

Dans cette activité , on utilise le logiciel *logisim*

• Un registre:

1. Sélectionner dans la rubrique *memory* le composant **Register**.
2. Dans les options de **Register**, fixer la valeur de *Data Bits* à 2 (ce qui permet de stocker des données à 2 bits dans le registre)
3. Explorer (à l'aide de la souris) la signification des différentes entrées du registre (points noirs sur les bords).
4. Positionner ensuite quatre *Pin* et les connecter aux 4 points d'entrées .
5. Faire des tests afin de mieux comprendre le fonctionnement du dispositif.On placera un *Pin* en sortie également.
6. Modifier (ou le recopier avant modification) le circuit précédent afin de pouvoir lire et écrire dans le registre des données à 8 bits.

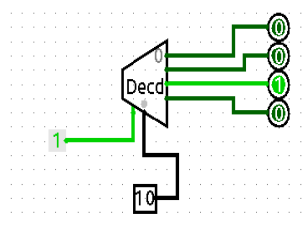
• Un décodeur:

Un décodeur "1 parmi 4" permet de sélectionner une sortie et une seule parmi 4 pour une entrée de 2 bits (plus généralement un décodeur permet de sélectionner une sortie parmi $2N$ pour une entrée de N bits).

On donne la table de vérité du décodeur "un parmi 4":

e1	e2	S1	S2	S3	S4
0	0	1	0	0	0
0	1	0	1	0	0
1	0	0	0	1	0
1	1	0	0	0	1

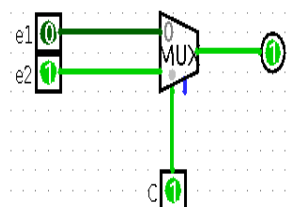
1. Donner une expression booléenne pour chacune des 4 sorties S1,S2,S3 et S4.
2. En déduire le schéma logique du décodeur et le réaliser avec *logisim*.
3. Le schéma suivant simplifie la représentation précédente en utilisant *Decoder* dans le menu *Plexers*:



Reproduire le circuit ci-dessus dans *logisim* et tester.

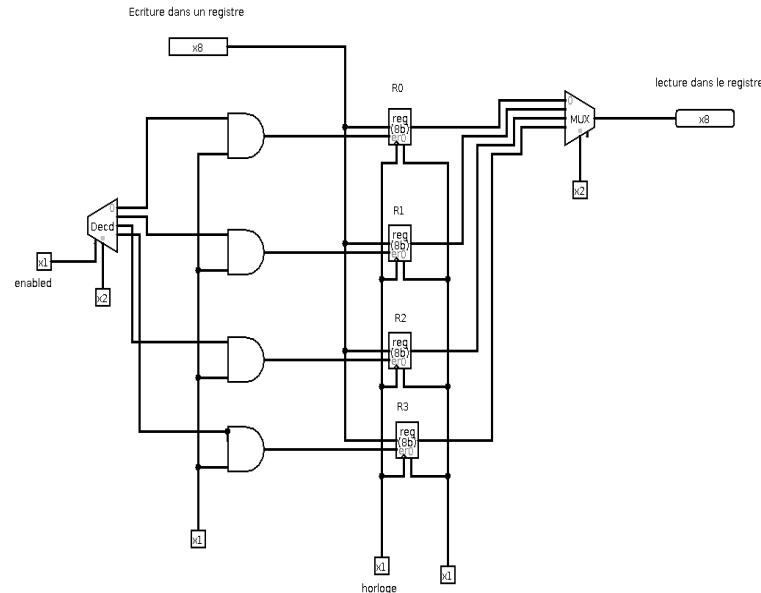
• Un multiplexeur:

Un multiplexeur (Mux) permet , grâce à des sélectionneurs (entrées de sélection) ,de sélectionner (sortie) une entrée parmi plusieurs .Par exemple :



- 1. Réaliser le schéma précédent dans *logisim* et tester les différentes entrées (e1,e2 pour les données à sélectionner et C pour le sélecteur).
- 2. De même réaliser le schéma d'un multiplexeur avec 4 données à sélectionner e1,e2,e3 et e4. Combien d'entrées de sélection faut-il envisager?
- Pour lire et écrire dans plusieurs registres , on utilise un décodeur "1 pour 4" , un multiplexeur et 4 registres R1,R2,R3 et R4.

On donne ci-dessous le schéma réalisé dans *logisim* et permettant de telles opérations.



Reproduire ce schéma et le tester.

- une mémoire (RAM)
Regarder d'abord ceci <http://www.cburch.com/logisim/docs/2.3.0/libs/mem/ram.html>
La mémoire peut fournir des valeurs situées à une adresse, ou bien écrire des valeurs à cette adresse. L'adresse doit être spécifiée en permanence sur l'entrée A.

Lecture: Pour faire fonctionner la mémoire en mode lecture, il faut mettre le port *ld* à 1. Alors, les données lues sont disponibles sur le port D sur le côté droite de la mémoire.

Écriture: Pour écrire une donnée en mémoire, il faut présenter cette donnée sur le port D sur le côté gauche, tout en appliquant une valeur 1 sur le port *store*.

La mémorisation est effectuée lorsqu'un front montant est envoyé sur l'entrée d'horloge (petit triangle situé en bas).

Important. Il faut faire en sorte que la valeur imposée sur le port *ld* soit toujours inversée par rapport à la valeur sur *store*.

Il n'est pas nécessaire de connecter les ports *sel* et *clr*.

Réaliser le schéma proposé (cas d'une mémoire synchrone) dans le lien ci-dessus et tester.

- Ajouter alors une mémoire RAM et compléter le schéma 1 en lui adjoignant une mémoire .