

## Introduction to Arduino IDE and getting started with the ESP32 microcontroller

### Part 3: Summary of the different pins on the microcontroller

Dr Ian Grout  
 Department of Electronic and Computer Engineering  
 Faculty of Science and Engineering  
 University of Limerick  
 Limerick, V94 T9PX  
 Ireland

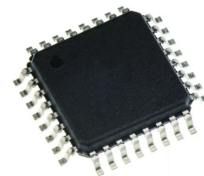
Email: [Ian.Grout@ul.ie](mailto:Ian.Grout@ul.ie)



## Introduction

- Summary of the different pins on the microcontroller:

- Power supply, digital GPIO, analogue I/O, UART/USART, SPI, I<sup>2</sup>C,)
- ... and their uses:



1. Connecting the microcontroller to external peripherals:

1. Power supply.
2. Digital GPIO (General Purpose I/O).
3. Analogue I/O.
4. UART (Universal Asynchronous Receiver/Transmitter) / USART (Universal Synchronous/Asynchronous Receiver/Transmitter).
5. SPI (Serial Peripheral Interface).
6. I<sup>2</sup>C (Inter-Integrated Circuit (IC)).

2. I/O pins on the ESP32:

1. The available I/O pins.
2. Serial communications: using the UART.

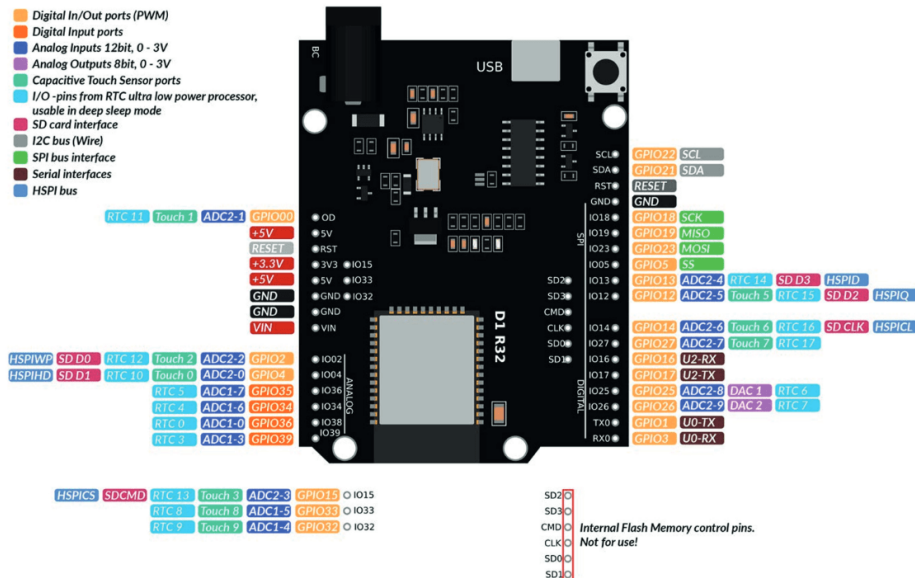


# ESP32 D1 R32 Board Pinout

## D1 R32 Board Pinout

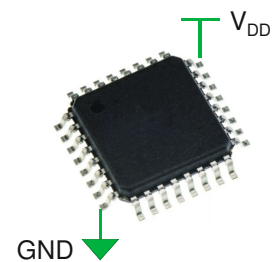
Image source:

[https://bpb-ap-se2.wpmucdn.com/bl-oqs.auckland.ac.nz/digital/9/698/files/2021/08/2\\_Pinout\\_D1\\_R32.png](https://bpb-ap-se2.wpmucdn.com/bl-oqs.auckland.ac.nz/digital/9/698/files/2021/08/2_Pinout_D1_R32.png)



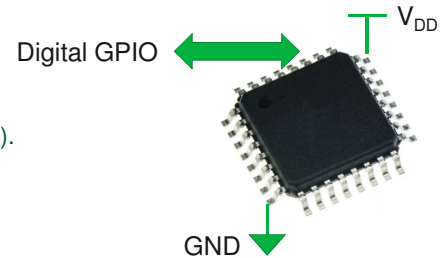
## Power supply

- Always required for an IC to work!
- Most microcontrollers will operate on a standard power supply voltage:
  - +5 V
  - +3.3 V
- Some microcontrollers will operate on a power supply voltage of less than 3.3 V.
- The ESP32 operates on a +3.3 V power supply.
- It is essential to operate the microcontroller on the correct power supply voltage and to connect peripheral devices to the microcontroller pins that operate on the same voltage levels.
- Some ICs have separate power supplies for digital and analogue circuitry.



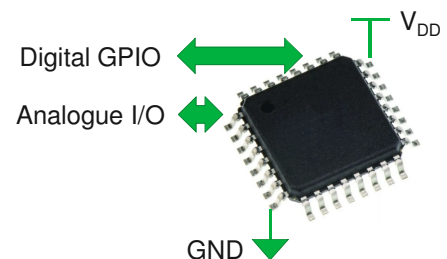
## Digital GPIO

- Digital pins for general use (general purpose).
- Can be input or output (IO) ... or bidirectional (both input and output).
- Creates two logic level outputs:
  - Logic **0** = GND (or 0 V).
  - Logic **1** =  $V_{DD}$  (positive power supply, typically +3.3 V or +5 V).
- In Arduino terminology:
  - Logic **0** = **LOW**.
  - Logic **1** = **HIGH**.



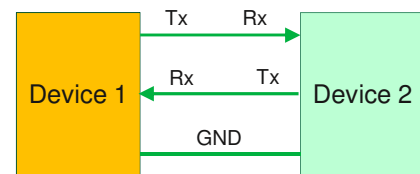
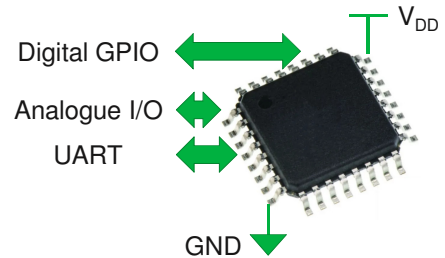
## Analogue I/O

- Provide an analogue voltage output from the microcontroller.
- Analogue voltage output from a microcontroller can be created using:
  1. An in-built Digital to Analogue Converter (DAC) ... within the microcontroller if available.
  2. An external DAC connected to digital pins on the microcontroller.
  3. Using a PWM (Pulse Width Modulation) digital output that is then low-pass filtered using an analogue low-pass filter.



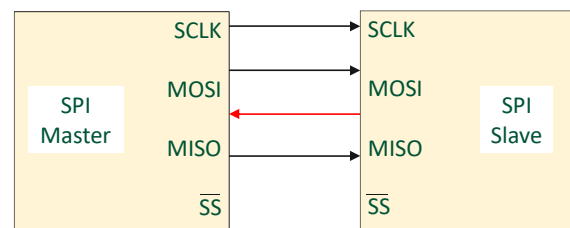
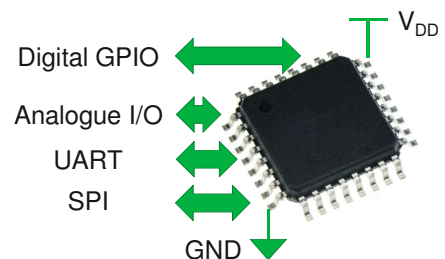
## UART

- **UART** (Universal asynchronous receiver/transmitter):
  - Hardware device.
  - Used for **asynchronous** serial communications.
  - Microcontrollers will typically incorporate one or possibly two UARTs.
  - The two devices communicating with a UART will need to use the same communications data transmission/reception speed (the Baud rate) based on an internal clock frequency.
  - In older computers, the UART would have been used to connect to a RS-232 port (with voltage level translation between the microcontroller and the computer RS-232 port).
  - In computers used today, USB communications is required and the UART would be connected to the computer USB port using a UART-to-USB converter IC.
- **USART** (Universal Synchronous/Asynchronous Receiver/Transmitter):
  - Hardware device.
  - Used for **synchronous** serial communications.
  - The data transmission/reception clock frequency is embedded within the signal.



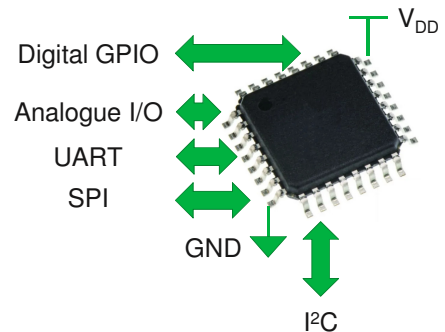
## SPI

- Serial Peripheral Interface (SPI) is an interface bus commonly used to communicate data and instructions between microcontrollers and small peripheral devices (Integrated Circuits (ICs)) in embedded systems.
- A synchronous serial communication interface specification used for short-distance communication, primarily in embedded systems.
- The interface has a master device and one or more slave devices. Four signals involved:
  - **SCLK**
    - Serial Clock (output from the master).
  - **MOSI**
    - Master Out Slave In (data output from the master).
  - **MISO**
    - Master In Slave Out (data output from the slave).
  - **SS**
    - Slave Select (often active low, output from the master).

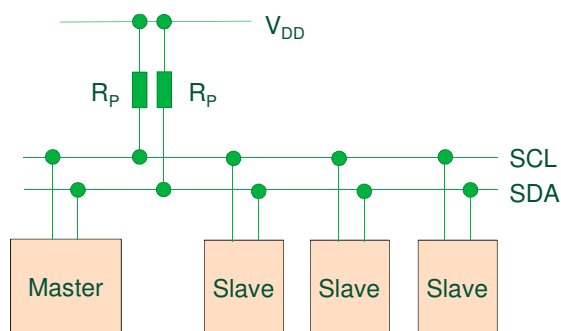


## I<sup>2</sup>C (1)

- I<sup>2</sup>C stands for Inter-IC bus. Developed by Philips.
- IC – integrated circuit.
- Developed to connect microprocessors and other ICs on a PCB (printed circuit board).
- Serial communications that connects devices with two wires that would otherwise have been connected using parallel communications -> saves on device pins, interconnects, and PCB area.
- Requires two wires:
  1. **SDA** – Serial Data.
  2. **SCL** – Serial Clock.
- All devices share these two wires. Each device has an exclusive, unique address.



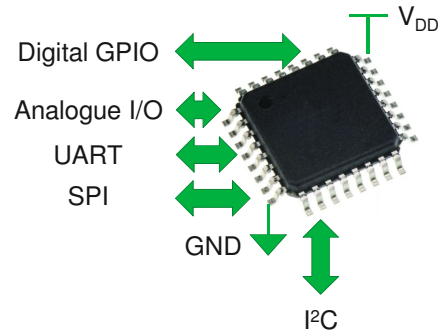
## I<sup>2</sup>C (2)



- Simple to use.
- There can be more than one master.
- Each device has a 7-bit address, so a single I<sup>2</sup>C network can theoretically support up to 128 devices.
- Only an upper bus speed is defined.
- Only two wires with pull-up resistors are required to connect the I<sup>2</sup>C devices on an I<sup>2</sup>C network.

## Pins with multiple possible uses

- The pins on the microcontroller used for digital and analogue I/O can be:
  - Dedicated to a single use.
  - Programmable so that the pin can be used for one of two or more possible uses.



## D1 R32 Board Pinout

# ESP32

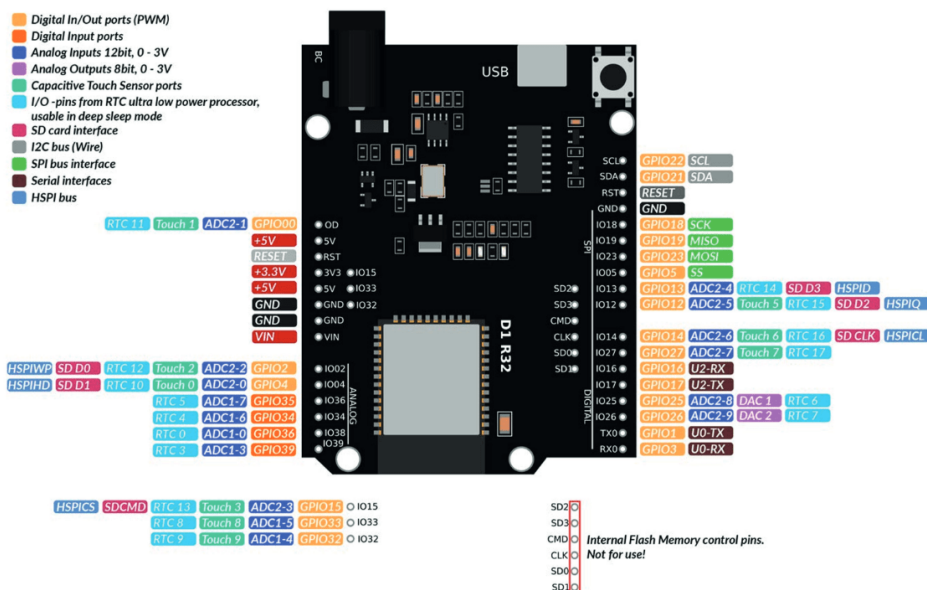


Image source:

[https://bpb-ap-se2.wpmucdn.com/blogs.auckland.ac.nz/dist/9/698/files/2021/08/2\\_Pinout\\_D1\\_R32.png](https://bpb-ap-se2.wpmucdn.com/blogs.auckland.ac.nz/dist/9/698/files/2021/08/2_Pinout_D1_R32.png)

**Any questions?**



University of Limerick,  
Limerick, V94 T9PX,  
Ireland.  
Oillscoil Luimnigh,  
Luimneach,  
V94 T9PX, Éire.  
+353 (0) 61 202020

[ul.ie](http://ul.ie)