



Raspberry Pi - Programmation horaire -

Programmation horaire avec CRON

Nom : Prénom : Classe : Date :	Appréciation :	Note : <div style="text-align: right; font-size: 2em; color: red;">/20</div>
Objectif :		durée : 4h

Matériel : alimentation de laboratoire - multimètre - plaque labdec - composants électroniques - smartphone Android

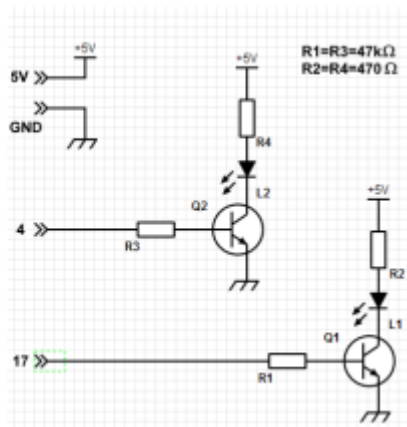
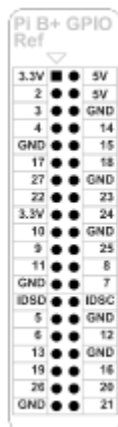
Prérequis : Connexion à distance avec SSH, utilisation des GPIO

Compétences et savoirs principalement visées :

Travail à réaliser :

-

Schéma du système :

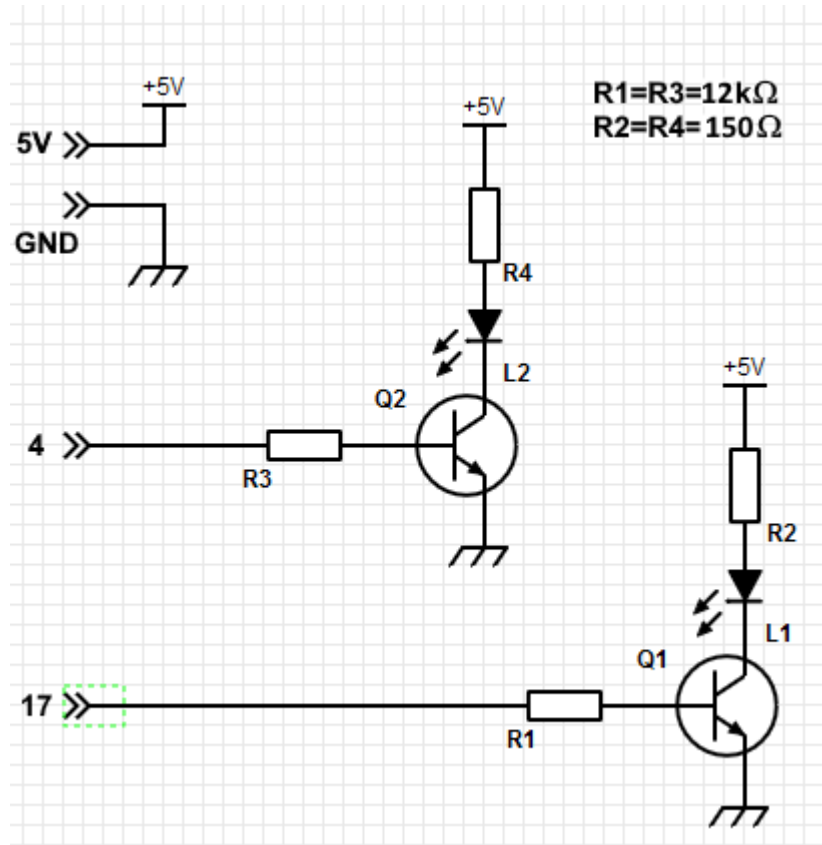


- ◆ Réalise le **montage suivant sur la plaque Labdec**

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



Le programme « cron »

Conseils :

Lis bien toutes les consignes **avant** de commencer

- ◆ L'objectif de l'exercice est de **programmer** l'allumage ou l'extinction d'un éclairage à **une heure** et (optionnellement) à **un jour précis**

ex : Allumer automatiquement un éclairage électrique à 17h00 et l'éteindre à 22h00 tous les dimanches

- ◆ Les systèmes basés sur Linux possèdent une fonction « tâche programmée » appelée « **cron** »
- ◆ Regarde « **cron** » sur Wikipedia pour te renseigner sur les possibilités offertes par « cron » ; recopie ici la définition de « cron » trouvée dans Wikipedia :

- ◆ Les tâches programmées par « cron » sont listées dans un fichier nommé « **crontab** » ; pour consulter ce fichier, utilise la commande « crontab -l » ; recopie ici le contenu de ce fichier :

- ◆ Y a-t-il des tâches programmées actuellement ? Justifie ta réponse

- ◆ D'après Wikipedia, chaque ligne du fichier « crontab » correspond à une *tâche* à exécuter et doit respecter cette notation :

mm hh jj MMM JJJ tâche

- ◆ Rempli le tableau suivant pour expliquer la signification de chacun des 6 champs :

Champ	signification
mm	
hh	
jj	
MMM	
JJJ	
tâche	

Info

Pour **consulter** la la liste (crontab) des taches programmées, il faut utiliser la commande « **crontab -l** » ; pour **modifier** cette liste : « **crontab -e** »

Entraîne-toi à créer une tâche programmée :

- 1) tapes la commande « **crontab -e** » pour pouvoir modifier le fichier crontab ; crée une règle qui lancera le script « **/root/eclairage.sh** » tous les mercredi à 17h04. Une fois la règle écrite, enregistre (Ctrl-O), puis quitte l'éditeur (Ctrl-X). [Recopie cette règle ci-dessous](#) :
- 2) Fais un « **crontab -l** » pour visualiser le contenu de crontab ; tes modifications ont-elles été bien prise en compte ?

Préparation et vérification du système

- ◆ **Vérifie** le câblage, et **testes** le fonctionnement des deux leds, en effectuant cette série de commande :

Commande	Résultat attendu *	Résultat obtenu
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		

* remplir cette case avant de tester la commande

vérifie en faisant : « gpio readall »

→ Les résultats sont-ils conformes ? (sinon essaie d'en découvrir la cause)

- ◆ crée les fichiers **clignote.sh** et **stop-clignote.sh** à l'aide des modèles donnés en annexe
- ◆ **Rends ces scripts exécutables** avec la commande appropriée ; [recopie ces commandes ci-dessous](#) :
- ◆ Pour que le programme **cron** fonctionne avec précision, le réglage de l'heure sur ton RPi doit être correct ; pour le vérifier, tapes la commande « **date** » ; recopie ci-dessous le résultat de cette commande :

Info

Si nécessaire, il est possible de changer l'heure en utilisant la syntaxe suivante :

date -s HH:MM:SS (pour changer uniquement l'heure)

Création d'une tâche programmée (1)

- ◆ Pour connaître l'heure actuelle réglée sur ton Rpi, tapes la commande « **date** »
- ◆ Note l'heure et la minute actuelle :

Heure actuelle	Minute actuelle

- ◆ **Ajoute 10 mn** à la minute actuelle pour définir l'heure programmée :

Heure programmée	Minute programmée

- ◆ Modifie le fichier crontab en tapant la commande « **crontab -e** » ; tu dois ajouter les lignes (attention aux espaces!) :

```
mm      hh    * * * /usr/bin/gpio write 4 1
mm+1    hh    * * * /usr/bin/gpio write 17 1 && /usr/bin/gpio write 4 0
mm+2    hh    * * * /usr/bin/gpio write 17 0
```

où :

- **hh** est l'heure programmée calculée ci-dessus
- **mm** est la minute programmée calculée ci-dessus
- **mm+1** est la minute programmée calculée ci-dessus + 1 mn
- **mm+2** est la minute programmée calculée ci-dessus + 2 mn
- ◆ Enregistre et quitte (Ctrl-O => Entrée => Ctrl-X)
- ◆ D'après toi, que doit-il se passer dans les minutes qui viennent ?
- ◆ Patiente jusqu'à l'heure/minute programmée, puis écris ce que tu constates
- ◆ Patiente encore 1 minute , puis écris ce que tu constates :
- ◆ Patiente encore 1 minute , puis écris ce que tu constates :

Création d'une tâche programmée (2)

- ◆ Pour connaître l'heure actuelle réglée sur ton Rpi, tapes la commande « **date** »
- ◆ Note l'heure et la minute actuelle :

Heure actuelle	Minute actuelle

- ◆ **Ajoute 10 mn** à la minute actuelle pour définir l'heure programmée :

Heure programmée	Minute programmée

- ◆ Modifie le fichier crontab en tapant la commande « **crontab -e** » ; tu dois ajouter les lignes (attention aux espaces!) :

```
mm      hh    * * * /root/clignote.sh  
mm+1    hh    * * * /root/stop-clignote.sh
```

où :

- **hh** est l'heure programmée calculée ci-dessus
- **mm** est la minute programmée calculée ci-dessus
- **mm+1** est la minute programmée calculée ci-dessus + 1mn

- ◆ Enregistre et quitte (**Ctrl-O** => *Entrée* => **Ctrl-X**)
- ◆ D'après toi, que doit-il se passer dans les minutes qui viennent ?
- ◆ Patiente jusqu'à l'heure/minute programmée, puis écris ce que tu constates :
- ◆ Patiente encore 1 minute , puis écris ce que tu constates :

Conclusion

Programmation de « crontab » depuis une interface HTTP

cette partie sera étudiée ultérieurement

Conclusion

Annexe 1 : fichier clignote.sh

```
#!/bin/bash

gpio -g mode 4 out
gpio -g mode 17 out

while [ 1 == 1 ]
do
    gpio -g write 4 0
    gpio -g write 17 0
    sleep 0.2
    gpio -g write 4 1
    gpio -g write 17 1
    sleep 0.2
done
```

Annexe 2 : fichier stop-clignote.sh

```
#!/bin/bash

killall clignote.sh
```

Annexe 2 : exercices

- 1) Écrit une règle qui lancera le script « **/var/www/bin/eclairage.sh** » tous les jours à 19h00 et le dimanche à 17h00.
- 2) Écrit une règle qui éteindra automatiquement le Rpi tous les jours à 22h00
- 3) Écrit une règle qui fera automatiquement la mise à jour du Rpi le 2ème jour de chaque mois à 06h30 du matin ; indice : la commande pour mettre à jour le Rpi est « apt-get upgrade »