

Mini-projet SIN 1
CAHIER DES CHARGES

TITRE DU PROJET : Shot Timer

Contexte :

Un **Shot Timer** est un dispositif électronique utilisé principalement dans le cadre du tir sportif ou d'entraînement au tir, que ce soit avec des armes à feu, des armes airsoft. Sa fonction principale est de mesurer les temps liés aux tirs, permettant ainsi aux tireurs d'évaluer leur vitesse, leur efficacité et leurs performances.

Ici, nous ne traiterons pas les fonctions d'un Shot Time classique tel que :

- L'affichage des statistiques (Split Time¹ moyen, temps de réaction du premier tir)
- La programmation de scénarios
- Sauvegarde des sessions de tirs
- Démarrage aléatoire de la session de tir

Fonctions principales :

- Lancer une session de tir
- Arrêter une session de tir si pas de tir pdt 15s
- Reset le Shot Timer
- Comptabiliser les tirs
- Afficher des statistiques basiques (meilleur Split Time, temps total, nombre de tirs total)

Interactions avec l'utilisateur (boutons, affichages, ...)

a) Les entrées

NOM	DESCRIPTION	INTERFACE	TYPE
Tir	Simule l'action du tir.	KEY0	IN
Start	Permet de lancer un décompte (de 3 secs) avant la session de tir	KEY3	IN
Stop/RST	Permet d'arrêter la session, si le bouton est appuyé de nouveau, le Shot Timer est réinitialisé.	KEY2	IN

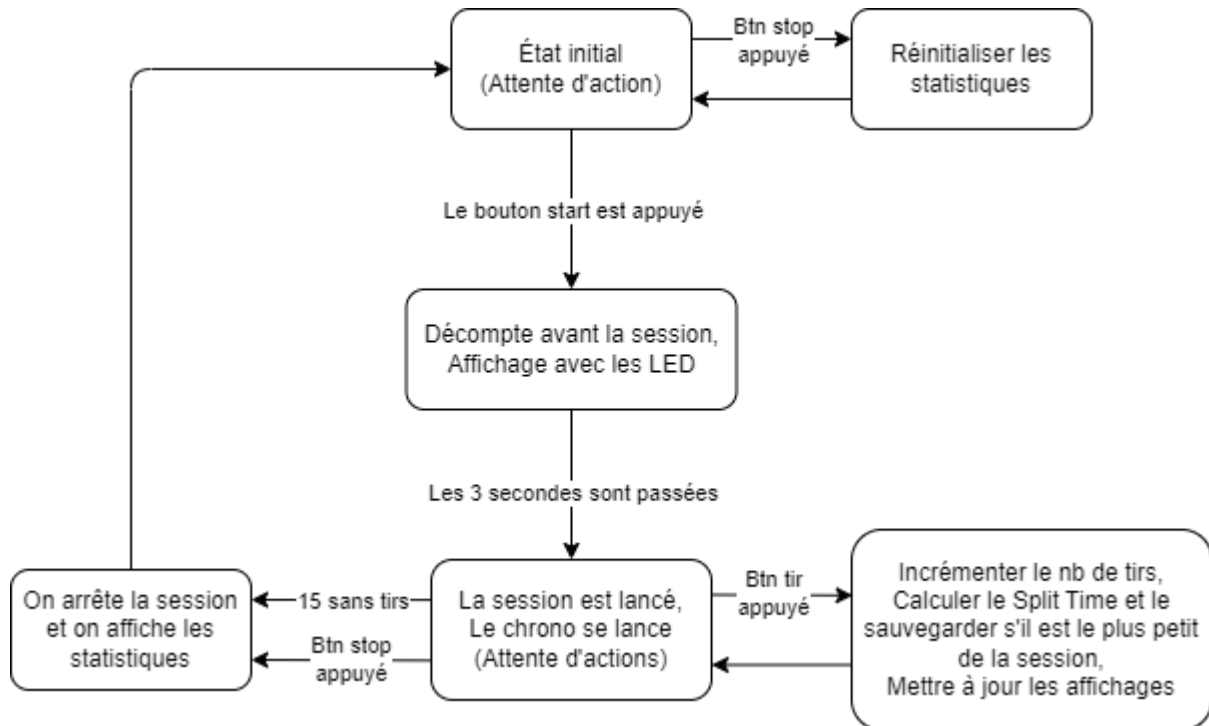
b) Les sorties

NOM	DESCRIPTION	INTERFACE	TYPE
Nombre Tirs	Affiche le nombre de tirs en temps réel de la session.	HEX6..7	OUT
Split Time	Affiche le Split Time courant pendant la session et le meilleur Split Time quand la session est terminée.	HEX4..5	OUT
Temps	Affiche le temps sous le format (SS : MSMS) si le temps dépasse une minute, on affiche sous le format (MM : SS).	HEX0..3	OUT
LED Décompte	Affiche le décompte de 3 secs avant la session de tir sous le format : 1sec (LEDR8..17) ; 2 sec (LEDR0..7) ; 3	LEDR0..17 LEDG0..7	OUT

¹ Split Time : Temps entre deux tirs

	sec (LEDG0..7)		
--	----------------	--	--

Modes de fonctionnement (texte, graphe, ...)



Simulation avec la carte Altéra

a) Les entrées

NOM	INTERFACE	TYPE
Tir	KEY0	IN
Start	KEY3	IN
Stop/RST	KEY2	IN

b) Les sorties

NOM	INTERFACE	TYPE
Nombre Tirs	HEX6..7	OUT
Split Time	HEX4..5	OUT
Temps	HEX0..3	OUT
LED Décompte	LEDRO..17 LEDG0..7	OUT

Problématiques à réfléchir :

- Le codage du temps en minutes, secondes, millisecondes.
- L'affichage du temps secondes millisecondes puis minutes secondes
- Soustraction du temps

NOM, Prénom : RAGON Matéis

d) Sauvegarde du plus petit temps

Pistes de réponses aux problématiques :

a) Le codage du temps :

Pour compter en seconde, on a besoin de 2 compteurs, l'un pour les unités et l'autre pour les dizaines, on a donc besoin de 2x4 bits soit 8 bits pour représenter les secondes. D'un autre côté, lorsque l'on compte une minute, on a besoin de 60 secondes, ainsi nous n'avons plus besoin de 2x4 bits, mais 1x4 bits pour les unités pour compter de 0 à 9. Et 1x3 bits pour les dizaines, puisqu'on s'arrête à 6, on peut prendre 3 bits, car la valeur binaire maximum de 3 bits est 111 à savoir 7. Donc nous n'avons plus besoin de 8, mais 7 bits. Une petite optimisation qui va être très importante dans nos calculs. Si l'on veut compter en plus les minutes, il faut alors la même quantité de bits que pour les secondes soit 7 bits.

Nous avons alors pour compter les minutes et seconds besoins de 14 bits. Là où réside le problème, c'est qu'un Shot Timer doit être précis ainsi, on peut tirer plusieurs fois à la seconde et donc avoir un Split Time de 0, ce qui n'est pas acceptable. Nous devons donc inclure les millisecondes à notre système temporel.

Les millisecondes vont à une vitesse de 1 kHz, car il y a 4 digits de précision, on va se permettre de réduire la précision des millisecondes à 2 digits, en effet, on a besoin d'être précis, mais pas à ce point. Nous allons donc passer des millisecondes aux centièmes de secondes. On va passer la vitesse de l'horloge de 1 kHz à 100 Hz. Les centièmes secondes ont pour valeur maximale 100, ainsi, on peut les représenter avec deux compteurs décimaux 4 bits.

Ce qui nous donne 7 bits pour les minutes, 7 bits pour les secondes et 8 bits pour les centièmes de secondes. C'est-à-dire 22 bits au total pour représenter le temps dans notre Shot Timer, avec une clock pour le temps de 100 Hz.

b) L'affichage du temps secondes millisecondes puis minutes secondes

Pour afficher le temps selon l'exigence ci-dessus, je pense qu'il faut poser la condition que s'il n'y a pas 0 minute, alors on n'affiche pas les minutes, sinon on les affiche et on n'affiche pas les centièmes de secondes.

c) Soustraction du temps

Pour commencer, nous avons besoin de faire un soustracteur 4 bits, et on fera la soustraction 4 bits par 4 bits avec un bit pour les retenues pour chaque cellule de temps. Maintenant, il faut réfléchir à faire un soustracteur 4 bits.

d) Sauvegarde du plus petit temps

Pour effectuer une sauvegarde du plus petit temps, on peut procéder de la manière suivante, on prend le temps lorsque l'utilisateur effectue un tir. On compare le Split Time en cours et celui dernièrement enregistré (celui-ci est stocké dans un compteur bloqué à la valeur du Split Time, à l'aide d'un chargement parallèle). Si aucun Split Time n'est enregistré, on compare donc avec [0..0].

Si la sortie de la comparaison est vraie (Split Time courant est plus petit que le Split Time enregistré), on effectue un chargement parallèle avec la valeur du Split Time courant. Il faut juste trouver le moyen d'activer la comparaison au moment du tir.