Confirm connection with:

```
ping joaopi.local
```

If pings are received then:

```
ssh pi@joaopi.local
```

it will ask for password: GUPI2025

Before disconnecting pi, run:

```
sudo halt
```

```
sudo raspi-config
```

Taking pictures

Setup:

```
libcamera-hello
```

or

```
libcamera-hello
```

Take an image

```
libcamera-still -o <output_name>.jpg --shutter <shutter_speed> --gain
<analogue gain> --awb <auto_white_balance>
```

`--shutter` (Shutter Speed)

- Controls **exposure time**, measured in **microseconds (µs)**.
- **Higher values** = brighter image (but can cause motion blur).
- **Lower values** = darker image (but sharper).
- `30000` = **30 milliseconds (ms)**.

Example Comparisons:

- `--shutter 1000` → 1 ms (very fast, good for bright conditions).
- `--shutter 50000` → 50 ms (longer exposure, better for low light).
- `--shutter 1000000` → 1 second (long exposure, not suitable for motion).

`--gain 2` (Analog Gain)

- Adjusts **sensor sensitivity** (similar to **ISO** in photography).
- **Higher gain** = brighter image, but more noise.
- **Lower gain** = darker image, but less noise.
- `2` is a **moderate** value; default is usually **1**.
- `--gain 1` → Low sensitivity, less noise (ideal for bright light).
- `--gain 5` → Higher brightness, but adds noise.
- `--gain 10` → Very bright, but very noisy.

`--awb auto` (Auto White Balance)**

- `--awb` controls **White Balance**, which adjusts colors to match lighting conditions.
- `auto` lets the camera **automatically balance colors**.
- If colors look off, you can manually set it.

- White Balance Modes:**

- `--awb daylight` → For outdoor shots.
- `--awb cloudy` → For overcast skies.
- `--awb tungsten` → For warm indoor lighting.
- `--awb fluorescent` → For office-style lighting.

Or use:

```
rpicam-still -o <output_name>.jpg --shutter <shutter_speed> --gain
<analogue_gain> --awbgains <red_gain>,<blue_gain> --immediate
```

same as libcamera-still but has manual red and blue colour balance. (can also use --awb auto)

Adjustments:

- `--awbgains 1.5,1.2` → Warmer tones (more red).
- `--awbgains 0.8,1.0` → Cooler tones (more blue).
- `--awbgains 1,1` → Neutral balance.

There is also

**`--immediate

- Captures the image **immediately**, **without showing a preview**.
- Useful when running via **SSH** (since there's no GUI preview).
- Without this flag, `rpicam-still` may show a **live preview before capturing**.

Streaming video

To stream a video:

```
libcamera-vid --inline --listen -t 0 -o tcp://0.0.0.0:5000
```

this is a TCP stream. The Pi acts as a server, sending video over a TCP socket to any device that connects to it.

--inline ensures that streaming metadata (SPS/PPS) is included in the stream which is required for video players like VLC, FFplay to properly interpret and display video

--listen puts the Raspberry Pi in server mode, meaning it will wait for the client (my laptop) to connect

-t controls the duration of the stream in milliseconds
-t 0 means run indefinitely

-o specifies the output destination for the video

tcp://0.0.0.0:5000 means we use the TCP protocol and use port 5000 for the stream and that we accept connections from any IP address.

Get Pi IP:

```
hostname -I
```

To watch the stream:

```
brew install ffmpeg

ffplay tcp://pi_ip:8888
```