Software Setup for ESP32

N Chakradhar*, P. N. V. S. S. K. HAVISH*, S. .S. Ashish* and G V V Sharma*

***Abstract*—This manual shows how to setup the software tools for ESP32 on Raspbian OS. The process is similar for any Debian style OS.**

1 Firmware

1) Go to http://micropython.org/download, scroll down to ESP32 downloads list and download the latest firmware for ESP32 boards (.bin file).
2) Now install **esptool**

```
sudo pip install esptool
```

2 Connecting ESP32

1) Connect the ESP32 to system and type the following commands to know which port it is connected to.

```
dmesg | grep tty
```

Suppose it is connected to ttyUSB0. Go to the folder where the .bin was saved and execute

```
esptool.py --port /dev/ttyUSB0
    erase_flash
esptool.py --chip esp32 --port
    /dev/ttyUSB0 write_flash -z
    0x1000 filename.bin
```

3 SCREEN

To test the firmware, we can use the screen command to communicate to the board directly. To do it, use these series of commands.

1)

```
sudo apt-get install screen
screen /dev/ttyUSB0 115200
```

Press enter key. You will be presented with a python-like terminal where in you can give python commands directly.

2) To test if it's working or not, just use this command

```
print (”Hello_World”)
```

The terminal output should be similar to this:

```
>>> print (”Hello_World”)
Hello_world
>>>
```

4 AMPY

ampy is used to upload or download codes from the ESP32. To get this tool, execute the following command

```
sudo pip install adafruit-ampy
```

5 Arduino IDE

1) Download the latest Arduino IDE from https://www.arduino.cc/en/Main/Software
2) For making the ESP32 board visible in the Arduino IDE, the following instructions have to be executed. Make sure that you are doing this using a downloaded IDE.

```
sudo apt-get install python-
    serial && \
mkdir -p ~/Arduino/hardware/
    espressif && \
cd ~/Arduino/hardware/espressif
    && \
git clone https://github.com/
    espressif/arduino-esp32.git
    esp32 && \
cd esp32 && \
git submodule update --init --
    recursive && \
```

*The authors are with the Department of Electrical Engineering, Indian Institute of Technology, Hyderabad 502285 India e-mail: gadepall@iith.ac.in. All material in this manual is released under GNU GPL. Free to use for anything.

```
cd tools && \
python get.py && \
wget https://github.com/
   gadepall/resources/raw/
   master/esp32/xtensa-
   toolchain-rpi.zip && \
unzip xtensa-toolchain-rpi.zip
```

6 Xtensa for RPI 3

You can also build the xtensa library through the following commands

```
sudo apt-get install gawk gperf
   grep gettext automake bison flex
    texinfo help2man libtool
   libtool-bin git wget make
   libncurses-dev python python-
   serial python-dev python-pip
sudo pip install pyserial
cd ~/esp
git clone -b xtensa-1.22.x https:
   //github.com/espressif/crosstool
   -NG.git
cd crosstool-NG
./bootstrap && ./configure --
   enable-local && make install
./ct-ng xtensa-esp32-elf
nano ./.config
-- > Find CT_PARALLEL_JOBS=0 and
   change 0 to 1
./ct-ng build
chmod -R u+w builds/xtensa-esp32-
   elf
```

7 Blink Program

1) Connect the ESP 32 to the computer and select the appropriate port.
2) Select Tools- >Board- >ESP 32 DEV MODULE
3) Run the following code. You should see the onboard LED blinking.

```
int ledPin = 2;

void setup()
{
    pinMode(ledPin, OUTPUT);
    Serial.begin(115200);
}

void loop()
{
    digitalWrite(ledPin, HIGH);
    delay(500);
    digitalWrite(ledPin, LOW);
    delay(500);
}
```