FUJITSU

SPARClite MB86860 Series User Manual

Edition 1.1 - Jul. 29, 1999

Fujitsu Ltd.

The User Manual consists of
Hardware Manual Edition 1.1 - Jul. 29, 1999 and
Programming Manual Edition 1.0 - Jun. 22, 1999

SPARClite MB86860 Series Hardware Manual

Edition 1.1 - Jul. 29, 1999

Fujitsu Ltd.

Rev.1.1 Jul.29/ 99

- **Fig 8-3 of Page 8-10 and Fig.8-4 of Page 8-11**
  DQ32, 33, 33, 34, , 62 => DQ32, 33, 34, 35, , 63
- **Page 8-12**
  Note is added.

Contents

1. Summary

1.1. Introduction

The MB8686 Series processors are high-end processors which conform to SPARC Architecture Manual Version 8. This Manual explains MB86860 Series Processor (MB86860/86861) specifications.

When developing software for use by these processors reference should also be made to the following manuals:

- SPARClite Instruction Set Manual
- MB86860 series Programming Manual

1.2. Features

1.2.1. Core Part

- Conforms to SPARC V8
- Maximum 200mhz Internal operating frequency
- Uses 2-issue superscalar architecture
- 16KB 4-way instruction cache
- 16KB 4-way data cache (write through)
- Power Down Mode
- Bi-endian support

1.2.2. Debug Support Function

- All types of break function (Instruction Address/external pin/software/single step)
- Equipped with 16-column address trace buffer
- Single step operation

1.2.3. Data Buffer Module

- 4-column instruction buffer
- 4-column x 2 read buffer
- 16-column write buffer

1.2.4. Address Conversion (TLB) Function (MB86861/MB86862)

- Number of TLB entries = 16
- Supports 4KB, 1MB, 4MB, 16MB and 64MB page sizes
- Different page sizes can be set for each entry
- When there are misses in address conversions, address from the CPU are output as they are as physical addresses without cativating traps

1.2.5. SDRAM Interface

- 64 and 32-bit data bus widths can be selected
- Maximum 100MHz operation

- Auto and self-refresh supported
- Parity functions supported

1.2.6. SPARClite Interface

- 64-bit, 32-bit-16-bit and 8-bit data buses
- Programmable Chip Select Generator Function (5-chip select)
- Wait State Control Function which generates waits for each chip select
- Burst Mode Support
- Parity Function Support
- Transparent access to SDRAM function (MB86861 only)

1.2.7. Bus-bridge DMA

- Equipped with 2 DMA channels. 1 chanel simultaneous operation.

1.2.8. Power Supply

- Internal: 2.5V
- I/O pins: 3.3V

1.3. Block Diagram

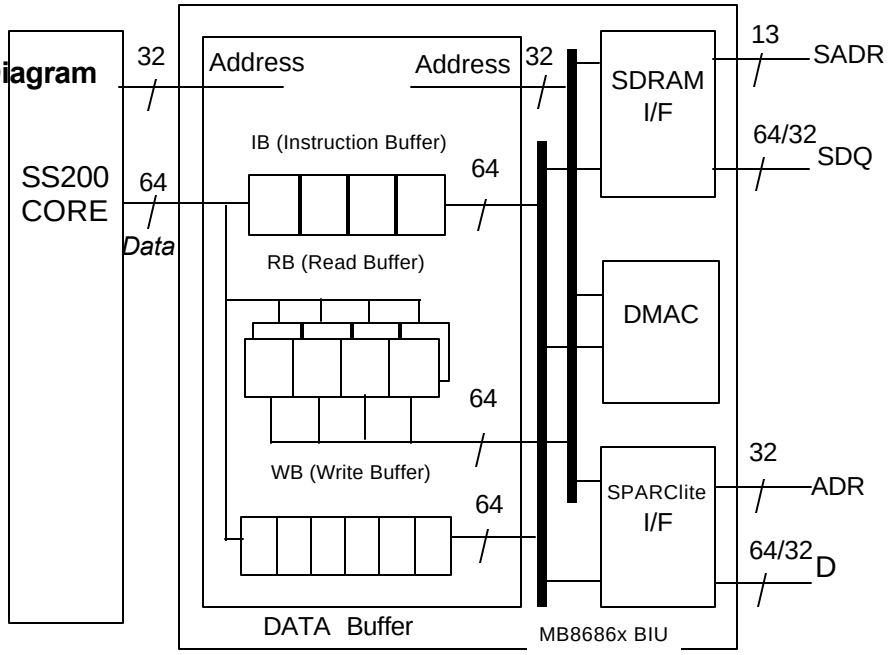
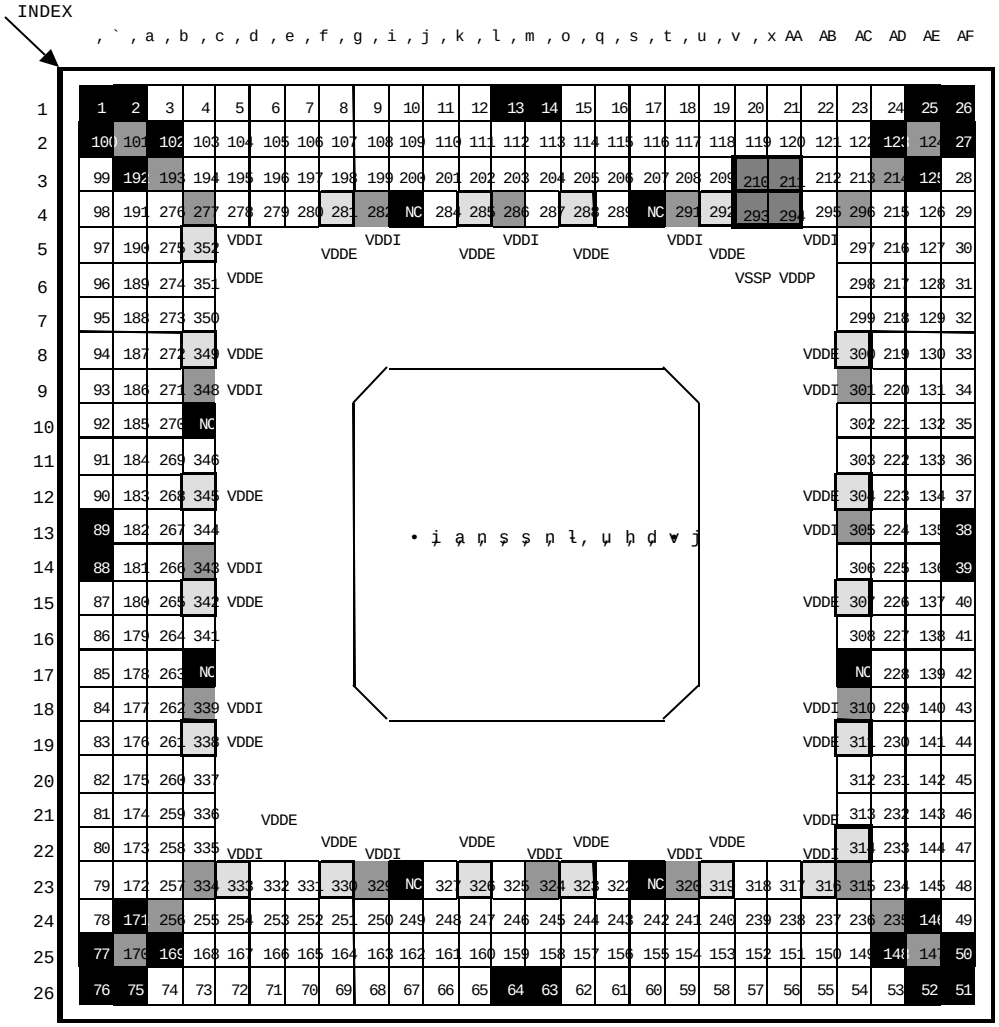


Figure 1.1 MB8686x Processor Block Diagram

2. Pins

2.1. Package Pin Assignments



: Signals (269pins)

: VDDI(24pins)	101, 124, 147, 170, 193, 214, 235, 256, 277, 282, 286, 291, 296, 301, 305, 310, 315, 320, 324, 329, 334, 339, 343, 348	
: VDDE(20pins)	281, 285, 288, 292, 300, 304, 307, 311, 319, 323, 326, 330, 338, 342, 345, 349	Additional 314,316 333,352
: VSS (28pins)	1, 2, 13, 14, 25, 26, 27, 38, 39, 50, 51, 52, 63, 64, 75, 76, 77, 88, 89, 100, 102, 123, 125, 146, 148, 169, 171, 192	NC 7pins 283,290 309,321 328,340 347

VDDP1 (for CPU Core PLL)211
VDDP2(for BIU part PLL) 294
VSSP1 (for CPU Core PLL)210
VSSP2(for BIU part PLL) 293

2.2. Pin Overview

Att.		
	I	- Input
	O	- Output
-	NC	- Non-connected
	VDD	- Internal Logic 2.5V Power Supply Pin
	VDDE	- 3.3V Power Supply Pin for I/O
	VSS	- Internal Logic and common grant pins for I/O
	VDDP1	- Power Supply Pin for PLL inside CP Core
	VSSP1	- Grant pin for PLL inside CPU Core
	VDDP2	- Power Supply Pin for BIU PLL.
	VSSP2	- Power Supply Pin for BIU PLL

This part shows pin properties in the MB86860 and MB86861 and the parts with differing functions.

2.2.1. MB86860 Pin Overview

Table 2-1 MB86860 Pin Overview

PKG Pin No.			Left side		PKG Pin No.			Bottom side		PKG Pin No.			Right side		PKG Pin No.			Top side	
(Fj)	(JEDEC)		Pin name	Att.	(Fj)	(JEDEC)		Pin name	Att.	(Fj)	(JEDEC)		Pin name	att.	(Fj)	(JEDEC)		Pin name	Att.
1	A	1	VSS	VSS	76	A	26	VSS	VSS	51	AF	26	VSS	VSS	26	AF	1	VSSP	VSS
277	D	4	VDDI	VDDI	334	D	23	VDDI	VDDI	315	AC	23	VDDI	VDDI	296	AC	4	VDDI	VDDI
276	C	4	D<23>	I/O	255	D	24	SDQ<15>	I/O	234	AD	23	SDQ<45>	I/O	213	AC	3	TRST#	I
352	D	5	VDDE	VDDE	333	E	23	VDDE	VDDE	314	AC	22	VDDE	VDDE	295	AB	4	TDI	I
99	A	3	D<24>	I/O	74	C	26	SDQ<16>	I/O	49	AF	24	SDQ<46>	I/O	24	AD	1	TMS	I
191	B	4	D<25>	I/O	168	D	25	SDQ<17>	I/O	145	AE	23	SDQ<47>	I/O	122	AC	2	TDO	O
275	C	5	D<26>	I/O	254	E	24	SDQ<18>	I/O	233	AD	22	SDQ<48>	I/O	212	AB	3	TCLK	I
351	D	6	DP4	I/O	332	F	23	SDP5	I/O	313	AC	21	SDP1	I/O	294	AA	4	VDDP	VDDP
101	B	2	VDDI	VDDI	170	B	25	VDDI	VDDI	147	AE	25	VDDI	VDDI	124	AE	2	VDDI	VDDI
98	A	4	D<27>	I/O	73	D	26	SDQ<19>	I/O	48	AF	23	SDQ<49>	I/O	23	AC	1	TEST1	I
190	B	5	D<28>	I/O	167	E	25	SDQ<20>	I/O	144	AE	22	SDQ<50>	I/O	121	AB	2	TEST2	I
274	C	6	D<29>	I/O	253	F	24	SDQ<21>	I/O	232	AD	21	SDQ<51>	I/O	211	AA	3	VDDP	VDDP
350	D	7	DP3	I/O	331	G	23	SDP4	I/O	312	AC	20	SDP0	I/O	293	Y	4	VSSP	VSSP
192	B	3	VSS	VSS	169	C	25	VSS	VSS	146	AE	24	VSS	VSS	123	AD	2	VSSP	VSS
97	A	5	D<30>	I/O	72	E	26	SDQ<22>	I/O	47	AF	22	SDQ<52>	I/O	22	AB	1	CLKSEL1	I
348	D	9	VDDI	VDDI	329	J	23	VDDI	VDDI	310	AC	18	VDDI	VDDI	291	V	4	VDDI	VDDI
189	B	6	D<31>	I/O	166	F	25	SDQ<23>	I/O	143	AE	21	SDQ<53>	I/O	120	AA	2	CLKIN	I
273	C	7	D<32>	I/O	252	G	24	SDQ<24>	I/O	231	AD	20	SDQ<54>	I/O	210	Y	3	VSSP	VSSP
349	D	8	VDDE	VDDE	330	H	23	VDDE	VDDE	311	AC	19	VDDE	VDDE	292	W	4	VDDE	VDDE
96	A	6	D<33>	I/O	71	F	26	SDQ<25>	I/O	46	AF	21	SDQ<55>	I/O	21	AA	1	CLKSEL0	I
188	B	7	D<34>	I/O	165	G	25	SDQ<26>	I/O	142	AE	20	SDQ<56>	I/O	119	Y	2	RESET#	I
100	A	2	VSS	VSS	75	B	26	VSS	VSS	50	AF	25	VSS	VSS	25	AE	1	VSSP	VSS
272	C	8	D<35>	I/O	251	H	24	SDQ<27>	I/O	230	AD	19	SDQ<57>	I/O	209	W	3	BCLK	I
95	A	7	D<36>	I/O	70	G	26	SDQ<28>	I/O	45	AF	20	SDQ<58>	I/O	20	Y	1	PLLCEN	I
187	B	8	D<37>	I/O	164	H	25	SDQ<29>	I/O	141	AE	19	SDQ<59>	I/O	118	W	2	STOP#	I
271	C	9	D<38>	I/O	250	J	24	SDQ<30>	I/O	229	AD	18	SDQ<60>	I/O	208	V	3	BEN#	I
347	D	10	NC	NC	328	K	23	NC	NC	309	AC	17	NC	NC	290	U	4	NC	NC
94	A	8	D<39>	I/O	69	H	26	SDQ<31>	I/O	44	AF	19	SDQ<61>	I/O	19	W	1	IRL<0>	I
186	B	9	D<40>	I/O	163	J	25	SDQM0	O	140	AE	18	SDQ<62>	I/O	117	V	2	IRL<1>	I
270	C	10	D<41>	I/O	249	K	24	SDQM1	O	228	AD	17	SDQ<63>	I/O	207	U	3	IRL<2>	I
93	A	9	D<42>	I/O	68	J	26	SDQM2	O	43	AF	18	ADR<2>	O	18	V	1	IRL<3>	I
346	D	11	DP2	I/O	327	L	23	FLOAT#	I	308	AC	16	ADR<3>	O	289	T	4	DBREAK#	I
185	B	10	D<43>	I/O	162	K	25	SDQM3	O	139	AE	17	ADR<4>	O	116	U	2	BRKEN#	I
269	C	11	D<44>	I/O	248	L	24	SDQM4	O	227	AD	16	ADR<5>	O	206	T	3	BRKGO	O
92	A	10	D<45>	I/O	67	K	26	SDQM5	O	42	AF	17	ADR<6>	O	17	U	1	BREQ#	I
343	D	14	VDDI	VDDI	324	P	23	VDDI	VDDI	305	AC	13	VDDI	VDDI	286	N	4	VDDI	VDDI
184	B	11	D<46>	I/O	161	L	25	DQM6	O	138	AE	16	ADR<7>	O	115	T	2	BGRNT#	O
345	D	12	VDDE	VDDE	326	M	23	VDDE	VDDE	307	AC	15	VDDE	VDDE	288	R	4	VDDE	VDDE
91	A	11	D<47>	I/O	66	L	26	DQM7	O	41	AF	16	ADR<8>	O	16	T	1	PBREQ#	O
268	C	12	D<48>	I/O	247	M	24	SADR<0>	O	226	AD	15	ADR<9>	O	205	R	3	BE0#	O
183	B	12	D<49>	I/O	160	M	25	SADR<1>	O	137	AE	15	ADR<10>	O	114	R	2	BE1#	O
89	A	13	VSS	VSS	64	N	26	VSS	VSS	39	AF	14	VSS	VSS	14	P	1	VSS	VSS
90	A	12	D<50>	I/O	65	M	26	SADR<2>	O	40	AF	15	ADR<11>	O	15	R	1	BE2#	O
344	D	13	DP1	I/O	325	N	23	SADR<3>	O	306	AC	14	ADR<12>	O	287	P	4	BE3#	O
267	C	13	D<51>	I/O	246	N	24	SADR<4>	O	225	AD	14	ADR<13>	O	204	P	3	BE4#	O

(Table 2-2 MB86860 Pins continued)

PKG Pin No.			Left side		PKG Pin No.			Bottom side		PKG Pin No.			Right side		PKG Pin No.			Top side	
(Fj)	(JEDEC)		Pin name	Att.	(Fj)	(JEDEC)		Pin name	Att.	(Fj)	(JEDEC)		Pin name	att.	(Fj)	(JEDEC)		Pin name	Att.
182	B	13	D<52>	I/O	159	N	25	SADR<5>	O	136	AE	14	ADR<14>	O	113	P	2	BE5#	O
181	B	14	D<53>	I/O	158	P	25	SADR<6>	O	135	AE	13	ADR<15>	O	112	N	2	BE6#	O
266	C	14	D<54>	I/O	245	P	24	SADR<7>	O	224	AD	13	ADR<16>	O	203	N	3	BE7#	O
87	A	15	D<55>	I/O	62	R	26	SADR<8>	O	37	AF	12	ADR<17>	O	12	M	1	WKUP#	I
88	A	14	VSS	VSS	63	P	26	VSS	VSS	38	AF	13	VSS	VSS	13	N	1	VSS	VSS
180	B	15	D<56>	I/O	157	R	25	SADR<9>	O	134	AE	12	ADR<18>	O	111	M	2	PDWN#	O
265	C	15	D<57>	I/O	244	R	24	SADR<10>	O	223	AD	12	ADR<19>	O	202	M	3	AS#	O
86	A	16	D<58>	I/O	61	T	26	SADR<11>	O	36	AF	11	ADR<20>	O	11	L	1	RD#	O
342	D	15	VDDE	VDDE	323	R	23	VDDE	VDDE	304	AC	12	VDDE	VDDE	285	M	4	VDDE	VDDE
179	B	16	D<59>	I/O	156	T	25	SADR<12>	O	133	AE	11	ADR<21>	O	110	L	2	RDWR#	O
85	A	17	D<60>	I/O	60	U	26	SRAS#	O	35	AF	10	ADR<22>	O	10	K	1	LOCK#	O
264	C	16	D<61>	I/O	243	T	24	SCKE#	O	222	AD	11	ADR<23>	O	201	L	3	D<0>	I/O
178	B	17	D<62>	I/O	155	U	25	SCAS#	O	132	AE	10	ADR<24>	O	109	K	2	D<1>	I/O
341	D	16	DP0	I/O	322	T	23	SCLK	O	303	AC	11	ADR<25>	O	284	L	4	D<2>	I/O
84	A	18	D<63>	I/O	59	V	26	SWE#	O	34	AF	9	ADR<26>	O	9	J	1	D<3>	I/O
263	C	17	MEXC#	I	242	U	24	SCS0#	O	221	AD	10	ADR<27>	O	200	K	3	D<4>	I/O
339	D	18	VDDI	VDDI	320	V	23	VDDI	VDDI	301	AC	9	VDDI	VDDI	282	J	4	VDDI	VDDI
177	B	18	READY#	I	154	V	25	SCS1#	O	131	AE	9	ADR<28>	O	108	J	2	D<5>	I/O
83	A	19	RDYOUT#	O	58	W	26	SCS2#	O	33	AF	8	ADR<29>	O	8	H	1	D<6>	I/O
340	D	17	NC	NC	321	U	23	NC	NC	302	AC	10	NC	NC	283	K	4	NC	NC
262	C	18	BMODE32#	I	241	V	24	SCS3#	O	220	AD	9	ADR<30>	O	199	J	3	D<7>	I/O
176	B	19	BMODE16#	I	153	W	25	SBA<0>	O	130	AE	8	ADR<31>	O	107	H	2	D<8>	I/O
82	A	20	SDQ<0>	I/O	57	Y	26	SBA<1>	O	32	AF	7	DTYP<1>	O	7	G	1	D<9>	I/O
261	C	19	SDQ<1>	I/O	240	W	24	SDQ<32>	I/O	219	AD	8	DTYP<0>	O	198	H	3	D<10>	I/O
171	B	24	VSS	VSS	148	AD	25	VSS	VSS	27	AF	2	VSS	VSS	2	B	1	VSS	VSS
175	B	20	SDQ<2>	I/O	152	Y	25	SDQ<33>	I/O	129	AE	7	BMACK#	I	106	G	2	D<11>	I/O
81	A	21	SDQ<3>	I/O	56	AA	26	SDQ<34>	I/O	31	AF	6	BMREQ#	O	6	F	1	D<12>	I/O
338	D	19	VDDE	VDDE	319	W	23	VDDE	VDDE	300	AC	8	VDDE	VDDE	281	H	4	VDDE	VDDE
260	C	20	SDQ<4>	I/O	239	Y	24	SDQ<35>	I/O	218	AD	7	ERROR#	O	197	G	3	D<13>	I/O
174	B	21	SDQ<5>	I/O	151	AA	25	SDQ<36>	I/O	128	AE	6	EOP#	O	105	F	2	D<14>	I/O
80	A	22	SDQ<6>	I/O	55	AB	26	SDQ<37>	I/O	30	AF	5	CS0#	O	5	E	1	D<15>	I/O
337	D	20	SDQ<7>	I/O	318	Y	23	SDP3	I/O	299	AC	7	CS1#	O	280	G	4	DP7	I/O
259	C	21	SDQ<8>	I/O	238	AA	24	SDQ<38>	I/O	217	AD	6	CS2#	O	196	F	3	D<16>	I/O
173	B	22	SDQ<9>	I/O	150	AB	25	SDQ<39>	I/O	127	AE	5	CS3#	O	104	E	2	D<17>	I/O
79	A	23	SDQ<10>	I/O	54	AC	26	SDQ<40>	I/O	29	AF	4	CS4#	O	4	D	1	D<18>	I/O
256	C	24	VDDI	VDDI	235	AD	24	VDDI	VDDI	214	AD	3	VDDI	VDDI	193	C	3	VDDI	VDDI
336	D	21	SDP7	I/O	317	AA	23	SDP2	I/O	298	AC	6	CS5#	O	279	F	4	DP6	I/O
77	A	25	VSS	VSS	52	AE	26	VSS	VSS	125	AE	3	VSS	VSS	102	C	2	VSS	VSS
258	C	22	SDQ<11>	I/O	237	AB	24	SDQ<41>	I/O	216	AD	5	ASI<0>	O	195	E	3	D<19>	I/O
172	B	23	SDQ<12>	I/O	149	AC	25	SDQ<42>	I/O	126	AE	4	ASI<1>	O	103	D	2	D<20>	I/O
78	A	24	SDQ<13>	I/O	53	AD	26	SDQ<43>	I/O	28	AF	3	ASI<2>	O	3	C	1	D<21>	I/O
335	D	22	SDP6	I/O	316	AB	23	VDDE	VDDE	297	AC	5	ASI<3>	O	278	E	4	DP5	I/O
257	C	23	SDQ<14>	I/O	236	AC	24	SDQ<44>	I/O	215	AD	4	NCTEST#	I	194	D	3	D<22>	I/O

2.2.2. MB86861 Pin Overview

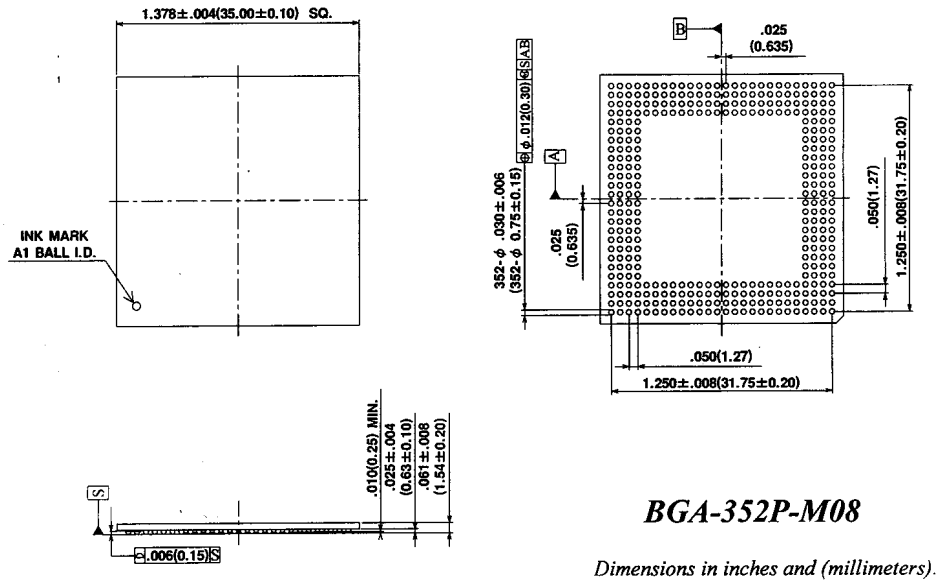
Table 2-3 Pin Overview (MB86861)

PKG Pin No.			Left side		PKG Pin No.			Bottom side		PKG Pin No.			Right side		PKG Pin No.			Top side	
(Fj)	(JEDEC)		Pin name	Att.	(Fj)	(JEDEC)		Pin name	Att.	(Fj)	(JEDEC)		Pin name	att.	(Fj)	(JEDEC)		Pin name	Att.
1	A	1	VSS	VSS	76	A	26	VSS	VSS	51	AF	26	VSS	VSS	26	AF	1	VSSP	VSS
277	D	4	VDDI	VDDI	334	D	23	VDDI	VDDI	315	AC	23	VDDI	VDDI	296	AC	4	VDDI	VDDI
276	C	4	D<23>	I/O	255	D	24	SDQ<15>	I/O	234	AD	23	SDQ<45>	I/O	213	AC	3	TRST#	I
352	D	5	VDDE	VDDE	333	E	23	VDDE	VDDE	314	AC	22	VDDE	VDDE	295	AB	4	TDI	I
99	A	3	D<24>	I/O	74	C	26	SDQ<16>	I/O	49	AF	24	SDQ<46>	I/O	24	AD	1	TMS	I
191	B	4	D<25>	I/O	168	D	25	SDQ<17>	I/O	145	AE	23	SDQ<47>	I/O	122	AC	2	TDO	O
275	C	5	D<26>	I/O	254	E	24	SDQ<18>	I/O	233	AD	22	SDQ<48>	I/O	212	AB	3	TCLK	I
351	D	6	DP4	I/O	332	F	23	SDP5	I/O	313	AC	21	SDP1	I/O	294	AA	4	VDDP	VDDP
101	B	2	VDDI	VDDI	170	B	25	VDDI	VDDI	147	AE	25	VDDI	VDDI	124	AE	2	VDDI	VDDI
98	A	4	D<27>	I/O	73	D	26	SDQ<19>	I/O	48	AF	23	SDQ<49>	I/O	23	AC	1	TEST1	I
190	B	5	D<28>	I/O	167	E	25	SDQ<20>	I/O	144	AE	22	SDQ<50>	I/O	121	AB	2	TEST2	I
274	C	6	D<29>	I/O	253	F	24	SDQ<21>	I/O	232	AD	21	SDQ<51>	I/O	211	AA	3	VDDP	VDDP
350	D	7	DP3	I/O	331	G	23	SDP4	I/O	312	AC	20	SDP0	I/O	293	Y	4	VSSP	VSSP
192	B	3	VSS	VSS	169	C	25	VSS	VSS	146	AE	24	VSS	VSS	123	AD	2	VSSP	VSS
97	A	5	D<30>	I/O	72	E	26	SDQ<22>	I/O	47	AF	22	SDQ<52>	I/O	22	AB	1	CLKSEL1	I
348	D	9	VDDI	VDDI	329	J	23	VDDI	VDDI	310	AC	18	VDDI	VDDI	291	V	4	VDDI	VDDI
189	B	6	D<31>	I/O	166	F	25	SDQ<23>	I/O	143	AE	21	SDQ<53>	I/O	120	AA	2	CLKIN	I
273	C	7	D<32>	I/O	252	G	24	SDQ<24>	I/O	231	AD	20	SDQ<54>	I/O	210	Y	3	VSSP	VSSP
349	D	8	VDDE	VDDE	330	H	23	VDDE	VDDE	311	AC	19	VDDE	VDDE	292	W	4	VDDE	VDDE
96	A	6	D<33>	I/O	71	F	26	SDQ<25>	I/O	46	AF	21	SDQ<55>	I/O	21	AA	1	CLKSEL0	I
188	B	7	D<34>	I/O	165	G	25	SDQ<26>	I/O	142	AE	20	SDQ<56>	I/O	119	Y	2	RESET#	I
100	A	2	VSS	VSS	75	B	26	VSS	VSS	50	AF	25	VSS	VSS	25	AE	1	VSSP	VSS
272	C	8	D<35>	I/O	251	H	24	SDQ<27>	I/O	230	AD	19	SDQ<57>	I/O	209	W	3	BCLK	I
95	A	7	D<36>	I/O	70	G	26	SDQ<28>	I/O	45	AF	20	SDQ<58>	I/O	20	Y	1	PLLCEN	I
187	B	8	D<37>	I/O	164	H	25	SDQ<29>	I/O	141	AE	19	SDQ<59>	I/O	118	W	2	STOP#	I
271	C	9	D<38>	I/O	250	J	24	SDQ<30>	I/O	229	AD	18	SDQ<60>	I/O	208	V	3	BEN#	I
347	D	10	NC	NC	328	K	23	NC	NC	309	AC	17	NC	NC	290	U	4	NC	NC
94	A	8	D<39>	I/O	69	H	26	SDQ<31>	I/O	44	AF	19	SDQ<61>	I/O	19	W	1	IRL<0>	I
186	B	9	D<40>	I/O	163	J	25	SDQM0	O	140	AE	18	SDQ<62>	I/O	117	V	2	IRL<1>	I
270	C	10	D<41>	I/O	249	K	24	SDQM1	O	228	AD	17	SDQ<63>	I/O	207	U	3	IRL<2>	I
93	A	9	D<42>	I/O	68	J	26	SDQM2	O	43	AF	18	ADR<2>	I/O	18	V	1	IRL<3>	I
346	D	11	DP2	I/O	327	L	23	FLOAT#	I	308	AC	16	ADR<3>	I/O	289	T	4	DBREAK#	I
185	B	10	D<43>	I/O	162	K	25	SDQM3	O	139	AE	17	ADR<4>	I/O	116	U	2	BRKEN#	I
269	C	11	D<44>	I/O	248	L	24	SDQM4	O	227	AD	16	ADR<5>	I/O	206	T	3	BRKGO	O
92	A	10	D<45>	I/O	67	K	26	SDQM5	O	42	AF	17	ADR<6>	I/O	17	U	1	BREQ#	I
343	D	14	VDDI	VDDI	324	P	23	VDDI	VDDI	305	AC	13	VDDI	VDDI	286	N	4	VDDI	VDDI
184	B	11	D<46>	I/O	161	L	25	DQM6	O	138	AE	16	ADR<7>	I/O	115	T	2	BGRNT#	O
345	D	12	VDDE	VDDE	326	M	23	VDDE	VDDE	307	AC	15	VDDE	VDDE	288	R	4	VDDE	VDDE
91	A	11	D<47>	I/O	66	L	26	DQM7	O	41	AF	16	ADR<8>	I/O	16	T	1	PBREQ#	O
268	C	12	D<48>	I/O	247	M	24	SADR<0>	O	226	AD	15	ADR<9>	I/O	205	R	3	BE0#	I/O
183	B	12	D<49>	I/O	160	M	25	SADR<1>	O	137	AE	15	ADR<10>	I/O	114	R	2	BE1#	I/O
89	A	13	VSS	VSS	64	N	26	VSS	VSS	39	AF	14	VSS	VSS	14	P	1	VSS	VSS
90	A	12	D<50>	I/O	65	M	26	SADR<2>	O	40	AF	15	ADR<11>	I/O	15	R	1	BE2#	I/O
344	D	13	DP1	I/O	325	N	23	SADR<3>	O	306	AC	14	ADR<12>	I/O	287	P	4	BE3#	I/O
267	C	13	D<51>	I/O	246	N	24	SADR<4>	O	225	AD	14	ADR<13>	I/O	204	P	3	BE4#	I/O

Table 2-4 Pin Overview (MB86861 continued)

PKG Pin No.			Left side		PKG Pin No.			Bottom side		PKG Pin No.			Right side		PKG Pin No.			Top side	
(Fj)	(JEDEC)		Pin name	Att.	(Fj)	(JEDEC)		Pin name	Att.	(Fj)	(JEDEC)		Pin name	att.	(Fj)	(JEDEC)		Pin name	Att.
182	B	13	D<52>	I/O	159	N	25	SADR<5>	O	136	AE	14	ADR<14>	I/O	113	P	2	BE5#	I/O
181	B	14	D<53>	I/O	158	P	25	SADR<6>	O	135	AE	13	ADR<15>	I/O	112	N	2	BE6#	I/O
266	C	14	D<54>	I/O	245	P	24	SADR<7>	O	224	AD	13	ADR<16>	I/O	203	N	3	BE7#	I/O
87	A	15	D<55>	I/O	62	R	26	SADR<8>	O	37	AF	12	ADR<17>	I/O	12	M	1	WKUP#	I
88	A	14	VSS	VSS	63	P	26	VSS	VSS	38	AF	13	VSS	VSS	13	N	1	VSS	VSS
180	B	15	D<56>	I/O	157	R	25	SADR<9>	O	134	AE	12	ADR<18>	I/O	111	M	2	PDWN#	O
265	C	15	D<57>	I/O	244	R	24	SADR<10>	O	223	AD	12	ADR<19>	I/O	202	M	3	AS#	I/O
86	A	16	D<58>	I/O	61	T	26	SADR<11>	O	36	AF	11	ADR<20>	I/O	11	L	1	RD#	O
342	D	15	VDDE	VDDE	323	R	23	VDDE	VDDE	304	AC	12	VDDE	VDDE	285	M	4	VDDE	VDDE
179	B	16	D<59>	I/O	156	T	25	SADR<12>	O	133	AE	11	ADR<21>	I/O	110	L	2	RDWR#	I/O
85	A	17	D<60>	I/O	60	U	26	SRAS#	O	35	AF	10	ADR<22>	I/O	10	K	1	LOCK#	O
264	C	16	D<61>	I/O	243	T	24	SCKE#	O	222	AD	11	ADR<23>	I/O	201	L	3	D<0>	I/O
178	B	17	D<62>	I/O	155	U	25	SCAS#	O	132	AE	10	ADR<24>	I/O	109	K	2	D<1>	I/O
341	D	16	DP0	I/O	322	T	23	SCLK	O	303	AC	11	ADR<25>	I/O	284	L	4	D<2>	I/O
84	A	18	D<63>	I/O	59	V	26	SWE#	O	34	AF	9	ADR<26>	I/O	9	J	1	D<3>	I/O
263	C	17	MEXC#	I	242	U	24	SCS0#	O	221	AD	10	ADR<27>	I/O	200	K	3	D<4>	I/O
339	D	18	VDDI	VDDI	320	V	23	VDDI	VDDI	301	AC	9	VDDI	VDDI	282	J	4	VDDI	VDDI
177	B	18	READY#	I	154	V	25	SCS1#	O	131	AE	9	ADR<28>	I/O	108	J	2	D<5>	I/O
83	A	19	RDYOUT#	O	58	W	26	SCS2#	O	33	AF	8	ADR<29>	I/O	8	H	1	D<6>	I/O
340	D	17	NC	NC	321	U	23	NC	NC	302	AC	10	BLEN8#	I	283	K	4	NC	NC
262	C	18	BMODE32#	I	241	V	24	SCS3#	O	220	AD	9	ADR<30>	I/O	199	J	3	D<7>	I/O
176	B	19	BMODE16#	I	153	W	25	SBA<0>	O	130	AE	8	ADR<31>	I/O	107	H	2	D<8>	I/O
82	A	20	SDQ<0>	I/O	57	Y	26	SBA<1>	O	32	AF	7	DTYP<1>	O	7	G	1	D<9>	I/O
261	C	19	SDQ<1>	I/O	240	W	24	SDQ<32>	I/O	219	AD	8	DTYP<0>	O	198	H	3	D<10>	I/O
171	B	24	VSS	VSS	148	AD	25	VSS	VSS	27	AF	2	VSS	VSS	2	B	1	VSS	VSS
175	B	20	SDQ<2>	I/O	152	Y	25	SDQ<33>	I/O	129	AE	7	BMACK#	I/O	106	G	2	D<11>	I/O
81	A	21	SDQ<3>	I/O	56	AA	26	SDQ<34>	I/O	31	AF	6	BMREQ#	I/O	6	F	1	D<12>	I/O
338	D	19	VDDE	VDDE	319	W	23	VDDE	VDDE	300	AC	8	VDDE	VDDE	281	H	4	VDDE	VDDE
260	C	20	SDQ<4>	I/O	239	Y	24	SDQ<35>	I/O	218	AD	7	ERROR#	O	197	G	3	D<13>	I/O
174	B	21	SDQ<5>	I/O	151	AA	25	SDQ<36>	I/O	128	AE	6	EOP#	O	105	F	2	D<14>	I/O
80	A	22	SDQ<6>	I/O	55	AB	26	SDQ<37>	I/O	30	AF	5	CS0#	O	5	E	1	D<15>	I/O
337	D	20	SDQ<7>	I/O	318	Y	23	SDP3	I/O	299	AC	7	CS1#	O	280	G	4	DP7	I/O
259	C	21	SDQ<8>	I/O	238	AA	24	SDQ<38>	I/O	217	AD	6	CS2#	O	196	F	3	D<16>	I/O
173	B	22	SDQ<9>	I/O	150	AB	25	SDQ<39>	I/O	127	AE	5	CS3#	O	104	E	2	D<17>	I/O
79	A	23	SDQ<10>	I/O	54	AC	26	SDQ<40>	I/O	29	AF	4	CS4#	O	4	D	1	D<18>	I/O
256	C	24	VDDI	VDDI	235	AD	24	VDDI	VDDI	214	AD	3	VDDI	VDDI	193	C	3	VDDI	VDDI
336	D	21	SDP7	I/O	317	AA	23	SDP2	I/O	298	AC	6	CS5#	O	279	F	4	DP6	I/O
77	A	25	VSS	VSS	52	AE	26	VSS	VSS	125	AE	3	VSS	VSS	102	C	2	VSS	VSS
258	C	22	SDQ<11>	I/O	237	AB	24	SDQ<41>	I/O	216	AD	5	ASI<0>	O	195	E	3	D<19>	I/O
172	B	23	SDQ<12>	I/O	149	AC	25	SDQ<42>	I/O	126	AE	4	ASI<1>	O	103	D	2	D<20>	I/O
78	A	24	SDQ<13>	I/O	53	AD	26	SDQ<43>	I/O	28	AF	3	ASI<2>	O	3	C	1	D<21>	I/O
335	D	22	SDP6	I/O	316	AB	23	VDDE	VDDE	297	AC	5	ASI<3>	O	278	E	4	DP5	I/O
257	C	23	SDQ<14>	I/O	236	AC	24	SDQ<44>	I/O	215	AD	4	SDSEL#	I	194	D	3	D<22>	I/O

2.3. Outer Package Dimension Drawing



BGA-352P-M08

Dimensions in inches and (millimeters).

BGA352008Sc-3

2.4. Explanation of Pin Functions

2.4.1. SPARClite Bus Signals

The notations about bus grant in the explanation column in Table 2-5 all assume that the transparent access mode to SDRAM is enabled.

Table 2-5 SPARClite BUS Signals

Pin Name	Normal / In Bus Grant / In Sleep Mode	Explanation
CLKIN	I	Clock Input Pin. This is the clock which regulates SPARClite bus operations, and SPARClite bus operations and AC characteristics are regulated with this clock as the standard
AS#	O	**When the CPU has the Bus Right** It ouputs an address strobe. "L" is output for 1 clock cycle period in the first bus cycle. Basically, the bus cycle starts with assertion of AS# and ends with assertion of READY# or RDYOUT# for a specified number of times. In 64-bit bus width, if for 1 assertion of AS# READY is input once in single transfers and 4 times in burst transfers, the bus cycle is completed.
	I	**In Bus Grant** This is address strobe input. If this signal is asserted together with SDSEL#, transparent access to SDRAM is deemed to have been requested, and a transparent access operation begins. A decision is made whether at the RDWR# level of the cycle in which this signal was asserted there was a read from SDRAM or a write, and accessed addresses are fetched externally.
	O (Z) (860)	**In Sleep Mode** High-Z Status is effective.
	O (H) (861)	**In Sleep Mode** Inactive Level "H" is output.
ADR<31:2>	O	**When the CPU has the Bus Right** It outputs address signals. This signal is for identifying instruction addresses and data addresses. This pin is valid during bus cycles, and output values during idle cycles are not guaranteed. During burst transfers values change sequentially (wrap-around system). In 64-bit bus width burst transfers, ADR<4:3> changes in the following 4 patterns: (1) 00→01→10→11 (2) 01→10→11→00 (3) 10→11→00→01 (4) 11→00→01→10 In 16-bit and 8-bit bus width ADR<1:0> information is output to BE4# and BE5#
	I	**In Bus Grant** This is the address input pin for requesting transparent access to SDRAM. It is sampled in AS# input cycles and is used as an address for accessing SDRAM. In single transfers ADR<31:3> is deemed the valid address. In burst transfers the CPU internally updates and uses addresses after fetching the first address. When there are 4 bursts, the address lower 2 bits ADR<4:3> must be 00, and for 8 bursts the lower 3 bits (ADR<5:3>) must be 000. 4 Burst ADR<4:3>: 00→01→10→11 8 Burst ADR<5:3>: 000001→010→011→100→101→110→111
	O (Z) (860)	**In Sleep Mode** High-Z Status is effective.

	O (V) (861)	**In Sleep Mode** “H” or “L” level is output.

Pin Name	Normal / In Bus Grant / In Sleep Mode	Explanation
ASI<3:0>	O	**When the CPU has the Bus Right** ASI output pin. Output is valid during bus cycle periods, as are addresses.
	O(Z)	**In Bus Grant** High-Z Status. Not used in access transparent function.
	O(Z)	**In Sleep Mode** High-Z Status.
D<63:0>	I/O	**When the CPU has the Bus Right** Data bus signal. This is a bi-directional data bus used for instruction fetches, data loads and data stores. It must be aligned respectively to addresses which are multiples of 8 for double word data type, multiples of 4 for word data type and multiples of 2 for half word data type. When the RDWR# signal is "L" a write cycle is in effect, and this data bus is in output status. D<31:0> is used in 32-bit bus mode, D<15:0> in 16-bit bus mode and D<7:0> in 8-bit bus mode. Since unused data buses are not driven when using in 32, 16 and 8-bit bus modes, pull up resistance must be applied externally.
	I/O	**In Bus Grant** Data bus for external access to SDRAM. It is in output status when the RDWR# pin is "H" and there is a transparent access request, and it reads SDRAM data and outputs it on this bus. When the RDWR# pin is "L" and there is a transparent access request, the CPU deems it a write to SDRAM, fetches the data on this bus to a buffer and writes it to SDRAM. Only 64-bit bus width is supported for transparent access to SDRAM, and 32, 16 and 8-bit bus widths are not supported. Valid bytes when writing to SDRAM in a 64-bit bus are specified by BE0#~7#. Corresponding BE# inactive ("H") bytes are not written to SDRAM.
	O(Z)	**In Sleep Mode** High-Z Status.
DP0-DP7	I/O	**When the CPU has the Bus Right** Data bus signals. Parity input during reads and parity output during writes. For correspondences with data bus bits, see the ***SPARClite Interface*** Chapter. DP4~DP7 are used In 32-bit bus mode, DP6~DP7 in 16-bit bus mode and DP7 in 8-bit bus mode. When using in 32, 16 and 8-bit bus modes, pull up resistance must be applied for unused DP signals. Parity functions are not supported when using SPARClite buses at 100MHz.
	I/O	**In Bus Grant** Parity functions are not supported when using transparent access functions.
	O(Z)	**In Sleep Mode** High-Z Status.
RDWR#	O	**When the CPU has the Bus Right** Read/write signal. When the current cycle is a bus cycle, "L" is output, and for read and idle cycles "H" is output. Output levels are maintained from start to finish of a bus cycle. Indicates current bus cycle status together with the RD# signal. Write Cycle: RDWR# "L", RD# "H" Read Cycle: RDWR# "H", RD# "L" Idle Cycle: RDWR# "H", RD# "H"
	I	**In Bus Grant** Signal which specifies read/write when there is a transparent access request to SDRAM. If the level of this signal for a cycle in which AS# is asserted is "H" and reads from SDRAM are "L", it becomes a write operation to SDRAM.
	O(Z) (860)	**In Sleep Mode** High-Z Status.
	O(H) (861)	**In Sleep Mode** Outputs "H"
RD#	O	**When the CPU has the Bus Right** Read signal. If the current bus cycle is a read cycle, "L" is output, and "H" is output in write cycle and idle cycle periods. Output levels are maintained from start to finish of a bus cycle.

Pin Name	Normal / In Bus Grant / In Sleep Mode	Explanation
	O(Z)	**In Bus Grant** High-Z Status. This signal is not used for access transparent control.
	O(Z) (860)	**In Sleep Mode** High-Z Status.
	O(H) (861)	**In Sleep Mode** Outputs inactive level "H".
BE0#-BE7#	O	**When the CPU has the Bus Right** Byte enable signals. Outputs "L" to valid bytes in both reads and writes. In 16-bit and 8-bit bus modes BE4# and BE5# are output respectively to ADR<1> and ADR<0>. These signals are valid during bus cycles. If a read misshit occurs in a cache area CPU instructions make external burst transfer requests independently of whether load instructions are bytes, half words, words or double words, and in that case BE# are asserted to all bytes. Also, in cache areas, data merge processing is performed in the buffers, and thus the combinations of BEx# which are asserted during stores are not fixed. In non-cache areas, 1, 2, 4 and 8 BEx# corresponding respectively to byte, half word, word and double word data types are asserted.
	I	**In Bus Grant** Signal which specifies valid byte data when write transparent access to SDRAM is requested. This signal is not used during requests for reads from SDRAM, and all 64-bit (8 byte) data is deemed valid. In writes to SDRAM, byte data which corresponds to active BEx# is written to SDRAM. BEx# input level is decided by the timing of the cycle in which data is fetched from D<63:0>, that is the rise timing of CLKIN in the cycle in which RDYOUT# is asserted. In burst transfer mode also it is likewise decided in the cycle in which RDYOUT# is output.
	O(Z)	**In Sleep Mode** High-Z Status.
	O(H)	**In Sleep Mode** Outputs "H" level.
CS0#-CS5#	O	**When the CPU has the Bus Right** Chip select signals. Asserted when accessing areas set in the ARSR and AMR registers. When the CPU starts external access after a reset, CS0# becomes active, and normally CS0# is used as the EPROM and FLASH ROM chip select signal at boot time. For details, see the explanations of the ARSR and AMR registers. In DMAC operations, CS# signals set in the DMCR Register are asserted. Also, these signals are not relevant when accessing SDRAM areas. (When using transparent mode, internal DMAC cannot be used.) These signals are valid during bus cycles and are negated in idle cycles.
	O(H)	**In Bus Grant** Outputs inactive level "H". Not used in access transparent functions.
	O(H)	**In Sleep Mode** Outputs "H" level.
MEXC#	I	**When the CPU has the Bus Right** Memory access exception pin. If "L" is input to this pin in the same cycle as READY, the CPU handles this as an instruction access exception or a data access exception and generates a trap. Operations in which this signal is asserted in timing other than the same cycle in which READY is input are not guaranteed (If an exception occurs when the ET bit of the PSR is "0", error status results). See the explanation of MEXC# for MEXC# during burst transfers. **In Bus Grant** Reserved pin. **In Sleep Mode** Input status (High-Z)
BREQ#	I	Bus request signal. When this signal is asserted by an external bus master, the CPU judges whether a bus can be opened after the end of the current bus cycle, asserts a BGRNT# signal if it can be opened, and opens a bus. When operating in 64-bit bus width,

Pin Name	Normal / In Bus Grant / In Sleep Mode	Explanation
		the timing in which the CPU opens buses is as shown below: (1) Burst reads from cache areas: opens bus after burst transfer ends. (2) Cache area, but BMACK# is not returned for a BMREQ#: Opens bus upon completion of 4 single transfers. (3) Single read from non-cache area (BMREQ# not asserted): Opens bus upon completion of 1 single transfer. (4) Execution of atomic load/store: Opens bus upon completion of store. (except when an exception occurs from MEXC# in a read cycle)
BGRNT#	O	Bus grant signal. When a bus request (BREQ#) is received, this signal is asserted, and external devices are notified that bus open status is effective.
PBREQ#	O	Processor bus request signal. Asserted when the CPU surrenders the bus right and access to an external bus is required (a cache miss occurs, and external access becomes necessary), and requests the return of a bus to the external bus master. When it receives this request, the external bus master must withdraw bus requests (BREQ#) to the CPU . If the external bus master continues to take the bus right, the CPU holds processing
BMREQ#	O	**When the CPU has the Bus Right** Burst transfer request signal. When the CPU generates a cache miss or a transfer request from the DMAC occurs, this signal is set to "L" and requests a burst transfer . Burst length is 4 when the CPU has the bus right and makes accesses (64-bit bus width).
	I	**In Bus Grant** Burst transfer request signal when a bus master makes transparent access to SDRAM. The CPU judges the level of this signal in cycles in which AS# is asserted, and performs burst transfers if the level is "L". The CPU never cancels burst requests.
	O(Z) (860)	**In Sleep Mode** High-Z Status.
	O(H) (861)	**In Sleep Mode** Outputs inactive "H" level.
BMACK#	I	**When the CPU has the Bus Right** Burst Mode acknowledge input. If "L" level is input in the same cycle as READY# during a burst transfer request (If "L" is output to the BMREQ# pin), burst transfer mode is effective ("L" can either be input in the same cycle as READY#, or "L" can be input starting in the cycle before that and continue until the READY# cycle).
	O	**In Bus Grant** Burst mode acknowledge output. If BMREQ# is input when using the transparent access function to SDRAM, burst mode will definitely be received. The CPU never cancels burst requests.
	O(Z) (860)	**In Sleep Mode** High-Z Status.
	O(Z) (861)	**In Sleep Mode** Outputs inactive "H" level.
BLEN8#	I	**When the CPU has the Bus Right**. Reserved pin. **In Bus Grant** Used together with the BMREQ# pin to specify transparent mode burst transfer length to 8 . When this pin is "H", burst length is deemed to be 4. **In Sleep Mode** Input High-Z status.
SDSEL#	I	**When the CPU has the Bus Right** Reserve pin. **In Bus Grant** Must be asserted together with AS# when requesting transparent access to SDRAM. **In Sleep Mode** Input High-Z status.

Pin Name	Normal / In Bus Grant / In Sleep Mode	Explanation
LOCK#	O	**When the CPU has the Bus Right** Bus lock signal. This signal asserts during execution of Atomic Load/Store instructions and indicates that the current transaction requires multiple transfers which cannot be split. When atomic instructions are executed, opens buses (asserts BGRNT#) when execution of instructions for bus requests (BREQ#) is completed, and thus use of this signal is not required when controlling bus rights in normal use configuration (BREQ#/BGRNT#).
	O(Z)	**In Bus Grant** Reserved pin. This pin is not used by transparent access to SDRAM functions. The LOCK# function is unnecessary, since except when an external bus master asserts BMREQ# the CPU interrupts and does not perform accesses.
	O(Z) (860)	**In Sleep Mode** High-Z status.
	O(H) (861)	**In Sleep Mode** Outputs inactive “H” level.
DTYP<1:0>	O	**When the CPU has the Bus Right** Data type signal. Shows data access types (double word, word, half word, byte). Asserts respectively 11: double word, 10: word, 01: half word, 00: byte.However, during burst transfers and cache area writes, it asserts 11 regardless of the access type.
	O(Z)	**In Bus Grant** High-Z status. This pin is not used by transparent access functions to SDRAM.
	O(Z)	**In Sleep Mode** High-Z status.
	O(Z)	**In Sleep Mode** Outputs “H” level.
READY#	I	**When the CPU has the Bus Right** READY input pin. “L” should be output to this pin when data has been provided to a data bus in read cycles and in write cycles when writes are made. In single transfers, the CPU opens the next bus cycle when it receives READY#=“L”. In burst transfers, a prescribed number of READYs must be input for each address strobe assertion **In Bus Grant** Not used. “H” should be input. **In Sleep Mode** Input High-Z. “H” should be input.
RDYOUT#	O	**When the CPU has the Bus Right** Internally generated READY output signal. Outputs READY signals generated by the built-in wait state generator. **In Bus Grant** In reads from SDRAM, outputs “L” at the point in time when data has been provided to a data bus. When writing to SDRAM, outputs “L” at the point in time when data is fetched to the buffers in the CPU. **In Sleep Mode** Outputs inactive level “H”.
ERROR#	O	Error signal. This signal indicates that a trap has occurred in trap disable (a double exception has occurred), and the CPU has stopped in error status. Error status can only be released by a reset.
BMODE16# BMODE32#	I	Boot mode bus width set signals. Specifies CS0# area (during boot) SPARClite bus width. The configuration is (BMODE16#, BMODE32#)=(0,0): 8-bit, (0,1): 16-bit, (1,0): 32-bit, (1,1): 64-bit. These signals are sampled during resets.

2.4.2. SDRAM-IF Signals

Table 2-6 SDRAM Signals

Pin Name	Normal / In Sleep Mode	Explanation
SCLK	O	**In Normal Operation** SDRAM clock output signal. Should be linked to SDRAM clock input. When the load is 4 pins or more it should be buffered by the Zero Propagation Delay Buffer which has built-in PLL. **In Sleep Mode** Outputs “L”.
CKE	O	**In Normal Operation** SDRAM CKE signal. Should be buffered by register devices when the load exceeds 4 devices. **In Sleep Mode** Outputs “H”.
SRAS#	O	**In Normal Operation** SDRAM RAS signal. Should be buffered by register devices when the load exceeds 4 devices. **In Sleep Mode** Outputs inactive “H”.
SCAS#	O	**In Normal Operation** SDRAM CAS signal. Should be buffered by register devices when the load exceeds 4 devices. **In Sleep Mode** Outputs inactive “H”.
SWE#	O	**In Normal Operation** SDRAM WE signal. Should be buffered by register devices when the load exceeds 4 devices. **In Sleep Mode** Outputs inactive “H”.
SCS0#~3#	O	**In Normal Operation** SDRAM chip select signals. Should be buffered by register devices when the load exceeds 4 devices. **In Sleep Mode** Outputs inactive “H”.
SBA<1:0>	O	**In Normal Operation** SDRAM bank select signal. Should be buffered by register devices when the load exceeds 4 devices. **In Sleep Mode** Fixed to “H” or “L”.
SADR<12:0>	O	SDRAM address signal. Row and column addresses are multiplexed and output. Should be buffered by register devices when the load exceeds 4 devices. **In Sleep Mode** Fixed to “H” or “L”.
SDQ<63:0>	I/O	**In Normal Operation** SDRAM data bus and bi-directional signal. **In Sleep Mode** Outputs “H” or “L” level.
SDP0~7	I/O	**In Normal Operation** SDRAM parity and bi-directional signal. SDP0,....,7 correspond respectively to each SDQ<63:56>,....,SDQ<7:0> byte data. **In Sleep Mode** Outputs “H” or “L” level.
SDQM0~7	I/O	**In Normal Operation** SDRAM input mask/output enable signal. **In Sleep Mode** Outputs “H”.

2.4.3. Interrupt Signals

Table 2-7 Interrupt Signals

Pin Name	I/O	Explanation
IRL<3:0>	I	Interrupt input pins. This is a signal for inputting encoded interrupt levels. These pins are grouped asynchronous input signals, and are first communicated to the IU (Integer Unit) when the same level is detected twice at external clock rise. IRL=$0000_{(2)}$ indicates a status of no interrupts, and IRL=$1111_{(2)}$ is designated as non-maskable interrupts by the SPARC architecture. IRLs decide priorities in external circuits and must be stored until confirmed by the CPU.

2.4.4. DMAC, DSU and Sleep Mode-related Signals

Table 2-8 DMAC, DSU and Sleep Mode

Pin Name	I/O	Explanation
EOP#	O	End-of-process signal. When a DMA transfer ends, indicates that “L” is output, and the transfer has ended. Internal DMAC cannot be used when transparent mode is used.
PDOWN#	O	Sleep mode output pin. Indicates by outputting “L” level that a move to Sleep mode (low power consumption mode) has been completed. When “L” is input to this pin, SPARClite buses are also in open status.
WKUP#	I	Sleep Mode cancel/release pin. When “L” is input to this pin, CPU Sleep Mode is canceled, and operation starts. This pin is asynchronous input, and “L” width of at least 2 clock periods in CLKIN is required. ”L” should be input to this pin only when PDOWN# is “L”.
STOP#	I	Internal clock stop signal. If “L” is input in Sleep Mode, Stop Mode becomes effective, PLL stops, and all internal clocks stop.
BRKEN#	I	Break enable pin. “L” should be input when debug functions are used. This signal is sampled during reset periods. To enable the built in CPU Debug Support Function, setup time and hold time of at least 2 clock periods in CLKIN for the rise of RESET# are required.
DBREAK#	I	Debug Break pin. Inputting “L” to this pin during Debug Mode causes a trap. This pin is asynchronous input, and “L” width of at least 2 clock periods in CLKIN is required. When it is desired to jump to a break processing routine right after a reset, setup time and hold time of at least 2 clock periods in CLKIN for the rise of RESET# are required.
BRKGO	O	Break Go pin. When this signal is “H”, it indicates that a break processing routine for debugging is running.

2.4.5. Other Signals

Table 2-9 Others

Pin Name	I/O	Explanation
RESET#	I	Reset input. CPU is placed in initial status by inputting “L” to this pin.
CLKSEL0 CLKSEL1 (*1)	I	These pins set input clock frequency multiplier modes. Operating clocks can be set to 1, 2, 3 and 4 times the input clock. (CLKSEL1, CLKSEL0)=(0,0): X 1 mode, (0,1): X 2 mode, (1,0): X 3 mode and (1,1): X 4 mode.

(*1) Settings which can use CLOSEL0 and CLKSEL1 differ by type.

Table 2-10 CLKSEL0,1 and Corresponding Clock Multiplier Numbers

CLKSEL0	CLKSEL1	MB86860	MB86861	MB86862
0	0	Use prohibited	X1	Use prohibited
0	1	X2	Use prohibited	X2

1	0	X3	Use prohibited	X3
1	1	X4	Use prohibited	X4

2.4.6. Test Signals

Table 2-11 Test Signals

Pin Name	I/O	Explanation
FLOAT#	I	Pin float input. By fixing this pin to “L”, all output pins and bi-directional pins except SDRAM-IF are set to High-Z status.
BEN#	I	PLL bypass enable input. When this signal is “L”, PLL bypass mode is effective. Should normally be fixed to “H” for use as a test pin.
BCLK	I	PLL bypass clock. When BEN# is “L” and bypass mode is set, clocks to the CPU core are provided from this pin.
PLLCEN#	I	PLLCLK enable pin. If “L” is input to this pin, it outputs a PLL clock to the PLLCLK pin. Should normally be fixed to “H” for use as a test pin.
TEST1	I	Test pin. Should be fixed to “H” in normal use.
TEST2	I	Test pin. Should be fixed to “H” in normal use.
TDI	I	Reserve input. Should be fixed to “H”.
TMS	I	Reserve input.
TDO	O	Reserve output. Should be fixed to “H”.
TCLK	I	Reserve input. Should be fixed to “H”.
TRST#	I	Reserve input. Should be fixed to “H”.

O : Indicates output status
O(Z) : Output pin (High-Z)
O(H) : “H” output
O(L) : “L” output
I : Input status and High-Z

3. Registers

MB8686 processor register settings can be classified into the following 3 types:

- IU r-register
  32-bit register which can be used for general purposes

- IU Status/Control Register
  Used for IU status display and control

- Memory-Mapped Registers
  Used for processor or system control/status display

- IU r-register
  32-bit general purpose register.

The SPARC architecture in the MB8686 processor provides for 8 groups of specified register windows, for a total of (8+16*8)=136 general purpose registers which are implemented (for details about register windows, see the *MB86860 SeriesProgramming Manual 1, Programming Models).*

Table 3-1 IU r-registers

Register Name	Symbol
Global Register	r[0]-r[7] (g[0]-g[7])
Out Register	r[8]-r[15] (o[0]-o[7])
Local Register	r[16]-r[23] (l[0]-g[7])
In Register	r[24]-r[31] (i[0]-i[7])

3.1. IU Status / Control Registers

This register is used for processor status display and control. In the MB8686 processor, the ASR30 (DIAG Register) and the ASR31 (Instruction Cache Control Register) are implemented as Ancillary State registers.

Table 3-2 IU Control / Status Registers

Register Name	Symbol
Program Counter	PC
next Program Counter	nPC
Processor Status Register	PS
Window Invalid Register	WIM
Trap Base Register	TBR
Y Register	Y
Ancillary State Register 30	ASR30(DIAG)
Ancillary State Register 31 (Instruction Cache Control Register)	ASR31(ICCR)

3.2. Memory Mapped Registers

These are registers located in memory address spaces. They are used for processor and system status display and control. The registers shown below can be accessed by the Load from Alternate Space (LDA) instruction and the Store into Alternate Space (STA) instruction.
However, the registers shown below should be accessed as word data. By accessing them as word data, assignment to the same addresses is guaranteed without regard to Big-endian Mode or Little-endian Mode. (Use of LDAand STA instructions)

Table 3-3 Memory-mapped Registers

ASI	Address	Register Name	Symbol	Class
0x01	0x00002004	Sleep mode register	SMR	CORE
0x01	0x0000FF00	Instruction Address Descriptor Register	IADR1	DSU
0x01	0x0000FF04	Instruction Address Descriptor Register	IADR2	DSU
0x01	0x0000FF18	Debug Control Register	DCR	DSU
0x01	0x0000FF1C	Debug Status Register	DSR	DSU
0x01	0x0000FF40 | 0x0000FF7C	PC Trace buffer	TB	DSU
0x04	0x80000000	Buffer Control Register	BCR	DBU
0x04	0x80000008	TLB Control Register	TCR	TLB
0x04	0x80000100	Address Range Specifier Register0	ARSR0	SP
0x04	0x80000108	Address Range Specifier Register1	ARSR1	SP
0x04	0x80000110	Address Range Specifier Register2	ARSR2	SP
0x04	0x80000118	Address Range Specifier Register3	ARSR3	SP
0x04	0x80000120	Address Range Specifier Register4	ARSR4	SP
0x04	0x80000128	Address Range Specifier Register5	ARSR5	SP
0x04	0x80000130	SDRAM Address Range Specifier Register0	SDARSR0	SDRAM
0x04	0x80000138	SDRAM Address Range Specifier Register1	SDARSR1	SDRAM
0x04	0x80000200	Address Mask Register0	AMR0	SP
0x04	0x80000208	Address Mask Register1	AMR1	SP
0x04	0x80000210	Address Mask Register2	AMR2	SP
0x04	0x80000218	Address Mask Register3	AMR3	SP
0x04	0x80000220	Address Mask Register4	AMR4	SP
0x04	0x80000228	Address Mask Register5	AMR5	SP
0x04	0X80000230	SDRAM Address Mask Register0	SDAMR0	SDRAM
0x04	0X80000238	SDRAM Address Mask Register1	SDAMR1	SDRAM
0x04	0x80000400	Wait State Specifier Register0	WSSR0	SP
0x04	0x80000408	Wait State Specifier Register1	WSSR1	SP
0x04	0x80000410	Wait State Specifier Register2	WSSR2	SP
0x04	0x80000418	Wait State Specifier Register3	WSSR3	SP
0x04	0x80000420	Wait State Specifier Register4	WSSR4	SP
0x04	0x80000428	Wait State Specifier Register5	WSSR5	SP
0x04	0x80000430	MEXC Parity Error Flag Register	MXPEF	SP
0x04	0x80000438	MEXC Parity Error Control Register	MXPECR	SP
0x04	0x80000440	Idle Cycle Control Register	IDCCR	SP
0x04	0x80000800	SDRAM I/F Configuration Register	SCR	SDRAM
0x04	0x80000808	Auto Refresh Timer Register	ART	SDRAM
0x04	0x80000810	SDRAM Status Register	SSR	SDRAM
0x04	0x80000820	SDRAM CS Characteristic Register0	CSCR0	SDRAM
0x04	0x80000828	SDRAM CS Characteristic Register1	CSCR1	SDRAM
0x04	0x80000830	SDRAM CS Characteristic Register2	CSCR2	SDRAM
0x04	0x80000838	SDRAM CS Characteristic Register3	CSCR3	SDRAM

0x04	0x80000840	SDRAM START Address Register0	SSAR0	SDRAM
0x04	0x80000848	SDRAM START Address Register1	SSAR1	SDRAM
0x04	0x80000850	SDRAM START Address Register2	SSAR2	SDRAM
0x04	0x80000858	SDRAM START Address Register3	SSAR3	SDRAM
0x04	0x80000860	SDRAM Address Mask Register0	SAMR0	SDRAM
ASI	Address	Register Name	Symbol	Class
0x04	0x80000868	SDRAM Address Mask Register1	SAMR1	SDRAM
0x04	0x80000870	SDRAM Address Mask Register2	SAMR2	SDRAM
0x04	0x80000878	SDRAM Address Mask Register3	SAMR3	SDRAM
0x04	0x80000C00	DMA Control Register0	DMCR0	DMAC
0x04	0x80000C08	DMA Source Address Register0	DMSAR0	DMAC
0x04	0x80000C10	DMA Destination Address Register0	DMDAR0	DMAC
0x04	0x80000C18	DMA Word Length Register0	DMWL0	DMAC
0x04	0x80000C20	DMA Control Register1	DMCR1	DMAC
0x04	0x80000C28	DMA Source Address Register1	DMSAR1	DMAC
0x04	0x80000C30	DMA Destination Address Register1	DMDAR1	DMAC
0x04	0x80000C38	DMA Word Length Register1	DMWL1	DMAC
0x04	0x80000FF0	ID Register	IDR	CORE
0x04	0x80001000	BTC Control Register	BTCR	SP
0x04	0x80001400	Virtual Word Address Entry Register00	VWE00	TLB
0x04	0x80001408	Physical Word Address Entry Register00	PWE00	TLB
TLB	0x80001410	Virtual Word Address Entry Register01	VWE01	TLB
0x04	0x80001418	Physical Word Address Entry Register01	PWE01	TLB
0x04	0x80001420	Virtual Word Address Entry Register02	VWE02	TLB
0x04	0x80001428	Physical Word Address Entry Register02	PWE00	TLB
0x04	0x80001430	Virtual Word Address Entry Register03	VWE03	TLB
0x04	0x80001438	Physical Word Address Entry Register03	PWE03	TLB
0x04	0x80001440	Virtual Word Address Entry Register04	VWE04	TLB
0x04	0x80001448	Physical Word Address Entry Register04	PWE04	TLB
0x04	0x80001450	Virtual Word Address Entry Register05	VWE05	TLB
0x04	0x80001458	Physical Word Address Entry Register05	PWE05	TLB
0x04	0x80001460	Virtual Word Address Entry Register06	VWE06	TLB
0x04	0x80001468	Physical Word Address Entry Register06	PWE06	TLB
0x04	0x80001470	Virtual Word Address Entry Register07	VWE07	TLB
0x04	0x80001478	Physical Word Address Entry Register07	PWE07	TLB
0x04	0x80001480	Virtual Word Address Entry Register08	VWE08	TLB
0x04	0x80001488	Physical Word Address Entry Register08	PWE08	TLB
0x04	0x80001490	Virtual Word Address Entry Register09	VWE09	TLB
0x04	0x80001498	Physical Word Address Entry Register09	PWE09	TLB
0x04	0x800014A0	Virtual Word Address Entry Register10	VWE10	TLB
0x04	0x800014A8	Physical Word Address Entry Register10	PWE10	TLB
0x04	0x800014B0	Virtual Word Address Entry Register11	VWE11	TLB
0x04	0x800014B8	Physical Word Address Entry Register11	PWE11	TLB
0x04	0x800014C0	Virtual Word Address Entry Register12	VWE12	TLB
0x04	0x800014C8	Physical Word Address Entry Register12	PWE12	TLB
0x04	0x800014D0	Virtual Word Address Entry Register13	VWE13	TLB
0x04	0x800014D8	Physical Word Address Entry Register13	PWE13	TLB
0x04	0x800014E0	Virtual Word Address Entry Register14	VWE14	TLB
0x04	0x800014E8	Physical Word Address Entry Register14	PWE14	TLB
0x04	0x800014F0	Virtual Word Address Entry Register15	VWE15	TLB
0x04	0x800014F8	Physical Word Address Entry Register15	PWE15	TLB
0x0c	0x00000000 | 0x00000FE0	ICache Tag Diagnostics set 0	ICTAG0	ICache

0x0c	0x00001000 | 0x00001FE0	ICache Tag Diagnostics set 1	ICTAG1	ICache
0x0c	0x00001000 | 0x00001FE0	ICache Tag Diagnostics set 2	ICTAG2	ICache
0x0c	0x00001000 | 0x00001FE0	ICache Tag Diagnostics set 3	ICTAG3	ICache
0x0d	0x00001000 | 0x00001FF8	ICache Data Diagnostics set 0	ICDATA0	ICache

ASI	Address	Register Name	Symbol	Class
0x0d	0x00002000 | 0x00002FF8	ICache Data Diagnostics set 1	ICDATA1	ICache
0x0d	0x00003000 | 0x00003FF8	ICache Data Diagnostics set 2	ICDATA2	ICache
0x0d	0x00004000 | 0x00004FF8	ICache Data Diagnostics set 3	ICDATA3	ICache
0x1c	0x00000000 | 0x00000FE0	DCache Tag Diagnostics set 0	DCTAG0	DCache
0x1c	0x00001000 | 0x00001FE0	DCache Tag Diagnostics set 1	DCTAG1	DCache
0x1c	0x00001000 | 0x00001FE0	DCache Tag Diagnostics set 2	DCTAG2	DCache
0x1c	0x00000000 | 0x00000FE0	DCache Tag Diagnostics set 3	DCTAG3	DCache
0x1d	0x00001000 | 0x00001FF8	DCache Data Diagnostics set 0	DCDATA0	DCache
0x1d	0x00002000 | 0x00002FF8	DCache Data Diagnostics set 1	DCDATA1	DCache
0x1d	0x00003000 | 0x00003FF8	DCache Data Diagnostics set 2	DCDATA2	DCache
0x1d	0x00004000 | 0x00004FF8	DCache Data Diagnostics set 3	DCDATA3	DCache

Shaded parts indicate items with which the MB8686 is not actually equipped.

[Class]

CORE	:MB8686x Core
SPB	:SPARClite-Bus I/F
DBU	:Data Buffer Unit
SDRAM	:SDRAM I/F
DMAC	:DMA Controller
DSU	:Debug Support Unit
ICache	:Instruction Cache
DCache	:Data Cache

4. Processor Core

4.1. Summary

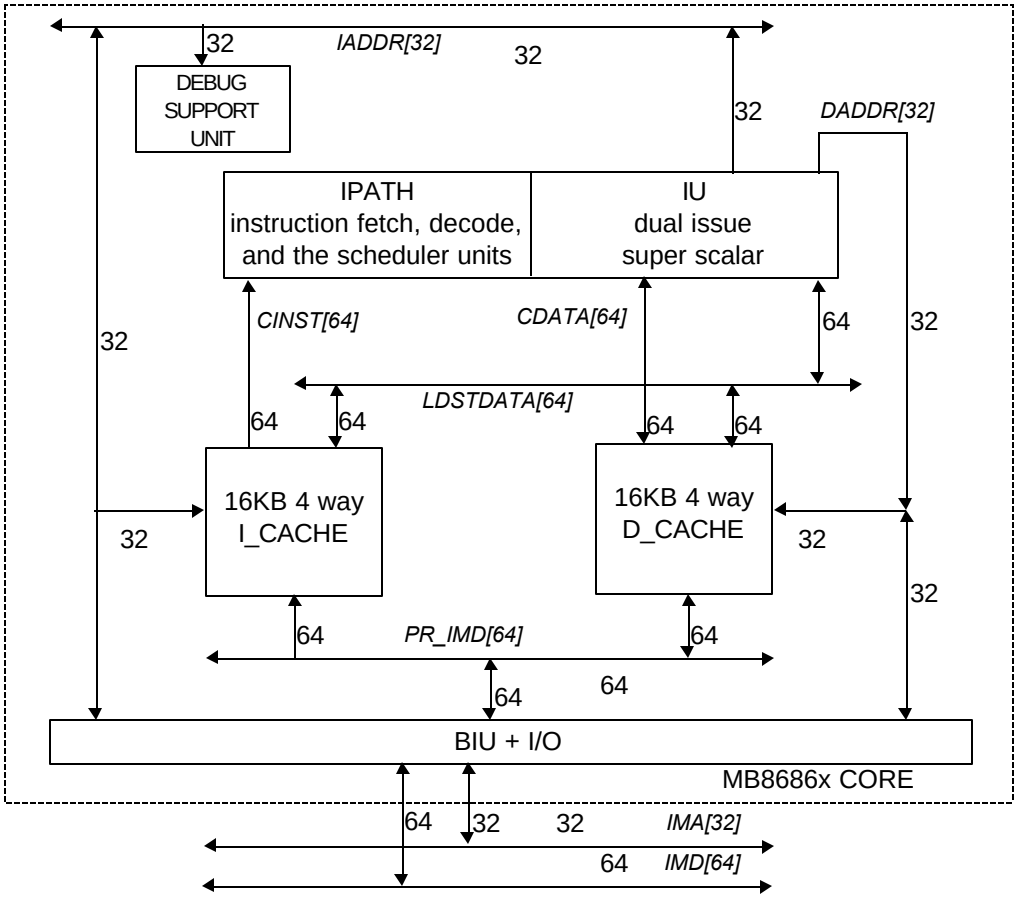
The MB8686 Processor Core (called the MB8686x Core below) is a High-end embedded SPARC processor Core developed for installation on the basis of the HyperSPARC (RT6xx). The MB6868 Core is configured from the
following function blocks:

- IU (Integer Unit)
  The MB8686 IU uses 2-issue Superscalar Architecture. It simultaneously fetches two 32-bit instructions using a 64-bit bus, and simultaneously executes 2 instructions in 2 ALUs (simultaneously executable instructions are partially restricted).

- Caches
  Instruction Cache — 16Kbyte/4-way set-associative
  Data Cache — 16Kbyte/4-way set-associative (write through system)

- IMB (Intra Module Bus) Interface
  This module controls internal buses which connect external modules with the core part. An IMB bus is composed of a 32-bit address bus, a 64-bit data bus and control signals, and the data bus supports Bi-endian.

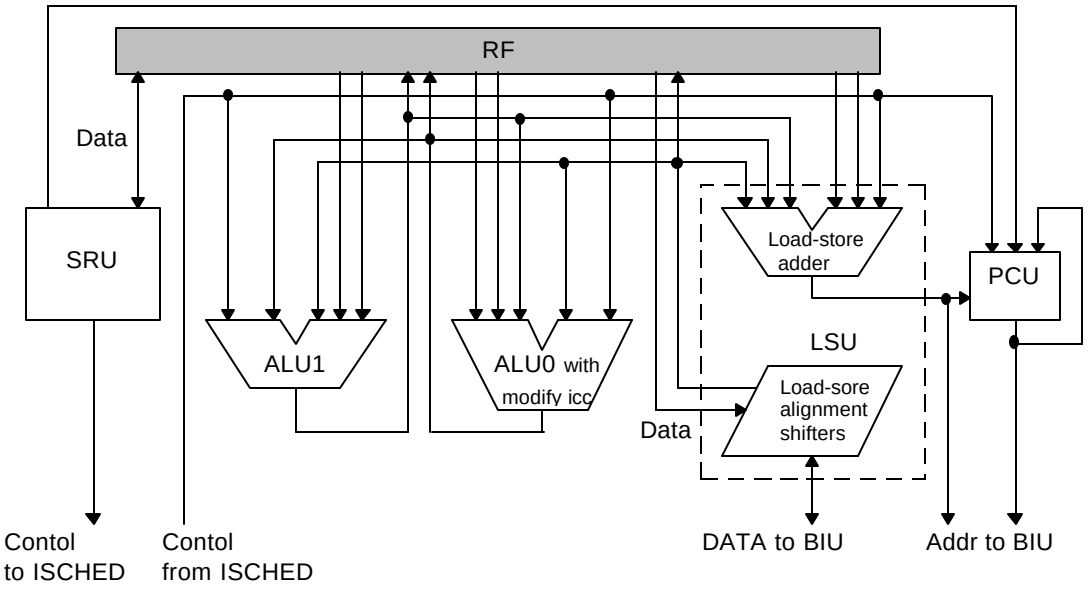
- DSU (Debug Support Unit)
  Module for support of debugging. Offers asynchronous debug traps by external inputs, internal hardware breaks and software and other functions.


32
IADDR[32]
32
DEBUG
SUPPORT
UNIT
32
DADDR[32]
IPATH
instruction fetch, decode,
and the scheduler units
IU
dual issue
super scalar
CINST[64]
CDATA[64]
64
32
32
LDSTDATA[64]
64
64
64
64
16KB 4 way
I_CACHE
16KB 4 way
D_CACHE
32
32
32
64
PR_IMD[64]
64
64
64
BIU + I/O
MB8686x CORE
64
32
32
IMA[32]
64
IMD[64]

4.2.1. Summary

Here a block diagram of the IU in the MB8686x Core is shown. The MB8686x processor Core uses super-scalar architecture, and it can both fetch 2 instructions at a time and issue 2 instructions at a time.

Figure 4-2 MB8686x Core IU Block Diagram



4.2.2. Register Overview

Table 4-1 IU r-registers Figure

ASI	Address	Register Name	Symbol	Reset Value			
				b31 b24	b23 b16	b15 b8	b7 b0
–	–	Global Register	r[0]	00000000	00000000	00000000	00000000
–	–		r[1] - r[7]	xxxxxxxx	xxxxxxxx	xxxxxxxx	xxxxxxxx
–	–	Out-register	r[8] - r[15]	xxxxxxxx	xxxxxxxx	xxxxxxxx	xxxxxxxx
–	–	Local-register	r[16] - r[23]	xxxxxxxx	xxxxxxxx	xxxxxxxx	xxxxxxxx
–	–	In-register	r[24] - r[31]	xxxxxxxx	xxxxxxxx	xxxxxxxx	xxxxxxxx

Table 4-2 IU Control/Status Registers

ASI	Address	Register Name	Symbol	Reset Value			
				b31 b24	b23 b16	b15 b8	b7 b0
–	–	Processor Status Register	PSR	00011111	xxxxrrrr	0r00xxxx	1x0xxxxx
–	–	Window Invalid Mask Register	WIM	xxxxxxxx	xxxxxxxx	xxxxxxxx	xxxxxxxx
–	–	Trap Base Register	TBR	xxxxxxxx	xxxxxxxx	xxxxxxxx	xxxxxxxx
–	–	Y Register	Y	xxxxxxxx	xxxxxxxx	xxxxxxxx	xxxxxxxx
–	–	Ancillary State Register 30	ASR30 (DIAG)	10000000	00000000	01010001	00000000

–	–	Ancillary State Register 31	ASR31 (ICCR)	00000000	00000000	00000000	00000000

4.2.3. Register Details

4.2.3.1. IU r-register

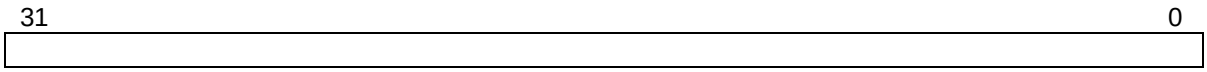


Figure 4-3 IU r-register

The IU Register is a 32-bit general purpose register.

4.2.3.2. Program Counter (PC)

The PC indicates the instruction addresses fetched in the IU (Integer Unit). The PC fetches the contents of nPC at the end of each instruction. The PC cannot be directly accessed. However, in the following cases the PC contents are saved to general purpose registers:

- Execution of a CALL instruction
  When a CALL instruction is executed, PC contents (CALL instruction address) are written to out[7](r[15])
- Occurrence of Traps / Interrupts
  If traps or interrupts occur, PC (Program Counter) contents and nPC (Next Program Counter) contents are written to local[1](r[17]) and local[2](r[18]) of a new window after the current window pointer is decremented.

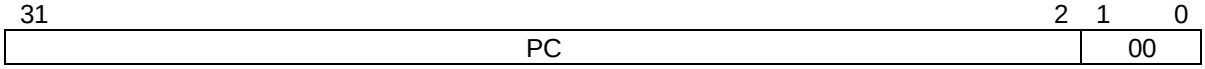


Figure 4-4 Program Count

4.2.3.3. next Program Counter (nPC)

nPC indicates the next instruction address to be fetched. Contents of nPC are updated as follows at the end of each instruction :

- When control does not move
  Incremented value of PC contents + 4.
- PC relative control move
  Value which adds the contents of instruction immediate data fields (offsets) to PC contents.
- Control move by JMPL instruction or RETT instruction
  Values specified by operands
- Control move by Ticc instruction or trap / interrupt
  Contents of register TBR becomes nPC value. Register TBR consists of TBA (Trap Base address) field, tt(Trap Type) field and Null field.

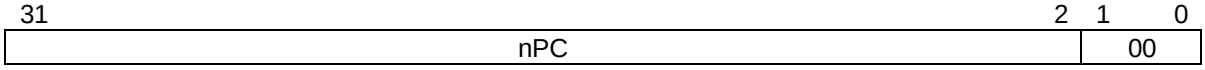


Figure 4-5 next Program Count

Symbol :nPC
Reset State :0x4

4.2.3.4. Processor State Register (PSR)

Register PS displays basic processor control and status. Register PS can be accessed by instruction RDPS (privileged instruction) and instruction WRPS (privileged instruction). In order to support the Little-endian function, the PSR.DE field has been added to the MB8686 processor.

31 28	27 24	23 20	19 16	15	14	13	12	11 8	7	6	5	4 0
impl	ver	icc	R	DE	R	EC	EF	PIL	S	PS	ET	CWP

Symbol :PS
Reset State

bit	Field Name	R/W	Description
31-28	impl	R	-Implementation Indicates implementation number Fixed to “0001.
27-24	ver	R	- Version Indicates Version number Fixed to “1111.
23-20	icc		- Integer Condition Code Integer Condition Code Changes in accordance with results of arithmetic operation instructions (cc attached operation instructions) and logic operation instructions which change integer condition codes. Bicc and Ticc instructions change control in accordance with the status of this icc. bit23:n(negative) bit22:z(zero) bit21:v(overflow) bit20:c(carry)
19-16	Reserved	R	
15	DE	R/W	- Default Endian Sets endian in data access. 0:Big Endian 1:Little Endian Reset state:0
14	Reserved	R	- Reserved
13	EC	R	- Enable Coprocessor Coprocessor functions are not supported by the MB8686 processor. This bit is fixed to “0”.
12	EF	R	- Enable Floating-Point Unit The MB8686 processor is not equipped with a floating point operation unit. This bit is fixed to “0”.
11-8	PIL	R/W	- Processor Interrupt Level Sets the interrupt level received by the processor. The processor receives only interrupts having level 15 or levels which are greater than the values in PIL fields. bit11 - MS bit8 - LS
7	S	R/W	- Supervisor 0:User mode 1:Supervisor mode

6	PS	R	- Previous Supervisor Preserves the value of the PSR.S bit when traps occur. PS bit is written back to S bit by a return from a trap (RETT instruction)

bit	Field Name	R/W	Description
5	ET	R/W	- Enable Traps Interrupt traps are enabled by writing “1” to this bit. 0: Interrupts prohibited (interrupt traps ignored) 1: Interrupts enabled
4-0	CWP	R/W	- Current Window Pointer Pointer which indicates current active register window. The range which can be set in the MB8686x processors is bit0-7. Writes to bits other than these are ignored.

Figure 4-6 Processor Status Register

4.2.3.5. Window Invalid Mask Register (WIM)

Register WIM is the register which specifies trap generation when the current window is updated. It is used for the prevention of underflows and overflows. Register WIM can be accessed by instruction RDWIM (privileged instruction) and instruction WRWIM (privileged instruction). This register contains 8 window mask bits which correspond respectively to its 8 windows.

If the window mask bits (W7~W0) which correspond to the windows indicated by CWP fields after execution are “1”when the contents of the CWP field of Register PS are decremented by the SAVE instruction, a window overflow trap occurs. A window underflow trap occurs if the window mask bits (W7~W0) which correspond to the windows indicated by CWP fields after execution are “1”when the contents of the CWP field of Register PS are incremented by the RESTORE instruction.

31 … 8	7	6	5	4	3	2	1	0
	w7	w6	w5	w4	w3	w2	w1	w0

Symbol :WIM
Reset State :undefined

bit	Field Name	R/W	Description
31-8	Reserved		
7-0	w0-w7	R/W	- Window Invalid Mask Each bit of w0-w7 corresponds respectively to the register window in which it is mounted. If the WIM bits corresponding to PSR.CWP which is changed by the SAVE, RESTORE or RERR instructions, window overflow or window underflow traps occur.

Figure 4-7 Window Invalid Mask Register

4.2.3.6. Trap Base Register

This register is for setting and generation of service routine start addresses which correspond to traps and interrupts. When traps and interrupts occur, the processor starts service program execution from an address indicated by this register. Register TBR can be accessed by the RDTBR instruction (privileged instruction) and the WRTBR instruction (privileged instruction). However, the only fields which can be written are TBA fields.

31 … 12	11 … 4	3 … 0

TBA<31:12>	tt	zeros

Symbol :WIM
Reset State :undefined

Table 4-1

bit	Field Name	R/W	Description
31-12	TBA	R/W	- Trap Base Address Sets the upper order 20b of the trap table base address.
11-4	tt	R/W	- trap type When traps occur, offset values are set to the trap tables corresponding to the traps.
3-0	zeros		This field is fixed to “0000.

Figure 4-8 Trap Base Register

4.2.3.7. Y-register (Y)

The Y-register is used by Integer Multiply instructions and Integer Divide instructions. When Integer Multiply instructions (UMUL, UMULcc, SUML, SMULcc) are executed, the upper order 32b of the product of 64b generated by a 32b X32-bit operation is stored in the Y-register, and the lower order 32-bit is stored in a destination register. When Integer Divide instructions (UDIV, UDIVcc, SDIV, SDIVcc) are executed, the upper order 32bit of the dividend is stored in the Y-register and used in the 64bit ÷ 32bit operation. Reads and writes can be done with this register using the RDY and WRY instructions.

31		0
	Y	

Symbol :Y
Reset State :undefined

bit	Field Name	R/W	Description
31-0	Y	R/W	- 32-bit Register Used by Integer Multiply instructions and Integer Divide instructions to generate 64b results.

Figure 4-9 Y Register

4.2.3.8. DIAG register (ASR30)

31	30	29	28	27	26 16	15 8	7 1	0
ATE	ICD	DCD	SIFD	SDFD	Reserved	01010001	Reserved	FPT

Symbol :DIAG or ASR30
Reset State :0x80005100

bit	Field Name	R/W	Description
31	ATE	R/W	- ALU TwoEnable Reset state: 1
30	ICD	R/W	- ICACHE Disable 0: ICache enable 1: ICache disable Reset state: 0
29	DCD	R/W	- DCache Disable 0: DCache enable 1: DCache disable Reset state: 0
28	SIFD	R/W	- Speculative Instruction Fetch Disable Reset state: 0
27	SDFD	R/W	- Speculative Data Fetch Disable Reset state: 0
26-16	Reserved	R	-
15-8		R	Fixed to “01010001.
7-1	Reserved	R/W	-
0	FPT	R/W	- FPU test mode enable Reset state: 0

Figure 4-10 DIAG Register

[**Note**] Bits FPT, SIFD, SDFD and ATE are only used internally. These bits should not be changed in normal operation.

4.2.3.9. Instruction Cache Control Register (ICCR) (ASR31)

This register is used for ICache and DCache control. The followig conditions must be met to turn ICache/DCache on:

ICache ON: ICCR.CE == 1 and DIAG.ICD == 0
DCache ON: ICCR.CE == 1 and DIAG.DCD == 0

31 2	1	0
Reserved	IFTD	CE

Symbol :ICCR or ASR31
Reset State :0x00000000

bit	Field Name	R/W	Description
31-2	Reserved	R	
30	IFTD	R/W	- ICACHE Flush Traps Disable

			0: By executing Flush Instruction 1: issued.

29	CE	R/W	- CACHE Enable for both ICache and DCache Enables ICache and DCache 0: ICache/DCache disable 1: ICache/DCache enable Reset state: 0

Figure 4-11 Instruction Cache Control Register

4.3. Caches

4.3.1. Summary

MB8686 processors are equipped with instruction and data caches which are each 16Kbyte (4-way set associative), and the data caches use one-way type write through.

4.3.2. Cache Memory Assignments

Table 4-3 Cache ASI Assignments

ASI	Address	Register Name	Symbol	Reset Value							
				b31	b24	b23	b16	b15	b8	b7	b0
0x0c	0x00000000 | 0x00000FE0	ICache Tag Diagnostics Set 0	ICTAG0								
0x0c	0x00000000 | 0x00000FE0	ICache Tag Diagnostics Set 1	ICTAG1								
0x0c	0x00000000 | 0x00000FE0	ICache Tag Diagnostics Set 2	ICTAG2								
0x0c	0x00000000 | 0x00000FE0	ICache Tag Diagnostics Set 3	ICTAG3								
0x0d	0x00000000 | 0x00000FF8	ICache Data Diagnostics Set 0	ICDATA0								
0x0d	0x00000000 | 0x00000FF8	ICache Data Diagnostics Set 1	ICDATA1								
0x0d	0x00000000 | 0x00000FF8	ICache Data Diagnostics Set 2	ICDATA2								
0x0d	0x00000000 | 0x00000FF8	ICache Data Diagnostics Set 3	ICDATA3								
0x10 | 0x14		ICache/DCache Line Flush									
0x18 | 0x1b		ICache Line Flush									
0x1c	0x00000000 | 0x00000FE0	DCache Tag Diagnostics Set 0	DTAG0								
0x1c	0x00000000 | 0x00000FE0	DCache Tag Diagnostics Set 1	DTAG1								
0x1c	0x00000000 | 0x00000FE0	DCache Tag Diagnostics Set 2	DTAG2								
0x1c	0x00000000 | 0x00000FE0	DCache Tag Diagnostics Set 3	DTAG3								
0x1d	0x00000000										

	| 0x00000FF8	DCache Data Diagnostics Set 0	DDATA0				
0x1d	0x00000000 | 0x00000FF8	DCache Data Diagnostics Set 1	DDATA1				

0x1d	0x00000000 | 0x00000FF8	DCache Data Diagnostics Set 2	DDATA2				
0x1d	0x00000000 | 0x00000FF8	DCache Data Diagnostics Set 3	DDATA3				
0x31		Flush Entire ICache/DCache					

4.3.3. Tag/Data Format

4.3.4. Operating Functions

4.3.4.1. Cache Control

ASR30(DIAG) and ASR31(DIAG) registers (see 4.2.3.8 and 4.2.3.9) are used to control cache operations. The settings shown below are required to enable cache functions. It is also necessary to set DBU (Data Buffer Unit) fuctions for valid use of cache functions (see *6. DBU*).

Table 4-4 ICache/DCache Enable Conditions

ICache Enable	ASR31.CE=1 & ASR30.ICD=0
DCache Enable	ASR31.CE=1 & ASR30.DCD=0

4.3.4.2. Cache Flushes

Cache flush methods are as shown below:

(1) For entire flushes of ICache and DCache:
Write access ASI=0x31. Write address can be any value.

(2) Entire ICache Flush
For entire ICache flushes, line-by-line writes must be made to tags. Use ICache Tag Diagnostics (ASI=0x0c) and write "0" to the S, V0, V1, V2, V3, Wu and Ws bits of all the tags of each set (0,1,2,3). Specify which lines to flush by address. The number of Tag lines is 128, and besides the tags it is necessary to clear Load Buffer-0 and Load Buffer-1. Instruction cache tags do not have Wu and Ws bits.

(3) Entire DCache[1] Flush
For entire DCache[1] flushes, line-by-line writes must be made to tags. Use DCache Tag Diagnostics (ASI=0x0d) and write "0" to the S, V0, V1, V2, V3, Wu and Ws bits of all the tags of each set (0,1,2,3). Specify which lines to flush by address. The number of Tag lines is 12, and besides the tags it is necessary to clear Store Buffer and Load Buffer-0 and Load Buffer-1.

(4) ICache and DCache Lines Cleared where Access Address and Tags Match
There is a function that compares addresses held by ICache and DCache tags and access addresses and clears them if they match. When ASI=0x10 is accessed and A<19:5> of the access address matches, matching lines of I$ are cleared. This does not depend on the bank. Any of ASI=0x11, 0x12, 0x13 and 0x14 have the same function.

(5) ICache Lines Cleared where Access Address and Tags Match
There is a function that compares addresses held by ICache tags and access addresses and clears them if they match. When ASI=0x18 is accessed and A<19:5> of the access address matches, matching lines of ICache are cleared. This does not depend on the bank. Any of ASI=0x19, 0x1a, 0x1a and 0x1b have the same function.

(6) There is no function which implements the same thing as (5) for DCache.

(7) Flush Instruction
The Flush instruction is for clearing 1valid bit (corresponding to a double word) of ICache, and the address to be cleared is specified by operand rs1+rs2 or rs1+simm13 in the instruction.

[Note] Clears in (1)~(3) must be executed during Cache OFF. (4), (5) and (7) are valid during Cache ON.

[1] Translator's Note: The Japanese text reads "ICache" here. This is probably an error, as the instructions for ICache are contained in the preceding paragraph.

{Note} In Cache ON status there is no function for flushing DCache only. The method in (5) must be used on the assumption that ICache is also inenabled.

4.4. DSU (Debug Support Unit)

4.4.1. Summary

The DSU (Debug Support Unit) is a hardware mechanism for supporting program debugging. The DSU function is valid when the value of BRKEN# sampled during a reset is "L". The DSU causes debug traps (ttt=255) to occur upon request from the hardware or software. Debug traps start execution from address 0x0000FF00 without regard to TBR values. The MB8686 processor DSU has the following features:

(1) Types of Break Function
Instruction address breaks by asynchronous debug trap instruction addresses because of external input (DBREAK#).
Single-step Mode
Software Break Point (TA255)
(2) 16-Column PC Trace buffer available
(3) Programming Model same as SPARClite
Debug Support Register addresses are the same as SPARClite. While not all SPARClite DSR and DCR bits are implemented, the ones that are implemented are the same.
(4) Debug Trap Features
- All debug trap types are are 255, and TBA is ignored.
(Debug traps always start at address 0x0000ff0).
- Debug trap priority is 2 (highest except for reset).
- Debug traps are not masked by PS ET=0.
- Flags (pET, pPS) for exclusive use of new debug routines.

(5) ROM (or Flash ROM) can be selected for debug trap routines by external pins (BRKGO) indicating debug status.
(6) In Normal Mode the Debug Support Unit stops operation to reduce power consumption.

4.4.2. Related Pin Overview

Table 4-5 DSU pin List

Pin Name	I/O	Function
BRKEN#	I	Access Enable Pin. Input "L" when using the Debug Support function. This signal is sampled during reset periods. To enable the built-in CPU Debug Support function, setup time and hold time of at least 2 CLKIN cycles is required for the rise of RESET#.
DBREAK#	I	Debug Break Pin. Inputting "L" to this pin in Debug Mode causes debug traps to occur. This pin is asynchronous input, and "L" width of at least 2 clock periods in CLKIN is necessary. When it is desired to jump to a break processing routine immediately following a reset, setup time and hold time of at least 2 CLKIN cycles is required for the rise of RESET#.
BRKGO	O	Break Go Pin. When this signal is "H", it indicates that a debug break processing routine is running.

4.4.3. Register Overview

Table 4-6 DSU Register

ASI	Address	Register Name	Symbol	Reset Value			
				b31 b24	b23 b16	b15 b8	b7 b0
0x01	0x0000FF00	Instruction Address Descriptor Register1	IADR1	xxxxxxxx	xxxxxxxx	xxxxxxxx	xxxxxx00
0x01	0x0000FF04	Instruction Address Descriptor Register2	IADR2	xxxxxxxx	xxxxxxxx	xxxxxxxx	xxxxxx00
0x01	0x0000FF18	Debug Control Register	DCR	00000000	00000000	00000100	00000000
0x01	0x0000FF1C	Debug Status Register	DSR	00000000	00000000	11110100	000000*0

4.4.4. Register Details

4.4.4.1. Instruction Address Descriptor Register (IADR1,2)

31 2	1 0
Instruction Address Descriptor	Reserved

Symbol :IADR1,2
Address :IADR1 - 0x0000FF00 (ASI=0x1)
IADR2 - 0x0000FF04 (ASI=0x1)
Reset State :0x00000000

bit	Field Name	R/W	Description
31-2	IADR	R/W	- IADR Sets PC values which cause internal hardware breaks to occur. The program PC values being executed and IADR Register values match, and when corresponding DSR.US bit==PSR.S corresponding DCR.EIA == 1 it causes a debug trap.
1-0	Reserved	R	

Figure 4-1 Instruction Address Descriptor Register

4.4.4.2. Debug Control Register (DCR)

31 16	15	14	13	12 11	10	9	8 7	6	5	4	3 0
Reserved	US2	US1	SWF	Reserved	TDBP	BG	Reserved	EIA2	EIA1	SS	Reserved

Symbol :DCR
Address :0x0000FF18 (ASI=0x1)
Reset State

bit	Field Name	R/W	Description
31-16	Reserved		
15	US2		- User/Supervisor bit for instruction address 2
14	US1		- User/Supervisor bit for instruction address 1
13	SWF		- Software Flag
12-11	Reserved		-
10	TDBP		- Trap routine,,Disabled Break Point

9	BG		- Break Go
8-7	Reserved		-
6	EIA2		- Enable Instruction Address 2
5	EIA1		- Enable Instruction Address 1

4	SS		= Single Step Enable
3-0	Reserved		-

Figure 4-2 Debug Control Register

4.4.4.3. Debug Status Register (DSR)

31 16	15 12	11 10	9	8	7	6	5 4	3	2	1	0
Reserved	TBPTR	Reserved	pPS	pET	Rsv	EB	Reserved	IAM2	IAM1	EN_BRK	Rsv

Symbol :DSR
Address :0x0000FF18 (ASI=0x1)
Reset State

bit	Field Name	R/W	Description
31-16	Reserved		-
15-12	TBPTR	R	- Trace Buffer Pointer Holds the pointer to the newest instruction which is being held in the PC Trace buffer in Debug Mode. Holds PC Trace Buffer (Address=0x0000FF40-0x0000FF7C, ASI=0x1) address bit[5:2].
11-10	Reserved	R	-
9	pPS	R	- previous PS(PSR) Preserves the PSR.PS bit when debug traps occur. Written back to the PSR.PS bit during returns from debug traps.
8	pET	R	- previous ET(PSR) Preserves the PSR.ET bit when debug traps occur. Written back to the PSR.ET bit during returns from debug traps.
7	Reserved	R	-
6	EB	R/W	- External Break occurred Indicates that the DBREAK# pin is asserted in Debug Mode. In debug traps, this bit must be cleared. I(Read) : Indicates a hardware break has occurred because of DBREA. O(Write) : Clears EB bit.
5-4	Reserved	R	-
3	IAM1	R/W	- Instruction Address Match 1
2	IAM2	R/W	- Instruction Address Match 2
1	EN_BRK	R	- Enable Breaks Displays BRKEN# value sampled during reset. 0 : ENBRK# asserted (Debug Mode) 1 : ENBRK# not asserted.
0	Reserved	R	-

Figure 4-12 Debug Control Register

4.4.5. Explanation of Functions

4.4.5.1. Moving to Debug Mode

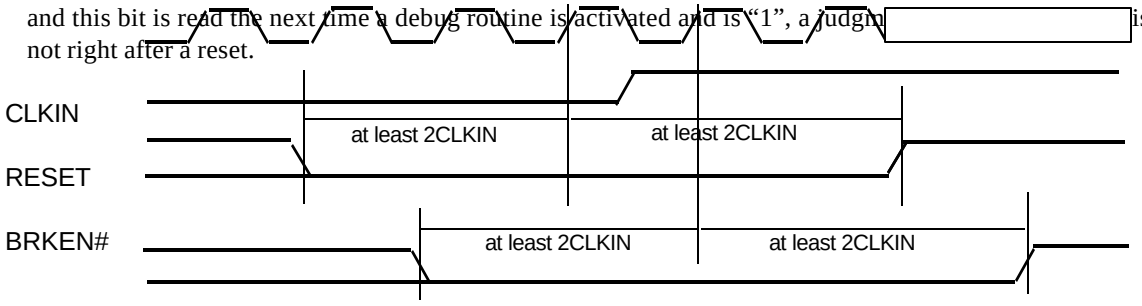
There are 2 CPU modes: Normal Mode and Debug Mode . The EN_BRK bit of the DSR Register indicates the current mode:

EN_BRK = 0 ··· Debug Mode
EN_BRK = 1 ··· Normal Mode

Mode is decided by the status of the BRKEN# pin upon reset signal cancellation. The value at that time is held by the EN_BRK bit and cannot be rewritten by the software until the next reset.

In Normal Mode, all breaks are disabled , and debug traps do not occur. In addition, the DSU stops functioning in order to reduce power consumption. If the DBREAK# pin and the BRKEN# pin are asserted at the same time upon reset, instead of fetching instructions from the CPU0 address, an immediate jump is made to a debug routine (0x00000FF0). By doing this the boot ROM can be replaced by the Debug Monitor ROM.

By using the SWF (in the DCR Register), a judgment can be made as to whether or not that debug routine was activated right after a reset. The SWF pin is "0" right after a reset, and thus if at that time "1" is written, and this bit is read the next time a debug routine is activated and is "1", a judgment is made that this is not right after a reset.



- BRKEN# signals maintain the input level in RESET# cancel timing and decide whether to enable debug support functions or not.
- DBREAK# signals are asynchronous signals. They are not executed from address 0 after RESET# cancel, and the timing shown in the above drawing must be met when it is desired to jump to a break processing routine (address 0x0000.0FF0) right after a RESET# cancel.

Figure 4-13 DSU Enable Signal Control Timing

4.4.5.2. Debug Trap Activation

There are 3 ways to activate debug traps:

Activation by External Breaks

The EB bit of the DSR Register is set by DBREAK# fall input, and an asynchronous debug trap is activated. This input is ignored between Normal Mode and the debug trap routine.

Activation by Internal Hardware Breaks

This break point is handled as a synchronous trap. The following are the 2 conditions for causing this trap:

(1) Instruction Address Breaks

The PC address matches 1 of ADR1/2, the PSR User / Supervisor bits match the US1/2 bits (DSR Register) which correspond to IADR, and the EIA1/2 bits of the DSR registers which correspond to these are set.

Accordingly, in order to activate this break point, the EIA bits of the DCR Register and the US bits of the IADR Register must be set beforehand. Instruction address break points are before-type break points. Thus, debug traps occur before matching address instructions are executed.

(2) Single Step Mode

When the SS bit of the DCR Register is set ("1"), the CPU is in Single Step Mode, and debug traps occur after each instruction without regard to PC and IADR values (there are several exceptions such as RETT, WRPS and the like). The SS bit of the DSR Register must be set if it is desired to select this mode.

Software Breaks

The TA255 Instruction (Trap always with tt=255) is an instruction for software debug traps. This instruction causes debug traps to occur when the EN_BRK bit of the DSR Register is "0".

4.4.5.3. Debug Trap Features

Debug trap priority is 2. This is the highest priority except for resets. All debug traps are trap type tt=255, and the trap address is 0x00000FF0. Trap Base Address (TBA) is ignored. When the TDBP bit of the DCR Register is "1", debug traps occur even during traps (ET=0). When the TDBP bit of the DCR Register is "0", hardware breaks and external breaks are ignored if ET=0. Software breaks occur even if ET=0. Debug traps occur only in Supervisor Mode. Debug traps cannot be nested. Debug traps cannot again be activated within debug traps.

When the BG bit of the DCR Register is "1", break conditions other than TA255 which occur are ignored, and traps do not occur. Software breaks (TA255) in debug traps are incorrect and the results cannot be forecast. TA255 instructions should not be used in debug trap routines. When debug trap routines are entered, condition codes (PSR CC fields) are not automatically saved by the hardware. When debug trap routines are able to rewrite these, saves and restores of these must be performed by the software.

4.4.5.4. Debug Trap Operations

In addition to normal trap operations, the following operations occur in debug traps:

ET → pET ··· ET bit saved to pET
PS → pPS ··· PS bit saved to pPS
1 → DCR.BG ··· BG bit of DCR Register becomes "1"
255 → TBR.tt ··· (TBR tt field is only set to 255 in software debug traps. tt field is not rewritten in hardware breaks and external breaks)

In addition, when a return is made from a debug trap (JMPL+RETT is executed when DCR.BG is "1"), the next operation is performed.

0 → DCR.BG ··· BG bit of DCR Register becomes "0"
pPS → PS ··· pPS bit returned to PS
pET → ET ··· pET bit returned to ET

pPS and pET are new bits for debug traps. These are assigned to bits 9 and 8 of the DSR Register and are Read Only bits.

4.4.5.5. BRKGO Output Pin

The MB8686 has a BRKGO output pin. This always reflects the BG bit of the DCR Register. When this bit is "1", it indicates that instruction fetches and data access are for debug trap routines. This pin can be used for ROM chip select for debug trap routines. Instructions and data indicated by BRKGO are not cached, but if data which has entered a cache in a normal operation (BRKGO="L") is accessed in a debug routine, a cache hit may occur. Therefore, Normal Mode programs and debug routines cannot be placed in the same address.

4.4.5.6. Register Fields

Handling of register fields in debug traps is the same as for normal traps, and the CWP is decremented. Handling of register fields in debug trap routines is left entirely to the software.

4.4.5.7. PC Trace Features

PC trace information is stored in a 16-column cyclic registers which are mapped in ASI=0x01 memory spaces. In Debug Mode, as instructions are fetched their addresses are stored in these registers until a break occurs. If the Trace Rergister becomes full before a break occurs, new addresses overwrite the oldest registers (cyclic operation).

Break trap routines can read those registers which are mapped in ASI=0x01 addresses 0x00000FF40~0x00000ff7C. The TBPTR field of the DSR Register holds the register address where the last PC address is stored. By shifting the TBPTR field to bit5~2 and doing 0x0FF40 and OR, the newest register address can be obtained. For example, if TBPTR is 5 and this is shifted, it becomes 0x14, and if 0x0FF40 and OR are done, 0x0FF54 is obtained. The most recent trace is stored in this register, and below it the new order is Reg4, Reg3, ‥Reg0, Reg15, Reg14 ‥:

4.4.5.8. Debug Support Function Restrictions

In the SPARClite DSU (Debug Support Unit) system, window overflow and underflow trap processing routines cannot be debugged. This is because of the following reasons:

The functions independently expand trap operations naturally specified by the SPARC architecture and implement them as a trap type. However, the method of implementation is of a special nature and differs from ordinary traps as shown below.

Ordinary traps are “A” in a double exception when the ET bit of the PSR (Processor State Register) is “0”, but with “BREAK” traps , either nothing happens (when the Trap_Disabled_break control bit is “0”), or a “BREAK” trap occurs even if ET=0 (When the Trap_Disabled_break control bit is “1”).

There are cases of using the Trap_Disabled_break control bit when it is “1” in which strong demands for debugging of trap processing routines themselves in step execution and the like come out. However, these are special cases, and, because “BREAK” is also implemented as a trap type, during the time the break is in effect the CWP (Current Window Pointer) is decremented and window registers are rotated. Thus, of trap processing routines, window overflow trap and window underflow trap processing routines cannot be debugged in the SPARClite DSU system because the contents of the next register window are rewritten by a break. It should be assumed that window register contents are destroyed during use when the Trap_Disabled_Break function is enabled.

While it is possible that, in addition to a window reserved for normal trap use, another reserved formatted window overflow and underflow routine may be used, one window is consumed.

4.5. Low Power Consumption Mode

4.5.1. Summary

The MB8686 has 2 modes for low power consumption: Sleep Mode and Stop Mode.

Sleep Mode

The PLL keeps on operating, but the clocks which provide peripheral functions stop. Normal operation can continue after Sleep Mode is canceled.

Stop Mode

The PLL stops, as do all internal clocks. A reset must be input in order to restart.

4.5.2. Related Pin Overview

Table 4-7 Sleep Mode Pin List

Pin Name	I/O	Function
PDOWN#	O	Sleep Mode Output Pin. Indicates by outputting "L" level that a move to Sleep Mode (Low Power Consumption Mode) has been completed, When "L" is output to this pin: (MB86860) SPARClite bus is also open. (MB86861) Drives inactive level.
WKUP#	I	Sleep Mode Cancel Pin. When "L" is input to this pin, CPU Sleep Mode is canceled, and operation starts. This pin is asynchronous input, and at least 2 "L" width clock periods are required by CLKIN. "L" should only be input to this pin when PDOWN# is "L".

4.5.3. Register Overview

Table 4-8 Sleep Mode Register

ASI	Address	Register Name	Symbol	Reset Value							
				b31	b24	b23	b16	b15	b8	b7	b0
0x01	0x00002004	Sleep Mode Register	SMR	RRRRRRRR		RRRRRRRR		RRRRRRRR		RRRRRRR0	

4.5.4. Register Details

4.5.4.1. Sleep Mode Register

31	1	0
Reserved		SLP

Symbol :SMR
Address :0x80002004 (ASI=0x4)
Reset State :0x00000000

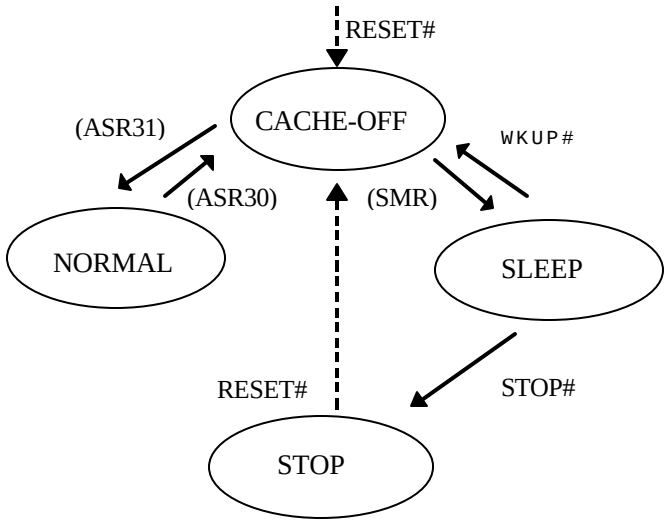
bit	Field Name	R/W	Description
31-1	Reserved		
0	SLP	W	In Cache-Off Mode, if "1" is written to this bit, the processor converts to Sleep Mode.

Figure 4-14 Sleep Mode Register

4.5.5. Explanation of Functions

Conversion to Low Power Consumption Mode must be done in Cache-Off status. In the MB8686 processor, cache-off status is called Cache-Off Mode, and operating status using caches is called Normal Mode. The following diagram shows MB8686 processor status conversions:

Figure 4-15 Power Save State Diagram



(7) CACHE-OFF
Converts to this mode after reset.
In Cache-Off Mode sensamp also stops.
Converts to Normal Mode by setting ICCR Register.
Converts to Sleep Mode by setting Sleep Mode Register.

(8) NORMAL Mode
Conversion is only possible from Cache-Off Mode.
Converts to Cache-Off Mode by setting ICCR Register.

(9) SLEEP Mode
Conversion is only possible from Cache-Off Mode.
Clocks stop except for PLL.
PDOWN is asserted in Sleep Mode periods.
During Sleep Mode conversions, SDRAM-IF issues self-refresh command to SDRAM.
Opens SPARClite bus right.
Returns to Cache-Off Mode if WKUP# pin is asserted.
Converts to Stop Mode if STOP# Pin is asserted.

(10) STOP Mode
Conversion is only possible from Sleep Mode.
PLL stops, and all internal clocks stop.
Reboot possible only by means of reset.

[Note] When converting from Cache-Off Mode to Normal Mode, a cache flush (access to ASI0x31space) is required before setting the ICCR Register.

4.5.6. Items Requiring Attention in Sleep Mode

In the Sleep Function in the MB8686, at the point in time when it goes into Sleep Mode the processor becomes unable to receive BREQ# signals. Moreover, operation in which Sleep Mode entry timing and BREQ# signal assert timing conflict is not guaranteed. Accordingly, BREQ# signals must be negated immediately before a Sleep Mode Set routine.

5. TLB

5.1. Summary

This TLB converts virtual addresses to physical addresses in accordance with the address conversion table. However, no memory protection function causing traps to occur when there is a TLB-miss is supported. When the TLB function is not used, addresses from the CPU are output as they are as physical addresses.

- Number of TLB entries = 16
- Page size of 14KB, 1MB, 4MB and 16MB supported.
- Different page sizes can be set for each entry
- When a miss occurs in address conversion, no trap occurs, and addresses are output as they are from the CPU.

[Note] The TLB function is not available in the MB8686.

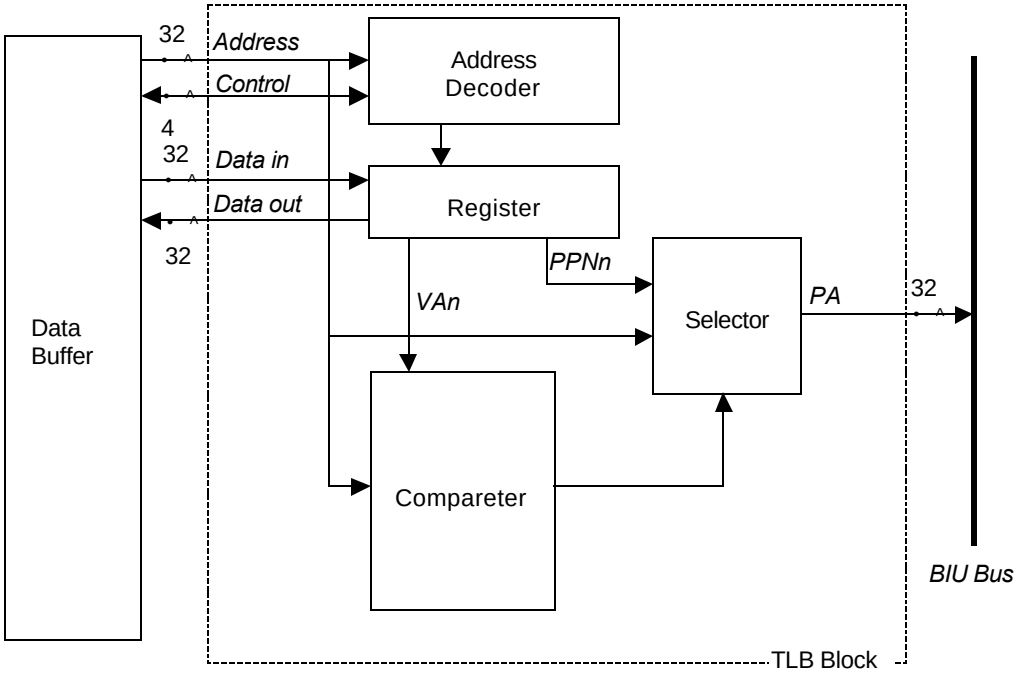


Figure 5-1 TLB Block Diagram

5.6. Register Overview

Table 5-1 TLB Registers

ASI	Address	Register Name	Symbol	Reset Value			
				b31 b24	b23 b16	b15 b8	b7 b0
0x04	0x80000008	TLB Control Register	TLB	xxxxxxxx	xxxxxxxx	xxxxxxxx	xxxxxxxx
0x04	0x80001400	Virtual Word Address Entry Register00	VWE00	xxxxxxxx	xxxxxxxx	xxxxxxxx	xxxxxxxx
0x04	0x80001408	Physical Word Address Entry Register00	PWE00	xxxxxxxx	xxxxxxxx	xxxxxxxx	xxxxxxxx
0x04	0x80001410	Virtual Word Address Entry Register01	VWE01	xxxxxxxx	xxxxxxxx	xxxxxxxx	xxxxxxxx
0x04	0x80001418	Physical Word Address Entry Registe01	PWE01	xxxxxxxx	xxxxxxxx	xxxxxxxx	xxxxxxxx
0x04	0x80001420	Virtual Word Address Entry Register02	VWE02	xxxxxxxx	xxxxxxxx	xxxxxxxx	xxxxxxxx
0x04	0x80001428	Physical Word Address Entry Register02	PWE02	xxxxxxxx	xxxxxxxx	xxxxxxxx	xxxxxxxx
0x04	0x80001430	Virtual Word Address Entry Register03	VWE03	xxxxxxxx	xxxxxxxx	xxxxxxxx	xxxxxxxx
0x04	0x80001438	Physical Word Address Entry Register03	PWE03	xxxxxxxx	xxxxxxxx	xxxxxxxx	xxxxxxxx
0x04	0x80001440	Virtual Word Address Entry Register04	VWE04	xxxxxxxx	xxxxxxxx	xxxxxxxx	xxxxxxxx
0x04	0x80001448	Physical Word Address Entry Register04	PWE04	xxxxxxxx	xxxxxxxx	xxxxxxxx	xxxxxxxx
0x04	0x80001450	Virtual Word Address Entry Register05	VWE05	xxxxxxxx	xxxxxxxx	xxxxxxxx	xxxxxxxx
0x04	0x80001458	Physical Word Address Entry Register05	PWE05	xxxxxxxx	xxxxxxxx	xxxxxxxx	xxxxxxxx
0x04	0x80001460	Virtual Word Address Entry Register06	VWE06	xxxxxxxx	xxxxxxxx	xxxxxxxx	xxxxxxxx
0x04	0x80001468	Physical Word Address Entry Register06	PWE06	xxxxxxxx	xxxxxxxx	xxxxxxxx	xxxxxxxx
0x04	0x80001470	Virtual Word Address Entry Register07	VWE07	xxxxxxxx	xxxxxxxx	xxxxxxxx	xxxxxxxx
0x04	0x80001478	Physical Word Address Entry Register07	PWE07	xxxxxxxx	xxxxxxxx	xxxxxxxx	xxxxxxxx
0x04	0x80001480	Virtual Word Address Entry Register08	VWE08	xxxxxxxx	xxxxxxxx	xxxxxxxx	xxxxxxxx
0x04	0x80001488	Physical Word Address Entry Register08	PWE08	xxxxxxxx	xxxxxxxx	xxxxxxxx	xxxxxxxx
0x04	0x80001490	Virtual Word Address Entry Register09	VWE09	xxxxxxxx	xxxxxxxx	xxxxxxxx	xxxxxxxx
0x04	0x80001498	Physical Word Address Entry Register09	PWE09	xxxxxxxx	xxxxxxxx	xxxxxxxx	xxxxxxxx
0x04	0x800014A0	Virtual Word Address Entry Register10	VWE10	xxxxxxxx	xxxxxxxx	xxxxxxxx	xxxxxxxx
0x04	0x800014A8	Physical Word Address Entry Register10	PWE10	xxxxxxxx	xxxxxxxx	xxxxxxxx	xxxxxxxx
0x04	0x800014B0	Virtual Word Address Entry Register11	VWE11	xxxxxxxx	xxxxxxxx	xxxxxxxx	xxxxxxxx
0x04	0x800014B8	Physical Word Address Entry Register11	PWE11	xxxxxxxx	xxxxxxxx	xxxxxxxx	xxxxxxxx

0x04	0x800014C0	Virtual Word Address Entry Register12	VWE12	xxxxxxxx	xxxxxxxx	xxxxxxxx	xxxxxxxx
0x04	0x800014C8	Physical Word Address Entry Register12	PWE12	xxxxxxxx	xxxxxxxx	xxxxxxxx	xxxxxxxx
0x04	0x800014D0	Virtual Word Address Entry Register13	VWE13	xxxxxxxx	xxxxxxxx	xxxxxxxx	xxxxxxxx
0x04	0x800014D8	Physical Word Address Entry Register13	PWE13	xxxxxxxx	xxxxxxxx	xxxxxxxx	xxxxxxxx
0x04	0x800014E0	Virtual Word Address Entry Register14	VWE14	xxxxxxxx	xxxxxxxx	xxxxxxxx	xxxxxxxx
0x04	0x800014E8	Physical Word Address Entry Register14	PWE14	xxxxxxxx	xxxxxxxx	xxxxxxxx	xxxxxxxx
0x04	0x800014F0	Virtual Word Address Entry Register15	VWE15	xxxxxxxx	xxxxxxxx	xxxxxxxx	xxxxxxxx
0x04	0x800014F8	Physical Word Address Entry Register15	PWE15	xxxxxxxx	xxxxxxxx	xxxxxxxx	xxxxxxxx

5.7. Register Details

5.7.1. TLB Control Register (TCR)

Setting "1" to the TE bit of this register enables the Address Conversion Function. At least one batch of entries (VWE, PWE) must be set. When TE="0", addresses are output from the CPU as they are as physical addresses.

31	1	0
Reserved		TE

Symbol :TCR
Address :0x80000008 (ASI=0x4)
Reset State :undefined

bit	Field Name	R/W	Description
31-1	Reserved		
0	TE	R/W	- TLB enable Sets whether to enable TLB or not 0: TLB invalid 1: TLB valid Reset state: 0

Figure 5-1 Physical Word Address Entry Register

5.7.2. Virtual Word Address Entry Register (VWE0-15)

For setting the upper order 20 bits of the virtual address to be converted. Performs settings in combination with the PWE Register.

31	12	11	0
VA		Reserved	

Symbol :VWE00-15
Address :0x800014000-0x800014F0 (ASI=0x4)
Reset State :undefined

bit	Field Name	R/W	Description
31-12	VA	R/W	- Virtual Address<31:12> Sets upper order 30b of virtual address Must be set in combination with Physical Word Address Entry Register (PWE).
11-0	Reserved		

Figure 5-2 Virtual Word Address Entry Register

5.7.3. Physical Word Address Entry Register (PWE0-15)

Register for setting the upper order 20b of physical addresses corresponding to virtual addresses set to corresponding VWEs. Page sizes can be set from the PS bit.

31 12	11 5	4 10	
PPN	Reserved	PS	V

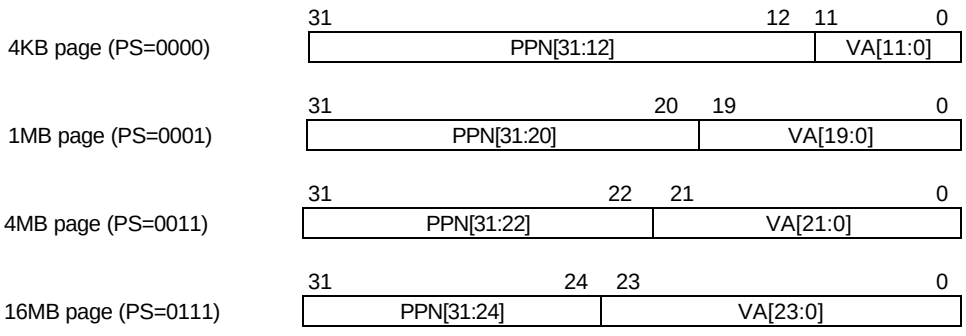
Symbol :PWE00-15
Address :0x800014008-0x800014F8 (ASI=0x4)
Reset State :undefined

bit	Field Name	R/W	Description
31-12	PPN	R/W	- Physical Page Number Sets upper order bits of the physical addresses corresponding to virtual addresses set in the corresponding VWEL registers. Page size is set by the PS bit.
11-5	Reserved		
4-1	PS	R/W	- Page Size Sets physical page size. Page size depends on PS bit settings as shown below: 0000: 4Kbyte 0001: 1Mbyte 0011: 4Mbyte 0111: 16Mbyte 1111: 64MByte
0	V	R/W	- Valid Indicates whether the currrent entry is valid or not. 0: invalid 1: valid

Figure 5-3 Physical Word Address Entry Register

5.8. Operating Functions

In order to use the TLB, the TCR and at least 1 TLB entry (combined VWE and PWE entry) must be set. When the TLB is enabled, addresses handled by the CPU (addresses seen from programs) are virtual addresses, and they are converted to physical addresses by the TLB. Physical addresses are configured from upper order PPN (Physical Page Numbers) and lower order offsets. Virtual addresses are compared with VWE entry VAs, and if they match, the Physical Word PPNs of the corresponding PWE entries become the upper order bits of the physical addresses. At that time how many bits of the PPN will be reflected in physical addresses is decided by the PS bit. Offsets are virtual addresses from the CPU used as they are.



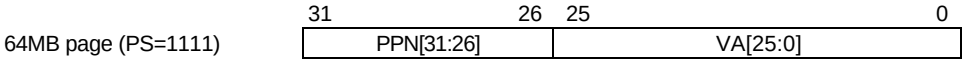


Figure 5-2 Physical Address

When the TE bit of the TCR is set to “0”, no TLB address conversions are performed, and virtual addresses go out as they are to the exterior as physical addresses. Also, if the Physical Word V bit is “0”, even if a VA hits in that entry, the VA becomes a physical address as is without being converted.

6. Data Buffer Unit

6.1. Summary

The Data Buffer Unit consists of an Instuction Buffer (IB), Data Read Buffer (RB) and Write Buffer (WB). It is a module for the efficient performance of instruction and data transfers between the CPU and external memory starting with SDRAM based on burst access. Since the basis of SDRAM access for single access operations which assume MB8686 CPU Core secondary cache connections is burst access, data buffers are divided into cases of cache area and non-cache area access from the CPU Core. For reads from cache areas, burst access to the exterior is requested. Moreover, instructions or data fetched externally in bursts are stored in buffers controlled by address tags and valid bits, hits and misshits in each access from the CPU are detected, and when there is a hit instructions and data inside the buffers are supplied to the CPU Core.

Write buffers are buffers for preventing holds for CPU Core store operations even when external buses are busy. The Data Buffer Unit basically gives priority to instruction fetches and data reads over data stores. However, when attempts are made to read data which has accumulated in write buffers, it exercises control such that once the write buffer contents is written to memory it is read. The write buffer operates basically as FIFO, performing merge processing which arranges 2 store operations into 1 when writes to the same address are consecutive and reduces external bus traffic.

Data buffer operations follow a Total Store Ordering Policy which guarantees that the order of store operations from the CPU Core to memory will not change.

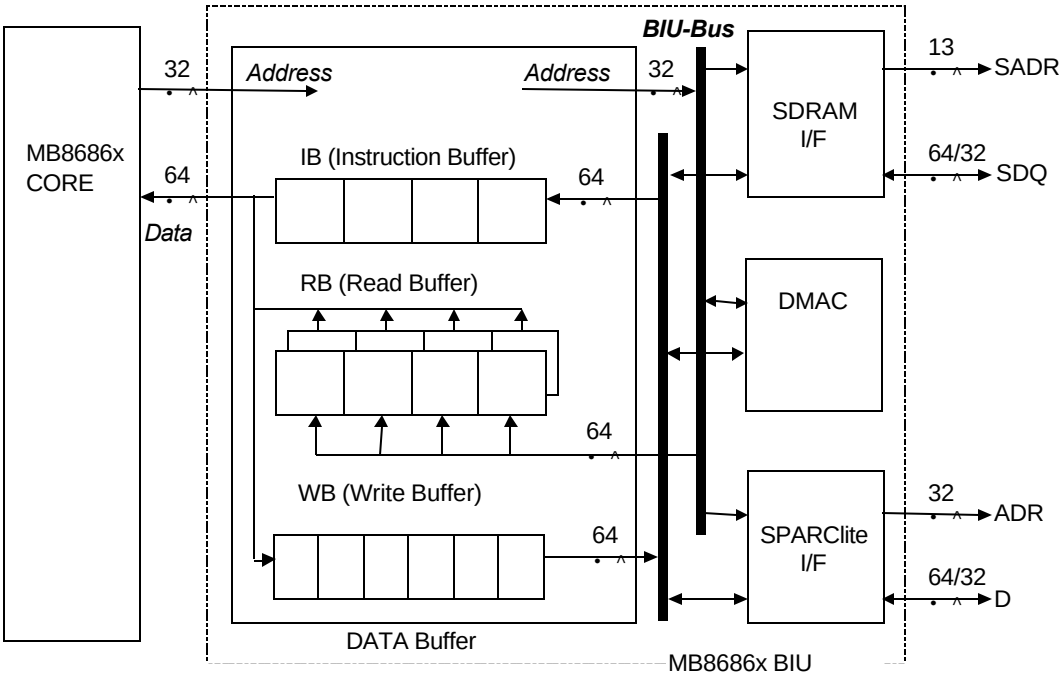


Figure 6-1 Data Buffer Unit

6.2. Register Overview

Table 6-1 Data Buffer Registers

ASI	Address	Register Name	Symbol	Reset Value			
				b31 b24	b23 b16	b15 b8	b7 b0
0x04	0x80000000	Buffer Control Register	BCR	00000000	00000000	00000000	00000000

6.3. Register Details

6.3.1. Buffer Control Register (BCR)

Register which performs all buffer controls and settings. The IB, RB and WB are enabled respectively by the IE, RE and WE bits. The IB and RB can be cleared by the ICL.RCL bit. There is no clear bit in the WB.

31 9	8 6	5	4	3	2	1	0
Reserved	WBDP	MCE	ICL	RCL	IE	RE	WE

Symbol :BCR
Address :0x80000000 (ASI=0x4)
Reset State :0x00000000

bit	Field Name	R/W	Description
31-9	Reserved		
8-6	WBDP	R/W	- Write Buffer DePth Sets number of Write Buffer columns. 000: 16 001: 14 010: 12 011: 10 100: 8 101: 6 110-111: Reserved Reset state: 0
5	MCE	R/W	- Merge & Collapse Enables Merge & Collapse function. 0: Merge & Collapse is disabled 1: Merge & Collapse is enabled Reset state: 0
4	ICL	R/W	- Instruction Buffer Clear Instruction Buffer is cleared by setting “1” to this bit. This bit returns to “0” upon completion of the clear operation. 0: No operation 1: Instruction Buffer Clear Reset state: 0
3	RCL	R/W	- Read Buffer Clear Read Buffer is cleared by setting “1” to this bit. This bit returns to “0” upon completion of the clear operation. 0: No operation 1: Read Buffer Clear 0: Reset state: 0

2	IE	R/W	- Instruction Buffer Enable Instruction Buffer is enabled by setting “1” to this bit. 0: Instruction Buffer Disable 1: Instruction Buffer Enable Reset state: 0

1	RE	R/W	- Read Buffer Enable Read Buffer is enabled by setting “1” to this bit. 0: Read Buffer Disable 1: Read Buffer Enable Reset state: 0
1	WE	R/W	- Write Buffer Enable Write Buffer is enabled by writing “1” to this bit. 0: Write Buffer Disable 1: Write Buffer Enable Reset state: 0

Figure 6-2 Buffer Control Register

6.4. Operating Functions

6.4.1. Instruction Buffer (IB)

Composed of 64-bit width 4-column buffer, buffer data address tags and 4b valid-bit which indicate that data is valid.

- When there is a miss in an Instruction Buffer Tag
  Reads 4 double word data by burst transfer in sequential addresses from requested address. Also outputs double word data from the relevant addresses to the IMB bus.
- When there is a Tag Hit
  Outputs data corresponding to relevant addresses to the IMB bus.
- When there is an Instruction Fetch from a Non-Cache Area
  Bypasses the buffer. Invalidates tag hits. Also, invalidates valid bits when a MEXC signal is asserted.
- CACHE Target Area is ASI=8,9
  Double word loads to this area are regarded as Instruction Fetches.
- Invalidates hits in Atomic-load/store instructions to ASI=8,9. (handles as Non-Cache)
- IB is also flushed when flush instructions are executed.

6.4.2. Read Buffer (RB)

- Has two 64-bit 4-column buffers. Each buffer has a valid bit which indicates that buffer data address tags and data are valid. Has 1 tag and 1 valid bit for double words.
- When the buffer needs to be replaced, it is replaced by an LRU algorithm.
- Outputs data corresponding to the relevant address to the IMB bus when an address hits a tag in a load instruction.
- When an address misses all RB tags in a load instruction, it reads 4 double word data from external memory by sequential address from the requested address and outputs data corresponding to relevant addresses to the IMB bus.
- Bypasses buffers when load instructions from non-cache areas are executed. Invalidates tag hits.
- Atomic-load/store instructions are handled in all areas as non-cache.

6.4.3. Write Buffer (WB)

- Outputs stored data via FIFO. Output to BIU-BUS is single transfer. FIFO is 64b X(changeable from 16column-6 column)
- When store instructions are in bytes, half words and words and the address is the same as the

immediately preceding data (hit in FIFO input column tag), it overwrites the valid locations. (When the merge function and Merge & Collapse are enabled).

- Controls WB data with valid bits in byte units and reflects to the Byte Enable pin during output.
- Handles atomic-load/store instructions as non-cache in all areas.

6.4.4. Buffering Policy

- The IB and RB are accessed in accordance with the order of instruction execution.
- When store data hits in the initial column of the WB, it overwrites that data (when Merge & Collapse is enabled).
- When Atomic-load/store is executed, contents of the WB are first written to memory, and then the data is read from memory to the RB. If there is a hit in the RB it is invalidated. Stores are handled as non-cache, and then they are executed. At that time, data is handled as non-cache. The order of data access between read and write buffers is as shown below.

Figure 6-3 Access Order between RB and WB

Access Type		bit Judgment	Operation	Burst Access
Cache Area Data Access	Store ASI==a, b ARSR.N	RB: miss & WB: miss	Writes data to WB	OFF
		RB: hit & WB: miss	Writes data to WB	OFF
		RB: miss & WB: hit	Merges with that data when there is a hit in initial WB data. In other cases writes data to WB	OFF
		RB: hit & WB: hit	Merges with that data when there is a hit in initial WB data. In other cases writes data to WB	OFF
	Load ASI==a, b & NC==0	RB: miss & WB: miss	Reads 4 double words from external memory and writes them to RB. Sends 1 double word to the Core at the same time. Updates or replaces RB tags.	ON
		RB: miss & WB: hit	If there are multiple or single hits in the WB, writes the hit data to external memory and then reads 4 double words from external memory to the RB. At the same time, sends 1 double word to the CPU Core. Updates or replaces RB tags.	ON
		RB: hit & WB: miss	Sends 1 double word from RB to CPU Core.	NA
		RB: hit & WB: hit	If there are multiple or single hits in the WB, writes the hit data to external memory (while updating the RB). After that sends 1 double word to the CPU Core.	NA
Cache Area Instruction Fetch	Load/Fetch ASI==8, 9 & NC==0	IB: miss	Reads 4 double words from external memory and writes them to the IB, At the same time, sends 1 double word to the Core. Replaces IB tags.	ON
		IB: hit	Sends 1 double word to the CPU Core.	NA

Figure 6-4 Access Order between RB and WB (continued)

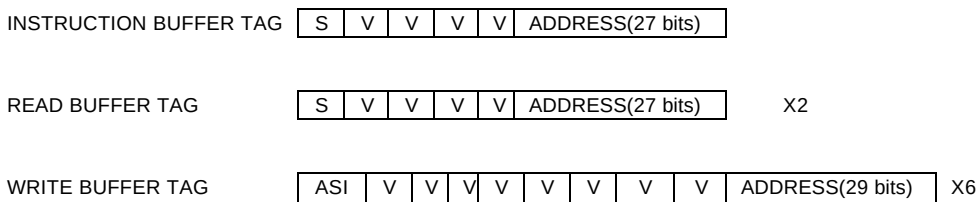
Access Type		bit Judgment	Operation	Burst Access
Cache Area Data Access	Store ASI==8, 9	IB: hit	First writes WB, then writes data to external memory.	OFF
		IB: miss		
	Store ASI==a, b & NC==1	RB: hit	First writes WB, then writes data to external memory. Invalidates tags when there is a hit in the RB.	OFF
		RB: miss		
	Store ASI!= 8,9,a, b	DON'T CARE	First writes WB, then writes data to external memory.	OFF
	Load ASI==a, b & NC==1	RB: hit	After first writing WB to external memory, reads data from external memory or I/O and sends it to the CPU Core. Basically, when RB tags are not updated and there is a hit in the RB, it is invalidated. Data is not written to the RB. Data size is decided by IMSIZE<1:0>.	OFF
		RB: miss		
	Load ASI!= 8.9.a, b	RB: hit	After first writing WB to external memory, reads data from external memory or I/O and sends it to the CPU Core. Data is not written to the RB. Data size is decided by IMSIZE<1:0>.	OFF
		RB: miss		
Cache Area Instruction Fetch	Load/Fetch ASI==8, 9 & NC==1	IB: miss	After first writing WB to external memory, reads data from external memory or I/O and sends it to the CPU Core. Data size is decided by IMSIZE<1:0>.	OFF
		IB: hit		

Figure 6-5 Access Order between RB and WB (continued)

Access Type		bit Judgment	Operation
Atomic Load/store	ASI==8, 9	IB: hit	After first writing WB to external memory, reads data from external memory or I/O and sends it to the CPU Core. Continues to store data to external memory or I/O.
		IB: miss	
	ASI== , b	RB: hit	After first writing WB to external memory, reads data from external memory or I/O and sends it to the CPU Core. Continues to store data to external memory or I/O. Invalidates the RB if there is a hit in the RB.
		RB: miss	
	ASI != 8,9,a b	DON'T CARE	After first writing WB to external memory, reads data from external memory or I/O and sends it to the CPU Core. Continues to store data to external memory or I/O.

6.4.5. Tag Format

Tag format for Instruction Buffers, Read Buffers and Write Buffers is shown below.



S: Supervisor bit V: Valid bit

Figure 6-6 Buffers Tag Format

7. SPARClite Bus Interface

7.1. Summary

The SPARClite Bus is the bus for connecting memory and I/O devices, and it operates synchronous to the bus clock (CLKIN). The SPARClite Bus Interface controls data transfers between the SPARClite Bus and the BIU Bus. The SPARClite Bus Interface supports transparent access mode from external bus masters to SDRAM.

7.2. Register Overview

ASI	Address	Register Name	Symbol	Reset Value			
				b31 b24	b23 b16	b15 b8	b7 b0
0x04	0x80000100	Address Range Specifier Register0	ARSR0	10000000	00001000	00000000	00000000
0x04	0x80000108	Address Range Specifier Register1	ARSR1	xxxxxxxx	xxxxxxxx	xxxxxxxx	xxxxxxxx
0x04	0x80000110	Address Range Specifier Register2	ARSR2	xxxxxxxx	xxxxxxxx	xxxxxxxx	xxxxxxxx
0x04	0x80000118	Address Range Specifier Register3	ARSR3	xxxxxxxx	xxxxxxxx	xxxxxxxx	xxxxxxxx
0x04	0x80000120	Address Range Specifier Register4	ARSR4	xxxxxxxx	xxxxxxxx	xxxxxxxx	xxxxxxxx
0x04	0x80000128	Address Range Specifier Register5	ARSR5	xxxxxxxx	xxxxxxxx	xxxxxxxx	xxxxxxxx
0x04	0x80000200	Address Mask Register0	AMR0	00000000	00000011	00000000	00000001
0x04	0x80000108	Address Mask Register1	AMR1	xxxxxxxx	xxxxxxxx	xxxxxxxx	xxxxxxxx
0x04	0x80000210	Address Mask Register2	AMR2	xxxxxxxx	xxxxxxxx	xxxxxxxx	xxxxxxxx
0x04	0x80000218	Address Mask Register3	AMR3	xxxxxxxx	xxxxxxxx	xxxxxxxx	xxxxxxxx
0x04	0x80000220	Address Mask Register4	AMR4	xxxxxxxx	xxxxxxxx	xxxxxxxx	xxxxxxxx
0x04	0x80000228	Address Mask Register5	AMR5	xxxxxxxx	xxxxxxxx	xxxxxxxx	xxxxxxxx
0x04	0x80000400	Wait State Specifier Register0	WSSR0	00000000	00000000	01111111	11110100
0x04	0x80000408	Wait State Specifier Register1	WSSR1	00000000	00000000	00000000	00000000
0x04	0x80000410	Wait State Specifier Register2	WSSR2	00000000	00000000	00000000	00000000
0x04	0x80000418	Wait State Specifier Register3	WSSR3	00000000	00000000	00000000	00000000
0x04	0x80000420	Wait State Specifier Register4	WSSR4	00000000	00000000	00000000	00000000
0x04	0x80000428	Wait State Specifier Register5	WSSR5	00000000	00000000	00000000	00000000
0x04	0x80000430	MEXC Parity Error Flag Register	MXPEF	00000000	00000000	00000000	00000000
0x04	0x80000430	MEXC Parity Error Control Register	MXPECR	00000000	00000000	00000000	00000000
0x04	0x80000438	Idle Cycle Control Register	IDCCR	00000000	00000000	00000000	00000000

7.3. Register Details

7.3.1. Address Range Specifier Registers (ARSR0 5)

These are the registers for setting SPARClite Bus CS0#~CS5# address ranges. The start address of an address range is set to these registers. Bits which do not perform address comparisons are set in the AMR Registers. If Nbit is 1, its range becomes a non-cache area. ARS0–5 make SPARClite Bus area settings. BW decides the bus width of SPARClite areas specified by CS#1~CS#5. CS0# is exclusively for ROM areas. The CS0# start address is 0. ARSR and CS# correspond as shown below. CS areas must be set so as not to overlap. CS1#–CS5# are not asserted unless both ARSR and AMR are set (CS# is not asserted if only ARSR or AMR is set).

31	30 29	28 22	21 16	15 0
N	BW	Reserved	ASI<5:0>	Address<31:16>

Symbol :ARSR0–5
Address :0x80000100-0x80000128 (ASI=0x4)
Reset State :undefined (ARSR0=0x80080000)

ARSR0 (Address 0x80000100, ASI=0x4) • • • • for setting SPARC area CS0# (0x80080000 on reset)
ARSR1 (Address 0x80000108, ASI=0x4) • • • • for setting SPARC area CS1# (undefined on reset)
ARSR2 (Address 0x80000110, ASI=0x4) • • • • for setting SPARC area CS2# (undefined on reset)
ARSR3 (Address 0x80000118, ASI=0x4) • • • • for setting SPARC area CS3# (undefined on reset)
ARSR4 (Address 0x80000120, ASI=0x4) • • • • for setting SPARC area CS4# (undefined on reset)
ARSR5 (Address 0x80000128, ASI=0x4) • • • • for setting SPARC area CS5# (undefined on reset)

bit	Field Name	R/W	Description
31	N	R/W	-Non-cache Sets cacheable/non-cacheable for corresponding chip select areas. 1: Sets corresponding chip select areas to non-cache areas. 0: Sets corresponding chip select areas to cache areas.
30-29	BW	R/W	- Bus width Sets bus width of corresponding chip select areas. 00: 64b bus width 01: 32b bus width 10: 16b bus width 11: 8b bus width
28-22	Reserved		
21-16	ASI<5:0>	R/W	Sets ASI<5:0> of corresponding chip select areas.
15-0	Address <31:16>	R/W	Sets start addresses of corresponding chip select areas. Sets upper order 16 bit (Address<31:16> of addresses.

Figure 7-1 Address Range Specifier Register

7.3.2. Address Mask Registers (AMR0 5)

Registers for setting CS0#~CS5# address ranges. They set address ranges in combination with the ARSR registers. Bits which set 1 to AMR registers do not compare addresses with the ARSR registers. Only bits set to 0 in the AMR registers compare memory addresses with ARSR registers and assert CS# if they match.

31 22	21 16	15 0
Reserved	ASI mask<5:0>	Address mask<31:16>

Symbol :AMR0–5
Address :0x80000200-0x80000228 (ASI=0x4)
Reset State :undefined (AMR0=0x00030001)

AMR0 (Address 0x80000200, ASI=0x4) • • • • for setting SPARC area CS0# (exclusively for ROM areas 0x00030001 on reset)
AMR1 (Address 0x80000208, ASI=0x4) • • • • for setting SPARC area CS1# (undefined on reset)
AMR2 (Address 0x80000210, ASI=0x4) • • • • for setting SPARC area CS2# (undefined on reset)
AMR3 (Address 0x80000218, ASI=0x4) • • • • for setting SPARC area CS3# (undefined on reset)
AMR4 (Address 0x80000220, ASI=0x4) • • • • for setting SPARC area CS4# (undefined on reset)
AMR5 (Address 0x80000228, ASI=0x4) • • • • for setting SPARC area CS5# (undefined on reset)

Figure 7-3 Address Mask Register

bit	Field Name	R/W	Description
31-22	Reserved		
21-16	ASI mask<5:0>	R/W	Sets ASI masks of corresponding chip select areas.
15-0	Address mask <31:16>	R/W	Sets Address masks of corresponding chip select areas.

7.3.3. Wait State Specifier Registers (WSSR0-5)

31 15	14 10	9 5	4	3	2	1	0
Reserved	CN1	CN2	WE	Reserved	OV	SCB	PE

Symbol :WSSR0–5
Address :0x80000400-0x80000428 (ASI=0x4)
Reset State :undefined (WSSR0=0x00007ff4)

WSSR0 (Address 0x80000400, ASI=0x4) • • • • CS0# Reset State: 0x00007ff4
WSSR1 (Address 0x80000408, ASI=0x4) • • • • CS1# Reset State: 0x00000000
WSSR2 (Address 0x80000410, ASI=0x4) • • • • CS2# Reset State: 0x00000000
WSSR3 (Address 0x80000418, ASI=0x4) • • • • CS3# Reset State: 0x00000000
WSSR4 (Address 0x80000420, ASI=0x4) • • • • CS4# Reset State: 0x00000000
WSSR5 (Address 0x80000428, ASI=0x4) • • • • CS5# Reset State: 0x00000000
Reset State: 0x00000000

Figure 7-4 Wait State Specifier Register

bit	Field Name	R/W	Description
31-15	Reserved		
14-10	CN1	R/W	- count1 Indicates number of waits for the first cycle of single transfers and burst transfers. The specified value + 1 is the number of waits. These bits are valid when [WE] is “1”.

9-5	CN2	R/W	- count2 Specifies the number of waits for the 2nd cycle on of burst transfers. The specified value of this field + 1 is the number of waits. Thus, if 0 is specified, there is 1 wait. These bits are valid when [WE] is “1”.

4	WE	R/W	- wait enable Enable generation of internal waits for corresponding CS areas. [OVR], [CN1], [CN2] and [SCB] are all enabled when [WE] is “1”.
3	Reserved		
2	OVR		-override When this bit is “1”, external READY# is also received. (Gives priority to READY# which comes first). When it is “0”, external READY# is masked and is not accepted. However, the mask function does not work for READY# which is input in the same cycle as AS#, i.e. READY# input by 0wait. This bit is valid when [WE] is “1”. Therefore, this bit should be set with [WE]. (Note): When this bit is “0”, READY# input externally from cw is always valid.
1	SCB		- single cycle burst mode When this bit is set to “1”, burst transfers from the 2nd cycle on are performed with 0wait. This bit is valid when [WE] is “1”. Therefore, this bit should be set with [WE].
0	PE		- parity enable Sets Parity Funcrion enable/disable for access to corresponding CS# areas.

7.3.4. MEXC Parity Error Control Register (MXPEF)

When memory exceptions or parity errors occur in data reads or writes, flags corresponding to this register are set to “1”. To clear all flags, write “0” to the corresponding bits.

31	7	6	5	4	3	2	1	0
Reserved		DMXRE	DMXWE	DPRE	Reserved	MXRE	MXWE	PRE

Symbol :MXPER
Address :0x80000430 (ASI=0x4)
Reset State :0x00000000

bit	Field Name	R/W	Description
31-7	Reserved		
6	DMXRE	R/W	- DMA MEXC Read Error
5	DMXWE	R/W	- DMA MEXC Write Error
4	DPRE	R/W	- DMA Parity Read Error
3	Reserved	R	
2	MXRE	R/W	- CPU MEXC Read Error
1	MXWE	R/W	- CPU MEXC Write Error
0	PRE	R/W	- CPU Parity Read Error

Figure 3-8 MXPEF Register

7.3.5. MEXC Parity Error Control Register

This register is for performing parity check odd/even settings and mode enable settings which return MEXC during the next read from the CPU when memory exception flags are set during data writes.

31	1	0
Reserved	PAR	MXWEE

Symbol :MXPECR
Address :0x80000438 (ASI=0x4)
Reset State :0x00000000

bit	Field Name	R/W	Description
31-2	Reserved		
1	PAR	R/W	- Parity 0: Even parity 1: Odd parity
0	MXWEE	R/W	- MEXC Write Drive Enable 1: Write Error enable 2: Write Error disable

Figure 3-9 MXPECR Register

7.3.6. Idle Cycle Control Register

31	4	3	0
Reserved		IDCC	IDCE

Symbol ICCR
Address :0x80000440 (ASI=0x4)
Reset State :0x00000000

Register for setting automatic insertion of idle cycles for CS0# area access.

bit	Field Name	R/W	Description
31-4	Reserved		-
3-1	IDCC	R/W	- Idle Cycle Count Sets number of idle cycles.
1	IDCE	R/W	- Idle Cycle Enable Idle cycle function is enabled by setting “1” to this bit. 0: idle cycle disable 1: idle cycle enable

Figure 3-10 ICCR Register

7.4. Operating Functions

7.4.1. Operations when the CPU has the Bus Right

The MB8686 has a 64-bit data bus, and bus width and cache/non-cache can be set CS (Chip Select) area by area. SPARClite bus cycles depend on read/write, cache/non-cache and bus width, and operations change in accordance with these. Below are shown variations in CPU access to SPARClite buses.

Table 7-2 SPARClite Interface Operating Overview

operation	bus master	cache/ non-cache	data type	case		access times (burst/non-burst)			
						SPARClite bus width			
						64-bit	32-bit	16-bit	8-bit
read	data buffer	cache	byte half word word double word	BMREQ# assert	BMACK# reply	4(burst)	8(burst)	16(burst)	32(burst)
					BMACK# not reply	4(single)	8(single)	16(single)	32(single)
		non-cache	byte	BMREQ# not assert		1(single)	1(single)	1(single)	1(single)
			half word			1(single)	1(single)	1(single)	2(single)
			word			1(single)	1(single)	2(single)	4(single)
			double word			1(single)	2(single)	4(single)	8(single)
	dmac	non-cache	double word	BMREQ# assert	BMACK# reply	4(burst)	8(burst)	16(burst)	32(burst)
					BMACK# not reply	4(single)	8(single)	16(single)	32(single)
write	data buffer	cache	double word *1	BMREQ# not assert		1(single)	2(single)	4(single	8(single)
		non-cache	byte	BMREQ# not assert		1(single)	1(single)	1(single)	1(single)
			half word			1(single)	1(single)	1(single)	2(single)
			word			1(single)	1(single)	2(single)	4(single)
			double word			1(single)	2(single)	4(single)	8(single)
	dmac	non-cache	double word	BMREQ# assert	BMACK# reply	4(burst)	8(burst	(16(burst)	32(burst)
					BMACK# not reply	4(single)	8(single)	16(single)	32(single)

*1 Data merge and collapse happens. Therefore the data type cannot be decided.

7.4.1.1. Read Operations (Burst Mode/Non-Burst Mode)

Reads from cache areas and read operations by the DMAC are basically processed as burst transfers. Accordingly, BMREQ# is asserted at the same time as AS#, and if BMACK# is returned in READY# timing for the first access, a burst transfer corresponding to the bus width is performed; and if BMACK# is not returned single (non-burst) transfers are performed a number of times corresponding to the bus width.

AS# is only asserted once in Burst Mode, but addresses change sequentially every time READY# is returned, and address pins indicate addresses of access destinations. The MB8686 caches perform data valid/invalid control in double word units. Thus data reads from cache areas are in double word units regardless of the data type requested by the CPU. In other words, DTYP<1:0> is (1,1) and BEx are all asserted without regard to whether the CPU load instruction is Byte, Half Word, Word or Double Word. In 64-bit bus, burst length is 4. If BMREQ# is output, and BMACK# is returned together with READY#, Burst Mode takes effect, and burst transfers of the lengths shown in the above table are performed. If BMACK# is not returned. burst transfers do not occur, and single transfers are performed the required number of times shown in the above table.

In 32-bit bus, burst length is 8, in 16-bit bus, burst length is 16 and in 8-bit bus it is 32. When burst transfers do not take place, the required number of single transfers is performed as shown in the above table.

Reads from non-cache areas are handled as single transfers, and only BEx corresponding to the data requested by a CPU instruction are asserted. In 64-bit bus width the operation definitely ends in one bus cycle. In 32, 16 and 8-bit bus widths, the decision as to how many times single transfers are repeated depends on bus width and data type. Whether a cache or non-cache area is decided by internal chip ASI<5:0> and the N bit of the ARSR (Address Range Specifier Register). Anything but ASI=0x08,09,0a,0b in the SPARC architecture is other than a cache object. Also, in the ARSR, areas which specify N (non-cache)=1 are also handled as non-cache.

7.4.1.2. Write Operations

- Write Operations to Cache Areas
  Since caches built in to the MB8686 are write through type, even if an operation is a write to a cache area it is processed as a single transfer. Hence BMREQ# is not asserted. However, in cache area write operations it decides whether write buffers are performing sequential writes to the same addresses, and if they are the same addresses data types are handled as double words, since it has a function which merges multiple stores into 1. Thus in cache area writes, when it comes to type DTYP<1:0> and BEx values, no fixed relationship is formed.

- Non-cache Area Writes
  Non-cache area writes are handled as single transfers, and the required number of single transfers is performed in proportion to the bus width. During non-cache area writes, only BEx which correspond to data types specified by CPU instructions are asserted. Make sure to map I/O devices in non-cache areas.

- DMAC Write Operations
  DMAC write operations are performed in burst transfers of 4 X double words (When BMREQ# is asserted, and BMACK# replies). When BMACK# does not reply, double word single transfers are performed 4 times.

7.4.1.3. Atomic Load/Store Operations

When the CPU executes atomic load/store instructions, unsplit load/stores appear in consecutive bus cycles on the SPARClite bus. At that time, while the load/stores are being executed LOCK# signals are asserted on the SPARClite bus, and even if there is a bus open request by BREQ#, BGRNT# is not returned until the stores are completed. Whatever area may be specified by an atomic load/store instruction will be handled as non-cache, and transfers will be single transfers together with the load/store instructions. (If a data cache is hit during a load of an atomic load/store, that entry is invalidated).

7.4.1.4. SPARClite Bus Width

SPARClite bus width is set CS# by CS#.

- CS0# Area Bus Width
  CS#0 are assigned to an area starting at a 0 address, and for normal boot purposes the ROM area is made equivalent to this. CS0# area bus width is set by the BMODE16# and BMODE32# pins. BMODE16# and BMODE32# values are sampled during reset periods.

- CS1#~CS5# Bus Width
  CS1#~CS5# bus widths are set by the BW bits (bits 30, 29) of the ARSR Register.

- Bus Widths of Areas belonging to no CS
  The default bus width of areas belonging to no CS is handled as 64-bit.

Table 7-3 CS0# Area Bus

BMODE16#	BMODE32#	Bus Width

Table 7-4 CS1#~CS5# Area Bus

ARSR:bit 30	ARSR:bit 29	Bus Width

0	0	8bit
0	1	16bit
1	0	32bit
1	1	64bit

0	0	64bit
0	1	32bit
1	0	16bit
1	1	8bit

7.4.1.5. Address Cycles

In read cycle and write cycle changeovers, an address cycle of at least one clock cycle is inserted. In sequential reads and writes, there are times when an address cycle is not inserted. When the CPU accesses SDRAM and when read access makes a cache hit, CPU access to the SPARClite bus does not take place, and there is no upper limit regulation of idle cycles.

A function for inserting any address cycle after a read from the CS0# area is offered by the IDCCR (Idle Cycle Control Register) for when data bus off time long EPROMs and the like are directly connected to the SPARClite bus. The function is validated by setting bit0 of the IDSSR to "1". When this function is valid, the space (idle cycle) from READY# for a read operation to the CS0# area to AS# of the next access (without dependence on CSO# and read, write) is guaranteed the number of external clocks set to bit3~bit1 of the IDCCR.

Table 7-5 Relation between IDCCR Set Values and Address Cycles

IDCCR<3:1>	No. of Idle Cycles
3'b000	1
3'b001	2
3'b010	3
3'b011	4
3'b100	5
3'b101	6
3'b110	7
3'b111	8

7.4.1.6. Internal READY# Generator Function

A function is available which, by making settings to registers, generates READY# internally in timing which differs from CS# to CS#. For details, see the explanation of the WSS (Wait State Specifier Register).

7.4.1.7. Bus Right Open Timing (BREQ#, BGRNT#, PBREQ#)

The MB8686 opens buses in the following timing when there is an external bus right request (BREQ# is asserted):

- Reads (burst) from Cache Areas
  Opens bus after burst transfers end.

- Reads from Cache Areas (non-burst: BMACK# not returned)
  Opens bus after required number of single transfers.

- Non-cache Area Reads/Writes
  Opens bus after required number of single transfers.

- Atomic Load/Store
  The CPU does not open external buses between the Load and Store operations of Atomic Load/Store instructions. The CPU opens buses after the end of Store operations. Also, LOCK# signals are asserted during these periods.

When the CPU requires SPARClite bus access by cache misshit and the like after a bus is opened, it asserts PBREQ# and issues a request to the external bus master to withdraw the bus right request (BREQ#). In this regard, there are no time constraints regarding within how many clocks BREQ# must be withdrawn. Once the bus right is opened, the CPU waits continuously until BREQ# is negated.

7.4.1.8. Memory Exceptions (MEXC#)

The MB8686 has a MEXC# input pin for notifying the CPU of an abnormality when for whatever reason an exception occurs externally during access (for example, an area with no resources is accessed or there is absolutely no reply and a time out occurs). If the CPU receives MEXC# together with READY# and that is an instruction fetch, it generates an instruction access exception (Trap Type=1). If it is a data read, it generates a data access exception (Trap Type=9).

However, the method of communicating a MEXC# on the SPARClite bus to the CPU Core differs according to the access type as shown below:

- Burst Read by Cache Misshit MEXC#
  If a MEXC# occurs during a burst transfer, the bus cycle continues without pause until the transfer is completed. When MEXC# is the first double word data of a burst transfer caused by a cache misshit (when it occurs with regard to the first double word access), MEXC# is also transmitted to the CPU Core, and "1" is written to bit2 of the MXPEF (MEXC Parity Error Flag) Register. After the burst transfer ends, the CPU generates an instruction access exception trap or a data access exception trap. MEXC occurring with regard to the 2nd, 3rd and 4th double word data are not directly transmitted to the CPU, but are only invalidated and no traps are generated when they are stored in instruction buffers or data buffers. Data in which errors occur does not enter caches or instruction and data buffers. Hence for access to such data external access is again performed.

- Non-cache Area Read MEXC#
  Non-cache area reads are single transfers, and if a MEXC# occurs at that time it is transmitted as is to the CPU Core. However, when the data type requested by the CPU is larger than the bus width, multiple bus cycles occur. For example, if word data reads are performed to an 8-bit bus width, access takes place 4 times, and if together with read Y# for at least one access from the 1st to the 4th access MEXC# is returned, the fact that a MEXC# occurred for that word data is transmitted to the CPU, and a trap occurs.

- DMAC Read MEXC#
  If a MEXC# occurs right in the middle of a data read from a SPARClite bus, "1" is set to bit5 of the MXPEF as a flag. Also, a MEXC# signal is transmitted to the DMAC, and the DMAC sets "1" to the ERR bit (bit7) of the DMCR (DMA Control Register) and stops the transfer.

- Write MEXC#
  If a MEXC# occurs during a write, "1" is set to bit1 of the MXPEF Register for a write from the CPU and to bit5 of the MXPEF Register for a write from the DMAC. When a MEXC# occurs during a write it is not transmitted to the CPU at the point in time at which it occurs. When 1 is set to bit0 of the MXPECR Register, during the next read access from the CPU it is transmitted to the CPU as a MEXC and a trap occurs. When 0 (0 after a Reset cancel) MEXC is not transmitted to the CPU, and no trap occurs. Thus, in this case the external circuits must notify the CPU of an abnormality by means of interrupts. By using a trap routine to check the MXPEF Register, users can confirm that a MEXC# occurred for the write operation before a read operation.

7.4.1.9. Parity Generate/Check Function

A Parity Check regarding access to corresponding CS# areas is performed by setting 1 to bit0 of the WSSR (Wait State Specifier Register). The respective modes take effect in accordance with the value of bit1 of the MXPECR (MEXC and Parity Error Control Register)- 0: Even Parity, 1: Odd Parity. Correspondence between data buses and DP signals used in each bus width are shown below.

Table 7-6 Bus Width and DP Pins used

SP Bus Width	DATA							
	<63:56>	<55:48>	<47:40>	<39:32>	<31:24>	<23:16>	<15:8>	<7:0>
64bit	DP0	DP1	DP2	DP3	DP4	DP5	DP6	DP7
32bit	—	—	—	—	DP4	DP5	DP6	DP7
16bit	—	—	—	—	—	—	DP6	DP7
8bit	—	—	—	—	—	—	—	DP7

- Reads
  If 1 is set to bit0 of the WSS, a Parity Check is performed. If a Parity Error then occurs, 1 is set to bit0 of the MXPEF as a flag, and this is transmitted to the CPU as a MEXC. By checking bit0 of the MXPEF with a trap routine, the user can confirm that a Parity Error occurred in a read.

- Writes
  When 1 is set to bit0 of the WSS, Parity information is output to the DP Pin at the same time as the corresponding data. The Parity Generator Function is always valid.

7.4.1.10. Bus Width and BE0#-BE7# in the SPARClite Bus

dData location in all Bus Width modes and the corresponding BE# signals in the SPARClite Bus are shown below.

Table 7-7 Bus Width and BE0#-7#

SP Bus Width	DATA							
	<63:56>	<55:48>	<47:40>	<39:32>	<31:24>	<23:16>	<15:8>	<7:0>
64bit	BE0#	BE1#	BE2#	BE3#	BE4#	BE5#	BE6#	BE7#
32bit	—	—	—	—	BE4#	BE5#	BE6#	BE7#
16bit	—	—	—	—	—	—	BE6#	BE7#
8bit	—	—	—	—	—	—	—	BE7#

- In reads from cache areas and DMAC read operations BE0#-7# are all asserted.
- In read operations by the CPU Core from non-cache areas, only BE for valid bytes are asserted.
- In writes, only BE for valid bytes are asserted. However, in stores to cache areas, the data type is handled as double word.
- In both Big-Endian and Little-Endian operations, the above correspondences are the same.

7.4.1.11 16-bit and 8-bit Bus Width ADR<1:0>

In 16-bit and 8-bit bus widths, ADR<1:0> is output to BE4# and BE5#.

Table 7-8 BE4#-5# Pins and ADR<1:0> Correspondence

SP-bus width	BE4#	BE5#
64bit	BE4#	BE5#
32bit	BE4#	BE5#
16bit	ADR<1>	ADR<0>
8bit	ADR<1>	ADR<0>

7.4.1.12. SPARClite Bus Timing

SPARClite Bus Cycle classifications when the CPU has the bus right are as shown below.

Table 7-9 SPARClite Bus Cycle Classifications

Bus Cycle	Bus Width	Data Type	Normal Operation	MEXC# Ending	Reply to BREQ#
Burst Read:	64bit bus		Fig. 2	--	--
	32bit bus		--	--	--
	16bit bus		--	--	--
	8b bus		--	--	--
Single Read 1:	64bit bus		--	--	--
(Single when BMREQ# is	32bit bus		--	--	--
asserted in Cache Area but	16bit bus		--	--	--
BMACK# does not return)	8b bus		--	--	--
Single Read 2:	64bit bus	byte/hw/word/dw	Fig. 1	--	--
Non-cache Area	32bit bus	byte/hw/word/dw	--	--	--
Big-Endian	16bit bus	byte/hw/word/dw	--	--	--
	8b bus	byte/hw/word/dw	--	--	--
Single Read 3:	64bit bus	byte/hw/word/dw	--	--	--
Non-cache Area	32bit bus	byte/hw/word/dw	--	--	--
Little-Endian	16bit bus	byte/hw/word/dw	--	--	--
	8b bus	byte/hw/word/dw	--	--	--
Burst Write:	64bit bus		Fig. 3	--	--
Write by DMAC	32bit bus		--	--	--
	16bit bus		--	--	--
	8b bus		--	--	--
Single Write 1:	64bit bus	byte/hw/word/dw	--	--	--
BMREQ# is asserted in	32bit bus	byte/hw/word/dw	--	--	--
write by DMAC but	16bit bus	byte/hw/word/dw	--	--	--
BMACK# does not return	8b bus	byte/hw/word/dw	--	--	--
Single Write 2:	64bit bus	byte/hw/word/dw	Fig. 1,3	--	--
Write by CPU Core	32bit bus	byte/hw/word/dw	--	--	--
Big-Endian	16bit bus	byte/hw/word/dw	--	--	--
	8b bus	byte/hw/word/dw	--	--	--
Single Write 3:	64bit bus	byte/hw/word/dw	--	--	--
Write by CPU Core	32bit bus	byte/hw/word/dw	--	--	--
Little-Endian	16bit bus	byte/hw/word/dw	--	--	--
	8b bus	byte/hw/word/dw	--	--	--
Atomic Load/Store:	64bit bus	byte/word	--	--	--
	32bit bus	byte/word	--	--	--
	16bit bus	byte/word	--	--	--
	8b bus	byte/word	--	--	--

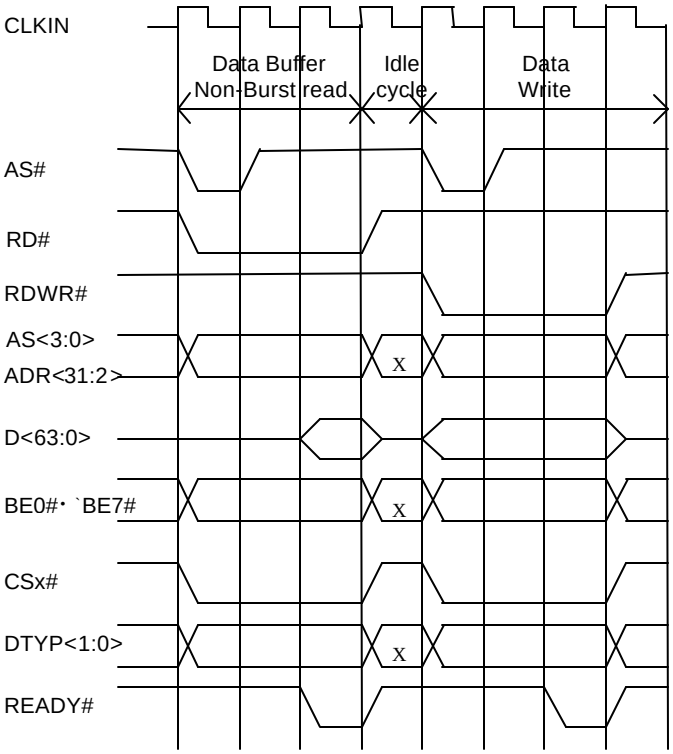


Figure 7-8 SPARClite Single Read/Write Operation

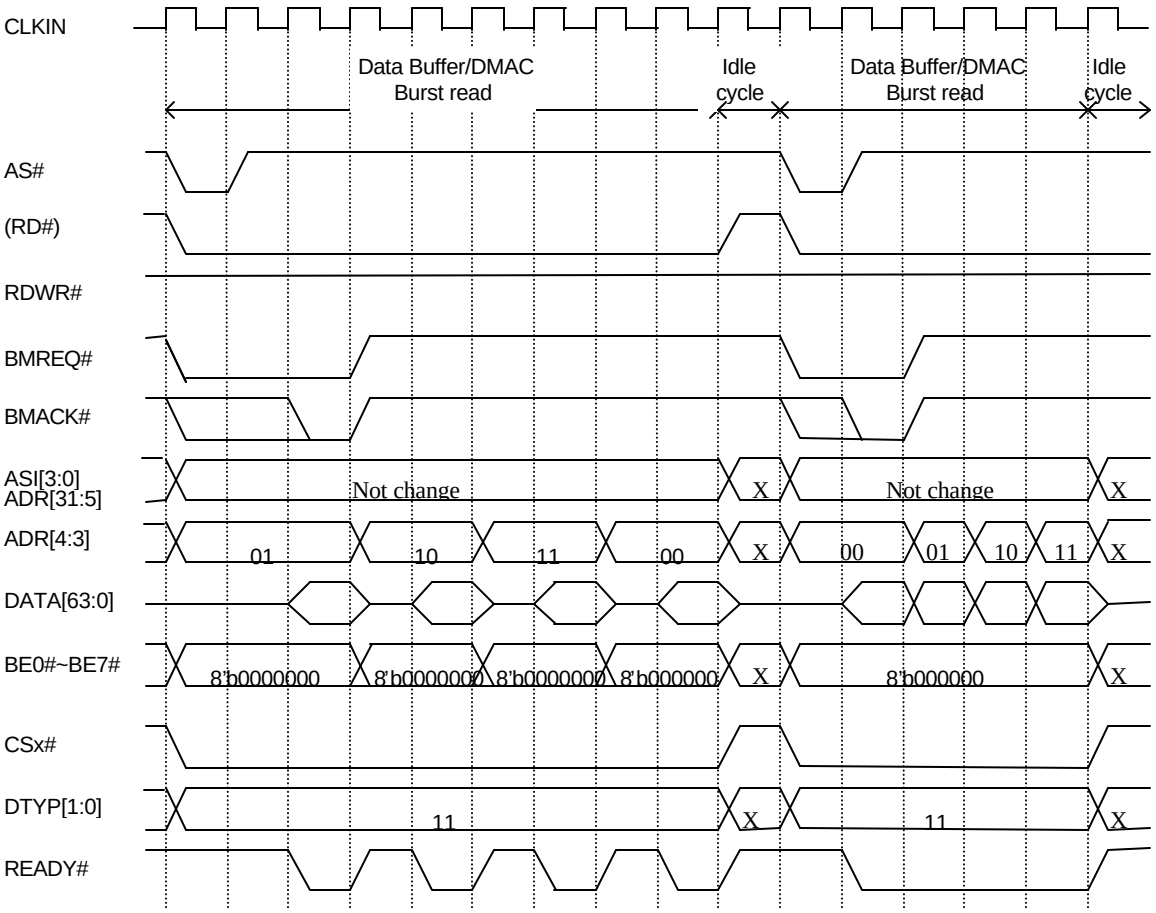


Figure 7-9 SPARClite Bus Burst Read Operation

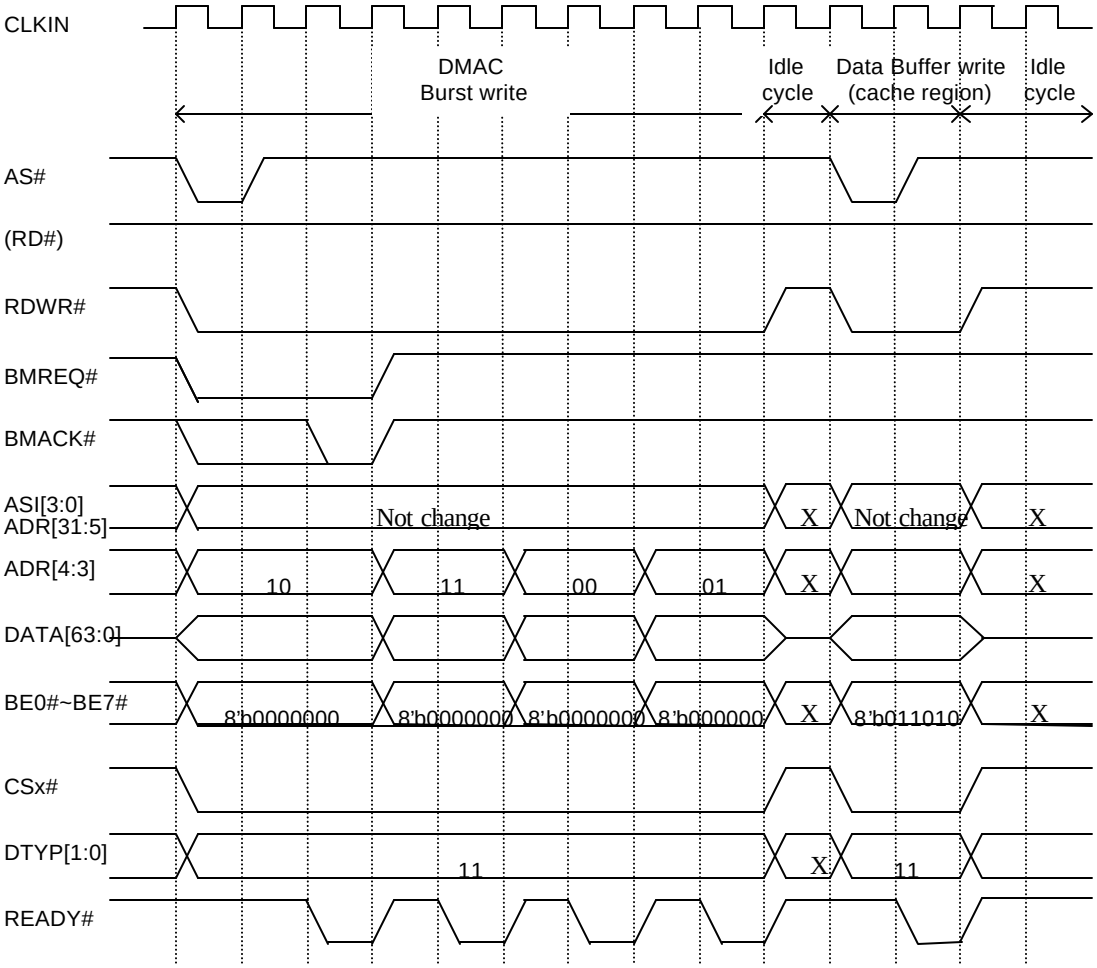


Figure 7-10 SPARClite Bus Burst WRITE Operation

7.4.2. Access to SDRAM Transparent Mode (MB86861 only)

During bus grant, the MB8686 supports a function whereby the external bus master on the SPARClite bus access SDRAM through the MB8686. Hereafter this function will be called Transparent Access Function, Transparent Access Mode or simply Transparent Mode, and transparent access to SDRAM will simply be called Transparent Access. The Transparent Access Function to SDRAM is only supported in 64-bit bus width, and access types are single mode and burst mode (burst length 4 or 8).

Transparent access is started by asserting AS# along with SDSEL#. The CPU fetches the address of the SDRAM to be accessed from ADR<31:3> in the cycle in which AS# is asserted and judges by the RDWR# level whether this is a read access or a write access. It also judges by the BMREQ# and BLEN8# levels whether this is a single transfer or 4 bursts or 8 bursts.

Figure 7-4 SDRAM Transparent Access Transfer Modes

BMREQ#	BLEN8#	Transfer Type
"H"	"H"	Single Transfer
"L"	"H"	4 bursts
"L"	"L"	8 bursts
"H"	"L"	Prohibited

7.4.2.1. Burst Reads

When RDWR#="H" and BMREQ#="L" are input in AS# assert timing, a burst read results. Since the CPU never cancels burst requests, BMACK# is always returned. RDYOUT# is asserted together with data output to the D<63:0> pin when the data is ready. During reads, all bytes are read from SDRAM whatever the level of BE0#-7#. With 4-burst addresses, the lower order 2 bits ADR<4:3> must be 00. Moreover, with 8 bursts, the address lower order 3 bits must be 000. After the CPU has fetched a AS# timing address, it updates and uses the address internally.

4-burst address ADR<4:3> update
00→01→10→11

8-burst address ADR<5:3> update
000→001→010→011→100→101→110→111

7.4.2.2. Burst Writes

When RDWR#="L" and BMREQ#="L" are input in AS# assert timing, a burst write results. Since the CPU never cancels burst requests, BMACK# is always returned. RDYOUT# is asserted when data is fetched from the D<63:0> pin to a buffer. During writes, BE0#~7# are fetched at the same time as the data is fetched, and only bytes which correspond to BE# which are "L" are written to SDRAM. With 4-burst addresses, the lower order 2 bits ADR<4:3> must be 00. Moreover, with 8 bursts, the address lower order 3 bits must be 000. After the CPU has fetched a AS# timing address, it updates and uses the address internally.

7.4.2.3. Single Reads

When RDWR#="H" and BMREQ#="H" are input in AS# assert timing, a single read results. RDYOUT# is asserted together with data output to the D<63:0> pin when the data is ready. During reads, all bytes are read from SDRAM whatever the level of BE0#-7#.

7.4.2.4. Single Writes

When RDWR#="L" and BMREQ#="H" are input in AS# assert timing, a single write results. RDYOUT# is

asserted when data is fetched from the D<63:0> pin to a buffer. During writes, BE0#~7# are fetched at the same time as the data is fetched, and only bytes which correspond to BE# which are “L” are written to SDRAM.

7.4.2.5. Atomic Access

In transparent access, there is no bus cycle equivalent to CPU Atomic Load/Store. Atomic access is possible by means of BREQ# control as long as an external bus master continues to assert BREQ#, since CPU Core access never interrupts between accesses.

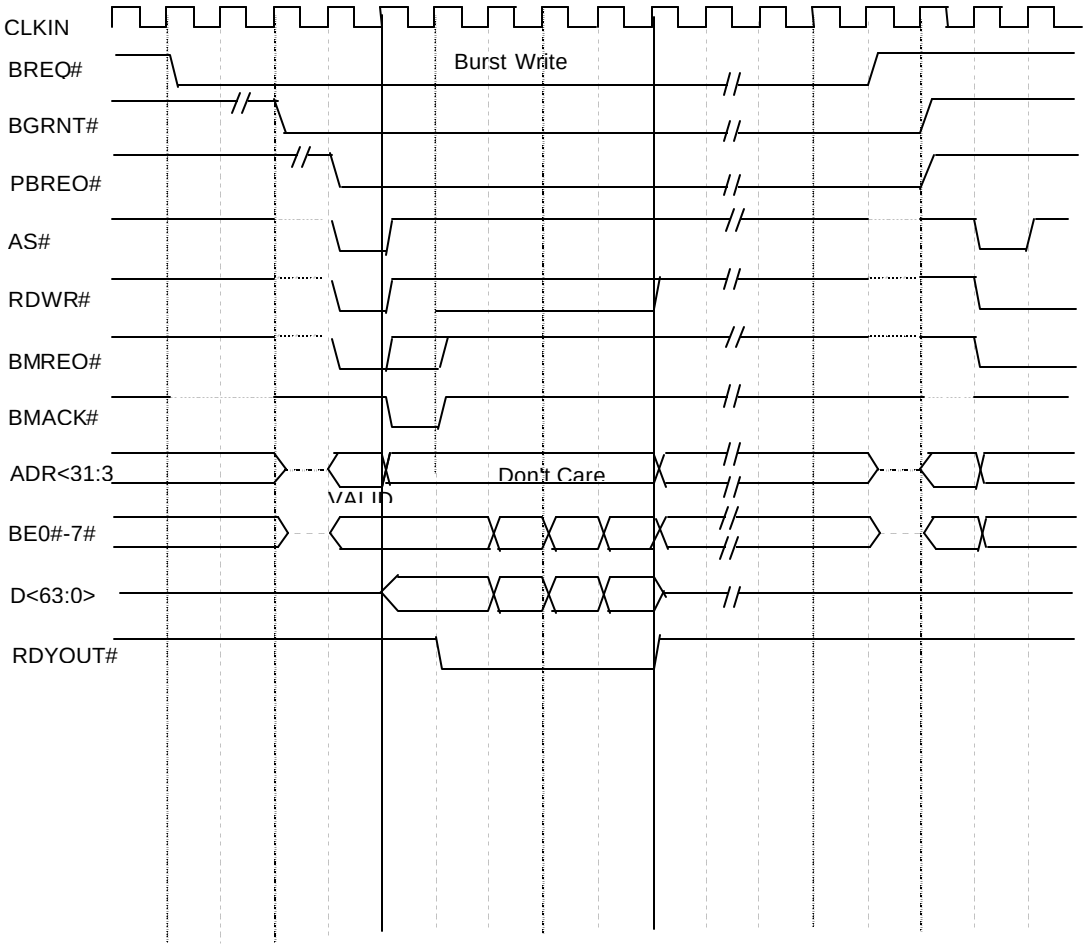


Figure 7-11 SPARClite Bus Open and Reopen

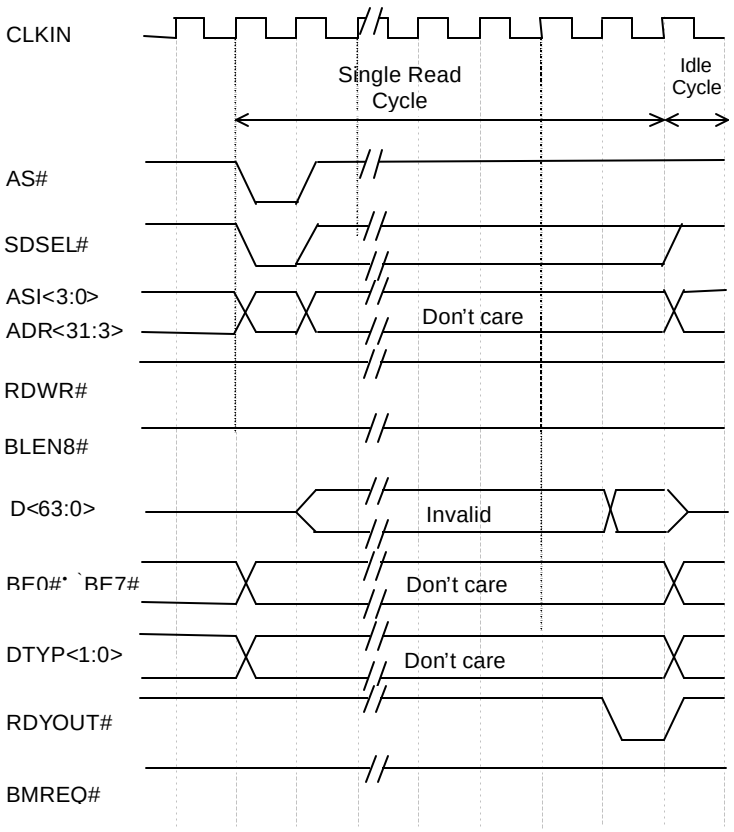


Figure 7-12 Single Reads in SDRAM Transparent Access

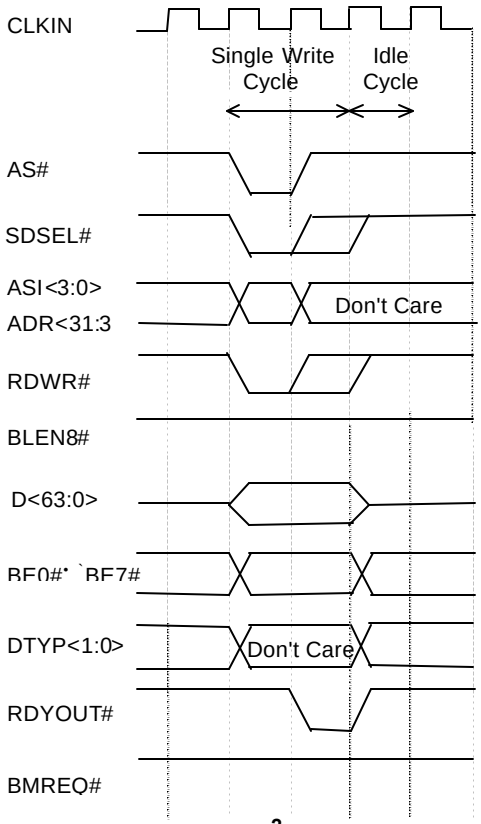


Figure 7-13[2] Single Writes in SDRAM Transparent Access

[2] Translator's Note: This is numbered Figure 7-5 in the Japanese text. Since there is already a Figure 7-5 on page 7-16, however, I have changed the number to read consecutively. The same is true of the figures in Section 7 which follow.

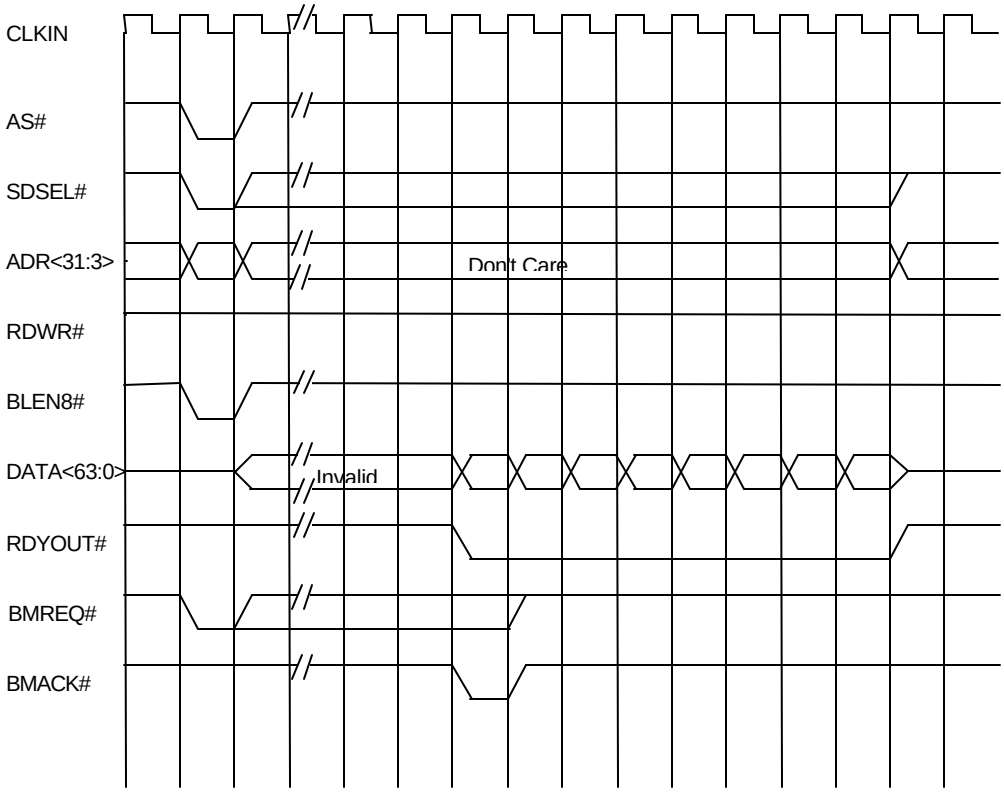


Figure 7-14 Burst Reads in SDRAM Transparent Access (Burst Length 8)

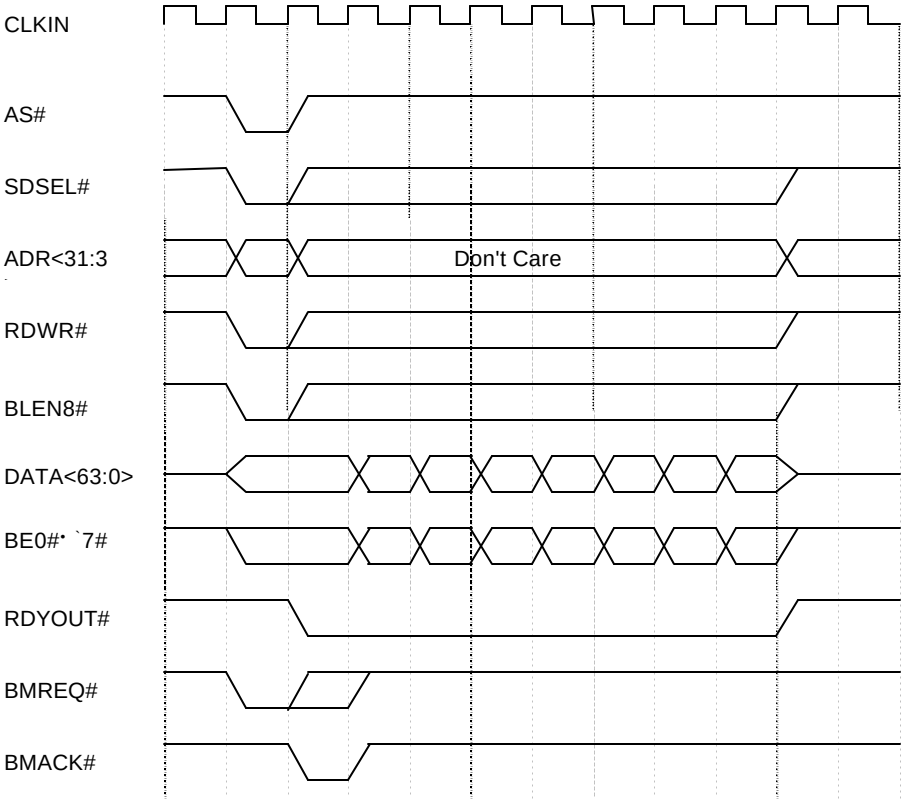


Figure 7-15 Burst Writes in SDRAM Transparent Access (Burst Length 8)

8. SDRAM-IF

8.1. Summary

The SDRAM-IF is a module attached to the Bus Interface Unit (BIU) which functions as an interface between the MB8686 Processor Core and external chip circuits, and it has the following functions:

(1) Has dedicated 13-bit address and 64-bit data bus pins.
(2) Performs address control with external SDRAM, data transfer control, command control etc.
(3) Performs SDRAM transfers in 64-bit bus at burst length 4. Also, in 32-bit bus, it performs them at burst length 8.
(4) CAS latency can be set at 2-3. Uses sequential burst types.
(5) RAS Precharge Time [tRP] and RAS-CAS Delay Time [tRCD] can be set respectively to 2-3 cycles.
(6) Outputs 4 SCS (SDRAM Chip Select) signals. The respective SCS areas can be specified by settings.
(7) Refresh can be set to auto-refresh and self-refresh.
(8) SDRAM read and write operations are performed in Burst Mode.

8.2. Pin Overview

Pin Name	I/O	Description
SCLK	O	SDRAM Clock Signal. Should be linked with SDRAM clock input. Clock frequency is the same as the internal IMB bus.
SCKE	O	SDRAM Clock Enable Signal.
SRAS#	O	SDRAM RAS Signal
SCAS#	O	SDRAM CAS Signal
SWE#	O	SDRAM Write Enable Signal
SCS3#-SCS0#	O	SDRAM Chip Select Signals
SBA<1:0>	O	SDRAM Bank Select Signal
SADR<12:0>	O	SDRAM Address Signal. Addresses are multiplexed.
SDQ<63:0>	I/O	SDRAM Data Bus
SDQM<0:7>	O	SDRAM Input Mask/Output Enable signal
SDP<0:7>	I/O	SDRAM Parity Signal

8.3. Register Overview

Table 8-2 SDRAM-IF Registers

ASI	Address	Register Name	Symbol	Reset Value			
				b31 b24	b23 b16	b15 b8	b7 b0
0x04	0x80000130	SDRAM Address Range Specifier Register0	SDARSR0	xxxxxxxx	xxxxxxxx	xxxxxxxx	xxxxxxxx
0x04	0x80000138	SDRAM Address Range Specifier Register1	SDARSR1	xxxxxxxx	xxxxxxxx	xxxxxxxx	xxxxxxxx
0x04	0X80000230	SDRAM Address Mask Register0	SDAMR0	xxxxxxxx	xxxxxxxx	xxxxxxxx	xxxxxxxx
0x04	0X80000238	SDRAM Address Mask Register1	SDAMR1	xxxxxxxx	xxxxxxxx	xxxxxxxx	xxxxxxxx
0x04	0x80000800	SDRAM I/F Configuration Register (*1)	SDCFG	00000000	00000000	00000100	00000000
0x04	0x80000808	Auto Refresh Timer Register	ART	00000000	00000000	11111111	11111111
0x04	0x80000810	SDRAM Status Register	SSR	00000000	00000000	00000000	00000000
0x04	0x80000820	SDRAM CS Characteristic Register0 (*2)	CSCR0	00000000	00000000	00000000	00000000
0x04	0x80000828	SDRAM CS Characteristic Register1 (*2)	CSCR1	00000000	00000000	00000000	00000000
0x04	0x80000830	SDRAM CS Characteristic Register2 (*2)	CSCR2	00000000	00000000	00000000	00000000
0x04	0x80000838	SDRAM CS Characteristic Register3 (*2)	CSCR3	00000000	00000000	00000000	00000000
0x04	0x80000840	SDRAM START Address Register0 (*2)	SSAR0	00000000	00000000	00000000	00000000
0x04	0x80000848	SDRAM START Address Register1 (*2)	SSAR1	00000000	00000000	00000000	00000000
0x04	0x80000850	SDRAM START Address Register2 (*2)	SSAR2	00000000	00000000	00000000	00000000
0x04	0x80000858	SDRAM START Address Register3 (*2)	SSAR3	00000000	00000000	00000000	00000000
0x04	0x80000860	SDRAM Address Mask Register0 (*2)	SAMR0	00000000	00000000	00000000	00000000
0x04	0x80000868	SDRAM Address Mask Register1 (*2)	SAMR1	00000000	00000000	00000000	00000000
0x04	0x80000870	SDRAM Address Mask Register2 (*2)	SAMR2	00000000	00000000	00000000	00000000
0x04	0x80000878	SDRAM Address Mask Register3 (*2)	SAMR3	00000000	00000000	00000000	00000000

(*1) Functions in the MB86860 and the MB86861differ in part.

(*2) This register is provided only in the MB86861. It is not provided in the MB86860.

8.4. Register Details

8.4.1. SDRAM Address Range Specifier Registers (SDARSR0 1)

These are the registers for setting SDRAM bus areas. Address range start addresses are set to these registers. Bits not performing address comparisons are set in the SDAM Registrers. When the SDARSR.N bit is 1, its range is made a non-cache area. All areas must be set so as not to overlap. (They must also be set so as not to overlap with CS0#–CS5# areas).

31	30	21	22	16	15	0
N	Reserved		ASI<5:0>		Address<31:16>	

Symbol : SDARSR0–5
Address : 0x80000130-0x80000138 (ASI=0x4)
Reset State : undefined

SDARSR0 (Address 0x80000130, ASI=0x4) •••• for setting SDRAM area no corresponding CS# (undefined on reset)
SDARSR1 (Address 0x80000138, ASI=0x4) •••• for setting SDRAM area no corresponding CS# (undefined on reset)

bit	Field Name	R/W	Description
31	N	R/W	- Non-cache Sets cacheable/non-cacheable of corresponding chip select areas. 1 : Sets corresponding chip select areas to non-cache areas. 0 : Sets corresponding chip select areas to cache areas.
30-22	Reserved		
21-16	ASI<5:0>	R/W	Sets ASI<5:0> of corresponding chip select areas.
15-0	Address<31:16>	R/W	Sets upper 6 bits of corresponding chip select area start addresses.

Figure 8-1 SDARSR Register

8.4.2. SDRAM Address Mask Registers (SDAMR0 1)

These are the registers for setting SDRAM address range. Address ranges must be set in combination with the SDARS Registers. Bits set to "1" in SDAMR Registers do not compare addresses with the SDARSR Registers. Only bits set to "0" in the SDAMR Registers compare memory addresses with the SDARSR Registers and, if they match, regard it as access to SDRAM.

31	22	21	16	15	0
Reserved		ASI mask<5:0>		Address mask <31:16>	

Symbol : SDAMR0–1
Address : 0x80000230-0x80000238 (ASI=0x4)
Reset State : undefined (AMR0=0x00030001)

SDAMR0 (Address 0x80000230, ASI=0x4) •••• no corresponding CS# (undefined on reset)
SDAMR1 (Address 0x80000238, ASI=0x4) •••• no corresponding CS# (undefined on reset)

bit	Field Name	R/W	Description
31-22	Reserved		

21-16	ASI mask<5:0>	R/W	Sets SDRAM Area ASI masks.
15-0	Address mask<31:16>	R/W	Sets SDRAM Area Address masks.

Figure 8-2 Address Mask Register

8.4.3. SDRAM Configuration Register (SDCFG)

In order to enable SDRAM I/F functions, the SE bit of this register must be set to "1". When an SDRAM which is connected to the MB8686 comes with register functions, Buffer Mode (BUF) should be enabled. Before SDRAM Enable [SE] is enabled, the SDRAM CS Characteristic Registers must be set.

31 13	12	11 9	8 5	4	3	2	1	0
MB86860								
Reserved	BUF	CL	ST	BS	PRC	SPC	SPE	SE
MB86861								
Reserved	BUF	Reserved	0000	Reserved				SE

Symbol : SDACFG
Address : 0x80000800 (ASI=0x4)
Reset State : 0x00000400

bit	Field Name	R/W	Description
31-22	Reserved		
21-16	ASI mask<5:0>	R/W	Sets SDRAM Area ASI masks.
15-0	Address mask<31:16>	R/W	Sets SDRAM Area Address masks.

Figure 8-3 SDRAM Configuration Register[3]

8.4.4. Auto Refresh Register (ART)

This is the register for setting the counter values which correspond to refresh spaces. Refresh counters operate synchronous to the SDRAM clock.

31 16	15 0
Reserved	TC

Symbol : ART
Address : 0x80000808 (ASI=0x4)
Reset State : 0x00000

[3] Translator's Note: This figure is reproduced as it is in the Japanese text, i.e. the same as Figure 8-2 on page 8-3.

bit	Field Name	R/W	Description
31-16	Reserved		
15-0	TC	R/W	- Timer Count Specifies issue space of the Auto Refresh (REF) command issued to SDRAM. (ART.TC / ((input clock frequency) × (clock frequency multiplier frequency)) (sec) is the REF command issue space.

Figure 8-4 Auto Refresh Timer Register

8.4.5. SDRAM Status Register (SSR)

This register indicates the status of the SDRAM I/F Unit. When the initial settings to SDRAM have been completed, 1'b1 is set to the SDI bit. This bit is Read Access Only. When parity errors occur in SDRAM data reads, 1'b1 is set to the PER bit. PER bit resets are performed by writing 1'b0 to the PER bit. In SDRAM access, should a request occur for access to an address which does not belong to any CS, 1'bl is set to the MEXC bit. MEXC bit resets are performed by writing 1'b0 to the MEXC bit.

31	6	5	4	3 1	0
Reserved		MEXC	PER	Reserved	SDI

Symbol : SS
Address : 0x80000810 (ASI=0x4)
Reset State : 0x00000000

bit	Field Name	R/W	Description
31-6	Reserved		
5	MEXC	R/W	- Memory Exception Flag (MB86861 only) In access to SDRAM, when a request occurs for access to addresses which do not belong to any SDCS areas, "1" is set. This bit is cleared by writing "0".
4	PER	R/W	- Parity Error Flag When parity errors occur in SDRAM data reads, "1" is set. This bit is cleared by writing "0".
3-1	Reserved		
0	SDI	R	- SDRAM Initial Wait Indicator This is a flag which indicates that initializing of SDRAM has been completed. 0: Initialization not completed 1: Initialization completed

Figure 8-5 SDRAM Status Register

8.4.6. SDRAM CS Characteristic Registers (CSCR0-3) (MB86861 only)

31	19	18	16	15	14	12	11	9	8	5	4	3	2	1	0
Reserved		tRP		R	TRCD		CL		ST		BS	PRC	SP	SP	SE

Symbol : CSCR0-3
Address : 0x80000820-0x80000838 (ASI=0x4)
Reset State : 0x00000000

CSCR0 (Address 0x80000820, ASI=0x4) • • • • for setting SDRAM area SDCS0# (0x00000000 on reset)
CSCR1 (Address 0x80000828, ASI=0x4) • • • • for setting SDRAM area SDCS1# (0x00000000 on reset)
CSCR2 (Address 0x80000830, ASI=0x4) • • • • for setting SDRAM area SDCS2# (0x00000000 on reset)
CSCR3 (Address 0x80000838, ASI=0x4) • • • • for setting SDRAM area SDCS3# (0x00000000 on reset)

bit	Field Name	R/W	Description
31-17	Reserved		
18-16	tRP	R/W	-RAS Precharge Time Sets SDRAM tRP (RAS Precharge Time) used in corresponding SDCS# areas. 000–001 : Reserved 010 : 2 cycle 011 : 3 cycle 100–111 : Reserved
15	Reserved		
14-12	tRCD	R/W	- RAS-CAS Delay Time Sets SDRAM tRCD (RAS-CAS Delay Time) used in corresponding SDCS# areas. 000–001 : Reserved 010 : 2 cycle 011 : 3 cycle 100–111 : Reserved
11-9	CL	R/W	- CAS Latency Sets SDRAM CL (CAS Latency) used in corresponding SDCS# areas. 000–001 : Reserved 010 : 2 cycle 011 : 3 cycle 100–111 : Reserved
8-5	SDRAM Type	R/W	- ST Sets SDRAM Types used in corresponding SDCS# areas. Indicates details by xxx.
4	BS	R/W	- Bus Size Sets bus width of corresponding SDCS# areas. 0 : 32-bit bus width 1 : 64-bit bus width
3	PRC	R/W	- Precharge Control Sets Precharge Systems of corresponding SDCS# areas 0 : uses all bank precharge command (PALL) 1 : uses precharge commands bank by bank (PRE)
2	SP	R/W	- SDRAM Parity Control Sets parity types used in corresponding SDCS# areas 0 : Odd Parity 1 : Even Parity
1	SP	R/W	- SDRAM Parity Enable Enables Parity Check Function for corresponding SDCS# areas. 0 : Parity Disable 1 : Parity Enable
0	Reserved		

Figure 8-6 SDRAM CS Characteristic Register

8.4.7. SDRAM Start Address Registers (SSAR0-3) (MB86861 only)

This register is for setting SDRAM address areas which correspond to the SDCS. The upper order 16 bits of the start addresses of corresponding SCS areas should be set to the SSAR Register. Address range is decided by making settings grouped with the SAM register.

31	15	0
Reserved	SDRAM Start Address [32:15]	

Symbol : SSAR0-3
Address : 0x80000840-0x80000858 (ASI=0x4)
Reset State : 0x00000000

SSAR0 (Address 0x80000840, ASI=0x4) • • • • for setting SDRAM area SDCS0# (0x00000000 on reset)
SSAR1 (Address 0x80000848, ASI=0x4) • • • • for setting SDRAM area SDCS1# (0x00000000 on reset)
SSAR2 (Address 0x80000850, ASI=0x4) • • • • for setting SDRAM area SDCS2# (0x00000000 on reset)
SSAR3 (Address 0x80000858, ASI=0x4) • • • • for setting SDRAM area SDCS3# (0x00000000 on reset)

bit	Field Name	R/W	Description
31-16	Reserved		
15-0	Address<31:16>	R/W	-SDRAM Start Address [31:16]

Figure 8-7 SDRAM Start Address Register

8.4.8. SDRAM Address Mask Registers (SAMR0-3) (MB86861 only)

This register is for setting SDRAM Area address mask values which correspond to the SDCS. Bits set "1" in the SAM Register do not compare addresses with the SSAR registers. Only bits set to "0" in the SAMR registers compare memory addresses with the SSAR registers, and when they match, the corresponding SDCS# are asserted.

31	15	0
Reserved	SDRAM Address Mask[32:15]	

Symbol : SAMR0-3
Address : 0x80000840-0x80000858 (ASI=0x4)
Reset State : 0x00000000

SAMR0 (Address 0x80000840, ASI=0x4) • • • • for setting SDRAM area SDCS0#
(0x00000000 on reset)
SAMR1 (Address 0x80000848, ASI=0x4) • • • • for setting SDRAM area SDCS1#
(0x00000000 on reset)
SAMR2 (Address 0x80000850, ASI=0x4) • • • • for setting SDRAM area SDCS2#
(0x00000000 on reset)
SAMR3 (Address 0x80000858, ASI=0x4) • • • • for setting SDRAM area SDCS3#

(0x00000000 on reset)

bit	Field Name	R/W	Description
31-16	Reserved		
15-0	Address Mask <31:15>	R/W	-SDRAM Address Mask<31:15>

Figure 8-8 SDRAM Address Mask Register

8.5. Operating Functions

8.5.1. Parity Generation and Check Function

During read access to the relevant SCS the parity check function is enabled, and during write access the parity generation function is enabled by setting "1" to bit1 (SPE) of the SDRAM CS Characteristic Register (CSCR). The SDRAM I/F Unit performs SDQ signal byte by byte odd/even parity generation and checks in accordance with the setting of CSCR bit2 (SPC).

8.5.2. SDRAM Connections with SDRAM Buses

8.5.2.1. Addresses

SDRAM-IF Address Ourput Pins and SDRAM-IF Address Input Pins in each operating mode are as shown below.

Table 8-3 SDRAM Addresses

SDRAM		MB8686x SDRAM-IF Pin Names				
		SBA1	SBA0	SAD12	SAD11	SAD10-0
256Mbit (T.B.D.)	8Mwordx8bitx4bank	A14	A13	A12	A11	A10-0
	4Mwordx16bitx4bank	A14	A13	A12	A11	A10-0
	2Mwordx32bitx4bank	A13	A12	-	A11	A10-0
128Mbit (T.B.D.)	4Mwordx8bitx4bank	A13	A12	-	A11	A10-0
	2Mwordx16bitx4bank	A13	A12	-	A11	A10-0
	1Mwordx32bitx4bank	A12	A11	-	-	A10-0
64Mb	2Mwordx8bitx4bank	A13	A12	-	A11	A10-0
	1Mwordx16bitx4bank	A13	A12	-	A11	A10-0
	512Kwordx32bitx4bank	A12	A11	-	-	A10-0
16Mb	1Mwordx8bitx2bank	-	A11	-	-	A10-0
	512Kwordx16bitx2bank	-	A11	-	-	A10-0

8.5.2.2. SDRAM Data Mask (SDQM)

All SDRAM Data Mask (SDQM) bits and SDRAM Data Correspondence are as shown below.

Table 8-4 SDRAM Data Mask

Bus Size	SDQ							
	<63:56>	<55:48>	<47:40>	<39:32>	<31:24>	<23:16>	<15:8>	<7:0>
64bit mode	SDQM0	SDQM1	SDQM2	SDQM3	SDQM4	SDQM5	SDQM6	SDQM7
32bit mode	NC	NC	NC	NC	SDQM4	SDQM5	SDQM6	SDQM7

8.5.2.3. SDRAM Parity

All SDRAM Parity bits and SDRAM Data Correspondence are as shown below.

Table 8-5 SDRAM Parity

Bus Size	SDQ							
	<63:56>	<55:48>	<47:40>	<39:32>	<31:24>	<23:16>	<15:8>	<7:0>
64bit mode	SDP0	SDP1	SDP2	SDP3	SDP4	SDP5	SDP6	SDP7
32bit mode	NC	NC	NC	NC	SDP4	SDP5	SDP6	SDP7

SDRAM DIMM Connection Example (under consideration)

This is an example of using SDRAM DIMM as a technique for attaining large capacity. SDRAM DIMM which can be connected to the MB8686x is limited to products loading PLL logic and register buffers. An example of a Block Diagram of SDRAM DIMM and its connections with the MB8686x is shown below.

Table 8-6 SDRAM Signals

CK[0:3]	Clock Input	A[0:9, 11:13]	Address Input
CKE[0:1]	Clock Enable	A10/AP	Address Input/Auto-precharge
RAS#	Row Address Strobe	BA0-BA1	SDRAM Bank Address
CAS#	Column Address Strobe	REGE	Register Enable
WE#	Write Enable	DQ[0:63]	Data Input/Output
S#[0:3]	Chip Select	DQMB[0:7]	Data Mask

- SDRAM DIMM Block Diagram Example-1

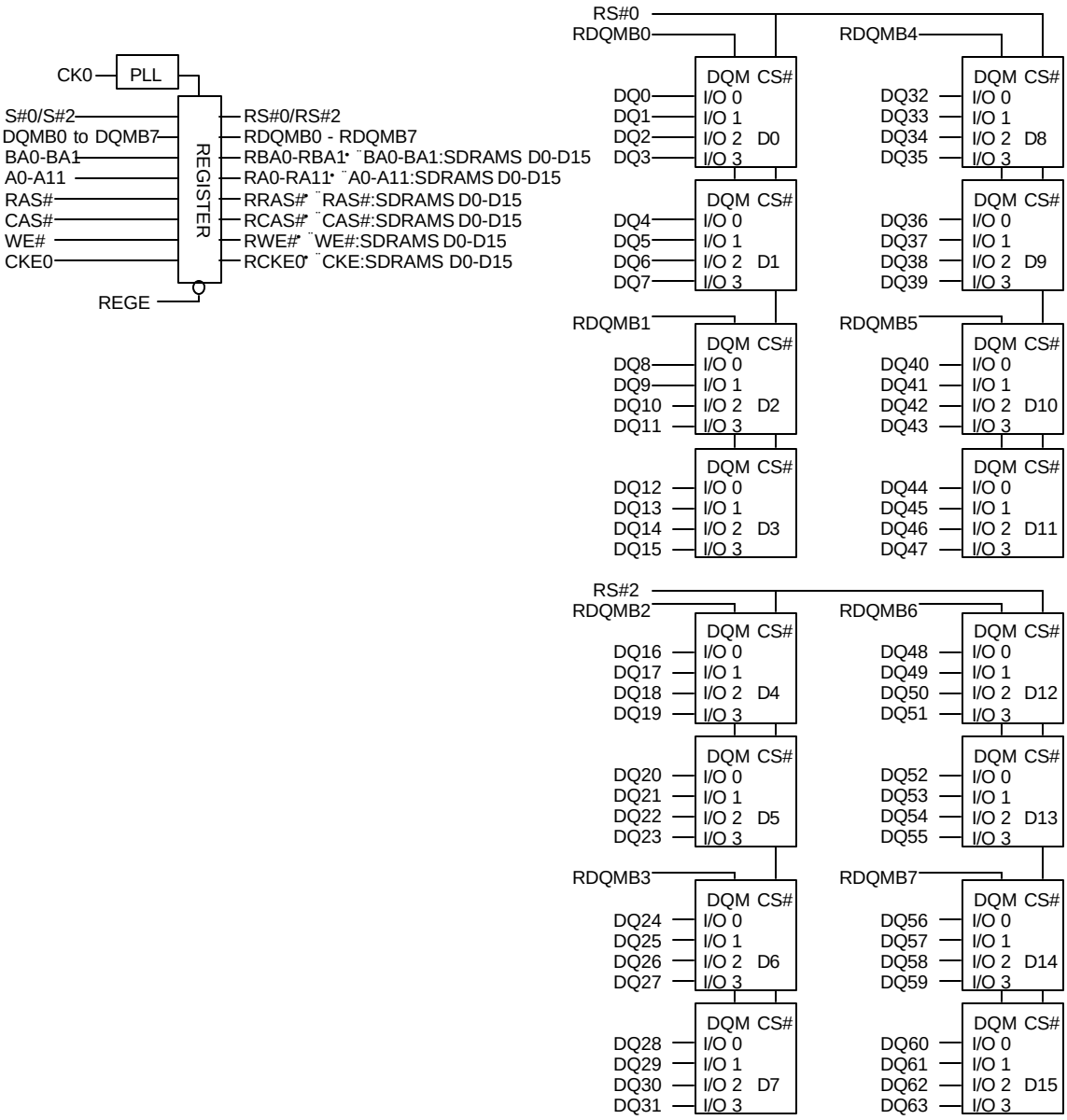


Figure 8-3 SDRAM DIMM Block Diagram Example-1

SDRAM DIMM Block Diagram Example-2

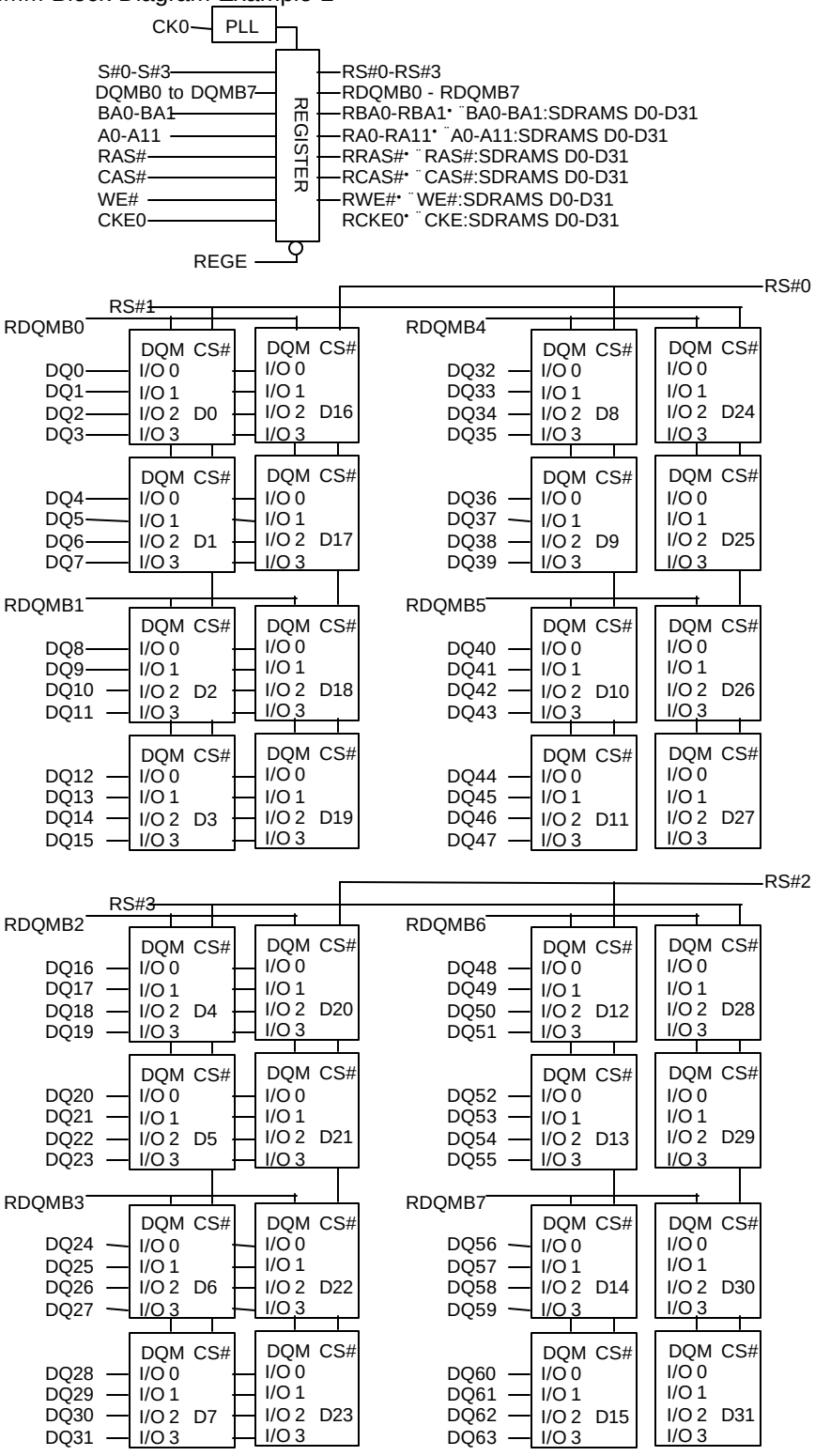


Figure 8-4 SDRAM DIMM Block Diagram Example-2

- Example of SDRAM DIMM, shown in Fig 8-3 SDRAM DIMM Block Diagram Example-1, connections with the MB8686x.

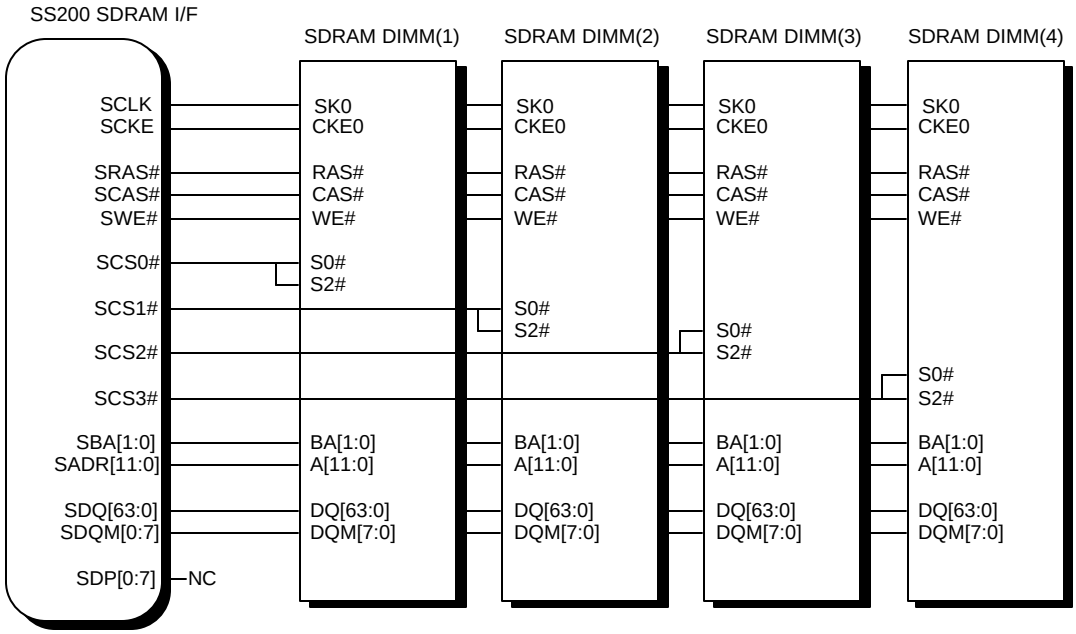


Figure 8-5 Connect to MB8686 SDRAM I/F

- Example of SDRAM DIMM, shown in Fig 8-4 SDRAM DIMM Block Diagram Example-2, connections with the MB8686x.

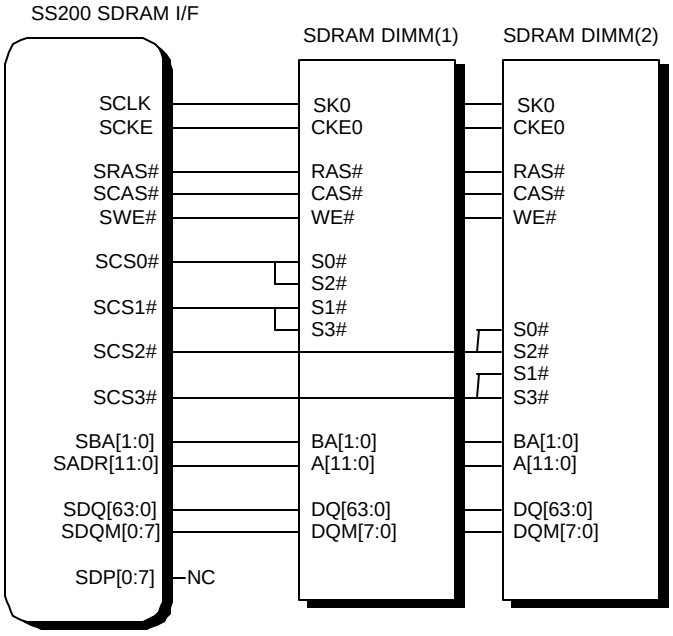


Figure 8-6 Connect to MB8686X SDRAM

Note) As shown in table 8-4, SDQM0 controls data<63:56> and SDQM[7:0] of the MB8686x should be

connected to DQM[7:0] respectively when using SDRAM DIMMs shown in Fig 8-3 and 8-4.

8.5.3. Timing Diagrams

8.5.3.1. Access Time

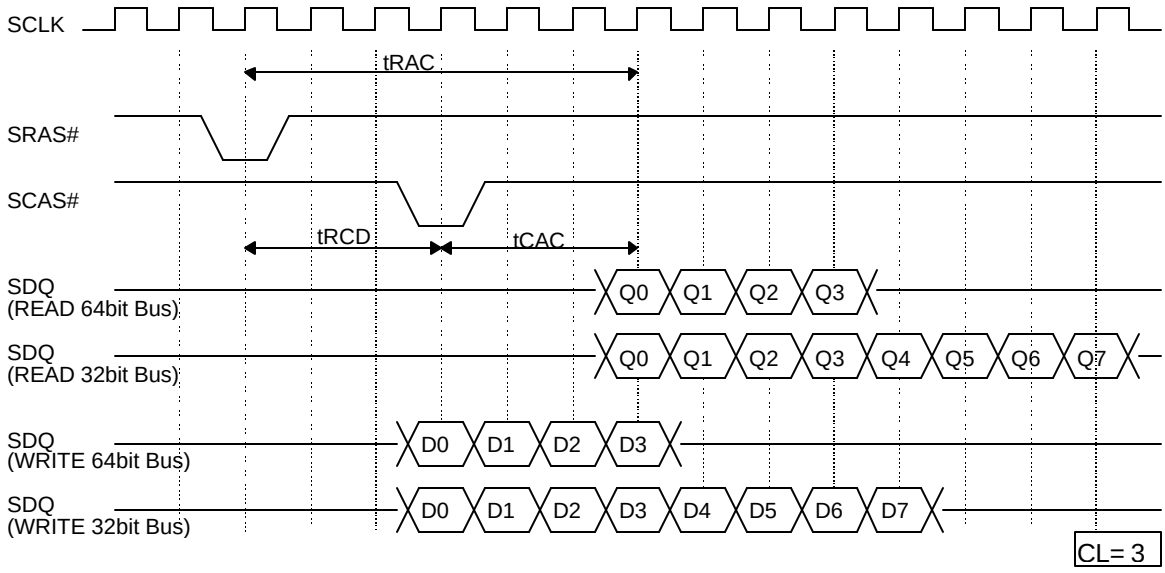


Figure 8-9 Access Time

8.5.3.2. CAS Output Delay

(1) Same Bank Access (read)

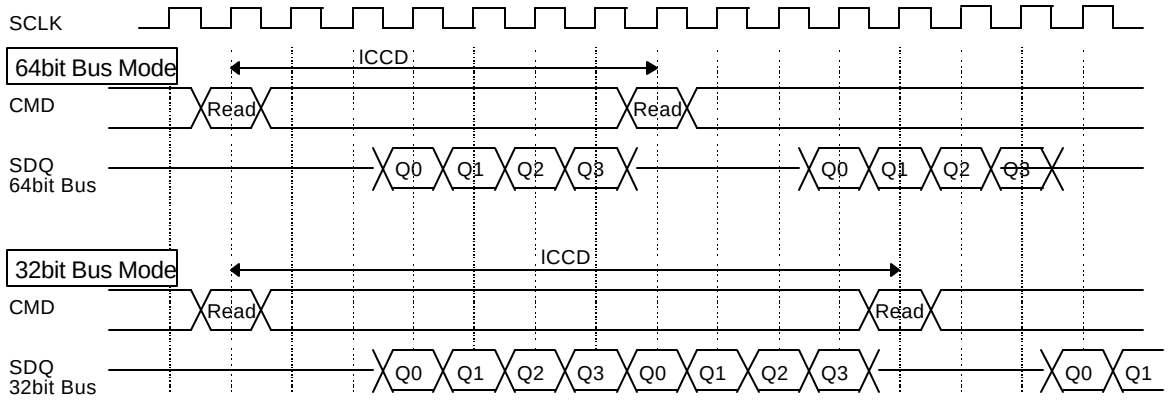
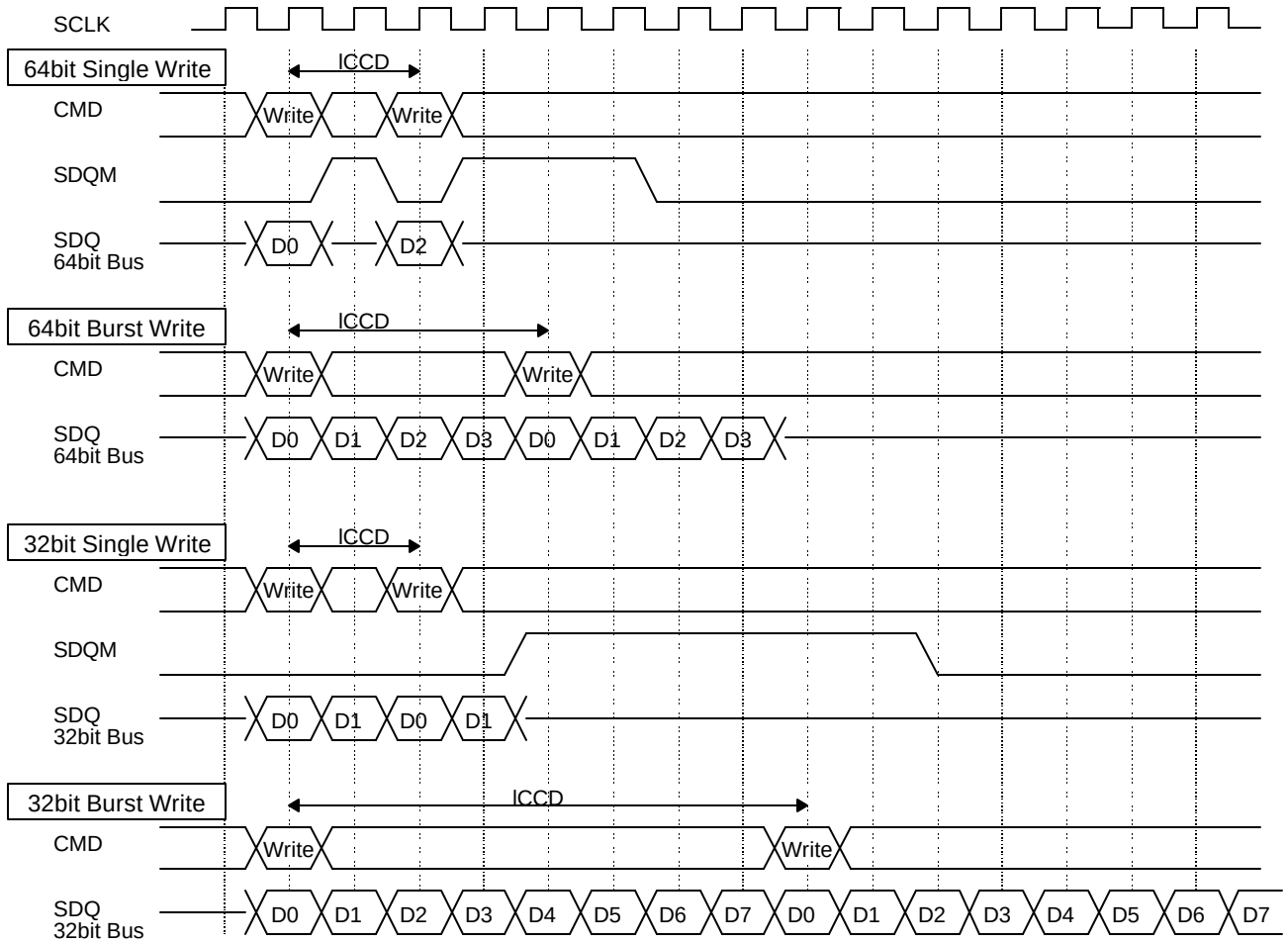


Figure 8-10 CAS-CAS Delay

(2) Same bank access (write) and mask by DQ

Data mask control by DQ is performed only for write data to SDRAM



(3) Different bank access

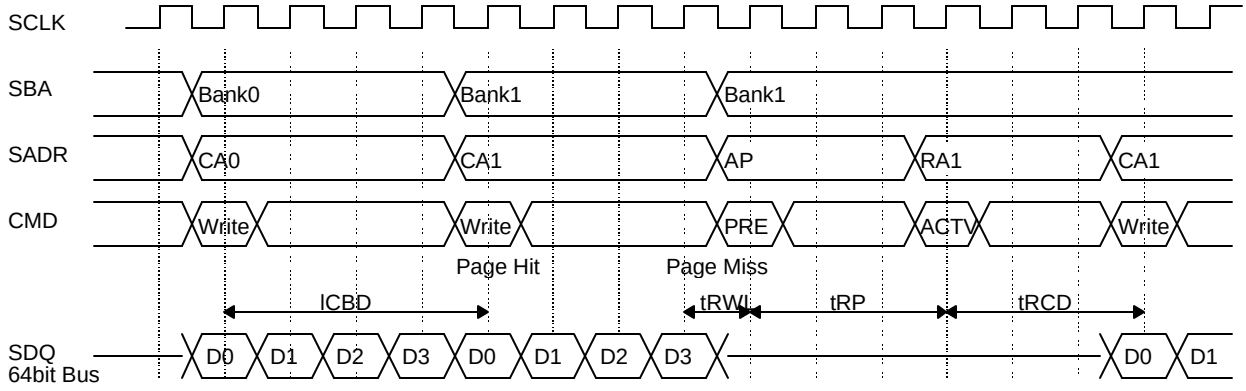


Figure 8-11 Access to Different Banks

8.5.3.3. RAS Output Delay

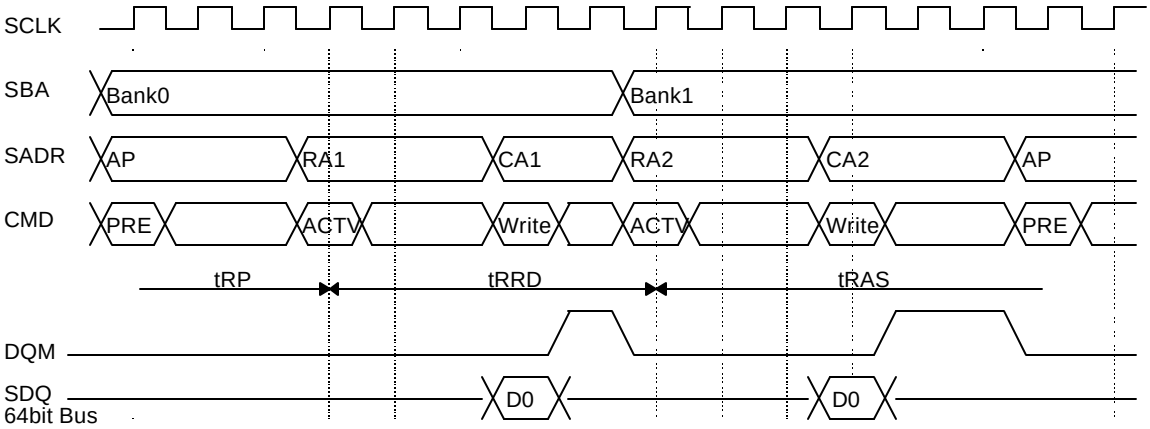
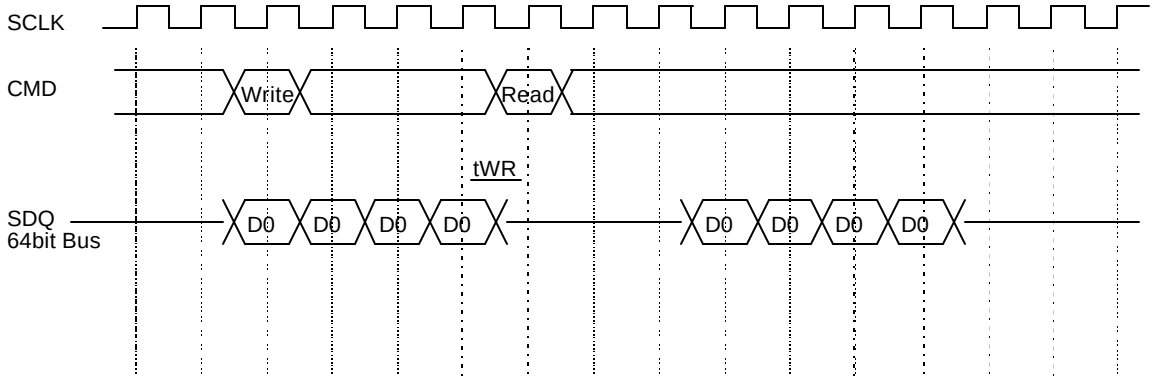


Figure 8-12 RAS-RAS Delay

8.5.3.4. Timing from Writes to Reads



8.5.3.5. Initialize Operation

The SDRAM I/F performs the following initialize operations for SDRAM after a Reset signal has been canceled:

(1) Transmits a 200-cycle nop.
(2) Issues a PALL command and precharges all SDRAM banks after the SDRAM I/F Control Register (SCR) SE bit becomes valid.
(3) Issues the Auto-refresh (REF) command 8 times. Issue space is tRC.
(4) Performs operations (2)-(3) for all CS defined by the SDRAM CS Characteristic Register (CSCR).
(5) Issues Mode Register Set Command (MRS) for SDRAM in accordance with contents defined by SDRAM CS Characteristic Register (CSCR).
(6) Sets “1” to the SDRAM Initial Wait Indicator (SDI) of the SDRAM Status Register (SSR).

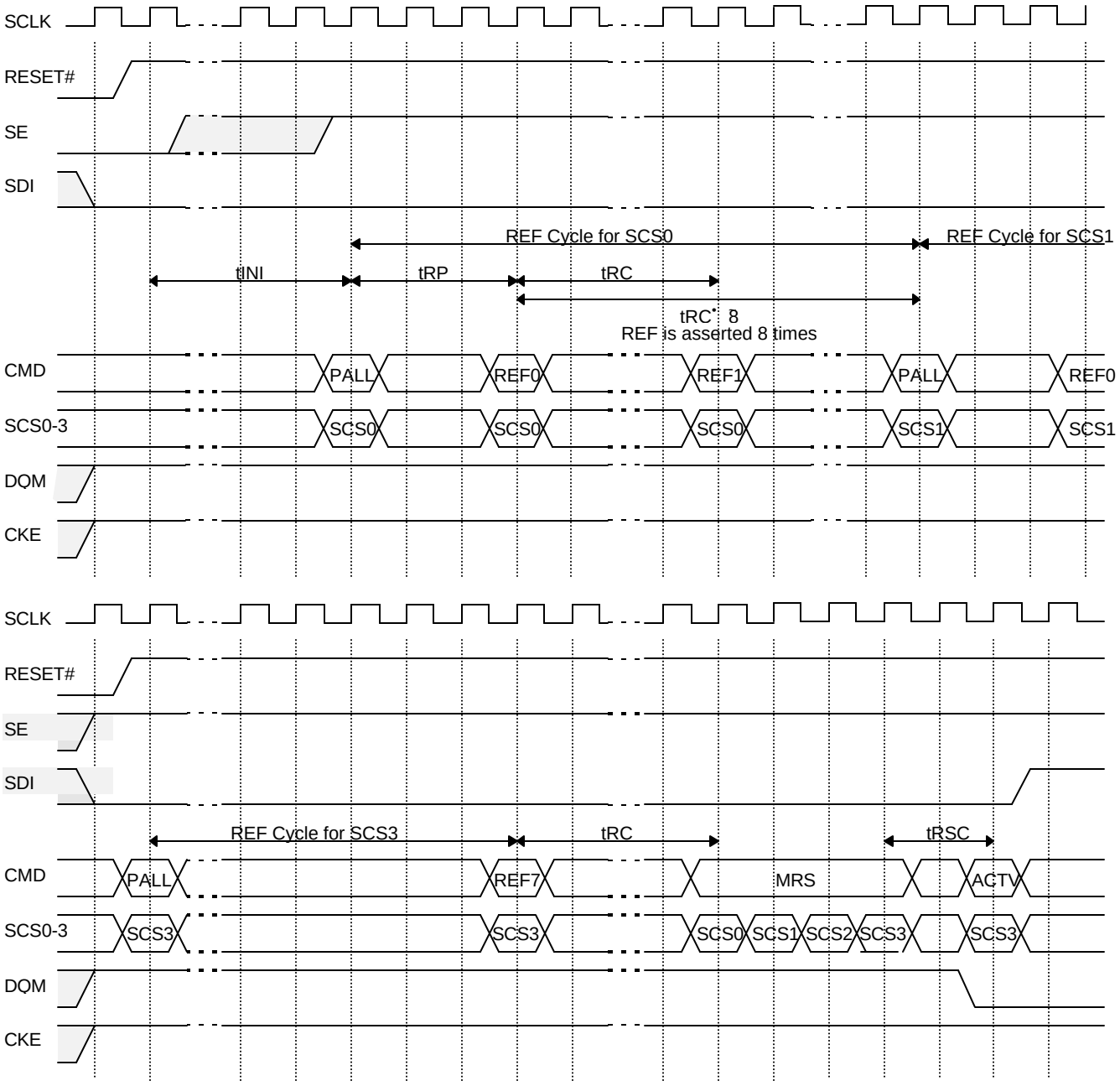


Figure 8-13 Initialization Sequence

Programmers should use the following procedures to confirm the initial settings of the SDRAM to be used as well as the end of the initialize operation:

(1) Use the SDRAM Start Address Registers (SSAR0-3) and the SDRAM Address Mask Registers (SAMR0-3) to specify SDRAM CS area ranges to be used.
(2) Set the characteristic information of the SDRAM connected to the SDRAM CS Characterietic Register (CSCR) to each CS. At the same time, enable the SDRAM Enable (SE) bit of the CSCR.
(3) Set Auto Refresh (REF) space to the Auto Refresh Timer Register (ART). Set values should meet Refresh Cycle requirements and should not exceed RAS Active TIME (tRAS) maximum values.
(4) Enable SDRAM I/F Enable (SE) of the SDRAM I/F Control Register (SCR).
(5) Wait until "1" is set to the SDRAM Initial Wait Indicator (SDI) bit of the SDRAM Status Register (SSR).
(6) Access SDRAM areas after the SSR SDI bit has changed to "1".

8.5.3.6. Refresh

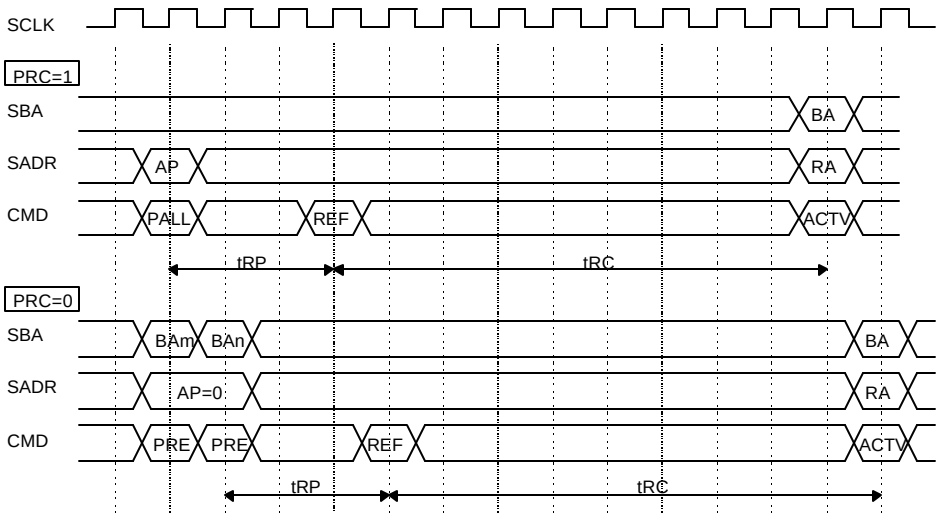


Figure 8-14 Refresh

8.5.3.7. Registered Mode

SDRAM DIMM ..bit12

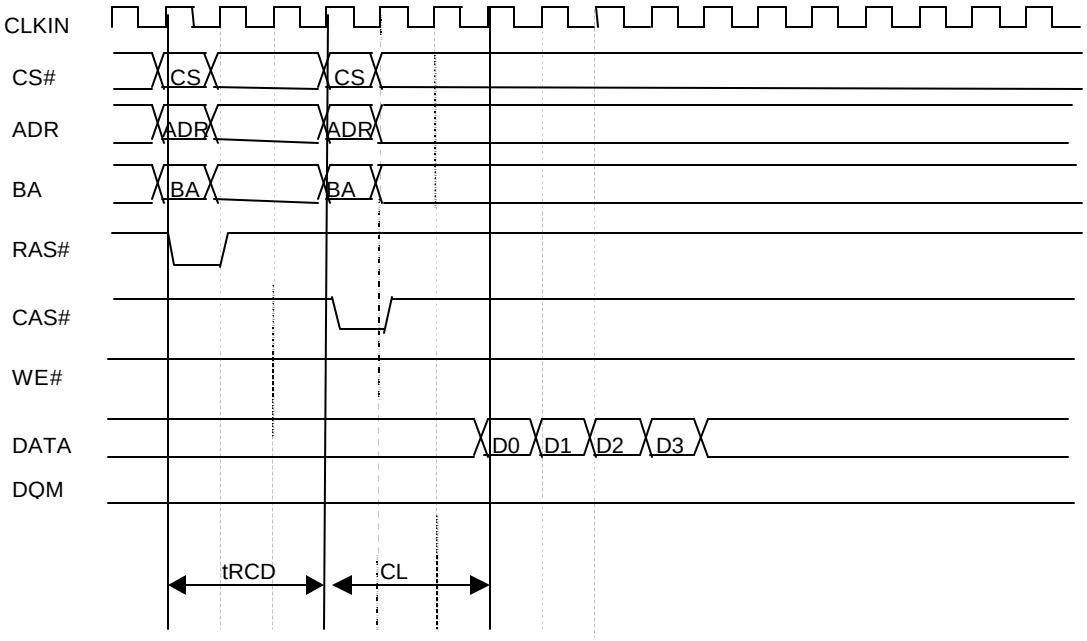
Unregistered 0
Registered 1

When SDRAM DIMM is used in registered mode, CAS Latency(CL) of SDRAM is increased by 1cycle.

SDRAM Ctlr. of the MB86860 aloso behaves in the way.

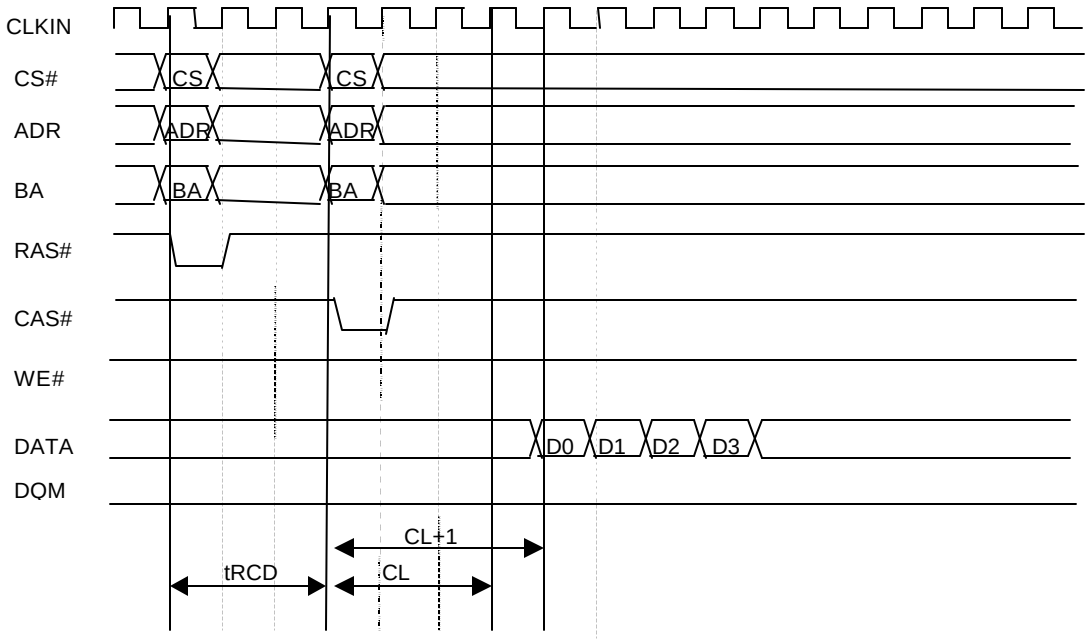
1. Read Sequence

<Unregistered Mode, CL=2>



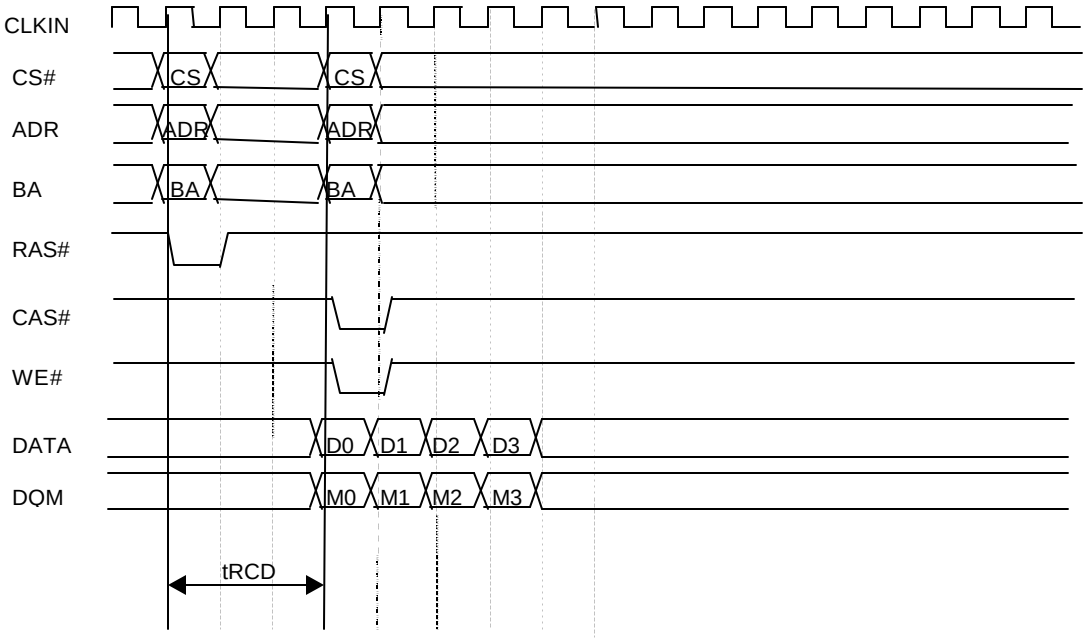
<Registered Mode, CL=2>

On the Registered SDRAM DIMM, Control Signals, which are CS,BA,ADR,RAS,CAS,WE and DQM, are delayed with 1 cycle and Data Output from SDRAM DIMM is also delayed with 1 cycle.

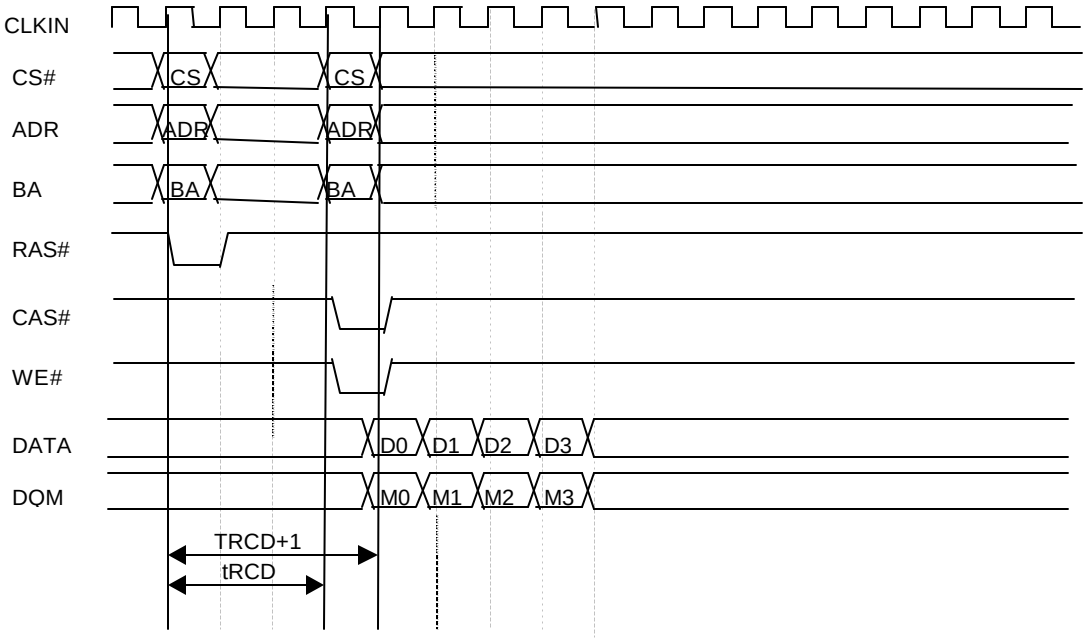


2. Write Sequence

<Write Operation on Unregistered Mode>



On the Registered SDRAM DIMM, Control Signals, which are CS,BA,ADR,RAS,CAS,WE and DQM, are delayed with 1 cycle and Data Input to SDRAM DIMM also needs to be delayed with 1 cycle.



REGE Signal

REGE(Register Enable) is a control signal of SDRAM DIMM itself and is not a signal from MB86860.

When REGE signal is active High, it needs to be High level on the user's system. It is the specification of SDRAM DIMM.

8.5.3.7. Values Predetermined by Number of SDRAM I/F Cycles

Table 8-7 Values Predetermined by Number of SDRAM I/F Cycles

Parameter	Symbol	MIN	unit	notes
RAS# Access Time	tRAC	4	cycle	
RAS#-CAS# Delay Time	tRCD	2	cycle	
CAS# Access Time	tCAC	2	cycle	
CAS# to CAS# Delay	CCD	2	cycle	
CAS# Bank Delay	CBD	4	cycle	
RAS# Cycle Time	tRC	9	cycle	
RAS# Precharge Time	tRP	2	cycle	
RAS# Active Time	tRAS	6	cycle	
Write Recovery Time	tWR	1	cycle	
RAS# -RAS# Bank Active Delay Time	tRRD	5	cycle	
Mode Register Set Cycle Time	tRSC	2	cycle	
SDRAM Initialization Time	tINI	200	µs	

9. DMAC

9.1. Summary

The DMAC supports flow-through transfers between SPARClite buses and SDRAM buses. All set registers have 2 channels. Simultaneous operation is 1 channel only. The DMAC cannot be used when Transparent Access Mode is in use.

- Flow-through transfers
  SDRAM Bus < —> SPARC Bus
  SDRAM Bus <—> SDRAM Bus
  SPARC Bus <—> SPARC Bus
- Burst transfers supported
- Transfer number burst unit (32 bytes)/ maximum setting 64k times (2Mbytes)

9.2. Related Pin Overview

9.3. Register Overview

Table 9-1 SDRAM-IF Registers

ASI	Address	Symbol	Reset Value			
			b31 b24	b33 b16	b15 b8	b7 b0
0x04	0x80000C00	DMA Control Register0	DMCR0	xxxxxxxx	xxxxxxxx	xxxxxxxx
0x04	0x80000C08	DMA Source Address Register0	DMSAR0	xxxxxxxx	xxxxxxxx	xxxxxxxx
0x04	0x80000C10	DMA Destination Address Register0	DMDAR0	xxxxxxxx	xxxxxxxx	xxxxxxxx
0x04	0x80000C18	DMA Word Length Register 0	DMWL0	xxxxxxxx	xxxxxxxx	xxxxxxxx
0x04	0x80000C20	DMA Control Register1	DMCR1[4]	xxxxxxxx	xxxxxxxx	xxxxxxxx
0x04	0x80000C28	DMA Source Address Register1	DMSAR1[4]	xxxxxxxx	xxxxxxxx	xxxxxxxx
0x04	0x80000C20	DMA Destination Address Register1	DMDAR1[4]	xxxxxxxx	xxxxxxxx	xxxxxxxx
0x04	0x80000C28	DMA Word Length Register 1	DMWL1[4]	xxxxxxxx	xxxxxxxx	xxxxxxxx

9.4. Register Details

9.4.1. DMA Control Registers (DCR0-1)

The DMA Control Registers perform all DMA operating mode settings and start/stop control. Whether a source address range is an SDRAM or a SPARClite bus is specified by the SBSn bit. The SIOn bit indicates that a source address range is I/O. When the SIOn bit is set to “1” (when the source address is an I/O area), the address is fixed and is not incremented. The SCSn bit specifies in what CS range the source address is. The SBWn bit specifies the source data bus width. The DBSn bit specifies whether a destination address range is SDRAM or SPARClite. The DCSn bit specifies in which CS range a destination address lies. The

[4] Translator’s Note: In the Japanese original these four symbols are followed by 0 instead of 1, which would seem to be an obvious error.

DBWn bit specifies the destinatioin data bus width. The TCn bit controls EOP signals. When this signal is set to “1”, an EOP signal is output at the end of a transfer. The SBn bit controls DMA operations. If this bit is set to “1”, DMA operation starts. If the SB bit in the same channel is set while a DMA operation is in progress, it has no effect on the DMA operation.

Also, if the SB bit in the other channel is set during a one-way DMA operation, a DMA transfer takes place in the other direction after the DMA operation in progress ends. The SIOn, SCSn and SDSn bits become valid when the SBSn bit is set. Likewise, the DCSNn and the DDSn bits become valid when the DBSn bit is set. If the SB0 bit is cleared to “0” during a DMA transfer, the DMA aborts and sets the ABTn bit. If a parity error or MEXC is detected during a DMA transfer, DMA stops and sets the ERRn bit. All DMACR bit settings and activations should be done at the same time.

31	30	29	28 26	25 24	23	22	21	20 18	17 16	15 9	8	7	6	5 1	0
SBS0	R	SIO0	SCSN0	SBW0	DBS 0	R	DIO0	DCSN0	DBW0	reserved	TC0	ERR0	ABT0	reserved	SB 0

Symbol : DMCR0-1
Address : 0x80000130-0x80000138 (ASI=0x4)
Reset State : 0x00000000

DMCR0 (Address 0x80000C00, ASI=0x4) • • • • channel 0
DMCR1 (Address 0x80000C20, ASI=0x4) • • • • channel 1

bit	Field Name	I/O	Description
31	SBS	R/W	- Source bus select
30	Reserved		
29	SIO	R/W	- Source I/O select-
28-26	SCSN	R/W	- Source CS Number
25-24	SBW	R/W	- Source Bus Width
23	DBS	R/W	- Destination Bus Select
22	Reserved		
21	DIO	R/W	- Destination I/O select
20-18	DCSN	R/W	- Destination CS Number
17-16	DBW	R/W	- Destination Bus Width
15-9	Reserved		
8	TC	R/W	- Transfer complete out
7	ERR	R/W	- Error
5-1	Reserved		
0	SB	R/W	- Start Bit

Figure 9-1 DMCR0 Register

9.4.2. DMA Source Address Registers

DMA Source Address Registers are used to specify DMA transfer source addresses. These registers are assigned to 0x80000c08 (DMSA0 for channel0) and 0x80000c28 (DMSA1 for channel1). The setting unit is burst length (32 bytes). Transfer source ASI is specified by the lower order 4 bits.

31 5	4	3 0
SA	R	ASI

Address : 0x80000C08 (ASI=0x4) • • • DMSAR0

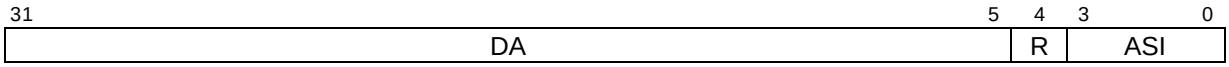
Address : 0x80000C28 (ASI=0x4) • • • DMSAR1
Reset State : 0x00000000

bit	Field Name	I/O	Description
31-5	SA	R/W	- Source Address Sets the upper order 27 bits of the transfer source address.
4	Reserved	R	-
3-0	ASI	R/W	- ASI Sets ASI of transfer source area.

Figure 9-2 DMSAR Register

9.4.3. DMA Destination Address Registers

DMA Destination Address Registers are used to specify destination addresses of DMA transfers. These registers are assigned to 0x80000c10 (DMDA0 for channel0) and 0x80000c30 (DMDA1 for channel1). The setting unit is burst length (32 bytes). Transfer destination ASI are specified by the lower order 4 bits.



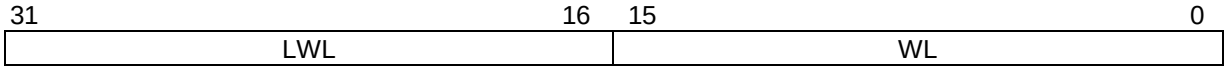
Address : 0x80000C10 (ASI=0x4) • • • DMDAR0
Address : 0x80000C30 (ASI=0x4) • • • DMDAR1
Reset State : 0x00000000

bit	Field Name	I/O	Description
31-5	DA	R/W	- Destination Address Sets the upper order 27 bits of the transfer destination address.
4	Reserved	R	-
3-0	ASI	R/W	- ASI Sets ASI of transfer destination area.

Figure 9-3 DMDAR Registers

9.4.4. DMA Word Length Registers

DMA Word Length Registers are for specifying the number of DMA transfer data. They are assigned to 0x80000c18 (DMWL0 for channel0) and 0x8000c38 (DMWL1 for channel1). The setting unit is burst length (32 bytes).



Address : 0x80000C18 (ASI=0x4) • • • DMWL0
Address : 0x80000C38 (ASI=0x4) • • • DMWL1
Reset State : 0x00000000

bit	Field Name	R/W	Description
31-16	LWL	R/W	- Left Word Length Displays the remaining number of transfers during DMA transfers or when exceptions occur
15-0	WL	R/W	- Word Length Sets number of transfer data. Sets transfer numbers in units of 32 bytes.

Figure 9-4 DMWL Registers

9.5. Operating Functions

9.5.1. Internal Bus Order of Priority

The DMAC is connected to the internal chip BIU bus. The CPU Core has priority for the BIU bus right. DMA transfers are activated by setting activation bit SB of the DMA Control Registers (DMCR0, 1) to 1. If a CPU Core access request occurs in a DMA operation, DMA opens a bus right in burst transfer units and, once the CPU has opened a bus right, continues to operate.

9.5.2. Transfer System

The DMAC has a 64-bit 4-column FIFO, and its basic bus cycle is a burst transfer of transfer length 4 (bus width 8 bytes X 4 times = 32 bytes). It reads 4 transfer data sequentially in burst length from the memory spaces indicating source addresses and outputs the 4 data sequentially in burst length to the memory spaces indicating destination addresses. Moreover, transfer number settings, bus arbitration, error processing and the like are basically in units of 32 bytes. DMA transfers basically assume access to non-cache spaces. Since snoop functions are not supported by the SS200, do not make cache area DMA transfers. Memory coherency during transfers is not guaranteed.

9.5.3. Register Summary

Bus selection, I/O area selection and CS area selection are set by the MCR Register for source and destination spaces. I/O areas and CS areas are valid only when bus selection has been specified to the SPB. CS area settings must be made in conjunction with ARSR and AMR Register settings. Addresses are not updated when settings are made to I/O areas.

Source and destination addresses are set respectively by the DMSAR and DMDAR registers. Address settings are in 32-byte units. ASI is specified by the lower order 4 bits of a register. Transfer number is set by the DMWL Register. Transfer number is set to the lower order 16 bits(WL) of the register in 32-byte (1 burst) units. WL set to 1 indicates transfers of 1 burst (32 bytes), and transfers of up to 64k bursts at a maximum of 0xffff are possible. The upper order 16 bits (LWL) of DMEWL indicate the remaining number of transfers during DMA transfers or when exceptions occur.

9.5.4. Activation/Termination

All DMA set registers can independently set 2 channels. When the start bit of the MCR register of each channel is set, DMA is activated. Following execution of the of the number of transfers set to the Transfer Number Register, it automatically clears the start bit of the control register and stops. DMA transfers operate only in 1 direction in 1 channel at a time. If the start bit of the other channel is set while a channel is activated in one direction, the channel in the other direction starts transfers after the set number of transfers has been completed in the first channel. EOP is output when the transfers end. Whether or not to output it can be specified by register settings.

9.5.5. Exception Processing

- Aborts
  If the activation bit of the MCR Register is cleared (1 —> 0) during a DMA transfer, DMA transfers in that channel are aborted. Abort timing is timing in which an MCR Register clear write cycle is inserted at a break in the burst unit bus.

  Following a stop because of an abort, the MCR abort bit (bit6) is set, but EOP is not output. When the activation bit in the other channel is set, if the channel in the one direction stops, transfers are automatically started. After the first channel is aborted, the activation bits of both channels must be cleared to keep the other channel from activating. The abort bit should be cleared by an MCR write.

- Errors
  If DMA detects a parity error or MEXC while reading source data, it stops writing to destination address spaces, clears the MCR activation bit in both channels and stops. While stopped, it outputs EOP and sets the ERR bit (bit7) of the MCR. The ERR bit is not cleared by MCR writes. It is automatically cleared upon reactivation of the channel.

9.5.6. Block Diagram

Figure 9-5 MB8686x DMA Block Diagram