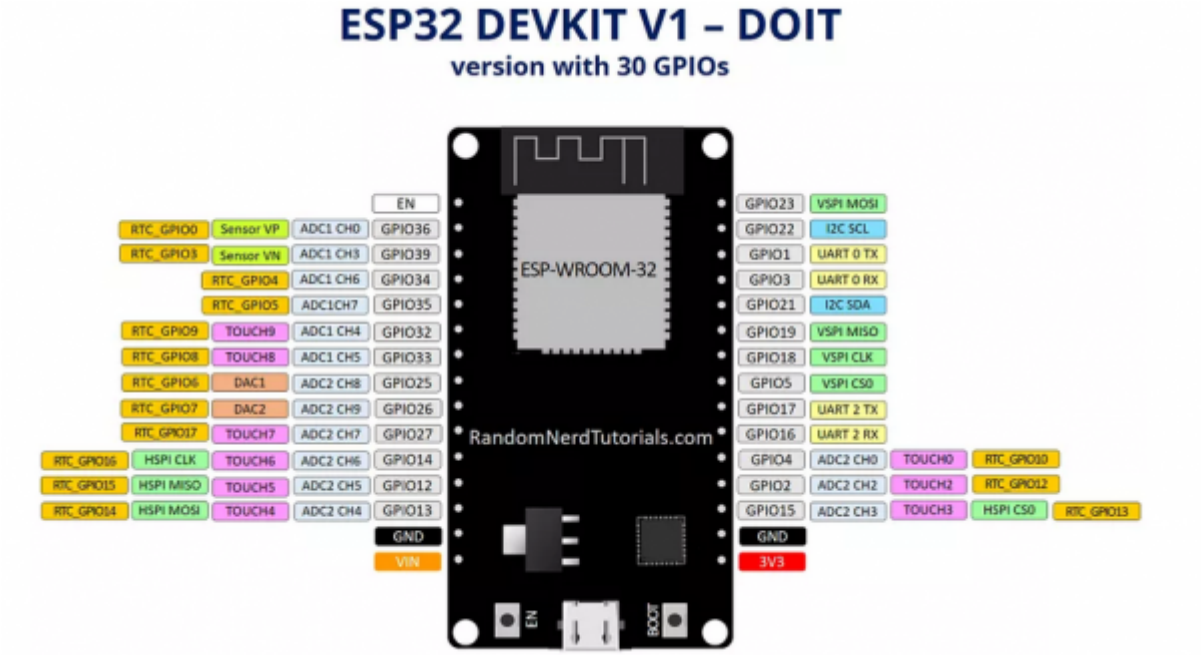
Board: ESP32-DEVKIT1

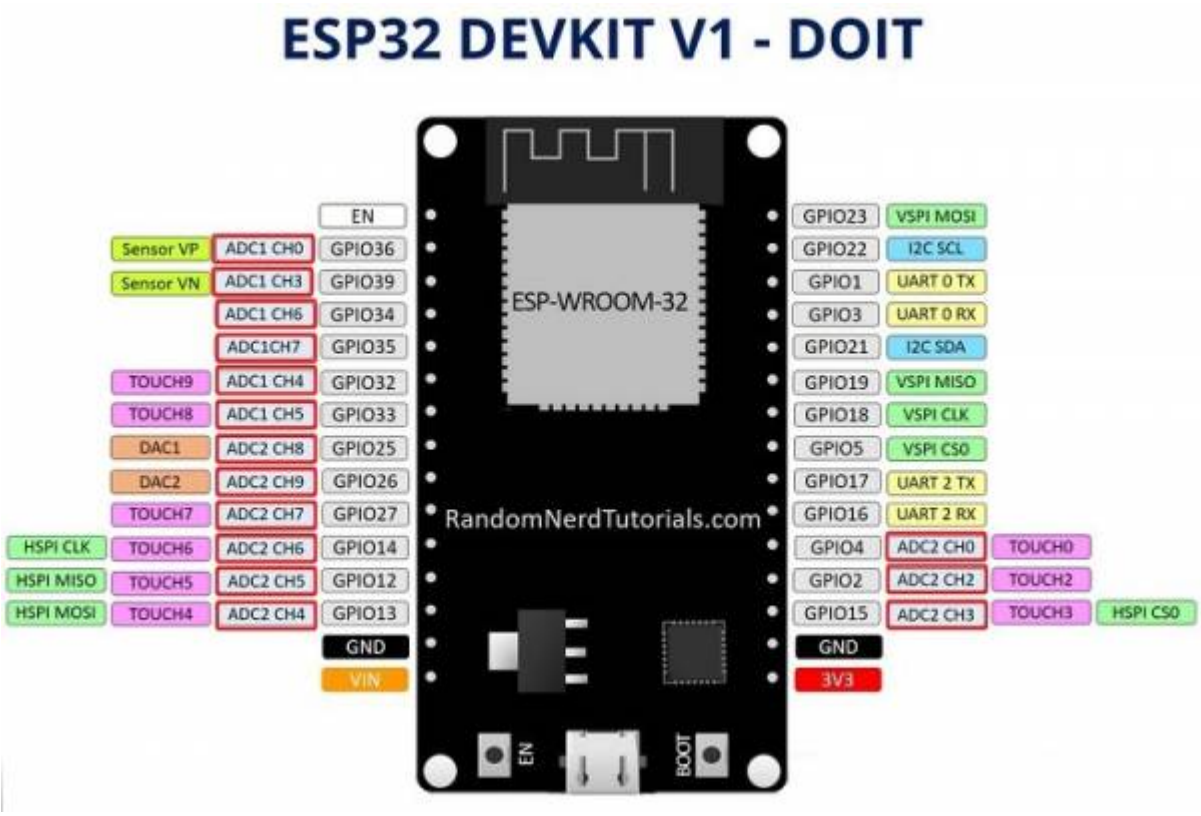
Got this board to play around with...

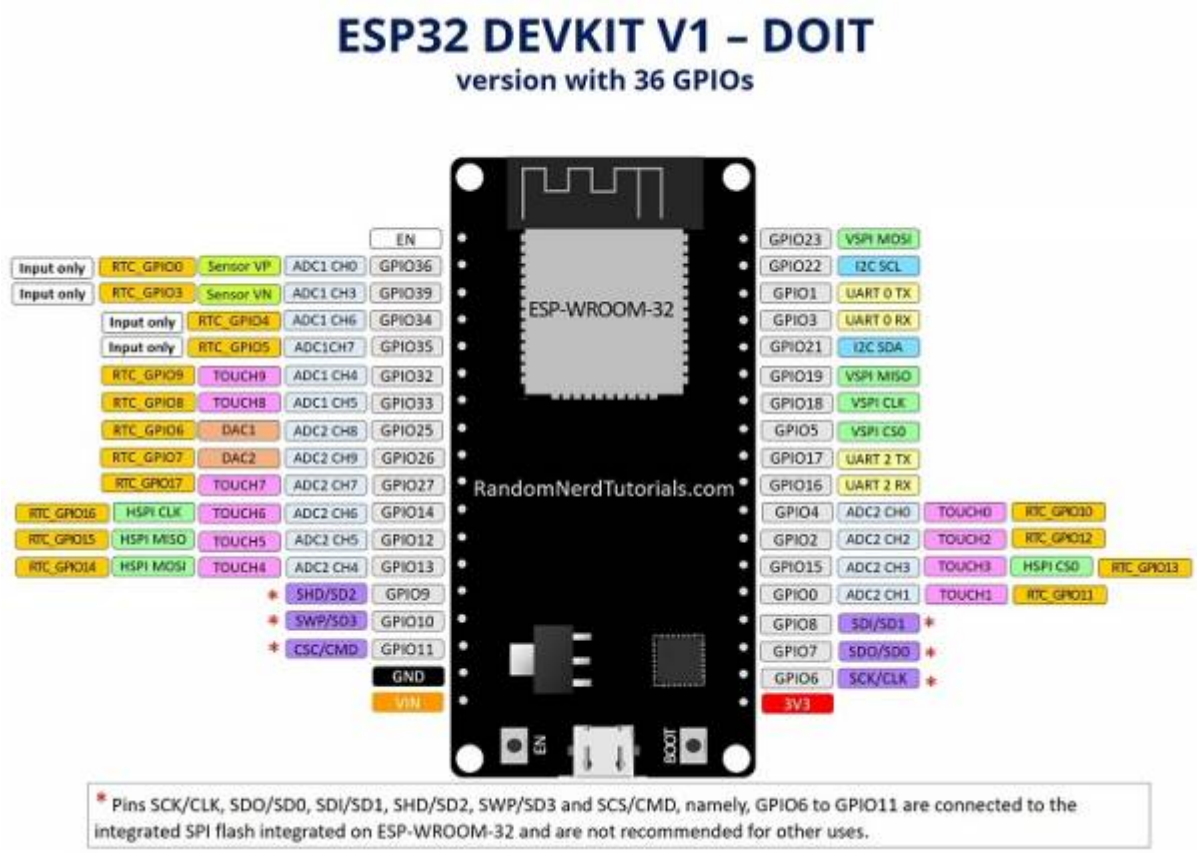
esp32_devkit1.txt

```
--------------
ESP32-DEVKITV1
--------------

- clone of DOIT's ESP32 DEVKIT V1?
- ESP-WROOM-32 (ESP32-D0WDQ6)
  = ESP32-D0WD-based modules with integrated flash
- DECISION20200822:
  = toolchain and build environment is simply not worth it!
  = WILL USE micropython instead!

-----------
micropython
-----------

- get some tools
  > pip install esptool rshell
- download binary from https://micropython.org/download/esp32/
  = selected esp32-idf3-20191220-v1.12.bin
  = idf4 seems to be missing LAN? stick with idf3 for now...
- erase flash
  > esptool.py --chip esp32 --port /dev/ttyUSB0 erase_flash
- flash micropython
  > esptool.py --chip esp32 --port /dev/ttyUSB0 --baud 460800
write_flash -z 0x1000 <bin>
- use rshell to manage code (vfs)
  > rshell --port /dev/ttyUSB0 --buffer-size=32
- download file in rshell
  > cp <file> /pyboard[/<more_path>]
```

Decided to use micropython. Pinouts shown below (*Disclaimer: Image copied from RandomNerdTutorials*).


ESP32 DEVKIT V1 – DOIT
version with 30 GPIOs
ESP-WROOM-32
RandomNerdTutorials.com
EN
RTC_GPIO0 Sensor VP ADC1 CH0 GPIO36
RTC_GPIO3 Sensor VN ADC1 CH3 GPIO39
RTC_GPIO4 ADC1 CH6 GPIO34
RTC_GPIO5 ADC1CH7 GPIO35
RTC_GPIO9 TOUCH9 ADC1 CH4 GPIO32
RTC_GPIO8 TOUCH8 ADC1 CH5 GPIO33
RTC_GPIO6 DAC1 ADC2 CH8 GPIO25
RTC_GPIO7 DAC2 ADC2 CH9 GPIO26
RTC_GPIO17 TOUCH7 ADC2 CH7 GPIO27
RTC_GPIO16 HSPI CLK TOUCH6 ADC2 CH6 GPIO14
RTC_GPIO15 HSPI MISO TOUCH5 ADC2 CH5 GPIO12
RTC_GPIO14 HSPI MOSI TOUCH4 ADC2 CH4 GPIO13
GND
VIN
GPIO23 VSPI MOSI
GPIO22 I2C SCL
GPIO1 UART 0 TX
GPIO3 UART 0 RX
GPIO21 I2C SDA
GPIO19 VSPI MISO
GPIO18 VSPI CLK
GPIO5 VSPI CS0
GPIO17 UART 2 TX
GPIO16 UART 2 RX
GPIO4 ADC2 CH0 TOUCH0 RTC_GPIO10
GPIO2 ADC2 CH2 TOUCH2 RTC_GPIO12
GPIO15 ADC2 CH3 TOUCH3 HSPI CS0 RTC_GPIO13
GND
3V3
EN
BOOT


ESP32 DEVKIT V1 - DOIT
ESP-WROOM-32
RandomNerdTutorials.com
EN
Sensor VP ADC1 CH0 GPIO36
Sensor VN ADC1 CH3 GPIO39
ADC1 CH6 GPIO34
ADC1CH7 GPIO35
TOUCH9 ADC1 CH4 GPIO32
TOUCH8 ADC1 CH5 GPIO33
DAC1 ADC2 CH8 GPIO25
DAC2 ADC2 CH9 GPIO26
TOUCH7 ADC2 CH7 GPIO27
HSPI CLK TOUCH6 ADC2 CH6 GPIO14
HSPI MISO TOUCH5 ADC2 CH5 GPIO12
HSPI MOSI TOUCH4 ADC2 CH4 GPIO13
GND
VIN
GPIO23 VSPI MOSI
GPIO22 I2C SCL
GPIO1 UART 0 TX
GPIO3 UART 0 RX
GPIO21 I2C SDA
GPIO19 VSPI MISO
GPIO18 VSPI CLK
GPIO5 VSPI CS0
GPIO17 UART 2 TX
GPIO16 UART 2 RX
GPIO4 ADC2 CH0 TOUCH0
GPIO2 ADC2 CH2 TOUCH2
GPIO15 ADC2 CH3 TOUCH3 HSPI CS0
GND
3V3
EN
BOOT



Below is a simple python script to test SSD1306 OLED display which makes use of this micropython library code.

test.py

```
from machine import Pin, I2C
from ssd1306 import SSD1306_I2C
from time import sleep

def run_this():
  i2c = I2C(-1, scl=Pin(22), sda=Pin(21))
  oled_width = 128
  oled_height = 64
  oled = SSD1306_I2C(oled_width, oled_height, i2c)

  oled.text('Hello, World 1!', 0, 0)
  oled.text('Hello, World 2!', 0, 10)
  oled.text('Hello, World 3!', 0, 20)
  oled.show()
```

Using Arduino IDE:

- select `Preferences>Additional Board Manager URLs`
- enter `https://raw.githubusercontent.com/espressif/arduino-esp32/gh-pages/package_esp32_index.json.`
- open `Boards Manager` and install esp32 (search for it)

- when selecting board for upload, select `DOIT ESP32 DEVKIT V1`

Note: esp32 flash tool requires python serial module (e.g. `pip install pyserial`)

Board: ESP01S

esp01s_info.txt

```
Upgraded ESP-01S ESP8266 Serial WIFI Wireless Module Wireless
Transceiver
- ESP01S ESP8266-01S (stronger antenna signal compared to esp01)
- 1MB flash (from 512M)
- Integrated WEP, TKIP, AES, and WAPI engines. 802.11 b/g/n
- Wake up and transmit packets in 2ms
- standby power consumption of 1.0mW (DTIM3)

Feature:
- Power down leakage current <10uA
- Integrated low power 32-bit CPU could be used as application
processor

------
|ANT.|
|    |
|    |
|8765|
|1234|
------

1|tx|
2|ch_pd|1-enable
3|rst|0-reset
4|vcc|3.3v
5|rx|
6|gpio0|0-flash mode (def), 1-uart mode (prog)
7|gpio2|gpio
8|gnd|
```