Chapter 13

USART Peripheral Interface, UART Mode

The universal synchronous/asynchronous receive/transmit (USART) peripheral interface supports two serial modes with one hardware module. This chapter discusses the operation of the asynchronous UART mode. USART0 is implemented on the MSP430x12xx, MSP430x13xx, and MSP430x15x devices. In addition to USART0, the MSP430x14x and MSP430x16x devices implement a second identical USART module, USART1.

Topic **Page**

13.1 USART Introduction: UART Mode

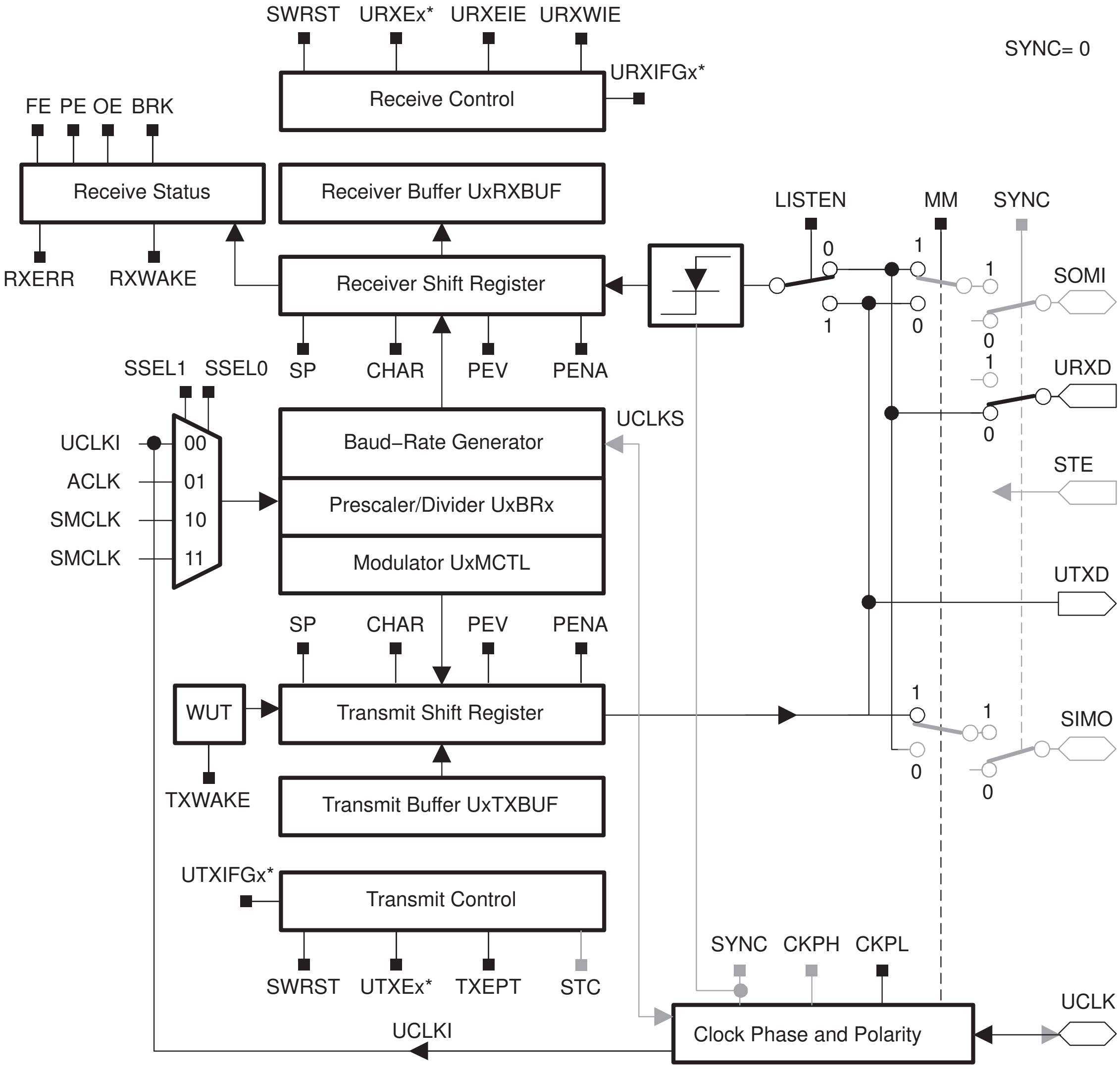
In asynchronous mode, the USART connects the MSP430 to an external system via two external pins, URXD and UTXD. UART mode is selected when the SYNC bit is cleared.

UART mode features include:

- 7- or 8-bit data with odd, even, or non-parity
- Independent transmit and receive shift registers
- Separate transmit and receive buffer registers
- LSB-first data transmit and receive
- Built-in idle-line and address-bit communication protocols for multiprocessor systems
- Receiver start-edge detection for auto-wake up from LPMx modes
- Programmable baud rate with modulation for fractional baud rate support
- Status flags for error detection and suppression and address detection
- Independent interrupt capability for receive and transmit

Figure 13–1 shows the USART when configured for UART mode.

Figure 13−1. USART Block Diagram: UART Mode



** Refer to the device-specific datasheet for SFR locations*

13.2 USART Operation: UART Mode

In UART mode, the USART transmits and receives characters at a bit rate asynchronous to another device. Timing for each character is based on the selected baud rate of the USART. The transmit and receive functions use the same baud rate frequency.

13.2.1 USART Initialization and Reset

The USART is reset by a PUC or by setting the SWRST bit. After a PUC, the SWRST bit is automatically set, keeping the USART in a reset condition. When set, the SWRST bit resets the URXIEx, UTXIEx, URXIFGx, RXWAKE, TXWAKE, RXERR, BRK, PE, OE, and FE bits and sets the UTXIFGx and TXEPT bits. The receive and transmit enable flags, URXEx and UTXEx, are not altered by SWRST. Clearing SWRST releases the USART for operation. See also chapter *USART Module, I2C mode* for USART0 when reconfiguring from I^{2}C mode to UART mode.

Note: Initializing or Re-Configuring the USART Module

The required USART initialization/re-configuration process is:

1) Set SWRST (`BIS.B   #SWRST,&UxCTL`)

2) Initialize all USART registers with SWRST = 1 (including UxCTL)

3) Enable USART module via the MEx SFRs (URXEx and/or UTXEx)

4) Clear SWRST via software (`BIC.B   #SWRST,&UxCTL`)

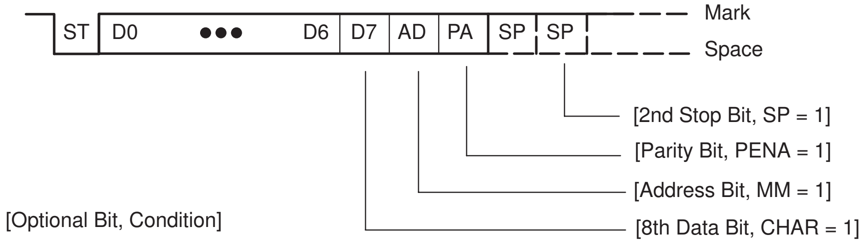
5) Enable interrupts (optional) via the IEx SFRs (URXIEx and/or UTXIEx)

Failure to follow this process may result in unpredictable USART behavior.

13.2.2 Character Format

The UART character format, shown in Figure 13–2, consists of a start bit, seven or eight data bits, an even/odd/no parity bit, an address bit (address-bit mode), and one or two stop bits. The bit period is defined by the selected clock source and setup of the baud rate registers.

Figure 13–2. Character Format



13.2.3 Asynchronous Communication Formats

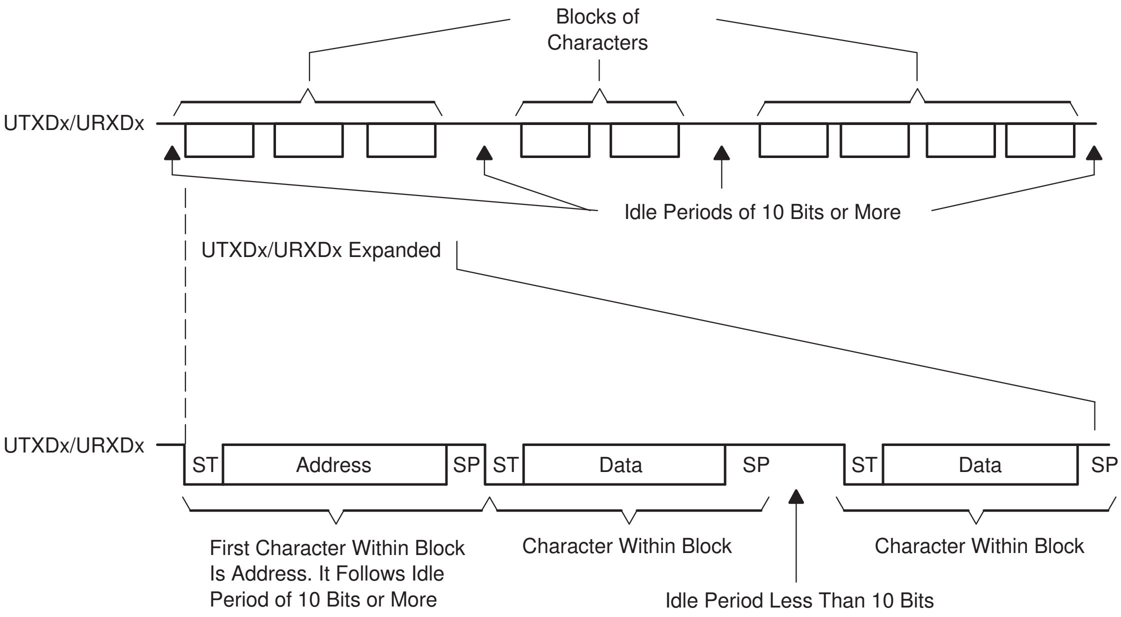
When two devices communicate asynchronously, the idle-line format is used for the protocol. When three or more devices communicate, the USART supports the idle-line and address-bit multiprocessor communication formats.

Idle-Line Multiprocessor Format

When MM = 0, the idle-line multiprocessor format is selected. Blocks of data are separated by an idle time on the transmit or receive lines as shown in Figure 13–3. An idle receive line is detected when 10 or more continuous ones (marks) are received after the first stop bit of a character. When two stop bits are used for the idle line the second stop bit is counted as the first mark bit of the idle period.

The first character received after an idle period is an address character. The RXWAKE bit is used as an address tag for each block of characters. In the idle-line multiprocessor format, this bit is set when a received character is an address and is transferred to UxRXBUF.

Figure 13–3. Idle-Line Format


Blocks of Characters
UTXDx/URXDx
Idle Periods of 10 Bits or More
UTXDx/URXDx Expanded
UTXDx/URXDx
ST
Address
SP
ST
Data
SP
ST
Data
SP
First Character Within Block Is Address. It Follows Idle Period of 10 Bits or More
Character Within Block
Character Within Block
Idle Period Less Than 10 Bits

The URXWIE bit is used to control data reception in the idle-line multiprocessor format. When the URXWIE bit is set, all non-address characters are assembled but not transferred into the UxRXBUF, and interrupts are not generated. When an address character is received, the receiver is temporarily activated to transfer the character to UxRXBUF and sets the URXIFGx interrupt flag. Any applicable error flag is also set. The user can then validate the received address.

If an address is received, user software can validate the address and must reset URXWIE to continue receiving data. If URXWIE remains set, only address characters will be received. The URXWIE bit is not modified by the USART hardware automatically.

For address transmission in idle-line multiprocessor format, a precise idle period can be generated by the USART to generate address character identifiers on UTXDx. The wake-up temporary (WUT) flag is an internal flag double-buffered with the user-accessible TXWAKE bit. When the transmitter is loaded from UxTXBUF, WUT is also loaded from TXWAKE resetting the TXWAKE bit.

The following procedure sends out an idle frame to indicate an address character will follow:

1) Set TXWAKE, then write any character to UxTXBUF. UxTXBUF must be ready for new data (UTXIFGx = 1).

   The TXWAKE value is shifted to WUT and the contents of UxTXBUF are shifted to the transmit shift register when the shift register is ready for new data. This sets WUT, which suppresses the start, data, and parity bits of a normal transmission, then transmits an idle period of exactly 11 bits. When two stop bits are used for the idle line, the second stop bit is counted as the first mark bit of the idle period. TXWAKE is reset automatically.

2) Write desired address character to UxTXBUF. UxTXBUF must be ready for new data (UTXIFGx = 1).

   The new character representing the specified address is shifted out following the address-identifying idle period on UTXDx. Writing the first “don’t care” character to UxTXBUF is necessary in order to shift the TXWAKE bit to WUT and generate an idle-line condition. This data is discarded and does not appear on UTXDx.

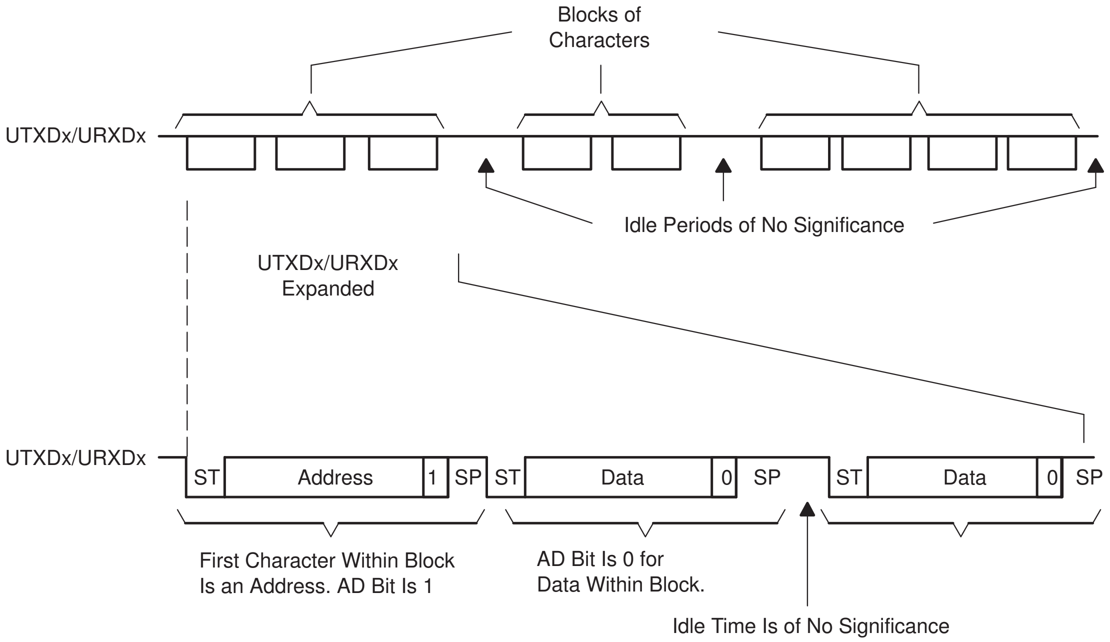
Address-Bit Multiprocessor Format

When MM = 1, the address-bit multiprocessor format is selected. Each processed character contains an extra bit used as an address indicator shown in Figure 13−4. The first character in a block of characters carries a set address bit which indicates that the character is an address. The USART RXWAKE bit is set when a received character is a valid address character and is transferred to UxRXBUF.

The URXWIE bit is used to control data reception in the address-bit multiprocessor format. If URXWIE is set, data characters (address bit = 0) are assembled by the receiver but are not transferred to UxRXBUF and no interrupts are generated. When a character containing a set address bit is received, the receiver is temporarily activated to transfer the character to UxRXBUF and set URXIFGx. All applicable error status flags are also set.

If an address is received, user software must reset URXWIE to continue receiving data. If URXWIE remains set, only address characters (address bit = 1) will be received. The URXWIE bit is not modified by the USART hardware automatically.

Figure 13−4. Address-Bit Multiprocessor Format


Blocks of Characters
UTXDx/URXDx
Idle Periods of No Significance
UTXDx/URXDx Expanded
UTXDx/URXDx
ST
Address
1
SP
ST
Data
0
SP
ST
Data
0
SP
First Character Within Block Is an Address. AD Bit Is 1
AD Bit Is 0 for Data Within Block.
Idle Time Is of No Significance

For address transmission in address-bit multiprocessor mode, the address bit of a character can be controlled by writing to the TXWAKE bit. The value of the TXWAKE bit is loaded into the address bit of the character transferred from UxTXBUF to the transmit shift register, automatically clearing the TXWAKE bit. TXWAKE must not be cleared by software. It is cleared by USART hardware after it is transferred to WUT or by setting SWRST.

Automatic Error Detection

Glitch suppression prevents the USART from being accidentally started. Any low-level on URXDx shorter than the deglitch time t_τ (approximately 300 ns) will be ignored. See the device-specific datasheet for parameters.

When a low period on URXDx exceeds t_τ a majority vote is taken for the start bit. If the majority vote fails to detect a valid start bit the USART halts character reception and waits for the next low period on URXDx. The majority vote is also used for each bit in a character to prevent bit errors.

The USART module automatically detects framing errors, parity errors, overrun errors, and break conditions when receiving characters. The bits FE, PE, OE, and BRK are set when their respective condition is detected. When any of these error flags are set, RXERR is also set. The error conditions are described in Table 13−1.

Table 13−1.Receive Error Conditions

Error Condition	Description
Framing error	A framing error occurs when a low stop bit is detected. When two stop bits are used, only the first stop bit is checked for framing error. When a framing error is detected, the FE bit is set.
Parity error	A parity error is a mismatch between the number of 1s in a character and the value of the parity bit. When an address bit is included in the character, it is included in the parity calculation. When a parity error is detected, the PE bit is set.
Receive overrun error	An overrun error occurs when a character is loaded into UxRXBUF before the prior character has been read. When an overrun occurs, the OE bit is set.
Break condition	A break condition is a period of 10 or more low bits received on URXDx after a missing stop bit. When a break condition is detected, the BRK bit is set. A break condition can also set the interrupt flag URXIFGx.

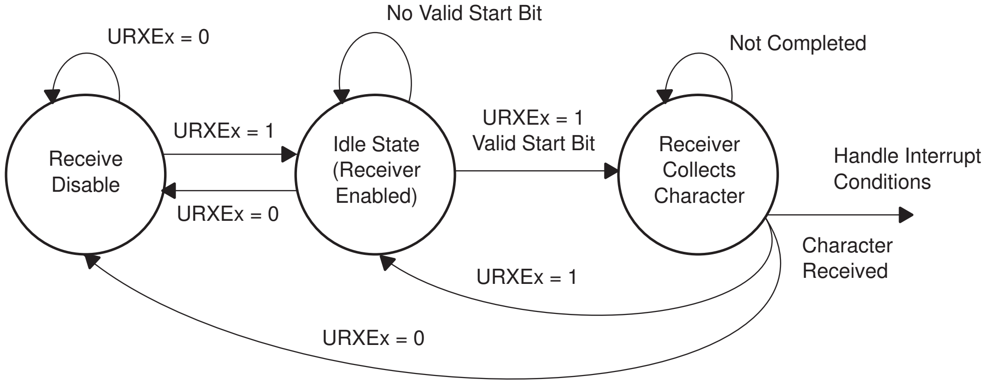
When URXEIE = 0 and a framing error, parity error, or break condition is detected, no character is received into UxRXBUF. When URXEIE = 1, characters are received into UxRXBUF and any applicable error bit is set.

When any of the FE, PE, OE, BRK, or RXERR bits is set, the bit remains set until user software resets it or UxRXBUF is read.

13.2.4 USART Receive Enable

The receive enable bit, URXEx, enables or disables data reception on URXDx as shown in Figure 13−5. Disabling the USART receiver stops the receive operation following completion of any character currently being received or immediately if no receive operation is active. The receive-data buffer, UxRXBUF, contains the character moved from the RX shift register after the character is received.

Figure 13−5. State Diagram of Receiver Enable



Note: Re-Enabling the Receiver (Setting URXEx): UART Mode

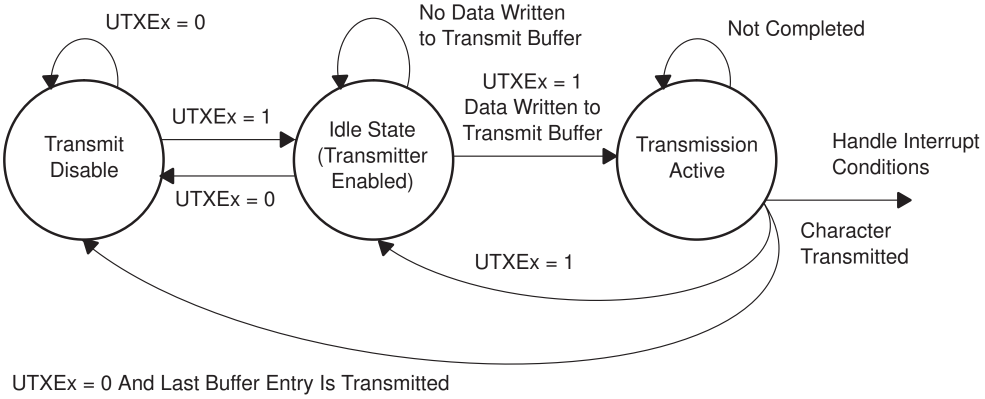
When the receiver is disabled (URXEx = 0), re-enabling the receiver (URXEx = 1) is asynchronous to any data stream that may be present on URXDx at the time. Synchronization can be performed by testing for an idle line condition before receiving a valid character (see URXWIE).

13.2.5 USART Transmit Enable

When UTXEx is set, the UART transmitter is enabled. Transmission is initiated by writing data to UxTXBUF. The data is then moved to the transmit shift register on the next BITCLK after the TX shift register is empty, and transmission begins. This process is shown in Figure 13−6.

When the UTXEx bit is reset the transmitter is stopped. Any data moved to UxTXBUF and any active transmission of data currently in the transmit shift register prior to clearing UTXEx will continue until all data transmission is completed.

Figure 13−6. State Diagram of Transmitter Enable


UTXEx = 0
Transmit Disable
UTXEx = 1
UTXEx = 0
No Data Written to Transmit Buffer
Idle State (Transmitter Enabled)
UTXEx = 1 Data Written to Transmit Buffer
Not Completed
Transmission Active
Handle Interrupt Conditions
Character Transmitted
UTXEx = 1
UTXEx = 0 And Last Buffer Entry Is Transmitted

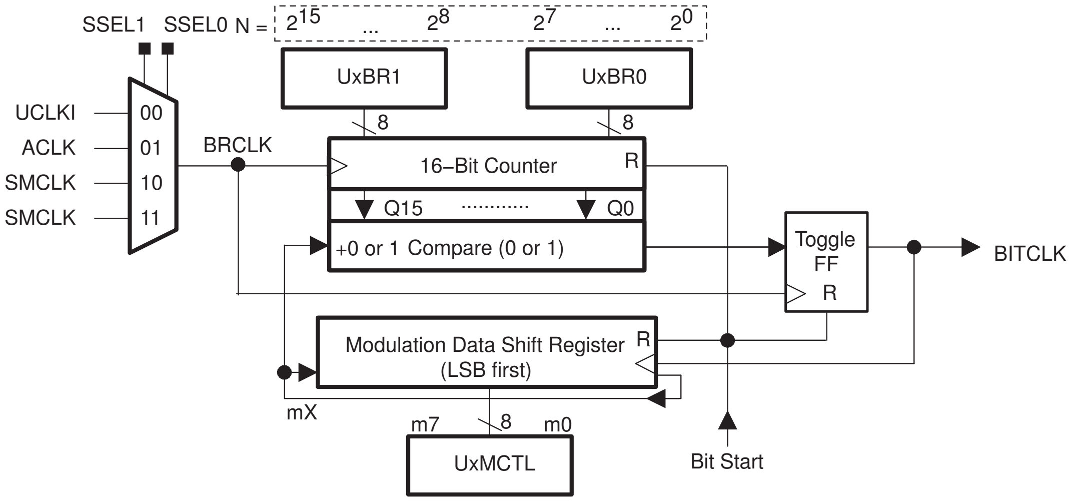
When the transmitter is enabled (UTXEx = 1), data should not be written to UxTXBUF unless it is ready for new data indicated by UTXIFGx = 1. Violation can result in an erroneous transmission if data in UxTXBUF is modified as it is being moved into the TX shift register.

It is recommended that the transmitter be disabled (UTXEx = 0) only after any active transmission is complete. This is indicated by a set transmitter empty bit (TXEPT = 1). Any data written to UxTXBUF while the transmitter is disabled will be held in the buffer but will not be moved to the transmit shift register or transmitted. Once UTXEx is set, the data in the transmit buffer is immediately loaded into the transmit shift register and character transmission resumes.

13.2.6 UART Baud Rate Generation

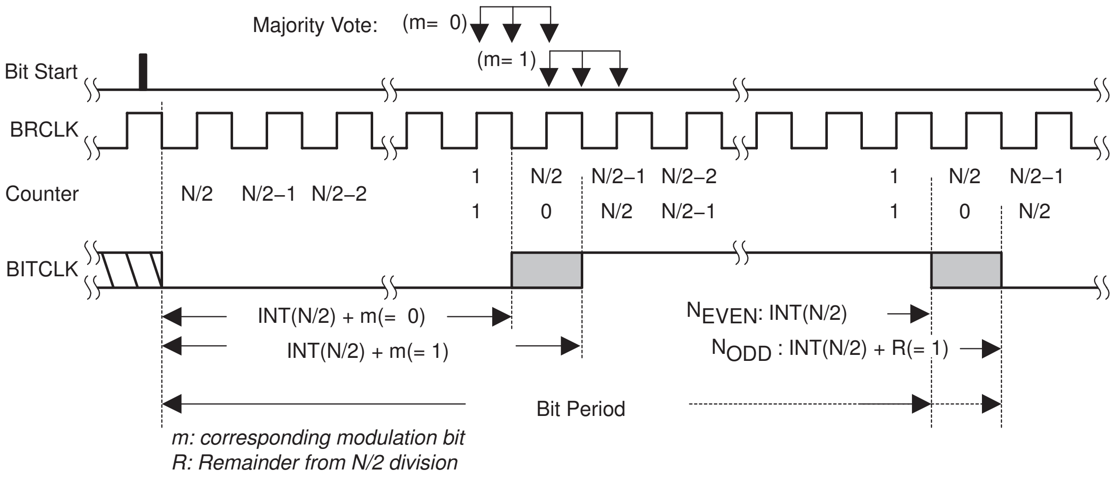
The USART baud rate generator is capable of producing standard baud rates from non-standard source frequencies. The baud rate generator uses one prescaler/divider and a modulator as shown in Figure 13−7. This combination supports fractional divisors for baud rate generation. The maximum USART baud rate is one-third the UART source clock frequency BRCLK.

Figure 13−7. MSP430 Baud Rate Generator



Timing for each bit is shown in Figure 13−8. For each bit received, a majority vote is taken to determine the bit value. These samples occur at the N/2−1, N/2, and N/2+1 BRCLK periods, where N is the number of BRCLKs per BITCLK.

Figure 13−8. BITCLK Baud Rate Timing



Baud Rate Bit Timing

The first stage of the baud rate generator is the 16-bit counter and comparator. At the beginning of each bit transmitted or received, the counter is loaded with INT(N/2) where N is the value stored in the combination of UxBR0 and UxBR1. The counter reloads INT(N/2) for each bit period half-cycle, giving a total bit period of N BRCLKs. For a given BRCLK clock source, the baud rate used determines the required division factor N:

$$\mathrm{N} = \frac{\mathrm{BRCLK}}{\mathrm{baud\ rate}}$$

The division factor N is often a non-integer value of which the integer portion can be realized by the prescaler/divider. The second stage of the baud rate generator, the modulator, is used to meet the fractional part as closely as possible. The factor *N* is then defined as:

$$N = UxBR + \frac{1}{n}\sum_{i=0}^{n-1} m_i$$

Where:

N: Target division factor
UxBR: 16-bit representation of registers UxBR0 and UxBR1
i: Bit position in the character
n: Total number of bits in the character
m_i: Data of each corresponding modulation bit (1 or 0)

$$Baud\ rate = \frac{BRCLK}{N} = \frac{BRCLK}{UxBR + \frac{1}{n}\sum_{i=0}^{n-1} m_i}$$

The BITCLK can be adjusted from bit to bit with the modulator to meet timing requirements when a non-integer divisor is needed. Timing of each bit is expanded by one BRCLK clock cycle if the modulator bit m_i is set. Each time a bit is received or transmitted, the next bit in the modulation control register determines the timing for that bit. A set modulation bit increases the division factor by one while a cleared modulation bit maintains the division factor given by UxBR.

The timing for the start bit is determined by UxBR plus m0, the next bit is determined by UxBR plus m1, and so on. The modulation sequence begins with the LSB. When the character is greater than 8 bits, the modulation sequence restarts with m0 and continues until all bits are processed.

Determining the Modulation Value

Determining the modulation value is an interactive process. Using the timing error formula provided, beginning with the start bit , the individual bit errors are calculated with the corresponding modulator bit set and cleared. The modulation bit setting with the lower error is selected and the next bit error is calculated. This process is continued until all bit errors are minimized. When a character contains more than 8 bits, the modulation bits repeat. For example, the 9th bit of a character uses modulation bit 0.

Transmit Bit Timing

The timing for each character is the sum of the individual bit timings. By modulating each bit, the cumulative bit error is reduced. The individual bit error can be calculated by:

$$\mathit{Error}\,[\%] = \left\{ \frac{\mathit{baud\ rate}}{\mathit{BRCLK}} \times \left[(j + 1) \times \mathit{UxBR} + \sum_{i=0}^{j} m_i \right] - (j + 1) \right\} \times 100\%$$

With:

baud rate: Desired baud rate
BRCLK: Input frequency – UCLKI, ACLK, or SMCLK
j: Bit position - 0 for the start bit, 1 for data bit D0, and so on
UxBR: Division factor in registers UxBR1 and UxBR0

For example, the transmit errors for the following conditions are calculated:

Baud rate = 2400
BRCLK = 32,768 Hz (ACLK)
UxBR = 13, since the ideal division factor is 13.65
UxMCTL = 6Bh: m7 = 0, m6 = 1, m5 = 1, m4 = 0, m3 = 1, m2 = 0, m1 = 1, and m0 = 1. The LSB of UxMCTL is used first.

$$\text{Start bit Error [\%]} = \left(\frac{\text{baud rate}}{\text{BRCLK}} \times ((0 + 1) \times \text{UxBR} + 1)–1 \right) \times 100\% = 2.54\%$$

$$\text{Data bit D0 Error [\%]} = \left(\frac{\text{baud rate}}{\text{BRCLK}} \times ((1 + 1) \times \text{UxBR} + 2)–2 \right) \times 100\% = 5.08\%$$

$$\text{Data bit D1 Error [\%]} = \left(\frac{\text{baud rate}}{\text{BRCLK}} \times ((2 + 1) \times \text{UxBR} + 2)–3 \right) \times 100\% = 0.29\%$$

$$\text{Data bit D2 Error [\%]} = \left(\frac{\text{baud rate}}{\text{BRCLK}} \times ((3 + 1) \times \text{UxBR} + 3)–4 \right) \times 100\% = 2.83\%$$

$$\text{Data bit D3 Error [\%]} = \left(\frac{\text{baud rate}}{\text{BRCLK}} \times ((4 + 1) \times \text{UxBR} + 3)–5 \right) \times 100\% = -1.95\%$$

$$\text{Data bit D4 Error [\%]} = \left(\frac{\text{baud rate}}{\text{BRCLK}} \times ((5 + 1) \times \text{UxBR} + 4)–6 \right) \times 100\% = 0.59\%$$

$$\text{Data bit D5 Error [\%]} = \left(\frac{\text{baud rate}}{\text{BRCLK}} \times ((6 + 1) \times \text{UxBR} + 5)–7 \right) \times 100\% = 3.13\%$$

$$\text{Data bit D6 Error [\%]} = \left(\frac{\text{baud rate}}{\text{BRCLK}} \times ((7 + 1) \times \text{UxBR} + 5)–8 \right) \times 100\% = -1.66\%$$

$$\text{Data bit D7 Error [\%]} = \left(\frac{\text{baud rate}}{\text{BRCLK}} \times ((8 + 1) \times \text{UxBR} + 6)–9 \right) \times 100\% = 0.88\%$$

$$\text{Parity bit Error [\%]} = \left(\frac{\text{baud rate}}{\text{BRCLK}} \times ((9 + 1) \times \text{UxBR} + 7)–10 \right) \times 100\% = 3.42\%$$

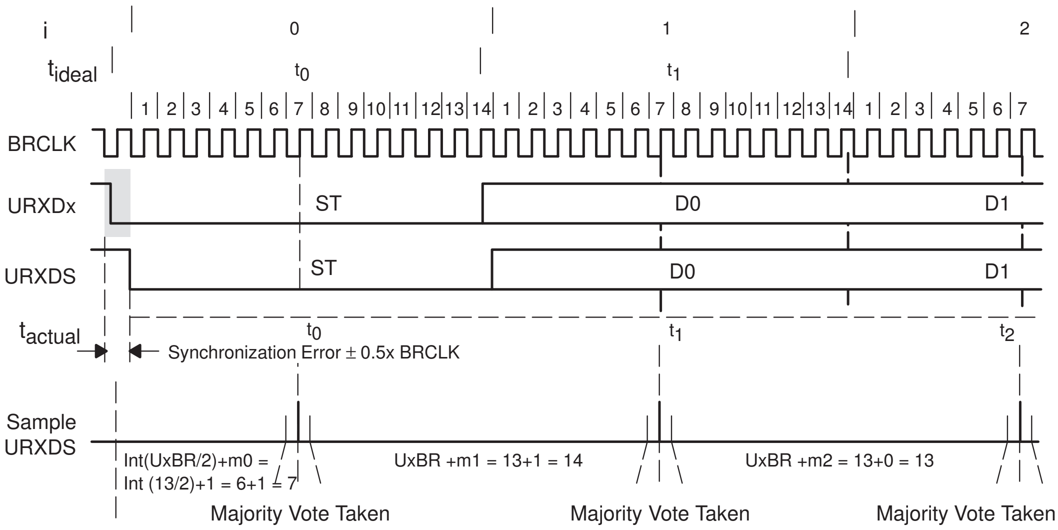
$$\text{Stop bit 1 Error [\%]} = \left(\frac{\text{baud rate}}{\text{BRCLK}} \times ((10 + 1) \times \text{UxBR} + 7)–11 \right) \times 100\% = -1.37\%$$

The results show the maximum per-bit error to be 5.08% of a BITCLK period.

Receive Bit Timing

Receive timing consists of two error sources. The first is the bit-to-bit timing error. The second is the error between a start edge occurring and the start edge being accepted by the USART. Figure 13–9 shows the asynchronous timing errors between data on the URXDx pin and the internal baud-rate clock.

Figure 13–9. Receive Error



The ideal start bit timing $t_{ideal(0)}$ is half the baud-rate timing $t_{baud\ rate}$ because the bit is tested in the middle of its period. The ideal baud rate timing $t_{ideal(i)}$ for the remaining character bits is the baud rate timing $t_{baud\ rate}$. The individual bit errors can be calculated by:

$$Error\,[\%] = \left(\frac{baud\ rate}{BRCLK} \times \left\{ 2 \times \left[m0 + int\left(\frac{UxBR}{2}\right)\right] + \left(i \times UxBR + \sum_{i=1}^{j} m_i \right) \right\} - 1 - j \right) \times 100\%$$

Where:

baud rate is the required baud rate
BRCLK is the input frequency—selected for UCLK, ACLK, or SMCLK
j = 0 for the start bit, 1 for data bit D0, and so on
UxBR is the division factor in registers UxBR1 and UxBR0

For example, the receive errors for the following conditions are calculated:

Baud rate = 2400
BRCLK = 32,768 Hz (ACLK)
UxBR = 13, since the ideal division factor is 13.65
UxMCTL = 6B:m7 = 0, m6 = 1, m5 = 1, m4 = 0, m3 = 1, m2 = 0, m1 = 1 and m0 = 1 The LSB of UxMCTL is used first.

$$\text{Start bit Error [\%]} = \left(\frac{\text{baud rate}}{\text{BRCLK}} \times [2x(1+6) + (0 \times \text{UxBR} + 0)] - 1 - 0\right) \times 100\% = 2.54\%$$

$$\text{Data bit D0 Error [\%]} = \left(\frac{\text{baud rate}}{\text{BRCLK}} \times [2x(1+6) + (1 \times \text{UxBR} + 1)] - 1 - 1\right) \times 100\% = 5.08\%$$

$$\text{Data bit D1 Error [\%]} = \left(\frac{\text{baud rate}}{\text{BRCLK}} \times [2x(1+6) + (2 \times \text{UxBR} + 1)] - 1 - 2\right) \times 100\% = 0.29\%$$

$$\text{Data bit D2 Error [\%]} = \left(\frac{\text{baud rate}}{\text{BRCLK}} \times [2x(1+6) + (3 \times \text{UxBR} + 2)] - 1 - 3\right) \times 100\% = 2.83\%$$

$$\text{Data bit D3 Error [\%]} = \left(\frac{\text{baud rate}}{\text{BRCLK}} \times [2x(1+6) + (4 \times \text{UxBR} + 2)] - 1 - 4\right) \times 100\% = -1.95\%$$

$$\text{Data bit D4 Error [\%]} = \left(\frac{\text{baud rate}}{\text{BRCLK}} \times [2x(1+6) + (5 \times \text{UxBR} + 3)] - 1 - 5\right) \times 100\% = 0.59\%$$

$$\text{Data bit D5 Error [\%]} = \left(\frac{\text{baud rate}}{\text{BRCLK}} \times [2x(1+6) + (6 \times \text{UxBR} + 4)] - 1 - 6\right) \times 100\% = 3.13\%$$

$$\text{Data bit D6 Error [\%]} = \left(\frac{\text{baud rate}}{\text{BRCLK}} \times [2x(1+6) + (7 \times \text{UxBR} + 4)] - 1 - 7\right) \times 100\% = -1.66\%$$

$$\text{Data bit D7 Error [\%]} = \left(\frac{\text{baud rate}}{\text{BRCLK}} \times [2x(1+6) + (8 \times \text{UxBR} + 5)] - 1 - 8\right) \times 100\% = 0.88\%$$

$$\text{Parity bit Error [\%]} = \left(\frac{\text{baud rate}}{\text{BRCLK}} \times [2x(1+6) + (9 \times \text{UxBR} + 6)] - 1 - 9\right) \times 100\% = 3.42\%$$

$$\text{Stop bit 1 Error [\%]} = \left(\frac{\text{baud rate}}{\text{BRCLK}} \times [2x(1+6) + (10 \times \text{UxBR} + 6)] - 1 - 10\right) \times 100\% = -1.37\%$$

The results show the maximum per-bit error to be 5.08% of a BITCLK period.

Typical Baud Rates and Errors

Standard baud rate frequency data for UxBRx and UxMCTL are listed in Table 13−2 for a 32,768-Hz watch crystal (ACLK) and a typical 1,048,576-Hz SMCLK.

The receive error is the accumulated time versus the ideal scanning time in the middle of each bit. The transmit error is the accumulated timing error versus the ideal time of the bit period.

Table 13−2. Commonly Used Baud Rates, Baud Rate Data, and Errors

	Divide by		A: BRCLK = 32,768 Hz						B: BRCLK = 1,048,576 Hz				
Baud Rate	A:	B:	UxBR1	UxBR0	UxMCTL	Max. TX Error %	Max. RX Error %	Synchr. RX Error %	UxBR1	UxBR0	UxMCTL	Max. TX Error %	Max. RX Error %
1200	27.31	873.81	0	1B	03	−4/3	−4/3	±2	03	69	FF	0/0.3	±2
2400	13.65	436.91	0	0D	6B	−6/3	−6/3	±4	01	B4	FF	0/0.3	±2
4800	6.83	218.45	0	06	6F	−9/11	−9/11	±7	0	DA	55	0/0.4	±2
9600	3.41	109.23	0	03	4A	−21/12	−21/12	±15	0	6D	03	−0.4/1	±2
19,200		54.61							0	36	6B	−0.2/2	±2
38,400		27.31							0	1B	03	−4/3	±2
76,800		13.65							0	0D	6B	−6/3	±4
115,200		9.1							0	09	08	−5/7	±7

13.2.7 USART Interrupts

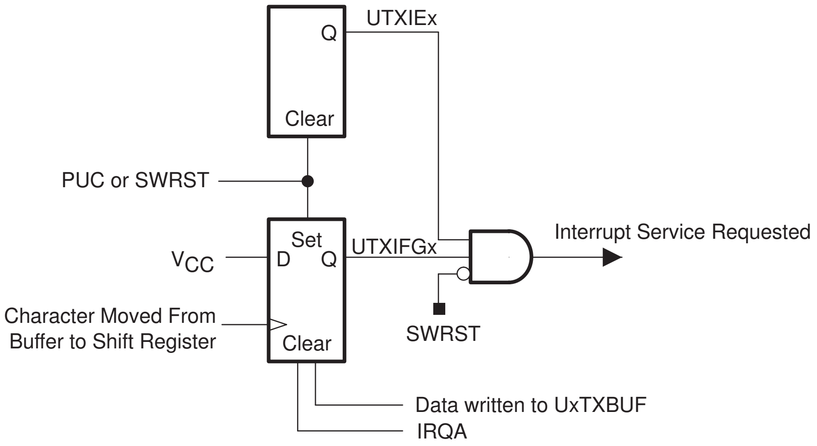
The USART has one interrupt vector for transmission and one interrupt vector for reception.

USART Transmit Interrupt Operation

The UTXIFGx interrupt flag is set by the transmitter to indicate that UxTXBUF is ready to accept another character. An interrupt request is generated if UTXIEx and GIE are also set. UTXIFGx is automatically reset if the interrupt request is serviced or if a character is written to UxTXBUF.

UTXIFGx is set after a PUC or when SWRST = 1. UTXIEx is reset after a PUC or when SWRST = 1. The operation is shown is Figure 13−10.

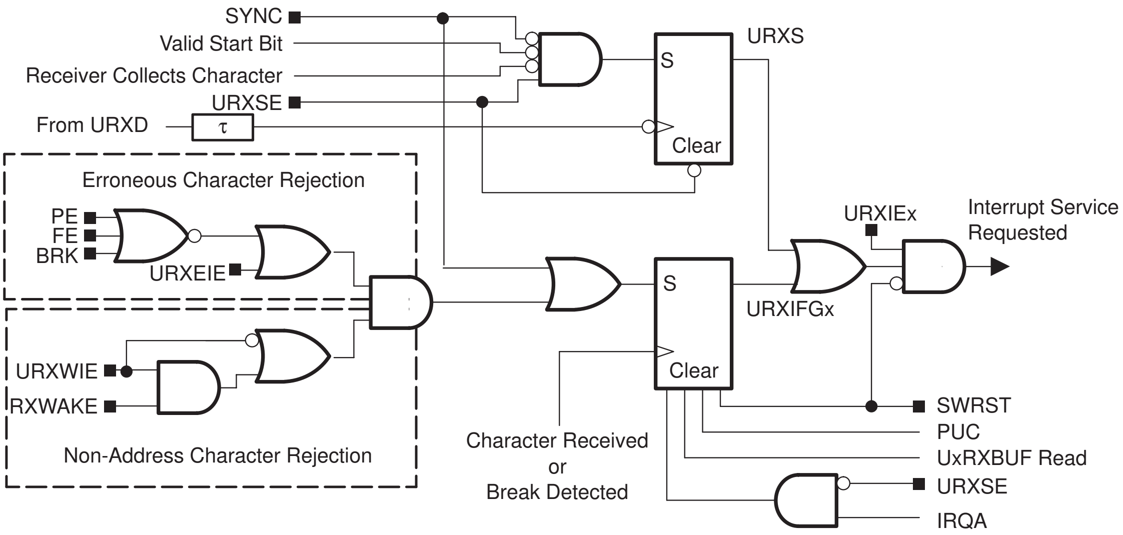
Figure 13−10. Transmit Interrupt Operation



USART Receive Interrupt Operation

The URXIFGx interrupt flag is set each time a character is received and loaded into UxRXBUF. An interrupt request is generated if URXIEx and GIE are also set. URXIFGx and URXIEx are reset by a system reset PUC signal or when SWRST = 1. URXIFGx is automatically reset if the pending interrupt is served (when URXSE = 0) or when UxRXBUF is read. The operation is shown in Figure 13−11.

Figure 13−11.Receive Interrupt Operation


SYNC
Valid Start Bit
Receiver Collects Character
URXSE
From URXD
τ
S
Clear
URXS
Erroneous Character Rejection
PE
FE
BRK
URXEIE
URXIEx
Interrupt Service Requested
S
Clear
URXIFGx
URXWIE
RXWAKE
Non-Address Character Rejection
Character Received or Break Detected
SWRST
PUC
UxRXBUF Read
URXSE
IRQA

URXEIE is used to enable or disable erroneous characters from setting URXIFGx. When using multiprocessor addressing modes, URXWIE is used to auto-detect valid address characters and reject unwanted data characters.

Two types of characters do not set URXIFGx:

- ❑ Erroneous characters when URXEIE = 0
- ❑ Non-address characters when URXWIE = 1

When URXEIE = 1 a break condition will set the BRK bit and the URXIFGx flag.

Receive-Start Edge Detect Operation

The URXSE bit enables the receive start-edge detection feature. The recommended usage of the receive-start edge feature is when BRCLK is sourced by the DCO and when the DCO is off because of low-power mode operation. The ultra-fast turn-on of the DCO allows character reception after the start edge detection.

When URXSE, URXIEx and GIE are set and a start edge occurs on URXDx, the internal signal URXS will be set. When URXS is set, a receive interrupt request is generated but URXIFGx is not set. User software in the receive interrupt service routine can test URXIFGx to determine the source of the interrupt. When URXIFGx = 0 a start edge was detected and when URXIFGx = 1 a valid character (or break) was received.

When the ISR determines the interrupt request was from a start edge, user software toggles URXSE, and must enable the BRCLK source by returning from the ISR to active mode or to a low-power mode where the source is active. If the ISR returns to a low-power mode where the BRCLK source is inactive, the character will not be received. Toggling URXSE clears the URXS signal and re-enables the start edge detect feature for future characters. See chapter *System Resets, Interrupts, and Operating Modes* for information on entering and exiting low-power modes.

The now active BRCLK allows the USART to receive the balance of the character. After the full character is received and moved to UxRXBUF, URXIFGx is set and an interrupt service is again requested. Upon ISR entry, URXIFGx = 1 indicating a character was received. The URXIFGx flag is cleared when user software reads UxRXBUF.

```
; Interrupt handler for start condition and
; Character receive. BRCLK = DCO.

U0RX_Int  BIT.B  #URXIFG0,&IFG2   ; Test URXIFGx to determine
          JNE    ST_COND          ; If start or character
          MOV.B  &UxRXBUF,dst     ; Read buffer
          ...                     ;
          RETI                    ;

ST_COND   BIC.B  #URXSE,&U0TCTL   ; Clear URXS signal
          BIS.B  #URXSE,&U0TCTL   ; Re-enable edge detect
          BIC    #SCG0+SCG1,0(SP) ; Enable BRCLK = DCO
          RETI                    ;
```

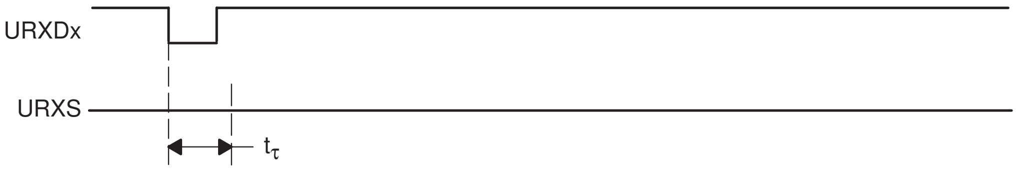
Note: Break Detect With Halted UART Clock

When using the receive start-edge detect feature a break condition cannot be detected when the BRCLK source is off.

Receive-Start Edge Detect Conditions

When URXSE = 1, glitch suppression prevents the USART from being accidentally started. Any low-level on URXDx shorter than the deglitch time t_τ (approximately 300 ns) will be ignored by the USART and no interrupt request will be generated as shown in Figure 13−12. See the device-specific datasheet for parameters.

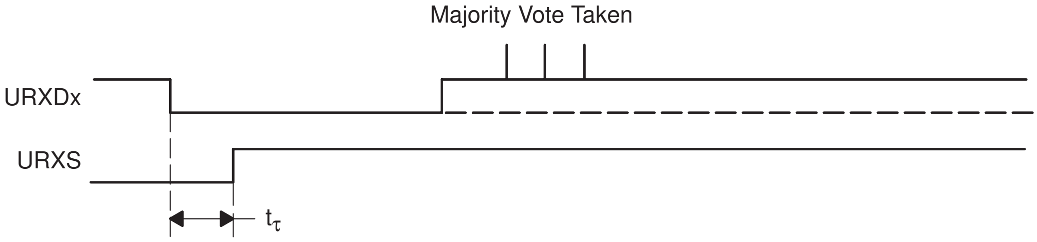
Figure 13−12. Glitch Suppression, USART Receive Not Started



When a glitch is longer than $t_{\tau,}$ or a valid start bit occurs on URXDx, the USART receive operation is started and a majority vote is taken as shown in Figure 13−13. If the majority vote fails to detect a start bit the USART halts character reception.

If character reception is halted, an active BRCLK is not necessary. A time-out period longer than the character receive duration can be used by software to indicate that a character was not received in the expected time and the software can disable BRCLK.

Figure 13−13. Glitch Suppression, USART Activated



13.3 USART Registers: UART Mode

Table 13−3 lists the registers for all devices implementing a USART module. Table 13−4 applies only to devices with a second USART module, USART1.

Table 13−3.USART0 Control and Status Registers

Register	Short Form	Register Type	Address	Initial State
USART control register	U0CTL	Read/write	070h	001h with PUC
Transmit control register	U0TCTL	Read/write	071h	001h with PUC
Receive control register	U0RCTL	Read/write	072h	000h with PUC
Modulation control register	U0MCTL	Read/write	073h	Unchanged
Baud rate control register 0	U0BR0	Read/write	074h	Unchanged
Baud rate control register 1	U0BR1	Read/write	075h	Unchanged
Receive buffer register	U0RXBUF	Read	076h	Unchanged
Transmit buffer register	U0TXBUF	Read/write	077h	Unchanged
SFR module enable register 1†	ME1	Read/write	004h	000h with PUC
SFR interrupt enable register 1†	IE1	Read/write	000h	000h with PUC
SFR interrupt flag register 1†	IFG1	Read/write	002h	082h with PUC

† Does not apply to ’12xx devices. Refer to the register definitions for registers and bit positions for these devices.

Table 13−4.USART1 Control and Status Registers

Register	Short Form	Register Type	Address	Initial State
USART control register	U1CTL	Read/write	078h	001h with PUC
Transmit control register	U1TCTL	Read/write	079h	001h with PUC
Receive control register	U1RCTL	Read/write	07Ah	000h with PUC
Modulation control register	U1MCTL	Read/write	07Bh	Unchanged
Baud rate control register 0	U1BR0	Read/write	07Ch	Unchanged
Baud rate control register 1	U1BR1	Read/write	07Dh	Unchanged
Receive buffer register	U1RXBUF	Read	07Eh	Unchanged
Transmit buffer register	U1TXBUF	Read/write	07Fh	Unchanged
SFR module enable register 2	ME2	Read/write	005h	000h with PUC
SFR interrupt enable register 2	IE2	Read/write	001h	000h with PUC
SFR interrupt flag register 2	IFG2	Read/write	003h	020h with PUC

Note: Modifying SFR bits

To avoid modifying control bits of other modules, it is recommended to set or clear the IEx and IFGx bits using `BIS.B` or `BIC.B` instructions, rather than `MOV.B` or `CLR.B` instructions.

UxCTL, USART Control Register

7	6	5	4	3	2	1	0
PENA	PEV	SPB	CHAR	LISTEN	SYNC	MM	SWRST
rw–0	rw–0	rw–0	rw–0	rw–0	rw–0	rw–0	rw–1

PENA Bit 7 Parity enable
0 Parity disabled.
1 Parity enabled. Parity bit is generated (UTXDx) and expected (URXDx). In address-bit multiprocessor mode, the address bit is included in the parity calculation.

PEV Bit 6 Parity select. PEV is not used when parity is disabled.
0 Odd parity
1 Even parity

SPB Bit 5 Stop bit select. Number of stop bits transmitted. The receiver always checks for one stop bit.
0 One stop bit
1 Two stop bits

CHAR Bit 4 Character length. Selects 7-bit or 8-bit character length.
0 7-bit data
1 8-bit data

LISTEN Bit 3 Listen enable. The LISTEN bit selects loopback mode.
0 Disabled
1 Enabled. UTXDx is internally fed back to the receiver.

SYNC Bit 2 Synchronous mode enable
0 UART mode
1 SPI Mode

MM Bit 1 Multiprocessor mode select
0 Idle-line multiprocessor protocol
1 Address-bit multiprocessor protocol

SWRST Bit 0 Software reset enable
0 Disabled. USART reset released for operation
1 Enabled. USART logic held in reset state

UxTCTL, USART Transmit Control Register

7	6	5	4	3	2	1	0
Unused	**CKPL**	**SSELx**		**URXSE**	**TXWAKE**	**Unused**	**TXEPT**
rw–0	rw–0	rw–0	rw–0	rw–0	rw–0	rw–0	rw–1

Unused Bit 7 Unused

CKPL Bit 6 Clock polarity select
0 UCLKI = UCLK
1 UCLKI = inverted UCLK

SSELx Bits 5-4 Source select. These bits select the BRCLK source clock.
00 UCLKI
01 ACLK
10 SMCLK
11 SMCLK

URXSE Bit 3 UART receive start-edge. The bit enables the UART receive start-edge feature.
0 Disabled
1 Enabled

TXWAKE Bit 2 Transmitter wake
0 Next character transmitted is data
1 Next character transmitted is an address

Unused Bit 1 Unused

TXEPT Bit 0 Transmitter empty flag
0 UART is transmitting data and/or data is waiting in UxTXBUF
1 Transmitter shift register and UxTXBUF are empty or SWRST=1

UxRCTL, USART Receive Control Register

7	6	5	4	3	2	1	0
FE	**PE**	**OE**	**BRK**	**URXEIE**	**URXWIE**	**RXWAKE**	**RXERR**
rw–0	rw–0	rw–0	rw–0	rw–0	rw–0	rw–0	rw–0

FE Bit 7 Framing error flag
0 No error
1 Character received with low stop bit

PE Bit 6 Parity error flag. When PENA = 0, PE is read as 0.
0 No error
1 Character received with parity error

OE Bit 5 Overrun error flag. This bit is set when a character is transferred into UxRXBUF before the previous character was read.
0 No error
1 Overrun error occurred

BRK Bit 4 Break detect flag
0 No break condition
1 Break condition occurred

URXEIE Bit 3 Receive erroneous-character interrupt-enable
0 Erroneous characters rejected and URXIFGx is not set
1 Erroneous characters received will set URXIFGx

URXWIE Bit 2 Receive wake-up interrupt-enable. This bit enables URXIFGx to be set when an address character is received. When URXEIE = 0, an address character will not set URXIFGx if it is received with errors.
0 All received characters set URXIFGx
1 Only received address characters set URXIFGx

RXWAKE Bit 1 Receive wake-up flag
0 Received character is data
1 Received character is an address

RXERR Bit 0 Receive error flag. This bit indicates a character was received with error(s). When RXERR = 1, on or more error flags (FE,PE,OE, BRK) is also set. RXERR is cleared when UxRXBUF is read.
0 No receive errors detected
1 Receive error detected

UxBR0, USART Baud Rate Control Register 0

7	6	5	4	3	2	1	0
2^7	**2^6**	**2^5**	**2^4**	**2^3**	**2^2**	**2^1**	**2^0**
rw	rw	rw	rw	rw	rw	rw	rw

UxBR1, USART Baud Rate Control Register 1

7	6	5	4	3	2	1	0
2^{15}	**2^{14}**	**2^{13}**	**2^{12}**	**2^{11}**	**2^{10}**	**2^9**	**2^8**
rw	rw	rw	rw	rw	rw	rw	rw

UxBRx The valid baud-rate control range is 3 ≤ UxBR < 0FFFFh, where UxBR = {UxBR1+UxBR0}. Unpredictable receive and transmit timing occurs if UxBR <3.

UxMCTL, USART Modulation Control Register

7	6	5	4	3	2	1	0
m7	**m6**	**m5**	**m4**	**m3**	**m2**	**m1**	**m0**
rw	rw	rw	rw	rw	rw	rw	rw

UxMCTLx Bits 7–0 Modulation bits. These bits select the modulation for BRCLK.

UxRXBUF, USART Receive Buffer Register

7	6	5	4	3	2	1	0
2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
r	r	r	r	r	r	r	r

UxRXBUFx Bits 7–0 The receive-data buffer is user accessible and contains the last received character from the receive shift register. Reading UxRXBUF resets the receive-error bits, the RXWAKE bit, and URXIFGx. In 7-bit data mode, UxRXBUF is LSB justified and the MSB is always reset.

UxTXBUF, USART Transmit Buffer Register

7	6	5	4	3	2	1	0
2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
rw	rw	rw	rw	rw	rw	rw	rw

UxTXBUFx Bits 7–0 The transmit data buffer is user accessible and holds the data waiting to be moved into the transmit shift register and transmitted on UTXDx. Writing to the transmit data buffer clears UTXIFGx. The MSB of UxTXBUF is not used for 7-bit data and is reset.

ME1, Module Enable Register 1

7	6	5	4	3	2	1	0
UTXE0†	**URXE0†**						
rw–0	rw–0						

UTXE0† Bit 7 USART0 transmit enable. This bit enables the transmitter for USART0.
0 Module not enabled
1 Module enabled

URXE0† Bit 6 USART0 receive enable. This bit enables the receiver for USART0.
0 Module not enabled
1 Module enabled

Bits 5-0 These bits may be used by other modules. See device-specific datasheet.

† Does not apply to MSP430x12xx devices. See ME2 for the MSP430x12xx USART0 module enable bits

ME2, Module Enable Register 2

7	6	5	4	3	2	1	0
		UTXE1	**URXE1**			**UTXE0‡**	**URXE0‡**
		rw–0	rw–0			rw–0	rw–0

Bits 7-6 These bits may be used by other modules. See device-specific datasheet.

UTXE1 Bit 5 USART1 transmit enable. This bit enables the transmitter for USART1.
0 Module not enabled
1 Module enabled

URXE1 Bit 4 USART1 receive enable. This bit enables the receiver for USART1.
0 Module not enabled
1 Module enabled

Bits 3-2 These bits may be used by other modules. See device-specific datasheet.

UTXE0‡ Bit 1 USART0 transmit enable. This bit enables the transmitter for USART0.
0 Module not enabled
1 Module enabled

URXE0‡ Bit 0 USART0 receive enable. This bit enables the receiver for USART0.
0 Module not enabled
1 Module enabled

‡ MSP430x12xx devices only

IE1, Interrupt Enable Register 1

7	6	5	4	3	2	1	0
UTXIE0†	**URXIE0†**						
rw–0	rw–0						

UTXIE0† Bit 7 USART0 transmit interrupt enable. This bit enables the UTXIFG0 interrupt.
0 Interrupt not enabled
1 Interrupt enabled

URXIE0† Bit 6 USART0 receive interrupt enable. This bit enables the URXIFG0 interrupt.
0 Interrupt not enabled
1 Interrupt enabled

Bits 5-0 These bits may be used by other modules. See device-specific datasheet.

† Does not apply to MSP430x12xx devices. See IE2 for the MSP430x12xx USART0 interrupt enable bits

IE2, Interrupt Enable Register 2

7	6	5	4	3	2	1	0
		UTXIE1	**URXIE1**			**UTXIE0‡**	**URXIE0‡**
		rw–0	rw–0			rw–0	rw–0

Bits 7-6 These bits may be used by other modules. See device-specific datasheet.

UTXIE1 Bit 5 USART1 transmit interrupt enable. This bit enables the UTXIFG1 interrupt.
0 Interrupt not enabled
1 Interrupt enabled

URXIE1 Bit 4 USART1 receive interrupt enable. This bit enables the URXIFG1 interrupt.
0 Interrupt not enabled
1 Interrupt enabled

Bits 3-2 These bits may be used by other modules. See device-specific datasheet.

UTXIE0‡ Bit 1 USART0 transmit interrupt enable. This bit enables the UTXIFG0 interrupt.
0 Interrupt not enabled
1 Interrupt enabled

URXIE0‡ Bit 0 USART0 receive interrupt enable. This bit enables the URXIFG0 interrupt.
0 Interrupt not enabled
1 Interrupt enabled

‡ MSP430x12xx devices only

IFG1, Interrupt Flag Register 1

7	6	5	4	3	2	1	0
UTXIFG0†	URXIFG0†						
rw–1	rw–0						

UTXIFG0† Bit 7 USART0 transmit interrupt flag. UTXIFG0 is set when U0TXBUF is empty.
0 No interrupt pending
1 Interrupt pending

URXIFG0† Bit 6 USART0 receive interrupt flag. URXIFG0 is set when U0RXBUF has received a complete character.
0 No interrupt pending
1 Interrupt pending

Bits 5-0 These bits may be used by other modules. See device-specific datasheet.

† Does not apply to MSP430x12xx devices. See IFG2 for the MSP430x12xx USART0 interrupt flag bits

IFG2, Interrupt Flag Register 2

7	6	5	4	3	2	1	0
		UTXIFG1	URXIFG1			UTXIFG0‡	URXIFG0‡
		rw–1	rw–0			rw–1	rw–0

Bits 7-6 These bits may be used by other modules. See device-specific datasheet.

UTXIFG1 Bit 5 USART1 transmit interrupt flag. UTXIFG1 is set when U1TXBUF empty.
0 No interrupt pending
1 Interrupt pending

URXIFG1 Bit 4 USART1 receive interrupt flag. URXIFG1 is set when U1RXBUF has received a complete character.
0 No interrupt pending
1 Interrupt pending

Bits 3-2 These bits may be used by other modules. See device-specific datasheet.

UTXIFG0‡	Bit 1	USART0 transmit interrupt flag. UTXIFG0 is set when U0TXBUF is empty. 0 No interrupt pending 1 Interrupt pending
URXIFG0‡	Bit 0	USART0 receive interrupt flag. URXIFG0 is set when U0RXBUF has received a complete character. 0 No interrupt pending 1 Interrupt pending

‡ MSP430x12xx devices only

Chapter 14

USART Peripheral Interface, SPI Mode

The universal synchronous/asynchronous receive/transmit (USART) peripheral interface supports two serial modes with one hardware module. This chapter discusses the operation of the synchronous peripheral interface or SPI mode. USART0 is implemented on the MSP430x12xx, MSP430x13xx, and MSP430x15x devices. In addition to USART0, the MSP430x14x and MSP430x16x devices implement a second identical USART module, USART1.

Topic **Page**