Tache CRON toutes les 10 secondes

<http://jerome.colombet.free.fr/?p=70>

Par défaut, le daemon **cron** ne sait pas gérer les secondes.

Voici comment contourner ce problème en créant un script et une nouvelle arborescence **cron**:

Rajouter au fichier **/etc/crontab** les lignes :

```
#Cron toutes les 60 secondes
* * * * * root /usr/sbin/cron.10sec > /dev/null
```

Remarque : Le paramètre **/dev/null** évite l'inondation de mail par le daemon cron.

Création du répertoire accueillant vos scripts à exécuter toutes les 10 secondes:

```
# mkdir /etc/cron.10sec
```

Création du script **/usr/sbin/cron.10sec** :

Ce script sera exécuté toutes les minutes et appellera 5 fois toutes les 10 secondes le script rangé dans le dossier **/etc/cron.10sec**

```
#!/bin/bash
for COUNT in `seq 5` ; do
    run-parts --report /etc/cron.10sec
    sleep 10
done
chmod 755 /usr/sbin/cron.10sec
```

Ensuite créez votre script flash qui sera exécuté toutes les 10 secondes dans **/etc/cron.10sec/**. Vos scripts doivent être en 755 sans extension.

```
#!/bin/bash
# Script flah
# 2 flash de LED toutes les 10 secondes

# Rendre le répertoire /sys/class/gpio actif
cd /sys/class/gpio

# Créer l'accès au port GPIO 20
# Ne rien faire s'il existe déjà
if [ -d "gpio20" ]; then
    echo "e" > /dev/null
else
    echo 20 > export
fi

# Rendre le répertoire gpio20 actif
cd gpio20/

# Configurer le port GPIO 20 en sortie
echo out > direction

# L'instruction echo envoie la valeur 1 dans le fichier value
# Ceci a pour effet d'allumer la LED
echo 1 > value

# La commande usleep suspend l'exécution du programme
# pendant la durée spécifiée (en micro secondes)

sleep 0.05

# L'instruction echo envoie la valeur 0 dans le fichier value
# Ceci a pour effet d'éteindre la LED
echo 0 > value

sleep 0.1
echo 1 > value
sleep 0.05
echo 0 > value
```

Notes :

>/dev/nul => envoie les données vers un « trou noir »

run-parts => Exécute les scripts ou les exécutables présents dans un répertoire