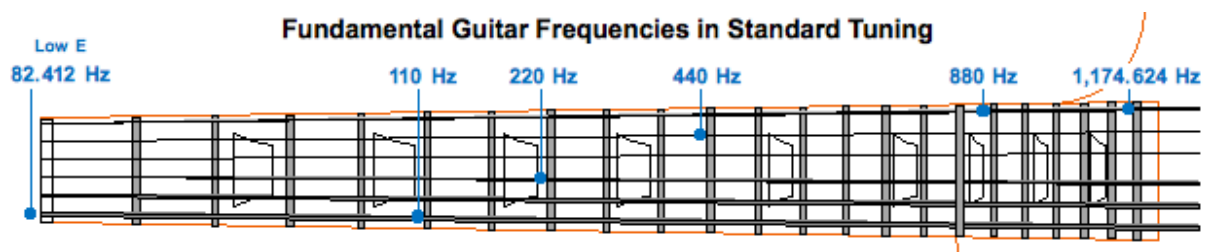


Figura 4. Rang de freqüències de la guitarra

Però la banda de freqüència no es troba limitada en aquest rang únicament; el senyal de sortida de la guitarra no és sinusoidal com es podria pensar en un principi. És una senyal que, a part de contenir la freqüència fonamental (les esmentades anteriorment), és la suma

d'aquesta més una sèrie d'armònics. Per tant la freqüència més elevada no serà el valor de 1.174,624Hz.

Tal i com s'ha esmentat anteriorment, hi ha una gran varietat de pastilles en el mercat, i cadascuna té una freqüència de ressonància diferent; normalment les freqüències de ressonància es troben compreses entre els 3 i 6kHz.

Podrem tenir valors en la banda mitja del àudio (compresa entre els 2kHz i els 6kHz). Això és degut a que les pastilles tenen una resposta en freqüència en la qual a partir de la freqüència de ressonància, les freqüències superiors es troben atenuades.

2.2. Estructura de disseny del pedal

Totes els pedals digitals utilitzen la tecnologia DSP (processament digital del senyal) per modificar el senyal de la guitarra. El processament digital del senyal consisteix en adquirir un senyal analògic i utilitzar un processador digital per tal de modificar o analitzar el senyal.

Gràcies al processament digital del senyal, es poden aconseguir els mateixos resultats que amb l'electrònica analògica, o realitzar aplicacions les quals no es podrien realitzar amb aquesta. Gràcies a la tecnologia DSP es poden re-programar els xips, sense haver de canviar la part física.

Per tal de poder analitzar o modificar els senyals analògics, existeixen els processadors digitals, també anomenats DSP, que tenen la rapidesa suficient per realitzar grans quantitats d'operacions matemàtiques.

El processament digital del senyal es realitza basant-se en el diagrama de blocs següent:

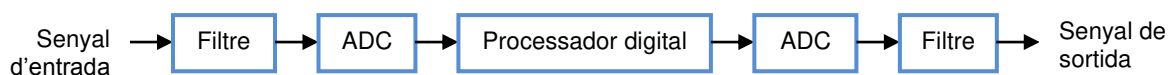


Figura 5. Diagrama de blocs del procés digital del senyal

La senyal d'entrada passa per un procés de filtratge, per treure el soroll no desitjat del senyal. Un cop filtrat, s'adquireix el senyal amb un ADC a una freqüència de mostreig adequada.

Es processa el senyal digital amb el processador, i es reconstrueix el senyal amb el DAC. Per últim, es torna a filtrar el senyal per eliminar soroll. El processador ha de realitzar les operacions en l'interval del temps de mostreig.

Per a dissenyar el pedal s'ha seguit una estructura similar:

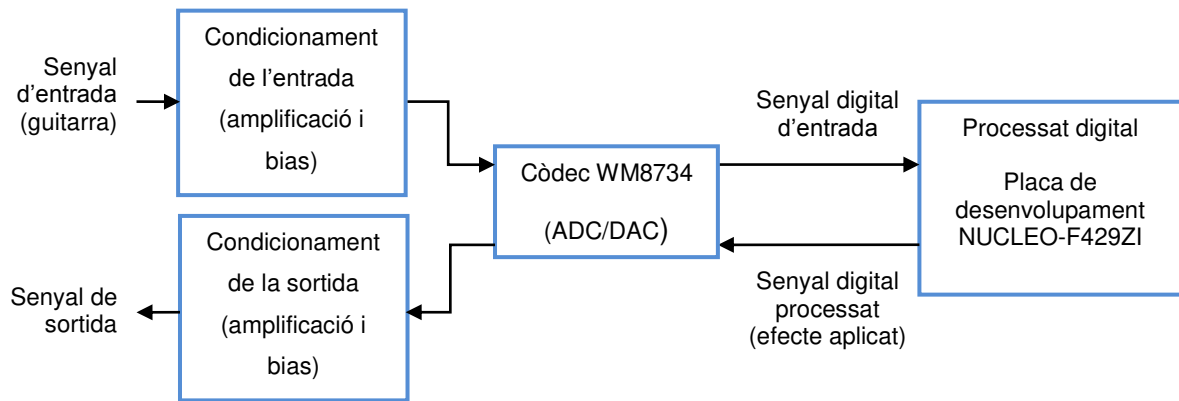


Figura 6. Estructura del projecte

El senyal d'entrada passarà per una etapa de condicionament, per tal d'adaptar el senyal a l'entrada del ADC; un cop condicionada, s'adquirirà el senyal amb el còdec WM8734 i aquest senyal s'enviarà via protocol I2S cap a una placa de desenvolupament, en concret la NUCLEO-F429ZI. En aquesta etapa aplicarem la modificació necessària al senyal per aconseguir l'efecte d'àudio desitjat.

Un cop modificada, es reconstrueix el senyal modificat amb el còdec, i es condiona aquest per a posteriorment connectar la sortida del pedal a un amplificador. En els següents capítols s'explicarà la selecció de tots els dispositius presents en el projecte.

3. SELECCIÓ DE COMPONENTS

En aquest capítol es pretén explicar de forma exhaustiva la selecció dels components essencials del projecte: la placa de desenvolupament per programar els efectes, el còdec per adquirir i regenerar posteriorment el senyal de la guitarra i l'amplificador operacional.

3.1. Placa de desenvolupament NUCLEO-F429ZI

S'ha escollit utilitzar el microcontrolador STM32F429ZI, inclòs a la placa de desenvolupament NUCLEO-F429ZI. S'ha utilitzat la placa directament, degut a que aquest microcontrolador té una gran quantitat de pins (en total 144), cosa que dificulta el soldatge del xip en una placa. A més, utilitzant aquesta placa és més fàcil definir els perifèrics a utilitzar, degut a que els pins d'aquest microcontrolador són programables.

A continuació es farà una comparativa entre microcontroladors i DSP's, per tal de poder justificar la utilització del microcontrolador esmentat anteriorment. Seguidament s'explicaran les característiques de la placa.

3.1.1. Comparació de dispositius

Existeixen a l'actualitat molts dispositius que poden dur a terme el processament digital del senyal; el que els diferencia serà el temps que tarden en executar el codi desitjat. Es pot realitzar amb microcontroladors o els xips anomenats DSP.

Els DSP són els xips per excel·lència alhora de processar digitalment qualsevol senyal. Poden realitzar operacions matemàtiques a gran velocitat; la operació matemàtica més utilitzada és la suma de productes, que és molt freqüent per exemple en filtres digitals i en transformades de Fourier (són aplicacions molt utilitzades en el processament digital del senyal).

Els DSP poden realitzar ambdues operacions (suma i multiplicació) en un cicle de rellotge, anomenant aquestes instruccions com a instruccions MAC. Gràcies a això, es poden processar senyals en temps real; amb microcontroladors genèrics, aquestes operacions tarden varis cicles de rellotge, per el qual no es pot processar en temps real.

Els DSP també permeten emmagatzemar els resultats de les sumes de productes sense arribar al “overflow”. Per exemple, un DSP amb dades de 16 bits, realitzant multiplicacions pot donar com a resultat una paraula de 40 bits.

Normalment els DSP tenen una arquitectura Harvard, el que permet tenir un bus d'adreces i un bus de dades, el que permet accedir més ràpidament a cada memòria. Una característica de la rapidesa dels DSP's és la capacitat de realitzar lectures, escriptures i executar instruccions en un cicle de rellotge (“pipeline”).

Pipelined (*Most conventional DSP processors*)



Figura 7. Representació del pipeline en DSP's

Aquestes característiques serien les ideals per aplicacions a temps real. Un inconvenient dels DSP's és que per utilitzar la màxima rapidesa dels xips, s'hauria de fer programació tant en un llenguatge d'alt nivell (com és el C) com en ensamblador (treballar amb registres interns de les memòries).

El pedal de guitarra ha de poder ser utilitzat tant per programadors com per no programadors, per tant es va descartar la idea d'utilitzar un DSP, i fer servir un xip amb un entorn entenedor i una programació només d'alt nivell. És més, les plaques de desenvolupament dels DSP's són excessivament cares, i els xips presenten un encapsulat difícil de soldar.

La alternativa als DSP's són els microcontroladors d'alt rendiment. En concret s'ha buscat la opció més semblant a un DSP dins de la gamma de microcontroladors de la marca STMicroelectronics. Els microcontroladors més comercials, com per exemple l'Arduino, no tenen la rapidesa adequada per realitzar filtres FIR, transformades de Fourier o filtres digitals. Per exemple, la velocitat màxima d'un Arduino UNO es troba als 16MHz, mentre que les velocitats dels DSP's poden arribar als 400MHz.

La gamma de microcontroladors de STMicroelectronics tenen en el seu interior un processador tipus Cortex-M de la marca ARM. En concret, el microcontrolador utilitzat té un

processador Cortex-M4 de 32 bits, amb arquitectura RISC. La velocitat màxima del processador és de 180MHz. Aquest processador permet la utilització d'operacions tant de 16 bits com 32 bits.

Aquest també permet treballar amb la capa de CMSIS creada per ARM. CMSIS és una capa d'abstracció que s'utilitza tant en processadors com el Cortex-M o en dispositius PSocs. Aquesta capa ofereix una interfície de software que permet la reducció del codi.

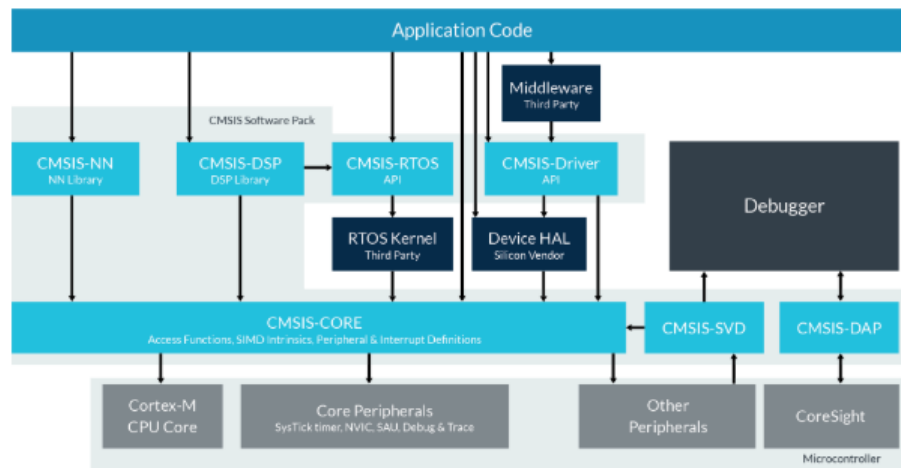


Figura 8. Capes d'abstracció CMSIS

Dins d'aquesta capa, es troben les funcions CMSIS DSP, una llibreria amb un total de 61 funcions matemàtiques; entre elles per a la realització de filtres o transformades de Fourier.

Aquestes funcions s'han fet servir en el projecte per tal d'implementar un codi eficient per a certs efectes implementats en el projecte. També hi han funcions per controlar perifèrics, o funcions per aplicacions a temps real (RTOS). En la figura es pot veure la reducció de cicles de rellotge al utilitzar la llibreria CMSIS DSP amb transformades de Fourier, en comptes de programar-les manualment.

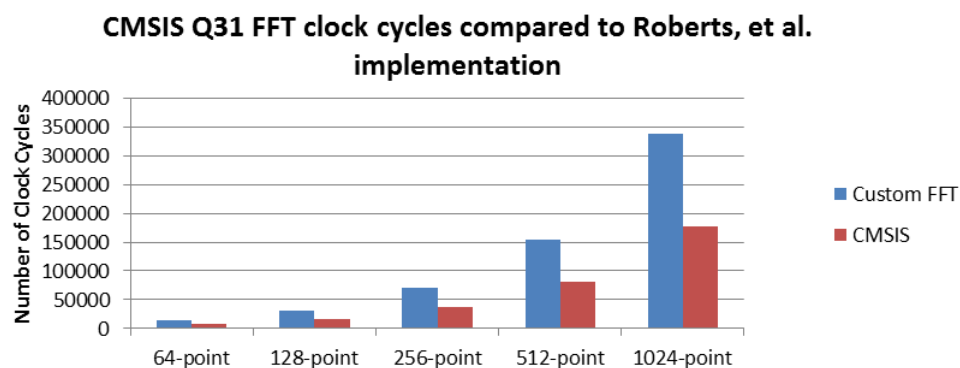


Figura 9. Comparació de rapidesa entre la llibreria DSP i una FFT customitzada

El processador Cortex-M4 no és el que disposa del rendiment més alt d'aquesta gamma; per sobre es troba el Cortex-M7. La única diferència és que en el Cortex-M7, els bucles no tenen teòricament cap cost d'execució. Aquesta característica fa al Cortex-M7 un processador amb un rendiment equivalent als DSP's. Al no tenir aquest cost d'execució, les operacions es realitzen amb més rapidesa que amb el Cortex-M4.

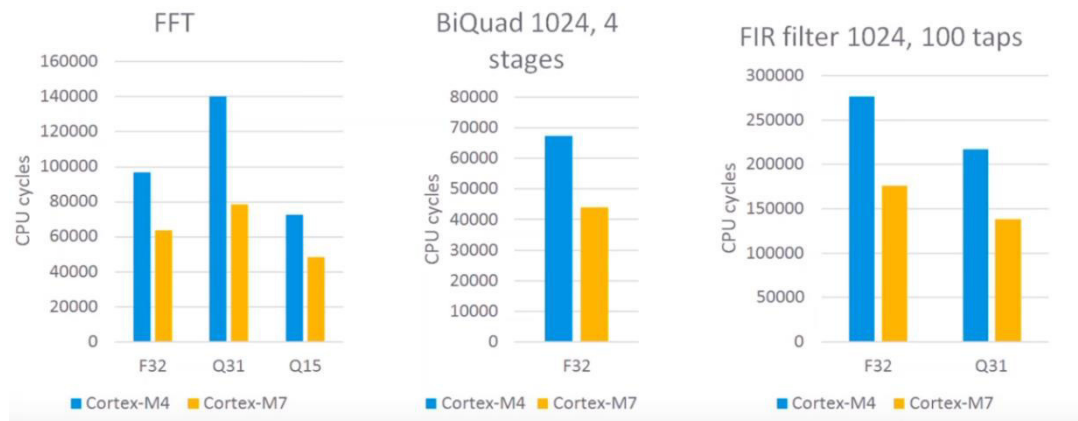


Figura 10. Comparació de cicles de rellotge amb funcions DSP entre els dos processadors

Com es pot observar a la figura, el processador Cortex-M7 realitza transformades de Fourier i filtres en menys cicles de rellotge que els Cortex-M4.

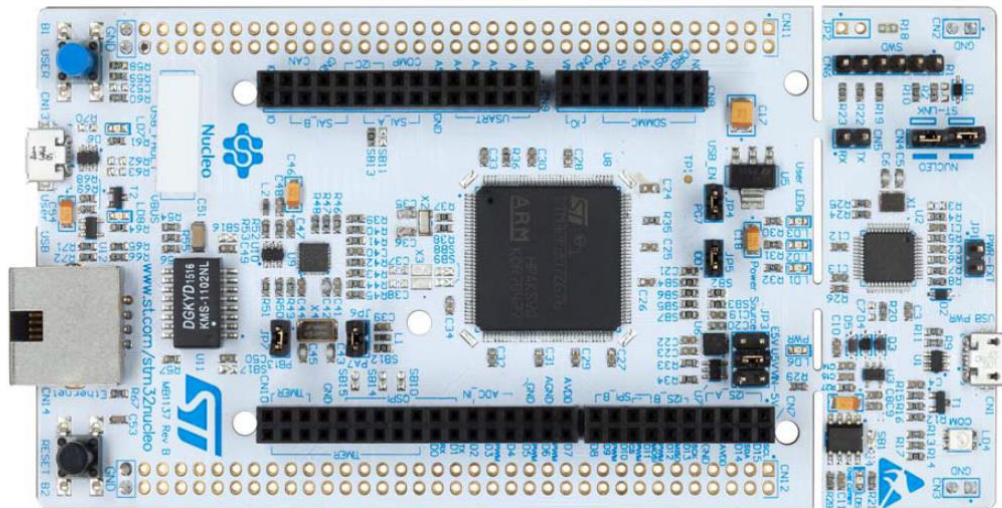
La raó per la qual s'ha escollit una placa de desenvolupament amb el Cortex-M4, és la possibilitat d'utilitzar el protocol I2S (protocol utilitzat per la transferència d'àudio entre microcontrolador i còdec) en mode full-duplex, característica que les plaques amb els microcontroladors amb processador Cortex-M7 no tenen.

S'ha d'utilitzar aquest tipus de transferència, ja que el còdec escollit treballa només en full-duplex (transmetre i rebre alhora).

A efectes pràctics, la diferència de temps de processament entre aquests Cortex-M són d'uns quants microsegons, per tant la diferència no impossibilita un bon rendiment del pedal amb el Cortex-M4.

3.1.2. Descripció de la placa de desenvolupament

La placa, com s'ha esmentat anteriorment, és la NUCLEO-F429ZI, de la marca STMicroelectronics. La placa ofereix un alt rendiment amb un baix consum d'energia.



. Figura 11. Placa de desenvolupament NUCLEO-F429ZI

.El microcontrolador integrat és el model STM32F429ZI; aquest té una velocitat màxima de 180MHz. Aquest conté memòries d'accés ràpid, com per exemple una memòria flash de 512Kbytes, 256Kbytes de SRAM i un gran rang de perifèrics connectats a busos diferents.

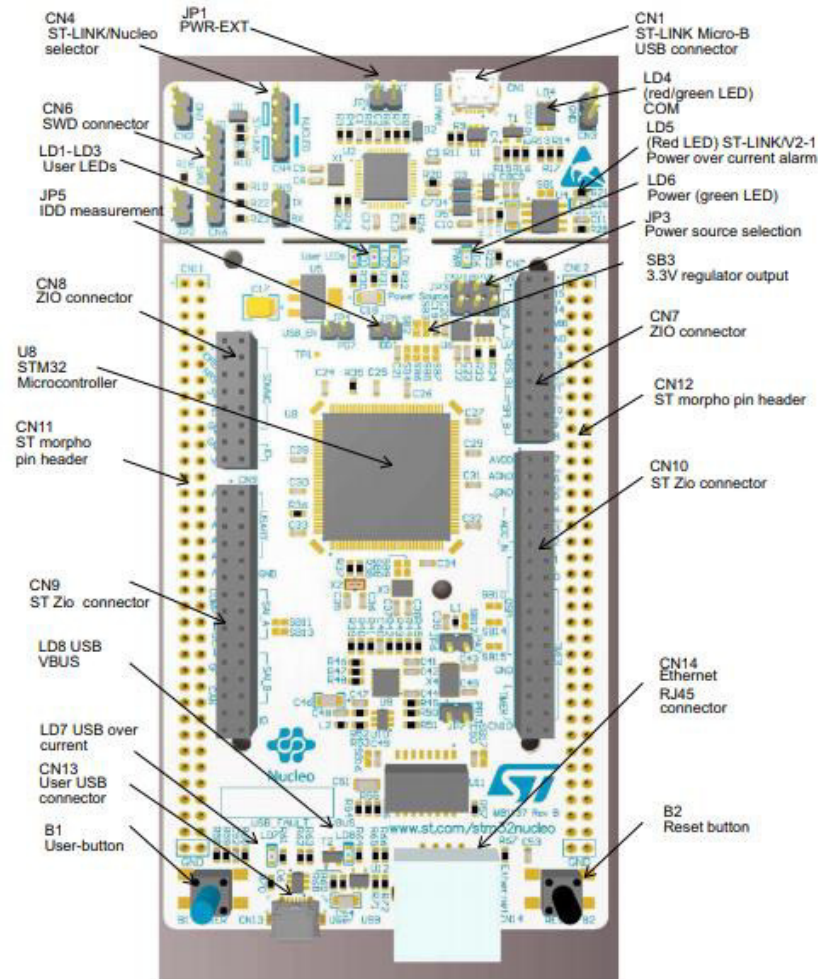
El microcontrolador té un total de 144 pins, amb els quals es poden programar multitud de perifèrics, té integrats ADC's i DAC's de 12 bits, així com pot treballar amb els protocols SPI, I2C, USART i SAI (I2S).

Els pins són configurables; cada pin es pot configurar per a diverses funcions diferents, el que permet una major flexibilitat per a desenvolupar plaques electròniques (GPIO's).

Els pins es s'han configurat a partir del software de STCubeMX, que permet generar codi per a la configuració dels pins, com per la inicialització de perifèrics. El funcionament i configuració d'aquest software es troba explicat en l'annex A.

La placa disposa dels elements mostrats a la figura 12. Disposa d'un botó de reset, un botó per a l'usuari, LED's d'usuari, un connector USB per programar el microcontrolador, així com connectors per tal de connectar pins. Alguns pins són compatibles amb els pins d'Arduino (tenen la mateixa posició).

La placa també permet treballar amb Ethernet; en aquest projecte es deshabilitarà l'Ethernet degut a que no s'utilitza.



. Figura 12. Components integrats a la placa NUCLEO-F429ZI

A continuació s'explica la selecció dels components integrats en el circuit del projecte.

3.2. Còdec WM8734

S'ha considerat afegir un còdec extern al microprocessador; la resolució dels ADC's i DAC's del STM32F429ZI no són les adequades per a obtenir un so de qualitat a la sortida (12 bits). Per a aplicacions d'àudio, la resolució mínima per obtenir un so de qualitat a la sortida és de 16 bits.

Una resolució de 16 bits seria suficient per a aplicacions d'àudio generals; per a aplicacions com pot ser la gravació de música en un estudi, la resolució és de 24 bits.

Normalment en aplicacions d'àudio no es fa servir un ADC i un DAC per separat, si no que s'utilitza un còdec, que simplement junta ADC i DAC en un únic component.

Hi han còdecs que permeten fer més enllà que la conversió digital-analògica (i viceversa) del senyal, podent programar filtres i també algunes funcions pròpies del processament digital del senyal.

Els còdecs d'àudio són tots normalment de tipus sigma-delta. La principal característica dels convertidors sigma-delta és que utilitzen la tècnica de mostrejar el senyal a una freqüència molt superior a la freqüència que serviria a partir del teorema de Nyquist (la freqüència mostreig ha de ser mínim dues vegades més gran que la freqüència del senyal). La freqüència de mostreig del senyal (f_s) es veu multiplicada per un factor K .

A partir d'aquesta tècnica, el que permet és enviar el soroll a una banda de freqüència més elevada que la banda del senyal. És més, els convertidors sigma-delta tenen un modulador integrat, que permet modificar el soroll per treure la màxima quantitat de soroll de la banda del senyal. Tots aquests paràmetres permeten fer servir la resolució màxima del convertidor.

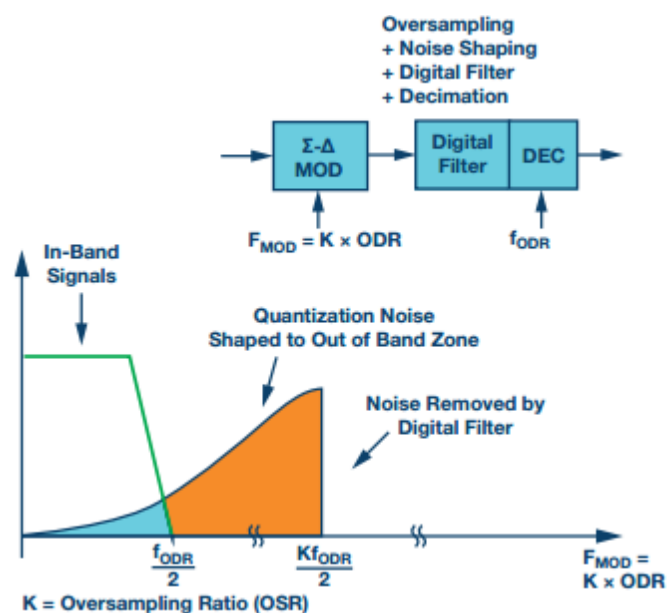


Figura 13. Extracció de soroll realitzat per els convertidors sigma-delta

El còdec escollit és el WM8734 de la marca Wolfson Microelectronics, un còdec sigma-delta específic per a àudio.

A part de la gran resolució i qualitat d'àudio que pot oferir, aquest còdec és el més simple de la marca a nivell intern; té les opcions de mutejar el senyal de les entrades, d'aplicar guanys a la senyal d'entrada (abans d'entrar al ADC) i la opció d'implementar un filtre pas alt.

Les característiques són les següents:

Paràmetre	Valor
Canals d'entrada	2
Canals de sortida	2
Nombre de bits	16-20-24-32
SNR ADC (dB)	94
SNR DAC (dB)	100
Freqüència de mostreig (kHz)	8-96
Entrades/sortides analògiques	No balancejades

Taula 3. Especificacions del WM8734

Les paraules de sortida del senyal digital poden ser de 16 a 32 bits. La configuració dels registres interns es pot fer mitjançant I2C o SPI, mentre que la transferència de dades es fa a partir del protocol I2S.

La tensió d'alimentació d'aquest còdec és de 3,3V.

3.2.1. Configuració I2C

El còdec WM8734 té uns registres interns que permeten configurar el còdec depenent de les necessitats de cada projecte. En total té vuit registres, els quals es poden configurar mitjançant SPI o I2C. En el cas del nostre projecte, s'ha escollit configurar els registres via I2C.

Per tal de poder escriure els registres de manera adequada, s'han de tenir en compte diverses especificacions definides en el full d'especificacions del còdec.

L'adreça del còdec és de set bits, i es determina en funció de l'estat del pin del còdec anomenat CSB. Al tenir el pin CSB a massa, l'adreça serà 0011010. Darrera de l'adreça, es col·loca el bit de llegir/escriure. En el nostre cas, només duren a terme accions d'escriure en el còdec; per tant segons el full de característiques, el bit següent és el 0. Per tant el conjunt d'adreça més el bit d'escriptura quedarà definit com a 34h.

Un cop el màster (en aquest cas el microcontrolador) detecta el còdec amb l'adreça adequada, es poden començar a enviar dades. S'hauran d'enviar en paraules de 16 bits, en els quals els 7 bits més significatius corresponen a l'adreça del registre (B[15:9]), i els 9 bits

menys significatius corresponen a les dades (B[8:0]). Degut a que l'enviament de dades en I2C es realitza per bytes, el bit més significatiu de les dades passarà a enviar-se amb els bits d'adreça dels registres.

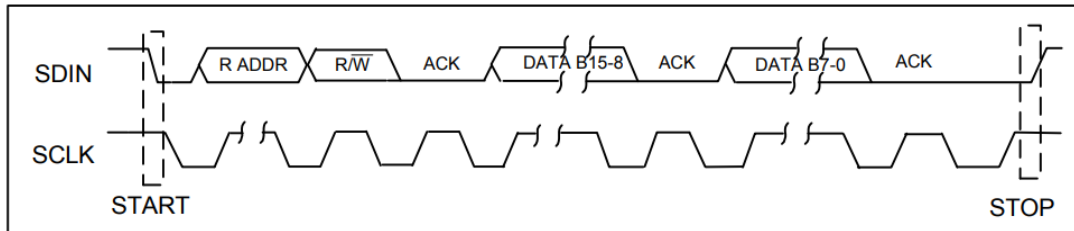


Figura 14. Comunicació I2C amb el còdec

Els registres de configuració del còdec amb les corresponents adreces són els següents:

Adreça	Nom registre
0000000	Left Line In
0000001	Right Line In
0000100	Analogue Audio Path control
0000101	Digital Audio Path control
0000110	Power Down Control
0000111	Digital Audio Interface Format
0001000	Sampling Control
0001001	Active Control

Taula 4. Registres configurables del WM8734

Els dos primers registres controlen la configuració de les entrades. Per a l'entrada esquerra, per on entrarem el senyal condicionat de la guitarra, es pot canviar el guany; en aquest cas s'ha regulat a 0dB, per tal de que la conversió de 0 a 3,3 V es mantingui amb la mateixa proporció. L'entrada dreta es des-habilita, ja que no la fem servir.

En el registre del control analògic només s'ha d'activar un bit, que determina que activem la sortida de dades per el DAC.

El registre de control digital serveix per activar o desactivar entrades i sortides. Es configura per tal de que tot estigui habilitat. Quan es produeixi la interrupció per "bypass", s'activa el bit per des-habilitar les sortides, per tal d'evitar sorolls.

El registre d'interfície digital serveix per definir les característiques del I2S. Es configura el còdec com a esclau, es determina que el protocol serà I2S tipus estàndard i que les dades seran de 16 bits.

En el registre de configuració de mostreig serveix per definir la freqüència de mostreig, que serà a 48kHz.

Finalment, l'últim registre serveix per activar el còdec, que en estat inicial es troba en "standby".

Nom registre	Adreça (hex)	Valor (hex)
Left Line In	00h	17h
Right Line In	02h	00h
Analogue Audio Path control	08h	10h
Digital Audio Path control	0Ah	01h
Power Down Control	0Ch	00h
Digital Audio Interface Format	0Eh	02h
Sampling Control	10h	00h
Active Control	12h	01h

Taula 5. Valors d'adreces i registres en hexadecimal

Els registres i les adreces es programaran amb els valors especificats a la taula.

3.2.2. Configuració I2S

El protocol I2S s'utilitzarà per a la transferència de dades entre el còdec i el microcontrolador. És el més utilitzat actualment per tots els dispositius d'àudio, permetent l'adquisició de dades des d'un ADC i enviar àudio cap a altaveus o auriculars. Aquest protocol ofereix molta flexibilitat, com permetre trames de dades des de 8 bits a 32 bits, permetre canviar la configuració a mono o estèreo, transferir i rebre dades en el mateix moment, entre d'altres.

El funcionament del I2S és semblant al SPI. Consta de tres pins principals: un pin de rellotge (SCK), un pin de selecció de canal (WS) i un pin de dades (SD). El dispositiu que es configuri com a mestre, ha de generar el senyal de rellotge i el de selecció de canal. Per a configuracions full-dúplex, existeix un altre pin addicional de dades.

Un canvi en el pin WS determina si estem enviant o rebent dades del canal esquerra (WS a 0) o del canal dret (WS a 1).

Aquests senyals poden treballar a diferents freqüències; la freqüència del WS la determina la freqüència de mostreig del senyal (per exemple 44.1kHz, 48kHz...). En funció d'aquesta freqüència, i el nombre de bits al qual programem la trama de dades, obtindrem la freqüència del rellotge. Per exemple, si tenim un ADC que treballa a 48kHz i té una resolució de 16 bits, significa que per a cada canvi de WS hem de transmetre 16 bits, per tant la freqüència del SCK serà de 1,53MHz.

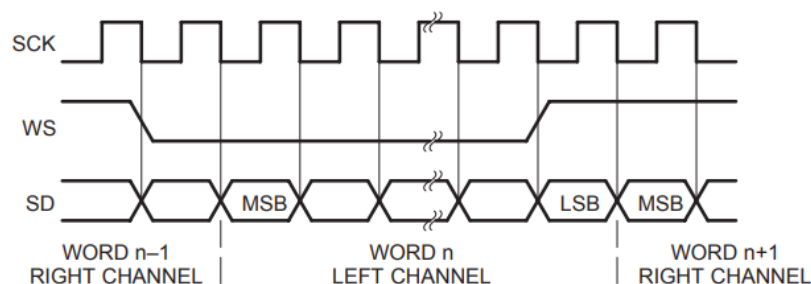


Figura 15. Senyals del protocol I2S

En el cas de l'esclau, el còdec necessita proveir-se d'un rellotge addicional per tal de realitzar el sobremostreig, o el mateix microcontrolador pot generar aquesta senyal. En el nostre cas generem aquest senyal des del microcontrolador.

Com s'ha esmentat anteriorment, s'ha configurat el còdec per tal de treballar a 48kHz i a 16 bits. S'ha escollit treballar a una resolució de 16 bits, degut a que és una resolució més que suficient per a la nostra aplicació, així com la freqüència de mostreig.

3.3. Amplificador operacional OPA725

En aquest apartat s'intenta explicar la selecció d'aquest amplificador operacional per tal de realitzar el condicionament tant de l'entrada com de la sortida. Les característiques principals d'aquest amplificador és que és de baix soroll i alta velocitat. Totes les característiques es troben especificades les taules 6 i 7.

Paràmetre	Valor
Ample de banda (MHz)	20
Soroll (nV/ $\sqrt{\text{Hz}}$)	6

Taula 6. Característiques del OPA725

Paràmetre	Valor
Slew Rate (V/us)	30
Tipus de sortida	Rail-to-Rail
Rang de tensió d'alimentació (V)	4-12
THD+N a 1kHz (%)	0,0003

Taula 7. Característiques del OPA725

Entre les seves aplicacions, podem veure que es pot utilitzar per a filtres actius, àudio i testeig d'equipament.

Per al nostre projecte, les característiques més importants per tal de que l'amplificador funcioni bé dins l'aplicació són el soroll, la distorsió, l'ample de banda, l'alimentació i l'slew rate.

Els amplificadors operacionals que s'utilitzen per àudio han de provocar el menor soroll possible, per tal de no afegir soroll al senyal, que després es pugui mostrejar per el ADC. Aquest operacional té un soroll així com una distorsió molt baixa ($6\text{nV}/\sqrt{\text{Hz}}$, 0,0003% THD+N), cosa que el fa ideal per condicionar la senyal que properament haurà de ser adquirida per el còdec.

L'ample de banda també és correcte per a l'aplicació, ja que les senyals d'àudio arriben fins als 20kHz, i aquest amplificador té un producte de guany i ample de banda de 20MHz.

L'slew rate és molt elevat (30V/us). Per a calcular l'slew rate per a la nostra aplicació es fa de la manera següent:

$$\text{SR} = 2 \cdot \pi \cdot f \cdot V \quad (\text{Eq.1})$$

On:

SR: slew rate (V/ μs)

f: freqüència màxima del senyal (Hz)

V: tensió màxima de pic del senyal (V)

S'ha considerat una tensió màxima de 9V, i una freqüència màxima de 20kHz. L'slew rate mínim que ha de tenir l'operacional és de 1,13V/ μs , per tant l'amplificador escollit compleix.

La condició de que l'amplificador sigui rail-to-rail és necessària, degut a que es vol aprofitar el màxim de sortida del pedal que surti cap a l'amplificador.

La font d'alimentació que s'utilitza és de 9V, que és la font estàndard que s'utilitza en els pedals de guitarra. Aquest amplificador, es pot alimentar de 4 a 12V, per tant en temes d'alimentació aquest operacional es pot connectar a la font de 9V.

4. DISSENY I FUNCIONAMENT DEL CIRCUIT

En aquest capítol s'explicarà el disseny adoptat per a cada part , així com la justificació de la utilització del disseny. Com s'ha esmentat en el capítol 1, el disseny s'ha separat per diferents etapes: condicionament de l'entrada, conversió digital-analògica (i viceversa), condicionament de la sortida i visualització i control.

Per tal de poder connectar la guitarra en el pedal, i posteriorment connectar el pedal a un amplificador, s'han utilitzat connectors jack per a PCB de 6,35mm de diàmetre.

La particularitat d'aquests connectors és que al desconnectar el cable, permeten la connexió a massa del connector per tal d'evitar l'entrada d'interferències al circuit.



Figura 16. Connector Jack tipus mono

El connector de l'entrada serà mono, degut a que la senyal de la guitarra només s'entra per un canal, mentre que la sortida es farà amb un connector estèreo; el connector estèreo permet treure senyal per el canal dret, necessari en alguns tipus d'efectes com els de retard o modulació del senyal.

Per a l'alimentació del pedal, s'utilitzarà un connector jack de 5,5 mm específic per a connectar fonts d'alimentació. La font d'alimentació tindrà una tensió de 9V; és el valor de tensió més utilitzat per a pedals de guitarra.

Aquest connector es connectarà a l'entrada d'alimentació del microcontrolador, que és el pin d'alimentació externa. Aquest pin aguanta com a màxim una intensitat de 800mA.

Les tensions de 5V i 3,3V s'extrauran dels corresponents pins de la placa de desenvolupament.

4.1. Condicionament de l'entrada

El condicionament de l'entrada es basa en la correcta adequació del senyal de la guitarra per tal de que sigui adquirit per el ADC del còdec. El disseny també ha de permetre filtrar senyals de DC que puguin sumar soroll al circuit, la protecció de l'entrada del ADC contra sobretensions i la preservació del senyal de guitarra.

El còdec disposa de dues entrades, però per a la guitarra només s'utilitzarà una, degut a que és un senyal tipus mono. En el full de característiques del còdec es pot observar que les entrades són no balancejades i unipolars, i aquestes van de 0 a 3,3V (amb el còdec alimentat a 3,3V). Però en realitat, les entrades poden tenir un valor com a màxim d'1V eficaç dins d'aquest marge entre 0 i 3,3V. El que fa el ADC és centrar aquest senyal a una tensió d'1,65V ($AVDD/2$).

Aquest tipus de valors dels senyals d'entrada és molt comú en els equips d'àudio, en els quals alguns es troben estandarditzats, com per el nivell anomenat de 0dB, el qual equival a 1Vrms. Aquests valors s'han estandarditzat per permetre la transferència d'àudio entre dispositius específics per a àudio, com poden ser amplificadors, mescladors o fins i tot televisors.

Per tant, el condicionament d'entrada ha de permetre l'amplificació del senyal fins a 1Vrms. Com s'ha explicat en el capítol, el senyal de sortida de la guitarra varia entre els 90mVpp i 1Vpp.

Per tal de veure l'amplificació necessària en els dos casos, convertim els valors de pic a pic a només de pic i també a valors eficaços.

Vp-p (V)	Vp (V)	Vrms (V)
0,09	0,045	0,0318
1,00	0,500	0,3535
2,82	1,414	1,0000

Taula 8. Valors de tensió màxims i mínims a obtenir

En qualsevol de les conversions anteriors, la ràtio de guany a dissenyar serà la mateixa.

Degut a que la guitarra, tal com s'ha explicat en el capítol, pot variar en aquest rang de tensions depenent del tipus de pastilles i circuit intern, s'ha decidit optar per realitzar una amplificació amb guany variable, gràcies a l'ajuda d'un potenciòmetre, que farà la funció de resistència variable.

L'últim paràmetre a considerar és la addició de tensió per convertir la senyal alterna de la guitarra a una senyal entre 0 i 3,3V. Per tal de centrar la senyal s'ha utilitzat un regulador a 1,6V.

S'utilitza l'amplificador operacional de Texas Instruments, el OPA725, amb el qual es dissenya amb la topologia d'un amplificador no inversor, ja que el senyal no es vol invertir.

El condicionament d'entrada és el següent:

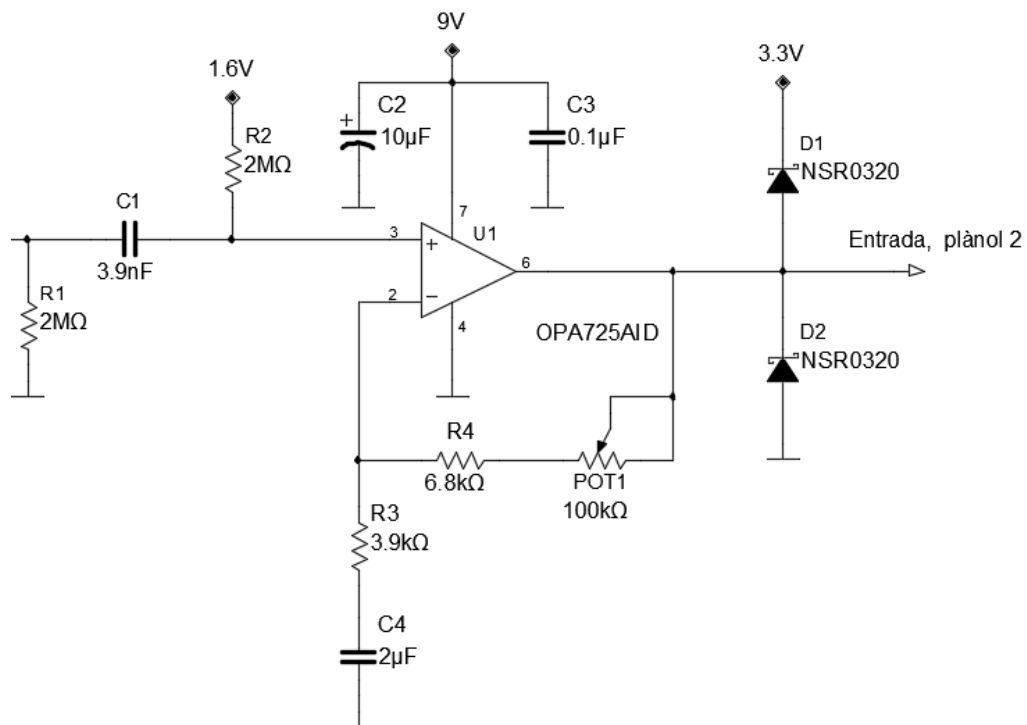


Figura 17. Circuit de l'etapa d'entrada

Amb aquesta configuració escollida, la tensió d'1,6V es veurà també a la sortida. L'equació de guany d'aquest circuit és el següent:

$$V_{out} = V_{in} * G + V_{bias} \quad (Eq.2)$$

On:

V_{out} : tensió de sortida (V)

V_{in} : tensió d'entrada (V)

G: valor del guany (adimensional)

V_{bias} : tensió de centrat (V)

On el guany ve determinat per la fórmula següent:

$$G = 1 + \frac{R_4 + POT_1}{R_3} \quad (Eq.3)$$

On:

G: valor del guany (adimensional)

R_3, R_4, POT_1 : valor de la resistències (Ω)

La resistència R_1 permet la preservació del senyal de la guitarra; degut al concepte de igualació d'impedàncies, el circuit de condicionament ha de tenir una impedància d'entrada superior a la impedància de la guitarra. Com s'ha vist anteriorment, la sortida de la guitarra presenta una resistència de sortida d'entre 10 i 15k Ω . Per tant la impedància d'entrada del circuit ha de ser major a 15k Ω .

Normalment en equipament d'àudio, la impedància d'entrada sol rondar en els 20 k Ω , pro en aplicacions com els pedals de guitarra, la impedància ronda en valors de M Ω . Per tant, s'ha escollit un valor per a la resistència de 2M Ω .

El condensador C_1 , junt amb la resistència R_2 , formen un filtre passiu pas alt de primer ordre. Aquest filtre es troba configurat per tallar freqüències inferiors a 20Hz. La resistència R_2 s'ha determinat a 2M Ω , per tal d'evitar la demanda de corrent del regulador. El valor del condensador C_1 per a una freqüència de 20Hz és de 3,97nF, que normalitzat serien uns 3,9nF.

El valor del condensador s'ha establert a partir de la fórmula següent:

$$f_o = \frac{1}{2 \cdot \pi \cdot R \cdot C} \quad (Eq.4)$$

On:

f_0 : freqüència de tall del filtre (Hz)

R: valor de la resistència (Ω)

C: Valor del condensador (F)

Per tal de que el circuit funcioni correctament, i obtinguem l'equació de sortida correctament, el valor de la freqüència de tall entre R_3 i C_4 haurà de ser el mateix que per a C_1 i R_2 .

Abans de calcular tots els valors anteriors, s'ha calculat la resistència R_3 , la qual pertany al guany variable de l'amplificador no inversor.

Els guanys de càlcul a obtenir surten de 2,82 per a un senyal de 90mVp-p, mentre que per a senyals d'1Vp-p s'obté un valor de 31,42.

Amb aquests valors de guany a obtenir, i fixant el valor del potenciòmetre POT_1 , es pot realitzar un sistema de dues equacions amb dues incògnites (R_4 i R_3). Per a l'equació del guany màxim s'ha tingut en compte el valor de POT_1 (s'ha determinat a 100k Ω) mentre que en l'equació del guany mínim, s'ha considerat amb un valor de 0 Ω .

Els valors de les resistències obtingudes no són resistències normalitzades, per el qual el valor dels guanys varia lleugerament. Els resultats són els següents:

Paràmetre	Càlcul	Real
R_3 (k Ω)	3,49	3,90
R_4 (k Ω)	6,39	6,80
Guany mínim	2,82	2,74
Guany màxim	31,42	28,38

Taula 9. Càlculs de guany amb resistències normalitzades

Per tant, amb resistències normalitzades, amb una senyal de 90mVp-p i aplicant el guany màxim, obtindrem un valor de 0.91Vrms, mentre que amb una senyal d'1Vp-p d'entrada aplicant el guany mínim, obtindrem una sortida de 0,96Vrms.

En el primer cas, comparant amb el màxim de 1Vrms, perdríem 273mV de resolució, mentre que en el segon cas, només perdríem 88mV de resolució.

A partir del valor de la resistència R_3 amb un valor de 3,9 k Ω , obtenim un valor del condensador de 2,04 μ F per a la freqüència de 20Hz, que normalitzat serien 2 μ F.

A l'alimentació positiva de l'amplificador s'ha col·locat un condensador de desacoblament, de tipus ceràmic amb un valor de 0,1 μ F; aquest condensador s'utilitza per eliminar les components de soroll que provenguin de l'alimentació. També s'ha connectat un condensador de desacoblament electrolític de 10 μ F, que actua com a reserva per a càrregues instantànies del circuit.

Al tenir l'amplificador operacional connectat a una font de 9V, es poden produir sobretensions que puguin danyar a l'entrada del còdec, ja que la tensió màxima que soporta, segons el seu full de característiques, és de 3,63V. Per tal d'evitar que es produeixin sobretensions a l'entrada, s'ha decidit col·locar un parell de díodes Schottky a l'entrada, entre la referència de 3,3V i massa.

Es col·loquen díodes Schottky ja que el seu voltatge en polarització directe és més baix que els díodes normals (per exemple, si un díode comú té una tensió llindar de 0,7V, la de un díode Schottky és de 0,2V). La tensió llindar màxima que es podria tenir seria de 300mV en el nostre cas, segons el full de característiques del còdec.

S'ha escollit utilitzar els díodes NSR0320, que presenten una tensió llindar de 240mV a una intensitat de 10mA. Poden suportar una tensió inversa de 23V de pic, i tenen un consum de 200mW a 25°C. Per tant, simulant el circuit, s'observa que la màxima tensió que es produeix és de 3,59V amb una intensitat de 12,7mA. Per tant aquest díode limita correctament la tensió que pot suportar l'entrada analògica del còdec.

S'ha decidit no filtrar el senyal de l'entrada abans d'entrar al còdec, degut que en el full de característiques s'especifica que internament el còdec té integrat un filtre pas baix específic per a àudio abans d'entrar al ADC.

4.2. Condicionament de la sortida

El condicionament de la sortida també tindrà la mateixa configuració que el condicionament de l'entrada; la senyal de sortida del còdec serà alterna, i la tensió de sortida final haurà de ser entre 0 i 9V, per tal de poder alimentar el pedal amb una font de 9V. Per tant per tenir la senyal centrada s'afegirà una tensió de referència de 4,5V.

En el cas de la sortida, el circuit es repeteix dues vegades, degut a que per a efectes que impliquen retard o modulació del senyal, es treu senyal tant pel canal esquerre com per el dret del còdec.

El guany variable ja no és necessari a l'etapa de la sortida ja que ja s'ha regulat el guany a l'entrada del còdec.

A la sortida del còdec tindrem com a màxim una senyal d'1Vrms, que seria la senyal d'entrada màxima que entrem. Per tant apliquem la mateixa metodologia que a l'entrada per calcular els valors de resistències i condensadors: es defineix la condició de freqüència, es calcula el valor de les resistències del guany deixant una de les resistències amb valor fixe, i finalment es calcula el valor del condensador que es troba juntament amb la resistència connectat a l'entrada no inversora.

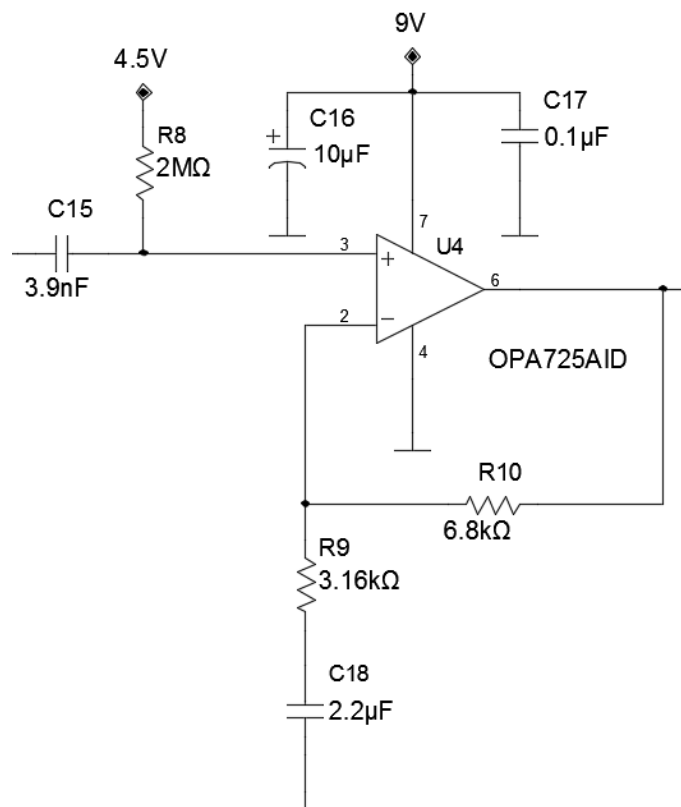


Figura 18. Circuit de l'etapa de sortida

La condició de freqüència serà la mateixa que en l'etapa d'entrada: 20Hz. Per tant el valor del filtre pas alt (format per C_{15} i R_8) seran els mateixos (3,9nF i 2MΩ).

El valor del guany de càlcul a obtenir en l'etapa de sortida és de 3,18. L'equació serà la mateixa que per a l'entrada:

$$G = 1 + \frac{R_{10}}{R_9} \quad (\text{Eq.5})$$

On:

G: valor del guany (adimensional)

R_{10} , R_9 : valor de la resistències (Ω)

S'ha determinat, provant diversos valors de R_{10} , el valor de la resistència R_9 i el condensador C_{18} . S'han hagut d'ajustar el valor de R_9 a un valor de resistència normalitzada amb un tant per cent inferior de tolerància, per tenir més exactitud en el valor del guany (3,16 k Ω), així com en el valor del condensador (2,2 μ F).

Els reguladors de tensió utilitzats en l'etapa d'entrada i sortida són el MAX6018 (1,6V), i el MAX6107, que dóna una tensió de sortida de 4,5V. Són reguladors de tensió que presenten.

S'ha preferit utilitzar reguladors de tensió abans que un divisor de tensió, per tal d'evitar les oscil·lacions de tensió que poden provocar les resistències. No necessiten cap condensador de sortida, només se li ha connectat un condensador ceràmic de desacoblament de 0,1 μ F.

4.3. Connexions del còdec

El còdec és la part més sensible del circuit, degut a que té tant senyals analògics com senyals digitals.

En el full de característiques es recomana que es connectin condensadors de desacoblament en cada pin d'alimentació, tal i com surt a la figura 19. El còdec disposa dos pins d'alimentació, un per a les parts digitals i un altre per a les parts analògiques. El pin VMID és el valor del senyal al qual el còdec centra l'entrada.

El full de característiques també recomana posar condensadors de desacoblament en aquest pin. Els condensadors electrolítics de 10 μ F i condensadors ceràmics de 0,1 μ F. Els condensadors que vagin connectats al pin d'alimentació analògica van referenciats a la

massa analògica, al igual que els condensadors connectats al pin VMID. Els condensadors del pin d'alimentació digital aniran al terra digital.

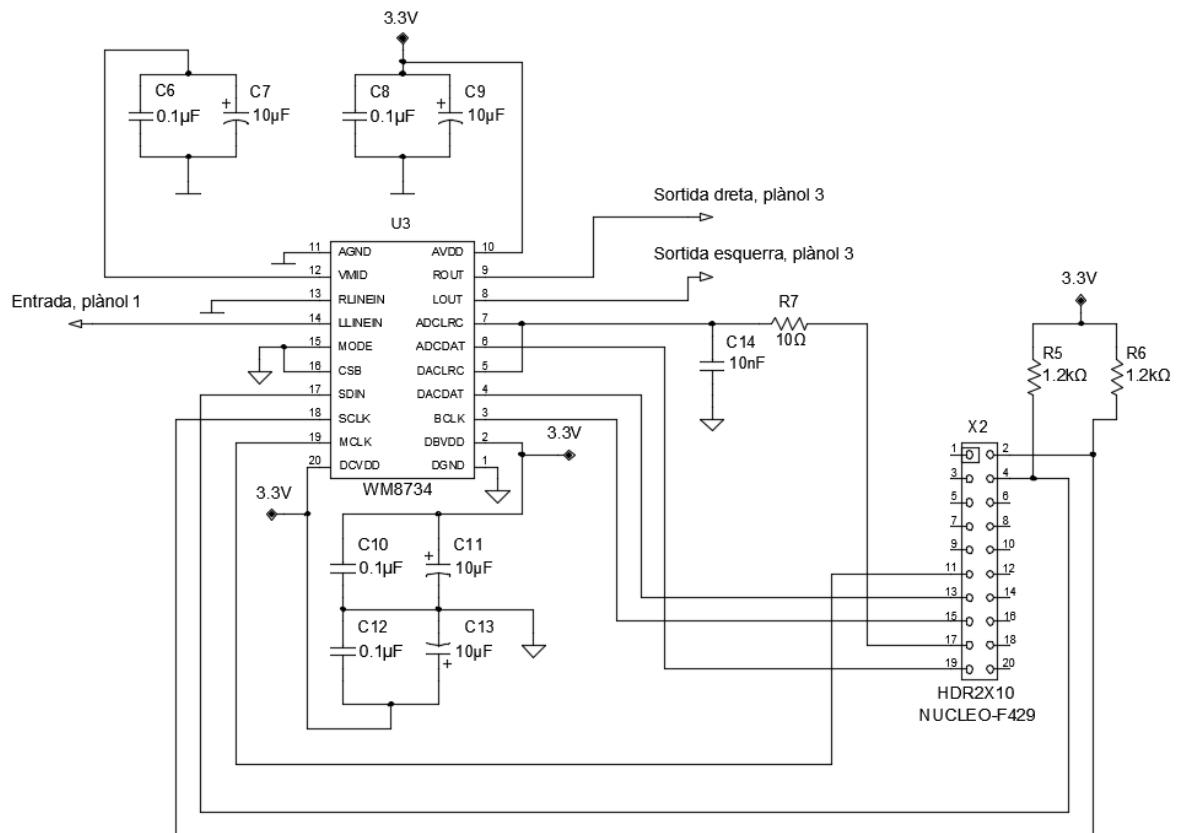


Figura 19. Connexions del còdec

Al utilitzar el protocol I2C per a la configuració de registres, és necessari la col·locació de dues resistències de “pull-up”, per tal de mantenir les entrades a 3,3V i poder començar les transmissions.

Per tal de calcular el valor de les resistències de “pull-up” del I2C, s’han utilitzat les fórmules a l’hora de treballar amb I2C. S’ha calculat la resistència mínima i màxima que es poden posar en el bus. Per tal de calcular la resistència mínima s’utilitza la fórmula:

$$R_{p(min)} = \frac{(V_{cc} - V_{OL})}{I_{OL}} \quad (\text{Eq.6})$$

On:

$R_{p(min)}$: resistència mínima de “pull-up” (Ω)

V_{cc} : tensió d’alimentació (V)

V_{OL} : tensió del bus en estat baix (V)

I_{OL} : intensitat en estat baix (A)

Per a una tensió d'alimentació i una tensió en estat baix de 0,33V, la resistència mínima és de 990Ω.

Per calcular la resistència màxima s'ha de tenir en compte el temps de duració del flanc de pujada dels senyals i la capacítància màxima que pot tenir el bus:

$$R_{p(max)} = \frac{t_r}{0,8473 * C_b} \quad (Eq.7)$$

On:

$R_{p(max)}$: resistència de "pull-up" màxima (Ω)

t_r : temps de duració del flanc de pujada dels senyals (s)

C_b : capacítància màxima del bus (F)

Per a un temps de durada de flanc de 300ns, i una capacítància màxima de 200pF (per a una freqüència de 400kHz), la resistència màxima serà de 1,77kΩ. Per tant entre aquests dos valors, s'han escollit col·locar dues resistències de 1,2kΩ.

El còdec pot funcionar tant en I2C com SPI en la configuració de registres. En el full de característiques es determina que per tal d'utilitzar l'I2C, s'ha de connectar el pin de MODE a 0V. Es connectarà a terra digital, així com el pin CSB, degut a que són pins relacionats amb la interfície de control de software del còdec.

En quant al protocol I2S, tenim sis pins a connectar: MCLK (rellotge proporcionat per el microcontrolador), BCLK (rellotge per a transferència de dades) , ADCLRC i DACLRC (selecció de canal esquerre o dret en el ADC i en el DAC), ADCDAT (sortida de dades des del ADC) i DACDAT (entrada de dades cap al DAC). Cadascun anirà connectat a una entrada configurada específicament per a cada pin cap al microcontrolador.

Els senyals ADCDAT i DACDAT s'han de connectar junts; així en el moment de la transferència, treballem en el mateix moment amb el mateix canal tant en el ADC com en el DAC.

En els protocols, i sobretot en el protocol I2S, s'ha d'obtenir una bona sincronització de cadascun dels senyals, per tal de poder enviar i rebre dades correctament. En aquest protocol en concret, la senyal de selecció de canal i la senyal BCLK no poden caure en el mateix moment; si això es produís, no obtindríem sincronització entre els dos dispositius i tampoc àudio de sortida. En el full de característiques apareix especificat que el senyal ha de caure més tard que el senyal de rellotge.

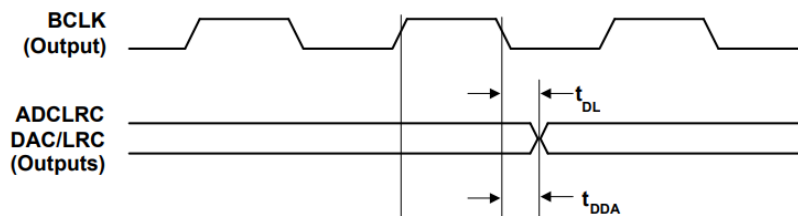


Figura 20. Canvi de canal en la caiguda del senyal BCLK

Per tal d'evitar que caiguin en el mateix moment, el col·loca un circuit RC en el senyal de selecció de canal, per tal de retardar aquest senyal. El temps de retard es calcula de la manera següent:

$$t_{\text{retard}} = R \cdot C \quad (\text{Eq.8})$$

On:

t_{retard} : temps de retard del senyal (s)

R: valor de la resistència (Ω)

C: Valor del condensador (F)

En el full de característiques s'especifica el valor màxim de retard. Per a una senyal de rellotge de 1,53MHz, el màxim retard pot ser de 162ns. Per tant s'ha fet un retard de 100ns amb els valors de R_7 i C_{14} de 10Ω i 10nF .

La sortida de l'etapa d'entrada del circuit es troba connectada en el pin LLINEIN, degut a que les entrades Mono sempre es connecten en el canal esquerre. L'entrada del canal dret, es recomana connectar-la a massa si no és utilitzada. També es silenciarà posteriorment per software.

4.4. Control i visualització

En aquest apartat s'expliquen els diferents elements de control i visualització del pedal.

4.4.1. Polsador 4PDT

En primer lloc, s'ha utilitzat un polsador de peu 4PDT, per tal de realitzar un “bypass” analògic. Aquest mètode s'utilitza en la gran majoria de pedals per tal d'escollir si utilitzar l'efecte, o no utilitzar-ne cap. El concepte s'anomena “true bypass”.



Figura 21. Polsador 4PDT

Els polsadors 4PDT són en realitat quatre polsadors SPDT independents. Cada polsador es pot mantenir en dues posicions, la qual cosa implica que poden estar en dos estats. L'accionador de peu canvia l'estat dels quatre polsadors simultàniament. Tres dels quatre polsadors es connecten a l'entrada i a les sortides analògiques del còdec.

Per tal de controlar l'estat del “bypass”, s'ha connectat una sortida de l'últim polsador a una entrada digital del microcontrolador, que ens permetrà canviar els paràmetres necessaris per passar a l'estat de “bypass”. A una sortida digital se li afegeix un LED per tal de visualitzar millor que el sistema es troba en “bypass”.

En aquesta entrada s'ha col·locat una resistència de “pull-up” de 10k Ω connectada a 3,3V (R_{17}). D'aquesta manera podrem diferenciar l'estat de “bypass” o no. Amb la resistència de “pull-up” protegim l'entrada digital de l'entrada de massa corrent.

4.4.2. Selector d'efectes i potenciòmetres

Per tal de seleccionar l'efecte, s'ha decidit utilitzar un interruptor rotatiu. En concret s'ha utilitzat la placa específica per a aquest interruptor de SparkFun.



Figura 22. Placa amb interruptor rotatiu i resistències

Aquest interruptor té deu posicions i un pol; en el circuit es pot veure que al canviar de posició l'interruptor, la tensió varia en funció de la posició, que permet actuar una resistència o una altre, actuant tot el circuit com un divisor de tensió. A efectes pràctics, és com si tinguéssim un potenciòmetre de valor de la suma de les resistències del circuit.

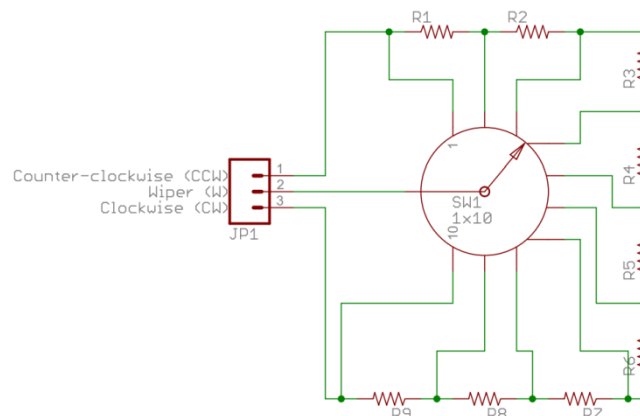


Figura 23. Circuit de la placa del selector

La placa té tres pins de sortida: W, CW i CCW. El pin de “wiper” es connectarà a una entrada configurada com a ADC; els altres dos pins es connectaran a 3,3V i 0V respectivament, per tal de augmentar el nivell de tensió al girar l'interruptor cap a la dreta. El connector que representa la sortida d'aquesta placa als plànols és el connector X5.

Les resistències que porta la placa són de 10kΩ; per tal de calcular la tensió de sortida que tindrem a cada posició, es calcula de la següent forma.

$$V_w = \frac{R_{CCW}}{R_{CCW} + R_{CW}} * V_{CW} \quad (\text{Eq.9})$$

On:

V_w : tensió de sortida en el “wiper” (V)

R_{CW} : valor de la resistència entre el pin CW i “wiper” (Ω)

R_{CCW} : valor de la resistència entre el pin CCW i “wiper” (Ω)

V_{CW} : valor de tensió en el pin CW (V)

Els resultats es troben a la taula:

Posició interruptor	V_w (V)	Conversió a bits (ADC 12 bits)
1	0,00	0
2	0,37	455
3	0,73	910
4	1,10	1365
5	1,47	1820
6	1,83	2275
7	2,20	2730
8	2,57	3185
9	2,93	3640
10	3,30	4095

Taula 10. Valors en bits i voltatge per a cada posició del selector

Per a cada canvi de posició, la tensió varia 0,37V del valor anterior. Al connectar el selector a un convertidor analògic-digital de 12 bits, suposa canvis de 455 nivells en el convertidor.

S'han afegit també dos potenciòmetres logarítmics de 10kΩ per tal de poder controlar paràmetres dels efectes, com per exemple en el “delay”, la quantitat de retard del senyal.

S'han escollit logarítmics degut a que l'oïda humana és més sensible als canvis de volum de manera logarítmica, el que permet que si amb aquests potenciòmetres volem regular volum, el canvi sigui més natural a l'oïda.

Els dos aniran connectats a 3,3V, i estaran connectats a entrades configurades com a entrades analògiques del microcontrolador de 12bits, per tant veurem canvis entre 0 (0V) i 4095 (3,3V).

Per tal d'identificar quin potenciòmetre utilitzem en cada efecte, s'han disposat de dos LED's indicadors, connectats a sortides digitals del microcontrolador amb resistències de 680Ω.

4.4.3. Mòdul LCD

S'ha afegit en el disseny un display LCD de 16x2, per tal de poder visualitzar l'efecte que estem utilitzant.

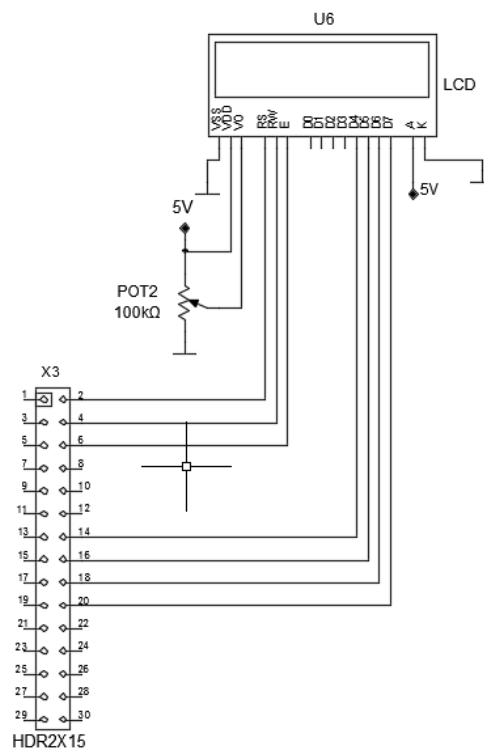


Figura 24. Connexions de la LCD

La LCD conté un díode LED per tal d'il·luminar a la pantalla (pins A i K), que es connecten a 5V i a massa respectivament. L'alimentació del mòdul són els pins VSS (0V) i VDD (5V). El pin V0 regula la nitidesa del text que es mostra a la pantalla; per tal de poder regular aquest paràmetre, s'ha col·locat un potenciòmetre entre 0 i 5V. S'ha escollit de 100kΩ per tal de tenir més marge en el rang de tensió en el qual el text és més nítid.

Per tal de poder escriure text en la LDC, s'ha utilitzat una llibreria específica per al controlador integrat en el mòdul HD44780. Amb aquesta llibreria només s'han hagut d connectar els pins RS (selecció de registre) , RW (llegir/escriure), E (habilitació), D4, D5, D6 i D7 (dades) a les respectives entrades configurades al microcontrolador.

5. DISSENY D'EFECTES DIGITALS

Tal com s'ha esmentat a la introducció del projecte, els efectes digitals permeten modificar el senyal de la guitarra per tal d'obtenir diverses sonoritats. Els efectes es poden classificar en diferents grups: efectes de filtratge, amplitud, retard, modulació, variació del temps i efectes especials del senyal.

A la taula es mostren els principals efectes en funció de la seva classificació.

Classificació	Tipus d'efecte
Filtratge	Pas-baix, pas-alt, passa-banda
Amplitud	Compressors, Limitadors, distorsió, fuzz
Retard	delay, chorus, fanger, vibrato, reverb
Modulació	Modulació, tremolo, vibrato
Variació del temps	Wah-wah, phaser
Especials	Loop, octave

Taula 11. Tipus de classificacions d'efectes

Tots els grups d'efectes, excepte els grups de filtratge i efectes especials, tenen un efecte base, és a dir, a partir d'aquest efecte es poden entendre els altres efectes de la mateixa classificació. Per exemple, en el grup d'efectes de retard del senyal, el model bàsic és el "delay", que simplement afegeix un retard al senyal al senyal actual; o en els efectes de modificació de l'amplitud d'ona, tenim del limitador. Després cadascun té les seves característiques diferents.

Podem veure que el món dels efectes digitals és molt ampli; en aquest projecte s'ha intentat acotar per tal de tenir una idea de disseny d'efectes, i poder implementar-los. En els pròxims apartats es parlarà de la definició dels efectes dissenyats i el seu modelatge.

5.1. Efectes de distorsió del senyal

Existeixen multitud d'efectes relacionats amb la distorsió del senyal; la base dels efectes de distorsió és el processament no lineal del senyal.

Es passa el senyal per un sistema no lineal, variant l'amplitud del senyal. En funció de la variació d'aquesta amplitud, obtindrem una sonoritat diferent; al variar l'amplitud de manera

no-lineal, es produeixen harmònics. Quant més es canvia l'amplitud i forma del senyal, més distorsió obtindrem en la sonoritat de sortida.

Hi ha múltiples maneres de canviar l'amplitud del senyal; una manera de crear distorsió és simplement limitar el senyal a partir d'un valor. Quant més es limiti el valor, més distorsió obtenim. En aquest efecte es coneix com a limitador. A la figura es pot veure com es veu afectat un senyal sinusoidal.

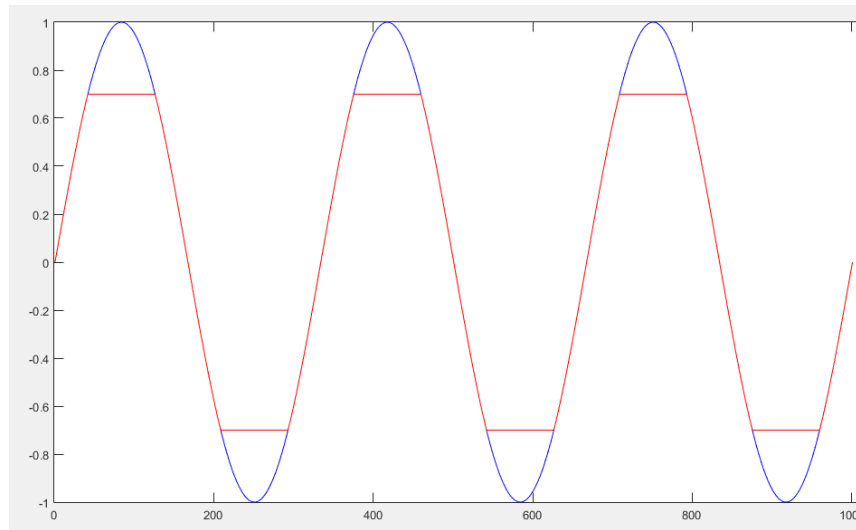


Figura 25. Efecte limitador

La limitació serà simètrica, és a dir, tant per als valors més alts com per els més baixos del senyal.

Aquest tipus de limitació provocarà un so bastant distorsionat. Una manera de fer una limitació simètrica més suau és fent servir l'equació següent:

$$y[n] = x[n] - g \left(\frac{x[n]}{3} \right)^3 \quad (\text{Eq.10})$$

On:

$y[n]$: senyal d'entrada (temps discret)

$x[n]$: senyal de sortida (temps discret)

n : número de mostra

g : guany de la distorsió

Amb aquesta funció al cub, es crea una distorsió més suau com es pot veure a la figura:

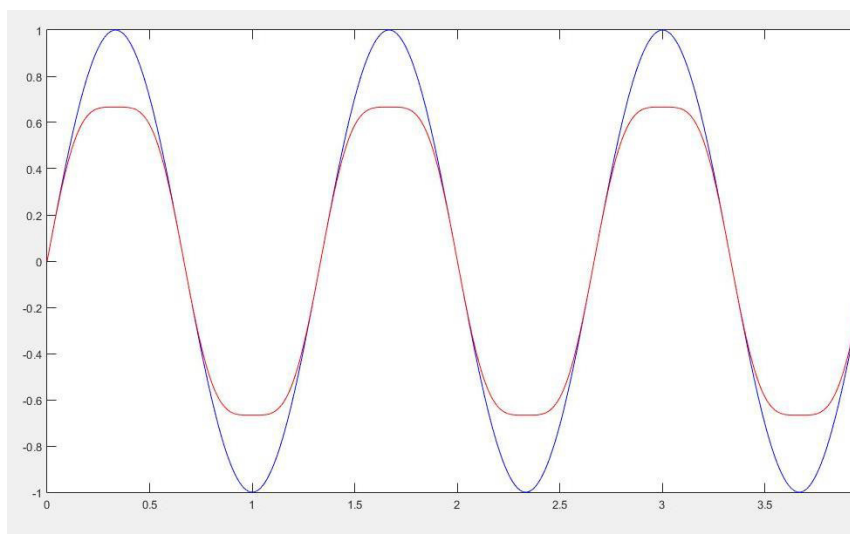


Figura 26. Efecte de distorsió

Aquesta funció representa de millor manera les no-linealitats que es poden obtenir en els amplificadors a vàlvules; quan les vàlvules es saturen, provoquen una no-linealitat semblant a la funció presentada. Alguns pedals comercials de guitarra emulen aquest tipus de saturació, com per exemple el Tube Screamer de la marca Ibanez.

Per a la programació d'aquesta funció, s'ha de tenir en compte que els valors han de ser tant positius com negatius. Amb el guany es podrà canviar la quantitat de distorsió aplicada; es programarà a partir d'un dels potenciòmetres externs. En el cas de l'efecte limitador, també s'aplicarà un guany que permetrà augmentar o disminuir el valor límit.

5.2. Efectes de retard del senyal

L'efecte bàsic del senyal de la guitarra el retard del seu senyal, anomenat "delay". L'objectiu d'aquest efecte és aplicar un retard a les mostres i després treure el senyal. Es fa com una suma entre el senyal actual i el senyal de retard.

La quantitat de retard implementat tindrà un efecte o altre en la sonoritat de l'efecte. La oïda humana, amb retards per sota dels 50ms, ajunta els sons de manera que es perceben com una reverberació; amb retards per sobre, el cervell és capaç d'assimilar-les com una separació del senyal.

La implementació del retard es fa a partir del senyal en temps discret, afegint un retard al senyal, i sumant el senyal actual. Aquest efecte es pot fer de dues maneres: implementant un filtre FIR (filtre de resposta finita) o un filtre IIR (filtre de resposta infinita) la diferència entre els dos serà l'eficiència del codi.

Per al disseny del nostre efecte s'ha dissenyat en base a un filtre FIR; els filtres FIR sempre són estables encara que presentin menys eficiència que els IIR.

Els filtres FIR són filtres de resposta finita. Aquest concepte es basa en què aquest tipus de filtre no té cap tipus de realimentació, el qual provoca que la resposta a impuls sigui finita en el temps (la resposta s'estableix a zero). El nombre de mostres és finit. La resposta a impuls d'aquest tipus de filtre dura $N+1$ mostres. L'estructura d'un filtre FIR seria la mostrada a la figura.

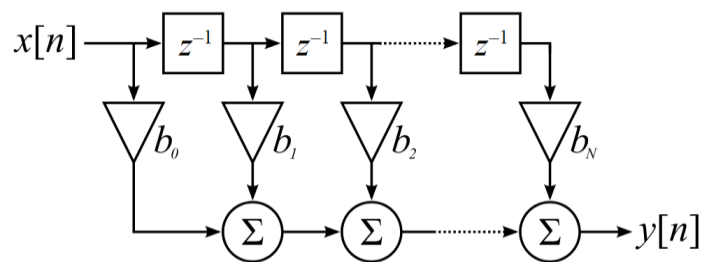


Figura 27. Estructura de filtre FIR

L'equació bàsica d'un filtre FIR, considerant el temps discret, és la següent:

$$y[n] = \sum_{k=0}^N b_k x[n-k] \quad (\text{Eq.11})$$

On:

N : ordre del filtre

$y[n]$: senyal d'entrada (temps discret)

$x[n-k]$: senyal retardada

b_k : coeficient del filtre

n : quantitat de retard assignat

k : factor de retard

L'ordre del filtre ve donat per el nombre de taps del filtre. Els taps del filtre són la parella de coeficients i retards. L'ordre de filtre serà el nombre de taps -1.

L'estructura bàsica de l'efecte de retard amb filtre FIR seria la següent:

$$y[n] = x[n] + g \cdot x[n-d] \quad (\text{Eq.12})$$

On:

$y[n]$: senyal d'entrada (temps discret)

$x[n]$: senyal de sortida (temps discret)

n : número de mostra

d : quantitat de retard assignat

g : guany del retard

Si passem el filtre al món freqüencial (transformada Z), l'equació seria la següent:

$$H(z) = 1 + g \cdot z^{-d} \quad (\text{Eq.13})$$

On:

$H(z)$: funció de transferència en món freqüencial

d : quantitat de retard assignat

g : guany del retard

L'equació quedaria representada gràficament d'aquesta manera:

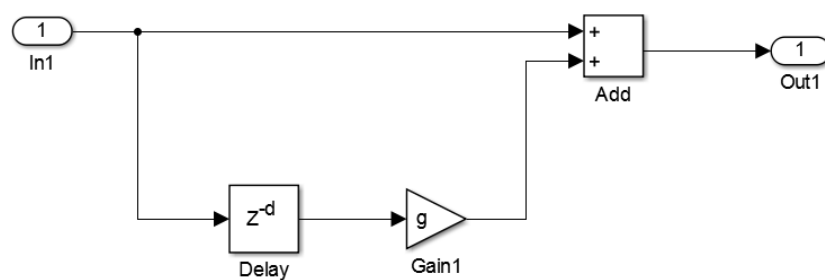


Figura 28. Estructura del "delay"

Degut a que la quantitat de “delay” es troba a l'exponent de la z , no s'ha pogut implementar amb les funcions que ofereix la llibreria CMSIS DSP, degut a que només deixen modificar els coeficients del filtre.

La implementació en el programa s'ha fet amb l'equació en temps discret; s'ha creat un buffer en el qual es guardaran la quantitat de mostres necessàries que seran equivalents al temps de retard a implementar. Aquest buffer s'anirà emplenant a mesura que es vagin adquirint noves mostres, i un cop s'acabi d'emplenar, es tornarà a sobre escriure des del principi. Aquest concepte de sobre escriure la mostra més antiga per la més actual s'anomena com buffer circular.

El concepte de buffer circular es basa en no moure les dades antigues contínuament de lloc, si no en emplenar el buffer i sobre escriure les mostres antigues. L'únic que canvia a cada iteració és la posició a on escrivim; la posició es guardarà en apuntadors que apuntaran a posicions del buffer; simplement s'aniran augmentant i tornant a iniciar els valors d'aquests apuntadors. A la figura es pot veure la diferència entre adreçament normal i circular:

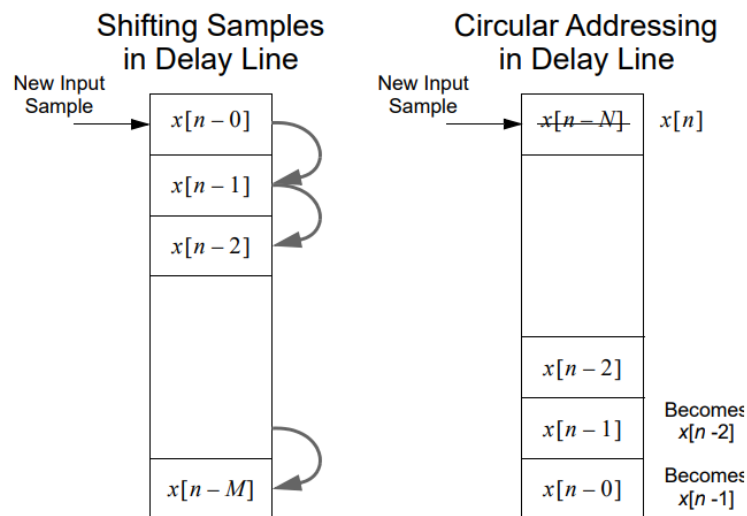


Figura 29. Concepte de buffer circular

L'adreçament circular presenta una major eficiència del codi que no pas l'adreçament normal; no es perd temps movent els valors del buffer de lloc. D'aquesta manera, assegurem que els efectes es troben implementats amb la major eficiència possible.

Per a l'efecte de “delay”, tindrem tres buffers; el buffer de “delay”, entrada i sortida. En tots tres s'aplicarà l'adreçament circular.

En els efectes de distorsió, només tindrem dos buffers (entrada i sortida) però també s'aplica el mateix concepte.

S'ha implementat un retard màxim de 500ms. Per tal de calcular el nombre de mostres necessàries a guardar, s'ha calculat el temps que dura una mostra (10µs per a una freqüència de 48kHz), i s'han comptat quantes mostres es transmeten fins arribar a 500ms. El resultat dona una quantitat de 25000 mostres a guardar (només tenim en compte les mostres del canal esquerre). El buffer anteriorment explicat s'emplenarà amb aquestes mostres, i es trauran per el canal dret.

5.3. Filtre pas-baix

Amb el disseny del filtre pas baix, es podrà provar la llibreria CMSIS DSP, que permetrà realitzar aquest filtre amb poques funcions.

Per tal d'agilitzar el disseny del filtre, s'ha utilitzat l'eina de disseny i anàlisi de filtres de MATLAB, en el qual podrem obtenir els coeficients del filtre de manera ràpida.

S'ha dissenyat un filtre pas baix que comença a 500Hz. S'ha escollit aquesta freqüència ja que a partir d'aquesta, es troben les freqüències mitges i altes de la guitarra; l'objectiu serà atenuar en part aquestes freqüències i els seus harmònics. S'ha definit que a la freqüència de 6kHz (freqüència on s'acaba l'espectre de la guitarra) hi hagi una atenuació de 40dB. En el programa s'ha especificat que es generi automàticament l'ordre del filtre, per tal de minimitzar al màxim el seu valor, el que implica un nombre menor de coeficients.

S'ha escollit realitzar un filtre FIR de tipus "Equiripple"; el filtre serà FIR degut a que sempre són estables, i la funció DSP del microcontrolador s'aplica per a filtres FIR; S'ha escollit de tipus Equiripple per reduir el nombre de coeficients. Per exemple, en el disseny basat en finestres s'obté una major quantitat de coeficients.

Es vol minimitzar al màxim el nombre de coeficients, ja que els filtres produeixen un retard inherent. Quant més coeficients (més ordre), més retard. En funció del nombre de taps i el període de mostreig del senyal, es pot obtenir el retard que produeix el filtre.

$$T = \frac{Taps-1}{2} * T_s \quad (Eq.14)$$

On:

T : retard provocat pel filtre (s)

T_s : període de mostreig del senyal (s)

El resultat de posar els valors a l'eina de disseny i anàlisi de filtres de Matlab és la següent.

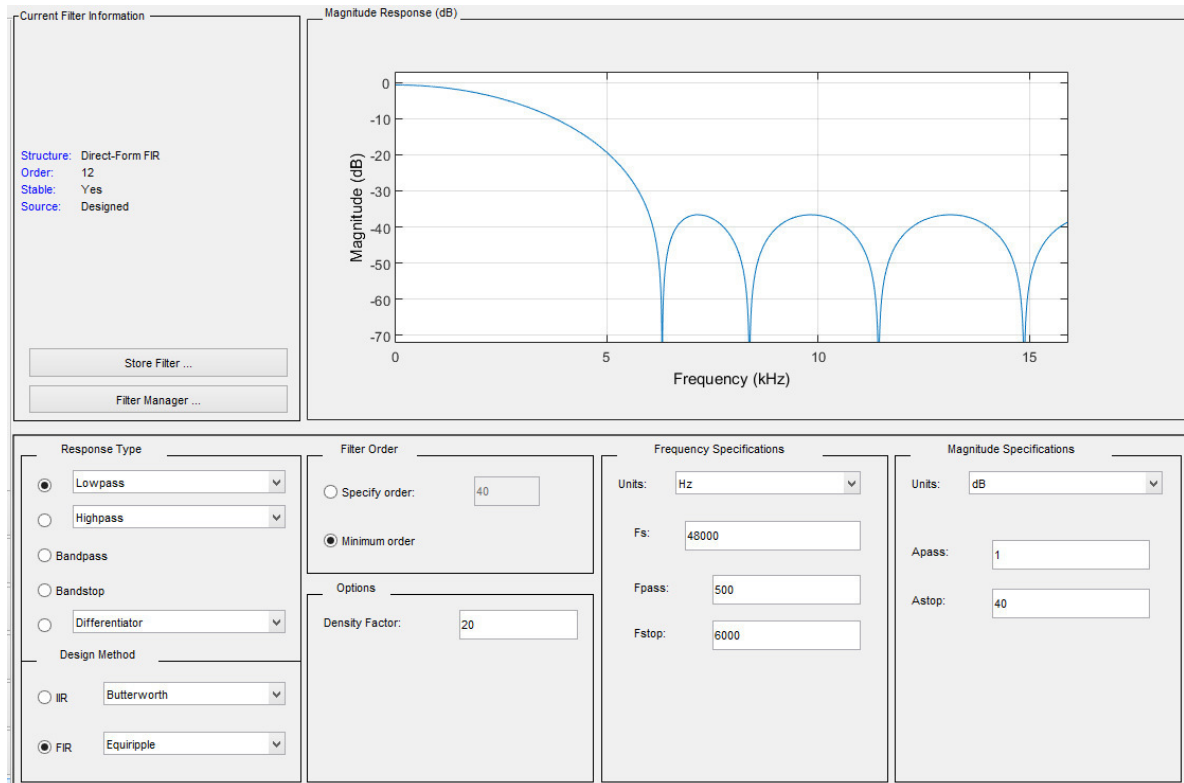


Figura 30. Especificacions del filtre en la eina de MATLAB

Obtenim un filtre d'ordre dotze, el qual aplicant la freqüència de mostreig del senyal de 48kHz, obtenim que el retard inherent del filtre és de 125us.

6. PROGRAMACIÓ NUCLEO-F429ZI

La programació de la placa NUCLEO-F429ZI s'ha realitzat a partir de l'entorn de programació Atollic True Studio (versió 9.0), i l'entorn STM32 CubeMX. Amb l'entorn d'Atollic True Studio s'ha escrit tot el codi relacionat amb la configuració del còdec i els efectes de guitarra, mentre que amb el STM32 CubeMX s'ha generat el codi d'inicialització de perifèrics amb les llibreries necessàries per al projecte.

Com es mostra a l'annex A, la interfície gràfica del STM32 CubeMX permet una inicialització ràpida al microcontrolador, permetent inicialitzar tots els perifèrics, així com configurar-los, i també configurar la disposició dels pins del microcontrolador.

Aquest programa treballa amb una capa d'abstracció (llibreria) creada per la marca STMicroelectronics, anomenada HAL. Aquesta capa treballa amb tots els perifèrics. S'utilitza per poder transportar codi entre els microcontroladors de la gamma de STMicroelectronics.

També s'ha treballat, com s'ha dit en el capítol 3, la utilització de les funcions que s'integren en el CMSIS DSP, que permeten optimitzar el codi de certes operacions utilitzades normalment per al processat digital del senyal, com poden ser filtres FIR, IIR o transformades de Fourier.

Per tal de millorar l'eficiència del codi, s'han desenvolupat alguns perifèrics a partir del DMA; aquest mètode de treball del microcontrolador permet llegir entrades o transferir dades sense la intervenció de la CPU. S'han habilitat canals de DMA per a les entrades analògiques i per al protocol I2S, i s'habilita la transferència contínua de dades, per tal d'estalviar cicles a la CPU i realitzar els efectes amb més rapidesa.

El codi es basa principalment en dues parts: la primera part és inicialitzar els perifèrics, incloure llibreries, inicialitzar variables i configurar el còdec via I2C; la segona part consisteix en una màquina d'estat que permetrà canviar l'efecte en funció del selector i processar el senyal.

La resta de funcions addicionals permetran l'ús del I2S i de les entrades analògiques en DMA, i la rutina d'interruptió per al "bypass".

6.1. Inicialització i configuració del còdec

S'ha fet un diagrama senzill per visualitzar els passos seguits en aquesta part del programa.

El primer pas ha sigut incloure les llibreries necessàries per al projecte. Les llibreries utilitzades, a part del main.h, són la HAL (stm32f4xx_hal.h), la llibreria per la LCD (lcd_hd44780_stm32f0.h), i la llibreria CMSIS DSP (arm_math.h). Per tal de que la llibreria DSP funcioni correctament, s'ha de definir el tipus de processador amb una macro (ARM_MATH_CM4) i incloure en el projecte l'arxiu libarm_cortexM4lf_math.a.

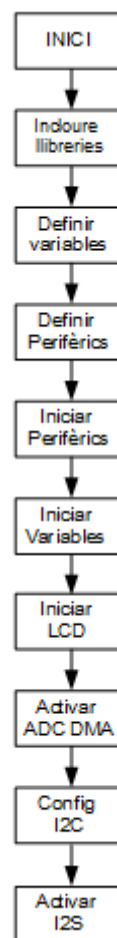


Figura 31. Diagrama d'inicialització de perifèrics i variables

El següent pas ha sigut definir totes les variables necessàries per al projecte, que s'aniran explicant en aquest capítol.

Abans d'entrar a la funció main, el programa CubeMX ha generat prèviament instàncies que serviran per configurar els paràmetres de cada perifèric, així com les funcions que els configuren. Els detalls d'aquestes instàncies es troben explicats a l'annex A.

Un cop dins del main, a la part del setup, es realitzen tots els passos definits en el diagrama; Primer es fa una crida a les funcions de configuració dels perifèrics i del sistema; aquest pas es troba generat per CubeMX.

Seguidament s'han inicialitzat certes variables, com per exemple els apuntadors que serviran per processar el senyal, i inicialitzar el filtre FIR.

També s'inicialitza la LCD amb la funció 'LCD_Init()', i es neteja la pantalla amb la funció 'LCD_Clear()'.

En el "setup" també s'activa el ADC per a començar a llegir registres de manera contínua. Es realitza amb la funció 'HAL_ADC_Start_DMA'. Els valors llegits de cada entrada es guarden al buffer anomenat 'dadapots'. Aquests valors es guarden posteriorment a les variables 'selector', 'pot1' i 'pot2', i es fa a partir de la funció 'HAL_ADC_ConClptCallback', que es definirà posteriorment.

Seguidament es realitza la configuració del còdec mitjançant I2C. Per a cada registre configurat s'ha declarat un "array" de dos bytes; un byte servirà per definir l'adreça del registre a configurar, i l'altre guardarà els valors de configuració del registre. Els valors de cada registre es troben explicats al capítol 3.

La transmissió dels arrays de configuració es fa a partir de la funció 'HAL_I2C_Master_Transmit', amb la qual es defineix l'adreça del còdec definida a la variable 'devaddress', i es defineix que es transmetin 2 bytes (registre i configuració). Es posa entre transmissions un retard de 100ms per tal d'assegurar la finalització de cada transmissió. Realitzant la configuració del I2C, el còdec queda activat.

Per últim, s'activa la transmissió contínua de dades amb la funció 'HAL_I2SEx_TransmitReceive_DMA', en el qual es fa una transferència de 50 dades per l'entrada i 50 per la sortida.

6.2. Màquina d'estat

A dins del bucle principal, s'ha creat una màquina d'estat, que permetrà seleccionar l'efecte desitjat per l'usuari, i programar a dins de cada estat per tal de crear l'efecte desitjat.

El selector permet seleccionar fins a deu efectes, però per simplificar el codi i degut a que, només s'han creat quatre efectes, només s'han creat quatre estats per als efectes.

Els estats principals seran 'SELECCIÓ', 'EFECTE1', 'EFECTE2', 'EFECTE3' i 'EFECTE4'. Aquests estats es veuran afectats per la posició del selector. També tindrem estats per inicialitzar els efectes ('EF1_INIT', 'EF2_INIT', 'EF3_INIT', 'EF4_INIT').

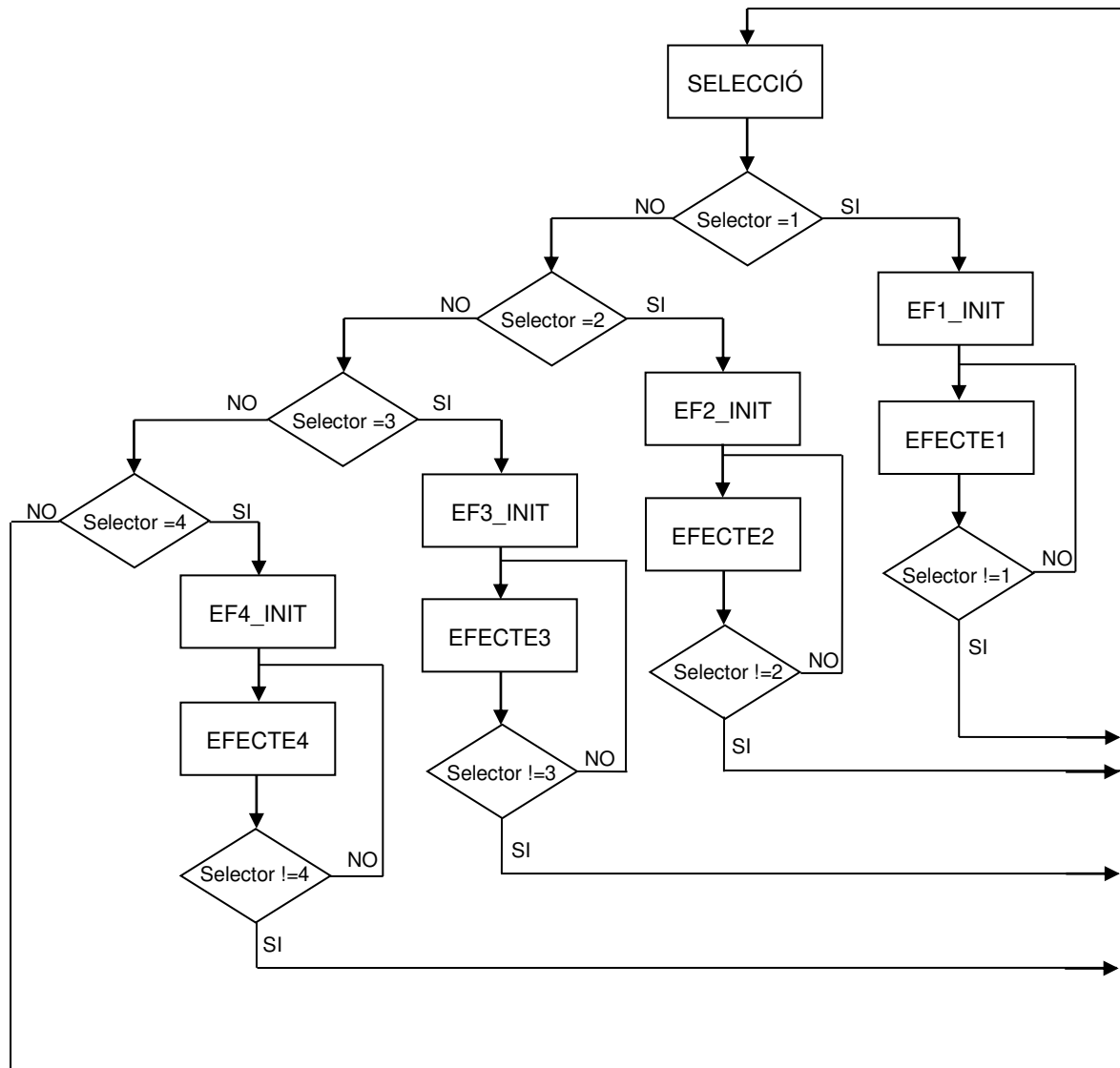


Figura 32. Diagrama de la màquina d'estat

En el diagrama de blocs s'ha posat la posició del selector com a 1,2,3 i 4. En realitat, aquests valors seran dins d'uns marges entre 0 i 4095.

Per tal de realitzar la màquina d'estat, s'ha creat una estructura "switch case", en el qual tindrem els nou possibles estats a dins. Per fer més entenedor e codi, s'ha creat la variable

‘estatactiu’, provinent de la classe ‘estat’, per tal de no enumerar els estats com a números (0,1,2..) i que l’usuari vegi de manera més orientativa els estats declarats (‘SELECCIÓ’, ‘INICI’, ‘EFECTE1’...).

En el estat ‘SELECCIÓ’, es comprova el valor del selector, entre un rang de valors. Els valors (en bits) de cada posició del selector són els següents:

Posició interruptor	Conversió a bits (ADC 12 bits)
1	0
2	455
3	910
4	1365
5	1820
6	2275
7	2730
8	3185
9	3640
10	4095

Taula 12. Conversió de les posicions del selector a valors de 0 a 4095

Per a determinar el valor del selector, es crea un marge de 100 valors per amunt i per avall, per tal de detectar el valor de posició del selector tot havent-hi oscil·lacions en el valor d’entrada. Per exemple, per a la posició 2 del selector, el marge variarà entre 355 i 555.

Els estats ‘EF1_INIT’, ‘EF2_INIT’, ‘EF3_INIT’ i ‘EF4_INIT’ simplement serveixen per escriure en pantalla el nom de l’efecte a partir de la funció ‘LCDWriteString’, i activar el LED corresponent al potenciòmetre que es vulgui fer servir, amb la funció ‘HAL_GPIO_WritePin’; la sortida es defineix amb el valor del pin associat a la GPIO corresponent. Els estats del pin poden ser ‘SET’ (alt) o ‘RESET’ (baix).

En cada estat dels efectes, es determinarà la condició oposada de la variable del selector, que en cas de no ser certa, l’estat tornarà a ser l’estat de ‘SELECCIÓ’. Al sortir de l’estat de cadascun, es posarà a ‘RESET’ la sortida del pin del LED.

6.2.1 Efectes

Tal i com s'ha dit anteriorment, a cada estat trobarem un efecte diferent. Les variables són les mateixes per a cada efecte (buffer d'entrada, buffer de sortida i valors de potenciòmetres associats al guany). S'han realitzat apuntadors que apunten als buffers d'entrada i sortida ('in' i 'out'). Els apuntadors són els anomenats 'input' i 'output'.

Cada efecte es fa a dins d'un cicle 'for'. A dins d'aquest cicle, s'aniran incrementant els valors de 'input' i 'output'. Gràcies al apuntador output guardarem la senyal ja processada en el buffer de sortida llesta per transmetre. Primer s'inicialitzen els apuntadors a la posició 0 del buffer i es va incrementant fins a 24 (els buffers d'entrada i de sortida són de 50 valors de 16 bits cadascun, però només agafem valors del canal esquerre). Un cop s'acaba el 'for', els apuntadors es tornen a inicialitzar a la posició 0 del buffer.

Per als efectes relacionats amb la distorsió del senyal (limitador i distorsió) s'ha creat la variable 'gain', que permetrà regular la quantitat de distorsió de 0 a 1 en funció del valor del potenciòmetre, en aquest cas la variable 'pot2'.

Per al efecte de distorsió (en l'estat 'EFECTE2') , simplement s'ha aplicat la funció explicada al capítol 5. Per al efecte limitador (en l'estat 'EFECTE3') , s'ha creat arbitràriament la limitació de 10000 a 55536 (quan la variable 'gain' sigui 1). Aquests valors límit es pot canviar en funció de la quantitat de distorsió final desitjada.

Per a l'efecte de "delay", s'ha creat un buffer anomenat 'bufferdelay', per tal de guardar les dades d'entrada i poder realitzar el retard. Per tal de regular la quantitat de "delay" s'ha fet un augment de 100 a 500ms, en funció del valor del potenciòmetre utilitzat (en aquest cas la variable 'pot1'). La quantitat de mostres a guardar en funció del retard a realitzar es fa a partir de la variable 'num_mostres'.

El buffer on es guarden les dades de "delay", té un apuntador ('dadadelay'), que permet també incrementar el valor de la posició del buffer a modificar. Primer s'inicialitza el buffer de retard amb zeros; després es va emplenant fins arribar al numero de mostres necessàries per realitzar el retard, i un cop s'acaba d'emplenar, s'extreuen les dades guardades i comencen a tenir efecte en el buffer de sortida. Les dades del retard només s'extreuen per el canal dret.

El buffer de retard és de 25000 mostres, degut a que és el màxim retard que hem especificat (500ms), però quan s'entra en el cicle 'for', només s'emplena el buffer fins al valor de la variable 'num_mostres'.

Quan sortim d'aquest estat, posem a zero les dades del buffer de retard.

Per al efecte del filtre pas baix ('EFECTE4'), es fan servir les funcions de CMSIS DSP; per tal de crear el filtre, primer s'ha de crear una instància de tipus 'arm_fir_instance_f32', la qual anomenarem filtre; es guarden els valors dels coeficients en un buffer anomenat coeficients, i apuntadors per als buffer d'entrada i sortida; seran diferents que els altres apuntadors d'altres efectes, ja que s'ha de treballar en coma flotant. S'inicialitza el filtre amb la funció 'arm_fir_init_f32'.

La funció es processarà per blocs de vint-i-cinc, per tant es fa un cicle for per tal de executar la funció del filtre fir que es diu 'arm_fir_f32' (es fa en dos cicles) . Un cop es surt del for es tornen a inicialitzar els apuntadors a la posició zero.

6.3. Programació I2S

Amb aquest perifèric s'habilita la transferència de dades per DMA; gràcies al DMA podrem tenir una transferència contínua de dades, sense tenir parades de transferència degut al processament del senyal programat.

Tal com s'ha dit anteriorment, s'activa la transmissió contínua de dades amb la funció 'HAL_I2SEx_TransmitReceive_DMA'. Per tal de rebre i enviar les dades, s'han creat dos buffers: 'rx_dma' i 'tx_dma' amb els seus respectius apuntadors. És necessari afegir aquests dos, a més dels buffers 'in' i 'out' abans explicats, per tal d'evitar la modificació de les dades durant la transferència.

Per tal de sincronitzar el processament de les dades amb el protocol, s'han utilitzat dues funcions en concret, que el propi DMA crida; una funció es crida un cop la transferència de dades s'ha completat, en aquest cas anomenada 'HAL_I2SEx_TxRxCpltCallback', i una altra quan la transferència es troba a la meitat ('HAL_I2SEx_TxRxHalfCpltCallback').

Un cop la transferència estigui completada, en la funció actualitzarem els buffers d'entrada i sortida; el buffer 'in' l'emplenarem amb les dades de 'rx_dma', i el 'tx_dma' l'emplenarem amb el buffer 'out'. S'ha posat la condició de només actualitzar les dades estant a dins dels estats (quan es produeix el processament del senyal); per a qualsevol altre estat, s'enviaran zeros a la sortida.

Un cop s'actualitzin els buffers, s'ha creat una variable anomenada 'trcpt', per tal d'avisar al estat d'efecte que ja pot començar a processar les dades. Per tal d'augmentar el sincronisme, en la funció cridada un cop la transferència es troba al mig, es posa a zero aquesta variable.

6.4. Interrupció i entrades analògiques

Per a la interrupció del "bypass", el programa CubeMX genera una sèrie de funcions, que permeten habilitar el servei d'interrupcions del sistema, i habilitar la generació d'una interrupció amb el pin seleccionat. Amb el codi generat, una funció anomenada 'HAL_GPIO_EXTI_IRQHandler' detecta el canvi d'estat en el pin, i genera una crida a una funció anomenada 'HAL_GPIO_EXTICallback'. Aquesta funció es posa en el 'main', i dins d'aquesta programem la rutina d'interrupció.

Degut a que cada servei d'interrupcions té múltiples línies, abans de fer qualsevol codi es comprova que la interrupció l'ha generat el pin correcte.

Un cop a dins, per tal d'evitar sorolls per part del còdec (encara que es desconnectin les sortides amb el propi 4PDT) es deshabiliten les sortides també per software canviant el bit OUTPD del registre "Power Down Control" del còdec. També s'habilita el LED que indica al usuari que el "bypass" està activat, i s'escriu per pantalla. Per tal de mantenir l'estat d'interrupció, s'ha creat un "while" que s'executa fins que l'estat del pin de "bypass" no torni a ser alt (la interrupció es genera en el flanc de baixada del pin). Un cop surt es tornen a habilitar les sortides, es deshabilita el LED i es neteja la pantalla.

Per al ADC, també s'ha treballat amb DMA, per tal de llegir contínuament les entrades del ADC, sense haver de passar per la CPU, i evitar el cicles necessaris per llegir les dades contínuament en el processament de senyal.

Tal com s'ha dit aneriorment, s'ha activat la conversió d'aquestes entrades amb la funció 'HAL_ADC_Start_DMA'. Cada vegada que es produeixen totes les conversions, aquesta crida a la funció 'HAL_ADC_ConCpltCallback'. A dins d'aquesta funció actualitzarem les variables del selector i els potenciòmetres amb els nous valors del buffer 'dadapots'. Dins d'aquesta també actualitzarem les variables 'num_mostres' i 'gain' en funció dels valors del potenciòmetres.

7. RESUM DEL PRESSUPOST

El cost econòmic de la realització del projecte del pedal digital programable, tant per el disseny com per el muntatge, ascendeix a mil vuitanta-un euros amb setanta-un cèntims, sense IVA.

8. CONCLUSIONS

En aquest document s'han especificat tots els càlculs i justificacions necessàries per tal de dissenyar el pedal digital de guitarra programable; amb el disseny escollit es poden programar tant els efectes dissenyats en el present projecte, com d'altres de diferents.

S'han pogut implementar diversos algorismes d'efectes; en essència s'han dissenyat els efectes més comuns i coneguts aplicats a la sonoritat de la guitarra, com poden ser efectes de retard, distorsió i equalització.

Amb el tipus de tecnologia escollida, en aquest cas la placa NUCLEO-F429ZI, es poden programar multitud d'efectes, degut a que es disposa de la velocitat necessària per realitzar els algorismes, així com disposa de DMA per al control de perifèrics, que permet una major eficiència del codi.

S'ha escollit el tipus de convertidor necessari per tal de convertir i reconstruir el senyal de la guitarra; el tipus de tecnologia escollida (còdec) ha permès una interacció simple amb el microcontrolador, podent transmetre dades amb el protocol I2S. S'ha adaptat correctament el senyal de la guitarra segons a les especificacions del còdec, així com les condicions de sortida d'un pedal de guitarra.

S'han afegit al disseny un selector, dos potenciòmetres, LED's i una pantalla LCD per tal de poder visualitzar l'efecte utilitzat en cada moment, així com regular paràmetres de guany i retard en els diferents efectes. També s'ha afegit un pulsador 4PDT amb el qual s'ha realitzat un "true bypass" al senyal.

Marta González Palomino

Graduada en Enginyeria Electrònica Industrial i Automàtica

Girona, 3 de setembre de 2018

9. RELACIÓ DE DOCUMENTS

Aquest projecte consta dels següents documents: memòria, plànols, plec de condicions, estat d'amidaments i pressupost.

10. BIBLIOGRAFIA

ANALOG DEVICES. Decoupling Techniques. (<http://www.analog.com/media/en/training-seminars/tutorials/MT-101.pdf>, 12 d'abril de 2018).

ANALOG DEVICES. Understanding Advanced Processor Features Promotes Efficient Coding(<http://www.analog.com/media/en/technical-documentation/technical-articles/350395352047424547665311ProgrammingTechniquesForDSPs.pdf>, 25 de febrer de 2018).

ANALOG DEVICES. Which ADC Architecture is right for your application?. (<http://www.analog.com/en/analog-dialogue/articles/the-right-adc-architecture.html>, 22 de febrer de 2018).

ARM. Designing with the Cortex-M4. (<https://www.arm.com/files/pdf/DSPConceptsM4Presentation.pdf>, 23 de març de 2018).

IFEACHOR, E.C., JERVIS, B.W. Digital Signal Processing a practical approach. Adisson-Wesley Publishing Company. Madrid 1993.

MIT OPEN COURSE WARE. How to bias an opamp. (https://ocw.mit.edu/courses/media-arts-and-sciences/mas-836-sensor-technologies-for-interactive-environments-spring-2011/readings/MITMAS_836S11_read02_bias.pdf, 5 de març de 2018).

NOVIELLO, C. Mastering STM32. Leanpub. 2018.

QUIRANTE CATALÁN, C. Implementación de algoritmos de efectos de audio en un procesador DSP de TI. Treball Final de Carrera. Ingeniería Técnica de Telecomunicación, Especialidad Sistema de Telecomunicación. Universitat Politècnica de Catalunya. Març de 2008.

ST. Audio playback and recording using the STM32F4DISCOVERY. (http://www.st.com/content/ccc/resource/technical/document/application_note/c7/2f/66/a5/cd/4c/4d/2a/DM00040802.pdf/files/DM00040802.pdf/jcr:content/translations/en.DM00040802.pdf, 10 d'abril de 2018).

TEXAS INSTRUMENTS. Digital Guitar effects pedal
(<http://www.ti.com/lit/ml/sprp499/sprp499.pdf>, 2 de febrer de 2018).

TEXAS INSTRUMENTS. How delta-sigma ADC work.
(<http://www.ti.com/lit/an/slyt438/slyt438.pdf>, 23 de febrer de 2018).

11. GLOSSARI

ADC: (Analog to Digital Converter) conversor analògic a digital

CMSIS: (Cortex Microcontroller Software Interface Standard) interfície de programació estàndard per a microcontroladors tipus Cortex

CPU: (Central Processing Unit) unitat central de processament

DAC: (Digital to Analog Converter) conversor digital a analògic

DC: (Direct Current) corrent contínua

DMA: (Direct Memory Access) accés directe a memòria

DSP: (Digital Signal Processing) processament digital del senyal, o (Digital Signal Processor) processador digital de senyal

FIR: (Finite Impulse Response) resposta a impuls finita

GPIO: (General Purpose Input/Output) entrada-sortida de propòsit general

HAL: (Hardware Abstraction Level) capa d'abstracció de hardware

I2C: (Inter- Integrated circuit) protocol de comunicació entre dispositius

I2S: (Inter-Integrated circuit Sound) protocol de comunicació d'àudio entre dispositius

IIR: (Infinite Impulse Response) resposta a impuls infinita

LCD: (Liquid Crystal Display)

LED: (Light Emitting Diode) díode d'emissió de llum

MAC: (Multiply-Accumulate operation) operació de multiplicació i suma

PCB: (Printed Circuit Board) placa electrònica

PSoC: (Programmable System on Chip) microcontroladors programables

RISC: (Reduced Instruction Set Computer) computador amb set d'instruccions reduïdes

RTOS: (Real Time Operating Systems) sistemes d'operació en temps real

SPDT: (Single-Pole Double Throw) un pol, llançament doble

SPI: (Serial Peripheral Interface) interfície sèrie per perifèrics

SRAM: (Static Random Access Memory) memòria d'accés aleatòria i estàtica

USART: (Universal Synchronous and Asynchronous Receiver-Transmitter) Protocol sèrie síncron i asíncron

USB: (Universal Serial Bus) port de dades sèrie universal

A. CONFIGURACIÓ STM CUBEMX

Gràcies a l'entorn gràfic de STM CubeMX, s'ha pogut generar automàticament el codi d'inicialització amb les llibreries adequades per al nostre projecte. En aquest annex es presenta com s'han configurat tots els paràmetres des de zero, el funcionament d'aquest entorn, i l'explicació del codi generat per aquest programa.

A.1. Interfície del programa

El entorn STM CubeMX permet d'una manera visual configurar els pins del microcontrolador (com poden ser perifèrics, rellotges, protocols...) i després permet la generació del codi C amb la configuració definida, incloent les llibreries adients. També permet fer el càlcul de consum del xip per a la nostra configuració.

El programa es pot descarregar a la pàgina web de STMicroelectronics de manera gratuïta.

Un cop s'instal·la el programa, la pantalla d'inici és la següent:

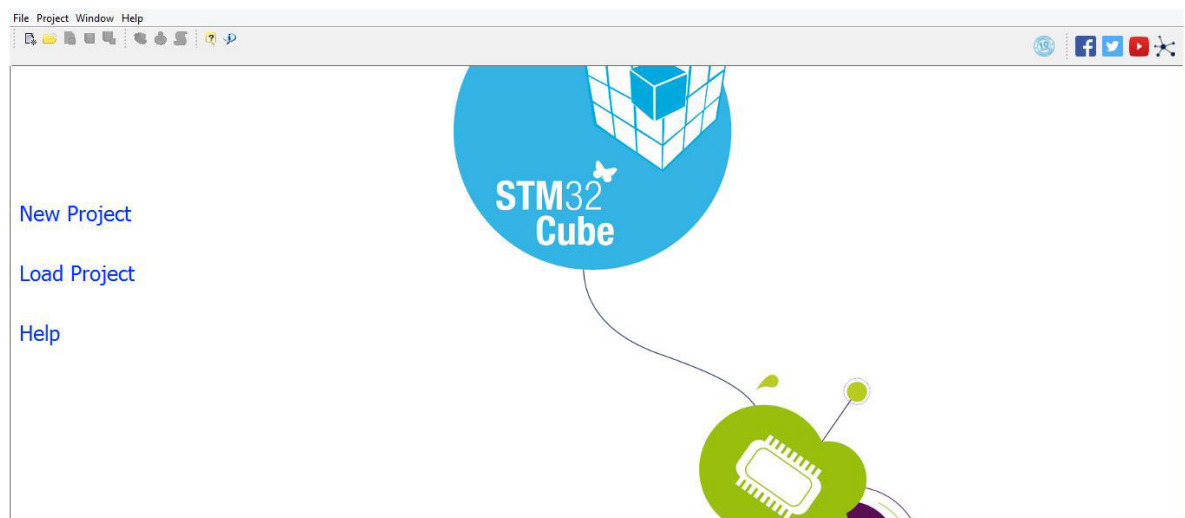


Figura 33. Pantalla d'inici del programa

Es clica a New Project, per obrir un nou projecte per a la nostra placa de desenvolupament. Seguidament s'obre una pantalla de selecció del component que tenim.

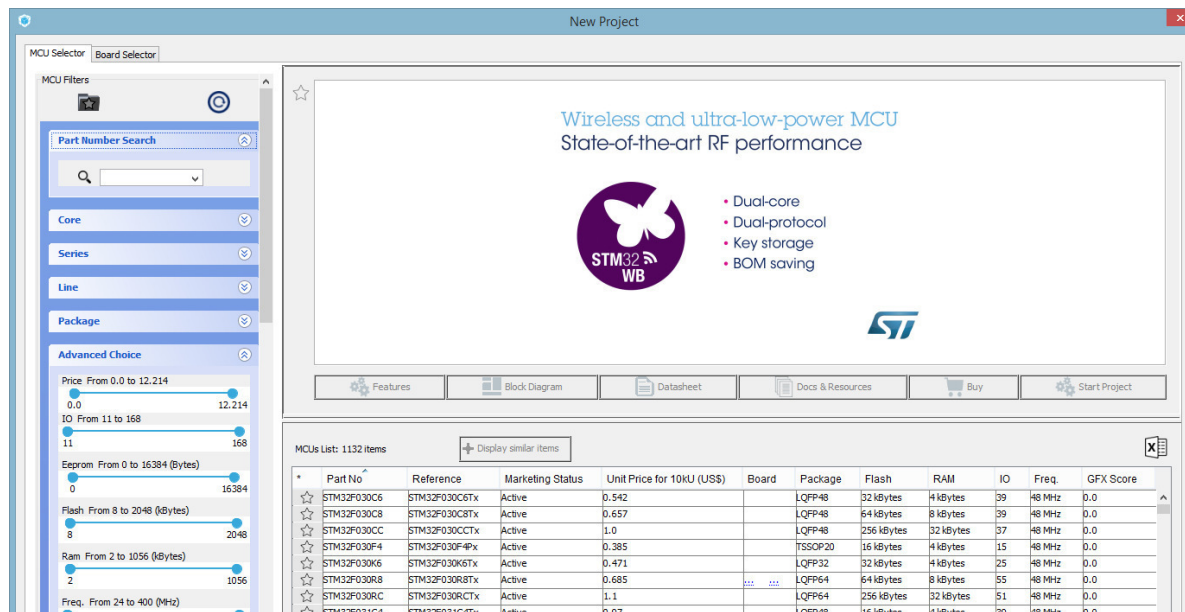


Figura 34. Selecció de placa de desenvolupament

Cliquem a la pestanya Board Selector, i es busca la placa NUCLEO-F429ZI

Es selecciona la placa i es clica Start Project. Seguidament el programa ens preguntarà que si volem inicialitzar els perifèrics per defecte de la placa, i cliquem que si. La pantalla un cop creat el projecte és la següent:

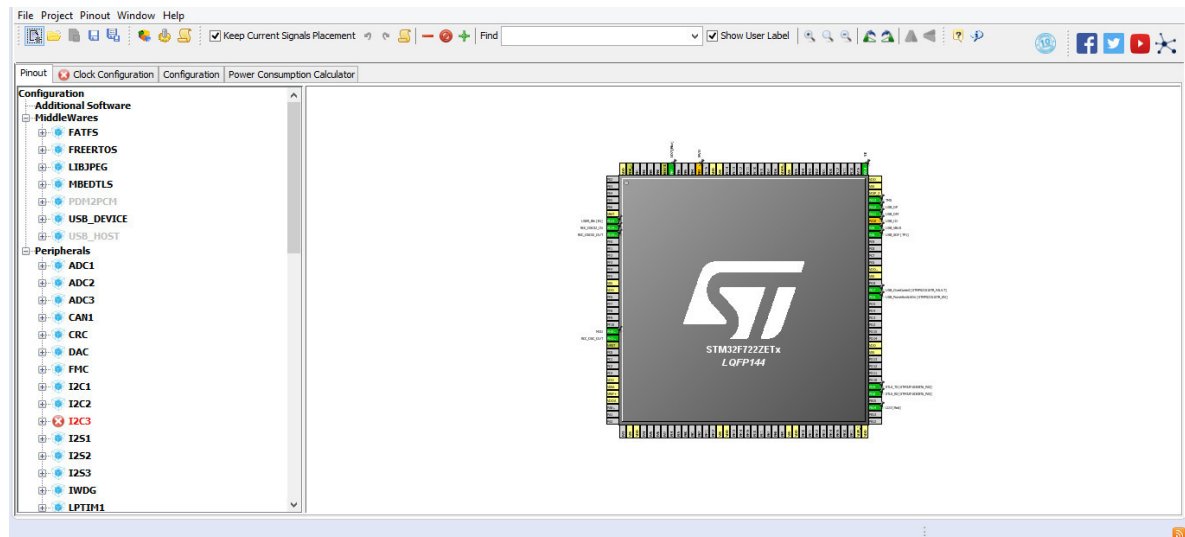


Figura 35. Pantalla principal

Com es pot veure, a la pantalla principal es podran configurar tots els pins del microcontrolador.

Tots els pins en color són els que venen configurats per defecte a la placa, com per exemple els pins per al USB, els LED's de la placa ,els pins d'alimentació del xip, els pins per al

rellotge extern de 32MHz. Hi ha pins que no es poden configurar, com per exemple el pin de RESET, o els pins d'alimentació de la placa.

La resta de pins (en gris) no es troben configurats, i cadascun pot fer més d'una funció (entrada/sortida general, pin per ADC's, DAC's, pins per la configuració de protocols (I2C, SPI...), el que permet més llibertat per moure pins i poder realitzar un millor disseny de la placa.

A la part esquerra de la pantalla es troben tots els perifèrics/serveis disponibles, que es poden habilitar o des-habilitar. La segona pantalla es troba dedicada íntegrament als rellotges que controlen els diversos perifèrics de la placa, permetent canviar els rellotges en funció de la velocitat que es vulgui aconseguir.

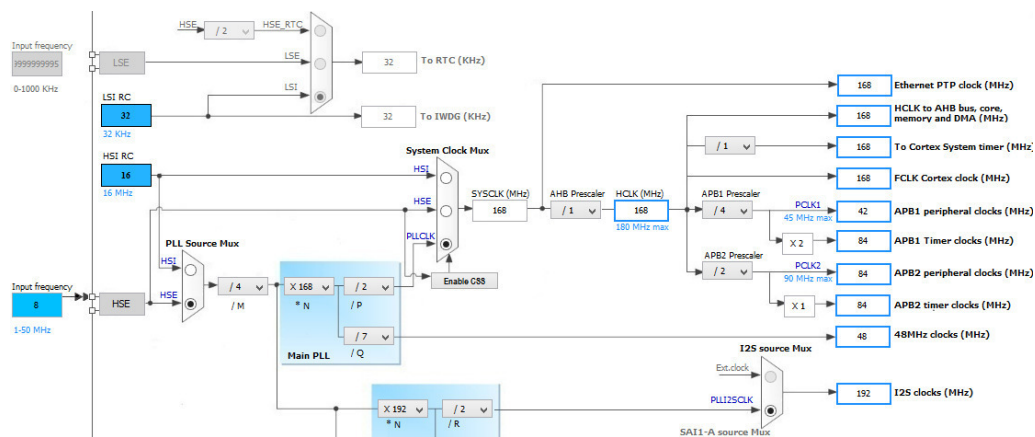


Figura 36. Pantalla de configuració de rellotges

La tercera pantalla permet configurar tots els perifèrics en funció de les necessitats del projecte.

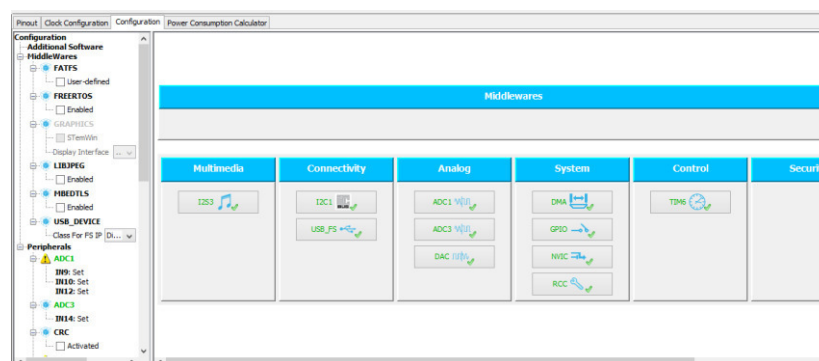


Figura 37. Pantalla de configuració de perifèrics

A continuació s'explicarà la configuració escollida per a cada perifèric.

A.2. Configuració de perifèrics

La interfície permet activar i desactivar qualsevol perifèric; per defecte, la placa tenia activat el Ethernet i el UART, cosa que s'han desactivat per tenir més quantitat de pins disponibles.

Els perifèrics que s'han activat són: I2C, I2S (DMA), ADC (DMA) i GPIO's (per a LED's). Els pins relacionats amb la pantalla LCD s'han configurat a dins de la llibreria.

Per al I2S, s'ha configurat la velocitat a 400kHz, degut a que el còdec treballa a aquesta velocitat. L'adreça s'ha configurat de 7 bits, degut a que el còdec té una adreça de 7 bits.

Master Features	
I2C Speed Mode	Fast Mode
I2C Clock Speed (Hz)	400000
Fast Mode Duty Cycle	Duty cycle Tlow/Thigh = 2
Slave Features	
Clock No Stretch Mode	Disabled
Primary Address Length selection	7-bit
Dual Address Acknowledged	Disabled
Primary slave address	0
General Call address detection	Disabled

Figura 38. Configuració del perifèric I2C

Les sortides que seran les SDA i SDL s'han configurat com a sortides “no “pull-up” no pull-down”, degut a que s'han afegit les resistències de “pull-up” externament.

Per al I2S, s'ha escollit treballar a una freqüència de 48kHz i amb un tamany de les dades de 16 bits, la qual cosa el perifèric s'ha configurat amb aquests paràmetres.

Transmission Mode	Mode Master Receive
Communication Standard	I2S Philips
Data and Frame Format	16 Bits Data on 16 Bits Frame
Selected Audio Frequency	48 KHz
Real Audio Frequency	48.0 KHz
Error between Selected and Real	0.0 %
Clock Parameters	
Clock Source	I2S PLL Clock
Clock Polarity	Low

Figura 39. Configuració del perifèric I2S

El paràmetre de polaritat s'ha escollit a baixa, degut a que el canvi de canal es produeix en el flanc de baixada del rellotge. A part també s'ha configurat un rellotge de sortida, que per defecte surt a la freqüència adequada, degut a que la freqüència ja s'ha establert a 48kHz.

S'ha activat també la configuració de DMA. Només s'han d'afegir els canals i automàticament s'adreça a un DMA del microcontrolador. El DMA treballa amb un buffer d'entrada de dades i de sortida de dades. Aquest buffer s'ha configurat amb dades tipus "word" degut a que les dades d'entrada i sortida seran de 16 bits.

DMA Request	Stream	Direction	Priority
SPI3_RX	DMA1 Stream 0	Peripheral To Memory	Low
I2S3_EXT_TX	DMA1 Stream 5	Memory To Peripheral	Low

DMA Request Settings

Mode: Normal

Increment Address: ☐

Use Fifo: ☐ Threshold:

Data Width: Word

Peripheral: ☐ Memory: ☒

Word

Figura 40. Configuració del DMA

Per a les entrades analògiques que són els potenciòmetres i el selector, es configura un sol ADC, degut a que un sol ADC té diverses entrades, i aquestes es van llegint en funció de l'ordre configurat per l'usuari.

ADCs_Common_Settings	
Mode	Independent mode
ADC_Settings	
Clock Prescaler	PCLK2 divided by 4
Resolution	12 bits (15 ADC Clock cycles)
Data Alignment	Right alignment
Scan Conversion Mode	Enabled
Continuous Conversion Mode	Enabled
Discontinuous Conversion Mode	Disabled
DMA Continuous Requests	Enabled
End Of Conversion Selection	EOC flag at the end of all conversions
ADC_Regular_ConversionMode	
Number Of Conversion	1
External Trigger Conversion Source	Regular Conversion launched by software
External Trigger Conversion Edge	None
Rank	1
Rank	2
Rank	3
ADC_Injected_ConversionMode	
Number Of Conversions	0

Figura 41. Configuració del ADC

S'ha configurat a 12 bits, per tenir més rang de valors entre 0 i 3,3V. S'han habilitat els paràmetres necessaris per tal d'habilitar el DMA, i habilitar la conversió contínua de dades, per tal de poder llegir els valors dels potenciòmetres i el selector de manera contínua sense haver de passar per la CPU.

En quant a la resta d'entrades i sortides, s'han configurat les sortides que van a LED's com a sortides "no pull-up no pull-down", per tal de poder controlar el seu estat (alt o baix).

L'entrada de la interrupció s'ha configurat per fer el trigger de la interrupció quan es produeixi un flanc de baixada, degut a que l'estat de "bypass" implica que l'entrada de la interrupció seran 0V.

A.3. Codi generat per CUBEMX

Un cop es configuren tots els perifèrics, es generen els arxius i les carpetes generals del programa, així com les llibreries incloses.

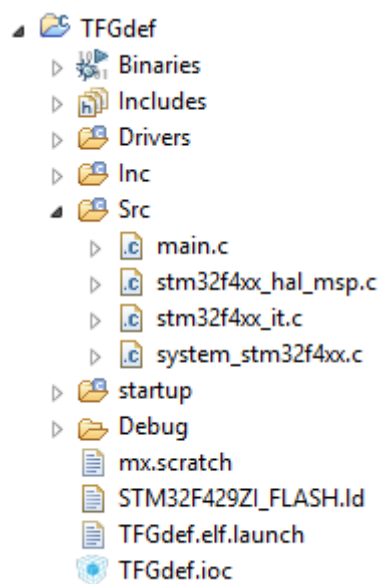


Figura 42. Arxius generats per CubeMX

En la carpeta Src trobem el main.c. També hi trobem l'arxiu on es configuren les entrades i sortides de cada perifèric així com l'habilitació dels rellotges de cada perifèric, anomenat stm32f4xx_hal_msp.c. En l'arxiu stm32f4xx_it.c, es configuren tot el relacionat amb les interrupcions, inclòs la rutina d'interrupció. En l'arxiu System_stm32f4xx.c s'escriuen valors a multitud de registres necessaris per al funcionament del microcontrolador.

En la carpeta de “Drivers” trobem les dues llibreries principals; la llibreria CMSIS, que és la llibreria estàndard, i la llibreria HAL, que és la que farem servir nosaltres. També es generen els ‘includes’ necessaris, així com els arxius de debug i l’arxiu del CubeMX. La configuració dels perifèrics es realitza en el main.c, a partir de funcions generades per el programa.

La configuració dels perifèrics la realitza de la manera següent: per a cada perifèric, es genera una instància per a el tipus de classe de cada perifèric.

Per exemple, per al I2S es generen tres instàncies; una per a la configuració del I2S(‘hi2s3’), i les altres dues per a la configuració de l’entrada i sortida per DMA (‘hdma_spi3_rx’ i ‘hdma_i2s3_ext_tx’).

Els paràmetres de cada instància es creen en funcions diferents; en el cas del I2S, és la funció anomenada MX_I2S3_Init; primer es declaren aquestes funcions abans del ‘void setup’, i es criden posteriorment en aquest. En aquest tipus de funcions trobarem les característiques de cada instància, que en definitiva són els paràmetres que haurem configurat en el programa abans de generar el codi.

En el codi, automàticament es genera un apartat de ‘main’, el qual el principi seria el ‘void setup’ (el codi que es genera un cop) i el ‘void loop’, que en definitiva es genera com un ‘while(1)’.

El programa també genera una funció anomenada ‘Error_Handler’, que permet programar un codi per tal d’identificar errors en el programa.

B. PROGRAMA

EL CD que s'adjunta a continuació conté tot el programa en detall, així com tots els arxius generats per el programa CubeMX. L'arxiu main.c es troba a la carpeta Src. En l'adjunt digital d'aquest projecte, també s'ha afegit el codi a la carpeta 1.Memòria\Programa.