www.elektor.fr/rpi

Raspberry Pi

l'alliance de la programmation et de l'électronique

45 applications utiles pour l'électronicien

Auteur : Maartje JANSEE

Éditeur : Elektor
ISBN : 978-2-86661-196-5
Format : 17 × 23,5 cm
Nbre de pages : 288
Prix : 37,50 €

Ce fichier contient le sommaire suivi de six pages extraites de l'ensemble du livre qui en compte 288.

Sommaire

Introduction 7

1. Le nécessaire 9

1.1 Raspberry Pi 10
1.2 Carte mémoire SD 10
1.3 Clavier 10
1.4 Souris 10
1.5 Écran 11
Option 1 : HDMI 11
Option 2 : téléviseur analogique 11
Option 3 : convertisseur HDMI-VGA 12

Sommaire - Copyright © 2014 - Elektor International Media

1.6 Connexion à l'internet 13
Option 1 : par un câble 13
Option 2 : sans fil (WiFi) 13
1.7 Alimentation 14
1.8 Hub USB 14
1.9 Haut-parleurs (ou écouteurs) 14
1.10 Platine d'essais 14
1.11 Composants 16
1.12 Oscilloscope logiciel 16

2. Petit tour du Raspberry Pi 21

2.1 LXterminal 22
2.2 LXterminal – Scrot 23
2.3 LXterminal – Nano 24
2.4 IdleX 25
2.5 Debian Reference 26
2.6 Midori 26
2.7 WiFi config 26
2.8 Scratch 26
2.9 Accessories – Leafpad 27
2.10 Accessories – File Manager 27
2.11 System Tools – Task Manager 28

3. Installation des logiciels 29

3.1 Raspberry Pi 29
3.1.1 Clavier 29
3.1.2 Cartes SD de plus forte capacité 30
3.1.3 Autres distributions de la mémoire (memory split) 31
3.1.4 Adresse IP 31
3.1.4.1 Adresse IP fixe avec bail statique 32
3.1.4.2 Adresse IP fixe avec IP statique 34
Liaison câblée 35
Liaison sans fil WiFi 36
3.2 PC de table (facultatif) 37
3.2.1 Disk Imager 37
3.2.2 Python, wxPython et IdleX 38
3.2.3 Putty (SSH) 39
3.2.4 Xming (XWindow) 41
3.2.5 WinSCP 46

4. Petite introduction à Linux, Python et JavaScript 49

4.1 Linux variété Debian 49
Terminal 49
Utilisateur root 51
Permissions 52
Méthode de travail 55
Commandes Linux 55
Installation d'applications 55

4.2 Programmer en bash 57

4.3 Programmation en Python 62
4.3.1 Opérateurs 72
4.3.2 Abréviations des opérateurs mathématiques 74
4.3.3 For 74
4.3.4 While 75
4.3.5 Pass 76
4.3.6 If-then 76
4.3.7 Binaire 77
4.3.8 Utilisation d'extensions 78
4.3.9 Fichiers 79
4.3.10 Try/except 81
4.3.11 Fonctions 82
4.3.12 Graphique 85
Button 88
Static Text 88
Spinbox 89
Slider 89
Paintbox 89
4.3.13 Threading 89
4.3.14 Conclusion 90

4.4 Programmer en JavaScript 90

5. GPIO 95

5.1 Introduction 95

5.2 LED 98

5.3 LED clignotante 101

5.4 Clignotants alternés 104

5.5 LED temporisée avec une fenêtre 106

5.6 Interrupteur 110

5.7 Minuterie 115

5.8 Interrupteur bistable 116

5.9 État de l'interrupteur dans une fenêtre 117
5.9.1 Avec un temporisateur 118
5.9.2 Avec plusieurs fils (multi-threading) 121

Sommaire - Copyright © 2014 - Elektor International Media

5.10 Un bouton et du son 126
5.11 Communication par les broches d'E/S 129

6. Plus de puissance 133

6.1 TD62783 (pilote source à 8 canaux) 133
6.1.1 Ampoule de 6 V/65 mA 135
6.1.2 Moteur 5 V/145 mA 136
6.2 ULN2003 (réseau de 7 darlingtons à collecteur ouvert) 137
6.2.1 Ventilateur 12 V/125 mA et ampoule 6 V/65 mA 139
6.3 MOSFET IRF740 141
6.3.1 Moteur 5 V/550 mA 141

7. Modulation de largeur d'impulsion 147

7.1 Modulation de largeur d'impulsion selon trois modes 148
7.1.1 Pilotage en fréquence 149
7.1.2 Impulsion-pause à fréquence constante 151
7.1.3 Impulsion-pause à fréquence variable 154
7.2 Allumage progressif de l'ampoule 155
7.3 Moteur avec réglage de vitesse 159

8. SPI 161

8.1 Présentation du bus SPI 161
8.2 MAX522 (convertisseur numérique-analogique, CN/A, DAC) 164
8.3 CN/A avec amplificateur opérationnel 170
8.4 Plus de deux périphériques SPI 175
8.5 MCP3008 (convertisseur analogique-numérique, CA/N) 177
8.6 MCP3008 (mesure pseudo-différentielle) 183

9. I2C 185

9.1 Présentation du bus I2C 185
9.2 MCP23008 interface d'entrées-sorties parallèles 187
9.3 Davantage de puissance pour l'interface 193
9.4 Thermomètre numérique TC74 193

10. Liaison sérielle 203

10.1 Présentation de la norme RS232 203
10.1.1 Boucle sérielle 204
10.2 Liaison sérielle RPi – Piccolino 205

10.2.1 Écho sériel 208
10.2.2 Mesure analogique sérielle (CA/N) 210

11. Serveur ouèbe 217

11.1 Introduction 217
11.2 Serveur HTML 217
11.3 CGI 220
11.3.1 Serveur CGI 221
11.3.2 Compter les visites dans un fichier 225
11.3.3 Transfert de variables du navigateur vers le serveur 227
11.3.4 GPIO 230
11.3.5 GPIO avec JavaScript 234
11.3.6 Alarme de réfrigérateur, sonde I2C et page ouèbe automatique 242

12. Client-serveur 247

12.1 Introduction 247
12.2 Multiplication TCP 247
Serveur 248
Client 251
12.3 Commande de LED par TCP 253
12.4 Conversion numérique-analogique par TCP 254
12.5 Générateur de dents de scie et d'ondes carrées par TCP 256
12.6 Voltmètre TCP à 8 canaux 264
12.7 Écho UDP 270
12.8 Luxmètre UDP 273

13. Appendices 277

13.1 Alimentation variable 277
13.2 Brochage de la barrette de connexion 280
13.3 Manifeste du téléchargement 281
RPi 281
PC Windows 281
13.4 Liste des composants 282

Index 283

Sommaire - Copyright © 2014 - Elektor International Media

1. Dans un *LXterminal*, tapez : `nano hello.c`
   pour démarrer l'éditeur et créer le fichier.
2. Saisissez le programme :
```
#include <stdio.h>
int main() {
printf("Mondo cane!");
return 0;
}
```
3. Enregistrez le programme avec Ctrl-O et fermez `nano` avec Ctrl-X.
4. Compilez le programme avec : `gcc -o hello hello.c`
   Apparemment, rien ne se passe jusqu'au retour de l'invite.
5. Tapez `./hello` pour exécuter le nouveau programme.
   Le texte « Mondo cane ! » apparaît à l'écran.

Les signes « ./ » avant la commande indiquent au système que le programme à exécuter se trouve dans le répertoire actuel et non dans les emplacements canoniques contenus dans le chemin (*path*) des exécutables.

Vous pouvez fermer le terminal par la commande `exit`. Nous ne programmons pas en C dans ce livre, mais en *Python,* toutefois vous voyez qu'il est possible d'utiliser le RPi en C.

2.4 *IdleX*

IdleX regroupe l'éditeur et l'environnement de programmation de *Python*. L'environnement de développement standard pour *Python* est *Idle*. Nous utilisons *IdleX* qui incorpore une vingtaine d'extensions et greffons relatifs à l'édition, la navigation dans le code, l'exécution du code, l'interaction avec le *shell* (le logiciel de bas niveau sous-jacent à toute application). Par exemple, si vous ne vous êtes pas habitué à la séquence Ctrl-C Ctrl-V pour copier et coller un bloc de texte en deux frappes au clavier, vous pourrez faire : clic droit pour dérouler le sous-menu, déplacement du curseur sur *Copy,* clic gauche, déplacement du curseur à la destination, clic droit pour le menu, déplacement du curseur sur *Paste,* clic gauche. Quand vous lancez *Idle*, c'est en fait *IdleX* qui s'exécute. Nous y reviendrons dans la section 4.3, page 62.

Raspberry Pi - l'alliance de la programmation et de l'électronique
45 applications utiles pour l'électronicien - ISBN 978-2-86661-196-5
Copyright © 2014 - Elektor International Media

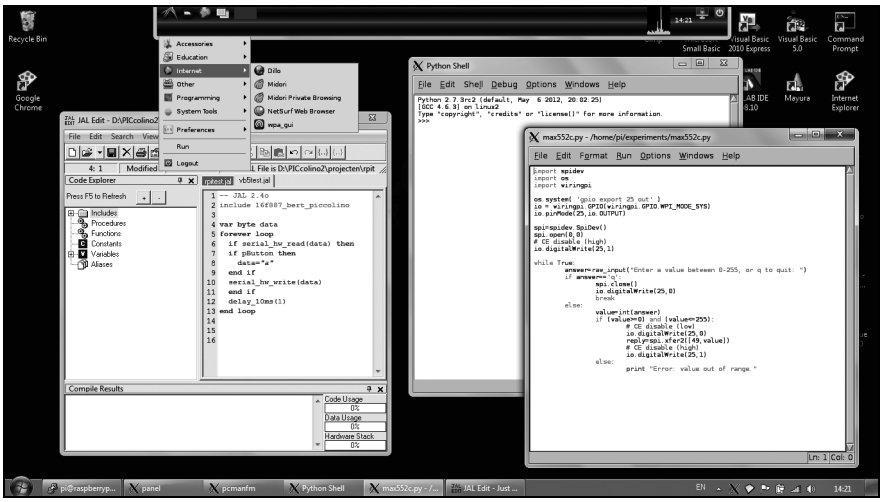

Figure 22 - Programmes Windows et Linux sous *XWindow* actifs simultanément.

Messages d'erreur possibles :

- pas de message, mais rien ne se passe

Vous essayez de démarrer *lxde* sans *Xming*. Démarrez d'abord *Xming* et *lxde* ensuite.

- `Unable to access the X Display, is $DISPLAY set properly?`

Vous avez oublié de démarrer *Xming*, faites-le maintenant et ré-essayez.

- `PuTTY X11 proxy: MIT-MAGIC-COOKIE-1 data did not match`

`Unable to access the X Display, is $DISPLAY set properly?`

Vous essayez de démarrer un programme graphique avec `sudo`. C'est possible, mais vous devez d'abord déclarer le changement d'utilisateur. Cela se fait par deux commandes successives (chacune suivie par *Enter*) :

```
cd
sudo bash sudoset.sh
```

Le changement est valide jusqu'à l'arrêt du RPi. Vous devrez répéter les commandes s'il a été arrêté.

- `Only 1 xsession can be established at a time`

Vous n'avez pas déclaré dans *Putty* la délégation du serveur X11. Retournez au texte sous la figure 19, et n'oubliez pas de sauvegarder la configuration quand vous aurez terminé les modifications.

- `Obt-Message: Xinerama extension is not present on the server`

`Openbox-Message: A window manager is already running on screen 0`

Copyright © 2014 - Elektor International Media

Le nombre binaire 101 vaut $1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0 = 5$ en décimal. Le problème est que vous ne pouvez pas voir que c'est un nombre binaire. Cela se prête fort bien à des plaisanteries. Par exemple : « il y a 10 sortes de gens dans ce monde, ceux qui comptent en binaire et les autres ».

Pour éviter cette confusion, nous utilisons un préfixe, qui sera 0b pour binaire. Remarquez que, contrairement à de nombreux autres langages de programmation, *Python* ne reconnaît pas la notation binaire des nombres.

Il est très pratique de pouvoir convertir de tête le binaire en décimal, mais si le calcul mental n'est pas votre fort, vous pouvez utiliser la calculette en accessoire sur le bureau Windows ou Linux. Vous la configurerez en « programmeur » ou « scientifique » pour pouvoir effectuer ces conversions.

4.3.8 Utilisation d'extensions

En dehors des instructions *Python*, vous pouvez charger des modules d'extension avec des instructions supplémentaires. Cela s'effectue avec la commande *import*. Par exemple, *Python* ne connaît pas d'instruction pour pause. Le module `time` en a une. Cet exemple de programme imprime un « a » toutes les 0,3 seconde :

```
import time
while True:
   print "a"
   time.sleep(0.3)
```

L'apposition de `time.` avant `sleep` signifie que cette instruction se trouve dans le module `time`. Vous pouvez renommer un module lors du chargement si vous trouvez cela plus commode :

```
import time as t
while True:
   print "a"
   t.sleep(0.3)
```

Si vous n'avez besoin que d'une instruction du module, vous pouvez la charger sélectivement de façon à ne plus rien avoir à préciser :

```
from time import sleep
while True:
   print "a"
   sleep(0.3)
```

Raspberry Pi - l'alliance de la programmation et de l'électronique
45 applications utiles pour l'électronicien - ISBN 978-2-86661-196-5
Copyright © 2014 - Elektor International Media

```
if __name__ == '__main__':
   main()
```

Lancez le programme avec `python timerled.pyw`. Si vous utilisez ce programme via une liaison SSH à partir d'un PC Windows, vous devez d'abord lancer un serveur X (*Xming* dans notre cas) avant de lancer une application graphique[1].

Quand vous cliquez sur le bouton, la LED s'allume et le compteur avance. Après 2 s, la LED s'éteint et le texte « Appuyez sur le bouton pour démarrer » s'affiche à nouveau. Vous pouvez arrêter le programme en fermant la fenêtre, d'un clic sur la croix en haut à droite.

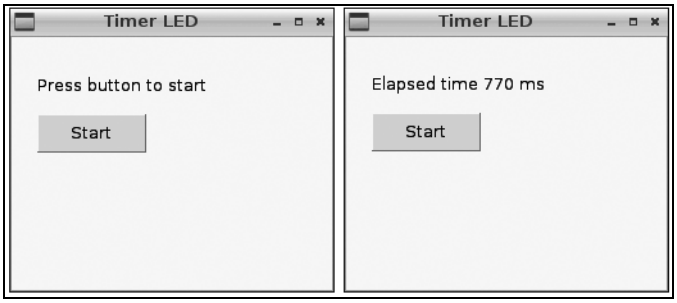

Figure 44 - Le programme en action.

5.6 Interrupteur

Une broche utilisée comme entrée ne doit jamais être laissée « en l'air ». C'est-à-dire que son potentiel doit toujours être fixé, que ce soit au plus ou à la masse. Normalement, nous devons raccorder un interrupteur suivant le schéma de la figure 45.

Quand l'interrupteur est ouvert, le RPi est relié à la masse par une résistance de 10 kΩ, ce qui porte la broche au niveau bas. Quand l'interrupteur est fermé, la broche est reliée directement au +3,3 V, donc au niveau haut. Naturellement, un courant traverse la résistance de 10 kΩ, mais son intensité est négligeable :

$$I = \frac{V}{R} = \frac{3{,}3}{10\,000} = 0{,}33 \text{ mA}$$

1. Reportez-vous à la section 3.2, page 37, pour plus d'informations sur SSH et *Xming*.

Raspberry Pi - l'alliance de la programmation et de l'électronique
45 applications utiles pour l'électronicien - ISBN 978-2-86661-196-5
Copyright © 2014 - Elektor International Media

avec : I = intensité en ampères
R = résistance en ohms
V = tension en volts
P = puissance en watts

Le remplacement de P et R donne $I = \sqrt{0{,}727/5} = 381$ mA.
Trop peu pour notre moteur, d'où la nécessité du MOSFET[1].

Le circuit est représenté sur la figure 64. **Attention** : les masses du RPi, du MAX4420 et de l'IRF420 sont reliées ensemble et le MAX4420 est alimenté par une source de tension externe. Si le moteur perturbe la ligne d'alimentation, vous pouvez le cas échéant raccorder un petit condensateur de 0,1 µF aux bornes du moteur (veillez à utiliser un type qui supporte la tension appliqué).

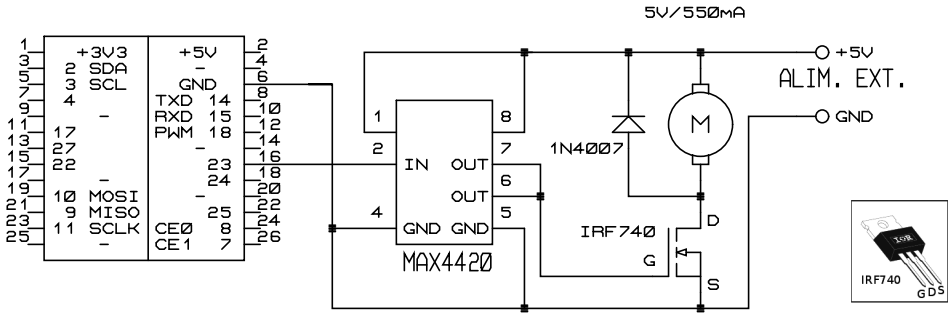

Figure 64 - Raccordement d'un moteur et d'un MOSFET à la barrette.

Vous pouvez utiliser le programme suivant :

```
import os
import wiringpi
from time import sleep

os.system( 'gpio export 23 out' )
os.system( 'gpio export 24 out' )|
sleep(0.5)
```

1. Un calcul similaire à partir de la feuille de caractéristiques de l'IRF740 donne : puissance maximale 125 W, résistance interne maximale (RDSon) 0,550 Ω.

   L'intensité maximale permanente sera $\sqrt{125/0{,}55} = 15$ A. Largement en dessous de la limite de 10 A du MOSFET, donc il n'y aura pas de problème de puissance, pourvu qu'on ait un radiateur suffisant pour évacuer la chaleur.

Raspberry Pi - l'alliance de la programmation et de l'électronique
45 applications utiles pour l'électronicien - ISBN 978-2-86661-196-5
Copyright © 2014 - Elektor International Media

Le canal choisi, l'utilisateur peut cliquer sur un bouton pour effectuer une mesure.

```
# bouton pour lancer la mesure
btn = wx.Button(self, label='Mesure', pos=(20, 120), size=(80, -1))
btn.Bind(wx.EVT_BUTTON, self.OnSelect)
```

Et voici l'objet qui reçoit le résultat de la mesure :

```
# texte statique
self.txt = wx.StaticText(self, label=' ', pos=(20, 100))

# placez ici vos propres fonctions
```

La fonction contient les formules que nous avons vues dans le programme précédent, auxquelles s'ajoute le pourcentage de l'échelle totale, calculé en même temps.

```
def  OnSelect(self, e):
     chan=self.spn.GetValue()
     reqadc = spi.xfer2([1,128+(chan<<4),0])
     value = (reqadc[1]&3)*256 + reqadc[2]
     pervalue = int(value*100/1024.0)
     self.txt.SetLabel('CA/N = '+str(value)+' unités,
                       ou '+str(pervalue)+' %')
```

Lancez le programme avec `sudo python mcp3008.pyw`[1], choisissez un canal (seuls le 0 et le 1 ont un potentiomètre) et cliquez sur Mesure pour effectuer une mesure.

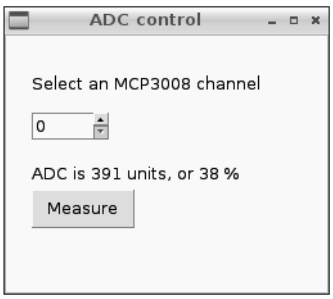

Figure 97 - Choix graphique du canal.

1. Si vous travaillez à distance via SSH, n'oubliez pas de lancer `sudo bash sudoset.sh` après chaque connexion pour autoriser `sudo` via *Xming* et *Putty*. Vous aurez installé *Xming* auparavant en suivant les instructions de la section 3.2.4, page 41.

Raspberry Pi - l'alliance de la programmation et de l'électronique
45 applications utiles pour l'électronicien - ISBN 978-2-86661-196-5
Copyright © 2014 - Elektor International Media

!

. . . . . 73, 74
– 74
!= 73
* 74
** 74
/ 74
/boot/config.txt 13
/etc/network/interfaces 35
/usr/bin/env 224
\ 64
& 180, 197
% 74
%c 68
%d 68
%s 68
+ 74
+3,3 V 97
+5 V 97
<> 73
= 74
> 73
>= 73
>> 74

Numérique

1N4007 137

A

addition 74
adduser 54
adresse IP 31
affectation 74
alsamixer 129
amixer 60, 128
ampli op 171
append 70, 200
apt-get 55
argv 127
ASCII 210

B

bail statique 32
ballast 98
barre de tâches 22
barregraphe 267
bash 57
baudrate 203
binaire 77
bind 248
bit 197
bit de signe 196
bloqué 136
bobine 136
boucle 204
    for 60
    while 61
bouton 91, 237
Button 88, 107

C

C, programmer en 24
CA/N 177, 210
carré 256
carte SD 22, 30
    plus grande 30

Raspberry Pi - l'alliance de la programmation et de l'électronique
45 applications utiles pour l'électronicien - ISBN 978-2-86661-196-5
Copyright © 2014 - Elektor International Media

case à cocher 237
cat 56
cd 56
CE 161, 175
CGI 222
cgi-bin 225
CGIHTTPServer 218
checkbox 92
chmod 53, 223
class 86
clavier 29
client-serveur 247
close 80, 164
CN/A 164, 170
COMMON 203
complément à 2 196
config.txt 13
connecteur 97
const byte 208
cp 56
CPHA 163
CPOL 163
Ctrl-C 81
curseur 19, 172

D

DAC 164, 170
darlington 137
dd 56
Debian 49
    Reference 26
débit 203
décalage
    d'un bit à droite 74
    d'un bit à gauche 74
def 82
defval 187
démon 258
dent de scie 256
df -h 56
DHCP 35
dialogue RS232 211
différent de 73
digitalRead 103
digitalWrite 102, 107
DIN 165
diode de roue libre 134, 135, 139
dir 79
Disk Imager 37
diviseurs de tension 206
division 74
drain 141
DrawBitmap 214
droits 50, 51
du -sh 56

E

echo 57
Edimax 12
élévation à la puissance 74
entier 66
entrée inverseuse 171
equals 59
erreur 404 219
ESTABLISHED 250
EVT_PAINT 213
exception 259

Raspberry Pi - l'alliance de la programmation et de l'électronique
45 applications utiles pour l'électronicien - ISBN 978-2-86661-196-5
Copyright © 2014 - Elektor International Media

F

false 104
favicon 219
File Manager 27, 54
fin de ligne 225
floats 66
flottant 66
fonction 82, 93
for 60
FORM NAME 92

G

gcc 25
global 84
gpinten 187
GPIO 95, 100
gpio 187
gppu 187
graisse 198
grille 141

H

HDMI 11
HDMI-VGA 12
HEAD 91
HTML 91

I

I2C 185
i2cdetect 189
IdleX 25, 39, 62
if-then 76
import 78
inconsistent indentation 75
index.html 217
inférieur à 73
inférieur ou égal à 73
inft 187
input 65
install 55
intcap 187
intcon 187
intensité 97
interface graphique 42
interrupteur 110
interruption-clavier 81
iocon 187
iodir 187
IP statique 34
ipol 187
IRF740 141

J

JAL 210
JavaScript 90, 234

K

KeyboardInterrupt 81

L

LDR 269
Leafpad 57
LED 98, 99

Raspberry Pi - l'alliance de la programmation et de l'électronique
45 applications utiles pour l'électronicien - ISBN 978-2-86661-196-5
Copyright © 2014 - Elektor International Media

len 70
ligature mérovingienne 229
Ligawo 12
ligne de commande 127
Linux 49
liste 69, 199
listen 248
LM358 171
loi d'Ohm 98
ls 53, 56
luxmètre 273
LXDE 21, 43
LXterminal 22, 49

M

man 56
mark-space 151
MAX4420 142
MAX522 . . . . . 165, 255
MCP23008 187
MCP3008 . . . . . 177, 264
Minicom 204, 209
MISO 162
mkdir 56
MLI 147
mode 100, 163, 167
modulation de largeur d'impulsion . 147
modulo 74
Monty Python 126
MOSFET 142
MOSI 162
mot de passe 40
moyenne glissante 200
mp3 127
mpg321 60
multiplication 74
mv 56

N

Nano 24, 57
netstat 250
niveau haut au repos 209
No such file 223
not 105
numéros de broches 97

O

olat 187
only 1 xsession 45
open 79, 164
oscilloscope 16
out 100

P

Paintbox 89
papier quadrillé 212
paramètre 127
parité 211
pass 76, 116
permissions 27, 52
Piccolino 210
pinMode 102, 148
police 197
pop 70, 200
port
    5000 248
    8080 218

Raspberry Pi - l'alliance de la programmation et de l'électronique
45 applications utiles pour l'électronicien - ISBN 978-2-86661-196-5
Copyright © 2014 - Elektor International Media

print 64
puissance, calcul de 142
Putty 39
PuTTY X11 proxy 45
PWM 147
pwmc 148, 153
pwm-ms 148, 151
pwmWrite 148
pyserial 211

R

R, résistance 98
rail to rail 167
range 74
raspi-config 29
raw_input 76
read 113
read_byte 186
read_byte_data 186
readline 79, 211
readlines 80
rebonds 116
reboot 35
réel 66
remove 55
replace 71
return 83
rm 56
rmdir 56
root 49, 50
RS232 203
RS232-TTL 205
rwx 54
RX 203

S

sauvegarde 37
SCLK 161
Scratch 26
SCRIPT 92
Scrot 23
search 55
Segmentation fault 151
self 88
serial_sw_byte() 208
serial_sw_print() 208
serial_sw_read() 208
serial_sw_write() 208
serveur
  CGI 221
  GPIO 230
  HTML 217
  TCP 248
  UDP 271
SetSizeHints 212
seudo-différentiel 183
shell 65, 66
sleep 78
slice 71
Slider 89
smbus 186
socket 248
son 13, 14
source 141
soustraction 74
SPI 161
Spinbox 89, 181
Squeeze 49
SSH 39
StaticText 88, 107

Raspberry Pi - l'alliance de la programmation et de l'électronique
45 applications utiles pour l'électronicien - ISBN 978-2-86661-196-5
Copyright © 2014 - Elektor International Media

Stontronics 14
string 68
style 198
submit 91, 236
sudo 50
sudoset.sh 45
sum 70
supérieur à 73
supérieur ou égal à 73
sys 127
system 103

T

tampon 171
Task Manager 28
TC74 . . . . . 193, 242
TCP 247
TD62783 . . . . . 133, 155
temporisateur 118
tension 98
terminal 49
thread 121, 256
Threading 89
time out 211
TIME_WAIT 250
timer 107
true 104
try/except 81
ttyAMA0 . . . . . 204, 209
TX 203

U

ULN2003 . . . . . 137, 139
unable to access 45
unable to open 151
update 55

V

valueError 228
ventilateur 139
voltmètre 255
VREF 168

W

while 61, 76
WiFi 10, 13, 26
WinOscillo 17
WinSCP 46, 58
wiringpi 102
write 80, 100
write_byte 186
write_byte_data 186
writebytes 164
writelines 80
wx.ClientDC 213

X

xfer 164
xfer2 164
Xinerama 43
Xming 41
XWindow 41

Raspberry Pi - l'alliance de la programmation et de l'électronique
45 applications utiles pour l'électronicien - ISBN 978-2-86661-196-5
Copyright © 2014 - Elektor International Media