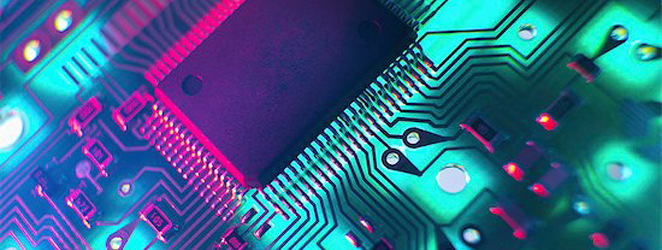


**ELECTRONICIEN/NE CFC  
 ELEKTRONIKER/IN-EFZ**



Jeu du Moulin 2025

V1.00

**Jeu du moulin**

Métier, Beruf

Bourquenoud Alexandre

2025

EMF - Ecole des Metiers de Fribourg

SWITZERLAND

Table des matières

[1 Information 2](#_Toc72079929)

[1.1 Objectifs du projet 2](#_Toc72079930)

[1.2 Schéma de principe de l'environnement 2](#_Toc72079931)

[2 Planification 3](#_Toc72079932)

[2.1 Calendrier prévisionnelle des tâches à réaliser 3](#_Toc72079933)

[2.2 Calendrier réel des tâches réalisées 3](#_Toc72079934)

[3 Décision 4](#_Toc72079935)

[4 Réalisation 5](#_Toc72079936)

[4.1 Schéma bloc de l'électronique à développer 5](#_Toc72079937)

[4.2 Description des éléments principaux 5](#_Toc72079938)

[4.3 Schématique, Dimensionnement & simulation 5](#_Toc72079939)

[4.4 Rédaction du protocole de mise en service du prototype 5](#_Toc72079940)

[4.5 Liste de matériel 6](#_Toc72079941)

[4.6 Layout 6](#_Toc72079942)

[4.7 Firmware 7](#_Toc72079943)

[4.8 Software 7](#_Toc72079944)

[4.9 Phase de production en série 7](#_Toc72079945)

[4.10 Liste des outils utilisés et leur version 7](#_Toc72079946)

[5 Contrôle 8](#_Toc72079947)

[5.1 Validation ERC 8](#_Toc72079948)

[5.2 Validation DRC 8](#_Toc72079949)

[5.3 Validation Eurocircuits 8](#_Toc72079950)

[5.4 Validation par la mise en service du prototype 8](#_Toc72079951)

[6 Evaluation 9](#_Toc72079952)

[6.1 Etat du projet 9](#_Toc72079953)

[6.2 Liste des modifications pour la prochaine version 9](#_Toc72079954)

[6.3 Liste des délivrables 9](#_Toc72079955)

[6.4 Amélioration possible 9](#_Toc72079956)

[6.5 Conclusion et avis personnel 9](#_Toc72079957)

# Information

## Objectifs du projet

L’objectif de ce projet est de concevoir un prototype fonctionnel du jeu du moulin. Cet objet pourrait être utiliser par l’EMF pour promouvoir le métier d’électronicien. J’ai pu m’aider d’une première version non terminer pour pouvoir l’utiliser comme fondation du projet.

Pour démontrer le fonctionnement du projet, je ne suis pas tâché de recréer le jeu du moulin en lui-même, mais plutôt de montrer que chaque touche et sa LED est fonctionnel.

### Résumé du cahier des charges

Je dois créer le prototype du jeu du moulin sur PCB en continuant et finissant le travail commencer par M. Samuel Egli et créer un code pour démontrer le fonctionnement du prototype en lisant les touches et en allumant les LEDs en fonction des touches. La programmation de l’uC ce fera grâce a un connecteur TC2030.

Pour ce faire je devrais adapter le schéma déjà fourni d’un STM32L072RBT à un MCU STM32L072CBU et rajouter les composants manquants en suivant certaines contraintes et la norme IEC 60617.

À cause du manque de pin de l’uC pour maitriser chaque touches et LEDs une part une, Je devrai utiliser 2 systèmes de matrices (un pour les LEDs et un pour les touches capacitives).

Je dois utiliser des composants SMD et des boitiers 0603 pour ne pas avoir d’agrandissement optique du montage.

Pour l’alimentation du circuit, je dois utiliser un USB-C ou une alimentation à 5V par 2 connecteurs et ensuite réguler la tension pour avoir un VCC = 3.3V.

Le PCB a des dimensions imposées de 10cm x 10xm mais ses dimensions peuvent être négocié si nécessaire. La majorité du circuit de contrôle des touches et des LEDs sera placé sur la face BOTTOM au centre de la plaque pour l’esthétique qui est essentiel au projet.

## Schéma de principe de l'environnement

### Description des blocs principaux

Le fonctionnement du projet est basé sur 4 parties principales.



**1 :** **Touches capacitives.**

Ces touches seront utilisées pour manipuler les pions des joueurs.

**2 :** **Microcontrôleur STM32L072CBU**

Ce microcontrôleur sera utilisé pour commander toutes les parties de l’objet.

**3 : LEDs**

Ces LEDs indiqueront le positionnement et l’équipe du joueurs. Elles seront contrôlées par l’IC et à l’aide des touches capacitives.

# Planification

## Une image contenant texte, capture d’écran, Parallèle, ligne Le contenu généré par l’IA peut être incorrect.Calendrier prévisionnelle des tâches à réaliser

## Calendrier réel des tâches réalisées

# Décision

|  |  |  |  |
| --- | --- | --- | --- |
| Date | Qui | Quoi | Pourquoi |
| 13.05.2025 | Bourquenoud Alexandre, Egli Samuel, Cunha Fabio | Changement de boitier de l’uC | Pour avoir plus de place pendant le layout. |
|  |  |  |  |
|  |  |  |  |

# Réalisation

## Schéma bloc de l'électronique à développer

Pour le schéma bloc, j’ai commencé par reprendre le même que celui imposer car il contient déjà tout ce qui est nécessaire. Le voici :



*Changements si nécessaire :*

## Description des éléments principaux

### Généralité sur les Touches Capacitives

Une image contenant cercle, Caractère coloré, Graphique, capture d’écran

Le contenu généré par l’IA peut être incorrect.Elles sont créées directement sur le PCB à l’aide d’empreintes spécifiques qui engendre une capacité. Cette capacité variera si un doigt est présent sur la touche.

Pour détecter si la touche est pressée ou non on utilise un autre condensateur de valeur connue (Cs, Condensateur de sampling, ou échantillonnage en français).

Une image contenant texte, Parallèle, capture d’écran, fournitures de bureau

Le contenu généré par l’IA peut être incorrect.La méthode de détection est la suivante : On charge la capacité de la touche en la soumettant à une tension constante, puis on transfert cette charge sur le condensateur d’échantillonnage en fermant le circuit de manière que le condensateur chargé soit en série avec Cs. Après l’équilibrage des charges, on mesure la tension aux bornes de Cs et ou pourra déterminer si un doigt (ou autre diélectrique) est présent sur la touche. Cette tension est ensuite passé à travers un CAN et est finalement interprété par le MCU en fonction des paramètre sélectionnés.

Avec cette formule, on peut voir la relation entre la tension, la capacité et les charges dans les condensateurs.

Quand on referme le circuit et qu’on lie les deux capacités, on a un schéma qui ressemblerait a celui-ci :

Et on pourra donc voir que la tension UCs change en fonction de la capacité de la touche qui elle-même change en fonction de si la touche est pressé et aussi de qu’est ce qui presse sur la touche à cause de la … différente des matériaux.

Je voulais inclure un calcul de la capacité des touches, mais après une conversation avec M. Berset, je me suis rendu compte que ce n’était pas nécessaire au fonctionnement des touches et très bancal.

### Généralité sur le MCU STM32 UFQFPN48

### Généralité sur les LEDs AAA3528SURKCGKC09

Ces composants sont des LEDs bicolores de couleur rouge et vert produites par Kigbright. Elles sont utilisées pour montrer quel joueur est en possession de la case -> Rouge pour un joueur, vert pour l’autre et orange/rien quand la case est vide. Éteindre les LEDs quand la case est inoccupée pourrait s’avérer être une meilleure solution pour sauver des opérations du MCU. La LED rouge fonctionne avec une tension typique de 1.95V et un courant de 20mA, sa tension max est de 2.5V tandis que la LED verte consomme 20mA à 2.1V typ. Avec le même maximum de 2.5V. voici le Pinout du composant prit dans le datasheet.

Une image contenant diagramme, Police, ligne, Dessin technique

Le contenu généré par l’IA peut être incorrect.Idéalement, pas plus que deux LEDs (donc un de ces composant) seront allumé en même temps pour ne pas demander une consommation de courant trop élevée. Je vais utiliser un code que j’ai écris il y a un moment pour gérer les LEDs de façon qu’on ait l’impression que toutes les LEDs allumé le soit en même temps. Pour ce faire je dois simplement les allumer puis éteindre dans un ordre de manière plus rapide que la perception humaine. Je dois garder en tête que les touches capacitives sont aussi dépendantes d’une interruption donc il est possible que le fonctionnement correcte de la matrice soit compromis.

## Schématique, Dimensionnement & simulation

### Schématique globale

J’ai aussi récupéré une schématique partiellement complète de Monsieur Egli mais il avait utilisé un autre IC que celui qui m’était imposer alors j’ai dû le changer. Dans les pages suivantes seront disposé les schémas effectués par Monsieur Egli :





Voici ci-après les schémas après mes changements avec des commentaires sur la raison des changements.

### Dimensionnement

#### Estimation du courant de consommation du système

#### Régulateur de tension

Une image contenant texte, écriture manuscrite, document

Le contenu généré par l’IA peut être incorrect.La valeur en tension m’a été demandé à 3.3V mais j’ai dû calculer le courant maximum avant de pouvoir la mesurer. Je suis donc aller voir dans les cas extrêmes pour l’uC ce qui nous donne un courant max de 105mA ([Datasheet - STM32L072x8 STM32L072xB STM32L072xZ - Ultra-low-power 32-bit MCU Arm®-based Cortex®-M0+, up to 192KB Flash, 20KB SRAM, 6KB EEPROM, USB, ADC, DACs](https://www.st.com/resource/en/datasheet/stm32l072v8.pdf) P.63). Les deux autres matrices utiliseront aussi du courant donc j’ai fait des calculs pour les cas extrêmes de leur côté aussi. Pour la matrice de LEDs, on pourra activer un maximum de 2 LEDs à la fois. Avec la matrice de touches, j’estime que toutes les touches tireront le courant maximum en tout temps. Voici les calculs :

Il me faudra donc un régulateur qui a la capacité de fournir plus de 146mA.

#### Résistance pour le port USB-C

#### ***Microcontrôleur***

Pour le dimensionnement des condensateurs de découplage, toutes les informations nécessaires étaient fournies la documentation officielle de ST ([Getting started with STM32L0xx hardware development](https://www.st.com/resource/en/application_note/an4467-getting-started-with-stm32l0xx-hardware-development-stmicroelectronics.pdf)):

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

Le contenu généré par l’IA peut être incorrect.

##### Port mapping

#### Protection contre polarisation inverse

Une image contenant texte, écriture manuscrite, document

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.Pour la protection du circuit contre une polarisation inverse des TP1 et TP2, j’ai choisi d’utiliser un MOSFET channel P avec cette configuration :

Suggérer par le papier de ONSEMI sur la sélection des MOSFET pour la protection de polarisation inverse ([AND90146 - MOSFET Selection for Reverse Polarity Protection](https://www.onsemi.com/download/application-notes/pdf/and90146-d.pdf))

J’ai donc du dimensionné la diode ZENER D25 et la résistance R14 pour qu’en polarisation correct, la Zener conduise et donc fait conduire le MOSFET en imposant la tension de seuil UGS sur la Grille et la source. En lisant le datasheet, j’ai choisi une diode Zener 2.4V 0.5W en boitier SOD-123 et j’ai simplifier le schéma de manière a pouvoir calculer R14.

## 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 le connecteur P1  Implanter les résistances R2..R9  … | - | - | - |  | - |
| 2 | Mesurer la tension du régulateur à vide | UVDD\_5Vtyp = 5.00V  UVDD\_5Vmin = 4.95V  UVDD\_5Vmax = 5.05V | UsK1 = 5.03V | …% |  | Mesure1 |
| 2 | Mesurer la tension et le courant du régulateur en charge |  |  |  |  | Mesure2 |
| 3 | Implanter le microcontrôleur IC1 | - | - | - |  | - |

### 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

## Layout

### Contraintes (Rules Check)

#### Largeurs des pistes de cuivre

#### Diamètres des vias

### Empreintes spécifiques

#### Régulateur de tension

#### Microcontrôleur

## Firmware

### Machine des états

### Flowchart

### Structogramme

## Software

### Machine des états

### Structogramme

## Phase de production en série

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

### Rédaction du mode d’emploi utilisateur

## Liste des outils utilisés et leur version

# Contrôle

## Validation ERC

Après avoir fait valider mon schéma par le superviseur de mon projet M. Cunha Fabio, il restait quand même un warning. Ce warning est dû au fait que le régulateur K n’a pas de VCC en entré. On ne peut malheureusement pas mettre de VCC vers régulateur sinon on risque de compromettre le circuit entier. En effet, en mettant un VCC ici, le VCC du circuit sera fixé à 5V et le MCU ne peut pas supporter de telles tensions. Ce warning ne sera donc pas pris en compte.

## Validation DRC

## Validation Eurocircuits

## Validation par la mise en service du prototype

# Evaluation

## Etat du projet

## Liste des modifications pour la prochaine version

## 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

## 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