

Systemes embarqués pour objets connectés

SEOC3



IUT de Béziers, dépt. R&T © 2018

<http://www.borelly.net/>

Christophe.BORELLY@umontpellier.fr

Généralités

- Wikipedia : « Un **système embarqué** est un système électronique et informatique **autonome**, souvent **temps réel**, spécialisé dans une tâche bien précise. »
- Ses ressources sont généralement limitées :
 - Encombrement réduit
 - Consommation restreinte
 - Espace mémoire limité
 - Puissance de calcul
 - ...

Généralités

- Les systèmes embarqués utilisent généralement des **microprocesseurs** à basse consommation d'énergie ou des **microcontrôleurs** dont la partie logicielle est stockée dans une mémoire **morte** en lecture seule (ROM), EPROM, EEPROM, FLASH, etc.
- On parle alors de **firmware**.
- Avec la miniaturisation des composants, on peut avoir le système embarqué complet sur une seule puce (**SoC** - System on a Chip).

Microprocesseurs

Microcontrôleurs

▪ Microprocesseurs

- Multi-tache
- Ajout de périphériques possible
- Consommation plus grande
- Prix plus élevé

▪ Microcontrôleurs

- Mono-tache (e.g.)
- Périphériques en nombre limité et embarqués dans la puce
- Consommation plus petite
- Prix plus bas

Contenu d'un SoC

- CPU (Central Processing Unit)
- Mémoire RAM
- Ports d'entrée/sortie (USB, GPIO, I2C, SPI, ...)
- Stockage de données (MicroSD)
- GPU (graphics processing unit)
- Modules Wifi et/ou Bluetooth
- ...

Principaux processeurs SoC

- Processeurs **ARM** :

- **Exynos** de Samsung
- **MediaTek, Rockchip**
- **OMAP** de Texas Instruments
- **Tegra** de Nvidia
- **Snapdragon** de Qualcomm
- **STM32** de STMicroelectronics
- ...

- Processeurs **Cell** :

- Sony, Toshiba, IBM

- Processeurs d'images :

- **DIGIC** de Canon
- **EXPEED** de Nikon
- ...

- Processeurs **x86** :

- **Atom** de Intel
- **AMD**
- ...

Architecture ARM

- Acorn RISC Machine depuis 1980
 - RISC : Reduced Instruction Set Computing
- 100 milliards de processeurs produits en 2017
- 32 bits jusqu'à ARMv7
 - Cortex-A7, A12, A17, A32, ...
- 64 bits depuis ARMv8
 - Cortex-A53, A55, A57, A72, A73, A75, A76, ...

Nano-ordinateurs

- Systèmes embarqués (au format carte de crédit) utilisant un système d'exploitation :
 - Raspberry pi
 - Beaglebone
 - Cubieboard
 - ODROID
 - Banana Pi
 - Tinker Board d'Asus
 - ...

Raspberry PI

- Conçu par David BRABEN
- Depuis février 2012
- Prix de 25 à 35 \$
- Modèles :
 - PI ZERO, PI ZERO W
 - PI 1A+, PI 1B+
 - PI 2B, **PI 3B** et PI 3B+



Raspberry PI 2B (2015)

- Broadcom BCM2836 (900MHz quad-core ARM Cortex-A7)
- 1GB RAM
- 100 Mbps Ethernet
- 4 USB ports
- 40 GPIO pins
- Full HDMI port
- Combined 3.5mm audio jack and composite video
- Camera interface (CSI)
- Display interface (DSI)
- Micro SD card slot
- VideoCore IV 3D graphics core



Raspberry Pi 3B (2016)

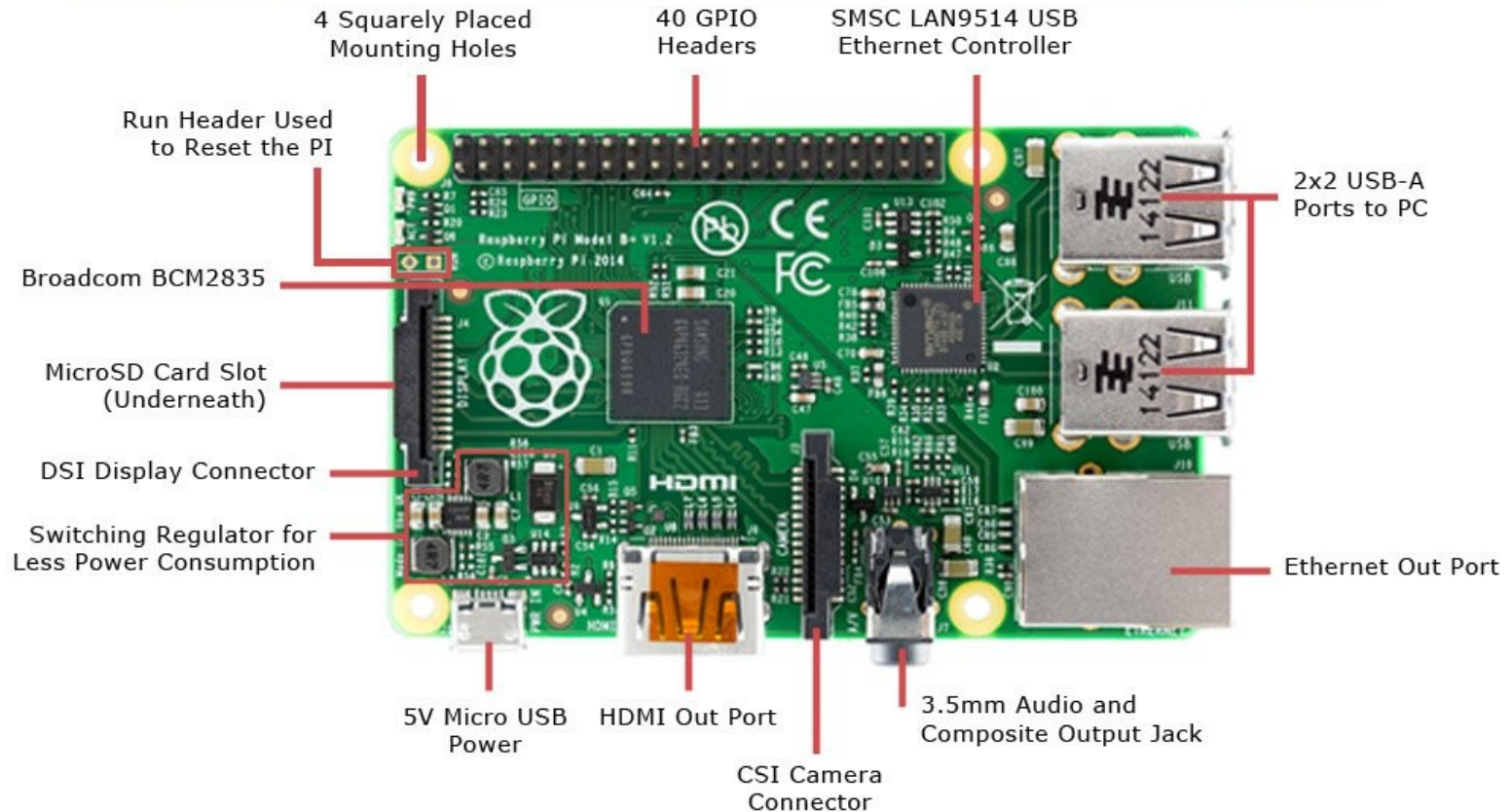
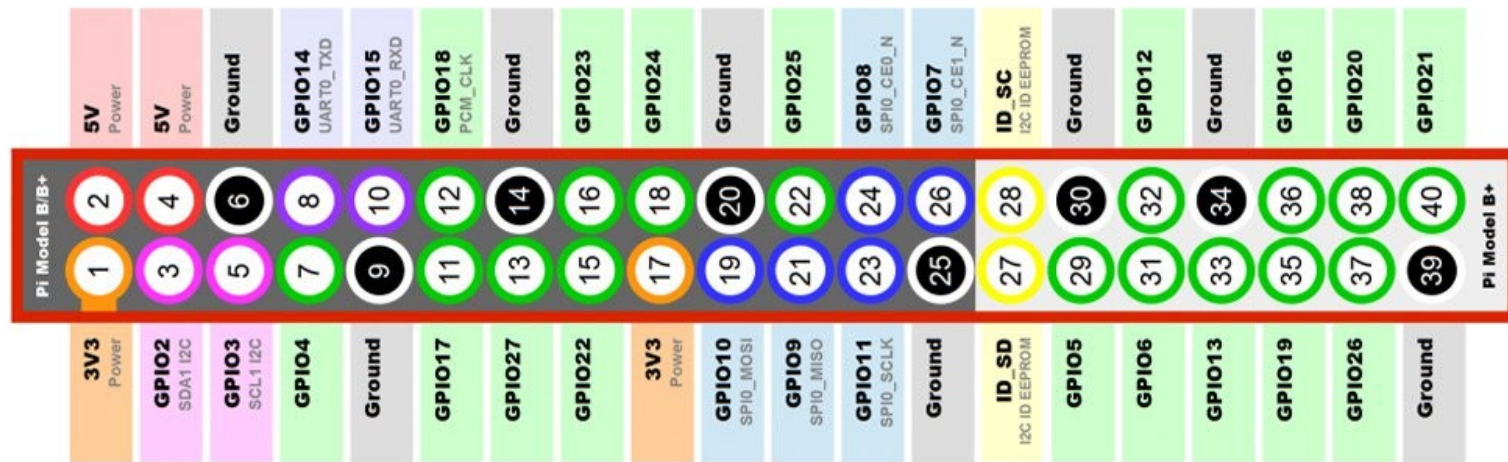
- Broadcom BCM2837 (1.2GHz quad-core ARM Cortex-A53 64bit)
- 1GB RAM
- BCM43438 IEEE 802.11n wireless LAN
- Bluetooth Low Energy (BLE) on board
- 100 Base Ethernet
- 40-pin extended GPIO
- 4 USB2 ports
- 4 Pole stereo output and composite video port
- Full size HDMI
- CSI camera port for connecting a Raspberry Pi camera
- DSI display port for connecting a Raspberry Pi touchscreen display
- Micro SD port for loading your operating system and storing data
- Upgraded switched Micro USB power source up to 2.5A



Raspberry PI 3B+ (2018)

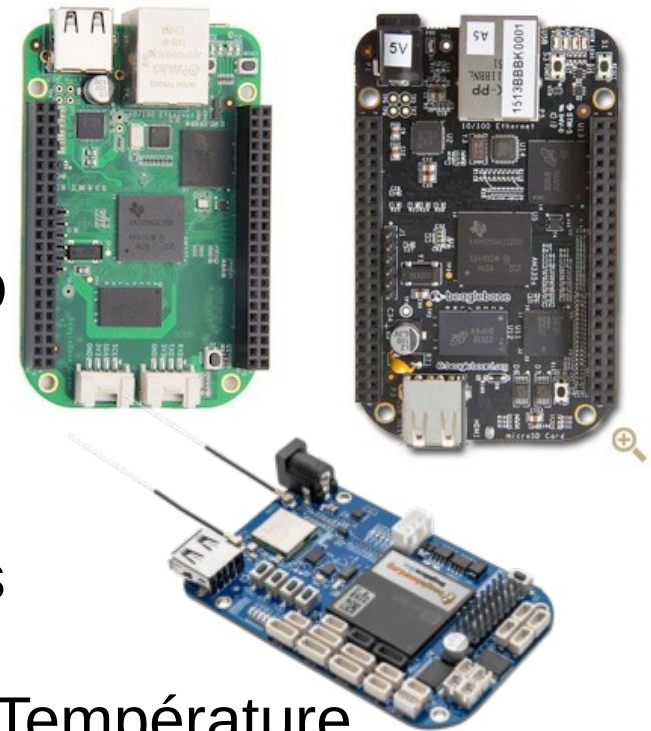
- Broadcom BCM2837B0 (1.4GHz Cortex-A53 (ARMv8) 64-bit)
- 1GB LPDDR2 SDRAM
- 2.4GHz and 5GHz IEEE 802.11.b/g/n/ac wireless LAN
- Bluetooth 4.2, BLE
- Gigabit Ethernet over USB 2.0 (maximum throughput 300 Mbps)
- Extended 40-pin GPIO header
- Full-size HDMI
- 4 USB2 ports
- CSI camera port for connecting a Raspberry Pi camera
- DSI display port for connecting a Raspberry Pi touchscreen display
- 4-pole stereo output and composite video port
- Micro SD port for loading your operating system and storing data
- 5V/2.5A DC power input
- Power-over-Ethernet (PoE) support (requires separate PoE HAT)

GPIO Pinout Diagram



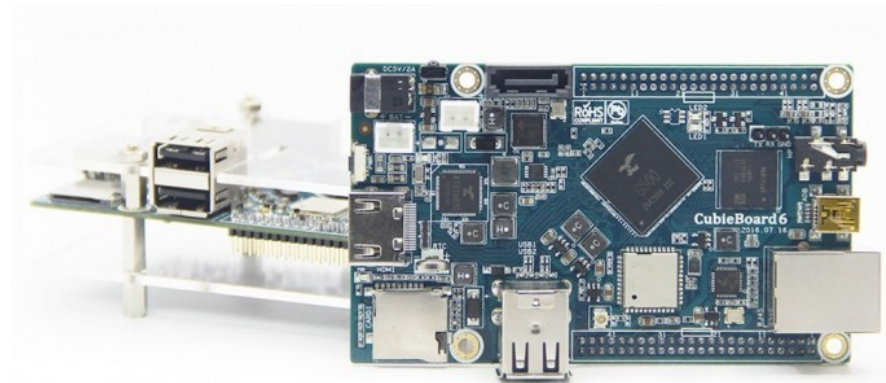
Beagleboard / Beaglebone

- Matériel libre produit par Texas Instruments en collaboration avec Digi-Key depuis 2008
- Modèles :
 - black, green, blue, Enhanced
- AM3358 (ARM Cortex-A8 @ 1 GHz)
- Mémoire de 512 MB à 4 GB + slot microSD
- USB 2.0 et microHDMI
- Ports : 7 Analogiques et 65 Numériques
- 4x UART, 12x PWM/Timers
- 2x SPI, 2x I2C, A/D Converter, 2x CAN Bus
- Capteurs intégrés suivant modèle :
 - Baromètre, Accéléromètre, Gyroscope, Température
- Prix de 39 à 79 \$



CubieBoard

- Fabriqué en Chine depuis 2012
- Modèles : CubieBoard1 à CubieBoard6
- Prix **CB6** env. 70 \$
- CPU Cortex A9 quad-core
- GPU Power VR SGX544
- 2GB RAM LPDDR3
- 8GB eMMC + microSD
- USB 2.0, HDMI, Wifi, Bluetooth
- Ethernet 10/100 Mbps
- ...



ODROID

- Modèles : XU4, HC2, HC1, MC1, C2, C1+, C0
- **ODROID-XU4** (60 \$)
 - Samsung Exynos5422 Cortex-A15 2Ghz and Cortex-A7 Octa core CPUs
 - Mali-T628 MP6 (OpenGL ES 3.1 and OpenCL 1.2)
 - 2GB LPDDR3 RAM PoP stacked
 - 8GB eMMC + microSD
 - 2 x USB 3.0 Host, 1 x USB 2.0 Host
 - Gigabit Ethernet port
 - HDMI 1.4a for display
 - Power: 5V/4A input



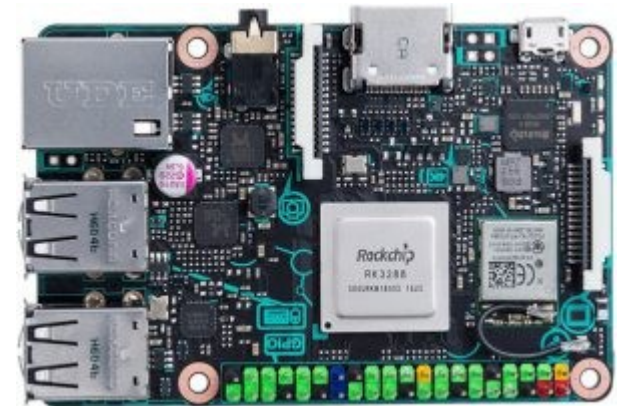
Banana Pi

- Développé par SINOVOIP en Chine depuis 2003
- Modèles : M1, M2, M3, M64, R1, R2, R64 (multimedia network router)...
- **BPI M64** (2016 - 80 €)
 - CPU Allwinner 64 Bit Quad Core ARM Cortex-A53 @ 1.2 GHz
 - GPU Dual core Mali 400 MP2
 - Memory 2GB LPDDR3
 - MicroSD slot + 8G eMMC flash
 - Network 10/100/1000 Mbit/s Ethernet
 - Wi-Fi 802.11 b/g/n + Bluetooth 4.0
 - Video Input(s) A CSI input connector
 - Video Output(s) 1080p capable HDMI port
 - Audio Input(s) On board microphone
 - Audio Output(s) 3.5mm jack and HDMI
 - USB 2.0 (x2), USB OTG (x1)
 - Remote IR Receiver
 - GPIO 40 Pin Header: GPIO (x28) and Power (+5V, +3.3V and GND).



Tinker Board d'Asus

- Sortie en 2017 (68 \$)
- SoC Rockchip RK3288
- CPU ARM Cortex-A17 (32-bit) Quadricoeur @ 1,8 GHz
- GPU ARM Mali-T764
- Mémoire vive 2 Go double-canal LPDDR3
- Prise USB 4 prises USB 2.0
- Plateau de 40 broches dont 28 GPIO
- Carte microSD conforme au UHS-I SDIO 3.0
- MIPI-DSI (MIPI Display Serial Interface)
- Sortie vidéo HDMI H.264 jusqu'à 4K H.265
- Entrée vidéo 1 x 15 broches pour caméra MIPI-CSI
- Gigabit Ethernet RTL8211E-VB-CG
- Bluetooth 4.0 + EDR
- WiFi 802.11 b/g/n, module AW-NB177NF contenant un Realtek RTL8723BS



Distributions Linux ARM

- <https://distrowatch.com/search.php?architecture=arm>
- Environ 55 distributions Linux :
 - Debian (raspbian)
 - Ubuntu
 - Fedora
 - Arch Linux
 - Slackware
 - OpenSUSE
 - Gentoo (pour utilisateurs Linux avertis)
 - ...

GNU/Linux : avantages et inconvénients

- Installation facile
- Idéal pour débiter et tester
- Empreinte mémoire importante
- Temps de démarrage plus ou moins rapide
- Environnement de développement avant tout
- Diminution des vecteurs d'attaques potentiels :
 - Suppression des paquets inutiles (pas si facile !)
 - Attention aux dépendances et aux mises à jour

Installation de raspbian

```
wget --content-disposition  
https://downloads.raspberrypi.org/raspbian_lite_latest  
sha256sum *-raspbian-*-lite.zip  
unzip *-raspbian-*-lite.zip  
dd if=*-raspbian-*.img of=/dev/mmcblk0 bs=4M conv=fsync
```

- Le device cible est déterminé avec **lsblk**

Séquence de démarrage d'un RPi

- A la mise sous tension, le CPU est hors ligne.
- C'est le GPU qui est chargé de démarrer le SoC.
- 1 étape : Chargement du bootloader depuis la ROM.
 - Permet de monter la partition boot (FAT32) de la carte SD.
- 2 étape (**bootcode.bin**) : Charge le firmware GPU (start.elf) depuis la carte SD.
- 3 étape (**start.elf**) : Démarrage du CPU.
 - Lecture de config.txt, cmdline.txt et *.dtb (Device Tree Blob)
 - fixup.dat : répartition de la SDRAM entre GPU et CPU.
- Exécution du programme utilisateur :
 - En général : **kernel.img**

Source: https://elinux.org/RPi_Software

Rpi firmware : <https://github.com/raspberrypi/firmware>

Fichier cmdline.txt

- Options fournies au kernel Linux

```
dwc_otg.lpm_enable=0 console=serial0,115200  
console=tty1 root=...
```

- Adresse IP fixe (Ajouter au début de la ligne) :
 - **ip**=<client-ip>:<dns-server-ip>:<gw-ip>:<netmask>:<hostname>:<device>:<autoconf>
 - Exemple :
ip=192.168.1.15:192.168.1.53:192.168.1.254:255.255.255.0

Fichier config.txt

- Remplace le BIOS des ordinateurs classiques
- Active l'I2C sur les GPIO :

`dtparam=i2c_arm=on`

- Active SPI :

`dtparam=spi=on`

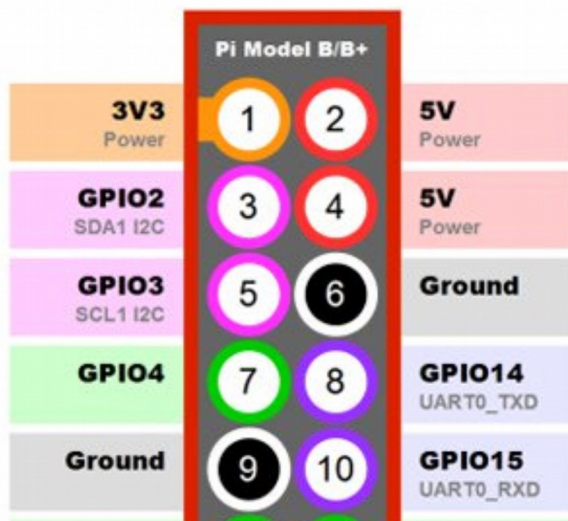
- Active le module camera :

`start_x=1`

...

Console de débogage série

- Câble USB to TTL Serial
- Rpi 3 boot/config.txt : `enable_uart=1`



Source : https://elinux.org/RPi_Serial_Connection

Paramétrage d'une image avec un « chroot »

- Utilisation du système Linux hôte
- Permet de tout préparer sans utiliser de Rpi
- Déploiement accéléré sur une flotte de Rpi
- Nécessite un émulateur ARM (qemu)
- Montage de l'image dans un périphérique de bouclage (losetup)
- Montage des dossiers spéciaux du système (proc, sys, dev)
- Chroot = Changement de répertoire racine

Commandes à utiliser

```
apt install qemu qemu-user-static binfmt-support
losetup -P /dev/loop0 rpi.img
mkdir /mnt/rpi
mount /dev/loop0p2 /mnt/rpi
mount /dev/loop0p1 /mnt/rpi/boot
mount --bind /proc /mnt/rpi/proc/
mount --bind /sys /mnt/rpi/sys/
mount --bind /dev /mnt/rpi/dev/
mount --bind /dev/pts /mnt/rpi/dev/pts
cp /usr/bin/qemu-arm-static /mnt/rpi/usr/bin/
chroot /mnt/rpi /bin/bash
...
exit
umount /mnt/rpi/{dev/pts,dev,sys,proc,boot,}
losetup -d /dev/loop0
```

Build system

- Compilation d'une distribution à partir des sources



- Yocto/OpenEmbedded (<https://www.yoctoproject.org/>)

- Buildroot (<https://buildroot.org/>)



- OpenWrt (<https://openwrt.org/>)



- ...

Composants d'un « build system »

- Chaîne de compilation croisée (cross-compiler)
- Un ou plusieurs bootloaders
- Un noyau Linux et les firmwares associés
- Un système de fichiers avec bibliothèques, outils et utilitaires
- Une ou plusieurs applications personnalisées

Paquets nécessaires à Yocto

```
apt install gawk wget git-core diffstat unzip  
texinfo gcc-multilib build-essential chrpath  
socat
```

- Graphique :

```
apt-get install libsdl1.2-dev xterm
```

- Documentation :

```
apt install make xsltproc docbook-utils fop dblatex  
xmlto
```

- Applications avec ADT :

```
apt install autoconf automake libtool libglib2.0-dev
```

Exemple x86 minimal

```
git clone git://git.yoctoproject.org/poky
```

```
cd poky
```

```
source oe-init-build-env qemux86-build
```

```
bitbake core-image-minimal
```

```
runqemu nographic qemux86
```

```
...
```

```
Poky (Yocto Project Reference Distro) 2.5+snapshot qemux86 /dev/ttyS0
```

```
qemux86 login: root
```

```
root@qemux86:~# df -h
```

Filesystem	Size	Used	Available	Use%	Mounted on
/dev/root	9.4M	7.1M	1.5M	82%	/
devtmpfs	119.4M	0	119.4M	0%	/dev
tmpfs	119.9M	104.0K	119.8M	0%	/run
tmpfs	119.9M	84.0K	119.8M	0%	/var/volatile

Liste des images possibles

- **core-image-minimal** A small image just capable of allowing a device to boot (no SSH).
- **core-image-lsb** An image containing packages that are required to conform to the Linux Standard Base (LSB) specification. (+OpenSSH)
- **core-image-full-cmdline** A console-only image with more full-featured Linux system functionality installed. (+OpenSSH)
- **core-image-sato** Image with Sato, a mobile environment and visual style for mobile devices.(+Dropbear SSH server)
- ...

Voir : <https://layers.openembedded.org/layerindex/branch/master/layer/openembedded-core/>

« build » pour raspberry

```
git clone git://git.yoctoproject.org/poky
cd poky
git clone git://git.yoctoproject.org/meta-raspberrypi
source oe-init-build-env rpi-build
bitbake-layers add-layer ../meta-raspberrypi
echo "MACHINE = \"raspberrypi3\"" >> conf/local.conf
bitbake core-image-minimal
cd tmp/deploy/images/raspberrypi3/
fdisk -l core-image-minimal-raspberrypi3.rpi-sdimg
Disque core-image-minimal-raspberrypi3.rpi-sdimg : 108 MiB, 113246208 octets,
221184 secteurs
Unités : secteur de 1 × 512 = 512 octets
Taille de secteur (logique / physique) : 512 octets / 512 octets
taille d'E/S (minimale / optimale) : 512 octets / 512 octets
Type d'étiquette de disque : dos
Identifiant de disque : 0x6f0af9cd
```

Périphérique	Amorçage	Début	Fin	Secteurs	Taille	Id	Type
core-image-minimal-raspberrypi3.rpi-sdimg1 *		8192	90111	81920	40M	c	W95
core-image-minimal-raspberrypi3.rpi-sdimg2		90112	221183	131072	64M	83	Linux

Références

- <https://fr.wikipedia.org/wiki/Nano-ordinateur>
- <https://www.arm.com/>
- <https://www.raspberrypi.org/>
- <https://beagleboard.org/bone>
- <http://cubieboard.org/>
- <https://www.hardkernel.com/>