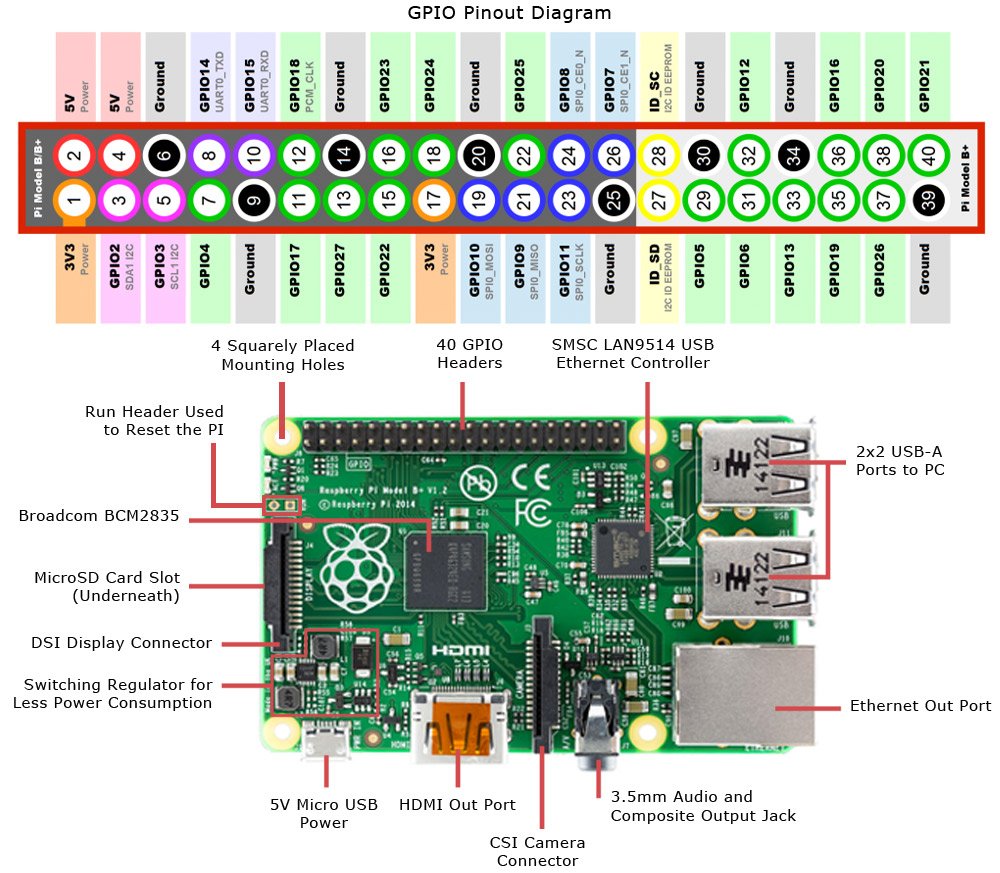
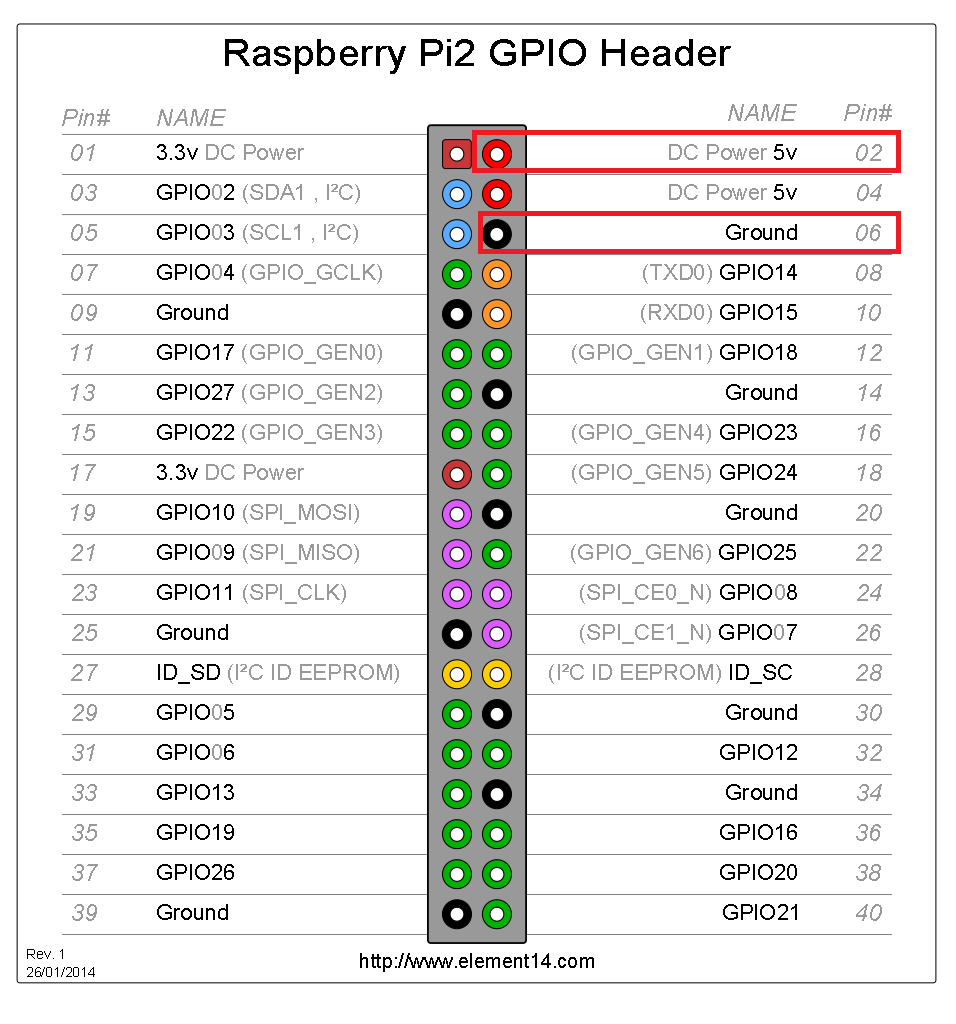
Instructions to use RoboClaw with Pi



****

<https://resources.basicmicro.com/configuring-the-raspberry-pi-3-serial-port/>

Boot up pi and plug in peripherals

Free up serial port on pi

5. Begin by enabling the miniUART. In the terminal enter the following to open the config file:

In terminal window type:

**sudo nano /boot/config.txt**

This opens the nano text editor so that an entry can be made to enable the UART. Scroll to the bottom of the file and add the following line:

**enable\_uart=1**

After entering the previous line press control+x to exit nano. When nano asks to save the changes press Y followed by enter and the configuration will be saved. After exiting nano the terminal prompt will be ready for the next set of commands.

6. Disable the serial console by entering the following commands:

**sudo systemctl stop serial-getty@ttyS0.service**

**sudo systemctl disable serial-getty@ttyS0.service**

7. Finally remove the serial console from the system by opening the cmdline.txt file:

**sudo nano /boot/cmdline.txt**

Once the file is open delete this portion of the entry in the file:

**console=serial0,115200**

Exit nano and save your changes as done in step 5

8. Reboot the Raspberry Pi and the changes will take effect.

9. The Raspberry Pi will now be able to use the miniUART on GPIO pins 14 and 15.

Getting started with Roboclaw

1. Install Basic Micro Motion studio
2. Attach the **negative** battery lead **and then** the **positive** battery lead to the center terminals on the terminal block labelled – and +
3. Connect USB to computer and open BM Motion Studio. Connect selected unit (in box on LH top side under “Attached Devices”)
4. Once finished click “disconnect selected unit”, then disconnect USB, then disconnect power from the **positive lead first** and **then the negative lead**. Disconnecting power in the opposite order can damage the Roboclaw.
5. Connect motors to terminal block.

M1A M1B Bat+ Bat- M2B M2A (A/B synonymous with +/-)

<https://resources.basicmicro.com/roboclaw-rc-controlled-differential-drive-setup/> RC control

<https://resources.basicmicro.com/dual-channel-roboclaw-quick-start-guide/> Quick Start

1. Connect encoders. EN1 for encoder from motor 1. Same for EN2. Using Pololu gearmotors #3237 4.4:1 25Dx48L MP 12V with 48 CPR encoder. 211.2 counts per revolution. <https://www.pololu.com/product/4861> -Superseded by #4861 which has encoder enclosure.

Example for motor 1

<https://resources.basicmicro.com/pololu-encoder-wiring/>

* 1. Red and black from motor 1 connector to M1A/B on Roboclaw
  2. Green to 1st – of +- power terminals. (Top right side corresponding to and not adjacent to EN1 terminals. Adjacent to LB in terminals) Encoder Ground
  3. Blue to 1st + of +- power terminals. Encoder Vcc
  4. Yellow to EN1+ Encoder channel A
  5. White to EN1- Encoder channel B

Connect motor 2 and encoder similarly.

1. Tune motors with BM motion studio. Verify motor and encoder directions are correct.
   1. <https://resources.basicmicro.com/auto-tuning-with-motion-studio/>
   2. <https://www.youtube.com/watch?v=UsKsnQNgVQk>
2. The control mode must be set correctly for the mode of operation desired. The options are: packet serial, simple serial, rc and analog. Click on General Setting on the left-hand side of the application. Packet serial is used with the available libraries to allow a microcontroller to operate the controller. Simple serial is a simplified version of packet serial for use with microcontrollers. RC mode allows an RC radio and receiver to operate the controller. Finally, analog mode allows analog devices such as joysticks and potentiometers to control one or both motor channels.

While still in Motion Studio click on General Settings on the left-hand side of the application. Locate the pane labelled Setup and set the dropdown labeled Control Mode to the desired mode. Save the settings to the board by clicking “Device” in the menu at the top of the application and then clicking “Write Settings”.

1. More information on using each mode can be found in our application notes. Some additional wiring and software configuration must be done for some modes. Below are links to the relevant articles.

Using a microcontroller in packet serial mode

<https://resources.basicmicro.com/simple-arduino-control-of-the-roboclaw/>

<https://resources.basicmicro.com/packet-serial-with-the-raspberry-pi-3/>

1. Prepare Roboclaw as above
2. Configure Pi serial port hardware as above
3. Disconnect power from Roboclaw and shut down Pi
4. Wire Pi and Roboclaw together
   1. GPIO 14 to S1 signal pin (+ pin closest to board edge)
   2. GPIO 15 to S2 signal pin
   3. Any ground pin to S1 ground pin (pin closest to inside of board)
   4. Middle pin is 5V power out. If Pi is powered alternatively, leave this unconnected.
5. Connect power to RoboClaw (Attach the **negative** battery lead **and then** the **positive** battery lead to the center terminals on the terminal block labelled – and +.)
6. Connect the RoboClaw to the computer with a micro USB cable. Open BasicMicro Motion Studio and connect the RoboClaw as done previously in the tutorial linked in step 1.
7. In the Motion Studio application click on “General Settings” in the left-hand pane. Now locate the pane titled “Setup”. Set the control mode to “Packet Serial”. Now move down to the pane labeled “Serial”. Set the address to 128 and the baud rate to 38400. In the menu at the top of the application select “Device” and then “Write Settings” to save the settings to the board. If this is not done the settings will not be saved to the board. Disconnect the RoboClaw by clicking “Disconnect Selected Unit” in the upper left-hand side of Motion Studio.
8. Power up the Raspberry Pi by connecting power to it with the micro USB cable and connect to internet.
9. Add the pyserial library to the Pi’s Python library. Open a terminal and enter the following command:

**pip3 install pyserial** (for python 3. Roboclaw developed for Python 2.)

**pip install pyserial** (Python 2)

If pip is not installed on the Raspberry Pi it can be installed with the following command:

**sudo apt-get install python-pip**

1. Download the test code for this tutorial [from here](https://github.com/basicmicro/raspberry_pi_packet_serial). <https://github.com/basicmicro/raspberry_pi_packet_serial>
   1. The repository can also be cloned from GitHub from the command line. Install Git with the following command: **sudo apt-get install git**
   2. Then clone the repository with the following command:

**git clone** [**https://github.com/basicmicro/raspberry\_pi\_packet\_serial.git**](https://github.com/basicmicro/raspberry_pi_packet_serial.git)

Make sure to use Python 3 version. Also use “import roboclaw\_3 from roboclaw” If you don’t and your computer is running Python 3, you will have to follow these instructions:

Lines changed in roboclaw.py in order to get it to work with python3

#Under: def \_sendcommand(self,address,command):

#Commented out line 126(Now line 130). Added line 127 and 128.

#Commented out line 127. March 10 2022. Mike Stromecki

#Issue between Python 2 and Python 3? Yes it is. PITA

#Changed line 135 self.\_port.write(chr(command)) to self.\_port.write(bytes([command])) 3/11/22

#Under def \_writebyte(self,val):

#Changed line 187 self.\_port.write(chr(val&0xFF)) to use bytes as above

#Under def SendRandomData(self,cnt):

#Changed line 656 self.\_port.write(chr(byte)) to use bytes as above

Then the rc library will work.

1. Unzip the code you’ve downloaded and navigate to the unzipped directory via the terminal. If you’ve cloned the repository with Git there is nothing to unzip. Run the code provided by entering the following: **python packet\_serial.py**
2. Now that the example code is running you should see motor 1 running for two seconds and then stopping followed by motor2 doing the same.

If the motors are not running as expected ensure that the connections between the Pi and the RoboClaw are secure. Also make sure that the TX and RX lines are not reversed. The TX from the Pi (GPIO 14) goes to the RX pin on the RoboClaw (S1 signal pin) and the RX from the Pi (GPIO 15) goes to the TX pin on the RoboClaw (S2 signal pin). Finally check to see if there is a solid common ground connection between the two boards.

In the example code the only two functions called are M1Forward and M2Forward. However there are dozens of functions to control the RoboClaw in the included Python library. In the downloaded example code the library file is called “roboclaw.py”. Browse through this file to see all of the functions that can be used with the RoboClaw.

To control the multiple RoboClaws with the Pi

<https://resources.basicmicro.com/packet-serial-bus-operation-with-the-raspberry-pi-3/>

or

<https://resources.basicmicro.com/standard-serial-bus-with-the-raspberry-pi-3/>

First link 2 wire connections and I believe ~~I2C~~ UART connection. Second link, serial control using different GPIO header pins. Prefer first link I think.

Roboclaw and Python

<https://resources.basicmicro.com/using-the-roboclaw-python-library/>

<https://resources.basicmicro.com/using-encoders-with-the-python-library/>

Avoid ground loops

<https://resources.basicmicro.com/roboclaw-and-mcp-grounding-issues/>

Connect a 1k Ohm resistor in series with on/off switch to slowly charge Roboclaws in 0.6s Less resistance for faster charging. Also consider diode as suggested. <https://resources.basicmicro.com/motor-controller-hardware-precautions/>

E-Stop wiring page 46 in user manual

Read quadrature encoder speed; manual page 87

RC 129 S3 set to Estop