AMENDMENT

Am186™CC/CH/CU

Microcontrollers User's Manual

This document amends the *Am186™CC/CH/CU Microcontrollers User's Manual*, order #21914B. It consists of these parts:

- "Documentation Defects and Corrections" on page 1 lists corrections to be made in page number order.
- "Changed Figures" on page 10 provides edited versions of changed tables and figures.
- An Index is included at the end of this amendment.

DOCUMENTATION DEFECTS AND CORRECTIONS

Table 1 on page 2 lists defects that have been found in the *Am186™CC/CH/CU Microcontrollers User's Manual,* order #21914B. Defects are listed in page order. Each entry lists the following:

- page number
- item to be corrected
- original text (or description of text to change)
- corrected text (or description of change to make)
- comment explaining the change

Entries that correct text in a diagram or figure do not contain the entire diagram or figure. If graphical information is changed, the table refers to the page in this amendment where the changed figure can be found.

Square brackets ([ ]) are used to indicate a description of the text or change to be made, as opposed to the actual text.

Unchanged portions of a paragraph are replaced by an ellipsis (...) in entries where this might make the change easier to find. The whole paragraph is included if it is useful for understanding why the change was made.

Publication# **21914** Rev: **B** Amendment/**1**
Issue Date: **February 2000**

Table 1. Corrections to the *Am186™CC/CH/CU Microcontrollers User's Manual*, Rev. B

Page	Item	Original Text	Change To	Comment
Chapter 1 Architectural Overview				
1-10	**1.4.3.2 General-Purpose DMA Channels (Chapter 8)**, 2nd paragraph	External peripherals support DMA transfers through the external DMA request pins (DRQ1–DRQ0). Each general-purpose channel accepts a DMA request from one of three sources: the DMA request signals (DRQ1–DRQ0), Timer 2, or the UARTs. (Note that Timer 2 acts only as a DMA request source; no data is transferred to or from Timer 2.)...	External peripherals support DMA transfers through the external DMA request pins (DRQ1–DRQ0). Each general-purpose channel can accept synchronized DMA requests from these sources: the DMA request signals (DRQ1–DRQ0), Timer 2, or the UARTs. (Note that Timer 2 acts only as a DMA request source; no data is transferred to or from Timer 2.) In addition, system software can initialize and start unsynchronized DMA transfers...	Note that this paragraph intentionally omits USB as a DMA request source. USB requests are mentioned in the subsequent paragraph in the manual.
Chapter 3 System Overview				
3-5	**3.4 Initialization and Reset**, 3rd paragraph from bottom, 2nd sentence	Pins are latched on the deassertion of $\overline{\text{RES}}$, and therefore are not affected by an internal watchdog-timer-generated reset.	Some pin states are latched only on the deassertion of $\overline{\text{RES}}$, and therefore are not affected by an internal watchdog-timer-generated reset.	
3-16	**Table 3-7 Signal Descriptions**, Reserved pins	On the Am186CH HDLC microcontroller, the RSVD_75 pin should be tied externally to V_{SS}. On the Am186CH HDLC microcontroller, pins RSVD_75, RSVD_76, RSVD_80, RSVD_81, and RSVD_101–RSVD_104 and are reserved. On the Am186CC and Am186CU microcontrollers, pins RSVD_101–RSVD_104 are reserved unless pinstrap {$\overline{\text{USBXCVR}}$} is sampled Low on the rising edge of RESET. On the Am186CU USB microcontroller, pins RSVD_119–RSVD_116 are reserved. All other reserved pins should not be connected.	Some pins are reserved only on certain microcontrollers or in a particular pinstrap configuration. On the Am186CH HDLC microcontroller, pins RSVD_75, RSVD_76, RSVD_80, RSVD_81, and RSVD_101–RSVD_104 are reserved. On the Am186CC and Am186CU microcontrollers, pins RSVD_101–RSVD_104 are reserved unless pinstrap {$\overline{\text{USBXCVR}}$} is sampled Low on the rising edge of RESET. On the Am186CU USB microcontroller, pins RSVD_119–RSVD_116 are reserved. With one exception, all reserved pins should be left unconnected. The exception is that, on the Am186CH HDLC microcontroller, the RSVD_75 pin should be tied externally to V_{SS}.	Clarify reserved pin usage.
Chapter 7 Interrupts				
7-20	**7.5.7 Software-Related Considerations**, second bullet	Writing a zero to the appropriate channel bit in the Interrupt Request (REQST) register clears the pending interrupt. This facility provides a simple way to clear a spurious edge-triggered interrupt that may have occurred when initially configuring a PIO pin as an interrupt source.	Writing a zero to the appropriate channel bit in the Interrupt Request (REQST) register clears the pending edge-triggered interrupt. This facility provides a simple way to clear a spurious edge-triggered interrupt that may have occurred when initially configuring a PIO pin as an interrupt source. Note that for level-triggered interrupts, the interrupt source must be cleared to clear the interrupt.	

Table 1. Corrections to the *Am186™CC/CH/CU Microcontrollers User's Manual*, Rev. B (Continued)

Page	Item	Original Text	Change To	Comment
Chapter 8 DMA Controller				
8-1	**8.1 Overview**, end of fourth paragraph, marked "CC"	...Each general-purpose channel accepts a DMA request from one of four sources: the DMA request signals (DRQ1–DRQ0), Timer 2, the UARTs, or the USB peripheral controller. (Note that Timer 2 acts only as a DMA request source; no data is transferred to or from Timer 2.)	...Each general-purpose channel can accept synchronized DMA requests from four sources: the DMA request signals (DRQ1–DRQ0), Timer 2, the UARTs, or the USB peripheral controller. (Note that Timer 2 acts only as a DMA request source; no data is transferred to or from Timer 2.) In addition, system software can initialize and start unsynchronized DMA transfers.	
8-1	**8.1 Overview**, end of fifth paragraph, marked "CH"	...Each general-purpose channel accepts a DMA request from one of three sources: the DMA request signals (DRQ1–DRQ0), Timer 2, or the UARTs. (Note that Timer 2 acts only as a DMA request source; no data is transferred to or from Timer 2.)	...Each general-purpose channel can accept synchronized DMA requests from four sources: the DMA request signals (DRQ1–DRQ0), Timer 2, or the UARTs. (Note that Timer 2 acts only as a DMA request source; no data is transferred to or from Timer 2.) In addition, system software can initialize and start unsynchronized DMA transfers.	
8-2	End of partial first paragraph (marked "CU" on the manual's previous page)	...Each general-purpose channel accepts a DMA request from one of four sources: the DMA request signals (DRQ1–DRQ0), Timer 2, the UARTs, or the USB peripheral controller. (Note that Timer 2 acts only as a DMA request source; no data is transferred to or from Timer 2.)	...Each general-purpose channel can accept synchronized DMA requests from four sources: the DMA request signals (DRQ1–DRQ0), Timer 2, the UARTs, or the USB peripheral controller. (Note that Timer 2 acts only as a DMA request source; no data is transferred to or from Timer 2.) In addition, system software can initialize and start unsynchronized DMA transfers.	
8-18	**Figure 8-4 Source-Synchronized General-Purpose DMA Transfers**, cycle labels at top of figure	Fetch Cycle Fetch Cycle	Fetch Cycle Deposit Cycle	Second phase of transfer is deposit cycle.
8-19	Last paragraph on page	A DMA request is not acknowledged from the same source for four processor clock cycles after the end of the deposit cycle. In a source-synchronized DMA transfer, the DRQ signal must be deasserted at least four clocks before the end of the transfer....	In a source-synchronized DMA transfer, the DRQ signal must be deasserted at least four clocks before the end of the transfer....	Delete first sentence of paragraph.

Table 1. Corrections to the *Am186™CC/CH/CU Microcontrollers User's Manual*, Rev. B (Continued)

Page	Item	Original Text	Change To	Comment
8-20	First paragraph on page	A destination-synchronized transfer differs from a source-synchronized transfer in that the four cycle delay allows the destination device to deassert its DRQ signal four clocks before another request is latched. Without this delay, the destination device would not have time to deassert its DRQ signal. Because of the four extra cycles, a destination-synchronized DMA channel allows other bus masters to take the bus during the idle states.	A destination-synchronized transfer differs from a source-synchronized transfer in that a destination-synchronized DRQ is masked off for four cycles after the deassertion of the $\overline{WR}$ signal. This allows an external or internal device to use $\overline{WR}$, in conjunction with chip selects or address lines, to signal the end of the deposit cycle. The destination-synchronized device must then deassert DRQ within four cycles in order to signal that the device is not ready for the next DMA transfer. While the destination-synchronized DRQ is masked off, the bus can be accessed by the CPU or other, possibly lower priority, bus masters.	Rewrite the paragraph.
8-25	Next to last paragraph on page, last two sentences	For this reason, the High-Speed UART has an additional Overrun Error-Immediate (OERIM) interrupt bit that is not placed in the FIFO. Software can monitor or interrupt on OERIM to detect and correct this sort of system programming error.	For this reason, the High-Speed UART has additional Overrun Error-Immediate (OERIM) and Break Immediate (BRKIM) interrupt bits that are not placed in the FIFO. When the receive FIFO is enabled, software can monitor or interrupt on OERIM or BRKIM to detect and correct this sort of system programming error. ***Note:*** *The BRKIM bit is not available in parts released prior to revision C1. For processor revision information, see the PRL register description in the* Am186CC/CH/CU Register Set Manual*, order #21916B [and its amendment, order #21916B/1].*	Add BRKIM bit.
8-28	**8.5.7.3 SmartDMA™ Channel Memory Overview**, third paragraph	[Existing third paragraph]	[Add note after paragraph] Note: The SmartDMA channel descriptor rings must reside in 16-bit memory. The buffers pointed to by the descriptors can be in either 8-bit or 16-bit memory.	

Table 1. Corrections to the *Am186™CC/CH/CU Microcontrollers User's Manual*, Rev. B (Continued)

Page	Item	Original Text	Change To	Comment
8-43	**8.5.9 Software-Related Considerations**	Software must stop DMA operation before writing to the GDxCON1 register, or the results are unpredictable. Stopping the SmartDMA channel has no effect while a request is pending on the channel. Before stopping the channel, make sure the requesting peripheral (HDLC channel or USB endpoint) is stopped.	Software must stop DMA operation before writing to the GDxCON1 register, or the results are unpredictable. Stopping the SmartDMA channel has no effect while a request is pending on the channel. Before stopping the channel, make sure the requesting peripheral (HDLC channel or USB endpoint) is stopped. If the requesting peripheral is stopped by an error, the error handler should stop the SmartDMA channel before clearing the status bit. Otherwise the SmartDMA request might be reasserted before software can stop the channel.	
Chapter 9 Programmable I/O Signals				
9-2	**Figure 9-1 PIO Operation Block Diagram**, pullup/ pulldown resistor value	100K	50 k-Ω	Correct resistor value.
9-5	**9.5.2 Defining the PIO Signal as Input or Output**, last sentence on page	The internal pullup and pulldown resistors each have a value of approximately 10 K Ω.	The internal pullup and pulldown resistors each have a value of approximately 50 k Ω.	
Chapter 11 Watchdog Timer				
11-1	**11.1 Overview**, last sentence of first paragraph.	... $\overline{\text{RESOUT}}$ signal, which is pulled Low during an external reset and can be pulled Low during an internal reset.	... RESOUT signal, which is pulled High during an external reset and can be pulled High during an internal reset.	RESOUT is active High, not active Low.
Chapter 13 Asynchronous Serial Ports				
13-5	**13.5.1.1.2 Transmitting Data**, steps 1 and 2.	1. Verify that the THRE bit in the (H)SPSTAT register is set to 1 to ensure the transmit register can be written without loss of data. 2. If FIFOs are being used (High-Speed UART only), instead of polling the THRE bit, verify that the FIFO is not yet full (TTHRSH bit in the HSPSTAT register is set to 1).	1. Verify that the THRE bit in the (H)SPSTAT register is set to 1 to ensure the transmit register can be written without loss of data. ***Note:*** *If FIFOs are not used, software must verify that the THRE bit is set to 1 even if the TEMT bit was set in the (H)SPSTAT register before the previous write.* 2. If FIFOs are being used (High-Speed UART only), software can omit polling the THRE bit only if it is certain there is space in the FIFO. The FIFO contains 16 empty slots if the TEMT bit is set. The FIFO contains at least eight empty slots if hardware sets the TTHRSH bit in the HSPSTAT register after software has cleared it.	

Table 1. Corrections to the *Am186™CC/CH/CU Microcontrollers User's Manual*, Rev. B (Continued)

Page	Item	Original Text	Change To	Comment
13-7	**13.5.1.3 Autobaud Mode**, step 7	7. Wait for the ABAUD bit in the HSPCON1 register to go to 0 to indicate that the autobaud operation is complete. The computed baud divisor is automatically copied into the HSPBDV register, and the autobaud (ABAUD) bit in the HSPCON1 register is cleared.	7. Wait for the ABAUD bit in the HSPCON1 register to go to 0 to indicate that the autobaud operation is complete. The computed baud divisor is automatically copied into the HSPBDV register, and the autobaud (ABAUD) bit in the HSPCON1 register is cleared. This also sets the ABDONE bit in the HSPSTAT register. ***Note:*** *The ABDONE bit is not available in parts released prior to revision C1. For processor revision information, see the PRL register description in the* Am186CC/CH/CU Register Set Manual*, order #21916B [and its amendment, order #21916B/1].*	
13-13	**13.5.4 CTS/RTR Hardware Flow Control**, second and third paragraphs	In the CTS/RTR protocol, the receiver asserts clear-to-send (CTS) whenever there is room in the receiver for more data. The transmitting device should sample CTS before beginning transmission of each frame. CTS is deasserted when the start bit is detected for the last frame that can be read without data loss. When FIFOs are disabled, CTS is deasserted after the start bit for each frame is detected and remains deasserted until the data is read from the receive data register. When the receive FIFO is enabled, CTS is deasserted after the start bit is received for the last frame that will fit in the FIFO. The transmitter samples ready-to-receive (RTR) before transmitting the start bit of each frame. The RTR signal is not sampled during frame transmission. This allows the receiving device to deassert RTR any time before the end of the stop bit. The transmitter does not begin transmitting the start bit for the next frame while RTR is deasserted.	In the CTS/RTR protocol, the UART's ready-to-receive (RTR) output is connected to the attached device's clear-to-send (CTS) input, and the attached device's RTR output is connected to the UART's CTS input. (I.e., the CTS and RTR signals are cross-connected.) The receiver asserts RTR whenever there is room in the receiver for more data. The transmitting device should sample this signal (at its CTS input) before beginning transmission of each frame. The receiver deasserts RTR when the start bit is detected for the last frame that can be read without data loss. When FIFOs are disabled, the receiver deasserts RTR after the start bit for each frame is detected, and holds RTR deasserted until the data is read from the receive data register. When the receive FIFO is enabled, the receiver deasserts RTR after the start bit is received for the last frame that will fit in the FIFO. The transmitter samples its CTS input before transmitting the start bit of each frame. The CTS input is not sampled during frame transmission. This allows the receiving device to deassert its RTR output any time before the end of the stop bit. The transmitter does not begin transmitting the start bit for the next frame while its CTS input is deasserted.	Clarify.

Table 1. Corrections to the *Am186™CC/CH/CU Microcontrollers User's Manual*, Rev. B (Continued)

Page	Item	Original Text	Change To	Comment
13-17–13-19	Discussion of autobaud enhancement; changes throughout	[References to registers HSPAB0–HSPAB3]	[Add a new register HSPAB4 at offset 27E, and modify text accordingly. Wherever the text states that the ABTHRSH3 bit field in the HSPAB3 register must contain the largest threshold, specify the ABTHRSH4 bit field in the HSPAB4 register instead. Also add the following note:] ***Note:*** *The HSPAB4 register is not available in parts released prior to revision C1. For processor revision information, see the PRL register description in the* Am186CC/CH/CU Register Set Manual*, order #21916B [and its amendment, order #21916B/1].* *To maintain compatibility with existing software that does not initialize the HSPAB4 register, the HSPAB3 register can contain the largest ABTHRSHx value, in which case the HSPAB4 register must be cleared or left in its default disabled state (00h).*	
13-18	Figure 13-9 Autobaud Enhancement	[Existing figure]	[Replace with Figure 13-9 on page 10 of this amendment.]	Update and clarify figure.
13-19	Table 13-5 UARTs Interrupt Sources	[Existing table]	[Add new row before "Overrun error on receive FIFO":] Break on receive FIFO, RSIE (Off), BRKIM (Off)	Add BRKIM bit.
13-20	**13.5.7 Break Detection and Generation**, first paragraph	[Existing first paragraph.]	[Add new sentence and note:] If the receive FIFO is enabled (High-Speed UART only), the break immediate (BRKIM) status bit indicates the break condition as soon as it occurs; the FER and BRK status bits are not set until the corresponding character is loaded into the HSPRXD register. ***Note:*** *The BRKIM bit is not available in parts released prior to revision C1. For processor revision information, see the PRL register description in the* Am186CC/CH/CU Register Set Manual*, order #21916B [and its amendment, order #21916B/1].*	Add BRKIM bit.

Table 1. Corrections to the *Am186™CC/CH/CU Microcontrollers User's Manual*, Rev. B (Continued)

Page	Item	Original Text	Change To	Comment
Chapter 14 Synchronous Serial Port				
14-4	**14.5.1 Usage**, item #5	5. Wait for the DR/DT bit in the SSSTAT register to go to 0 to indicate the transmit or receive has completed.	5. Wait for the DR/DT bit in the SSSTAT register to go to 1 to indicate the transmit or receive has completed.	
Chapter 16 HDLC External Serial Interface Configuration (TSAs)				
16-5	**16.3 System Design**, first sentence	lists the signals that are multiplexed with other microcontroller functions.	Table 16-1 lists the signals that are multiplexed with other microcontroller functions.	
Chapter 17 General Circuit Interface (GCI)				
17-8	**Figure 17-3 GCI Terminal Mode Frame Structure**	[Existing figure]	[Change to new figure 17-3, shown on page 10 of this amendment.]	Change FSC signal waveform.
Chapter 18 Universal Serial Bus (USB)				
18-4	Last paragraph before figure	[Existing text]	[Add the following sentence:] In these examples, software defines a PIO input (PIO_USB_DETECT) to monitor the USB's V_{USB} signal, and a PIO output (PIO_USB_VCC) to control the 1.5 k-Ω pullup on USBD+.	Clarify PIO use in example.
	Figure 18-2 USB With Internal Transceiver	[Existing figure]	[Change to new figure 18-2, shown on page 11 of this amendment.]	Add pulldown to V_{USB}. Clarify PIO use in example.
18-5	**Figure 18-3 USB With External Transceiver**	[Existing figure]	[Change to new figure 18-3, shown on page 12 of this amendment.]	Add pulldown to V_{USB}.
18-30	**18.5.12 Endpoint Definitions**, second paragraph	The USB Specification, Version 1.0 defines the endpoint configuration process:	The USB Specification, Version 1.0 defines the endpoint configuration process.	Change colon at end of sentence to a period.
	18.5.12 Endpoint Definitions, third paragraph	"Host software should only set configuration and interface values that match a device descriptor returned by the device in response to a GET_DESCRIPTOR command. However, the USB hardware accepts as valid any configuration or feature setting in the range of 0d to 3d, regardless of the available descriptors. To help ensure reliable operation in any USB environment, device software can define a minimal descriptor (i.e., Endpoint 0 with no bandwidth allocation) for any configuration and interface settings that it does not define otherwise."	***Note:*** *Host software should only set configuration and interface values that match a device descriptor returned by the device in response to a GET_DESCRIPTOR command. However, the USB hardware accepts as valid any configuration or feature setting in the range of 0d to 3d, regardless of the available descriptors. To help ensure reliable operation in any USB environment, device software can define a minimal descriptor (i.e., Endpoint 0 with no bandwidth allocation) for any configuration and interface settings that it does not define otherwise.*	Remove quotation marks. Paragraph is not a quotation.

Table 1. Corrections to the *Am186™CC/CH/CU Microcontrollers User's Manual*, Rev. B (Continued)

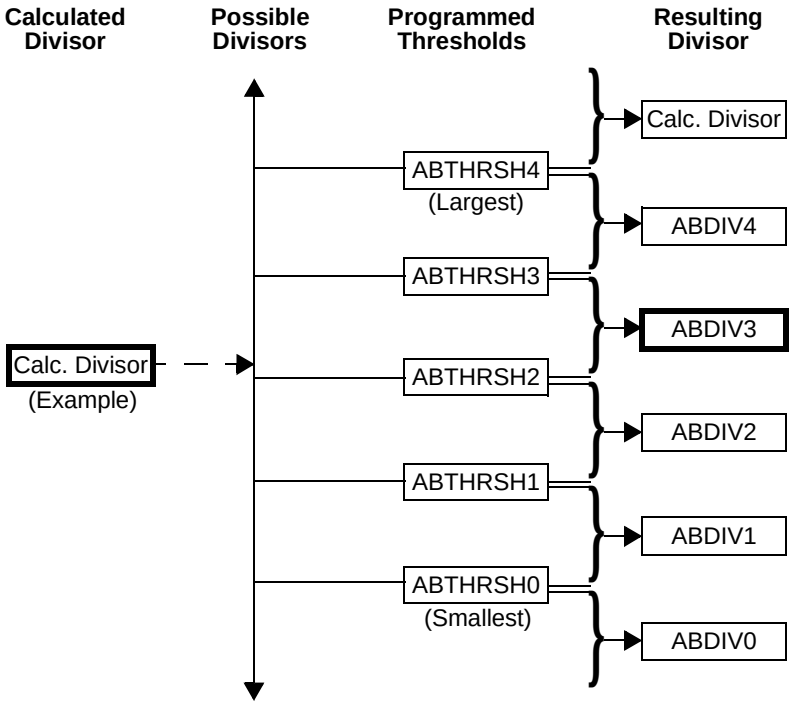
Page	Item	Original Text	Change To	Comment
Appendix A Register Summary				
A-8	CNTCTL register, bit 7	Res	HNDSHK	Add bit.
A-9	IEPCTL register, bit 7			
	AEPCTL register, bit 7			
	BEPCTL register, bit 7			
A-10	CEPCTL register, bit 7			
	DEPCTL register, bit 7			
A-11	HSPSTAT register, bit 13	Res	BRKIM	Add bit.
	HSPSTAT register, bit 11	Res	ABDONE	Add bit.
	HSPIMSK register, bit 13	Res	BRKIM	Add bit.
	HSPIMSK register, bit 11	Res	ABDONE	Add bit.
	HSPABx registers	[Existing rows]	[Add a row for the new HSPAB4 register, similar to the other HSPABx registers, with offset 27Eh, default location FE7Eh, default value 0h, and bit fields ABDIV4 and ABTHRSH4.]	Add register.

CHANGED FIGURES

For your convenience, the following pages contain edited copies of some figures. Figures with minor text changes are not reproduced here; see Table 1 beginning on page 2 for complete change descriptions.

Replace Figure 13-9 on page 13-18 of the manual with the following figure.

Figure 13-9 Autobaud Enhancement

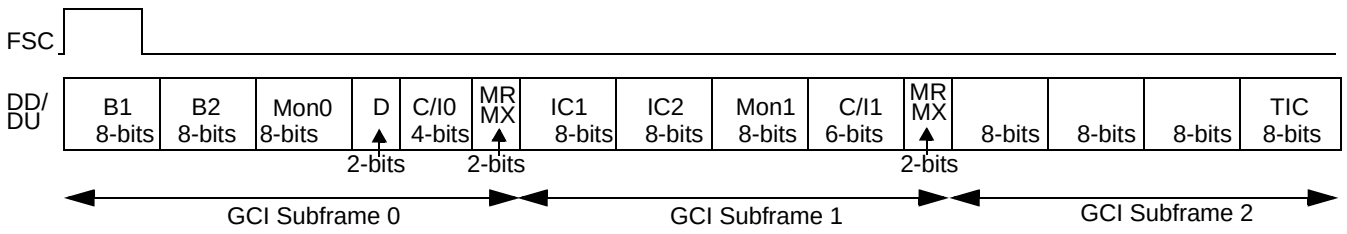


Notes:

1. *If the calculated divisor is larger than the largest ABTHRSHx bit field value, the resulting divisor is the same as the calculated divisor.*
2. *The ABTHRSH4 or ABTHRSH3 bit field must contain the largest threshold value. If the ABTHRSH4 bit field is not the largest threshold, it must be 0. The remaining ABTHRSHx bit fields must be programmed with successively smaller thresholds for lower-numbered ABTHRSHx bit fields. If all five thresholds are not needed, the lowest-numbered ABTHRSHx bit fields (ABTHRSH0, ABTHRSH1, etc.) can be cleared or left in their default disabled state (00h).*

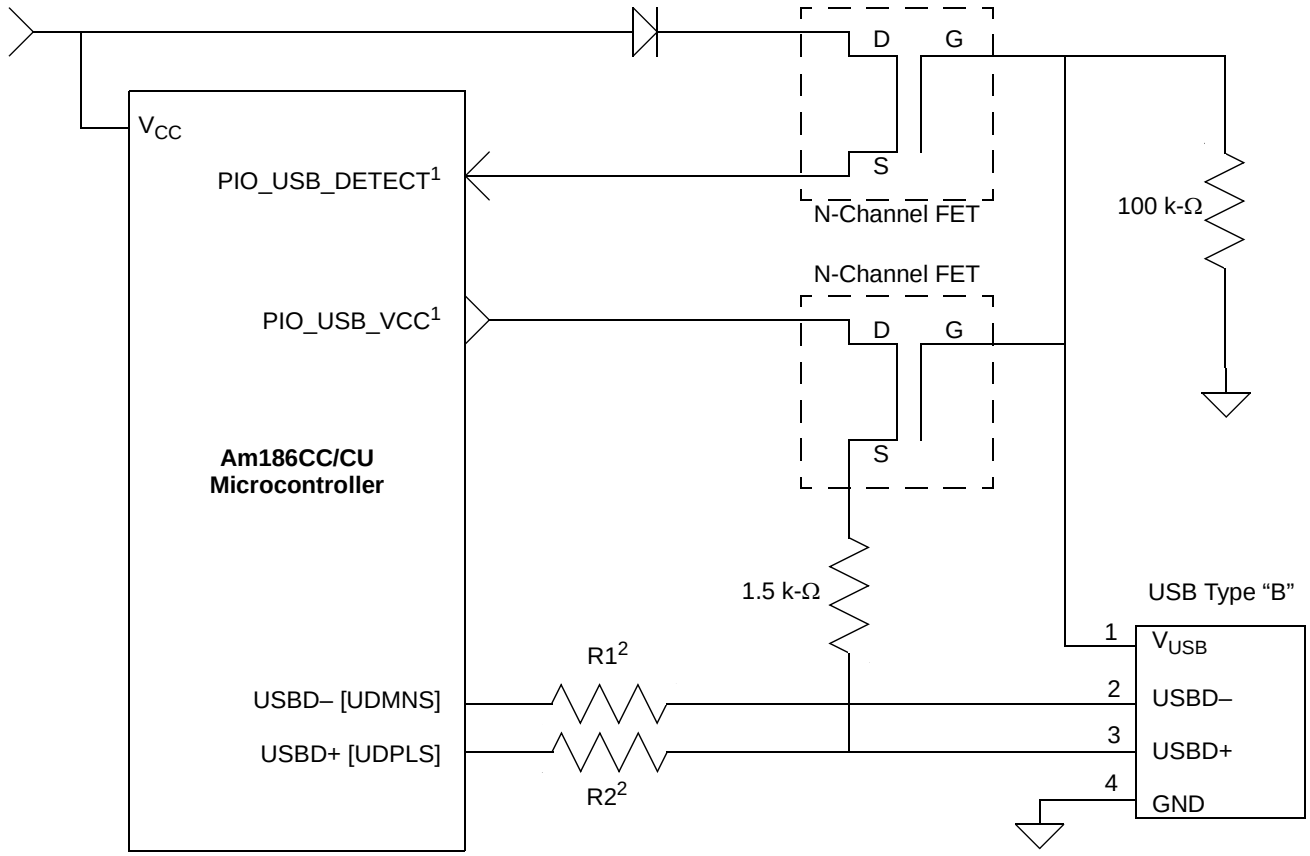
Change Figure 17-3 on page 17-8 to the following. The new figure changes the FSC waveform.

Figure 17-3 GCI Terminal Mode Frame Structure



Change Figure 18-2 on page 18-4 to the following. The new figure adds a 100 k-Ω pulldown resistor to V_{USB} and a note to clarify PIO use in the example.

Figure 18-2 USB With Internal Transceiver

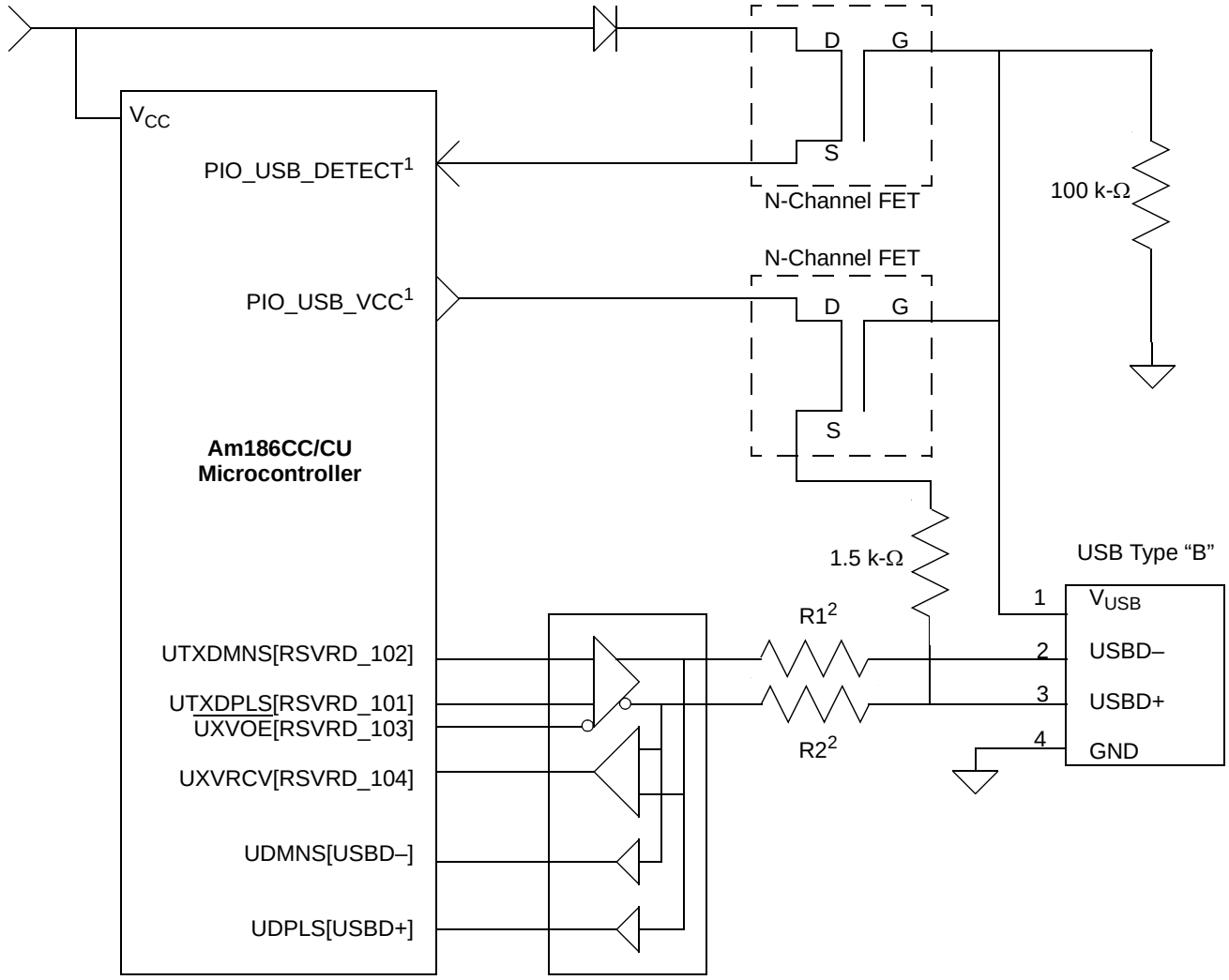


Notes:

1. *For the PIO_USB_DETECT and PIO_USB_VCC signals, select PIO pins that default to input operation. If a PIO with an internal pullup resistor is used, add a 10 k-Ω external pulldown resister to override the internal 50 k-Ω pullup. See Table 9-1 on page 9-3 [of the manual] for PIO signal defaults.*
2. *The USB specification requires a driver impedance between 29 Ω and 44 Ω on the USBD+ and USBD– signals. For information about driver characteristics and selecting a series resistor value, see the data sheets for the Am186CC and Am186CU microcontrollers.*

Change Figure 18-3 on page 18-5 to the following. The new figure adds a 100 k-Ω pulldown resistor to V_{USB} and a note to clarify PIO use in the example.

Figure 18-3 USB With External Transceiver



Notes:

1. *For the PIO_USB_DETECT and PIO_USB_VCC signals, select PIO pins that default to input operation. If a PIO with an internal pullup resistor is used, add a 10 k-Ω external pulldown resister to override the internal 50 k-Ω pullup. See Table 9-1 on page 9-3 [of the manual] for PIO signal defaults.*
2. *The USB specification requires a driver impedance between 29 W and 44 W on the USBD+ and USBD– signals. For information about driver characteristics and selecting a series resistor value, see the documentation for the external transceiver.*

INDEX

Numerics

16-bit memory, and SmartDMA channel, 4

A

ABAUD bit, 6
ABDONE bit, 6, 9
asynchronous serial port (UART), 4–7
autobaud
  detection, 6
  enhancement, 7, 9

B

break-immediate bit, 4, 7, 9
BRKIM bit, 4, 7, 9

C

changed figures, 10
clearing an edge-triggered interrupt request, 2
CTS signal, 6

D

deposit cycle, DMA transfers, 3
DMA
  request sources, 2–3
  transfers, figure, 3

E

edge-triggered interrupt request, clearing, 2
ellipsis in changed text, 1

F

FER bit, 7
fetch cycle, DMA transfers, 3
FIFO and immediate status bits, 4, 7
figures, changed, 10
frame structure figure, GCI terminal mode, 8, 10
FSC signal waveform figure, 8, 10

G

GCI terminal mode frame structure figure, 8, 10

H

HSPABx registers, 7, 9
HSPIMSK register, 9
HSPSTAT register, 5–6, 9

I

immediate status bits, 4, 7, 9
initialization and reset, 2
Interrupt Request (REQST) register, clearing bits, 2
interrupt sources table, 7

N

no-connect pins, 2

O

OERIM bit, 4
overrun error immediate bit, 4

P

pin states, latched on reset, 2
pinstrap, {$\overline{\text{USBXCVR}}$}, 2
PIO pullup or pulldown resistor, 5
pulldown resistor
  PIO, 5
  V_{USB} signal, 8, 11–12
pullup resistor
  PIO, 5
  USBD+ signal, 8

R

REQST register, 2
reserved pins, 2
reset, watchdog-timer-generated, 2
resistor, pulldown, V_{USB} signal, 8, 11–12
RESOUT signal, 5
RSVD_xx pins, 2
RTR/CTS protocol, 6

S

serial port, asynchronous (UART), 4–7
SmartDMA channel, 4–5
software, unsynchronized DMA, 2–3
source synchronized DMA transfers figure, 3
spurious interrupt, clearing, 2
square brackets in changed text, 1
synchronized DMA, 2–3

U

unsynchronized DMA, 2–3
USB with internal transceiver, figure, 8, 11–12
{$\overline{\text{USBXCVR}}$} pinstrap, 2

V

V_{USB} signal, 8, 11–12

W

watchdog-timer-generated reset, 2