Errata Sheet

Rel. 1.2, 03.12.2010

Device	**XC822/824MT Series**
Marking/Step	**AA**
Package	**PG-TSSOP-16/DSO-20**

This Errata Sheet describes the deviations from the current user documentation.The module oriented classification and numbering system uses an ascending sequence over several derivatives, including already solved deviations. So gaps inside this enumeration can occur.

This Errata Sheet covers the following devices:

- XC822T-0FR
- XC822-1FR
- XC822T-1FR
- XC822M-1FR
- XC822MT-1FR
- XC824M-1FG
- XC824MT-1FG

Table 1 Current Documentation

XC82x User's Manual	V1.1	Jun 2010
XC822/824 Data Sheet	V1.1	Oct 2010

Each erratum identifier follows the pattern **Module_Arch.TypeNumber**:

- **Module**: subsystem, peripheral, or function affected by the erratum
- **Arch**: microcontroller architecture where the erratum was firstly detected
  - **AI**: Architecture Independent
  - **CIC**: Companion ICs
  - **TC**: TriCore

  - **X**: XC166 / XE166 / XC2000 Family
  - **XC8**: XC800 Family
  - **[none]**: C166 Family
- **Type**: category of deviation
  - **[none]**: Functional Deviation
  - **P**: Parametric Deviation
  - **H**: Application Hint
  - **D**: Documentation Update
- **Number**: ascending sequential number within the three previous fields. As this sequence is used over several derivatives, including already solved deviations, gaps inside this enumeration can occur.

Note: Devices marked with EES or ES are engineering samples which may not be completely tested in all functional and electrical characteristics, therefore they should be used for evaluation only.

The specific test conditions for EES and ES are documented in a separate Status Sheet.

1 History List / Change Summary

Table 2 History List

Version	Date	Remark
1.0	19.02.2010	
1.1	30.06.2010	

Table 3 Errata fixed in this step

Errata	Short Description	Chg

Table 4 Functional Deviations

Functional Deviation	Short Description	Chg	Pg
PM_XC8.003	Wake-up triggered before power down sequence completed		5

Table 5 Deviations from Electrical- and Timing Specification

AC/DC/ADC Deviation	Short Description	Chg	Pg

Table 6 Application Hints

Hint	Short Description	Chg	Pg
ADC_AI.H001	Arbitration Mode with disabled Arbitration Slots		7
ADC_XC8.H001	Arbitration mode when using external trigger at the selected input line REQTR		7
CCU6_XC8.H002	CCU6 PM event in center-aligned mode	Update	8
LIN_XC8.H002	LIN Break/Synch field detection		9
SYS_XC8.H003	Effective write for Read-Modify-Write instructions of two bytes, one machine cycle	New	9

2 Functional Deviations

PM_XC8.003 Wake-up triggered before power down sequence completed

Setting the PD bit in PMCON0 register will cause the device to go into power down mode. When this bit is set, flash memory will start the powered down sequence immediately. If a wakeup event happens before flash powers down completely, the device will not be able to wake-up normally. The flash memory requires the following maximum time to complete power down:

- 160 usec if the flash is in program or erase mode
- 350 nsec if the flash is in read or idle mode

Workaround

None.

3 Deviations from Electrical- and Timing Specification

4 Application Hints

ADC_AI.H001 Arbitration Mode with disabled Arbitration Slots

In arbitration mode (bit ARBM = 1_B in register PRAR), the arbiter only runs while at least one conversion request is pending, otherwise it waits in an idle state for a request to become active. This leads to a constant and reproducible latency from an incoming request to the conversion start.

Each request source x (x = 0, 1) can be individually selected (via the Arbitration Slot Enable bits ASENx in register PRAR) to take part in the arbitration round. However, if a disabled request source (bit ASENx = 0_B) has a pending request, the related conversion is not started, but the arbiter does not stop and wait. As a result, the latency for requests generated on other arbitration slots is not constant.

To avoid this effect, first disable the request generation of the respective source by setting bit ENGT = 0_B in the corresponding register QMR0 or CRMR1 before disabling the arbitration slot via ASENx = 0_B.

ADC_XC8.H001 Arbitration mode when using external trigger at the selected input line REQTR

If an external trigger is expected at the selected input line REQTR to trigger a pending request, the arbitration mode should be set (PRAR.ARBM=1) where the arbitration is started by pending conversion request. This selection will minimize the jitter between asynchronous external trigger with respect to the arbiter and the start of the conversion. The jitter can only be minimized while no other conversion is running and no higher priority conversion can cancel the triggered conversion. In this case, a constant delay (no jitter) has to be taken into account between the trigger event and the start of the conversion.

CCU6_XC8.H002 CCU6 PM event in center-aligned mode

After detecting a period match (PM A) in centre-aligned mode, T12 counts down from PM + 1 as shown below:

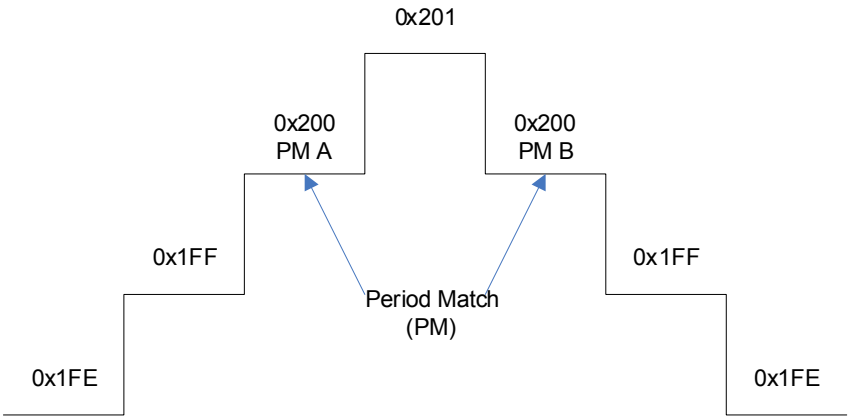

Figure 1 Counting sequence of T12 in center-aligned mode

This means a second PM event (PM B) will occur during the counting down. If ADC is triggered externally via REQTRxC (T12PM), it will be triggered twice in succession. Depending on how real-time the application code is running as well as the T12 count rate and ADC conversion rate, the application could observe two ADC interrupts - once at PM A and once at PM B.

To avoid triggering twice the ADC interrupts, it is suggested to use REQTRxF from multi-channel mode instead of REQTRxC as the trigger source for ADC. Additional initialization are as follows:

- Configure MCMCTR.SWSEL = 101_B (Transfer on T12 period match)
- Configure MCMCTR.SWSYN = 00_B (Direct transfer)
- Write to MCMOUTSTL = CF_H (To enable multi-Channel PWM pattern on CC6x and COUT6x)

Note: Independent of the external trigger, the CCU6 internal triggers based on T12 PM (e.g. T12 PM interrupt or shadow transfer) are only activiated once while T12 is counting up.

LIN_XC8.H002 LIN Break/Synch field detection

The LIN Break/Synch field detection is default enabled after every reset (`BCON.BRDIS` bit = 0). However, to ensure the first standard LIN frame can always be detected, it is recommended to initialize the detection logic in the user code before receiving the frame. This is through the following two steps:

1. Toggle `BCON.BRDIS` bit (set the bit to 1 before clearing it back to 0).
2. Clear the three status flags `LINST.BRK`, `LINST.EOFSYN` and `LINST.ERRSYN` to 0.

It is also recommended to toggle the `BCON.BRDIS` bit after the reception of each complete LIN frame to avoid a wrong Break field detection in noisy environments (i.e. spikes on the LIN bus).

SYS_XC8.H003 Effective write for Read-Modify-Write instructions of two bytes, one machine cycle

When read-modify-write instructions requiring 2 bytes and 1 machine cycle (equivalent to 2 CCLK cycles) for execution, such as INC dir, are executed from memories without any wait states[1)], the actual write to the destination is delayed by the internal bus for up to one CCLK cycle. This means that even though the CPU completes the instruction execution after 2 CCLK cycles, the write through the internal bus may take effect only after a further CCLK cycle.

The list of affected read-modify-write instructions is shown below:

Table 7

Mnemonic	Hex Code	Bytes	No. of CCLK cycles (without wait states)
INC dir	05	2	2
DEC dir	15	2	2
ANL dir, A	52	2	2

1) Applicable also to Flash memory with parallel read feature.

Table 7

Mnemonic	Hex Code	Bytes	No. of CCLK cycles (without wait states)
ORL dir, A	42	2	2
XRL dir, A	62	2	2
XCH A, dir	C5	2	2
CLR bit	C2	2	2
SETB bit	D2	2	2
CPL bit	B2	2	2