Microcontrollers

Errata Sheet

29 June 2001 / Release 1.5

Device:	**C541U-1E**
Stepping Code / Marking:	**CA**
Package:	**P-LCC-44**
	P-SDIP-52

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: User's Manual 04.99
Data Sheet 05.99
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:

- New item numbered OTP.1.

Functional Problems:

USB.2: Move Immediate Data to USBVAL Register.

If instructions to move immediate data to USBVAL register occur in order to writing to the USB memory, then incorrect data may be written. An example sequence is shown below :

```
MOV    USBVAL, #const8

…..
MOV    USBVAL, #const8
```

Workaround:

As a workaround, a temporary register or accumulator can be used to transfer data to the USB memory. Examples are shown below:

1. Using Accumulator

```
MOV    A, #const8
MOV    USBVAL, A
...
MOV    A, #const8
MOV    USBVAL, A
```

2. Using Registers

```
MOV    Rr, #const8
MOV    USBVAL, Rr
...
MOV    Rr, #const8
MOV    USBVAL, Rr
```

where, r = 0 to 7.

USB.3: Accessing the EPSEL Register will Generate SODn Interrupts.

When accessing the EPSEL register to select one of the sets of registers related to the Endpoints (EP1~EP4), an SODn interrupt corresponding to the selected EP will be generated continually and the SOD0 (for EP0) may also be generated. Depending on the software this could cause the device sending wrong data packets and packet length to the USB Host.

Workaround:

After accessing the EPSEL register, the SODn should be cleared by a Read operation on the EPIRn register, where n is the EP number. A Read operation on the EPIRn for EP0 is required at the end of the procedure.

E.g. after selecting the EP2, a Read operation is performed on EPIRn for EP2 and the same for EP0 at the end of the EP procedure, as shown below

```
MOV    EPSEL, #02H
MOV    A, EPIRn
……..
Instruction sequence handling EP related data
……..
MOV    EPSEL, #00H

MOV    A, EPIRn
```

USB.4: ESP0 bit can not be reset by hardware.

The ESP0 bit is set to enable the Control endpoint, EP0, to acknowledge the Status Phase of a Control transfer. After the Status Phase is acknowledged by C541U, the EPS0 bit should be reset by hardware. However, when performing OUT transfers on EP0, the EPS0 may not be cleared by hardware even after completed the Status Phase.

Workaround:

During the normal operation, ESP0 bit is reset by hardware after the Status Phase of Control transfer, the ACK packet sent by C541U will not cause an ACK interrupt to the CPU. When the problem occurs, the ESP0 can not be reset by hardware, an ACK interrupt will be generated at the end of a successful Status Phase. This ACK interrupt can be used to reset the ESP0 bit.
The AIE0 bit should be set in the Device Interrupt Service Routine (ISR) to enable an ACK interrupt.
In the ACK interrupt service routine, the CLREP0 bit can be set to clear the ESP0 bit.
For example: (In the EP ISR)

```
. . . . . .
save_GEPIR = GEPIR;          // save the contents of Global EP interrupt request.
//   End Point 0 Interrupt
if  (save_GEPIR & EPI0_)
{       ……..
        ……..
        EPSEL = 0;              // select EP0.
        save_EPIRn = EPIRn;
        if  (  (save_EPIRn & ACK_)   &&   (EPIEn & AIE_)  )     // if USB ACK packet interrupt is received
                                                                // where ACK0 & AIE0 bits are set.
           {    EPBSn |=  CLREP_;        // to reset the ESP0 bit.
                ……..
                ……..
            }
……..
……..
```

USB.5: The Status Phase for OUT Transfers is not acknowledged by C541U.

During the Bi-directional Control transfers on EP0, the Status Phase for OUT transfers might not be acknowledged by C541U even thought the ESP0 bit has been set.

Workaround:

The NAIE0 bit should be set in the EP Interrupt Service Routine (ISR) to enable an NACK interrupt for EP0 after all of data sent by host has been read. When the problem occurs, a NACK will cause an EP interrupt for EP0. A NACK ISR should be included in order to solve the problem. In the NACK ISR, the DONE bit is set to cause the zero length data packet to be sent to the host. This will end the Control transfer.

For example:

```
//  EndPoint ISR
. . . . . .
save_GEPIR = GEPIR;          // save the contents of Global EP interrupt request.
//    End Point 0 Interrupt
if  (save_GEPIR & EPI0_)
{       ……..
        ……..
        EPSEL = 0;               // select EP0.
        save_EPIRn = EPIRn;
        if  (  (save_EPIRn & NACK_)   &&   (EPIEn & NAIE_)  )
// if Status Phase failed, the NACK0 and NAIE0 bits will be set. Once detected the NACK interrupt,
// clear the ESP0 bit and set the DONE0 bit.
          {     EPBSn  |=  CLREP_;          // to reset the ESP0 bit.
                EPBSn  |=  DONE_;           // send zero length data packet to terminate the Status
                                            // phase.
                ……..
                ……..
           }
……..
……..
```

USB.6: Data Byte may be lost during BULK IN Transfers on Data EPs.

To perform the BULK IN transfers on data EPs, the microcontroller writes data bytes to USBVAL, with the buffer address auto-increment feature, until the USB buffer is full (64 bytes per packet). During this transfer, data byte may be lost because of the data loading into USBVAL is too fast.

Workaround:

Place some NOP instructions in the software subroutine to slow down the speed of writing data to USBVAL. This will synchronize the performance of USB buffer.

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 Watchdog Timer 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, 1or 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.

WDT.2: Setting the WDT and SWDT bits in WDCON Register by using Immediate Addressing mode instruction may not refresh the Watchdog Timer

To initiate a refresh of the Watchdog timer, the WDT bit should be set directly before the SWDT bit in WDCON register (WDCON.1 and WDCON.0). However, using the Immediate Addressing mode instruction to set the WDT and SWDT bits may not refresh the Watchdog timer, the Watchdog timer may keep counting until overflow and reset the device.

For example:

```
MOV   WDCON, #02h
MOV   WDCON, #01h
```

These instructions will not refresh the Watchdog timer.

Workaround:

Use the Bit Manipulation Instruction to set the WDT and SWDT bits.

For example:

```
SETB  WDCON.1
SETB  WDCON.0
```

OTP.1: OTP module may fail under special conditions, leading to undefined operation

The OTP module may malfunction, causing the chip to enter an undefined state with unsteady operation, if there is a remaining voltage at the V_{DD} pin before powering up. The critical remaining voltage is approximately 100-400mV. The undefined state can only be left by a complete power off (V_{DD}=0V) and not by any RESET-source (e.g. hardware reset, WDT-reset). The problem is due to variation in technology and manufacturing parameters.

Workaround:

The device should always be powered up from V_{DD}=0V, ensuring that there is no voltage at any pins which leads to a remaining voltage level at V_{DD} pin (coupling over the ESD-structure).

Deviation from Electrical- and Timing Specification:

DC.3: V_{IH} (min) for pin EA# .

The minimum input high voltage, V_{IH} (min), for pin EA# (active low) does not meet the specified values:

V_{IH} (min) for EA pin is $0.6 \bullet V_{DD}$ (instead of $0.2 \bullet V_{DD} + 0.9$ V)

The parameter has been updated in the new C541U Data Sheet version 05.99.

DC.4: V_{OL} (max.) for Port 0.

The maximum output low voltage, V_{OL} (max.), for the Port 0 does not meet the specified values:

V_{OL} (max.) for Port 0 is 0.6 V (instead of 0.45 V)

The parameter has been updated in the new C541U Data Sheet version 05.99.

DC.5: The maximum Logic 0 Input Current I_{IL} (max.) for Ports 1,2 & 3.

The maximum Logic 0 input current, I_L (max.), for the Ports 1,2 & 3 does not meet the specified values:

I_{IL} (max.) for Ports 1,2 & 3 is –60 µA (instead of –50 µA)

The parameter has been updated in the new C541U Data Sheet version 05.99.

DC.6: V_{DD} is valid for a smaller range than specified on documents

V_{DD} is valid in the range from 4.35V to 5.5V at all specified temperatures, instead of 4.25V to 5.5V as specified on the documents. This smaller range is effective on devices with date code starting from *0116.*

History List (since last CPU Step)

Functional Problems

Functional Problem	Short Description	Fixed
USB. 1	USB EP1 packet size configuration.	Step CA
USB. 2	Move Immediate Data to USBVAL Register.	
USB. 3	Accessing the EPSEL Register will Generate SODn Interrupts.	
USB. 4	ESP0 bit can not be reset by hardware.	
USB. 5	The Status Phase for OUT Transfers is not acknowledged by C541U.	
USB. 6	Data Byte may be lost during BULK IN Transfers on Data EPs.	
WDT. 1	Watchdog Timer is not halted in Idle Mode	
WDT. 2	To refresh the Watchdog Timer, the WDT and SWDT bits in WDCON Register can not be set by using Immediate Addressing mode instruction	
OTP.1	OTP module may fail under special conditions, leading to undefined operation	

AC/DC Deviations

AC/DC Deviation	Short Description	Fixed
DC. 1	Device transceiver has voltage overshoot.	Step CA
DC. 2	Power consumption in suspend mode.	Step CA
DC. 3	V_{IH} (min.) for the pin EA#.	DS 05.99
DC. 4	V_{OL} (max.) for Port 0.	DS 05.99
DC. 5	The maximum Logic 0 Input Current I_{IL} (max.) for Ports 1,2 & 3.	DS 05.99
DC. 6	V_{DD} is valid for a smaller range than specified on documents	

Application Support Group, Singapore