CEC Continuing Education Center

DesignNews

Embedded Linux Primer

Day 4:

Generate a Custom Raspberry Pi 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

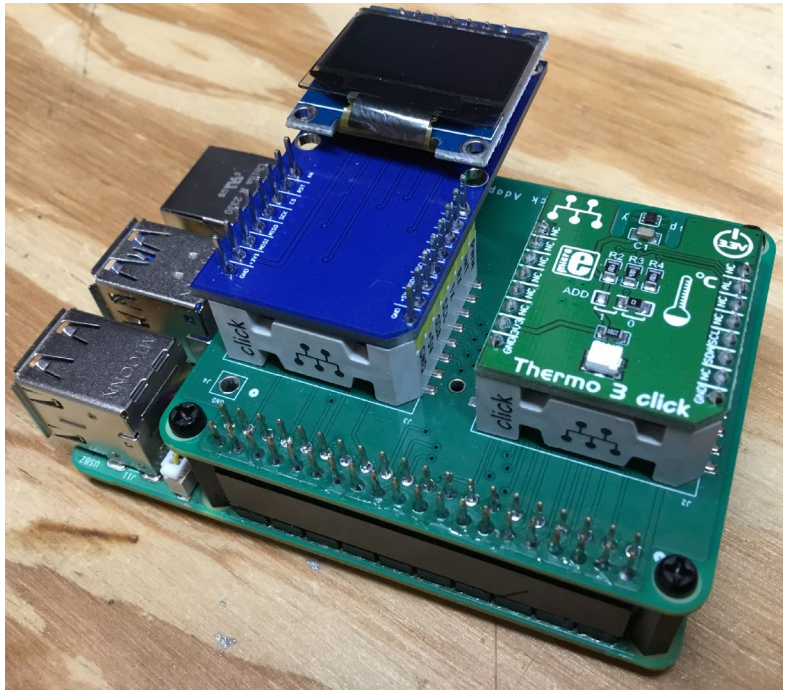
Visit ‘Lecturer Profile’ in your console for more details.

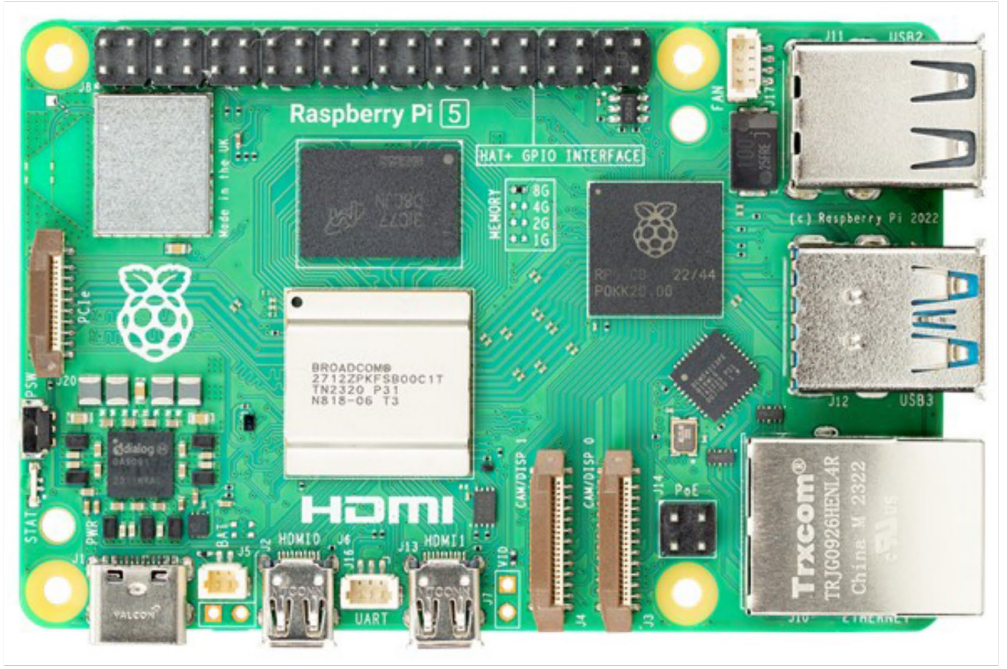
CEC Continuing Education Center

AGENDA

DigiKey

- Yocto RPI5 Image Configuration
- Cook Our Yocto RPI5 Image
- Load and Boot Our Yocto RPI5 Image
- Enable the RPI5 WiFi Radio
- Create a TMP102 Yocto Application





yocto PROJECT

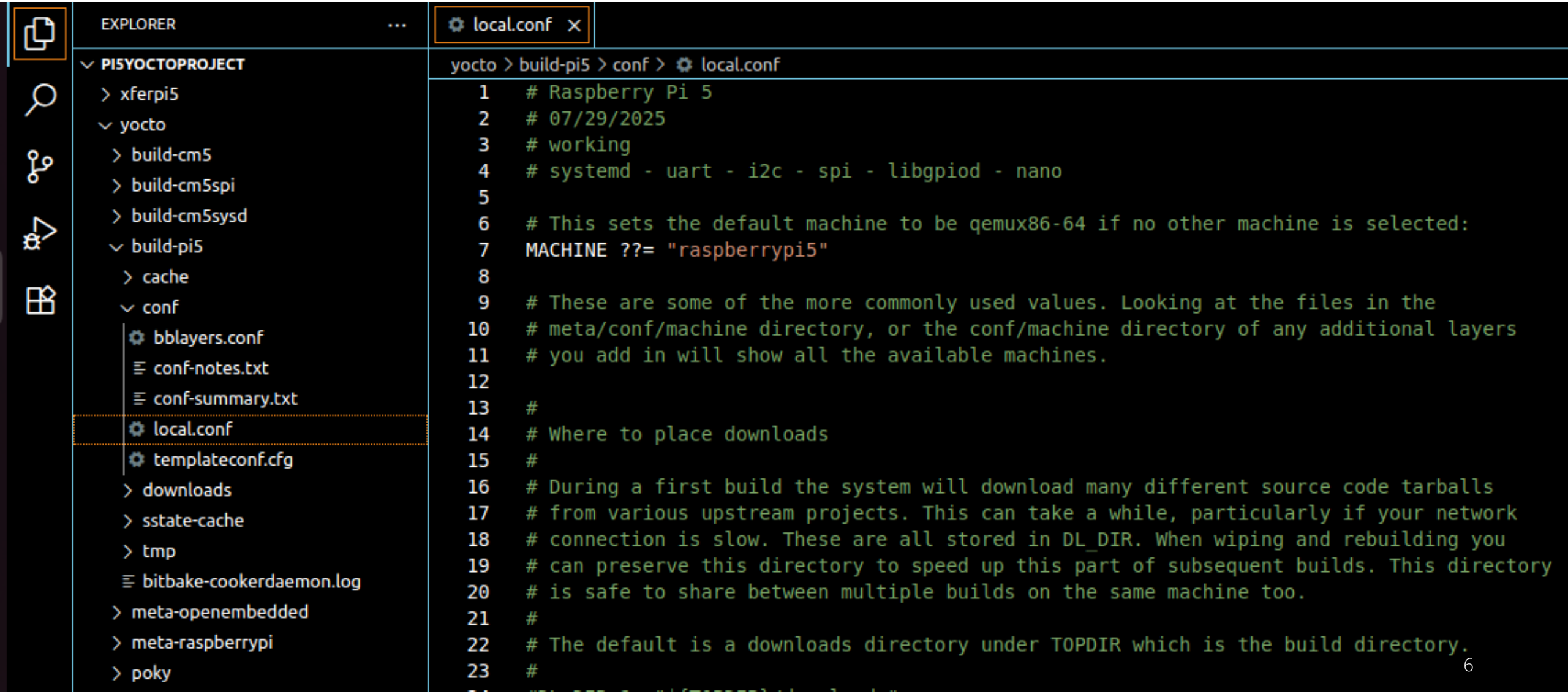
bblayers.conf

bblayers.conf

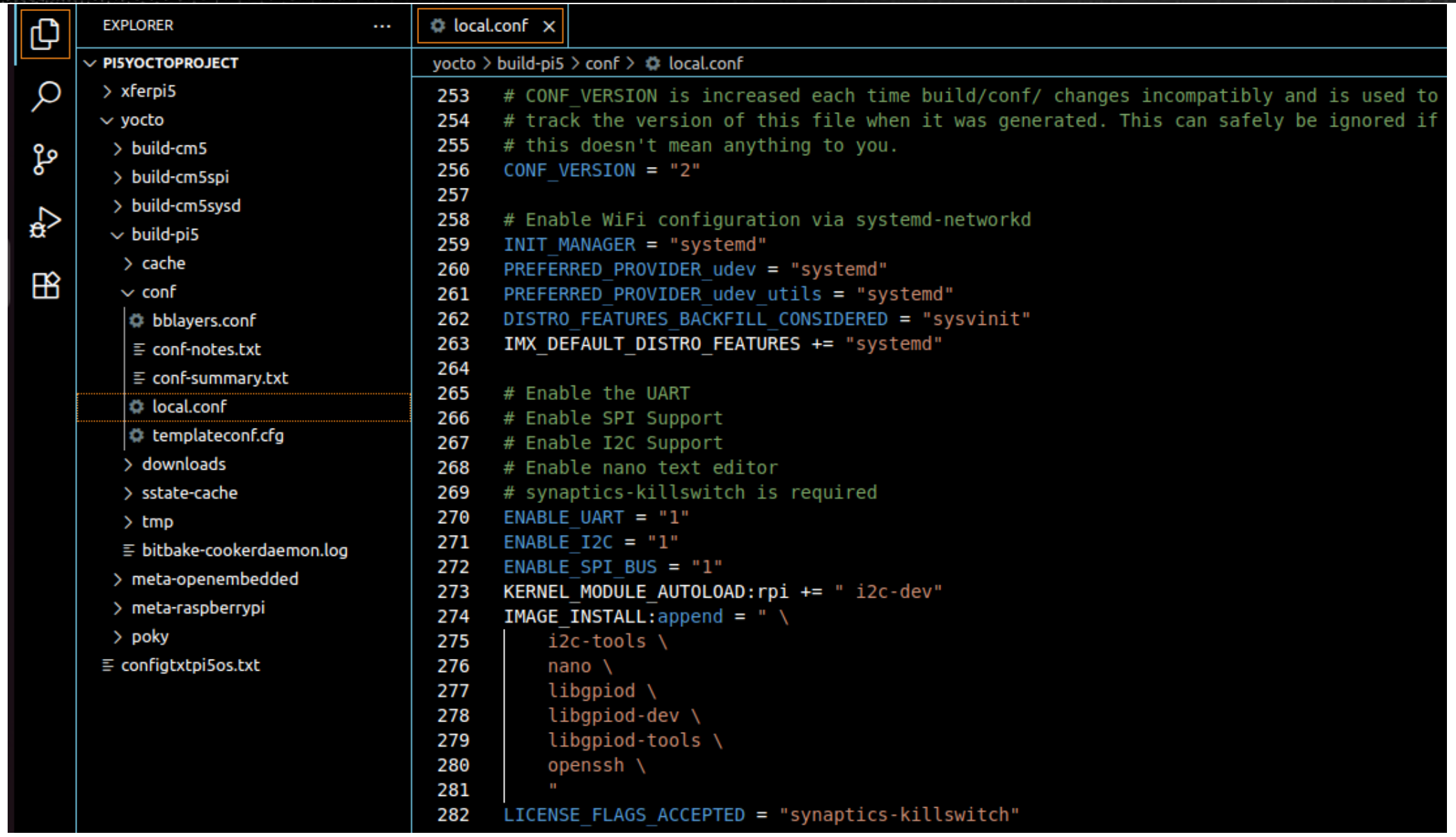
yocto > build-pi5 > conf > bblayers.conf

```
1   # RASPBERRY PI 5
2   # 07/30/2025
3   # working
4   # base build with nano - ssh - libgpiod - systemd - spi - i2c
5   # optional   /home/fred/pi5yoctoProject/yocto/meta-openembedded/meta-networking \
6   #
7   #
8   # POKY_BBLAYERS_CONF_VERSION is increased each time build/conf/bblayers.conf
9   # changes incompatibly
10  POKY_BBLAYERS_CONF_VERSION = "2"
11
12  BBPATH = "${TOPDIR}"
13  BBFILES ?= ""
14
15  BBLAYERS ?= " \
16    /home/fred/pi5yoctoProject/yocto/poky/meta \
17    /home/fred/pi5yoctoProject/yocto/poky/meta-poky \
18    /home/fred/pi5yoctoProject/yocto/poky/meta-yocto-bsp \
19    /home/fred/pi5yoctoProject/yocto/meta-openembedded/meta-oe \
20    /home/fred/pi5yoctoProject/yocto/meta-openembedded/meta-python \
21    /home/fred/pi5yoctoProject/yocto/meta-raspberrypi \
22    "
```

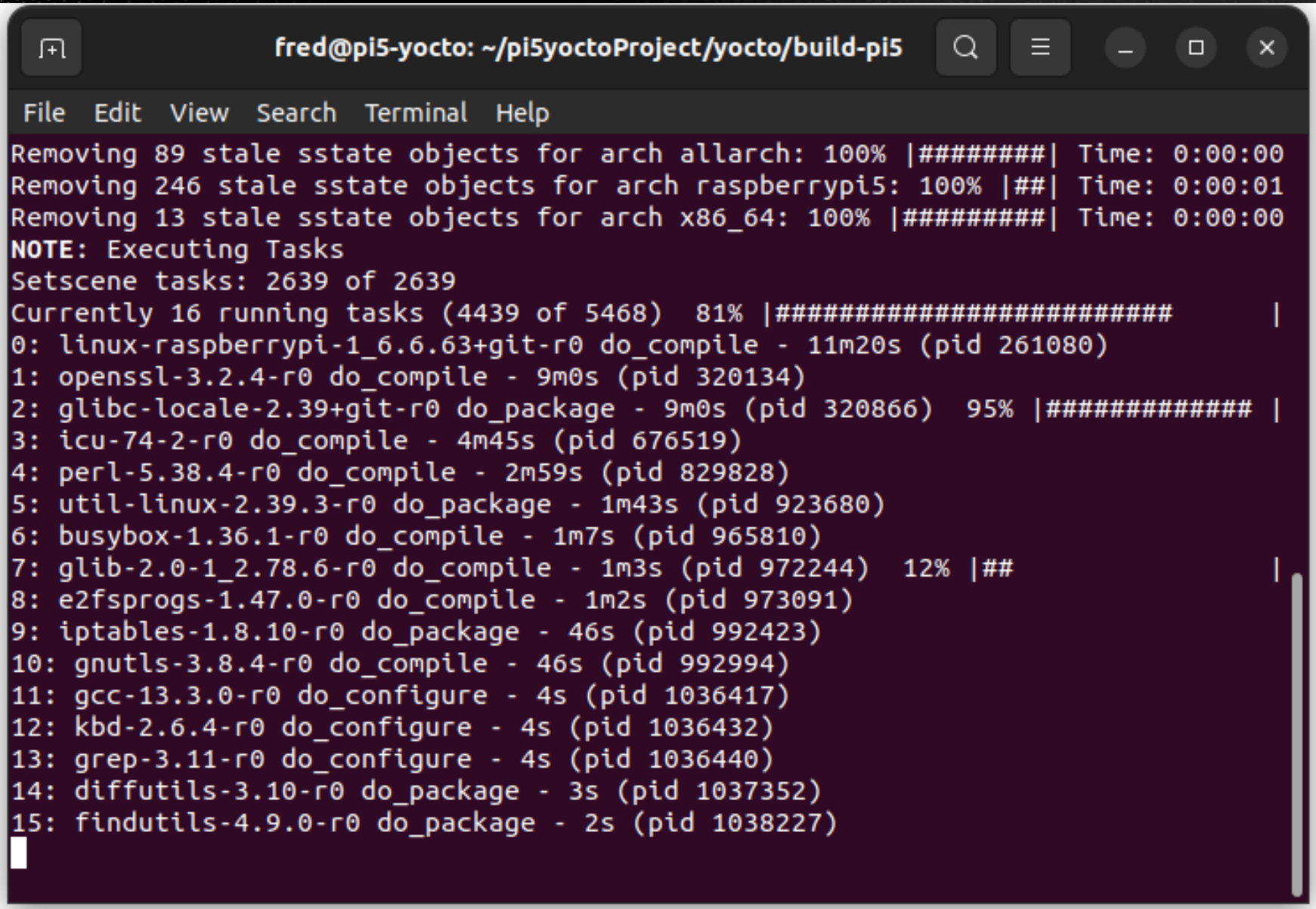
local.conf


EXPLORER
local.conf
PI5YOCTOPROJECT
xferpi5
yocto
build-cm5
build-cm5spi
build-cm5sysd
build-pi5
cache
conf
bblayers.conf
conf-notes.txt
conf-summary.txt
local.conf
templateconf.cfg
downloads
sstate-cache
tmp
bitbake-cookerdaemon.log
meta-openembedded
meta-raspberrypi
poky
yocto > build-pi5 > conf > local.conf
1 # Raspberry Pi 5
2 # 07/29/2025
3 # working
4 # systemd - uart - i2c - spi - libgpiod - nano
5
6 # This sets the default machine to be qemux86-64 if no other machine is selected:
7 MACHINE ??= "raspberrypi5"
8
9 # These are some of the more commonly used values. Looking at the files in the
10 # meta/conf/machine directory, or the conf/machine directory of any additional layers
11 # you add in will show all the available machines.
12
13 #
14 # Where to place downloads
15 #
16 # During a first build the system will download many different source code tarballs
17 # from various upstream projects. This can take a while, particularly if your network
18 # connection is slow. These are all stored in DL_DIR. When wiping and rebuilding you
19 # can preserve this directory to speed up this part of subsequent builds. This directory
20 # is safe to share between multiple builds on the same machine too.
21 #
22 # The default is a downloads directory under TOPDIR which is the build directory.
23 #

local.conf


EXPLORER
local.conf
PI5YOCTOPROJECT
xferpi5
yocto
build-cm5
build-cm5spi
build-cm5sysd
build-pi5
cache
conf
bblayers.conf
conf-notes.txt
conf-summary.txt
local.conf
templateconf.cfg
downloads
sstate-cache
tmp
bitbake-cookerdaemon.log
meta-openembedded
meta-raspberrypi
poky
configtxtpi5os.txt
yocto > build-pi5 > conf > local.conf
253 # CONF_VERSION is increased each time build/conf/ changes incompatibly and is used to
254 # track the version of this file when it was generated. This can safely be ignored if
255 # this doesn't mean anything to you.
256 CONF_VERSION = "2"
257
258 # Enable WiFi configuration via systemd-networkd
259 INIT_MANAGER = "systemd"
260 PREFERRED_PROVIDER_udev = "systemd"
261 PREFERRED_PROVIDER_udev_utils = "systemd"
262 DISTRO_FEATURES_BACKFILL_CONSIDERED = "sysvinit"
263 IMX_DEFAULT_DISTRO_FEATURES += "systemd"
264
265 # Enable the UART
266 # Enable SPI Support
267 # Enable I2C Support
268 # Enable nano text editor
269 # synaptics-killswitch is required
270 ENABLE_UART = "1"
271 ENABLE_I2C = "1"
272 ENABLE_SPI_BUS = "1"
273 KERNEL_MODULE_AUTOLOAD:rpi += " i2c-dev"
274 IMAGE_INSTALL:append = " \
275 i2c-tools \
276 nano \
277 libgpiod \
278 libgpiod-dev \
279 libgpiod-tools \
280 openssh \
281 "
282 LICENSE_FLAGS_ACCEPTED = "synaptics-killswitch"

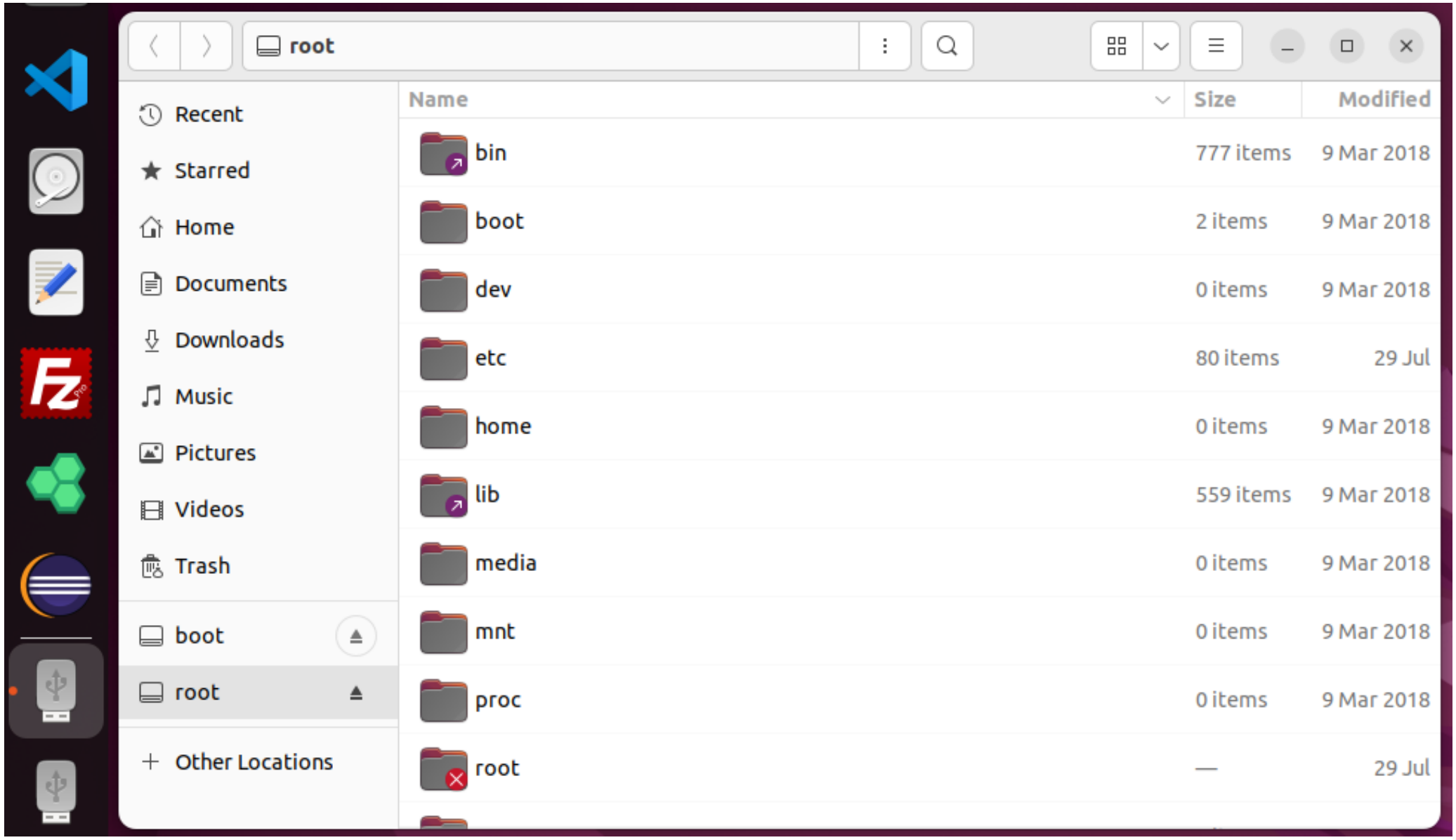
Invoke BitBake



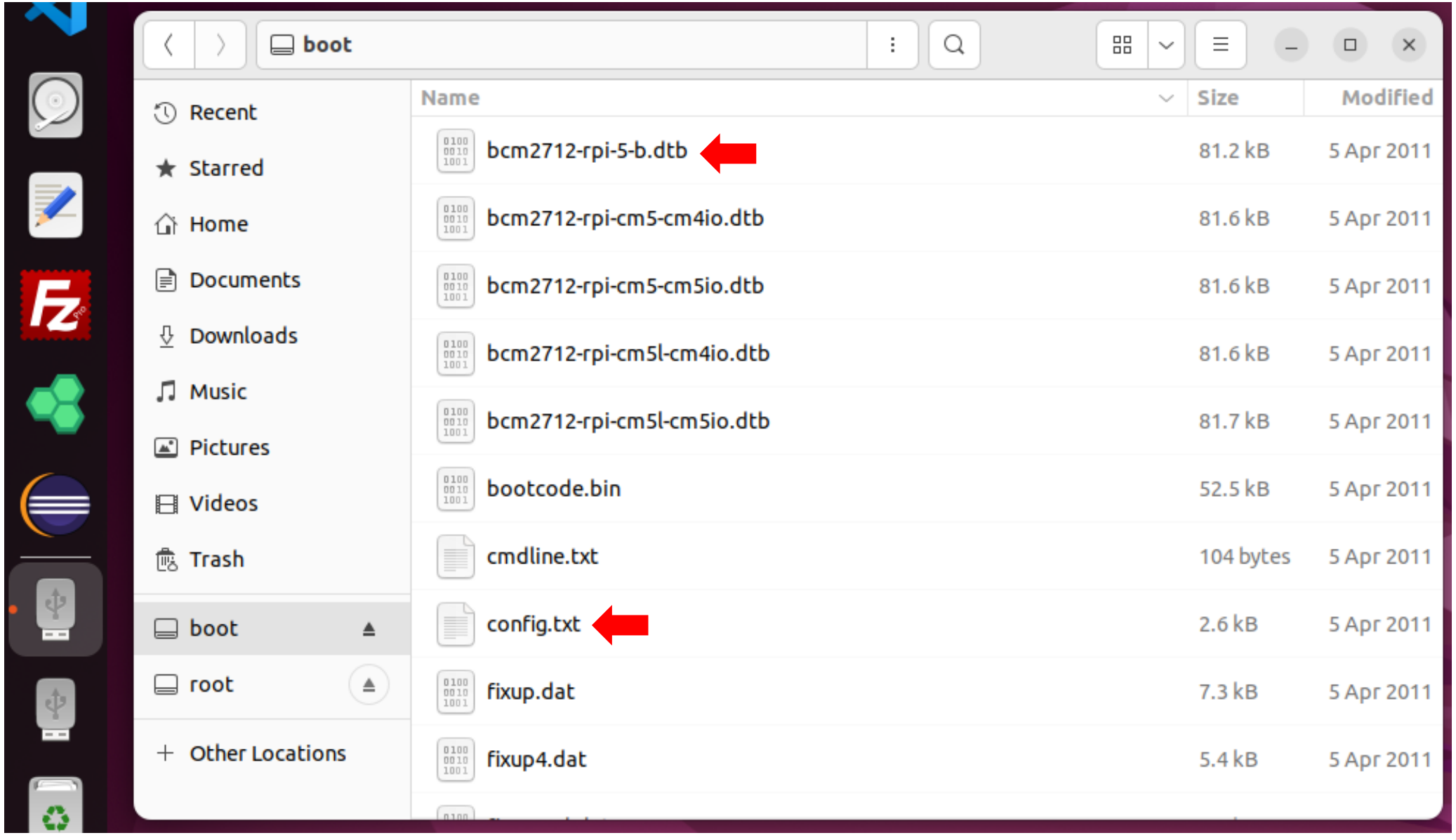
Load the microSD Card With Our New Base Image

```
fred@pi5-yocto: ~/pi5yoctoProject/yocto/build-pi5/tmp/deploy/images/raspberrypi5
File  Edit  View  Search  Terminal  Help
meta-poky
meta-yocto-bsp       = "scarthgap:ac257900c33754957b2696529682029d997a8f28"
meta-oe
meta-python          = "scarthgap:491671faee11ea131feab5a3a451d1a01deb2ab1"
meta-raspberrypi     = "scarthgap:bce7b3acd2e0d9d127fcb57eae4a512bd7e7924a"

Sstate summary: Wanted 1763 Local 0 Mirrors 0 Missed 1763 Current 876 (0% match, 33% complete)
Removing 1334 stale sstate objects for arch cortexa76: 100% |####| Time: 0:00:02
Removing 89 stale sstate objects for arch allarch: 100% |########| Time: 0:00:00
Removing 246 stale sstate objects for arch raspberrypi5: 100% |##| Time: 0:00:01
Removing 13 stale sstate objects for arch x86_64: 100% |#########| Time: 0:00:00
NOTE: Executing Tasks
NOTE: Tasks Summary: Attempted 5468 tasks of which 2072 didn't need to be rerun and all succeeded.
fred@pi5-yocto:~/pi5yoctoProject/yocto/build-pi5$ cd tmp/deploy/images/raspberrypi5/
fred@pi5-yocto:~/pi5yoctoProject/yocto/build-pi5/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'
fred@pi5-yocto:~/pi5yoctoProject/yocto/build-pi5/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
```

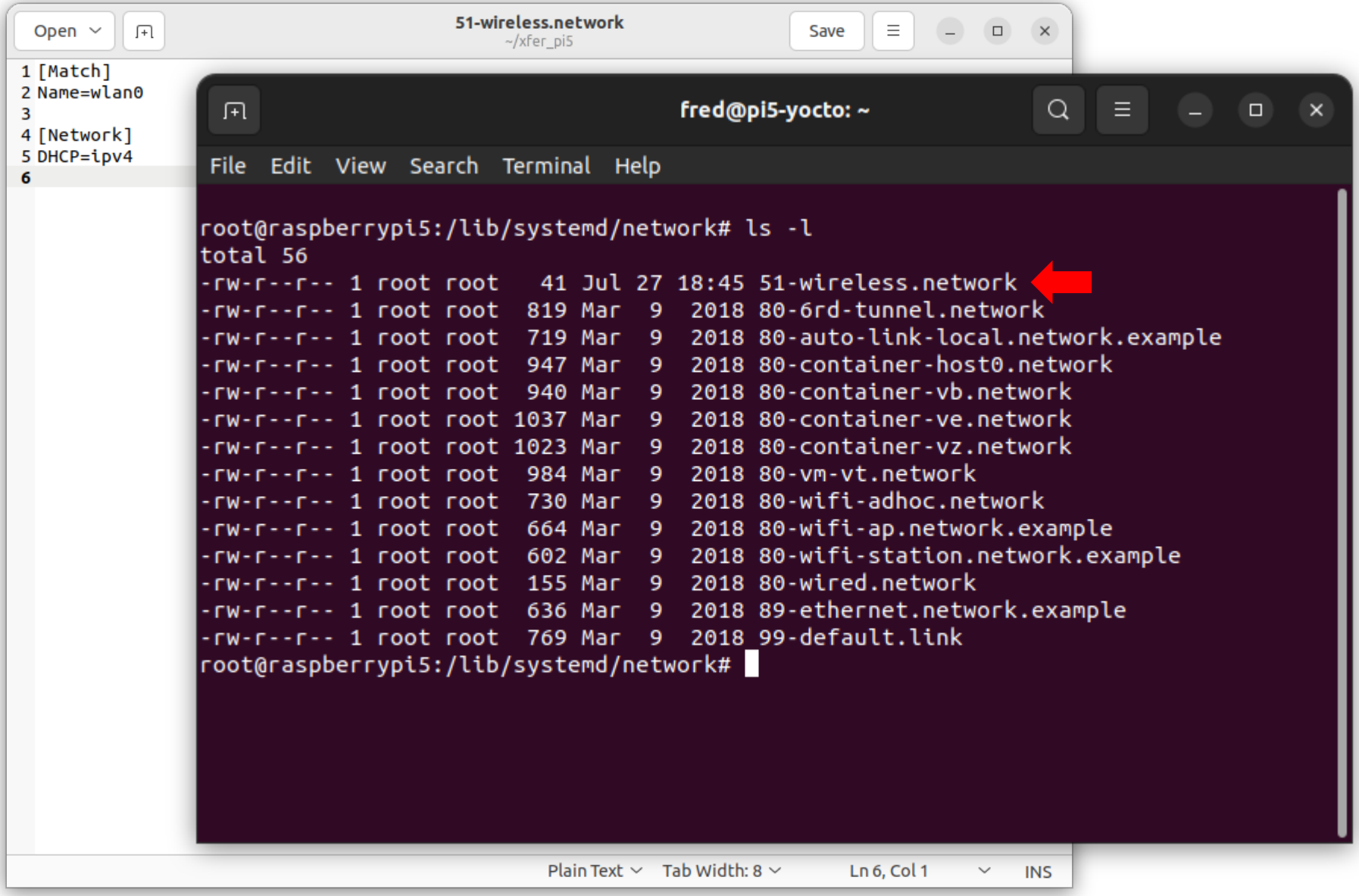
Plain Text | Tab Width: 8 | Ln 1, Col 10 | INS

local.conf

yocto > build-pi5 > conf > local.conf

```
266 # Enable SPI Support
267 # Enable I2C Support
268 # Enable nano text editor
269 # synaptics-killswitch is required
270 ENABLE_UART = "1"
271 ENABLE_I2C = "1"
272 ENABLE_SPI_BUS = "1"
273 KERNEL_MODULE_AUTOLOAD:rpi += " i2c-dev"
274 IMAGE_INSTALL:append = " \
275     i2c-tools \
276     nano \
277     libgpiod \
278     libgpiod-dev \
279     libgpiod-tools \
280     openssh \
281     "
```

Create and Place 51-wireless.network



Create and Place wpa_supplicant-wlan0.conf

*wpa_supplicant-wlan0.conf
~/xfer_pi5

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

*wpa_supplicant-wlan0.conf
~/xfer_pi5

```
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

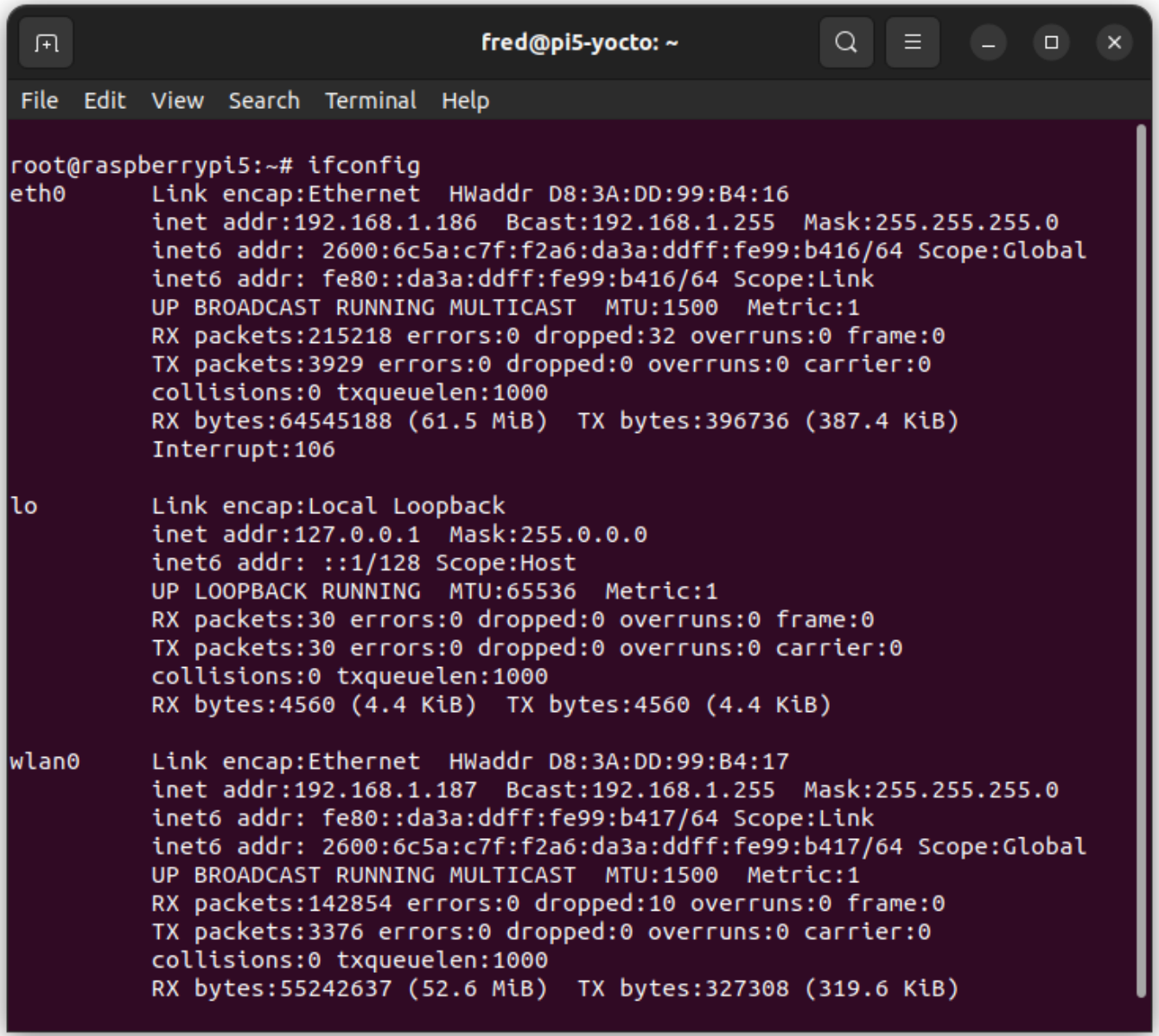
fred@pi5-yocto: ~

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

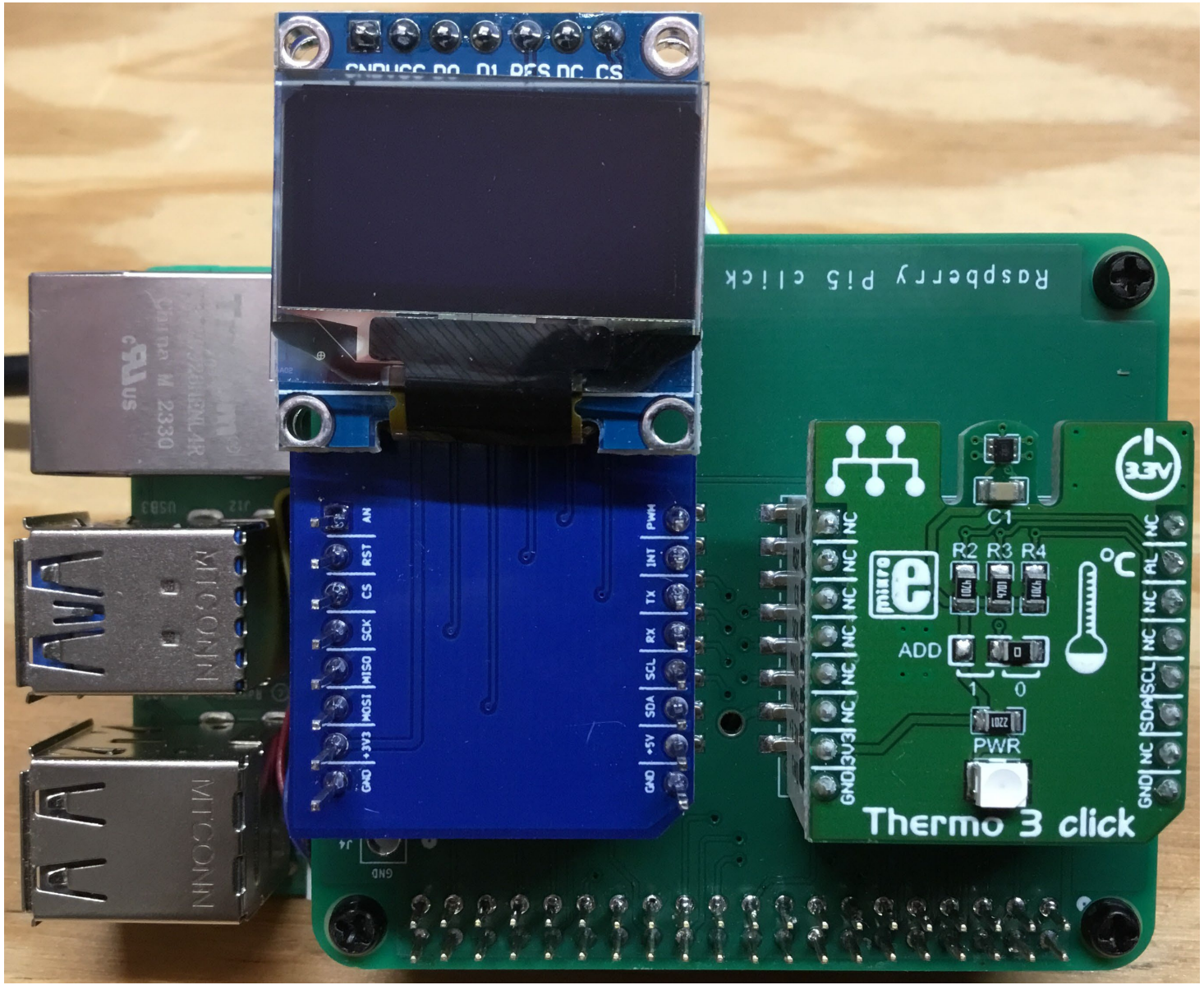
Enable WiFi



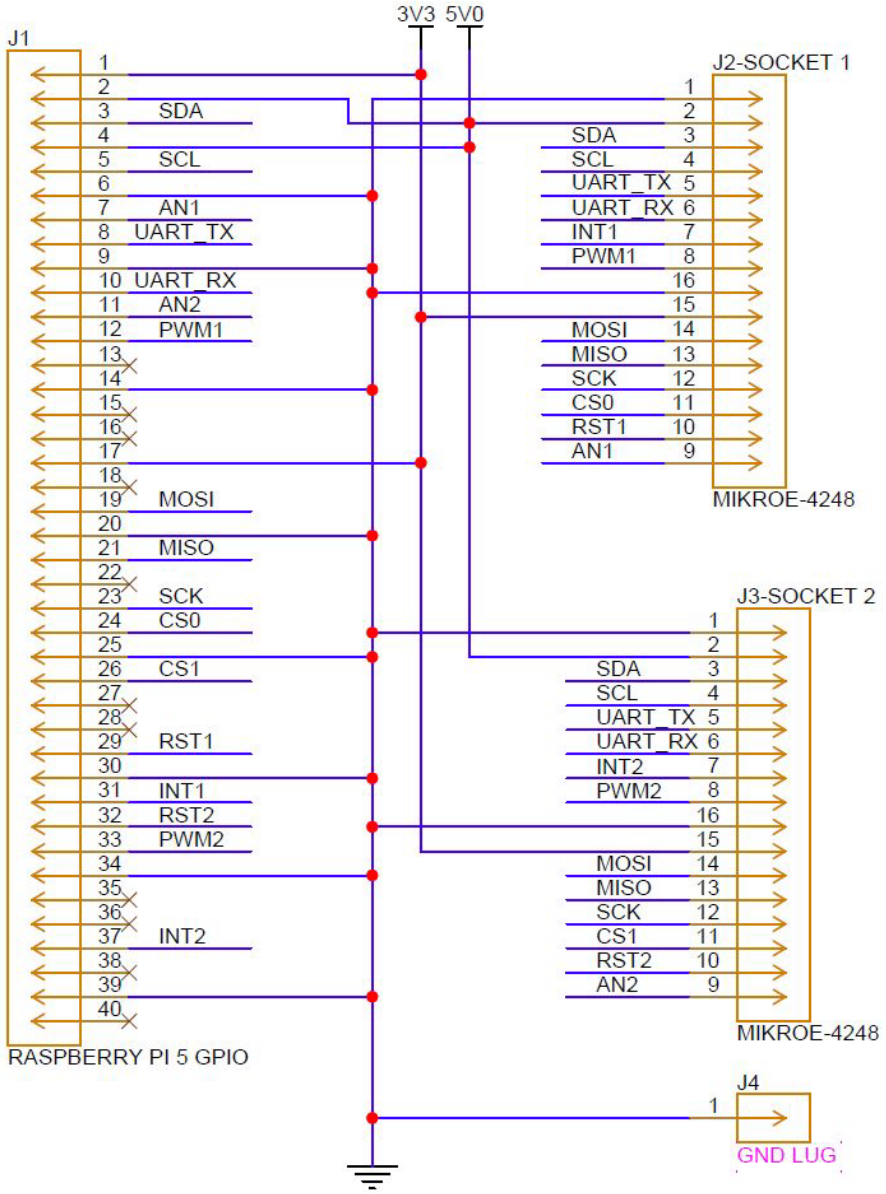
To enable WiFi , execute the following 3 commands from the Raspberry Pi 5 Linux console :

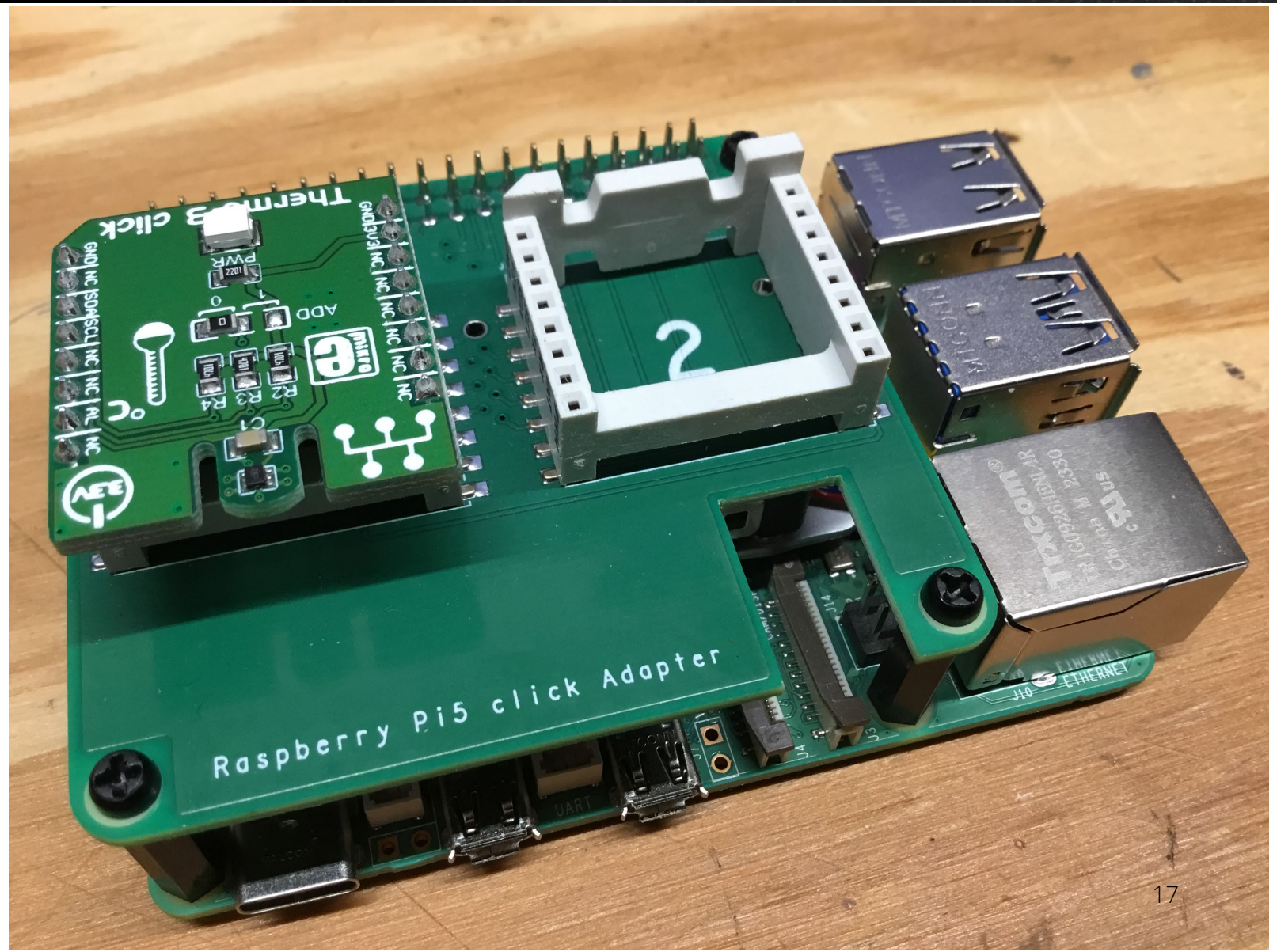
```
systemctl enable wpa_supplicant@wlan0.service
systemctl restart systemd-networkd.service
systemctl restart wpa_supplicant@wlan0.service
```

Assemble the Raspberry Pi 5 Hardware

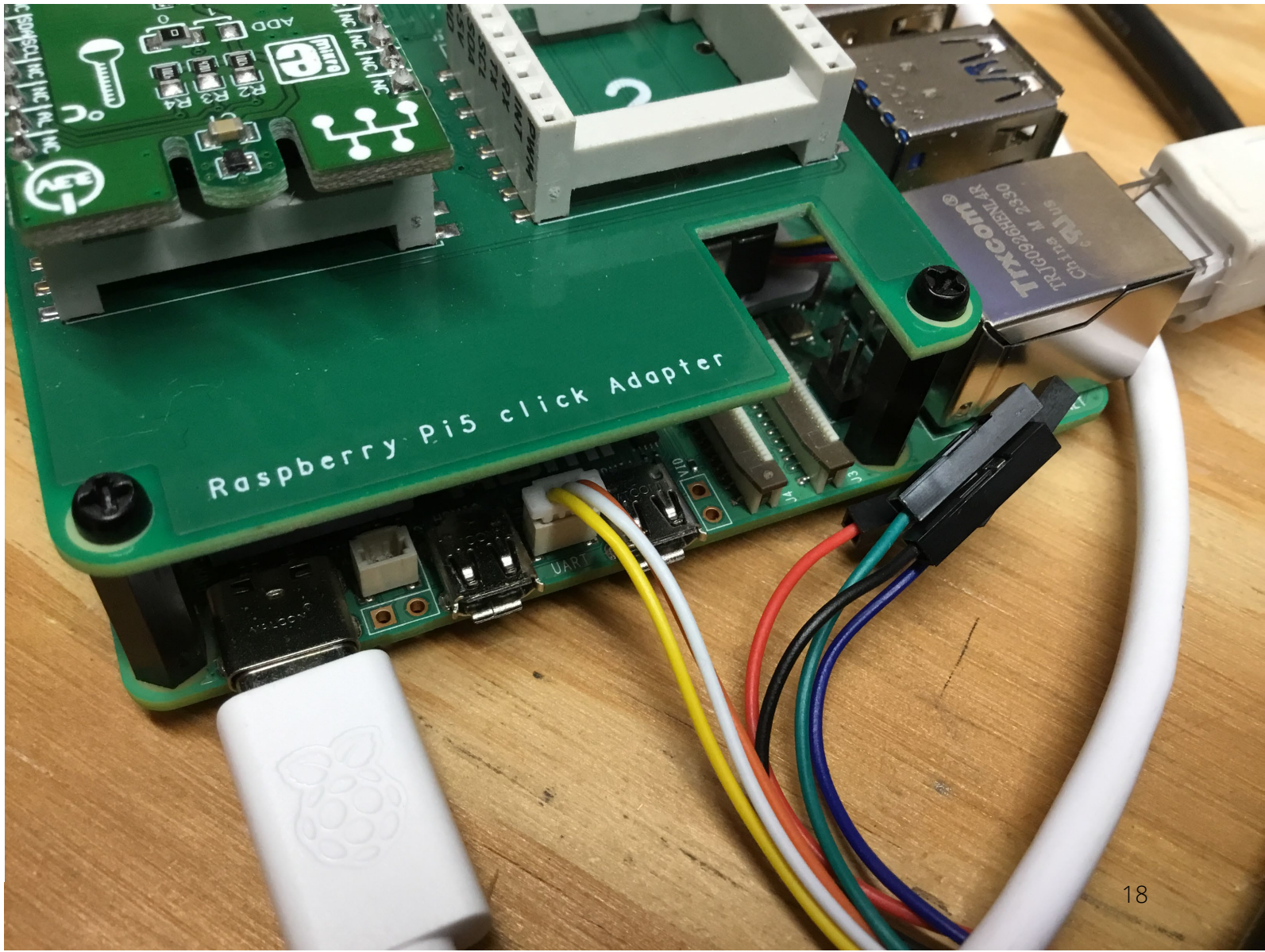


Assemble the Raspberry Pi 5 Hardware

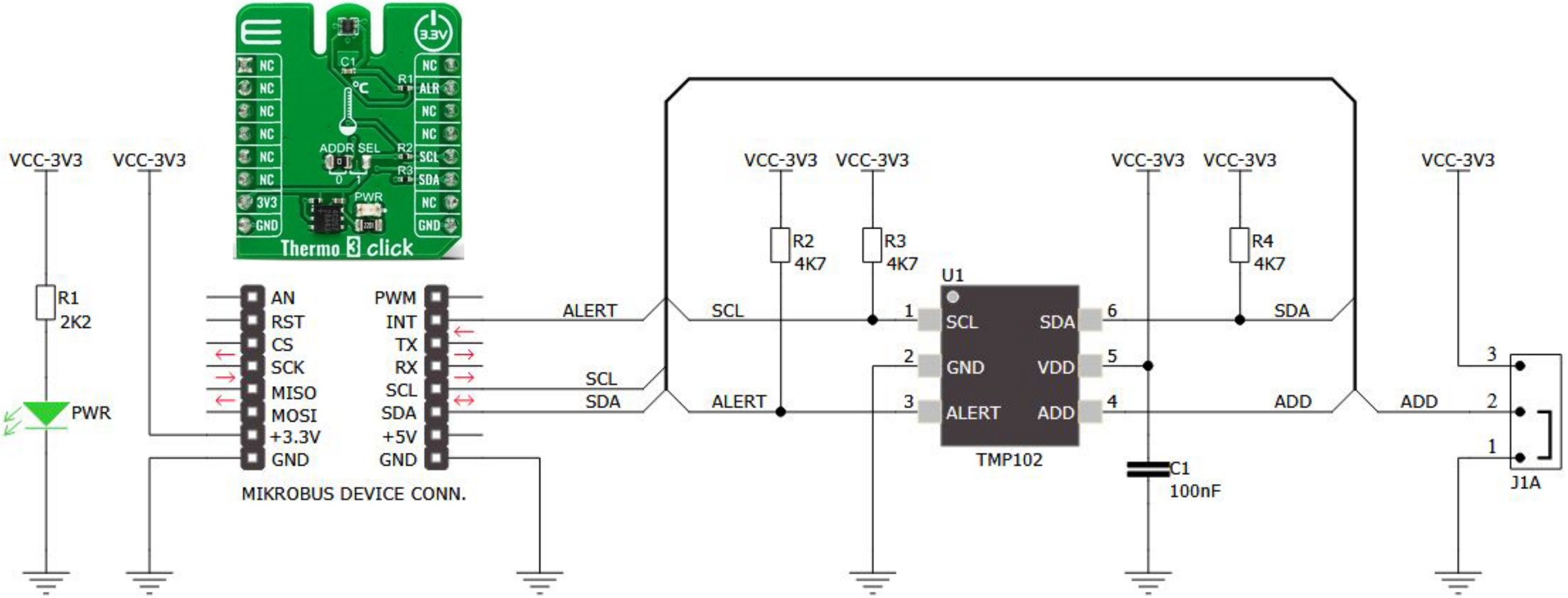




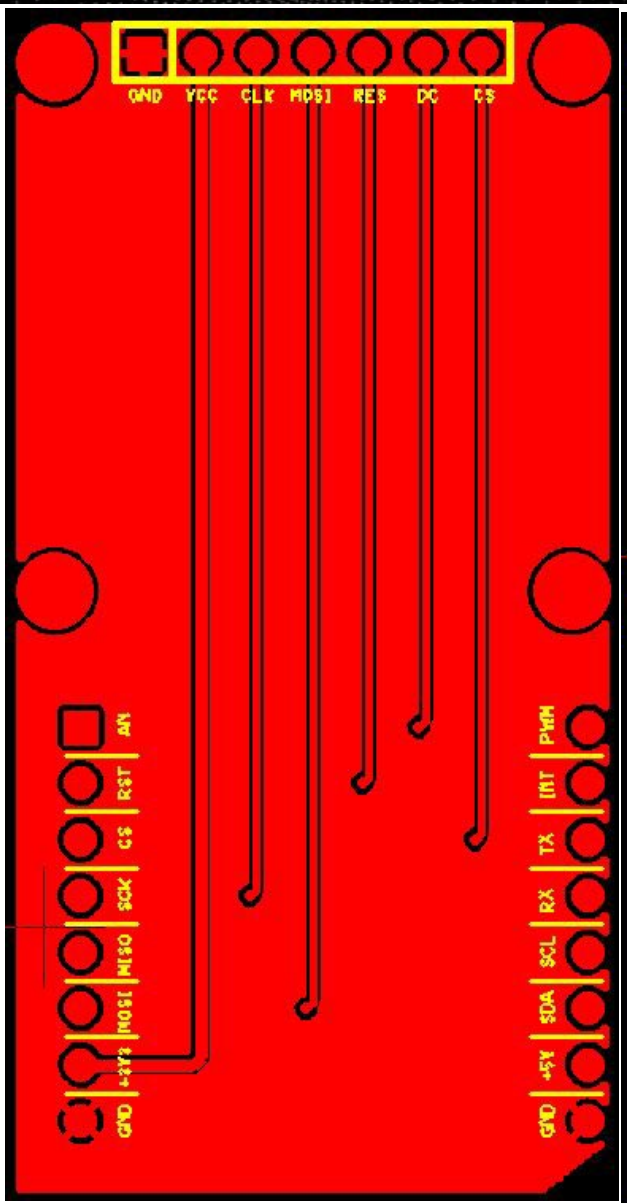
Assemble the Raspberry Pi 5 Hardware

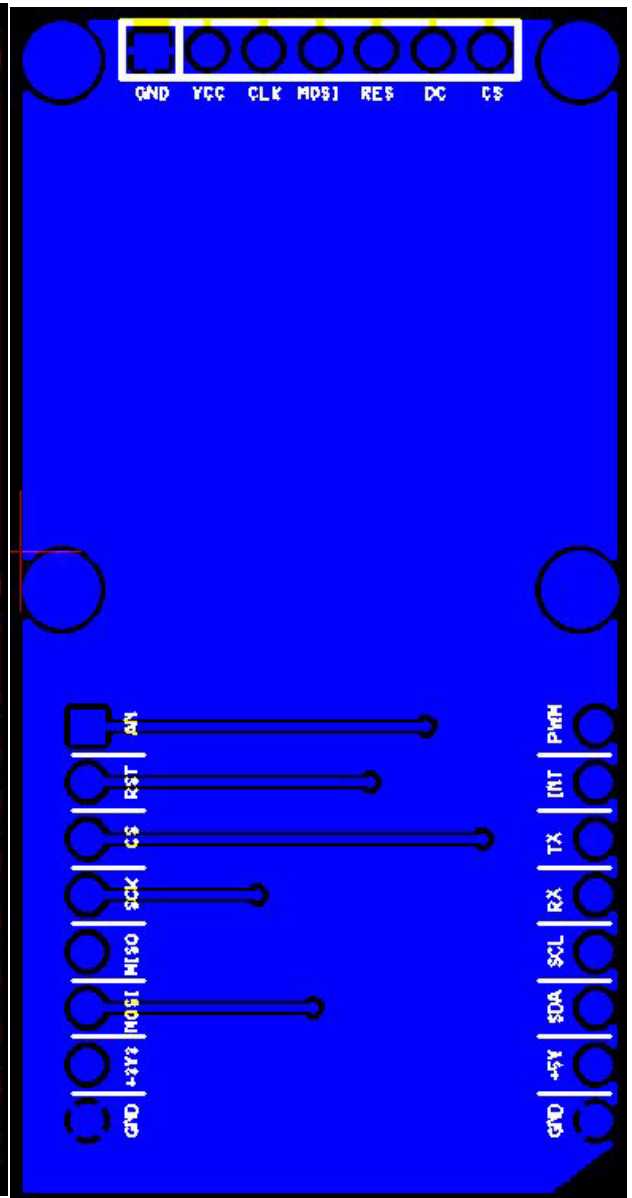


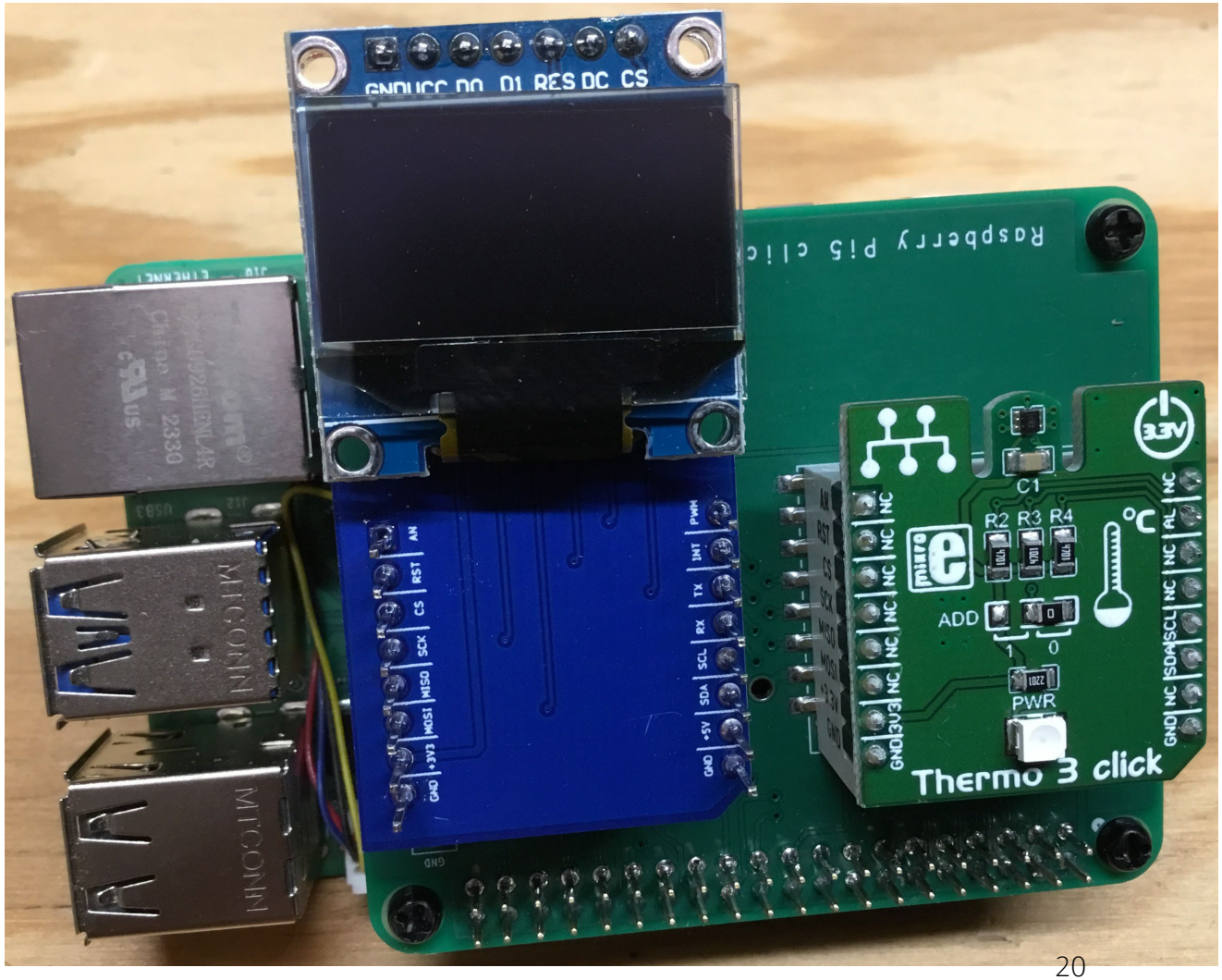
Assemble the Raspberry Pi 5 Hardware



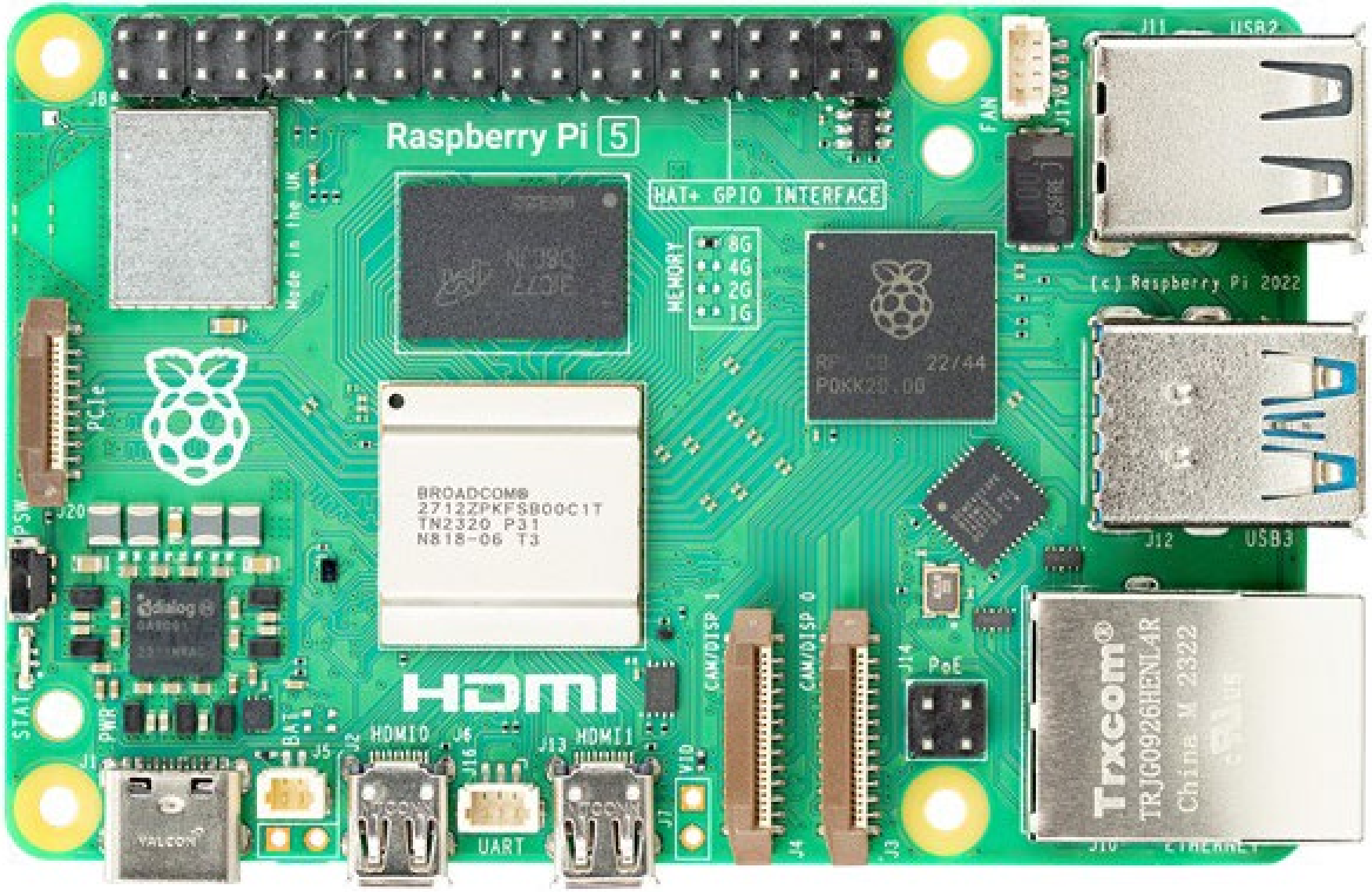
Assemble the Raspberry Pi 5 Hardware



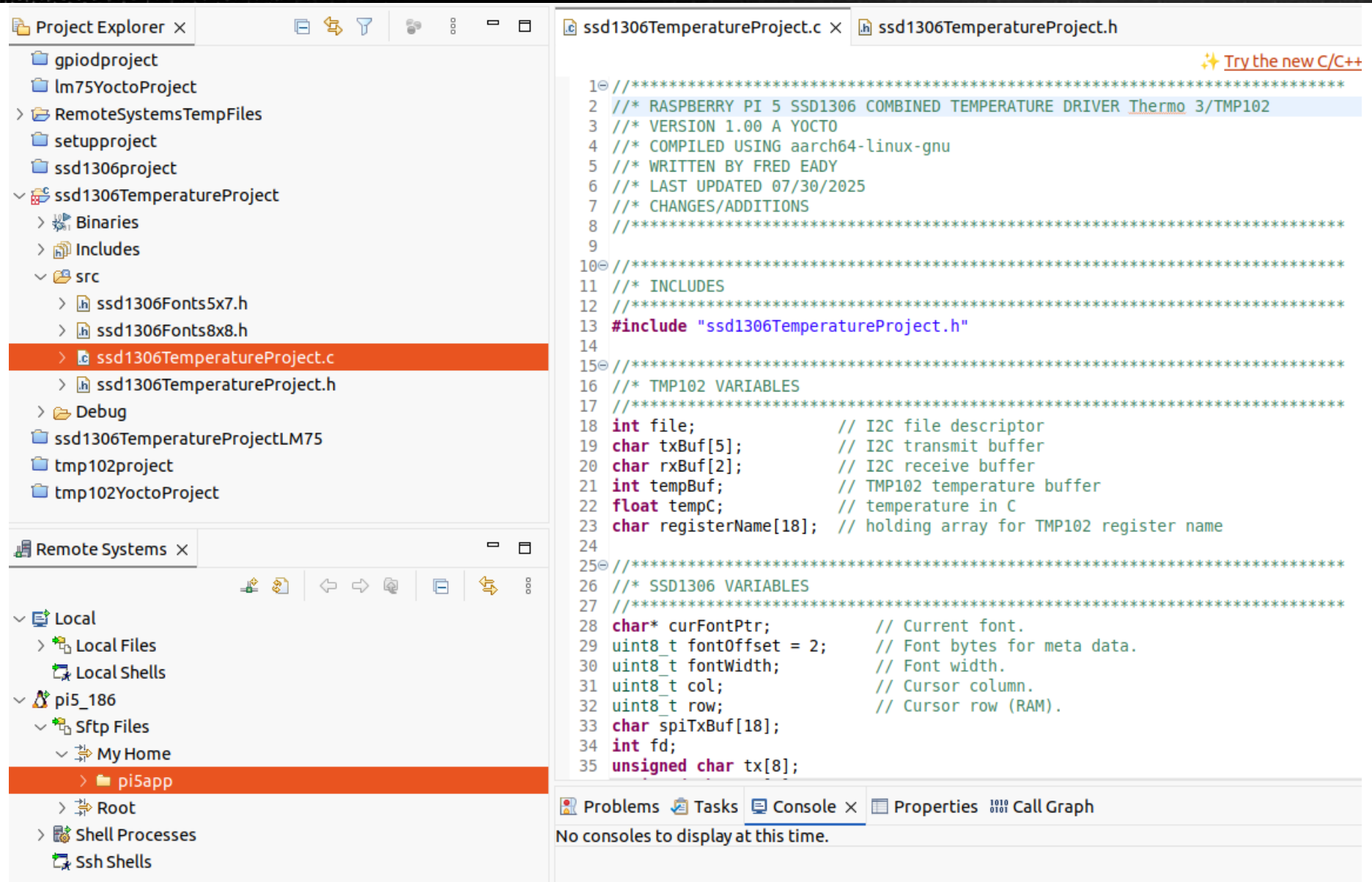




Eclipse Yocto Application Development



Create and Compile the Application Code



Enable the I2C Portal

```
//*****************************************************************************
// Open /dev/i2c-1 for Read/Write
//*****************************************************************************
if((file = open(filename, O_RDWR)) < 0)
{
    printf("ERROR: Failed to open %s.\n",filename);
    exit(1);
}
printf("INFO: %s opened for Read/Write.\n",filename);

//*****************************************************************************
// Change slave address to match TMP102 (0x48)
//*****************************************************************************
if(ioctl(file, I2C_SLAVE, tmp102_addr) < 0)
{
    printf("ERROR: Failed to acquire bus access or communicate with the device.\n");
    exit(1);
}
printf("INFO: Slave address changed to 0x%02x.\n",tmp102_addr);
```

Enable the SPI Portal

```
//*****************************************************************************
// Open /dev/spidev0.1 for Read/Write
//*****************************************************************************
fd = open("/dev/spidev0.1",O_RDWR);
if(fd < 0)
{
    printf("ERROR: /dev/spidev0.1 didn't open");
    exit(1);
}
printf("INFO: (file descriptor) fd = %d\n",fd);

line_request = request_output_lines(chip_path, line_offsets, line_values,
                                    NUM_LINES,
                                    "SSD1306-DC-RST");
if (!line_request) {
    printf("ERROR: Failed to request line\n");
    return EXIT_FAILURE;
}
```

Main Application Flow

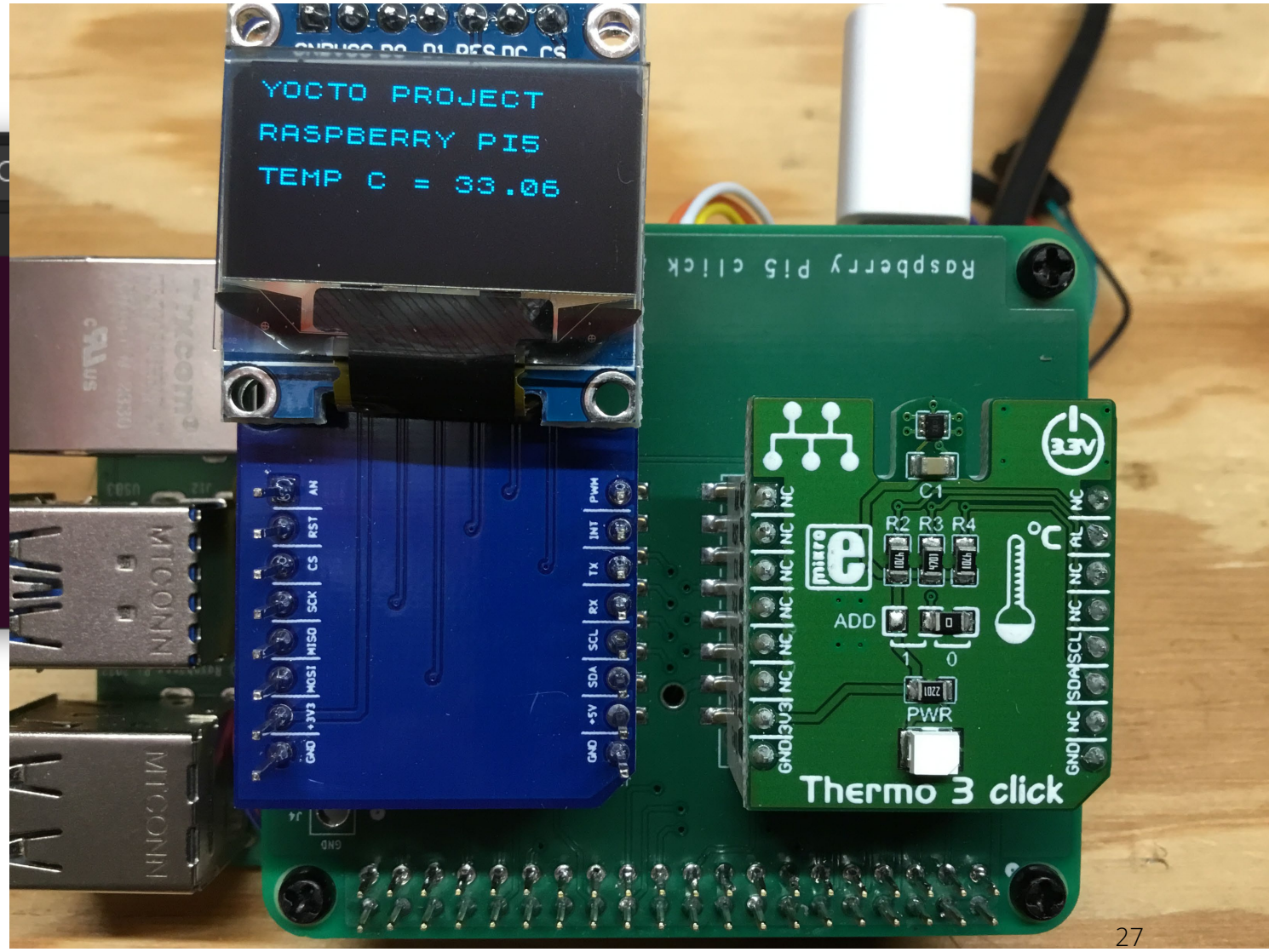
```
//*****************************************************************************
// INITIALIZE THE SSD1306 DISPLAY
//*****************************************************************************
initSSD1306();

//*****************************************************************************
// 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

```
//*****************************************************************************
// Compute and print the temperature
//*****************************************************************************
if(tempBuf & (1 << 11))
{
    tempBuf |= 0xF800;
}

tempC = (float)tempBuf * 0.0625;

printf("INFO: Current Temperature -> %.2f deg C\n\n",tempC);

setTextXY(0,0);
sprintf(spiTxBuf,"YOCTO PROJECT");
displayString(spiTxBuf);
setTextXY(2,0);
sprintf(spiTxBuf,"RASPBERRY PI5" );
displayString(spiTxBuf);
setTextXY(4,0);
sprintf(spiTxBuf,"TEMP C = %3.2F",tempC);
displayString(spiTxBuf);
```

Execute the Application Code

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

root@raspberrypi5:~/pi5app# ./ssd1306TemperatureProject
INFO: /dev/i2c-1 opened for Read/Write.
INFO: Slave address changed to 0x48.
INFO: Temperature Register Selected.
INFO: (file descriptor) fd = 4
INFO: 2 bytes were read-> rxBuf[0] 0x21 -- rxBuf[1] 0x10
INFO: 16-bit value = 0x0211
INFO: Current Temperature -> 33.06 deg C

root@raspberrypi5:~/pi5app#
```



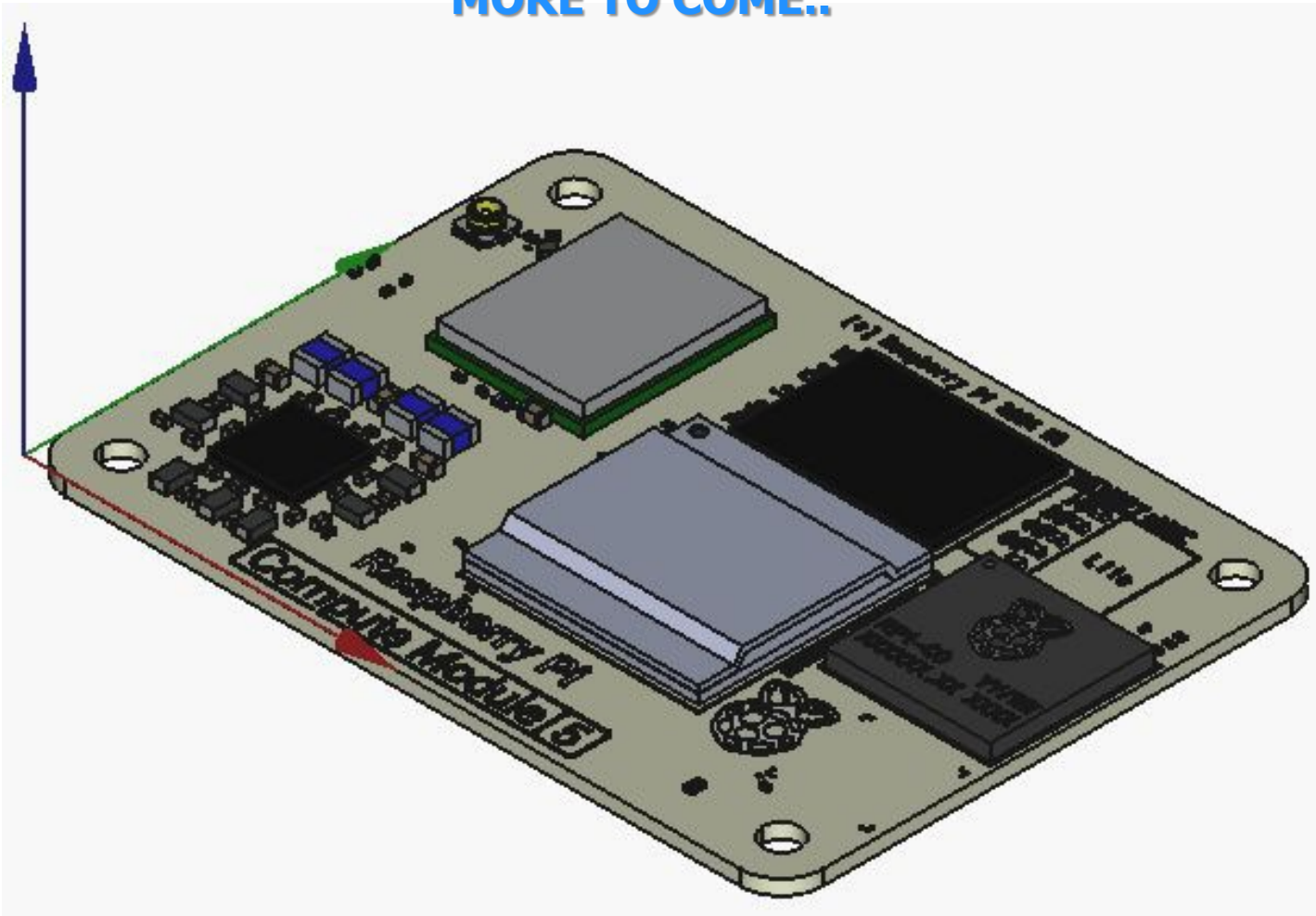
Next Time…

MORE TO COME..

Thank you for attending!!!

Please consider the resources below:

- Today's Download Package
- digikey.com



CEC Continuing Education Center

DesignNews

Thank You

Sponsored by

DigiKey

informa markets