

**CONTROLE INTERMEDIARE
D'ARCHITECTURE - CPI 2**

Durée : 2 h.

Tous documents interdits.

Exercice 1 : (4 points)

On veut réaliser une mémoire vive de 2048x8 bits à l'aide de boîtiers mémoires de 256x4bits répartie en 2 modules avec un degré d'entrelacement $d=2$ à l'intérieur de chaque module.

Question :

Faire le schéma détaillé d'un seul des deux modules.

Détailler toutes les connexions avec les différents bus (bus d'adresses, de données et de commande), ainsi que toutes les connexions à tous les boîtiers de ce module (sélection, adresse, données, commandes).

Préciser clairement tous les poids des bits du bus d'adresse global.

Exercice 2 : (8 points)

Une société de service dans le transport dispose de véhicule de différents types. Les types sont numérotés de 0 à 3 comme suit :

0 = véhicule léger (tourisme)

1 = véhicule de petit tonnage (camionnette)

2 = véhicule de moyen tonnage (camion)

3 = véhicule de grand tonnage (semi-remorque)

Tous les véhicule doivent effectuer un déplacement par jour, et doivent rejoindre les parkings de l'entreprise. L'entreprise dispose de 4 parkings A, B, C, D ; chaque parking est associé à un type de véhicule.

PARTIE A

L'entrée dans l'entreprise est contrôlée par une barrière unique. A l'arrivée d'un véhicule, un signal est généré automatiquement par un détecteur, la barrière est ouverte, le type de véhicule, le numéro du véhicule, ainsi que l'heure d'arrivée, sont automatiquement enregistrés dans une mémoire unique qui conserve l'ordre d'arrivée.

Questions :

1. Quel est le type de mémoire qu'on doit utiliser pour réaliser ce circuit ?
2. Donner, un schéma détaillé de la mémoire et du circuit de commande ainsi que des signaux qui indiquent si l'enregistrement est autorisé ou non.

PARTIE B

Les véhicules doivent ensuite s'orienter vers le parking réservé au type de véhicule. L'entrée à chaque parking est contrôlée par une barrière qui s'ouvre de manière automatique selon le type de véhicule.

On veut réaliser un circuit qui ouvre automatiquement la barrière du parking selon le type de véhicule. Pour cela, on utilisera une mémoire.

Questions :

1. Quel est le type de mémoire qu'on doit utiliser pour réaliser ce circuit et quelle est sa taille (nombre de mots et nombre de bits) ?
2. Faire le schéma de ce circuit en utilisant uniquement la mémoire proposée.

PARTIE C

Lorsqu'un véhicule rentre dans le parking qui lui est réservé, les informations concernant le véhicule (numéro, heure d'arrivée ainsi que le nombre de kilomètres parcourus par le véhicule) sont enregistrées dans une mémoire associative. Il existe une mémoire associative pour chaque parking.

Questions :

1. Faire le schéma global du circuit en montrant la connexion avec le circuit de la partie A et de la partie B.
2. Ecrire le programme qui permet de lire les informations concernant le véhicule qui a parcouru le plus grand nombre de kilomètres pour un type de voiture donné (le classement se fait pour des véhicules de même type uniquement).

Remarque : les différentes mémoires associatives ont suffisamment d'espace pour enregistrer toutes les voitures.

Exercice 3 : (3 points)

On veut réaliser un circuit qui multiplie ou divise par deux un nombre A codé sur 3 bits. Une commande **OP** permet de choisir l'opération à réaliser :

- Si **OP** = 0 : opération de multiplication ;
- Si **OP** = 1 : opération de division.

Question :

- 1- Faire la table de vérité du circuit.
- 2- Réaliser le schéma du circuit avec une ROM de 16 mots.

Nom:.....

Prénom:

Groupe:.....

Exercice 4: (5 points)

Répondre directement sur la feuille jointe. Pour chaque question, cocher la ou les réponses correctes.

1.	Technologie des mémoires Cochez la ou les affirmations correctes : <input type="checkbox"/> l'écriture dans une mémoire dynamique prend deux fois plus de temps que la lecture; <input checked="" type="checkbox"/> une cellule de mémoire statique occupe plus d'espace qu'une cellule dynamique; <input type="checkbox"/> le rafraîchissement se fait uniquement sur les mots non modifiés; <input type="checkbox"/> les cellules des mémoires dynamiques sont plus rapides que les mémoires statiques.
2.	Mémoires cache Dans les mémoires cache, la localité spatiale signifie : <input type="checkbox"/> que la donnée et l'instruction utilisent la même adresse; <input checked="" type="checkbox"/> que si une information est utilisée à l'adresse i, alors la probabilité est élevée pour qu'elle le soit aussi à une adresse très proche ; <input type="checkbox"/> que les données utilisées souvent ne sont jamais transférées en mémoire cache; <input type="checkbox"/> que les données et les instructions sont stockées dans des mémoires différentes.
3.	Mémoires cache Cochez la ou les affirmations correctes : <input type="checkbox"/> les mémoires cache actuelles sont aussi grandes que les mémoires centrales ; <input type="checkbox"/> les caches internes sont toujours plus grandes que les caches externes ; <input type="checkbox"/> les caches de grandes tailles sont plus rapides que les petites caches ; <input checked="" type="checkbox"/> le remplacement aléatoire donne d'aussi bonnes performances que le LRU pour des mémoires de grandes tailles.
4.	Mémoires mortes Cochez la ou les affirmations correctes : <input type="checkbox"/> les PROM sont réalisées à base de diodes uniquement; <input type="checkbox"/> les EEPROM peuvent remplacer les mémoires centrales ; <input checked="" type="checkbox"/> les EEPROM sont effaçables électriquement ; <input type="checkbox"/> l'écriture dans les EPROM se fait en exposant la mémoire à des rayons ultraviolets.
5.	Organisation des Mémoires Cochez la ou les affirmations correctes : <input type="checkbox"/> il ne peut jamais y avoir de conflit d'accès dans des mémoires modulaires ; <input type="checkbox"/> les mémoires entrelacées permettent de réaliser des mémoires plus grandes que les mémoires modulaires; <input checked="" type="checkbox"/> les mémoires entrelacées sont utiles pour des architectures avec plusieurs processeurs ; <input type="checkbox"/> les mémoires modulaires sont plus rapides que les mémoires en un seul bloc.