

Rapport TP3 MCSED

BÉNISTANT Raphaël

BOURLLOT Xavier

3IMACS-AE-C, Binôme 1
30 mai 2020

Table des matières

1	Introduction	2
2	Mise en place de l'éclairage	2
3	Mise en place du réglage de l'alarme	2
4	Mise en place du chronomètre	3
4.1	Version 1	3
4.2	Version 2	5
5	Mise en place du minuteur	6
5.1	Version 1	6
5.2	Version 2	8
6	Conclusion	9
A	Annexes	9
A.1	Vidéos du montage en fonctionnement	9

1 Introduction

L'objectif de ce TP est d'effectuer la commande d'une montre digitale à l'aide d'un PC sous noyau temps-réel. Pour ce faire, nous traiterons dans un premier temps la mise en place de l'éclairage, puis, dans un deuxième temps, nous implémenterons la réglage de l'alarme, ensuite, dans un troisième temps, nous gèrerons la commande du chronomètre, et terminerons dans une dernière partie par la celle du minuteur. Toutes ces commandes seront directement codées sur Simulink sous forme de *statecharts* grâce à l'interface qu'il existe entre MATLAB et la maquette cible.

2 Mise en place de l'éclairage

La commande de l'éclairage est l'élément le plus simple à implémenter sur la maquette. En effet, le *statechart* correspondant n'est composé que d'un macro-état qui est lui-même décomposé en deux sous-état "lumière allumée" et "lumière éteinte". Le statechart implémenté sur MATLAB est le suivant :

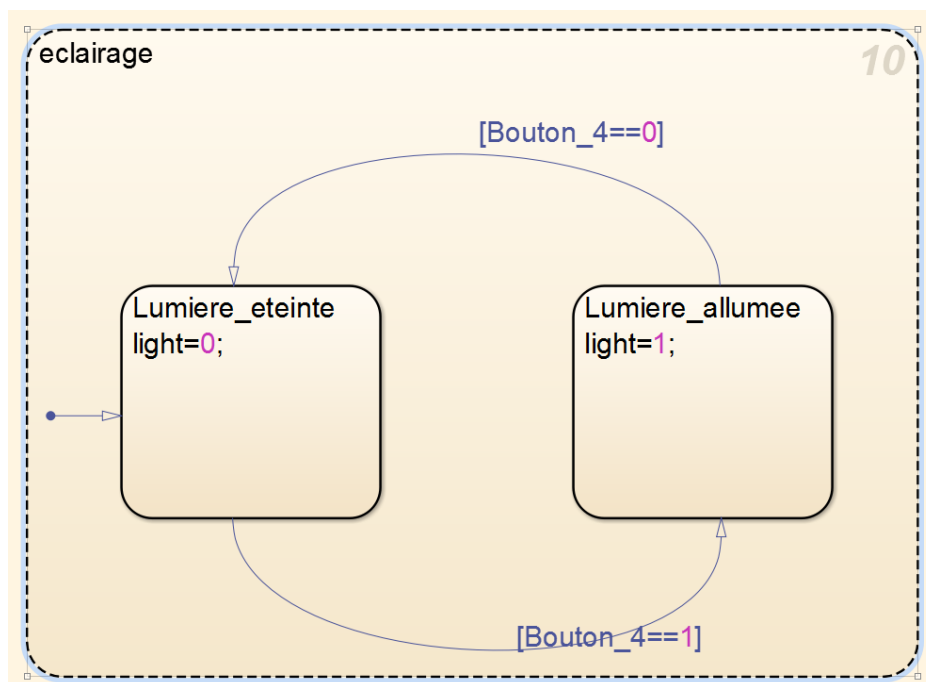


FIGURE 1 – Statechart pour la gestion de l'éclairage

avec

Sortie	Fonction
light	Allumage/extinction lumière
Bouton_4	Bouton d'éclairage

Quel que soit le mode dans lequel on se trouve, en appuyant sur le bouton 4 nous allumerons le voyant correspondant à la lumière.

3 Mise en place du réglage de l'alarme

Dans cette partie, nous nous occupons de l'implémentation du mode alarme de la montre digitale qui n'est pas encore utilisable sur la version initiale. Le *statechart* de cet étape est le suivant :

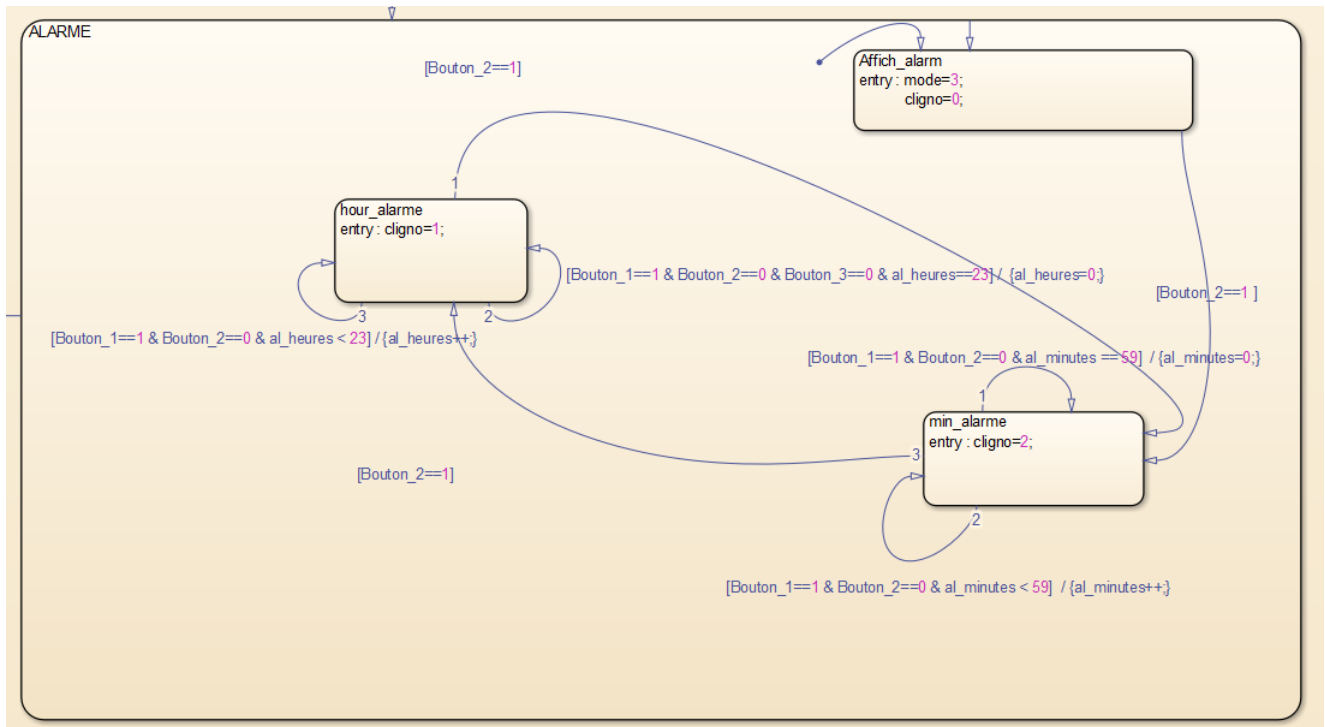


FIGURE 2 – Statechart du réglage de l'alarme

avec

Sortie	Fonction
mode=3	On est dans le mode alarme
Bouton_1	Bouton d'incrémentatation
Bouton_2	Bouton de réglage
cligno=0	Rien ne clignotte
cligno=1	La partie <i>Visu_1</i> clignotte
cligno=2	La partie <i>Visu_2</i> clignotte
al_heures	Heure(s) réglée pour l'alarme
al_minutes	Minute(s) réglée pour l'alarme

Lorsque l'on se trouve dans le mode date et que l'on appuie sur le bouton 3, on se retrouve dans le macro-état ci-dessus. Initialement, le sous-état dans lequel le système se trouve est le sous-état neutre *Affich_alarm* où l'on affiche la dernière heure réglée ou par défaut minuit ainsi que le jour et la date. lors de l'entrée dans cet état on met également à jours le voyant qui indique le mode dans lequel on se trouve.

Lors de l'appui sur le bouton 2, on passe dans le mode réglage de l'alarme, ce qui correspond en premier lieu sur le *statechart* au passage dans le sous-état *min_alarm* qui permet le réglage des minutes, une seconde pression sur le bouton 2 permet de régler les minutes, ce qui correspond au passage dans le sous-état *hour_alarm*.

Afin de coller le plus possible à ce qui a été codé dans la version initial, le seul moyen de sortie du réglage de l'alarme est de changer de mode jusqu'à retomber sur le mode alarme.

4 Mise en place du chronomètre

4.1 Version 1

A ce stade, l'alarme est entièrement configurée, passons maintenant à la mise en place du chronomètre. Pour pouvoir gérer le temps, nous avons créé quatre nouvelles variables : *minuteChrono*, *secondeChrono*, *centièmeChrono*, *runChrono*. Le *statechart* correspondant est le suivant :

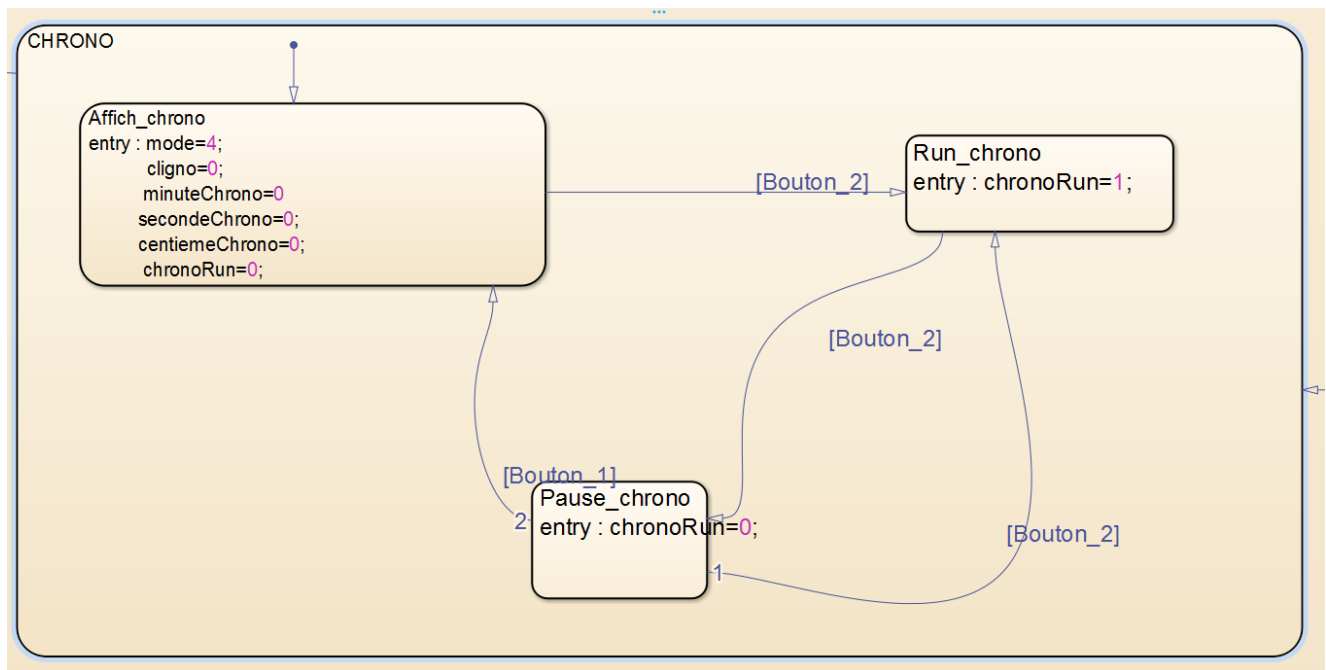


FIGURE 3 – Statechart du réglage du chronomètre V1

avec

Sortie	Fonction
mode=4	On est dans le mode chronomètre
Bouton_1	Bouton de réinitialisation
Bouton_2	Bouton de réglage
cligno=0	Rien ne clignotte
minuteChrono	Valeur courante des minutes du chronomètre
secondeChrono	Valeur courante des secondes du chronomètre
centièmeChrono	Valeur courante des centièmes du chronomètre
runChrono	Allumage/Extinction du chronomètre

Lorsque nous sommes dans le mode alarme et que nous exerçons une pression sur le bouton 3, nous entrons dans le macro-état *CHRONO* ci-dessus. Initialement, le système se trouve dans le sous-état de réinitialisation *Affich_chrono*. Cet état permet de paramétrer toutes les temps à 0 ainsi que de stopper le chronomètre. Lors d'une pression sur le bouton 2, le chronomètre est lancé, ce qui correspond au sous-état *Run_chrono*. Lors de l'exercice d'une nouvelle pression sur le bouton 2, le chronomètre est stoppé, ce qui correspond au sous-état *Pause_chrono*. Si l'utilisateur réappuie sur le bouton 2, on retourne dans le sous-état précédent, s'il appuie sur le bouton 1, il réinitialise le chronomètre.

Afin de gérer le temps, nous nous sommes inspiré du macro-état *Gest_horloge*. Une fois les quelques modifications apportées, le résultat est le suivant :

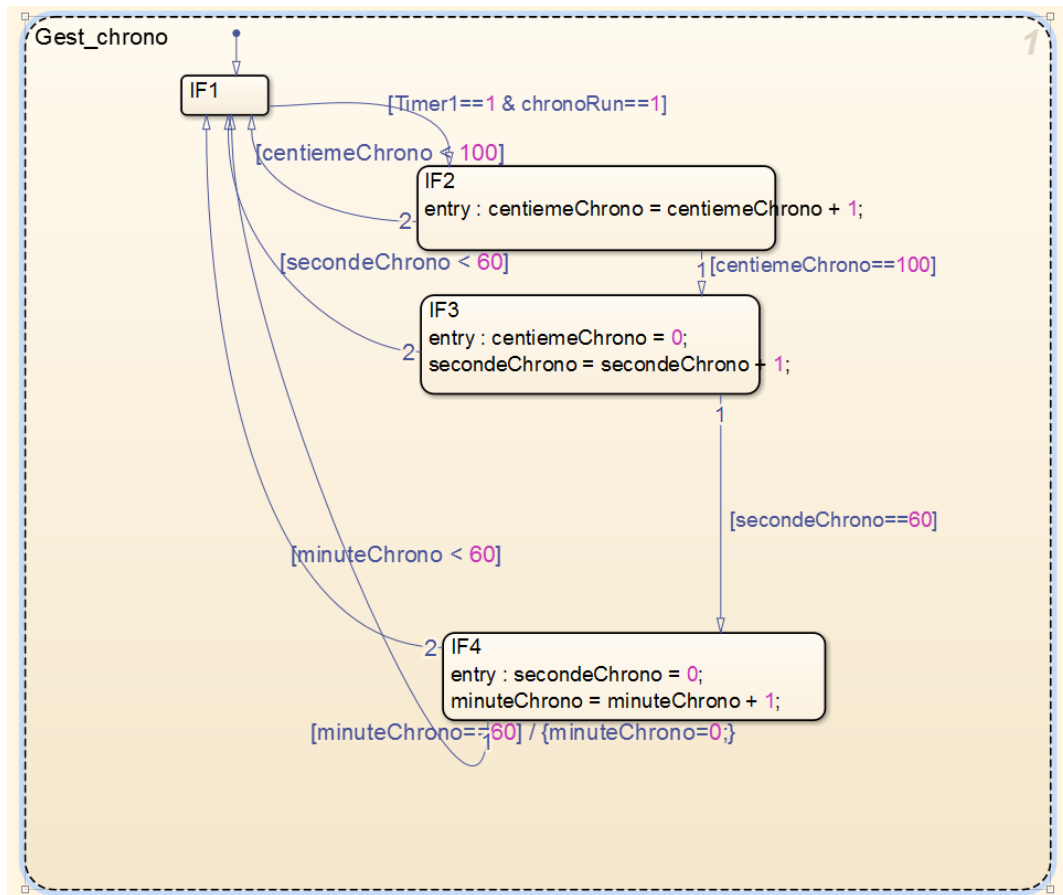


FIGURE 4 – Statechart de la gestion du temps du chronomètre

4.2 Version 2

Dans cette deuxième version du chronomètre, on ne désire plus stopper le chronomètre mais figer l’affichage de la valeur courante lors de l’appui sur le bouton 2. Pour ce faire, nous créons quatre nouvelles variables : *refresh*, *min_fige*, *sec_fige* et *cent_fige*. Le *statechart* est le suivant :

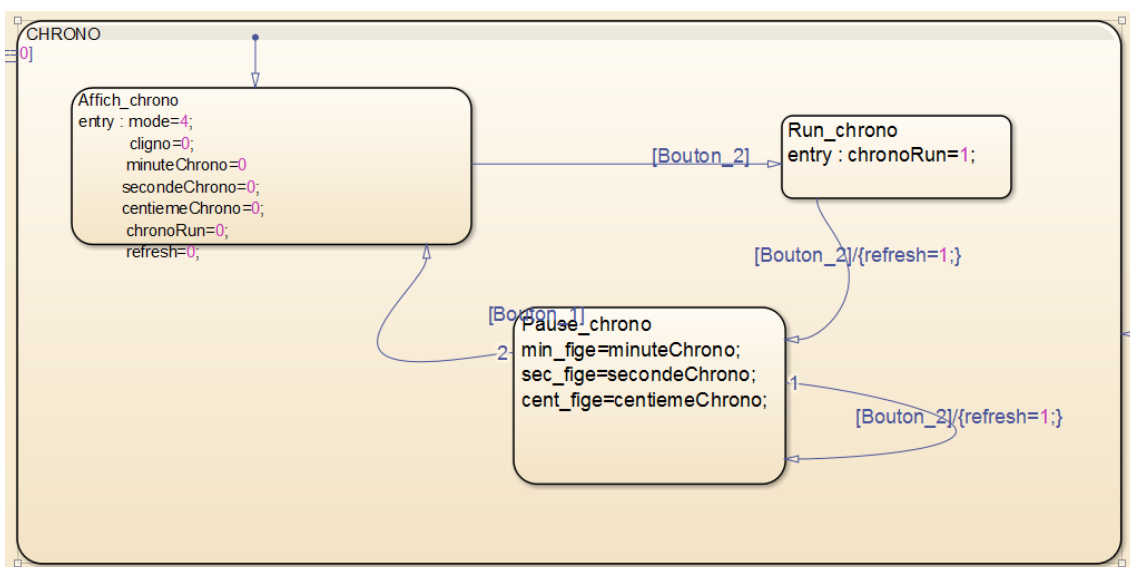


FIGURE 5 – Statechart du réglage du chronomètre V2

avec

Sortie	Fonction
refresh	Mise à jour de la valeur stockée
minute_fige	Valeur des minutes du temps stockée
sec_fige	Valeur des secondes du temps stockée
cent_fige	Valeur des centièmes du temps stockée

La différence par rapport à la première version est que le sous-état *Pause_chrono* ne stoppe plus le chronomètre mais il stocke la valeur courante dans les nouvelles variables de temps. Toute nouvelle pression sur le bouton 2 met à jour le temps figé grâce à la variable *refresh* qui déclenche un nouvel affichage. L'affichage est réglé en ajoutant à *GestionAffichage* un sous-état qui affiche la valeur figée à partir de la première pression sur le bouton 2 :

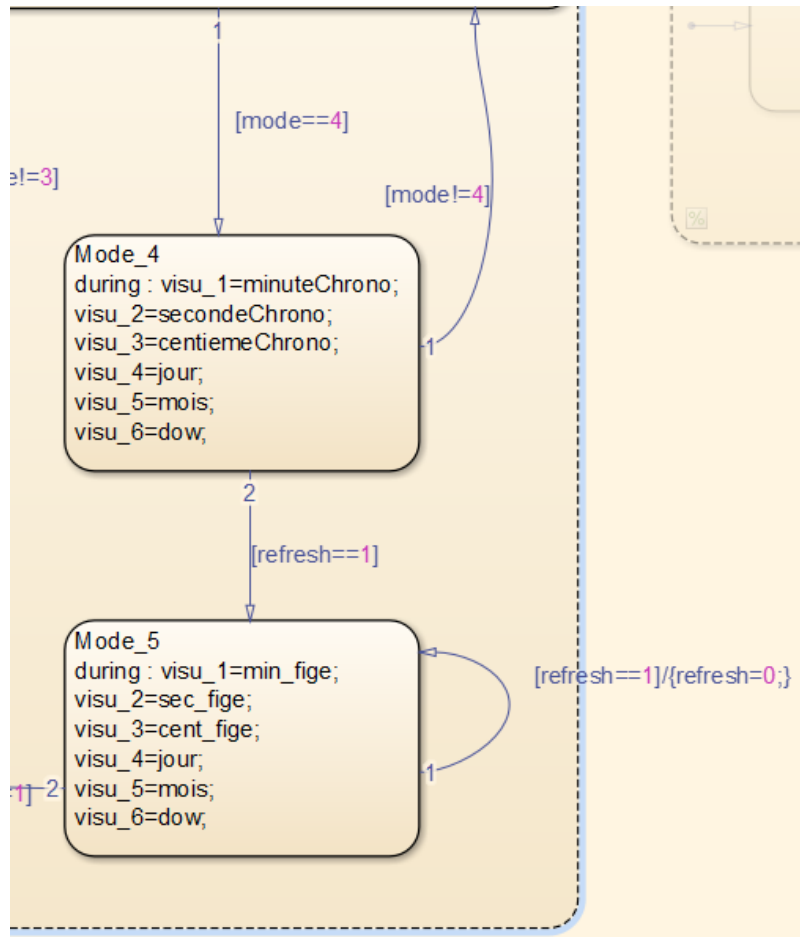


FIGURE 6 – Modification de GestionAffichage

5 Mise en place du minuteur

Ayant au préalable implémenté la fonction chronomètre, il nous manquait un voyant pour indiquer le mode minuteur. Nous avons choisi de supprimer la partie "éclairage", afin de pouvoir indiquer le mode minuteur par l'allumage simultané des voyants *light* et *mode = 4*.

5.1 Version 1

Le fonctionnement du minuteur est très proche de celui du chronomètre. Par souci d'économie de variables, et puisque les deux modes de fonctionnements sont mutuellement exclusifs, nous réutiliserons les variables de temps du chronomètre pour le minuteur.

Le minuteur est réinitialisé à l'entrée du menu. Un appui sur *Bouton_2* lance le décompte, pause et relance le minuteur. Le *Bouton_1* permet de réinitialiser le minuteur.

Une valeur initiale de 1 minute est fournie par défaut au compteur. Cette valeur sera ajustable dans la version 2. Le statechart obtenu est le suivant :

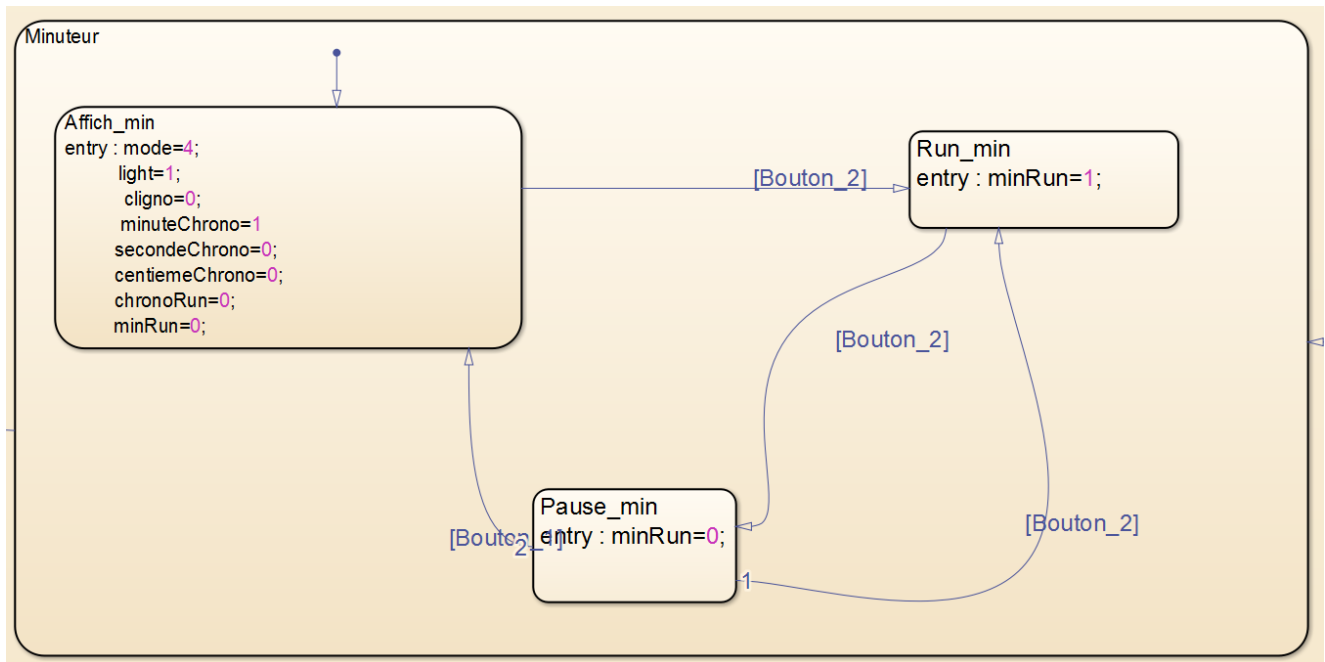


FIGURE 7 – Statechart du minuteur V1

Afin de gérer le temps, nous avons répété le macro-état *Gest_chrono*, mais cette fois avec un fonctionnement en décrémentation.

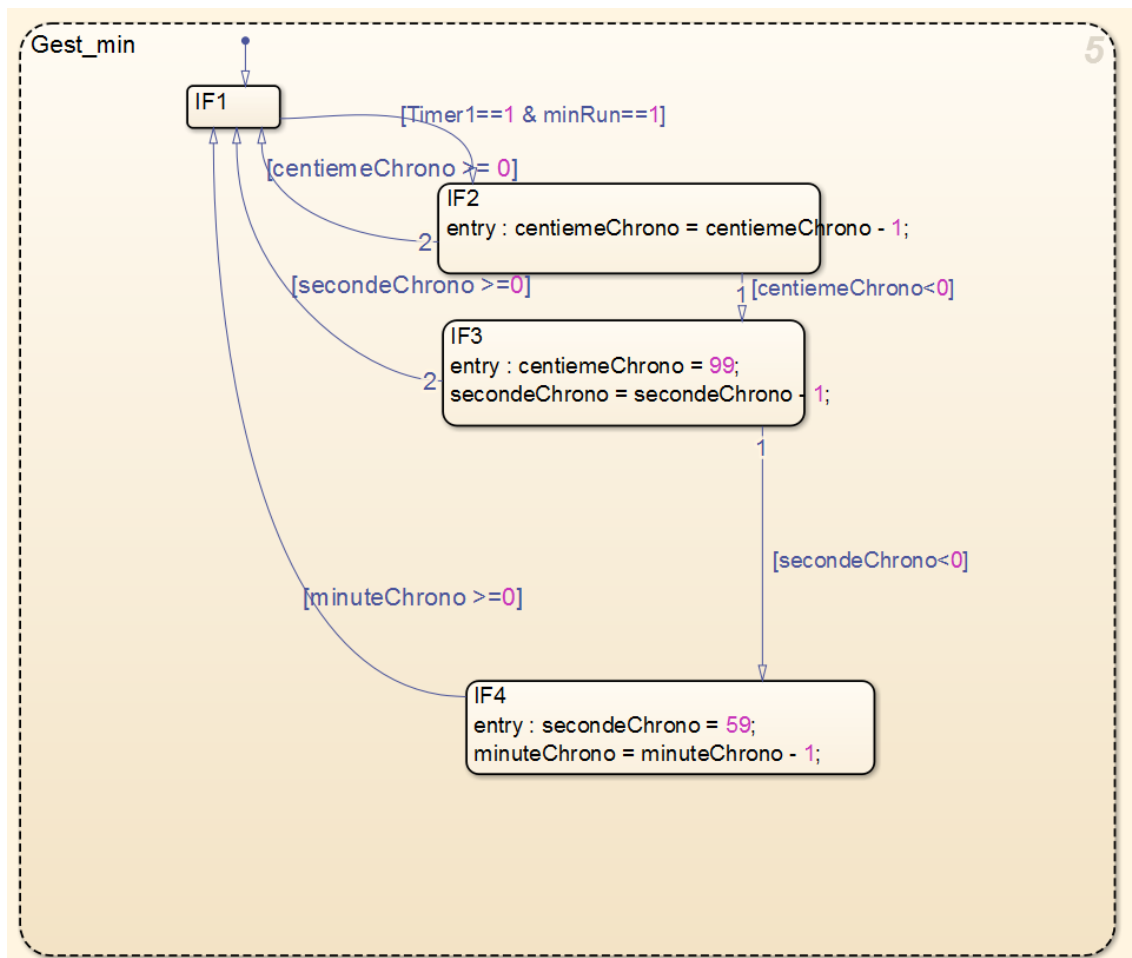


FIGURE 8 – Statechart du macro-état *Gest_Min* V1

Le déclenchement de l’alarme se fait sur un modèle identique à celui réalisé pour l’alarme. On déclenche le buzzer si le minuteur est actif et toutes les variables de temps sont à zéro.

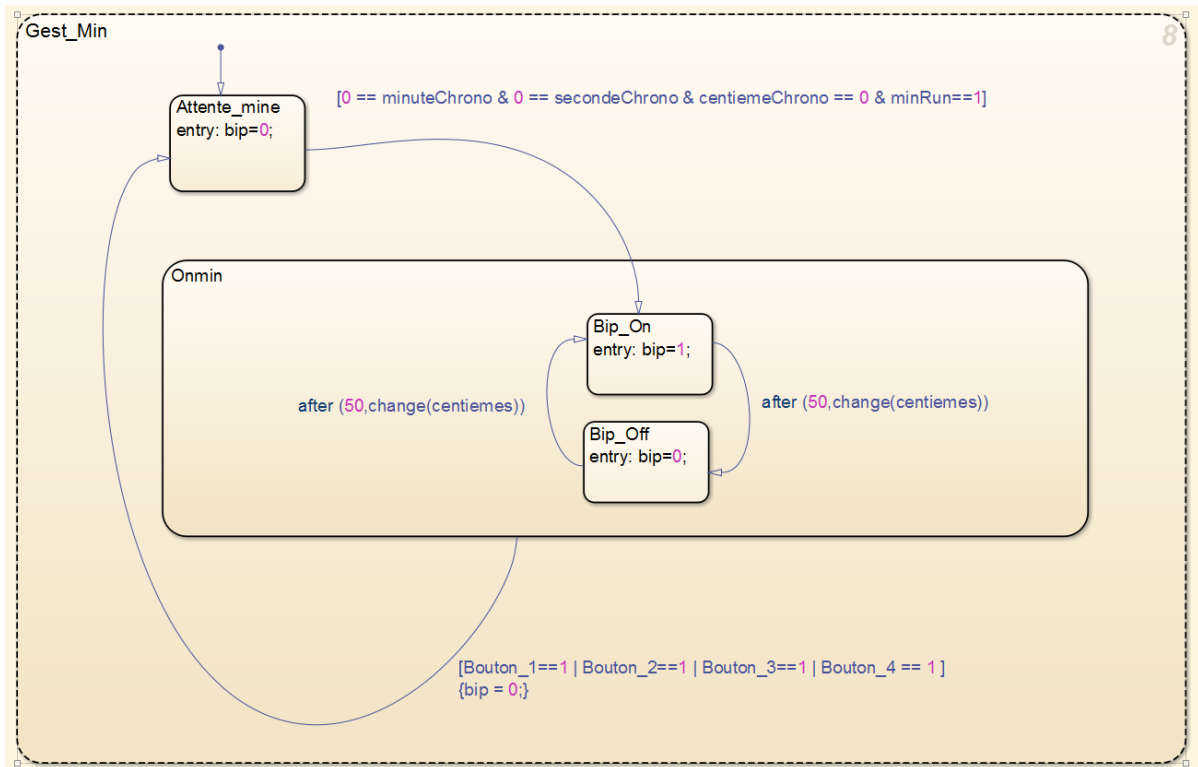


FIGURE 9 – Statechart de l’alarme minuteur V1

La sortie de l’alarme s’effectue par la pression d’un des boutons.

5.2 Version 2

Dans cette deuxième version du chronomètre, on désire pouvoir régler le temps à décompter. On s’inspire du statechart réalisé pour le réglage de l’alarme. Le statechart obtenu est le suivant :

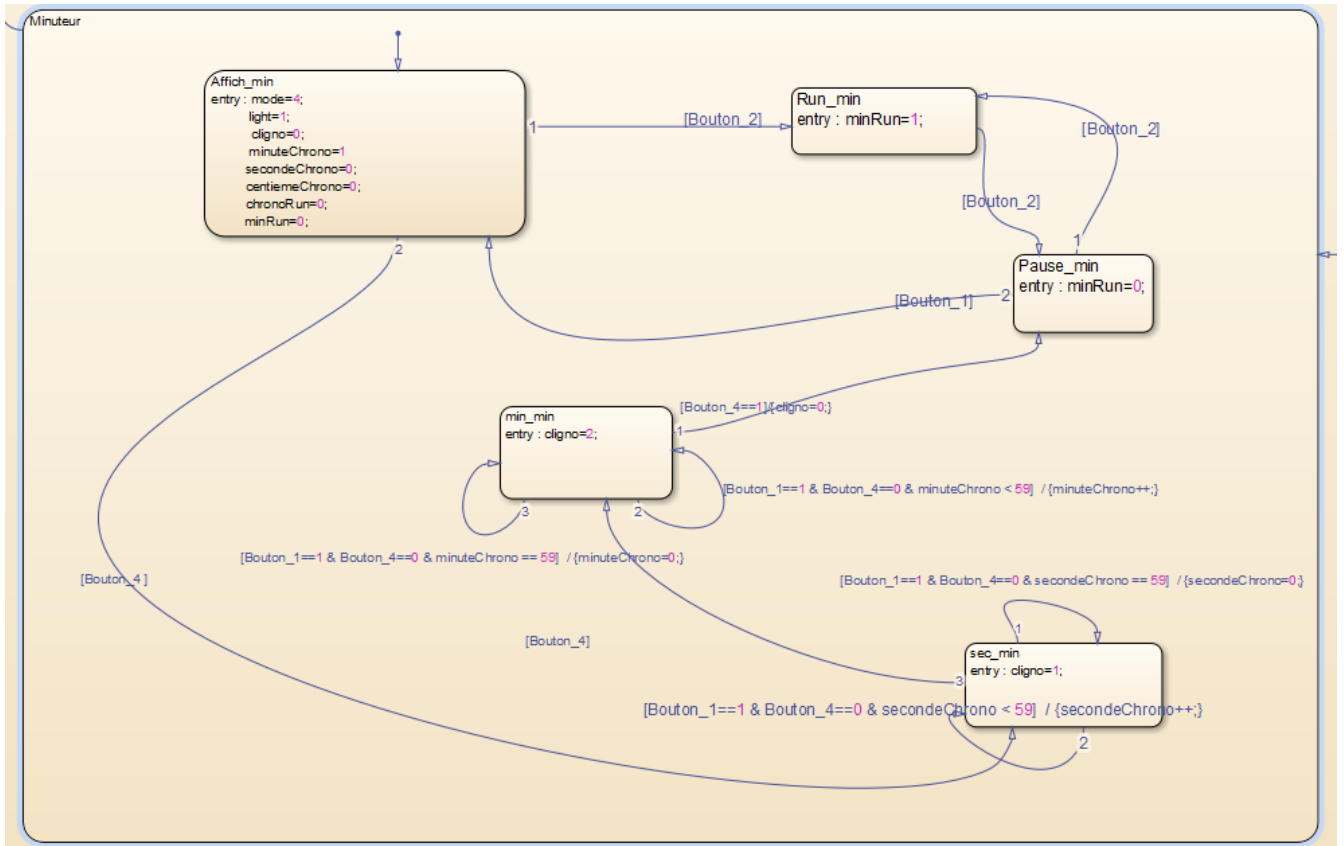


FIGURE 10 – Statechart du minuteur V2

On remarque qu’après avoir réglé les secondes et les minutes, le programme revient au menu de pause du minuteur.

Les autres statecharts restent inchangés.

6 Conclusion

Ainsi, nous avons pu implémenter plusieurs fonctions d’une montre digitale à l’aide de statecharts. La modélisation sous Simulink des blocs et leur enchaînement nous a permis de comprendre le caractère synchrone des évènements, ainsi que d’aborder les bases d’une interface utilisateur par menus imbriqués.

A Annexes

A.1 Vidéos du montage en fonctionnement

[Vidéos du montage en fonctionnement](#)