

Arduino编程之Serial.println（）和Serial.print（）

Arduino编程之Serial.println()和Serial.print()

Arduino 的输出基本就用两个函数 print 和 println， 区别在于后者比前者多了回车换行

Serial.println(data)

从串行端口输出数据， 跟随一个回车（ASCII 13, 或 'r'） 和一个换行符（ASCII 10, 或 'n'）。这个函数所取得的值与 Serial.print()一样。

Serial.println(b) 以十进制形式输出b的ASCII编码值， 并同时跟随一个回车和换行符。

Serial.println(b, DEC) 以十进制形式输出b的ASCII编码值， 并同时跟随一个回车和换行符。

Serial.println(b, HEX) 以十六进数据形式输出b的ASCII编码值， 并同时跟随一个回车和换行符。

Serial.println(b, OCT)以八进数据形式输出b的ASCII编码值， 并同时跟随一个回车和换行符。

Serial.println(b, BIN)以二进数据形式输出b的ASCII编码值， 并同时跟随一个回车和换行符。

Serial.print(b, BYTE)以单个字节输出b， 并同时跟随一个回车和换行符。

Serial.println(str)如果 str是一个字符串或数组， 输出整个 str的 ASCII编码字符串。

Serial.println()仅输出一个回车和换行符。

参数

b:需要输出的字节。

str:需要输出的字符串。

Returns

None

```
/*
模拟输入
在0号模拟插口读取一个模拟值， 并输出。
created 24 March 2006 by Tom Igoe
*/
int analogValue = 0;  // 需要保留的模拟值


void setup() {
//在9600 bps打开串行端口:
Serial.begin(9600);
}
void loop() {
// 在0号模拟输入插口读取值
analogValue = analogRead(0);
// 以多种格式输出
Serial.println(analogValue);
// 以ASCII编码十进制浮点值输出
Serial.print(analogValue, DEC);
// 以ASCII编码十进制浮点值输出
Serial.println(analogValue, HEX);
// 以ASCII编码十六进制输出
Serial.println(analogValue, OCT);
// 以ASCII编码八进制输出
Serial.println(analogValue, BIN);
// 以ASCII编码二进制输出
Serial.println(analogValue, BYTE);
// 以原始数据输出
// 在下次读取之前延迟10毫秒
delay(10);
}
```