

LABORATOIRE D'ELECTRONIQUE

APPLIQUEE Q2

Cours destiné aux étudiants de première année en
Technologie de l'Informatique et orientation
sécurité des systèmes

Table des matières

Laboratoire sur Raspberry (Configuration).....	3
Petits rappels (Raspberry) :	3
Manip 1 : Installation de la RaspBerry	4
Considérations pratiques :	4
Récupération de l'image de l'OS :	4
Implantation de l'OS :	4
Configuration réseau :	5
Connaître l'adresse IP de votre RPI (DHCP) :	5
Expansion du Filesystem :	10
Sauvegarde de votre travail :	10
Manip 2 : Arduino sur RPI	11
Pense-bête :	12

Laboratoire sur Raspberry (Configuration)

Petits rappels (Raspberry) :

- Attention : La RPI¹ est alimentée en 5V par un adaptateur en 5V DC. **MAIS ATTENTION** : les composants (ce qui est connecté sur la GPIO) travaillent en 3,3V. Ceci signifie que c'est bien cette tension qu'il faut prendre en compte pour les calculs (de résistance sur une LED, par exemple)
- Attention : Une Led se raccorde **toujours** avec une résistance en série !
- La tension d'une Led varie en fonction de la couleur. (Rappel : Led verte = 2V, 20mA)
- La Raspberry est un SoC². Il faut donc installer un OS³ sur le support du système de fichiers.
- Sur la RPI, il n'y a pas, à priori, de disque dur. L'OS et tous les autres fichiers sont sur un support mémoire : la carte micro-SD.
- Pour l'installation de l'OS sur la carte mémoire, il faut graver une image.
- Installation d'une RPI **sans** intervention sur la **configuration des PCs du laboratoire**.
- L'installation d'une Raspberry peut se faire avec une intervention minimale sur une configuration quelle qu'elle fut. Plus qu'une contrainte, la démarche proposée ici offre un net avantage en terme de temps et de confort. Les seules connexions à réaliser seront l'alimentation et le réseau. On peut même pousser la simplification à la seule connexion de l'alimentation, si on est dans un environnement wifi disponible.

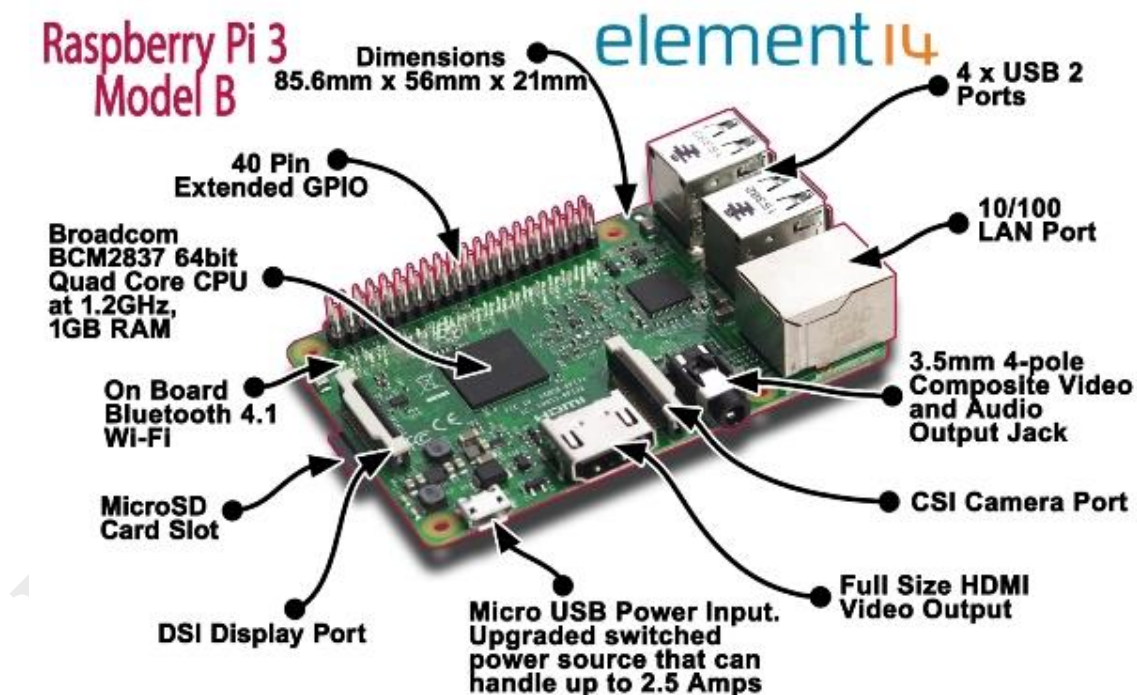


Image venant de : <https://www.element14.com>

¹ Raspberry

² SoC: System on Chip

³ OS: Operating System

Manip 1 : Installation de la RaspBerry

Récupération de la distribution à installer.

Considérations pratiques :

Chaque étudiant a **sa propre** carte micro-SD. Cette carte doit avoir une capacité **d'au moins 8 Go**. Ceci permet de conserver son travail, ses développements et ses configurations. En ce compris une éventuelle adresse IP fixe, voire un script de démarrage envoyant un mail avec l'adresse délivrée par le serveur DHCP.

Plusieurs distributions ou OS sont disponibles au téléchargement. Nous travaillerons de préférence avec une distribution Raspbian. Certaines applications que nous installerons utiliseront une interface graphique (« Arduino » par exemple), c'est pourquoi, nous ne choisirons **pas la version light** de Raspbian.

Récupération de l'image de l'OS :

Les images sont disponibles au téléchargement sur le site de Raspberry à l'adresse <https://www.raspberrypi.org/downloads/>.

Implantation de l'OS :

Une fois l'image récupérée, il faut l'installer sur la carte µSD. Deux possibilités en fonction de l'OS avec lequel on travaille, Windows ou Linux. (**Attention : lire la suite avant de mettre la carte µSD dans la RPI !!!**)

Windows :



Utilisation du logiciel "win32 disk imager" :

Linux :

Lancement en ligne de commande (en sudo):

```
"dd BS=1M if=<chemin et nom du fichier image récupéré sur le site de RPI>  
of=<périphérique carte> conv=fsync"
```

<chemin et nom du fichier récupéré sur le site de RPI> , c'est le chemin et le fichier ...

<périphérique de la carte>, c'est le chemin qui pointe vers le périphérique de la carte mémoire. Pour le trouver, entrez la commande "ls /dev/sd*" et une lecture réfléchie du résultat de sortie devrait vous aider à identifier l'emplacement de la carte dans le système de fichiers.

Avant de mettre la carte µSD dans la RPI, il faut activer le serveur SSH sur la RPI (ce n'est plus activé par défaut). Pour ce faire, **laissez la carte µSD dans l'ordinateur** et ouvrez la clé pour y créer un fichier **“ssh” sans extension à la racine de la partition “boot”** ! Une fois cela fait, Vous pouvez enfin mettre la carte µSD dans la RPI et lister les adresses IP avec "NMAP" (Linux) ou "Angry IP Scanner"(Windows).

Configuration réseau :

Maintenant que le fichier SSH est sur votre partition « boot » de la carte μ SD, vous pouvez la mettre dans la RPI, ensuite branchez le câble réseau RJ45 **et pour finir, branchez l'alimentation.**

Il faudrait se connecter en SSH sur la RPI. (Le SSH est un protocole de communication sécurisé qui passe par le câble RJ45 ou wifi, suivant le cas, entre votre PC et votre RPI. « Sécurisé » car les données en TCP sont cryptées.)

Pour ce faire, soit vous utilisez Putty soit vous utilisez **MRemoteNG.exe** qui se trouve sur Moodle. (Vous pouvez déjà télécharger les autres logiciels portables qui sont avec MRemoteNG car vous en aurez besoin bientôt.)

Connaître l'adresse IP de votre RPI (DHCP) :

Comment retrouver votre adresse quand la RPI n'est pas la seule sur le réseau :

Voici donc plusieurs méthodes pour essayer de la retrouver :

1ère méthode : Utilisez "NMAP" (Linux) ou "Angry IP scanner" (Windows) pour scanner toutes les adresses IP connectées sur le sous réseau du laboratoire.

Le problème est qu'il va falloir trouver votre RPI dans toutes celles qui seront reconnues (avec l'Hostname « raspberry »).

Il faudra donc réaliser un ping sur une adresse IP qui comporte l'hostname avec « raspberry » et débrancher le câble de **votre** RPI pour voir si le ping s'interrompt. Si c'est le cas, c'est bien l'adresse de votre RPI, si ce n'est pas le cas, ce n'est pas votre adresse IP et vous en testez une autre. (Il est intéressant de le faire plusieurs fois pour être sûr que c'est bien l'adresse IP de votre RPI et pas celle de quelqu'un d'autre qui aurait fait la même manip que vous en même temps).

Utilisation de NMAP sous Linux :

Pour plus de facilité par la suite, **prenez une clé USB bootable Ubuntu et démarrez linux sur la tour de l'école.** (F11 ensuite il ne faut pas sélectionner "sata" ni "realteak" mais bien votre clé ou votre DVD ou CD) **NE PAS INSTALLER LINUX SUR LE PC DU LABO !!!**

Changez le clavier en AZERTY Belgique.

Installez NMAP en insérant en ligne de commande "sudo apt-get install nmap".

Utilisez NMAP : `sudo nmap -sn *.*.*./24 | grep -B2 Raspberry`. "B2" pour les 2 lignes précédentes et "Raspberry" pour ce qu'on recherche. Il n'est pas obligatoire de faire une recherche avec grep.

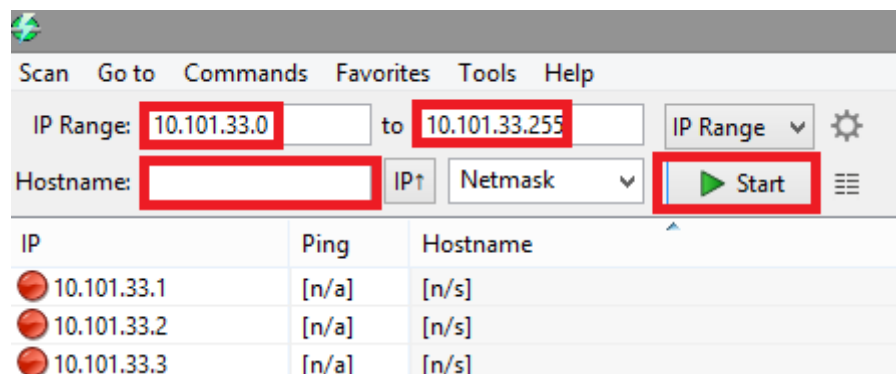


Source : <https://nmap.org/>

La commande renverra les adresses IP présentes sur le réseau, reste à trouver la sienne car au laboratoire il y aura beaucoup de "Raspberry Pi Foundation".

Utilisation de "Angry IP Scanner" sous Windows :

Lancez le logiciel, mettez le « **Hostname** » vide et cliquez sur « start ». Attention, vérifier l'« IP range » des adresses IP à trouver. (Si vous ne connaissez pas le sous réseau, utilisez « ipconfig »)



2ième méthode : L'"hostname" par défaut de la RPI est le même pour tout le monde.

Si vous connaissez l'adresse IP de la RPI, vous pouvez modifier l'"hostname" via des commandes (`sudo raspi-config` ou modifier les fichiers "**hostname**" et "**hosts**") en ssh avec putty ou mRemoteNG.

Si vous ne connaissez pas l'adresse IP de la RPI, il faut modifier les fichiers "**hostname**" et "**hosts**" qui se trouve dans l'OS sur la carte SD. Pour ce faire, vous devez utiliser Ubuntu sur la clé USB et lire la carte SD. Attention au droit d'accès (sudo), il faudra lancer le terminal et aller dans « etc » puis modifier le fichier « hostname ».

Vous pourrez vous assurer que vous êtes sur votre RPI lors de la connexion avec putty. Le "hostname" sera affiché à l'invité de commande. Vérifiez que ce "hostname" est bien le bon dans le cas contraire testez une autre adresse.

3ième méthode : Envoi de l'adresse IP de la RPI sur votre boîte mail. Par sécurité, vous pouvez créer une adresse dédiée à vos applications Raspberry. **Les identifiants sont en clair dans les scripts.** (La classe pourrait également créer une boîte mail pour recevoir les adresses des RPI. A vous d'écrire un objet dans le mail qui vous est propre.) Vous pouvez aussi vous créer une adresse mail à l'usage exclusif de votre Raspberry.

Pour cela, on peut lancer un script au démarrage de la RPI qui peut envoyer un mail vers un destinataire au choix.

Si le script s'appelle « Envoi_IP_Mail.py », il faut configurer son exécution automatique dans le fichier /etc/rc.local à la mise en service de la Rasp. (Il est préférable de mettre 20 secondes

pour être sûr.) Attention d'avoir démarré la Raspberry au moins une fois avant d'y installer vos fichiers personnels. Lors de la première mise en service, la Raspberry lance une procédure d'installation qui pourrait effacer une partie de votre travail.

Remarques : Si vous ne recevez pas de mail dans votre gmail, il est possible que google bloque les applications non sécurisées. Soit, vous recevez un mail et vous suivez le lien, soit, vous pouvez suivre ce lien : <https://support.google.com/a/answer/6260879?hl=fr> pour activer les applications non sécurisées.

Les scripts :

```
# By default this script does nothing.

sleep 15s
sudo python /home/pi/programme_python/Envoi_IP_Mail.py

# Print the IP address
_IP=$(hostname -I) || true
if [ "$_IP" ]; then
    printf "My IP address is %s\n" "$_IP"
fi

exit 0
```

Script du programme « Envoi_IP_Mail.py » page suivante !

Quant au script python : **(utilisation de Geany)**

Un petit mot sur les interfaces graphiques : elles fonctionnent si et seulement si le client SSH (putty sur windows ou SSH sur Linux) sont configurées pour rapatrier les éléments graphiques.

C'est-à-dire, avec putty (Windows) :

- lancer Xming sur le PC,
- cocher la case dans l'onglet X11 des paramètres de putty,

Avec SSH (Linux) :

- "ssh -X pi@<adresse ip de la RPI>".

Lancez Geany sur RPI :

"geany". (Ne pas mettre sudo !)

Message à destination des "root addicts". Un fichier créé par "root" reste la propriété de "root".


```

1  #!/usr/bin/env python
2  # coding: utf-8 -*-
3
4  import socket
5  import fcntl
6  import struct
7  import smtplib,os,sys
8  from email.MIMEText import MIMEText
9  from email.MIMEMultipart import MIMEMultipart
10
11  from_adrs = "<l'adresse de l'envoyeur (proprio du mail de départ)>"
12  to_adrs = "<l'adresse de celui qui doit recevoir l'info>"
13  login_mail = "<login du mail>"
14
15  def mon_ip(iframe):
16      s = socket.socket(socket.AF_INET, socket.SOCK_DGRAM)
17      return socket.inet_ntoa(fcntl.ioctl(
18          s.fileno(),
19          0x8915,
20          struct.pack('256s', iframe[:15])
21      )[20:24])
22
23  osts = ""
24  try:
25      sts1 = "interface wlan: " + mon_ip('wlan0')
26      # print sts
27  except:
28      sts1 = "rien sur wlan0"
29      # print sts
30
31  osts = sts1 + "\n"
32
33  try:
34      sts2 = "interface eth0: " + mon_ip('eth0')
35      # print sts2
36  except:
37      sts2 = "rien sur eth0"
38      # print sts2
39
40  osts = osts + sts2
41
42  msg = MIMEMultipart()
43  msg['From'] = from_adrs
44  msg['To'] = to_adrs
45  msg['Subject'] = 'adresse ip de la raspberry'
46  msg.attach(MIMEText(osts))
47  mailserver = smtplib.SMTP('smtp.skynet.be', 587)
48  mailserver.ehlo()
49  mailserver.starttls()
50  mailserver.ehlo()
51  mailserver.login(from_adrs, "<mot de passe du mail>")
52  mailserver.sendmail(from_adrs, to_adrs, msg.as_string())
53  mailserver.quit()
54

```

Dépassement : Adaptez le programme pour qu'il vous donne votre adresse IP externe.

Expansion du Filesystem :

L'expansion du système de fichiers n'est pas obligatoire car lorsque nous voulons faire une image de notre OS raspbian pour réaliser une sauvegarde de la RPI, cela pourrait poser des problèmes. (Le 16Go d'une carte μ SD d'un constructeur n'est pas forcément le même 16Go d'un autre constructeur.)

Avec l'utilitaire « raspi-config », procéder à l'extension du système de fichiers. Ceci permettra d'utiliser toute la capacité de la carte (μ)SD. Attention, pour ce faire, il faut être connecté en SSH sur la RPI et donc connaître l'adresse IP de la RPI !

Sauvegarde de votre travail :

Pour sauvegarde votre travail, vous pouvez procéder de deux façons :

- Sauvegarde de la partition (on procède comme pour l'installation, mais en sens inverse)
- Sauvegarder seulement les fichiers que vous avez modifié. Ceci vous impose une certaine rigueur dans l'utilisation des fichiers et dossiers dans lesquels vous faites vos actions

La sauvegarde de la partition est plus complète et permet la récupération directe sur une autre carte μ SD. Elle est aussi plus lente. À ce titre, il peut être intéressant de limiter la taille de la partition que l'on veut sauvegarder.

La sauvegarde des fichiers ciblés est beaucoup plus rapide et vous permet une installation rapide sur la base d'une distribution (Raspbian) nouvellement installée. Elle vous permet aussi de récupérer un historique des différentes versions de votre travail.

Manip 2 : Arduino sur RPI

Connectez-vous à la RPI en SSH et lancez les commandes de mise à jour :

```
"sudo apt-get update"
```

```
"sudo apt-get upgrade".
```

Installez l'IDE de l'Arduino :

Allez sur Moodle (ou sur le site Arduino) télécharger le logiciel Arduino (arduino-1.8.X-linuxarm.tar.xz). (dernière version 32 bits)

Envoyez le fichier compressé sur la RPI (via SCP (Linux) ou WinSCP (Windows)).

Tapez les commandes :

```
tar -xvf arduino-1.8.X-linuxarm.tar.xz
```

```
rm arduino-1.8.X-linuxarm.tar.xz
```

```
cd arduino-1.8.X/
```

```
./arduino-linux-setup.sh $USER
```

```
sudo ./install.sh
```

```
ln -s /home/pi/<.....>/arduino-1.8.X/arduino /home/pi/bin/arduino
```

```
sudo chown pi arduino-1.8.7
```

```
sudo chgrp pi arduino-1.8.7
```

Lancez Arduino sur RPI :

```
"arduino". (Ne pas mettre sudo !)
```

Installez la librairie "Tempo" dans le dossier ad-ok de la RPI. Les fichiers de la librairie se trouvent sur Moodle. Utilisez le programme "Win-SCP" pour y arriver (si vous êtes sous Windows). Si vous êtes sous linux, utilisez la commande "scp".

Pour le bon fonctionnement de l'IDE de l'Arduino, il est important de respecter la structure des répertoires (surtout pour les librairies). Étudiez bien la structure des répertoires pour la comprendre et bien l'utiliser.

Chargez le programme "blink2" (qui se trouve au même endroit que la librairie) et testez le programme.

Calculez les résistances nécessaires pour alimenter deux LEDs qui seront respectivement connectées aux sorties 10 et 11 de l'Arduino. Adaptez le programme blink2 pour faire clignoter les deux LEDs à des fréquences voisines (pas des harmoniques).

Pense-bête :

Commandes utiles :

- "sudo" : pour exécuter une commande avec des droits d'administrateur.
- "nano" : pour éditer un fichier en mode texte. (Permet également de créer un fichier)
- "geany" : pour éditer un fichier en mode graphique. (Permet également de créer un fichier voire un dossier et de l'exécuter)
- "rm" : supprime un fichier
- "rmdir" : supprime un dossier vide
- "rm -r" : supprime un dossier non vide
- "mkdir" : créé un dossier
- "cd" : voyage dans les dossiers
- "ls" : liste les fichiers et dossiers
- "reboot" : redémarre la RPI
- "shutdown" : éteint la RPI
- "python" ou "python3" : Lance le programme python (avec la version 2 ou 3)
- "curl www.icanhazip.com" pour connaître son adresse IP externe (IPv4)
- "scp" : transfert de fichier vers/de un ordinateur ayant un serveur SSH actif (linux)
- "winscp" : ... devinez?
- "touch" : créé un fichier vide ou met à jour la date de modification du fichier
- "iwlist wlan0 scan" : liste les réseaux Wifi détecté par la RPI
- "wpa_passphrase" : permet de crypter le mot de passe du Wifi
- "soffice --calc VotreFichierALire" : permet de lire un fichier CSV avec calc