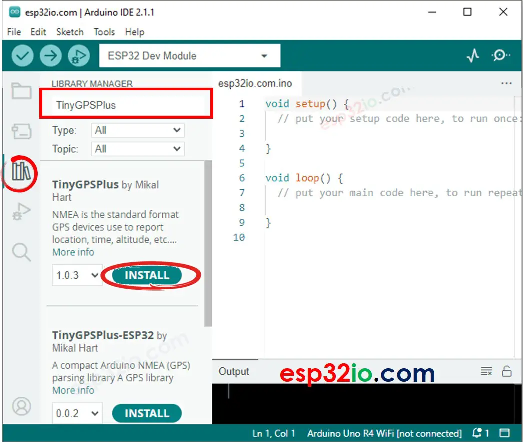
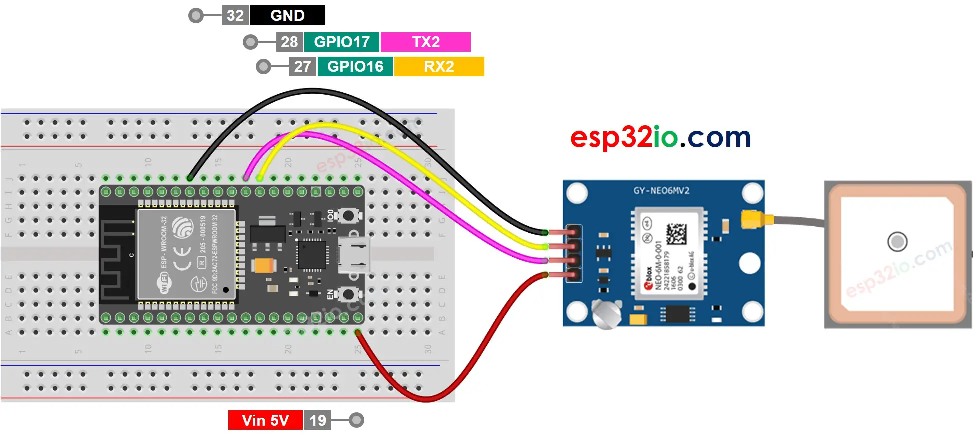
4.GPS module interfacing with ESP 32 Microcontroller(Practical)

**Required Components**

* One ESP32 development boards
* One NEO-6M GPS Module
* Jumper Wires
* Breadboard

Install library: <https://github.com/Tinyu-Zhao/TinyGPSPlus-ESP32>



Circuit Connection Diagram:

Source code:

#include <TinyGPS++.h>

#define GPS\_BAUDRATE 9600  // The default baudrate of NEO-6M is 9600

TinyGPSPlus gps;  // the TinyGPS++ object

void setup() {

  Serial.begin(9600);

  Serial2.begin(GPS\_BAUDRATE);

  Serial.println(F("ESP32 - GPS module"));

}

void loop() {

  if (Serial2.available() > 0) {

    if (gps.encode(Serial2.read())) {

      if (gps.location.isValid()) {

        Serial.print(F("- latitude: "));

        Serial.println(gps.location.lat());

        Serial.print(F("- longitude: "));

        Serial.println(gps.location.lng());

        Serial.print(F("- altitude: "));

        if (gps.altitude.isValid())

          Serial.println(gps.altitude.meters());

        else

          Serial.println(F("INVALID"));

      } else {

        Serial.println(F("- location: INVALID"));

      }

      Serial.print(F("- speed: "));

      if (gps.speed.isValid()) {

        Serial.print(gps.speed.kmph());

        Serial.println(F(" km/h"));

      } else {

        Serial.println(F("INVALID"));

      }

      Serial.print(F("- GPS date&time: "));

      if (gps.date.isValid() && gps.time.isValid()) {

        Serial.print(gps.date.year());

        Serial.print(F("-"));

        Serial.print(gps.date.month());

        Serial.print(F("-"));

        Serial.print(gps.date.day());

        Serial.print(F(" "));

        Serial.print(gps.time.hour());

        Serial.print(F(":"));

        Serial.print(gps.time.minute());

        Serial.print(F(":"));

        Serial.println(gps.time.second());

      } else {

        Serial.println(F("INVALID"));

      }

      Serial.println();

    }

  }

  if (millis() > 5000 && gps.charsProcessed() < 10)

    Serial.println(F("No GPS data received: check wiring"));

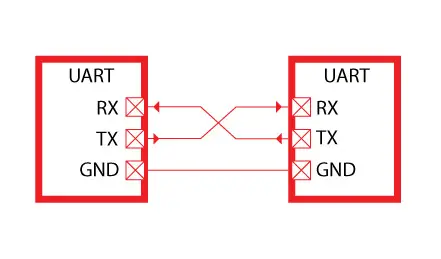
}

4.Serial Communication between Two ESP 32(Skilling)

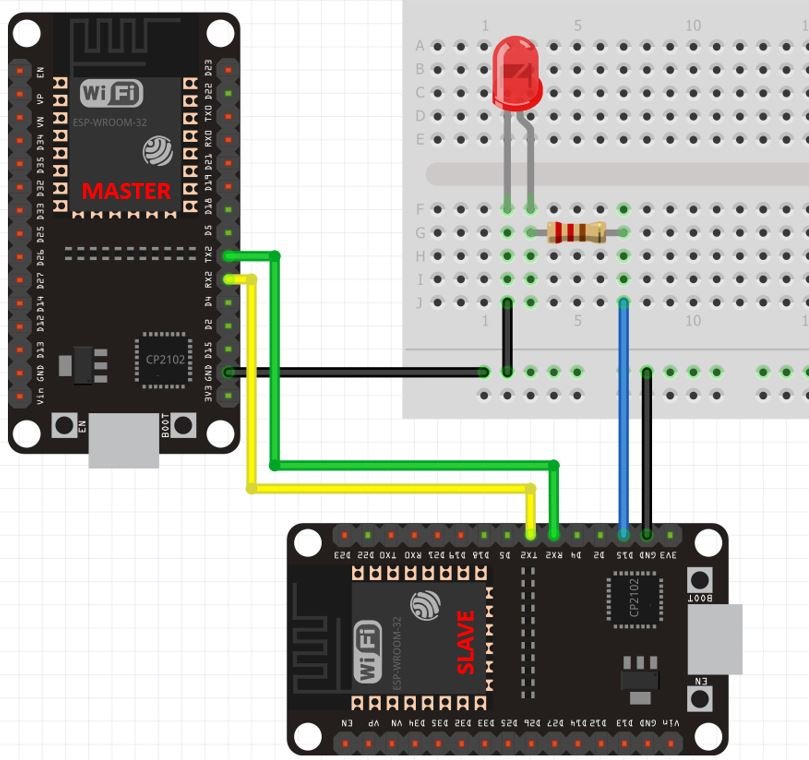
**Required Components**

* Two ESP32 development boards
* One 5mm LED
* One 220 ohm resistor
* Connecting Wires
* Breadboard

Copy Path: cores/esp32/HardwareSerial.h



Circuit connection diagram:



Source code:

### Master ESP32 Arduino Sketch

#include <HardwareSerial.h>

HardwareSerial SerialPort(2); // use UART2

void setup()

{

SerialPort.begin(15200, SERIAL\_8N1, 16, 17);

}

void loop()

{

SerialPort.print(1);

delay(5000);

SerialPort.print(0);

delay(5000);

}

### Slave ESP32

#include <HardwareSerial.h>

HardwareSerial SerialPort(2); // use UART2

char number = ' ';

int LED = 15;

void setup()

{

SerialPort.begin(15200, SERIAL\_8N1, 16, 17);

pinMode(LED, OUTPUT);

}

void loop()

{

if (SerialPort.available())

{

char number = SerialPort.read();

if (number == '0') {

digitalWrite(LED, LOW);

}

if (number == '1') {

digitalWrite(LED, HIGH);

}

}

}