
Rapport de Projet

Ecole supérieure
Électronique

Salle R112

Métronome Electronique

Projet N° : 2318

Réalisé par :

Julien Decrausaz

A l'attention de :

M. Bovey
M. Moreno
M. Castoldi

Dates :

Début : 16 novembre 2023

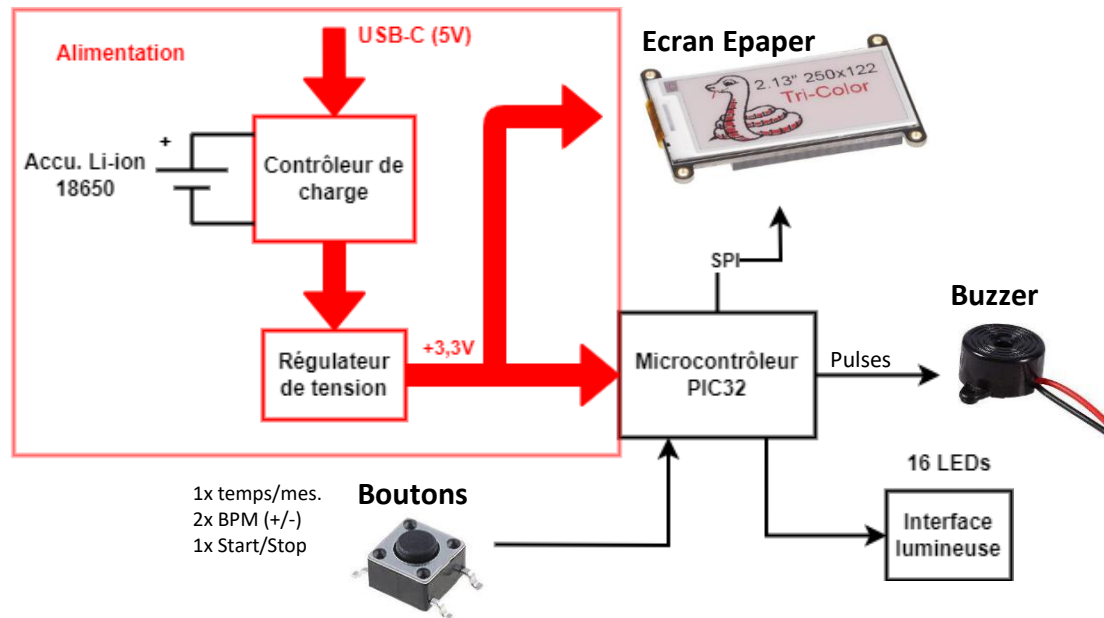
Fin : 1^{er} février 2024

Table des matières

1 Pré-étude.....	3
1.1 Schéma bloc détaillé et description des blocs	3
1.2 Choix technologiques clés du système	4
1.3 Représentation des interactions du système avec le monde extérieur	5
1.4 Estimation des coûts	6
1.5 Planning du projet.....	7
2 Design	8
2.1 Principe et caractéristiques	8
2.2 Schéma bloc modifié.....	8
2.3 Schématique	9
2.3.1 Alimentation	9
2.3.2 Général	11
2.3.3 Microcontrôleur.....	14
2.4 Concepts du logiciel	16
2.5 Perspectives et travail restant	17
2.5.1 Travail restant.....	17

1 Pré-étude

1.1 Schéma bloc détaillé et description des blocs



Alimentation :

Le système est alimenté via un connecteur USB-C 5V et dispose d'un système de stockage par accu Li-ion 18650. La tension passe premièrement par un contrôleur de charge suivi d'un régulateur de tension 3,3V qui alimentera le système.

µC :

Le microcontrôleur que je vais utiliser sera de la famille des PIC32. Il devra contenir assez d'I/Os pour permettre de connecter le reste de mes composants étant donné qu'il y a déjà 16 sorties pour mes LEDs.

Ecran Epaper :

L'écran utilisé est de style Epaper. Il affichera le nombre de BPM (Battements/Beats par minute).

Buzzer :

Un buzzer servira d'interface son. Des pulses de différentes fréquences lui seront transmises. Le Stress First Beat sera d'une fréquence plus élevée que les autres afin de marquer le début de la mesure.

Boutons :

4 boutons seront utilisés pour ce module. Le premier servira à régler le nombre de temps par mesure, deux autres serviront à régler le nombre de BPM désiré et le dernier servira à démarrer/stopper le métronome.

Interface lumineuse (LEDs) :


Composée de 16 LEDs, elles indiqueront chacun des temps de la mesure. Une LED rouge marquera le début de la mesure et des LEDs vertes serviront pour les autres temps.

Boîtier :

Le système devra être intégré dans un boîtier (pas de contrainte d'étanchéité). Il devra contenir la batterie ainsi que toute l'électronique. Il aura des ouvertures afin que l'écran, les LEDs et les boutons soient visibles. Des trous devront être créés afin de laisser passer le son du buzzer ainsi que le connecteur USB-C. Le boîtier pourra être acheté ou à concevoir.

1.2 Choix technologiques clés du système

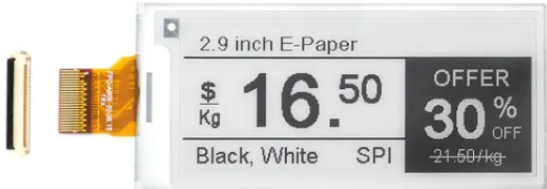
Le composant auquel je dois faire attention est l'écran Epaper. L'Ecran que je voulais initialement n'est plus disponible chez le fournisseur. Je me suis dirigé vers un substitut : le GDEY029T94 de chez Good display. C'est un écran rigide de 2,9 pouces avec 4 nuances de gris à temps d'actualisation rapide. Il consomme très peu de courant lorsqu'il est en mode veille (entre 1 et 5 μ A) et conserve la dernière image affichée.



2.9 inch e-paper screen 296x128 customizable 1.5s fast refresh Raspberry pi SPI display, GDEY029T94

This is a e ink screen, 2.9 inch, with 296x128 resolution, SSD1680, SPI interface. The 2.9" e ink supports 4 grayscales and partial refresh. It can be customized to achieve 1.5s fast refresh.

★★★★★ 12 Review(s) Sold:256






Price:

USD ▾ **\$8.70**

Wholesale:

1-9	10-49	50+
\$8.70	\$8.46	\$8.26

Product:

Quantity:

(1713 available)

Paramètre	Conditions	Min.	Typ.	Max	Unité
Full update time	25°C		3		sec
Fast update time	25°C		1,5		sec
Partial refresh time	25°C		0,3		sec

Tableau 1 : spécification des taux d'actualisation de l'écran (valeurs tirées de la datasheet)

Il dispose de plusieurs sortes d'actualisations :

- Full update time : Permet d'actualiser entièrement l'écran.
- Fast update time : Permet d'actualiser entièrement l'écran de manière rapide.
- Partial refresh time : Permet d'actualiser une partie de l'écran (bien pour un chronomètre par exemple).

Le fabricant conseille d'utiliser une Full Update après 5 utilisations de Fast update ou de Partial refresh afin de réduire l'accumulation d'images différées.

Dans le cas où cet écran ne serait pas compatible avec les librairies existantes à l'ES (pour sa programmation) je vais prévoir un connecteur SPI/I²C pour LCD classique.

1.3 Représentation des interactions du système avec le monde extérieur

Eléments visibles du boîtier :

- 4 boutons : BPM- ; BPM+ ; Temps/Mes ; Start/Stop
- LCD Epaper (Affichage du nombre de BPM)
- 16 LEDs (1 rouge et 15 vertes)
- Trous pour haut-parleur (H.P.)
- Trou pour le connecteur USB-C

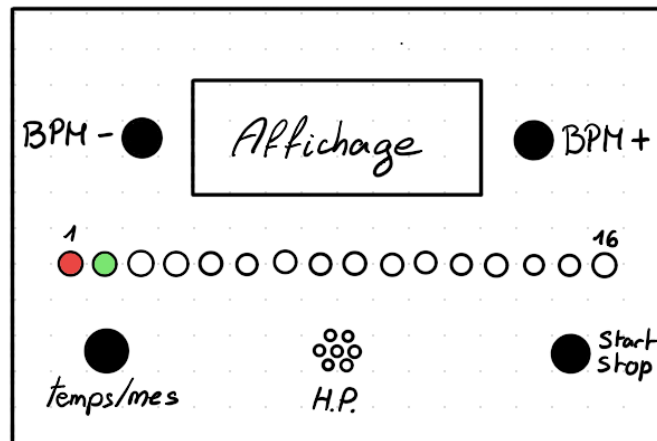


Figure 1 : Illustration de la face avant du boîtier

1.4 Estimation des coûts

J'ai choisi de n'inclure que les composants importants de mon module pour estimer les coûts car ce sont les plus chers et donc ceux qui auront un fort impact sur le prix final.

Composant	Description	Fournisseur	N° de série fournisseur	Prix à l'unité	Prix/u pour 20p
Microcontrôleur PIC32MX				4.00 CHF	3.89 CHF
Ecran Epaper + Adapter Board	2.9" E-Paper Display	GooDisplay	GDEW029T5D	13.88 CHF	13.51 CHF
Accu. Li-ion				10.00 CHF	9.73 CHF
PCB		Eurocircuit		50.00 CHF	48.65 CHF
Boîtier				10.00 CHF	9.73 CHF
TOTAL CHF				87.88 CHF	85.51 CHF

1.5 Planning du projet

No semaine projet	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27						
No Semaine	46	47	48	49	50	51	52	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26
Date	16.11.2023	23.11.2023	30.11.2023	07.12.2023	14.12.2023	21.12.2023	28.12.2023	14.01.2024	11.01.2024	18.01.2024	25.01.2024	01.02.2024	08.02.2024	15.02.2024	22.02.2024	29.02.2024	07.03.2024	14.03.2024	21.03.2024	28.03.2024	04.04.2024	11.04.2024	18.04.2024	25.04.2024	02.05.2024	09.05.2024	16.05.2024	23.05.2024	30.05.2024	06.06.2024	13.06.2024	20.06.2024	29.06.2024
Tâches							V	V						V							V	V				V							
Pré-étude				R	P																												
Rédaction du cahier des charges																																	
Choix des composants + commande des composants critiques																																	
Estimation des coûts																																	
Rédaction du rapport de pré-étude																																	
Rédaction du PowerPoint																																	
Design + Schéma												R	P																				
Conception des schémas électriques																																	
Choix des footprints																																	
Rédaction du rapport de design																																	
Rédaction du PowerPoint																																	
PCB																		R															
Placement des composants																																	
Routage																																	
Optimisation du PCB																																	
Envoi des fichiers de fabrication																																	
Commande du PCB																																	
Montage																																	
Software																																	
Réalisation des diagrammes / structo																																	
Programmation du software																																	
Tests et mise au point																																	
Rédaction du rapport																																	
Préparation présentation + démo																																	
Présentation finale																																R	P
Finalisation/correction/documentation																																	

2 Design

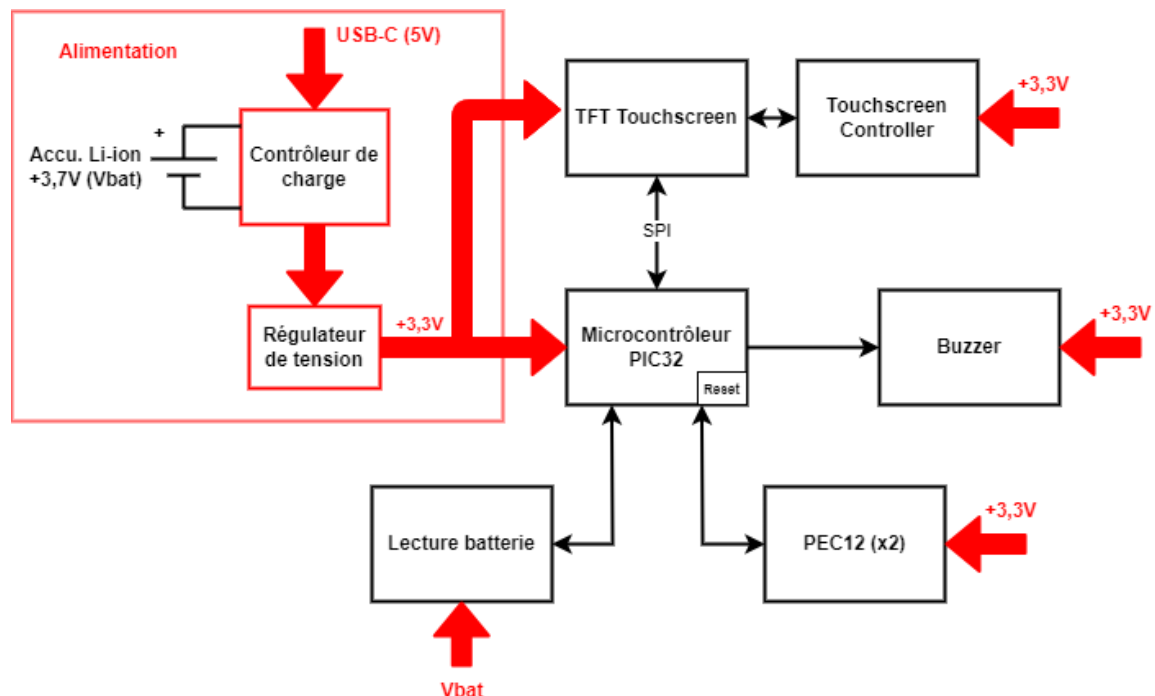
2.1 Principe et caractéristiques

Ce métronome se base sur le modèle du site :

<https://www.imusic-school.com/fr/outils/metronome-en-ligne/>

Il dispose de configurations permettant à l'utilisateur de le paramétrer comme bon lui semble. Il dispose d'un écran TFT Touchscreen tactile offrant une interface minime et facile à utiliser. Deux PEC12 permettent de modifier le nombre de BPM (Beats Per Minutes) ainsi que le rythme du métronome comme sur l'interface du site web (croche, double-croche, etc...). Il peut varier entre 20 et 300BPM et dispose d'une plage de mesures entre 2 et 16 temps par mesure. Le produit est muni d'un boîtier fermé.

2.2 Schéma bloc modifié



2.3 Schématique

2.3.1 Alimentation

2.3.1.1 Alimentation USB-C et batterie

Power Supply USB-C to Battery

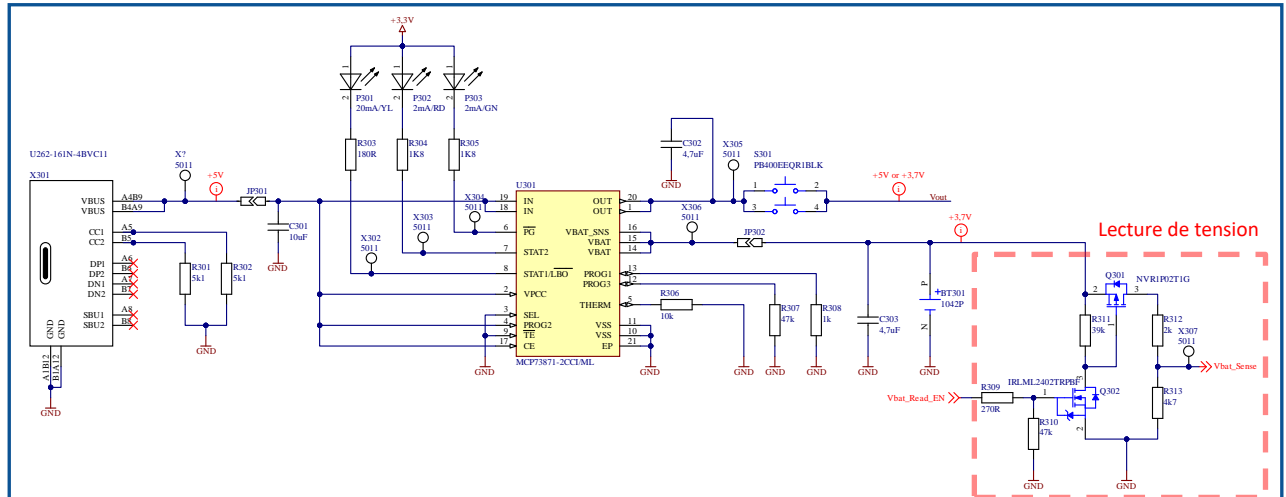


Figure 2

Le métronome peut être alimenté par un connecteur USB-c +5V ou par un accu. +3,7V. La tension passe d'abord par un contrôleur de charge, le MCP73871, qui servira à charger la batterie pendant que l'USB-c est branché. Un schéma de lecture de tension de la batterie est présent afin de garantir son bon fonctionnement. Selon si l'USB-C est actif ou inactif, la tension de sortie varie entre 3,7V (batterie) et 5V (USB-C). Cette tension va ensuite dans un convertisseur de tension pour l'abaisser à 3,3V.

2.3.1.1.1 Lecture de tension de la batterie

Ce montage est repris du projet de M. Zaco (N°2206).

Relié au microcontrôleur, il permet d'activer la lecture de la batterie. Permettant ainsi à la batterie de ne pas être lue en continu et de ne pas tirer un courant constamment sur celle-ci. La sortie du microcontrôleur pilote le MOSFET-N Q302 qui pilote le MOSFET-P Q301.

Le diviseur de tension R312 et R313 est calculé de cette manière :

- Uref = 3V (tension de référence de l'ADC, plage de variation de l'ADC de 0 à 3V)
- Ubat max = 4.2V (tension max sur la batterie)
- R313 = 4.7kΩ (choisie arbitrairement)
- R312 = la résistance à calculer

$$R312 = \frac{U_{bat\ max} - U_{ref}}{\frac{U_{ref}}{R313}} = \frac{4,2 - 3}{\frac{3}{4700}} = 1880\Omega \xrightarrow{E48} 2k\Omega$$

2.3.1.1.2 Contrôleur de charge MCP73871

J'ai choisi ce composant car il a déjà été utilisé dans d'autres projets. Ce qui me donne un aperçu du montage et de la schématique des composants autour.

Pin Number	Symbol	I/O	Function
1, 20	OUT	O	System Output Terminal
2	VPCC	I	Voltage proportional charge control
3	SEL	I	Input type selection (low for USB port, high for AC-DC adapter)
4	PROG2	I	USB port input current limit selection when SEL = Low (Low = 100 mA, High = 500 mA)
5	THERM	I/O	Thermistor monitoring input and bias current
6	PG	O	Power Good Status Output (Open-Drain)
7	STAT2	O	Charge Status Output 2 (Open-Drain)
8	STAT1/LBO	O	Charge Status Output 1 (Open-Drain). Low battery output indicator when $V_{BAT} \geq V_{IN}$
9	TE	I	Timer Enable; Enables Safety Timer when active-low
10, 11, EP	VSS	—	Battery Management 0V Reference. EP (Exposed Thermal Pad). There is an internal electrical connection between the exposed thermal pad and VSS. The EP must be connected to the same potential as the VSS pin on the Printed Circuit Board (PCB)
12	PROG3	I/O	Termination set point for both AC-DC adapter and USB port
13	PROG1	I/O	Fast charge current regulation setting with SEL = high. Preconditioning set point for both USB port and AC-DC adapter
14, 15	VBAT	I/O	Battery Positive Input and Output connection
16	VBAT_SENSE	I/O	Battery Voltage Sense
17	CE	I	Device Charge Enable; Enabled when CE = high
18, 19	IN	I	Power Supply Input

Legend: I = Input, O = Output, I/O = Input/Output

Note: To ensure proper operation, the input pins must not allow floating and should always tie to either high or low.

Figure 3 : Pin description du MCP73871 (tiré de la datasheet p17)

Les pins SEL, PROG2, /TE et CE sont des pins pouvant être mises à l'état bas ou haut selon l'utilisation désirée.

- **SEL :** Permet de sélectionner le type de source en entrée pour la limite de courant. Avec le pin en mode HIGH, le MCP73871 est capable de fournir un courant de 1,65A (typical) pour la charge de la batterie Li-Ion et le système. A l'état LOW, la source d'entrée est conçue pour fournir de la tension au système et de charger la batterie depuis un port USB. Dans mon cas, je lui mets un état LOW.
- **PROG2 :** est le pin de régulation du courant d'entrée. Un état logique LOW sélectionne une limite de courant de 100mA et un état logique HIGH limite le courant à 500mA. Dans mon cas, 100mA suffisent.
- **/TE :** pour « Timer Enable » permet d'activer/désactiver un timer interne. LOW pour actif et HIGH pour inactif. Je n'en ai pas l'utilité pour ce projet.
- **CE :** pour « Charge Enable ». Avec le pin à l'état LOW, la charge de la batterie est désactivée. Je l'ai donc mis à l'état haut pour activer la charger.

Comme je n'utilise pas de mesure de température, j'ai remplacé la résistance NTC du pin THERM par une résistance de même valeur (10kΩ).

Les entrées /PG, STAT2 et STAT1 servent d'indicateurs. Des LEDs y sont connectées afin d'avertir l'utilisateur d'un éventuel problème. J'ai choisi trois couleurs de LEDs différentes afin de les différencier. J'ai choisi les LEDs disponibles à l'ES puis j'ai calculé leur résistance en fonction de leur courant de fonctionnement et la tension de 3,3V.

$$R_{LED_jaune} = \frac{U}{I} = \frac{3,3}{20 \cdot 10^{-3}} = 165\Omega \xrightarrow{E24} \mathbf{180\Omega}$$

$$R_{LED_rouge} = R_{LED_verte} = \frac{U}{I} = \frac{3,3}{2 \cdot 10^{-3}} = 1650\Omega \xrightarrow{E24} \mathbf{1800\Omega}$$

2.3.1.2 Convertisseur +3,3V

Linear Voltage Regulator | +3,3V output

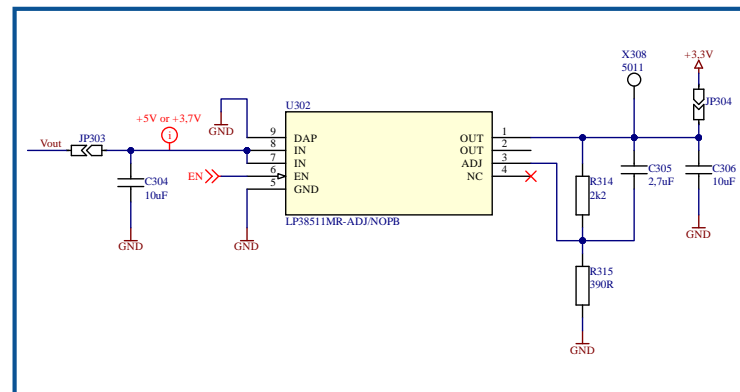


Figure 4

Ce convertisseur permet d'alimenter tous les composants du métronome. Il sera alimenté de 3,7V (avec la batterie) et 5V (avec USB). J'ai choisi ce modèle car je l'avais utilisé lors d'un précédent projet et qu'il avait bien fonctionné. Pour dimensionner le pont diviseur R314 et R315, j'ai utilisé la formule de la datasheet du composant en page 10.

Avec $V_{ADJ} = 500\text{mV}$:

$$V_{out} = V_{ADJ} \cdot \left(1 + \frac{R314}{R315}\right) = 0,5 \cdot \left(1 + \frac{2200}{390}\right) = \boxed{3,32V}$$

Après discussion avec les professeurs, il a été décidé que l'entrée EN serait gérée par le microcontrôleur.

2.3.2 Général

Lors de la phase de pré-étude ainsi que dans le cahier des charges, il était indiqué qu'un Ecran Epaper serait utilisé. Cependant, et après discussion avec Messieurs Moreno et Castoldi, nous avons conclu qu'un écran Epaper ne convenait pas à mon projet. C'est pourquoi ils m'ont orienté vers un écran tactile TFT. Grâce à cet écran, j'ai pu réduire grandement le nombre de composants présent dans mes schémas électriques. Car la plupart des réglages et paramètres s'effectueront via l'écran. J'aurai alors plus de Software à concevoir.

L'écran choisi est un ILI9341 de chez Adafruit. C'est un écran de 2,8" fonctionnant en SPI ou 8bits. Selon sa datasheet, l'écran doit utiliser un « Touchscreen Controller » car il n'en a pas. Adafruit suggère d'utiliser le TSC20071. C'est donc ce composant que j'ai utilisé pour mon schéma.

L'écran servira d'interface avec l'utilisateur. A partir de celui-ci, il pourra régler le nombre de temps par mesure, ainsi que le volume du buzzer.

2.3.2.1 Touchscreen

Touchscreen

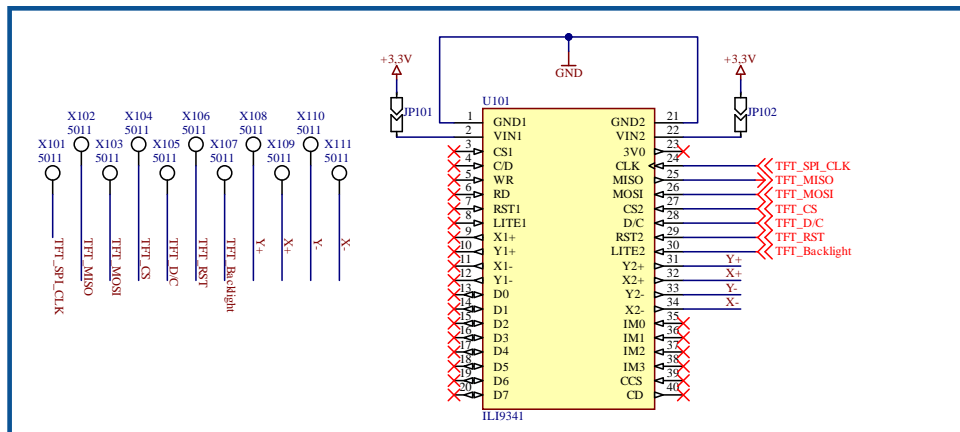


Figure 5 : Ecran TFT tactile

Comme je vais utiliser l'écran en SPI, tous les pins du côté gauche de l'écran (Figure 5) doivent être en mode no-connect. Les pins X2+, X2-, Y+ et Y- sont reliés TSC20071. Tous les autres pins sont reliés au microcontrôleur sur des entrées SPI afin de gérer l'écran. J'ai placé des points de tests sur chacun de ces pins dans le but de pouvoir mesurer les signaux y passant lors d'un éventuel dépannage.

2.3.2.2 Touchscreen Controller

Le « touchscreen controller » utilisé est le TSC20071 proposé par Adafruit. Il permet de convertir les données des pin X et Y en resistive touch. En consultant sa datasheet, j'ai connecté chacun des pins selon ce qui était proposé. Les pins SCL et SDA sont connectées à des résistances de Pullup de 10kΩ. Les pins A0 et A1 correspondent aux pins d'adresse déterminant les deux LSB de l'adresse du composant. Elles ont une adresse initiale qui peut être modifiée en les connectant au 3,3V. Dans mon cas, je n'ai pas besoin de les changer. C'est pour cela que j'ai ajouté des jumpers SMD entre les résistances de Pullup et l'alimentation. Dans l'éventuel cas où les adresses devaient être modifiées, il suffit de souder les jumpers.

Touchscreen Controller

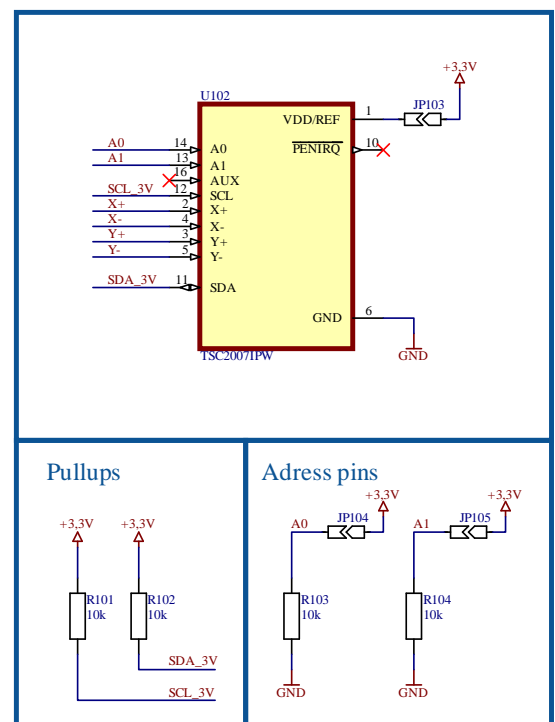


Figure 6

2.3.2.3 Buzzer

Le buzzer que j'utilise est le SMTB-0827. Il a déjà été utilisé sur des projets précédents. C'est un buzzer magnétique de petite taille commandé par un MOSFET-N. Sur sa Gate, j'ai ajouté une Pulldown de **100kΩ** (R106) afin de le bloquer lorsque l'entrée est flottante lors du démarrage du uC. R105 sert de limitation de courant.

J'ai tenu compte du courant maximum que les pins de mon uC peuvent supporter pour la dimensionner (25mA). J'ai trouvé cette valeur dans la datasheet du uC en page 183.

$$R105 = \frac{V_{cc}}{I_{max}} = \frac{3,3}{25 \cdot 10^{-3}} = \mathbf{132\Omega}$$

Buzzer

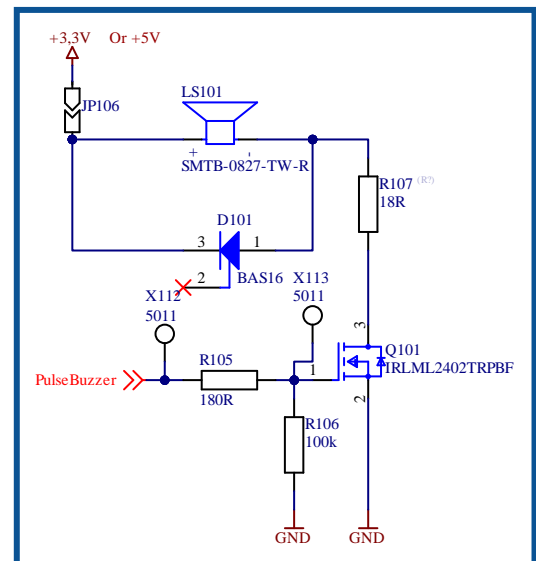


Figure 7

Par question de sécurité, je vais utiliser une valeur plus élevée pour R105. J'ai choisi **180Ω**.

Comme j'utilise un buzzer magnétique, une diode de roue libre est présente afin de décharger le courant présent dans sa bobine. J'ai utilisé un modèle de diode disponible en stock à l'ES.

La résistance R107 permet de limiter le courant sur le buzzer. Pour son dimensionnement, j'ai tenu compte du courant maximum supporté par le buzzer (100mA). Une fois la résistance obtenue, je vais soustraire la valeur de la résistance de la bobine (valeur datasheet de **17Ω**) à celle-ci.

$$R107 = \frac{V_{cc}}{I_{max_{buzzer}}} - R_{Bobine} = \frac{3,3}{100 \cdot 10^{-3}} - 17 = \mathbf{18\Omega}$$

2.3.3 Microcontrôleur

Le microcontrôleur que j'utilise est le **PIC32MX795F512H** qui possède 64 pins. J'ai choisi ce modèle car il dispose d'assez de pin SPI pour le Touchscreen ainsi qu'assez d'I/Os. Initialement, je devais utiliser un écran Epaper. Je comptais utiliser 16 I/Os connectées à des LEDs pour l'affichage des temps par mesure. Toutefois, j'utilise désormais un TFT Touchscreen. J'ai choisi d'afficher ces 16 temps par mesure sur l'écran plutôt que d'utiliser toutes ces I/Os. C'est pour cela que j'ai beaucoup de pins non connectés.

2.3.3.1 Reset et debug

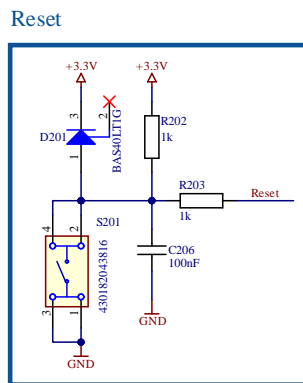


Figure 8

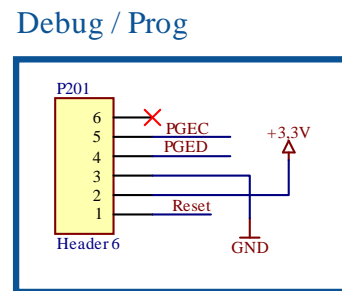


Figure 9

Ces montages sont repris de la carte du kit de l'ES et sont également utilisés sur d'autres projets. Le reset permettra de redémarrer le microcontrôleur manuellement. Un connecteur de Debug/Programmation est présent afin d'implémenter le futur Software.

2.3.3.2 PEC12

Je vais utiliser deux PEC12 afin de régler le nombre de BPM et le mode du métronome. J'ai choisi de ne pas mettre ces paramètres réglables à l'écran afin de ne pas surcharger l'écran d'informations. J'ai repris le montage du PEC12 présent sur le kit PIC32 de l'ES.

Le premier PEC12 (ENC201), règle le nombre de BPM (Beats Per Minutes) allant de 20 à 300.

Le deuxième PEC12 (ENC202) permet à l'utilisateur de choisir le mode dans lequel le métronome va être utilisé. Comme sur le site web de référence, il y aura 5 modes : Croche, Double croche, Triple croche, Quadruple croche et Croche Ternier.

PEC12

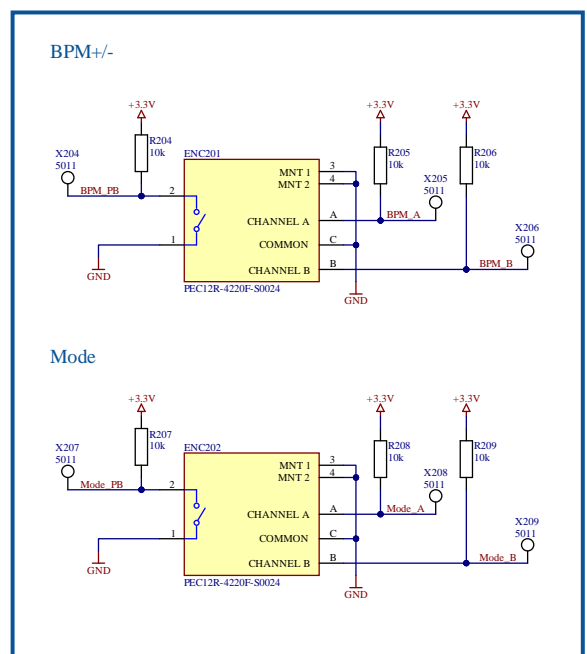


Figure 10

2.3.3.3 Définition des pins du microcontrôleur

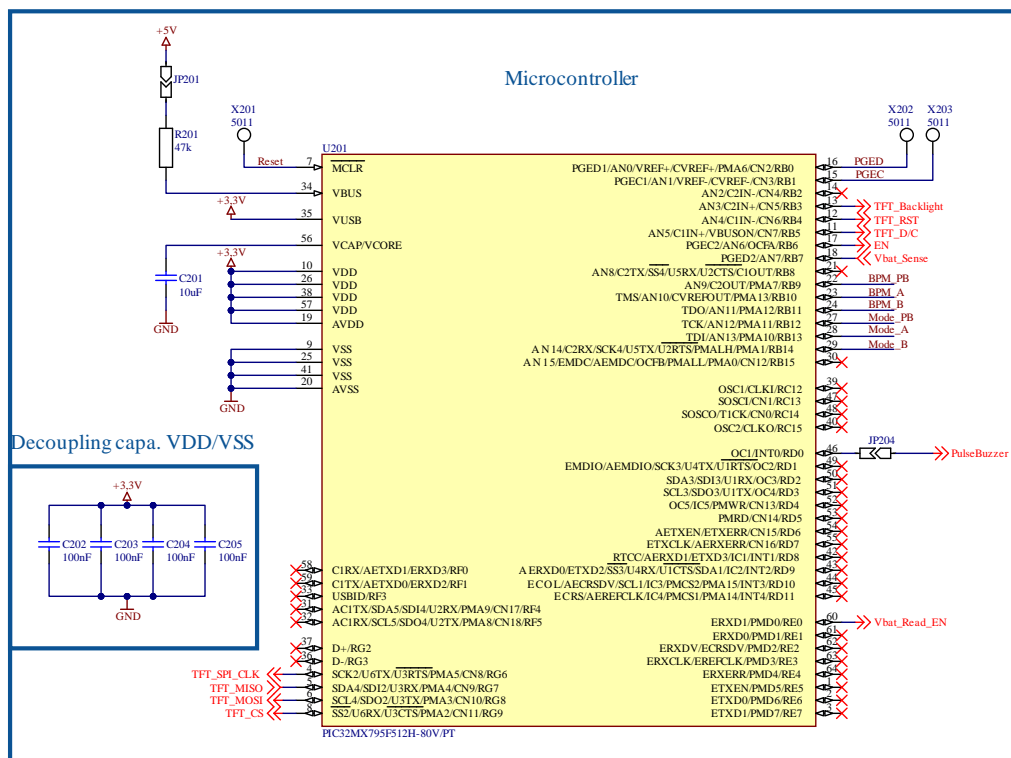


Figure 11

- Unavailable
- Available
- Assigned

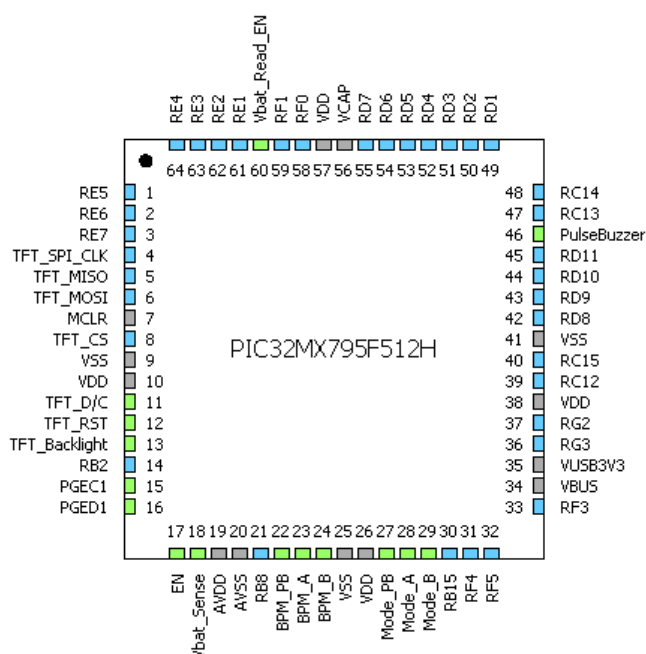


Figure 12 : Pin assignment

2.4 Concepts du logiciel

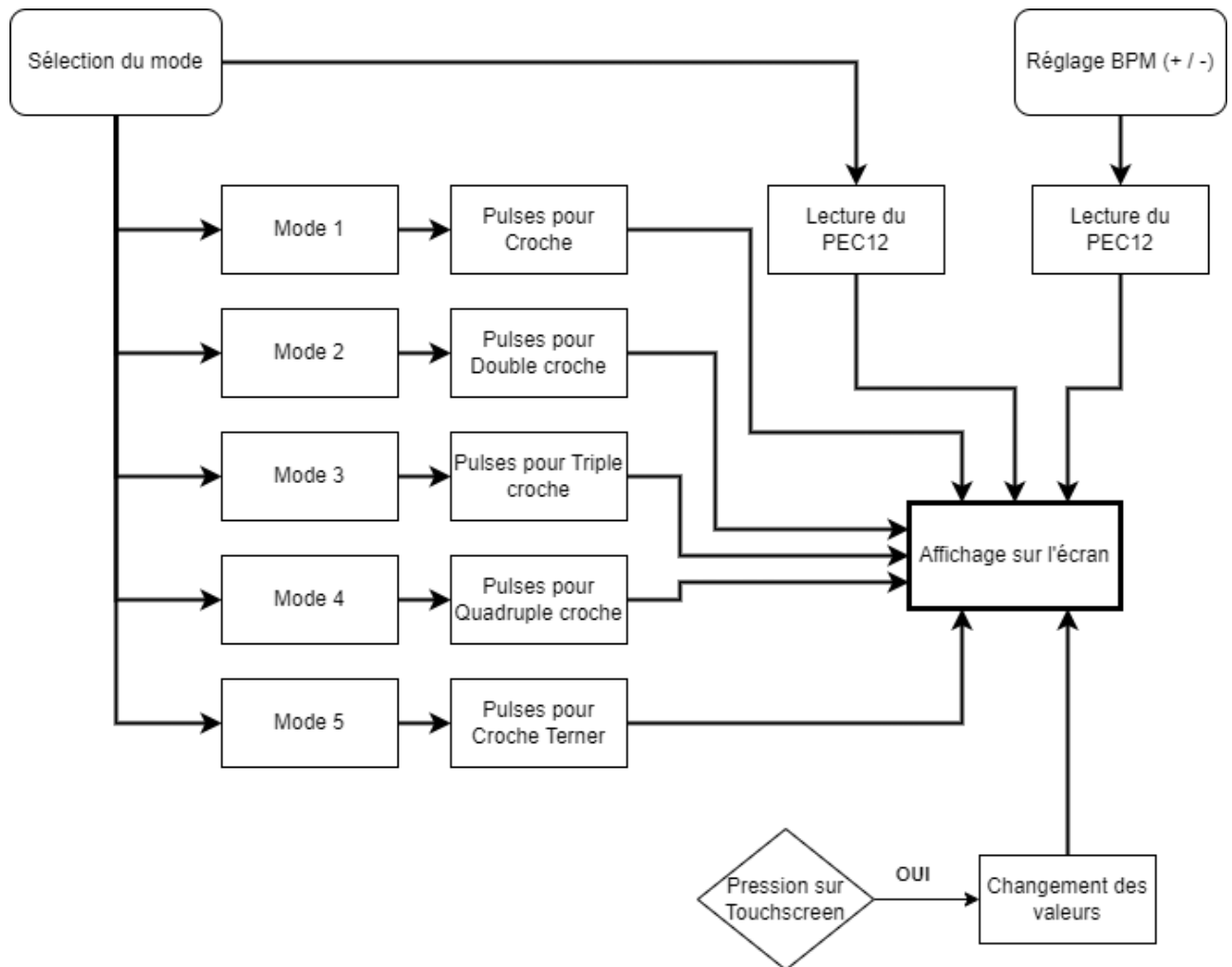


Figure 13

2.5 Perspectives et travail restant

2.5.1 Travail restant

1. Finir de choisir les composants et réaliser la BOM
2. Ajouter les footprints manquants sur le schéma
3. Choisir Accu Li-Ion
4. Commander les composants critiques
5. Design et routage du PCB
6. Monter le PCB et effectuer les tests hardwares
7. Réaliser le Software
8. Tests et mesures finaux

Lausanne, le 1^{er} février 2024

Julien Decrausaz