

## 2.Driving\_the\_LCD\_screen

After CPU Info LCD screen is correctly inserted into the Raspberry Pi, you need to install the driver. The user can display information such as CPU occupancy, Memory occupancy, CPU temperature,etc.

### 1. Install the wiringPi library

CPU Info LCD screen is used for data communication through the GPIO port of the Raspberry Pi, so we must install the wiringPi library file.

Enter the following command to install the wiringPi library. Users who have already installed the wiringPi library can ignore this step.

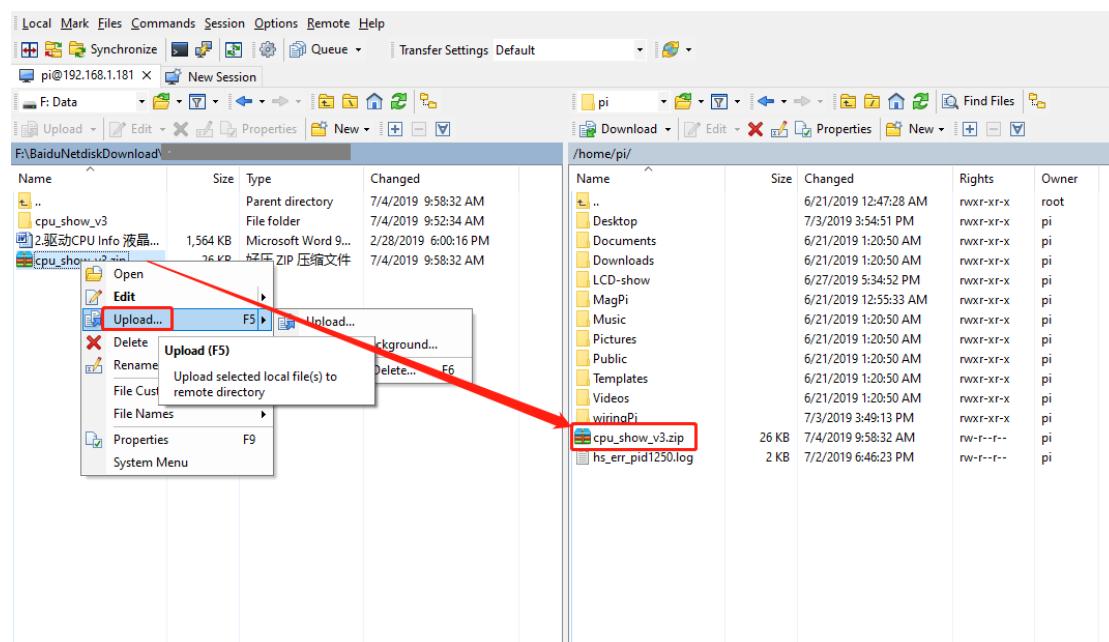
```
cd ~
git clone git://git.drogon.net/wiringPi
cd wiringPi
./build
```

### 2.Install Drive

#### 2.1 Transfer the driver file to the Raspberry Pi

You need to install the WinSCP tool on your computer. After connecting to the Raspberry Pi, transfer the [cpu\\_show\\_v3.zip](#) package from this folder to the pi directory of the Raspberry Pi.

As shown below, drag and drop [cpu\\_show\\_v3.zip](#) directly into the Raspberry Pi system.



### 2.2 Extract file

Open the Raspberry Pi terminal and find the [cpu\\_show\\_v3.zip](#) file.

Enter command:

```
ls
```

```
pi@raspberrypi:~ $ ls
cpu_show_v3.zip  LCD-show  pi_cpu_V2.0.zip  Templates
Desktop          MagPi    Pictures        thinclient_drives
Documents         matchbox-keyboard  Public       Videos
Downloads         Music    python_games   wiringPi
pi@raspberrypi:~ $
```

Enter command:

**unzip cpu\_show\_v3.zip**

```
pi@raspberrypi:~ $ unzip cpu_show_v3.zip
Archive:  cpu_show_v3.zip
  creating: cpu_show_v3/cpu_show/
  creating: cpu_show_v3/cpu_show/BL/
  inflating: cpu_show_v3/cpu_show/BL/bl
  inflating: cpu_show_v3/cpu_show/BL/test.c
  inflating: cpu_show_v3/cpu_show/cpushow
  creating: cpu_show_v3/cpu_show/cputemp/
  inflating: cpu_show_v3/cpu_show/cputemp/cputemp.c
  inflating: cpu_show_v3/cpu_show/cputemp/temp
  inflating: cpu_show_v3/cpu_show/PCD8544.c
  inflating: cpu_show_v3/cpu_show/PCD8544.h
  inflating: cpu_show_v3/cpu_show/pcd8544_rpi.c
  inflating: cpu_show_v3/cpu_show/README.txt
pi@raspberrypi:~ $
```

## 2.3 Enter the program folder

Enter command:

**cd ~/cpu\_show\_v3/cpu\_show**

**ls**

```
pi@raspberrypi:~ $ cd ~/cpu_show_v3/cpu_show
pi@raspberrypi:~/cpu_show_v3/cpu_show $ ls
BL  cpushow  cputemp  PCD8544.c  PCD8544.h  pcd8544_rpi.c  README.txt
pi@raspberrypi:~/cpu_show_v3/cpu_show $
```

## 2.4 Compiler file

Enter command:

**cc -o cpushow pcd8544\_rpi.c PCD8544.c -L/usr/local/lib -lwiringPi**

```
pi@raspberrypi:~/cpu_show_v3/cpu_show $ cc -o cpushow pcd8544_rpi.c PCD8544.c -L/usr/local/lib -lwiringPi
pcd8544_rpi.c: In function 'main':
pcd8544_rpi.c:176:6: warning: implicit declaration of function 'read' [-Wimplicit-function-declaration]
  if (read(fd, buf, MAX_SIZE) < 0)
  ^
pcd8544_rpi.c:190:2: warning: implicit declaration of function 'close' [-Wimplicit-function-declaration]
  close(fd);
  ^
pi@raspberrypi:~/cpu_show_v3/cpu_show $ ls
BL  cpushow  cputemp  PCD8544.c  PCD8544.h  pcd8544_rpi.c  README.txt
```

Check again with the **ls** command, cpushow has become an executable file.

cc is the compile command, -o is the compile parameter, cpushow is the generated program name, pcd8544\_rpi.c and PCD8544.c are the source files in the current directory, -L/usr/local/lib and -lwiringPi are referenced libraries

file.

## 2.5 Running procedure

Enter command:

**./cpushow**

```
pi@raspberrypi:~/cpu_show_v3/cpu_show $ ./cpushow
Raspberry Pi PCD8544 sysinfo display
=====
temp: 55.50
temp: 55.50
temp: 55.99
temp: 54.53
temp: 55.02
temp: 56.48
temp: 55.99
temp: 55.99
```

The system will print out of the current CPU temperature value and display the following on the CPU Info screen.



After the above steps are performed, the basic display can be normally performed.

## 3. Set the system to automatically start the program when booting

Enter command to create startup script start.sh

**nano /home/pi/cpu\_show\_v3/start.sh**

Enter following content in the start.sh:

```
#!/bin/sh  
cd /home/pi/cpu_show_v3/cpu_show  
.cpushow
```

After enter is complete, we need to press **ctrl+X**, then we need to press “Y” to save, then we need to press “Enter” on the keyboard.

Next, we need to create a new boot program:

Input the following command to open the .config folder:

```
cd /home/pi/.config
```

Create a new autostart folder and ignore this step if it already exists:

```
mkdir autostart
```

Input the following command to enter the autostart folder:

```
cd autostart
```

Create self-starting shortcut:

```
nano start.desktop
```

Then,we need to enter the following:

[Desktop Entry]  
Type=Application  
Exec=/home/pi/cpu\_show\_v3/start.sh

After enter is complete, we need to press **ctrl+X**, then we need to press “Y” to save, then we need to press “Enter” on the keyboard.

Exec = “the path to the startup script.”

Since this self-starting method needs to be started after the desktop is started, the startup will be slower.

If it is found that it cannot be self-started after adding, please check if there is a “#” in front of **hdmi\_force\_hotplug=1** in the **/boot/config.txt** file.

If there is a # number, we need to delete the “#” . As shown below.

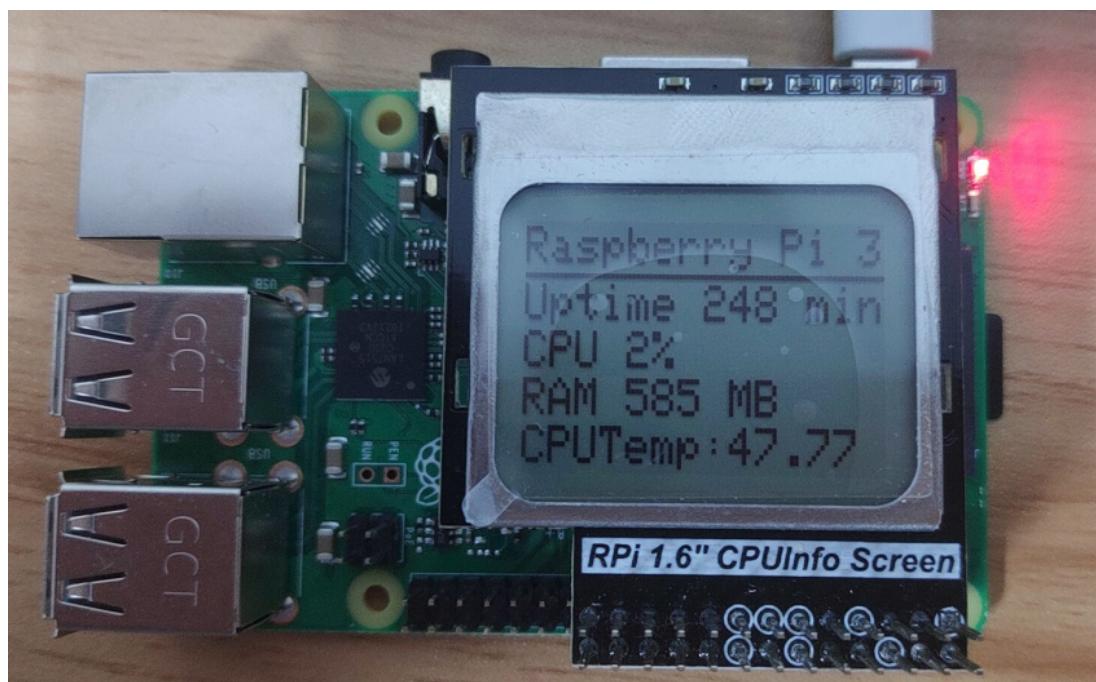
```
# uncomment to force a console size. By default it will be display's size minus  
# overscan.  
#framebuffer_width=1280  
#framebuffer_height=720  
  
# uncomment if hdmi display is not detected and composite is being output  
hdmi_force_hotplug=1  
  
# uncomment to force a specific HDMI mode (this will force VGA)  
#hdmi_group=1  
#hdmi_mode=1
```

#### 4.Restart Raspberry Pi

Enter command:

**sudo reboot**

After restarting, the CPU Info LCD will display the Raspberry Pi usage time, CPU usage, Memory occupancy, and CPU temperature. As shown in the figure below.



After completing the above steps, the CPU Info LCD screen is ready for use.