

ASR1 – TD3 : Mémoires

{ Jeremie.Detrey, Patrick.Loiseau, Nicolas.Veyrat-Charvillon }@ens-lyon.fr
http://perso.ens-lyon.fr/jeremie.detrey/06_asr1/
9 et 10 octobre 2006

1 Construisons un registre

On se place ici en technologie CMOS, qui nous permet de construire uniquement des portes non-et (NAND) et des portes non-ou (NOR) avec un nombre quelconque d'entrées – la porte non (NOT) étant un cas particulier de ces portes.

1. Quelle est la structure la plus simple permettant de “se souvenir” d'un état? Il n'est pas demandé que l'on puisse initialiser l'état; il suffit que la structure soit capable de se souvenir de l'importe quel état (0 ou 1).
2. Tant que vous y êtes, construisez un oscillateur.
3. Sachant que l'on s'interdit de faire des court-circuits et autres cochonneries du même acabit, dessinez une bascule RS. Une bascule RS (ou *RS latch*) est une structure à deux entrées (R et S) et deux sorties (Q et \overline{Q}). La sortie Q garde l'état courant quand R et S sont à 0, passe à 0 lorsque R (*reset*) vaut 1, et passe à 1 lorsque S (*set*) vaut 1. (On suppose que R et S ne sont jamais simultanément à 1.)
4. Réutilisez la bascule RS pour définir un registre sur état (aussi appelée bascule D, ou encore *D latch*): une structure à deux entrées (D et Clk) et deux sorties (Q et \overline{Q}) qui recopie D (*data*) sur Q lorsque Clk (*clock*, l'horloge) vaut 1, et qui garde l'état courant lorsque Clk vaut 0.
5. Réutilisez la bascule D pour définir un registre sur front (aussi appelé *D flip-flop*): une structure à deux entrées (D et Clk) et deux sorties (Q et \overline{Q}) qui recopie D sur Q lorsque Clk passe de 0 à 1 (*front montant*), et qui garde l'état courant le reste du temps.
6. Philosophiez sur l'intérêt fondamental du registre sur front.
7. Dessinez un registre “débrayable”: rajoutez une entrée CE telle que la recopie de D sur Q ne se fait que si CE (*clock enable*) est à 1 lors du front montant de Clk .

2 Compteurs

1. Dessinez un circuit qui compte modulo 2^n .
2. Étendez-le en un compteur/décompteur. (Rajoutez une entrée qui dit si on compte ou si on décompte.)
3. Rajoutez une entrée qui désactive le comptage/décomptage.
4. Rajoutez une entrée de remise à zéro (*reset*) synchrone.
5. Rajoutez la possibilité de précharger le compteur/décompteur à une valeur donnée en entrée.

3 Mémoire adressable

On dit qu'une mémoire adressable est de type $2^k \times m$ si elle contient 2^k mots de m bits, adressés par une adresse sur k bits.

1. Construisez, en utilisant des registres et des portes logiques de base, une mémoire $2^1 \times 1$, une mémoire $2^2 \times 1$, une mémoire $2^k \times 1$, une mémoire $2^k \times m$.
2. En plus des portes logiques, on se donne la porte trois-états du cours. Mêmes questions. Discutez des avantages et des inconvénients.
3. Quelle est la différence entre espace d'adressage et taille mémoire ?
4. On dispose de boîtiers $32K \times 8$, combien y a-t-il de broches d'adresse et de broches de données ?
5. On veut fabriquer une mémoire 1M mots avec des mots de 32 bits avec ces boîtiers, comment fait-on ?
6. Dessinez une mémoire associative à 4 positions, à clefs sur 4 bits et à données sur 4 bits.