



## Introduction

The öchin CM4v2 it's a tiny carrier board for the Raspberry Pi Compute Module 4. It is designed for applications where a powerful machine with low consumption and small dimensions is required. The small form factor makes it interesting for all those applications where the space available is not much and containing the weight is important, such as in robotics, home automation and IOT.

The board is compatible with all Raspberry Pi CM4 modules equipped with eMMC. Depending on your needs, you can select a CM4 module with an SDRAM starting from 1GB up to 8GB and the eMMC from 8GB up to 32GB, with or without the Wi-Fi / BT4 connection.

Key features of the Raspberry Pi CM4 compatible with öchin are as follows:

- Broadcom BCM2711, Quad core Cortex-A72 (ARM v8) 64-bit SoC @ 1.5GHz
- Small Footprint 55mm × 40mm × 4.7mm module
- H.265 (HEVC) (up to 4Kp60 decode), H.264 (up to 1080p60 decode, 1080p30 encode)
- OpenGL ES 3.0 graphics
- Options for 1GB, 2GB, 4GB or 8GB LPDDR4-3200 SDRAM with ECC
- Options for 8GB, 16GB, or 32GB eMMC Flash
  - Peak eMMC bandwidth 100MBytes/s
  - 2.4 GHz, 5.0 GHz IEEE 802.11 b/g/n/ac wireless
  - Bluetooth 5.0, BLE
  - On board electronic switch to select between PCB trace or external antenna
- 1 × USB 2.0 port (highspeed)
- MIPI CSI-2:
  - 1 × 2-lane MIPI CSI camera port, 1 × 4-lane MIPI CSI camera port
- 1 × USART, 3 × UART, 1 × I2C port, 1 × SPI port, Analog Video Output

## Design

The board has the same dimensions as the RPi CM4 module, 55mm x 40mm with rounded corners.

Several interfaces like USARTs, USBs are addressable by the SM06B-GHS and SM04B-GHS connectors, available on the sides of the board. Two FCC/FPC connectors are available to connect two MIPI CSI-2 cameras. The front camera is the “camera 1” which is a MIPI CSI-2 4lane camera. The side camera is the “camera 0” which is a MIPI CSI-2 2lane camera. The two FCC/FPC connectors have contacts on the bottom, the interface is the same used on the Raspberry Pi Zero.

On top of the board there is a 14x1.27mm connector, that could be used to connect a custom board on top, replacing the top PCB cover.

On the bottom of the öchin carrier board there are the DF40HC(3.0)-100DS-0.4v mating connectors for the CM4 module. The bottom cover is meant to protect the boards and mount the whole stack. The bottom cover could be eventually replaced by a commercial heatsink for the CM4 module.

## Applications

The powerful Quad core Cortex-A72 CPU and GPU, the big RAM size and the CODECs and Graphic Accelerators available on the CM4 makes the system very interesting for a huge number of applications. Are suitable all the applications related to audio / video manipulation and streaming, neural networks, machine learning, computer vision, etc.

There are several communication interfaces available on board that enables to connect the öchin to external hardware to connect to the lower abstraction layers, closer to the physical world.

An example of use could be a drone, where a flight controller board sense the physical references and control the actuators and motors to guarantee the stability of the flight. It could handle the stability of the vehicle, collect the data from the sensors and from the remote control and send the telemetry data back to the ground station. This is a very common structure nowadays for a UAS. The öchin board could be used to add a wide range of functionality to the system.

The öchin could handles up two MIPI cameras and others connected to the USB interfaces. Other sensors could also be added to the öchin using the UART, SPI, I2C

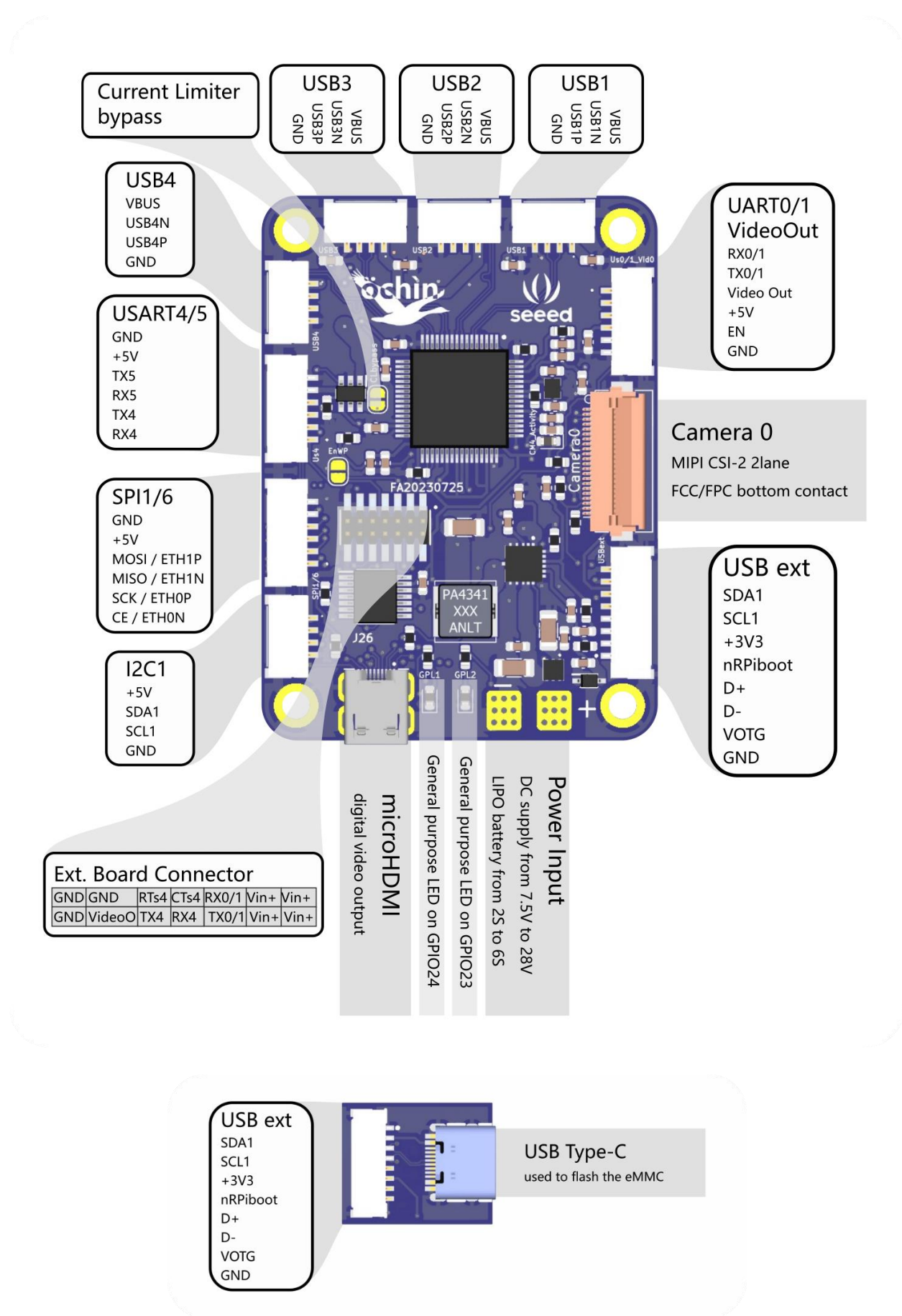
interfaces. All those data could be processed by several threads running on the CM4. The software infrastructure available within the CM4 enables to code with C/C++, python, MATLAB, and several other languages. There is also a wide number of frameworks that can be used to develop algorithms with Deep Learning, Neural Networks or Computer Vision technique, such as OpenCV, Caffe, PyTorch, TensorFlow etc. The öchìn board also provide the analog video output from the CM4. It means that it's possible to create a video output from the whole set of data, including images and its elaboration, to transmit like a common analog video out. This is very convenient because it gives the chance to see the output of the processing in real time without taking care of creating a digital wireless bridge.

This is only an example, a similar structure could be though for rovers, submarines, airplanes, and robots in general. Of course, the öchìn board it's very interesting for the IOT world also, where the size and weight is important. In this scenario, the AI is relevant and the tendency to miniaturize IOT devices and make them increasingly intelligent requires adequate hardware.

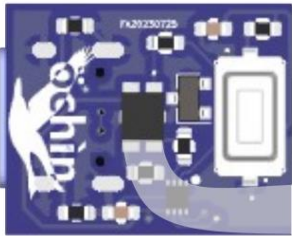
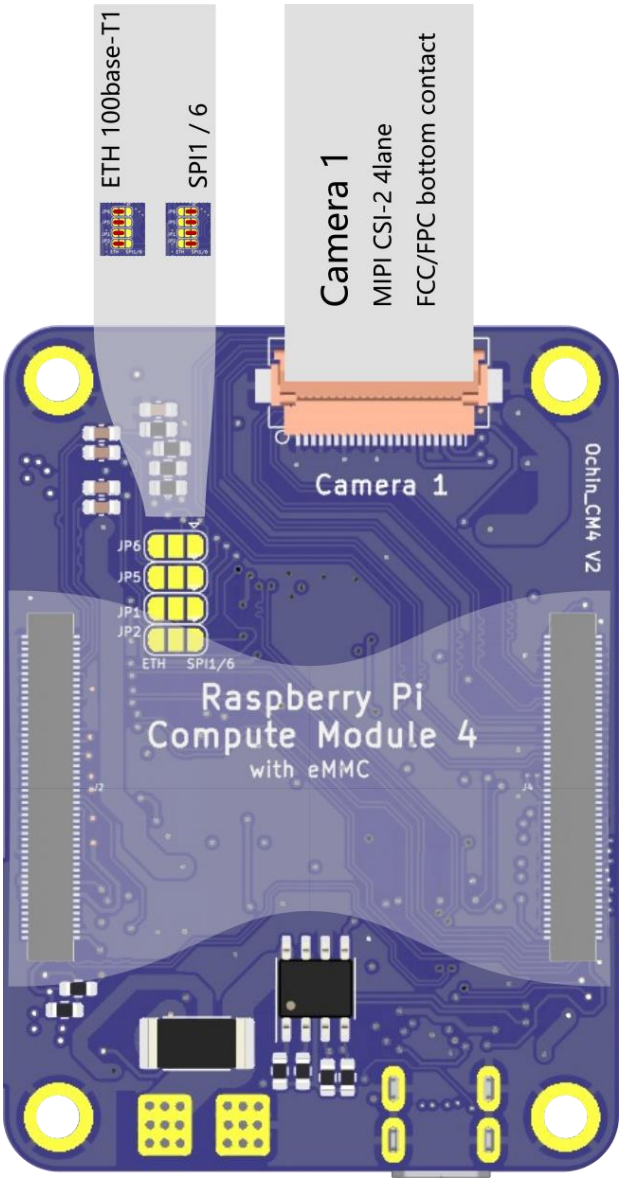
## Hardware interfaces

The öchìn CM4 board provides the following interfaces on the side's connectors:

- 4x USB 2.0 480Mbps (4x SM04B-GHS-TB(LF)(SN) connectors)
- 1x USB Type-C (for flashing eMMC)
- 2x CSI camera (2x FH12-22S-0.5SH (55) connectors)
- I2C1 (SM04B-GHS-TB(LF)(SN) connector)
- SPI1 / 6 (SM06B-GHS-TB(LF)(SN) connector)
- UART0 / 1 + Video Out (SM06B-GHS-TB(LF)(SN) connector)
- UART4 / UART5 (SM06B-GHS-TB(LF)(SN) connector)
- 1x microHDMI
- 2x general purpose LEDs
- 1x RGB LED on external tiny board
- 1x general purpose button on external tiny board



Top view

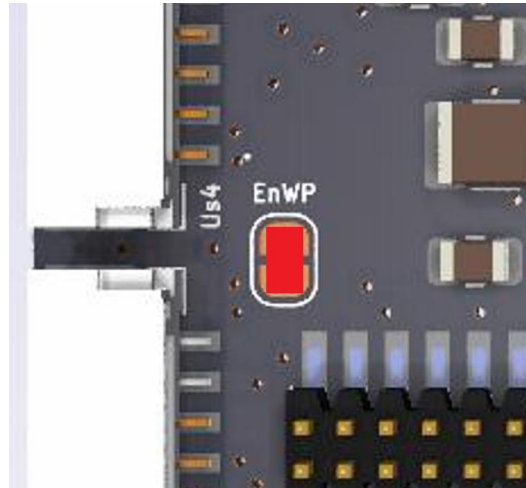


Boot CM4 as flash storage device  
General purpose Push Button

General purpose RGB LED

Bot view

To write protect the eMMC it's necessary to short the jumper EnWp:



## The Current Limiter

The IC4 - AP22615AWU-7 is a single-channel current-limited high-side power switch with output OVP optimized for USB and other hot-swap applications. On the ochin\_CM4 board, the current threshold of the AP22615AWU-7 is set to 3A. If the IC4 it's triggered, it will cut the VBUS on the USBs to keep the CM4 supplied while an overcurrent condition happens. There are four possible condition that triggers the current limiter:

(from the AP22615 datasheet Document number: DS41022 Rev. 5 – 2):

**Overcurrent and Short-Circuit Protection**

An internal-sensing FET is employed to check for overcurrent conditions. Unlike current-sense resistors, sense FETs do not increase the series resistance of the current path. When an overcurrent condition is detected, the device maintains a constant output current and reduces the output voltage accordingly. Complete shutdown occurs only if the fault stays long enough to activate thermal limiting.

Three possible overload conditions can occur. In the first condition, the output has been shorted to GND before the device is enabled or before  $V_{IN}$  has been applied. The AP22815/AP22615 senses the short-circuit and immediately clamps output current to a certain safe level.

In the second condition, an output short or an overload occurs while the device is enabled. At the instance the overload occurs, higher current can flow for a very short period of time before the current limit function can react. After the current limit function has tripped, the device switches into current limiting mode, and the current is clamped at  $I_{LIMIT}$  or  $I_{SHORT}$ .

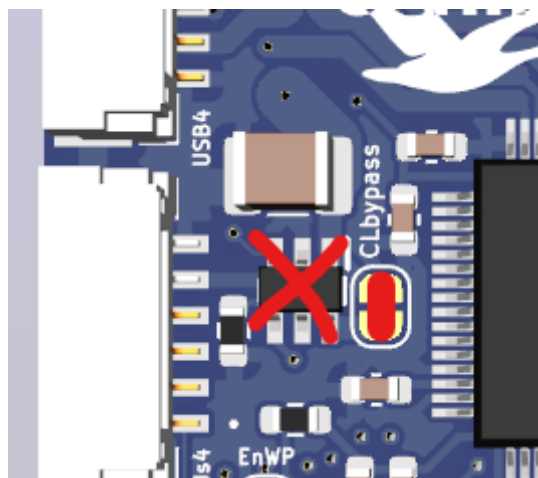
In the third condition, the load is gradually increased beyond the recommended operating current. The current is permitted to rise until the current-limit threshold ( $I_{TRIG}$ ) is reached or until the thermal limit of the device is exceeded. The AP22815/AP22615 is capable of delivering current up to the current-limit threshold without damaging the device. Once the threshold is reached, the device switches into its current limiting mode and is set at  $I_{LIMIT}$ .

**Thermal Protection**

Thermal protection prevents the IC from damage when heavy-overload or short-circuit faults are present for extended periods of time. The AP22815/AP22615 implements a thermal sensing to monitor the operating junction temperature of the power distribution switch. Once the die temperature rises to approximately  $+140^{\circ}\text{C}$  due to excessive power dissipation in an overcurrent or short-circuit condition the internal thermal sense circuitry turns the power switch off, thus preventing the power switch from damage. Hysteresis is built into the thermal sense circuit allowing the device to cool down approximately  $+35^{\circ}\text{C}$  before the switch turns back on. The switch continues to cycle in this manner until the load fault or input power is removed. The FLG open-drain output is asserted when an overtemperature shutdown or overcurrent occurs with 7ms deglitch.

When  $V_{IN}$  operates below 4V, the lower  $V_{IN}$  voltage results in higher equivalent  $R_{ON}$  and might potentially cause the chip to enter thermal cycling condition by higher output current.

In the case you need more current on the USBs, it's necessary to remove the IC4 - AP22615AWU-7 and short the JP4 (CLbypass). Before short the JP4 it's important that IC4 has been removed. If the current limiter is not present, the USBs supply is directly connected to the output of the +5V switching regulator, the same that supply the CM4. The power supply can provide up to 8A, leaving 2A for the CM4 (the manual says 1.4A but better be conservative), 6A remain available for the USBs and the external peripherals. That said, before remove the limiter it's important to know how much current the external devices will drawn, to leave the juice for the CM4 or it will reboot.



## Software Configuration

The öchìn board is only compatible with CM4 module with eMMC installed, because we decided to not mount the SD card slot to save space. Having the eMMC instead of SD also gives several advantages:

1. Faster than SD with speed up to 100Mbytes/s
2. More robust to physical corruption
3. Longer lifetime
4. More rugged

Having to write the ISO image directly to the eMMC is a bit more complicated than flash an SD card, but not so much thanks to the great work done by Raspberry Pi.

The procedure to flash the CM4 module is well explained in the Raspberry Pi Documentation:

<https://www.raspberrypi.org/documentation/hardware/computemodule/cm-emmc-flashing.md>

The procedure it's straightforward, what you need to do in synthesis is the following:

1. Power up the board with boot configuration as "mass storage device".
  - To do so you need to power off the board (no Vin connected)
  - Press the "nRPiboot" button on the external tiny board and, keeping it pressed power up the board using the "Vin" Connector.
  - Connect the board to your PC using the USB Type-C port
2. Run the "RPiboot" software, downloaded from the [Raspberry Pi website](https://www.raspberrypi.org/documentation/hardware/computemodule/cm-emmc-flashing.md). After the software starts, the PC will see the partition of the eMMC like if it would an SD into the SDcard reader.
3. Flash the eMMC as you normally do with the SD. You could use Win32DiskImager on Windows or "dd" command on Linux.

Once you have the system running for the first time, the USB interface will not work because on CM4 it is disabled by default. To use the USBs, you need to enable them in the "config.txt" file, adding the following line at the end of the file:



`dtoverlay=dwc2,dr_mode=host`

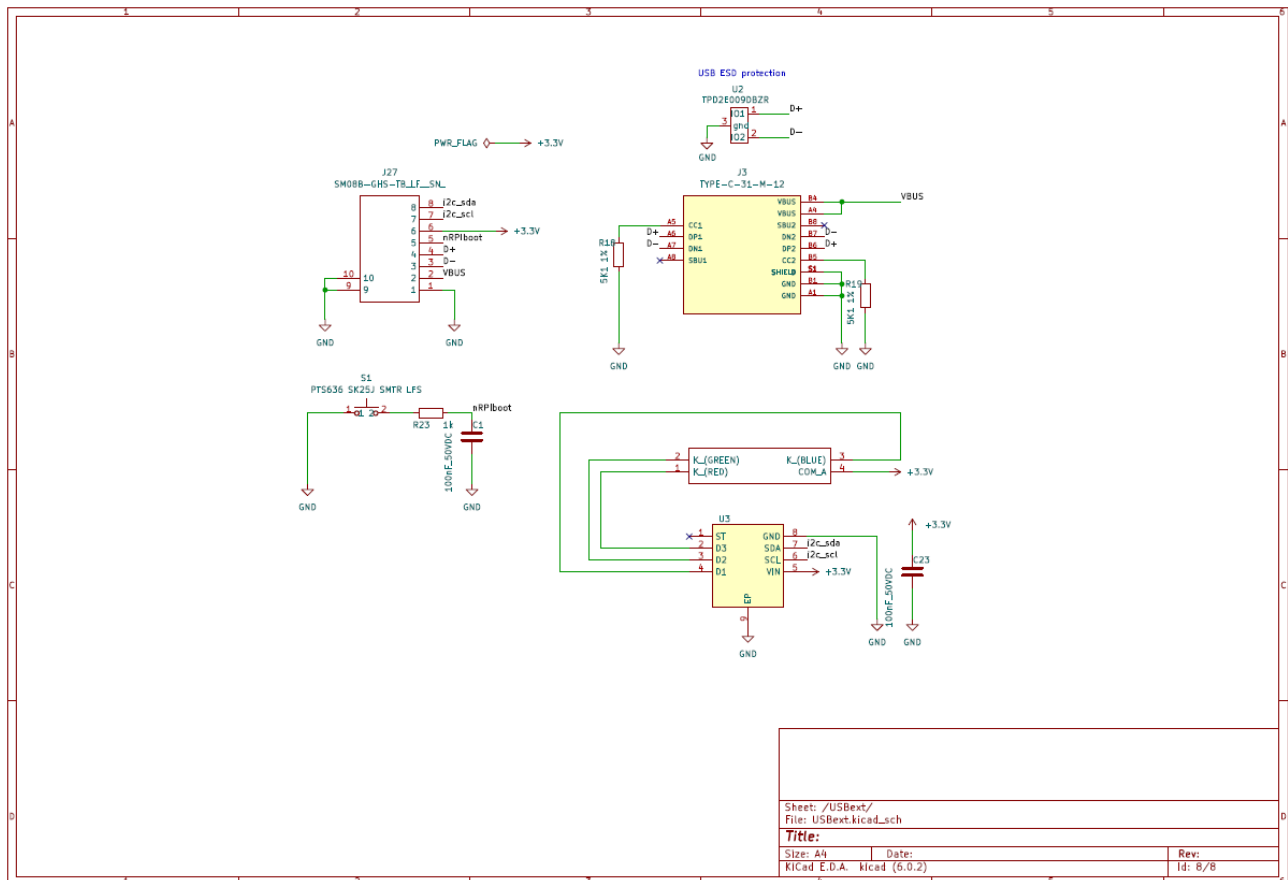
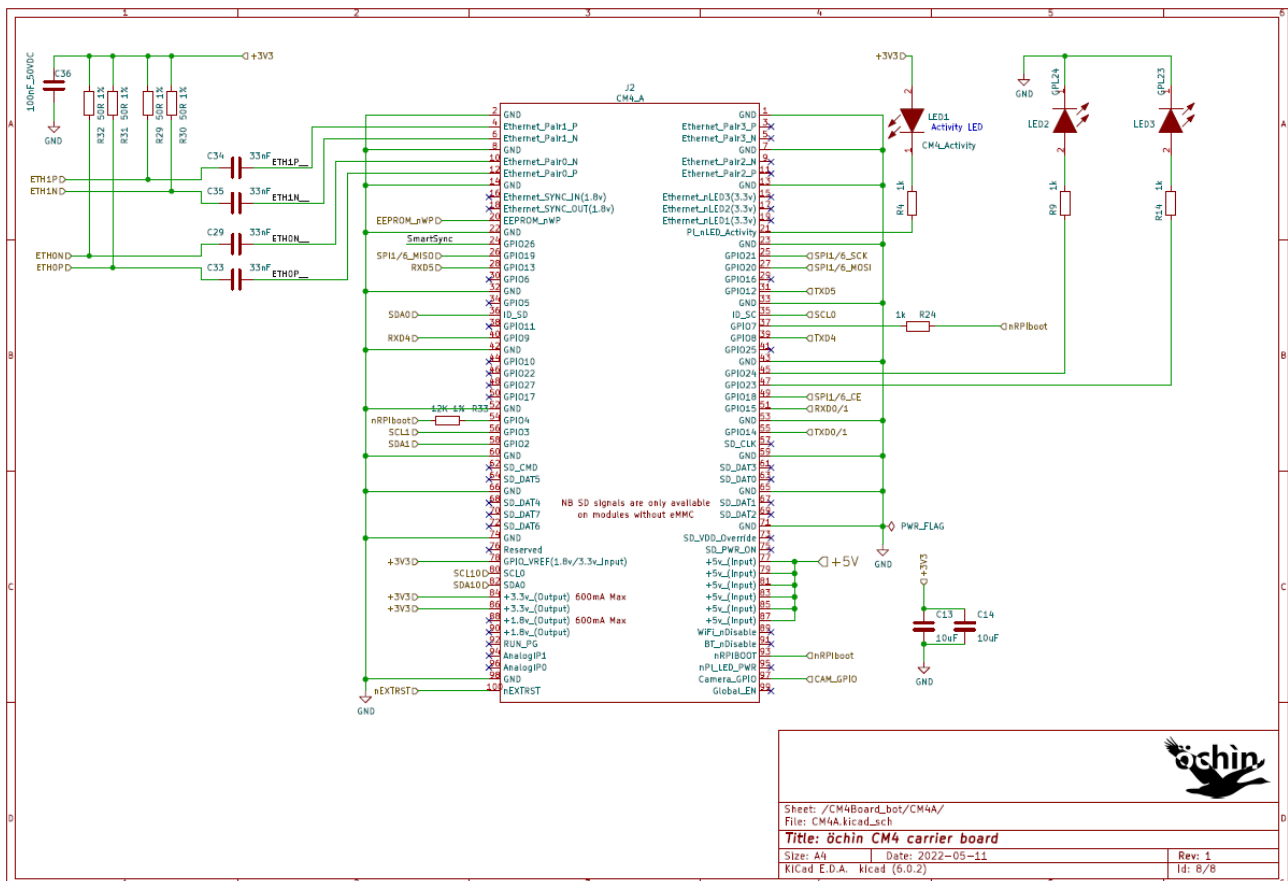
The “config.txt” file could be found within the “boot” partition. To be noted is that the “boot” partition is also accessible when you connect the CM4 to the PC booted in MSD mode, since it is formatted as FAT32.

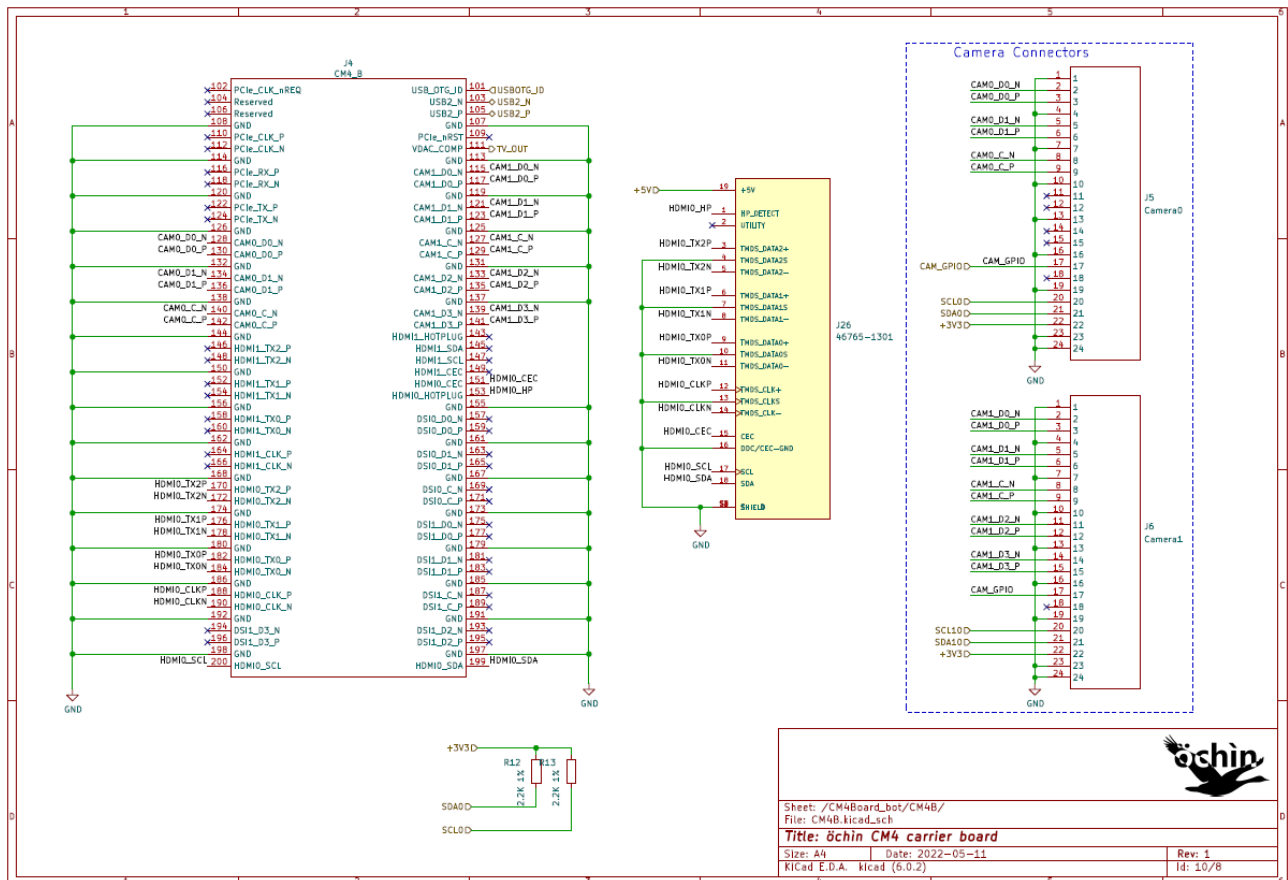
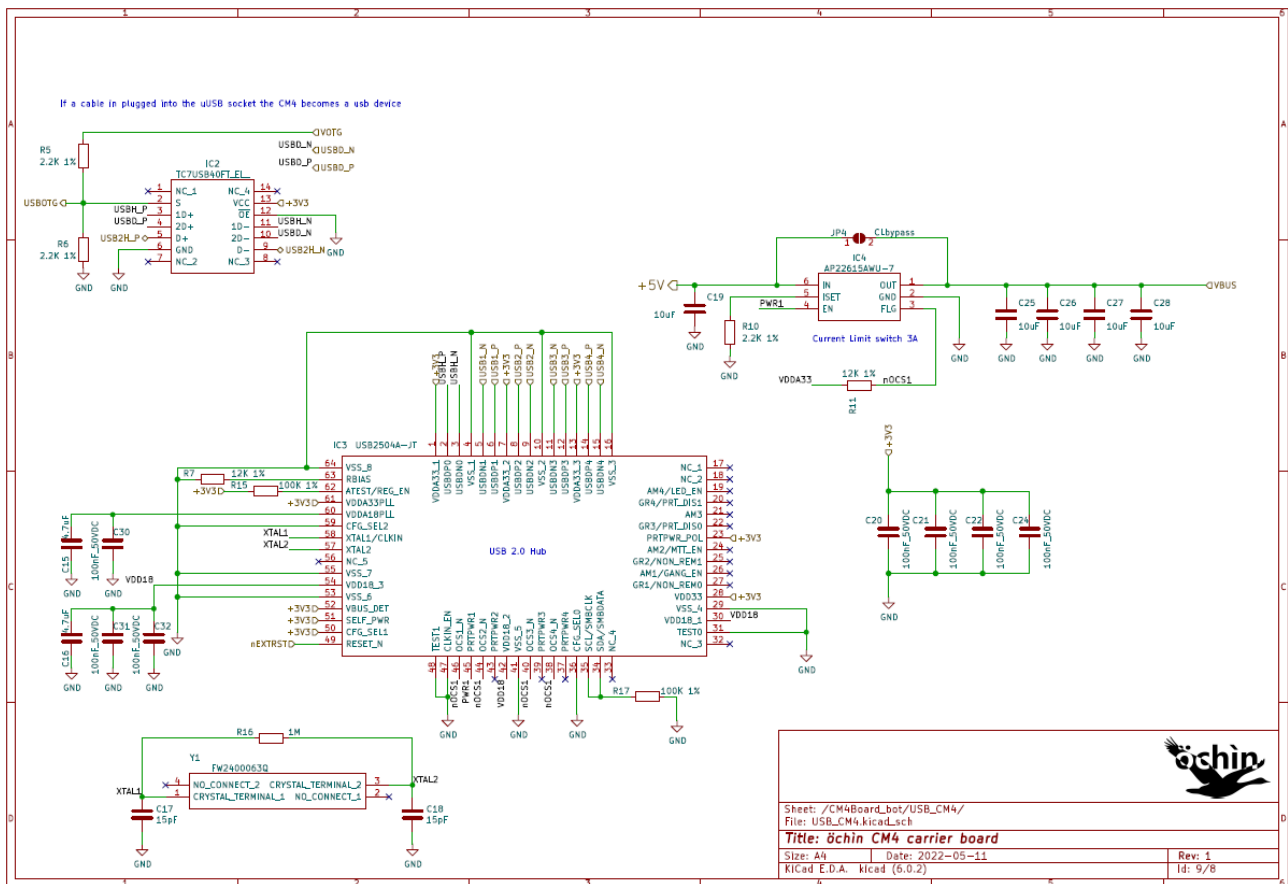
NB

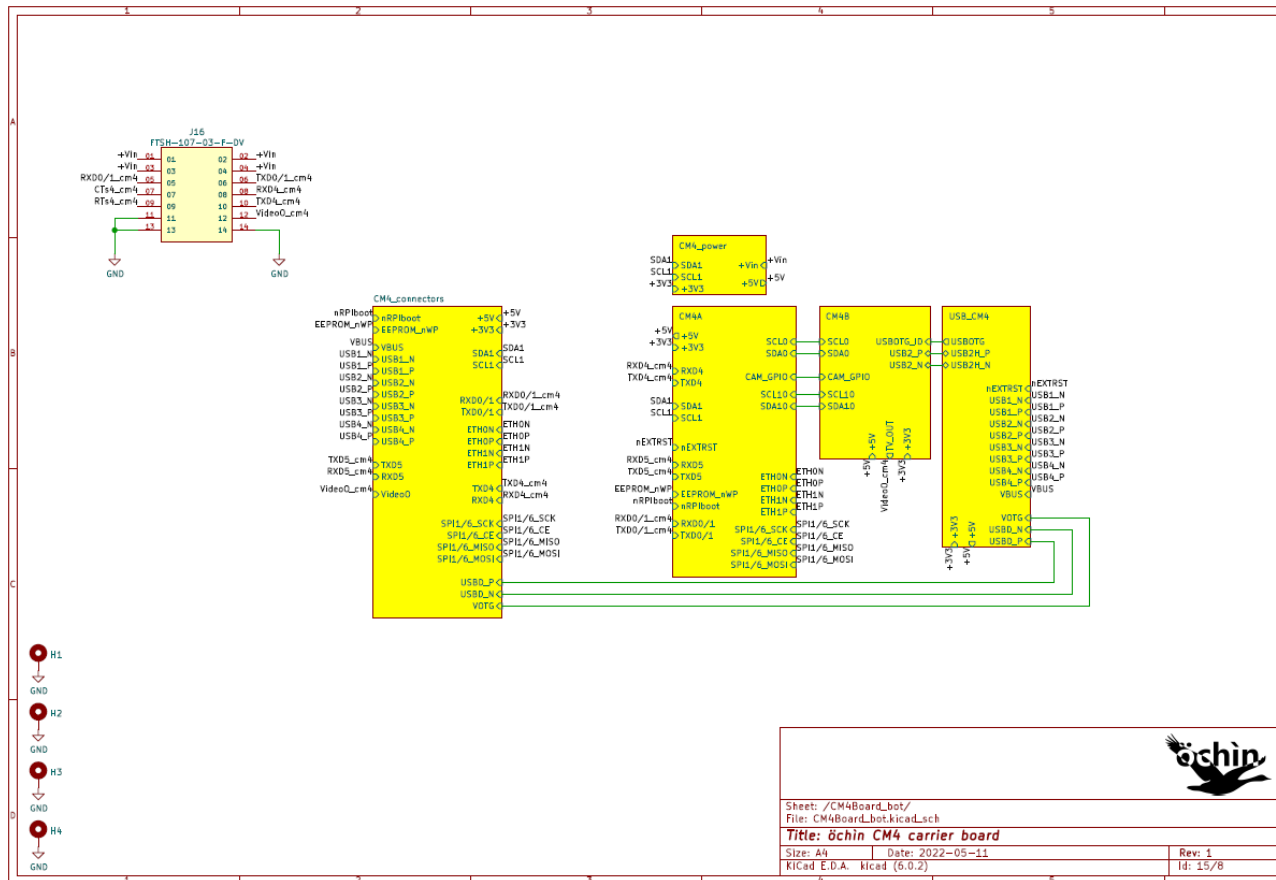
The Type-C VBUS is not connected to the power supply of the ochin board. This is to not supply everything you may have attached to the HUB (Wi-Fi dongle, cameras etc..), the power provided by the PC may not be sufficient. It means that if you only plug the USB Type-C nothing is going to happens... To power on the board the supply on the Vin connector is needed.

The Type-C VBUS it is used to switch the CM4 usb from the HUB to the Type-C port, this is to be able to flash the eMMC. For this reason, during the normal operations, the Type-C connector shouldn't be connected to anything.

[illegible]







# Mechanical stack view

