[Clock divide factor of on-chip clock divider]

Bitrate on layer I (MODE=0);

bitrate_index	bitrate (kHz)	clock divide 56.448MHz	49.152MHz
0001	32	-----	-----
0010	64	882	768
0011	96	588	512
0100	128	441	384
0101	160	-----	-----
0110	192	294	256
0111	224	{252}	-----
1000	256	220.5	192
1001	288	-----	-----
1010	320	-----	-----
1011	352	-----	-----
1100	384	147	128
1101	416	-----	-----
1110	448	-----	-----

Bitrate on layer II (MODE=0);

bitrate_index	bitrate (kHz)	clock divide 56.448MHz	49.152MHz
0001	32	-----	-----
0010	48	-----	-----
0011	56	-----	-----
0100	64	882	768
0101	80	-----	-----
0110	96	588	512
0111	112	{504}	-----
1000	128	441	384
1001	160	-----	-----
1010	192	294	256
1011	224	{252}	-----
1100	256	220.5	192
1101	320	-----	-----
1110	384	147	128

Sampling frequency (MODE=0)

sampling_frequency	fs (kHz)	256*fs (MHz)	clock divide 56.448MHz	49.152MHz
00	44.1	11.290	5	----
01	48	12.288	-----	4
10	32	8.192	-----	6

Bitrate on layer I (MODE=1);

bitrate_index	bitrate (kHz)	clock divide 24.576MHz	22.5792MHz	16.384MHz
0001	32	768	(705.6)	512
0010	64	384	(352.8)	256
0011	96	256	(235.2)	(170.7)
0100	128	192	(176.4)	128
0101	160	(153.6)	(141.1)	(102.4)
0110	192	128	(117.6)	(85.3)
0111	224	(109.7)	(100.8)	(73.1)
1000	256	96	(88.2)	64
1001	288	(85.3)	(78.4)	(56.9)
1010	320	(76.8)	(70.6)	(51.2)
1011	352	(69.8)	(64.1)	(46.5)
1100	384	64	(58.8)	(42.7)
1101	416	(59.1)	(54.3)	(39.4)
1110	448	(54.9)	(50.4)	(36.6)

Bitrate on layer II (MODE=1);

bitrate_index	bitrate (kHz)	clock divide 24.576MHz	22.5792MHz	16.384MHz
0001	32	768	(705.6)	512
0010	48	512	(470.4)	(341.3)
0011	56	(438.9)	(403.2)	(292.6)
0100	64	384	(352.8)	256
0101	80	(307.2)	(282.2)	(204.8)
0110	96	256	(235.2)	(170.7)
0111	112	(219.4)	(201.6)	(146.3)
1000	128	192	(176.4)	128
1001	160	192	(176.4)	128
1010	192	128	(117.6)	(85.3)
1011	224	(109.7)	(100.8)	(73.1)
1100	256	96	(88.2)	64
1101	320	(76.8)	(70.6)	(51.2)
1110	384	64	(58.8)	(42.7)

Sampling frequency (MODE=1)

sampling_ frequency	fs (kHz)	256*fs (MHz)	clock divide 24.576MHz	22.5792MHz	16.384MHz
00	44.1	11.290	-----	2	-----
01	48	12.288	2	-----	-----
10	32	8.192	-----	-----	2

Bitrate on layer I (MODE=1);

bitrate_index	bitrate (kHz)	R_CLK (kHz) 24.576MHz	22.5792MHz	16.384MHz
0001	32	438.857	403.2	292.57
0010	64	438.857	403.2	292.57
0011	96	438.857	403.2	292.57
0100	128	438.857	403.2	292.57
0101	160	438.857	403.2	292.57
0110	192	438.857	403.2	292.57
0111	224	438.857	403.2	292.57
1000	256	438.857	403.2	292.57
1001	288	438.857	403.2	292.57
1010	320	438.857	403.2	292.57*
1011	352	438.857	403.2	292.57*
1100	384	438.857	403.2	292.57*
1101	416	438.857	403.2*	292.57*
1110	448	438.857*	403.2*	292.57*

Bitrate on layer II (MODE=1);

bitrate_index	bitrate (kHz)	R_CLK (kHz) 24.576MHz	22.5792MHz	16.384MHz
0001	32	438.857	403.2	292.57
0010	48	438.857	403.2	292.57
0011	56	438.857	403.2	292.57
0100	64	438.857	403.2	292.57
0101	80	438.857	403.2	292.57
0110	96	438.857	403.2	292.57
0111	112	438.857	403.2	292.57
1000	128	438.857	403.2	292.57
1001	160	438.857	403.2	292.57
1010	192	438.857	403.2	292.57
1011	224	438.857	403.2	292.57
1100	256	438.857	403.2	292.57
1101	320	438.857	403.2	292.57*
1110	384	438.857	403.2	292.57*

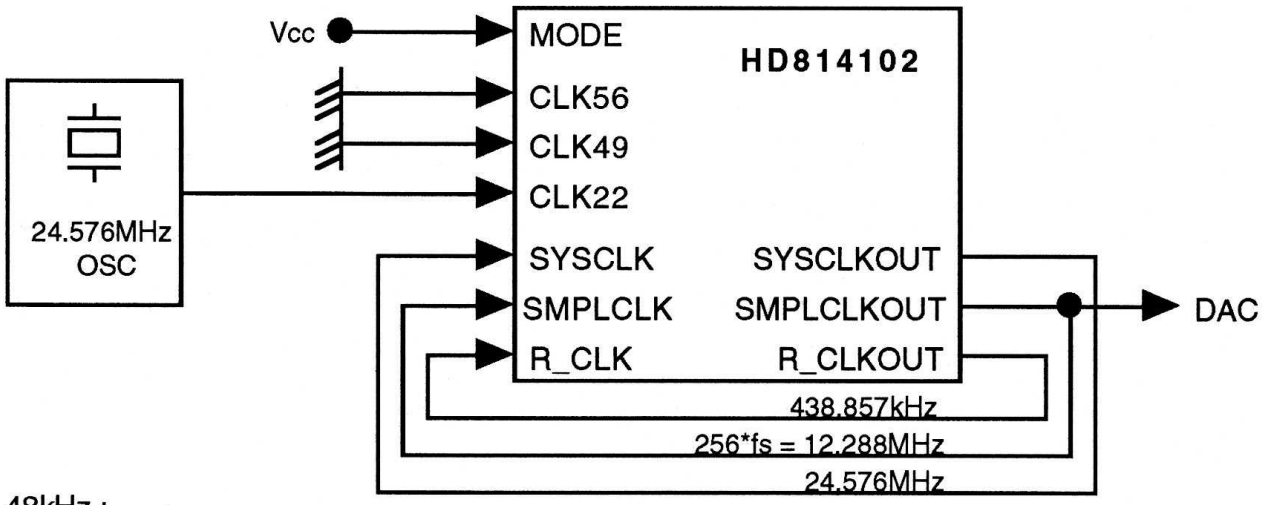
R_CLK = CLK22/56

* : External 4*CLK22 clock generator provides ;
SYSCLK = 4*CLK22/3, CLK22 = 4*CLK22/4

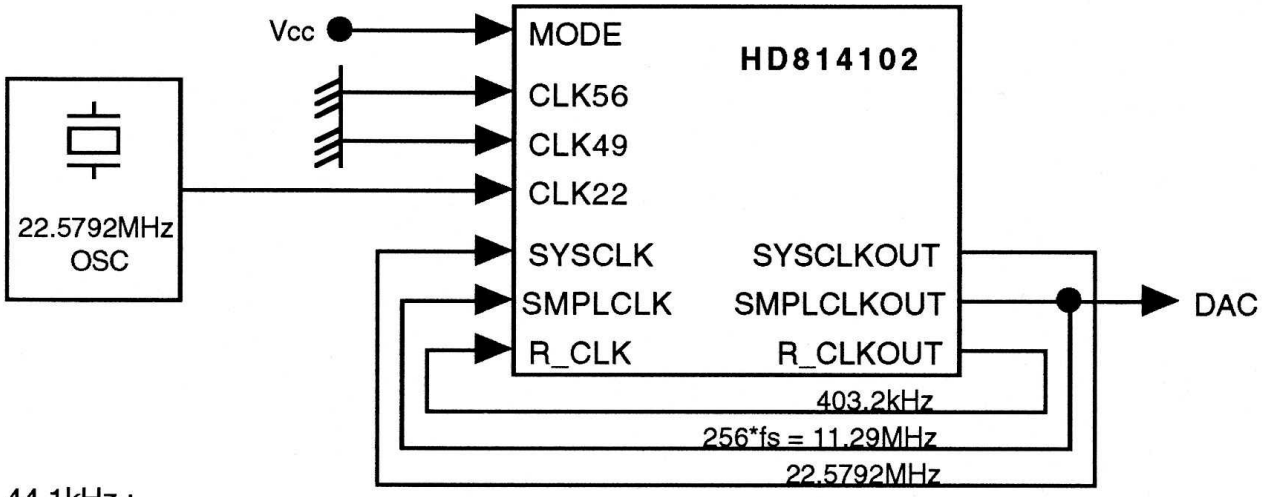
none : Only external CLK22 clock generator is needed.

Sampling frequency (MODE=1)

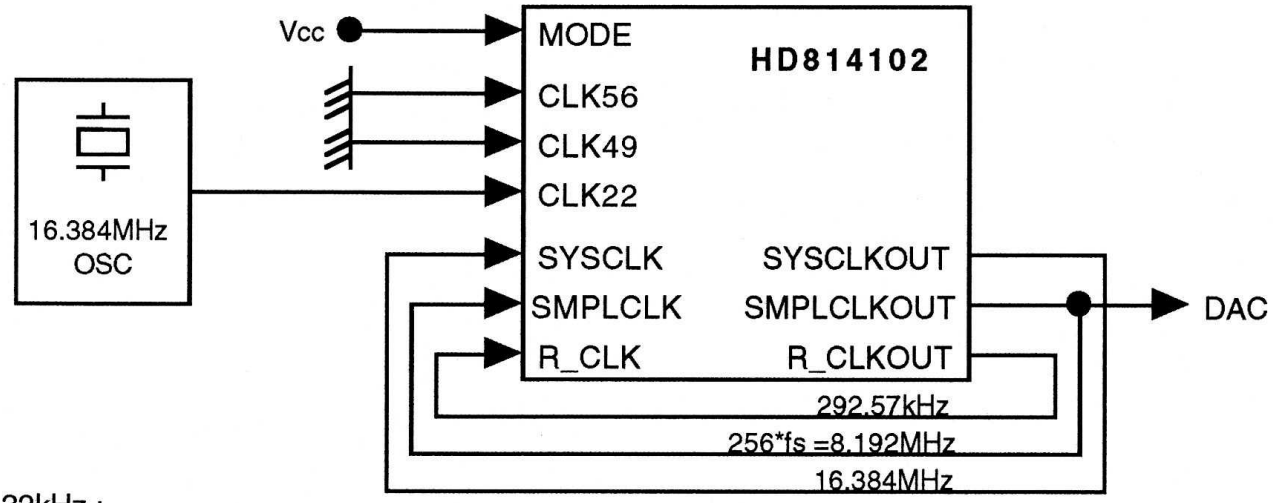
sampling_ frequency	fs (kHz)	256*fs (MHz)	clock divide 24.576MHz	22.5792MHz	16.384MHz
00	44.1	11.290	-----	2	-----
01	48	12.288	2	-----	-----
10	32	8.192	-----	-----	2



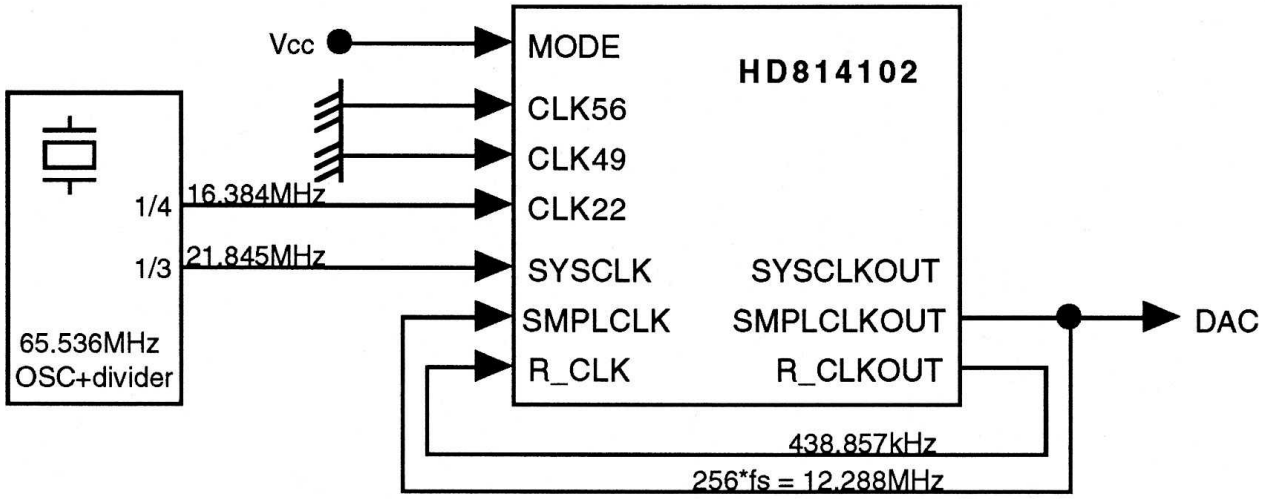
fs = 48kHz ;
Layer I ; 32/64/96/128/160/192/224/256/288/320/352/384/416 kHz bit rate
Layer II ; 32/48/56/64/80/96/112/128/160/192/224/256/320/384 kHz bit rate



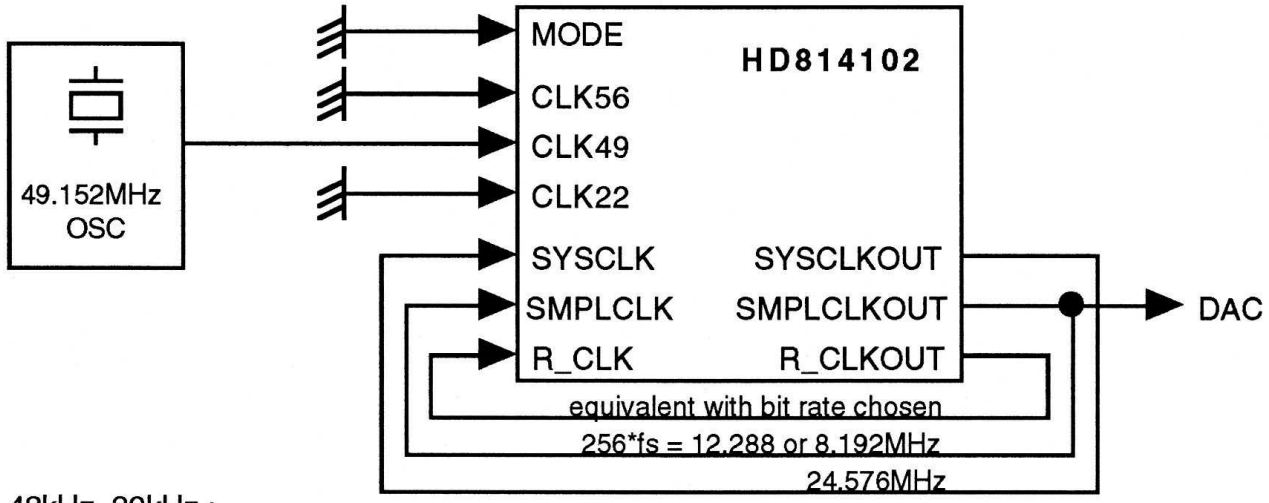
fs = 44.1kHz ;
Layer I ; 32/64/96/128/160/192/224/256/288/320/352/384 kHz bit rate
Layer II ; 32/48/56/64/80/96/112/128/160/192/224/256/320/384 kHz bit rate



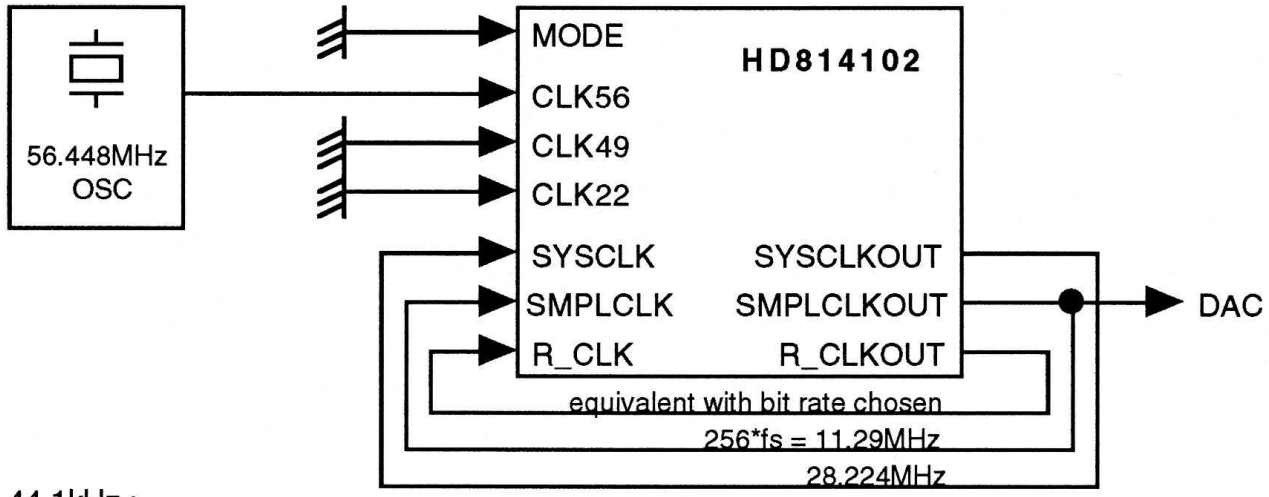
fs = 32kHz ;
Layer I ; 32/64/96/128/160/192/224/256/288 kHz bit rate
Layer II ; 32/48/56/64/80/96/112/128/160/192/224/256 kHz bit rate



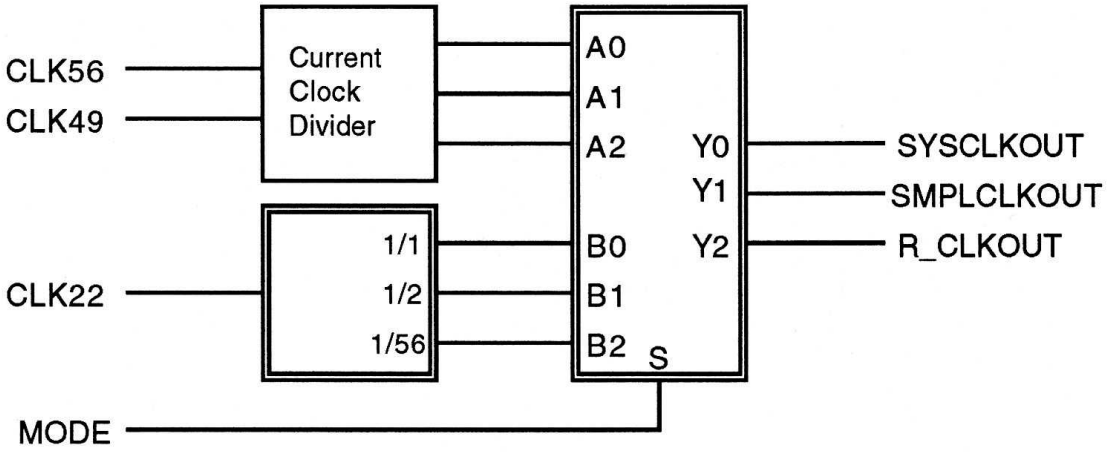
fs = 32kHz ;
Layer I ; 320/352/384/416/448 kHz bit rate
Layer II ; 320/384 kHz bit rate



fs = 48kHz, 32kHz ;
Layer I ; 64/96/128/192/224/256/384 kHz bit rate
Layer II ;64/96/112/128/192/224/256/384 kHz bit rate



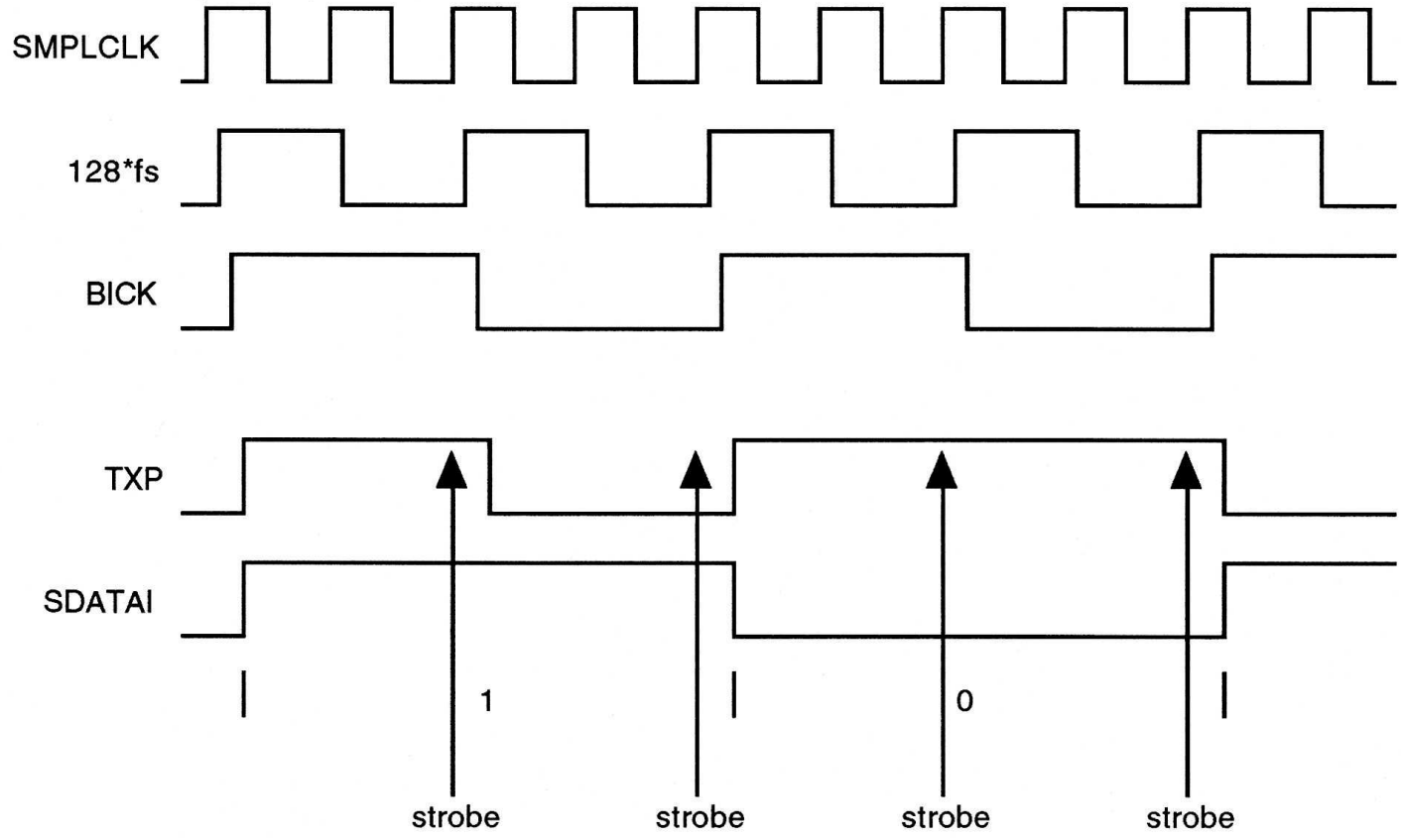
fs = 44.1kHz ;
Layer I ; 64/96/128/192/224/256/384 kHz bit rate
Layer II ; 64/96/112/128/192/224/256/384 kHz bit rate



Current Clock Divide supports ;
Layer I ; 64/96/128/192/224/256/384 kbps
Layer II ; 64/96/112/128/192/224/256/384 kbps

Block newly implemented

Logic change of Clock Divider Section



System level simulation strobe position

```
          111111111122222
12345678901234567890123 4

              S T S B L O
              Y X D I R D
              N P A C C V
              C   T K L L
              E   A   L D
              D

nnnnnnnnnnn   1 1 1 0 0 1
nnnnnnnnnnn   1 0 1 1 1 1
nnnnnnnnnnn   1 0 0 0 1 1
nnnnnnnnnnn   1 1 0 1 1 1
nnnnnnnnnnn   1 0 1 0 1 1
```

[Sysclk/Smplclk/Rdclk frequency on Compass simulation]

fs.mode.bitrate	sysclk (nsec)	smplclk (nsec)	rdclk (nsec)
32.m.64	40.69 - **40**	122.07 - **120**	15625 - **15600**
32.m.96	40.69 - **40**	122.07 - **120**	10416.7 - **10400**
32.m/dc/s/js.128	40.69 - **40**	122.07 - **120**	7812.5 - **7800**
32.m/dc/s/js.192	40.69 - **40**	122.07 - **120**	5208.3 - **5200**
32.m/dc/s/js.224	40.69 - **40**	122.07 - **120**	4464.3 - **4440**
32.dc/s/js.256	40.69 - **40**	122.07 - **120**	3906.3 - **3880**
32.dc/s/js.384	40.69 - **40**	122.07 - **120**	2604.2 - **2600**
44.m.64	35.43 - **44**	88.58 - **88**	15625 - **15620**
44.m.96	35.43 - **44**	88.58 - **88**	10416.7 - **10384**
44.m/dc/s/js.128	35.43 - **44**	88.58 - **88**	7812.5 - **7788**
44.m/dc/s/js.192	35.43 - **44**	88.58 - **88**	5208.3 - **5192**
44.m/dc/s/js.224	35.43 - **44**	88.58 - **88**	4464.3 - **4488**
44.dc/s/js.256	35.43 - **44**	88.58 - **88**	3906.3 - **3872**
44.dc/s/js.384	35.43 - **44**	88.58 - **88**	2604.2 - **2596**
48.m.64	40.69 - **40**	81.38 - **80**	15625 - **15600**
48.m.96	40.69 - **40**	81.38 - **80**	10416.7 - **10400**
48.m/dc/s/js.128	40.69 - **40**	81.38 - **80**	7812.5 - **7800**
48.m/dc/s/js.192	40.69 - **40**	81.38 - **80**	5208.3 - **5200**
48.m/dc/s/js.224	40.69 - **40**	81.38 - **80**	4464.3 - **4440**
48.dc/s/js.256	40.69 - **40**	81.38 - **80**	3906.3 - **3880**
48.dc/s/js.384	40.69 - **40**	81.38 - **80**	2604.2 - **2600**

Wait for BIST ;
1.474 msec at 40 nsec SYSCLK

Ex. ste44m164.sim ; 175
ste48s2384.sim ; 566
ste32d2224.sim ; 332

[Status flag read]

RD*	CS*	A1	A0	BHE*	Data format	RDHL	RDHB*	RDLB*
0	0	1	0	0	Even address word	0	0	0
0	0	1	0	1	Even address byte	0	1	0
0	0	1	1	X	Odd address byte	1	1	0

[Control data write]

WR*	CS*	A1	A0	BHE*	Data format	WRLH	WRHB*	WRLB*
0	0	1	0	0	Even address word	0	0	0
0	0	1	0	1	Even address byte	0	1	0
0	0	1	1	X	Odd address byte	1	0	1

Compressed data interface

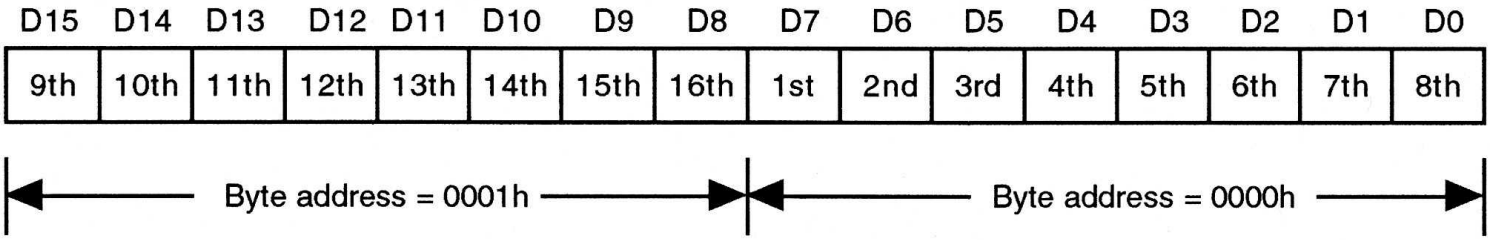
(1) Bit position inside the byte

The first bit must be positioned at MSB and the eighth bit must be at LSB as shown below.

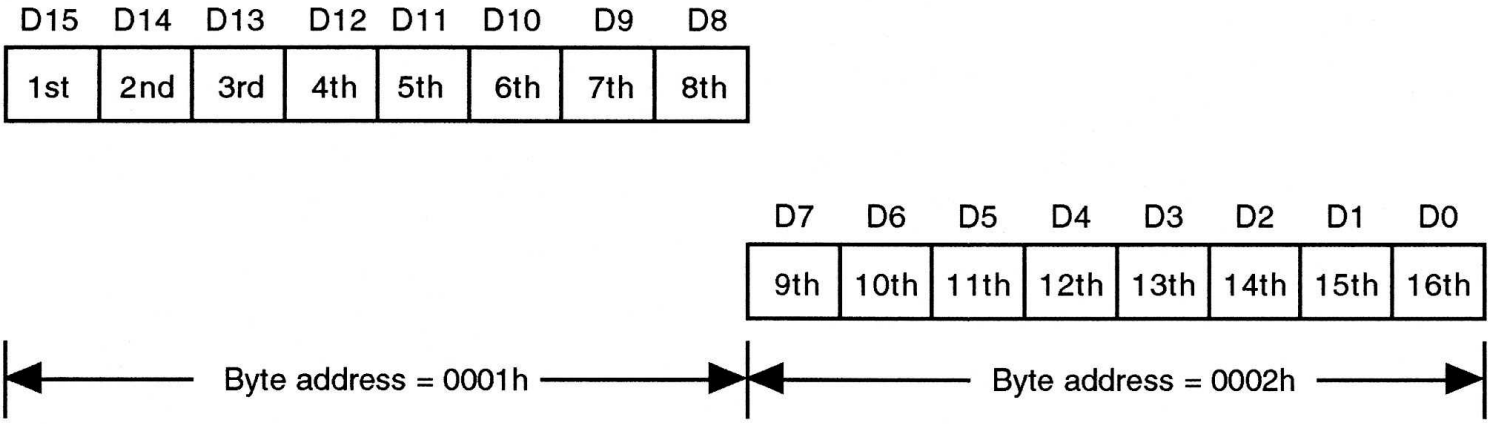
(2) Relationship between byte and word

The first byte must be positioned at lower byte address than the second byte. The second byte must be stored at next byte address of the first byte as shown below.

Word address = 0000h



Word address = 0001h



Data register write sequence (Word)

FH	FL	FH2	FL2	OBFULL	WR*	WR_1d*	sckmsk*	sck	DRQGO
0	0	0	0	0	1	1	0	x	0
				1					
									1
0->0->1	0->0->1				1->0->1				1->0->0
1->1->0	1->1->0	0->0->1	0->0->1		1	1->0->1	0->0->1		0->0->1
0	0	1->0->0	1					0->1->0	1
0->0->1	0->0->1	0			1->0->1				1->0->0
			1->0->0		1			0->1->0	0
1->1->0	1->1->0	0->0->1	0->0->1			1->0->1			0->0->1
0	0	1->0->0	1					0->1->0	1

Data register write sequence (Low byte)

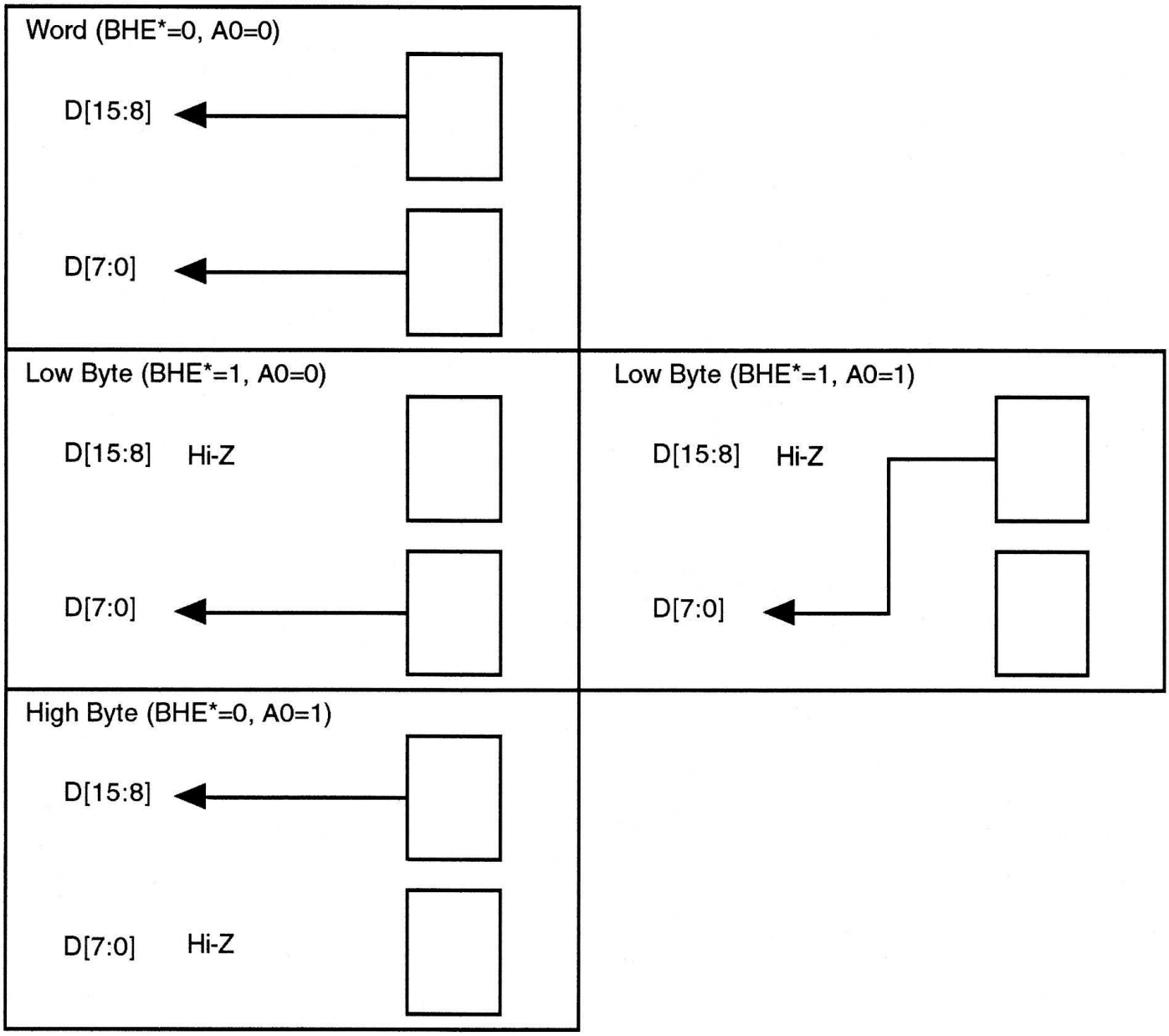
FH	FL	FH2	FL2	OBFULL	WR*	WR_1d*	sckmsk*	sck	DRQGO
0	0	0	0	0	1	1	0	x	0
				1					
									1
	0->0->1				1->0->1				1->0->0
	1->1->0		0->0->1		1	1->0->1	0->0->1		0->0->1
	0		1->0->0					0->1->0	1
	0->0->1				1->0->1				1->0->0
	1->1->0		0->0->1		1	1->0->1			0->0->1
	0		1->0->0					0->1->0	1

Data register write sequence (High byte)

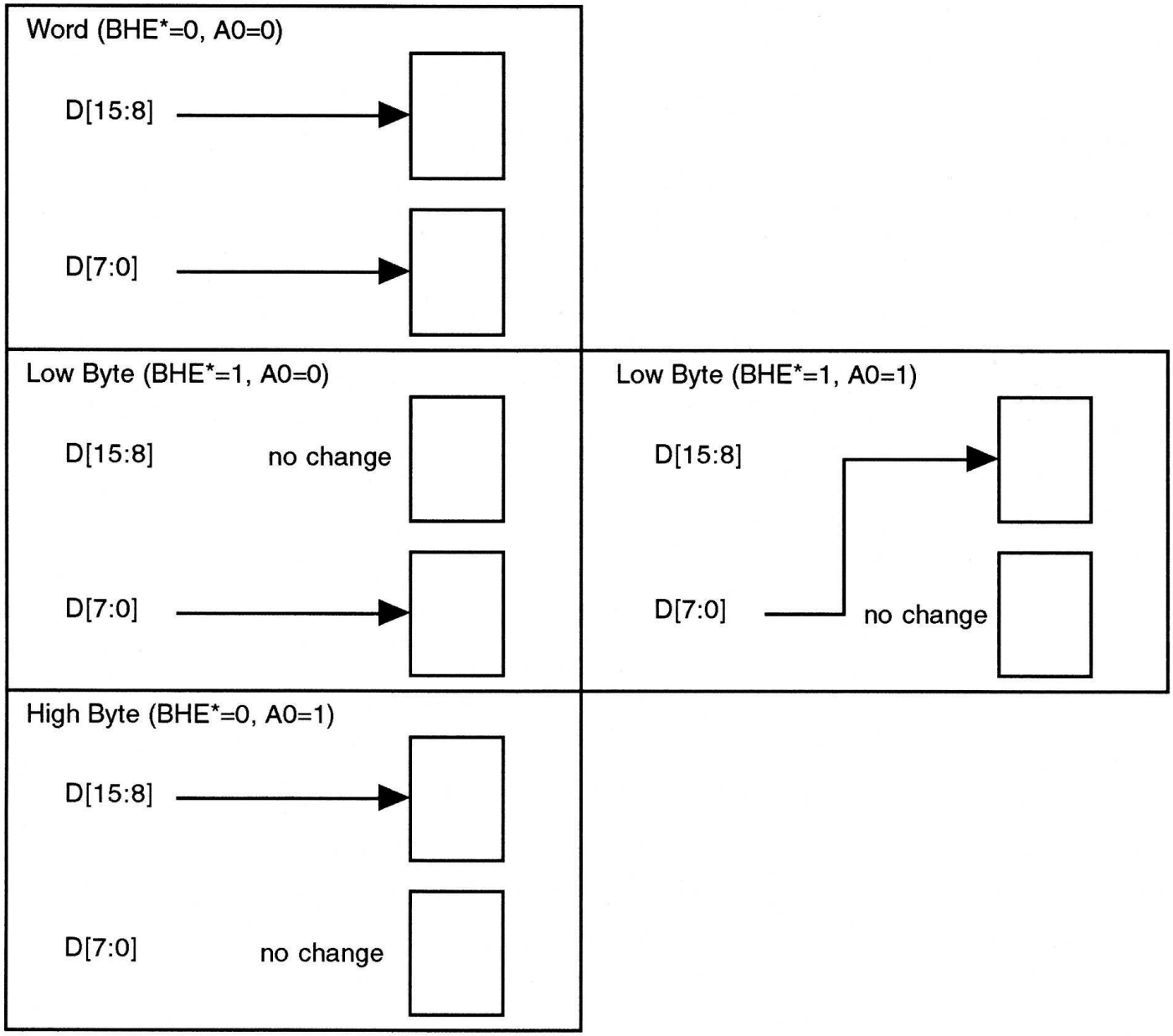
FH	FL	FH2	FL2	OBFULL	WR*	WR_1d*	sckmsk*	sck	DRQGO
0	0	0	0	0	1	1	0	x	0
				1					
									1
0->0->1					1->0->1				1->0->0
1->1->0		0->0->1			1	1->0->1	0->0->1		0->0->1
0		1->0->0						0->1->0	1
0->0->1					1->0->1				1->0->0
1->1->0		0->0->1			1	1->0->1			0->0->1
0		1->0->0						0->1->0	1

CPU interface

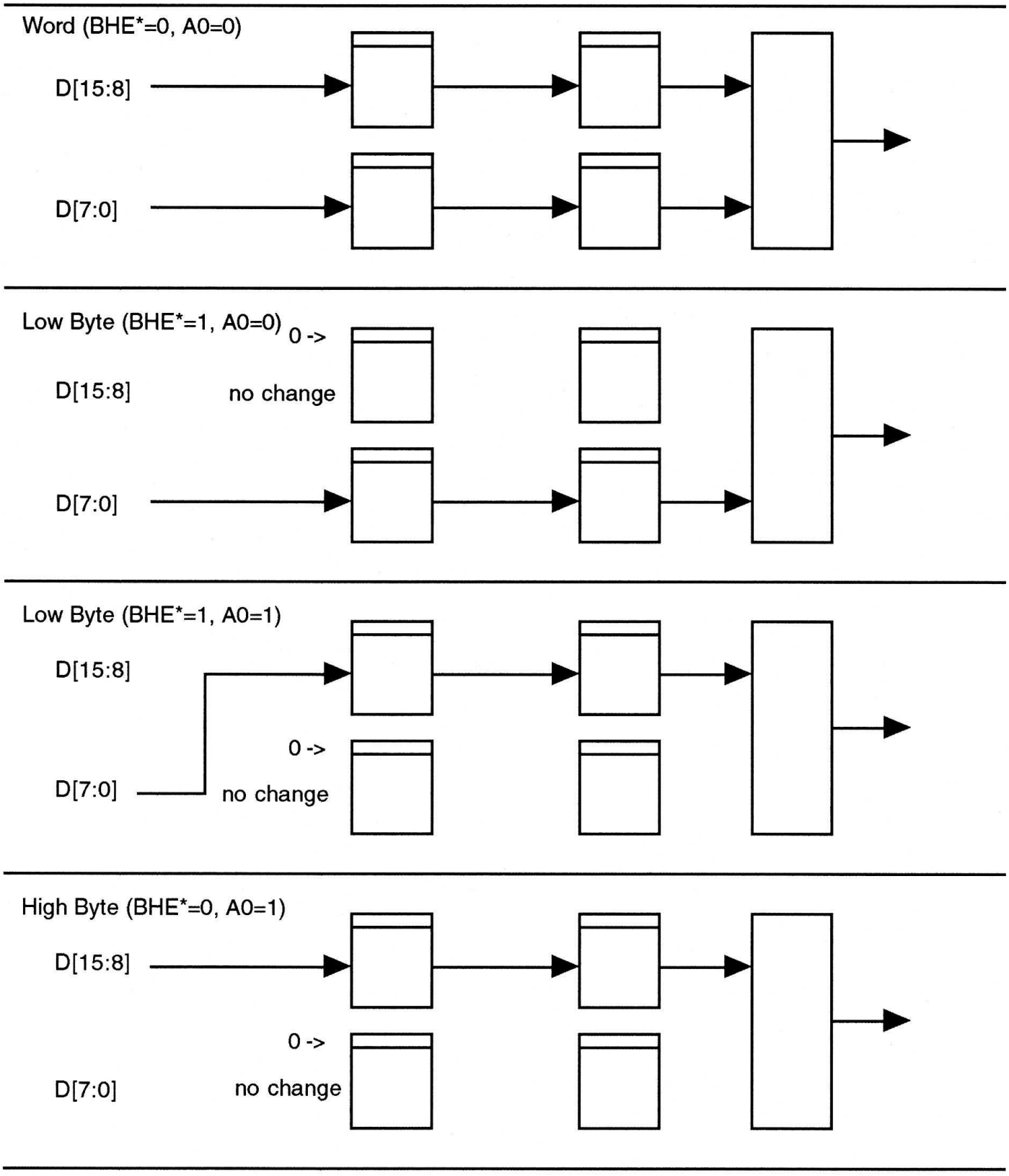
Status flag read (A1=1, CS*=0, RD*=0)



Control register write (A1=1, CS*=0, WR*=0)

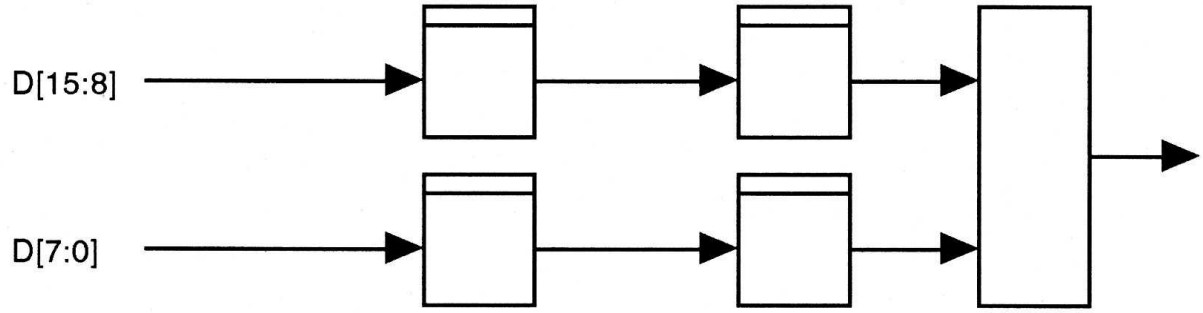


Data register write (A1=0, CS*=0, WR*=0)

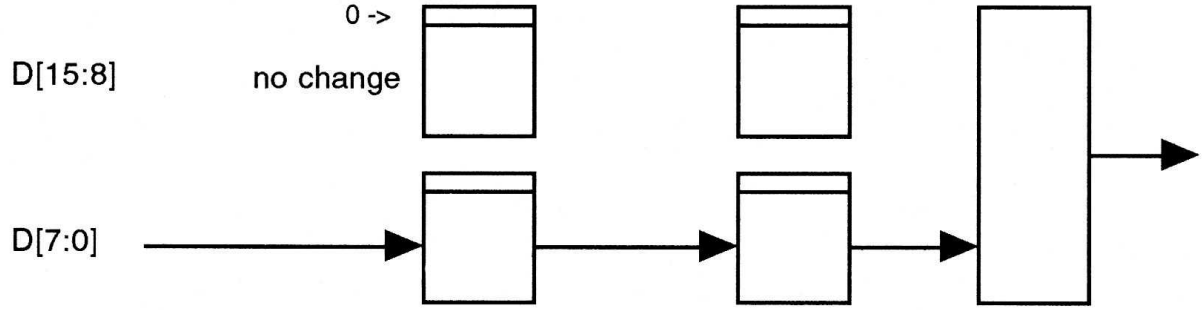


Data register write by DMA (DACK*=0, WR*=0)

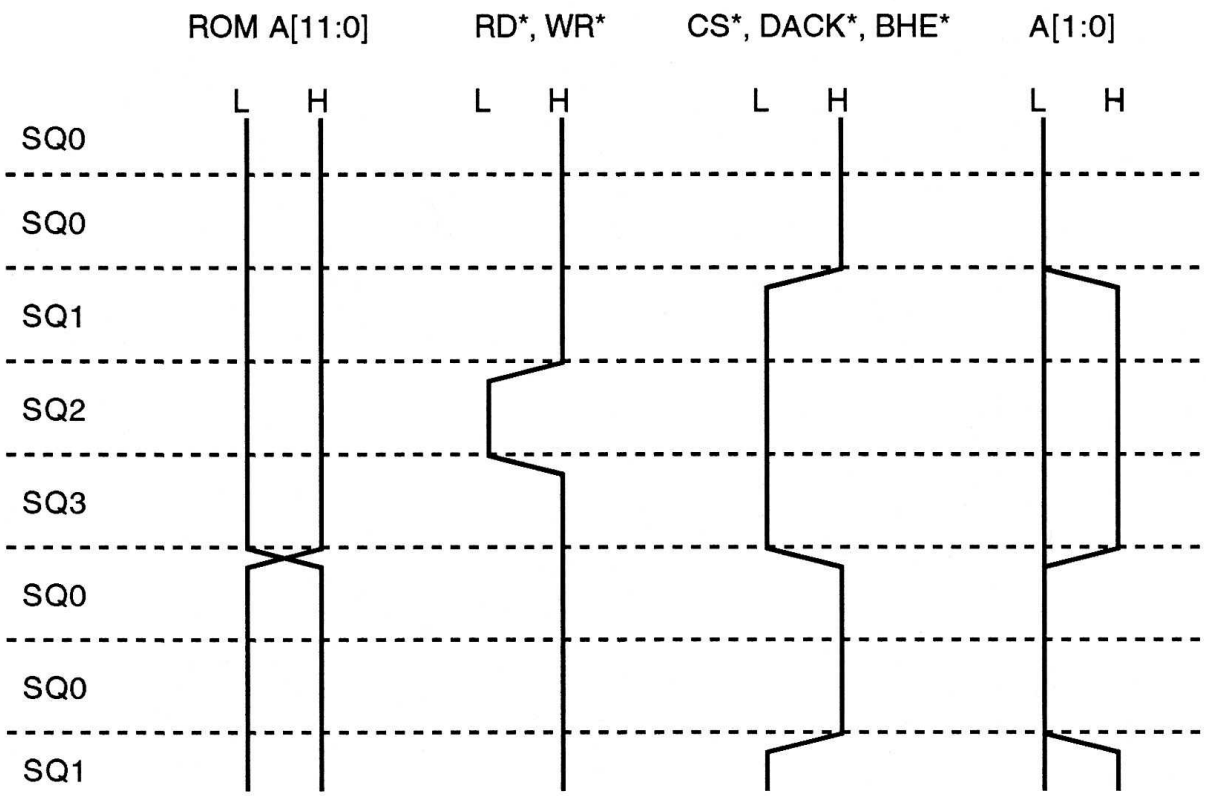
Word (DMA16=1)

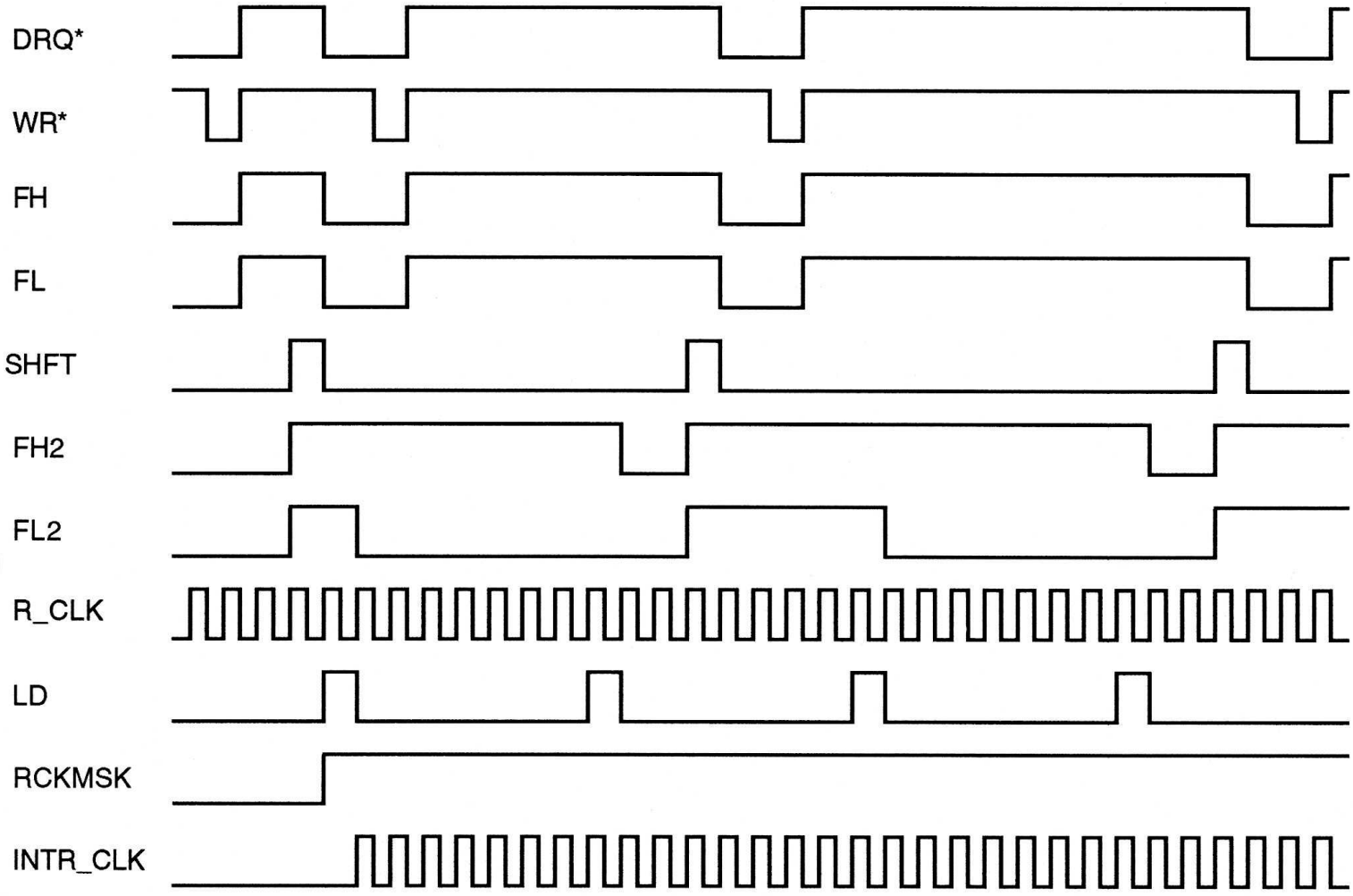


Low Byte (DMA16=0)

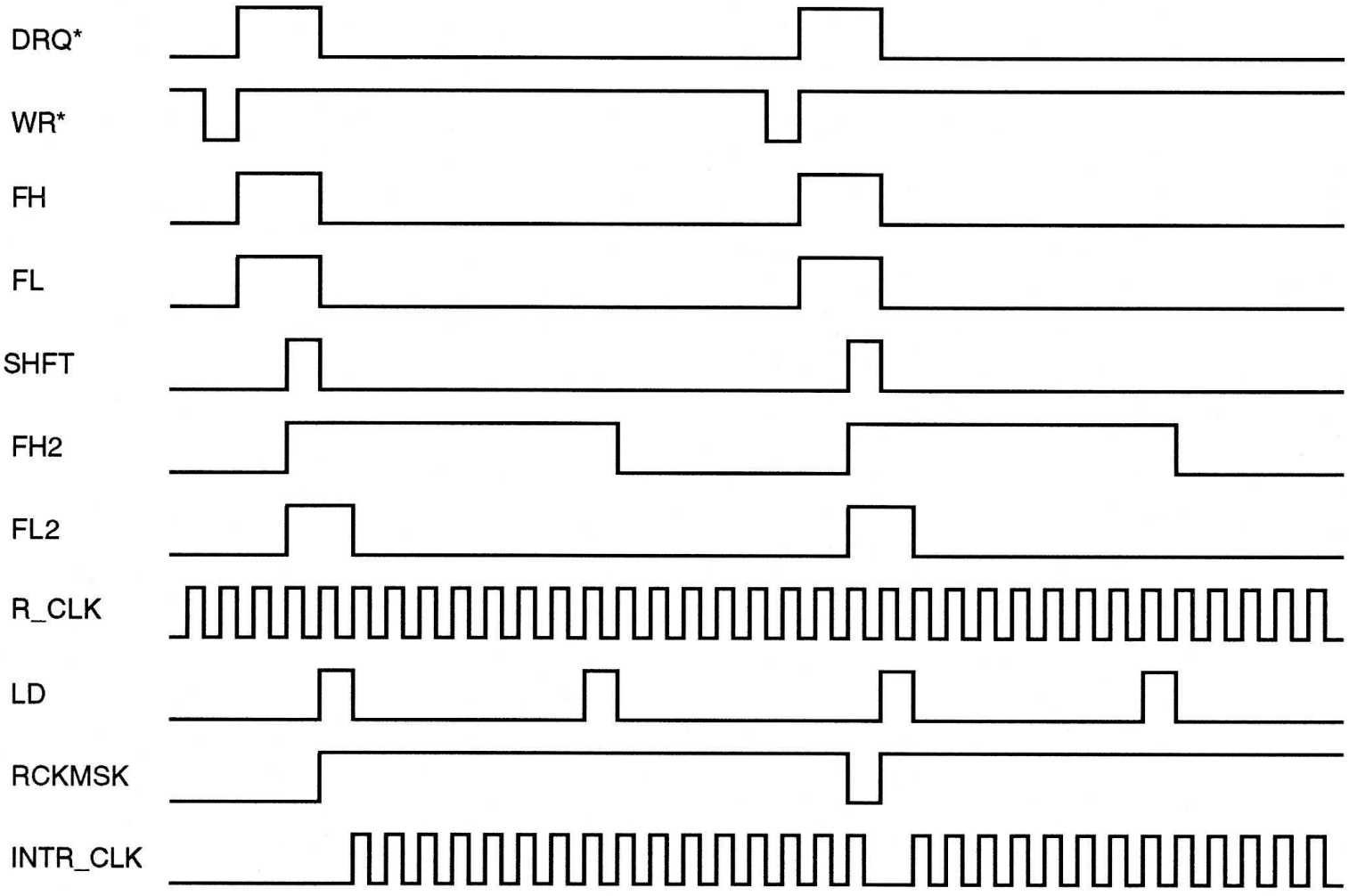


	D23	D22	D21	D20	D19	D18	D17	D16	D15 D8	D7 D0
		DACK*	RD*	WR*	CS*	BHE*	A1	A0		
No operation	0	1	1	1	1	1	0	0	00h	00h
		DACK*	RD*	WR*	CS*	BHE*	A1	A0		
Data write (word)	0	1	1	0	0	0	0	0	D[15:8]	D[7:0]
Data write (low byte)	0	1	1	0	0	1	0	0	00h	D[7:0]
Data write (low byte)	0	1	1	0	0	1	0	1	00h	D[7:0]
Data write (high byte)	0	1	1	0	0	0	0	1	D[15:8]	00h
		DACK*	RD*	WR*	CS*	BHE*	A1	A0		
Status flag read (word)	1	1	0	1	0	0	1	0	00h	00h
Status flag read (low byte)	1	1	0	1	0	1	1	0	00h	00h
Status flag read (low byte)	1	1	0	1	0	1	1	1	00h	00h
Status flag read (high byte)	1	1	0	1	0	0	1	1	00h	00h
		DACK*	RD*	WR*	CS*	BHE*	A1	A0		
Control write (word)	1	1	1	0	0	0	1	0	D[15:8]	D[7:0]
Control write (low byte)	1	1	1	0	0	1	1	0	00h	D[7:0]
Control write (low byte)	1	1	1	0	0	1	1	1	00h	D[7:0]
Control write (high byte)	1	1	1	0	0	0	1	1	D[15:8]	00h
		DACK*	RD*	WR*	CS*	BHE*	A1	A0		
Data write by DMA (word)	0	0	X	0	X	X	X	X	D[15:8]	D[7:0]
Data write by DMA (low byte)	0	0	X	0	X	X	X	X	00h	D[7:0]

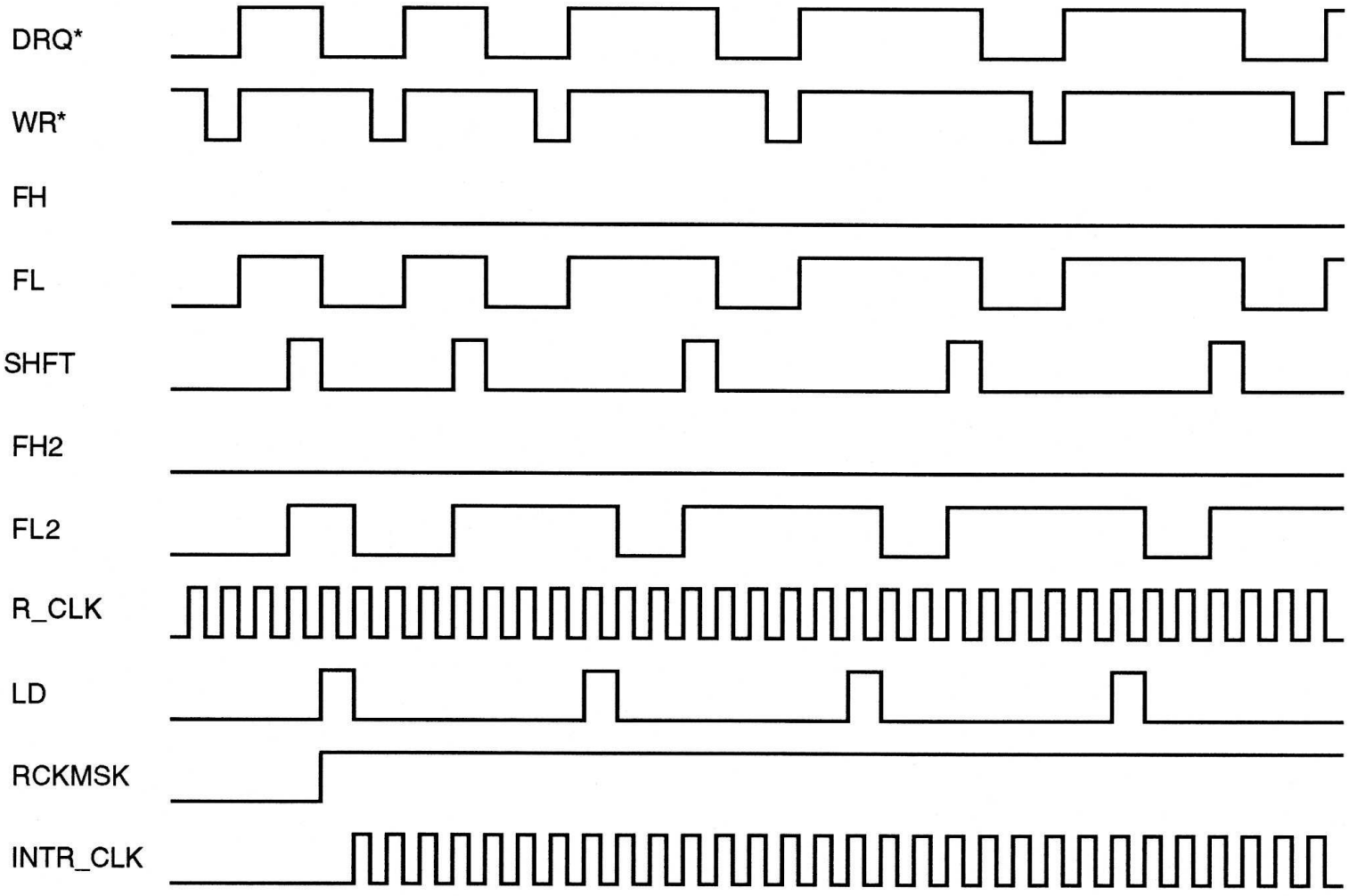

ROM A[11:0]
RD*, WR*
CS*, DACK*, BHE*
A[1:0]
L
H
L
H
L
H
L
H
SQ0
SQ0
SQ1
SQ2
SQ3
SQ0
SQ0
SQ1



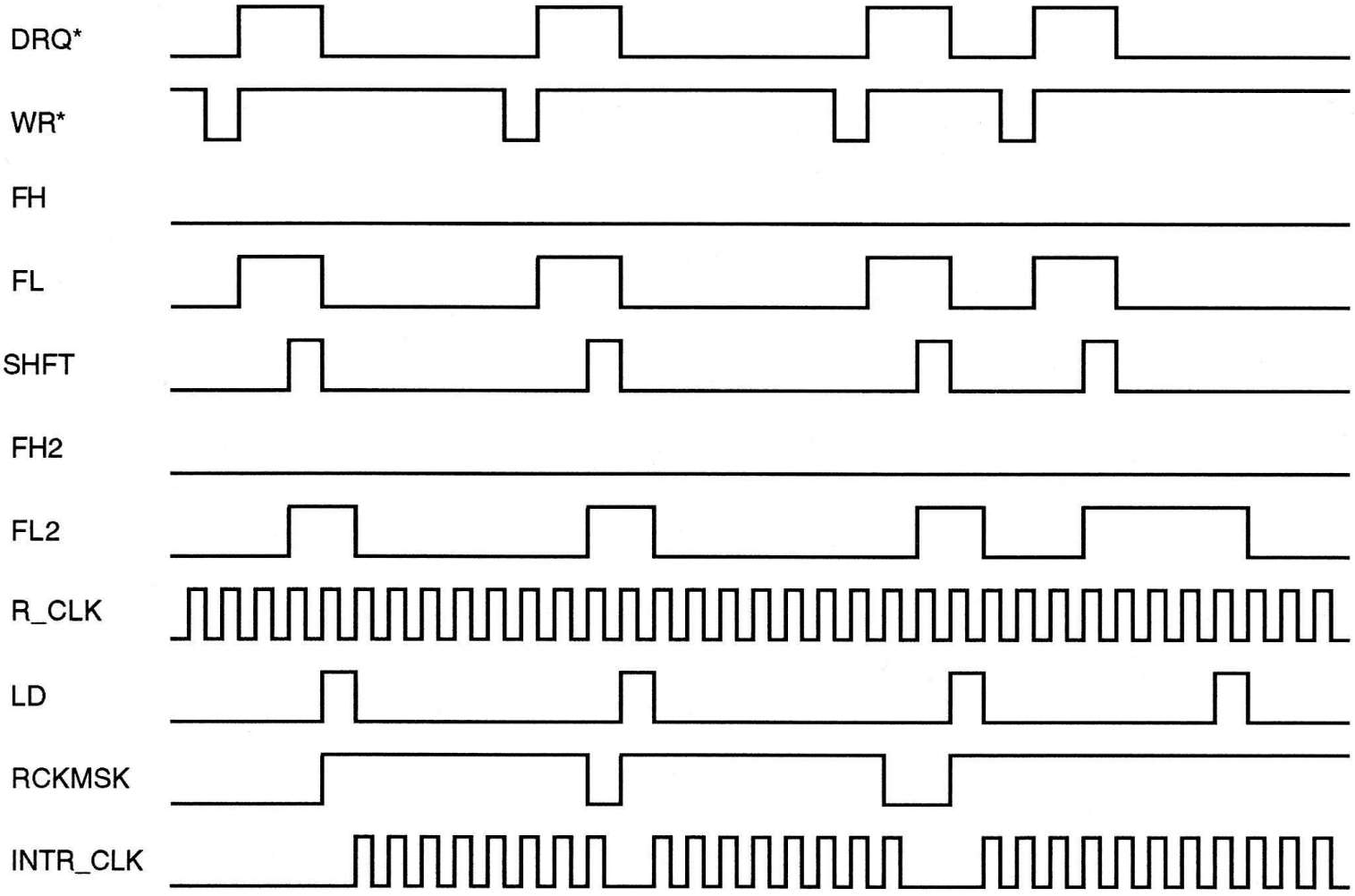
Data Register write (Word, continuous)



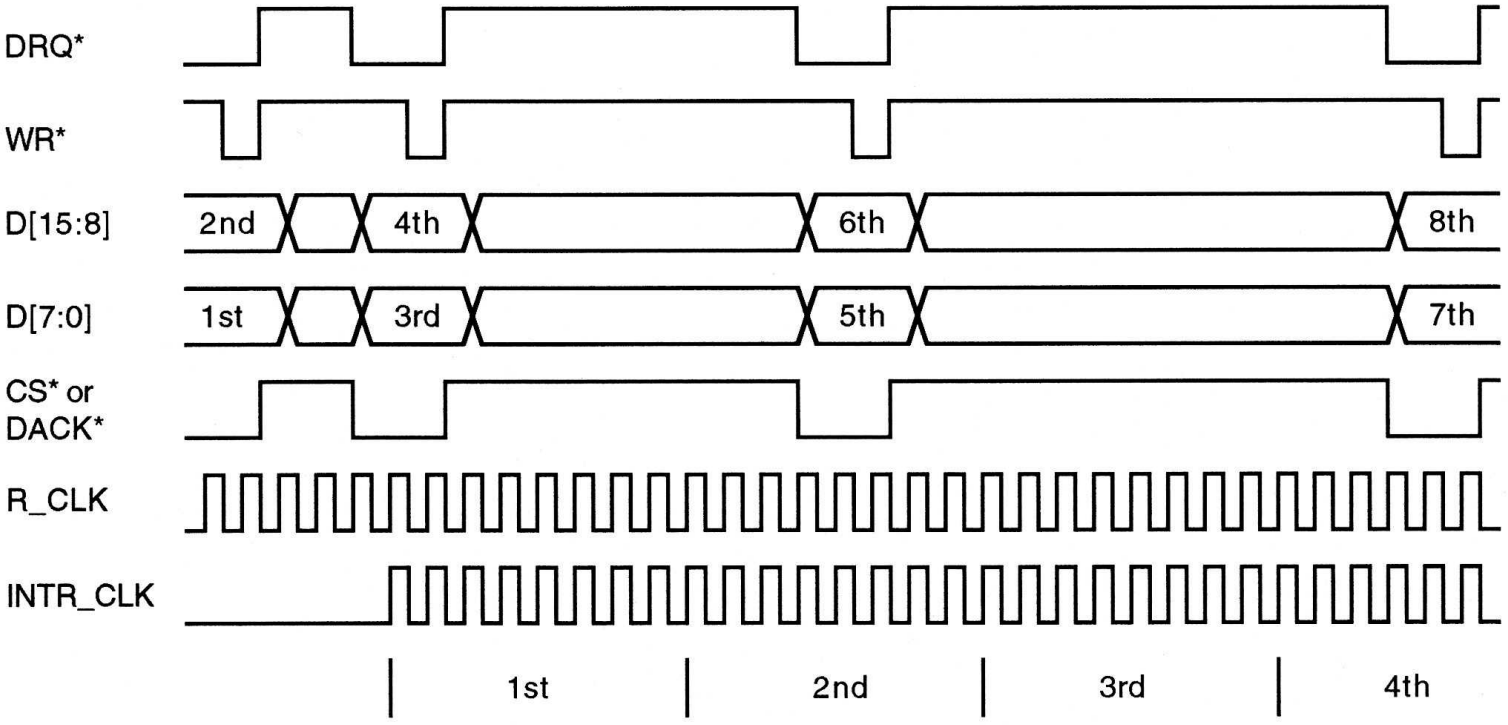
Data Register write (Word, discontinuous)



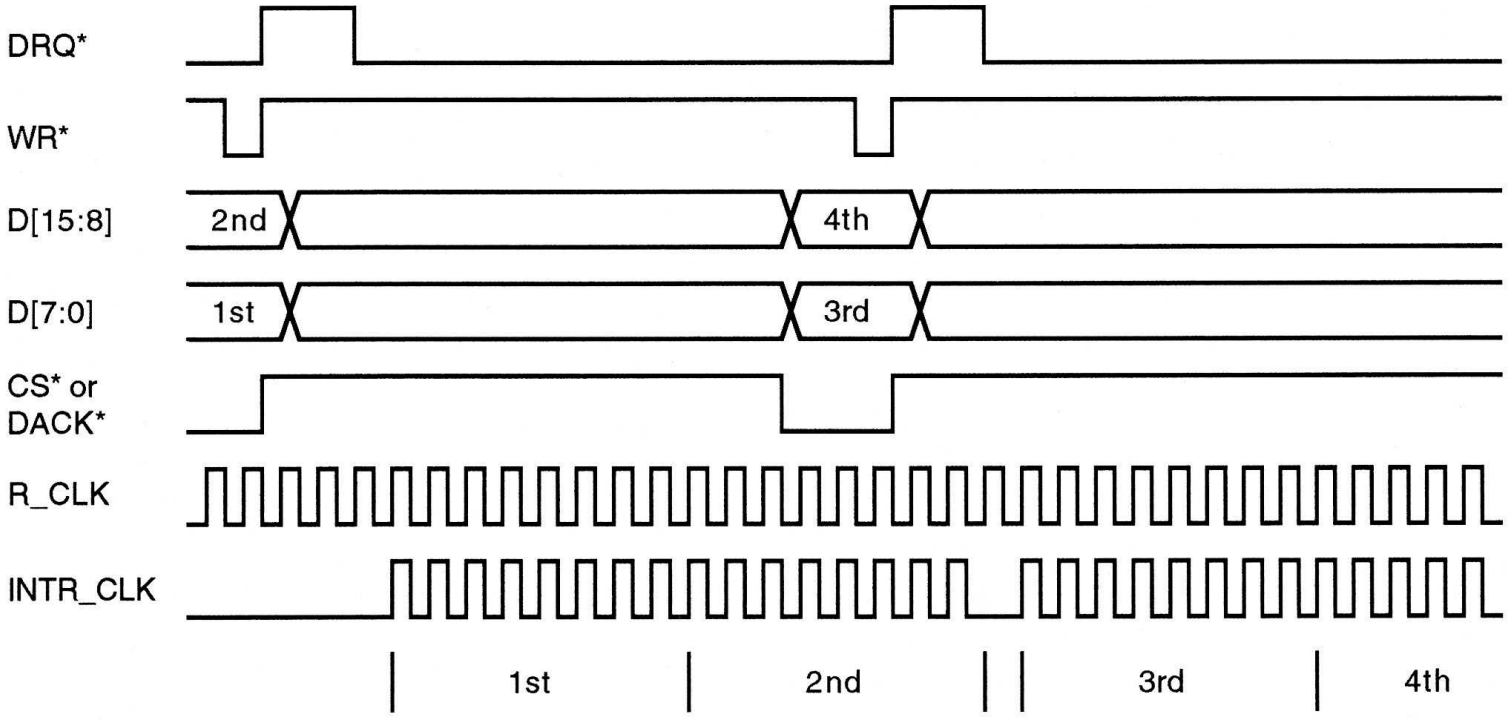
Data Register write (Low byte, continuous)



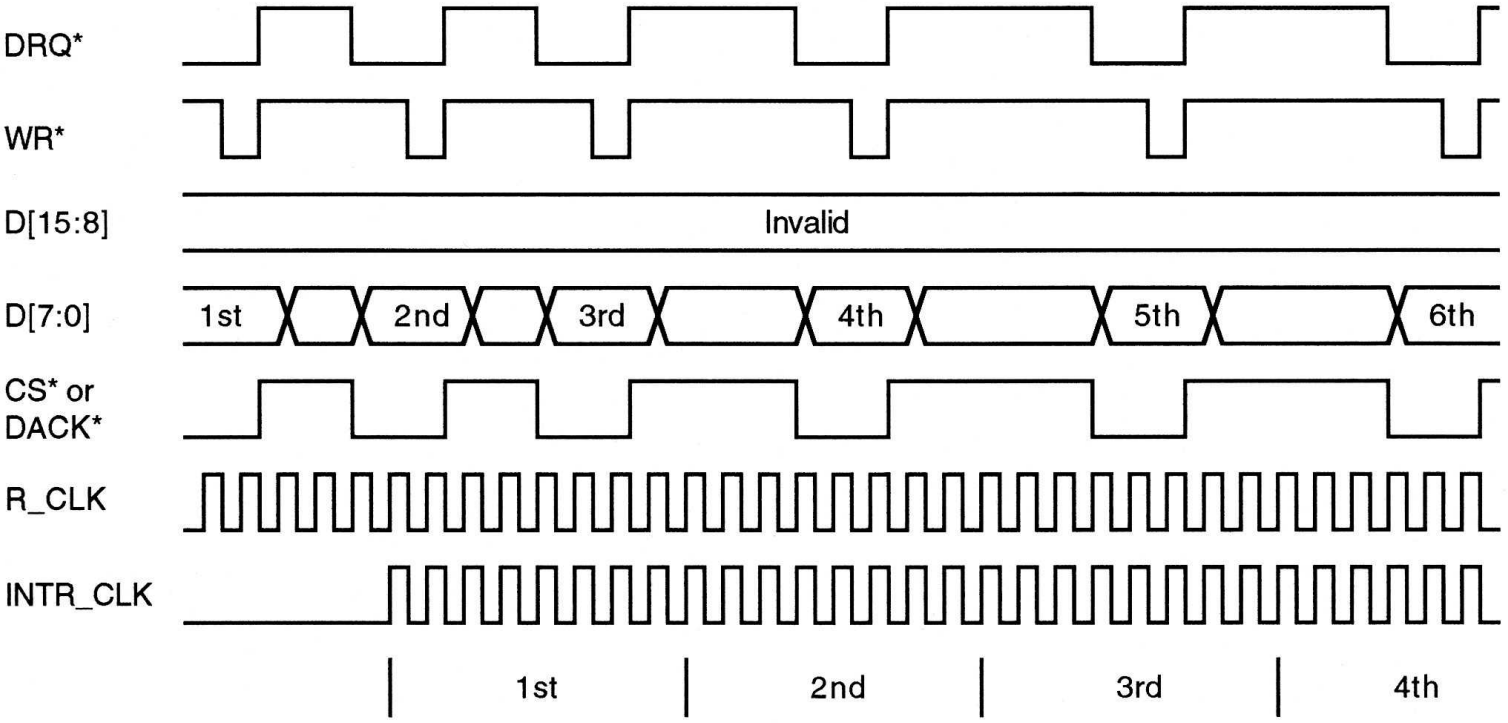
Data Register write (Low byte, discontinuous)



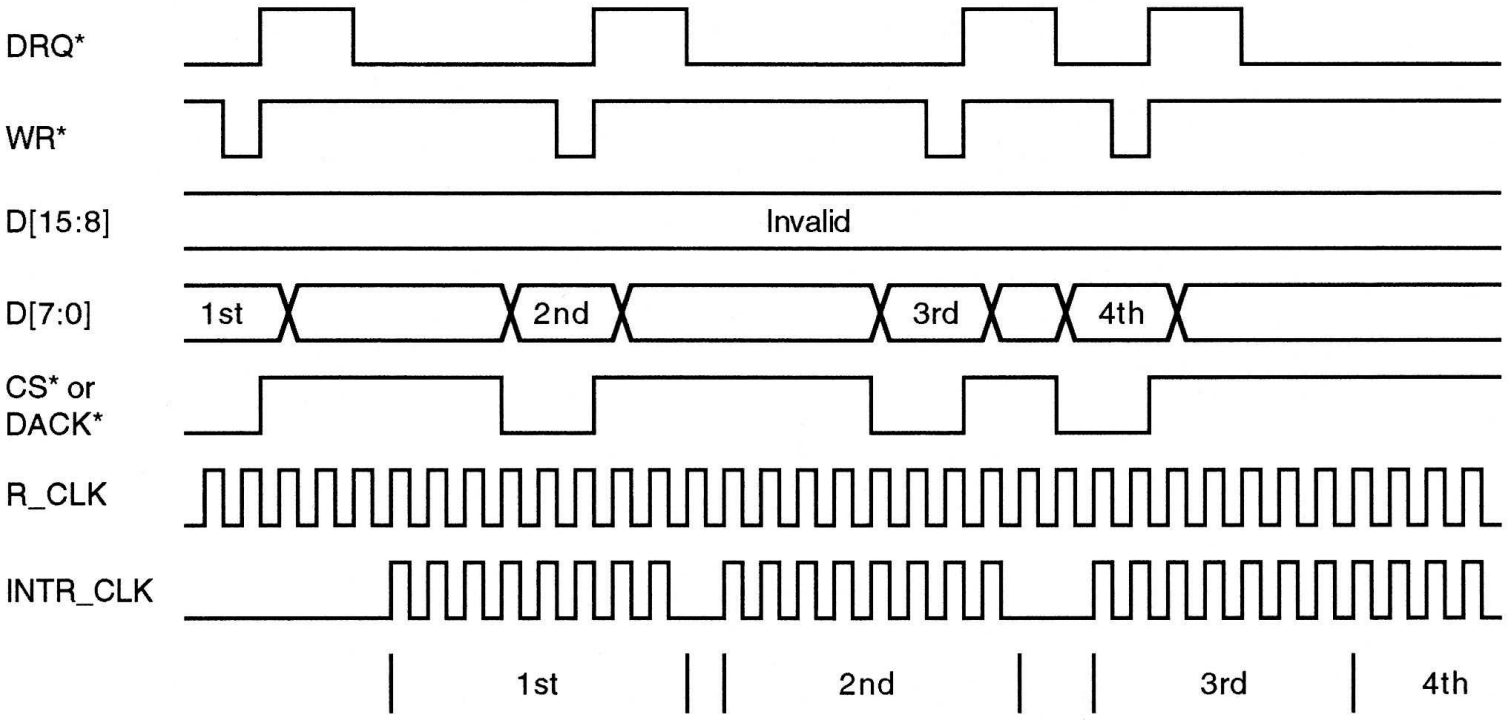
Data Register write (Word, continuous)



Data Register write (Word, discontinuous)



Data Register write (Low byte, continuous)



Data Register write (Low byte, discontinuous)