

Exercice 3.

Exo 3: $\frac{03,28}{05}$

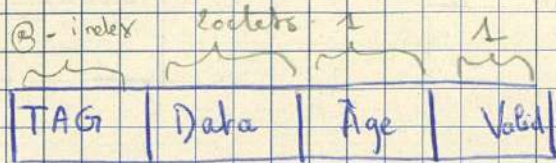
1) La capacité cache est de :

$$4 * 2 * 8 = 64 \text{ octets.}$$
$$= 2^6 \text{ octets.}$$



2) ~~La capacité cache est de :~~ ~~Index de la Requête, donnée~~

3)



TAG : Permet de vérifier l'adresse mémoire complète.

Data : Permet de stocker la donnée.

Age : Depuis quand la donnée est présente ; si elle l'est déjà.

Validation : Bits de Validité, pour savoir si la donnée est valide ou pas.

4) Le cache a correspondance direct place le mot en mémoire à l'adresse ou on lui demande. Le problème, c'est que on ne peut pas faire autrement.

Le cache associatif permet une rapidité supplémentaire, mais ~~au même~~ si deux données ont le même index on peut avoir un ~~cache~~ cache hit même si cette donnée n'est pas en mémoire.

5) La propriété qui permet cela est la propriété spatiale. Car le programme s'exécute dans un flux qui peut être interrompu d'interceptions. L'exécution linéaire, c'est à dire que si on accède à une instruction suivante sans attendre.



NOM BLOQUET

Prénom Romain

Promo 2018

Date 03/06/2015



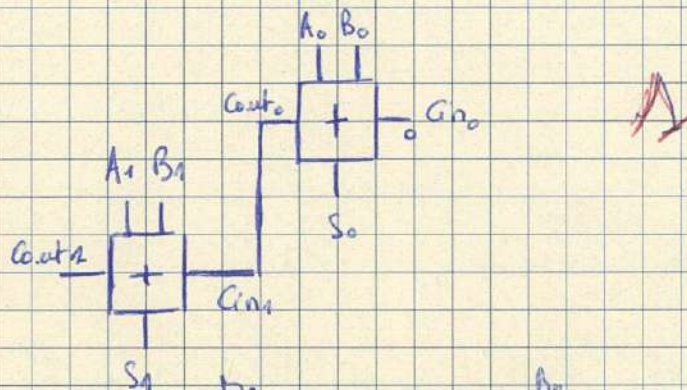
BLOQUET Romain
PL2 - 2014

MATIÈRE ADO

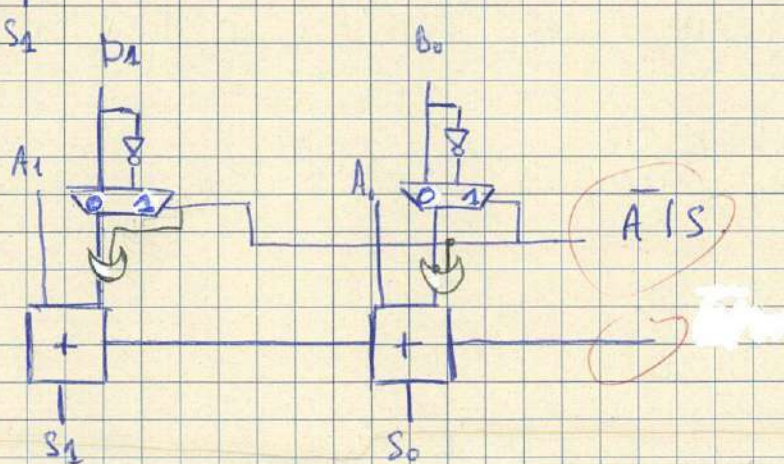
Exercice 1:

Il faut relier deux additionneurs 1 bits pour faire un additionneur complet à deux bits. A l'étage de poids faible, aucune retenue ne sera extraite, car aucune addition n'a encore été faite.

Exo 1: 0,45
05



Exercice 2:



Au Multiplexeur On Ne Branche que B et B Car seule le ligne de B change (A+B) et (A-B).

Exercice 3:

Si $C_1 = 0$ alors addition ou soustraction
Si $C_1 = 1$ alors sélection de A ou B.

Exercice 2

Exp2: $\frac{0,15}{0,5}$

1.

Broches adresses : 8

Broches données : 27

1

2.

Bus adresses : 8

Bus données : 28

0,5

3.

~~Bus adresses : 8~~

~~Bus données : 14~~

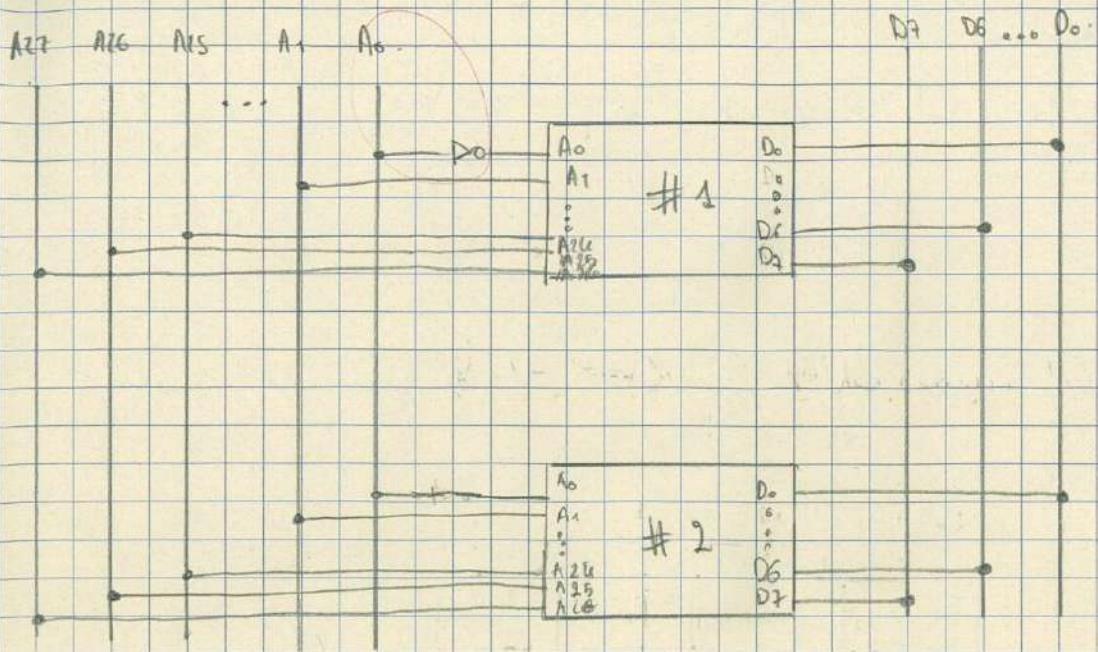
0

4.

Bus adresses : 8

Bus données : 14

0



$0x AB = 1010 \ 1011_2$

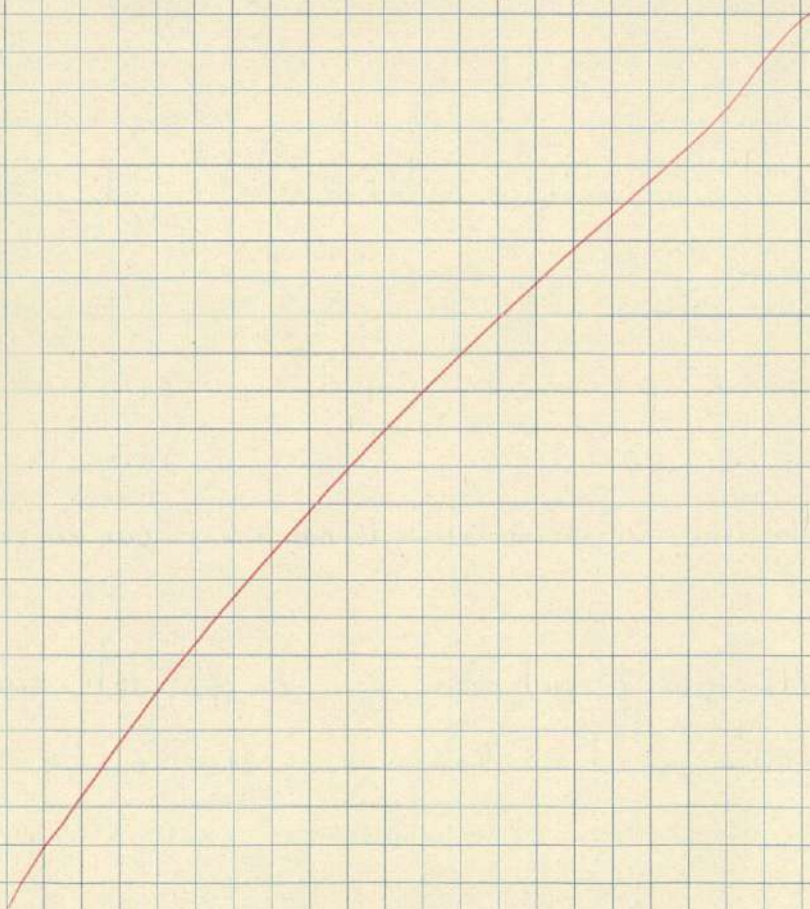
Bit de poids faible égal a 1
alors il sera stocké dans la barrette
numero 1.

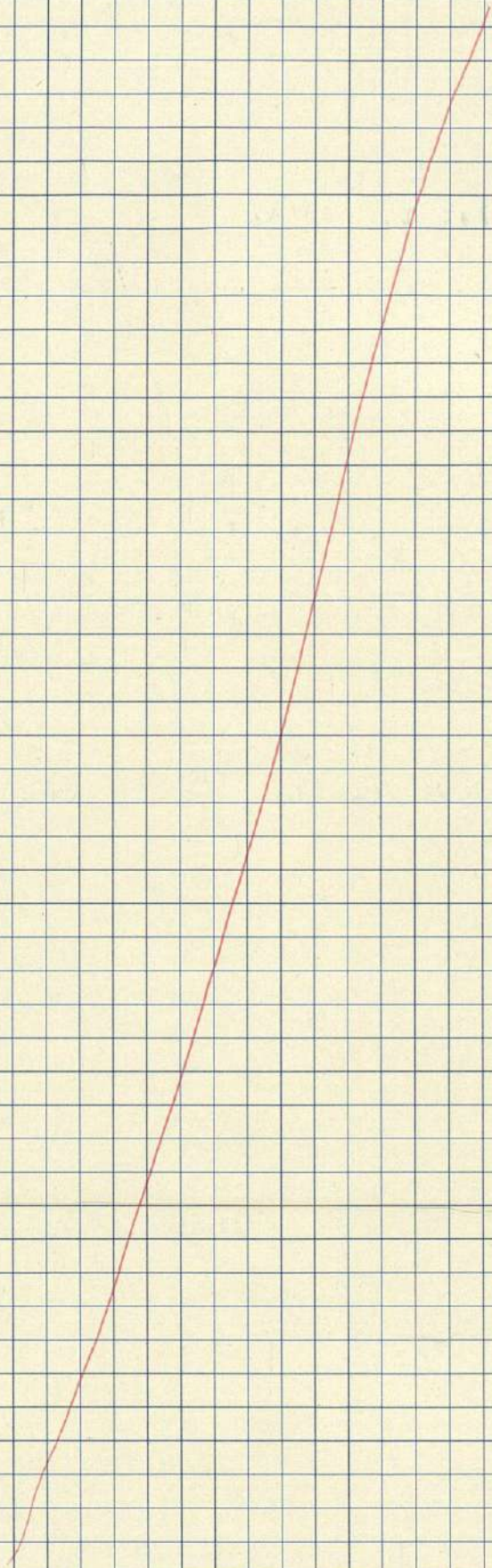
$0x CD = 1100 \ 1101_2$

Il sera aussi stocké dans la barrette
numero 1.

3) 2 octets \rightarrow 16 bits, 16 Broches données.

0124





NOM BLOQUET
 Prénom Romain
 Promo 2018
 Date 3/06/2015

MATIÈRE ADO. PL2.

2) Une adresse mémoire est envoyée par le processeur, le TAG est vérifié.
 Si il est valide et que le Bit de Validité est à 1 alors on a un
 Cache HIT (Données Trouvées).
 Si le TAG est invalide ou le Bits de Validité est à 0 alors
 on a un cache MISS.

Exercice 48

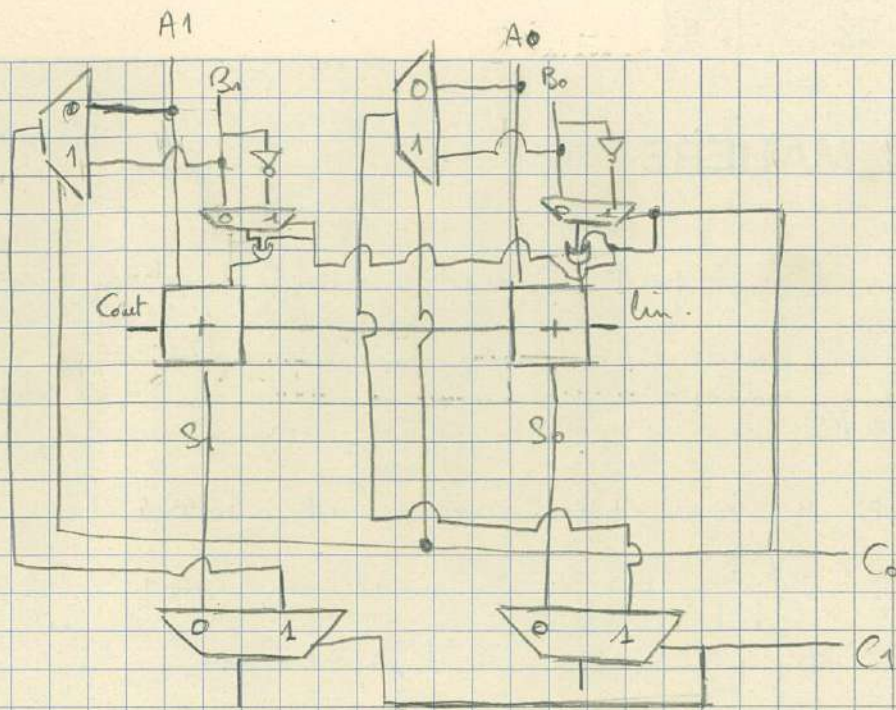
65
04

Fetch : le Program Counter lit l'adresse de l'instruction
 en mémoire et l'envoi à l'IR.

DECODE : l'instruction est décodée, l'unité de contrôle envoi
 des signaux pour indiquer le chemin que vont prendre les
 données.

Exécute : les données suivent le chemin indiqués dans la phase
 de code et subissent l'opération dans l'ALU.

FETCH		Mem	DECODE		Mem	EXÉCUTE	Mem
LDA	1L	I	0	I	1L	D	
STA	1L	I	0	I	1E	D	
ADD	1L	I	0	I	1L	D	
SUB	1L	I	0	I	1L	D	
MUL	1L	I	0	I	1L	D	
DIV	1L	I	0	I	1L	D	
CMP	1L	I	0	I	1L	D	
BRx	1L	I	0	X	0	0	



Exercice 2:

Exo 2: ~~3/10~~
05

1) Les Barrettes font 2^{27} octets. il faut donc 27 broches adresses. (n).

Les Mots sont de 8 bits il faut donc 8 broches données (m).

2) La largeur du Bus adresse de la mémoire centrale est de 28 bits adresses. A_0, A_1, \dots, A_{27}

La largeur du Bus de données sera de 8 bits. D_0, D_1, \dots, D_7 .

Pour faire le schéma, il nous faudra 2 barrette de 128 Mo.

A_0 nous servira de bit de sélection pour la barrette.

Si $A_0 = 0$ alors la barrette n° 1 sera sélectionné, sinon

si $A_0 = 1$ alors se sera la barrette n° 2.

