

Universal Serial Communication Interface, SPI Mode

The universal serial communication interface (USCI) supports multiple serial communication modes with one hardware module. This chapter discusses the operation of the synchronous peripheral interface or SPI mode.

Topic	Page
20.1 USCI Overview	20-2
20.2 USCI Introduction: SPI Mode	20-3
20.3 USCI Operation: SPI Mode	20-5
20.4 USCI Registers: SPI Mode	20-14

20.1 USCI Overview

The universal serial communication interface (USCI) modules support multiple serial communication modes. Different USCI modules support different modes. Each different USCI module is named with a different letter. For example, USCI_A is different from USCI_B, etc. If more than one identical USCI module is implemented on one device, those modules are named with incrementing numbers. For example, if one device has two USCI_A modules, they are named USCI_A0 and USCI_A1. See the device-specific data sheet to determine which USCI modules, if any, are implemented on which devices.

The USCI_Ax modules support:

- UART mode
- Pulse shaping for IrDA communications
- Automatic baud rate detection for LIN communications
- SPI mode

The USCI_Bx modules support:

- I²C mode
- SPI mode

20.2 USCI Introduction: SPI Mode

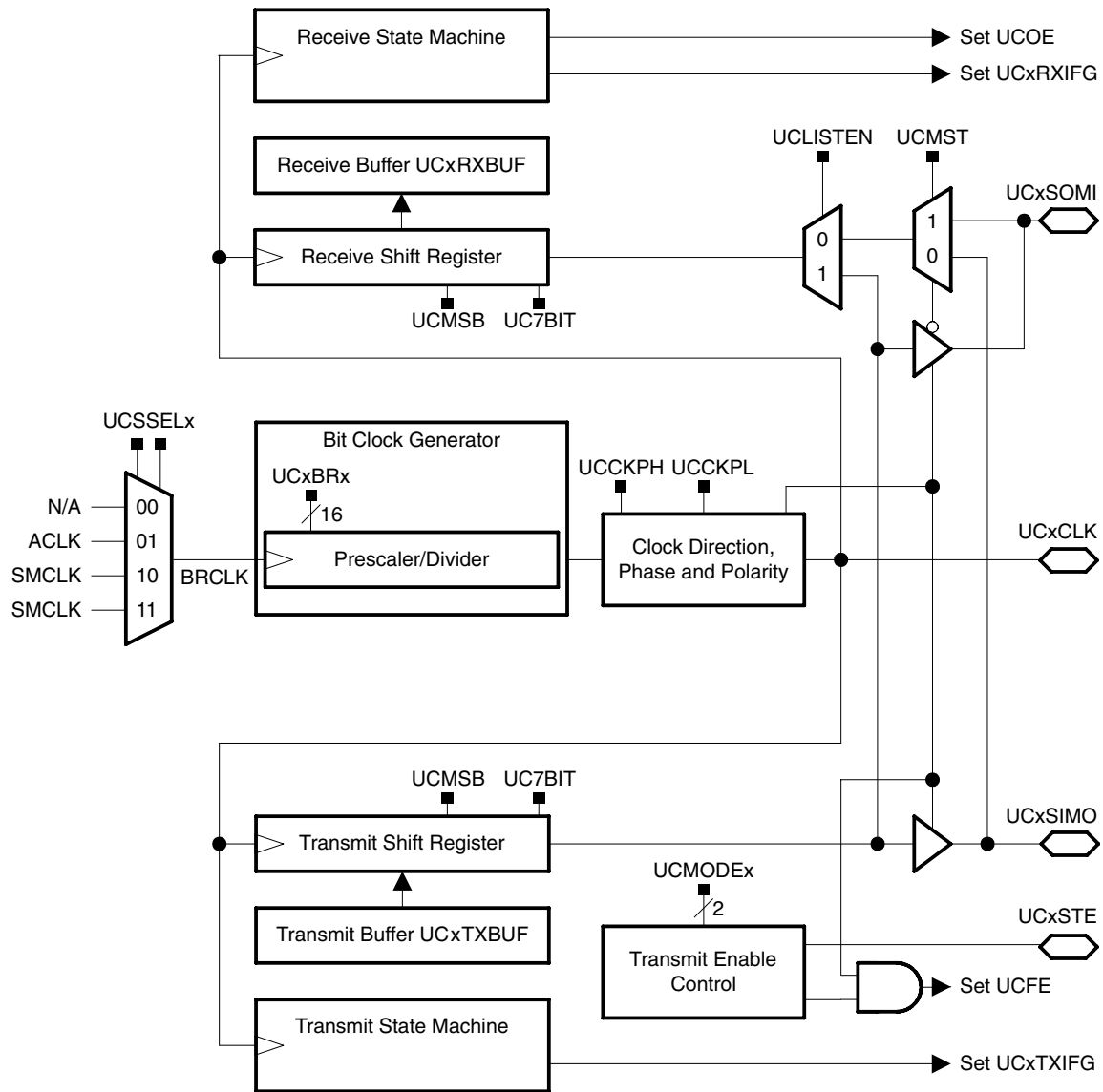
In synchronous mode, the USCI connects the MSP430 to an external system via three or four pins: UCxSIMO, UCxSOMI, UCxCLK, and UCxSTE. SPI mode is selected when the UCSYNC bit is set and SPI mode (3-pin or 4-pin) is selected with the UCMODEx bits.

SPI mode features include:

- 7- or 8-bit data length
- LSB-first or MSB-first data transmit and receive
- 3-pin and 4-pin SPI operation
- Master or slave modes
- Independent transmit and receive shift registers
- Separate transmit and receive buffer registers
- Continuous transmit and receive operation
- Selectable clock polarity and phase control
- Programmable clock frequency in master mode
- Independent interrupt capability for receive and transmit
- Slave operation in LPM4

Figure 20–1 shows the USCI when configured for SPI mode.

Figure 20–1. USCI Block Diagram: SPI Mode



20.3 USCI Operation: SPI Mode

In SPI mode, serial data is transmitted and received by multiple devices using a shared clock provided by the master. An additional pin, UCxSTE, is provided to enable a device to receive and transmit data and is controlled by the master.

Three or four signals are used for SPI data exchange:

- UCxSIMO Slave in, master out
Master mode: UCxSIMO is the data output line.
Slave mode: UCxSIMO is the data input line.
- UCxSOMI Slave out, master in
Master mode: UCxSOMI is the data input line.
Slave mode: UCxSOMI is the data output line.
- UCxCLK USCI SPI clock
Master mode: UCxCLK is an output.
Slave mode: UCxCLK is an input.
- UCxSTE Slave transmit enable. Used in 4-pin mode to allow multiple masters on a single bus. Not used in 3-pin mode. Table 20–1 describes the UCxSTE operation.

Table 20–1. UCxSTE Operation

UCMODEx	UCxSTE Active State	UCxSTE	Slave	Master
01	high	0	inactive	active
		1	active	inactive
10	low	0	active	inactive
		1	inactive	active

20.3.1 USCI Initialization and Reset

The USCI is reset by a PUC or by the UCSWRST bit. After a PUC, the UCSWRST bit is automatically set, keeping the USCI in a reset condition. When set, the UCSWRST bit resets the UCxRXIE, UCxTXIE, UCxRXIFG, UCOE, and UCFE bits and sets the UCxTXIFG flag. Clearing UCSWRST releases the USCI for operation.

Note: Initializing or Re-Configuring the USCI Module

The recommended USCI initialization/re-configuration process is:

- 1) Set UCSWRST (BIS.B #UCSWRST, &UCxCTL1)
 - 2) Initialize all USCI registers with UCSWRST=1 (including UCxCTL1)
 - 3) Configure ports.
 - 4) Clear UCSWRST via software (BIC.B #UCSWRST, &UCxCTL1)
 - 5) Enable interrupts (optional) via UCxRXIE and/or UCxTXIE
-

20.3.2 Character Format

The USCI module in SPI mode supports 7- and 8-bit character lengths selected by the UC7BIT bit. In 7-bit data mode, UCxRXBUF is LSB justified and the MSB is always reset. The UCMSB bit controls the direction of the transfer and selects LSB or MSB first.

Note: Default Character Format

The default SPI character transmission is LSB first. For communication with other SPI interfaces it MSB-first mode may be required.

Note: Character Format for Figures

Figures throughout this chapter use MSB first format.

20.3.3 Master Mode

Figure 20–2. USCI Master and External Slave

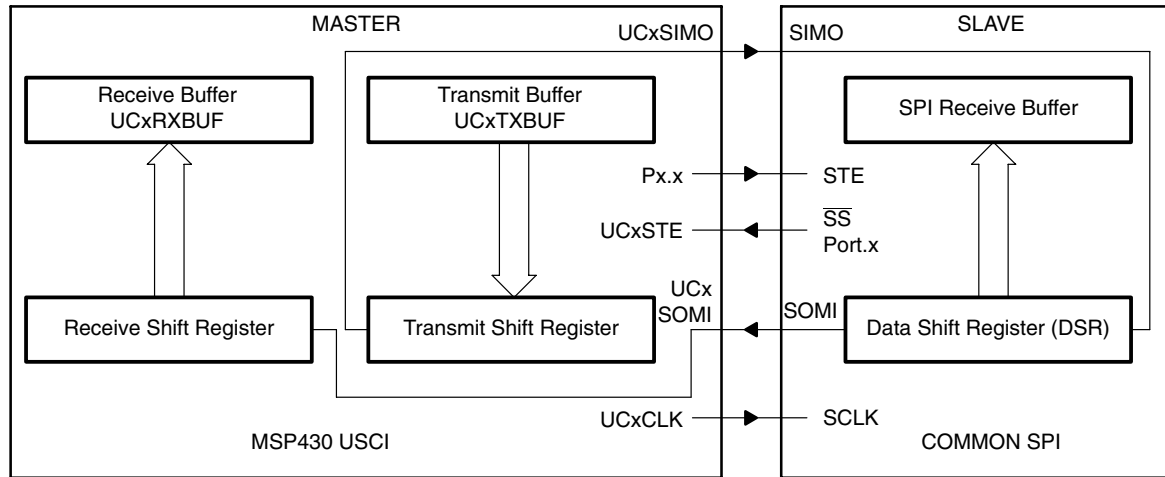


Figure 20–2 shows the USCI as a master in both 3-pin and 4-pin configurations. The USCI initiates data transfer when data is moved to the transmit data buffer UCxTXBUF. The UCxTXBUF data is moved to the TX shift register when the TX shift register is empty, initiating data transfer on UCxSIMO starting with either the most-significant or least-significant bit depending on the UCMSB setting. Data on UCxSOMI is shifted into the receive shift register on the opposite clock edge. When the character is received, the receive data is moved from the RX shift register to the received data buffer UCxRXBUF and the receive interrupt flag, UCxRXIFG, is set, indicating the RX/TX operation is complete.

A set transmit interrupt flag, UCxTXIFG, indicates that data has moved from UCxTXBUF to the TX shift register and UCxTXBUF is ready for new data. It does not indicate RX/TX completion.

To receive data into the USCI in master mode, data must be written to UCxTXBUF because receive and transmit operations operate concurrently.

Four-Pin SPI Master Mode

In 4-pin master mode, UCxSTE is used to prevent conflicts with another master and controls the master as described in Table 20–1. When UCxSTE is in the master-inactive state:

- UCxSIMO and UCxCLK are set to inputs and no longer drive the bus
- The error bit UCFE is set indicating a communication integrity violation to be handled by the user.
- The internal state machines are reset and the shift operation is aborted.

If data is written into UCxTXBUF while the master is held inactive by UCxSTE, it will be transmit as soon as UCxSTE transitions to the master-active state. If an active transfer is aborted by UCxSTE transitioning to the master-inactive state, the data must be re-written into UCxTXBUF to be transferred when UCxSTE transitions back to the master-active state. The UCxSTE input signal is not used in 3-pin master mode.

20.3.4 Slave Mode

Figure 20–3. USCI Slave and External Master

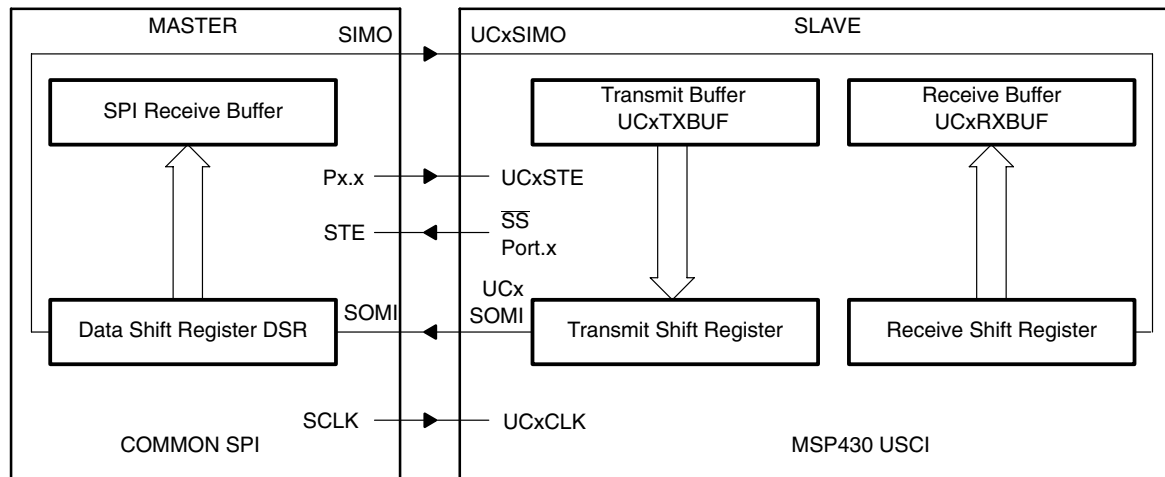


Figure 20–3 shows the USCI as a slave in both 3-pin and 4-pin configurations. UCxCLK is used as the input for the SPI clock and must be supplied by the external master. The data-transfer rate is determined by this clock and not by the internal bit clock generator. Data written to UCxTXBUF and moved to the TX shift register before the start of UCxCLK is transmitted on UCxSOMI. Data on UCxSIMO is shifted into the receive shift register on the opposite edge of UCxCLK and moved to UCxRXBUF when the set number of bits are received. When data is moved from the RX shift register to UCxRXBUF, the UCxRXIFG interrupt flag is set, indicating that data has been received. The overrun error bit, UCOE, is set when the previously received data is not read from UCxRXBUF before new data is moved to UCxRXBUF.

Four-Pin SPI Slave Mode

In 4-pin slave mode, UCxSTE is used by the slave to enable the transmit and receive operations and is provided by the SPI master. When UCxSTE is in the slave-active state, the slave operates normally. When UCxSTE is in the slave-inactive state:

- Any receive operation in progress on UCxSIMO is halted
- UCxSOMI is set to the input direction
- The shift operation is halted until the UCxSTE line transitions into the slave transmit active state.

The UCxSTE input signal is not used in 3-pin slave mode.

20.3.5 SPI Enable

When the USCI module is enabled by clearing the UCSWRST bit it is ready to receive and transmit. In master mode the bit clock generator is ready, but is not clocked nor producing any clocks. In slave mode the bit clock generator is disabled and the clock is provided by the master.

A transmit or receive operation is indicated by UCBUSY = 1.

A PUC or set UCSWRST bit disables the USCI immediately and any active transfer is terminated.

Transmit Enable

In master mode, writing to UCxTXBUF activates the bit clock generator and the data will begin to transmit.

In slave mode, transmission begins when a master provides a clock and, in 4-pin mode, when the UCxSTE is in the slave-active state.

Receive Enable

The SPI receives data when a transmission is active. Receive and transmit operations operate concurrently.

20.3.6 Serial Clock Control

UCxCLK is provided by the master on the SPI bus. When UCMST = 1, the bit clock is provided by the USCI bit clock generator on the UCxCLK pin. The clock used to generate the bit clock is selected with the UCSSELx bits. When UCMST = 0, the USCI clock is provided on the UCxCLK pin by the master, the bit clock generator is not used, and the UCSSELx bits are don't care. The SPI receiver and transmitter operate in parallel and use the same clock source for data transfer.

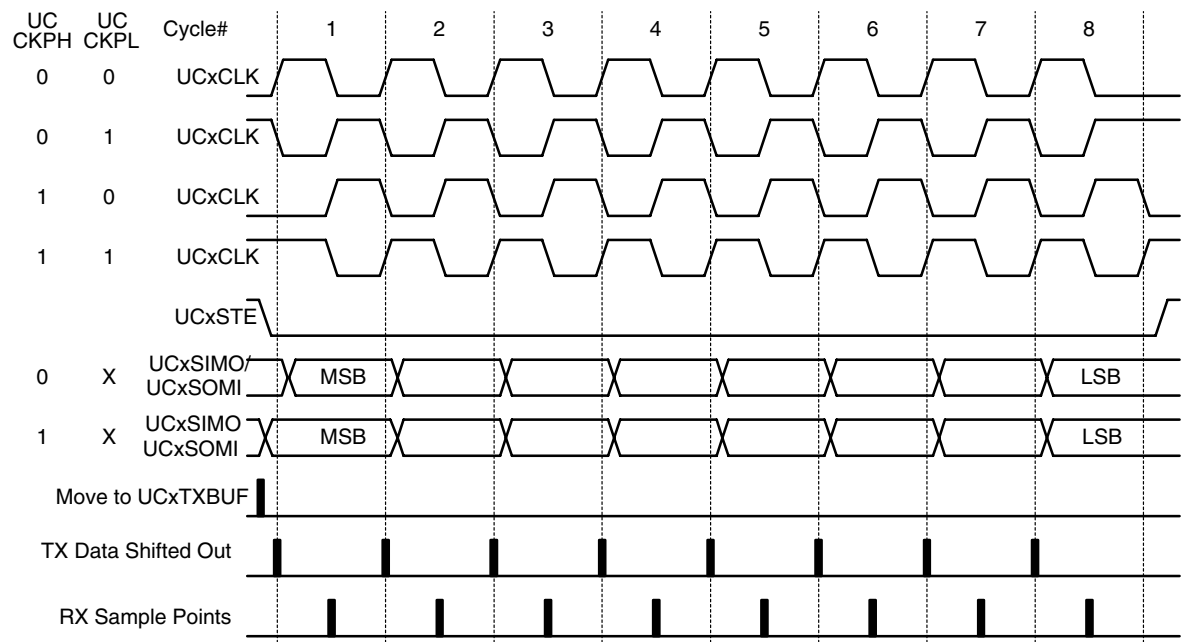
The 16-bit value of UCBRx in the bit rate control registers UCxxBR1 and UCxxBR0 is the division factor of the USCI clock source, BRCLK. The maximum bit clock that can be generated in master mode is BRCLK. Modulation is not used in SPI mode and UCAxMCTL should be cleared when using SPI mode for USCI_A. The UCAxCLK/UCBxCLK frequency is given by:

$$f_{\text{BitClock}} = \frac{f_{\text{BRCLK}}}{\text{UCBRx}}$$

Serial Clock Polarity and Phase

The polarity and phase of UCxCLK are independently configured via the UCCKPL and UCCKPH control bits of the USCI. Timing for each case is shown in Figure 20–4.

Figure 20–4. USCI SPI Timing with UCMSB = 1



20.3.7 Using the SPI Mode with Low Power Modes

The USCI module provides automatic clock activation for SMCLK for use with low-power modes. When SMCLK is the USCI clock source, and is inactive because the device is in a low-power mode, the USCI module automatically activates it when needed, regardless of the control-bit settings for the clock source. The clock remains active until the USCI module returns to its idle condition. After the USCI module returns to the idle condition, control of the clock source reverts to the settings of its control bits. Automatic clock activation is not provided for ACLK.

When the USCI module activates an inactive clock source, the clock source becomes active for the whole device and any peripheral configured to use the clock source may be affected. For example, a timer using SMCLK will increment while the USCI module forces SMCLK active.

In SPI slave mode no internal clock source is required because the clock is provided by the external master. It is possible to operate the USCI in SPI slave mode while the device is in LPM4 and all clock sources are disabled. The receive or transmit interrupt can wake up the CPU from any low power mode.

20.3.8 SPI Interrupts

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

SPI Transmit Interrupt Operation

The UCxTXIFG interrupt flag is set by the transmitter to indicate that UCxTXBUF is ready to accept another character. An interrupt request is generated if UCxTXIE and GIE are also set. UCxTXIFG is automatically reset if a character is written to UCxTXBUF. UCxTXIFG is set after a PUC or when UCSWRST = 1. UCxTXIE is reset after a PUC or when UCSWRST = 1.

Note: Writing to UCxTXBUF in SPI Mode

Data written to UCxTXBUF when UCxTXIFG = 0 may result in erroneous data transmission.

SPI Receive Interrupt Operation

The UCxRXIFG interrupt flag is set each time a character is received and loaded into UCxRXBUF. An interrupt request is generated if UCxRXIE and GIE are also set. UCxRXIFG and UCxRXIE are reset by a system reset PUC signal or when UCSWRST = 1. UCxRXIFG is automatically reset when UCxRXBUF is read.

USCI Interrupt Usage

USCI_Ax and USCI_Bx share the same interrupt vectors. The receive interrupt flags UCAxRXIFG and UCBxRXIFG are routed to one interrupt vector, the transmit interrupt flags UCAxTXIFG and UCBxTXIFG share another interrupt vector.

Shared Interrupt Vectors Software Example

The following software example shows an extract of an interrupt service routine to handle data receive interrupts from USCI_A0 in either UART or SPI mode and USCI_B0 in SPI mode.

```

USCIA0_RX_USCIB0_RX_ISR
    BIT.B #UCA0RXIFG, &IFG2 ; USCI_A0 Receive Interrupt?
    JNZ    USCIA0_RX_ISR
USCIB0_RX_ISR?
    ; Read UCB0RXBUF (clears UCB0RXIFG)
    ...
    RETI
USCIA0_RX_ISR
    ; Read UCA0RXBUF (clears UCA0RXIFG)
    ...
    RETI

```

The following software example shows an extract of an interrupt service routine to handle data transmit interrupts from USCI_A0 in either UART or SPI mode and USCI_B0 in SPI mode.

```

USCIA0_TX_USCIB0_TX_ISR
    BIT.B #UCA0TXIFG, &IFG2 ; USCI_A0 Transmit Interrupt?
    JNZ    USCIA0_TX_ISR
USCIB0_TX_ISR
    ; Write UCB0TXBUF (clears UCB0TXIFG)
    ...
    RETI
USCIA0_TX_ISR
    ; Write UCA0TXBUF (clears UCA0TXIFG)
    ...
    RETI

```

20.4 USCI Registers: SPI Mode

The USCI registers applicable in SPI mode for USCI_A0 and USCI_B0 are listed in Table 20–2. Registers applicable in SPI mode for USCI_A1 and USCI_B1 are listed in Table 20–3.

Table 20–2. USCI_A0 and USCI_B0 Control and Status Registers

Register	Short Form	Register Type	Address	Initial State
USCI_A0 control register 0	UCA0CTL0	Read/write	060h	Reset with PUC
USCI_A0 control register 1	UCA0CTL1	Read/write	061h	001h with PUC
USCI_A0 Baud rate control register 0	UCA0BR0	Read/write	062h	Reset with PUC
USCI_A0 Baud rate control register 1	UCA0BR1	Read/write	063h	Reset with PUC
USCI_A0 modulation control register	UCA0MCTL	Read/write	064h	Reset with PUC
USCI_A0 status register	UCA0STAT	Read/write	065h	Reset with PUC
USCI_A0 Receive buffer register	UCA0RXBUF	Read	066h	Reset with PUC
USCI_A0 Transmit buffer register	UCA0TXBUF	Read/write	067h	Reset with PUC
USCI_B0 control register 0	UCB0CTL0	Read/write	068h	001h with PUC
USCI_B0 control register 1	UCB0CTL1	Read/write	069h	001h with PUC
USCI_B0 Bit rate control register 0	UCB0BR0	Read/write	06Ah	Reset with PUC
USCI_B0 Bit rate control register 1	UCB0BR1	Read/write	06Bh	Reset with PUC
USCI_B0 status register	UCB0STAT	Read/write	06Dh	Reset with PUC
USCI_B0 Receive buffer register	UCB0RXBUF	Read	06Eh	Reset with PUC
USCI_B0 Transmit buffer register	UCB0TXBUF	Read/write	06Fh	Reset with PUC
SFR interrupt enable register 2	IE2	Read/write	001h	Reset with PUC
SFR interrupt flag register 2	IFG2	Read/write	003h	00Ah with PUC

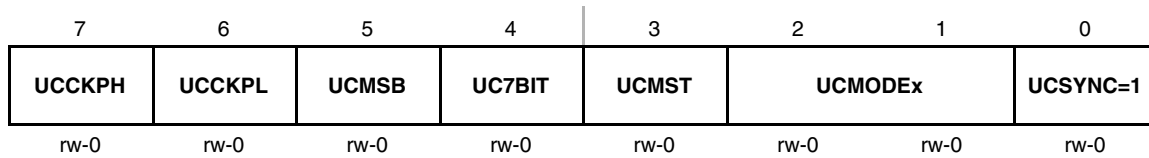
Note: Modifying SFR bits

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.

Table 20–3. USCI_A1 and USCI_B1 Control and Status Registers

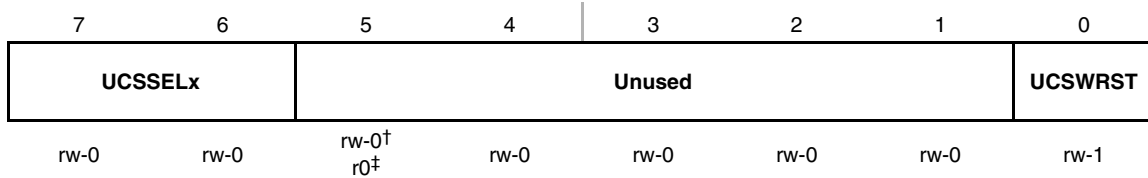
Register	Short Form	Register Type	Address	Initial State
USCI_A1 control register 0	UCA1CTL0	Read/write	0D0h	Reset with PUC
USCI_A1 control register 1	UCA1CTL1	Read/write	0D1h	001h with PUC
USCI_A1 Baud rate control register 0	UCA1BR0	Read/write	0D2h	Reset with PUC
USCI_A1 Baud rate control register 1	UCA1BR1	Read/write	0D3h	Reset with PUC
USCI_A1 modulation control register	UCA1MCTL	Read/write	0D4h	Reset with PUC
USCI_A1 status register	UCA1STAT	Read/write	0D5h	Reset with PUC
USCI_A1 Receive buffer register	UCA1RXBUF	Read	0D6h	Reset with PUC
USCI_A1 Transmit buffer register	UCA1TXBUF	Read/write	0D7h	Reset with PUC
USCI_B1 control register 0	UCB1CTL0	Read/write	0D8h	001h with PUC
USCI_B1 control register 1	UCB1CTL1	Read/write	0D9h	001h with PUC
USCI_B1 Bit rate control register 0	UCB1BR0	Read/write	0DAh	Reset with PUC
USCI_B1 Bit rate control register 1	UCB1BR1	Read/write	0DBh	Reset with PUC
USCI_B1 status register	UCB1STAT	Read/write	0DDh	Reset with PUC
USCI_B1 Receive buffer register	UCB1RXBUF	Read	0DEh	Reset with PUC
USCI_B1 Transmit buffer register	UCB1TXBUF	Read/write	0DFh	Reset with PUC
USCI_A1/B1 interrupt enable register	UC1IE	Read/write	006h	Reset with PUC
USCI_A1/B1 interrupt flag register	UC1IFG	Read/write	007h	00Ah with PUC

UCAxCTL0, USCI_Ax Control Register 0
UCBxCTL0, USCI_Bx Control Register 0



- UCCKPH** Bit 7 Clock phase select.
 0 Data is changed on the first UCLK edge and captured on the following edge.
 1 Data is captured on the first UCLK edge and changed on the following edge.
- UCCKPL** Bit 6 Clock polarity select.
 0 The inactive state is low.
 1 The inactive state is high.
- UCMSB** Bit 5 MSB first select. Controls the direction of the receive and transmit shift register.
 0 LSB first
 1 MSB first
- UC7BIT** Bit 4 Character length. Selects 7-bit or 8-bit character length.
 0 8-bit data
 1 7-bit data
- UCMST** Bit 3 Master mode select
 0 Slave mode
 1 Master mode
- UCMODEx** Bits 2-1 USCI Mode. The UCMODEx bits select the synchronous mode when UCSYNC = 1.
 00 3-Pin SPI
 01 4-Pin SPI with UCxSTE active high: slave enabled when UCxSTE = 1
 10 4-Pin SPI with UCxSTE active low: slave enabled when UCxSTE = 0
 11 I²C Mode
- UCSYNC** Bit 0 Synchronous mode enable
 0 Asynchronous mode
 1 Synchronous Mode

UCAxCTL1, USCI_Ax Control Register 1
UCBxCTL1, USCI_Bx Control Register 1

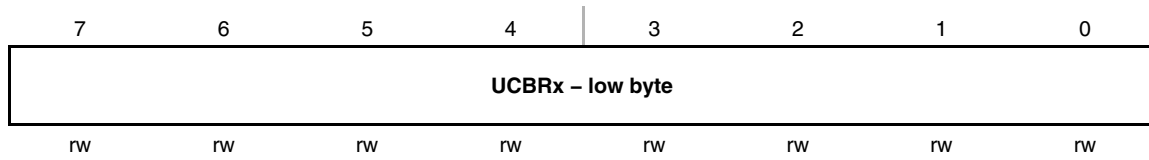


[†] UCAxCTL1 (USCI_Ax)

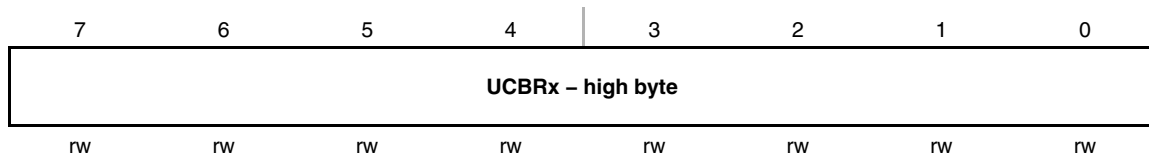
[‡] UCBxCTL1 (USCI_Bx)

UCSSELx	Bits 7-6	USCI clock source select. These bits select the BRCLK source clock in master mode. UCxCLK is always used in slave mode. 00 NA 01 ACLK 10 SMCLK 11 SMCLK
Unused	Bits 5-1	Unused in synchronous mode (UCSYNC=1).
UCSWRST	Bit 0	Software reset enable 0 Disabled. USCI reset released for operation. 1 Enabled. USCI logic held in reset state.

UCAxBR0, USCI_Ax Bit Rate Control Register 0
UCBxBR0, USCI_Bx Bit Rate Control Register 0



UCAxBR1, USCI_Ax Bit Rate Control Register 1
UCBxBR1, USCI_Bx Bit Rate Control Register 1



UCBRx

Bit clock prescaler setting.
The 16-bit value of {UCxxBR0+UCxxBR1×256} form the prescaler value.

UCAxSTAT, USCI_Ax Status Register
UCBxSTAT, USCI_Bx Status Register

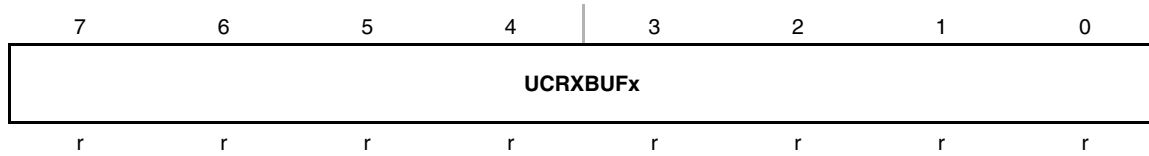
7	6	5	4	3	2	1	0
UCLISTEN	UCFE	UCOE	Unused	Unused	Unused	Unused	UCBUSY
rw-0	rw-0	rw-0	rw-0 [†] r0 [‡]	rw-0	rw-0	rw-0	r-0

[†] UCAxSTAT (USCI_Ax)

[‡] UCBxSTAT (USCI_Bx)

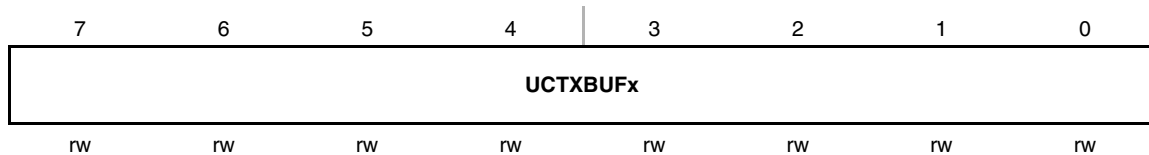
UCLISTEN	Bit 7	Listen enable. The UCLISTEN bit selects loopback mode. 0 Disabled 1 Enabled. The transmitter output is internally fed back to the receiver.
UCFE	Bit 6	Framing error flag. This bit indicates a bus conflict in 4-wire master mode. UCFE is not used in 3-wire master or any slave mode. 0 No error 1 Bus conflict occurred
UCOE	Bit 5	Overrun error flag. This bit is set when a character is transferred into UCxRXBUF before the previous character was read. UCOE is cleared automatically when UCxRXBUF is read, and must not be cleared by software. Otherwise, it will not function correctly. 0 No error 1 Overrun error occurred
Unused	Bits 4–1	Unused in synchronous mode (UCSYNC=1).
UCBUSY	Bit 0	USCI busy. This bit indicates if a transmit or receive operation is in progress. 0 USCI inactive 1 USCI transmitting or receiving

UCAxRXBUF, USCI_Ax Receive Buffer Register
UCBxRXBUF, USCI_Bx Receive Buffer Register



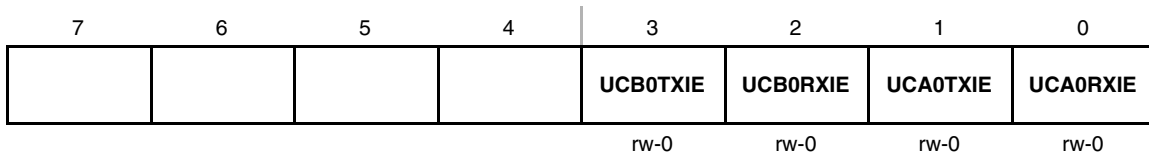
UCRXBUFx Bits 7-0 The receive-data buffer is user accessible and contains the last received character from the receive shift register. Reading UCxRXBUF resets the receive-error bits, and UCxRXIFG. In 7-bit data mode, UCxRXBUF is LSB justified and the MSB is always reset.

UCAxTXBUF, USCI_Ax Transmit Buffer Register
UCBxTXBUF, USCI_Bx Transmit Buffer Register



UCTXBUFx 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. Writing to the transmit data buffer clears UCxTXIFG. The MSB of UCxTXBUF is not used for 7-bit data and is reset.

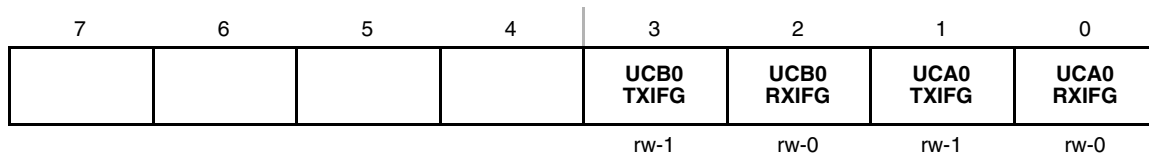
IE2, Interrupt Enable Register 2



Bits 7-4: These bits may be used by other modules. See device-specific data sheet.

- UCB0TXIE** Bit 3 USCI_B0 transmit interrupt enable
0 Interrupt disabled
1 Interrupt enabled
- UCB0RXIE** Bit 2 USCI_B0 receive interrupt enable
0 Interrupt disabled
1 Interrupt enabled
- UCA0TXIE** Bit 1 USCI_A0 transmit interrupt enable
0 Interrupt disabled
1 Interrupt enabled
- UCA0RXIE** Bit 0 USCI_A0 receive interrupt enable
0 Interrupt disabled
1 Interrupt enabled

IFG2, Interrupt Flag Register 2



Bits 7-4: These bits may be used by other modules. See device-specific data sheet.

- UCB0 TXIFG** Bit 3 USCI_B0 transmit interrupt flag. UCB0TXIFG is set when UCB0TXBUF is empty.
 0 No interrupt pending
 1 Interrupt pending
- UCB0 RXIFG** Bit 2 USCI_B0 receive interrupt flag. UCB0RXIFG is set when UCB0RXBUF has received a complete character.
 0 No interrupt pending
 1 Interrupt pending
- UCA0 TXIFG** Bit 1 USCI_A0 transmit interrupt flag. UCA0TXIFG is set when UCA0TXBUF empty.
 0 No interrupt pending
 1 Interrupt pending
- UCA0 RXIFG** Bit 0 USCI_A0 receive interrupt flag. UCA0RXIFG is set when UCA0RXBUF has received a complete character.
 0 No interrupt pending
 1 Interrupt pending

UC1IE, USCI_A1/USCI_B1 Interrupt Enable Register

7	6	5	4	3	2	1	0
Unused	Unused	Unused	Unused	UCB1TXIE	UCB1RXIE	UCA1TXIE	UCA1RXIE
rw-0	rw-0	rw-0	rw-0	rw-0	rw-0	rw-0	rw-0

Unused	Bits 7-4	Unused
UCB1TXIE	Bit 3	USCI_B1 transmit interrupt enable 0 Interrupt disabled 1 Interrupt enabled
UCB1RXIE	Bit 2	USCI_B1 receive interrupt enable 0 Interrupt disabled 1 Interrupt enabled
UCA1TXIE	Bit 1	USCI_A1 transmit interrupt enable 0 Interrupt disabled 1 Interrupt enabled
UCA1RXIE	Bit 0	USCI_A1 receive interrupt enable 0 Interrupt disabled 1 Interrupt enabled

UC1IFG, USCI_A1/USCI_B1 Interrupt Flag Register

7	6	5	4	3	2	1	0
Unused	Unused	Unused	Unused	UCB1 TXIFG	UCB1 RXIFG	UCA1 TXIFG	UCA1 RXIFG
rw-0	rw-0	rw-0	rw-0	rw-1	rw-0	rw-1	rw-0

Unused	Bits 7-4	Unused
UCB1 TXIFG	Bit 3	USCI_B1 transmit interrupt flag. UCB1TXIFG is set when UCB1TXBUF is empty. 0 No interrupt pending 1 Interrupt pending
UCB1 RXIFG	Bit 2	USCI_B1 receive interrupt flag. UCB1RXIFG is set when UCB1RXBUF has received a complete character. 0 No interrupt pending 1 Interrupt pending
UCA1 TXIFG	Bit 1	USCI_A1 transmit interrupt flag. UCA1TXIFG is set when UCA1TXBUF empty. 0 No interrupt pending 1 Interrupt pending
UCA1 RXIFG	Bit 0	USCI_A1 receive interrupt flag. UCA1RXIFG is set when UCA1RXBUF has received a complete character. 0 No interrupt pending 1 Interrupt pending