

# Errata Sheet

10 July 2000 / Release 1.5

<b>Device:</b>	<b>C505A-4RM, C505CA-4RM</b>
<b>Stepping Code / Marking:</b>	<b>ES-BA, BA</b>
<b>Package:</b>	<b>P-MQFP-44</b>

This Errata Sheet describes the deviations from the current user documentation. The classification and numbering system is module oriented in a continual ascending sequence over several derivatives, as well already solved deviations are included. So gaps inside this enumeration could occur.

The current documentation is: C505(C) User's Manual 08.97  
C505A/C505CA Addendum to C505(C) User's Manual 09.97  
C505(A)/C505C(A) Data Sheet 06.00  
Instruction Set Manual 05.98

**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.

## **Change summary to last Errata Sheet Rel. 1.4:**

- Added new items numbered CAN.3, CAN.4, CAN.5.

# Functional Problems:

## **CAN.2: Unexpected Remote Frame Transmission (C505CA only)**

The on-chip CAN module may send an unexpected remote frame with the identifier=0, when a pending transmit request of a message object is disabled by software.

There are three possibilities to disable a pending transmit request of a message object (n=1..14):

- Set CPUUPDn element
- Reset TXRQn element
- Reset MSGVALn element

Either of these actions will prevent further transmissions of message object n.

The symptom described above occurs when the CPU accesses CPUUPD, TXRQ or MSGVAL, while the pending transmit request of the corresponding message object is transferred to the CAN state machine (just before start of frame transmission). At this particular time the transmit request is transferred to the CAN state machine before the CPU prevents transmission. In this case the transmit request is still accepted from the CAN state machine. However the transfer of the identifier, the data length code and the data of the corresponding message object is prevented. Then the pre-charge values of the internal “hidden buffer” are transmitted instead, this causes a remote frame transmission with identifier=0 (11 bit) and data length code=0.

This behavior occurs only when the transmit request of message object n is pending and the transmit requests of other message objects are **not** active (single transmit request).

If this remote frame loses arbitration (to a data frame with identifier=0) or if it is disturbed by an error frame, it is **not** retransmitted.

### **Effects to other CAN nodes in the network**

The effect leads to delays of other pending messages in the CAN network due to the high priority of the Remote Frame. Furthermore the unexpected remote frame can trigger other data frames depending on the CAN node's configuration.

### **Workaround:**

1. The behavior can be avoided if a message object is not updated by software when a transmission of the corresponding message object is pending (TXRQ element is set) **and** the CAN module is active (INIT = 0). If a re-transmission of a message (e.g. after lost arbitration or after the occurrence of an error frame) needs to be cancelled, the TXRQ element should be cleared by software as soon as NEWDAT is reset from the CAN module.

2. The nodes in the CAN system ignore the remote frame with the identifier=0 and **no** data frame is triggered by this remote frame.

**CAN.3: Description in User's Manual regarding the reception of remote frames and the data length code (DLC) field is incorrect (C505CA only)**

It is inaccurately described in the User's Manual on page 6-78 under '*Arbitration Registers*' that '*When the CAN controller stores a remote frame, only the data length code is stored into the corresponding message object*'. The correct should be that the DLC field remains unchanged in the receiving message object, and that the CPU has the responsibility to define the DLC of the answering data frame.

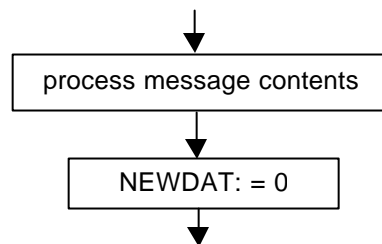
This correction will be updated to the future versions of the User's Manuals.

**Workaround:**

Not applicable.

**CAN.4: Flowchart sequence in figure in User's Manual regarding Micro-controller handling of the Last Message Object is partly incorrect (C505CA only)**

For the software flowchart figure 6-37 in User's Manual 08.97, the correct would be to first 'process message contents' and then to 'clear bit NEWDAT'.



This correction will be updated to the future versions of the User's Manuals.

**Workaround:**

Not applicable.

**CAN.5: Description in User's Manual section 6.4.5 regarding the Configuration of the Bit Timing is partly incorrect (C505CA only)**

As described for the CAN Bit Timing Register High BTR1, the minimum total time requirement for segment 1 and segment 2 is as follows:

$$t_{TSeg1} \geq 3 \times t_q,$$

$$t_{TSeg2} \geq 2 \times t_q.$$

The total bit time remains at ( $t_{TSeg1} + t_{TSeg2} \geq 7 \times t_q$ ).

This correction will be updated to the future versions of the User's Manuals.

**Workaround:**

Not applicable.

**WDT.1: Watchdog Timer is not halted in idle mode**

The Watchdog Timer (WDT) is not halted in the idle mode as defined. However, during the idle mode, an overflow condition of the WDT does not initiate an internal reset. In such a case the WDT starts a new count sequence.

**Workaround:**

1. Do not use the WDT function in combination with the idle mode.
2. In case of WDT is running before entry into idle mode, to avoid a WDT initiated reset upon exit of the idle mode, the following methods can be used.
  - (A) The WDT is refreshed immediately upon exit from idle mode.
  - (B) A timed interrupt can be used to exit the idle mode before the WDT reaches the counter state 7FFCh. This can be achieved by using Timer 0, 1 or 2. This timer can be programmed to generate an interrupt at a WDT counter state prior to overflow, for e.g., at 7F00h. Prior to entering idle mode, the WDT can be refreshed and the timer 0, 1 or 2 can be started immediately to synchronize the WDT. In the interrupt service routine of the Timer 0, 1 or 2, the WDT must be refreshed. If required, idle mode could be entered again.

**SWPD.1: Triggering of Software Power Down (SWPD) wake up immediately after SWPD entry via external interrupt on a frequent basis is not recommended.**

When the micro-controller is running at frequencies lower than 10MHz and the external wake up from SWPD occurs very soon (e.g. <200ms) after entering this mode and this happens on a regular basis, the internal clock may still be valid when the wake up trigger occurs. In this rare case, the micro-controller may get confused with its state and program execution becomes unpredictable.

**Workaround:**

In applications running at 10MHz or below, that enters and exits the SWPD mode on a frequent basis, it is recommended to enter Slow Down mode before Power Down mode entry. On SWPD wake up, the first instruction in the interrupt routine at 07bH should disable Slow Down mode. This method would only cause an insignificant delay in the range of  $\mu$ s and would ensure specified behavior of the micro-controller. Note that when Slow Down mode has been entered, there is no longer a minimum time requirement before SWPD external wake up is triggered.

## **Deviation from Electrical- and Timing Specification:**

**DC.1: Maximum limit of logical 0-to-1 transition current  $I_{TL}$  max is higher**

The maximum for the logical 0-to-1 transition current at the ports does not meet the specified value:

$I_{TL}$  max. is  $-1.05$  mA (instead of  $-0.65$  mA)

**Application Hint:**

If an application is intended to be transferred from C505(C)-L, C505(C)-2R or C505(C)A-4E to C505(C)A-4R step ES-BA, special attention should be paid to the deviation of the  $I_{TL}$  value. The  $I_{TL}$  limits the maximum resistance of the external path when the pin is externally pulled low. Therefore, this should be taken into account when designing and evaluating applications, where the C505(C)A-4R is intended to replace the derivatives mentioned above.

**DC.3: Minimum power supply voltage  $V_{CC}$  min is higher when exiting IDLE mode via external reset**

The operating voltage of the device is 4.85V - 5.5V:

$V_{CC}$  min is 4.85V (instead of 4.25V)

When operating the device below 4.85V, leaving IDLE mode via an external reset on the device may result in erroneous reset state. In such cases, the high byte of program counter may not be reset to 00h correctly.

Provided that IDLE mode is not exited via an external reset on the device, the device is fully functional with power supply in the specified range of  $V_{CC} = 4.25V - 5.5V$ .

**DC.4: Minimum limit of logic 0 input current  $I_{IL}$  min is lower**

The minimum for the logic 0 input current at the ports is lower than the specified values:

$I_{IL}$  min. is  $-7 \mu A$  (instead of  $-10 \mu A$ )

# History List (since last CPU Step ES-AA)

## Functional Problems

Functional Problem	Short Description	Fixed
CAN.2	Unexpected Remote Frame Transmission (C505CA only)	
CAN.3	Description in User's Manual regarding the reception of remote frames and the data length code (DLC) field is incorrect (C505CA only)	
CAN.4	Flowchart sequence in figure in User's Manual regarding Micro-controller handling of the Last Message Object is partly incorrect (C505CA only)	
CAN.5	Description in User's Manual section 6.4.5 regarding the Configuration of the Bit Timing is partly incorrect (C505CA only)	
WDT.1	Watchdog Timer is not halted in idle mode	
SWPD.1	Triggering of Software Power Down (SWPD) wake up immediately after SWPD entry via external interrupt on a frequent basis is not recommended.	

## AC/DC Deviations

AC/DC Deviation	Short Description	Fixed
DC.1	Maximum limit of logical 0-to-1 transition current $I_{TL}$ max is higher	
DC.2	Minimum input high voltage $V_{IH}$ min on all pins except XTAL1 and RESET	Step BA
DC.3	Minimum power supply voltage $V_{CC}$ min is higher when exiting IDLE mode via external reset	
DC.4	Minimum limit of logic 0 input current $I_{IL}$ min is lower	

Application Support Group, Singapore