G32A10x5

Errata Sheet

Version: V 1.0

Content

1 Introduction

This document describes known limitations of the G32A10x5 series products. If your application matches any scenario described in this errata sheet, use the recommended solution. If no solution is provided, avoid using the product in that scenario.

2 Errata List

Table 1 Errata List

Category	Introduction
USART	Automatic baud rate detection Mode 1 does not check bit-duration consistency.
	When automatic baud rate detection fails, the frame error flag (FEFLG) remains 0.
	If TXEN is cleared immediately after TXDATA is written, the last byte written to TXDATA may not be transmitted.
	A Break request may prevent the TXCFLG flag from being set.
	When RXEN=0 or UEN=0, the RTS signal remains active.
SMS	RCM->APBRST2[SMSRST] cannot reset the ECC_LOGn and INTRRn registers.
CAN	Priority inversion may occur inside the message handler when TX scan selects the next high-priority message
	The PSR.PXE flag may not always be cleared by a host read
	Incomplete IR.TCF bit description in the CAN Interrupt Register
TMR	In auto output enable mode, PWM may be re-enabled after a system break event.
	TRGO trigger output may fail.
	Output compare clear may fail when the counter is reset externally.

Note: √ indicates present, × indicates not present.

3 USART

3.1 Automatic baud rate detection Mode 1 does not check bit-duration consistency

Problem Description

In automatic baud rate detection Mode 1 (ABRDCFG=01), the USART measures the start bit duration and the first data bit duration. If either measured value is within the valid range, baud rate detection is considered successful, even if the two durations are inconsistent. Therefore, Mode 1 gives reliable results for normal input frames, but not for abnormal input frames.

Solution

Use another detection mode, such as Mode 0, Mode 2, or Mode 3.

3.2 When automatic baud rate detection fails, the frame error flag (FEFLG) remains 0

Problem Description

When the ABRDEFLG flag is set to indicate an automatic baud rate detection error, the frame error flag (FEFLG) should also be set. However, FEFLG remains 0.

Solution

Check only the ABRDEFLG flag to detect automatic baud rate errors. Do not rely on FEFLG.

3.3 If TXEN is cleared immediately after TXDATA is written, the last byte written to TXDATA may not be transmitted

Problem Description

If the USART baud rate is low and TXEN is cleared immediately after the last write to TXDATA, the last byte may not be transmitted.

Solution

Take one of the following measures:

- Wait until the TXBEFLG flag is set before clearing TXEN.
- Wait until the TXCFLG flag is set before clearing TXEN.

3.4 A Break request may prevent the TXCFLG flag from being set

Problem Description

After data D1 is transmitted, the transmission complete flag (TXCFLG) may not be set if all of the following conditions are met:

- CTS hardware flow control is enabled
- D1 transmission is in progress
- A Break transmission is requested before D1 transmission ends
- The nCTS signal becomes inactive before D1 transmission ends

As a result, software that depends on TXCFLG may not detect the end of transmission.

Solution

In the application, issue a Break request only after TXCFLG has been set.

3.5 When RXEN=0 or UEN=0, the RTS signal remains active

Problem Description

After RTSEN is set, the RTS pin is pulled low and becomes active, even if the USART is disabled (UEN=0), or the receiver is disabled (RXEN=0). In this state, the USART is not actually ready to receive data.

Solution

Configure the I/O pin used for RTS as an alternate-function pin only after UEN and RXEN have been set.

4 SMS

4.1 RCM->APBRST2[SMSRST] cannot reset the ECC_LOGn and INTRRn registers

Problem Description

When the SMS module is reset using the APBRST2 register in the RCM module, the ECC_LOGn and INTRRn registers are not cleared because they are in a different clock domain. These registers require a system reset to be cleared.

Solution

For the SMS->INTRRn register, write 0 to reset it. For the SMS->ECC_LOGn register, treat the value as valid only when an error flag is set.

5 CAN

5.1 Priority inversion may occur inside the message handler when TX scan selects the next high-priority message

Problem Description

Assume that the Tx message handler output pipeline contains two messages from the Tx FIFO:

Position 1: Tx FIFO message 1

Position 2: Tx FIFO message 2

Position 3: --

Tx FIFO message 2 has a higher priority than Tx FIFO message 1.

A non-Tx-FIFO message is then requested for transmission. Its priority is higher than Tx FIFO message 1 but lower than Tx FIFO message 2.

Because the non-Tx-FIFO message has lower priority than Tx FIFO message 2, the two Tx FIFO messages remain in Position 1 and Position 2.

As a result, Tx FIFO message 1, which has lower priority than the non-Tx-FIFO message, is transmitted first. The non-Tx-FIFO message is not placed before the two Tx FIFO messages. This is an internal priority inversion.

This can occur when dedicated Tx buffers and the Tx FIFO are used at the same time (TXBC.TFQM='0'), and the second message in the Tx FIFO has higher priority than the first message. In this case, a high-priority non-Tx-FIFO message may be transmitted after a lower-priority Tx FIFO message.

Solutions

When transmitting high-priority data from a dedicated Tx buffer while the Tx FIFO contains lower-priority data, use one of the following solutions:

Solution 1

Use two dedicated Tx buffers, such as Tx buffers 4 and 5, instead of the Tx FIFO. The following pseudocode can replace the function that fills the Tx FIFO.

Write the first message to Tx buffer 4.

Transmission loop:

- Request transmission of Tx buffer 4 — write TXBAR.AR4
- Write the next message to Tx buffer 5
- Wait until Tx buffer 4 transmission is complete — IR.TC, read TXBTO.TO4
- Request transmission of Tx buffer 5 — write TXBAR.AR5
- Write the next message to Tx buffer 4
- Wait until Tx buffer 5 transmission is complete — IR.TC, read TXBTO.TO45

Solution 2

Ensure that only one Tx FIFO message is queued for transmission at a time.

Multiple Tx FIFO messages may be prepared at any time, but their transmission requests must be issued one by one. After each Tx FIFO message is transmitted and the Tx FIFO becomes empty (IR.TFE='1'), request transmission of the next Tx FIFO message.

Solution 3

Use only the Tx FIFO. Send high-priority messages through the Tx FIFO as well.

Disadvantage: A high-priority message must wait until all earlier Tx FIFO messages have been transmitted.

Solution 4

Cancel the Tx FIFO transmission.

- Request transmission of the non-Tx-FIFO message as usual — write TXBAR.ARx
- Cancel all requested Tx FIFO messages with one write to the TXBCR register
- Wait until Tx FIFO cancellation is complete (TXCR==0) and the non-Tx-FIFO message has been transmitted — IR.TC, read TXBTO.TOx
- Refill all canceled but untransmitted Tx FIFO messages (TXBTO.TOyy==0) into message RAM, then request their transmission again — write TXBAR.ARyy

5.2 The PSR.PXE flag may not always be cleared by a host read

Problem Description

CAN has two configuration flags: CCCR.FDOE and CCCR.PXHD.

(1) When CCCR.FDOE is set to “0”, CAN operates as a classic CAN CC controller and does not support CAN FD frames. In this mode, CCCR.PXHD is ignored. The controller does not tolerate received CAN FD frames, which causes an error state and generates an error flag.

(2) When CCCR.FDOE is set to “1”, CAN operates as a CAN FD controller and supports both CAN FD and CAN CC frames. In this mode, CCCR.PXHD determines how the controller handles a recessive bit detected after the FDF recessive bit in a received frame. If CCCR.PXHD= “1”, the condition is treated as a format error. Otherwise, it is treated as a protocol exception event. In that case, CAN re-synchronizes by waiting for the next bus idle state. This mechanism allows compatibility with frame formats that were not yet defined when CAN FD was introduced, such as CAN XL.

(3) When a protocol exception event is detected, the PSR.PXE status bit is set to “1”.

(4) The PSR status register is updated once per CAN bit time by the protocol controller finite state machine (FSM). PSR.PXE should be cleared after each read of the PSR register. However, if the read occurs in the same clock cycle in which the FSM updates the register, PSR.PXE may not be cleared.

(5) This erratum applies only when CAN detects a protocol exception event (PSR.PXE='1') and

the CPU tries to clear PSR.PXE by reading the PSR register.

(6) Failure to clear PSR.PXE does not affect the protocol exception function. Even if PSR.PXE was not cleared, the protocol exception event still sets PSR.ACT="00" to indicate the synchronized state.

Solutions

Workaround 1

Disable protocol exception handling by setting CCCR.PXHD to “1”.

Workaround 2

Read the PSR register repeatedly until PSR.PXE= “0”.

5.3 Incomplete IR.TCF bit description in the CAN Interrupt Register

Problem Description

(1) The IR.TCF bit is set only when a regular cancellation is completed. A regular cancellation is started by setting the corresponding TXBCR bit for a Tx buffer. The corresponding TXBCIE bit must also be enabled.

(2) In DAR mode, the TXBCF bit for a Tx buffer with a failed transmission is set because only successful transmissions generate entries in the Tx event FIFO. This condition is not intended to trigger an interrupt.

(3) Supplementary note for bit 10, TCF, in the IR register:

(4) The IR.TCF bit is set only when a regular cancellation is completed. A regular cancellation is started by setting the CR bit in the Tx buffer cancellation request register (TXBCR).

(5) Note: In DAR mode, all failed transmissions are automatically canceled. For each failed transmission, the corresponding CF bit in the Tx buffer cancellation finished register (TXBCF) is set.

(6) In this case, although the CF bit in TXBCF is set, the TCF bit is not set because the cancellation was not requested by setting the CR bit in TXBCR.

(7) This erratum applies only to message transmission in DAR mode. Normal CAN/CAN FD operation is not affected.

(8) In DAR mode, IR.TCF is not set for failed transmissions.

Solution

Update the documentation.

6 TMR

6.1 In auto output enable mode, PWM may be re-enabled after a system break event

Problem Description

(1) In auto output enable mode, when the AOEN bit in the TMRx_BDT register is set to 1, the break input is typically used for cycle-by-cycle PWM control, such as current-mode regulation. When a break signal is received, usually from a comparator with a current threshold, the PWM output is disabled and then re-enabled in the next counter cycle.

(2) However, a system break event, typically triggered by the CSS clock security system, is intended to fully stop PWM output to prevent abnormal system operation, such as an incorrect PWM frequency.

(3) In the current hardware implementation, the timer system break input is not latched. Therefore, when a system break event occurs, PWM output is disabled immediately, but is automatically re-enabled in the next counter cycle.

Solution

Use the output clear enable function, the OCxCEN bit in the TMRx_CCM1/CCM2 register, for the control loop when possible. Use the hardware break circuit only for internal and/or external fault protection (AOEN=0).

6.2 TRGO trigger output may fail

Problem Description

Timers can be cascaded through the ITRx input and TRGO output. TRGO can also be used as a trigger signal for other peripherals, such as the ADC. Because this circuit uses pulse generation, observe the following requirements when initializing master/slave peripherals or using different master/slave clock frequencies:

- If the master timer generates a TRGO trigger pulse before the target peripheral clock is enabled, the trigger may be lost.
- If the target peripheral frequency is changed during operation, such as by changing the clock prescaler, the trigger may be lost.

In summary, the slave timer or slave peripheral clock must be enabled before it receives events from the master timer. Its clock frequency must not be changed while it is receiving trigger signals from the master timer.

Solution

No additional workaround is required or available if the application correctly handles clock enabling and timing as described above.

6.3 Output compare clear may fail when the counter is reset externally

Problem Description

When the timer is configured in slave reset mode, the output compare clear event may not be generated correctly.

If the following sequence occurs, the PWM output remains inactive for one extra PWM cycle:

- The output is cleared by a clear event.
- The timer is reset before the programmed compare event occurs.

Solution

Take one of the following measures:

- Use the BKIN input to clear the output: use BKIN to clear the output and select auto output enable mode by setting AOEN=1.
- Mask the timer reset while the PWM output is active: prevent the reset from occurring before the compare event. For example, use an unused timer compare channel, configure it as an open-drain output, and connect it to the reset signal to pull the timer reset line low.

7 Revision History

Table 2 Document Revision History

Date	Version	Revision History
July, 2006	1.0	● Initial release

Statement

This document is formulated and published by Geehy Semiconductor Co., Ltd. (hereinafter referred to as “Geehy”). The contents in this document are protected by laws and regulations of trademark, copyright and software copyright. Geehy reserves the right to make corrections and modifications to this document at any time. Read this document carefully before using Geehy products. Once you use the Geehy product, it means that you (hereinafter referred to as the “users”) have known and accepted all the contents of this document. Users shall use the Geehy product in accordance with relevant laws and regulations and the requirements of this document.

1. Ownership

This document can only be used in connection with the corresponding chip products or software products provided by Geehy. Without the prior permission of Geehy, no unit or individual may copy, transcribe, modify, edit or disseminate all or part of the contents of this document for any reason or in any form.

The “极海” or “Geehy” words or graphics with “®” or “™” in this document are trademarks of Geehy. Other product or service names displayed on Geehy products are the property of their respective owners.

2. No Intellectual Property License

Geehy owns all rights, ownership and intellectual property rights involved in this document.

Geehy shall not be deemed to grant the license or right of any intellectual property to users explicitly or implicitly due to the sale or distribution of Geehy products or this document.

If any third party’s products, services or intellectual property are involved in this document, it shall not be deemed that Geehy authorizes users to use the aforesaid third party’s products, services or intellectual property. Any information regarding the application of the product, Geehy hereby disclaims any and all warranties and liabilities of any kind, including without limitation warranties of non-infringement of intellectual property rights of any third party, unless otherwise agreed in sales order or sales contract.

3. Version Update

Users can obtain the latest document of the corresponding models when ordering Geehy products.

If the contents in this document are inconsistent with Geehy products, the agreement in the sales

order or the sales contract shall prevail.

4. Information Reliability

The relevant data in this document are obtained from batch test by Geehy Laboratory or cooperative third-party testing organization. However, clerical errors in correction or errors caused by differences in testing environment may occur inevitably. Therefore, users should understand that Geehy does not bear any responsibility for such errors that may occur in this document. The relevant data in this document are only used to guide users as performance parameter reference and do not constitute Geehy's guarantee for any product performance.

Users shall select appropriate Geehy products according to their own needs, and effectively verify and test the applicability of Geehy products to confirm that Geehy products meet their own needs, corresponding standards, safety or other reliability requirements. If losses are caused to users due to user's failure to fully verify and test Geehy products, Geehy will not bear any responsibility.

5. Legality

USERS SHALL ABIDE BY ALL APPLICABLE LOCAL LAWS AND REGULATIONS WHEN USING THIS DOCUMENT AND THE MATCHING GEEHY PRODUCTS. USERS SHALL UNDERSTAND THAT THE PRODUCTS MAY BE RESTRICTED BY THE EXPORT, RE-EXPORT OR OTHER LAWS OF THE COUNTRIES OF THE PRODUCTS SUPPLIERS, GEEHY, GEEHY DISTRIBUTORS AND USERS. USERS (ON BEHALF OR ITSELF, SUBSIDIARIES AND AFFILIATED ENTERPRISES) SHALL AGREE AND PROMISE TO ABIDE BY ALL APPLICABLE LAWS AND REGULATIONS ON THE EXPORT AND RE-EXPORT OF GEEHY PRODUCTS AND/OR TECHNOLOGIES AND DIRECT PRODUCTS.

6. Disclaimer of Warranty

THIS DOCUMENT IS PROVIDED BY GEEHY "AS IS" AND THERE IS NO WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE, TO THE EXTENT PERMITTED BY APPLICABLE LAW.

GEEHY'S PRODUCTS ARE NOT DESIGNED, AUTHORIZED, OR WARRANTED FOR USE AS CRITICAL COMPONENTS IN MILITARY, LIFE-SUPPORT, POLLUTION CONTROL, OR HAZARDOUS SUBSTANCES MANAGEMENT SYSTEMS, NOR WHERE FAILURE COULD RESULT IN INJURY, DEATH, PROPERTY OR ENVIRONMENTAL DAMAGE.

IF THE PRODUCT IS NOT LABELED AS "AUTOMOTIVE GRADE," IT SHOULD NOT BE CONSIDERED SUITABLE FOR AUTOMOTIVE APPLICATIONS. GEEHY ASSUMES NO LIABILITY FOR THE USE BEYOND ITS SPECIFICATIONS OR GUIDELINES.

THE USER SHOULD ENSURE THAT THE APPLICATION OF THE PRODUCTS COMPLIES WITH ALL RELEVANT STANDARDS, INCLUDING BUT NOT LIMITED TO SAFETY, INFORMATION SECURITY, AND ENVIRONMENTAL REQUIREMENTS. THE USER ASSUMES FULL RESPONSIBILITY FOR THE SELECTION AND USE OF GEEHY PRODUCTS. GEEHY WILL BEAR NO RESPONSIBILITY FOR ANY DISPUTES ARISING FROM THE SUBSEQUENT DESIGN OR USE BY USERS.

7. Limitation of Liability

IN NO EVENT, UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL GEEHY OR ANY OTHER PARTY WHO PROVIDES THE DOCUMENT AND PRODUCTS "AS IS", BE LIABLE FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, DIRECT, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE DOCUMENT AND PRODUCTS (INCLUDING BUT NOT LIMITED TO LOSSES OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY USERS OR THIRD PARTIES). THIS COVERS POTENTIAL DAMAGES TO PERSONAL SAFETY, PROPERTY, OR THE ENVIRONMENT, FOR WHICH GEEHY WILL NOT BE RESPONSIBLE.

8. Scope of Application

The information in this document replaces the information provided in all previous versions of the document.