

# Arduino SoftwareSerial Library

---

## Limitations

The library has the following known limitations:

- 如果使用多个串口软件，一次只有一个软件可以接收数据。
- 在 Mega 和 Mega 2560 上，不是所有的引脚都支持中断，允许用于RX的引脚包括：10, 11, 12, 13, 14, 15, 50, 51, 52, 53, A8 (62), A9 (63), A10 (64), A11 (65), A12 (66), A13 (67), A14 (68), A15 (69).
- 在 Leonardo和 Micro support 上，不是所有的引脚都支持中断，允许用于RX的引脚包括：10, 11, 12, 13, 14, 15, 50, 51, 52, 53, A8 (62), A9 (63), A10 (64), A11 (65), A12 (66), A13 (67), A14 (68), A15 (69).
- 在 Arduino 或 Genuino 101 上，当前最大 RX 速度为 57600bps
- 在 Arduino 或 Genuino 101 上，引脚13不被允许用于RX

如果您的项目需要同步数据流，请参阅 Paul Stoffregen 的 [AltSoftSerial library](#). AltSoftSerial 克服了核心 SoftwareSerial 的许多其他问题，但有其自身的局限性。有关详细信息，请参阅 [AltSoftSerial site](#)。

## 使用场景

- 软件串行示例：有时一个串行端口是不够的！
- 二端口接收：与多个软件串口配合使用。

## 函数

---

### SoftwareSerial(rxPin, txPin, inverse\_logic)

#### 说明

调用SoftwareSerial(rxPin, txPin)来创建一个新的软件串行对象，正如下面的例子，其名称需要您自己创建。您需要调用 SoftwareSerial.begin()来进行通讯。

#### 参数

rxPin：接受串行数据的引脚

txPin：传输串行数据的引脚

inverse\_logic：用于反转传入位的意义（默认为正常逻辑）。如果设置，SoftwareSerial 将 Rx 引脚上的 LOW （引脚上通常为 0 伏）视为 1 位（空闲状态），将 HIGH （引脚上通常为 5 伏）视为 0 位。它还会影响写入 Tx 引脚的方式。默认值为假。

## Warning:

您不应该连接输出串行数据超出 Arduino 可以处理的范围的设备，通常为 0V 到 5V，对于以 5V 运行的电路板，对于以 3.3V 运行的电路板，输出为 0V 到 3.3V。

## Example

```
1 #include <SoftwareSerial.h>
2
3 const byte rxPin = 2;
4 const byte txPin = 3;
5
6 // set up a new serial object
7 SoftwareSerial mySerial (rxPin, txPin);
```

## SoftwareSerial:available()

### 说明

获取字节数（字符），可用于读取软串行端口。读取已经到达并存储在串行接收缓冲区的数据。

### 语法

mySerial.available()

### 参数

无

### 返回

可读取的字节数

```
1 // include the SoftwareSerial library so you can use its functions:
2 #include <SoftwareSerial.h>
3
4 #define rxPin 10
5 #define txPin 11
6
7 // set up a new serial port
8 SoftwareSerial mySerial = SoftwareSerial(rxPin, txPin);
9
10 void setup()
11 {
12     // define pin modes for tx, rx:
13     pinMode(rxPin, INPUT);
14     pinMode(txPin, OUTPUT);
15     // set the data rate for the SoftwareSerial port
16     mySerial.begin(9600);
17 }
18
19 void loop()
20 {
21     if (mySerial.available()>0)
22     {
```

```
23     mySerial.read();
24   }
25 }
```

## SoftwareSerial:begin(speed)

---

### 说明

设置串行通信速度（波特率）。支持的波特率有300, 1200, 2400, 4800, 9600, 14400, 19200, 28800, 31250, 38400, 57600和115200。

### 参数

speed: 波特率(长)

### 返回

无

### Example

```
1  // include the SoftwareSerial library so you can use its functions:
2  #include <SoftwareSerial.h>
3
4  #define rxPin 10
5  #define txPin 11
6
7  // set up a new serial port
8  SoftwareSerial mySerial = SoftwareSerial(rxPin, txPin);
9
10 void setup() {
11   // define pin modes for tx, rx:
12   pinMode(rxPin, INPUT);
13   pinMode(txPin, OUTPUT);
14   // set the data rate for the SoftwareSerial port
15   mySerial.begin(9600);
16 }
17
18 void loop() {
19   // ...
20 }
```

## SoftwareSerial:isListening()

---

### 说明

测试软件串口是否正在听取数值。

## 语法

mySerial.isListening()

## 参数

无

## 返回

布尔类型变量(boolean)

## Example

```
1  #include <SoftwareSerial.h>
2
3  // software serial : TX = digital pin 10, RX = digital pin 11
4  SoftwareSerial portOne(10,11);
5
6  void setup()
7  {
8      // Start the hardware serial port
9      Serial.begin(9600);
10
11     // Start software serial port
12     portOne.begin(9600);
13 }
14
15 void loop()
16 {
17     if (portOne.isListening()) {
18         Serial.println("Port One is listening!");
19     }
```

## SoftwareSerial:overflow()

---

### 说明

测试软件串行已缓冲区是否溢出。调用此函数清除溢出标志，这意味着它将返回false，除非在此期间另字节的数据已收到并丢弃。

软件串行缓冲器可容纳64个字节的数据。

### 语法

mySerial.overflow()

### 参数

无

## 返回

布尔类型变量

### Example

```
1  #include <SoftwareSerial.h>
2
3  // software serial : TX = digital pin 10, RX = digital pin 11
4  SoftwareSerial portOne(10,11);
5
6  void setup()
7  {
8      // Start the hardware serial port
9      Serial.begin(9600);
10
11     // Start software serial port
12     portOne.begin(9600);
13 }
14
15 void loop()
16 {
17     if (portOne.overflow()) {
18         Serial.println("SoftwareSerial overflow!");
19     }
```

## SoftwareSerial: Peek()

### 说明

返回软件串口RX引脚上接收到的字符。不像 read(), 当然, 随后调用这个函数会返回相同的字符。

请注意, 在同一时间只有一个串口可以接收输入的数据 (与 listen()函数二选一) 。

### 参数

无

### 返回

读取到的字符, 如果无变量输入则返回-1

### Example

```
1  SoftwareSerial mySerial(10,11);
2
3  void setup()
4  {
5      mySerial.begin(9600);
6  }
7
8  void loop()
9  {
10     char c = mySerial.peek();
11 }
```

## SoftwareSerial:int read()

---

### 说明

返回软件串口RX引脚上接收到的字符。请注意，在同一时间只有一个串口可以接收输入的数据（与listen() 函数二选一）。

### 参数

无

### 返回

读取到的字符，如果无变量输入时返回-1

### Example

```
1  SoftwareSerial mySerial(10,11);
2
3  void setup()
4  {
5      mySerial.begin(9600);
6  }
7
8  void loop()
9  {
10     char c = mySerial.read();
11 }
```

## SoftwareSerial:print(data)

---

### 说明

打印由引脚发送到软件串口的数据。和函数 serial.print ()的功能相同。

### 参数

多种多样，详情请参考 Serial.print

### 返回

byte print()将返回写入数据的字节数，虽然读取那个数组的可选择的

### Example

```
1  SoftwareSerial serial(10,11);
2  int analogValue;
3
4  void setup()
5  {
6      serial.begin(9600);
7  }
8
```

```

9 void loop()
10 {
11     // read the analog input on pin 0:
12     analogValue = analogRead(A0);
13
14     // print it out in many formats:
15     serial.print(analogValue);           // print as an ASCII-encoded decimal
16     serial.print("\t");                  // print a tab character
17     serial.print(analogValue, DEC);      // print as an ASCII-encoded decimal
18     serial.print("\t");                  // print a tab character
19     serial.print(analogValue, HEX);      // print as an ASCII-encoded
hexadecimal
20     serial.print("\t");                  // print a tab character
21     serial.print(analogValue, OCT);      // print as an ASCII-encoded octal
22     serial.print("\t");                  // print a tab character
23     serial.print(analogValue, BIN);      // print as an ASCII-encoded binary
24     serial.print("\t");                  // print a tab character
25     serial.print(analogValue/4, BYTE);   // print as a raw byte value (divide
the
26                                         // value by 4 because analogRead()
returns numbers
27                                         // from 0 to 1023, but a byte can only
hold values
28                                         // up to 255)
29     serial.print("\t");                  // print a tab character
30     serial.println();                    // print a linefeed character
31
32     // delay 10 milliseconds before the next reading:
33     delay(10);
34 }

```

## SoftwareSerial:println(data)

### 说明

打印引脚发送到软件串口的数据以及换行符。和函数serial.println()的功能相同。

### 参数

有多种参数，详情请参考 Serial.println()

### 返回

byte

println()将返回写入数据的字节数，虽然读取那个数组的可选择的

### Example

```

1 SoftwareSerial serial(10,11);
2 int analogValue;
3
4 void setup()
5 {
6     serial.begin(9600);
7 }

```

```

8
9 void loop()
10 {
11     // read the analog input on pin 0:
12     analogValue = analogRead(A0);
13
14     // print it out in many formats:
15     serial.print(analogValue);           // print as an ASCII-encoded decimal
16     serial.print("\t");                  // print a tab character
17     serial.print(analogValue, DEC);      // print as an ASCII-encoded decimal
18     serial.print("\t");                  // print a tab character
19     serial.print(analogValue, HEX);      // print as an ASCII-encoded
hexadecimal
20     serial.print("\t");                  // print a tab character
21     serial.print(analogValue, OCT);      // print as an ASCII-encoded octal
22     serial.print("\t");                  // print a tab character
23     serial.print(analogValue, BIN);      // print as an ASCII-encoded binary
24     serial.print("\t");                  // print a tab character
25     serial.print(analogValue/4, BYTE);   // print as a raw byte value (divide
the
26                                         // value by 4 because analogRead()
returns numbers
27                                         // from 0 to 1023, but a byte can only
hold values
28                                         // up to 255)
29     serial.print("\t");                  // print a tab character
30     serial.println();                    // print a linefeed character
31
32     // delay 10 milliseconds before the next reading:
33     delay(10);
34 }

```

## SoftwareSerial:listen()

### 说明

能够选择软件串口进行听取数据。每次只有一个软件串口可以听取数据，到达其他串口的数据将被丢弃。在调用 listen() 期间，任何数据将会被丢弃。(除非被给予的实例已经在听取数据)。

### 语法

mySerial.listen()

### 参数

mySerial：听取的实例的名称

### 返回

无



## Example

```
1  #include <SoftwareSerial.h>
2
3  // software serial : TX = digital pin 10, RX = digital pin 11
4  SoftwareSerial portOne(10, 11);
5
6  // software serial : TX = digital pin 8, RX = digital pin 9
7  SoftwareSerial portTwo(8, 9);
8
9  void setup()
10 {
11     // Start the hardware serial port
12     Serial.begin(9600);
13
14     // Start both software serial ports
15     portOne.begin(9600);
16     portTwo.begin(9600);
17
18 }
19
20 void loop()
21 {
22     portOne.listen();
23
24     if (portOne.isListening())
25     {
26         Serial.println("Port One is listening!");
27     }
28     else
29     {
30         Serial.println("Port One is not listening!");
31     }
32
33     if (portTwo.isListening())
34     {
35         Serial.println("Port Two is listening!");
36     }
37     else
38     {
39         Serial.println("Port Two is not listening!");
40     }
41 }
42
```

## SoftwareSerial:write(data)

### 说明

将引脚的数据作为原始数据传送到软件串口打印出来。和函数 `Serial.write ()` 的功能相同。

## 参数

详情请参考 `serial.write ()`

## 返回

byte

`write()`将返回写入数据的字节数，虽然读取那个数组的可选择的

## Example

```
1  SoftwareSerial mySerial(10, 11);
2
3  void setup()
4  {
5      mySerial.begin(9600);
6  }
7
8  void loop()
9  {
10     mySerial.write(45); // send a byte with the value 45
11
12     int bytesSent = mySerial.write("hello"); //send the string "hello" and
        return the                               //length of the
        string.
13 }
14
```

Reference:

1. [Arduino - SoftwareSerial](#)
2. [arduino:libraries [创客百科](#)]([nxez.com](#))
- 3.