ESP32 Certification and Test Guide

Version 1.5

About This Document

This document provides instructions on how to test ESP32 to meet the certification requirements. The document is structured as follows:

Chapter	Title	Subject
Chapter 1	Overview	Introduction to test procedures.
Chapter 2	Test Preparation	Presentation of test preparations.
Chapter 3	Connecting Device	Instruction on how to connect devices.
Chapter 4	Adaptivity Test	Presentation of adaptivity testing procedures.
Chapter 5	Wi-Fi/BT Operation Commands	Explanation of Wi-Fi/BT operation commands.

Release Notes

Date	Version	Release notes
2016.12	V1.0	Initial release.
2017.03	V1.1	Updated the BIN files used for the tests; Added Notice in Chapter 1; Updated Chapter 5.
2017.06	V1.2	Added download addresses in Chapter 5.
2017.06	V1.3	Added description about the adaptivity test in AP mode in Chapter 4.
2017.08	V1.4	Corrected frequency of channel zero from 2042 MHz to 2402 MHz in Chapter 5; Updated description about the serial port configuration tools in Chapter 3.
2017.11	V1.5	Corrected a typo in Section 5.2.1: • change *frequency modulation* to *frequency hopping*.

Documentation Change Notification

Espressif provides email notifications to keep customers updated on changes to technical documentation. Please subscribe *here*.

Certifications

Please download the product certification(s) *here*.

Table of Contents

1. Overview

This document provides guidance on how to test ESP32 to meet certification requirements. Table 1-1 shows two binary files for the tests.

Table 1-1. Test Procedure Binary Files

BIN	Test Item
ESP32_Adaptivity_20170110.bin	Adaptivity (baud rate 115200)
ESP32_FCC_WIFI_BT_20170209.bin	Wi-Fi/BT performance (baud rate 115200)

⚠ ***Notice:***

- *The BIN files in this document are used only as examples. Please download the BIN files from http://www.espressif.com/en/support/download/other-tools.*
- *Please distinguish the two testing procedures. Two testing boards with the relevant binary file downloaded are required. Customers can carry out two tests in separate labs simultaneously.*

2. Test Preparation

2.1. Hardware Connection and Configuration

Please follow the instructions below:

Table 2-1. Pin Configuration Instructions

Pin	Configuration Instructions
3V3/CH_EN	Connects to 3.3V power.
RXD/TXD/GND	Lead the three pins out to serial lines for the communication between the PC and ESP32 and for the control of ESP32.
GPIO0 GPIO2	These two pins can switch the boot mode of the ESP32.

2.2. Hardware Boot Mode

2.2.1. Download Mode

When GPIO0=0, GPIO2=0, ESP32 is in the Download mode and the firmware can be downloaded to the external flash.

2.2.2. Flash Mode

When GPIO0=1, ESP32 is in the Flash mode. ESP32 will automatically read and run programs from flash during power on.

> ⚠ ***Notice:***
>
> *Download mode is for downloading the firmware. Flash mode is the normal working mode.*

2.3. Antenna Impedance Matching Requirement

For the EMC test, the π impedance matching circuit of the external antenna should meet the following requirements.

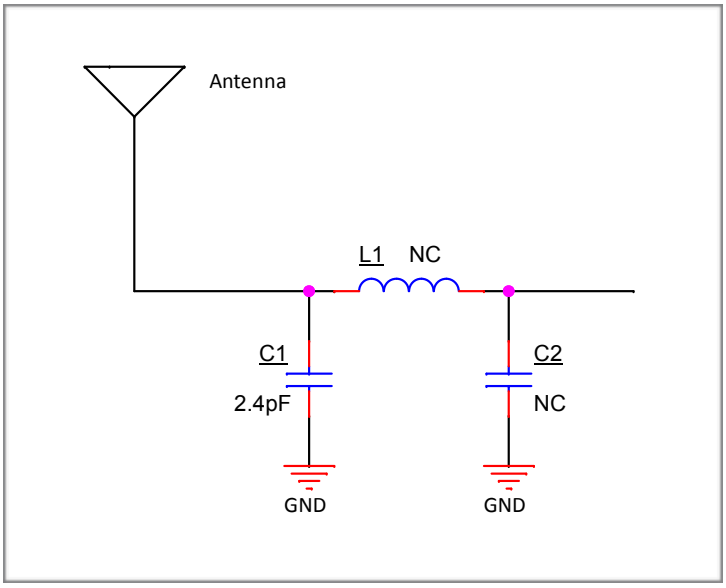

Figure 2-1. π-type Impedance Matching Circuit of External Antenna

> 📖 ***Note:***
>
> *C1 must be a 2.4-pF capacitor. L1, C2, along with C1 perform a 50Ω impedance matching for the Antenna. The value depends on the antenna impedance.*

3. Connecting Device

3.1. Serial Port Configuration Tool

3.1.1. Tool Introduction

Serial port configuration tools can be used for configuring the serial COM ports between ESP32 and PC.

> 📖 ***Note:***
>
> *Customers need to pre-download and install the serial port configuration tools, such as SecureCRT.*

3.1.2. Procedure

Please follow the steps below.

1. Run the serial port configuration tool and go to the main interface;
2. Open the window for connecting ESP32 and the PC, and configure all the necessary parameters. Below you can find four of them, as an example:

Protocol	Configure the protocol to “Serial”.
Port	Select a port, such as “COM6”.
Baud Rate	Set the baud rate to “115200”.
Flow Control	Deselect “RTS/CTS”.

3. Start the connection after setting these parameters, and make sure the devices are connected successfully.

3.2. Download Tool

3.2.1. Tool Introduction

> 📖 ***Note:***
>
> *The ESP Flash Download Tool (hereinafter termed as Flash Download Tool) is used in this chapter. Please download the tool from: http://www.espressif.com/en/support/download/other-tools.*

3.2.2. Procedure

Please follow the steps below:

1. Run the Flash Download Tool.
   Make sure ESP32 works in the Download mode and that the port number of the serial port is not used by other applications.

2. Select the BIN files to be downloaded in the “SPIDownload” menu box.
   If the path is valid, it appears against a green background; if the path is invalid, then it appears against a red background.
3. Enter the starting address in the OFFSET text box, such as “0X000” or “0x1000”.
   The download addresses of the two binary files start from “0X000” or “0x1000”.
4. Set the configuration options.
5. Click the “Start” button to download the files.
   If the configuration is incorrect, the Flash Download Tool will show this in the status bar under “Download Panel 1”, and a specific type of error in the command prompt window beside the status bar.
6. After the downloading is complete, the status bar displays “Finish”, as Figure 3-5 shows.

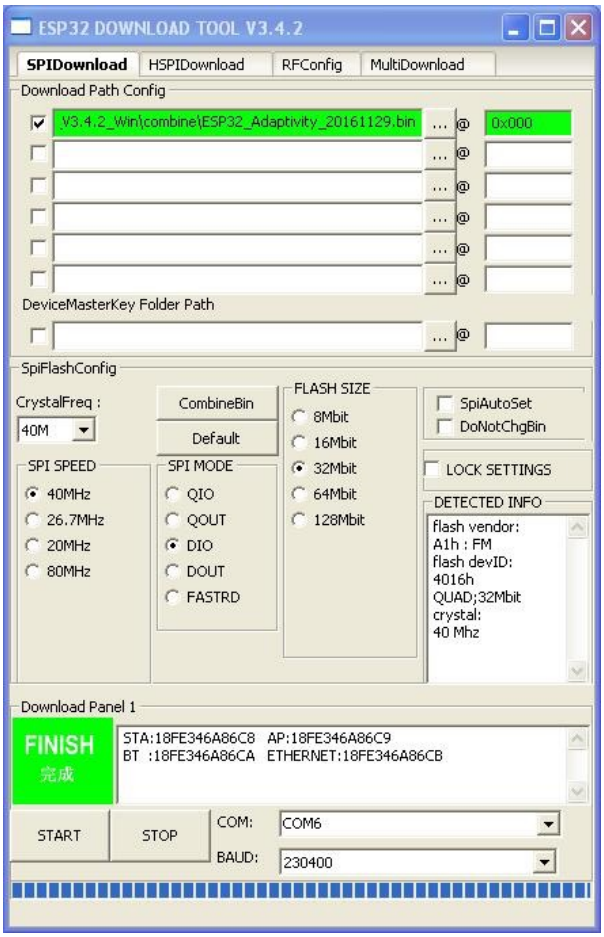

Figure 3-5. Flash Download Tool

3.2.3. Configuration Description

Table 3-1 describes the configuration options.

Table 3-1. Configuration Description

Options	Description	Configuration Instructions
CrystalFreq	The type of external crystals	Select the CrystalFreq according to the crystal type that ESP32 is using. Select the option “40M”.
CombineBin	Combine the binary files	Combine several binary files into one.
SPI Speed	SPI speed	Select the SPI speed of ESP32. The options are: 40 MHz, 26.7 MHz, 20 MHz and 80 MHz.
SPI Mode	SPI mode	Select the SPI connecting mode of ESP32. The options are: QIO, QOUT, DIO and DOUT.
Flash Size	The size of the flash	Select the flash size that ESP32 uses.
COM	The port number of the device	Select the port number of ESP32.
Baudrate	Baud rate	Select the downloading speed of the binary file. The default option is 115200.
MAC Address	MAC address	After the downloading is complete, the system will show the MAC address where the binary files are saved.

4. Adaptivity Test

4.1. Test Tools

- ESP32 module + backplane
- Binaries to download (*ESP32_Adaptivity_20170110.bin*)
- 1 × UART
- 1 × PC with a Windows operating system (and serial port tools installed)
- 1 × network packet-sending tool
- 1 × AP

4.2. Test Procedure

4.2.1. Downloading the Binaries

Table 4-1 shows the download addresses.

Table 4-1. Download Addresses for the Binaries

Binaries	Download Address
ESP32_Adaptivity_20170110.bin	0x000

> ⚠ ***Notice:***
> *Select DIO for flash_mode.*

4.2.2. Station Mode

1. Connecting ESP32 to AP

Open the serial port tool (and set the baud rate to 115200). Input ssc command and connect ESP32 to the specified AP. The connection steps and the corresponding ssc commands are shown below:

Table 4-2. AP Connection Steps

Steps	ssc Command
1. Set ESP32 to Station mode.	`op -S -o 1`
2. Connect ESP32 to the specified AP.	`sta -C -s <AP SSID> -p <AP password>`

> ⚠ ***Notice:***
>
> *If the AP encryption mode is open, customers only need to input* `sta -C -s <AP SSID>`.

2. Ping Packets

   - Install a network debugging tool on the PC and connect the PC to the same AP to which the ESP32 connects.
   - Run the network debugging tool to set up the UDP server.
   - Input the local IP address and port number on the network debugging tool to complete the configuration. Then, click the “Connect” button.
   - In the serial tool, input the ssc command to set up socket connection and ping packets.

- Configure parameter:

  ```
  gpiotmp -C -e 1 -c <value>
  ```

  `value` should be a number between 40 ~ 64; the default value is 50.

  Example:

  ```
  gpiotmp -C -e 1 -c 50
  ```

- Configure UDP packets:

  ```
  soc -B -t UDP -p <server port>
  ```

  ```
  soc -S -s <socket_id> -l <pkt len> -i <server IP _address> -p <server port> -n <pkt num> -j <send pkt delay>
  ```

  Example:

  ```
  soc -B -t UDP -p 10004
  ```

  ```
  soc -S -s 0 -l 1000 -i 255.255.255.255 -p 10004 -n 2000 -j 10
  ```

4.2.3. AP Mode

1. Configuring ESP32 to SoftAP Mode

Open the serial port tool (and set the baud rate to 115200). Input ssc command and configure ESP32 to SoftAP mode. The connection steps and the corresponding ssc commands are shown below:

Table 4-3. SoftAP Configuration Steps

Steps	ssc Command	Remark
1. Set ESP32 to SoftAP mode.	`op -S -o 2`	-
2. Set up an AP.	`ap -S -s <AP_ssid> -p <AP_password> -n <AP_channel>`	Example: `ap -S -s esp_ap1 -p 12345678 -n 1` Note: The value of `AP_channel` can be set to 1 to 13.
3. Set the B/G/N mode.	`phy -S -o 1 -m b/g/n`	Note: `phy -Q -o 1` command can be used to confirm B/G/N mode.
4. Request the IP address.	`ip`	-

2. Ping Packets

- Install a network debugging tool on the PC and connect the PC to the AP that has been set up in Step 2.
- Run the network debugging tool to set up the UDP server.
- Input the local IP address and port number on the network debugging tool to complete the configuration. Then, click the “Connect” button.
- In the serial tool, input the ssc command to set up socket connection and ping packets.

- Configure parameter:

  ```
  gpiotmp -C -e 1 -c <value>
  ```

  `value` should be a number between 40 ~ 64; the default value is 50.

  Example:

  ```
  gpiotmp -C -e 1 -c 50
  ```

- Configure UDP packets:

  ```
  soc -B -t UDP -p <server port>
  ```

  ```
  soc -S -s <socket_id> -l <pkt len> -i <server IP _address> -p <server port> -n <pkt num> -j <send pkt delay>
  ```

  Example:

  ```
  soc -B -t UDP -p 10004
  ```

  ```
  soc -S -s 0 -l 1000 -i 255.255.255.255 -p 10004 -n 2000 -j 10
  ```

5. Wi-Fi/BT Operation Command

To test the EMC/RF performance, please download *ESP32_FCC_WIFI_BT_20170209.bin*. Table 5-1 shows the download addresses.

Table 5-1. Download Addresses for the Binaries

Binaries	Download Address
ESP32_FCC_WIFI_BT_20170209.bin	0x1000

5.1. Wi-Fi Test Commands

> 📖 ***Note:***
>
> *The baud rate is 115200.*

5.1.1. Selecting Test Mode

- Command for the RF certification mode: `fcc_mode_sel  2`

  Print: `fcc_mode_sel=2, RF test start!!!`
- Command for the EMC certification mode: `fcc_mode_sel  1`

  Print: `fcc_mode_sel=1, EMC Certification start!!!`
- Command for the FCC certification mode: `fcc_mode_sel  0`

  Print: `fcc_mode_sel=0, FCC Certification start!!!`

5.1.2. Tx Start Command

`wifitxout  <Parameter1>  <Parameter2>  <Parameter3>`

`<Parameter1>`: Select the Tx channel from 1 ~ 14.

`<Parameter2>`: Select the Tx data rate according to Table 5-2.

`<Parameter3>`: Tx power attenuation. It is an 8-bit signed operand, in multiples of 0.25 dB. For example, 4 indicates an attenuation of 1 dB.

Table 5-2. Parameter2 and Tx Data Rate

11b		11g		11n	
Parameter	Data rate	Parameter	Data rate	Parameter	Data rate
0x0	1 Mbps	0xb	6 Mbps	0x10	6.5 Mbps / MCS0
0x1	2 Mbps	0xf	9 Mbps	0x11	13 Mbps / MCS1
0x2	5.5 Mbps	0xa	12 Mbps	0x12	19.5 Mbps / MCS2
0x3	11 Mbps	0xe	18 Mbps	0x13	26 Mbps / MCS3
-	-	0x9	24 Mbps	0x14	39 Mbps / MCS4
-	-	0xd	36 Mbps	0x15	52 Mbps / MCS5
-	-	0x8	48 Mbps	0x16	58.5 Mbps / MCS6
-	-	0xc	54 Mbps	0x17	65 Mbps / MCS7

> ⚠ ***Notice:***
> *After sending data packets, please input the Stop command to put an end to the process.*

Example:

```
wifitxout  1  0x0  0
```

Print: `Wifi tx out: channel=1, rate=0x0, BK=0`

Send the packets through channel 1 (2412 MHz) with a data rate of 1 Mbps.

5.1.3. Tx Stop Command

```
cmdstop
```

Print: `Tx Over`

> ⚠ ***Notice:***
> *After sending data packets, please input the Stop command to change the channel or data rate.*

5.1.4. Command for Selecting Tx 11n Mode 20 Mbps or 40 Mbps

```
tx_cbw40m_en  <Parameter1>
```

- Command for selecting the 11n HT20 mode:

  ```
  tx_cbw40m_en  0
  ```

  Print: `tx_cbw40m_en: 0`

- Command for selecting the 11n HT40 mode:

  ```
  tx_cbw40m_en  1
  ```

  Print: `tx_cbw40m_en: 1`

> 📖 ***Note:***
>
> *The 20 Mbps and 40 Mbps modes share the same command.*

5.1.5. Rx Start Command

`esp_rx <Parameter1> <Parameter2>`

`<Parameter1>`: Select the Rx channel from 1 ~ 14.

`<Parameter2>`: Select the Rx data rate according to Table 5-2.

> ⚠ ***Notice:***
>
> *After receiving data packets, please input Stop command to put an end to the process.*

Example:

`esp_rx 1 0x0`

Print: `wifi rx start: channel is 1, rate is 0x0`

Receive the packets in channel 1 (2412 MHz) with a data rate of 1 Mbps.

5.1.6. Rx Stop Command

`cmdstop`

Print: `Correct: 0 Desired: 0 RSSI: 0`

`Correct` indicates the number of received packets.

`Desired` indicates the number of received packets with the corresponding data rate of `<Parameter2>`.

`RSSI` indicates the average power of the `Desired` packets received.

5.1.7. SCW Tx Command

`wifiscwout <Parameter1> <Parameter2> <Parameter3>`

`<Parameter1>`: SCW Tx enable signal, 1 = send; 0 = stop.

`<Parameter2>`: Select the SCW Tx channel from 1 ~ 14.

`<Parameter3>`: SCW power attenuation. The unit is 0.25 dB. For example, 4 indicates an attenuation of 1 dB and 8 indicates an attenuation of 2 dB.

- SCW Tx example:

  `wifiscwout 1 14 0`

  Print: `wifi single carrier tx out`

  Transmit SCW in channel 14 (2484 MHz)

- SCW Tx stop example:

  `wifiscwout 0 14 0`

  Print: `wifi single carrier tx stop`

Stop SCW transmission.

5.2. BT Test Commands

> 📖 ***Note:***
>
> *Baud rate is 115200.*

5.2.1. BR/EDR Tx Command

`fcc_bt_tx <Parameter1> <Parameter2> <Parameter3> <Parameter4> <Parameter5> <Parameter6> <Parameter7>`

`<Parameter1>`: Tx power attenuation. The range is 0 ~ 9. The unit is 2 dB. Normally the value is 4.

`<Parameter2>`: Enable or disable frequency hopping. 1: enable; 0: disable.

`<Parameter3>`: Select the Tx channel from 0 ~ 78.

`<Parameter4>`: Select the modulation mode. 1: 1M; 2: 2M; 3: 3M.

`<Parameter5>`: Select the DH type. 1: DH1; 3: DH3; 5: DH5.

`<Parameter6>`: Select the Data type. 0: 1010; 1: 00001111; 2: prbs9.

`<Parameter7>`: Reserved: 0 (optional in the command).

Example:

```
fcc_bt_tx  4  0  0  1  3  1  0
```

Print:

```
fcc_bt_tx: txpwr=4, hoppe=0, chan=0, rate=1, DH_type=3, data_type=1
RW TX TEST
```

The command indicates that the Tx power attenuation level is 4; frequency hopping is disabled; the channel is 0 (2402 MHz); the data rate is BR1M; the DH type is DH3 and the data type is 00001111.

5.2.2. LE Tx Command

`fcc_le_tx <Parameter1> <Parameter2> <Parameter3> <Parameter4> <Parameter5>`

`<Parameter1>`: Tx power attenuation. The range is 0 ~ 9. The unit is 2 dB. Normally the value is 4.

`<Parameter2>`: Select the Tx channel from 0 ~ 39.

`<Parameter3>`: Select the payload length. The range is 0 ~ 255. The unit is byte. Normally the value is 250.

`<Parameter4>`: Select the Data type. 0: 1010; 1: 00001111; 2: prbs9.

`<Parameter5>`: Reserved: 0 (optional in the command).

Example:

```
fcc_le_tx  4  0  250  2  0
```

Print:

```
fcc_le_tx: txpwr=4, chan=0, length=250, data_type=2
RW LE TX NHP
```

The command indicates that the Tx power attenuation level is 4; the channel is 0 (2402 MHz); the data rate is LE1M and the data type is prbs9.

5.2.3. Tx Stop Command

```
cmdstop
```

5.2.4. BR/EDR Rx Start Command

```
rw_rx_per <Parameter1>  <Parameter2>  <Parameter3>  <Parameter4>  <Parameter5>
```

<Parameter1>: 0: BR; 1: EDR.

<Parameter2>: Select the Rx channel from 0 ~ 78. 0 to 39 represent even-numbered channels, and 40 to 78 represent odd-numbered channels. For example, if Parameter2 is 0, channel 0 is selected; if Parameter2 is 1, channel 2 is selected; if Parameter2 is 2, channel 4 is selected, and so on. So if Parameter2 is 39, channel 78 is selected. In contrast, if Parameter2 is 40, channel 1 is selected; if Parameter2 is 41, channel 3 is selected; if Parameter2 is 42, channel 5 is selected, and so on. So if Parameter2 is 78, channel 77 is selected.

<Parameter3>: 32-bit Bluetooth address, including UAP (8-bit) and LAP (24-bit). The value is determined by the testing equipment.

<Parameter4>: The logical transport address specified by the protocol. The value is determined by the testing equipment with a range of 0 ~ 7.

<Parameter5>: Reserved: 0 (optional in the command).

Example:

```
rw_rx_per  1  40  0x6BC6967e  0  0
```

Print:

```
rw_rx_per:type=1, chan=40, ulap=0x6BC6967e, ltaddr=0
RW RX PER
```

The command indicates that the Rx data package is EDR (The DH type, Data type are selected by the equipment and can be modulated. DH1 is recommended.), the channel is 1 (2403 MHz), the data rate is 2M or 3M, the bluetooth address is 0x6BC6967e and the logical transport address is 0.

5.2.5. LE Rx Start Command

```
rw_le_rx_per <Parameter1>  <Parameter2>  <Parameter3>
```

<Parameter1>: Select the Rx channel number from 0 ~ 39. Channel 0 ,1, 2 ~ 10 correspond to the frequency of 2404 MHz, 2406 MHz, 2408 MHz ~ 2424 MHz, respectively. Channel 11, 12, 13 ~ 16 correspond to the frequency of 2428 MHz, 2430 MHz, 2432 MHz ~ 2478 MHz, respectively. Channel 37 corresponds to the frequency of 2402 MHz. Channel 38 corresponds to the frequency of 2426 MHz. Channel 39 corresponds to the frequency of 2480 MHz.

<Parameter2>: The data package identify (provided by the data generator or the equipment supplier).

<Parameter3>: Reserved: 0 (optional in the command).

Example:

```
rw_le_rx_per  11  0x71764129  0
```

Print:

```
rw_le_per_syncw: chan=11, syncw=0x71764129
RW RX PER
```

The command indicates that the Rx data package is LE (The Data type is usually prbs9.); the channel is 11 (2428 MHz); the data rate is 1M and the data identify is 0x71764129.

5.2.6. Rx Stop Command

```
cmdstop
```

Input `cmdstop` command to stop receiving packets. The serial port will print the number of packets that have been received.

The format of the number of received packets is
`0 0 0 0 0 0 0 0 w 0 0 0 0 0 0 0 0 p 0 0 0 b 0 0`

The last parameter (based on hexadecimal system) indicates the number of bit errors.

The second-to-last parameter (based on hexadecimal system) indicates the overall number of the received bits at a certain data rate.

5.2.7. SCW Tx Command

```
bt_tx_tone  <Parameter1>  <Parameter2>  <Parameter3>
```

<Parameter1>: SCW Tx enable signal, 1 = send; 0 = stop.

<Parameter2>: Select the SCW Tx channel from 0 ~ 78.

<Parameter3>: SCW power attenuation. The unit is 0.25 dB. For example, 4 indicates an attenuation of 1 dB.

- SCW Tx example:

  ```
  bt_tx_tone  1  0  0
  ```

  Print: `BT TX TONE START!`

  Transmit SCW in channel 14 (2402 MHz)

- SCW Tx stop example:

  ```
  bt_tx_tone  0  0  0
  ```

  Print: `BT TX TONE STOP!`

  Stop SCW transmission.

Espressif IoT Team

www.espressif.com