

# Pré-étude

---

**Ecole supérieure**  
Électronique

Laboratoire R110 PROJ

---

## Buzzer Wire Game

---

**Réalisé par :**

Santiago Valiante

**A l'attention de :**

Mr. Bovey et Mr. Castoldi

**Dates:**

Début du laboratoire : 30 Novembre 2022

Fin du laboratoire : 8 Décembre 2022



## Table des matières

Table des matières .....	3
Cahier des charges.....	5
Pré-étude du projet .....	5
Résumé du projet.....	5
Principe .....	5
But .....	6
Composition du projet .....	6
Schéma bloc Hardware.....	6
Hardware : .....	6
Boîtier .....	8
Firmware & Software .....	8
Représentation et explications des interactions externes .....	9
Choix de composant avec justification .....	10
Alimentation (pile) .....	10
Switch On Off .....	10
Microcontrôleur $\mu$ C .....	10
Connecteur USB + USB to Uart.....	11
Affichage LCD.....	11
Encodeur .....	11
Bande LED RGB.....	11
Buzzer.....	11
Evaluation des coûts.....	13
Planning.....	15
Conclusion et perspective .....	16
Perspective .....	16
Conclusion .....	16



## Cahier des charges

Voir le cahier des charges détaillé dans le répertoire :

K:\ES\PROJETS\SLO\2224\_BuzzerWireGame\doc

Nom du fichier : 2224\_BuzzerWireGame-CDC-v1.doc.

## Pré-étude du projet

### Résumé du projet

#### Principe

WireGame est un jeu qui consiste à guider une boucle métallique le long d'une longueur de fil en serpentín sans toucher la boucle au fil. La boucle et le fil sont connectés à une source d'alimentation de telle manière que, s'ils se touchent, ils forment un circuit électrique fermé.

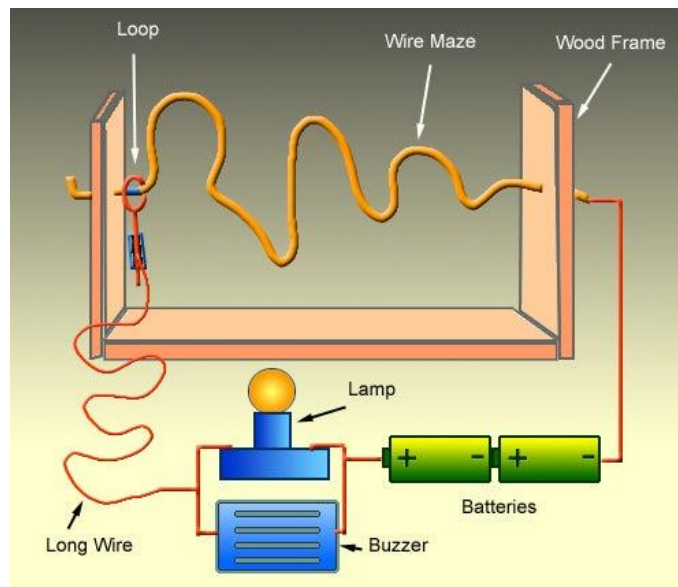


Figure 1

## But

Le but de ce projet est de développer un WireGame autonome et optimisé pour qu'une personne mal entendant puisse jouer. Il sera possible de gérer le système à travers une application C# depuis un PC connecter en USB ou manuellement avec l'encodeur.

## Schéma bloc global

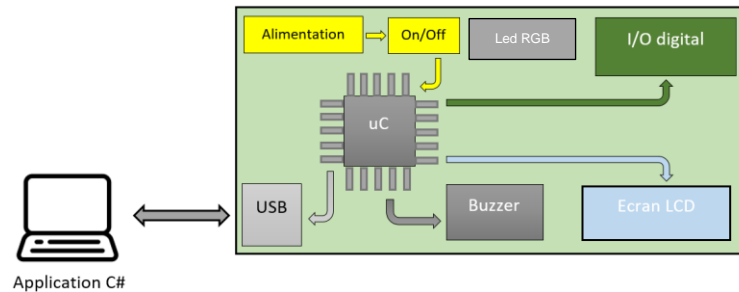
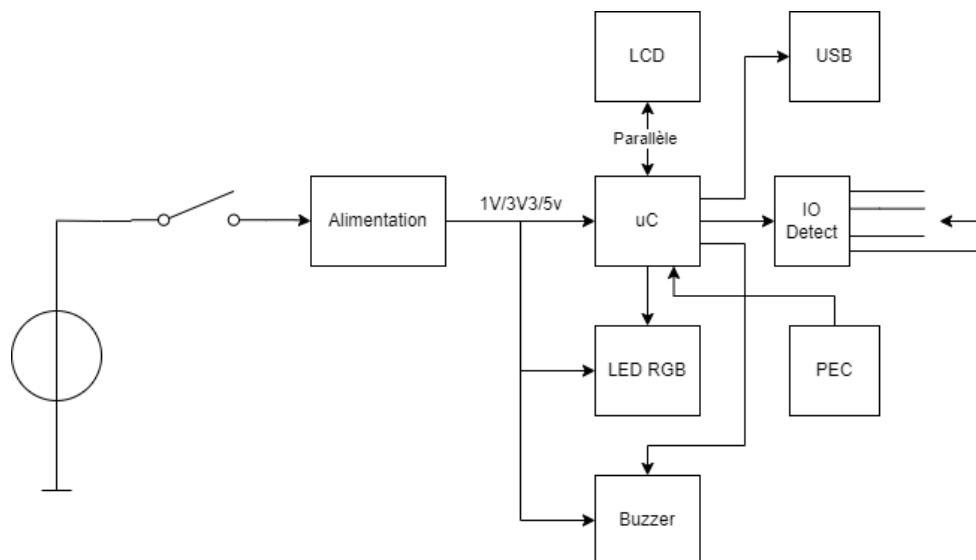


Figure 2

## Composition du projet

## Schéma bloc Hardware



## Hardware :

### Alimentation

Le projet devra être alimenté avec une pile, ensuite il y aura un bouton ON/OFF pour gérer l'alimentation.

### uC

Utilisation d'un pic 32 (Imposé par le client) pour la gestion de la communication USB, du LCD, des I/O et ect.

### LCD

L'affichage LCD sera utilisé afin d'indiquer le nombre de fois que l'on a touché le fil ainsi que la durée du parcours. De plus, afficher le nom du joueur et d'interfacer le menu du jeu.

### Port USB

Utilisation d'un port USB afin de communiquer avec un PC, pour que celui-ci prenne le lead sur la gestion du menu et du mode de jeu. De plus faire la gestion de statistiques (meilleurs scores avec nom du joueur).

### I/O detect

Les I/O seront utilisées afin de gérer le départ, l'arrivée et le toucher de la boucle.

### Start, touch et end :

Pour savoir quand le jeu débute, se fini ou que l'on touche le fils de cuivre deux idées me sont venues en tête.

- **1<sup>ère</sup> idée :**  
Alimenter les deux plaques en cuivre ainsi que le fils. Et une fois que l'on touche avec la boucle. Cela créera une boucle fermée et nous pourrons détecter un changement d'état.
- **2<sup>ème</sup> idée :**  
Avoir différentes valeurs résistives sur les trois parties afin de les différencier.

### Buzzer

Utiliser un buzzer comme aspect sonore lors d'un toucher.

### Leds RGB

Les leds RGB comme aspect visuel lors d'un toucher.

### Encodeur

Utilisation d'un encodeur pour pouvoir naviguer dans le menu.

## Boîtier

Pour le boîtier, nous avons l'idée de faire une boîte en plexi glace avec un socle en bois. Il faudra faire les plans du boîtier sur Solidworks.

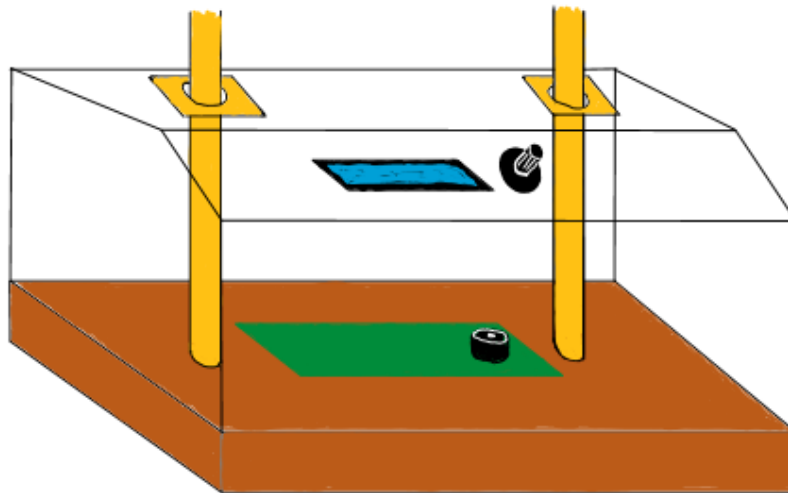


Figure 3

## Firmware & Software

### Application C#

Réaliser une interface de statistiques : meilleurs temps, nombre de fois toucher.  
Réaliser différent mode de jeu : compte à rebours, nbr de touche max, ect.

### Flowchart

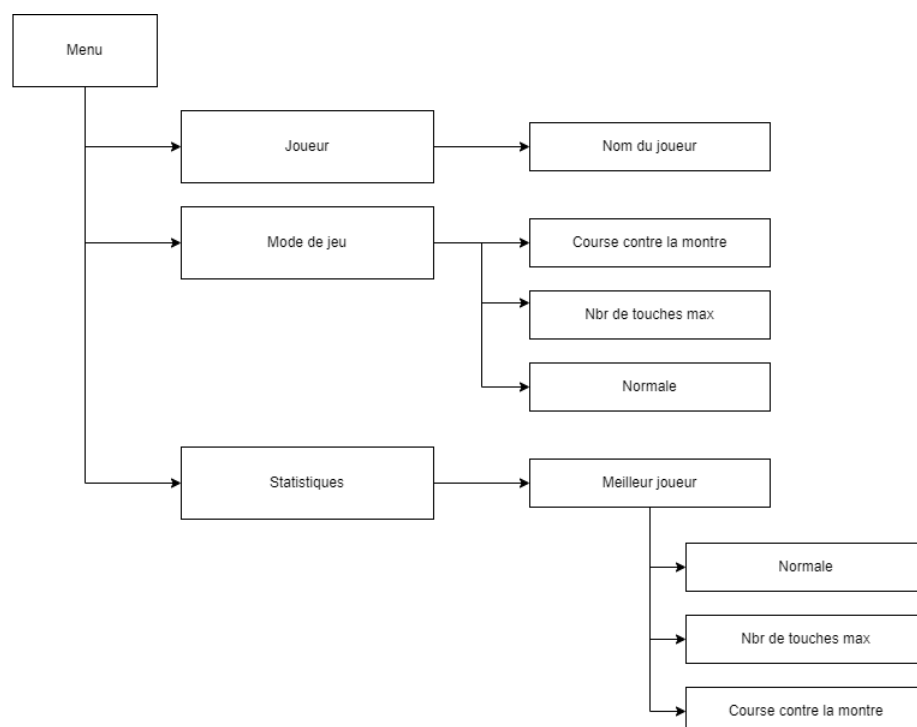


Figure 4



## Firmware uC

Réaliser différent mode de jeu : compte à rebours, nbr de touche max, ect.  
Gestion joueur.

### Flowchart

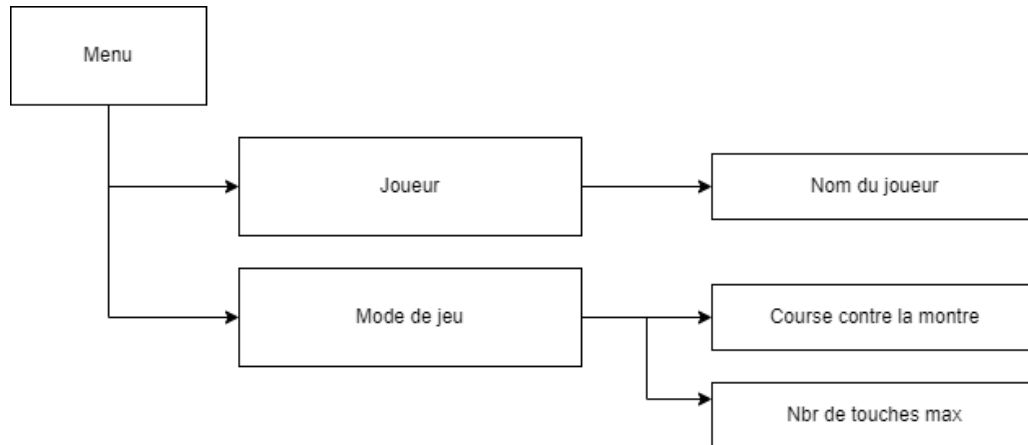


Figure 5

## Représentation et explications des interactions externes

Le joueur pourra interagir directement avec le système. Il y aura un bouton ON/OFF pour activer ou désactiver l'alimentation. A l'aide d'un affichage LCD et d'un encodeur, il pourra naviguer dans le menu. Mais, il pourra également interagir avec le système via un PC connecté via un port USB pour naviguer également dans le menu et en plus voir les statistiques des joueurs. Pour lancer le jeu, il suffira de décoller la boucle de la plaque de cuivre et de même pour y mettre fin mais à l'arrivée.

Voir les figures 2, 3, 4 et 5 pour un meilleur aperçu du menu et des interactions manuelles.

## Choix de composant avec justification

### Alimentation (pile)

La pile sera choisie en fonction de la puissance que consommera le circuit qui est encore à déterminer avec les LEDs RGB (priorité). Je pourrais prendre par exemples 2 piles en séries de 3v de 2Ah pour avoir une alimentation assez élevée et fournir assez de courant pendant 2 à 3h. Il faudra également des régulateurs step down pour alimenter en 5v, en 3v3 et 1v mes divers périphériques.

### Switch On Off

J'ai choisi un switch à bascule rond (RR11122FWC).



### Microcontrôleur µC

Dans le cadre de mon projet, on m'a imposé d'utiliser le PIC32. Ce µC est déjà utilisé dans les différents projets du client et nous avons déjà de bonnes connaissances de bases du µC.

Dans mon cas, le microcontrôleur va gérer **4 différentes GPIO**, cela permettra de commander une ou des Leds RGB, un écran LCD, un encodeur et un buzzer. Il y aura besoin d'une **communication UART** afin de communiquer avec un PC via un convertisseur USB to UART.

Choix : PIC32MX130F064B

Device	Pins	Program Memory (KB) <sup>(1)</sup>	Data Memory (KB)	Remappable Peripherals					Analog Comparators	USB On-The-Go (OTG)	I <sup>2</sup> C	PMP	DMA Channels (Programmable/Dedicated)	CTMU	10-bit 1 Msp ADC (Channels)	RTCC	I/O Pins	JTAG	Packages
				Remappable Pins	Timers <sup>(2)</sup> /Capture/Compare	UART	SPI/I <sup>2</sup> S	External Interrupts <sup>(3)</sup>											
PIC32MX130F064B	28	64+3	16	20	5/5/5	2	2	5	3	N	2	Y	4/0	Y	10	Y	21	Y	SOIC, SSOP, SPDIP, QFN



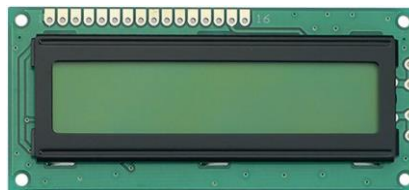
## Connecteur USB + USB to Uart

J'ai choisi de prendre un connecteur USB B coudé ainsi qu'utilisé un USB-to-UART Bridge Controller (CY7C64225-28PVXC) afin de convertir mon signal USB en UART pour communiquer depuis le PC avec mon  $\mu$ C. Le chip peut communiquer en Full Speed 12Mb/s.



## Affichage LCD

Pour l'écran LCD, je l'ai choisi afin de gérer au mieux l'interface du menu. Pour ça, j'ai décidé de prendre un écran LCD 4 lignes avec 20 caractères. Communication parallèle.



## Encodeur

J'ai choisi de prendre un encodeur (Pec12) qui a été lui aussi déjà utilisé lors de différents projets du client. Afin de naviguer dans le menu du jeu.



## Bande LED RGB

Pour un aspect physique, nous avons décidé de mettre une bande led. J'ai trouvé une bande de led de 1m, fonctionnant en 5v. Elle consomme 9W au max si les 30 leds sont allumées. J'utiliserais seulement la couleur rouge comme indication lors d'un toucher et je pense pas plus de 5 sec afin de ne pas vider la pile.



## Buzzer

J'ai choisi un buzzer (CVS-2308) qui a été déjà utilisé dans un projet du client. Les décibels sont assez élevés (88db) à titre d'indication lors d'un toucher.





## Evaluation des coûts

Nom	Quantité	Type	Fabricant	N° Fabricant	Fournisseur	Prix u	Total
Microcontrôleur	1	PIC32MX130F064B	Microchip	PIC32MX130F064B-I/SS	Mouser	fr. 3,55	fr. 3,55
Pile 3v 2 Ah	2	Lithium	Varta	CR AA	Distrelec	fr. 11,70	fr. 23,40
Switch On/Off	1	Switch à bascule	CIT Relay and Switch	RR11122FWC	Digi-Key	fr. 1,00	fr. 1,00
Connecteur USB	1	RND 205-01048	RND connect	RND 205-01048	Distrelec	fr. 0,47	fr. 0,47
USB-to-UART Bridge Controller	1	Bridge Controller	Infineon Technologies	CY7C64225-28PVXC	Mouser	fr. 4,28	fr. 4,28
Affichage LCD	1	DEM 20485 SYH	Display Elektronik GmbH	DEM 20485 SYH	Distrelec	fr. 22,83	fr. 22,83
Encodeur	1	PEC12	Bourns	PEC12R-4220F-S0024	Mouser	fr. 1,49	fr. 1,49
Bande LED RGB	1	LES RGB 9W 1m 5v	Seeed Studio	104020108	Distrelec	fr. 7,72	fr. 7,72
Buzzer	1	CVS-2308	CUI Device	CVS-2308	Mouser	fr. 3,00	fr. 3,00
PCB	1	-	Eurocircuit	-	Eurocircuit	fr. 60,00	fr. 60,00
Flis de cuivres 5mm	1	2m	-	-	Distrelec	fr. 10,00	fr. 10,00
Planche en bois	1	25X15 cm				fr. 5,00	fr. 5,00
Plexis	1					fr. 15,00	fr. 15,00
						<b>Total</b>	fr. 157,74



## Planning

No semaine projet	1	2	3	4	5	6			7	8	9	10	11		12	13	14	15	16	17	18			19	20	21	22	23	24	25	26	27	28
No semaine	46	47	48	49	50	51	52	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26
Date	16.11.22	23.11.22	30.11.22	07.12.22	14.12.22	21.12.22	28.12.22	04.01.23	11.01.23	18.01.23	25.01.23	01.02.23	08.02.23	15.02.23	22.02.23	01.03.23	08.03.23	15.03.23	22.03.23	29.03.23	05.04.23	12.04.23	19.04.23	26.04.23	03.05.23	10.05.23	17.05.23	24.05.23	31.05.23	07.06.23	14.06.23	21.06.23	28.06.23
Tâches							V	V						V								V	V										
Pré-étude				R	P																												
Design + Schéma											R	P																					
PCB																			R														
Montage																																	
Softwares																																	
Tests et mise au point																																	
Rédaction du rapport																																	
Préparation présentation + démo																																	
Présentations finales																															R	P	
Finalisation/corrections/documentation																																	

Commentaires

Rapport de pré-étude

Rapport de design

Fichiers de fabrication

 Rapport final  
 y compris annexes,  
 fichiers, résumé,  
 affiche

V : Vacances

R : Remise de documentation/dossier/fichiers. Peut être remis plus tôt.

P : Présentation.

Remise d'un rapport chaque semaine précédant une présentation.

## Conclusion et perspective

### Perspective

Je pense que les parties la plus compliquées du projet va être la partie software. En ce qui concerne la partie hardware du projet, je me fais moins de soucis.

### Conclusion

Etant ma première pré-étude, cela a été compliqué pour moi de commencer et de se poser les bonnes questions.

Il me faudra en premier lieu, calculer la puissance dissipée de mon système afin de déterminer mes piles pour l'alimentation de mon circuit.

Il me faudra également clarifier avec le client, le nombre de temps maximum que les LEDs RGB soient allumées lors d'un touché avec la boucle pour ne pas vider la pile si on la laisse pendant 1h en contact au fil de cuivre.

Ensuite, je vais devoir commencer par le design du schéma. Une fois cette étape réalisée, je vais être amené à faire le design du PCB, par la suite ceci me permettra de définir la taille finale du boitier. Ensuite, je vais être amené à gérer la communication entre mes différents modules par programmation.

Lausanne ETML-ES

Santiago Valiante

17.12.22