

# INF1500 : Logique des systèmes numériques

## Laboratoire 5

### Bloc heure/minuterie de cuisinière

Hiver 2016

---

#### Dates de remise par groupe de laboratoire

- **Groupe 1 (Lundi B1) : Lundi 4 Avril 2016, avant 23h55**
- **Groupe 2 (Lundi B2) : Lundi 11 Avril 2016, avant 23h55**

**A noter que 25% des points sont retranchés par jour de retard.**

### Introduction/objectif

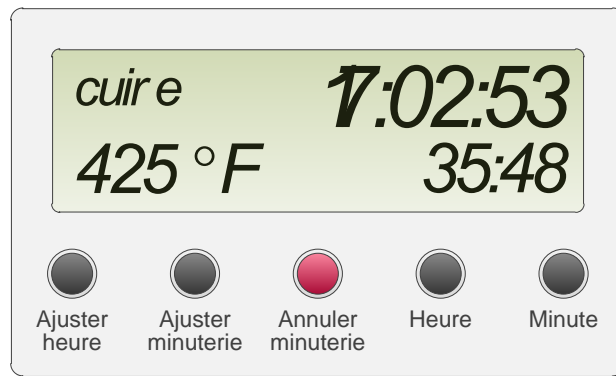
---

Dans le dernier TP de la session et après l'introduction au code VHDL dans le cadre du TP4, nous vous proposons maintenant un TP où vous n'aurez qu'à écrire du VHDL. Vous allez créer et tester un bloc heure/minuterie de cuisinière. Vous connaissez probablement déjà l'utilité d'un tel bloc en l'ayant utilisé vous-même : la partie heure permet de voir l'heure actuelle sur l'afficheur de la cuisinière et la minuterie permet de prévoir un arrêt automatique du four après un délai paramétrable. Aujourd'hui, la majorité des blocs heure/minuterie sont numériques.

### Contexte

---

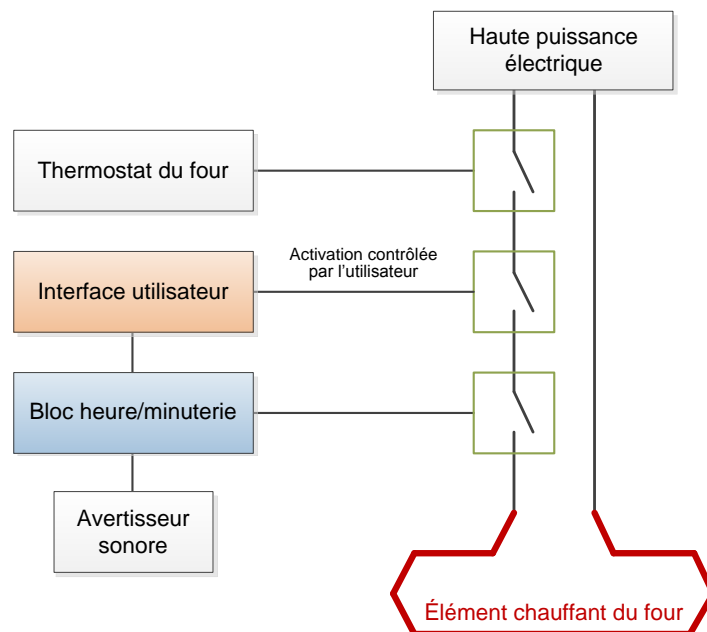
On vous demande pour ce TP d'implémenter un bloc heure/minuterie de cuisinière. L'interface envisagée concernant ce bloc sur la cuisinière ressemble à ceci :



L'afficheur LCD montre, en haut à droite, l'heure actuelle (heures au format 24h, minutes et secondes). Juste en-dessous, il indique le temps restant avant l'échéance de la minuterie (minutes et secondes). L'afficheur montre aussi, à gauche, le mode du four et sa température actuelle, mais les contrôles qui concernent cette partie ne sont pas pertinents au bloc heure/minuterie.

Sous l'afficheur, on voit cinq boutons permettant d'ajuster l'heure ou la minuterie et d'annuler cette dernière. Les boutons « Ajuster l'heure » et « Ajuster minuterie » peuvent également être rétroéclairés en contexte d'ajustement de l'heure ou de la minuterie respectivement.

Globalement, le bloc heure/minuterie a un contrôle sur l'activation de l'élément chauffant du four. Le thermostat et l'utilisateur en tant que tel (avec une partie de l'interface qu'on ne voit pas sur l'image ci-dessus) ont également leur mot à dire quant à l'activation de l'élément.



Lorsque la minuterie est désactivée ou lorsqu'elle est activée sans être à échéance, l'interrupteur permettant au courant de passer dans l'élément chauffant est fermé (pour activer l'élément). Ce n'est que lorsque la minuterie est activée et rendue à échéance que le bloc heure/minuterie coupe l'élément chauffant afin que les aliments dans le four cessent de cuire.

## Guide d'utilisation du bloc heure/minute

Suivez cette sous-section pour comprendre les différentes fonctionnalités du bloc heure/minute de la cuisinière.

- **Ajustement de l'heure.** Pour ajuster l'heure, la minuterie doit ni être activée, ni en cours d'ajustement.
  1. Appuyez sur le bouton « Ajuster heure ». Le bouton devient rétroéclairé. L'heure actuelle cesse d'être mise à jour automatiquement.
  2. Appuyez sur le bouton « Heure » autant de fois que nécessaire pour fixer les heures de l'heure. La partie des heures de l'heure actuelle est incrémentée à chaque appui. Si les heures valent 23, elles retournent à 0 au prochain appui.
  3. Appuyez sur le bouton « Minute » autant de fois que nécessaire pour fixer les minutes de l'heure. La partie des minutes de l'heure actuelle est incrémentée à chaque appui. Si les minutes valent 59, elles retournent à 0 au prochain appui.
  4. Appuyez de nouveau sur le bouton « Ajuster l'heure » pour quitter le mode d'ajustement de l'heure. L'heure actuelle recommence à être mise à jour automatiquement. Le bouton cesse d'être rétroéclairé.
- **Ajustement et activation de la minuterie.** Pour ajuster la minuterie, la minuterie ne doit pas être activée. L'heure actuelle ne doit pas non plus être en cours d'ajustement.
  1. Appuyez sur le bouton « Ajuster minuterie ». Le bouton devient rétroéclairé.
  2. Appuyez sur le bouton « Minute » autant de fois que nécessaire pour fixer les minutes de la minuterie. La partie des minutes de la minuterie est incrémentée à chaque appui. Si les minutes valent 90, elles retournent à 0 au prochain appui.
  3. Optionnellement, appuyez n'importe quand sur le bouton « Annuler minuterie » pour annuler l'ajustement de la minuterie. La minuterie affiche 0 heures et 0 minutes. Le bouton cesse d'être rétroéclairé.
  4. Appuyez de nouveau sur le bouton « Ajuster minuterie » pour activer la minuterie. Le bouton cesse d'être rétroéclairé. La minuterie commence à décrémente le temps affiché chaque seconde.
- **Minuterie.** Une fois la minuterie activée en suivant le processus détaillé ci-haut, les événements suivants sont possibles :
  - Vous appuyez sur le bouton « Annuler minuterie ». La minuterie est alors annulée : ses parties heures et minutes retournent à 0 et le four continue de fonctionner.
  - La minuterie atteint 0 heure et 0 minute. Un avertissement sonore répétitif se fait entendre. L'élément chauffant du four est coupé. L'utilisateur doit appuyer sur le bouton « Annuler minuterie » afin d'arrêter l'avertissement sonore et rétablir le courant dans l'élément chauffant du four.

## Travail à effectuer

Pour ce TP, **tous** les fichiers VHDL et le fichier schématique de plus haut niveau sont fournis. Certains sont complets, d'autres partiels (à compléter). Vous trouverez ces fichiers dans l'archive fournie sous le lien vers cet énoncé sur Moodle.

Vous ne devez **absolument pas** modifier les noms des blocs ou de leurs ports (entrées/sorties).

Créez un nouveau projet Active-HDL et ajoutez tous les fichiers fournis à celui-ci (en vous assurant de cocher la case permettant de créer des copies). Vous n'avez pas à créer d'autres fichiers. Le fichier de contraintes UCF est également fourni au complet.

Voici un tableau résumant, pour tous les blocs fournis, leurs noms, leurs descriptions et s'ils sont complets ou partiels.

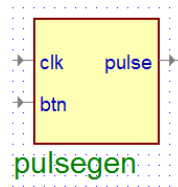
Nom du bloc	Description	Complet ou partiel
asciizero2space	Convertisseur du caractère 0 au caractère d'espacement. Ce bloc sert à éliminer la décimale des dizaines si elle est 0.	Complet
bcd2ascii	Convertisseur BCD (décimale codée sur 4 bits) vers son caractère ASCII.	Complet
bin2bcd2d	Convertisseur binaire vers BCD à 2 décimales.	Complet
buzzer	Animateur de DEL pour signifier un avertissement (remplace l'avertisseur sonore qui n'est pas présent sur la carte Genesys).	Complet
clk1hz	Générateur d'une impulsion d'horloge à chaque seconde.	Complet
colongen	Générateur du caractère ASCII :.	Complet
deb	Antirebond pour boutons-poussoirs.	Complet
hd44780_8chars_2lines	Bloc simplifiant l'utilisation de <code>hd44780_simple</code> pour 2 lignes de 8 caractères sur l'afficheur LCD.	Complet
hd44780_simple	Bloc gérant la communication avec l'afficheur LCD.	Complet
oventimetimerfsm	Machine à états finis de bloc heure/minuterie.	Partiel
pulsegen	Générateur d'impulsion d'un cycle à partir d'un signal sans rebond venant d'un bouton-poussoir.	Partiel
spacegen	Générateur du caractère d'espacement ASCII.	Complet
timekeeper	Bloc heure (gardien du temps).	Partiel
timer	Bloc minuterie.	Partiel
top	Schéma de haut-niveau à tester sur la carte Genesys.	Complet

Les blocs partiels sont à compléter pour ce TP. Ils sont détaillés dans les sous-sections subséquentes.

Bien que les blocs soient assez simples lorsqu'étudiés un par un, vous n'avez pas à décortiquer les blocs fournis au complet. Ils existent principalement pour que votre projet puisse être testé sur la carte Genesys. Les modules les plus importants pour l'implémentation du bloc heure/minuterie d'une cuisinière sont `timekeeper`, `timer` et `oventimetimerfsm`.

## Générateur d'impulsion (pulsegen)

Le bloc `pulsegen` permet de générer une impulsion lorsqu'un signal de bouton-poussoir est détecté. Il existe pour simplifier la machine à états `oventimetimerfsm`, qui vérifie les entrées à chaque cycle, puisque le fait d'appuyer humainement sur un bouton-poussoir crée nécessairement un signal vrai sur plusieurs millions de cycles (comme l'horloge du système à une fréquence de 100 MHz).



Le générateur est très simple. Chaque cycle de l'horloge `clk` en entrée, il doit vérifier si l'entrée `btn` vaut 1. Si c'est le cas, il doit générer une impulsion (le signal de sortie `pulse` doit valoir 1 pendant **un seul cycle**, sinon 0).

Ce bloc doit être implémenté en tant qu'une petite machine à états finis.

Voici ce que vous devez effectuer et remettre concernant ce bloc :

1. Spécifiez (dans le rapport) si vous avez créé une machine à états finis de Moore ou de Mealy.
2. Dessinez le diagramme d'états de ce bloc. Précisez bien l'état initial, le nom des états, les conditions de transition et les sorties (pour chaque état si vous créez une machine à états finis de Moore, ou pour chaque paire entrée/état si vous créez une machine à états finis de Mealy).
3. Créez le tableau d'états pour ce bloc. On doit y voir l'entrée `btn`, l'état en cours, le prochain état et la sortie `pulse`.
4. Complétez le code VHDL nécessaire pour implémenter ce bloc dans le fichier `pulsegen.vhd`.

Notez qu'en VHDL, tous les signaux (`signal`) sont mis à jour à la fin de l'évaluation d'un processus (`process`). Ceci signifie que vous ne pouvez pas, dans la même évaluation, mettre un signal à jour et vous y fier. Par exemple, en supposant que `sig` vaut 110 avant l'évaluation du processus suivant :

```
sig <= "010";
autre_sig <= sig;
```

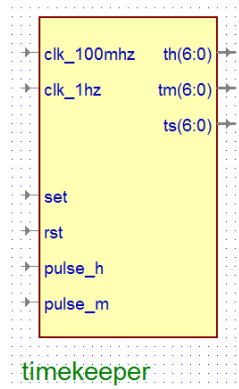
Ici, `autre_sig` vaudra 110, et non 010, à la fin de l'évaluation, parce que `sig` n'est mis à jour à 010 qu'à la fin du processus. Ceci permet de réaliser des opérations intéressantes, comme une rotation, sans avoir recours à des signaux temporaires :

```
mon_signal(7 downto 1) <= mon_signal(6 downto 0);
mon_signal(0) <= mon_signal(7);
```

Rappelez-vous de ce concept si votre bloc séquentiel se comporte étrangement.

## Gardien du temps (timekeeper)

Le bloc qui garde le temps, ou l'heure, met en sortie les valeurs en cours des heures (**th**), des minutes (**tm**) et des secondes (**ts**). En entrée, il prend une horloge rapide **clk** (100 MHz) et une horloge à 1 Hz **clk\_1hz** pour incrémenter le temps.



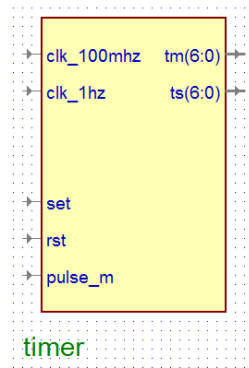
On peut aussi voir les entrées **set**, **rst**, **pulse\_h** et **pulse\_m**. Voici leurs rôles :

- **rst** : réinitialisation asynchrone qui remet le temps à 0 heure, 0 minute et 0 seconde.
- **set** : lorsque ce signal vaut 1, la mise à jour automatique du temps est arrêtée et le module écoute les entrées **pulse\_h** et **pulse\_m** pour modifier l'heure actuelle. Lorsqu'il retourne à 0, l'heure actuelle continue d'être mise à jour à partir de sa nouvelle valeur. Les secondes ne sont pas touchées pendant l'ajustement de l'heure.
- **pulse\_h** et **pulse\_m** : incrémentent respectivement les heures et les minutes de l'heure actuelle lorsque **set** vaut 1. La partie « pulse » des noms ici signifie que ces entrées valent 1 pendant un seul cycle pour chaque incrémentation désirée.

Les sorties **th**, **tm** et **ts** sont toutes sur 7 bits pour faciliter l'encodage BCD à l'extérieur du bloc (déjà implémenté pour vous). Notez que, comme l'entrée **clk\_1hz** provient de la sortie d'un bloc **clk1hz**, il s'agit en fait d'une impulsion d'un cycle à chaque seconde (et non d'une onde parfaitement carrée).

## Minuterie (timer)

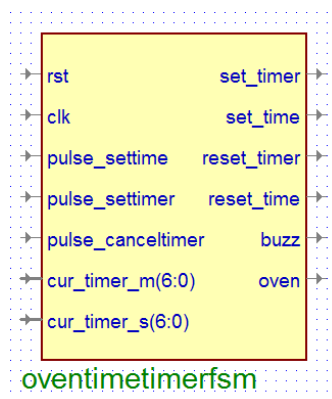
La minuterie ressemble fonctionnellement au gardien du temps, sauf qu'elle décrémente le temps au lieu de l'incrémenter. De plus, elle ne met en sortie qu'un nombre de minutes **tm** (de 0 à 90 inclusivement) et un nombre de secondes **ts**. L'entrée de réinitialisation asynchrone **rst** remet à 0 les minutes et les secondes. Tout comme le gardien du temps, tant que **set** vaut 1, le compte à rebours est arrêté et chaque impulsion de **pulse\_m** augmente la valeur des minutes.



**Important** : lorsque les minutes et les secondes atteignent 0, elles **doivent rester à 0** tant que la minuterie n'est pas réajustée.

### Machine à états finis du bloc heure/minuterie (`oventimtimerfsm`)

Ceci représente le plus gros bloc du circuit total. Il s'agit de la machine à états qui, selon certaines entrées, contrôle l'heure, la minuterie, l'avertissement sonore et l'activation du four.



Pour implémenter cette machine à états finis, vous devez vous fier au guide d'utilisation de la cuisinière présenté plus haut dans ce document.

Voici une liste des entrées et sorties et leurs descriptions :

- **Entrées :**
  - `rst` : réinitialisation asynchrone (retourne à l'état initial).
  - `clk` : horloge rapide (100 MHz).
  - `pulse_settime` : impulsion d'un cycle provenant du bouton-poussoir « Ajuster heure ».
  - `pulse_settimer` : impulsion d'un cycle provenant du bouton-poussoir « Ajuster minuterie ».
  - `pulse_canceltimer` : impulsion d'un cycle provenant du bouton-poussoir « Annuler minuterie ».
  - `cur_timer_m` : valeur en cours des minutes de la minuterie.
  - `cur_timer_s` : valeur en cours des secondes de la minuterie.
- **Sorties :**
  - `set_timer` : signal envoyé directement à l'entrée `set` de la minuterie.

- `set_time` : signal envoyé directement à l'entrée `set` du gardien du temps.
- `reset_timer` : signal envoyé directement à l'entrée `rst` de la minuterie.
- `reset_time` : signal envoyé directement à l'entrée `rst` du gardien du temps.
- `buzz` : 1 pour activer l'avertissement sonore.
- `oven` : 1 pour activer l'élément chauffant du four.

Les sorties `set_timer`, `set_time`, `buzz` et `oven` sont également envoyées vers des DEL de la carte Genesys lors de l'implémentation, ce qui vous permettra de voir leurs valeurs en testant physiquement le circuit.

La machine à état ne s'occupe pas des signaux `pulse_m` et `pulse_h` : ceux-ci sont envoyés directement au gardien de temps et à la minuterie puisque, de toute façon, ces modules ne s'en préoccupent pas tant que leur entrée `set` ne vaut pas 1 (qui elle est contrôlée par la machine à états finis).

Voici ce que vous devez effectuer et remettre concernant ce bloc :

1. Spécifiez (dans le rapport) si vous avez créé une machine à états finis de Moore ou de Mealy.
2. Dessinez le diagramme d'états de ce bloc. Précisez bien l'état initial, le nom des états, les conditions de transition et les sorties (pour chaque état si vous créez une machine à états finis de Moore, ou pour chaque paire entrée/état si vous créez une machine à états finis de Mealy).
3. Créez le tableau d'états pour ce bloc. On doit y voir toutes les entrées du bloc (sauf `clk` et `rst`) et toutes ses sorties.
4. Complétez le code VHDL nécessaire pour implémenter ce bloc dans le fichier `oventimetimerfsm.vhd`.

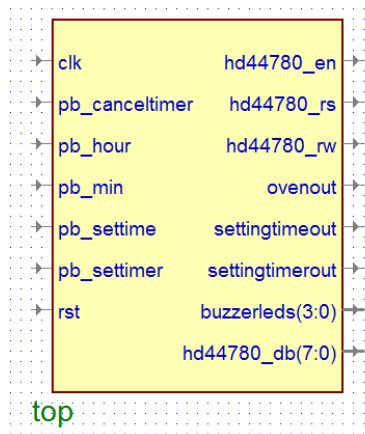
**Remarque** : l'auteur de cet énoncé n'a pas eu besoin de plus de 7 états pour réaliser ce module complètement.

## Implémentation sur la carte Genesys

Pour tester physiquement votre circuit, assurez-vous d'avoir complétés et compilés avec succès tous les blocs. Puis, dans les paramètres de synthèse, mettez comme *top level* le bloc `top` et incluez tous les blocs dans la synthèse.

Le bloc `top` ressemble à ceci de l'extérieur :





N'oubliez pas de spécifier, dans les paramètres de l'implémentation, le chemin vers le fichier de contraintes UCF fourni (que vous devriez copier dans le répertoire de votre projet). Ce fichier de contraintes assigne les entrées/sorties suivantes sur la carte :

Nom du signal	Description	Contrainte d'emplacement
pb_canceltimer	Bouton « Annuler minuterie »	BTN1
pb_hour	Bouton « Heure »	Mini-joystick vers la gauche
pb_min	Bouton « Minute »	Mini-joystick vers la droite
pb_settime	Bouton « Ajuster heure »	Mini-joystick vers le haut
pb_settimer	Bouton « Ajuster minuterie »	Mini-joystick vers le bas
rst	Réinitialisation asynchrone	BTN0
hd44780_*	Affichage LCD	Afficheur LCD (heure actuelle en haut et minuterie en bas)
ovenout	Activation de l'élément chauffant du four	LD3
settingtimeout	En cours d'ajustement de l'heure actuelle	LD0
settingtimerout	En cours d'ajustement de la minuterie	LD1
buzzerleds	Sortie lumineuse pour simuler ce qui devrait être l'avertissement sonore	Animation sur LD7 à LD4

Avant toute opération, vous devriez appuyer sur BTN0 pour réinitialiser tous les sous-circuits qui le demandent. Par la suite, vous pouvez tester toutes les fonctionnalités sans réinitialiser.

## Rapport PDF

Le rapport PDF doit contenir :

- une page titre avec le sigle et le titre du cours, les noms et matricules des deux coéquipiers, la section du cours, la date de remise du laboratoire et le nom de l'École;
- une courte introduction expliquant le but du laboratoire réalisé, sans copier/coller cet énoncé;
- les diagrammes d'états, tableaux d'états et spécifications Moore/Mealy tels que demandés pour les blocs `pulse_gen` et `oventimetimer_fsm`;

- les captures d'écran, bien identifiées et **commentées** (i.e. pourquoi le résultat montré est-il bon?) des simulations des blocs suivants :
  - pulsegen;
  - timekeeper;
  - timer;
- une conclusion résumant ce que vous avez appris grâce à ce laboratoire et proposant une amélioration intéressante au module réalisé.

## Barème

Ce travail pratique totalise **7 points** du cours INF1500. Les 7 points seront partagés ainsi :

Contenu	Point(s) sur 7
Rapport : spécifications Moore/Mealy, schémas et diagrammes d'états des blocs pulsegen et oventimetimerfsm	2
Rapport : simulations	1
Reste du rapport	1
Sources VHDL des circuits pulsegen, timekeeper, timer et oventimetimerfsm	2
Fonctionnement de l'implémentation (top)	1