

ESP32-C6

Series SoC Errata Version 1.1



ESPRESSIF

Table of contents

Table of contents	i
1 Chip Revision Identification	1
1.1 Chip Revision Numbering Scheme	1
1.2 Primary Identification Methods	1
1.3 Additional Identification Methods	3
1.4 ESP-IDF Release Compatibility	4
1.5 Related Documents	5
2 Errata Summary	5
3 All Errata Descriptions	5
3.1 [CPU] Possible Deadlock Due to Out-of-Order Execution of Instructions When Writing to LP SRAM Is Involved	5
3.2 [Clock] Inaccurate Calibration of RC_FAST_CLK Clock	6
3.3 [Reset] System Reset Triggered by RTC Watchdog Timer Cannot Be Correctly Reported	6
3.4 [SPI] Enabling Flash Auto Suspend May Cause Abnormalities in Data Read	7
3.5 [RMT] The Idle State Signal Level Might Run into Error in RMT Continuous TX Mode	7
3.6 [SAR ADC] Data Duplication May Occur When SAR ADC Accessing GDMA	8
3.7 [SAR ADC] Loss of Precision in Lower Four Bits of SAR ADC	8
3.8 [Wi-Fi] ESP32-C6 Cannot Be 802.11mc FTM Initiator	9
4 Revision History	9
5 Related Documentation and Resources	10
5.1 Related Documentation	10
5.2 Developer Zone	11
5.3 Products	11
5.4 Contact Us	11

1 Chip Revision Identification

Espressif is introducing a new **vM.X** numbering scheme to indicate chip revisions. This guide outlines the structure of this scheme and provides information on chip errata and additional identification methods.

1.1 Chip Revision Numbering Scheme

The new numbering scheme **vM.X** consists of the major and minor numbers described below.

M –Major number, indicating the major revision of the chip product. If this number changes, it means the software used for the previous version of the product is incompatible with the new product, and the software version shall be upgraded for the use of the new product.

X –Minor number, indicating the minor revision of the chip product. If this number changes, it means the software used for the previous version of the product is compatible with the new product, and there is no need to upgrade the software.

The **vM.X** scheme replaces previously used chip revision schemes, including ECOx numbers, Vxxx, and other formats if any.

1.2 Primary Identification Methods

eFuse Bits

The chip revision is encoded using two eFuse fields:

- EFUSE_RD_MAC_SPI_SYS_3_REG[23:22]
- EFUSE_RD_MAC_SPI_SYS_3_REG[21:18]

Table 1.1: Chip Revision Identification by eFuse Bits

	eFuse Bit	Chip Revision		
		v0.0	v0.1	v0.2
Major Number	EFUSE_RD_MAC_SPI_SYS_3_REG[23]	0	0	0
	EFUSE_RD_MAC_SPI_SYS_3_REG[22]	0	0	0
Minor Number	EFUSE_RD_MAC_SPI_SYS_3_REG[21]	0	0	0
	EFUSE_RD_MAC_SPI_SYS_3_REG[20]	0	0	0
	EFUSE_RD_MAC_SPI_SYS_3_REG[19]	0	0	1
	EFUSE_RD_MAC_SPI_SYS_3_REG[18]	0	1	0

Chip Marking

- **Espressif Tracking Information** line in chip marking

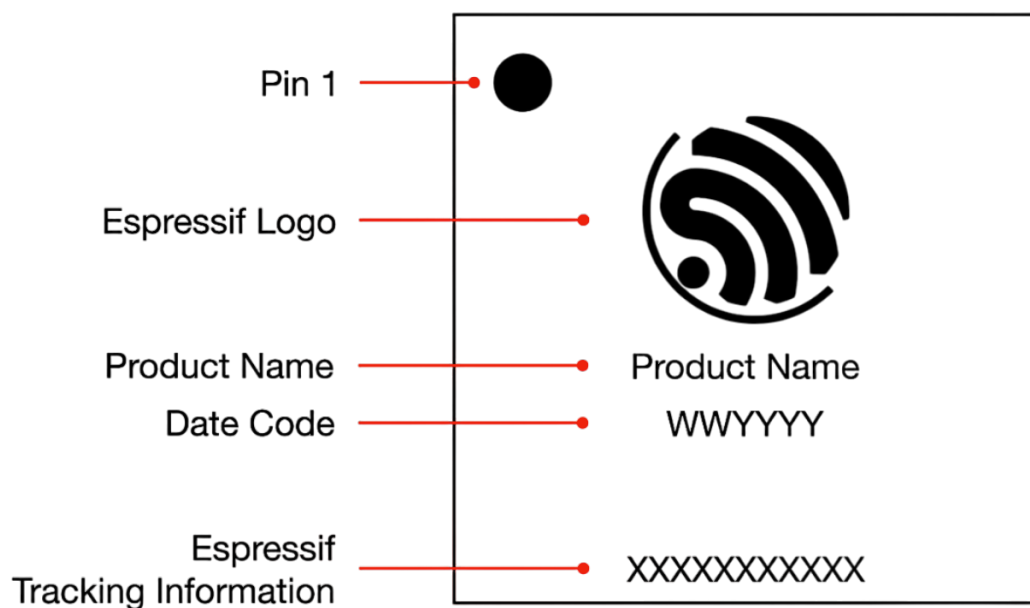


Figure 1.1: Chip Marking Diagram

Table 1.2: Chip Revision Identification by Chip Marking

Chip Revision	Espressif Tracking Information
v0.0	X A XXXXXXXXXX
v0.1	X B XXXXXXXXXX
v0.2	X C XXXXXXXXXX

Module Marking

- **Specification Identifier** line in module marking

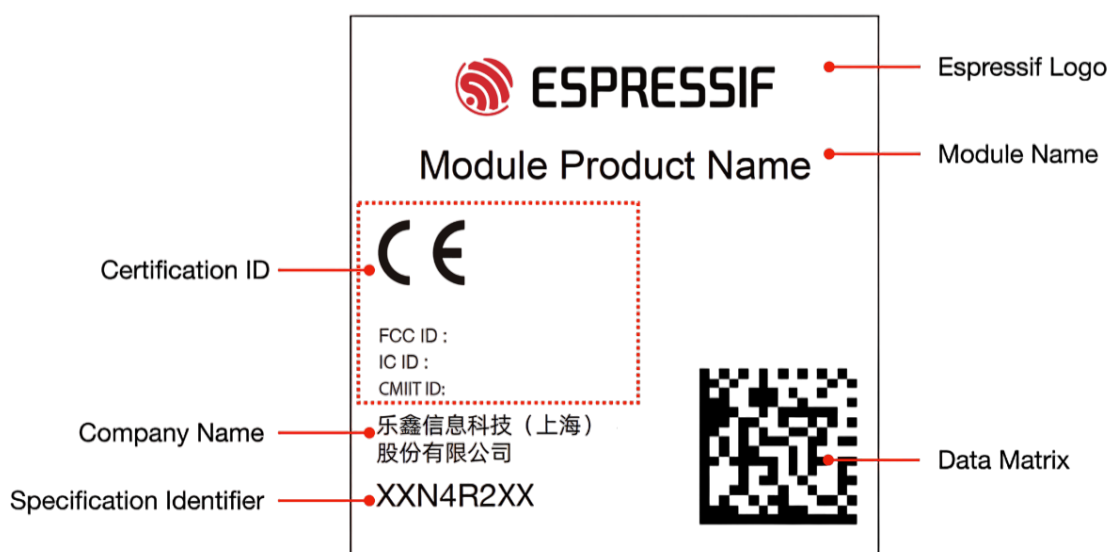


Figure 1.2: Module Marking Diagram

Table 1.3: Chip Revision Identification by Module Marking

Chip Revision	Specification Identifier
v0.0	XA XXXX
v0.1	MB XXXX
v0.2	MC XXXX

1.3 Additional Identification Methods

Date Code

Some errors in the chip product don't need to be fixed at the silicon level, or in other words in a new chip revision.

In this case, the chip may be identified by **Date Code** in chip marking (see [Chip Marking Diagram](#)). For more information, please refer to [Espressif Chip Packaging Information](#).

PW Number

Modules built around the chip may be identified by **PW Number** in product label (see [Module Product Label](#)). For more information, please refer to [Espressif Module Packaging Information](#).

 ESPRESSIF 乐鑫信息科技(上海)股份有限公司	
生产工单 PW Number	PW-2020-11-0001
产品型号 Product Name	ESP32-WROOM-32D
产品料号 Product Number	M21EH3264PH3Q0
数量 Quantity	650 pcs
固件版本 Firmware Ver	IDF: AT: FW P/N:
原产国 Country of Origin	MADE IN CHINA
生产日期 Seal Date	2020-11-30
批次号 Lot Number	202048-000001 202048-000002 202048-000003 202048-000004 202048-000005
出货检验 OQC	产品条码 QR code
	

Figure 1.3: Module Product Label

Note: Please note that **PW Number** is only provided for reels packaged in aluminum moisture barrier bags (MBB).

1.4 ESP-IDF Release Compatibility

Information about ESP-IDF release that supports a specific chip revision is provided in [Compatibility Between ESP-IDF Releases and Revisions of Espressif SoCs](#).

1.5 Related Documents

- For more information about the chip revision upgrade and their identification of series products, please refer to [ESP32-C6 Product/Process Change Notifications \(PCN\)](#).
- For more information about the chip revision numbering scheme, see [Compatibility Advisory for Chip Revision Numbering Scheme](#).

2 Errata Summary

Table 2.1: Errata summary

Category	Descriptions	Affected Revisions		
		v0.0	v0.1	v0.2
RISC-V CPU	<i>[CPU] Possible Deadlock Due to Out-of-Order Execution of Instructions When Writing to LP SRAM Is Involved</i>	Y	Y	
Clock	<i>[Clock] Inaccurate Calibration of RC_FAST_CLK Clock</i>	Y		
Reset	<i>[Reset] System Reset Triggered by RTC Watchdog Timer Cannot Be Correctly Reported</i>	Y		
SPI	<i>[SPI] Enabling Flash Auto Suspend May Cause Abnormalities in Data Read</i>	Y	Y	
RMT	<i>[RMT] The Idle State Signal Level Might Run into Error in RMT Continuous TX Mode</i>	Y	Y	Y
SAR ADC	<i>[SAR ADC] Data Duplication May Occur When SAR ADC Accessing GDMA</i>	Y	Y	
	<i>[SAR ADC] Loss of Precision in Lower Four Bits of SAR ADC</i>	Y	Y	
Wi-Fi	<i>[Wi-Fi] ESP32-C6 Cannot Be 802.11mc FTM Initiator</i>	Y	Y	

3 All Errata Descriptions

3.1 [CPU] Possible Deadlock Due to Out-of-Order Execution of Instructions When Writing to LP SRAM Is Involved

Affected revisions: v0.0 v0.1

Description

When HP CPU executes instructions (instruction A and instruction B successively) in LP SRAM, and instruction A and instruction B happen to follow the following patterns:

- Instruction A involves writing to memory. Examples: **sw/sh/sb**
- Instruction B involves only accessing the instruction bus. Examples: **nop/jal/jalr/lui/auipc**

- The address of instruction B is not 4-byte aligned

The data written by instruction A to memory is only committed after instruction B has completed execution. This introduces a risk where, after instruction A writing to memory, if an infinite loop is executed in instruction B, the writing of instruction A will never complete.

Workarounds

When you experience this problem, or when you check the assembly code and see the above mentioned pattern,

- Add a **fence** instruction between instruction A and the infinite loop. This can be achieved by using the `rv_utils_memory_barrier` interface in ESP-IDF.
- Replace the infinite loop with instruction **wfi**. This can be achieved by using the `rv_utils_wait_for_intr` interface in ESP-IDF.
- Disable the RV32C (compressed) extension when compiling code that to be executed in LP SRAM to avoid instructions with not 4-byte aligned addresses.

Solution

Fixed in chip revision v0.2.

3.2 [Clock] Inaccurate Calibration of RC_FAST_CLK Clock

Affected revisions: v0.0

Description

In the ESP32-C6 chip, the frequency of the RC_FAST_CLK clock source is too close to the reference clock (40 MHz XTAL_CLK) frequency, making it impossible to calibrate accurately. This may affect peripherals that use RC_FAST_CLK and have stringent requirements for its accurate clock frequency.

For peripherals using RC_FAST_CLK, please refer to *ESP32-C6 Technical Reference Manual > Chapter Reset and Clock* [PDF].

Workarounds

Use other clock sources instead of RC_FAST_CLK.

Solution

Fixed in chip revision v0.1.

3.3 [Reset] System Reset Triggered by RTC Watchdog Timer Cannot Be Correctly Reported

Affected revisions: v0.0

Description

When the RTC watchdog timer (RWDT) triggers a system reset, the reset source code can not be latched correctly. As a result, the reset cause reported is indeterminate and might be wrong.

Workarounds

No workaround.

Solution

Fixed in chip revision v0.1.

3.4 [SPI] Enabling Flash Auto Suspend May Cause Abnormalities in Data Read

Affected revisions: v0.0 v0.1

Description

After the flash auto suspend feature is enabled, read operations on the SPI0 bus and erase/program operations on the SPI1 bus can be executed concurrently. When software performs erase or program operations on flash via SPI1, and the cache reads flash via SPI0 from time to time, if the erase or program operation is executed first, the expected request sequence is: **ERASE or PROGRAM > SUSPEND or WFI (wait for idle) > READ**.

In practice, when the erase or program operation is executed first, the request sequence is: **ERASE or PROGRAM > READ**, which may cause data read abnormalities and program execution issues.

Workarounds

Disable the auto suspend feature.

Solution

Fixed in chip revision v0.2.

3.5 [RMT] The Idle State Signal Level Might Run into Error in RMT Continuous TX Mode

Affected revisions: v0.0 v0.1 v0.2

Description

In ESP32-C6's RMT module, if the continuous TX mode is enabled, it is expected that the data transmission stops after the data is sent for RMT_TX_LOOP_NUM_CHn rounds, and after that, the signal level in idle state should be controlled by the "level" field of the end-marker.

However, in real situation, after the data transmission stops, the channel's idle state signal level is not controlled by the "level" field of the end-marker, but by the level in the data wrapped back, which is indeterminate.

Workarounds

Users are suggested to set RMT_IDLE_OUT_EN_CHn to 1 to only use registers to control the idle level.

This issue has been bypassed since the first ESP-IDF version that supports continuous TX mode (v5.1). In these versions of ESP-IDF, it is configured that the idle level can only be controlled by registers.

Solution

No fix scheduled.

3.6 [SAR ADC] Data Duplication May Occur When SAR ADC Accessing GDMA

Affected revisions: v0.0 v0.1

Description

When the SAR ADC accesses the DMA, if the clock frequency of AHB_CLK and APB_CLK are different, multiple DMA access will be triggered. The number of repeated access is directly proportional to the frequency ratio, resulting in the same data being stored repeatedly and wasting storage space.

Workarounds

When using the SAR ADC, divide AHB_CLK by 1 to generate APB_CLK (configure the PCR_APB_DIV_NUM field to 0, which is the default value).

Solution

Fixed in chip revision v0.2.

3.7 [SAR ADC] Loss of Precision in Lower Four Bits of SAR ADC

Affected revisions: v0.0 v0.1

Description

The lower four bits of the SAR ADC data bits are missing, causing a loss of precision in the corresponding bits.

Workarounds

No workaround.

Solution

Fixed in chip revision v0.2.

3.8 [Wi-Fi] ESP32-C6 Cannot Be 802.11mc FTM Initiator

Affected revisions: v0.0 v0.1

Description

The time of T3 (i.e., time of departure of ACK from Initiator) used in 802.11mc Fine Time Measurement (FTM) cannot be acquired correctly, and as a result ESP32-C6 cannot be the FTM Initiator.

Workarounds

No workaround.

Solution

Fixed in chip revision v0.2.

4 Revision History

Table 4.1: Revision History

Date	Ver- sion	Release Notes
2024-08-	v1.1	<ul style="list-style-type: none"> • Chip Revision Identification <ul style="list-style-type: none"> – Added information about chip revision v0.2 • All Errata Descriptions <ul style="list-style-type: none"> – Added Section <i>[SPI] Enabling Flash Auto Suspend May Cause Abnormalities in Data Read</i> – Added Section <i>[SAR ADC] Data Duplication May Occur When SAR ADC Accessing GDMA</i> – Added Section <i>[SAR ADC] Loss of Precision in Lower Four Bits of SAR ADC</i> – Updated solution to “Fixed in chip revision v0.2” in Section <i>[CPU] Possible Deadlock Due to Out-of-Order Execution of Instructions When Writing to LP SRAM Is Involved</i> and Section <i>[Wi-Fi] ESP32-C6 Cannot Be 802.11mc FTM Initiator</i>
2023-11-14	v1.0	First release

5 Related Documentation and Resources

5.1 Related Documentation

- [ESP32-C6 Datasheet](#) –Specifications of the ESP32-C6 hardware.
- [ESP32-C6 Technical Reference Manual](#) –Detailed information on how to use the ESP32-C6 memory and peripherals.
- [ESP32-C6 Hardware Design Guidelines](#) –Guidelines on how to integrate the ESP32-C6 into your hardware product.
- Certificates
<https://espressif.com/en/support/documents/certificates>
- ESP32-C6 Product/Process Change Notifications (PCN)
<https://espressif.com/en/support/documents/pcns?keys=ESP32-C6>
- ESP32-C6 Advisories –Information on security, bugs, compatibility, component reliability.
<https://espressif.com/en/support/documents/advisories?keys=ESP32-C6>
- Documentation Updates and Update Notification Subscription
<https://espressif.com/en/support/download/documents>

5.2 Developer Zone

- [ESP-IDF Programming Guide for ESP32-C6](#) –Extensive documentation for the ESP-IDF development framework.
- ESP-IDF and other development frameworks on GitHub.
<https://github.com/espressif>
- ESP32 BBS Forum –Engineer-to-Engineer (E2E) Community for Espressif products where you can post questions, share knowledge, explore ideas, and help solve problems with fellow engineers.
<https://esp32.com/>
- The ESP Journal –Best Practices, Articles, and Notes from Espressif folks.
<https://blog.espressif.com/>
- See the tabs SDKs and Demos, Apps, Tools, AT Firmware.
<https://espressif.com/en/support/download/sdks-demos>

5.3 Products

- ESP32-C6 Series SoCs –Browse through all ESP32-C6 SoCs.
<https://espressif.com/en/products/socs?id=ESP32-C6>
- ESP32-C6 Series Modules –Browse through all ESP32-C6-based modules.
<https://espressif.com/en/products/modules?id=ESP32-C6>
- ESP32-C6 Series DevKits –Browse through all ESP32-C6-based devkits.
<https://espressif.com/en/products/devkits?id=ESP32-C6>
- ESP Product Selector –Find an Espressif hardware product suitable for your needs by comparing or applying filters.
<https://products.espressif.com/#/product-selector>

5.4 Contact Us

- See the tabs Sales Questions, Technical Enquiries, Circuit Schematic & PCB Design Review, Get Samples (Online stores), Become Our Supplier, Comments & Suggestions.
<https://espressif.com/en/contact-us/sales-questions>