



Raspberry Pi (4) - Interface HTTP -

Contrôler le RPi avec une interface HTTP

Nom :	Appréciation :	Note :
Prénom :		
Classe :		
Date :		

<p>Objectif : Contrôler les GPIO du RPi depuis n'importe quel appareil possédant un navigateur internet</p> <p>Utilité : L'interface HTTP est une interface agréable et ergonomique qui permettra à un simple utilisateur de contrôler des équipements électriques reliés au RPi</p>	durée : 4h
----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	------------

Matériel : alimentation de laboratoire – multimètre – plaque labdec – composants électroniques

Prérequis : Connexion à distance avec SSH, commande GPIO

Compétences et savoirs principalement visées :
C2-1, C2-2 (page 3a), C3-2, C3-3 (page 3b à 6)

Travail à réaliser :

- installation d'application PHP
- utilisation d'application PHP
- modification d'application PHP

Schéma du système :

Pi B+ GPIO Ref

3.3V	1	5V
	2	5V
	3	GND
	4	14
GND	5	15
	17	18
	27	GND
	22	23
3.3V	24	
	10	GND
	3	25
	11	6
GND	7	
IDSD	1	IDSC
	5	12
	13	GND
	19	16
	26	20
GND	21	

$R1=R3=47k\Omega$
 $R2=R4=470\Omega$

COMMANDE A DISTANCE AVEC LE SERVEUR HTTP

Serveur Web APACHE

Un service Web est un service auquel on accède à distance à l'aide d'un « **navigateur** » ; généralement ce service affiche des « pages Web » ; le RPi contient un serveur Web que nous allons utiliser, pas pour afficher des pages Web, mais pour contrôler les GPIO de manière plus conviviale qu'avec Putty !!

Grâce à cette technique, tu pourras contrôler les GPIO depuis n'importe quel appareil disposant d'une connexion Internet ! Voici le principe de fonctionnement :



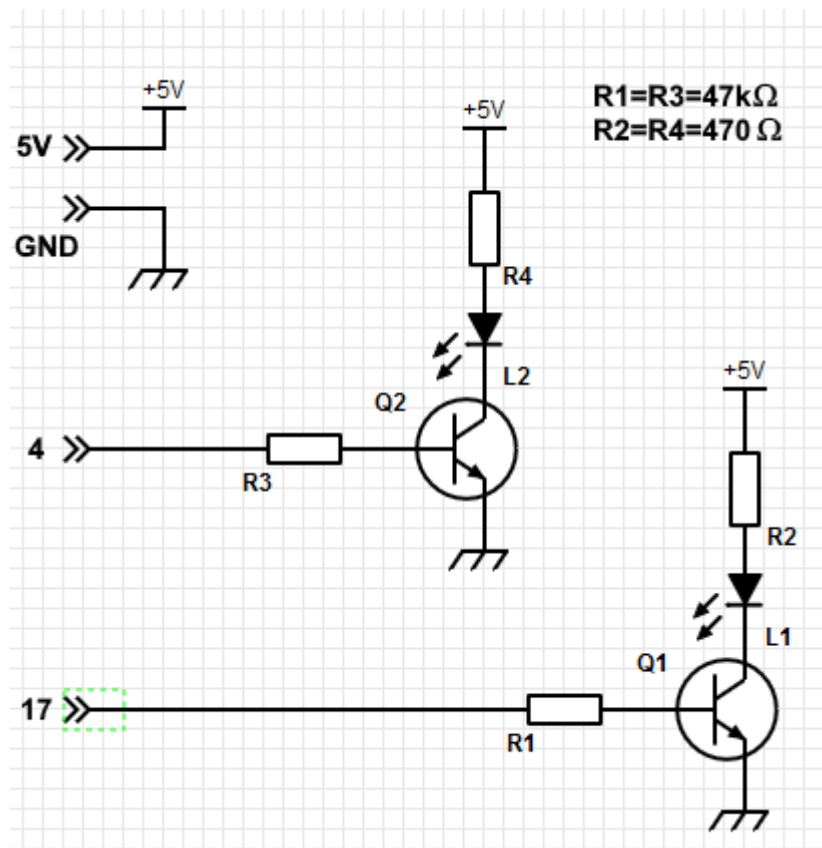
Cites les noms des navigateurs internet que tu connais : _____

- ◆ Réalise le **montage suivant sur la plaque Labdec**

AVERTISSEMENT

Attention : en cas d'erreur de branchement, ton Raspberry Pi risque d'être **détruit** !!! Ne mets pas le circuit sous tension **avant** que le professeur l'ai vérifié.

Pi B+ GPIO Ref	
3.3V	5V
2	5V
3	GND
4	14
GND	15
17	18
27	GND
22	23
3.3V	24
10	GND
9	25
11	8
GND	7
IDSD	IDSC
5	GND
6	12
13	GND
19	16
26	20
GND	21



- ◆ Connectes-toi en SSH au RPi avec **Putty** (utilisateur : **root**)
- ◆ Lance les commandes « **apt-get update** » et « **apt-get install php5** »
- ◆ Vérifie que le montage fonctionne en effectuant la série de commande :

```
gpio -g mode 4 out
gpio -g mode 17 out
gpio -g write 4 1
gpio -g write 4 0
gpio -g write 17 1
gpio -g write 17 0
```

→ A quel résultat t'attendais-tu en faisant ces commandes ?

→ Le résultat est-il conforme ?

- ◆ Télécharger et installer l'application PHP :

```
cd /var/www/html
rm index.html
wget http://cvardon.fr/annexes/rpi-4-page8.tar.gz
tar xzf rpi-4-page8.tar.gz
mv rpi-4-page8.php index.php
```

→ Compte tenu du nom du fichier contenant l'application (*index.php*), quelle adresse faudra-t-il taper dans le navigateur internet pour y accéder ? (compléter le cadre ci-dessous)

http://10.3.14. __ / _____

→ Est-il possible d'utiliser une URL (adresse) simplifiée ? Pourquoi ?

- ◆ Lance l'application avec le navigateur internet de ton smartphone et teste-là ; résumes tes observations en remplissant le tableau suivant :

Bouton	Led impactée*	Action(s) réalisée(s)
1	L1 - L2	
2	L1 - L2	
3	L1 - L2	

* Entourer la ou les LED concernées

- ◆ *Le fonctionnement constaté ne convient pas, tu dois être capable de le modifier selon les critères suivants :*

Bouton	Led impactée*	Action(s) réalisée(s)
1	L1 - L2	Allumer / Éteindre
2	L1 - L2	Allumer / Éteindre
3	L1 - L2	Clignoter / Éteindre

- ◆ A partir de WinSCP, **modifie** le fichier « **index.php** » dans le dossier « **/var/www/html** », pour obtenir le fonctionnement désiré. **Avant de répondre, regardes les indices ci-dessous.**

un indice pour t'aider :

- ➔ L2 est connecté au GPIO 4, le GPIO 4 doit donc être commandé par le bouton 2 ;
- ➔ L1 est connecté au GPIO 17, le GPIO 17 doit donc être commandé par le bouton 1 ;
- ➔ Tu vas donc localiser les chiffres 4 et 17 dans les balises concernées ; les balises qui créent les boutons sont **des balises** « **<button ...>** » ; tu vas constater qu'il y a une inversion entre bouton 1 et 2

Information : balises HTML des boutons

Bouton 1

```
<button class=subon style='background-color:$couleur;' type=submit name=cmd value=4-$myval>1</td>;
```

Bouton 2

```
<button class=subon style='background-color:$couleur;' type=submit name=cmd value=17-$myval>2</td>;
```

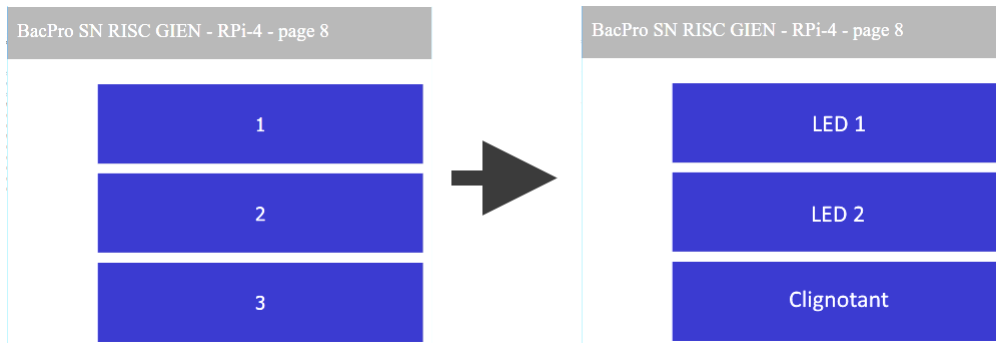
- ◆ Recopie ci-dessous les 2 balises modifiées

- ◆ **Après modification**, teste à nouveau l'application ; résumes tes observations en remplissant le tableau suivant ; cela correspond-il au fonctionnement souhaité ?

Bouton	Led impactée*	Action(s) réalisée(s)
1	L1 - L2	
2	L1 - L2	
3	L1 - L2	

* Entourer la ou les LED concernées

- ◆ Les boutons portent le texte « 1 », « 2 » et « 3 » ; cela n'est satisfaisant ; tu devras modifier ce texte par « LED 1 », « LED 2 », « Clignotant »



un indice pour t'aider : regarde la syntaxe de la balise « **button** » pour localiser le paramètre « **texte** » qui s'affiche sur le bouton ; puis localise ce paramètre pour chacun des 3 boutons dans le fichier « index.php »

Syntaxe de la valise « Button »

Bouton 1

```
<button type=submit name=cmd value=4-$myval>texte</td>;
```

où **texte** est le texte qui va apparaître sur le bouton

- ◆ Recopie ci-dessous les 3 lignes modifiées

- ◆ **Après modification**, teste à nouveau l'application ; l'affichage correspond-il au fonctionnement souhaité ? (Appelle le professeur pour validation)

Annexe 2 : fichier index.php page 8

```
<!doctype html>
<html lang="fr">
<head>
  <meta charset="utf-8">
  <title>BacPro SN RISC </title>
<?php
if (isset($_GET["cmd"])) {$cmd=$_GET["cmd"]} else {$cmd=""};

if ($cmd != "") {echo "<META HTTP-EQUIV='REFRESH' CONTENT='1;URL=index.php'>";}

echo "<link rel='stylesheet' type='text/css' href='pi.css'>";

echo "</head>";
echo "<body>";

$le1="1";$le2="2";$le3="3";
$hw1="direct-4"; $hw2="direct-17"; $hw3="direct-99";

if ($cmd != "" ) {
$bof=explode("-",$cmd);
if ($bof[1]=="on") {exec("/var/www/exo1.sh $bof[0] 1 > /dev/null &");} else {exec("/var/www/$exo1.sh $bof[0] 0 > /dev/null &"); }
}

echo "<div class=titre>&nbsp; PiCommande 3.3</div>";
$haut="height:70px; font-size:2.0em;"; $w="25%";

//Affichage des boutons

echo " <div style='margin-top:-10px;'><form> <table border=0 align=right width=100% cellspacing=20 style='font-size:3.0em;'> <tr >";
if (file_exists("/var/www/etats/$hw1")) {$couleur="#CF0000"; $myval="off";} else {$couleur="#000099"; $myval="on";}

echo "<td align=right><button class=subon style= ' background-color:$couleur;' type=submit name=cmd value=4-$myval>$le1</td>";
echo "<td class=boff></td>";
//echo "<td class=boff><button class=suboff type=submit name=cmd value=1-off>OFF</td>";

echo "</tr><tr >";
if (file_exists("/var/www/etats/$hw2")) {$couleur="#CF0000"; $myval="off";} else {$couleur="#000099"; $myval="on";}

echo "<td align=right><button class=subon style= ' background-color:$couleur;' type=submit name=cmd value=17-$myval>$le2</td>";
//echo "<td class=boff><button class=suboff type=submit name=cmd value=2-off>OFF</td>";
echo "<td></td>";

echo "</tr><tr >";
if (file_exists("/var/www/etats/$hw3")) {$couleur="#CF0000"; $myval="off";} else {$couleur="#000099"; $myval="on";}

echo "<td align=right><button class=subon style= ' background-color:$couleur;' type=submit name=cmd value=99-$myval>$le3</td>";
//echo "<td class=boff><button class=suboff type=submit name=cmd value=3-off>OFF</td>";

echo "</tr></table></div>";

echo "</form>";

?>
</body>
</html>
```

Annexe 3 : fichier exo1.sh page 8

```
#!/bin/bash
# syntax : direct "n° gpio BCM" "0/1"

if (( $2 != 0 && $2 !=1 ))
then exit 1
fi

if [[ "$1" != "4" && "$1" != "17" && "$1" != "99" ]]
then exit 1
fi
/usr/local/bin/gpio -g mode 4 out
/usr/local/bin/gpio -g mode 17 out

#adresser le bon gpio

pin=$1

echo "gpio = $pin"

# memorise la commande !! mais pas de garantie de resultat!
if [[ $2 == "1" ]]
then
touch /var/www/etats/direct-$1
else
rm -f /var/www/etats/direct-$1
fi

if [[ $1 == "99" ]]
then
while [ 1 ]; do
gpio -g write 4 1
gpio -g write 17 1
sleep 0.5
gpio -g write 4 0
gpio -g write 17 0
sleep 0.5
done
else
/usr/local/bin/gpio -g write $pin $2
fi

exit 0
```