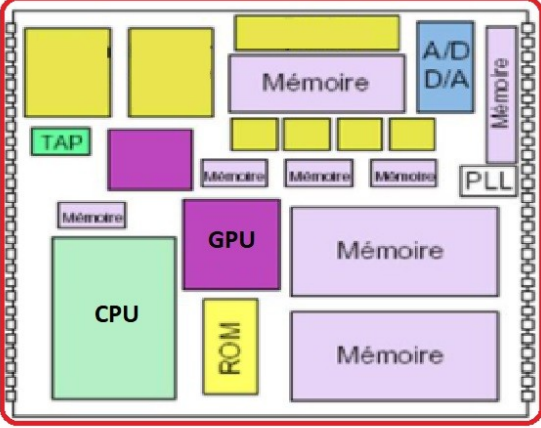
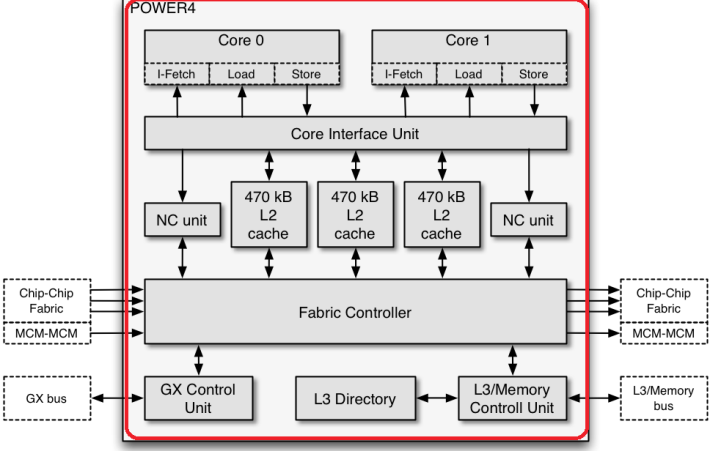


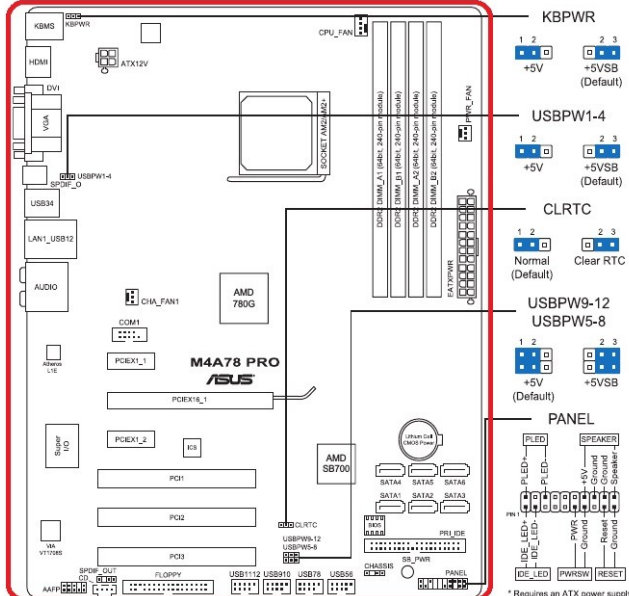
CH4 Exercices

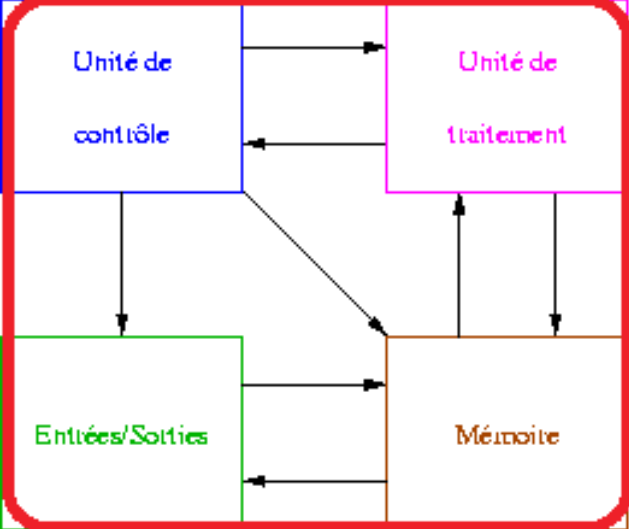
Exercice 1 :

Parmi les représentations suivantes, la(les)quelle(s) représente(nt) un SoC ?









12

Exercice 2 :

A partir de l'article du site elektormagazine.fr

- 1) Relever les différentes caractéristiques du SoC du Raspberry Pi 3 modèle B+.
- 2) Les comparer au SoC du Raspberry Pi 4.
- 3) Quelles sont les principales évolutions qui contribuent à ce gain ? Voir le doc [article_Pi_3.pdf](#)



Exercice 3 :

La photo ci-dessous montre le détail d'un SoC Kirin 990. Identifier les différentes parties de ce SoC.



Exercice 4 :

Utiliser le terminal pour répondre aux questions suivantes:

- 1) Ouvrir le navigateur web. Combien de processus sont créés? Noter son(ses) PID et PPID.
- 2) Ouvrir des nouveaux onglets. De nouveaux processus sont-ils créés?
- 3) Ouvrir des nouvelles fenêtres de navigateur. De nouveaux processus sont-ils créés?
- 4) Fermer puis ré-ouvrir le navigateur. Les PID et PPID ont-ils changés?

Exercice 5 :

Lancer la commande `ps -o pid,ppid,user,stat,comm` et noter son PID(identifiant de processus) et PPID(identifiant du processus parent).

- 1) A l'aide de la commande `ps -ef`, remonter au processus parent, et ainsi de suite jusqu'au processus racine.
- 2) Refermer le terminal, puis recommencer. Que constate-t-on?

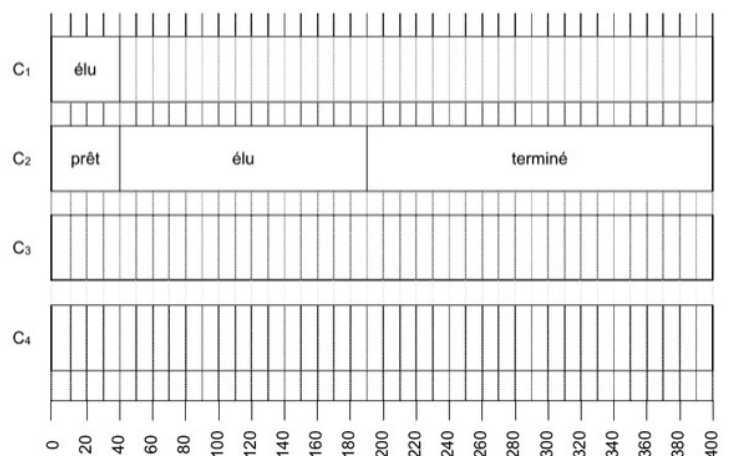
Exercice 6 : (type bac)

- 1) Les états possibles d'un processus sont : *prêt*, *élu*, *terminé* et *bloqué*.
 - (a) Expliquer à quoi correspond l'état *élu*.
 - (b) Proposer un schéma illustrant les passages entre les différents états.

On suppose que quatre processus C_1 , C_2 , C_3 et C_4 sont créés sur un ordinateur, et qu'aucun autre processus n'est lancé sur celui-ci, ni préalablement ni pendant l'exécution des quatre processus. L'ordonnanceur, pour exécuter les différents processus prêts, les place dans une structure de données de type file. Un processus prêt est enfilé et un processus élu est défilé.

- 2)
 - (a) Parmi les propositions suivantes, recopier celle qui décrit le fonctionnement des entrées/sorties dans une file :
 - Premier entré, dernier sorti
 - Premier entré, premier sorti
 - Dernier entré, premier sorti
 - (b) On suppose que les quatre processus arrivent dans la file et y sont placés dans l'ordre C_1 , C_2 , C_3 et C_4 . Les temps d'exécution totaux de C_1 , C_2 , C_3 et C_4 sont respectivement 100 ms, 150 ms, 80 ms et 60 ms.

- Après 40 ms d'exécution, le processus C_1 demande une opération d'écriture disque, opération qui dure 200 ms. Pendant cette opération d'écriture, le processus C_1 passe à l'état bloqué.
- Après 20 ms d'exécution, le processus C_3 demande une opération d'écriture disque, opération qui dure 10 ms. Pendant cette opération d'écriture, le processus C_3 passe à l'état bloqué.



Sur la frise chronologique, les états du processus C_2 sont donnés. Compléter la frise avec les états des processus C_1 , C_3 et C_4 .

On trouvera ci-dessous deux programmes rédigés en pseudo-code.

Verrouiller un fichier signifie que le programme demande un accès exclusif au fichier et l'obtient si le fichier est disponible.

Programme 1

```
Verrouiller fichier_1
Calculs sur fichier_1
Verrouiller fichier_2
Calculs sur fichier_1
Calculs sur fichier_2
Calculs sur fichier_1
Calculs sur fichier_2
Déverrouiller fichier_2
Déverrouiller fichier_1
```

Programme 2

```
Verrouiller fichier_2
Verrouiller fichier_1
Calculs sur fichier_1
Calculs sur fichier_2
Déverrouiller fichier_1
Déverrouiller fichier_2
```

3)

- En supposant que les processus correspondant à ces programmes s'exécutent simultanément (exécution concurrente), expliquer le problème qui peut être rencontré.
- Proposer une modification du programme 2 permettant d'éviter ce problème.

Exercice 5 : (type bac)

La commande UNIX `ps` présente un cliché instantané des processus en cours d'exécution.

Avec l'option `-eo pid,ppid,stat,command`, cette commande affiche dans l'ordre l'identifiant du processus PID (process identifier), le PPID (parent process identifier), l'état STAT et le nom de la commande à l'origine du processus.

Les valeurs du champ STAT indique l'état des processus :

- R : processus en cours d'exécution
- S : processus endormi

Sur un ordinateur, on exécute la commande `ps -eo pid,ppid,stat,command` et on obtient un affichage dont on donne ci-dessous un extrait.

\$ ps -eo pid,ppid,stat,command

PID	PPID	STAT	COMMAND
1	0	Ss	/sbin/init
....
1912	1908	Ss	Bash
2014	1912	Ss	Bash
1920	1747	Sl	Gedit
2013	1912	Ss	Bash
2091	1593	Sl	/usr/lib/firefox/firefox
5437	1912	Sl	python programme1.py
5440	2013	R	python programme2.py
5450	1912	R+	ps -eo pid,ppid,stat,command

- Quel est le nom de la première commande exécutée par le système d'exploitation lors du démarrage?
- Quels sont les identifiants des processus actifs sur cet ordinateur au moment de l'appel de la commande `ps` ? Justifier la réponse.

- 3) Depuis quelle application a-t-on exécuté la commande `ps` ? Donner les autres commandes qui ont été exécutées à partir de cette application.
- 4) Expliquer l'ordre dans lequel les deux commandes `python programme1.py` et `python programme2.py` ont été exécutées.
- 5) Peut-on prédire que l'une des deux commandes `python programme1.py` et `python programme2.py` finira avant l'autre?