

# NSI - Terminale - Architecture

## Composants et systèmes sur puce

qkzk

2020/12/23

## Travaux dirigés

Dans ce TP nous allons mener une petite étude comparative de différentes machines utilisant des systèmes sur puce.

Les objectifs du programme étant très vagues (cf fin du TP) nous allons nous contenter de peu de choses...

### Questions générales

1. À propos de la RAM
  - a. Donner une définition de la RAM
  - b. Quand on allume un ordinateur, qu'est-ce qui est avant tout stocké dans la RAM ?
2. À propos des processeurs
  - a. À quoi correspond la fréquence d'un processeur ?
  - b. Par quel sigle anglosaxon désigne-t-on un processeur ?
  - c. Dans un processeur, qu'est-ce qu'un coeur ?
  - d. Quel est l'avantage d'avoir plusieurs coeurs
  - e. Est-il vrai qu'il ne peut y avoir qu'un unique processeur dans une machine ?
3. À propos des SoCs
  - a. Que signifie l'acronyme "SoC" ?
  - b. Donner une définition d'un SoC.
  - c. En plus de sa petite taille, quel est l'autre principal avantage d'un SoC ?
  - d. Citer un inconvénient majeur des SoCs.

## Le raspberry Pi

1. Documentez-vous sur le Raspberry Pi ici.

### Document 1 : pinout

En utilisant une carte Raspberry Pi, on ouvre un terminal et on tape la commande `pinout`. Voici ce que le terminal renvoie :

```
,-----.  
| oooooooooooooooooooooo J8  +=====  
| 1ooooooooooooooooooooo PoE | Net  
| Wi                          oo +=====  
| Fi Pi Model 4B V1.1 oo      |  
|                               +=====  
| |D|   |SoC|                  |USB3  
| |S|   |   |                  +=====  
| |I|   `-----'                |  
|                               +=====  
|                               |S|   |USB2  
| pwr   |HD|   |HD| |I| |A|    +=====
```

```
`-| |---|MI|---|MI|----|V|-----'
```

```
Revision      : b03111
SoC           : BCM2711
RAM           : 2048Mb
Storage       : MicroSD
USB ports     : 4 (excluding power)
Ethernet ports : 1
Wi-fi         : True
Bluetooth     : True
Camera ports (CSI) : 1
Display ports (DSI): 1
```

J8:

```
3V3 (1) (2) 5V
GPIO2 (3) (4) 5V
GPIO3 (5) (6) GND
GPIO4 (7) (8) GPIO14
GND (9) (10) GPIO15
GPIO17 (11) (12) GPIO18
GPIO27 (13) (14) GND
GPIO22 (15) (16) GPIO23
3V3 (17) (18) GPIO24
GPIO10 (19) (20) GND
GPIO9 (21) (22) GPIO25
GPIO11 (23) (24) GPIO8
GND (25) (26) GPIO7
GPIO0 (27) (28) GPIO1
GPIO5 (29) (30) GND
GPIO6 (31) (32) GPIO12
GPIO13 (33) (34) GND
GPIO19 (35) (36) GPIO16
GPIO26 (37) (38) GPIO20
GND (39) (40) GPIO21
```

For further information, please refer to <https://pinout.xyz/>

## Document 2 : numéro de révision

Le numéro de révision est un code utilisé depuis le Raspberry Pi 2. C'est un ensemble de 6 caractères au format hexadécimal.

## Document 3 : lecture du numéro de révision

En numérotant les 24 bits de gauche à droite ( $b_1$  à  $b_{24}$ ), on peut obtenir les informations suivantes :

- valeur de  $b_1$  : 1 pour new-style revision, 0 pour old-style revision
- valeur de  $k = b_2b_3b_4$  : taille de la mémoire  $2^{8+k}$  Mo
- valeur de  $k' = b_5b_6b_7b_8$  : correspond au fabricant avec 0 pour Sony UK, 1 pour Egoman, 2 pour Embest, 3 pour Sony Japan, 4 pour Embest et 5 pour Stadium
- Valeur de  $k'' = b_{13}b_{14}b_{15}b_{16}b_{17}b_{18}b_{19}b_{20}$  : le type avec 0: A, 1: B, 2: A+, 3: B+, 4 : 2B etc.
- valeur de  $k''' = b_{21}b_{22}b_{23}b_{24}$  : le numéro de révision, soit 0, 1, 2 etc.

### Répondez aux questions suivantes :

1. Quelle entreprise a *conçu* le SoC utilisé sur le Raspberry Pi ? Cette entreprise *fabrique*-t-elle quelque chose ?
2. Quel est le numéro du processeur du raspberry utilisé ?
3. *pinout* affiche le modèle employé, vérifier le nom du modèle à l'aide du numéro de révision.
4. Déterminer la taille en Go, sans l'exprimer avec une puissance.
5. Pourquoi est-il obligatoire d'utiliser au moins trois bits pour le fabricant ? Le déterminer.
6. Que signifie l'acronyme PoE (se documenter si nécessaire). Quel usage cela permet-il pour le raspberry Pi ?

7. Que signifie l'acronyme GPIO (se documenter si nécessaire). Quel usage peut-on donner à ces pins ?

## La console Nintendo Switch

1. Se documenter sur la Nintendo Switch

### Répondez aux questions suivantes :

1. Quel est le SoC de la Nintendo Switch ? Qui le fabrique ? De quelle famille de SoC dérive-t-il ?
2. De combien de cœurs de processeurs dispose la Nintendo Switch ? On entend parfois que c'est une machine "quad-core", parfois que c'est une machine "octo-core". Expliquer la confusion.
3. Vous souhaitez développer un jeu pour la Nintendo Switch. Voici les étapes à suivre :
  1. Acheter un SDK Nintendo. Qu'est-ce ? Répondez. Combien coûte le SDK Nintendo ? Par comparaison, combien coûte le SDK la PlayStation 4 ?
  2. Apprendre à programmer en 3D avec Unity. Qu'est-ce qu'Unity ? Dans quel langage de programmation sont écrits la majorité des programmes Unity ? De quel langage est-il le plus proche ?
  3. Déployer le jeu sur le Nintendo eShop. Quelles sont les étapes principales à respecter ?
4. Lassé du développement, vous voulez hacker la Switch. Il vous faut quelques informations supplémentaires. Quel autre type de matériel disposant d'un système d'exploitation *open source* utilise le SoC de la Nintendo Switch ? Peut-on trouver des références sur le matériel fourni par le constructeur ?

## Le smartphone OnePlus 6T

*J'ai choisi ce modèle relativement ancien car on dispose de beaucoup d'information le concernant*

1. Les bases : système d'exploitation, SoC, connectivité, date de sortie. Répondez.
2. Comparez les performances de ce modèle avec celles du modèle phare de la même marque sorti cette année : processeur, batterie, photographie.
3. "Right to Repair" : qu'est-ce que c'est ? Que souhaitent les animateurs de ce mouvement ? Citez une immense entreprise (que vous connaissez tous) qui milite *contre* ce mouvement. Quels sont ses arguments ?
4. Fier possesseur d'un OnePlus 6T, vous l'avez fait tomber et il ne démarre plus.
  1. Quelles sont les pièces que vous pouvez remplacer vous-même (en supposant que vous disposez des outils et des compétences) ?
  2. Sont-elles disponibles dans le commerce en ligne ?
  3. D'après les guides qu'on peut trouver en ligne, ce téléphone est noté 5/10 en "repairability". Comparez avec deux téléphones plus récents (Samsung Galaxy S20 et iPhone 12 Pro). D'où viennent les différences ?
5. Maintenant que vous êtes un grand connaisseur des smartphones et de l'Android, vous souhaitez développer une application mobile.
  1. Dans quel langage sont écrits la majorité des applications Android ?
  2. Peut-on développer dans ce langage sur iPhone ?
  3. Il existe d'autres projets permettant de développer des applications mobiles qui se compilent avec le même code sur Android et sur iPhone. Citez-en un.

---

## Programme

**Contenus :** Composants intégrés d'un système sur puce.

**Capacités attendues :** Identifier les principaux composants sur un schéma de circuit et les avantages de leur intégration en termes de vitesse et de consommation.

**Commentaires :** Le circuit d'un téléphone peut être pris comme un exemple : microprocesseurs, mémoires locales, interfaces radio et filaires, gestion d'énergie, contrôleurs vidéo, accélérateur graphique, réseaux sur puce, etc.