

NodeMCU - 32S 核心开发板

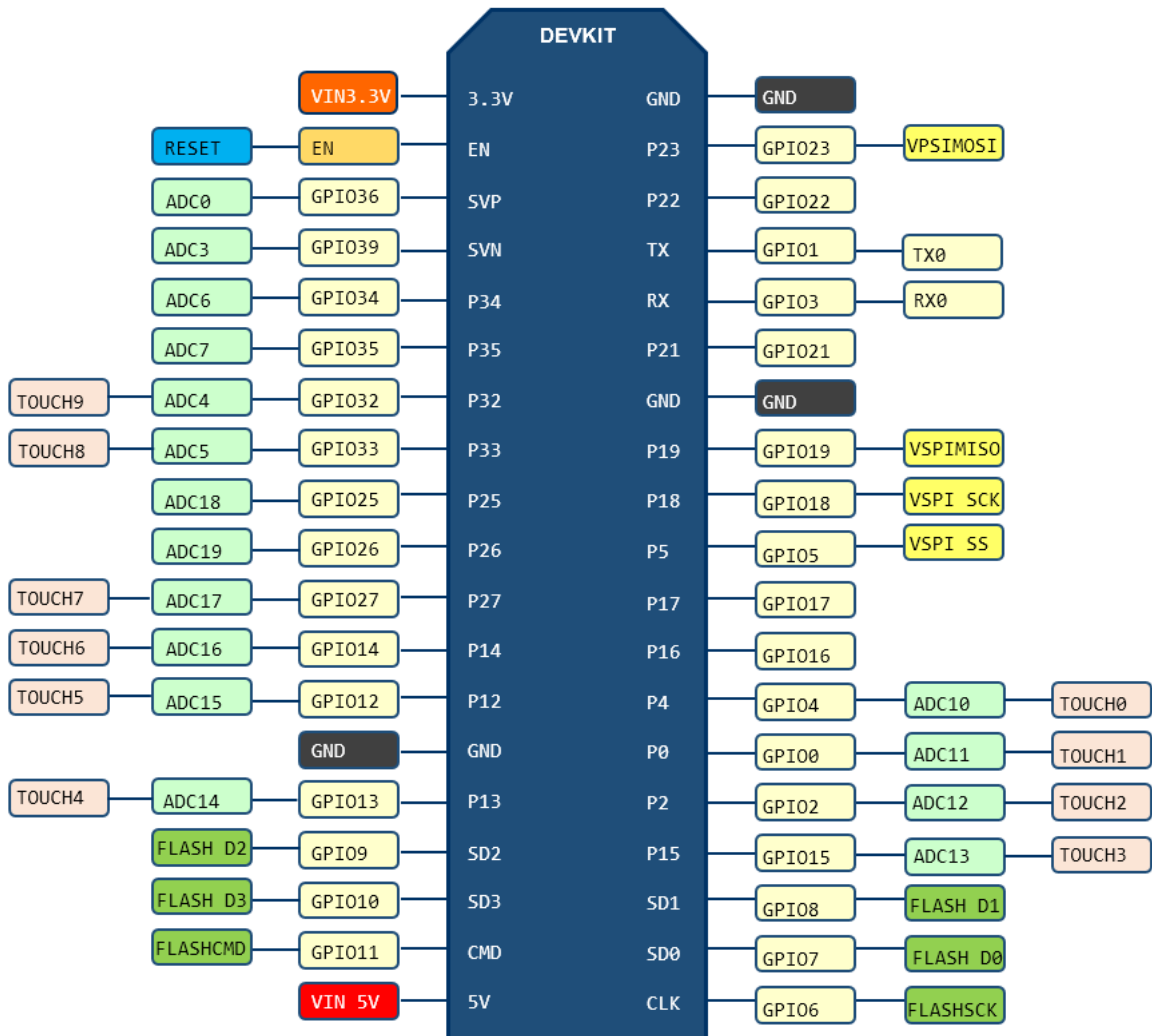


概述

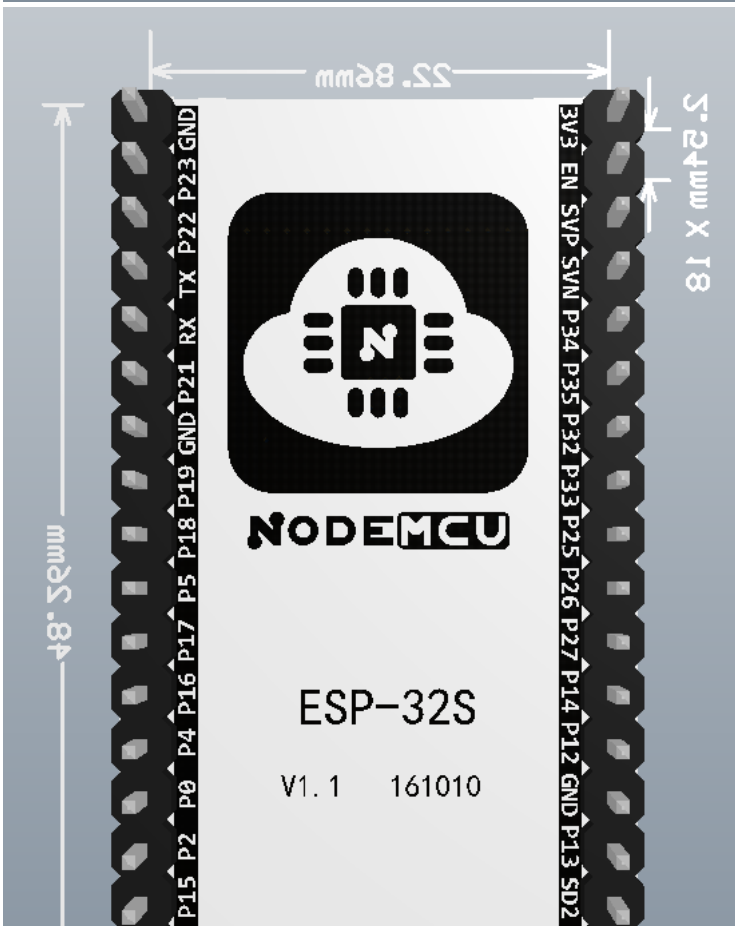
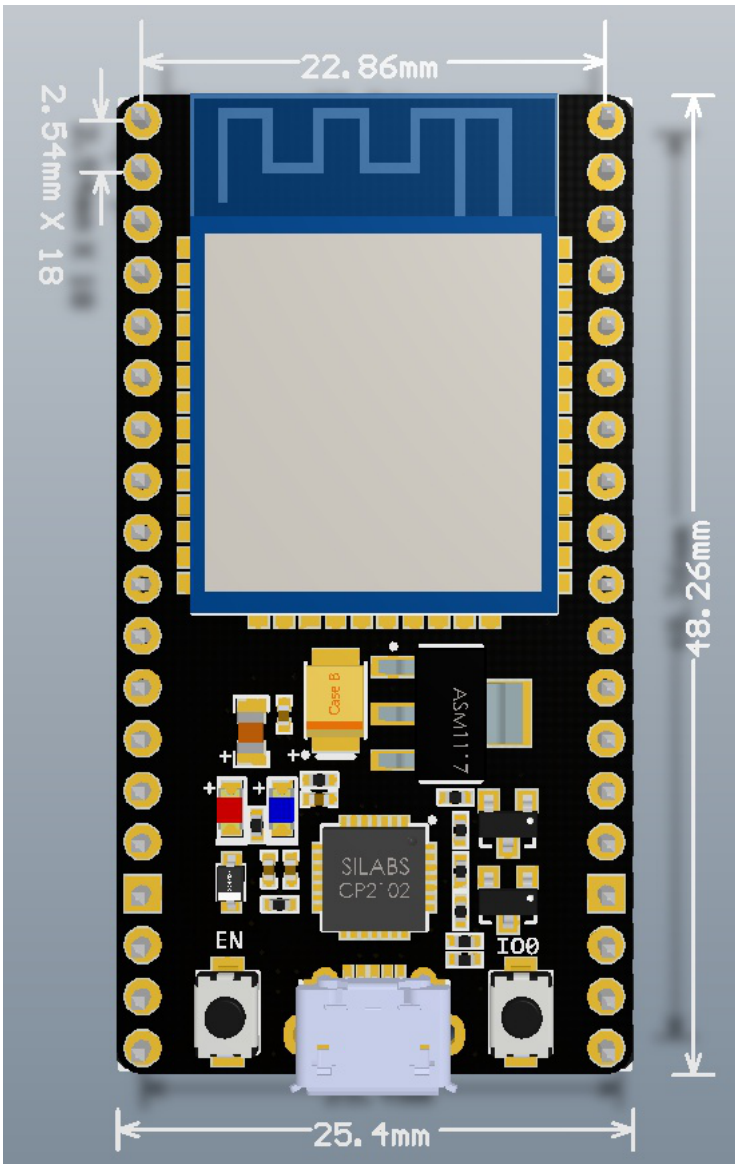
NodeMCU-32S 是安信可基于 ESP32-32S 模组所设计的核心开发板。该开发板延续了 NodeMCU 1.0 经典设计，引出大部分 I/O 至两侧的排针，开发者可以根据自己的需求连接外设。使用面包板进行开发和调试时，两侧的标准排针可以使操作更加简单方便。

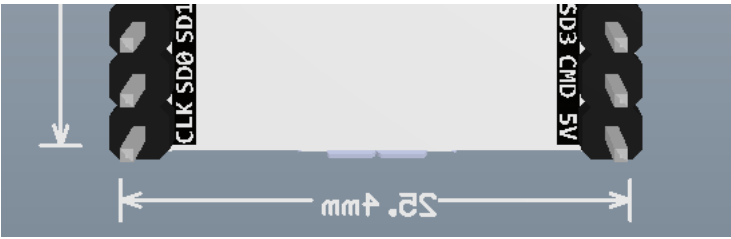
引脚图

PIN DEFINITION

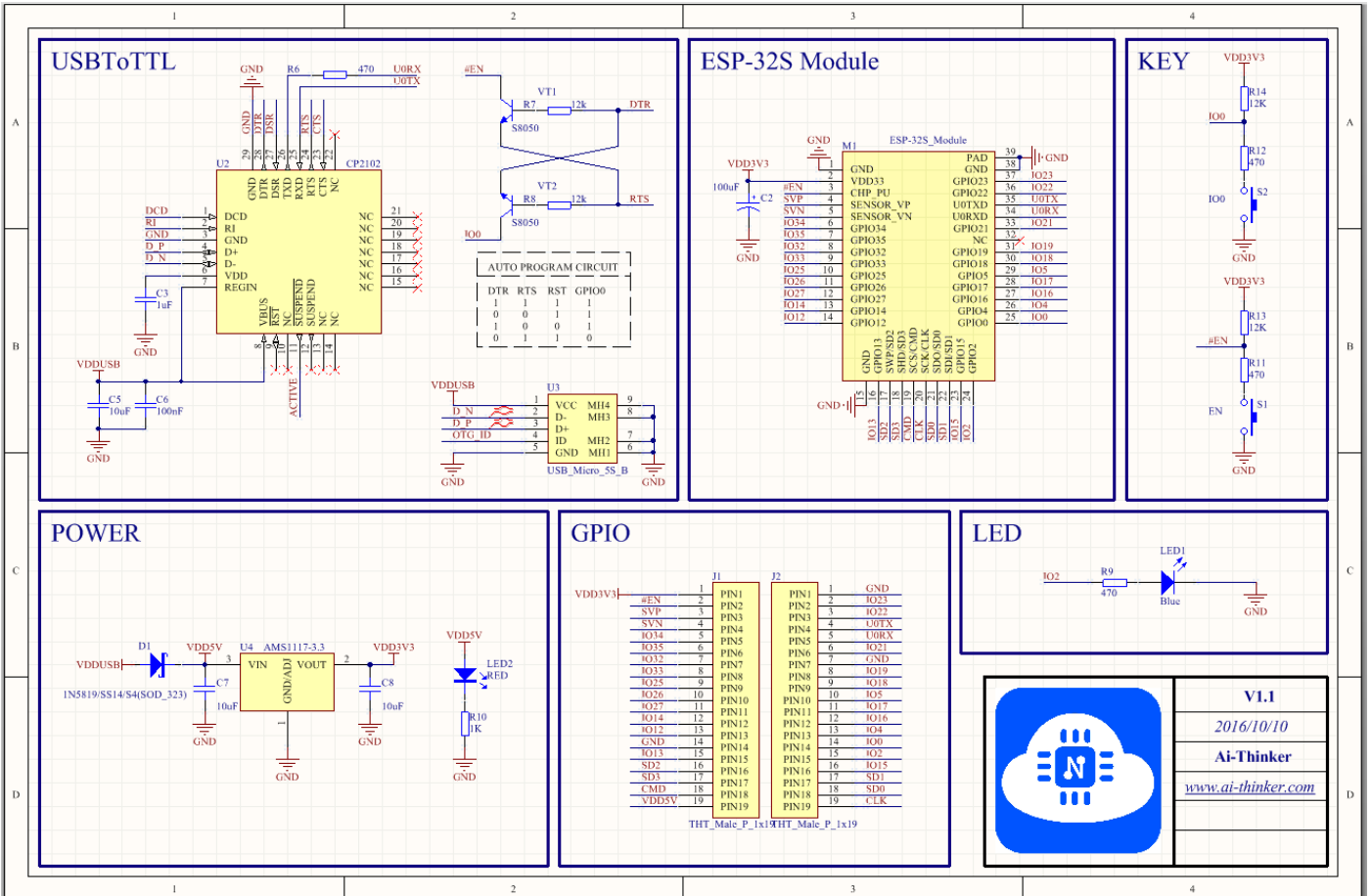


尺寸图





原理图



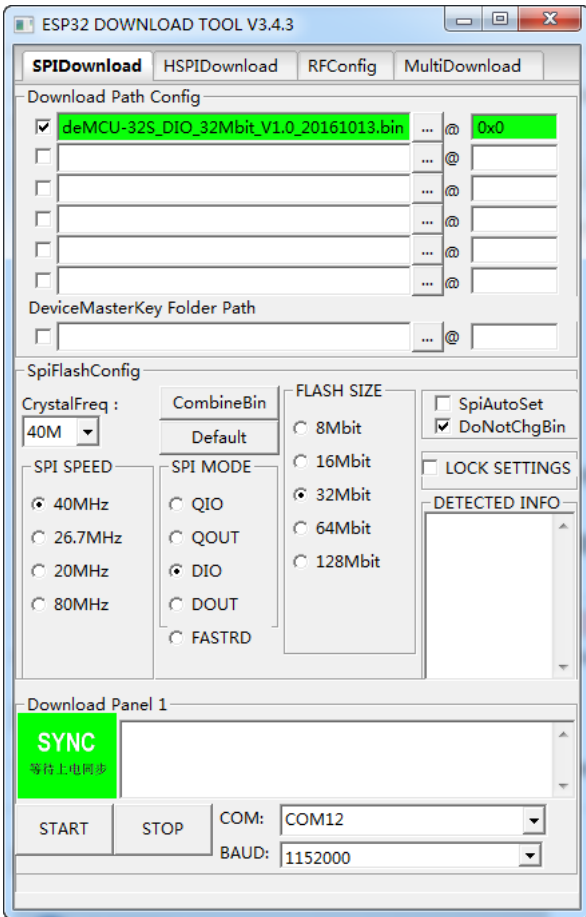
如何使用

1. 连接线缆

使用 Micro USB 数据线连接 PC 与 Nodemcu 核心开发板，并从 Windows 设备管理器中确认芯片的 COM 口。

2. 下载出厂默认固件

下载出厂固件 ai-thinker_nodemcu-32s_dio_32mbit_20170705.zip，根据 如何为 ESP 系列模组烧录固件 章节配置



3. 验证下载结果

下载完毕后，使用串口工具打开端口，查看数据输出，打印如下：

```

I (202) heap_alloc_caps: Initializing heap allocator:
I (203) heap_alloc_caps: Region 19: 3FFC0D24 len 0001F2DC tag 0
I (203) heap_alloc_caps: Region 25: 3FFE8000 len 00018000 tag 1
I (210) cpu_start: Pro cpu up.
I (214) cpu_start: Starting app cpu, entry point is 0x400808e4
I (0) cpu_start: App cpu up.
I (226) cpu_start: Pro cpu start user code
rtc v112 Sep 26 2016 22:32:10
XTAL 40M
I (261) cpu_start: Starting scheduler on PRO CPU.
I (40) cpu_start: Starting scheduler on APP CPU.

Mounting flash filesystem...
Formatting file system. Please wait...
mount res: -10025, -10025
Formatting: size 0x70000, addr 0x190000
mount res: 0, 0
frf2_timer_task_hdl:3ffc6214, prio:22, stack:2048
tcpip_task_hdlxxx : 3ffc6d78, prio:18, stack:2048
Task task_lua started.
phy_version: 123, Sep 13 2016, 20:01:58, 0
pp_task_hdl : 3ffc9ff0, prio:23, stack:8192
NodeMCU ESP32 build unspecified powered by Lua 5.1.4 on IDF 1.0.0(970dec6)
lua: cannot open init.lua
> Heap size::184592.

```

首次执行程序时，会自动格式化文件系统，时间会比较长，请耐心等待。

使用串口工具发送 `print("hello NodeMCU")`（注意加换行符）

程序结果显示如下：

```

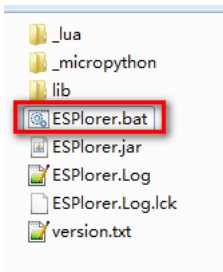
> print("hello world!")
hello world!
>

```

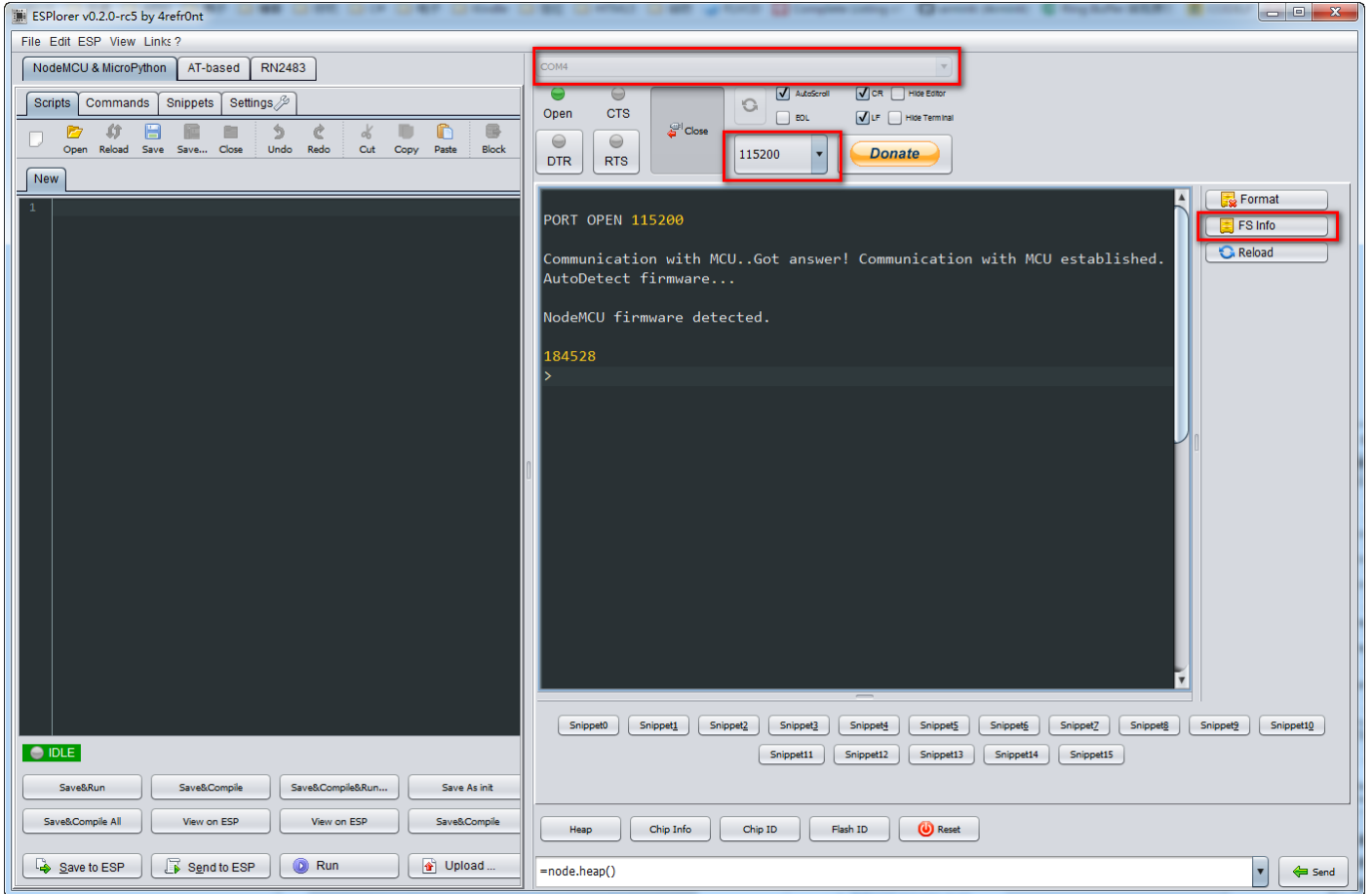
4. 使用 ESPlorer 调试 Lua

首先下载 ESPlorer (<https://esp8266.ru/esplorer>)（注意需要java运行环境）

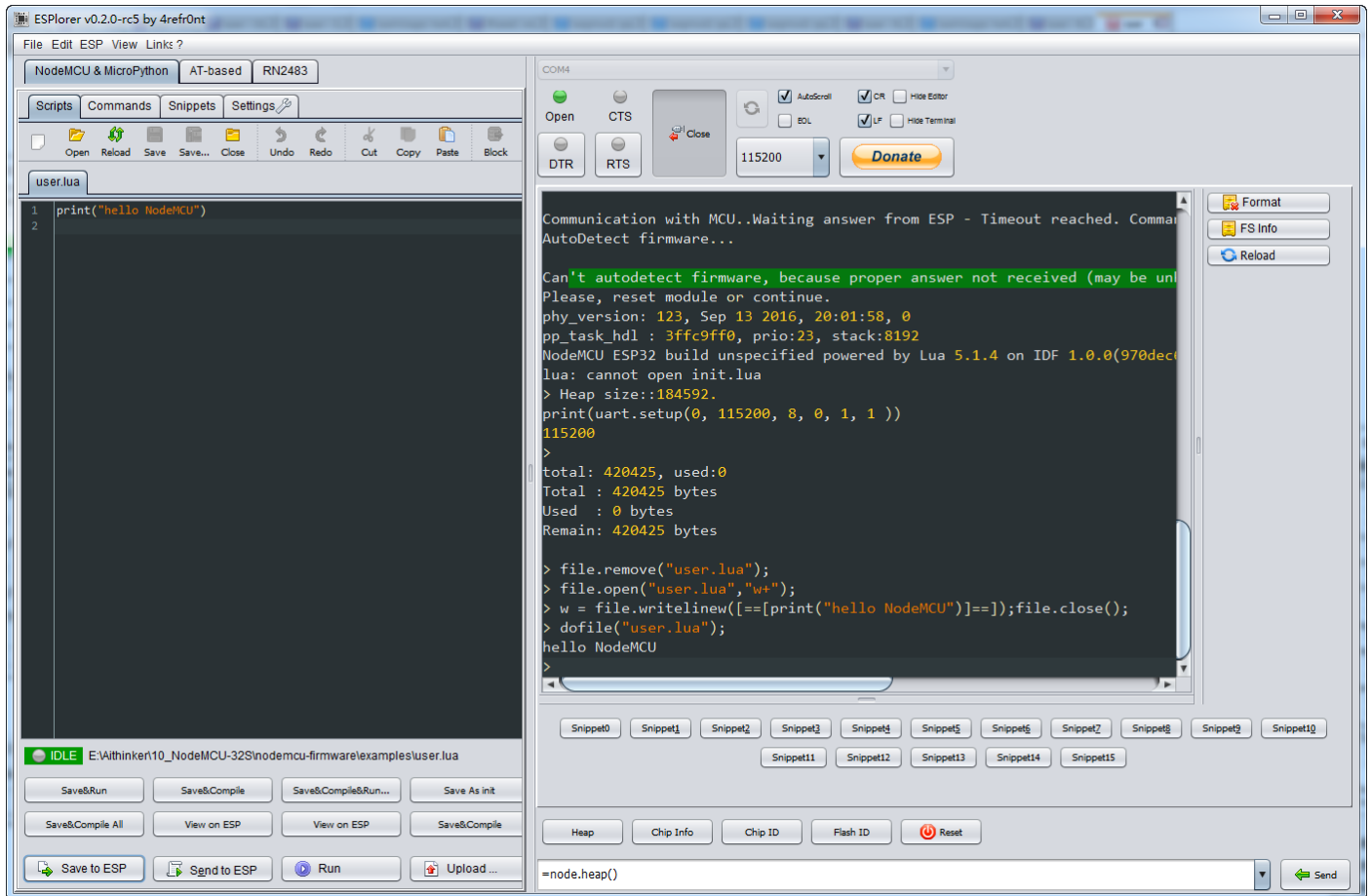
下载后打开 ESPlorer.bat 运行 ESPlorer



选择正确的端口，并打开串口（115200,8,n,1），点击【FS Info】或者最下方的【Reset】，可以看到模组输出，此时连接正常。



点击左边【Open】打开一个lua文件，并点击【Save to ESP】或者【Send to ESP】上传到esp32，即可以看到 NodeMCU-32S 运行该 lua 脚本的结果。



注意: ESP-Lua 目前尚处于开发阶段, 部分功能依然无法使用, 仅作为开发学习和体验。相关bug和反馈请提交到<https://github.com/nodemcu/nodemcu-firmware/issues> (<https://github.com/nodemcu/nodemcu-firmware/issues>)

如何编译

如何编译 ESP32 NodeMCU 固件

相关资源

出厂固件: ai-thinker_nodemcu-32s_dio_32mbit_v1.0_20161101.7z

硬件资源: nodemcu_32s_hardware_resources.7z

NodeMCU: <https://github.com/nodemcu/nodemcu-firmware/tree/dev-esp32> (<https://github.com/nodemcu/nodemcu-firmware/tree/dev-esp32>)

Arduino: <https://github.com/espressif/arduino-esp32> (<https://github.com/espressif/arduino-esp32>)

[esp32/boards/nodemcu_32s.txt](#) 最后更改: 4周前 由 [lvx](#)

除额外注明的地方外, 本维基上的内容按下列许可协议发布: CC Attribution-Share Alike 4.0 International (<http://creativecommons.org/licenses/by-sa/4.0/>)

X-ON Electronics

Largest Supplier of Electrical and Electronic Components

Click to view similar products for [WiFi Modules \(802.11\)](#) category:

Click to view products by [Ai-Thinker](#) manufacturer:

Other Similar products are found below :

[WISE-1520ITB-TDA1E](#) [SX-PCEAN2C-SP](#) [BCM43602KMLG](#) [7265.NGWG.W](#) [ENW-49801A1JF](#) [WH-M2SD50NBT](#) [SX-680-2700-SP](#)
[RN171-IRM481](#) [FXX-3061-MIX](#) [9668C52W10E](#) [EMIO-1533-00A2](#) [EWM-W162M201E](#) [ISM43340-L77-TR](#) [BCM4352KMLG](#)
[BCM43520KMLG](#) [BCM43217KMLG](#) [7265.NGWWB.W](#) [PPC-WL-KIT02-R11](#) [RC-CC2640-A](#) [M113DH3200PS3Q0](#) [SX-PCEAN2c](#) [WT-](#)
[01S](#) [WT8266-S3](#) [ESP-07S](#) [WT8266-S6](#) [ESP-12S](#) [WT-01F](#) [WT8266-S5](#) [ESP-12F](#) [WT32-S1](#) [ESP-WROOM-02UC](#) [ESP-WROOM-02DC](#) [WT-](#)
[01N](#) [ESP32-WROOM-32UC](#) [ESP32-WROOM-32DC](#) [ESP-01](#) [ESP-01S](#) [ESP32-WROOM-32\(16MB\)](#) [ESP32-WROVER-E\(8MB\)](#) [ESP32-](#)
[WROVER-IB\(16MB\)](#) [ESP32-WROVER-E\(16MB\)](#) [ESP32-WROVER-IB\(8MB\)](#) [ESP32-WROOM-32D\(16MB\)](#) [ESP32-WROOM-32U\(8MB\)](#)
[ESP32-WROOM-32U\(16MB\)](#) [ESP-WROOM-02\(4MB\)](#) [ESP-WROOM-02D\(4MB\)](#) [ESP32-WROVER-E\(4MB\)](#) [ESP32-WROVER-B\(16MB\)](#)
[ESP32-WROVER\(IPEX 4MB\)](#) [EAR00370](#) [EAR00373](#) [EAR00364](#)