

# TP Découverte

## Carte RASPBERRY PI



DATE : 02/09/2012

SIN

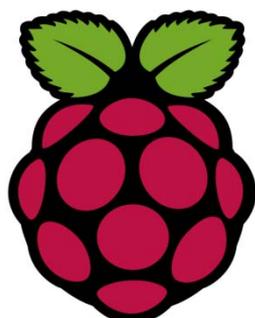
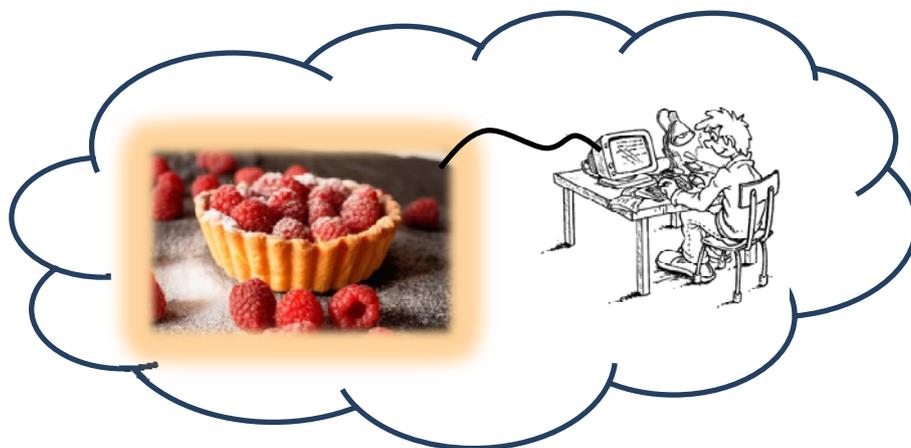
TSTI2D

### Objectifs :

- Mise en œuvre de la carte RASPBERRY PI
- Comprendre une architecture RISC/ARM
- Définir les acronymes
- Mise en œuvre d'un IOS
- Tester le GPIO

### Activités

- TD
- Recherche de documentation
- Recherche internet
- TP



# 1 Histoire

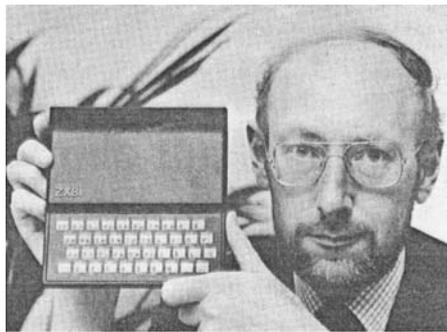
## 1.1 La fondation Raspberry Pi

La fondation Raspberry Pi est une organisation à but non lucratif.

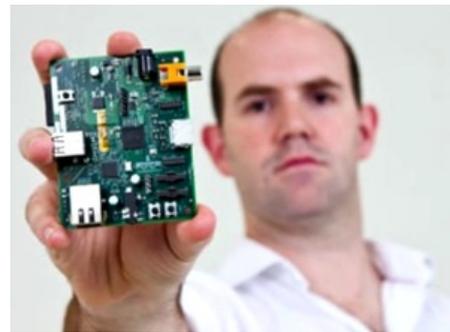
Eben Upton a constaté l'évolution des candidats à l'université de Cambridge. Les candidats initialement passionnés par la programmation ont été remplacés, année après année, par des candidats ayant au mieux pratiqué la conception de site web, parfois même pas cela. Le constat est d'autant plus amer que le nombre de candidatures s'amenuisait avec le temps.

Avec ses collègues de l'université, il a identifié quelques-unes des causes de ce changement du rapport à l'informatique. Notamment, le phagocytage de l'enseignement des technologies de l'information par des cours d'utilisation de logiciels comme Word ou Excel. Plus étonnant, l'avènement des PC et consoles de jeu au détriment des machines Sinclair, Amiga ou Commodore 64 sur lesquels la génération précédente avait appris à programmer.

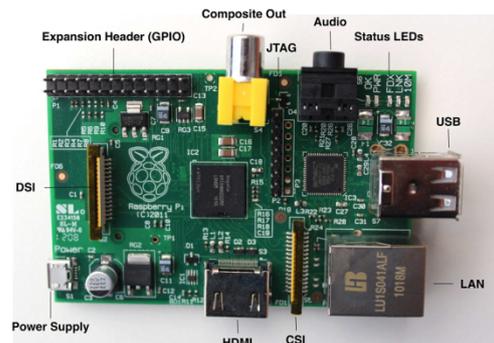
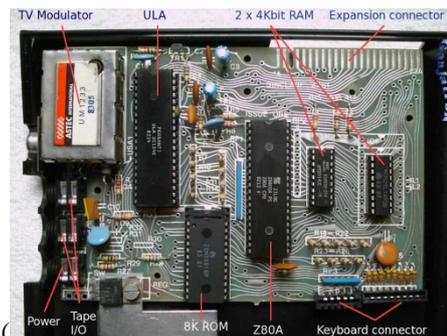
### *31 ans les séparent !*



Sire Clive Sinclair et son ZX81® (1981)



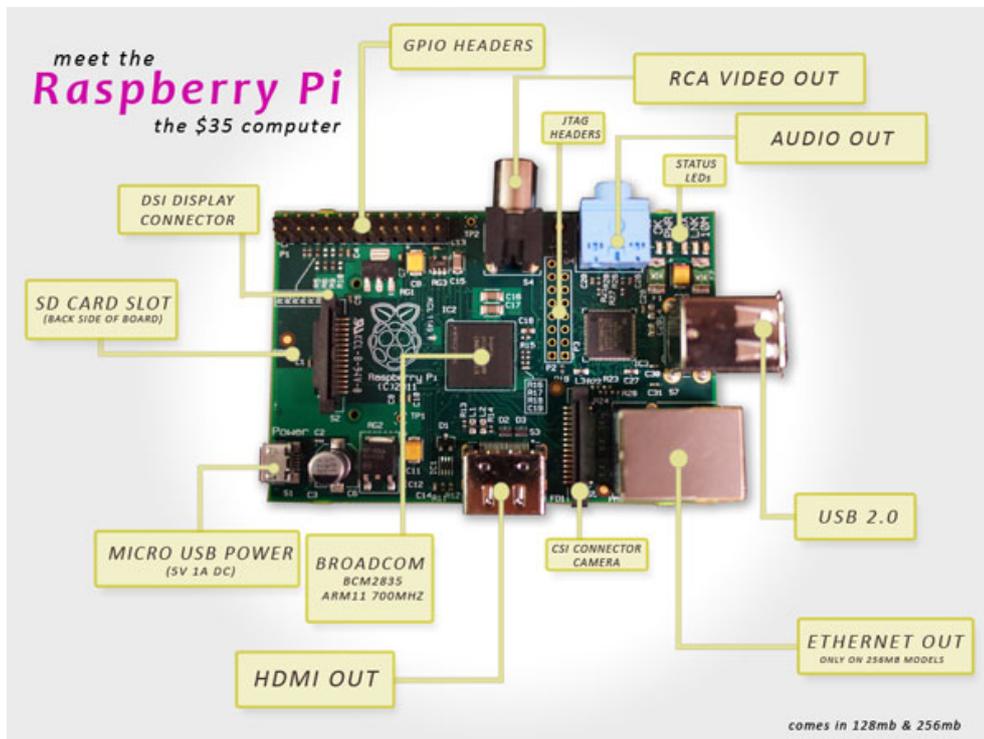
Dr Eben Upton et son Raspberry® (2012)



L'objectif de l'association est donc d'amener les jeunes à la programmation au moyen d'un ordinateur attractif aussi bien par son aspect ludique (lecture de vidéos de qualité Blu-Ray) que par son prix (compter 25 \$, le recyclage d'une carte SD de 2 Go et d'une alimentation). Bien évidemment, les écoles sont des lieux dans lesquels la fondation espère particulièrement diffuser l'ordinateur qu'elle a conçu.

## 2 La carte Raspberry Pi :

Tous les composants tiennent sur un circuit imprimé de la taille d'une carte de crédit : 85.60 mm x 53.98 mm x 17 mm, pour un poids total de 45 g.



La carte met en œuvre un SoC Broadcom BCM2835 lui-même constitué d'un processeur ARMv6 (ARM1176JZFS) cadencé à 700 MHz et d'un processeur graphique (Videocore). Ce dernier est capable de décompresser les vidéos codées en [H.264](#), pour une résolution « Full HD » (1080p30, soit une résolution de 1920 par 1080 pixels à une vitesse de 30 images par seconde). La fondation compare la puissance du Pi à une Xbox 1ère génération pour la partie graphique, et à un Pentium 2 cadencé à 300 MHz pour un usage généraliste.

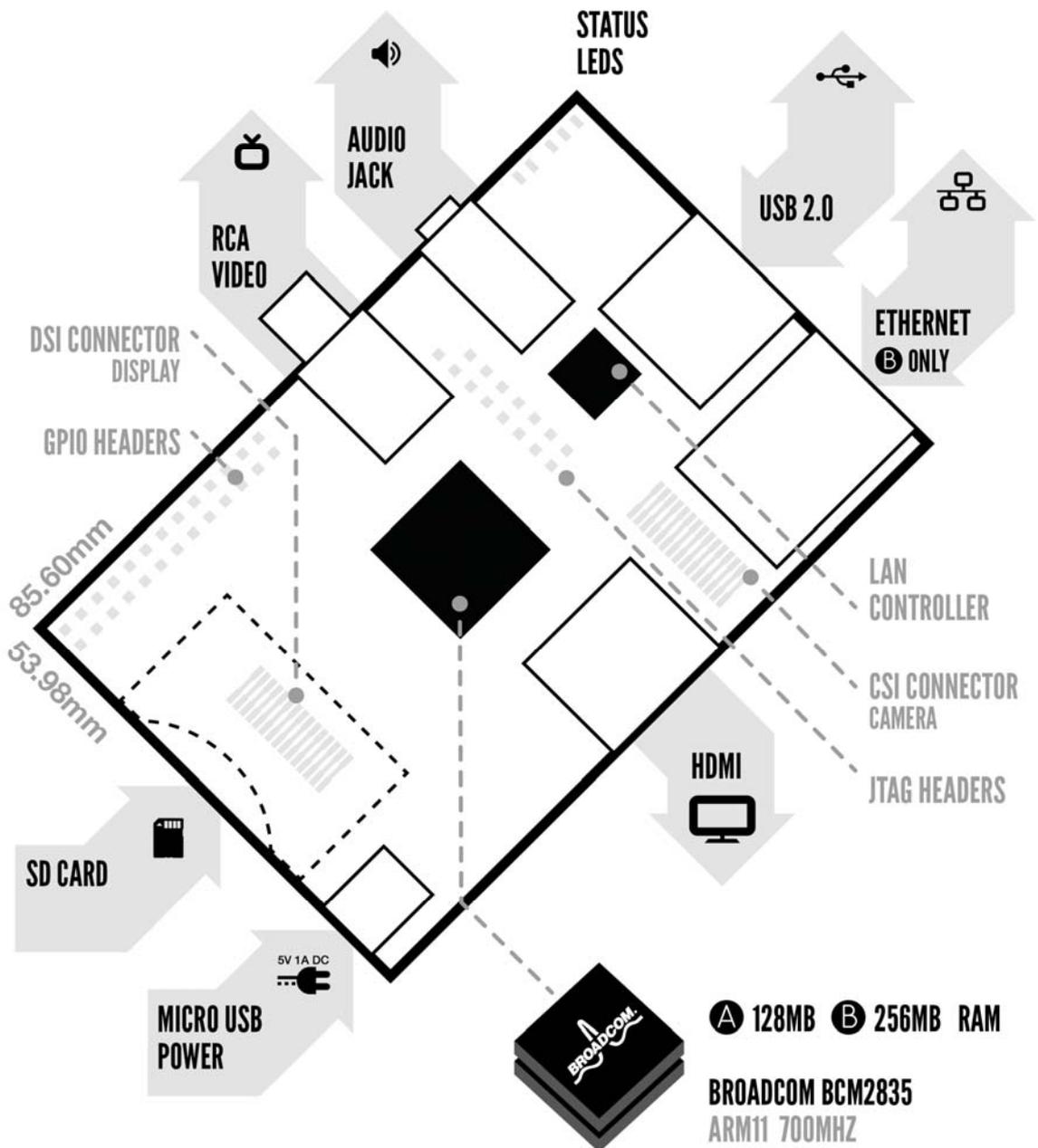
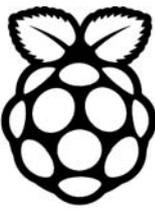
Deux déclinaisons du Raspberry Pi sont commercialisées. La seconde est mieux dotée, car elle bénéficie d'un port Ethernet 100 Mb/s. Si le modèle A est très orienté vers la programmation et l'usage en mode console, voire en lecteur multimédia, le modèle B ouvre des perspectives intéressantes pour un usage de serveur hébergé à domicile.

La consommation électrique est de l'ordre de 3,5 W pour le modèle B, en l'absence de périphériques et d'over-clocking.

2.1 Les constituants :

# Raspberry Pi

Model **A** **B**



 D'après le synoptique d'implantation (page précédente) et la documentation annexe compléter le tableau suivant

<i>Acronyme</i>	<i>définition</i>	<i>fonction</i>
GPIO header	<i>General Purpose In/Out</i>	<i>Entrée sortie pour un usage général</i>
JTAG	<i>Joint Test Action Group</i>	<i>Tester et vérifier la bonne connexion des périphériques.</i>
Micro USB power		<i>Essentiellement utilisé pour l'alimentation de la carte (5v)</i>
DSI connector display	<i>Display serial interface</i>	<i>Connexion à un écran</i>
LEDs d'états	<i>ACT: Activity</i>	<i>Indique l'accès a la carte SD</i>
	<i>PWR: POWER</i>	<i>Mise sous tension</i>
	<i>FDX: Full Duplex</i>	<i>Indique que le réseau fonctionne en Full Duplex</i>
	<i>LNK:LiNK</i>	<i>Indique que la connexion au réseau est établie.</i>
	<i>100: 100Mbit/s</i>	<i>Eteinte: débit a 10Mbit/s Allumé: debit à 100Mbit/s</i>
HDMI	<i>High Definition Multimedia Interface</i>	<i>Connexion audio/vidéo haute définition, tout numérique</i>
CSI	<i>Camera serial interface</i>	<i>Connexion à une camera numérique</i>
LAN	<i>L area network</i>	<i>Connexion Ethernet</i>
Audio Jack		<i>Connexion audio analogique</i>
RCA video		<i>Connexion vidéo analogique, dit « composite »</i>

## 2.2 Présentation du GPIO

D'après la documentation du GPIO combien y a-t-il d'entrée/sortie logique quels sont les principaux protocoles communicant géré par la carte

wiringPi Pin	BCM GPIO	Name	Header	Name	BCM GPIO	wiringPi Pin
-	-	3.3v	1 2	5v	-	-
8	R1:0/R2:2	SDA0	3 4	DNC	-	-
9	R1:1/R2:3	SCL0	5 6	0v	-	-
7	4	GPIO7	7 8	TxD	14	15
-	-	DNC	9 10	RxD	15	16
0	17	GPIO0	11 12	GPIO1	18	1
2	R1:21/R2:27	GPIO2	13 14	DNC	-	-
3	22	GPIO3	15 16	GPIO4	23	4
-	-	DNC	17 18	GPIO5	24	5
12	10	MOSI	19 20	DNC	-	-
13	9	MISO	21 22	GPIO6	25	6
14	11	SCLK	23 24	CE0	8	10
-	-	DNC	25 26	CE1	7	11
wiringPi Pin	BCM GPIO	Name	Header	Name	BCM GPIO	wiringPi Pin

- Définir les différents ports communicants et leurs protocoles, Compléter le tableau suivant :

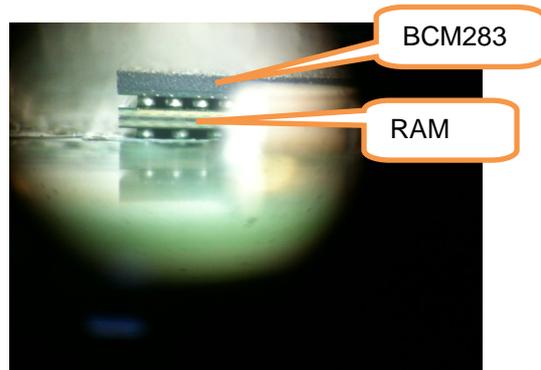
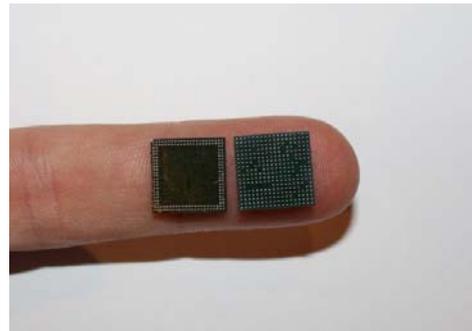
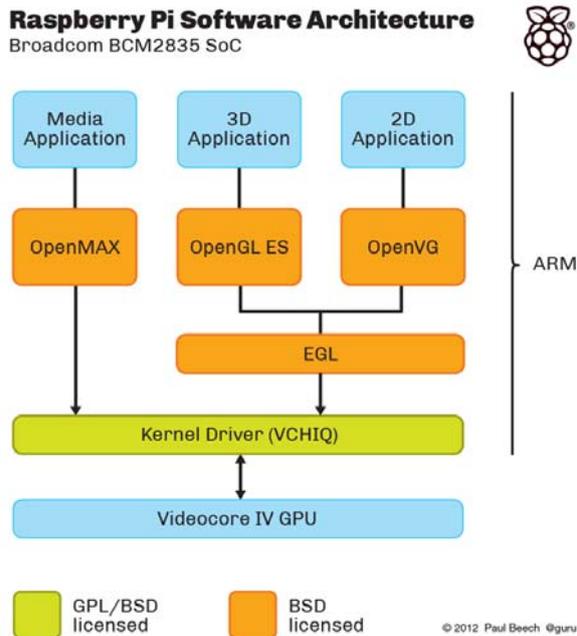
Noms des broches	type de liaison	protocole
SDA0/SCL0		
TX/RX	Liaison série asynchrone	RS232
MOSI/MISO/SCLK/CE0/CE1		
DNC		
GPIO 0 à 7		

- Combien y a-t-il de liaison série, de liaison parallèle, d'entrée/sortie logique, de convertisseur Analogique/ numérique ? Compléter le tableau suivant.

Nb port série	Nb port //	Nb port TTL	Nb port CAN	

# 3 Le SOC Broadcom BCM2835

Comme les briques Lego<sup>®</sup>, les circuits intégrés sont un empilement de plusieurs circuits spécialisés. Le BCM2835 en fait état, elle intègre 3 puces, le GPU (VideoCore IV), CPU (ARM11), RAM (256/512 Mo)



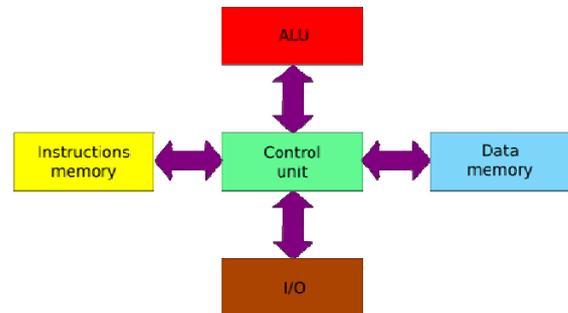
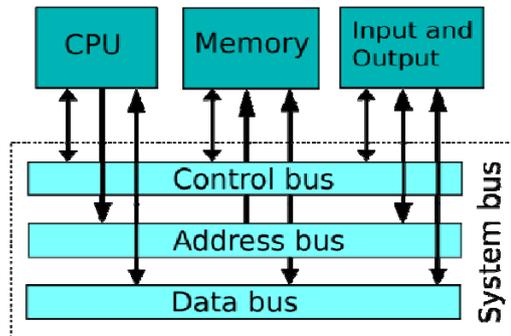
# 4 Le processeur ARM :

## 4.1 Deux familles de microprocesseurs / microcontrôleurs (micros) :

- Les micros **CISC** (Complete Instruction Set Computers) possèdent un grand nombre d'instructions (plusieurs dizaines, voir centaines) permettant de réaliser des opérations très variées. Chaque instruction est exécutée en un certain nombre de cycles d'horloge, dépendant de l'instruction elle-même, mais aussi (et surtout) du mode d'adressage utilisé. Ex : x86, 68hc11
- L'autre famille de micros, **RISC** (Reduced Instruction Set Computers) possède un nombre d'instructions bien plus faible (quelques dizaines au maximum) qui permettent de réaliser des opérations simples (opérations logiques, sommes, etc...). Mais chaque instruction est exécutée en un seul cycle d'horloge. Il en résulte une vitesse de calcul très supérieure à celle de la famille CISC. Ex : les microprocesseurs de Mac, les Pic, les processeurs Alpha

## 4.2 Deux architectures complètent ces familles :

- **Von Neumann** : dont les codes opérations (opcodes) et les paramètres sont stockés les uns derrière les autres dans le programme. Ex x86, 68HC11
- **Harvard** : les opcodes et leurs paramètres sont situés à des endroits différents. Ex : Les Pic



D'après les documents (annexes 1 et 3), définissez la famille et l'architecture de la carte Raspberry :

## 4.3 Taxonomie des processeurs :

Les processeurs sont classés en 4 catégories (dit Taxonomie de Flynn)

**SISD** (instruction simple, une seule mémoire) Il s'agit d'un ordinateur séquentiel qui n'exploite aucun parallélisme, tant au niveau des instructions qu'à celui de la mémoire. Cette catégorie correspond à l'architecture de von Neumann.

**SIMD** (instruction simple, plusieurs mémoires) Il s'agit d'un ordinateur qui utilise le parallélisme au niveau de la mémoire, par exemple le processeur vectoriel].

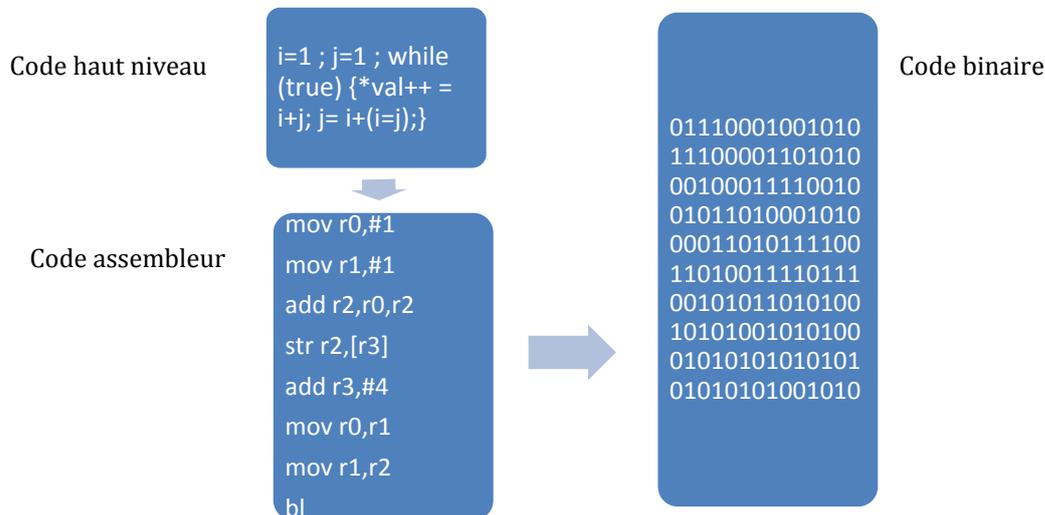
**MISD** (instructions multiples, une seule mémoire) Il s'agit d'un ordinateur dans lequel une même donnée est traitée par plusieurs processeurs en parallèle. Il existe peu d'implémentations en pratique. Cette catégorie peut être utilisée dans le filtrage numérique et la vérification de redondance dans les systèmes critiques.

**MIMD** (instructions multiples, plusieurs mémoires) Dans ce cas, plusieurs processeurs traitent des données différentes, car chacun d'eux possède une mémoire distincte. Il s'agit de l'architecture parallèle la plus utilisée — [Multiple Instructions on Multiple Data](#).

✍ A votre avis, quelle est la catégorie du processeur ARM :

## 5 Logiciels

Au premier abord la carte raspberryPi n'est pas accessible directement. Il est possible de la programmer sous différents langages haut niveau , C + +, Java, C #, Basic, etc..., qu'il faut ensuite compiler en code assembleur avant d'être interprété en code binaire.



Le code binaire est ce que l'ordinateur comprend, mais il est presque impossible pour l'humain de lire. Code assembleur est beaucoup plus performant, mais le nombre de commande est limité.

Pour faciliter la programmation la carte et compte tenu de sa puissance de calcul, on lui adjoint un OS (operating system). Du fait de son architecture ARM et au regard de ces caractéristiques, le choix des logiciels est plus limité que pour les PC et présente avec son lot d'inconvénients : pas d'auto-détection du matériel, quelques binaires opaques, une partie des spécifications gardée secrète...

La fondation propose deux distributions GNU/Linux spécialement compilées pour cet ordinateur. Fedora 13, Debian 6, raspbian (version debian optimisée).

## 6 Préparer la carte SD

La carte SD jouant le rôle de mémoire morte sur le Raspberry Pi, tout comme un disque dur d'ordinateur. Nous y stockerons le système d'exploitation ainsi que les documents, photos, musiques, vidéos... (Pour plus de capacité il est possible d'ajouter via le port usb un Hub-USB ainsi qu'un disque dur).

Pour gagner du temps l'OS raspbian est déjà téléchargé et disponible sur le compte commun (TSTI2D).



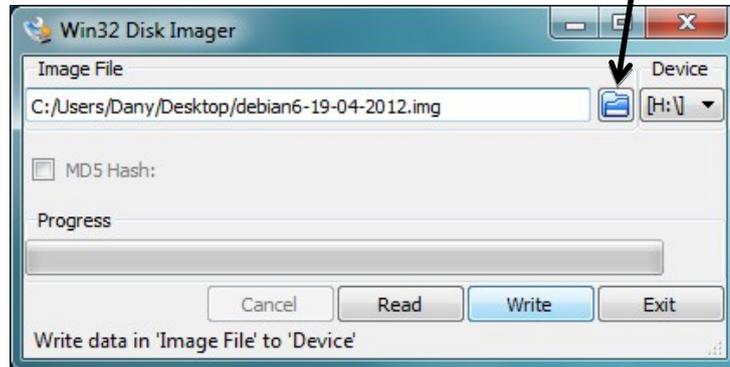
Insérer la carteSD avec son adaptateur sur un port USB du PC





Lancer le logiciel **Win32 disk imager**

- Vérifier la destination (que la carte SD corresponde bien a la lettre affichée)
- Sélectionner l'image de l'OS à charger (OS-RASPBIAN.img)
- Ecrire le programme -> "write".



Cette étape terminée retirer la carte SD du PC et l'insérer sur le connecteur du Raspberry Pi.

## 7 Connecter la carte du Raspberry Pi aux périphériques

Pré requis avant utilisation:

- Clavier et une souris USB. (sans fil éventuellement)
- Préparé carte SD (voir ci-dessus).
- Un adaptateur HDMI et un moniteur (HDMI/DVI).
- D'une alimentation Micro USB - 700mA / 5V. (chargeur de portable)
- Câble LAN Ethernet (en option).

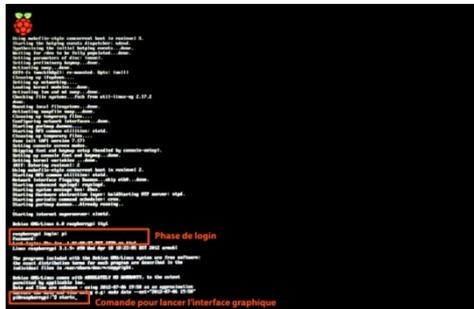


Branchez alors le câble HDMI sur votre écran, le clavier et la souris sur les ports USB, le câble Ethernet, et pour finir, l'alimentation (branchez-la sur le secteur). Le système devrait automatiquement démarrer à en juger le clignotement des LEDs.



# 8 Démarrage

Lorsque le système démarre, vous arriverez sur un écran totalement noir avec des inscriptions blanches. C'est normal : c'est ainsi que démarre Linux, le système d'exploitation que l'on a chargé sur la carte SD. Attendez que les instructions aient fini de défiler, et que cette ligne s'affiche :



Saisir alors :

```
Login as : root

Password : root
```

Si tout va bien ces message apparaissent, votre « Framboise est prête ! »

```
login as: root
root@192.168.1.18's password:
Linux raspberrypi 3.2.27+ #250 PREEMP
      T Thu Oct 18 19:03:02 BST 2012 armv6l
The programs included with the Debian
GNU/Linux system are free software;
the exact distribution terms for each
program are described in the
individual files in /usr/share/doc/*/
Debian GNU/Linux comes with ABSOLUTEL
Y NO WARRANTY, to the extent
permitted by applicable law.
Last login: Thu Dec 6 10:21:59 2012
from
root@raspberrypi:~#
```

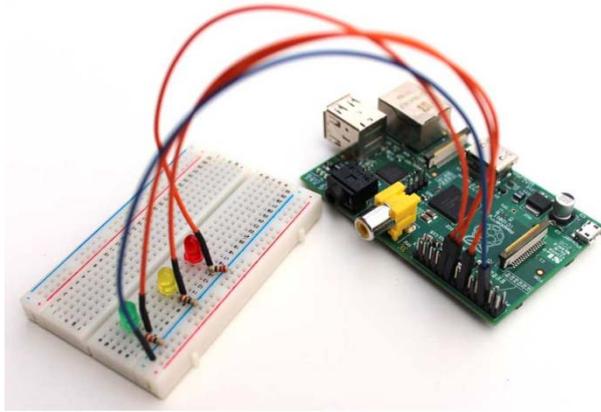
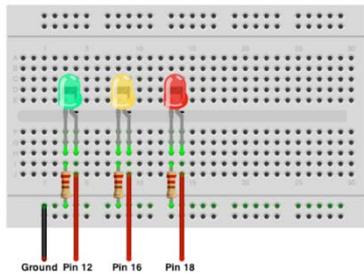
*Cet environnement s'appelle le mode console.*

# 9 Test du GPIO :



Pour ce test, nous câblerons 3 Leds sur les broches 12, 14, 18

	5V	-	Ground	GPIO14	GPIO15	GPIO18	-	GPIO23	GPIO24	-	GPIO25	GPIO8	GPIO7
	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑
Pin	2	4	6	8	10	12	14	16	18	20	22	24	26
Pin	1	3	5	7	9	11	13	15	17	19	21	23	25
	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓
	3v3	GPIO0	GPIO1	GPIO4	-	GPIO17	GPIO21	GPIO22	-	GPIO10	GPIO9	GPIO11	-



Instructions elementaires pour commander le GPIO, exemple donné pour le GPIO18 :

- Dans un premier temps, il faut déclarer la sortie 18 du gpio en mode lecture ecriture:

```
root@raspberrypi:~# gpio -g mode 18 out
```

- pour allumer la LED,

```
root@raspberrypi:~# gpio -g write 18 1
```

- pour eteindre la LED

```
root@raspberrypi:~# gpio -g write 18 0
```

- lecture de l'etat du GPIO18:

```
root@raspberrypi:~# gpio -g read 18
```

la console retourne alors la valeur du port 0 ou 1

~~☒~~ de quelle LED s'agit il ?

- ~~☒~~ Compléter le tableau suivant :

LED	broche	GPIO
verte	12	
rouge	16	
jaune	18	

- ~~☒~~ Tester la Led jaune et donner les lignes de commandes necessaire :

Led jaune	Commande write	Commande read
allumée		
eteinte		

# 10 Vérification de la connexion réseau

- Identifier sa carte (IP)

```
root@raspberrypi:~# ifconfig
```

```
eth0      Link encap:Ethernet  HWaddr b8:27:eb:fa:8c:8d
          inet adr:192.168.1.18 Bcast:192.168.1.255 Masque:255.255.255.0
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:6458 errors:0 dropped:0 overruns:0 frame:0
          TX packets:4413 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 lg file transmission:1000
          RX bytes:388590 (379.4 KiB)  TX bytes:2397723 (2.2 MiB)
```

- Pinguer les autres cartes branchées sur le réseau:

```
root@raspberrypi:~# ping -nc4 172.17.____
```

```
PING 172.17.75.105 (172.17.75.105) 56(84) bytes of data.
64 bytes from 172.17.75.105: icmp_req=1 ttl=60 time=2.71 ms
64 bytes from 172.17.75.105: icmp_req=2 ttl=60 time=1.37 ms
64 bytes from 172.17.75.105: icmp_req=3 ttl=60 time=1.44 ms
64 bytes from 172.17.75.105: icmp_req=4 ttl=60 time=1.28 ms

--- 172.17.75.105 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3004ms
rtt min/avg/max/mdev = 1.289/1.706/2.712/0.584 ms
```

- ✎ Expliquer la ligne suivante :

```
min/avg/max/mdev = 1.289/1.706/2.712/0.584 ms
```

- ✎ compléter le tableau suivant en indiquant l'adresse Mac, IP et le temps moyen du ping pour chaque carte Raspberry connectée au réseau

groupe	@mac	IP	Temps moyen
1			
2			
3			
4			
5			
6			
7			
8			