Année 2003 – 2004 Session de juin 2004
DEUG MIAS et LST Informatique
Épreuve d'Architecture de l'ordinateur
Date : le premier juin 2004 Durée : 1h30
Tous documents légaux autorisés.
Votre voisin n'est pas un document
Vous devez répondre directement sur le sujet
Épreuve de M Strandh

Répondre directement sur le sujet dans les cadres prévus à cet effet. Aucune réponse ou partie de réponse en dehors des cadres ne sera prise en compte.

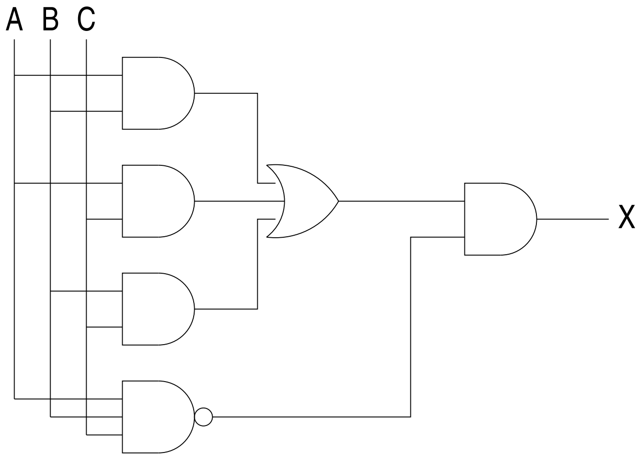
Écrire votre numéro d'anonymat sur chaque feuille du sujet.

Rappel : Soyez bref et clair ! Personne n'est impressionné par la quantité !

Bon courage !

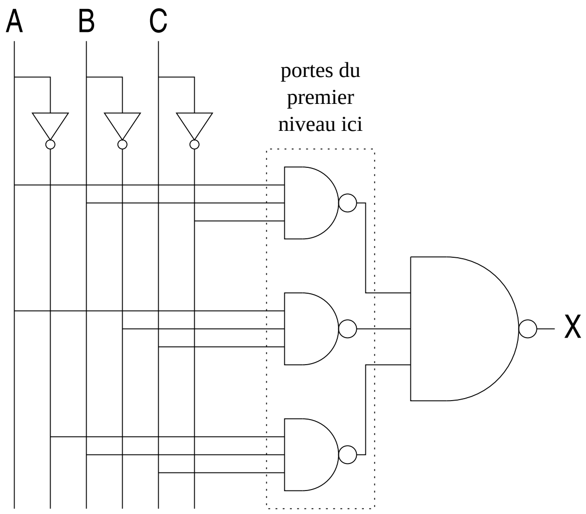
Exercice 1 (3 p)

Voici un circuit combinatoire avec trois entrées (`A`, `B` et `C`) et une sortie (`X`) :



Réaliser un circuit à deux niveaux utilisant uniquement des portes **non-et** et des portes **non** équivalent au circuit ci-dessus.

Réponse :



Exercice 2 (5 p)

On imagine un circuit séquentiel avec la table d'états suivante :

x_2	x_1	x_0	a	b	x'_2	x'_1	x'_0	a'
0	0	0	0	0	0	0	0	0
0	0	0	0	1	0	0	1	0
0	0	0	1	0	0	0	1	0
0	0	0	1	1	0	0	1	0
0	0	1	0	0	0	1	0	0
0	0	1	0	1	0	1	0	0
0	0	1	1	0	0	0	0	0
0	0	1	1	1	0	0	0	0
0	1	0	0	0	0	1	1	0
0	1	0	0	1	0	1	1	0
0	1	0	1	0	0	0	1	1
0	1	0	1	1	0	0	1	1
0	1	1	0	0	1	0	0	0
0	1	1	0	1	1	0	0	0
0	1	1	1	0	0	1	0	1
0	1	1	1	1	0	1	0	1
1	0	0	0	0	1	0	1	0
1	0	0	0	1	1	0	1	0
1	0	0	1	0	0	1	1	1
1	0	0	1	1	0	1	1	1
1	0	1	0	0	1	1	0	0
1	0	1	0	1	1	1	0	0
1	0	1	1	0	1	0	0	1
1	0	1	1	1	1	0	0	1
1	1	0	0	0	1	1	1	0
1	1	0	0	1	1	1	1	0
1	1	0	1	0	1	0	1	1
1	1	0	1	1	1	0	1	1
1	1	1	0	0	1	1	1	0
1	1	1	0	1	1	1	0	1
1	1	1	1	0	1	1	1	1
1	1	1	1	1	1	1	0	1

Les signaux $x_2x_1x_0$ représente une tension pour piloter un moteur telle que 000 représente 0 Volt (moteur arrêté) et 111 représente 240 Volt (moteur en vitesse maximale). Le signal b est branché à un bouton sur lequel on doit appuyer pour démarrer ou arrêter le moteur.

Donner une explication brève de ce que l'on observe lorsqu'on appuie sur le bouton en fonction de l'état de marche du moteur (arrêté, vitesse maximale, entre les deux). [2p]

Réponse :

Moteur arrêté : le moteur démarre lentement jusqu'à l'obtention de la vitesse maximale Moteur en vitesse maximale : le moteur s'arrête lentement jusqu'à l'arrêt complet Entre les deux : le moteur continue à démarrer ou à s'arrêter selon le cas.

À quoi sert la sortie a ? [1p]

Réponse :

Elle détermine si le moteur est en train de démarrer ou en train de s'arrêter.

Quel peut être l'utilité d'un tel circuit ? [2p]

Réponse :

Il évite le démarrage ou l'arrêt trop brutal du moteur.

Exercice 3 (5 p)

Annexe 1 montre une modification de l'architecture traitée en cours. Cette modification permet de sortir la somme du contenu du registre R1 et celui du registre d'adresses sur le bus d'adresses.

Voici le micro programme d'une instruction (appelée `LIX`) qui profite de cette modification :

Cycle	Micro opérations dont la valeur est égale à 1
1	5 3 9 7
2	1 3 11 A

Donner une description de l'instruction destinée aux programmeur de l'architecture, c'est à dire la personne qui se sert de cette architecture pour programmer des applications (et non la personne chargé d'implémenter l'architecture). [2p]

Réponse :

L'argument de l'instruction est additionné au contenu de R1. Le résultat est une adresse en mémoire dont le contenu sera chargé dans R0.

Montrer le code généré par un compilateur à partir de la fonction `tail` dans le code C ci-dessous. [3p]

```
struct cell
{
  int val;
  struct cell *next;
};

struct cell *
tail(struct cell *cp)
{
  return cp -> next;
}
```

Réponse :

```
tail :   lds 1
         copy
         lix 1
         copy
         ret
```

Exercice 4 (7 p)

Voici un sous programme en code *assembleur* (les *noms* plutôt que les codes des instruction sont employés, et les adresses sont *symboliques* plutôt que numériques) :

```
mystere: lds 2
         copy
         jz out
         ldimm -1
         add
         push
         lds 2
         copy
         push
         jsr mystere
         addsp 2
         lds 1
         add
         ret
out:     ldimm 0
         copy
         ret
```

En supposant que la première instruction du sous programme `mystere` se trouve à l'adresse 30 (en base 10), quel est le contenu de la mémoire (sous la forme de suite d'octets) entre l'adresse 30 et l'adresse 54 (se servir de l'annexe 2 pour les codes des instructions) ? [4p]

(réponse sur la page suivante)

Réponse :

Adresse	Contenu (en base 10)
30	21
31	2
32	4
33	14
34	51
35	1
36	-1
37	7
38	19
39	21
40	2
41	4
42	19
43	17
44	30
45	23
46	2
47	21
48	1
49	7
50	18
51	1
52	0
53	4
54	18

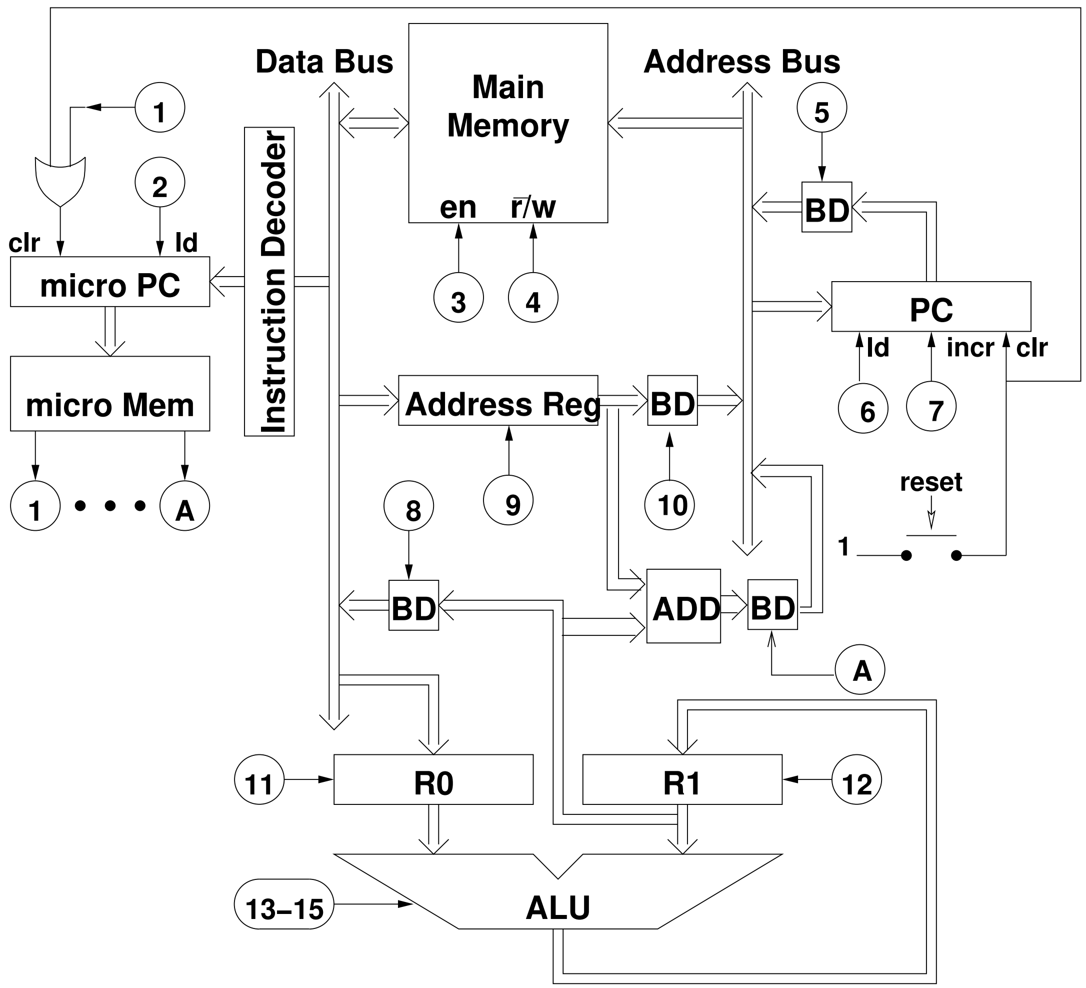
Combien de paramètres sont attendus par le programme `mystere` ? [1p]

Réponse : 2

Quelle est la fonction calculée par le sous programme `mystere` ? [2p]

Réponse : multiplication

Annexe 1


Data Bus
Main Memory
Address Bus
en
r/w
Instruction Decoder
micro PC
micro Mem
clr
ld
PC
ld
incr
clr
BD
Address Reg
ADD
reset
R0
R1
ALU
1
2
3
4
5
6
7
8
9
10
11
12
13–15
A

Annexe 2

Nom	Code	Description
nop	0	ne rien faire
ldimm	1	chargement immédiat
ld	2	chargement
st	3	stockage
copy	4	copier R0 dans R1
shl	5	décalage de R1 à gauche
shr	6	décalage de R1 à droite
add	7	R1 reçoit la somme de R0 et R1
sub	8	R1 reçoit la différence de R1 et R0
and	9	R1 reçoit le *et* logique entre R0 et R1
or	10	R1 reçoit le *ou* logique entre R0 et R1
not	11	R1 reçoit la négation de R1
jal	12	saut non conditionnel
jn	13	saut si le résultat négatif
jz	14	saut si le résultat zéro
jv	15	saut si débordement
jc	16	saut si retenue
jsr	17	appel à sous programme
ret	18	retour de sous programme
push	19	empiler R1
pop	20	dépiler dans R0
lds	21	chargement d'une valeur dans la pile
sts	22	stockage dans la pile
addsp	23	additionner une constante à SP