

RTL8720CF-AT 指令手册

修改记录.....	2
1. 基本信息.....	2
2. 通用 AT 指令.....	2
2.1 AT-测试 AT 指令.....	2
2.2 ATSP?-列出所有 AT 指令.....	3
2.3 ATSR 模块重启.....	3
2.4 ATSV 查询版本.....	3
2.5 ATSE 设置串口回显.....	4
2.6 ATSY 恢复出厂设置.....	4
2.7 ATSU 设置串口参数.....	4
2.8 ATSO OTA 升级.....	5
3. WIFI 指令.....	6
3.1 ATPW-设置 wifi 模式.....	6
3.2 ATPN-连接到 AP.....	6
3.3 ATWD-从 AP 连接中断开.....	7
3.4 ATWS 扫描 AP 热点.....	7
3.5 ATPA-开启 AP 模式.....	7
3.6 ATW?-查看连接信息.....	8
3.7 ATPH-设置 DHCP 模式.....	8
3.8 ATPE-设置 STA 模式的 DHCP 规则.....	9
3.9 ATPF-设置 AP 模式的 DHCP 规则.....	9
3.10 ATPG-设置自动连接.....	10
3.11 ATPM-修改 MAC 地址.....	10
3.12 ATWQ simple config 配网.....	11
4. TCP/UDP 指令.....	11
4.1 ATPS-创建 TCP/UDP/SSL 服务端.....	11
4.2 ATPC-创建 TCP/UDP/SSL 客户端.....	12
4.3 ATPD-删除连接.....	13
4.4 ATPT 发送数据.....	14
4.5 ATPR-接收数据.....	15
4.6 ATPK-自动接收数据模式.....	16
4.7 ATPI-查看连接列表.....	17
4.8 ATPP-ping 命令.....	17
4.9 ATPU-设置透传模式.....	18
4.10 ATPL-开机自动启动透传模式.....	18
5. 蓝牙指令.....	19
5.1 ATBB-蓝牙配网指令.....	19
6. 附录.....	20
6.1 附录 1: OAT 升级流程.....	20
6.2 附录 2:simple config 配网流程.....	23
6.3 附录 3:蓝牙配网流程.....	28

修改记录

类型	修改内容	修改人	日期	软件版本
A	初稿版本	杨宾	2020/05/08	V2.2.1
M	修改 flash 分区中镜像大小为最大 0x90000 避免镜像超过默认分配的 flash 空间导致另外的固件损坏	杨宾	2020/6/3	V2.3.0

类型：A-新增 M-修改 D-删除

1. 基本信息

Log 串口

Log 串口主要用于输出 log

TX:PA_16 RX:PA_15

波特率：115200

AT 指令串口 uart0

AT 指令串口用来输入 AT 指令，并查看返回结果

TX:PB_14 RX:PB_13

波特率 38400

注意 AT 指令以换行符结束(\r\n)

最小系统需要连接 VCC,GND 和 AT 串口就可以进行操作了。

2. 通用 AT 指令

2.1 AT-测试 AT 指令

AT	
描述	测试 AT 指令是否已经启动
响应	[AT] OK
例子	#AT [AT] OK

2.2 ATS?-列出所有 AT 指令

ATS?	
描述	列出所有 AT 指令
响应	[ATS?] <command list> [ATS?] OK [ATS?] ERROR: <error_no>
错误码	1: 获取命令列表失败
例子	# ATS? [ATS?] Common AT Command: AT ATS? ... ATPL [ATS?] OK

2.3 ATSR 模块重启

ATSR	
描述	重启模块
响应	AT COMMAND READY(模块重启完成)
例子	ATSR

2.4 ATSV 查询版本

ATSV	
描述	查询版本信息
响应	[ATSV] OK:AT Version:<at-version> SDK Version:<sdk-version>(<compile_time>)
例子	# ATSV [ATSV] OK:AT Version:v2.2.1 SDK Version:v7.1(2020/05/07-17:20:42)

2.5 ATSE 设置串口回显

ATSE=<echo>	
描述	设置回显和 debug 信息等级
响应	[ATSE] OK
参数	echo:0:关闭命令回显 1:开启命令回显
错误码	1: 参数数量错误 2: 参数错误
例子	# ATSE=1 [ATSE] OK

2.6 ATSY 恢复出厂设置

ATSY	
描述	恢复出厂默认设置
响应	[ATSY] OK
错误码	1: 恢复默认设置失败 2: 恢复默认镜像失败
例子	# ATSY [ATSY] OK AT COMMAND READY

2.7 ATSU 设置串口参数

ATSU=<baudrate>,<databits>,<stopbits>,<parity>,<flowcontrol>,<configmode>	
描述	设置串口参数
响应	成功 [ATSU] OK 失败 [ATSU] ERROR:<error_code>
参数	baudrate:波特率 2400, 4800, 9600, 19200, 38400(default),57600, 115200, 921600, 1152000 databits:数据位 5: 5 bit data 6: 6 bit data 7: 7 bit data 8: 8 bit data (default)

	stopbits: 停止位 1: 1 bit stop (default) 2: 2 bit stop Parity: 校验位 0: None parity (default) 1: Odd parity 2: Even parity Flowcontrol: 流控 0: disable flowcontrol (default) 1: enable RTS and CTS Configmode: 生效方式 0: 配置立即生效但是不保存到 flash 1: 保存配置到 flash, 并立即生效 2: 保存配置到 flash, 重启后生效
错误码	1: 参数格式错误 2: 参数错误
例子	# ATSU=38400,8,1,0,0,2 [ATSU] OK

2.8 ATSO OTA 升级

ATSO=<IP>,<port>	
描述	OTA 升级
响应	[ATS0] OK 这个是异步操作, 收到 OK 只是表示任务开始了, 其他消息要在 log 口查看, 升级成功后会自动重启
参数	IP: 服务器 IP Port: 服务器端口
例子	参考 附录 1: OAT 升级流程

3. WIFI 指令

3.1 ATPW-设置 wifi 模式

ATPW=<mode>	
描述	设置 wifi 模式（使用 ATPN 和 ATPA 必须配合指定的模式） 连接热点必须是 mode1/mode3，启动 AP 必须是 mode2/mode3
参数	mode: 1 : Station mode (default) 2 : AP mode 3 : Concurrent mode
响应	成功 [ATPW] OK 失败 [ATPW] ERROR:<error_no>
错误码	1: 参数格式错误 2: 参数错误
例子	# ATPW=3 [ATPW] OK

3.2 ATPN-连接到 AP

ATPN=<ssid>,<pwd>	
描述	连接到指定 AP
参数	ssid: AP 名字(特殊字符前要加'\') pwd: 密码
响应	成功 [ATPN] OK 失败 [ATPN] ERROR:<error_code>
错误码	1: 命令格式错误 2: 参数错误 3: wifi 初始化错误 4: 连接 AP 错误 5: wifi 模式错误（ATPW 设置模式） 6: 获取安全类型错误

	7: dhcp 超时，使用静态 IP 192.168.1.80
例子	# ATPN=test,12345678 [ATPN] OK

3.3 ATWD-从 AP 连接中断开

ATWD	
描述	断开当前连接的 wifi
响应	成功 [ATWD] OK 失败 [ATWD] ERROR:<error_code>
错误码	3: 操作失败 4: 操作超时
例子	# ATWD [ATWD] OK

3.4 ATWS 扫描 AP 热点

ATWS	
描述	扫描 AT 热点
参数	AP 模式的 SSID
响应	AP : <num>,<ssid>,<chl>,<sec>,<rssi>,<bssid> [ATWS] OK
例子	# ATWS # AP : 2,test-AP,1,WPA/WPA2 AES,-44,2c:3b:44:54:66:77 [ATWS] OK

3.5 ATPA-开启 AP 模式

ATPA=<ssid>,<pwd>,<chl>,<hidden>[,<max_conn>]	
描述	开启 AP 模式
参数	ssid: AP 名字(特殊字符前要加'\') pwd: 密码 chl: 信道(1~11) hidden: 0: 不隐藏 SSID 1: 隐藏 SSID max_conn:最大连接人数(1~3 默认 3)

响应	成功 [ATPA] OK 失败 [ATPA] ERROR:<error_code>
错误码	1: 命令格式错误 2: 参数错误 3: wifi 初始化错误 4: 开启 AP 错误 5: wifi 模式错误 (ATPW 设置模式)
例子	# ATPW=3 [ATPW] OK # ATPA="SSID",,11,0 (这个密码为空表示没有密码) [ATPA] OK

3.6 ATW?-查看连接信息

ATW?	
描述	查看当前网络连接信息 (IP、MAC 等)
响应	<mode>,<SSID>,<chl>,<sec>[,<key_id>],<pwd>,<mac>,<ip>,<gw> CLIENT : <num>,<mac> [ATW?] OK
例子	# ATW? STA,,1,OPEN,,11:22:33:44:55:68,192.168.1.80,192.168.1.1 [ATW?] OK

3.7 ATPH-设置 DHCP 模式

ATPH=<mode>,<enable>	
描述	设置 DHCP 模式
响应	[ATPH] OK [ATPH] ERROR:<error_no>
参数	Mode: 1: AP 模式 2: STA 模式 Enable: 1: DHCP 模式 2: 静态 IP 模式
错误码	1: 命令格式错误 2: 参数错误
注	1. 默认 AP 和 STA 都是 DHCP 模式 2. 用 ATPE 为 STA 模式设置静态 IP 3. 用 ATPF 为 AP 模式设置静态 IP
例子	# ATPH=2,2 [ATPH] OK

3.8 ATPE-设置 STA 模式的 DHCP 规则

ATPE=<ip>[,<gateway>,<mask>]	
描述	设置 STA 模式的 DHCP 规则
响应	[ATPE] OK [ATPE] ERROR:<error_no>
参数	ip: 要设置的 IP 地址, eg:192.168.1.100 [gateway]: 网关 IP [mask]: IP 掩码
错误码	1: 命令格式错误 2: 参数错误
注	1. 默认 IP 为 192.168.1.80 2. 使用静态 IP 需要禁用 DHCP 功能(ATPH=2,2)
例子	# ATWD [ATWD] OK # ATPH=2,2 //设置 sta 使用静态 IP [ATPH] OK # ATPE=192.168.20.7,192.168.20.1,255.255.255.0 //设置 sta 的静态 IP [ATPE] OK ATPN=B&T-1,test123456

3.9 ATPF-设置 AP 模式的 DHCP 规则

ATPF=<start_ip>,<end_ip>,<gateway>	
描述	设置 AP 模式的 DHCP 规则
响应	[ATPF] OK [ATPF] ERROR:<error_no>
参数	start_ip: DHCP 起始 IP, eg:192.168.1.100 end_ip: DHCP 结束 IP(注意结束 IP 必须和起始 IP 属于同一网段且非网关 IP, 也不能是 255) gateway: 网关 IP(必须和起始结束 IP 属于同一网段)
错误码	1: 命令格式错误 2: 参数错误
注	3. 默认网关 IP 为 192.168.43.1 4. 启动 DHCP 需要启用 DHCP 功能(ATPH=1,1) 5. 不启用 DHCP 需要设置为静态 IP 模式(ATPH=1,2)
例子	# ATPH=1,1 [ATPH] OK # ATPF=192.168.55.100,192.168.55.199,192.168.55.1 [ATPF] OK # ATPW=3

	[ATPW] OK # ATPA="SSID",,11,0 [ATPA] OK # ATW? STA,,1,OPEN,,00:e0:4c:87:20:28,192.168.1.80,192.168.1.1 AP,SSID,11,OPEN,,00:e0:4c:87:20:29,192.168.55.1,192.168.55.1 [ATW?] OK
--	---

3.10 ATPG-设置自动连接

ATPG=<enable>	
描述	设置自动连接
响应	成功 [ATPG] OK 失败 [ATPG] ERROR:<error_no>
参数	enable: 0: 关闭自动连接 1: 开启自动连接
错误码	1: 命令格式错误 2: 参数错误
例子	# ATPG=0 [ATPG] OK

3.11 ATPM-修改 MAC 地址

ATPM=<mac>	
描述	修改 MAC 地址
响应	成功 [ATPM] OK 失败 [ATPM] ERROR:<error_no>
参数	mac: 6 字节 hex 数字
错误码	1: 命令格式错误 2: 参数错误
注	设置 MAC 成功后需要重启才能生效
例子	# ATPM=aa2233cc55ff [ATPM] OK

3.12 ATWQ simple config 配网

ATWQ	
描述	Simple config 配网 APP 和参考资料见 “SDK/tools/simple_config_wizard_3.4b”
响应	[ATWQ] OK 输入不会立即返回, 需要配网成功或者失败后才会有返回, 其他消息要在 log 口查看。

4. TCP/UDP 指令

4.1 ATPS-创建 TCP/UDP/SSL 服务端

ATPS =<mode>,<Local Port>	
描述	创建 TCP/UDP/SSL server
响应	成功 [ATPS] OK [ATPS] con_id=x (x=[1,9], con_id 0 is reserved) TCP 模式下当有客户端连接会返回 [ATPS] A client connected to server[<server_id> con_id:<x>,seed,tcp,address:xxx.xxx.xxx.xxx,port:<x>,socket:<x> (response format refer to section 4.8 ATPI) 失败 [ATPS] ERROR:<error_no>
参数	mod: 0:TCP 1:UDP 2:SSL Local Port: 1~65535
错误码	1: 参数数量错误 2: Local Port 范围出错 3: 创建 con_id 出错 4: 创建 server task 出错 5: 创建 socket 出错 6: 设置 socket 参数失败 7: bind 出错 8: listen 出错 9: tcp server 已经存在 10: accept 出错 11: 创建 seed con_id 出错

	12: udpserver 已经存在 13: 透传模式中不能创建 server 14: 未知的连接方式 15: 监听 socket 失败 16: server certificate malloc 失败 17: server key malloc 失败 18: x509_crt_parse failed for server certificate 19: x509_crt_parse failed for server ca list 20: pk_parse_key failed for server key 21: hang node failed for ssl server 22: accept error for ssl server 23: malloc failed for ssl seed 24: initialization failed for ssl context 25: ssl_set_own_cert error 26: ssl handshake failed for ssl seed 27: create node failed for ssl seed
例子	<pre># ATPN=B&T-1,test123456 [ATPN] OK # ATPS=0,4004 # [ATPS] OK [ATPS] con_id=1 当客户端连接后 [ATPS] A client connected to server[1] con_id:3,seed,tcp,address:192.168.20.112,port:52451,socket:2 # ATPi con_id:1,server,tcp,address:192.168.20.115,port:4004,socket:0 con_id:3,seed,tcp,address:192.168.20.112,port:52451,socket:2 con_id:2,server,udp,address:192.168.20.115,port:4005,socket:1 [ATPi] OK</pre>

4.2 ATPC-创建 TCP/UDP/SSL 客户端

ATPC =<mode>,< Remote Addr>,< Remote Port>[,<Local Port>]	
描述	创建 TCP/UDP/SSL 客户端
响应	成功 [ATPC] OK [ATPC] con_id=x (x=[1,9], con_id 0 is reserved) 失败 [ATPC] ERROR:<error_no>
参数	mod: 0:TCP

	1:UDP 2:SSL Remote Addr: 要连接的 server IP Remote Port: 要连接的 server port(1~65535) Local Port: 本地端口号(1~65535)
错误码	1: 参数数量错误 2: remote IP 格式错误或无法连接 3: remote Port 范围出错 4: 创建 con_id 出错 5: 创建 client task 出错 6: remote address 转换出错 7: 创建 socket 出错 8: 挂起 tcp 连接失败 9: connect error for tcp client 10: ang node error for udp client 11: local port should be 1~65535 12: bind local port error 13: connection already exists for TT(transparent transmission) mode 14: set broadcast on socket failed 15: set multicast add membership on socket failed 16: set multicast interface failed 17: connection type is unknown (SSL isn't supported) 18: Initiate a TCP connection with host:port failed for ssl client 19: memory allocation failed for ssl context structure 20: ssl context initialization failed 21: ssl handshake failed 22: hang node failed for ssl client
例子	# ATPN=B&T-1,test123456 [ATPN] OK //使用 socket 工具创建一个 tcp server # ATPC=0,192.168.20.112,10086 # [ATPC] OK [ATPC] con_id=1

4.3 ATPD-删除连接

ATPD=<con_id>	
描述	关闭连接
响应	成功 [ATPD] OK 失败 [ATPD] ERROR:<error_no>

表单编号: B&T-QR-EN-002 版本: B2 保存期限: 5 年 生效日期: 2014 年 7 月 4 日

参数	con_id: 0: 关闭所有连接 1~9: 关闭指定连接
错误码	1: 命令格式错误 2: 参数错误 3: con_id 没有找到
例子	# ATPD=3 [ATPD] OK

4.4 ATPT 发送数据

ATPT=<data_size>,<con_id>:<data>	
描述	发送数据
响应	成功 [ATPT] OK,<con_id> 失败 [ATPT] ERROR:<error_no>
参数	data_size: 数据长度 con_id: 1~9 data: payload 数据
错误码	1: 命令格式错误 2: <Buffer Size> 超过 ATPT 限制 3: con_id 没有找到 4: <dst_ip> 或 <dst_port> error for udp server case 5: sendto() error for udp server 6: sendto() error for udp client 7: TCP server should send data to the seed (server 需要发送给客户端, 不能直接发给 server) 8: write error for tcp client/server
注	如果是 TCP server 的话不能直接从 server 这个 con_id 读写数据, 必须针对他的连接子 con_id 进行读写数据
例子	//TCP server 读取数据 # ATPN=B&T-1,test123456 [ATPN] OK # ATPS=0,4004 # [ATPS] OK [ATPS] con_id=1 //当客户端连接后 [ATPS] A client connected to server[1] con_id:5,seed,tcp,address:192.168.20.112,port:52581,socket:2 # ATPi

	<pre> con_id:1,server,tcp,address:192.168.20.115,port:4004,socket:0 con_id:3,seed,tcp,address:192.168.20.112,port:52451,socket:-1 con_id:4,seed,tcp,address:192.168.20.112,port:52575,socket:-1 con_id:5,seed,tcp,address:192.168.20.112,port:52581,socket:2 con_id:2,server,udp,address:192.168.20.115,port:4005,socket:1 [ATPI] OK # ATPT=6,5:123456 //发送数据 # [ATPT] OK,5 </pre>
--	--

4.5 ATPR-接收数据

ATPR =<con_id>,<Buffer Size>	
描述	接收数据
响应	成功 [ATPR] OK,<data size>,<con_id>[,<dst_ip>,<dst_port>]:<data> 失败 [ATPR] ERROR:<error_no>
参数	con_id: 1~9 Buffer Size: bufter 大小(暂不生效)
错误码	1: 命令格式错误 2: <Buffer Size> 范围是 1 ~ MAX_BUFFER(默认 1600) 3: <con_id> 没有找到 4: recvfrom() error for udp server 5: recvfrom() error for udp client/seed 6: TCP server should receive from seed (server 需要从客户端读取, 不能直接读取 server) 7: connection lost 8: read() error for tcp con_id
注	如果是 TCP server 的话不能直接从 server 这个 con_id 读写数据, 必须针对他的连接子 con_id 进行读写数据
例子	<pre> //UDP server 读取数据 # ATPN=B&T-1,test123456 [ATPN] OK # ATPS=1,4005 # [ATPS] OK [ATPS] con_id=2 # ATPI con_id:1,server,tcp,address:192.168.20.115,port:4004,socket:0 con_id:2,server,udp,address:192.168.20.115,port:4005,socket:1 [ATPI] OK //此时使用 socket 工具向 192.168.20.115:4005 发送数据 </pre>

	<pre># ATPR=2,100 [ATPR] OK,8,2,192.168.20.112,10086:test str //TCP server 读取数据 # ATPN=B&T-1,test123456 [ATPN] OK # ATPS=0,4004 # [ATPS] OK [ATPS] con_id=1 //当客户端连接后 [ATPS] A client connected to server[1] con_id:3,seed,tcp,address:192.168.20.112,port:52451,socket:2 # ATPI con_id:1,server,tcp,address:192.168.20.115,port:4004,socket:0 con_id:3,seed,tcp,address:192.168.20.112,port:52451,socket:2 con_id:2,server,udp,address:192.168.20.115,port:4005,socket:1 [ATPI] OK //此时客户端发送数据 # ATPR=3,100 [ATPR] OK,8,3:test str</pre>
--	--

4.6 ATPK-自动接收数据模式

ATPK=<enable>	
描述	设置自动接收模式，接收到的数据在普通模式下和 ATPR 返回格式相同，透传模式下只返回数据，没有其他格式
响应	成功 [ATPK] OK 失败 [ATPK] ERROR:<error_no>
参数	enable: 0: 关闭自动接收 1: 开启自动接收
错误码	1: 参数错误 2: 开启自动接收任务失败
例子	<pre>//UDP server 读取数据 # ATPN=B&T-1,test123456 [ATPN] OK # ATPS=1,4005 # [ATPS] OK [ATPS] con_id=2</pre>

	<pre># ATPI con_id:1,server,tcp,address:192.168.20.115,port:4004,socket:0 con_id:2,server,udp,address:192.168.20.115,port:4005,socket:1 [ATPI] OK # ATPK=1 [ATPK] OK //此时使用 socket 工具向 192.168.20.115:4005 发送数据 # [ATPR] OK,8,2,192.168.20.112,10086:test str</pre>
--	---

4.7 ATPI-查看连接列表

ATPI	
描述	查看连接列表
响应	<pre>con_id :<con_id >,<server/seed(TCP client)/client>,\ <tcp/udp>,address:<IP ADDRESS>,port:<PORT>,socket:<socket id> ... [ATPI] OK</pre>
例子	<pre># ATPN=B&T-1,test123456 [ATPN] OK # ATPS=0,4004 # [ATPS] OK [ATPS] con_id=1 # ATPI con_id:1,server,tcp,address:192.168.20.115,port:4004,socket:0 [ATPI] OK</pre>

4.8 ATPP-ping 命令

<pre>ATPP=<Remote IP>,[count] Or ATPP=<con_id>,[count/loop]</pre>	
描述	修改 MAC 地址
响应	<pre>成功 [ATPP] OK 失败 [ATPP] ERROR:<error_no></pre>

参数	Remote IP: IP 或者域名 count: ping 几次 loop: 一直 ping(注意执行这个操作后将无法返回, log 口有 ping 的消息打印)
错误码	1: 命令格式错误 2: con_id 无效
例子	# ATPN=B&T-1,test123456 [ATPN] OK # ATPP=www.baidu.com,1 [ATPP] OK

4.9 ATPU-设置透传模式

ATPU=<enable>	
描述	设置透传模式
响应	成功 [ATPU] OK 失败 [ATPU] ERROR:<error_no>
参数	enable: 1(开启透传模式, 开启后不能关闭)
错误码	1: 参数错误 2: 当前没有连接 3: server mode 无法进入透传模式 4: 进入透传只能有一个连接 5: 开启透传失败
注	1. 进入透传模式后发送四个减号"----"可以退出透传模式 2. ATPC 创建 client 之后需要在 20s 内让 server 发送任意数据到 client, 或者直接 ATPU=1 进入透传模式, 否则会导致认为这是一个无效连接而无法进入透传模式
例子	# ATPN=B&T-1,test123456 [ATPN] OK //使用 socket 工具创建一个 tcp server # ATPC=0,192.168.20.112,10086 # [ATPC] OK [ATPC] con_id=1 # ATPU=1 [ATPU] OK > pc send to 8720

4.10 ATPL-开机自动启动透传模式

ATPL=<enable>

描述	设置开机自动进入透传模式 注意：设置这个需要先设置开机自动连接热点，并启动创建好要透传的 socket 连接
响应	成功 [ATPL] OK 失败 [ATPL] ERROR:<error_no>
参数	enable: 0: 擦除透传参数，并关闭开机启动透传 1: 保存透传信息，并开启开机启动透传
错误码	1: 参数错误 2: 参数数量错误 3: 没有找到连接
例子	# ATPN=hwap,1234567890 [ATPN] OK # ATPG=1 [ATPG] OK # ATPC=0,192.168.2.62,5002 # [ATPC] OK [ATPC] con_id=1 # ATPL=1 [ATPL] OK

5. 蓝牙指令

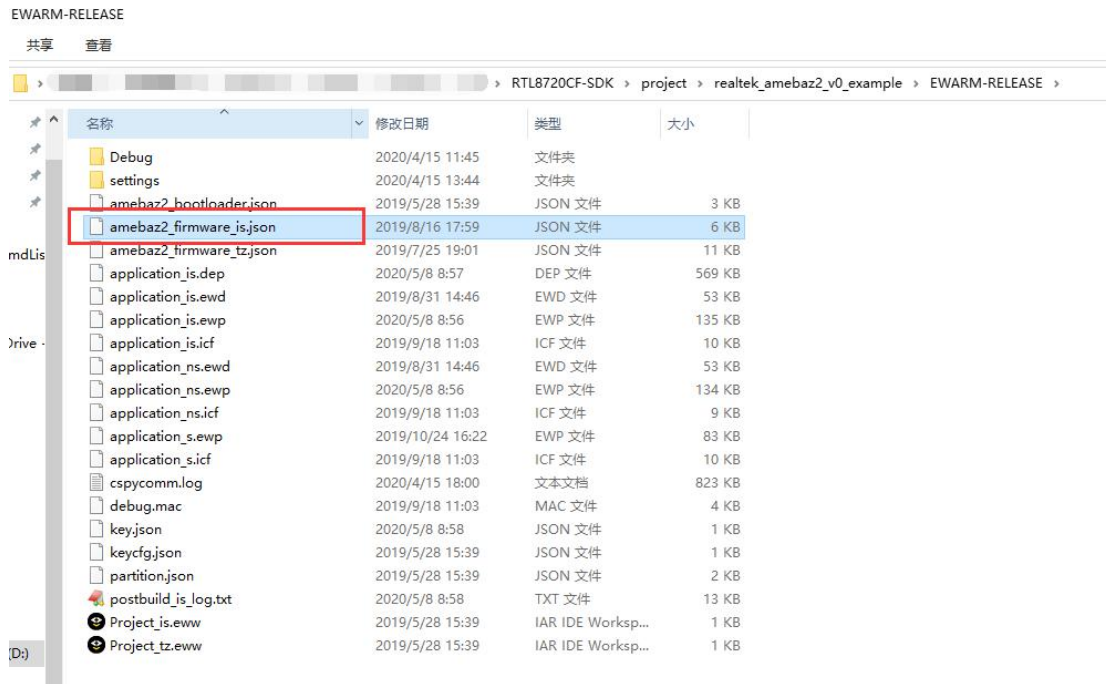
5.1 ATBB-蓝牙配网指令

ATPS =<mode>	
描述	开启/关闭蓝牙配网
响应	开启 [ATBB]:_AT_BT_CONFIG_[ON] 关闭 [ATBB]:_AT_BT_CONFIG_[OFF, deinit BT STACK]
参数	mod: 0:关闭蓝牙配网 1:开启蓝牙配网
例子	参考 附录 3:蓝牙配网流程

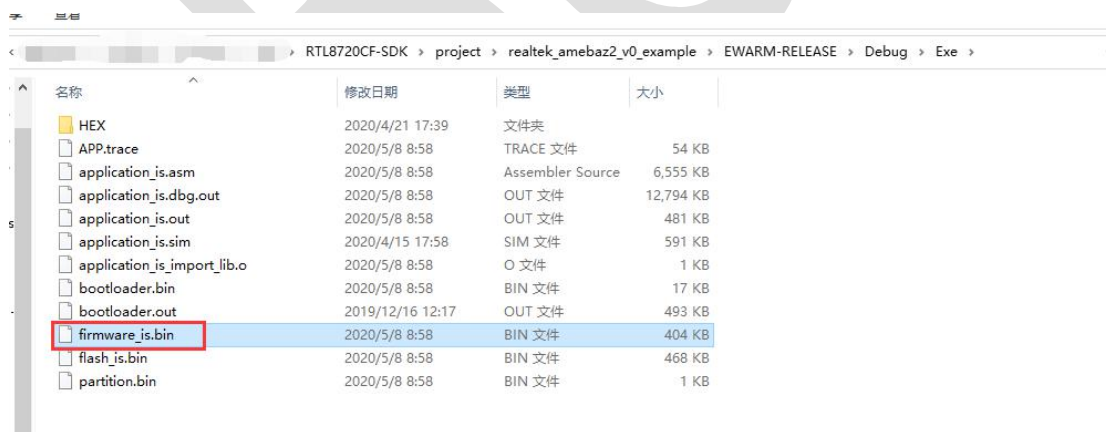
6. 附录

6.1 附录 1: OAT 升级流程

(1) 编辑 `SDK\project\realtek_amebaz2_v0_example\EWARM-RELEASE\amebaz2_firmware_is.json`



(2) 使用 IAR 编译程序 OTA 升级使用 `firmware_is.bin`



```
28  | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 |
    |---|---|---|---|---|---|---|---|---|---|---|
28  |   |   |   |   |   |   |   |   |   |   |   |
29  |   |   |   |   |   |   |   |   |   |   |   |
30  |   |   |   |   |   |   |   |   |   |   |   |
31  |   |   |   |   |   |   |   |   |   |   |   |
32  |   |   |   |   |   |   |   |   |   |   |   |
33  |   |   |   |   |   |   |   |   |   |   |   |
34  |   |   |   |   |   |   |   |   |   |   |   |
35  |   |   |   |   |   |   |   |   |   |   |   |
36  |   |   |   |   |   |   |   |   |   |   |   |
37  |   |   |   |   |   |   |   |   |   |   |   |
38  |   |   |   |   |   |   |   |   |   |   |   |
39  |   |   |   |   |   |   |   |   |   |   |   |
40  |   |   |   |   |   |   |   |   |   |   |   |
41  |   |   |   |   |   |   |   |   |   |   |   |
42  |   |   |   |   |   |   |   |   |   |   |   |
43  |   |   |   |   |   |   |   |   |   |   |   |
44  |   |   |   |   |   |   |   |   |   |   |   |
45  |   |   |   |   |   |   |   |   |   |   |   |
46  |   |   |   |   |   |   |   |   |   |   |   |
47  |   |   |   |   |   |   |   |   |   |   |   |
48  |   |   |   |   |   |   |   |   |   |   |   |

    "__comment_images": "offset = null => cascade ( align to 64 ), should be zero if valid",
    "images": [
        { "img": "FWHS", "offset": "0x00" },
        { "img": "XIP_FLASH_C", "offset": "0x00" },
        { "img": "XIP_FLASH_P", "offset": "0x00" }
    ],
    "FWHS": {
        "source": "Debug/Exe/application_is.out",
        "header": {
            "next": null,
            "__comment_type": "Support Type: PARTAB, BOOT, FWHS_S, FWHS_NS, FWLS, ISP, VOE, WLN, DTCM, ITCM, SRAM",
            "type": "FWHS_S",
            "enc": false,
            "user_key2": "BB0102030405060708090A0B0C0D0E0F101112131415161718191A1B1C1D1E1F",
            "comment_pkey_idx": "assign by program, no need to configurate",
            "serial": 100
        },
        "FST": {
            "__comment_FST0": "enc_algorithm: cbc/ecb with cipher key",
            "__comment_FST1": "validpat is used for section header validation",
        }
    }
}
```

将 serial 更改为比当前值大的数字(这个表示固件的版本号, 程序启动的时候默认启动固件号大的镜像, 如果不修改这个固件号, 那么每次启动的都是镜像 2)

(3) 打开 server 程序

Server 程序在 sdk\tools\DownloadServer\DownloadServer.exe

在 windows cmd 命令行模式下进入 server 目录, 并将要升级的 OTA 文件拷贝到这里

执行 .\DownloadServer.exe 10086 .\firmware_is.bin (10086 是自己定义的一个端口号, OTA3.bin 是升级文件)启动 server

```
>.DownloadServer.exe 10086 image02\firmware_is.bin
c():checksum 0x2aa2770
Listening on port (10086) to send image02\firmware_is.bin (412736 bytes)
Waiting for client ...
```

(4) 模块连接网络并开始下载

使用 ATPN 和 server 连接到同一网络(模块可以访 ping 通 server IP 就可以)

执行 ATSO=192.168.202.10,10086 //10086 是 server 端自己定义的端口号

```
ATPN=Test03,123456798
[Driver]: set ssid [Test03]
[Driver]: start auth to 20:6b:e7:92:3f:90
[Driver]: auth success, start assoc
[Driver]: association success(res=3)
[Driver]: set pairwise key to hw: alg:4(WEP40-1 WEP104-5 TKIP-2 AES-4)
[Driver]: set group key to hw: alg:4(WEP40-1 WEP104-5 TKIP-2 AES-4) keyid:1
Interface 0 IP address : 192.168.202.7
[MEM] After do cmd, available heap 102624

#
ATSO=192.168.202.10,10086
[MEM] After do cmd, available heap 98416

#
[update_ota_local_task] Update task start
[sys_update_ota_prepare_addr]fw1 sn is 100, fw2 sn is 100
[sys_update_ota_prepare_addr] NewFWAddr 00090000
[update_ota_local_task] Read info first
[update_ota_local_task] info 12 bytes
[update_ota_local_task] tx file size 0x64c40
[update_ota_local_task] Current firmware index is 1
[update_ota_erase_upg_region] NewFWLen 412736
[update_ota_erase_upg_region] NewFWBlkSize 101 0x90000
[update_ota_local_task] Start to read data 412736 bytes
[update_ota_local_task] sig_backup for 32 bytes from index 0
.....
Read data finished
[update_ota_signature] Append OTA signature
[update_ota_signature] signature:
AD EF 4C AA 2E A8 AC AC C5 41 9E 64 A0 8F 00 E1
10 F0 27 0F C0 CE 50 5B E2 2D 4B A0 80 E7 49 2D
[update_ota_local_task] Update task exit
[update_ota_local_task] Ready to reboot
-- $18710: I.T Platform --
Chip VID: 5, Ver: 1
ROM Version: v2.0

== Boot Loader ==
Dec 16 2019:12:16:16

Boot Loader <==

== RAM Start ==
Build @ 14:24:23, Apr 15 2020

$8710c>interface 0 is initialized
interface 1 is initialized

Initializing WIFI ...
```

此时 server 端显示如下

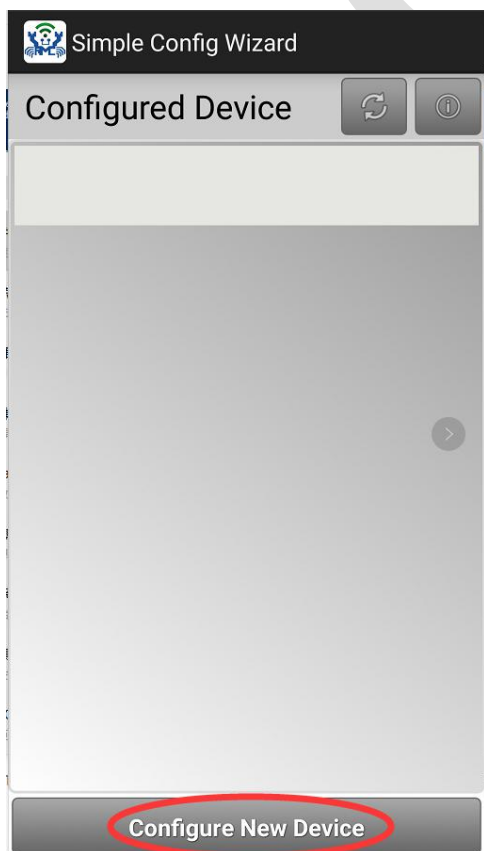
```
>DownloadServer.exe 10086 image02\firmware_is.bin
c():checksum 0x2aa2770
Listening on port (10086) to send image02\firmware_is.bin (412736 bytes)

Waiting for client ...
Accept client connection from 192.168.202.7
Send checksum and file size first
Send checksum byte 12
Sending file...
.....
.....
.....
.....
.....
Total send 412736 bytes
Client Disconnected.
Waiting for client ...
```

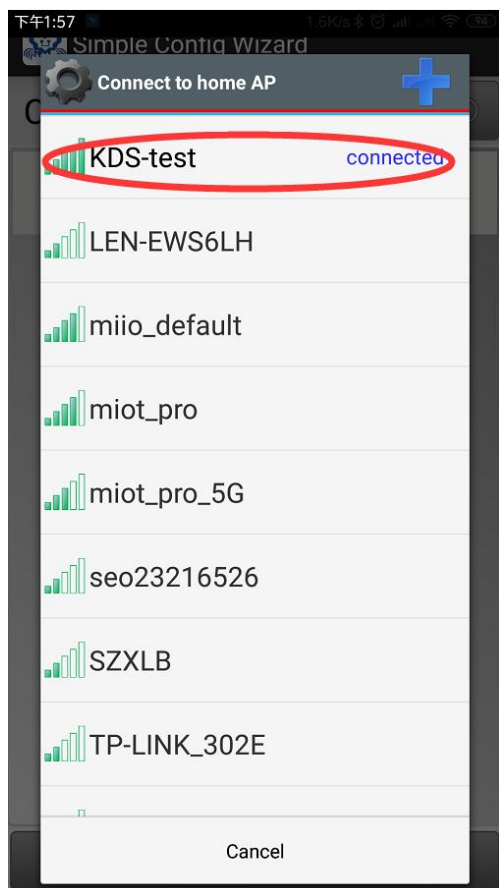
下载完成后 log 串口可以看到 [update_ota_local_task] Ready to reboot. 此时重启模块 OTA 更新就完成了

6.2 附录 2:simple config 配网流程

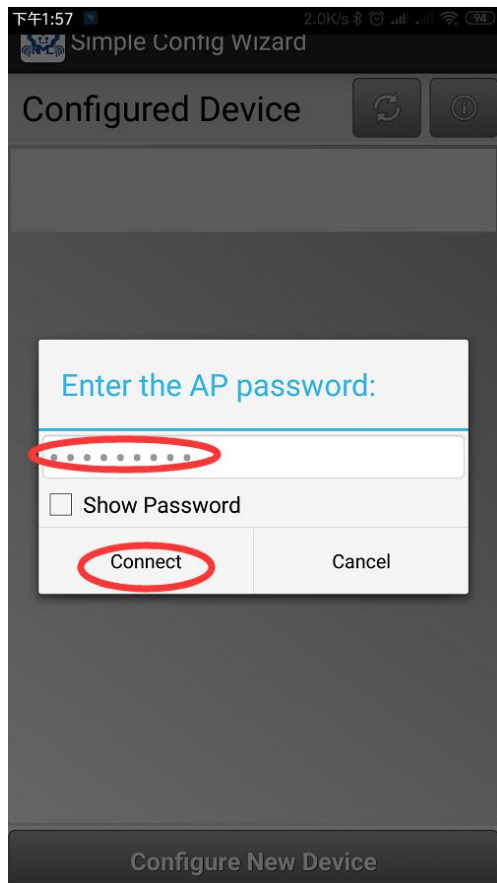
- (1) 手机安装 sdk\tools\simple_config_wizard_3.4b\Android 目录下的配网工具 SimpleConfigWizard_v115.apk
- (2) 手机连接要配网的 wifi
- (3) 点击配置新设备



- (4) 选择要连接的网络



(5) 输入 wifi 密码，点击链接



(6) 此时在模块上输入 ATWQ 进入配网模式，此时 log 口有如下打印

```
[ATWQ]: _AT_WLAN_SIMPLE_CONFIG_
softAP ssid: @RSC-CC55FF00E0, password: 12345678

LwIP_DHCP: dhcp stop.
Deinitializing WIFI ...
WIFI deinitialized
Initializing WIFI ...
WIFI initialized

Switch to channel(2)
Switch to channel(3)
Switch to channel(4)
Switch to channel(5)
Switch to channel(6)
Switch to channel(7)
Switch to channel(8)
Switch to channel(9)
Switch to channel(10)
Switch to channel(11)
```

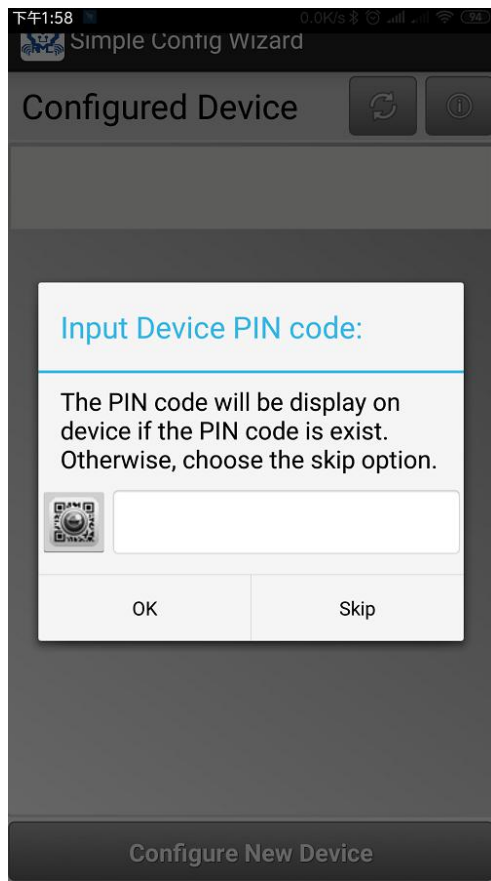
(7) 手机 APP 上点击跳过

表单编号: B&T-QR-EN-002

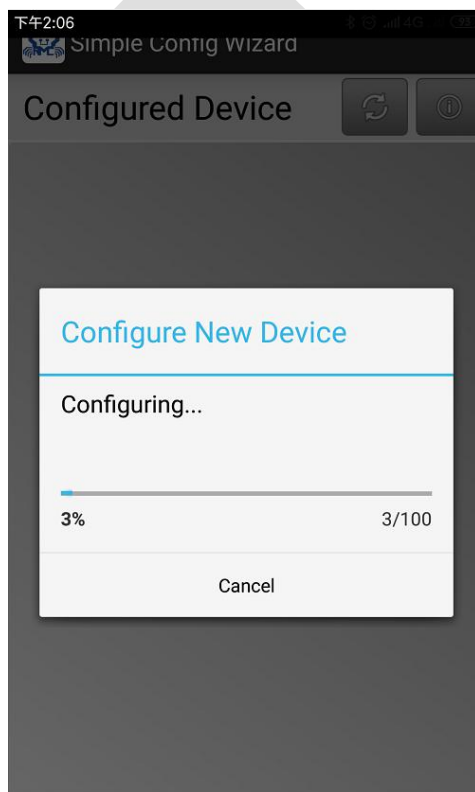
版本: B2

保存期限: 5 年

生效日期: 2014 年 7 月 4 日



(8) 此时进入配网过程



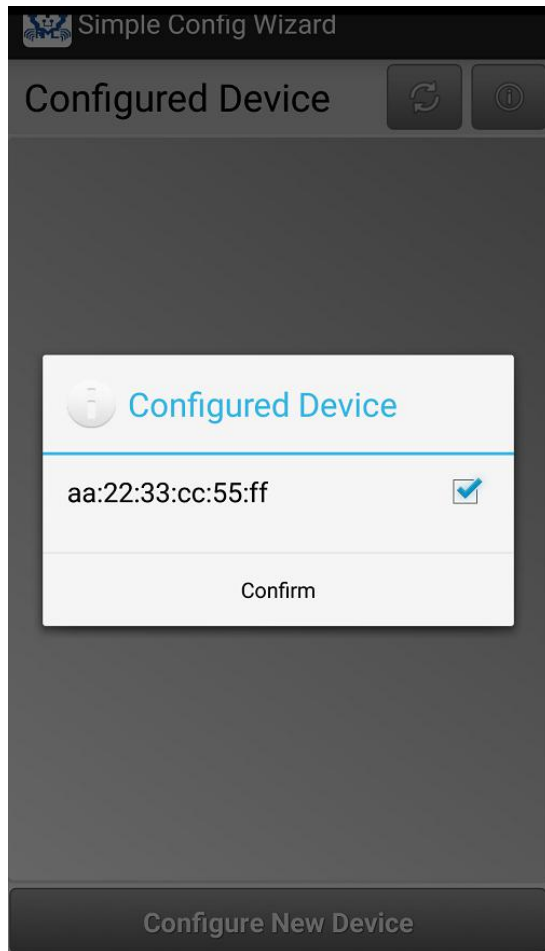
(9) 配网成功后有如下显示

表单编号: B&T-QR-EN-002

版本: B2

保存期限: 5 年

生效日期: 2014 年 7 月 4 日



模块 log 打印如下

```
===== start to connect target AP =====
RTL8721D[Driver]: set ssid [KDS-test]
RTL8721D[Driver]: start auth to 6c:e8:73:aa:2b:e0
RTL8721D[Driver]: auth success, start assoc
RTL8721D[Driver]: association success(res=2)
wlan1: 1 DL RSVD page success! DLBcnCount:01, poll:00000001
RTL8721D[Driver]: set pairwise key to hw: alg:4(WEP40-1 WEP104-5 TKIP-2 AES-4)
RTL8721D[Driver]: set group key to hw: alg:4(WEP40-1 WEP104-5 TKIP-2 AES-4) keyid:2
Interface 0 IP address : 192.168.12.101
Sending simple config ack
recv 92 bytes from 192.168.12.100:8864 at round=0, num=2
Simple Config success
[MEM] After do cmd, available heap 170776
#
```

6.3 附录 3:蓝牙配网流程

(1) 手机安装 SDK\tools\bluetooth\BT Config\Release_Android_BTConfig_APP_v216

(2) 模块端执行 ATBB=1 开启蓝牙配网

```
ATBB=1
[ATBB]:_AT_BT_CONFIG_[ON]
##
```

(3) 手机端打开 APP，点击放大镜开始配网



(4) 如果要连接当前手机连接的网络直接输入密码点击 Continue 就可以，如果要连接其它 AP 可以选择 Slecet AP 选择需要连接的网络

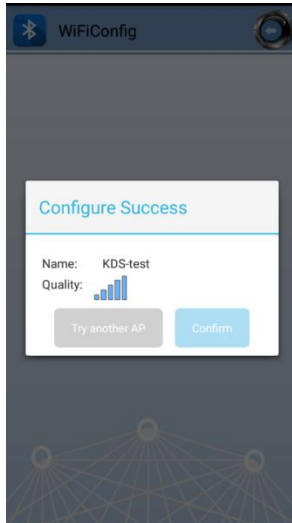


此时的 log

```
[BT Config Wifi] Bluetooth Connection Established
[BT Config Wifi] Band Request
[BT Config Wifi] Scan Request
[BT Config Wifi] Scan 2.4G AP

[BT Config Wifi] Scan Request
[BT Config Wifi] Scan 5G AP
```

(5) 连接成功后 APP 端显示如下



模块端 log 如下

```
[BT Config Wifi] Connect Request
RTL8721D[Driver]: set BSSID: [REDACTED]
RTL8721D[Driver]: set ssid [KDS-test]
RTL8721D[Driver]: start auth to 4 [REDACTED]
RTL8721D[Driver]: auth success, start assoc
RTL8721D[Driver]: association success(res=17)
wlan1: 1 DL RSVD page success! DLBcnCount:01, poll:00000001
RTL8721D[Driver]: set pairwise key to hw: alg:4(WEP40-1 WEP104-5 TKIP-2 AES-4)
RTL8721D[Driver]: set group key to hw: alg:4(WEP40-1 WEP104-5 TKIP-2 AES-4) keyid:1
[BT Config Wifi] Connected after 3680ms.
Interface 0 IP address : 192.168.43.29
[BT Config Wifi] Got IP after 4782ms.
```