

|  |  |
| --- | --- |
| **참여자** | **역할 분배** |
| 조승준 | Node.js, TensorFlow(얼굴인식), python, Java |
| 박재환 | Node.js, Java, Javascript |
| 정태양 | Node.js, CSS, mySQL, PHP |
| 주종원 | TensorFlow (번호판인식), C++, ATmega2560 |
| 노하늘 | 프로그래밍 언어 |
| 천예준, 노예원, 김형철 | Java, 안드로이드 개발 |
| 지소민, 심규리, 이시온 | 적외선(수, 발광)센서를 이용한 high & low 회로설계  NE555를 이용한 회로 설계  마이크, 스피커 회로 설계 데이터시트 조사 및 해석  OrCAD 이용 |
| 김경민 | 프로젝트에 사용 가능한 통신 모듈 2가지 조사  통신 최신 기술 동향 파악 |
| 김은서, 남수연, 최미란 | 납땜 작업 |
| 최주영, 김민희 | USART 코드 작성, ATmega128 핀, 레지스터 조사 |

◈ 프로젝트 목적

◈ 학습 계획

|  |  |
| --- | --- |
| **참여자** | **학습 계획** |
| 조승준 | Node js 교재 학습, 파이썬(TF), JAVA(NIO2/자바반) |
| 박재환 | HTML, JAVA, JAVA Script, Node js 인터넷 강의 학습 |
| 정태양 | 웹 전반, Node js 인터넷 강의 학습 |
| 주종원 | JAVA(자바반) |
| 노하늘 |  |
| 천예준, 노예원, 김형철 | 안드로이드 스튜디오 학습 |
| 지소민, 심규리, 이시온 |  |
| 김경민 |  |
| 김은서, 남수연, 최미란 |  |
| 최주영, 김민희 |  |

◈ 진행 상황

* 2017/12/26

프로젝트 역할 분배, 밴드 모임 개설

* 2018/02/05

김민혁, 정태양, 조승준

스터디 개설 오후 2시 ~ 5시

장소 : 카페 쉼표 2층 대여공간

* 2018/02/09

조승준, 정태양, 박재환, 주종원, 김형철

두 번째 스터디 오전 10시 30분 ~ 12시 30분

장소 : 인천전자마이스터고 방송부실

* 2018/02/14

조승준, 정태양, 김민혁, 김기태

세 번째 스터디 오후 2시 ~ 오후 5시

|  |  |  |
| --- | --- | --- |
| **이름** | **목적** | **성과** |
| **조승준** | 깃허브 설명 프로젝트 설계 |  |
| **김민혁** | Nodejs 복습 및 코딩환경 설정 |  |
| **노예원** |  |  |
| **정태양** | 프로젝트 설계 |  |
| **주종원** |  |  |
| **노하늘** |  |  |

문서 번역

# esp\_wifi\_repeater

A full functional WiFi repeater (correctly: a WiFi NAT router)

This is an implementation of a WiFi NAT router on the esp8266 and esp8285. It also includes a basic firewall with ACLs, port mapping, traffic shaping, hooks for remote monitoring (or packet sniffing), MQTT management interface, and power management. For a setup with multiple routers in a mesh to cover a larger area a new mode "Automesh" has been included <https://github.com/martin-ger/esp_wifi_repeater#automesh-mode> .

esp\_wipi\_repeater  
전체 기능을 갖춘 WiFi리피터(WiFiNAT라우터)  
이는 esl0266 및 es. 285에서 WiFiNAT라우터를 구현한 것입니다. 또한 ACL, 포트 매핑, 트래픽 조절, 원격 모니터링을 위한 후크(또는 패킷 스니핑), pgTT관리 인터페이스 및 전원 관리 기능이 포함된 기본 방화벽도 포함하고 있습니다. 더 큰 영역을 커버하기 위해 메시에 여러개의 라우터가 있는 설정의 경우, 새로운 모드"자동 메쉬"가 https://github.com/martin-ger/esp\_wifi\_repeater#automesh-mode에 포함되었습니다.

Typical usage scenarios include:

* Simple range extender for an existing WiFi network
* Setting up an additional WiFi network with different SSID/password for guests or IoT devices
* Battery powered outdoor (mesh) networks
* Monitor probe for WiFi traffic analysis

The ESP acts as STA and as soft-AP and transparently forwards any IP traffic through it. As it uses NAT no routing entries are required neither on the network side nor on the connected stations. Stations are configured via DHCP by default in the 192.168.4.0/24 net and receive their DNS responder address from the existing WiFi network.

Measurements show, that it can achieve about 5 Mbps in both directions, so even streaming is possible.

Some details are explained in this video: <https://www.youtube.com/watch?v=OM2FqnMFCLw>

일반적인 사용 시나리오는 다음과 같습니다.  
기존 WiFi네트워크를 위한 간단한 범위 익스텐더  
게스트 또는 IoT디바이스를 위해 서로 다른 SSID0암호를 가진 추가 WiFi네트워크를 설정하는 중  
배터리 구동식 실외(메시)네트워크  
WiFi트래픽 분석을 위한 모니터 프로브  
ESP는 STA및 소프트 CHAP로 작동하며 이 기능을 통해 모든 IP트래픽을 투명하게 전달합니다. NAT을 사용하기 때문에 네트워크 측에서나 연결된 스테이션에서나 라우팅 항목이 필요하지 않습니다. 스테이션은 기본적으로 192.168.4.0 IP/24네트워크에서 DHCP를 통해 구성되며 기존 WiFi네트워크에서 해당 DNS응답자 주소를 받습니다.  
측정 결과에 따르면 양 방향에서 약 5Mbps까지 달성할 수 있어 스트리밍도 가능한 것으로 나타났다.  
이 비디오에는 몇가지 세부 정보가 설명되어 있습니다. <https://www.youtube.com/watch?v=OM2FqnMFCLw>

### Note on WPA2 enterprise (PEAP)

If you need a "converter" that translates a WPA2 enterprise network with PEAP authentication into a WPA2-PSK network, have a look at <https://github.com/martin-ger/esp_peap_psk> . Was trying to integrate this functionality into this project - however this is difficult, as WPA2 enterprise requires so much free heap during authentication that there is hardly any mem left for anything else. So I decided to leave that in a separate project.

### Note on WPA2 KRACK security issue

The lastest firmware (after 17/Oct/2017) has been build with the patched version of the SDK 2.1.0 from Espressif that mitigates the KRACK (<https://www.krackattacks.com/> ) attack.

WPA2엔터프라이즈에 대한 참고 사항(PEAP)  
PEAP인증을 통해 WPA2엔터프라이즈 네트워크를 WPA-PSK네트워크로 변환하는 "변환기"가 필요한 경우 https://github.com/martin-ger/esp\_peap\_psk를 참조하십시오. 이 기능을 이 프로젝트에 통합하려고 했는데, 하지만 WPA2엔터프라이즈가 인증 중에 사용 가능한 힙을 너무 많이 필요로 하여 다른 메모리가 거의 남아 있지 않기 때문에 이 작업은 어렵습니다. 그래서 저는 그것을 다른 프로젝트에 남겨두기로 했습니다.  
WPA2CRACK보안 문제에 대한 참고 사항  
최신 펌웨어 ( af te r . Oc / 2017)은 17ACK (https://www.krackattacks.com/ 공격을 완화시키는 Espressif의 SDK2.1.0 패치 버전으로 제작되었습니다.

# First Boot

The esp\_wifi\_repeater starts with the following default configuration:

* ap\_ssid: MyAP, ap\_password: none, ap\_on: 1, ap\_open: 1
* network: 192.168.4.0/24

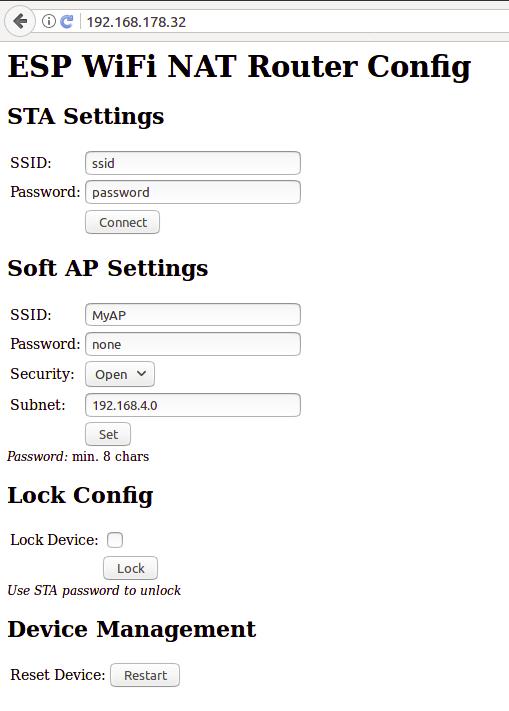
After first boot (or factory reset) it will offer a WiFi network with an open AP and the ssid "MyAP". It does not yet try to automatically re-connect to an uplink AP (as it does not know a valid ssid or password).

Connect to this WiFi network and do the basic configuration either via a simple web interface or the full config with all options via the console.

# Basic Web Config Interface

The web interface allows for the configuration of all parameters required for the basic forwarding functionality. Thanks to rubfi for the major work on that: <https://github.com/rubfi/esp_wifi_repeater/> . Point your browser to "[http://192.168.4.1](http://192.168.4.1/)". This page should appear:

퍼스트 부트  
esp\_wipi\_repeater는 다음 기본 구성으로 시작합니다.  
ap\_ssid:MyAP, ap\_password:no, ap\_on:1, ap\_open:1  
네트워크:192.168.4.0 v/24  
처음 부팅(또는 공장 초기화)후에는 개방형 AP와 ssid"MyAP"가 포함된 WiFi네트워크를 제공합니다. 올바른 ssid또는 비밀 번호를 모르므로 아직 업링크 AP에 자동으로 다시 연결하려고 시도하지 않습니다.  
이 WiFi네트워크에 연결하고 간단한 웹 인터페이스를 통해 기본 구성을 하거나 콘솔을 통해 모든 옵션을 사용하여 전체 구성을 수행합니다.  
기본 웹 구성 인터페이스  
웹 인터페이스를 통해 기본 전달 기능에 필요한 모든 파라미터를 구성할 수 있습니다. 주요한 일에 대한 루피 덕분입니다. https://github.com/rubfi/esp\_wifi\_repeater/ 브라우저에서 "http://192.168.4.1"을 가리킵니다. 이 페이지가 나타납니다.



First enter the appropriate values for the uplink WiFi network, the "STA Settings", and click "Connect". Use password "none" for open networks. The ESP reboots and will connect to your WiFi router. The status LED should be blinking after some seconds.

Now you can reload the page and change the "Soft AP Settings". Click "Set" and again the ESP reboots. Now it is ready for forwarding traffic over the newly configured Soft AP. Be aware that these changes also affect the config interface, i.e. to do further configuration, connect to the ESP through one of the newly configured WiFi networks. For access through the Soft AP remember the address of the Soft APs network if you have changed that (the ESP has always the address x.x.x.1 in this network).

If you like, you can mark the "lock" checkbox and click "Lock". Now the config cannot be changed anymore without first unlocking it with the uplink WiFi network's password (define one even if the network is open).

If you did a mistake and you lost any contact with ESP you can still use the serial console to recover it ("reset facory", see below).

먼저 업링크 WiFi네트워크,"STA설정"에 적절한 값을 입력하고"연결"을 클릭합니다. 열려 있는 네트워크에 대해서는 암호"none"을 사용합니다. ESP가 재부팅되고 WiFi라우터에 연결됩니다. 상태 LED가 몇초 후에 깜박입니다.  
이제 페이지를 다시 로드하여"소프트 AP설정"을 변경할 수 있습니다. 설정을 클릭하고 ESP재부팅을 다시 수행한다. 이제 새로 구성된 소프트 AP를 통해 트래픽을 전달할 준비가 되었습니다. 이러한 변경 사항도 구성 인터페이스에 영향을 미치므로, 추가 구성을 위해 새로 구성된 WiFi네트워크 중 하나를 통해 ESP에 연결하십시오. 소프트 AP를 통해 접속하려면 소프트 AP네트워크의 주소를 변경한 경우 이를 기억한다(ESP의 주소 x.x.1은 항상 이 네트워크에 있음).  
원하는 경우"잠금"확인 란을 표시하고"잠금"을 클릭할 수 있습니다. 이제 업링크 WiFi네트워크의 비밀 번호로 잠금을 해제하지 않고 구성을 더 이상 변경할 수 없습니다(네트워크가 열려 있는 경우에도 구성을 정의).  
실수로 ESP와 연락이 두절된 경우에도 직렬 콘솔을 사용해 이 콘솔을 복구할 수 있습니다(아래"초기화 상태"참조).

**Command Line Interface**

Advanced configuration has to be done via the command line on the console interface. This console is available either via the serial port at 115200 baud or via tcp port 7777 (e.g. "telnet 192.168.4.1 7777" from a connected STA).

Use the following commands for an initial setup:

* set ssid your\_home\_router's\_SSID
* set password your\_home\_router's\_password
* set ap\_ssid ESP's\_ssid
* set ap\_password ESP's\_password
* show (to check the parameters)
* save
* reset

If you want to enter non-ASCII or special characters on the command line you can use quoting: either use C-style quotes with backslash like this "My\ AccessPoint" or use HTTP-style hex encoding like "My%20AccessPoint". Both methods will result in a string "My AccessPoint". With the hex encoding you can enter any byte value you like, except for 0 (for C-internal reasons).

The command line understands a lot more commands:

명령줄 인터페이스  
고급 구성은 콘솔 인터페이스의 명령줄을 통해 수행해야 합니다. 이 콘솔은 115200 보드의 시리얼 포트를 통해서 나 tcp포트 7777(예:연결된 STA의 "telnet192.168.4.1 7777")을 통해 사용합니다.  
초기 설정에는 다음 명령을 사용합니다.  
ssidYou\_home\_라우터\_SSID를 설정합니다.  
암호 설정 You\_home\_louters\_password  
ap\_ssidESP's\_siid설정  
ap\_passwordESP\_password설정  
표시(파라미터 확인)  
절약하다  
재설정  
명령줄에 ASCII가 아닌 문자나 특수 문자를 입력하려면 다음과 같은 인용문을 사용할 수 있습니다."My\AccessPoint"와 같은 백 슬래시를 사용하거나 HT스타일의 "hex"를 사용하십시오. 두 방법 모두"MyAccessPoint"문자열로 이어집니다. 16진수 인코딩을 사용하면 0(C-내부적인 이유로)을 제외하고 원하는 바이트 값을 입력할 수 있습니다.  
명령줄은 훨씬 더 많은 명령을 이해합니다.

## Basic commands

Enough to get it working in nearly all environments.

* help: prints a short help message
* set [ssid|password] value: changes the settings for the uplink AP (WiFi config of your home-router), use password "none" for open networks.
* set [ap\_ssid|ap\_password] value: changes the settings for the soft-AP of the ESP (for your stations)
* show [config|stats]: prints the current config or some status information and statistics
* save [dhcp]: saves the current config parameters [+ the current DHCP leases] to flash
* lock [password]: saves and locks the current config, changes are not allowed. Password can be left open if already set before (Default is the password of the uplink WiFi)
* unlock password: unlocks the config, requires password from the lock command
* reset [factory]: resets the esp, 'factory' optionally resets WiFi params to default values (works on a locked device only from serial console)
* quit: terminates a remote session

## Advanced commands

Most of the set-commands are effective only after save and reset.

### Automesh config

* set automesh [0|1]: selects, whether the automesh mode is on or off (default), see details here <https://github.com/martin-ger/esp_wifi_repeater#automesh-mode>
* set am\_threshold dB: sets the threshold for a "bad" connection (in negative dB, default 85, i.e. -85 dB)
* set am\_scan\_time secs: sets the time interval in seconds the ESP tries in automesh mode to find an uplink AP before going to sleep (0 disabled, default)
* set am\_sleep\_time secs: sets the time interval in seconds the ESP sleeps in automesh mode if no uplink AP is found (0 disabled, default)

### WiFi config

* set ap\_on [0|1]: selects, whether the soft-AP is disabled (ap\_on=0) or enabled (ap\_on=1, default)
* set ap\_open [0|1]: selects, whether the soft-AP uses WPA2 security (ap\_open=0, automatic, if an ap\_password is set) or open (ap\_open=1)
* set auto\_connect [0|1]: selects, whether the STA should keep retrying to reconnect to the AP. auto\_connect is off (0) after first flashing or after "reset factory". When you enter a new SSID it will be automatically set on (1).
* set ssid\_hidden [0|1]: selects, whether the SSID of the soft-AP is hidden (ssid\_hidden=1) or visible (ssid\_hidden=0, default)
* set phy\_mode [1|2|3]: sets the PHY\_MODE of the WiFi (1=b, 2=g, 3=n(default))
* set bssid xx:xx:xx:xx:xx:xx: sets the specific BSSID of the uplink IP to connect to (default 00:00:00:00:00:00 which means any)
* set [ap\_mac|sta\_mac] xx:xx:xx:xx:xx:xx: sets the MAC address of the STA and SOFTAP to a user defined value (bit 0 of the first byte of the MAC address can not be 1)
* set sta\_hostname name: sets the name of the STA (visible to the uplink AP)
* scan: does a scan for APs
* connect: tries to connect to an AP with the currently configured ssid and password
* disconnect: disconnects from any uplink AP

### TCP/IP config

* ping ip-addr: checks IP connectivity with ICMP echo request/reply
* set network ip-addr: sets the IP address of the internal network, network is always /24, router is always x.x.x.1
* set dns dns-addr: sets a static DNS address that is distributed to clients via DHCP
* set dns dhcp: configures use of the dynamic DNS address from DHCP, default
* set ip ip-addr: sets a static IP address for the ESP in the uplink network
* set ip dhcp: configures dynamic IP address for the ESP in the uplink network, default
* set netmask netmask: sets a static netmask for the uplink network
* set gw gw-addr: sets a static gateway address in the uplink network
* portmap add [TCP|UDP] external\_port internal\_ip internal\_port: adds a port forwarding
* portmap remove [TCP|UDP] external\_port: deletes a port forwarding

### Firewall/Monitor config

* acl [from\_sta|to\_sta] [TCP|UDP|IP] src-ip [src\_port] desr-ip [dest\_port] [allow|deny|allow\_monitor|deny\_monitor]: adds a new rule to the ACL
* acl [from\_sta|to\_sta] clear: clears the whole ACL
* show acl: shows the defined ACLs and some stats
* set acl\_debug [0|1]: switches ACL debug output on/off - a denied packets will be logged to the terminal
* set [upstream\_kbps|downstream\_kbps] bitrate: sets a maximum upstream/downstream bitrate (0 = no limit)
* monitor [on|off|acl] port: starts and stops monitor server on a given port

### Interface config

* set config\_port portno: sets the port number of the console login (default is 7777, 0 disables remote console config)
* set web\_port portno: sets the port number of the web config server (default is 80, 0 disables web config)
* set config\_access mode: controls the networks that allow config access for console and web (0: no access, 1: only internal, 2: only external, 3: both (default))

### Chip config

* set speed [80|160]: sets the CPU clock frequency (default 80 Mhz)
* sleep seconds: Put ESP into deep sleep for the specified amount of seconds. Valid values between 1 and 4294 (aprox. 71 minutes)
* set status\_led GPIOno: selects a GPIO pin for the status LED (default 2, >16 disabled)
* set vmin voltage: sets the minimum battery voltage in mV. If Vdd drops below, the ESP goes into deep sleep. If 0, nothing happens
* set vmin\_sleep secs: sets the time interval in seconds the ESP sleeps on low voltage

기본 명령어  
거의 모든 환경에서 이를 활용할 수 있는 충분한 성능.  
Help: 짧은 도움말 메시지를 출력합니다.  
[ssid.password]값을 설정합니다. 업링크 AP에 대한 설정을 변경하고(홈 라우터의 WiFi구성), 열려 있는 네트워크에 대해 암호" 없음"을 사용합니다.  
[ap\_side|ap\_password]값 설정:ESP의 소프트 CHAP에 대한 설정을 변경합니다(사용자의 스테이션에 해당).  
[config|stats]표시:현재 구성 또는 일부 상태 정보 및 통계를 인쇄합니다.  
저장[DHCP]:현재 구성 매개 변수[+현재 DHCP리스]를 플래시에 저장합니다.  
Lock[password]:현재 구성을 저장하고 잠급니다. 변경은 허용되지 않습니다. 이미 설정된 경우 비밀 번호를 열어 둘 수 있습니다(기본 값은 업링크 WiFi의 비밀 번호임).  
잠금 해제 비밀 번호:구성을 잠금 해제하고 잠금 명령의 비밀 번호 필요  
재설정[공장]:esp,'공장'을 선택적으로 재설정합니다. WiFi매개 변수는 기본 값으로 재설정합니다(시리얼 콘솔에서 잠금 장치에서만 작동).  
종료:원격 세션을 종료합니다.  
고급 명령어  
대부분의 설정 명령은 저장하고 재설정한 후에만 유효합니다.  
자동 메쉬 구성  
자동 메쉬 설정[0x11]:를 선택하고, 자동 메쉬 모드가 켜져 있든 꺼져 있든 상관 없이(기본 값), 자세한 내용은 여기 https://github.com/martin-ger/esp\_wifi\_repeater#automesh-mode을 참조하십시오.  
A\_thresholddB설정:" 잘못된 "연결에 대한 임계값을 설정합니다(음 dB, 기본 값 85, 즉-85dB).  
A\_scan\_timesecs설정:유휴 상태가 되기 전에 ESP가 업링크 AP를 찾기 위해 자동 메쉬 모드에서 시도하는 시간 간격(초)을 설정합니다(0으로 설정됨).  
A\_sleep\_timesecs설정:업링크 AP가 없는 경우 자동 메쉬 모드에서 ESP가 대기하는 시간 간격(초)을 설정합니다(0은 비활성화됨, 기본 값).  
무선 인터넷 환경 설정  
ap\_on[0a1]설정:소프트 CHAP의 사용 여부(ap\_on=0)또는 사용(ap\_on=1, 기본 값)을 선택합니다.  
ap\_open[0a1]설정:소프트 CHAP가 WPA2보안을 사용하는지(ap\_open=0, ap\_password가 설정된 경우 자동)또는 열기(ap\_open1)를 선택합니다.  
auto\_connect[0v01]을 설정합니다.를 선택하면, STA가 AP에 다시 연결하기 위해 계속 재시도해야 하는지 여부를 선택합니다. 새 SSID를 입력하면(1)에 자동으로 설정됩니다.  
ssid\_hedden[0/0/1]을 설정합니다. 소프트 CHAP의 SSID(ssid\_hedden=1)또는 표시되는지(ssid\_hedden=0, 기본 값)  
phy\_mode[1/2/3]설정:WiFi의 PHY\_MODE를 설정합니다(1=b, 2=g, 3=n).  
bssid xxx:xxx:xxx:xx:xx:xx:xx:는 연결할 업링크 IP의 특정 BSSID를 설정합니다(기본 값:00:00:00:00:00:00, 즉  
[ap\_mac\_sta\_mac]xxx:xxx:xxx:xxx:xx:xx:xx://STA및 SOFTAP의 MAC주소를 사용자 정의 값으로 설정합니다(첫번째 MAC주소의 비트 0은 바이트가 될 수 없음).  
sta\_hostname이름 설정:STA이름을 설정합니다(업링크 AP에 표시됨).  
scan:AP를 검색합니다.  
연결:현재 구성된 ssid및 비밀 번호를 사용하여 AP에 연결하려고 시도합니다.  
Disconnect(연결 끊김):모든 업링크 AP와의 연결이 끊김  
TCP/IP구성  
Ping-adad:ICMPEchoRequestest와 IP연결을 확인하는 중  
네트워크 IP-adad설정:내부 네트워크의 IP주소를 설정합니다. 네트워크는 항상/24이며 라우터는 항상 x.x.x.1입니다.  
Setdns-ad-Gand:DHCP를 통해 클라이언트에 배포되는 정적 DNS주소를 설정합니다.  
SetmdsDHCP:DHCP에서 동적 DNS주소 사용을 구성합니다. 기본 값  
IPIP-adad설정:업링크 네트워크에서 ESP에 대한 정적 IP주소를 설정합니다.  
SetIPexe:업링크 네트워크의 ESP에 대해 동적 IP주소를 구성합니다(기본 값).  
넷 마스크 설정:업링크 네트워크에 대해 정적 넷 마스크를 설정합니다.  
Guk-adad설정:업링크 네트워크에서 정적 게이트 웨이 주소를 설정합니다.  
portmapadd[TCP/UDP/UDP]external\_port\_ip내장\_port:포트 포워딩을 추가합니다.  
포트 맵 제거[TCP/UDP]external\_port:포트 포워딩 삭제  
방화벽 /Monitor 구성  
acl[from\_sta/to\_sta][TCP/UDP/UDP/IP]패킷[src\_port]에[dest\_port]의 다음 규칙을 추가합니다.[allow]  
acb[from\_sta/to\_sta] 지우기:전체 ACL을 지웁니다.  
Showatb:정의된 ACL및 일부 통계를 보여 줍니다.  
acl\_debug[0a1]설정:스위치 ACL디버그 출력 켜기/ 끄기-거부된 패킷이 터미널에 기록됨  
[upstream\_vcps| downstream\_scream]비트 전송률 설정:최대 가동/정지 속도 설정(0=제한 없음)  
[on](on)/off( 끄기)/페이지 포트:지정된 포트에서 모니터 서버 시작 및 중지  
인터페이스 구성  
config\_portportno설정:콘솔 로그인의 포트 번호를 설정합니다(기본 값은 7777이고, 0은 원격 콘솔 구성을 비활성화함).  
웹\_portportno설정:웹 구성 서버의 포트 번호를 설정합니다(기본 값:80,0은 웹 구성을 실행 중지함).  
config\_access모드 설정:콘솔과 웹에 대한 구성 액세스를 허용하는 네트워크를 제어합니다(0:액세스 불가, 1:내부에만 해당, 2:외부에만 해당, 3:둘 다(기본 값))

.

칩 구성  
설정 속도[801/160]:CPU클럭 주파수(기본 값 80MHz)를 설정합니다.  
슬립 초:ESP를 지정된 시간(초)동안 깊은 잠에 빠뜨립니다. 유효한 값은 1에서 4294 사이입니다. 71분)  
Setstatus\_ledGPIOno:상태 LED(기본 값 2,>16비활성화됨)에 대한 GPIO핀을 선택합니다.  
Setvmin전압:최소 배터리 전압을 mV로 설정합니다. 만약에 vDisk가 아래로 떨어지면 ESP는 깊은 잠에 빠진다. 0이면 아무 일도 일어나지 않습니다.  
Setvmin\_sleepsecs:ESP가 저전압에서 대기하는 시간 간격(초)을 설정합니다.

https://github.com/martin-ger/esp\_wifi\_repeater