Cours sur le binaire

Note préliminaire :

Pour mieux comprendre ce cours vous pouvez utiliser le simulateur logique JFLogSim, diffusé en shareware (50 séance en libre essai) sur mon site : www.flucas.com

Numération

Travaux pratiques de numération

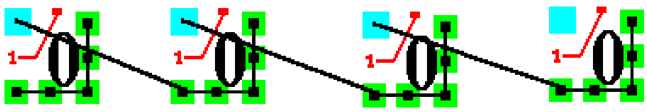
- Couper une feuille A4 en quatre morceaux. Réaliser 4 cartes, sur une des faces dessiner un grand 1, sur l'autre un 0.
- les aligner pour obtenir un vecteur de 4 digits (binaires)

Manipulation avec ces 4 cartes :

- Énumérer tous les états possibles de ce compteur depuis 0000 jusqu'à 1111.
- Il faut en trouver 16.
- 0000, 0001, 0010, 0011, 0100 …

Manipulation avec le simulateur logique :

Câbler ce montage, et en cliquant sur le plot bleu de l'interrupteur de droite, vérifier que le système décrit 16 états de 0000 à 1111.



Conversion binaire décimal

Notion de base.

Autres bases de numération.

L'homme ne compte pas seulement en base 10. Voici quelques autres bases :

- la base binaire 2.
- Base 7, pour les jours de la semaine et dans la Bible.
- Base 8 sur le PDP de keneth Olsen de DEC.
- Base 12 pour les mois, les heures et dans l'alimentation. Pour moi c'est la base la plus belle.
- Base 60 pour les minutes et les secondes.

Notion de poids binaire

- En base 10 on a :

. Centaines, Dizaine, Unité, ce qui correspond à :
. 10^2, 10^1, 10^0.
. $Base^2$, $Base^1$, $Base^0$.
. a_2, a_1, a_0.
. 1, 2, 3.
. 1.100+2.10+3.1=123

- En base 2 on a :

. Unité deuzaine quatraine huitaines
. 2^3, 2^2, 2^1, 2^0.
. $Base^3$, $Base^2$, $Base^1$, $Base^0$.
. a_3, a_2, a_1, a_0.
. 1, 0, 1, 1.
. 1.8 +0.4 +1.2 +1.1.

- Et de façon générale on a la notation mathématique correspondante :

Soit V, la valeur d'un vecteur à N digits dans une base B donnée :
$V = \Sigma_{\text{(pour i variant de 0 à N)}} a_i.Base^i$

Combinatoire d'un vecteur binaire

Calcul de la combinatoire :

Un vecteur binaire de N digits a une combinatoire de 2^N.
Le nombre de combinaisons possibles pour un vecteur binaire de N digits est de 2^N.
Exemple : un vecteur binaire de 4 digits peut se présenter sous $2^4 = 16$ combinaisons possibles.

Fonctions logiques

Intro des boîtes noires

Elles sont vues seulement sous cet angle :
- Un vecteur de N digits en entrée.
- Un vecteur de M digits en sortie.
- Une table pour décrire exhaustivement son fonctionnement en terme de stimuli-réponse

Combinatoire des boîtes noires

- Combinatoire en entrée.
- Combinatoire en sortie.
- Notion de table de vérité.

Fonctions logiques à 0 entrée

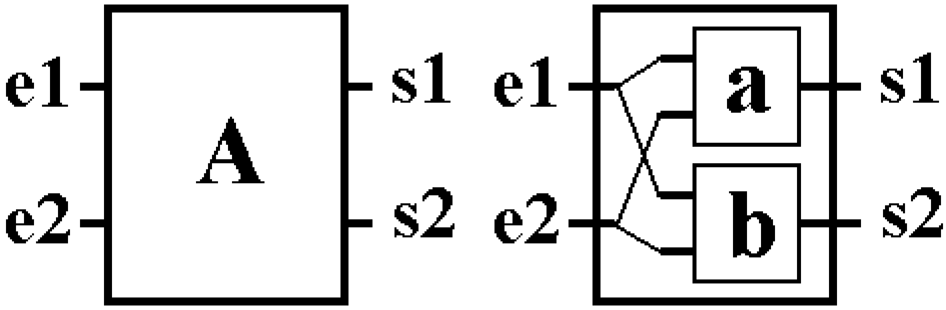
Si ces boîtes n'ont pas d'entrée, leur fonctionnement est forcément stable.

1 bit de sortie => 2 états possibles en sortie.

1 : Générateur

0 : Masse

Fonctions logiques à 2 sorties



Une fonction logique A, à 2 sorties s1 et s2, peut toujours se ramener à la superposition de 2 fonctions logiques (a et b) à une sortie (s1 pour a et s2 pour b) et qui ont les mêmes entrées (e1 et e2).

Donc nous étudierons seulement les fonctions logiques à une sortie.

Le même raisonnement est faisable avec une fonction logique à N sorties.

Fonctions logiques à 1 entrée

1 bit de sortie :

- Dans une première approche brute, on dégage 4 possibilités de fonctionnement :

E1	S0	S1	S2	S3
0	0	1	0	1
1	0	0	1	1

- Mais rapidement on aboutit à 2 possibilités nettes :

E1	S1	S2
0	1	0
1	0	1

Car nous avons déjà rencontré les fonctionnements S0 et S3, qui correspondent à une sortie constemment à 1 pour S3 et constamment à 0 pour S0.

Étude du fonctionnement S2 : Suiveur.

E1	S2
0	0
1	1

- Semble servir à rien, mais c'est un relais, un amplificateur de courant.
- Un suiveur rebouclé fait une mémoire.
- Forçage d'un suiveur.
- Notion de court circuit.
- La durée du court-circuit lors d'un forçage est celle du temps de réponse du composant.

Étude du fonctionnement S1 : Inverseur.

E1	S1
0	1
1	0

- Un suiveur rebouclé fait un oscillateur.
- Temps de réponse et fréquence d'oscillation.

Fonctions logiques à 2 entrées

2 entrées, une sortie.

E1	E2	S1
0	0	
0	1	
1	0	
1	1	

- 4 combinaisons en entrée.
- Pour chacune de ces 4 combinaisons je dois fournir une sortie. J'ai donc un vecteur de 4 bits : 2^4= 16 fonctions possibles
- Nous focaliserons seulement sur 3 de ces circuits logiques

ET

- La sortie S1 est à 1 quand l'entrée E1 ***et*** l'entrée E2 est à 1.

E1	E2	S1
0	0	0
0	1	0
1	0	0
1	1	1

OU

- La sortie S1 est à 1 quand l'entrée E1 ***ou*** l'entrée E2 est à 1. Si les deux entrées sont à 1 les entrées se renforcent mutuellement.

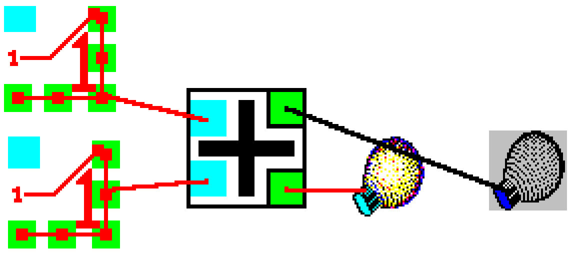
E1	E2	S1
0	0	0
0	1	1
1	0	1
1	1	1

OUEX

- La sortie S1 est à 1 quand l'entrée E1 ***ou*** l'entrée E2 est à 1. Si les deux entrées sont à 1 les entrées s'excluent mutuellement.

E1	E2	S1
0	0	0
0	1	1
1	0	1
1	1	0

Additionneur 2bits



Synoptique :

- 2 bits en entrée : 4 combinaisons.
- 2 bits en sortie car 1+1=10

Énumération de la combinatoire du vecteur d’entrée

E1	E2
0	0
0	1
1	0
1	1

Pour chaque combinaison, on remplit la table de vérité

E1	E2	S1	S0
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	0

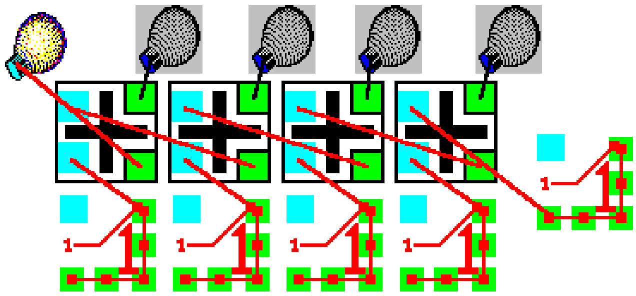
Identification de la solution :

- S1 est un ET.
- S2 est un OUEX.

Incrémenteur

Fonctionnement :

- De façon générale, il s’agit d’additionner 1 à un nombre de N bits.
- Dans le cas présent, il s’agit d’additionner 1 à un nombre de 4 bits que l’on entre sur les interrupteurs. On peut lire $1111_{(2)}$ soit $15_{(10)}$.
- La sortie se fait sur les afficheurs lumineux, on lit : $10000_{(2)}$ soit $16_{(10)}$.
- Note : Mais ce résultat est sur 5 bits. En toute rigueur, on lit : $0000_{(2)}$ soit $0_{(10)}$.

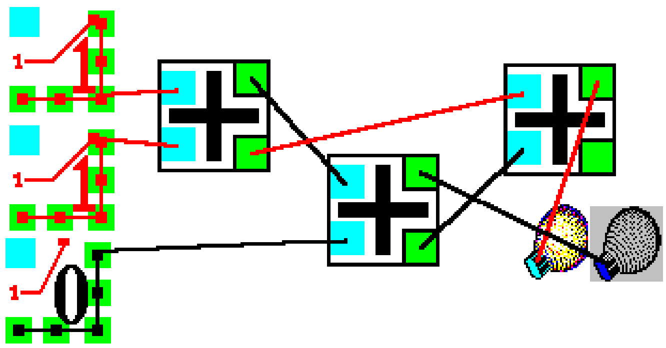


Additionneur 3bits

Synoptique :

- 3 bits en entrée : 8 combinaisons.
- 2 bits en sortie car 1+1+1=11

Câblage



Explication :

- Avec le premier additionneur j'additionne les unités.
- Je mets de côté le résultat deuzaine.
- Le résultat unité est encore une unité : je l'additionne avec la dernière unité existante non traitée.

Le résultat unité obtenu est le résultat unité définitif.

Le résultat deuzaine obtenu est additionné avec le résultat deuzaine mis de côté et donne le résultat deuzaine définitif.

Addition : le problème des virgules

En base 10, les chiffres après la virgule ont un poids de 1/10, 1/100, 1/1000… Ceci correspond à une division par la base 10 à chaque fois qu'on se décale vers la droite.

En base 2 les chiffres après la virgule ont un poids de 1/2, 1/4, 1/8… Ceci correspond aussi à une division par la base à chaque fois qu'on se décale vers la droite, car ici la base est 2.

En gardant présent à l'esprit cette convention, on peut parfaitement traiter les chiffres après la virgule selon le même processus.

Complément à 1

Définition 1

Sur un digit, le complément à 1 C1 d'un chiffre A est tel que A+C1=Base-1.

Le complément à 1 en base 10 de 2 est 7, car 2+7 = 10-1 = 9.

Définition 2

Soit en base 10 un compteur sur 2 digits (de 0 à 99). Le complément à 1 C1 d'un chiffre A est tel que A+C1=Base^2 - 1.

Le complément à 1 en base 10 de 42 est 57, car 42+57 = 100-1 = 99.

Note 1

- Le complément à 1 d'un nombre de plusieurs digits se calcule très facilement en concaténant le complément à 1 de chaque digit.
- Le complément à 1 en base 10 de 42 est 57, car :

. Le complément de 4 est 5.

. Le complément de 2 est 7.

Note 2

En binaire, sur un digit, le complément à 1 d'un digit se calcule très facilement :

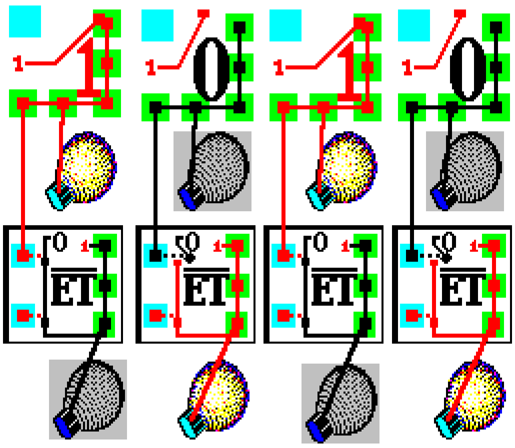
Le complément à 1 de 0 = 1.

Le complément à 1 de 1 = 0.

Il suffit d'inverser le chiffre.

Note 3

- En binaire, le complément à 1 d'un nombre de plusieurs digits se calcule très facilement en concaténant l'inversion de chaque digit.



- Sur ce schéma, le nombre en entrée est construit en haut sur les interrupteurs.
- Il est visualisé en dessous sur les afficheurs lumineux.
- Les NonEt fonctionnent comme des inverseurs.
- Le résultat est disponible et affiché en bas.

C2 : le complément à 2

Définition :

- Le complément à 2 est une incrémentation du complément à 1.
- C2 = C1+1.
- Complément à 1 + 1 = complément à 2.
- Exemple :

. Le complément à 1 en base 10 de 2 est 7, car 2+7 = 9.

. Le complément à 2 en base 10 de 2 est 8, car 1+7 = 8.

Conséquence

Soit un nombre A de n digits, C1 son complément.

On sait que :

(1) $A+C1=Base^n - 1$. | Définition déjà vu précédemment

(2) $C1+1= C2$ | Une des définitions du complément à 2

A partir de (2) on tire (2') $C1= C2 – 1$. | En faisant passer le 1 de l'autre côté du égal

(1) $A + C1=Base^n - 1$ | Repartant de (1) on a :

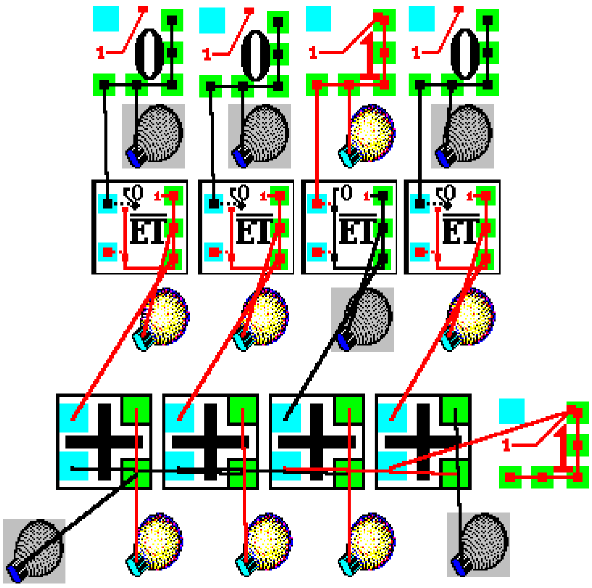
(1') $A + C2 – 1=Base^n - 1$ | En remplaçant C1 par sa valeur 'C2 – 1' dans (1) on a :

(1'') $A + C2 =Base^n$ | En supprimant '-1' de chaque côté

(1''') $C2 - Base^n = {}^{-}A$ | En passant 'Base' de l'autre côté

Explication du schéma :

De haut en bas on a :



- Les interrupteurs fournissent le nombre en entrée.
- Il est visualisé sur les ampoules.
- Les NonEt utilisés en inverseurs calculent le complément à 1.
- Lequel est visualisé sur les ampoules.
- Les additionneurs incrément le complément à 1 pour obtenir un complément à 2.
- Le résultat est disponible sur les ampoules tout en bas.
- Notez, en bas à gauche, la retenue qui est délaissée.

Additionneur de deux mots de N bits

Synoptique :

- 2 nombres de N bits en entrée.
- Un nombre de N+1 bits en sortie.

-- Exemple avec N = 4 :

Calculer la somme de 1010 + 0110, 2 nombres de 4 bits.

Dans ce cas la sortie est de 5 bits, car on peut être amené à additionner 1111+1111 = 11110.

Câblage

Soustraction

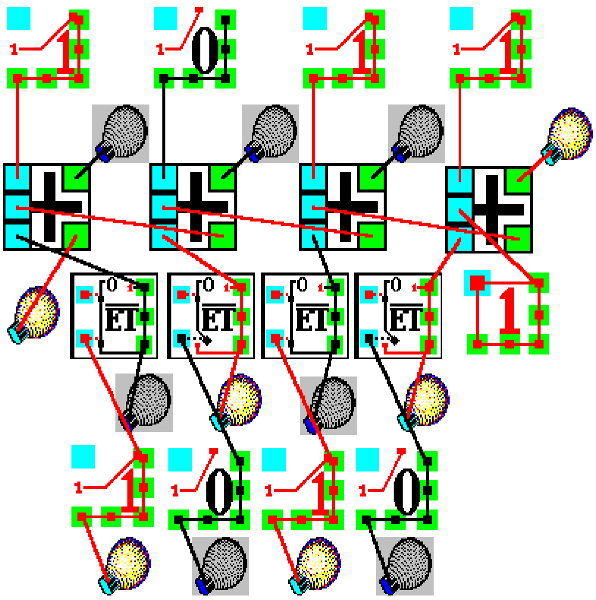
Application du complément à 2 au calcul de B - A

( ) B – A = B – A	| Partant de cette égalité, je peux écrire (3) qui est trivial.
(3) B – A = B + (- A)	| En utilisant (1''') C2 - Base^n = -A
(3') B – A = B + (C2 - Base^n)	| Dans la partie droite de (3), je remplace – A par C2 - Base^n
(3'') B – A = B + C2 - Base^n	| Je supprime les parenthèses
(3''') Base^n + B – A = B + C2	| Je passé Base^n de l'autre côté.

Mais Base^n c'est un 1 suivi de n 0, c'est en fait une retenue qu'on laisse tomber. Donc dans un nombre de n digit Base^n disparaît sans laisser de trace, sans perturber le calcul. On obtient :

(3'''') B – A = B + C2

Pour soustraire A à B il suffit de lui additionner son complément à 2.



Explication du schéma pour calculer B - A:

- Tout en haut les interrupteurs fournissent B.

Puis, de bas en haut on a :

- Tout en bas les ampoules visualisent A.
- En bas les interrupteurs fournissent A.
- Les ampoules visualisent le complément à 1 de A.
- Les NonEt calculent le complément à 1 de A.
- Les additionneurs calculent B – A : 11 –10 = 1
- Le résultat 0001 est affiché au-dessus des additionneurs.

Multiplication par 2

Elle se fait par décalage à gauche.
Le circuit peut être câblé avec de simples fils.

Division par 2

Elle se fait par décalage à droite.
Le circuit peut être câblé avec de simples fils.

Multiplication de 2 nombres binaires de N chiffres

Synoptique :

- 2 nombres de N bits en entrée.
- Un nombre de 2N bits en sortie.

Exemple avec N = 4 :

Calculer le produit de 1010 * 0110, 2 nombres de 4 bits.
Dans ce cas la sortie est de 8 bits, car on peut être amené à additionner 1111*1111 = 11100001.

Câblage

Division de 2 nombres binaires : A/B

Division par soustraction successives

Résu= 0;
A = 1234;
B= 5;
Tant que A>B faire :
- A := A-B;
- Résu := résu + 1;

Division

Conversion décimal binaire

Sur 8 ou 16 bits avec un dessin.

Classique par division.

J'ai fait un poly là-dessus. Ceux qui sont intéressés peuvent le trouver sur le site :
70_ElectricBoucleDiodeEtOuPasCodeMuxRomBooleOuExAddIncCompSub.doc.