ESP32-DevKitC

Getting Started Guide

Version 1.2

About This Guide

This user guide introduces the basic features and operations of the ESP32-DevKitC.

The document is structured as follows:

Chapter	Title	Content
Chapter 1	Overview	Introduction to the ESP32-DevKitC.
Chapter 2	Download Process	Introduction to the boot modes and download operations.

Release Notes

Date	Version	Release notes
2016.09	V1.0	Initial release.
2016.10	V1.1	Updated figure “ESP32-DevKitC Layout” in Chapter 1.
2016.11	V1.2	Updated figure “ESP32-DevKitC Layout” in Chapter 1. Added figure “ESP32-CoreBoard Dimensions” in Chapter 1.

Table of Contents

1. Overview

ESP32-DevKitC is a small-sized ESP32-based development board produced by Espressif. Most of the I/O pins are led out to the pin headers on both sides for easy interfacing. Developers can connect these pins to peripherals as needed. Standard headers also make development easy and convenient when using a breadboard.

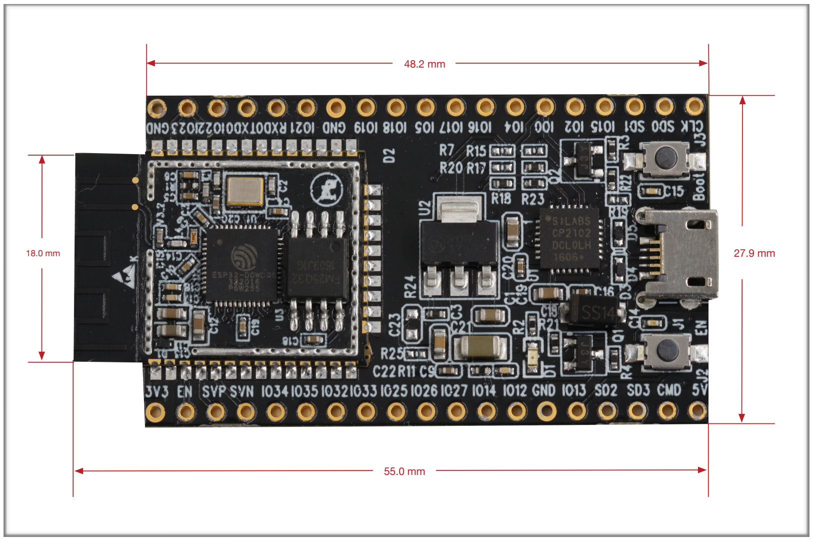


Figure 1-1. ESP32-CoreBoard Dimensions

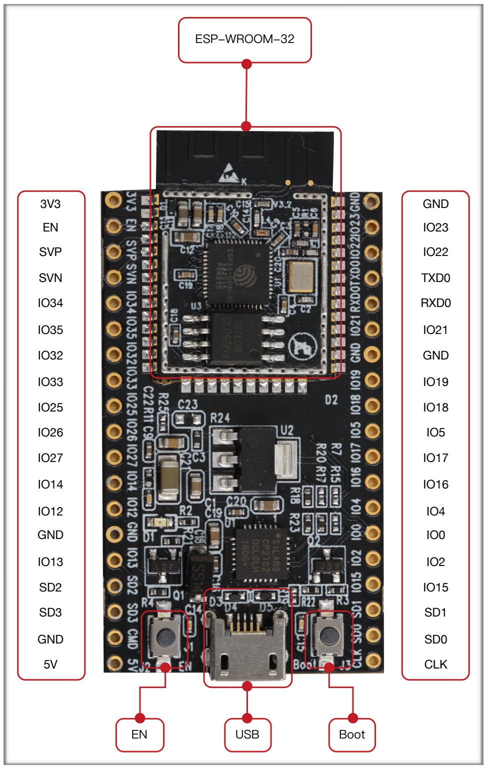


Figure 1-2. ESP32-DevKitC Layout

2. Download Process

You will need the hardware listed below:

- 1 × ESP32-DevKitC
- 1 × PC (with Windows OS as an example in this document)
- 1 × USB cable

2.1. Create Serial Communication

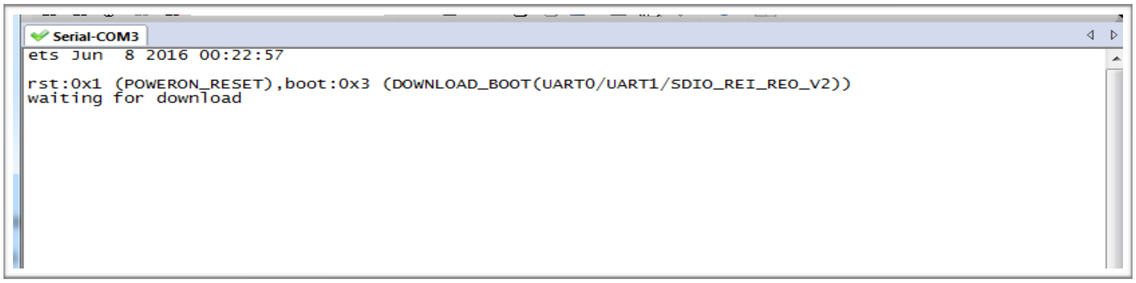
Connect the ESP32-DevKitC to the PC using the USB cable. Check the list of identified external COM ports in the Windows Device Manager and confirm the COM port number of the ESP32-DevKitC.

2.2. Set Download Mode and SPI Boot Mode

As shown in Figure 1-1, the ESP32-DevKitC features two buttons: one marked as "EN", which is the reset button and is located on the bottom left corner, and another one marked "Boot" on the bottom right corner, which is the download button.

- Download mode:

  While pressing the "Boot" button, also press the "EN" button to initiate the download mode. The system should then enter the download mode and output the following on the terminal:



```
ets Jun  8 2016 00:22:57

rst:0x1 (POWERON_RESET),boot:0x3 (DOWNLOAD_BOOT(UART0/UART1/SDIO_REI_REO_V2))
waiting for download
```

Figure 2-1. Download Mode Output

- SPI boot mode:

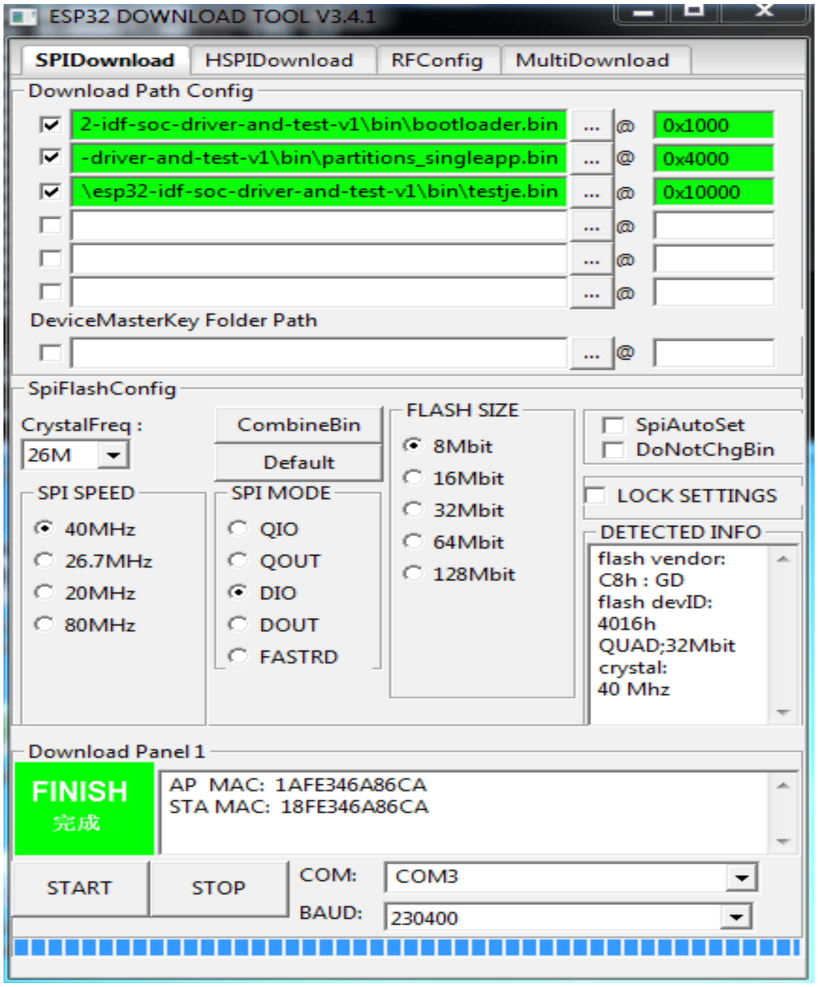
  Simply press the "EN" button and the ESP32-DevKitC will enter the SPI boot mode. If download operation is completed successfully, the system will output some information on the terminal, as shown in Figure 2-3.

2.3. Download Methods

2.3.1. Use ESP32 DOWNLOAD TOOL

Please download the latest *ESP32 DOWNLOAD TOOL* from Espressif website.

Open the ESP32 DOWNLOAD TOOL. Select *bootloader.bin*, *paritions_singeapp.bin*, and *testje.bin* to be downloaded to addresses 0x1000, 0x4000, and 0x10000, respectively. Press “*START*” and wait for the prompt that indicates the download result.



ESP32 DOWNLOAD TOOL V3.4.1

SPIDownload | HSPIDownload | RFConfig | MultiDownload

Download Path Config

☑ 2-idf-soc-driver-and-test-v1\bin\bootloader.bin ... @ 0x1000
☑ -driver-and-test-v1\bin\partitions_singleapp.bin ... @ 0x4000
☑ \esp32-idf-soc-driver-and-test-v1\bin\testje.bin ... @ 0x10000
□ ... @
□ ... @
□ ... @

DeviceMasterKey Folder Path

□ ... @

SpiFlashConfig

CrystalFreq : 26M

CombineBin | Default

SPI SPEED: ● 40MHz ○ 26.7MHz ○ 20MHz ○ 80MHz

SPI MODE: ○ QIO ○ QOUT ● DIO ○ DOUT ○ FASTRD

FLASH SIZE: ● 8Mbit ○ 16Mbit ○ 32Mbit ○ 64Mbit ○ 128Mbit

□ SpiAutoSet
□ DoNotChgBin

□ LOCK SETTINGS

DETECTED INFO

```
flash vendor:
C8h : GD
flash devID:
4016h
QUAD;32Mbit
crystal:
40 Mhz
```

Download Panel 1

FINISH 完成

```
AP  MAC: 1AFE346A86CA
STA MAC: 18FE346A86CA
```

START | STOP

COM: COM3

BAUD: 230400

Figure 2-2. ESP32 DOWNLOAD TOOL Interface

2.3.2. Use Python Scripts

Get Python scripts from *esp-idf/components/esptool_py/esptool/esptool.py* in the ESP-IDF.

The download process will require the system to have a command terminal. For the Windows System, open the “Windows PowerShell” or “Windows Command Terminal”.

On the terminal, set the current directory to where the downloaded Python scripts are saved, and then type in the following commands:

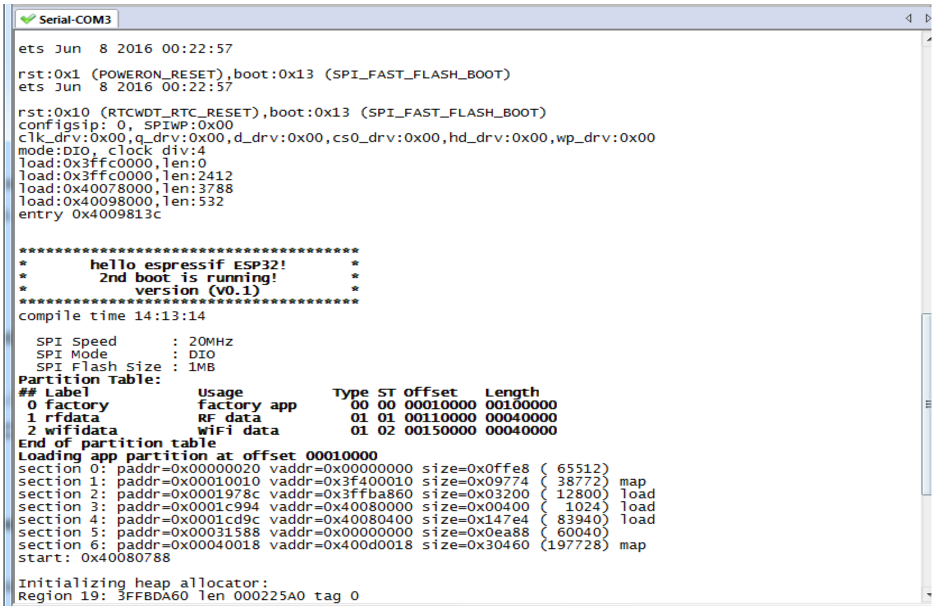
```
python esptool.py -b 115200 -p COM3 write_flash -ff 40m -fm qio   -ih
0x0 -il 0x00  0x1000 bootloader.bin 0x4000 partitions_singleapp.bin
0x10000 testje.bin
```

Notes:

- *The parameters highlighted in blue represent the baud rate, serial port, flash frequency, and the flash mode, respectively. You can change these parameters as necessary.*
- *Only* **DIO** *mode is currently supported. Further updates of download tools and scripts will allow support for both the* **QIO** *and* **DIO** *modes.*
- *To retrieve "help information", input* `python esptool.py -h`*. For information on download commands, input* `python esptool.py write_flash -h`*.*

2.4. Check the Serial Output

Launch the serial terminal on the PC and press the "EN" button on the development board. The system is now expected to enter the SPI boot mode and output the following:



```
ets Jun  8 2016 00:22:57

rst:0x1 (POWERON_RESET),boot:0x13 (SPI_FAST_FLASH_BOOT)
ets Jun  8 2016 00:22:57

rst:0x10 (RTCWDT_RTC_RESET),boot:0x13 (SPI_FAST_FLASH_BOOT)
configsip: 0, SPIWP:0x00
clk_drv:0x00,q_drv:0x00,d_drv:0x00,cs0_drv:0x00,hd_drv:0x00,wp_drv:0x00
mode:DIO, clock div:4
load:0x3ffc0000,len:0
load:0x3ffc0000,len:2412
load:0x40078000,len:3788
load:0x40098000,len:532
entry 0x4009813c


**************************************
*       hello espressif ESP32!       *
*        2nd boot is running!        *
*            version (V0.1)          *
**************************************
compile time 14:13:14

  SPI Speed      : 20MHz
  SPI Mode       : DIO
  SPI Flash Size : 1MB
Partition Table:
## Label            Usage          Type ST Offset   Length
 0 factory          factory app      00 00 00010000 00100000
 1 rfdata           RF data          01 01 00110000 00040000
 2 wifidata         WiFi data        01 02 00150000 00040000
End of partition table
Loading app partition at offset 00010000
section 0: paddr=0x00000020 vaddr=0x00000000 size=0x0ffe8 ( 65512)
section 1: paddr=0x00010010 vaddr=0x3f400010 size=0x09774 ( 38772) map
section 2: paddr=0x0001978c vaddr=0x3ffba860 size=0x03200 ( 12800) load
section 3: paddr=0x0001c994 vaddr=0x40080000 size=0x00400 (  1024) load
section 4: paddr=0x0001cd9c vaddr=0x40080400 size=0x147e4 ( 83940) load
section 5: paddr=0x00031588 vaddr=0x00000000 size=0x0ea88 ( 60040)
section 6: paddr=0x00040018 vaddr=0x400d0018 size=0x30460 (197728) map
start: 0x40080788

Initializing heap allocator:
Region 19: 3FFBDA60 len 000225A0 tag 0
```

Figure 2-3. SPI Boot Mode Output

Espressif IOT Team

www.espressif.com

Disclaimer and Copyright Notice

Information in this document, including URL references, is subject to change without notice.

THIS DOCUMENT IS PROVIDED AS IS WITH NO WARRANTIES WHATSOEVER, INCLUDING ANY WARRANTY OF MERCHANTABILITY, NON-INFRINGEMENT, FITNESS FOR ANY PARTICULAR PURPOSE, OR ANY WARRANTY OTHERWISE ARISING OUT OF ANY PROPOSAL, SPECIFICATION OR SAMPLE.

All liability, including liability for infringement of any proprietary rights, relating to the use of information in this document, is disclaimed. No licenses expressed or implied, by estoppel or otherwise, to any intellectual property rights are granted herein.

The Wi-Fi Alliance Member logo is a trademark of the Wi-Fi Alliance. The Bluetooth logo is a registered trademark of Bluetooth SIG.