
Rapport technique

POLYTECHNIQUE
MONTREAL

UNIVERSITÉ
D'INGÉNIERIE



Séquenceur numérique de huit pas

ELE3000 - Projets personnels en génie électrique

Automne 2020

Département de génie électrique

École Polytechnique de Montréal

Dernière mise à jour: 22 décembre 2020

Michael **Flandez**

1825018

Table des matières

1	Identification du problème	4
1.1	Mise en contexte	4
1.1.1	Musique Électronique	4
1.1.2	Théorie Musicale	5
1.1.3	Synthétiseur Analogique	5
1.1.4	Synthétiseurs Modulaires	6
2	Spécifications fonctionnelles	8
3	Design préliminaire	10
3.1	Différents approches de résolution	10
3.2	Étude de praticabilité de la solution retenue	10
3.3	Diagramme de flux de données	11
3.3.1	Entrées	11
3.3.2	Sorties	11
3.4	Description des modules	12
3.4.1	Sélection de mode	12
3.4.2	Horloge externe	12
3.4.3	Horloge interne	13
3.4.4	Lecture de front montant	13
3.4.5	Transmission	14
4	Design détaillé	16
4.1	Conceptualisation ou modélisation des éléments	16
4.1.1	Entrées et sorties	16
4.1.2	Le horloge externe	17
4.1.3	Multiplexage	18
4.1.4	Microcontrôleur	19
4.2	Réalisation du prototype	20
4.2.1	Électrique	20
4.2.2	Logiciel	21
4.3	Réalisation préliminaire et itérations du design	22
5	Validation	23
5.1	Procédures de test et résultats	23
5.2	Analyse et validation des résultats	23
5.3	Analyse des résultats en fonction des hypothèses initiales	24
6	Conclusion	24
6.1	Évaluation des limites et des incertitudes	24
6.2	Proposition des nouvelles connaissances et habiletés acquises durant le projet	25

7 Apprentissage continu

25

Table des figures

1	Diagramme bloc d'un synthétiseur. En entrée on voit le musicien/utilisateur qui joue le synthétiseur par un clavier qui rentre ensuite dans un filtre pour finalement passer à l'amplificateur	6
2	Diagramme bloc d'un synthétiseur modifié. En entrée on voit le séquenceur qui joue le synthétiseur, ensuite le signal rentre dans un filtre pour finalement passer à l'amplificateur.	6
3	Diagramme de flux de données	11
4	Schéma électrique d'un bouton poussoir et une résistance de 12 k Ω en mode Pull down	16
5	Schéma électrique d'un potentiomètre et un interrupteur	16
6	Schéma électrique d'un DEL et un connecteur audio	17
7	Schéma électrique d'un connecteur audio et deux transistors en configuration d'interrupteur	18
8	Schéma électrique d'un multiplexeur 8 à 1	18
9	Schéma logique d'un multiplexeur 8 à 1	19
10	Schéma électrique du microcontrôleur Arduino UNO	19
11	Schéma électrique composée sur Kicad©	20
12	Allure général du programme	22
13	Écart maximal entre les valeurs de tension trouvées à la sortie SOMME	23
14	Valeurs mesurées des impulsions à la sortie SOMME TRIG	24

1 Identification du problème

Ce séquenceur de huit pas est un module qui répond aux besoins des musiciens et artistes électroniques. Il est conçu comme un module à être ajouté à un système modulaire ou semi-modulaire supportant des entrées en tension telles que des impulsions ou échelons. Dans un contexte musical, le séquenceur enverra des séquences de huit pas en boucle programmables par l'utilisateur par moyen des potentiomètres vers d'autres modules capables de produire du son. L'exemple classique est pour la création des mélodies sans avoir besoin d'un instrumentiste.

Le module aura 8 pas et suivra une horloge en entrée (BPM : battements par minute) pour changer d'un pas à l'autre dans la boucle. À chaque pas, grâce à la position d'un potentiomètre, il enverra une impulsion et un échelon en sortie (deux sorties indépendantes pour chaque pas) que l'utilisateur pourra assigner ou envoyer vers d'autres modules. En plus, il y aura une sortie SOMME qu'enverra la combinaison des toutes les pas dans une même sortie. De cette façon, l'utilisateur pourra utiliser la séquence complète de huit pas ou utiliser des pas différents pour des fonctions différentes.

Ce module aura deux modes d'opération. Dans le premier mode MAÎTRE, il sera autonome, dans le deuxième mode ESCLAVE il aura une horloge externe en entrée qui dictera les BPM à suivre. En ce mode l'utilisateur pourra utiliser un autre module MAÎTRE pour contrôler la cadence du séquenceur et de la boucle. Ainsi on pourra avoir une mélodie qui se répète constamment en synchronisation avec la composition de l'ensemble.

Le séquenceur lira en entrée des fronts montants de l'horloge externe et des niveaux de tensions sélectionnées par les potentiomètres qui seront assignés vers les sorties par un microcontrôleur. Un programme attribuera aux sorties individuelles et principales des échelons et des impulsions en forme analogique. Il y aura des diodes électroluminescentes (DEL) pour chaque pas, indiquant quel pas est activé(allumé).

1.1 Mise en contexte

À continuation, on nombre des aspects importants pour mettre un peu plus en contexte ce design de séquenceur de huit pas.

1.1.1 Musique Électronique

La musique électronique est un type de musique qui utilise des générateurs de signaux et des sons synthétiques au lieu des instruments musicaux traditionnels. Ces sons donnent une expérience grâce à la multiplicité et originalité dès qu'on peut générer à partir des instruments électroniques.

Essentiellement, il est possible pour un musicien électronique d'utiliser plusieurs modules de son indépendants, synchronisés entre eux ou pas.

Parmi les familles d'instruments électroniques le plus importants, on retrouve les synthétiseurs musicaux, les échantillonneurs et les boîtes à rythmes. Les synthétiseurs musicaux émettent des tons à l'aide des oscillateurs ou générateurs d'ondes à des fréquences audibles entre 20 Hz et 20 kHz, les échantillonneurs émettent des sons en se servant des échantillons préenregistrés et les boîtes à rythmes s'en servent de ces deux techniques pour créer des rythmes en boucle avec des sons de batterie.

Les boîtes à rythmes incluent un séquenceur pour pouvoir être utilisées directement en boucle tandis que seulement quelques synthétiseurs et échantillonneurs les ont intégrées. Dans le cas où le séquenceur est absent, c'est au musicien de jouer un instrument à la fois. Étant donnée la limite physique de ceci, il devient utile de pouvoir avoir ses instruments fonctionnant en même temps.

1.1.2 Théorie Musicale

La théorie musicale pourrait se définir comme un ensemble de règles étudiant les rapports ou relations entre fréquences.

Donc d'une fréquence donnée, on peut trouver une autre fréquence qui est la plus consonante (plus agréable à l'oreille) à une octave d'intervalle. Cet intervalle est un rapport de fréquences deux à un. La norme d'occident place la note A_4 à $440Hz$, donc l'octave suivante A_5 se trouve à $880Hz$.

$$A_4 = 440Hz$$

$$A_5 = 2 * A_4 = 880Hz$$

Donc on voit bien une relation logarithmique de l'aspect fréquentiel, à chaque octave on double la fréquence.

Les normes de musique électronique telle que faite populaire par le très connu producteur des synthétiseurs Robert Moog et maintenant utilisées par la grande majorité des synthétiseurs modulaires ('*Eurorack*') se basent sur le rapport $1V/octave$. Donc à chaque volt d'augmentation, l'on augment également d'une octave.

L'approche occidentale indique qu'entre une octave et la suivante il y a 12 notes équidistantes géométriquement. Cette distance introduit la notion de ton. La distance minimale entre une note et une autre suivante est d'un demi-ton. Cette distance d'un demi-ton se traduit comme suit.

$$demi - ton = \frac{1octave}{12notes} = \frac{1Volt}{12} = 0.0833Volt$$

1.1.3 Synthétiseur Analogique

Comme expliqué précédemment, un synthétiseur émet du son grâce à l'utilisation d'un oscillateur ou générateur de fonctions. Ensuite, ce signal pourrait être filtré, amplifié, etc. Les blocs fondamentaux sont les suivants.

- VCO : Oscillateur électronique qui génère un signal périodique dont la fréquence se stabilise en fonction de la tension d'entrée.
- VCF : Filtre fréquentiel dont la fréquence de coupure est contrôlée par la tension en entrée.
- VCA : Amplificateur dont le gain varie en fonction d'une tension d'entrée.

Une configuration simple de synthétiseur est présentée sur la figure suivante. En général, le synthétiseur compte avec un clavier, et avec un ou plusieurs VCO, VCF et VCA.

En entrée on a le musicien/utilisateur qui joue le synthétiseur par un clavier. Lorsqu'une touche est actionnée, l'oscillateur émet une onde d'une certaine fréquence. Ce signal passe au filtre où le son est sculpté pour ensuite passer à l'amplificateur pour être amplifié.



FIGURE 1 – Diagramme bloc d'un synthétiseur. En entrée on voit le musicien/utilisateur qui joue le synthétiseur par un clavier qui rentre ensuite dans un filtre pour finalement passer à l'amplificateur

1.1.4 Synthétiseurs Modulaires

Au lieu d'avoir un synthétiseur qui vient avec ses trois étages (VCO, VCF, VCA), on cherche à se procurer ces différents modules indépendamment. Ceci amène à un degré de liberté supérieur pour configurer un ensemble de modules. En suivant la norme la plus populaire à ce jour ('*Eurorack*' on peut installer ces modules qui respectent des consignes de tension de 12volts, et utilise le standard 1V/oct. Là, il s'agit d'utiliser des fils externes de 1/8" pour faire des interconnexions au goût personnel de l'utilisateur.

Et là la question apparaît, *comment peut-on faire pour synchroniser et faire jouer plusieurs modules en même temps et sans clavier ?*

Voilà encore une fois le but du séquenceur. Le séquenceur donne la possibilité pour la création de mélodies qui jouent en boucle, sans se fatiguer ou se tromper. En plus, en utilisant plusieurs façons de contrôler les modules, les sons résultants sont beaucoup plus complexes, riches, voire intéressants.

Avant d'avancer, il faut expliquer les deux types de signaux utilisés dans les interconnexions.

CV : tension de contrôle. C'est un signal de tension utilisé pour contrôler un autre module. Par exemple, il pourrait être utilisé pour contrôler la fréquence de coupure d'un filtre.

Impulsion/GATE : déclenchement d'une action. Ce signal est utilisé par des modules qui sont activés par des fronts montants ou descendants. Un exemple est activé une note à jouer lorsqu'un module reçoit une impulsion (comme le fait un clavier).

Alors, il est possible d'utiliser un séquenceur pour envoyer une sortie CV vers un synthétiseur pour faire jouer une mélodie, en boucle. À chaque fois qu'on veut que la note soit jouée, il faudrait aussi l'envoyer un signal GÂTÉ.

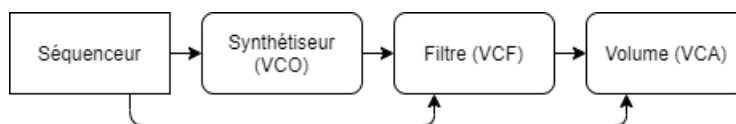


FIGURE 2 – Diagramme bloc d'un synthétiseur modifié. En entrée on voit le séquenceur qui joue le synthétiseur, ensuite le signal rentre dans un filtre pour finalement passer à l'amplificateur.

Sur cette image on voit qu'on c'est maintenant le séquenceur qui joue une mélodie à l'oscillateur. En plus, le séquenceur envoie des signaux de contrôle aux modules de filtrage et d'amplification.

La dernière notion à approfondir est l'horloge et la synchronisation. Comme dans la musique traditionnelle, on s'attend à avoir un 'directeur' qui dirigera la vitesse auquel la musique sera jouée.

Dans la musique électronique, il y aura le module avec l'horloge MAÎTRE qui enverra des signaux GATE vers les autres modules. Les autres modules ESCLAVE liront les fronts montants de cette horloge.

2 Spécifications fonctionnelles

Séquenceur numérique analogique de 8 pas : Module créateur de séquences rythmiques et mélodiques.

Pour les amateurs de la musique électronique, sont-ils des musiciens ou des amateurs, il y a en ce moment un intérêt croissant pour de nouveaux modules à inclure dans leurs systèmes de synthèse. Ces modules sont distingués par leurs attributs tels que leur fonctionnalité, leur taille, leur degré de difficulté, etc. La question qu'on se pose fondamentalement comme utilisateur est si le module est plus intéressant pour ce qu'on essaie de faire qu'un autre module similaire ?

Les modules se divisent en deux familles : les sources et les processeurs.

Les sources : Ces modules se caractérisent par avoir au moins une sortie, mais pas d'entrée (ex : un oscillateur ou un séquenceur). Il pourrait y avoir aussi des entrées de contrôle.

Ici on peut nommer entre autres : des sources de son comme les oscillateurs contrôlés par voltage ou générateur des fonctions audibles (VCO : voltage-controlled oscillator) ou à basse fréquence (LFO : low-frequency oscillator), des sources de bruit, des générateurs d'enveloppe (ADSR EG : attack, decay, sustain, release envelope generator) et des séquenceurs.

Les processeurs : Ces modules se caractérisent par avoir une entrée et une sortie de signal. Il pourrait avoir aussi des entrées de contrôle.

Par exemple on trouve ici : des filtres contrôlés par voltage (VCF : voltage controlled filter), des amplificateurs contrôlés par voltage (VCA : voltage controlled amplifiers), des mélangeurs pour additionner des signaux (MIXER), et encore une fois des séquenceurs.

Les séquenceurs Alors pour quoi les séquenceurs tombent dans les deux familles ? Ceci est dû au fait que les séquenceurs peuvent avoir deux modes d'opération : mode MAÎTRE et mode ESCLAVE. En mode MAÎTRE, le séquenceur est activé manuellement (bouton-poussoir) et commence à rouler indépendamment à une vitesse sélectionnée donc là on peut le qualifier de source. En mode ESCLAVE, il est activé seulement lorsqu'il reçoit un signal en entrée pour le faire changer d'état.

Normalement, les séquenceurs n'ont qu'une sortie d'où sortent toutes les impulsions. Et ils fonctionnent en mode esclave. Ceci pourrait être assez comme fonctionnalité dépendamment de ce qu'on cherche à faire dans l'ensemble des modules. C'est là où la pensée critique amène au questionnement de la validité de ce module dans la configuration. En exemple : un module peut-être intéressant, mais peut-être qu'on aimerait avoir un plus de fonctionnalités. Si on souhaite le garder dans le système, il nous sera plus intéressant de pouvoir décortiquer la séquence et d'avoir plus de signaux de sortie pour contrôler plus les autres modules.

L'attribut le plus intéressant qui n'est pas vu dans d'autres séquenceurs modulaires est la présence des sorties indépendantes, des échelons et des impulsions, pour chaque pas (huit en total). Il y a aussi une neuvième sortie qui reproduit toutes les impulsions et tous les échelons. Ceci permet de créer de mélodies très facilement et de les envoyer à plusieurs modules en même temps.

1. Entrées générales :

- MODE : bouton poussoir qui sélection entre modes esclaves ou maître.
- CLOCK EXT : horloge externe. Des impulsions générées par un autre module qui signalera au séquenceur d'avancer au pas suivant. (MODE : ESCLAVE)

- BPM : potentiomètre permettant de réguler les battements par minute en mode esclave entre 60 et 220 BPM. (MODE : MAÎTRE)
 - RESET : réinitialiser la séquence au premier pas.
 - Longueur de la séquence : potentiomètre permettant de sélectionner le nombre de pas, entre 1 et 8 pas.
 - TON : potentiomètre qu'indique au système la hauteur (note musicale) que l'utilisateur voudrait avoir pour ce pas. Il y en a un TON par pas.
 - SOMME : switch qu'assigne le pas sélectionné à la sortie principale SOMME.
2. Sorties :
- Sortie principale :
- TRIG : sortie d'où sortent les impulsions de tous les pas.
 - TON : sortie d'où sortent échelons de tous les pas.
 - DEL : lumière qui s'illumine avec l'intensité de l'échelon courant.
- Sorties par pas :
- TRIG : sortie d'où sort l'impulsion du pas.
 - TON : Sortie d'où sort l'échelon de ce pas.
 - DEL : lumière que s'illumine lorsque la séquence passe par ce pas.
3. Fonctions :
- Changer de mode.
 - Allumer la fonction de BPM si en mode ESCLAVE.
 - Calcule les battements par minute en mode ESCLAVE.
 - Lire un front montant (MODE : ESCLAVE) ou une instruction de l'horloge interne (MODE : MAÎTRE).
 - Passer d'un pas au pas suivant en forme ascendante, du pas un jusqu'au pas huit en boucle.
 - Réinitialiser la séquence à un.
 - Lire le potentiomètre et l'état de bouton SOMME pour tous les pas.
 - Envoyer en sortie une impulsion et un échelon au pas courant et à la sortie principale.
4. Facteurs humains :
- L'utilisateur sélectionnera en premier le mode d'opération, sinon le séquenceur commencera en mode MAÎTRE.
 - Si en mode esclave il utilisera un module externe pour envoyer un signal d'horloge entrée.
 - Ensuite il fera des ajustements à son goût avec le potentiomètre pour sélectionner de mode générale le TON pour chaque pas.
 - Il branchera les sorties vers d'autre module à l'aide de fils électriques.
5. Réactions aux erreurs :
- En cas de changement de mode lorsqu'en train de rouler, le séquenceur s'arrêtera et attendra une nouvelle instruction.
 - Défaut du mode ESCLAVE : la séquence se met au pas un et attends une instruction de l'entrée CLock EXT.
 - Défaut du mode MAÎTRE : la séquence se met au pas un et attends une instruction de l'horloge interne.

3 Design préliminaire

3.1 Différents approches de résolution

Le séquenceur de huit pas est basé sur un microcontrôleur de petite taille et de basses utilisations de puissance. Lors du prototypage, il sera branché à un ordinateur par un port USB. Le microcontrôleur choisi pour ce module est microcontrôleur Arduino Uno. Le prototype fait usage de huit potentiomètres pour créer une mélodie. Les tensions de ces potentiomètres seront assignées vers les différentes sorties (principale et individuelles, de TON et de GATE, dix-huit en total. Un séquenceur définira vers quelle sortie et en quel moment ces assignations sont faites. Le programme lit l'entrée de l'horloge (interne ou externe) et fait avancer la séquence.

Deux approches sont expliquées à continuation.

Multiplexage interne : En première instance, les potentiomètres seront branchés à des entrées ADC (convertisseurs analogiques numériques) pour être ensuite assignées à l'interne du programme vers les sorties. Étant donné le nombre limité d'entrées du microcontrôleur (six entrées analogiques), il devient nécessaire l'utilisation d'un multiplexeur huit à un en entrée. On utilise ensuite trois sorties digitales du microcontrôleur pour des bits de sélection du multiplexeur. Ensuite, en sortie on ne compte pas avec dix-huit sorties donc deux multiplexeurs externes seront utilisés également sous contrôle des mêmes bits de sélection qui pour le multiplexeur en entrée.

Multiplexage externe : Les potentiomètres sont branchés à un multiplexeur huit à un, sous contrôle des bits de sélection provenant des sorties digitales du microcontrôleur. Ensuite le signal composé est envoyé vers la sortie TON principale et après vers un deuxième multiplexeur qui est chargé de multiplexer le signal vers les sorties individuelles TON. Il y a une sortie digitale du multiplexeur d'où sort le signal GATE (copie de l'horloge externe ou l'horloge interne) qui est envoyé vers la sortie principale GATE, ce même signal est envoyé vers le troisième démultiplexeur qui envoie le GATE vers les GATES individuelles.

3.2 Étude de praticabilité de la solution retenue

La solution la plus intéressante est celle du multiplexage externe parce que les signaux ne sont pas passés par les convertisseurs, ce qui économise en ressources du processeur. Ces ressources pourraient être dédiées au roulement correct du programme. Un autre aspect intéressant de la deuxième solution est la calibration. Étant donné que le signal composé sortant du premier multiplexeur est envoyé directement vers la sortie principale et le deuxième multiplexeur, il est très convenable d'utiliser un seul diviseur sur le signal composé de tension au lieu d'un diviseur de tension sur chaque sortie. Ceci est prouvé dans la section des résultats.

3.3 Diagramme de flux de données

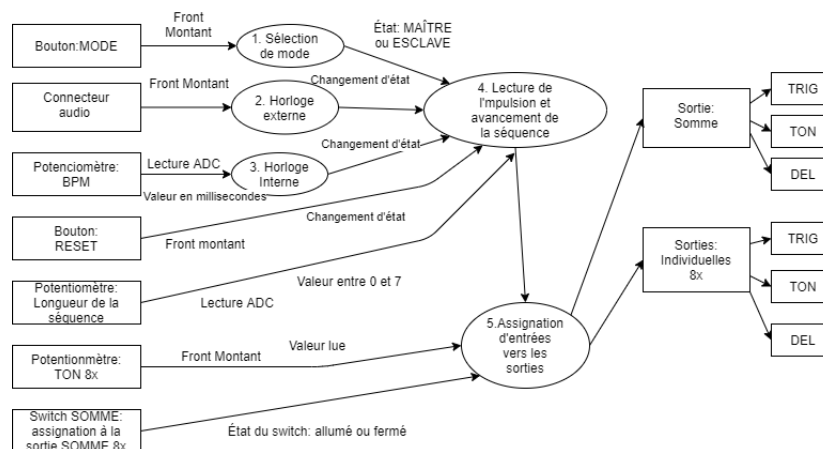


FIGURE 3 – Diagramme de flux de données

3.3.1 Entrées

- Mode : permet de sélectionner le mode d'opération.
- CLOCK EXT : impulsion externe.
- BPM : cadence de l'horloge interne
- RESET : remets le compteur à 1.
- Longueur de séquence.
- TON : note musicale.
- SOMME : assignation à la sortie principale

3.3.2 Sorties

Sortie principale

- TRIG : émet tous les signaux GATE de tous les pas.
- TON : Émet tous les signaux CV de tous les pas.
- DEL : Montre l'activité de TRIG. La DEL s'allume à chaque battement d'horloge.

Sortie individuelle

- TRIG : émet le signal GATE lorsque ce pas est activé par la séquence.
- TON : Émet le signal CV de ce pas lorsque le pas est actif.
- DEL : Montre l'activité de TRIG de ce pas. La DEL s'allume lorsque le pas est actif.

En chaque sortie TON avoir une différence de 2V entre la position minimale et maximale du potentiomètre. En chaque sortie TRIG avoir une impulsion entre 3V et 5V. Que la séquence soit entre 60 et 180 BPM et que la longueur de séquence varie entre 1 et 8 pas lorsque le potentiomètre correspondant est manipulé.

3.4 Description des modules

À continuation on décrit les différents modules fonctionnels.

3.4.1 Sélection de mode

1. En tête :
 - Titre : Sélection de mode (MAÎTRE/ESCLAVE)
 - Version / date : 1.0 / 12 octobre 2020
2. Fonctions réalisées :
 - Lire le front montant du bouton poussoir branché en mode pull-up.
 - Inverser l'état à la sortie.
3. Informations retenues :
 - État interne. 1 bit.
4. Interface :
 - Bouton-poussoir physique en entrée digitale du pin D12 du microcontrôleur.
 - Deux états internes, soit 'HIGH', soit 'LOW' dans une variable d'état.
5. Remarque / Restriction / Procédure :
 - À concevoir en dernière puisque ce module agit comme sélectionneur de mode. S'il y a des problèmes, ils seront résolus plus rapidement.
6. Procédure de tests :
 - Mode MAÎTRE : changer d'état pendant qu'il y a des impulsions à l'entrée du CLOCK.
 - Mode ESCLAVE : changer d'état pendant que l'horloge interne roule.

3.4.2 Horloge externe

1. En tête :
 - Titre : Horloge externe (CLOCK EXT)
 - Version / date : 1.0 / 12 octobre 2020
2. Fonctions réalisées :
 - Lire les fronts montants des signaux.
 - Changer l'état d'une variable dans le programme sur Arduino lorsqu'un front montant est lu.
3. Informations retenues :
 - État de la variable booléenne.
4. Interface :
 - Entrée 1/8" audio en mode pull-down en entrée au pin digital D7 du microcontrôleur. Utilisation d'antirebonds en utilisant la fonction Bounce().
 - La sortie est gérée par le programme en Arduino qui l'envoie par le pin digital 6 du microcontrôleur à 5 V.
5. Remarque / Restriction / Procédure :
 - À concevoir en première en même temps que le module de lecture d'impulsion.

6. Procédure de tests :

- Mode MAÎTRE : Recevoir des fronts en entrée et voir si le lecteur d'impulsions l'ignore.
- ESCLAVE : Recevoir des fronts montants en entrée et voir si la séquence avance.

Bounce() : Fonction Open Source créée par Thomas Fredericks.

<https://github.com/thomasfredericks/Bounce2>

3.4.3 Horloge interne

1. En tête :

- Titre : Horloge interne
- Version / date : 1.0 / 12 octobre 2020

2. Fonctions réalisées :

- Lire la valeur du potentiomètre BPM (millisecondes) et le stocker dans une variable en Arduino.
- Actualiser continuellement la variable.

3. Informations retenues :

- Valeur de temps stocké dans un variable.

4. Interface :

- Potentiomètre en entrée pour modifier le délai de temps.
- Remarque / Restriction / Procédure :
- À concevoir avant le module de lecture d'impulsions pour que ce module puisse l'utiliser déjà.

5. Procédure de tests :

- Mode MAÎTRE : faire bouger le potentiomètre et voir si le module de lecture d'impulsion le reconnaît.
- Mode ESCLAVE : faire bouger le potentiomètre et voir si le module de lecture d'impulsions l'ignore.

3.4.4 Lecture de front montant

1. En tête :

- Titre : Lecture d'impulsion et avancement de la séquence
- Version / date : 1.0 / 17 octobre 2020

2. Fonctions réalisées :

- Lire le MODE d'opération, soit en 'MAÎTRE' soit en 'ESCLAVE'.
- Selon le mode d'opération, attendre un changement d'état et faire avancer la séquence. Si en mode MAÎTRE, attendre un changement d'état du module horloge interne et faire avancer la séquence d'un pas. Si en mode ESCLAVE, attendre un changement d'état du module horloge externe et faire avancer la séquence d'un pas.
- Lire le front montant du bouton 'RESET' et changer la valeur de son état. Si l'état change, réinitialiser la séquence au pas 1.

- Lire la position du potentiomètre de 'Longueur de la Séquence'. Il est possible de choisir une séquence de 1 à 8 pas. Dépendamment de la longueur de séquence voulue, la séquence avance d'un pas ou est réinitialisée au pas 1.
- Indiquer au module d'assignation d'entrées et de sorties que la séquence avance.
- 3. Informations retenues :
 - Le pas courant.
- 4. Interface :
 - Lit l'état en entrée provenant du module de sélection de mode.
 - Lit l'état en entrée provenant du module de l'horloge interne.
 - Lit l'état en entrée provenant du module de l'horloge externe.
 - Lit l'état du bouton 'RESET' en entrée.
 - Lit la position du potentiomètre de 'longueur de la séquence' afin de gérer le pas courant et pas suivant de la séquence.
 - Remarque / Restriction / Procédure :
 - À concevoir au début ensemble avec l'horloge interne. L'un aide à déboguer l'autre.
- 5. Procédure de tests :
 - Mode MAÎTRE : vérifier qu'une impulsion sorte vers SOMME (DEL) lorsqu'on lit un changement d'état de l'horloge interne.
 - Mode ESCLAVE : vérifier qu'une impulsion sorte vers SOMME (DEL) lorsqu'on lit un changement d'état de l'horloge externe.
 - Changer de mode pendant que ce module est allumé.

3.4.5 Transmission

1. En tête :
 - Titre : Assignation d'entrées vers les sorties.
 - Version / date : 1.0 / 17 octobre 2020
2. Fonctions réalisées :
 - Ce module est utilisé comme multiplexeur analogique huit à un et un à huit sous le contrôle du microcontrôleur.
 - Changer d'état (0 à 7) lorsqu'on reçoit une instruction de la part du module d'avancement de séquence.
 - Dépendamment de la position du compteur, on assigne les entrées (POT : potentiomètre, SOMME : assignation à la sortie somme) pour le pas courant aux sorties SOMME (TRIG : sortie d'une l'impulsion reçue en entrée, TON : sortie de l'échelon équivalent de l'entrée POT lorsque le bouton SOMME est enfoncé pour le pas courant, DEL : lumière indiquant qu'une impulsion est sur la sortie TRIG).
 - Les entrées POT sont assignées vers les sorties individuelles du pas courant. TRIG : est la sortie d'une impulsion. TON : est une sortie échelon équivalente à la position de POT. DEL : est une lumière qui s'allume pour indiquer quel est le pas courant.
3. Informations retenues :
 - Aucune.

4. Interface :

- Reçoit une instruction du module d'avancement de séquence et émet via trois pins digitaux du microcontrôleur des bits de sélection pour gérer les multiplexeurs.
- Lit l'état des interrupteurs SOMME de chaque pas pour ensuite les assigner ou pas à la sortie SOMME. Les valeurs lues seront assignées par multiplexage vers SOMME.
- SOMME : La sortie SOMME consiste en trois sous modules (TRIG, TON, DEL). La sortie TRIG sortira une impulsion de 5 V à chaque fois que la séquence avance d'un pas. Cette impulsion pourra être lue par un autre module externe au séquenceur. La sortie TON émettra un échelon du pas courant lorsque le switch SOMME est activé sur ce pas, sinon elle sera mise à zéro en attendant la valeur suivante. DEL indiquera que la sortie TRIG émet pour chaque pas.
- Sorties individuelles : Chaque pas individuelle a trois sorties individuelles (TRIG, TON, DEL). La sortie TRIG sortira une impulsion de 2.2 V lorsque le pas courant est activé. La sortie TON émettra un échelon correspondant à la valeur sélectionné par le potentiomètre individuel. Une DEL pour indiquer que ce pas est actif.
- Remarque / Restriction / Procédure :
- À concevoir une fois que les modules des horloges et de lecture d'impulsion sont en plein fonctionnement.
- Procédure de tests :
- S'assurer que les tensions mesurées après les potentiomètres sont proportionnelles à la sortie SOMME.
- Changer la position des switches et voir si la sortie SOMME se comporte correctement
- En mode MAÎTRE, faire changer la longueur de séquence et BPM et voir si le comportement général du séquenceur est correct.

4 Design détaillé

4.1 Conceptualisation ou modélisation des éléments

Les différents composants sont présentés à continuation.

4.1.1 Entrées et sorties

1. Entrées

Bouton poussoir et résistance utilisé en mode PULL DOWN. Sur la figure 4 on voit un bouton-poussoir enchaîné avec une résistance avant de passer à la masse. Cette résistance s'assure de mettre la valeur du bouton à 0 V lorsqu'il n'est pas enfoncé. Lorsqu'enfoncé, il passe à 5 V. Ces 5 V sont ensuite branchés à une entrée digitale du microcontrôleur, qui les lira comme un 1 logique. C'est le front montant qui est lu par le programme.

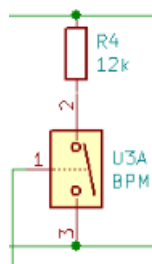


FIGURE 4 – Schéma électrique d'un bouton poussoir et une résistance de 12 kΩ en mode Pull down

Potentiomètre et interrupteur appelé TON et SOMME. Sur la figure 5 se trouve un potentiomètre connecté entre 0 V et 5 V, avec le signal atténué enchaîné avec un interrupteur.

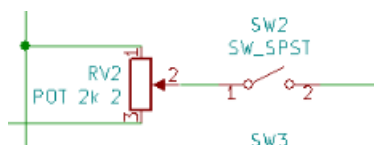


FIGURE 5 – Schéma électrique d'un potentiomètre et un interrupteur

2. Sorties

DEL et connecteur audio pour les sorties TON et GATE de la sortie principale et des sorties individuelles. Sur la figure 6 se trouve une DEL sur avec l'anode branché au signal TRIG provenant du microcontrôleur et la cathode branchée au 0 V. Elle est suivie par le connecteur audio branché directement à la même source.

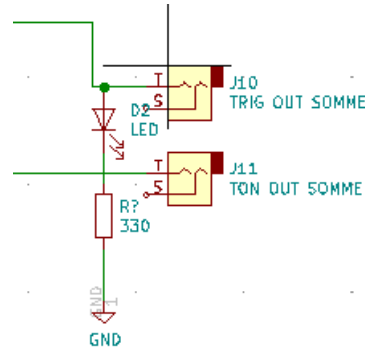


FIGURE 6 – Schéma électrique d'un DEL et un connecteur audio

4.1.2 Le horloge externe

Connecteur audio, une diode et deux transistors pour réguler la tension en entrée et utiliser une entrée digitale du microcontrôleur. Les sources sur la figure 7 sont la masse et V_{cc} à 5 V. Le signal entre par le connecteur audio et passe par une résistance de $100k\Omega$ de façon à réguler la tension d'entrée. Ensuite, le premier transistor bipolaire 2N3904, en configuration interrupteur, varie entre deux états, soit ouverte ou fermé, avec une résistance $R3$ placé pour réguler le courant. Si la tension n'est pas assez grande pour faire actionner le transistor, il reste fermé. Lorsque fermé, le pin 3 du collecteur aura 5 V, parce qu'il n'y aura aucun courant circulant par la résistance $R3$. Lorsque le transistor est actionné, l'interrupteur est ouvert et du courant commence à circuler par la résistance $R3$. Ce courant passe à travers le transistor du pin 3 vers le pin 1 à la masse. Ceci fait diminuer la tension presque à zéro et agit comme une porte logique inverseur. Cette sortie est administrée à un deuxième transistor dans la même configuration d'interrupteur pour renverser le signal.

Particularités. — Tension d'entrée : Pour la plupart de synthétiseurs modulaires les signaux d'horloge sont des signaux carrés périodiques d'entre 3 et 10 V. En fait, il est possible d'utiliser presque n'importe quel signal entré parce qu'on s'intéresse à lire le front montant du signal. Après le connecteur audio, lorsque la tension est positive, le signal trouve en chemin un diviseur de tension.

$$V_{out} = \frac{1M * V_{in}}{1.1M} = 0.909 * V_{in}$$

Le transistor a besoin de 0.7 V pour être actionné, donc :

$$V_{in} \geq \frac{0.7}{0.909}$$

$$V_{in} \geq 0.77V$$

Donc il n'est nécessaire que 0.77 V en entrée pour activer le transistor.

- Diode : Si la tension en entrée diminue à moins de -0.7 V le signal passe à la masse donc le transistor n'est pas affecté.

- Transistor 1 : En mode de saturation lorsque la tension au pin 2 est supérieure à la tension au pin 1 et au pin 3. On s'assure de n'avoir pas avoir un courant supérieur à 0.0005 mA à travers la résistance $R3$. La tension au pin 3 est de 5 V lorsque V_{in} est inférieure à 0.77 V et tombe à presque 0 V lorsque V_{in} est supérieure à 0.77 V. On voit que le fonctionnement suit une logique du type inverseur.
- Transistor 2 : Son ajout permet d'inverser l'effet précédent de logique du transistor 1 pour avoir de nouveau le front montant de V_{in} et du V_{out} du transistor 2 en même temps.

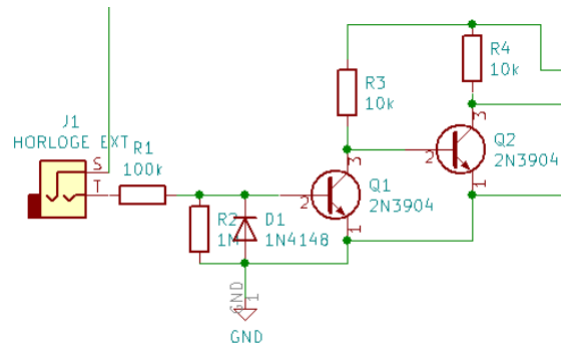


FIGURE 7 – Schéma électrique d'un connecteur audio et deux transistors en configuration d'interrupteur

4.1.3 Multiplexage

Les trois chips de multiplexage utilisés sont les 74HC4051. Ils fonctionnent comme multiplexeurs ou dé-multiplexeurs 8 à 1. Sur la figure 8 on voit la représentation générique du multiplexeur dans le programme Kicad®.

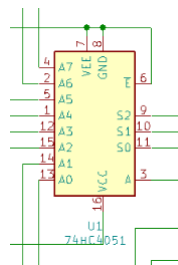
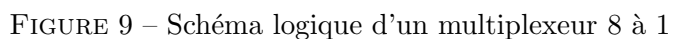


FIGURE 8 – Schéma électrique d'un multiplexeur 8 à 1

Sur la figure 10 à gauche on voit que le multiplexeur a huit entrées I avec trois entrées de sélection A et une sortie Z . Sur la partie droite, on voit la logique à l'intérieur du chip.

Les bits de sélection choisissent quelle entrée est envoyé à la sortie, donc il y n'y a qu'une entrée sélectionnée à la fois.



4.1.4 Microcôntroleur

Pinout diagram of the Arduino Uno R2 showing the connections for the DS18B1C temperature sensor. The sensor's VCC pin is connected to the 5V pin (pin 5) of the Arduino. The sensor's GND pin is connected to the GND pin (pin 1). The sensor's data pin is connected to digital pin 2 of the Arduino. The Arduino's 5V pin is also connected to the 5V pin of the USB Type-A connector. The Arduino's GND pin is also connected to the GND pin of the USB Type-A connector.

19/25

4.2 Réalisation du prototype

4.2.1 Électrique

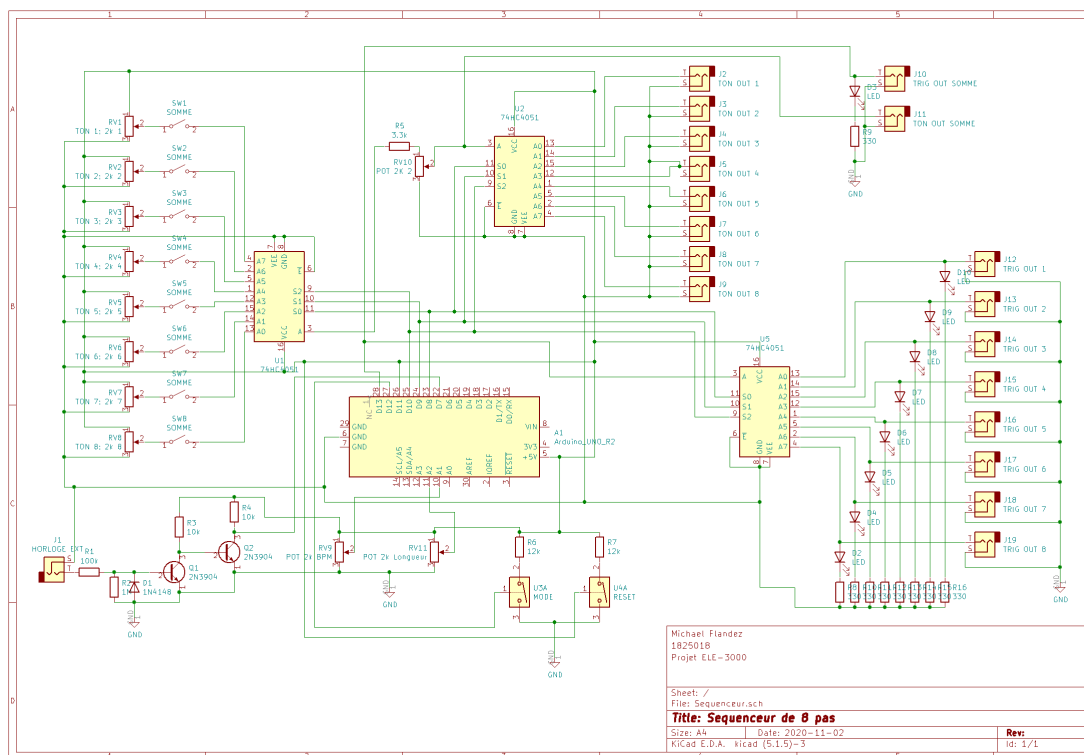


FIGURE 11 – Schéma électrique composée sur Kicad©

TON. La figure 11 présente la configuration finale de l'ensemble des composants. À gauche se trouve huit potentiomètres de 2.2 kΩ branchés à des interrupteurs qui suivent vers les huit entrées du premier multiplexeur. Ce multiplexeur utilise ses trois bits de sélection $S0$, $S1$ et $S2$ provenant des pins digitales $D8$, $D9$ et $D10$ du microcontrôleur. Ce sont les mêmes bit de sélection qui sont utilisés par les deux autres multiplexeurs. Chaque potentiomètre représente un pas de la séquence (mélodie), donc la sortie A du multiplexeur est la mélodie. Cette mélodie est acheminée vers l'entrée A du deuxième multiplexeur et vers la sortie principale $SOMME$. Avant d'y arriver, ce signal passe par une résistance de 3.3 kΩ en série avec un autre potentiomètre de 2.2 kΩ. Ce potentiomètre permet d'ajuster le signal de mélodie pour avoir justement une différence de 2 V entre la TON minimum et TON maximum possible. Une fois ajusté, le signal de mélodie arrive au multiplexeur et est redistribué par le bit de sélection vers huit sorties individuelles TON, un pour chaque potentiomètre de départ. Également, la

copie du signal de mélodie principale arrive à la sortie SOMME.

En faisant comme ceci, On s'est assuré que tant la sortie principale comme les sorties individuelles ont la même différence de 2 V.

TRIG. Sur la même figure 11, on voit que le troisième multiplexeur (en bas à droit) prend en entrée *A* la sortie du pin *D13* du microcontrôleur. Le programme envoie une impulsion courte vers la sortie *D13*, laquelle s'achemine vers la sortie principale TRIG SOMME et vers le troisième multiplexeur. Le de-multiplexeur distribue alors les impulsions vers les huit sorties individuelles TRIG grâce aux mêmes bits de sélection *D8*, *D9* et *D10*. Chacune de ces sorties a une DEL et une résistance attachées pour indiquer à l'utilisateur le pas courant de la séquence.

Sur la partie inférieure de l'image 11, en dessous du microcontrôleur se trouvent cinq entrées.

Horloge externe. L'horloge externe active la configuration de transistors en mode interrupteur donnant des tensions de 0 V ou 5 V. Cette tension est branchée à une entrée digitale du microcontrôleur (pin *D7*) est utilisé dans le programme comme variable booléenne. Ceci se passe lorsqu'en mode ESCLAVE.

BPM et longueur de la séquence. Ces deux potentiomètres sont branchés entre 0 V et 5 V et leurs signaux atténués rentrent dans les pins *A2* et *A1* respectivement. Là ils sont convertis en valeurs numériques entre 0 et 1023 pour être utilisés par le programme.

MODE et RESET. Ces deux derniers boutons poussoirs sont connectés aux pins *D12* et *D11* respectivement. Le programme lit les fronts montants de ces signaux.

4.2.2 Logiciel

Le programme mesure continuellement les entrées mentionnées dans la section 4.2.1, gère la séquence et les bits de sélection des multiplexeurs et envoie des impulsions. La figure 12 contient les instructions générales programmées.

Sur la figure 12 on voit la description d'une librairie *open source* qui a été trouvé sur le site <https://github.com/> et qui est chargée de traiter l'antirrebonds lors de l'appui sur les boutons poussoir. Ceci empêche le programme de traiter plus d'une fois la même action sur le bouton. Ensuite, configure les entrées et sorties utilisées, analogiques et digitales comme vues sur l'image 11. La partie MAIN présente le corps du programme. Ça débute avec la sélection de la longueur de la section. Ensuite, le programme roulera dépendamment de MODE de sélection (MAÎTRE ou ESCLAVE) choisi avec le bouton de sélection de MODE. Dans les cas de MODE MAÎTRE, le programme attend pour l'impulsion interne qui est contrôlée par BPM, et qui fera avancer la séquence d'un pas. À ce moment-là, le programme fait changer les bits de sélection vers le pas suivant dans la séquence et assigne les tensions des multiplexeurs externes vers les sorties correspondants. Le programme fait envoyer une impulsion vers le multiplexeur d'impulsions (TGIG) et de là vers les sorties SOMME TRIG et TRIG individuelle. Ceci finit avec cette partie du MODE MAÎTRE.

Finalement, lorsqu'en mode ESCLAVE, si le bouton de sélection de MODE est mis sur l'option ESCLAVE, le programme attend à recevoir l'impulsion d'un module externe. Cette impulsion fait ensuite avance la séquence de la même manière comme expliquée antérieurement.

```
// librairie externe, https://github.com/thomasfredericks/Bounce2
// #include <Bounce2.h>, fonction que lit si un bouton est pressonné
// avec antirebonds

// configuration des pins
// bouton MODE, RESET
// connecteur CLOCK EXT
// pot BPM, LONGUEUR
// sorties TRIG et bits de Sélection de multiplexeur S0, S1, S2

// MAIN
// sélection de LONGUEUR de la séquence
// si MODE MAÎTRE
// faire avancer la séquence d'accord à BPM et à la LONGUEUR de séquence
// sélection des bits de multiplexage
// envoyer une impulsion vers la sortie TRIG

// si MODE ESCLAVE
// lire le front montant de l'horloge externe
// faire avances la séquenced'accord à la LONGUEUR de séquence
// sélection des bit de multiplxage
// envoyer une impulsion vers la sortie TRIG
```

FIGURE 12 – Allure général du programme

4.3 Réalisation préliminaire et itérations du design

La conception de ce module a amené à la bonne exécution des spécifications fonctionnelles et du problème de départ : la création des mélodies facilement et la synchronisation de ce module dans un ensemble.

1. Utilisation des entrées analogiques du microcontrôleur pour traiter tous les potentiomètres, boutons et interrupteurs. Les boutons seraient intégrés en forme *pulldown* et leurs sorties branchées directement aux entrées, utilisant dans le programme une fonction antirrebonds. L'entrée audio est branchée avec la même configuration qu'avec les boutons. Tout le multiplexage est géré à l'intérieur du programme.
2. L'étude de praticabilité a expliqué la préférence du système hybride. Là on adapte le design pour utiliser des multiplexeurs externes au microcontrôleur sous le contrôle de trois bits de sélection du multiplexeur.
3. Étant donnée la multiplicité de modules externes qu'on aimerait brancher à l'entrée jack du microcontrôleur ayant de très différentes tensions (0 V - 12 V), on implémente une configuration de switch ç l'aide d'un transistor BJT. Ceci ajoute un étage de sécurité pour éviter d'endommager l'entrée de microntronroleur. Une résistance et un potentiomètre ont été rajoutés entre le microcontrôleur et le multiplexeur du TON de façon à ajuster cette sortie pour avoir l'écarte de tensions voulues.

5 Validation

Une fois que le prototype a été conçu, c'était le moment pour le tester au laboratoire. On s'est équipé d'un oscilloscope pour mesurer des tensions et des durées de signaux.

5.1 Procédures de test et résultats

Comme mentionné précédemment, la description de ce module est un séquenceur créateur de mélodies de deux octaves, dont une octave par volt de tension. Ceci amène à vouloir avoir 2 V d'écart entre la tension minimale et maximale possible atteignable par les potentiomètres. Il est important aussi de mesurer la tension et la longueur de l'impulsion à la sortie SOMME TRIG, parce qu'on veut bien que ce signal soit détectable par d'autres modules.

Test 1 On fait rouler le séquenceur et on positionne deux potentiomètres de façon à avoir une tension minimale sur un pas et une autre tension maximale sur les pas suivants.

Test 2 On mesure la sortie du connecteur SOMME TRIG lorsque le séquenceur roule.

5.2 Analyse et validation des résultats

Résultats du test 1 Sur l'image 13 on voit sur l'axe des coordonnées trois pas successives dans le temps. Le premier à gauche est placé à -0.910 V sur l'échelle des coordonnées, le deuxième pas à 1.090 V. Il existe une troisième qui suit aussi à -0.910 V. La différence entre le premier et deuxième pas est indiquée par ΔY et est égale à 2 V.

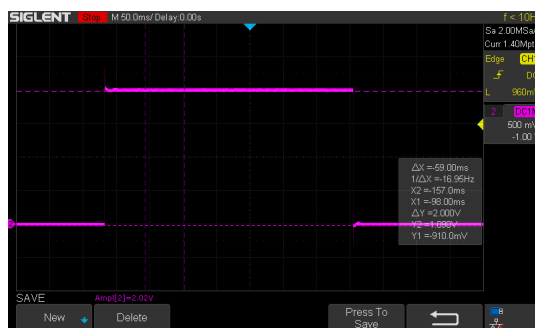


FIGURE 13 – Écart maximal entre les valeurs de tension trouvées à la sortie SOMME

Résultats du test 1 L'image 14 présente une capture d'une impulsion à la sortie SOMME TRIG. On voit bien que ΔY est de 3.240 V. D'un autre côté, ΔX indique la durée de cette impulsion qui est de 13.60 ms.

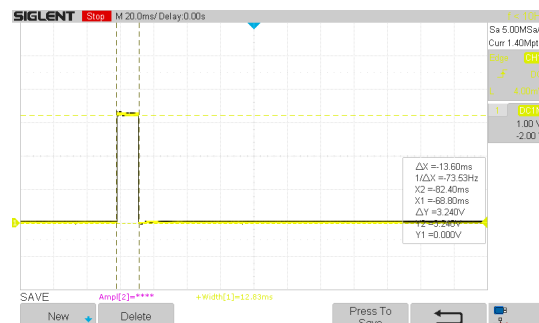


FIGURE 14 – Valeurs mesurées des impulsions à la sortie SOMME TRIG

5.3 Analyse des résultats en fonction des hypothèses initiales

Le premier test donne des résultats précis de 2 V, ceci après ajustement d'un potentiomètre. Ces 2 V se traduisent en 2 octaves musicales. Si bien sur l'image 13 on voit que les tensions sur l'axe des coordonnées ne sont pas de chiffres ronds, on s'intéresse seulement à l'écart. Normalement, les modules auxquelles l'utilisateur branchera ces signaux auront un potentiomètre pour les ajuster. D'un autre côté, si jamais on sera intéressé à étendre à 3 octaves ou plus, ce requerra juste un ajustement du potentiomètre en question.

Le deuxième test montre une sortie de plus de 3 V et de plus de 10 ms, ce qui déclenchera un front montant assez important pour que d'autres modules puissent le lire.

6 Conclusion

Les résultats démontrent la validité de design de séquenceur de huit pas. Le séquenceur fonctionne parfaitement soit en mode MAÎTRE, actionné par l'horloge interne, soit en mode ESCLAVE, actionné par l'horloge externe. À l'appui du bouton de sélection de mode, la séquence continue son travail en s'adaptant à la cadence de nouveau mode sans sauter des pas ni se bloquer.

Le module change la vitesse de la séquence lorsque l'horloge externe change de vitesse (mode ESCLAVE), et aussi lorsqu'on change la position du potentiomètre BPM (mode MAÎTRE).

Dans les deux modes, la longueur de séquence peut être modifiée grâce au potentiomètre *longueur de séquence*, choisissant entre 1 et huit pas. La mélodie principale est créée par le positionnement de huit potentiomètres TON, facilement ajustables qui ont un switch d'assignation aux sorties principales et individuelles TON. Il y a aussi des sorties GATE d'où sortent des impulsions. Ce module séquenceur s'intègre dans un ensemble comme horloge principale ou comme sous-module d'horloge.

6.1 Évaluation des limites et des incertitudes

Comme tel, le séquenceur est capable de créer des mélodies de deux octaves. Si jamais l'intérêt arrive, il est possible de recalibrer la sortie de TON à qu'elle soit d'une autre valeur, allant jusqu'à

5 Volts, ou 5 octaves.

L'incertitude comme vue aux tests de validation n'est nul, ce qui veut dire que l'incertitude ou erreur de la sortie TON dépend de l'erreur propre de l'oscilloscope.

6.2 Proposition des nouvelles connaissances et habiletés acquises durant le projet

Si bien le projet de départ a été réussi, on s'inspire pour déjà pour le futur. À continuation on nomme quelques fonctionnalités à rajouter.

- Prolonger la séquence à un maximum de 16 pas. Rajouter une deuxième séquence de 16 pas en parallèle.
- Chaque séquence aura des horloges internes indépendants avec la possibilité de les synchroniser, avec des longueurs indépendants. Ceci avec le but de créer de poli rythmique.
- Rajouter un adaptateur de puissance indépendant pour que le séquenceur fonctionne sans être branché à l'ordinateur.
- Inclure des tampons aux sorties pour les rendent plus robustes.

7 Apprentissage continu

En termes de connaissances apprises, l'environnement de développement Arduino a été découvert. Arduino compte avec une ligne de cartes de développement à très bas prix, idéale pour de petits projets comme ce séquenceur de huit pas. Du côté logiciel on voit que c'est très facile à programmer grâce aux connaissances de langage C++ et grâce aux à la grande quantité de ressources disponibles en ligne. La communauté Arduino est active et leur site internet est très bien fait. Malheureusement le côté matériel n'a pas été actualisé d'autant que d'autres cartes programmables, par exemple on sait que les cartes Arduino ne comptent pas avec des *buffers DMA* qui permettent de faire de l'échantillonnage cyclique et les convertisseurs ne sont pas plus grands que 8 bits, ce qui empêche le travail plus exhaustif de *DSP*, (*traitement de signaux numérique*).