9.3.8 DMA Transfer End

The DMA transfer ending conditions vary when channels end individually and when both channels end together.

Conditions for Channels Ending Individually: When either of the following conditions are met, the transfer will end in the relevant channel only:

- The value of the channel's DMA transfer count register (TCR) becomes 0.
  When the TCR value becomes 0, the DMA transfer for that channel ends and the transfer-end flag bit (TE) is set in CHCR. If the IE (interrupt enable) bit has already been set, a DMAC interrupt (DEI) request is sent to the CPU. In 16-byte transfer, when the TCR is 3,2,1 during the final transfer, the source address will be output four times, but the destination address will only be output the number of times found in TCR before transfer ends.

- The DE bit of the DMA channel control register (CHCR) is cleared to 0.
  When the DMA enable bit (DE) in CHCR is cleared, DMA transfers in the affected channel are halted. The TE bit is not set when this happens.

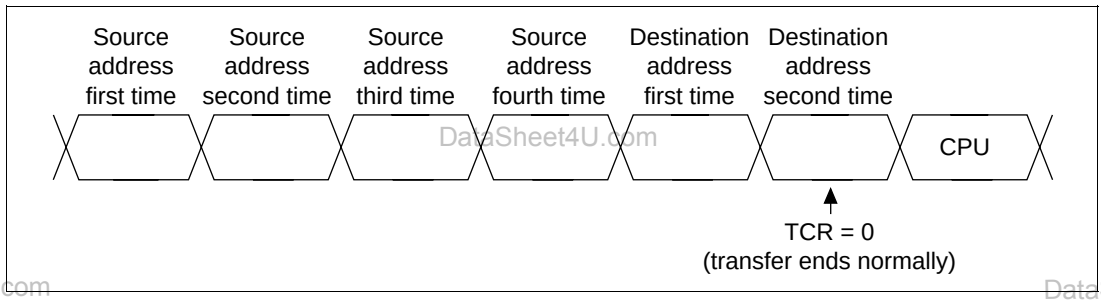


Figure 9.52 16-Byte Transfer when TCR = 2

Conditions for Both Channels Ending Simultaneously: Transfers on both channels end when either of the following conditions is met:

- The NMIF (NMI flag) bit or AE (address error flag) bit is set to 1 in DMAOR.
  When an NMI interrupt or DMAC address error occurs and the NMIF or AE bit is set to 1 in DMAOR, all channels stop their transfers. The DMA source address register (SAR), designation address register (DAR), and transfer count register (TCR) are all updated by the transfer immediately preceding the halt. When this transfer is the final transfer, TE = 1 and the transfer ends. To resume transfer after NMI interrupt exception handling or address error exception handling, clear the appropriate flag bit. When the DE bit is then set to 1, the transfer on that channel will restart. To avoid this, keep its DE bit at 0. In dual address mode, DMA transfer will be halted after the completion of the following write cycle even when the address error occurs in the initial read cycle. SAR, DAR and TCR are updated by the final transfer.

- The DMA master enable (DME) bit in DMAOR is cleared to 0.

  Clearing the DME bit in DMAOR forcibly aborts the transfers on both channels at the end of the current bus cycle. When the transfer is the final transfer, TE = 1 and the transfer ends.

9.4 Examples of Use

9.4.1 DMA Transfer Between On-Chip SCI and External Memory

In the following example, data received on the on-chip serial communication interface (SCI) is transferred to external memory using DMAC channel 1. Table 9.9 shows the transfer conditions and register settings.

Table 9.9 Register Settings for Transfers between On-Chip SCI and External Memory

Transfer Conditions	Register	Setting
Transfer source: RDR of on-chip SCI	SAR1	H'FFFFFE05
Transfer destination: external memory (word space)	DAR1	Destination address
Number of transfers: 64	TCR1	H'0040
Transfer destination address: incremented	CHCR1	H'4045
Transfer source address: fixed		
Bus mode: cycle-steal		
Transfer unit: byte		
DEI interrupt request generated at end of transfer (DE = 1)		
Channel priority: Fixed (0 > 1) (DME = 1)	DMAOR	H'0001
Transfer request source (transfer request signal): SCI (RXI)	DRCR1	H'01

Note: Check the CPU interrupt level when interrupts are enabled in the SCI.

9.5 Usage Notes

1. DMA request/response selection control registers 0 and 1 (DRCR0 and DRCR1) should be accessed in bytes. All other registers should be accessed in longword units.
2. Before rewriting CHCR0, CHCR1, DRCR0, and DRCR1, first clear the DE bit for the specified channel to 0 or clear the DME bit in DMAOR to 0.
3. When the DMAC is not operating, the NMIF bit in DMAOR is set even when an NMI interrupt is input.
4. When the cache is used as on-chip RAM, the DMAC cannot access this RAM.
5. Set to standby mode after the DME bit in DMAOR is set to 0.
6. Do not access the DMAC, BSC, and UBC on-chip peripheral modules.
7. Do not access the cache (address array, data array, associative purge area).
8. To detect the DREQ pin signal in single address mode, use edge detection.

Section 10 Division Unit

10.1 Overview

The division unit (DIVU) divides 64 bits by 32 bits and 32 bits by 32 bits. The results are expressed as a 32-bit quotient and a 32-bit remainder. When the operation produces an overflow, an interrupt can be generated as specified.

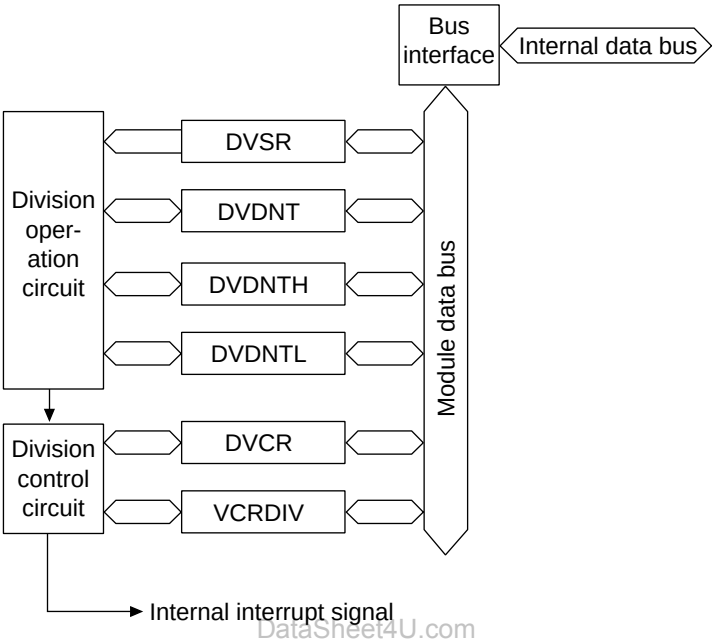
10.1.1 Features

The division unit has the following features:

- Performs signed division of 64 bits by 32 bits and 32 bits by 32 bits
- Handles 32-bit quotient, 32-bit remainder
- Completes operation execution in 39 cycles
- Controls enabling/disabling of over/underflow interrupts
- Even during the division process, instructions not accessing the division unit can be parallel-processed

10.1.2 Block Diagram

Figure 10.1 shows a block diagram of the division unit.



DVSR: Divisor register
DVDNT: Dividend register L for 32-bit division
DVDNTH: Dividend register H
DVDNTL: Dividend register L
DVCR: Division control register
VCRDIV: Vector number setting register DIV

Figure 10.1 Division Unit Block Diagram

10.1.3 Register Configuration

Table 10.1 shows the register configuration of the division unit.

Table 10.1 Division Unit Register Configuration

Register	Abbr.	R/W	Initial Value	Address	Access Size*1
Divisor register	DVSR	R/W	Undefined	H'FFFFFF00	32
Dividend register L for 32-bit division	DVDNT	R/W	Undefined	H'FFFFFF04	32
Division control register	DVCR	R/W	H'00000000	H'FFFFFF08	16, 32
Vector number setting register DIV	VCRDIV	R/W	Undefined*2	H'FFFFFF0C	16, 32
Dividend register H	DVDNTH	R/W	Undefined	H'FFFFFF10	32
Dividend register L	DVDNTL	R/W	Undefined	H'FFFFFF14	32

Notes: 1. Accesses to the division unit are read and written in 32-bit units. DVCR and VCRDIV permit 16 and 32-bit accesses. When registers other than CONT and VCRDIV are accessed with word accesses, undefined values are read or written.
2. The initial value of VCRDIV is H'0000**** (asterisks represent undefined values).

10.2 Description of Registers

10.2.1 Divisor Register (DVSR)

Bit:	31	30	29	…	3	2	1	0
Bit name:				…				
Initial value:	—	—	—	…	—	—	—	—
R/W:	R/W	R/W	R/W	…	R/W	R/W	R/W	R/W

The divisor register (DVSR) is a 32-bit read/write register in which the divisor for the operation is written. It is not initialized by a power-on reset or manual reset, in standby mode, or during module standbys.

10.2.2 Dividend Register L for 32-Bit Division (DVDNT)

Bit:	31	30	29	…	3	2	1	0
Bit name:				…				
Initial value:	—	—	—	…	—	—	—	—
R/W:	R/W	R/W	R/W	…	R/W	R/W	R/W	R/W

The dividend register L for 32-bit division (DVDNT) is a 32-bit read/write register in which the 32-bit dividend used for 32-bit ÷ 32-bit division operations is written. When 32-bit ÷ 32-bit division is run, the value set as the dividend is lost and the quotient written at the end of division. When this register is written to, the same value is written in the DVDNTL register. The MSB written is sign-extended in the DVDNTH register. Writing to this register starts the 32-bit ÷ 32-bit division operation. It is not initialized by a power-on reset or manual reset, in standby mode, or during module standbys.

10.2.3 Division Control Register (DVCR)

Bit:	31	30	29	…	3	2	1	0
Bit name:	—	—	—	…	—	—	OVFIE	OVF
Initial value:	0	0	0	…	0	0	0	0
R/W:	R	R	R	…	R	R	R/W	R/W

The division control register (DVCR) is a 32-bit read/write register, but is also 16-bit accessible. It controls enabling/disabling of the overflow interrupt. This register is initialized to H'00000000 by a power-on reset or manual reset. It is not initialized in standby mode or during module standbys.

- Bits 31 to 2: Reserved. These bits always read 0. The write value should always be 0.
- Bit 1: OVF Interrupt Enable (OVFIE): Selects enabling or disabling of the OVF interrupt request (OVFI) upon overflow.

Bit 1: OVFIE	Description	
0	Interrupt request (OVFI) caused by OVF disabled	(Initial value)
1	Interrupt request (OVFI) caused by OVF enabled	

Note: Always set the OVFIE bit before starting the operation whenever executing interrupt handling for overflows.

- Bit 0: Overflow Flag (OVF). Flag indicating an overflow has occurred.

Bit 0: OVF	Description	
0	No overflow has occurred	(Initial value)
1	Overflow has occurred	

10.2.4 Vector Number Setting Register DIV (VCRDIV)

Bit:	31	30	29	…	19	18	17	16
Bit name:	—	—	—	…	—	—	—	—
Initial value:	0	0	0	…	0	0	0	0
R/W:	R	R	R	…	R	R	R	R

Bit:	15	14	13	…	3	2	1	0
Bit name:				…				
Initial value:	—	—	—	…	—	—	—	—
R/W:	R/W	R/W	R/W	…	R/W	R/W	R/W	R/W

Vector number setting register DIV (VCRDIV) is a 32-bit read/write register, but is also 16-bit accessible. The destination vector number is set in VCRDIV when an interrupt occurs in the division unit due to an overflow or underflow. Values can be set in the 16 bits from bit 15 to bit 0, but only the last 7 bits (bits 6–0) are valid. Always set 0 for the 9 bits from bit 15 to bit 7. VCRDIV is not initialized by a power-on reset or manual reset, in standby mode, or during module standbys.

- Bits 31 to 7: Reserved. These bits always read 0. The write value should always be 0.
- Bits 6 to 0: Interrupt Vector Number. Sets the interrupt destination vector number. Only the 7 bits 6–0 are valid (as the vector number).

10.2.5 Dividend Register H (DVDNTH)

Bit:	31	30	39	…	3	2	1	0
Bit name:				…				
Initial value:	—	—	—	…	—	—	—	—
R/W:	R/W	R/W	R/W	…	R/W	R/W	R/W	R/W

Dividend register H (DVDNTH) is a 32-bit read/write register in which the upper 32 bits of the dividend used for 64 bit ÷ 32 bit division operations are written. When a division operation is executed, the value set as the dividend is lost and the remainder written here at the end of the operation. The initial value of DVDNTH is undefined, and its value is also undefined after a power-on reset or manual reset, in standby mode, and during in module standbys. When the DVDNT register is set with a dividend value, the previous DVDNTH value is lost and the MSB of the DVDNT register is extended to all bits in the DVDNTH register.

10.2.6 Dividend Register L (DVDNTL)

Bit:	31	30	39	…	3	2	1	0
Bit name:				…				
Initial value:	—	—	—	…	—	—	—	—
R/W:	R/W	R/W	R/W	…	R/W	R/W	R/W	R/W

Dividend register L (DVDNTL) is a 32-bit read/write register in which the lower 32 bits of the dividend used for 64-bit ÷ 32-bit division operations are written. When a value is set in this register, the 64-bit ÷ 32-bit division operation begins. The value written in the DVDNT register for 32-bit ÷ 32-bit division is also set in this register. When a 64-bit ÷ 32-bit division operation is executed, the value set as the dividend is lost and the quotient written here at the end of the operation. The contents of this register are undefined after a power-on reset or manual reset, in standby mode, and during module standbys.

10.3 Operation

10.3.1 64-Bit ÷ 32-Bit Operations

64-bit ÷ 32-bit operations work as follows:

1. The 32-bit divisor is set in the divisor register (DVSR).
2. The 64-bit dividend is set in dividend registers H and L (DVDNTH and DVDNTL). First set the value in DVDNTH. When a value is written to DVDNTL, the 64-bit ÷ 32-bit operation begins.
3. This unit finishes a single operation in 39 cycles (starting from the setting of the value in DVDNTL). When an overflow occurs, however, the operation ends in 6 cycles. See section 10.3.3, Handling of Overflows, for more information. Note that operation is signed.
4. After the operation, the 32-bit remainder is written to DVDNTH and the 32-bit quotient is written to DVDNTL.

10.3.2 32-Bit ÷ 32-Bit Operations

32-bit ÷ 32-bit operations work as follows:

1. The 32-bit divisor is set in the divisor register (DVSR).
2. The 32-bit dividend is set in dividend register L (DVDNT) for 32-bit division. When a value is written to DVDNT, the 32-bit ÷ 32-bit operation begins.
3. This unit finishes a single operation in 39 cycles (starting from the setting of the value in DVDNT). When an overflow occurs, however, the operation ends in 6 cycles. See section 10.3.3, Handling of Overflows, for more information. Note that the operation is signed.

4. After the operation, the 32-bit remainder is written to DVDNTH and the 32-bit quotient is written to DVDNT.

10.3.3 Handling of Overflows

When the results of operations exceed the ranges expressed as signed 32 bits (when, in division between two negative numbers, the quotient is the maximum value and a remainder (negative number) is generated) or when the divisor is 0, an overflow will result.

When an overflow occurs, the OVF bit is set and an overflow interrupt is generated if interrupt generation is enabled (the OVFIE bit in DVCR is 1). The operation will then end with the result after 6 cycles of operation stored in the DVDNTH and DVDNTL registers. If interrupt generation is disabled (the OVFIE bit is 0), the operation will end with the operation result at 6 cycles set in DVDNTH and the maximum value H'7FFFFFFF or minimum value H'80000000 set in DVDNTL. In the SH7604, the maximum value results when a positive quotient overflows; the minimum value results when a negative quotient overflows. The first three cycles of the 6 cycles executed when an overflow occurs are used for flag setting within the division unit and the next three for division.

10.4 Usage Notes

10.4.1 Access

All accesses to the division unit except DVCR and VCRDIV must be 32-bit reads or writes. Word accesses to registers other than DVCR and VCRDIV result in reading or writing of undefined values. In the division unit, a read instruction is extended for one cycle immediately after an instruction that writes to a register, even if the register is the same, to ensure that the value written is accurately set in the destination register in the division unit.

When a read or write instruction is issued while the division unit is operating, the read or write instruction is continuously extended until the operation ends. This means that instructions that do not access the division unit can be parallel-processed. When an instruction is executed that writes to any register of the division unit immediately following an instruction that writes to the division start-up registers (DVDNTL or DVDNT), the correct value may not be set in the start-up register. Specify an instruction other than one that writes to a division unit register for the instruction immediately following instruction that writes to a start-up register.

Because of the above restrictions, efficient processing can be achieved by executing instructions that do not access the division unit for 39 cycles after starting the operation, then issuing a read instruction after the 39th cycle.

10.4.2 Overflow Flag

When an overflow occurs, the overflow flag (OVF) is set and is not automatically reset. When OVF is set, the operation is not affected. When necessary, clear it before the operation. The states of registers when overflow occurs are shown in table 10.2.

Table 10.2 Overflow Processing

Register	Overflow Interrupt Enabled	Overflow Interrupt Disabled
DVSR	Holds the value written	Holds the value written
DVDNT	Holds the results of operations until overflow generation is detected*	The maximum value is set for overflow to the plus side, or the minimum value for overflow to the minus side
DVCR	The OVF bit is set	The OVF bit is set
VCRDIV	Holds the value written	Holds the value written
DVDNTH	Holds the results of operations until overflow generation is detected*	Holds the results of operations until overflow generation is detected *
DVDNTL	Holds the results of operations until overflow generation is detected*	The maximum value is set for overflow to the plus side, or the minimum value for overflow to the minus side

Note: In division processing, the intermediate operation result is written for cycles up to detection of overflow generation.

Section 11 16-Bit Free-Running Timer

11.1 Overview

The SH7604 has a single-channel, 16-bit free-running timer (FRT) on-chip. The FRT is based on a 16-bit free-running counter (FRC) and can output two types of independent waveforms. The FRT can also measure the width of input pulses and the cycle of external clocks.

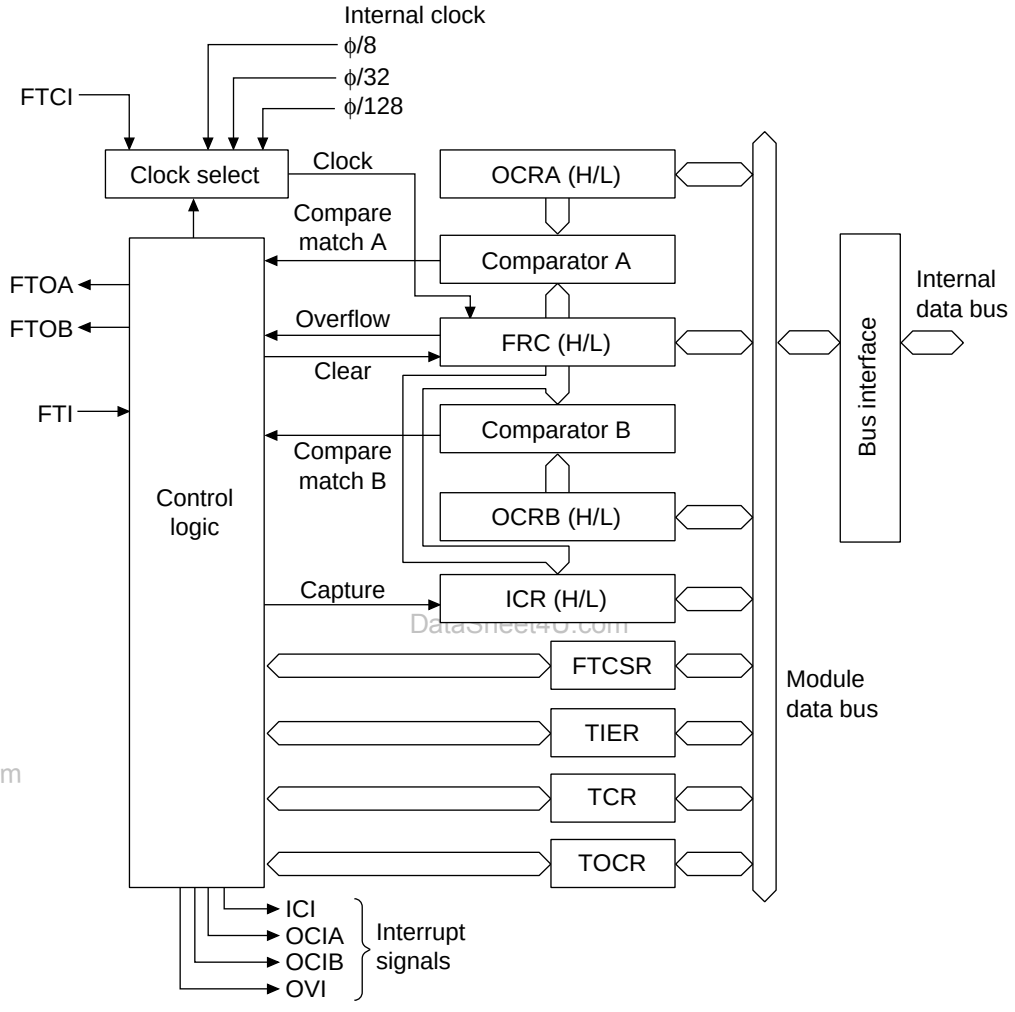
11.1.1 Features

The FRT has the following features:

- Allows selection between four types of counter input clocks. Select from external clock or three types of internal clocks (ϕ/8, ϕ/32, and ϕ/128). (External events can be counted.)
- Two independent comparators. Two types of waveforms can be output.
- Input capture. Select rising edge or falling edge.
- Counter clear can be specified. The counter value can be cleared upon compare match A.
- Four types of interrupt sources. Two compare matches, one input capture, and one overflow are available as interrupt sources, and interrupts can be requested independently for each.

11.1.2 Block Diagram

Figure 11.1 shows a block diagram of the FRT.



OCRA,B: Output compare registers A,B (16 bits)
FRC: Free-running counter (16 bits)
ICR: Input capture register (16 bits)
TCR: Timer control register (8 bits)
TIER: Timer interrupt enable register (8 bits)
FTCSR: Free-running timer control/status register (8 bits)
TOCR: Timer output compare control register (8 bits)

Figure 11.1 FRT Block Diagram

11.1.3 Pin Configuration

Table 11.1 lists FRT I/O pins and their functions.

Table 11.1 Pin Configuration

Channel	Pin	I/O	Function
Counter clock input pin	FTCI	I	FRC counter clock input pin
Output compare A output pin	FTOA	O	Output pin for output compare A
Output compare B output pin	FTOB	O	Output pin for output compare B
Input capture input pin	FTI	I	Input pin for input capture

11.1.4 Register Configuration

Table 11.2 shows the FRT register configuration.

Table 11.2 Register Configuration

Register	Abbreviation	R/W	Initial Value	Address
Timer interrupt enable register	TIER	R/W	H'01	HFFFFFE10
Free-running timer control/status register	FTCSR	R/(W)*1	H'00	HFFFFFE11
Free-running counter H	FRC H	R/W	H'00	HFFFFFE12
Free-running counter L	FRC L	R/W	H'00	HFFFFFE13
Output compare register A H	OCRA H	R/W	H'FF	HFFFFFE14*2
Output compare register A L	OCRA L	R/W	H'FF	HFFFFFE15*2
Output compare register B H	OCRB H	R/W	H'FF	HFFFFFE14*2
Output compare register B L	OCRB L	R/W	H'FF	HFFFFFE15*2
Timer control register	TCR	R/W	H'00	HFFFFFE16
Timer output compare control register	TOCR	R/W	H'E0	HFFFFFE17
Input capture register H	ICR H	R	H'00	HFFFFFE18
Input capture register L	ICR L	R	H'00	HFFFFFE19

Notes: 1. Bits 7 to 1 are read-only. The only value that can be written is a 0, which is used to clear flags. Bit 0 can be read or written.
2. OCRA and OCRB have the same address. The OCRS bit in TOCR is used to switch between them.
3. Use byte-size access for all registers.

11.2 Register Descriptions

11.2.1 Free-Running Counter (FRC)

Bit:	15	14	13	…	3	2	1	0
Bit name:				…				
Initial value:	0	0	0	…	0	0	0	0
R/W:	R/W	R/W	R/W	…	R/W	R/W	R/W	R/W

FRC is a 16-bit read/write up-counter. It increments upon input of a clock. The input clock can be selected using clock select bits 1 and 0 (CKS1, CKS0) in TCR. FRC can be cleared upon compare match A.

When FRC overflows (H'FFFF → H'0000), the overflow flag (OVF) in FTCSR is set to 1. FRC can be read or written to by the CPU, but because it is 16 bits long, data transfers involving the CPU are performed via a temporary register (TEMP). See section 11.3, CPU Interface, for more detailed information.

FRC is initialized to H'0000 by a reset, in standby mode, and when the module standby function is used.

11.2.2 Output Compare Registers A and B (OCRA and OCRB)

Bit:	15	14	13	…	3	2	1	0
Bit name:				…				
Initial value:	1	1	1	…	1	1	1	1
R/W:	R/W	R/W	R/W	…	R/W	R/W	R/W	R/W

OCR is composed of two 16-bit read/write registers (OCRA and OCRB). The contents of OCR are always compared to the FRC value. When the two values are the same, the output compare flags in FTCSR (OCFA and OCFB) are set to 1.

When the OCR and FRC values are the same (compare match), the output level values set in the output level bits (OLVLA and OLVLB) are output to the output compare pins (FTOA and FTOB). After a reset, FTOA and FTOB output 0 until the first compare match occurs.

Because OCR is a 16-bit register, data transfers involving the CPU are performed via a temporary register (TEMP). See section 11.3, CPU Interface, for more detailed information.

OCR is initialized to H'FFFF by a reset, in standby mode, and when the module standby function is used.

11.2.3 Input Capture Register (ICR)

Bit:	15	14	13	…	3	2	1	0
Bit name:				…				
Initial value:	0	0	0	…	0	0	0	0
R/W:	R	R	R	…	R	R	R	R

ICR is a 16-bit read-only register. When a rising edge or falling edge of the input capture signal is detected, the current FRC value is transferred to ICR. At the same time, the input capture flag (ICF) in FTCSR is set to 1. The edge of the input signal can be selected using the input edge select bit (IEDGA) in TCR.

Because ICR is a 16-bit register, data transfers involving the CPU are performed via a temporary register (TEMP). See Section 11.3, CPU Interface, for more detailed information. To ensure that the input capture operation is reliably performed, set the pulse width of the input capture input signal to six system clocks (ϕ) or more.

ICR is initialized to H'0000 by a reset, in standby mode, and when the module standby function is used.

11.2.4 Timer Interrupt Enable Register (TIER)

Bit:	7	6	5	4	3	2	1	0
Bit name:	ICIE	—	—	—	OCIAE	OCIBE	OVIE	—
Initial value:	0	0	0	0	0	0	0	1
R/W:	R/W	—	—	—	R/W	R/W	R/W	—

TIER is an 8-bit read/write register that controls enabling of all interrupt requests. TIER is initialized to H'01 by a reset, in standby mode, and when the module standby function is used.

- Bit 7—Input Capture Interrupt Enable (ICIE): Selects enabling/disabling of the ICI interrupt request when the input capture flag (ICF) in FTCSR is set to 1.

Bit 7: ICIE	Description	
0	Interrupt request (ICI) caused by ICF disabled	(Initial value)
1	Interrupt request (ICI) caused by ICF enabled	

- Bits 6 to 4—Reserved: These bits always read 0. The write value should always be 0. Do not write 1.

- Bit 3—Output Compare Interrupt A Enable (OCIAE): Selects enabling/disabling of the OCIA interrupt request when the output compare flag A (OCFA) in FTCSR is set to 1.

Bit 3: OCIAE	Description	
0	Interrupt request (OCIA) caused by OCFA disabled	(Initial value)
1	Interrupt request (OCIA) caused by OCFA enabled	

- Bit 2—Output Compare Interrupt B Enable (OCIBE): Selects enabling/disabling of the OCIB interrupt request when the output compare flag B (OCFB) in FTCSR is set to 1.

Bit 2: OCIBE	Description	
0	Interrupt request (OCIB) caused by OCFB disabled	(Initial value)
1	Interrupt request (OCIB) caused by OCFB enabled	

- Bit 1—Timer Overflow Interrupt Enable (OVIE): Selects enabling/disabling of the OVI interrupt request when the overflow flag (OVF) in FTCSR is set to 1.

Bit 1: OVIE	Description	
0	Interrupt request (FOVI) caused by OVF disabled	(initial value)
1	Interrupt request (FOVI) caused by OVF enabled	

- Bit 0—Reserved: This bit always reads 1. The write value should always be 1.

11.2.5 Free-Running Timer Control/Status Register (FTCSR)

Bit:	7	6	5	4	3	2	1	0
Bit name:	ICF	—	—	—	OCFA	OCFB	OVF	CCLRA
Initial value:	0	0	0	0	0	0	0	0
R/W:	R/(W)*	—	—	—	R/(W)*	R/(W)*	R/(W)*	R/W

Note: For bits 7, and 3 to 1, the only value that can be written is 0 (to clear the flags).

FTCSR is an 8-bit register that selects counter clearing and controls interrupt request signals. FTCSR is initialized to H'00 by a reset, in standby mode, and when the module standby function is used. See section 11.4, Operation, for the timing.

- Bit 7—Input Capture Flag (ICF): Status flag that indicates that the FRC value has been sent to FICR by the input capture signal. This flag is cleared by software and set by hardware. It cannot be set by software.

Bit 7: ICF	Description
0	Clear conditions: When ICF is read while set to 1, and then 0 is written to it (Initial value)
1	Set conditions: When the FRC value is sent to ICR by the input capture signal

- Bits 6 to 4—Reserved: These bits always read 0. The write value should always be 0.
- Bit 3—Output Compare Flag A (OCFA): Status flag that indicates when the values of the FRC and OCRA match. This flag is cleared by software and set by hardware. It cannot be set by software.

Bit 3: OCFA	Description
0	Clear conditions: When OCFA is read while set to 1, and then 0 is written to it (Initial value)
1	Set conditions: When the FRC value becomes equal to OCRA

- Bit 2—Output Compare Flag B (OCFB): Status flag that indicates when the values of FRC and OCRB match. This flag is cleared by software and set by hardware. It cannot be set by software.

Bit 2: OCFB	Description
0	Clear conditions: When OCFB is read while set to 1, and then 0 is written to it (Initial value)
1	Set conditions: When the FRC value becomes equal to OCRB

- Bit 1—Timer Overflow Flag (OVF): Status flag that indicates when FRC overflows (from H'FFFF to H'0000). This flag is cleared by software and set by hardware. It cannot be set by software.

Bit 1: OVF	Description
0	Clear conditions: When OVF is read while set to 1, and then 0 is written to it (Initial value)
1	Set conditions: When the FRC value changes from H'FFFF to H'0000

- Bit 0—Counter Clear A (CCLRA): Selects whether or not to clear FRC on compare match A (signal indicating match of FRC and OCRA).

Bit 0: CCLRA	Description	
0	FRC clear disabled	(Initial value)
1	FRC cleared on compare match A	

11.2.6 Timer Control Register (TCR)

Bit:	7	6	5	4	3	2	1	0
Bit name:	IEDGA	—	—	—	—	—	CKS1	CKS0
Initial value:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

TCR is an 8-bit read/write register that selects the input edge for input capture and selects the input clock for FRC. TCR is initialized to H'00 by a reset, in standby mode, and when the module standby function is used.

- Bit 7—Input Edge Select (IEDG): Selects whether to capture the input capture input (FTI) on the falling edge or rising edge.

Bit 7: IEDG	Description	
0	Input captured on falling edge	(Initial value)
1	Input captured on rising edge	

- Bits 6 to 2—Reserved: These bits always read 0. The write value should always be 0. Do not write 1.

- Bits 1 and 0—Clock Select (CKS1, CKS0): These bits select whether to use an external clock or one of three internal clocks for input to FRC. The external clock is counted at the rising edge.

Bit 1: CKS1	Bit 0: CKS0	Description	
0	0	Internal clock: count at ϕ/8	(Initial value)
	1	Internal clock: count at ϕ/32	
1	0	Internal clock: count at ϕ/128	
	1	External clock: count at rising edge	

11.2.7 Timer Output Compare Control Register (TOCR)

Bit:	7	6	5	4	3	2	1	0
Bit name:	—	—	—	OCRS	—	—	OLVLA	OLVLB
Initial value:	1	1	1	0	0	0	0	0
R/W:	—	—	—	R/W	R/W	R/W	R/W	R/W

TOCR is an 8-bit read/write register that selects the output level for output compare, enables output compare output, and controls switching between access of output compare registers A and B. TOCR is initialized to H'E0 by a reset, in standby mode, and when the module standby function is used.

Bits 7 to 5—Reserved: These bits always read 1. The write value should always be 1. Do not write 0.

Bit 4—Output Compare Register Select (OCRS): OCRA and OCRB share the same address. The OCRS bit controls which register is selected when reading/writing to this address. It does not affect the operation of OCRA and OCRB.

Bit 4: OCRS	Description	
0	OCRA register selected	(Initial value)
1	OCRB register selected	

Bits 3 and 2—Reserved: These bits always read 0. The write value should always be 0. Do not write 1.

Bit 1—Output Level A (OLVLA): Selects the level output to the output compare A output pin upon compare match A (signal indicating match of FRC and OCRA).

Bit 1: OLVLA	Description	
0	0 output on compare match A	(Initial value)
1	1 output on compare match A	

Bit 0—Output Level B (OLVLB): Selects the level output to the output compare B output pin upon compare match B (signal indicating match of FRC and OCRB).

Bit 0: OLVLB	Description	
0	0 output on compare match B	(Initial value)
1	1 output on compare match B	

11.3 CPU Interface

FRC, OCRA, OCRB, and FICR are 16-bit registers. The data bus width between the CPU and FRT, however, is only 8 bits. Access of these three types of registers from the CPU therefore needs to be performed via an 8-bit temporary register called TEMP.

The following describes how these registers are read from and written to:

- Writing to 16-bit Registers

  The upper byte is written, which results in the upper byte of data being stored in TEMP. The lower byte is then written, which results in 16 bits of data being written to the register when combined with the upper byte value in TEMP.

- Reading from 16-bit Registers

  The upper byte of data is read, which results in the upper byte value being transferred to the CPU. The lower byte value is transferred to TEMP. The lower byte is then read, which results in the lower byte value in TEMP being sent to the CPU.

When registers of these three types are accessed, two byte accesses should always be performed, first to the upper byte, then the lower byte. The same applies to accesses with the on-chip direct memory access controller. If only the upper byte or lower byte is accessed, the data will not be transferred properly.

Figure 11.2 and 11.3 show the flow of data when FRC is accessed. Other registers function in the same way. When reading OCRA and OCRB, however, both upper and lower-byte data is transferred directly to the CPU without passing through TEMP.

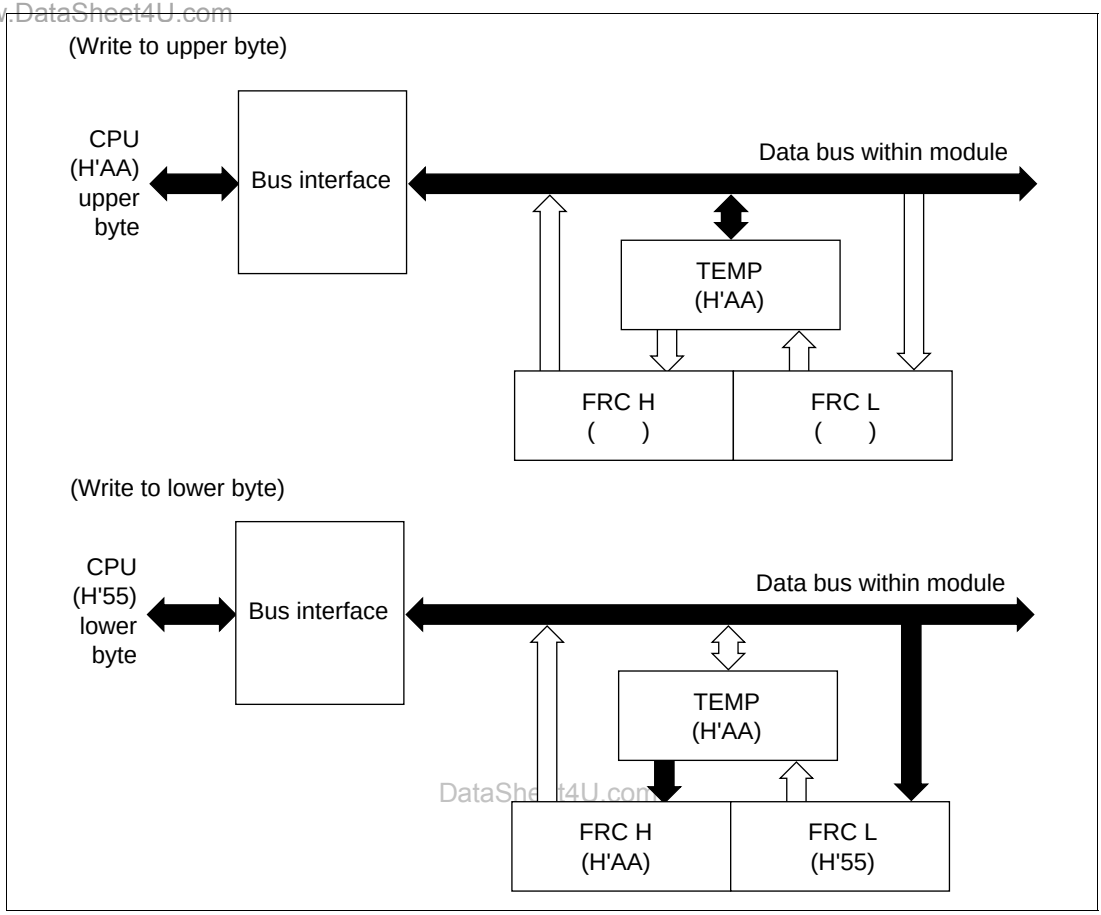


Figure 11.2 FRC Access Operation (CPU Writes H'AA55 to FRC)

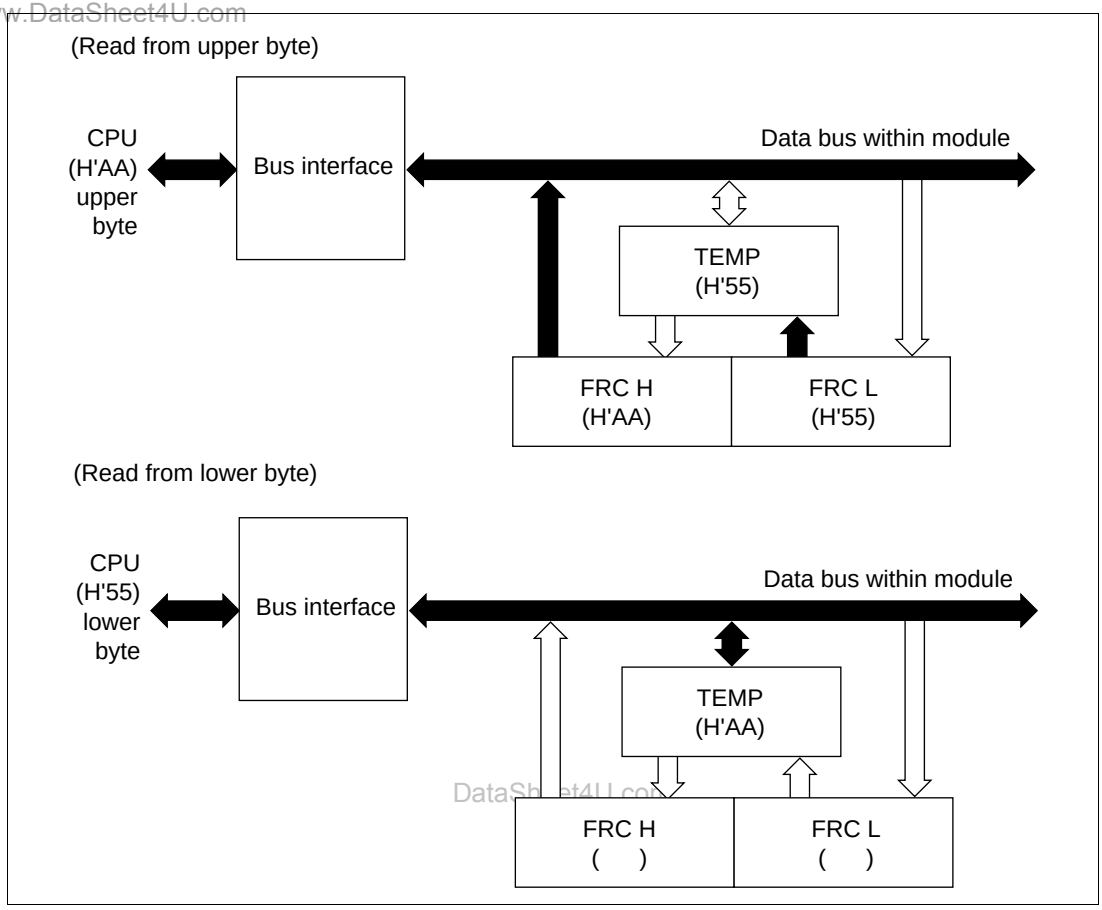


Figure 11.3 FRC Access Operation (CPU Reads H'AA55 from FRC)

11.4 Operation

11.4.1 FRC Count Timing

The FRC increments on clock input (internal or external).

Internal Clock Operation: Set the CKS1 and CKS0 bits in TCR to select which of the three internal clocks created by dividing system clock ϕ (ϕ/8, ϕ/32, ϕ/128) is used. Figure 11.4 shows the timing.

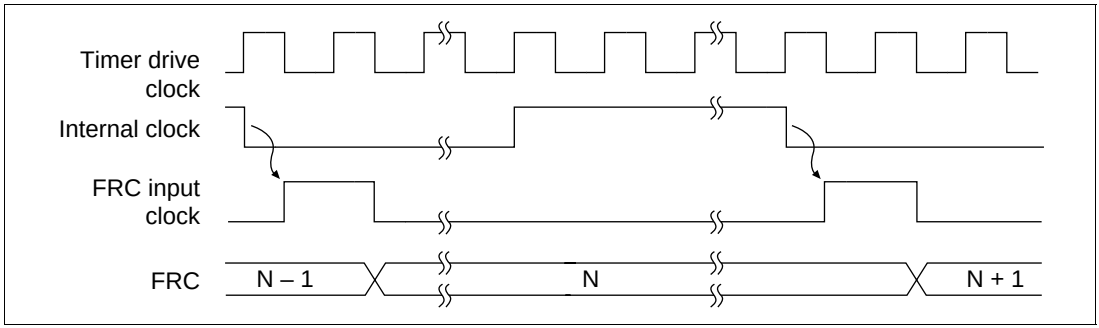


Figure 11.4 Count Timing (Internal Clock Operation)

External Clock Operation: Set the CKS1 and CKS0 bits in TCR to select the external clock. External clock pulses are counted on the rising edge. The pulse width of the external clock must be at least 6 system clocks (ϕ). A smaller pulse width will result in inaccurate operation. Figures 11.5 shows the timing.

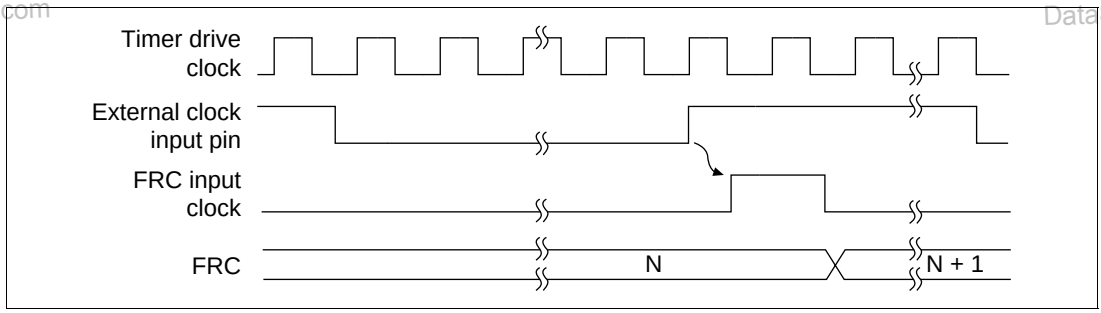


Figure 11.5 Count Timing (External Clock Operation)

11.4.2 Output Timing for Output Compare

When a compare match occurs, the output level set in the OLVL bit in TOCR is output from the output compare output pins (FTOA, FTOB). Figure 11.6 shows the timing for output of output compare A.

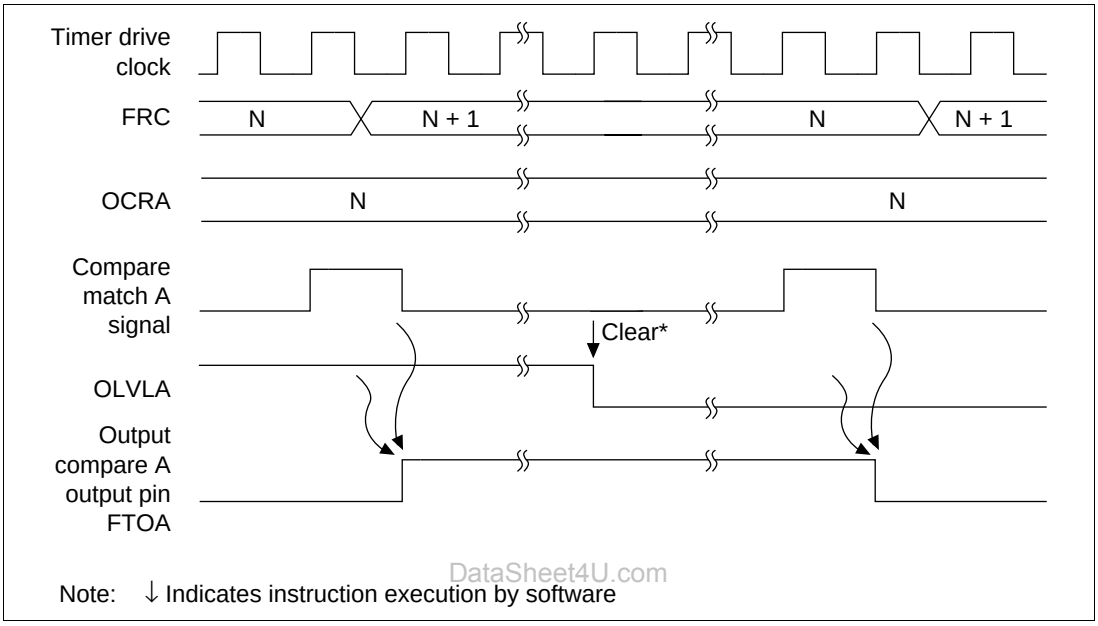


Figure 11.6 Output Timing for Output Compare A

11.4.3 FRC Clear Timing

FRC can be cleared on compare match A. Figure 11.7 shows the timing.

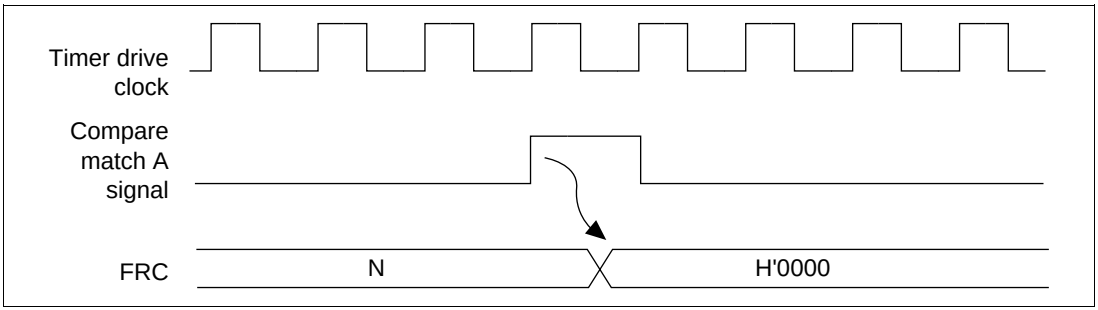


Figure 11.7 Compare Match A Clear Timing

11.4.4 Input Capture Input Timing

Either the rising edge or falling edge, can be selected for input capture input using the IEDG bit in TCR. Figure 11.8 shows the timing when the rising edge is selected (IEDG = 1).

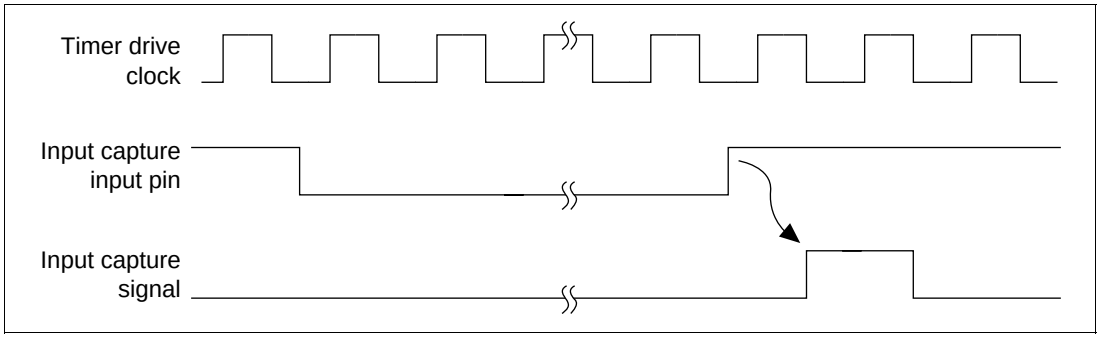


Figure 11.8 Input Capture Signal Timing (Normal)

When the input capture signal is input when ICR is read (upper-byte read), the input capture signal is delayed by one cycle of the clock that drives the timer. Figure 11.9 shows the timing.

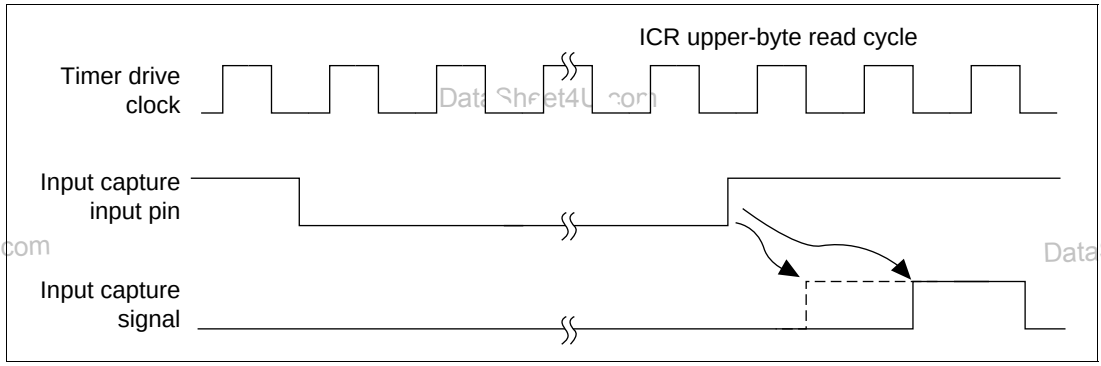


Figure 11.9 Input Capture Signal Timing (Input Capture Input when ICR is Read)

11.4.5 Input Capture Flag (ICF) Setting Timing

Input capture input sets the input capture flag (ICF) to 1 and simultaneously transfers the FRC value to ICR. Figure 11.10 shows the timing.

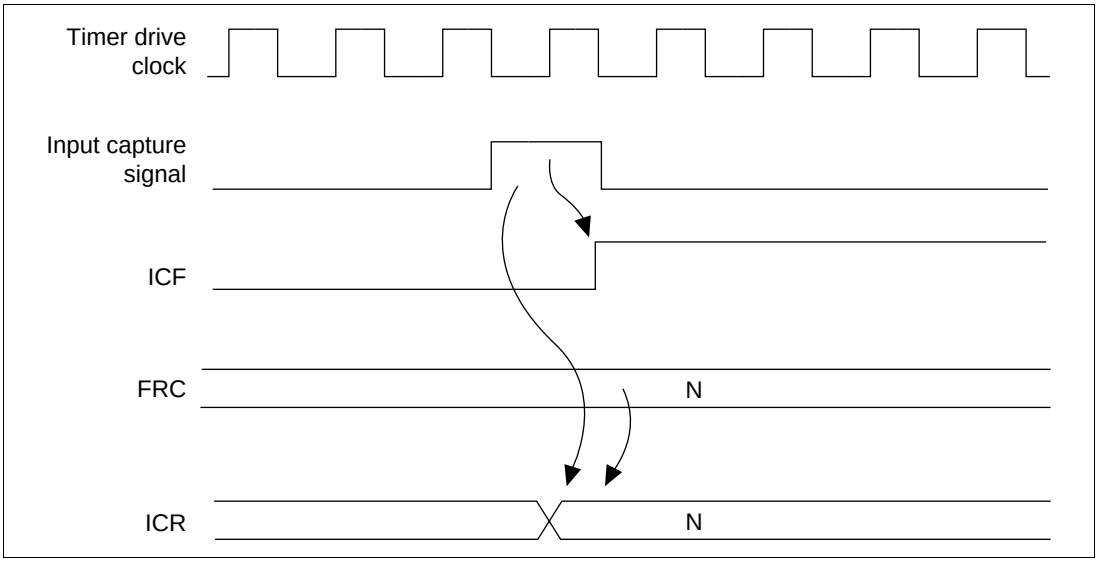


Figure 11.10 ICF Setting Timing

11.4.6 Output Compare Flag (OCFA, OCFB) Setting Timing

The compare match signal output (when OCRA or OCRB matches the FRC value) sets output compare flag OCFA or OCFB to 1. The compare match signal is generated in the last state in which the values matched (at the timing for updating the count value that matched the FRC). After OCRA or OCRB matches the FRC, no compare match signal is generated until the increment lock is generated. Figure 11.11 shows the timing for setting OCFA and OCFB.

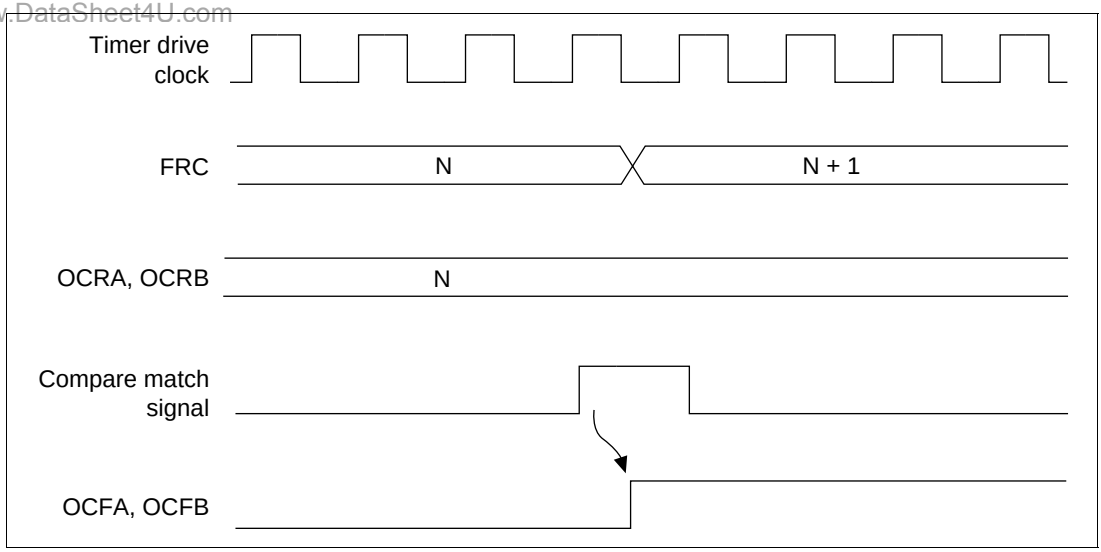


Figure 11.11 OCF Setting Timing

11.4.7 Timer Overflow Flag (OVF) Setting Timing

FRC overflow (from H'FFFF to H'0000) sets the timer overflow flag (OVF) to 1. Figure 11.12 shows the timing.

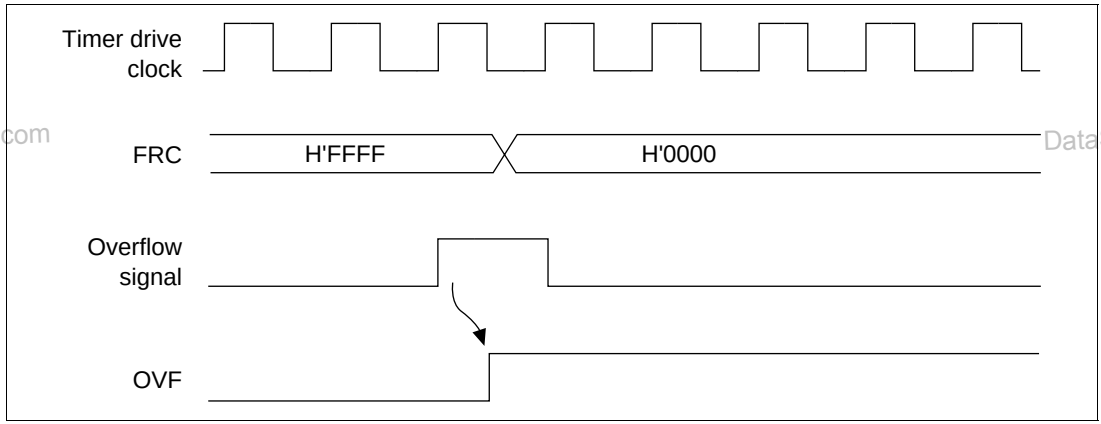


Figure 11.12 OVF Setting Timing

11.5 Interrupt Sources

There are four FRT interrupt sources of three types (ICI, OCIA/OCIB, and OVI). Table 11.3 lists the interrupt sources and their priorities after a reset is cleared. The interrupt enable bits in TIER are used to enable or disable the interrupt bits. Each interrupt request is sent to the interrupt controller independently. See section 4, Exception Handling, for more information about priorities and the relationship to interrupts other than those of the FRT.

Table 11.3 FRT Interrupt Sources and Priorities

Interrupt Source	Description	Priority
ICI	Interrupt by ICF	High
OCIA, OCIB	Interrupt by OCFA or OCFB	↑
OVI	Interrupt by OVF	Low

11.6 Example of FRT Use

Figure 11.13 shows an example in which pulses with a 50% duty factor and arbitrary phase relationship are output. The procedure is as follows:

1. Set the CCLRA bit in FTCSR to 1.
2. The OLVLA and OLVLB bits are inverted by software whenever a compare match occurs.

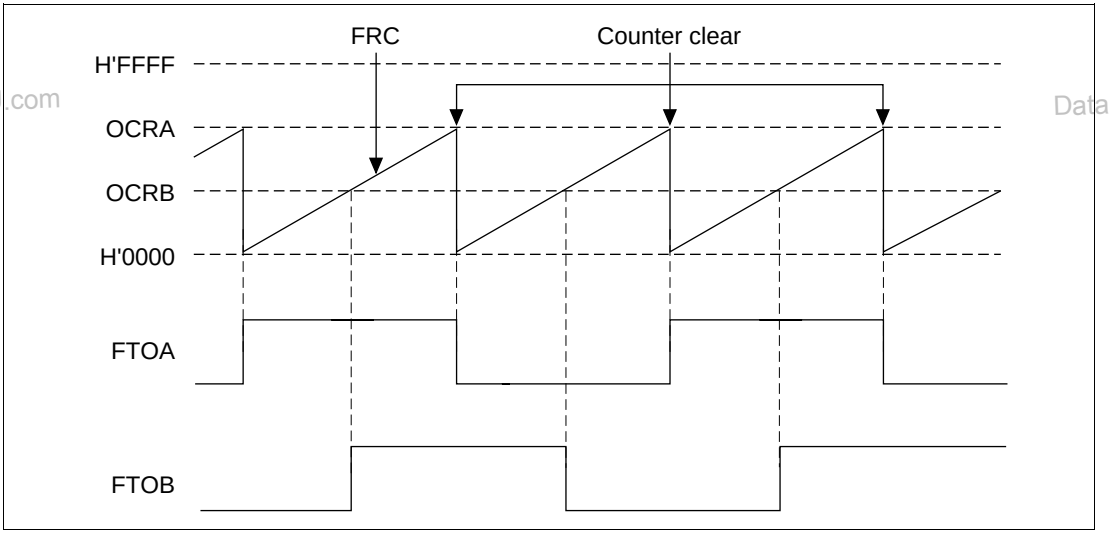


Figure 11.13 Example of Pulse Output

11.7 Usage Notes

Note that the following contention and operations occur when the FRT is operating:

1. FRC operates on the timer drive clock (ϕ/4), which has a cycle of 4 times the system clock (ϕ). For this reason, when the CPU performs an access, both the CPU and FRT will be operating, so a WAIT request will be generated from the FRT to the CPU. The number of access cycles thus varies by between 3 and 12 cycles.

2. Contention between FRC Write and Clear

   When a counter clear signal is generated with the timing shown in figure 11.14 during the write cycle for the lower byte of FRC, writing does not occur to the FRC, and the FRC clear takes priority.

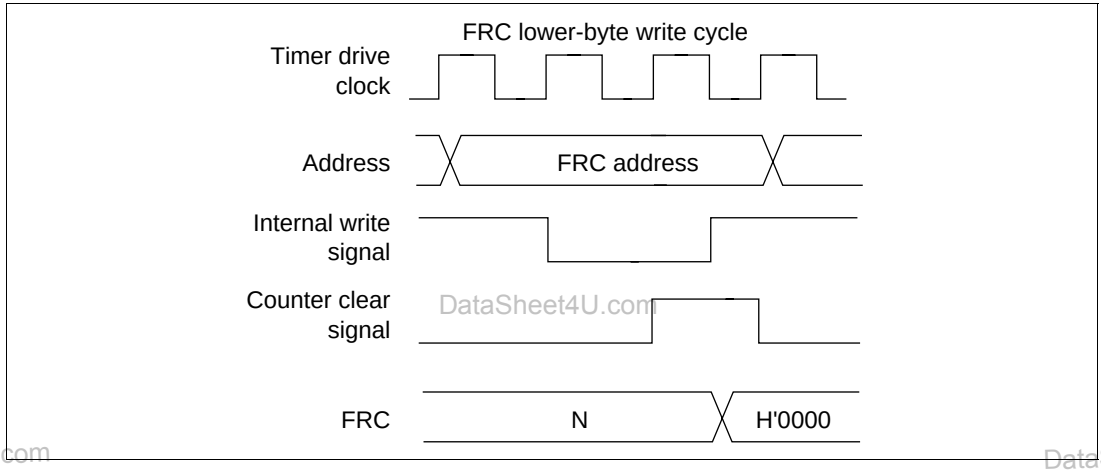


Figure 11.14 Contention between FRC Write and Clear

3. Contention between FRC Write and Increment

   When an increment occurs with the timing shown in figure 11.15 during the write cycle for the lower byte of FRC, no increment is performed and the counter write takes priority.

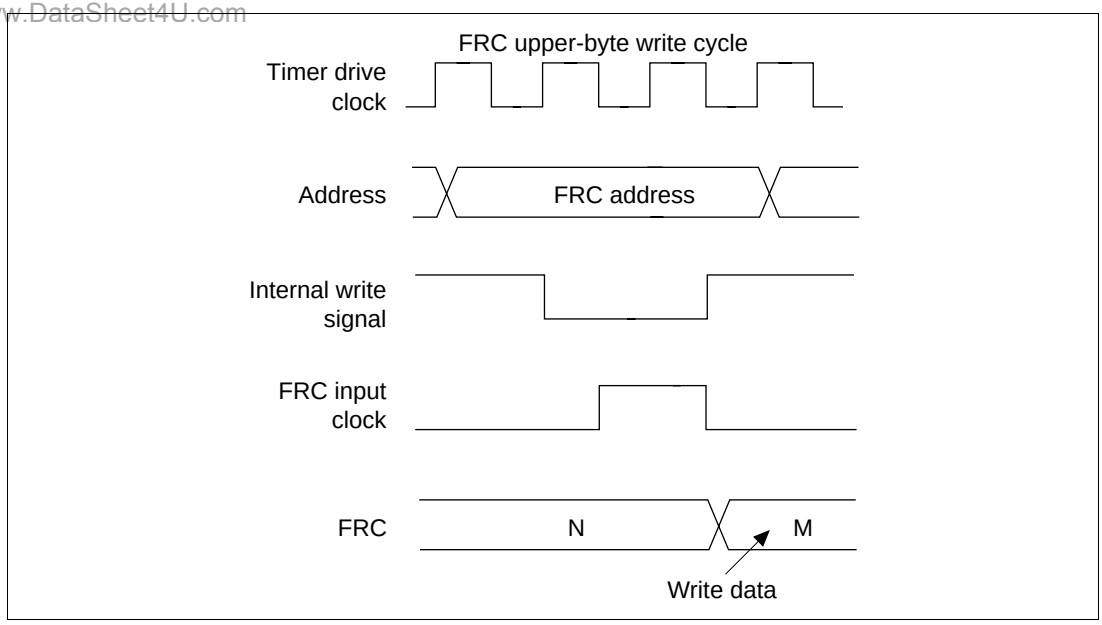


Figure 11.15 Contention between FRC Write and Increment

4. Contention between OCR Write and Compare Match

   When a compare match occurs with the timing shown in figure 11.16, during the write cycle for the lower byte of OCRA or OCRB, the OCR write takes priority and the compare match signal is disabled.

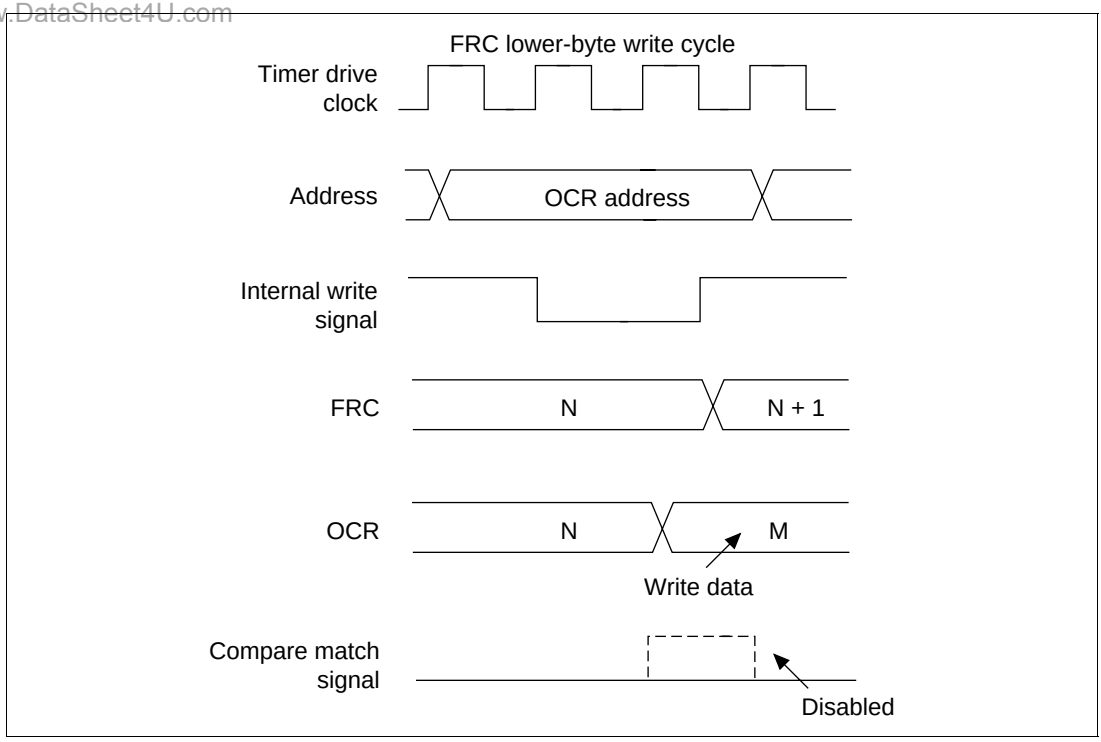


Figure 11.16 Contention between OCR and Compare Match

5. Internal Clock Switching and Counter Operation

   FRC will sometimes begin incrementing because of the timing of switching between internal clocks. Table 11.4 shows the relationship between internal clock switching timing (CKS1 and CKS0 bit rewrites) and FRC operation.

   When an internal clock is used, the FRC clock is generated when the falling edge of an internal clock (created by dividing the system clock (ϕ)) is detected. When a clock is switched to high before the switching and to low after switching, as shown in case 3 in table 11.4, the switchover is considered a falling edge and an FRC clock pulse is generated, causing FRC to increment. FRC may also increment when switching between an internal clock and an external clock.

Table 11.4 Internal Clock Switching and FRC Operation

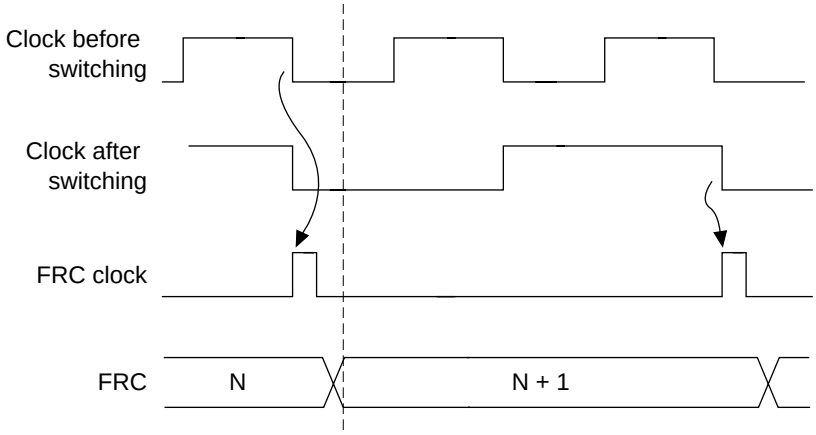
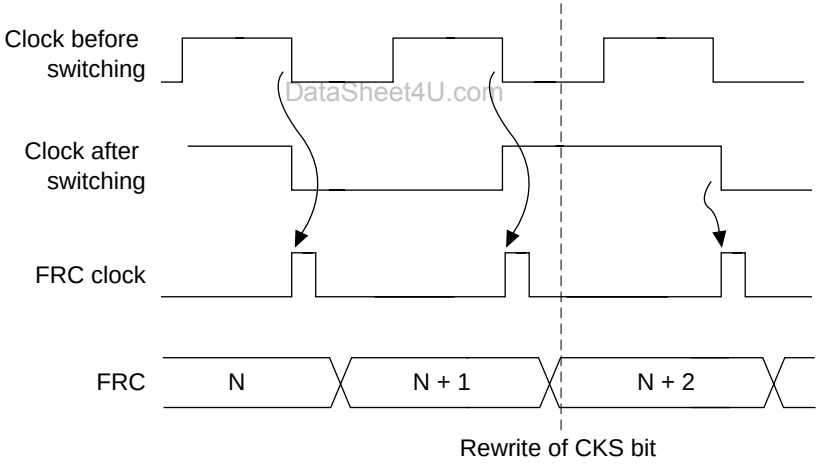
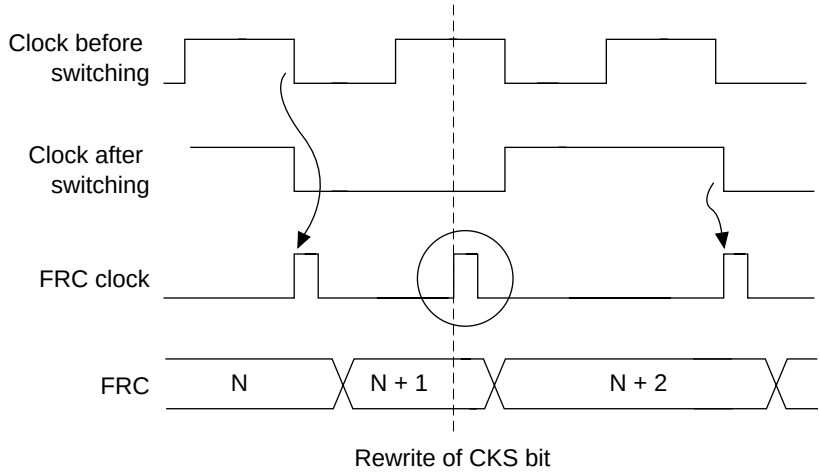
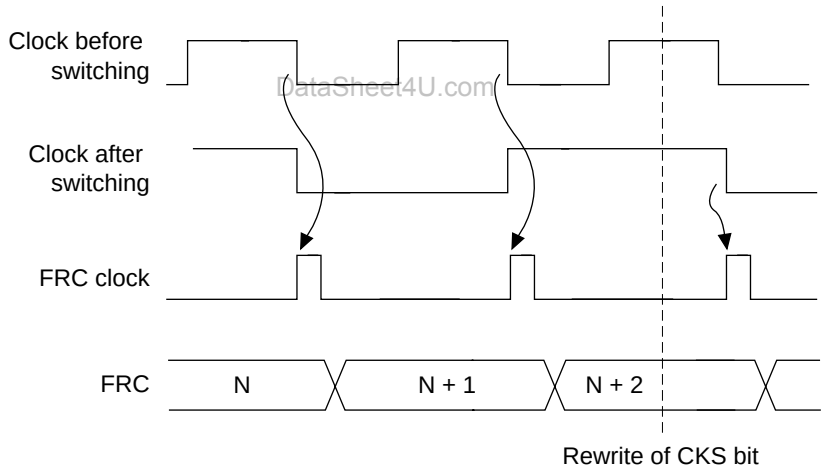
No.	Timing of Rewrite of CKS1 and CKS0 Bits	FRC Operation
1	Low-to-low switch	
2	Low-to-high switch	

Table 11.4 Internal Clock Switching and FRC Operation (cont)

No.	Timing of Rewrite of CKS1 and CKS0 Bits	FRC Operation
3	High-to-low switch	
4	High-to-high switch	

Note: Because the switchover is considered a falling edge, FRC starts counting up.

6. Timer Output (FTOA, FTOB)

During a power-on reset, the timer outputs (FTOA, FTOB) will be unreliable until the oscillation stabilizes. The initial value is output after the oscillation settling time has elapsed.

Section 12 Watchdog Timer (WDT)

12.1 Overview

The SH7604 has a single-channel watchdog timer (WDT) for monitoring system operations. If a system becomes uncontrolled and the timer counter overflows without being rewritten correctly by the CPU, an overflow signal ($\overline{\text{WDTOVF}}$) is output externally. The WDT can simultaneously generate an internal reset signal for the entire chip.

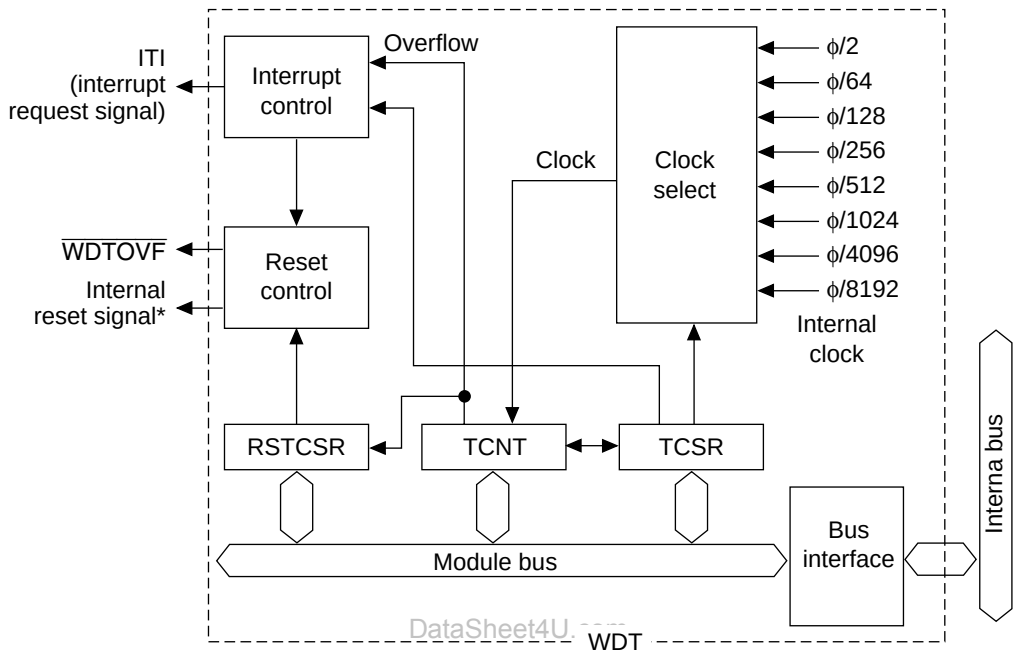
When this watchdog function is not needed, the WDT can be used as an interval timer. In the interval timer operation, an interval timer interrupt is generated at each counter overflow. The WDT is also used when recovering from standby mode, in modifying a clock frequency, and in clock pause mode.

12.1.1 Features

- Works in watchdog timer mode or interval timer mode.
- Outputs $\overline{\text{WDTOVF}}$ in watchdog timer mode. When the counter overflows in watchdog timer mode, overflow signal $\overline{\text{WDTOVF}}$ is output externally. It is possible to select whether to reset the chip internally when this happens. Either the power-on reset or manual reset signal can be selected as the internal reset signal.
- Generates interrupts in interval timer mode. When the counter overflows, it generates an interval timer interrupt.
- Used for standby mode clearing, clock frequency modification, and clock pause mode.
- Works with eight counter clock sources.

12.1.2 Block Diagram

Figure 12.1 shows a block diagram of the WDT.



WTCSR: Watchdog timer control/status register

WTCNT: Watchdog timer counter

RSTCSR: Reset control/status register

Note: The internal reset signal can be generated by a register setting. The type of reset can be selected (power-on or manual reset).

Figure 12.1 WDT Block Diagram

12.1.3 Pin Configuration

Table 12.1 shows the pin configuration.

Table 12.1 Pin Configuration

Pin	Abbreviation	I/O	Function
Watchdog timer overflow	$\overline{\text{WDTOVF}}$	O	Outputs the counter overflow signal in watchdog mode

12.1.4 Register Configuration

Table 12.2 summarizes the three WDT registers. They are used to select the clock, switch the WDT mode, and control the reset signal.

Table 12.2 WDT Registers

Name	Abbreviation	R/W	Initial Value	Address	
				Write*1	Read*2
Watchdog timer control/status register	WTCSR	R/(W)*3	H'18	H'FFFFFE80	H'FFFFFE80
Watchdog timer counter	WTCNT	R/W	H'00	H'FFFFFE80	H'FFFFFE81
Reset control/status register	RSTCSR	R/(W)*3	H'1F	H'FFFFFE82	H'FFFFFE83

Notes: 1. Write by word access. It cannot be written by byte or longword access.
2. Read by byte access. The correct value cannot be read by word or longword access.
3. Only 0 can be written in bit 7 to clear the flag.

12.2 Register Descriptions

12.2.1 Watchdog Timer Counter (WTCNT)

WTCNT is an 8-bit read/write up-counter. WTCNT differs from other registers in that it is more difficult to write. See section 12.2.4, Register Access, for details. When the timer enable bit (TME) in the watchdog timer control/status register (WTCSR) is set to 1, the watchdog timer counter starts counting pulses of an internal clock source selected by clock select bits 2 to 0 (CKS2 to CKS0) in WTCSR. When the value of WTCNT overflows (changes from H'FF to H'00), a watchdog timer overflow signal ($\overline{\text{WDTOVF}}$) or interval timer interrupt (ITI) is generated, depending on the mode selected in the WT/$\overline{\text{IT}}$ bit in WTCSR. WTCNT is initialized to H'00 by a reset and when the TME bit is cleared to 0. It is not initialized in standby mode.

Bit:	7	6	5	4	3	2	1	0
Bit name:								
Initial value:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

12.2.2 Watchdog Timer Control/Status Register (WTCSR)

The watchdog timer control/status register (WTCSR) is an 8-bit read/write register. WTCSR differs from other registers in being more difficult to write. See section 12.2.4, Register Access, for details. Its functions include selecting the timer mode and clock source. Bits 7 to 5 are initialized to 000 by a reset and in standby mode. Bits 2 to 0 are initialized to 000 by a reset, but retain their values in standby mode.

Bit:	7	6	5	4	3	2	1	0
Bit name:	OVF	WT/$\overline{IT}$	TME	—	—	CKS2	CKS1	CKS0
Initial value:	0	0	0	1	1	0	0	0
R/W:	R/(W)	R/W	R/W	—	—	R/W	R/W	R/W

- Bit 7—Overflow Flag (OVF): Indicates that WTCNT has overflowed from H'FF to H'00. It is not set in watchdog timer mode.

Bit 7: OVF	Description
0	No overflow of WTCNT in interval timer mode (Initial value) Cleared by reading OVF, then writing 0 in OVF
1	WTCNT overflow in interval timer mode

- Bit 6—Timer Mode Select (WT/$\overline{IT}$): Selects whether to use the WDT as a watchdog timer or interval timer. When WTCNT overflows, the WDT either generates an interval timer interrupt (ITI) or generates a $\overline{WDTOVF}$ signal, depending on the mode selected.

Bit 6: WT/$\overline{IT}$	Description
0	Interval timer mode: interval timer interrupt (ITI) request to the CPU when WTCNT overflows (Initial value)
1	Watchdog timer mode: $\overline{WDTOVF}$ signal output externally when WTCNT overflows. Section 12.2.3, Reset Control/Status Register (RSTCSR), describes in detail what happens when WTCNT overflows in watchdog timer mode.

- Bit 5—Timer Enable (TME): Enables or disables the timer.

Bit 5: TME	Description
0	Timer disabled: WTCNT is initialized to H'00 and count-up stops (Initial value)
1	Timer enabled: WTCNT starts counting. A $\overline{WDTOVF}$ signal or interrupt is generated when WTCNT overflows.

- Bits 4 and 3—Reserved: These bits always read 1. The write value shoul always be 1.

- Bits 2 to 0—Clock Select 2 to 0 (CKS2 to CKS0): These bits select one of eight internal clock sources for input to WTCNT. The clock signals are obtained by dividing the frequency of the system clock (ϕ).

			Description	
Bit 2: CKS2	**Bit 1: CKS1**	**Bit 0: CKS0**	**Clock Source**	**Overflow Interval* (ϕ = 28.7 MHz)**
0	0	0	ϕ/2 (Initial value)	17.8 μs
0	0	1	ϕ/64	570.8 μs
0	1	0	ϕ/128	1.1 ms
0	1	1	ϕ/256	2.2 ms
1	0	0	ϕ/512	4.5 ms
1	0	1	ϕ/1024	9.1 ms
1	1	0	ϕ/4096	36.5 ms
1	1	1	ϕ/8192	73.0 ms

Note: The overflow interval listed is the time from when the WTCNT begins counting at H'00 until an overflow occurs.

12.2.3 Reset Control/Status Register (RSTCSR)

RSTCSR is an eight-bit read/write register that controls output of the reset signal generated by watchdog timer counter (WTCNT) overflow and selects the internal reset signal type. RSTCSR differs from other registers in that it is more difficult to write. See section 12.2.4, Register Access, for details. RSTCR is initialized to H'1F by input of a reset signal from the $\overline{\text{RES}}$ pin, but is not initialized by the internal reset signal generated by overflow of the WDT. It is initialized to H'1F in standby mode.

Bit:	7	6	5	4	3	2	1	0
Bit name:	WOVF	RSTE	RSTS	—	—	—	—	—
Initial value:	0	0	0	1	1	1	1	1
R/W:	R/(W)*	R/W	R/W	—	—	—	—	—

Note: Only 0 can be written in bit 7 to clear the flag.

- Bit 7—Watchdog Timer Overflow Flag (WOVF): Indicates that WTCNT has overflowed (from H'FF to H'00) in watchdog timer mode. It is not set in interval timer mode.

Bit 7: WOVF	Description	
0	No WTCNT overflow in watchdog timer mode	(Initial value)
	Cleared by reading WOVF, then writing 0 in WOVF	
1	Set by WTCNT overflow in watchdog timer mode	

- Bit 6—Reset Enable (RSTE): Selects whether to reset the chip internally if WTCNT overflows in watchdog timer mode.

Bit 6: RSTE	Description	
0	Not reset when WTCNT overflows	(Initial value)
	LSI not reset internally, but WTCNT and WTCSR reset within WDT	
1	Reset when WTCNT overflows	

- Bit 5—Reset Select (RSTS): Selects the type of internal reset generated if WTCNT overflows in watchdog timer mode.

Bit 5: RSTS	Description	
0	Power-on reset	(Initial value)
1	Manual reset	

- Bits 4 to 0—Reserved: These bits always read as 1. The write value should always be 1.

12.2.4 Register Access

The watchdog timer's WTCNT, WTCSR, and RSTCSR registers differ from other registers in that they are more difficult to write. The procedures for writing and reading these registers are given below.

Writing to WTCNT and WTCSR: These registers must be written by a word transfer instruction. They cannot be written by byte or longword transfer instructions. WTCNT and WTCSR both have the same write address. The write data must be contained in the lower byte of the written word. The upper byte must be H'5A (for WTCNT) or H'A5 (for WTCSR) (figure 12.2). This transfers the write data from the lower byte to WTCNT or WTCSR.

Writing to WTCNT

		15 8	7 0
Address:	H'FFFFFE80	H'5A	Write data

Writing to WTCSR

		15 8	7 0
Address:	H'FFFFFE80	H'A5	Write data

Figure 12.2 Writing to WTCNT and WTCSR

Writing to RSTCSR: RSTCSR must be written by a word access to address H'FFFFFE82. It cannot be written by byte or longword transfer instructions. Procedures for writing 0 in WOVF (bit 7) and for writing to RSTE (bit 6) and RSTS (bit 5) are different, as shown in figure 12.3. To write 0 in the WOVF bit, the write data must be H'A5 in the upper byte and H'00 in the lower byte. This clears the WOVF bit to 0. The RSTE and RSTS bits are not affected. To write to the RSTE and RSTS bits, the upper byte must be H'5A and the lower byte must be the write data. The values of bits 6 and 5 of the lower byte are transferred to the RSTE and RSTS bits, respectively. The WOVF bit is not affected.

Writing 0 to the WOVF bit

		15 8	7 0
Address:	H'FFFFFE82	H'A5	H'00

Writing to the RSTE and RSTS bits

		15 8	7 0
Address:	H'FFFFFE82	H'5A	Write data

Figure 12.3 Writing to RSTCSR

Reading from WTCNT, WTCSR, and RSTCSR: WTCNT, WTCSR, and RSTCSR are read like other registers. Use byte transfer instructions. The read addresses are H'FFFFFE80 for WTCSR, H'FFFFFE81 for WTCNT, and H'FFFFFE83 for RSTCSR.

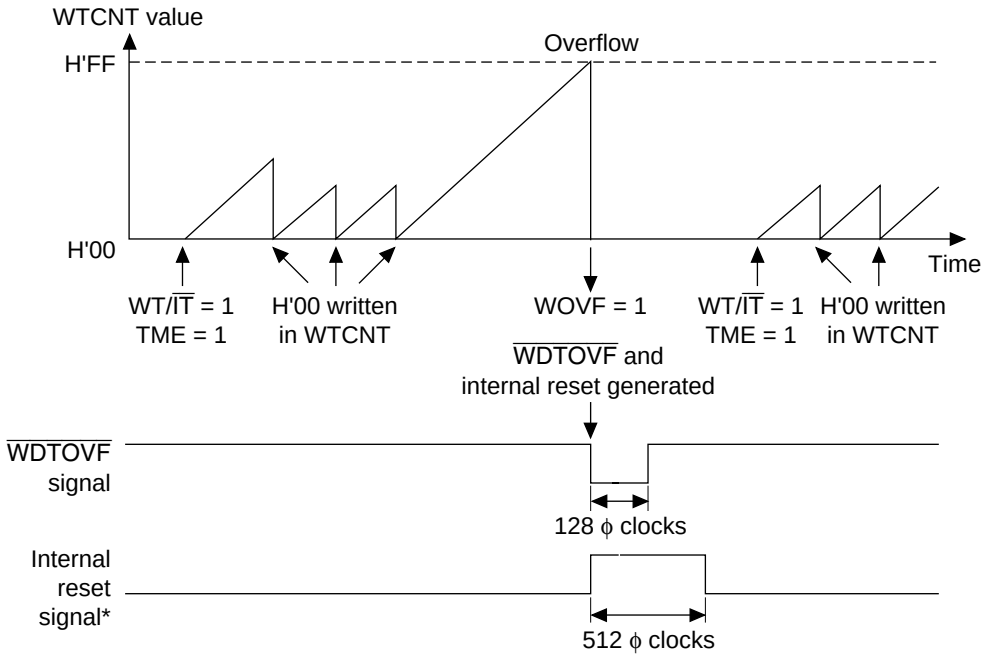
12.3 Operation

12.3.1 Operation in Watchdog Timer Mode

To use the WDT as a watchdog timer, set the WT/$\overline{IT}$ and TME bits in WTCSR to 1. Software must prevent WTCNT overflow by rewriting the WTCNT value (normally by writing H'00) before overflow occurs. If WTCNT fails to be rewritten and overflows occur due to a system crash or the like, a $\overline{WDTOVF}$ signal is output (figure 12.4). The $\overline{WDTOVF}$ signal can be used to reset the system. The $\overline{WDTOVF}$ signal is output for 128 ϕ clock cycles.

If the RSTE bit in RSTCSR is set to 1, a signal to reset the chip will be generated internally simultaneously with the $\overline{WDTOVF}$ signal when WTCNT overflows. Either a power-on reset or a manual reset can be selected by the RSTS bit. The internal reset signal is output for 512 ϕ clock cycles.

When a watchdog reset is generated simultaneously with input at the $\overline{RES}$ pin, the software distinguishes the $\overline{RES}$ reset from the watchdog reset by checking the WOVF bit in RSTCSR. The $\overline{RES}$ reset takes priority. The WOVF bit is cleared to 0.



WT/$\overline{IT}$: Timer mode select bit

TME: Timer enable bit

Note: Internal reset signal is generated only when the RSTE bit is set to 1.

Figure 12.4 Operation in Watchdog Timer Mode

12.3.2 Operation in Interval Timer Mode

To use the WDT as an interval timer, clear WT/$\overline{IT}$ to 0 and set TME to 1 in WTSCR. An interval timer interrupt (ITI) is generated each time the watchdog timer counter (WTCNT) overflows. This function can be used to generate interval timer interrupts at regular intervals (figure 12.5).

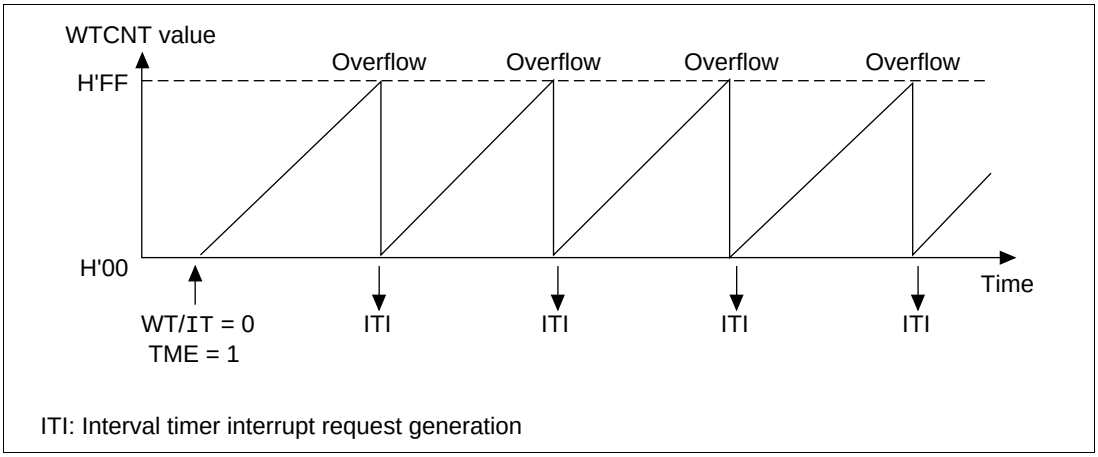


Figure 12.5 Operation in Interval Timer Mode

12.3.3 Operation in Standby Mode

The watchdog timer has a special function to clear standby mode with an NMI interrupt. When using standby mode, set the WDT as described below.

Transition to Standby Mode: The TME bit in WTCSR must be cleared to 0 to stop the watchdog timer counter before it enters standby mode. The chip cannot enter standby mode while the TME bit is set to 1. Set bits CKS2 to CKS0 in WTCSR so that the counter overflow interval is equal to or longer than the oscillation settling time. See section15.3, AC Characteristics, for the oscillation settling time.

Recovery from Standby Mode: When an NMI request signal is received in standby mode the clock oscillator starts running and the watchdog timer starts counting at the rate selected by bits CKS2 to CKS0 before standby mode was entered. When WTCNT overflows (changes from H'FF to H'00) the system clock (ϕ) is presumed to be stable and usable; clock signals are supplied to the entire chip and standby mode ends.

For details on standby mode, see section 14, Power Down Modes.

12.3.4 Timing of Overflow Flag (OVF) Setting

In interval timer mode, when WTCNT overflows, the OVF flag in WTCSR is set to 1 and an interval timer interrupt (ITI) is requested (figure 12.6).

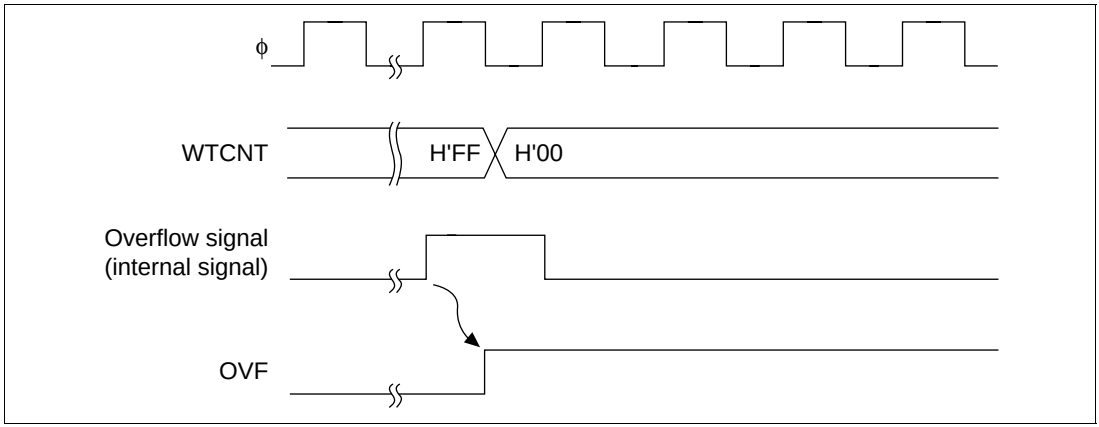


Figure 12.6 Timing of OVF Setting

12.3.5 Timing of Watchdog Timer Overflow Flag (WOVF) Setting

When WTCNT overflows the WOVF flag in RSTCSR is set to 1 and a $\overline{\text{WDTOVF}}$ signal is output. When the RSTE bit is set to 1, WTCNT overflow enables an internal reset signal to be generated for the entire chip (figure 12.7).

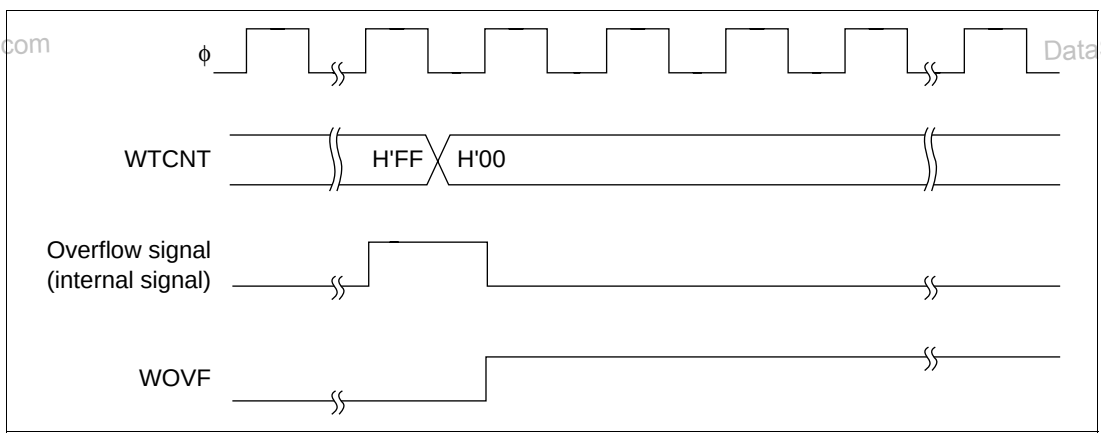


Figure 12.7 Timing of WOVF Setting

12.4 Usage Notes

12.4.1 Contention between WTCNT Write and Increment

If a timer counter clock pulse is generated during the T3 state of a write cycle to WTCNT, the write takes priority and the timer counter is not incremented (figure 12.8).

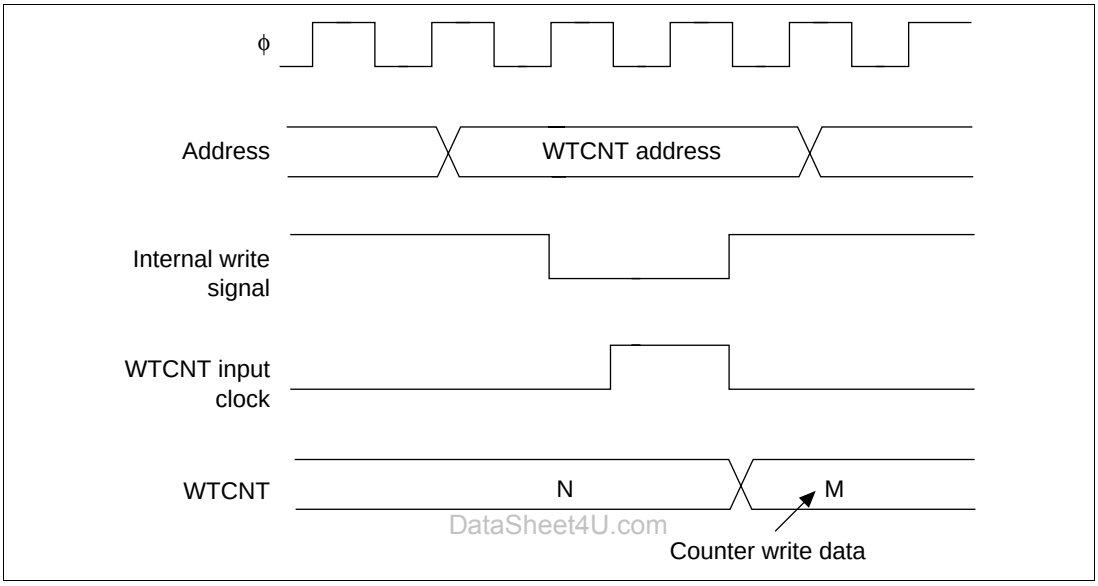


Figure 12.8 Contention between WTCNT Write and Increment

12.4.2 Changing CKS2 to CKS0 Bit Values

If the values of bits CKS2 to CKS0 are altered while the WDT is running, the count may increment incorrectly. Always stop the watchdog timer (by clearing the TME bit to 0) before changing the values of bits CKS2 to CKS0.

12.4.3 Switching between Watchdog Timer and Interval Timer Mode

To prevent incorrect operation, always stop the watchdog timer (by clearing the TME bit to 0) before switching between interval timer mode and watchdog timer mode.

12.4.4 System Reset with $\overline{\text{WDTOVF}}$

If a $\overline{\text{WDTOVF}}$ signal is input to the $\overline{\text{RES}}$ pin, the device cannot initialize correctly. Avoid logical input of the $\overline{\text{WDTOVF}}$ output signal to the $\overline{\text{RES}}$ input pin. To reset the entire system with the $\overline{\text{WDTOVF}}$ signal, use the circuit shown in figure 12.9.

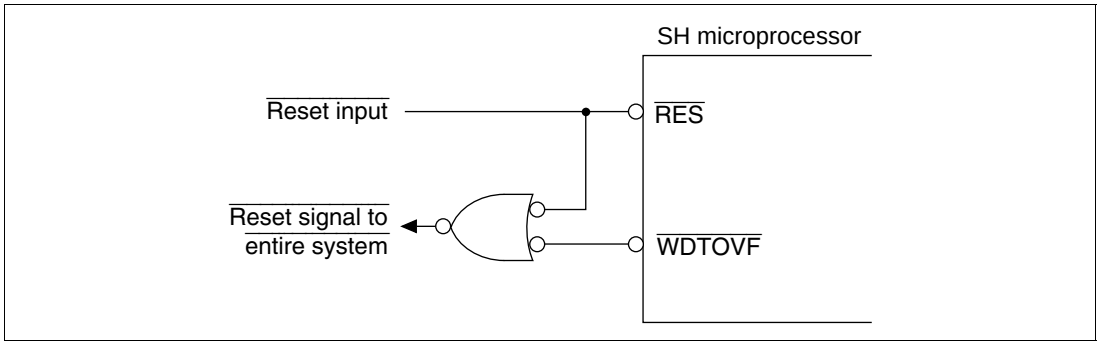


Figure 12.9 Example of Circuit for System Reset with $\overline{\text{WDTOVF}}$ Signal

12.4.5 Internal Reset in Watchdog Timer Mode

If the RSTE bit is cleared to 0 in watchdog timer mode, the chip will not reset internally when a WTCNT overflow occurs, but WTCNT and WTCSR in the WDT will reset.

Section 13 Serial Communication Interface

13.1 Overview

The SH7604 has a serial communication interface (SCI) that supports both asynchronous and clocked synchronous serial communication. It also has a multiprocessor communication function for serial communication among two or more processors.

13.1.1 Features

Selection of asynchronous or clock synchronous as the serial communication mode

- Asynchronous mode:
  - Serial data communication is synchronized by the start-stop method in character units. The SCI can communicate with a universal asynchronous receiver/transmitter (UART), an asynchronous communication interface adapter (ACIA), or any other chip that employs standard asynchronous serial communication. It can also communicate with two or more other processors using the multiprocessor communication function. There are twelve selectable serial data communication formats.
  - Data length: seven or eight bits
  - Stop bit length: one or two bits
  - Parity: even, odd, or none
  - Multiprocessor bit: one or none
  - Receive error detection: parity, overrun, and framing errors
- Clocked synchronous mode:
  - Serial data communication is synchronized with a clock signal. The SCI can communicate with other chips having a clocked synchronous communication function. There is one serial data communication format.
  - Data length: eight bits
  - Receive error detection: overrun errors
- Full duplex communication. The transmitting and receiving sections are independent, so the SCI can transmit and receive simultaneously. Both sections use double buffering, so continuous data transfer is possible in both the transmit and receive directions.
- Built-in baud rate generator with selectable bit rates
- Internal or external transmit/receive clock source. Baud rate generator (internal) or SCK pin (external)
- Four types of interrupts. Transmit-data-empty, transmit-end, receive-data-full, and receive-error interrupts are requested independently. The transmit-data-empty and receive-data-full interrupts can start the direct memory access controller (DMAC) to transfer data.

13.1.2 Block Diagram

Figure 13.1 shows a block diagram of the SCI.

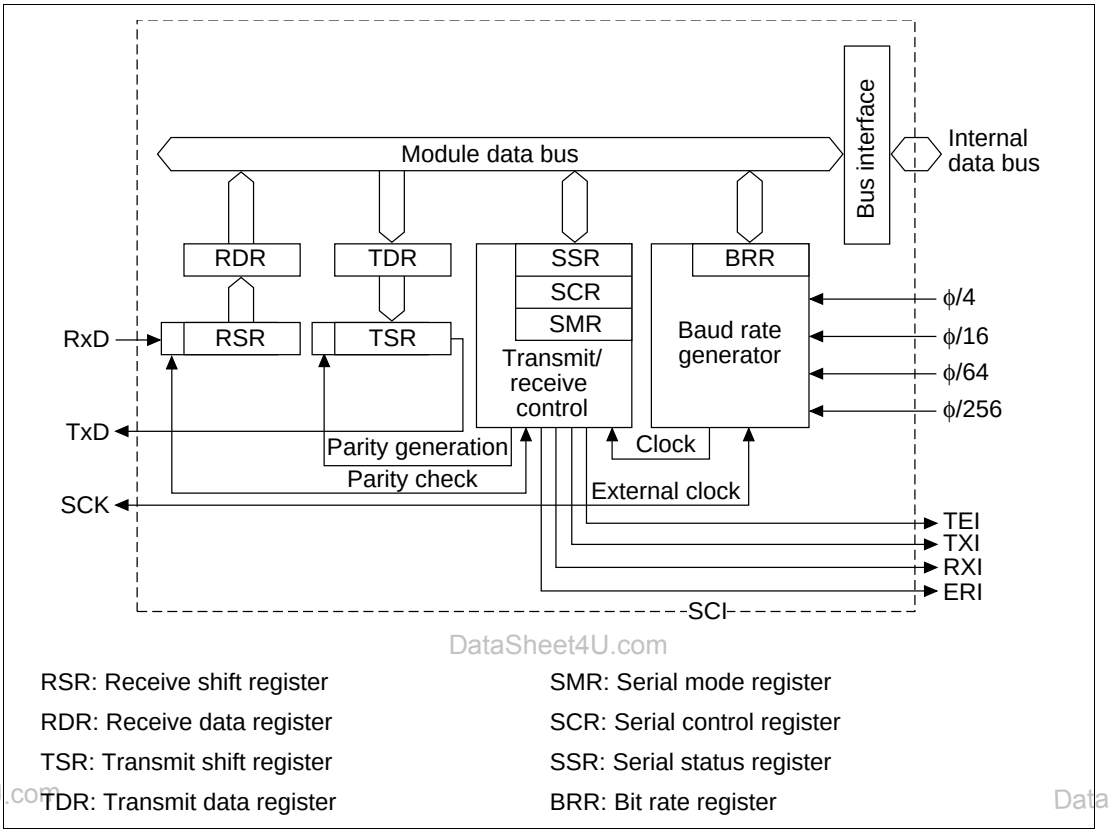


Figure 13.1 SCI Block Diagram

13.1.3 Pin Configuration

Table 13.1 summarizes the SCI pins.

Table 13.1 SCI Pins

Pin Name	Abbreviation	Input/Output	Function
Serial clock pin	SCK	Input/output	Clock input/output
Receive data pin	RxD	Input	Receive data input
Transmit data pin	TxD	Output	Transmit data output

13.1.4 Register Configuration

Table 13.2 summarizes the SCI internal registers. These registers select the communication mode (asynchronous or clock synchronous), specify the data format and bit rate, and control the transmitter and receiver sections.

Table 13.2 Registers

Name	Abbreviation	R/W	Initial Value	Address	Access size
Serial mode register	SMR	R/W	H'00	H'FFFFFE00	8
Bit rate register	BRR	R/W	H'FF	H'FFFFFE01	8
Serial control register	SCR	R/W	H'00	H'FFFFFE02	8
Transmit data register	TDR	R/W	H'FF	H'FFFFFE03	8
Serial status register	SSR	R/(W)*	H'84	H'FFFFFE04	8
Receive data register	RDR	R	H'00	H'FFFFFE05	8

Note: The only value that can be written is a 0 to clear the flags.

13.2 Register Descriptions

13.2.1 Receive Shift Register (RSR)

The receive shift register (RSR) receives serial data. Data input at the RxD pin is loaded into RSR in the order received, LSB (bit 0) first, converting the data to parallel form. When one byte has been received, it is automatically transferred to RDR. The CPU cannot read or write to RSR directly.

Bit:	7	6	5	4	3	2	1	0
Bit name:								
R/W:	—	—	—	—	—	—	—	—

13.2.2 Receive Data Register (RDR)

The receive data register (RDR) stores serial receive data. The SCI completes the reception of one byte of serial data by moving the received data from the receive shift register (RSR) into RDR for storage. RSR is then ready to receive the next data. This double buffering allows the SCI to receive data continuously.

The CPU can read but not write to RDR. RDR is initialized to H'00 by a reset and in standby and module standby mode.

Bit:	7	6	5	4	3	2	1	0
Bit name:								
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

13.2.3 Transmit Shift Register (TSR)

The transmit shift register (TSR) transmits serial data. The SCI loads transmit data from the transmit data register (TDR) into TSR, then transmits the data serially from the TxD pin, LSB (bit 0) first. After transmitting one data byte, the SCI automatically loads the next transmit data from TDR into TSR and starts transmitting again. If the TDRE bit in SSR is 1, however, the SCI does not load the TDR contents into TSR. The CPU cannot read or write to TSR directly.

Bit:	7	6	5	4	3	2	1	0
Bit name:								
R/W:	—	—	—	—	—	—	—	—

13.2.4 Transmit Data Register (TDR)

The transmit data register (TDR) is an 8-bit register that stores data for serial transmission. When the SCI detects that the transmit shift register (TSR) is empty, it moves transmit data written in TDR into TSR and starts serial transmission. Continuous serial transmission is possible by writing the next transmit data in TDR during serial transmission from TSR.

The CPU can always read and write to TDR. TDR is initialized to H'FF by a reset and in standby and module standby mode.

Bit:	7	6	5	4	3	2	1	0
Bit name:								
Initial value:	1	1	1	1	1	1	1	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

13.2.5 Serial Mode Register (SMR)

The serial mode register (SMR) is an 8-bit register that specifies the SCI serial communication format and selects the clock source for the baud rate generator.

The CPU can always read and write to SMR. SMR is initialized to H'00 by a reset and in standby and module standby mode.

Bit:	7	6	5	4	3	2	1	0
Bit name:	C/$\overline{A}$	CHR	PE	O/$\overline{E}$	STOP	MP	CKS1	CKS0
Initial value:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

- Bit 7—Communication Mode (C/$\overline{A}$): Selects whether the SCI operates in asynchronous or clocked synchronous mode.

Bit 7: C/A	Description	
0	Asynchronous mode	(Initial value)
1	Clocked synchronous mode	

- Bit 6—Character Length (CHR): Selects 7-bit or 8-bit data in asynchronous mode. In clocked synchronous mode, the data length is always eight bits, regardless of the CHR setting.

Bit 6: CHR	Description	
0	8-bit data	(Initial value)
1	7-bit data. (When 7-bit data is selected, the MSB (bit 7) of the transmit data register is not transmitted.)	

- Bit 5—Parity Enable (PE): Selects whether to add a parity bit to transmit data and to check the parity of receive data, in asynchronous mode. In clocked synchronous mode, a parity bit is neither added nor checked, regardless of the PE setting.

Bit 5: PE	Description	
0	Parity bit not added or checked	(Initial value)
1	Parity bit added and checked. When PE is set to 1, an even or odd parity bit is added to transmit data, depending on the parity mode (O/$\overline{E}$) setting. Receive data parity is checked according to the even/odd (O/$\overline{E}$) mode setting.	

- Bit 4—Parity Mode (O/$\overline{E}$): Selects even or odd parity when parity bits are added and checked. The O/$\overline{E}$ setting is used only in asynchronous mode and only when the parity enable bit (PE) is set to 1 to enable parity addition and checking. The O/$\overline{E}$ setting is ignored in clocked synchronous mode, and in asynchronous mode when parity addition and checking is disabled.

Bit 4: O/E	Description
0	Even parity (Initial value) If even parity is selected, the parity bit is added to transmit data to make an even number of 1s in the transmitted character and parity bit combined. Receive data is checked to see if it has an even number of 1s in the received character and parity bit combined.
1	Odd parity If odd parity is selected, the parity bit is added to transmit data to make an odd number of 1s in the transmitted character and parity bit combined. Receive data is checked to see if it has an odd number of 1s in the received character and parity bit combined.

- Bit 3—Stop Bit Length (STOP): Selects one or two bits as the stop bit length in asynchronous mode. This setting is used only in asynchronous mode. It is ignored in clocked synchronous mode because no stop bits are added.

- In receiving, only the first stop bit is checked, regardless of the STOP bit setting. If the second stop bit is 1, it is treated as a stop bit, but if the second stop bit is 0, it is treated as the start bit of the next incoming character.

Bit 3: STOP	Description
0	One stop bit (Initial value) In transmitting, a single 1-bit is added at the end of each transmitted character
1	Two stop bits In transmitting, two 1-bits are added at the end of each transmitted character

- Bit 2—Multiprocessor Mode (MP): Selects multiprocessor format. When multiprocessor format is selected, settings of the parity enable (PE) and parity mode (O/$\overline{E}$) bits are ignored. The MP bit setting is used only in asynchronous mode; it is ignored in clocked synchronous mode. For the multiprocessor communication function, see section 13.3.3, Multiprocessor Communication.

Bit 2: MP	Description
0	Multiprocessor function disabled (Initial value)
1	Multiprocessor format selected

- Bits 1 and 0—Clock Select 1 and 0 (CKS1 and CKS0): These bits select the internal clock source of the built-in baud rate generator. Four clock sources are available. $\phi/4$, $\phi/16$, $\phi/64$ and $\phi/256$. For further information on the clock source, bit rate register settings, and baud rate, see section 13.2.8, Bit Rate Register.

Bit 1: CKS1	Bit 0: CKS0	Description	
0	0	$\phi/4$	(Initial value)
	1	$\phi/16$	
1	0	$\phi/64$	
	1	$\phi/256$	

13.2.6 Serial Control Register (SCR)

The serial control register (SCR) operates the SCI transmitter/receiver, selects the serial clock output in asynchronous mode, enables/disables interrupts, and selects the transmit/receive clock source. The CPU can always read and write to SCR. SCR is initialized to H'00 by a reset and in standby and module standby modes.

Bit:	7	6	5	4	3	2	1	0
Bit name:	TIE	RIE	TE	RE	MPIE	TEIE	CKE1	CKE0
Initial value:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

- Bit 7—Transmit Interrupt Enable (TIE): Enables or disables the transmit-data-empty interrupt (TXI) requested when the transmit data register empty bit (TDRE) in the serial status register (SSR) is set to 1 due to transfer of serial transmit data from TDR to TSR.

Bit 7: TIE	Description	
0	Transmit-data-empty interrupt request (TXI) is disabled	(Initial value)
	The TXI interrupt request can be cleared by reading TDRE after it has been set to 1, then clearing TDRE to 0, or by clearing TIE to 0.	
1	Transmit-data-empty interrupt request (TXI) is enabled	

- Bit 6—Receive Interrupt Enable (RIE): Enables or disables the receive-data-full interrupt (RXI) requested when the receive data register full bit (RDRF) in the serial status register (SSR) is set to 1 due to transfer of serial receive data from RSR to RDR. It also enables or disables receive-error interrupt (ERI) requests.

Bit 6: RIE	Description
0	Receive-data-full interrupt (RXI) and receive-error interrupt (ERI) requests are disabled (Initial value) RXI and ERI interrupt requests can be cleared by reading the RDRF flag or error flag (FER, PER, or ORER) after it has been set to 1, then clearing the flag to 0, or by clearing RIE to 0.
1	Receive-data-full interrupt (RXI) and receive-error interrupt (ERI) requests are enabled

- Bit 5—Transmit Enable (TE): Enables or disables the SCI serial transmitter.

Bit 5: TE	Description
0	Transmitter disabled (Initial value) The transmit data register empty bit (TDRE) in the serial status register (SSR) is locked at 1
1	Transmitter enabled Serial transmission starts when the transmit data register empty (TDRE) bit in the serial status register (SSR) is cleared to 0 after writing of transmit data into TDR. Select the transmit format in SMR before setting TE to 1.

- Bit 4—Receive Enable (RE): Enables or disables the SCI serial receiver.

Bit 4: RE	Description
0	Receiver disabled (Initial value) Clearing RE to 0 does not affect the receive flags (RDRF, FER, PER, ORER). These flags retain their previous values.
1	Receiver enabled Serial reception starts when a start bit is detected in asynchronous mode, or synchronous clock input is detected in clocked synchronous mode. Select the receive format in SMR before setting RE to 1.

- Bit 3—Multiprocessor Interrupt Enable (MPIE): Enables or disables multiprocessor interrupts. The MPIE setting is used only in asynchronous mode, and only if the multiprocessor mode bit (MP) in the serial mode register (SMR) is set to 1 during reception. The MPIE setting is ignored in clocked synchronous mode or when the MP bit is cleared to 0.

Bit 3: MPIE	Description
0	Multiprocessor interrupts are disabled (normal receive operation) (Initial value) MPE is cleared to 0 when MPIE is cleared to 0, or the multiprocessor bit (MPB) is set to 1 in receive data.
1	Multiprocessor interrupts are enabled Receive-data-full interrupt requests (RXI), receive-error interrupt requests (ERI), and setting of the RDRF, FER, and ORER status flags in the serial status register (SSR) are disabled until the multiprocessor bit is set to 1. The SCI does not transfer receive data from RSR to RDR, does not detect receive errors, and does not set the RDRF, FER, and ORER flags in the serial status register (SSR). When it receives data that includes MPB = 1, MPB is set to 1 in SSR, and the SCI automatically clears MPIE to 0, generates RXI and ERI interrupts (if the TIE and RIE bits in SCR are set to 1), and enables the FER and ORER bits to be set.

- Bit 2—Transmit-End Iinterrupt Enable (TEIE): Enables or disables the transmit-end interrupt (TEI) requested if TDR does not contain new transmit data when the MSB is transmitted.

Bit 2: TEIE	Description
0	Transmit-end interrupt (TEI) requests are disabled* (Initial value)
1	Transmit-end interrupt (TEI) requests are enabled*

Note: The TEI request can be cleared by reading the TDRE bit in the serial status register (SSR) after it has been set to 1, then clearing TDRE to 0; by clearing the transmit end (TEND) bit to 0; or by clearing the TEIE bit to 0.

- Bits 1 and 0—Clock Enable 1 and 0 (CKE1 and CKE0): These bits select the SCI clock source and enable or disable clock output from the SCK pin. Depending on the combination of CKE1 and CKE0, the SCK pin can be used for general-purpose input/output, serial clock output, or serial clock input.

  The CKE0 setting is valid only in asynchronous mode, and only when the SCI is internally clocked (CKE1 = 0). The CKE0 setting is ignored in clocked synchronous mode, or when an external clock source is selected (CKE1 = 1). Select the SCI operating mode in the serial mode register (SMR) before setting CKE1 and CKE0. For further details on selection of the SCI clock source, see table 13.9 in section 13.3, Operation.

Bit 1: CKE1	Bit 0: CKE0	Description	
0	0	Asynchronous mode	Internal clock, SCK pin used for input pin (input signal is ignored or output pin output level is undefined)*1
		Clocked synchronous mode	Internal clock, SCK pin used for synchronous clock output*1
0	1	Asynchronous mode	Internal clock, SCK pin used for clock output*2
		Clocked synchronous mode	Internal clock, SCK pin used for synchronous clock output
1	0	Asynchronous mode	External clock, SCK pin used for clock input*3
		Clocked synchronous mode	External clock, SCK pin used for synchronous clock input
1	1	Asynchronous mode	External clock, SCK pin used for clock input*3
		Clocked synchronous mode	External clock, SCK pin used for synchronous clock input

Notes: 1. Initial value
2. The output clock frequency is the same as the bit rate.
3. The input clock frequency is 16 times the bit rate.

13.2.7 Serial Status Register (SSR)

The serial status register (SSR) is an 8-bit register containing multiprocessor bit values, and status flags that indicate the SCI operating status.

The CPU can always read and write to SSR, but cannot write 1 in the status flags (TDRE, RDRF, ORER, PER, and FER). These flags can be cleared to 0 only if they have first been read (after being set to 1). Bits 2 (TEND) and 1 (MPB) are read-only bits that cannot be written. SSR is initialized to H'84 by a reset and in standby and module standby mode.

Bit:	7	6	5	4	3	2	1	0
Bit name:	TDRE	RDRF	ORER	FER	PER	TEND	MPB	MPBT
Initial value:	1	0	0	0	0	1	0	0
R/W:	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R	R	R/W

Note: The only value that can be written is a 0 to clear the flag.

- Bit 7—Transmit Data Register Empty (TDRE): Indicates that the SCI has loaded transmit data from TDR into TSR and new serial transmit data can be written in TDR.

Bit 7: TDRE	Description
0	TDR contains valid transmit data TDRE is cleared to 0 when software reads TDRE after it has been set to 1, then writes 0 in TDRE, or the DMAC writes data in TDR.
1	TDR does not contain valid transmit data (Initial value) TDRE is set to 1 when the chip is reset or enters standby mode, the TE bit in the serial control register (SCR) is cleared to 0, or TDR contents are loaded into TSR, so new data can be written in TDR.

- Bit 6—Receive Data Register Full (RDRF): Indicates that RDR contains received data.

Bit 6: RDRF	Description
0	RDR does not contain valid receive data (Initial value) RDRF is cleared to 0 when the chip is reset or enters standby mode, software reads RDRF after it has been set to 1, then writes 0 in RDRF, or the DMAC reads data from RDR.
1	RDR contains valid received data RDRF is set to 1 when serial data is received normally and transferred from RSR to RDR.

Note: RDR and RDRF are not affected by detection of receive errors or by clearing of the RE bit to 0 in the serial control register. They retain their previous contents. If RDRF is still set to 1 when reception of the next data ends, an overrun error (ORER) occurs and the received data is lost.

- Bit 5—Overrun Error (ORER): Indicates that data reception ended abnormally due to an overrun error.

Bit 5: ORER	Description
0	Receiving is in progress or has ended normally[*1] (Initial value) ORER is cleared to 0 when the chip is reset or enters standby mode, or software reads ORER after it has been set to 1, then writes 0 in ORER.
1	A receive overrun error occurred[*2] ORER is set to 1 if reception of the next serial data ends when RDRF is set to 1.

Notes: 1. Clearing the RE bit to 0 in the serial control register does not affect the ORER bit, which retains its previous value.

2. RDR continues to hold the data received before the overrun error, so subsequent receive data is lost. Serial receiving cannot continue while ORER is set to 1. In clocked synchronous mode, serial transmitting is disabled.

- Bit 4—Framing Error (FER): Indicates that data reception ended abnormally due to a framing error in asynchronous mode.

Bit 4: FER	Description
0	Receiving is in progress or has ended normally (Initial value) Clearing the RE bit to 0 in the serial control register does not affect the FER bit, which retains its previous value. FER is cleared to 0 when the chip is reset or enters standby mode, or software reads FER after it has been set to 1, then writes 0 in FER.
1	A receive framing error occurred When the stop bit length is two bits, only the first bit is checked. The second stop bit is not checked. When a framing error occurs, the SCI transfers the receive data into RDR but does not set RDRF. Serial receiving cannot continue while FER is set to 1. In clocked synchronous mode, serial transmitting is also disabled. FER is set to 1 if the stop bit at the end of receive data is checked and found to be 0.

- Bit 3—Parity Error (PER): Indicates that data reception (with parity) ended abnormally due to a parity error in asynchronous mode.

Bit 3: PER	Description
0	Receiving is in progress or has ended normally (Initial value) Clearing the RE bit to 0 in the serial control register does not affect the PER bit, which retains its previous value. PER is cleared to 0 when the chip is reset or enters standby mode, or software reads PER after it has been set to 1, then writes 0 in PER.
1	A receive parity error occurred When a parity error occurs, the SCI transfers the receive data into RDR but does not RDRF. Serial receiving cannot continue while PER is set to 1. In clocked synchronous mode, serial transmitting is also disabled. PER is set to 1 if the number of 1s in receive data, including the parity bit, does not match the even or odd parity setting of the parity mode bit (O/$\overline{E}$) in the serial mode register (SMR).

- Bit 2—Transmit End (TEND): Indicates that when the last bit of a serial character was transmitted, TDR did not contain valid data, so transmission has ended. TEND is a read-only bit and cannot be written.

Bit 2: TEND	Description
0	Transmission is in progress TEND is cleared to 0 when software reads TDRE after it has been set to 1, then writes 0 in TDRE, or the DMAC writes data in TDR.
1	End of transmission (Initial value) TEND is set to 1 when the chip is reset or enters standby mode, TE is cleared to 0 in the serial control register (SCR), or TDRE is 1 when the last bit of a one-byte serial character is transmitted.

- Bit 1—Multiprocessor Bit (MPB): Stores the value of the multiprocessor bit in receive data when a multiprocessor format is selected for receiving in asynchronous mode. MPB is a read-only bit and cannot be written.

Bit 1: MPB	Description
0	Multiprocessor bit value in receive data is 0 (Initial value) If RE is cleared to 0 when a multiprocessor format is selected, MPB retains its previous value.
1	Multiprocessor bit value in receive data is 1

- Bit 0—Multiprocessor Bit Transfer (MPBT): Stores the value of the multiprocessor bit added to transmit data when a multiprocessor format is selected for transmitting in asynchronous mode. The MPBT setting is ignored in clocked synchronous mode, when a multiprocessor format is not selected, or when the SCI is not transmitting.

Bit 0: MPBT	Description
0	Multiprocessor bit value in transmit data is 0 (Initial value)
1	Multiprocessor bit value in transmit data is 1

13.2.8 Bit Rate Register (BRR)

The bit rate register (BRR) is an 8-bit register that, together with the baud rate generator clock source selected by the CKS1 and CKS0 bits in the serial mode register (SMR), determines the serial transmit/receive bit rate.

The CPU can always read and write to BRR. BRR is initialized to H'FF by a reset and in standby mode.

Bit:	7	6	5	4	3	2	1	0
Bit name:								
Initial value:	1	1	1	1	1	1	1	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 13.3 shows examples of BRR settings in asynchronous mode; table 13.4 shows examples of BBR settings in clocked synchronous mode.

Table 13.3 Bit Rates and BRR Settings in Asynchronous Mode

	φ (MHz)											
	4			4.9152			8			9.8304		
Bit Rate (bits/s)	n	N	Error (%)	n	N	Error (%)	n	N	Error (%)	n	N	Error (%)
110	1	70	0.03	1	86	0.31	1	141	0.03	1	174	–0.26
150	0	207	0.16	0	255	0.00	1	103	0.16	1	127	0.00
300	0	103	0.16	0	127	0.00	0	207	0.16	0	255	0.00
600	0	51	0.16	0	63	0.00	0	103	0.16	0	127	0.00
1200	0	25	0.16	0	31	0.00	0	51	0.16	0	63	0.00
2400	0	12	0.16	0	15	0.00	0	25	0.16	0	31	0.00
4800	—	—	—	0	7	0.00	0	12	0.16	0	15	0.00
9600	—	—	—	0	3	0.00	—	—	—	0	7	0.00
19200	—	—	—	0	1	0.00	—	—	—	0	3	0.00
31250	0	0	0.00	—	—	—	0	1	0.00	—	—	—
38400	—	—	—	0	0	0.00	—	—	—	0	1	0.00

	φ (MHz)											
	12			14.7456			16			19.6608		
Bit Rate (bits/s)	n	N	Error (%)	n	N	Error (%)	n	N	Error (%)	n	N	Error (%)
110	1	212	0.03	2	64	0.70	2	70	0.03	2	86	0.31
150	1	155	0.16	1	191	0.00	1	207	0.16	1	255	0.00
300	1	77	0.16	1	95	0.00	1	103	0.16	1	127	0.00
600	0	155	0.16	0	191	0.00	0	207	0.16	0	255	0.00
1200	0	77	0.16	0	95	0.00	0	103	0.16	0	127	0.00
2400	0	38	0.16	0	47	0.00	0	51	0.16	0	63	0.00
4800	0	19	–2.34	0	23	0.00	0	25	0.16	0	31	0.00
9600	0	9	–2.34	0	11	0.00	0	12	0.16	0	15	0.00
19200	0	4	–2.34	0	5	0.00	—	—	—	0	7	0.00
31250	0	2	0.00	—	—	—	0	3	0.00	0	4	–1.70
38400	—	—	—	0	2	0.00	—	—	—	0	3	0.00

Table 13.3 Bit Rates and BRR Settings in Asynchronous Mode (cont)

	φ (MHz)											
	20			24			24.576			28.7		
Bit Rate (bits/s)	n	N	Error (%)	n	N	Error (%)	n	N	Error (%)	n	N	Error (%)
110	2	88	–0.25	2	106	–0.44	2	108	0.08	2	126	0.31
150	2	64	0.16	2	77	0.16	2	79	0.00	2	92	0.46
300	1	129	0.16	1	155	0.16	1	159	0.00	1	186	-0.08
600	1	64	0.16	1	77	0.16	1	79	0.00	1	92	0.46
1200	0	129	0.16	0	155	0.16	0	159	0.00	0	186	-0.08
2400	0	64	0.16	0	77	0.16	0	79	0.00	0	92	0.46
4800	0	32	–1.36	0	38	0.16	0	39	0.00	0	46	-0.61
9600	0	15	1.73	0	19	–2.34	0	19	0.00	0	22	1.55
19200	0	7	1.73	0	9	–2.34	0	9	0.00	0	11	-2.68
31250	0	4	0.00	0	5	0.00	0	5	2.40	0	6	2.50
38400	0	3	1.73	0	4	–2.34	0	4	0.00	0	5	-2.68

Table 13.4 Bit Rates and BRR Settings in Clocked Synchronous Mode

Bit Rate (bits/s)	φ (MHz)							
	4		8		16		28.7	
	n	N	n	N	n	N	n	N
110	2	141	3	70	3	141	3	254
250	1	249	2	124	2	249	3	111
500	1	124	1	249	2	124	2	223
1k	0	249	1	124	1	249	2	111
2.5k	0	99	0	199	1	99	1	178
5k	0	49	0	99	0	199	1	89
10k	0	24	0	49	0	99	0	178
25k	0	9	0	19	0	39	0	71
50k	0	4	0	9	0	19	0	35
100k	—	—	0	4	0	9	0	17
250k	0	0*	0	1	0	3	—	—
500k			0	0*	0	1	—	—
1M					0	0*	—	—

Note: Settings with an error of 1% or less are recommended.

Explanation of symbols:

Blank: No setting possible

—: Setting possible, but error occurs

*: Continuous transmission/reception not possible

The BRR setting is calculated as follows:

Asynchronous mode:

$$N = \frac{\phi}{256 \times 2^{2n-1} \times B} \times 10^6 - 1$$

Clocked synchronous mode:

$$N = \frac{\phi}{32 \times 2^{2n-1} \times B} \times 10^6 - 1$$

B: Bit rate (bit/s)

N: BRR setting for baud rate generator ($0 \leq N \leq 255$)

f: Operating frequency (MHz)

n: Baud rate generator clock source (n = 0, 1, 2, 3) (For the clock sources and values of n, see table 13.6.)

Table 13.5 SMR Settings

		SMR Settings	
n	**Clock Source**	**CKS1**	**CKS0**
0	φ/4	0	0
1	φ/16	0	1
2	φ/164	1	0
3	φ/256	1	1

The bit rate error for asynchronous mode is given by the following equation:

$$\text{Error (\%)} = \left(\frac{\phi \times 10^6}{(N + 1) \times B \times 256 \times 2^{2n-1}} - 1 \right) \times 100$$

Table 13.6 shows the maximum bit rates in asynchronous mode when the baud rate generator is being used. Tables 13.7 and 13.8 show the maximum rates for external clock input.

Table 13.6 Maximum Bit Rates for Various Frequencies with Baud Rate Generator (Asynchronous Mode)

		Settings	
φ (MHz)	**Maximum Bit Rate (bits/s)**	**n**	**N**
4	31250	0	0
4.9152	38400	0	0
8	62500	0	0
9.8304	76800	0	0
12	93750	0	0
14.7456	115200	0	0
16	125000	0	0
19.6608	153600	0	0
20	156250	0	0
24	187500	0	0
24.576	192000	0	0
28.7	224218	0	0

Table 13.7 Maximum Bit Rates with External Clock Input (Asynchronous Mode)

ϕ (MHz)	External Input Clock (MHz)	Maximum Bit Rate (bits/s)
4	0.2500	15625
4.9152	0.3072	19200
8	0.5000	31250
9.8304	0.6144	38400
12	0.7500	46875
14.7456	0.9216	57600
16	1.0000	62500
19.6608	1.2288	76800
20	1.2500	78125
24	1.5000	93750
24.576	1.5360	96000
28.7	1.79375	112109

Table 13.8 Maximum Bit Rates with External Clock Input (Clocked Synchronous Mode)

ϕ (MHz)	External Input Clock (MHz)	Maximum Bit Rate (bits/s)
8	0.3333	333333.3
16	0.6667	666666.7
24	1.0000	1000000.0
28.7	1.1958	1195833.3

13.3 Operation

13.3.1 Overview

For serial communication, the SCI has an asynchronous mode in which characters are synchronized individually, and a clocked synchronous mode in which communication is synchronized with clock pulses. Asynchronous/clocked synchronous mode and the communication format are selected in the serial mode register (SMR), as shown in table 13.9. The SCI clock source is selected by the C/$\overline{A}$ bit in the serial mode register (SMR) and the CKE1 and CKE0 bits in the serial control register (SCR), as shown in table 13.10.

Asynchronous Mode:

- Data length is selectable. seven or eight bits.
- Parity and multiprocessor bits are selectable, as is the stop bit length (one or two bits). The preceding selections constitute the communication format and character length.
- In receiving, it is possible to detect framing errors, parity errors, overrun errors, and the break state.
- An internal or external clock can be selected as the SCI clock source.
  - When an internal clock is selected, the SCI operates using the built-in baud rate generator, and can output a serial clock signal with a frequency matching the bit rate.
  - When an external clock is selected, the external clock input must have a frequency 16 times the bit rate. (The built-in baud rate generator is not used.)

Clocked Synchronous Mode:

- The communication format has a fixed eight-bit data length.
- In receiving, it is possible to detect overrun errors.
- An internal or external clock can be selected as the SCI clock source.
  - When an internal clock is selected, the SCI operates using the built-in baud rate generator, and outputs a synchronous clock signal to external devices.
  - When an external clock is selected, the SCI operates on the input synchronous clock. The built-in baud rate generator is not used.

Table 13.9 Serial Mode Register Settings and SCI Communication Formats

	SMR Settings					SCI Communication Format			
Mode	Bit 7 C/A	Bit 6 CHR	Bit 5 PE	Bit 2 MP	Bit 3 STOP	Data Length	Parity Bit	Multipro-cessor Bit	Stop Bit Length
Asynchronous	0	0	0	0	0	8-bit	Not set	Not set	1 bit
					1				2 bits
			1		0		Set		1 bit
					1				2 bits
		1	0		0	7-bit	Not set		1 bit
					1				2 bits
			1		0		Set		1 bit
					1				2 bits
Asynchronous (multiprocessor format)	0	0	*	1	0	8-bit	Not set	Set	1 bit
			*		1				2 bits
		1	*		0	7-bit			1 bit
			*		1				2 bits
Clocked synchronous	1	*	*	*	*	8-bit	Not set	Not set	None

Note: Asterisks (*) in the table indicate don't care bits.

Table 13.10 SMR and SCR Settings and SCI Clock Source Selection

	SMR	SCR Settings		SCI Transmit/Receive Clock	
Mode	Bit 7 C/$\overline{A}$	Bit 1 CKE1	Bit 0 CKE0	Clock Source	SCK Pin Function
Asynchronous	0	0	0	Internal	SCI does not use the SCK pin
			1		Outputs a clock with frequency matching the bit rate
		1	0	External	Inputs a clock with frequency 16 times the bit rate
			1		
Clocked synch-ronous	1	0	0	Internal	Outputs the synchronous clock
			1		
		1	0	External	Inputs the synchronous clock
			1		

13.3.2 Operation in Asynchronous Mode

In asynchronous mode, each transmitted or received character begins with a start bit and ends with a stop bit. Serial communication is synchronized one character at a time.

The transmitting and receiving sections of the SCI are independent, so full duplex communication is possible. The transmitter and receiver are both double buffered, so data can be written and read while transmitting and receiving are in progress, enabling continuous transmitting and receiving.

Figure 13.2 shows the general format of asynchronous serial communication. In asynchronous serial communication, the communication line is normally held in the mark (high) state. The SCI monitors the line and starts serial communication when the line goes to the space (low) state, indicating a start bit. One serial character consists of a start bit (low), data (LSB first), parity bit (high or low), and stop bit (high), in that order.

When receiving in asynchronous mode, the SCI synchronizes on the falling edge of the start bit. The SCI samples each data bit on the eighth pulse of a clock with a frequency 16 times the bit rate. Receive data is latched at the center of each bit.

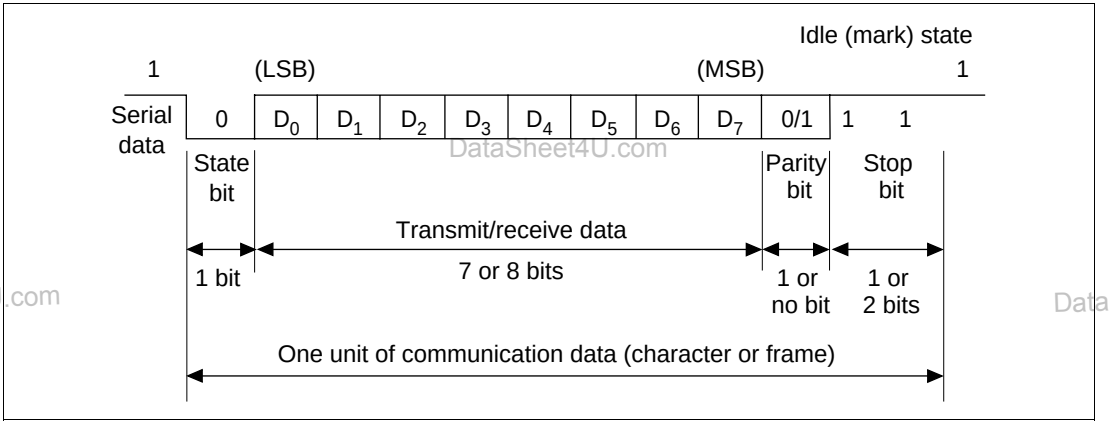


Figure 13.2 Example of Data Format in Asynchronous Communication (8-Bit Data with Parity and Two Stop Bits)

Transmit/Receive Formats. Table 13.11 shows the 12 communication formats that can be selected in asynchronous mode. The format is selected by settings in the serial mode register (SMR).

Table 13.11 Serial Communication Formats (Asynchronous Mode)

SMR Bits				Serial Transmit/Receive Format and Frame Length											
CHR	PE	MP	STOP	1	2	3	4	5	6	7	8	9	10	11	12
0	0	0	0	START	8-bit data								STOP		
0	0	0	1	START	8-bit data								STOP	STOP	
0	1	0	0	START	8-bit data								P	STOP	
0	1	0	1	START	8-bit data								P	STOP	STOP
1	0	0	0	START	7-bit data							STOP			
1	0	0	1	START	7-bit data							STOP	STOP		
1	1	0	0	START	7-bit data							P	STOP		
1	1	0	1	START	7-bit data							P	STOP	STOP	
0	—	1	0	START	8-bit data								MPB	STOP	
0	—	1	1	START	8-bit data								MPB	STOP	STOP
1	—	1	0	START	7-bit data							MPB	STOP		
1	—	1	1	START	7-bit data							MPB	STOP	STOP	

—: Don’t care bits.
START: Start bit
STOP: Stop bit
P: Parity bit
MPB: Multiprocessor bit

Clock: An internal clock generated by the built-in baud rate generator or an external clock input from the SCK pin can be selected as the SCI transmit/receive clock. The clock source is selected by the C/$\overline{A}$ bit in the serial mode register (SMR) and bits CKE1 and CKE0 in the serial control register (SCR) (table 13.9).

When an external clock is input at the SCK pin, it must have a frequency equal to 16 times the desired bit rate.

When the SCI operates on an internal clock, it can output a clock signal at the SCK pin. The frequency of this output clock is equal to the bit rate. The phase is aligned as in figure 13.3 so that the rising edge of the clock occurs at the center of each transmit data bit.

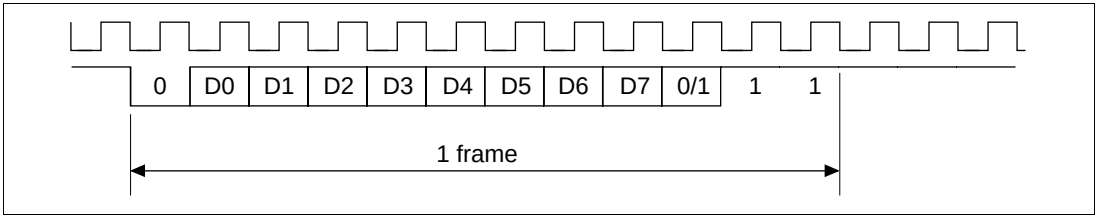


Figure 13.3 Output Clock and Serial Data Timing (Asynchronous Mode)

Transmitting and Receiving Data

SCI Initialization (Asynchronous Mode): Before transmitting or receiving, clear the TE and RE bits to 0 in the serial control register (SCR), then initialize the SCI as follows.

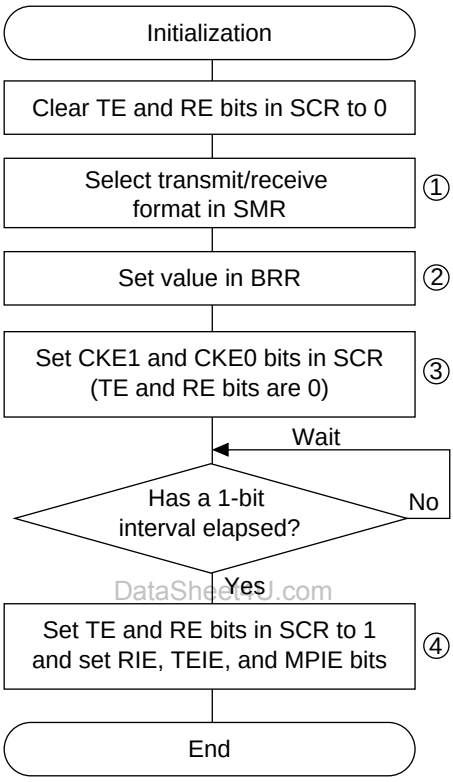
When changing the operation mode or communication format, always clear the TE and RE bits to 0 before following the procedure given below. Clearing TE to 0 sets TDRE to 1 and initializes the transmit shift register (TSR). Clearing RE to 0, however, does not initialize the RDRF, PER, FER, and ORER flags and receive data register (RDR), which retain their previous contents.

When an external clock is used, the clock should not be stopped during initialization or subsequent operation. SCI operation becomes unreliable if the clock is stopped.

Figure 13.4 shows a sample flowchart for initializing the SCI. The procedure for initializing the SCI is as follows:

1. Select the communication format in the serial mode register (SMR).
2. Write the value corresponding to the bit rate in the bit rate register (BRR) unless an external clock is used.
3. Select the clock source in the serial control register (SCR). Leave RIE, TIE, TEIE, MPIE, TE and RE cleared to 0. If clock output is selected in asynchronous mode, clock output starts immediately after the setting is made in SCR.

4. Wait for at least the interval required to transmit or receive one bit, then set TE or RE in the serial control register (SCR) to 1. Also set RIE, TIE, TEIE and MPIE as necessary. Setting TE or RE enables the SCI to use the TxD or RxD pin. The initial states are the mark state when transmitting, and the idle state when receiving (waiting for a start bit).

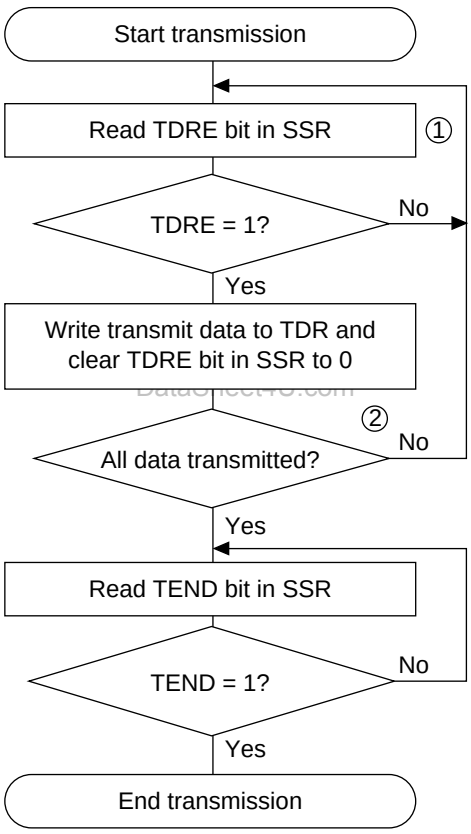


Note: Circled numbers refer to the preceding description of the procedure in the text.

Figure 13.4 Sample Flowchart for SCI Initialization

Transmitting Serial Data (Asynchronous Mode): Figure 13.5 shows a sample flowchart for transmitting serial data. The procedure for transmitting serial data is as follows:

1. SCI status check and transmit data write: read the serial status register (SSR), check that the TDRE bit is 1, then write transmit data in the transmit data register (TDR) and clear TDRE to 0.
2. To continue transmitting serial data, read the TDRE bit to check whether it is safe to write (if it reads 1); if so, write data in TDR, then clear TDRE to 0. When the DMAC is started by a transmit-data-empty interrupt request (TXI) in order to write data in TDR, the TDRE bit is checked and cleared automatically.



Note: Circled numbers refer to the preceding description of the procedure in the text.

Figure 13.5 Sample Flowchart for Transmitting Serial Data

In transmitting serial data, the SCI operates as follows:

1. The SCI monitors the TDRE bit in SSR. When TDRE is cleared to 0, the SCI recognizes that the transmit data register (TDR) contains new data, and loads this data from TDR into the transmit shift register (TSR).

2. After loading the data from TDR into TSR, the SCI sets the TDRE bit to 1 and starts transmitting. If the transmit-data-empty interrupt enable bit (TIE) is set to 1 in SCR, the SCI requests a transmit-data-empty interrupt (TXI) at this time.

   Serial transmit data is transmitted in the following order from the TxD pin:

   a. Start bit: one 0-bit is output.
   b. Transmit data: seven or eight bits of data are output, LSB first.
   c. Parity bit or multiprocessor bit: one parity bit (even or odd parity) or one multiprocessor bit is output. Formats in which neither a parity bit nor a multiprocessor bit is output can also be selected.
   d. Stop bit: one or two 1-bits (stop bits) are output.
   e. Mark state: output of 1-bits continues until the start bit of the next transmit data.

3. The SCI checks the TDRE bit when it outputs the stop bit. If TDRE is 0, the SCI loads new data from TDR into TSR, outputs the stop bit, then begins serial transmission of the next frame. If TDRE is 1, the SCI sets the TEND bit to 1 in SSR, outputs the stop bit, then continues output of 1-bits (mark state). If the transmit-end interrupt enable bit (TEIE) in SCR is set to 1, a transmit-end interrupt (TEI) is requested.

Figure 13.6 shows an example of SCI transmit operation in asynchronous mode.

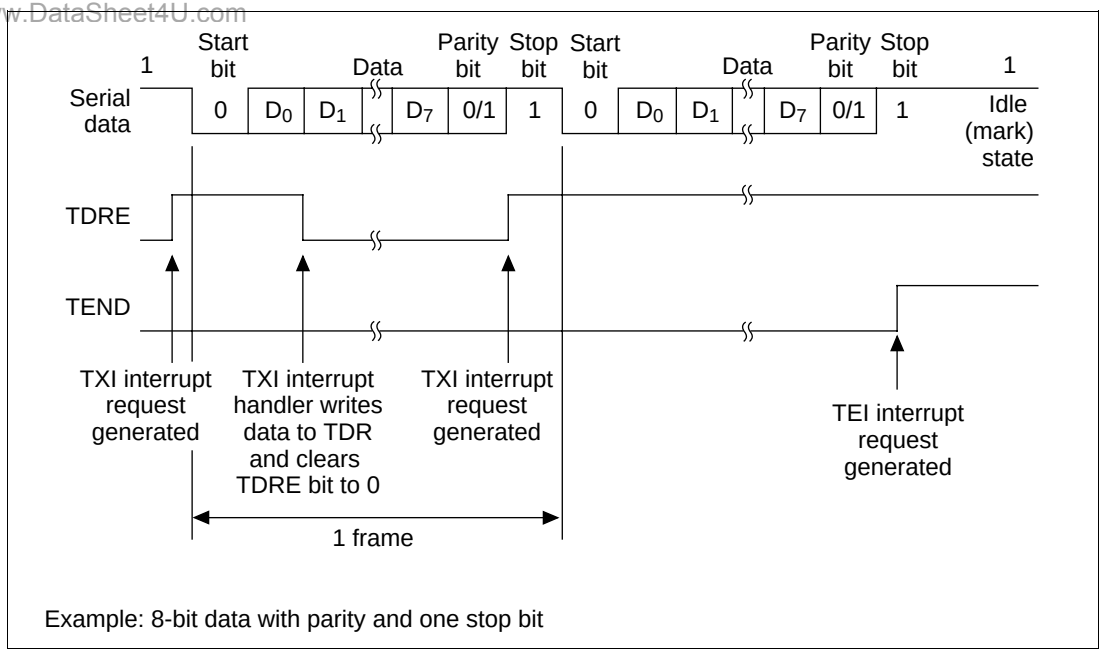
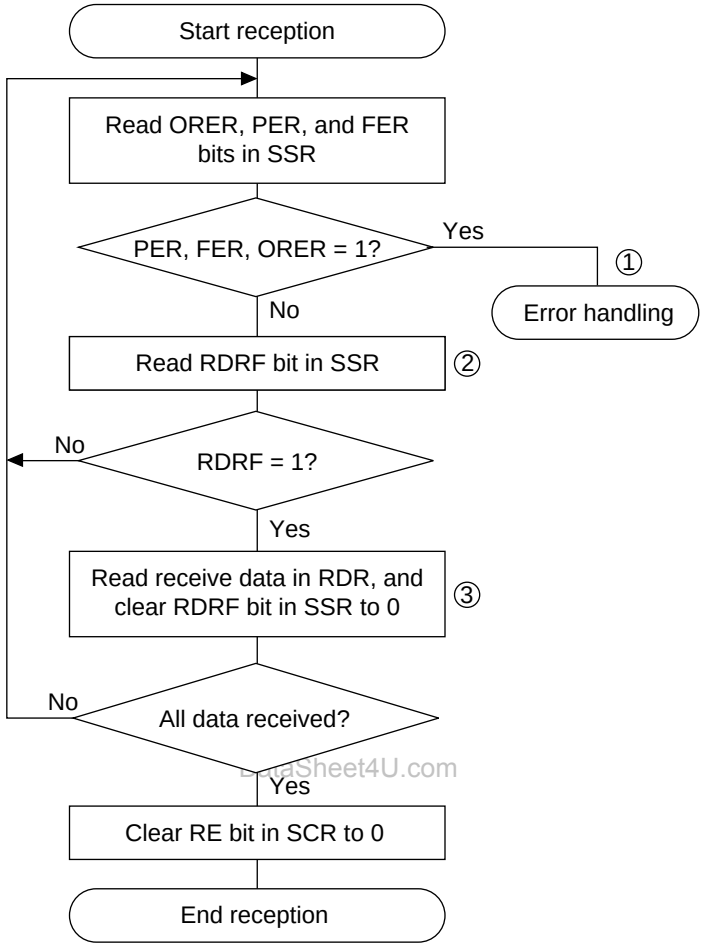


Figure 13.6 Example of SCI Transmit Operation in Asynchronous Mode (8-Bit Data with Parity and One Stop Bit)

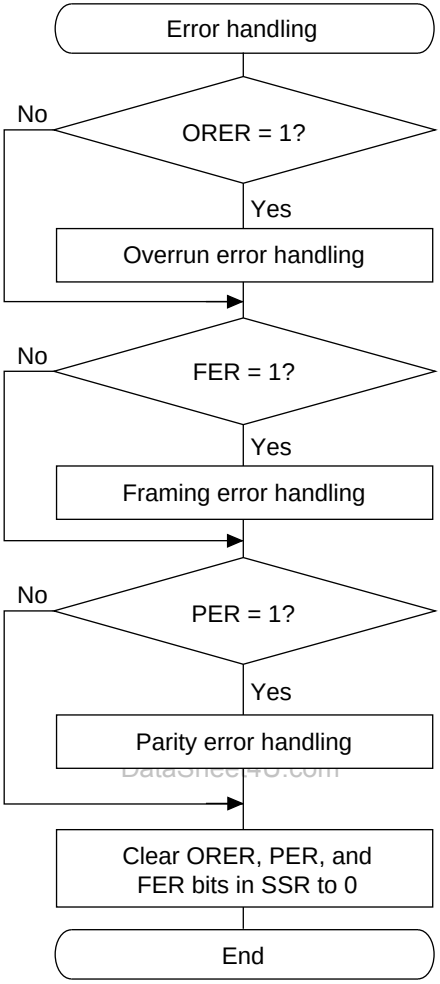
Receiving Serial Data (Asynchronous Mode): Figure 13.7 shows a sample flowchart for receiving serial data. The procedure for receiving serial data is as follows:

1. Receive error handling: if a receive error occurs, read the ORER, PER and FER bits of the SSR to identify the error. After executing the necessary error handling, clear ORER, PER and FER all to 0. Receiving cannot resume if ORER, PER or FER remain set to 1.
2. SCI status check and receive-data read: read the serial status register (SSR), check that RDRF is set to 1, then read receive data from the receive data register (RDR) and clear RDRF to 0. The RXI interrupt can also be used to determine if the RDRF bit has changed from 0 to 1.
3. To continue receiving serial data: read the RDRF and RDR bits and clear RDRF to 0 before the stop bit of the current frame is received. If the DMAC is started by a receive-data-full interrupt (RXI) to read RDR, the RDRF bit is cleared automatically so this step is unnecessary.



Note: Circled numbers refer to the preceding description of the procedure in the text.

Figure 13.7 Sample Flowchart for Receiving Serial Data



Note: Circled numbers refer to the preceding description of the procedure in the text.

Figure 13.7 Sample Flowchart for Receiving Serial Data (cont)

In receiving, the SCI operates as follows:

1. The SCI monitors the receive data line. When it detects a start bit (0), the SCI synchronizes internally and starts receiving.
2. Receive data is shifted into RSR in order from LSB to MSB.
3. The parity bit and stop bit are received. After receiving these bits, the SCI makes the following checks:
   a. Parity check: the number of 1s in the receive data must match the even or odd parity setting of the O/$\overline{E}$ bit in SMR.
   b. Stop bit check: the stop bit value must be 1. If there are two stop bits, only the first stop bit is checked.
   c. Status check: RDRF must be 0 so that receive data can be loaded from RSR into RDR.

   If these checks all pass, the SCI sets RDRF to 1 and stores the received data in RDR. If one of the checks fails (receive error), the SCI operates as indicated in table 13.11.

   Note: When a receive error flag is set, further receiving is disabled. In reception, the RDRF bit is not set to 1. Be sure to clear the error flags.

4. After setting RDRF to 1, if the receive-data-full interrupt enable bit (RIE) is set to 1 in SCR, the SCI requests a receive-data-full interrupt (RXI). If one of the error flags (ORER, PER, or FER) is set to 1 and the receive-data-full interrupt enable bit (RIE) in SCR is also set to 1, the SCI requests a receive-error interrupt (ERI).

Figure 13.8 shows an example of SCI receive operation in asynchronous mode.

Table 13.12 Receive Error Conditions and SCI Operation

Receive Error	Abbreviation	Condition	Data Transfer
Overrun error	ORER	Receiving of next data ends while RDRF is still set to 1 in SSR	Receive data not loaded from RSR into RDR
Framing error	FER	Stop bit is 0	Receive data loaded from RSR into RDR
Parity error	PER	Parity of receive data differs from even/odd parity setting in SMR	Receive data loaded from RSR into RDR

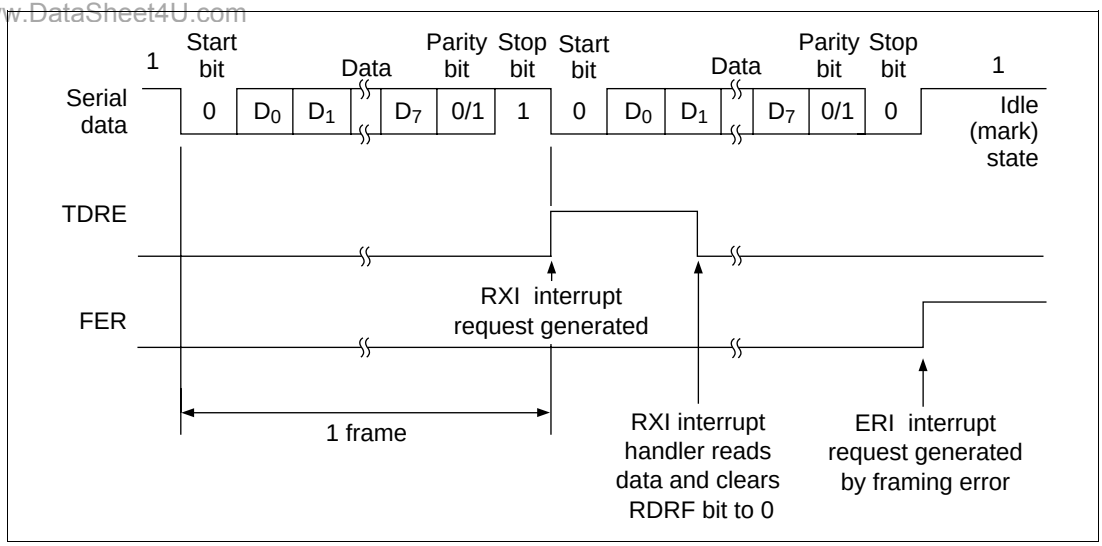


Figure 13.8 Example of SCI Receive Operation (8-Bit Data with Parity and One Stop Bit)

13.3.3 Multiprocessor Communication

The multiprocessor communication function enables several processors to share a single serial communication line. The processors communicate in asynchronous mode using a format with an additional multiprocessor bit (multiprocessor format).

In multiprocessor communication, each receiving processor is addressed by a unique ID. A serial communication cycle consists of an ID-sending cycle that identifies the receiving processor, and a data-sending cycle. The multiprocessor bit distinguishes ID-sending cycles from data-sending cycles. The transmitting processor starts by sending the ID of the receiving processor with which it wants to communicate as data with the multiprocessor bit set to 1. Next, the transmitting processor sends transmit data with the multiprocessor bit cleared to 0.

Receiving processors skip incoming data until they receive data with the multiprocessor bit set to 1. When they receive data with the multiprocessor bit set to 1, receiving processors compare the data with their IDs. The receiving processor with a matching ID continues to receive further incoming data. Processors with IDs not matching the received data skip further incoming data until they again receive data with the multiprocessor bit set to 1. Multiple processors can send and receive data in this way.

Figure 13.9 shows an example of communication among processors using the multiprocessor format.

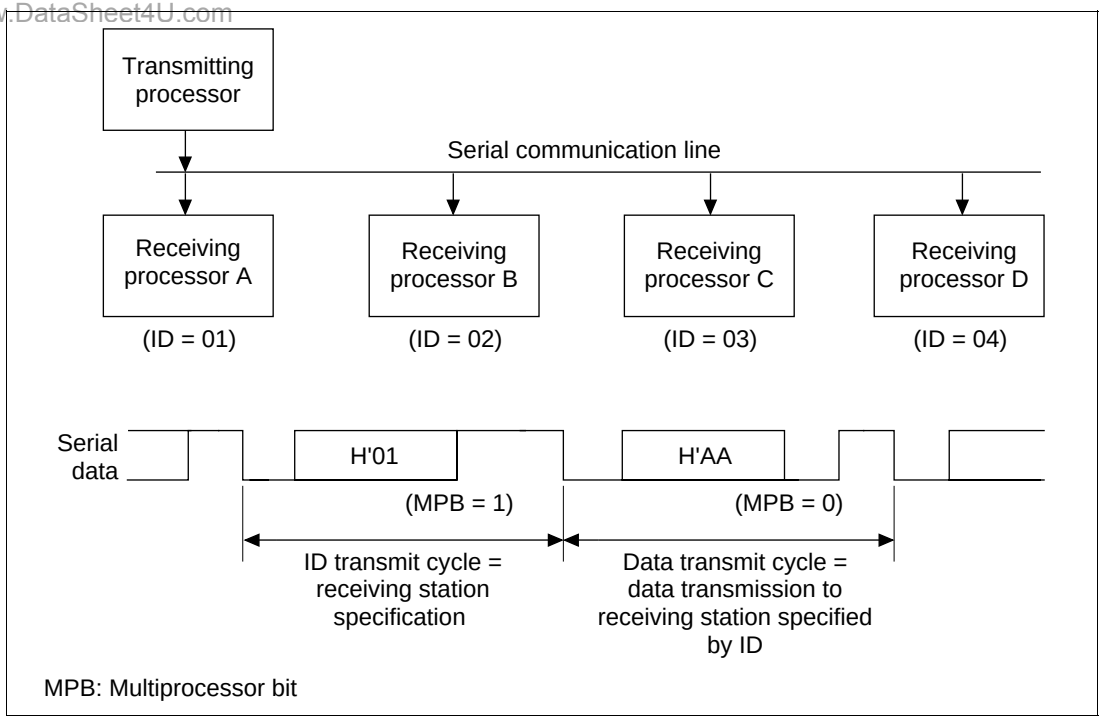


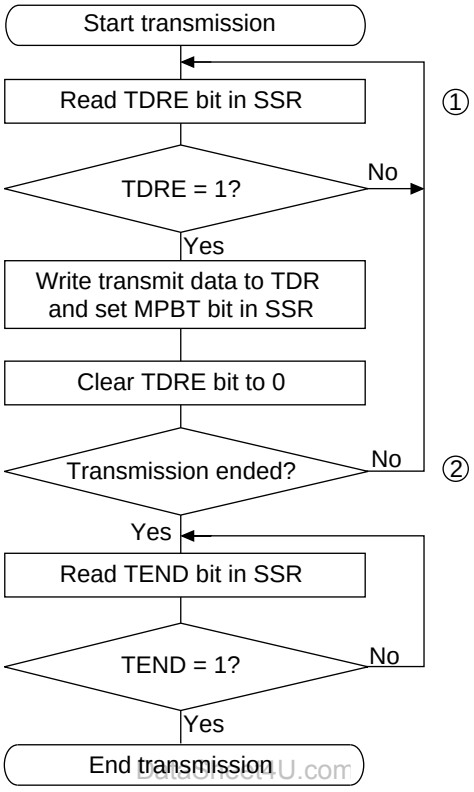
Figure 13.9 Example of Communication among Processors Using Multiprocessor Format (Sending Data H'AA to Receiving Processor A)

Communication Formats: Four formats are available. Parity-bit settings are ignored when the multiprocessor format is selected. For details see table 13.8.

Clock: See the description in the asynchronous mode section.

Transmitting Multiprocessor Serial Data: Figure 13.10 shows a sample flowchart for transmitting multiprocessor serial data. The procedure for transmitting multiprocessor serial data is as follows:

1. SCI status check and transmit data write: read the serial status register (SSR), check that the TDRE bit is 1, then write transmit data in the transmit data register (TDR). Also set the MPBT (multiprocessor bit transfer) bit to 0 or 1 in SSR. Finally, clear TDRE to 0.
2. To continue transmitting serial data, read the TDRE bit to check whether it is safe to write (if it reads 1); if so, write data in TDR, then clear TDRE to 0. When the DMAC is started by a transmit-data-empty interrupt request (TXI) to write data in TDR, the TDRE bit is checked and cleared automatically.



Note: Circled numbers refer to the preceding description of the procedure in the text.

Figure 13.10 Sample Flowchart for Transmitting Multiprocessor Serial Data

In transmitting serial data, the SCI operates as follows:

1. The SCI monitors the TDRE bit in SSR. When TDRE is cleared to 0 the SCI recognizes that the transmit data register (TDR) contains new data, and loads this data from TDR into the transmit shift register (TSR).

2. After loading the data from TDR into TSR, the SCI sets the TDRE bit to 1 and starts transmitting. If the transmit-data-empty interrupt enable bit (TIE) in SCR is set to 1, the SCI requests a transmit-data-empty interrupt (TXI) at this time.

   Serial transmit data is transmitted in the following order from the TxD pin:

   a. Start bit: one 0 bit is output.
   b. Transmit data: seven or eight bits are output, LSB first.
   c. Multiprocessor bit: one multiprocessor bit (MPBT value) is output.
   d. Stop bit: one or two 1-bits (stop bits) are output.
   e. Mark state: output of 1-bits continues until the start bit of the next transmit data.

3. The SCI checks the TDRE bit when it outputs the stop bit. If TDRE is 0, the SCI loads data from TDR into TSR, outputs the stop bit, then begins serial transmission of the next frame. If TDRE is 1, the SCI sets the TEND bit in SSR to 1, outputs the stop bit, then continues output of 1-bits in the mark state. If the transmit-end interrupt enable bit (TEIE) in SCR is set to 1, a transmit-end interrupt (TEI) is requested at this time.

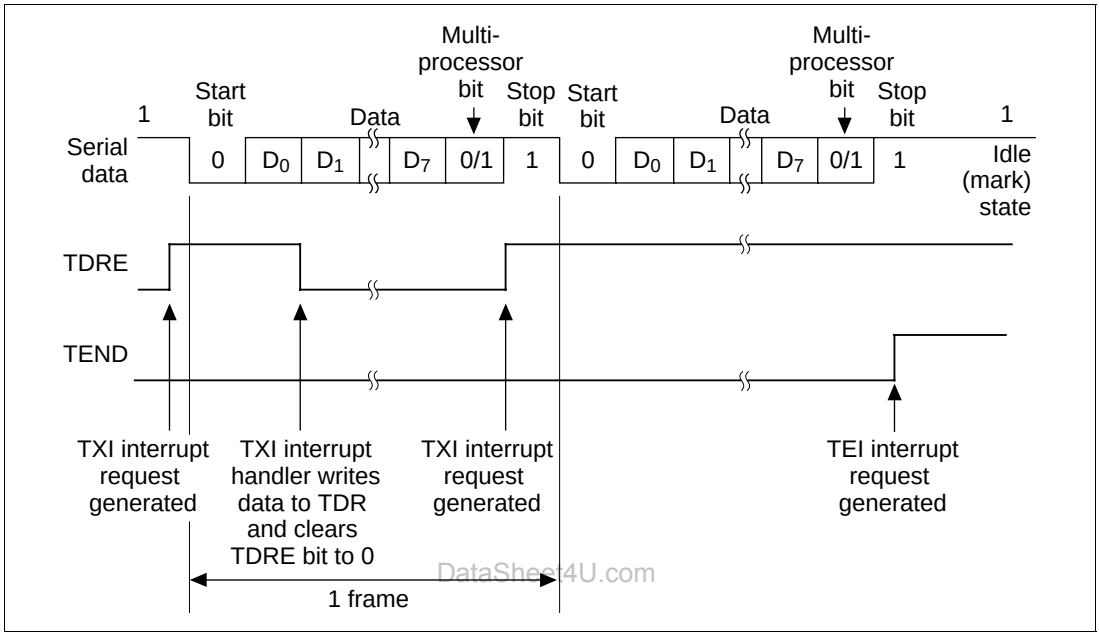
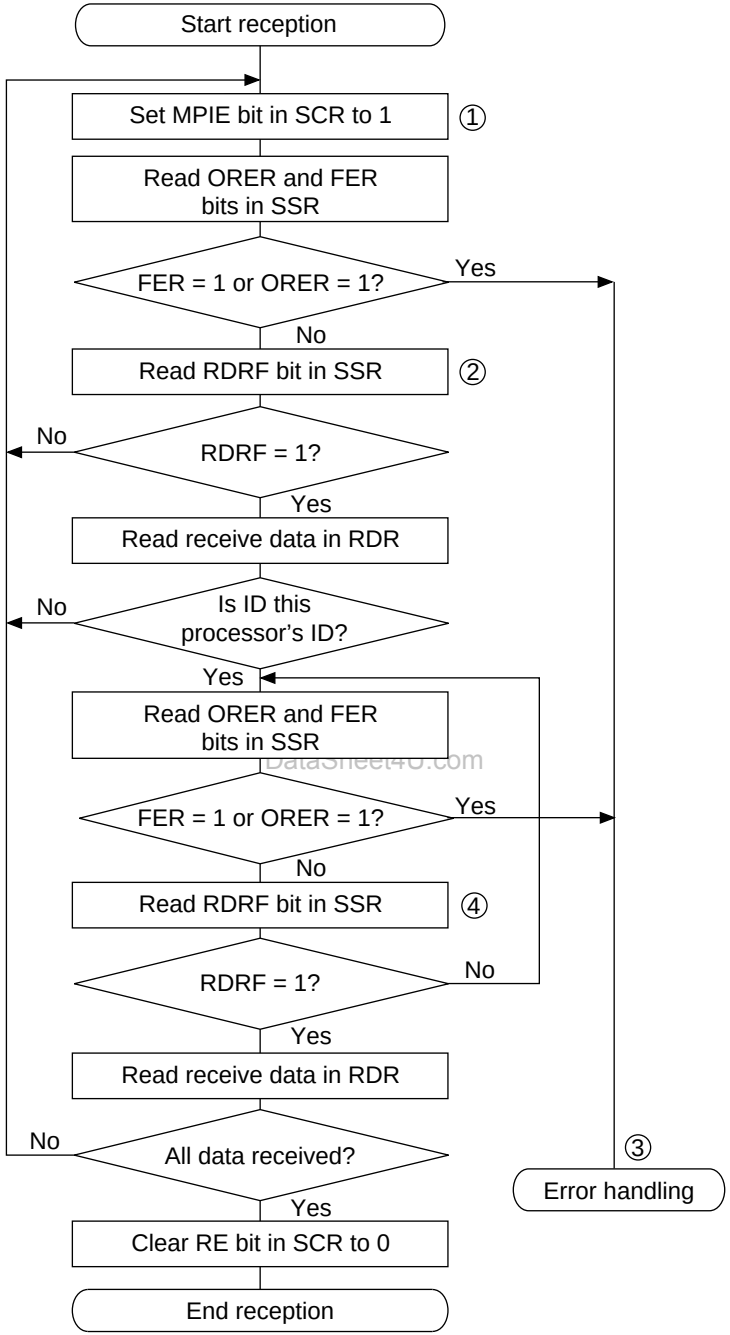


Figure 13.11 Example of SCI Multiprocessor Transmit Operation (8-Bit Data with Multiprocessor Bit and One Stop Bit)

Receiving Multiprocessor Serial Data: Figure 13.12 shows a sample flowchart for receiving multiprocessor serial data. The procedure for receiving multiprocessor serial data is as follows.

1. ID receive cycle: set the MPIE bit in the serial control register (SCR) to 1.
2. SCI status check, ID reception and comparison: read the serial status register (SSR), check that RDRF is set to 1, then read data from the receive data register (RDR) and compare with the processor's own ID. If the ID does not match the receive data, set MPIE to 1 again and clear RDRF to 0. If the ID matches the receive data, clear RDRF to 0.
3. Receive error handling: if a receive error occurs (figure 13.12 (cont)), read the ORER and FER bits in SSR to identify the error. After executing the necessary error handling, clear both ORER and FER to 0. Receiving cannot resume if ORER or FER remains set to 1. When a framing error occurs, the RxD pin can be read to detect the break state.
4. SCI status check and data receiving: read SSR, check that RDRF is set to 1, then read data from the receive data register (RDR).



Note: Circled numbers refer to the preceding description of the procedure in the text.

Figure 13.12 Sample Flowchart for Receiving Multiprocessor Serial Data

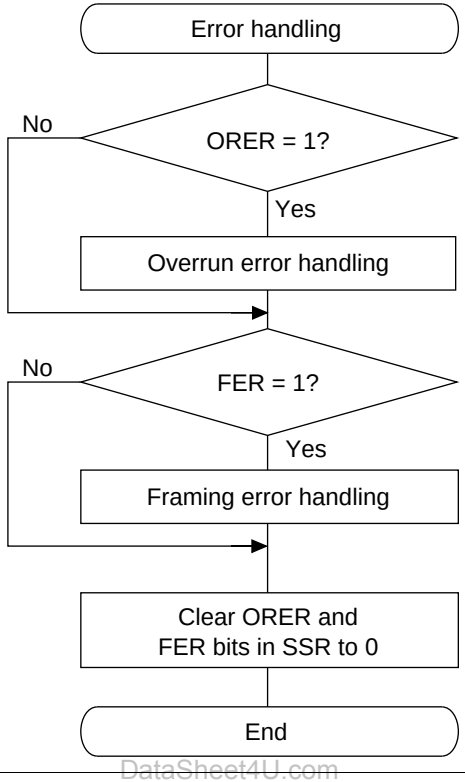


Figure 13.12 Sample Flowchart for Receiving Multiprocessor Serial Data (cont)

Figure 13.13 shows an example of SCI receive operation using a multiprocessor format.

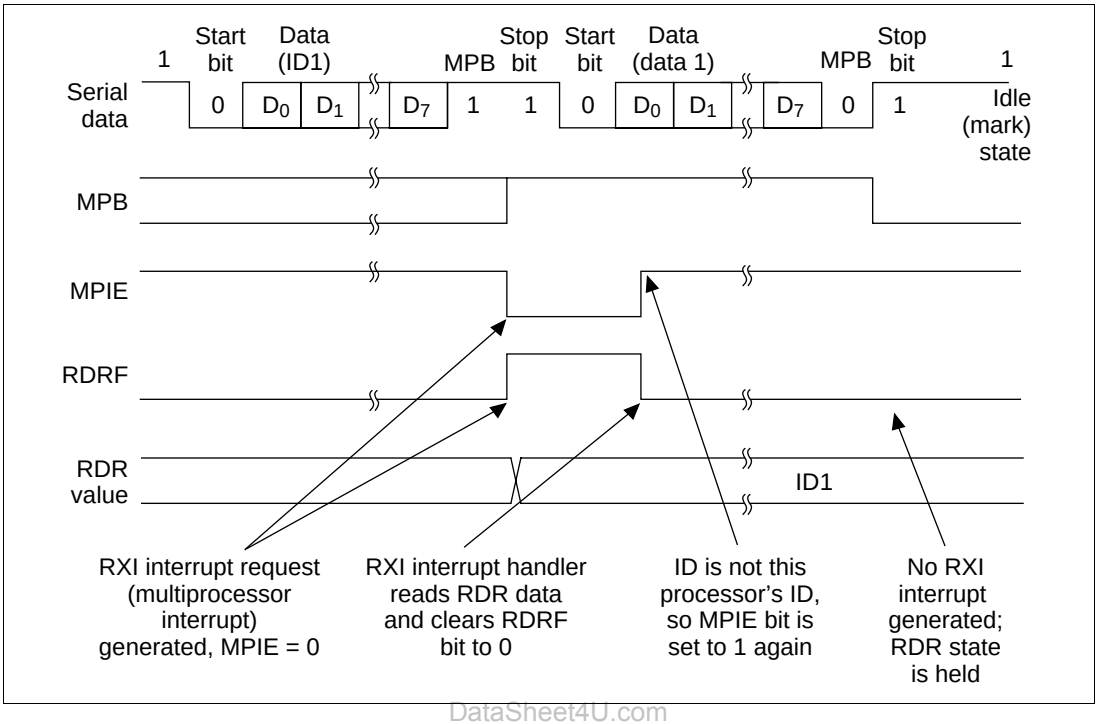


Figure 13.13 Example of SCI Receive Operation
(Own ID Does Not Match Data, 8-Bit Data with Multiprocessor Bit and One Stop Bit)

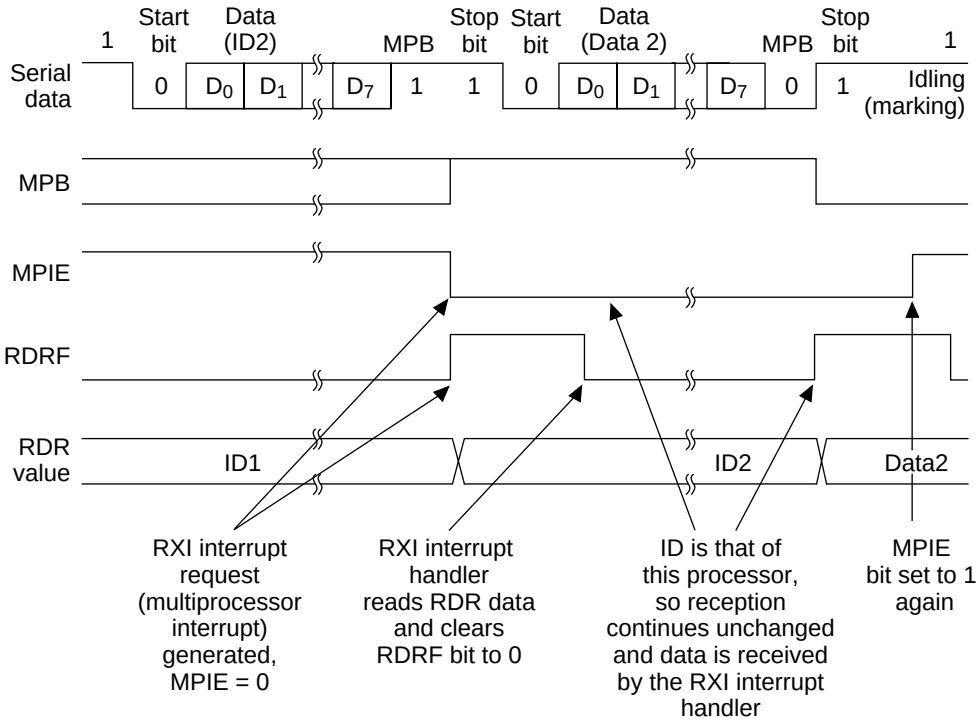


Figure 13.13 Example of SCI Receive Operation
(Own ID Matches Data, 8-Bit Data with Multiprocessor Bit and One Stop Bit) (cont)

13.3.4 Clocked Synchronous Operation

In clocked synchronous mode, the SCI transmits and receives data in synchronization with clock pulses. This mode is suitable for high-speed serial communication.

The SCI transmitter and receiver are independent, so full duplex communication is possible while sharing the same clock. The transmitter and receiver are also double buffered, so continuous transmitting or receiving is possible by reading or writing data while transmitting or receiving is in progress.

Figure 13.14 shows the general format in clocked synchronous serial communication.

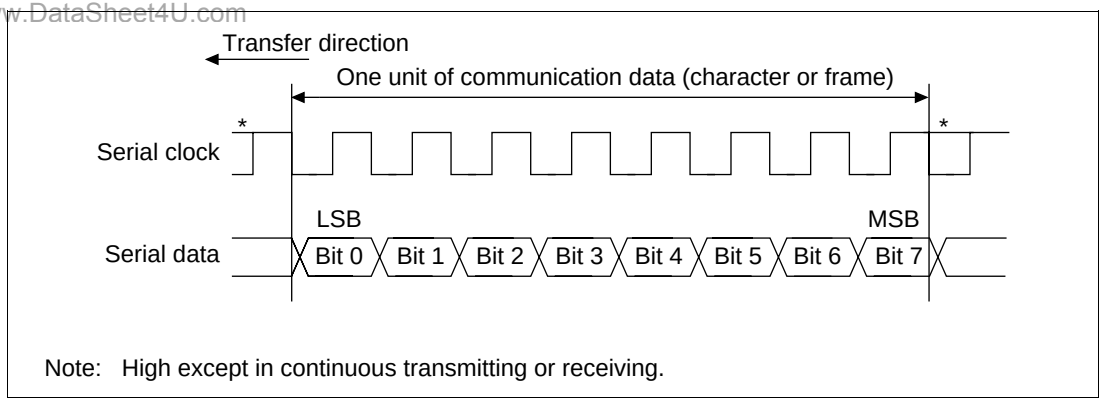


Figure 13.14 Data Format in Clocked Synchronous Communication

In clocked synchronous serial communication, each data bit is output on the communication line from one falling edge of the serial clock to the next. Data is guaranteed valid at the rising edge of the serial clock. In each character, the serial data bits are transmitted in order from LSB (first) to MSB (last). After output of the MSB, the communication line remains in the state of the MSB. In clocked synchronous mode, the SCI transmits or receives data by synchronizing with the falling edge of the serial clock.

Communication Format: The data length is fixed at eight bits. No parity bit or multiprocessor bit can be added.

Clock: An internal clock generated by the built-in baud rate generator or an external clock input from the SCK pin can be selected as the SCI transmit/receive clock. The clock source is selected by the C/$\overline{A}$ bit in the serial mode register (SMR) and bits CKE1 and CKE0 in the serial control register (SCR). See table 13.9.

When the SCI operates on an internal clock, it outputs the clock signal at the SCK pin. Eight clock pulses are output per transmitted or received character. When the SCI is not transmitting or receiving, the clock signal remains in the high state.

Figure 13.15 shows an example of SCI transmit operation. In transmitting serial data, the SCI operates as follows.

1. The SCI monitors the TDRE bit in SSR. When TDRE is cleared to 0 the SCI recognizes that the transmit data register (TDR) contains new data and loads this data from TDR into the transmit shift register (TSR).
2. After loading the data from TDR into TSR, the SCI sets the TDRE bit to 1 and starts transmitting. If the transmit-data-empty interrupt enable bit (TIE) in SCR is set to 1, the SCI requests a transmit-data-empty interrupt (TXI) at this time.

   If clock output mode is selected, the SCI outputs eight synchronous clock pulses. If an external clock source is selected, the SCI outputs data in synchronization with the input clock. Data is output from the TxD pin in order from LSB (bit 0) to MSB (bit 7).

3. The SCI checks the TDRE bit when it outputs the MSB (bit 7). If TDRE is 0, the SCI loads data from TDR into TSR, transmits the MSB, then begins serial transmission of the next frame. If TDRE is 1, the SCI sets the TEND bit in SSR to 1, transmits the MSB, then holds the transmit data pin (TxD) in the MSB state. If the transmit-end interrupt enable bit (TEIE) in SCR is set to 1, a transmit-end interrupt (TEI) is requested at this time.
4. After the end of serial transmission, the SCK pin is held in the high state.

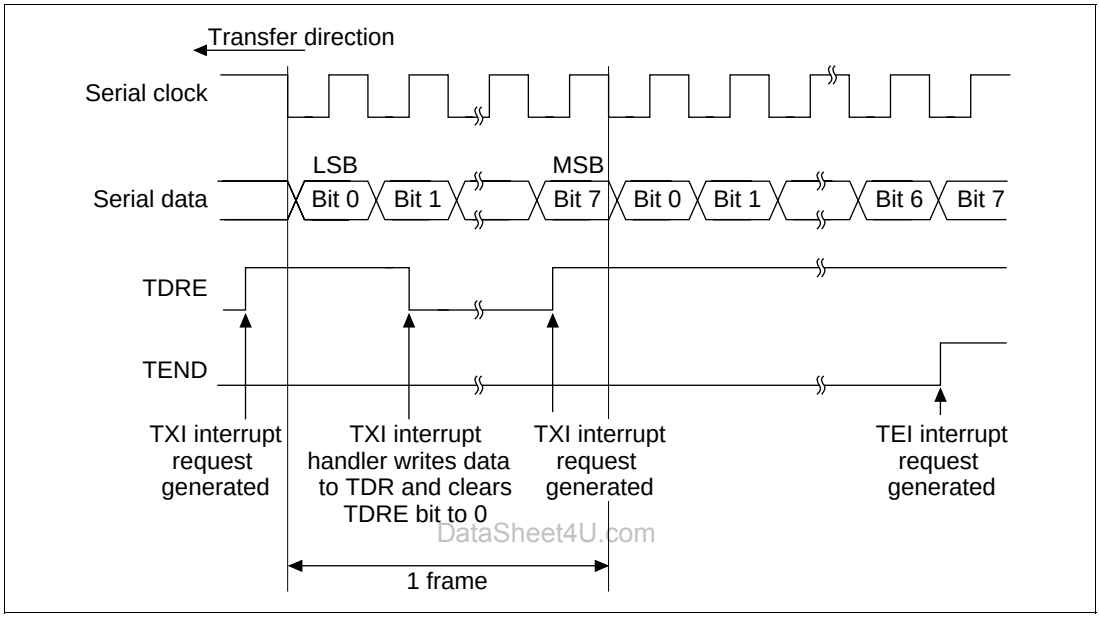


Figure 13.15 Example of SCI Transmit Operation

Transmitting and Receiving Data

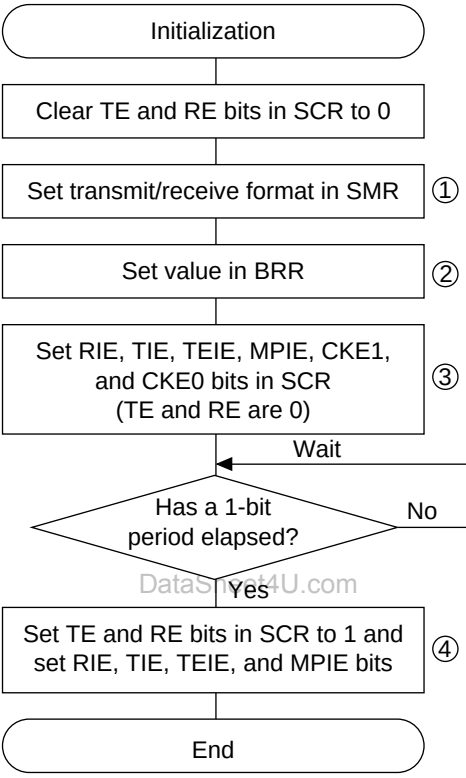
SCI Initialization (Clocked Synchronous Mode): Before transmitting or receiving, software must clear the TE and RE bits to 0 in the serial control register (SCR), then initialize the SCI as follows.

When changing the mode or communication format, always clear the TE and RE bits to 0 before following the procedure given below. Clearing TE to 0 sets TDRE to 1 and initializes the transmit shift register (TSR). Clearing RE to 0, however, does not initialize the RDRF, PER, FER, and ORER flags and receive data register (RDR), which retain their previous contents.

Figure 13.16 shows a sample flowchart for initializing the SCI. The procedure for initializing the SCI is as follows.

1. Select the communication format in the serial mode register (SMR).
2. Write the value corresponding to the bit rate in the bit rate register (BRR) unless an external clock is used.

3. Select the clock source in the serial control register (SCR). Leave RIE, TIE, TEIE, MPIE, TE and RE cleared to 0.
4. Wait for at least the interval required to transmit or receive one bit, then set TE or RE in the serial control register (SCR) to 1. Also set RIE, TIE, TEIE and MPIE.

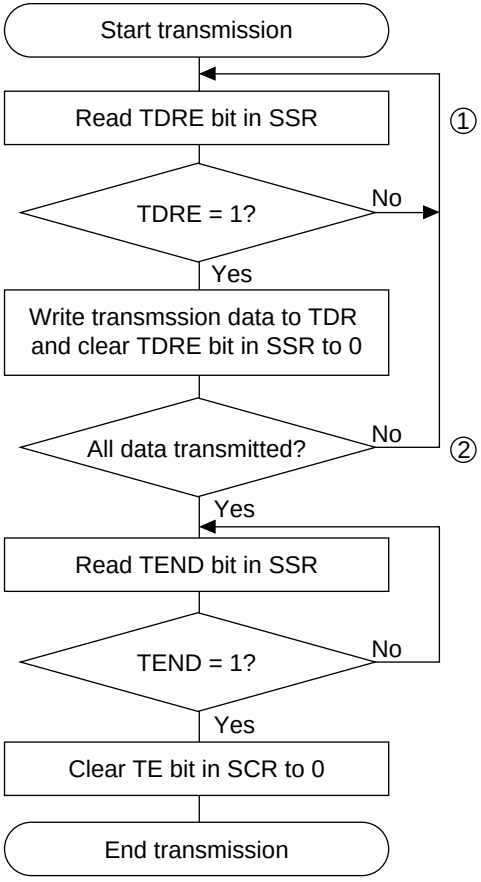


Note: Circled numbers refer to the preceding description of the procedure in the text.

Figure 13.16 Sample Flowchart for SCI Initialization

Transmitting Serial Data (Clocked Synchronous Mode): Figure 13.17 shows a sample flowchart for transmitting serial data. The procedure for transmitting serial data is as follows.

1. SCI status check and transmit data write: read the serial status register (SSR), check that the TDRE bit is 1, then write transmit data in the transmit data register (TDR) and clear TDRE to 0.
2. To continue transmitting serial data, read the TDRE bit to check whether it is safe to write (if it reads 1); if so, write data in TDR, then clear TDRE to 0. When the DMAC is started by a transmit-data-empty interrupt request (TXI) to write data in TDR, the TDRE bit is checked and cleared automatically.



Note: Circled numbers refer to the preceding description of the procedure in the text.

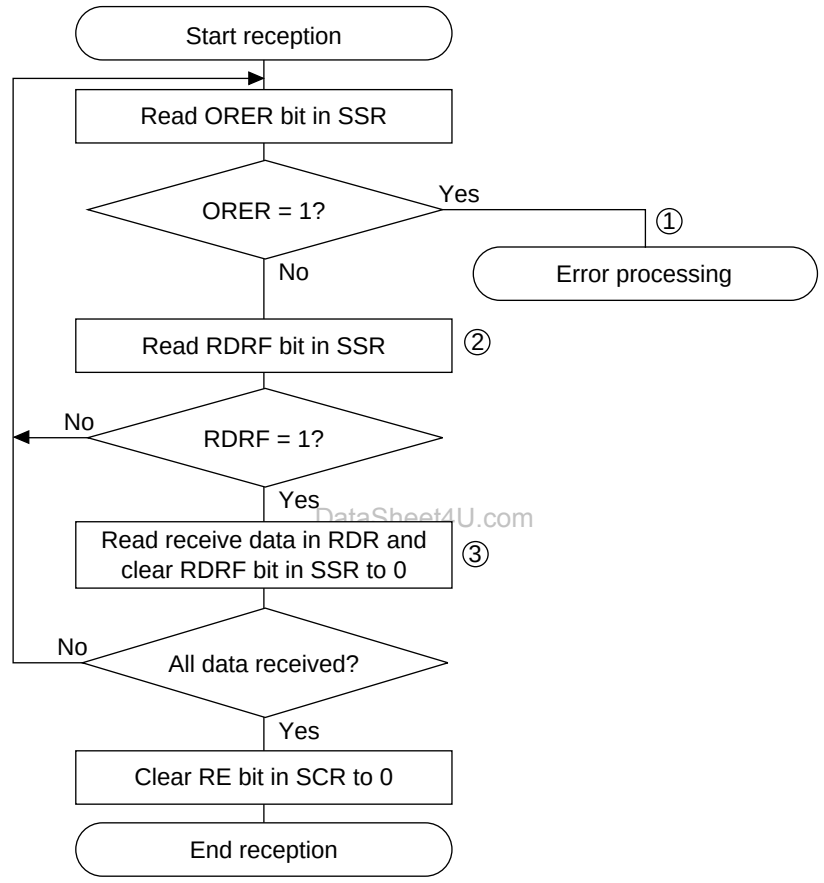
Figure 13.17 Sample Flowchart for Serial Transmitting

Receiving Serial Data (Clocked Synchronous Mode): Figure 13.18 shows a sample flowchart for receiving serial data. When switching from asynchronous mode to clocked synchronous mode, make sure that ORER, PER, and FER are cleared to 0. If PER or FER is set to 1, the RDRF bit will not be set and both transmitting and receiving will be disabled. Figure 13.19 shows an example of the SCI receive operation.

The procedure for receiving serial data is as follows:

1. Receive error handling: if a receive error occurs, read the ORER bit in SSR to identify the error. After executing the necessary error handling, clear ORER to 0. Transmitting/receiving cannot resume if ORER remains set to 1.

2. SCI status check and receive data read: read the serial status register (SSR), check that RDRF is set to 1, then read receive data from the receive data register (RDR) and clear RDRF to 0. The RXI interrupt can also be used to determine if the RDRF bit has changed from 0 to 1.
3. To continue receiving serial data: read RDR, and clear RDRF to 0 before the MSB (bit 7) of the current frame is received. If the DMAC is started by a receive-data-full interrupt (RXI) to read RDR, the RDRF bit is cleared automatically so this step is unnecessary.



Note: Circled numbers refer to the preceding description of the procedure in the text.

Figure 13.18 Sample Flowchart for Serial Receiving

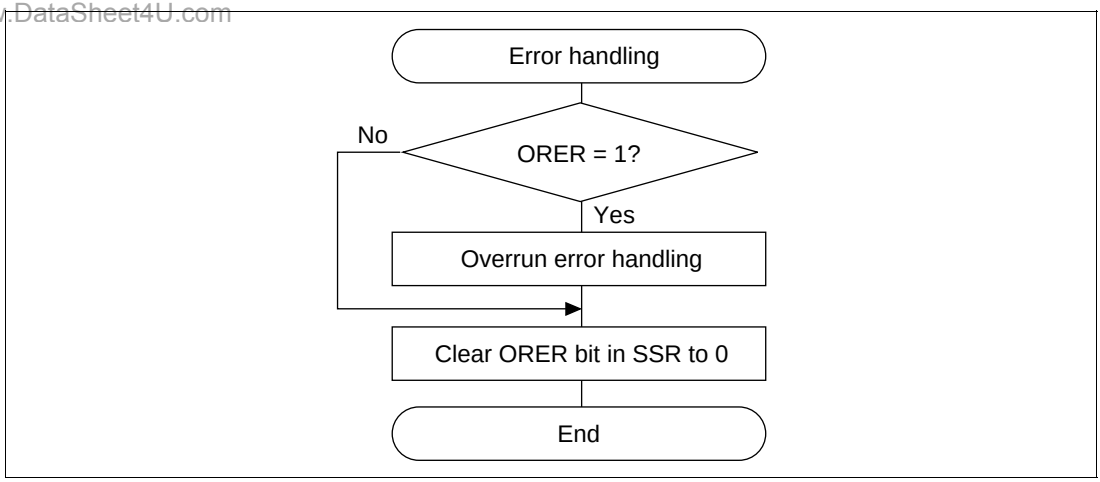


Figure 13.18 Sample Flowchart for Serial Receiving (cont)

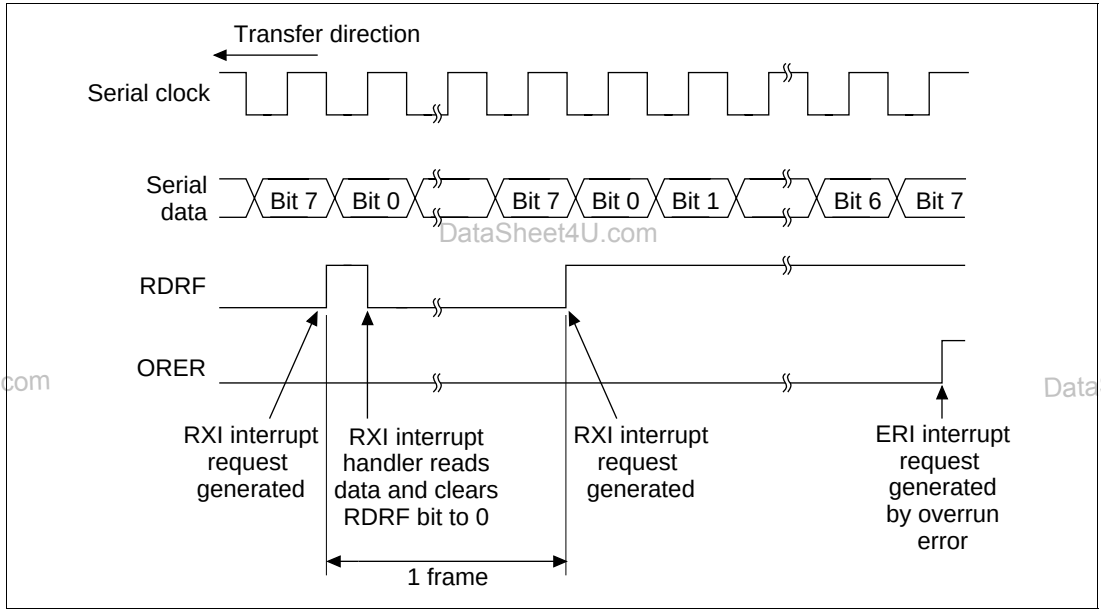


Figure 13.19 Example of SCI Receive Operation

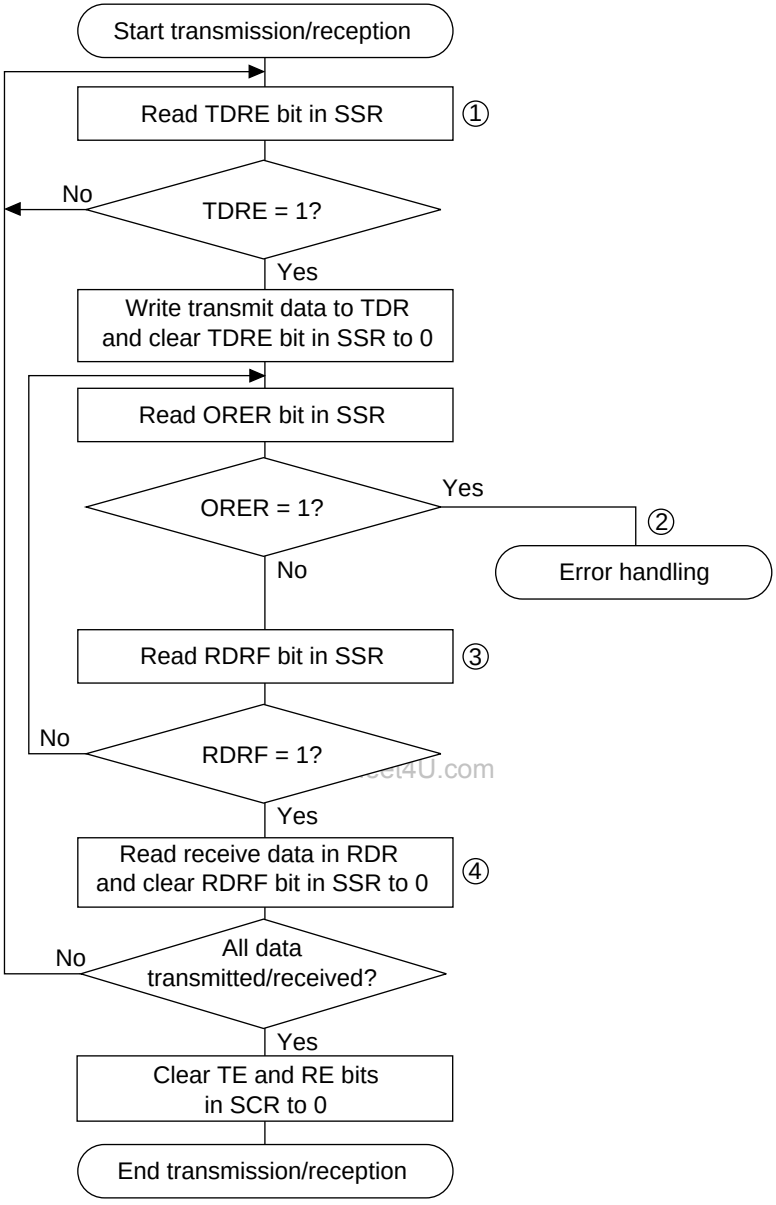
In receiving, the SCI operates as follows:

1. The SCI synchronizes with serial clock input or output and initializes internally.
2. Receive data is shifted into RSR in order from LSB to MSB. After receiving the data, the SCI checks that RDRF is 0 so that receive data can be loaded from RSR into RDR. If this check passes, the SCI sets RDRF to 1 and stores the received data in RDR. If the check does not pass (receive error), the SCI operates as indicated in table 13.8. The RDRF bit is not set to 1. Be sure to clear the error flag.
3. After setting RDRF to 1, if the receive-data-full interrupt enable bit (RIE) is set to 1 in SCR, the SCI requests a receive-data-full interrupt (RXI). If the ORER bit is set to 1 and the receive-data-full interrupt enable bit (RIE) in SCR is also set to 1, the SCI requests a receive-error interrupt (ERI).

Transmitting and Receiving Serial Data Simultaneously (Clocked Synchronous Mode):
Figure 13.20 shows a sample flowchart for transmitting and receiving serial data simultaneously. The procedure for transmitting and receiving serial data simultaneously is as follows:

1. SCI status check and transmit data write: read the serial status register (SSR), check that the TDRE bit is 1, then write transmit data in the transmit data register (TDR) and clear TDRE to 0. The TXI interrupt can also be used to determine if the TDRE bit has changed from 0 to 1.
2. Receive error handling: if a receive error occurs, read the ORER bit in SSR to identify the error. After executing the necessary error handling, clear ORER to 0. Transmitting/receiving cannot resume if ORER remains set to 1.

3. SCI status check and receive data read: read the serial status register (SSR), check that RDRF is set to 1, then read receive data from the receive data register (RDR) and clear RDRF to 0. The RXI interrupt can also be used to determine if the RDRF bit has changed from 0 to 1.
4. To continue transmitting and receiving serial data: read the RDRF bit and RDR, and clear RDRF to 0 before the MSB (bit 7) of the current frame is received. Also read the TDRE bit to check whether it is safe to write (if it reads 1); if so, write data in TDR, then clear TDRE to 0 before the MSB (bit 7) of the current frame is transmitted. When the DMAC is started by a transmit-data-empty interrupt request (TXI) to write data in TDR, the TDRE bit is checked and cleared automatically. When the DMAC is started by a receive-data-full interrupt (RXI) to read RDR, the RDRF bit is cleared automatically.



Note: When switching from transmitting or receiving to simultaneous transmitting and receiving, clear both TE and RE to 0, then set both TE and RE to 1.
Circled numbers refer to the preceding description of the procedure in the text.

Figure 13.20 Sample Flowchart for Serial Transmitting

13.4 SCI Interrupt Sources and the DMAC

The SCI has four interrupt sources in each channel: transmit-end (TEI), receive-error (ERI), receive-data-full (RXI), and transmit-data-empty (TXI). Table 13.13 lists the interrupt sources and indicates their priority. These interrupts can be enabled and disabled by the TIE, RIE, and TEIE bits in the serial control register (SCR). Each interrupt request is sent separately to the interrupt controller.

TXI is requested when the TDRE bit in SSR is set to 1. TXI can start the direct memory access controller (DMAC) to transfer data. TDRE is automatically cleared to 0 when the DMAC writes data in the transmit data register (TDR).

RXI is requested when the RDRF bit in SSR is set to 1. RXI can start the DMAC to transfer data. RDRF is automatically cleared to 0 when the DMAC reads the receive data register (RDR).

ERI is requested when the ORER, PER, or FER bit in SSR is set to 1. ERI cannot start the DMAC.

TEI is requested when the TEND bit in SSR is set to 1. TEI cannot start the DMAC. Where the TXI interrupt indicates that transmit data writing is enabled, the TEI interrupt indicates that the transmit operation is complete.

Table 13.13 SCI Interrupt Sources

Interrupt Source	Description	DMAC Availability	Priority
ERI	Receive error (ORER, PER, or FER)	No	High
RXI	Receive data register full (RDRF)	Yes	↑
TXI	Transmit data register empty (TDRE)	Yes	↓
TEI	Transmit end (TEND)	No	Low

See section 4, Exception Handling, for information on the priority order and relationship to non-SCI interrupts.

13.5 Usage Notes

Note the following points when using the SCI.

TDR Write and TDRE Flag: The TDRE bit in the serial status register (SSR) is a status flag indicating loading of transmit data from TDR into TSR. The SCI sets TDRE to 1 when it transfers data from TDR to TSR. Data can be written to TDR regardless of the TDRE bit status. If new data is written in TDR when TDRE is 0, however, the old data stored in TDR will be lost because the data has not yet been transferred to TSR. Before writing transmit data to TDR, be sure to check that TDRE is set to 1.

Simultaneous Multiple Receive Errors: Table 13.14 indicates the state of the SSR status flags when multiple receive errors occur simultaneously. When an overrun error occurs, the RSR contents cannot be transferred to RDR, so receive data is lost.

Table 13.14 SSR Status Flags and Transfer of Receive Data

	SSR Status Flags				Receive Data Transfer
Receive Error Status	**RDRF**	**ORER**	**FER**	**PER**	**RSR → RDR**
Overrun error	1	1	0	0	X
Framing error	0	0	1	0	O
Parity error	0	0	0	1	O
Overrun error + framing error	1	1	1	0	X
Overrun error + parity error	1	1	0	1	X
Framing error + parity error	0	0	1	1	O
Overrun error + framing error + parity error	1	1	1	1	X

O: Receive data is transferred from RSR to RDR.

X: Receive data is not transferred from RSR to RDR.

Break Detection and Processing: In the break state, the input from the RxD pin consists of all 0s, so FER is set and the parity error flag (PER) may also be set. In the break state, the SCI receiver continues to operate, so if the FER bit is cleared to 0, it will be set to 1 again.

Receive Error Flags and Transmitter Operation (Clocked Synchronous Mode Only): When a receive error flag (ORER, PER, or FER) is set to 1, the SCI will not start transmitting even if TDRE is set to 1. Be sure to clear the receive error flags to 0 before starting to transmit. Note that clearing RE to 0 does not clear the receive error flags.

Receive Data Sampling Timing and Receive Margin in Asynchronous Mode: In asynchronous mode, the SCI operates on a base clock of 16 times the bit rate frequency. In receiving, the SCI synchronizes internally with the falling edge of the start bit, which it samples on the base clock. Receive data is latched on the rising edge of the eighth base clock pulse. See figure 13.21.

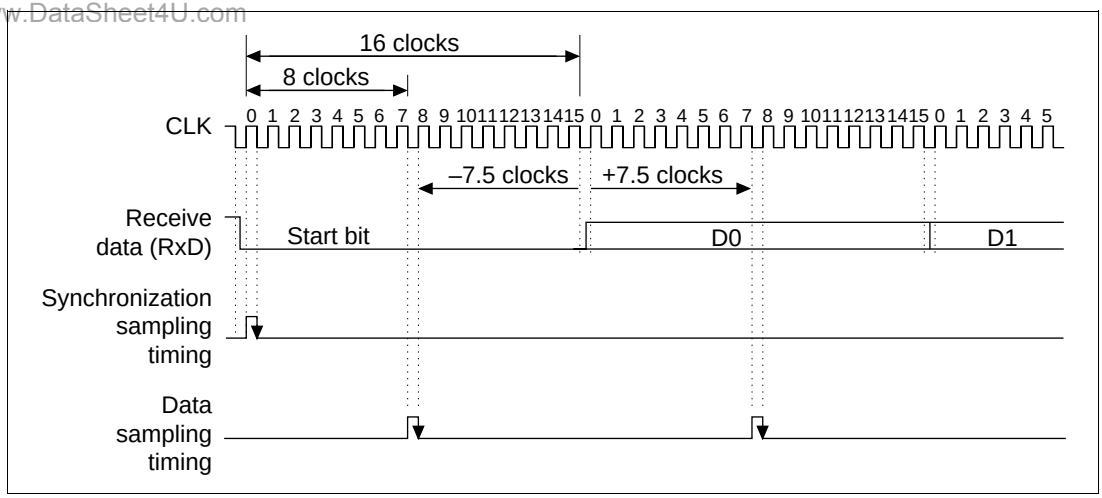


Figure 13.21 Receive Data Sampling Timing in Asynchronous Mode

The receive margin in asynchronous mode can therefore be expressed as in equation 1.

Equation 1:

$$M = \left| \left(0.5 - \frac{1}{2N} - 1 \right) - (L - 0.5)F - \frac{|D - 0.5|}{N}(1 + F) \right| \times 100\%$$

M : Receive margin (%)
N : Ratio of clock frequency to bit rate (N = 16)
D : Clock duty cycle (D = 0–1.0)
L : Frame length (L = 9–12)
F : Absolute deviation of clock frequency

From equation (1), if F = 0 and D = 0.5 the receive margin is 46.875%, as given by equation 2.

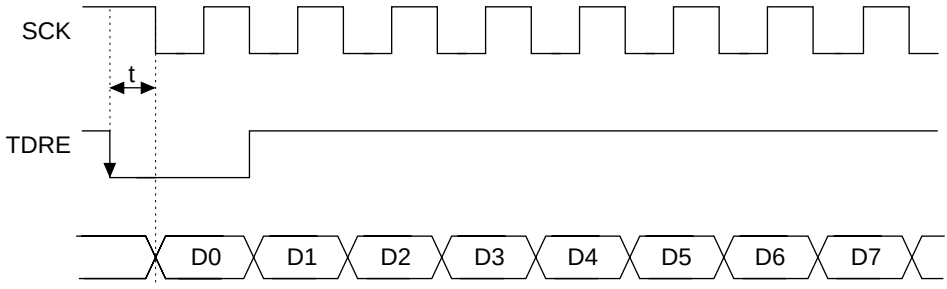
Equation 2:

$D = 0.5, F = 0$
$M = (0.5 - 1/(2 \times 16)) \times 100\%$
$= 46.875\%$

This is a theoretical value. A reasonable margin to allow in system designs is 20–30%.

Constraints on DMAC Use:

- When using an external clock source for the serial clock, update TDR with the DMAC, and then after twenty system clock cycles or more elapse, input a transmit clock. If a transmit clock is input in the first four states after TDR is written, an error may occur (figure 13.22).
- Before reading the receive data register (RDR) with the DMAC, select the receive-data-full interrupt of the SCI as an activation source using the resource select bit (RS) in the channel control register (CHCR).



Note: During external clock operation, an error may occur if t is 4 states or less.

Figure 13.22 Example of Clocked Synchronous Transmission with DMAC

Cautions for Clocked Synchronous External Clock Mode:

- Set TE = RE = 1 only when external clock SCK is 1.
- Do not set TE = RE = 1 until at least four clock cycles after external clock SCK has changed from 0 to 1.
- When receiving, RDRF is set to 1 when RE is cleared to 0 2.5–3.5 clocks after the rising edge of the RxD D7 bit SCK input, but it cannot be copied to RDR.

Caution for Clocked Synchronous Internal Clock Mode: When receiving, RDRF is set to 1 when RE is cleared to 0 1.5 clocks after the rising edge of the RxD D7 bit SCK output, but it cannot be copied to RDR.

Section 14 Power-Down Modes

14.1 Overview

The SH7604 has a module standby function (which selectively halts operation of some on-chip peripheral modules), a sleep mode (which halts CPU function), and a standby mode (which halts all functions).

14.1.1 Power-Down Modes

In addition to the sleep mode and standby mode, the SH7604 also has a third power-down mode, the module standby function, which halts the DMAC, multiplication unit, division unit, free-running timer, and SCI on-chip peripheral modules.

Table 14.1 shows the transition conditions for entering the modes from the program execution state, as well as the CPU and peripheral module states in each mode and the procedures for canceling each mode.

Table 14.1 Power-Down Modes

Mode	Transition Condition	State: Clock	State: CPU, MULT, Cache	State: UBC, BSC	State: FRT, SCI, DMAC, DIV, INTC, WDT,	State: Pins	Canceling Procedure
Sleep mode	SLEEP instruction executed with SBY bit set to 0 in SBYCR	Runs	Halted	Runs	Runs	Runs	1. Interrupt 2. DMA address error 3. Power-on reset 4. Manual reset
Standby mode	SLEEP instruction executed with SBY bit set to 1 in SBYCR	Halted	Halted	Held	Halted	Held or high impedance	1. NMI interrupt 2. Power-on reset 3. Manual reset
Module standby function	MSTP bit for relevant module is set to 1	Runs	Run (MULT is held)	Runs	When MSTP bit is 1, the supply of the clock to the relevant module is halted.	FRT and SCI pins are initialized, and others operate.	Clear MSTP bit to 0

14.1.2 Register

Table 14.2 shows the register configuration.

Table 14.2 Register Configuration

Name	Abbreviation	R/W	Initial Value	Address
Standby control register	SBYCR	R/W	H'60	H'FFFFFE91

14.2 Description of Register

14.2.1 Standby Control Register (SBYCR)

The standby control register (SBYCR) is an 8-bit read/write register that sets the power-down mode. SBYCR is initialized to H'00 by a reset.

Bit:	7	6	5	4	3	2	1	0
Bit name:	SBY	HIZ	—	MSTP4	MSTP3	MSTP2	MSTP1	MSTP0
Initial value:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	—	R/W	R/W	R/W	R/W	R/W

- Bit 7—Standby (SBY): Specifies transition to standby mode. The SBY bit cannot be set to 1 while the watchdog timer is running (when the TME bit in the WDt's WTCSR register is 1). To enter the standby mode, halt the WDT (set the TME bit in WTCSR to 0) and set the SBY bit.

Bit 7: SBY	Description	
0	Executing a SLEEP instruction puts the chip into sleep mode	(Initial value)
1	Executing a SLEEP instruction puts the chip into standby mode	

- Bit 6—Port High Impedance (HIZ): Selects whether output pins are set to high impedance or retain the output state in standby mode. When HIZ = 0 (initial state), the specified pin retains its output state. When HIZ = 1, the pin goes to the high-impedance state. See Appendix A.1, Pin States during Resets, Power-Down States and Bus Release State, for which pins are controlled.

Bit 6: HIZ	Description	
0	Pin state retained in standby mode	(Initial value)
1	Pin goes to high impedance in standby mode	

- Bit 5—Reserved: This bit always reads 0. The write value should always be 0.

- Bit 4: Module stop 4 (MSTP4): Specifies halting the clock supply to the DMAC. When MSTP4 bit is set to 1, the supply of the clock to the DMAC is halted. When the clock halts, the DMAC retains its pre-halt state. When MSTP4 is cleared to 0 and the DMAC begins running again, its starts operating from its pre-halt state. Set this bit while the DMAC is halted; this bit cannot be set while the DMAC is operating (transferring data).

Bit 4: MSTP4	Description	
0	DMAC running	(Initial value)
1	Clock supply to DMAC halted	

- Bit 3—Module Stop 3 (MSTP3): Specifies halting the clock supply to the multiplication unit (MULT). When the MSTP3 bit is set to 1, the supply of the clock to MULT is halted. When the clock halts, MULT retains its pre-halt state. This bit should be set when the MULT is halted.

Bit 3: MSTP3	Description	
0	MULT running	(Initial value)
1	Clock supply to MULT halted	

- Bit 2—Module Stop 2 (MSTP2): Specifies halting the clock supply to the division unit (DIVU). When the MSTP2 bit is set to 1, the supply of the clock to DIVU is halted. When the clock halts, the DIVU registers retain their pre-halt state. This bit should be set when the DIVU is halted.

Bit 2: MSTP2	Description	
0	DIVU running	(Initial value)
1	Clock supply to DIVU halted	

- Bit 1—Module Stop 1 (MSTP1): Specifies halting the clock supply to the 16-bit free-running timer (FRT). When the MSTP1 bit is set to 1, the supply of the clock to the FRT is halted. When the clock halts, all FRT registers are initialized except the FRT interrupt vector register in INTC, which holds its previous value. When MSTP1 is cleared to 0 and the FRT begins running again, its starts operating from its initial state.

Bit 1: MSTP1	Description	
0	FRT running	(Initial value)
1	Clock supply to FRT halted	

- Bit 0—Module Stop 0 (MSTP0): Specifies halting the clock supply to the serial communication interface (SCI). When the MSTP0 bit is set to 1, the supply of the clock to the SCI is halted. When the clock halts, all SCI registers are initialized except the SCI interrupt vector register in INTC, which holds its previous value. When MSTP0 is cleared to 0 and the SCI begins running again, its starts operating from its initial state.

Bit 0: MSTP0	Description	
0	SCI running	(Initial value)
1	Clock supply to SCI halted	

14.3 Sleep Mode

14.3.1 Transition to Sleep Mode

Executing the SLEEP instruction when the SBY bit in SBYCR is 0 causes a transition from the program execution state to sleep mode. Although the CPU halts immediately after executing the SLEEP instruction, the contents of its internal registers remain unchanged. The on-chip peripheral modules continue to run in sleep mode.

14.3.2 Canceling Sleep Mode

Sleep mode is canceled by an interrupt, DMA address error, power-on reset, or manual reset.

Cancellation by an Interrupt: When an interrupt occurs, sleep mode is canceled and interrupt exception handling is executed. Sleep mode is not canceled if the interrupt cannot be accepted because its priority level is equal to or less than the mask level set in the CPU's status register (SR) or if an interrupt by an on-chip peripheral module is disabled at the peripheral module.

Cancellation by a DMA Address Error: If a DMA address error occurs, sleep mode is canceled and DMA address error exception handling is executed.

Cancellation by a Power-On Reset: A power-on reset cancels sleep mode.

Cancellation by a Manual Reset: A manual reset cancels sleep mode.

14.4 Standby Mode

14.4.1 Transition to Standby Mode

To enter standby mode, set the SBY bit to 1 in SBYCR, then execute the SLEEP instruction. The chip switches from the program execution state to standby mode. The NMI interrupt cannot be accepted when the SLEEP instruction is executed, or for the following five cycles. In standby mode, power consumption is greatly reduced by halting not only the CPU, but the clock and on-chip peripheral modules as well. CPU register contents are held, and some on-chip peripheral modules are initialized.

Table 14.3 Register States in Standby Mode

Module	Registers Initialized	Registers that Retain Data	Registers with Undefined Contents
Interrupt controller (INTC)	—	All registers	—
User break controller (UBC)	—	All registers	—
Bus state controller (BSC)	—	All registers	—
DMAC	DMA channel control register 0 DMA channel control register 1 DMA operation register	All registers except DMA channel control register 0, DMA channel control register 1, and DMA operation register	—
DIVU	—	—	All registers
Watchdog timer (WDT)	Bits 7–5 of the timer control/status register Reset control/status register	Bits 2–0 of the timer control/status register Timer counter	—
16-bit free-running timer (FRT)	All registers	—	—
Serial communication interface (SCI)	All registers	—	—
Others	—	Standby control register Frequency modification register	—

14.4.2 Canceling Standby Mode

Standby mode is canceled by an NMI interrupt, a power-on reset, or a manual reset.

Cancellation by an NMI: When a rising edge or falling edge is detected in the NMI signal, after the elapse of the time set in the WDT timer control/status register, clocks are supplied to the entire chip, standby mode is canceled, and NMI exception handling begins.

Cancellation by a Power-On Reset: A power-on reset cancels standby mode.

Cancellation by a Manual Reset: A manual reset cancels standby mode.

14.4.3 Standby Mode Cancellation by NMI

The following example describes moving to the standby mode upon the fall of the NMI signal and clearing the standby when the NMI signal rises. Figure 14.1 shows the timing.

When the NMI pin level changes from high to low after the NMI edge select bit (NMIE) of the interrupt control register (ICR) has been set to 0 (detect falling edge), an NMI interrupt is accepted. When the NMIE bit is set to 1 (detect rising edge) by the NMI exception service routine, the standby bit (SBY) of the standby control register (SBYCR) is set to 1 and a SLEEP instruction is executed, the standby mode is entered. The standby mode is cleared the next time the NMI pin level changes from low level to high level.

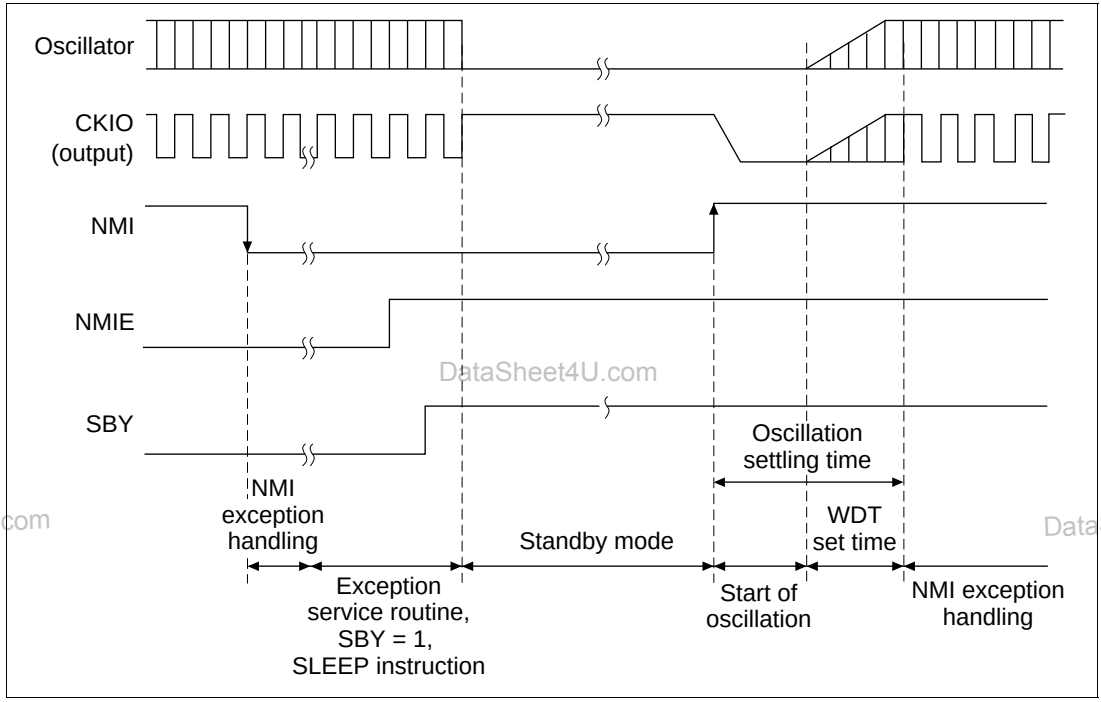


Figure 14.1 Standby Mode Cancellation by NMI

14.4.4 Clock Pause Function

When the clock is input from the CKIO pin, the clock frequency can be modified or the clock stopped. The SH7604 has a $\overline{\text{CKPREQ}}$/CKM pin for this purpose. The clock pause function is used as described below. Note that clock pauses are not accepted while the watchdog timer (WDT) is operating (i.e. when the timer enable bit (TME) in the WDT's timer control/status register (WTCSR) is 1).

1. Set the TME bit in the watchdog timer's WTCSR register to 0.
2. Set the overflow time in bits CKS2 to CKS0 bits in the watchdog timer's WTCSR register (overflow time should be calculated using the clock frequency after modification).
3. After the SLEEP instruction is executed and standby mode is entered, apply a low level from the $\overline{\text{CKPREQ}}$/CKM pin.
4. When the chip is internally ready to modify the operating clock, a low level is output from the $\overline{\text{CKPACK}}$ pin.
5. After the $\overline{\text{CKPACK}}$ pin goes low, the clocks are stopped and the frequency is modified. The internal chip state is the same as in standby mode.
6. When the clock pause state (standby) is canceled, the WDT starts to count up at the falling edge or rising edge of the NMI pin (when the NMIE bit of INTC is set).
7. When a frequency is modified, the $\overline{\text{CKPACK}}$ pin goes high after the time set by the WDT, and the clock pause function gives external notification that the chip can again be operated (standby mode is canceled).
8. When a clock is halted, the clock is applied again to the CKIO pin and NMI input is generated. After the time set by the WDT, the $\overline{\text{CKPACK}}$ pin goes high, and the clock pause function gives external notification that the chip can again be operated (standby mode is canceled).

The standby state, all internal functions and all pin states during clock pause are equivalent to those of the normal standby mode. Figure 14.2 shows the timing chart for the clock pause function.

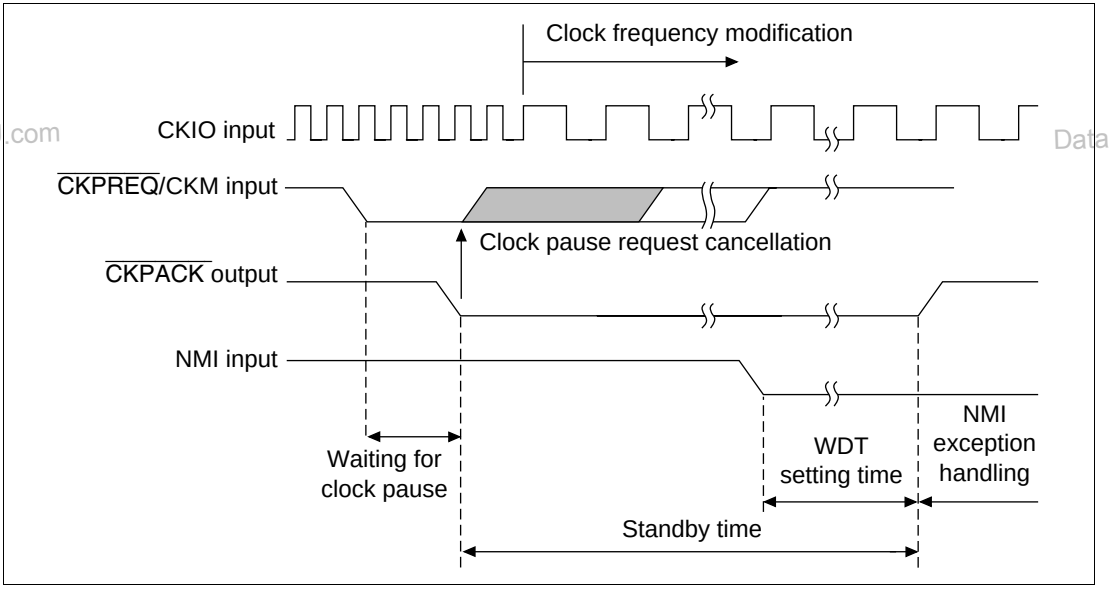


Figure 14.2 Clock Pause Function Timing

14.4.5 Notes on Standby Mode

1. When the SH7604 enters standby mode during use of the cache, disable the cache before making the mode transition. Initialize the cache beforehand when the cache is used after returning to standby mode. The contents of the on-chip RAM are not retained in standby mode when cache is used as on-chip RAM.
2. If an on-chip peripheral register is written in the 10 clock cycles before the SH7604 transits to standby mode, read the register before executing the SLEEP instruction.
3. When using clock mode 0, 1, or 2, the CKIO pin is the clock output pin. Note the following when standby mode is used in these clock modes. When standby mode is canceled by NMI, an unstable clock is output from the CKIO pin during the oscillation settling time after NMI input. This also applies to clock output in the case of cancellation by a power-on reset or manual reset. Power-on reset and manual reset input should be continued for a period at least equal to for the oscillation settling time.

14.5 Module Standby Function

14.5.1 Transition to Module Standby Function

By setting one of standby control register bits MSTP4–MSTP0 to 1, the supply of the clock to the corresponding on-chip peripheral module can be halted. This function can be used to reduce the power consumption in sleep mode. Do not perform read/write operations for a module in module standby mode.

The external pins and registers of the DMAC, MULT, and DIVU on-chip peripheral modules retain their states prior to halting. The external pins of the FRT and SCI are reset and all their registers are initialized.

Do not switch on-chip peripheral modules to module standby mode while they are running.

14.5.2 Clearing the Module Standby Function

Clear the module standby function by clearing the MSTP4–MSTP0 bits, or by a power-on reset or manual reset.

To effect a module stop, halt the relevant module or disable interrupts.

Section 15 Electrical Characteristics (5V Version)

15.1 Absolute Maximum Ratings

Table 15.1 shows the absolute maximum ratings.

Table 15.1 Absolute Maximum Ratings

Item	Symbol	Rating	Unit
Power supply voltage	V_{CC}	–0.3 to +7.0	V
Input voltage	Vin	–0.3 to V_{CC} + 0.3	V
Operating temperature	Topr	–20 to +75	°C
Storage temperature	Tstg	–55 to +125	°C

Caution: Operating the chip in excess of the absolute maximum rating may result in permanent damage.

15.2 DC Characteristics

Tables 15.2 and 15.3 list DC characteristics.

Table 15.2 DC Characteristics (Conditions: V_{CC} = 5.0 V ± 10%, Ta = –20 to +75° C)

Item		Symbol	Min	Typ	Max	Unit	Test Conditions
Input high-level voltage	$\overline{RES}$, NMI, MD5–MD0	V_{IH}	V_{CC} – 0.5	—	V_{CC} + 0.3	V	During standby
			V_{CC} – 0.7	—	V_{CC} + 0.3	V	Normal operation
	EXTAL, CKIO		V_{CC} – 0.7	—	V_{CC} + 0.3	V	
	Other input pins		2.2	—	V_{CC} + 0.3	V	
Input low-level voltage	$\overline{RES}$, NMI, MD5–MD0	V_{IL}	–0.3	—	0.5	V	During standby
			–0.3	—	0.8	V	Normal operation
	Other input pins		–0.3	—	0.8	V	
Input leak current	$\overline{RES}$	|Iin|	—	—	1.0	µA	Vin = 0.5 to V_{CC} – 0.5 V
	NMI, MD5–MD0		—	—	1.0	µA	Vin = 0.5 to V_{CC} – 0.5 V
	Other input pins		—	—	1.0	µA	Vin = 0.5 to V_{CC} – 0.5 V
3-state leak current (while off)	A26–A0, D31–D0, $\overline{BS}$, $\overline{CS3}$–$\overline{CS0}$, RD/$\overline{WR}$, $\overline{RAS}$, $\overline{CAS}$, $\overline{WE3}$–$\overline{WE0}$, $\overline{RD}$, $\overline{IVECF}$	$\|I_{STI}\|$	—	—	1.0	µA	Vin = 0.5 to V_{CC} – 0.5 V
Output high-level voltage	All output pins	V_{OH}	V_{CC} – 0.5	—	—	V	I_{OH} = –200 µA
			3.5	—	—	V	I_{OH} = –1 mA
Output low-level voltage	All output pins	V_{OL}	—	—	0.4	V	I_{OL} = 1.6 mA
Input capacitance	$\overline{RES}$	Cin	—	—	15	pF	Vin = 0 V f = 1 MHz Ta = 25°C
	NMI		—	—	15	pF	
	All other input pins (including D31–D0)		—	—	15	pF	

Table 15.2 DC Characteristics (Conditions: V_{CC} = 5.0 V ± 10%, Ta = –20 to +75° C) (cont)

Item		Symbol	Min	Typ	Max	Unit	Test Conditions
Current consump-tion	Normal operation	I_{CC}	—	60	80	mA	f = 8 MHz
			—	80	100	mA	f = 16 MHz
			—	110	160	mA	f = 28.7 MHz
	Sleep		—	30	55	mA	f = 8 MHz
			—	50	70	mA	f = 16 MHz
			—	80	100	mA	f = 28.7 MHz
	Standby		—	1	15	µA	Ta ≤ 50°C
			—	—	60	µA	50°C < Ta

Notes: 1. When no PLL is used, do not leave the $PLLV_{CC}$ and $PLLV_{SS}$ pins open. Connect $PLLV_{CC}$ to V_{CC} and $PLLV_{SS}$ to V_{SS}.
2. Current consumption values shown are the values at which all output pins are without load under conditions of V_{IH} min = V_{CC} – 0.5 V, V_{IL} max = 0.5 V.

Table 15.3 Permitted Output Current Values (Conditions: V_{CC} = 5.0 V ± 10%, Ta = –20 to +75°C)

Item	Symbol	Min	Typ	Max	Unit
Output low-level permissible current (per pin)	I_{OL}	—	—	2.0	mA
Output low-level permissible current (total)	ΣI_{OL}	—	—	80	mA
Output high-level permissible current (per pin)	$-I_{OH}$	—	—	2.0	mA
Output high-level permissible current (total)	$\Sigma(-I_{OH})$	—	—	25	mA

Caution: To ensure chip reliability, do not exceed the output current values given in table 15.3.

15.3 AC Characteristics

15.3.1 Clock Timing

Table 15.4 Clock Timing (Conditions: V_{CC} = 5.0 V ± 10%, Ta = –20 to +75°C)

Item	Symbol	Min	Max	Unit	Figures
Operating frequency	f_{OP}	4	28.7	MHz	15.1
Clock cycle time	t_{cyc}	35	143*1 or 250*2	ns	
Clock high pulse width	t_{CH}	8*1 or 15*2	—	ns	
Clock low pulse width	t_{CL}	8*1 or 15*2	—	ns	
Clock rise time	t_{CR}	—	5	ns	
Clock fall time	t_{CF}	—	5	ns	
EXTAL clock input frequency	f_{EX}	4	8	MHz	15.2
EXTAL clock input cycle time	t_{EXcyc}	125	250	ns	
EXTAL clock input low-level pulse width	t_{EXL}	50	—	ns	
EXTAL clock input high-level pulse width	t_{EXH}	50	—	ns	
EXTAL clock input rise time	t_{EXR}	—	5	ns	
EXTAL clock input clock fall time	t_{EXF}	—	5	ns	
Power-on oscillation settling time	t_{OSC1}	10	—	ms	15.3
Software standby oscillation settling time 1	t_{OSC2}	10	—	ms	15.4
Software standby oscillation settling time 2	t_{OSC3}	10	—	ms	15.5
PLL synchronization settling time	t_{PLL}	1	—	µs	15.6

Notes: 1. With PLL circuit 1 operating.
2. With PLL circuit 1 not used.

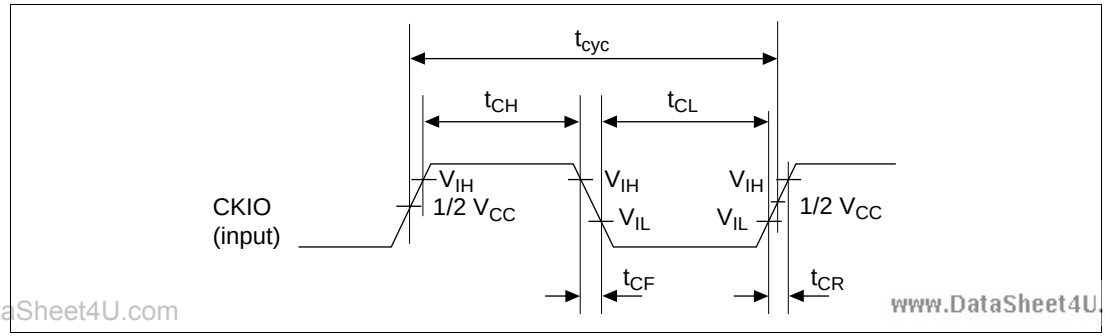
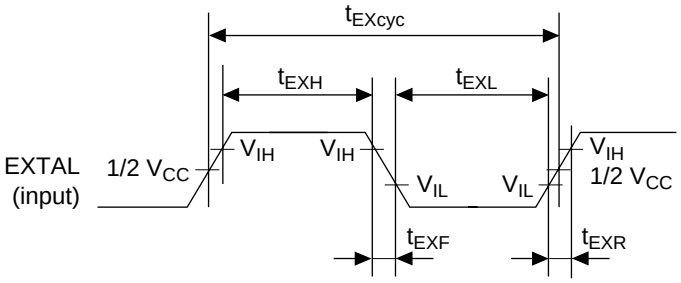
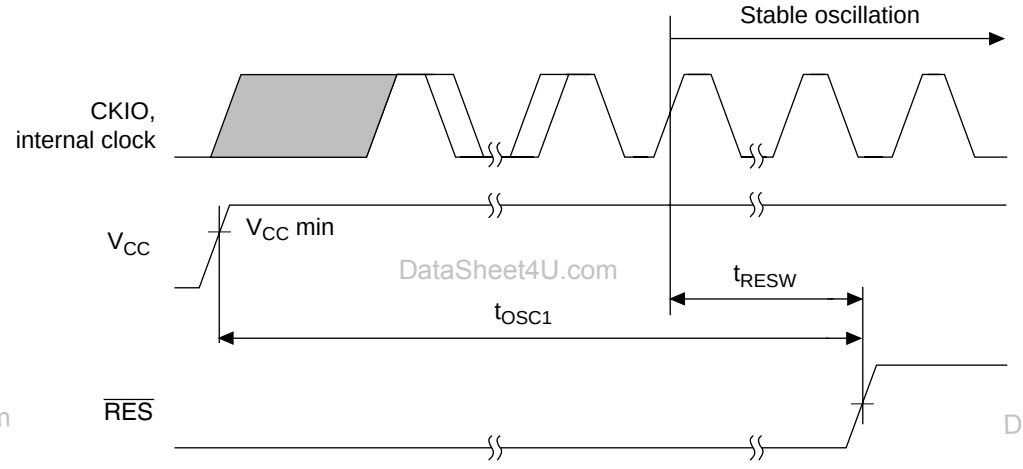


Figure 15.1 CKIO Input Timing



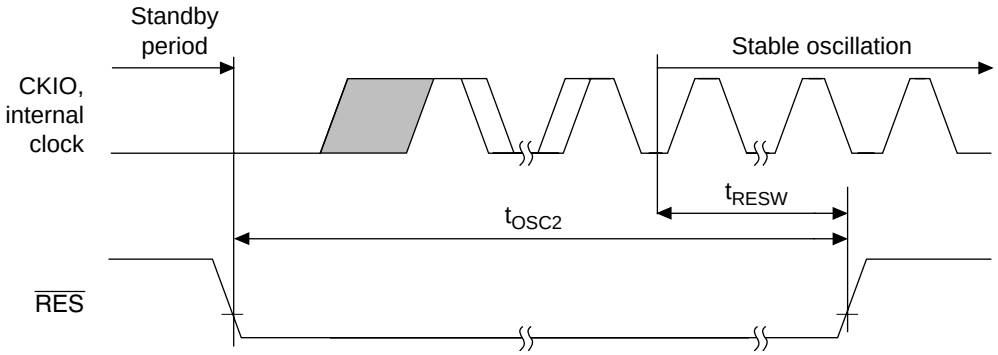
Note: External clock input from EXTAL pin.

Figure 15.2 EXTAL Clock Input Timing



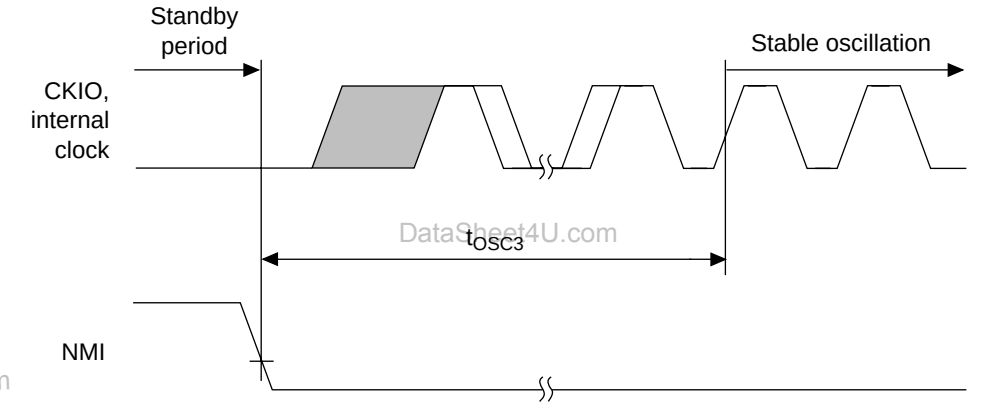
Note: Oscillation settling time when on-chip crystal oscillator is used.

Figure 15.3 Oscillation Settling Time at Power-On



Note: Oscillation settling time when on-chip crystal oscillator is used.

Figure 15.4 Oscillation Settling Time at Standby Return (via $\overline{RES}$)



Note: Oscillation settling time when on-chip crystal oscillator is used.

Figure 15.5 Oscillation Settling Time at Standby Return (via NMI)

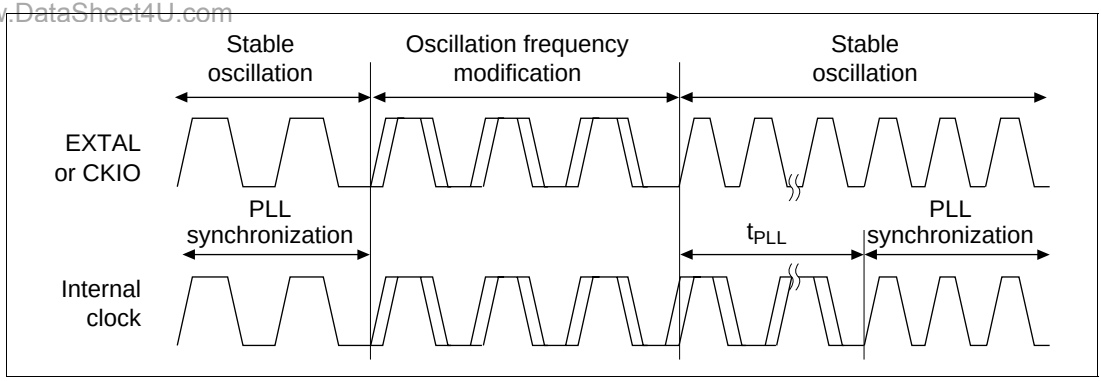


Figure 15.6 PLL Synchronization Settling Time

15.3.2 Control Signal Timing

Table 15.5 Control Signal Timing (Conditions: V_{CC} = 5.0 V ± 10%, Ta = –20 to +75°C)

Item	Symbol	Min	Max	Unit	Figure
$\overline{RES}$ rise, fall	t_{RESr}, t_{RESf}	—	200	ns	15.7
$\overline{RES}$ pulse width	t_{RESW}	20	—	t_{cyc}	
NMI reset setup time	t_{NMIRS}	tcyc + 10	—	ns	
NMI reset hold time	t_{NMIRH}	tcyc + 10	—	ns	
NMI rise, fall	t_{NMIr}, t_{NMIf}	—	200	ns	
NMI minimum pulse width	t_{IRQES}	3	—	tcyc	
$\overline{RES}$ setup time*	t_{RESS}	30	—	ns	15.8, 15.9
NMI setup time*	t_{NMIS}	30	—	ns	
$\overline{IRL3}$–$\overline{IRL0}$ setup time*	t_{IRLS}	30	—	ns	
$\overline{RES}$ hold time	t_{RESH}	10	—	ns	15.8, 15.9
NMI hold time	t_{NMIH}	10	—	ns	
$\overline{IRL3}$–$\overline{IRL0}$ hold time	t_{IRLH}	10	—	ns	
$\overline{BRLS}$ setup time 1 (PLL on)	t_{BLSS1}	1/2 tcyc + 9	—	ns	15.10
$\overline{BRLS}$ hold time 1 (PLL on)	t_{BLSH1}	9 – 1/2 tcyc	—	ns	
$\overline{BGR}$ delay time 1 (PLL on)	t_{BGRD1}	—	1/2 tcyc + 18	ns	
$\overline{BRLS}$ setup time 1 (PLL on, 1/4 cycle delay)	t_{BLSS1}	1/4 tcyc + 9	—	ns	15.10
$\overline{BRLS}$ hold time 1 (PLL on, 1/4 cycle delay)	t_{BLSH1}	9 – 1/4 tcyc	—	ns	
$\overline{BGR}$ delay time 1 (PLL on, 1/4 cycle delay)	t_{BGRD1}	—	3/4 tcyc + 18	ns	
$\overline{BRLS}$ setup time 2 (PLL off)	t_{BLSS2}	9	—	ns	15.11
$\overline{BRLS}$ hold time 2 (PLL off)	t_{BLSH2}	19	—	ns	
$\overline{BGR}$ delay time 2 (PLL off)	t_{BGRD2}	—	28	ns	

Note: The $\overline{RES}$, NMI and $\overline{IRL3}$-$\overline{IRL0}$ signals are asynchronous inputs, but when the setup times shown here are observed, the signals are considered to have changed at clock fall. If the setup times are not observed, recognition may be delayed until the next clock fall.

Table 15.5 Control Signal Timing (Conditions: V_{CC} = 5.0 V ± 10%, Ta = –20 to +75°C) (cont)

Item	Symbol	Min	Max	Unit	Figure
$\overline{\text{BREQ}}$ delay time 1 (PLL on)	t_{BRQD1}	—	1/2 tcyc + 18	ns	15.12
$\overline{\text{BACK}}$ setup time 1 (PLL on)	t_{BAKS1}	1/2 tcyc + 9	—	ns	
$\overline{\text{BACK}}$ hold time 1 (PLL on)	t_{BAKH1}	9 – 1/2 tcyc	—	ns	
$\overline{\text{BREQ}}$ delay time 1 (PLL on, 1/4 cycle delay)	t_{BRQD1}	—	3/4 tcyc + 18	ns	15.12
$\overline{\text{BACK}}$ setup time 1 (PLL on, 1/4 cycle delay)	t_{BAKS1}	1/4 tcyc + 9	—	ns	
$\overline{\text{BACK}}$ hold time 1 (PLL on, 1/4 cycle delay)	t_{BAKH1}	9 – 1/4 tcyc	—	ns	
$\overline{\text{BREQ}}$ delay time 2 (PLL off)	t_{BRQD2}	—	28	ns	15.13
$\overline{\text{BACK}}$ setup time 2 (PLL off)	t_{BAKS2}	9	—	ns	
$\overline{\text{BACK}}$ hold time 2 (PLL off)	t_{BAKH2}	19	—	ns	
Bus tri-state delay time 1 (PLL on)	t_{BOFF1}	0	25	ns	15.10, 15.12
Bus buffer on time 1 (PLL on)	t_{BON1}	0	18	ns	
Bus tri-state delay time 1 (PLL on, 1/4 cycle delay)	t_{BOFF1}	1/4 tcyc	1/4 tcyc + 25	ns	15.10, 15.12
Bus buffer on time 1 (PLL on, 1/4 cycle delay)	t_{BON1}	1/4 tcyc	1/4 tcyc + 18	ns	
Bus tri-state delay time 1 (PLL off)	t_{BOFF1}	0	30	ns	15.11, 15.13
Bus buffer on time 1 (PLL off)	t_{BON1}	0	25	ns	
Bus tri-state delay time 2 (PLL on)	t_{BOFF2}	1/2 tcyc	1/2 tcyc + 25	ns	15.10, 15.12
Bus buffer on time 2 (PLL on)	t_{BON2}	1/2 tcyc	1/2 tcyc + 18	ns	
Bus tri-state delay time 2 (PLL on, 1/4 cycle delay)	t_{BOFF2}	3/4 tcyc	3/4 tcyc + 25	ns	15.10, 15.12
Bus buffer on time 2 (PLL on, 1/4 cycle delay)	t_{BON2}	3/4 tcyc	3/4 tcyc + 18	ns	
Bus tri-state delay time 3 (PLL off)	t_{BOFF3}	0	30	ns	15.11, 15.13
Bus buffer on time 3 (PLL off)	t_{BON3}	0	25	ns	

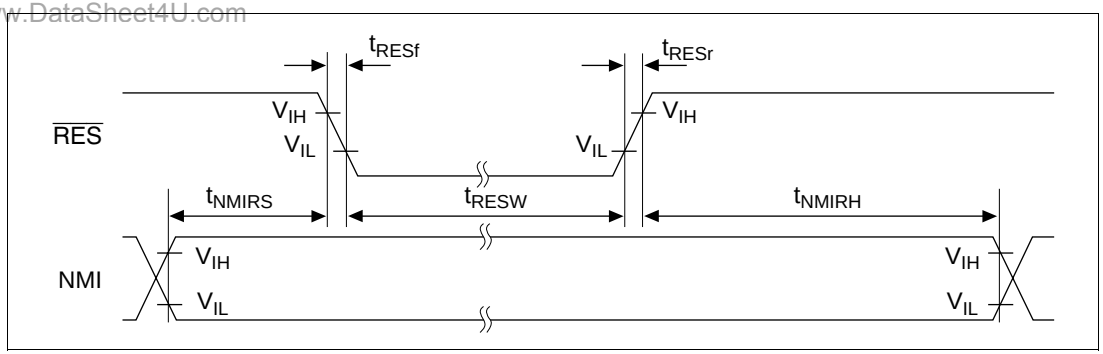


Figure 15.7 Reset Input Timing

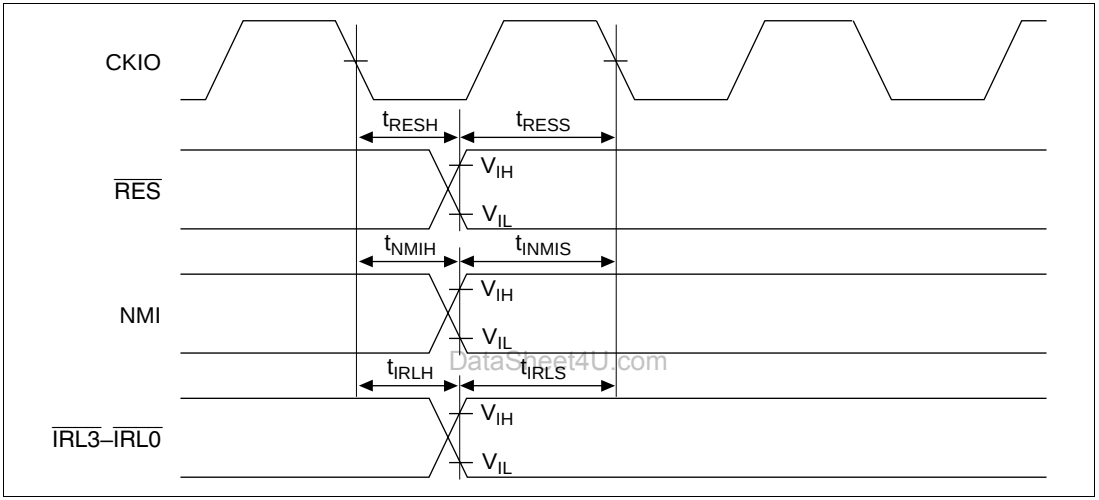


Figure 15.8 Interrupt Signal Input Timing (With PLL1 Off)

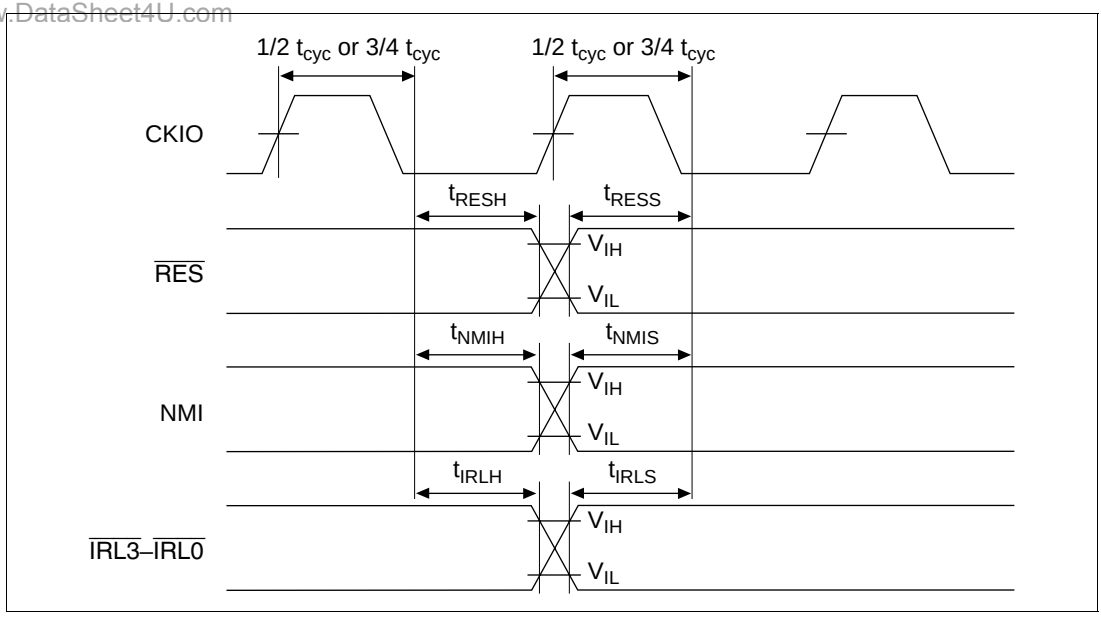


Figure 15.9 Interrupt Signal Input Timing (PLL1 On)

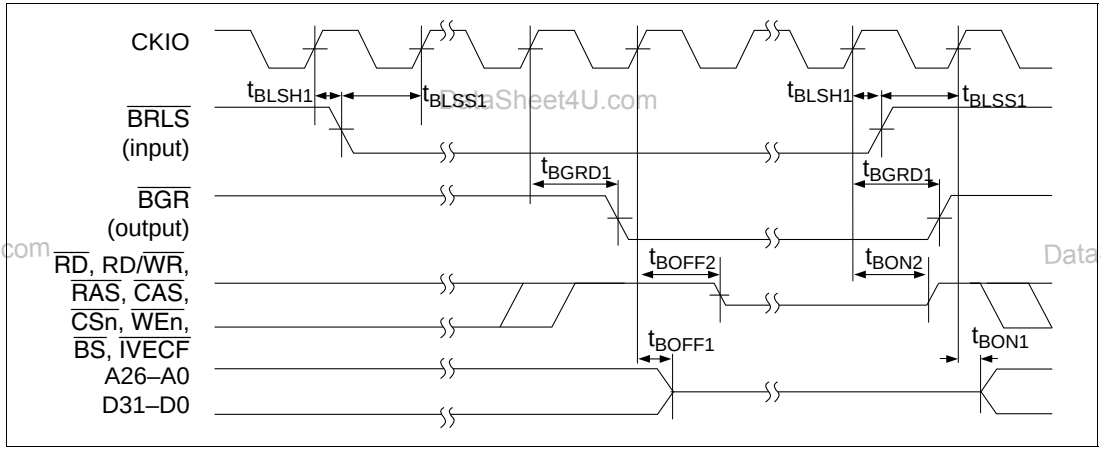


Figure 15.10 Bus Release Timing (Master Mode, PLL1 On)

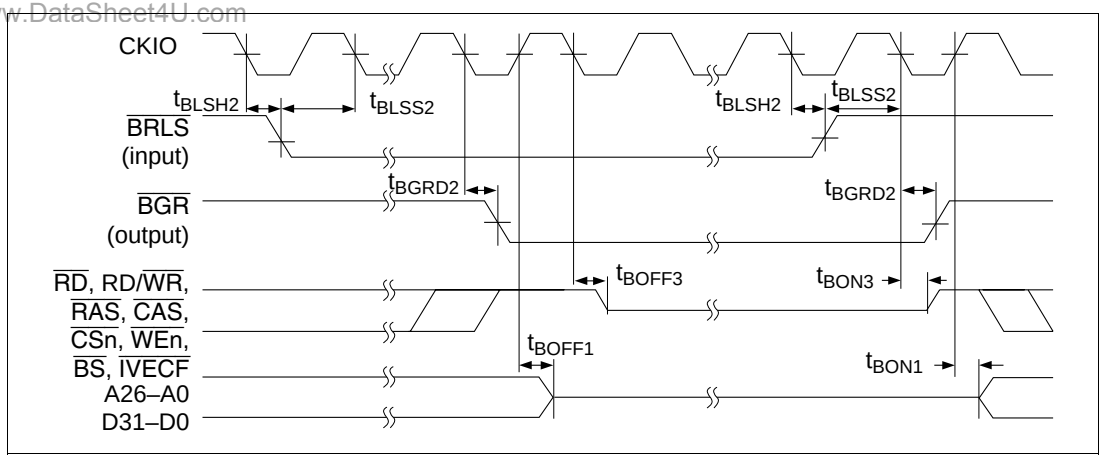


Figure 15.11 Bus Release Timing (Master Mode, PLL1 Off)

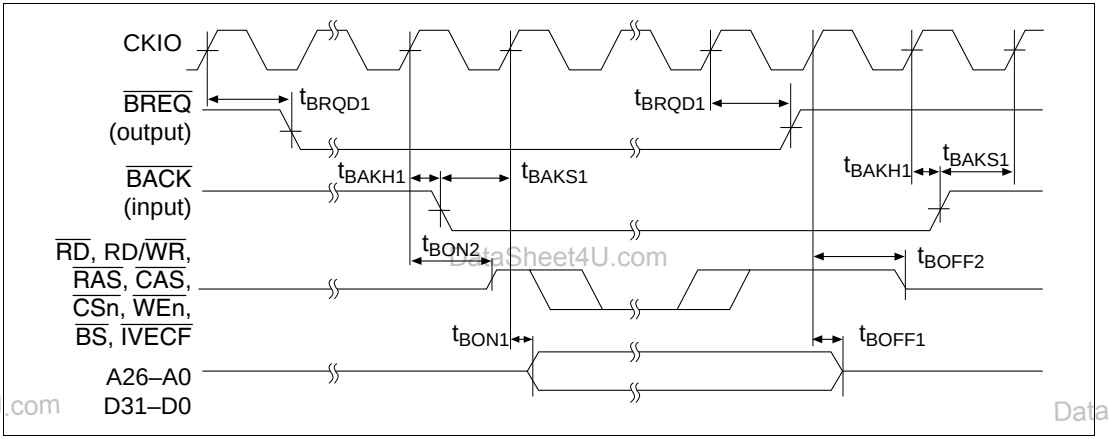


Figure 15.12 Bus Release Timing (Slave Mode, PLL1 On)

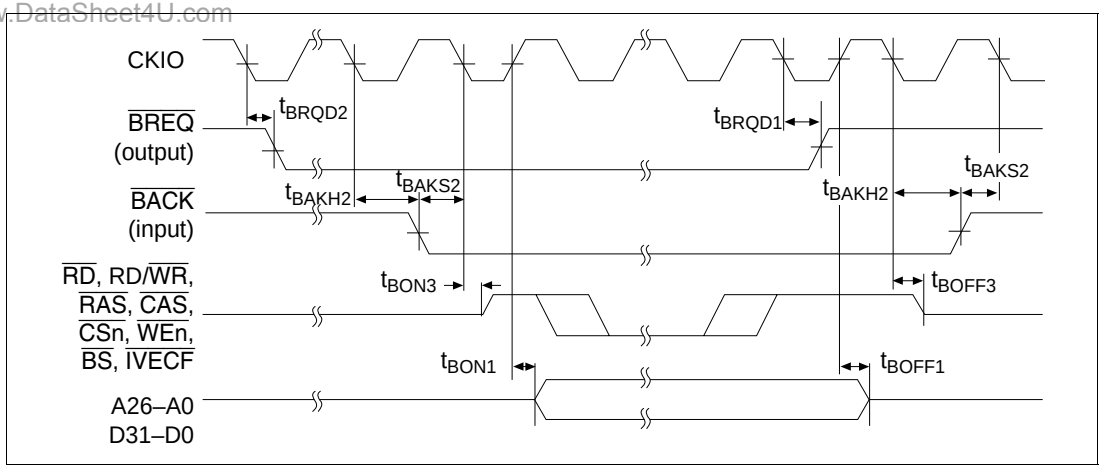


Figure 15.13 Bus Release Timing (Slave Mode, PLL1 Off)

15.3.3 Bus Timing

Table 15.6 Bus Timing With PLL On [Mode 0, 4] (Conditions: V_{CC} = 5.0 V ±10%, Ta = –20 to +75°C)

Item	Symbol	Min	Max	Unit	Figures
Address delay time	t_{AD}	3	18	ns	15.14, 15.20, 15.40, 15.52, 15.66, 15.68
$\overline{BS}$ delay time	t_{BSD}	—	21	ns	15.14, 15.20, 15.40, 15.52, 15.66
$\overline{CS}$ delay time 1	t_{CSD1}	—	21	ns	15.14, 15.20, 15.40, 15.52, 15.66
$\overline{CS}$ delay time 2	t_{CSD2}	—	1/2 tcyc + 21	ns	15.14, 15.66
Read/write delay time	t_{RWD}	3	18	ns	15.14, 15.20, 15.40, 15.52, 15.66
Read strobe delay time 1	t_{RSD1}	—	1/2 tcyc + 16	ns	15.14, 15.40, 15.52, 15.66, 15.68
Read data setup time 1	t_{RDS1}	1/2 tcyc + 10	—	ns	15.14, 15.40, 15.52, 15.66, 15.68
Read data setup time 3 (SDRAM)	t_{RDS3}	1/2 tcyc + 8	—	ns	15.20
Read data hold time 2	t_{RDH2}	0	—	ns	15.14, 15.66
Read data hold time 4 (SDRAM)	t_{RDH4}	0	—	ns	15.20
Read data hold time 5 (DRAM)	t_{RDH5}	0	—	ns	15.40
Read data hold time 6 (PSRAM)	t_{RDH6}	0	—	ns	15.52
Read data hold time 7 (interrupt vector)	t_{RDH7}	0	—	ns	15.68
Write enable delay time	t_{WED1}	1/2 tcyc + 3	1/2 tcyc + 18	ns	15.14, 15.15, 15.52, 15.53
Write data delay time 1	t_{WDD}	3	18	ns	15.15, 15.27, 15.41, 15.53
Write data hold time 1	t_{WDH1}	3	—	ns	15.15, 15.27, 15.41, 15.53
Data buffer on time	t_{DON}	—	18	ns	15.15, 15.27, 15.41, 15.53
Data buffer off time	t_{DOF}	—	18	ns	15.15, 15.27, 15.41, 15.53

Table 15.6 Bus Timing With PLL On [Mode 0, 4] (cont)
(Conditions: V_{CC} = 5.0 V ±10%, Ta = –20 to +75°C)

Item	Symbol	Min	Max	Unit	Figures
DACK delay time 1	t_{DACD1}	—	18	ns	15.14, 15.20, 15.40, 15.52, 15.66
DACK delay time 2	t_{DACD2}	—	1/2 tcyc + 18	ns	15.14, 15.20, 15.40, 15.52, 15.66
$\overline{WAIT}$ setup time	t_{WTS}	20	—	ns	15.19, 15.43, 15.55, 15.66, 15.70
$\overline{WAIT}$ hold time	t_{WTH}	5	—	ns	15.19, 15.43, 15.55, 15.66, 15.70
$\overline{RAS}$ delay time 1 (SDRAM)	t_{RASD1}	—	18	ns	15.20
$\overline{RAS}$ delay time 2 (DRAM)	t_{RASD2}	1/2 tcyc + 3	1/2 tcyc + 18	ns	15.40
$\overline{CAS}$ delay time 1 (SDRAM)	t_{CASD1}	—	18	ns	15.20
$\overline{CAS}$ delay time 2 (DRAM)	t_{CASD2}	1/2 tcyc + 3	1/2 tcyc + 18	ns	15.40
DQM delay time	t_{DQMD}	—	18	ns	15.20
CKE delay time	t_{CKED}	—	21	ns	15.37
$\overline{CE}$ delay time 1	t_{CED1}	1/2 tcyc + 3	1/2 tcyc + 18	ns	15.52
$\overline{OE}$ delay time 1	t_{OED1}	—	1/2 tcyc + 18	ns	15.52
$\overline{IVECF}$ delay time	t_{IVD}	—	18	ns	15.68
Address input setup time	t_{ASIN}	14	—	ns	15.71
Address input hold time	t_{AHIN}	3	—	ns	15.71
$\overline{BS}$ input setup time	t_{BSS}	15	—	ns	15.71
$\overline{BS}$ input hold time	t_{BSH}	3	—	ns	15.71
Read/write input setup time	t_{RWS}	15	—	ns	15.71
Read/write input hold time	t_{RWH}	3	—	ns	15.71
Address hold time 1	t_{AH1}	5	—	ns	15.15

Table 15.7 Bus Timing With PLL On and 1/4 Cycle Delay [Mode 1, 5]
(Conditions: V_{CC} = 5.0 V ±10%, Ta = –20 to +75°C)

Item	Symbol	Min	Max	Unit	Figures
Address delay time	t_{AD}	1/4 tcyc + 3	1/4 tcyc + 18	ns	15.14, 15.20, 15.40, 15.52, 15.66, 15.68
$\overline{BS}$ delay time	t_{BSD}	—	1/4 tcyc + 21	ns	15.14, 15.20, 15.40, 15.52, 15.66
$\overline{CS}$ delay time 1	t_{CSD1}	—	1/4 tcyc + 21	ns	15.14, 15.20, 15.40, 15.52, 15.66
$\overline{CS}$ delay time 2	t_{CSD2}	—	3/4 tcyc + 21	ns	15.14, 15.66
Read/write delay time	t_{RWD}	1/4 tcyc + 3	1/4 tcyc + 18	ns	15.14, 15.20, 15.40, 15.52, 15.66
Read strobe delay time 1	t_{RSD1}	—	3/4 tcyc + 16	ns	15.14, 15.40, 15.52, 15.66, 15.68
Read data setup time 1	t_{RDS1}	1/4 tcyc + 10	—	ns	15.14, 15.40, 15.52, 15.66, 15.68
Read data setup time 3 (SDRAM)	t_{RDS3}	1/4 tcyc + 8	—	ns	15.20
Read data hold time 2	t_{RDH2}	0	—	ns	15.14, 15.66
Read data hold time 4 (SDRAM)	t_{RDH4}	0	—	ns	15.20
Read data hold time 5 (DRAM)	t_{RDH5}	0	—	ns	15.40
Read data hold time 6 (PSRAM)	t_{RDH6}	0	—	ns	15.52
Read data hold time 7 (interrupt vector)	t_{RDH7}	0	—	ns	15.68
Write enable delay time	t_{WED1}	3/4 tcyc + 3	3/4 tcyc + 18	ns	15.14, 15.15, 15.52, 15.53
Write data delay time 1	t_{WDD}	1/4 tcyc + 3	1/4 tcyc + 18	ns	15.15, 15.27, 15.41, 15.53
Write data hold time 1	t_{WDH1}	1/4 tcyc + 3	—	ns	15.15, 15.27, 15.41, 15.53
Data buffer on time	t_{DON}	—	1/4 tcyc + 18	ns	15.15, 15.27, 15.41, 15.53
Data buffer off time	t_{DOF}	—	1/4 tcyc + 18	ns	15.15, 15.27, 15.41, 15.53

Table 15.7 Bus Timing With PLL On and 1/4 Cycle Delay [Mode 1, 5] (cont)
(Conditions: V_{CC} = 5.0 V ±10%, Ta = –20 to +75°C)

Item	Symbol	Min	Max	Unit	Figures
DACK delay time 1	t_{DACD1}	—	1/4 tcyc + 18	ns	15.14, 15.20, 15.40, 15.52, 15.66
DACK delay time 2	t_{DACD2}	—	3/4 tcyc + 18	ns	15.14, 15.20, 15.40, 15.52, 15.66
$\overline{WAIT}$ setup time	t_{WTS}	20 – 1/4 tcyc	—	ns	15.19, 15.43, 15.55, 15.66, 15.70
$\overline{WAIT}$ hold time	t_{WTH}	1/4 tcyc + 5	—	ns	15.19, 15.43, 15.55, 15.66, 15.70
$\overline{RAS}$ delay time 1 (SDRAM)	t_{RASD1}	—	1/4 tcyc + 18	ns	15.20
$\overline{RAS}$ delay time 2 (DRAM)	t_{RASD2}	3/4 tcyc + 3	3/4 tcyc + 18	ns	15.40
$\overline{CAS}$ delay time 1 (SDRAM)	t_{CASD1}	—	1/4 tcyc + 18	ns	15.20
$\overline{CAS}$ delay time 2 (DRAM)	t_{CASD2}	3/4 tcyc + 3	3/4 tcyc + 18	ns	15.40
DQM delay time	t_{DQMD}	—	1/4 tcyc + 18	ns	15.20
CKE delay time	t_{CKED}	—	1/4 tcyc + 21	ns	15.37
$\overline{CE}$ delay time 1	t_{CED1}	3/4 tcyc + 3	3/4 tcyc + 18	ns	15.52
$\overline{OE}$ delay time 1	t_{OED1}	—	3/4 tcyc + 18	ns	15.52
$\overline{IVECF}$ delay time	t_{IVD}	—	1/4 tcyc + 18	ns	15.68
Address input setup time	t_{ASIN}	14 – 1/4 tcyc	—	ns	15.71
Address input hold time	t_{AHIN}	1/4 tcyc + 3	—	ns	15.71
$\overline{BS}$ input setup time	t_{BSS}	15 – 1/4 tcyc	—	ns	15.71
$\overline{BS}$ input hold time	t_{BSH}	1/4 tcyc + 3	—	ns	15.71
Read/write input setup time	t_{RWS}	15 – 1/4 tcyc	—	ns	15.71
Read/write input hold time	t_{RWH}	1/4 tcyc + 3	—	ns	15.71
Address hold time 1	t_{AH1}	5	—	ns	15.15

Table 15.8 Bus Timing With PLL Off (CKIO Input) [Mode 6]
(Conditions: V_{CC} = 5.0 V ±10%, Ta = –20 to +75°C)

Item	Symbol	Min	Max	Unit	Figures
Address delay time	t_{AD}	13	28	ns	15.16, 15.38, 15.47, 15.60, 15.67, 15.69
$\overline{BS}$ delay time	t_{BSD}	—	30	ns	15.16, 15.38, 15.47, 15.60, 15.67
$\overline{CS}$ delay time 1	t_{CSD1}	—	30	ns	15.16, 15.38, 15.47, 15.60, 15.67
$\overline{CS}$ delay time 3	t_{CSD3}	—	28	ns	15.16, 15.67
Read write delay time	t_{RWD}	13	28	ns	15.16, 15.38, 15.47, 15.60, 15.67
Read strobe delay time 2	t_{RSD2}	—	26	ns	15.16, 15.47, 15.60, 15.67, 15.69
Read data setup time 2	t_{RDS2}	10	—	ns	15.16, 15.38, 15.47, 15.60, 15.67, 15.69
Read data hold time 2	t_{RDH2}	0	—	ns	15.16, 15.67
Read data hold time 3	t_{RDH3}	15	—	ns	15.38
Read data hold time 5 (DRAM)	t_{RDH5}	0	—	ns	15.47
Read data hold time 6 (PSRAM)	t_{RDH6}	0	—	ns	15.60
Read data hold time 7 (interrupt vector)	t_{RDH7}	0	—	ns	15.69
Write enable delay time 2	t_{WED2}	10	25	ns	15.17, 15.61
Write data delay time	t_{WDD}	10	25	ns	15.17, 15.39, 15.48, 15.61
Write data hold time 1	t_{WDH1}	3		ns	15.17, 15.39, 15.48, 15.61
Write data hold time 2	t_{WDH2}	5		ns	15.17
Write data hold time 3	t_{WDH3}	3		ns	15.61
DACK delay time 1	t_{DACD1}	—	25	ns	15.16, 15.38, 15.47, 15.60, 15.67
DACK delay time 3	t_{DACD3}	—	25	ns	15.16, 15.38, 15.47, 15.60, 15.67

Table 15.8 Bus Timing With PLL Off (CKIO Input) [Mode 6] (cont)
(Conditions: V_{CC} = 5.0 V ±10%, Ta = –20 to +75°C)

Item	Symbol	Min	Max	Unit	Figures
$\overline{WAIT}$ setup time	t_{WTS}	20	—	ns	15.19, 15.43, 15.55, 15.67, 15.70
$\overline{WAIT}$ hold time	t_{WTH}	15	—	ns	15.19, 15.43, 15.55, 15.67, 15.70
$\overline{RAS}$ delay time 1 (SDRAM)	t_{RASD1}	—	25	ns	15.38
$\overline{RAS}$ delay time 3 (DRAM)	t_{RASD3}	10	25	ns	15.47
$\overline{CAS}$ delay time 1 (SDRAM)	t_{CASD1}	—	25	ns	15.38
$\overline{CAS}$ delay time 3 (DRAM)	t_{CASD3}	10	25	ns	15.47
DQM delay time	t_{DQMD}	—	25	ns	15.38
CKE delay time	t_{CKED}	—	25	ns	15.37
$\overline{CE}$ delay time 2	t_{CED2}	10	25	ns	15.60
$\overline{OE}$ delay time 2	t_{OED2}	—	25	ns	15.60
$\overline{IVECF}$ delay time	t_{IVD}	—	25	ns	15.69
$\overline{WE}$ setup time	t_{WES1}	0	—	ns	15.16
Address setup time 1	t_{AS1}	0	—	ns	15.17
Address setup time 2	t_{AS2}	3	—	ns	15.60
Address hold time 2	t_{AH2}	0	—	ns	15.17
Row address setup time	t_{ASR}	3	—	ns	15.47
Column address setup time	t_{ASC}	3	—	ns	15.47
Write command setup time	t_{WCS}	3	—	ns	15.48
Write data setup time	t_{WDS}	3	—	ns	15.48
Address input setup time*	t_{ASIN}	15	—	ns	15.71
Address input hold time*	t_{AHIN}	10	—	ns	15.71
$\overline{BS}$ input setup time*	t_{BSS}	15	—	ns	15.71
$\overline{BS}$ input hold time*	t_{BSH}	10	—	ns	15.71
Read/write input setup time*	t_{RWS}	15	—	ns	15.71
Read/write input hold time*	t_{RWH}	10	—	ns	15.71
Data buffer on time	t_{DON}	—	25	ns	15.17, 15.39, 15.48, 15.61
Data buffer off time	t_{DOF}	—	25	ns	15.17, 15.39, 15.48, 15.61

Note: * When the external addresses monitor function is used, the PLL must be on.

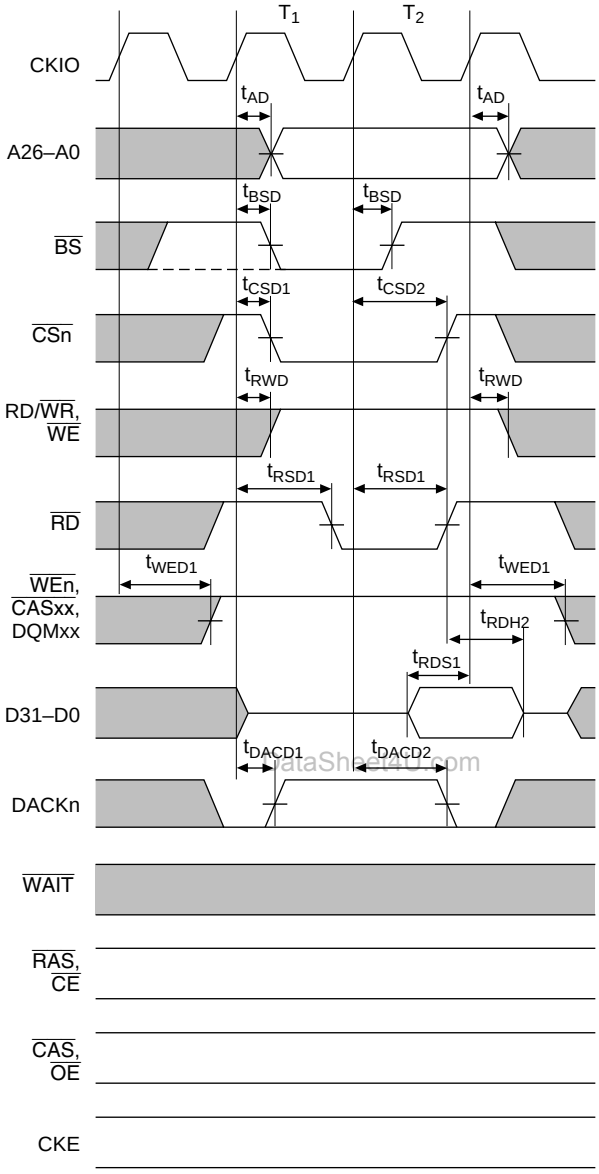
Table 15.9 Bus Timing With PLL Off (CKIO Output) [Mode 2]
(Conditions: V_{CC} = 5.0 V ±10%, Ta = –20 to +75°C)

Item	Symbol	Min	Max	Unit	Figures
Address delay time	t_{AD}	3	18	ns	15.16, 15.38, 15.47, 15.60, 15.67, 15.69
$\overline{BS}$ delay time	t_{BSD}	—	21	ns	15.16, 15.38, 15.47, 15.60, 15.67
$\overline{CS}$ delay time 1	t_{CSD1}	—	21	ns	15.16, 15.38, 15.47, 15.60, 15.67
$\overline{CS}$ delay time 3	t_{CSD3}	—	21	ns	15.16, 15.67
Read write delay time	t_{RWD}	3	18	ns	15.16, 15.38, 15.47, 15.60, 15.67
Read strobe delay time 2	t_{RSD2}	—	16	ns	15.16, 15.47, 15.60, 15.67, 15.69
Read data setup time 2	t_{RDS2}	12	—	ns	15.16, 15.38, 15.47, 15.60, 15.67, 15.69
Read data hold time 2	t_{RDH2}	0	—	ns	15.16, 15.67
Read data hold time 3 (SDRAM)	t_{RDH3}	1/2 tcyc	—	ns	15.38
Read data hold time 5 (DRAM)	t_{RDH5}	0	—	ns	15.47
Read data hold time 6 (PSRAM)	t_{RDH6}	0	—	ns	15.60
Read data hold time 7 (interrupt vector)	t_{RDH7}	0	—	ns	15.69
Write enable delay time 2	t_{WED2}	3	18	ns	15.17, 15.61
Write data delay time	t_{WDD}	3	18	ns	15.17, 15.39, 15.48, 15.61
Write data hold time 1	t_{WDH1}	3	—	ns	15.17, 15.39, 15.48, 15.61
Write data hold time 2	t_{WDH2}	5	—	ns	15.17
Write data hold time 3	t_{WDH3}	3	—	ns	15.61
DACK delay time 1	t_{DACD1}	—	18	ns	15.16, 15.38, 15.47, 15.60, 15.67
DACK delay time 3	t_{DACD3}	—	18	ns	15.16, 15.38, 15.47, 15.60, 15.67

Table 15.9 Bus Timing With PLL Off (CKIO Output) [Mode 2] (cont)
(Conditions: V_{CC} = 5.0 V ±10%, Ta = –20 to +75°C)

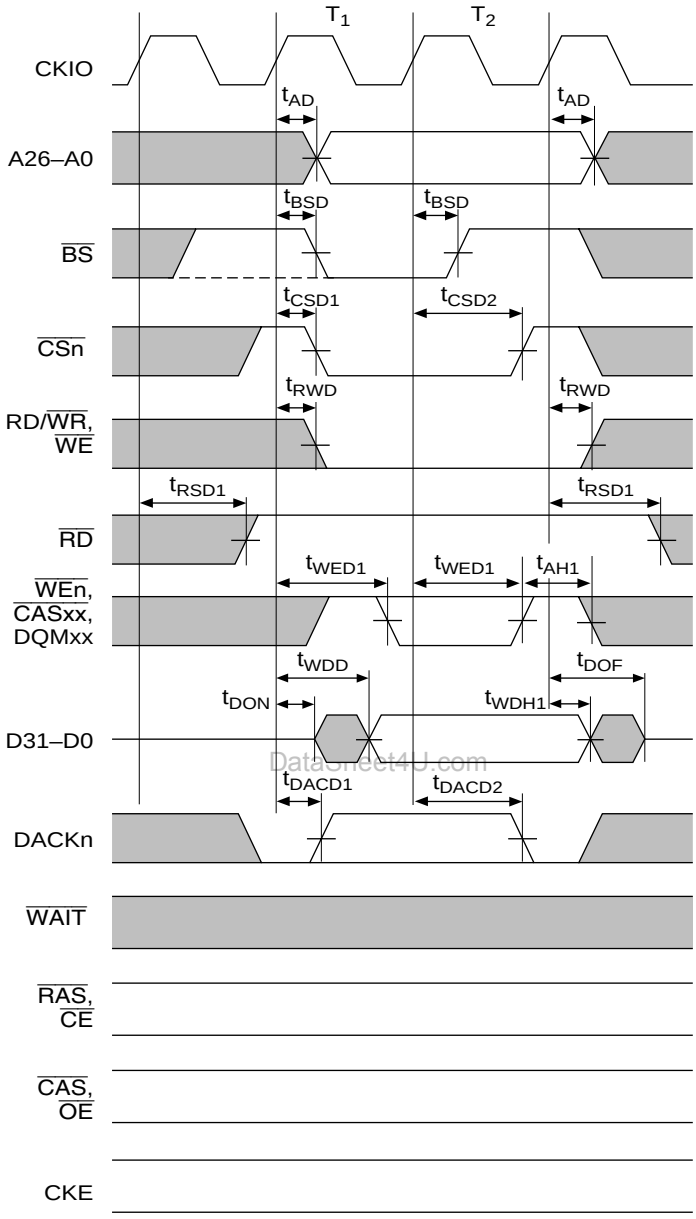
Item	Symbol	Min	Max	Unit	Figures
$\overline{WAIT}$ setup time	t_{WTS}	22	—	ns	15.19, 15.43, 15.55, 15.67, 15.70
$\overline{WAIT}$ hold time	t_{WTH}	5	—	ns	15.19, 15.43, 15.55, 15.67, 15.70
$\overline{RAS}$ delay time 1 (SDRAM)	t_{RASD1}	—	18	ns	15.38
$\overline{RAS}$ delay time 3 (DRAM)	t_{RASD3}	3	18	ns	15.47
$\overline{CAS}$ delay time 1 (SDRAM)	t_{CASD1}	—	18	ns	15.38
$\overline{CAS}$ delay time 3 (DRAM)	t_{CASD3}	3	18	ns	15.47
DQM delay time	t_{DQMD}	—	18	ns	15.38
CKE delay time	t_{CKED}	—	21	ns	15.37
$\overline{CE}$ delay time 2	t_{CED2}	3	18	ns	15.60
$\overline{OE}$ delay time 2	t_{OED2}	—	18	ns	15.60
$\overline{IVECF}$ delay time	t_{IVD}	—	18	ns	15.69
Address input setup time*	t_{ASIN}	14	—	ns	15.71
Address input hold time*	t_{AHIN}	3	—	ns	15.71
$\overline{BS}$ input setup time*	t_{BSS}	15	—	ns	15.71
$\overline{BS}$ input hold time*	t_{BSH}	3	—	ns	15.71
Read/write input setup time*	t_{RWS}	15	—	ns	15.71
Read/write input hold time*	t_{RWH}	3	—	ns	15.71
Data buffer on time	t_{DON}	—	18	ns	15.17, 15.39, 15.48, 15.61
Data buffer off time	t_{DOF}	—	18	ns	15.17, 15.39, 15.48, 15.61
Address hold time 2	t_{AH2}	5	—	ns	15.17

Note: When the external addresses monitor function is used, the PLL must be on.



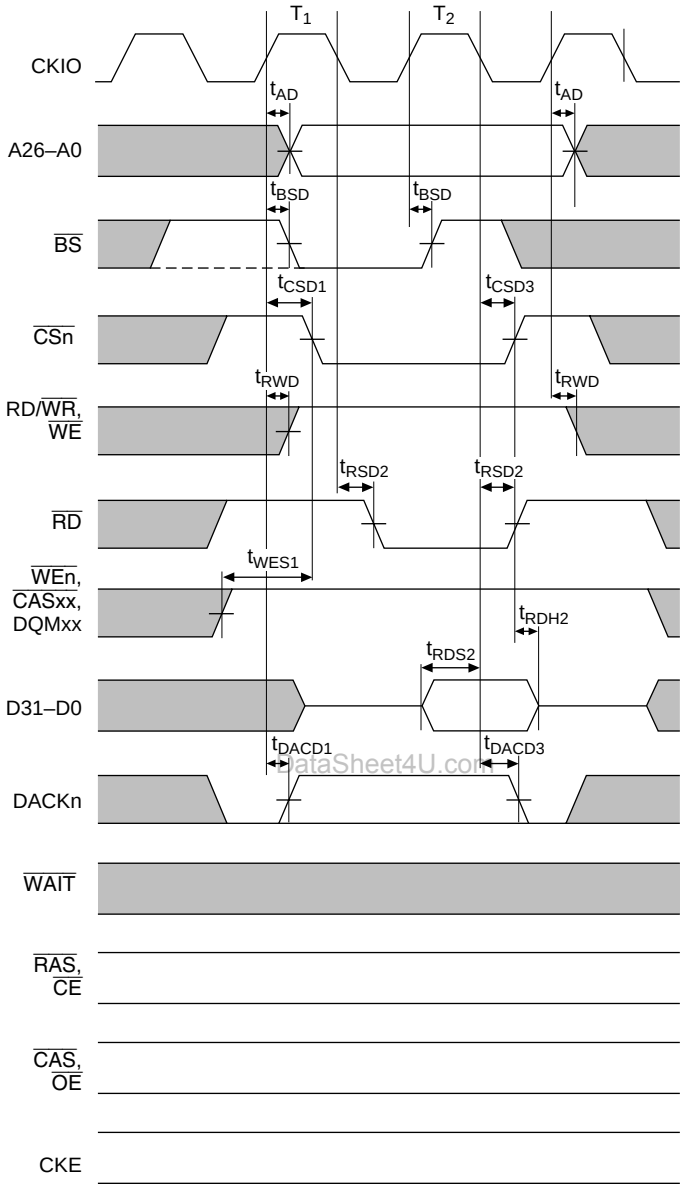
Notes: 1. The dotted line shows the waveform when synchronous DRAM is connected.
2. t_{RDH2} is specified from the rise of $\overline{CSn}$ or $\overline{RD}$, whichever is first.
3. The DACKn waveform shown is for the case where active-high has been specified.

Figure 15.14 Basic Read Cycle (No Waits, PLL On)



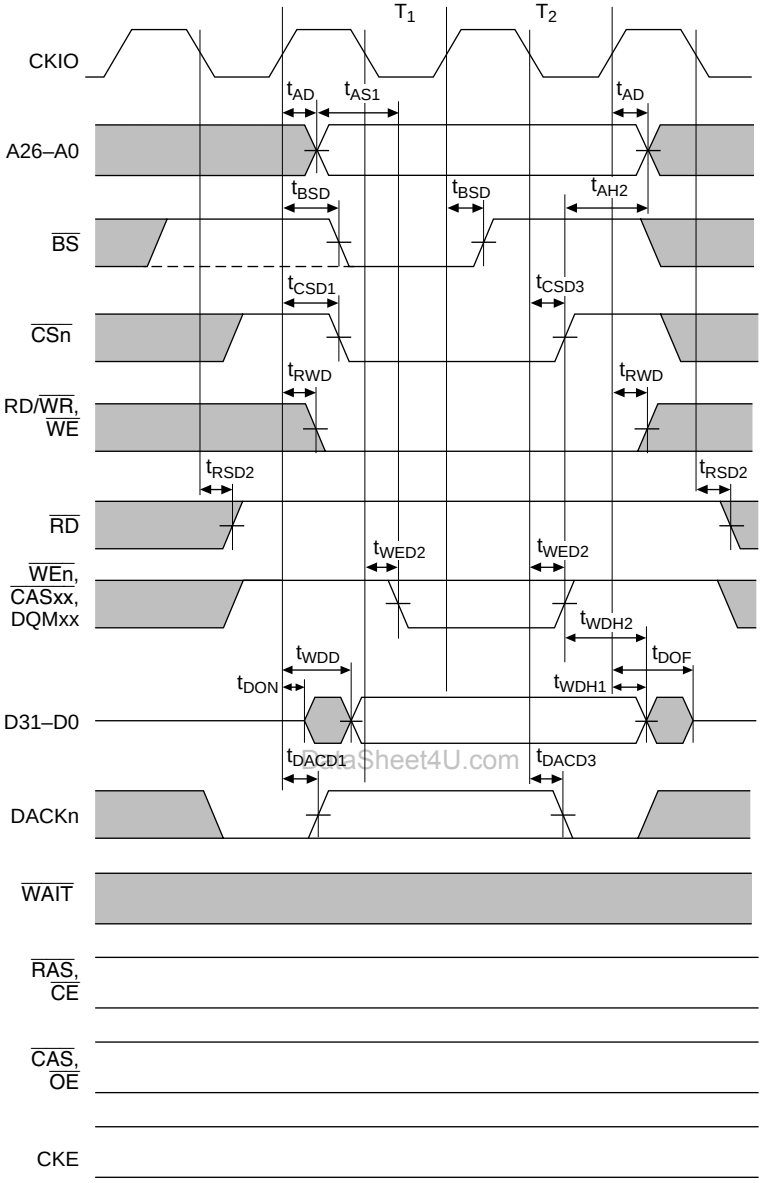
Notes: 1. The dotted line shows the waveform when synchronous DRAM is connected.
2. The DACKn waveform shown is for the case where active-high has been specified.

Figure 15.15 Basic Write Cycle (No Waits, PLL On)



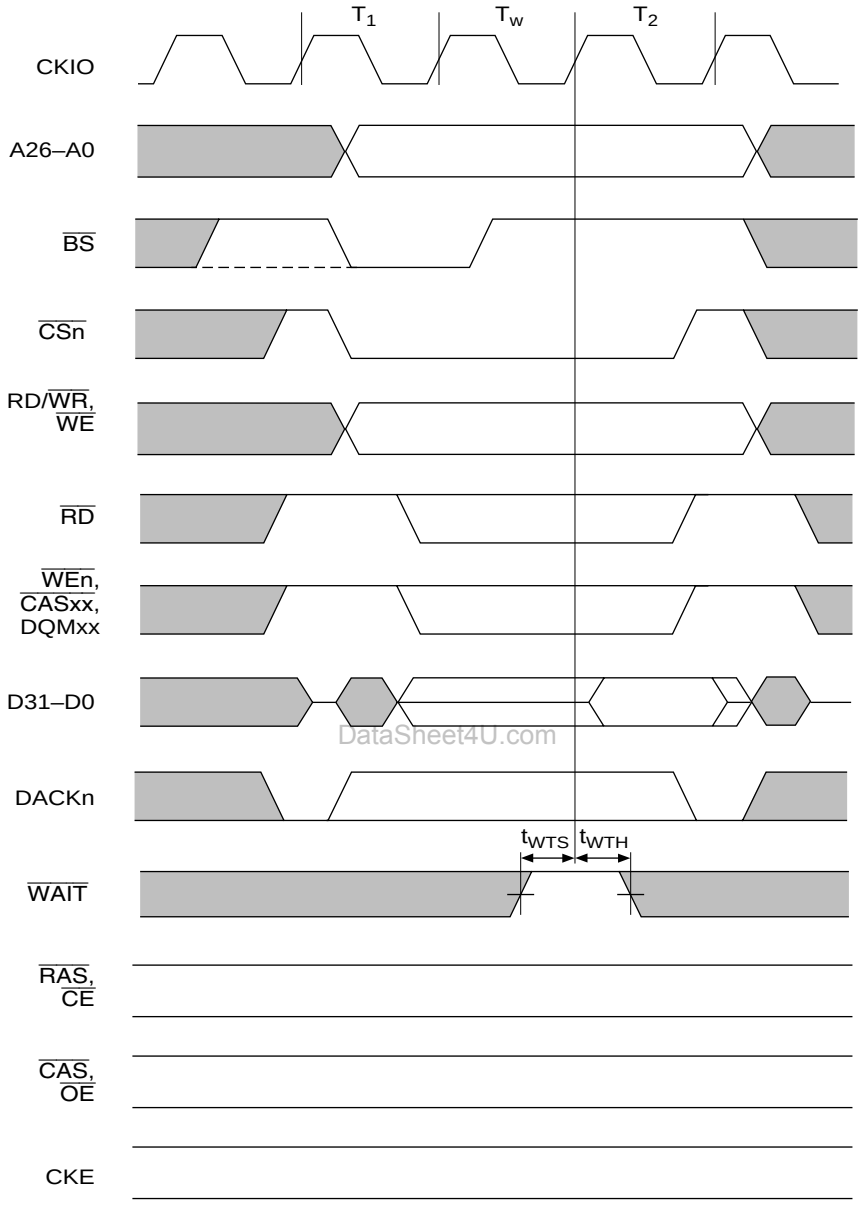
Notes: 1. The dotted line shows the waveform when synchronous DRAM is connected.
2. t_{RDH2} is specified from the rise of $\overline{CSn}$ or $\overline{RD}$, whichever is first.
3. The DACKn waveform shown is for the case where active-high has been specified.

Figure 15.16 Basic Read Cycle (No Waits, PLL Off)



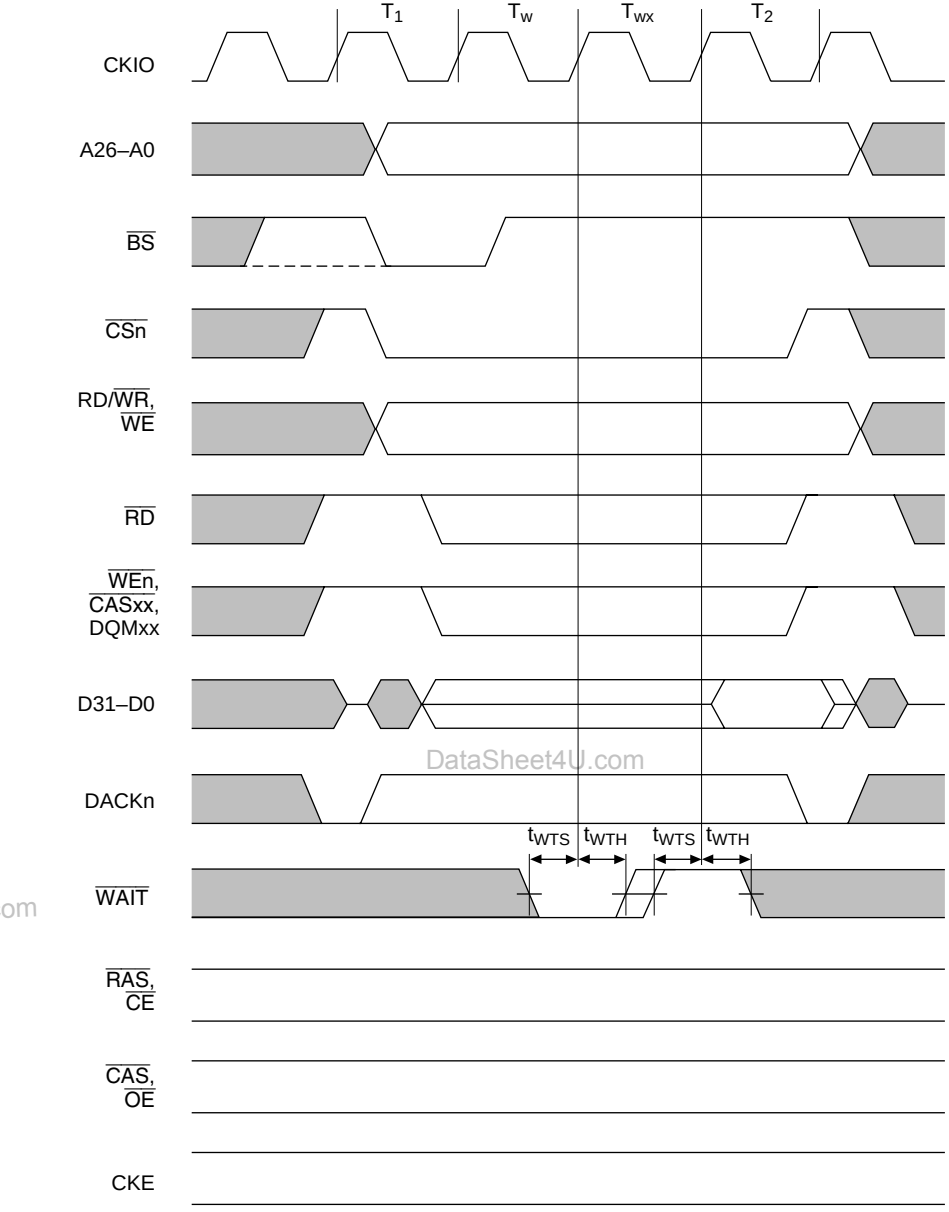
Notes: 1. The dotted line shows the waveform when synchronous DRAM is connected.
2. The DACKn waveform shown is for the case where active-high has been specified.

Figure 15.17 Basic Write Cycle (No Waits, PLL Off)



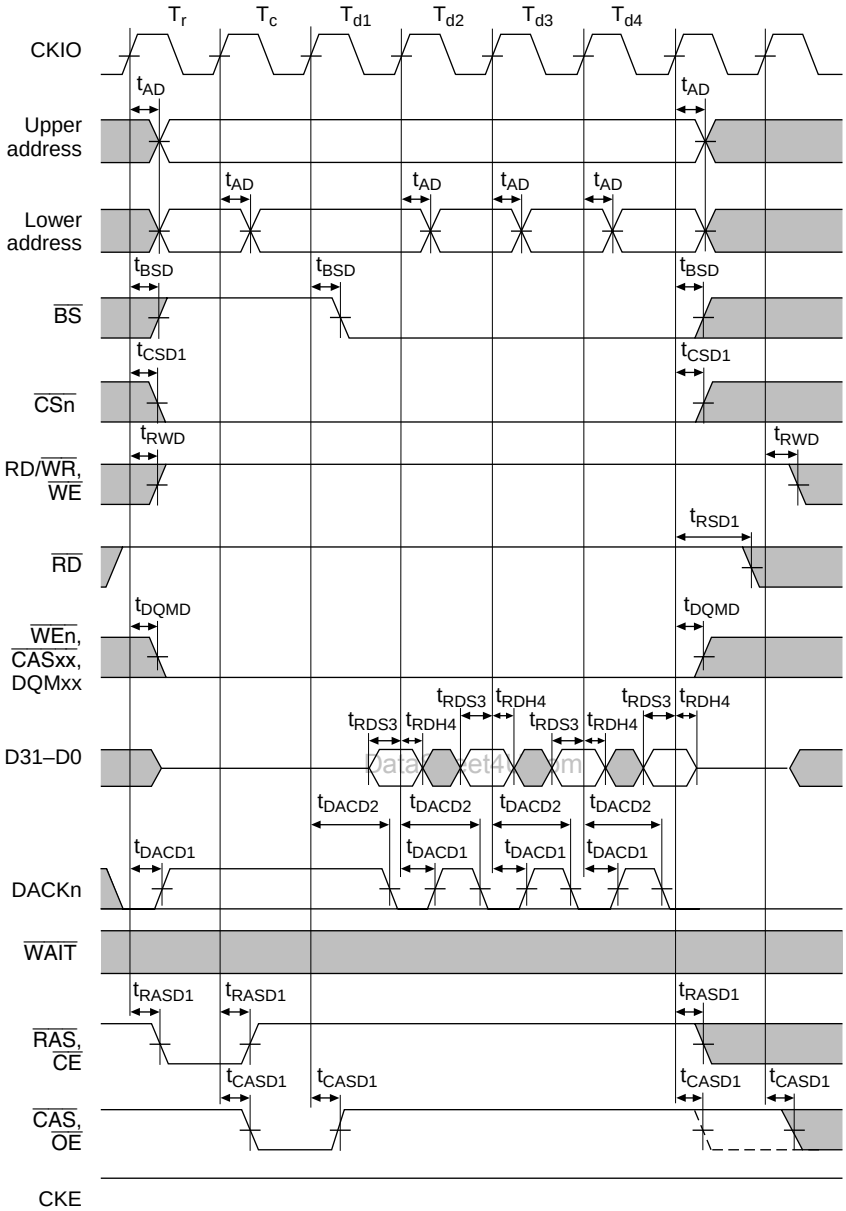
Notes: 1. The dotted line shows the waveform when synchronous DRAM is connected.
2. The DACKn waveform shown is for the case where active-high has been specified.

Figure 15.18 Basic Bus Cycle (1 Wait Cycle)



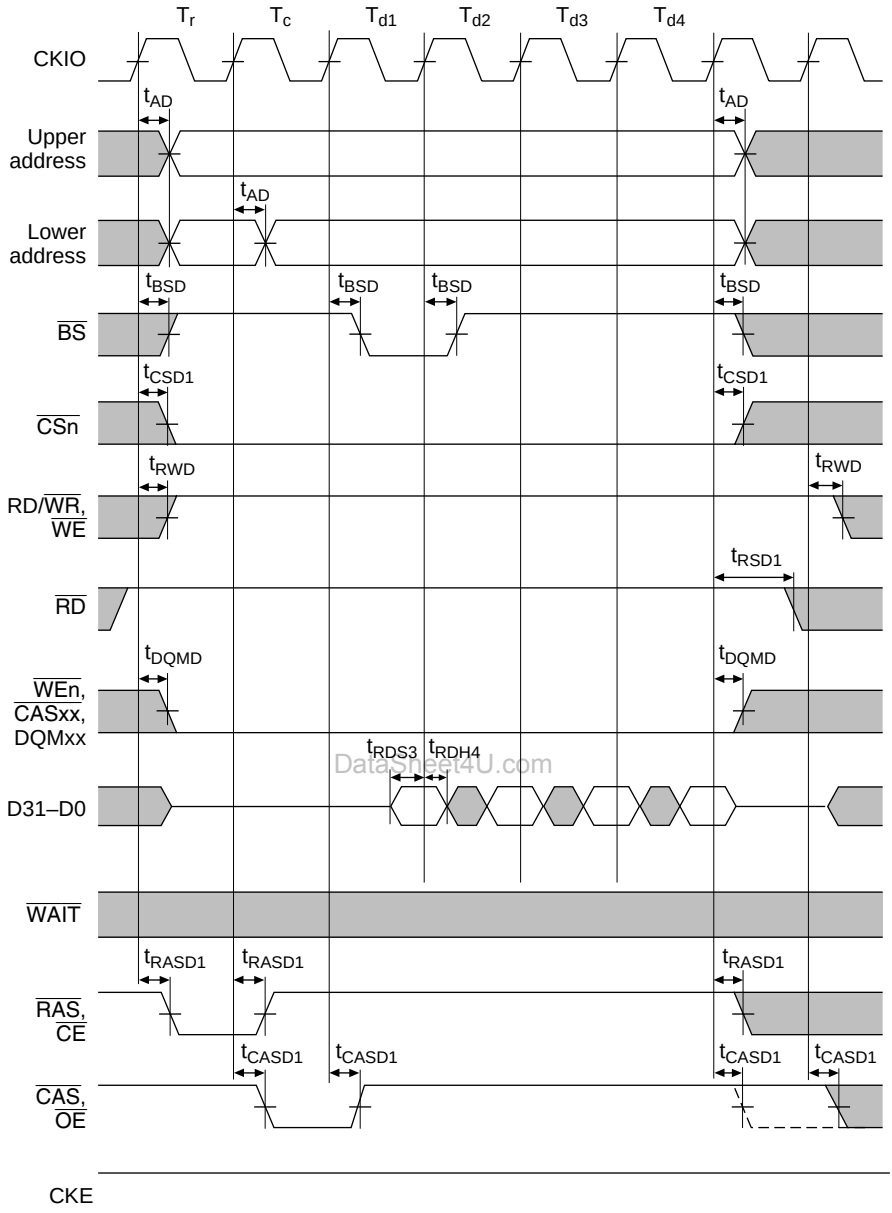
Notes: 1. The dotted line shows the waveform when synchronous DRAM is connected.
2. The DACKn waveform shown is for the case where active-high has been specified.

Figure 15.19 Basic Bus Cycle (External Wait Input)



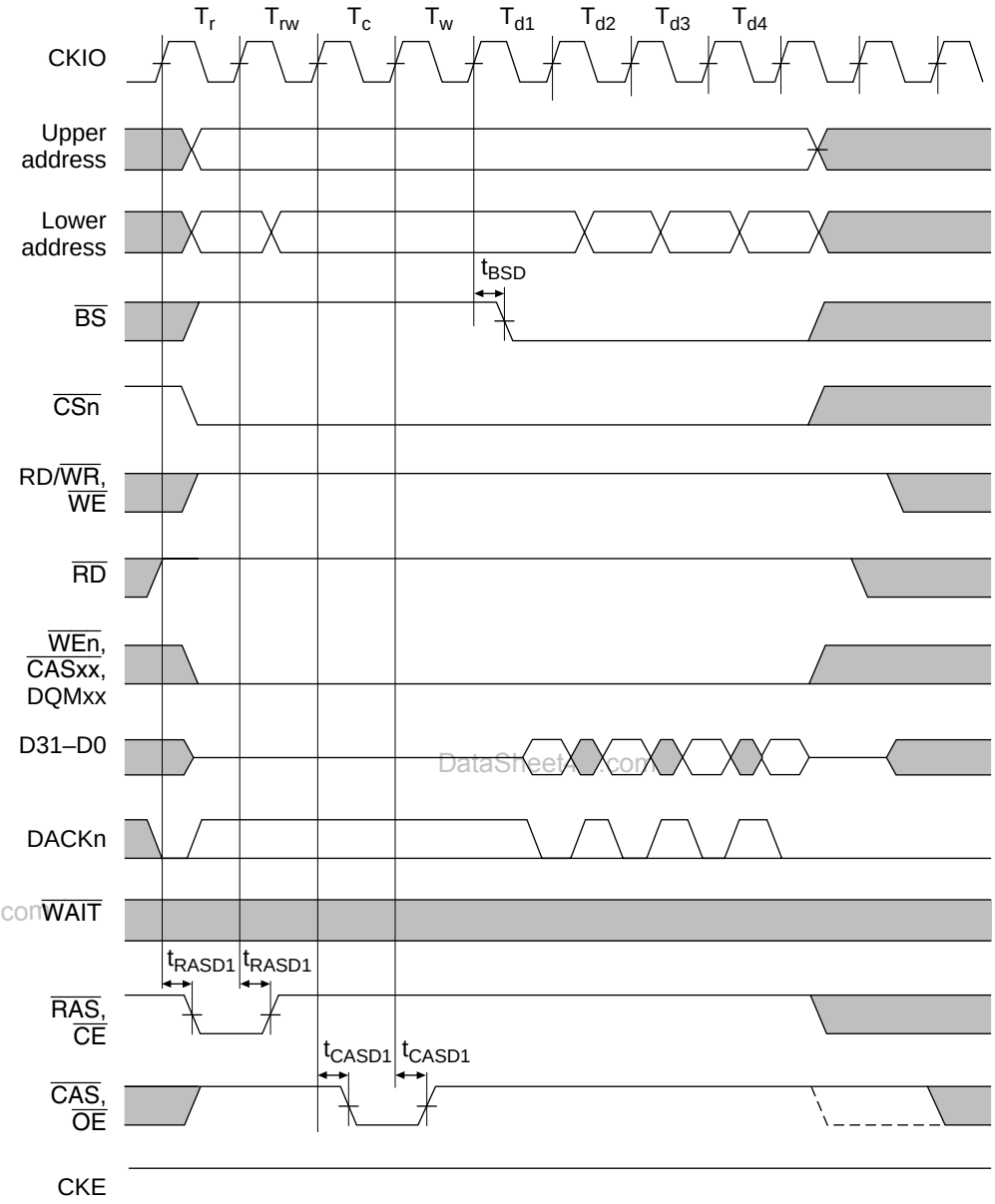
Notes: 1. The dotted line shows the waveform when synchronous DRAM in another CS space is accessed.
2. The DACKn waveform shown is for the case where active-high has been specified.

Figure 15.20 Synchronous DRAM Read Bus Cycle (RCD = 1 Cycle, CAS Latency = 1 Cycle, Bursts = 4, PLL On)



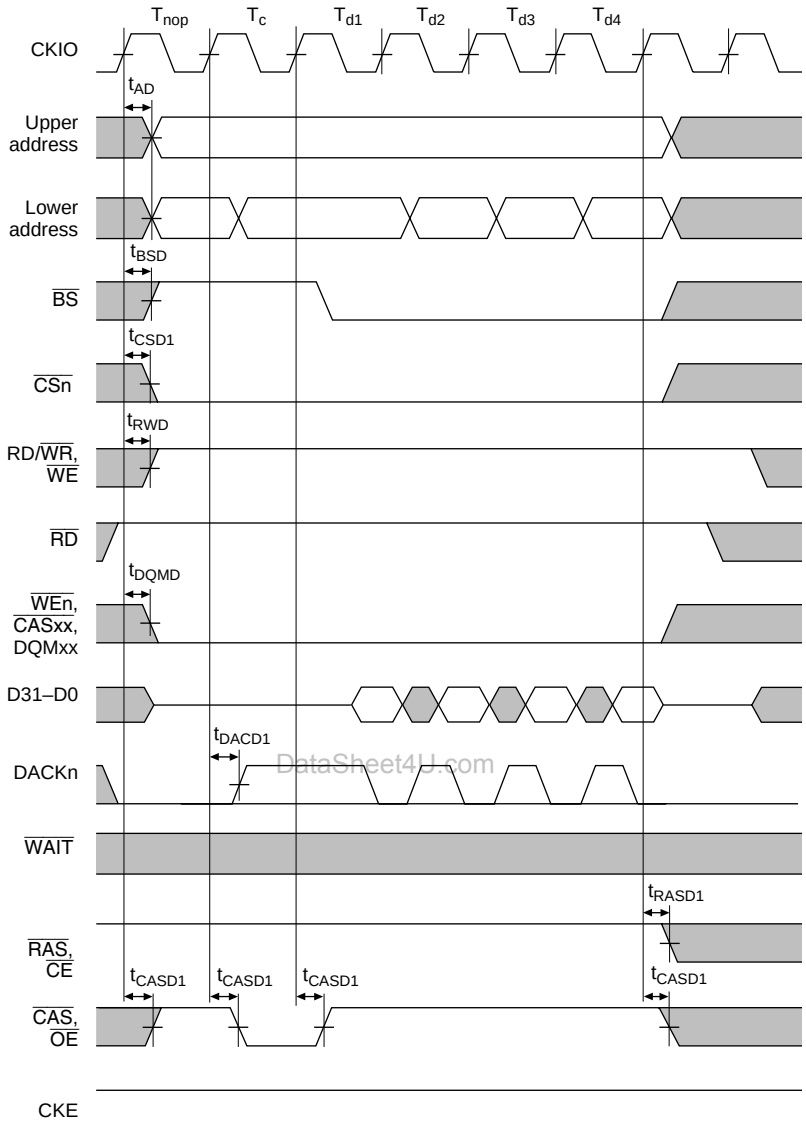
Note: The dotted line shows the waveform when synchronous DRAM in another CS space is accessed.

Figure 15.21 Synchronous DRAM Single Read Bus Cycle (RCD = 1 Cycle, CAS Latency = 1 Cycle, Bursts = 4, PLL On)



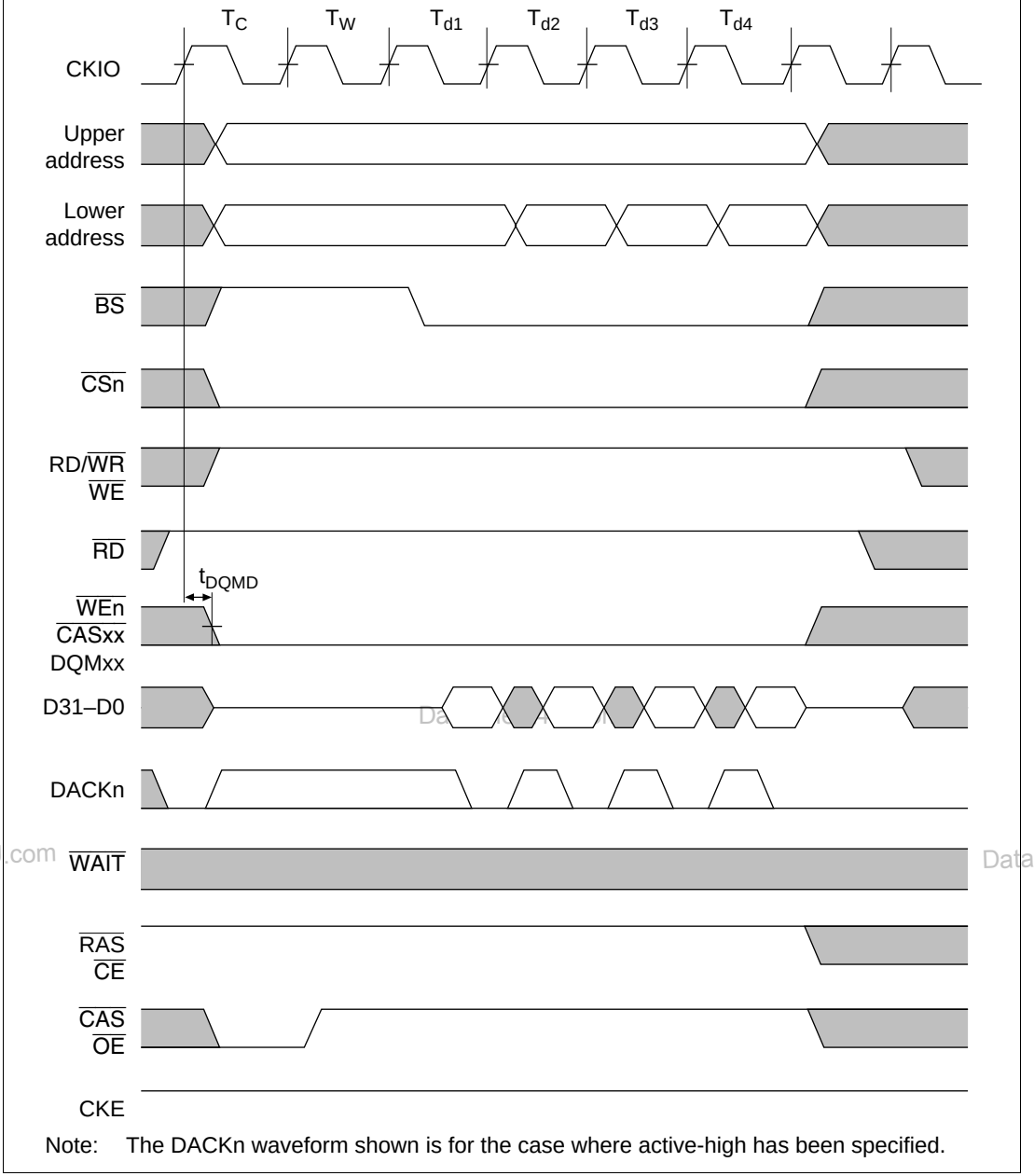
Notes: 1. The dotted line shows the waveform when synchronous DRAM in another CS space is accessed.
2. The DACKn waveform shown is for the case where active-high has been specified.

Figure 15.22 Synchronous DRAM Read Bus Cycle (RCD = 2 Cycles, CAS Latency = 2 Cycles, Bursts = 4)



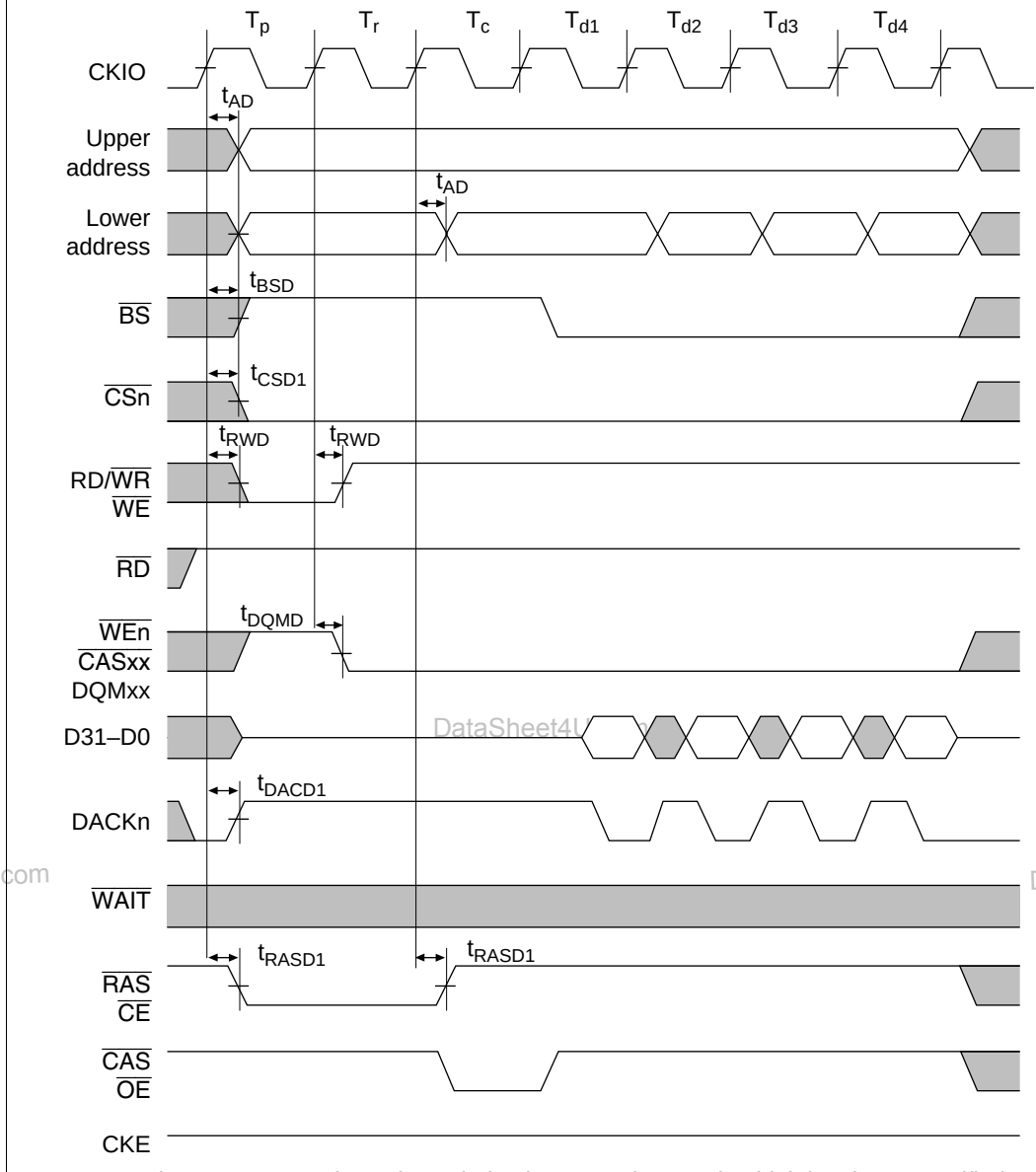
Note: The DACKn waveform shown is for the case where active-high has been specified.

Figure 15.23 Synchronous DRAM Read Bus Cycle (Bank Active, Same Row Access, CAS Latency = 1 Cycle)



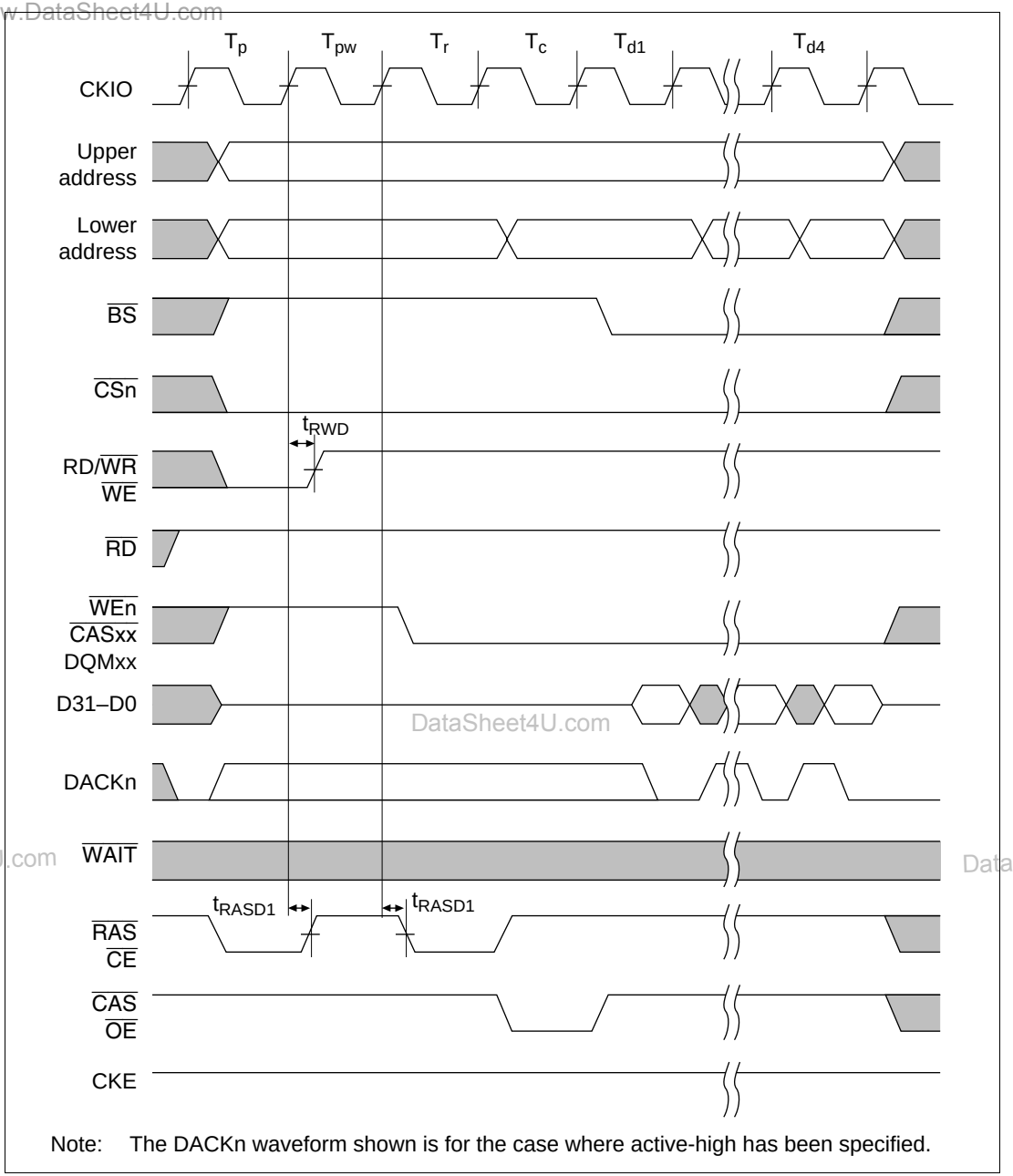
Note: The DACKn waveform shown is for the case where active-high has been specified.

Figure 15.24 Synchronous DRAM Read Bus Cycle (Bank Active, Same Row Access, CAS Latency = 2 Cycles)



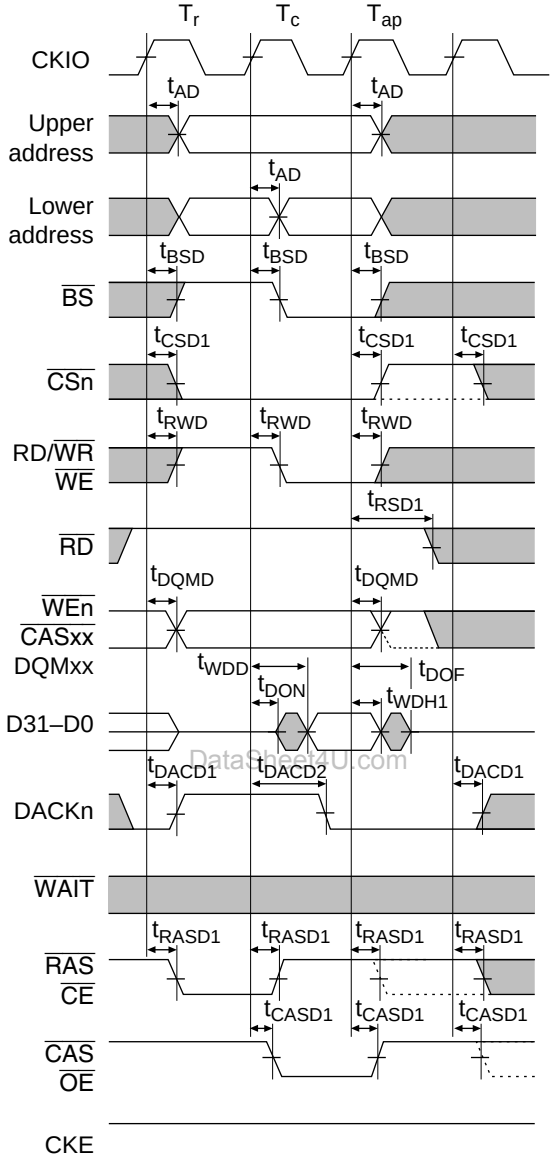
Note: The DACKn waveform shown is for the case where active-high has been specified.

Figure 15.25 Synchronous DRAM Read Bus Cycle (Bank Active, Different Row Access, TRP = 1 Cycle, RCD = 1 Cycle, CAS Latency = 1 Cycle)



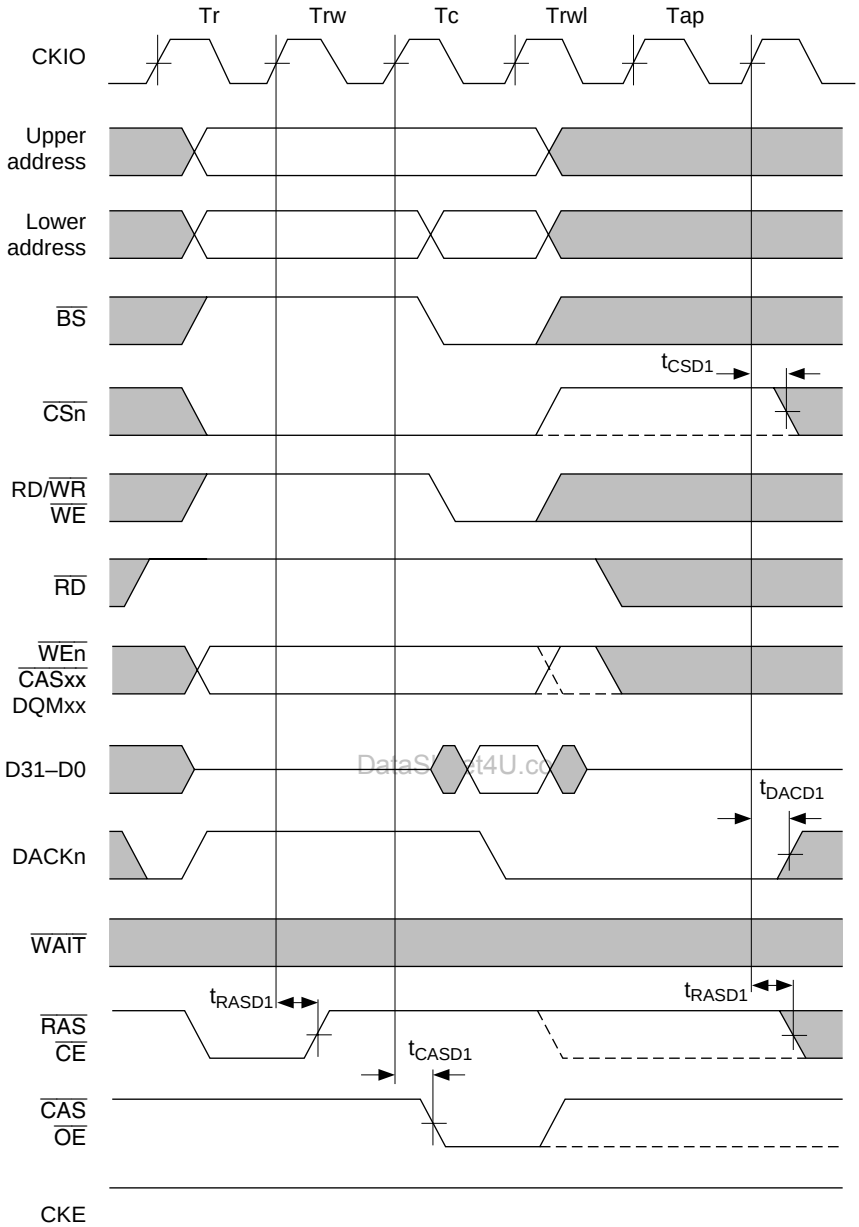
Note: The DACKn waveform shown is for the case where active-high has been specified.

Figure 15.26 Synchronous DRAM Read Bus Cycle (Bank Active, Different Row Access, TRP = 2 Cycles, RCD = 1 Cycle, CAS Latency = 1 Cycle)



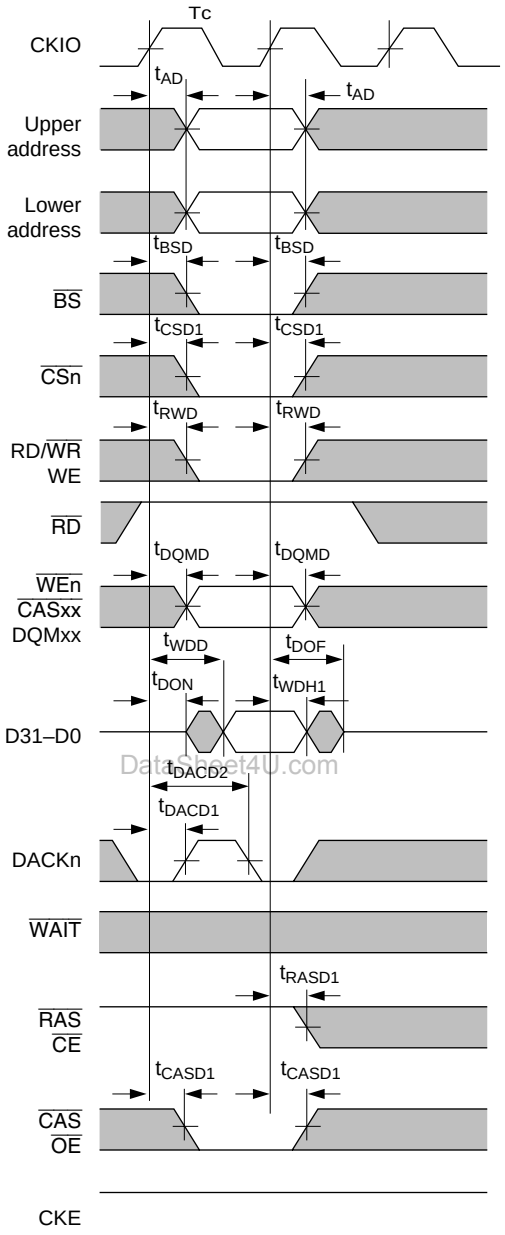
Notes: 1. Dotted lines show the waveforms when synchronous DRAM in another CS space is accessed.
2. The DACKn waveform shown is for the case where active-high has been specified.

Figure 15.27 Synchronous DRAM Write Bus Cycle (RCD = 1 Cycle, TRWL = 1 Cycle, PLL On)



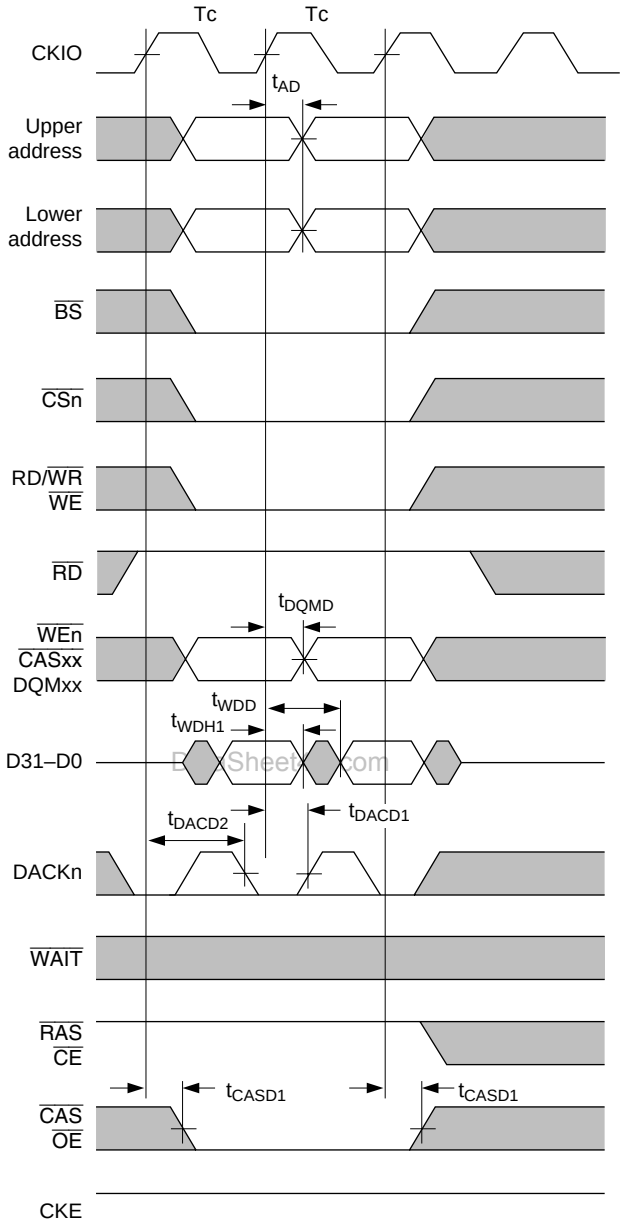
Notes: 1. Dotted lines show the waveforms when synchronous DRAM in another CS space is accessed.
2. The DACKn waveform shown is for the case where active-high has been specified.

Figure 15.28 Synchronous DRAM Write Bus Cycle (RCD = 2 Cycles, TRWL = 2 Cycles)



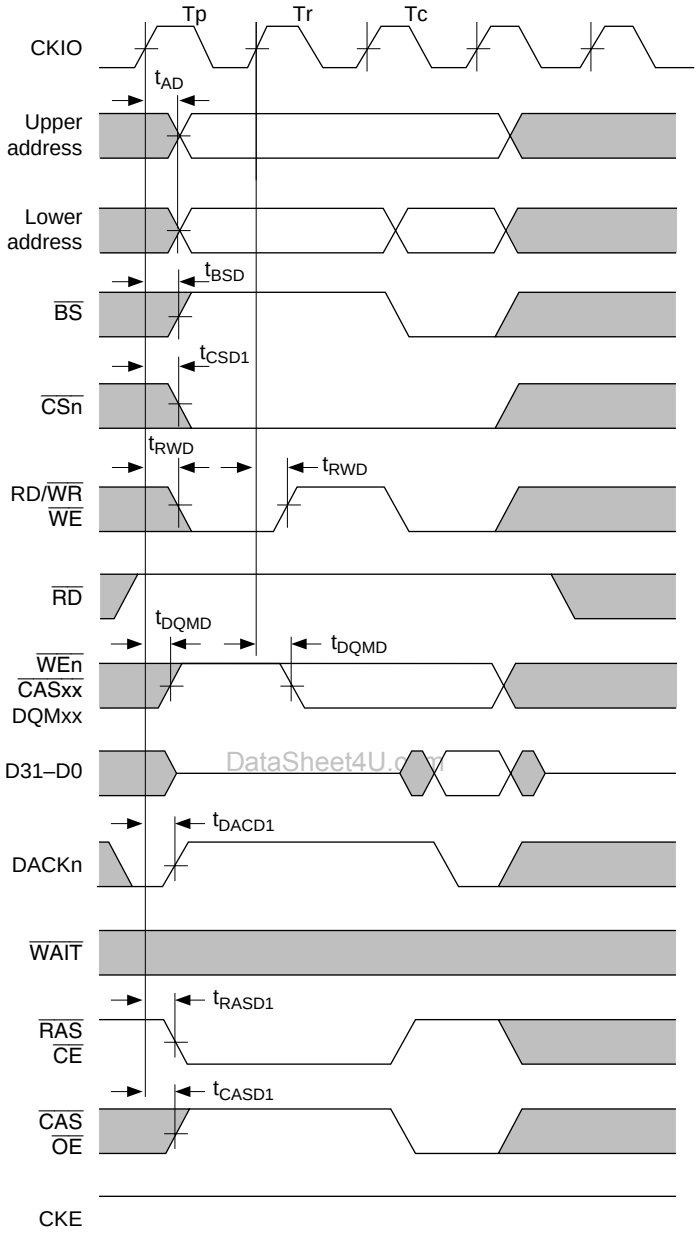
Note: The DACKn waveform shown is for the case where active-high has been specified.

Figure 15.29 Synchronous DRAM Write Bus Cycle (Bank Active, Same Row Access)



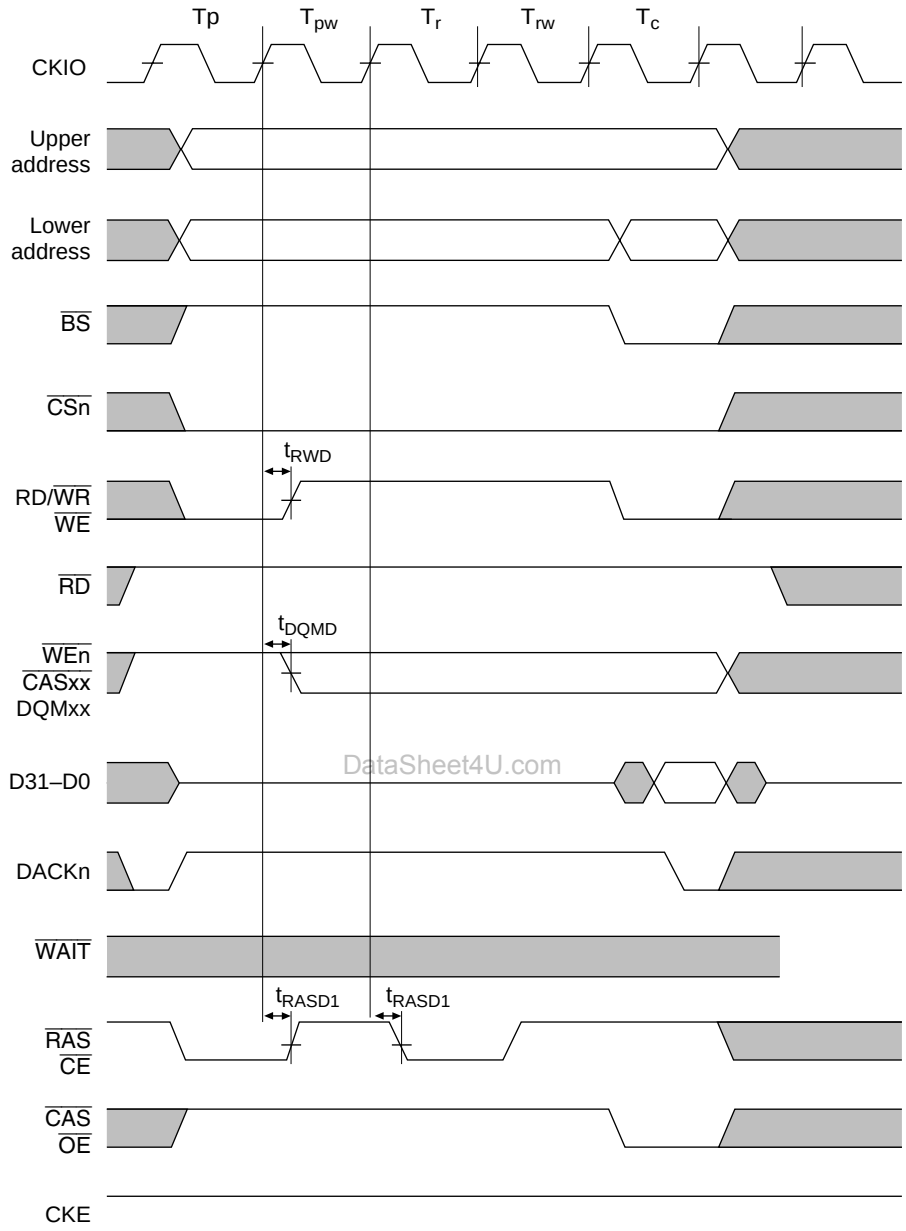
Note: The DACKn waveform shown is for the case where active-high has been specified.

Figure 15.30 Synchronous DRAM Consecutive Write Cycles (Bank Active, Same Row Access)



Note: The DACKn waveform shown is for the case where active-high has been specified.

Figure 15.31 Synchronous DRAM Write Bus Cycle (Bank Active, Different Row Access, TRP = 1 Cycle, RCD = 1 Cycle)



Note: The DACKn waveform shown is for the case where active-high has been specified.

Figure 15.32 Synchronous DRAM Write Bus Cycle (Bank Active, Different Row Access, TRP = 2 Cycles, RCD = 2 Cycles)

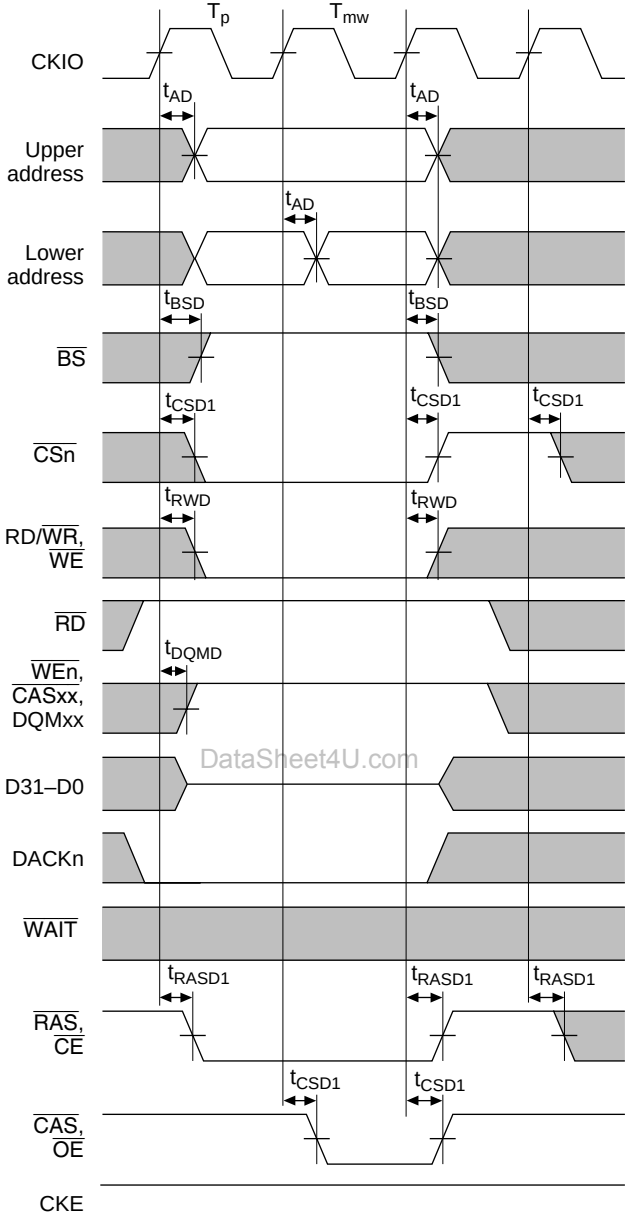


Figure 15.33 Synchronous DRAM Mode Register Write Cycle (TRP = 1 Cycle)

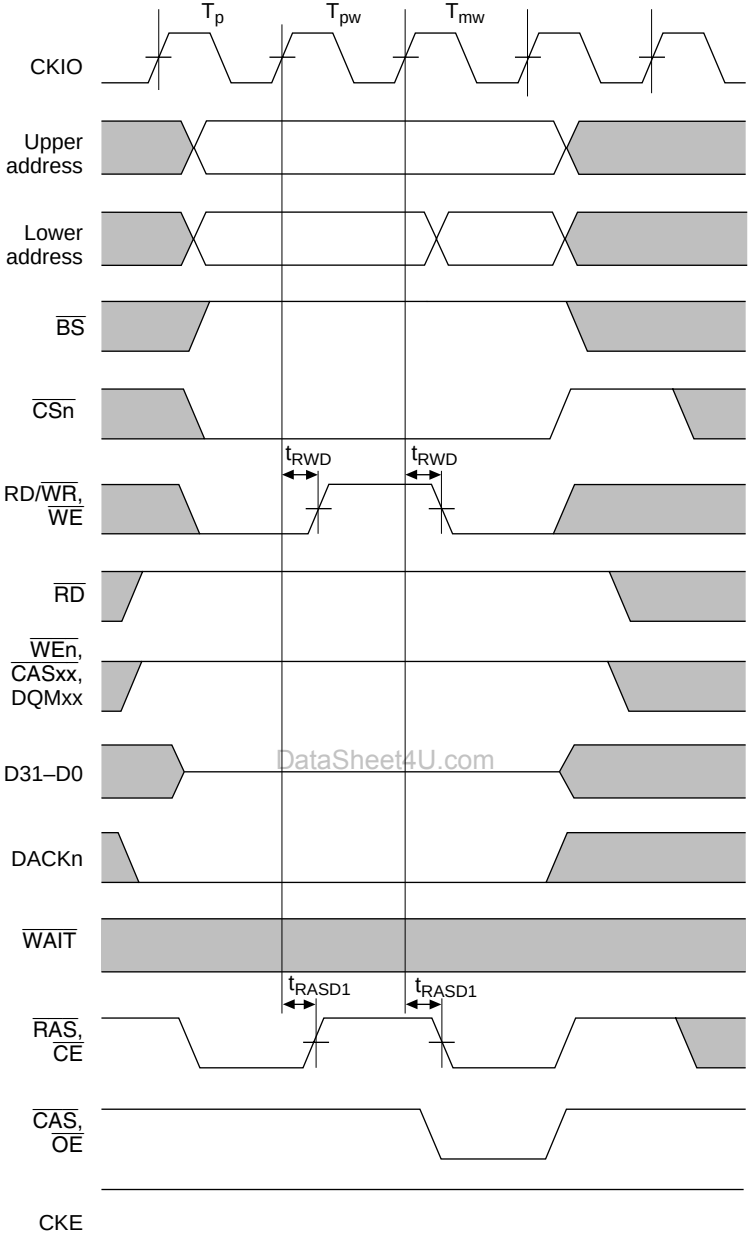
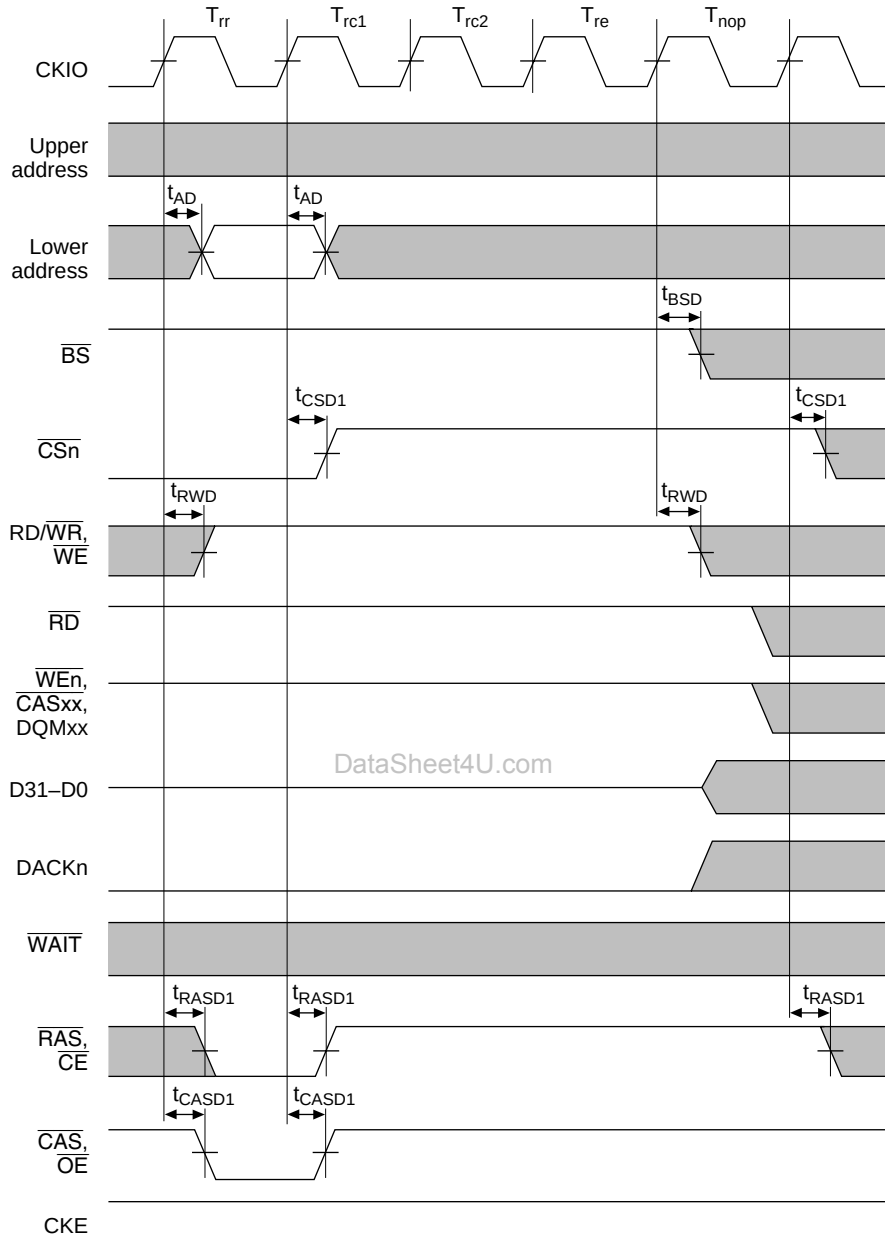


Figure 15.34 Synchronous DRAM Mode Register Write Cycle (TRP = 2 Cycles)



Note: A precharge cycle always precedes the auto-refresh cycle by the number of cycles specified by TRP.

Figure 15.35 Synchronous DRAM Auto-Refresh Cycle (TRAS = 2 Cycles)

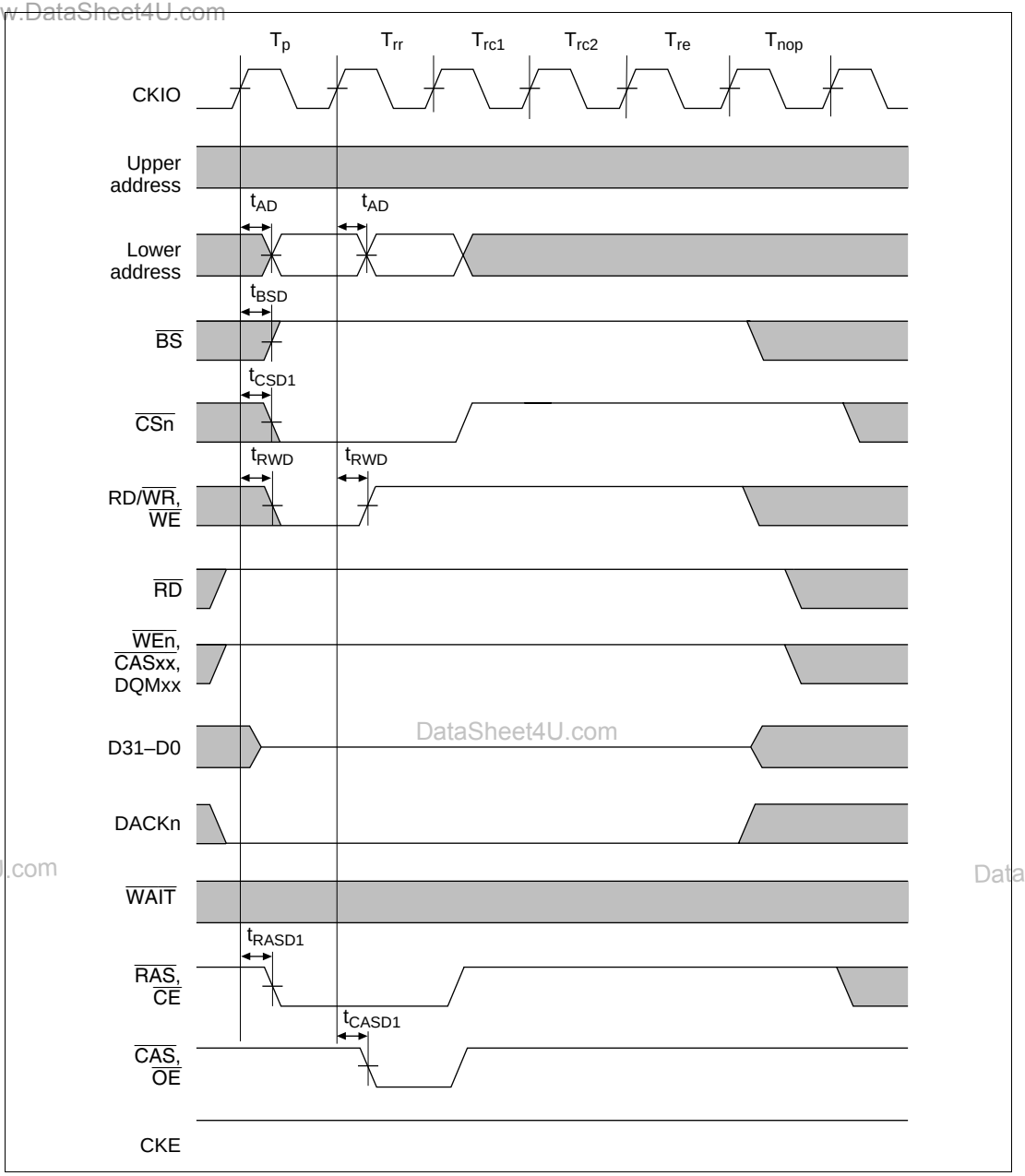
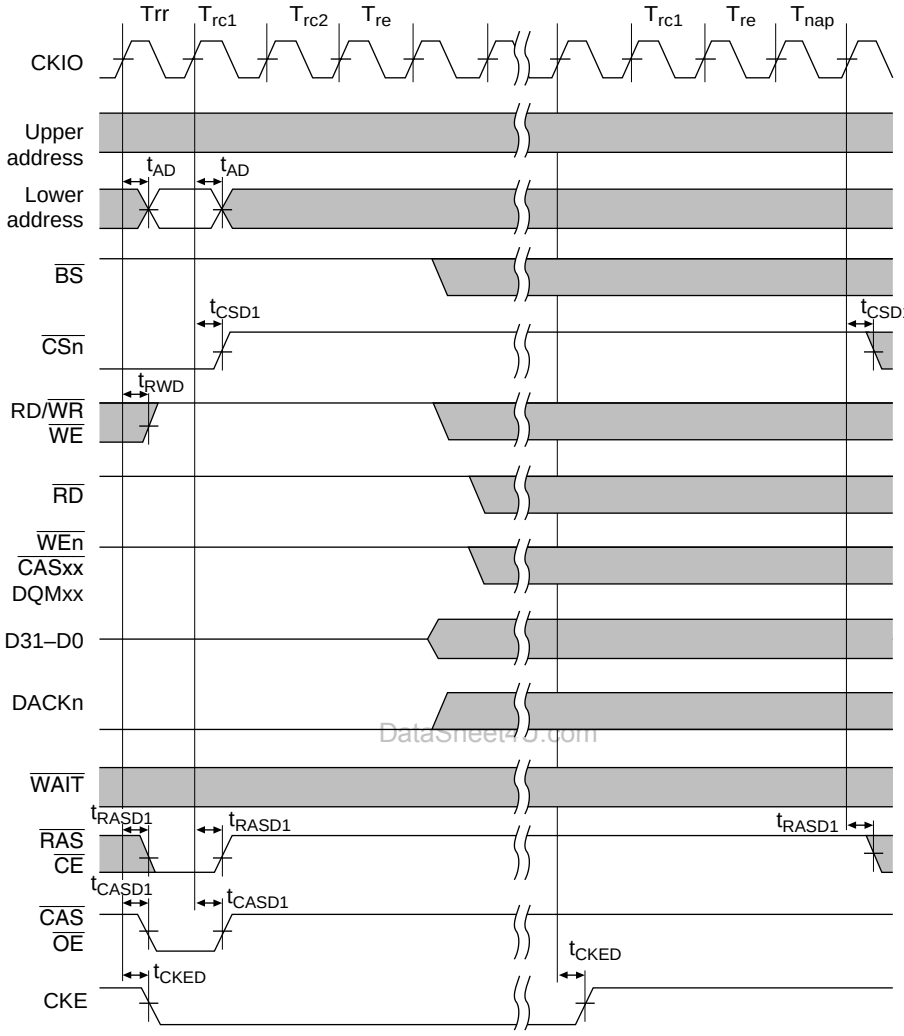
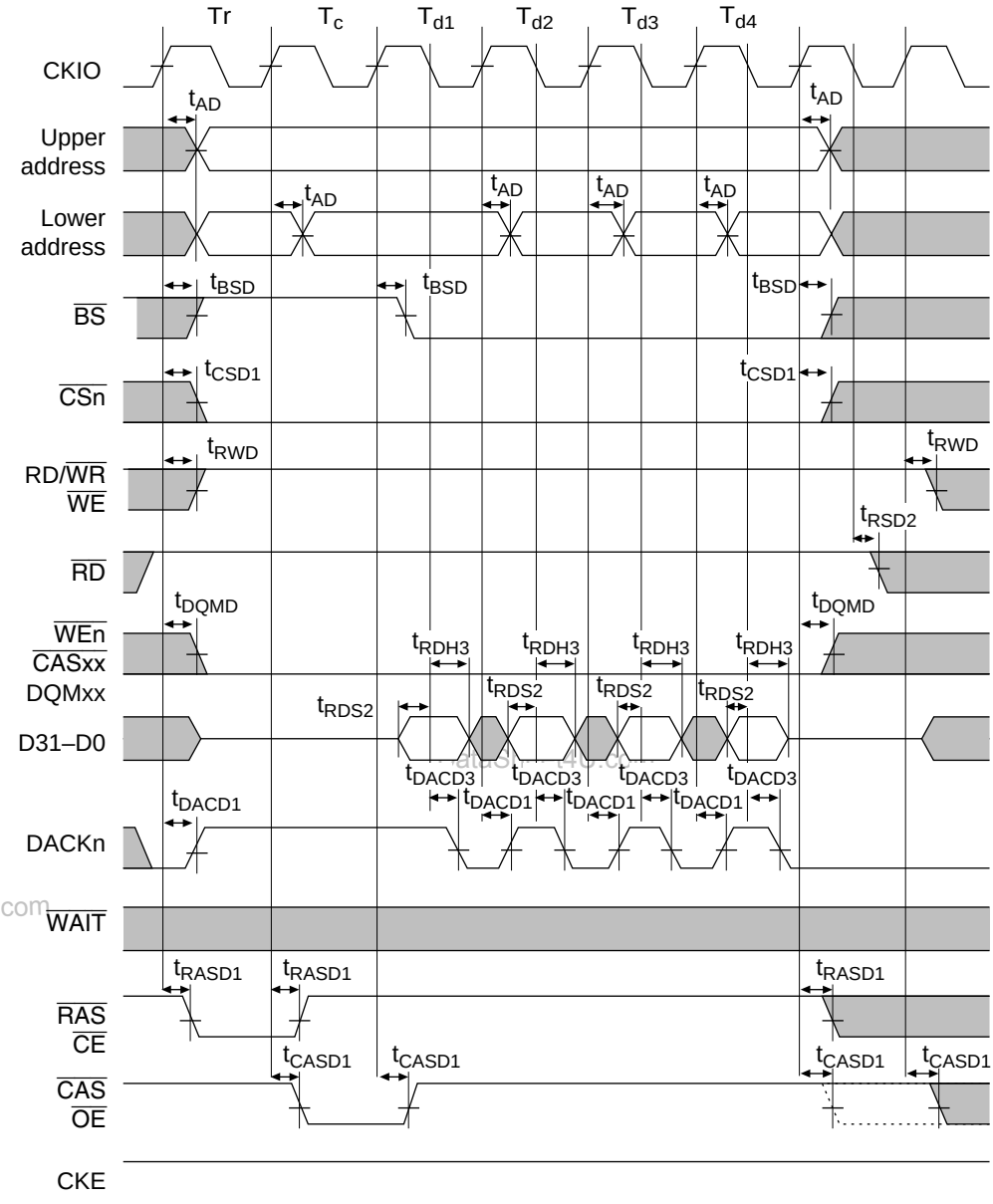


Figure 15.36 Synchronous DRAM Auto-Refresh Cycle (Shown From Precharge Cycle, TRP = 1 Cycle, TRAS = 2 Cycles)



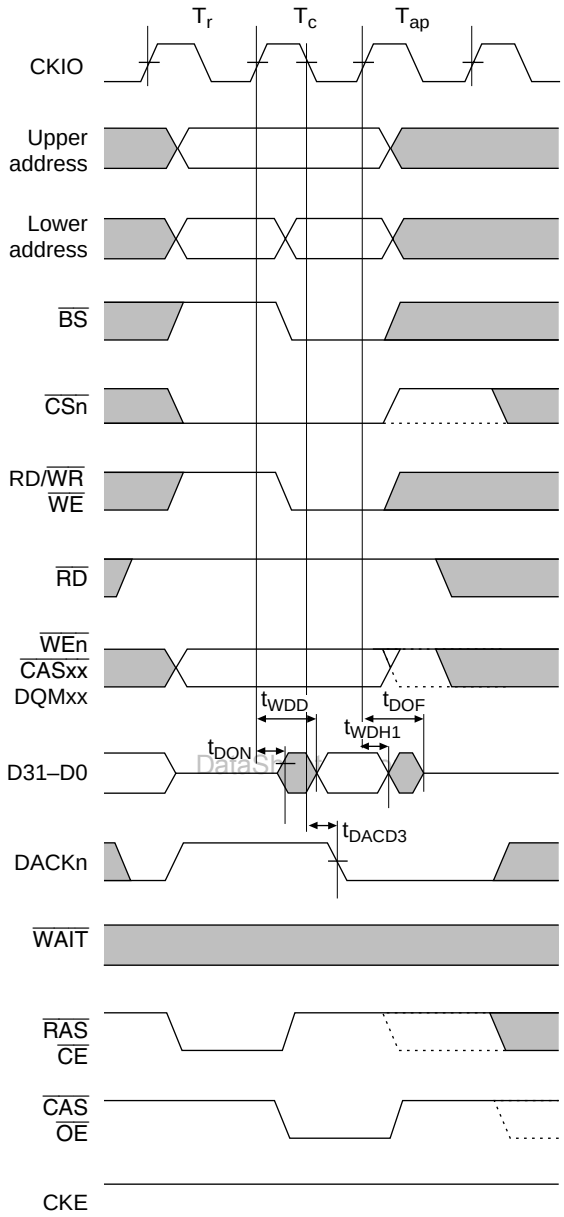
Note: A precharge cycle always precedes the self-refresh cycle by the number of cycles specified by TRP.

Figure 15.37 Synchronous DRAM Self-Refresh Cycle (TRAS = 2)



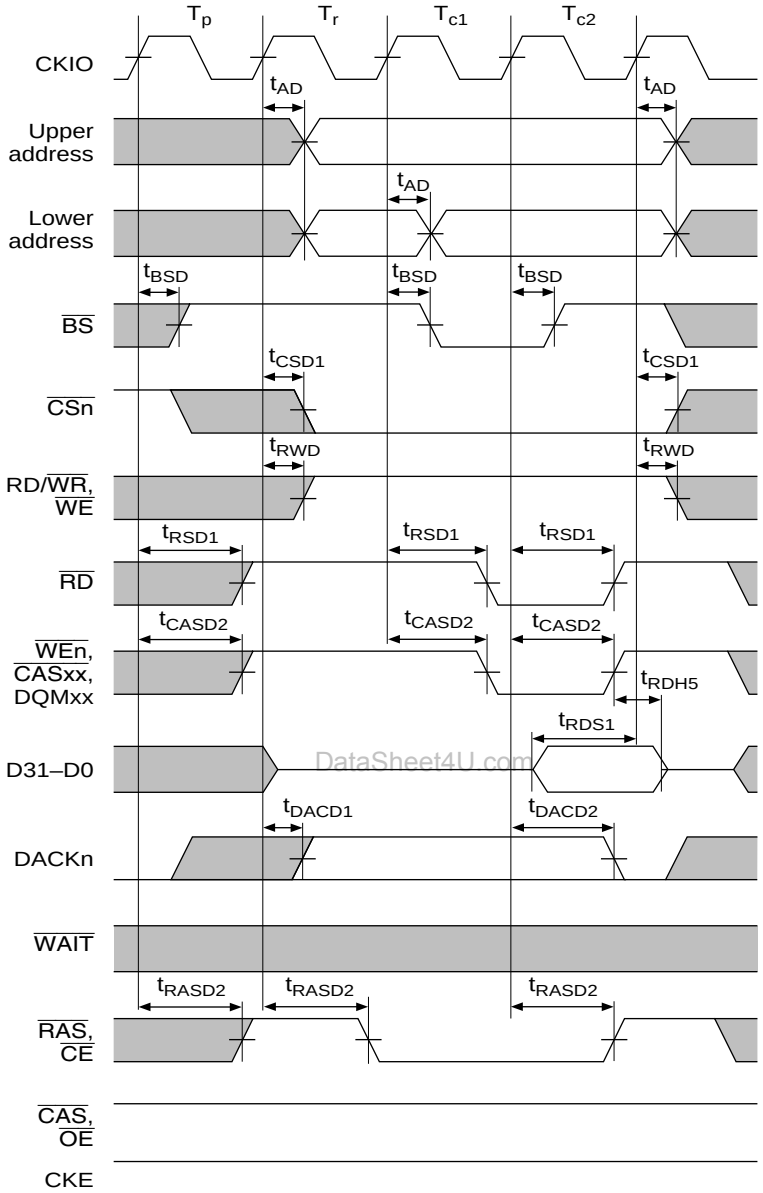
Notes: 1. The dotted line shows the waveform when synchronous DRAM in another CS space is accessed.
2. The DACKn waveform shown is for the case where active-high has been specified.

Figure 15.38 Synchronous DRAM Read Bus Cycle (RCD = 1 Cycle, CAS Latency = 1 Cycle, TRP = 1 Cycle, Bursts = 4, PLL Off)



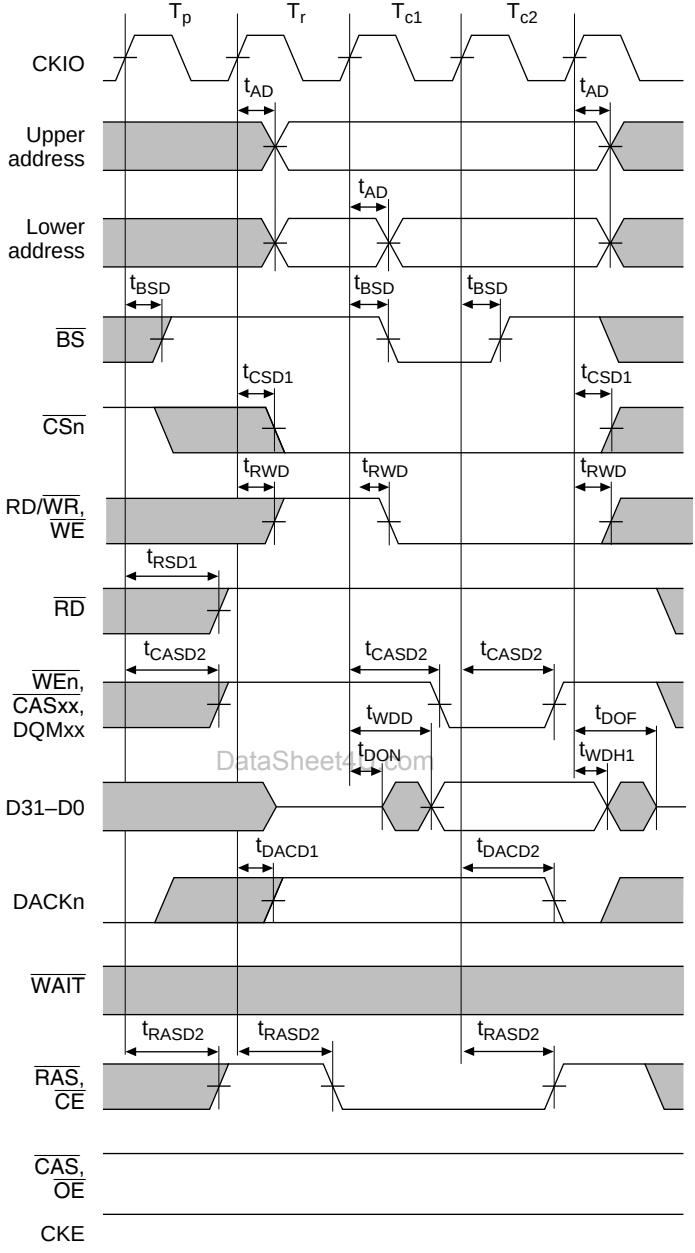
Notes: 1. Dotted lines show the waveforms when synchronous DRAM in another CS space is accessed.
2. The DACKn waveform shown is for the case where active-high has been specified.

Figure 15.39 Synchronous DRAM Write Bus Cycle (RCD = 1 Cycle, TRWL = 1 Cycle, PLL Off)



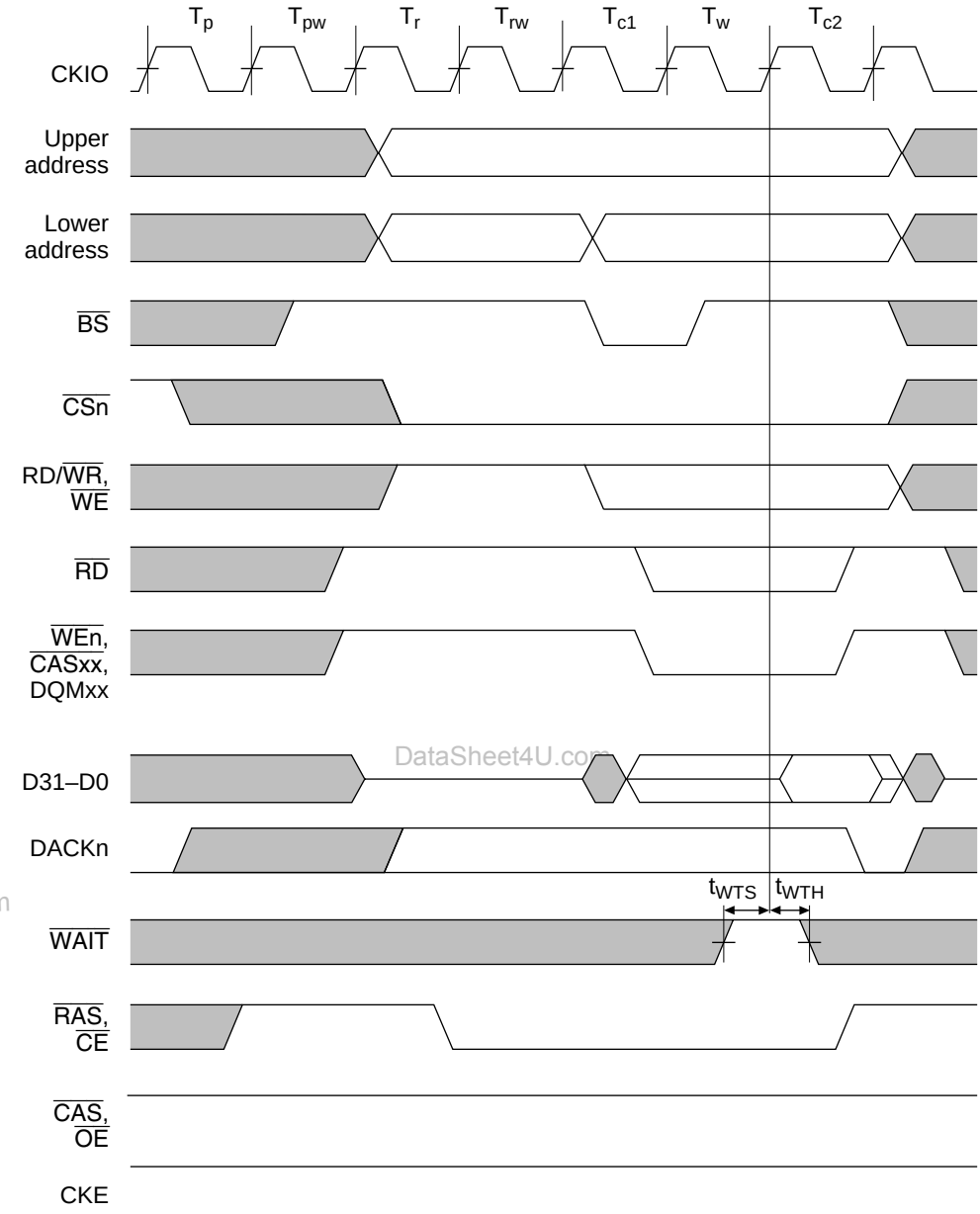
Notes: 1. t_{RDH5} is specified from the rise of $\overline{RD}$ or $\overline{CASxx}$, whichever is first.
2. The DACKn waveform shown is for the case where active-high has been specified.

Figure 15.40 DRAM Read Cycle (TRP = 1 Cycle, RCD = 1 Cycle, No Waits, PLL On)



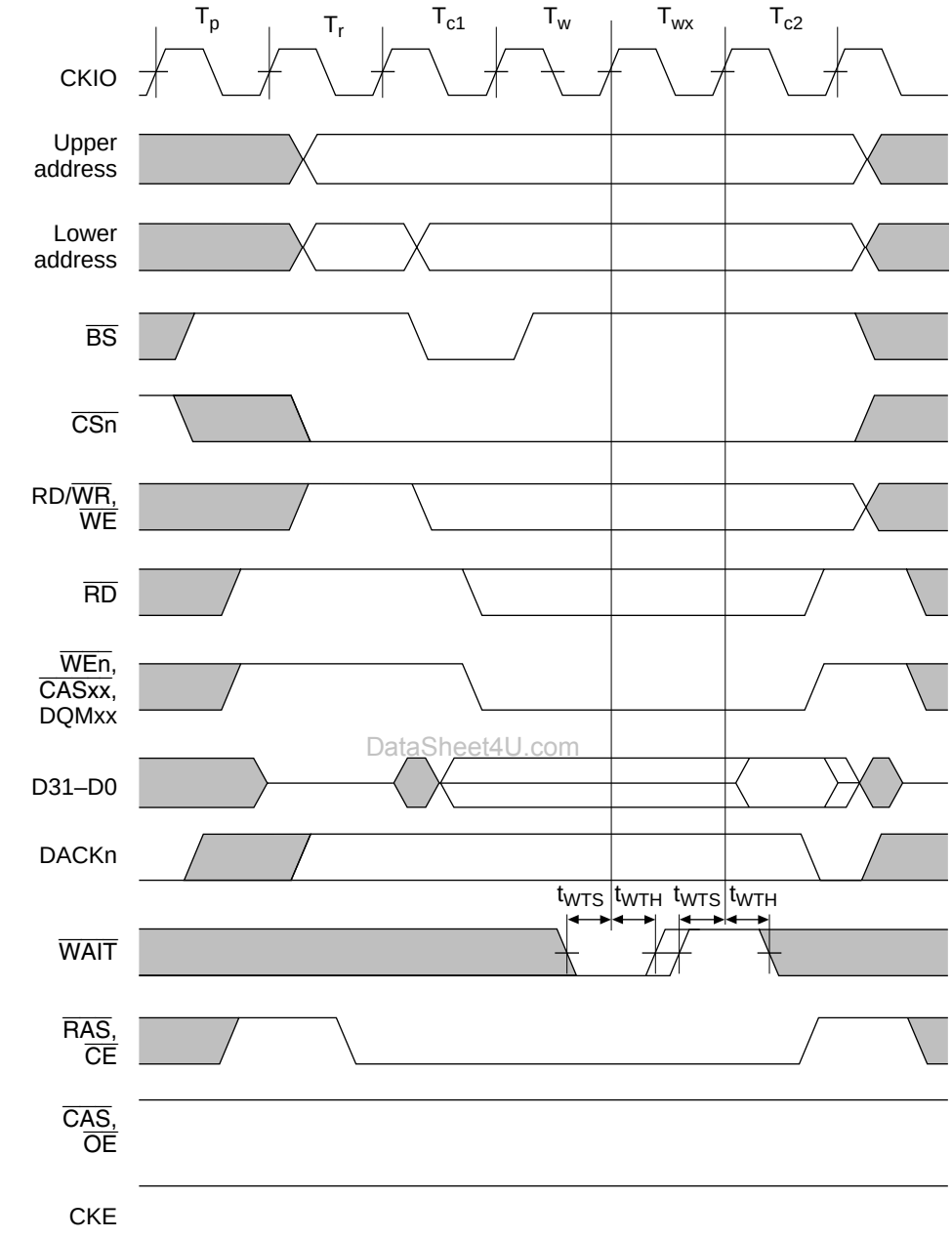
Note: The DACKn waveform shown is for the case where active-high has been specified.

Figure 15.41 DRAM Write Cycle
(TRP = 1 Cycle, RCD = 1 Cycle, No Waits, PLL On)



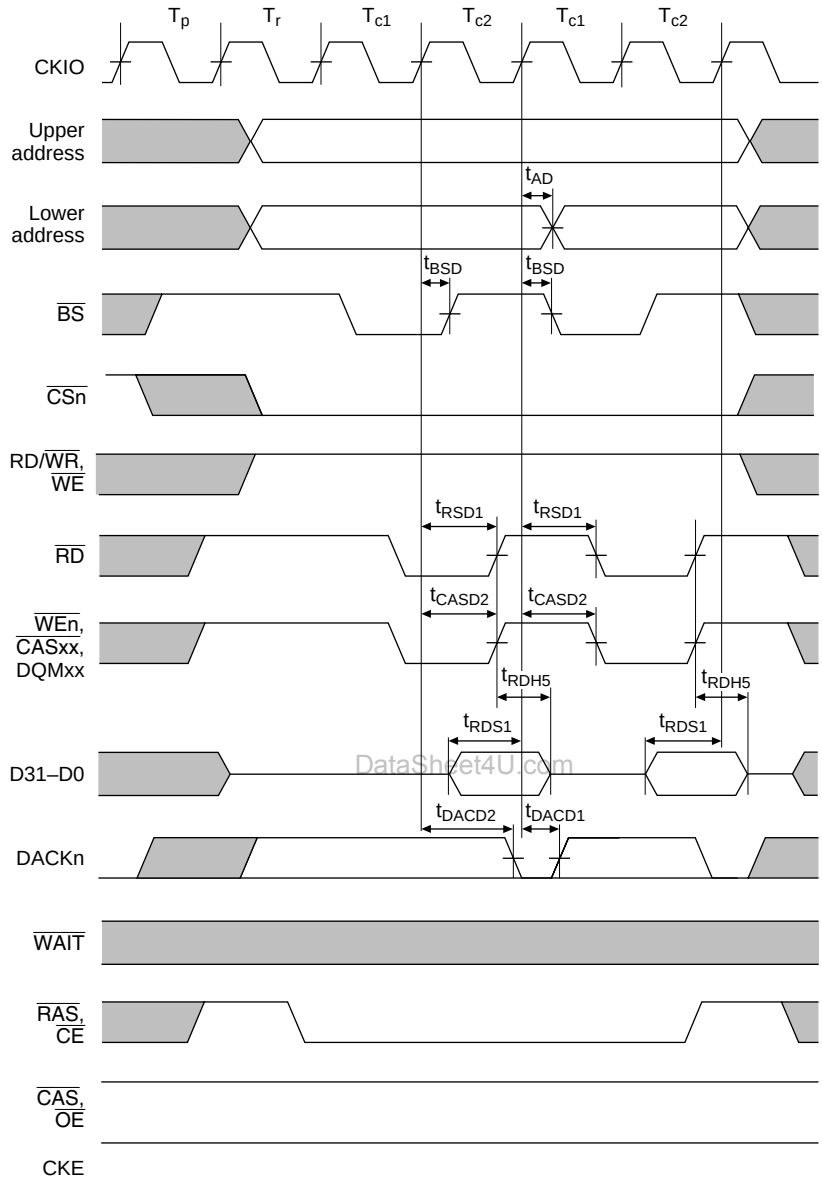
Note: The DACKn waveform shown is for the case where active-high has been specified.

Figure 15.42 DRAM Bus Cycle (TRP = 2 Cycles, RCD = 2 Cycles, 1 Wait)



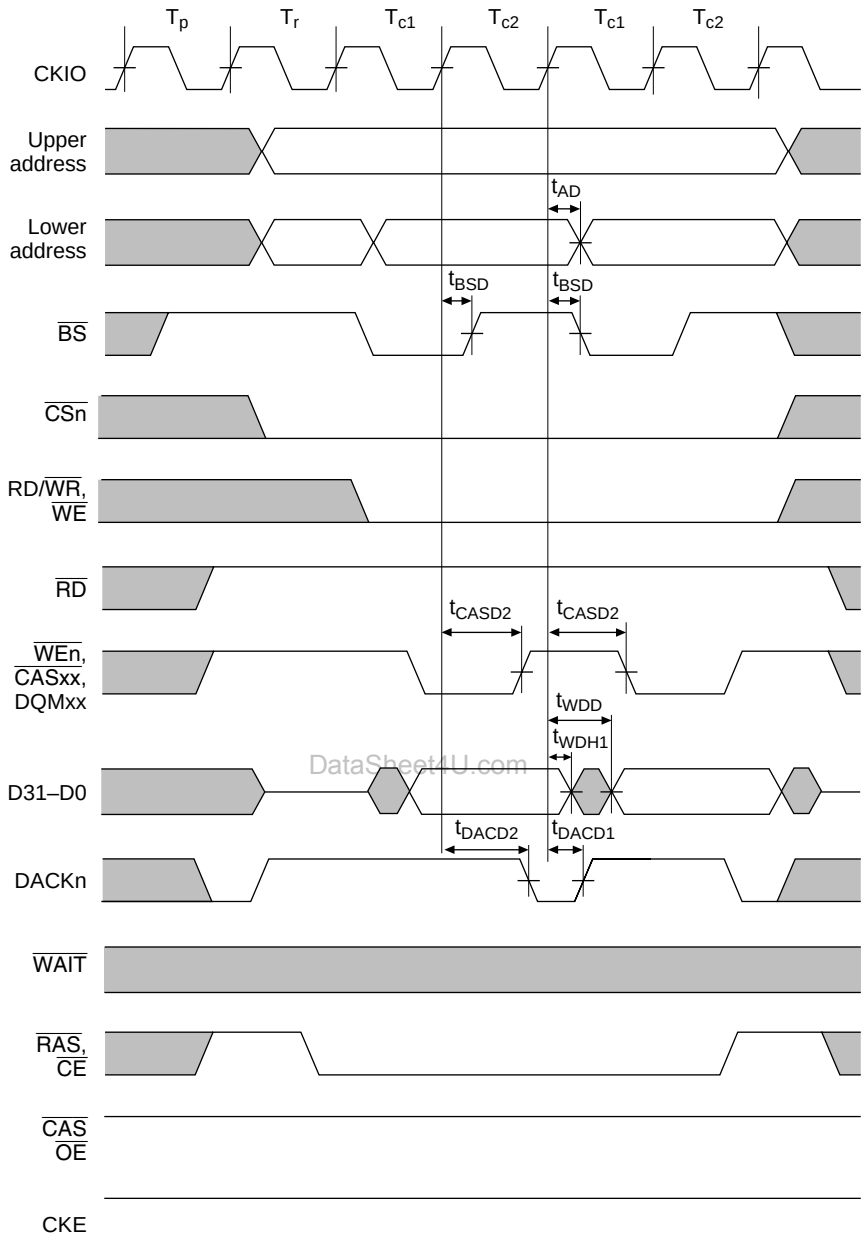
Note: The DACKn waveform shown is for the case where active-high has been specified.

Figure 15.43 DRAM Bus Cycle (TRP = 1 Cycle, RCD = 1 Cycle, External Wait Input)



Notes: 1. t_{RDH5} is specified from the rise of $\overline{RD}$ or $\overline{CASxx}$, whichever is first.
2. The DACKn waveform shown is for the case where active-high has been specified.

Figure 15.44 DRAM Burst Read Cycle
(TRP = 1 Cycle, RCD = 1 Cycle, No Waits, PLL On)



Note: The DACKn waveform shown is for the case where active-high has been specified.

Figure 15.45 DRAM Burst Write Cycle
(TRP = 1 Cycle, RCD = 1 Cycle, No Waits, PLL On)

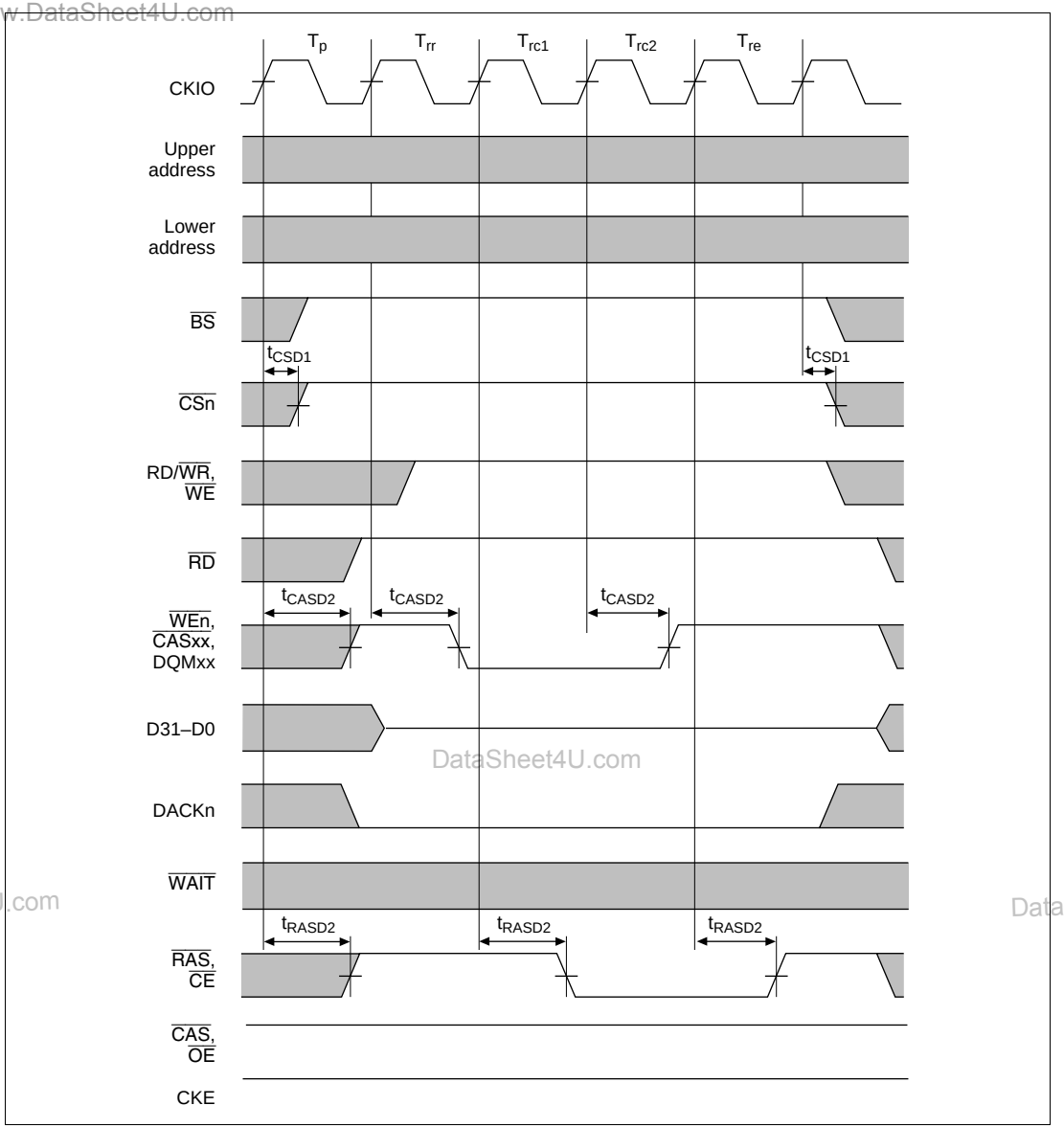
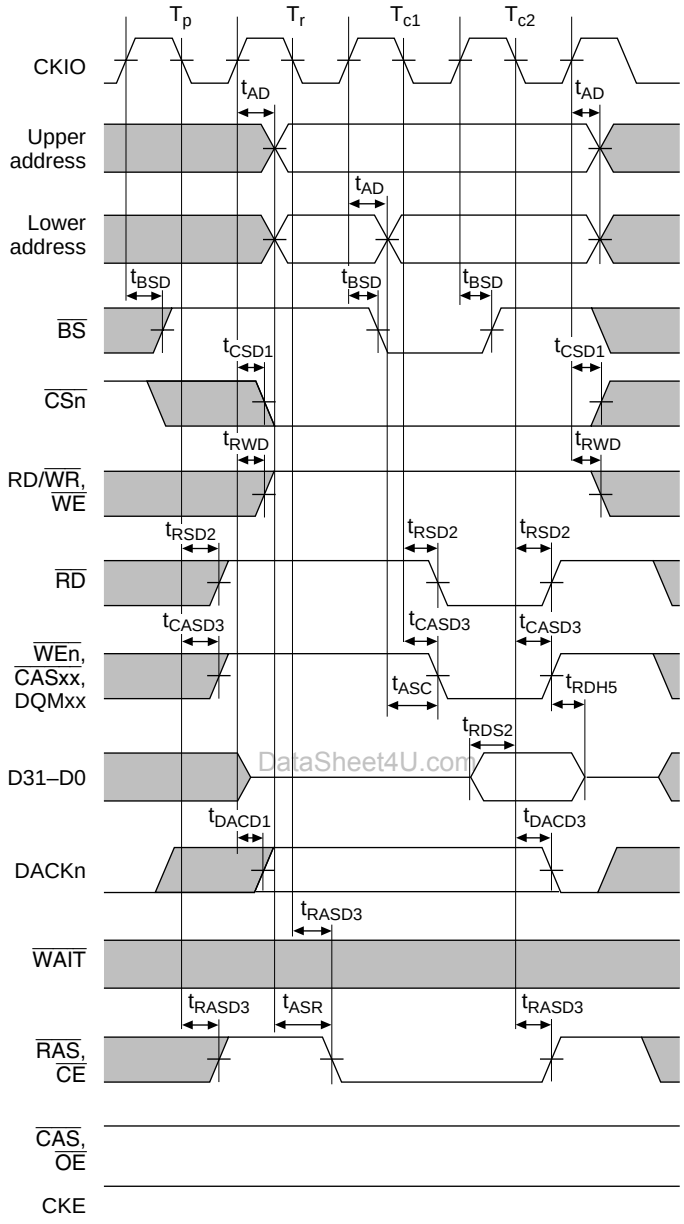
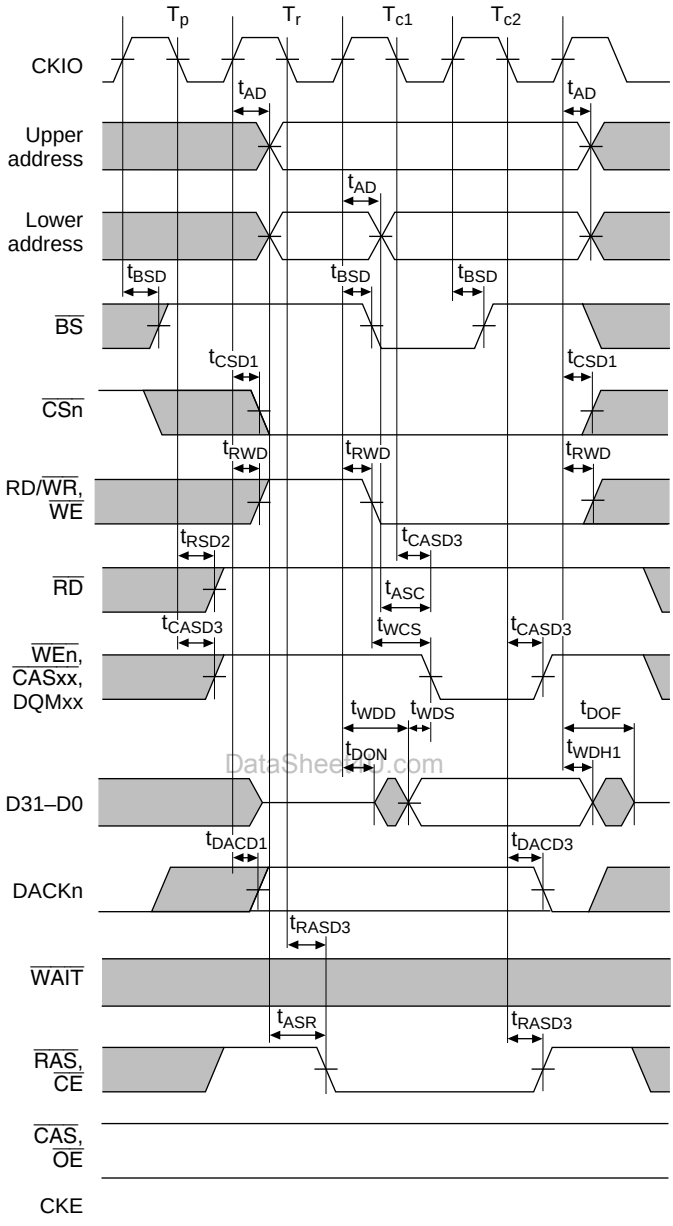


Figure 15.46 DRAM CAS-Before-RAS Refresh Cycle (TRP = 1 Cycle, TRAS = 2 Cycles, PLL On)



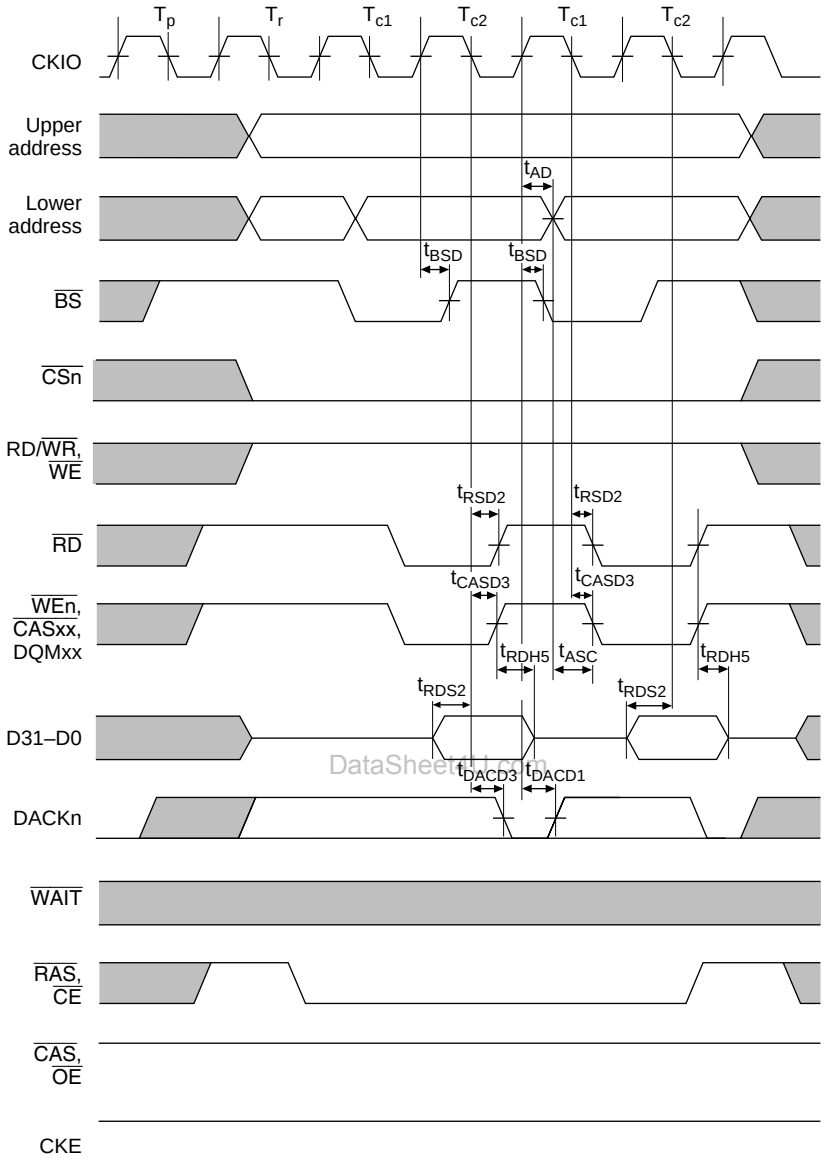
Notes: 1. t_{RDH5} is specified from the rise of $\overline{RD}$ or $\overline{CASxx}$, whichever is first.
2. The DACKn waveform shown is for the case where active-high has been specified.

Figure 15.47 DRAM Read Cycle (TRP = 1 Cycle, RCD = 1 Cycle, No Waits, PLL Off)



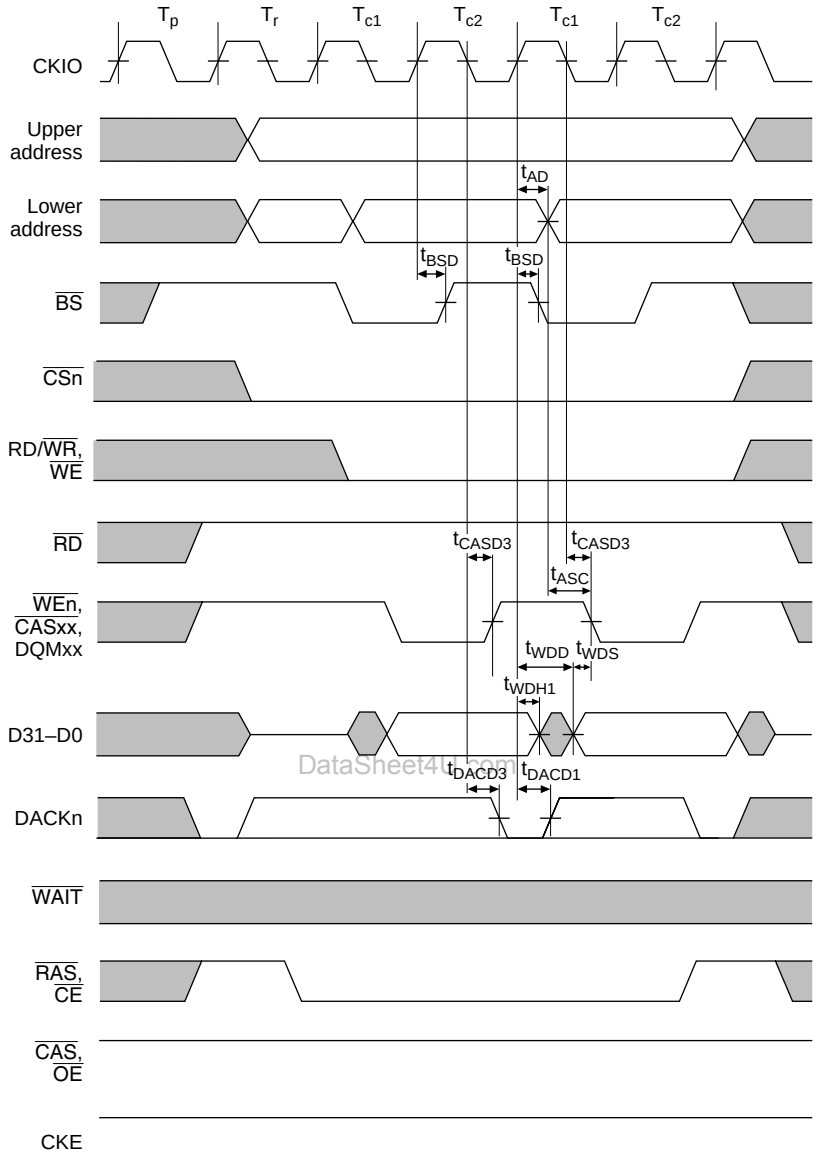
Note: The DACKn waveform shown is for the case where active-high has been specified.

Figure 15.48 DRAM Write Cycle (TRP = 1 Cycle, RCD = 1 Cycle, No Waits, PLL Off)



Notes: 1. t_{RDH5} is specified from the rise of $\overline{RD}$ or $\overline{CASxx}$, whichever is first.
2. The DACKn waveform shown is for the case where active-high has been specified.

Figure 15.49 DRAM Burst Read Cycle
(TRP = 1 Cycle, RCD = 1 Cycle, No Waits, PLL Off)



Note: The DACKn waveform shown is for the case where active-high has been specified.

Figure 15.50 DRAM Burst Write Cycle (TRP = 1 Cycle, RCD = 1 Cycle, No Waits, PLL Off)

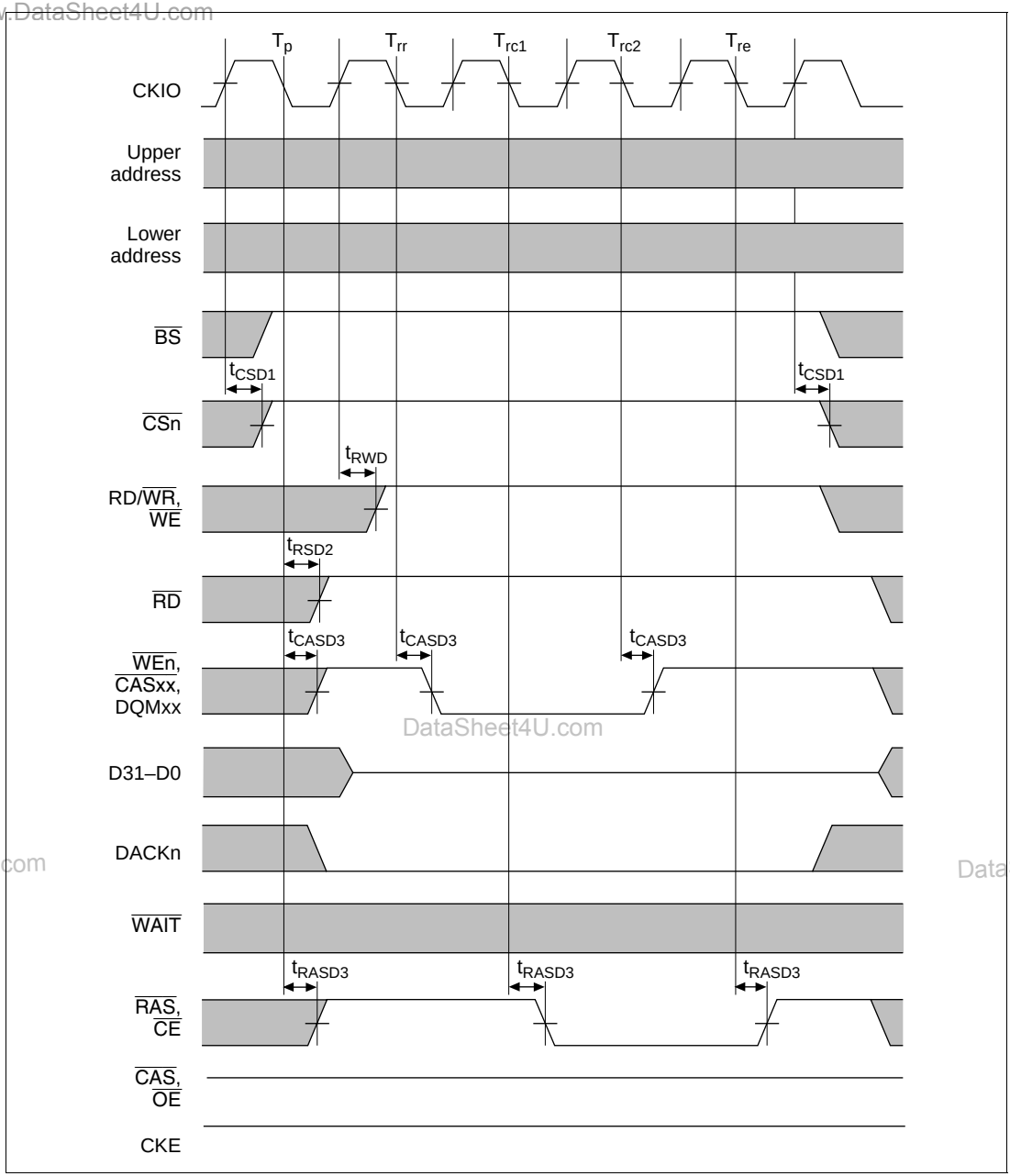
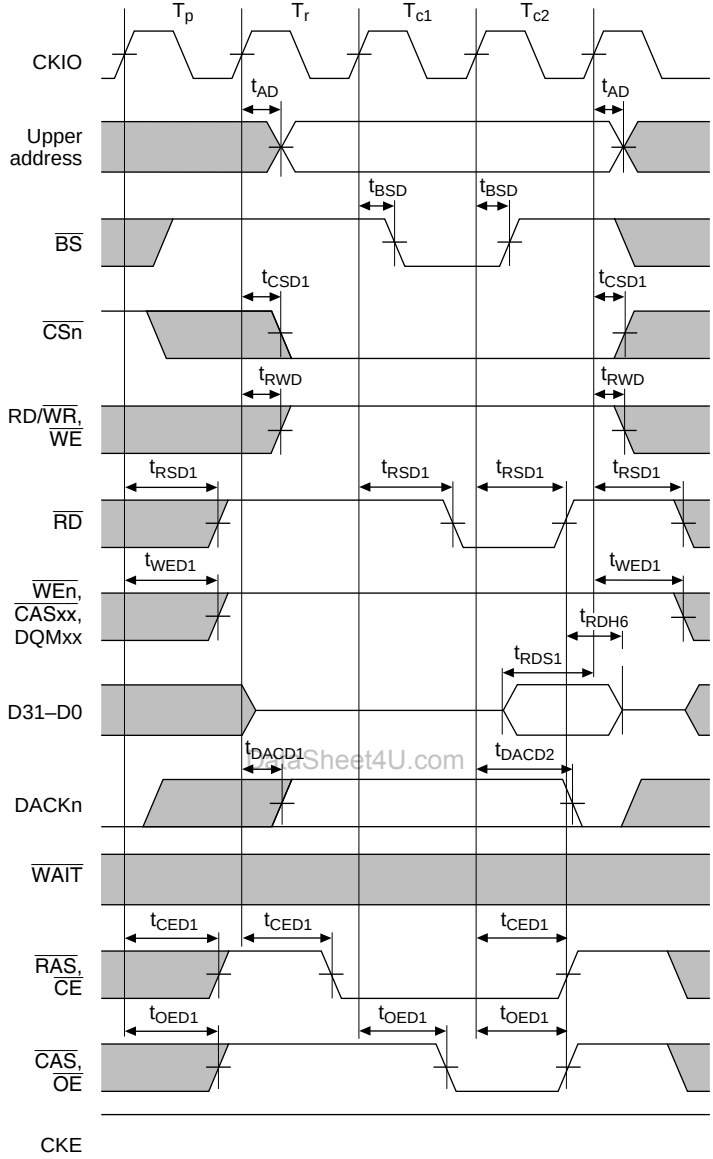
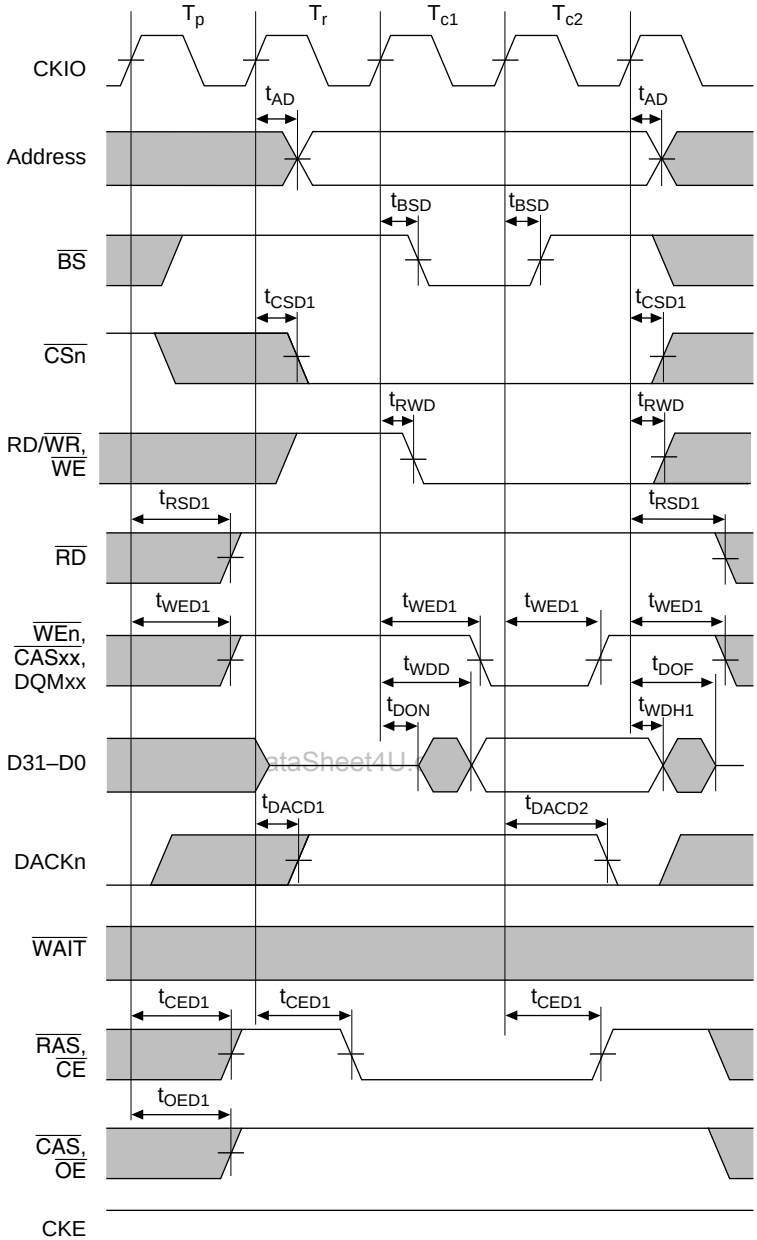


Figure 15.51 DRAM CAS-Before-RAS Refresh Cycle (TRP = 1 Cycle, TRAS = 2 Cycles, PLL Off)



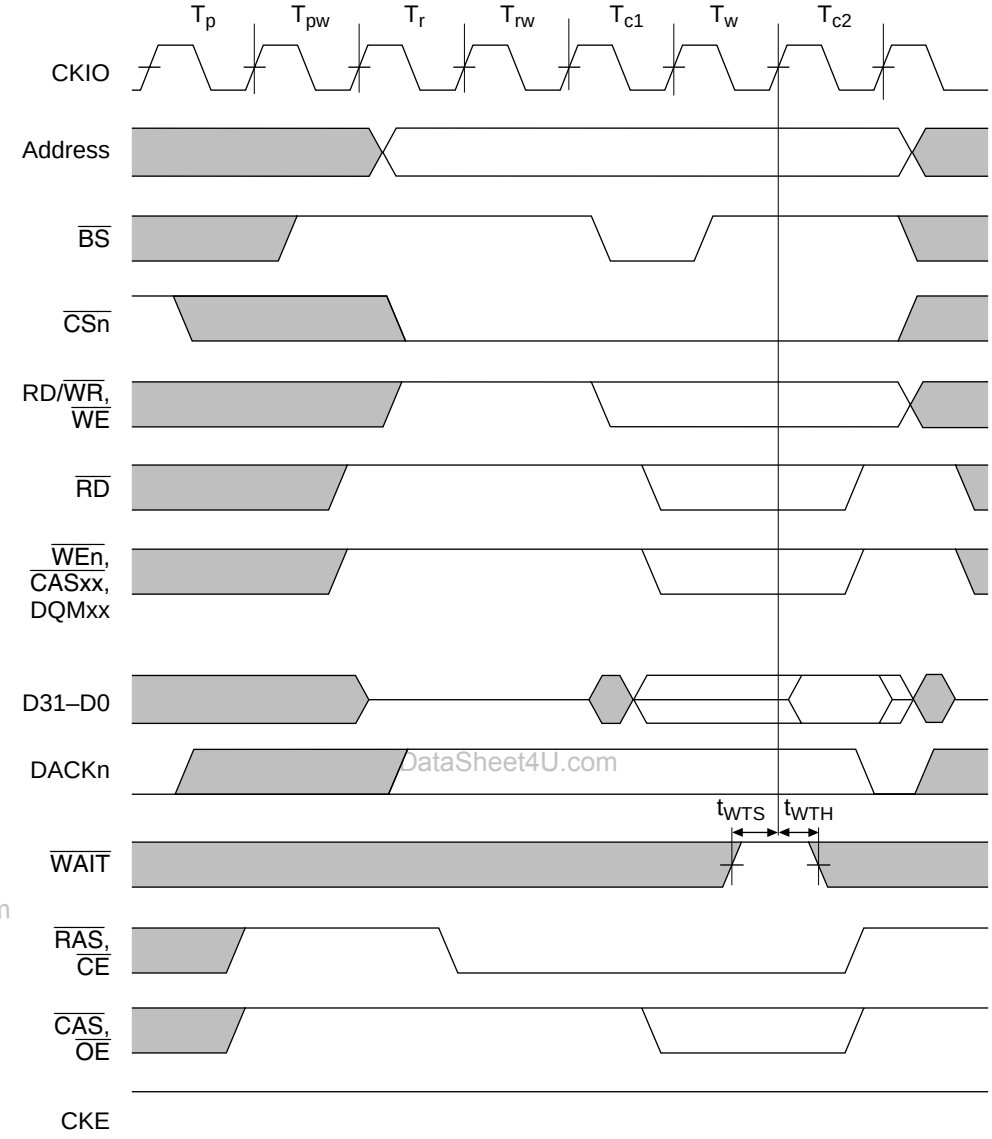
Note: The DACKn waveform shown is for the case where active-high has been specified.

Figure 15.52 Pseudo-SRAM Read Cycle
(PLL On, TRP = 1 Cycle, RCD = 1 Cycle, No Waits)



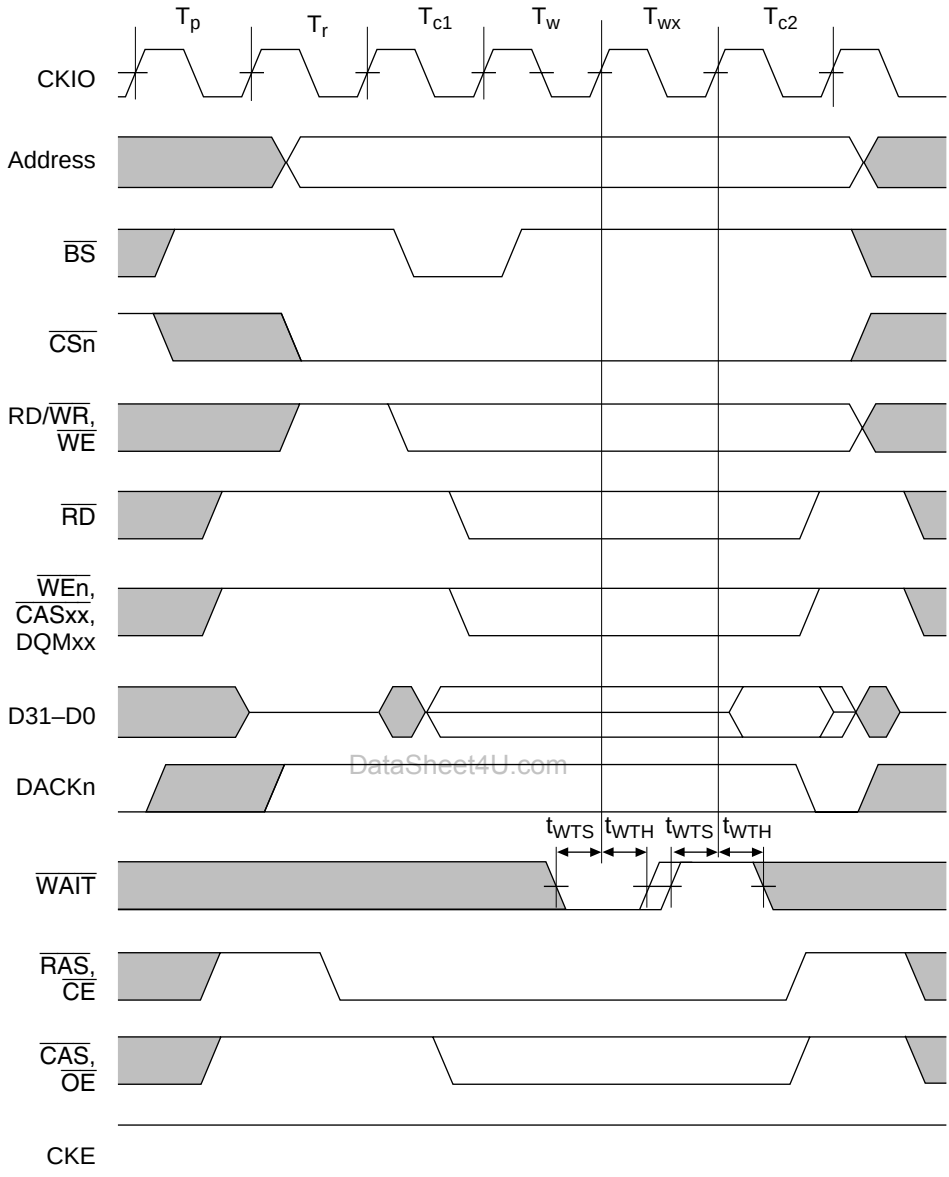
Note: The DACKn waveform shown is for the case where active-high has been specified.

Figure 15.53 Pseudo-SRAM Write Cycle
(PLL On, TRP = 1 Cycle, RCD = 1 Cycle, No Waits)



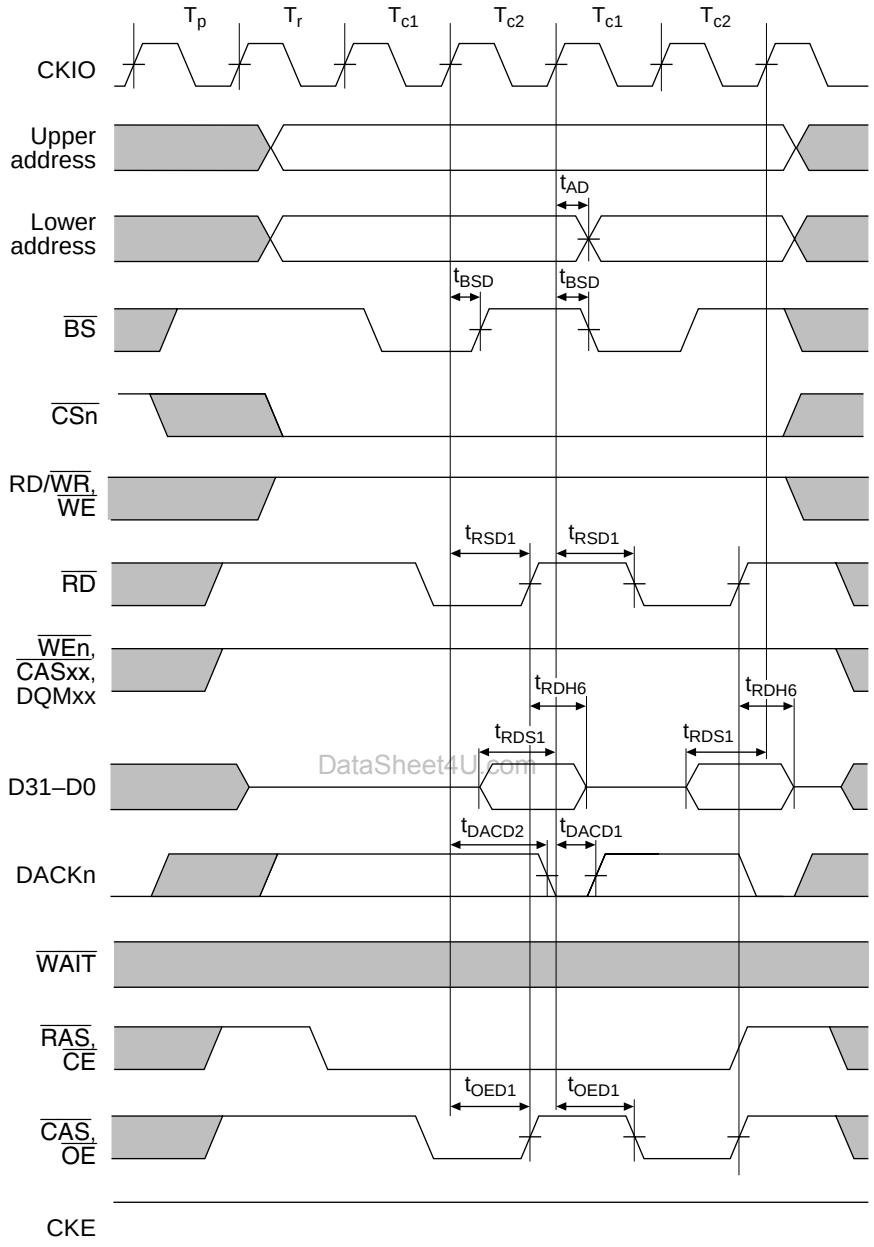
Note: The DACKn waveform shown is for the case where active-high has been specified.

Figure 15.54 Pseudo-SRAM Bus Cycle (TRP = 2 Cycles, RCD = 2 Cycles, 1 Wait)



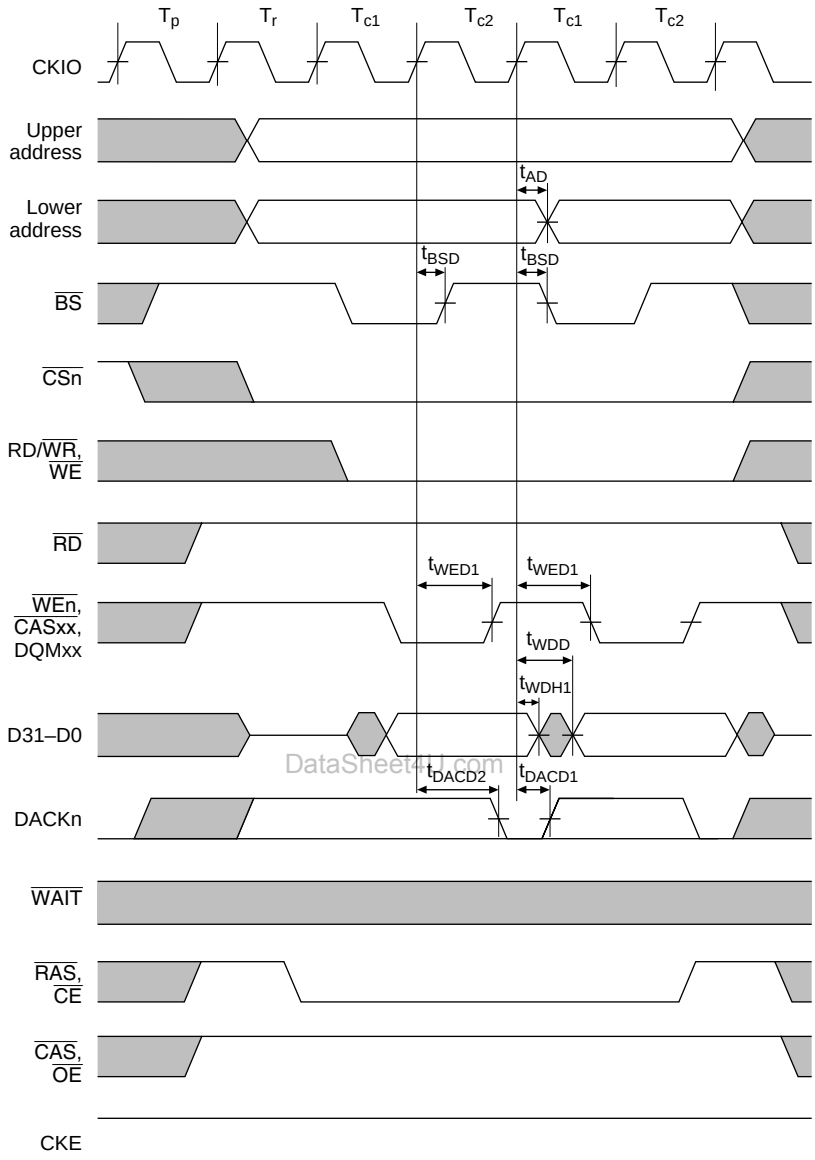
Note: The DACKn waveform shown is for the case where active-high has been specified.

Figure 15.55 Pseudo-SRAM Bus Cycle
(TRP = 1 Cycle, RCD = 1 Cycle, External Wait Input)



Note: The DACKn waveform shown is for the case where active-high has been specified.

Figure 15.56 Pseudo-SRAM Read Cycle
(Static Column Mode, PLL On, TRP = 1 Cycle, RCD = 1 Cycle, No Waits)



Note: The DACKn waveform shown is for the case where active-high has been specified.

Figure 15.57 Pseudo-SRAM Write Cycle (Static Column Mode, PLL On, TRP = 1 Cycle, RCD = 1 Cycle, No Waits)

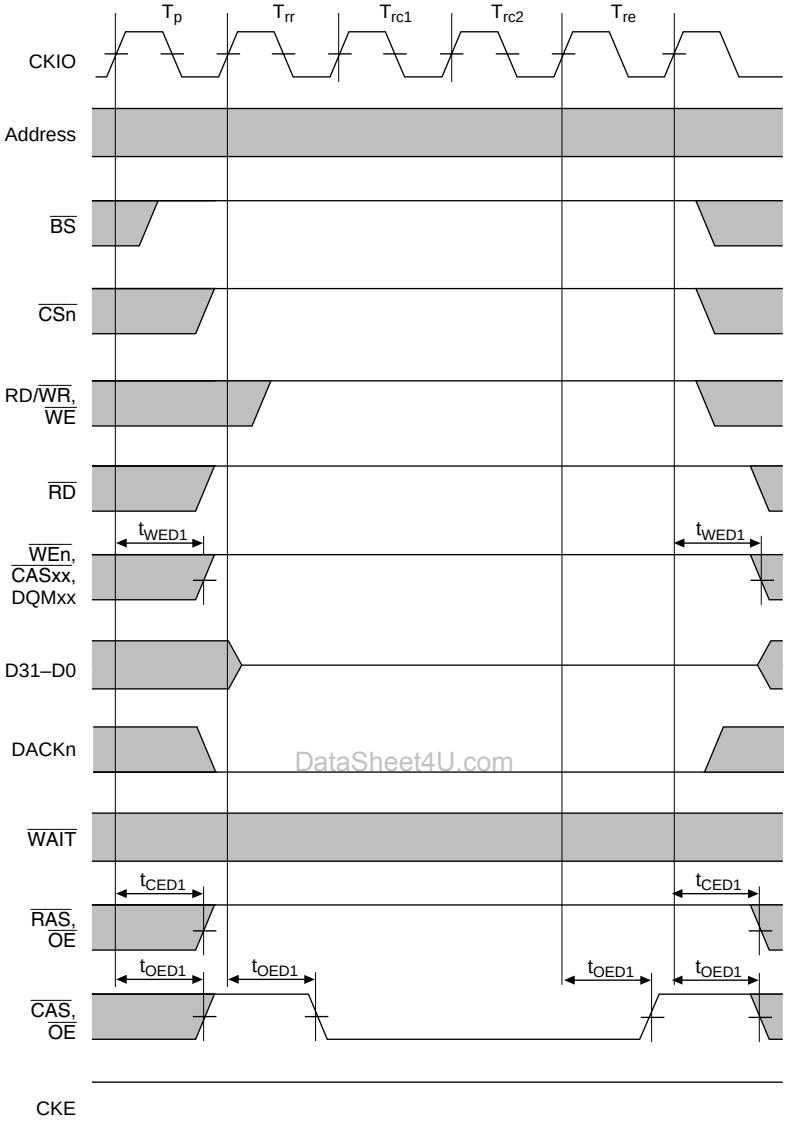


Figure 15.58 Pseudo-SRAM Auto-Refresh Cycle (PLL On, TRP = 1 Cycle, TRAS = 2 Cycles)

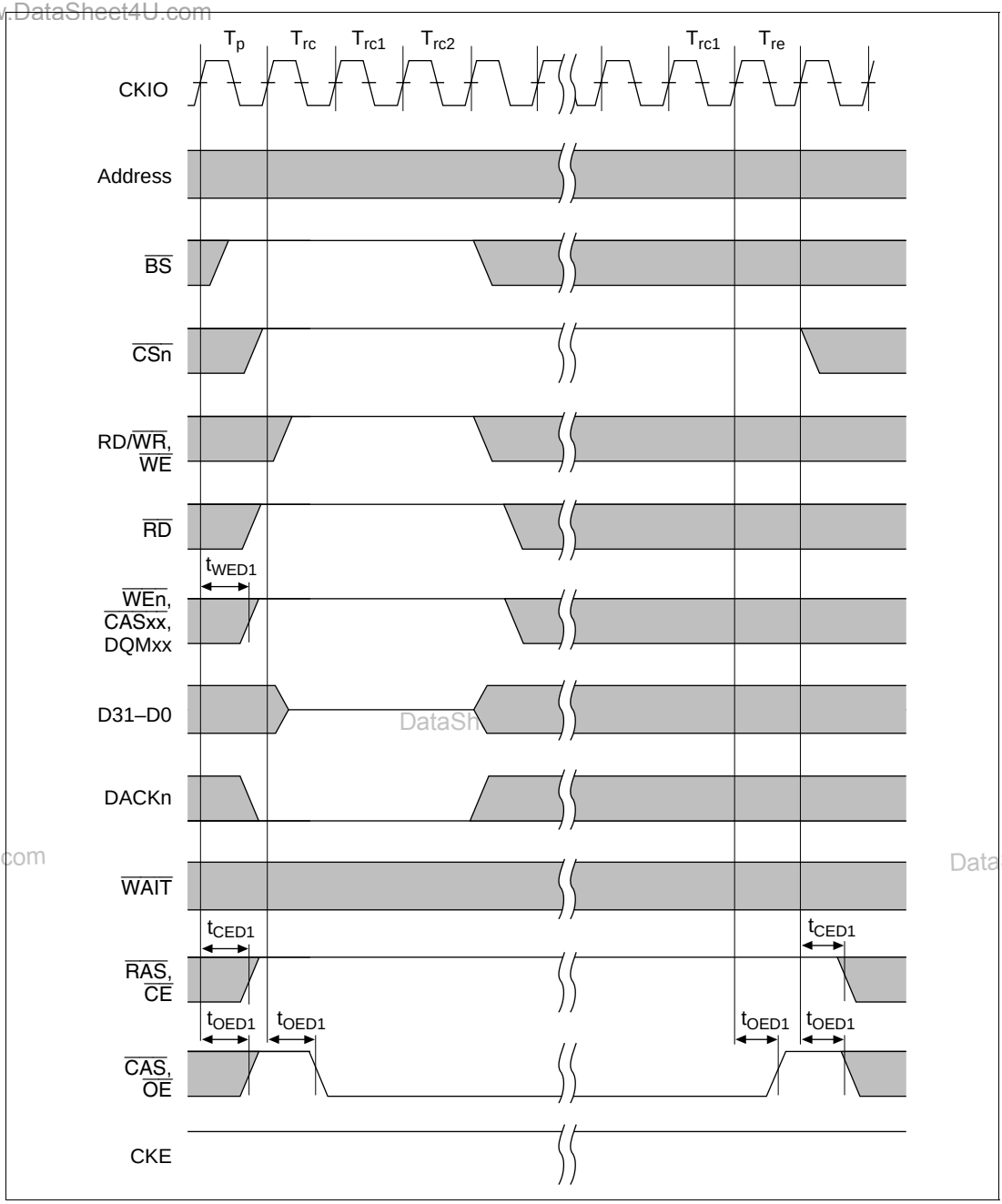
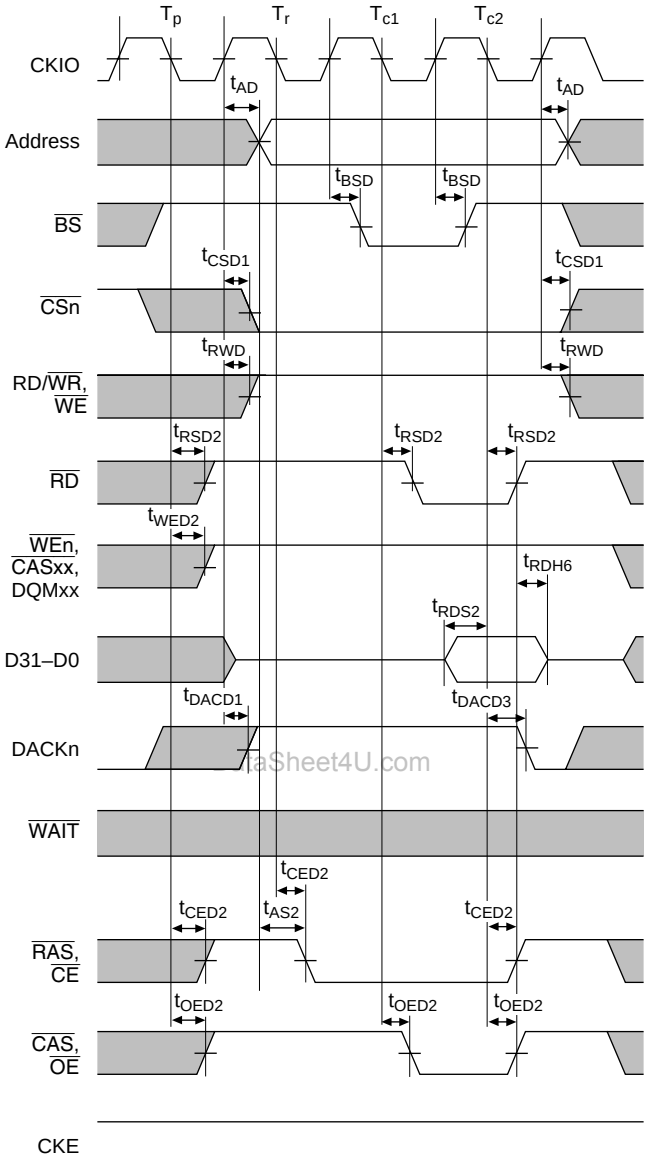
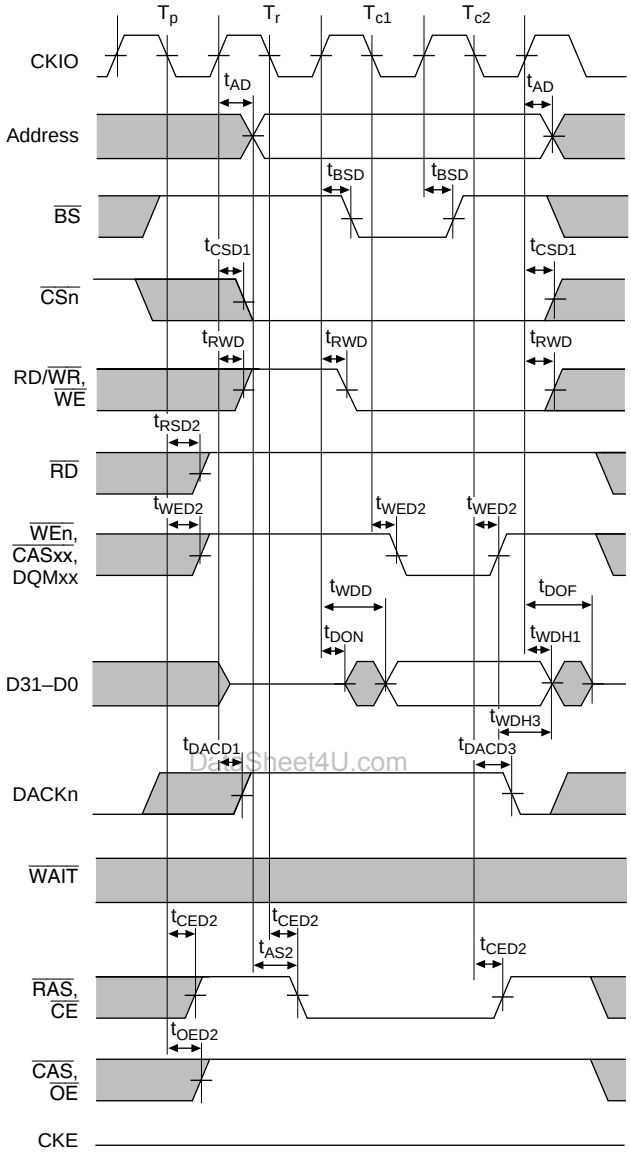


Figure 15.59 Pseudo-SRAM Self-Refresh Cycle (PLL On, TRP = 1 Cycle, TRAS = 2 Cycles)



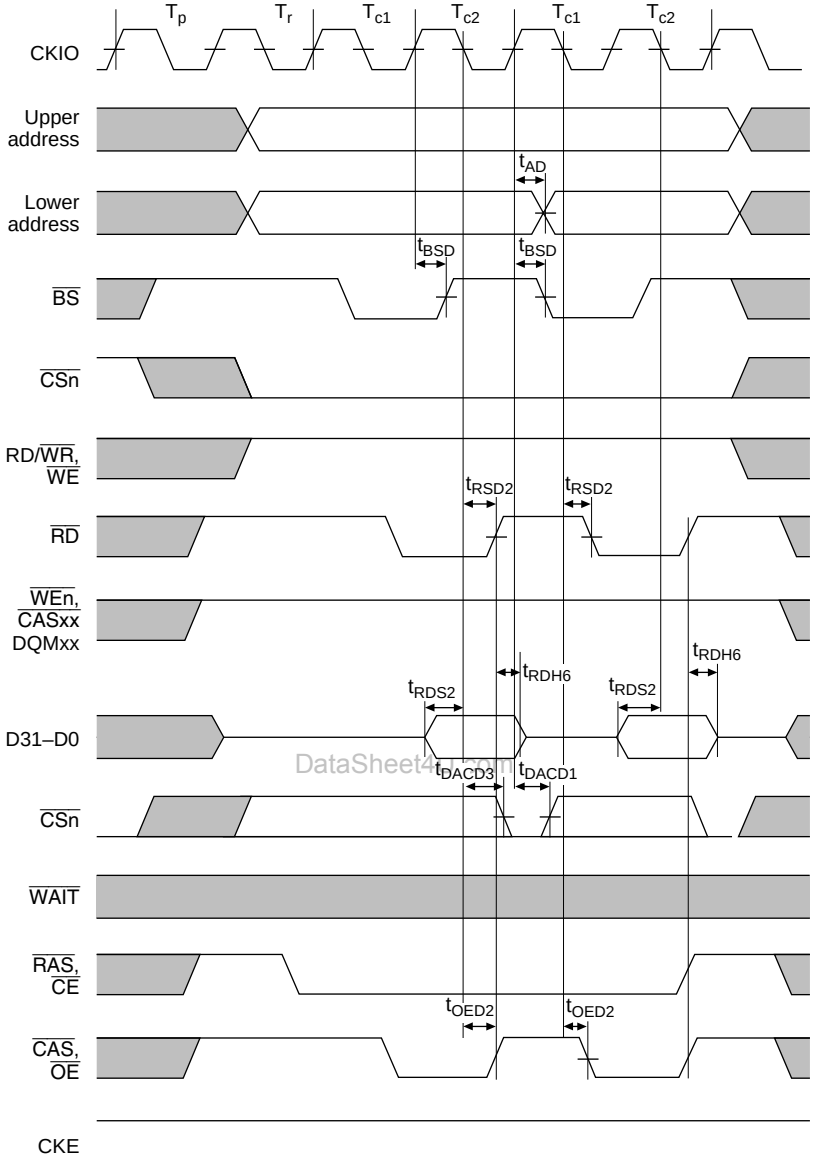
Note: The DACKn waveform shown is for the case where active-high has been specified.

Figure 15.60 Pseudo-SRAM Read Cycle
(PLL Off, TRP = 1 Cycle, RCD = 1 Cycle, No Waits)



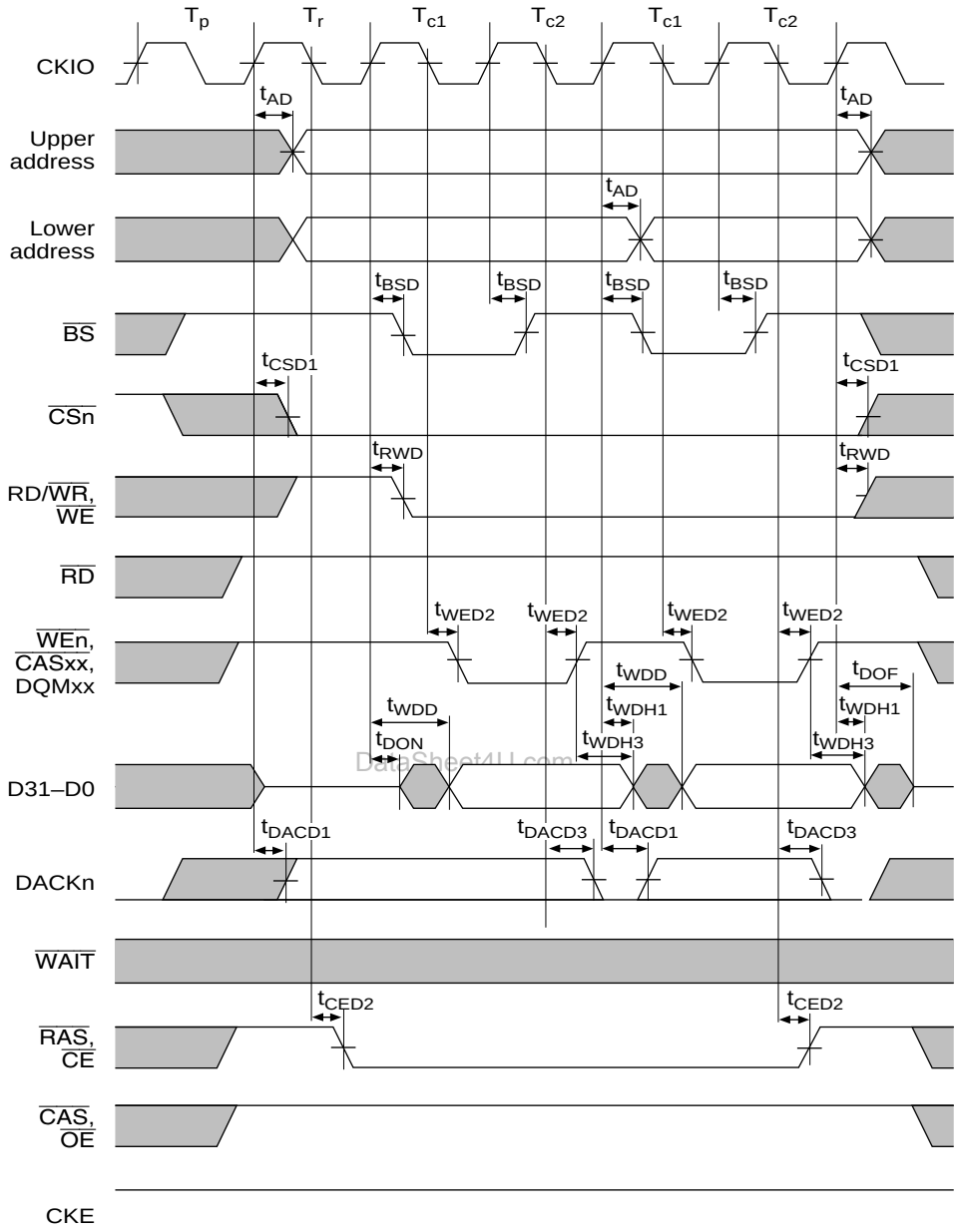
Note: The DACKn waveform shown is for the case where active-high has been specified.

Figure 15.61 Pseudo-SRAM Write Cycle (PLL Off, TRP = 1 Cycle, RCD = 1 Cycle, No Waits)



Note: The DACKn waveform shown is for the case where active-high has been specified.

Figure 15.62 Pseudo-SRAM Read Cycle
(Static Column Mode, PLL Off, TRP = 1 Cycle, RCD = 1 Cycle, No Waits)



Note: The DACKn waveform shown is for the case where active-high has been specified.

Figure 15.63 Pseudo-SRAM Write Cycle
(Static Column Mode, PLL Off, TRP = 1 Cycle, RCD = 1 Cycle, No Waits)

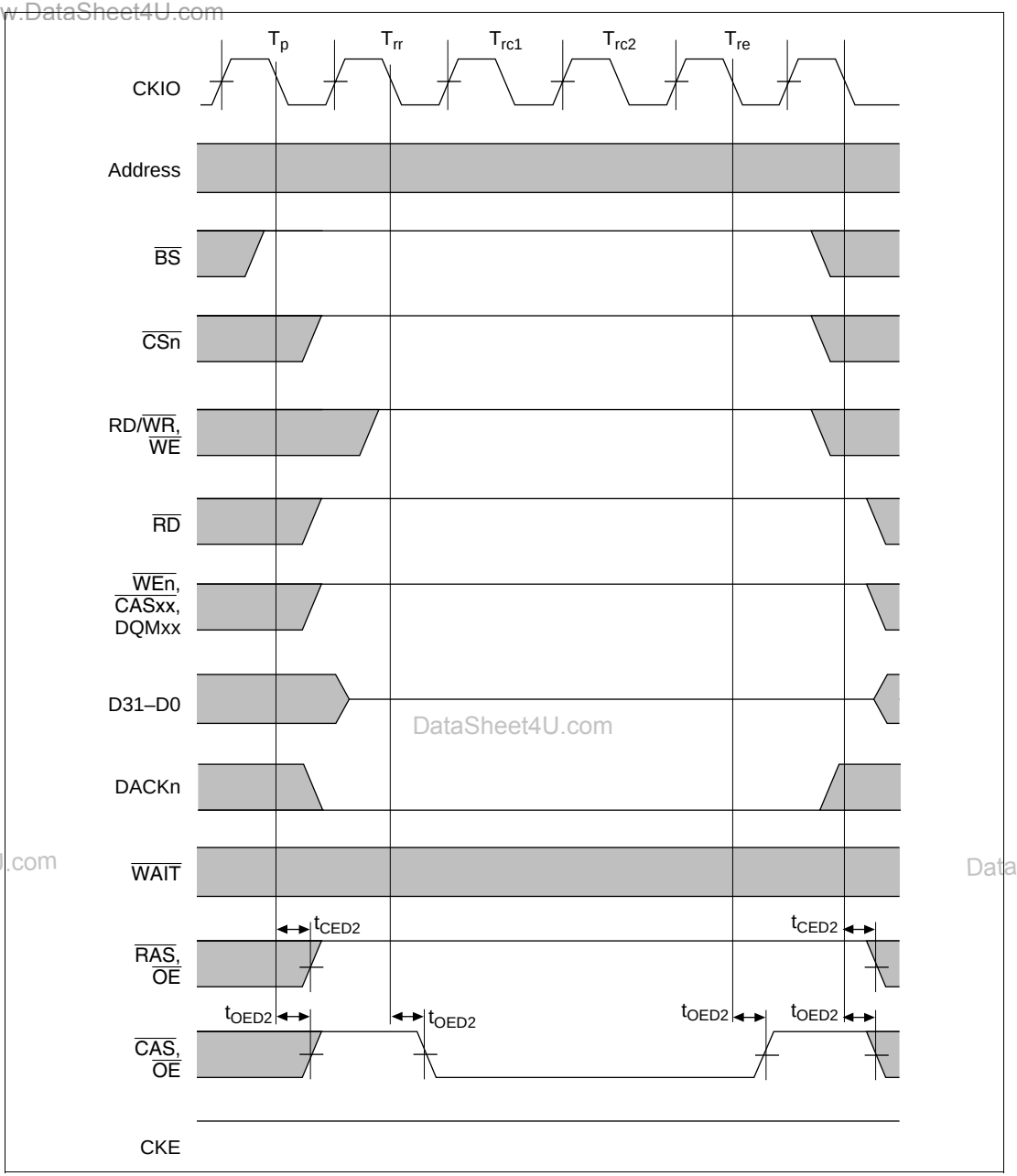


Figure 15.64 Pseudo-SRAM Auto-Refresh Cycle (PLL Off, TRP = 1 Cycle, TRAS = 2 Cycles)

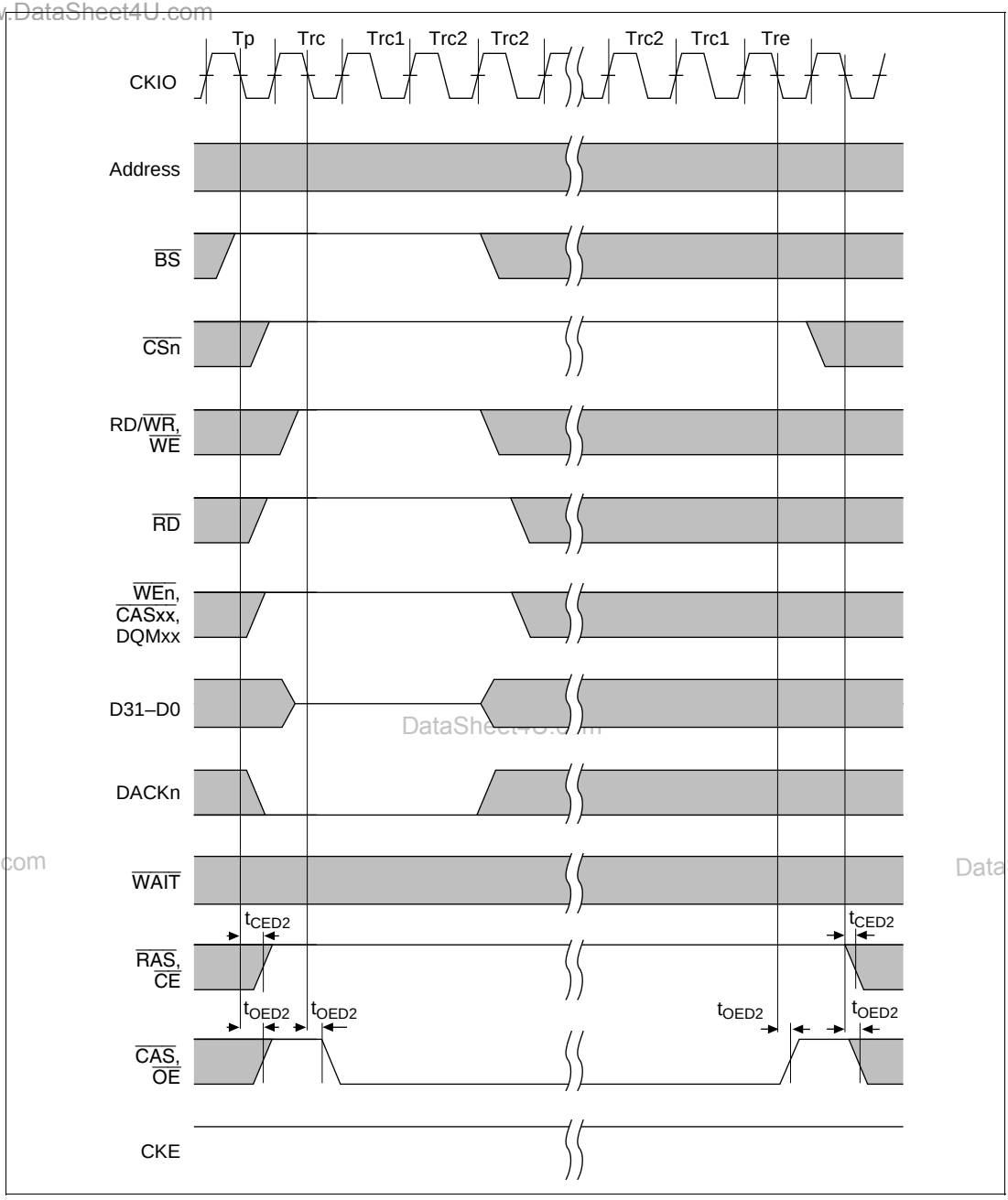
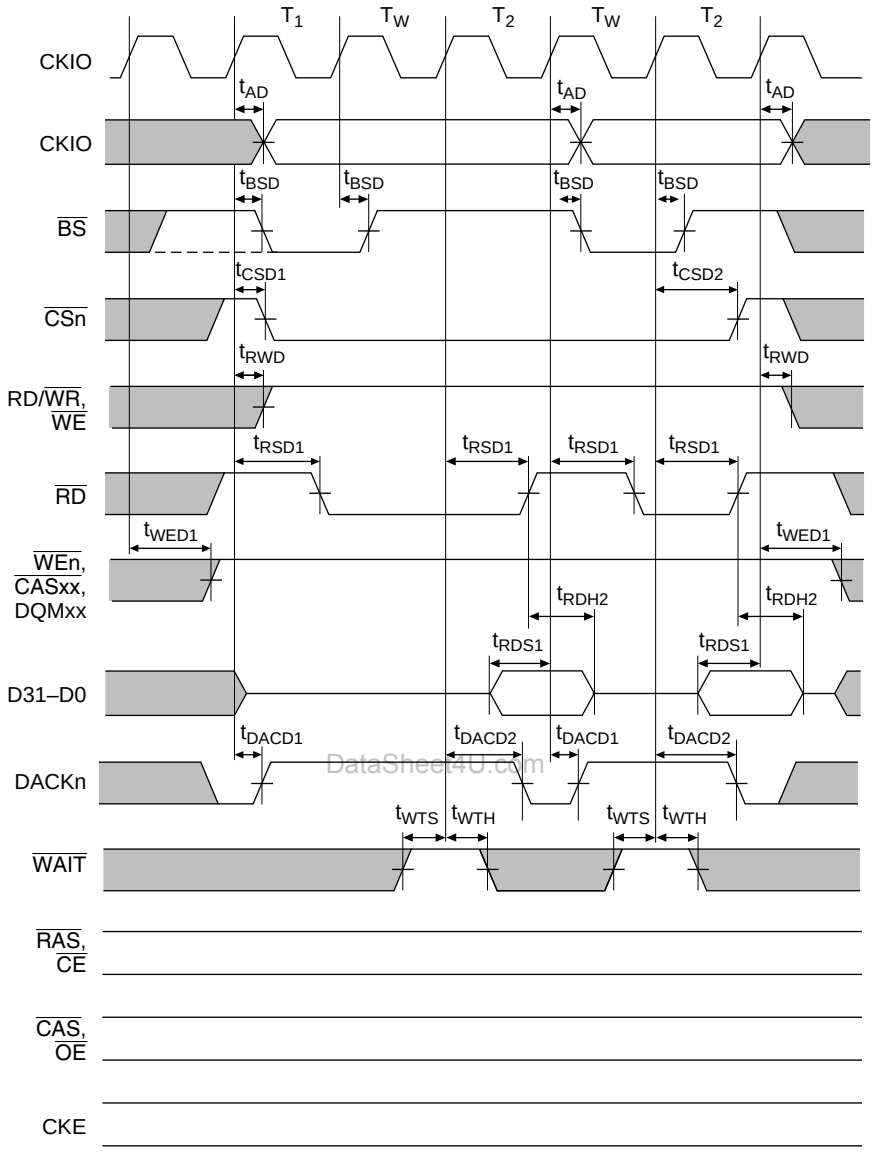
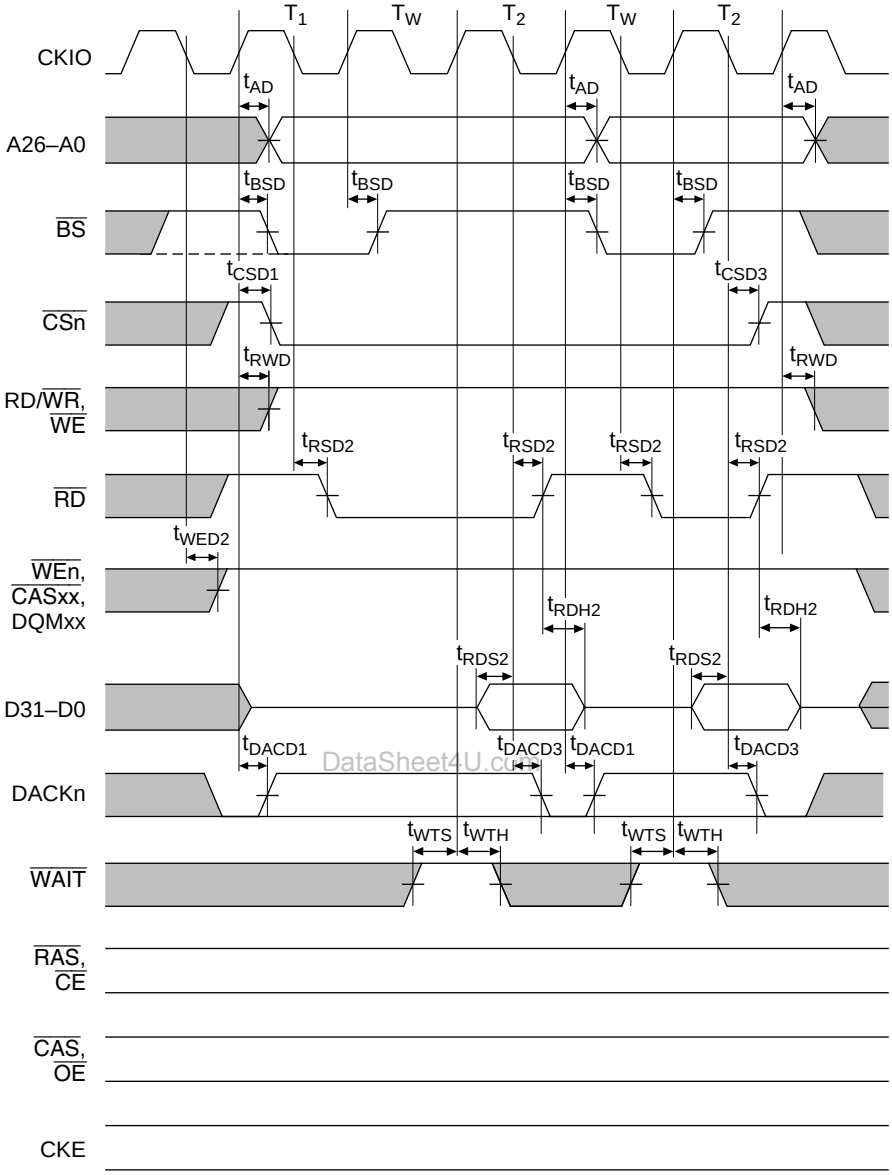


Figure 15.65 Pseudo-SRAM Self-Refresh Cycle (PLL Off, TRP = 1 Cycle, TRAS = 2 Cycles)



Note: The DACKn waveform shown is for the case where active-high has been specified.

Figure 15.66 Burst ROM Read Cycle (PLL On, 1 Wait)



Note: The DACKn waveform shown is for the case where active-high has been specified.

Figure 15.67 Burst ROM Read Cycle (PLL Off, 1 Wait)

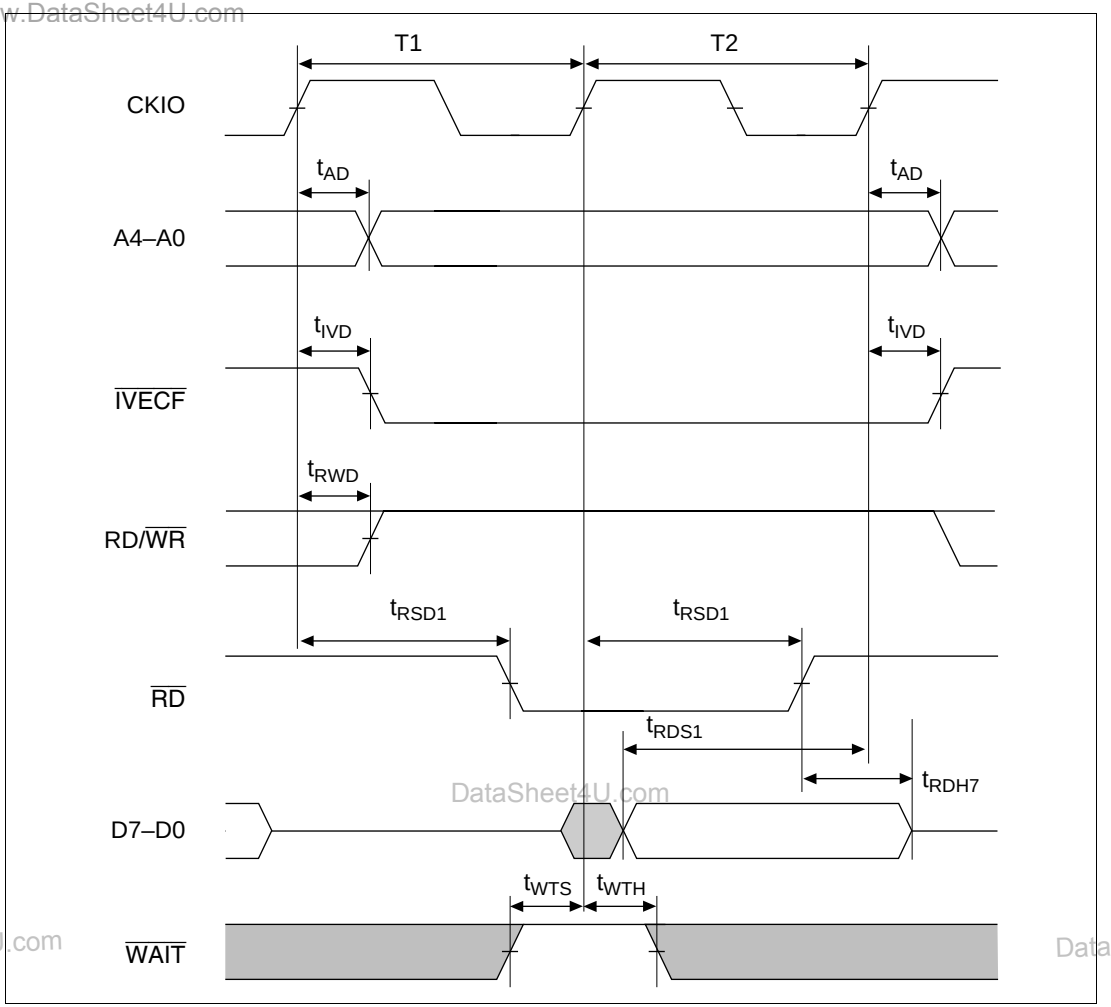


Figure 15.68 Interrupt Vector Fetch Cycle (PLL On, No Waits)

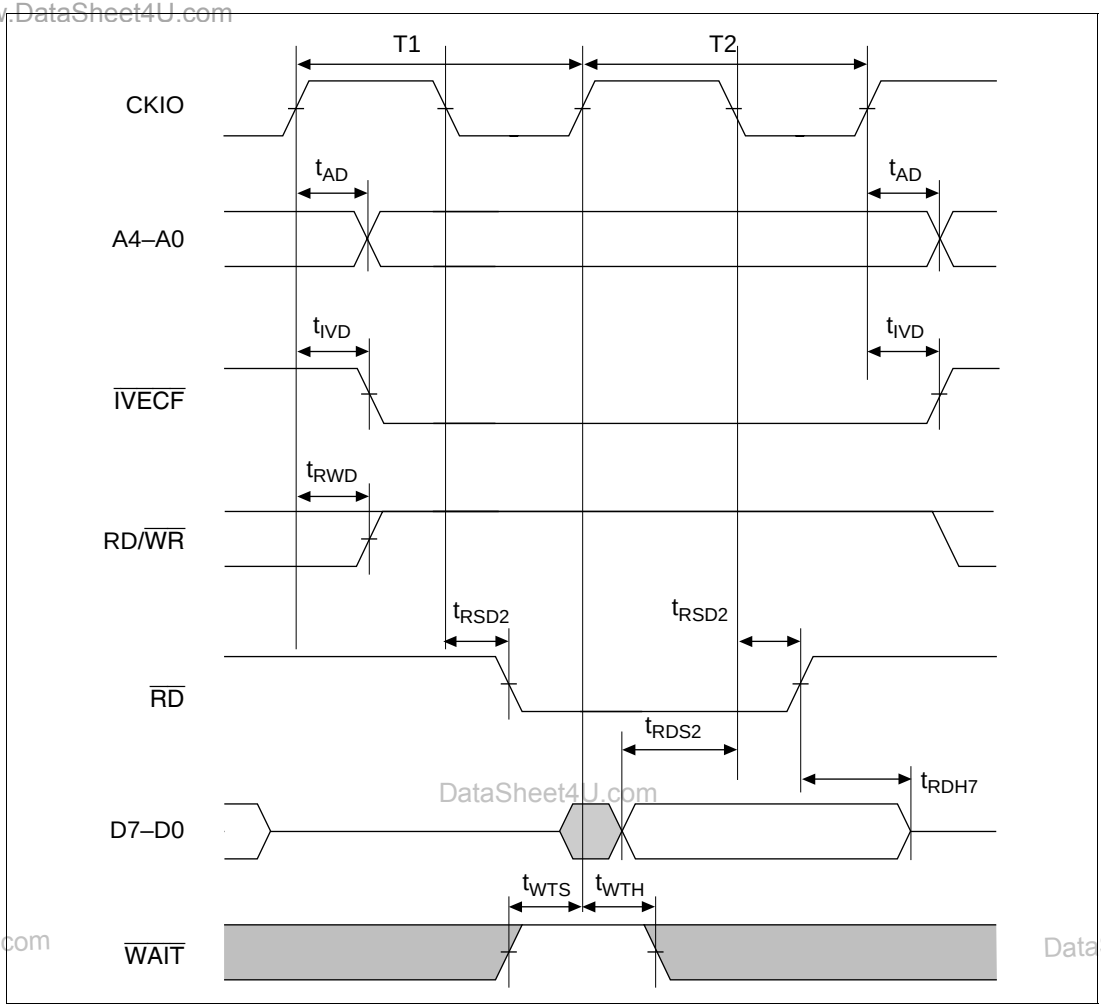


Figure 15.69 Interrupt Vector Fetch Cycle (PLL Off, No Waits)

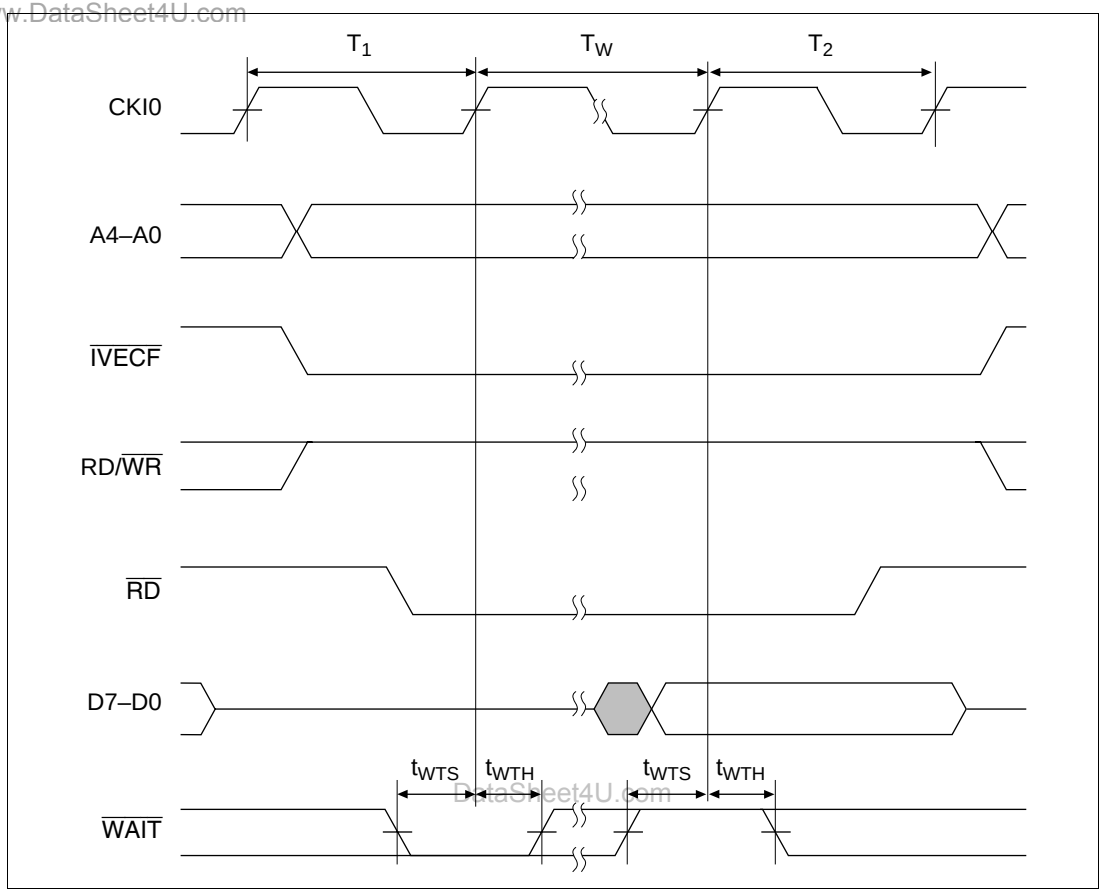


Figure 15.70 Interrupt Vector Fetch Cycle (1 External Wait Cycle)

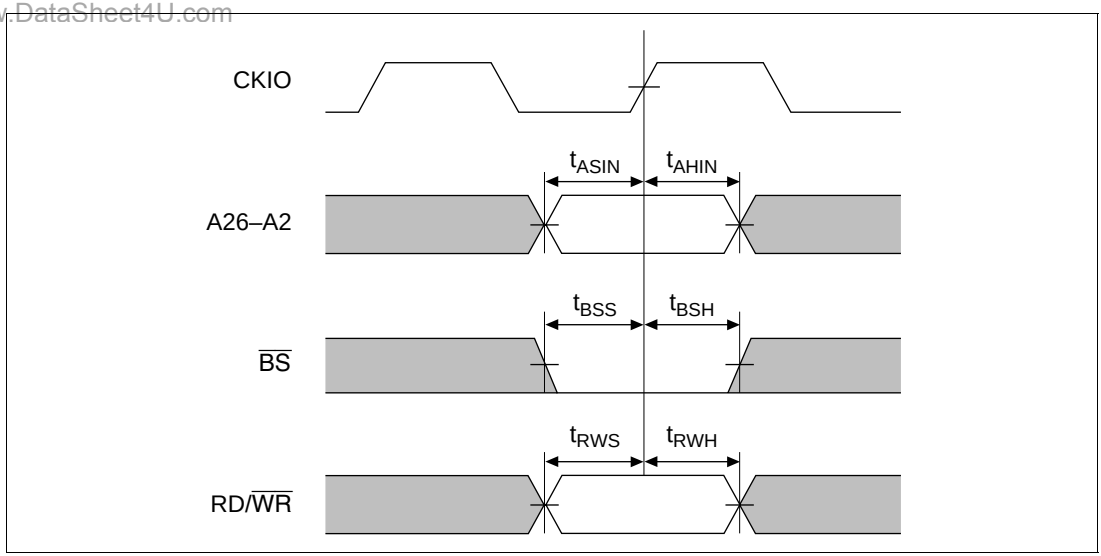


Figure 15.71 Address Monitor Cycle

15.3.4 DMAC Timing

Table 15.10 DMAC Timing (Conditions: V_{CC} = 5.0 V ±10%, Ta = –20 to +75°C)

Item	Symbol	Min	Max	Unit	Figure
DREQ0, DREQ1 setup time (PLL Off, On)	t_{DRQS}	30	—	ns	15.72
DREQ0, DREQ1 setup time (PLL On, 1/4 cycle delay)	t_{DRQS}	30 – 1/4 tcyc	—	ns	
DREQ0, DREQ1 hold time (PLL Off, On)	t_{DRQH}	15	—	ns	
DREQ0, DREQ1 hold time (PLL On, 1/4 cycle delay)	t_{DRQH}	1/4 tcyc + 15	—	ns	
DREQ0, DREQ1 low level width	t_{DRQW}	1.5	—	t_{cyc}	

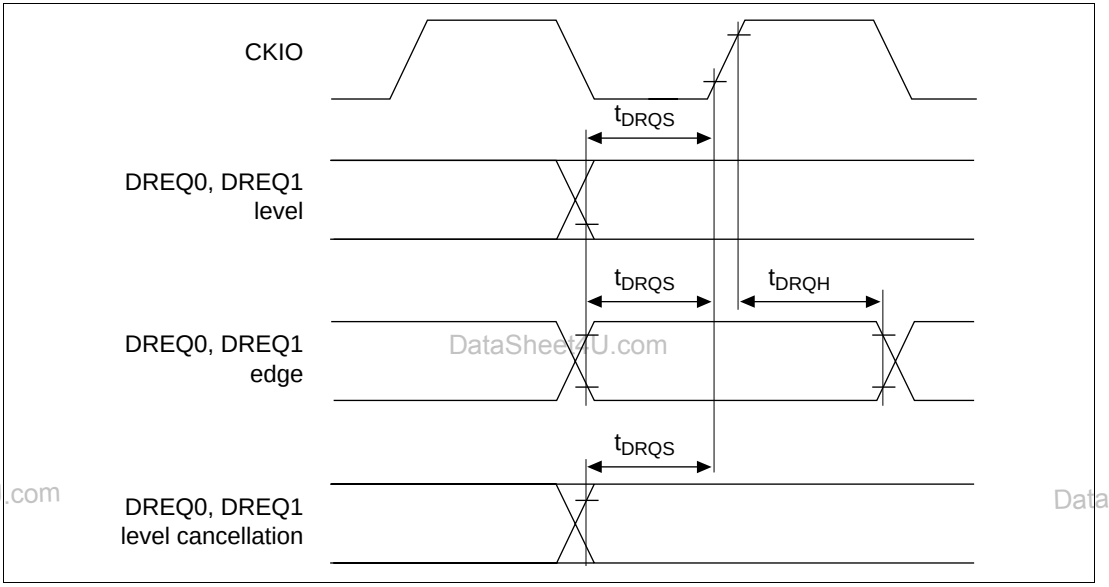


Figure 15.72 DREQ0, DREQ1 Input Timing

15.3.5 Free-Running Timer Timing

Table 15.11 Free-Running Timer Timing (Conditions: V_{CC} = 5.0 V ±10%, Ta = –20 to +75°C)

Item	Symbol	Min	Max	Unit	Figure
Output compare output delay time (PLL Off, On)	t_{TOCD}	—	160	ns	15.73
Output compare output delay time (PLL On, 1/4 cycle delay)	t_{TOCD}	—	1/4 tcyc + 160	ns	
Input capture input setup time (PLL Off, On)	t_{TICS}	80	—	ns	
Input capture input setup time (PLL On, 1/4 cycle delay)	t_{TICS}	80 – 1/4 tcyc	—	ns	
Timer clock input setup time (PLL Off, On)	t_{TCKS}	80	—	ns	15.74
Timer clock input setup time (PLL On, 1/4 cycle delay)	t_{TCKS}	80 –1/4 tcyc	—	ns	
Timer clock pulse width (single edge)	t_{TCKWH}	4.5	—	t_{cyc}	
Timer clock pulse width (both edges)	t_{TCKWL}	8.5	—	t_{cyc}	

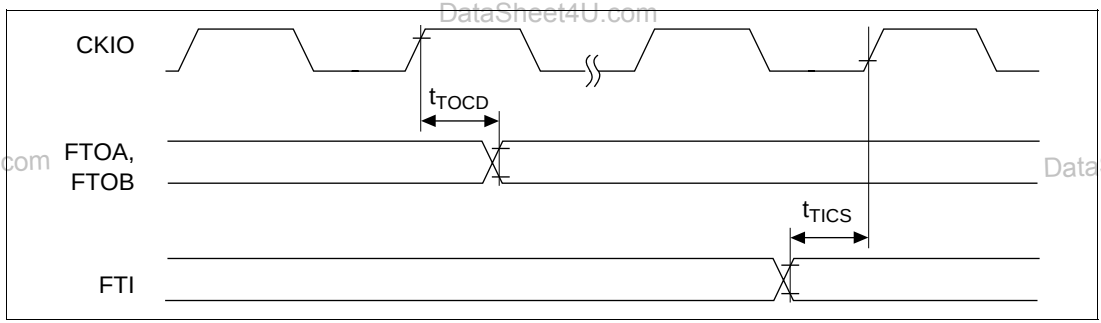


Figure 15.73 FRT Input/Output Timing

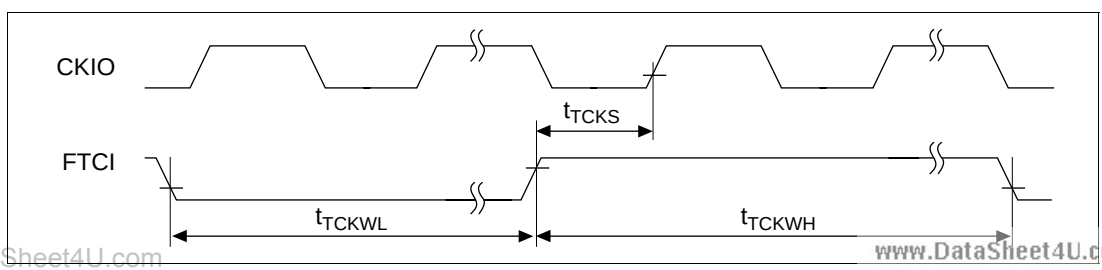


Figure 15.74 FRT Clock Input Timing

15.3.6 Watchdog Timer Timing

Table 15.12 Watchdog Timer Timing (Conditions: V_{CC} = 5.0 V ±10%, Ta = –20 to +75°C)

Item	Symbol	Min	Max	Unit	Figure
$\overline{\text{WDTOVF}}$ delay time (PLL Off, On)	t_{WOVD}	—	70	ns	15.75
$\overline{\text{WDTOVF}}$ delay time (PLL On, 1/4 cycle delay)	t_{WOVD}	—	1/4 tcyc + 70	ns	

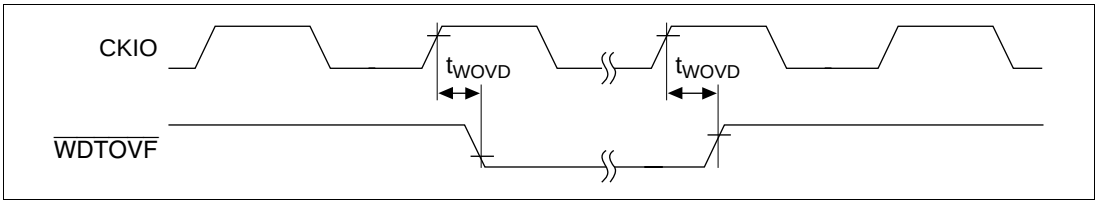


Figure 15.75 Watchdog Timer Output Timing

15.3.7 Serial Communication Interface Timing

Table 15.13 Serial Communication Interface Timing
(Conditions: V_{CC} = 5.0 V ±10%, Ta = –20 to +75°C)

Item	Symbol	Min	Max	Unit	Figure
Input clock cycle	t_{scyc}	16	—	t_{cyc}	15.76
Input clock cycle (clocked synchronous mode)	t_{scyc}	24	—	t_{cyc}	
Input clock pulse width	t_{sckw}	0.4	0.6	t_{scyc}	
Transmit data delay time (clocked synchronous mode)	t_{TXD}	—	70	ns	15.77
Receive data setup time (clocked synchronous mode)	t_{RXS}	70	—	ns	
Receive data hold time (clocked synchronous mode)	t_{RXH}	70	—	ns	

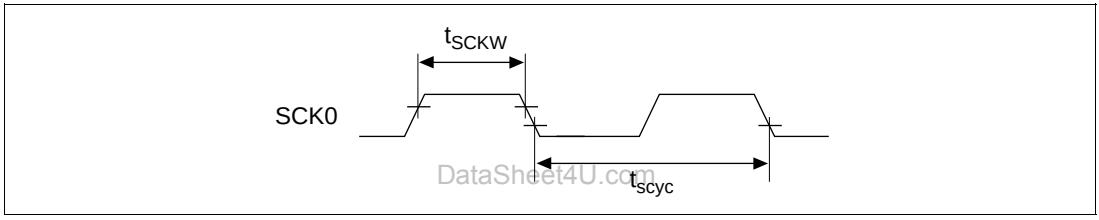


Figure 15.76 Input Clock Input/Output Timing

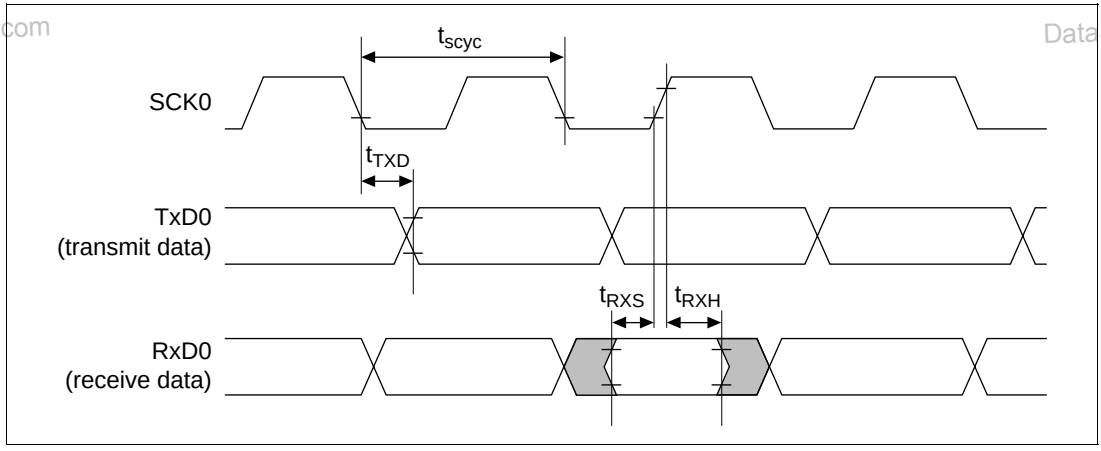
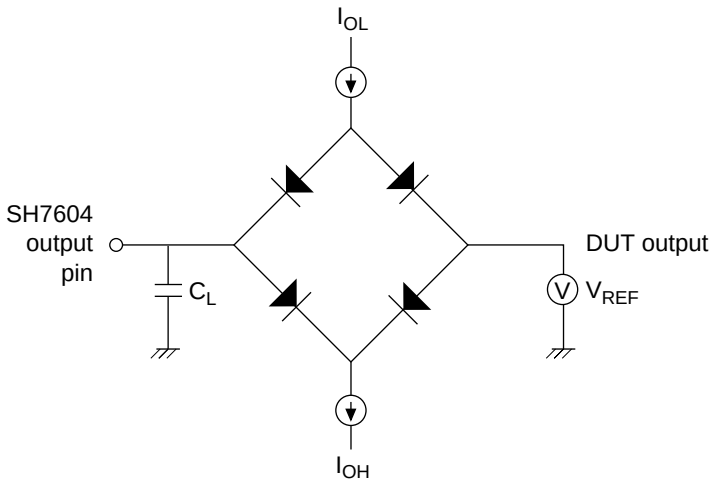


Figure 15.77 SCI Input/Output Timing (Clocked Synchronous Mode)

15.3.8 AC Characteristics Measurement Conditions

- I/O signal reference level: 1.5 V
- Input pulse level: V_{SS} to 3.0 V (where RES, NMI, CKIO and MD5-MD0 are within the range V_{SS} to V_{CC})
- Input rise and fall times: 1 ns



Notes: 1. C_L is a total value that includes the capacitance of measurement instruments, etc., and is set as follows for each pin.
30 pF: CKIO, $\overline{RAS}$, $\overline{CAS}$, CKE, $\overline{CS0}$–$\overline{CS3}$, $\overline{BREQ}$, $\overline{BACK}$, DACK0, DACK1, $\overline{IVECF}$, $\overline{CKPACK}$.
50 pF: All output pins other than the above.

2. I_{OL} and I_{OH} values are as shown in section 15.2, DC Characteristics, and table 15.3, Permitted Output Current Values.

Figure 15.78 Output Load Circuit

Section 16 Electrical Characteristics (3V Version)

16.1 Absolute Maximum Ratings

Table 16.1 shows the absolute maximum ratings.

Table 16.1 Absolute Maximum Ratings

Item	Symbol	Rating	Unit
Power supply voltage	V_{CC}	–0.3 to +7.0	V
Input voltage	Vin	–0.3 to V_{CC} + 0.3	V
Operating temperature	Topr	–20 to +75	°C
Storage temperature	Tstg	–55 to +125	°C

Caution: Operating the chip in excess of the absolute maximum rating may result in permanent damage.

16.2 DC Characteristics

Tables 16.2 and 16.3 list DC characteristics.

Table 16.2 DC Characteristics (Conditions: V_{CC} = 3.0 to 5.5 V, Ta = –20 to +75° C)

Item		Symbol	Min	Typ	Max	Unit	Test Conditions
Input high-level voltage	$\overline{RES}$, NMI, MD5–MD0	V_{IH}	$V_{CC} \times 0.9$	—	V_{CC} + 0.3	V	During standby
			$V_{CC} \times 0.9$	—	V_{CC} + 0.3	V	Normal operation
	EXTAL, CKIO		$V_{CC} \times 0.9$	—	V_{CC} + 0.3	V	
	Other input pins		$V_{CC} \times 0.7$	—	V_{CC} + 0.3	V	
Input low-level voltage	$\overline{RES}$, NMI, MD5–MD0	V_{IL}	–0.3	—	V_{CC}×0.1	V	During standby
			–0.3	—	V_{CC}×0.1	V	Normal operation
	Other input pins		–0.3	—	V_{CC}×0.1	V	
Input leak current	$\overline{RES}$	|Iin|	—	—	1.0	µA	Vin = 0.5 to V_{CC} – 0.5 V
	NMI, MD5–MD0		—	—	1.0	µA	Vin = 0.5 to V_{CC} – 0.5 V
	Other input pins		—	—	1.0	µA	Vin = 0.5 to V_{CC} – 0.5 V
3-state leak current (while off)	A26–A0, D31–D0, $\overline{BS}$, $\overline{CS3}$–$\overline{CS0}$, RD/$\overline{WR}$, $\overline{RAS}$, $\overline{CAS}$, $\overline{WE3}$–$\overline{WE0}$, $\overline{RD}$, $\overline{IVECF}$	$\|I_{STI}\|$	—	—	1.0	µA	Vin = 0.5 to V_{CC} – 0.5 V
Output high-level voltage	All output pins	V_{OH}	V_{CC} – 0.5	—	—	V	I_{OH} = –200 µA
			V_{CC} – 1.0	—	—	V	I_{OH} = –1 mA
Output low level voltage	All output pins	V_{OL}	—	—	0.4	V	I_{OL} = 1.6 mA
Input capacitance	$\overline{RES}$	Cin	—	—	15	pF	Vin = 0 V f = 1 MHz Ta = 25°C
	NMI		—	—	15	pF	
	All other input pins (including D31–D0)		—	—	15	pF	

Table 16.2 DC Characteristics (Conditions: V_{CC} = 3.0 to 5.5 V, Ta = –20 to +75° C) (cont)

Item		Symbol	Min	Typ	Max	Unit	Test Conditions
Current consump-tion	Normal operation	I_{CC}	—	25	30	mA	f = 8 MHz
			—	45	55	mA	f = 16 MHz
			—	60	70	mA	f = 28.7 MHz
	Sleep		—	15	20	mA	f = 8 MHz
			—	30	40	mA	f = 16 MHz
			—	40	50	mA	f = 28.7 MHz
	Standby		—	1	5	µA	Ta ≤ 50°C
			—	—	20	µA	50°C < Ta

Notes: 1. When no PLL is used, do not leave the $PLLV_{CC}$ and $PLLV_{SS}$ pins open. Connect $PLLV_{CC}$ to V_{CC} and $PLLV_{SS}$ to V_{SS}.
2. Current consumption values shown are the values at which all output pins are without load under conditions of V_{IH} min = V_{CC} – 0.5 V, V_{IL} max = 0.5 V.

Table 16.3 Permitted Output Current Values (Conditions: V_{CC} = 5.0 V ± 10%, Ta = –20 to +75°C)

Item	Symbol	Min	Typ	Max	Unit
Output low-level permissible current (per pin)	I_{OL}	—	—	2.0	mA
Output low-level permissible current (total)	ΣI_{OL}	—	—	80	mA
Output high-level permissible current (per pin)	$-I_{OH}$	—	—	2.0	mA
Output high-level permissible current (total)	$\Sigma(-I_{OH})$	—	—	25	mA

Caution: To ensure chip reliability, do not exceed the output current values given in table 16.3.

16.3 AC Characteristics

16.3.1 Clock Timing

Table 16.4 Clock Timing (Conditions: V_{CC} = 3.0 to 0.5 V, Ta = –20 to +75°C)

Item	Symbol	Min	Max	Unit	Figures
Operating frequency	f_{OP}	4	20	MHz	16.1
Clock cycle time	t_{cyc}	50	143[*1] or 250[*2]	ns	
Clock high pulse width	t_{CH}	8[*1] or 15[*2]	—	ns	
Clock low pulse width	t_{CL}	8[*1] or 15[*2]	—	ns	
Clock rise time	t_{CR}	—	5	ns	
Clock fall time	t_{CF}	—	5	ns	
EXTAL clock input frequency	f_{EX}	4	8	MHz	16.2
EXTAL clock input cycle time	t_{EXcyc}	125	250	ns	
EXTAL clock input low level pulse width	t_{EXL}	50	—	ns	
EXTAL clock input high level pulse width	t_{EXH}	50	—	ns	
EXTAL clock input rise time	t_{EXR}	—	5	ns	
EXTAL clock input clock fall time	t_{EXF}	—	5	ns	
Power-on oscillation settling time	t_{OSC1}	10	—	ms	16.3
Software standby oscillation settling time 1	t_{OSC2}	10	—	ms	16.4
Software standby oscillation settling time 2	t_{OSC3}	10	—	ms	16.5
PLL synchronization settling time	t_{PLL}	1	—	µs	16.6

Notes: 1. With PLL circuit 1 operating.
2. With PLL circuit 1 not used.

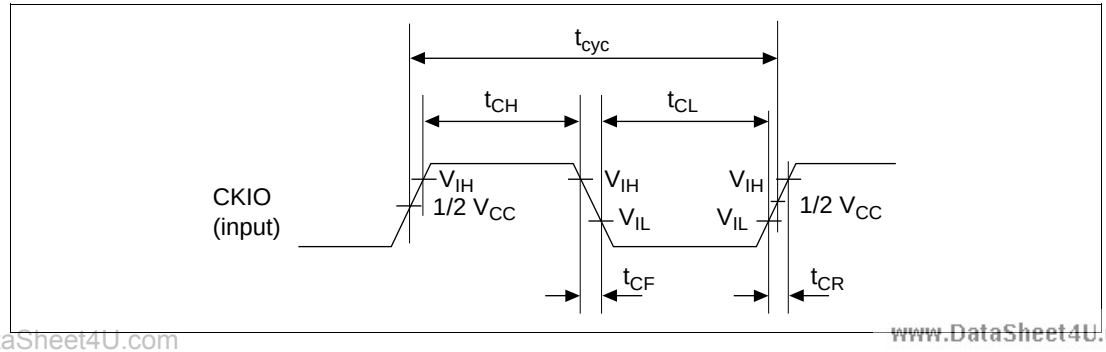
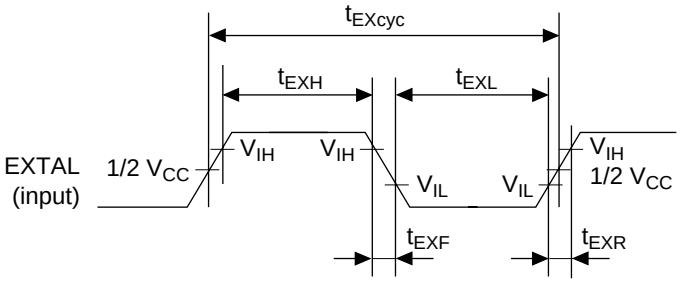
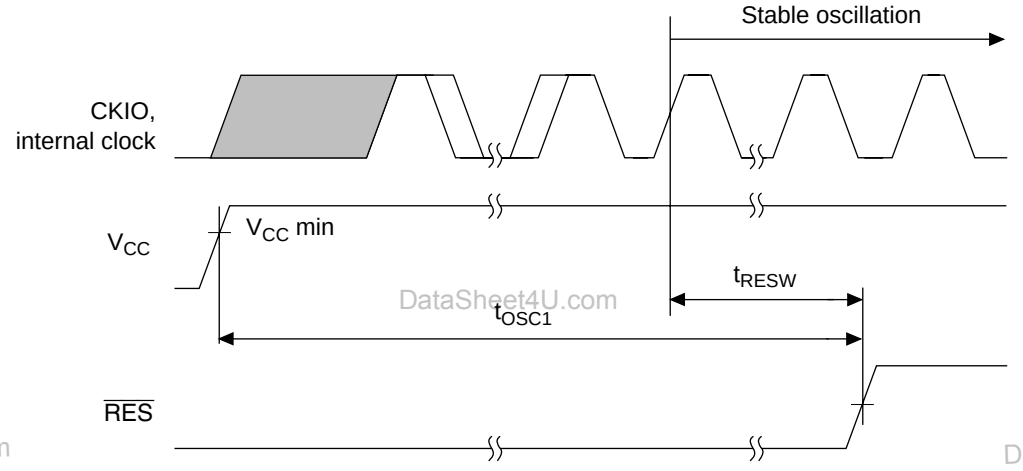


Figure 16.1 CKIO Input Timing



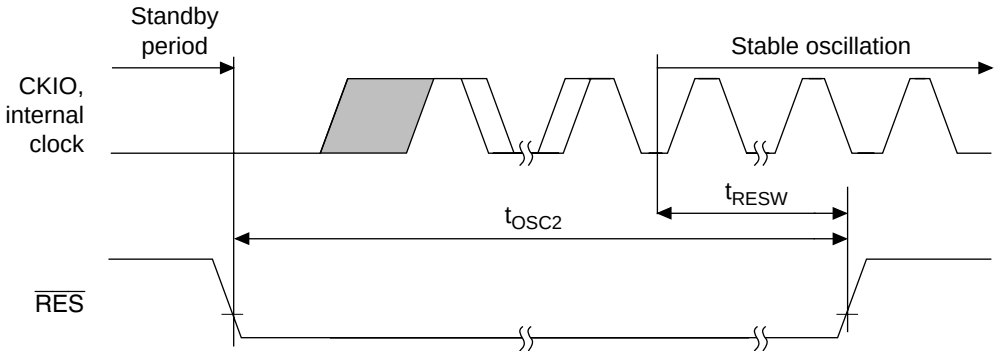
Note: External clock input from EXTAL pin.

Figure 16.2 EXTAL Clock Input Timing



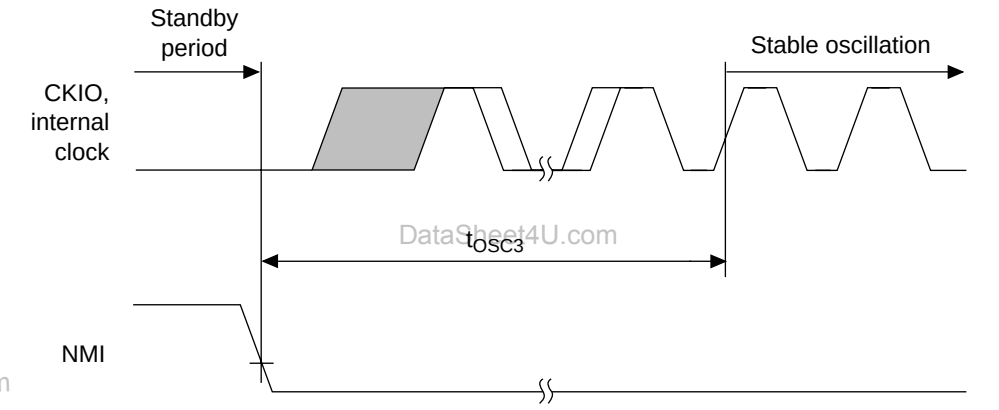
Note: Oscillation settling time when on-chip crystal oscillator is used.

Figure 16.3 Oscillation Settling Time at Power-On



Note: Oscillation settling time when on-chip crystal oscillator is used.

Figure 16.4 Oscillation Settling Time at Standby Return (via $\overline{RES}$)



Note: Oscillation settling time when on-chip crystal oscillator is used.

Figure 16.5 Oscillation Settling Time at Standby Return (via NMI)

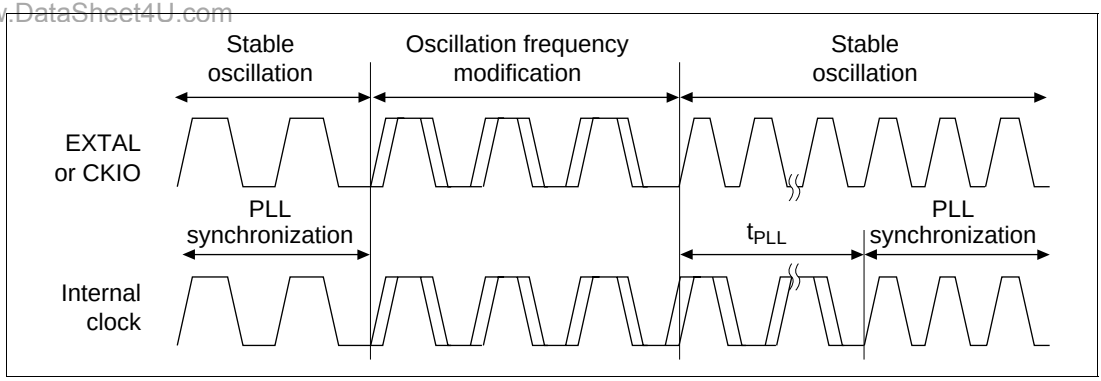


Figure 16.6 PLL Synchronization Settling Time

16.3.2 Control Signal Timing

Table 16.5 Control Signal Timing (Conditions: V_{CC} = 3.0 to 5.5 V, Ta = –20 to +75°C)

Item	Symbol	Min	Max	Unit	Figure
$\overline{RES}$ rise, fall	t_{RESr}, t_{RESf}	—	200	ns	16.7
$\overline{RES}$ pulse width	t_{RESW}	20	—	t_{cyc}	
NMI reset setup time	t_{NMIRS}	tcyc + 10	—	ns	
NMI reset hold time	t_{NMIRH}	tcyc + 10	—	ns	
NMI rise, fall	t_{NMIr}, t_{NMIf}	—	200	ns	
NMI minimum pulse width	t_{IRQES}	3	—	tcyc	
$\overline{RES}$ setup time*	t_{RESS}	40	—	ns	16.8, 16.9
NMI setup time*	t_{NMIS}	40	—	ns	
$\overline{IRL3}$–$\overline{IRL0}$ setup time*	t_{IRLS}	40	—	ns	
$\overline{RES}$ hold time	t_{RESH}	20	—	ns	16.8, 16.9
NMI hold time	t_{NMIH}	20	—	ns	
$\overline{IRL3}$–$\overline{IRL0}$ hold time	t_{IRLH}	20	—	ns	
$\overline{BRLS}$ setup time 1 (PLL on)	t_{BLSS1}	1/2 tcyc + 20	—	ns	16.10
$\overline{BRLS}$ hold time 1 (PLL on)	t_{BLSH1}	15 – 1/2 tcyc	—	ns	
$\overline{BGR}$ delay time 1 (PLL on)	t_{BGRD1}	—	1/2 tcyc + 25	ns	
$\overline{BRLS}$ setup time 1 (PLL on, 1/4 cycle delay)	t_{BLSS1}	1/4 tcyc + 20	—	ns	16.10
$\overline{BRLS}$ hold time 1 (PLL on, 1/4 cycle delay)	t_{BLSH1}	15 – 1/4 tcyc	—	ns	
$\overline{BGR}$ delay time 1 (PLL on, 1/4 cycle delay)	t_{BGRD1}	—	3/4 tcyc + 25	ns	
$\overline{BRLS}$ setup time 2 (PLL off)	t_{BLSS2}	20	—	ns	16.11
$\overline{BRLS}$ hold time 2 (PLL off)	t_{BLSH2}	30	—	ns	
$\overline{BGR}$ delay time 2 (PLL off)	t_{BGRD2}	—	40	ns	

Note: The $\overline{RES}$, NMI and $\overline{IRL3}$-$\overline{IRL0}$ signals are asynchronous inputs, but when the setup times shown here are observed, the signals are considered to have changed at clock fall. If the setup times are not observed, recognition may be delayed until the next clock fall.

Table 16.5 Control Signal Timing (cont)

(Conditions: V_{CC} = 3.0 to 5.5 V, Ta = –20 to +75°C)

Item	Symbol	Min	Max	Unit	Figure
$\overline{\text{BREQ}}$ delay time 1 (PLL on)	t_{BRQD1}	—	1/2 tcyc + 25	ns	16.12
$\overline{\text{BACK}}$ setup time 1 (PLL on)	t_{BAKS1}	1/2 tcyc + 20	—	ns	
$\overline{\text{BACK}}$ hold time 1 (PLL on)	t_{BAKH1}	15 – 1/2 tcyc	—	ns	
$\overline{\text{BREQ}}$ delay time 1 (PLL on, 1/4 cycle delay)	t_{BRQD1}	—	3/4 tcyc + 25	ns	16.12
$\overline{\text{BACK}}$ setup time 1 (PLL on, 1/4 cycle delay)	t_{BAKS1}	1/4 tcyc + 20	—	ns	
$\overline{\text{BACK}}$ hold time 1 (PLL on, 1/4 cycle delay)	t_{BAKH1}	15 – 1/4 tcyc	—	ns	
$\overline{\text{BREQ}}$ delay time 2 (PLL off)	t_{BRQD2}	—	40	ns	16.13
$\overline{\text{BACK}}$ setup time 2 (PLL off)	t_{BAKS2}	20	—	ns	
$\overline{\text{BACK}}$ hold time 2 (PLL off)	t_{BAKH2}	30	—	ns	
Bus tri-state delay time 1 (PLL on)	t_{BOFF1}	0	35	ns	16.10, 16.12
Bus buffer on time 1 (PLL on)	t_{BON1}	0	33	ns	
Bus tri-state delay time 1 (PLL on, 1/4 cycle delay)	t_{BOFF1}	1/4 tcyc	1/4 tcyc + 35	ns	16.10, 16.12
Bus buffer on time 1 (PLL on, 1/4 cycle delay)	t_{BON1}	1/4 tcyc	1/4 tcyc + 33	ns	
Bus tri-state delay time 1 (PLL off)	t_{BOFF1}	0	45	ns	16.11, 16.13
Bus buffer on time 1 (PLL off)	t_{BON1}	0	40	ns	
Bus tri-state delay time 2 (PLL on)	t_{BOFF2}	1/2 tcyc	1/2 tcyc + 35	ns	16.10, 16.12
Bus buffer on time 2 (PLL on)	t_{BON2}	1/2 tcyc	1/2 tcyc + 33	ns	
Bus tri-state delay time 2 (PLL on, 1/4 cycle delay)	t_{BOFF2}	3/4 tcyc	3/4 tcyc + 35	ns	16.10, 16.12
Bus buffer on time 2 (PLL on, 1/4 cycle delay)	t_{BON2}	3/4 tcyc	3/4 tcyc + 33	ns	
Bus tri-state delay time 3 (PLL off)	t_{BOFF3}	0	45	ns	16.11, 16.13
Bus buffer on time 3 (PLL off)	t_{BON3}	0	40	ns	

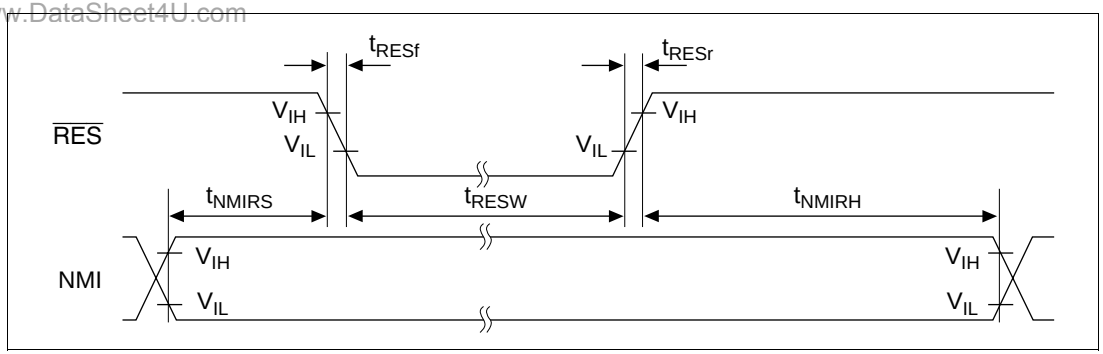


Figure 16.7 Reset Input Timing

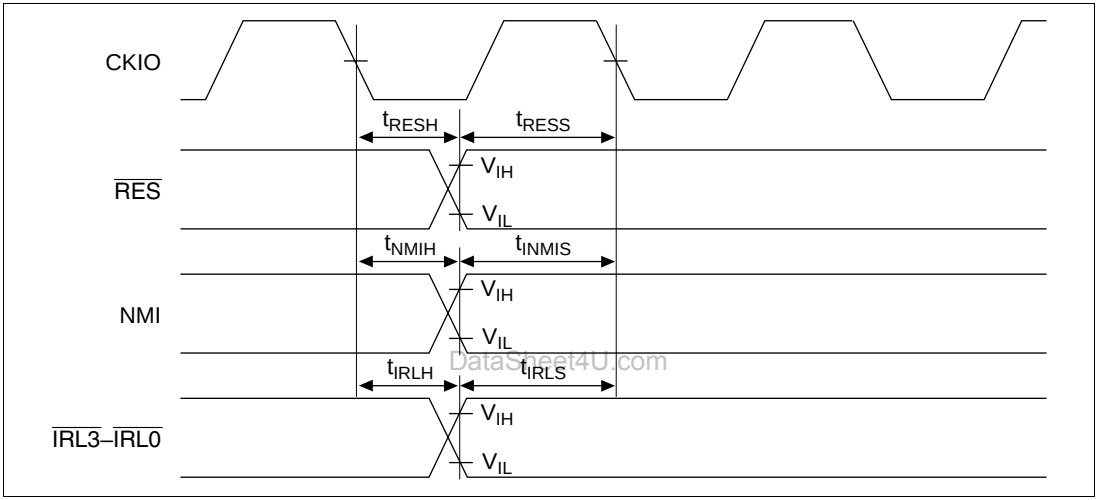


Figure 16.8 Interrupt Signal Input Timing (PLL1 Off)

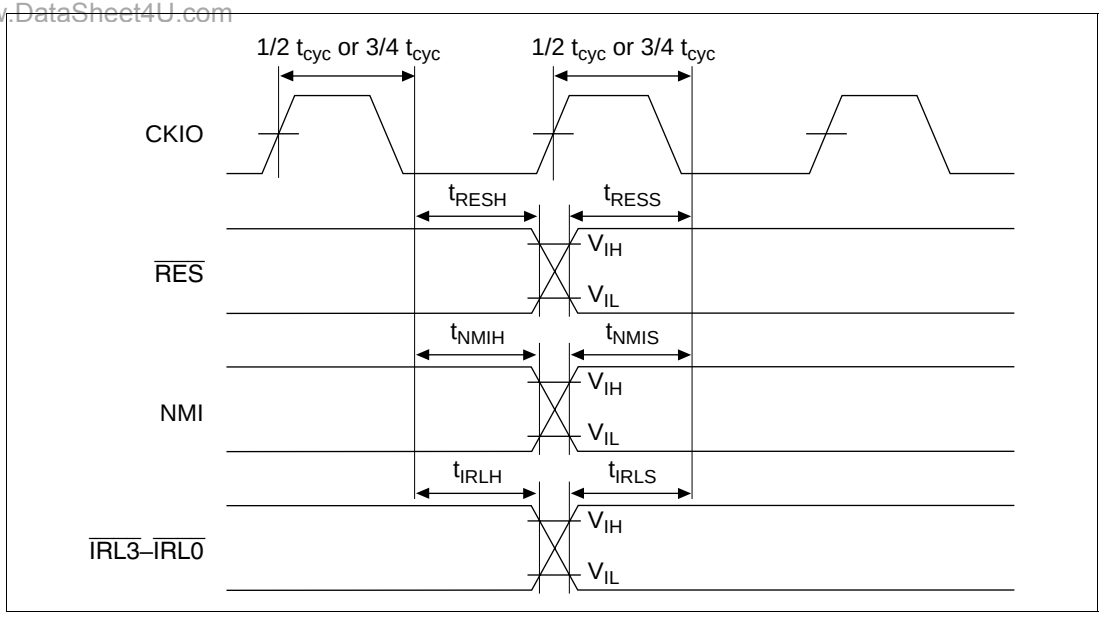


Figure 16.9 Interrupt Signal Input Timing (PLL1 On)

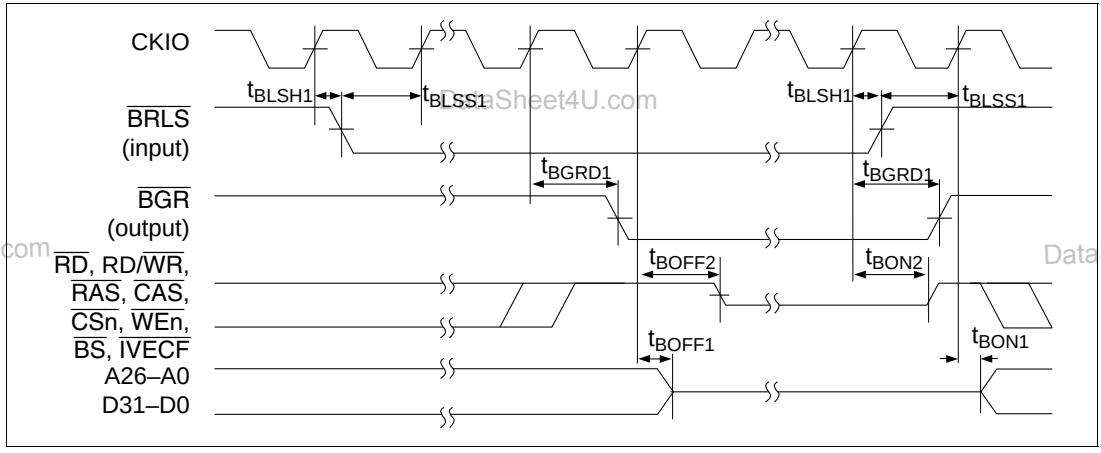


Figure 16.10 Bus Release Timing (Master Mode, PLL1 On)

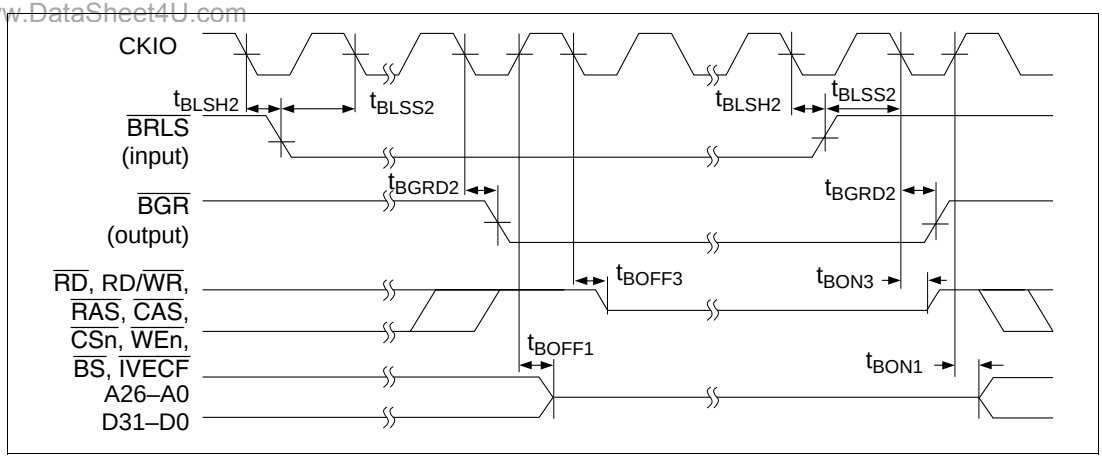


Figure 16.11 Bus Release Timing (Master Mode, PLL1 Off)

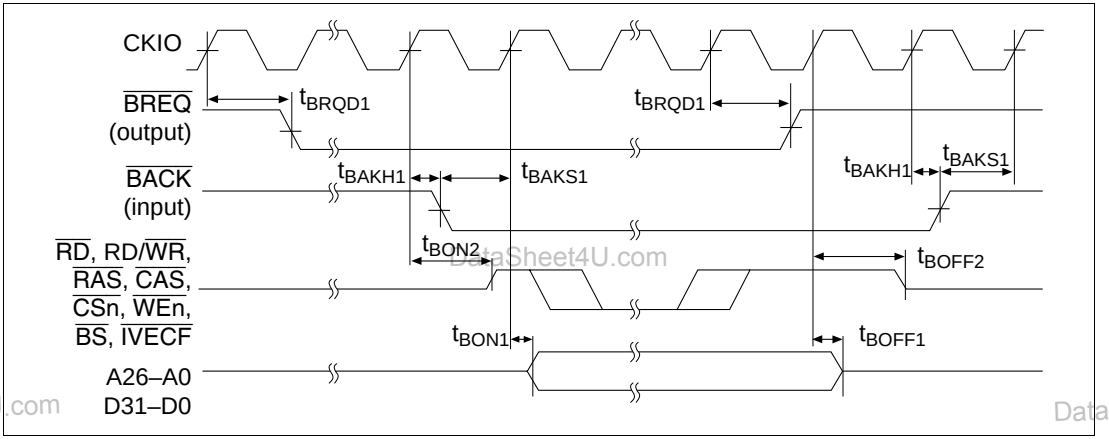


Figure 16.12 Bus Release Timing (Slave Mode, PLL1 On)

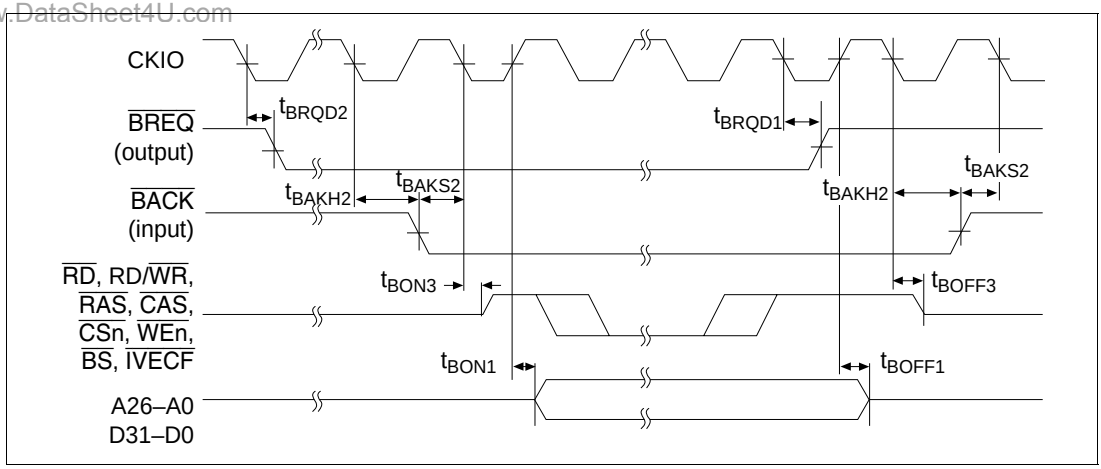


Figure 16.13 Bus Release Timing (Slave Mode, PLL1 Off)

16.3.3 Bus Timing

Table 16.6 Bus Timing With PLL On [Mode 0, 4]
(Conditions: V_{CC} = 3.0 to 5.5 V, Ta = –20 to +75°C)

Item	Symbol	Min	Max	Unit	Figures
Address delay time	t_{AD}	—	28	ns	16.14, 16.20, 16.40, 16.52, 16.66, 16.68
$\overline{BS}$ delay time	t_{BSD}	—	25	ns	16.14, 16.20, 16.40, 16.52, 16.66
$\overline{CS}$ delay time 1	t_{CSD1}	—	25	ns	16.14, 16.20, 16.40, 16.52, 16.66
$\overline{CS}$ delay time 2	t_{CSD2}	—	1/2 tcyc + 25	ns	16.14, 16.66
Read/write delay time	t_{RWD}	—	25	ns	16.14, 16.20, 16.40, 16.52, 16.66
Read strobe delay time 1	t_{RSD1}	—	1/2 tcyc + 25	ns	16.14, 16.40, 16.52, 16.66, 16.68
Read data setup time 1	t_{RDS1}	1/2 tcyc + 10	—	ns	16.14, 16.40, 16.52, 16.66, 16.68
Read data setup time 3 (SDRAM)	t_{RDS3}	1/2 tcyc + 10	—	ns	16.20
Read data hold time 2	t_{RDH2}	0	—	ns	16.14, 16.66
Read data hold time 4 (SDRAM)	t_{RDH4}	0	—	ns	16.20
Read data hold time 5 (DRAM)	t_{RDH5}	0	—	ns	16.40
Read data hold time 6 (PSRAM)	t_{RDH6}	0	—	ns	16.52
Read data hold time 7 (interrupt vector)	t_{RDH7}	0	—	ns	16.68
Write enable delay time	t_{WED1}	1/2 tcyc + 3	1/2 tcyc + 25	ns	16.14, 16.15, 16.52, 16.53
Write data delay time 1	t_{WDD}	—	25	ns	16.15, 16.27, 16.41, 16.53
Write data hold time 1	t_{WDH1}	3	—	ns	16.15, 16.27, 16.41, 16.53
Data buffer on time	t_{DON}	—	25	ns	16.15, 16.27, 16.41, 16.53
Data buffer off time	t_{DOF}	—	25	ns	16.15, 16.27, 16.41, 16.53

Table 16.6 Bus Timing With PLL On [Mode 0, 4] (cont)
(Conditions: V_{CC} = 3.0 to 5.5 V, Ta = –20 to +75°C)

Item	Symbol	Min	Max	Unit	Figures
DACK delay time 1	t_{DACD1}	—	25	ns	16.14, 16.20, 16.40, 16.52, 16.66
DACK delay time 2	t_{DACD2}	—	1/2 tcyc + 25	ns	16.14, 16.20, 16.40, 16.52, 16.66
$\overline{WAIT}$ setup time	t_{WTS}	20	—	ns	16.19, 16.43, 16.55, 16.66, 16.70
$\overline{WAIT}$ hold time	t_{WTH}	10	—	ns	16.19, 16.43, 16.55, 16.66, 16.70
$\overline{RAS}$ delay time 1 (SDRAM)	t_{RASD1}	—	25	ns	16.20
$\overline{RAS}$ delay time 2 (DRAM)	t_{RASD2}	1/2 tcyc + 3	1/2 tcyc + 25	ns	16.40
$\overline{CAS}$ delay time 1 (SDRAM)	t_{CASD1}	—	25	ns	16.20
$\overline{CAS}$ delay time 2 (DRAM)	t_{CASD2}	1/2 tcyc + 3	1/2 tcyc + 25	ns	16.40
DQM delay time	t_{DQMD}	—	25	ns	16.20
CKE delay time	t_{CKED}	—	33	ns	16.37
$\overline{CE}$ delay time 1	t_{CED1}	1/2 tcyc + 3	1/2 tcyc + 25	ns	16.52
$\overline{OE}$ delay time 1	t_{OED1}	—	1/2 tcyc + 25	ns	16.52
$\overline{IVECF}$ delay time	t_{IVD}	—	25	ns	16.68
Address input setup time	t_{ASIN}	25	—	ns	16.71
Address input hold time	t_{AHIN}	10	—	ns	16.71
$\overline{BS}$ input setup time	t_{BSS}	25	—	ns	16.71
$\overline{BS}$ input hold time	t_{BSH}	10	—	ns	16.71
Read/write input setup time	t_{RWS}	25	—	ns	16.71
Read/write input hold time	t_{RWH}	10	—	ns	16.71

Table 16.7 Bus Timing With PLL On and 1/4 Cycle Delay [Mode 1, 5]
(Conditions: V_{CC} = 3.0 to 5.5 V, Ta = –20 to +75°C)

Item	Symbol	Min	Max	Unit	Figures
Address delay time	t_{AD}	—	1/4 tcyc + 28	ns	16.14, 16.20, 16.40, 16.52, 16.66, 16.68
$\overline{BS}$ delay time	t_{BSD}	—	1/4 tcyc + 25	ns	16.14, 16.20, 16.40, 16.52, 16.66
$\overline{CS}$ delay time 1	t_{CSD1}	—	1/4 tcyc + 25	ns	16.14, 16.20, 16.40, 16.52, 16.66
$\overline{CS}$ delay time 2	t_{CSD2}	—	3/4 tcyc + 25	ns	16.14, 16.66
Read/write delay time	t_{RWD}	—	1/4 tcyc + 25	ns	16.14, 16.20, 16.40, 16.52, 16.66
Read strobe delay time 1	t_{RSD1}	—	3/4 tcyc + 25	ns	16.14, 16.40, 16.52, 16.66, 16.68
Read data setup time 1	t_{RDS1}	1/4 tcyc + 10	—	ns	16.14, 16.40, 16.52, 16.66, 16.68
Read data setup time 3 (SDRAM)	t_{RDS3}	1/4 tcyc + 10	—	ns	16.20
Read data hold time 2	t_{RDH2}	0	—	ns	16.14, 16.66
Read data hold time 4 (SDRAM)	t_{RDH4}	0	—	ns	16.20
Read data hold time 5 (DRAM)	t_{RDH5}	0	—	ns	16.40
Read data hold time 6 (PSRAM)	t_{RDH6}	0	—	ns	16.52
Read data hold time 7 (interrupt vector)	t_{RDH7}	0	—	ns	16.68
Write enable delay time	t_{WED1}	3/4 tcyc + 3	3/4 tcyc + 25	ns	16.14, 16.15, 16.52, 16.53
Write data delay time 1	t_{WDD}	—	1/4 tcyc + 25	ns	16.15, 16.27, 16.41, 16.53
Write data hold time 1	t_{WDH1}	1/4 tcyc + 3	—	ns	16.15, 16.27, 16.41, 16.53
Data buffer on time	t_{DON}	—	1/4 tcyc + 25	ns	16.15, 16.27, 16.41, 16.53
Data buffer off time	t_{DOF}	—	1/4 tcyc + 25	ns	16.15, 16.27, 16.41, 16.53

Table 16.7 Bus Timing With PLL On and 1/4 Cycle Delay [Mode 1, 5] (cont)
(Conditions: V_{CC} = 3.0 to 5.5 V, Ta = –20 to +75°C)

Item	Symbol	Min	Max	Unit	Figures
DACK delay time 1	t_{DACD1}	—	1/4 tcyc + 25	ns	16.14, 16.20, 16.40, 16.52, 16.66
DACK delay time 2	t_{DACD2}	—	3/4 tcyc + 25	ns	16.14, 16.20, 16.40, 16.52, 16.66
$\overline{WAIT}$ setup time	t_{WTS}	20 –1/4 tcyc	—	ns	16.19, 16.43, 16.55, 16.66, 16.70
$\overline{WAIT}$ hold time	t_{WTH}	1/4 tcyc+10	—	ns	16.19, 16.43, 16.55, 16.66, 16.70
$\overline{RAS}$ delay time 1 (SDRAM)	t_{RASD1}	—	1/4 tcyc + 25	ns	16.20
$\overline{RAS}$ delay time 2 (DRAM)	t_{RASD2}	3/4 tcyc + 3	3/4 tcyc + 25	ns	16.40
$\overline{CAS}$ delay time 1 (SDRAM)	t_{CASD1}	—	1/4 tcyc + 25	ns	16.20
$\overline{CAS}$ delay time 2 (DRAM)	t_{CASD2}	3/4 tcyc + 3	3/4 tcyc + 25	ns	16.40
DQM delay time	t_{DQMD}	—	1/4 tcyc + 25	ns	16.20
CKE delay time	t_{CKED}	—	1/4 tcyc + 33	ns	16.37
$\overline{CE}$ delay time 1	t_{CED1}	3/4 tcyc + 3	3/4 tcyc + 25	ns	16.52
$\overline{OE}$ delay time 1	t_{OED1}	—	3/4 tcyc + 25	ns	16.52
$\overline{IVECF}$ delay time	t_{IVD}	—	1/4 tcyc + 25	ns	16.68
Address input setup time	t_{ASIN}	25 – 1/4 tcyc	—	ns	16.71
Address input hold time	t_{AHIN}	1/4 tcyc+10	—	ns	16.71
$\overline{BS}$ input setup time	t_{BSS}	25 – 1/4 tcyc	—	ns	16.71
$\overline{BS}$ input hold time	t_{BSH}	1/4 tcyc +10	—	ns	16.71
Read/write input setup time	t_{RWS}	25 – 1/4 tcyc	—	ns	16.71
Read/write input hold time	t_{RWH}	1/4 tcyc +10	—	ns	16.71

Table 16.8 Bus Timing With PLL Off (CKIO Input) [Mode 6]
(Conditions: V_{CC} = 3.0 to 5.5 V, Ta = –20 to +75°C)

Item	Symbol	Min	Max	Unit	Figures
Address delay time	t_{AD}	—	43	ns	16.16, 16.38, 16.47, 16.60, 16.67, 16.69
$\overline{BS}$ delay time	t_{BSD}	—	40	ns	16.16, 16.38, 16.47, 16.60, 16.67
$\overline{CS}$ delay time 1	t_{CSD1}	—	40	ns	16.16, 16.38, 16.47, 16.60, 16.67
$\overline{CS}$ delay time 3	t_{CSD3}	—	40	ns	16.16, 16.67
Read write delay time	t_{RWD}	—	40	ns	16.16, 16.38, 16.47, 16.60, 16.67
Read strobe delay time 2	t_{RSD2}	—	40	ns	16.16, 16.47, 16.60, 16.67, 16.69
Read data setup time 2	t_{RDS2}	10	—	ns	16.16, 16.38, 16.47, 16.60, 16.67, 16.69
Read data hold time 2	t_{RDH2}	0	—	ns	16.16, 16.67
Read data hold time 3	t_{RDH3}	30	—	ns	16.38
Read data hold time 5 (DRAM)	t_{RDH5}	0	—	ns	16.47
Read data hold time 6 (PSRAM)	t_{RDH6}	0	—	ns	16.60
Read data hold time 7 (interrupt vector)	t_{RDH7}	0	—	ns	16.69
Write enable delay time 2	t_{WED2}	—	40	ns	16.17, 16.61
Write data delay time	t_{WDD}	—	40	ns	16.17, 16.39, 16.48, 16.61
Write data hold time 1	t_{WDH1}	3		ns	16.17, 16.39, 16.48, 16.61
Write data hold time 2	t_{WDH2}	5		ns	16.17
Write data hold time 3	t_{WDH3}	3		ns	16.61
DACK delay time 1	t_{DACD1}	—	40	ns	16.16, 16.38, 16.47, 16.60, 16.67
DACK delay time 3	t_{DACD3}	—	40	ns	16.16, 16.38, 16.47, 16.60, 16.67

Table 16.8 Bus Timing With PLL Off (CKIO Input) [Mode 6] (cont)
(Conditions: V_{CC} = 3.0 to 5.5 V, Ta = –20 to +75°C)

Item	Symbol	Min	Max	Unit	Figures
$\overline{\text{WAIT}}$ setup time	t_{WTS}	20	—	ns	16.19, 16.43, 16.55, 16.67, 16.70
$\overline{\text{WAIT}}$ hold time	t_{WTH}	25	—	ns	16.19, 16.43, 16.55, 16.67, 16.70
$\overline{\text{RAS}}$ delay time 1 (SDRAM)	t_{RASD1}	—	40	ns	16.38
$\overline{\text{RAS}}$ delay time 3 (DRAM)	t_{RASD3}	—	40	ns	16.47
$\overline{\text{CAS}}$ delay time 1 (SDRAM)	t_{CASD1}	—	40	ns	16.38
$\overline{\text{CAS}}$ delay time 3 (DRAM)	t_{CASD3}	—	40	ns	16.47
DQM delay time	t_{DQMD}	—	40	ns	16.38
CKE delay time	t_{CKED}	—	48	ns	16.37
$\overline{\text{CE}}$ delay time 2	t_{CED2}	—	40	ns	16.60
$\overline{\text{OE}}$ delay time 2	t_{OED2}	—	40	ns	16.60
$\overline{\text{IVECF}}$ delay time	t_{IVD}	—	40	ns	16.69
$\overline{\text{WE}}$ setup time	t_{WES1}	0	—	ns	16.16
Address setup time 1	t_{AS1}	0	—	ns	16.17
Address setup time 2	t_{AS2}	3	—	ns	16.60
Address hold time 2	t_{AH2}	0	—	ns	16.17
Row address setup time	t_{ASR}	3	—	ns	16.47
Column address setup time	t_{ASC}	3	—	ns	16.47
Write command setup time	t_{WCS}	3	—	ns	16.48
Write data setup time	t_{WDS}	3	—	ns	16.48
Address input setup time*	t_{ASIN}	20	—	ns	16.71
Address input hold time*	t_{AHIN}	25	—	ns	16.71
$\overline{\text{BS}}$ input setup time*	t_{BSS}	20	—	ns	16.71
$\overline{\text{BS}}$ input hold time*	t_{BSH}	25	—	ns	16.71
Read/write input setup time*	t_{RWS}	20	—	ns	16.71
Read/write input hold time*	t_{RWH}	25	—	ns	16.71
Data buffer on time	t_{DON}	—	40	ns	16.17, 16.39, 16.48, 16.61
Data buffer off time	t_{DOF}	—	40	ns	16.17, 16.39, 16.48, 16.61

Note: * When the external addresses monitor function is used, the PLL must be on.

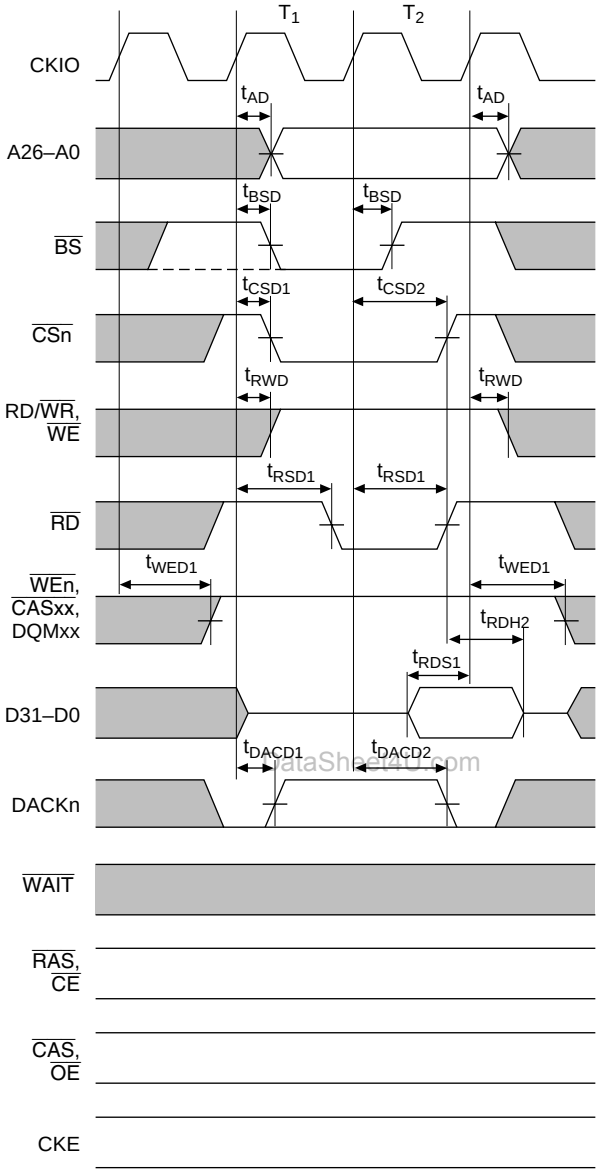
Table 16.9 Bus Timing With PLL Off (CKIO Output) [Mode 2]
(Conditions: V_{CC} = 3.0 to 5.5 V, Ta = –20 to +75°C)

Item	Symbol	Min	Max	Unit	Figures
Address delay time	t_{AD}	—	28	ns	16.16, 16.38, 16.47, 16.60, 16.67, 16.69
$\overline{BS}$ delay time	t_{BSD}	—	25	ns	16.16, 16.38, 16.47, 16.60, 16.67
$\overline{CS}$ delay time 1	t_{CSD1}	—	25	ns	16.16, 16.38, 16.47, 16.60, 16.67
$\overline{CS}$ delay time 3	t_{CSD3}	—	25	ns	16.16, 16.67
Read write delay time	t_{RWD}	—	25	ns	16.16, 16.38, 16.47, 16.60, 16.67
Read strobe delay time 2	t_{RSD2}	—	25	ns	16.16, 16.47, 16.60, 16.67, 16.69
Read data setup time 2	t_{RDS2}	10	—	ns	16.16, 16.38, 16.47, 16.60, 16.67, 16.69
Read data hold time 2	t_{RDH2}	0	—	ns	16.16, 16.67
Read data hold time 3 (SDRAM)	t_{RDH3}	1/2 tcyc	—	ns	16.38
Read data hold time 5 (DRAM)	t_{RDH5}	0	—	ns	16.47
Read data hold time 6 (PSRAM)	t_{RDH6}	0	—	ns	16.60
Read data hold time 7 (interrupt vector)	t_{RDH7}	0	—	ns	16.69
Write enable delay time 2	t_{WED2}	3	25	ns	16.17, 16.61
Write data delay time	t_{WDD}	—	25	ns	16.17, 16.39, 16.48, 16.61
Write data hold time 1	t_{WDH1}	3	—	ns	16.17, 16.39, 16.48, 16.61
Write data hold time 2	t_{WDH2}	5	—	ns	16.17
Write data hold time 3	t_{WDH3}	3	—	ns	16.61
DACK delay time 1	t_{DACD1}	—	25	ns	16.16, 16.38, 16.47, 16.60, 16.67
DACK delay time 3	t_{DACD3}	—	25	ns	16.16, 16.38, 16.47, 16.60, 16.67

Table 16.9 Bus Timing With PLL Off (CKIO Output) [Mode 2] (cont)
(Conditions: V_{CC} = 3.0 to 5.5 V, Ta = –20 to +75°C)

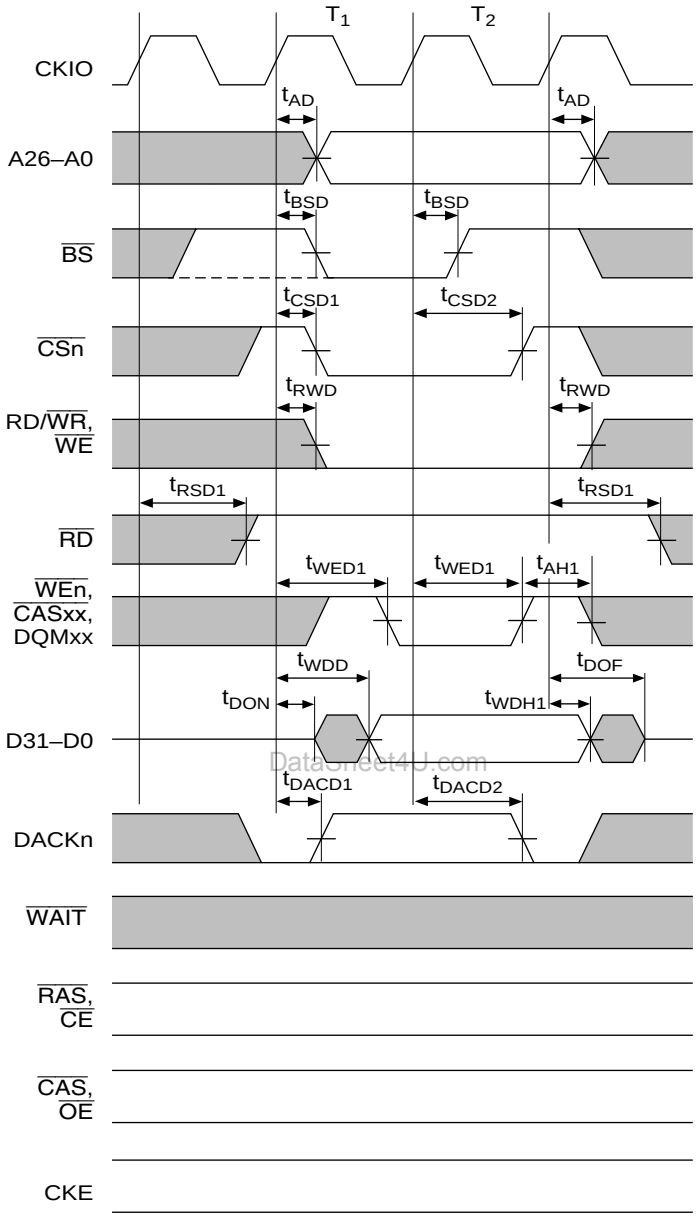
Item	Symbol	Min	Max	Unit	Figures
$\overline{\text{WAIT}}$ setup time	t_{WTS}	20	—	ns	16.19, 16.43, 16.55, 16.67, 16.70
$\overline{\text{WAIT}}$ hold time	t_{WTH}	10	—	ns	16.19, 16.43, 16.55, 16.67, 16.70
$\overline{\text{RAS}}$ delay time 1 (SDRAM)	t_{RASD1}	—	25	ns	16.38
$\overline{\text{RAS}}$ delay time 3 (DRAM)	t_{RASD3}	3	25	ns	16.47
$\overline{\text{CAS}}$ delay time 1 (SDRAM)	t_{CASD1}	—	25	ns	16.38
$\overline{\text{CAS}}$ delay time 3 (DRAM)	t_{CASD3}	3	25	ns	16.47
DQM delay time	t_{DQMD}	—	25	ns	16.38
CKE delay time	t_{CKED}	—	33	ns	16.37
$\overline{\text{CE}}$ delay time 2	t_{CED2}	3	25	ns	16.60
$\overline{\text{OE}}$ delay time 2	t_{OED2}	—	25	ns	16.60
$\overline{\text{IVECF}}$ delay time	t_{IVD}	—	25	ns	16.69
Address input setup time*	t_{ASIN}	25	—	ns	16.71
Address input hold time*	t_{AHIN}	10	—	ns	16.71
$\overline{\text{BS}}$ input setup time*	t_{BSS}	25	—	ns	16.71
$\overline{\text{BS}}$ input hold time*	t_{BSH}	10	—	ns	16.71
Read/write input setup time*	t_{RWS}	25	—	ns	16.71
Read/write input hold time*	t_{RWH}	10	—	ns	16.71
Data buffer on time	t_{DON}	—	25	ns	16.17, 16.39, 16.48, 16.61
Data buffer off time	t_{DOF}	—	25	ns	16.17, 16.39, 16.48, 16.61

Note: When the external addresses monitor function is used, the PLL must be on.



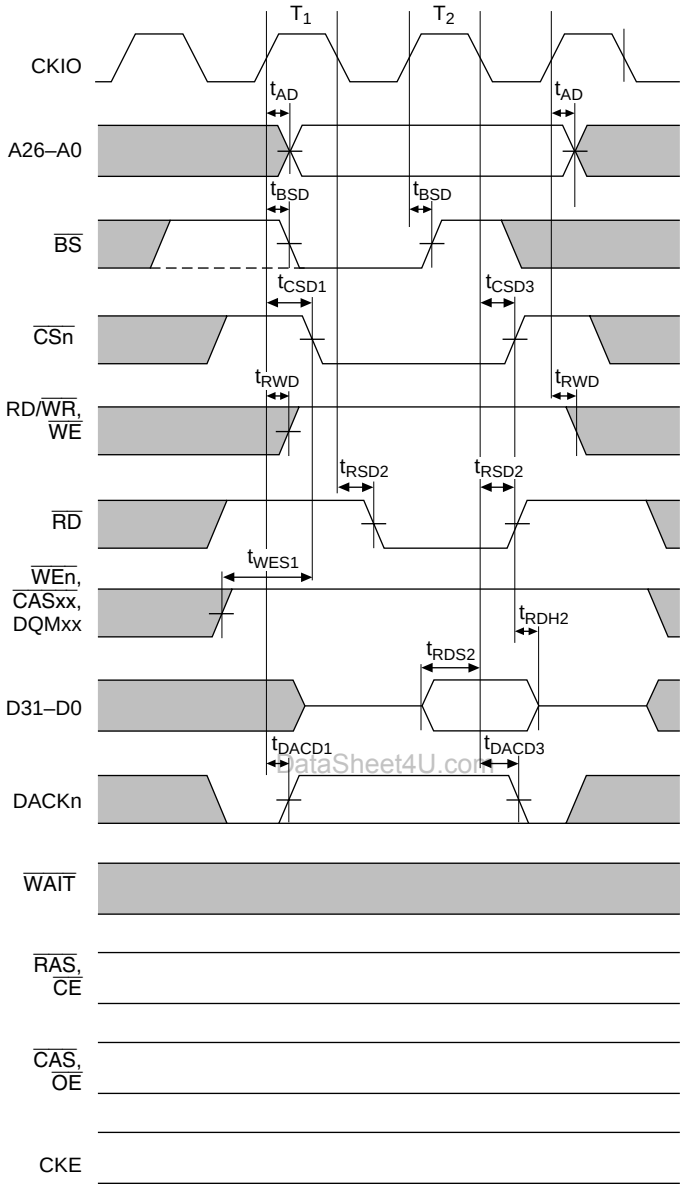
Notes: 1. The dotted line shows the waveform when synchronous DRAM is connected.
2. t_{RDH2} is specified from the rise of $\overline{CSn}$ or $\overline{RD}$, whichever is first.
3. The DACKn waveform shown is for the case where active-high has been specified.

Figure 16.14 Basic Read Cycle (No Waits, PLL On)



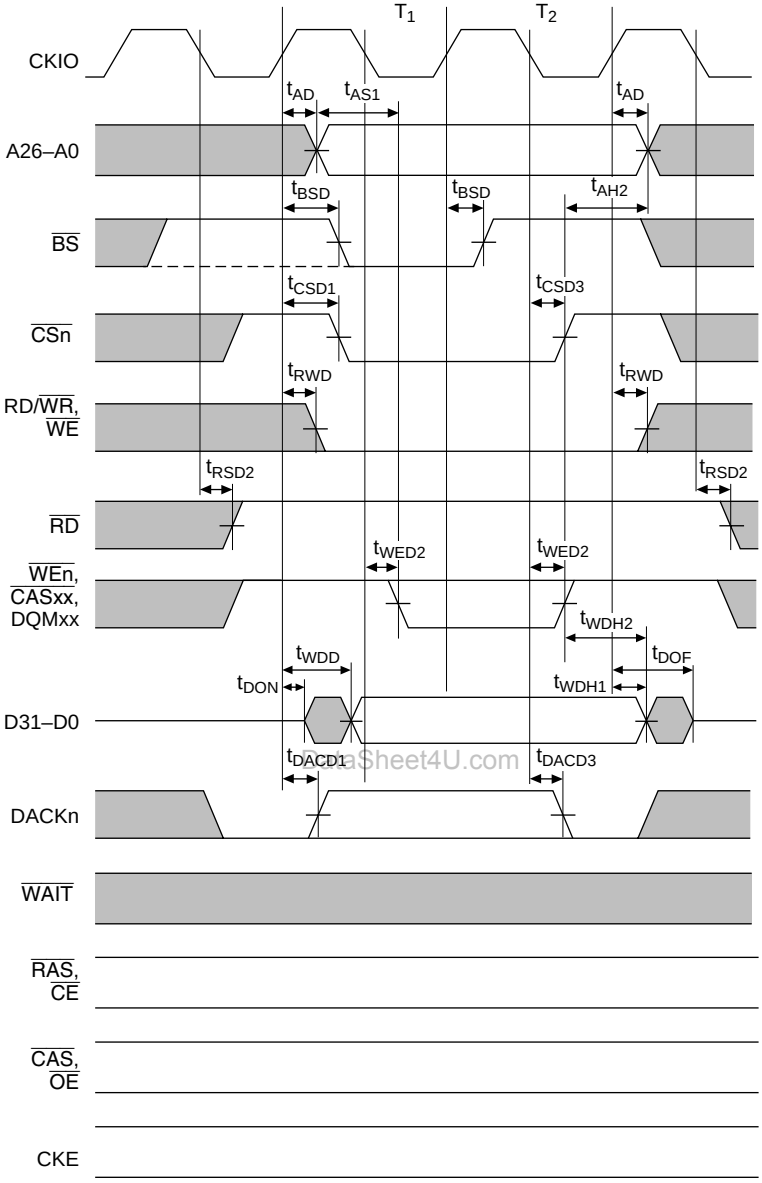
Notes: 1. The dotted line shows the waveform when synchronous DRAM is connected.
2. The DACKn waveform shown is for the case where active-high has been specified.

Figure 16.15 Basic Write Cycle (No Waits, PLL On)



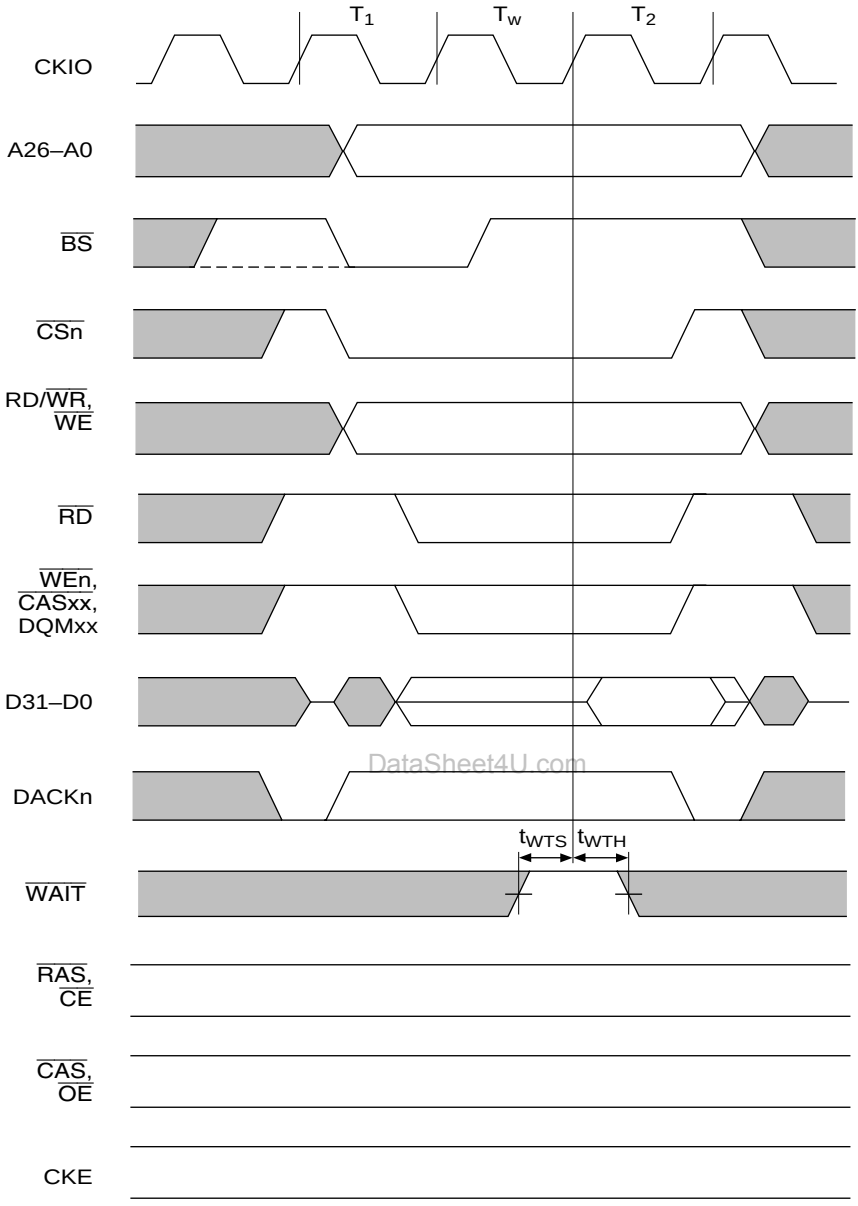
Notes: 1. The dotted line shows the waveform when synchronous DRAM is connected.
2. t_{RDH2} is specified from the rise of $\overline{CSn}$ or $\overline{RD}$, whichever is first.
3. The DACKn waveform shown is for the case where active-high has been specified.

Figure 16.16 Basic Read Cycle (No Waits, PLL Off)



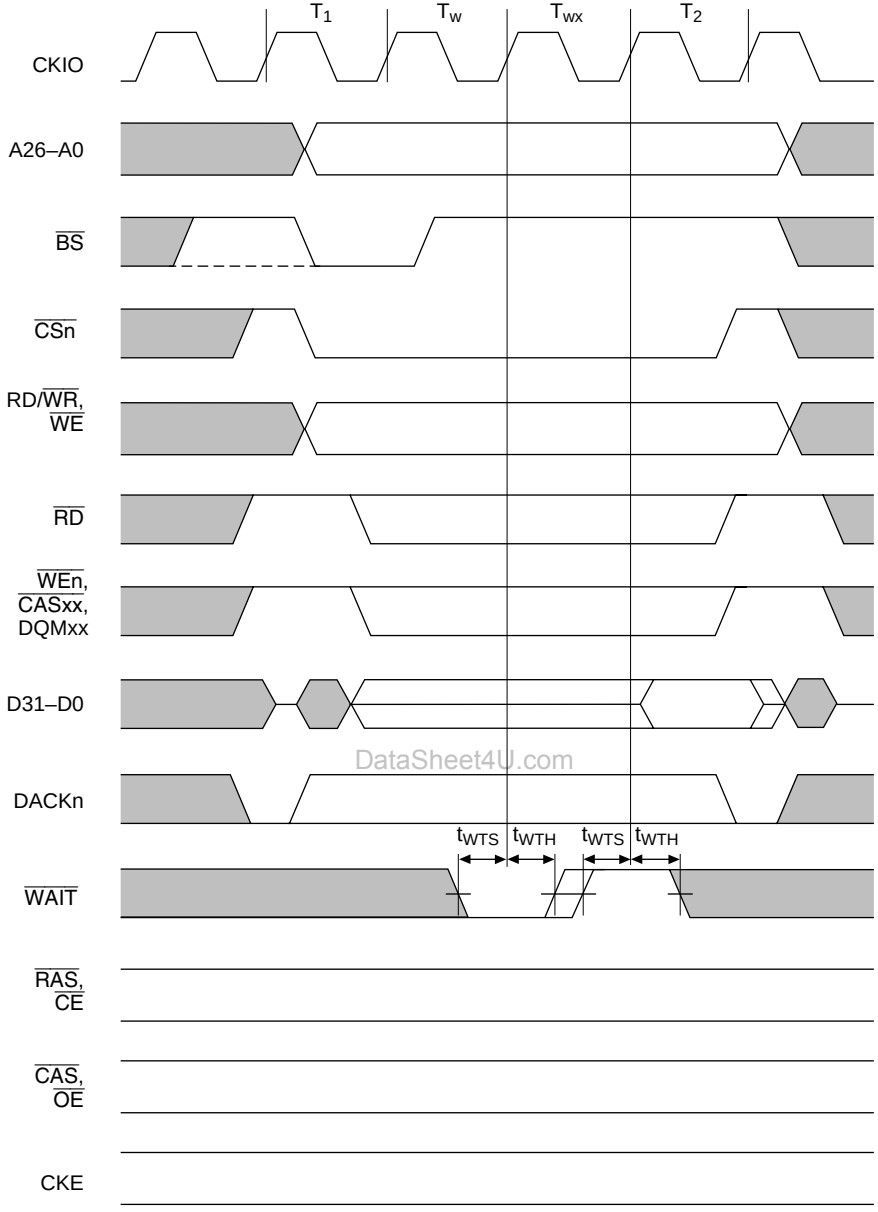
Notes: 1. The dotted line shows the waveform when synchronous DRAM is connected.
2. The DACKn waveform shown is for the case where active-high has been specified.

Figure 16.17 Basic Write Cycle (No Waits, PLL Off)



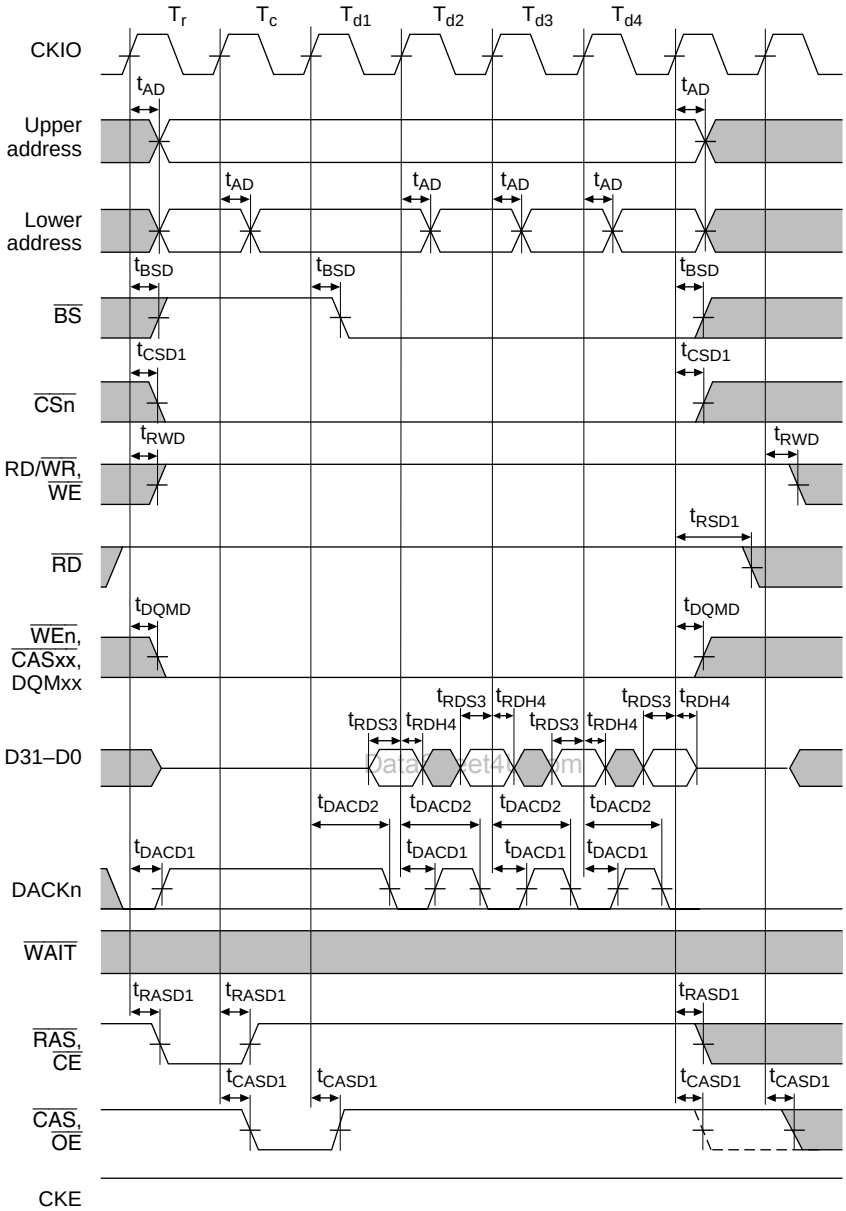
Notes: 1. The dotted line shows the waveform when synchronous DRAM is connected.
2. The DACKn waveform shown is for the case where active-high has been specified.

Figure 16.18 Basic Bus Cycle (1 Wait Cycle)



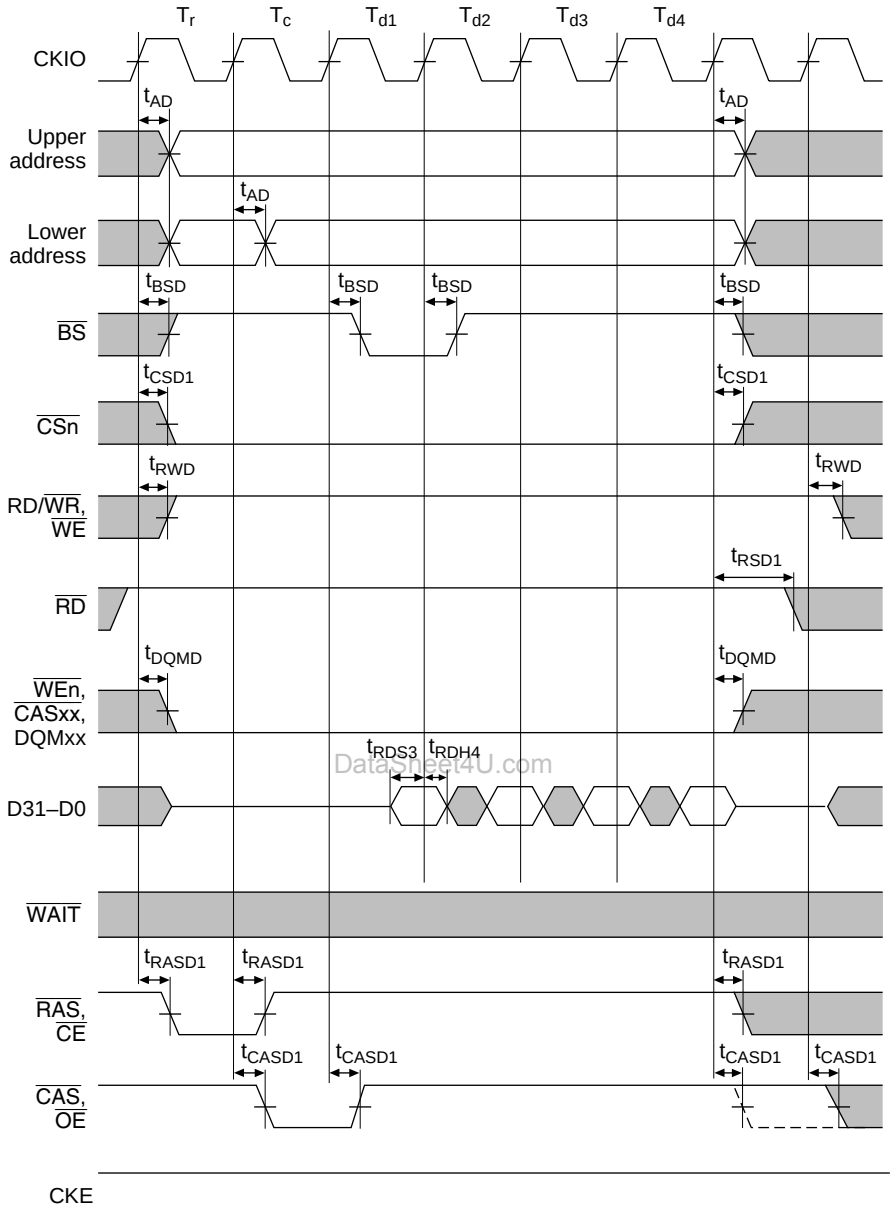
Notes: 1. The dotted line shows the waveform when synchronous DRAM is connected.
2. The DACKn waveform shown is for the case where active-high has been specified.

Figure 16.19 Basic Bus Cycle (External Wait Input)



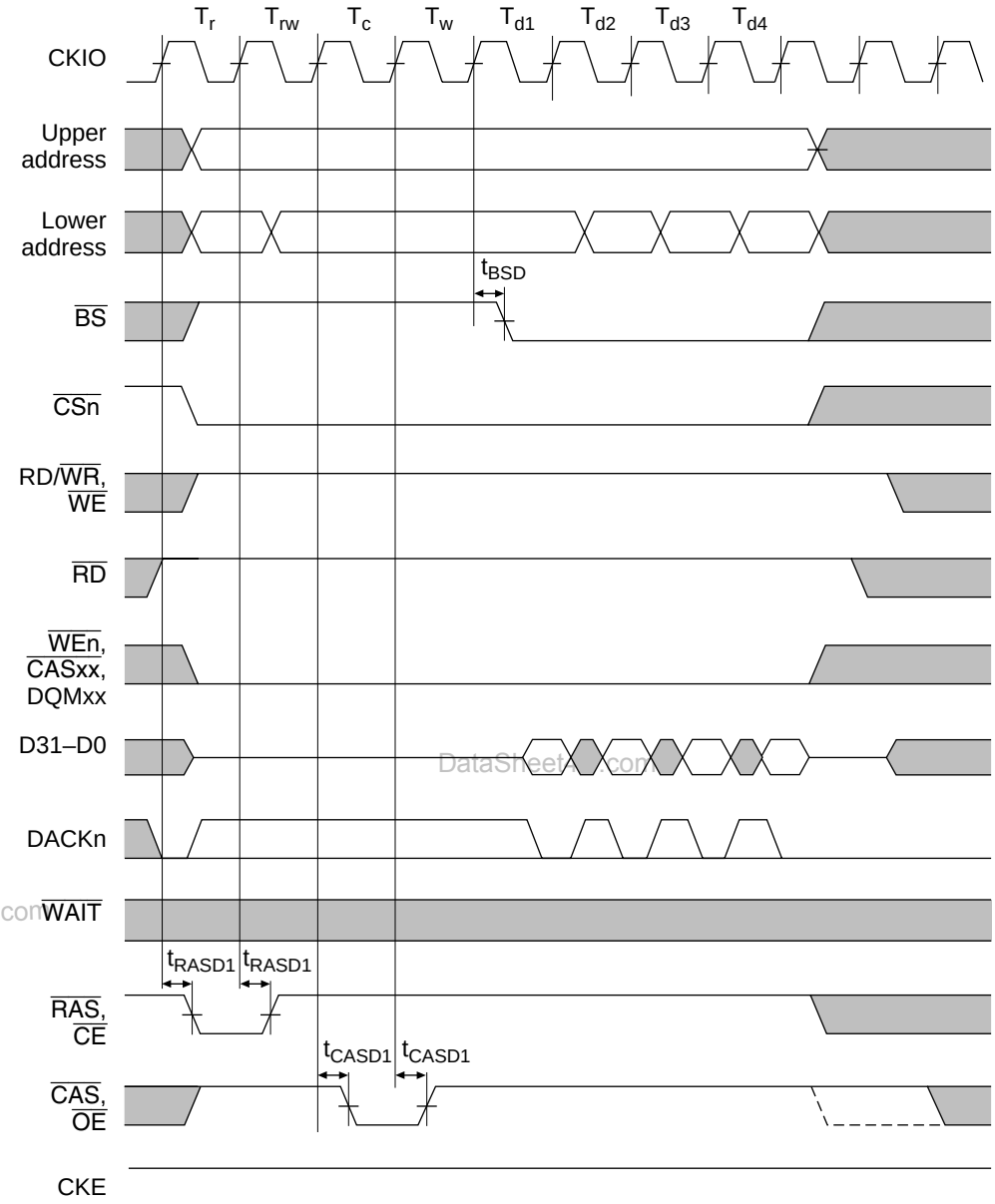
Notes: 1. The dotted line shows the waveform when synchronous DRAM in another CS space is accessed.
2. The DACKn waveform shown is for the case where active-high has been specified.

Figure 16.20 Synchronous DRAM Read Bus Cycle (RCD = 1 Cycle, CAS Latency = 1 Cycle, Bursts = 4, PLL On)



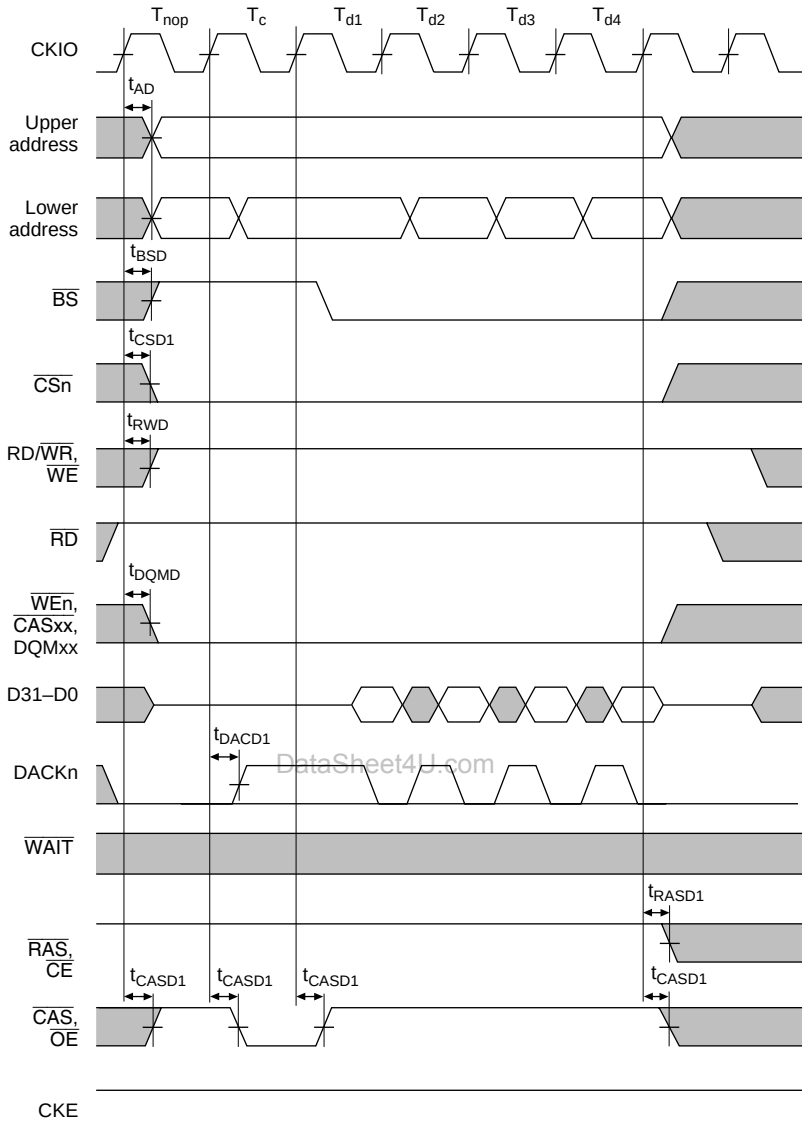
Note: The dotted line shows the waveform when synchronous DRAM in another CS space is accessed.

Figure 16.21 Synchronous DRAM Single Read Bus Cycle (RCD = 1 Cycle, CAS Latency = 1 Cycle, Bursts = 4, PLL On)



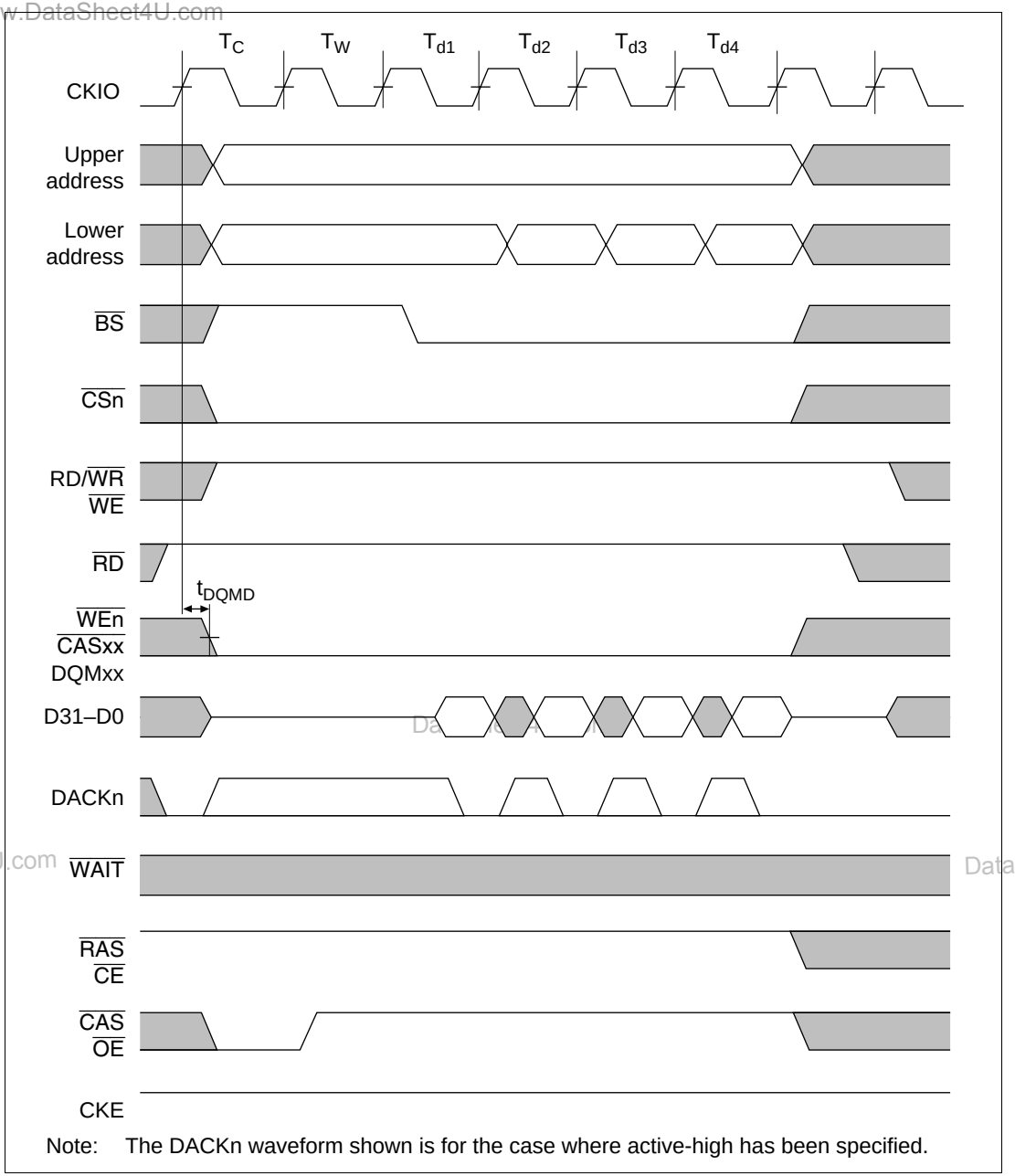
Notes: 1. The dotted line shows the waveform when synchronous DRAM in another CS space is accessed.
2. The DACKn waveform shown is for the case where active-high has been specified.

Figure 16.22 Synchronous DRAM Read Bus Cycle (RCD = 2 Cycles, CAS Latency = 2 Cycles, Bursts = 4)



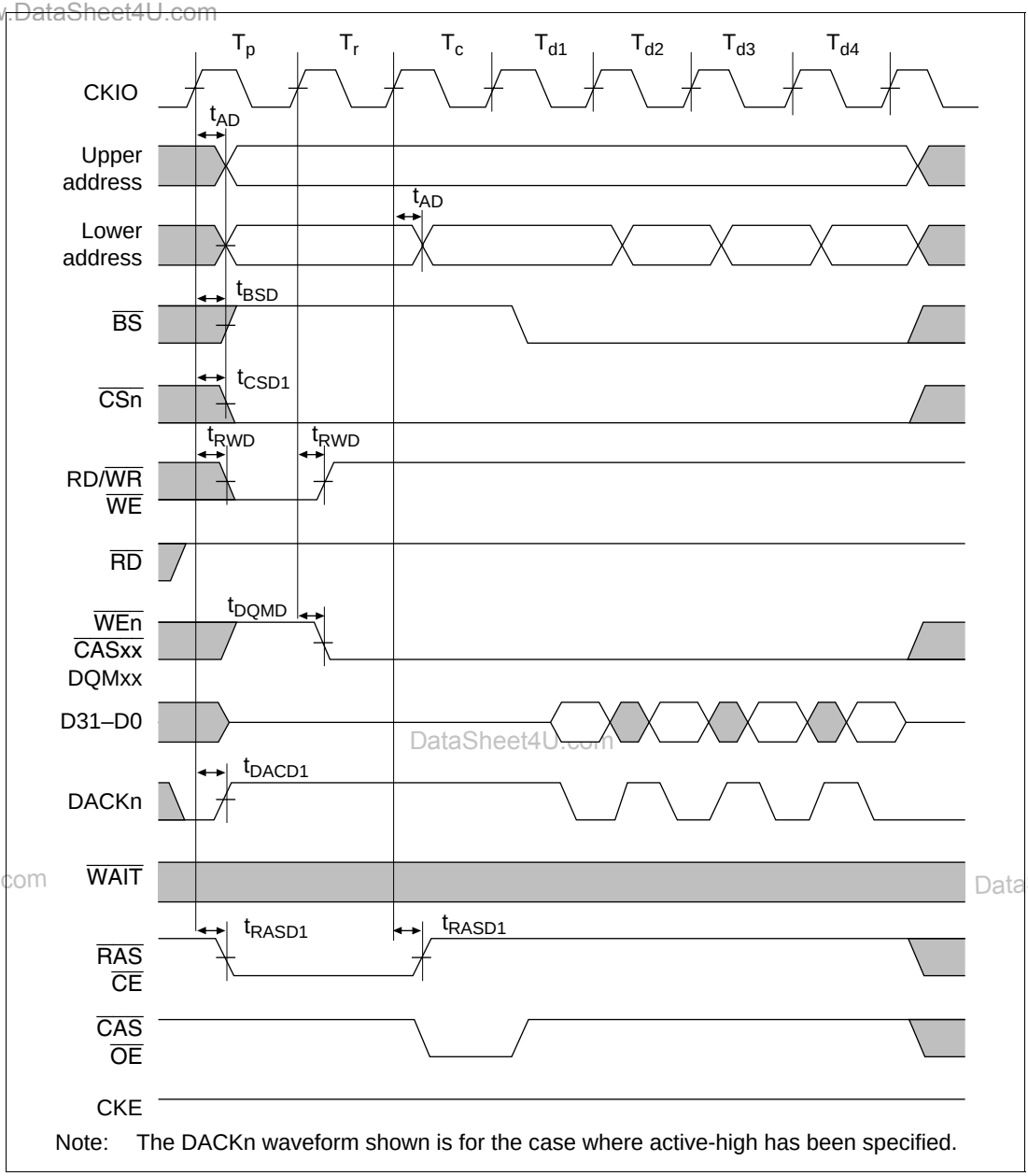
Note: The DACKn waveform shown is for the case where active-high has been specified.

Figure 16.23 Synchronous DRAM Read Bus Cycle (Bank Active, Same Row Access, CAS Latency = 1 Cycle)



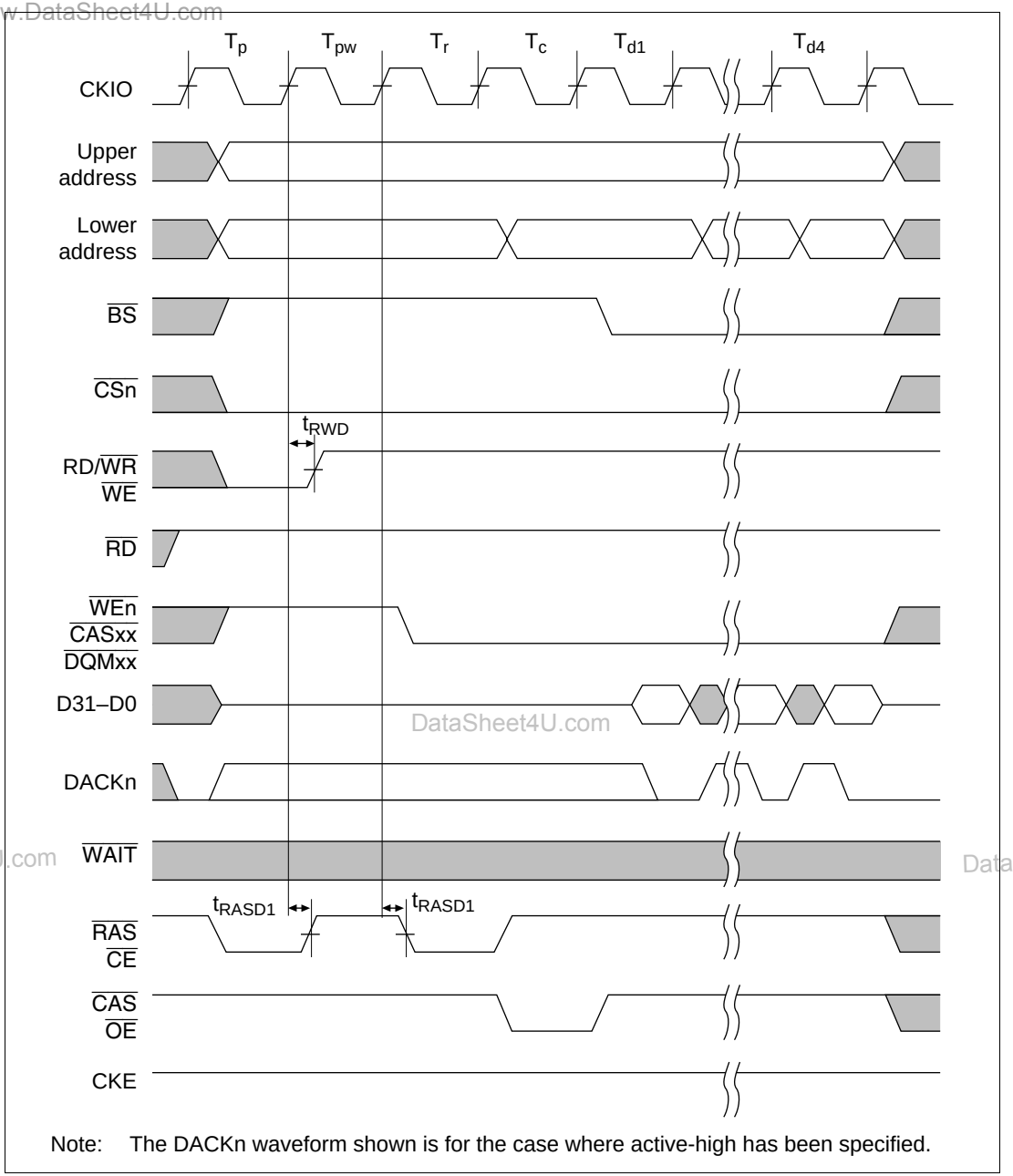
Note: The DACKn waveform shown is for the case where active-high has been specified.

Figure 16.24 Synchronous DRAM Read Bus Cycle (Bank Active, Same Row Access, CAS Latency = 2 Cycles)



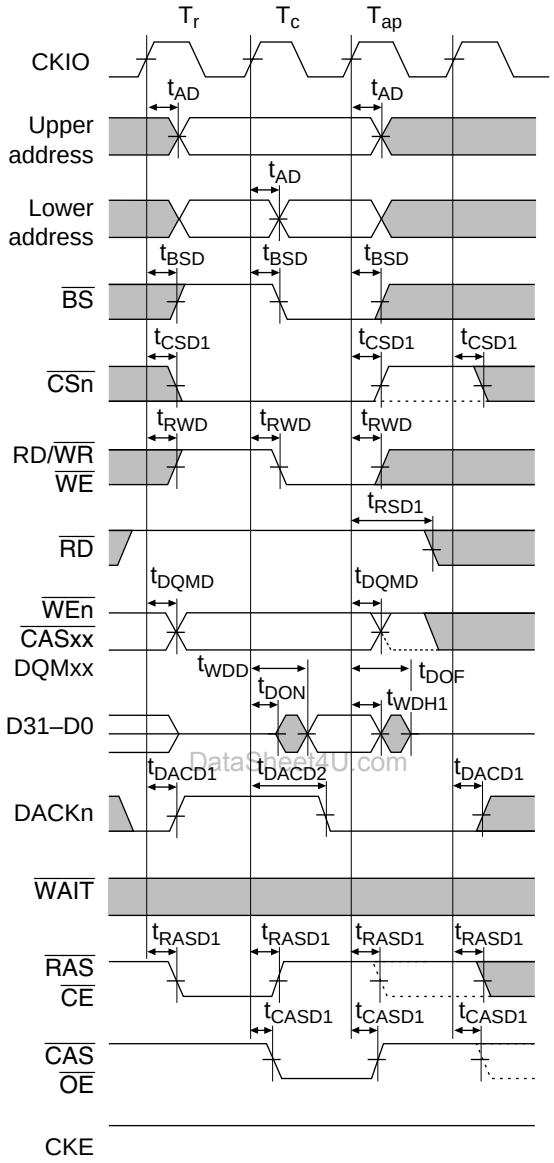
Note: The DACKn waveform shown is for the case where active-high has been specified.

Figure 16.25 Synchronous DRAM Read Bus Cycle (Bank Active, Different Row Access, TRP = 1 Cycle, RCD = 1 Cycle, CAS Latency = 1 Cycle)



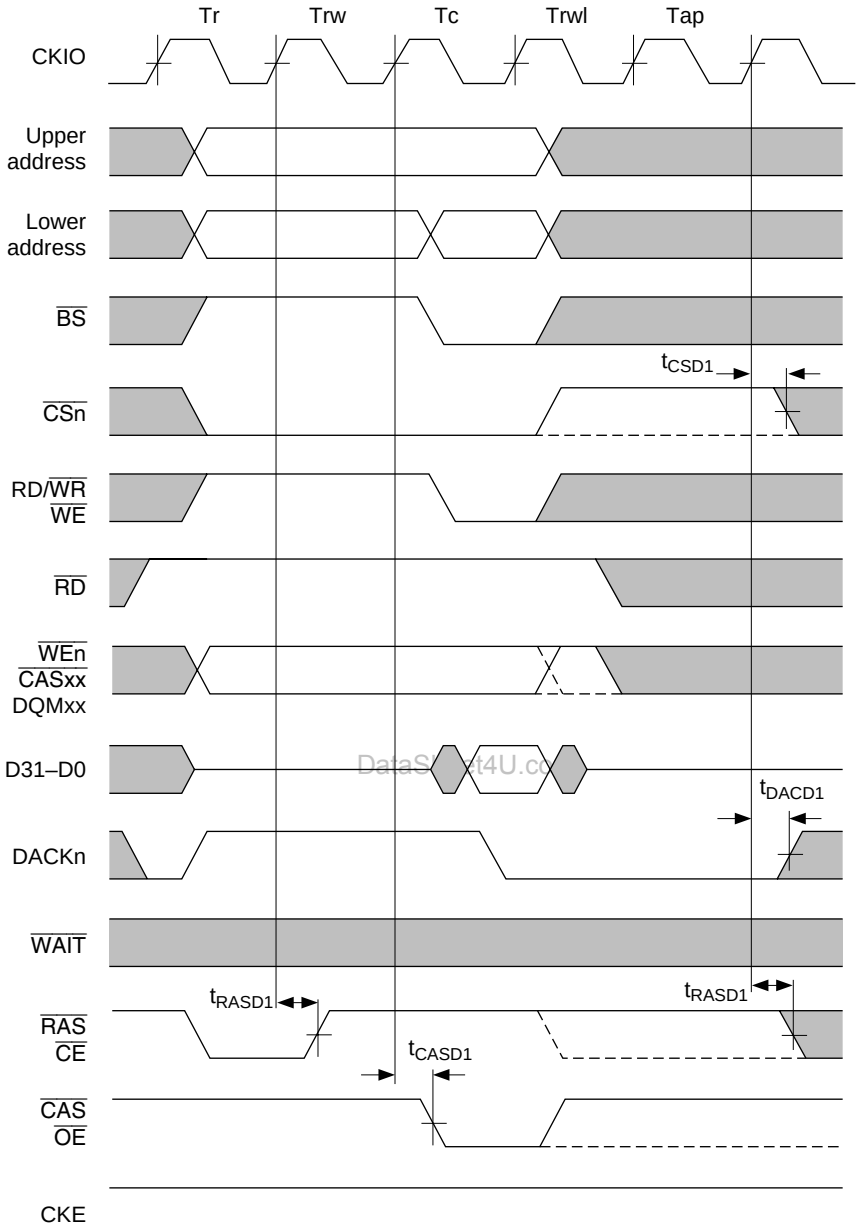
Note: The DACKn waveform shown is for the case where active-high has been specified.

Figure 16.26 Synchronous DRAM Read Bus Cycle (Bank Active, Different Row Access, TRP = 2 Cycles, RCD = 1 Cycle, CAS Latency = 1 Cycle)



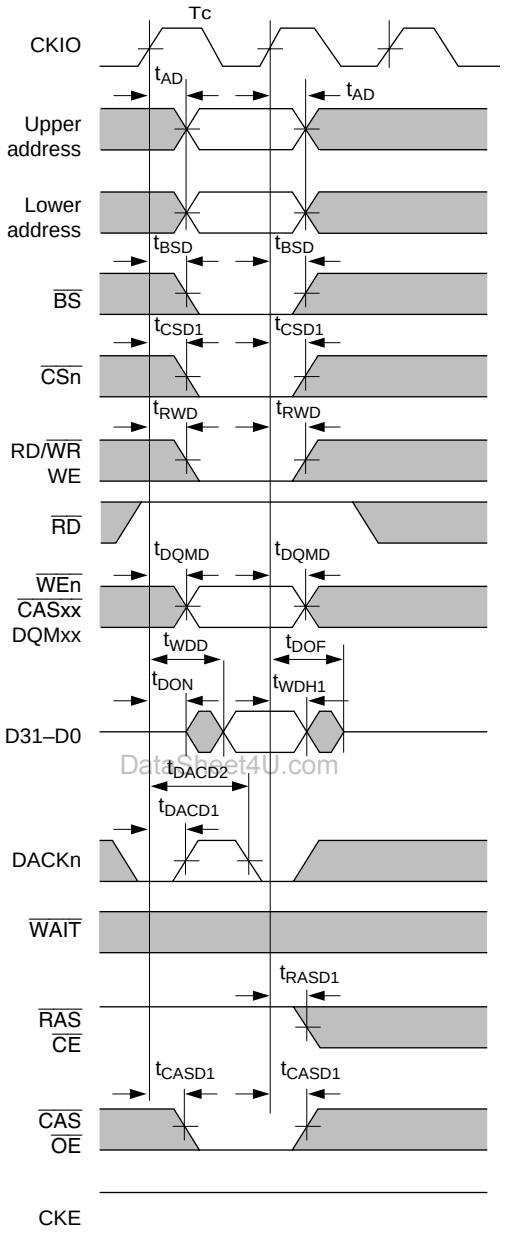
Notes: 1. Dotted lines show the waveforms when synchronous DRAM in another CS space is accessed.
2. The DACKn waveform shown is for the case where active-high has been specified.

Figure 16.27 Synchronous DRAM Write Bus Cycle (RCD = 1 Cycle, TRWL = 1 Cycle, PLL On)



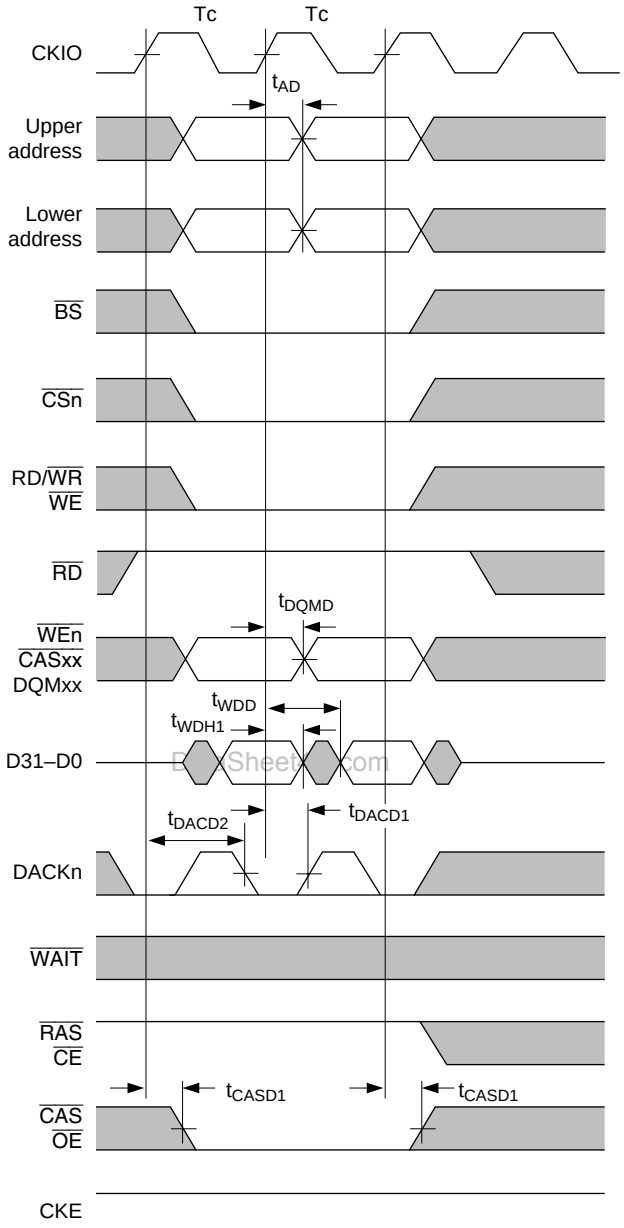
Notes: 1. Dotted lines show the waveforms when synchronous DRAM in another CS space is accessed.
2. The DACKn waveform shown is for the case where active-high has been specified.

Figure 16.28 Synchronous DRAM Write Bus Cycle (RCD = 2 Cycles, TRWL = 2 Cycles)



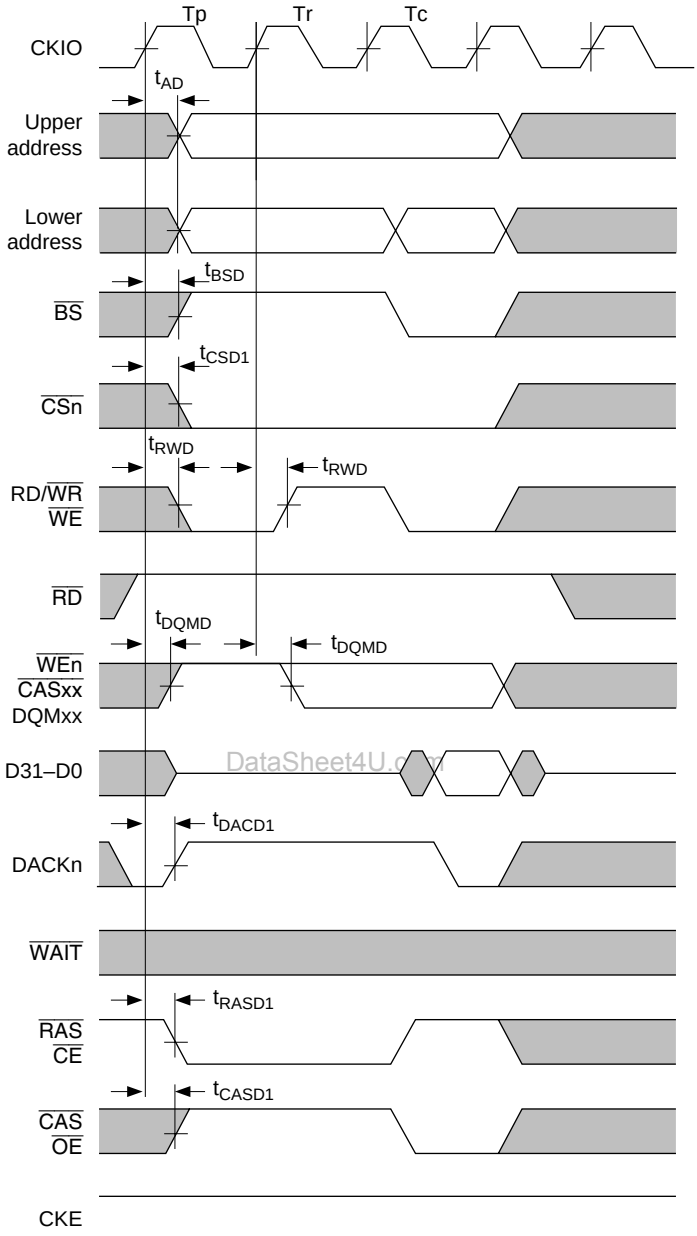
Note: The DACKn waveform shown is for the case where active-high has been specified.

Figure 16.29 Synchronous DRAM Write Bus Cycle (Bank Active, Same Row Access)



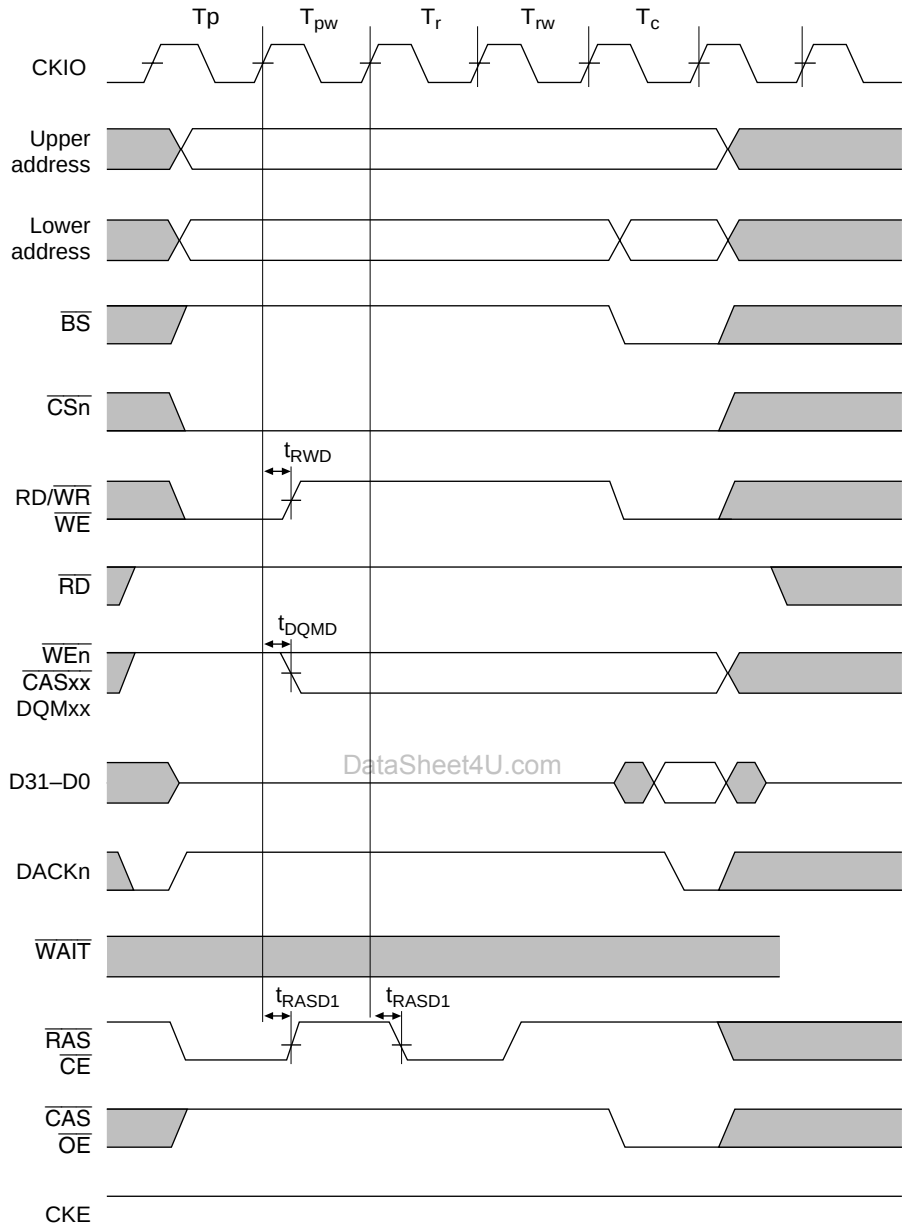
Note: The DACKn waveform shown is for the case where active-high has been specified.

Figure 16.30 Synchronous DRAM Consecutive Write Cycles (Bank Active, Same Row Access)



Note: The DACKn waveform shown is for the case where active-high has been specified.

Figure 16.31 Synchronous DRAM Write Bus Cycle (Bank Active, Different Row Access, TRP = 1 Cycle, RCD = 1 Cycle)



Note: The DACKn waveform shown is for the case where active-high has been specified.

Figure 16.32 Synchronous DRAM Write Bus Cycle (Bank Active, Different Row Access, TRP = 2 Cycles, RCD = 2 Cycles)

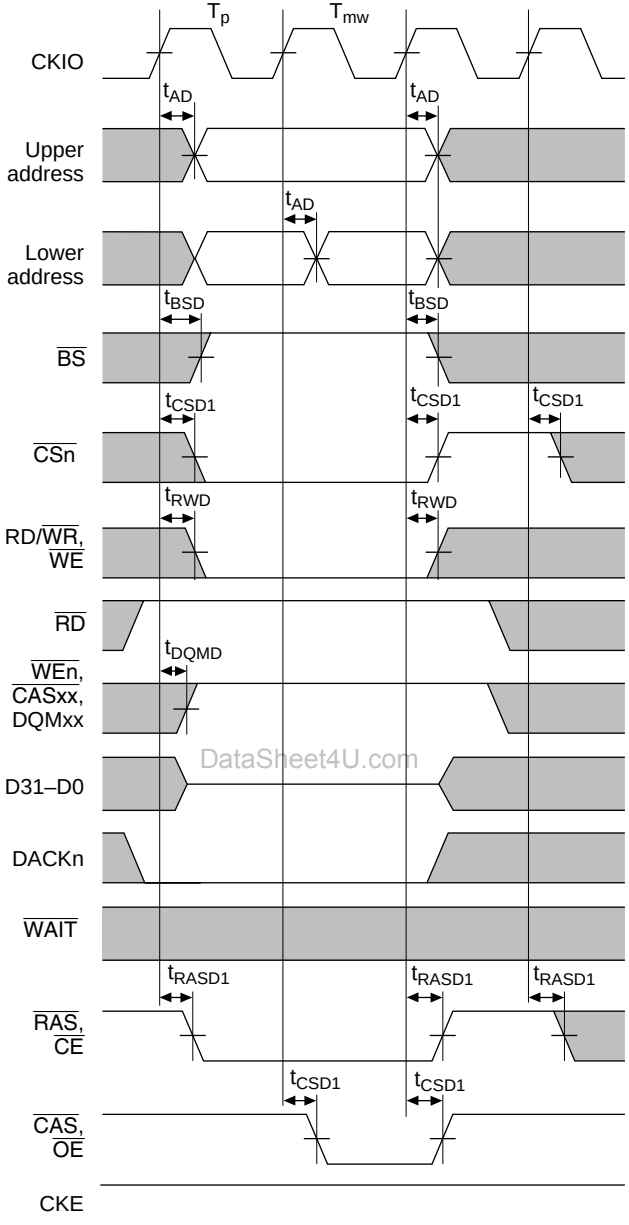


Figure 16.33 Synchronous DRAM Mode Register Write Cycle (TRP = 1 Cycle)

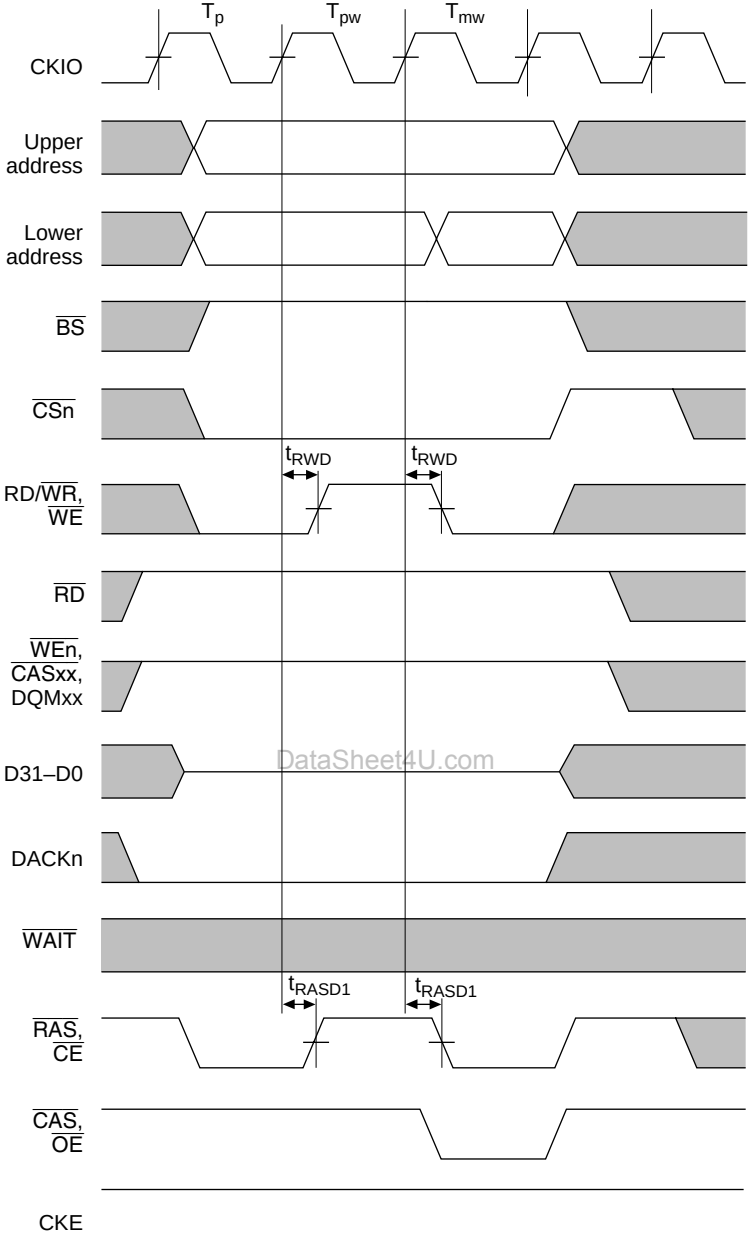
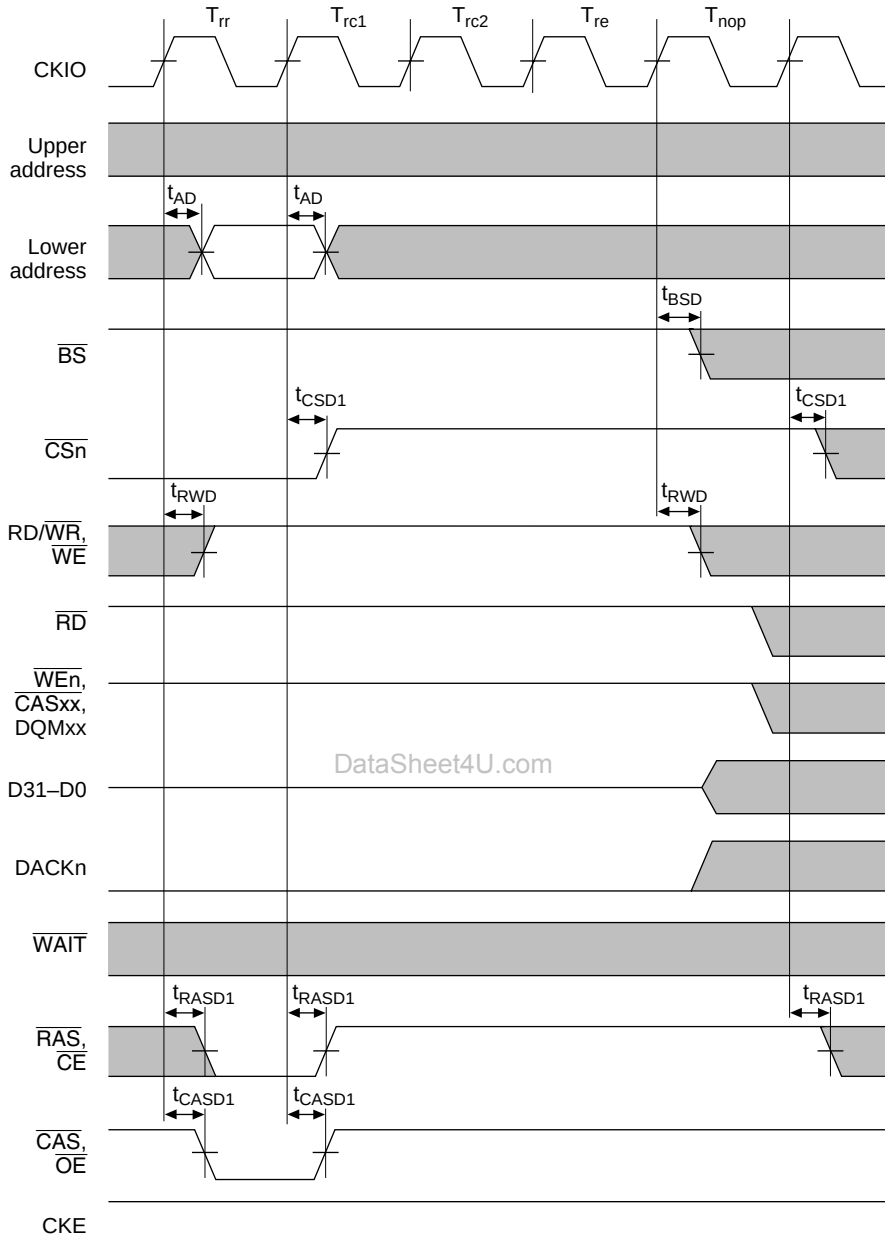


Figure 16.34 Synchronous DRAM Mode Register Write Cycle (TRP = 2 Cycles)



Note: A precharge cycle always precedes the auto-refresh cycle by the number of cycles specified by TRP.

Figure 16.35 Synchronous DRAM Auto-Refresh Cycle (TRAS = 2 Cycles)

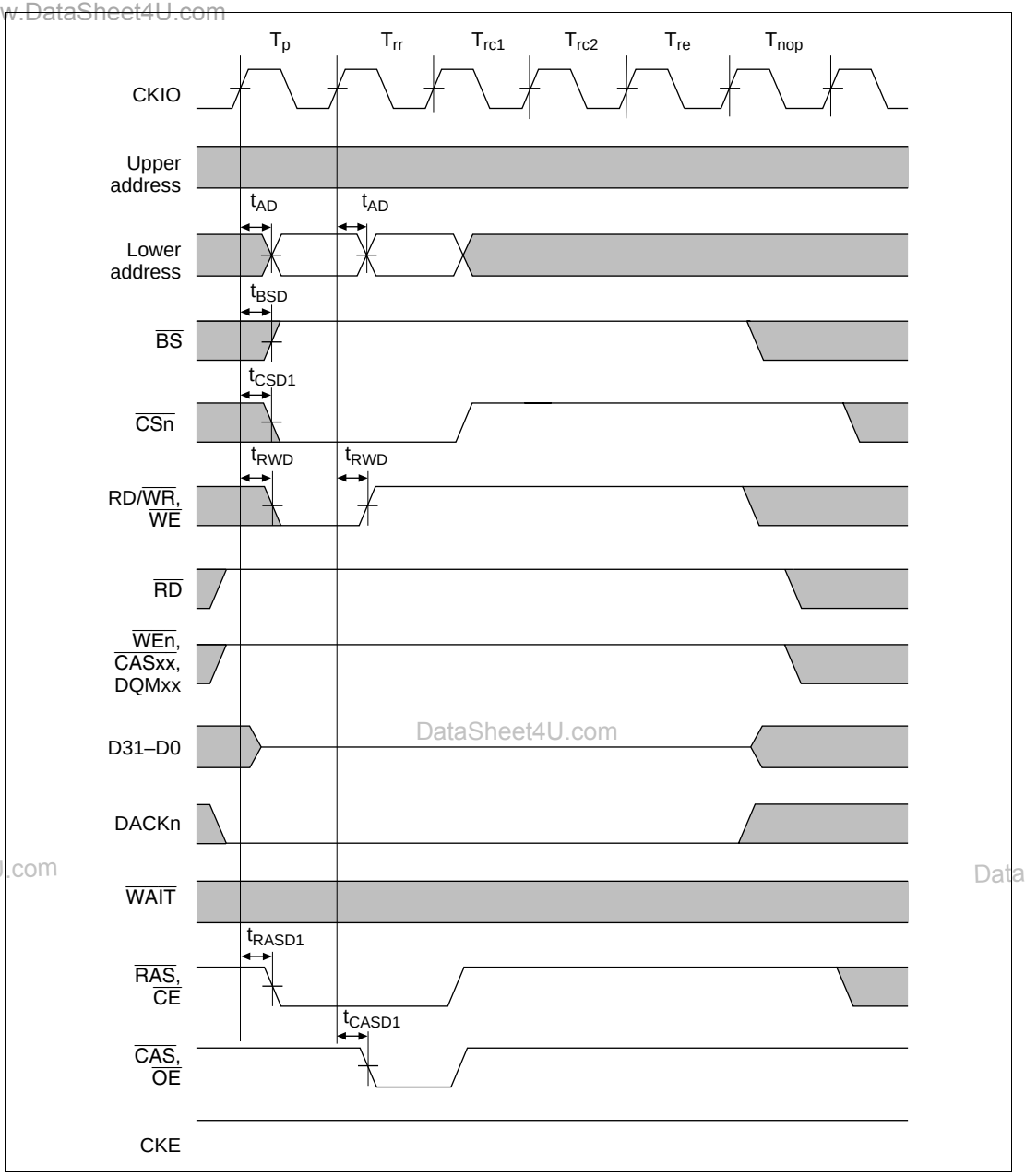
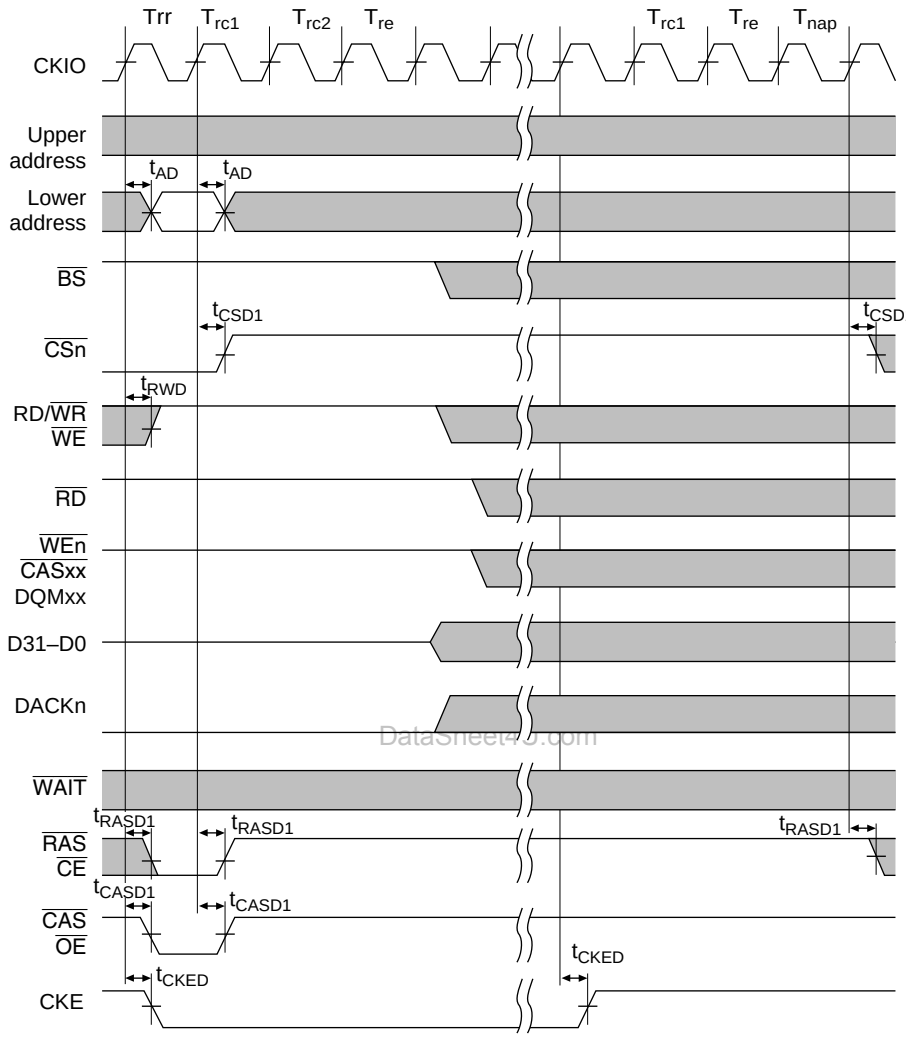
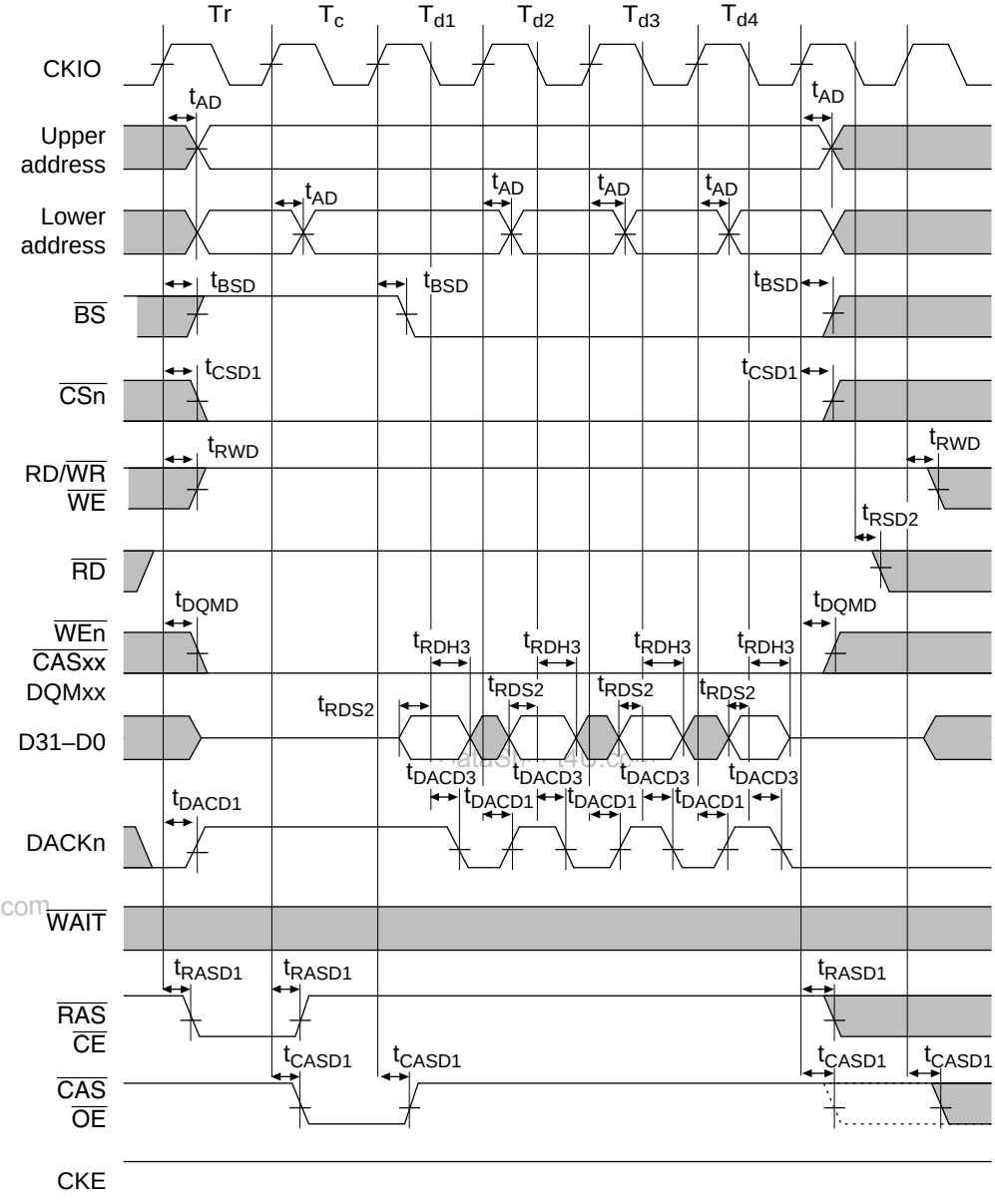


Figure 16.36 Synchronous DRAM Auto-Refresh Cycle (Shown From Precharge Cycle, TRP = 1 Cycle, TRAS = 2 Cycles)



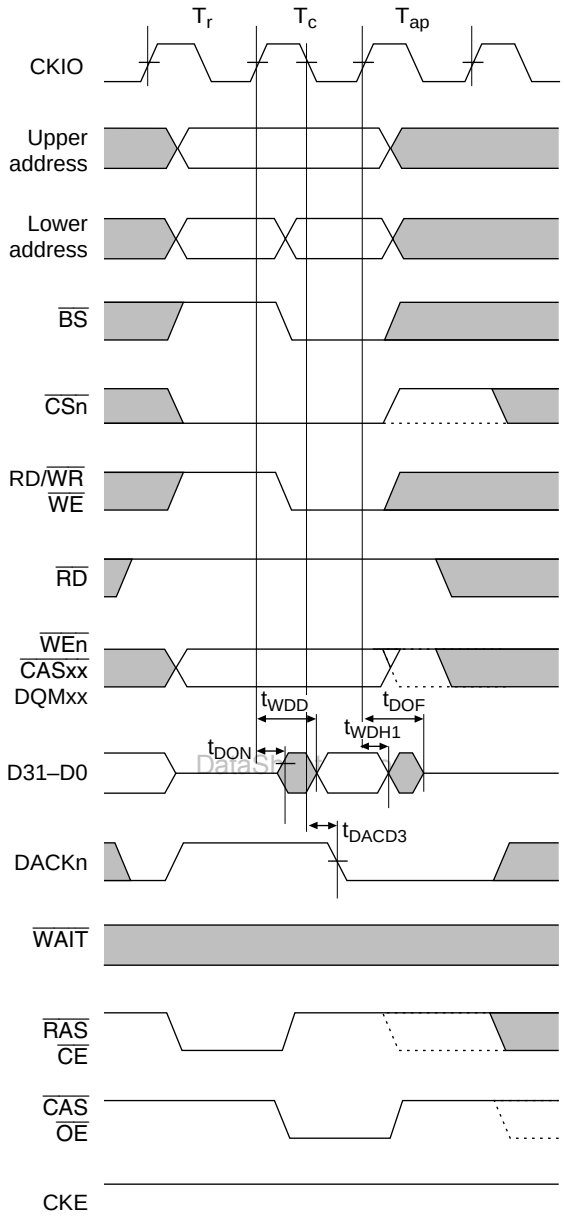
Note: A precharge cycle always precedes the self-refresh cycle by the number of cycles specified by TRP.

Figure 16.37 Synchronous DRAM Self-Refresh Cycle (TRAS = 2)



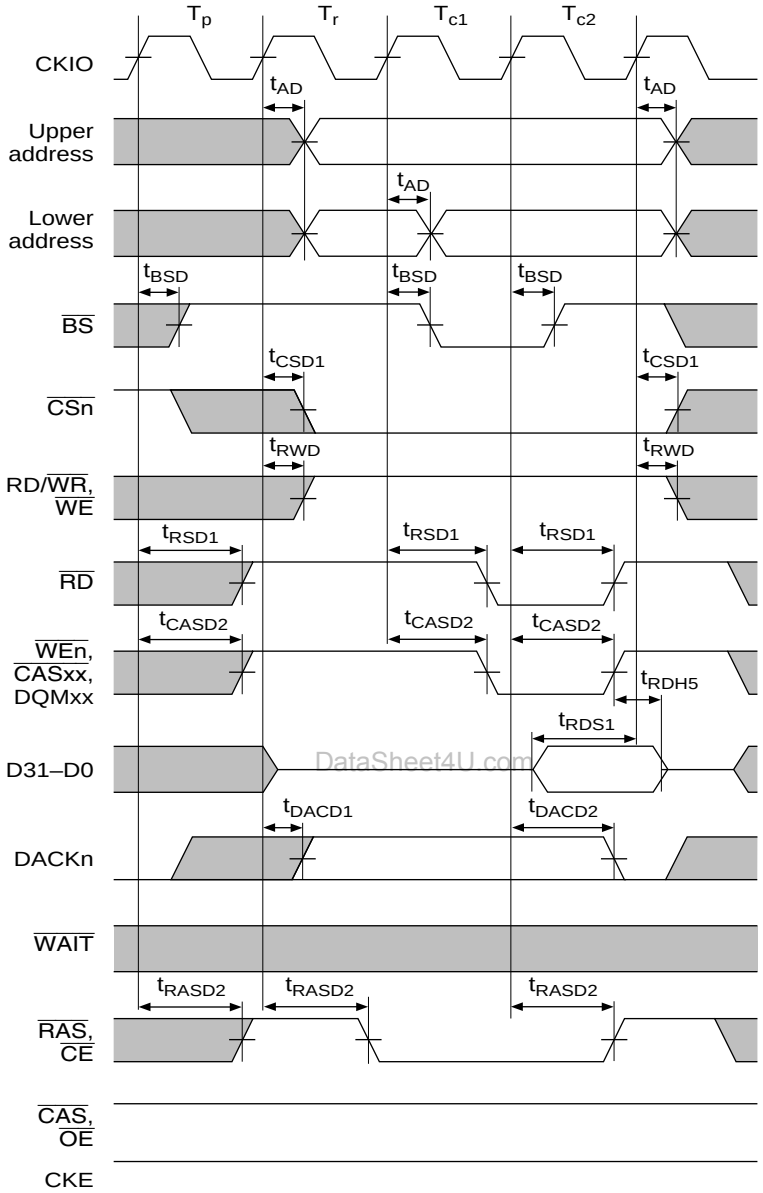
Notes: 1. The dotted line shows the waveform when synchronous DRAM in another CS space is accessed.
2. The DACKn waveform shown is for the case where active-high has been specified.

Figure 16.38 Synchronous DRAM Read Bus Cycle (RCD = 1 Cycle, CAS Latency = 1 Cycle, TRP = 1 Cycle, Bursts = 4, PLL Off)



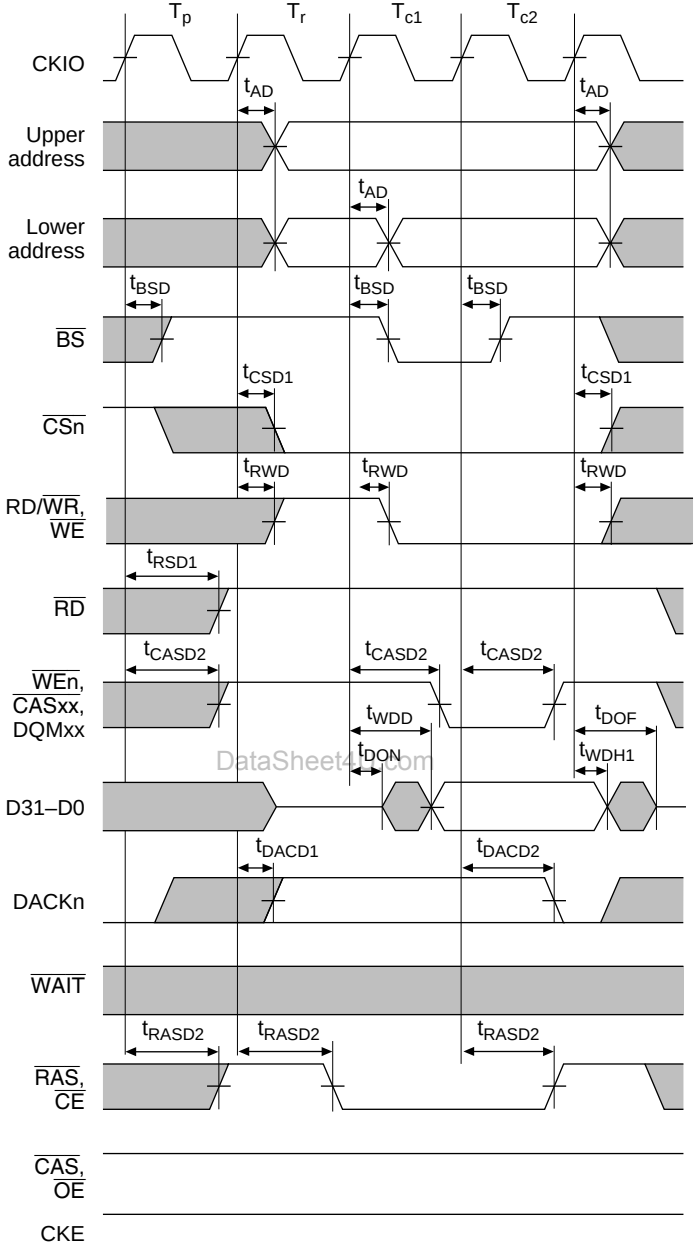
Notes: 1. Dotted lines show the waveforms when synchronous DRAM in another CS space is accessed.
2. The DACKn waveform shown is for the case where active-high has been specified.

Figure 16.39 Synchronous DRAM Write Bus Cycle (RCD = 1 Cycle, TRWL = 1 Cycle, PLL Off)



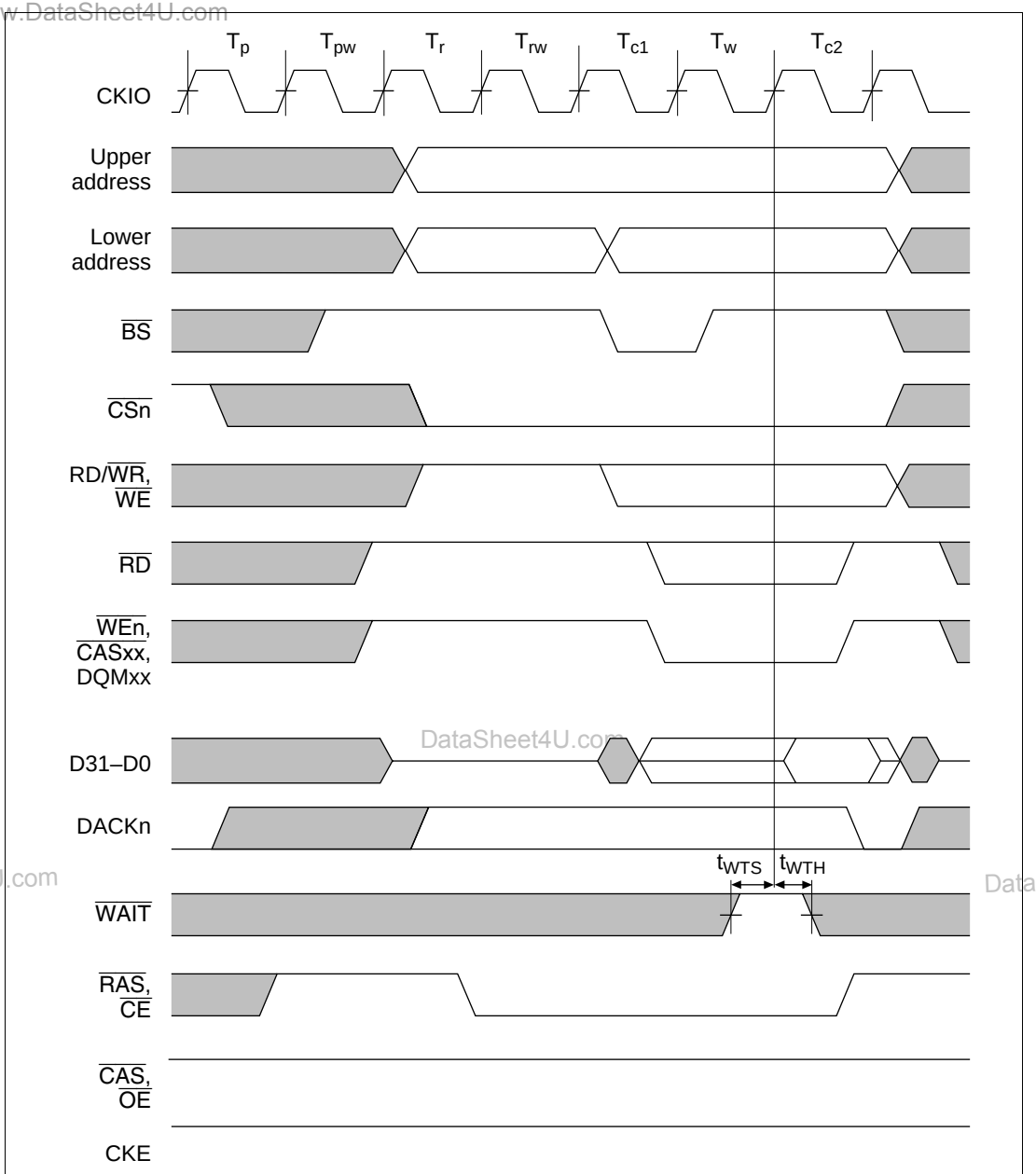
Notes: 1. t_{RDH5} is specified from the rise of $\overline{RD}$ or $\overline{CASxx}$, whichever is first.
2. The DACKn waveform shown is for the case where active-high has been specified.

Figure 16.40 DRAM Read Cycle (TRP = 1 Cycle, RCD = 1 Cycle, No Waits, PLL On)



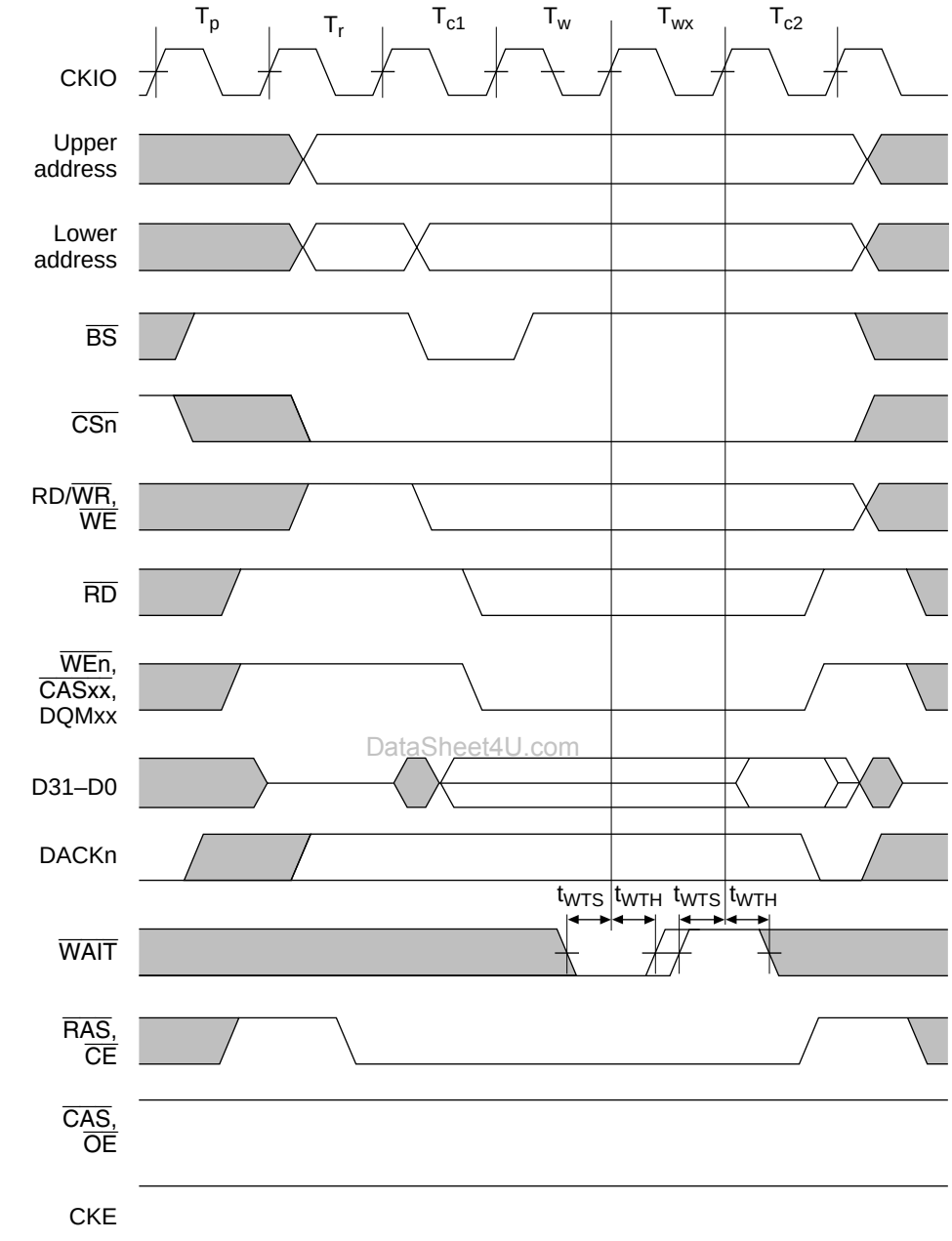
Note: The DACKn waveform shown is for the case where active-high has been specified.

Figure 16.41 DRAM Write Cycle
(TRP = 1 Cycle, RCD = 1 Cycle, No Waits, PLL On)



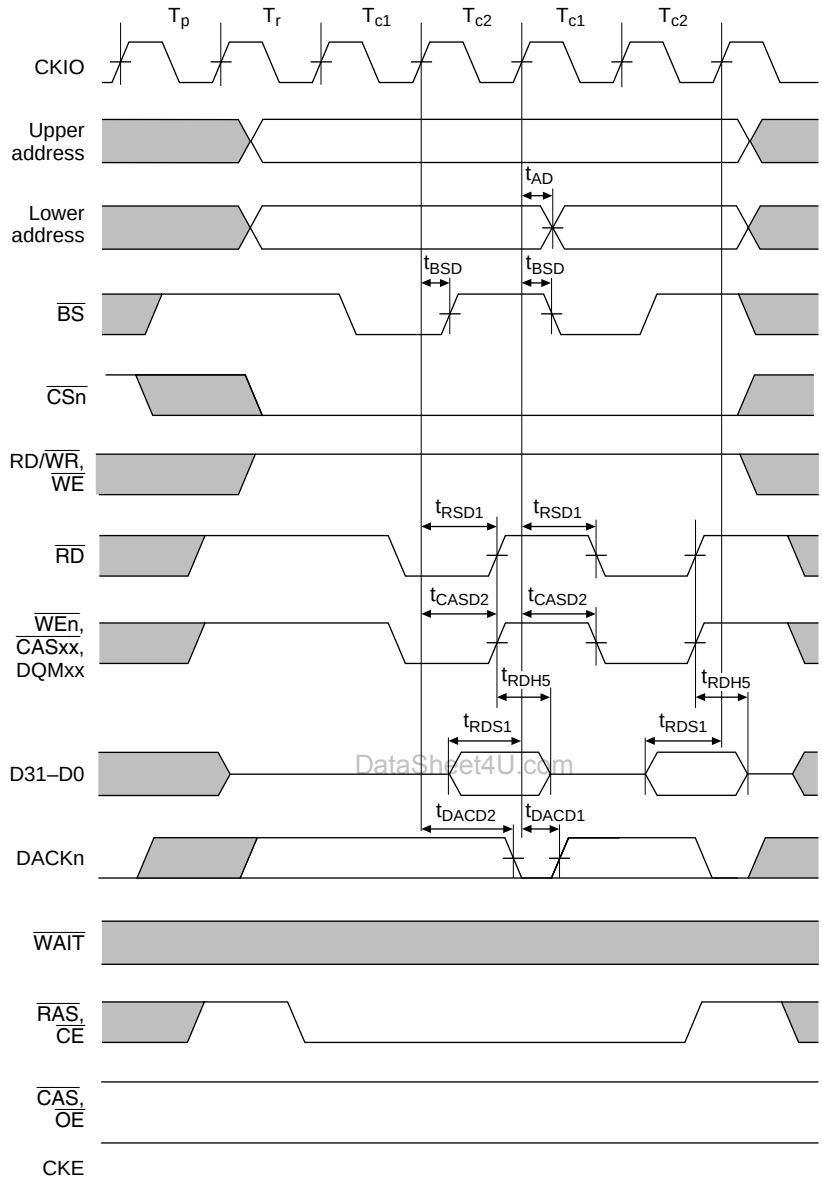
Note: The DACKn waveform shown is for the case where active-high has been specified.

Figure 16.42 DRAM Bus Cycle (TRP = 2 Cycles, RCD = 2 Cycles, 1 Wait)



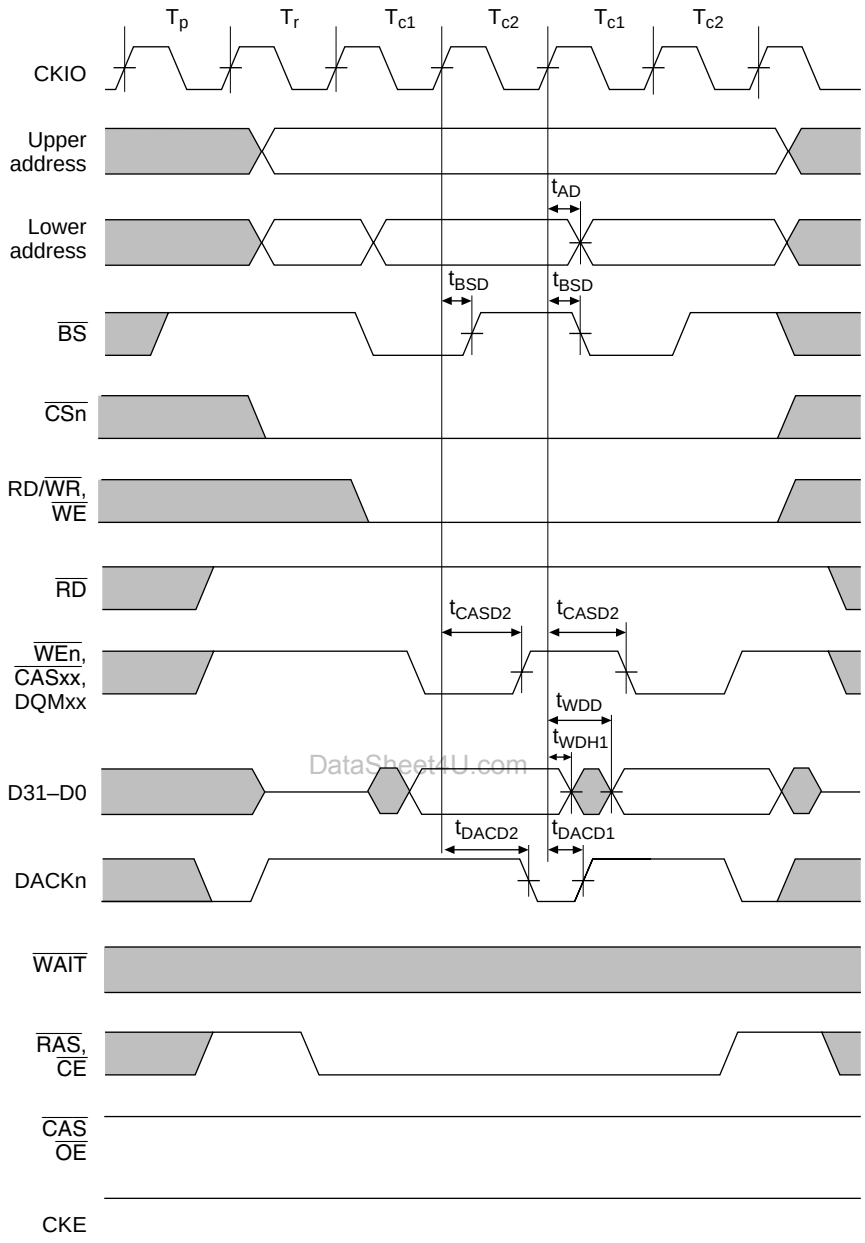
Note: The DACKn waveform shown is for the case where active-high has been specified.

Figure 16.43 DRAM Bus Cycle (TRP = 1 Cycle, RCD = 1 Cycle, External Wait Input)



Notes: 1. t_{RDH5} is specified from the rise of $\overline{RD}$ or $\overline{CASxx}$, whichever is first.
2. The DACKn waveform shown is for the case where active-high has been specified.

Figure 16.44 DRAM Burst Read Cycle
(TRP = 1 Cycle, RCD = 1 Cycle, No Waits, PLL On)



Note: The DACKn waveform shown is for the case where active-high has been specified.

Figure 16.45 DRAM Burst Write Cycle (TRP = 1 Cycle, RCD = 1 Cycle, No Waits, PLL On)

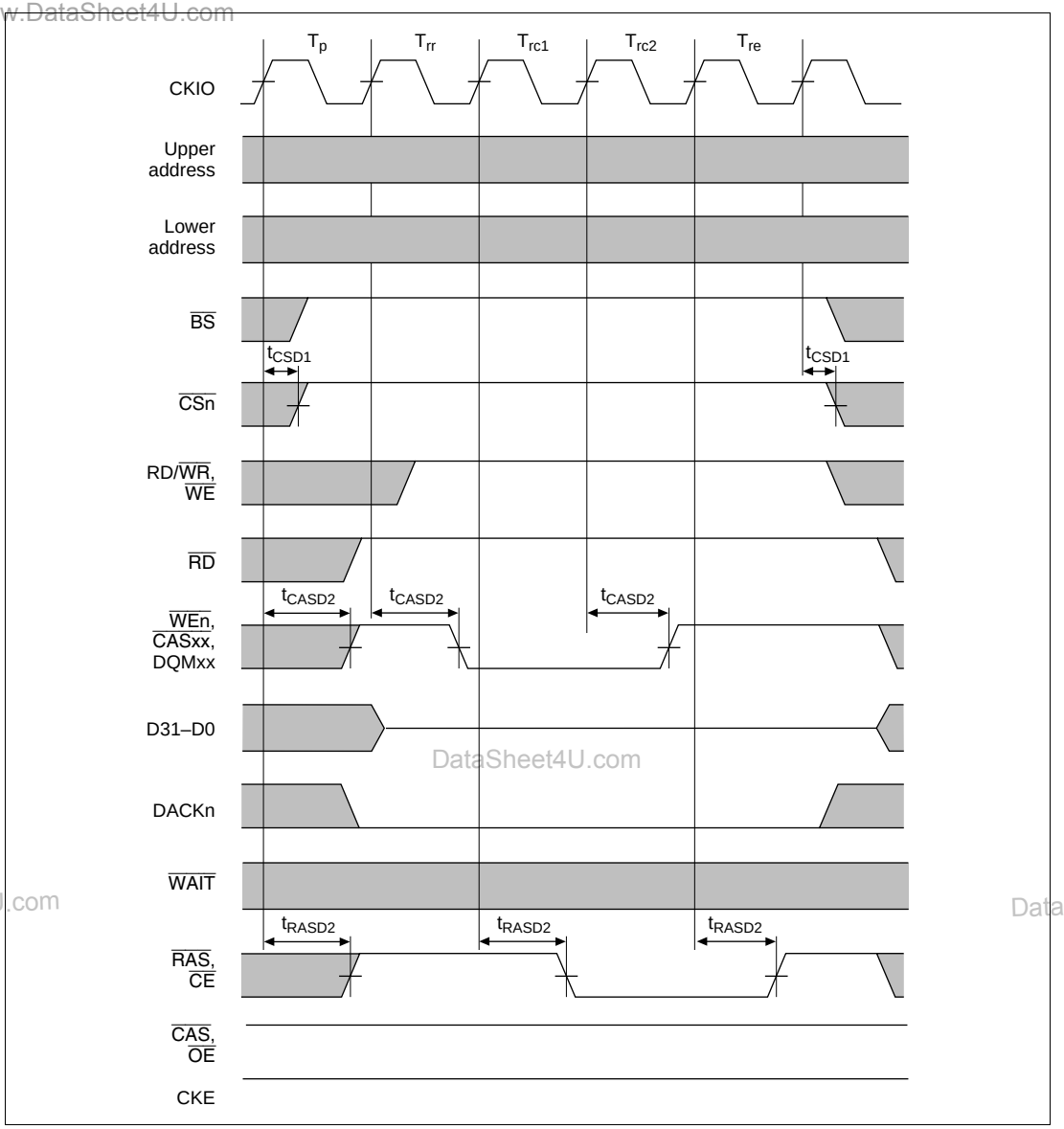
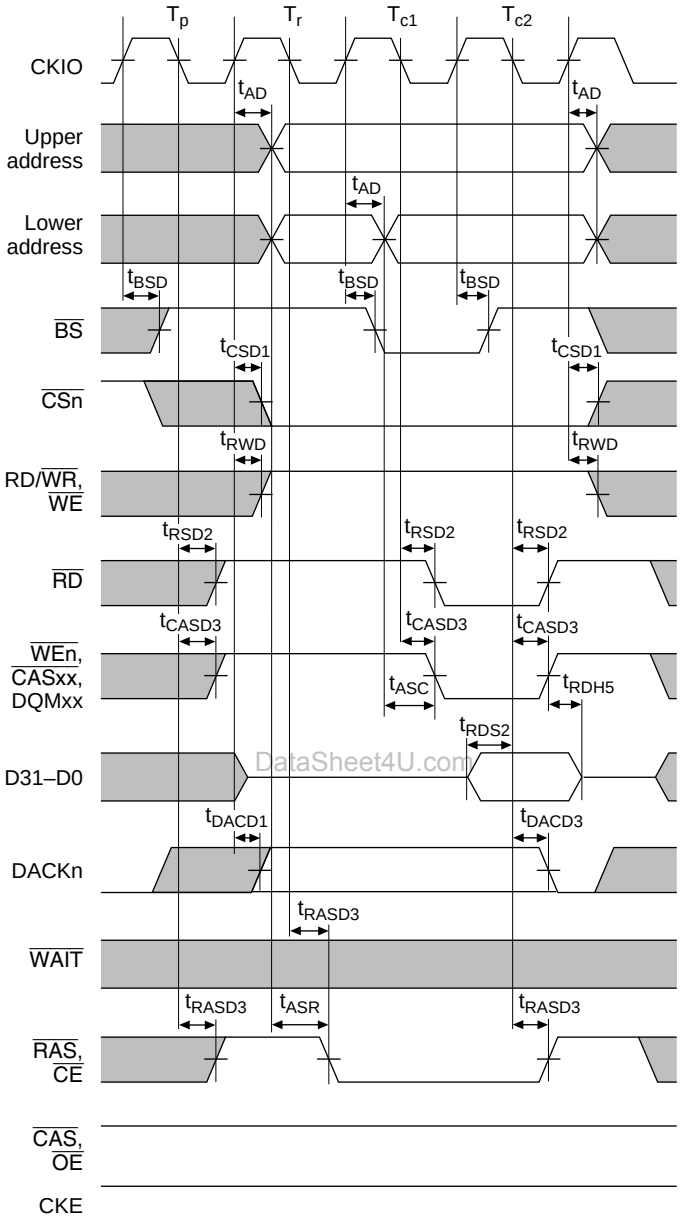
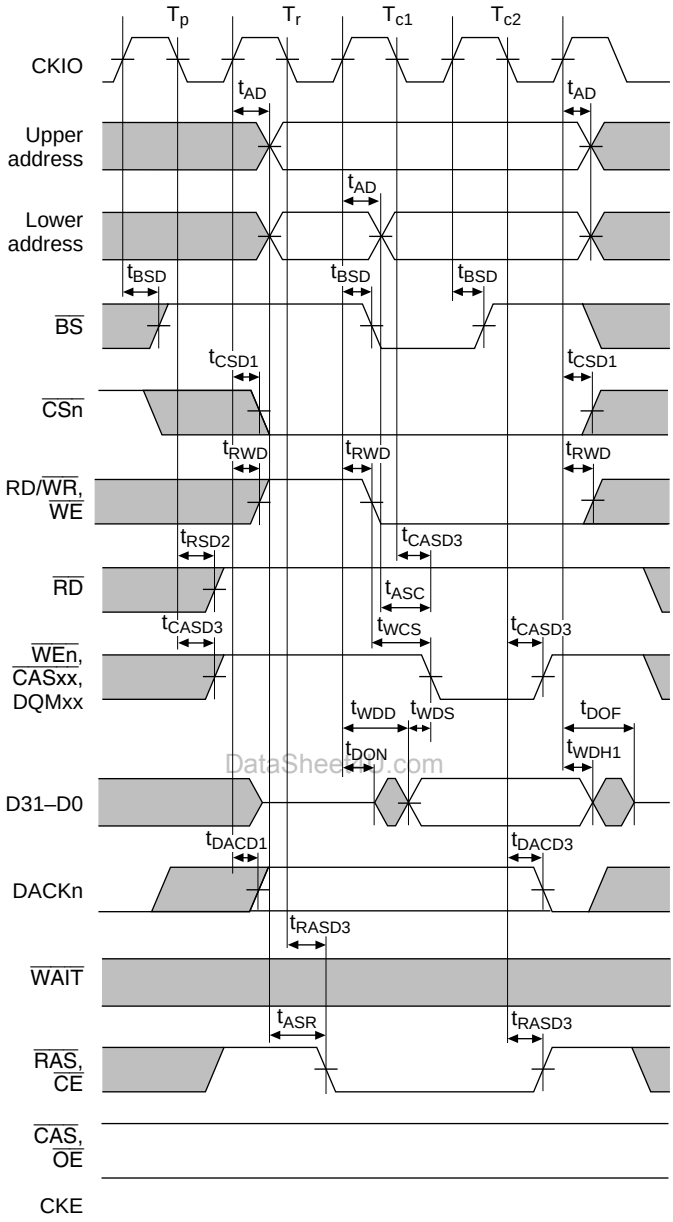


Figure 16.46 DRAM CAS-Before-RAS Refresh Cycle (TRP = 1 Cycle, TRAS = 2 Cycles, PLL On)



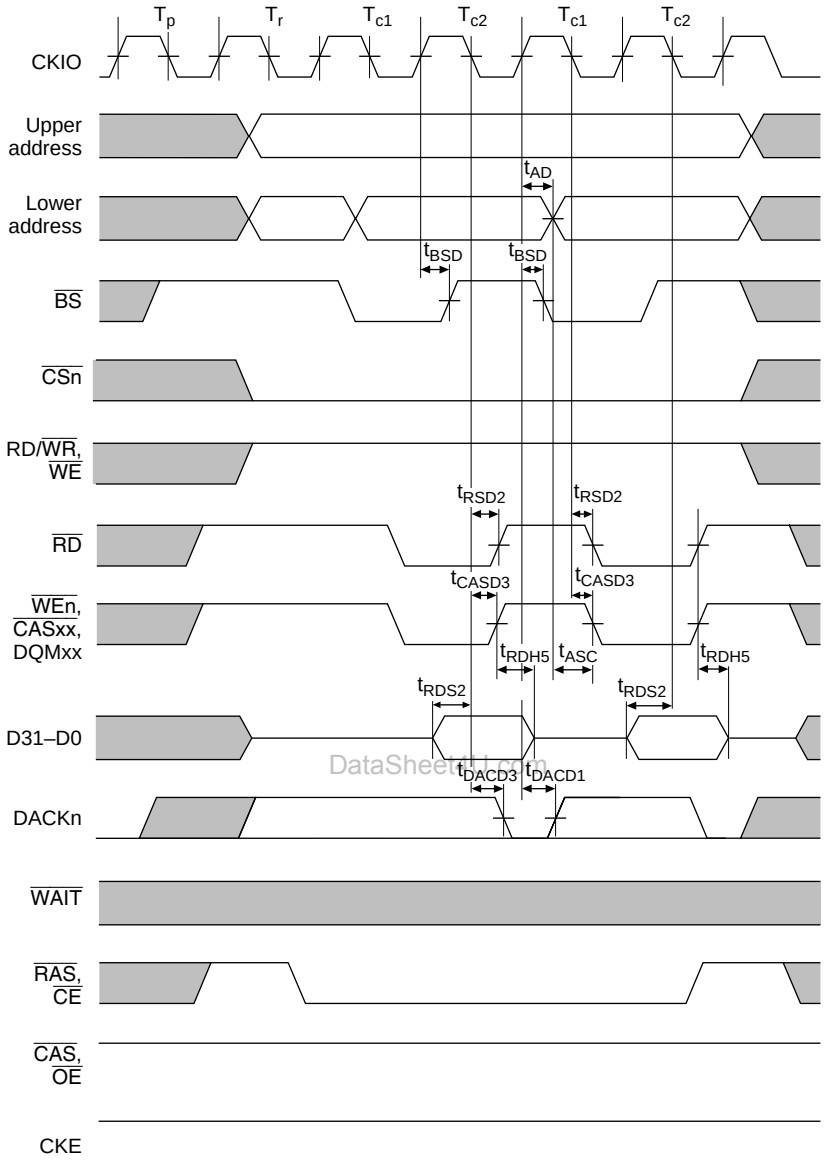
Notes: 1. t_{RDH5} is specified from the rise of $\overline{RD}$ or $\overline{CASxx}$, whichever is first.
2. The DACKn waveform shown is for the case where active-high has been specified.

Figure 16.47 DRAM Read Cycle (TRP = 1 Cycle, RCD = 1 Cycle, No Waits, PLL Off)



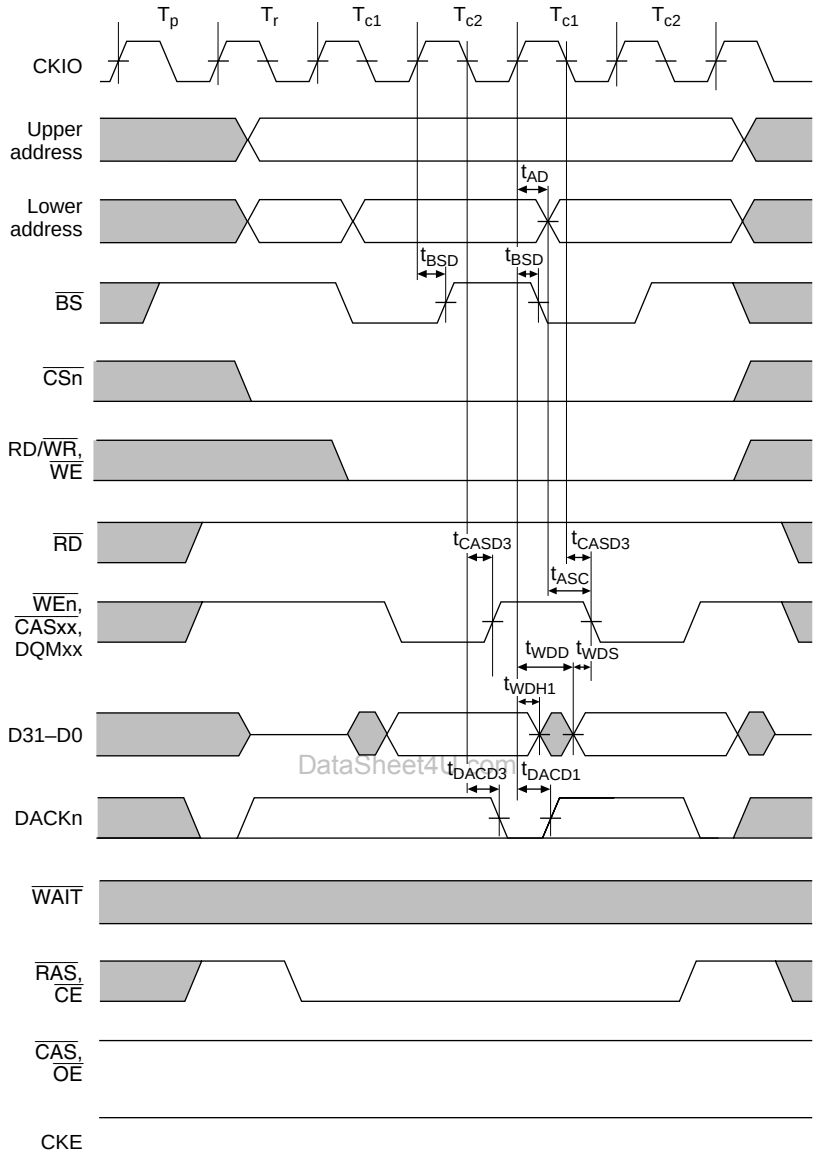
Note: The DACKn waveform shown is for the case where active-high has been specified.

Figure 16.48 DRAM Write Cycle (TRP = 1 Cycle, RCD = 1 Cycle, No Waits, PLL Off)



Notes: 1. t_{RDH5} is specified from the rise of $\overline{RD}$ or $\overline{CASxx}$, whichever is first.
2. The DACKn waveform shown is for the case where active-high has been specified.

Figure 16.49 DRAM Burst Read Cycle
(TRP = 1 Cycle, RCD = 1 Cycle, No Waits, PLL Off)



Note: The DACKn waveform shown is for the case where active-high has been specified.

Figure 16.50 DRAM Burst Write Cycle (TRP = 1 Cycle, RCD = 1 Cycle, No Waits, PLL Off)

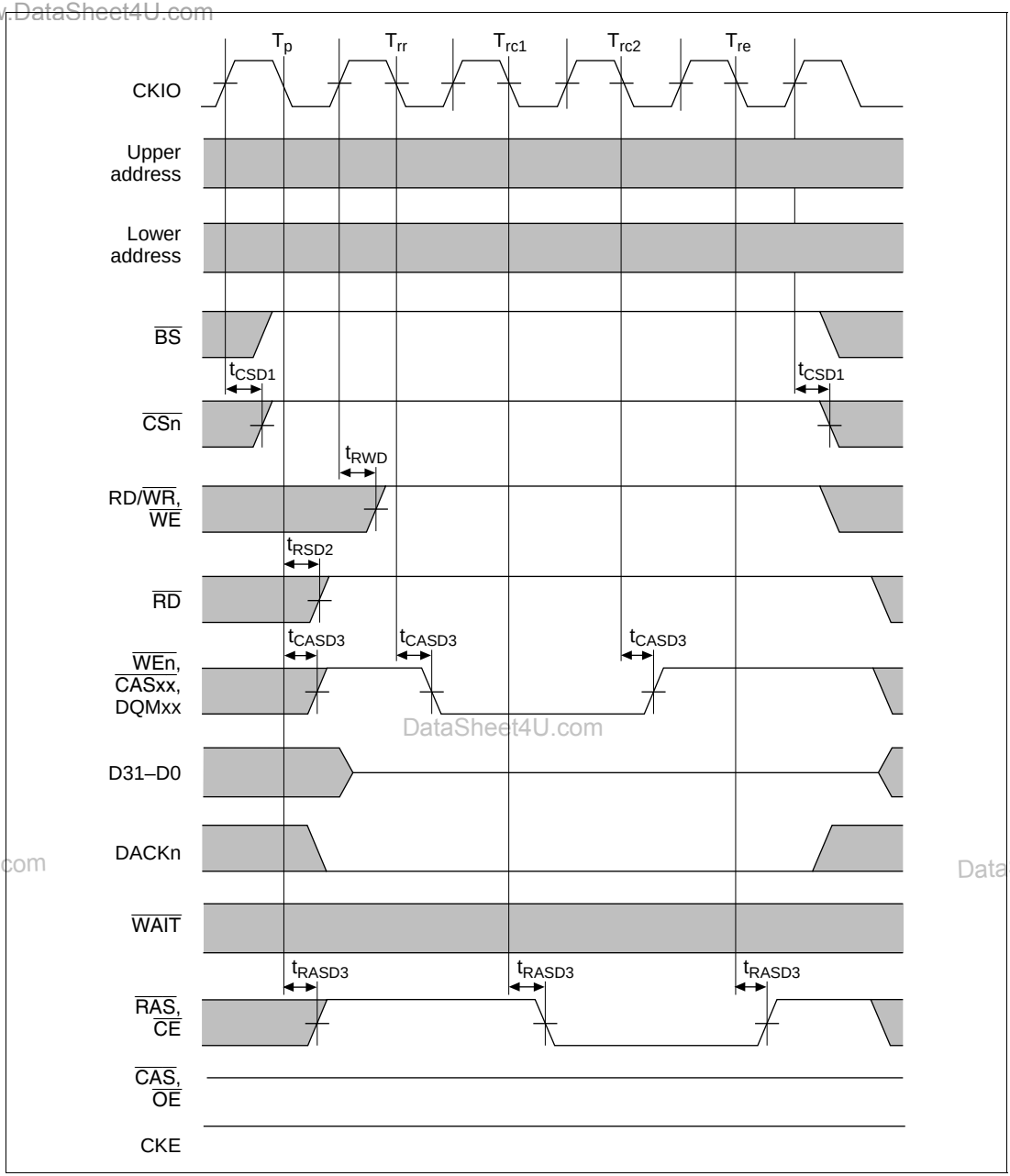
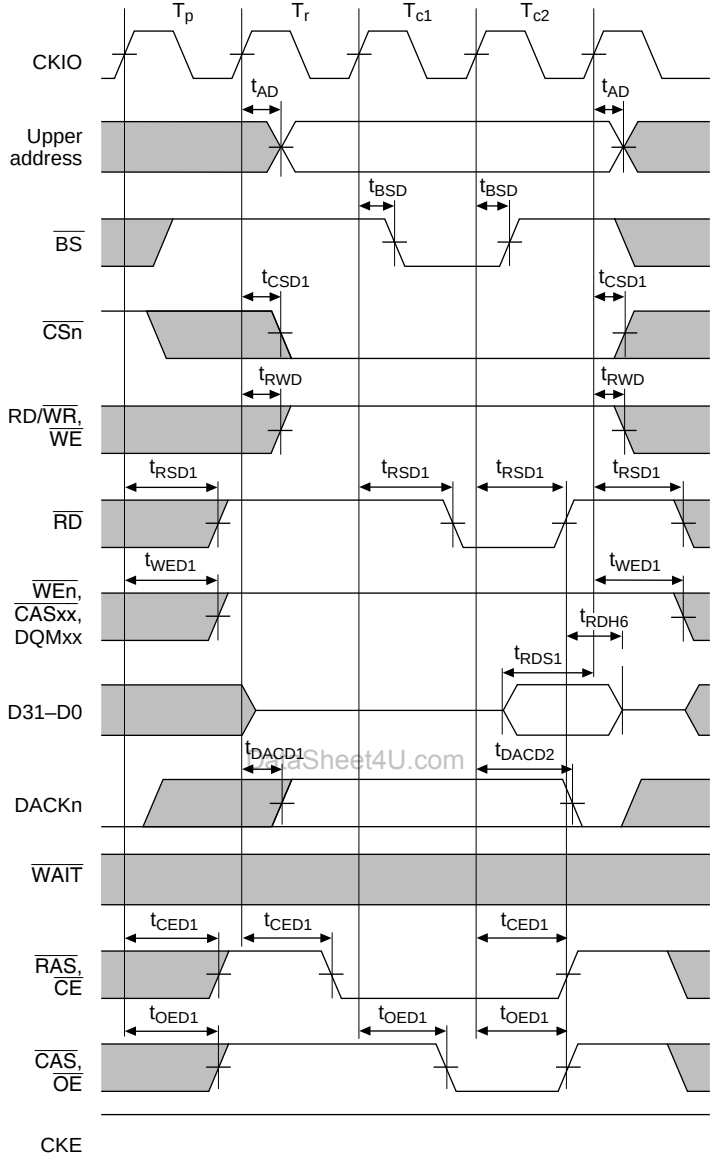
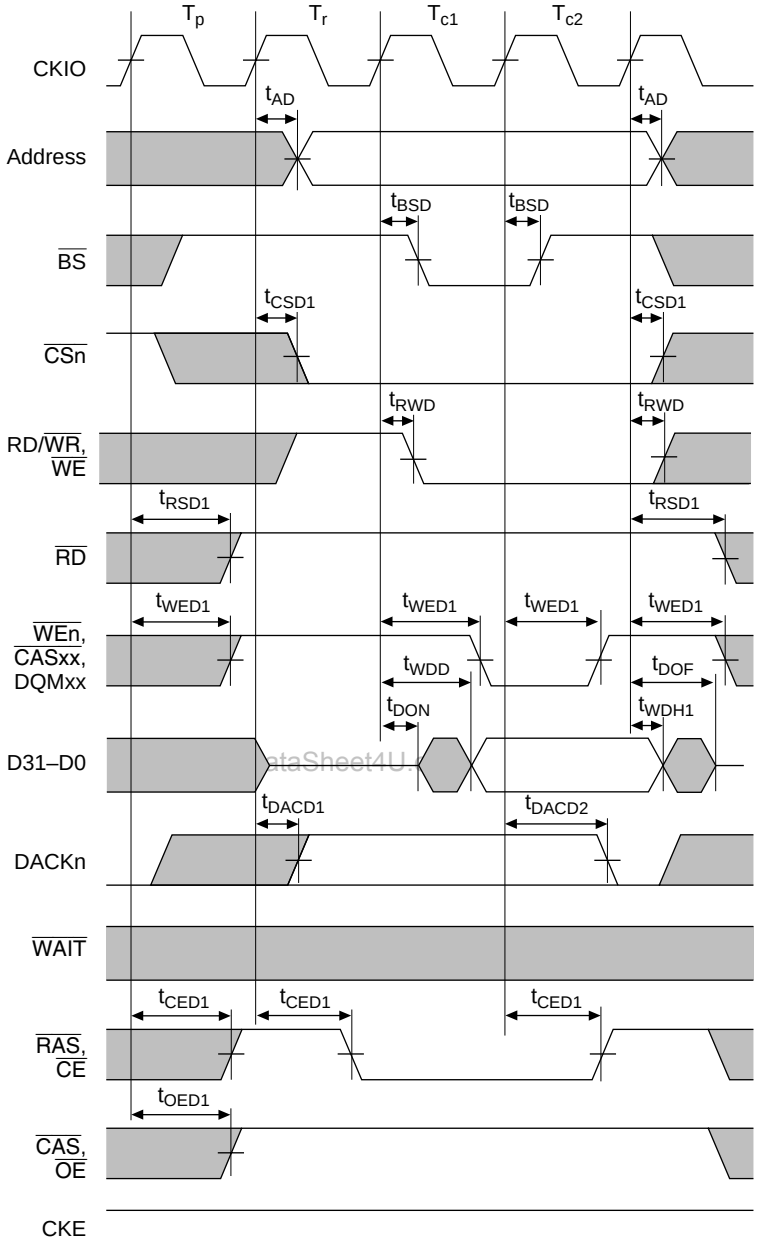


Figure 16.51 DRAM CAS-Before-RAS Refresh Cycle (TRP = 1 Cycle, TRAS = 2 Cycles, PLL Off)



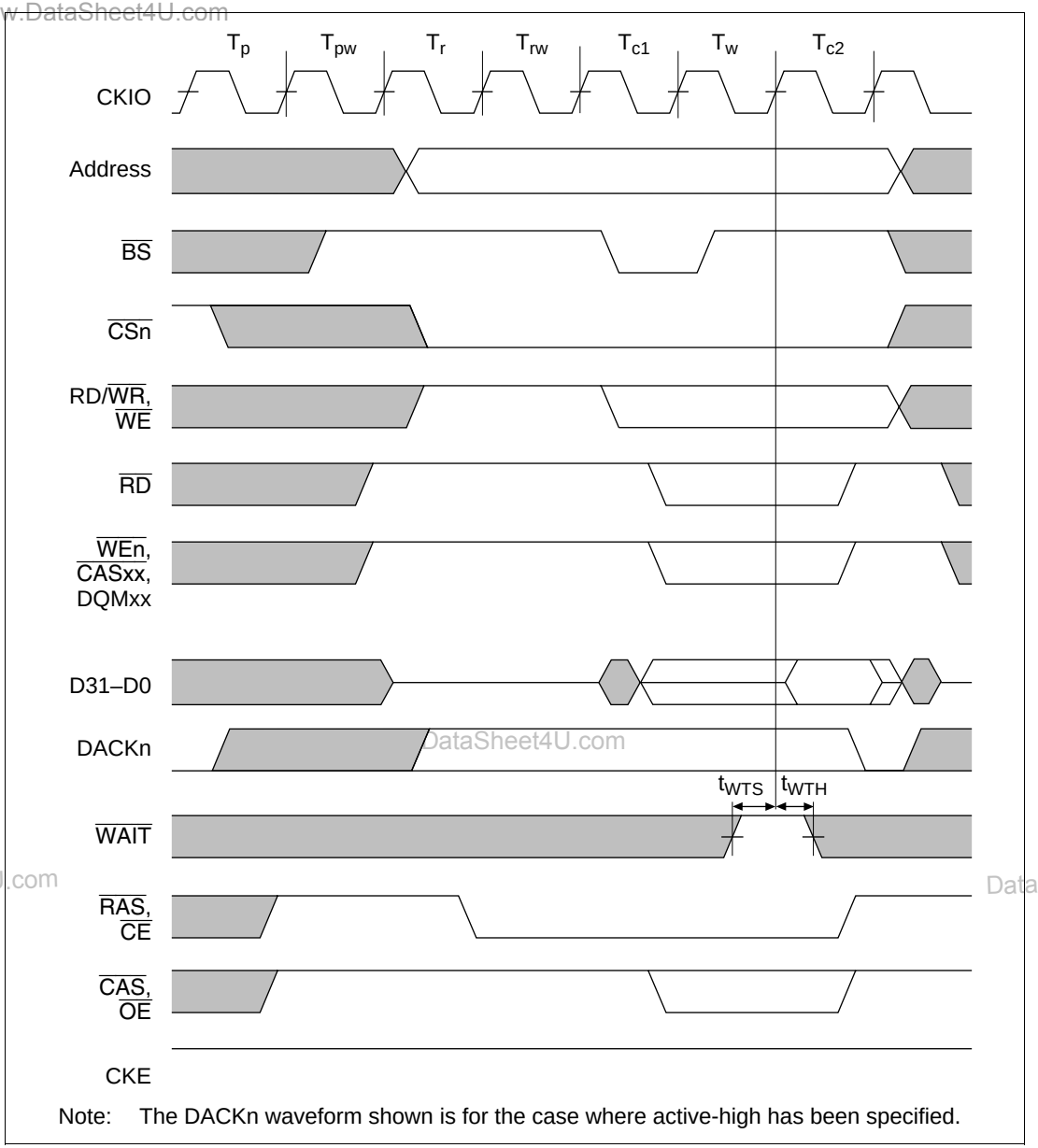
Note: The DACKn waveform shown is for the case where active-high has been specified.

Figure 16.52 Pseudo-SRAM Read Cycle (PLL On, TRP = 1 Cycle, RCD = 1 Cycle, No Waits)



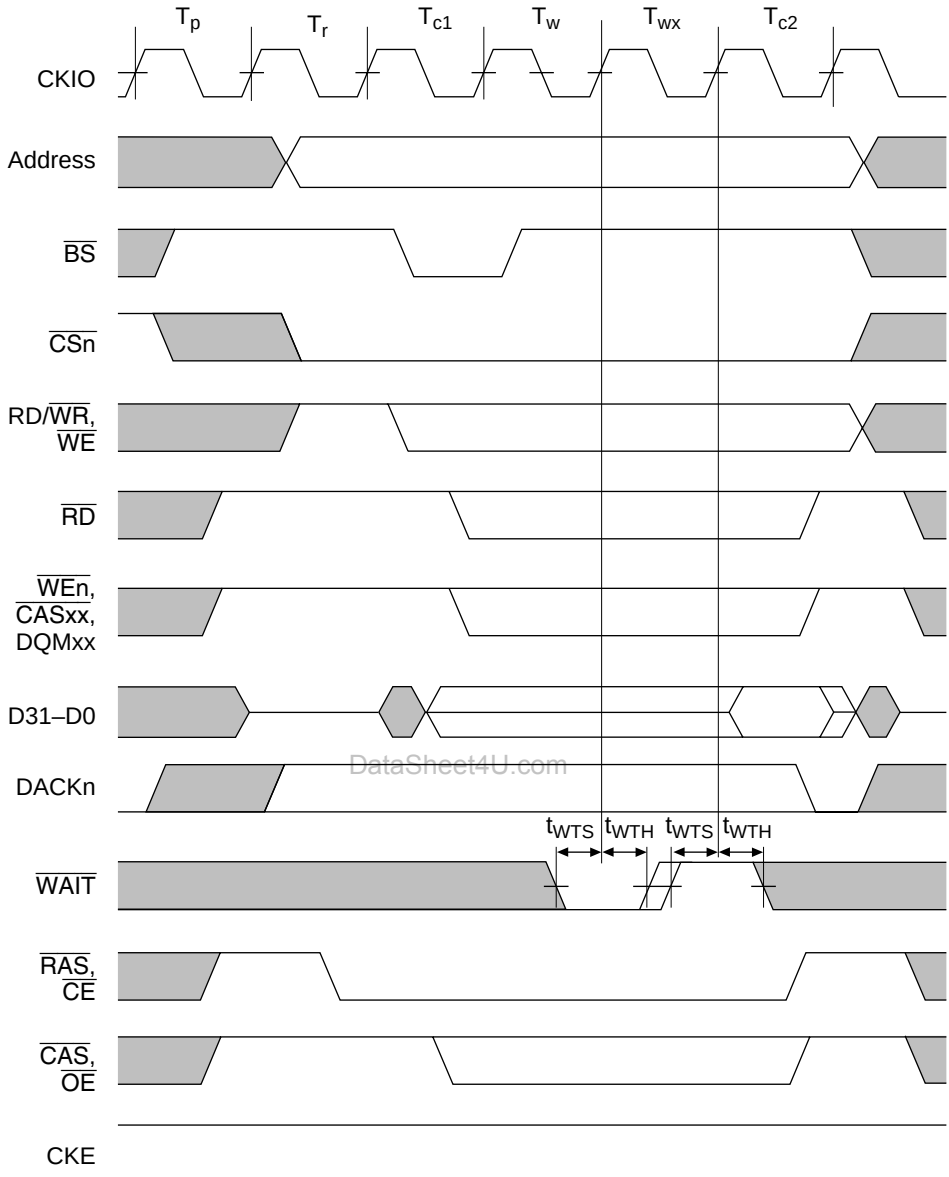
Note: The DACKn waveform shown is for the case where active-high has been specified.

Figure 16.53 Pseudo-SRAM Write Cycle
(PLL On, TRP = 1 Cycle, RCD = 1 Cycle, No Waits)



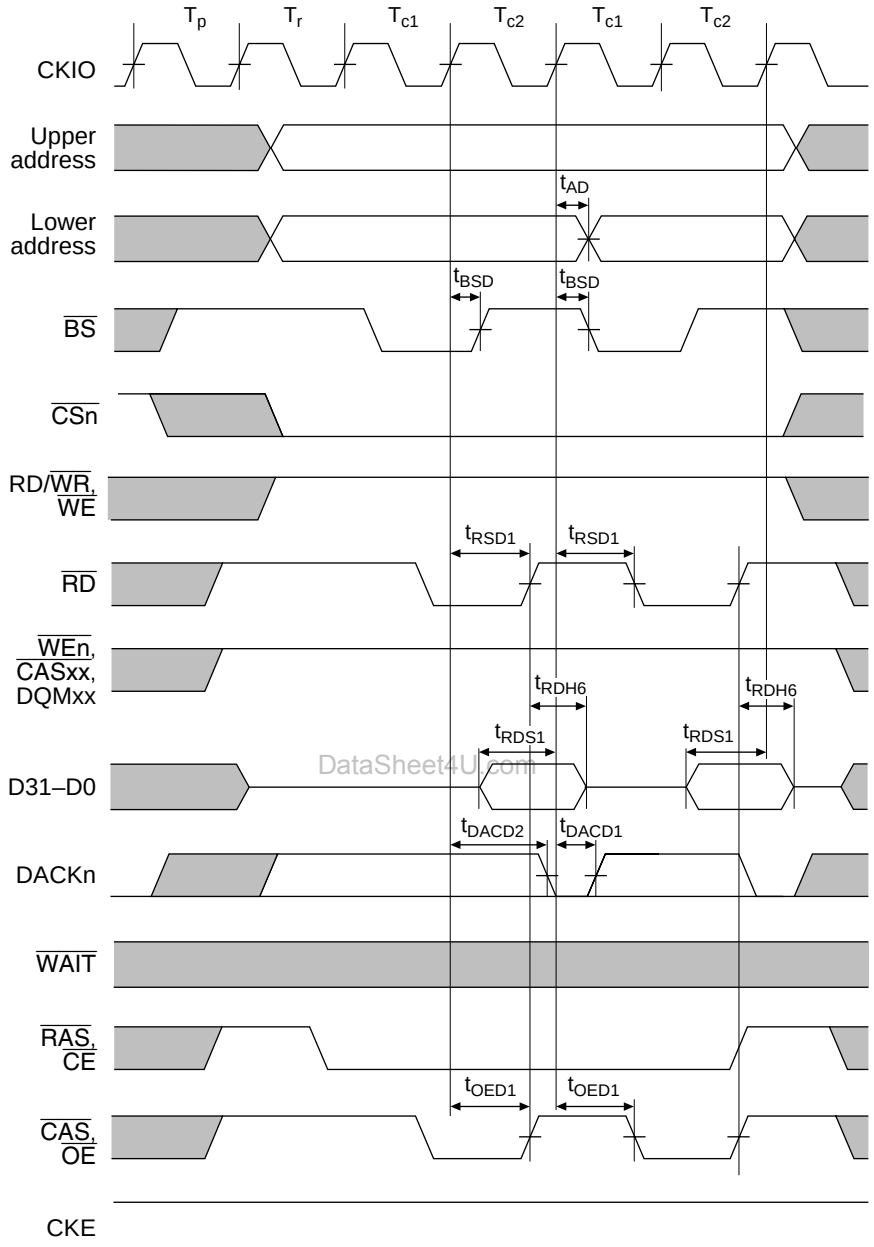
Note: The DACKn waveform shown is for the case where active-high has been specified.

Figure 16.54 Pseudo-SRAM Bus Cycle (TRP = 2 Cycles, RCD = 2 Cycles, 1 Wait)



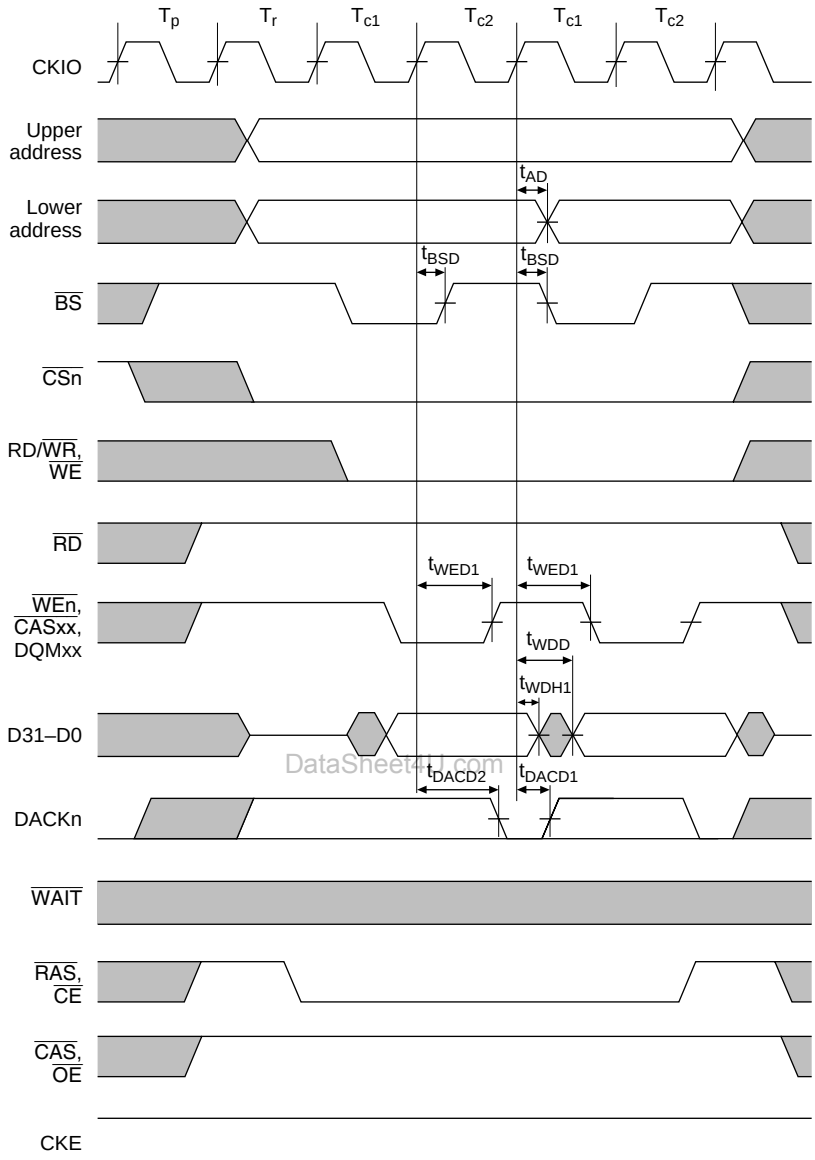
Note: The DACKn waveform shown is for the case where active-high has been specified.

Figure 16.55 Pseudo-SRAM Bus Cycle
(TRP = 1 Cycle, RCD = 1 Cycle, External Wait Input)



Note: The DACKn waveform shown is for the case where active-high has been specified.

Figure 16.56 Pseudo-SRAM Read Cycle
(Static Column Mode, PLL On, TRP = 1 Cycle, RCD = 1 Cycle, No Waits)



Note: The DACKn waveform shown is for the case where active-high has been specified.

Figure 16.57 Pseudo-SRAM Write Cycle (Static Column Mode, PLL On, TRP = 1 Cycle, RCD = 1 Cycle, No Waits)

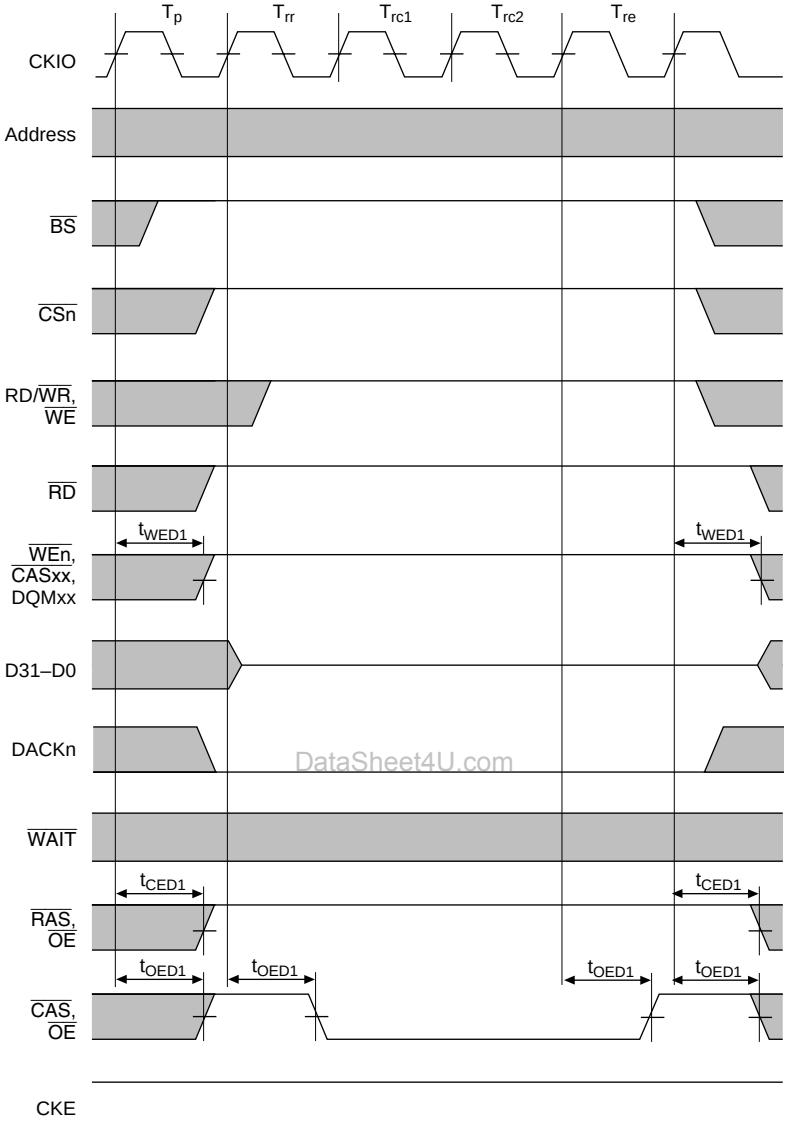


Figure 16.58 Pseudo-SRAM Auto-Refresh Cycle (PLL On, TRP = 1 Cycle, TRAS = 2 Cycles)

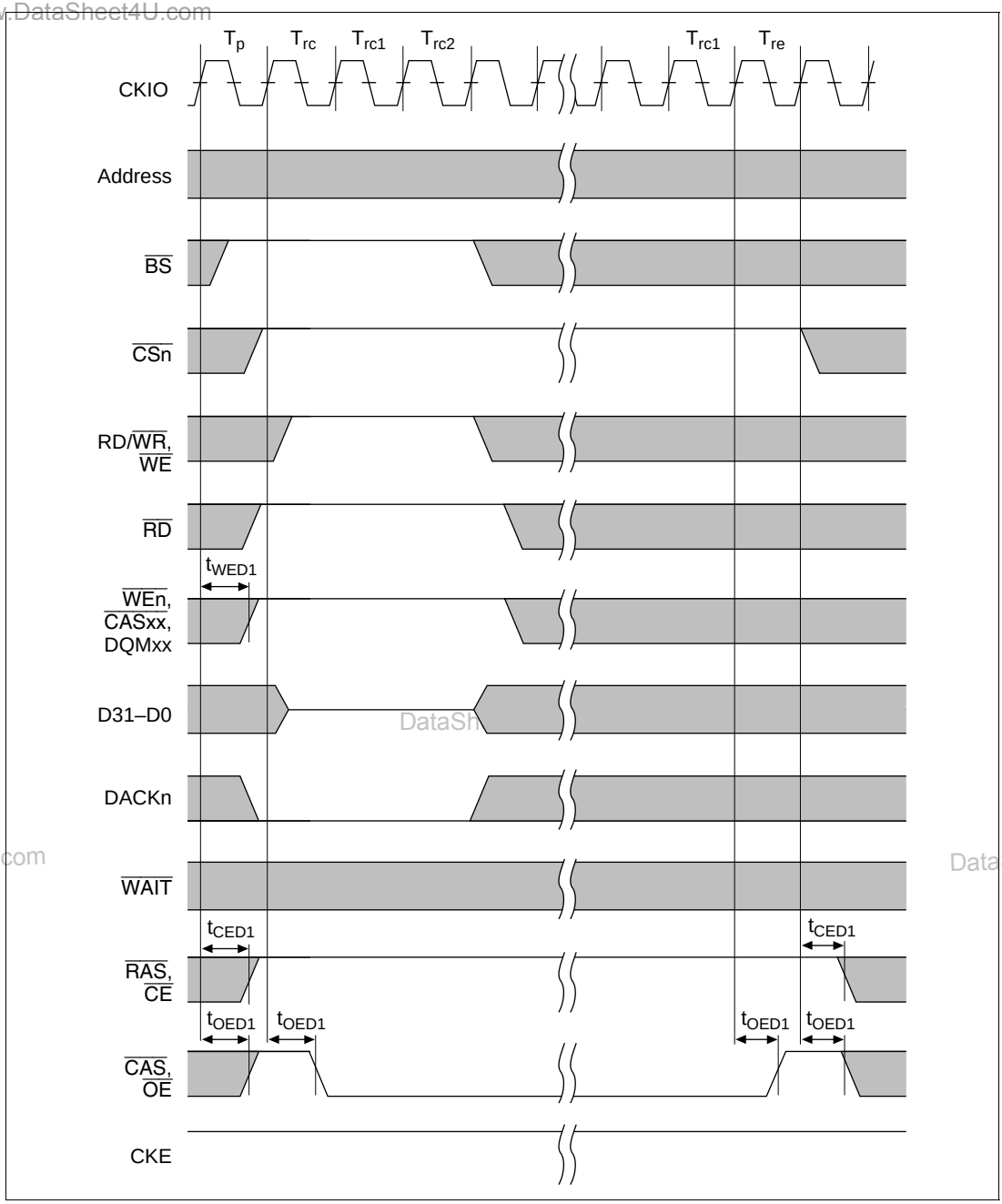
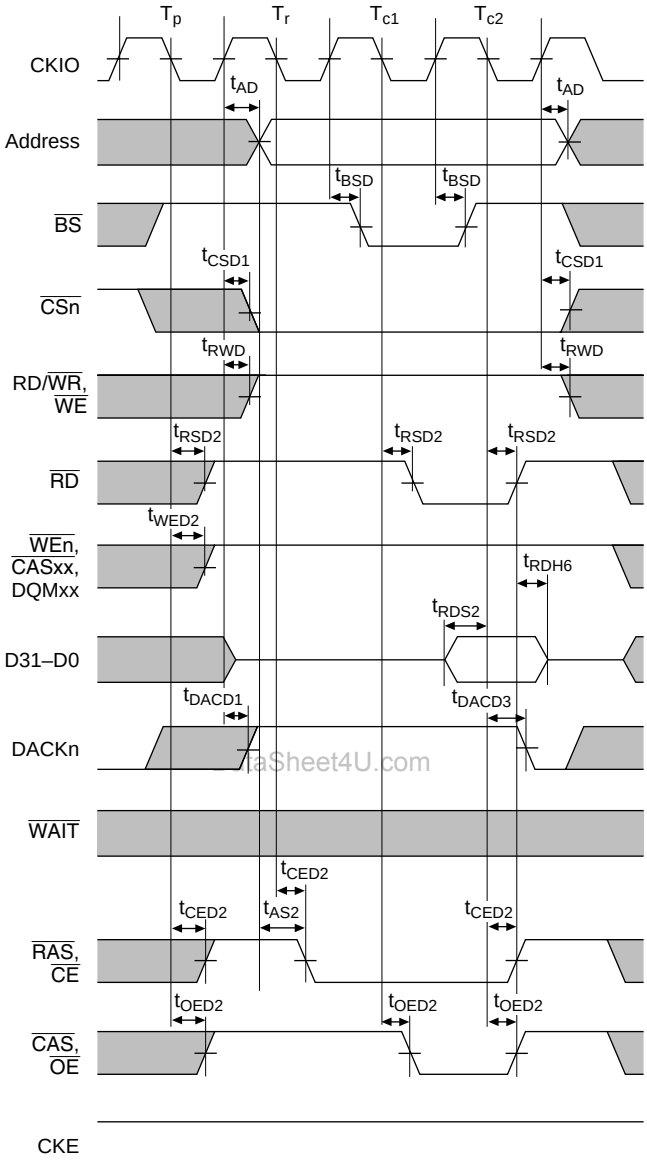
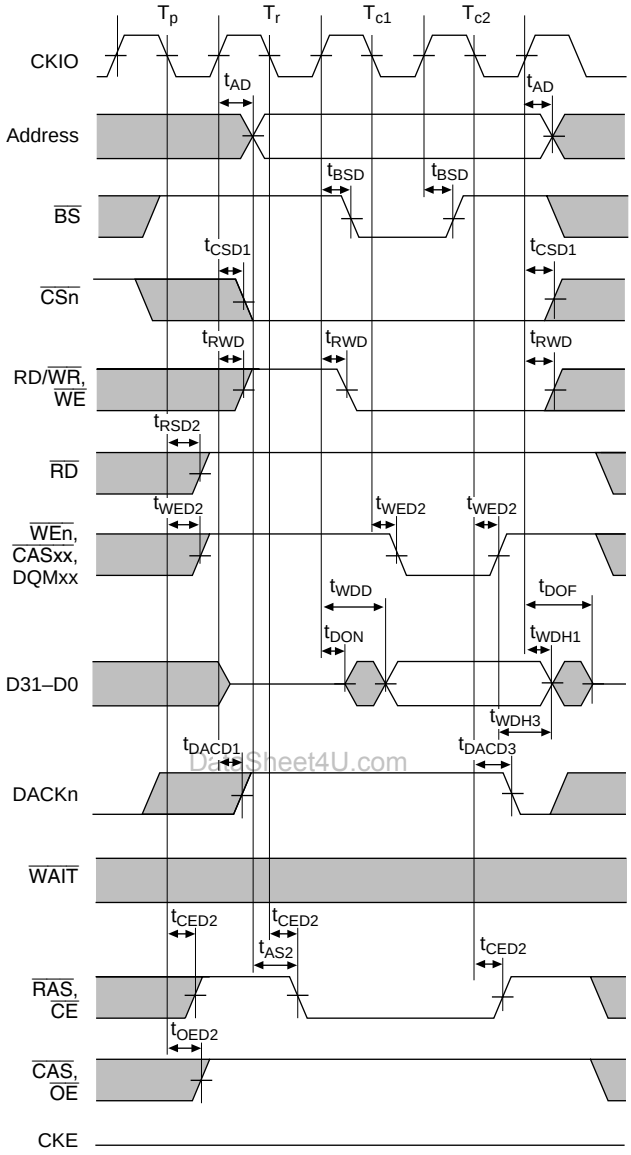


Figure 16.59 Pseudo-SRAM Self-Refresh Cycle (PLL On, TRP = 1 Cycle, TRAS = 2 Cycles)



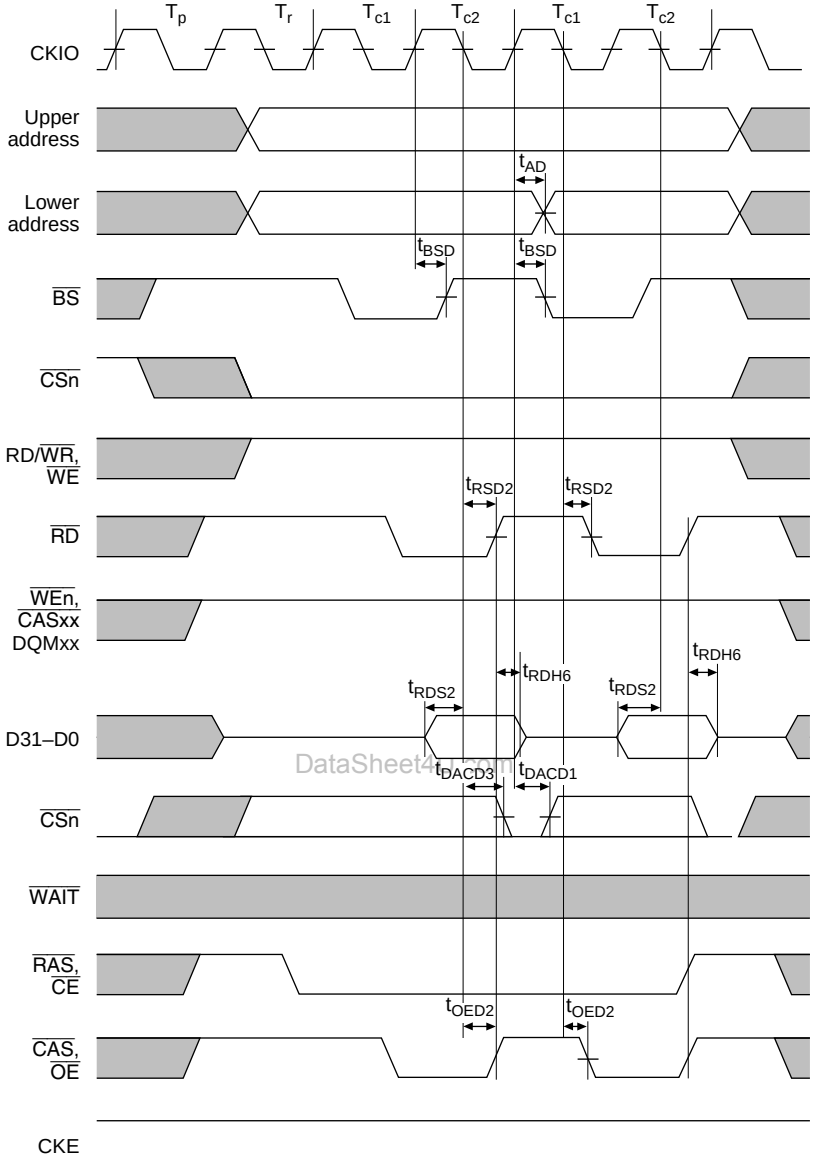
Note: The DACKn waveform shown is for the case where active-high has been specified.

Figure 16.60 Pseudo-SRAM Read Cycle
(PLL Off, TRP = 1 Cycle, RCD = 1 Cycle, No Waits)



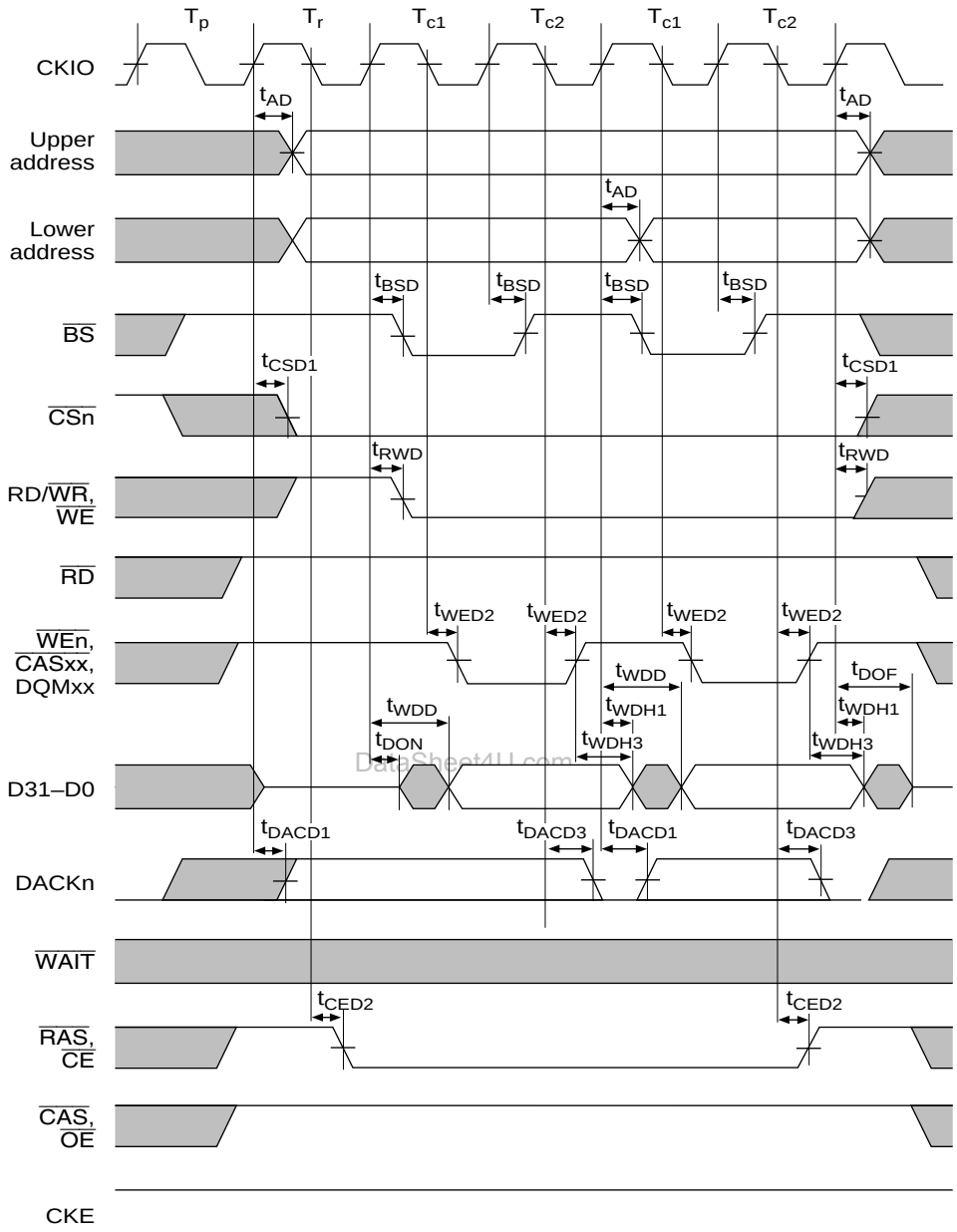
Note: The DACKn waveform shown is for the case where active-high has been specified.

Figure 16.61 Pseudo-SRAM Write Cycle
(PLL Off, TRP = 1 Cycle, RCD = 1 Cycle, No Waits)



Note: The DACKn waveform shown is for the case where active-high has been specified.

Figure 16.62 Pseudo-SRAM Read Cycle
(Static Column Mode, PLL Off, TRP = 1 Cycle, RCD = 1 Cycle, No Waits)



Note: The DACKn waveform shown is for the case where active-high has been specified.

Figure 16.63 Pseudo-SRAM Write Cycle
(Static Column Mode, PLL Off, TRP = 1 Cycle, RCD = 1 Cycle, No Waits)

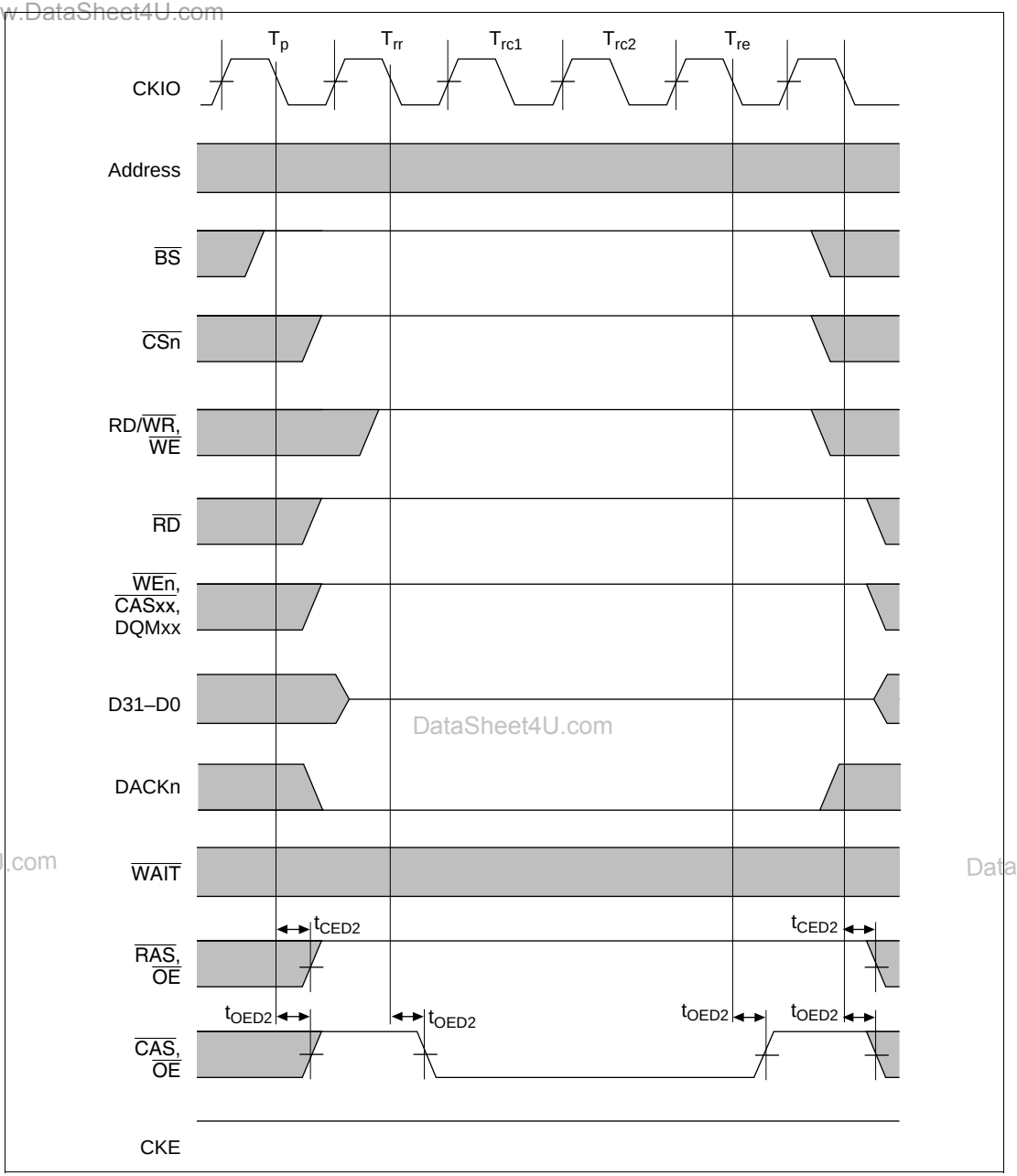


Figure 16.64 Pseudo-SRAM Auto-Refresh Cycle (PLL Off, TRP = 1 Cycle, TRAS = 2 Cycles)

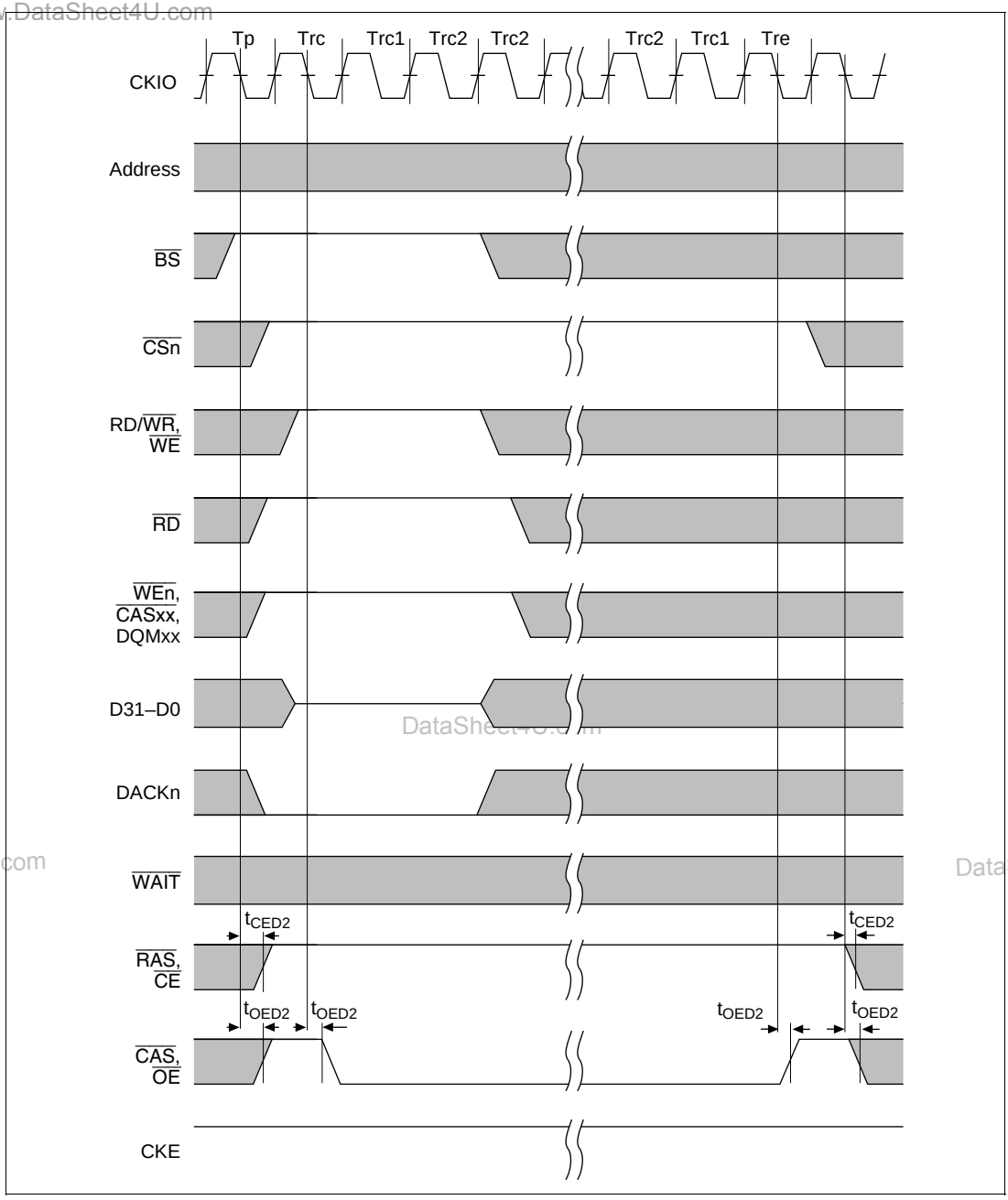
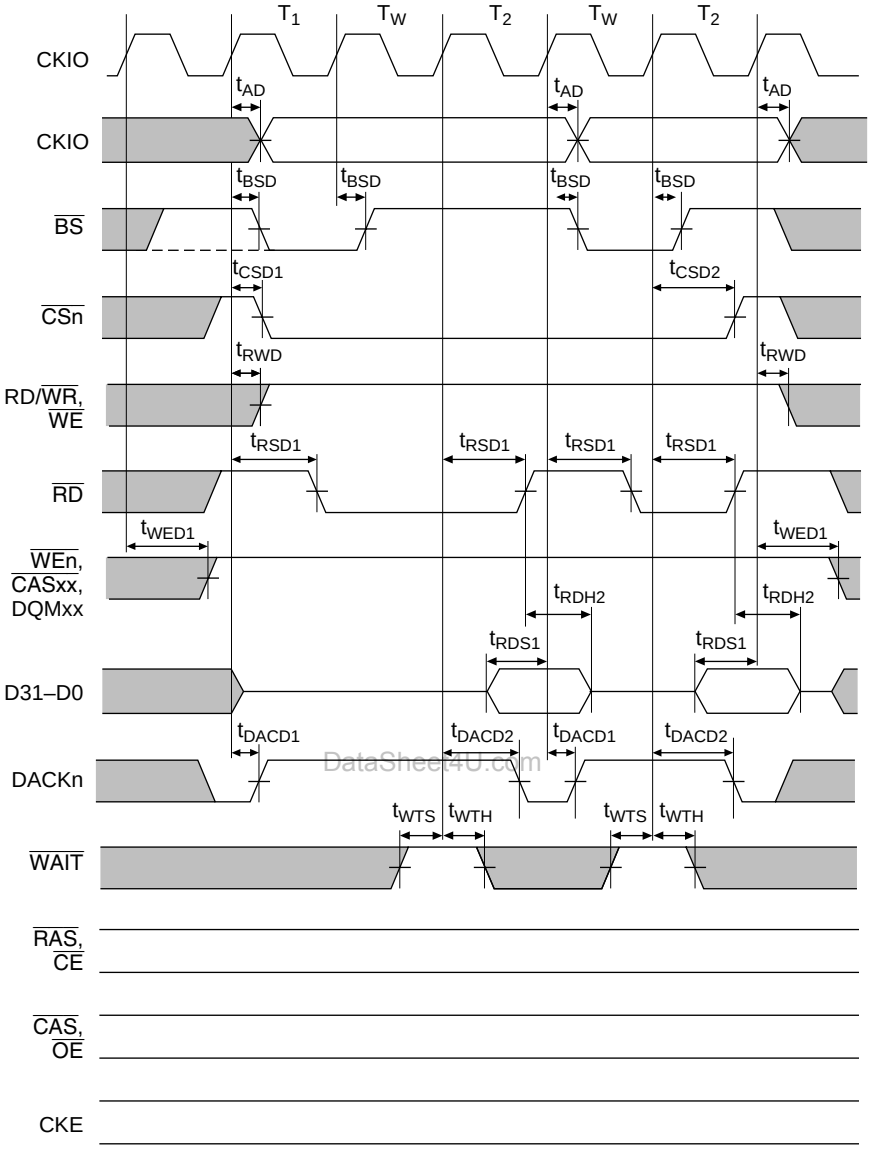
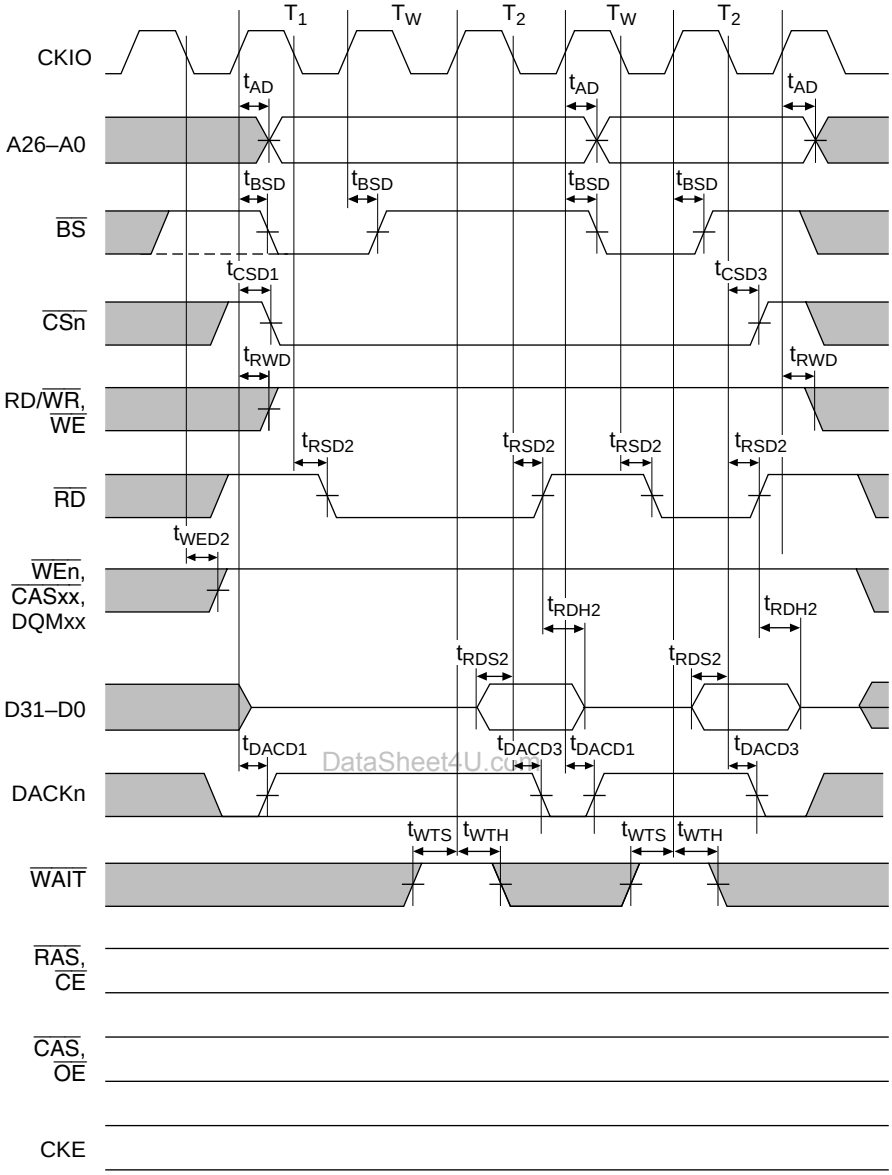


Figure 16.65 Pseudo-SRAM Self-Refresh Cycle (PLL Off, TRP = 1 Cycle, TRAS = 2 Cycles)



Note: The DACKn waveform shown is for the case where active-high has been specified.

Figure 16.66 Burst ROM Read Cycle (PLL On, 1 Wait)



Note: The DACKn waveform shown is for the case where active-high has been specified.

Figure 16.67 Burst ROM Read Cycle (PLL Off, 1 Wait)

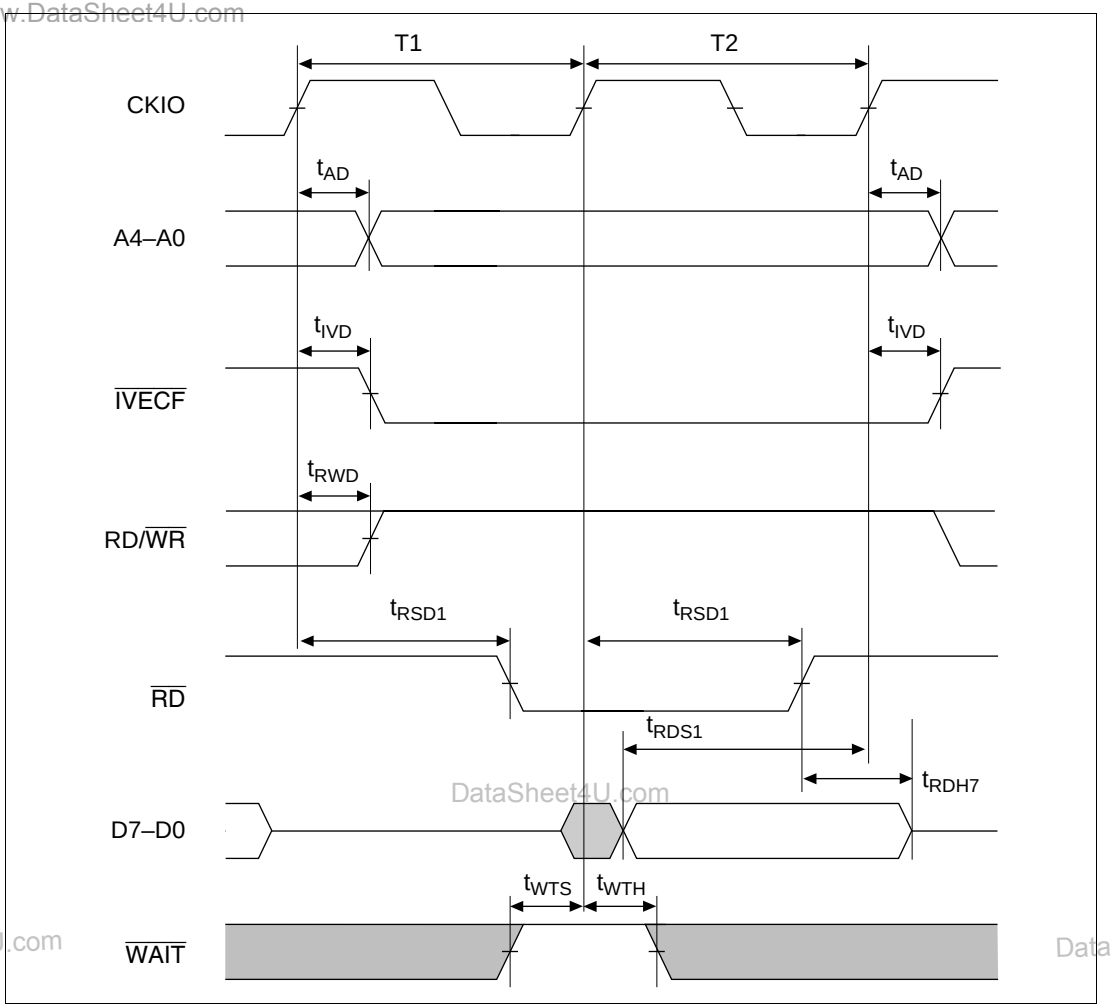


Figure 16.68 Interrupt Vector Fetch Cycle (PLL On, No Waits)

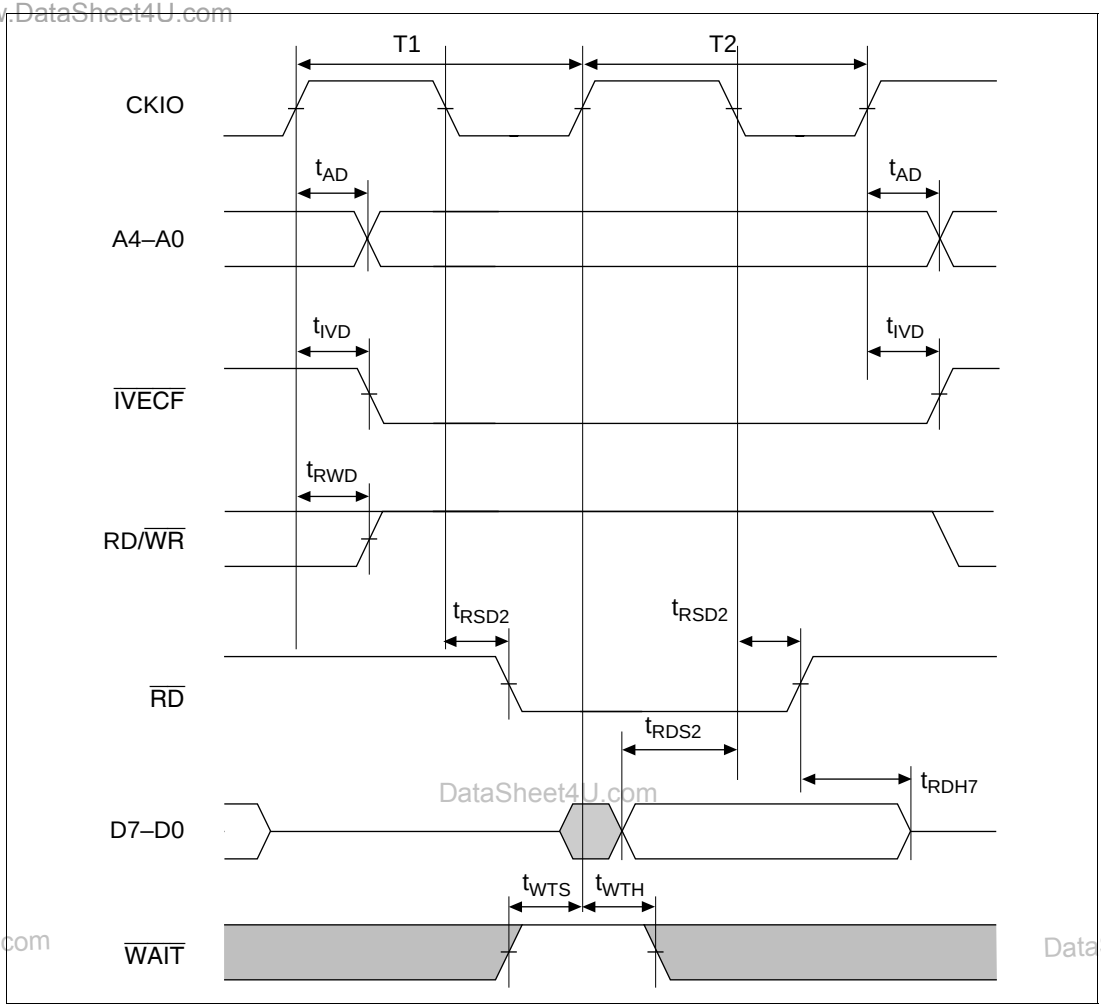


Figure 16.69 Interrupt Vector Fetch Cycle (PLL Off, No Waits)

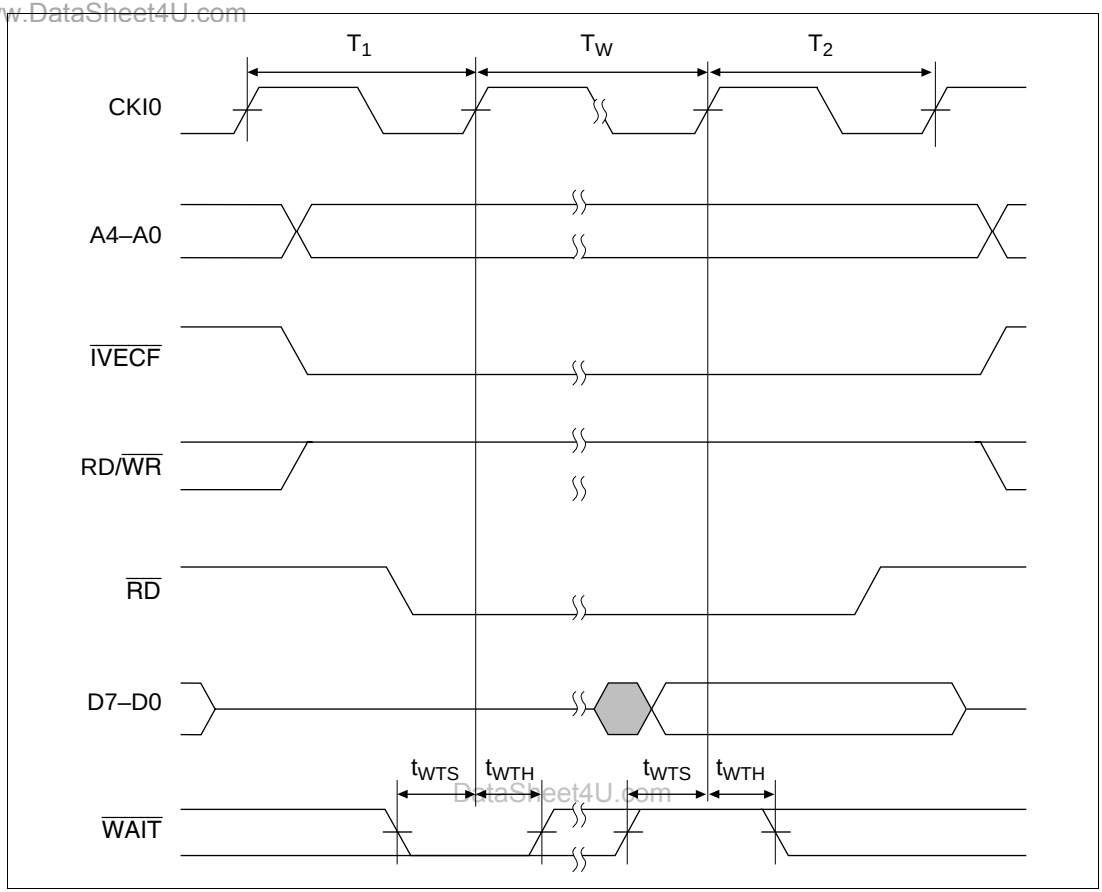


Figure 16.70 Interrupt Vector Fetch Cycle (1 External Wait Cycle)

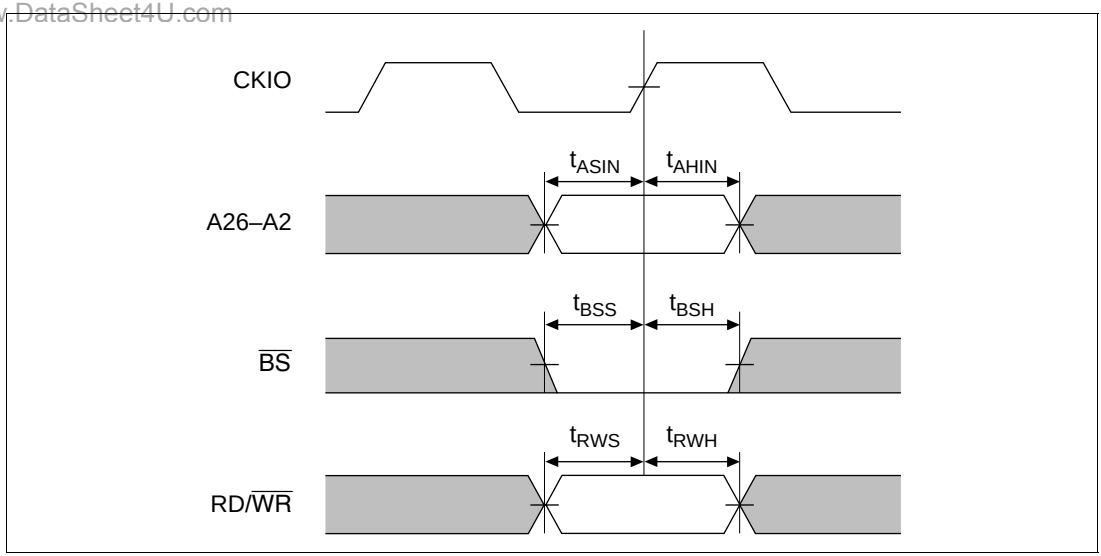


Figure 16.71 Address Monitor Cycle

16.3.4 DMAC Timing

Table 16.10 DMAC Timing (Conditions: V_{CC} = 3.0 to 5.5 V, Ta = –20 to +75°C)

Item	Symbol	Min	Max	Unit	Figure
DREQ0, DREQ1 setup time (PLL Off, On)	t_{DRQS}	50	—	ns	16.72
DREQ0, DREQ1 setup time (PLL On, 1/4 cycle delay)	t_{DRQS}	50 – 1/4 tcyc	—	ns	
DREQ0, DREQ1 hold time (PLL Off, On)	t_{DRQH}	50	—	ns	
DREQ0, DREQ1 hold time (PLL On, 1/4 cycle delay)	t_{DRQH}	1/4 tcyc + 50	—	ns	
DREQ0, DREQ1 low level width	t_{DRQW}	1.5	—	t_{cyc}	

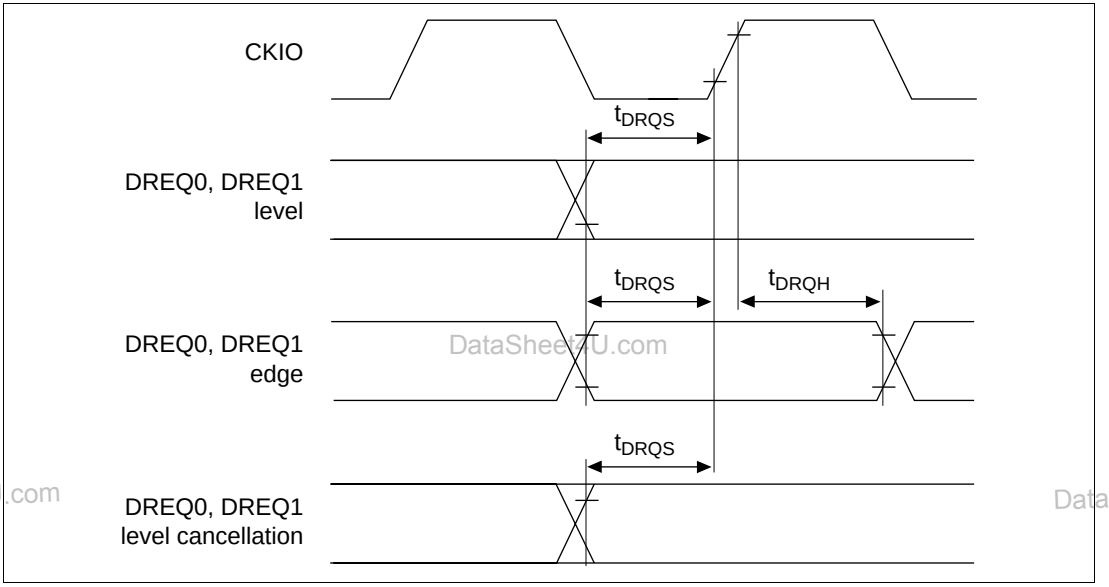


Figure 16.72 DREQ0, DREQ1 Input Timing

16.3.5 Free-Running Timer Timing

Table 16.11 Free-Running Timer Timing (Conditions: V_{CC} = 3.0 to 5.5 V, Ta = –20 to +75°C)

Item	Symbol	Min	Max	Unit	Figure
Output compare output delay time (PLL Off, On)	t_{TOCD}	—	320	ns	16.73
Output compare output delay time (PLL On, 1/4 cycle delay)	t_{TOCD}	—	1/4 tcyc + 320	ns	
Input capture input setup time (PLL Off, On)	t_{TICS}	80	—	ns	
Input capture input setup time (PLL On, 1/4 cycle delay)	t_{TICS}	80 – 1/4 tcyc	—	ns	
Timer clock input setup time (PLL Off, On)	t_{TCKS}	80	—	ns	16.74
Timer clock input setup time (PLL On, 1/4 cycle delay)	t_{TCKS}	80 –1/4 tcyc	—	ns	
Timer clock pulse width (single edge)	t_{TCKWH}	4.5	—	t_{cyc}	
Timer clock pulse width (both edges)	t_{TCKWL}	8.5	—	t_{cyc}	

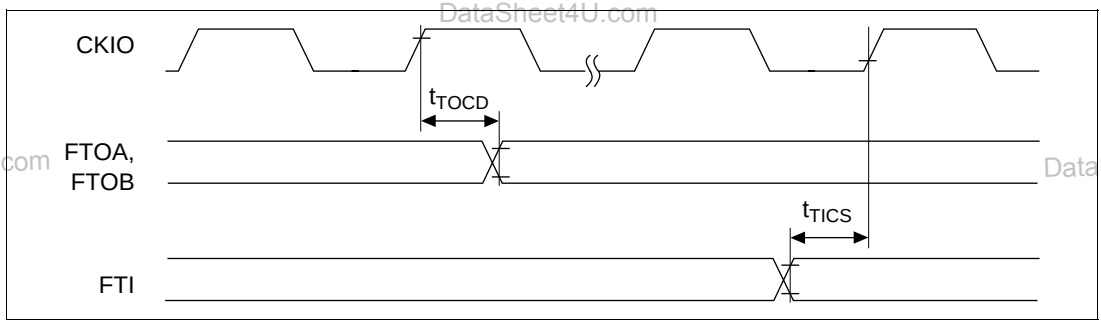


Figure 16.73 FRT Input/Output Timing

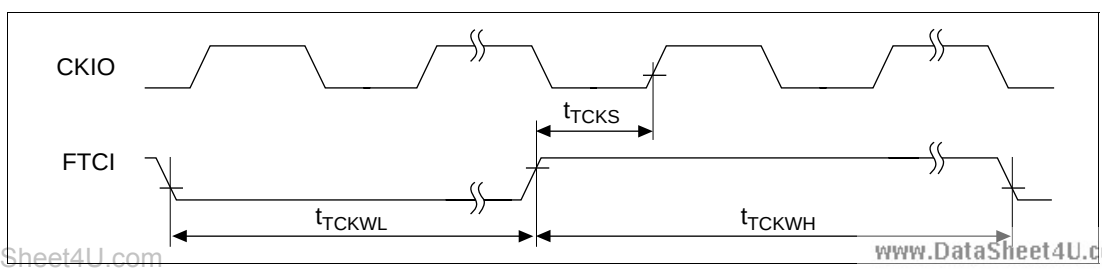


Figure 16.74 FRT Clock Input Timing

16.3.6 Watchdog Timer Timing

Table 16.12 Watchdog Timer Timing (Conditions: V_{CC} = 3.0 to 5.5 V, Ta = –20 to +75°C)

Item	Symbol	Min	Max	Unit	Figure
$\overline{\text{WDTOVF}}$ delay time (PLL Off, On)	t_{WOVD}	—	70	ns	16.75
$\overline{\text{WDTOVF}}$ delay time (PLL On, 1/4 cycle delay)	t_{WOVD}	—	1/4 tcyc + 70	ns	

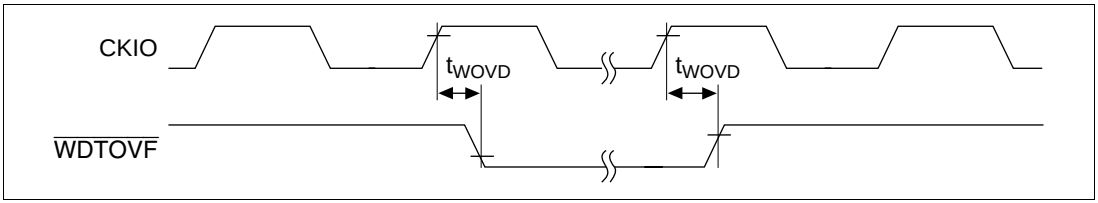


Figure 16.75 Watchdog Timer Output Timing

16.3.7 Serial Communication Interface Timing

Table 16.13 Serial Communication Interface Timing
(Conditions: V_{CC} = 3.0 to 5.5 V, Ta = –20 to +75°C)

Item	Symbol	Min	Max	Unit	Figure
Input clock cycle	t_{scyc}	16	—	t_{cyc}	16.76
Input clock cycle (clocked synchronous mode)	t_{scyc}	24	—	t_{cyc}	
Input clock pulse width	t_{sckw}	0.4	0.6	t_{scyc}	
Transmission data delay time (clocked synchronous mode)	t_{TXD}	—	70	ns	16.77
Receive data setup time (clocked synchronous mode)	t_{RXS}	70	—	ns	
Receive data hold time (clocked synchronous mode)	t_{RXH}	70	—	ns	

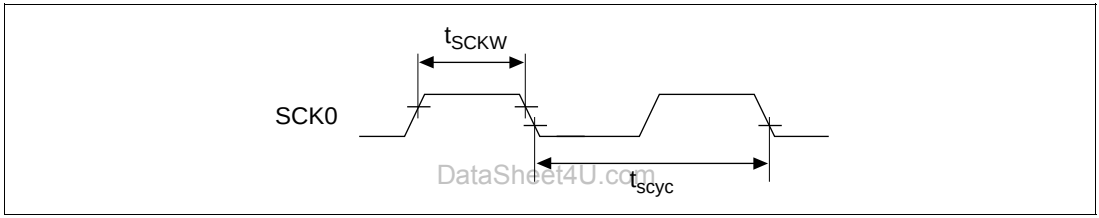


Figure 16.76 Input Clock Input/Output Timing

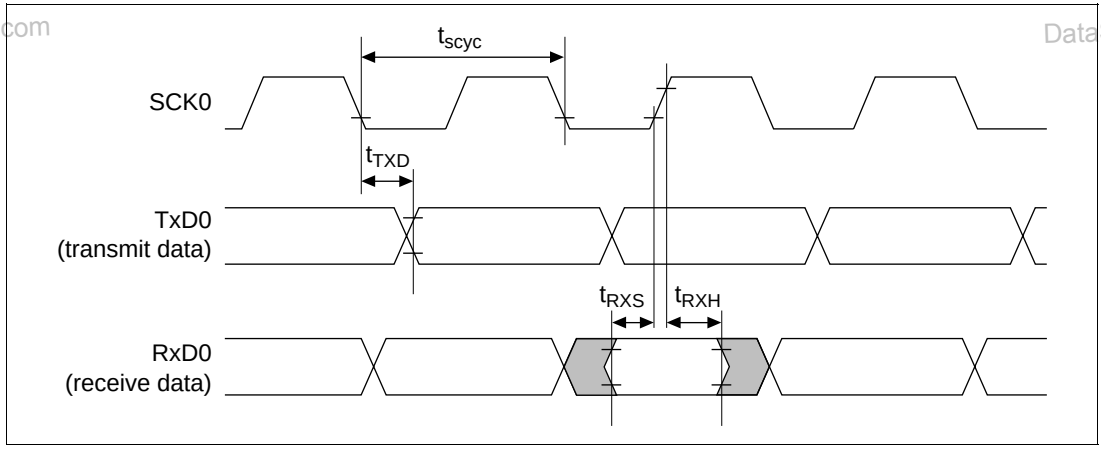
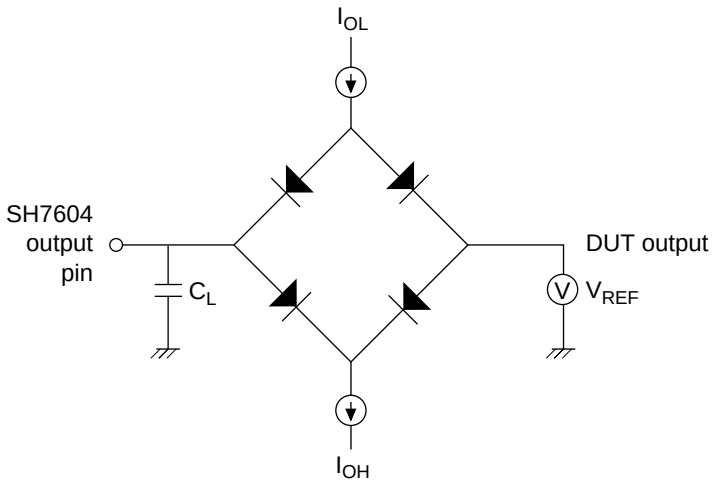


Figure 16.77 SCI Input/Output Timing (Clocked Synchronous Mode)

16.3.8 AC Characteristics Measurement Conditions

- I/O signal reference level: 1.5 V
- Input pulse level: V_{SS} to 3.0 V (where RES, NMI, CKIO and MD5-MD0 are within the range V_{SS} to V_{CC})
- Input rise and fall times: 1 ns



Notes: 1. C_L is a total value that includes the capacitance of measurement instruments, etc., and is set as follows for each pin.
30 pF: CKIO, $\overline{RAS}$, $\overline{CAS}$, CKE, $\overline{CS0}$–$\overline{CS3}$, $\overline{BREQ}$, $\overline{BACK}$, DACK0, DACK1, $\overline{IVECF}$, $\overline{CKPACK}$.
50 pF: All output pins other than the above.

2. I_{OL} and I_{OH} values are as shown in section 16.2, DC Characteristics, and table 16.3, Permitted Output Current Values.

Figure 16.78 Output Load Circuit

Appendix A Pin States

Table A.1 Pin States During Resets, Power-Down State, and Bus-Released State

		Pin States						
		Reset Power-On		Reset Manual		Power-Down Modes		Bus-Released Mode
Category	Pin	Master	Slave	Bus Acquired	Bus Released	Standby	Sleep	
Clock	CKIO	IO*1	IO*1	IO*1	IO*1	IO*1	IO*1	IO*1
	EXTAL	I*1	I*1	I*1	I*1	I*1	I*1	I*1
	XTAL	O*1	O*1	O*1	O*1	O*1	O*1	O*1
	$\overline{\text{CKPREQ}}$	Z	Z	I	I	I	I	I
	$\overline{\text{CKPACK}}$	H	H	H	H	H*2	H	H
System control	$\overline{\text{RESET}}$	I	I	I	I	I	I	I
	$\overline{\text{WDTOVF}}$	H	H	H	H	O	O	O
	$\overline{\text{BACK}}$, $\overline{\text{BRLS}}$	Z	Z	I	I	Z	I	I
	$\overline{\text{BREQ}}$, $\overline{\text{BGR}}$	H	H	O	O	H	O	O
	MD5–MD0	I	I	I	I	I	I	I
Interrupt	NMI	I	I	I	I	I	I	I
	$\overline{\text{IRL3}}$–$\overline{\text{IRL0}}$	Z	Z	Z	Z	I	I	I
	$\overline{\text{IVECF}}$	H	H	H	H	H*3	H	H
Address bus	A26–A0	O	Z	O	Z	Z	O	Z*4
Data bus	D31–D0	Z	Z	IO	Z	Z	Z	Z
Bus control	$\overline{\text{CS3}}$–$\overline{\text{CS0}}$	H	Z	O	Z	H	H	Z*4
	$\overline{\text{BS}}$	H	Z	O	Z	H	H	Z
	RD/$\overline{\text{WR}}$	H	Z	O	Z	H	H	Z*4
	$\overline{\text{RAS}}$, $\overline{\text{CE}}$	H	Z	O	Z	H	H	Z
	$\overline{\text{CAS}}$, $\overline{\text{OE}}$	H	Z	O	Z	H	H	Z
	$\overline{\text{CASHH}}$, DQMUU	H	Z	O	Z	H	H	Z
	$\overline{\text{CASHL}}$, DQMUL	H	Z	O	Z	H	H	Z
	$\overline{\text{CASLH}}$, DQMLU	H	Z	O	Z	H	H	Z
	$\overline{\text{CASLL}}$, DQMLL	H	Z	O	Z	H	H	Z
	$\overline{\text{RD}}$	H	Z	O	Z	H	H	Z
	CKE	H	H	O	H	O	O	H
	$\overline{\text{WAIT}}$	Z	Z	I	Z	Z	I	Ignored

Table A.1 Pin States During Resets, Power-Down State, and Bus-Released State (cont)

		Pin States						
		Reset Power-On		Reset Manual		Power-Down Modes		Bus-Released Mode
Category	Pin	Master	Slave	Bus Acquired	Bus Released	Standby	Sleep	
Direct memory access controller (DMAC)	DACK0, DACK1	H	H	H	H	K*3	O	O
	DREQ0, DREQ1	Z	Z	Z	Z	Z	I	I
16-bit free-running timer (FRT)	FTOA	L	L	L	L	K*3	O	O
	FTOB	L	L	L	L	K*3	O	O
	FTI	Z	Z	Z	Z	K*3	I	I
	FTCI	Z	Z	Z	Z	K*3	I	I
Serial communication interface (SCI)	RXD	Z	Z	Z	Z	K*3	I	I
	TXD	H	H	H	H	K*3	O	O
	SCK	Z	Z	Z	Z	K*3	IO	I

I: Input

O: Output

H: High-level output

L: Low-level output

Z: High impedance

K: Input pins are high impedance, output pins retain their state

Notes: 1. Depends on the clock mode (MD2–MD0 setting).

2. Low-level output in standby mode when the clock is paused.

3. When the high impedance bit (HIZ) in the standby control register (SBYCR) is set to 1, output pins become high impedance.

4. Input when the external bus cycle address monitor function is used.

Other: In sleep mode, if the DMAC is running, the address/data bus and bus control signals change according to the DMAC operation (the same applies during refreshing).

Appendix B List of Registers

B.1 List of I/O Registers

Address	Abbreviation of Register	Bit Name								Module
		Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	
H'FFFFFE00	SMR	C/$\overline{A}$	CHR	PE	O/$\overline{E}$	STOP	MP	CKS1	CKS0	SCI
H'FFFFFE01	BRR									
H'FFFFFE02	SCR	TIE	RIE	TE	RE	MPIE	TEIE	CKE1	CKE0	
H'FFFFFE03	TDR									
H'FFFFFE04	SSR	TDRE	RDRF	ORER	FER	PER	TEND	MPB	MPBT	
H'FFFFFE05	RDR									
H'FFFFFE06 to H'FFFFFE09	—	—	—	—	—	—	—	—	—	—
H'FFFFFE10	TIER	ICIE	—	—	—	OCIAE	OCIBE	OVIE	—	FRT
H'FFFFFE11	FTCSR	ICF	—	—	—	OCFA	OCFB	OVF	CCLRA	
H'FFFFFE12	FRC									
H'FFFFFE13	OCRA/B									
H'FFFFFE14										
H'FFFFFE15	TCR									
H'FFFFFE16		IEDGA	—	—	—	—	—	CKS1	CKS0	
H'FFFFFE17	TOCR	—	—	—	OCRS	—	—	OLVLA	OLVLB	
H'FFFFFE18	FICR									
H'FFFFFE19										
H'FFFFFE20 to H'FFFFFE59	—	—	—	—	—	—	—	—	—	—
H'FFFFFE60	IPRB	SCIIP3	SCIIP2	SCIIP1	SCIIP0	FRTIP3	FRTIP2	FRTIP1	FRTIP0	INTC
H'FFFFFE61		—	—	—	—	—	—	—	—	
H'FFFFFE62	VCRA	—	SERV6	SERV5	SERV4	SERV3	SERV2	SERV1	SERV0	
H'FFFFFE63		—	SRXV6	SRXV5	SRXV4	SRXV3	SRXV2	SRXV1	SRXV0	
H'FFFFFE64	VCRB	—	STXV6	STXV5	STXV4	STXV3	STXV2	STXV1	STXV0	
H'FFFFFE65		—	STEV6	STEV5	STEV4	STEV3	STEV2	STEV1	STEV0	

Address	Abbreviation of Register	Bit Name								Module
		Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	
H'FFFFFE66	VCRC	—	FICV6	FICV5	FICV4	FICV3	FICV2	FICV1	FICV0	INTC
H'FFFFFE67		—	FOCV6	FOCV5	FOCV4	FOCV3	FOCV2	FOCV1	FOCV0	
H'FFFFFE68	VCRD	—	FOVV6	FOVV5	FOVV4	FOVV3	FOVV2	FOVV1	FOVV0	
H'FFFFFE69		—	—	—	—	—	—	—	—	
H'FFFFFE6A to H'FFFFFE70	—	—	—	—	—	—	—	—	—	—
H'FFFFFE71	DRCR0	—	—	—	—	—	—	RS1	RS0	DMAC (channel 0)
H'FFFFFE72	DRCR1	—	—	—	—	—	—	RS1	RS0	DMAC (channel 1)
H'FFFFFE73 to H'FFFFFE7F	—	—	—	—	—	—	—	—	—	—
H'FFFFFE80	WTCSR*	OVF	WT/$\overline{IT}$	TME	—-	—	CKS2	CKS1	CKS0	WDT
H'FFFFFE81	WTCNT*									
H'FFFFFE82	—	—	—	—	—	—	—	—	—	
H'FFFFFE83	RSTCSR*	WOVF	RSTE	RSTS	—	—	—	—	—	
H'FFFFFE84 to H'FFFFFE90	—	—	—	—	—	—	—	—	—	
H'FFFFFE91	SBYCR	SBY	HIZ	—	MSTP4	MSTP3	MSTP2	MSTP1	MSTP0	Power-down
H'FFFFFE92	CCR	W1	W0	—	CP	TW	OC	ID	CE	Cache
H'FFFFFE93 to H'FFFFFE9F	—	—	—	—	—	—	—	—	—	—

Note: Address for reading. When writing, the address is H'FFFFFE80 for WTCSR and WTCNT, and H'FFFFFE82 for RSTCSR. See Section 12.2.4, Register Access, in Section 12, Watchdog Timer (WDT), for more information.

Address	Abbreviation of Register	Bit Name								Module
		Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	
H'FFFFFEE0	ICR	NIMIL	—	—	—	—	—	—	NIMIE	INTC
H'FFFFFEE1		—	—	—	—	—	—	—	VECMD	
H'FFFFFEE2	IPRA	DIVUIP3	DIVUIP2	DIVUIP1	DIVUIP0	DMACI3	DMACI2	DMACI1	DMACI0	
H'FFFFFEE3		WDTIP3	WDTIP2	WDTIP1	WDTIP0	—	—	—	—	
H'FFFFFEE4	VCRWDT	—	WITV6	WITV5	WITV4	WITV3	WITV2	WITV1	WITV0	
H'FFFFFEE5		—	BCMV6	BCMV5	BCMV4	BCMV3	BCMV2	BCMV1	BCMV0	
H'FFFFFEE6 to H'FFFFFEFF	—	—	—	—	—	—	—	—	—	—
H'FFFFFF00	DVSR									DIVU
H'FFFFFF01										
H'FFFFFF02										
H'FFFFFF03										
H'FFFFFF04	DVDNT									
H'FFFFFF05										
H'FFFFFF06										
H'FFFFFF07										
H'FFFFFF08	DVCR	—	—	—	—	—	—	—	—	
H'FFFFFF09		—	—	—	—	—	—	—	—	
H'FFFFFF0A		—	—	—	—	—	—	—	—	
H'FFFFFF0B		—	—	—	—	—	—	OVFIE	OVF	
H'FFFFFF0C	VCRDIV	—	—	—	—	—	—	—	—	
H'FFFFFF0D		—	—	—	—	—	—	—	—	
H'FFFFFF0E										
H'FFFFFF0F										
H'FFFFFF10	DVDNTH									
H'FFFFFF11										
H'FFFFFF12										
H'FFFFFF13										

Address	Abbreviation of Register	Bit Name								Module
		Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	
H'FFFFFF14	DVDNTL									DIVU
H'FFFFFF15										
H'FFFFFF16										
H'FFFFFF17										
H'FFFFFF18 to H'FFFFFF3F	—	—	—	—	—	—	—	—	—	—
H'FFFFFF40	BARAH	BAA31	BAA30	BAA29	BAA28	BAA27	BAA26	BAA25	BAA24	UBC (channel A)
H'FFFFFF41		BAA23	BAA22	BAA21	BAA20	BAA19	BAA18	BAA17	BAA16	
H'FFFFFF42	BARAL	BAA15	BAA14	BAA13	BAA12	BAA11	BAA10	BAA9	BAA8	
H'FFFFFF43		BAA7	BAA6	BAA5	BAA4	BAA3	BAA2	BAA1	BAA0	
H'FFFFFF44	BAMRAH	BAMA31	BAMA30	BAMA29	BAMA28	BAMA27	BAMA26	BAMA25	BAMA24	
H'FFFFFF45		BAMA23	BAMA22	BAMA21	BAMA20	BAMA19	BAMA18	BAMA17	BAMA16	
H'FFFFFF46	BAMRAL	BAMA15	BAMA14	BAMA13	BAMA12	BAMA11	BAMA10	BAMA9	BAMA8	
H'FFFFFF47		BAMA7	BAMA6	BAMA5	BAMA4	BAMA3	BAMA2	BAMA1	BAMA0	
H'FFFFFF48	BBRA	—	—	—	—	—	—	—	—	
H'FFFFFF49		CPA1	CPA0	IDA1	IDA0	RWA1	RWA0	SZA1	SZA0	
H'FFFFFF4A to H'FFFFFF5F	—	—	—	—	—	—	—	—	—	—
H'FFFFFF60	BARBH	BAB31	BAB30	BAB29	BAB28	BAB27	BAB26	BAB25	BAB24	UBC (channel B)
H'FFFFFF61		BAB23	BAB22	BAB21	BAB20	BAB19	BAB18	BAB17	BAB16	
H'FFFFFF62	BARBL	BAB15	BAB14	BAB13	BAB12	BAB11	BAB10	BAB9	BAB8	
H'FFFFFF63		BAB7	BAB6	BAB5	BAB4	BAB3	BAB2	BAB1	BAB0	
H'FFFFFF64	BAMRBH	BAMB31	BAMB30	BAMB29	BAMB28	BAMB27	BAMB26	BAMB25	BAMB24	
H'FFFFFF65		BAMB23	BAMB22	BAMB21	BAMB20	BAMB19	BAMB18	BAMB17	BAMB16	

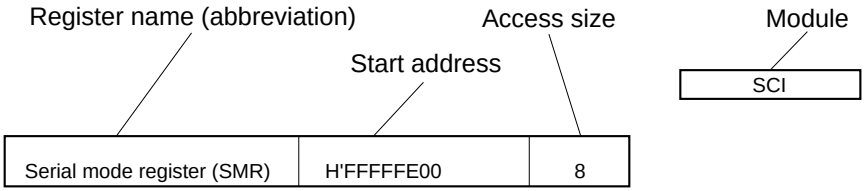
Address	Abbreviation of Register	Bit Name								Module
		Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	
H'FFFFFF66	BAMRBL	BAMB15	BAMB14	BAMB13	BAMB12	BAMB11	BAMB10	BAMB9	BAMB8	UBC (channel B)
H'FFFFFF67		BAMB7	BAMB6	BAMB5	BAMB4	BAMB3	BAMB2	BAMB1	BAMB0	
H'FFFFFF68	BBRB	—	—	—	—	—	—	—	—	
H'FFFFFF69		CPB1	CPB0	IDB1	IDB0	RWB1	RWB0	SZB1	SZB0	
H'FFFFFF6A to H'FFFFFF6F	—	—	—	—	—	—	—	—	—	
H'FFFFFF70	BDRBH	BDB31	BDB30	BDB29	BDB28	BDB27	BDB26	BDB25	BDB24	
H'FFFFFF71		BDB23	BDB22	BDB21	BDB20	BDB19	BDB18	BDB17	BDB16	
H'FFFFFF72	BDRBL	BDB15	BDB14	BDB13	BDB12	BDB11	BDB10	BDB9	BDB8	
H'FFFFFF73		BDB7	BDB6	BDB5	BDB4	BDB3	BDB2	BDB1	BDB0	
H'FFFFFF74	BDMRBH	BDMB31	BDMB30	BDMB29	BDMB28	BDMB27	BDMB26	BDMB25	BDMB24	
H'FFFFFF75		BDMB23	BDMB22	BDMB21	BDMB20	BDMB19	BDMB18	BDMB17	BDMB16	
H'FFFFFF76	BDMRBL	BDMB15	BDMB14	BDMB13	BDMB12	BDMB11	BDMB10	BDMB9	BDMB8	
H'FFFFFF77		BDMB7	BDMB6	BDMB5	BDMB4	BDMB3	BDMB2	BDMB1	BDMB0	
H'FFFFFF78	BRCR	CMFCA	CMFPA	EBBE	UMD	—	PCBA	—	—	
H'FFFFFF79		CMFCB	CMFPB	—	SEQ	DBEB	PCBB	—	—	
H'FFFFFF7A to H'FFFFFF7F	—	—	—	—	—	—	—	—	—	—

Address	Abbreviation of Register	Bit Name								Module
		Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	
H'FFFFFF80	SAR0									DMAC (channel 0)
H'FFFFFF81										
H'FFFFFF82										
H'FFFFFF83										
H'FFFFFF84	DAR0									
H'FFFFFF85										
H'FFFFFF86										
H'FFFFFF87										
H'FFFFFF88	TCR0	—	—	—	—	—	—	—	—	
H'FFFFFF89										
H'FFFFFF8A										
H'FFFFFF8B										
H'FFFFFF8C	CHCR0	—	—	—	—	—	—	—	—	
H'FFFFFF8D		—	—	—	—	—	—	—	—	
H'FFFFFF8E		DM1	DM0	SM1	SM0	TS1	TS0	AR	AM	
H'FFFFFF8F		AL	DS	DL	TB	TA	IE	TE	DE	
H'FFFFFF90	SAR1									DMAC (channel 1)
H'FFFFFF91										
H'FFFFFF92										
H'FFFFFF93										
H'FFFFFF94	DAR1									
H'FFFFFF95										
H'FFFFFF96										
H'FFFFFF97										
H'FFFFFF98	TCR1	—	—	—	—	—	—	—	—	
H'FFFFFF99										
H'FFFFFF9A										
H'FFFFFF9B										

Address	Abbreviation of Register	Bit Name								Module
		Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	
H'FFFFFF9C	CHCR1	—	—	—	—	—	—	—	—	DMAC (channel 1)
H'FFFFFF9D		—	—	—	—	—	—	—	—	
H'FFFFFF9E		DM1	MD0	SM1	SM0	TS1	TS0	AR	AM	
H'FFFFFF9F		AL	DS	DL	TB	TA	IE	TE	DE	
H'FFFFFFA0	VCRMA0	—	—	—	—	—	—	—	—	DMAC (channel 0)
H'FFFFFFA1		—	—	—	—	—	—	—	—	
H'FFFFFFA2		—	—	—	—	—	—	—	—	
H'FFFFFFA3		VC7	VC6	VC5	VC4	VC3	VC2	VC1	VC0	
H'FFFFFFA4 to H'FFFFFFA7	—	—	—	—	—	—	—	—	—	
H'FFFFFFA8	VCRDMA1	—	—	—	—	—	—	—	—	DMAC (channel 1)
H'FFFFFFA9		—	—	—	—	—	—	—	—	
H'FFFFFFAA		—	—	—	—	—	—	—	—	
H'FFFFFFAB		VC7	VC6	VC5	VC4	VC3	VC2	VC1	VC0	
H'FFFFFFAC to H'FFFFFFAF	—	—	—	—	—	—	—	—	—	
H'FFFFFFB0	DMAOR	—	—	—	—	—	—	—	—	DMAC (channels 0 and 1)
H'FFFFFFB1		—	—	—	—	—	—	—	—	
H'FFFFFFB2		—	—	—	—	—	—	—	—	
H'FFFFFFB3		—	—	—	—	PR	AE	NMIF	DME	
H'FFFFFFB4 to H'FFFFFFDF	—	—	—	—	—	—	—	—	—	

Address	Abbreviation of Register	Bit Name								Module
		Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	
H'FFFFFFE0		—	—	—	—	—	—	—	—	BSC
H'FFFFFFE1		—	—	—	—	—	—	—	—	
H'FFFFFFE2	BCR1	MASTR	—	—	ENDIAN	BSTROM	PSHR	AHLW1	AHLW0	
H'FFFFFFE3		A1LW1	A1LW0	A0LW1	A0LW0	—	DRAM2	DRAM1	DRAM0	
H'FFFFFFE4	BCR2	—	—	—	—	—	—	—	—	
H'FFFFFFE5		—	—	—	—	—	—	—	—	
H'FFFFFFE6		—	—	—	—	—	—	—	—	
H'FFFFFFE7		A3SZ1	A3SZ0	A2SZ1	A2SZ0	A1SZ1	A1SZ0	—	—	
H'FFFFFFE8	WCR	—	—	—	—	—	—	—	—	
H'FFFFFFE9		—	—	—	—	—	—	—	—	
H'FFFFFFEA		IW31	IW30	IW20	IW21	IW10	IW11	IW01	IW00	
H'FFFFFFEB		W31	W30	W20	W21	W10	W11	W01	W00	
H'FFFFFFEC	MCR	—	—	—	—	—	—	—	—	
H'FFFFFFED		—	—	—	—	—	—	—	—	
H'FFFFFFEE		TRP	RCD	TRWL	TRAS1	TRS0	BE	RASD	—	
H'FFFFFFEF		AMX2	SZ	AMX1	AMX0	RFSH	RMD	—	—	
H'FFFFFFF0	RTCSR	—	—	—	—	—	—	—	—	
H'FFFFFFF1		—	—	—	—	—	—	—	—	
H'FFFFFFF2		—	—	—	—	—	—	—	—	
H'FFFFFFF3		CMF	CMIE	CKS2	CKS1	CKS0	—	—	—	
H'FFFFFFF4	RTCNT	—	—	—	—	—	—	—	—	
H'FFFFFFF5										
H'FFFFFFF6										
H'FFFFFFF7										
H'FFFFFFF8	RTCOR	—	—	—	—	—	—	—	—	
H'FFFFFFF9										
H'FFFFFFFA										
H'FFFFFFFB										
H'FFFFFFFC to H'FFFFFFFF	—	—	—	—	—	—	—	—	—	—

B.2 Register Chart



Register overview

Item	Bit							
	7	6	5	4	3	2	1	0
Bit Name	C/$\overline{A}$	CHR	PE	O/$\overline{E}$	STOP	MP	CKS1	CKS0
Initial Value	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit function

Bit	Bit Name	Value		Description
7	Communication mode (C/$\overline{A}$)	0		Asynchronous mode (Initial value)
		1		Clocked synchronous mode
6	Character length (CHR)	0		Eight-bit data (Initial value)
		1		Seven-bit data
5	Parity enable (PE)	0		Parity bit not added or checked (Initial value)
		1		Parity bit added and checked
4	Parity mode (O/$\overline{E}$)	0		Even parity (Initial value)
		1		Odd parity
3	Stop bit length (STOP)	0		One stop bit (Initial value)
		1		Two stop bits
2	Multiprocessor mode (MP)	0		Multiprocessor function disabled (Initial value)
		1		Multiprocessor format selected
1	Clock select 1 and 0 (CKS1, CKS0)	0	0	ϕ/4 (Initial value)
		0	1	ϕ/16
0		1	0	ϕ/64
		1	1	ϕ/256

Bit number

Bit name (abbreviation)

Bit value (When there is a set of bits, the upper bit is on the left, and the lower bit on the right.)

Bit description

Serial mode register (SMR)	H'FFFFFE00	8

Item	Bit 7	6	5	4	3	2	1	0
Bit Name	C/$\overline{A}$	CHR	PE	O/E	STOP	MP	CKS1	CKS0
Initial Value	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Value	Description
7	Communication mode (C/A)	0	Asynchronous mode (Initial value)
		1	Clocked synchronous mode
6	Character length (CHR)	0	Eight-bit data (Initial value)
		1	Seven-bit data
5	Parity enable (PE)	0	Parity bit not added or checked (Initial value)
		1	Parity bit added and checked
4	Parity mode (OE)	0	Even parity (Initial value)
		1	Odd parity
3	Stop bit length (STOP)	0	One stop bit (Initial value)
		1	Two stop bits
2	Multiprocessor mode (MP)	0	Multiprocessor function disabled (Initial value)
		1	Multiprocessor format selected
1	Clock select 1 and 0 (CKS1, CKS0)	0 0	ϕ/4 (Initial value)
		0 1	ϕ/16
0		1 0	ϕ/64
		1 1	ϕ/256

Bit rate register (BRR)	H'FFFFFE01	8

Item	Bit 7	6	5	4	3	2	1	0
Bit Name								
Initial Value	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Description
7 to 0	(Bit rate setting)	Sets serial transmit/receive bit rate

Serial control register (SCR)	H'FFFFFE02	8

Item	Bit 7	6	5	4	3	2	1	0
Bit Name	TIE	RIE	TE	RE	MPIE	TEIE	CKE1	CKE0
Initial Value	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Value	Description
7	Transmit interrupt enable (TIE)	0	Transmit-data-empty interrupt request (TXI) is disabled (Initial value)
		1	Transmit-data-empty interrupt request (TXI) is enabled
6	Receive interrupt enable (RIE)	0	Receive-data-full interrupt (RXI) and receive-error interrupt (ERI) requests are disabled (Initial value)
		1	Receive-data-full interrupt (RXI) and receive-error interrupt (ERI) requests are enabled
5	Transmit enable (TE)	0	Transmitter disabled (Initial value)
		1	Transmitter enabled
4	Receive enable (RE)	0	Receiver disabled (Initial value)
		1	Receiver enabled
3	Multiprocessor interrupt enable (MPIE)	0	Multiprocessor interrupts are disabled (nomal receive operation) (Initial value). MPE is cleared to 0 when MPIE is cleared to 0, or the multiprocessor bit (MPB) is set to 1 in receive data.
		1	Multiprocessor interrupts are enabled. Receive-data-full interrupt requests (RXI), receive-error interrupt requests (ERI), and setting of the RDRF, FER, and ORER status flags in the serial status register (SSR) are disabled until the multiprocessor bit is set to 1.
2	Transmit-end interrupt enable (TEIE)	0	Transmit-end interrupt (TEI) requests are disabled (Initial value)
		1	Transmit-end interrupt (TEI) requests are enabled
1, 0	Clock enable 1 and 0 (CKE1 and CKE2)	0 0	Asynchronous mode: Internal clock, SCK pin used for input pin (input signal is ignored or output pin output level is undefined)
			Clocked synchronous mode: Internal clock, SCK pin used for synchronous clock output
		0 1	Asynchronous mode: Internal clock, SCK pin used for clock output
			Clocked synchronous mode: Internal clock, SCK pin used for synchronous clock output
		1 0	Asynchronous mode: Internal clock, SCK pin used for clock input
			Clocked synchronous mode: Internal clock, SCK pin used for synchronous clock input
		1 1	Asynchronous mode: Internal clock, SCK pin used for clock input
			Clocked synchronous mode: Internal clock, SCK pin used for synchronous clock input

Transmit data register (TDR)	H'FFFFFE03	8

Item	Bit 7	6	5	4	3	2	1	0
Bit Name								
Initial Value	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Description
7 to 0	(Stores transmit data)	Stores data for serial transmission

Serial status register (SSR)	H'FFFFFE04	8

Item	Bit 7	6	5	4	3	2	1	0
Bit Name	TDRE	RDRF	ORER	FER	PER	TEND	MPB	MPBT
Initial Value	1	0	0	0	0	1	0	0
R/W	R(W)*	R(W)*	R(W)*	R(W)*	R(W)*	R	R	R/W

Note: Only 0 can be written to clear flags.

Bit	Bit Name	Value	Description
7	Transmit data register empty (TDRE)	0	TDR contains valid transmit data TDRE is cleared to 0 when software reads TDRE after it has been set to 1, then writes 0 in TDRE, or the DMAC writes data in TDR.
		1	TDR does not contain valid transmit data (Initial value) TDRE is set to 1 when the chip is reset or enters standby mode, the TE bit in the serial control register (SCR) is cleared to 0, or TDR contents are loaded into TSR, so new data can be written in TDR.
6	Receive data register full (RDRF)	0	RDR does not contain valid received data (Initial value) RDRF is cleared to 0 when the chip is reset or enters standby mode, software reads RDRF after it has been set to 1, then writes 0 in RDRF, or the DMAC reads data from RDR.
		1	RDR contains valid received data RDRF is set to 1 when serial data is received normally and transferred from RSR to RDR.

Bit	Bit Name	Value	Description
5	Overrun error (ORER)	0	Receiving is in progress or has ended normally (Initial value) ORER is cleared to 0 when the chip is reset or enters standby mode, or software reads ORER after it has been set to 1, then writes 0 in ORER.
		1	A receive overrun error occurred ORER is set to 1 if reception of the next serial data ends when RDRF is set to 1.
4	Framing error (FER)	0	Receiving is in progress or has ended normally (Initial value) FER is cleared to 0 when the chip is reset or enters standby mode, or software reads FER after it has been set to 1, then writes 0 in FER.
		1	A receive framing error occurred FER is set to 1 if the stop bit at the end of receive data is checked and found to be 0.
3	Parity error (PER)	0	Receiving is in progress or has ended nomally (Initial value) PER is cleared to 0 when the chip is reset or enters standby mode or software reads PER after it has been set to 1, then writes 0 in PER.
		1	A receive parity error occurred PER is set to 1 if the number of ls in receive data, including the parity bit, does not match the even or odd parity setting of the parity mode bit (O/$\overline{E}$) in the serial mode register (SMR).
2	Transmit end (TEND)	0	Transmission is in progress TEND is cleared to 0 when software reads TDRD after it has been set to 1, then writes 0 in TDRE, or the DMAC writes data in TDR.
		1	End of transmission (Initial value) TEND is set to 1 when the chip is reset or enters standby mode, TE is cleared to 0 in the serial control register (SCR), or TDRE is 1 when the last bit of a one-byte serial character is transmitted.
1	Multiprocessor bit (MPB)	0	Multiprocessor bit value in receive data is 0 (Initial value)
		1	Multiprocessor bit value in receive data is 1
0	Multiprocessor bit transfer (MPBT)	0	Multiprocessor bit value in transmit data is 0 (Initial value)
		1	Multiprocessor bit value in transmit data is 1

Receive data register (RDR)	H'FFFFFE05	8

	Bit							
Item	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Bit Name								
Initial Value	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Bit	Bit Name	Description
7 to 0	(Stores serial receive data)	Stores the received serial data

Timer interrupt enable register (TIER)	H'FFFFFE10	8

	Bit							
Item	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	ICIE	—	—	—	OCIAE	OCIBE	OVIE	—
Initial Value	0	0	0	0	0	0	0	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Value	Description
7	Input capture interrupt enable (ICIE)	0	Disables interrupt requests (ICI) from ICF (Initial value)
		1	Enables interrupt requests (ICI) from the ICF
3	Output compare interrupt A enable (OCIAE)	0	Disables interrupt requests (OCIA) from OCFA (Initial value)
		1	Enables interrupt requests (OCIA) from OCFA
2	Output compare interrupt B enable (OCIBE)	0	Disables interrupt requests (OCIB) from OCFB (Initial value)
		1	Enables interrupt requests (OCIB) from OCFB
1	Timer overflow interrupt enable (OVIE)	0	Disables interrupt requests (OVI) from OVF (Initial value)
		1	Enables interrupt requests (OVI) from OVF

Free-running timer control/status register (FTCSR)	H'FFFFFE11	8

	Bit							
Item	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Bit Name	ICF	—	—	—	OCFA	OCFB	OVF	CCLRA
Initial Value	0	0	0	0	0	0	0	0
R/W	R/(W)*	—	—	—	R/(W)*	R/(W)*	R/(W)*	R/W

Note: For bits 7, and 3 to 1, the only value that can be written is 0 (to clear the flags)

Bit	Bit Name	Value	Description
7	Input capture flag (ICF)	0	Clear conditions: When ICF = 1, ICF is read and then 0 is written to it (Initial value)
		1	Set conditions: When FRC value is sent to ICR by the input capture signal
3	Output compare flag A (OCFA)	0	Clear conditions: When OCFA = 1, OCFA is read and then 0 is written to it (Initial value)
		1	Set conditions: When FRC value becomes equal to OCRA

Bit	Bit Name	Value	Description
2	Output compare flag B (OCFB)	0	Clear conditions: When OCFB = 1, OCFB is read and then 0 is written to it (Initial value)
		1	Set conditions: When FRC value becomes equal to OCRB
1	Timer overflow flag (OVF)	0	Clear conditions: When OVF = 1, OVF is read and then 0 is written to it (Initial value)
		1	Set conditions: When FRC value changes from H'FFFF to H'0000
0	Counter clear A (CCLRA)	0	Disables FRC clear (Initial value)
		1	Clears FRC on compare match A

Free-running counter (FRC)	H'FFFFFE12(FRCH) H'FFFFFE13(FRCL)	16*

Note: Access FRCH first and then FRCL, two 8-bit units.

	Bit															
Item	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit Name																
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Description
15 to 0	(Count value)	Counts input clock pulses

Output compare register A/B*1 (OCRA/B)	H'FFFFFE14(OCRA/BH H'FFFFFE15(OCRA/BL)	16*2

Notes: 1. Switch registers with OCRS in TOCR.
2. Access OCRA/BH first and then OCRA/BL, in two 8-bit units.

	Bit															
Item	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit Name																
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Description
15 to 0	(FRC value comparison)	Sets OCFA when OCFA = FRC Sets OCFB when OCFB = FFC

Timer control register (TCR)	H'FFFFFE16	8

	Bit							
Item	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Bit Name	IEDGA	—	—	—	—	—	CKS1	CKS0
Initial Value	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Value	Description
7	Input edge select (IEDG)	0	Captures input on falling edge (Initial value)
		1	Captures input on rising edge
1, 0	Clock selects (CKS1 and CKS0)	0 0	Internal clock: count on ϕ/8 (Initial value)
		0 1	Internal clock: count on ϕ/32
		1 0	Internal clock: count on ϕ/128
		1 1	External clock: count on rising edge

Timer output compare control register (TOCR)	H'FFFFFE17	8

	Bit							
Item	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Bit Name	—	—	—	OCRS	—	—	OLVLA	OLVLB
Initial Value	1	1	1	0	0	0	0	0
R/W	—	—	—	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Value	Description	
4	Output compare register select (OCRS)	0	Selects OCRA register	(Initial value)
		1	Selects OCRB register	
1	Output level A (OLVLA)	0	Outputs 0 on compare match A	(Initial value)
		1	Outputs 1 on compare match A	
0	Output level B (OLVLB)	0	Outputs 0 on compare match B	(Initial value)
		1	Outputs 1 on compare match B	

Input capture register (ICR)	H'FFFFFE18 (ICRH) H'FFFFFE19 (ICRL)	16*

Note: Access ICRH first and then ICRL, in two 8-bit units.

	Bit															
Item	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Bit Name																
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	**Bit Name**	**Description**
15 to 0	(Stores FRC value)	Stores FRC value when an input capture signal occurs

Interrupt priority level setting register A (IPRA)	H'FFFFFEE2	8/16

	Bit															
Item	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Bit Name	DIVU IP3	DIVU IP2	DIVU IP1	DIVU IP0	DMAC IP3	DMAC IP2	DMAC IP1	DMAC IP0	WDT P3	WDT IP2	WDT IP1	WDT IP0	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R

Bit	Bit Name	Description
15 to 12	Division unit (DIVU) interrupt priority level (DIVUIP3–DIVUIP0)	These bits set the division unit (DIVU) interrupt priority level
11 to 8	DMA controller interrupt priority level (DMACIP3–DMACIP0)	These bits set the DMA controller (DMAC) interrupt priority level
7 to 4	Watchdog timer (WDT) interrupt priority level (WDTIP3–WDTIP0)	These bits set the watchdog timer (WDT) interrupt priority level and bus state controller (BSC) interrupt priority level

Interrupt priority level setting register B (IPRB)	H'FFFFFE60	8/16

	Bit															
Item	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Bit Name	SCI IP3	SCI IP2	SCI IP1	SCI IP0	FRT IP3	FRT IP2	FRT IP1	FRT IP0	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R

Bit	Bit Name	Description
15 to 12	Serial communication interface (SCI) interrupt priority level (SCIIP3–SCIIP0)	These bits set the serial communication interface (SCI) interrupt priority level
11 to 8	Free-running timer (FRT) interrupt priority level (FRTIP3–FRTIP0)	These bits set the free-running timer (FRT) interrupt priority level

Vector number setting register A (VCRA)	H'FFFFFE62	8/16

Item	Bit 15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit Name	—	SERV6	SERV5	SERV4	SERV3	SERV2	SERV1	SERV0	—	SRXV6	SRXV5	SRXV4	SRXV3	SRXV2	SRXV1	SRXV0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Description
14 to 8	Serial communication interface (SCI) receive-error interrupt vector number (SERV6–SERV0)	These bits set the vector number for the serial communication interface (SCI) receive-error interrupt (ERI)
6 to 0	Serial communication interface (SCI) receive-data-full interrupt vector number (SRXV6–SRXV0)	These bits set the vector number for the serial communication interface (SCI) receive-data-full interrupt (RXI)

Vector number setting register B (VCRB)	H'FFFFFE64	8/16

Item	Bit 15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit Name	—	STXV6	STXV5	STXV4	STXV3	STXV2	STXV1	STXV0	—	STEV6	STEV5	STEV4	STEV3	STEV2	STEV1	STEV0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Description
14 to 8	Serial communication interface (SCI) transmit-data-empty interrupt vector number (STXV6–STXV0)	These bits set the vector number for the serial communication interface (SCI) transmit-data-empty interrupt (TXI)
6 to 0	Serial communication interface (SCI) transmit-end interrupt vector number (STEV6–STEV0)	These bits set the vector number for the serial communication interface (SCI) transmit-end interrupt (TEI)

Vector number setting register C (VCRC)	H'FFFFFE66	8/16

	Bit															
Item	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Bit Name	—	FIC V6	FIC V5	FIC V4	FIC V3	FIC V2	FIC V1	FIC V0	—	FOC V6	FOC V5	FOC V4	FOC V3	FOC V2	FOC V1	FOC V0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Description
14 to 8	Free-running timer (FRT) input-capture interrupt vector number (FICV6–FICV0)	These bits set the vector number for the free-running timer (FRT) input-capture interrupt (ICI)
6 to 0	Free-running timer (FRT) output-compare interrupt vector number (FOCV6–FOCV0)	These bits set the vector number for the free-running timer (FRT) output-compare interrupt (OCI)

Vector number setting register D (VCRD)	H'FFFFFE68	8/16

	Bit															
Item	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Bit Name	—	FOV V6	FOV V5	FOV V4	FOV V3	FOV V2	FOV V1	FOV V0	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R

Bit	Bit Name	Description
14 to 8	Free-running timer (FRT) overflow interrupt vector number (FOVV6–FOVV0)	These bit set the vector number for the free-running timer(FRT) overflow interrupt (OVI)

Vector number setting register WDT (VCRWDT)	H'FFFFFEE4	8/16

	Bit															
Item	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Bit Name	—	WITV6	WITV5	WITV4	WITV3	WITV2	WITV1	WITV0	—	BCMV6	BCMV5	BCMV4	BCMV3	BCMV2	BCMV1	BCMV0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Description
14 to 8	Watchdog timer (WDT) interval interupt vector number (WITV6–WITV0)	These bits set the vector number for the interval interrupt (ITI) of the watchdog timer (WDT)
6 to 0	Bus state controller (BSC) compare match interrupt vector number (BCMV6–BCMV0)	These bits set the vector number for the compare match interrupt (CMI) of the bus state controller (BSC)

Vector number setting register DIV (VCRDIV)	H'FFFFFF0C	32

	Bit															
Item	**31**	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**
Bit Name	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Item	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Bit Name																
Initial Value	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Description
15 to 0	(Vector number setting)	These bits set the vector number for the interrupt when caused by overflow or underflow of the division unit

Vector number setting registers DMA0 and DMA1 (VCRDMA0, VCRDMA1)	H'FFFFFFA0 (channel 0) H'FFFFFFA8 (channel 1)	32

	Bit															
Item	**31**	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**
Bit Name	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Item	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Bit Name	—	—	—	—	—	—	—	—	VC7	VC6	VC5	VC4	VC3	VC2	VC1	VC0
Initial Value	0	0	0	0	0	0	0	0	—	—	—	—	—	—	—	—
R/W	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Description
7 to 0	Vector number bits (VC7–VC0)	These bits set the vector number at the end of DMA transfer

Interrupt control register (ICR)	H'FFFFFEE0	8/16

	Bit															
Item	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Bit Name	NMIL	—	—	—	—	—	—	NMIE	—	—	—	—	—	—	—	VECMD
Initial Value	0/1*	0	0	0	0	0	0	0	—	—	—	—	—	—	—	—
R/W	R	R	R	R	R	R	R	R/W	R	R	R	R	R	R	R	R/W

Note: When NMI input is high: 1; when NMI input is low: 0

Bit	Bit Name	Value	Description
15	NMI input level (NMIL)	0	NMI input level is low
		1	NMI input level is high
8	NMI edge select (NMIE)	0	Interrupt request is detected on falling edge of NMI input (Initial value)
		1	Interrupt request is detected on rising edge of NMI input
1	RL interrupt vector mode select (VECMD)	0	Auto-vector mode, automatically set internall (Initial value)
		1	External vector mode, external input

Watchdog timer control/status register (WTCSR)	H'FFFFFE80	8 (read) 16 (write)

Item	Bit 7	6	5	4	3	2	1	0
Bit Name	OVF	WT/$\overline{IT}$	TME	—	—	CKS2	CKS1	CKS0
Initial Value	0	0	0	1	1	0	0	0
R/W	R/(W)*	R/W	R/W	—	—	R/W	R/W	R/W

Note: WTCSR differs from other registers in being more difficult to write. See section 12.2.4, Register Access, for details.

Bit	Bit Name	Value	Description
7	Overflow flag (OVF)	0	No overflow of WTCNT in interval timer mode (Initial value) Cleared by reading OVF, then writing 0 in OVF
		1	WTCNT overflow in interval timer mode
6	Timer mode select (WT/$\overline{IT}$)	0	Interval timer mode: Interval timer interrupt (ITI) request to the CPU when WTCNT overflows (Initial value)
		1	Watchdog timer mode: $\overline{WDTOVF}$ signal is output externally when WTCNT overflows
5	Timer enable (TME)	0	Timer disabled: WTCNT is initialized to H'00 and count-up stops (Initial value)
		1	Timer enabled: WTCNT starts counting A $\overline{WDTOVF}$ signal or interrupt is generated when WTCNT overflows

Bit	Bit Name	CKS2	CKS1	CKS0	Clock Source	Overflow Interval (ϕ = 28.7 MHz)
2 to 0	Clock select 2 to 0 (CKS2 to CKS0)	0	0	0	ϕ/2 (Initial value)	17.8µs
		0	0	1	ϕ/64	570.8µs
		0	1	0	ϕ/128	1.1ms
		0	1	1	ϕ/256	2.2ms
		1	0	0	ϕ/512	4.5ms
		1	0	1	ϕ/1024	9.1ms
		1	1	0	ϕ/4096	35.5ms
		1	1	1	ϕ/8192	73.0ms

Watchdog timer counter (WTCNT)	H'FFFFFE80 (write) H'FFFFFE81 (read)	16 (write) 8 (read)

	Bit							
Item	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Bit Name								
Initial Value	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Description
7 to 0	(Count value)	Input clock count value

Reset control/status register (RSTCSR)	H'FFFFFE82 (write) H'FFFFFE83 (read)	16 (write) 8 (read)

	Bit							
Item	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Bit Name	WOVF	RSTE	RSTS	—	—	—	—	—
Initial Value	0	0	0	1	1	1	1	1
R/W	R/(W)*	R/W	R/W	—	—	—	—	—

Note: Only 0 can be written in bit 7 to clear the flag.

Bit	Bit Name	Value	Description
7	Watchdog timer overflow flag (WOVF)	0	No WTCNT overflow in watchdog timer mode (Initial value) Cleared when software reads WOVF, then writes 0 in WOVF
		1	Set by WTCNT overflow in watchdog timer mode
6	Reset enable (RSTE)	0	No internal reset when WTCNT overflows (Initial value)
		1	Internal reset when WTCNT overflows
5	Reset select (RSTS)	0	Power-on reset (Initial value)
		1	Manual reset

Divisor register (DVSR)	H'FFFFFE00	32

Item	Bit 31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Bit Name																
Initial Value	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Item	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Bit Name																
Initial Value	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Description
31 to 0	(Written with divisor)	Used to write the divisor for the operation

Dividend register L for 32-bit division (DVDNT)	H'FFFFFE04	32

Item	Bit 31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Bit Name																
Initial Value	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Item	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Bit Name																
Initial Value	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Description
31 to 0	(Dividend setting)	Set with the 32-bit dividend used for 32-bit/32-bit division operations

Division control register (DVCR)	H'FFFFFF08	16/32

	Bit															
Item	**31**	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**
Bit Name	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Item	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Bit Name	—	—	—	—	—	—	—	—	—	—	—	—	—	—	OVFIE	OVF
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Value	Description
1	OVF interrupt enable (OVFIE)	0	Disables interrupt request (OVFI) caused by OVF (Initial value)
		1	Enables interrupt request (OVFI) caused by OVF
0	Overflow flag (OVF)	0	No overflow has occurred (Initial value)
		1	Overflow has occurred

Dividend register H (DVDNTH)	H'FFFFFF10	32

	Bit															
Item	**31**	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**
Bit Name																
Initial Value	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Item	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Bit Name																
Initial Value	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Description
31 to 1	(Dividend setting)	Set with the upper 32 bits of the dividend used for 64-bit/32-bit division operations

Dividend regiater L (DVDNTL)	H'FFFFFF14	32

Item	Bit 31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Bit Name																
Initial Value	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Item	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Bit Name																
Initial Value	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Description
31 to 1	(Dividend setting)	Set with the lower 32 bits of the dividend used for 64-bit/32-bit division operations

Break address register AH (BARAH)	H'FFFFFF40	16/32

	Bit															
Item	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Bit Name	BAA 31	BAA 30	BAA 29	BAA 28	BAA 27	BAA 26	BAA 25	BAA 24	BAA 23	BAA 22	BAA 21	BAA 20	BAA 19	BAA 18	BAA 17	BAA 16
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Description
15 to 0	Break address BAA31–BAA16	These bits specify the upper bits (bit 31 to bit 16) of the channel A break condition address

Break address register AL (BARAL)	H'FFFFFF42	16

	Bit															
Item	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Bit Name	BAA 15	BAA 14	BAA 13	BAA 12	BAA 11	BAA 10	BAA 9	BAA 8	BAA 7	BAA 6	BAA 5	BAA 4	BAA 3	BAA 2	BAA 1	BAA 0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Description
15 to 0	Break address BAA15–BAA0	These bits specify the lower bits (bit 15 to bit 0) of the channel A break condition address

Break address mask register AH (BAMRAH)	H'FFFFFF44	16/32

Item	Bit 15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit Name	BAMA31	BAMA30	BAMA29	BAMA28	BAMA27	BAMA26	BAMA25	BAMA24	BAMA23	BAMA22	BAMA21	BAMA20	BAMA19	BAMA18	BAMA17	BAMA16
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Value	Description
15 to 0	Break address BAMA31–BAMA16	0	Channel A break address BAAn is included in the break conditions (Initial value)
		1	Channel A break address BAAn is not included in the break conditions

n = 31 to 16

Break address mask register AL (BAMRAL)	H'FFFFFF46	16

Item	Bit 15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit Name	BAMA15	BAMA14	BAMA13	BAMA12	BAMA11	BAMA10	BAMA9	BAMA8	BAMA7	BAMA6	BAMA5	BAMA4	BAMA3	BAMA2	BAMA1	BAMA0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Value	Description
15 to 0	Break address BAMA15–BAMA0	0	Channel A break address BAAn is included in the break conditions (Initial value)
		1	Channel A break address BAAn is not included in the break conditions

n = 15 to 0

Break bus cycle register A (BBRA)	H'FFFFFF48	16/32

Item	Bit 15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit Name	—	—	—	—	—	—	—	—	CPA1	CPA0	IDA1	IDA0	RWA1	RWA0	SZA1	SZA0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Value	Description
7, 6	CPU cycle/peripheral cycle select A (CPA1, CPA0)	0 0	No channel A user break interrupt generated (Initial value)
		0 1	Break only on CPU cycles
		1 0	Break only on peripheral cycles
		1 1	Break on both CPU and peripheral cycles
5, 4	Instruction fetch/data access select A (IDA1, IDA0)	0 0	No channel A user break interrupt generated (Initial value)
		0 1	Break only on instruction fetch cycles
		1 0	Break only on data access cycles
		1 1	Break on both instruction fetch and data access cycles
3, 2	Read/write select A (RWA1, RWA0)	0 0	No channel A user break interrupt generated (Initial value)
		0 1	Break only on read cycles
		1 0	Break only on write cycles
		1 1	Break on both read and write cycles
1, 0	Operand size select A (SZA1, SZA0)	0 0	Operand size is not a break condition (Initial value)
		0 1	Break on byte access
		1 0	Break on word access
		1 1	Break on longword access

Break address register BH (BARBH)	H'FFFFFF60	16/32

Item	Bit 15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit Name	BAB31	BAB30	BAB29	BAB28	BAB27	BAB26	BAB25	BAB24	BAB23	BAB22	BAB21	BAB20	BAB19	BAB18	BAB17	BAB16
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Description
15 to 0	Break address BAB31–BAB16	These bits specify the upper bits (bit 31 to bit 16) of the channel B break condition address

Break address register BL (BARBL)	H'FFFFFF62	16

Item	Bit 15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit Name	BAB15	BAB14	BAB13	BAB12	BAB11	BAB10	BAB9	BAB8	BAB7	BAB6	BAB5	BAB4	BAB3	BAB2	BAB1	BAB0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Description
15 to 0	Break address BAB15–BAB0	These bits specify the lower bits (bit 15 to bit 0) of the channel B break condition address

Break address mask register BH (BAMRBH)	H'FFFFFF64	16/32

Item	Bit 15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit Name	BAMB31	BAMB30	BAMB29	BAMB28	BAMB27	BAMB26	BAMB25	BAMB24	BAMB23	BAMB22	BAMB21	BAMB20	BAMB19	BAMB18	BAMB17	BAMB16
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Value	Description
15 to 0	Break address mask BAMB31–BAMB16	0	Channel B break address BABn is included in the break conditions (Initial value)
		1	Channel B break address BABn is not included in the break conditions

n = 31 to 16

Break address mask register BL (BAMRBL)	H'FFFFFF66	16

	Bit															
Item	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Bit Name	BAMB15	BAMB14	BAMB13	BAMB12	BAMB11	BAMB10	BAMB9	BAMB8	BAMB7	BAMB6	BAMB5	BAMB4	BAMB3	BAMB2	BAMB1	BAMB0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Value	Description
15 to 0	Break address mask BAMB15–BAMB0	0	Channel B break address BABn is included in the break conditions (Initial value)
		1	Channel B break address BABn is not included in the break conditions

n = 15 to 0

Break data register BH (BDRBH)	H'FFFFFF70	16/32

	Bit															
Item	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Bit Name	BDB31	BDB30	BDB29	BDB28	BDB27	BDB26	BDB25	BDB24	BDB23	BDB22	BDB21	BDB20	BDB19	BDB18	BDB17	BDB16
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Description
15 to 0	Break data BDB31–BDB16	These bits specify the upper bits (bit 31 to bit 16) of the channel B break condition data

Break data register BL (BDRBL)	H'FFFFFF72	16/32

Item	Bit 15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit Name	BDB15	BDB14	BDB13	BDB12	BDB11	BDB10	BDB9	BDB8	BDB7	BDB6	BDB5	BDB4	BDB3	BDB2	BDB1	BDB0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Description
15 to 0	Break data BDB15–BDB0	These bits specify the lower bits (bit 15 to bit 0) of the channel B break condition data

Break data mask register BH (BDMRBH)	H'FFFFFF74	16/32

Item	Bit 15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit Name	BDMB31	BDMB30	BDMB29	BDMB28	BDMB27	BDMB26	BDMB25	BDMB24	BDMB23	BDMB22	BDMB21	BDMB20	BDMB19	BDMB18	BDMB17	BDMB16
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Value	Description
15 to 0	Break data mask BDMB31–BDMB16	0	Channel B break address BDBn is included in the break conditions (Initial value)
		1	Channel B break address BDBn is masked and therefore not included in the break conditions

n = 31 to 16

Break data mask register BL (BDMRBL)	H'FFFFFF76	16

Item	Bit 15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit Name	BDMB15	BDMB14	BDMB13	BDMB12	BDMB11	BDMB10	BDMB9	BDMB8	BDMB7	BDMB6	BDMB5	BDMB4	BDMB3	BDMB2	BDMB1	BDMB0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Value	Description
15 to 0	Break data mask BDMB15–BDMB0	0	Channel B break address BDBn is included in the break conditions (Initial value)
		1	Channel B break address BDBn is masked and therefore not included in the break conditions

n = 15 to 0

Break bus cycle register B (BBRB)	H'FFFFFF68	16/32

Item	Bit 15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit Name	—	—	—	—	—	—	—	—	CPB1	CPB0	IDB1	IDB0	RWB1	RWB0	SZB1	SZB0
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Value	Description
7, 6	CPU cycle/peripheral cycle select B (CPB1, CPB0)	0 0	No channel B user break interrupt generated (Initial value)
		0 1	Break only on CPU cycles
		1 0	Break only on peripheral cycles
		1 1	Break on both CPU and peripheral cycles
5, 4	Instruction fetch/data access select B (IDB1, IDB0)	0 0	No channel B user break interrupt generated (Initial value)
		0 1	Break only on instruction fetch cyclcs
		1 0	Break only on data access cycles
		1 1	Break on both instruction fetch and data access cycles
3, 2	Read/write select B (RWB1, RWB0)	0 0	No channel B user break interrupt generated (Initial value)
		0 1	Break only on read cycles
		1 0	Break only on write cycles
		1 1	Break on both read and write cycles
1, 0	Operand size select B (SZB1, SZB0)	0 0	Operand size is not a break condition (Initial value)
		0 1	Break on byte access
		1 0	Break on word access
		1 1	Break on longword access

Break control register (BRCR)	H'FFFFFF78	16/32

Item	Bit 15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit Name	CMFCA	CMFPA	EBBE	UMD	—	PCBA	—	—	CMFCB	CMFPB	—	SEQ	DBEB	PCBB	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R	R/W	R	R	R/W	R/W	R	R/W	R/W	R/W	R	R

Bit	Bit Name	Value	Description
15	CPU condition match flag A (CMFCA)	0	Channel A CPU cycle conditions do not match, no user break interrupt generated (Initial value)
		1	Channel A CPU cycle conditions have matched, user break interrupt generated
14	Peripheral condition match flag A (CMFPA)	0	Channel A peripheral cycle conditions do not match, no user break interrupt generated (Initial value)
		1	Channel A peripheral cycle conditions have matched, user break interrupt generated
13	External bus break enable (EBBE)	0	Chip-external bus cycle not included in break conditions (Initial value)
		1	Chip-external bus cycle included in break conditions
12	UBC mode (UMD)	0	Compatible mode for SH7000-series UBCs (Initial value)
		1	SH7604 mode
10	PC break select A (PCBA)	0	Places the channel A instruction fetch cycle break before instruction execution (Initial value)
		1	Places the channel A instruction fetch cycle break after instruction execution
7	CPU condition match flag B (CMFCB)	1	Channel B CPU cycle conditions do not match, no user break interrupt generated (Initial value)
		0	Channel B CPU cycle conditions have matched, user break interrupt generated
6	Peripheral condition match flag B (CMFPB)	0	Channel B peripheral cycle conditions do not match, no user break interrupt generated (Initial value)
		1	Channel B peripheral cycle conditions have matched, user break interrupt generated
4	Sequence condition select (SEQ)	0	Compare channel A and B conditions independently (Initial value)
		1	Compare channel A and B conditions sequentially (channel A, then channel B)
3	Data break enable B (DBEB)	0	Do not include data bus conditions in the channel B conditions (Initial value)
		1	Include data bus conditions in the channel B conditions
2	Instruction break select B (PCBB)	0	Places the channel B instruction fetch cycle break before instruction execution (Initial value)
		1	Places the channel B instruction fetch cycle break after instruction execution

DMA source address registers 0 and 1 (SAR0 and SAR1)	H'FFFFFF80 (channel 0) H'FFFFFF90 (channel 1)	32

	Bit															
Item	**31**	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**
Bit Name																
Initial Value	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Item	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Bit Name																
Initial Value	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Description
31 to 0	(Transfer source address specification)	These bits specify the DMA transfer source address

DMA destination address registers 0 and 1 (DAR0 and DAR1)	H'FFFFFF84 (channel 0) H'FFFFFF94 (channel 1)	32

	Bit															
Item	**31**	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**
Bit Name																
Initial Value	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Item	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Bit Name																
Initial Value	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Description
31 to 0	(Transfer destination address specification)	These bits specify the DMA transfer destination address

DMA transfer count registers 0 and 1 (TCR0 and TCR1)	H'FFFFFF88 (channel 0) H'FFFFFF98 (channel 1)	32

	Bit															
Item	**31**	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**
Bit Name	—	—	—	—	—	—	—	—								
Initial Value	0	0	0	0	0	0	0	0	—	—	—	—	—	—	—	—
R/W	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Item	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Bit Name																
Initial Value	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Description
23 to 0	(Transfer count specification)	Specifies the DMA transfer count (during a DMA transfer, these bits indicate the remaining transfer count)

DMA channel control registers 0, 1 (CHCR0, CHCR1)	H'FFFFFF8C (channel 0) H'FFFFFF9C (channel 1)	32

	Bit															
Item	**31**	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**
Bit Name	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Item	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Bit Name	DM1	DM0	SM1	SM0	TS1	TS0	AR	AM	AL	DS	DL	TB	TA	IE	TE	DE
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/(W)*	R/W

Note: Only 0 can be written, after reading 1, to clear the flag.

Bit	Bit Name	Value		Description
14, 15	Destination address mode bits 1, 0 (DM1, DM0)	0	0	Fixed destination address (Initial value)
		0	1	Destination address is incremented (+1 for byte transfer size, +2 for word transfer size, +4 for longword transfer size, and +16 for 16-byte transfer size)
		1	0	Destination address is decremented (–1 for byte transfer size, –2 for word transfer size, –4 for longword transfer size, and –16 for 16-byte transfer size)
		1	1	Reserved (setting prohibited)

Bit	Bit Name	Value	Description
13, 12	Source address mode bits 1, 0 (SM1, SM0)	0 0	Fixed source address (+16 for 16-byte transfer size) (Initial value)
		0 1	Source address is incremented (+1 for byte transfer size, +2 for word transfer size, +4 for longword transfer size, and +16 for 16-byte transfer size)
		1 0	Source address is decremented (–1 for byte transfer size, –2 for word transfer size, –4 for longword transfer size, and +16 for 16-byte transfer size)
		1 1	Reserved (setting prohibited)
11, 10	Transfer size bits 1, 0 (TS1, TS0)	0	Byte unit (Initial value)
		0	Word (2-byte) unit
		1	Longword (4-byte) unit
		1	16-byte unit (4 longword transfers)
9	Auto-request mode bit (AR)	0	Module request mode (Initial value)
		1	Auto-request mode
8	Acknowledge/transfer mode bit (AM)	0	DACK output in read cycle/transfer from memory to device (Initial value)
		1	DACK output in write cycle/transfer from device to memory
7	Acknowledge level bit (AL)	0	DACK is an active-low signal (Initial value)
		1	DACK is an active-high signal
6	DREQ select bit (DS)	0	Detected by level (Initial value)
		1	Detected by edge
5	DREQ level bit (DL)	0	When DS is 0, DREQ is detected by low level; when DS is 1, DREQ is detected by fall (Initial value)
		1	When DS is 0, DREQ is detected by high level; when DS is 1, DREQ is detected by rise
4	Transfer bus mode bit (TB)	0	Cycle-steal mode
		1	Burst mode
3	Transfer address mode bit (TA)	0	Dnal address mode
		1	Single address mode
2	Interrupt enable bit (IE)	0	Interrupt disabled (Initial value)
		1	Interrupt enabled
1	Transfer-end flag bit (TE)	0	DMA has not ended or was aborted (Initial value) Cleared by reading 1 from the TE bit and then writing 0
		1	DMA has ended nomally (by TCR = 0)
0	DMA enable bit (DE)	0	DMA transfer disabled (Initial value)
		1	DMA transfer enabled

DMA request/response selection control registers 0 and 1 (DRCR0, DRCR1)	H'FFFFFE71 (channel 0) H'FFFFFE72 (channel 1)	8

	Bit							
Item	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Bit Name	—	—	—	—	—	—	RS1	RS0
Initial Value	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Value	Description
1, 0	Resource select bits 1, 0 (RS1, RS0)	0 0	DREQ (external request) (Initial value)
		0 1	RXI (receive-data-full interrupt transfer request of the on-chip serial communication interface (SCI))
		1 0	TXI (transmit-data-full interrupt transfer request of the on-chip SCI)
		1 1	Reserved (setting prohibited)

DMA operation register (DMAOR)	H'FFFFFFB0	32

Item	Bit 31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Bit Name	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	
Item	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Bit Name	—	—	—	—	—	—	—	—	—	—	—	—	PR	AE	NMIF	DME
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/(W)*	R/(W)*	R/W

Note: Only 0 can be written, to clear the flag.

Bit	Bit Name	Value	Description
3	Priority mode bit (PR)	0	Fixed priority (Ch 0 > Ch 1) (Initial value)
		1	Round-robin mode (High priority switches to low after each transfer) (The priority for the first DMA transfer after a reset is Ch 1 > Ch 0)
2	Address error flag bit (AE)	0	No DMAC address error (Initial value)
		1	Address error by DMAC
1	NMI flag bit (NMIF)	0	No NMIF interrupt (Initial value) To clear the NMIF bit, read 1 from it and then write 0
		1	NMIF has occurred
0	DMA master enable bit (DME)	0	DMA transfers disabled on all channels (Initial value)
		1	DMA transfers enabled on all channels

Bus control register 1 (BCR1)	H'FFFFFFE0	16/32

Item	Bit 15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit Name	MASTER	—	—	ENDIAN	BSTROM	PSHR	AHLW1	AHLW0	A1LW1	A1LW0	A0LW1	A0LW0	—	DRAM2	DRAM1	DRAM0
Initial Value	—	0	0	0	0	0	1	1	1	1	1	1	0	0	0	0
R/W	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W

Bit	Bit Name	Value	Description
15	Bus arbitration (MASTER)	0	Master mode
		1	Slave mode
12	Endian specification for area 2 (ENDIAN)	0	Big-endian, as in other areas (Initial value)
		1	Little-endian
11	Area 0 burst ROM enable (BSTROM)	0	Area 0 is accessed normally (Initial value)
		1	Area 0 is accessed as burst ROM
10	Partial space share specification (PSHR)	0	Total master mode when MD5 = 0 (Initial mode)
		1	Partial-share master mode when MD5 = 0
9, 8	Long wait specification for areas 2 and 3 (AHLW1, AHLW0)	0 0	3 waits (Initial value)
		0 1	4 waits
		1 0	5 waits
		1 1	6 waits
7, 6	Long wait specification for area 1 (A1LW1, A1LW0)	0 0	3 waits (Initial value)
		0 1	4 waits
		1 0	5 waits
		1 1	6 waits
5, 4	Long wait specification for area 0 (A0LW1, A0LW0)	0 0	3 waits (Initial value)
		0 1	4 waits
		1 0	5 waits
		1 1	6 waits
2 to 0	Enable for DRAM and other memory (DRAM2–DRAM0)	0 0 0	Areas 2 and 3 are ordinary spaces (Initial value)
		0 0 1	Area 2 is ordinary space; area 3 is synchronous DRAM space
		0 1 0	Area 2 is ordinary space; area 3 is DRAM space
		0 1 1	Area 2 is ordinary space; area 3 is pseudo-SRAM space
		1 0 0	Area 2 is synchronous DRAM space; area 3 is ordinary space
		1 0 1	Areas 2 and 3 are synchronous DRAM spaces
		1 1 0	Reserved (setting prohibited)
		1 1 1	Reserved (setting prohibited)

Bus control register 2 (BCR2)	H'FFFFFFE4	16/32

Item	Bit 15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit Name									A3 SZ1	A3 SZ0	A2 SZ1	A2 SZ0	A1 SZ1	A1 SZ0		
	—	—	—	—	—	—	—	—							—	—
Initial Value	0	0	0	0	0	0	0	0	1	1	1	1	1	1	0	0
R/W	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Bit	Bit Name	Value		Description
7, 6	Bus size specification for area 3 (A3SZ1–A3SZ0)(Valid only when setting ordinary space)	0	0	Reserved (setting prohibited)
		0	1	Byte (8-bit) size
		1	0	Word (16-bit) size
		1	1	Longword (32-bit) size (Initial value)
5, 4	Bus size specification for area 2 (A2SZ1–A2SZ0) (Valid only when setting ordinary space)	0	0	Reserved (setting prohibited)
		0	1	Byte (8-bit) size
		1	0	Word (16-bit) size
		1	1	Longword (32-bit) size (Initial value)
3, 2	Bus size specification for area 1 (A1SZ1–A1SZ0)	0	0	Reserved (setting prohibited)
		0	1	Byte (8-bit) size
		1	0	Word (16-bit) size
		1	1	Longword (32-bit) size (Initial value)

Wait control register (WCR)	H'FFFFFFE8	16/32

Item	Bit 15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit Name	IW31	IW30	IW21	IW20	IW11	IW10	IW01	IW00	W31	W30	W21	W20	W11	W10	W01	W00
Initial Value	1	0	1	0	1	0	1	0	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Value		Description
15 to 8	Idles between cycles for areas 3 to 0 (IW31–IW00)	IW31 IW30 IW21 IW20 IW11 IW10 IW01 IW00		
		0	0	No idle cycle
		0	1	One idle cycle inserted
		1	0	Two idle cycles inserted (Initial value)
		1	1	Reserved (setting prohibited)

Bit	Bit Name	Value	Description
7 to 0	Wait control of areas 3 to 0 (W31–W00)	During basic cycle	
		W31 W30 W21 W20 W11 W10 W01 W00	
		0 0	External wait input disabled without waits
		0 1	External wait input enabled with one wait
		1 0	External wait input enabled with two waits
		1 1	Complies with the long wait specification of bus control register 1 (BCR1) External wait input is enabled (Initial value)
		When area 3 is DRAM	
		W31 W30	
		0 0	1 CAS assert cycle
		0 1	2 CAS assert cycles
		1 0	3 CAS assert cycles
		1 1	Reserved (setting prohibited)
		When area 2 or 3 is synchronous DRAM	
		W31 W30 W21 W20	
		0 0	1 CAS latency cycle
		0 1	2 CAS latency cycles
		1 0	3 CAS latency cycles
		1 1	4 CAS latency cycles (Initial value)
		When area 3 is pseudo-SRAM	
		W31 W30	
		0 0	2 cycles from BS signal assertion to end of cycle
		0 1	3 cycles from BS signal assertion to end of cycle
		1 0	4 cycles from BS signal assertion to end of cycle
		1 1	Reserved (setting prohibited)

Individual memory control register (MCR)	H'FFFFFFEC	16/32

	Bit															
Item	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit Name	TRP	RCD	TRWL	TRAS 1	TRAS 0	BE	RASD	—	AMX2	SZ	AMX1	AMX0	RFSH	RMD	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Bit	Bit Name	Value	Description
15	RAS precharge time (TRP)	0	1 cycle (Initial value)
		1	2 cycles
14	RAS-CAS delay (RCD)	0	1 cycle (Initial value)
		1	2 cycles
13	Write-precharge delay (TRWL)	0	1 cycle (Initial value)
		1	2 cycles
12, 11	CAS-before-RAS refresh RAS assert time (TRAS1, TRAS0)	0 0	2 cycles (Initial value)
		0 1	3 cycles
		1 0	4 cycles
		1 1	Reserved (setting prohibited)
10	Burst enable (BE)	0	Burst disabled (Initial value)
		1	High-speed page mode during DRAM interface is enabled. Data is continuously transferred in static column mode during pseudo-SRAM interfacing. During synchronous DRAM access, burst is always enabled regardless of this bit.
9	Bank active mode (RASD)	0	For synchronous DRAM, read or write is performed using auto-precharge mode. The next access always starts with a bank active command.
		1	For synchronous DRAM, access ends with bank active status. This is only valid for area 3. When area 2 is synchronous DRAM, the mode is always auto-precharge.

Bit	Bit Name	Value	Description
7, 5, 4	Address multiplex (AMX2–AMX0)	For DRAM interface	
		0 0 0	8-bit column address DRAM (Initial value)
		0 0 1	9-bit column address DRAM
		0 1 0	10-bit column address DRAM
		0 1 1	11-bit column address DRAM
		1 0 0	Reserved (setting prohibited)
		1 0 1	Reserved (setting prohibited)
		1 1 0	Reserved (setting prohibited)
		1 1 1	Reserved (setting prohibited)
		For synchronous DRAM interface	
		0 0 0	16-Mbit DRAM (1M × 16 bits) (Initial value)
		0 0 1	16-Mbit DRAM (2M × 8 bits)
		0 1 0	16-Mbit DRAM (4M × 4 bits)
		0 1 1	4-Mbit DRAM (256k × 16 bits)
		1 0 0	Reserved (setting prohibited)
		1 0 1	Reserved (setting prohibited)
		1 1 0	Reserved (setting prohibited)
		1 1 1	2-Mbit DRAM (128k × 16 bits)
6	Memory data size (SZ)	0	Word (Initial value)
		1	Longword
3	Refresh control (RFSH)	0	No refresh (Initial value)
		1	Refresh
2	Refresh mode (RMODE)	0	Normal refresh (Initial value)
		1	Self-refresh

Refresh timer control/status register (RTCSR)	H'FFFFFFF0	16/32

Item	Bit 15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit Name	—	—	—	—	—	—	—	—	CMF	CMIE	CKS2	CKS1	CKS0	—	—	—
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R	R	R

Bit	Bit Name	Value	Description
7	Compare match flag (CMF)	—	RTCNT and RTCOR match Clear condition: After RTCSR is read when CMF is 1, 0 is written in CMF
6	Compare match interrupt enable (CMIE)	0	Disables interrupt request caused by CMF (Initial value)
		1	Enables interrupt request caused by CMF
5 to 3	Clock select bits (CKS2–CKS0)	0 0 0	Disables count up (Initial value)
		0 0 1	CLK/4
		0 1 0	CLK/16
		0 1 1	CLK/64
		1 0 0	CLK/256
		1 0 1	CLK/1024
		1 1 0	CLK/2048
		1 1 1	CLK/4096

Refresh timer counter (RTCNT)	H'FFFFFFF4	16/32

Item	Bit 15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit Name	—	—	—	—	—	—	—	—								
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Description
7 to 0	(Count value)	Input clock count value

Refresh time constant register (RTCOR)	H'FFFFFFF8	16/32

Item	Bit 15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit Name	—	—	—	—	—	—	—	—								
Initial Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Description
7 to 0	(Timer constant)	Sets the refresh cycle

Cache control register (CCR)	H'FFFFFE92	8

	Bit							
Item	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Bit Name	W1	W0	—	CP	TW	OD	ID	CE
Initial Value	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Value	Description
7, 6	Way specification (W1, W0)	0 0	Way 0 (Initial value)
		0 1	Way 1
		1 0	Way 2
		1 1	Way 3
4	Cache purge (CP)	0	Normal operation (Initial value)
		1	Cache purge
3	Two-way mode (TW)	0	Four-way mode (Initial value)
		1	Two-way mode
2	Data replacement disable (OD)	0	Normal operation (Initial value)
		1	Data not replaced even when cache miss occurs in data access
1	Instruction replacement disable (ID)	0	Normal operation (Initial value)
		1	Data not replaced even when cache miss occurs in instruction fetch
0	Cache enable (CE)	0	Cache disabled (Initial value)
		1	Cache enabled

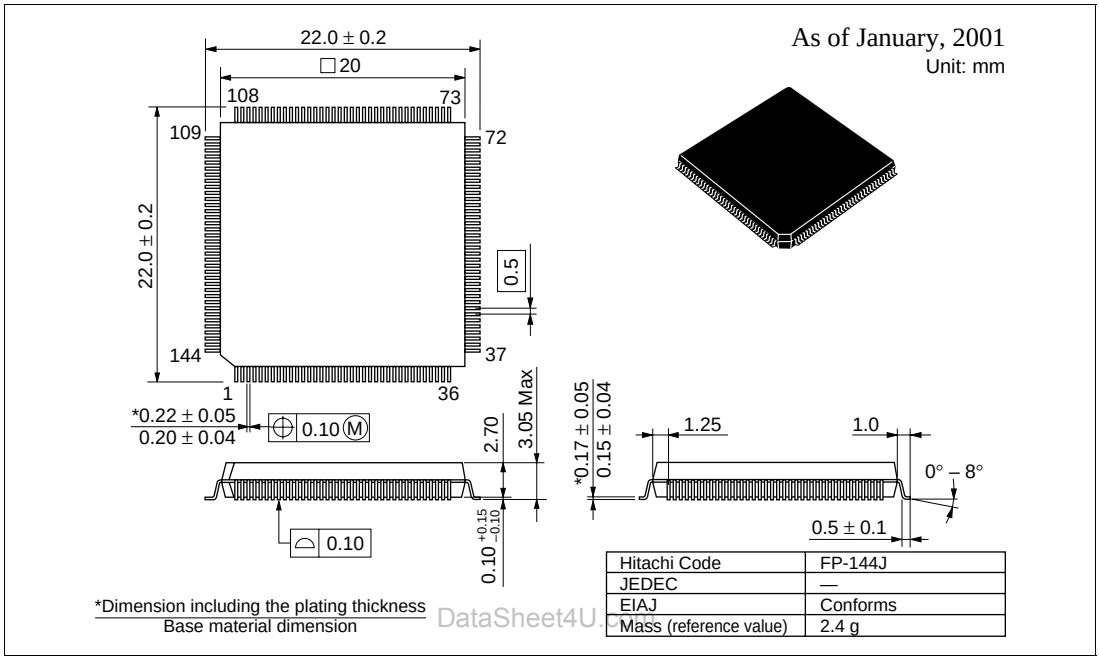
Standby control register (SBYCR)	H'FFFFFE91	8

Item	Bit 7	6	5	4	3	2	1	0
Bit Name	SBY	HIZ	—	MSTP4	MSTP3	MSTP2	MSTP1	MSTP0
Initial Value	0	0	0	0	0	0	0	0
R/W	R/W	R/W	—	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Value	Description
7	Standby (SBY)	0	Executing SLEEP instruction puts the chip into sleep mode (Initial value)
		1	Executing SLEEP instruction puts the chip into standby mode
6	Port high impedance (HIZ)	0	Pin states held in standby mode (Initial value)
		1	Pins at high impedance in standby mode
4	Module stop 4 (MSTP4)	0	DMAC running (Initial value)
		1	Clock supply to DMAC halted
3	Module stop 3 (MSTP3)	0	MULT running (Initial value)
		1	Clock supply to MULT halted
2	Module stop 2 (MSTP2)	0	DIVU running (Initial value)
		1	Clock supply to DIVU halted
1	Module stop 1 (MSTP1)	0	FRT running (Initial value)
		1	Clock supply to FRT halted
0	Module stop 0 (MSTP0)	0	SCI running (Initial value)
		1	Clock supply to SCI halted

Appendix C External Dimensions

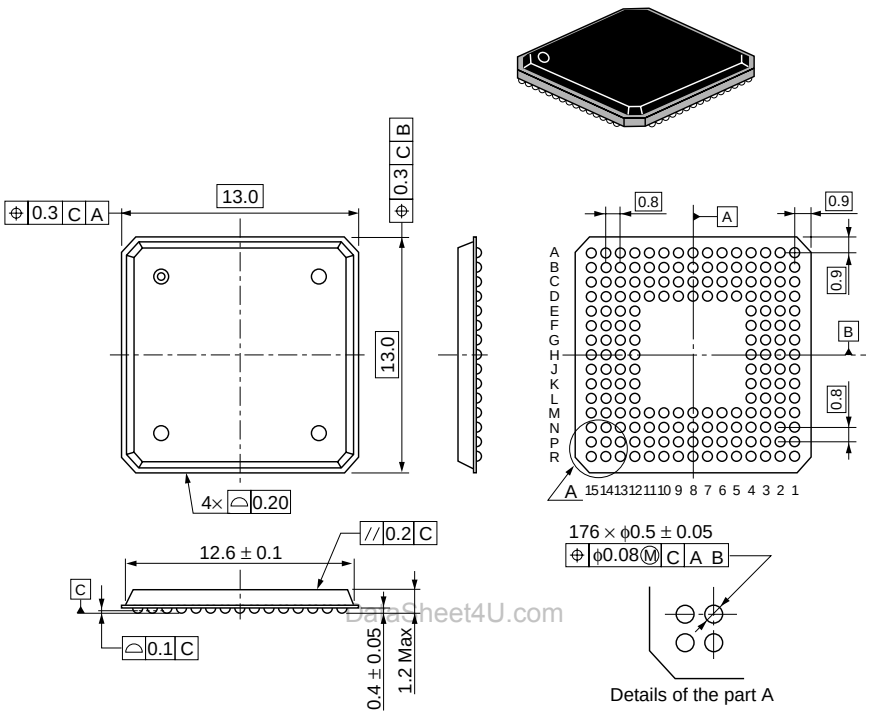
Figure C.1 shows the external dimensions of the SH7604 (FP144J).



Hitachi Code	FP-144J
JEDEC	—
EIAJ	Conforms
Mass (reference value)	2.4 g

Figure C.1 External Dimensions

Figure C.2 shows the external dimensions of the SH7604 (TBP-176).



Hitachi Code	TBP-176
JEDEC	—
EIAJ	—
Mass (reference value)	0.32 g

Figure C.2 External Dimensions

SH7604 Hardware Manual
Publication Date: 1st Edition, March 1995
4th Edition, September 2001
Published by: Customer Service Division
Hitachi, Ltd.
Edited by: Technical Documentation Center
Hitachi Kodaira Semiconductor Co., Ltd.