

# **Lab # 4: CIRCUITS ARITHMÉTIQUES**

**ITI 1500 - Systèmes Numériques I**

**Hiver 2021**  
**École de Génie Électrique et Science Informatique**  
**Université d'Ottawa**

Professeur: Dr. Ahmed Karmouch

Groupe # 22

Étudiant 1: GUIGMA Armand 300105138

Étudiant 2: SISSOKO Ramatoullaye Bahio 300144949

Date de l'expérimentation: 25/03/2021

Date de la soumission: 01/04/2021

## Objectifs:

Après avoir expérimenté l'implémentation des fonctions booléennes d'un décodeur puis d'un multiplexeur lors du laboratoire précédent, nous avons à présent pour objectifs essentiels :

- l'implémentation, la compilation et la simulation d'un additionneur complet;
- le téléversement de ce circuit sur une carte altera DE2-115;
- l'utilisation de l'additionneur complet pour implémenter et simuler un additionneur/soustracteur 8-bit;
- la réalisation et la simulation d'un détecteur de débordement utilisé dans un additionneur soustracteur complément à 2.

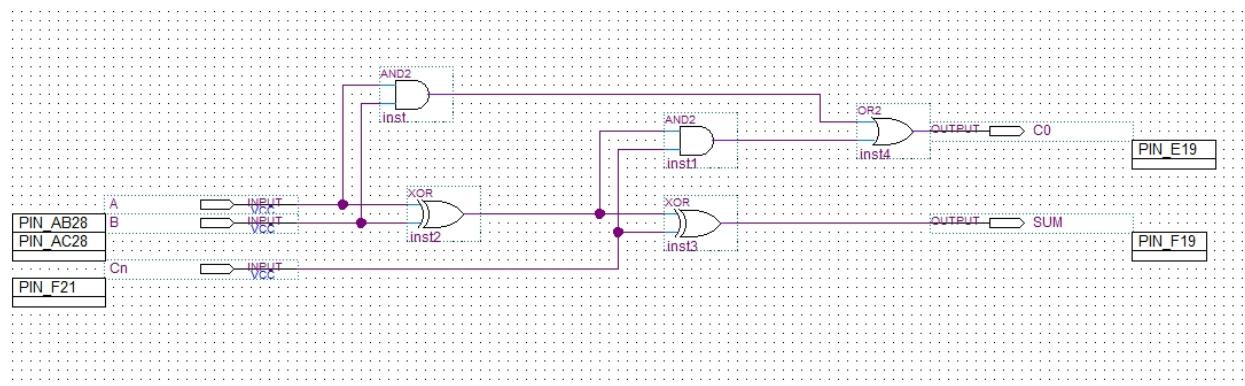
## Équipements & Composants

- Logiciel QUARTUS II 13.0 Service Pack 1
- Carte Altera DE2-115

## Diagrammes de circuit

### Partie I – Additionneur Complet

Les figures ci-dessous représentent les captures d'écrans des circuits que nous avons construits sur le logiciel Quartus.



**Figure 1:** Capture d'écran du diagramme du circuit d'un additionneur complet  
(Figure 5.4.1 du manuel de laboratoire)

## Partie II – Additionneur 8 bit

### Diagramme de circuit d'un additionneur 8 bit

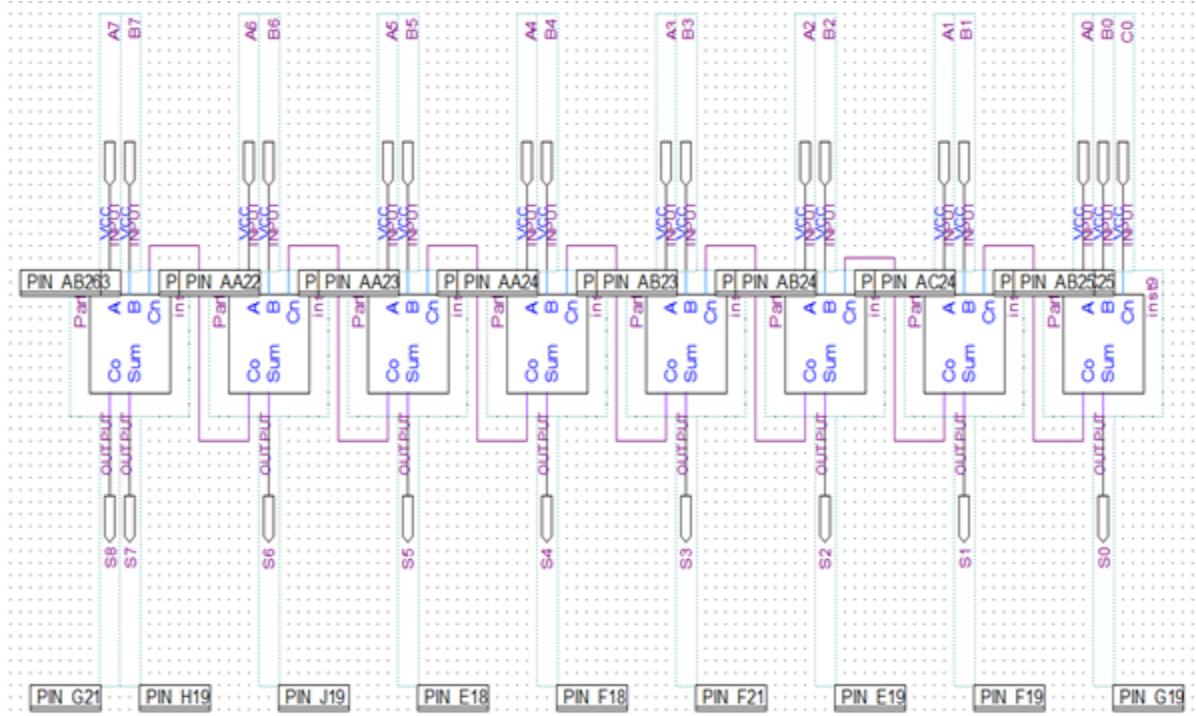


Figure 2: Circuit logique d'un additionneur 8 bit

## Données expérimentales et traitement de données:

### Partie I - Additionneur complet

Circuit logique créé à l'aide de la figure 5.4.1 (du manuel de lab).

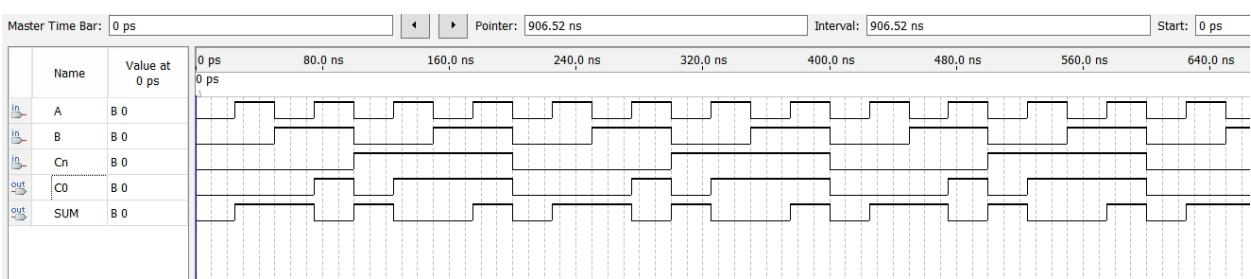


Figure 3: Simulation de l'onde de sortie du circuit de la figure 5.4.1(manuel de lab).

**Table 1 : Données expérimentales observées à partir de la carte Altera**

Entrées binaires déterminées à partir des commutateurs DIP			Sorties observées à partir du LED	
A	B	Cn	C0	Sum
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	1

## Partie II - Additionneur 8 bit

### Additionneur 8 bit

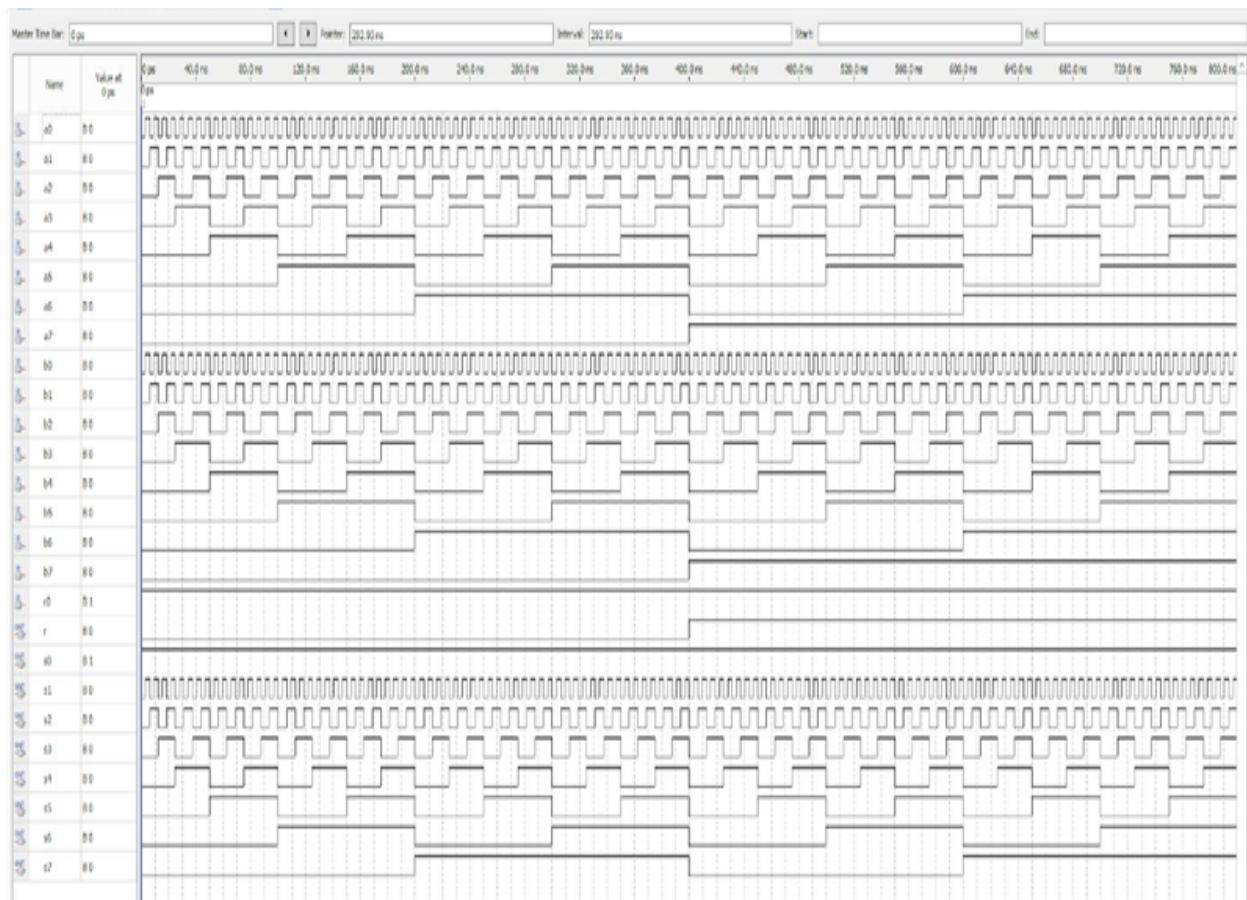


Figure 4: Simulation de l'onde de sortie d'un additionneur 8 bit

**Table 2 : Affichage correspondant aux chiffres 0-9**

Entrées binaires déterminées à partir des commutateurs DIP	Retenue observé sur la sortie LED	Somme binaire observées sur les sorties LED

A=A7A6A5A4A3A2A1A0	B=B7B6B5B4B3B2B1B0	C0	S
01111111	00000001	0	10000000
11111111	00000001	1	00000000
11000000	01000000	1	00000000
11000000	10000000	1	01000000

## Comparaison entre les Données Théoriques et les Données Expérimentales

### Partie I: Additionneur complet

**Table 3:** Comparaison entre les données expérimentales et théoriques pour l'additionneur complet (figure 5.4.1 du manuel de lab)

Entrées binaires déterminées à partir des commutateurs DIP			Sorties observées à partir du LED		Résultats attendus	Résultats attendus
A	B	Cn	C0	SUM	C0	SUM
0	0	0	0	0	0	0
0	0	1	0	1	0	1

0	1	0	0	1	0	1
0	1	1	1	0	1	0
1	0	0	0	1	0	1
1	0	1	1	0	1	0
1	1	0	1	0	1	0
1	1	1	1	1	1	1

Les résultats obtenus théoriquement sont identiques à ceux obtenus expérimentalement pour l'additionneur complet de la partie I comme prévu.

## Partie II: Additionneur 8 bit

**Table 4:** Comparaison entre les données expérimentales et théoriques pour l'additionneur 8 bit

Entrées binaires déterminées à partir des commutateurs DIP		Retenue observé sur la sortie LED	Somme binaire observées sur les sorties LED	Résultats attendus	Résultats attendus
A=A7A6A5A4A3A2A1A0	B=B7B6B5B4B3B2B1B0	C0	S	C0	S
01111111	00000001	0	10000000	0	10000000

11111111	00000001	1	00000000	1	00000000
11000000	01000000	1	00000000	1	00000000
11000000	10000000	1	01000000	1	01000000

Les résultats obtenus théoriquement sont identiques à ceux obtenus expérimentalement pour l'additionneur 8 bit de la partie II comme prévu.

## Discussion & Conclusion

Dans ce laboratoire les objectifs premiers sont entre autre:

- Créer et simuler un additionneur complet
- Tester un additionneur complet sur la carte Altera DE2-115
- Utiliser des additionneurs complets pour implémenter une addition/soustraction 8-bit
- Créer un schéma hiérarchique incluant des composants additionneurs complets et décodeur 7 segment
- Implémenter un détecteur de débordement utilisé dans un additionneur à complément à 2

Pour la méthodologie expérimentale, les circuits ont été réalisés, compilés et simulés sur le logiciel Quartus. La simulation nous a permis de dresser les différentes tables de vérité. Les instructions sont dans le manuel de laboratoire.

L'analyse théorique nous a permis de vérifier si nos résultats expérimentaux sont identiques aux résultats théoriques et ils concordent.

Dans ce laboratoire on a appris à maîtriser les additionneurs complets et à 8-bits, la création d'un schéma hiérarchique combinant additionneurs et décodeurs. Aussi on a augmenté nos connaissances dans les opérations d'arithmétique binaire.

Les lois de l'arithmétique binaire ont été utilisées pour ce laboratoire

Certaines difficultés ont été rencontrées au niveau de la partie II dans l'implémentation de l'additionneur 8-bit mais le problème a été réglé en me documentant plus sur la bonne façon de faire.

## Annexe (Pré-Lab):

### Pré-lab de la partie I

$$C_0 = (AB) + (A \oplus B)(C_n)$$

$$SUM = (A \oplus B) \oplus (C_n)$$

A	B	C <sub>n</sub>	AB	A $\oplus$ B	(A $\oplus$ B)(C <sub>n</sub> )	C <sub>0</sub>	SUM
0	0	0	0	0	0	0	0
0	0	1	0	0	0	0	1
0	1	0	0	1	0	0	1
0	1	1	0	1	1	1	0
1	0	0	0	1	0	0	1
1	0	1	0	1	1	1	0
1	1	0	1	0	0	1	0
1	1	1	1	0	0	1	1

### Pré-lab de la partie II

Suivez les exemples suivants pour additionner les nombres binaires non signés présentés dans le Tableau 5.4.2. Donnez la somme à la fois en binaire et hexadécimal.

Montrez la retenue séparément.

$$01111111 + 00000001 = 10000000 \text{ (retenue} = 0\text{)}$$

$$\text{Équivalent hexadécimal: } (7F)_{16} + (01)_{16} = (80)_{16} \quad (\text{retenue} = 0)$$

$$11111111 + 00000001 = 00000000 \quad (\text{retenue} = 1)$$

$$\text{Équivalent hexadécimal: } (FF)_{16} + (01)_{16} = (00)_{16} \quad (\text{retenue} = 1)$$

$$11000000 + 01000000 = 00000000 \quad (\text{retenue} = 1)$$

Équivalent hexadécimal:  $(C0)_{16} + (40)_{16} = (00)_{16}$  ( $\text{retenue} = 1$ )

$$11000000 + 10000000 = 01000000 \quad (\text{retenue} = 1)$$

Équivalent hexadécimal:  $(C0)_{16} + (80)_{16} = (40)_{16}$  ( $\text{retenue} = 1$ )