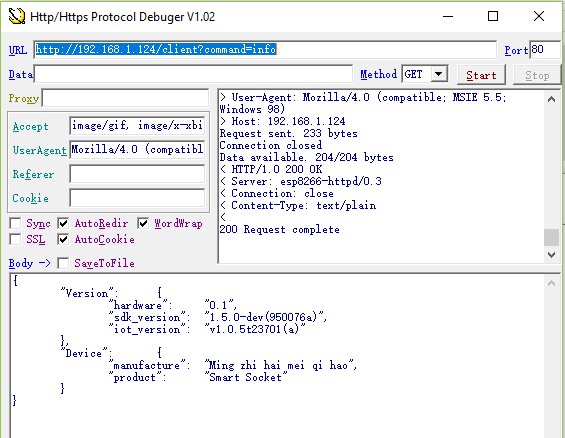
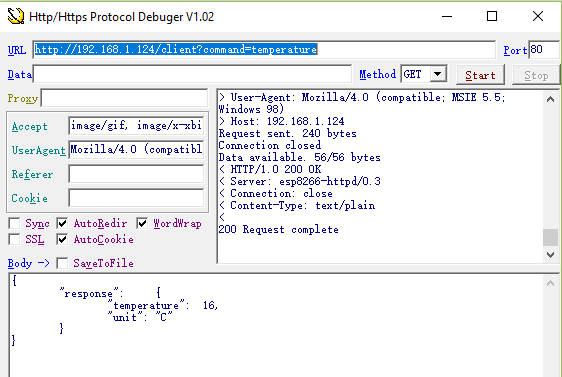
1. Httpdebug.exe是一个网络测试的小工具，可以用来发送基本的http POST或者GET请求。
2. 查询智能插座设备信息，使用GET请求（后续URL地址都请填写由手机APP配置时智能插座实际获取的IP地址）。

<http://192.168.1.124/client?command=info>， Data为空，点击start。



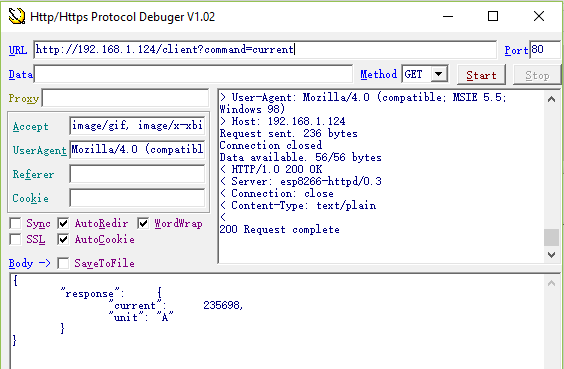
1. 查询智能插座温度信息（CS5463提供），使用GET请求。

http://192.168.1.124/client?command=temperature， Data为空，点击start。



1. 查询智能插座电流信息（CS5463提供），使用GET请求。

http://192.168.1.124/client?command=current， Data为空，点击start。

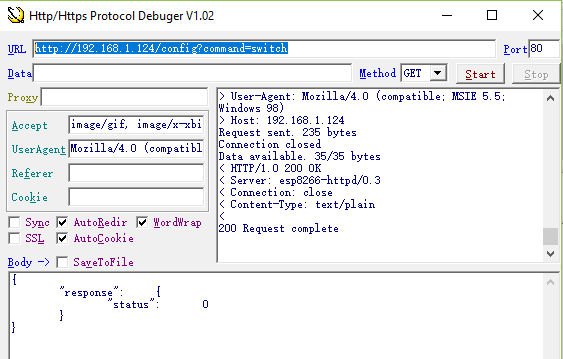


注：电流为CS5463原始数据读取值，还未进行数据加工。

将current替换为power就是查询功率信息，替换成voltage就是电压信息

1. 查询智能插座继电器状态，使用GET请求。

http://192.168.1.124/config?command=switch， Data为空，点击start。



1. 控制智能插座继电器状态，使用POST请求。

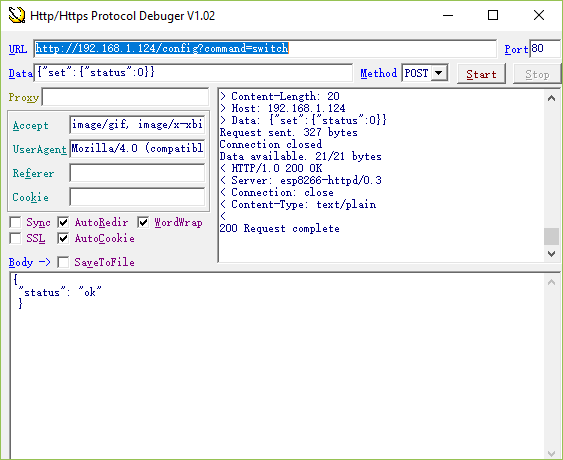
http://192.168.1.124/config?command=switch， Data为

{"Response":{"status":0}}

或者

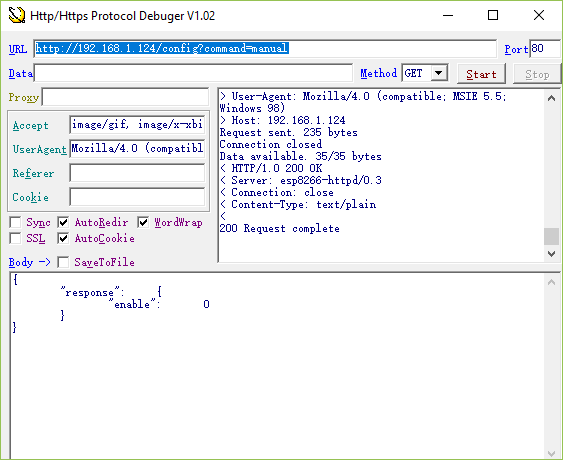
{"Response":{"status":1}}

0是open继电器，1是close继电器，点击Start。



1. 查询是否允许按键轻触控制继电器，使用GET请求。

http://192.168.1.124/config?command=manual， Data为空，点击start。



1. 控制智是否允许按键轻触控制继电器，使用POST请求。

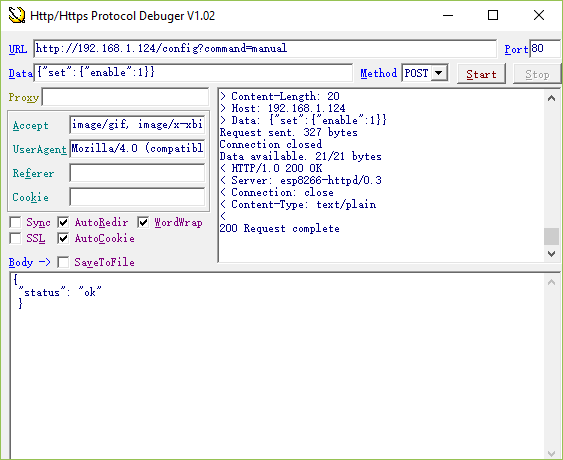
http://192.168.1.124/config?command=manual， Data为

{"set":{"enable":0}}

或者

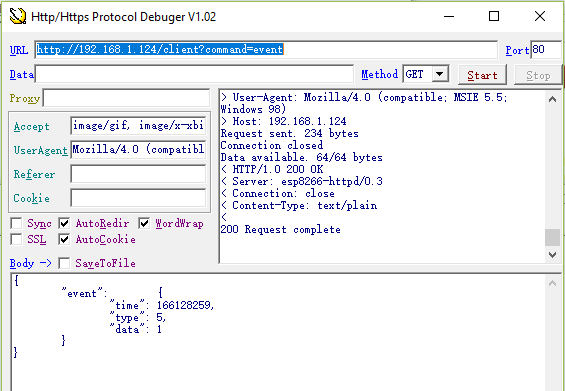
{"set":{"enable":1}}

0是禁止，1是允许，点击Start。



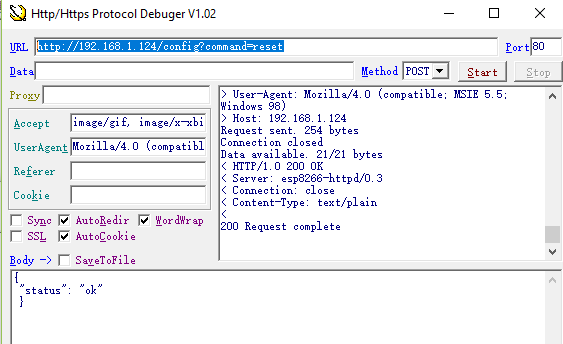
1. 远程控制继电器的开合会记录事件消息在flash中，最多保存40条记录（后续会增加其他类型的记录，比如欠电压，过流等事件），断电保存，可以通过GET请求查询。

<http://192.168.1.124/client?command=event>， Data为空（后续通过指定Data参数来进行事件选择，比如0就是最新的记录事件，1就是倒数第二个……），点击start。（系统时间为原始数据，后续会进行数据加工，以后可通过NTP网络同步时间）



1. 擦除智能插座事件记录，使用POST请求。

<http://192.168.1.124/config?command=reset，Data>为空，点击Start。（事件记录好像还有点bug，如果遇到读出来的是空就用这个擦除一下。）



1. 获取SNTP配置。使用GET请求。

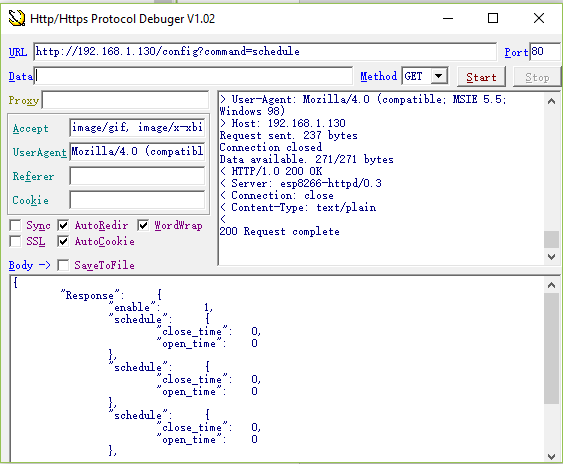
<http://192.168.1.124/config?command=sntp，Data>为空，点击Start。

1. 设置SNTP服务器，使用POST请求。一共允许设置三个SNTP服务器。

{"set":{"sntp":{"index":0,"url":"cn.pool.ntp.org"}}}

1. 查询是否允许定时开关继电器，使用GET请求。

http://192.168.1.130/config?command=schedule， Data为空，点击start。



1. 使能允许定时开关继电器，使用POST请求。

如果Enable了定时开关继电器，按键控制继电器将会失效（过流保护依然有效）。

http://192.168.1.130/config?command=schedule， Data为

{"set":{"enable":0}}

或者

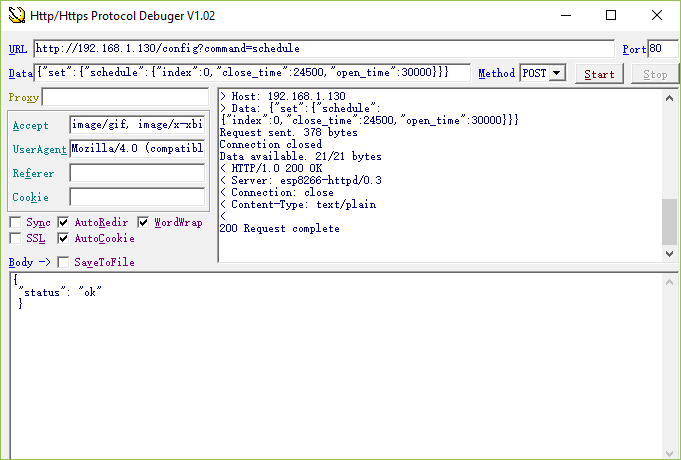
{"set":{"enable":1}}

0是Disable，1是Enable，点击Start。

1. 设置定时开关定时器时间，使用POST请求。一共支持4个定时器的设置，每个定时器可设置每天继电器close和open的时间，每个定时器的时间设置不能有交叠，否则设置无效。设置的单位为距离0时0分0秒的秒数，并且最大值不能超过一天的最大秒数。如果需要删除一条定时器记录，只需将此条记录的close和open时间都设置为0.

请使用UTC时间，timezone为0。

http://192.168.1.130/config?command=schedule， Data为{"set":{"schedule":{"index":0,"close\_time":24500,"open\_time":30000}}}



1. 在线固件升级使用POST。

http://192.168.1.130/upgrade?command=start， Data为

{"set":{"upgrade\_server":"120.41.31.19","upgrade\_host":"iot.zulolo.cn","upgrade\_port":80,"upgrade\_token":"123456789ABCDEF","upgrade\_url":"/Smart\_plug\_upgrade/fw\_download/version/user"}}

其中upgrade\_server为升级服务器的IP地址，upgrade\_host为升级服务器的域名（手机APP需要将升级服务器的域名解析为IP地址），upgrade\_url为需下载固件的路径

点击Start。

升级完成后会自动重启并使用之前设置的AP SSID和密码连接路由器，可通过<http://192.168.1.130/client?command=info> 参看"iot\_version"了解是否升级成功。（第三位为固件编译时间）

注：同一个版本的程序需要同时编译为可存放user space 1 和2的两个固件（设置app=1编译一次，再设置app=2编译一次）。然后将生成的两个bin文件（user1.4096.new.6.bin和user2.4096.new.6.bin）改名为user1和user2并部署到升级服务器上。智能插座程序会自动识别当前正在使用的固件是位于user space 1还是2，然后下载对应的固件。例如当前使用的是user1，智能插座程序就会去下载user2。

