

Serial

Hardware Serial

Đối tượng `Serial` làm việc giống nhiều với Arduino bình thường. Ngoại trừ việc phần cứng của ESP8266 có thêm 128 bytes RAM FIFO và 256 bytes RX-TX Buffer. Cả TX và RX đều truyền nhận dựa vào interrupt. Việc đọc và ghi dữ liệu chỉ bị block lại khi FIFO/buffer đầy/rỗng.

`Serial` sử dụng là UART0, được map thẳng vào chân GPIO1 (TX) và GPIO3 (RX). Serial có thể REMAP lại vào GPIO15 (TX) và GPIO13 (RX) bởi việc gọi hàm `Serial.swap()` sau khi gọi `Serial.begin`. Gọi `swap` lại sẽ MAP UART0 trở lại GPIO1 và GPIO3.

`Serial1` sử dụng UART1, TX pin là chân GPIO2. UART1 không thể sử dụng để nhận dữ liệu bởi vì bình thường nó được sử dụng để kết nối với Flash. Để sử dụng `Serial1`, gọi `Serial1.begin(baudrate)`.

! Ghi chú

Nếu `Serial1` không được sử dụng và `Serial` không bị REMAP - TX cho UART0 có thể MAP sang GPIO2 bởi gọi hàm `Serial.set_tx(2)` sau `Serial.begin` hay trực tiếp với `Serial.begin(baud, config, mode, 2)`.

Mặc định, tất cả thông tin chuẩn đoán hệ thống và thư viện sẽ bị bỏ qua nếu gọi hàm `Serial.begin`. Để cho phép những thông tin đó, có thể gọi `Serial.setDebugOutput(true)`. Để chuyển thông tin đó ra `Serial1`, gọi hàm `Serial1.setDebugOutput(true)`.

! Ghi chú

Bạn cũng cần phải gọi `Serial.setDebugOutput(true)` để cho phép hàm `printf()`.

Cả đối tượng `Serial` và `Serial1` hỗ trợ 5, 6, 7, 8 bits dữ liệu, odd (O), even (E), và no (N) parity, và 1 hay 2 bits stop. Để cấu hình các mode trên, gọi `Serial.begin(baudrate, SERIAL_8N1)`, `Serial.begin(baudrate, SERIAL_6E2)`, v.v..

Một phương thức được hiện thực trên cả 2 `Serial` và `Serial1` để lấy baud rate hiện tại như sau:
`Serial.baudRate()`, `Serial1.baudRate()` trả về một số `int` của tốc độ Baud hiện tại.

Ví dụ

```
// Set Baud rate 57600
Serial.begin(57600);

// Kiểm tra baud rate hiện tại
int br = Serial.baudRate();

// sẽ xuất ra "Serial is 57600 bps"
Serial.printf("Serial is %d bps", br);
```

Software Serial

Ngoài ra, các Contributor Team ESP8266 Arduino cũng hoàn thiện thư viện [Software Serial](#) cho ESP8266, xem [pull request](#).

⚠ Cảnh báo

Thư viện này được thực hiện chỉ **duy nhất cho ESP8266 boards**, và sẽ không làm việc với các thư viện Arduino khác.