SAMA7G5 Series Silicon Errata and Data Sheet Clarification

SAMA7G5 Series

The SAMA7G5 Series device that you have received conforms functionally to the current Device Data Sheet (DS60001765), except for the anomalies described in this document.

The silicon issues discussed in the following pages are for silicon revisions with the Device and Revision IDs listed in the following table. The silicon issues are summarized in 1. Silicon Issue Summary.

Note: This document summarizes all silicon errata issues from all revisions of silicon, previous as well as current.

Data Sheet clarifications and corrections (if applicable) are located in 22. Data Sheet Clarifications, following the discussion of silicon issues.

The Device and Revision ID values for the SAMA7G54 silicon device are shown in the following table.

Table 1. SAMA7G54 Silicon Device Identification

Part Number	Silicon Revision	Device Identification	
		CHIPID_CIDR[31:0]	CHIPID_EXID[31:0]
SAMA7G54-V/4HB	A1	0x80162111	0x00000000
	A0	0x80162110	

Note: Refer to the “Chip Identifier (CHIPID)” and “Product Identification System” sections in the current device data sheet (DS60001765) for detailed information on chip identification and version for your specific device.

1. Silicon Issue Summary

Table 1-1. Silicon Issue Summary

Module	Item/Feature	Summary	Affected Silicon Revisions		
			A0	A1	
ROM Code	NAND Flash, octal SPI boot	NAND Flash and octal SPI not supported by the ROM code	X	X	
Cortex-A7	PMU interrupt	PMU interrupt rises as soon as enabled	X	X	
	AXIERRIRQ interrupt	AXIERRIRQ interrupt rises as soon as enabled	X	X	
EIC	EIC_WPSR.WPVS	WPVS never rises to 1	X	X	
XDMAC	AXI transactions	Limit to 1 the number of AXI outstanding transactions	X	X	
	XDMAC0 and XDMAC1 channels	Corrupted values returned by some XDMAC0 and XDMAC1 channels	X		
RSTC	RSTC_SR.RSTTYP	RSTTYP shows BACKUP_RST instead of GENERAL_RST	X	X	
RTC	RTC_TSTR0 timestamping error	Incorrect reporting of number of tamper event occurrences	X	X	
OTPC	OTPC number of packets	Number of packets limited to 2	X	X	
	OTPC operating range in Write mode	Write operations restricted over temperature and VDDIN33 ranges	X	X	
	OTPC wrong default configuration	The default configuration cannot be used to access the OTP memory in Write mode	X	X	
PMC	MCKRDY flag error	MCKRDY signal may be stuck at 0	X	X	
PIO	Open drain management	Open drain configuration not allowed when a peripheral is selected	X	X	

...........continued

Module	Item/Feature	Summary	Affected Silicon Revisions		
			A0	A1	
ADC	ADC_EOC_IDR (1)	EOC interrupts enabled spuriously when zeros are written	X	X	
	ADC_EOC_IDR (2)	EOC interrupts not disabled when ones are written	X	X	
	ADC_EOC_IDR (3)	Interrupts pending in ADC_ISR enabled spuriously when ones are written	X	X	
	Temperature sensor (1)	Disabling the temperature sensor via TEMPON is ineffective	X	X	
	Temperature sensor (2)	Enabling ADC channel 30 enables the temperature sensor	X	X	
	Sleep mode	ADC_MR.SLEEP ineffective	X	X	
ISC	DMA descriptors	DMA descriptors writing may occur before last image writing in the memory	X	X	
	Overload	Overload can entail corruption of incoming pixels	X	X	
	Operating frequency	Operating frequency limited to 184 MHz	X		
SSC	L/R data alignment	Channel inversion can occur when SSC in Client mode	X	X	
	TD output	TD output delayed	X	X	
SPDIFRX	L/R data alignment	Right channel sample location error	X	X	
SECUMOD	Dynamic detection intrusion (PIOBU) alarm	Error counter reinitializing issue	X	X	
	Tamper timestamping polarity	Tamper detection signal polarity inverted	X	X	
GMAC	GMAC0 with multiple queues, 10/100 Half Duplex mode	In 10/100 Half Duplex mode, GMAC0 does not work with multiple queues	X	X	
	Specific Address filter registers	Reading of some Specific Address filter registers corrupted on GMAC0 and GMAC1	X		
	Type 1 Screener registers	Reading of some Type 1 Screener registers corrupted on GMAC0 and GMAC1	X		
	Type 2 Screener registers	Reading of some Type 2 Screener registers corrupted on GMAC0 and GMAC1	X		
	GTSUCOMP signal	GTSUCOMP signal failure	X	X	

...........continued

Module	Item/Feature	Summary	Affected Silicon Revisions		
			A0	A1	
FLEXCOM	Write Protection	Write Protection ineffective on FLEXCOM8 to 11	X	X	
SDMMC	Speed mode change, ALL soft reset on-the-fly	On-the-fly actions can lead to SDMMC failure	X	X	
	SDR104, HS200, HS400 modes	Tuning issues, data read errors or clock switching failures	X	X	
	GCLK	GCLK unpredictable behavior when speed mode other than DefaultSpeed or SDR12 used	X	X	
	Mode switching	SDHC blocked after switch from high-speed mode	X	X	
MCAN	Edge filtering	Edge filtering causes mis-synchronization when falling edge at Rx input pin coincides with end of integration phase	X	X	
	MCAN_NBTP.NTSEG2	Configuration of MCAN_NBTP.NTSEG2 = ’0’ not allowed	X	X	
	DAR mode	Retransmission in DAR mode due to lost arbitration at the first two identifier bits	X	X	
	Tx FIFO message	Tx FIFO message sequence inversion	X	X	
	HPM interrupt	Unexpected High Priority Message (HPM) interrupt	X	X	
	Transmitted message	Issue message transmitted with wrong arbitration and control fields	X	X	
	Debug message handling state machine	Debug message handling state machine not reset to Idle when CCCR.INIT is set	X	X	
	Order of messages	Message order inverted when transmitting from dedicated Tx buffers configured with same message ID	X	X	
	Frame transmitted despite cancellation	CAN FD message frame transmitted despite transmit cancellation	X	X	
TC	TC0 Channel 2 registers	TC0 Channel 2 registers reading incorrect	X		
UDPHS	EHCI clock	EHCI spurious stop when Suspend mode occurs on port A	X	X	

2. ROM Code

2.1 NAND Flash and Octal SPI boot not supported

The ROM code does not support booting out of NAND Flash or Octal SPI.

Work Around

None

Affected Silicon Revisions

A0	A1						
X	X						

3. Cortex-A7 Processor (Arm)

3.1 PMU interrupt spurious rise

The Performance Monitoring Unit (PMU) interrupt rises as soon as enabled.

Work Around
Do not enable the PMU interrupt.

Affected Silicon Revisions

A0	A1						
X	X						

3.2 AXIERRIRQ interrupt spurious rise

The AXIERRIRQ interrupt rises as soon as enabled.

Work Around
Do not enable the AXIERRIRQ interrupt.

Affected Silicon Revisions

A0	A1						
X	X						

4. External Interrupt Controller (EIC)

4.1 WPVS bit incorrect behavior

The EIC_WPSR.WPVS bit never rises to 1.

Work Around

None

Affected Silicon Revisions

A0	A1						
X	X						

5. DMA Controller (XDMAC)

5.1 Data corrupted when number of AXI outstanding transactions differs from 1

Data corruption may occur when the number of AXI outstanding transactions differs from 1.

Work Around

Limit to 1 the number of AXI outstanding transactions to access the AHB part in NICGPV. Performance of multichannel DMA transfers to the AHB part (SRAM, EBI, QSPI) is slightly impacted. Performance of transfers to the DDR memory is not affected. Apply the following settings:

```
NICGPV->NICGPV_AMIB[6].NICGPV_AMIB_FN_MOD = 0x3;
NICGPV->NICGPV_AMIB[13].NICGPV_AMIB_FN_MOD = 0x3;
```

Affected Silicon Revisions

A0	A1						
X	X						

5.2 Some XDMAC0 and XDMAC1 channels ineffective

Reading any register of channels 0, 1, 18, 19, 26 and 27 of XDMAC0 and XDMAC1 may return corrupted values.

Work Around

None. Do not use XDMAC0 and XDMAC1 channels 0, 1, 18, 19, 26 and 27.

Affected Silicon Revisions

A0	A1						
X							

6. Reset Controller (RSTC)

6.1 RSTC_SR.RSTTYP not showing GENERAL_RST

In the Status register (RSTC_SR), the RSTTYP field shows BACKUP_RST instead of GENERAL_RST.

Work Around

None

Affected Silicon Revisions

A0	A1						
X	X						

7. Real-time Clock (RTC)

7.1 RTC_TSTR0 timestamping error

RTC_TSTR0.TEVCNT fails to report the correct number of tamper event occurrences.

Work Around

None

Affected Silicon Revisions

A0	A1						
X	X						

8. OTP Controller (OTPC)

8.1 OTPC limited number of packets

The number of OTP packets allowed to be written in the user area, in addition to those necessary to configure the ROM code boot features, is limited to 2. The maximum size of the payload for each packet is 8192 bits.

Work Around
None

Affected Silicon Revisions

A0	A1						
X	X						

8.2 OTPC restricted operating range in Write mode

The write operations in the OTPC cannot be performed over the full temperature and VDDIN33 power supply ranges specified.

Work Around
The write operations in the OTPC are restricted to the following ambient temperature and VDDIN33 power supply ranges:

- T_A = [0°C to 50°C]
- VDDIN33 = [3.15V to 3.6V]

Affected Silicon Revisions

A0	A1						
X	X						

8.3 OTPC wrong default configuration

The default configuration of the OTPC cannot be used to access the OTP memory in Write mode.

Work Around
Prior to any write operation in the OTPC, the OTPC must be configured using the following code. This operation needs to be performed only once before the first write operation and whenever the peripheral reset (signal periph_nreset) is asserted.

```
#define ARRAY_SIZE(a) (sizeof(a) / sizeof((a)[0]))

/*
* writing one word lasts 350us
* the timeout was chosen to be enough for writing 10 words  **/
#define TIMEOUT 500000
#define OTPC_0    (0x1u << 0)
#define OTPC_1    16
#define OTPC_2    (0xffffu << OTPC_1)
#define OTPC_3    (0x4391u << OTPC_1)

static void otp_sama7g5_fixup(void)
{
       static const uint32_t fixup0[4] = {0x04194801, 0x01000000, 0x00000008,
0x00000000};
       static const uint32_t fixup1[4] = {0xfb164801, 0x4c017d12, 0x02120e01,
0x00004000};
       __IO uint32_t *OTPC_4   = (__IO uint32_t *)((uint8_t *)OTPC + 0x090);
       __IO uint32_t *OTPC_5 = (__IO uint32_t *)((uint8_t *)OTPC + 0x0A0);
       __IO uint32_t *OTPC_6 = (__IO uint32_t *)((uint8_t *)OTPC + 0x0B0);
       uint32_t timeout;
       int i;

       timeout = TIMEOUT;
       *OTPC_4 = OTPC_0 | OTPC_3;
       while (!(OTPC->OTPC_SR & OTPC_SR_UNLOCK) && --timeout > 0);

       for (i = 0; i < ARRAY_SIZE(fixup0); i++)
              OTPC_5[i] = fixup0[i];

       for (i = 0; i < ARRAY_SIZE(fixup1); i++)
              OTPC_6[i] = fixup1[i];

       timeout = TIMEOUT;
       *OTPC_4 = OTPC_3;
       while ((OTPC->OTPC_SR & OTPC_SR_UNLOCK) && --timeout > 0); }
```

Affected Silicon Revisions

A0	A1						
X	X						

9. Power Management Controller (PMC)

9.1 MCKRDY flag error

When PMC_CPU_CKR.MDIV is greater than 1, if the PMC_CPU_CKR.CSS field is modified, the MCKRDY signal may be stuck at 0.

Work Around

Use a software timeout of 64 cycles of CPU clock instead of polling the MCKRDY bit.

Affected Silicon Revisions

A0	A1						
X	X						

10. Parallel Input/Output Controller (PIO)

10.1 Open drain management limitation

PIOC does not allow open drain configuration (PIO_CFGRx.OPD=1) when a peripheral is selected (PIO_CFGRx.FUNC different from 0).

As TWI/TWIHS has an internal configuration, it is not affected.

Work Around
None

Affected Silicon Revisions

A0	A1						
X	X						

11. Analog-to-Digital Converter (ADC) Controller

11.1 Spurious effect when zeros written to ADC_EOC_IDR

Writing 0s to ADC_EOC_IDR enables EOC interrupts instead of having no effect.

Work Around
None

Affected Silicon Revisions

A0	A1						
X	X						

11.2 EOC interrupts not disabled when ones written to ADC_EOC_IDR

Writing 1s to ADC_EOC_IDR does not disable the EOC interrupts as it should.

Work Around
None, but the channels can be disabled from ADC_CHDR if unused.

Affected Silicon Revisions

A0	A1						
X	X						

11.3 ADC_IMR interrupts enabled when ones written to ADC_EOC_IDR

Writing 1s to ADC_EOC_IDR enables interrupts in ADC_IMR. If interrupts are pending in ADC_ISR, an interrupt is triggered to the interrupt controller.

Writing to ADC_EOC_IDR is not recommended.

Work Around
Immediately after writing 1s to ADC_EOC_IDR, write 1s to ADC_IDR to disable the unwanted interrupts (store previous value, disable all, reenable previous value).

Affected Silicon Revisions

A0	A1						
X	X						

11.4 Temperature sensor still enabled when stopped without conversion

The temperature sensor remains active even when ADC_TEMPMR.TEMPON is set to 0.

Work Around
To stop the temperature sensor and save its power consumption, perform a conversion prior to writing ADC_TEMPMR.TEMPON=0.

Affected Silicon Revisions

A0	A1						
X	X						

11.5 Temperature sensor spurious activation with CH30

Enabling ADC channel 30 enables the temperature sensor.

Work Around

To stop the temperature sensor and save its power consumption, perform a conversion prior to writing ADC_TEMPMR.TEMPON=0.

Affected Silicon Revisions

A0	A1						
X	X						

11.6 Sleep mode ineffective

Even if Sleep mode is selected by setting ADC_MR.SLEEP, ADC does not enter Sleep mode after a conversion.

Work Around

Reset the ADC Controller with ADC_CR.SWRST.

Affected Silicon Revisions

A0	A1						
X	X						

12. Image Sensor Controller (ISC)

12.1 Sleep mode ineffective

Even if Sleep mode is selected by setting ADC_MR.SLEEP, ADC does not enter Sleep mode after a conversion.

Work Around
Reset the ADC Controller with ADC_CR.SWRST.

Affected Silicon Revisions

A0	A1						
X	X						

12.2 Spurious DMA descriptor writing

Due to AXI transaction reordering, DMA descriptors writing may occur before the last image is written in the memory, even if the DONE flag is set. The user cannot read the DMA descriptors before the full image is written.

Work Around

1. Poll the bit ISC_INTSR.DDONE.
2. Perform an extra read of ISC_INTSR to enable DDR Controller writing.
3. Poll the bit UDDRC_PSTAT.WR_PORT_BUSY_3 with:
   ```
   while (DDRUMCTL_REGS->UDDRC_PSTAT & UDDRC_PSTAT_WR_PORT_BUSY_3(1));
   ```

Once done, the full image is written in the memory and the DMA descriptors can be written.

Affected Silicon Revisions

A0	A1						
X	X						

12.3 Incoming pixels corrupted after overload

In case of overload, incoming pixels may be corrupted.

Work Around
Discard the last frame when the ISC_INTSR.DAOV bit is set.

Affected Silicon Revisions

A0	A1						
X	X						

12.4 Frequency limitation

Timing issues may occur in harsh conditions (low voltage and/or high temperature).

Work Around
In harsh conditions, limit the operating frequency to f_{MCK3} < 184 MHz.

Affected Silicon Revisions

A0	A1						
X							

13. Synchronous Serial Controller (SSC)

13.1 Inverted left/right channels

When the SSC is in Client mode, the TF signal is derived from the codec and not controlled by the SSC. The SSC transmits the data when detecting the falling edge on the TF signal after the SSC transmission is enabled. In some overflow cases, a left/right channel inversion may occur and requires SSC reinitializing.

Work Around
Use the SSC in Host mode so that TF is controlled by the SSC. If the SSC must be used in TF Client mode, start the SSC by writing TXEN and RXEN synchronously with the TXSYN flag rising.

Affected Silicon Revisions

A0	A1						
X	X						

13.2 TD output delay

The TD output is delayed by two or three extra system clock cycles when SSC is configured with the following conditions:

- RCMR.START = Start on falling edge/Start on rising edge/Start on any edge
- RFMR.FSOS = None (input)
- TCMR.START = Receive Start

Work Around
None

Affected Silicon Revisions

A0	A1						
X	X						

14. Sony/Philips Digital Interface Receiver (SPDIFRX)

14.1 SPDIFRX left/right inversion

The SPDIFRX can always provide the left channel sample in the first DMA buffer location when SPDIFRX_MR.SBMODE=1. If the Soft Reset command is applied, the right channel sample may be located in the first DMA buffer location.

Work Around

Perform actions in the following order:

1. Disable the DMA channel.
2. Apply the Soft Reset command.
3. Perform a dummy read in the Mode register (SPDIFRX_MR).
4. Read the SPDIFRX_ISR.RXRDY flag. If set to 1, read the SPDIF Receiver Holding register (SPDIFRX_RHR).

Affected Silicon Revisions

A0	A1						
X	X						

15. Security Module (SECUMOD)

15.1 Dynamic detection intrusion (PIOBU) alarm issue

The error counter fails to reinitialize after a dynamic detection intrusion (PIOBU) alarm.

Work Around

None

Affected Silicon Revisions

A0	A1						
X	X						

15.2 Tamper timestamping polarity error

The tamper detection signal polarity is inverted, with the following consequences:

- Key erasing in TZAEB, AES and TDES if the respective Clear On Tamper features are enabled, with TZAESB_MR.TAMPCLR = 1, AES_MR.TAMPCLR = 1 and TDES_MR.TAMPCLR = 1
- Scrambling key erasing in QSPI0 or QSPI1 if Clear On Tamper is enabled with QSPI0_MR.TAMPCLR = 1 or QSPI1_MR.TAMPCLR = 1
- SHA locking if Tamper Lock is enabled with SHA_MR.TMPLCK = 1. SHA is locked until SHA_CR.UNLOCK is written to 1.

Work Around

Do not enable the following bits:

- TZAESB_MR.TAMPCLR
- AES_MR.TAMPCLR
- TDES_MR.TAMPCLR
- QSPI0_MR.TAMPCLR
- QSPI1_MR.TAMPCLR
- SHA_MR.TMPLCK

Affected Silicon Revisions

A0	A1						
X	X						

16. Gigabit Ethernet MAC (GMAC)

16.1 GMAC0 not functional with multiple queues in 10/100 Half Duplex mode

When operating in 10/100 Half Duplex mode, GMAC0 does not work with multiple queues.

Work Around
Configure the controller for transmission over a single queue.

Affected Silicon Revisions

A0	A1						
X	X						

16.2 Incorrect reading of Specific Address filter registers on GMAC0 and GMAC1

On GMAC0 and GMAC1, Specific Address filter register reading may be corrupted. This may impact in particular the device MAC address.

Affected registers are GMAC0_SABx, GMAC0_SATx, GMAC1_SABx, GMAC1_SATx.

Work Around
Use a shadow copy of the registers and fake the read operation using the shadow copy.

⚠WARNING With the debugger, a value read in a register may be corrupted.

Affected Silicon Revisions

A0	A1						
X							

16.3 Incorrect reading of Type 1 Screener registers on GMAC0 and GMAC1

On GMAC0 and GMAC1, Type 1 Screener register reading may be corrupted. This may impact QoS-based applications using DS/TC fields or TCP/UDP ports.

Affected registers are GMAC0_ST1RPQx and GMAC1_ST1RPQx.

Work Around
Use a shadow copy of the registers and fake the read operation using the shadow copy.

⚠WARNING With the debugger, a value read in a register may be corrupted.

Affected Silicon Revisions

A0	A1						
X							

16.4 Incorrect reading of Type 2 Screener registers on GMAC0 and GMAC1

On GMAC0 and GMAC1, Type 2 Screener register reading may be corrupted. This may impact QoS applications based on source/destination IP, source/destination TCP/UDP, Ethertype or VLAN.

Affected registers are GMAC0_ST2RPQx, GMAC0_ST2ERx, GMAC0_ST2CWxRy, GMAC1_ST2RPQx, GMAC1_ST2ERx, GMAC1_ST2CWxRy.

Work Around

Use a shadow copy of the registers and fake the read operation using the shadow copy.

⚠WARNING With the debugger, a value read in a register may be corrupted.

Affected Silicon Revisions

A0	A1						
X							

16.5 GTSUCOMP feature ineffective

The trigger/capture input B of TC1.TIOB1 is driven internally by the GTSUCOMP signal of the Ethernet MAC (GMAC), but the feature does not work.

Work Around

None

Affected Silicon Revisions

A0	A1						
X	X						

17. Flexible Serial Communication Controller (FLEXCOM)

17.1 Write Protection ineffective on FLEXCOM8 to FLEXCOM11

On FLEXCOM8, FLEXCOM9, FLEXCOM10 and FLEXCOM11, TWIx_WPMR and TWIx_WPSR reading may return corrupted values.

Work Around
None. Do not use the Write Protection feature on FLEXCOM8, FLEXCOM9, FLEXCOM10 and FLEXCOM11.

Affected Silicon Revisions

A0	A1						
X	X						

18. Secure Digital MultiMedia Card Controller (SDMMC)

18.1 SDMMC failure when changing speed mode or performing ALL soft reset on-the-fly

Changing speed mode or performing an ALL soft reset while SDCK is active may lead to block the SDMMC.

Work Around
Stop SDCLK before changing speed mode or performing an ALL soft reset, then re-enable SDCLK by writing SDMMC_CCR.SDCLKEN to 0 before writing SDMMC_MC1R/SDMMC_HC1R/SDMMC_HC2R.

Example:

```
//FIX : stop SDCLK
pSDMMC->SDMMC_CCR = pSDMMC->SDMMC_CCR & ~SDMMC_CCR_SDCLKEN;
switch (speed_mode) {
case DS : pSDMMC->SDMMC_HC1R = pSDMMC->SDMMC_HC1R & ~SDMMC_HC1R_HSEN;
break;
case HS : pSDMMC->SDMMC_HC1R = pSDMMC->SDMMC_HC1R | SDMMC_HC1R_HSEN;
break;
case SDR12 : pSDMMC->SDMMC_HC2R = (pSDMMC->SDMMC_HC2R & ~SDMMC_HC2R_UHSMS_Msk) |
SDMMC_HC2R_UHSMS_SDR12;
break;
case SDR25 : pSDMMC->SDMMC_HC2R = (pSDMMC->SDMMC_HC2R & ~SDMMC_HC2R_UHSMS_Msk) |
SDMMC_HC2R_UHSMS_SDR25;
break;
case SDR50 : pSDMMC->SDMMC_HC2R = (pSDMMC->SDMMC_HC2R & ~SDMMC_HC2R_UHSMS_Msk) |
SDMMC_HC2R_UHSMS_SDR50;
break;
case SDR104 : pSDMMC->SDMMC_HC2R = (pSDMMC->SDMMC_HC2R & ~SDMMC_HC2R_UHSMS_Msk) |
SDMMC_HC2R_UHSMS_SDR104;
break;
case DDR50 : pSDMMC->SDMMC_HC2R = (pSDMMC->SDMMC_HC2R & ~SDMMC_HC2R_UHSMS_Msk) |
SDMMC_HC2R_UHSMS_DDR50;
break;
}
//FIX : re-start SDCLK
pSDMMC->SDMMC_CCR = pSDMMC->SDMMC_CCR | SDMMC_CCR_SDCLKEN;
```

Affected Silicon Revisions

A0	A1						
X	X						

18.2 SDR104, HS200, HS400 modes are not functional

Using mode SDR104, HS200 or HS400 may lead to tuning issues, data read errors or clock switching failures.

Work Around
None

Affected Silicon Revisions

A0	A1						
X	X						

18.3 GCLK cannot be stopped

If a speed mode other than DefaultSpeed or SDR12 is used, GCLK cannot be stopped, leading to unpredictable behavior.

Work Around
Perform an ALL soft reset before any operation to ensure the internal clock DLL can be stopped properly.

Affected Silicon Revisions

A0	A1						
X	X						

18.4 SDHC blocked after switch from high-speed mode

If the current speed mode is Default Speed or SDR12, the SDCLK frequency must be higher than 25 MHz before switching to another mode, otherwise SDHC is blocked.

Work Around
Set the SDCLK frequency to a value higher than 25 MHz before switching to another mode.

Affected Silicon Revisions

A0	A1						
X	X						

19. Controller Area Network (MCAN)

19.1 Edge filtering causes mis-synchronization when falling edge at Rx input pin coincides with end of integration phase

Attention: This erratum is not relevant for CAN 2.0.

When edge filtering is enabled (MCAN_CCCR.EFBI = ’1’) and when the end of the integration phase coincides with a falling edge at the Rx input pin, it may happen that the MCAN synchronizes itself wrongly and does not correctly receive the first bit of the frame. In this case the CRC will detect that the first bit was received incorrectly; it will rate the received FD frame as faulty and an error frame will be sent.

The issue only occurs when there is a falling edge at the Rx input pin (CANRX) within the last time quantum (tq) before the end of the integration phase. The last time quantum of the integration phase is at the sample point of the 11th recessive bit of the integration phase. When the edge filtering is enabled, the bit timing logic of the MCAN sees the Rx input signal delayed by the edge filtering. When the integration phase ends, the edge filtering is automatically disabled. This affects the reset of the FD CRC registers at the beginning of the frame. The Classical CRC registers are not affected, so this issue does not affect the reception of Classical frames.

In CAN communication, the MCAN may enter integrating state (either by resetting MCAN_CCCR.INIT or by protocol exception event) while a frame is active on the bus. In this case the 11 recessive bits are counted between the Acknowledge bit and the following start of frame. All nodes have synchronized at the beginning of the dominant Acknowledge bit. This means that the edge of the following Start-of-Frame bit cannot fall on the sample point, so the issue does not occur. The issue occurs only when the MCAN is, by local errors, mis-synchronized with regard to the other nodes, or not synchronized at all.

Glitch filtering as specified in ISO 11898-1:2015 is fully functional.

Edge filtering was introduced for applications where the data bit time is at least two tq (of the nominal bit time) long. In that case, edge filtering requires at least two consecutive dominant time quanta before the counter counting the 11 recessive bits for idle detection is restarted. This means edge filtering covers the theoretical case of occasional 1-tq-long dominant spikes on the CAN bus that would delay idle detection. Repeated dominant spikes on the CAN bus would disturb all CAN communication, so the filtering to speed up idle detection would not help network performance.

When this rare event occurs, the MCAN sends an error frame and the sender of the affected frame retransmits the frame. When the retransmitted frame is received, the MCAN has left the integration phase and the frame will be received correctly. Edge filtering is only applied during integration phase; it is never used during normal operation. As the integration phase is very short with respect to “active communication time”, the impact on total error frame rate is negligible. The issue has no impact on data integrity.

The MCAN enters integration phase under the following conditions:

- when MCAN_CCCR.INIT is set to ’0’ after start-up
- after a protocol exception event (only when MCAN_CCCR.PXHD = ’0’)

Work around
Disable edge filtering or wait on retransmission in case this rare event happens.

Affected Silicon Revisions

A0	A1						
X	X						

19.2 Configuration of MCAN_NBTP.NTSEG2 = ’0’ not allowed

Attention: This erratum is applicable for CAN 2.0.

When MCAN_NBTP.NTSEG2 is configured to zero (Phase_Seg2(N) = 1), and when there is a pending transmission request, a dominant third bit of Intermission may cause the MCAN to wrongly transmit the first identifier bit dominant instead of recessive, even if this bit was configured as ’1’ in the MCAN’s Tx Buffer Element.

A phase buffer segment 2 of length ’1’ (Phase_Seg2(N) = 1) is not sufficient to switch to the first identifier bit after the sample point in Intermission where the dominant bit was detected.

The CAN protocol according to ISO 11898-1 defines that a dominant third bit of Intermission causes a pending transmission to be started immediately. The received dominant bit is handled as if the MCAN has transmitted a Start-of-Frame (SoF) bit.

The ISO 11898-1 specifies the minimum configuration range for Phase_Seg2(N) to be 2..8 tq. Therefore excluding a Phase_Seg2(N) of ’1’ will not affect MCAN conformance.

Work around
Use the range 1..127 for MCAN_NBTP.NTSEG2 instead of 0..127.

Affected Silicon Revisions

A0	A1						
X	X						

19.3 Retransmission in DAR mode due to lost arbitration at the first two identifier bits

Attention: This erratum is applicable for CAN 2.0.

When the MCAN is configured in DAR mode (MCAN_CCCR.DAR = ’1’) the Automatic Retransmission for transmitted messages that have been disturbed by an error or have lost arbitration is disabled. When the transmission attempt is not successful, the Tx Buffer’s transmission request bit (MCAN_TXBRP.TRPxx) shall be cleared and its Cancellation Finished bit (MCAN_TXBCF.CFxx) shall be set.

When the transmitted message loses arbitration at one of the first two identifier bits, it may happen that instead of the bits of the actually transmitted Tx Buffer, the MCAN_TXBRP.TRPxx and MCAN_TXBCF.CFxx bits of the previously started Tx Buffer (or Tx Buffer 0 if there is no previous transmission attempt) are written (MCAN_TXBRP.TRPxx = ’0’, MCAN_TXBCF.CFxx = ’1’).

If in this case the MCAN_TXBRP.TRPxx bit of the Tx Buffer that lost arbitration at the first two identifier bits has not been cleared, retransmission is attempted.

When the MCAN loses arbitration again at the immediately following retransmission, then actually and previously transmitted Tx Buffers are the same and this Tx Buffer’s MCAN_TXBRP.TRPxx bit is cleared and its MCAN_TXBCF.CFxx bit is set.

Work around
None

Affected Silicon Revisions

A0	A1						
X	X						

19.4 Tx FIFO message sequence inversion

Attention: This erratum is applicable for CAN 2.0.

Assume the case that there are two Tx FIFO messages in the output pipeline of the Tx Message Handler. Transmission of Tx FIFO message 1 is started:

- Position 1: Tx FIFO message 1 (transmission ongoing)
- Position 2: Tx FIFO message 2
- Position 3: --

Now a non-Tx FIFO message with a higher CAN priority is requested. Due to its priority it will be inserted into the output pipeline. The TxMH performs so called "message scans" to keep the output pipeline up to date with the highest priority messages from the Message RAM. After the following two message scans, the output pipeline has the following content:

- Position 1: Tx FIFO message 1 (transmission ongoing)
- Position 2: non-Tx FIFO message with higher CAN priority
- Position 3: Tx FIFO message 2

If the transmission of Tx FIFO message 1 is not successful (lost arbitration or CAN bus error) it is pushed from the output pipeline by the non-Tx FIFO message with higher CAN priority. The following scan re-inserts Tx FIFO message 1 into the output pipeline at position 3:

- Position 1: non-Tx FIFO message with higher CAN priority (transmission ongoing)
- Position 2: Tx FIFO message 2
- Position 3: Tx FIFO message 1

Now Tx FIFO message 2 is in the output pipeline in front of Tx FIFO message 1 and they are transmitted in that order, resulting in a message sequence inversion.

Work around

1. First Work Around

Use two dedicated Tx Buffers, e.g. use Tx Buffers 4 and 5 instead of the Tx FIFO. The pseudo-code below replaces the function that fills the Tx FIFO.

Write message to Tx Buffer 4.

Transmit loop:

- Request Tx Buffer 4 - write MCAN_TXBAR.A4
- Write message to Tx Buffer 5
- Wait until transmission of Tx Buffer 4 completed - MCAN_IR.TC, read MCAN_TXBTO.TO4
- Request Tx Buffer 5 - write MCAN_TXBAR.A5
- Write message to Tx Buffer 4
- Wait until transmission of Tx Buffer 5 is completed - MCAN_IR.TC, read MCAN_TXBTO.TO5

2. Second Work Around

Make sure that only one Tx FIFO element is pending for transmission at any time. The Tx FIFO elements may be filled at any time with messages to be transmitted, but their transmission requests are handled separately. Each time a Tx FIFO transmission has completed and the Tx FIFO gets empty (MCAN_IR.TFE = ’1’), the next Tx FIFO element is requested.

3. Third Work Around

Use only a Tx FIFO. Send the message with the higher priority also from Tx FIFO.

One drawback is that the higher priority message has to wait until the preceding messages in the Tx FIFO have been sent.

Affected Silicon Revisions

A0	A1						
X	X						

19.5 Unexpected High Priority Message (HPM) interrupt

Attention: This erratum is applicable for CAN 2.0.

This issue occurs in two configurations:

Configuration A:

- At least one Standard Message ID Filter Element is configured with Priority flag set (S0.SFEC = "100"/"101"/"110").
- No Extended Message ID Filter Element is configured.
- Non-matching extended frames are accepted (MCAN_GFC.ANFE = "00"/"01").

The HPM Interrupt flag MCAN_IR.HPM is set erroneously on reception of a non-high-priority extended message under the following conditions:

1. A standard HPM frame is received, and accepted by a filter with Priority flag set. Then, Interrupt flag MCAN_IR.HPM is set as expected.
2. Next, an extended frame is received and accepted due to the MCAN_GFC.ANFE configuration. Then, Interrupt flag MCAN_IR.HPM is set erroneously.

Configuration B:

- At least one Extended Message ID Filter Element is configured with Priority flag set (F0.EFEC = "100"/"101"/"110").
- No Standard Message ID Filter Element is configured.
- Non-matching standard frames are accepted (MCAN_GFC.ANFS = "00"/"01").

The HPM Interrupt flag MCAN_IR.HPM is set erroneously on reception of a non-high-priority standard message under the following conditions:

1. An extended HPM frame is received, and accepted by a filter with Priority flag set. Then, Interrupt flag MCAN_IR.HPM is set as expected.
2. Next, a standard frame is received and accepted due to the MCAN_GFC.ANFS configuration. Then, Interrupt flag MCAN_IR.HPM is set erroneously.

Work around

Configuration A:

Set up an Extended Message ID Filter Element with the following configuration:

- F0.EFEC = "001"/"010" - select Rx FIFO for storage of extended frames
- F0.EFID1 = any value - value not relevant as all ID bits are masked out by F1.EFID2
- F1.EFT = "10" - classic filter, F0.EFID1 = filter, F1.EFID2 = mask

- F1.EFID2 = zero - all bits of the received extended ID are masked out

Now, all extended frames are stored in Rx FIFO 0 respectively Rx FIFO 1 depending on the configuration of F0.EFEC.

Configuration B:

Set up a Standard Message ID Filter Element with the following configuration:

- S0.SFEC = "001"/"010" - select Rx FIFO for storage of standard frames
- S0.SFID1 = any value - value not relevant as all ID bits are masked out by S0.SFID2
- S0.SFT = "10" - classic filter, S0.SFID1 = filter, S0.SFID2 = mask
- S0.SFID2 = zero - all bits of the received standard ID are masked out

Now, all standard frames are stored in Rx FIFO 0 respectively Rx FIFO 1 depending on the configuration of S0.SFEC.

Affected Silicon Revisions

A0	A1						
X	X						

19.6 Issue message transmitted with wrong arbitration and control fields

Attention: This erratum is applicable for CAN 2.0.

When the following conditions are met, a message with wrong ID, format, and DLC is transmitted:

- M_CAN is in state "Receiver" (PSR.ACT = "10") and there is no pending transmission.
- A new transmission is requested before the third Intermission bit is reached.
- The CAN bus is sampled dominant at the third Intermission bit which is treated as SoF (see ISO11898-1:2015 Section 10.4.2.2).

Then, it can happen that:

- the Shift register is not loaded with the ID, format and DLC of the requested message,
- the MCAN starts arbitration with wrong ID, format, and DLC on the next bit,
- if the ID wins arbitration, a CAN message with valid CRC is transmitted,
- if this message is acknowledged, the ID stored in the Tx Event FIFO is the ID of the requested Tx message, and not the ID of the message transmitted on the CAN bus, and no error is detected by the transmitting MCAN.

Work around

Request a new transmission only if another transmission is already pending or when the MCAN is not in "Receiver" state (when PSR.ACT ≠ "10").

To avoid activating the transmission request in the critical time window between the sample points of the second and third Intermission bits, the application software can evaluate the Rx Interrupt flags IR.DRX, IR.RF0N and IR.RF1N, which are set at the last EoF bit when a received and accepted message becomes valid.

The last EoF bit is followed by three Intermission bits. Therefore, the critical time window has safely terminated three bit times after the Rx interrupt. Now a transmission can be requested by writing to TXBAR.

After the interrupt, the application has to take care that the transmission request for the CAN Protocol Controller is activated before the critical window of the following reception is reached.

A checksum covering the arbitration and control fields can be added to the data field of the message to be transmitted, to detect frames transmitted with wrong arbitration and control fields.

Affected Silicon Revisions

A0	A1						
X	X						

19.7 Debug message handling state machine not reset to Idle when CCCR.INIT is set

When the host sets the MCAN_CCCR.INIT bit through the MCAN_CCCRn register, or when the CAN enters Bus Off state, the debug message handling state machine stays in its current state instead of resetting to Idle state. Setting MCAN_CCCR.CCE does not change MCAN_RXF1S.DMS.

Work Around
If the debug message handling state machine stopped while MCAN_RXF1S.DMS="01" or MCAN_RXF1S.DMS="10", it can be reset to Idle state by hardware reset or by reception of debug messages after MCAN_CCCR.INIT is reset to zero.

Affected Silicon Revisions

A0	A1						
X	X						

19.8 Message order inversion when transmitting from dedicated Tx buffers configured with same message ID

When several Tx buffers are configured with the same message ID, transmission of these Tx buffers is requested sequentially with a delay between the individual Tx requests. They are transmitted in ascending order of their numbers, so the Tx buffer with the lowest number and pending Tx request is transmitted first.

However, depending on the delay between the individual Tx requests, it can happen that the lowest Tx buffer number is not transmitted first and that the message order is inverted.

Work Around
First write the group of Tx messages having the same message ID to the message RAM, and then request transmission of all these messages concurrently by a single write access to MCAN_TXBAR.

Affected Silicon Revisions

A0	A1						
X	X						

19.9 Frame transmitted despite confirmed transmit cancellation for CAN-FD messages with more than 8 data bytes

Attention: This erratum is not relevant for CAN 2.0.

In case the transmission of Tx Buffer nn was not successful and is restarted immediately afterwards by automatic retransmission, and the software requests a Tx cancellation for this Tx Buffer by setting the cancellation request bit MCAN_TXBCR.CRnn during transmission of the first 4 identifier bits, a successful cancellation is incorrectly signalled by setting MCAN_TXBCF.CFnn = ’1’ and by clearing MCAN_TXBRP.TRPnn. In addition, the respective Transmission Occurred bit remains zero (MCAN_TXBTO.TOnn = ’0’), incorrectly indicating that the frame was not transmitted on the bus.

Other than signalled by MCAN_TXBCF.CFnn and MCAN_TXBTO.TOnn, the transmission continues until the complete frame has been sent on the CAN bus. If the transmission is successful, MCAN_TXBTO.TOnn will be set.

If in this case new data is written to Tx Buffer nn while the transmission is still ongoing, a frame with inconsistent data may appear on the bus.

This problem is limited to the case of transmit cancellation of CAN-FD messages with more than 8 data bytes while automatic retransmission is enabled (MCAN_CCCR.DAR = ’0’).

Transmit cancellation of Classical CAN messages and CAN-FD messages with up to 8 data bytes is not affected.

CAN 2.0 operation is not impacted.

Work around

Do not use transmit cancellation for CAN-FD messages with more than 8 data bytes.

Alternatively, wait for the duration of the expected transmission time of the cancelled Tx Buffer before writing new data to that Tx Buffer.

Affected Silicon Revisions

A0	A1						
X	X						

20. Timer Counter (TC)

20.1 TC0 Channel 2 registers incorrect reading

Reading the following registers (TC0 Channel 2 registers) may return corrupted values: TC0_CCR2, TC0_CMR2, TC0_SMMR2, TC0_RAB2, TC0_CV2, TC0_RA2, TC0_RB2, TC0_RC2, TC0_SR2, TC0_IER2.

Work Around
None. Do not use TC0 Channel 2.

Affected Silicon Revisions

A0	A1						
X							

21. USB Device High Speed Port (UDPHS)

21.1 EHCI spurious stop when Suspend mode occurs on port A

If port A enters Suspend mode (via the USB device controller or the USB host controller), the EHCI clock stops, which can block the EHCI if it uses the other ports (B/C). This occurs after about 2 µs (≈ 120 clock cycles).

Work Around

Set the SFR_EHCIOHCI.PHYCLK bit so that the EHCI clock remains activated in Suspend mode, with no power saving.

Affected Silicon Revisions

A0	A1						
X	X						

22. Data Sheet Clarifications

22.1 Electrical Characteristics

22.1.1 Temperature sensor wrong parameter value

In table “Temperature Sensor Characteristics”, the minimum value for the t_{TH} parameter (V_{TEMP} or V_{BG} track-and-hold time) should read 1 µs and not 20 µs.

23. Revision History

23.1 DS80001016B - 08/2022

Added:

- 11.4. Temperature sensor still enabled when stopped without conversion
- 11.5. Temperature sensor spurious activation with CH30
- 11.6. Sleep mode ineffective
- 19.9. Frame transmitted despite confirmed transmit cancellation for CAN-FD messages with more than 8 data bytes
- 22.1.1. Temperature sensor wrong parameter value

Updated 12.2. Spurious DMA descriptor writing

Removed ACC erratum “WPVS bit incorrect behavior”

23.2 DS80001016A - 03/2022

First issue.

Microchip Information

The Microchip Website

Microchip provides online support via our website at www.microchip.com/. This website is used to make files and information easily available to customers. Some of the content available includes:

- **Product Support** – Data sheets and errata, application notes and sample programs, design resources, user's guides and hardware support documents, latest software releases and archived software
- **General Technical Support** – Frequently Asked Questions (FAQs), technical support requests, online discussion groups, Microchip design partner program member listing
- **Business of Microchip** – Product selector and ordering guides, latest Microchip press releases, listing of seminars and events, listings of Microchip sales offices, distributors and factory representatives

Product Change Notification Service

Microchip's product change notification service helps keep customers current on Microchip products. Subscribers will receive email notification whenever there are changes, updates, revisions or errata related to a specified product family or development tool of interest.

To register, go to www.microchip.com/pcn and follow the registration instructions.

Customer Support

Users of Microchip products can receive assistance through several channels:

- Distributor or Representative
- Local Sales Office
- Embedded Solutions Engineer (ESE)
- Technical Support

Customers should contact their distributor, representative or ESE for support. Local sales offices are also available to help customers. A listing of sales offices and locations is included in this document.

Technical support is available through the website at: www.microchip.com/support

Microchip Devices Code Protection Feature

Note the following details of the code protection feature on Microchip products:

- Microchip products meet the specifications contained in their particular Microchip Data Sheet.
- Microchip believes that its family of products is secure when used in the intended manner, within operating specifications, and under normal conditions.
- Microchip values and aggressively protects its intellectual property rights. Attempts to breach the code protection features of Microchip product is strictly prohibited and may violate the Digital Millennium Copyright Act.
- Neither Microchip nor any other semiconductor manufacturer can guarantee the security of its code. Code protection does not mean that we are guaranteeing the product is "unbreakable". Code protection is constantly evolving. Microchip is committed to continuously improving the code protection features of our products.

Legal Notice

This publication and the information herein may be used only with Microchip products, including to design, test, and integrate Microchip products with your application. Use of this information in any other manner violates these terms. Information regarding device applications is provided only for your convenience and may be superseded by updates. It is your responsibility to ensure that your application meets with your specifications. Contact your local Microchip sales office for additional support or, obtain additional support at www.microchip.com/en-us/support/design-help/client-support-services.

THIS INFORMATION IS PROVIDED BY MICROCHIP "AS IS". MICROCHIP MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND WHETHER EXPRESS OR IMPLIED, WRITTEN OR ORAL, STATUTORY OR OTHERWISE, RELATED TO THE INFORMATION INCLUDING BUT NOT LIMITED TO ANY IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY, AND FITNESS FOR A PARTICULAR PURPOSE, OR WARRANTIES RELATED TO ITS CONDITION, QUALITY, OR PERFORMANCE.

IN NO EVENT WILL MICROCHIP BE LIABLE FOR ANY INDIRECT, SPECIAL, PUNITIVE, INCIDENTAL, OR CONSEQUENTIAL LOSS, DAMAGE, COST, OR EXPENSE OF ANY KIND WHATSOEVER RELATED TO THE INFORMATION OR ITS USE, HOWEVER CAUSED, EVEN IF MICROCHIP HAS BEEN ADVISED OF THE POSSIBILITY OR THE DAMAGES ARE FORESEEABLE. TO THE FULLEST EXTENT ALLOWED BY LAW, MICROCHIP'S TOTAL LIABILITY ON ALL CLAIMS IN ANY WAY RELATED TO THE INFORMATION OR ITS USE WILL NOT EXCEED THE AMOUNT OF FEES, IF ANY, THAT YOU HAVE PAID DIRECTLY TO MICROCHIP FOR THE INFORMATION.

Use of Microchip devices in life support and/or safety applications is entirely at the buyer's risk, and the buyer agrees to defend, indemnify and hold harmless Microchip from any and all damages, claims, suits, or expenses resulting from such use. No licenses are conveyed, implicitly or otherwise, under any Microchip intellectual property rights unless otherwise stated.

Trademarks

The Microchip name and logo, the Microchip logo, Adaptec, AVR, AVR logo, AVR Freaks, BesTime, BitCloud, CryptoMemory, CryptoRF, dsPIC, flexPWR, HELDO, IGLOO, JukeBlox, KeeLoq, Kleer, LANCheck, LinkMD, maXStylus, maXTouch, MediaLB, megaAVR, Microsemi, Microsemi logo, MOST, MOST logo, MPLAB, OptoLyzer, PIC, picoPower, PICSTART, PIC32 logo, PolarFire, Prochip Designer, QTouch, SAM-BA, SenGenuity, SpyNIC, SST, SST Logo, SuperFlash, Symmetricom, SyncServer, Tachyon, TimeSource, tinyAVR, UNI/O, Vectron, and XMEGA are registered trademarks of Microchip Technology Incorporated in the U.S.A. and other countries.

AgileSwitch, APT, ClockWorks, The Embedded Control Solutions Company, EtherSynch, Flashtec, Hyper Speed Control, HyperLight Load, Libero, motorBench, mTouch, Powermite 3, Precision Edge, ProASIC, ProASIC Plus, ProASIC Plus logo, Quiet- Wire, SmartFusion, SyncWorld, Temux, TimeCesium, TimeHub, TimePictra, TimeProvider, TrueTime, and ZL are registered trademarks of Microchip Technology Incorporated in the U.S.A.

Adjacent Key Suppression, AKS, Analog-for-the-Digital Age, Any Capacitor, AnyIn, AnyOut, Augmented Switching, BlueSky, BodyCom, Clockstudio, CodeGuard, CryptoAuthentication, CryptoAutomotive, CryptoCompanion, CryptoController, dsPICDEM, dsPICDEM.net, Dynamic Average Matching, DAM, ECAN, Espresso T1S, EtherGREEN, GridTime, IdealBridge, In-Circuit Serial Programming, ICSP, INICnet, Intelligent Paralleling, IntelliMOS, Inter-Chip Connectivity, JitterBlocker, Knob-on-Display, KoD, maxCrypto, maxView, memBrain, Mindi, MiWi, MPASM, MPF, MPLAB Certified logo, MPLIB, MPLINK, MultiTRAK, NetDetach, Omniscient Code Generation, PICDEM, PICDEM.net, PICkit, PICtail, PowerSmart, PureSilicon, QMatrix, REAL ICE, Ripple Blocker, RTAX, RTG4, SAM-ICE, Serial Quad I/O, simpleMAP, SimpliPHY, SmartBuffer, SmartHLS, SMART-I.S., storClad, SQI, SuperSwitcher, SuperSwitcher II, Switchtec, SynchroPHY, Total Endurance, Trusted Time, TSHARC, USBCheck, VariSense, VectorBlox, VeriPHY, ViewSpan, WiperLock, XpressConnect, and ZENA are trademarks of Microchip Technology Incorporated in the U.S.A. and other countries.

SQTP is a service mark of Microchip Technology Incorporated in the U.S.A.

The Adaptec logo, Frequency on Demand, Silicon Storage Technology, and Symmcom are registered trademarks of Microchip Technology Inc. in other countries.

GestIC is a registered trademark of Microchip Technology Germany II GmbH & Co. KG, a subsidiary of Microchip Technology Inc., in other countries.

All other trademarks mentioned herein are property of their respective companies.

ISBN: 978-1-6683-0987-2

Quality Management System

For information regarding Microchip’s Quality Management Systems, please visit www.microchip.com/quality.

Worldwide Sales and Service

AMERICAS

Corporate Office
2355 West Chandler Blvd.
Chandler, AZ 85224-6199
Tel: 480-792-7200
Fax: 480-792-7277
Technical Support:
www.microchip.com/support
Web Address:
www.microchip.com

Atlanta
Duluth, GA
Tel: 678-957-9614
Fax: 678-957-1455

Austin, TX
Tel: 512-257-3370

Boston
Westborough, MA
Tel: 774-760-0087
Fax: 774-760-0088

Chicago
Itasca, IL
Tel: 630-285-0071
Fax: 630-285-0075

Dallas
Addison, TX
Tel: 972-818-7423
Fax: 972-818-2924

Detroit
Novi, MI
Tel: 248-848-4000

Houston, TX
Tel: 281-894-5983

Indianapolis
Noblesville, IN
Tel: 317-773-8323
Fax: 317-773-5453
Tel: 317-536-2380

Los Angeles
Mission Viejo, CA
Tel: 949-462-9523
Fax: 949-462-9608
Tel: 951-273-7800

Raleigh, NC
Tel: 919-844-7510

New York, NY
Tel: 631-435-6000

San Jose, CA
Tel: 408-735-9110
Tel: 408-436-4270

Canada - Toronto
Tel: 905-695-1980
Fax: 905-695-2078

ASIA/PACIFIC

Australia - Sydney
Tel: 61-2-9868-6733

China - Beijing
Tel: 86-10-8569-7000

China - Chengdu
Tel: 86-28-8665-5511

China - Chongqing
Tel: 86-23-8980-9588

China - Dongguan
Tel: 86-769-8702-9880

China - Guangzhou
Tel: 86-20-8755-8029

China - Hangzhou
Tel: 86-571-8792-8115

China - Hong Kong SAR
Tel: 852-2943-5100

China - Nanjing
Tel: 86-25-8473-2460

China - Qingdao
Tel: 86-532-8502-7355

China - Shanghai
Tel: 86-21-3326-8000

China - Shenyang
Tel: 86-24-2334-2829

China - Shenzhen
Tel: 86-755-8864-2200

China - Suzhou
Tel: 86-186-6233-1526

China - Wuhan
Tel: 86-27-5980-5300

China - Xian
Tel: 86-29-8833-7252

China - Xiamen
Tel: 86-592-2388138

China - Zhuhai
Tel: 86-756-3210040

ASIA/PACIFIC

India - Bangalore
Tel: 91-80-3090-4444

India - New Delhi
Tel: 91-11-4160-8631

India - Pune
Tel: 91-20-4121-0141

Japan - Osaka
Tel: 81-6-6152-7160

Japan - Tokyo
Tel: 81-3-6880- 3770

Korea - Daegu
Tel: 82-53-744-4301

Korea - Seoul
Tel: 82-2-554-7200

Malaysia - Kuala Lumpur
Tel: 60-3-7651-7906

Malaysia - Penang
Tel: 60-4-227-8870

Philippines - Manila
Tel: 63-2-634-9065

Singapore
Tel: 65-6334-8870

Taiwan - Hsin Chu
Tel: 886-3-577-8366

Taiwan - Kaohsiung
Tel: 886-7-213-7830

Taiwan - Taipei
Tel: 886-2-2508-8600

Thailand - Bangkok
Tel: 66-2-694-1351

Vietnam - Ho Chi Minh
Tel: 84-28-5448-2100

EUROPE

Austria - Wels
Tel: 43-7242-2244-39
Fax: 43-7242-2244-393

Denmark - Copenhagen
Tel: 45-4485-5910
Fax: 45-4485-2829

Finland - Espoo
Tel: 358-9-4520-820

France - Paris
Tel: 33-1-69-53-63-20
Fax: 33-1-69-30-90-79

Germany - Garching
Tel: 49-8931-9700

Germany - Haan
Tel: 49-2129-3766400

Germany - Heilbronn
Tel: 49-7131-72400

Germany - Karlsruhe
Tel: 49-721-625370

Germany - Munich
Tel: 49-89-627-144-0
Fax: 49-89-627-144-44

Germany - Rosenheim
Tel: 49-8031-354-560

Israel - Ra’anana
Tel: 972-9-744-7705

Italy - Milan
Tel: 39-0331-742611
Fax: 39-0331-466781

Italy - Padova
Tel: 39-049-7625286

Netherlands - Drunen
Tel: 31-416-690399
Fax: 31-416-690340

Norway - Trondheim
Tel: 47-72884388

Poland - Warsaw
Tel: 48-22-3325737

Romania - Bucharest
Tel: 40-21-407-87-50

Spain - Madrid
Tel: 34-91-708-08-90
Fax: 34-91-708-08-91

Sweden - Gothenberg
Tel: 46-31-704-60-40

Sweden - Stockholm
Tel: 46-8-5090-4654

UK - Wokingham
Tel: 44-118-921-5800
Fax: 44-118-921-5820