









## Description et mode d'emploi :

Dans le cadre du projet de système logique nous avons développé un réveil-matin de voyage multifonction pour la carte DE10-lite. Ce projet a été conçu sur la base de la molette rotatif et d'un bouton poussoir (Reset Général), nous permettant de naviguer et initialiser les différentes fonctions qui seront décrite ci-dessous. Par soucis de simplicité pour l'utilisateur, les différentes fonctions seront accessible via un système de menu. Afin de distinguer les différents clic exécuté, nous émettons des sons distinct pour chacun. Un clic simple (rapide) émet un bip court et veut dire « Suivant », une double pression du bouton fait entendre un double bip et veut dire *Reset* dans le cas du **Chrono**, alors qu'un clic long fait entendre un long bip et fait l'effet d'un retour au menu principale. De plus, si aucune touche n'est pressé pendant 20 secondes, celui-ci revient automatiquement sur l'horloge.

### Système de menu :

- \* **Hour**, affiche l'heure : 
- \* **Set Sunrise**, permet de choisir l'heure du réveil : 
- \* **Set Hour**, permet de régler l'heure après un *Reset* : 
- \* **Set Hourglass**, permet de mettre une minuterie qui sonne au bout du temps choisit : 
- \* **Hourglass**, permet de surveiller l'état de la minuterie : 
- \* **Chrono**, permet de visualiser un chronomètre, un clic pour start/stop, double clic = *Reset* : 
- \* **Set Ringtone**, permet de composé une sonnerie personnalisé : 
- \* **Set Display**, permet de composer un message personnalisé affiché avec la sonnerie du réveil : 

### Premier démarrage et réglage de l'heure :

Au démarrage, la carte affiche l'heure. Celle-ci commence à minuit (000000) et doit être régler. Pour ce faire il suffit de tourner la molette jusqu'à ce que le texte **Set Hour** défile sur l'écran. Une simple pression sur la molette permet de rentrer dans le menu. De là, les heures se mettent à clignoter (*Figure 1*), et peuvent être changer en tournant la molette. Il suffit de cliquer pour valider l'entrée et passer au minute puis au seconde. En cas d'erreur, il suffit de cliquer jusqu'au digit à modifier. Quand l'heure est entrée il suffit de presser longtemps (1 sec) pour valider l'heure et revenir au menu principale. En cas de bug, le bouton à droite (*Figure 2*) de la carte permet un *Reset* général, tout le reste de la navigation se fait uniquement avec la molette.



Figure 1



Figure 2

### Fonctionnement de Set Ringtone :

Une fois entrée dans le menu **Set Ringtone**, le menu position (*Figure 3*) permet de choisir la position de la note a enregistré/modifié. Par défaut le compteur est incrémenté de 1 (afin de facilité l'utilisation) par rapport à la dernière modification et il suffit de tourner la molette pour la changer.

Une fois ce choix validé, la sélection de la note peut être faite en tournant la molette. Il y 32 notes (*Figure 4*) qui comprenne les demi-ton (uniquement les dièse #) et le silence, pour un total de 2,5 octaves, entre le La 2 (220 Hz) et le Mi 5 (1397 Hz). Lors de la sélection la note est émise par le buzzer. Il suffit de faire un clic simple pour valider et passer à l'étape suivante. Pour finir vient la durée de la note (*Figure 5*) qui peut être choisit entre 50-800 ms par incrémentation en tournant la molette.

Lorsque la note est validé le menu **Jingle** (*Figure 6*) apparaît et la mélodie complète est joué. Un clic simple permet d'ajouter une autre note et un clic long permet de quitter menu.



Figure 3

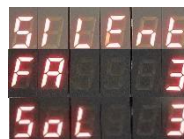


Figure 4



Figure 5



Figure 6

## Fonctionnement de Set Display :

**Set display** utilise le même principe de navigation que **Set Ringtone**, une fois dans le menu, il faut aussi choisir la position (Figure 3) du caractère à éditer. Un clic simple passe au menu de sélection du caractère (Figure 7). Les lettres: k, m, q, v, w, x, z ne peuvent pas être affichées dû à la limitation du 7 segments. Après chaque édition de caractère, le mot dans son état actuelle est affiché (Figure 8). Pour rajouter des lettres il suffit de cliquer pour recommencer avec le caractère suivant. La position est automatiquement incrémentée de 1 pour qu'il ne reste qu'à valider. La taille est limitée à 14 caractères. Si la taille est supérieure à 6 caractères, l'écran défile automatiquement.



Figure 7



Figure 8

## Architecture du projet :

### FSM général :

Notre machine d'état prend la forme d'un compteur qui est incrémenté/décrémenté par la rotation de la molette. Cette approche a l'avantage de permettre d'ajouter de nouveaux modules sans difficulté. Ce compteur n'est actif que si aucun sous-module n'est actif. La valeur actuelle du compteur ainsi que l'adresse précédemment validée par l'utilisateur sont passées aux autres périphériques et ceux-ci comparent l'adresse validée avec la leur pour déterminer si ils sont sollicités.

### Système de menu et délégation :

Pour déterminer quel module est actif, nous utilisons un système de délégation. Chaque machine d'état de sous-module possède un état **IDLE** tel que les commandes utilisateur sont ignorées dans cet état. Pour sortir un module de l'état **IDLE**, il faut que celui-ci reçoive une adresse correspondante à la sienne (Figure 9).

Après cet événement les commandes et l'affichage lui sont transférés et le sous-module les garde aussi longtemps qu'il ne reçoit pas un clic long qui le ramène à son état **IDLE** et désactive l'état **BUSY** (Figure 10 et 11).

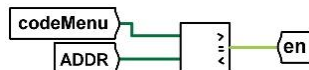


Figure 9

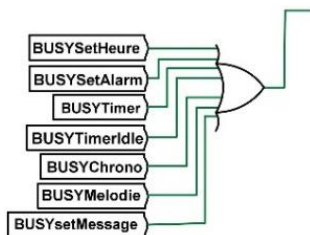


Figure 10

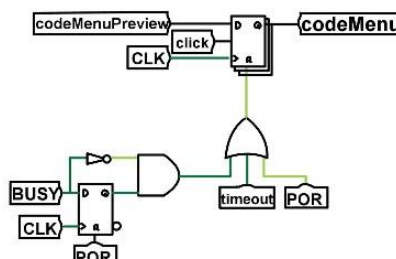
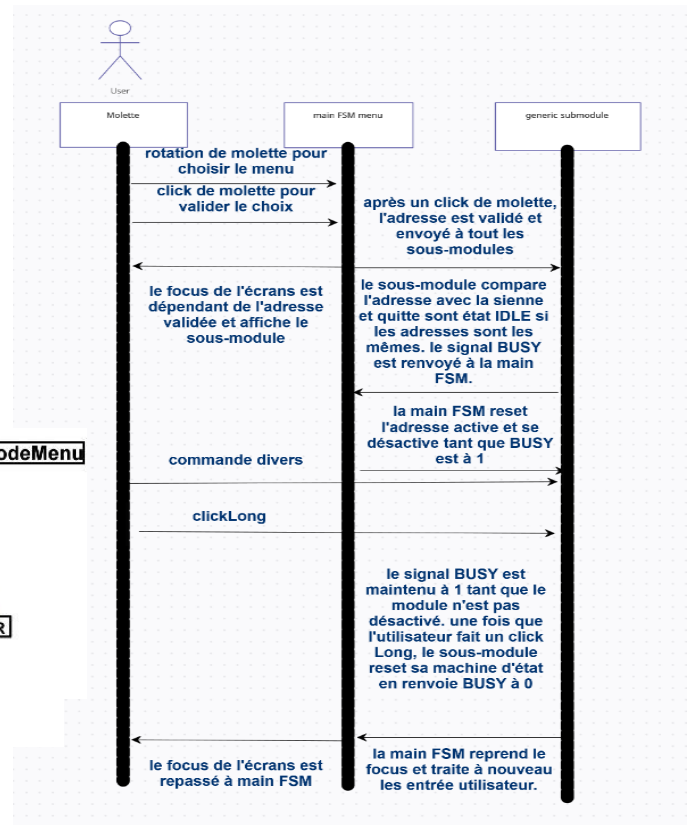


Figure 11



## Solution technique originales :

### Logique du module des messages défilant et taille des messages personnalisé :

Pour afficher des messages de taille supérieure à six caractères, nous avons développé un module pour faire défiler le message. Nous avons fixé la taille interne maximale des messages à 14, le meilleur compromis à nos yeux entre longueur de message et temps d'affichage pour 14 caractères. Le menu défilant utilise un registre à décalage dans lequel sont chargé les 14 lettres. Un compteur se charge de *Reset* le registre à décalage au bout du message. La taille du message personnalisé est calculé via une série de comparateur qui cherche les 0 depuis la fin du message (les mémoires non modifié étant à 0). Ainsi on peut en déduire la taille effective du message personnalisé. Si la taille est inférieure à 7, le message restera statique.

### Décodeur seconde vers minutes DCB :

Notre *Horloge* utilise trois compteurs en parallèle. L'un pour les secondes avec une période d'une minute, un second pour les minutes avec une période de 3599 secondes et un troisième avec une période de 86399 secondes. Pour décoder le nombre de secondes en heures nous avons utilisé 24 comparateurs cependant cette approche nécessite beaucoup de place et n'était pas envisageable pour les minutes. Nous avons écrit un script python qui a généré la table de vérité pour le décodeur (4096 lignes). Puis avons laissé *Logisim* la créer. Nous avons ainsi obtenu un convertisseur capable de transformer l'état du compteur 12 bits en deux nombres en DCB.

### Machine d'état et registre pour les mélodies/messages personnalisé :

La machine d'état tourne en boucle entre le choix de l'adresse de la note/caractère, la valeur de la note/caractère ainsi que le tempo dans le cas de la mélodie. Chaque état active un compteur qui est incrémenté par la rotation de la molette. La transition du dernier état au premier génère un signal de validation qui permet à la mémoire d'enregistrer les données. Ce signal de validation est aussi utilisé pour incrémenter l'adresse et ainsi reprendre l'édition au caractère suivant de manière automatique.

### Driver 7seg et alphabet étendu :

Pour afficher autres choses que des chiffres, nous avons dû créer notre propre décodeur 7-segment. Les caractères sont codés sur 6 bit. Le premier contrôle le point décimale, le second indique si on est en mode avancé pour ne pas casser la rétrocompatibilité avec les décodeurs classique tel que celui proposé par Logisim. Si le mode avancé est sélectionné, on continue l'alphabet au-delà de F. Comme énoncé précédemment certaines lettres ne peuvent pas être affichées tel que le « M », c'est pourquoi nous avons dû utiliser des synonymes pour certains mots tel *Sunrise* pour Alarme ou encore *Hourglass* pour *Timer* et n'avons pas été satisfait par un symbole qui puisse les remplacer.

### Gestionnaire de clic :

Pour faciliter l'utilisation du réveil, nous avons décidé de n'utiliser que l'encodeur rotatif pour traiter les entrées données par l'utilisateur. De là est apparu la nécessité de différencier plusieurs types de clic tel que le clic simple, double et long. Pour se faire nous avons conçu un détecteur de double clic ainsi qu'un détecteur de clic long. Nous attendons la réponse des 2 modules pour passer l'information en aval. Le détecteur de clic long utilise un compteur qui est amorcé au flanc montant de notre signal. Si il n'y a pas eu de flanc descendant dans le temps imparti prédéfini, il s'agit d'un clic long.

De manière analogue, le détecteur de double clic utilise un compteur qui génère un *Timeout* amorcé par le premier flanc montant. Si un second flanc montant arrive dans le temps imparti, alors il s'agit d'un double clic.

## Conclusion :

Nous avons réalisé un réveil multifonction avec un chronomètre ainsi qu'un minuteur. De plus il possède plusieurs aspects complexe tel qu'un menu animé avec le titre de chaque sous-menu, une interface intuitive qui ne fait appel qu'à l'encodeur rotatif ainsi qu'une sonnerie personnalisable et des messages programmable par l'utilisateur. Nous avons fait de notre mieux pour fournir des schémas clairs et consistants qui sont agréables à lire.