TuboPark - Remote device

📦 Components

List of elements used to build the device

- **Raspberry Pi 5**
- **5V DC Stepper motor & drive module**
- **Jumper wires - female to female**
- **Camera module v3 - 75° angle**
- **Ribbon cable CSI / MIPI**
- **AI kit - Hailo AI & M.2 HAT**

🔍 Table of content

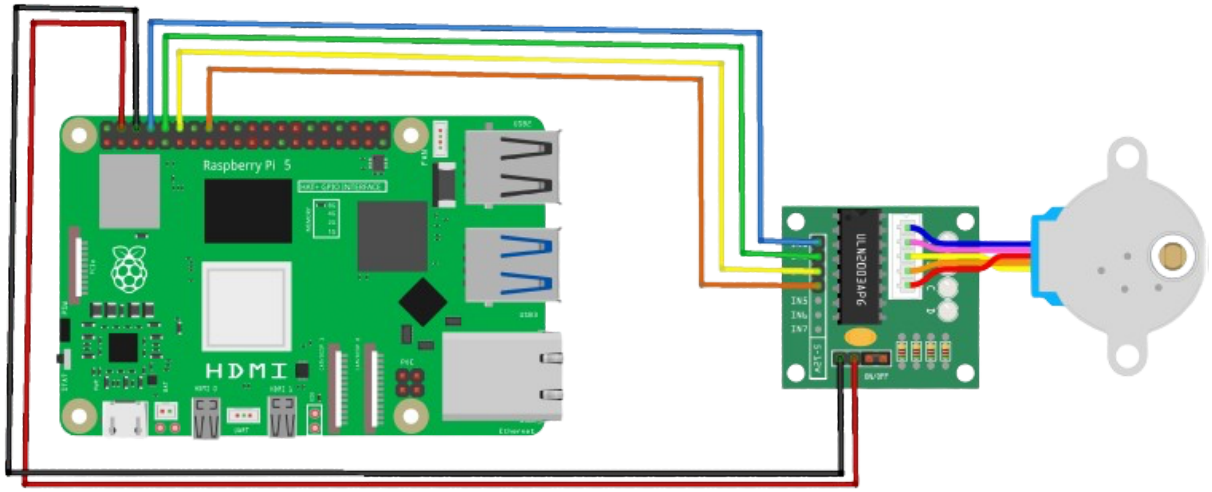
⚡ Wiring up & controlling the motor

Using:

- Raspberry Pi
- the stepper motor and its drive module

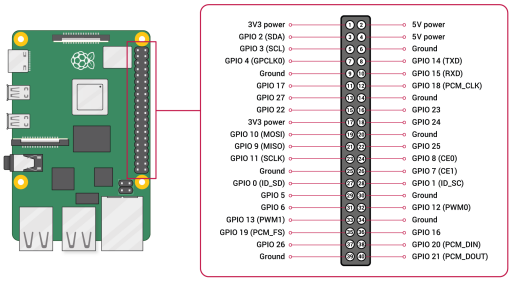
Libraries:

- gpiod
- time



GPIO pins

The GPIO (General Purpose Input/Output) pins are used to receive on/off signals to and from electronic components, in our case a motor.



GPIO pin capabilities

Input/Output:

- All GPIO pins can be configured as inputs or outputs.

Special Functions:

- Some GPIO pins have special or alternative functions. For example, certain pins can be used for `I2C`, `SPI`, `UART`, or `PWM`. These pins can still be used as generic GPIOs, but their specialized functions may limit their use in certain projects.

Voltages and Currents:

- The ability to source or sink current may vary slightly, but generally, it is about `16 mA` per pin with a total maximum for all pins combined (around 50 mA).

Stepper motor control

A stepper motor operates by receiving electrical pulses in a specific sequence, which turns the motor by a certain angle with each pulse. These pulses are sent to the motor coils via the driver module.

Control sequence:

- The `IN1`, `IN2`, `IN3`, and `IN4` pins receive signals in a determined sequence to control the different phases of the stepper motor. This sequence is essential to ensure the motor moves as desired (forward, backward, or holding position).

Generating movement:

- By sending `high/low` signals `(1/0)` to these four inputs in a certain order, the motor's rotation can be controlled:
  - A high signal to a pin/group of pins activates the corresponding coil of the motor.
  - This sequential activation creates a rotating magnetic field that turns the motor.

Example of a sequence used in the project:

> ℹ️ the rows of the sequences bellow can be translated as [IN1, IN2, IN3, IN4]

```
SEG_LEFT = [
    [0, 0, 0, 1],
    [0, 0, 1, 1],
```

```
SEG_RIGHT = [
    [1, 0, 0, 0],
    [1, 1, 0, 0],
```

```
    [0, 0, 1, 0],
    [0, 1, 1, 0],
    [0, 1, 0, 0],
    [1, 1, 0, 0],
    [1, 0, 0, 0],
    [1, 0, 0, 1]
]
```

```
    [0, 1, 0, 0],
    [0, 1, 1, 0],
    [0, 0, 1, 0],
    [0, 0, 1, 1],
    [0, 0, 0, 1],
    [1, 0, 0, 1]
]
```

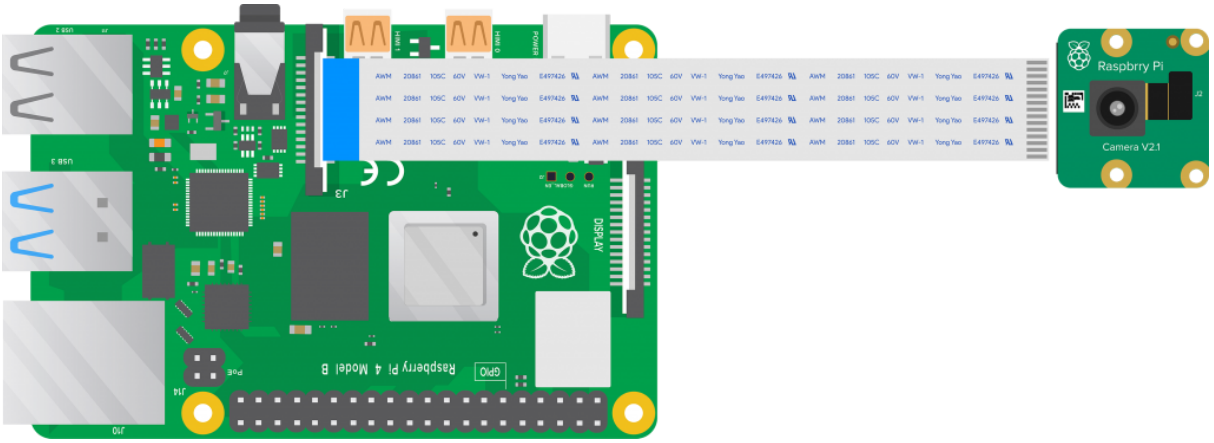
📸 Wiring up & controlling the camera

Using:

- Raspberry Pi,
- camera module v3
- the CSI / MIPI ribbon

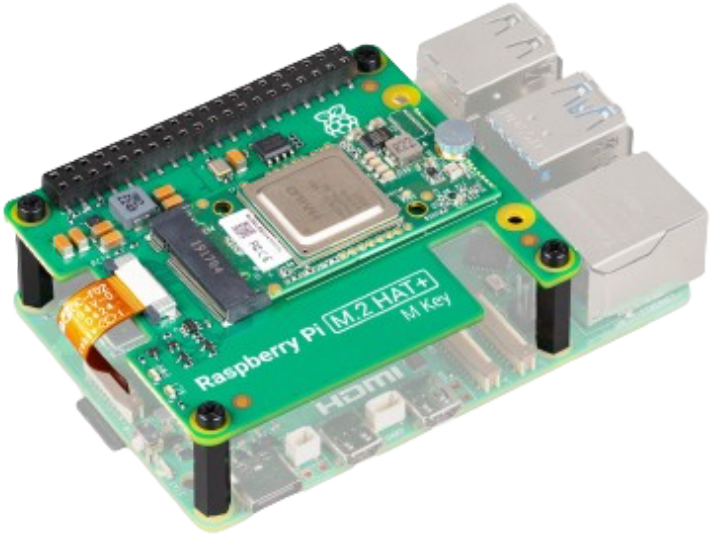
Packages & modules:

- picamera2
- JpegEncoder
- FileOutput



🚧 **In progress**

🤖 Wiring up & controlling the AI kit


Raspberry Pi M.2 HAT+
M Key
HDMI