To start with the board

1. Connect your power supply to the board's USB power plug (the left plug, as shown in figure 1) and connect it to an outlet.
2. Connect your USB data cable to the other USB plug and to your host computer.

When you connect the USB cable to your computer, the board automatically boots up, and the board's LED turns green. **It then takes 20-30 seconds for the system to boot up**.

Note: We've released a system update that instead turns the LED orange during boot-up, and then green when the system is booted and ready for login. (Further below, you'll find instruction to install this update.)

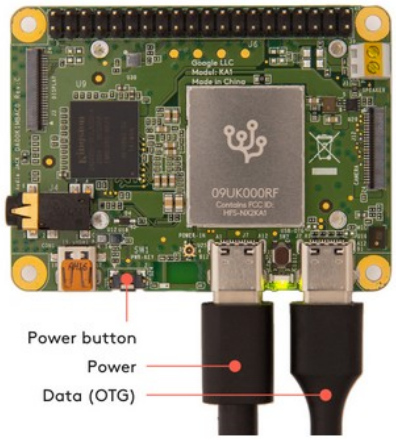

Fig 1. The USB power and data cables connected

About the USB ports

Both USB ports can power the board, but they behave differently:

- **USB power port (left):** This port supports power input only (no data). If you connect only this port, you must firmly press the power button to boot the board. Once the board is booted, you can press the power button to safely shut it down.
- **USB OTG port (right):** This port supports power input and USB data (as host or device). If you connect power here, the board boots up automatically. If you press the power button, the board reboots—you cannot fully shut down the board when the **OTG** port is connected to power (even if you shut down the operating system, it will reboot).

Attention: Safely shut down the board

Caution: Do not unplug the board while it is running. Doing so could corrupt the system image.

You should shut down the Dev Board Mini as follows:

1. **Be sure the board is connected to power through the USB power port** (see figure 1).
2. **Unplug the USB OTG cable** (if connected). This is important, because if the OTG port is connected, the board will immediately reboot upon shutdown.
3. Either **press the board power button** or run `sudo shutdown now` from the board terminal.
4. When the board's LED turns off, you can unplug the power supply.

Google CORAL min Dev software installation

Make sure MDT can see your device by running this command from your host computer:

```
mdt devices
```

You should see output showing your board hostname and IP address (your name will be different):

```
fun-zebra (192.168.100.2)
```

If you don't see your device, it might be because the system is still booting up. Wait a moment and try again.

2. Now to open the device shell, run this command:

```
mdt shell
```

After a moment, you should see the board's shell prompt, which looks like this:
mendel@fun-zebra:~$

If there is **problem**:

First solution:
You can connect your board directly to your HDMI monitor and remove the key from

```
/home/mendel/.ssh/authorized_keys
```

Second solution:
You can connect to the Dev Board Mini's **serial console** from Linux as follows:

1. First make sure your Linux user account is in the `plugdev` and `dialout` system groups by running this command:

   ```
   sudo usermod -aG plugdev,dialout <username>
   ```

   Then reboot your computer for the new groups to take effect.

2. Connect the USB-to-TTL serial cable to your computer and the board, as shown in figure 1:

   - Pin 6 is ground
   - Pin 8 is UART TX
   - Pin 10 is UART RX

Note that the cables must be **cross-connected**:
TX from board connects to RX on converter
RX from board connects to TX on converter

Warning: Do not connect the power wire (if provided). Doing so allows power to flow between the USB power and your computer, which degrades the USB power supply's ability to power the board, and can potentially damage to your hardware.

Now connect the board to power using the USB power port, and firmly press the power button to boot the board.

- Determine the USB-to-TTL cable's device name by running this command on your Linux computer:

  ```
  dmesg | grep ttyUSB
  ```

You should see results such as this:

```
[ 6437.706335] usb 2-13.1: cp210x converter now attached to ttyUSB0
```

```
dmesg | grep ttyUSB
```

If you don't see anything like this, double-check your USB cable is connected.

- Then connect with this command (using the name of the device listed in the previous step):

```
screen /dev/ttyUSB0 115200
Mendel GNU/Linux (eagle) mocha-kid ttyS0
mocha-kid login: mendel
Password: [mendel]
Last login: Thu Feb 14 10:12:04 UTC 2019 on tty7
Linux mocha-kid 4.19.125-mtk #1 SMP PREEMPT Thu Nov 12 23:33:16 UTC 2020 aarch64

The programs included with the Mendel GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Mendel GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
mendel@mocha-kid:~$ ls
END_USER_LICENSE
mendel@mocha-kid:~$ [   35.805543] usb0_vbus: disabling
[   35.808769] usb0_vbus_old: disabling
[   35.812387] backlight_vbus: disabling
[   35.816173] ldo_vm25: disabling
mendel@mocha-kid:~$ ls
END_USER_LICENSE
mendel@mocha-kid:~$
```

Now you can remove the authorization key.

Connect to the internet via WiFi and set the date

• Enter your network name and password with this command:

```
nmcli dev wifi connect <NETWORK_NAME> password <PASSWORD> ifname wlan0
nmcli dev wifi connect Livebox-08B0 password G79ji6dtEptVTPWmZP ifname wlan0
sudo date -s "20221015"           #put your date
sudo update-ca-certificates --fresh

sudo apt-get update
sudo apt-get upgrade
reboot now
```

Pycoral – inference (image) examples

If you have the problem with CA certificates do this:

```
sudo update-ca-certificates --fresh
sudo date -s "20221015"
-------------------------------------------
mkdir coral && cd coral
git clone https://github.com/google-coral/pycoral.git
cd pycoral
bash examples/install_requirements.sh
```

Coral Camera :

```
mkdir google-coral && cd google-coral
git clone https://github.com/google-coral/examples-camera.git --depth 1
mendel@elusive-dog:~/google-coral/examples-camera$ snapshot
Press space to take a snap, r to refocus, or q to quit
Saving image: img0001.jpg
Saving image: img0002.jpg
mendel@elusive-dog:~/google-coral/examples-camera$ ls
all_models
sh download_models.sh
```

Video – Camera examples :

```
sudo apt-get update
sudo apt-get dist-upgrade
export DEMO_FILES="$HOME/demo_files"

# The image classification model and labels file
wget -P ${DEMO_FILES}/
https://github.com/google-coral/test_data/raw/master/mobilenet_v2_1.0_224_quant_edgetpu.tflite
wget -P ${DEMO_FILES}/ https://raw.githubusercontent.com/google-coral/test_data/release-frogfish/
imagenet_labels.txt
# The face detection model (does not require a labels file)
wget -P ${DEMO_FILES}/
https://github.com/google-coral/test_data/raw/master/ssd_mobilenet_v2_face_quant_postprocess_edgetpu
.tflite
```

Audio :

```
git clone https://github.com/google-coral/project-keyword-spotter.git
cd project-keyword-spotter
sh install_requirements.sh
sudo apt-get install python3-pygame
bash run_snake.sh
```

Repositories:

```
mendel@xenial-yarn:/etc/apt/sources.list.d$ ls
00-mdt-services.list  multistrap-bsp.list  multistrap-edgetpu.list  multistrap-main.list  security.list
```

```
00-mdt-services.list
deb [trusted=yes] file:///var/cache/mdt/packages ./
```

```
multistrap-bsp.list
deb [arch=arm64] https://mendel-linux.org/apt/eagle-bsp-excelsior eagle main
deb-src https://mendel-linux.org/apt/eagle-bsp-excelsior eagle main
```

```
multistrap-edgetpu.list
deb [arch=arm64] https://packages.cloud.google.com/apt coral-edgetpu-stable main
deb-src https://packages.cloud.google.com/apt coral-edgetpu-stable main
```

```
multistrap-main.list
deb [arch=arm64] https://mendel-linux.org/apt/eagle eagle main
deb-src https://mendel-linux.org/apt/eagle eagle main
```

```
security.list
deb https://deb.debian.org/debian-security/ buster/updates main
deb-src https://deb.debian.org/debian-security/ buster/updates main
```