Use the Crontab Timing Task on pcDuino

Cron is the a process of Linux that runs on background. It can be used to execute some tasks at predetermined time.

The format of the command is:

```
crontab[-u user][parameters]
```

In this case, we are using root user.

The parameters are as follows:

- -e: Use the test editor to formulate the task. the default editor is vi, if you want to use other editor you can first set the VISUAL environment variable to specify which editor will be used. ($setenv VISUAL nano)
- -r: Delete the current timing task
- -l: display the current timing tasks

Crontab command format is as following:

```
1
2   f1 f2 f3 f4 f5 [program]
```

while f1 represents minute, f2 represents hour, f3 represents the No. of day in a month, f4 represents the month, f5 represents the No. of day in a week. 'Program' represents the program need to be executed.

Example:

For example, we need to write a string "haha" into the /home/ubuntu/directory test.txt every second, like this:

```
1   $ crontab -e
```

```
 6 #
 7 # To define the time you can provide concrete values for
 8 # minute (m), hour (h), day of month (dom), month (mon),
 9 # and day of week (dow) or use '*' in these fields (for 'any').#
10 # Notice that tasks will be started based on the cron's system
11 # daemon's notion of time and timezones.
12 #
13 # Output of the crontab jobs (including errors) is sent through
14 # email to the user the crontab file belongs to (unless redirected).
15 #
16 # For example, you can run a backup of all your user accounts
17 # at 5 a.m every week with:
18 # 0 5 * * 1 tar -zcf /var/backups/home.tgz /home/
19 #
20 # For more information see the manual pages of crontab(5) and cron(8)
21 #
22 # m h  dom mon dow   command
23 #*/1 * * * * /home/ubuntu/yeelink.sh
24 */1 * * * * echo "haha" > /home/ubuntu/test.txt
```

Here I will explain the means of each line at the crontab. The each lines at the crontab represent tasks performed regularly, divided into six parts. F1-f5 indicates when to execute the command , [program] indicate the execution command, Each section separated by the space, except for the last section(command) can use the space inside, the other section cannot use the space. The former 5 parts represent the minute, hour, day, month, week, each part of the value range as follows:

Minute   0 – 59

Hours   0 – 23

Day   1 – 31

Month   1 – 12

Week   0 – 6   0 represent the Sunday

In addition to these fixed values, it can also match with an asterisk (*), comma (,), and the slash (/) to indicate some other meaning:

Asterisk   indicates an arbitrary value, such as in the hour part fill * represents any hour (hourly)

Comma   allows multiple values to fill in a section, for example, in comma part to fill 1,3 represents a minute or three minutes

Slash   in general match with * representatives how often, for example, to fill in the hours section * / 2 for every two minutes. So * / 1 and * there is no difference

Results are as follows:

```
root@ubuntu:/home/ubuntu# vi test.txt

  1 haha
```