julien.braine@ens-lyon.fr,mateusz.skomra@ens-lyon.fr, florent.de-dinechin@insa-lyon.fr

ASR1, TD/TP : circuits séquentiels et automates

Oh non, encore un TP sous Digital. Mais parfois il faut réfléchir sur papier.

Exercice 1: Compteur :

1. D'abord il faut un registre 8 bits avec reset. Si vous voulez recycler votre registre à vous, c'est encouragé.
   Sinon, allez chercher un composant flip-flop avec reset (synchrone ou asynchrone) dans la bibliothèque de Digital. Faites-en un registre 8 bits en modifiant son paramètre "data bits". Dessinez sur un bout de papier comment ce registre 8 bits serait construit à partir de flip-flops 1 bit, et faites valider par un enseignant. En particulier, précisez combien de bits font désormais les différentes entrées et sorties.
2. Allez chercher l'additionneur de bibliothèque et mettez aussi son "data bits" à 8.
3. Avec tout cela, construisez un compteur 8 bits. Vous aurez aussi besoin du composant "constant value" dans Wires.

Exercice 2: Compteur cracra : Construisez en Digital un compteur 4 bits en assemblant uniquement des Flip-Flop. (Ce n'est pas un circuit séquentiel synchrone selon la définition vue en cours, que vous en profiterez pour rappeler).

Pour vous convaincre que cette version de hackerz du compteur n'a pas que des avantages, voici quelques variations à appliquer sur les deux versions du compteur (dessinez-les en papier seulement). Elles sont en principe faciles en partant du compteur de l'exercice 1. Elles seront moins facile avec le compteur de l'exercice 2.

— un décompteur
— un compteur/décompteur avec une entrée qui dit s'il faut compter ou décompter
— un compteur jusqu'à 60 (avec remise à 0 à 60) pour mettre dans votre montre à quartz

Effacez désormais de votre esprit les circuits séquentiels non synchrone. Je ne veux pas en voir à l'examen.

Exercice 3: Considérer un circuit séquentiel comme un automate :

Quels sont les états possibles d'un compteur?

Sur une feuille de papier, dessinez l'automate correspondant.

Donnez un encodage des états de cet automate tel que la construction mécanique d'un automate synchrone retombe sur le compteur de la première question. (cet exercice tourne un peu en rond mais c'est exprès).

Exercice 4: Un circuit séquentiel qui calcule la suite de Fibonacci :

Proposez un circuit séquentiel pour calculer les termes de la suite de Fibonacci : $u_0 = 0$, $u_1 = 1$ et $u_i = u_{i-1} + u_{i-2}$ pour $i \geq 2$. Le circuit sera remis à zéro à l'aide d'un bouton `reset` : quand `reset` = 1 sur un front montant de l'horloge, le circuit est réinitialise, et la sortie produite est $u_2 = 1$. On suppose ensuite que l'on remet `reset` à 0, et à chaque cycle d'horloge le circuit produit un nouveau terme de la suite. Vous utiliserez des registres et un additionneur 16 bits. Faites valider par un enseignant, ensuite implémentez-le dans Digital (en utilisant l'additionneur de la bibliothèque "Arithmetic" et un flip-flop bien choisi de "Flip-Flops").

Exercice 5: Un multiplieur séquentiel :

En C par exemple, on peut implanter de la façon suivante l'algorithme de multiplication de la petite école pour multiplier deux entiers naturels `a_init` et `b_init` sur 8 bits, et obtenir le résultat sur 16 bits.

```
uint16_t mul(uint8_t a_init, uint8_t b_init) {
  uint16_t ax = a_init;
  uint8_t   b = b_init;
  uint16_t  c = 0;

  while(b != 0) {
    if(b & 0x01) c += ax; // b & 0x01 est le bit de poids faible de b
    ax <<= 1;             // décalage d'un bit à gauche
    b  >>= 1;             // décalage d'un bit à droite
  }
  return c;
}
```

Le but est d'implanter cet algorithme sous la forme d'un circuit séquentiel dans Digital. Votre circuit prendra en entrée les entiers naturels `a_init` et `b_init` sur 8 bits, un signal `run` et un signal d'horloge. Il produira en sortie la valeur « courante » de `c`, ainsi qu'un signal `finished`, initialement à 0, et qui passera à 1 dès que le calcul est terminé.