SPC563M64
Errata sheet

SPC563M64 device errata
JTAG_ID = 0x2AE01041

Introduction

This errata sheet describes all the functional and electrical problems known in the revision 2.1 of the SPC563M64 devices, identified with the JTAG_ID = 0x2AE01041 and the Flash controller micro-code version 3.7.

All the topics covered in this document refer to *RM0015* rev 4 and *SPC563M64 datasheet* rev 8 (see *B.1: Reference document*).

Device identification:

- JTAG_ID = 0x2AE01041
- MIDR register:
  - MAJOR_MASK[3:0]: 2
  - MINOR_MASK[3:0]: 1
- Flash micro-code (EmbAlgo) version 3.7

Package device marking mask identifier: BB

This errata sheet applies to SPC563M64 device in accordance with *Table 1*.

Table 1. Device summary

Part number	Package
SPC563M64L5	LQFP144
SPC563M64L7	LQFP176

Contents

List of tables

1 Functional problems

1.1 e4890 PS: Pad Ring: CANTXC and eTPUA25 pins changed to medium speed pins

Description:

The pad types of the CAN_C_TX_/GPIO[87] and eTPUA[25]/IRQ[13]/DSPI_C_SCK_LVDS+/GPIO[139] pins were changed from slow speed pads as defined in the current customer documentation (*RM0015* and *SPC563M64 datasheet*, see *Appendix B: Further information*) to medium speed pads.

Workaround:

Expect that the CAN_C_TX_/GPIO[87] and eTPUA[25]/IRQ[13]/DSPI_C_SCK_LVDS+/GPIO[139] pins have faster slew rates (medium speed instead of slow speed). The documentation will be updated.

1.2 e6988 PS : SPC563M: MIDR MASKNUM field is set to 0x21

Description:

The mask revision field (MASKNUM[Major, Minor]) of the MCU identification register is 0b0010_0001 (0x21).

Workaround:

Expect that the MASKNUM fields of the MIDR register will change in the future.

1.3 e8141 PS : FLASH: PFCR3 is not directly writable

Description:

The Flash configuration register 3 (PFCR3) that can control the prefetching settings (data prefetch enable [DPFEN], instruction prefetch enable [IPFEN], prefetch limit [PFLIM], and buffer enable [BFEN]) of the bank 1 (array 1 and array 2) flash modules is not directly writable. These settings are enabled by setting the global configuration enable bit in the Flash bus interface unit control register (BIUCR).

Workaround:

Set the GCE bit (BIUCR[GCE=1) to allow the bank 0, array 0 prefetch settings to also control bank 1 (array 1 and array 2); or program a default value for the PFCR3 register that gets loaded into the register at reset into the Flash shadow block at address 0x00FF_FE08.

1.4 e8431 PS: NEXUS: EVTI not functional on QFP176 and BGA208 packages

Description:

Event In (EVTI) is an input that is read on the negation of TRST (or JCOMP) to enable (if asserted) or disable (if deasserted) the Nexus debug port. After reset, EVTI is an input which, when asserted, will initiate one of two events based on the EIC (EVTI control) bits in the DC1 (development control 1) register (if the Nexus Class 2+ module is enabled at reset):

1. Program trace and data trace synchronization messages (provided program trace and EIC = 0b00).
2. Debug request to e200z335 Nexus Class 1 module (provided EIC = 0b01 and this feature is implemented).

Workaround:

1. Do not expect Program Trace Sync messages after EVTI assertion. Other condition for the sync messaging are not impacted.
2. Do not use EVTI to request the CPU to enter the debug state. Other requests are functional. In case EVTI functionality is needed, CSP496 package can also be used to emulate the 176QFP or the BGA208 packages.

1.5 e6865252 PDM : Pad Ring: Leakage if VDDE is greater than VDD33

Description:

If the VDDEx supplies (provided by an external supply) are greater than the VDD33 supplies (provided by the internal regulator), leakage current can occur through all pins powered by VDDEx from the VDDEx supply on the pad output driver through the pad towards ground. The highest leakage current occurs at high temperatures and is exponentially proportional to the VDDEx-VDD33 differential. Worst case leakage, per grounded pad at 150 C is 29 uA with a 200 mV differential, and 590 uA with a 400 mV differential in the VDDEx-VDD33. Any I/O configured as an input with the weak pull down enabled will rise towards VDDE level as the VDDE-VDD33 voltage differential increases (as the leakage current exceeds the weak pull-down capability). The reset state of most Nexus pads is pull-down, so this would not be guaranteed. EVTI is pulled up internally during and after RESET. EVTO must be pulled low externally for auto-baud rate detection. I/O pads configured as outputs driving LOW will remain below VOL level but will consume the leakage current through the pad driver. External logic driving pads configured as inputs will have to sink this leakage current when driving LOW.

Workaround:

Maintain a VDDE-VDD33 voltage difference below 200 mV. If VDDE is greater than 3.45 V, the PMC_TRIMR[VDD33TRIM] for the internal regulator can be increased to 4 steps above typical (0b1011) to increase VDD33 default voltage by a nominal value of 120 mV.

1.6 e6860916 PDM: eQADC: 25% calibration channel sampling requires at least 64 sampling cycles

Description:

The 25% (VRH - VRL) calibration channel (ADC channel 44) do not convert to specification with an ADC sample time less than 64 cycles.

Workaround:

For accurate calibration, the 25% calibration channel should be converted using the long sample time (LST) setting for either 64 or 128 ADC sample cycles in the ADC conversion command message (LST = 0b10 or 0b11).

1.7 e6877374 PDM: eQADC: 50% reference channels reads 20 mV low

Description:

The equation given for the definition of the 50% reference channel (channel 42) of the enhanced queued analog to digital converter (eQADC) is not correct. The 50% reference point actually returns approximately 20 mV (after calibration) lower than the expected 50% of difference between the high reference voltage (VRH) and the low reference voltage (VRL).

Workaround:

Do not use the 50% point to calibrate the ADC. Only use the 25% and 75% points for calibration. After calibration, software should expect that the 50% reference reads 20 mV low.

1.8 e12982697 PDM : EQADC: Conversions can be inaccurate when only one ADC is enabled

Description:

the enhanced queued analog to digital converter (eQADC) reference capacitor charging circuit can produce a reference voltage mismatch on the REFBYPC pin(s) when one ADC of an eQADC pair is disabled and the other ADC of the same pair is enabled. This reference voltage mismatch can cause an ADC measurement error.

Workaround:

Always enable both ADCs of a eQADC module pair by setting the ADC0_EN bit the ADC0 control register and ADC1_EN bit in the ADC1 control register before performing calibration of the ADC and before beginning use of either ADC.

1.9 e8773 PS: PMC: SRAM standby power low voltage detect circuit is not accurate

Description:

The power management controller (PMC) SRAM standby voltage low power detect circuit cannot reliably detect the brown-out condition if the standby supply is below 1.0 volts. The Status Register Brown Out Flag (PMC.SR[LVFSTBY]) bit may not be set during a brownout condition of the SRAM standby voltage or may be set even though no data has been lost.

Workaround:

The application software should not rely on the PMC.SR[LVFSTBY] bit to detect corrupted SRAM values.

1.10 e6845184 PDM: BAM: Pull RXD_A high during CAN serial boot mode

Description:

The boot assist module (BAM) disables the internal pull-up resistor on serial port A receive data pin (eSCI RXD_A) during serial boot mode operation. If the pin is not actively driven by system, its input level may drift below low threshold and be recognized as a valid eSCI boot operation thus preventing CAN boot mode selection.

Workaround:

Always drive the eSCI RXD_A pin high during CAN serial boot mode. A external 10 KΩ pull-up resistor can be used for this purpose.

1.11 e6884992 PDM: BAM: Serial download unavailable to last 16 byte (4 words) of system RAM

Description:

When using the BAM serial boot download feature, the BAM initializes an additional 4 32-bit words after the end of the downloaded records. This is done to insure that if the core fetches the last instruction of the downloaded code from the internal SRAM while executing the code, it does not prefetch instructions from memory locations that have not been initialized.

Note: If the download image has the exact same size as the internal SRAM, the 20 byte at the beginning of the SRAM is written with zero value due to incomplete memory decoding.

Workaround:

When using the serial download feature of the BAM, make sure that the maximum address of the downloaded code does not exceed the end address of the SRAM minus 16 bytes or the last address of the Memory Management Unit (MMU) entry minus 16 bytes (for devices with MMU and the SRAM MMU setting less than the full SRAM size), whichever is smaller.

1.12 e6873605 PDM: FMPLL: Loss-of-clock detection may cause unexpected reset

Description:

An unexpected loss-of-clock (LOC) event may occur in the following scenario:

1. The FMPLL is initially powered down in bypass mode.
2. The FMPLL is then powered on (still in bypass mode).
3. The LOCK bit of the SYNSR register is polled to determine when the FMPLL is ready.
4. After the LOCK flag becomes set, the FMPLL is switched to normal mode.
5. Loss-of-clock detection is enabled by setting the LOCEN bit of the enhanced synthesizer control register 2 (ESYNCR2), either before or immediately after switching to normal mode.

The unexpected LOC event activates the backup clock switching feature, causing the reference clock to be selected as the system clock. If LOC reset was also enabled by setting the LOCRE bit in the ESYNCR2 register, a system reset occurs.

The reason for the unexpected LOC event is that the time it takes for the clock quality monitor (CQM) to detect a valid FMPLL clock is typically larger than the time it takes for the FMPLL to lock. Polling the LOC flag does not help because (the way it is defined) it does not flag LOC in bypass mode.

This issue only occurs when the FMPLL is turned off and then on again without going through a reset cycle. Immediately following reset, the issue can not occur because the CQM keeps the part in reset until it detects a valid crystal clock with plenty of time to detect a valid FMPLL clock.

Workaround:

Any time the FMPLL is powered down, wait for 600 µs before activating the loss-of-clock function. If the intent is just to re-program the FMPLL, it is not required to turn it off. FMPLL settings can be changed on the fly, and then the CQM does not indicate loss-of-clock.

1.13 e6183 PS : DSPI: Correct programming of frames in Tx FIFO in master and slave mode

Description:

In master and slave standard DSPI transfer with FIFO enabled and using continuous PSC, if software does not provide correct number of frames in slave mode or does not program the CONT bit correctly, an underflow error is generated.

Workaround:

For correct operation:

In master mode, frames that are to be transmitted under a continuous chip selection assertion (Continuous Selection Format) must have CONT=1, with the exception of last frame, where this bit will be set to 0.

In slave mode, user software must ensure that the slave's Tx FIFO has the desired number of entries required in a continuous PCS transfer and when the last frame is sent out, the slave is de-selected to avoid an underflow error.

1.14 e8845 PS : DSPI: Correct usage for Tx FIFO when continuous clock mode is enabled

Description:

If the FIFO is enabled with continuous serial communication clock (SCK) mode, a change in SCK frequency may occur if the Tx FIFO is not cleared and the clock and transfer attributes register 0 (CTAR0) is not used.

Workaround:

When in continuous SCK mode, always use CTAR0 for the SPI transfer and always clear the TX-FIFO using the clear TX FIFO (CLR_TXF) field of the module configuration register (MCR) before initiating transfer.

1.15 e6850448 PDM: DSPI: Changing CTARs between frames in continuous PCS mode causes error

Description:

Erroneous data could be transmitted if multiple clock and transfer attribute registers (CTAR) are used while using the continuous peripheral chip select mode (DSPIx_PUSHR[CONT = 1]).

The conditions that can generate an error are:

1. If DSPIx_CTARn[CPHA] = 1 and DSPIx_MCR[CONT_SCKE = 0] and DSPIx_CTARn[CPOL, CPHA, PCSSCK or PBR] change between between frames.
2. If DSPIx_CTARn[CPHA] = 0 or DSPIx_MCR[CONT_SCKE = 1] and any bit field of DSPIx_CTARn changes between frames except DSPIx_CTARn[PBR].

Workaround:

When generating DSPI bit frames in continuous PCS mode, adhere to the aforementioned conditions when changing DSPIx_CTARn bit fields between frames.

1.16 e6866802 PDM: DSPI: Using DSPI in DSI mode with MTO may cause data corruption

Description:

Using the DSPI in deserial serial interface (DSI) configuration (DSPIx_MCR[DCONF] = 0b01]) with multiple transfer operation (DSPIx_DSICR[MTOE = 1]) enabled, may cause corruption of data transmitted out on the DSPI master if the clock Phase is set for leading edge capture DSPIx_CTARn[CPHA] = 0.

The first bit shifted out of the master DSPI into the slave DSPI module is corrupted and converts a '0' to read as a '1'.

Workaround:

There are three possible workarounds for this issue.

1. Select CPHA = 1 if suitable for external slave devices.
2. Set first bit to '1', or ignore first bit. This may not be a workable solution if this bit is required.
3. Connect SOUT from the master to SIN of the first slave externally instead of using internal signals. This is achieved by setting the DSPI input select register (SIU_DISR) to set the SINSELx field of the first slave DSPI to '00' and configuring this slave sin pin and master sout pin as DSPI sin/sout functions respectively. This workaround is suitable only if these two signals are available to be connected externally to each other.

1.17 e6847076 PDM : eSCI : DMA stalled after return from stop or doze mode

Description:

If the eSCI module enters stop or doze mode while the transmit DMA is enabled and messages are being transmitted, when the CPU exits stop or doze mode, it is possible that DMA requests will not be generated by the eSCI module.

Workaround:

The application should ensure that the eSCI module is idle before entering the stop mode. The eSCI module is idle when both transmitter and receiver active status bits in the Interrupt flag and status register 1 (eSCI_IFSR1) are not set. The application should not trigger a new transmission on the eSCI module if the application is preparing for the stop mode.

1.18 e6849526 PDM: eSCI : Stop mode not entered in LIN mode

Description:

When the eSCI module is in LIN mode and transmits the header of an LIN RX frame, if the CPU requests stop mode, the eSCI module may not acknowledge the stop mode request and stays in normal operating mode (not in lower power stop mode).

Workaround:

The application should ensure that no LIN transmission is running before it requests stop mode by checking the transmit active and receive active status bits in the eSCI interrupt flag and status register (eSCI_IFSR1[TACT] and eSCI_IFSR1[RACT]).

1.19 e6864631 PDM: eSCI : reads of the SCI data register, which clears the RDRF flag, may cause loss of frame if read occurs during reception of the STOP bit

Description:

A received SCI frame is not written into the SCI data registers and the overrun (OR) flag is not set in the SCI status register 1 (SCISR1), if:

1. The eSCI has received the last data bit of an SCI frame n.
2. The receive data register full (RDRF) flag is still set in the SCISR1 after the reception of SCI frame n-1.
3. During the reception of the STOP bit of frame n the host reads the SCI data registers, and clears the RDRF flag In this case the RDRF flag is erroneously set again by the controller instead of the OR flag. Thus, the host reads the data of frame n-1 a second time, and the data of frame n is lost.

Workaround:

The application should ensure that the data of the foregoing frame is read out from the SCI data registers before the last data bit of the actual frame is received.

1.20 e6870961 PDM: eSCI: LIN bit error indicated at start of transmission after LIN reset

Description:

If the eSCI module is in LIN mode and is transmitting a LIN frame, and the application sets and subsequently clears the LIN reset bit LRES in the LIN control register 1 (ESCI_LCR1), the next LIN frame transmission might incorrectly signal the occurrence of bit errors (ESCI_IFSR1[BERR]) and frame error (ESCI_IFSR1[FE]), and the transmitted frame might be incorrect.

Workaround:

There is no generic work around. The implementation of a suitable workaround is highly dependent on the application and a workaround may not be possible for all applications.

1.21 e6875199 PDM : eSCI : No LIN frame reception after leaving stop mode

Description:

When the eSCI module is in LIN mode and transmits or receives an LIN frame, if the CPU requests stop mode, and the stop mode is left, an subsequent triggered LIN RX frame reception may hang. The module will never assert the eSCI_IFSR2[RXRDY] and eSCI_IFSR2[TXRDY] flags.

Workaround:

The application should ensure that no LIN transmission is running before it requests stop mode by checking the status of the eSCI_IFSR1[TACT] and eSCI_IFSR1[RACT] status flags.

1.22 e6976403 PDM: eSCI: LIN wake-up flag set after aborted LIN frame transmission

Description:

If the eSCI module is transmitting a LIN frame and the application sets and clears the LIN finite state machine resync bit in the LIN control register 1 (eSCI_LCR1[LRES]) to abort the transmission, the LIN wakeup receive flag in the LIN status register may be set (LWAKE = 1).

Workaround:

If the application has triggered LIN protocol engine reset via the eSCI_LCR1[LRES], it should wait for the duration of a frame and clear the eSCI_IFSR2[LWAKE] flag before waiting for a wakeup.

1.23 e6871272 PDM: FlexCAN: Global masks misalignment

Description:

Convention: MSB = 0.

During CAN messages reception by FlexCAN, the RXGMASK (Rx global mask) is used as acceptance mask for most of the Rx message buffers (MB). When the FIFO enable bit in the FlexCAN module configuration register (CANx_MCR[FEN], bit 2) is set, the RXGMASK also applies to most of the elements of the ID filter table.

However there is a misalignment between the position of the ID field in the Rx MB and in RXIDA, RXIDB and RXIDC fields of the ID tables. In fact RXIDA filter in the ID tables is shifted one bit to the left from Rx MBs ID position as shown below:

- Rx MB ID = bits 3 to 31 of ID word corresponding to message ID bits 0 to 28.
- RXIDA = bits 2 to 30 of ID table corresponding to message ID bits 0 to 28 .

Note that the mask bits one-to-one correspondence occurs with the filters bits, not with the incoming message ID bits. This leads the RXGMASK to affect Rx MB and Rx FIFO filtering in different ways.

For example, if the user intends to mask out the bit 24 of the ID filter of message buffers then the RXGMASK is configured as 0xffff_ffef. As result, bit 24 of the ID field of the incoming message is ignored during filtering process for message buffers. This very same configuration of RXGMASK would lead bit 24 of RXIDA to be "do not care" and thus bit 25 of the ID field of the incoming message would be ignored during filtering process for Rx FIFO.

Similarly, both RXIDB and RXIDC filters have multiple misalignments with regards to position of ID field in Rx MBs, which can lead to erroneous masking during filtering process for either Rx FIFO or MBs. RX14MASK (Rx 14 mask) and RX15MASK (Rx 15 mask) have the same structure as the RXGMASK. This includes the misalignment problem between the position of the ID field in the Rx MBs and in RXIDA, RXIDB and RXIDC fields of the ID tables.

Workaround:

Therefore it is recommended that one of the following actions be taken to avoid problems:

- Do not enable the RxFIFO. If CANx_MCR[FEN] = 0 then the Rx FIFO is disabled and thus the masks RXGMASK, RX14MASK and RX15MASK do not affect it.
- Enable Rx individual mask registers. If the backwards compatibility configuration bit in the FlexCAN module configuration register (CANx_MCR[BCC], bit 15) is set then the Rx individual mask registers (RXIMR[0:63]) are enabled and thus the masks RXGMASK, RX14MASK and RX15MASK are not used.
- Do not use masks RXGMASK, RX14MASK and RX15MASK (that is, let them in reset value which is 0xffff_ffff) when CANx_MCR[FEN] = 1 and CANx_MCR[BCC] = 0. In this case, filtering processes for both Rx MBs and Rx FIFO are not affected by those masks.
- Do not configure any MB as Rx (that is, let all MBs as either Tx or inactive) when CANx_MCR[FEN] = 1 and CANx_MCR[BCC] = 0. In this case, the masks RXGMASK, RX14MASK and RX15MASK can be used to affect ID tables without affecting filtering process for Rx MBs.

1.24 e6852073 PDM: FlexCAN: switching CAN protocol interface (CPI) to system clock has very small chance of causing the CPI to enter an indeterminate state

Description:

The reset value for the clock source of the CAN protocol interface (CPI) is the oscillator clock. If the CPI clock source is switched to the system clock while the FlexCAN is not in freeze mode, then the CPI has a very small chance of entering an indeterminate state.

Workaround:

Switch the clock source while the FlexCAN is in a halted state by setting HALT bit in the FlexCAN Module Configuration Register (CANx_MCR[HALT]=1). If the write to the CAN control register to change the clock source (CANx_CR[CLK_SRC]=1) is done in the same oscillator clock period as the write to CANx_MCR[HALT], then chance of the CPI entering an indeterminate state is extremely small.

If those writes are done on different oscillator clock periods, then the corruption is impossible. Even if the writes happen back-to-back, as long as the system clock to oscillator clock frequency ratio is less than three, then the writes will happen on different oscillator clock periods.

1.25 e6887863 PDM: JTAGC: EVTI and RDY require TCK to toggle

Description:

The Nexus/JTAG read/write access control/status register (RWCS) write (to begin a read access) or the write to the read/write access data register (RWD)(to begin a write access) does not actually begin its action until 1 JTAG clock (TCK) after leaving the JTAG update-DR state.

This prevents the access from being performed and therefore will not signal its completion via the READY (RDY) output unless the JTAG controller receives an additional TCK. In addition, EVTI is not latched into the device unless there are clock transitions on TCK.

Workaround:

The tool/debugger must provide at least one TCK clock for the EVTI signal to be recognized by the MCU. When using the RDY signal to indicate the end of a Nexus read/write access, ensure that TCK continues to run for at least 1 TCK after leaving the update-DR state. This can be just a TCK with TMS low while in the run-test/idle state or by continuing with the next Nexus/JTAG command. Expect the affect of EVTI and RDY to be delayed by edges of TCK.

Note: RDY is not available in all packages of all devices.

1.26 e6384 PS: Flash: Erroneous update of the ADR register in case of multiple ECC errors

Description:

An erroneous update of the address register (ADR) occurs whenever there is a sequence of 3 or more events affecting the ADR (ECC single or double bit errors or RWW error) and both the following conditions apply:

- The priorities are ordered in such a way that only the first event should update ADR.
- The last event although it does not update ADR sets the read while write event error (RWE) or the ECC data correction (EDC) in the module configuration register (MCR). For this case the ADR is wrongly updated with the address related to one of the intervening events.

For example, If a sequence of two double-bit ECC errors is followed by a single-bit correction without clearing the ECC event error flag (EER) in the MCR, then the value found in ADR after the single-bit correction event is the one related to the second double-bit error (instead of the first one, as specified).

Workaround:

Always process Flash ECC errors as soon as they are detected. Clear MCR[RWE] at the end of each Flash operation (program, erase, array integrity check, etc.).

Appendix A Defect across silicon version

Table 2. Defect across silicon version

Error	Title	Cut			
		1.0	1.1	2.0	2.1
PDM6860103	eQADC: Analog inputs R, S, T, and U can not be converted.	Yes	Yes	Yes	Yes
PDM6860916	EQADC : 25% calibration channel sampling requires at least 64 sampling cycles	Yes	Yes	Yes	Yes
PDM6877374	EQADC: 50% reference channels reads 20 mv low	Yes	Yes	Yes	Yes
PDM12982697	EQADC: Conversions can be inaccurate when only one ADC is enabled	Yes	Yes	Yes	Yes
PS8773	PMC: SRAM standby power low voltage detect circuit is not accurate		Yes	Yes	Yes
PDM6865285	PMC: The SRAM standby power low voltage detect circuit is not operational	Yes			
PDM6845184	BAM: Pull RXD_A high during CAN serial boot mode	Yes	Yes	Yes	Yes
PDM6884992	BAM: Serial download unavailable to last 16 bytes (4 words) of System RAM	Yes	Yes	Yes	Yes
PDM6873605	FMPLL: Loss-of-clock detection may cause unexpected reset	Yes	Yes	Yes	Yes
PDM6993936	DECFIL: In Stand-alone mode, DMA Input fill overruns the Decimation filter	Yes	Yes		
PS6183	DSPI: Correct programming of frames in Tx FIFO in master and slave mode.	Yes	Yes	Yes	Yes
PS8845	DSPI: Correct usage for Tx FIFO when continuous clock mode is enabled	Yes	Yes	Yes	Yes
PDM6850448	DSPI: Changing CTARs between frames in continuous PCS mode causes error	Yes	Yes	Yes	Yes
PDM6856645	DSPI: DSPI Microsecond Bus Limitations	Yes	Yes		
PDM6858450	DSPI: PCS Continuous Selection Format limitation	Yes	Yes		
PDM6866802	DSPI: Using DSPI in DSI mode with MTO may cause data corruption	Yes	Yes	Yes	Yes
PDM6847076	eSCI : DMA stalled after return from stop or doze mode	Yes	Yes	Yes	Yes
PDM6849526	eSCI : Stop Mode not entered in LIN mode	Yes	Yes	Yes	Yes
PDM6864631	eSCI : reads of the SCI Data Register, which clears the RDRF flag, may cause loss of frame if read occurs during reception of the STOP bit	Yes	Yes	Yes	Yes
PDM6869199	eSCI: LIN Receive Register cleared before data has been read	Yes	Yes		
PDM6870961	eSCI: LIN bit error indicated at start of transmission after LIN reset	Yes	Yes	Yes	Yes

Table 2. Defect across silicon version (continued)

Error	Title	Cut			
		1.0	1.1	2.0	2.1
PDM6875199	eSCI : No LIN frame reception after leaving stop mode	Yes	Yes	Yes	Yes
PDM6976403	eSCI: LIN Wakeup flag set after aborted LIN frame transmission	Yes	Yes	Yes	Yes
PDM6847364	eTPU: MRL Branch Condition at the start of the thread may differ from the actual MRL state if channel runs at full speed	Yes	Yes		
PDM6845641	eTPU: Independent disablement of Matches A and B in Microinstruction Formats* D3 and D7 is not available	Yes	Yes		
PDM6871272	FlexCAN: Global Masks misalignment	Yes	Yes	Yes	Yes
PDM6865252	Pad Ring: Leakage if VDDE is greater than VDD33	Yes	Yes	Yes	
PS5445	Pad Ring: Pins with LVDS input have interaction if current is injected on one pin	Yes	Yes		
PS3692	SIU: GPIO function of the eTPU_A[27]_IRQ[15]_DSPI_C_SOUT_LVDS+_DSPI_B_SOUT_GPIO[141] not available	Yes	Yes		
PS3942	Flash: APC bits are in the PFAPR instead of PFCR1	Yes			
PS4127	SIU: Improper data error exceptions if modules disabled by SIU_HLT	Yes			
PS4374	Pad_Ring: GPIO98 and GPIO99 Output Mode not available	Yes			
PS4375	Pad_Ring: GPIO206 and GPIO207 Output Mode not available in GPIO mode	Yes			
PS4651	eTPU: eTPU cannot write to the Parameter RAM	Yes			
PS4682	eQADC: Internal monitor channels not available	Yes	Yes		
PS4890	Pad Ring: CANTXC and eTPUA25 pins changed to Medium Speed pins	Yes	Yes	Yes	Yes
PS4891	MPC563xM/SPC563M: Mask Number is set to 0x0	Yes			
PS4967	PMC: LVFR and LVFH bit incorrect value after reset	Yes			
PS5539	ECSM: ECC single bit error reporting not available	Yes	Yes		
PS5900	NZ3C2: Nexus trace doesn't work with divide by 2 MCKO	Yes	Yes		
PS5941	Flash: Flash shadow password bus has the 32-bit words swapped	Yes	Yes		
PS5959	PMC: Trimming configuration may change during flash erase/programming	Yes	Yes		
PS6004	JTAGC: JTAG censorship password length is 65-bit	Yes	Yes		
PS6049	Pad Ring: During power-up VFLASH must always be higher than VDD	Yes	Yes		
PS6077	eQADC: Repeatability of ADC measurements reduced	Yes	Yes		

Table 2. Defect across silicon version (continued)

Error	Title	Cut			
		1.0	1.1	2.0	2.1
PS6111	FLASH: Censorship Control Word must be replicated	Yes	Yes	Yes	
PS6223	NPC: High current consumption when running trace in QFP176 package	Yes	Yes		
PS7112	EBI: Calibration bus doesn't work in divide by 2 mode	Yes	Yes		
PS7256	SIU: No data error exceptions if modules disabled by SIU_HLT		Yes		
PS7404	MPC563xM/SPC563M: Mask Number is set to 0x11		Yes		
PS7405	MPC563xM/SPC563M: Mask Number is set to 0x20			Yes	
PDM6865252	Pad Ring: Leakage if VDDE is greater than VDD33				Yes
PS8141	FLASH: PFCR3 is not directly writable	Yes	Yes	Yes	Yes
PS8431	NEXUS: EVTI not functional on QFP176 and BGA208 packages	Yes	Yes	Yes	Yes
PS8715	eQADC: Additional noise seen on conversions	Yes	Yes	Yes	
PS8752PS	Pad Ring: VDDEH1 and VDDEH6 I/O power segments shorted together in BGA208 and CSP496 packages	Yes	Yes	Yes	
PS6988	MPC563xM/SPC563M: MIDR MASKNUM field is set to 0x21				Yes
PS3989	PIT_RTI: RTI timer corruption when debugging	Yes	Yes		
PDM6849634	SIU: NMI interrupt can be disabled	Yes	Yes		
PS5315	FLASH: Incorrect Prefetch at the end of Bank0 boundary (to Bank1)	Yes	Yes		
PDM6887863	JTAGC: EVTI and RDY require TCK to toggle	Yes	Yes	Yes	Yes
PS5007	FLASH: Programming a value over another value may not indicate an error if no bits are being programmed (1-->0)	Yes	Yes	Yes	
PS5008	FLASH: Array Integrity Check does not check the first two double words	Yes	Yes	Yes	
PS5034	FLASH: SLL and HBL not properly initialized	Yes	Yes	Yes	
PS5060	FLASH: Double ECC errors of default Non Volatile image of BIU2 (PFCR2/PFAPR) are not checked	Yes	Yes	Yes	
PS5673	FLASH: Margin Mode is not supported	Yes	Yes	Yes	
PS7587	FLASH: Erase Suspend Latency is out of spec in Flash except Code Flash 0.	Yes	Yes	Yes	
PS6384	Flash: Erroneous update of the ADR register in case of multiple ECC errors	Yes	Yes	Yes	Yes

Table 2. Defect across silicon version (continued)

Error	Title	Cut			
		1.0	1.1	2.0	2.1
PDM6852073	FlexCAN: switching CAN protocol interface (CPI) to system clock has very small chance of causing the CPI to enter an indeterminate state	Yes	Yes	Yes	Yes
PS8650	Flash: Temperature Sensor Information not correctly configured	Yes	Yes	Yes	

Appendix B Further information

B.1 Reference document

1. *SPC563Mxx - 32-bit Power Architecture™ based MCU with up to 1.5 Mbyte Flash and 111 Kbyte RAM memories* (RM0015, Doc ID 14499).
2. *32-bit Power Architecture™ based MCU for automotive powertrain applications* (SPC563Mxx datasheet, Doc ID 14642).

B.2 Acronyms

Table 3. Acronyms

Acronym	Name
ADC	Analog-to-digital converter
AIE	Array integrity enable
APC	Address pipelining control
BAM	Boot assist mode
CQM	Clock quality monitor
CTAR	Transfer attribute registers
SIU_DIRER	DMA/Interrupt request enable register
DMA	Direct memory access
DSI	Deserial serial interface
ECC	Error correction code
ECR	ECC configuration register
ECSM	Error correction status module
EDC	ECC data correction
EIRE	Interrupt request enable
eSCI_LRR	eSCI LIN receive register
ESYNCR2	Enhanced synthesizer control register 2
eQADC	Enhanced queued analog-to-digital converter
FIFO	First in first out
Flash_MCR	Flash module configuration register
JTAGC	JTAG controller
HBL	High address space block locking register
LOC	Loss-of-clock
LVD	Low voltage detector
LVDS	Low voltage differential signalling
LVI	Low voltage inhibit

Table 3. Acronyms (continued)

Acronym	Name
LST	Long sample time
MB	Message buffer
MCKO	Nexus clock
MCR	Module configuration register
MDO	Nexus message data output
ME	Match enabled
MRL	Match recognition latch
PCR	Pad configuration register
PFAPR	Platform Flash access protection register
PFCR	Platform Flash configuration register
PRAM	Parameter RAM
RDRF	Receive data register full
RTI	Real-time interrupt
RWE	Read-while-write event error
RXGMASK	RX global mask
RXIMR	RX individual mask register
SHIFT-DR	Shift data register
SCI	Serial communications interface
SIU_HLT	System integration unit halt register
SLL	Secondary low/mid address space block lock register
TST	Time slot transition
VRH	High reference voltage
VRL	Low reference voltage

Revision history

Table 4. Document revision history

Date	Revision	Changes
17-Feb-2011	1	Initial release.

Please Read Carefully:

Information in this document is provided solely in connection with ST products. STMicroelectronics NV and its subsidiaries (“ST”) reserve the right to make changes, corrections, modifications or improvements, to this document, and the products and services described herein at any time, without notice.

All ST products are sold pursuant to ST’s terms and conditions of sale.

Purchasers are solely responsible for the choice, selection and use of the ST products and services described herein, and ST assumes no liability whatsoever relating to the choice, selection or use of the ST products and services described herein.

No license, express or implied, by estoppel or otherwise, to any intellectual property rights is granted under this document. If any part of this document refers to any third party products or services it shall not be deemed a license grant by ST for the use of such third party products or services, or any intellectual property contained therein or considered as a warranty covering the use in any manner whatsoever of such third party products or services or any intellectual property contained therein.

UNLESS OTHERWISE SET FORTH IN ST’S TERMS AND CONDITIONS OF SALE ST DISCLAIMS ANY EXPRESS OR IMPLIED WARRANTY WITH RESPECT TO THE USE AND/OR SALE OF ST PRODUCTS INCLUDING WITHOUT LIMITATION IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE (AND THEIR EQUIVALENTS UNDER THE LAWS OF ANY JURISDICTION), OR INFRINGEMENT OF ANY PATENT, COPYRIGHT OR OTHER INTELLECTUAL PROPERTY RIGHT.

UNLESS EXPRESSLY APPROVED IN WRITING BY AN AUTHORIZED ST REPRESENTATIVE, ST PRODUCTS ARE NOT RECOMMENDED, AUTHORIZED OR WARRANTED FOR USE IN MILITARY, AIR CRAFT, SPACE, LIFE SAVING, OR LIFE SUSTAINING APPLICATIONS, NOR IN PRODUCTS OR SYSTEMS WHERE FAILURE OR MALFUNCTION MAY RESULT IN PERSONAL INJURY, DEATH, OR SEVERE PROPERTY OR ENVIRONMENTAL DAMAGE. ST PRODUCTS WHICH ARE NOT SPECIFIED AS "AUTOMOTIVE GRADE" MAY ONLY BE USED IN AUTOMOTIVE APPLICATIONS AT USER’S OWN RISK.

Resale of ST products with provisions different from the statements and/or technical features set forth in this document shall immediately void any warranty granted by ST for the ST product or service described herein and shall not create or extend in any manner whatsoever, any liability of ST.

ST and the ST logo are trademarks or registered trademarks of ST in various countries.

Information in this document supersedes and replaces all information previously supplied.

The ST logo is a registered trademark of STMicroelectronics. All other names are the property of their respective owners.

STMicroelectronics group of companies

Australia - Belgium - Brazil - Canada - China - Czech Republic - Finland - France - Germany - Hong Kong - India - Israel - Italy - Japan - Malaysia - Malta - Morocco - Philippines - Singapore - Spain - Sweden - Switzerland - United Kingdom - United States of America

www.st.com