Specification Update

1. Revision History

Date of revision	Version	Description
10 July 1997	A	Creation
20 August 1997	B	New version of the device (Date Code 9727)
12 February 1998	C	New errata discovered
27 April 1998	D	C251G1–11 errata clarification and C251G1–19 errata correction and add of step A core errata

World Wide Web: http://www.temic.de/semi/

2. Preface

This document is an update to the specifications contained in Table 1.

TSC80251G1 products implement a C251 architecture compliant to INTEL's MCS®251. Hence some of INTEL's specification update apply to TEMIC products. This includes INTEL's core stepping as outlined in Table 2.

***Note:** This document only applies to TEMIC's products and does not imply any product from INTEL.*

Table 1 Affected Documents/Related Documents

Title	Reference
TSC80251G1 Design Guide 1996	Rev. A – 18 December 1996
TSC80251 Programmer's Guide 1996	Rev. B – 23 October 1996
TSC80251G1 EPROM Programming	Rev. B – 27 May 1997

Table 2 TEMIC Products to INTEL's Core Version Reference

TEMIC Products	INTEL's Core Version			
	A	B	C	D
TSC87251G1–xyyyy	X			

Notes:
x provides factory programmed configuration option when needed.
yyyy provides speed, temperature range, packaging and conditioning options.
zzzz provides the customer code for MaskROM.
Please refer to Ordering Information in the Design Guide for full information on the options.

3. Nomenclature

Errata are design defects or errors. These may cause the published (component, board, system) behavior to deviate from published specifications. Hardware and software designed to be used with any component, board, and system must consider all errata documented.

Documentation Changes include typos, errors, or omissions from the current published specifications. These changes will be incorporated in any new release of the specifications (See Table 1).

Note:
Errata removed from the specification update are archived and available upon request.

Table 3 Codes Used in Summary Table

Page	
(Page)	Page location of item in this document.
Status	
Doc	Document change or update will be implemented.
Fix	This erratum is intended to be fixed in a future version of the component.
No Fix	There are no plans to fix this erratum.
Eval	Plans to fix this erratum are under evaluation.

4. Summary Tables of Changes

The following tables indicate the errata or documentation changes which apply to TSC80251G1 derivatives. TEMIC may fix some of the errata in a future version of the component, and account for the other outstanding issues through documentation or specification changes as noted.

Table 4 Errata

TEMIC Reference	INTEL Reference	Page	Status	Errata
C251G1–01	9600005	3	Fix	Interrupt when CPU is executing user code not in FF: region
C251G1–04	–	4	Fix	SSLC serial clock rate in µwire/SPI mode when Timer1 is selected
C251G1–06	–	5	Eval	P1F read–modify–write
C251G1–11	–	6	Fix	Timers 0 and 1 autoreload
C251G1–12	–	7	Fix	Stop PCA counter when CL overflows
C251G1–13	–	8	Fix	Timer 2 is in clock–out mode with FFh in RCAP2L
C251G1–14	–	9	Fix	UART in mode 2 and 3 with framing error
C251G1–15	9600007	10	Fix	Watchdog timer in idle mode
C251G1–18	–	11	Fix	JBC instruction
C251G1–19	–	12	Fix	Watchdog timer refresh
C251G1–20	9600001	13	Fix	Negative flag
C251G1–21	9600002	14	Fix	WSB – wait state for memory region 01:
C251G1–22	9600003	15	Fix	EJMP at upper boundary of any 64–Kbyte region of memory
C251G1–23	9600004	16	Fix	Short jumps from memory region FF: to FE:

Table 5 Documentation Changes

TEMIC Reference	INTEL Reference	Page	Status	Documentation Changes
	–	6	Doc	EPROM Programming Revision B
	–	6	Doc	T_{RLDR} Values

5. Errata

Reference	Erratum
C251G1–01	**Interrupt when CPU is executing User Code not in FF: region**

Problem

The reset vector and interrupt vectors are defined at the FF: region. If an interrupt occurs, the CPU is not able to jump to the interrupt vector when the CPU is executing the user code at regions other than FF:.

Example

If the user code size is 128–Kbyte, the first 64–Kbyte of the user code is located at FF: region and the other 64–Kbyte of the user code is located at FE: region. The reset vector and the interrupt vector for the external interrupt 0 (INT0#) are defined at the FF: region as shown below.

```
                                ORG   FF:0000h
FF0000 02 0100                  LJMP  MAIN
                                ORG   FF:0003h
FF0003 02 0300                  LJMP  EXT_INT0
                                ORG   FF:0100h
FF0100 12 0200   MAIN:          LCALL INITIALIZE_INT0
                                ORG   FF:0300h
FF0000 02 0100   EXT_INT0:      PUSH  PSW
                                ...
                                RETI
```

When the CPU is executing the user code from the FE: region, if an external interrupt (INT0#) occurs, the CPU is not able to jump to interrupt vector (EXT_INT0) as defined at FF: region. It is jumping to the FE: region instead.

Implication

This error affects additional interrupt vectors to be initialized at other memory regions.

Workaround

Bit 4 (INTR bit) of configuration byte (CONFIG1) must be set to 1. All needed interrupt vectors must be defined at other regions (other than FF: region) as well. For an example, if the user code size is 128-Kbyte and the first 64-Kbyte of the user code and the interrupt vector for external interrupt 0 is defined at FF: region.

```
ORG   FF:0003h
LJMP  EXT_INT0
```

Then there are two possible ways of locating the other 64-Kbyte user code and the workaround:

- If the other 64-Kbyte user code is located at FE: region, the following instructions must be added to the user code at FE: region:.

```
ORG   FE:0003h
EJMP  EXT_INT0
```

- If the other 64-Kbyte user code is located at 00: region, the following instructions must be added to the user code at 00: region.

```
ORG   00:0003h
EJMP  EXT_INT0
```

Affected Products

All TSC80251G1 Step A derivatives are affected.

Reference	Erratum
C251G1–04	**SSLC serial clock rate in µwire/SPI mode when Timer1 is selected**

Problem

Timer1 may be selected to produce low serial clock speed on the Synchronous Serial Link Controller (SSLC) in µwire/ SPI mode. The bit rate (BR) is specified by the following formula: BR = F_{OSC}/(96.(256–TH1)). However, the actual bit rate is BR = F_{OSC}/(24.(256–TH1)).

Implication

When Timer1 is selected to produce the serial clock rate, the bit rate in µwire/SPI mode is 4 times greater than specified and than in I2C mode.

Workaround

If the actual bit rate value is critical, do not select Timer1 to generate the serial clock when using the TSC80251G1 derivatives for upward compatibility with the new versions of the product. The dedicated synchronous serial bit rate (SSBR) generator should be used instead. It allows to transmit data down to 15.6 kHz when the on–chip oscillator frequency is 16 MHz.

If the actual bit rate value is not critical or if the software may be updated when using new versions of the device, Timer1 may be selected. It provides the lowest speed, down to 2.4 kHz when the on–chip oscillator frequency is 16 MHz.

Affected Products

All TSC80251G1 Step A derivatives are affected.

Reference	Erratum
C251G1–06	**P1F read–modify–write**

Problem

Port 1 flag (P1F) register contains eight flags (P1F.x, x=0, 1, 2, 3, 4, 5, 6, 7) which can be set by hardware according to a real–time event and which must be reset by software. When a zero is written to one of these bits after it has been set by hardware, different cases occur:

- A simple Write instruction has been used, then the flag must be cleared.
- A Read–Modify–Write instruction (See Table 6) has been used and the event has occured before the read operation, then the flag must be cleared.
- A Read–Modify–Write instruction (See Table 6) has been used and the event has occured after the read operation and before the write operation, then the flag **must not** be cleared.

For some devices the flag is always cleared, even in the last case while it should not.

Implication

Some events may got lost if more than one Port line flag is used on the keyboard interface.

Workaround

As Read–Modify–Write instructions execute in less than 5 states, no event will be lost if the pulses to be detect are asserted for at least 5 states. The original specification was a minimal width of 2 states.
Beware, the interrupt is issued as long as one input stays to the programmed level.

Please ask for TEMIC support if you need help to implement edge detection with the keyboard interface.

Affected Products

All TSC80251G1 Step A derivatives are affected.

Table 6 Read–Modify–Write Instructions

Instruction	Description	Example
ANL	logical AND	ANL CCON, A
ORL	logical OR	ORL CCON, A
XRL	logical XOR	XRL CCON, A
JBC	jump if bit = 1 and clear bit	JBC CCF1, LABEL
CPL	complement bit	CPL CCF0
INC	increment	INC CCON
DEC	decrement	DEC CCON
DJNZ	decrement and jump if not zero	DJNZ CCON, LABEL
MOV SFR.x, C	move carry bit to bit x of SFR	MOV CR, C
CLR SFR.x	clear bit x of SFR	CLR CF
SET SFR.x	set bit x of SFR	SET CCF3

Reference	Erratum
C251G1–11	**Timers 0 and 1 Autoreload**

Problem

The overflow flags TF0 and TF1 are not set when timer 0 and timer 1 are in Autoreload mode (mode 2) with reload value FFh or when a software reload is done with value FFFFh.

Implication

Only the flag is affected, the count and reload operations will run normally.

Workaround

Do not use autoreload mode with TH0 or TH1 set to FFh. Do not reload timer 0 or timer 1 with FFFFh value.

Affected Products

All TSC80251G1 Step A derivatives are affected.

Reference	Erratum
C251G1–12	**Stop PCA counter when CL overflows**

Problem

If the PCA Timer is stopped when CL (Counter Low register) overflows, CH (Counter High register) is not incremented.

Implication

If the PCA Timer is resumed, the carry out from CL is lost, so CH missed one increment.

Workaround

The only way to workaround this trouble is to check that CL will not overflow while stopping the PCA Timer.

Please ask for TEMIC support if you need help to implement such a workaround in your application.

Affected Products

All TSC80251G1 Step A derivatives are affected.

Reference	Erratum
C251G1–13	**Timer 2 is in clock–out mode with FFh in RCAP2L**

Problem

Depending on RCAP2L value, the Timer 2 in clock–out mode works or not.

RCAP2H	RCAP2L	Clock–Out Mode
FFh	FFh	Working
FFh	xx	Working
xx	xx	Working
xx	FFh	Not Working

xx = [00h ... FEh]

Workaround

The only way to workaround this trouble is to change RCAP2L value in FEh when the register contains FFh.

Please ask for TEMIC support if you need help to implement such a workaround in your application.

Affected Products

All TSC80251G1 Step A derivatives are affected.

Reference	Erratum
C251G1–14	**UART in mode 2 and 3 with Framing Error**

Problem

When in 9–bit format (mode 2 or 3) and framing error detection is set, FE flags does not test STOP bit but 9nth bit.

Implication

This error affects the usage of framing error recognition when in mode 2 or 3.

Workaround

Do not use framing error detection in 9–bit format.

Affected Products

All TSC80251G1 Step A derivatives are affected.

Reference	Erratum
C251G1–15	**Watchdog timer in idle mode**

Problem

During the hardware or software Watchdog timer operation in the Idle mode, it should reset the controller and wake up from the idle mode after the counter overflow. The controller fails to perform this operation.

Implication

This error affects the usage of hardware and software watchdog timer during the idle mode.

Workaround

There is no workaround for this erratum.

Affected Products

All TSC80251G1 Step A derivatives are affected.

Reference	Erratum
C251G1–18	**JBC instruction**

Problem

The JBC instruction uses a Read Read Modify Write instruction. The write back is performed in any case, regardless of the value of the bit. In such case a bit set between the first and second read will be overwritten and information is lost.

Implication

This error affects only the following bits: external interrupts flag (IE0 and IE1) and keyboard interrupt flags (P1F.x).

Workaround

The workaround for this erratum is to not use the JBC instruction for the flags IE0, IE1 and P1F.x.

Affected Products

All TSC80251G1 Step A derivatives are affected.

Reference	Erratum
C251G1–19	**Watchdog timer refresh**

Problem

Hardware watchdog refresh fails depending on the position of the refresh sequence in the execution flow.

Implication

This error affects the usage of hardware watchdog timer: the watchdog timer resets controller.

Workaround

The workaround for this erratum in internal execution is as follow: refreshment must be done twice and execution must not be interrupted. In external execution, the workaround depends on the configuration programmed: number of wait state, page or non page mode. Please contact c251 hotline if you need help to implement this workaround.

```
C2 AF           clr      EA                ; sequence must not be interrupted
75 A6 1E        mov      WDTRST,#1Eh
75 A6 E1        mov      WDTRST,#E1h
75 A6 1E        mov      WDTRST,#1Eh
75 A6 E1        mov      WDTRST,#E1h
D2 AF           setb     EA                ; release interruptions
```

Affected Products

All TSC80251G1 Step A derivatives are affected.

Reference	Erratum
C251G1–20	**Negative flag**

Problem

The Negative (N) Flag of PSW1 should be set or cleared corresponding to bit 15 of the result. Instead, it corresponds to bit 7 of the result.

Implication

The following instructions are affected: SRL WRj – SRA WRj – SLL WRj – INC WRj, #short – DEC WRj, #short.

Workaround

There are two ways to work around this erratum:

- Follow the affected instructions listed above with an operation that will rectify the Negative (N) flag correctly.

```
SRL WRj
ANL WRj,#0FFFFh

SLL WRj
ANL WRj,WRj
```

- Check bit 15 of the result vs. relying on the N flag for sign.

Affected Products

All TSC80251G1 Step A derivatives are affected.

Reference	Erratum
C251G1–21	**WSB – Wait state for memory region 01:**

Problem

The WSB bit of configuration byte CONFIG1 is intended to configure 0 or 1 wait states for all MOVX (including MOVX @DPTR and MOVX @Ri) instructions for the region 01:0000h–01:FFFFh. However, in Step A devices, the MOVX @Ri uses WSA of the configuration byte CONFIG0 instead of WSB.

Implication

The error affects the generating wait state by using WSB bit of configuration byte CONFIG0 during MOVX instruction at region 01:0000h to 01:FFFFh.

Workaround

There are two ways to work around this erratum:

- Configure both WSA and WSB bits with the same wait state if both MOVX @DPTR and MOVX @Ri instructions are used and require the same wait state.
- If it is required that WSB be configured different from WSA, then limit usage of the MOVX instruction to the MOVX @DPTR format rather than the MOVX @Ri format.

Affected Products

All TSC80251G1 Step A derivatives are affected.

Reference	Erratum
C251G1–22	**EJMP at upper boundary of any 64-Kbyte region of memory**

Problem

The EJMP instruction is intended for extended jumps from one 64-Kbyte region of memory to another 64-Kbyte region of memory. However, when EJMP occurs at the upper boundary of any 64-Kbyte region of memory, the Program Counter (PC) will land in the wrong 64K–byte region of memory.

Example

The code: “FEFFFD 8A00XXXX EJMP #00XXXXh” should branch to address 00:XXXXh, but it branches to address region 01:XXXXh instead.

The operation of this instruction is shown below:

```
(PC)         <–– (PC) + 2        PC = FEFFFFh
(PC.23:16)   <–– (Addr.23:16)    PC = 00FFFFh
(PC)         <–– (PC) + 2        PC = 010001h*
(PC.15:0)    <–– (Addr.15:0)     PC = 01XXXXh
```

(*) Overflow occurs when incrementing the PC for the lower two bytes of destination address. The overflow has incremented the PC.23:16 and caused a deviation from the destination Addr.23:16.

Implication

The erratum affects the function of EJMP instruction at the upper boundary region.

Workaround

Avoid this instruction being used at the upper boundary of any 64-Kbyte memory region.

Affected Products

All TSC80251G1 Step A derivatives are affected.

Reference	Erratum
C251G1–23	**Short jumps from memory region FF: to FE:**

Problem

The short jump instruction is intended for a +127 or –128 byte jump relative to the current instruction. However, a short jump from the address in the lower boundary of region FF: will not branch to the address in upper boundary of region FE: even though the destination address is within this –128 byte range. Instead of landing at address in region FE:, it will remain at address in region FF:.

Example

```
FEFFF0                      ORG FEFFF0h
FEFFF0 00       REL_ADR:  NOP
FF0002                      ORG FF0002h
FF0002 80EC                 SJMP REL_ADR
```

This code should branch to address FEFFF0h, but it branches to address FFFFF0h in the same region FF: instead.

Implication

The following instructions are affected: SJMP – CJNE – DJNZ – JB – JBC – JC – JE – JG – JLE – JNB – JNC – JNE – JNZ – JSG – JSGE – JSL – JSLE – JZ.

Please note that there is no errata for short jumps from upper boundary of region FE: to lower boundary of region FF: and no errata for short jumps from upper boundary of region 00: to lower boundary of region 01: and vice–versa.

Workaround

Avoid these instructions being used for a short jump from the lower boundary of region FF: to the upper boundary of region FE:.

Affected Products

All TSC80251G1 Step A derivatives are affected.

6. Documentation Changes

6.1. EPROM Programming Revision B

Chapter 4 in Section III of TSC80251G1 Design Guide 1996 (Rev. A) is replaced by the following document: EPROM Programming Rev. B, 27 May 1997. This document will be included in the next revision of the TSC80251G1 Design Guide.

6.2. T_{RLDV} Values

Table 2.1, Chapter 2, Section III of TSC80251G1 Design Guide 1996 (Rev. A) contains wrong T_{RLDV} values. At 12 MHz, the maximum equals to 72 ns instead of 33 ns, at 16 MHz the maximum equals to 45 ns instead of 13 ns.