安信可ESP32-CAM摄像头开发demo--局域网拍照、实时视频、人脸识别

原创 安信可科技 于 2020-08-14 15:57:36 发布 42467 收藏 291 版权

分类专栏： 安信可 ESP32 专题 文章标签： esp32-cam 人脸识别 ov2640 摄像头监控

安信可 ESP32 专题 专栏收录该内容

ESP32 -CAM内嵌网页局域网通信

目录

目录

一、前言

ESP32-CAM是安信可发布小尺寸的摄像头模组。该模块可以作为最小系统独立工作，尺寸仅为2740.54.5mm。 ESP32-CAM可广泛应用于各种物联网场合，适用于家庭智能设备、工业无线控制、无线监控、人脸识别 以及其它物联网应用，是物联网应用的理想解决方案。 ESP32-CAM采用DIP封装，直接插上底板即可使用，实现产品的快速生产，为客户提供高可靠性的连接方式，方便应用于各种物联网硬件终端场合

准备工作

1. ESP_CAM模块开发板 链接
2. 串口模块（USB-TTL）
3. 杜邦线

4. PC串口调试助手
5. 硬件接线
6. 请保证模块输入电源至少5V 2A,否则图片会有几率出现水纹。

二、环境搭建及编译

2.1 Linux环境安装

本文开发环境基于Linux，环境搭建步骤如下：

1. 下载安装 VM 虚拟机，点我下载
   和谐码：FC7D0-D1YDL-M8DXZ-CYPZE-P2AY6（仅学习使用）
2. 下载镜像，这里选择ubuntu桌面版18.04.4版本，点我下载。
3. 安装Ubuntu ,安装教程参考

我们还需要安装几个常用的软件：

```
sudo apt-get purge vim-common
sudo apt-get install vim
sudo apt install git
```

然后安装一堆依赖环境

```
sudo apt-get install git wget flex bison gperf python python-pip python-setuptools python-serial python-click python-cryptography python-future python-pyparsing python-pyelftools cmake nin
```

2.2 代码获取

```
git clone --recurse-submodules https://github.com/Ai-Thinker-Open/Ai-Thinker-Open_ESP32-CAMERA_LAN.git
```

网络没翻墙的情况下克隆失败，用以下命令克隆：

```
git clone --recurse-submodules https://github.com.cnpmjs.org/Ai-Thinker-Open/Ai-Thinker-Open_ESP32-CAMERA_LAN.git
```

子模块拉取完后我们可以看见Ai-Thinker-Open_ESP32-CAMERA_LAN里面有个esp-idf，此版本为v4.0。

设置 IDF_PATH 路径
以我的环境为例：

```
export IDF_PATH=/mnt/hgfs/share/Ai-Thinker-Open_ESP32-CAMERA_LAN/esp-idf
```

之后按下 i 表示嵌入代码： vim ~/.bashrc
任意一处添加 表示嵌入上面代码！

按下esc 再 :wq 表示写入保存： source ~/.bashrc

IDF_PATH 路径测试是否设置成功： echo $IDF_PATH

除了 ESP-IDF 本身，您还需要安装 ESP-IDF 使用的各种工具，比如编译器、调试器、Python 包等!

```
cd /mnt/hgfs/share/Ai-Thinker-Open_ESP32-CAMERA_LAN/esp-idf    //以我的路径为例
./install.sh
```

出现这个 , 表示 安装成功

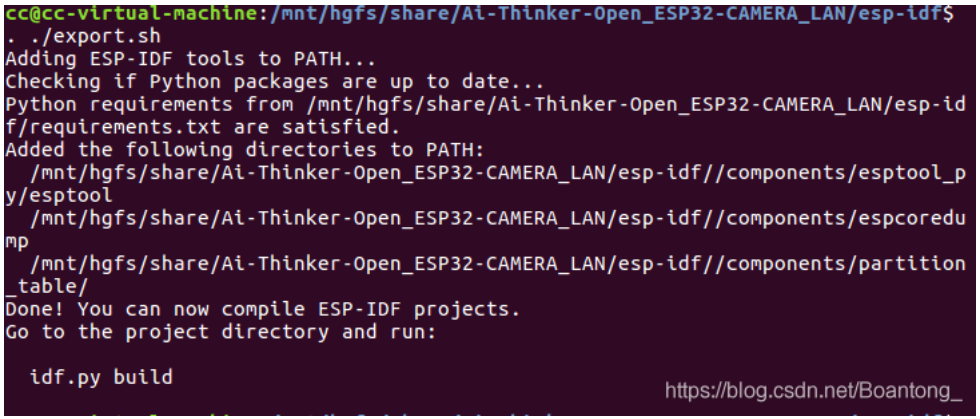
```
Installing Python packages from /home/xuhongv/ESPRESSIF/ESP32/esp-idf/requirements.txt
DEPRECATION: Python 2.7 reached the end of its life on January 1st, 2020. Please upgrade your Python as Python 2.7 is no longer maintained. A future version of pip will drop suppor
etails about Python 2 support in pip, can be found at https://pip.pypa.io/en/latest/development/release-process/#python-2-support
WARNING: The directory '/home/xuhongv/.cache/pip' or its parent directory is not owned or is not writable by the current user. The cache has been disabled. Check the permissions an
y. If executing pip with sudo, you may want sudo's -H flag.
Ignoring None: markers 'sys_platform == "win32"' don't match your environment
Requirement already satisfied: setuptools>=21 in /home/xuhongv/.espressif/python_env/idf4.2_py2.7_env/lib/python2.7/site-packages (from -r /home/xuhongv/ESPRESSIF/ESP32/esp-idf/req
(44.0.0)
Collecting click>=5.0
  Downloading click-7.1.1-py2.py3-none-any.whl (82 kB)
     |████████████████████████████████| 82 kB 18 kB/s
Collecting pyserial>=3.0
  Downloading pyserial-3.4-py2.py3-none-any.whl (193 kB)
     |████████████████████████████████| 193 kB 11 kB/s
Collecting future>=0.15.2
  Downloading future-0.18.2.tar.gz (829 kB)
     |████████████████████████████████| 829 kB 21 kB/s
Collecting cryptography>=2.1.4
  Downloading cryptography-2.8-cp27-cp27mu-manylinux2010_x86_64.whl (2.3 MB)
     |████████████████████████████████| 2.3 MB 28 kB/s
Collecting pyparsing<2.4.0,>=2.0.3
  Downloading pyparsing-2.3.1-py2.py3-none-any.whl (61 kB)
     |████████████████████████████████| 61 kB 33 kB/s
Collecting pyelftools>=0.22
  Downloading pyelftools-0.26-py2.py3-none-any.whl (136 kB)
     |████████████████████████████████| 136 kB 31 kB/s
Collecting ipaddress; python_version < "3"
  Downloading ipaddress-1.0.23-py2.py3-none-any.whl (18 kB)
Collecting six>=1.4.1
  Downloading six-1.14.0-py2.py3-none-any.whl (10 kB)
Collecting cffi!=1.11.3,>=1.8
  Downloading cffi-1.14.0-cp27-cp27mu-manylinux1_x86_64.whl (387 kB)
     |████████████████████████████████| 387 kB 24 kB/s
Collecting enum34; python_version < "3"
  Downloading enum34-1.1.9-py2-none-any.whl (11 kB)
Collecting pycparser
  Downloading pycparser-2.20-py2.py3-none-any.whl (112 kB)
     |████████████████████████████████| 112 kB 23 kB/s
Building wheels for collected packages: future
  Building wheel for future (setup.py) ... done
  Created wheel for future: filename=future-0.18.2-py2-none-any.whl size=502566 sha256=4ff100058c5d0dd3e603f32c45409a1aa790b0518023816bdd10cd5e21601d5e
  Stored in directory: /tmp/pip-ephem-wheel-cache-el9JLC/wheels/5f/11/0c/aad680baf5ef4fbcbab992c9f03e1130357e0c173a4fdabfff
Successfully built future
Installing collected packages: click, pyserial, future, ipaddress, six, pycparser, cffi, enum34, cryptography, pyparsing, pyelftools
Successfully installed cffi-1.14.0 click-7.1.1 cryptography-2.8 enum34-1.1.9 future-0.18.2 ipaddress-1.0.23 pycparser-2.20 pyelftools-0.26 pyparsing-2.3.1 pyserial-3.4 six-1.14.0
All done! You can now run:

  . ./export.sh
```

此时，您刚刚安装的工具尚未添加至 PATH 环境变量，无法通过“命令窗口”使用这些工具。因此，必须设置一些环境变量，这可以通过 ESP-IDF 提供的另一个脚本完成。
在esp-idf目录下运行下面命令，注意下面命令2个小数点中间有一个空格!

```
. ./export.sh
```

注意：每次打开终端进入sdk都要执行一次此命令
成功后,便这样提示:



2.3 编译工程

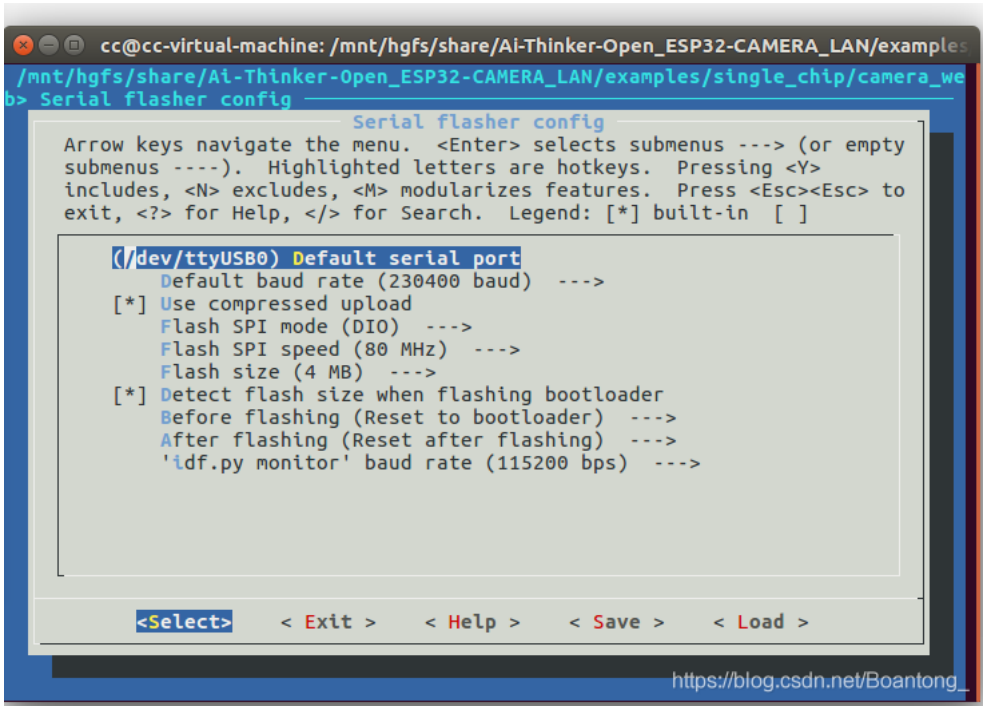
1. 进入示例工程 esp_cam\examples\single_chip\camera_web_server

```
cd esp_cam\examples\single_chip\camera_web_server
```

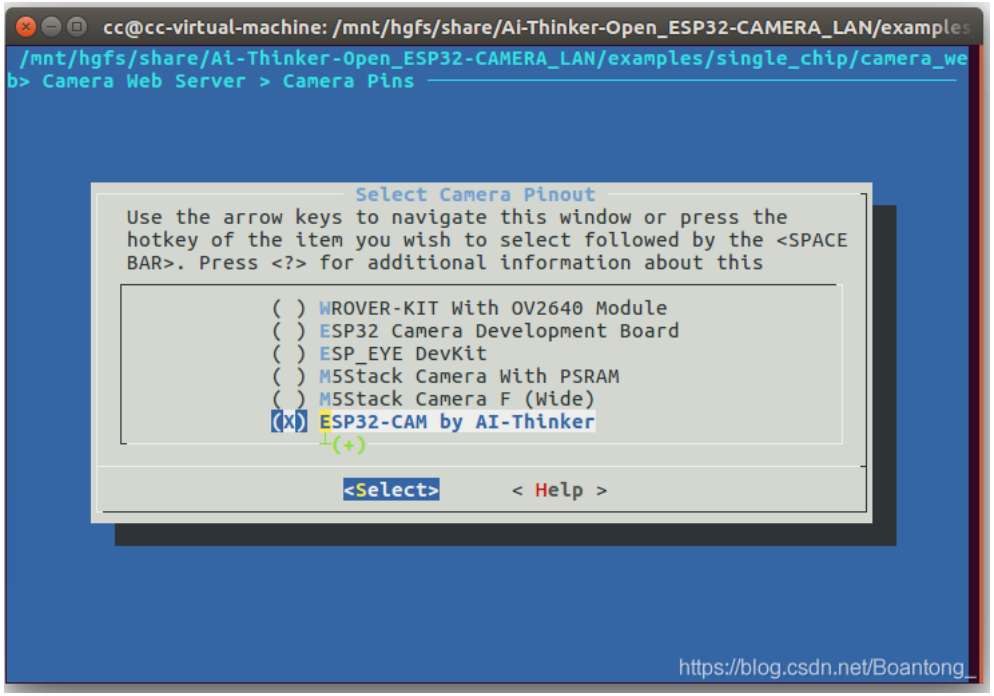
2. 设置menuconfig

```
make menuconfig
```

进入Serial flasher config设置如下

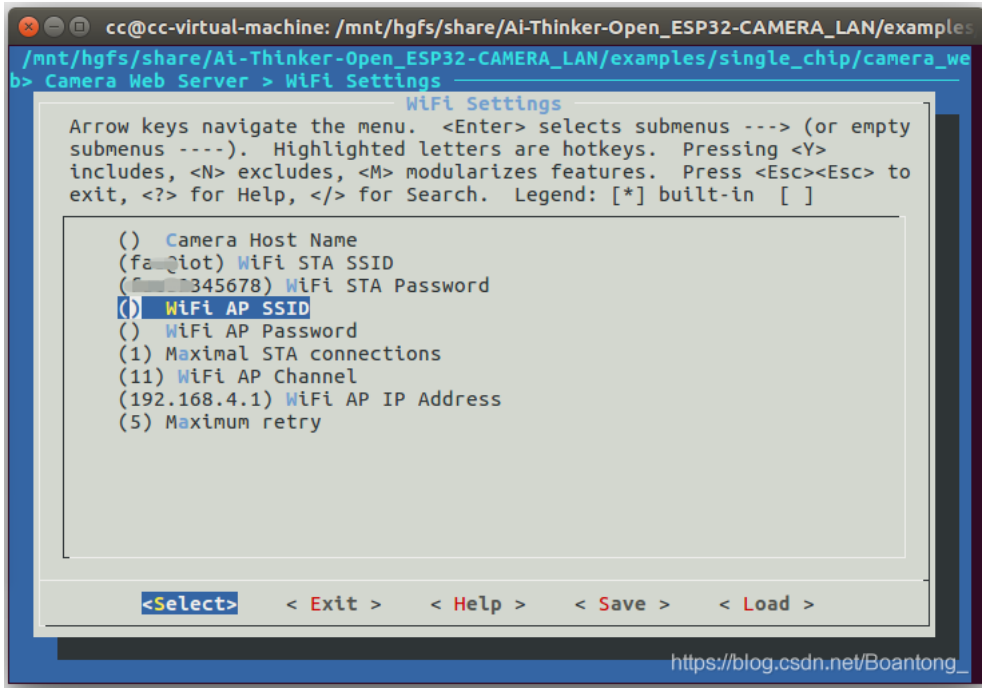


进入Camera Web Server —>Camera Pins —> Select Camera Pinout —> 选择ESP32-CAM by AI-Thinker

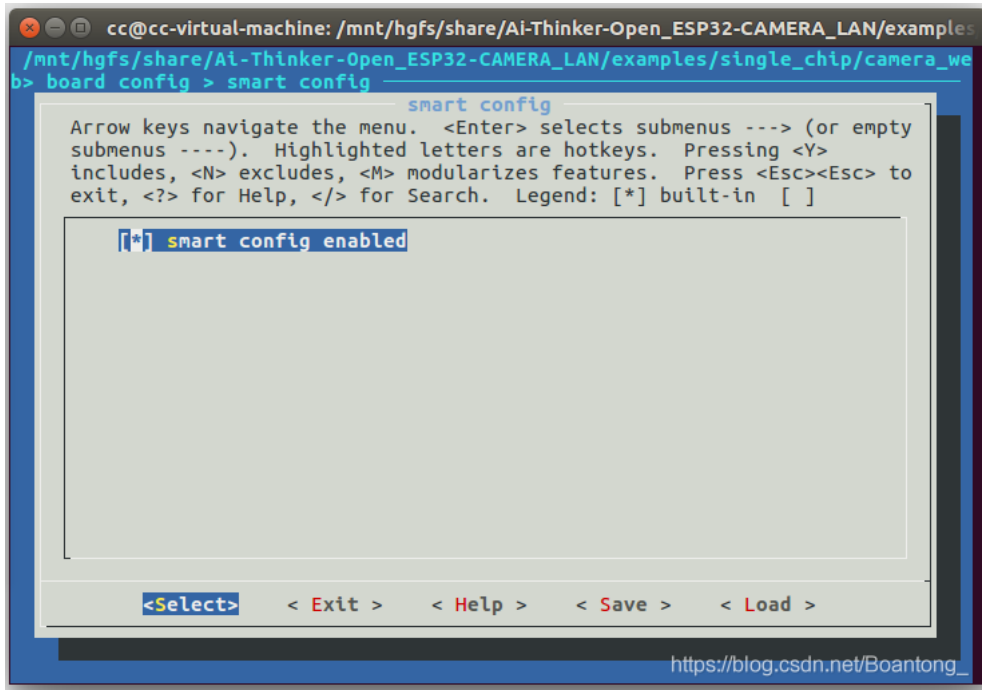


设置WIFI模式（station/ AP模式二选一即可）

1. station模式
   进入camera_web> Camera Web Server > WiFi Settings，设置WIFI名字和密码

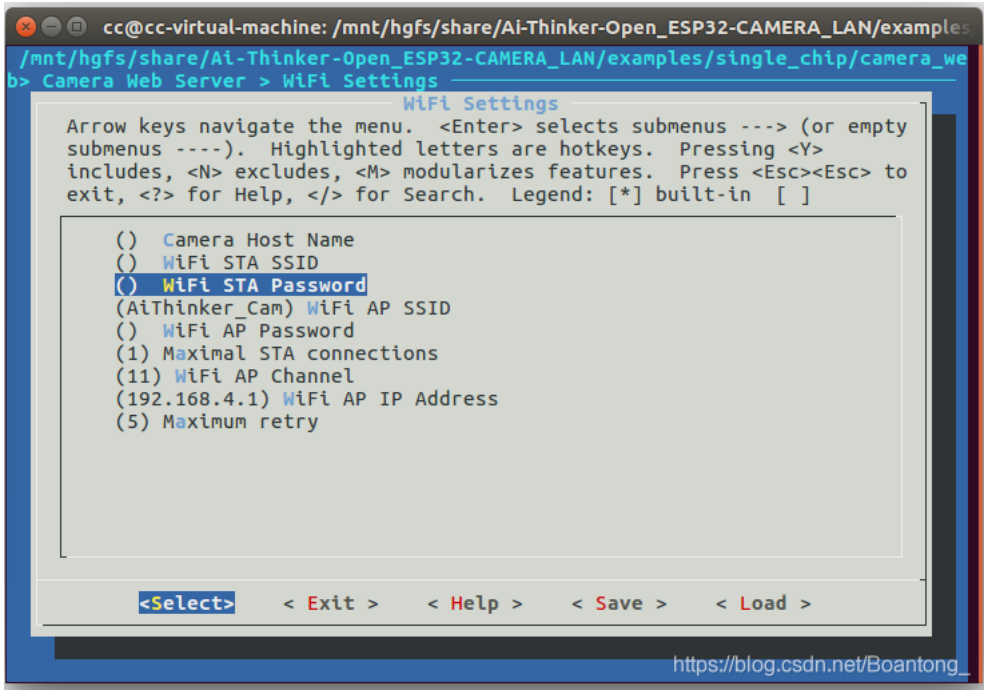


   或者使用Smartconfig 配网， 进入board config > smart config， 输入Y使能

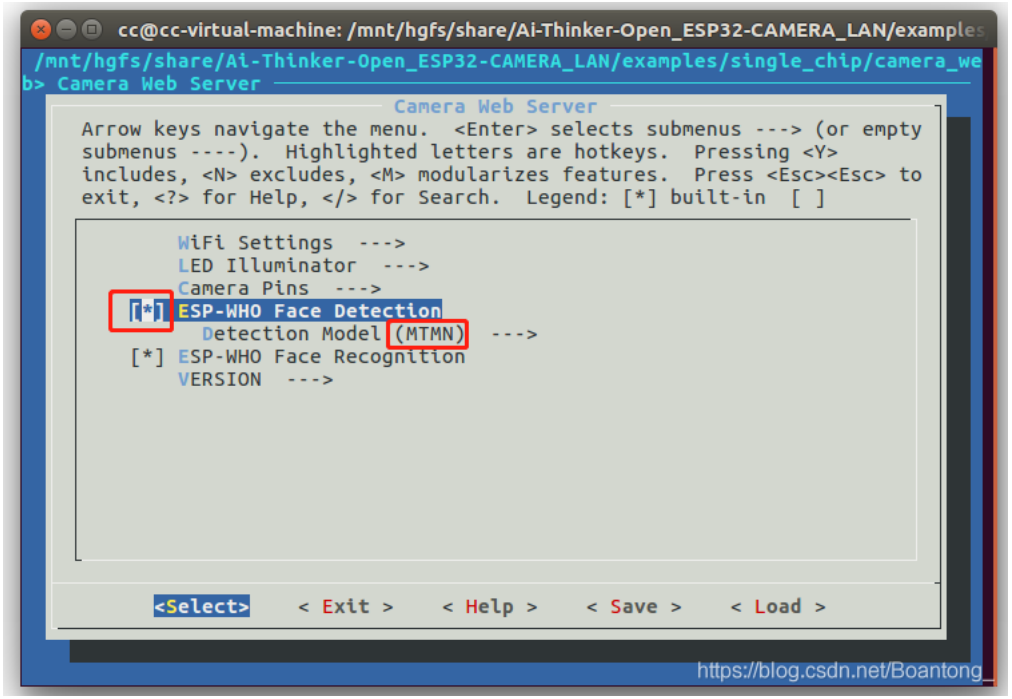


2. AP模式

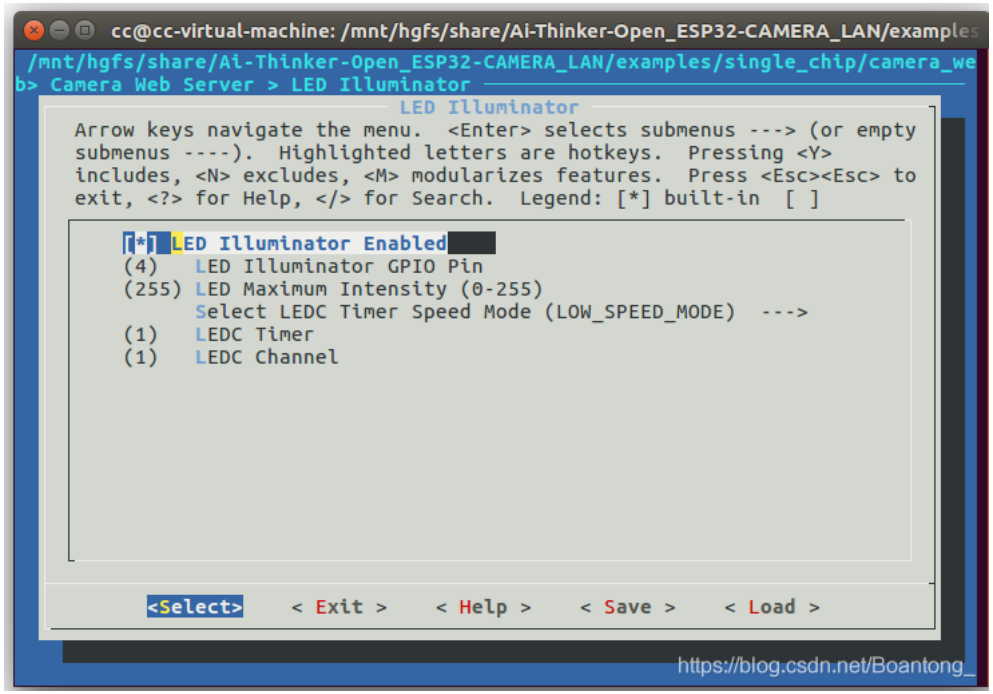
进入Camera Web Server > WiFi Settings，输入AP SSID、AP password、AP Channel、AP IP Address等



如果需要人脸识别功能，进入 Camera Web Server，选择ESP-WHO Face Detection，输入Y ，进入Detection Model,选择MTMN



如果需要LED灯补光，进入 Camera Web Server > LED Illuminator，选择LED Illuminator Enabled 输入Y使能



保存退出，编译下载

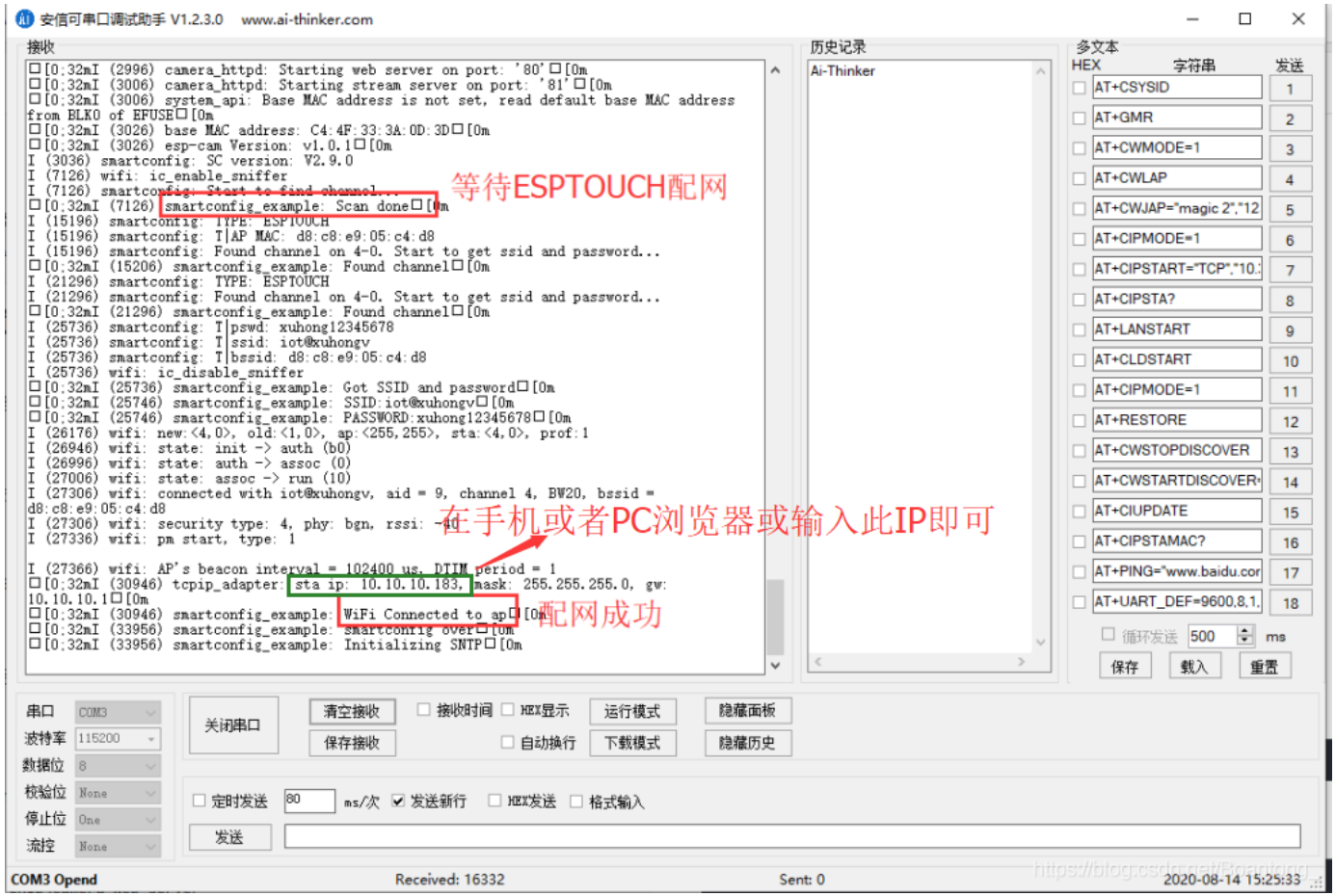
```
make    //编译
make flash  //下载
make monitor  //串口打印
```

三、建立局域网

station模式和AP模式建立局域网的步骤有所不同，根据自己配置的WIFI模式选择

3.1 station 模式

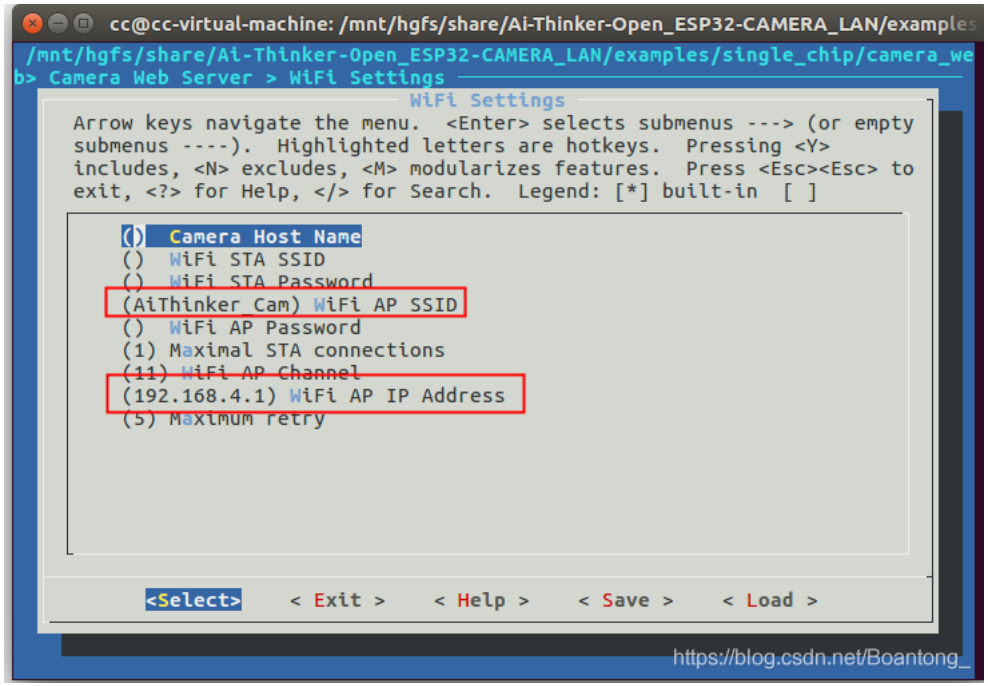
1. 如果没有使能smartconfig，并且写入了WiFi的名字和密码，那么上电之后能自动连接WiFi
2. 如果使能了smartconfig，则可以使用EspTouch进行smartconfig配网 ，EspTouch配网APP点我下载



3. 配好网后，可以在串口打印中看到模组分配到的IP地址，如上图，电脑或手机连接到同一个WiFi，在浏览器中输入模组的IP地址

3.2 AP模式

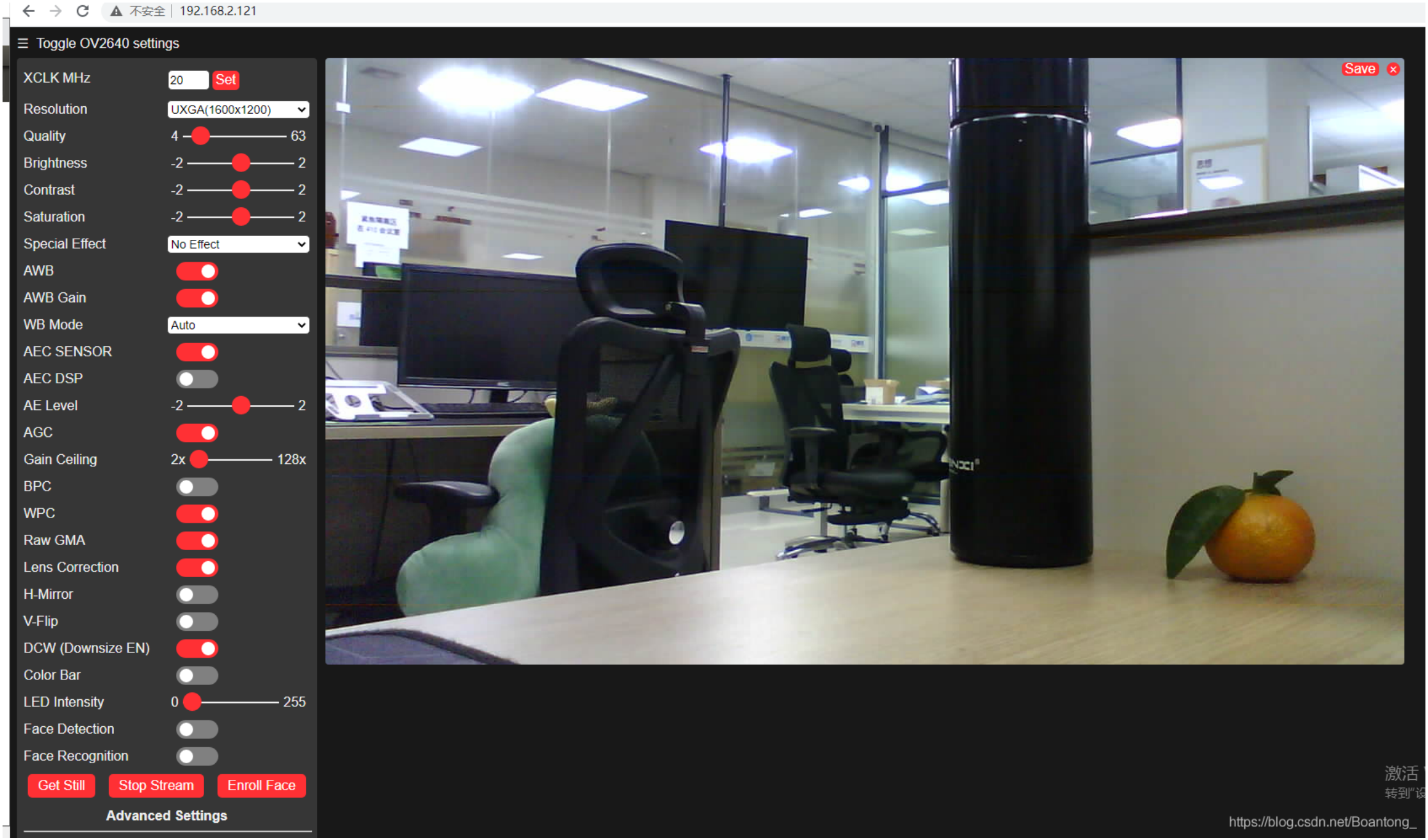
电脑或手机连接模组的热点，在浏览器中输设置好的IP地址即可



四、拍照， 视频流 ，人脸识别

4.1 拍照，视频流

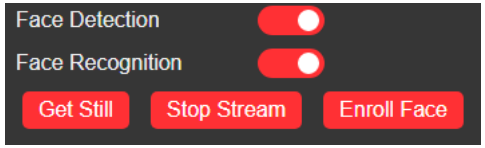
电脑与模组建立局域网后，点击Get Still拍照，点击Start Stream拍视频，效果如下图：



点击右上角save可保存当前照片，点击Resolution设置分辨率，LED Intensity设置LED灯亮度

4.2 人脸检测、人脸识别

人脸检测和人脸识别会占用大量内存，不能使用大分辨率，需把分辨率设置为320x240



打开Face Detection即可实现人脸检测，打开Face Recognition实现人脸识别，点击Enroll Face采集人脸，采集成功显示如下图

当识别出采集过的人脸时，显示如下图