CEC Continuing Education Center

DesignNews

Embedded Linux Primer

Day 5:

Generate a Custom Raspberry Pi Compute Module 5 Yocto Image

Sponsored by

DigiKey

informa markets

Webinar Logistics

- Turn on your system sound to hear the streaming presentation.
- If you have technical problems, click “Help” or submit a question asking for assistance.
- Participate in ‘Attendee Chat’ by maximizing the chat widget in your dock.

Fred Eady

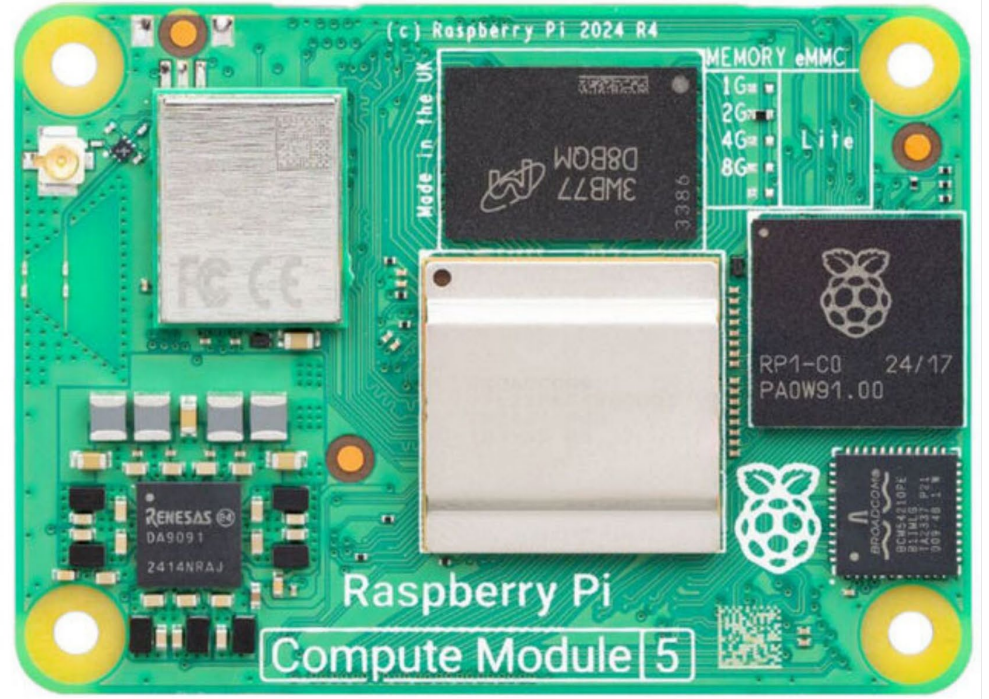
Visit ‘Lecturer Profile’ in your console for more details.

CEC Continuing Education Center

AGENDA

DigiKey

- Yocto CM5 Image Configuration
- Cook Our CM5 Yocto Base Image
- Load and Boot Our Yocto CM5 Image
- Setup and Enable the CM5 WiFi Radio
- Create a GPIO Yocto Application
- Create an LM75 Yocto Application



yocto PROJECT

bblayers.conf

bblayers.conf

yocto > build-cm5spi > conf > bblayers.conf

```
# RASPBERRY PI COMPUTE MODULE 5
# 08/01/2025
# working
# base build with nano - ssh - libgpiod - systemd - spi - i2c
# optional   /home/fred/pi5yoctoProject/yocto/meta-openembedded/meta-networking \
#
#
# POKY_BBLAYERS_CONF_VERSION is increased each time build/conf/bblayers.conf
# changes incompatibly
POKY_BBLAYERS_CONF_VERSION = "2"

BBPATH = "${TOPDIR}"
BBFILES ?= ""

BBLAYERS ?= " \
  /home/fred/pi5yoctoProject/yocto/poky/meta \
  /home/fred/pi5yoctoProject/yocto/poky/meta-poky \
  /home/fred/pi5yoctoProject/yocto/poky/meta-yocto-bsp \
  /home/fred/pi5yoctoProject/yocto/meta-openembedded/meta-oe \
  /home/fred/pi5yoctoProject/yocto/meta-openembedded/meta-python \
  /home/fred/pi5yoctoProject/yocto/meta-raspberrypi \
  "
```

local.conf

local.conf

yocto > build-cm5spi > conf > local.conf

```
1   # Raspberry Pi CM5
2   # 08/01/2025
3   # working
4   # systemd - uart - i2c - spi - libgpiod - SSH
5
6
7   # There are also the following hardware board target machines included for
8   # demonstration purposes:
9   #
10  #MACHINE ?= "beaglebone-yocto"
11  #MACHINE ?= "genericarm64"
12  #MACHINE ?= "genericx86"
13  #MACHINE ?= "genericx86-64"
14  #
15  # This sets the default machine to be qemux86-64 if no other machine is selected:
16  MACHINE ??= "raspberrypi5"
```

local.conf

local.conf

yocto > build-cm5spi > conf > local.conf

```
289  # CONF_VERSION is increased each time build/conf/ changes incompatibly and is used to
290  # track the version of this file when it was generated. This can safely be ignored if
291  # this doesn't mean anything to you.
292  CONF_VERSION = "2"
293
294  # Enable WiFi configuration via networkctl
295  INIT_MANAGER = "systemd"
296  PREFERRED_PROVIDER_udev = "systemd"
297  PREFERRED_PROVIDER_udev_utils = "systemd"
298  DISTRO_FEATURES_BACKFILL_CONSIDERED = "sysvinit"
299  IMX_DEFAULT_DISTRO_FEATURES += "systemd"
300
301  # Enable the UART
302  # Enable SPI Support
303  # Enable I2C Support
304  # Enable nano text editor
305  # synaptics-killswitch is required
306  ENABLE_UART = "1"
307  ENABLE_I2C = "1"
308  ENABLE_SPI_BUS = "1"
309  KERNEL_MODULE_AUTOLOAD:rpi += " i2c-dev"
310  IMAGE_INSTALL:append = " \
311      i2c-tools \
312      nano \
313      libgpiod \
314      libgpiod-dev \
315      libgpiod-tools \
316      openssh \
317      "
318  LICENSE_FLAGS_ACCEPTED = "synaptics-killswitch"
```

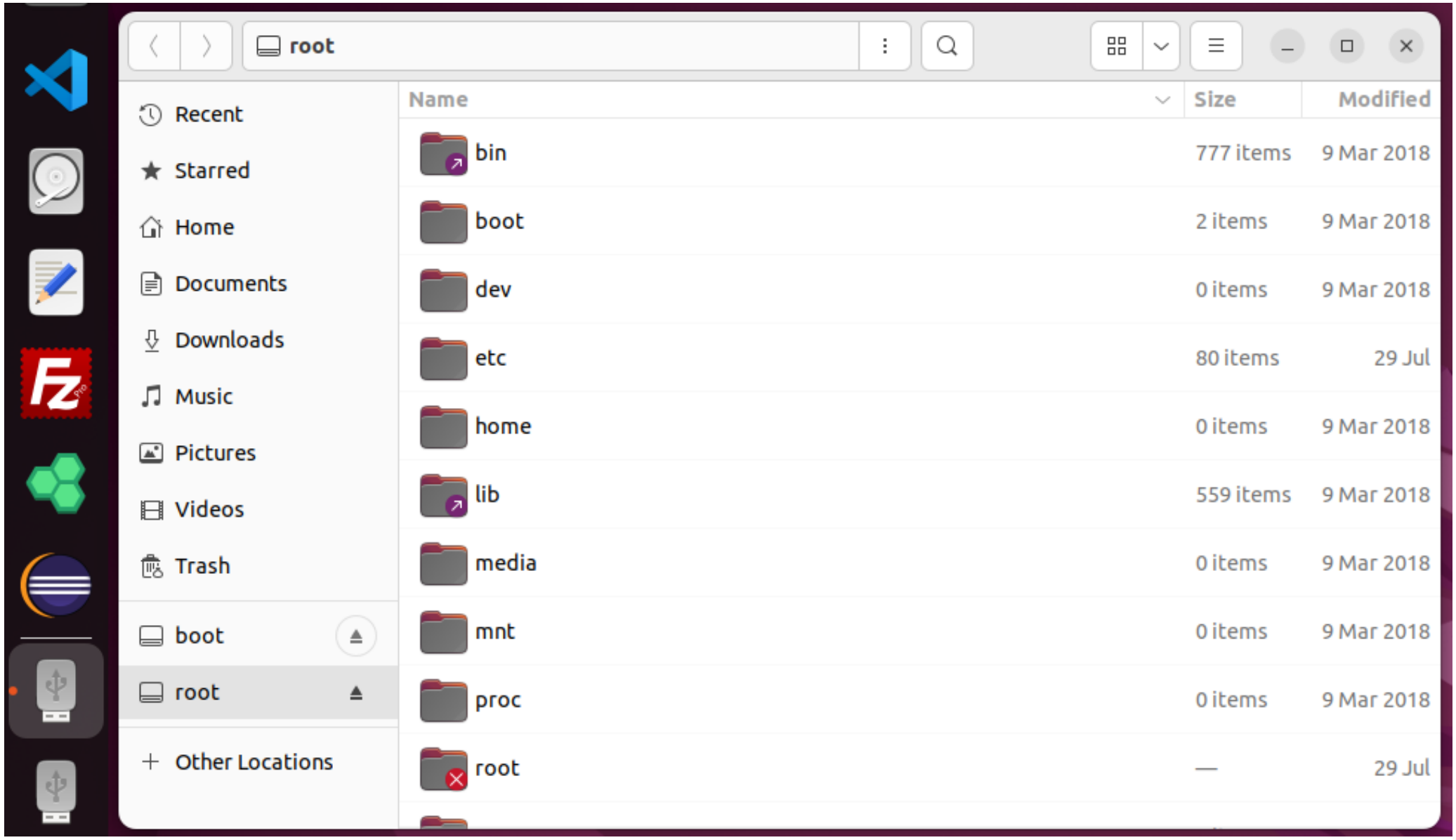
Invoke BitBake

```
fred@pi5-yocto: ~/pi5yoctoProject/yocto/build-cm5spi
File  Edit  View  Search  Terminal  Help

fred@pi5-yocto:~/pi5yoctoProject/yocto/build-cm5spi$ bitbake core-image-base
Loading cache: 100% |                                              | ETA:  --:--:--
Loaded 0 entries from dependency cache.
Parsing recipes: 100% |##########################################| Time: 0:00:29
Parsing of 2513 .bb files complete (0 cached, 2513 parsed). 4353 targets, 124 sk
ipped, 0 masked, 0 errors.
NOTE: Resolving any missing task queue dependencies

Build Configuration:
BB_VERSION           = "2.8.0"
BUILD_SYS            = "x86_64-linux"
NATIVELSBSTRING      = "universal"
TARGET_SYS           = "aarch64-poky-linux"
MACHINE              = "raspberrypi5"
DISTRO               = "poky"
DISTRO_VERSION       = "5.0.10"
TUNE_FEATURES        = "aarch64 crypto cortexa76"
TARGET_FPU           = ""
meta
meta-poky
meta-yocto-bsp       = "scarthgap:ac257900c33754957b2696529682029d997a8f28"
meta-oe
meta-python          = "scarthgap:491671faee11ea131feab5a3a451d1a01deb2ab1"
```

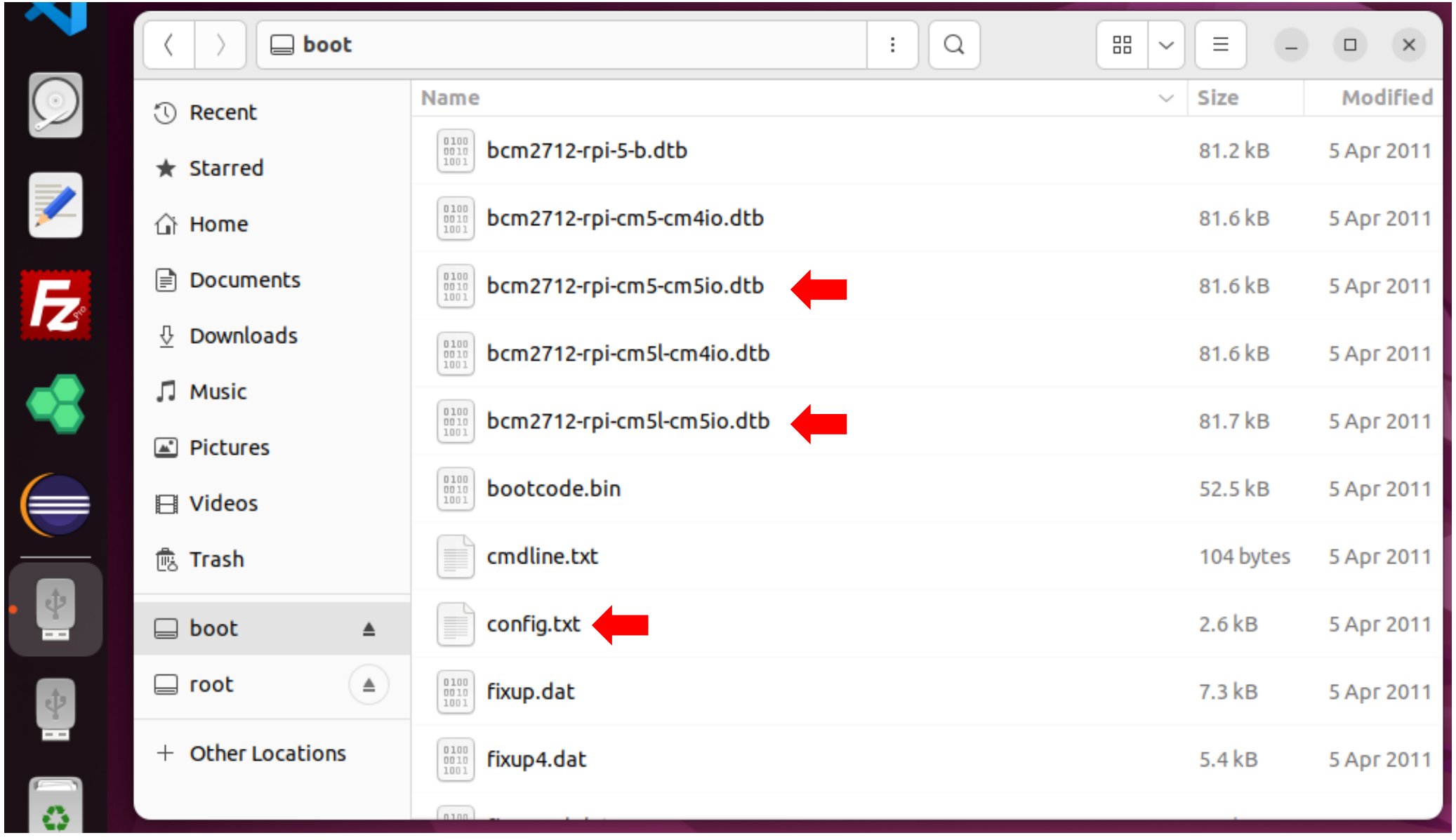
Load the microSD Card With Our New CM5 Image

```
fred@pi5-yocto: ~/pi5yoctoProject/yocto/build-cm5spi/tmp/deploy/images/raspberrypi5
File  Edit  View  Search  Terminal  Help
fred@pi5-yocto:~/pi5yoctoProject/yocto/build-cm5spi/tmp/deploy/images/raspberrypi5$ sudo bmaptool copy core-image-base-raspberrypi5.rootfs.wic.bz2 /dev/sdb
[sudo] password for fred:
bmaptool: info: discovered bmap file 'core-image-base-raspberrypi5.rootfs.wic.bmap'
bmaptool: info: block map format version 2.0
bmaptool: info: 242483 blocks of size 4096 (947.2 MiB), mapped 130166 blocks (508.5 MiB or 53.7%)
bmaptool: info: copying image 'core-image-base-raspberrypi5.rootfs.wic.bz2' to block device '/dev/sdb' using bmap file 'core-image-base-raspberrypi5.rootfs.
wic.bmap'
bmaptool: info: 100% copied
bmaptool: info: synchronizing '/dev/sdb'
bmaptool: info: copying time: 23.6s, copying speed 21.6 MiB/sec
fred@pi5-yocto:~/pi5yoctoProject/yocto/build-cm5spi/tmp/deploy/images/raspberrypi5$
```

microSD Card root Partition (rootfs)



microSD Card boot Partition (bootfs)



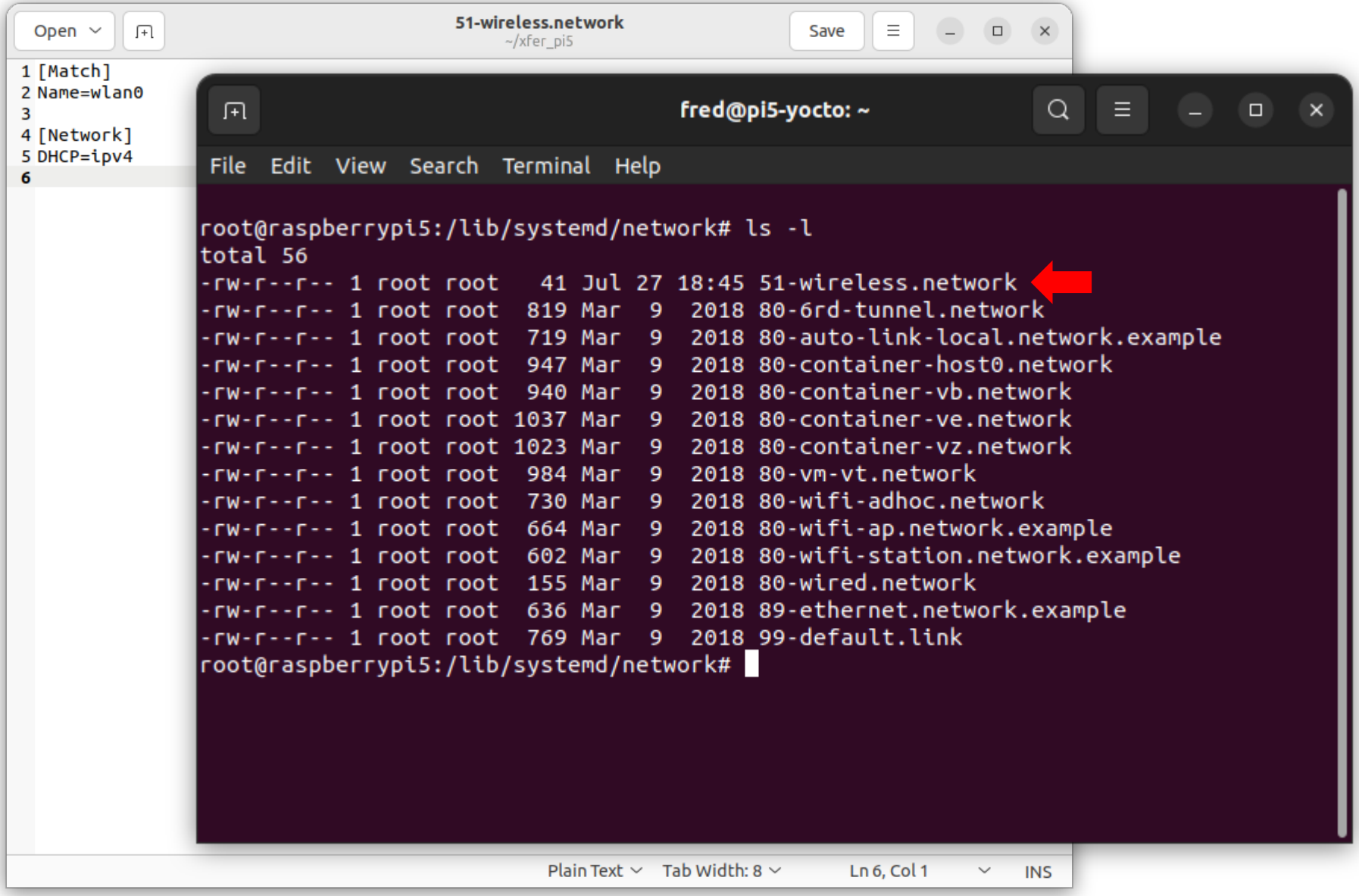
microSD Card boot Partition (bootfs) – config.txt

Open | config.txt | boot /media/fred/boot | Save

```
217
218 #dtparam=i2c_vc_baudrate=100000
219
220 #dtparam=i2s=off
221
222 #dtparam=spi=off
223
224 #dtparam=random=off
225
226 #dtparam=uart0=on
227
228 #dtparam=watchdog=off
229
230 #dtparam=act_led_trigger=mmc
231
232 #dtparam=act_led_activelow=off
233
234 #dtparam=act_led_gpio=47
235
236 #dtparam=pwr_led_trigger=mmc
237
238 #dtparam=pwr_led_activelow=off
239
240 #dtparam=pwr_led_gpio=35
241
242
243 #dtoverlay=act-led,activelow=off
244 # Enable SPI bus
245 dtparam=spi=on
246 # Enable I2C
247 dtparam=i2c1=on
248 dtparam=i2c_arm=on
249 # Enable UART
250 enable_uart=1
251 # Enable VC4 Graphics
252 dtoverlay=vc4-kms-v3d
253
```

Plain Text | Tab Width: 8 | Ln 1, Col 10 | INS

Create and Place 51-wireless.network



Create and Place wpa_supplicant-wlan0.conf

Open *wpa_supplicant-wlan0.conf ~/xfer_pi5 Save

```
1 ctrl_interface=/var/run/wpa_supplicant
2 eapol_version=1
3 ap_scan=1
4 fast_reauth=1
5
6
```

Open *wpa_supplicant-wlan0.conf ~/xfer_pi5 Save

```
1 ctrl_interface=/var/run/wpa_supplicant
2 eapol_version=1
3 ap_scan=1
4 fast_reauth=1
5
6 network={
7         ssid="SSID_OF_NETWORK "
8         #psk="PASSWORD_OF_NETWORK"
9         psk=9c792aa9d02f7054501f156a58f9c73478f81f82b972a73da347fb19971bf045
10 }
11
```

wpa_passphrase SSID_OF_NETWORK PASSWORD_OF_NETWORK >> /etc/wpa_supplicant/wpa_supplicant-wlan0.conf

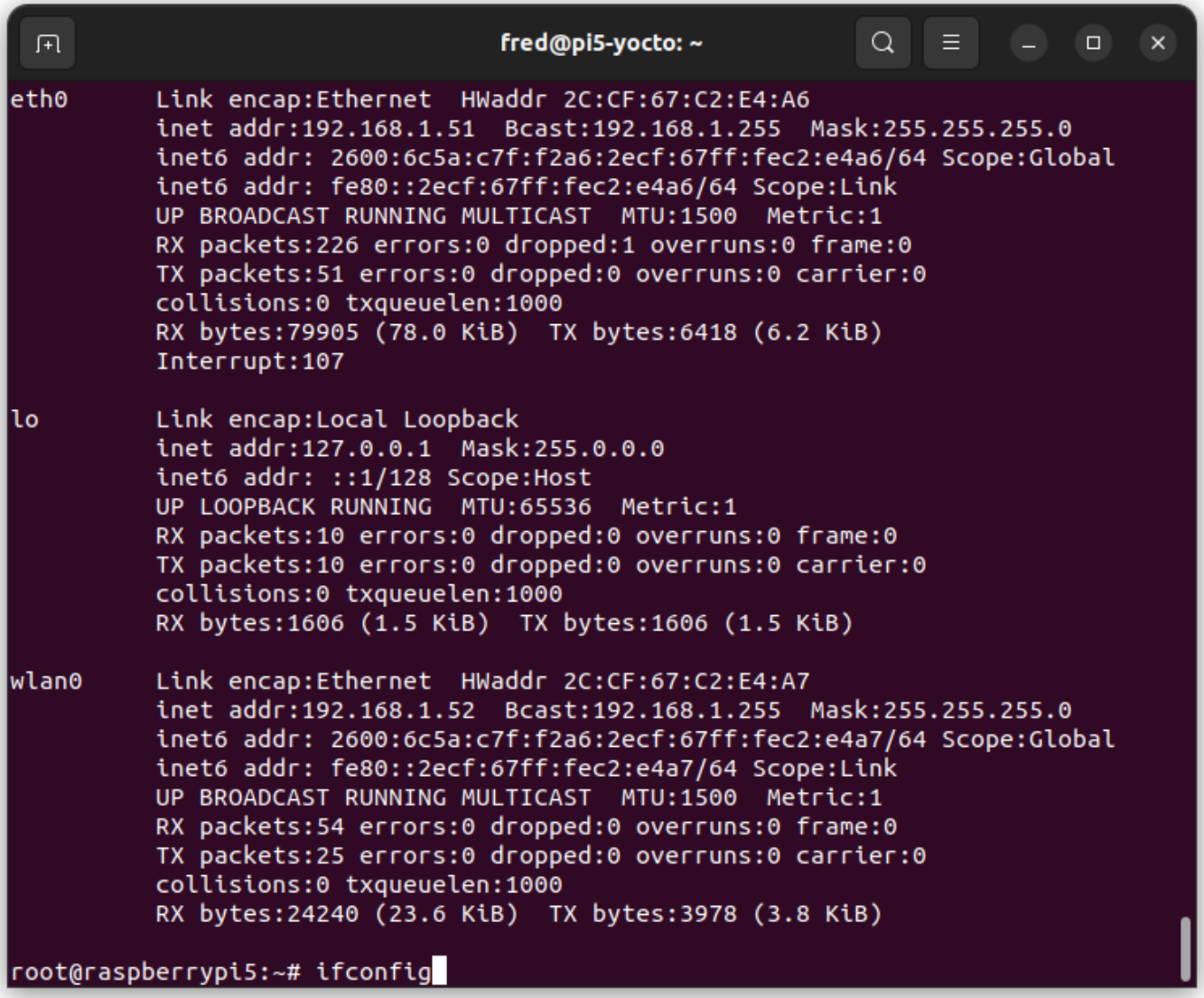
fred@pi5-yocto: ~

File Edit View Search Terminal Help

```
root@raspberrypi5:~# cd /etc/wpa_supplicant
root@raspberrypi5:/etc/wpa_supplicant# ls -l
total 4
-rw-r--r-- 1 root root 202 Jul 29 22:40 wpa_supplicant-wlan0.conf
root@raspberrypi5:/etc/wpa_supplicant#
```

Plain Text Tab Width: 8 Plain Text Tab Width: 8 Ln 11, Col 1 INS

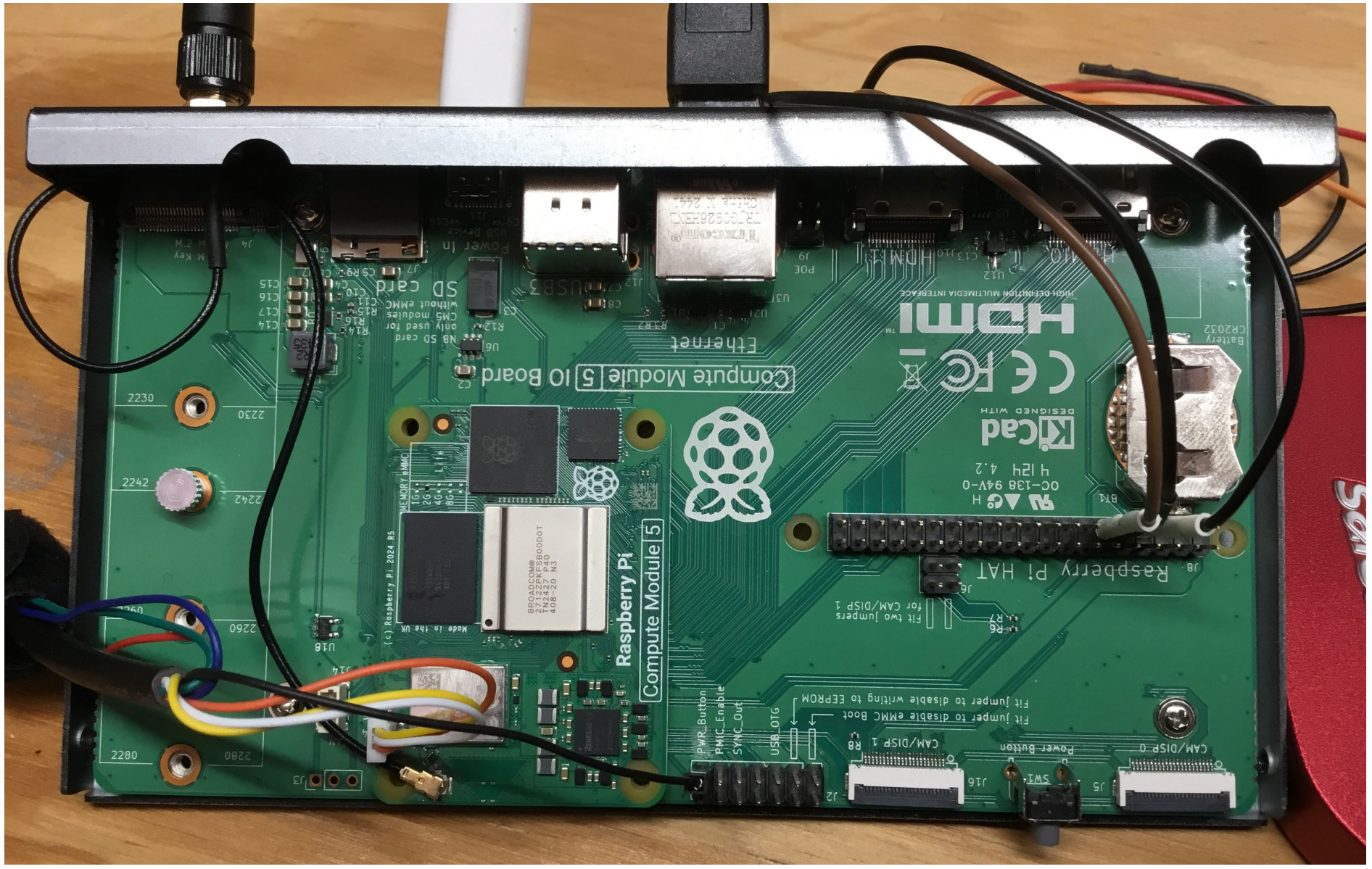
Enable the Compute Module 5 WiFi Radio



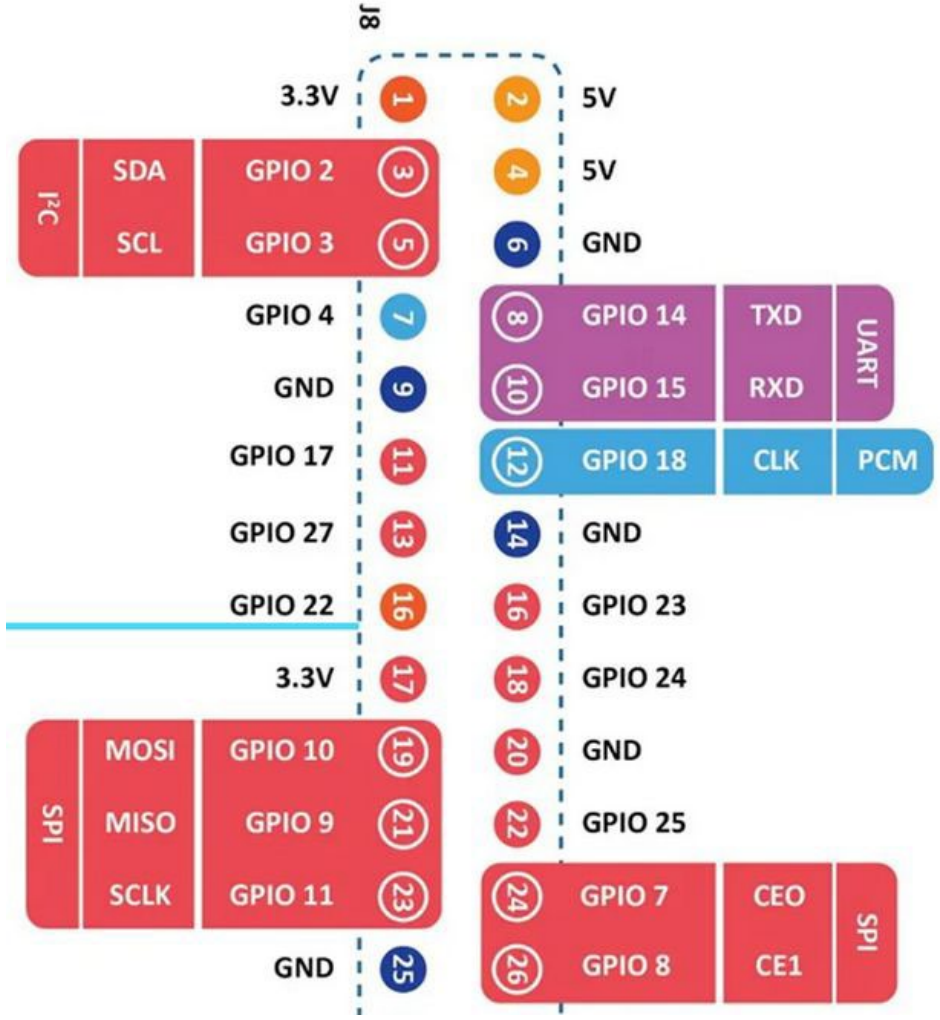
To enable WiFi , execute the following 3 commands from the Raspberry Pi Compute Module 5 Linux console :

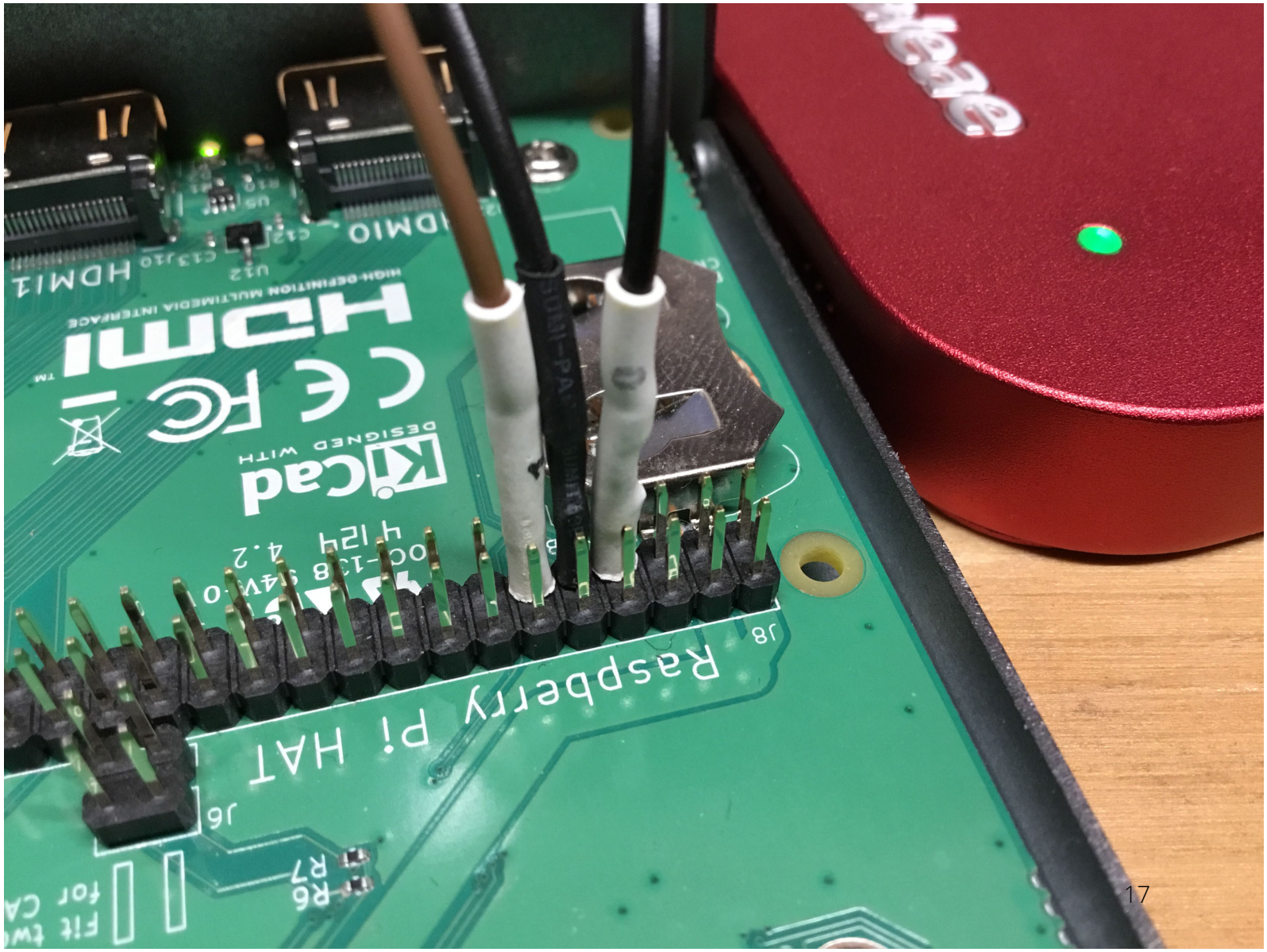
```
systemctl enable wpa_supplicant@wlan0.service
systemctl restart systemd-networkd.service
systemctl restart wpa_supplicant@wlan0.service
```

Assemble the Raspberry Pi Compute Module 5 Hardware

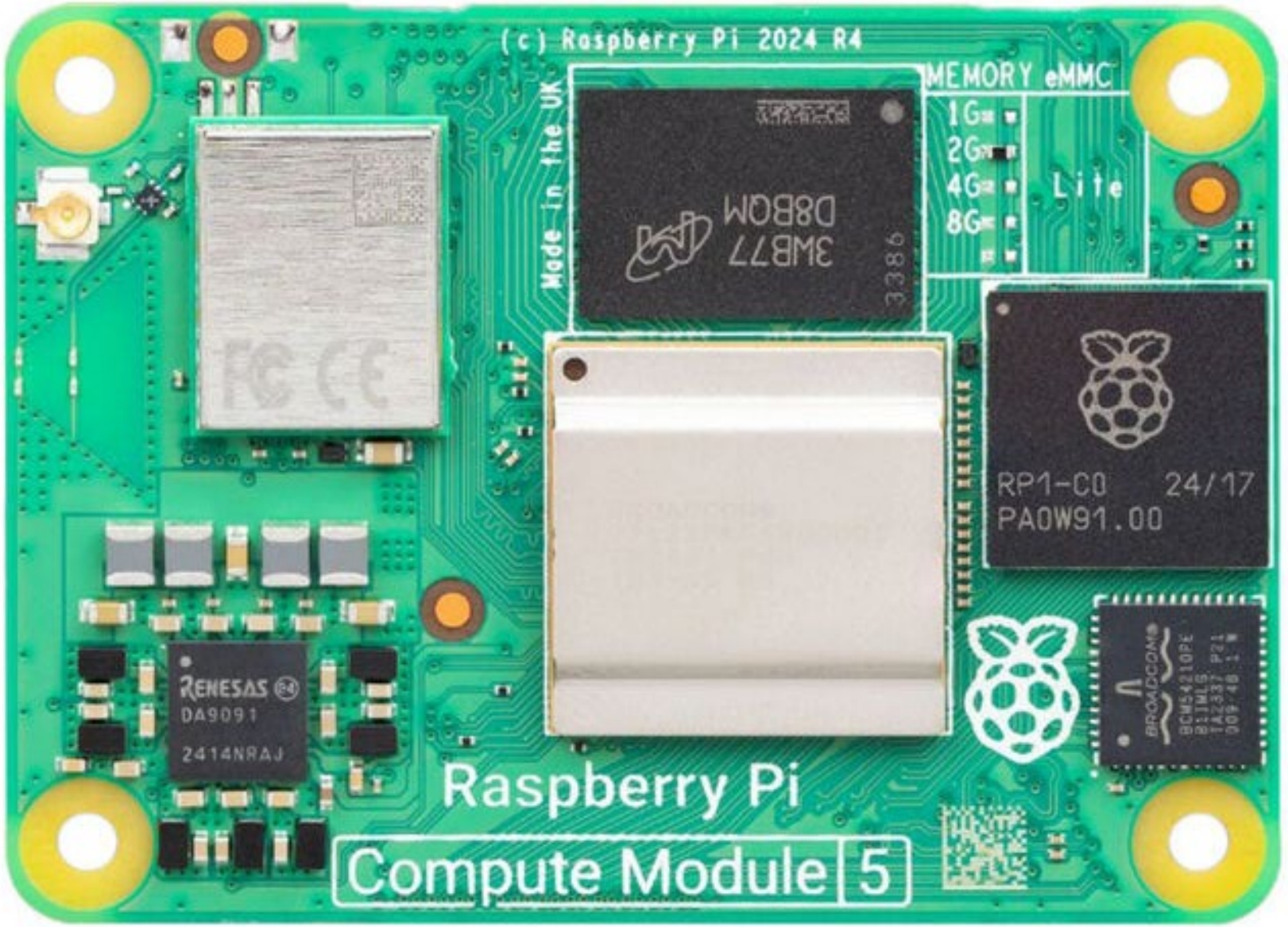


Assemble the Raspberry Pi Compute Module 5 Hardware

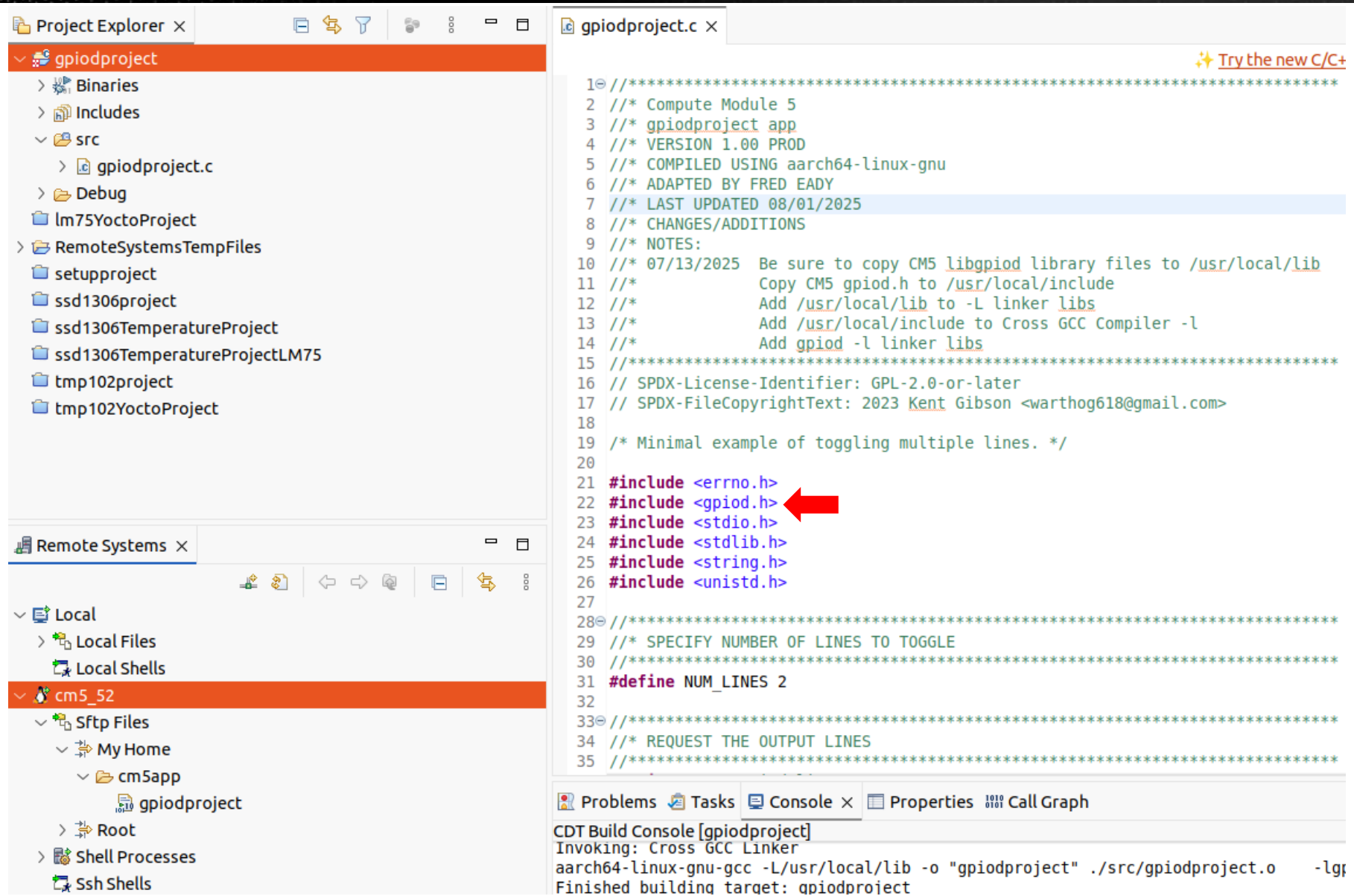




Eclipse Yocto Application Development



Create and Compile the Application Code



Main Application Flow

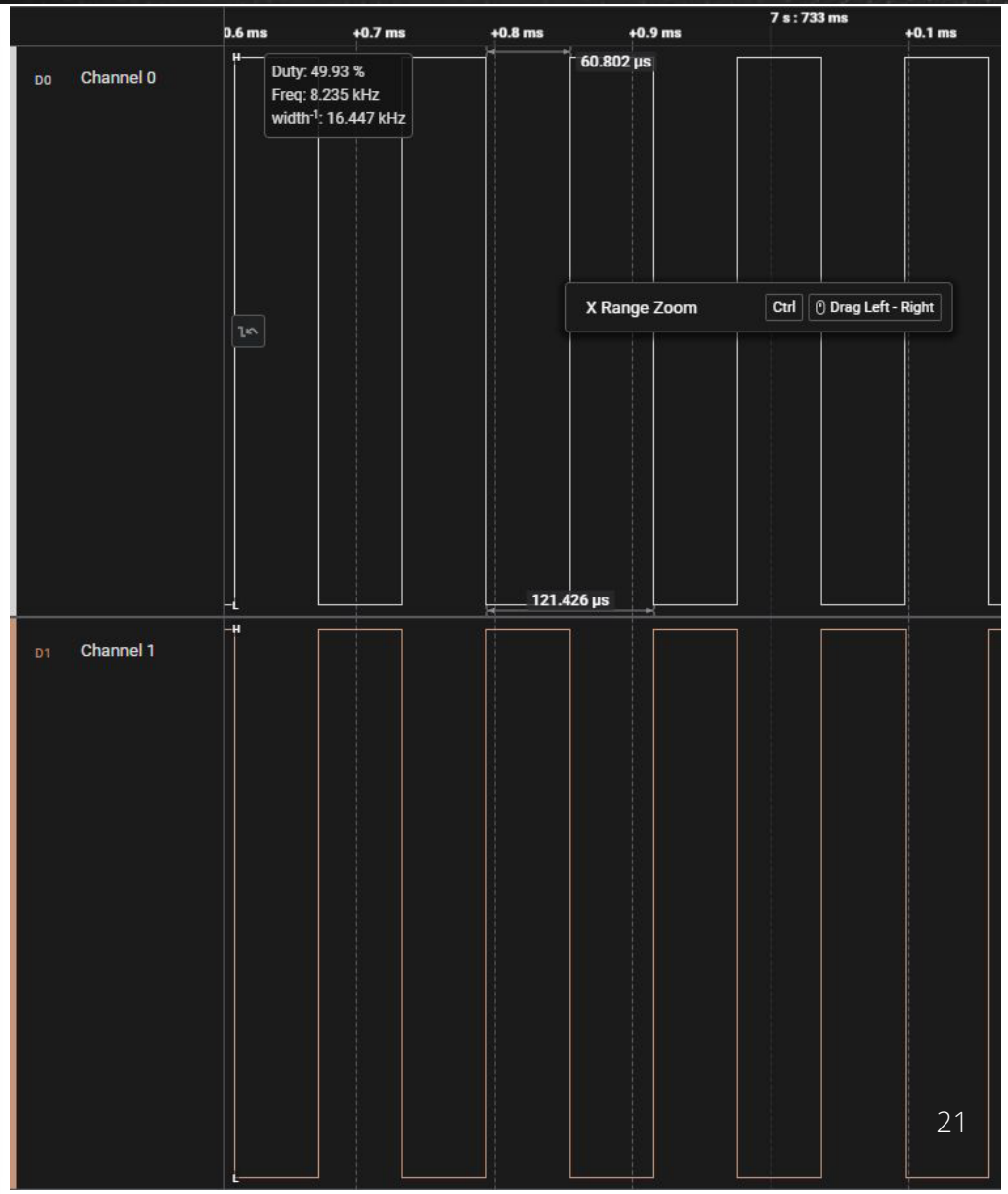
gpiodproject.c

gpiodproject > src > gpiodproject.c

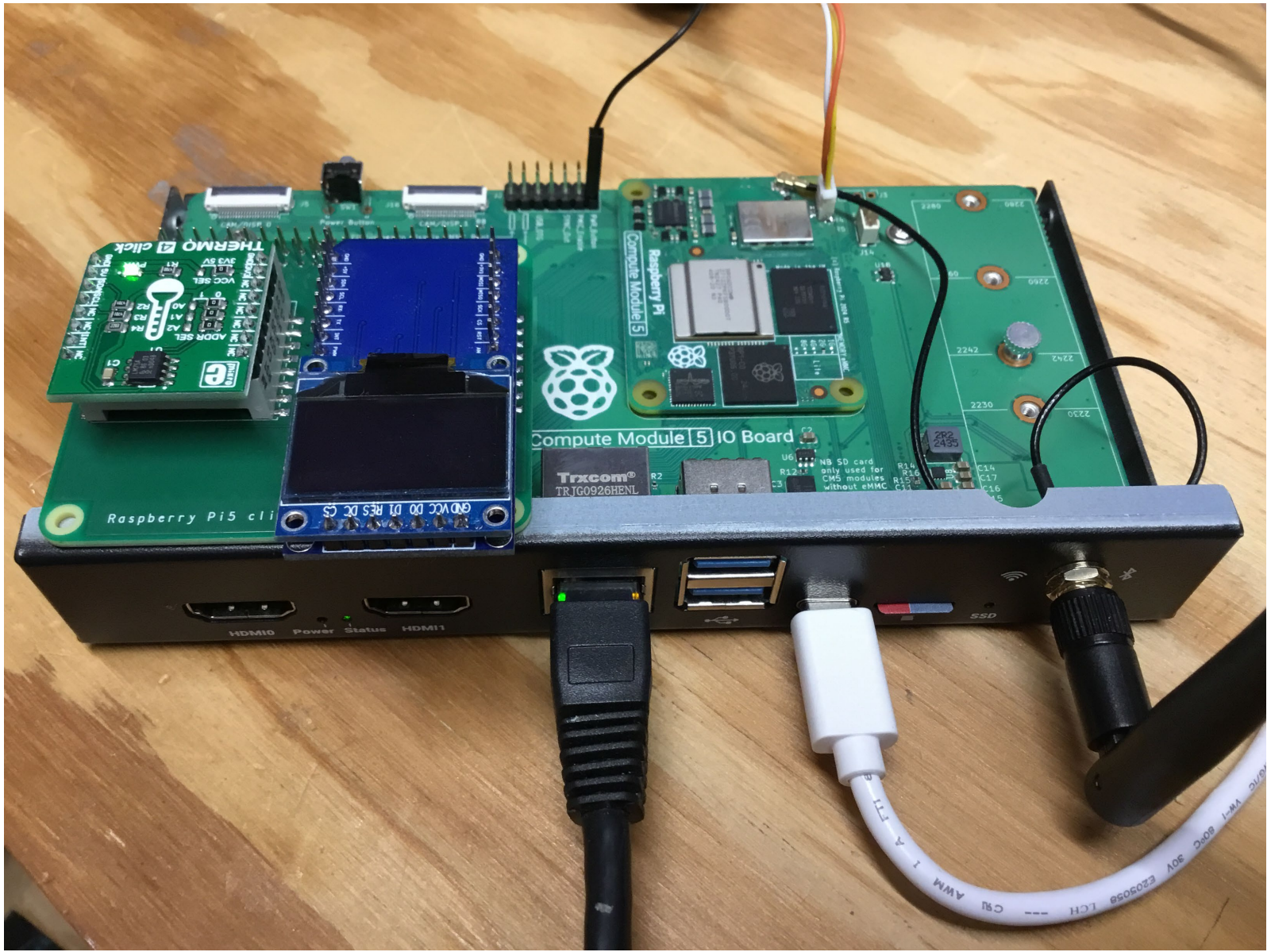
```c
int main(void)
{
    /* Example configuration - customize to suit your situation. */
    static const char *const chip_path = "/dev/gpiochip0";
    static const unsigned int line_offsets[NUM_LINES] = { 4, 17 };

    enum gpiod_line_value line_values[NUM_LINES] = { GPIOD_LINE_VALUE_ACTIVE,
                                                     GPIOD_LINE_VALUE_INACTIVE };
    struct gpiod_line_request *line_request;

    line_request = request_output_lines(chip_path, line_offsets, line_values,
                                        NUM_LINES,
                                        "CEC-toggle-multiple-lines");
    if (!line_request) {
        fprintf(stderr, "ERROR: Failed to request line: %s\n",
                strerror(errno));
        return EXIT_FAILURE;
    }

    for (;;) {
        print_line_values(line_offsets, line_values, NUM_LINES);
        sleep(.25);
        toggle_line_values(line_values, NUM_LINES);
        gpiod_line_request_set_values(line_request, line_values);
    }

    gpiod_line_request_release(line_request);

    return EXIT_SUCCESS;
}
```

Execute the Application Code

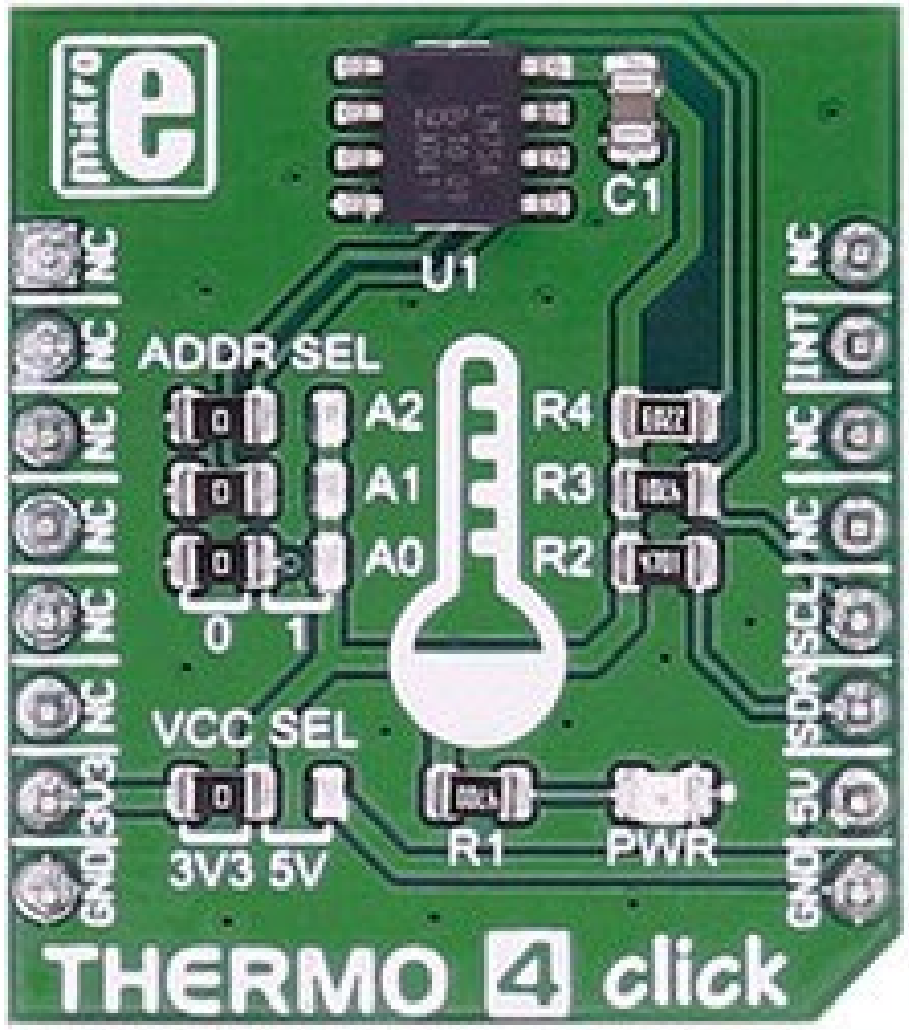
```
fred@pi5-yocto: ~
File  Edit  View  Search  Terminal  Help

root@raspberrypi5:~/cm5app# ./gpiodproject
4=Active 17=Inactive
4=Inactive 17=Active
4=Active 17=Inactive
4=Inactive 17=Active
4=Active 17=Inactive
4=Inactive 17=Active
4=Active 17=Inactive
4=Inactive 17=Active
4=Active 17=Inactive
4=Inactive 17=Active
4=Active 17=Inactive
4=Inactive 17=Active
4=Active 17=Inactive
```

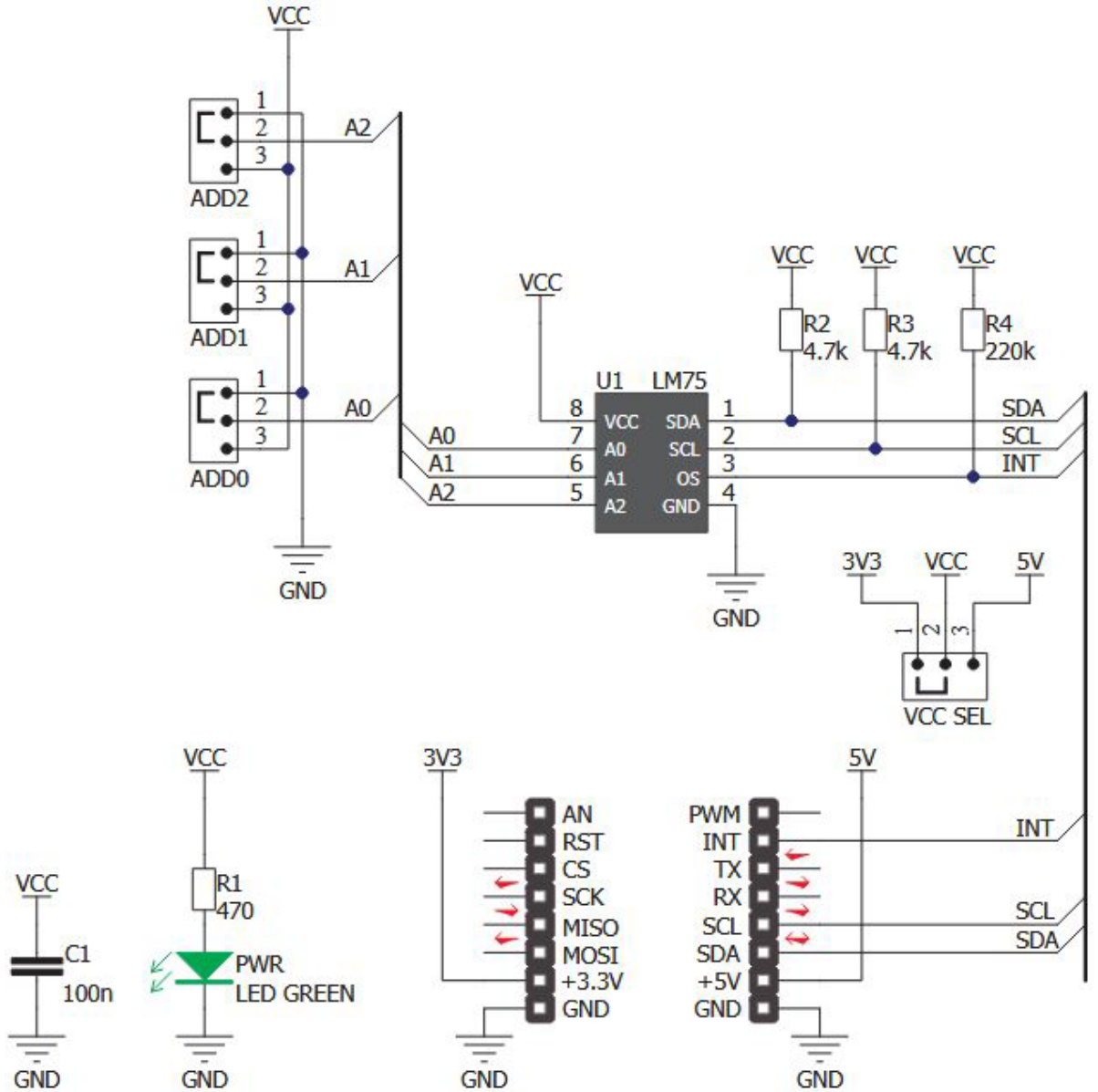


Assemble the Raspberry Pi Compute Module 5 Hardware

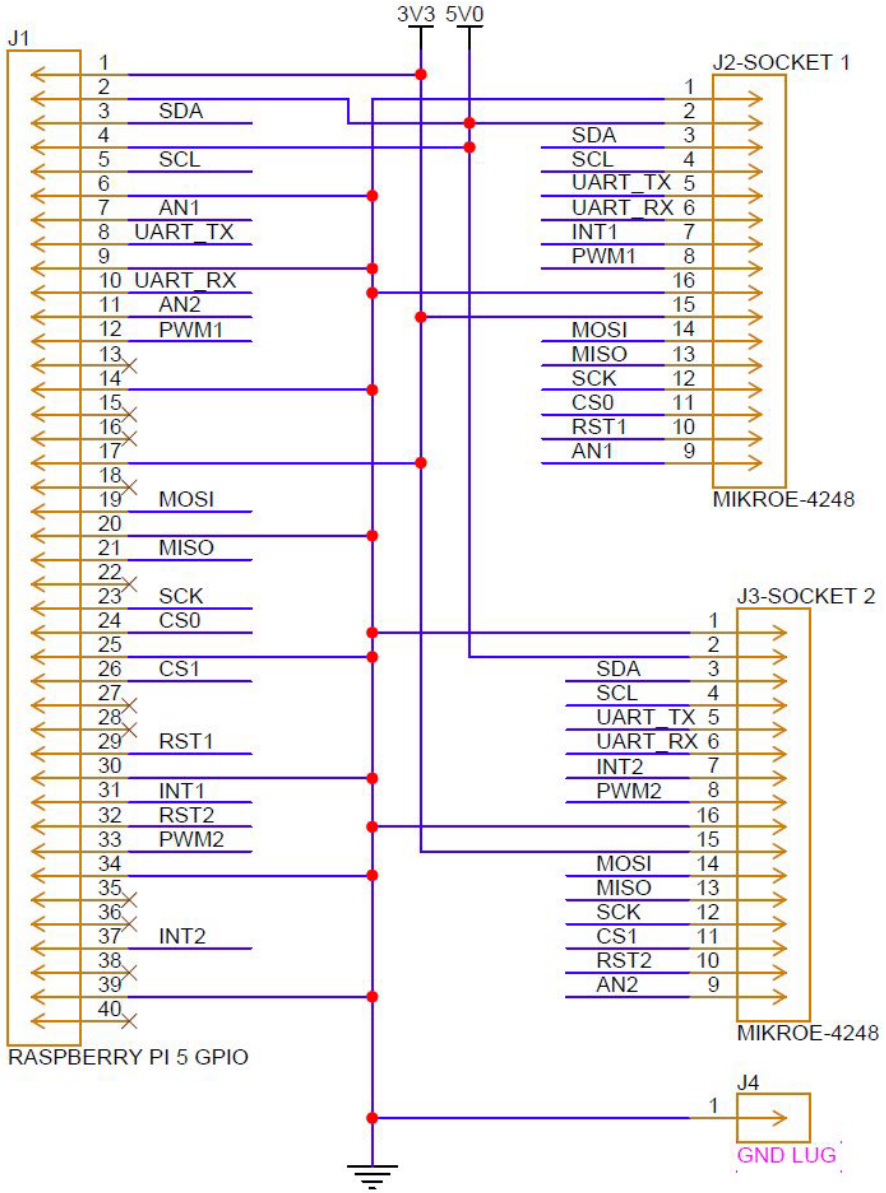


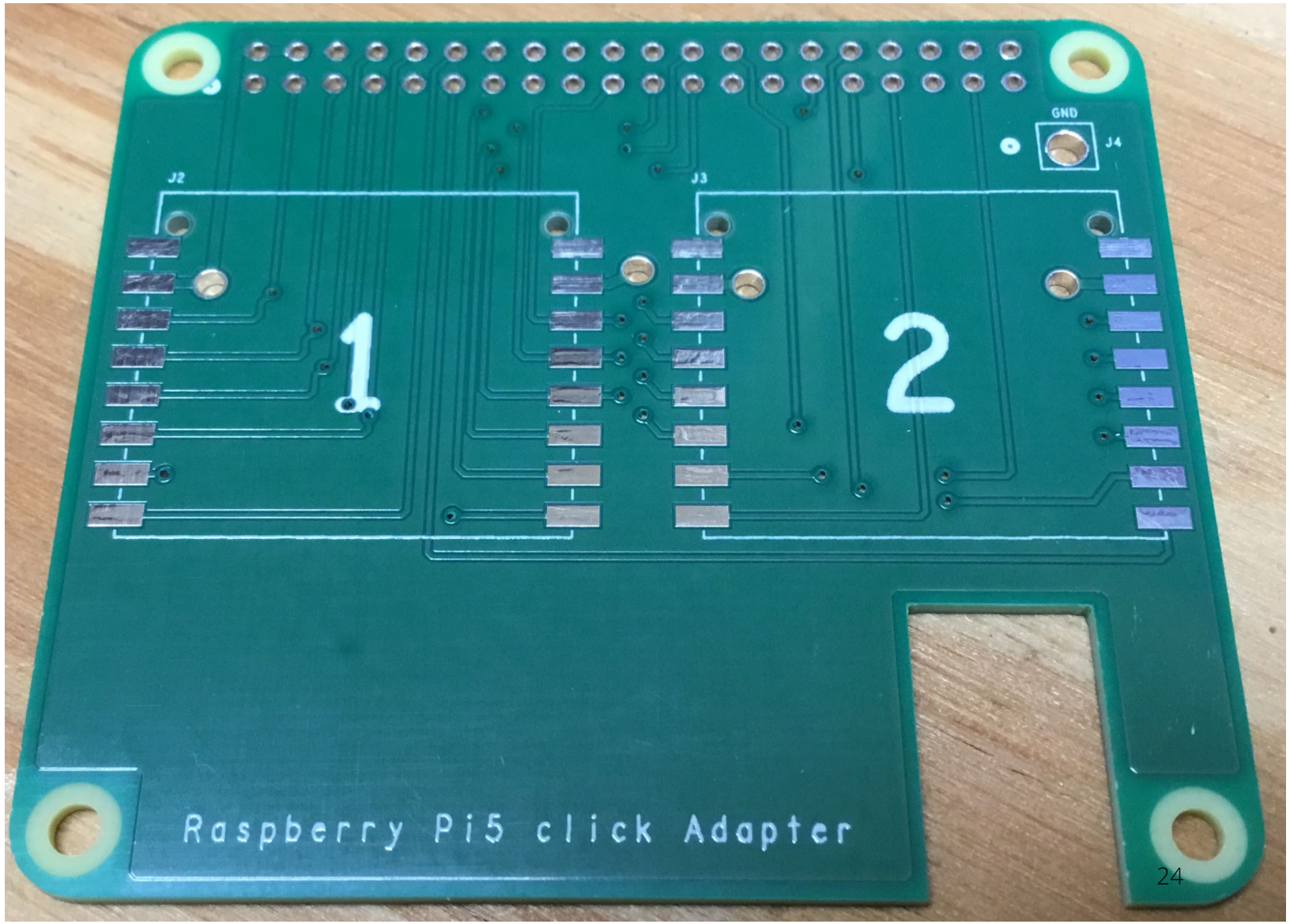
Assemble the Raspberry Pi 5 Hardware



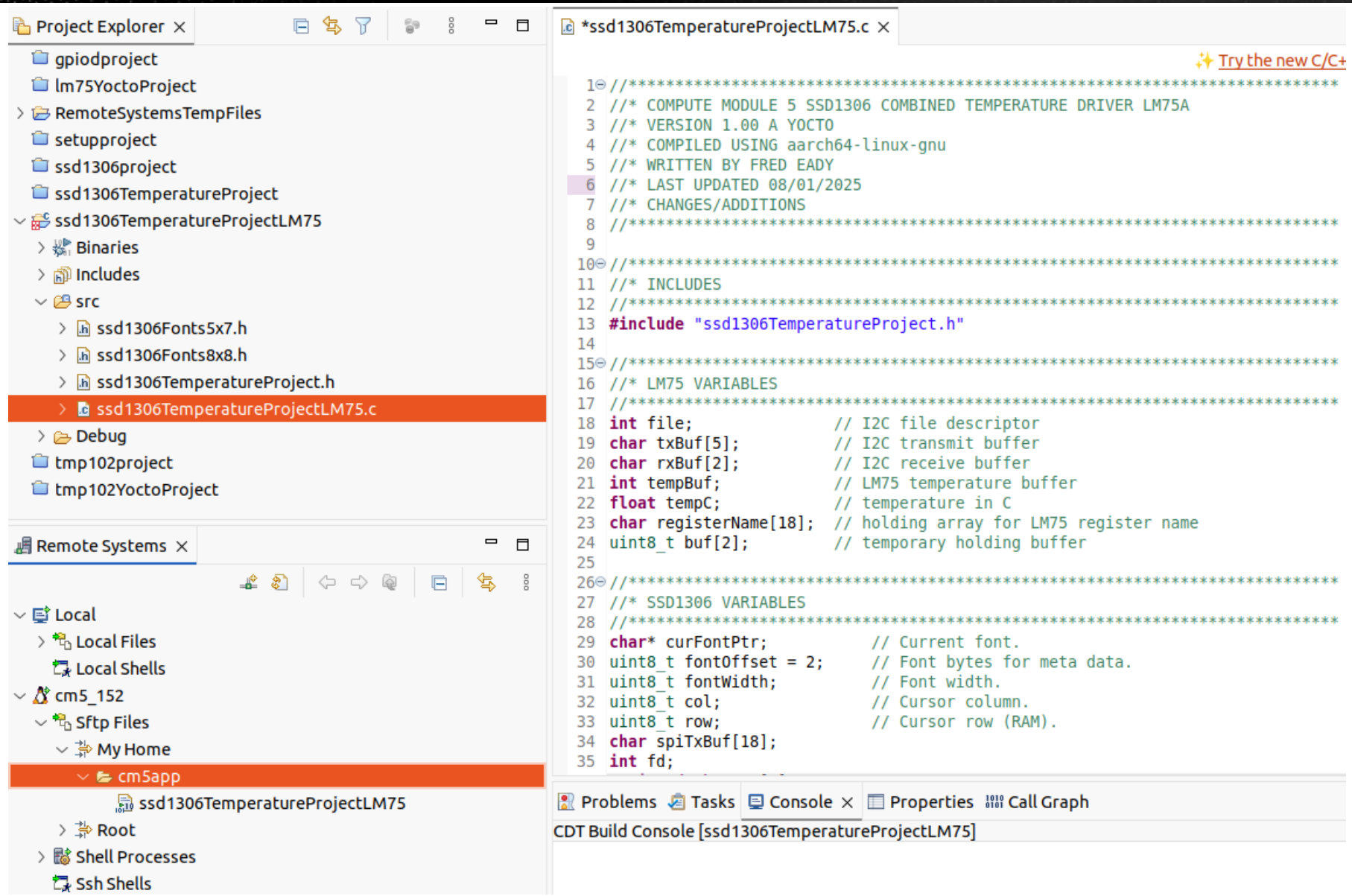


Assemble the Raspberry Pi 5 Hardware





Create and Compile the Application Code



Main Application Flow

```
//*****************************************************************************
// INITIALIZE THE SSD1306 DISPLAY
//*****************************************************************************
initSSD1306();

//*****************************************************************************
// Read the temperature
//*****************************************************************************
if(read(file,rxBuf,2) != 2)
{
    printf("Could not read from I2C device.\n");
    exit(1);
}
printf("INFO: 2 bytes were read-> rxBuf[0] 0x%02X -- rxBuf[1] 0x%02X\n",rxBuf[0],rxBuf[1]);

//*****************************************************************************
// Convert temperature bytes to 16-bit value
//*****************************************************************************
tempBuf = (rxBuf[0] << 4) | (rxBuf[1] >> 4);
printf("INFO: 16-bit value = 0x%04X\n",tempBuf);
```

Main Application Flow

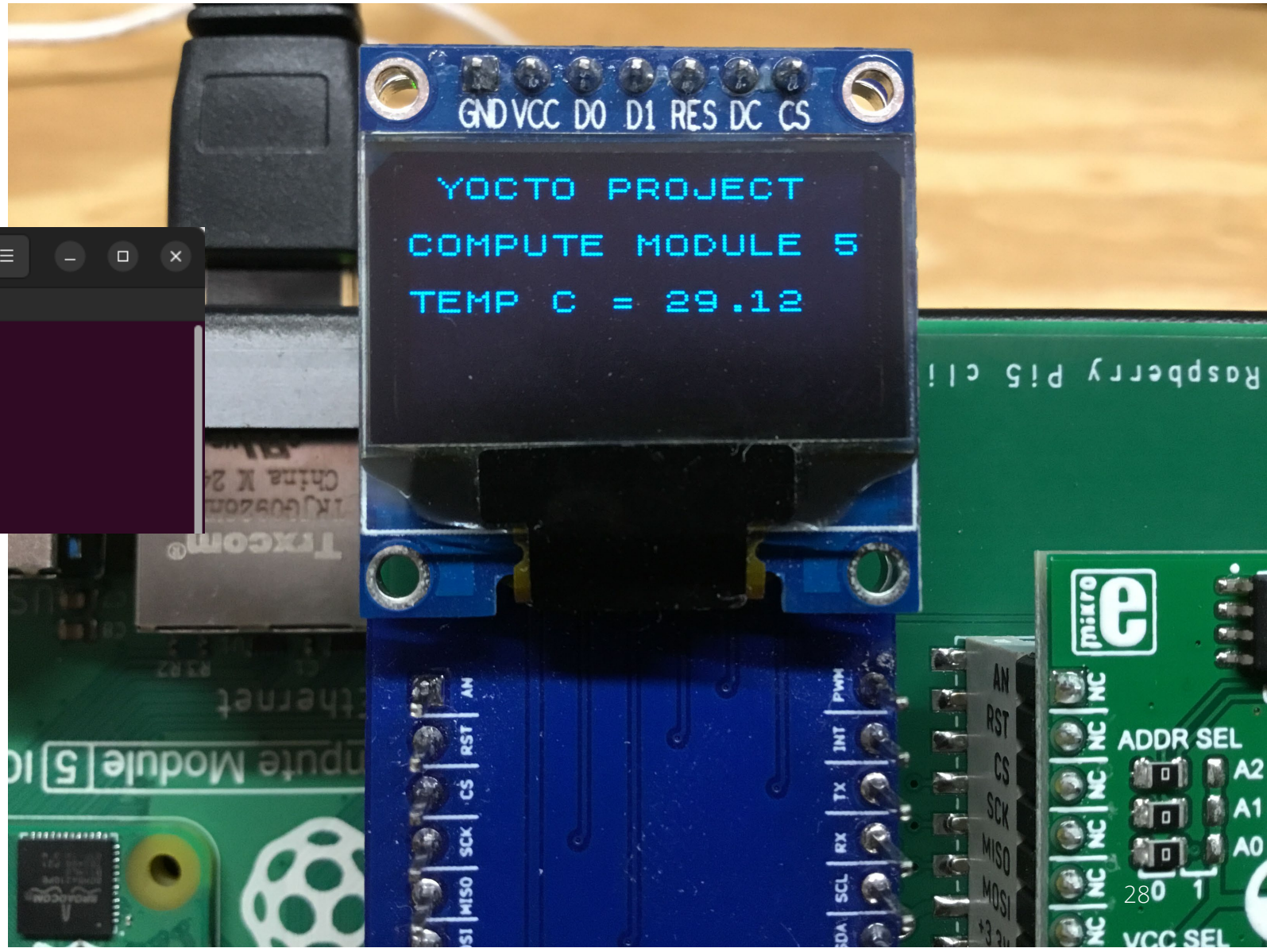
ssd1306TemperatureProjectLM75.c

ssd1306TemperatureProjectLM75 > src > ssd1306TemperatureProjectLM75.c

```
345     //*****************************************************************************
346     // Compute and print the temperature
347     //*****************************************************************************
348     buf[0] = (uint8_t)(tempBuf >> 5);
349     buf[0] &= 0x07;
350     buf[1] = (uint8_t)(tempBuf >> 8);
351
352     if(buf[1] > 0x80)
353     {
354         buf[1] = (~buf[1]) + 1;
355         tempC = (float)buf[1] + ((float)buf[0] * 0.125);
356         tempC *= -1.0;
357     }
358     else
359     {
360         tempC = (float)buf[1] + ((float)buf[0] * 0.125);
361     }
362
363     printf("INFO: Current Temperature -> %.2f deg C\n\n",tempC);
364
365     setTextXY(0,0);
366     sprintf(spiTxBuf," YOCTO PROJECT");
367     displayString(spiTxBuf);
368     setTextXY(2,0);
369     sprintf(spiTxBuf,"COMPUTE MODULE 5" );
370     displayString(spiTxBuf);
371     setTextXY(4,0);
372     sprintf(spiTxBuf,"TEMP C = %3.2F",tempC);
373     displayString(spiTxBuf);
374 }
```

Execute the Application Code

```
fred@pi5-yocto: ~
File  Edit  View  Search  Terminal  Help

root@raspberrypi5:~/cm5app# ./ssd1306TemperatureProjectLM75
INFO: /dev/i2c-1 opened for Read/Write.
INFO: Slave address changed to 0x48.
INFO: (file descriptor) fd = 4
INFO: 2 bytes were read-> rxBuf[0] 0x1D -- rxBuf[1] 0x20
INFO: 16-bit value = 0x1D20
INFO: Current Temperature -> 29.12 deg C
```



Thank you for attending!!!

Please consider the resources below:

- Today's Download Package
- digikey.com

LET'S EAT!

CEC Continuing Education Center

DesignNews

Thank You

Sponsored by

DigiKey

informa markets