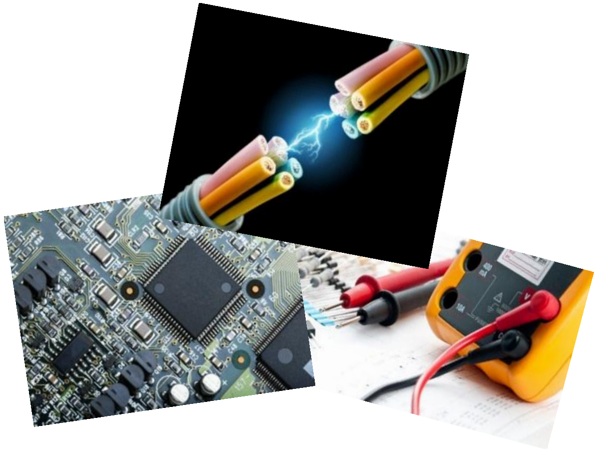
Électronique

numérique et analogique

Electronique

Niveau : 1A-3B-2P

Nombre d'heure : 42 heures



Electronique Numérique

Electronique analogique

C'est quoi un système électronique?

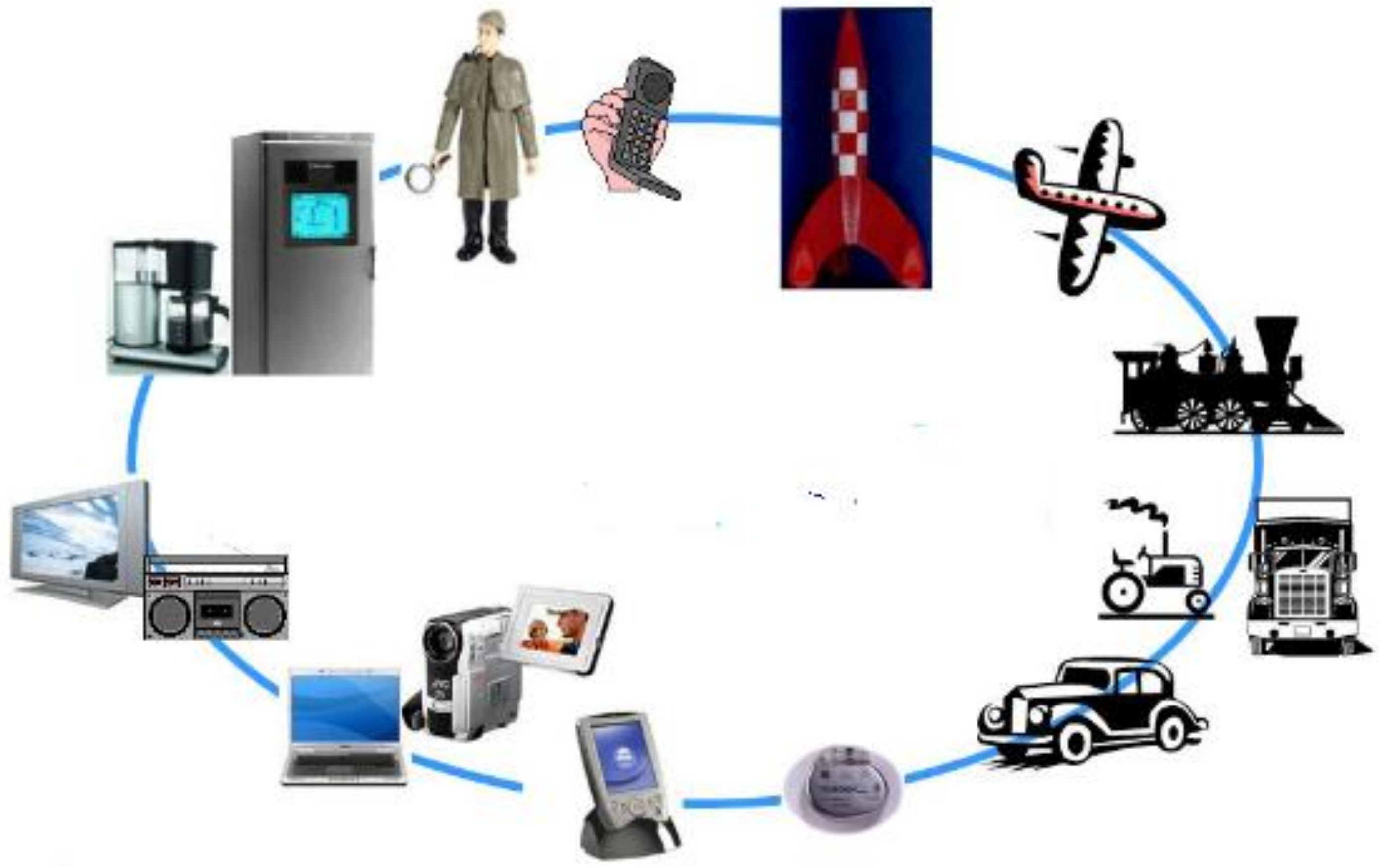
Un système électronique est un système opérant sur des faibles courants. Il se compose essentiellement d'un calculateur, discret ou intégré, câblé ou programmé, et est relié sur son entrée à des capteurs et sur sa sorties à des actionneurs. Tous les appareils informatiques sont des systèmes électroniques.

Electronique et informatique ?

Un système embarqué est un système **électronique** et **informatique**.

Un système embarqué est un dispositif matériel (hardware) comportant des partie logicielles (software).

Exemples des systèmes Embarqués



L'embarqué à Esprit

SLEAM: **S**ystèmes **L**ogiciels **E**mbarqués **A**mbiants et **M**obiles

- Diplôme National d'ingénieur en Informatique : Option embarqué
- Collaboration avec l'université de Nice , France

Plan du module: Partie numérique

Séquence 0 — Système de numération et représentation des nombres

Séquence 1 — L’algèbre de Boole

Séquence 2 — Les circuits Combinatoires

Séquence 3 — Les circuits séquentiels 1 : les bascules

Séquence 4 — Les circuits séquentiels 2 : les compteurs

Séquence 5 — Les circuits séquentiels 3: Les registres à décalage

Plan du module: Partie analogique

Séquence 0 Analyse des circuits électriques

Séquence 1 Théorèmes fondamentaux

Séquence 2 La diode

Séquence 3 Le transistor

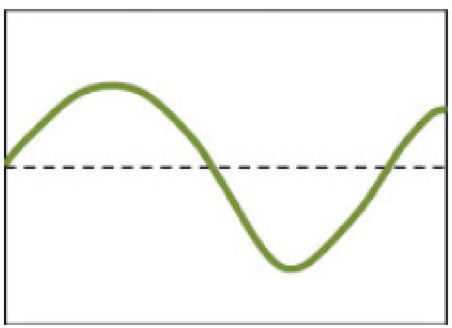
Séquence 4 AOP

Introduire le module

Nombreux sont les systèmes qui utilisent des grandeurs en entrées, les traitent et les délivrent en sortie des commandes ou des information pour l'utilisateur. IL y a deux façons pour représenter ces grandeurs:

- ❑ **Représentation analogique**

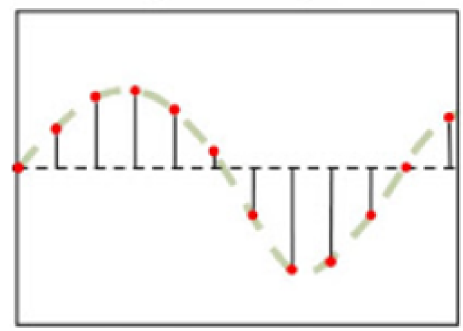
Infinité des valeurs



Exemple 1: Tachymètre d'une voiture analogique

- ❑ **Représentation numérique**

Nombres finis des valeurs



Exemple 2: Tachymètre d'une voiture numérique

Electronique Numérique

Niveau :1A-2P-3B

Prérequis : Mathématiques : addition, division , multiplication , puissance, informatique

Acquis d'apprentissage:

- ✓ Analyser des circuits numériques simples comportant des composants vus en cours
- ✓ Analyser une discreption textuelle et la traduire en un schéma de circuit conceptuel
- ✓ Concevoir des circuits simples combinant des circuits vus en cours
- ✓ Se sera familiarisé avec:

- ❑ l'utilisation du logiciel pour la simulation des circuits électroniques 'ISIS'
- ❑ La notion « datesheet »ou fiche technique d'un composant

Environnement du travail

- **Code blocks :** est un environnement de développement intégré libre, il est orienté C et C++

- **ISIS Proteus:** Le **logiciel ISIS Proteus** est un logiciel de réalisation de carte électronique qui permet aussi la simulation de montages électroniques.

Chapitre 1: Système de numération et représentation des nombres

Plan de la séquence 0

Séquence 0

Système de numération et représentation des nombres

- ❑ Introduction
- ❑ Système de numération et représentation des bases
- ❑ Conversion entre les bases
- ❑ Représentation des nombres négatifs et débordement
- ❑ Opérations arithmétiques
- ❑ Les codes: Code BCD, code Gray

Séquence 0 Système de numération et représentation des nombres

<u>Les pré-requis:</u>

- Architecture des ordinateurs
- Les opérations mathématiques
- La valeur absolue

Séquence 0

Système de numération et représentation des nombres

Les Objectifs:

À la fin de cette séance, l'étudiant doit être capable de:

- ✓ **Objectif principal:**

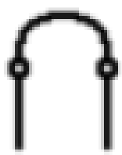
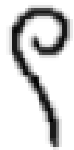
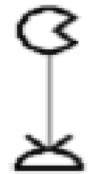
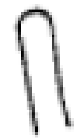
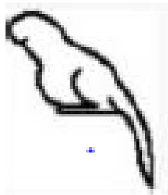
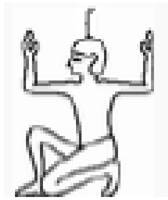
Comprendre le système de numération et la conversion entre les différentes bases

Objectifs spécifiques:

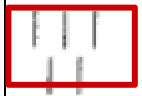
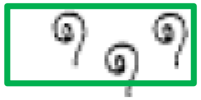
- ❑ Connaitre les différentes bases
- ❑ Connaitre le débordement
- ❑ Connaitre le complément à 2 (CA2)
- ❑ Distinguer entre la représentation en CA2 et la représentation binaire naturel
- ❑ Appliquer le CA2 .
- ❑ Effectuer des opérations en appliquant la représentation CA2

Introduction

L’écriture Hiéroglyphique

1	10	100	1 000	10 000	100 000	1 000 000
						
Baton	Anse de panier	Corde enroulée	Fleur de lotus	Doigt coupé	têtard	Dieu assis

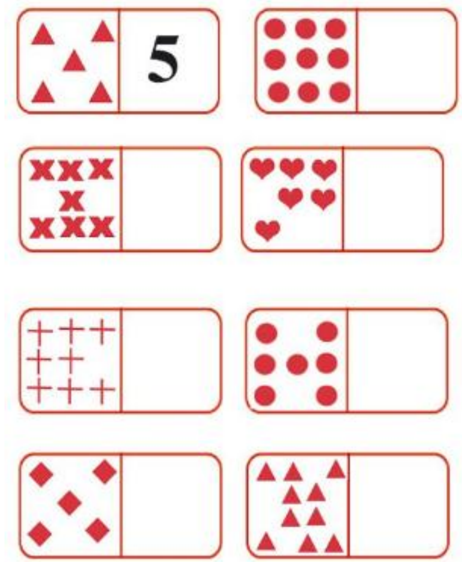
Pour lire un nombre, on additionne la valeur de l’ensemble des symboles utilisés dans une représentation d’un nombre donné

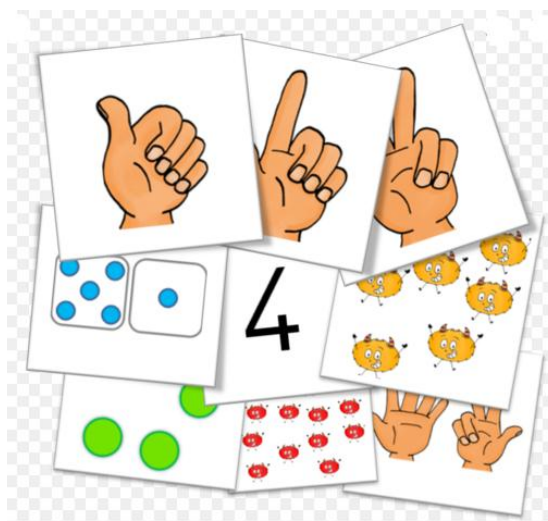
Nombre en hieroglyphes	Nombre associé
  	5 + 40000 + 300 =40 305

Les systèmes de numération

Le système décimal : base 10

0, 1 , 2, 3, 4, 5, 6 , 7, 8, 9

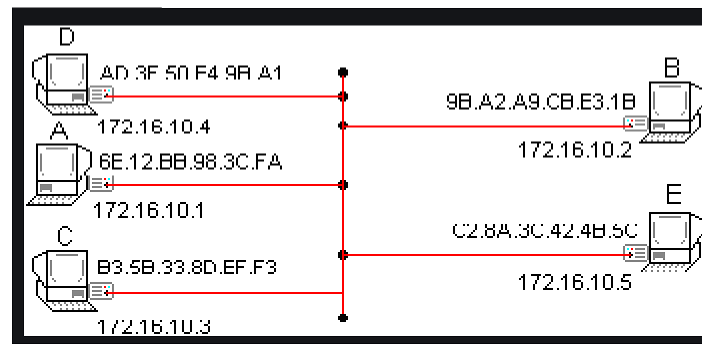




Les systèmes de numération

❑ **Système décimal**

❑ **Système binaire:0, 1**



❑ **Système octal : 0,1,2,3,4,5,6,7,8**

❑ **Système hexadécimal :0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F**

Les systèmes de numération

Quelle est la base utilisée pour représenter le nombre 457?

Ce nombre peut être à la base 10, 8 ou bien 16

Comment alors on peut savoir la base correspondante ?

$(457)_{10}$ La base 10 $(457)_{10} = 4 \times 10^2 + 5 \times 10^1 + 7 \times 10^0$

$(457)_{16}$ La base 16 $(457)_{16} = 4 \times 16^2 + 5 \times 16^1 + 7 \times 16^0$

Exemple : $(1001110011010101011111110)_2$

Les systèmes de numération

Le système décimal : la base 10

Soit le nombre $(1978)_{10}$

Milliers	Centaines	Dizaines	Unités
10^3	10^2	10^1	10^0
1	9	7	8

Le poids le plus fort

Le poids le plus faible

$$1978=1000+900+70+8=\boxed{1*10^3+9*10^2+7*10^1+8*10^0}$$

Forme polynomiale en 10

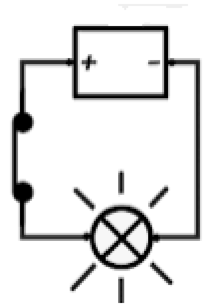
Les systèmes de numération

Le système binaire : la base 2

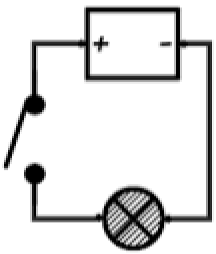
Les systèmes électroniques et électriques sont basées sur les interrupteurs:

- ❑ Interrupteur ouvert
- ❑ Interrupteur fermé

Dans cette constatation est née d'utiliser le système binaire (base 2) qui possèdes que deux symboles :**0 et 1**.



1: Si l'interrupteur est fermé alors la lampe est allumée



0: Si l'interrupteur est ouvert alors la lampe est éteinte

Les systèmes de numération

Le système binaire : la base 2

Soit le nombre $(1001)_2$

2^3	2^2	2^1	2^0
1	0	0	1

Le poids le plus fort
Most Significant Bit, ou **MSB**

Le poids le plus faible
Least Significant Bit, ou **LSB**

$$1001 = \boxed{1*2^3+0*2^2+0*2^1+1*2^0}$$

Forme polynomiale en 2

Les systèmes de numération

Le système binaire : la base 2

- Sur un seul bit : 0 , 1

. Sur 2 bits :

Binaire	Décimal
00 $= 0 \times 2^0$	0
01 $= 1 \times 2^0$	1
10 $= 1 \times 2^1 + 0 \times 2^0$	2
11 $= 1 \times 2^1 + 1 \times 2^0$	3

4 combinaisons= 2^2

Sur 3 Bits

Binaire	Décimal
000 $= 0 \times 2^2 + 0 \times 2^1 + 0 \times 2^0$	0
001 $= 0 \times 2^2 + 0 \times 2^1 + 1 \times 2^0$	1
010	2
011	3
100	4
101	5
110	6
111 $= 1 \times 2^2 + 1 \times 2^1 + 1 \times 2^0$	7

8 combinaisons= 2^3

esprit

Les systèmes de numération

Le système hexadécimal : la base 16

16 symboles sont utilisés dans cette base qui sont : {0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F}

$(231)_{16}=2*16^{2}+3*16^{1}+1*16^{0}$

Forme polynomiale en 16

$(AB2)_{16}=$
$A*16^{2}+B*16^{1}+2*16^{0}=10*16^{2}+11*16^{1}+2*16^{0}$

forme hexadecimal en 16

Décimal	Hexadécimal
0	0
1	1
2	2
3	3
4	4
5	5
6	6
7	7
8	8
9	9
10	A
11	B
12	C
13	D
14	E
15	F

Conversion entre les bases :

Le transcodage (ou conversion de base) est l'opération qui permet de passer de la représentation d'un nombre exprimé dans une représentation du même nombre mais exprimé dans une autre base.

Les différentes conversions sont :

- **De la base 10 à une base X (X peut être 2 ou 16)**
- **De la base X (X peut être 2 ou 16) vers la base 10**
- **De la base 16 vers la base 2**
- **De la base 2 vers la base 16**
-

Conversion entre les bases :

De la base X (X peut être 2 ou 16) vers la base 10:

La méthode utilisée est **la forme polynomiale en X .**

2 vers 10:
la forme polynomiale en 2

Exemple:

$(1001)_2 = \underbrace{1*2^3}_{=8} + \underbrace{0*2^2}_{0} + \underbrace{0*2^1}_{0} + \underbrace{1*2^0}_{1} = (9)_{10}$

$= 8 + 0 + 0 + 1$

$= (9)_{10}$

16 vers 10:
la forme polynomiale en 16

Exemple:

$(231)_{16} = \underbrace{2*16^2}_{} + 3*16^1 + 1*16^0 = (561)_{10}$

$= 512 + 48 + 1$

$= (561)_{10}$

Conversion entre les bases :

De la base 10 (X peut être 2 ou 8 ou 16) vers la base X: (X peut être 2 ou 8 ou 16)

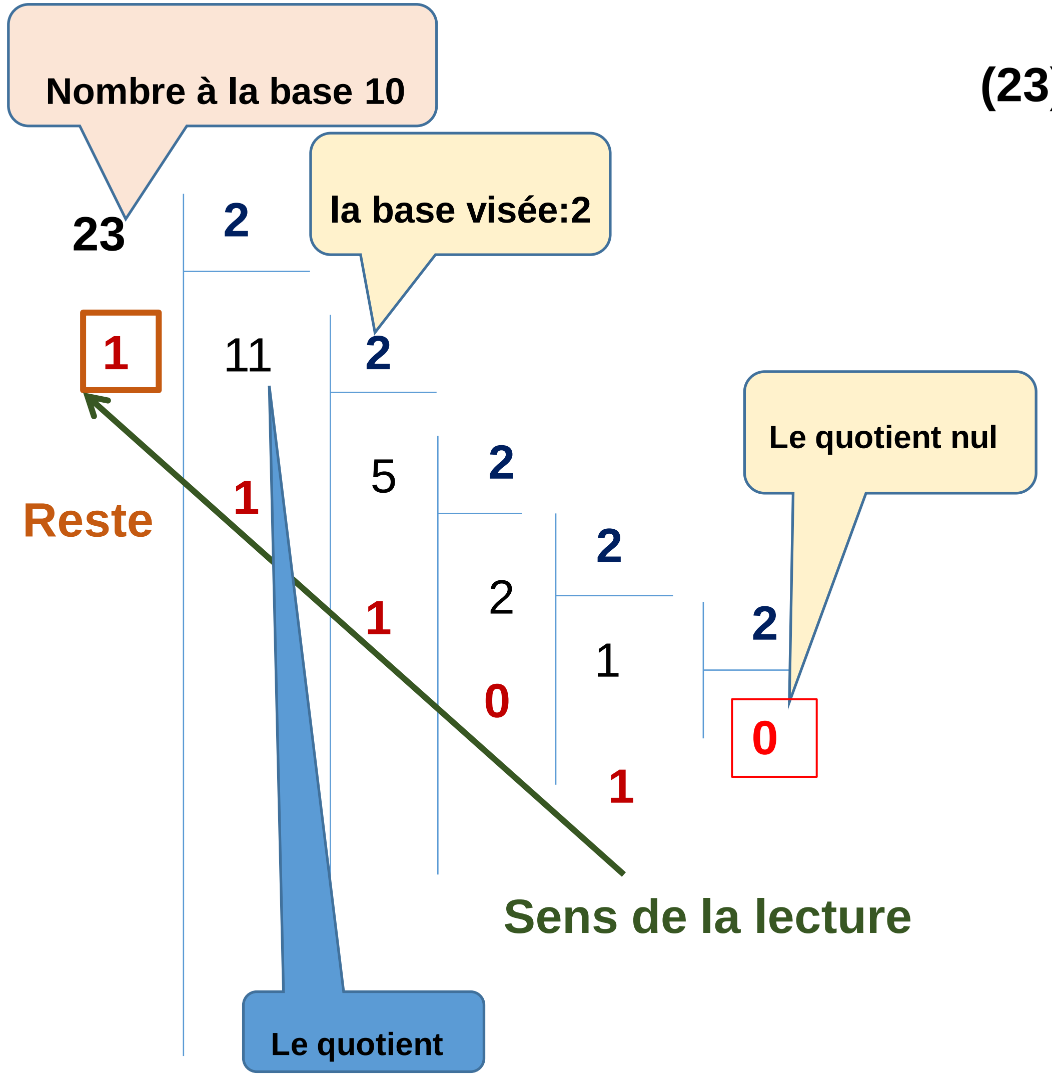
Méthode : La méthode utilisée est **la division successive en X.**

1) On divise le nombre par X (2, 8, 16) en gardant le reste.
2) Puis le quotient par la base X en gardant le reste.
3) Ainsi de suite jusqu'on obtient un quotient nul.
4) La suite des restes correspond à la base visée.
5) Prendre le reste dans le sens inverse.

Conversion entre les bases :

De la base 10 vers la base 2:

$(23)_{10}=(?)_2 = (10\ 111)_2$



Méthode : La méthode utilisée est la division successive en 2.

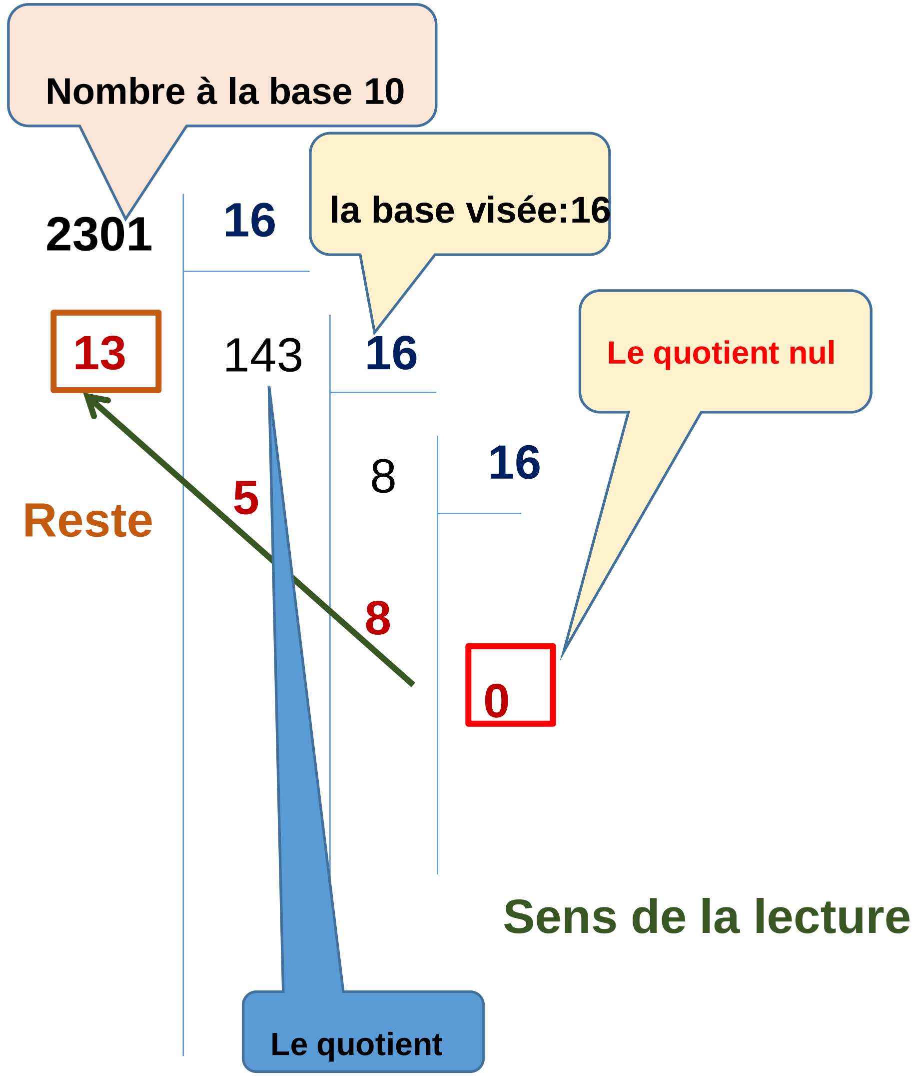
1) On divise le nombre par 2 en gardant le reste.
2) Puis le quotient par la base 2 en gardant le reste.
3) Ainsi de suite jusqu'on obtient **un quotient nul.**
4) La suite des restes correspond à la base visée.
5) Prendre le reste dans le sens inverse.

$(23)_{10}=(10111)_2$

Conversion entre les bases :

De la base 10 vers la base 16:

$(2301)_{10}=(?)_{16} = (8\ 5\ 13)_{16}$



Méthode : La méthode utilisée est la division successive en 16**.**

1) On divise le nombre par 16 en gardant le reste.
2) Puis le quotient par la base 16 en gardant le reste.
3) Ainsi de suite jusqu'on obtient **un quotient nul.**
4) La suite des restes correspond à la base visée.
5) Prendre le reste dans le sens inverse.

$(2301)_{10}=(85d)_{16}$

Conversion entre les bases :

De la base 16 vers la base 2:

<u>Méthode :</u> La méthode utilisée est **<u>l'éclatement sur 4 bits .</u>**

L'idée de base est de remplacer chaque symbole dans la base 16 par son équivalent binaire sur **4 bits**

Exemples :

$(A\ 3\ \ B\ )_{16} = (1010\ \ 0011\ \ 1011)_2$

$(F\ \ E\ \ 1\ )_{16} = (1111\ \ 1110\ \ 0001)_2$

Correspondance

Hexadécimale \Binaire

Hexadécimale\Binaire

S. Hexad.	suite binaire	S. Hexad.	suite binaire
0	0000	8	1000
1	0001	9	1001
2	0010	A	1010
3	0011	B	1011
4	0100	C	1100
5	0101	D	1101
6	0110	E	1110
7	0111	F	1111

esprit

Conversion entre les bases :

De la base 2 vers la base 16:

Méthode : La méthode utilisée est le regroupement par 4 bits.

L'idée de base est de faire des regroupements par 4 bits dans la base 2 en commençant par le bit le plus faible et le remplacer par son équivalent à hexadécimal.

Exemples :

$(1100\ 1001\ 1011\ 0011)_2 = (C\ \ 9\ \ B\ \ 3)_{16}$

$(0\ 01\ 1\ 10\ 1\ 011\ 11)_2 = (3\ \ A\ \ F)_{16}$

Correspondance

Hexadécimale \Binaire

Hexadécimale\Binaire

S. Hexad.	suite binaire	S. Hexad.	suite binaire
0	0000	8	1000
1	0001	9	1001
2	0010	A	1010
3	0011	B	1011
4	0100	C	1100
5	0101	D	1101
6	0110	E	1110
7	0111	F	1111

Exercice 1

1. Comptez jusqu'à 20 en binaire en indiquant la valeur équivalente en décimal.
2. Combien d'octets font 32 bits ?
3. Dans l'octet suivant : $(1001\ 1010)_2$, quel est le bit de poids fort, le bit de poids faible ?

Exercice 2

Faites les opérations ci-dessous, en utilisant un tableau de huit colonnes (une pour chaque bit) :

1. Calculer : $(1100\ 0110)_2 + (0010\ 0110)_2$
2. Calculer : $(1110\ 1110)_2 + (1110\ 1110)_2$: Quelle constatation faites-vous sur le résultat ?
3. Calculer : $(1110\ 1110)_2 - (1110\ 1111)_2$: Quelle constatation faites-vous sur le résultat ?

Exercice 3

1. Dans le nombre $(40\ 04)_{10}$, Qu'est-ce qui différencie le 4 de gauche de celui de droite ?
2. Convertir $(128)_{10}$ en binaire.
3. Convertir $(1100\ 0110)_2$ en décimal.

Exercice 1

binaire	décimal
0	0
1	1
10	2
11	3
100	4
101	5
110	6
111	7
1000	8
1001	9
1010	10
1011	11
1100	12
1101	13
1110	14
1111	15
10000	16
10001	17
10010	18
10011	19
10100	20

2_ $\frac{32}{4} = 8$
donc 32 bits font 8 octets

3_ Dans l'octet $(1001\ 1010)_2$
le bit de poit fort (MSB) est le 1
et le bit de poit faible (LSB) est le 0

Exercice 2 :

1 _

1100 0110	1×2^7	1×2^6	0×2^5	0×2^4	0×2^3	1×2^2	1×2^1	0×2^0
0010 0110	0×2^7	0×2^6	1×2^5	0×2^4	0×2^3	1×2^2	1×2^1	0×2^0
	1×2^7	1×2^6	1×2^5	0×2^4	1×2^3	1×2^2	0×2^1	0×2^0

$\Rightarrow (11000110)_2 + (0010\ 0110)_2 = (11101100)_2$

2 _

1110 1110	1×2^7	1×2^6	1×2^5	0×2^4	1×2^3	1×2^2	1×2^1	0×2^0
1110 1110	1×2^7	1×2^6	1×2^5	0×2^4	1×2^3	1×2^2	1×2^1	0×2^0
1×2^8	1×2^7	1×2^6	0×2^5	1×2^4	1×2^3	1×2^2	0×2^1	0×2^0

$\Rightarrow (1110\ 1110)_2 + (1110\ 1110)_2 = (111011100)_2$, on'a plus de 8 bits

3 -

11101110	1×2^7	1×2^6	1×2^5	0×2^4	1×2^3	1×2^2	1×2^1	0×2^0
11101111	1×2^7	1×2^6	1×2^5	0×2^4	1×2^3	1×2^2	1×2^1	1×2^0
1×2^8	1×2^7	1×2^6	0×2^5	1×2^4	1×2^3	1×2^2	0×2^1	1×2^0

$\Rightarrow (11101110)_2 + (11101111)_2 = (111011101)_2$, ce n'est plus un octet de 8 bits

Les codes

- **Code binaire naturel**
- **Code BCD (Binary Coded Decimal)**
- **Code réfléchi (Gray)**
- **Code ASCII**

Les codes

❑ Code BCD (Binary Coded Decimal)

- Le principe consiste à faire des éclatements sur 4 bits et de remplacer chaque chiffre décimal par sa valeur correspondante.
- Les combinaisons supérieures à 9 sont interdites.

Décimal	Binaire
0	0000
1	0001
2	0010
3	0011
4	0100
5	0101
6	0110
7	0111
8	1000
9	1001

Les codes

❑ Code BCD (Binary Coded Decimal)

Exemples

8 2 9

1000 0010 1001

1 7 3

0001 0111 0011

$(829)_{10}=(1000\ \ 0010\ 1001)_{BCD}$

$(173)_{10}=(1000\ \ 0010\ 1001)_{BCD}$

Décimal	Binaire
0	0000
1	0001
2	0010
3	0011
4	0100
5	0101
6	0110
7	0111
8	1000
9	1001

Les codes:

- Conversion binaire naturel (B, R) -binaire réflichi (B, N)

Le principe est simple ; on reproduit le bit du poids le plus fort et on compare les bits Bn+1 et Bn du nombre écrit en binaire naturel:

- **Si Bn+1 et Bn ont la même valeur , le chiffre en Br est 0.**
- **Si Bn+1 et Bn n'ont pas la même valeur , le chiffre en Br est 1**

Exemples : $(1011110)_{2Naturel} = (?)_{2Réflichi}$

Naturel : 1 0 1 1 1 1 0

Réfléchi: 1 1 1 0 0 0 1

$$(1011110)_{2Naturel} = (1110001)_{2Réflichi}$$

Les codes:

- Conversion binaire réflichi (B, R) -binaire naturel (B, N)

Le principe est simple ; on reproduit le bit du poids le plus fort et on compare le chiffre du rand n+1 du binaire naturel au chiffre du rand n de celui réfléchi

- S' ils sont égaux , on met un 0
- S'ils sont différents , on écrit 1

Exemples : $(1110001)_{2Réflichi} = (?)_{2Naturel}$

Réfléchi : 1 1 1 0 0 0 1

Naturel 1 0 1 1 1 1 0

$(1110001)_{2Naturel} = (1011110)_{2Réflichi}$

Les opérations arithmétiques:

L’addition dans la base 2:

- Addition élémentaire:

$$
\begin{array}{r}
0 \\
+ \quad 0 \\
\hline
= 0
\end{array}
\qquad
\begin{array}{r}
0 \\
+ \quad 1 \\
\hline
= 1
\end{array}
\qquad
\begin{array}{r}
1 \\
+ \quad 0 \\
\hline
= 1
\end{array}
\qquad
\begin{array}{r}
1 \\
+ \quad 1 \\
\hline
= 10
\end{array}
$$

- Exemple :

$$
\begin{array}{rcccccc}
 & \boxed{1} & & \boxed{1} & \boxed{1} & & \\
 & & 0 & 1 & 0 & 1 & 1 \\
+ & & 1 & 1 & 0 & 1 & 1 \\
\hline
= & 1 & 0 & 0 & 1 & 1 & 0
\end{array}
$$

Les références:

- Livre:

« Circuits logiques combinatoires et séquentiels », Hichem TRABELSI

- Site web

https://www.technologuepro.com/cours-systemes-logiques-3/chapitre-1-1-systeme-de-numeration-et-codage-des-informations.html

http://villemin.gerard.free.fr/Wwwgvmm/Numerati/BINAIRE/Negatif.htm

https://www.rocq.inria.fr/secret/Anne.Canteaut/COURS_C/chapitre1.html


Merci

Atelier n°2 : Serrure à code à base des portes logiques

1 _ Comparer A_0 à B_0

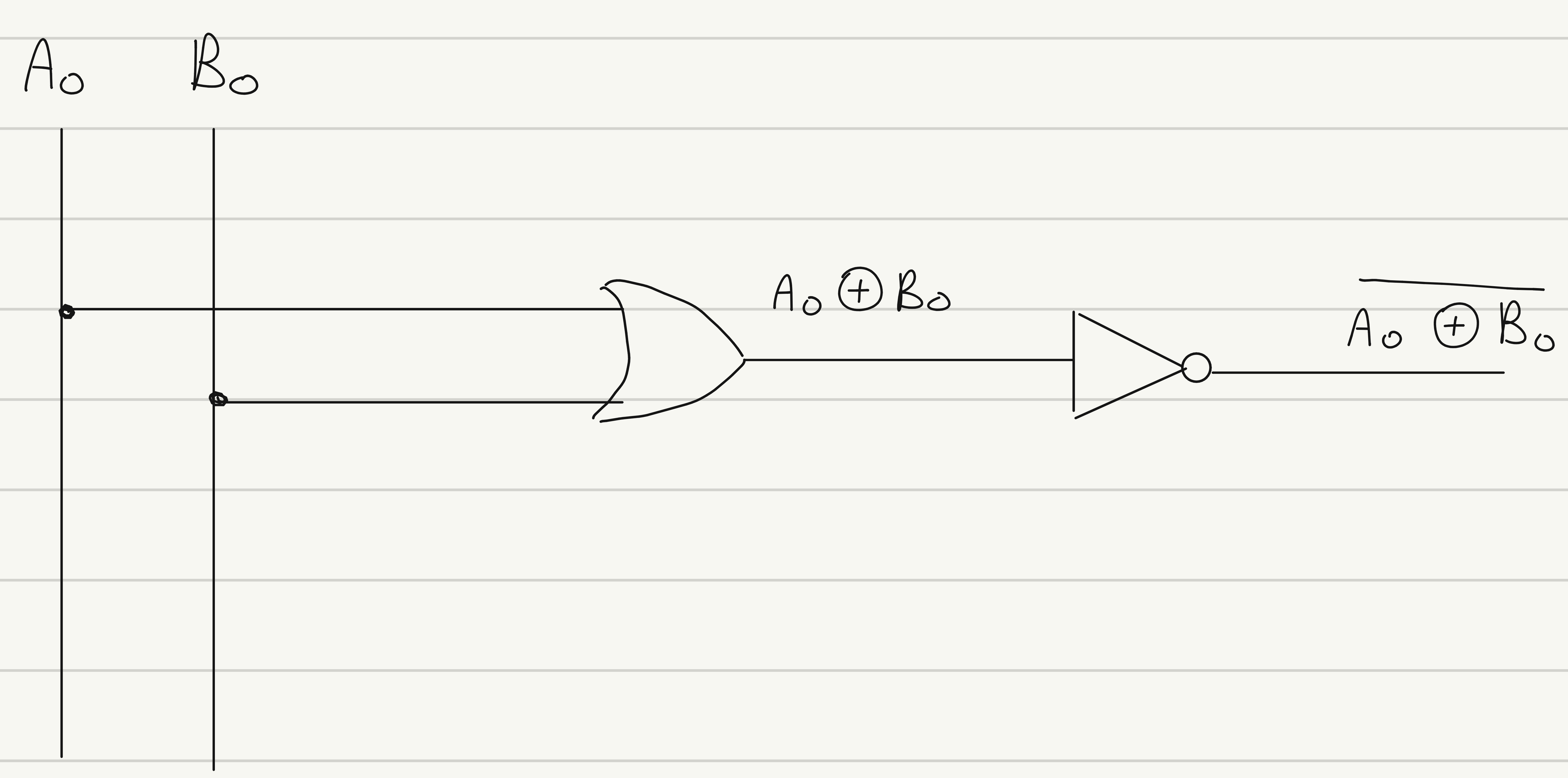
a/ A_0, B_0 : entrées du circuit de comparaison.
S_0 : sorties du circuit de comparaison. ($A_0 = B_0$)

b/

A_0	B_0	S_0
0	0	1
0	1	0
1	0	0
1	1	1

$$S_0 = \overline{A_0} . \overline{B_0} + A_0 B_0$$
$$= \overline{A_0 \oplus B_0}$$

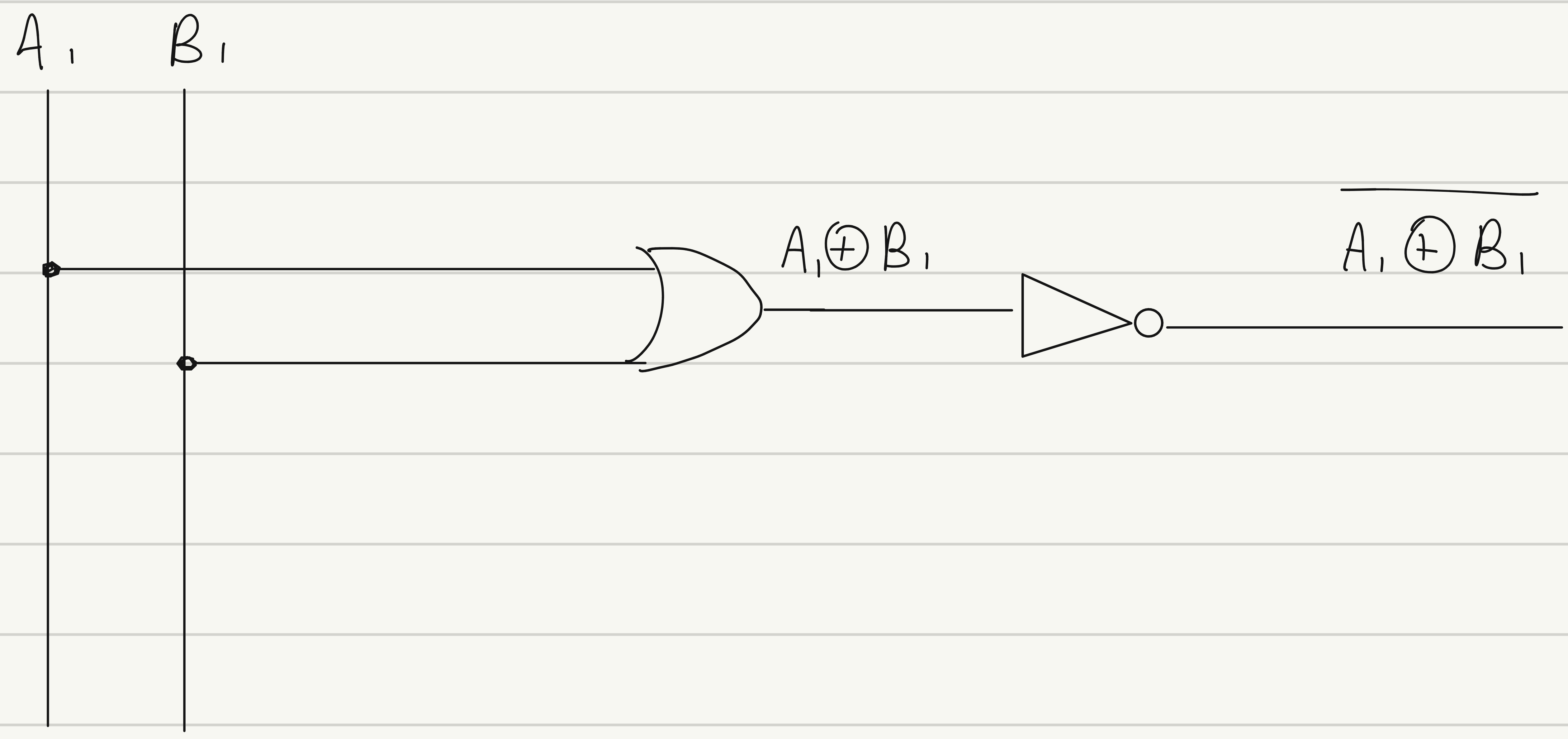
c/



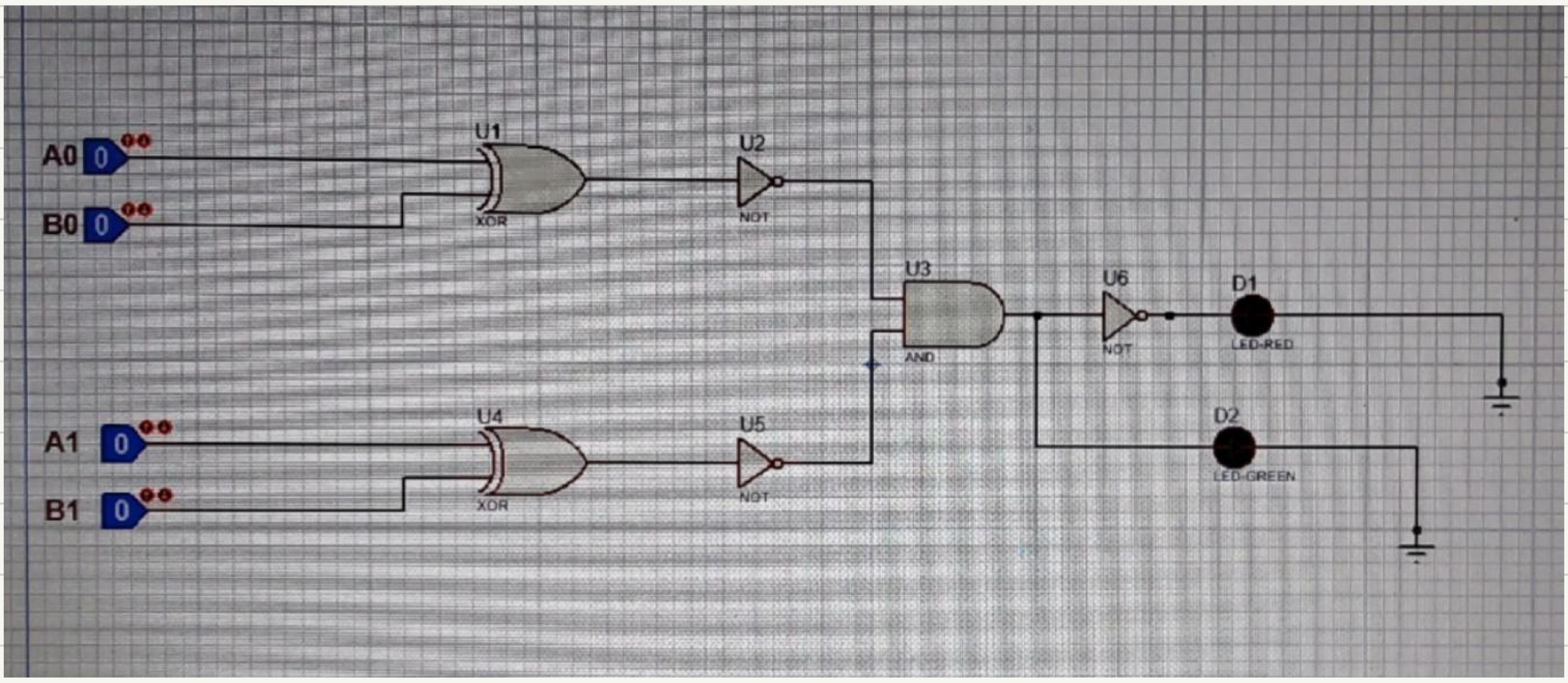
2 _ Comparer A_1 à B_1

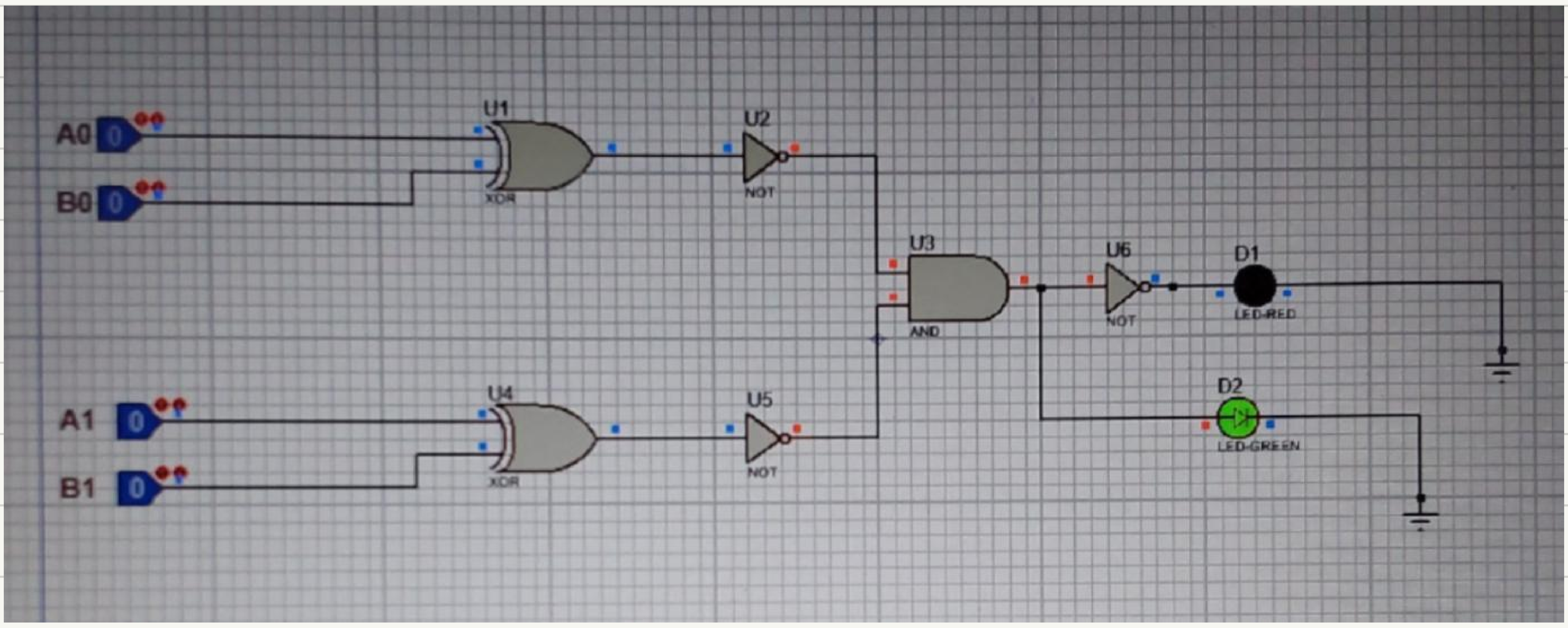
A_1	B_1	S_1
0	0	1
0	1	0
1	0	0
1	1	1

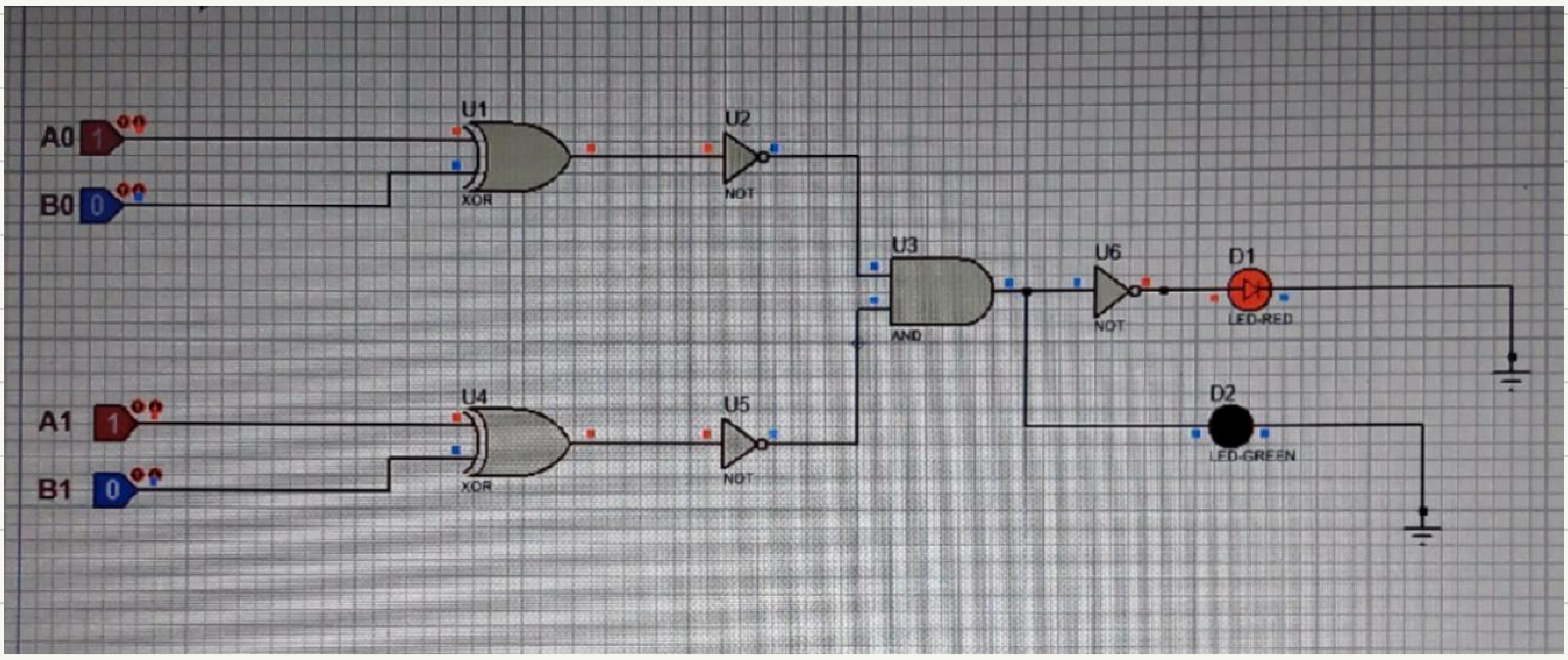
$$S_1 = \overline{A_1 \oplus B_1}$$



4 -


A0
0
B0
0
A1
0
B1
0
U1
XOR
U2
NOT
U3
AND
U4
XOR
U5
NOT
U6
NOT
D1
LED-RED
D2
LED-GREEN


A0
0
B0
0
A1
0
B1
0
U1
XOR
U2
NOT
U3
AND
U4
XOR
U5
NOT
U6
NOT
D1
LED-RED
D2
LED-GREEN


A0
1
B0
0
A1
1
B1
0
U1
XOR
U2
NOT
U3
AND
U4
XOR
U5
NOT
U6
NOT
D1
LED-RED
D2
LED-GREEN

Atelier n°3 'Additionneur complet 2 bits'

Les objectifs de l'atelier :

1. Se familiariser avec ISIS
2. Connaitre les circuits combinatoires.
3. Comprendre la fiche technique d'un circuit électronique.
4. Comprendre le rôle et le fonctionnement d'un additionneur 2 bits.
5. Comprendre le rôle et le fonctionnement d'un décodeur.
6. Comprendre le rôle et le fonctionnement d'un afficheur 7 segments.

Après la compréhension de l'addition binaire qui a été mise en évidence durant le premier chapitre qui s'intitule système de numération, des étudiants en première année informatique qui sont passionnés de l'électronique proposent de concevoir **un additionneur 2 bits** avec un **afficheur 7 segments** qui permet d'afficher le résultat trouvé et qui sera utilisé en TP pour les années prochaines à ESPRIT.

Comme première étape de l'étude et la réalisation de ce mini projet, les étudiants décident de concevoir le système qui permet de faire **l'addition** de deux nombres binaire chacun à 2 bits (**A_2A_1+ B_2B_1**) et afficher le résultat à la base 10.

Les étudiants ont demandé de l'enseignant de l'électronique de les encadrer et superviser ce mini projet. Pour commencer l'enseignant a proposé en première lieu de comprendre le fonctionnement de l'additionneur 2 bits théoriquement puis il a proposé de faire la simulation sur **ISIS** en donnant la référence de l'additionneur 2 bits '**7482**'.

- L'addition binaire:

R_2 R_1 $R_0=0$

A_2 A_1

\+

B_2 B_1

= R_2 S_2 R_1 S_1

esprit

Ecole Supérieure Privée
D'Ingénierie et de Technologies

Atelier n°3 'Additionneur complet 2 bits'

Matière : Electronique numérique **Chapitre 3 : Circuits Logique combinatoires**	**Unité pédagogique : Systèmes Embarqués Version du document : 0.1 (03/02/2019)** www.esprit.tn

1ère Partie : additionneur binaire

Pour l'addition binaire, nous allons utiliser le circuit additionneur à 2 bits montré par la figure 1 et qui a pour référence **7482** sur ISIS.

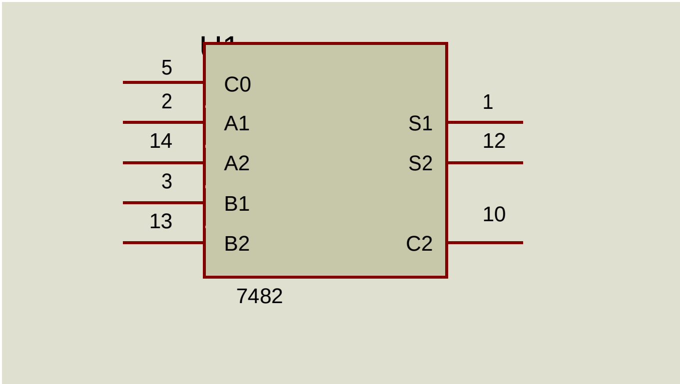


Figure1 : *Additionneur 2 bits 7482*

C0 : retenue
$A(A_2A_1)$: premier nombre binaire
$B(B_2B_1)$: deuxième nombre binaire
$(C_2S_2S_1)$: résultat addition A+B

Le circuit 7482 n'est autre que la mise en cascade de deux additionneurs à 1bit qui permet l'addition binaire de Ai et Bi (position i) et qui tient en compte de la retenue R_{i-1} de l'opération élémentaire précédente (position i-1), appelé **additionneur complet à 1 bit**.

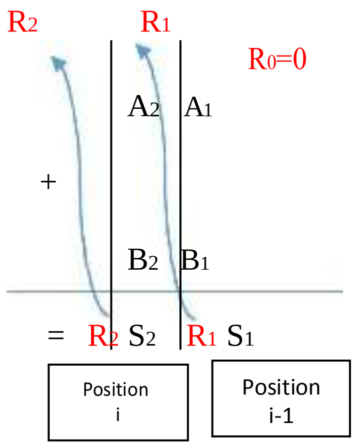


Figure 1 : addition binaire 2 bits

1- Complétez la table de vérité suivante (tableau 1) qui permet de décrire le fonctionnement d'un additionneur complet à un bit.

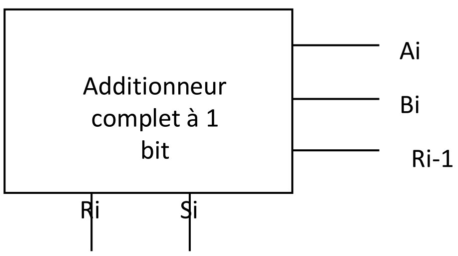


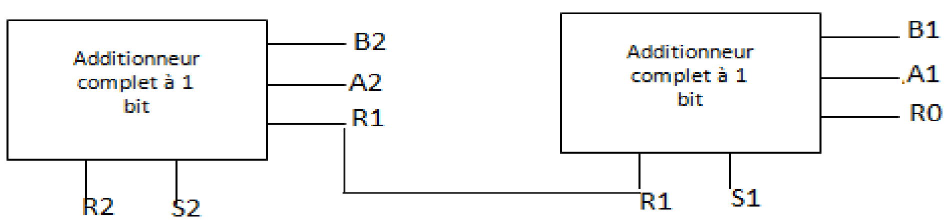
Tableau 1 : *Table de vérité de l'additionneur complet à 1 bit*

Ai	Bi	Ri-1	Ri	Si
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

Ri= $\overline{A_i}.B_i.R_{i-1} + A_i\overline{B_i}R_{i-1} + A_iB_i\overline{R_{i-1}} + A_iB_iR_{i-1} = R_{i-1}.(A_i \oplus B_i) + A_iB_i$

Si= $\overline{A_i}\,\overline{B_i}\,R_{i-1} + \overline{A_i}B_i\overline{R_{i-1}} + A_i\overline{B_i}\,\overline{R_{i-1}} + A_iB_iR_{i-1} = A_i \oplus B_i \oplus R_{i-1}$

2) Compléter le câblage suivant afin de réaliser un additionneur binaire à 2 bits et qui permet de réaliser l'opération de la figure 1. Puisque on a 2 positions (2 bits), nous utilisons 2 additionneurs complets à 1 bit, mis en cascade.



2ème Partie : afficheur 7 segments

En deuxième lieu l'enseignant leur demande de faire l'affichage sur un afficheur 7 segments qui permet la conversion des donnés binaires en une forme prête à un affichage numérique et qui est commandé par un décodeur qui a pour référence **7448** montré par la figure 2.a (pour commander l'afficheur 7 segments (montré par la figure 2.b) et remplir le table de vérité correspondante.

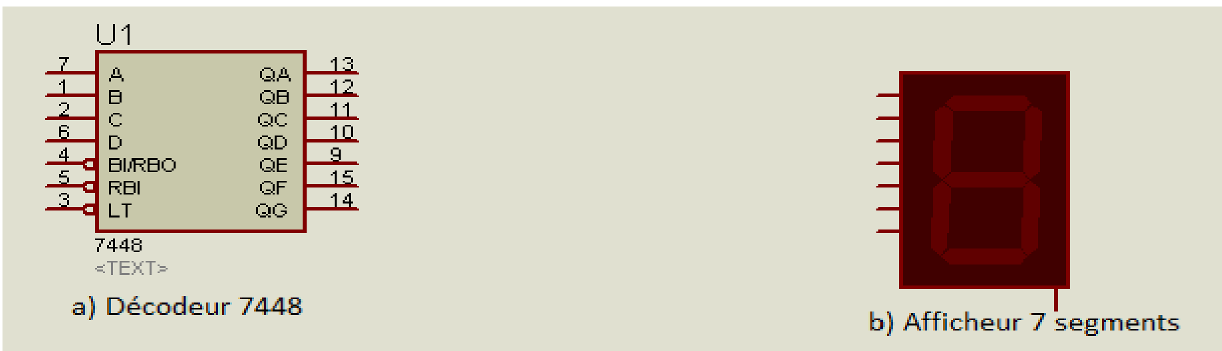


Figure 2

esprit
Ecole Supérieure Privée D'Ingénierie et de Technologies

Atelier n°3 'Additionneur complet 2 bits'

Matière : Electronique numérique Chapitre 3 : Circuits Logique combinatoires	Unité pédagogique : Systèmes Embarqués Version du document : 0.1 (03/02/2019) www.esprit.tn

Tableau 1 : *Table de vérité de décodeur '7448'*

Chiffres	D C B A	a	b	c	d	e	f	g
0	0000	1	1	1	1	1	1	0
1	0001	)	1	1	0	0	0	0
2	0010	1	1	0	1	1	0	1
3	0011	1	1	1	1	0	0	1
4	0100	0	1	1	0	0	1	1
5	0101	1	0	1	1	0	1	1
6	0110	1	0	1	1	1	0	1
7	0111	1	1	1	0	0	0	0
8	1000	1	1	1	1	1	1	1
9	1001	1	1	1	1	0	1	1

Pour réaliser le circuit 7448, vous devrez déterminer les expressions simplifiées des sorties a etc,. Pour cet atelier, Donnez seulement les équations simplifiées de la sortie a et c. Utiliser les tableaux de karnaugh.

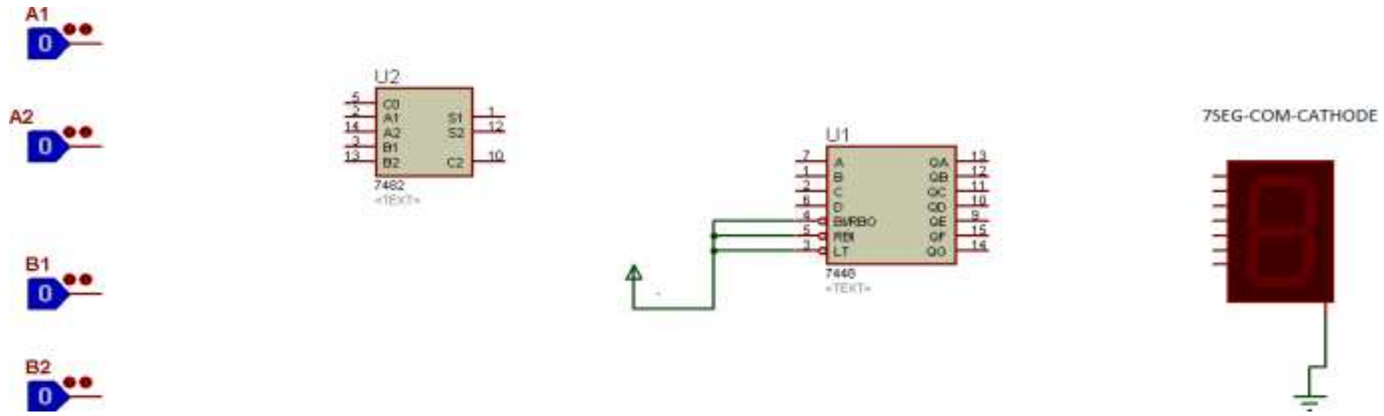
$$\bar{a} = \bar{D}\bar{C}\bar{B}A + \bar{D}C\bar{B}\bar{A} \Rightarrow a = \overline{\bar{D}\bar{C}\bar{B}A + \bar{D}C\bar{B}\bar{A}} = \overline{\bar{D}\bar{C}\bar{B}A} \cdot \overline{\bar{D}C\bar{B}\bar{A}} = D \cdot A$$

$$\bar{b} = \bar{D}C\bar{B}A + \bar{D}CB\bar{A} \Rightarrow b = \overline{\bar{D}C\bar{B}A + \bar{D}CB\bar{A}} = \overline{\bar{D}C\bar{B}A} \cdot \overline{\bar{D}CB\bar{A}} = D\bar{C}$$

$$\bar{c} = DC\bar{B}A$$

3ème partie câblage sur ISIS

Complétez le montage suivant et câblez le sur ISIS.



Atelier n°3 : ' Additionneur complets 2 bits '

I _ Partie théorique :

1ère partie : additionneur binaire :

1/

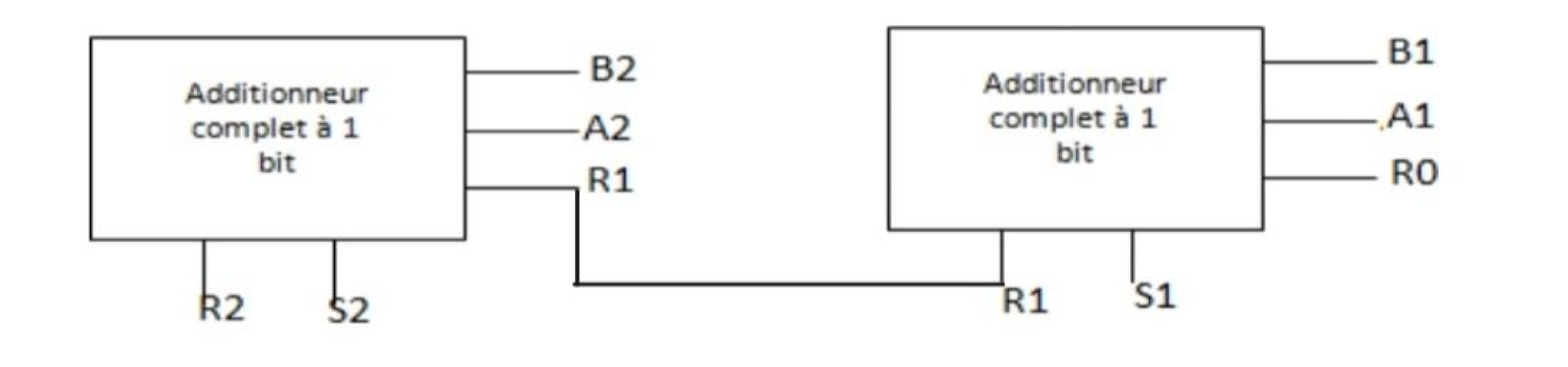
Tableau 1 : *Table de vérité de l'additionneur complet à 1 bit*

Ai	Bi	Ri-1	Ri	Si
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

$$R_i = \overline{A_i}.B_i.R_{i-1} + A_i.\overline{B_i}.R_{i-1} + A_iB_i\overline{R_{i-1}} + A_i.B_iR_{i-1}$$
$$= R_{i-1}(\overline{A_i}.B_i + A_i\overline{B_i}) + A_iB_i(\overline{R_{i-1}} + R_{i-1})$$
$$= R_{i-1}(A_i \oplus B_i) + A_iB_i$$

$$S_i = \overline{A_i}\,\overline{B_i}\,R_{i-1} + \overline{A_i}\,B_i\overline{R_{i-1}} + A_i\overline{B_i}\,\overline{R_{i-1}} + A_iB_iR_{i-1}$$
$$= \overline{R_{i-1}}(\overline{A_i}B_i + A_iB_i) + R_{i-1}(\overline{A_i}\,\overline{B_i} + A_iB_i)$$
$$= \overline{R_{i-1}}(A_i \oplus B_i) + R_{i-1}$$
$$= R_{i-1} \oplus A_i \oplus B_i$$

2/



2ème partie : afficheur 7 segments

Tableau 1 : *Table de vérité de décodeur '7448'*

Chiffres	D C B A	a	b	c	d	e	f	g
0	0000	1	1	1	1	1	1	0
1	0001	0	1	1	0	0	0	0
2	0010	1	1	0	1	1	0	1
3	0011	1	1	1	1	0	0	1
4	0100	0	1	1	0	0	1	1
5	0101	1	0	1	1	0	1	1
6	0110	1	0	1	1	1	0	1
7	0111	1	1	1	0	0	0	0
8	1000	1	1	1	1	1	1	1
9	1001	1	1	1	1	0	1	1

Pour a :

$$a = D + B.\overline{D} + AC\overline{D} + \overline{A}\,\overline{B}\,\overline{C}$$

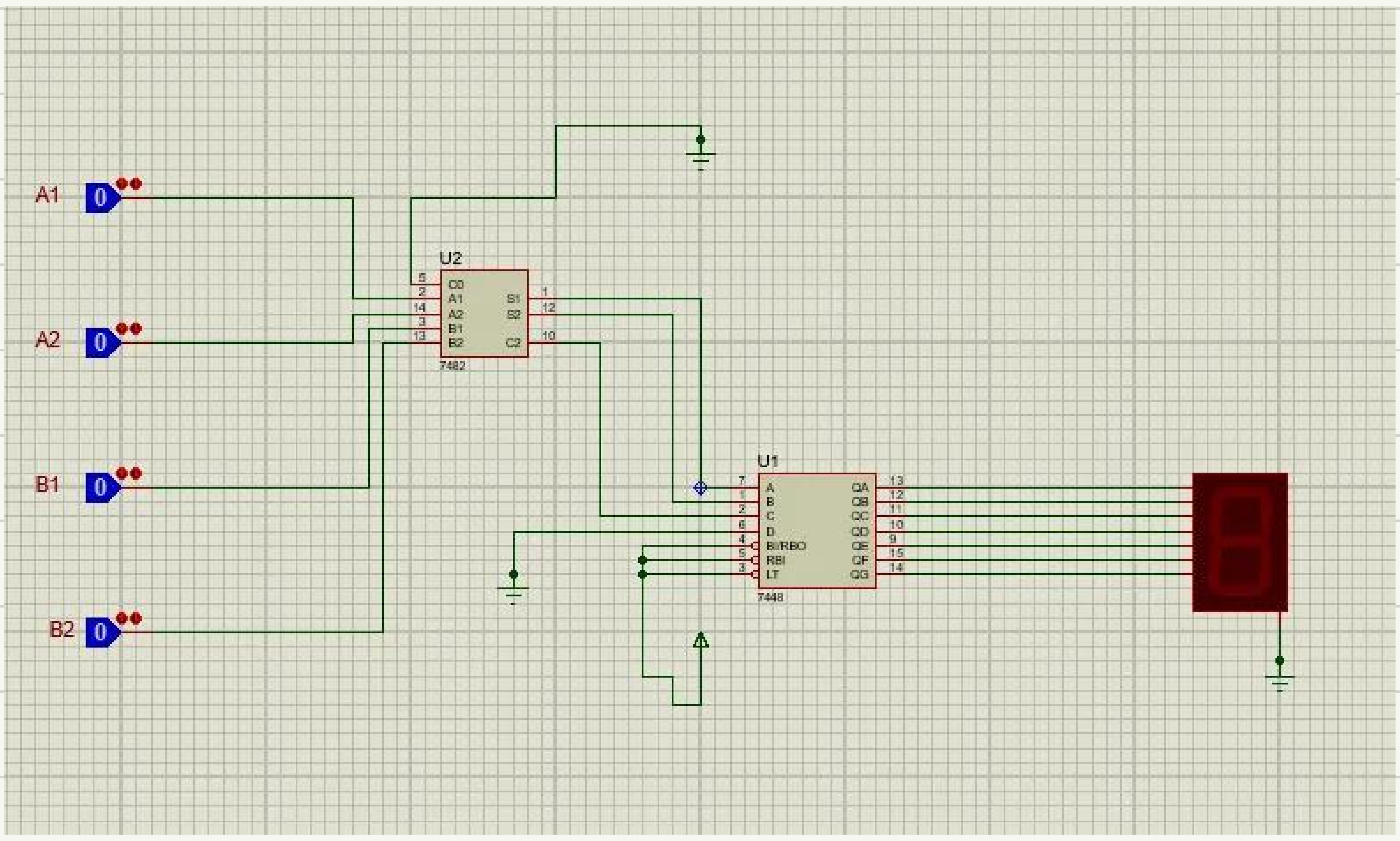
DC \ BA	00	01	11	10
00	1	0	1	1
01	0	1	1	1
11	1	1	1	1
10	1	1	1	1

Pour c :

$$c = \overline{B} + BD + BC\overline{D} + AB\overline{D}$$

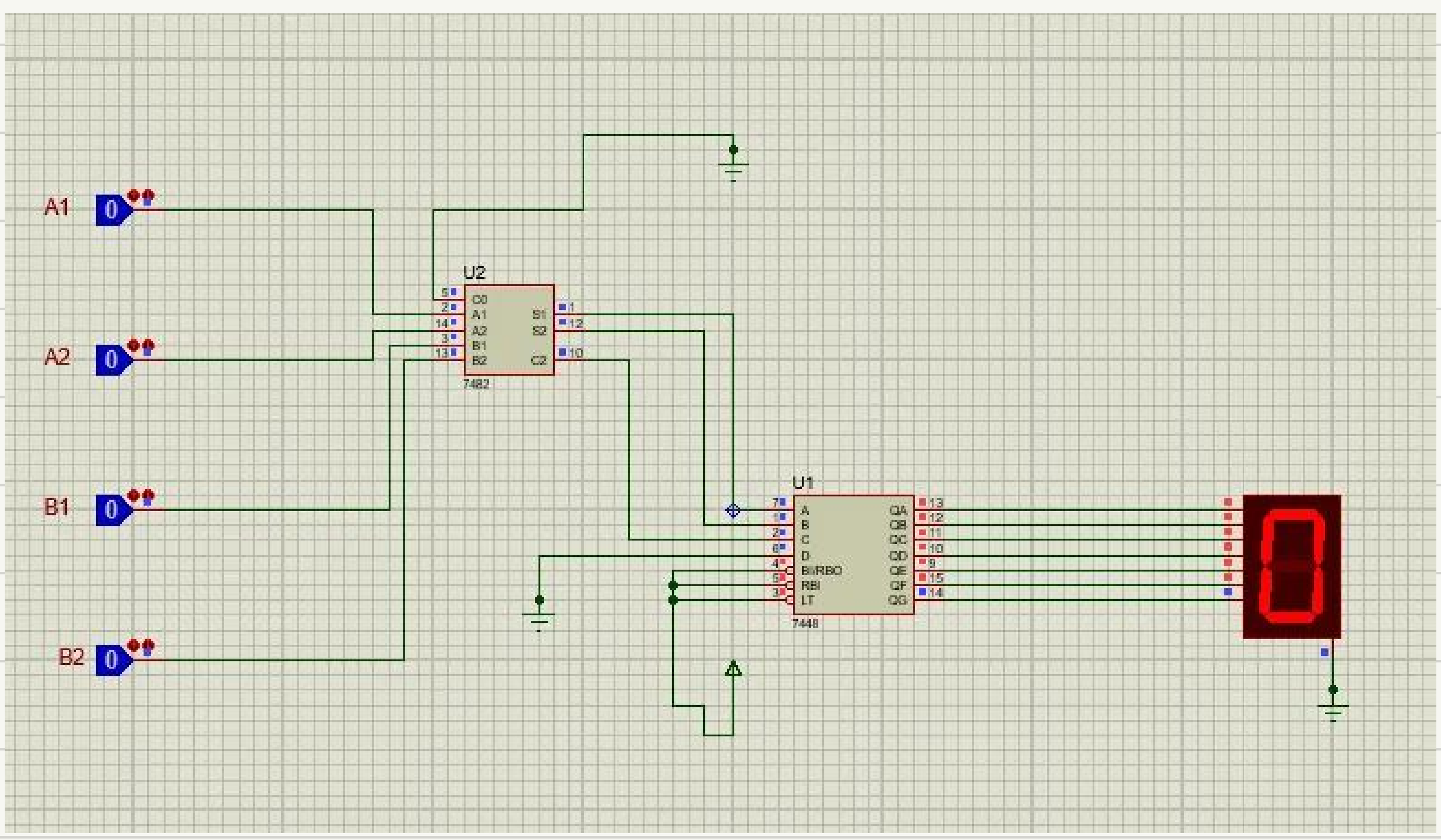
DC \ BA	00	01	11	10
00	1	1	1	0
01	1	1	1	1
11	1	1	1	1
10	1	1	1	1

II - Partie pratique :

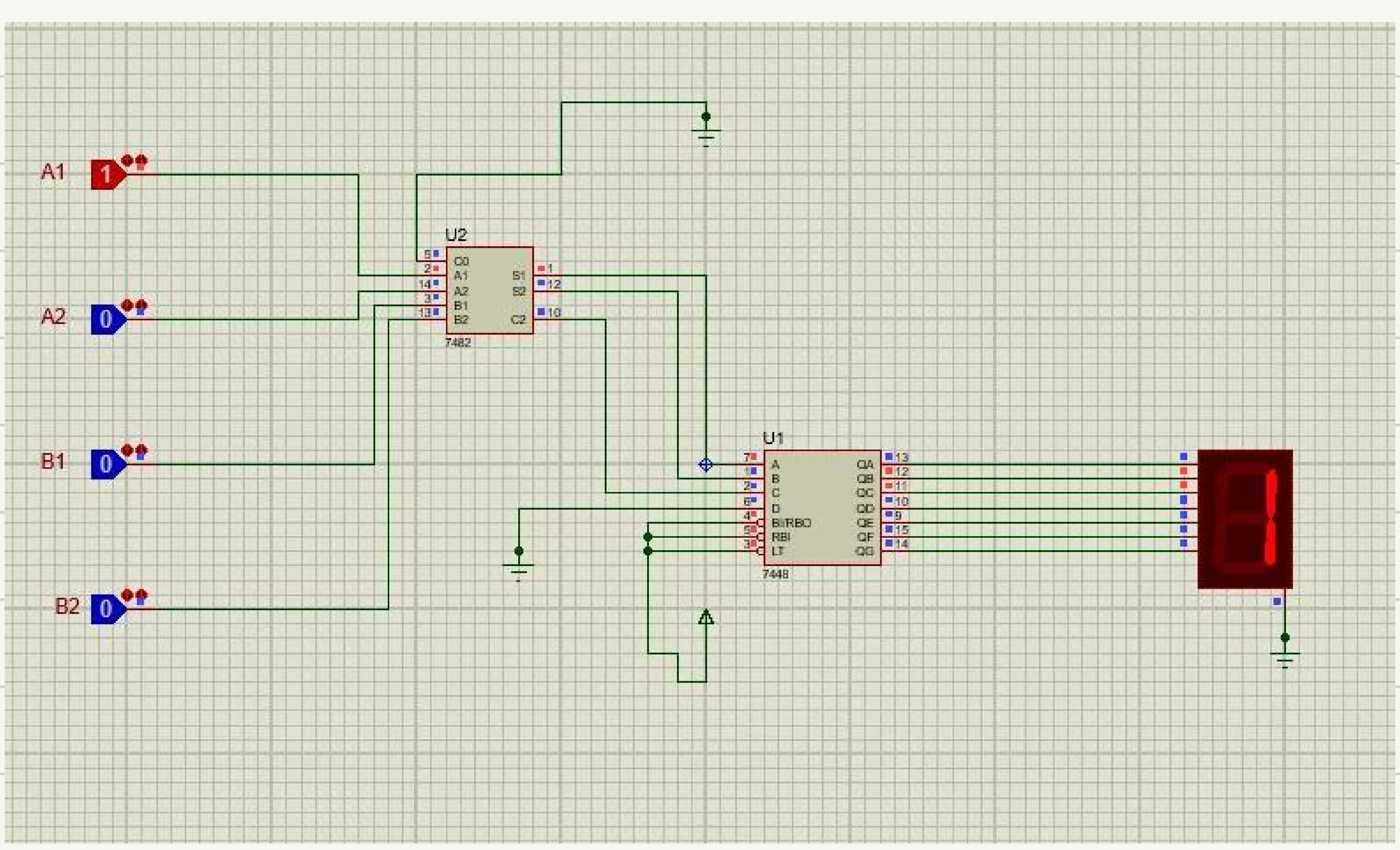


Cablage d'un afficheur 7 segment

affichage de "0"



affichage de "1"



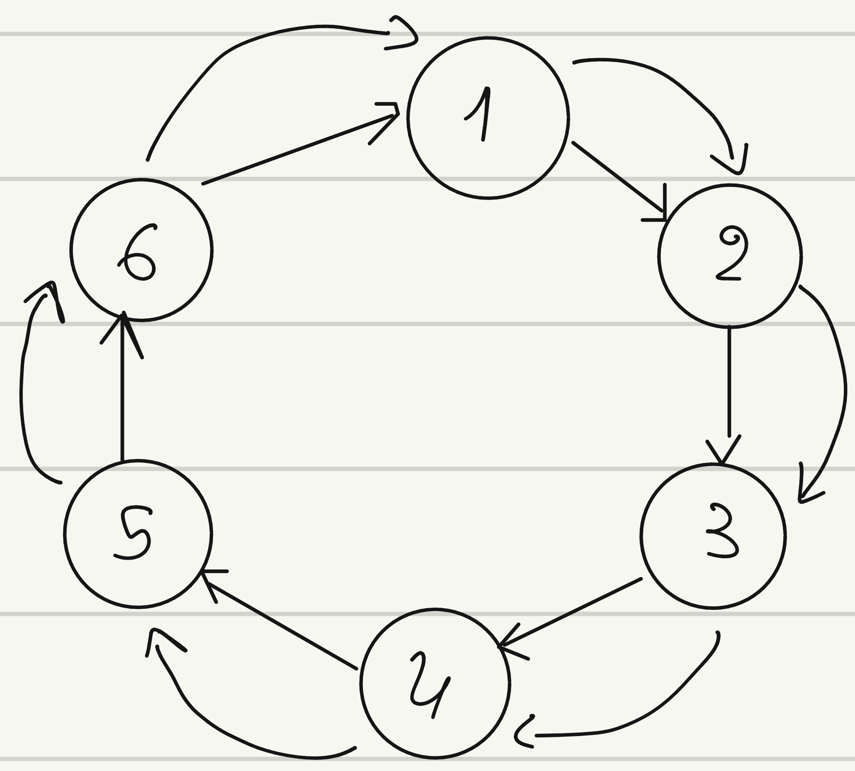
Atelier n° 4

Chronomètre 60 s

I - Partie 1 : compteur asynchrone des unités de dizaines

a/ Partie théorique :

1/



cycle de 0 → 5

$V_{max} = 5 \Rightarrow$ compteur modulo 6

$(5)_{10} = (101)_2$

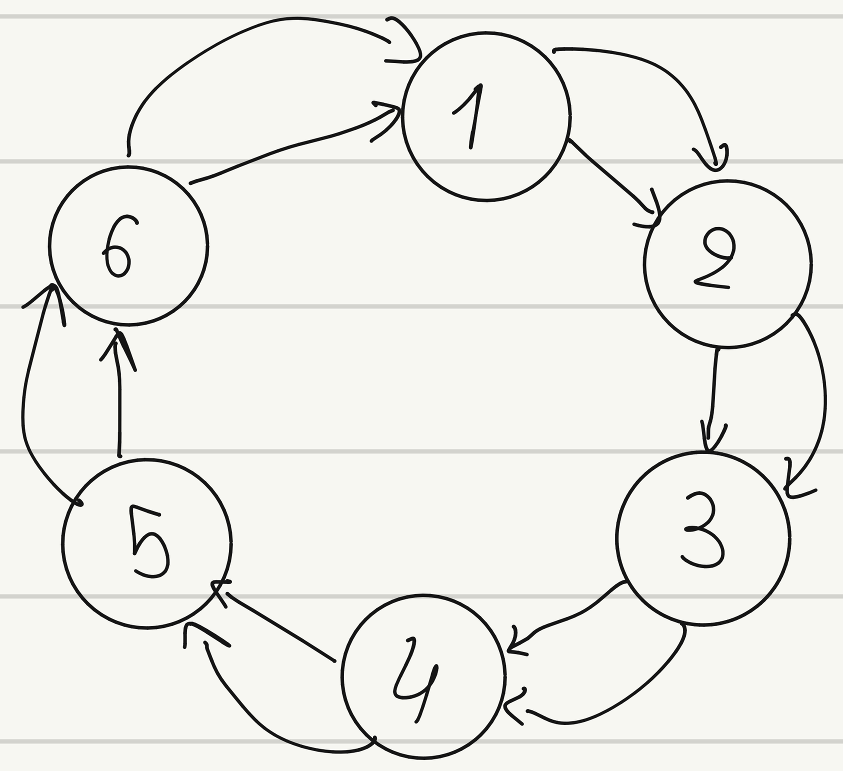
Trois bits pour écrire 5.

$\Rightarrow$ Nombre de bascule : 3

2/

Q2	Q1	Q0		Q2+	Q1+	Q0+		Pr	Cl
0	0	0		0	0	1		0	0
0	0	1		0	1	0		0	0
0	1	0		0	1	1		0	0
0	1	1		1	0	0		0	0
1	0	0		1	0	1		0	0
1	0	1		1	1	0		0	0
1	1	0		0	0	0		0	1

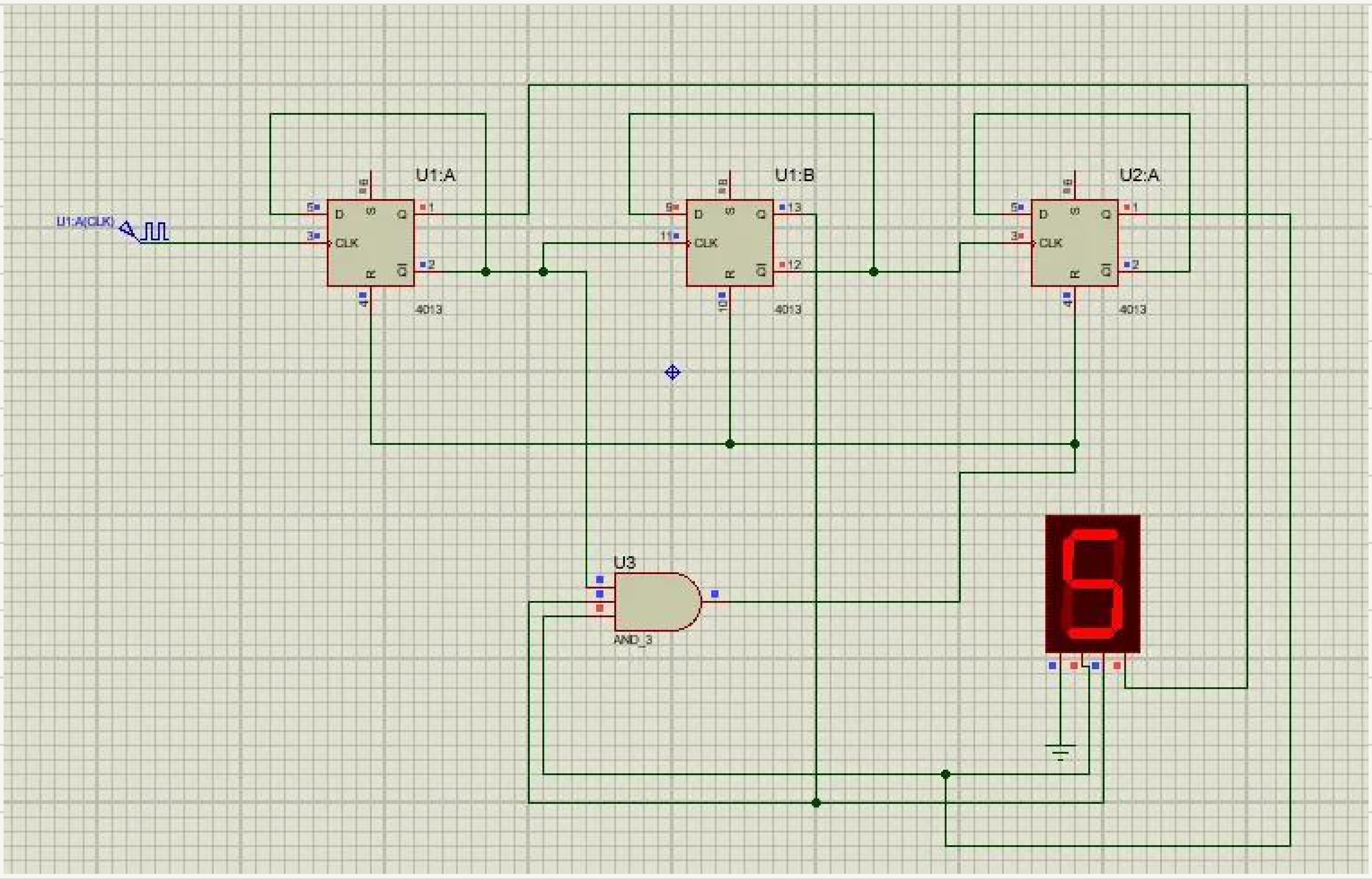
3/



$Cl_i = Q2\,Q1\,.\,\overline{Q0}$

$Pr_i = 0$

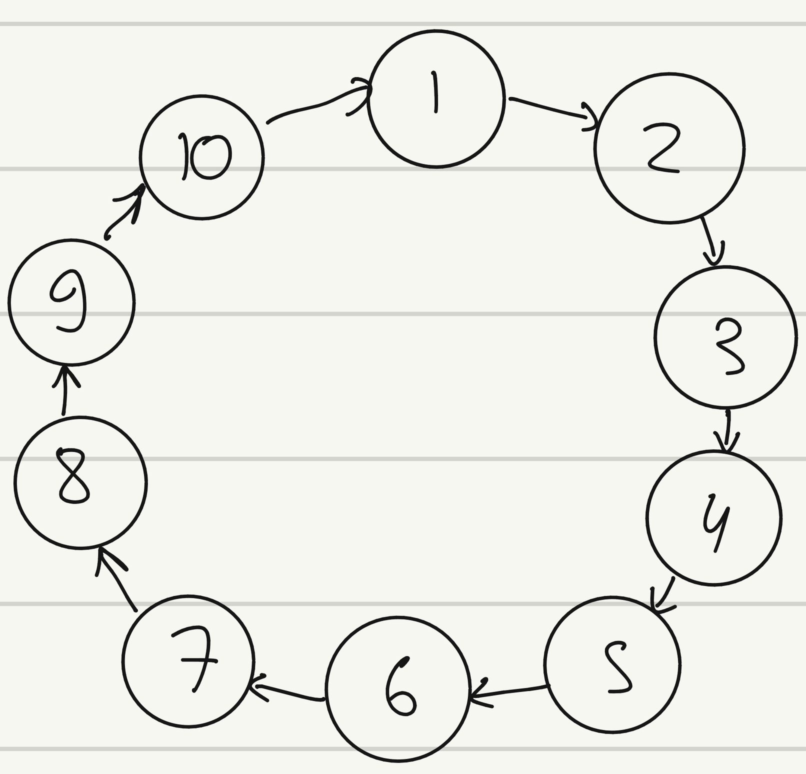
b/ Partie pratique :



II _ Partie 2 : compteur asynchrone des dizaines de secondes :

a/ Partie théorique :

1/



cycle $0 \longrightarrow 9$

$V_{max} = 9$

$\Rightarrow$ compteur modulo 10

$(9)_{10} = (1001)_2$

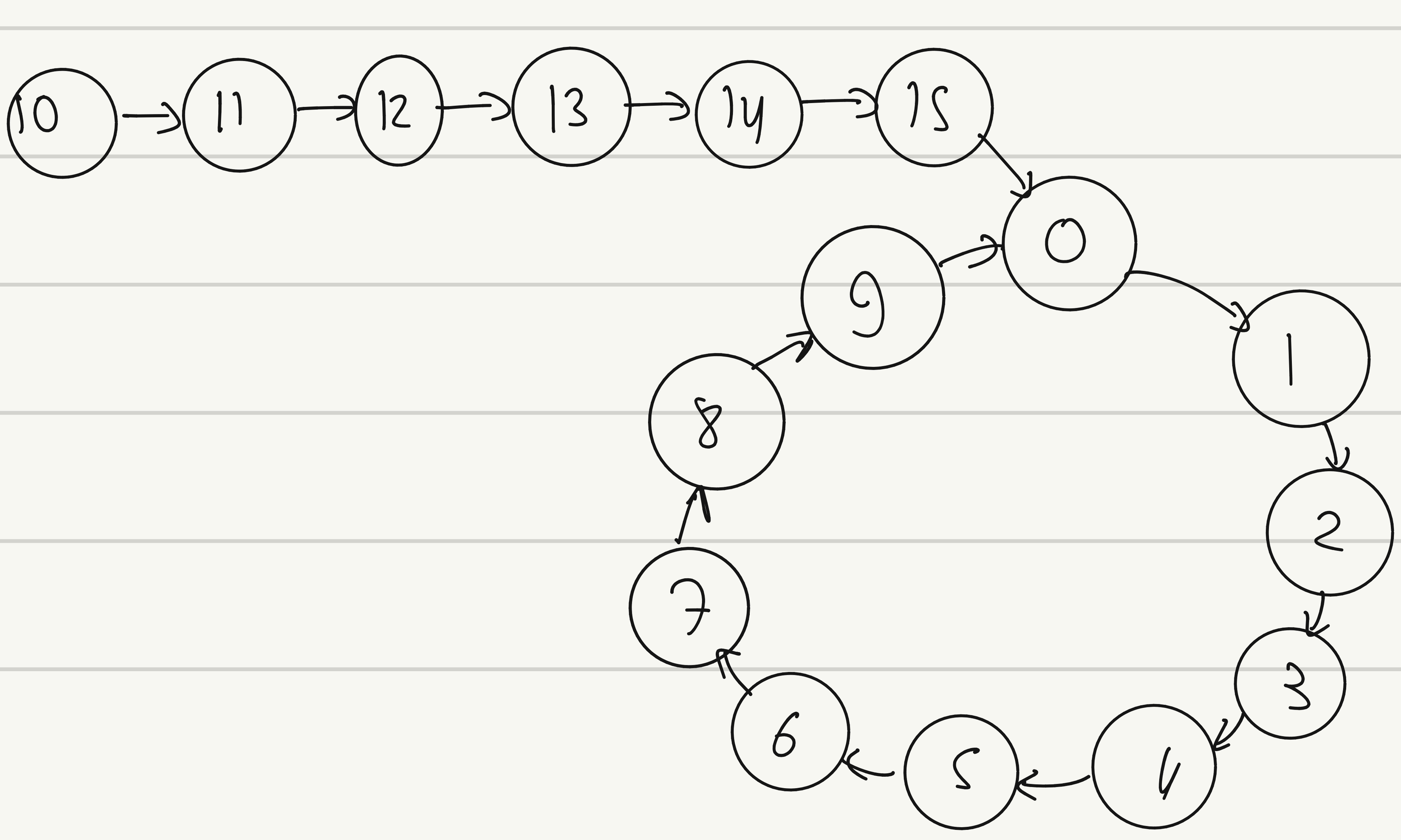
4 bits pour écrire 9

$\Rightarrow$ Nombre de bascule = 4

2/

$Q3$	$Q2$	$Q1$	$Q0$	Q_3+	Q_2+	Q_1+	Q_0+	Cl	Pr
1	0	1	0	0	0	0	0	1	0

3/



$Cl_i = Q3 \cdot \overline{Q2} \cdot Q1 \cdot \overline{Q0}$

$Pr_i = 0$

b/ Partie Pratique :