UART Communication (TXD0, RXD0, TXD2, RXD2 Pins)

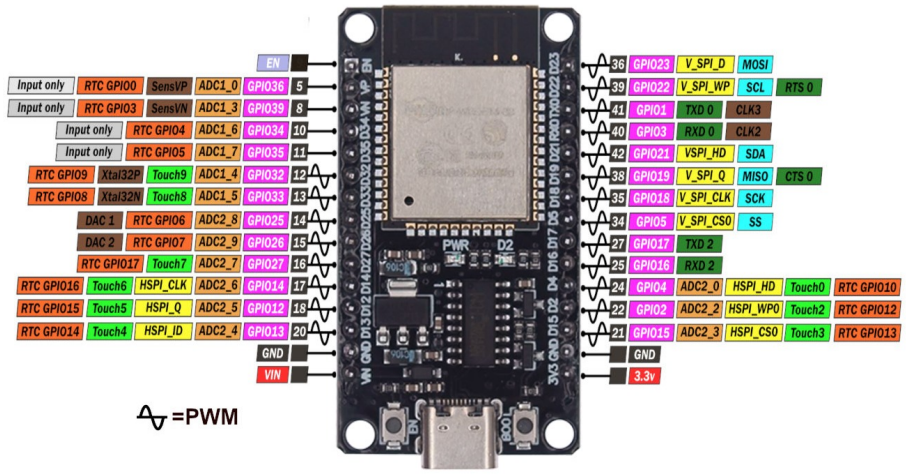

Figure 1: **ESP32 UART Communication Pins**

The ESP32 supports multiple **UART (Universal Asynchronous Receiver/Transmitter)** interfaces for serial communication. On the ESP32-WROOM, the default communication interface with the computer (via USB) uses **TXD0 (GPIO1)** and **RXD0 (GPIO3)**, while another commonly used interface is **TXD2 (GPIO17)** and **RXD2 (GPIO16)**.

What are TXD and RXD Pins?

- **TXD (Transmit Data):** Sends serial data out of the ESP32.
- **RXD (Receive Data):** Receives serial data into the ESP32.

Default UART Pins

- **TXD0 (GPIO1)** – Transmit pin used for programming/debugging via USB.
- **RXD0 (GPIO3)** – Receive pin used for programming/debugging via USB.
- These are connected internally to the USB-to-UART bridge chip, enabling uploading of code and printing logs with `Serial.print()`.

Additional UART Pins

- **TXD2 (GPIO17)** – Transmit pin of UART2, available for user applications.

- **RXD2 (GPIO16)** – Receive pin of UART2, available for user applications.
- Commonly used to connect external devices such as:
  - GPS modules
  - GSM/GPRS modules
  - Bluetooth modules (HC-05, HC-06)
  - Communication with other microcontrollers

ESP32 UART Features

- The ESP32 has **3 UART controllers**:
  - **UART0:** TXD0 (GPIO1), RXD0 (GPIO3) – default programming/debugging.
  - **UART1:** User-configurable pins (no fixed default).
  - **UART2:** TXD2 (GPIO17), RXD2 (GPIO16) – available for peripherals.
- Baud rates supported: up to **5 Mbps**.
- UART pins can be re-mapped to different GPIOs using software if needed.

Example Usage

When using `Serial.begin(115200);` in Arduino IDE, the ESP32 communicates through **TXD0 (GPIO1)** and **RXD0 (GPIO3)**.
If you want to communicate with an external module like a GPS via UART2, you can use:

```
HardwareSerial mySerial(2);
mySerial.begin(9600, SERIAL_8N1, 16, 17); // RX=16, TX=17
```

This initializes UART2 using **RXD2 (GPIO16)** and **TXD2 (GPIO17)** for serial communication.