

简介.....	1
1 Uno 控制 wifiIO gpio 闪烁.....	2
1.1 Uno 与 wifiIO 接线.....	2
1.2 配置 wifiIO.....	2
1.3 编程实现.....	2
2 Uno 从 wifiIO 获取网络数据.....	3
2.1 Uno 与 wifiIO 接线.....	3
2.2 用 Uno 配置 wifiIO.....	4
2.3 编程实现.....	4
2.4 操作演示.....	5
3 其他.....	7

简介

本文将介绍如何使用 `arduino` 控制 `wifiIO`。其实用 `arduino` 控制 `wifiIO` 的原理就是让 `arduino` 和 `wifiIO` 串口通信，`wifiIO` 收到数据后做出相应的处理。本文将着重从以下几个方面介绍：

- 1、通过 `Uno` 控制 `wifiIO` 的 `gpio` 闪烁
- 2、`Uno` 从 `wifiIO` 获取网络数据。
- 3、其他

1 Uno 控制 wifiIO gpio 闪烁

用 Uno 控制 wifiIO 的 gpio 操作很简单，就是 Uno 发送串口 AT 指令给 wifiIO，wifiIO 收到指令后作出相应的响应。

1.1 Uno 与 wifiIO 接线

把 Uno 的 Serial 的 Rx 与 wifiIO 的 Tx 相连接，把 Uno 的 Serial 的 Tx 与 wifiIO 的 Rx 相连接。把 Uno 的 GND 和 wifiIO 的 GND 相连接，即共地。

值得注意的时，应该在 arduino IDE 中把 Uno 的串口波特率设置为 115200，这样 Uno 的波特率就和 wifiIO 的一致了。如果不知道当前 wifiIO 的波特率可以通过 wifiIO_IoTManager 发送 AT+BAUDRATE=?就可以查询得到。

1.2 配置 wifiIO

控制 wifiIO 的 D0 口闪烁，只需要两步：

第一步：发送+++使 wifiIO 进入串口 AT 指令模式。

第二步：周期性的发送 AT 指令拉高和拉低 D0 口即可，而用到的 AT 指令就是 AT+GPIO0=1(拉高)和 AT+GPIO0=0(拉低)。AT 指令说明请参考《wifiIO AT 指令一览表》。

1.3 编程实现

下面是 Uno 的完整例程：

```
/*-----分割线-----*/
/*
 * @description:
 * This demo shows how to control the wifiIO through arduino such as Uno, Bluno and so on.
 * * @author: jansion
 * @date: 2015.11.16
 * @version: 1.0.0
 * @note:
 * @detail description:
 * when the program is running, the D0 of wifiIO will last high state for 2s and low state for 2s.
 *
 */
//define a buffer name info for receiving the data for wifiIO.
const short info_size = 256;
char info[info_size];
```

```

void setup() {
    //set Serial for communication with the wifiIO
    Serial.begin(115200);
    delay(300);
    //send the AT command +++ to wifiIO for force the wifiIO into AT command mode
    Serial.print("+++");
    //delay 1s for wifiIO dealing
    delay(1000);

}

void loop() {

    //pull up the pin of wifiIO D0 to high.
    Serial.println("AT+GPIO0=1");

    //last high state for 2s
    delay(2000);
    //draw down the pin of wifiIO D0 to low.
    Serial.println("AT+GPIO0=0");
    //last low state for 2s
    delay(2000);

}
/*-----分割线-----*/

```

2 Uno 从 wifiIO 获取网络数据

本节将介绍 Uno 如何从 wifiIo 获取网络数据，并把数据发送到网络。我们将通过以下几点去实现：

- 第一步搭建软件硬件环境()
- 第二步配置 wifiIO 的数据通道
- 第三步编程实现

2.1 Uno 与 wifiIO 接线

Uno 与 wifiIO 的连线可以参见如 1.1 节的描述。当 Uno 和 wifiIO 通过串口连接后，我们需要让 wifiIO_IotManager 和 wifiIO 建立通信，至于如何使这两者建立通信，可以参见《如何使用 wifiIO_IoTManager 控制 wifiIO》的。

2.2 用 Uno 配置 wifiIO

我们需要达到的配置效果是：Uno 可以通过 uart 从 wifiIO 获取到 wifiIO_IotManager 发送过来的数据；Uno 也可以通过 uart 发送数据到 wifiIO_IotManager。如此我们需要如下配置：

- 1、发送+++
进入 AT 指令模式。
- 2、发送 AT+U2W
使 wifiIO 的 uart 进入纯数据模式中的 uart2wifi 子模式。

2.3 编程实现

下面是 Uno 的完整例程：

```
/*
 * @description:
 * this demo shows how Uno to get the data from the internet var wifiIO..
 * * @author: jansion
 * @date: 2015.11.16
 * @version: 1.0.0
 * @note:
 * @detail description:
 *
 *
 */

//define a buffer name info for receiving the data for wifiIO.
const short info_size = 256;
char info[info_size];
int readlen = 0;

void setup() {
    //set Serial for communication with the wifiIO
    Serial.begin(115200);
    delay(300);
    //send the AT command +++ to wifiIO for force the wifiIO into AT command mode
    Serial.println("+++");
    //delay 200ms for wifiIO dealing.
    delay(200);
    // entry the mode U2W e.i. Pure data mode and the data received var uart will
    //be transmit to internet.
    Serial.println("AT+U2W");
    delay(200);
    Serial.flush();
}
```

```

}

void loop() {
    //loop read the data from the wifiIO that receiving the data through inner wifi stack.
    if ((readlen = Serial.available()) > 0){
        Serial.readBytes(info, readlen);
        //output the data read form wifiIO to
        Serial.print(info);
    }
}
}

```

2.4 操作演示

把 2.3 节中的程序烧写进 Uno, 然后打开 arduino 串口监视串口, 此时我们会看到如图 2.4 - 1 所示的界面。这三行输出正是 Uno 在 setup() 函数中发送给 wifiIO 的 AT 指令, 在 loop 函数中 Uno 循环的从 wifiIO 读取数据并答应出来。但是此时 wifiIO 并没有收到数据, 因此会没有数据输出。

现在我们可以使用 wifiIO_IotManager 手机 app 发送数据给 wifiIO, wifiIO 然后会把数据通过串口发送给 Uno。关于如何使用 wifiIO_IotManager 发送数据给 wifiIO 可以参见《如何使用 wifiIO_IoTManager 控制 wifiIO》的第 2.4 节所述。打开 wifiIO_IoTManager, 点击 Uart 按钮进入 wifi2Uart 数据发送界面。我们在 UART 输入框中输入 “hello world!”, 点击 send 按钮, 在输入 “hello world! I am best!” 点击 send 按钮。

我们已经发送了两条字符串给 Uno, 实际上 Uno 到底有没有收到这两条数据呢? 我们看看 arduino 串口监视器中, 我们会看到如图 2.4 - 2 所示的输出信息。当然由于 Uno 会把收到的程序马上回传会手机, 因此我会在 wifiIO_IoTManager 中看到图 2.4 - 3 所示的信息。图 2.4 - 2 和图 2.4 - 3 表明 Uno 和手机通讯成功了

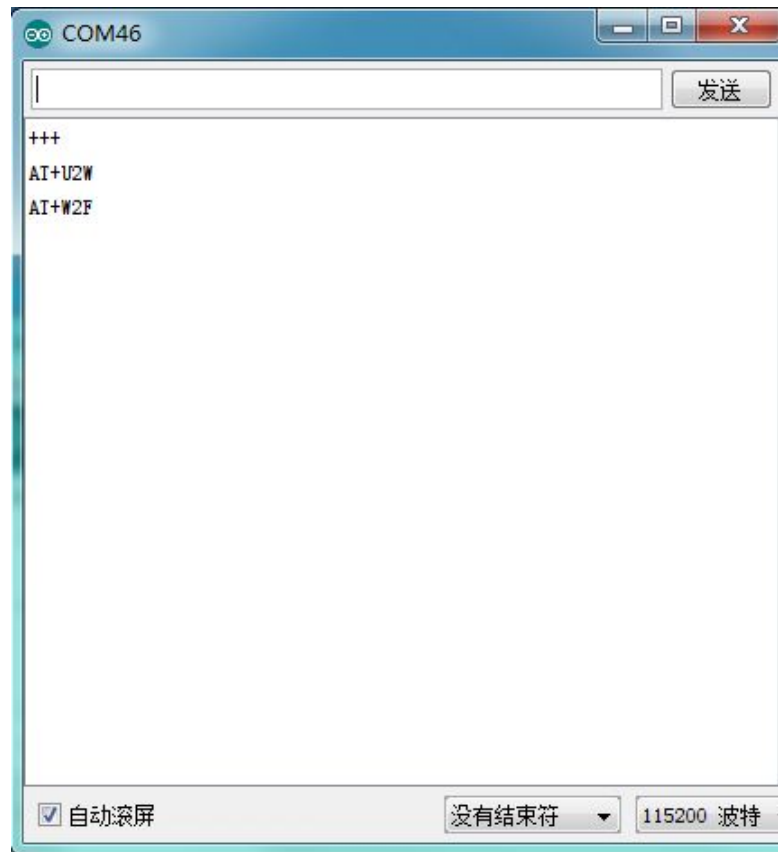


图 2.4 - 1

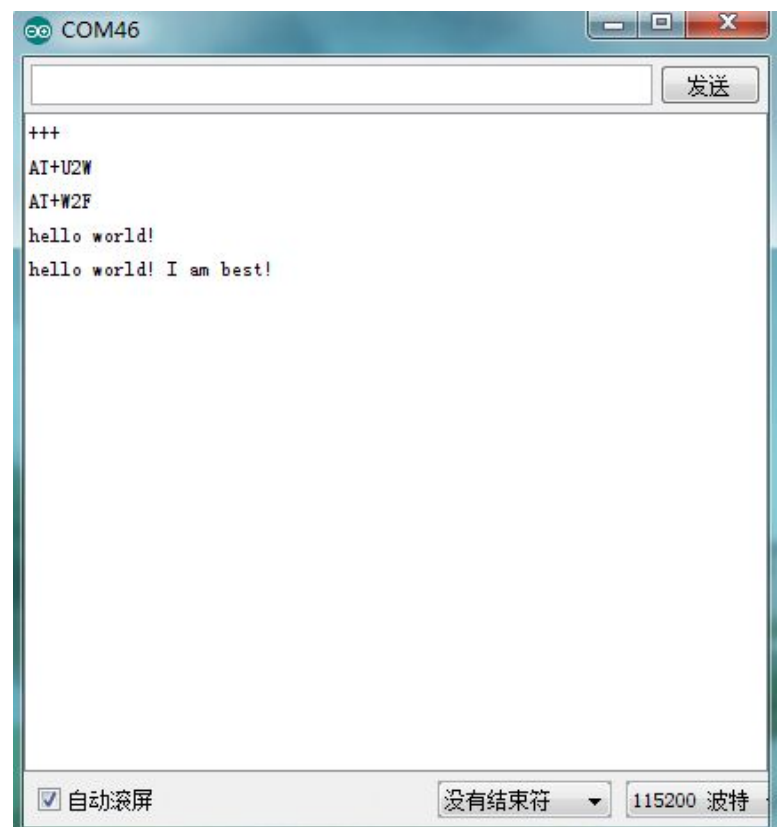


图 2.4 - 2



图 2.4 - 3

3 其他

如第 2 节、第 3 节所示，通过 `arduino` 发送 AT 指令就可以轻松的配置 `wifiIO`，更多的 AT 指令可以参见《`wifiIO` AT 指令一览表》。