Features

- **Compatible with an Embedded ARM7TDMI™ Processor**
- **Programmable Baud Rate Generator**
- **Parity, Framing and Overrun Error Detection**
- **Line Break Generation and Detection**
- **Automatic Echo, Local Loopback and Remote Loopback Channel Modes**
- **Multi-drop Mode: Address Detection and Generation**
- **Interrupt Generation**
- **5-, 6-, 7-, 8- and 9-bit Character Length**
- **Protocol ISO7816 T = 0 and T = 1**
- **Modem, Handshaking (Hardware) and RS485 Signals**
- **Fully Scan Testable (up to 98%)**
- **Two Dedicated Peripheral Data Controller Channels Can be Easily Implemented**
- **Can be Directly Connected to the Atmel Implementation of the AMBA™ Peripheral Bus (APB)**

Description

The two-channel, full-duplex USART2 features parity, framing and overrun error detection. A baud rate generator provides the bit period clock, named the Baud Rate Clock, to both the receiver and the transmitter. The USART2 can be programmed to operate in five different modes: normal, modem, handshaking, RS485, and ISO7816 (T = 0 and T = 1). In normal mode, three different test configurations are available: automatic echo, local loopback, and remote loopback.

Two dedicated Peripheral Data Controller channels can be easily implemented. One is dedicated to the receiver. The other is dedicated to the transmitter. They can be connected to either the PDC or PDC2 block.

The generation of interrupts is controlled in the status register by asserting the corresponding interrupt line.

The USART2 can be used with any 32-bit microcontroller core if the timing diagram shown in Figure 5 on page 8 is respected. When using an ARM7TDMI as the core, the Atmel Bridge must be used to provide the correct bus interface to the peripheral.

32-bit Embedded ASIC Core Peripheral

USART2

Rev. 1733C–CASIC03/02

Figure 1. USART2 Symbol

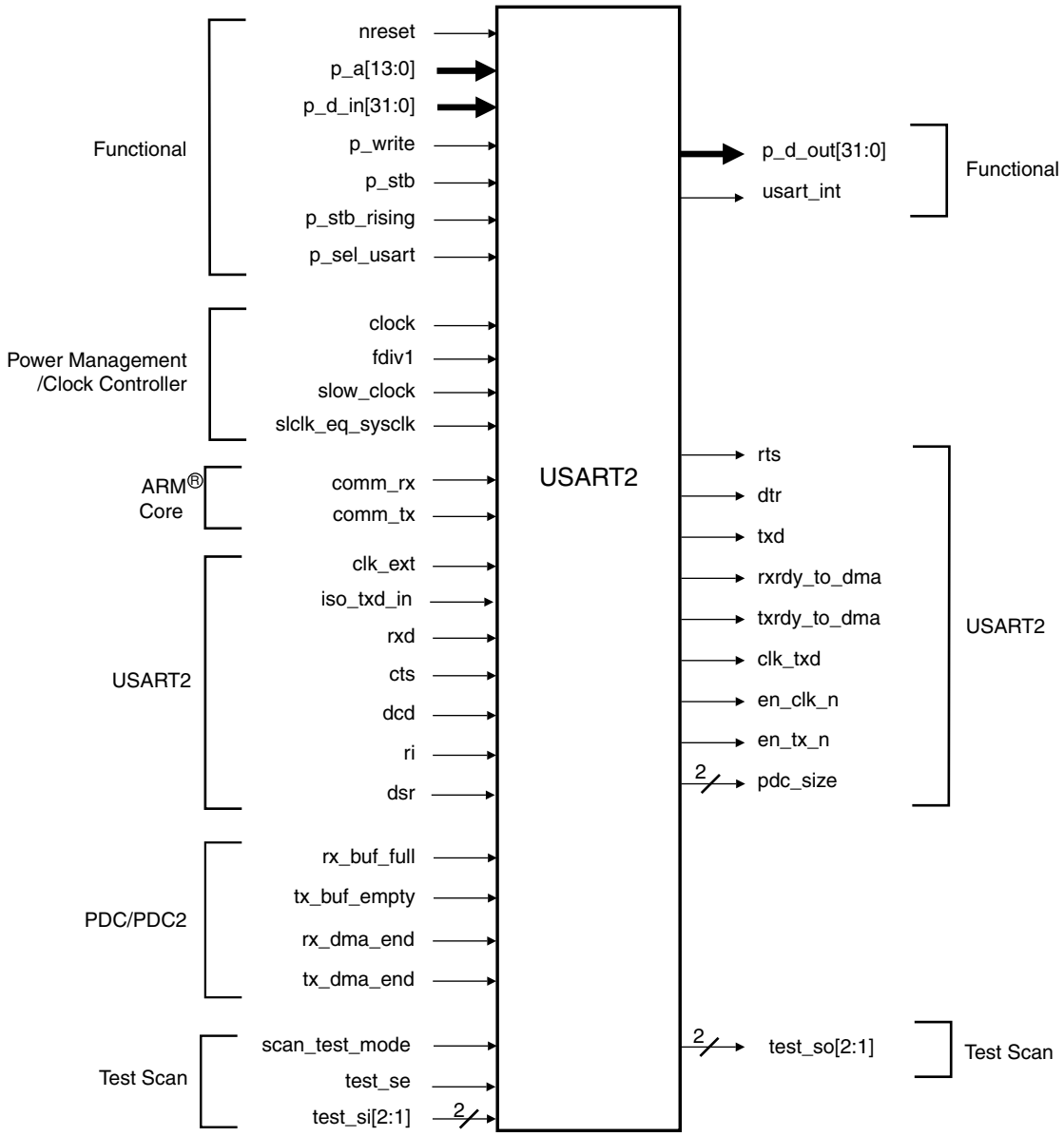


Table 1. USART2 Pin Description

Name	Function	Type	Active Level	Comments
Functional				
nreset	Reset System	Input	Low	Resets all the counters and signals
p_a[13:0]	Address Bus	Input	–	The address takes into account the 7 LSBS [1:0], but the macrocell does not take these bits into account (left unconnected).
p_d_in[31:0]	Input Data Bus	Input	–	From host (bridge)
p_d_out[31:0]	Output Data Bus	Output	–	To host (bridge)
p_write	Write Enable	Input	High	From host (bridge)
p_stb	Peripheral Strobe	Input	High	To host (bridge)
p_stb_rising	User Interface Check Signal	Input	–	From host (bridge), Clock for all DFFs controlling the configuration registers
p_sel_usart	Selection of the Block	Input	High	From host (bridge)
usart_int	Interrupt Signal to AIC	Output	High	
Power Management/Clock Controller				
clock	System Clock	Input	–	System clock for the USART2 output wave forms
fdiv1	USART2 Clock Enable	Input	–	System Clock (clock) divided
slow_clock	ARM® Core Operation	Input	–	
slclk_eq_sysclk	ARM Core Operation	Input	–	
ARM Core				
comm_rx	ARM Core Operation	Input	High	Must be connected to ARM Core
comm_tx	ARM Core Operation	Input	High	Must be connected to ARM Core
USART2				
clk_ext	Baud rate signal	Input	–	From SCK pad
iso_txd_in	Feedback from TXD pad for ISO7816 line	Input	–	Must be connected to TXD pad feedback when using ISO functions
rxd	Receive serial data pin	Input	–	
txd	Transmit serial data pin	Output	–	
rxrdy_to_dma	Output signal to DMA channel	Output	High	Byte available in the Receiver Holding Register (RHR). This signal connects to the PDC/PDC2[1]
txrdy_to_dma	Output signal to the DMA channel	Output	High	There are no more characters in the Transmitter Holding Register (THR). This signal connects to the PDC/PDC2[1]
clk_txd	Output of the baud rate generator	Output	–	To SCK pad
en_clk_n	Direction signal for SCK pad	Output	–	Active in synchronous mode

Table 1. USART2 Pin Description (Continued)

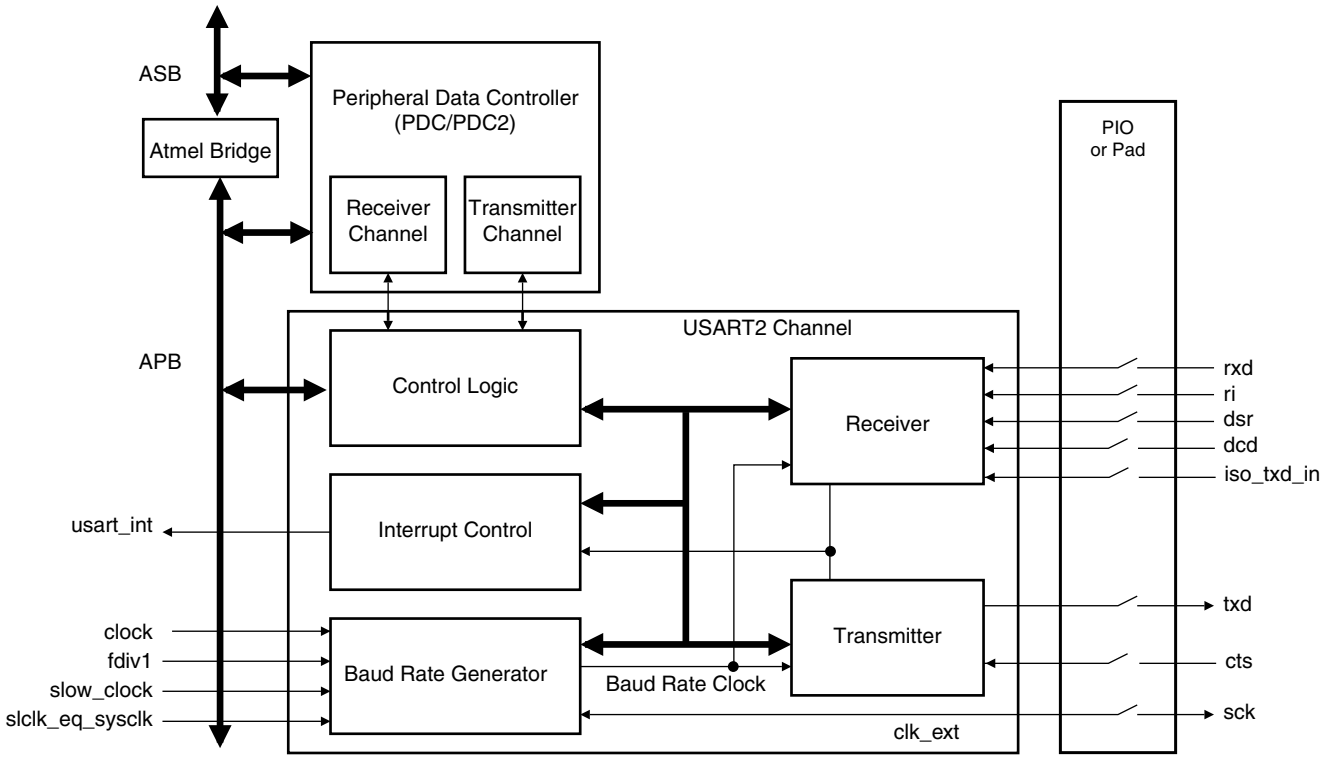
Name	Function	Type	Active Level	Comments
en_tx_n	Direction signal for SCIC pad	Output	–	
rts	Request to Send	Output	Low	Used in Handshaking, Modem and RS485 modes
dtr	Data Terminal Ready	Output	Low	Only used in Modem mode
cts	Clear to Send	Input	Low	Used in Handshaking and Modem modes
dcd	Data Carrier Detect	Input	Low	Only used in Modem mode
ri	Ring Indicator	Input	Low	Only used in Modem mode
dsr	Data Set Ready	Input	Low	Only used in Modem mode
pdc_size[1:0]	Size of transfer	Output	–	Connected to PDC2
PDC/PDC2				
rx_buf_full	Input signal from DMA channel	Input	High	Generated by PDC2
tx_buf_empty	Input signal from DMA channel	Input	High	Generated by PDC2
rx_dma_end	End of receive DMA transfer	Input	High	Generated by PDC/PDC2(1)
tx_dma_end	End of transmit DMA transfer	Input	High	Generated by PDC/PDC2(1)
Test Scan				
scan_test_mode	Must be set when running the scan vectors	Input	High	
test_se	Scan test enable	Input	High/Low	Scan shift/scan capture
test_si[2:1]	Scan test input	Input	High	Entry of scan chain
test_so[2:1]	Scan test output	Output	–	Output of scan chain

Note: 1. The Peripheral Data Controllers (PDC and PDC2) are separate blocks. Please refer to the corresponding datasheets.

Scan Test Configuration

The fault coverage is maximum if all non-scan inputs can be controlled and all non-scan outputs can be observed. In order to achieve this, the ATPG vectors must be generated on the entire circuit (top-level), which includes the USART2, or all USART2 I/Os (must have a top-level access) and ATPG vectors must be applied to these pins.

Figure 2. USART2 Block Diagram

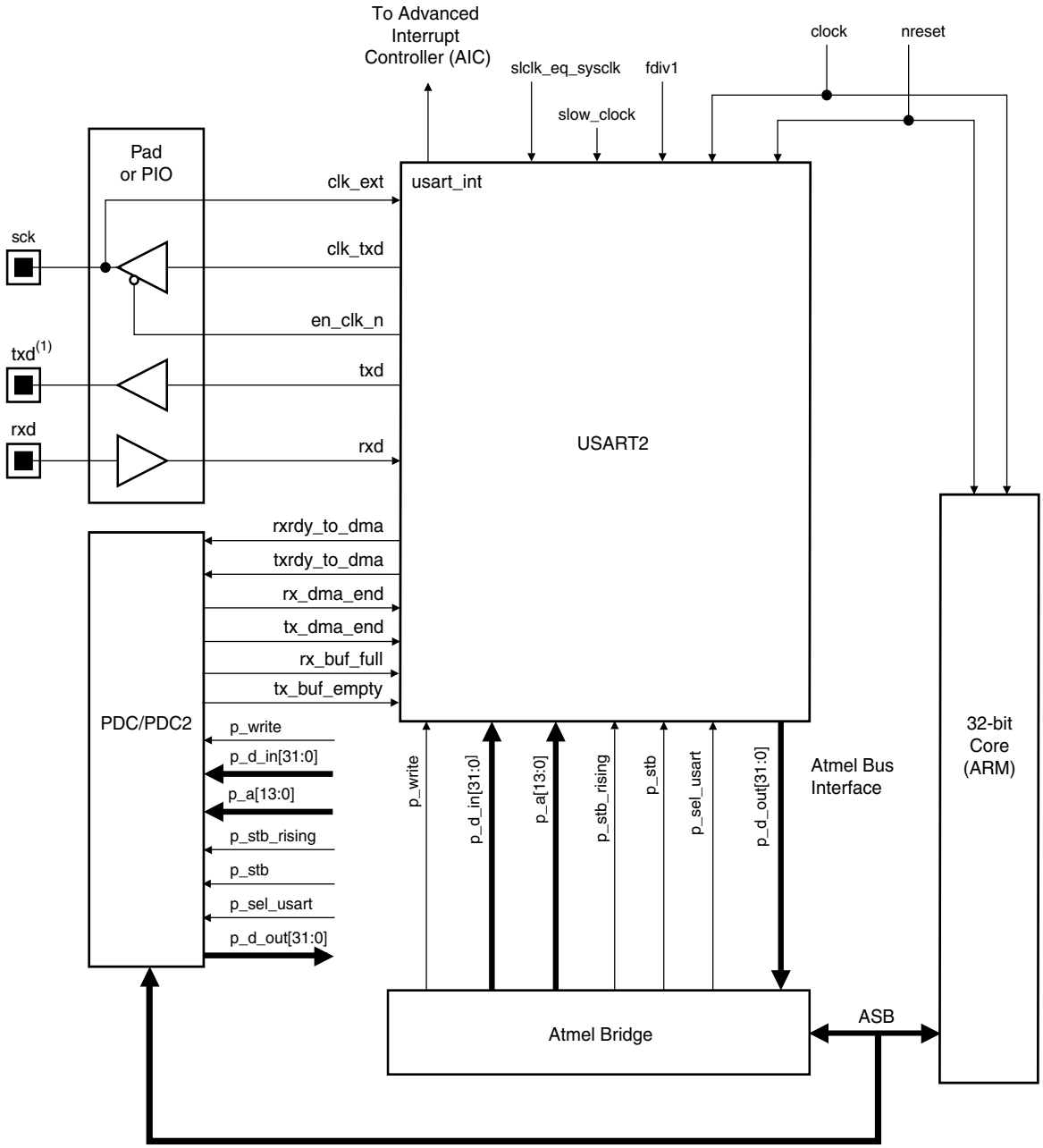


Peripheral Data Controller (PDC/PDC2)

When the dedicated Atmel PDC/PDC2 is used, four additional registers are available in the USART2 (see page 24). These registers are physically located in the PDC/PDC2 and accessed when selecting the USART2. For more details concerning these registers, please refer to the PDC and PDC2 datasheets.

The following pins are exclusively reserved for use with the PDC/PDC2: rxrdy_to_dma, txrdy_to_dma, rx_dma_end, tx_dma_end, rx_buf_full and tx_buf_empty. If the PDC/PDC2 is not used, rx_dma_end, tx_dma_end, rx_buf_full and tx_buf_empty must be tied to zero.

Figure 3. Connecting the USART2 to an ARM-based Microcontroller



Pin Description

Each USART2 channel has the following external signals:

Name	Description
sck	USART2 Serial clock can be configured as input or output: sck is configured as input if an external clock is selected (USCLKS = 11) sck is driven as output if the external clock is disabled (USCLKS[1] = 0) and clock output is enabled (CLKO = 1)
txd	Transmit Serial Data is an output
rxd	Receive Serial Data is an input
ri	Ring Indicator is an input used only in modem mode.
dsr	Data Set Ready is an input used only in modem mode.
dcd	Data Carrier Detect is an input used only in modem mode.
cts	Clear to Send is an input used in handshaking and modem modes.

Timing Diagrams

Figure 4. USART2 Timing Diagram: Write/Read Cycle

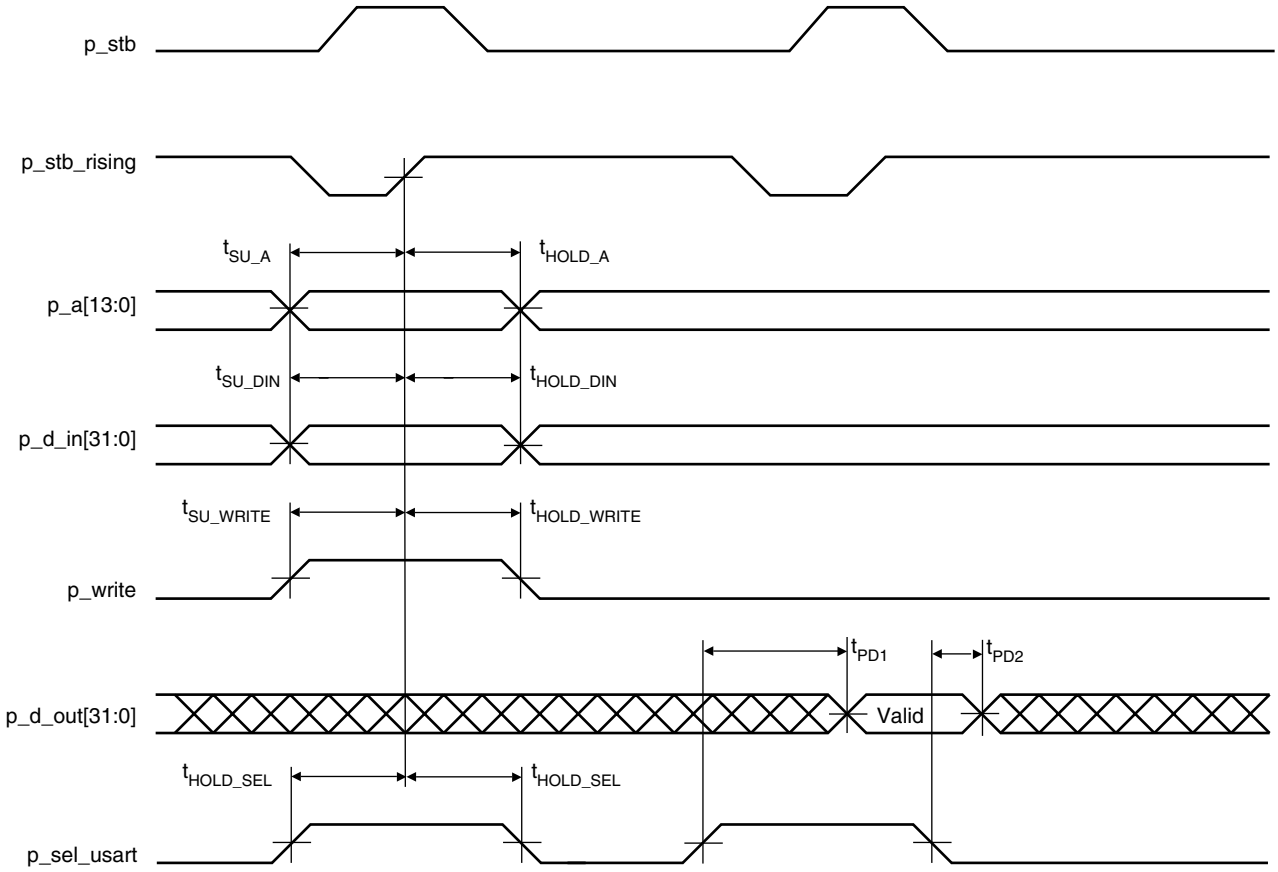
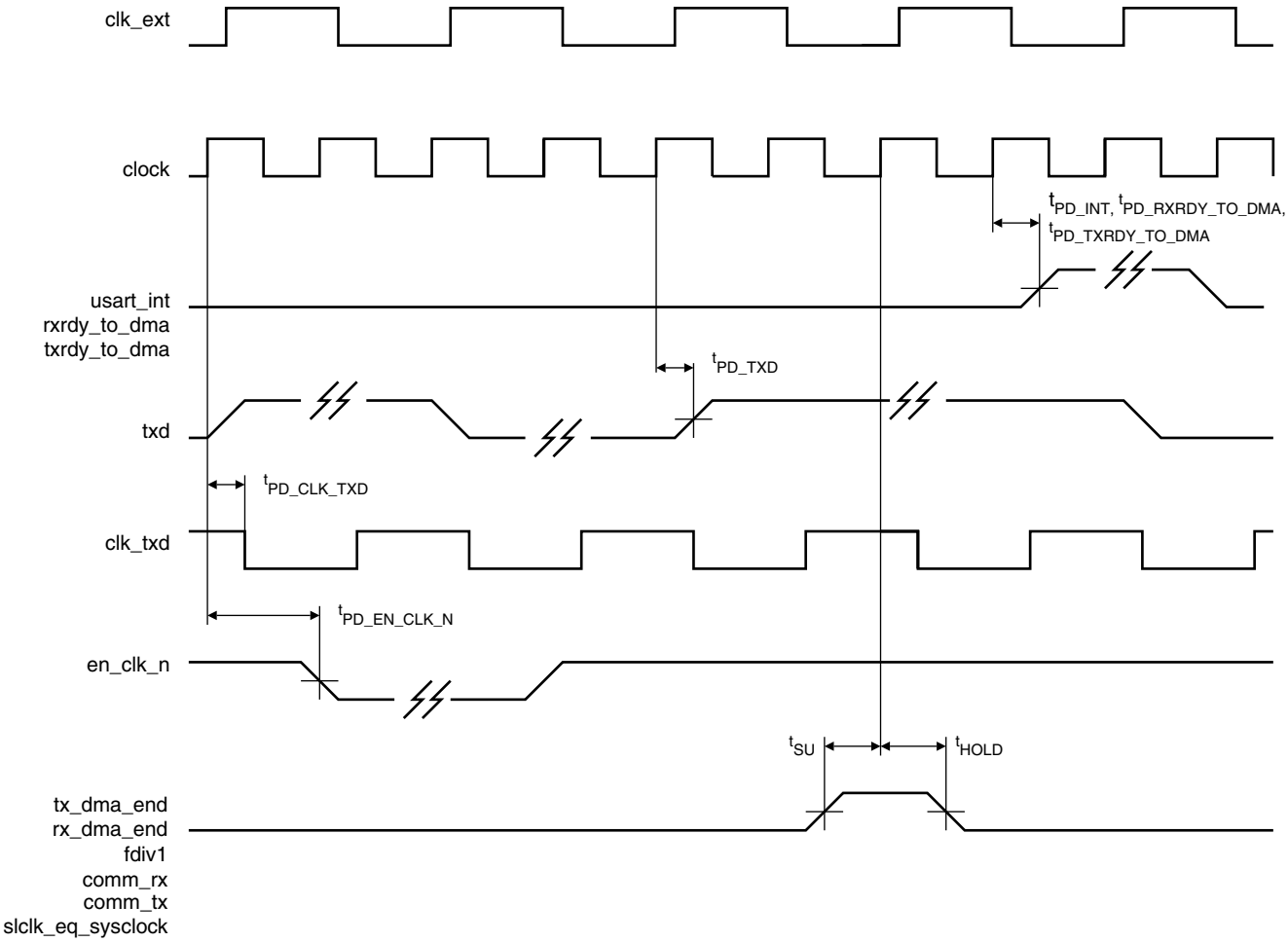


Figure 5. USART2 Timing Diagram: Propagation Delays, Control Signals



Baud Rate Generator

The Baud Rate Generator provides the bit period clock named the Baud Rate Clock to both the receiver and the transmitter.

The Baud Rate Generator can select between external and internal clock sources. The external clock source is sck (clk_ext) or slow_clock. The internal clock sources can be either the master clock (clock) or the master clock divided (fdiv1).

Note: In all cases, if an external clock is used, the duration of each of its levels must be longer than the system clock (clock) period. The external clock frequency must be at least 4.5 times lower than the system clock.

When the USART2 is programmed to operate in asynchronous mode (SYNC = 0 in the Mode Register US_MR), the selected clock is divided by 16, or, 8 times the value (CD) written in US_BRGR (Baud Rate Generator Register), depending on the value of the OVER bit in US_MR. Furthermore, if US_BRGR is set to 0, the Baud Rate Clock is disabled.

$$\text{Baud Rate} = \frac{\text{Selected Clock}}{16 \times \text{CD}} \quad \text{or} \quad \text{Baud Rate} = \frac{\text{Selected Clock}}{8 \times \text{CD}}$$

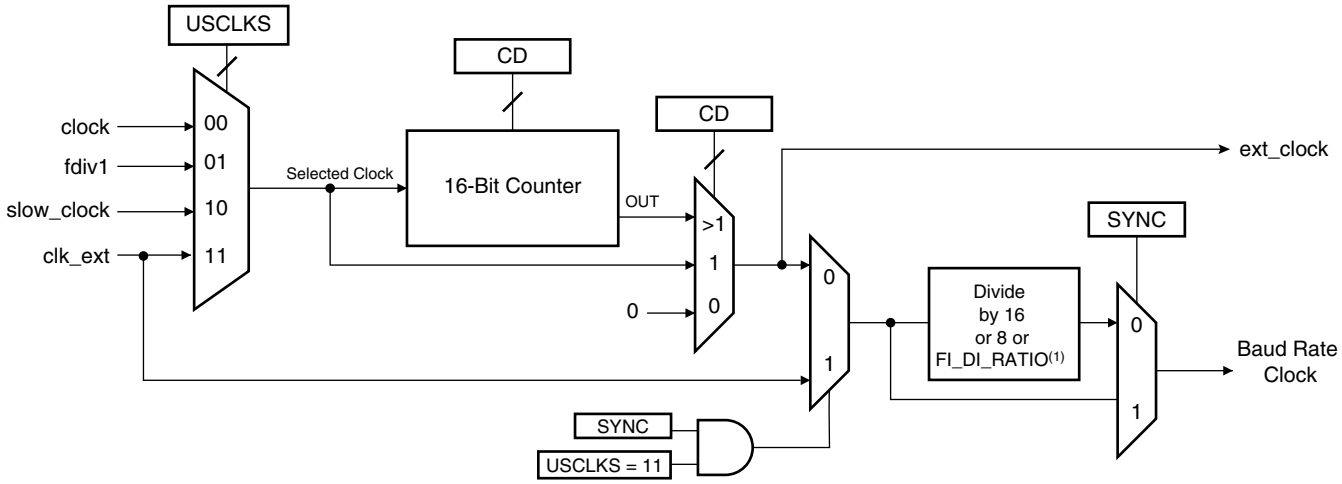
When the USART2 is programmed to operate in synchronous mode (SYNC = 1) and the selected clock is internal (USCLKS[1] = 0 in the Mode Register US_MR), the Baud Rate Clock is the internal selected clock divided by the value written in US_BRGR. If US_BRGR is set to 0, the Baud Rate Clock is disabled.

$$\text{Baud Rate} = \frac{\text{Selected Clock}}{\text{CD}}$$

In synchronous mode with external clock selected (USCLKS = 11), the clock is provided directly by the signal on the sck pin (clk_ext). No division is active. The value written in US_BRGR has no effect.

When the USART2 is programmed to operate in IS07816 mode (USART2_MODE field in the Mode Register US_MR), the selected clock is divided by the value (FI_DI_RATIO) written in US_FIDI. If US_FIDI is set to 0, the Baud Rate Clock is disabled.

Figure 6. Baud Rate Generator



Note: 1. Divide by 16 or 8 or FI_DI_RATIO depends on mode.

Receiver

Asynchronous Receiver

The USART2 is configured for asynchronous operation when SYNC = 0 (bit 7 of US_MR). In asynchronous mode, the USART2 detects the start of a received character by sampling the rxd signal until it detects a valid start bit. A low level (space) on rxd is interpreted as a valid start bit if it is detected for more than 7 cycles of the sampling clock, which is 16 times the baud rate. Hence, a space that is longer than 7/16 of the bit period is detected as a valid start bit. A space which is 7/16 of a bit period or shorter is ignored and the receiver continues to wait for a valid start bit.

When a valid start bit has been detected, the receiver samples the rxd at the theoretical mid-point of each bit. It is assumed that each bit lasts 16 cycles of the sampling clock (1-bit period) so the sampling point is eight cycles (0.5 bit period) after the start of the bit. The first sampling point is therefore 24 cycles (1.5 bit periods) after the falling edge of the start bit was detected. Each subsequent bit is sampled 16 cycles (1-bit period) after the previous one.

Figure 7. Asynchronous Mode: Start Bit Detection

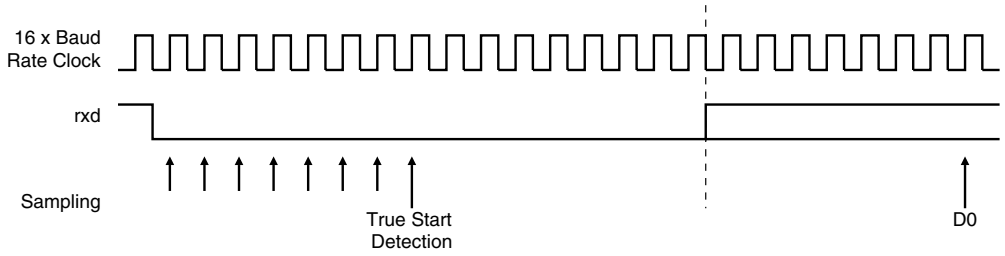
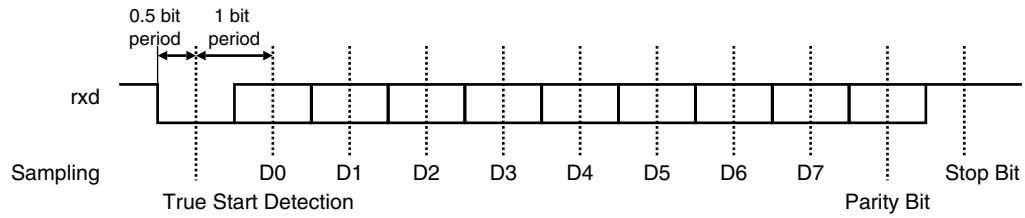


Figure 8. Asynchronous Mode: Character Reception

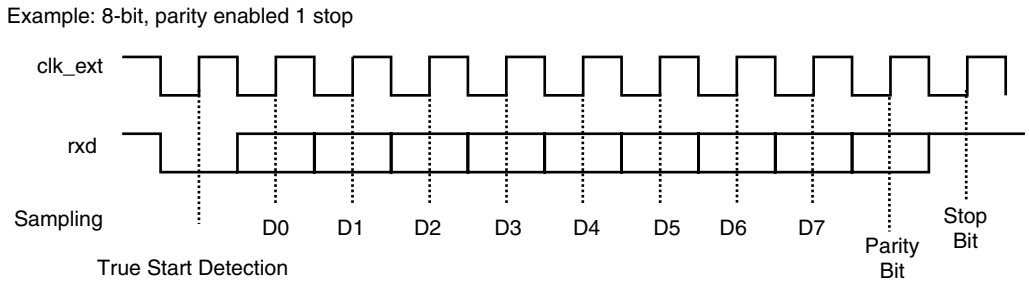
Example: 8-bit, parity enabled 1 stop



Synchronous Receiver

When configured for synchronous operation (SYNC = 1), the receiver samples the rxd signal on each rising edge of the Baud Rate Clock. If a low level is detected, it is considered a start. The data bits, parity bit and stop bit are sampled and the receiver waits for the next start bit. See the example in Figure 9.

Figure 9. Synchronous Mode: Character Reception



Receiver Ready

When a complete character is received, it is transferred to the US_RHR and the RXRDY status bit in US_CSR is set. If US_RHR has not been read since the last transfer, the OVRE status bit in US_CSR is set.

Parity Error

Each time a character is received, the receiver calculates the parity of the received data bits, in accordance with the field PAR in US_MR. It then compares the result with the received parity bit. If different, the parity error bit PARE in US_CSR is set.

Framing Error

If a character is received with a stop bit at low level and with at least one data bit at high level, a framing error is generated. This sets FRAME in US_CSR.

Time-out

This function allows an idle condition on the rxd line to be detected. The maximum delay for which the USART2 should wait for a new character to arrive while the rxd line is inactive (high level) is programmed in US_RTOR (Receiver Time-out). When this register is set to 0, no time-out is detected. Otherwise, the receiver waits for a first character and then initializes a counter, which is decremented at each bit period and reloaded at each byte reception. When the counter reaches 0, the TIMEOUT bit in US_CSR is set. The user can restart the wait for a first character with the STTTO (Start Time-out) bit in US_CR.

Calculation of time-out duration:

$$\text{Duration} = \text{US\_RTOR Value} \times \text{Bit Period}$$

Generating CLK_TXD

In synchronous mode, clk_txd is the clock as defined in Figure 9.

In asynchronous mode the clock is defined below:

$$\text{clk\_txd} = 16 \times \text{Baud Rate Clock}$$

or

$$\text{clk\_txd} = 8 \times \text{Baud Rate Clock}$$

depending on the OVER bit in US_MR register,

or

$$\text{clk\_txd} = \text{FI\_DI\_RATIO} \times \text{Baud Rate Clock}$$ in ISO7816 mode.

Transmitter

The transmitter has the same behavior in both synchronous and asynchronous operating modes. The start bit, data bits, parity bit and stop bits are serially shifted, lowest significant bit first, on the falling edge of the serial clock. See the example in Figure 10.

The number of data bits is selected in the CHRL field in US_MR.

The parity bit is set according to the PAR field in US_MR.

The number of stop bits is selected in the NBSTOP field in US_MR.

When a character is written to US_THR (Transmit Holding), it is transferred to the Shift Register as soon as it is empty. When the transfer occurs, the TXRDY bit in US_CSR is set until a new character is written to US_THR. If Transmit Shift Register and US_THR are both empty, the TXEMPTY bit in US_CSR is set.

Time-guard

The time-guard function allows the transmitter to insert an idle state on the txd line between two characters. The duration of the idle state is programmed in US_TTGR (Transmitter Time-guard). When this register is set to zero, no time-guard is generated. Otherwise, the transmitter holds a high level on txd after each transmitted byte during the number of bit periods programmed in US_TTGR:

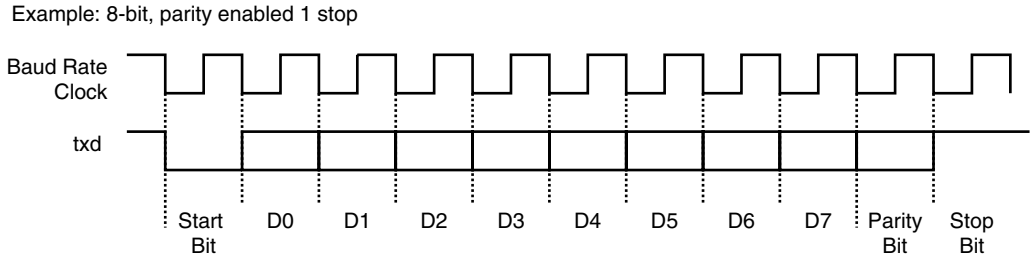
$$\text{Idle State Duration between Two Characters} = \text{Time-guard Value} \times \text{Bit Period}$$

Multi-drop Mode

When the field PAR in US_MR equals 11X (binary value), the USART2 is configured to run in multi-drop mode. In this case, the parity error bit PARE in US_CSR is set when data is detected with a parity bit set to identify an address byte. PARE is cleared with the Reset Status Bits command (RSTSTA) in US_CR. If the parity bit is detected low, identifying a data byte, PARE is not set.

The transmitter sends an address byte (parity bit set) when a Send Address command (SENDA) is written to US_CR. In this case, the next byte written to US_THR will be transmitted as an address. After this, any byte transmitted will have the parity bit cleared.

Figure 10. Synchronous and Asynchronous Modes: Character Transmission



Break

A break condition is a low signal level that has a duration of at least one character (including start/stop bits and parity).

Transmit Break

The transmitter generates a break condition on the txd line when STTBRK is set in US_CR (Control Register). In this case, the character present in the Transmit Shift Register is completed before the line is held low.

To cancel a break condition on the txd line, the STPBRK command in US_CR must be set. The USART2 completes a minimum break duration of one character length. The txd line then returns to high level (idle state) for at least 12-bit periods, or the value of the Time-guard register if it is greater than 12, to ensure that the end of break is correctly detected. Then the transmitter resumes normal operation.

The BREAK is managed like a character:

- The STTBRK and the STPBRK commands are performed only if the transmitter is ready (bit TXRDY = 1 in US_CSR).
- The STTBRK command blocks the transmitter holding register (bit TXRDY is cleared in US_CSR) until the break has started.
- A break is started when the Shift Register is empty (any previous character is fully transmitted). US_CSR.TXEMPTY is cleared. The break blocks the transmitter shift register until it is completed (high level for at least 12-bit periods after the STPBRK command is requested).

In order to avoid unpredictable states:

- STTBRK and STPBRK commands must not be requested at the same time.
- Once an STTBRK command is requested, further STTBRK commands are ignored until the BREAK is ended (high level for at least 12-bit periods).
- All STPBRK commands requested without a previous STTBRK command are ignored.
- A byte written into the Transmit Holding Register while a break is pending but not started (bit TXRDY = 0 in US_CSR) is ignored.
- It is *not permitted* to write new data in the Transmit Holding Register while a break is in progress (STPBRK has not been requested), even though TXRDY = 1 in US_CSR.
- A new STTBRK command *must not* be issued until an existing break has ended (TXEMPTY=1 in US_CSR).

The standard break transmission sequence is:

1. Wait for the transmitter ready.
   (US_CSR.TXRDY = 1)
2. Send the STTBRK command.
   (Write 0x0200 to US_CR)
3. Wait for the transmitter ready.
   (bit TXRDY = 1 in US_CSR)
4. Send the STPBRK command.
   (Write 0x0400 to US_CR)
5. The next byte can then be sent:
6. Wait for the transmitter ready.
   (bit TXRDY = 1 in US_CSR)
7. Send the next byte.
   (Write byte to US_THR)

Each of these steps can be scheduled by using the interrupt if the bit TXRDY in US_IMR is set.

For character transmission, the USART2 channel must be enabled before sending a break.

Receive Break

The receiver detects a break condition when all data, parity and stop bits are low. When the low stop bit is detected, the receiver asserts the RXBRK bit in US_CSR. An end of receive break is detected by a high level for at least 2/16 of a bit period in asynchronous operating mode or at least one sample in synchronous operating mode. RXBRK is also asserted when an end-of-break is detected.

Both the beginning and the end of a break can be detected by interrupt if the bit US_IMR.RXBRK is set.

Peripheral Data Controller Channels (PDC/PDC2)

Each USART2 channel is closely connected to a corresponding Peripheral Data Controller channel (either the PDC or the PDC2). One is dedicated to the receiver. The other is dedicated to the transmitter.

The PDC/PDC2 channel is programmed using US_TPR (Transmit Pointer) and US_TCR (Transmit Counter) for the transmitter and US_RPR (Receive Pointer) and US_RCR (Receive Counter) for the receiver. The status of the PDC/PDC2 is given in US_CSR by the ENDTX bit for the transmitter and by the ENDRX bit for the receiver.

PDC2 = PDC + one additional buffer. When the first buffer is full, the PDC2 buffers may filled, thereby permitting the continued transfer of data.

The pointer registers (US_TPR and US_RPR) are used to store the address of the transmit or receive buffers. The counter registers (US_TCR and US_RCR) are used to store the size of these buffers.

The receiver data transfer is triggered by the RXRDY bit and the transmitter data transfer is triggered by TXRDY. When a transfer is performed, the counter is decremented and the pointer is incremented. When the counter reaches 0, the status bit is set (ENDRX for the receiver, ENDTX for the transmitter in US_CSR) and can be programmed to generate an interrupt. While the counter is at 0, the status bit is asserted and transfers are disabled.

Interrupt Generation

Each status bit in US_CSR has a corresponding bit in US_IER (Interrupt Enable) and US_IDR (Interrupt Disable), which controls the generation of interrupts by asserting the USART2 interrupt line connected to the Advanced Interrupt Controller. US_IMR (Interrupt Mask Register) indicates the status of the corresponding bits.

When a bit is set in US_CSR and the same bit is set in US_IMR, the interrupt line is asserted.

Channel Test Configurations

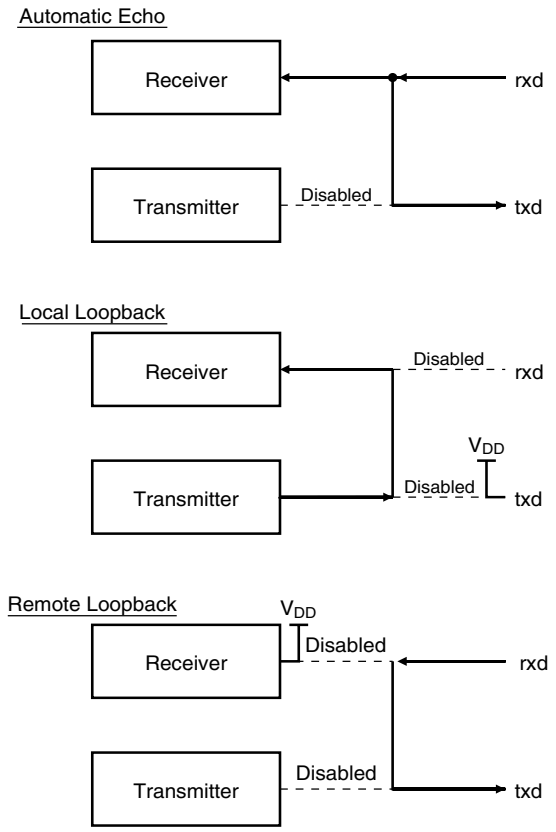
The USART2 can be programmed to operate in three different test modes, using the field CHMODE in US_MR.

Automatic echo mode allows bit-by-bit retransmission. When a bit is received on the rxd line, it is sent to the txd line. Programming the transmitter has no effect.

Local loopback mode allows the transmitted characters to be received. txd and rxd pins are not used and the output of the transmitter is internally connected to the input of the receiver. The rxd pin level has no effect and the txd pin is held high, as in idle state.

Remote loopback mode directly connects the rxd pin to the txd pin. The Transmitter and the Receiver are disabled and have no effect. This mode allows bit-by-bit retransmission.

Figure 11. Channel Test Configurations



Note: In ISO mode, the channel test configurations are not functional.

IS07816 Mode

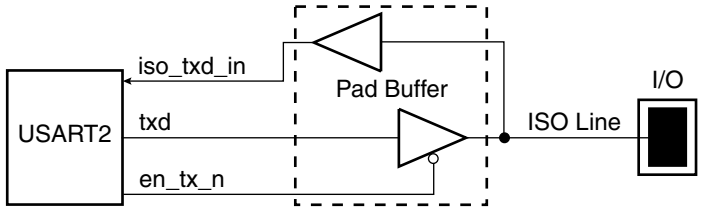
The USART2 handles all specific requirements defined in ISO7816 T = 0 and T = 1 protocol types. To select ISO7816 mode, the protocol and master or slave field USART2_MODE in the Mode Register (US_MR) has to be set correctly depending on the desired configuration, as described in Table 7 on page 24.

Note: The channel test configuration is not available in ISO mode.

The IS07816 USART2 is configured and controlled via several registers:

- USART2 Mode Register (US_MR)
- USART2 Control Register (US_CR)
- USART2 Interrupt Enable Register (US_IER)
- USART2 Interrupt Disable Register (US_IDR)
- USART2 Interrupt Mask Register (US_IMR)
- USART2 Channel Status Register (US_CSR)
- USART2 Fi_Di_Ratio Register (US_FIDI)
- USART2 Nb Errors Register (US_NER)
- USART2 Receiver Time Out Register (US_RTOR)

Figure 12. ISO Mode Connection



Programmable Bit Rate

The bit rate is determined by the following formula:

$$B = \frac{D}{F} \times f$$

where:

B = bit rate

D = bit-rate adjustment factor

F = clock frequency division factor

f = iso clock frequency (Hz) as defined with iso clock

D is a 4-bit encoded binary value (DI) with the corresponding decimal index (Di) as represented in Table 2.

F is a 4-bit encoded binary value (FI) with the corresponding decimal index (Fi) as represented in Table 3.

Table 2. Binary and Decimal Values for D

DI (bits)	0001	0010	0011	0100	0101	0110	1000	1001
Di (decimal)	1	2	4	8	16	32	12	20

Table 3. Binary and Decimal Values for F

FI (bits)	0000	0001	0010	0011	0100	0101	0110	1001	1010	1011	1100	1101
Fi (decimal	372	372	558	744	1116	1488	1860	512	768	1024	1536	2048

Table 4. Presentation of the Possible Values for the Fi/Di Ratio

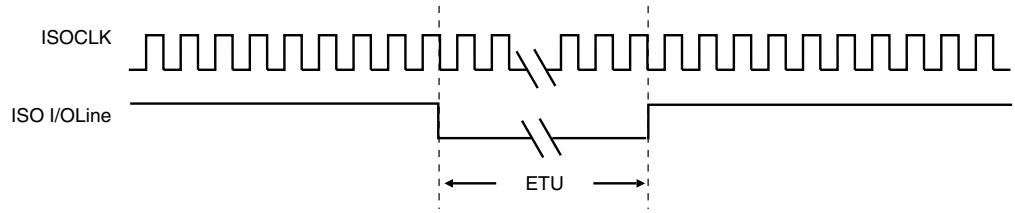
Fi/Di	372	558	774	1116	1488	1806	512	768	1024	1536	2048
1	372	558	744	1116	1488	1860	512	768	1024	1536	2048
2	186	279	372	558	744	930	256	384	512	768	1024
4	93	139.5	186	279	372	465	128	192	256	384	512
8	46.5	69.75	93	139.5	186	232.5	64	96	128	192	256
16	23.25	34.87	46.5	69.75	93	116.2	32	48	64	96	128
32	11.62	17.43	23.25	34.87	46.5	58.13	16	24	32	48	64
12	31	46.5	62	93	124	155	42.66	64	85.33	128	170.6
20	18.6	27.9	37.2	55.8	74.4	93	25.6	38.4	51.2	76.8	102.4

The Baud Rate Clock used in mode ISO7816 is configured via the field FI_DI_RATIO in US_FIDI. This is a 11-bit divider, hence the divider must be programmed between 0 and 2047. The FI_DI_RATIO value loaded in US_FIDI will define the bit rate. The FI/DI value shown in Table 4 is loaded in the US_FIDI register. For example, if an FI/DI of 372 is desired, 174 H is programmed into the US_FIDI register. Reset value for FI_DI_RATIO is 174.

Bit Time ETU

The bit duration is determined by the relation between the ISO clock and the baud rate clock divisor factor FI_DI_RATIO. As a result, in ISO7816 mode, there are FI_DI_RATIO clock cycles in one elementary time unit (ETU).

Figure 13. Bit Duration

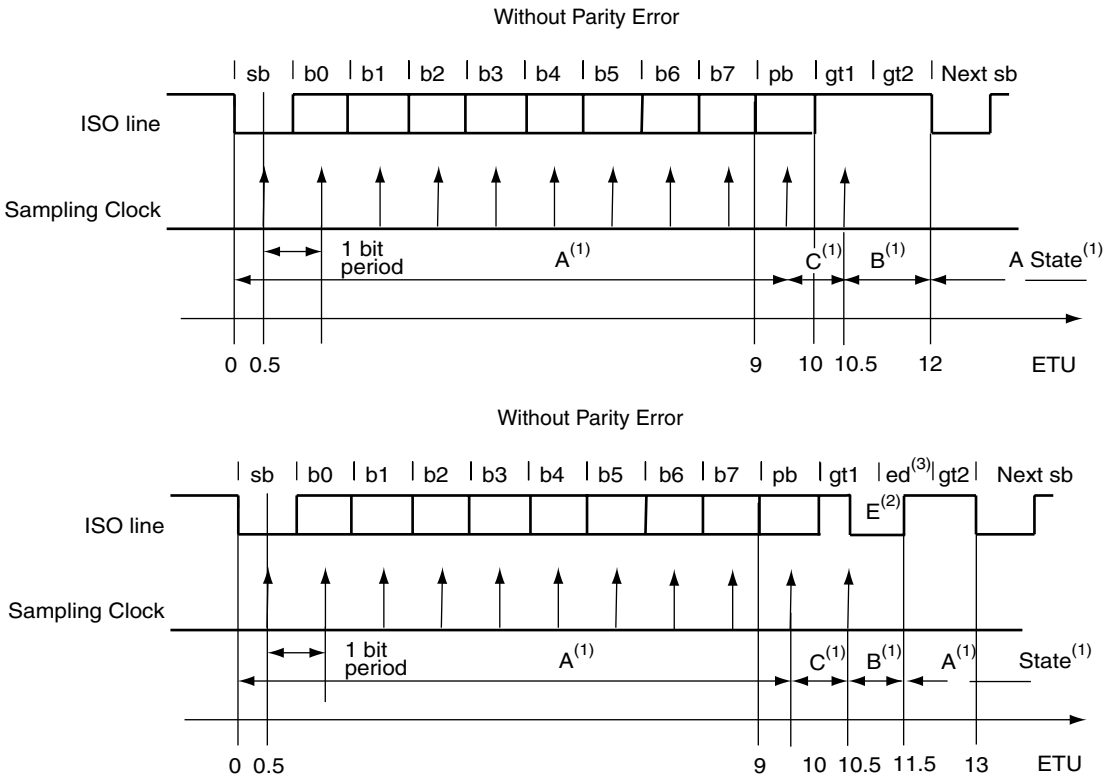


Protocols

Table 5. Description of Receiver and Transmitter Sides for Protocols T = 1 and T = 0

Protocol T = 1	
Receiver Side	Upon detection of a start bit, the following data byte and the parity bit are shifted in the Receiver Holding Register (US_RHR) when the shift is completed and the parity is checked. If a parity error is detected, the PARE bit is set in US_CSR. In this protocol there is only one stop bit.
Transmitter Side	The data is loaded in the Transmit Holding Register and parity is calculated. The data and the parity bit are shifted out following the start bit.
Protocol T = 0	
Receiver Side	Upon detection of a start bit, the following data byte and the parity bit are shifted in the Receiver Holding Register (US_RHR) when the shift is completed and the parity is checked. If a parity error is detected, a low error signal is sent for one Elementary Time Unit (ETU), 10.5 ETUs after the start bit. This error signal is sent from the receiver to the transmitter. The shift register value is not loaded in the Receiver Holding Register.
Transmitter Side	The data is loaded in the Transmit Holding Register and parity is calculated. The data and the parity bit are shifted out following the start bit. The TxD line returns to high impedance 10 ETUs after the start bit. In this protocol there are two stop bits.

Figure 14. T = 0 Protocol Without and With Parity Error



(1) State
A - Receiver Mode
B - Transit Mode
C - Check Parity
(2) E - Error Signal asserted by the receiver in receiver mode
(3) ed - Error delay (extra guard time added by the transmitter)

Table 6. Description of Supplemental Protocol T = 0 Characteristics

Additional Features in Protocol T = 0	
Error Number	If errors occurred during a transfer, it is possible to know the total amount of errors by reading the field NB_ERRORS in US_NER. This is a read only register reset by a read action. Up to 255 errors can be recorded.
Non Acknowledge Inhibited	In some cases it can be important to inhibit an error. This can be achieved by setting the INACK bit in US_MR. This means that even if an error is detected by the receiver, it will not send an error signal (NACK). If INACK is reset, on each error, the NACK bit in US_SR will be set. It can be reset by using the RSTNACK command in US_CR. Moreover, when INACK is reset and the USART2 receives a byte with a parity error, this value is available in the US_RHR register but the US_RXRDY bit will not be set.
Character Repetition	A character repetition can be executed if the MAX_ITERATION field in US_MR is different from 0. MAX_ITERATION is a 3-bit field configurable with a value between 0 and 7. This implies that a character can be repeated up to seven times. (Initial send + seven repetitions). If MAX_ITERATION is equal to zero, the protocol T = 0 transfer format explained previously is still valid. If MAX_ITERATION is different from zero and no parity error has been detected, the protocol T = 0 transfer format explained previously is still valid. If MAX_ITERATION is different from zero and a parity error has been detected, the transmitter will again send the value. If a parity error is still detected, the value is sent as many times as the value loaded in the MAX_ITERATION field. If the number of repetitions of the value reaches the value loaded in the MAX_ITERATION field, the ITERATION (US_CSR) flag is set. If at some stage during the repetition schematic, no error parity is detected by the far-end receiver, the repetition is stopped. To reset the ITERATION (US_CSR) flag, the RSIT bit must be set in the USART Control Register (US_CR).
Disable Successive NACK	In some cases it may be useful to limit the number of successive NACKs sent on the line. This can be achieved by setting the DSNACK bit in US_MR. If this bit is set, successive parity errors are counted up to the value specified in the MAX_ITERATION field. As soon as this value is reached, no additional NACK is sent on the ISO line and the flag ITERATION is asserted. If it is not set, NACK is sent on the ISO line as soon as a parity error occurs in the received character (unless INACK is set).
Bit Order	If set, MSB is sent first, otherwise LSB.

Hardware Handshaking Mode

In this mode, an additional output RTS (Request to Send) is generated. If an active RX_DMA_END is received, the receiver will force the RTS pin to a high level on the next start bit reception.

An additional input CTS (Clear to Send) is activated in this mode. If a high level signal is received on this pin, the transmitter will be disabled after finishing the current character transmission.

To select mode HANDSHAKING, the USART2_MODE field in the Mode Register (US_MR) has to be set correctly. See “USART2 Mode Register” on page 27.

Figure 15. Connections in Hardware Handshaking Mode

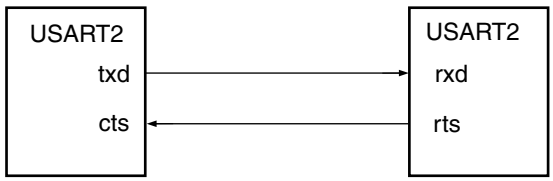
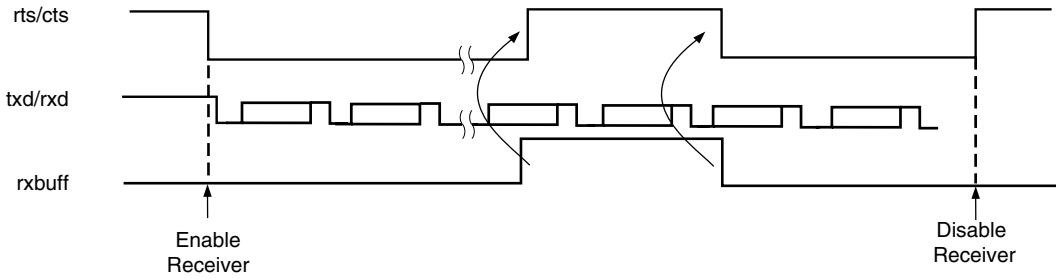


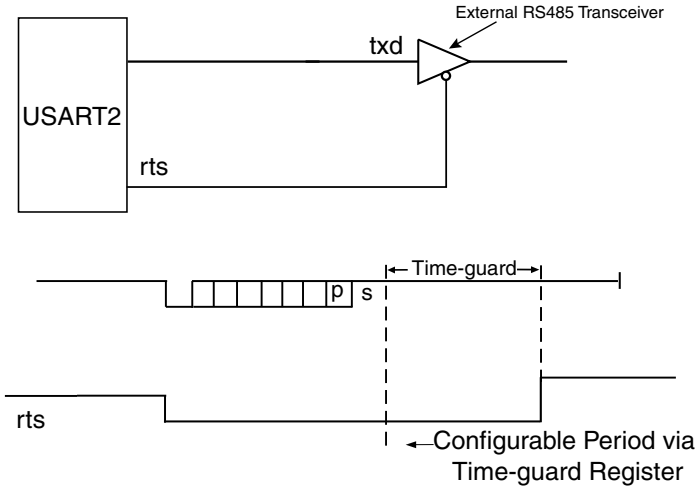
Figure 16. RTS/CTS Behavior in Hardware Handshaking Mode



RS485 Mode

In this mode, an additional output RTS (Request to Send) is generated. The behavior of RTS will follow the behavior of TXEMPTY. To select RS485 Mode, the USART2_MODE field in the Mode Register (US_MR) has to be set correctly. See “USART2 Mode Register” on page 27.

Figure 17. RTS Behavior in RS485 Mode



Modem Mode

In this mode, two additional outputs are generated:

1. Request to Send (RTS)
2. Data Terminal Ready (DTR)

If the RTS transmitter is enabled, the RTS pin will be forced to its active level (low). The behavior of the DTR pin is controlled via 2 bits (DTRDIS and DTREN) located in the Control Register (US_CR). If DTREN is set, the DTR pin will be forced to its active level (low). If DTRDIS is set, the DTR pin will be forced to its inactive level (high).

The Modem mode also activates four additional inputs:

1. Ring Indicator (RI)
2. Data Set Ready (DSR)
3. Data Carrier Detect (DCD)
4. Clear to Send (CTS)

If an input change is detected on the RI pin, the RIIC flag in the Channel Status Register (US_CSR) is raised. To reset this flag, the US_CSR register must be read.

Furthermore, if an input change is received on the DSR pin, the flag DSRIC in the Channel Status Register (US_CSR) is raised. To reset this flag, the US_CSR register must be read.

In addition, if an input change is received on the DCD pin, the DCDIC flag in the Channel Status Register (US_CSR) is raised. To reset this flag, the US_CSR register must be read.

Moreover, if an active signal is detected on the CTS pin, the transmitter will be disabled after finishing the current character transmission. If an input change is received on this pin, the CTSIC flag in the Channel Status Register (US_CSR) is raised. To reset this flag, the US_CSR register must be read.

USART2 User Interface

Table 7. USART2 Memory Map[1] [2] [3]

Offset	Register	Name	Access	Reset State
0x0000	Control Register	US_CR	Write-only	–
0x0004	Mode Register	US_MR	Read/Write	–
0x0008	Interrupt Enable Register	US_IER	Write-only	–
0x000C	Interrupt Disable Register	US_IDR	Write-only	–
0x0010	Interrupt Mask Register	US_IMR	Read-only	0
0x0014	Channel Status Register	US_CSR	Read-only	–
0x0018	Receiver Holding Register	US_RHR	Read-only	0
0x001C	Transmitter Holding Register	US_THR	Write-only	–
0x0020	Baud Rate Generator Register	US_BRGR	Read/Write	0
0x0024	Receiver Time-out Register	US_RTOR	Read/Write	0
0x0028	Transmitter Time-guard Register	US_TTGR	Read/Write	0
0x0030	Reserved for PDC/PDC2 connection	–	–	–
0x0034	Reserved for PDC/PDC2 connection	–	–	–
0x0038	Reserved for PDC/PDC2 connection	–	–	–
0x003C	Reserved for PDC/PDC2 connection	–	–	–
0x0040	FI_DI_Ratio Register	US_FIDI	Read/Write	0x174[4]
0x0044	Nb Errors Register	US_NER	Read-only	–

Notes:
1. The address takes into account the 2 LSBs [1:0], but the macrocell does not take these bits into account (left unconnected). Therefore loading 0x0001, 0x0002 or 0x0003 on P_A[13:0] addresses the Control Register.
2. In the following register description, all undefined bits (“–”) read “0”.
3. If the user selects an address that is not defined in the above table, the value of P_D_OUT[31:0] is 0x00000000.
4. The corresponding decimal value is 372.

USART2 Control Register

Name: US_CR

Access Type: Write-only

31	30	29	28	27	26	25	24
–	–	–	–	–	–	–	–
23	22	21	20	19	18	17	16
–	–	–	–	RTSDIS	RTSEN	DTRDIS	DTREN
15	14	13	12	11	10	9	8
RETTO	RST_NACK	RSTIT	SENDA	STTTO	STPBRK	STTBRK	RSTSTA
7	6	5	4	3	2	1	0
TXDIS	TXEN	RXDIS	RXEN	RSTTX	RSTRX	–	–

- **RSTRX: Reset Receiver**

0 = No effect.

1 = The receiver logic is reset, disabling the receive function (RXDIS is set internally).

- **RSTTX: Reset Transmitter**

0 = No effect.

1 = The transmitter logic is reset, disabling the transmit function (TXDIS and STPBRK are set internally).

- **RXEN: Receiver Enable**

0 = No effect.

1 = The receiver is enabled if RXDIS is 0.

- **RXDIS: Receiver Disable**

0 = No effect.

1 = The receiver is disabled.

- **TXEN: Transmitter Enable**

0 = No effect.

1 = The transmitter is enabled if TXDIS is 0.

- **TXDIS: Transmitter Disable**

0 = No effect.

1 = The transmitter is disabled.

- **RSTSTA: Reset Status Bits**

0 = No effect.

1 = Resets the status bits PARE, FRAME, OVRE and RXBRK in the US_CSR.

- **STTBRK: Start Break**

0 = No effect.

1 = If break is not being transmitted, start transmission of a break after the characters present in US_THR and the Transmit Shift Register have been transmitted.

- **STPBRK: Stop Break**

0 = No effect.

1 = If a break is being transmitted, stop transmission of the break after a minimum of one character length and transmit a high level during 12 bit periods.

- **STTTO: Start Time-out**

0 = No effect.

1 = Start waiting for a character before clocking the time-out counter.

- **SENDA: Send Address**

0 = No effect.

1 = In Multi-drop Mode only, the next character written to the US_THR is sent with the address bit set.

- **RSTIT: Reset Iterations**

Note: This bit only takes effect in ISO7816 Mode.

0 = No effect.

1 = Resets the status bit Iteration.

- **RSTNACK: Reset Non-acknowledge**

0 = No effect.

1 = Resets the Nack status bit.

- **RETTO: Rearm Time-out**

0 = No effect.

1 = Restarts a time-out.

- **DTREN: Data Terminal Ready Enable**

0 = No effect.

1 = The DTR pin is forced to 0.

- **DTRDIS: Data Terminal Ready Disable**

0 = No effect.

1 = The DTR pin is forced to 1.

- **RTSEN: Request to Send Enable**

0 = No effect.

1 = The RTS pin is forced to 0.

- **RTSDIS: Request to Send Disable**

0 = No effect.

1 = The RTS pin is forced to 1.

USART2 Mode Register

Name: US_MR

Access Type: Read/Write

31	30	29	28	27	26	25	24
–	–	–	FILTER	–	MAX_ITERATION		

23	22	21	20	19	18	17	16
–	–	DSACK	INACK	OVER	CLKO	MODE9	MSBF

15	14	13	12	11	10	9	8
CHMODE		NBSTOP		PAR			SYNC

7	6	5	4	3	2	1	0
CHRL		USCLKS		–	USART2_MODE		

- **USART2_MODE**

USART2_MODE			Mode of the USART2
0	0	0	Normal
0	0	1	RS485
0	1	0	Hardware Handshaking
0	1	1	Modem
1	0	0	IS07816 Reader Protocol: T = 0
1	1	0	IS07816 Reader Protocol: T = 1

Note: The Baud Rate Clock used in mode IS07816 can be configured via the register FI_DI_RATIO.

.

The modes are described in detail in the paragraphs which follow.

- **USCLKS: Clock Selection (Baud Rate Generator Input Clock)**

USCLKS		Selected Clock
0	0	CLOCK
0	1	FDIV1
1	0	Slow Clock (ARM)
1	1	External (SCK)

- **CHRL: Character Length**

Start, stop and parity bits are added to the character length.

CHRL		Character Length
0	0	5 bits
0	1	6 bits
1	0	7 bits
1	1	8 bits

- **SYNC: Synchronous Mode Select**

0 = USART operates in Asynchronous Mode.

1 = USART operates in Synchronous Mode

- **PAR: Parity Type**

When the PAR field is set to Even parity, the parity bit is set (“1”) if the data parity is Odd in order to ensure an even parity on the Data and Parity field.

PAR			Parity Type
0	0	0	Even parity
0	0	1	Odd parity
0	1	0	Parity forced to 0 (Space)
0	1	1	Parity forced to 1 (Mark)
1	0	x	No parity
1	1	x	Multi-drop mode

- **NBSTOP: Number of Stop Bits**

The interpretation of the number of stop bits depends on SYNC.

NBSTOP		Asynchronous (SYNC = 0)	Synchronous (SYNC = 1)
0	0	1 stop bit	1 stop bit
0	1	1.5 stop bits	Reserved
1	0	2 stop bits	2 stop bits
1	1	Reserved	Reserved

Note: 1.5 or 2 stop bits are reserved for the TX function. The RX function uses only the 1 stop bit (there is no check on the 2 stop bit timeslot if NBSTOP = 10).

- **CHMODE: Channel Mode**

CHMODE		Mode Description
0	0	Normal Mode The USART2 Channel operates as an RX/TX USART2.
0	1	Automatic Echo Receiver Data Input is connected to the TXD pin.
1	0	Local Loopback Transmitter Output Signal is connected to Receiver Input Signal.
1	1	Remote Loopback RXD pin is internally connected to TXD pin.

- **MSBF: Bit Order**

0 = LSB First.

1 = MSB First.

- **MODE9: 9-bit Character Length**

0 = CHRL defines character length.

1 = 9-bit character length.

MODE9 has priority on character length.

- **CKLO: Clock Output Select**

0 = The USART does not drive the SCK pin.

1 = The USART drives the SCK pin if USCLKS[1] is 0.

- **OVER: Oversampling Mode**

0 = 16x Oversampling.

1 = 8x Oversampling.

- **INACK: Inhibit Non-acknowledge**

Note: This bit will be used only in ISO7816 mode, protocol T = 0 receiver.

0 = The NACK is generated.

1 = The NACK is not generated.

- **DSNACK: Disable Successive NACK**

0 = NACK is sent on the ISO line as soon as a parity error occurs in the received character (unless INACK is set).

1 = Successive parity errors are counted up to the value specified in the MAX_ITERATION field. These parity errors generate a NACK on the ISO line. As soon as this value is reached, no additional NACK is sent on the ISO line. The flag ITERATION is asserted.

- **MAX_ITERATION**

MAX_ITERATION	Number of Repetitions
0 -7	This will operate in mode ISO7816, Protocol T = 0 only

- **FILTER: Receive Line Filter**

0 = The USART2 does not filter receive line.

1 = The USART2 filters receive line using a three-sample filter (1/16-bit clock) (2 over 3 majority).

USART2 Interrupt Enable Register

Name: US_IER

Access Type: Write-only

31	30	29	28	27	26	25	24
COMM_RX	COMM_TX	–	–	–	–	–	–
23	22	21	20	19	18	17	16
–	–	–	–	CTSIC	DCDIC	DSRIC	RIIC
15	14	13	12	11	10	9	8
–	–	NACK	RXBUFF	TXBUFE	ITERATION	TXEMPTY	TIMEOUT
7	6	5	4	3	2	1	0
PARE	FRAME	OVRE	ENDTX	ENDRX	RXBRK	TXRDY	RXRDY

- **RXRDY: Enable RXRDY Interrupt**

0 = No effect.

1 = Enables RXRDY Interrupt.

- **TXRDY: Enable TXRDY Interrupt**

0 = No effect.

1 = Enables TXRDY Interrupt.

- **RXBRK: Enable Receiver Break Interrupt**

0 = No effect.

1 = Enables Receiver Break Interrupt.

- **ENDRX: Enable End of Receive Transfer Interrupt**

0 = No effect.

1 = Enables End of Receive Transfer Interrupt.

- **ENDTX: Enable End of Transmit Interrupt**

0 = No effect.

1 = Enables End of Transmit Interrupt.

- **OVRE: Enable Overrun Error Interrupt**

0 = No effect.

1 = Enables Overrun Error Interrupt.

- **FRAME: Enable Framing Error Interrupt**

0 = No effect.

1 = Enables Framing Error Interrupt.

- **PARE: Enable Parity Error Interrupt**

0 = No effect.

1 = Enables Parity Error Interrupt.

- **TIMEOUT: Enable Time-out Interrupt**

0 = No effect.

1 = Enables Reception Time-out Interrupt.

- **TXEMPTY: Enable TXEMPTY Interrupt**

0 = No effect.

1 = Enables TXEMPTY Interrupt.

- **ITERATION: Enable Iteration Interrupt**

Note: This will operate only in IS07816 mode, Protocol T = 0.

0 = No effect.

1 = Enables ITERATION interrupt.

- **TXBUFE: Enable Buffer Empty Interrupt**

0 = No effect.

1 = Enables Buffer Empty Interrupt.

- **RXBUFF: Enable Buffer Full Interrupt**

0 = No effect.

1 = Enable Buffer Full Interrupt.

- **NACK: Enable Non-acknowledge Interrupt**

0 = No effect.

1 = Enable Non-acknowledge Interrupt.

- **RIIC: Enable Ring Indicator Input Change**

0 = No effect.

1 = Enables Ring Indicator Input Change Interrupt.

- **DSRIC: Enable Data Set Ready Input Change**

0 = No effect.

1 = Enables Data Set Ready Input Change Interrupt.

- **DCDIC: Enable Data Carrier Detect Input Change Interrupt**

0 = No effect.

1 = Enables Data Carrier Detect Input Change Interrupt.

- **CTSIC: Enable Clear to Send Input Change Interrupt**

0 = No effect.

1 = Enables Clear to Send Input Change Interrupt.

- **COMM_TX: Enable COMM_TX (from ARM) Interrupt**

0 = No effect.

1 = Enables COMM_TX Interrupt.

- **COMM_RX: Enable COMM_RX (from ARM) Interrupt**

0 = No Effect.

1 = Enables COMM_RX Interrupt.

USART2 Interrupt Disable Register

Name: US_IDR

Access Type: Write-only

31	30	29	28	27	26	25	24
COMM_RX	COMM_TX	–	–	–	–	–	–
23	22	21	20	19	18	17	16
–	–	–	–	CTSIC	DCDIC	DSRIC	RIIC
15	14	13	12	11	10	9	8
–	–	NACK	RXBUFF	TXBUFE	ITERATION	TXEMPTY	TIMEOUT
7	6	5	4	3	2	1	0
PARE	FRAME	OVRE	ENDTX	ENDRX	RXBRK	TXRDY	RXRDY

- **RXRDY: Disable RXRDY Interrupt**

0 = No effect.

1 = Disables RXRDY Interrupt.

- **TXRDY: Disable TXRDY Interrupt**

0 = No effect.

1 = Disables TXRDY Interrupt.

- **RXBRK: Disable Receiver Break Interrupt**

0 = No effect.

1 = Disables Receiver Break Interrupt.

- **ENDRX: Disable End of Receive Transfer Interrupt**

0 = No effect.

1 = Disables End of Receive Transfer Interrupt.

- **ENDTX: Disable End of Transmit Interrupt**

0 = No effect.

1 = Disables End of Transmit Interrupt.

- **OVRE: Disable Overrun Error Interrupt**

0 = No effect.

1 = Disables Overrun Error Interrupt.

- **FRAME: Disable Framing Error Interrupt**

0 = No effect.

1 = Disables Framing Error Interrupt.

- **PARE: Disable Parity Error Interrupt**

0 = No effect.

1 = Disables Parity Error Interrupt.

- **TIMEOUT: Disable Time-out Interrupt**

0 = No effect.

1 = Disables Receiver Time-out Interrupt.

- **TXEMPTY: Disable TXEMPTY Interrupt**

0 = No effect.

1 = Disables TXEMPTY Interrupt.

- **ITERATION: Disable ITERATION Interrupt**

Note: This will operate only in IS07816 mode, protocol T = 0.

0 = No Effect.

1 = Disables ITERATION Interrupt.

- **TXBUFE: Disable Buffer Empty Interrupt**

0 = No effect.

1 = Disables Buffer Empty Interrupt.

- **RXBUFF: Disable Buffer Full Interrupt**

0 = No effect.

1 = Disables Buffer Full Interrupt.

- **NACK: Disable Non-acknowledge Interrupt**

0 = No effect.

1 = Disables Non-acknowledge Interrupt.

- **RIIC: Disable Ring Indicator Input Change**

0 = No effect.

1 = Disables Ring Indicator Input Change Interrupt.

- **DSRIC: Disable Data Set Ready Input Change**

0 = No effect.

1 = Disables Data Set Ready Input Change Interrupt.

- **DCDIC: Disable Data Carrier Detect Input Change Interrupt**

0 = No effect.

1 = Disables Data Carrier Detect Input Change Interrupt.

- **CTSIC: Disable Clear to Send Input Change Interrupt**

0 = No effect.

1 = Disables Clear to Send Input Change Interrupt.

- **COMM_TX: Disable COMM_TX (from ARM) Interrupt**

0 = No effect.

1 = Disables COMM_TX Interrupt.

- **COMM_RX: Disable COMM_RX (from ARM) Interrupt**

0 = No Effect.

1 = Disables COMM_RX Interrupt.

- **COMM_TX: Disable COMM_TX (from ARM) Interrupt**

0 = No effect.

1 = Disables COMM_TX Interrupt.

- **COMM_RX: Disable COMM_RX (from ARM) Interrupt**

0 = No effect.

1 = Disables COMM_RX Interrupt.

USART2 Interrupt Mask Register

Name: US_IMR

Access Type: Read-only

31	30	29	28	27	26	25	24
COMM_RX	COMM_TX	–	–	–	–	–	–
23	22	21	20	19	18	17	16
–	–	–	–	CTSIC	DCDIC	DSRIC	RIIC
15	14	13	12	11	10	9	8
–	–	NACK	RXBUFF	TXBUFE	ITERATION	TXEMPTY	TIMEOUT
7	6	5	4	3	2	1	0
PARE	FRAME	OVRE	ENDTX	ENDRX	RXBRK	TXRDY	RXRDY

- **RXRDY: Mask RXRDY Interrupt**

0 = RXRDY Interrupt is disabled.

1 = RXRDY Interrupt is enabled.

- **TXRDY: Mask TXRDY Interrupt**

0 = TXRDY Interrupt is disabled.

1 = TXRDY Interrupt is enabled.

- **RXBRK: Mask Receiver Break Interrupt**

0 = Receiver Break Interrupt is disabled.

1 = Receiver Break Interrupt is enabled.

- **ENDRX: Mask End of Receive Transfer Interrupt**

0 = End of Receive Transfer Interrupt is disabled.

1 = End of Receive Transfer Interrupt is enabled.

- **ENDTX: Mask End of Transmit Interrupt**

0 = End of Transmit Interrupt is disabled.

1 = End of Transmit Interrupt is enabled.

- **OVRE: Mask Overrun Error Interrupt**

0 = Overrun Error Interrupt is disabled.

1 = Overrun Error Interrupt is enabled.

- **FRAME: Mask Framing Error Interrupt**

0 = Framing Error Interrupt is disabled.

1 = Framing Error Interrupt is enabled.

- **PARE: Mask Parity Error Interrupt**

0 = Parity Error Interrupt is disabled.

1 = Parity Error Interrupt is enabled.

- **TIMEOUT: Mask Time-out Interrupt**

0 = Receive Time-out Interrupt is disabled.

1 = Receive Time-out Interrupt is enabled.

- **TXEMPTY: Mask TXEMPTY Interrupt**

0 = TXEMPTY Interrupt is disabled.

1 = TXEMPTY Interrupt is enabled.

- **ITERATION: Mask ITERATION Interrupt**

Note: This will operate only in IS07816 mode, protocol T = 0.

0 = ITERATION Interrupt is disabled.

1 = ITERATION Interrupt is enabled.

- **TXBUFE: Mask Buffer Empty Interrupt**

0 = TXBUFE Interrupt is disabled.

1 = TXBUFE Interrupt is enabled.

- **RXBUFF: Mask Buffer Full Interrupt**

0 = RXBUFF Interrupt is disabled.

1 = RXBUFF Interrupt is enabled.

- **NACK: Mask Non-acknowledge Interrupt**

0 = NACK Interrupt is disabled.

1 = NACK Interrupt is enabled.

- **RIIC: Mask Ring Indicator Input Change**

0 = RIIC Interrupt is disabled.

1 = RIIC Interrupt is enabled.

- **DSRIC: Mask Data Set Ready Input Change**

0 = DSRIC Interrupt is disabled.

1 = DSRIC Interrupt is enabled.

- **DCDIC: Mask Data Carrier Detect Input Change Interrupt**

0 = DCDIC Interrupt is disabled.

1 = DCDIC Interrupt is enabled.

- **CTSIC: Mask Clear to Send Input Change Interrupt**

0 = CTSIC Interrupt is disabled.

1 = CTSIC Interrupt is enabled.

- **COMM_TX: Mask COMM_TX (from ARM) Interrupt**

0 = COMM_TX Interrupt is disabled.

1 = COMM_TX Interrupt is enabled.

- **COMM_RX: Mask COMM_RX (from ARM) Interrupt**

0 = Comm_RX Interrupt is disabled.

1 = Comm_RX Interrupt is enabled.

USART2 Channel Status Register

Name: US_CSR

Access Type: Read-only

31	30	29	28	27	26	25	24
COMM_RX	COMM_TX	–	–	–	–	–	–
23	22	21	20	19	18	17	16
CTS	DCD	DSR	RI	CTSIC	DCDIC	DSRIC	RIIC
15	14	13	12	11	10	9	8
–	–	NACK	RXBUFF	TXBUFE	ITERATION	TXEMPTY	TIMEOUT
7	6	5	4	3	2	1	0
PARE	FRAME	OVRE	ENDTX	ENDRX	RXBRK	TXRDY	RXRDY

- **RXRDY: Receiver Ready**

0 = No complete character has been received since the last read of the US_RHR or the receiver is disabled. If characters were being received when the receiver was disabled, RXRDY changes to 1 when the receiver is enabled.

1 = At least one complete character has been received and the US_RHR has not yet been read.

- **TXRDY: Transmitter Ready**

0 = A character is in the US_THR waiting to be transferred to the Transmit Shift Register, or an STTBRK command has been requested, or the transmitter is disabled. As soon as the transmitter is enabled, TXRDY becomes 1.

1 = There is no character in the US_THR.

Equal to zero when the USART2 is disabled or at reset. The Transmitter Enable command (in US_CR) sets this bit to one if the transmitter was previously disabled.

- **RXBRK: Break Received/End of Break**

0 = No Break Received or End of Break detected since the last Reset Status Bits command in the Control Register.

1 = Break Received or End of Break detected since the last Reset Status Bits command in the Control Register.

- **ENDRX: End of Receiver Transfer**

0 = The End of Transfer signal from the Peripheral Data Controller channel dedicated to the receiver is inactive.

1 = The End of Transfer signal from the Peripheral Data Controller channel dedicated to the receiver is active.

- **ENDTX: End of Transmitter Transfer**

0 = The End of Transfer signal from the Peripheral Data Controller channel dedicated to the transmitter is inactive.

1 = The End of Transfer signal from the Peripheral Data Controller channel dedicated to the transmitter is active.

- **OVRE: Overrun Error**

0 = No byte has been transferred from the Receive Shift Register to the US_RHR when RxRDY was asserted since the last Reset Status Bits command.

1 = At least one byte has been transferred from the Receive Shift Register to the US_RHR when RxRDY was asserted since the last Reset Status Bits command.

- **FRAME: Framing Error**

0 = No stop bit has been detected low since the last Reset Status Bits command.

1 = At least one stop bit has been detected low since the last Reset Status Bits command.

- **PARE: Parity Error**

1 = At least one parity bit has been detected false (or a parity bit high in multi-drop mode) since the last Reset Status Bits command.

0 = No parity bit has been detected false (or a parity bit high in multi-drop mode) since last Reset Status Bits command.

- **TIMEOUT: Receiver Time-out**

0 = There has not been a time-out since the last Start Time-out command or the Time-out Register is 0.

1 = There has been a time-out since the last Start Time-out command.

- **TXEMPTY: Transmitter Empty**

0 = There are characters in either US_THR or the Transmit Shift Register, or the transmitter is disabled.

1 = There are no characters in either US_THR or the Transmit Shift Register. TXEMPTY is 1 after Parity, Stop Bit and Time-guard have been transmitted. TXEMPTY is 1 after stop bit has been sent, or after Time-guard has been sent if US_TTGR is not 0.

Equal to zero when the USART2 is disabled or at reset. Transmitter Enable command (in US_CR) sets this bit to one if the transmitter is disabled.

- **ITERATION: Max Number of Repetitions Reached**

Note: This bit will operate only in IS07816 mode, Protocol T = 0.

0 = Max number of repetitions has not been reached.

1 = Max number of repetitions has been reached.

- **TXBUFE: Transmission Buffer Empty**

0 = PDC2 Transmission Buffer is not empty.

1 = PDC2 Transmission Buffer is empty.

- **RXBUFF: Reception Buffer Full**

0 = PDC2 Reception Buffer is not full.

1 = PDC2 Reception Buffer is full.

- **NACK: Non-acknowledge**

0 = A Non-acknowledgement has not been detected.

1 = A Non-acknowledgement has been detected.

- **RIIC: Ring Indicator Input Change Flag**

0 = No input change has been detected on RI pin since last read of US_CSR.

1 = An input change has been detected on RI pin.

- **DSRIC: Data Set Ready Input Change Flag**

0 = No input change has been detected on DSR pin since last read of US_CSR.

1 = An input change has been detected on DSR pin.

- **DCDIC: Data Carrier Detect Input Change Flag**

0 = No input change has been detected on DCD pin since last read of US_CSR.

1 = An input change has been detected on DCD pin.

- **CTSIC: Clear to Send Input Change Flag**

0 = No input change has been detected on CTS pin since last read of US_CSR.

1 = An input change has been detected on CTS pin.

- **RI: Image of RI Input**

0 = RI is at 0.

1 = RI is at 1.

- **DSR: Image of DSR Input**

0 = DSR is at 0.

1 = DSR is at 1.

- **DCD: Image of DCD Input**

0 = DCD is at 0.

1 = DCD is at 1.

- **CTS: Image of CTS Input**

0 = CTS is at 0.

1 = CTS is at 1.

- **COMM_TX: (from ARM)**

0 = COMM_TX is at 0.

1 = COMM_TX is at 1.

- **COMM_RX: (from ARM)**

0 = COMM_RX is at 0.

1 = COMM_RX is at 1.

USART2 Receiver Holding Register

Name: US_RHR

Access Type: Read-only

31	30	29	28	27	26	25	24
–	–	–	–	–	–	–	–
23	22	21	20	19	18	17	16
–	–	–	–	–	–	–	–
15	14	13	12	11	10	9	8
–	–	–	–	–	–	–	RXCHR
7	6	5	4	3	2	1	0
RXCHR							

- **RXCHR: Received Character**

Last character received if RXRDY is set. When number of data bits is less than 9 bits, the bits are right-aligned. All non-significant bits read zero.

USART2 Transmitter Holding Register

Name: US_THR

Access Type: Write-only

31	30	29	28	27	26	25	24
–	–	–	–	–	–	–	–

23	22	21	20	19	18	17	16
–	–	–	–	–	–	–	–

15	14	13	12	11	10	9	8
–	–	–	–	–	–	–	TXCHR

7	6	5	4	3	2	1	0
TXCHR							

- **TXCHR: Character to be Transmitted**

Next character to be transmitted after the current character if TXRDY is not set. When number of data bits is less than 9 bits, the bits are right-aligned.

USART2 Baud Rate Generator Register

Name: US_BRGR

Access Type: Read/Write

31	30	29	28	27	26	25	24
–	–	–	–	–	–	–	–

23	22	21	20	19	18	17	16
–	–	–	–	–	–	–	–

15	14	13	12	11	10	9	8
CD							

7	6	5	4	3	2	1	0
CD							

- **CD: Clock Divisor**

This register has no effect if Synchronous Mode is selected with an external clock.

CD	Description
0	Disables Clock
1	Clock Divisor Bypass
2 to 65535	Baud Rate (Asynchronous Mode) = Selected Clock / (16 x CD) Baud Rate (Synchronous Mode) = Selected Clock / CD

Notes: 1. In Synchronous Mode, when either external clock (clk_ext or fdiv1) is selected, the value programmed must be even to ensure a 50:50 mark:space ratio.
In Synchronous Mode, when the internal clock (clock) is selected, the CD can be even and the duty clock is 50:50.
2. Clock divisor bypass (CD = 1) must not be used when the internal clock (clock) is selected (USCLKS = 0).

USART2 Receiver Time-out Register

Name: US_RTOR

Access Type: Read/Write

31	30	29	28	27	26	25	24
–	–	–	–	–	–	–	–
23	22	21	20	19	18	17	16
–	–	–	–	–	–	–	–
15	14	13	12	11	10	9	8
TO							
7	6	5	4	3	2	1	0
TO							

- **TO: Time-out Value**

TO	Description
0	Disables the RX Time-out function.
1 - 65535	The Time-out counter is loaded with TO (16-bits) when the Start Time-out command is given or when each new data character is received (after reception has started).

USART2 Transmitter Time-guard Register

Name: US_TTGR

Access Type: Read/Write

31	30	29	28	27	26	25	24
–	–	–	–	–	–	–	–
23	22	21	20	19	18	17	16
–	–	–	–	–	–	–	–
15	14	13	12	11	10	9	8
–	–	–	–	–	–	–	–
7	6	5	4	3	2	1	0
TG							

- **TG: Time-guard Value**

Time-guard duration = TG x Bit Period

TG	
0	Disables the TX Time-guard function.
1 - 255	TXD is inactive high after the transmission of each character for the time-guard duration.

USART2 FI_DI_RATIO Register[1]

Name: US_FIDI

Access Type: Read/Write

Reset Value: 0x174 [2]

31	30	29	28	27	26	25	24
–	–	–	–	–	–	–	–

23	22	21	20	19	18	17	16
–	–	–	–	–	–	–	–

15	14	13	12	11	10	9	8
–	–	–	–	–	FI_DI_RATIO		

7	6	5	4	3	2	1	0
FI_DI_RATIO							

Note: 1. This register has no effect if ISO7816 mode is not selected.

2. The hexadecimal value is 0x174. The corresponding decimal value is 372.

- **FI_DI_RATIO: FI Over DI Ratio Value**

Parameter used in mode ISO7816 to generate a specific bit rate.

FI_DI_RATIO	Description
0	Baud Rate = 0
1 - 2047	Baud Rate = Selected clock/FI_DI_RATIO/16

USART2 Nb Errors Register

Name: US_NER

Access Type: Read-only

31	30	29	28	27	26	25	24
–	–	–	–	–	–	–	–
23	22	21	20	19	18	17	16
–	–	–	–	–	–	–	–
15	14	13	12	11	10	9	8
–	–	–	–	–	–	–	–
7	6	5	4	3	2	1	0
NB_ERRORS							

- **NB_ERRORS: Number of Errors During ISO7816 Transfers**

This 8-bit register informs the user of the total amount of errors that occured during an ISO7816 transfer. It is a read-only register and is reset by reading the register.

Document Details

Title	USART2
Literature Number	1733C

Revision History

Version A	**Publication Date:** 17-July-2001
Version B	**Publication Date:** 11-Nov-2001
Version C	**Publication Date:** 26-Feb-2002

Revisions Since Previous Version

Page: 2	“iso_txd_in” Input pin added to Symbol.
Page: 3	“iso_txd_in” Input pin added to Pin Description Table.
Page: 4	Function of “rx_buf_full” signal changed in Pin Description Table.
Page: 4	Function of “tx_buf_empty” signal changed in Pin Description Table.
Page: 5	“iso_txd_in Input” pin added to Block Diagram.
Page: 17	Figure 12. ISO Mode Connection, new drawing added.
Page:21	Table 6. Description of Supplemental Protocol T = 0 Characteristics, text changed. Table 6. Non Acknowledge Inhibited, text changed.
Page: 22	Text changed in “Hardware Handshaking Mode” paragraph.
Page: 23	Text changed in “Modem Mode” section.
Page: 24	Table 7. USART2 Memory Map, Fi_Di Ratio Register, Reset State noted.
Page: 27	USART_2 Mode, table note amended.
Page: 44	Fi_Di Ratio Register, Reset State noted.

Atmel Headquarters

Corporate Headquarters
2325 Orchard Parkway
San Jose, CA 95131
TEL 1(408) 441-0311
FAX 1(408) 487-2600

Europe
Atmel SarL
Route des Arsenaux 41
Casa Postale 80
CH-1705 Fribourg
Switzerland
TEL (41) 26-426-5555
FAX (41) 26-426-5500

Asia
Atmel Asia, Ltd.
Room 1219
Chinachem Golden Plaza
77 Mody Road Tsimhatsui
East Kowloon
Hong Kong
TEL (852) 2721-9778
FAX (852) 2722-1369

Japan
Atmel Japan K.K.
9F, Tonetsu Shinkawa Bldg.
1-24-8 Shinkawa
Chuo-ku, Tokyo 104-0033
Japan
TEL (81) 3-3523-3551
FAX (81) 3-3523-7581

Atmel Operations

Memory
Atmel Corporate
2325 Orchard Parkway
San Jose, CA 95131
TEL 1(408) 436-4270
FAX 1(408) 436-4314

Microcontrollers
Atmel Corporate
2325 Orchard Parkway
San Jose, CA 95131
TEL 1(408) 436-4270
FAX 1(408) 436-4314

Atmel Nantes
La Chantrerie
BP 70602
44306 Nantes Cedex 3, France
TEL (33) 2-40-18-18-18
FAX (33) 2-40-18-19-60

ASIC/ASSP/Smart Cards
Atmel Rousset
Zone Industrielle
13106 Rousset Cedex, France
TEL (33) 4-42-53-60-00
FAX (33) 4-42-53-60-01

Atmel Colorado Springs
1150 East Cheyenne Mtn. Blvd.
Colorado Springs, CO 80906
TEL 1(719) 576-3300
FAX 1(719) 540-1759

Atmel Smart Card ICs
Scottish Enterprise Technology Park
Maxwell Building
East Kilbride G75 0QR, Scotland
TEL (44) 1355-803-000
FAX (44) 1355-242-743

RF/Automotive
Atmel Heilbronn
Theresienstrasse 2
Postfach 3535
74025 Heilbronn, Germany
TEL (49) 71-31-67-0
FAX (49) 71-31-67-2340

Atmel Colorado Springs
1150 East Cheyenne Mtn. Blvd.
Colorado Springs, CO 80906
TEL 1(719) 576-3300
FAX 1(719) 540-1759

Biometrics/Imaging/Hi-Rel MPU/
High Speed Converters/RF Datacom
Atmel Grenoble
Avenue de Rochepleine
BP 123
38521 Saint-Egreve Cedex, France
TEL (33) 4-76-58-30-00
FAX (33) 4-76-58-34-80

e-mail
literature@atmel.com

Web Site
http://www.atmel.com

1733C–CASIC–03/02 0M