1 External Memory Controller

1.1 Overview

The External Memory Controller (EMC) divides the off-chip memory space and outputs control signals complying with specifications of various types of memory and bus interfaces. It enables the connection of static memory, NAND flash memory, synchronous DRAM, etc., to this processor.

- Static memory interface
  - Direct interface to ROM, Burst ROM, SRAM and NOR Flash.
  - Support 4 external chip selection CS4~1#. Each bank can be configured separately.
  - The size and base address of static memory banks are programmable.
  - Output of control signals allowing direct connection of memory to each bank. Write strobe setup time and hold time periods can be inserted in an access cycle to enable connection to low-speed memory
  - Wait state insertion can be controlled by program.
  - Wait insertion by WAIT pin.
  - Automatic wait cycle insertion to prevent data bus collisions in case of consecutive memory accesses to different banks, or a read access followed by a write access to the same bank

- NAND flash interface
  - Support on CS4~CS1, sharing with static memory bank4~bank1.
  - Support most types of NAND flashes, including 8-bit and 16-bit bus width, 512B and 2KB page size. For 512B page size, 3 and 4 address cycles are supported. For 2KB page size, 4 and 5 address cycles are supported.
  - Hardware ECC generation including Hamming and RS codes correction.
  - Support read/erase/program NAND flash memory.
  - Support boot from NAND flash.

- SDRAM Interface
  - Support 1 chip selection DCS#.
  - Support both 32-bit and 16-bit bus width.
  - Support both two-bank and four-bank type SDRAM.
  - Support burst operation.
  - Support both auto-refresh and self-refresh functions.
  - The size and base address of each bank is configurable.
  - Multiplexes row/column addresses according to SDRAM capacity
  - Controls timing of SDRAM direct-connection control signals according to register setting
  - Supports power-down mode to minimize the power consumption of SDRAM
  - Support page mode

1.2 Pin Description

Following table list the EMC pins.

Table 1-1 EMC Pin Description

Pin Name	I/O	Signal	Description
Data Bus	I/O	D31 – D0	Data I/O
Address bus	O	A22–A0	Address output
Static chip select 4 ~ 1	O	CS4~1#	Chip select signal that indicates the static bank being accessed
SDRAM chip select	O	DCS#	Chip select signal that indicates the SDRAM bank being accessed
Read enable	O	RD# /	For Static memory read enable signal
Write enable	O	WE# /	Static memory write enable signal
Column address strobe	O	CAS#	SDRAM column address strobe signal
Row address strobe	O	RAS#	SDRAM row address strobe signal
Read/write	O	RD/WR#	Data bus direction designation signal Also used as SDRAM write enable signal
Byte enable 0	O	WE0# / BE0# / DQM0 /	For non-byte-control static memory , D7-0 write enable signal, For byte-control static memory , D7-0 selection signal For SDRAM, D7–D0 selection signal
Byte enable 1	O	WE1# / BE1# / DQM1/	For non-byte-control static memory , D15-8 write enable signal For byte-control static memory , D15-8 selection signal For SDRAM, D15–D8 selection signal
Byte enable 2	O	WE2# / BE2# / DQM2 /	For non-byte-control static memory, D23-16 write enable signal For byte-control static memory, D23-16 selection signal For SDRAM , D23–D16 selection signal
Byte enable 3	O	WE3# / BE3# / DQM3	For static memory , D31-24 write enable signal For byte-control static memory , D31-24 selection signal For SDRAM, D31–D24 selection signal.
SDRAM Clock enable	O	CKE	Enable the SDRAM clock
Wait	I	Wait# /	External wait state request signal for memory-like devices
NAND flash read enable	O	FRE#	NAND flash read enable signal
NAND flash write enable	O	FWE#	NAND flash write enable signal
NAND flash ready/busy	I	FRB#	Indicates NAND flash is ready or busy (When Nand flash boot, GPC30 is used as FRB# of CS1#)

1.3 Physical Address Space Map

Both virtual spaces and physical spaces are 32-bit wide in this architecture. Virtual addresses are translated by MMU into physical address which is further divided into several partitions for static memory, SDRAM, and internal I/O.

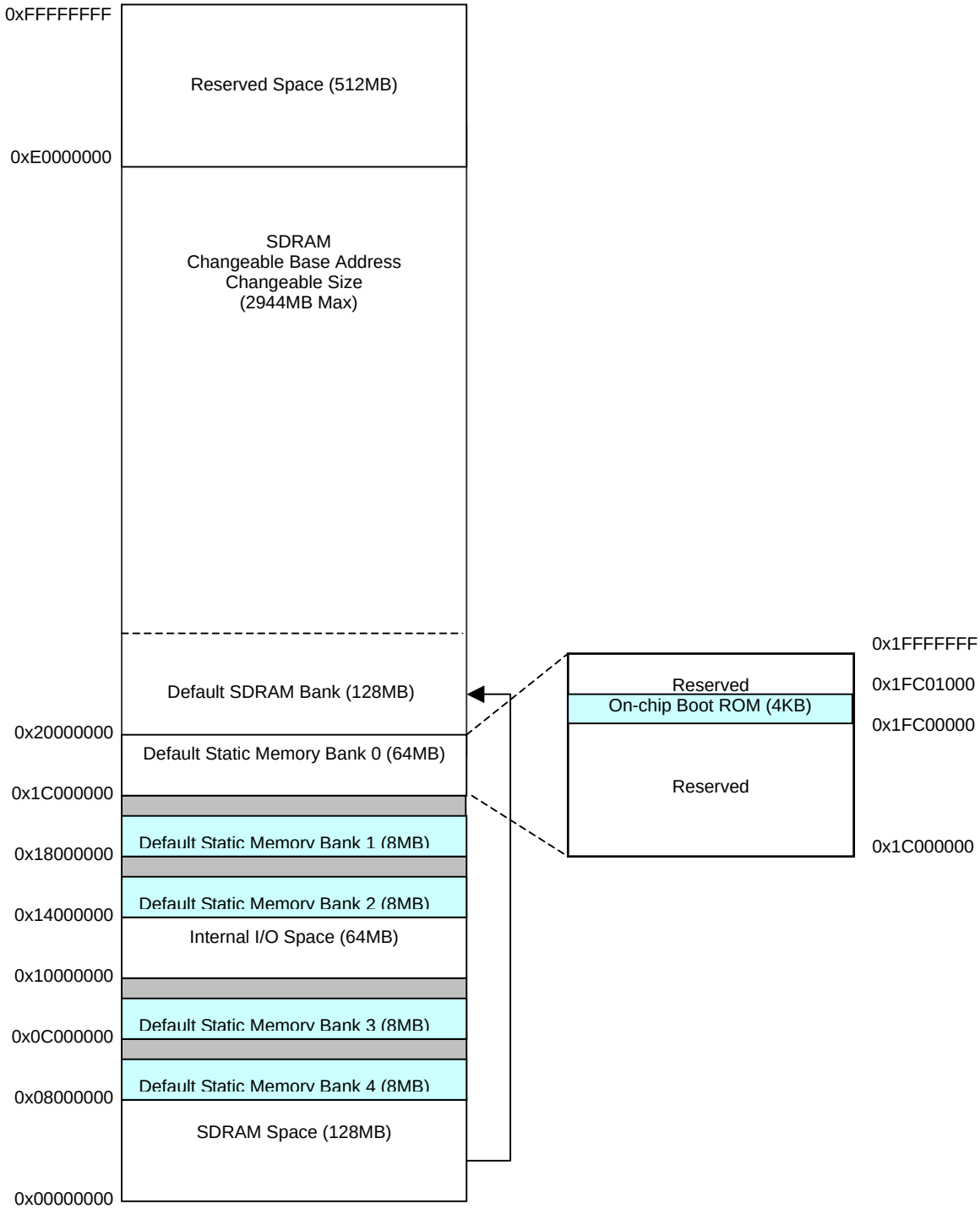


Figure 1-1 Physical Address Space Map

Table 1-2 Physical Address Space Map

Start Address	End Address	Connectable Memory	Capacity
H'0000 0000	H'07FF FFF	SDRAM space	128 MB
H'0800 0000	H'0FFF FFFF	Static memory space	128 MB
H'1000 0000	H'13FF FFFF	Internal I/O space	64 MB
H'1400 0000	H'1BFF FFFF	Static memory space	128MB
H'1C00 0000	H'1FBF FFFF	Un-used	60MB
H'1FC0 0000	H'1FC0 0FFF	On-chip boot ROM	4KB
H'1FC0 1000	H'1FFF FFFF	Un-used	4095KB
H'2000 0000	H'BFFF FFFF	SDRAM space	2944 MB
H'D000 0000	H'FFFF FFFF	Reserved space	512 MB

The base address and size of each memory banks are configurable. Software can re-configure these memory banks according to the actual connected memories. Following table lists the default configuration after reset.

Table 1-3 Default Configuration of EMC Chip Select Signals

Chip-Select Signal	Connected Memory	Capacity	Memory Width *1	Start Address	End Address
CS1#	Static memory bank 1	8 MB	8, 16, 32	H'1800 0000	H'1BFF FFFF
CS2#	Static memory bank 2	8 MB	8, 16, 32	H'1400 0000	H'17FFFFFF
CS3#	Static memory bank 3	8 MB	8, 16, 32	H'0C00 0000	H'0FFF FFFF
CS4#	Static memory bank 4	8 MB	8, 16, 32	H'0800 0000	H'0BFF FFFF
DCS#*3	SDRAM bank	128 MB	16, 32	H'2000 0000	H'27FF FFFF

Notes:

1. Data width of static memory banks can be configured to 8, 16 or 32 bits by software.

2. The 4KB address space from H'1FC00000 to H'1FC00FFF in bank 0 is mapped to on-chip boot ROM. The other memory spaces in bank 0 are not used.

3. To support large SDRAM space, EMC re-maps the physical address H'00000000-H07FFFFFF to H'20000000-H'27FFFFFF. Software must configure the SDRAM base address by the re-mapped address.

1.4 Static Memory Interface

The static memory controller provides a glueless interface to SRAM's, ROMs (PROMs/EPROMs/FLASH), dual port memory, IO devices, and many other peripherals devices. It can directly control up to 4 devices using four chip select lines. Additional devices may be supported through external decoding of the address bus. The Device Controller shares the data and address busses with the SDRAM controller. Thus, only one memory subsection (SDRAM, memory, or IO) can be active at any time.

Each chip select can directly access memory or IO devices that are 8-bits, 16-bits, or 32-bits wide. Each device connected to a chip select line has 2 associated registers that control its operation and the access timing to the external device. The Static Memory Control Register SMCRn specifies various configurations for the device. The Static Memory Address Configuration Register SACRn specifies the base address and size for each device, enabling any device to be located anywhere in the physical address range.

The static memory interface includes the following signals:

- Four chip selects, CS4~1#
- Twenty-three address signals, A22-A0
- One read enable, RD#
- One write enable, WE#
- Four byte enable, BE3~1#
- One wait pin, WAIT#

The SMT field in SMCRn registers specifies the type of memory and BW field specifies the bus width. BOOT_SEL[1:0] defines whether system boot from Nor or Nand flash and the page size when boot from Nand flash.

1.4.1 Register Description

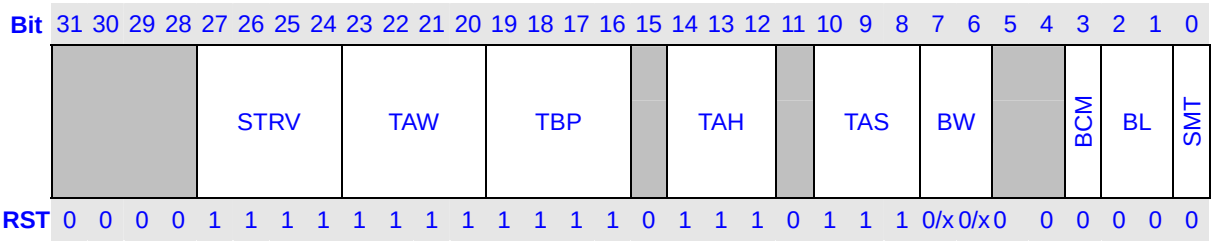
Table 1-4 Static Memory Interface Registers

Name	Description	RW	Reset Value	Address	Access Width
SMCR1	Static memory control register 1	RW	0x0FFF7700	0x13010014	32
SMCR2	Static memory control register 2	RW	0x0FFF7700	0x13010018	32
SMCR3	Static memory control register 3	RW	0x0FFF7700	0x1301001C	32
SMCR4	Static memory control register 4	RW	0x0FFF7700	0x13010020	32
SACR1	Static memory bank 1 address configuration register	RW	0x000018FC	0x13010034	32
SACR2	Static memory bank 2 address configuration register	RW	0x000016FE	0x13010038	32
SACR3	Static memory bank 3 address configuration register	RW	0x000014FE	0x1301003C	32
SACR4	Static memory bank 4 address configuration register	RW	0x00000CFC	0x13010040	32

1.4.1.1 Static Memory Control Register (SMCR1~4)

SMCR1~4 are 32-bit read/write registers that contain control bits for static memory. On reset, SMCR1~4 are initialized to 0x0FFF7700.

SMCR1 **0x13010014**
SMCR2 **0x13010018**
SMCR3 **0x1301001C**
SMCR4 **0x13010020**



Bits	Name	Description	RW
31:28	Reserved	Writes to these bits have no effect and always read as 0.	R
27:24	STRV	**Static Memory Recovery Time:** Its value is the number of idle cycles (0~15 cycles) inserted between bus cycles when switching from one bank to another bank or between a read access to a write access in the same	RW

		bank. Its initial value is 0xF (15 cycles).	
23:20	TAW	**Access Wait Time:** For normal memory, these bits specify the number of wait cycles to be inserted in read strobe time. For burst ROM, these bits specify the number of wait cycles to be inserted in first data read strobe time. **TAW3~0 Wait cycle Wait# Pin** 0000 0 cycle Ignored 0001 1 cycle Enabled 0010 2 cycles Enabled 0011 3 cycles Enabled 0100 4 cycles Enabled 0101 5 cycles Enabled 0110 6 cycles Enabled 0111 7 cycles Enabled 1000 8 cycles Enabled 1001 9 cycles Enabled 1010 10 cycles Enabled 1011 12 cycles Enabled 1100 15 cycles Enabled 1101 20 cycles Enabled 1110 25 cycles Enabled 1111 31 cycles Enabled (Initial Value)	RW
19:16	TBP	**Burst Pitch Time:** For burst ROM, these bits specify the number of wait cycles to be inserted in subsequent access. For normal memory, these bits specify the number of wait cycles to be inserted in write strobe time. **TBP3~0 Wait cycle Wait# Pin** 0000 0 cycle Ignord 0001 1 cycle Enabled 0010 2 cycles Enabled 0011 3 cycles Enabled 0100 4 cycles Enabled 0101 5 cycles Enabled 0110 6 cycles Enabled 0111 7 cycles Enabled 1000 8 cycles Enabled 1001 9 cycles Enabled 1010 10 cycles Enabled 1011 12 cycles Enabled 1100 15 cycles Enabled 1101 20 cycles Enabled 1110 25 cycles Enabled 1111 31 cycles Enabled (Initial Value)	RW
15	Reserved	Writes to these bits have no effect and always read as 0.	R

14:12	TAH	**Address Hold Time:** These bits specify the number of wait cycles to be inserted from negation of read/write strobe to address. **TAH2~0 Wait cycle** 000 0 cycle 001 1 cycle 010 2 cycles 011 3 cycles 100 4 cycles 101 5 cycles 110 6 cycles 111 7 cycles (Initial Value)	RW
11	Reserved	Writes to these bits have no effect and always read as 0.	R
10:8	TAS	**Address Setup Time:** These bits specify the number of wait cycles (0~7 cycles) to be inserted from address to assertion of read/write strobe. **TAS2~0 Wait cycle** 000 0 cycle 001 1 cycle 010 2 cycles 011 3 cycles 100 4 cycles 101 5 cycles 110 6 cycles 111 7 cycles (Initial Value)	RW
7:6	BW	**Bus Width :** These bits specify the bus width. this filed is writeable and are initialized to 0 by a reset. **BW1~0 Bus Width** 00 8 bits (Initial Value) 01 16 bits 10 32 bits 11 Reserved	RW
5:4	Reserved	Writes to these bits have no effect and always read as 0.	R
3	BCM	**SRAM Byte Control Mode (BCM):** When SRAM is connected; this bit specifies the type of SRAM. This bit is only valid when SMT is set to 0. **BCM Description** 0 SRAM is set to normal mode (Initial Value) 1 SRAM is set to byte control mode	RW
2:1	BL	**Burst Length (BL1, BL0):** When Burst ROM is connected; these bits specify the number of burst in an access. These bits are only valid when SMT is set to 1. **BL1~0 Burst Length** 00 4 consecutive accesses. Can be used with 8-, 16-, or 32-bit bus width (Initial Value). 01 8 consecutive accesses. Can be used with 8-, 16-, or	

		32-bit bus width 10 16 consecutive accesses. Can only be used with 8- or 16-bit bus width. Do not specify for 32-bit bus width 11 32 consecutive accesses. Can only be used with 8-bit bus width	
0	SMT	**Static Memory Type (SMT):** This bit specifies the type of static memory. **SMT** **Description** 0 Normal Memory (Initial Value) 1 Burst ROM	RW

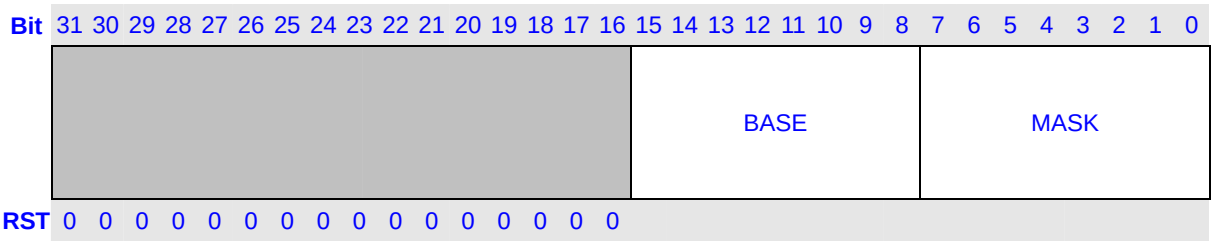
1.4.1.2 Static Bank Address Configuration Register (SACR1~4)

SACR1~4 defines the physical address for static memory bank 1 to 4, respectively. Each register contains a base address and a mask. When the following equation is met:

(*physical_address* [31:24] & **MASK$_n$**) == **BASE$_n$**

The bank n is active. The *physical_address* is address output on internal system bus. Static bank regions must be programmed so that each bank occupies a unique area of the physical address space. Bank 0 base address must be 0 because it's system boot address. Programming overlapping bank regions will result in unpredictable error. These registers are initialized by a reset.

SACR1 **0x13010034**
SACR2 **0x13010038**
SACR3 **0x1301003C0x13010040**
SACR4



Bits	Name	Description	RW
31:16	Reserved	Writes to these bits have no effect and read always as 0.	R
15:8	BASE	**Address Base:** Defines the base address of Static Bank n (n = 1 to 4). The initial values are: SACR1.BASE 0x18 SACR2.BASE 0x14 SACR3.BASE 0x0C SACR4.BASE 0x08	RW
23:20	MASK	**Address Mask**: Defines the mask of Static Bank n (n = 1 to 4). The initial values are: SACR1.MASK 0xFC SACR2.MASK 0xFC SACR3.MASK 0xFC SACR4.MASK 0xFC	RW

1.4.2 Example of Connection

Following figures shows examples of connection to 32-, 16- and 8-bit data width normal memory.

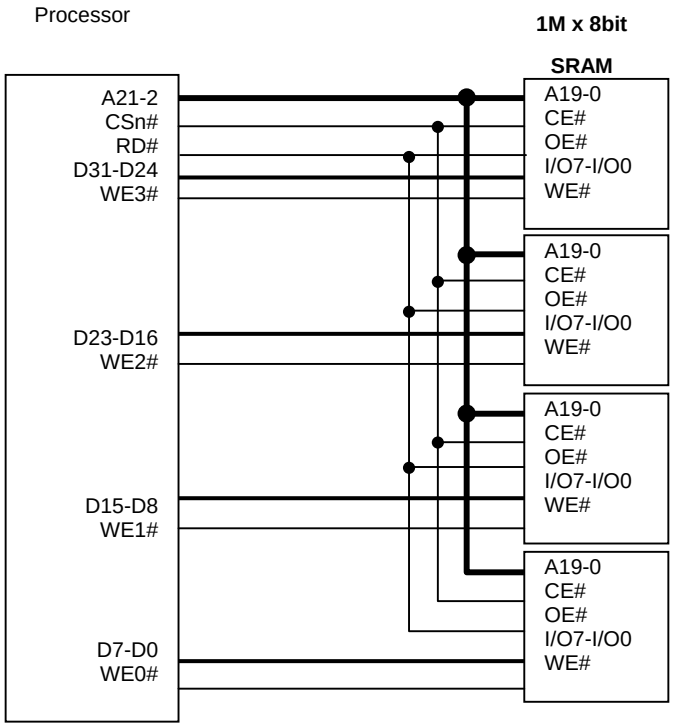


Figure 1-2 Example of 32-Bit Data Width SRAM Connection

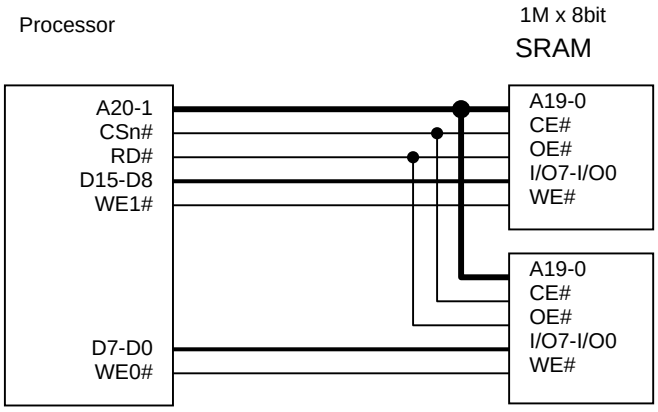


Figure 1-3 Example of 16-Bit Data Width SRAM Connection

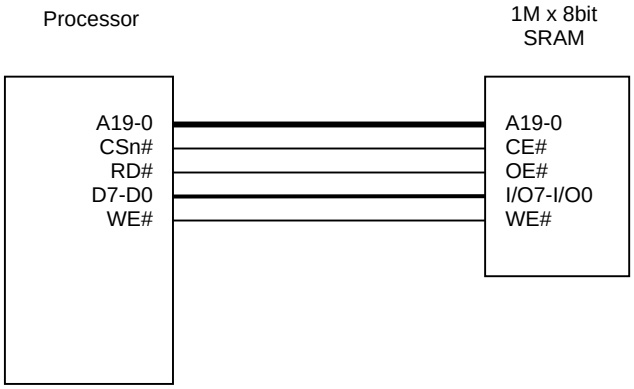


Figure 1-4 Example of 8-Bit Data Width SRAM Connection

1.4.3 Basic Interface

When SMT field in SMCRn (n = 1 to 4) is 0 and BCM field is 0, normal memory (non-burst ROM, Flash, normal SRAM or memory-like device) is connected to bank n. When bank n (n = 1 to 4) is accessed, CSn# is asserted as soon as address is output. In addition, the RD# signal, which can be used as OE#, and write control signals, WE0# to WE3#, are asserted.

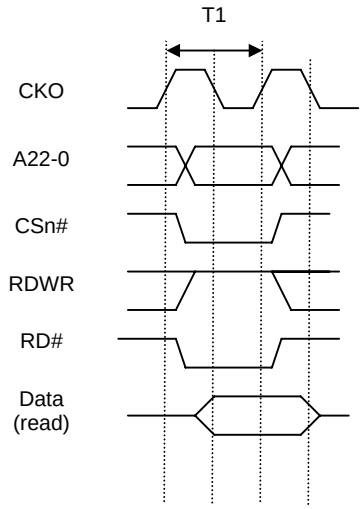
The TAS field in SMCRn is the latency from CSn# to read/write strobe. The TAW3 field is the delay time of RD# in read access. TBP3~0 field is the delay time of WE# and WEn# in write access. In addition, any number of waits can be inserted by means of the external pin (WAIT#). The TAH field is the latency from RD# and WEn# negation to CSn# negation, also the hold time to address and write data.

All kinds of normal memories (non-burst ROM, normal SRAM and Flash) have the same read and write timing. There are some requirements for writes to flash memory. Flash memory space must be un-cacheable and un-buffered. Writes must be exactly the width of the populated Flash devices on the data bus (no byte writes to a 32-bit bus or word writes to a 16-bit bus, and so on). Software is responsible for partitioning commands and data, and writing them out to Flash in the appropriate sequence.

Glossary

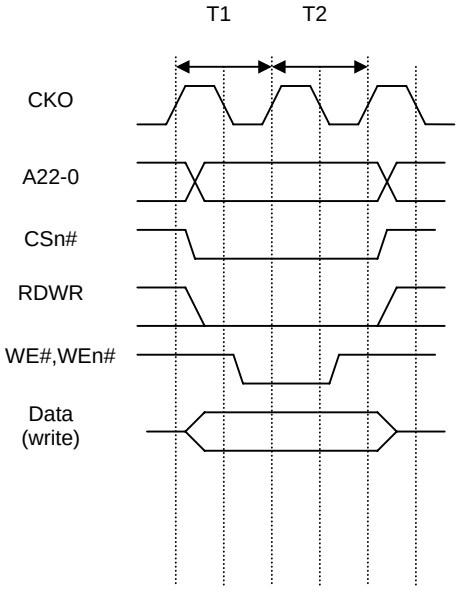
Th – hold cycle
Tw – wait cycle
Ts – setup cycle
T1 – read inherent cycle or first write inherent cycle
T2 – last write inherent cycle
Tb – burst read inherent cycle

Following figures show the timing of normal memory. A no-wait read access is completed in one cycle and a no-wait write access is completed in two cycles. Therefore, there is no negation period in case of access at minimum pitch.



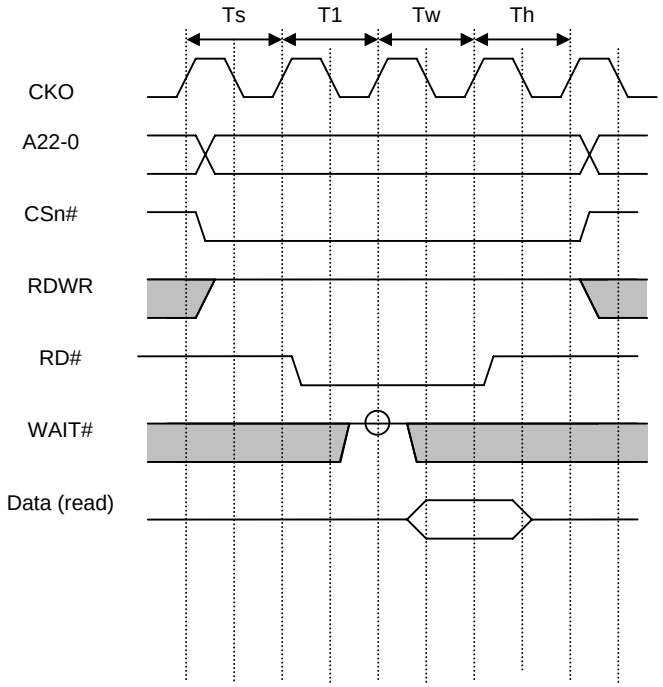
*In this example, SMCRn:MT = 0, BCM = 0, TAS = 0, TAW = 0, TAH = 0

Figure 1-5 Basic Timing of Normal Memory Read



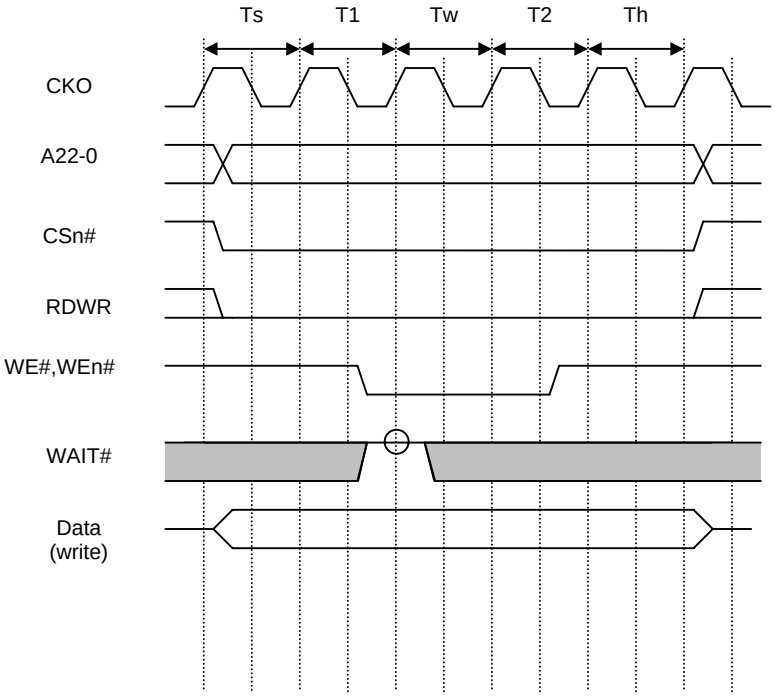
*In this example, SMCRn: SMT = 0, BCM = 0, TAS = 0, TBP = 0, TAH = 0

Figure 1-6 Basic Timing of Normal Memory Write



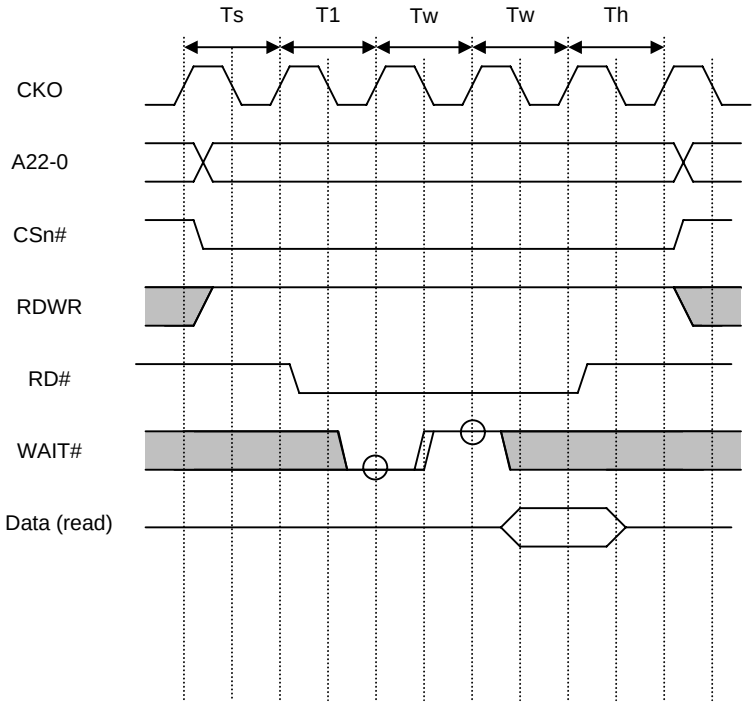
*In this example, SMCRn: SMT = 0, BCM = 0, TAS = 1, TAW = 1, TAH = 1

Figure 1-7 Normal Memory Read Timing With Wait (Software Wait Only)



*In this example, SMCRn: SMT = 0, BCM = 0, TAS = 1, TBP = 1, TAH = 1

Figure 1-8 Normal Memory Write Timing With Wait (Software Wait Only)



*In this example, SMCRn: SMT = 0, BCM = 0, TAS = 1, TAW = 1, TAH=1

Figure 1-9 Normal Memory Read Timing With Wait (Wait Cycle Insertion by WAIT# pin)

1.4.4 Byte Control

The byte control SRAM interface is a memory interface that outputs a byte select strobe WEn# in both read and write bus cycles. It has 16 bit data pins, and can be directly connected to SRAM which has an upper byte select strobe and lower byte select strobe function such as UB# and LB#.

In read/write access, RD#/WE# is used as read/write strobe signal and WEn# are used as byte select signals.

Following figure shows an example of byte control SRAM connection to processor.

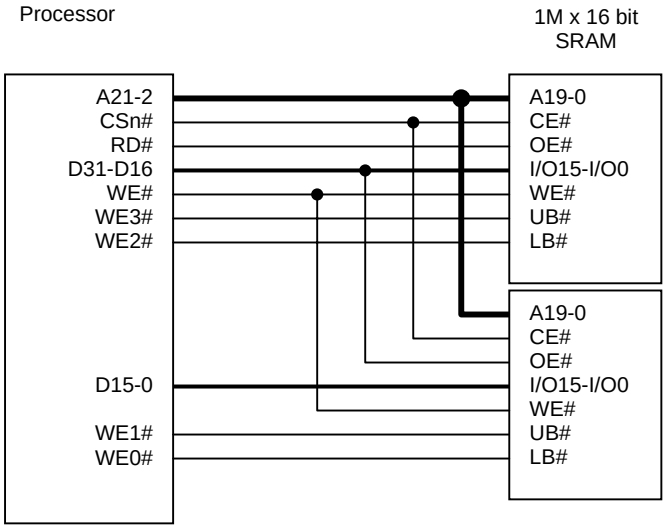
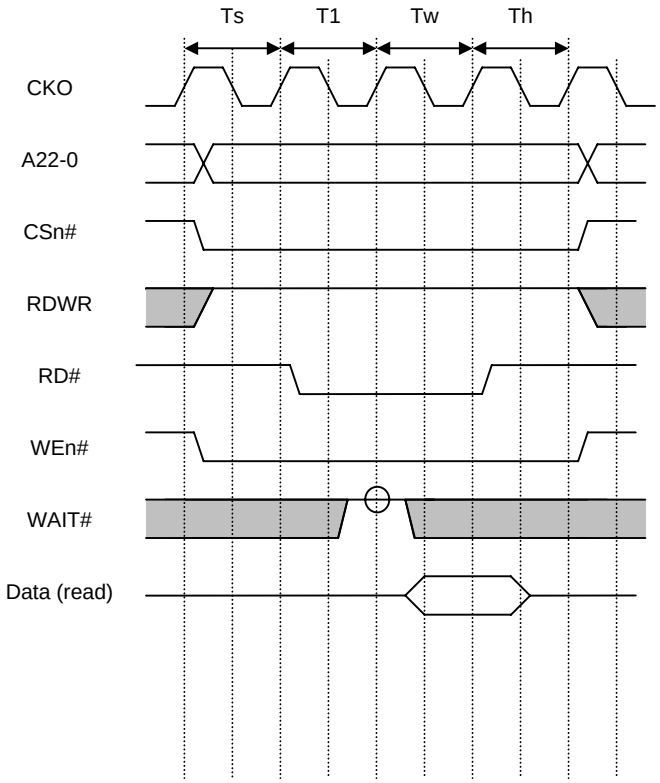


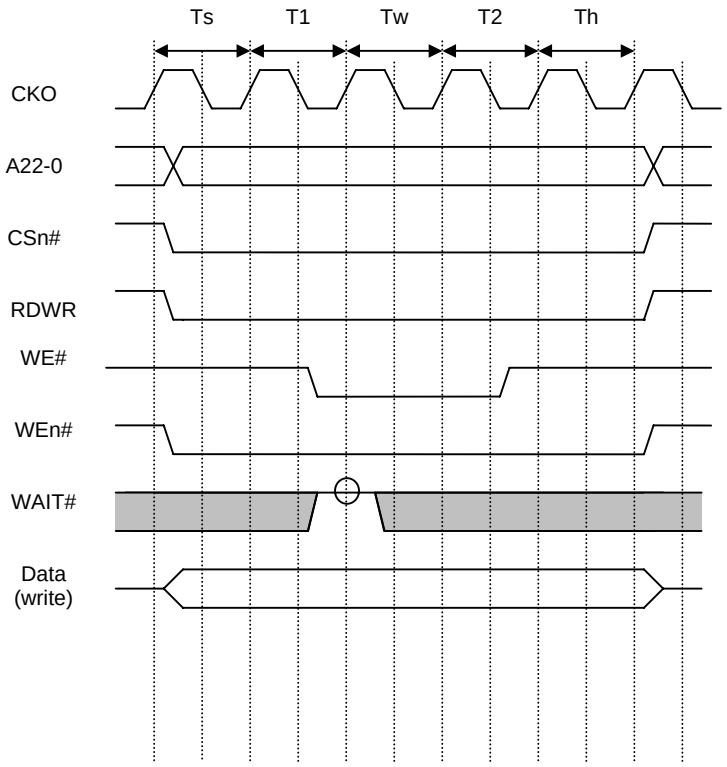
Figure 1-10 Example of 32-Bit Data Width Byte Control SRAM Connection

Following figures show examples of Byte Control SRAM timing.



*In this example, SMCRn: SMT = 0, BCM = 1, TAS = 1, TAW = 1, TAH = 1

Figure 1-11 Byte Control SRAM Read Timing



*In this example, SMCRn: SMT = 0, BCM = 1, TAS = 1, TBP = 1, TAH = 1

Figure 1-12 Byte Control SRAM Write Timing

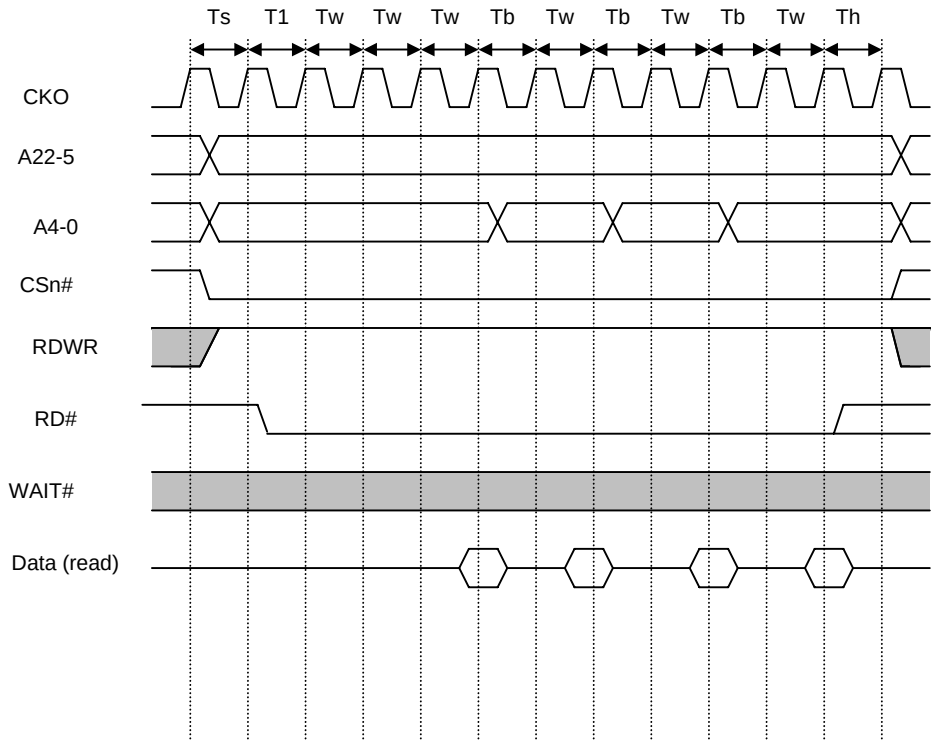
1.4.5 Burst ROM Interface

Setting SMT to 1 in SMCRn allows burst ROM to be connected to bank n (n = 1 to 4). The burst ROM interface provides high-speed access to ROM that has a nibble access function. Basically, access is performed in the same way as for normal memory, but when the first cycle ends, only the address is changed before the next access is executed. When 8-bit burst ROM is connected, the number of consecutive accesses can be set as 4, 8, 16, or 32 with bits BL1~0. When 16-bit ROM is connected, 4, 8, or 16 can be set in the same way. When 32-bit ROM is connected, 4 or 8 can be set.

For burst ROM read, TAW sets the delay time from read strobe to the first data, TBP sets the delay time from consecutive address to data. Burst ROM writes have the same timing as normal memory except TAW instead of TBP is used to set the delay time of write strobe.

WAIT# pin sampling is always performed when one or more wait states are set.

Following figures show the timing of burst ROM.



*In this example, SMT = 1, BL = 0, TAS = 1, TAW = 3,TBP = 1,TAH = 1

Figure 1-13 Burst ROM Read Timing (Software Wait Only)

1.5 NAND Flash Interface

NAND flash can be connected to static memory bank 4~ band 1. Both 8-bit and 16-bit NAND flashes are supported. Hardware ECC generator is implemented (including Hamming and RS codes correction). A mechanism for booting from NAND flash is also supported.

1.5.1 Register Description

Table 1-5 NAND Flash Interface Registers

Name	Description	RW	Reset Value	Address	Access Width
NFCSR	NAND flash control/status register	RW	0x00000000	0x13010050	32
NFECCR	NAND flash ECC control register	RW	0x00000000	0x13010100	32
NFECC	NAND flash ECC data register	R	Undefined	0x13010104	32
NFPAR0	NAND flash RS Parity 0 register	RW	0x00000000	0x13010108	32/16/8
NFPAR1	NAND flash RS Parity 1 register	RW	0x00000000	0x1301010C	32/16/8
NFPAR2	NAND flash RS Parity 2 register	RW	0x00000000	0x13010110	32/16/8
NFINTS	NAND flash Interrupt Status register	RW	0x00000000	0x13010114	32
NFINTE	NAND flash Interrupt Enable register	RW	0x00000000	0x13010118	32
NFERR0	NAND flash RS Error Report 0 register	R	0x00000000	0x1301011C	32/16
NFERR1	NAND flash RS Error Report 1 register	R	0x00000000	0x13010120	32/16
NFERR2	NAND flash RS Error Report 2 register	R	0x00000000	0x13010124	32/16
NFERR3	NAND flash RS Error Report 3 register	R	0x00000000	0x13010128	32/16

1.5.1.1 NAND Flash Control/Status Register (NFCSR)

NFCSR is a 32-bit read/write register that configure NAND flash. It is initialized by any reset.

NFCSR **0x13010050**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
																									NFCE4	NFE4	NFCE3	NFE3	NFCE2	NFE2	NFCE1	NFE1
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:16	Reserved	Writes to these bits have no effect and read always as 0.	R
1/3/5/7	FCEn (n=1,2,3,4)	**NAND Flash FCE# Assertion Control :** Controls the assertion of NAND Flash FCEn#. When set, FCEn# is always asserted until this bit is cleared. When the NAND flash require FCEn# to be asserted during read busy time, this bit should be set	RW

		FCE **Description** 0 FCEn# is asserted as normal static chip enable(Initial value) 1 FCEn# is always asserted	
0/2/4/6	NFEn (n=1,2,3,4)	**NAND Flash Enable:** Specifies if NAND flash is connected to static bank n. When system is configured to boot from NAND flash, this bit is initialized to 1. **NFE** **Description** 0 Static bank n is not used as NAND flash. 1 Static bank n is used as NAND flash.	RW

1.5.1.2 NAND Flash ECC Control Register (NFECCR)

NFECCR is a 32-bit read/write register that is used to control ECC calculation. It is initialized by any reset.

NFECCR **0x13010100**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
																												PRDY	ENCE	RSE	ERST	ECCE
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:5	Reserved	Writes to these bits have no effect and read always as 0.	R
4	PRDY	**PAR Ready**: It is used to indicate the parity data is ready in NFPAR0~2 register during RS decoding. It is automatically cleared by hardware and always read as 0. **PRDY** **Description** 0 Parity data is not available (Initial value) 1 Parity data is ready in NFPAR0~2 registers	W
3	ENCE	**RS Encoding/Decoding Select**: It is used to define whether in encoding or in decoding phase when RS is used. **ENCE** **Description** 0 Decoding (Initial value) 1 Encoding	RW
2	RSE	**Hamming and RS codes Select**: It is used to select the correction algorithm between Hamming and RS codes. **RSE** **Description** 0 Hamming (Initial value) 1 Reed-Solomn (RS)	RW
1	ERST	**NAND Flash ECC Reset:** It is used to reset ECC controller. This bit is cleared automatically by hardware and always read as 0.	W

		ERST **Description** 0 ECC controller is not reset(Initial value) 1 ECC controller is reset	
0	ECCE	**NAND Flash ECC Enable:** ECC correction is enable/disable. **ECCE** **Description** 0 ECC is disabled (initial value) 1 ECC is enabled	RW

1.5.1.3 NAND Flash ECC Data Register (NFECC)

NFECC is a 32-bit read only register that contains the result of ECC calculation. It is not initialized by any reset. When ERST of NFECCR is set, NFECC is initialized to 0.

NFECC **0x130100104**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
									ECC2								ECC1								ECC0							
RST	0	0	0	0	0	0	0	0	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	X	x	x	X

Bits	Name	Description	RW
31:24	Reserved	Writes to these bits have no effect and read always as 0.	R
23:16	ECC2	Byte 2 of ECC	R
15:8	ECC1	Byte 1 of ECC	R
7:0	ECC0	Byte 0 of ECC	R

1.5.1.4 NAND Flash Parity Register (NFPARn, n=0,1,2)

NFPAR0, NFPAR1 and NFPAR2 are all 32-bit read/write register that contains the encoding and decoding parity data during RS correction. It is initialized by any reset and ERST of NFECCR.

PARn (n=0~7), total 8 9-bit register together contains the parity data during RS correction. In encoding, they are writtern by hardware and software needs to read out and write into NAND flash spare space after NFINTS.ENCF bit is set to 1. In decoding, they are writtern by software. Software should first read out the 512B nand flash data and then 8 9-bit parity data and write the parity data into PARn registers.

NFPAR0 **0x13010108**

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

	PAR4(4~0 bits)	PAR5	PAR6	PAR7
RST	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0			

NFPAR1 **0x1301010C**

Bit	31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
	PAR0(0 bit) / PAR1 / PAR2 / PAR3 / PAR4(8~5 bits)
RST	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

NFPAR2 **0x13010110**

Bit	31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
	Reserved / PAR0 (8~1 bits)
RST	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

1.5.1.5 NAND Flash Interrupt Status Register (NFINTS)

NFINTS is a 32-bit read-only register that contains the interrupt flag and error count information during RS correction. It is initialized by any reset. Software write 0 to clear the corresponding bit except ERRC.

NFINTS **0x13010114**

Bit	31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
	ERRC / Reserved / PADF / DECF / ENCF / UNCOR / ERR
RST	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

Bits	Name	Description	RW
31:29	ERRC	**ERR Count:** It indicates the number of errors in the nand flash data block and these bits are also reset by NFECCR.ERST bit. **ERRC** **Description** 0 No errors or uncorrection error occurs (Initial value) 1 One error in the nand flash data block 2 Two errors in the nand flash data block	R

		3 Three errors 4 Four errors	
28:5	Reserved	Writes to these bits have no effect and read always as 0.	R
4	PADF	**Padding Finish:** It indicates that hardware finish padding zero after reading the 512B nand flash data block during RS decoding. **PADF Description** 0 Padding not finish (Initial value) 1 Padding finish	R
3	DECF	**Decoding Finish:** It indicates that hardware finish RS decoding. **PADF Description** 0 Decoding not Finish (Initial value) 1 Decoding Finish	R
2	ENCF	**Encoding Finish:** It indicates that hardware finish RS encoding. **PADF Description** 0 Encoding not Finish (Initial value) 1 Encoding Finish	R
1	UNCOR	**Uncorrection Error:** It indicates that hardware finish RS encoding. **UNCOR Description** 0 No uncorrectable error (Initial value) 1 Uncorrectable error occur	R
0	ERR	**Error:** It indicates that hardware detects error data in the 512B nand flash data block during RS decoding. **ERR Description** 0 No error (Initial value) 1 Error occur	R

1.5.1.6 NAND Flash Interrupt Enable Register (NFINTE)

NFINTE is a 32-bit read/write register that is used to enable/disable nand flash interrupt during RS correction. It is initialized by any reset.

NFINTE **0x13010118**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved																											PADFE	DECFE	ENCFE	UNCORE	ERRE
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:5	Reserved	Writes to these bits have no effect and read always as 0.	R
4	PADFE	**Padding Finish Interrupt Enable:** It is used enable or disable padding finish interrupt. **PADFE Description**	RW

		0 Disable Padding finish interrupt (Initial value) 1 Enable Padding finish interrupt	
3	DECFE	**Decoding Finish Interrupt Enable:** It is used to enable or disable decoding finish interrupt. **DECFE Description** 0 Disable Decoding Finish Interrupt (Initial value) 1 Enable Decoding Finish Interrupt	RW
2	ENCFE	**Encoding Finish Interrupt Enable:** It is used to enable or disable encoding finish interrupt. **ENCFE Description** 0 Disable Encoding Finish Interrupt (Initial value) 1 Enable Encoding Finish Interrupt	RW
1	UNCORE	**Uncorrection Error Interrupt Enable:** It is used to enable or disable uncorrection error interrupt. **UNCORE Description** 0 Disable Uncorrectable Error interrut (Initial value) 1 Enable Uncorrectable Error Interrupt	RW
0	ERRE	**Error Interrupt Enable:** It is used to enable or disable error interrupt. **ERRE Description** 0 Disable Error interrupt (Initial value) 1 Enable Error interrupt	RW

1.5.1.7 NAND Flash Error Report Register (NFERRn, n=0,1,2,3)

NFERRn is 32-bit read/write register that contains the index and error value for each error symbol after RS decoding. It is initialized by any reset and ERST of NFECCR.

NFERR0 **0x1301011C**
NFERR1 **0x13010120**
NFERR2 **0x13010124**
NFERR3 **0x13010128**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved							INDEXn(n=0,1,2,3)									Reserved							MASKn(n=0,1,2,3)								
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
31:25	Reserved	Writes to these bits have no effect and read always as 0.	R
24:16	INDEXn	**Error Symbol Index:** It is used to indicate the location of the error symbol in the 511 symbols. For example, INDEX=1, it means the first symbol has error bits.	R

15:9	Reserved	Writes to these bits have no effect and read always as 0.	R
8:0	MASKn	**Error Symbol Value:** It is used to indicate the error value of the indexed symbol. For example, INDEX=1, and MASK=3, it means the first two bits of the first symbol are wrong, and software need to XOR MASK and the indexed symbol to get the right data.	R

1.5.2 NAND Flash Boot Loader

To support boot from NAND flash, 4KB on-chip Boot ROM is implemented. Following figure illustrates the structure of NAND Flash Boot Loader.

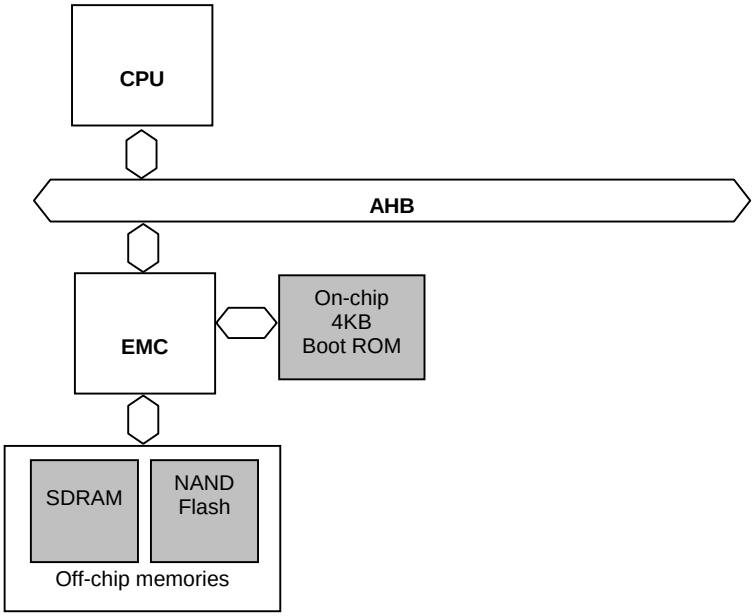


Figure 1-14 Structure of NAND Flash Boot Loader

When system is configured to boot from NAND flash, after reset, the program in Boot ROM is executed and the program will copy the first 4K bytes of NAND flash to internal memory for further initialization.

Generally, the boot code will copy more NAND flash content to SDRAM. Hardware ECC can be utilized to check the data validity. Then the main program will be executed on SDRAM.

When system is configured to boot from NAND flash, software may know the nand flash page size through BOOT_SEL[1:0] pin.

1.5.3 NAND Flash Operation

Set NFEn bit of NAND Flash Control/Status Register (NFCSR) will enable access to NAND flash. The partition of static bank n (n=1~4) is changed as following figure. Writes to any of address space will be translated to NAND flash address cycle. Writes to any of command space will be translated to NAND flash command cycle. Caution: don't read to address and command space, and these two partitions should be uncacheable. Reads and writes to any of data space will be translated to NAND flash data read/write cycle. DMA access to data space is supported to increase the speed of data read/write. The DMA access cannot exceed the page boundary (512 bytes or 2K bytes) of NAND flash.

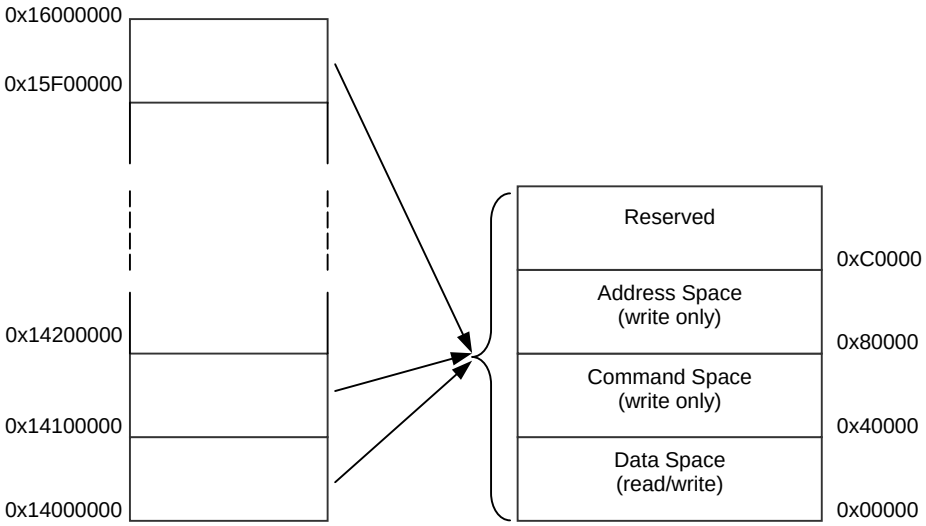


Figure 1-15 Static Bank 2 Partition When NAND Flash is Used (an example)

The timing of NAND flash access is configured by SMCRn and is same as normal static memory timing, except that CSn# is controlled by NFCE bit NFCSR. CSn# is always asserted when NFCE is 1. When NFCE is 0, CSn# is asserted as normal static memory access.

The control signals for direction connection of NAND flash are CSn#, FRE#, FWE#, FRB#(GPIO), A16 and A15. Following figure shows the connection between processor and NAND Flash.

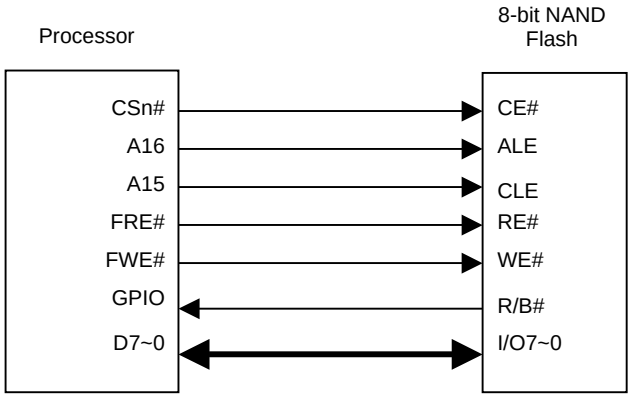


Figure 1-16 Example of 8-bit NAND Flash Connection

Hardware ECC generation for 8-/16-bit organization is implemented. There are two algorithm that could be used.

1.5.3.1 Hamming

When using Hamming algorithm, ECC parity code consists of 24 bits per 512 bytes (256 halfwords) and 22 bits per 256 bytes. Following table shows the ECC code assignment.

Table 1-6 512-Byte ECC Parity Code Assignment Table For 8-bit NAND Flash

	Data7	Data6	Data5	Data4	Data3	Data2	Data1	Data0
ECC0	P64	P64’	P32	P32’	P16	P16’	P8	P8’
ECC1	P1024	P1024’	P512	P512’	P256	P256’	P128	P128’
ECC2	P4	P4’	P2	P2’	P1	P1’	P2048	P2048’

24-bit ECC parity code = 18-bit line parity + 6-bit column parity

Table 1-7 256-Byte ECC Parity Code Assignment Table For 8-bit NAND Flash

	Data7	Data6	Data5	Data4	Data3	Data2	Data1	Data0
ECC0	P64	P64’	P32	P32’	P16	P16’	P8	P8’
ECC1	P1024	P1024’	P512	P512’	P256	P256’	P128	P128’
ECC2	P4	P4’	P2	P2’	P1	P1’	X	X

22-bit ECC parity code = 16-bit line parity + 6-bit column parity

Table 1-8 256-Halfword ECC Parity Code Assignment Table For 16-bit NAND Flash

	Data7	Data6	Data5	Data4	Data3	Data2	Data1	Data0
ECC0	P128	P128’	P64	P64’	P32	P32’	P16	P16’
ECC1	P2048	P2048’	P1024	P1024’	P512	P512’	P256	P256’

ECC2	P8	P8’	P4	P4’	P2	P2’	P1	P1’

24-bit ECC parity code = 16-bit line parity + 8-bit column parity

1.5.3.2 Reed-Solomn

RS controller uses RS(511, 503) codes. The total codes have 511 symbols, each symbol is 9-bit. The message has 503 9-bit symbols. So 512B nand flash data will be split into 9-bit symbols and it is 455 1/9 symbols, and then hardware will padding 47 8/9 zero symbol to build the 503 symbol message. During encoding, after padding zero, hardware will generate 8 9-bit parity symbol, software should read out the parity and write into nand flash spare space. During decoding, after reading out the 512B data and padding zero, hardware will using the parity data in NFPAR0~2 written by software to generate error information in NFERR0~3. Software should first read out the 512B nand flash data and then the parity data and write the parity data into NFPAR0~2.

NAND Flash Initialize Sequence

1. Configure SMCRn (n=1,2,3,4) according to the NAND flash AC characteristics.
2. Set NFEn bit of NFCSR to 1

NAND Flash Program Sequence

1. Set NFCEn bit of NFCSR to 1 to assert CSn# continuously
2. Write 0x80 to command space to issue Page Program command
3. Write 2 or 3 bytes of address to address space
4. Enable and reset the ECC generator by setting ECCE and ERST bits of NFECCR.
5. Select Haming or RS algorithm by setting NFECCR.RSE, 0: Hamming, 1: RS
6. If using RS algorithm, set NFECCR.ENCE to 1
7. Write 256 or 512 bytes data to data space using Hamming, write 512B data using RS
8. When using Hamming, clear ECCE bit of NFECCR to disable ECC generator and read out the ECC parity code from NFECC register, write 16-byte redundant data
9. When using RS, poll NFINTS.ENCF bit or wait for ENCF interrupt, and read out parity data from NFPAR0~2 register and write them into nand flash spare space
10. Write 0x10 to command space to issue Page Program command
11. Clear NFCEn bit of NFCSR to deassert CSn#
12. Check RB bit of NFCSR to wait the program complete

NAND Flash Read Sequence

1. Set NFCEn bit of NFCSR to 1 to assert CSn# continuously
2. Write 0x00 to command space to issue Read command
3. Write 2 or 3 bytes of address-to-address space

4. Enable and reset the ECC generator by setting ECCE and ERST bits of NFECCR.
5. Select Haming or RS algorithm by setting NFECCR.RSE, 0: Hamming, 1: RS
6. Check RB# pin (GPIO) to wait for NAND flash is not busy
7. When using Hamming, Read 256 or 512 bytes from NAND flash (DMA can be used), clear ECCE bit of NFECCR to disable ECC generator and Read 16-byte redundant data from NAND flash, then read out the ECC parity code from NFECC register, compare these two parity codes and correct the error bit or run the error routine
8. When using RS, Read 512 bytes data and 9 bytes parity data from NAND flash (DMA can be used), poll NFINTS.DECF bit or wait for decoding finish interrupt, and then analyze error status in NFINTS register, if there is any error, read NFERR0~3 registers, find the error byte according to NFERRn.INDEXn and XOR NFERRn.MASKn with the indexed bytes to get the correct data
9. If continuous pages are needed to be read, repeat steps from 5 to 9.
10. Clear NFCEn bit of NFCSR to deassert CSn#

ECC Generation Sequence for large page

In large page (2KBX8/1KHWX16 org.), 24-bit ECC code is generated for every 512 bytes or 256 halfwords data. Software gets ECC codes as following steps:

1. Set ERST bit of NFECCR to reset ECC generator
2. Read/write 512 bytes / 256 halfwords data
3. Read out 24-bit ECC code from NFECC register
4. Repeat step 1 to 3 until whole page is completed
5. Write spare area according to above ECC codes.

1.6 SDRAM Interface

The SDRAM controller provides a glueless interface to industry standard SDRAM chip. The SDRAM controller provides one chip selects DCS# supporting 16-bit or 32-bit wide SDRAM.

Both 2-bank and 4-bank SDRAM modules are supported. The bank select signals are always output from the A13 pin and A14 pin of processor.

The SDRAM interface includes the following signals:

- One chip selects, DCS#
- Four byte mask signals, DQM3~0#
- 15 multiplexed bank/row/column address signals, A14-A0
- One write enable, RD/WR#
- One column-address strobe CAS#
- One row-address strobe RAS#
- One clock enable CKE
- One clock CKO

The processor performs auto-refresh (CBR) during normal operation and supports self-refreshing SDRAM during sleep, hibernate, and frequency-change modes. An SDRAM power-down mode bit (DMCR[PDM]]) can be set so that the CKO and the clock-enable signal CKE to SDRAM are automatically deasserted whenever none of the corresponding banks is being accessed.

1.6.1 Register Description

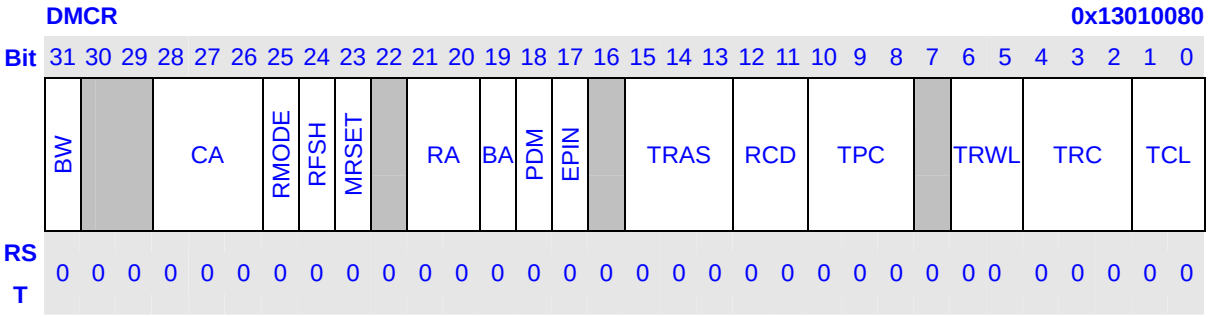
Table 1-9 SDRAM Registers

Name	Description	RW	Reset Value	Address	Access Width
DMCR	DRAM control register	RW	0x0000 0000	0x13010080	32
RTCSR	Refresh time control/status register	RW	0x0000	0x13010084	16
RTCNT	Refresh timer counter	RW	0x0000	0x13010088	16
RTCOR	Refresh time constant register	RW	0x0000	0x1301008C	16
DMAR	SDRAM bank address configuration register	RW	0x000020F8	0x13010090	32
SDMR	Mode register of SDRAM bank	W	--	0x1301A000	8

1.6.1.1 SDRAM Control Register (DMCR)

DMCR is a 32-bit read/write register that specifies the timing, address multiplexing and refresh control of SDRAM. This enables direct connection of SDRAM without external circuits.

The DMCR is initialized to 0x00000000 by any resets. SDRAM bank should not be accessed until initialization is completed.



Bits	Name	Description	RW
31	BW	Specifies the data bus width of SDRAM **BW** **Description** 0 Data width is 32 bits (Initial value) 1 Data width is 16 bits	RW
30:29	Reserved	Writes to these bits have no effect and always read as 0.	R
28:26	CA	**Column Address Width**: Specify the column address width of connected SDRAM chip. **CA** **Description** 000 8 bits column address 001 9 bits column address 010 10 bits column address	RW

		011 11 bits column address 100 12 bits column address 101 Reserved 110 Reserved 111 Reserved	
25	RMODE	**Refresh Mode.** **RMODE Description** 0 Auto-refresh 1 Self-refresh	RW
24	RFSH	**Refresh Control.** **RFSH Description** 0 No refresh is performed (Initial value) 1 Refresh is performed	RW
23	MRSET	**Mode Register Set:** Set when a SDRAM mode register setting is used. When this bit is 0 and SDRAM mode register is written, a Pre-charge all banks command (PALL) is performed. When this bit is 1 and SDRAM mode register is written, a Mode Register Set command (MRS) is performed. **MRSET Description** 0 All-bank pre-charge (Initial value) 1 Mode register setting	RW
22	Reserved	Writes to these bits have no effect and always read as 0.	R
21:20	RA	**Row Address Width:** Specify the row address width of connected SDRAM. **RA Description** 00 11-bit row address (Initial value) 01 12-bit row address 10 13-bit row address 11 Reserved	RW
19	BA	**Bank Address Width:** Specify the number of bank select signals for one chip select. **BA Description** **0** 1-bit bank address is used (2 banks each chip select) (Initial value) **1** 2-bit bank address is used (4 banks each chip select)	RW
18	PDM	**Power Down Mode:** Set power-down mode. When power-down mode is set, SDRAM will be driven to power-down mode when it is not accessing and refreshing. Clock supply to SDRAM will be stopped also. **PDM Description** 0 Non-power-down mode (Initial value) 1 Power-down mode	RW
17	EPIN	**CKE Pin Control:** Controls the level of CKE pin. Clearing this bit by software causes a power-down command (if CKOEN of CPM is 1).	RW

		Caution: after power-down command, all commands except power-down-exit are prohibited. Setting this bit by software causes a power-down-exit command. Setting EPIN is a part of initializes procedure for SDRAM. **EPIN** **Description** 0 CKE pin is deserted (Initial value) 1 CKE pin is asserted	
16	Reserved	Writes to these bits have no effect and always read as 0.	R
15:13	TRAS	**RAS Assertion Time:** When synchronous DRAM is connected, these bits set the minimum CKE negation time after self-refresh command is issued. **TRAS** **Description** 000 4 (Initial value) 001 5 010 6 011 7 100 8 101 9 110 10 111 11	RW
12:11	RCD	**RAS–CAS Delay**: Set the SDRAM bank active-read/write command delay time. **RCD** **Description** 00 1(Initial value) 01 2 10 3 11 4	RW
10:8	TPC	**RAS Precharge Time:** Specify the minimum number of cycles until the next bank active command is output after precharging. TPC Description 000 1 cycle (Initial value) 001 2 cycles 010 3 cycles 011 4 cycles 100 5 cycles 101 6 cycles 110 7 cycles 111 8 cycles	RW
7	Reserved	Writes to these bits have no effect and always read as 0.	R
6:5	TRWL	**Write Precharge Time:** Set the SDRAM write precharge delay time. In auto-precharge mode, they specify the time until the next bank active command is issued after a write cycle. After a write cycle, the next active command is not issued for a period of TRWL + TPC.	RW

		TRWL **Description** 00 1 cycle (Initial value) 01 2 cycles 10 3 cycles 11 4 cycles	
4:2	TRC	**RAS Cycle Time:** For SDRAM, no bank active command is issued during the period TRC after an auto-refresh command. In self-refresh, these bits also specify the delay cycles to be inserted after CKE assertion. **TRC** **Description** 000 1 cycle (Initial value) 001 3 cycle 010 5 cycle 011 7 cycle 100 9 cycle 101 11 cycle 110 13 cycle 111 15 cycle	RW
1:0	TCL	**CAS Latency:** Specify the delay from read command to data becomes available at the outputs. **TCL** **Description** 00 Inhibit (Initial value) 01 2 cycles 10 3 cycles 11 Inhibit	RW

1.6.1.2 SDRAM Mode Register (SDMR)

SDMR is written to via the SDRAM address bus and is a 10-bit write-only register. It sets SDRAM mode for SDRAM bank. SDMR is undefined after a reset.

Write to the SDRAM mode register use the address bus rather than the data bus. If the value to be set is X and the SDMR address is Y, the value X is written in the SDRAM mode register by writing in address X + Y. Here Y is 0xA000, X is value for SDRAM configuration. For example Y is 0x022, random data is writer to the address offset 0xA022, as a result, 0x022 is written to the SDMR register. The range for value X is 0x000 to 0x3FF.

SDMR **0x1301A000**

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

The Mode Register is used to define the specific mode of operation of the SDRAM. This definition includes the section of a burst length, a burst type, a CAS latency, an operating mode and a write burst mode, as shown in following figure.

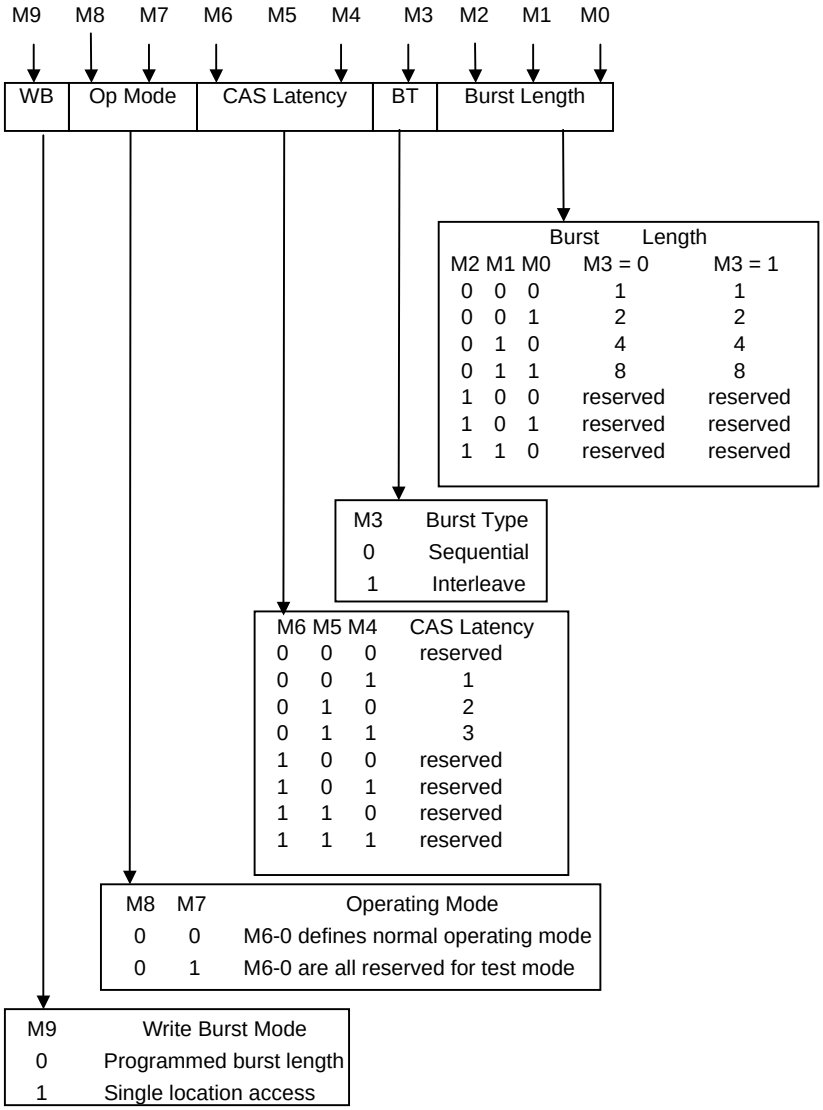
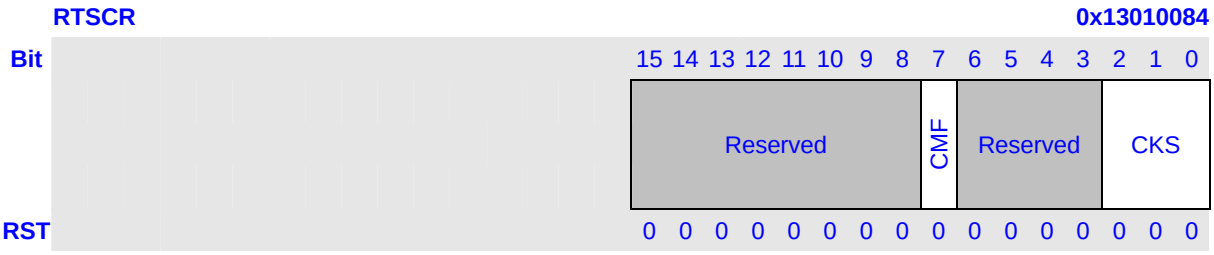


Figure 1-17 Synchronous DRAM Mode Register Configuration

1.6.1.3 Refresh Timer Control/Status Register (RTCSR)

RTCSR is a 16-bit readable/writable register that specifies the refresh cycle and the status of RTCNT.

RTCSR is initialized to 0x0000 by a reset.



Bits	Name	Description	RW
15:8	Reserved	These bits always read 0. Data written to these bits are ignored	R
7	CMF	**Compare-Match Flag (CMF):** Status flag that indicates a match between the refresh timer counter (RTCNT) and refresh time constant register (RTCOR) values. Writes to 1 of this bit have no effect. **CMF** **Description** 0 RTCNT and RTCOR values do not match (Initial value) Clear condition: When 0 is written 1 RTCNT and RTCOR values match Set condition: When RTCNT = RTCOR	
2:0	CKS	**Refresh Clock Select Bits:** These bits select the clock input to RTCNT. The source clock is the external bus clock (CKO). The RTCNT count clock is CKO divided by the specified ratio. **CKS** **Description** 000 Disable clock input (Initial value) 001 Bus lock CKO/4 010 CKO/16 011 CKO/64 100 CKO/256 101 CKO/1024 110 CKO/2048 111 CKO/4096	

1.6.1.4 Refresh Timer Counter (RTCNT)

RTCNT is a 16-bit read/write register. RTCNT is a 16-bit counter that counts up with input clocks. The clock select bits (CKS2–CKS0) of RTCSR select the input clock. When the refresh bit (RFSH) of the memory control register (DMCR) is set to 1 and the refresh mode is set to auto-refresh, a memory refresh cycle starts when RTCNT matches RTCOR. RTCNT is initialized to 0x0000 by a reset.

RTCNT **0x13010088**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved															
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

1.6.2 Refresh Time Constant Register (RTCOR)

RTCOR is a 16-bit read/write register. The values of RTCOR and RTCNT (bottom 8 bits) are constantly compared. When the refresh bit (RFSH) of the memory control register (DMCR) is set to 1 and the refresh mode bit (RMODE) is set to auto-refresh, a memory refresh cycle starts when RTCNT matches RTCOR. RTCOR is initialized to 0x0000 by a reset.

RTCOR **0x1301008C**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved															
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

1.6.2.1 DRAM Bank Address Configuration Register (DMAR)

DMAR define the physical address for SDRAM bank, respectively. Each register contains a base address and a mask. When the following equation is met:

(*physical_address* [31:24] & **MASK$_n$**) == **BASE$_n$**

The bank n is active. The *physical_address* is address output on internal system bus. DRAM bank regions must be programmed so that each bank occupies a unique area of the physical address space. Programming overlapping bank regions will result in unpredictable error.

These registers are initialized by a reset.

DMAR **0x13010090**

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

Bits	Name	Description	RW
31:16	Reserved	Writes to these bits have no effect and read always as 0.	R
15:8	BASE	**Address Base:** Defines the base address of SDRAM Bank. The initial values are: DMAR.BASE 0x20	RW
23:20	MASK	**Address Mask**: Defines the mask of SDRAM Bank. The initial values are: DMAR.MASK 0xF8	RW

1.6.3 Example of Connection

Following figure shows an example of connection of 512K x 16-bit x 2-bank SDRAM.

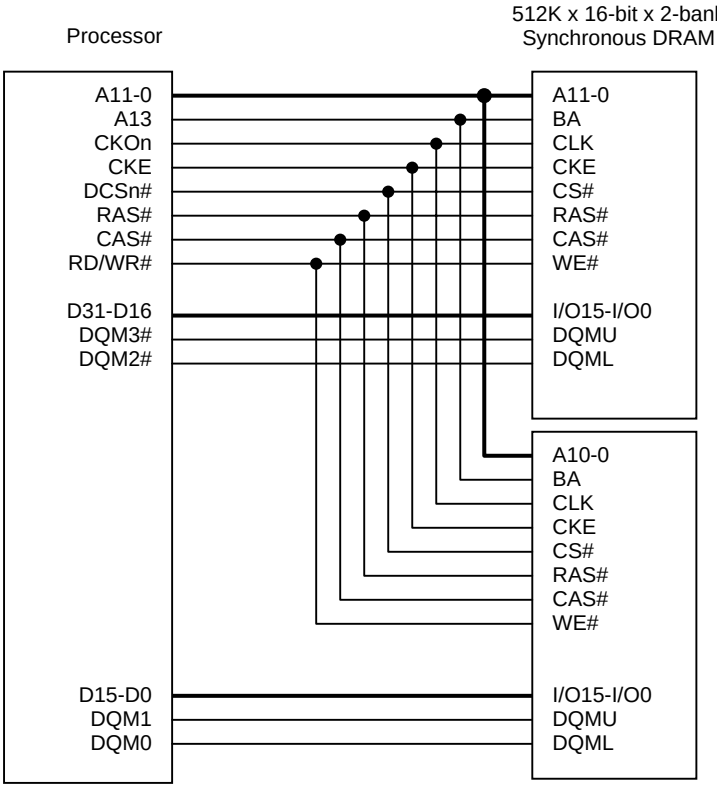


Figure 1-18 Example of Synchronous DRAM Chip Connection (1)

Following figure shows an example of connection of 1M x 16-bit x 4-bank synchronous DRAM.

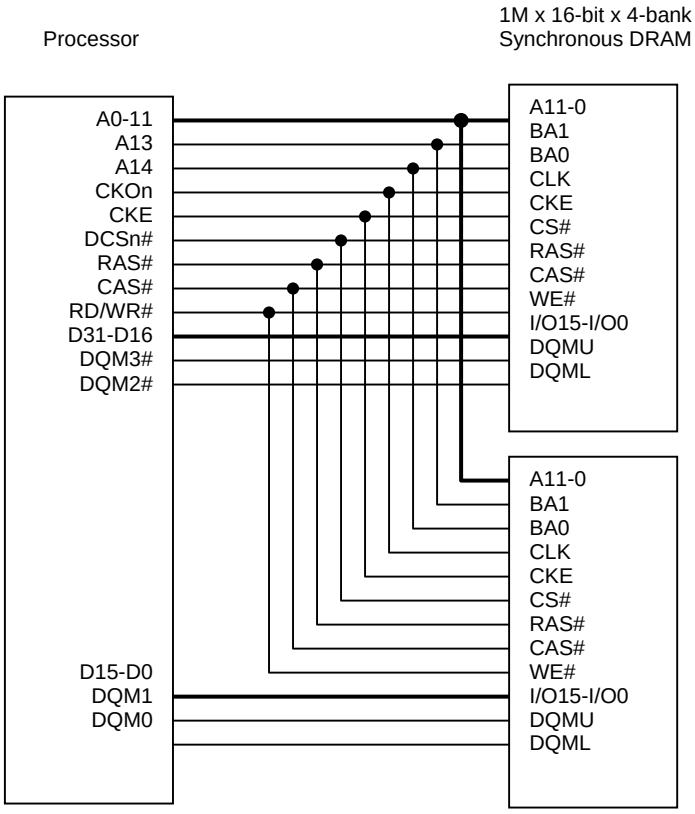


Figure 1-19 Example of Synchronous DRAM Chip Connection (2)

1.6.4 Address Multiplexing

SDRAM can be connected without external multiplexing circuitry in accordance the address multiplex specification bits CA2~0, RA1~0 and BA in DMCR. Table 1-10 shows the relationship between the address multiplex specification bits and the bits output at the address pins.

A14-0 is used as SDRAM address. The original values are always output at these pins.

Table 1-10 SDRAM Address Multiplexing (32-bit data width) *[4]

CA2~0	RA1~0	Output Timing	A0-A9, A10, A11, A12	A13	A14	Note
8 bits	11 bits	Column	A2-A11, L/H*[1], A12, A13	A21	A22	3, 4
		Row	A10-A22			
	12 bits	Column	A2-A11, L/H*[1], A12, A13	A22	A23	3, 4
		Row	A10-A22			
	13 bits	Column	A2-A11, L/H*[1], A12, A13	A23	A24	3, 4
		Row	A10-A22			
9 bits	11 bits	Column	A2-A11, L/H*[1], A12, A13	A22	A23	3, 4
		Row	A11-A23			
	12 bits	Column	A2-A11, L/H*[1], A12, A13	A23	A24	3, 4
		Row	A11-A23			
	13 bits	Column	A2-A11, L/H*[1], A12, A13	A24	A25	3, 4
		Row	A11-A23			
10 bits	11 bits	Column	A2-A11, L/H*[1], A12, A13	A23	A24	3, 4
		Row	A12-A24			
	12 bits	Column	A2-A11, L/H*[1], A12, A13	A24	A25	3, 4
		Row	A12-A24			
	13 bits	Column	A2-A11, L/H*[1], A12, A13	A25	A26	3, 4
		Row	A12-A24			
11 bits	11 bits	Column	A2-A11, L/H*[1], A12, A13	A24	A25	3, 4
		Row	A13-A25,			
	12 bits	Column	A2-A11, L/H*[1], A12, A13	A25	A26	3, 4
		Row	A13-A25,			
	13 bits	Column	A2-A11, L/H*[1], A12-A17	A26	A27	3, 4
		Row	A13-A25,			
12 bits	11 bits	Column	A2-A11, L/H*[1], A12, A13	A25	A26	3, 4
		Row	A14-A26			
	12 bits	Column	A2-A11, L/H*[1], A12, A13	A26	A27	3, 4
		Row	A14-A26			
	13 bits	Column	A2-A11, L/H*[1], A12, A13	A27	A28	3, 4
		Row	A14-A26			

Notes:

1. L/H is a bit used in the command specification; it is fixed at L or H according to the Access mode.
2. Bank address specification
3. If one bank select signal is used (BA = 0), take A13 as bank select signal. If two bank select signals are used (BA = 1), take A13 and A14 as bank select signals
4. The A0 to A14 in table head are output pins. The A2 to A28 in table body are physical address.

Table 1-11 SDRAM Address Multiplexing (16-bit data width) *[4]

CA2~0	RA1~0	Output Timing	A0-A9, A10, A11, A12	A13	A14	Note
8 bits	11 bits	Column	A2-A11, L/H*[1], A12, A13	A21	A22	3, 4
		Row	A10-A22			
	12 bits	Column	A2-A11, L/H*[1], A12, A13	A22	A23	3, 4
		Row	A10-A22			
	13 bits	Column	A2-A11, L/H*[1], A12, A13	A23	A24	3, 4
		Row	A10-A22			
9 bits	11 bits	Column	A2-A11, L/H*[1], A12, A13	A22	A23	3, 4
		Row	A11-A23			
	12 bits	Column	A2-A11, L/H*[1], A12, A13	A23	A24	3, 4
		Row	A11-A23			
	13 bits	Column	A2-A11, L/H*[1], A12, A13	A24	A25	3, 4
		Row	A11-A23			
10 bits	11 bits	Column	A2-A11, L/H*[1], A12, A13	A23	A24	3, 4
		Row	A12-A24			
	12 bits	Column	A2-A11, L/H*[1], A12, A13	A24	A25	3, 4
		Row	A12-A24			
	13 bits	Column	A2-A11, L/H*[1], A12, A13	A25	A26	3, 4
		Row	A12-A24			
11 bits	11 bits	Column	A2-A11, L/H*[1], A12, A13	A24	A25	3, 4
		Row	A13-A25,			
	12 bits	Column	A2-A11, L/H*[1], A12, A13	A25	A26	3, 4
		Row	A13-A25,			
	13 bits	Column	A2-A11, L/H*[1], A12-A17	A26	A27	3, 4
		Row	A13-A25,			
12 bits	11 bits	Column	A2-A11, L/H*[1], A12, A13	A25	A26	3, 4
		Row	A14-A26			
	12 bits	Column	A2-A11, L/H*[1], A12, A13	A26	A27	3, 4
		Row	A14-A26			
	13 bits	Column	A2-A11, L/H*[1], A12, A13	A27	A28	3, 4
		Row	A14-A26			

Notes:

1. L/H is a bit used in the command specification; it is fixed at L or H according to the Access mode.
2. Bank address specification
3. If one bank select signal is used (BA = 0), take A13 as bank select signal. If two bank select signals are used (BA = 1), take A13 and A14 as bank select signals
4. The A0 to A14 in table head are output pins. The A2 to A28 in table body are physical address.

1.6.5 SDRAM Command

Commands for SDRAM are specified by RAS#, CAS#, RD/WR and special address signals. The processor accesses SDRAM by using the following subset of standard interface commands.

- Mode Register Set (MRS)
- Bank Activate (ACTV)
- Read (READ)
- Write (WRIT)
- Burst Terminate
- Precharge All Banks (PALL)
- Auto-Refresh (CBR)
- Enter Self-Refresh (SLFRSH)
- No Operation (NOP)

Table 1-12 SDRAM Command Encoding (Notes: 1)

Command	Processor Pins							
	CS#	RAS#	CAS#	RD/WR#	DQM	A14-11, A9-0	A10	Note
INHIBIT	H	X	X	X	X	X	X	
NOP	L	H	H	H	X	X	X	
MRS	L	L	L	L	X	Op-Code		
ACTV	L	L	H	H	X	Bank, Row	X	2
READ	L	H	L	H	L/H	Bank, Col	L	3
WRIT	L	H	L	L	L/H	Bank, Col	L	3
Burst Terminate	L	H	H	L	X	X	X	
PRE	L	L	H	L	X	Bank	L	
PALL	L	L	H	L	X	X	H	
CBR/SLFRSH	L	L	L	H	X	X	X	4

Note:

- CKE is HIGH for all commands shown except SLFRSH
- A0-A12 provides row address, and A13-A14 determines which bank is active.
- A0-A9 provides column address, and A13-A14 determines which bank is being read from or written to.
- This command is CBR if CKE is HIGH, SLFRSH if CKE is LOW.

1.6.6 SDRAM Timing

The SDRAM bank function is used to support high-speed accesses to the same row address. As SDRAM is internally divided into two or four banks, it is possible to activate one row address in each bank.

When a de-active bank is accessed, an access is performed by issuing an ACTV command following by READ or WRIT command.

When an active bank is accessed and just hit the open row, an access is performed by issuing READ or WRIT command immediately without issuing an ACTV command.

When an active bank is accessed but hit a closed row, a PRE command is first issued to precharge the bank, then the access is performed by issuing an ACTV command followed by a READ or WRIT command.

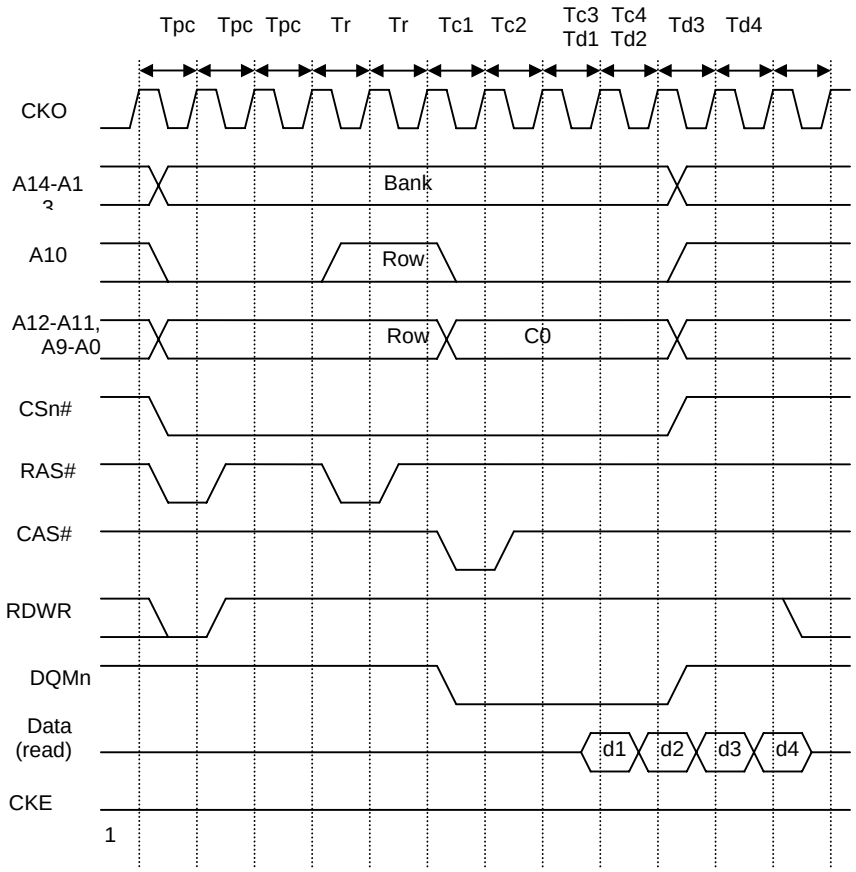
There is a limit on Tras, the time for placing each bank in the active state. If there is no guarantee that there will not be a cache hit and another row address will be accessed within the period in which this value is maintained by program execution, it is necessary to set auto-refresh and set the refresh cycle to no more than the maximum value of Tras. In this way, it is possible to observe the restrictions on the maximum active state time for each bank. If auto-refresh is not used, measures must be taken in the program to ensure that the banks do not remain active for longer than the prescribed time.

Glossary

Tr – row active cycle
Trw – row active wait cycle
Trwl – write latency cycle
Tpc – precharge cycle
TRr – refresh command cycle
Trc – RAS cycle
Trs1 – self refresh cycle 1
Trs2 – self refresh cycle 2
Trs3 – self refresh cycle 3
Trsw – self refresh wait cycle
Tc1 – command cycle 1
Tc2 – command cycle 2
Tc3 – command cycle 3
Tc4 – command cycle 4
Tc5 – command cycle 5
Tc6 – command cycle 6
Tc7 – command cycle 7
Tc8 – command cycle 8

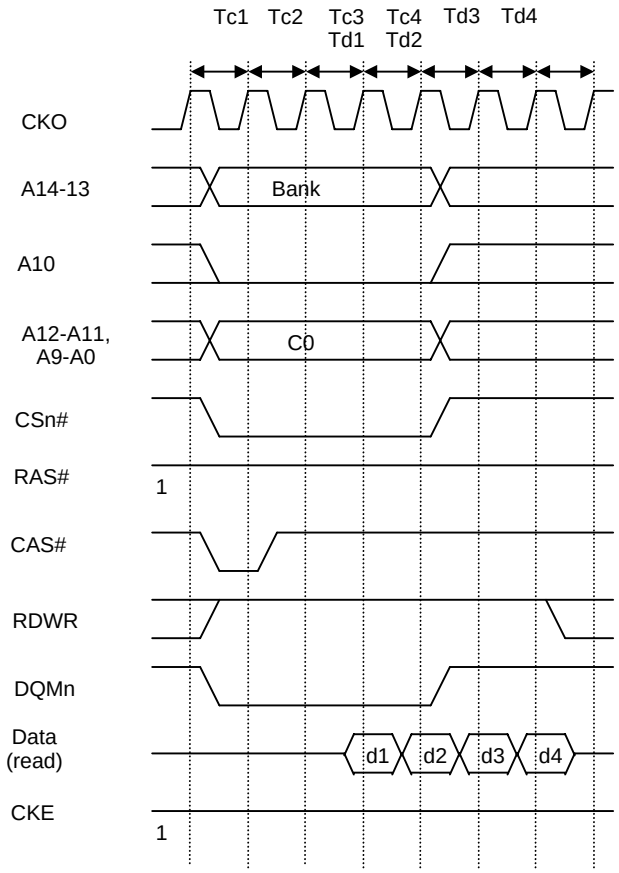
Td1 – data cycle 1
Td2 – data cycle 2
Td3 – data cycle 3
Td4 – data cycle 4
Td5 – data cycle 5
Td6 – data cycle 6
Td7 – data cycle 7
Td8 – data cycle 8
TRp1 – precharge-all cycle 1
TRp2 – precharge-all cycle 2
TRp3 – precharge-all cycle 3
TRp4 – precharge-all cycle 4
TMw1 – mode register set cycle 1
TMw2 – mode register set cycle 2
TMw3 – mode register set cycle 3
TMw4 – mode register set cycle 4

Following figures show the timing of 4-beat burst access, 8-beat burst access and single access.



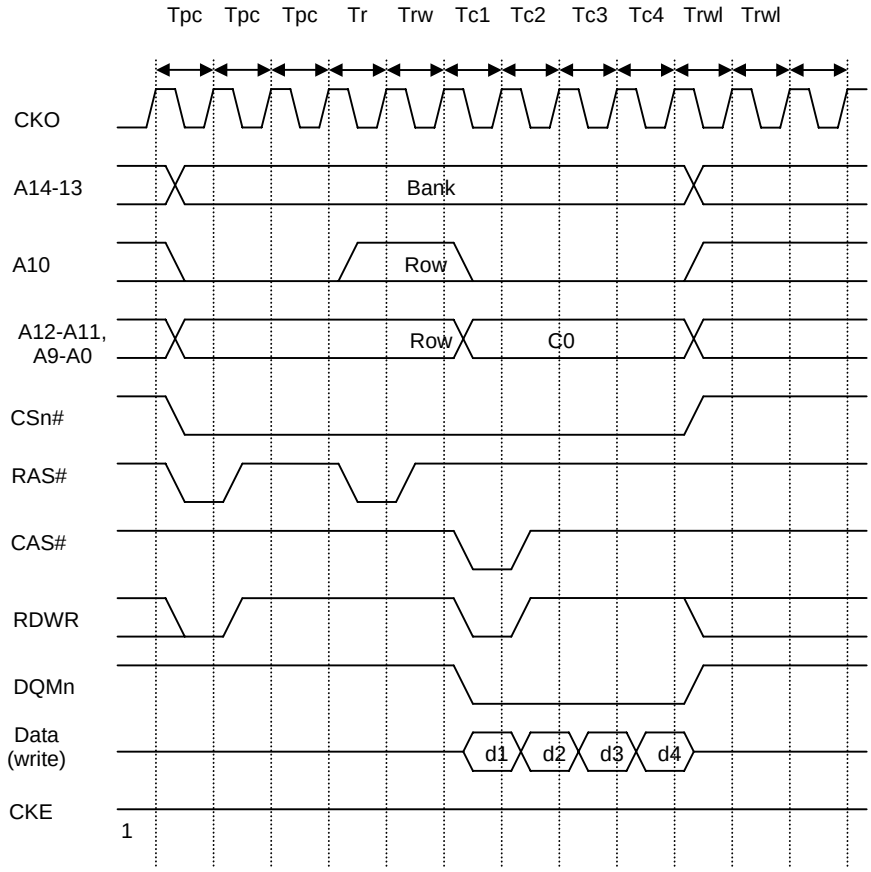
*DMCR: RCD = 1, TCL = 1, TPC = 2

Figure 1-20 Synchronous DRAM 4-beat Burst Read Timing (Different Row)



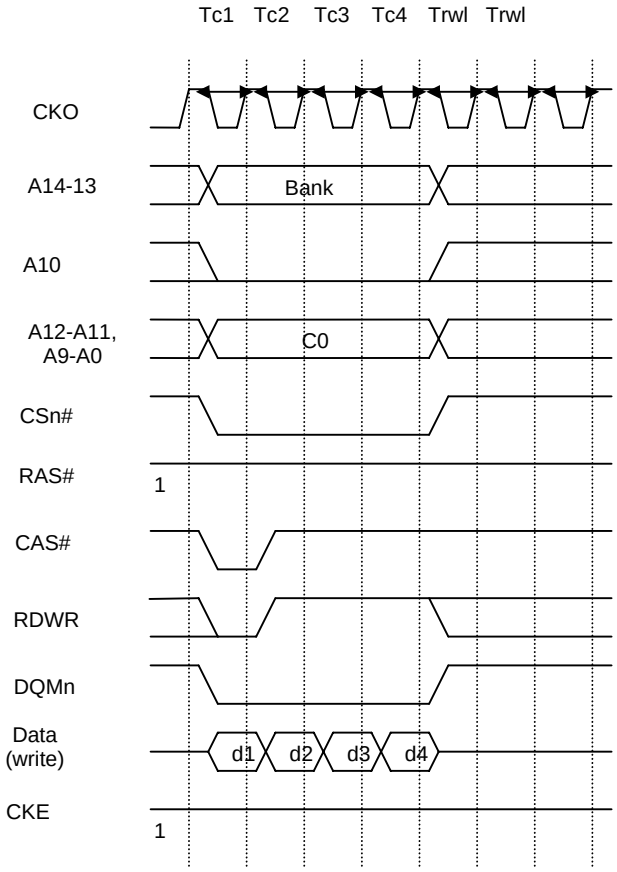
*DMCR: RCD = 1, TCL = 1, TPC = 2

Figure 1-21 Synchronous DRAM 4-beat Burst Read Timing (Same Row)



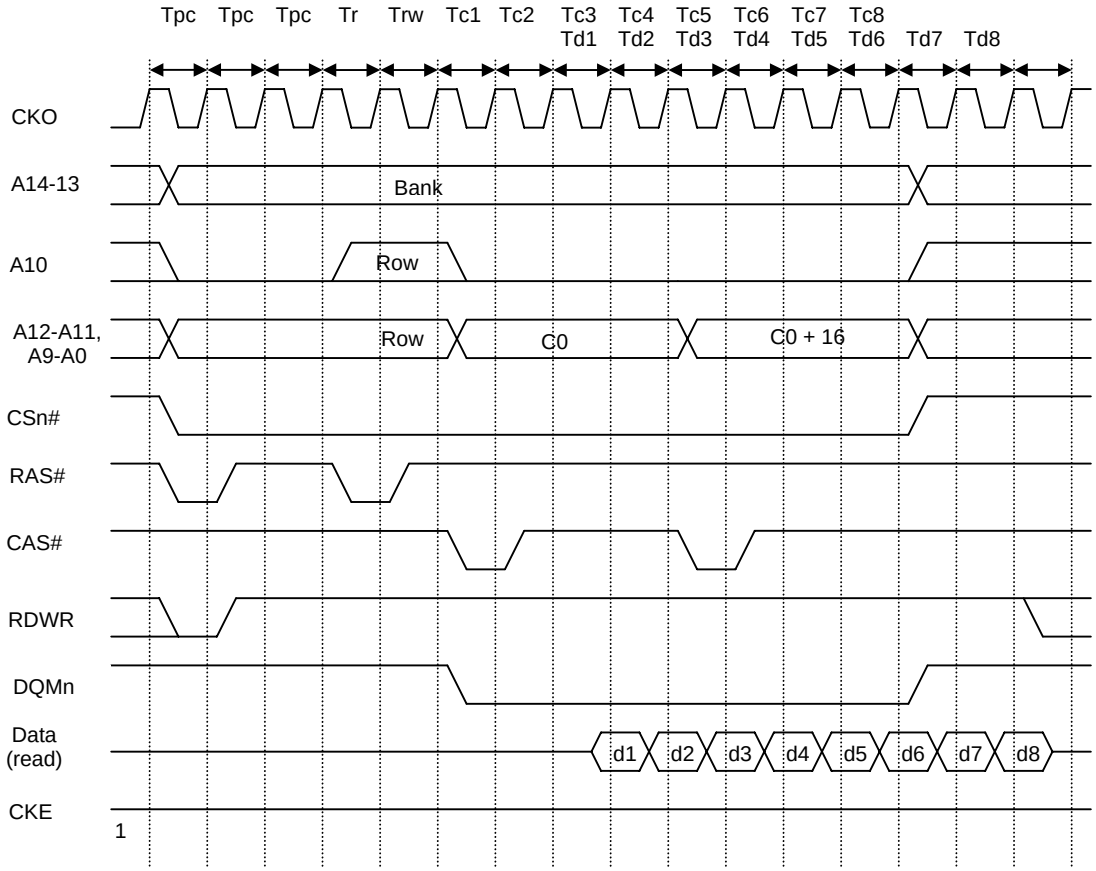
*DMCR: RCD = 1, TCL = 1, TPC = 2, TRWL = 1

Figure 1-22 Synchronous DRAM 4-beat Burst Write Timing (Different Row)



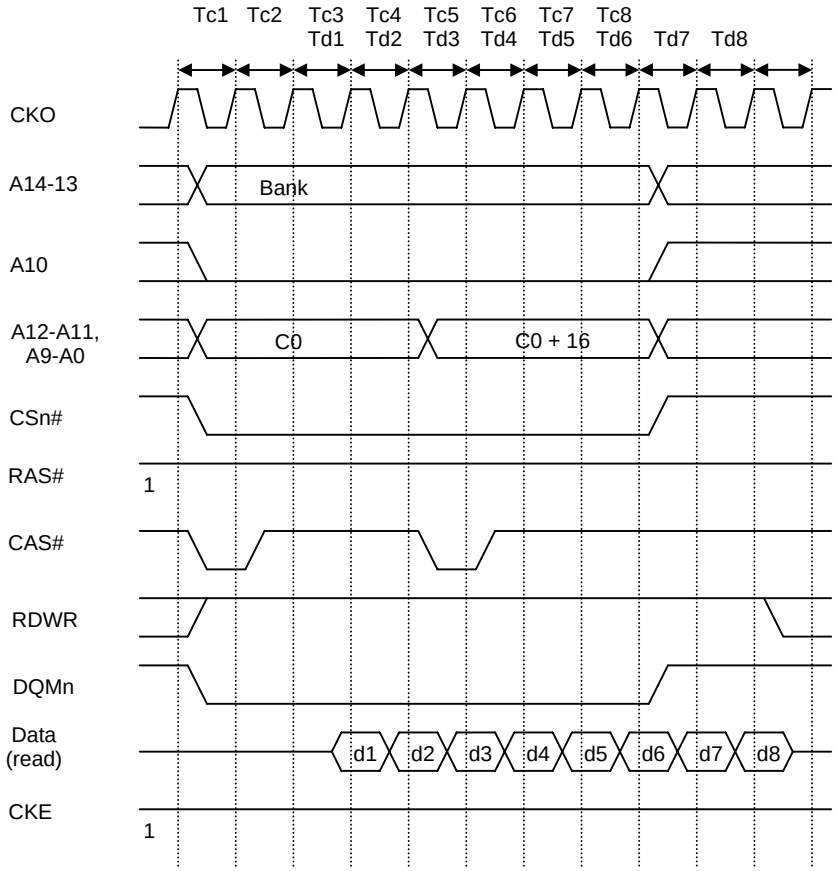
*DMCR: RCD = 1, TCL = 1, TPC = 2, TRWL = 1

Figure 1-23 Synchronous DRAM 4-beat Burst Write Timing (Same Row)



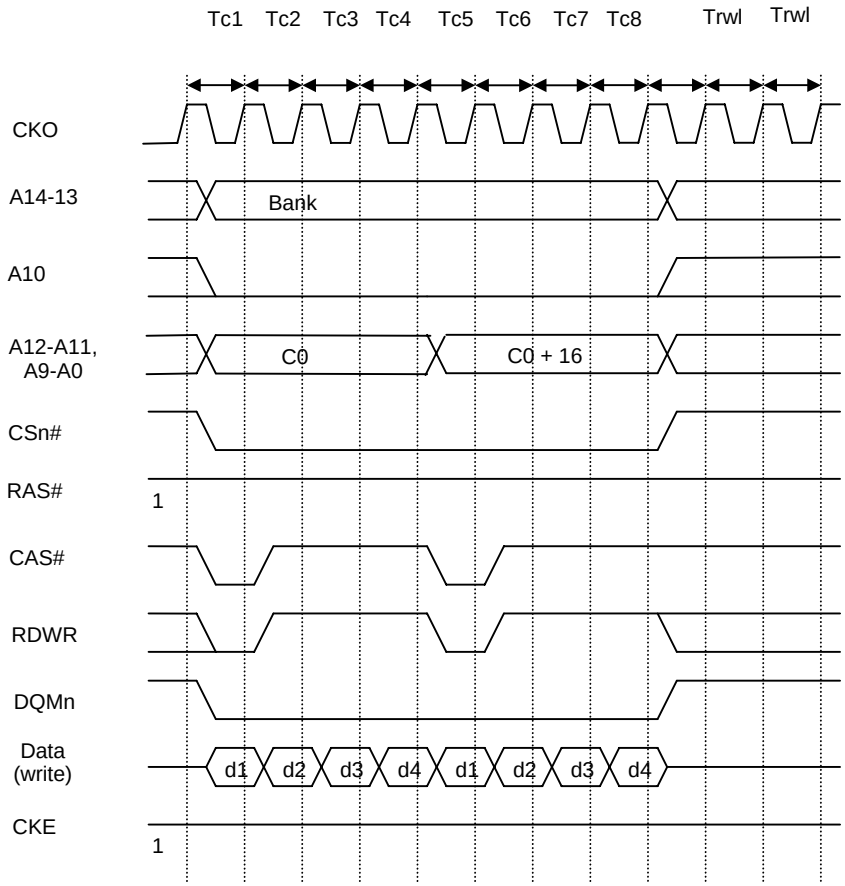
*DMCR: RCD = 1, TCL = 1, TPC = 2

Figure 1-24 Synchronous DRAM 8-beat Burst Read Timing (Different Row)



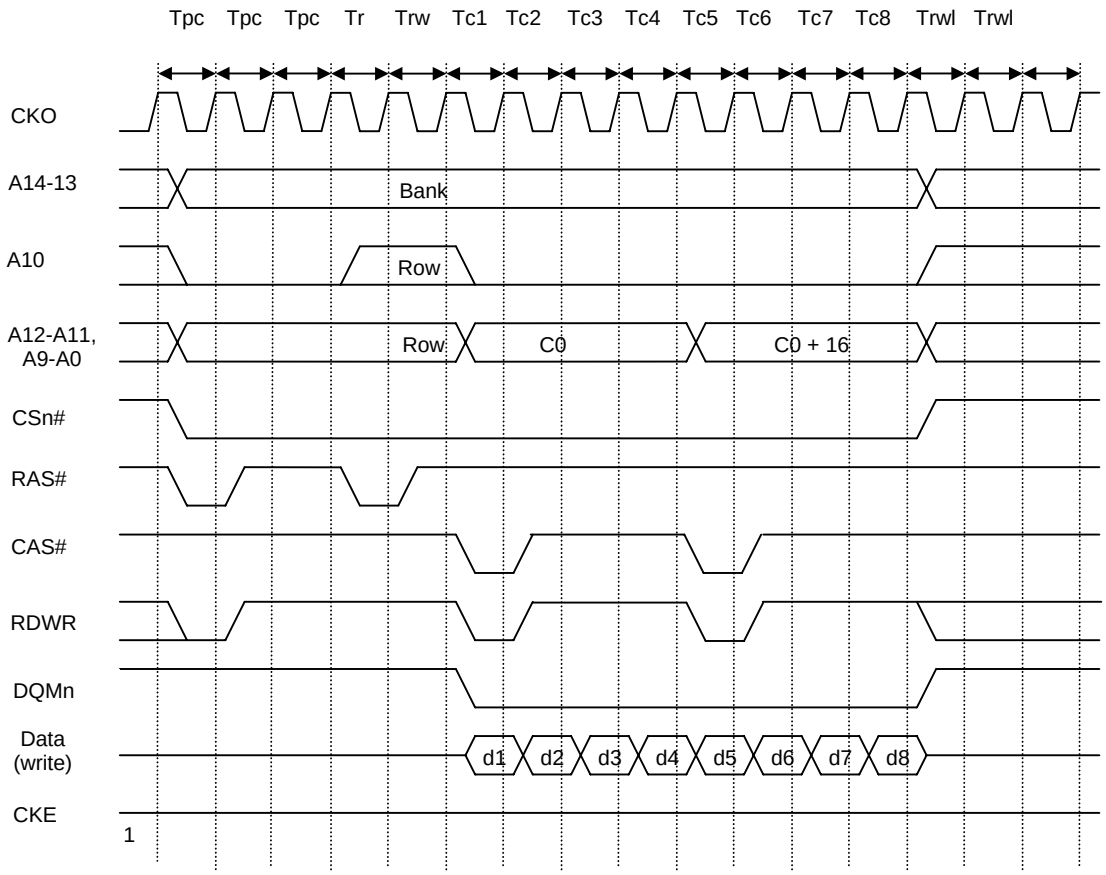
*DMCR: RCD = 1, TCL = 1, TPC = 2

Figure 1-25 Synchronous DRAM 8-beat Burst Read Timing (Same Row)



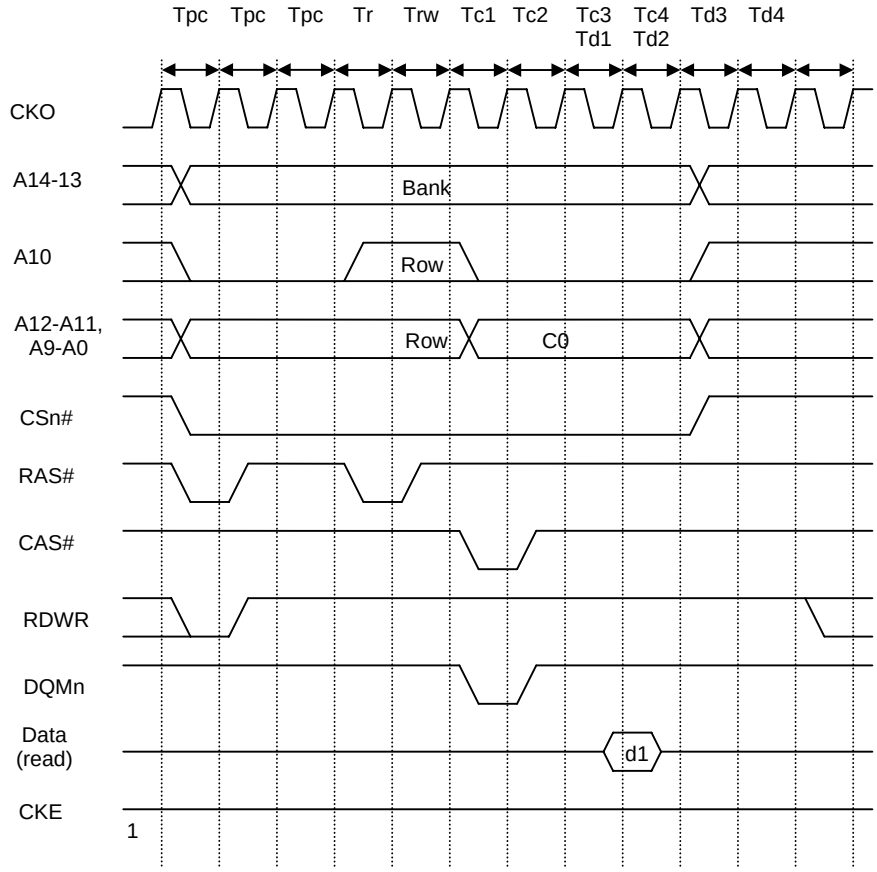
*DMCR: RCD = 1, TCL = 1, TPC = 2, TRWL = 1

Figure 1-26 Synchronous DRAM 8-beat Burst Write Timing (Same Row)



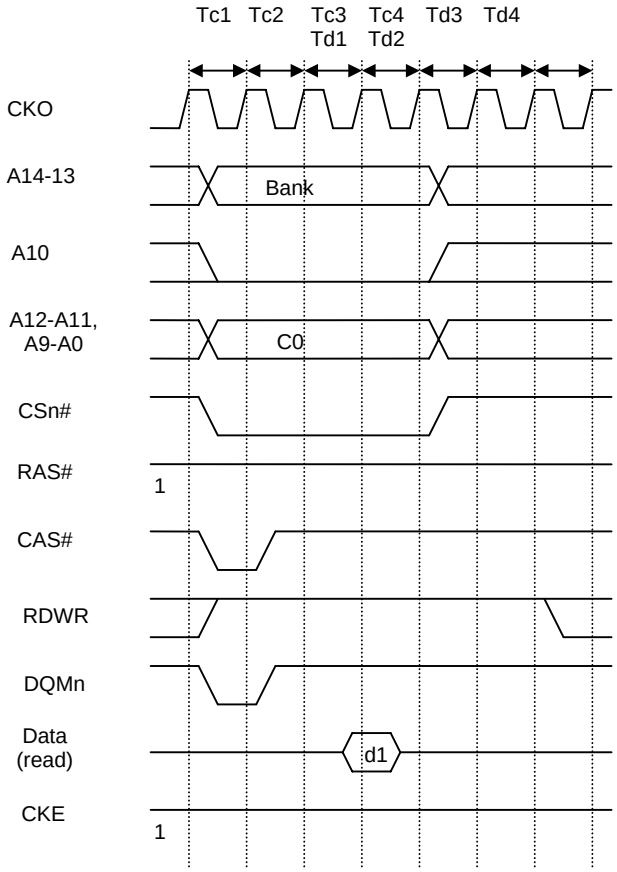
*DMCR: RCD = 1, TCL = 1, TPC = 2, TRWL = 1

Figure 1-27 Synchronous DRAM 8-beat Burst Write Timing (Different Row)



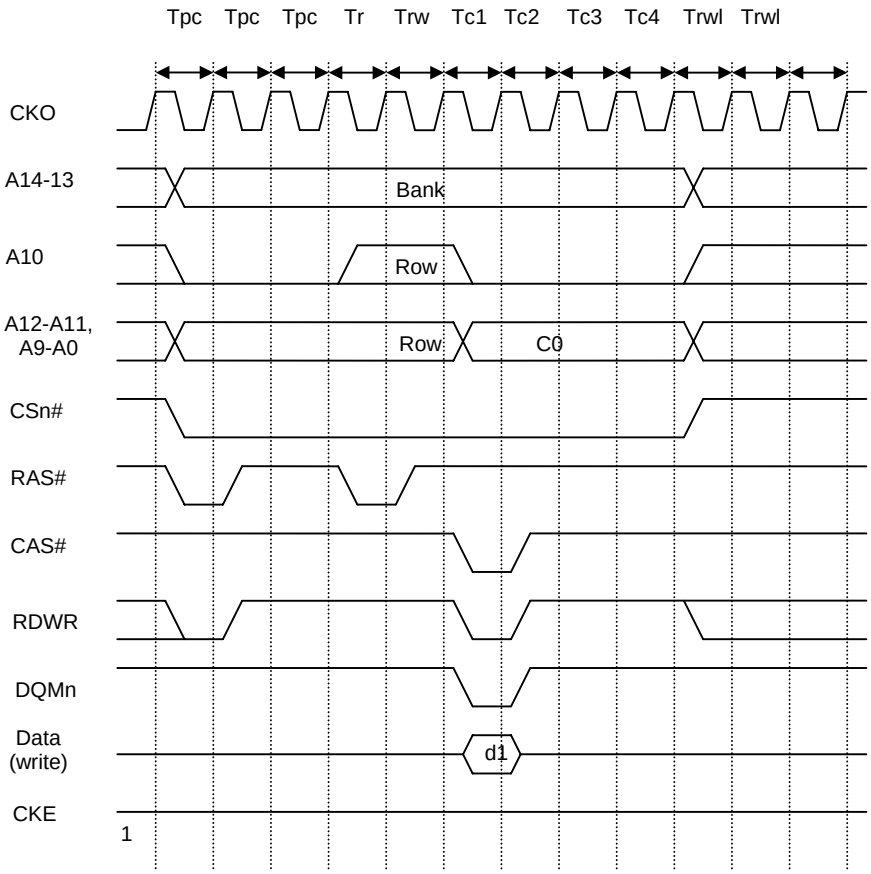
*DMCR: RCD = 1, TCL = 1, TPC = 2

Figure 1-28 Synchronous DRAM Single Read Timing (Different Row)



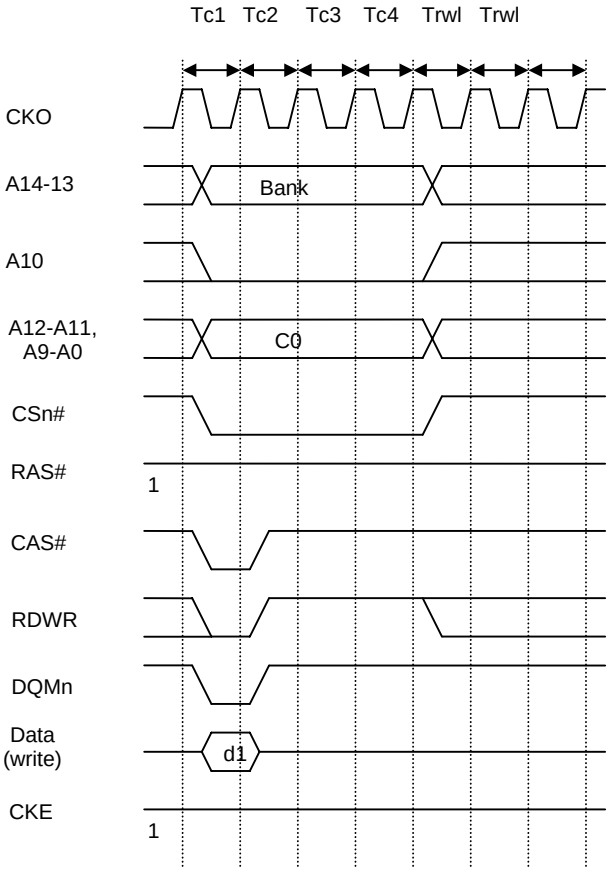
*DMCR: RCD = 1, TCL = 1, TPC = 2

Figure 1-29 Synchronous DRAM Single Read Timing (Same Row)



*DMCR: RCD = 1, TCL = 1, TPC = 2, TRWL = 1

Figure 1-30 Synchronous DRAM Single Write Timing (Different Row)



*DMCR: RCD = 1, TCL = 1, TPC = 2, TRWL = 1

Figure 1-31 Synchronous DRAM Single Write Timing (Same Row)

1.6.7 Power-Down Mode

The SDRAM power-down mode is supported to minimize the power consumption. CKE going to low level when SDRAM is idle/active state will drive SDRAM to precharge/active power-down mode. The clock supplies to SDRAM may be stopped also when CKE keep in low level more than two cycles. When a new access start or a refresh request, CKE is driven to high level and clock supplies is re-enabled. In power-down mode, clock of the accessed SDRAM bank pair is supplied. Clock of the other pair is stopped.

Following figures shows the timing of power-down mode and clock stopping.

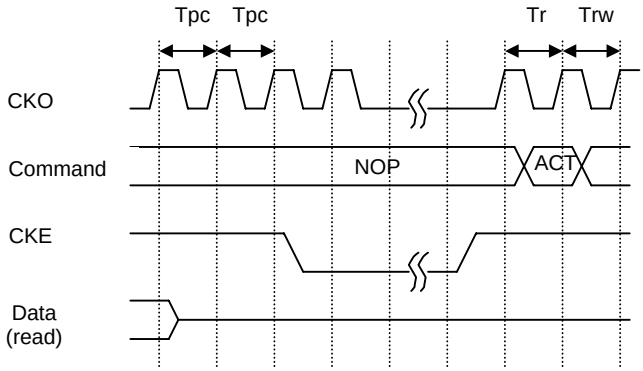


Figure 1-32 SDRAM Power-Down Mode Timing (CKO Stopped)

Following figure shows the power-down mode timing that CKE low level less than two cycles and clock is not stopped.

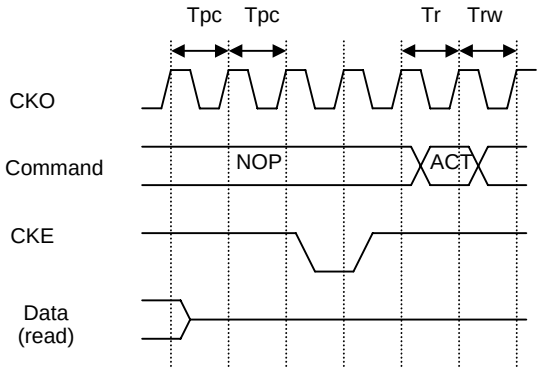


Figure 1-33 SDRAM Power-Down Mode Timing (Clock Supplied)

1.6.8 Refreshing

EMC provide a function for controlling the refresh of synchronous DRAM, Auto-refresh can be performed by clearing the RMODE bit to 0 and setting the RFSH bit to 1 in DMCR. If SDRAM is not accessed for a long period, self-refresh mode can be activated by set both the RMODE bit and the RFSH bit to 1.

1.6.8.1 AUTO-Refresh

Refreshing is performed at intervals determined by the input clock selected by bits CKS2-0 in RTCSR, and the value set in RTCOR. The value of bits CKS2-0 in RTCSR should be set so as to satisfy the refresh interval stipulation for the synchronous DRAM used. First make the settings for RTCOR, RTCNT, and the RMODE and RFSH bits in MCR, and then make the CKS2-CKS0 setting. When the clock is selected by CKS2-CKS0, RTCNT starts counting up from the value at that time. The RTCNT value is constantly compared with the RTCOR value, and if the two values are the same, a refresh request is generated and an auto-refresh is performed. At the same time, RTCNT is cleared to zero and the count-up is restarted. Figure 1-34 shows the auto-refresh cycle operation.

First, a REF command is issued in the TRr cycle. After the TRr cycle, new command output cannot be performed for the duration of the number of cycles specified by the TRC bits in DMCR. The TRC bits must be set so as to satisfy the synchronous DRAM refresh cycle time stipulation (active/active command delay time). Following figure shows the auto-refresh timing when TRC is set to 2.

Auto-refresh is performed in normal operation and sleep mode.

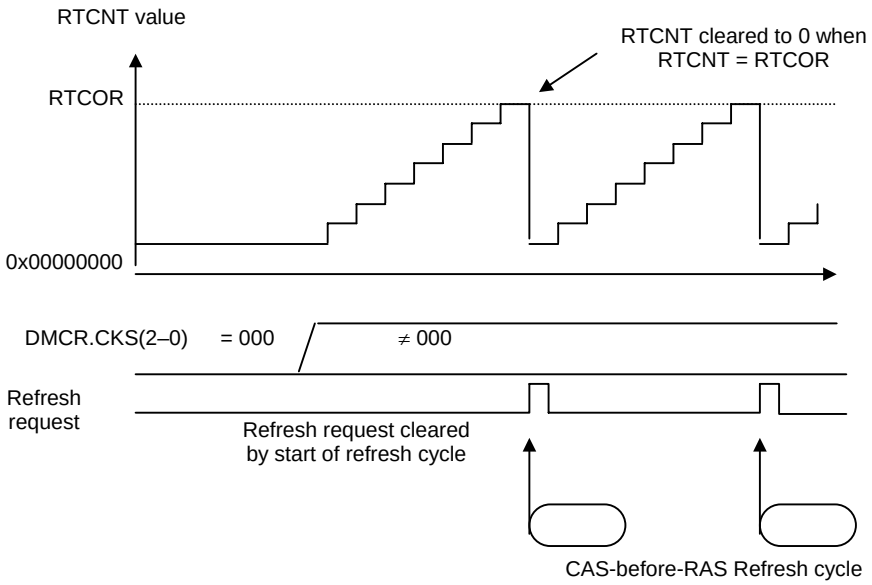


Figure 1-34 Synchronous DRAM Auto-Refresh Operation

A PALL command is issues firstly to precharge all banks. Then a REF command is issued in the TRr cycle.

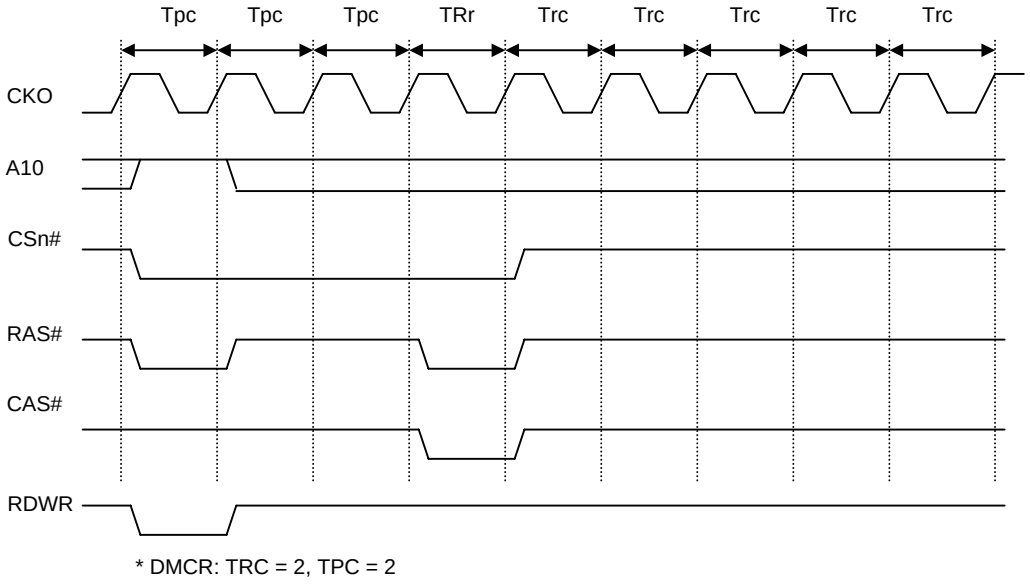


Figure 1-35 Synchronous DRAM Auto-Refresh Timing

1.6.8.2 SELF-Refresh

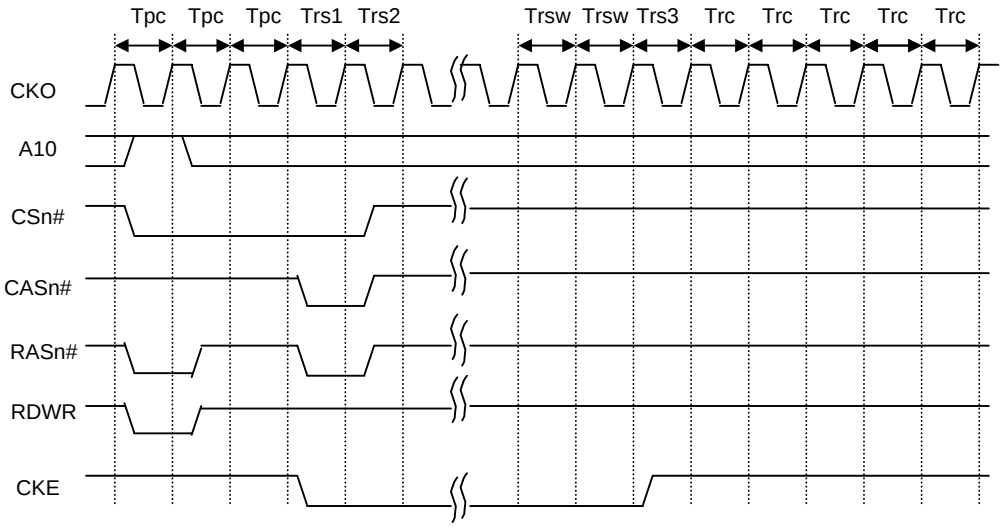
Self-refresh mode is a kind of sleep mode in which the refresh timing and refresh addresses are generated within the SDRAM. Self-refreshing is activated by setting both the RMODE bit and the RFSH bit to 1. The self-refresh state is maintained while the CKE signal is low. SDRAM cannot be accessed while in the self-refresh state. Self-refresh mode is cleared by clearing the RMODE bit to 0. After self-refresh mode has been cleared, command issuance is disabled for the number of cycles specified by the TRC bits in DMCR. Trsw cycles are inserted to meet the minimum CKE negation time specified by the TRAS bits in DMCR. Self-refresh timing is shown in following figure. Settings must be made so that self-refresh clearing and data retention are performed correctly, and auto-refresh is performed at the correct intervals. When self-refreshing is activated from the state in which auto-refreshing is set, or when exiting sleep mode other than through a reset, auto-refreshing is restarted if RFSH is set to 1 and RMODE is cleared to 0 when self-refresh mode is cleared. If the transition from clearing of self-refresh mode to the start of auto-refresh takes time, this time should be taken into consideration when setting the initial value of RTCNT. Making the RTCNT value 1 less than the RTCOR value will enable refreshing to be started immediately. After self-refreshing has been set, the self-refresh state continues even if the chip standby state is entered using the processor's sleep function, and is maintained even after recovery from sleep mode other than through a reset. In the case of a reset, the bus state controller's registers are initialized, and therefore the self-refresh state is cleared.

Self-refreshing is performed in normal operation, in idle mode and in sleep mode. In sleep mode, if RFSH bit in DMCR is 1, self-refresh is always performed in spite of RMODE field in DMCR until sleep mode is canceled.

Relationship between Refresh Requests and Bus Cycle Requests:

If a refresh request is generated during execution of a bus cycle, execution of the refresh is deferred until the bus cycle is completed. If a match between RTCNT and RTCOR occurs while a refresh is waiting to be executed, so that a new Refresh request is generated, the previous refresh request is eliminated. In order for refreshing to be performed normally, care must be taken to ensure that no bus cycle is longer than the refresh interval.

A PALL command is issued firstly to precharge all banks.



* DMCR: TRAS = 0, TRC = 2

Figure 1-36 Synchronous DRAM Self-Refresh Timing

1.6.9 Initialize Sequence

In order to use SDRAM, mode setting must first be performed after powering on. To perform SDRAM initialization correctly, the EMC registers must first be set, followed by a write to the SDRAM mode register.

In SDRAM mode register setting, the address signal value at that time is latched by MRS command. If the value to be set is X, the bus state controller provides for value X to be written to the synchronous DRAM mode register by performing a write to address offset 0xA000 + X for bank 0. In this operation the data is ignored, but the mode write is performed as a byte-size access. To set burst read/write, CAS latency 2 to 3, wrap type = sequential, and burst length 4 supported by the processor, arbitrary data is written in a byte-size access to the following addresses.

Table 1-13 SDRAM Mode Register Setting Address Example (32-bit)

	Bank 0			
CAS latency 2	A022	B088	C088	D088
CAS latency 3	A032	B0C8	C0C8	D0C8

Table 1-14 SDRAM Mode Register Setting Address Example (16-bit)

	Bank 0			
CAS latency 2	A011	B044	C044	D044
CAS latency 3	A019	B064	C064	D064

The value set in DMCR.MRSET is used to select whether a Pre-charge All Banks command (PALL) or a Mode Register Set command (MRS) is issued. The timing for the Pre-charge All Banks command is shown in Figure 1-37, and the timing for the Mode Register Set command in Figure 1-38

Before mode register setting, a 200 μs idle time (depending on the memory manufacturer) must be guaranteed after powering on requested by the synchronous DRAM. If the reset signal pulse width is greater than this idle time, there is no problem in performing initialize sequence immediately.

First, a pre-CHARGE all bank (PALL) command must be issued by performing a write to address offset 0xA000 + X for bank 0, while DMCR.MRSET = 0.

Next the NUMBER of dummy auto-refresh cycles specified by the manufacturer (usually 8) or more must be executed. This is usually achieved automatically while various kinds of initialization are being performed after auto-refresh setting, but a way of carrying this out more dependably is to set a short refresh request generation interval just while these dummy cycles are being executed. With

simple read or write access, the address counter in the synchronous DRAM used for auto-refreshing is not initialized, and so the cycle must always be an auto-refresh cycle.

After auto-REFRESH has been executed at least the prescribed number of times, a Mode Register Set command (MRS) is issued in the TMw1 cycle by setting DMCR.MRSET to 1 and performing a write to address offset 0xA000 + X.

An example of SDRAM operation flow is as the following:

1. Disable Bus release
   Write 0x00000000 to BCR
2. Initialize RTCOR and RTCNT for auto-refresh cycle
   Before configure SDRAM SDMR, SDRAM needs to execute auto-refresh, the number of times depends on the type of SDRAM. It's better to set a short refresh request generation interval here. For example, set RTCOR to 0x0000000F, and set RTCNT 0x00000000.
3. Initialize DMCR for Precharge all bank and auto-refresh
   When DMCR.RMODE=0 and DMCR.RFSH=1, enter auto-refresh mode;
   When DMCR.MRSET=0, write SDMR whill generates Precharge all bank cycle.
   DMCR.TPC must be defined for precharge.
4. Disable refresh counter clock
   Write 0x00000000 to RTCSR
5. Execute Precharge all bank before auto-refresh
   Because DMCR.MRSET=0, writing SDMR generates a Precharge all bank cycle, for example, write address (0x1301A000).
6. Enable fast refresh counter clock for auto-refresh cycle
   For example, write 0x00000001 to RTCSR
7. Wait for number of auto-refresh cycles (defined by SDRAM chip)
   When RTCSR.CMF=1, it indicates value of RTCOR and RTCNT match and an auto-refresh cycle occurs.
8. Configure DMCR for SDRAM MODE Register Set
   When DMCR.MRSET=1, write SDMR generate MRSET cycle.
   For example, write 0x059A5231 to DMCR, so that:
   Bus-width: 32-bit; Column Address: 9-bit; Row Address: 12-bit; Auto-refresh mode; SDMR Set mode; 4-bank; etc..
9. SDRAM Mode Register Set
   Because DMCR.MRSET=1, for example, write address 0x1301A022 to configure SDMR as:
   Burst Length: 4 burst
   Burst Type: Sequential
   CAS Latency: 2
10. Set normal auto-refresh counter clock
    For example, write 0x00000005 to RTCSR
11. Then Read/Write SDRAM can be executed

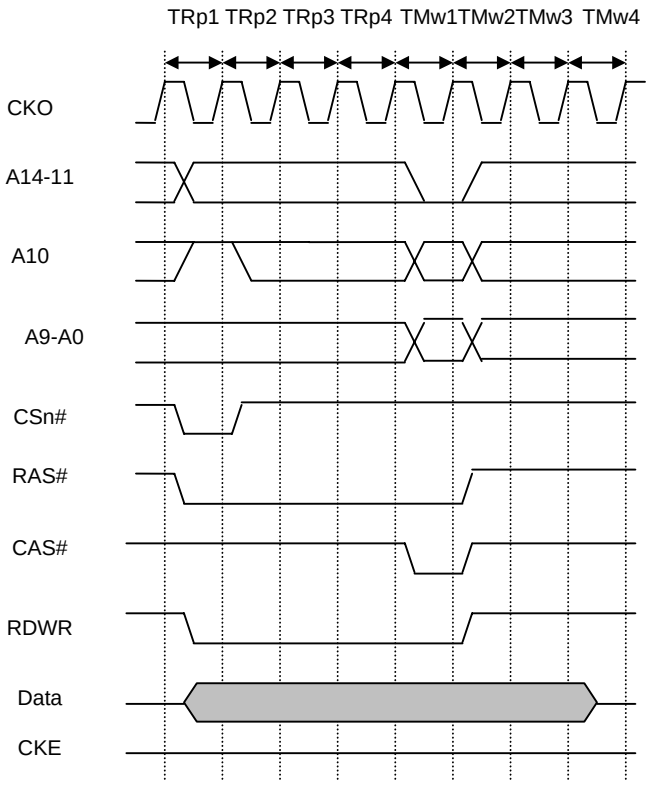


Figure 1-37 SDRAM Mode Register Write Timing 1 (Pre-charge All Banks)

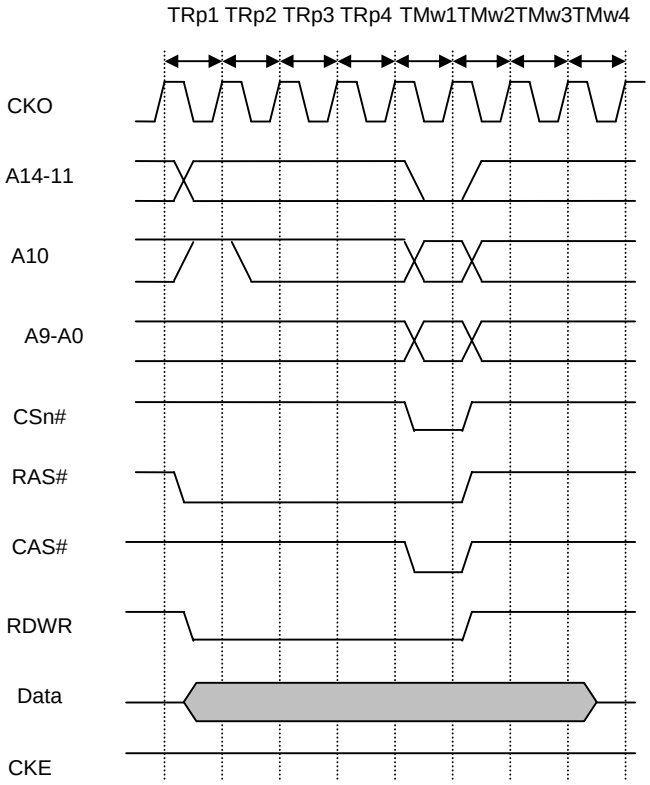


Figure 1-38 SDRAM Mode Register Write Timing 2 (Mode Register Set)

1.7 Bus Control Register (BCR)

BCR is used to specify the behavior of EMC on system bus and indicate the BOOT_SEL[1:0] status which defines the boot configure. It is initialized to 0x00000001 by any reset.

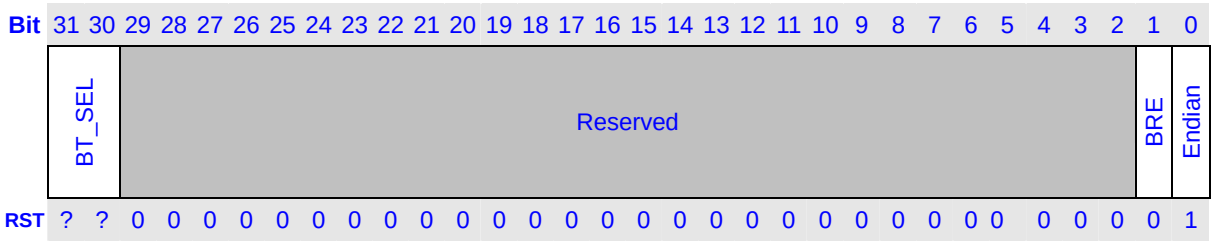
BOOT_SEL[1:0] pins define the boot time configurations as listed in the following table.

Table 1-15 Boot Configuration

Boot_sel[1]	Boot_sel[0]	Description
0	0	Boot from external ROM at CS3_
0	1	Boot from USB device
1	0	Boot from 512 Byte page NAND flash at CS1_
1	1	Boot from 2k Byte page NAND flash at CS1_

Name	Description	RW	Reset Value	Address	Access Width
BCR	Bus Control Register	RW	0x?0000001	0x13010000	32

BCR **0x13010000**



Bits	Name	Description	RW
31:30	BT_SEL	**BOOT_SEL (BT_SEL[1:0]):** Status of BOOT_SEL pins that indicate the boot configure. See the above boot configuration table.	R
29:2	Reserved	Writes to these bits has no effect and always read as 0.	R
1	BRE	**Bus Release Enable**: When clear, once a transaction to EMC begins on the system bus; it must be completed before another transaction starts. When set, the system bus may be released to allow other transaction before EMC prepare the read data or be able to receipt the write data. If slow memory devices are used in the system, setting this bit will improve the efficiency of the whole system. The efficiency of SDRAM access may be improved by setting this bit. But the power consumption is increased if this bit is set. **BRE Description**	RW

		0 The system bus can not be released during an access (Initial value) 1 The system bus can be released during an access	
0	Endian	**Endian:** Indicates the system is little-endian.	R