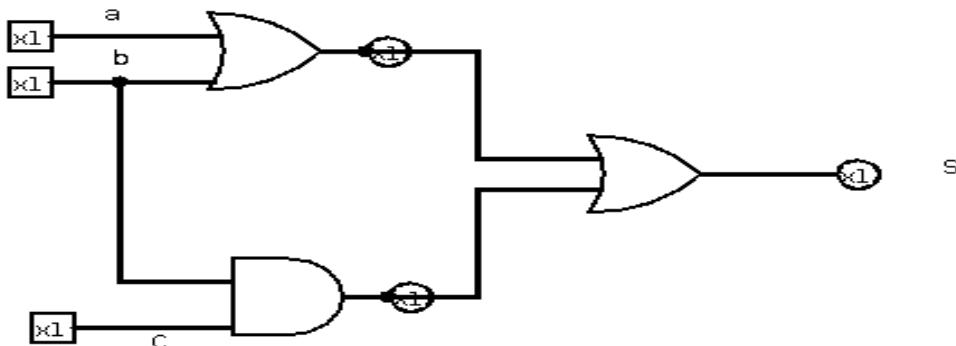


Un guide sur l'utilisation de *logisim* est disponible à l'adresse suivante.

Consulter ce guide écrit en anglais avant de réaliser les exercices ci-dessous:

<http://www.cburch.com/logisim/docs/2.7/en/html/guide/>

Exercice 1



1. Donner l'expression booléenne de S en fonction des variables a, b et c .

a	b	c	S

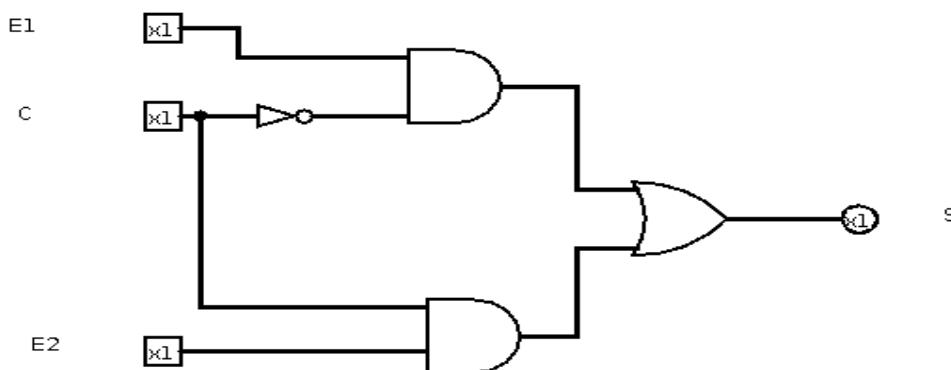
2. Compléter la table de vérité ci-contre.

Exercice 2

1. Réaliser à l'aide de logisim le circuit logique des expressions booléennes:

$$A = (a + b).c \quad \text{et} \quad B = a.c + b.c$$
2. Vérifier par simulation l'égalité des deux expressions A et B

Exercice 3 (multiplexeur)



1. Donner l'expression booléenne de S en fonction des variables C,E1 et E2 .

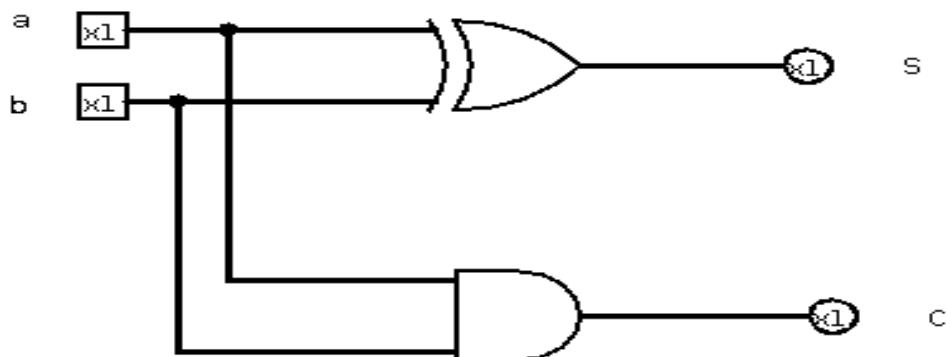
C	E1	E2	S

2. Compléter la table de vérité ci-contre.

Le circuit étudié est appelé multiplexeur à 2 entrées. Selon la valeur de la commande (C), il permet de reproduire en sortie (S) :

- le signal E1 si C est à 0.
- le signal E2 si C est à 1.

Exercice 4 (demi-additionneur)

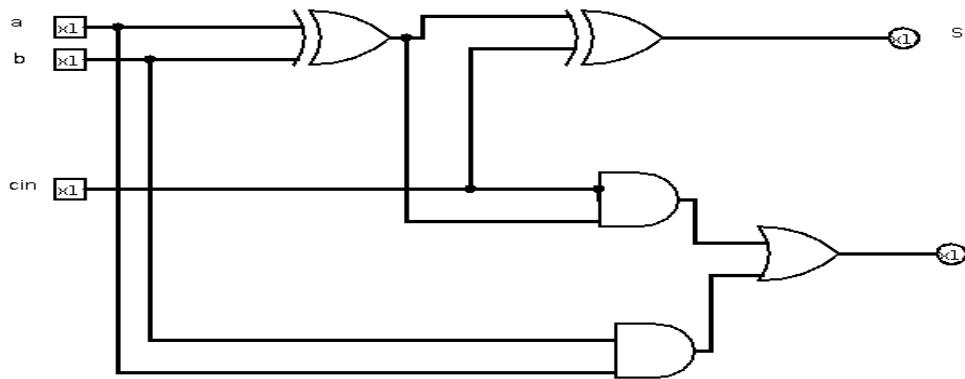


Le circuit étudié, appelé demi-additionneur, permet d'additionner deux bits a et b. Il comporte deux sorties C et S qui représentent deux expressions booléennes.

1. Donner les expressions booléennes de C et S en fonction de A et B .
2. Compléter la table de vérité de C et S.

a	b	S

a	b	C

Exercice 5 (additionneur complet)

1. Réaliser le circuit ci-dessus à l'aide du logiciel logisim.
2. Compléter la table de vérité ci-dessous:

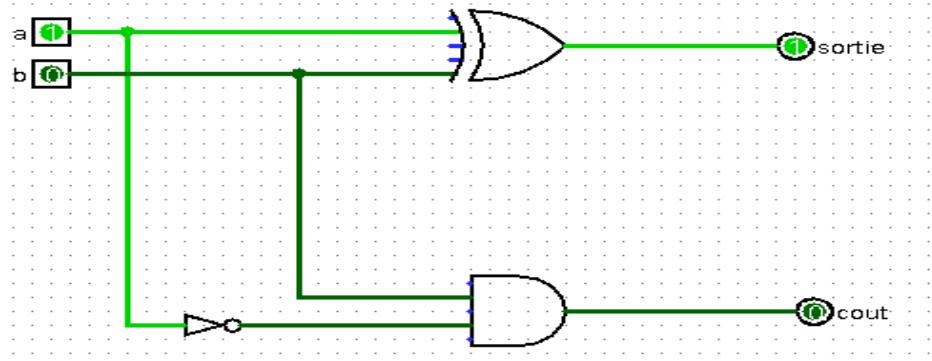
Cin	a	b	Cout	S

Exercice 6(comparateur binaire)

1. On donne la fonction booléenne PGQ (plus grand que) définie sur $\mathcal{B} \times \mathcal{B}$ qui à un couple (a, b) associe 1 si l'entier égal à a est strictement supérieur à l'entier égal à b et 0 sinon.

En utilisant une table de vérité , donner l'expression booléenne de la fonction .

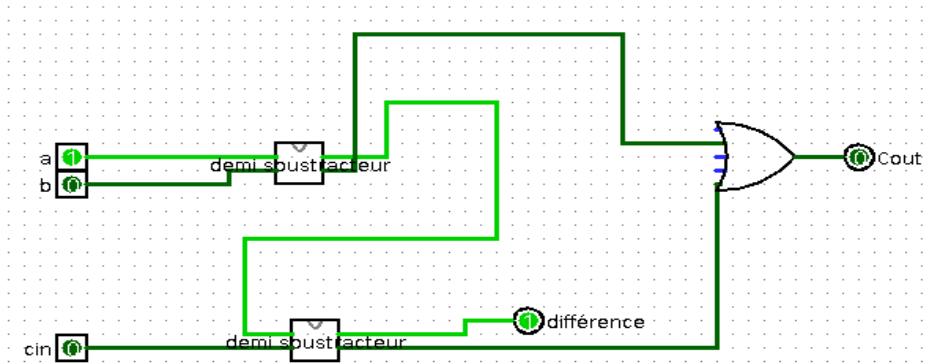
2. De même , donner l'expression booléenne de la fonction PPQ (plus petit que) en remplaçant strictement supérieur par strictement inférieur dans l'énoncé de la question 1)
3. Faire de même pour la fonction EG (égalité).
4. réaliser le circuit logique (avec logisim) qui a pour entrées deux booléens a et b et trois sorties correspondant aux trois fonctions précédentes.

Exercice 7

Le circuit étudié, appelé demi-soustracteur, permet de soustraire deux bits a et b . Il comporte deux sorties $Cout(C)$ et $Sortie(S)$ qui représentent deux expressions booléennes.

1. Donner les expressions booléennes de C et S en fonction de a et b .
2. Compléter la table de vérité de $Cout$ et S .

a	b	S	cout

Exercice 8

Le soustracteur complet est l'élément essentiel pour soustraire des nombres multi-bits, car il prend en compte la retenue $Cout$ de la soustraction précédente. Il a trois entrées A, B, Cin et deux sorties D et $Cout$.

$$D = A \oplus B \oplus C_{in}$$

$$\text{Retenue: } C_{out} = \overline{A} \cdot B + \overline{A} \cdot C_{in} + B \cdot C_{in}$$

Construction (Logigramme)

Le soustracteur complet peut être réalisé en utilisant deux demi-soustracteurs et une porte OU (OR) :

Un premier demi-soustracteur calcule $D_1 = A \oplus B$ et $E_1 = \overline{A} \cdot B$

Un second demi-soustracteur est utilisé pour calculer la différence finale : $D = D_1 \oplus C_{in}$

et la retenue finale : $C_{out} = \overline{D_1} \cdot C_{in}$

1. Faire le logigramme du soustracteur complet à l'aide de *logisim*.
2. Utiliser le soustracteur complet pour construire un soustracteur qui effectue la soustraction de deux binaires a et b formés de plusieurs bits (par exemple 2 ou 3 bits).