

Radio Ducretet

Date: 19 mars 2022

Table des matières

- [1. Présentation](#)
- [2. Protocoles des bus I2C et SPI](#)
 - [2.1. Les commandes I2C](#)
- [3. Câblage](#)
 - [3.1. Câblage Ecran TFT](#)
 - [3.2. Lexique Bus SPI](#)
 - [3.3. Câblage bus SPI](#)
 - [3.4. SelfReturn Button](#)
 - [3.5. Adaptation de niveau pour bus I2C](#)
- [4. Conception hardware](#)
 - [4.1. Conception des schémas \(Eagle free edition\)](#)
 - [4.2. Librairies et composants utiles](#)
 - [4.3. Paramètres des circuits Eagle](#)
 - [4.4. Script Eagle pour Export des Layouts](#)
- [5. Cartes spécifiques](#)
 - [5.1. Présentation](#)
 - [5.2. Carte Alimentation Board](#)
 - [5.3. Carte Bus Board](#)
 - [5.3. Extension Board](#)
- [6. Fabrication](#)
 - [6.1. Réalisation des PCB](#)
 - [6.2. Retour d'expérience](#)
- [7. Utilisation de Radial-V](#)
 - [7.1. Préparation des MP3](#)
- [8. Annexes](#)

1. Présentation

Le système comporte une carte Arduino Mega

- La carte utilise xx I/O (dont 6 avec interruption)
- Elle consomme xx kB de RAM.

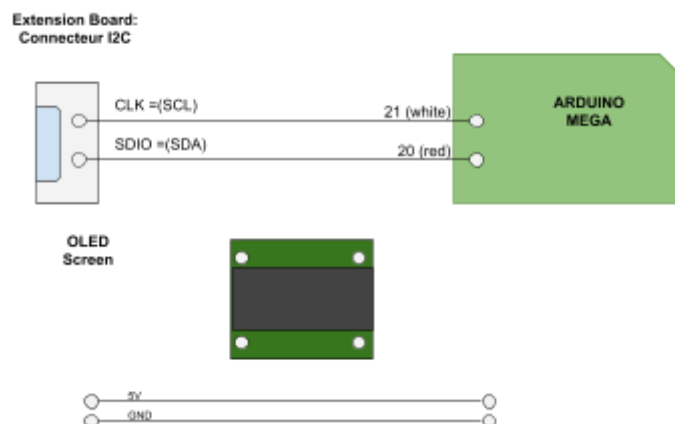
Mega	16Mhz	16 analog	54 I/O	4 PWM	8 kB RAM	256 kB flash
Uno	16Mhz	6 analog	14 I/O	6 PWM	2 kB RAM	32 kB flash
Nano	16Mhz	8 analog	14 I/O	6 PWM	1 kB RAM	16 kB flash

2. Protocole du bus I2C

Le bus I2C est utilisé par l'Arduino pour envoyer des commandes d'affichage à un écran OLED, via la librairie Arduino **Wire.h** et la librairie **U8g2lib.h**.

Cette librairie permet de dessiner sur l'écran OLED des formes géométriques (cercles, rectangles, droites, etc) et des textes dans un grand nombre de polices.

attention à la place prise par les polices dans l'EEPROM



3. Conception hardware


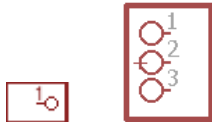




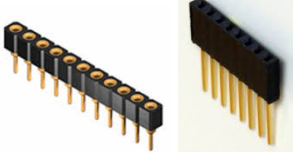
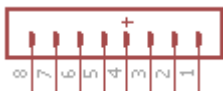


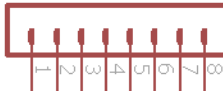

3.1. Logiciels de CAO


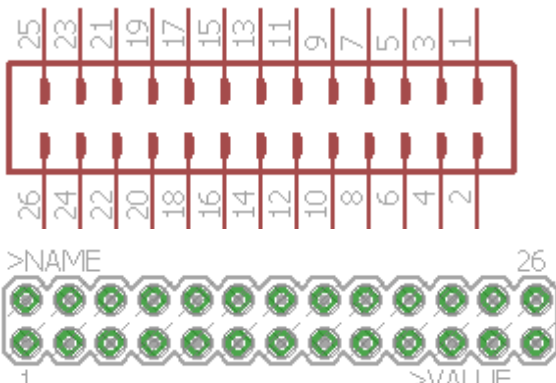



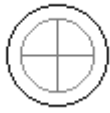

Logiciels utilisés pour la conception des circuits imprimés:

- 2022 : **KiCad** (freeware open source)

3.2. Librairies et Composants utiles

Pour les connecteurs, on peut utiliser:

Composant (Photo)	Libellé KiCad (librairie)	Schéma	Layout
 Pin Header	PINHD-1x1 PINHD-1x3 (library pinheader)		
 Connecteurs Molex 254 KK series	22-27-2031-03 (library con-molex)		
 Female Header	FE08-1 (library cons-lsta)		
 Male header	MA08-1 (library con-lstb)		

Dual Male Header 	MA13-2 (library con-1stb)		
Screw Terminal Block Phoenix 	C-GRID-02-70543 (library con-molex)	>VALUE 	>VALUE 
	Holes		
	VCC, GND (library supply)		

3.3. Paramétrage pour l'édition des PCB

Note sur les unités:

Deux unités sont utilisées: les millimètres (**mm**) et les millièmes de inch (**mil**). Attention à ne pas confondre.

0.4mm = 16 mil

0.8mm = 32 mil

Width (largeur des pistes)

La largeur des pistes dépend de la façon dont les circuits seront fabriqués.

default : width=0.4 mm (réalisation artisanale) ou 0.3mm (réalisation industrielle)

power : width=0.8 mm (réalisation artisanale) ou 0.5mm (réalisation industrielle)

Réalisation artisanale: fabrication par UV et perchlorure.

Réalisation industrielle: fabrication par machine.

Paramétrage Eagle: Edit → Net Classes

Puis, éditer les *properties* des *wires* voulus dans le *schematic*, pour leur donner la *net class* "power".

Clearance (distance entre pistes):

réalisation artisanale:

réalisation industrielle: 2 mm

Drill (perçage pour les pins des composants):

drill = 32 mil = 0.8 mm (résistances, capa, transistors, CI)

drill = 35 mil = 0.9 mm (headers, molex, diodes)

Paramétrage Eagle: Edit → Net Classes

Paramétrage KiCad: Se fait pour chaque empreinte de composant (*footprint*)

Mounting Holes (trous de perçage pour les vis de fixation):

drill = 2.8mm = 110 mil = 0.110 inch

Vias:

perçage (drill) = 32 mil = 0.8 mm

largeur de la pastille (diamètre) = 0.07

Si on doit les souder manuellement: préférer les straps plutôt que les vias, positionner les vias éloignés des pistes et des autres vias (clearance).

4. Cartes électroniques

4.1. Présentation

Alimentation Board

Cette carte est alimentée par l'alimentation générale. Elle répartit l'alimentation vers les différents composants:

- Arduino Mega (9v)
- Ampli (12v)
- Lampes de la face avant (5v)

Extension Board

Cette carte se plugge directement sur l'Arduino. Elle comporte les connecteurs d'entrée/sortie pour les différents capteurs (boutons, Leds, ...) et leurs adaptations.

Elle peut être unifiée avec l'alimentation.

Audio Board

Cette carte fait passer le signal audio par le convertisseur Stéréo→Mono, le potentiomètre de volume et l'amplificateur.

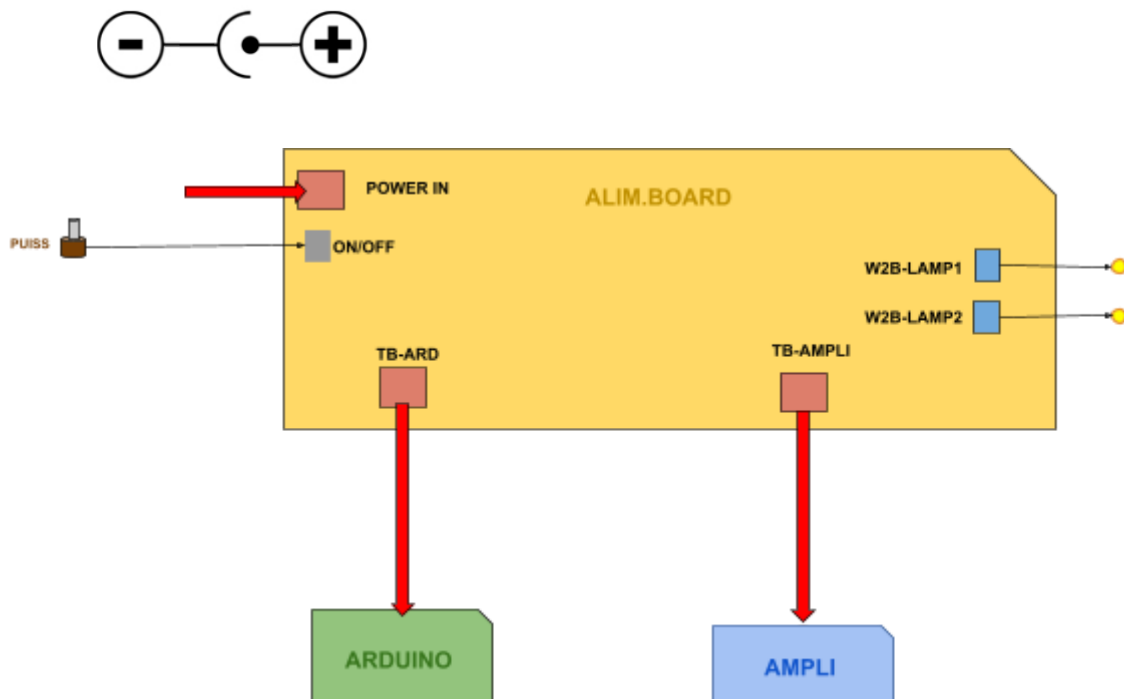
4.2. Carte Alimentation Board

Entrées:

- **POWER IN** (VCC 12v)
- **Bouton ON/OFF**

Sorties:

- **W2B-LAMP1** : Alimentation Lampe 1 (+6v)
 - **W2B-LAMP2** : Alimentation Lampe 2 (+6v)
 - **TB-AMPLI** : Alimentation Amplificateur Audio (12v)
 - **TB-ARD**: Alimentation pour carte Arduino (+6v)
- + Capacité 2200uF sur la sortie 12V (*recommandation Ampli Velleman*)



W2B = Connecteur Wire-to-Board.

4.3. Extension Board

Interfaces:

- Plug de la carte Arduino Mega

Bus d'Interfaces:

- **CNX-I2C** : Bus I2C (+ connecteur de test)

Entrées:

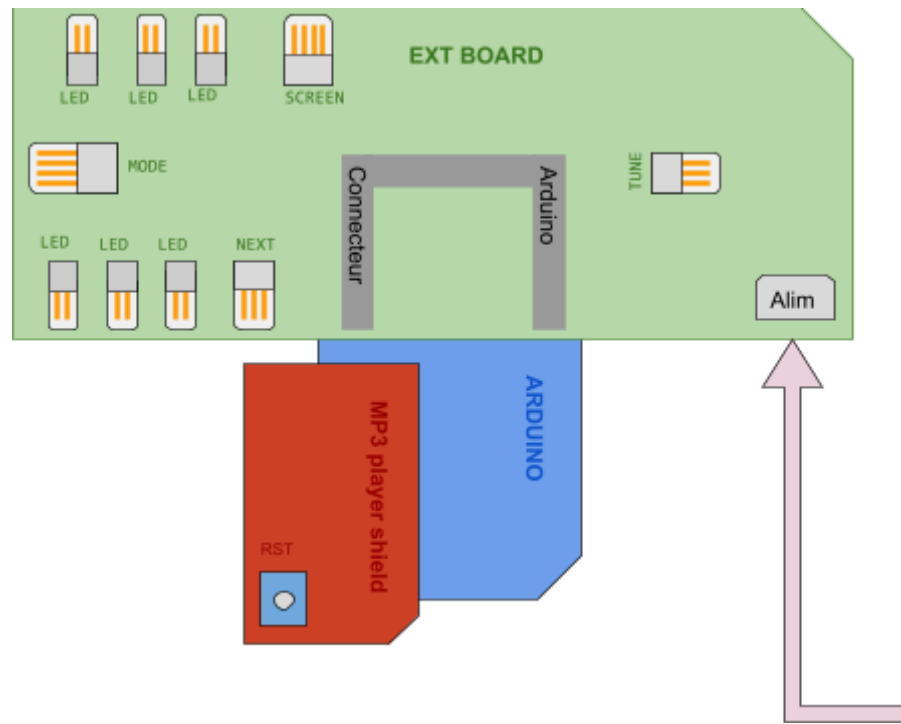
- **CNX-MODE**: commutateur rotatif 3 positions (4 pins)
- **CNX-AGAIN**: commutateur avec retour Next/Again (3 pins).
- **CNX-TUNE**: entrées analogiques pour lecture Capa variable (3 pins)

Sorties:

- **LEDs 1-2-3** : LEDs ambres (Beat, Beat x4, Beat x16)
- **LED 4** : LED (témoin d'activité carte SD du catalogue)
- **LED 5** : LED (spare)
- **LED 6** : LED (spare)

Configuration:

- **Jumper** : Sélection des Capa mesurés pour le Tuning

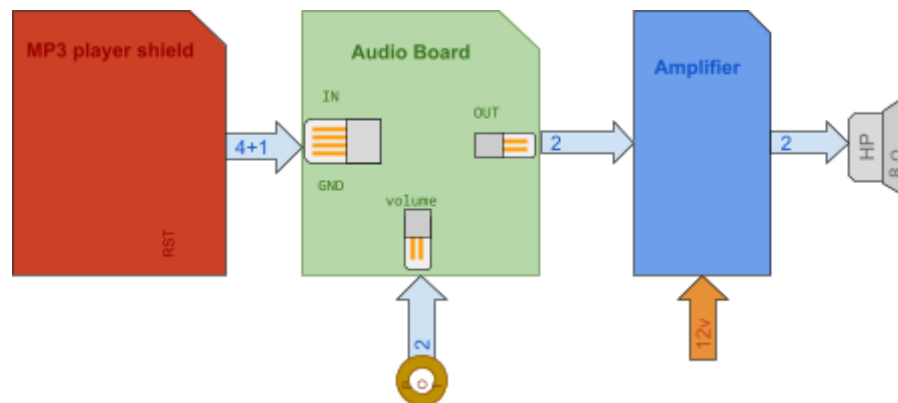


4.4 Carte Audio board

Description

Cette carte envoie le signal audio entrant vers:

- un étage de fusion stéréo → mono
- un étage d'adaptation (voir ci-dessous). Etait recommandé pour le Sparkfun. A voir si ça l'est aussi pour l'Adafruit...
- un Potentiomètre de contrôle de volume (47 k Ω LOG)
- un Amplificateur mono



Entrées / Sorties

- **Audio IN** : Signal audio stéréo issu de la carte MP3: Left + / - et Right + / - (4 pins)
- **GND** : GND de la carte Arduino (ou MP3)
- **CNX-POT** : Potentiomètre de volume
- **Audio OUT** : Signal audio Mono vers Ampli

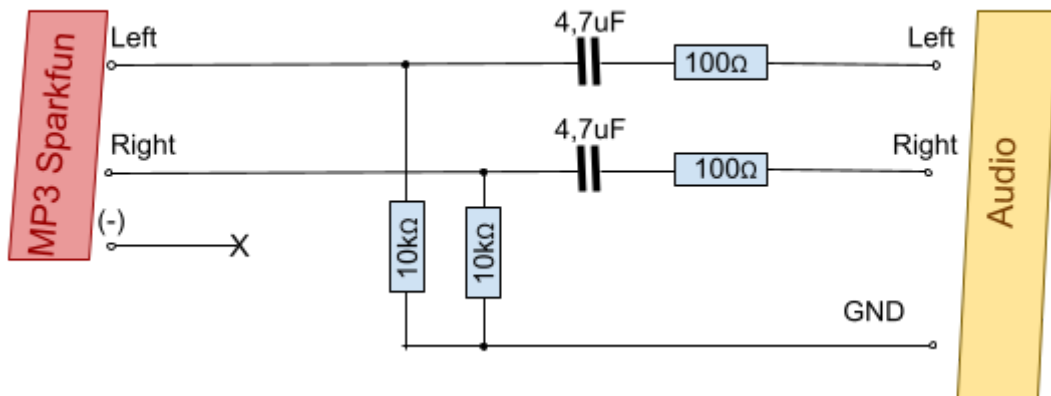
Configuration

- **Jumper** : Sélection GND envoyé à l'ampli: celui de la carte MP3 ou celui de l'Arduino ?
A voir après les tests...

Etage fusion

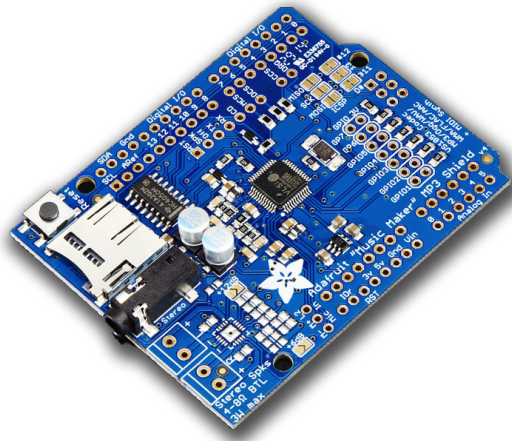
Sur Radial-V : L'étage de fusion **Stéréo** → **Mono** ne fonctionne pas: les signaux L et R s'annulent réciproquement. Après avoir essayé différentes valeurs de R1 et R2 (de 100 Ω à 10k Ω), seule la voie gauche a été conservée.

Etage Adaptation



4.5 Carte MP3 Player

Cette carte comporte un player MP3 et une carte SD. Elle est alimentée en 5v.
La carte est disponible chez [Adafruit](#) (Produit [1790](#)).



Connectique :

Elle se plugge directement sur l'Arduino.

- Chip select SPI: module SD, module player MP3, MP3 data
- Bus SPI (SCLK, MISO et MOSI)
- Alimentation 5v

MISO	MOSI	SCLK	MP3 CS	MP3 data CS	SD CS	Data Request	MIDI IN	GPIO	RESET
Bus SPI	Bus SPI	Bus SPI	SPI select MP3 player	SPI: select data for MP3	SPI: select SD reader		N/U	N/U	Reset

Software:

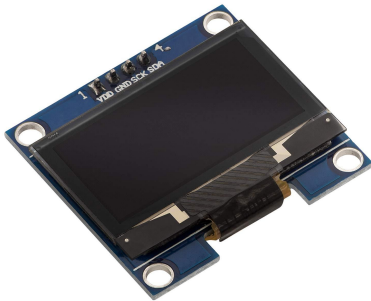
Pour utiliser le shield, il faut ajouter au code:

- La librairie **SPI** (gestion du bus SPI)
- Les librairies **SdFat** et **SdFatUtil** (gestion du lecteur SD)
- La librairie **SFEMP3Shield** de William Greiman (gestion du chip player mp3)

Pour les tests, cette carte peut être simulée par *bouchon.h*

4.6 Ecran OLED

Cet écran OLED monochrome 128x64 est alimenté en 5v et piloté via un bus I2C. Diagonale de 1.3" (3,7 cm). C'est un écran OLED de [AZ Delivery](#).



AZ-Delivery
for Experts for Microelectronics

Connectique :

Cette carte comprend un connecteur 4 pins:

VCC	GND	SCK	SDA
Alim 5v	Alim 0v	Bus I22 : clock	Bus I2C : data

Software:

Pour utiliser la carte, il faut ajouter dans le code:

- La librairie **WIRE** pour la gestion du protocole I2C.
- La [librairie u8g2](#) pour les commandes primitives d'affichage.

4.7 Carte Amplificateur

Cette carte est un amplificateur audio mono BF, alimenté entre 8 et 18v (ici 12v).
C'est un kit Velleman ([WSAH4001 / VM114](#))

Puissance de sortie RMS = 2W / 8Ω

Gain = 86dB

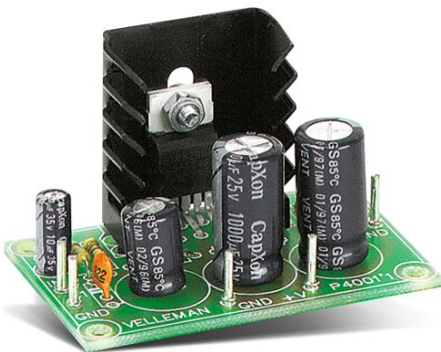
Connectique :

La carte comporte 6 connecteurs.

GND	+V	GND	IN	GND	LS
POWER 0v	POWER 12v	AUDIO IN Ground	AUDIO IN signal (Pot. volume)	AUDIO OUT Ground (Haut parleur)	AUDIO OUT signal (Haut parleur)

Software:

Aucun.



Remarque:

La qualité du son est très sensible à la qualité et à la stabilité de l'alimentation.
Les câbles audio doivent être courts et blindés.

4.8 Lecture de la valeur de la Capa variable

On utilise un algorithme basé sur la mesure du temps de décharge
Deux pattes de la capacité sont reliées à 2 pins analogiques de l'Arduino.

Pin "Charge" : On envoie des créniaux de 5 volts sur cette pin.

Pin "Discharge":

Etat OUTPUT: qui revient à mettre une grosse résistance.

Etat INPUT: (faible résistance + capacité parasite stray).

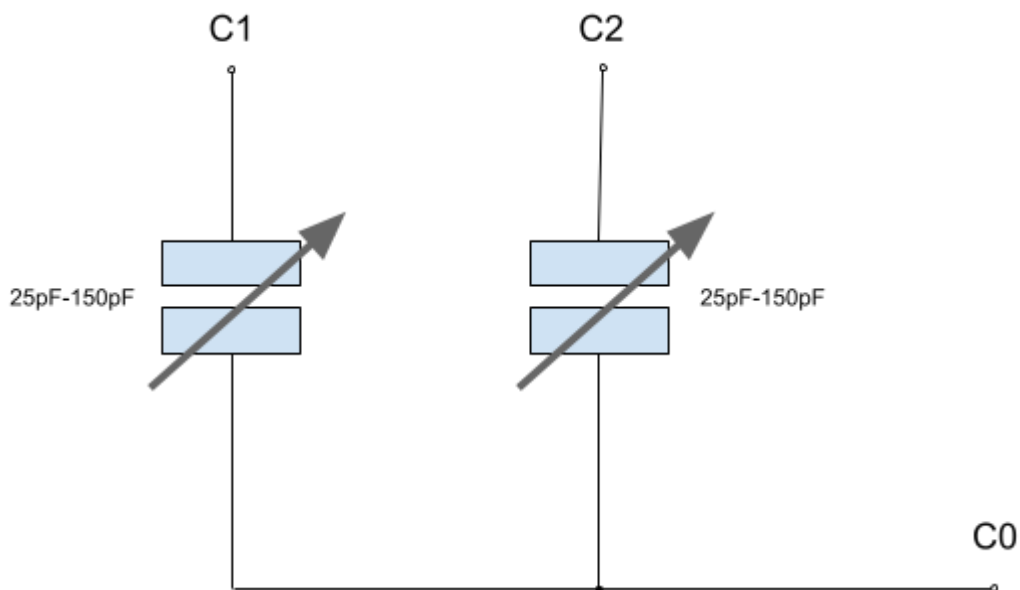
On lit la valeur de la tension au bout de 0,1ms.

Les deux capacités évoluent en même temps, de façon similaire, lorsque l'on tourne le bouton de Tuning.

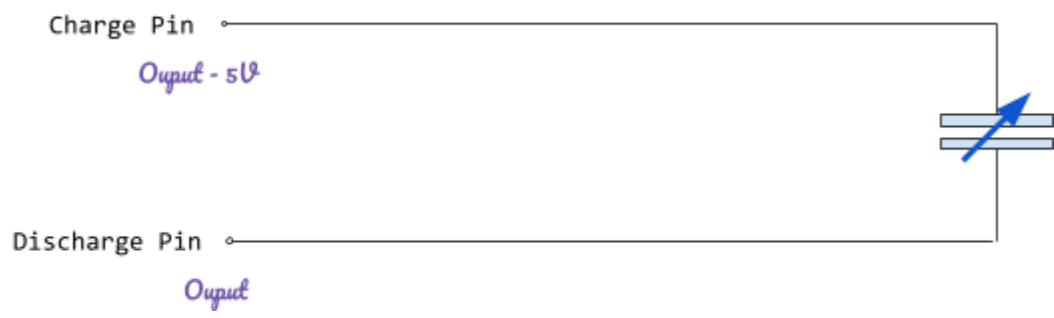
On mesure la capacitance entre les broches **C1 et C2**.

On prévoit un circuit pour vider la capa: Relier momentanément les deux pins de la capa, via une résistance de 10 Ω . (non réalisé: trop complexe)

Schéma de la capacité variable:



1 : Phase de Charge



6. Fabrication

6.1. Réalisation des PCB

J'ai trouvé deux sites français qui proposent la réalisation de PCB.

<https://pcbshopper.com/> est un comparateur de fabricants.

- <http://bxtronics.free.fr>
e-tronic est limité à des PCB simple-face, en 35 et 70µm.
Ses tarifs sont de 14€/dm² + outillage (5€) + 100 trous (3.50€) + port (3.50€)
Perçage = 0.8mm par défaut.
Sérigraphie = non
Étamage = gratuit (à froid)
Découpe = gratuit
- http://www.util-pocket.com/electronique/circuit_imprime.htm
util-pocket fait simple-face et double face, en 35µm.
Ses tarifs sont de 14.50€/dm² + outillage (3.60€) + port (4.10€)
Perçage = 0.8mm par défaut.
Étamage = gratuit (à chaud).
Vias (trous) non métallisés.
- <https://jlcpcb.com>
Entreprise chinoise aux tarifs très compétitifs.
Réalise les PCB par 5 unités minimum.
Ses tarifs sont de 5€/dm² + port (8€)
Perçage inclus (fournir fichier drill).
Sérigraphie incluse.
Vias métallisés.

6.2. Retour d'expérience

PCB réalisés par util-pocket:

Délai : 4 semaines.

Qualité OK.

Pas de sérigraphie. Perçage à refaire. Vias à souder.

Tarifs: 36 € au total pour les 3 boards.

Fabrication à partir des fichiers .brd.

PCB réalisés par jlc-pcb

Délai : 6 semaines.

Qualité OK.

Sérigraphie. Perçage fait selon drill-map. via métallisés.

Tarifs: 8€ au total pour 2 boards.

Fabrication à partir des fichiers gerber et drill.

Le perçage (drills)

Util-pocket a percé tous les trous en 0.8mm. Certains nécessitent du 1mm (headers, connecteurs...), que j'ai dû repercer. C'est pas très grave.

JLC-PCB utilise le fichier .drill pour percer chaque trou avec le diamètre demandé.

les Vias:

Util-pocket les perce en 0.6mm : c'est OK.

Par contre la largeur de cuivre des pastilles que j'avais défini, était à 0.4mm : c'est trop petit. Il aurait fallu 0.6mm. Le métal se décolle si on chauffe trop. La soudure du via est difficile.

JLC-PCB les perce en 0.4mm: c'est OK car ils sont déjà métallisés et ne demandent aucun post-traitement (ni soudure).

Autres points:

- Ne pas mettre 2 vias trop proches. (par ex: 1mm c'est trop proche). Positionner **Clearance Via-Via** à 64mil (1.6mm), par exemple, dans Design Rules.
- Favoriser deux vias formant un "pont" où l'on pourra mettre un cable (strap).
- Ne pas faire de connexion "fil-carte" **à un seul point** avec un header: le fil se casse.
- Mettre un Test Point (header 2 pins) "+VCC/GND" sur chaque carte pour favoriser les vérifications.
- Eviter de mettre de la sérigraphie sur les pistes. C'est peu lisible.
- Sérigraphie: prendre des caractères de dimensions 1mm ou 1.2mm (la dimension par défaut de KiCad 0.8mm est trop petite).

7. Utilisation de la Radio

Même type de carte SD que pour Radial-V.

Ajouter la gestion des Beat.

7.1. Carte SD pour les MP3

Structure de la carte SD

Les fichiers MP3 sont sur une carte SD (16 GB) formatée en **FAT32**.

Répertoire racine = fichier **Catalog.ndx**

Répertoire **Music** = fichiers mp3 (dont NOISE.MP3)

Contrainte du driver SD : les noms de fichiers doivent être en 8+3

Fichiers particuliers

Fichier **NOISE.MP3** = fichier de quelques secondes, joué lorsque l'application n'a pas réussi à trouver le fichier mp3 à jouer.

Fichier **Catalog.ndx** = fichier index dont la préparation est décrite ci-après.

Structure du fichier Catalog

Le fichier est trié selon les **années**.

Le format de chaque ligne est le suivant:

YEAR[4];HASCHCODE[8];GENRE;RATING[1];
--

Exemple:

1927;E72C4A38;Ragtime;3;

Cas particuliers:

0000 : année inconnue

<ERROR> : le hash n'a pas pu être calculé

0 : rating non renseigné

Préparation de la carte SD

Les MP3 doivent être copiés sur une carte micro-SD accompagné d'un fichier index.

Un utilitaire permet de générer cet index, et de copier les fichiers MP3 sur la carte SD en les renommant en 8.3.

- Lancer l'utilitaire **mp3tag**
- Sélectionner l'ensemble des clips mp3 à mettre sur la carte SD (Ctrl-A).
- Cliquer sur **Export**, et choisir **radial_catalog**
- Ceci génère un fichier **catalog.txt** (contenant: Année, Filename, Genre et Rating)
- Cliquer sur **Export** et choisir **radial_transfert**
- Ceci génère un fichier **TransfertFiles.txt** (au format UTF-16).

Pour bien gérer les accents dans les filename, ce fichier de commande a besoin d'être converti.

- Exécuter **ConvertTransfertBatch.bat** qui génère **TransfertFiles.bat**.

Conversion en CP850 avec la commande:

```
iconv -f UTF-16 -t CP850 music\TransfertFiles.txt > TransfertFiles.bat
```

Pour copier les fichiers sur la carte SD:

- Exécuter **TransfertFiles.bat**:
- Ce script génère le catalogue final **Catalog.ndx**
 - Tri du catalog selon le champ Year
 - Copie les fichiers mp3 sur la carte SD (I:\Music) en les renommant.
 - Copie le catalogue sur la carte
 - Copie quelques bruitages sur la carte SD (NOISE.mp3, par exemple)

Note: le passage par utf-16 permet de convertir correctement les noms de fichiers qui comportent des caractères accentués. (le fichier batch devant comporter des caractères au format CP850).

Nécessite de disposer de l'utilitaire **iconv.exe**

8. Annexes: Utilitaires

8.2 Utilitaire BMP PistonSoft

PistonSoft permet de détecter le rythme (Beat) d'un morceau MP3.

- Site web: <http://www.imagemagick.org/script/binary-releases.php>
- Script **InvertImage.bat**

8.3 Utilitaire Mp3Tag

MP3Tag permet de gérer les tags des mp3, et possède un système de scripting très poussé.

- Script **radial_catalog.mpe** (génère *Catalog.txt*)
- Script **radial_transfert.mpe** (génère *TransfertFiles.txt*)
- Site web: www.mp3tag.de/en/

8.4 Utilitaire iconv

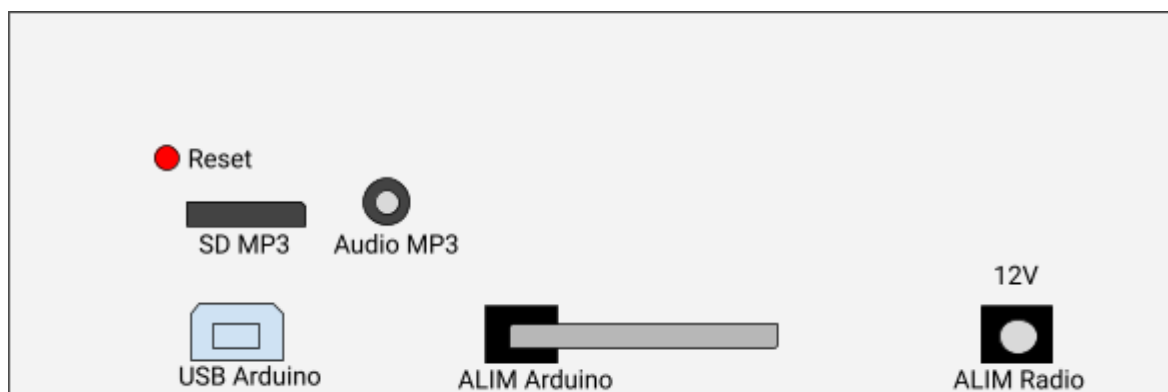
Cet utilitaire permet de convertir des fichiers ANSI en OEM, etc.

Utile pour gérer des caractères accentués dans des batch qui s'exécutent dans une fenêtre DOS en Code page 850..

Installer la commande *iconv* en téléchargeant *libiconv-1.9.2-1.exe* à partir de ce lien : <http://gnuwin32.sourceforge.net/downloads/libiconv.php>

- Script **ConvertTransfertBatch.bat**

9. Connectique



Port USB de debug ARDUINO : Accès direct sur la carte en USB-C (115200 bauds).

Barillet alimentation ARDUINO : Normalement connecté à un câble venant de la carte Alimentation. On est un peu obligé de passer par l'extérieur car l'accès à la pin VIN de l'arduino est bloqué par la carte MP3.

Carte SD MP3 : Permet de modifier le catalogue ou d'ajouter des morceaux MP3.

Sortie Audio MP3 : Sortie audio de contrôle (Prise casque).

ALIM Radio : Prise d'alimentation. Connecter une alimentation continue 12-15v.

10. Réalisation de la face arrière (en plexi)

A voir