

08:26

VoLTE 54%



# Terminal



```

08:26:30.449
08:26:30.451 $NGA,3(02(0p(0z(1p(1r,23,30,30,34,
,(p.8v(011h0.46,7*00
08:26:30.452 GN$S(Ah7,r1,r3,34,41,47,48,73h74h77h
01h42(43(0.83,0.51,p.6t(4l07
08:26:30.452
08:26:30.453 $NGA,3(,
(h,h,h(0l83(0.31,0.64,5.0M
08:26:30.455 `GP$SV(3,3,0,,04,38,286h79h08h79(214,3
4,10,48,012h03h14h10(327,20,1,6F
08:26:30.458 @GP$WVh7,t(0:,14,03,185,61h61h(v6,t3,3
1,047(37(27(5p(1p7,v9,r*5;
08:26:30.463 GP$SV(3,7,p9,v6,v1,r06(30(1*3E^M^L$G
$G$V;t(1h06h08(79(214,40,10,p8,p12h70h14(10(327,2
1,27,r1,p07h37(8*45^M^L$G$GS,2(2h06h67h10(147,
34,34,31,106h72h8*t@^M^L$G$GS,2(1,08H65h07h18
p(26,73,44,033,72h72h12h13z,34,77,13,200,14h1*v7^M^
L$G$GS,2(2,08,78,61h69z(2z(8t,34,087,78,87,64h00v(
3v(8x,24,302,40,3*77^M^L$G$GS(3h1,09,02,74,429(2x
(0p(2r(028,05,09,72,114(1z(1p(0p(083,40,7*7;
08:26:30.465 GAGS(3h6,p9,21,49,076(35(2r(0r(1x6,t2,
60,72,494(24(3p(3z(2x1,t9,7*7
08:26:30.466 GA$SV(3h7,p9,v6,v2,020(28(7*02^M^N$
$AG$V,v(1h09(02(34(22:,36,h76h72h020,34,1,4F^K
08:26:30.468 $B$WVh0,r(1v,03,,
(29(11(31(1p7,v1,r7,r6,048(39(14(61(1r7,v0,r.4x
08:26:30.472 GB$SV(4,4,17,t1,p8,p75h22(27(12(297,27
,t8,r7,v06h34(33(49(163,30,r.7t
08:26:30.473 `GB$SV(4,7,17,30,08,r15h19h37(08(258,2
0,41,08,r97h62h02(50(068,23,1,78^
08:26:30.478 `GB$WVh0,0,17,47,50,109H69h1*p1^M^L
$G$GS,3(1,09,61h08h03r(2t,27,14,2;7,49,68h17h70t(3
p,37,4,,145,76,1*v1^M^L$G$GS,3(2,09,74,08,11r(2v(3v(
0x,238,47,01,08,19v(2p(4t(5p,048,74,3*79
08:26:30.481 G$GS(3h3,09,03,34,309(30(5l06

```

M1 M2 M3 M4 M5 M6



```

#include <BluetoothSerial.h>

// Creazione dell'istanza Bluetooth
BluetoothSerial SerialBT;

// Creazione dell'istanza per la comunicazione seriale con il GPS
HardwareSerial GPS(1); // Serial1 (TX=GPIO17, RX=GPIO16)

String gpsBuffer = ""; // Buffer per memorizzare i dati NMEA

void setup() {
  // Inizializzazione della seriale connessa al GPS (velocità 460800 baud)
  GPS.begin(460800, SERIAL_8N1, 16, 17); // RX=GPIO16, TX=GPIO17

  // Inizializzazione del Bluetooth con il nome del dispositivo
  if (!SerialBT.begin("ESP32_GPS_Server")) {
    // Se si verifica un errore, blocca il programma
    while (1);
  }
}

void loop() {
  // Ricezione dei dati dal GPS carattere per carattere
  while (GPS.available()) {
    char c = GPS.read();
    gpsBuffer += c;

    // Verifica se una stringa NMEA è completa (finisce con '\n')
    if (c == '\n') {
      // Invio della stringa NMEA completa tramite Bluetooth
      SerialBT.print(gpsBuffer);

      // Svuotamento del buffer per la prossima stringa NMEA
      gpsBuffer = "";
    }
  }

  // Ricezione dei dati via Bluetooth
  if (SerialBT.available()) {
    String btData = "";
    while (SerialBT.available()) {
      btData += (char)SerialBT.read();
    }

    // Invio dei dati ricevuti al GPS
    GPS.print(btData);
  }
}

```