ADO, TD n°2

Exercice 1

L’additionneur est composé de :

* 2 registres à décalages a (5 bits) et b (4bits) qui gardent les opérandes. Le signal new permet de charger une valeur dans le ebit 4.
* Un additionneur
* Un registre c pour conserver la retenue (raz permet de la remettre à zero)
* Compteur 2 bits (00 à 11) avec un signal raz et un signal d’horloge clock

Initialisation :

* Entrée des opérandes a (3..0) et b (3..0)
* Mise à zéro de c
* Mise à zéro du compteur

Un cycle de fonctionnement :

* Addition des bits de poids faible ; positionnement de la retenue et du résultat à la sortie de l’additionneur
* Le signal new charge le résultat dans la bit a(4)
* Décalage des bits
* Incrémentation du compteur

Voici ce que donnera l’addition de a=6 et de b=3 :

6+3

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Compteur** | **a(4..0)** | **b(3..0)** | **Retenue** | **Etat** |
| 00 | X0110 | 0011 | 0 | init |
| **00** | **X0110** | **0011** | **0** | **add** |
| 00 | 10110 | 0011 | 0 | new |
| 00 | X1011 | X001 | **0** | dec |
| **01** | **X1011** | **X001** | **1** | **add** |
| 01 | 01011 | X001 | 1 | new |
| 01 | X0101 | XX00 | **1** | dec |
| **10** | **X0101** | **XX00** | **1** | **add** |
| 10 | 00101 | XX00 | 1 | new |
| 10 | X0010 | XXX0 | **1** | dec |
| **11** | **X0010** | **XXX0** | **0** | **add** |
| 11 | 10010 | XX00 | 0 | new |
| 11 | X1001 | XXXX | **0** | dec |

6+11

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Compteur** | **a(4..0)** | **b(3..0)** | **Retenue** | **Etat** |
| 00 | X0110 | 1011 | 0 | init |
| **00** | **X0110** | **1011** | **0** | **add** |
| 00 | 10110 | 1011 | 0 | new |
| 00 | X1011 | X101 | **0** | dec |
| **01** | **X1011** | **X101** | **1** | **add** |
| 01 | 01011 | X101 | 1 | new |
| 01 | X0101 | XX10 | **1** | dec |
| **10** | **X0101** | **XX10** | **1** | **add** |
| 10 | 00101 | XX10 | 1 | new |
| 10 | X0010 | XXX1 | **1** | dec |
| **11** | **X0010** | **XXX1** | **1** | **add** |
| 11 | 00010 | XX00 | 1 | new |
| 11 | X0001 | XXXX | **1** | dec |

Il y a overflow ssi à la fin du calcul, la retenue (c) est à 1.

Exercice 2

2+4

0010

+ 0100

= 0110

(6)

4+5

 0100

+ 0101

= 1001

(-7) => overflow

(-2) + (-4)

 1110

+ 1100

= *1|*1010

(-6)

(-4)+(-5)

 1100

+ 1011

= 1|0111

(7)=>overflow

Il y a ‘overflow’ lorsque les deux dernières retenues ont des valeurs différentes :

$$Overflow=c\_{n}⊕c\_{n-1} $$

Exercice 3

La génération de la retenue est donnée par $g\_{i}=a\_{i}.b\_{i}$

La propagation de la retenue est donnée par $p\_{i}=a\_{i}⊕b\_{i}$

La retenue est donnée par $c\_{i}=g\_{i}+\left(p\_{i}.c\_{i-1}\right)$

La somme est donnée par $s\_{i}=p\_{i}⨁c\_{i-1}$

Ecriture de l’équation qui lie les variables :

$$\left\{\begin{matrix}s\_{0}=p\_{0}⊕c\_{e}\\c\_{0}=g\_{0}+\left(p\_{0}.c\_{e}\right)\end{matrix}\right\}$$

$$\left\{\begin{matrix}s\_{1}=p\_{1}⊕\left(g\_{0}+\left(p\_{0}.c\_{e}\right)\right)\\c\_{1}=g\_{1}+\left(p\_{1}.\left(g\_{0}+\left(p\_{0}.c\_{e}\right)\right)\right)\end{matrix}\right\}$$

$$\left\{\begin{matrix}s\_{2}=p\_{2}⊕\left(g\_{1}+\left(p\_{1}.\left(g\_{0}+\left(p\_{0}.c\_{e}\right)\right)\right)\right)\\c\_{2}=g\_{2}+\left(p\_{2}.\left(g\_{1}+\left(p\_{1}.\left(g\_{0}+\left(p\_{0}.c\_{e}\right)\right)\right)\right)\right)\end{matrix}\right\}$$

Pour voir le résultat, référence la correction page 5 sur 10.

Exercice 4

$$448\_{10}=1 1100 0000\_{2}$$

$$17\_{10}=10001\_{2}$$

* ***1 1100*** 0000/ ***10001*** = 1 + 0 1011 0000/10001
* ***1011 0***000 / ***10001*** = 1 + 0010 1000/10001
* ***010 10***00 / ***10001*** = 0
* ***10 100***0 / ***10001*** = 1 + 00 0110/10001
* ***0 0110*** / ***10001*** = 0