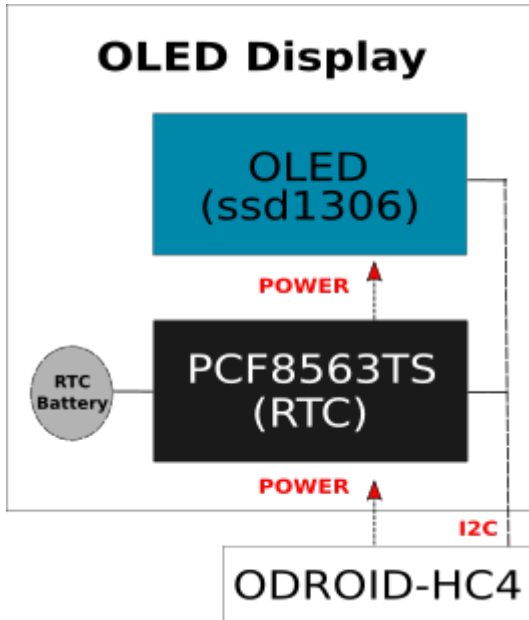
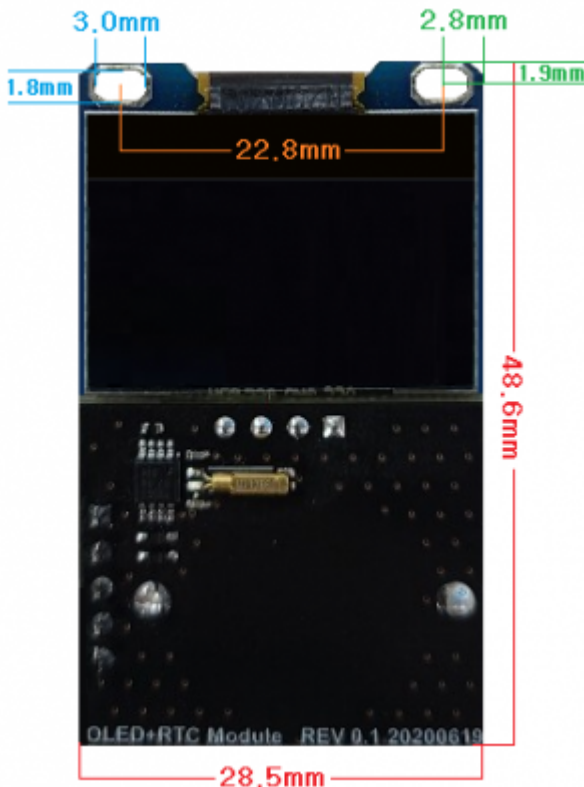


OLED display

This page introduces how to use OLED display with ODROID-HC4(-P) and the commands and packages in this page are based on Ubuntu 20.04.



Rough Drawings



Schematics

RTC Module for HC4

Connection Check

If you have done the wiring well, you can see the device as the following commands.


```
sudo apt install i2c-tools
```

When you have wired the RTC module and OLED

```
sudo i2cdetect -y -r 0
```

RTC

```
odroid@hc4:~$ sudo i2cdetect -y -r 0
    0  1  2  3  4  5  6  7  8  9  a  b  c  d  e  f
00:  -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
10:  -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
20:  -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
30:  -- -- -- -- -- -- -- -- -- -- -- -- UU -- --
40:  -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
50:  -- 51 -- -- -- -- -- -- -- -- -- -- -- -- --
60:  -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
70:  -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
```

 I2C slave address of OLED is 0x3C. "UU" indicates that probing of this address was skipped because the address is currently in use by a driver.

Installation instructions

Simplest way for simple usage

The simplest way to install the default packages to display date and time on OLED is to install the package prebuilt.

target

```
$ sudo apt update
```

```
$ sudo apt install odroid-homecloud-display
```

Source code of this package : <https://github.com/tobetter/odroid-homecloud>

For development

If you are willing to use the OLED for a development, you will need to install more packages after 'odroid-homecloud-display'. So please follow the instructions.

Firstly, install the dependency packages.

target

```
$ sudo apt install git python-dev python3-pip libfreetype6-dev libjpeg-dev build-essential
$ sudo apt install libSDL-dev libportmidi-dev libSDL-ttf2.0-dev libSDL-mixer1.2-dev libSDL-mixer1.2 libfluidsynth2 libSDL2-2.0-0 libSDL-image1.2-dev
```

Secondly, download the example codes from Github.

target

```
$ git clone https://github.com/rm-hull/luma.examples.git
$ cd luma.examples
```

Finally, install the python packages to run the examples.

target

```
$ sudo -H pip3 install -e .
```

Please refer to the link for more detail about the examples.

<https://github.com/rm-hull/luma.examples#running-the-examples>

Testing examples

target

```
$ cd examples
$ sudo python3 3d_box.py --i2c-port 0 --rotate 2
Version: luma.oled 3.6.0 (luma.core 1.17.2)
Display: ssd1306
```

```
Interface: i2c
Dimensions: 128 x 64
```

Digital clock

This example is to show the current date and time to the display, download the below code as **clock.py**.

```
#!/usr/bin/env python3
# -*- coding: utf-8 -*-
# Copyright (c) 2014-18 Richard Hull and contributors
# See LICENSE.rst for details.
# PYTHON_ARGCOMPLETE_OK

"""
An analog clockface with date & time.
"""

from luma.core.interface.serial import i2c
from luma.core.render import canvas
from luma.oled.device import ssd1306, ssd1325, ssd1331, sh1106
from time import sleep
from PIL import ImageFont, ImageDraw, Image
import time
import datetime

serial = i2c(port=0, address=0x3C)
# device = sh1106(serial, rotate=0)
device = ssd1306(serial, rotate=2)

def main():
    today_last_time = "Unknown"
    while True:
        now = datetime.datetime.now()
        today_date = now.strftime("%d %b %y")
        today_time = now.strftime("%H:%M")
        if today_time != today_last_time:
            today_last_time = today_time
            with canvas(device) as draw:
                font = ImageFont.truetype('/root/fonts-DSEG_v046/DSEG7-Modern/DSEG7Modern-Bold.ttf', 36)
                # font =
                ImageFont.truetype('/root/luma.examples/examples/fonts/FreePixel.ttf', 36)
                draw.text((0, 27), today_time, font=font, fill=1)
                font = ImageFont.truetype('/root/fonts-DSEG_v046/DSEG7-Modern/DSEG7Modern-Bold.ttf', 20)
                # font =
```

```
ImageFont.truetype('/root/luma.examples/examples/fonts/FreePixel.ttf',20)
    draw.text((0, 0), today_date, font=font, fill=1)
    time.sleep(0.1)

if __name__ == "__main__":
    try:
        main()
    except KeyboardInterrupt:
        pass
```

This script uses the true type font that is not included in the library, therefore additional fonts must be downloaded and installed to a directory. We will assume that the font file and python script is copied to **/root** directory in this example.

target

```
root@odroid ~# wget
https://github.com/keshikan/DSEG/releases/download/v0.46/fonts-DSEG_v046.zip
root@odroid ~# unzip fonts-DSEG_v046.zip
root@odroid ~# ls -l
drwxr-xr-x 12 root root  4096 Mar 15  2020 fonts-DSEG_v046
```

After copying the font files, run the python script **clock.py**.

target

```
root@odroid ~# ./clock.py
```

If you want to change the font or its size, change the font name and its size in the script.

```
font = ImageFont.truetype('/root/fonts-DSEG_v046/DSEG7-Modern/DSEG7Modern-Bold.ttf',36)
```

From:
<https://wiki.odroid.com/> - **ODROID Wiki**

Permanent link:
https://wiki.odroid.com/odroid-hc4/application_note/oled

Last update: **2022/11/15 17:53**

