

Réaliser des opérations avec des portes logiques (partie 1)

Nous allons réaliser une opération entre deux nombres binaires, uniquement avec des portes logiques. Pour cela, construire le circuit suivant dans logisim. N'oubliez pas de nommer votre circuit comme indiqué dans la vidéo, cela va nous servir après...

<https://picassciences.com/2019/12/15/realiser-des-operations-avec-des-portes-logiques/>

Utiliser la vidéo 0.

Donner les tables de vérité des portes AND, OR, NOT, XOR, en testant dans logisim.

x	not (x)
0	
1	

a	b	a and b
0	0	
0	1	
1	0	
1	1	

a	b	a or b

a	b	a xor b

Utiliser la vidéo 1.

1. Les deux entrées binaire (à gauche) sont nommées A et B. Trouver la table de vérité de ce montage.

--

2. A quelle opération mathématique simple en binaire cela correspond-il ?

--

3. Quelle est l'appellation simple utilisée en mathématiques pour parler du 2ème bit de sortie (celui du bas) ?

--

Ensuite, nous allons mettre notre circuit dans une boîte afin de pouvoir l'utiliser de nombreuses fois facilement.

Utiliser la vidéo 2.

Nous avons jusqu'à présent réaliser une opération sur des nombres à un seul bit, ce qui est plutôt limité... Nous allons maintenant faire des opérations sur des nombres à 4 bits.

Utiliser la vidéo 3.

4. Que fait-on dans la vidéo 3 et quel est le but de ces associations ?

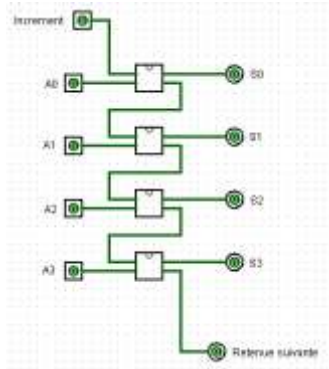
Utiliser la vidéo 4.

Expliquer le fonctionnement et l'intérêt du circuit

5. Que représente A0, A1, A2 et A3 ?
6. Pour différentes valeurs de : Incrément, A0, A1, A2 et A3, consigner les résultats obtenus et expliquez le fonctionnement du circuit et son utilité.
7. Est-il possible d'obtenir un integer overflow, décrire dans quel cas ?

Réaliser des opérations avec des portes logiques (partie 2)

Dans l'activité précédente, nous avons réalisé un demi-additionneur (half adder), car nous n'ajoutons, à un nombre de 4 bits, seulement 1. Il serait maintenant intéressant de pouvoir ajouter ensemble, deux nombres de 4 bits.



(Half Adder)

Le problème, c'est qu'il va falloir propager, non pas propager une seule et unique retenue, mais de nombreuses retenues à chaque étape du montage. Pour cela, nous allons réaliser un additionneur complet (Full Adder), qui dispose d'une entrée ET une sortie, toutes deux, entièrement dévolues à propager les retenues.

8. Utiliser la vidéo 5 pour construire un Full Adder et remplir la table de vérité suivante.

A	B	Retenue Entrée	Somme	Retenue sortie

9. Nous allons maintenant réaliser un circuit additionneur de deux nombres de 4 bits. **Utiliser la vidéo 6** qui vous montre le début de la construction du circuit, à vous de faire le reste !

10. Utiliser 5 couples de nombres binaires sur 4 bits de votre choix (par exemple 0111 et 1100) et décrire ce que l'on obtient. Commenter les exemples.

11. Mettre en évidence un cas d'integer overflow.