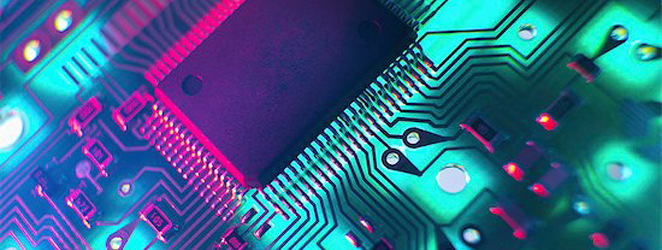


**ELECTRONICIEN/NE CFC  
 ELEKTRONIKER/IN-EFZ**



Carte Electronique Didactique

1.0.2

**Module E213**

Métier, Beruf

Pittet Loïc

2024/2025

EMF - Ecole des Metiers de Fribourg

SWITZERLAND

Table des matières

[1 Information 2](#_Toc199225248)

[1.1 Objectifs du projet 2](#_Toc199225249)

[2 Planification 3](#_Toc199225250)

[2.1 Calendrier prévisionnelle des tâches à réaliser 3](#_Toc199225251)

[2.2 Calendrier réel des tâches réalisées 3](#_Toc199225252)

[3 Décision 4](#_Toc199225253)

[4 Réalisation 5](#_Toc199225254)

[4.1 Schéma bloc de l'électronique à développer 5](#_Toc199225255)

[7](#_Toc199225256)

[4.2 Schématique, Dimensionnement & simulation 9](#_Toc199225257)

[4.3 Rédaction du protocole de mise en service du prototype 15](#_Toc199225258)

[4.4 Liste de matériel 17](#_Toc199225259)

[4.5 Layout 17](#_Toc199225260)

[4.6 Firmware 20](#_Toc199225261)

[4.7 Phase de production en série 21](#_Toc199225262)

[4.8 Liste des outils utilisés et leur version 21](#_Toc199225263)

[5 Contrôle 22](#_Toc199225264)

[5.1 Validation ERC 22](#_Toc199225265)

[5.2 Validation DRC 22](#_Toc199225266)

[5.3 Validation Eurocircuits 22](#_Toc199225267)

[5.4 Validation par la mise en service du prototype 22](#_Toc199225268)

[6 Evaluation 23](#_Toc199225269)

[6.1 Etat du projet 23](#_Toc199225270)

[6.2 Liste des modifications pour la prochaine version 23](#_Toc199225271)

[6.3 Liste des délivrables 23](#_Toc199225272)

[6.4 Amélioration possible 24](#_Toc199225273)

[6.5 Conclusion et avis personnel 24](#_Toc199225274)

# Information

## Objectifs du projet

Le but du projet est de réalisé un PCB qui met en valeur l’effet de l’hystérèse sur un comparateur en prenant en compte une tension d’entrée variable grâce à un capteur analogique de lumière pour pouvoir voir si l’on doit allumer une LED (éclairage public) s’il y’a trop peu de lumière. Un bruit sera généré avec un PIC.

### Résumé du cahier des charges

Une image contenant texte, capture d’écran, Police, nombre

Le contenu généré par l’IA peut être incorrect.

# Planification

## Calendrier prévisionnelle des tâches à réaliser

Une image contenant texte, mots croisés, capture d’écran, diagramme

Le contenu généré par l’IA peut être incorrect.

## Calendrier réel des tâches réalisées

# Décision

|  |  |  |  |
| --- | --- | --- | --- |
| Date | Qui | Quoi | Pourquoi |
| 13.05.2025 | Pittet | Changer de capteur de lumière.   * SFH-3310\_EN | Capteur actuel plus disponible sur le marché avant 18 semaines… Les LDRs sont fabriqué avec du Cadmium qui est très polluant. |
| 14.05.2025 | Pittet | Choix du moyen de faire du Bruit avec le PIC. 🡪 Bruit numérique par pseudo-aléatoire (LSFR) | Méthode la plus simple à mettre en place sans rajouter trop de composants. |
| 14.05.2025 | Pittet, Berset | Choix du PIC.   * PIC16F15214 | Pic déjà connu dans l’entreprise et il n’a pas trop d’I/O. On a besoin que d’une seule sortie. |
| 14.05.2025 | Berset, Pittet | Mettre un deuxième capteur de lumière | Avoir une tension de référence fixe qui ne sera jamais plus grande que celle que l’on mesure. (Moins il y’a de lumière, plus la tension baisse) |
| 15.05.2025 | Berset, Pittet | Ajout d’un filtre passe-bas | Passer d’un signal carré en sortie du PIC en un signal plus rond après le filtre. Cela rend le scintillement plus réaliste. |
| 16.05.2025 | Berset, Pittet | Inverser les tensions à soustraire. | Avoir une tension positive à la sortie de l’additionneur **inverseur** pour avoir une tension positive en entrée du comparateur. |
| 16.05.2025 | Pittet | Changement résistance en série du capteur de lumière  6k8 🡪 10k | Réajustement de la résistance pour le nouveau capteur de lumière. |

# Réalisation

## Schéma bloc de l'électronique à développer

Le schéma bloc nous sert à mieux comprendre l’idée de base au début d’un projet. On peut y voir directement quels sont les groupes de fonctions qui sont importants dans le fonctionnement du circuit.

Une image contenant diagramme, texte, Dessin technique, Plan

Le contenu généré par l’IA peut être incorrect.

Le but de ce circuit est d’avoir une tension utilisable variable pour pouvoir l’utilisé comme tension comparatif à 2,5V. Pour ce faire, nous pouvons voir le conditionneur qui précède cette tension. Nous allons générer un petit bruit d’environ 20mV.

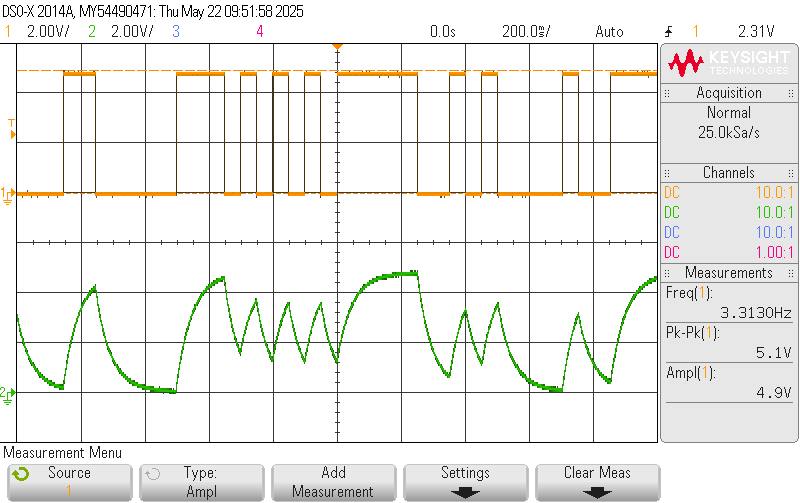
La première partie est constitué de deux capteurs de lumières (phototransistor). Un premier capteur est là pour avoir une tension de référence qui ne changera jamais (à part si un nuage vient perturber la lumière ambiante.) Cette tension sera soustraite de l’autre capteur pour que cette tension ne soit jamais positive (à voir après pourquoi). A savoir que si la lumière capté baisse, le courant baisse et en conséquence la tension aussi. Donc si l’on cache ce second capteur, la tension de référence sera de toute façon plus grande. C’est pourquoi que la tension de sortie sera négative ou nulle.

Nous allons ensuite passer cette tension de sortie du soustracteur dans un **additionneur inverseur**. C’est pourquoi il nous fallait une tension négative après le soustracteur pour pouvoir retrouver une tension positive à la sortie de l’additionneur. A cette tension, nous allons ajouter une composante continue de **-2,5V** pour pouvoir travailler autour de la tension de seuil qui est de 2.5V au lieu de travailler sur une tension pratiquement nulle.

Et pour terminer, un bruit d’environ 20mV est ajouté au signal pour pouvoir voir l’effet du bruit sur un comparateur avec et sans hystérèse. Ce bruit est généré grâce à la méthode de Bruit Numérique par Pseudo-Aléatoire (LFSR). J’utilise un PIC dans lequel j’ai codé un registre à décalage. Le bit 0,3 et 5 sont récupéré et sont passé dans des portes XOR pour avoir un nouveau bit « aléatoire » sur le 8ème bit. Ce qui nous permet d’avoir un bruit **pseudo**-aléatoire. Cependant, nous ne pouvons pas traiter cette tension tel quel. C’est pourquoi nous allons créer un filtre passe-bas pour avoir une tension un peu plus **analogique** et non numérique.

Une image contenant texte, capture d’écran, ligne, Police

Le contenu généré par l’IA peut être incorrect.



Le signal orange montre la sortie sur PIC. Nous pouvons observer sur le signal vert la tension filtrée en passe-bas.

Après que ces trois tensions sont additionnées, nous pouvons la traiter dans une entrée de notre comparateur. Nous allons comparer cette tension d’entrée variable à une tension de seuil fixe de 2,5V. Voilà pourquoi nous avions ajouté un offset de 2.5V auparavant.

### Schéma bloc de l'électronique à développer 2

## Une image contenant diagramme, texte, ligne, Plan Le contenu généré par l’IA peut être incorrect.

Cette proposition a été jugé trop compliqué à gérer la partie PWM.

Le but était de reset le programme avec un bouton lorsque la lumière ambiante change. Mais dans un atelier, il y’a du mouvement, des zones d’ombres et s’en parler des perturbations externes tels que les nuages, un coucher de soleil, une lampe…

Cette version avait été imaginer pour que le PIC s’occupe de tout en interne.

### Schéma bloc de l'électronique à développer 3

Une image contenant diagramme, texte, Plan, Dessin technique

Le contenu généré par l’IA peut être incorrect.

Solution abordable mais pas très pratique puisqu’à chaque fois qu’un évènement vient perturber la lumière, il faut réajuster la tension (de référence) avec un trimmer (donc manuellement) par exemple la tension de la lumière actuelle. Par exemple lorsqu’un nuage passe devant le soleil, cela va biaiser les mesures. Pareil si quelqu’un éteint la lumière…

Cette solution est un entre deux entre la version PWM et la version choisie.

## Schématique, Dimensionnement & simulation

### Schématique globale

Une image contenant texte, diagramme, Parallèle, Plan

Le contenu généré par l’IA peut être incorrect.

#### Schématique Soustracteur

Une image contenant texte, diagramme, ligne, nombre

Le contenu généré par l’IA peut être incorrect.

Voici la première partie de notre schéma. On peut observer que l’on soustrait une tension de référence « fixe » à une tension mesurable (Umes).

Q1 et Q2 sont deux phototransistors qui mesurent l’intensité lumineuse.

Q1 est fixe, c’est-à-dire que l’on ne va jamais modifier la lumière qu’il capte.

Q2 est variable, c’est-à-dire que si l’on vient cacher le capteur de la lumière, le courant va baisser et donc la tension aussi.

On n’a pas besoin d’amplifier le signal donc j’ai dimensionné un gain de 1.

La tension de sortie du soustracteur sera toujours négative ou égal à 0V. Comme on vient que cacher Q2 et que l’on ne va jamais venir ajouter de la lumière dessus, alors la tension de sortie ne sera jamais positive. (Voir plus bas avec simulation TINA)

#### Schématique Additionneur

Une image contenant texte, diagramme, ligne, Parallèle

Le contenu généré par l’IA peut être incorrect.

Voici la deuxième partie du notre schéma. On peut observer que l’on additionne trois tensions. Nous avons la tension de sortie du soustracteur sur R10, une composante continue de -2,5V sur R7 et une dernière tension de 20mV qui est une simulation de bruit.

Le montage est un additionneur **inverseur.** Voilà pourquoi on voulait une tension négative en sortie du soustracteur. On peut avoir une tension minimale de -2,5V et maximale de 0V sur R10 avec un gain de 1.

Pour générer un offset de 2,5V pour pouvoir travailler autour de la tension de seuil du comparateur qui est de 2.5V, on va additionner une composante continue de -2.5V avec un gain de -0.5V/V (-R6/R7).

La dernière tension additionnée est une tension d’environ 1.6V sur R14 avec un gain de 12mV/V pour avoir une tension de bruit de 20mV (-R6/R14). Cette tension est générée par un PIC (détaillé plus bas).

Nous allons retrouver une tension positive ou égal à 0V en sortie de l’additionneur.

#### Schématique Générateur de bruit

Une image contenant texte, diagramme, ligne, capture d’écran

Le contenu généré par l’IA peut être incorrect.

Voici une partie importante de la schématique. Puisque c’est grâce à ce bruit que toute l’application ait du sens.

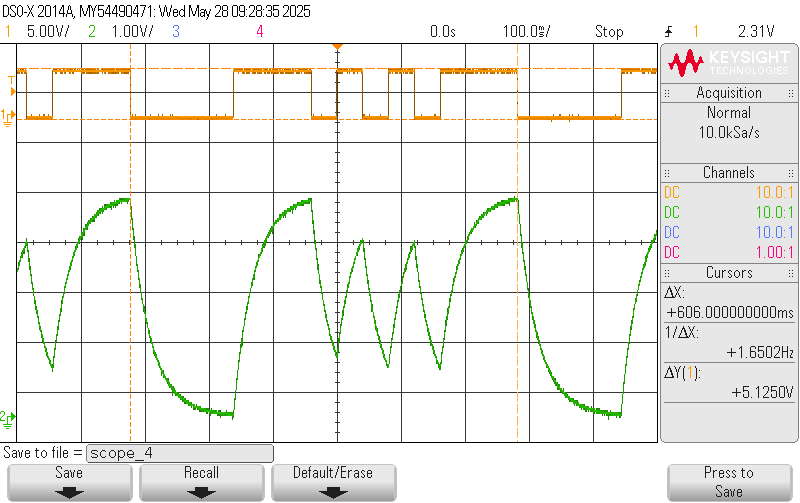
Le PIC génère une tension numérique « aléatoire » grâce à un registre à décalage de 8 bits. On ressort le bit 0,3 et 5 que l’on passe dans des portes XOR et l’on remet ce bit en entrée du registre à décalage. Avec ce procédé, on obtient une tension numérique pseudo-aléatoire. Ce qui nous permet de dire que cela est du bruit après avoir filtré cette tension avec C7 pour la rendre analogique.

Le diviseur de tension nous permet de réduire la tension pour éviter d’avoir une résistance R14 (additionneur) trop élevé. On a un peu près une tension de 1.6V sur R14 (5V x (R19/R19+R16)).

Une image contenant texte, capture d’écran, ligne, Police

Le contenu généré par l’IA peut être incorrect.

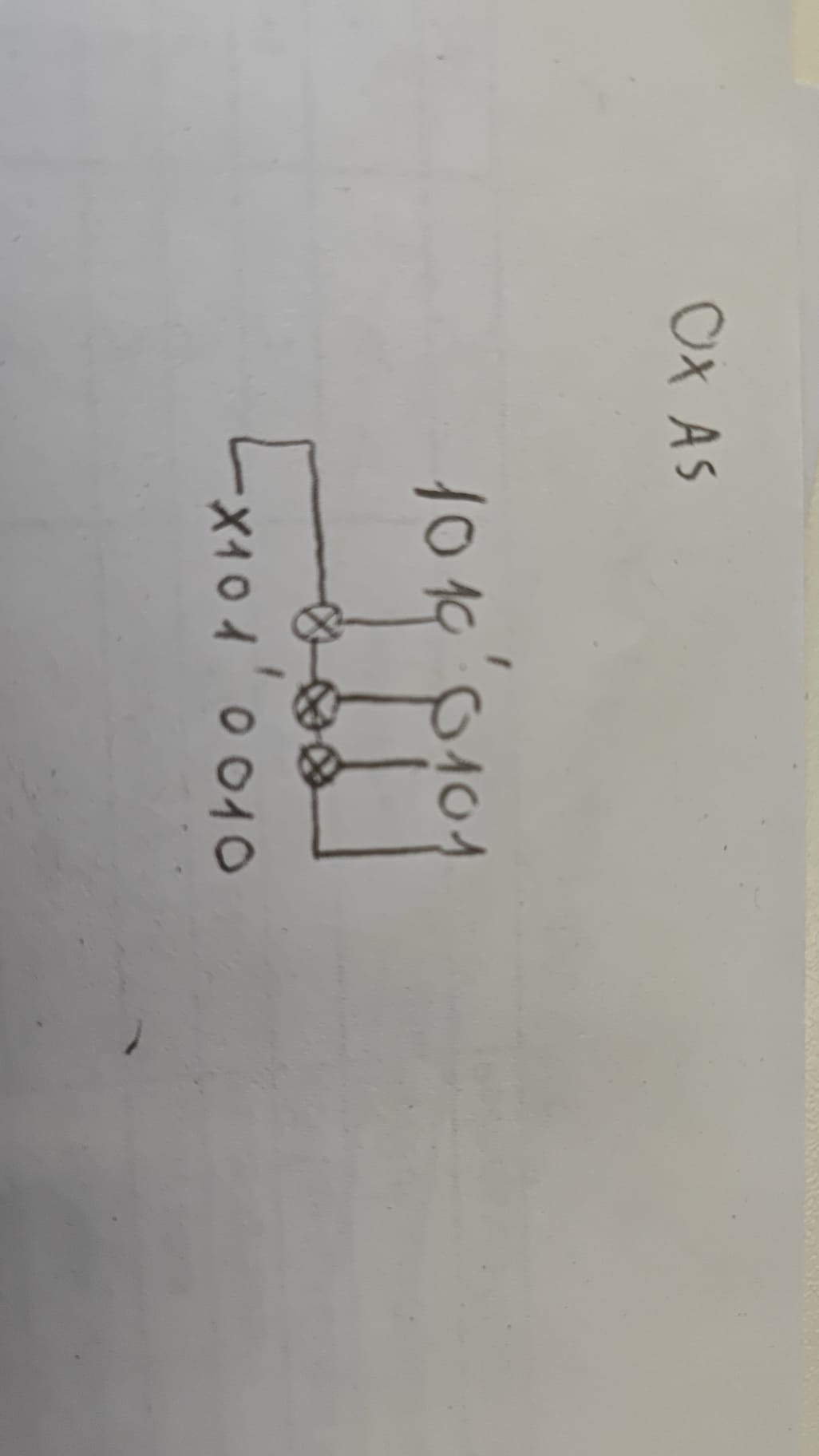
Maintenant, pour pouvoir créer un bruit le plus aléatoire possible, il faut trouver la meilleure trame d’entrée pour que la boucle soit la plus grande possible. Si l’on prend par exemple la trame que Wikipédia nous propose, nous avons :

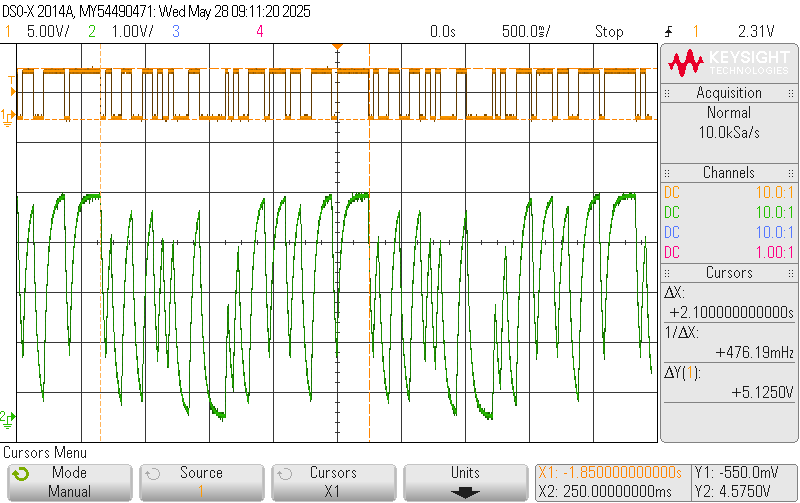


On observe que la boucle est bel et bien le même. Le code fonctionne donc bien. Les colonnes surlignées en jaune sont les bits qui sont passé dans des XOR et ou le résultat est renvoyé au début de la tram suivante.

On voit aussi qu’il ne faut que 16 lignes de code pour devoir faire la boucle soit 606ms environ.

Cependant, j’ai demandé à ChatGPT s’il existait une meilleure trame de bit pour avoir pour finir une plus grande boucle et donc un signal plus aléatoire. Et en effet, il existe selon l’IA une meilleure trame et surtout un meilleur moyen de traiter cette trame. Voici la proposition de ChatGPT :

Tram de bit = 0xA5

On se retrouve avec une porte logique XOR en plus, ce qui ajoute encore plus d’aléatoire. On voit aussi que l’on ne prend pas en compte les mêmes Bits. Ici on travaille avec le bit 0,2,3 et 4.

Voyons maintenant ce que cela donne avec le signal :

On observe que la période dure maintenant environ 2.1s au lieu de 606ms. Donc on a une boucle trois fois plus grande que la méthode proposée par Wikipédia. On remarque aussi que le signal est un peu plus aléatoire dans l’amplitude que l’autre version.

C’est selon ChatGPT la meilleure trame qui produit une séquence maximale (255 valeurs). C’est aussi une trame classique lorsque l’on emploie la méthode de pseudo-aléatoire.

Il faut par contre ne pas se tromper sur le décalage de bit. Si l’on décale les bits par la gauche ou par la droite, la boucle ne sera pas du tout la même. Pour la méthode de Wikipédia, c’est un décalage à droite que l’on fait.

Cependant, pour la version de ChatGPT, c’est un décalage vers la gauche qu’il faut faire. Je lui ai demandé de me sortir des tableaux de valeurs après 10 décalages :

Décalage vers la gauche :

Une image contenant texte, capture d’écran, nombre, Police

Le contenu généré par l’IA peut être incorrect.

Décalage vers la droite :

Une image contenant texte, capture d’écran, nombre, Police

Le contenu généré par l’IA peut être incorrect.

On voit que lorsque l’on fait le décalage vers la droite, la boucle ne dure même pas 10 coups (6). Alors qu’avec le décalage par la gauche, le résultat est nettement meilleur.

#### Schématique Comparateur à Hystérèse

Une image contenant texte, diagramme, ligne, Tracé

Le contenu généré par l’IA peut être incorrect.

Cette dernière partie est la partie principale du schéma. Après avoir conditionné le signal avant, on peut enfin venir utiliser cette tension.

On va venir comparer la tension variable conditionné auparavant à une tension fixe de 2.5V (diviseur de tension entre R2 et R3 qui sont identiques).

On a le choix de connecté ou non la résistance R4 (hystérèse) pour pouvoir voir l’effet de celle-ci.

Lorsque la tension d’entrée est plus grande que 2.5V, alors le comparateur va sortir un 1 logique et ainsi 5V en sortie.

Lorsque la tension d’entrée est plus petite que 2.5V, alors le comparateur va sortir un 0 logique et donc 0V.

Pour pouvoir voir l’effet du bruit et de l’hystérèse, on a branché une LED en sortie du comparateur pour pouvoir bien voire quand la LED scintille ou non.

#### Schématique Capteur Phototransistor

Une image contenant texte, ligne, diagramme, Police

Le contenu généré par l’IA peut être incorrect.Je vais apporter quelques informations au sujet du fonctionnement d’un phototransistor :

Pour commencer, parlons de l’unité lumineuse. Ici, on va parler d’éclairement lumineux avec comme unité du système international le **lux.** Par définition, un lux représente l’éclairement d’une surface qui reçoit, d’une manière uniformément répartie, un flux lumineux d’un lumen par mètre carré. (1lx = 1 lm/m2) On parle aussi de l’éclairement énergétique qui peut se définir par des W/m2.

L’œil humain peut s’accommoder à des niveaux d’éclairement très variables, de 130'000 lux (une journée ensoleillée d’été) jusqu’à 1 lux (une nuit de pleine lune). Lors d’une journée avec un ciel nuageux, environ 20'000 lux sont présent. Un ciel couvert en hiver représente 3'500 lux. Une salle de classe équivaut un peu-près à 1000 lux.

Cependant, certains niveaux minimaux sont requis. Par exemple pour pouvoir se déplacer, un minimum de 5 lux est requis. Pour pouvoir faire de la lecture ou de l’écriture, il faudrait au moins 150 lux.

Une image contenant texte, ligne, diagramme, Tracé

Le contenu généré par l’IA peut être incorrect.

On peut observer que le courant ICE augmente selon la luminosité présente sur le capteur. Si l’on branche une résistance en série à ce capteur, nous pouvons donc faire une chute de tension variable en jouant avec la luminosité.

Ainsi, on peut affirmer qu’avec ce phototransistor, le courant augmente si la luminosité augmente. C’est-à-dire que la tension de sortie va aussi augmenter.

Une image contenant diagramme, texte, ligne, Tracé

Le contenu généré par l’IA peut être incorrect.

Avec ce graphique, on peut observer l’angle dans lequel le capteur va le mieux absorber la lumière. On voit que ce n’est pas une lumière verticale qui est la plus adaptée, mais une lumière avec un angle de +/- 70°.

Il faut quand même bien interpréter cette notion d’angle. On pourrait croire que 90° représente une lumière verticale. Mais non, cela représente une lumière horizontale. Ce qui est cohérant avec le fait qu’il capte moins de lumière à 90°. Avec une lumière qui arrive verticalement, nous pouvons sans autre travailler correctement.

### Dimensionnement

Pour une tension de 2.5V à lumière ambiante, il faut calculer la résistance à partir d’une autre choisi « aléatoirement ». Dans notre cas, nous mesurons la tensions (à la lumière ambiante) sur une résistance de 6.8kOhms. Nous pouvons ainsi déterminer la résistance pour 2.5V.

Dimensionnement Résistance série du capteur de lumière :

Une image contenant texte, diagramme, ligne, Police

Le contenu généré par l’IA peut être incorrect.Une image contenant texte, diagramme, ligne, Police

Le contenu généré par l’IA peut être incorrect.

Une image contenant texte, Police, blanc, écriture manuscrite

Le contenu généré par l’IA peut être incorrect.

Une image contenant texte, diagramme, ligne, Police

Le contenu généré par l’IA peut être incorrect.Dimensionnement Résistance de Bruit :

Une image contenant texte, diagramme, ligne, Tracé

Le contenu généré par l’IA peut être incorrect.

Une image contenant texte, Police, reçu, blanc

Le contenu généré par l’IA peut être incorrect.

Le but de ce diviseur de tension entre R16 et R19 est d’abaisser la tension pour ensuite pouvoir abaisser la valeur de R14 et éviter qu’elle soit trop grande.

Dimensionnement Condensateur du filtre passe-bas :

Pour une fréquence de coupure d’environ 50Hz et une résistance de 50kOhms mesuré sur plaque d’essais (Potentiomètre en série avec le condo pour pouvoir ajuster au mieux ) :

Une image contenant texte, Police, ligne, capture d’écran

Le contenu généré par l’IA peut être incorrect.

Après test, la constante de temps est jugée trop courte.

🡪x10 sur C 🡪680nF

Une image contenant texte, Police, reçu, blanc

Le contenu généré par l’IA peut être incorrect.

Il faut réajuster le condensateur pour garder la même constance de temps avec les nouvelles résistances.

Dimensionnement Résistance soustracteur :

Pour le soustracteur, nous avons besoin d’un gain de 1V/V

C’est pourquoi nous avons les mêmes valeurs de résistances pour R5,R9,R11 et R13.

Une image contenant texte, diagramme, ligne, Plan

Le contenu généré par l’IA peut être incorrect.

#### Estimation du courant de consommation du système

A lumière ambiante :

Une image contenant texte, diagramme, ligne, Plan

Le contenu généré par l’IA peut être incorrect.

A lumière maximale :

Une image contenant diagramme, Plan, ligne, texte

Le contenu généré par l’IA peut être incorrect.

A lumière minimale :

Une image contenant texte, diagramme, ligne, Plan

Le contenu généré par l’IA peut être incorrect.

## Rédaction du protocole de mise en service du prototype

### Protocole de mise en service

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Etape | Quoi faire | Valeur attendue | Valeur mesurée | Erreur | Validée | Mes. Ref. |
| 1 | Implanter les connecteurs d’alimentations  Mesurer les tensions d’alimentations aux différents endroits dans le pcb (connexions d’alimentation sur ICs, PIC)  … | - +5V  -5V | +4.9V  -4.98V | - |  | - |
| 2 | Implanter les condensateurs de découplages et la diode de protection  Mesurer les tensions d’alimentations | +5V  -5V | +4.99V  -4.98V | …% |  |  |
| 3 | Implanter le bloc Soustracteur  Mesurer la tension d’entrée et de sortie du bloc | Ue = ~2.5V  Us =~0V | Ue = 1.3V  Us = 9mV  (Ue est plus bas car la lumière ambiante est plus basse que les autres jours) |  |  |  |
| 4 | Implanter le bloc Additionneur Inverseur  Mesurer la tension d’entrée et de sortie du bloc | - Ue = ~0V  Us = ~2,2727V | -Ue = 9mV  Us = 2.278V | - |  | - |
| 5 | Implanter le bloc Générateur de bruit  Mesurer la tension de sortie du bloc | Us = ~1,6V |  |  |  |  |
| 6 | Implanter le bloc Comparateur à Hystérèse  Mesurer la tension d’entrée et de sortie du bloc  Mesurer la tension sur la PIN 2 | Ue = ~2.5V  Us = ~0V  ~2.5V |  |  |  |  |
| 7 | Mesure tension d’entrée et de sortie du comparateur avec Q2 complètement dans le noir | Uecomp = ~5V  Uscomp = ~5V |  |  |  |  |
| 8 | Mesure tension d’entrée et de sortie du comparateur avec Q2 un peu caché | 5V > Ue > 2.5V  Us = ~5V |  |  |  |  |
| 9 | Mesure du courant d’entrée à lumière ambiante | Ie+ = ~6.2mA  Ie- = ~-1.39mA |  |  |  |  |

### Mesure1

#### But de la mesure

#### Schéma de mesure

#### Liste des composants

#### Liste des instruments et leurs réglages

#### Tableaux des valeurs mesurées

### Mesure2

#### But de la mesure

#### Schéma de mesure

#### Liste des composants

#### Liste des instruments et leurs réglages

#### Tableaux des valeurs mesurées

## Liste de matériel

Une image contenant texte, capture d’écran, menu

Le contenu généré par l’IA peut être incorrect.

## Layout

Lors de la réalisation du Layout, il faut respecter plusieurs règles pour que le PCB puisse être commandé auprès de notre fabricant de PCB qui est EuroCircuit. Pour ce faire nous devons respecter une norme qui est la Classe 6C :

Une image contenant texte, capture d’écran

Le contenu généré par l’IA peut être incorrect.



Si cette classe 6C n’est pas respecté, le PCB passera en classe 7 ou D ce qui coûtera plus cher pour la fabrication du PCB, mais surtout, le PCB ne saura pas de très bonne qualité et a plus de risque d’avoir des défauts dessus et de ne pas bien tenir le temps.

### Contraintes (Rules Check)

J’ai essayé de mettre une clearance la plus grande possible. Malheureusement, comme il y’a des ICs sur le PCB, il n’est pas possible de faire plus grand que 0.2mm sinon les pads de l’IC vont crier lors du DRC.

Une image contenant texte, capture d’écran, Police, logo

Le contenu généré par l’IA peut être incorrect.

Une image contenant texte, capture d’écran, diagramme, carte

Le contenu généré par l’IA peut être incorrect.

J’ai défini une BoardOutlineClearance de 0.5mm pour éviter de positionner des composants trop proches du bord. C’est une sécurité pour éviter d’avoir des dommages dû à la fraise de découpage.

#### Largeurs des pistes de cuivre

Pour la largeur de piste, j’ai mis une largeur de piste assez grande (0.762mm) pour les pistes d’alimentation tel que le -5V et le GND. Pour le reste des pistes, elles ont une largeur de 0.4mm. Ces pistes sont normalement assez grandes pour pouvoir accueillir au maximum 10mA.

Une image contenant texte, capture d’écran, Police

Le contenu généré par l’IA peut être incorrect.

#### Diamètres des vias

Comme nous avons des pistes maximales de 0.762mm, on ne peut pas mettre de vias de diamètre plus petit. C’est pourquoi j’ai décidé de mettre des vias de diamètre 1mm pour avoir de bonnes connexions. Sur Altium, les vias sont configuré de base avec des freins thermiques. Je l’ai enlevé pour avoir une meilleure connexion.

Une image contenant capture d’écran, texte, Graphique, graphisme

Le contenu généré par l’IA peut être incorrect.

### Empreintes spécifiques

#### Connecteur d’alimentation

Une image contenant capture d’écran, cercle, texte, conception

Le contenu généré par l’IA peut être incorrect. Une image contenant conception

Le contenu généré par l’IA peut être incorrect.

Cette empreinte est un connecteur banane pour pouvoir facilement brancher les alimentations. Il y’en 1 en noir pour le GND, un rouge pour le +5V et un jaune pour le -5V.

#### AOP LMC6482\_SMD/THT

Une image contenant texte, capture d’écran, cercle, conception

Le contenu généré par l’IA peut être incorrect.Une image contenant texte, capture d’écran, Rectangle, diagramme

Le contenu généré par l’IA peut être incorrect.

Dans le projet, nous utilisons plusieurs AOP que ce soit comme soustracteur, additionneur ou encore comme comparateur, cela en fait 3. Nous avons donc choisi d’utiliser 2 ICs dual. Le comparateur est sur la version THT pour que les apprentis puissent le manipuler / le voir sur le TOP. Et le soustracteur ainsi que l’additionneur sont sur une version SMD qui sera présente sur le BOT car ceci représente le conditionneur, chose que les apprentis n’ont pas besoin de voir sur la couche TOP.

#### Phototransistor : SFH-3310

Une image contenant cercle, texte, capture d’écran, Graphique

Le contenu généré par l’IA peut être incorrect.Une image contenant plastique

Le contenu généré par l’IA peut être incorrect.

Voici l’empreinte du capteur de lumière (phototransistor). Il faut faire attention à la polarité.

#### Microcontrôleur

Une image contenant texte, capture d’écran, diagramme, Rectangle

Le contenu généré par l’IA peut être incorrect.Une image contenant Rectangle, capture d’écran, conception, carré

Le contenu généré par l’IA peut être incorrect.

Une image contenant texte, ligne, capture d’écran, Police

Le contenu généré par l’IA peut être incorrect.

Nous utilisons le PIC 16F15214-I/SN parce que nous avons très peu d’I/O (un seul). Donc pas nécessaire de prendre un PIC trop grand inutilement. Sur le PIC, nous avons les connexions pour le connecteur de programmation soit : ICSPDAT sur RA0 (Pin 7), ICSPCLK sur RA1 (Pin 6), MCLR sur RA3 (pin 4) et la sortie du PIC sur RA5(Pin 2). Un condensateur de découplage(100nF) a été connecté aux bornes d’alimentations de celui-ci pour lisser la tension et éviter des pics de tensions et donc des surtensions. Nous utilisons ce PIC précisément parce-que nous l’avons déjà employé à l’EMF donc le PIC est connu et déjà testé.

#### Connecteur de programmation

Une image contenant capture d’écran, Graphique, cercle, conception

Le contenu généré par l’IA peut être incorrect.Une image contenant texte, Police, nombre, capture d’écran

Le contenu généré par l’IA peut être incorrect.

Ceci est le connecteur de programmation. C’est un connecteur TagConnect assez utile puisqu’après que la programmation ait été implémenter, aucune contrainte mécanique peut venir perturber le PCB. Seul bémol, il faut acheter un câble spécial qui coûte assez cher…

## Firmware

Le firmware pour notre application n’est pas très conséquent. Cependant, il joue quand même un rôle clé parce-que sans ce code, le projet perd tout son sens étant donné que le but du projet est de démontrer l’effet du bruit sur un comparateur à hystérèse.

Donc comme énoncé au-dessus, le but est de générer un code pouvant créer un bruit de 20mV environ. Pour ce faire j’utilise la méthode de registre à décalage. C’est-à-dire que je vais décaler les bits et sortir certains bits pour les passer dans différentes portes XOR pour qu’au final je rajoute ce bit de sortie à l’entrée du registre à décalage. Ce qui nous permet d’avoir un bruit pseudo-aléatoire.

Une image contenant texte, capture d’écran, ligne, Police

Le contenu généré par l’IA peut être incorrect.

### Structogramme

Un structogramme est très utile pour pouvoir avoir une idée claire avant de commencer à écrire les lignes de codes. Il permet de déjà avoir les blocs et de pouvoir suivre une ligne et de ne pas se perdre.

#### Structogramme Main.C

Une image contenant texte, capture d’écran, Police, nombre

Le contenu généré par l’IA peut être incorrect.

#### Structogramme fonction registre à décalage

Une image contenant texte, Police, capture d’écran, nombre

Le contenu généré par l’IA peut être incorrect.

#### Structogramme InitPic()

Une image contenant texte, Police, capture d’écran, ligne

Le contenu généré par l’IA peut être incorrect.

### Code

/\*

\* File: test\_capteur.c

\* Author: PittetL02

\*

\* Created on May 23, 2025, 1:00 PM

\*/

//---------------------------CONFIGURATION--------------------------------

// CONFIG1

#pragma config FEXTOSC = OFF // External Oscillator Mode Selection bits (Oscillator not enabled)

#pragma config RSTOSC = HFINTOSC\_1MHZ// Power-up Default Value for COSC bits (HFINTOSC (32 MHz))

#pragma config CLKOUTEN = OFF // Clock Out Enable bit (CLKOUT function is disabled; I/O function on RA4)

#pragma config VDDAR = HI // VDD Range Analog Calibration Selection bit (Internal analog systems are calibrated for operation between VDD = 2.3V - 5.5V)

// CONFIG2

#pragma config MCLRE = EXTMCLR // Master Clear Enable bit (If LVP = 0, MCLR pin is MCLR; If LVP = 1, RA3 pin function is MCLR)

#pragma config PWRTS = PWRT\_OFF // Power-up Timer Selection bits (PWRT is disabled)

#pragma config WDTE = ON // WDT Operating Mode bits (WDT enabled regardless of Sleep; SEN bit is ignored)

#pragma config BOREN = ON // Brown-out Reset Enable bits (Brown-out Reset Enabled, SBOREN bit is ignored)

#pragma config BORV = LO // Brown-out Reset Voltage Selection bit (Brown-out Reset Voltage (VBOR) set to 1.9V)

#pragma config PPS1WAY = ON // PPSLOCKED One-Way Set Enable bit (The PPSLOCKED bit can be set once after an unlocking sequence is executed; once PPSLOCKED is set, all future changes to PPS registers are prevented)

#pragma config STVREN = ON // Stack Overflow/Underflow Reset Enable bit (Stack Overflow or Underflow will cause a reset)

// CONFIG3

// CONFIG4

#pragma config BBSIZE = BB512 // Boot Block Size Selection bits (512 words boot block size)

#pragma config BBEN = OFF // Boot Block Enable bit (Boot Block is disabled)

#pragma config SAFEN = OFF // SAF Enable bit (SAF is disabled)

#pragma config WRTAPP = OFF // Application Block Write Protection bit (Application Block is not write-protected)

#pragma config WRTB = OFF // Boot Block Write Protection bit (Boot Block is not write-protected)

#pragma config WRTC = OFF // Configuration Registers Write Protection bit (Configuration Registers are not write-protected)

#pragma config WRTSAF = OFF // Storage Area Flash (SAF) Write Protection bit (SAF is not write-protected)

#pragma config LVP = ON // Low Voltage Programming Enable bit (Low Voltage programming enabled. MCLR/Vpp pin function is MCLR. MCLRE Configuration bit is ignored.)

// CONFIG5

#pragma config CP = OFF // User Program Flash Memory Code Protection bit (User Program Flash Memory code protection is disabled)

// #pragma config statements should precede project file includes.

// Use project enums instead of #define for ON and OFF.

----------------------------------------------------------------------------------------

#include <xc.h>

#include <stdint.h>

//----- Constant Definiton ----------------------------

#define \_XTAL\_FREQ 4000000 // Fréquence d'oscillateur interne

//----- Global Variable Definitions------------------------

// LFSR 8 bits : taps sur bits 7, 5, 4, 3 (polynôme x^8 + x^6 + x^5 + x^4 + 1)

uint8\_t lfsr = 0xA5; // Seed non nul

//----- Function Declarations -----------------------------

uint8\_t lfsr\_next(void);

void InitPic(void);

//===== Main Function Implementation ======================

void main(void)

{

InitPic();

while (1)

{

uint8\_t out = lfsr\_next(); // bit pseudo-aléatoire

LATAbits.LATA5 = out; // sortie sur RA5

\_\_delay\_ms(10); // fréquence de sortie (ajuste au besoin)

}

}

//---------------------------------------------------------

//---------------------------------------------------------

// Sous programme lfsr\_next

// Auteur: yassinn

// Desc.: Initialisation du PIC

// Ver. Date: V00 20230207 Création (YYYYMMDD)

//--------------------------------------------------------

uint8\_t lfsr\_next(void)

{

uint8\_t bit = ((lfsr >> 7) ^ (lfsr >> 5) ^ (lfsr >> 4) ^ (lfsr >> 3)) & 1;

lfsr = (lfsr << 1) | bit;

return lfsr & 1;

}

//---------------------------------------------------------

//---------------------------------------------------------

// Sous programme InitPic

// Auteur: yassinn

// Desc.: Initialisation du PIC

// Ver. Date: V00 20230207 Création (YYYYMMDD)

//--------------------------------------------------------

void InitPic(void)

{

// TRISA0 = 0; // RA0 en sortie

// ANSELA = 0; // Tous les ports A en numérique

// LATAbits.LATA0 = 0; // État initial

// Config Oscillator

OSCCON = 0b01101010; // 4 MHz

//Configuration AD PIC

ANSELA = 0; // Configuration des IO en mode Digital

//ANSELB = 0; // Configuration des IO en mode Digital

//ANSELC = 0; // Configuration des IO en mode Digital

//Config Port

LATA = 0b00000000; //

//LATB = 0b00000000; //

//LATC = 0b00000000; //

//Config Tris

TRISA = 0b00000000; //RA2 = input (Switch), RA0 output

//TRISB = 0b00000000; //tout outputs

//TRISC = 0b00000000; //

}

## Phase de production en série

### Rédaction du protocole de mise en service de post-production

Sans Résistance d’hystérèse

|  |  |  |  |
| --- | --- | --- | --- |
| **Capteur Uref** | **Capteur Umes** | **Ue comparateur** | **Etat Led (Us)** |
| Lumière ambiante | Lumière ambiante | 2.5V | Scintille |
| Lumière ambiante | Caché | 5V | Allumée |
| Lumière ambiante | Partiellement caché | 2.5V > Ue > 5V | Allumée |
| Caché | Lumière ambiante | 0V | Eteinte |
| Partiellement caché | Lumière ambiante | 0V > Ue > 2.5V | Eteinte |
| Caché | Caché | 2.5V | Scintille |

Avec Résistance d’hystérèse

|  |  |  |  |
| --- | --- | --- | --- |
| **Capteur Uref** | **Capteur Umes** | **Ue comparateur** | **Etat Led (Us)** |
| Lumière ambiante | Lumière ambiante | 2.5V | Eteinte |
| Lumière ambiante | Caché | 5V | Allumée |
| Lumière ambiante | Partiellement caché | 2.5V > Ue > 5V | Allumée |
| Caché | Lumière ambiante | 0V | Eteinte |
| Partiellement caché | Lumière ambiante | 0V > Ue > 2.5V | Eteinte |
| Caché | Caché | 2.5V | Eteinte |

### Rédaction du mode d’emploi utilisateur

L’utilisateur peut choisir entre différents modes. Il peut manuellement ajouter ou enlever du bruit fournit par le PIC. Il peut déconnecter l’entrée pour mettre une source de tension externe. De même pour la sortie, il peut déconnecter « l’éclairage public » pour mesurer la sortie avec un oscilloscope. Et pour finir, l’utilisateur peut manuellement aussi connecter / déconnecter la résistance en contre-réaction aux bornes du comparateur pour pouvoir observer les différences avec ou sans hystérèse.

Pour faire les mesures, il faut faire attention à ne pas faire trop d’ombre et perturber le capteur de lumière Q1 qui doit rester exposer au maximum à la lumière ambiante.

Pour commencer, il faut venir déconnecter le bruit et déconnecter la résistance R4(hystérèse). Il faut maintenant observer le comportement de la Led P1 (devrait être éteinte). Tirer des hypothèses.

Ensuite, avec la main, il faut venir l’approcher gentiment du capteur de lumière Q2 (Il fait nuit donc l’éclairage public devrait s’allumer). On peut venir brancher un oscilloscope pour pourvoir voir la tension d’entrée pour observer la variation de tension selon la lumière que reçoit Q2.

Refaire la même démarche mais avec l’ajout de bruit. Observez le comportement de la Led P1 (devrait scintiller). Tirer des hypothèses.

Maintenant, il faut connecter R4 et le bruit et refaire les tests avec le capteur Q2. Qu’observez-vous ? Pourquoi selon vous P1 ne se comporte plus comme lors de la mesure avec le bruit ?

## Liste des outils utilisés et leur version

Oscilloscope KEYSIGHT InfiniiVision 2014A EMF-INDU-PHIE-0074

Alimentation DC xantrex XPL-30-2T TRIPLE DC Power Supply EMF-INDU-PHIE-0371

Générateur de fonction Agilent 33250A EMF-INDU-PHIE-0006

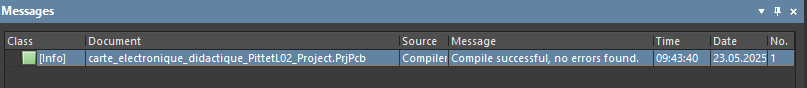
Multi-mètre AMPRODE 34XR-A 19100479A

PIC KIT 4 MICROCHIP 10-10094-R1

# Contrôle

## Validation ERC

La validation ERC (Electrical Rules Check) sert à valider les connexions électriques du schéma pour que le routage puisse se faire sans erreurs par la suite.



## Validation DRC

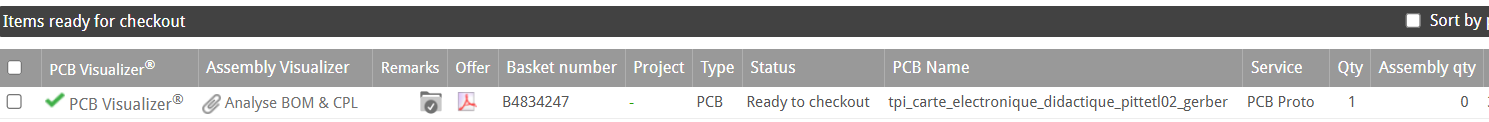
La validation DRC (Design Rules Check) sert à valider le PCB selon les règles que l’on met pour la conception (Règles de la classe 6C). Cette validation sert à pouvoir générer les fichiers gerbers sans craintes d’avoir des erreurs de routage et de placements de composants.

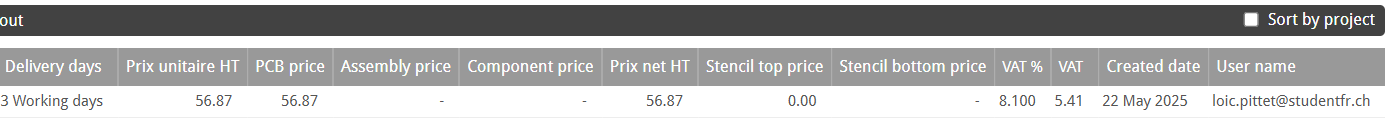
Une image contenant texte, capture d’écran, Police, ligne

Le contenu généré par l’IA peut être incorrect.

## Validation Eurocircuits

La validation EuroCircuit sert à certifier que notre PCB rentre bel et bien dans la classe 6C et que tous les gerbers sont présents pour que la fabrication se fasse avec tous ce dont nous avons besoin.





## Validation par la mise en service du prototype

# Evaluation

## Etat du projet

Fonctionnel

## Liste des modifications pour la prochaine version

### AOP Comparateur

Le comparateur inverseur a été routé à l’envers et est donc devenu un AOP inverseur. Il a donc fallu donc couper les pistes et tirer des fils pour inverser les connexions. Les pins d’alimentations sont donc aussi inversés. Il a fallu inverser ces pins en coupant la PIN du comparateur et tirer des fils pour refaire la connexion correcte.

J’ai aussi mis la sortie du deuxième comparateur à la masse. Ce qui est une grosse erreur. L’AOP va chercher à sortir une tension sur cette sortie, ce que l’on ne veut pas. Il a fallu couper la pin 7 du comparateur.

Ces erreurs ont déjà été corrigées et une nouvelle version des gerbers a déjà été généré.Une image contenant texte, diagramme, ligne, Tracé

Le contenu généré par l’IA peut être incorrect.Une image contenant texte, diagramme, ligne, nombre

Le contenu généré par l’IA peut être incorrect.

## Liste des délivrables

|  |  |  |  |
| --- | --- | --- | --- |
| pcb | **Schématique (.pdf)** | | |
|  | Fichier : | projectname-pcb-schema-vx.x.x.pdf | annexé Flèche : pivoter à droite |
|  | Répertoire : | .\projectname\pcb\pdf\ |  |
| pcb | **Plan d’implantation (.pdf)** | |  |
|  | Fichier : | projectname-pcb-implantation-bot-vx.x.x.pdf | annexé Flèche : pivoter à droite |
|  | Fichier : | projectname-pcb-implantation-top-vx.x.x.pdf | annexé Flèche : pivoter à droite |
|  | Répertoire : | .\projectname\pcb\pdf\ |  |
| pcb | **Gerber (.zip)** | |  |
|  | Fichier : | projectname-pcb-gerber-vx.x.x.zip |  |
|  | Répertoire : | .\projectname\pcb\ |  |
| fwr | **Firmware (.hex)** | |  |
|  | Fichier : |  |  |
|  | Répertoire : |  |  |
| swr | **Software (.exe)** | |  |
|  | Fichier : |  |  |
|  | Répertoire : |  |  |
| mec | **Plan mécanique (.pdf)** | |  |
|  | Fichier : |  |  |
|  | Répertoire : |  |  |
| rpt | **Liste de matériel (.pdf)** | |  |
|  | Fichier : | projectname-pcb-bom-vx.x.x.pdf | annexé Flèche : pivoter à droite |
|  | Répertoire : | .\projectname\pcb\bom\ |  |

## Amélioration possible

Ajour d’une source lumineuse sur le capteur de référence pour imiter un « soleil » et sur le capteur de mesure pour avoir des capteurs qui ne sont pas perturbé par l’extérieur (nuages, éclairage…). Ajout d’une LED lampadaire pour avoir une application plus réaliste et sympa a utilisé.

Une image contenant outil, Équipement médical, personne, crochet

Le contenu généré par l’IA peut être incorrect.

Ajout d’un petit rail avec un cache coulissant sur le capteur de mesure pour pouvoir plus facilement et plus précisément cacher la lumière et ainsi avoir des valeurs de mesures plus précises.

## Conclusion et avis personnel

### Signatures

|  |  |  |
| --- | --- | --- |
| Lieu: | EMF, 1705 Fribourg, Switzerland | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |
| Date: |  | myname myfirstname |

### Sources

* <https://www.fr.ch/emf>

### Version de ce document

|  |  |  |  |
| --- | --- | --- | --- |
| version | date | qui | modification |
| V1.0.0 | 20210510 | gauchl | Première version |
|  |  |  |  |

### Annexes