ESP32-S2 Series SoC

Errata

Introduction

This document describes known errata in ESP32-S2 series of SoCs.

Version 1.1

Espressif Systems

Contents

Chip Identification

Note:

Check the link or the QR code to make sure that you use the latest version of this document:
https://espressif.com/sites/default/files/documentation/esp32-s2_errata_en.pdf

1 Chip Revision

Espressif is introducing **vM.X** numbering scheme to indicate chip revisions.

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.

The chip revision is identified by:

- eFuse field EFUSE_RD_MAC_SPI_SYS_3_REG[20:18] and EFUSE_RD_MAC_SPI_SYS_4_REG[6:4]

Table 1: Chip Revision Identification by eFuse Bits

	eFuse Bit	Chip Revision	
		v0.0	v1.0
Major Number	EFUSE_RD_MAC_SPI_SYS_3_REG[19]	0	0
	EFUSE_RD_MAC_SPI_SYS_3_REG[18]	0	1
Minor Number	EFUSE_RD_MAC_SPI_SYS_3_REG[20]	0	0
	EFUSE_RD_MAC_SPI_SYS_4_REG[6]	0	0
	EFUSE_RD_MAC_SPI_SYS_4_REG[5]	0	0
	EFUSE_RD_MAC_SPI_SYS_4_REG[4]	0	0

- **Main Die** line in chip marking

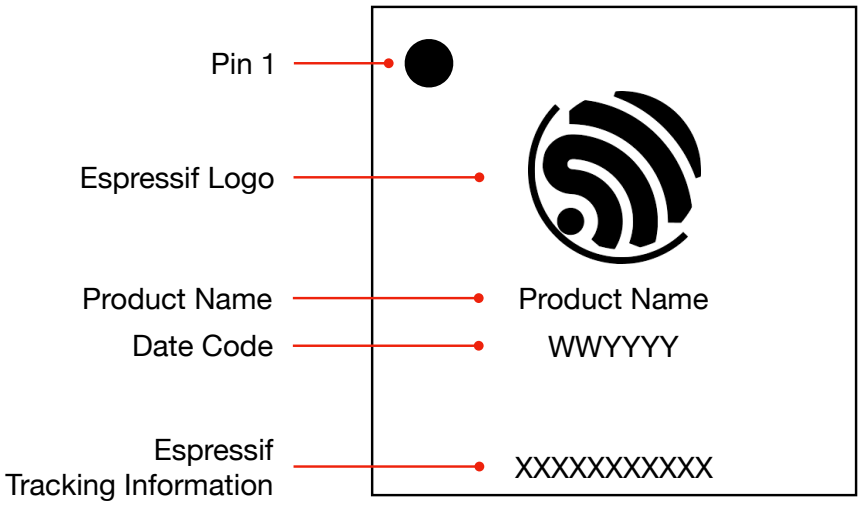

Figure 1: Chip Marking Diagram

Table 2: Chip Revision Identification by Silk Print

Chip Revision	Main Die
v0.0	XAXXXXXXX
v1.0	XBXXXXXXX

Note:

- Information about ESP-IDF release that supports a specific chip revision is provided in Compatibility Between ESP-IDF Releases and Revisions of Espressif SoCs.
- For more information about the chip revision upgrade and their identification of ESP32-S2 series products, please refer to ESP32-S2 Product/Process Change Notifications (PCN).
- For more information about the chip revision numbering scheme, see Compatibility Advisory for Chip Revision Numbering Scheme.

2 Additional Methods

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 Figure 1). For more information, please refer to *Espressif Chip Packaging Information*.

Modules built around the chip may be identified by **PW Number** in product label (see Figure 2). For more information, please refer to *Espressif Module Packaging Information*.

Figure 2: Module Product Label

Note:

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

Errata Description

Table 3: Errata Summary

Category	Description	Affected Revisions[1]	
		v0.0	v1.0
System	3.1 *Leakage current at the VDDA and VDD3P3_RTC pin during shutdown*	Y	
	3.2 *Random flash download failure*	Y	
RTC I2C	4.1 *The falling edge of RTC_I2C_RESET triggers reset at low temperature*	Y	
SPI	5.1 *SPI is stuck after soft restart from auto suspension*	Y	
USB OTG	6.1 *Abnormal data during AHB bus arbitration by USB OTG*	Y	
SAR ADC	7.1 *Bit 1 of SAR ADC does not flip*	Y	
RTC	8.1 *RTC register read error after wake-up from Light-sleep mode*	Y	
Touch Sensor	9.1 *The TOUCH_SCAN_DONE_INT interrupt raw data value is undefined*	Y	
	9.2 *The scan done interrupt RTC_CNTL_TOUCH_SCAN_DONE_INT_ENA occurs twice during a single scan*	Y	

[1] Y* means some batches of a revision are affected.

3 System

3.1 Leakage current at the VDDA and VDD3P3_RTC pin during shutdown

Description

When a chip is connected to the power supply, but the CHIP_PU pin is held low (meaning that the chip powers off), there will be a leakage current in the μA range at power pins such as VDDA and VDD3P3_RTC.

Workarounds

None.

Solution

Fixed in chip revision v1.0.

3.2 Random flash download failure

Description

In download mode, the first stage bootloader in ROM receives serial data from two different input pins. Among the two intput pins, pin 24 DAC_2 (GPIO18) is not pulled up by default. If this pin is not pulled up in PCB design

and is left floating, in download mode the first stage bootloader will not function properly (including download applications) due to interference.

Workarounds

This problem can be bypassed in PCB design by pulling up pin 24 DAC_2. The typical value of the pull-up resistor is 10 kΩ. All official development boards by Espressif pull this pin up, while official modules are not.

Solution

Fixed in chip revision v1.0 by pulling pin 24 up by default.

4 RTC I2C

4.1 The falling edge of RTC_I2C_RESET triggers reset at low temperature

Description

At −40 °C, the chip will be restarted during wake-up.

Workarounds

None.

Solution

Fixed in chip revision v1.0.

5 SPI

5.1 SPI is stuck after soft restart from auto suspension

Description

After auto suspend is enabled, if caching is requested while Memory SPI is erasing flash, Memory SPI will automatically send a SUSPEND command (0×75). If there is a system reset, and Memory SPI is restarted before sending a RESUME command (0×7A), the state machine of Memory SPI will not be restored. As a result, the system cannot continue operations.

Workarounds

Disable auto suspend function.

Solution

Fixed in chip revision v1.0.

6 USB OTG

6.1 Abnormal data during AHB bus arbitration by USB OTG

Description

When the USB OTG peripheral and some other competing peripherals (listed below) simultaneously execute a request on the Advanced High-performance Bus (AHB), the AHB may generate incorrect arbitration signals, which results in the USB OTG peripheral reading or writing erroneous data. The competing peripherals include:

- I2S
- SPI

Workarounds

1. Avoid AHB bus competition between USB OTG and above peripherals by not using DMA mode of USB OTG, or disabling DMA mode of above peripherals.
2. Avoid competing with the USB OTG's AHB bus access. Specifically, set USB OTG's AHB burst transfer mode to INCR to prevent competition from the other peripherals. In this mode, USB OTG will occupy the AHB bus exclusively until the burst transfer is completed.

> **Note:**
> Use the INCR burst mode with care, as it requires adjustment to maximum packet size (MPS) for USB OTG endpoints, so that burst time is smaller than the timeout period of the competing peripherals.

Solution

Fixed in chip revision v1.0. With this fix, the AHB bus will correctly arbitrate competing access.

ESP-IDF adds USB OTG support starting from v4.4. When the specific conditions listed below are met, ESP-IDF enables the INCR mode workaround, i.e., using the INCR mode to guarantee that the USB OTG's exclusive access to the AHB. The conditions for ESP-IDF to enable this workaround are as follows:

1. For chip revision v0.0, ESP-IDF always enables the workaround.
2. ESP-IDF added support for chip revision v1.0 in ESP-IDF v4.4.6, v5.0.4, v5.1.2, and v5.2. In these and above version, the software automatically detects the chip revision. When chip revision v1.0 or later revisions are detected, ESP-IDF no longer enables the workaround.
3. In ESP-IDF versions that do not support chip revision v1.0, i.e., v4.4-v4.4.5, v5.0-v5.0.3, v5.1-v5.1.1, ESP-IDF always enables the workaround.

7 SAR ADC

7.1 Bit 1 of SAR ADC does not flip

Description

Bit 1 of SAR ADC is always 0, and does not change with measured voltage.

Workarounds

None.

Solution

Fixed in chip revision v1.0. The effective resolution of SAR ADC on chip revision v1.0 is changed from 13 bits to 12 bits. That is, bit 0 is not valid, and the valid bits are bit 1 ~ bit 12 inclusive.

8 RTC

8.1 RTC register read error after wake-up from Light-sleep mode

Description

If an RTC peripheral is turned off in Light-sleep mode, there is a certain probability that after waking up from Light-sleep, the CPU of ESP32-S2 will read the registers in the RTC power domain incorrectly.

Workarounds

No workaround. Users are suggested not to power down RTC peripherals in Light-sleep mode. There will be no impact on power consumption.

Solution

No fix scheduled.

9 Touch Sensor

9.1 The TOUCH_SCAN_DONE_INT interrupt raw data value is undefined

Description

For ESP32-S2's touch sensor, the raw data value is undefined for the first two TOUCH_SCAN_DONE_INT interrupts.

Workarounds

Users are suggested to skip the first two TOUCH_SCAN_DONE_INT interrupts, then turn them off and stop using them.

This issue has been bypassed in all versions of ESP-IDF through this method.

This issue has been bypassed in the Touch Element component (touch_element) in ESP-IDF (introduced in ESP-IDF release version v4.3). If you are directly developing on the lower-level touch sensor driver, please follow the implementation provided within the Touch Element component and the recommendations mentioned above to bypass the issue.

Solution

No fix scheduled.

9.2 The scan done interrupt RTC_CNTL_TOUCH_SCAN_DONE_INT_ENA occurs twice during a single scan

Description

The touch sensor of ESP32-S2 triggered the scan done interrupt RTC_CNTL_TOUCH_SCAN_DONE_INT_ENA twice during a single scan, occurring when scanning the last two channels.

Workarounds

Users are suggested to register one more interrupt in the RTC driver to filter, checking if the current measuring channel is the last channel. If it is not the last channel, then clear the RTC_CNTL_TOUCH_SCAN_DONE_INT_ENA interrupt directly. If it is, then the current RTC_CNTL_TOUCH_SCAN_DONE_INT_ENA interrupt can be regarded as a valid interrupt.

This issue has been bypassed in all versions of ESP-IDF through this method.

Solution

No fix scheduled.

Related Documentation and Resources

Related Documentation

- ESP32-S2 Series Datasheet – Specifications of the ESP32-S2 hardware.
- ESP32-S2 Technical Reference Manual – Detailed information on how to use the ESP32-S2 memory and peripherals.
- ESP32-S2 Hardware Design Guidelines – Guidelines on how to integrate the ESP32-S2 into your hardware product.
- *Certificates*
  https://espressif.com/en/support/documents/certificates
- *ESP32-S2 Product/Process Change Notifications (PCN)*
  https://espressif.com/en/support/documents/pcns?keys=ESP32-S2
- *ESP32-S2 Advisories* – Information on security, bugs, compatibility, component reliability.
  https://espressif.com/en/support/documents/advisories?keys=ESP32-S2
- *Documentation Updates and Update Notification Subscription*
  https://espressif.com/en/support/download/documents

Developer Zone

- ESP-IDF Programming Guide for ESP32-S2 – 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

Products

- *ESP32-S2 Series SoCs* – Browse through all ESP32-S2 SoCs.
  https://espressif.com/en/products/socs?id=ESP32-S2
- *ESP32-S2 Series Modules* – Browse through all ESP32-S2-based modules.
  https://espressif.com/en/products/modules?id=ESP32-S2
- *ESP32-S2 Series DevKits* – Browse through all ESP32-S2-based devkits.
  https://espressif.com/en/products/devkits?id=ESP32-S2
- *ESP Product Selector* – Find an Espressif hardware product suitable for your needs by comparing or applying filters.
  https://products.espressif.com/#/product-selector?language=en

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

Revision History

Date	Version	Release Notes
2023-	v1.1	• Added Chapter 2 *Additional Methods* • Added a note under Table 3
2022-09-19	v1.0	First release

www.espressif.com

Disclaimer and Copyright Notice