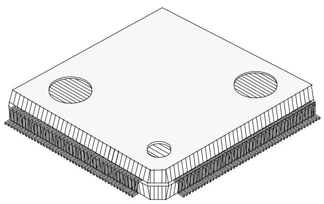
ASSP CMOS

SPARClite Series 32-Bit RISC Embedded Processor

MB86832



Package

- 176-pin, Plastic SQFP
- FPT-176P-M01

Features

- 66, 80, or 100 MHz CPU with on-chip clock multiplier
- SPARC high performance RISC architecture
- 8 window, 136 word register file
- 16 address spaces, 256 Mbyte each
- Harvard-style separate on-chip instruction and data buses
- 8 Kbyte 2-way set-associative instruction cache
- 8 Kbyte 2-way set-associative data cache
- Flexible locking mechanism for data and instruction cache entries
- Option to force non-cached operation for memory areas selected by the programmable chip selects; also qualification on a cycle-by-cycle basis using Non-Cache Pin
- Four-level buffered writes and one-level instruction pre-fetching
- CPU and core logic up to 5 times the frequency of the bus interface unit using on-chip clock multiplier (BIU frequency–40 MHz for the 100 MHz version and 33 MHz for the 66 MHz and 80 MHz versions)
- Bus interface support for 8-, 16-, or 32-bit wide memory
- Support for burst mode cache fills
- DRAM controller with fast page or burst-mode EDO DRAM support
- Interrupt controller with fast response time and programmable priority
- Burst mode ROM support
- Power-saving sleep mode with clock gear function
- Programmable address decoder and wait-state generator
- Single vector trapping
- Debug Support Unit
- 0.35 micron gate, 2-level metal CMOS technology, 3.3V internal with 3.3 or 5V I/O

Table of Contents

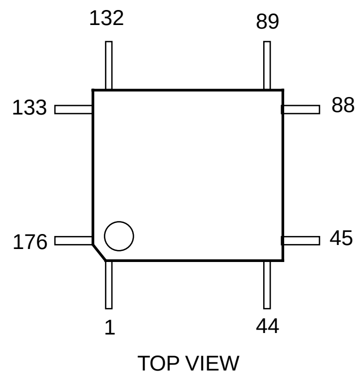
General Discussion

The MB86832 is a member of the MB8683X series of RISC processors which offers high performance and high integration for a wide range of embedded applications. The processor is based on the SPARC architecture and is upward code-compatible with previous implementations. At 66, 80, and 100 MHz, the processor runs at up to 66, 80, and 100 MIPS, respectively.

The MB86832 is housed in a low-profile 176-pin plastic package. It is upward pin-compatible from the MB86831. The on-chip data and instruction caches help decouple the processor from external memory latency. Separate on-chip instruction and data paths provide a high-bandwidth interface between the integer unit (IU) and caches.

For maximum performance with a minimum of glue logic, the MB86832 includes: programmable chip select outputs and wait state generation, built-in support for page-mode DRAM, EDO DRAM, and page-mode EEPROM, and support for 8- and 16- and 32-bit memory. These features combine to give the MB86832 high integration and high performance, with the flexibility and efficiency to make it the ideal choice for a wide variety of low cost, embedded systems.

Pin Configuration



Block Diagram of MB86832

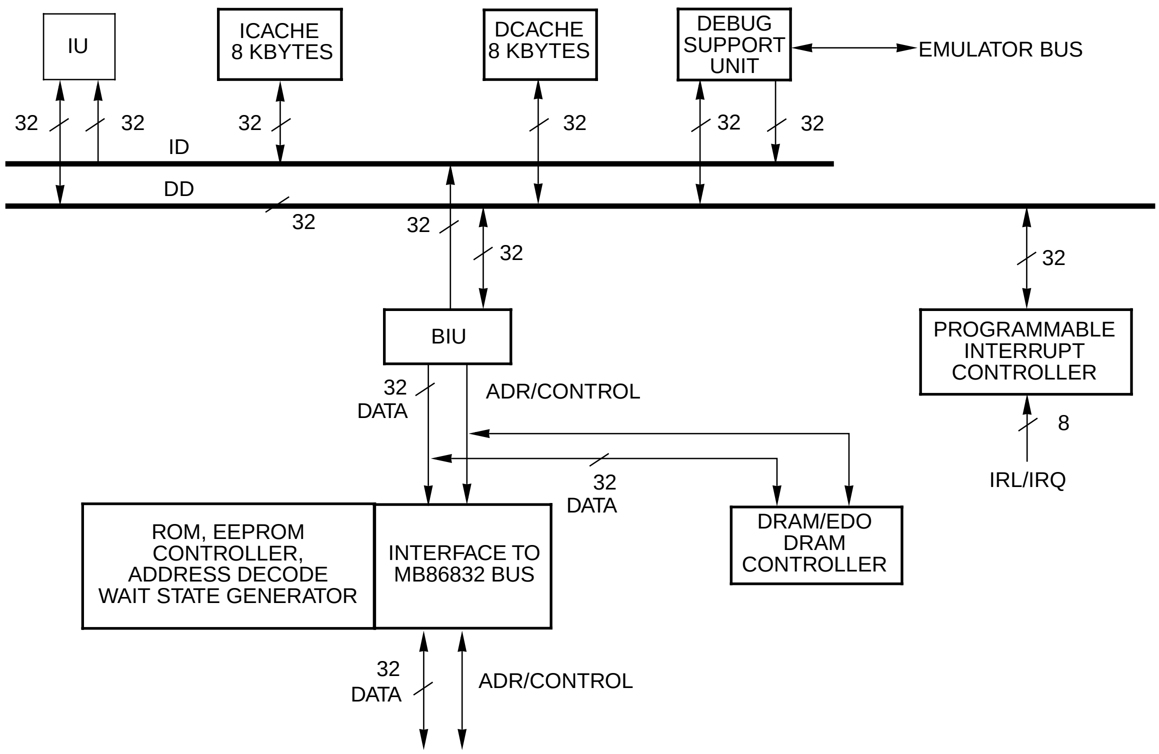


Table 1. Ordering Code

Clock Frequency (MHz)	Ordering Code	Package Type
66	MB86832-66PFV-G	Plastic SQFP 176
80	MB86832-80PFV-G	Plastic SQFP 176
100	MB86832-100PFV-G	Plastic SQFP 176

Note: The ordering code for production level product. Early shipments of this device may be marked with ”ES” to indicate that the part is not yet at full production status.

Table 2. Pin Assignment – 176-pin SQFP

Pin No.	I/O	Pin Name	Pin No.	I/O	Pin Name	Pin No.	I/O	Pin Name	Pin No.	I/O	Pin Name
1		VDD	45		VSS	89		VDD	133		VSS
2	I/O	D31	46	O	DWE3	90	O	BE3 / ADR0	134	I/O	ASI3 / ADR28
3	I/O	D30	47	O	DWE2	91	I/O	BE2 / ADR1	135	I/O	ASI2 / ADR29
4	I/O	D29	48	O	DWE1	92	O	BE1	136	I/O	ASI1 / ADR30
5	I/O	D28	49	O	DWE0	93	O	BE0	137	I/O	ASI0 / ADR31
6		VSS	50		VSS	94		VSS	138		VSS
7	I	BMODE16	51		IO_VDD	95	I	NONCACHE	139		IO_VDD
8	I/O	D27	52	O	RAS0	96	N/A	N.C.	140	I	IRL3 / IRQ15
9	I/O	D26	53	O	RAS1	97	N/A	N.C.	141	I	IRL2 / IRQ14
10	I/O	D25	54	O	RAS2	98	I/O	ADR2	142	I	IRL1 / IRQ13
11	I/O	D24	55	O	RAS3	99	I/O	ADR3	143	I	IRL0 / IRQ12
12		IO_VDD	56		VDD	100		IO_VDD	144		VDD
13	I/O	D23	57	O	CAS0	101	I/O	ADR4	145	I	FLOAT
14	I/O	D22	58	O	CAS1	102	I/O	ADR5	146	O	PDOWN
15	I/O	D21	59	O	CAS2	103	I/O	ADR6	147	I	WKUP
16	I/O	D20	60	O	CAS3	104	I/O	ADR7	148	I	RESET
17		VSS	61		VSS	105		VSS	149		VSS
18	I/O	D19	62	O	DOE	106	I/O	ADR8	150	I	IDLEEN
19	I/O	D18	63	I	CLKSEL2 *	107	I/O	ADR9	151	I	CLKSEL1
20	I/O	D17	64	O	ERROR	108	I/O	ADR10	152	I	CLKSEL0
21	I/O	D16	65	O	LOCK	109	I/O	ADR11	153	I	CLKEXT
22	I	BTEST	66	I	CTEST	110	I	READY	154	I	CLKIN
23		VDD	67		IO_VDD	111		VDD	155		IO_VDD
24	I/O	D15	68	I	BREQ	112	I/O	ADR12	156	I	IRQ11
25	I/O	D14	69	O	PBREQ	113	I/O	ADR13	157	I	IRQ10
26	I/O	D13	70	O	BGRNT	114	I/O	ADR14	158	I	IRQ9
27	I/O	D12	71	I	BMACK	115	I/O	ADR15	159	I	IRQ8
28		VSS	72		VSS	116		VSS	160		VSS
29	I/O	D11	73	O	BMREQ	117	I/O	ADR16	161	N/A	N.C.
30	I/O	D10	74	O	TIMER_OVF	118	I/O	ADR17	162	I	ASISEL *
31	I/O	D9	75	O	SAMEPAGE	119	I/O	ADR18	163	I	EMU_BRK *
32	I/O	D8	76	I/O	AS	120	I/O	ADR19	164	I/O	EMU_ENB *
33		IO_VDD	77		VDD	121		IO_VDD	165		VDD
34	I/O	D7	78	I/O	RDWR	122	I/O	ADR20	166	I/O	EMU_SD3
35	I/O	D6	79	O	RDYOUT	123	I/O	ADR21	167	I/O	EMU_SD2
36	I/O	D5	80	O	CS5	124	I/O	ADR22	168	I/O	EMU_SD1
37	I/O	D4	81	O	CS4	125	I/O	ADR23	169	I/O	EMU_SD0
38	I	BMODE8	82		IO_VDD	126	I	MEXC	170		IO_VDD
39		VSS	83		VSS	127		VSS	171		VSS
40	I/O	D3	84	O	CS3	128	I/O	ADR24	172	I/O	EMU_D3
41	I/O	D2	85	O	CS2	129	I/O	ADR25	173	I/O	EMU_D2
42	I/O	D1	86	O	CS1	130	I/O	ADR26	174	I/O	EMU_D1
43	I/O	D0	87	O	CS0	131	I/O	ADR27	175	I/O	EMU_D0
44		VDD	88		VSS	132		VDD	176		VSS

* These inputs have a 50K Ω internal pullup resistor.

Table 3. Signal Descriptions

<table>
<tr><th>Symbol</th><th>Type</th><th>Description</th></tr>
<tr><td>CLKIN</td><td>I</td><td>CLOCK: The clock input pin. The clock is the timebase for the operation of the bus interface unit (BIU). An on-chip clock multiplier allows the CPU and core logic to run at integer multiples of the clock frequency (×1, ×2, ×3, ×4, or ×5).</td></tr>
<tr><td>CLKEXT</td><td>I</td><td>EXTERNAL CLOCK BYPASS: The external clock selection pin. If tied low, the clock is generated by the internal PLL/clock multiplier. If tied high, the external clock (i.e., the signal on CLKIN) is used without modification.</td></tr>
<tr><td>$\overline{RESET}$</td><td>I</td><td>SYSTEM RESET: The reset input. The CPU and core logic are initialized by pulsing this input low. The clock must be stable for 100 ms before the reset pulse is de-asserted. The reset pulse must be a minimum of four CLKIN cycles in length. The CPU begins execution at location 0 three CLKIN cycles after the reset pulse is de-asserted.</td></tr>
<tr><td>CLKSEL0
CLKSEL1
CLKSEL2</td><td>I</td><td>INTERNAL CLOCK SELECT: These pins select the clock frequency multiplier, as described in the table below.
<table>
<tr><th>CLKSEL2</th><th>CLKSEL1</th><th>CLKSEL0</th><th>Internal Clock</th></tr>
<tr><td>H</td><td>L</td><td>L</td><td>x1</td></tr>
<tr><td>H</td><td>L</td><td>H</td><td>x2</td></tr>
<tr><td>H</td><td>H</td><td>L</td><td>x3</td></tr>
<tr><td>H</td><td>H</td><td>H</td><td>x4</td></tr>
<tr><td>L</td><td>H</td><td>H</td><td>x5</td></tr>
</table>
Note: CLKSEL0 and CLKSEL1 do not have internal pullup resistors, so they must be tied high or low. CLKSEL2 has an internal 50K Ω pullup resistor.</td></tr>
<tr><td>$\overline{CTEST}$
$\overline{BTEST}$</td><td>I</td><td>CTEST BTEST: Test pins. Must be tied high.</td></tr>
<tr><td>ADR<27:2></td><td>I/O</td><td>ADDRESS BUS: The 26-bit address bus ADR<27:2> references a 32-bit word. ADR1 and ADR0 are generated for 8- and 16-bit bus width transactions and are driven on the byte enables $\overline{BE2}$ and $\overline{BE3}$. The address is not valid during idle cycles. During bus grant mode, the address bus becomes an input, and it is used by the $\overline{CS}$ generator circuit and on-chip core logic. When ADR<27:2> is driven in this mode, ADR<31:28> are treated as 0 internally. If the DRAM controller is enabled, it multiplexes row and column addresses and drives them on ADR<13:2>.</td></tr>
<tr><td>ASI<3:0>/
ADR<28:31></td><td>I/O</td><td>ADDRESS SPACE IDENTIFIER: The address space identifier (ASI) selects one of 16 separate address spaces referenced by the address bus. These spaces distinguish between user and supervisor space, instruction and data space, memory and peripheral control registers, and other addressable areas. When ASISEL is high these pins are ASI<3:0>, while when ASISEL is low these pins are ADR<28:31>. The timing is identical to ADR<27:2>. The ASI signals become inputs during bus grant mode when ASISEL is taken low. This is used for CS generation and internal address decoding. In this mode, ASI<7:4> is treated as 0 internally by the CPU and core logic.</td></tr>
<tr><td>ASISEL</td><td>I</td><td>ASI SELECT SIGNAL: In bus grant mode, used to select whether the signals on pins 112 through 115 drive ASI<3:0> or ADR<24:27>.
This pin has a pullup resistor.
<table>
<tr><th>ASISEL</th><th>ASI<3:0> / ADR <28:31></th></tr>
<tr><td>L</td><td>ADR <28:31></td></tr>
<tr><td>H</td><td>ASI <3:0></td></tr>
</table></td></tr>
<tr><td>$\overline{AS}$</td><td>I/O</td><td>ADDRESS STROBE: A one-cycle low pulse is driven on $\overline{AS}$ during the first clock of the bus cycle. The bus cycle starts with assertion of $\overline{AS}$ and ends with assertion of $\overline{READY}$ or $\overline{RDYOUT}$. The $\overline{AS}$ signal is an input during bus grant mode and is used as an activation signal for CS generator circuits and wait-state generator circuits.</td></tr>
<tr><td>$\overline{CS0}$
$\overline{CS1}$
$\overline{CS2}$
$\overline{CS3}$
$\overline{CS4}$
$\overline{CS5}$</td><td>O</td><td>CHIP SELECT: The chip select signals are asserted if the address ranges indicated in the Address Range Specifier Register (ARSR) and the Address Mask Register (AMR) are referenced with the System Support Control Register (SSCR) CS enable bit (bit 4) set. An exception is $\overline{CS0}$ (i.e. the boot ROM chip select), which has no Address Range Specifier Register and is always enabled. Each address range has a corresponding wait specifier which is used to automatically assert the $\overline{READY}$ signal after a user defined number of processor clocks. This allows a variety of memory and I/O devices with different access times to be connected to the processor without the need for additional logic.</td></tr>
<tr><td>D<31:0></td><td>I/O</td><td>DATA BUS: This is the 32-bit data bus. It is a bidirectional data bus used for instruction fetch, data reads, and data writes. Instruction and word data must be aligned to 32-bit boundaries, and half words must be aligned to even addresses. Double words must be aligned to addresses which are multiples of 8. In 8-bit bus mode D<7:0> is used, and in 16-bit bus mode D<15:0> is used.</td></tr>
</table>

Table 3. Signal Descriptions (Continued)

Symbol	Type	Description
BE0	O	**BYTE ENABLE:** Indicates the valid byte during a write when 32-bit bus width is used. During reads, BE<3:0> are asserted without regard to the bus width. When 8-bit bus width is used, BE2 and BE3 are driven with ADR1 and ADR0, respectively. When 16-bit bus width is used, BE2 is driven with ADR1. BE<3:0> are valid during the bus cycle period. During idle cycles, the output is undefined. During bus grant, the byte enables are high impedance, the DRAM controller may be enabled, and BE2 becomes the ADR1 input if 16-bit bus width is used.
BE1	O	
BE2/ADR1	I/O	
BE3/ADR0	O	

Bus Width	Access Type		BE0,1,2,3
32-bit	Write	Byte 0 (D<31:24>) *1	0111
		Byte 1 (D<23:16>)	1011
		Byte 2 (D<15:8>)	1101
		Byte 3 (D<7:0>)	1110
		Half Word 0 (D<31:16>)	0011
		Half Word 1 (D<15:0>)	1100
		Word	0000
	Read	All data types	0000
16-bit	Write	Byte 0 (D<15:8>)	1000
		Byte 1 (D<7:0>)	0100
		Byte 2 (D<15:8>)	1010
		Byte 3 (D<7:0>)	0110
		Half Word 0 (D<15:0>)	0000
		Half Word 1 (D<15:0>)	0010
		Word (D<15:0>) Access 0	0010
		Word (D<15:0>) Access 1	0000
	Read	Access 0	0000
		Access 1	0010
8-bit	Write	Byte 0	XX00
		Byte 1	XX01
		Byte 2	XX10
		Byte 3	XX11
		Half Word 0 Access 0	XX11
		Half Word 0 Access 1	XX10
		Half Word 1 Access 0	XX01
		Half Word 1 Access 1	XX00
		Word Access 0	XX11
		Word Access 1	XX10
		Word Access 2	XX01
		Word Access 3	XX00
	Read	Access 0	XX00
		Access 1	XX01
		Access 2	XX10
		Access 3	XX11

*Notation such as (D<31:24>) shows the data bus bits being used.

Symbol	Type	Description
RDWR	I/O	**READ/WRITE BUS TRANSACTION:** This signal is low during write cycles and high during read cycles and idle cycles. It is an input during bus grant mode, and it is used for generating DWE0 and DOE when the DRAM controller is enabled. This signal is not used in bus grant mode when the DRAM controller is disabled.
READY	I	**EXTERNAL READY:** This is a control signal asserted by the external memory system to indicate that the current bus transaction is being completed and that it is ready to start with the next bus transaction in the following cycle. In case of a fetch from memory, the processor will strobe the value on the data bus at the rising edge of CLK_IN following the assertion of READY. For the case of a write, the memory system will assert READY when the appropriate access time has been met. In most cases, no additional logic is required to generate the READY signal. On-chip circuitry can be programmed to assert READY based on the address of the current transaction. The external system can override the internal ready generator to terminate the current bus cycle early. READY must be pulsed an appropriate number of times during a burst transfer or when multiple bus cycles are required to transfer data with 8- or 16-bit bus width.

Table 3. Signal Descriptions (Continued)

Symbol	Type	Description
RDYOUT	O	**READY OUTPUT:** Assertions of the READY signal generated by any source, including the internal wait state generator, are visible to external devices on this pin. Internally generated READY assertions are synchronized to the clock. Externally generated READY assertions will appear on this signal with a small amount of propagation delay.
IRQ15/IRL3 IRQ14/IRL2 IRQ13/IRL1 IRQ12/IRL0 IRQ11 IRQ10 IRQ9 IRQ8	I	**INTERRUPT REQUEST/INTERRUPT REQUEST LEVEL:** These are interrupt inputs. Four of the inputs have a dual function. When the interrupt controller (IRC) is disabled (i.e. bits 0 and 1 in the IRC Mode Register are clear), IRL<3:0> is the encoded priority level of external interrupt requests, which compete with the on-chip interrupt sources for service by the CPU. Typically, IRL<3:0> would be generated by an external interrupt controller. Higher values have greater priority. IRL=$0000_{(2)}$ indicates no interrupt requests are pending, and IRL=$1111_{(2)}$ is defined by the SPARC architecture as a non-maskable interrupt. The external interrupt requests are sampled on two successive CLKIN clock periods to prevent false interrupts. When the interrupt controller is enabled, these pins are unencoded interrupt requests IRQ<15:8>. Each interrupt request can be programmed to be triggered on a high level, low level, rising edge, or falling edge. When an interrupt signal meets the qualifications to invoke an interrupt, an interrupt request is loaded in the IRC Request Sense Register. The IRC performs priority resolution and encoding to generate IRL<3:0>, which is passed to the CPU core.
BREQ	I	**BUS REQUEST:** When this signal is asserted by an external bus master, the CPU releases control of the bus after the current bus operation is completed. Certain operations require more than one bus cycle: (1) When an atomic load-store instruction is executed, the bus is released upon completion of both the load and the store. (2) In the case of a double word load or store, the bus is released if BREQ is asserted during the transfer of word 1 after the first word has been transferred, and if BREQ is asserted during the transfer of word 2, the bus is released after transferring the second word. (3) Store in 8- and 16-bit bus widths: The bus is released after transmission of the entire data object (for example, when a 32-bit word is transferred with 8-bit bus size, four 8-bit bus cycles occur before the bus is released). (4) Load in 8 and 16-bit bus width: The bus is released after the entire word has been transferred.
BGRNT	O	**BUS GRANT:** This signal is asserted following a bus request (i.e. BREQ assertion) to indicate that control of the bus has been released to an external device.
PBREQ	O	**PROCESSOR BUS REQUEST:** This signal is asserted by the processor to indicate to an external bus arbiter that the CPU wants to regain control of the bus. This provides a handshake between the arbiter and the processor to allow the bus to be allocated based on demand.
LOCK	O	**BUS LOCK:** The CPU asserts this signal during execution of an atomic load-store. It indicates that the current bus transaction requires more than one bus cycle which cannot be split by releasing the bus to another bus master.
MEXC	I	**MEMORY EXCEPTION:** If this signal is low on the same clock that READY is asserted, the bus cycle is handled like a page fault, i.e. an instruction or data access exception is invoked. This signal must not be asserted except on the same clock as READY. If a memory exception is signalled when the ET bit of the PSR is clear (i.e. traps are disabled), the CPU enters error mode.
ERROR	O	**ERROR SIGNAL:** This signal indicates that a trap has occurred while traps were disabled. When this happens, the CPU saves the PC and nPC to a register, loads the trap type in the TBR, and goes into error mode. Error mode can only be exited by a reset.
IDLEEN	I	**IDLE ENABLE:** When this pin is high and the previous cycle was loaded or stored to the CS0 area, the next cycle is started after insertion of a two-clock idle cycle. This is intended to accommodate EPROM boot memory with slow output disable, saving the addition of a buffer chip. When this pin is low, and a write cycle immediately follows a read cycle, an address cycle is inserted for one clock only (compatible with former versions of SPARClite).
BMODE8 BMODE16	I	**BOOT MODE8 and BOOT MODE16:** These signals select the bus width of the CS0 area. They are sampled during reset initialization. When BMODE8 is low, 8-bit bus width is selected, and when BMODE16 is low, 16-bit bus width is selected. When both pins are tied high, the bus width is 32 bits. The bus width of the CS1 through CS5 areas are not affected by these signals. The CS1 through CS5 areas are programmed through the Bus Width/Cacheable Control Register BWCR. Do not tie both of these inputs low.
NONCACHE	I	**NON-CACHEABLE:** This signal is asserted by external logic to indicate the data on the bus is non-cacheable. Low indicates non-cacheable and high indicates cacheable. This signal is used when the CBIR (Cache/BIU Control Register) Cacheability Enable bit (bit 7) is set. Normally, the NONCACHE signal is driven on a clock in which address strobe is asserted. However, if NONCACHE must be one clock or more late, it can be used by setting the Cache/BIU Control Register (CBIR) Noncacheable Wait State bits (bits 9,8). This signal is ignored during instruction fetches and when the internal cacheability is used.
PDOWN	O	**POWER DOWN:** This signal indicates transition to sleep mode (i.e. low power consumption mode) when it is low.
WKUP	I	**WAKE-UP:** This pin is driven low to break the CPU out of sleep mode. This pin is an asynchronous input, and must be driven with a pulse of 2 or more CLKIN periods. This pin must only be driven low when PDOWN is low, otherwise the behavior of the processor is undefined.
FLOAT	I	**FLOAT:** Driving this input low causes all output pins and bidirectional pins to go into high-impedance mode.

Table 3. Signal Descriptions (Continued)

Symbol	Type	Description
BMREQ	O	**BURST MODE REQUEST:** This signal is asserted by the processor to indicate to the external system that the processor's burst mode is enabled in the Bus Control Register (BCR) and the current transaction can be a burst. If the external system supports burst mode, BMACK can be asserted concurrently with READY to begin the burst mode transfer. BMREQ is asserted even when the DRAM burst enable bit in the System Support Control Register (SSCR) is set. However, in this case the internal DRAM controller drives the signal, so it is not necessary for external logic to drive BMREQ.
BMACK	I	**BURST MODE ACKNOWLEDGE:** This signal is asserted by external logic to indicate that it can support burst mode for the address currently on the bus. If driven low on or before the clock on which READY is asserted, burst mode data transfer is used. The signal can be driven low in the same clock as READY, or a clock before that and sustained until READY is driven. If the DRAM burst enable bit of the System Support Control Register (SSCR) is set, burst mode is used even if BMACK is not asserted.
SAMEPAGE	O	**SAME PAGE DETECT:** When bit 5 of the System Support Control Register (SSCR) is set, RAS page-hit detection is enabled. The address in the Same Page Master Register (SPSMR) and the previously accessed address are compared, and if they match SAMEPAGE is asserted. SAMEPAGE is never asserted in the first bus cycle following a transfer of bus control. The page size is specified in the Same Page Mask Register.
RAS0 RAS1 RAS2 RAS3	O	**DRAM ROW ADDRESS STROBE:** These are the row-address strobes from the DRAM controller.
CAS0 CAS1 CAS2 CAS3	O	**DRAM COLUMN ADDRESS STROBE:** These are the column address strobes from the DRAM controller. When 32-bit bus width is selected and 2-CAS DRAM configuration is used, CAS<0:3> correspond to byte 0 (b31-b24), byte 1 (b23-b16), byte 2 (b15-b8) and byte 3 (b7-b0). With 16-bit bus width and 2-CAS DRAM, CAS2 and CAS3 correspond to byte 0 (even byte address) and byte 1 (odd byte address), respectively. CAS0 and CAS1 are undefined when 16-bit bus width is selected. When 2-WE DRAM is used, CAS<0:3> are driven with identical signals.
DWE0 DWE1 DWE2 DWE3	O	**DRAM WRITE ENABLE:** These are the DRAM write enables. When a 2-WE DRAM configuration is used, DWE<0:3> correspond respectively to byte 0 (b31-b24), byte 1 (b23-b16), byte 2 (b15-b8) and byte 3 (b7-b0). When a 2-CAS DRAM is used, DWE<0:3> are driven with identical signals.
DOE	O	**DRAM OUTPUT ENABLE:** This is the output enable from the DRAM controller. DRAM interface is possible without using this signal for control of DWE0 and CAS<3:0> in early-write timing when a page-mode DRAM is used, but it is required for controlling the DRAM output drivers when EDO (hyper page-mode) is used.
TIMER_OVF	O	**TIMER OVERFLOW:** When the DRAM refresh timer is enabled in the System Support Control Register (SSCR), transparent DRAM refresh can be set up. The period between refresh cycles is programmed in the DRAM Refresh Timer Preload Register, and the signal pulse widths are programmed in the DRAM Refresh Timer Register. TIMER_OVF is pulsed low when the timer counts down to zero. The timer is clocked by CLKIN, without frequency multiplication. Bit 31 of the DRAM Refresh Timer Preload Register controls the pulse width, either one clock or three clocks long. When a three-clock pulse width is selected, TIMER_OVF can be connected to an interrupt input (IRQx) of the interrupt controller.
EMU_SD <3:0>	I/O	**EMULATOR STATUS/DATA BITS:** Bidirectional pins used by a hardware emulator to control and monitor processor execution. These pins should be left unconnected.
EMU_D <3:0>	I/O	**EMULATOR DATA BITS:** Bidirectional pins used by a hardware emulator to control and monitor processor execution. These pins should be left unconnected.
EMU_BRK	I	**EMULATOR BREAK REQUEST LINE:** Input used by a hardware emulator to request a trap when emulation is enabled. This pin should be left unconnected.
EMU_ENB	I/O	**EMULATOR ENABLE:** Tied low while the processor is being reset to enable hardware emulator mode on the chip. This pin should be left unconnected.

Note: In the signal descriptions, names with an overbar indicate active low assertion. Dual function pins have two names separated by a slash (/).

Table 4. Pin Status Description

Pin	During Reset	During Bus Grant
ADR <27:2>	O(X)	I(D)
$\overline{\text{AS}}$	O(H)	I(Z)
$\overline{\text{BE0}}$	O(X)	O(Z)
$\overline{\text{BE1}}$	O(X)	O(Z)
$\overline{\text{BE2/ADR1}}$	O(X)	O(Z), I(Z)
$\overline{\text{BE3/ADR0}}$	O(X)	O(Z)
$\overline{\text{CS<5:0>}}$	O(H)	I(Z), O(Z)
$\overline{\text{ERROR}}$	O(H)	O(V)
$\overline{\text{LOCK}}$	O(H)	O(Z)
$\overline{\text{PDOWN}}$	O(H)	O(H)
$\overline{\text{PBREQ}}$	O(H)	O(V)
$\overline{\text{SAMEPAGE}}$	O(H)	O(V)
D<31:0>	I(Z)	I(Z)
RDWR	O(H)	I(Z)
$\overline{\text{BGRNT}}$	O(H)	O(L)
ASI<3:0>	O(X)	I(Z)
$\overline{\text{RDYOUT}}$	O(V)	O(V)
$\overline{\text{BMREQ}}$	O(H)	O(H)
$\overline{\text{TIMER\_OVF}}$	O(H)	O(V)
$\overline{\text{RAS<3:0>}}$	O(H)	O(V)
$\overline{\text{CAS<3:0>}}$	O(H)	O(V)
$\overline{\text{DOE}}$	O(H)	O(V)

Access Type

O(V) : Output driven to a valid level.

O(X) : Output is undefined.

O(Z) : Output is in high impedance.

O(H) : Output is driven high.

O(L) : Output is driven low.

I(Z) : Input is high impedance.

I(D) : If the DRAM controller is enabled, the address sampled on the assertion of $\overline{\text{AS}}$ will be driven from the next clock until $\overline{\text{READY}}$ is asserted. If the address maps to a DRAM area, a multiplexed address will be driven. If the DRAM controller is disabled, the signal will be in high impedance.

Overview

The Fujitsu MB86832 is a high performance, 32-bit RISC processor with up to 66, 80, or 100 MIPS peak performance at 66, 80, or 100 MHz clock frequency, respectively. Like its predecessors, the MB86832 is based on the SPARC architecture and is upward code compatible with previous implementations. The MB86832 has been developed specifically for the needs of embedded applications that require high performance and high integration.

The MB86832 instruction set was designed for fast execution, with most instructions executing in a single cycle. The Integer Unit (IU) features a 5-stage pipeline which has been designed to handle data interlocks, an optimized branch handler for efficient control transfers, and a bus interface to handle single-cycle bus accesses to on-chip memory.

An internal register file consisting of 136 registers organized into eight overlapping windows provide rapid interrupt response time and context switches. The register file minimizes accesses to memory during procedure linkages and facilitates passing of parameters and assignment of variables.

On-chip 8-Kbyte, 2-way set-associative instruction and 8-Kbyte, 2-way set-associative data caches have been added to decouple the processor from external memory. These caches have been designed for maximum flexibility. For example, they allow cache lines to be locked for faster access to critical data.

Separate 32-bit on-chip instruction and data paths (i.e. Harvard-style architecture) provide a high-bandwidth interface between the IU and on-chip caches. These buses support single-cycle instruction execution as well as single cycle data transfers with the cache.

The MB86832 also includes hardware for integer multiply and divide step. The hardware support significantly improves the performance of these operations, with 32-bit integer multiplies executing in 5 clock cycles, 16-bit integer multiplies in 3 cycles, and 8-bit integer multiplies in 2 cycles.

Key Features

Fast Integer Unit Instruction Execution: Simple operations make up the bulk of instructions in most programs, so execution speed can be greatly improved by designing these instructions to execute in as short a time as possible. In the SPARC architecture, the majority of instructions execute in one cycle with only a few of the more complex, such as integer multiply, taking additional cycles.

Large Register Set: The large register set for the IU reduces the number of required accesses to data memory. The registers are organized in overlapping groups called register windows, which allow registers to be reserved for high priority tasks, such as interrupts, or for frequently called tasks such as operating system working registers. The overlapping windows also simplify parameter passing and reduce instruction overhead for procedure linkage.

On-Chip Caches: Separate 2-way set associative instruction and data caches have been added to IU. This decouples the fast IU from off-chip memory because external memory access is only required on cache misses.

Cache Locking: Both the instruction and data cache lines can be locked to ensure deterministic response and highest performance for critical or frequently called routines. Maximum flexibility has been designed into the cache to allow all or selected portions to be locked.

Bus Interface: The requirement for glue logic between the MB86832 and the system is minimized by providing programmable chip selects, programmable wait state circuitry, programmable cacheable and non-cacheable memory address, and support for connection to fast page-mode DRAM or burst-mode EDO DRAM. Multiple bus masters are supported through a simple handshaking protocol. The MB86832 can boot from either 8-, 16-, or 32-bit wide memory. In addition, the programmable data bus allows reading/writing of different memory widths. For high frequency operation, the core can run at up to 5 times the bus. Note however, that the BIU frequency should not exceed 40 MHz for the 100 MHz version and 33 MHz for the 66 and 80 MHz version.

Clock Generator: An external clock source must be supplied. Unlike some other members of the SPARClite family, there is no on-chip oscillator. A built-in phase-locked loop minimizes the skew between on and off-chip clocks.

Enhanced Instruction Set: The MB86832 includes a fast integer multiply instruction which executes in 5, 3, or 2 cycles for 32-bit, 16-bit, or 8-bit multiplicands, respectively. An integer divide-step instruction cuts divide times by a factor of 10 over previous SPARC implementations. A scan instruction supports a single cycle search for the most significant 1 or 0 in a word or bit differing from sign bit.

Sleep Mode

The MB86832 has a Sleep Mode, i.e. a power-saving mode in which program execution is temporarily suspended. The $\overline{\text{PDOWN}}$ and $\overline{\text{WKUP}}$ pins and the Sleep Mode Register provide a mechanism to enter and exit Sleep Mode.

Debug Support Unit (DSU)

The debug support unit (DSU) supports monitor mode and hardware emulation mode, can set breakpoints, and allows single stepping of the CPU. When the CPU is reset, the DSU can be enabled by driving the $\overline{\text{EMU\_BRK}}$ and $\overline{\text{EMU\_ENB}}$ signals.

Data on the DSU emulator bus can be monitored by external logic, and operations between the IU and the cache can be traced. In-circuit emulators and other debug hardware can determine the status of the CPU by monitoring these emulator busses EMU_SD<3:0> and EMU_D<3:0>. Please refer to the MB86832 User Manual for further details.

Table 5. MB86832 Instruction Set

LOGICAL	ARITHMETIC/SHIFT	DATA MOVEMENT
CONDITION CODES UNCHANGED AND OR XOR AND NOT OR NOT XNOR **CONDITION CODES SET** AND OR XOR AND NOT OR NOT XNOR	**CONDITION CODES UNCHANGED** ADD SUBTRACT MULTIPLY(SIGNED/UNSIGNED) SCAN SETHI SHIFT LEFT LOGICAL SHIFT RIGHT LOGICAL SHIFT RIGHT ARITHMETIC **CONDITION CODES SET** ADD SUBTRACT MULTIPLY(SIGNED/UNSIGNED) MULTIPLY STEP DIVIDE STEP **EXTENDED AND CONDITION CODES UNCHANGED** ADD SUBTRACT **EXTENDED AND CONDITION CODES SET** ADD SUBTRACT **TAGGED AND CONDITION CODES SET (WITH AND WITHOUT TRAP ON OVERFLOW)** ADD SUBTRACT	**TO USER/SUPERVISOR SPACE SIGNED** LOAD BYTE LOAD HALF-WORD LOAD WORD LOAD DOUBLE WORD STORE BYTE STORE HALF-WORD STORE WORD STORE DOUBLE WORD **TO USER SPACE UNSIGNED** LOAD BYTE LOAD HALF-WORD **TO ALTERNATE SPACE SIGNED** LOAD BYTE LOAD HALF-WORD LOAD WORD LOAD DOUBLE WORD STORE BYTE STORE HALF-WORD STORE WORD STORE DOUBLEWORD **TO ALTERNATE SPACE UNSIGNED** LOAD BYTE LOAD HALF-WORD **ATOMIC OPERATION IN USER SPACE** SWAP WORD LOAD/STORE UNSIGNED BYTE **ATOMIC OPERATION IN ALTERNATE SPACE** SWAP WORD LOAD/STORE UNSIGNED BYTE
CONTROL TRANSFER		
CONDITIONAL BRANCH CONDITIONAL TRAP CALL RETURN SAVE RESTORE JUMP AND LINK		
READ/WRITE CONTROL REGISTER		
READ PSR WRITE PSR READ TBR WRITE TBR	READ WIM WRITE WIM READ Y WRITE Y	READ ASR WRITE ASR

CPU

The MB86832 core is a high-performance, fully custom implementation of the SPARC architecture. The core is compact to leave chip real estate available for peripheral integration. The modular architecture allows the device family to be customized for varying application requirements. The core is made up of three functional units: the Instruction block, the Address block and the Execute block. (see Figure 1 below)

When a reset or trap occurs, the processor enters supervisor mode. In this mode, instructions and data come from supervisor space. While in supervisor mode, the processor has access to all protected ASI spaces. Four ASI spaces have been reserved for application-defined data spaces.

The distinction between user and supervisor space allows the hardware to protect against program errors. In developing real-time applications, for example, the separate spaces provide a mechanism for protecting the operating system from bugs in application code.

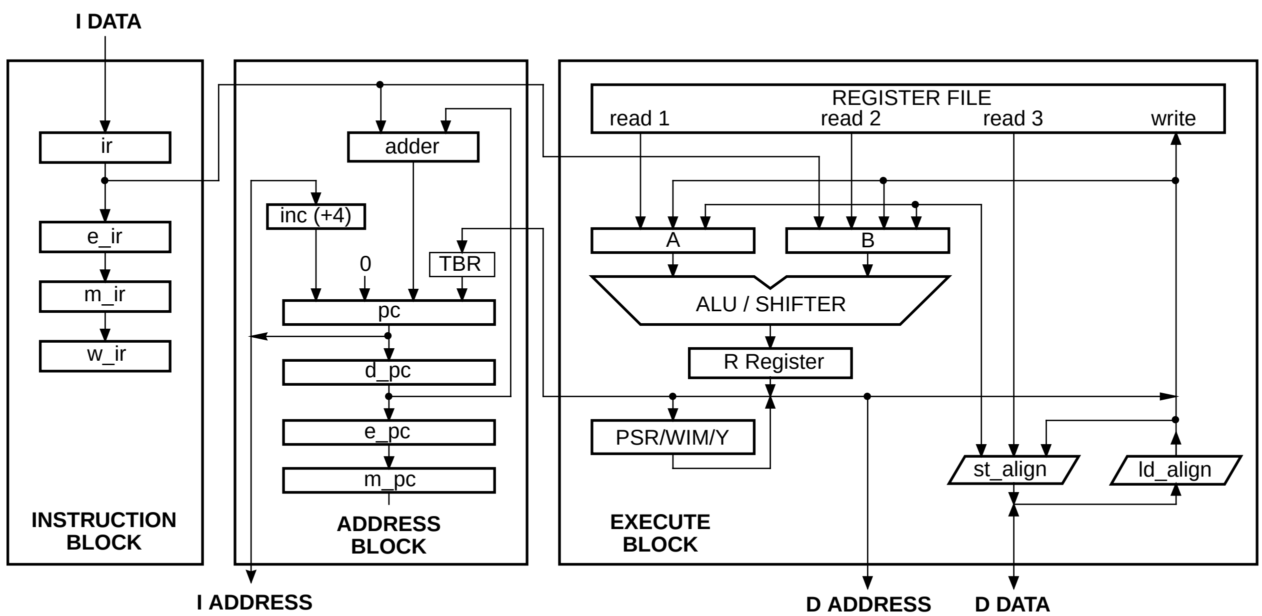


Figure 1. MB86832 Integer Unit Data Path

A five-stage instruction pipeline decodes the instructions and generates the control signals to the other blocks. The pipeline consists of Fetch (F), Decode (D), Execute (E), Memory (M), and Writeback (W) stages. Instruction memory is addressed and instructions are returned in the (F) stage, the register file is addressed and operands are returned in the (D) stage, the ALU produces results in the (E) stage, external memory is addressed in the (M) stage, and the register file is written back in the (W) stage.

Address Space

The MB86832 offers multiple large address spaces including separate user and supervisor spaces. In addition to 26 address lines, 8 alternate address space identifier bits (ASIs) distinguish between protected and unprotected space. Of the 256 possible ASI values, two select user instruction and data spaces, while the remaining ASI values define supervisor spaces. Included in the latter are peripheral control registers and direct access to the cache tag and data memories.

Registers

The MB86832 integer unit register set consists of both general-purpose registers and dedicated registers used for control and status.

The 136 general-purpose registers are divided into 8 global registers and 8 overlapping blocks or “windows”. Each window contains 24 registers. Of these, 8 are local to the window, 8 “out” registers overlap with the next window, and 8 “in” registers overlap with the previous window (see Figure 2 on page 16).

This organization makes it easy to pass parameters to subroutines. Parameters are written to the “out” registers and the subsequent procedure call decrements the window pointer to make a new set of registers available. The passed parameters are now available to the subroutine in the current window’s “in” registers.

Register windows improve performance in embedded applications because they function as local variable caches which retain either interrupt, subroutine, context, or operating system variables with no additional overhead. In addition, code space can be reduced by exploiting the efficient execution of procedure linkage.

The registers that make up the register file each have three read ports and one write port. The use of a four-port register file allows instructions to execute at one instruction per cycle, even in the case of the store instruction which can require reading up to three register operands.

The control and status registers include those defined by the SPARC architecture (see Table 11 on page 24) and those mapped into alternate address space to control peripheral functions (see Table 12 on page 25).

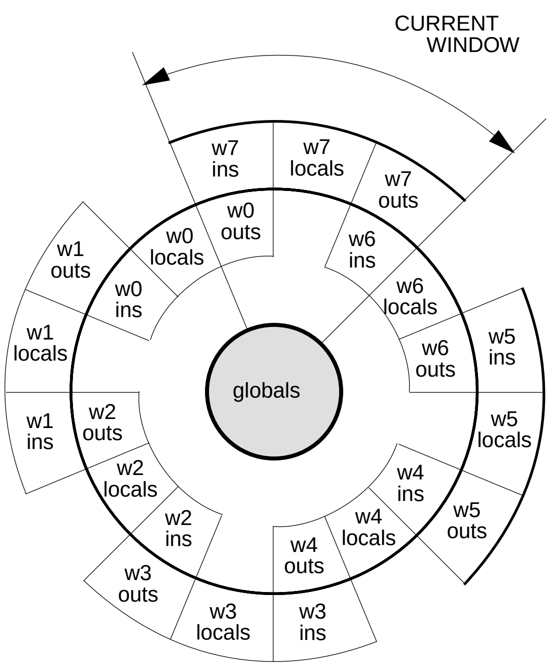


Figure 2. General-Purpose Register Window Organization

Instruction Sets

The MB86832 is upward-code compatible with other SPARC processors. Integer divide step, and scan for first changed bit have been added to the already powerful SPARC instruction set to improve performance in embedded applications. See Table 5 on page 14 for a list of the instructions.

Interrupt

A key measure of a processor's suitability for use in an embedded application is its ability to handle interrupts with a minimum of latency and in a deterministic fashion. The MB86832 implementation has been optimized to insure not only low average latency but low maximum latency as well.

Interrupt response time is the sum of the time it takes the processor to finish its current task after recognizing an interrupt and the time it takes to begin executing interrupt service routine instructions. The MB86832 implements numerous features to minimize both factors.

To minimize the time it takes to finish the current task, the MB86832 is designed so that tasks can either be interrupted or completed in a minimum number of cycles. Implementation details that accomplish this include cache line misses that can be filled one word at a time through a prefetch buffer, integer divide that is interruptible through the use of a divide step instruction, fast multiply and a four-stage write buffer to defer pending bus transactions.

To minimize the time required to start executing the interrupt service routine, the processor switches to a new register window when an interrupt is detected. This feature allows the service routine to begin execution without first saving any registers on the stack. For even faster response, the application can also lock the service routine into the cache. This eliminates any latency caused by cache misses. The on-chip data cache can be used by the service routine as a fast local stack for minimum delay in accessing data.

Single vector trapping is a technique for saving code space and improving interrupt latency. When the SVT bit of the Ancillary State Register 17 is set, all traps vector through the first entry in the trap table rather than indexing to individual entries for each trap type. In some applications, this can allow the trap table to fit in the cache.

There are 15 different interrupt levels. The highest interrupt level is non-maskable.

Cache

The MB86832 has separate on-chip 2-way set-associative instruction and data caches. This allows the user to build a high-performance system without incurring the cost of requiring fast external memory and the associated control logic.

The caches use a buffered writethrough mechanism. Read hits can be satisfied by the caches without generating external bus cycles. Write hits are applied to both the cached data and the image of the data in external memory. Because writes to memory are buffered, the CPU can continue execution from cache without pausing to allow writes to external memory to complete. The caches are physically mapped.

The data cache is organized as two banks of 128 lines (see Figure 3 on page 18 for the organization of one bank). The instruction cache is organized as two banks of 128 lines (see Figure 4 on page 18 for the organization of one bank).

Lines are divided into sub-blocks each four bytes wide. On a cache miss, the caches are updated either 1 word (4 bytes) at a time, or 4 words at a time using the processor's burst mode feature. Single-word updates minimize interrupt latency associated with long cache line replacements, while 4-word burst refills maximize the use of available bus bandwidth. An instruction pre-fetch buffer fetches the next sequential instruction anticipating that it will be needed to satisfy the next instruction cache miss.

The caches can be used in either normal mode or one of two lock modes. The two lock modes allow either the entire cache or just selected cache lines to be locked. The lock modes allow time-critical or performance-sensitive instructions and data to be locked in cache.

Global locking affects the entire content of either the instruction or data cache. Two control bits in the Cache/BIU Control Register enable or disable locking for either cache. With the entire cache locked, no valid cache line can be kicked out of the cache. To insure best performance however, invalid lines are allocated if possible. This is done automatically and incurs no time penalty.

Local cache locking makes it possible to dynamically lock selected instructions or data in the appropriate cache. This feature provides the flexibility, for example, to implement a known, deterministic response for certain critical interrupt routines by locking the routine's code and data into the cache. Cache lines can also be locked to give priority to often used instructions or data which might otherwise be removed from cache.

In local lock mode, each entry can either be locked individually by software or automatically with hardware assist. For individual locking, software writes the lock bit in the appropriate cache tag line. For automatic locking, a bit in each Lock Control Register enables or disables the feature. The enable bit is set at the beginning of a routine for which the entries are to be locked. This causes the location of any cache access occurring while the bit is enabled to be locked into the cache. In addition to requiring just one initial cycle to enable, automatic entry locking incurs no overhead while in effect. Locked locations can be cleared with a single write to a control register.

In unlocked operation, the data cache uses a write-through update policy and allocates a cache line only on a load. Writes are buffered so that the processor can continue executing while data is written back to memory. In contrast, writes to locked data cache locations are not written through to main memory. Besides reducing external bus activity, this effectively configures a portion of data cache as on-chip RAM which does not map to external memory.

The data and instruction caches are designed to be accessed independently over separate data and instruction buses to allow data to be loaded from and stored to the cache at peak rates of 1 per instruction.

Different data memory spaces can be configured as cacheable or non-cacheable through either software programming or hardware control.

Following reset, bit 7 of Cache/BIU Control Register (ASI=0x01 ADR=0x0000 0000) is initialized so that cacheability is controlled by a hardware pin, $\overline{\text{NONCACHE}}$. When the $\overline{\text{NONCACHE}}$ pin is low, the data associated with the address is non-cacheable, otherwise it is cacheable. In this mode, the hardware control of cacheability is independent of the chip selects.

The user can set bit 7 of Cache/BIU Control Register to allow software to control cacheability. By programming a few bits in the Bus Width and Cacheable Register (ASI = 0x01, address = 0x0000 016C), cacheability can be controlled by which chip select is used.

Cacheability for the $\overline{\text{CS4}}$ and $\overline{\text{CS5}}$ chip selects are special cases. When the internal DRAM controller is disabled, the cacheability of $\overline{\text{CS4}}$ and $\overline{\text{CS5}}$ is the same as the other chip selects. When the internal DRAM controller is enabled, the data memory space selected by $\overline{\text{CS4}}$ is cacheable, and the data memory space referred by $\overline{\text{CS5}}$ is non-cacheable. Unlike the other chip selects, $\overline{\text{CS5}}$ can be programmed to overlap the $\overline{\text{CS4}}$ address range to define a noncachable region within DRAM.

Bus Interface

The bus interface unit (BIU) is designed for simplicity and high performance. Separate address and data buses make it easy to build fast systems. At the same time, on-chip circuitry allows these systems to be built with a minimum of external hardware.

The BIU runs at the rate of the external clock, however the CPU and core logic can run at rates of ×1, ×2, ×3, ×4, or ×5 that rate. This is provided to ease the system design for applications where the CPU is running at a high frequency.

The bus interface supports fully programmable wait-state generation, address decoding with chip select outputs, booting from 8-, 16-, and 32-bit wide memory, and an auto-reload timer. A burst mode supports fast cache line fills. Address pins ADR<3:2> track the internal address changes during burst mode.

Each chip select can also be programmed to support 8-, 16-, or 32-bit wide memory. An exception is when $\overline{CS4}$ and $\overline{CS5}$ are used with the internal DRAM controller enabled, only 16- and 32-bit width is supported for these areas. See the section on the DRAM controller for a more detailed description of DRAM access.

Interrupt Controller

The interrupt controller (IRC) functions are a superset of the IRC functions of the MB86930 and MB86940 devices. It has four modes:

Trigger Mode Registers 0 and 1 set the trigger modes for each channel of channels 8 through 15 and channels 1 through 7, respectively. As shown in Table 6, an interrupt can be triggered by a high level, low level, rising edge, or falling edge.

Note: IRQx signals that do not have pins are tied low internally. If the trigger mode for these signals is set to low level, continuous interrupts will be generated.

Table 6. Interrupt Trigger Modes

TRGMD	Trigger Mode
00	High Level (Initial Value)
01	Low Level
10	Rising Edge
11	Falling Edge

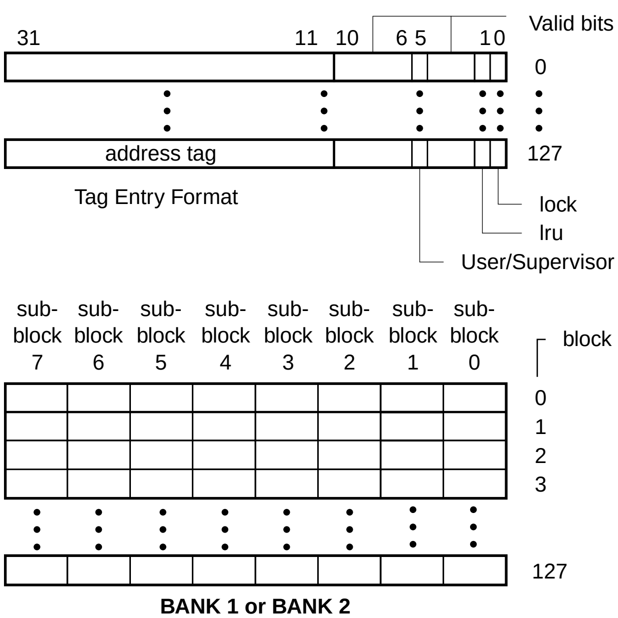


Figure 3. Data Cache Organization

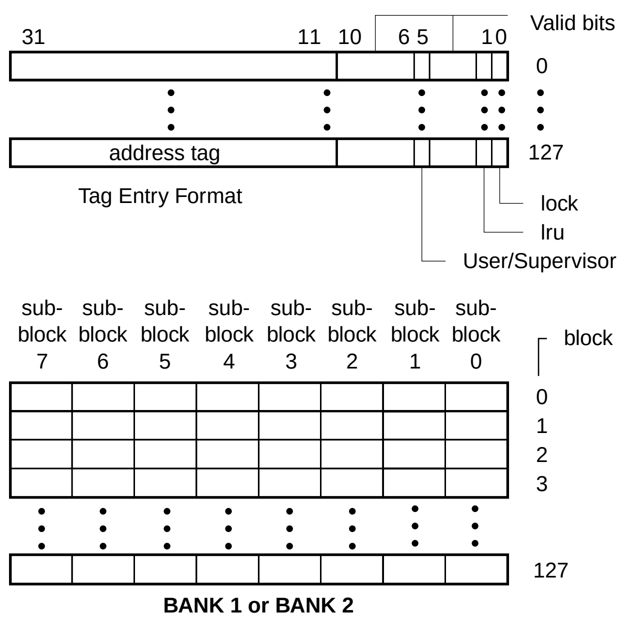


Figure 4. Instruction Cache Organization

Request Sense Register (Read only)

31 — 16	15 — 0
Reserved (no bits)	Request Sense 15-1

Bit 15-1: REQSNS15-1 (initial value all 0)
Shows channel 15-1 interrupt requests. If 1, indicates that an interrupt request is pending.
Bit 0: Reserved (0 during reads)

If events set by Trigger Mode Registers are detected, bits corresponding the events are set in the Request Sense Register.
The Request Sense Register is read-only, and it is cleared by a reset.
If an IRQx signal may be high following reset (i.e. the default trigger mode), its trigger mode should be programmed and any pending interrupt request cleared before enabling the interrupt controller.

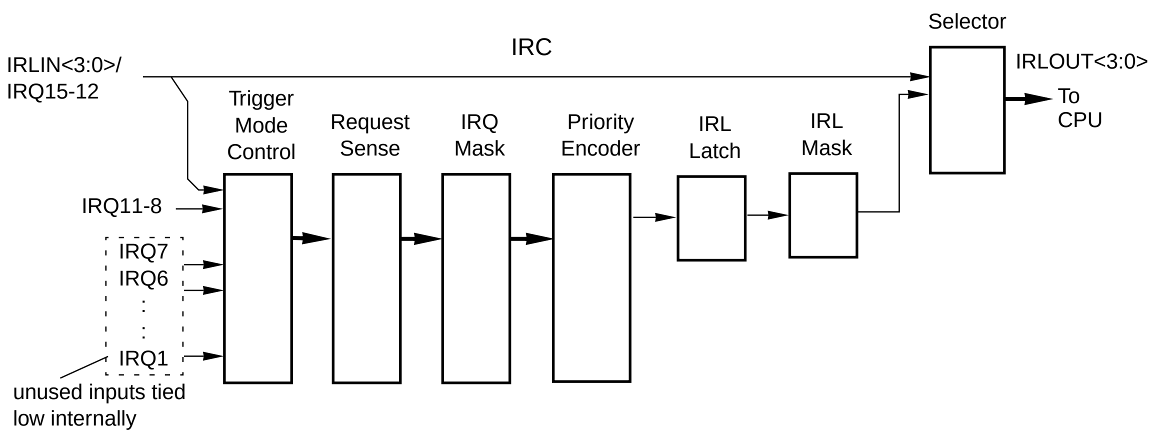


Figure 5. Interrupt Controller Configuration

IRC Mode Register

31 — 16	15 — 2	1 — 0
Reserved (no bits)	Reserved (reads as zero)	IRCMD

Bit1, 0: IRCMD (Read/Write). (Initial value 00). IRC Mode Bit.

Table 7. Interrupt Controller Modes

IRCMD	IRC Mode
00	IRC disabled. IRL<3:0> inputs are selected.
01	IRC enabled. IRL<3:0> are replaced by IRQ<15:12>, and the IRQ<11:8> inputs are available.
10	Reserved setting. *Do not use.*
11	Reserved setting. *Do not use.*

The IRC Mode Register is used to enable the interrupt controller. When the IRCMD bits in this register are $00_{(2)}$, the interrupt controller is disabled, and the encoded IRL<3:0> is passed to the IU as received from an external interrupt controller. When the IRCMD bits are $01_{(2)}$, the interrupt controller is enabled, and the IRL<3:0> pins become interrupt request inputs IRQ<15:12>.

Interrupt Controller Operation

The interrupt controller is enabled by setting the IRCMD bits of the IRC Mode Register to $01_{(2)}$. The interrupt requests then come from IRQ<15:1>. The number of interrupt inputs available at pins varies depending on how IRC resources are programmed. Interrupt requests are stored in the Request Sense Register when their trigger mode conditions are met. Of the stored interrupts not masked in the Interrupt Mask Register, those with the highest priority are encoded and stored in the IRL Latch/Clear Register.

If the IRL Mask (IM) bit in the Interrupt Mask Register is clear, the IRL is passed to the CPU. The interrupt is acknowledged by setting the CL bit of the IRL Latch/Clear Register, which clears the current IRL and allows the next interrupt level to be loaded.

If unmasked interrupt requests are pending in the Request Sense Register, they are loaded into the IRL Latch/Clear Register in order of priority.

If the IRCMD bits of the IRC Mode Register are $00_{(2)}$, the interrupt controller is disabled and the signals on IRL<3:0> are passed to the CPU without modification.

1. **Processing Interrupts as Traps**

   After reset, all of the mask bits in the Interrupt Mask Register are set (i.e. interrupts are masked). Software should then program the trigger modes and set bits 1 to 15 of the Request Clear Register to clear any pending interrupts. Next, the interrupt masks for the desired interrupts should be cleared and the IRL mask (bit 0 in the Interrupt Mask Register) should be cleared.

   In the trap processing routine, software acknowledges the interrupt by first clearing the bit in the Request Clear Register corresponding to the interrupt, then clears the IRL latch by setting the CL bit in the IRL Latch/Clear Register. This allows the next pending interrupt request to be stored in the IRL latch.

2. **Processing Interrupts By Polling**

   a. Polling the Request Sense Register
   Software can read the masked interrupt request sense bits in the Request Sense Register and call the corresponding service routines when set bits are found. The service routine acknowledges the interrupt by clearing the bit through the Request Clear Register. This method is compatible with the method described above, with unmasked interrupts processed as traps and masked interrupts processed by polling.

   b. Polling the IRL Latch/Clear Register
   This is similar to processing interrupts as traps, except software polls the IRL Latch/Clear Register without generating traps. Unmasked interrupts are prioritized, encoded, and loaded into the IRL latch, however the IM bit in the Interrupt Mask Register is set, which prevents a trap from being called. Software can poll the IRL latch bits, call a service routine if it has any value other than $0000_{(2)}$, then use the Request Clear Register to clear the corresponding bits in the Request Sense Register. The service routine acknowledges the interrupt by first clearing the bit in the Request Clear Register corresponding to the interrupt, then clears the IRL latch by setting the CL bit in the IRL Latch/Clear Register. This allows the next pending interrupt request to be stored in the IRL latch. Because the IM bit is set, this method is not compatible with processing some interrupts as traps.

Clock Generator

The on-chip clock generator requires an external clock source (i.e. there is no on-chip oscillator). The external clock frequency is the same as the bus interface unit (BIU) operating frequency. The skew between the internal clock and an external input clock source is minimized through the use of an on-chip phase-locked loop.

The CPU and core logic can run at up to 5 times the frequency of external clock (Max. BIU frequency = 40 MHz for 100 MHz and 33 MHz for 66 and 80 MHz version. This is enabled by the use of CLKSEL pins, as shown below.

Table 8. Clock Multiplication Factor

CLKSEL2	CLKSEL1	CLKSEL0	Internal Clock
H	L	L	x1
H	L	H	x2
H	H	L	x3
H	H	H	x4
L	H	H	x5

Clock Gear

A mechanism has been provided for software to modify the clock multiplication factor. This can be used for reducing power consumption in Sleep Mode.

Internal Clock Control/Status (ICCS) Register CS3 = L, Offset = 0x02C

31–5	4	3	2–0
Reserved	CLKST	CE	CLKSEL

bits 31-5 : Reserved—Write as 0, read undefined.
bit 4 : CLKST—Current value of the clock multiplication factor. At reset, it is set to the values sampled on the CLKSEL<2:0> input pins
bit 3 : CE—Enables loading of the new clock multiplication factor when Sleep Mode is entered, when set to 1. At reset, cleared to 0.
bits 2-0 : CLKSEL—Specifies the new value for the clock multiplication factor. At reset, set to 111_2.

Encoding of CLKST and CLKSEL Multiplication Factors

Bit Value	Multiplication Factor
000-010	Reserved
011	x5
100	x1
101	x2
110	x3
111	x4

Changing the Clock Multiplication Factor

The following sequence changes the clock multiplication factor:

1. Disable caching, so access becomes entirely external access.
2. Bit 3 of the Internal Clock Control/Status Register (ICCS: CS3=L, ADR=0x02C) is set to 1, and bits 2:0 are set to the desired clock multiplication factor. Note that at this point the clock multiplication factor has not changed.
3. Enter Sleep Mode as described in the Key Features section. The clock multiplication factor will change at that time.

DRAM Controller

- High-speed page-mode DRAM support (burst mode and non-burst mode supported)
- EDO (hyperpage) mode DRAM support (burst mode use only)
- Memory bus width: 16- and 32-bit bus width supported
- DRAM page size 256 to 4096 columns
- Self-refresh during sleep mode support
- Programmable RAS and CAS timing parameters
- DRAM controller enabled by setting the DRAM controller enable bit (bit 6) of the System Support Control Register (SSCR)
- Burst mode enabled by setting the DRAM burst enable bit (bit 7) of the System Support Control Register

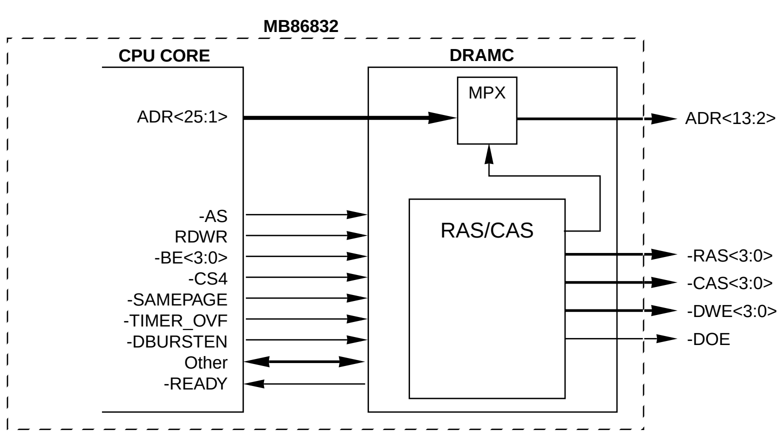


Figure 6. DRAM Controller Configuration

Programmable Chip Select

The core logic includes six programmable chip selects with wait-state generator circuits. Chip select address ranges must not overlap. Some chip selects have assigned functions, as shown in Table 9, "Use of Chip Select," on this page.

Chip select settings and wait state settings must be made while the cache is disabled. If the write buffer is operating with the cache enabled and these settings are changed, the bus interface unit may stop operating normally. ARSR (Address Range Specifier Register), AMR (Address Mask Register), SPGMR (Same Page Mask Register) and WSSR (Wait State Specifier Register) settings must be made during reset initialization, before enabling the cache.

Table 9. Use of Chip Select

CS0	Boot ROM area
CS1	General purpose
CS2	General purpose
CS3	Control and status registers for external peripheral control and status registers
CS4	DRAM area
CS5	Defines a noncacheable region when the noncache function is enabled and the built-in DRAM controller is enabled

CS0

The CS0 output is a chip select for boot ROM. No Address Range Specifier Register exists for CS0. It has a base address of 0. Following reset, the CPU vectors to address 0. Unlike the other chip select outputs, the CS0 output defaults to an enabled condition after reset.

CS1 and CS2

General-purpose chip selects with no special limitations.

CS3

The CS3 area is assigned to peripheral control and status registers, and it is limited to a range of 1 Kbyte including both internal and external registers. When on-chip peripherals are accessed, the bus signals ADR<27:2>, AS, RDWR, BE<3:0>, and RDYOUT are driven with the same timing used for external bus cycles. Data also appears on D<31:0> when storing to on-chip peripherals selected with CS3.

The CS3 area can be set to 16- or 32-bit bus width, but 8-bit is not available. 32-bit bus-width peripherals can be assigned to CS3, and when using the Wait State Specifier Register, the Single Cycle Non Burst Mode bit and the Single Cycle Burst Mode bit must be clear. One or more wait states must be programmed, and the Override bit must be set. These rules apply to both on-chip and off-chip peripherals. On-chip peripherals, however, must be accessed with ADR<1:0> equal to zero using Half-Word Load and Half-Word Store instructions, because the on-chip peripherals only use D<15:0>.

CS4

The CS4 area is assigned to DRAM. When the built-in DRAM controller is used, CS4 is the DRAM chip select. Even if an external DRAM controller is used, CS4 must be assigned to DRAM, because the page-hit detection circuit (i.e. SAMEPAGE generator circuit) is enabled for the CS4 area. Whether the on-chip DRAM controller or an external controller is used, CS4 wait-state generation must be disabled. In the Wait State Specifier Register, the Wait Enable bit and the Single Cycle Non Burst Mode bit must be clear.

CS5

When a noncacheable area is not present, CS5 can be used as a general-purpose chip select. When both a noncachable area and the on-chip DRAM controller are used, the CS5 area is forced to be noncacheable. Unlike the other chip selects, the CS5 address range may overlap the CS4 area, to define a noncacheable area in DRAM. These areas may overlap whether or not the DRAM controller is enabled. When CS5 is used in a DRAM area, the CS5 wait state generator must be disabled. In the Wait State Specifier Register, the Wait Enable bit and the Single Cycle Non Burst Mode bit must be clear. If the DRAM controller is enabled, the bus width of the CS5 area will be the same as that specified for the CS4 area.

Idle Cycle Insertion Function

IDLEEN Pin Tied Low

When the IDLEEN pin is tied low and a write cycle comes immediately after a read cycle, an idle cycle is automatically inserted for one CLKIN period.

IDLEEN Pin Tied High

When the IDLEEN pin is tied high and a bus cycle immediately follows an access to the CS0 area, an idle cycle is inserted for two CLKIN periods unless the second bus cycle also accesses the CS0 area. Because EPROM output disable time is long, this function can eliminate a tristate buffer in some designs. A single idle cycle is inserted when a DRAM write immediately follows a DRAM read, as shown in Table 10, "Idle Cycle Insertion Function," on page 23.

Table 10. Idle Cycle Insertion Function

Previous Cycle	Following Cycle	Inserted Idle Cycles
Instruction fetch from ROM	Instruction fetch	0
Instruction fetch from ROM	Data read	2
Instruction fetch from ROM	Data read	2
Data read from ROM	Instruction fetch	0
Data read from ROM	Data read	2
Data read from ROM	Data write	2
Instruction fetch from DRAM	Instruction fetch	0
Instruction fetch from DRAM	Data read	0
Instruction fetch from DRAM	Data write	1
Data read from DRAM	Instruction fetch	0
Data read from DRAM	Data read	0
Data read from DRAM	Data write	1
Data write from DRAM	Instruction fetch	0
Data write from DRAM	Data read	0
Data write from DRAM	Data write	0

Table 11. MB86832 Control and Status Registers—Read/Write

Processor State Register	PSR

31–28	27–24	23–20	19–12	11–8	7	6	5	4–3	2–0
0 0 0 0	1 0 0 1	icc: n z v c	reserved	PIL					

Conditions
n: (Negative=1, Non-Negative=0)
z: (Zero=1, Non-Zero=0)
v: (Overflow=1, No Overflow=0)
c: (Carry=1, No Carry=0)

- Bits 11–8: Processor Interrupt Level (Value 1-15, RST=Undefined)
- Bit 7: S MODE (Supervisor=1, User=0, RST=Undefined)
- Bit 6: Prior S Mode
- Bit 5: Enable Trap (Enable=1, Disable=0, RST=0)
- Bits 4–3: Reserved
- Bits 2–0: Current Window Pointer (Value=0-7, RST=Undefined)

Window Invalid Mask	WIM

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

Bits 7–0: Window Invalid Mask (Invalid=1, Valid=0, RST=Undefined)

Trap Base Register	TBR

31–12	11–4	3–0
Trap Base Address (RST=Undefined)	Trap Type (RST=0)	NULL

Y Register	Y

31–0
used for MULScc, DIVScc, MU

Ancillary State Register 17	ASR 17

31–1	0
reserved (must be zero on writes)	

Bit 0: Single Vector Trapping (Enabled=1, Disabled=0, RST=0)

Program Counter	PC

31–0
instruction address

Next Program Counter	NPC

31–0
instruction address

Table 12. MB86832 Memory Mapped Control Registers—Read/Write

Cache/BIU Control

ASI	ADDRESS
0x 1	0x 0000 0000

Bit	Field
31–10	reserved
9–8	Non-Cacheable Wait-state
7	Cacheability Enable (Enabled=1, Disabled=0, RST=0)
6	Reserved
5	Write Buffer Enable (Enabled=1, Disabled=0, RST=0)
4	Prefetch Buffer Enable (Enabled=1, Disabled=0, RST=0)
3	Global Data Cache Lock (Lock=1, Unlock=0, RST=0)
2	Data Cache Enable (Enabled=1, Disabled=0, RST=0)
1	Global Instruction Cache Lock (Lock=1, Unlock=0, RST=0)
0	Instruction Cache Enable (Enabled=1, Disabled=0, RST=0)

Lock Control

ASI	ADDRESS
0x 1	0x 0000 0004

Bit	Field
31–2	reserved
1	Data Cache Entry Auto Lock (Disabled=0, Enabled=1, RST=0)
0	Instruction Cache Entry Auto Lock (Disabled=0, Enabled=1, RST=0)

Lock Control Save

ASI	ADDRESS
0x 1	0x 0000 0008

Bit	Field
31–2	reserved
1	Previous Instruction Cache Auto Lock (Off=0, On=1, RST=0)
0	Previous Data Cache Auto Lock (Off=0, On=1, RST=0)

Cache Status

ASI	ADDRESS
0x 1	0x 0000 000C

Bit	Field
31–1	reserved
0	Auto Lock Failed (False=0, True=1, RST=0)

Restore Lock Control

ASI	ADDRESS
0x 1	0x 0000 0010

Bit	Field
31–1	reserved
0	Restore Lock Control Register (Restore=1, Ignore=0, RST=0)

Bus Control

ASI	ADDRESS
0x 1	0x 0000 0020

Bit	Field
31–2	reserved
1	Data Burst Enable (Enable=1, Disable=0, RST=0)
0	Instruction Burst Enable (Enable=1, Disable=0, RST=0)

System Support Control Register (SSCR)

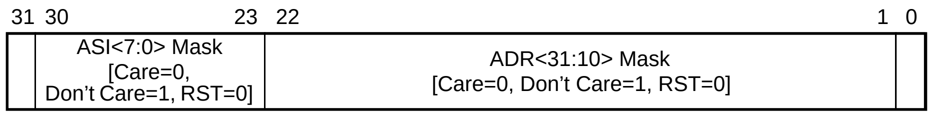
ASI	ADDRESS
0x 1	0x 0000 0080

Bit	Field
31–8	reserved
7	DRAM Burst enable
6	DRAM controller enable
5	Same Page Enable (Enabled=1, Disabled=0, RST=0)
4	Chip Select Enable (Enabled=1, Disabled=0, RST=0)
3	Programmable Wait-State (Enabled=1, Disabled=0, RST=1)
2	Timer On/Off (Enabled=1, Disabled=0, RST=0)
1	DMA Cycle Steal (Enabled=1, Disabled=0, RST=0)
0	Parity (Odd=1, Even=0, RST=0)

Table 12. MB86832 Memory Mapped Control Registers—Read/Write (continued)

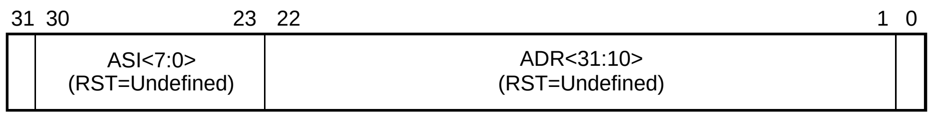
Same Page Mask (SPGMR)

ASI	ADDRESS
0x 1	0x 0000 0120



Address Range Specifier Register (ARSR)[1]

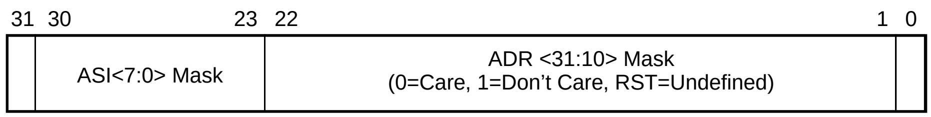
ASI	ADDRESS	
0x 1	CS1	0x 0000 0124
	CS2	0x 0000 0128
	CS3	0x 0000 012C
	CS4	0x 0000 0130
	CS5	0x 0000 0134



NOTE: CS0 is hardwired to ASI=0x9 ADR<31:10> = <0..0>

Address Mask (AMR)

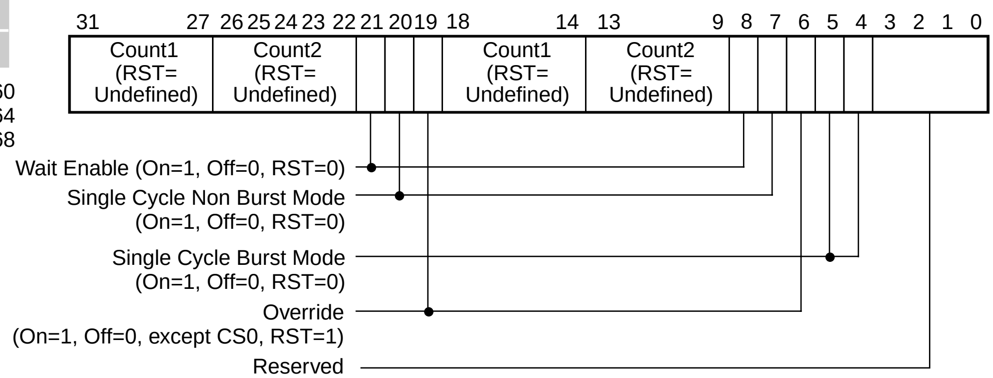
ASI	ADDRESS	
0x 1	CS0	0x 0000 0140
	CS1	0x 0000 0144
	CS2	0x 0000 0148
	CS3	0x 0000 014C
	CS4	0x 0000 0150
	CS5	0x 0000 0154



NOTE: CS0 ADR<14:10> = 1, ADR<31:15> = 0, ASI = 0x9 at reset.

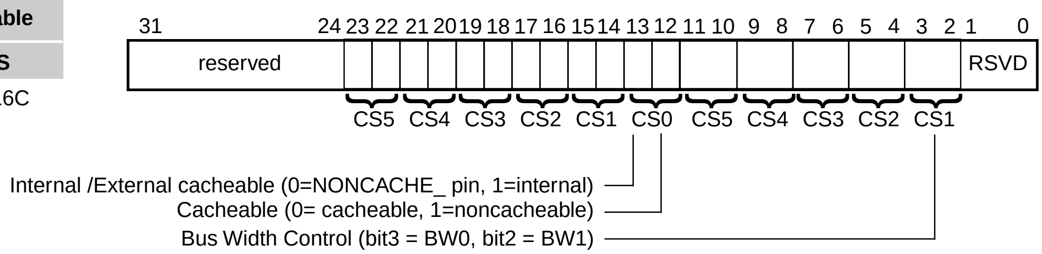
Wait State Specifier (WSSR)

ASI	ADDRESS	
0x 1	CS1,CS0	0x 0000 0160
	CS3,CS2	0x 0000 0164
	CS5,CS4	0x 0000 0168



Bus Width and Cacheable

ASI	ADDRESS
0x 1	0x 0000 016C



1. This register is write only

Table 12. MB86832 Memory Mapped Control Registers—Read/Write (continued)

DRAM Refresh Timer Pre-Load	
ASI	ADDRESS
0x 1	0x 0000 0178

31	30 … 16	15 … 0
	reserved	Timer Pre-Load Value (RST=Undefined)

bit31 : 3Cycle Mode (On-1, Off=0, RST=0)
bit15-0 : Timer Pre-load Value (RST=0xffff)
bit30-16 : Reserved [“0” write, read undefined]

DRAM Refresh Timer	
ASI	ADDRESS
0x 1	0x 0000 0174

31 … 16	15 … 0
reserved	Timer Value (RST=Undefined)

bit31-16 : Reserved [“0” write, read undefined]
bit15-0 : Timer Value (RST=undefined)

Ancillary Version Register (V E R 2) [Read Only] ASI=0x01, Address=0x00020000

31 … 16	15															0
Reserved	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

bit31-16 : Reserved [Read undefined]
bit15-0 : MB86831 Version Number (Value=1)

Sleep Mode Register (S L P M D) [Write Only] ASI=0x01, Address=0x00020004

31 … 1	0
Reserved	

bit31-1 : Reserved
bit0 : Sleep Mode (On=1, Off=0, RST=0)

Internal Clock Control/Status Register (ICCS) $\overline{CS3}$=L, Offset=0x02C

31 … 5	4	3	2 … 0
Reserved	CLKST	CE	CLKSEL

bits 31-5 : Reserved—Write as 0, read undefined.
bit 4 : CLKST—Current value of the clock multiplication factor. At reset, it is set to the values sampled on the CLKSEL<2:0> input pins
bit 3 : CE—Enables loading of the new clock multiplication factor when Sleep Mode is entered, when set to 1. At reset, cleared to 0.
bits 2-0 : CLKSEL—Specifies the new value for the clock multiplication factor. At reset, set to 111_2.

Encoding of CLKST and CLKSEL Multiplication Factors

Bit Value	Multiplication Factor
000-010	Reserved
011	x5
100	x1
101	x2
110	x3
111	x4

Table 12. MB86832 Memory Mapped Control Registers—Read/Write (continued)

Trigger Mode0 Register (T R G M 0) — $\overline{CS3}$=L, Address<9:2>=0x00

31–16	15–14	13–12	11–10	9–8	7–6	5–4	3–2	1–0
Reserved	ch15	ch14	ch13	ch12	ch11	ch10	ch9	ch8

bit31-16 : Reserved [“0” write, read undefined]
bit15-0 : Trigger Mode (High Level=00, Low Level=01, High Edge=10, Low Edge=11, RST=00)

Trigger Mode1 Register (T R G M1) — $\overline{CS3}$=L, Address<9:2>=0x01

31–16	15–14	13–12	11–10	9–8	7–6	5–4	3–2	1–0
Reserved	ch7	ch6	ch5	ch4	ch3	ch2	ch1	00

bit31-16 : Reserved [“0” write, read undefined]
bit15-2 : Trigger Mode (High Level=00, Low Level=01, High Edge=10, Low Edge=11, RST=00)
bit1-0 : Reserved [“0” write, read “0”]

Request Sense Register (R E Q S N S) [Read Only] — CS3=L, Address<9:2>=0x02

31–16	15–1	0
Reserved	Request Sense 15-1	

bit31-16 : Reserved [“0” write, read undefined]
bit15-1 : Request Sense 15-1 (RST=0)
bit0 : Reserved [“0” write, read “0”]

Request Clear Register (R E Q C L R) [Write Only] — CS3=L, Address<9:2>=0x03

31–16	15–1	0
Reserved	Request Clear 15-1	

bit31-16 : Reserved [“0” write]
bit15-1 : Request Clear 15-1 (Clear=1, Not Clear=0)
bit0 : Reserved [“0” write]

Interrupt Mask Register (I M A S K) — $\overline{CS3}$=L, Address<9:2>=0x04

31–16	15–1	0
Reserved	Mask 15-1	IM

bit31-16 : Reserved [“0” write, read undefined]
bit15-1 : Mask 15-1 (Mask=1, Not Mask=0, RST=1)
bit0 : IRL Mask (Mask=1, Not Mask=0, RST=1)

IRL Latch/Clear Register (I R L A T) — $\overline{CS3}$=L, Address<9:2>=0x05

31–16	15–5	4	3–0
Reserved	Reserved	CL	IRL

bit31-16 : Reserved [“0” write, read undefined]
bit15-5 : Reserved [“0” write, read “0”]
bit4 : IRL Clear [write only] Clear=1, Not Clear=0
bit3-0 : IRL Latch [read only] (RST=0000)

IRC Mode Register (I M O D E) — $\overline{CS3}$=L, Address<9:2>=0x06

31–16	15–2	1–0
Reserved	Reserved	IRCMD

bit31-16 : Reserved [“0” write, read undefined]
bit15-2 : Reserved [“0” write, read “0”]
bit1-0 : IRC Mode [I R C M D] (Disable=00, Enable=01, RST=00)

Table 12. MB86832 Memory Mapped Control Registers—Read/Write (continued)

DRAM Bank Configuration Register (D B A N K R)	$\overline{CS3}$=L, Address<9:2>=0x08

31	30–11	10–9	8	7	6–4	3–0
ERR	Reserved	STADR	HE	TP	COL	BKSIZE

bit31 : Access Error (ERR) (Error=1, No Error=0, RST=0, "0" write Clear)
bit30-11 : Reserved ["0" write, read undefined]
bit10-9 : DRAM Start Address [STADR] (RST=0)
bit8 : Hyper Page Enable [HE] (Page Mode DRAM=0, EDO DRAM=1, RST=0)
bit7 : DRAM Type [TP] (4CAS-1WE=0, 4WE-1CAS=1, RST=0)
bit6-4 : Column Address [COL] (RST=011)
bit3-0 : Bank Size [BKSIZE] (RST=0011)

DRAM Timing Register (D T I M R)	CS3=L, Address<9:2>=0x09

31–5	4	3–2	1	0
Reserved	T_{RPS}	T_{RASCBR}	T_{CAS}	T_{RP}

bit31-5 : Reserved ["0" write, read undefined]
bit4 : R-AS Pre-Charge Time specifier bit during Self-Refresh [T_{RPS}] (2Cycle=0, 4Cycle=1, RST=1)
bit3-2 : R-AS Pulse Width specifier bit during CBR Refresh [T_{RASCBR}] (1cycle=00, 2cycle=01, 3cycle=10, RST=01)
bit1 : C-AS Pulse Width specifier bit [T_{CAS}] (1Cycle=0, 2Cycle=1, RST=1)
bit0 : R-AS Pre-Charge Width specifier bit [T_{RP}] (1Cycle=0, 2Cycle=1, RST=0)

Table 12. MB86832 Memory Mapped Control Registers—Read/Write (continued)

InstructionTag Lock Bits

ASI	ADDRESS
0x 2	Bank 1 0x 0000 0000 ↓ by 8 0x 0000 0FE0 Bank 2 0x 8000 0000 ↓ by 8 0x 8000 0FE0

Bits 31–1: reserved
Bit 0: Entry Lock (Locked=1, Unlocked=0, RST=Undefined)

Data Tag Lock Bits

ASI	ADDRESS
0x 3	Bank 1 0x 0000 0000 ↓ by 8 0x 0000 0FE0 Bank 2 0x 8000 0000 ↓ by 8 0x 8000 0FE0

Bits 31–1: reserved
Bit 0: Entry Lock (Locked=1, Unlocked=0, RST=Undefined)

Instruction Cache Tag

ASI	ADDRESS
0x C	Bank 1 0x 0000 0000 ↓ by 8 0x 0000 0FE0 Bank 2 0x 8000 0000 ↓ by 8 0x 8000 0FE0

Bits 31–12: ADDRESS TAG [RST=Undefined]
Bit 11: RSVD
Bits 10–6, 4–2: Sub Block Valid (Valid=1, Invalid=0, RST=Undefined)
Bit 5: User/Supervisor (User=0, Supervisor=1, RST=Undefined)
Bit 1: Least Recently Used (RST=Undefined)
Bit 0: Entry Lock (Locked=1, Unlocked=0, RST=Undefined)

Instruction Cache Invalidate

ASI	ADDRESS
0x C	Bank 1 0x 0000 1000 Bank 2 0x 8000 1000

Bits 31–2: reserved
Bit 1: Cache LRU, Lock Bit Clear (Write Only) (Clear = 1, Not Clear = 0)
Bit 0: Valid Bit Clear (Write Only) (Clear = 1, Not Clear = 0)

Data Cache Tag

ASI	ADDRESS
0x E	Bank 1 0x 0000 0000 ↓ by 8 0x 0000 0FE0 Bank 2 0x 8000 0000 ↓ by 8 0x 8000 0FE0

Bits 31–12: ADDRESS TAG [RST=Undefined]
Bit 11: RSVD
Bit 10, 9–6: Sub Block Valid (Valid=1, Invalid=0, RST=Undefined)
Bit 5: User/Supervisor (User=0, Supervisor=1, RST=Undefined)
Bits 4–2: reserved
Bit 1: Least Recently Used (RST=Undefined)
Bit 0: Entry Lock (Locked=1, Unlocked=0, RST=Undefined)

Data Cache Invalidate

ASI	ADDRESS
0x E	Bank 1 0x 00001 000 Bank 2 0x 80001 000

Bits 31–2: reserved
Bit 1: Cache LRU, Lock Bit Clear (Write Only)
Bit 0: Valid Bit Clear (Write Only)

Instruction Cache Data

ASI	ADDRESS
0x D	Bank 1 0x 0000 0000 ↓ by 8 0x 0000 0FE0 Bank 2 0x 8000 0000 ↓ by 8 0x 8000 0FE0

Bits 31–0: Address Tag (RST=undefined)

Data Cache Data

ASI	ADDRESS
0x F	Bank 1 0x 0000 0000 ↓ by 8 0x 0000 0FE0 Bank 2 0x 8000 0000 ↓ by 8 0x 8000 0FE0

Bits 31–0: Address Tag (RST=undefined)

Bus Operation

The bus interface unit (BIU) is the interface to external devices. It includes the address and data buses, the interrupt request bus, and various control signals. At any time, the BIU is either handling transfers to or from off-chip devices, arbitrating for bus access, or idle.

Operation of the BIU

On writes to external memory, the BIU uses a four-word write buffer. When the BIU receives a request for a write transaction it stores the write data and address in the write buffer, which releases the IU to continue executing out of on-chip cache. The BIU then proceeds to complete the write to external memory. In most cases, the write buffer will hide external memory latency from the IU. The exceptions are in cases where the write buffer is full from previous transactions or if the subsequent IU cycle results in an instruction cache or data cache read miss. In these cases, IU execution is held until the write buffer is emptied.

The write buffer operates only when both the instruction and data caches are enabled. When the bus is granted to an external bus master, a store to the write buffer does not cause the assertion of PBREQ. This allows the external bus master to continue operating while the CPU is executing out of the on-chip caches.

The BIU includes a one-stage prefetch buffer for instruction fetches. This buffer is used to fetch the next sequential instruction after an instruction cache miss. The instruction is prefetched only if the BIU does not have a request for a bus transaction from the IU and no external device is requesting use of the bus. The prefetch buffer operation is suspended if the buffer is full. This occurs if the prefetched instruction is a hit in the instruction cache. The buffer restarts after another instruction cache miss. If an exception occurs during an instruction prefetch, the exception is not sent to the IU unless the instruction is actually requested by the IU. The prefetch buffer operates only when the instruction cache is enabled.

In any cycle, the BIU can receive a request for access to either or both instruction and data memory. If it receives a request for both in the same cycle, it completes the data memory transaction first.

Exception Handling

The external memory system can indicate an error during a memory operation. The BIU signals the appropriate data or instruction exception to the IU which calls the appropriate trap.

As mentioned above, the IU can continue operation after putting the data and address for a store in the write buffer. If an exception is detected while completing this buffered write, then the BIU signals a data access exception to the IU.

Any system which needs to recover from this error should store the address and data of these write transactions in hardware. If the system can generate both read and write exceptions, then the system must also provide a status bit which indicates whether the exception was generated on a read or a write transaction. With this information, the data access exception service routine can determine the cause of the exception and recover.

If the write buffer is operating, an exception can potentially cause other exceptions due to the flush of the four write buffer levels. A system that needs the ability to recover must store up to four separate sets of address and data.

Bus Cycles

Figure 7 on page 34 through Figure 28 on page 45 illustrate representative combinations of bus cycles.

Load

Regardless of the external bus width (8, 16, or 32 bits), all instruction fetches and data reads (including load byte and load half word) load a 32-bit quantity. This is done for compatibility with MB8693X processors with data cache where the smallest granularity in the cache is one word. Bus width can be programmed based on chip select areas to be 8, 16, or 32 bits.

Load (32-bit bus width)

Whenever a load from data memory is requested or an instruction cache miss occurs, the BIU performs a read from external memory (see Figure 7 on page 34).

With a 32-bit external data bus, a read transaction begins with the BIU asserting AS, to indicate a new bus transaction. The AS signal is de-asserted after one cycle. At the same time the ADR<27:2> and ASI<3:0> bits are driven with the location to be read. The BIU drives the RDWR signal high to indicate a read transaction. Because all loads transfer 32 bits, BE<3:0> are all driven low.

The external memory system responds with the read data on pins D<31:0>. It also asserts the READY signal when the external device is ready for the bus cycle to complete. For slow memory, the READY signal can be delayed until data is valid.

A load double operation is treated as two back-to-back word reads.

Load (16-bit bus width)
When the bus is programmed to be 16 bits wide (defined by the chip select region) every load will transfer 32 bits. Figure 17 on page 40 shows a load (byte, half word, word) operating with an 16-bit bus. For the load byte and load half word instructions, the IU masks off the bits which are not required. For a 16-bit bus, the BE2 pin is defined to be the ADR1 address bit. BE<1:0> and BE3 are unused and driven low.

Load (8-bit wide bus)
When the bus is programmed to be 8 bits wide (defined by the chip select region) every load will transfer 32 bits. Figure 15 on page 39 shows a load (byte, half word, word) operating with an 8-bit bus. For the load byte and load half word instructions, the IU masks off the bits which are not required. For an 8-bit bus BE<3:2> are the ADR<0:1> address bits, respectively. BE<1:0> are unused and are driven to undefined states.

Load with Exception
If the external memory system sees a memory exception it can terminate the current memory transaction by asserting the MEXC and READY signals. The data on the data bus is ignored. See Figure 8 on page 35.

Store
Unlike loads, which always transfer 32 bits, only the minimum number of bus cycles required to complete the store are performed. For example, only two bus cycles are required to do a half-word store on an 8-bit bus.

Store (32-bit bus width)
A write transaction begins with the BIU asserting AS, to indicate the start of a new bus transaction. The AS signal is de-asserted after one clock. At the same time, the ADR<27:2> and ASI<3:0> pins are driven with the location to be written while the D<31:0> pins carry the write data. BE<3:0> indicate which bytes to write for a given type of store operation (byte, half-word, or word). The BIU drives the RDWR signal low to indicate a write transaction. See Figure 9 on page 35.

The external memory system responds with the assertion of READY when it has stored the data. Or, if the internal wait-state generator is enabled, READY is generated internally.

A store double operation is treated as two back-to-back word writes.

Store (16-bit wide bus)
Stores to 16-bit memory are sized to the bus. That is, for a 16-bit bus, a store word requires two cycles while a store halfword or store byte requires a single cycle. Figure 18 on page 40 shows the timing for different types of stores. For a 16-bit bus, BE2 is driven with ADR<1>. BE3 is unused and is driven low. BE<1:0> are defined to be the high and low order byte enables, respectively.

Store (8-bit wide bus)
Stores to 8-bit memory are sized to the bus. That is, for a 8-bit bus, a store word requires four cycles, a store halfword requires two cycles, and store byte requires a single cycle. Figure 17 on page 40 shows the timing for different types of stores. For an 8-bit bus, BE<2:3> are driven with ADR<1:0>. BE<1:0> are unused and are driven to undefined states.

Store with Exception
If an access exception occurs on a write, the external memory system can terminate the current memory transaction by asserting the MEXC and READY signals. The external memory system is expected to ignore the data on the data bus in this situation. See Figure 10 on page 36.

Atomic Load Store
An atomic load store executes as a load followed by a store with no operation allowed in between. The LOCK signal is asserted to indicate that the bus is being used for more than one external memory operation. See Figure 11 on page 36.

There is one idle cycle between the termination of the read and the beginning of the write to provide time for changing the direction of the data bus drivers.

External Bus Request and Grant
Any external device can request ownership of the bus by asserting the BREQ signal. When control of the bus is granted, the BIU asserts the BGRNT signal and floats its bus drivers. In the following cycle, the external device can begin its transaction. On completion of its transaction the external device de-asserts the BREQ signal. The BIU responds by de-asserting the BGRNT signal in the following cycle. See Figure 12 on page 37.

A separate signal, PBREQ, is asserted by the processor to indicate to a bus arbiter that it has a pending bus transaction. This allows the bus to be allocated based on demand. The signal, PBREQ, is asserted when the write buffer is full or the CPU is doing an instruction or data fetch. The CPU is the default owner of the bus.

8- and 16-Bit Bus Modes

Any chip select (CS<5:0>) can be mapped to 8-, 16-, or 32-bit bus width, however CS4 and CS5 cannot be mapped to 8-bit bus width if the DRAM controller is enabled. Memory width for CS0 is selected at system reset with the BMODE8 and BMODE16 signals. Table 13, also below, shows the bus width options available for the CS0 area. Memory width for the CS<5:1> areas is programmed by two bits for each chip select in the Bus Width and Cacheable Register. Table 14, also below, shows the programming bits and the corresponding bus width of each chip select.

Table 13. Bus Width Control of CS0

BMODE16	BMODE8	Bus Width
0	0	Illegal
0	1	16-bit Bus
1	0	8-bit Bus
1	1	32-bit Bus

Table 14. Bus Width Control Bits of CS1 to CS5

BW1	BW0	Bus Width
0	0	32-bit Bus
0	1	8-bit Bus
1	0	16-bit Bus
1	1	Illegal

8- and 16-bit transactions are similar to 32-bit transactions except that AS is asserted only once at the beginning of the bus cycle for a load operation, and READY is asserted at the end of each byte or halfword transfer. The BE<3:0> signals indicate the byte or halfword being read or written (see Figure 15 on page 39 through Figure 18 on page 40).

For 32-bit writes to 8- or 16-bit memory and 16-bit writes to 8-bit memory, the BIU drives BE<2:3> as ADR<1:0>, and initiates multiple transactions.

When the internal DRAM controller is enabled, 8-, 16-, and 32-bit bus width are available for CS0 through CS3, and only 16- or 32-bit memory bus are available for CS4 and CS5, depending on the DRAM bus width.

Burst Mode Transactions

For systems that can support burst mode transactions, the DRAM controller can be programmed to support four-word bursts. When burst mode is enabled, BMREQ is asserted at the beginning of each bus cycle for which a burst transfer is allowed (see Table 15, also below,). If the memory system can support a burst for the current bus address, the memory system asserts BMACK to begin the burst transaction. BMACK is asserted on the first word of the burst transaction only. READY is asserted for each word of the burst. Systems that do not support burst mode for the current address should not assert BMACK. If either BMREQ or BMACK is not asserted for a transaction, only one word is transferred for each assertion of AS.

Table 15. ADR<3:2> Sequence in Burst Mode

Bus Cycle 1	Bus Cycle 2	Bus Cycle 3	Bus Cycle 4
00	01	10	11
01	00	11	10
10	11	00	01
11	10	01	00

Selection of Hyperpage Mode (EDO Mode)

By setting bit 8 of the DRAM Bank Configuration Register, hyperpage DRAM can be connected and controlled. However, if the DRAM Burst Enable Bit (bit 7 of the System Support Control Register) is clear, access timing is controlled in the same cycle as normal page-mode DRAM.

When the DRAM Burst Enable Bit is set and hyperpage is enabled, hyperpage burst access is performed. In this case, ADR<3:2> is not a value from the CPU but is generated by the DRAM controller itself. Because data is driven before it is received from the DRAM, the CAS cycle time can be shortened compared to the normal page mode burst transactions.

Basic DRAM Access Timing

Table 16. DRAM Access Timing

Bus Width	Burst Mode	Page/EDO DRAM Mode	R/W	Timing Chart
32-bit	Off	Page	Read	See Figure 19
			Write	See Figure 19
	On	Page	Read	See Figure 20
			Write	See Figure 20
		EDO	Read	See Figures 21 and 22
			Write	See Figure 22
16-bit	Off	Page	Read	See Figure 23
			Write	See Figure 23
	On	Page	Read	See Figure 24
			Write	See Figure 27(Page and EDO have same timing)
		EDO	Read	See Figures 25 and 26
			Write	See Figure 27(Page and EDO have same timing)

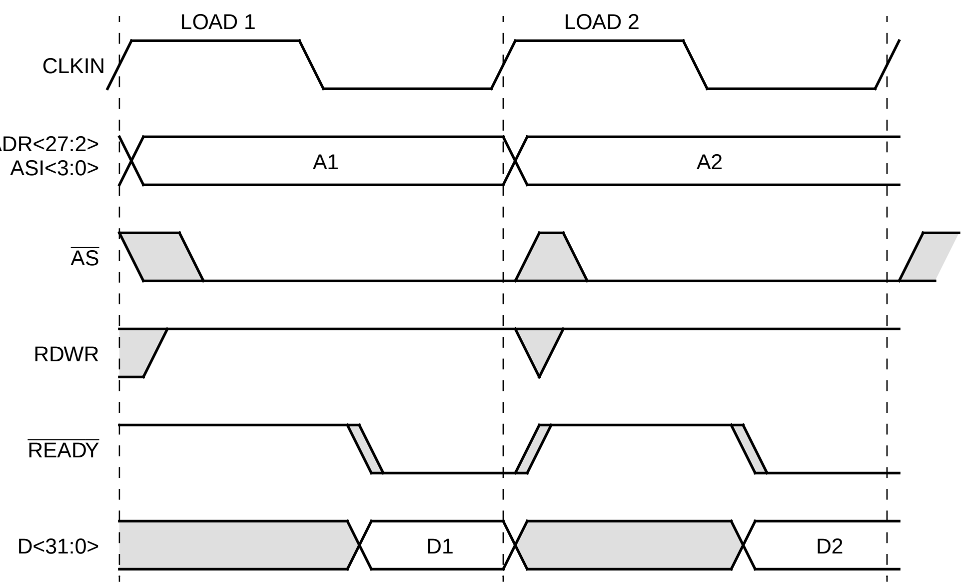


Figure 7. Typical Back-to-Back Loads (Same as Load Double)

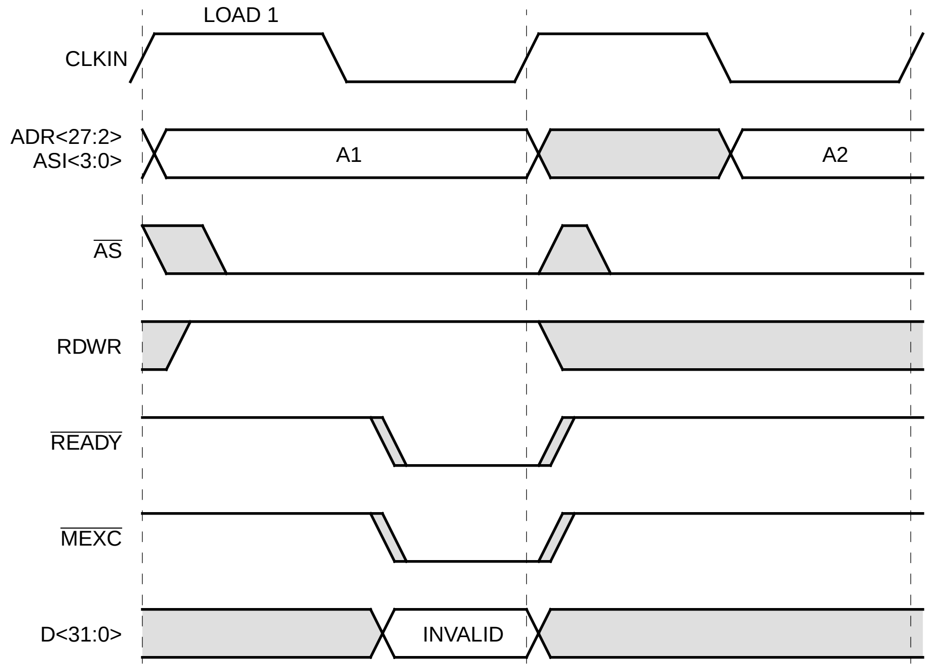


Figure 8. Load with Exception

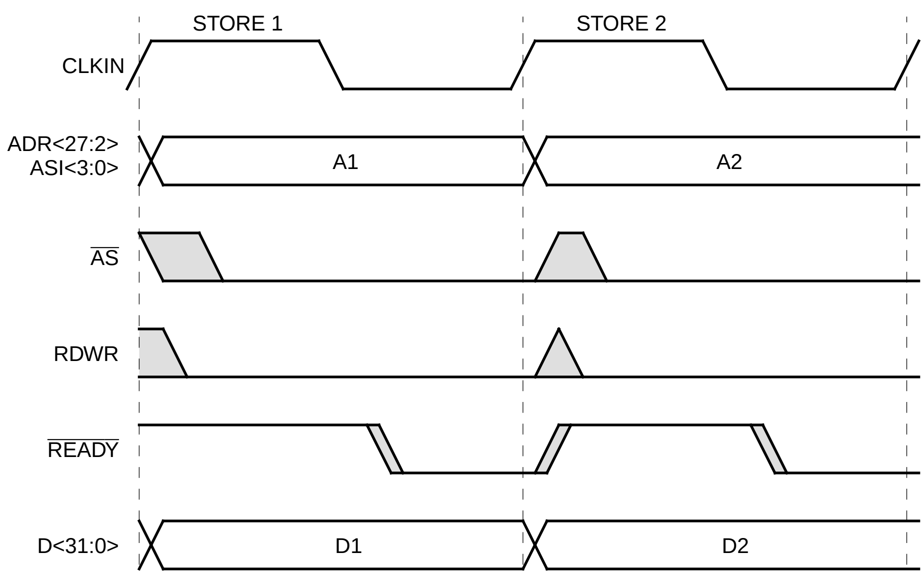


Figure 9. Typical Back-to-Back Stores (Same as Store Double)

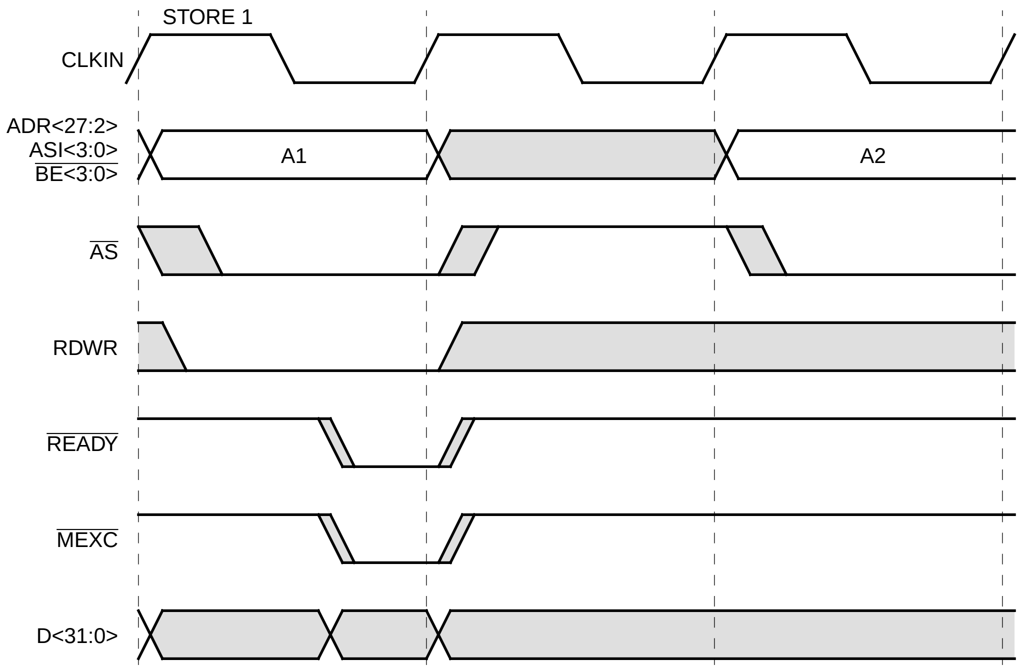
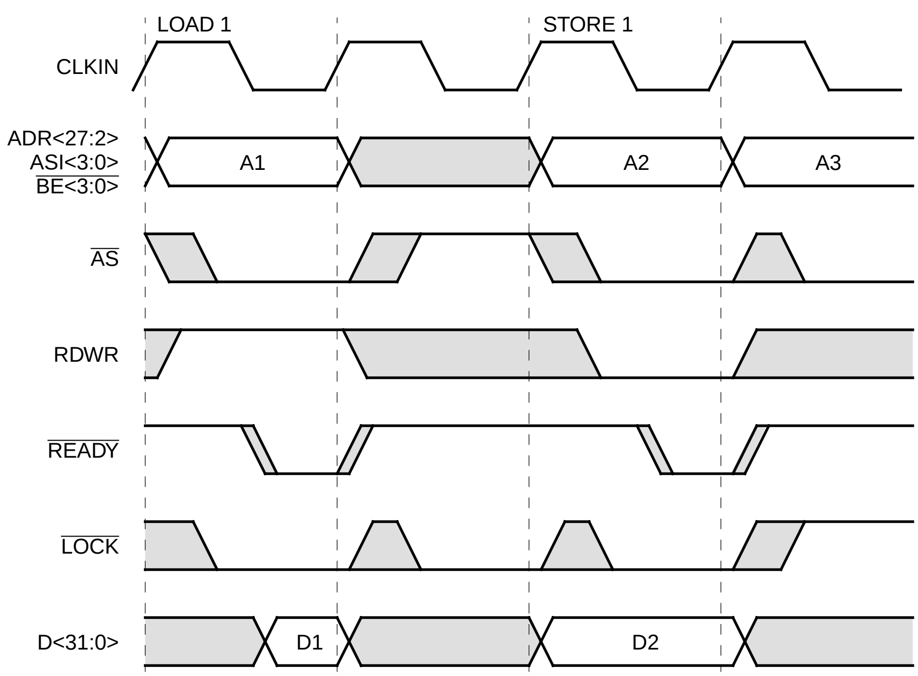


Figure 10. Store with Exception



Note: A load followed by a store requires an intervening clock cycle on the bus while a store followed by a load can occur in consecutive clock cycles.

Figure 11. Atomic Operation

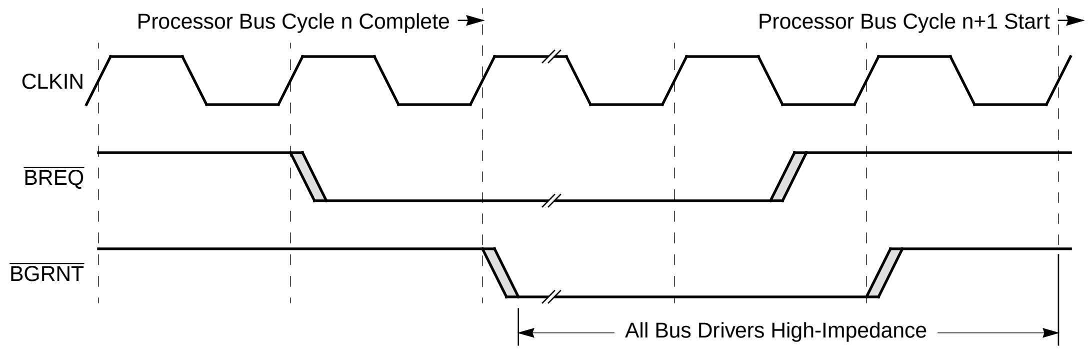


Figure 12. Bus Request and Grant Cycle

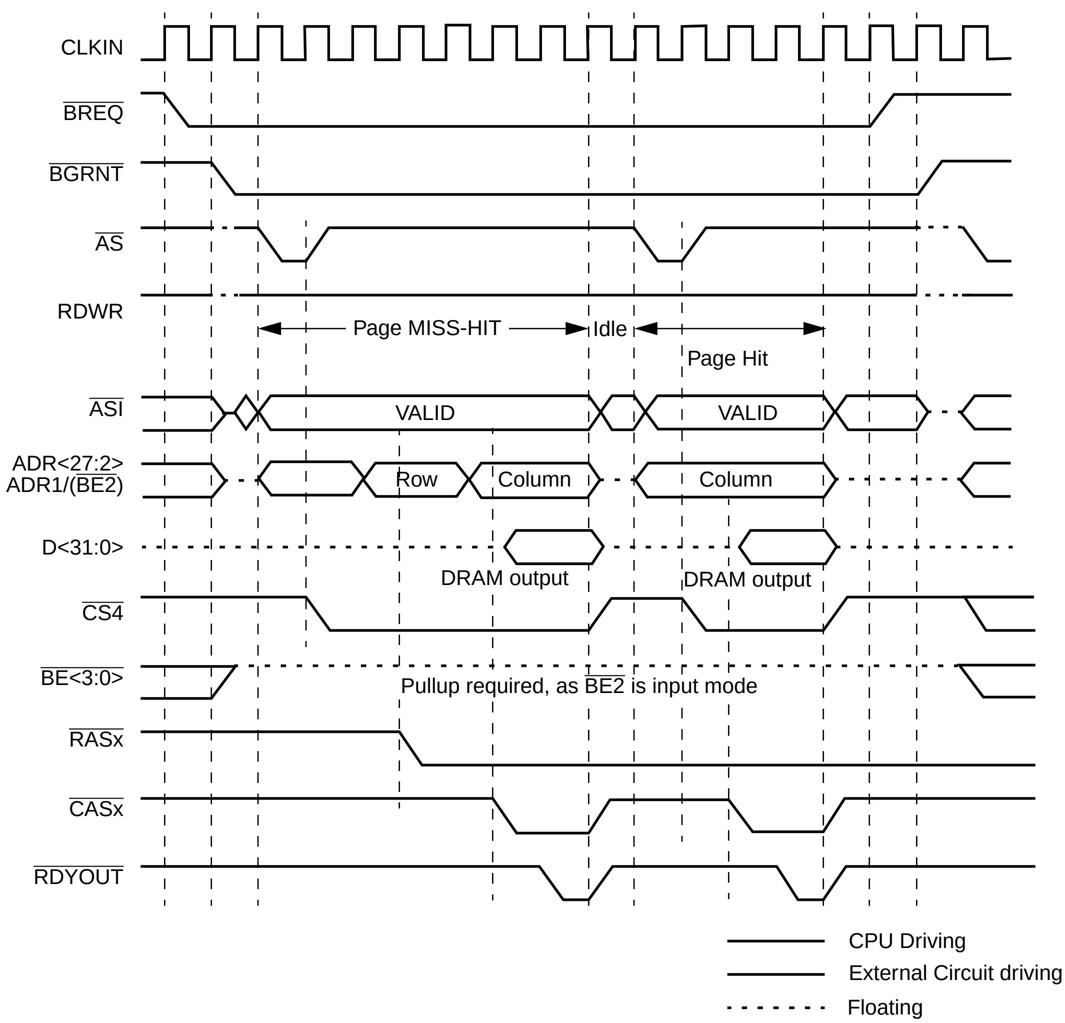


Figure 13. Timing During a DRAM Read During Bus Grant Mode

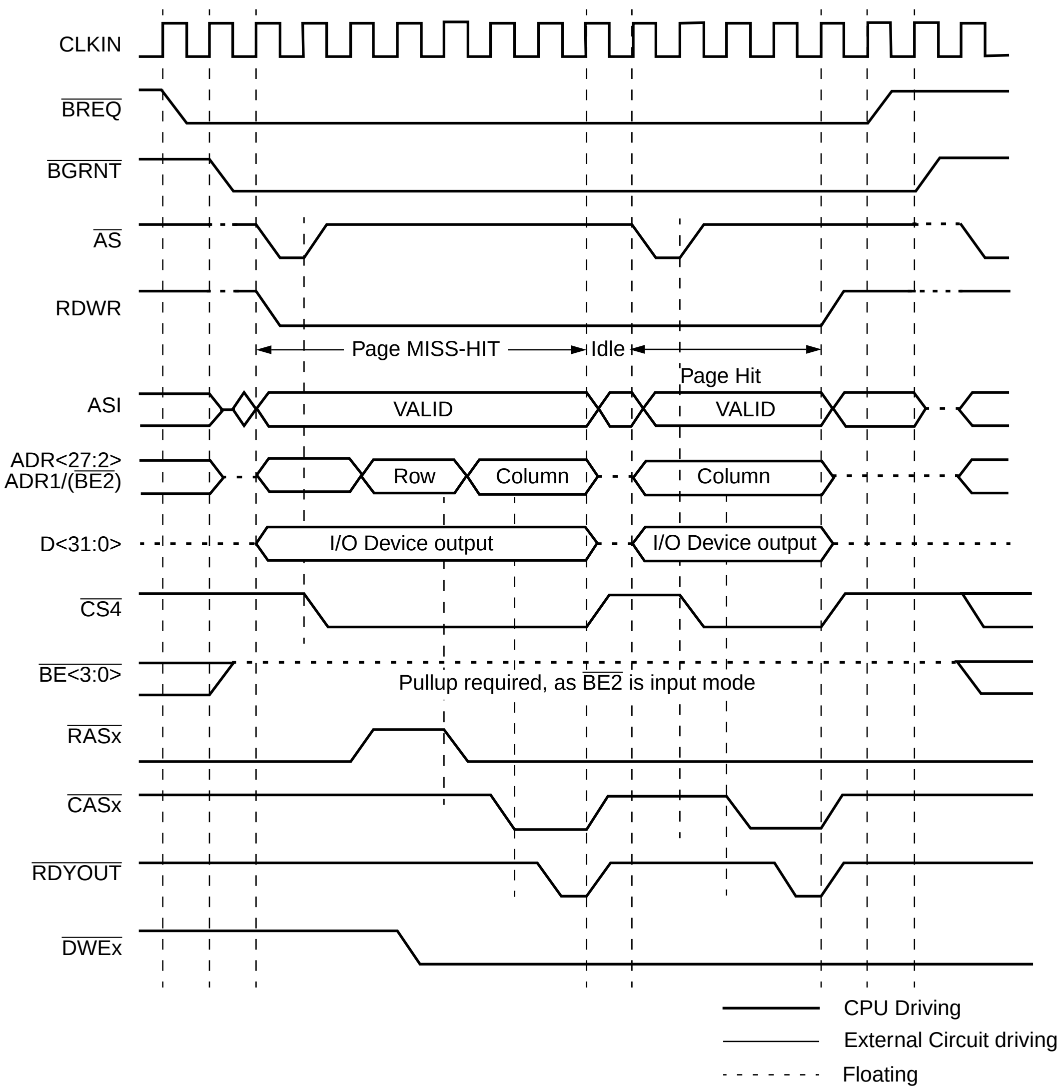


Figure 14. Timing During a DRAM Write During Bus Grant Mode

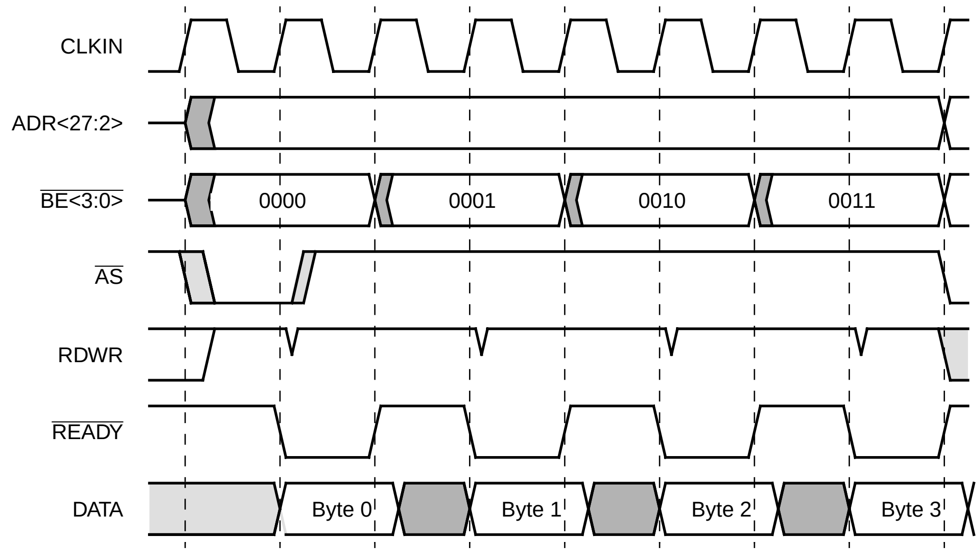


Figure 15. 8-Bit Bus Mode (1 Wait State) for Read Only

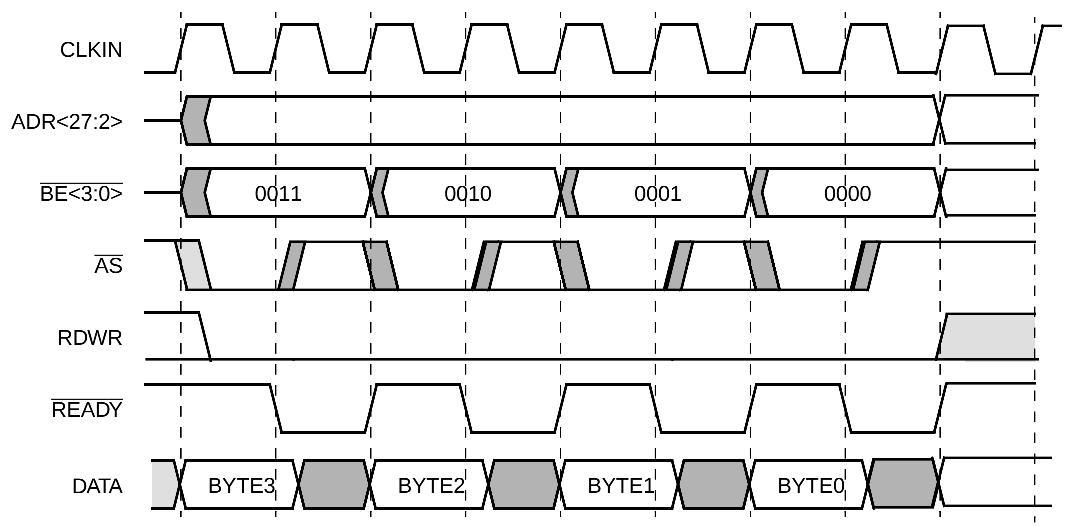


Figure 16. 8-Bit Bus Mode (1 Wait State) for Word Write

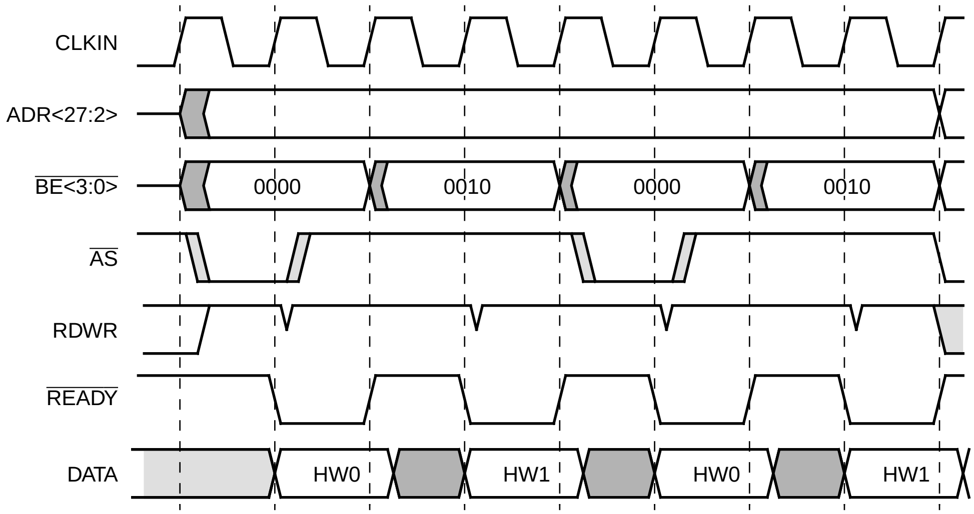


Figure 17. 16-Bit Bus Mode (1 Wait State) for Read only

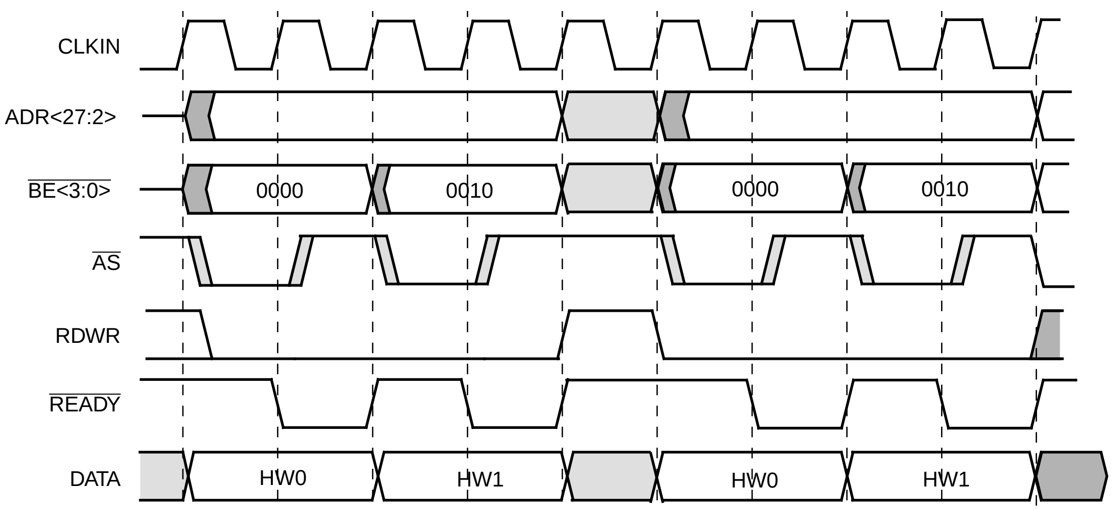
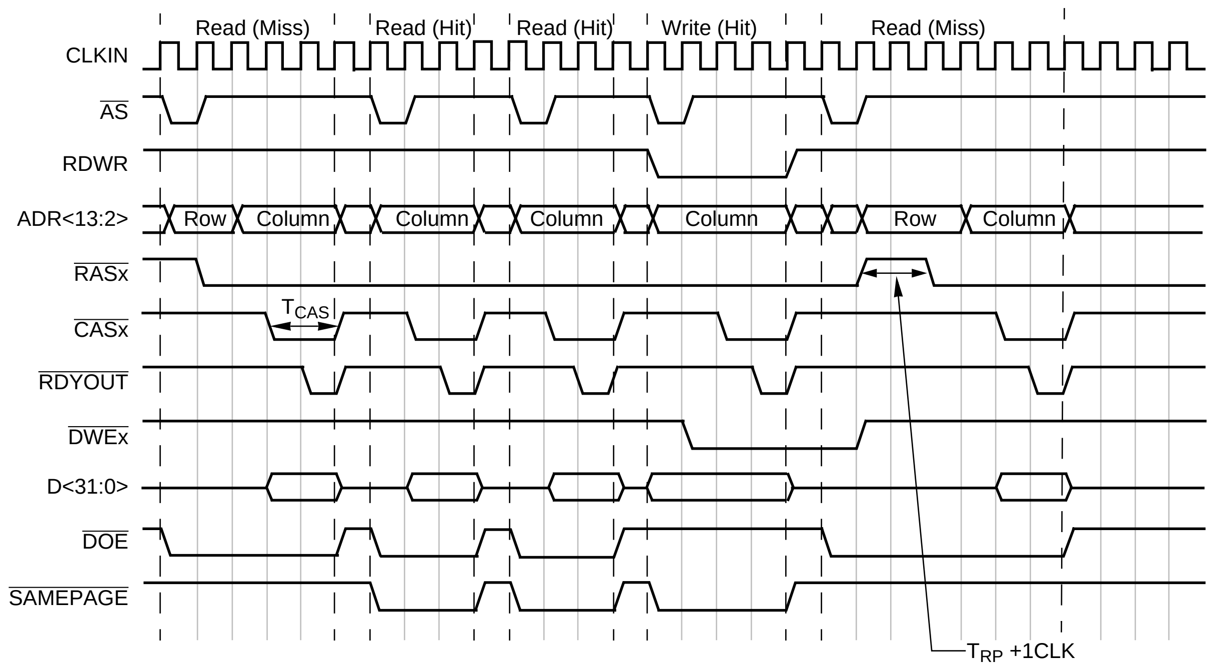


Figure 18. 16-Bit Bus Mode (1 Wait State) for Word Write



Note: During reads there are instruction fetches and loads. If the pre-fetch buffer is operating during instruction fetches, idle cycles may not open between reads.

Figure 19. DRAM Controller Enabled, DRAM Burst Disabled, 32-Bit Page Mode, Read/Write Timing

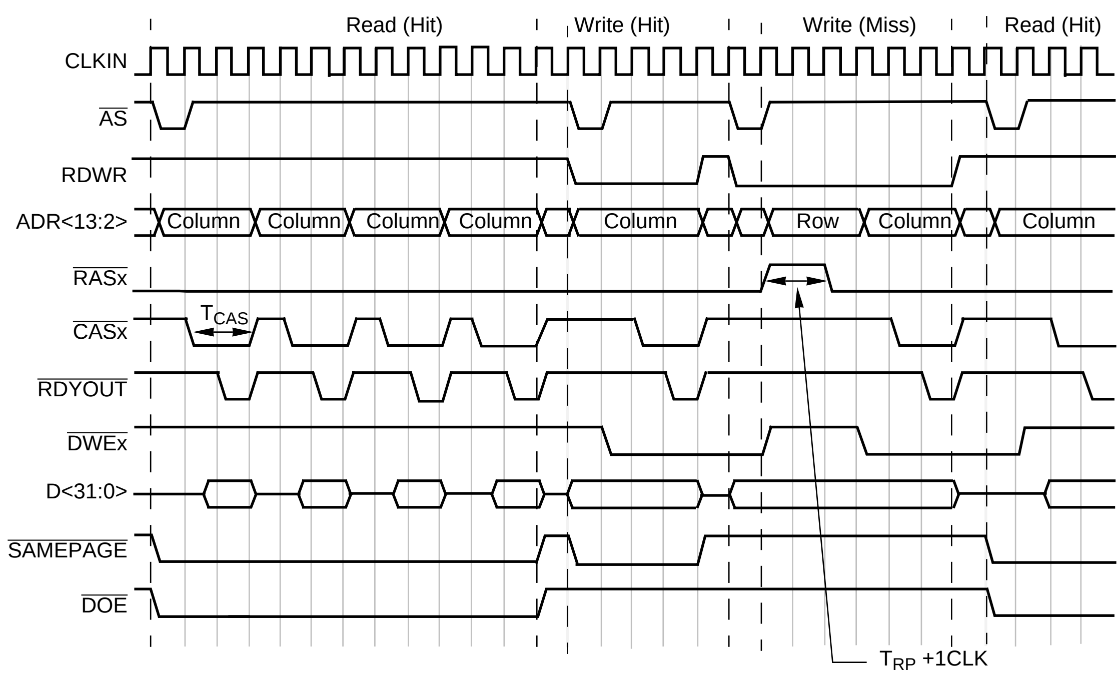


Figure 20. DRAM Controller Enabled, DRAM Burst Enabled, 32-Bit Page Mode, Read/Write Timing

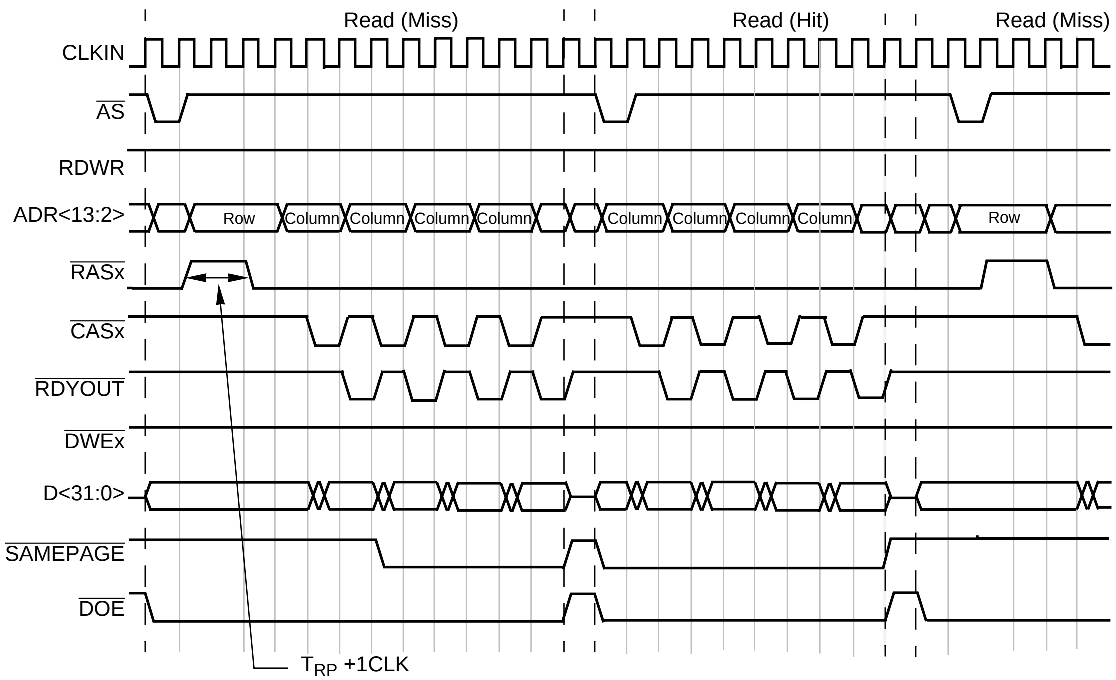
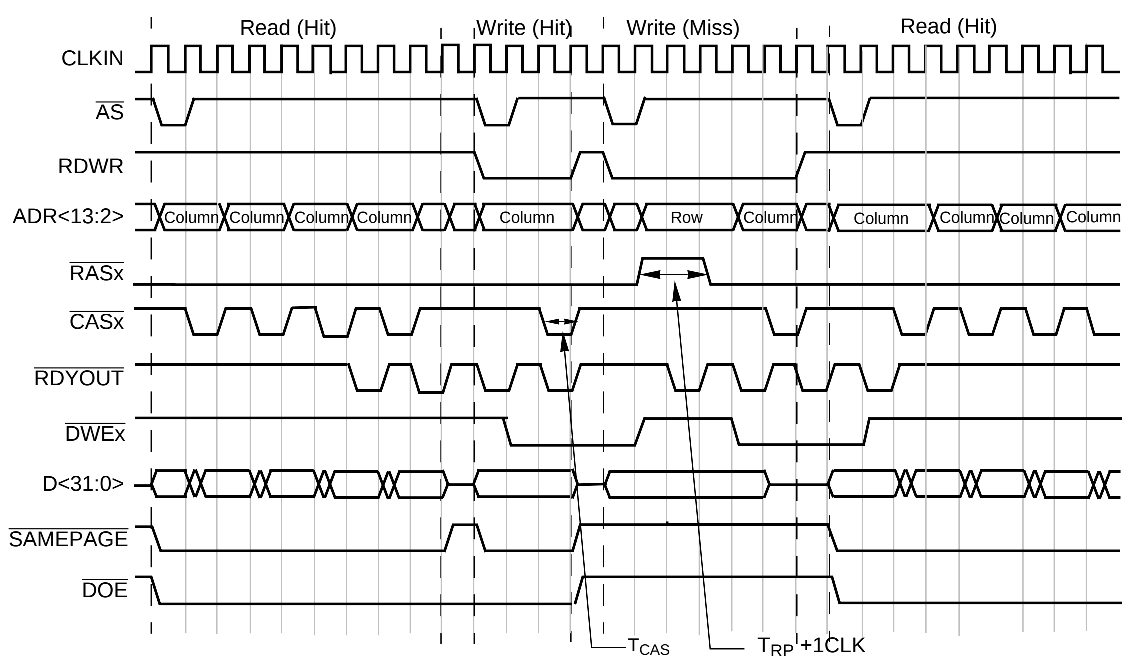
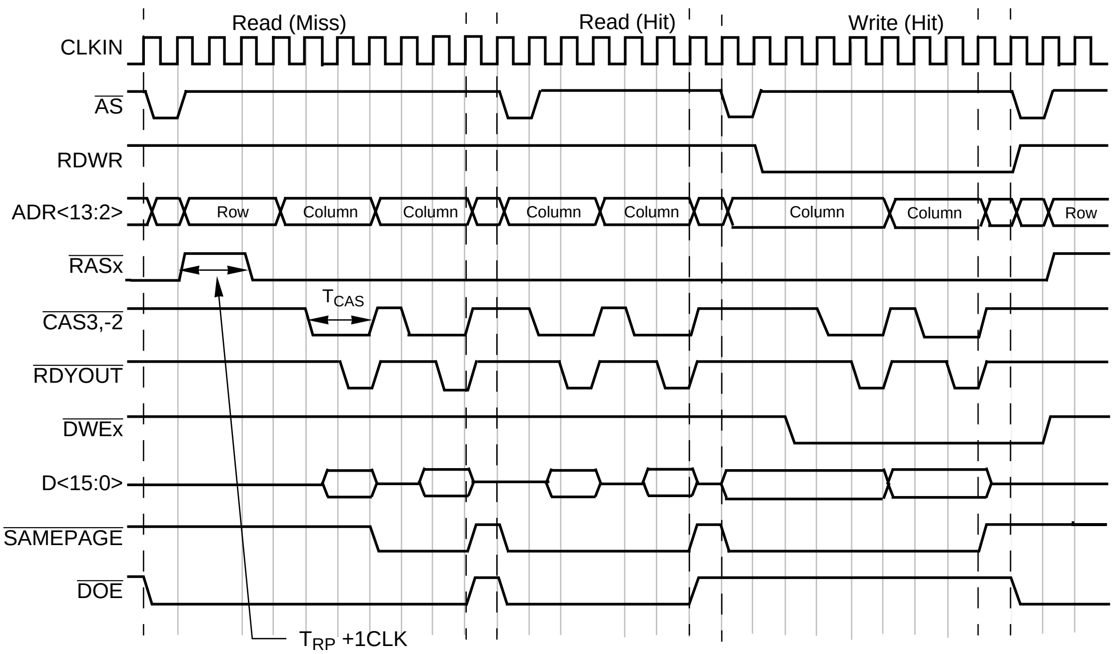


Figure 21. DRAM Controller Enabled, DRAM Burst Enabled, 32-Bit EDO, Read Timing



Note: If EDO DRAM is selected, CAS width during reads is 1 clock, and CAS width during writes is the T_{CAS} setting.

Figure 22. DRAM Controller Enabled, DRAM Burst Enabled, 32-Bit EDO, Read/Write Timing



Note: During reads there are instruction fetches and loads. If the pre-fetch buffer is working during instruction fetches, idle cycles may not open between reads.

Figure 23. DRAM Controller Enabled, DRAM Burst Disabled, 16-Bit Page Mode, Read/Write Timing

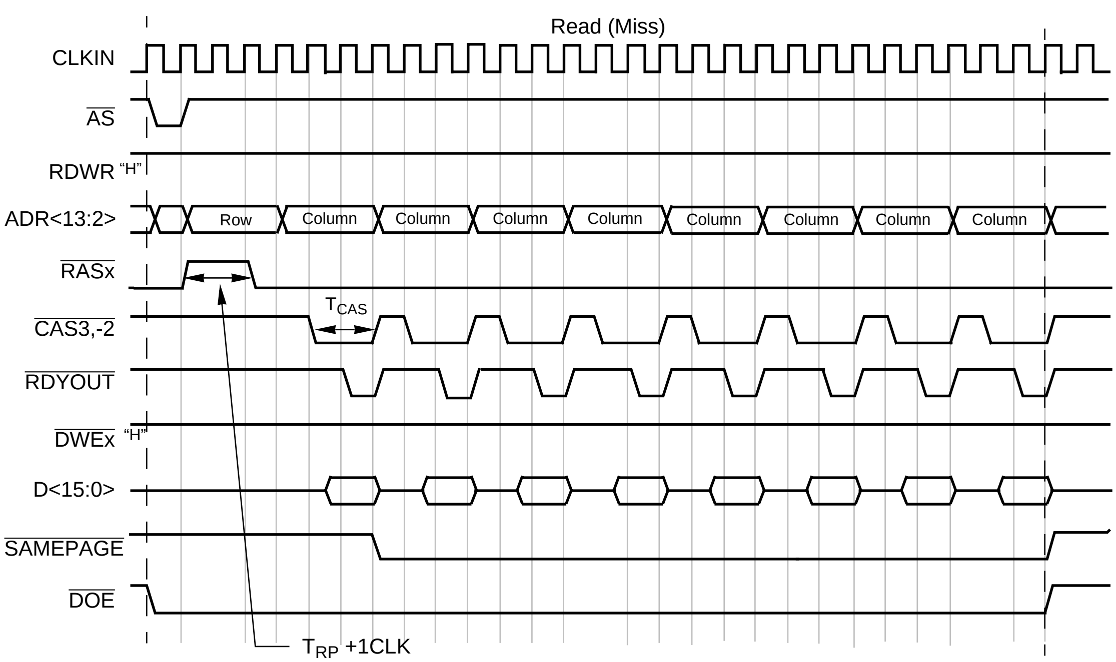


Figure 24. DRAM Controller Enabled, DRAM Burst Enabled, 16-Bit Page Mode Read Timing 1

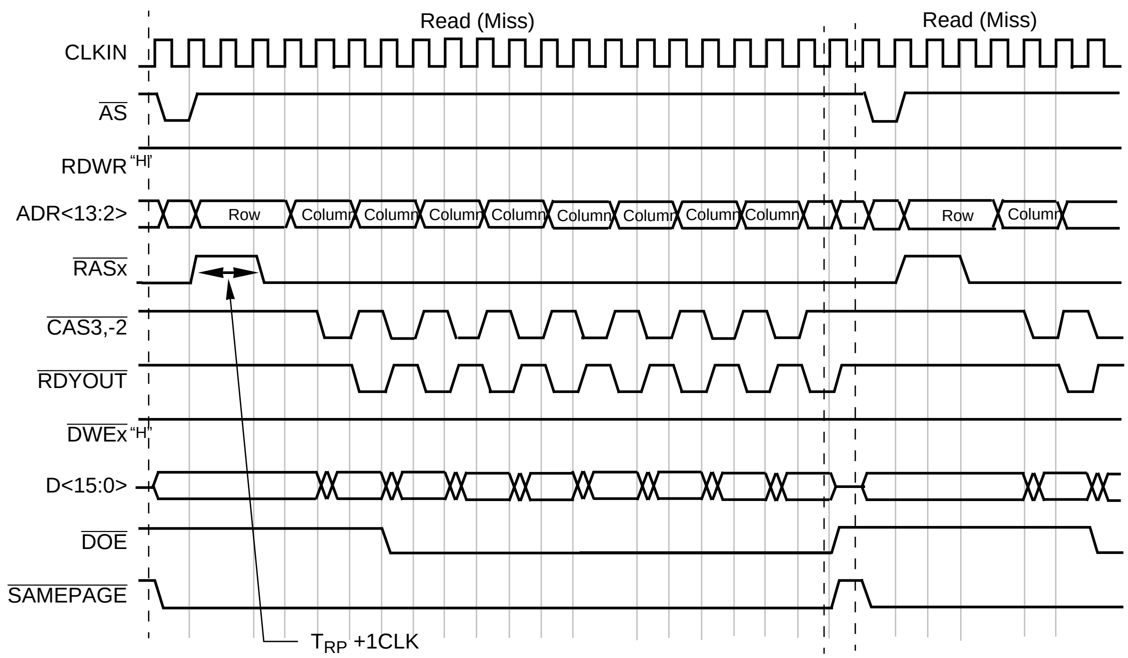


Figure 25. DRAM Controller Enabled, DRAM Burst Enabled, 16-Bit EDO, Read Timing 1

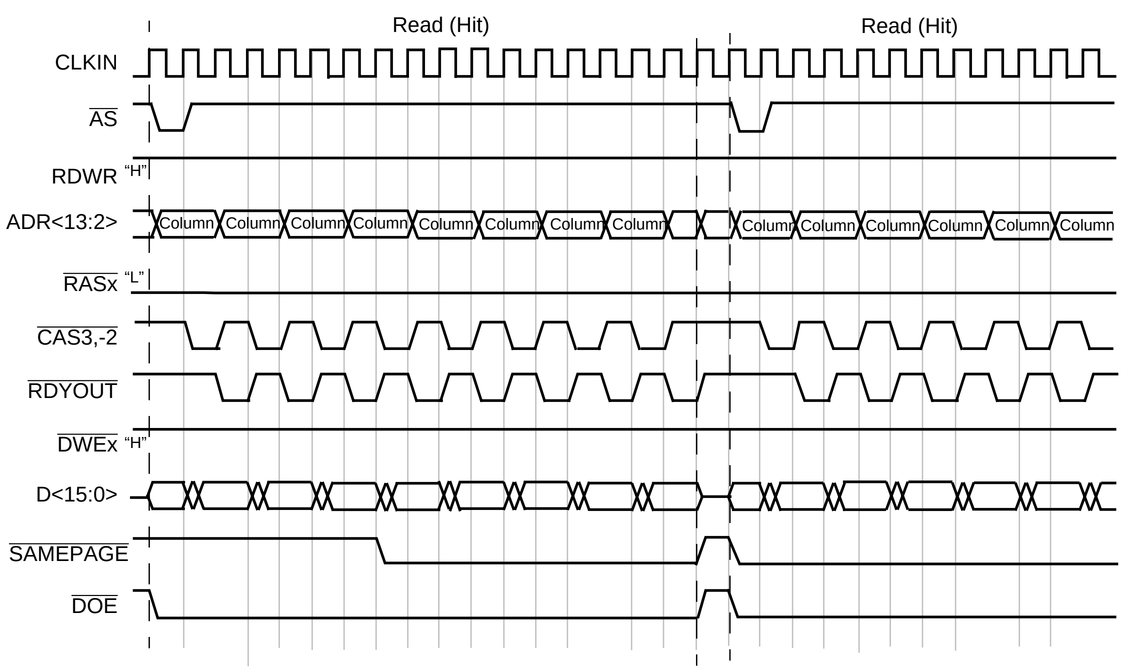


Figure 26. DRAM Controller Enabled, DRAM Burst Enabled, 16-Bit EDO, Read Timing 2

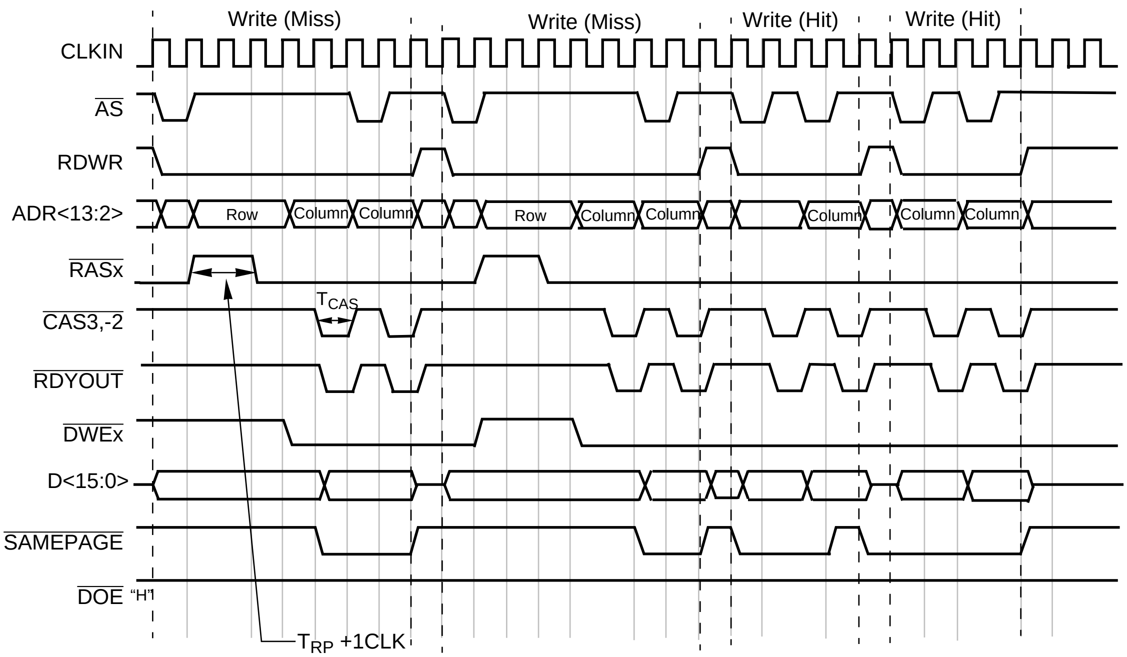


Figure 27. DRAM Controller Enabled, DRAM Burst Enabled, 16-Bit EDO, Write Timing

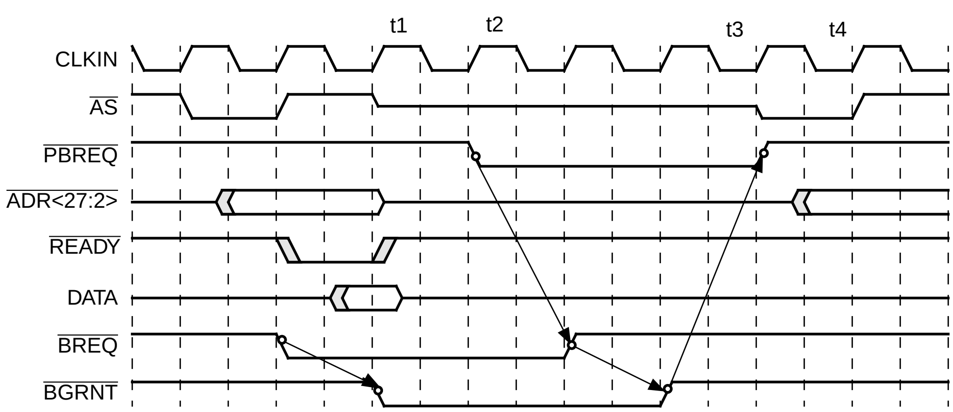


Figure 28. Example of PBREQ Timing

Electrical Characteristics

Table 17. Absolute Maximum Ratings

$V_{SS}=0V$

Symbol	Rating	Min.	Max.	Units
IO_V_{DD}	Power Supply Voltage (I/O)	-0.5	6	V
V_{DD}	Power Supply Voltage (Core)	-0.5	4	V
V_1	Input Voltage	-0.5	IO_V_{DD}+0.5	V
T_{STG}	Storage Ambient Temperature	-55	125	°C
T_{BIAS}	Temperature during Bias	0	70	°C
—	Overshoot	IO_V_{DD}+1.0V or less (50ns or less)		
—	Undershoot	V_{SS}-1.0V or less (50ns or less)		

Notes: 1. Stresses above those listed under Absolute Rating may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any other condition above those indicated in the operation section of this specification is not implied. Exposure to Absolute Maximum Ratings conditions for extended periods may affect device reliability.

Recommended Connections:

1. Power and ground connections must be made to multiple V_{DD} and V_{SS} pins. Every MB86832 based circuit board should include power (V_{DD}) and ground (V_{SS}) planes for power distribution. Every V_{DD} pin must be connected to the power plane, and every V_{SS} pin must be connected to the ground plane. Pins identified as "N.C." must not be connected in the system.
2. Liberal decoupling capacitance should be placed near the MB86832. The processor can cause transient power surges when multiple output buffers transition, particularly when connected to large capacitive loads.
3. Low inductance capacitors and interconnections are recommended for best high frequency electrical performance. Inductance can be reduced by shortening the board traces between the processor and decoupling capacitors as much as possible. Capacitors specifically designed for the QFP package offer the least inductance.
4. For reliable operation, alternate bus masters must drive any pins that are floated by the MB86833 when it has granted the bus, in particular the $\overline{\text{LOCK}}$, ADR<27:2>, ASI<3:0>, $\overline{\text{BE<3:0>}}$, D<31:0>, $\overline{\text{AS}}$ and RDWR signals must be driven by alternate bus masters. These pins are normally driven by the processor during active and idle bus states and don't require external pullups (except for $\overline{\text{BE2}}$, which becomes an address input). N.C. pins must always remain unconnected.

Table 18. Recommended Operating Conditions

Symbol	Rating	Min.	Typ.	Max.	Units
IO_V_{DD}	Power Supply Voltage (I/O = 5V)	4.75	5.0	5.25	V
	Power Supply Voltage (I/O = 3.3V)	3.0	3.3	3.6	V
V_{DD}	Power Supply Voltage (Core)	3.0	3.3	3.6	V
V_{IL}	Low Level Input Voltage	0	-	$V_{DD}\times0.25$	V
V_{IH}	High Level Input Voltage	$V_{DD}\times0.65$	-	IO_V_{DD}	V
T_{opr}	Operating Temperature	0	25	70	°C

- The MB86832 can be used with a 5V system or a 3.3V system interface:

5V System Interface	IO_V_{DD} = 5V and V_{DD} = 3.3V	Two Power Supplies
3.3V System Interface	IO_V_{DD} = V_{DD} = 3.3V	Single Power Supply

- Because 5V inputs cannot be accepted using the 3.3V system interface, all input signals must be driven at 3.3V logic levels.
- With the 5V system interface, outputs are driven with 5V logic levels. The input is defined by 3.3V logic levels, but inputs greater than 3.3V can also be tolerated. The CPU and core logic power supply current is the same for both 3.3V and 5V operation.
- In the 5V system interface, two power supplies are required. While there are basically no restrictions on power supply sequencing, the following sequence is recommended:

At power-on	V_{DD}	IO_V_{DD}	Signal
At power-off	Signal	IO_V_{DD}	V_{DD}

DC Characteristics

5V Interface IO_V_{DD} = 5V ± 5% V_{DD} = 3.3V ± 0.3V, V_{SS} = 0V, Ta = 0~70°C

Symbol	Rating	Conditions	Min.	Typ.	Max.	Units
V_{IL}	Low Level Input Voltage		0	-	VDD×0.25	V
V_{IH}	High Level Input Voltage		V_{DD}×0.65	-	IO_V_{DD}	V
V_{OL}	Low Level Output Voltage	I_{OL} = 4 mA	0	-	0.4	V
V_{OH}	High Level Output Voltage	I_{OH} = -4 mA	IO_V_{DD}-0.5	-	IO_V_{DD}	V
I_{LI}	Input Leak Current	V_{IN} = 0 or IO_V_{DD}	-10	-	10	µA
I_{LZ}	Floating Output Leakage Current	V_{OUT} = 0 or IO_V_{DD}	-10	-	10	µA
I_{DD}	I/O Power Supply Current (IO_V_{DD})	33 MHz No Load	-	40	50	mA
I_{DD}	I/O Power Supply Current (IO_V_{DD})	40 MHz No Load	-	50	62.5	mA
	CPU/Core Power Supply Current (V_{DD})	66 MHz	-	200	250	mA
	CPU/Core Power Supply Current (V_{DD})	80 MHz	-	250	312.5	mA
	CPU/Core Power Supply Current (V_{DD})	100 MHz	-	300	375	mA
I_{SLEEP}	CPU/Core Power Supply Current (V_{DD}) in Sleep Mode	66 MHz	-	10	-	mA
P_D	Power Consumption (IO_V_{DD}+V_{DD})	66 MHz No Load	-	1200	1500	mW
C_{PIN}	Pin Capacity	IO_VDD = V_1 = 0 f = 1 MHz	-	-	16	pF

3.3V Interface IO_VDD = VDD = 3.3V ± 0.3V, VSS = 0V, Ta = 0~70∞C

Symbol	Rating	Conditions	Min.	Typ.	Max.	Units
V_{IL}	Low Level Input Voltage		0	-	V_{DD}×0.25	V
V_{IH}	High Level Input Voltage		V_{DD}×0.65	-	IO_V_{DD}	V
V_{OL}	Low Level Output Voltage	I_{OL} = 2.0 mA	0	-	0.4	V
V_{OH}	High Level Output Voltage	I_{OH} = -2.0 mA	IO_V_{DD}-0.5	-	IO_V_{DD}	V
I_{LI}	Input Leak Current	V_{IN} = 0 or IO_V_{DD}	-10	-	10	µA
I_{LZ}	Floating Output Leakage Current	V_{OUT}=0 or IO_V_{DD}	-10	-	10	µA
I_{DD}	I/O Power Supply Current (IO_V_{DD})	33 MHz No Load	-	30	37.5	mA
I_{DD}	I/O Power Supply Current (IO_V_{DD})	40 MHz No Load	-	36	45	mA
I_{SLEEP}	Power Supply Current (V_{DD}) in Sleep Mode	66 MHz	-	10	-	mA
P_D	Power Consumption (IO_V_{DD}+V_{DD})	66 MHz No Load	-	759	949	mW
C_{PIN}	Pin Capacity	IO_V_{DD} = V_1 = 0 f = 1 MHz	-	-	16	pF

Alternating current characteristics are all regulated by CLKIN (BIU clock), and they do not depend on the CPU internal operating frequency.

AC Characteristics

V_{DD} = 3.3V ± 0.3V, V_{SS} = 0V, Ta = 0~70°C, P=Period (CLKIN period)

Symbol	Parameter Description		IO_V_{DD} = 5V ± 5%		IO_V_{DD} = 3.3V ± 0.3V		Units
			Min.	Max.	Min.	Max.	
t1	CLKIN period		30/25 [1]	100	30/25 [1]	100	ns
t2	CLKIN high time		10/8 [1]	-	10/8 [1]	-	ns
t3	CLKIN low time		10/8 [1]	-	10/8 [1]	-	ns
t4	CLKIN rise time		-	3	-	3	ns
t5	CLKIN fall time		-	3	-	3	ns
t6	D<31:0>	Output valid delay	-	20/16 [1]	-	20	ns
		Output hold	2	-	2	-	ns
	ADR<27:2>	Output valid delay	-	20/18 [1]	-	21	ns
		Output hold	2	-	2	-	ns
	$\overline{BE}$<3:0>	Output valid delay	-	20/16 [1]	-	21	ns
		Output hold	2	-	2	-	ns
	ASI<3:0>	Output valid delay	-	20/16 [1]	-	21	ns
		Output hold	2	-	2	-	ns
	$\overline{CS}$<5:0>	Output valid delay	-	20/16 [1]	-	21	ns
		Output hold	2	-	2	-	ns
t7	RDWR	Output valid delay	-	18/14 [1]	-	19	ns
		Output hold	2	-	2	-	ns
	$\overline{LOCK}$	Output valid delay	-	18/14 [1]	-	19	ns
		Output hold	2	-	2	-	ns
	$\overline{AS}$	Output valid delay	-	18/14 [1]	-	19	ns
		Output hold	2	-	2	-	ns
	$\overline{TIMER\_OVF}$	Output valid delay	-	20/16 [1]	-	21	ns
		Output hold	2	-	2	-	ns
	$\overline{BGRNT}$	Output valid delay	-	18/14 [1]	-	19	ns
		Output hold	2	-	2	-	ns
	$\overline{PBREQ}$	Output valid delay	-	18/14 [1]	-	19	ns
		Output hold	2	-	2	-	ns
	$\overline{RDYOUT}$ (Internal READY mode)	Output valid delay	-	20/16 [1]	-	21	ns
		Output hold	2	-	2	-	ns
	$\overline{RDYOUT}$ (External READY mode)	Output valid delay	-	15/14 [1]	-	15	ns
		Output hold	2	-	2	-	ns
	$\overline{ERROR}$	Output valid delay	-	20/14 [1]	-	21	ns
		Output hold	2	-	2	-	ns
	$\overline{PDOWN}$	Output valid delay	-	20/14 [1]	-	21	ns
		Output hold	2	-	2	-	ns

AC Characteristics (continued)

Symbol	Parameter Description		$IO\_V_{DD} = 5V \pm 5\%$		$IO\_V_{DD} = 3.3V \pm 0.3V$		Units
			Min.	Max.	Min.	Max.	
t8	$\overline{\text{SAMEPAGE}}$	Output valid delay	-	20/16 [1]	-	21	ns
		Output hold	2	-	2	-	ns
	$\overline{\text{TIMER\_OVF}}$	Output valid delay	-	20/14 [1]	-	21	ns
		Output hold	2	-	2	-	ns
	$\overline{\text{BMREQ}}$	Output valid delay	-	18/16 [1]	-	19	ns
		Output hold	2	-	2	-	ns
t9	$\overline{\text{RAS}<3:0>}$	Output valid delay	-	15/12 [1]	-	15	ns
		Output hold	2	-	2	-	ns
	$\overline{\text{CAS}<3:0>}$	Output valid delay	-	15/12 [1]	-	15	ns
		Output hold	2	-	2	-	ns
	$\overline{\text{DWE}<3:0>}$	Output valid delay	-	15/12 [1]	-	15	ns
		Output hold	2	-	2	-	ns
	$\overline{\text{DOE}}$	Output valid delay	-	15/12 [1]	-	15	ns
		Output hold	2	-	2	-	ns
t10	$\overline{\text{READY}}$	Input setup time	14/10 [1]	-	14/10 [1]	-	ns
		Input hold time	2	-	2	-	ns
	$\overline{\text{MEXC}}$	Input setup time	14/10 [1]	-	14/10 [1]	-	ns
		Input hold time	2	-	2	-	ns
	$\overline{\text{BREQ}}$	Input setup time	12/10 [1]	-	12/10 [1]	-	ns
		Input hold time	2	-	2	-	ns
	$\overline{\text{BMACK}}$	Input setup time	12/10 [1]	-	12/10 [1]	-	ns
		Input hold time	2	-	2	-	ns
	RDWR	Input setup time	12	-	12	-	ns
		Input hold time	2	-	2	-	ns
	$\overline{\text{AS}}$	Input setup time	12	-	12	-	ns
		Input hold time	2	-	2	-	ns
t11	$\overline{\text{ASI}<3:0>}$	Input setup time	12	-	12	-	ns
		Input hold time	2	-	2	-	ns
	ADR<27:2>	Input setup time	12	-	12	-	ns
		Input hold time	2	-	2	-	ns
	$\overline{\text{BE2}}$	Input setup time	12	-	12	-	ns
		Input hold time	2	-	2	-	ns
t12	D<31:0>	Input setup time	14/12 [1]	-	14/12 [1]	-	ns
		Input hold time	2	-	2	-	ns
	IRL<3:0>	Input setup time	Asynchronous		Asynchronous		
		Input hold time	Asynchronous		Asynchronous		
	$\overline{\text{WKUP}}$	Input setup time	Asynchronous		Asynchronous		
		Input hold time	Asynchronous		Asynchronous		

AC Characteristics (continued)

Symbol	Parameter Description	IO_V$_{DD}$ = 5V ± 5%		IO_V$_{DD}$ = 3.3V ± 0.3V		Units
		Min.	Max.	Min.	Max.	
	IRQ<15:8> Input setup time	Asynchronous		Asynchronous		ns
	IRQ<15:8> Input hold time	Asynchronous		Asynchronous		ns
	IRQ<15:8> Input High level duration	2×P+10	—	2×P+10	—	ns
	IRQ<15:8> Input Low level duration	2×P+10	—	2×P+10	—	ns

Notes:
1. The first value is for 33 MHz bus operation, and the second value is for 40 MHz bus operation.
2. In the absence of documentation to the contrary, all parameters are valid within the specified temperature and power supply ranges. These specifications are subject to change for performance improvement.
3. All voltage values have the GND (V_{SS} = 0V) as their reference. Timing measurement reference points are 1.5V, input level from 0.4V to 2.4V, and input rise and fall times are 2ns or less.
4. Do not short multiple output pins for more than 1 second. External output load capacity is 30 pF.
5. Pins which are specified for asynchronous input, and standards other than $\overline{\text{RDYOUT}}$ during external READY mode, are referenced from the external clock (CLKIN) rising edge.
6. $\overline{\text{RDYOUT}}$ during external READY mode is referenced from the $\overline{\text{READY}}$ input.
7. A minimum of four CLKIN Cycles are required for reset initialization. Also, 4,000 CLKIN cycles are required for PLL oscillation stabilization time.

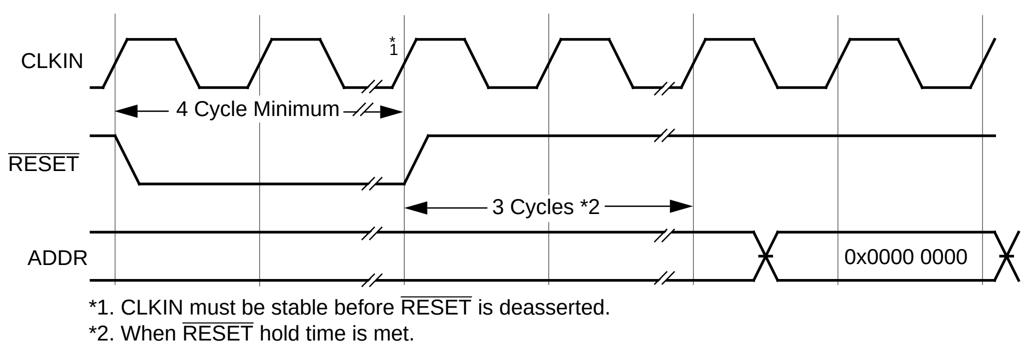


Figure 29. Reset Timing

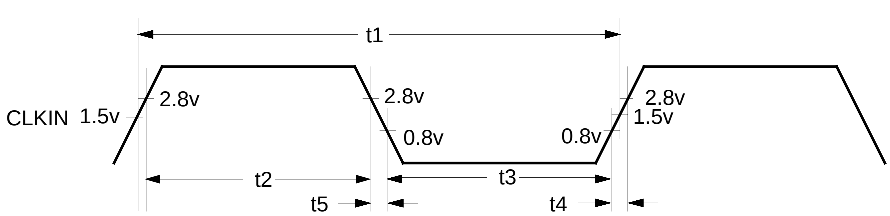


Figure 30. Clock Timing

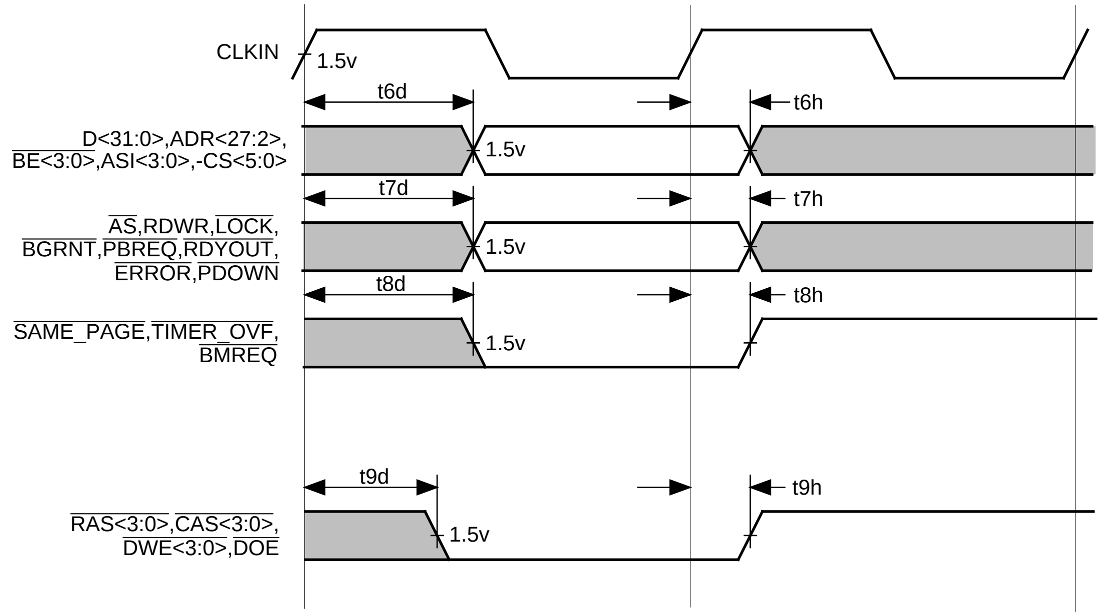
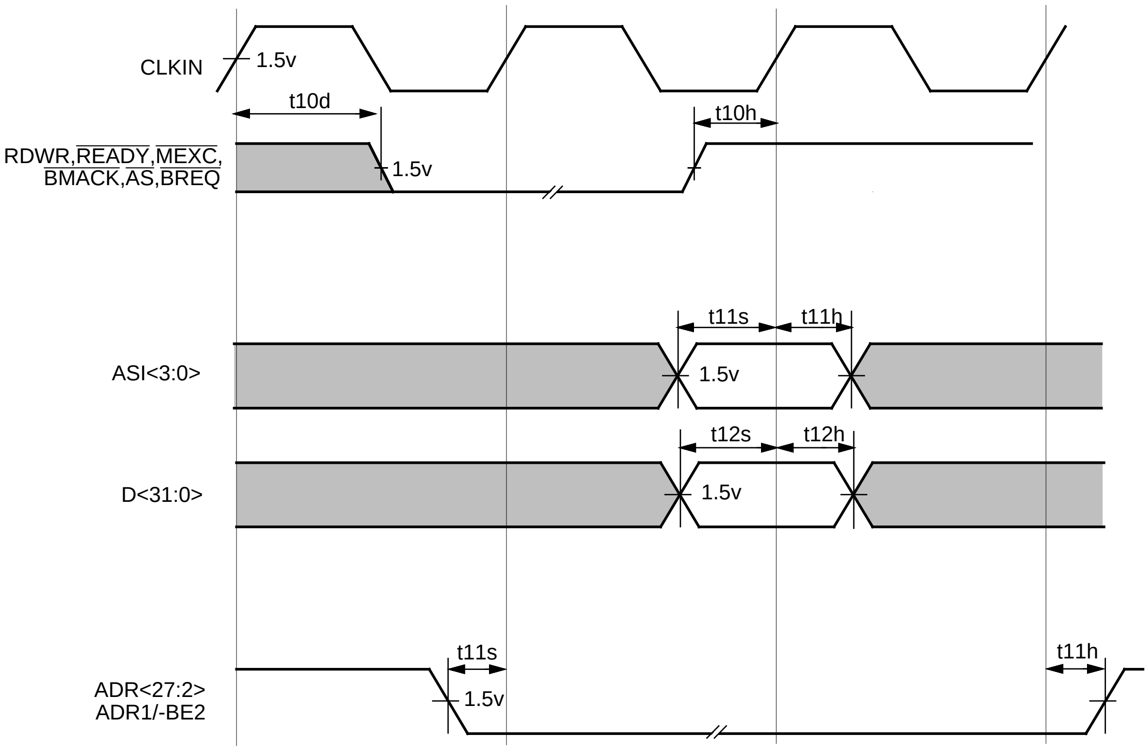

CLKIN
1.5v
t6d
t6h
D<31:0>,ADR<27:2>,
BE<3:0>,ASI<3:0>,-CS<5:0>
1.5v
t7d
t7h
AS,RDWR,LOCK,
BGRNT,PBREQ,RDYOUT,
ERROR,PDOWN
1.5v
t8d
t8h
SAME_PAGE,TIMER_OVF,
BMREQ
1.5v
t9d
t9h
RAS<3:0>,CAS<3:0>,
DWE<3:0>,DOE
1.5v

Figure 31. I/O Output Timing

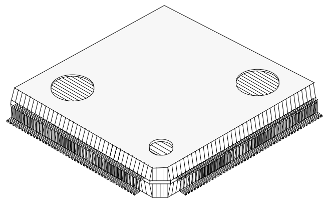


NOTE: s=setup, h=hold

Figure 32. I/O Input Timing

Exterior Package Drawing

Plastic. SQFP, 176 Pin **EIAJ Code: *QFP176-P-2424-1**

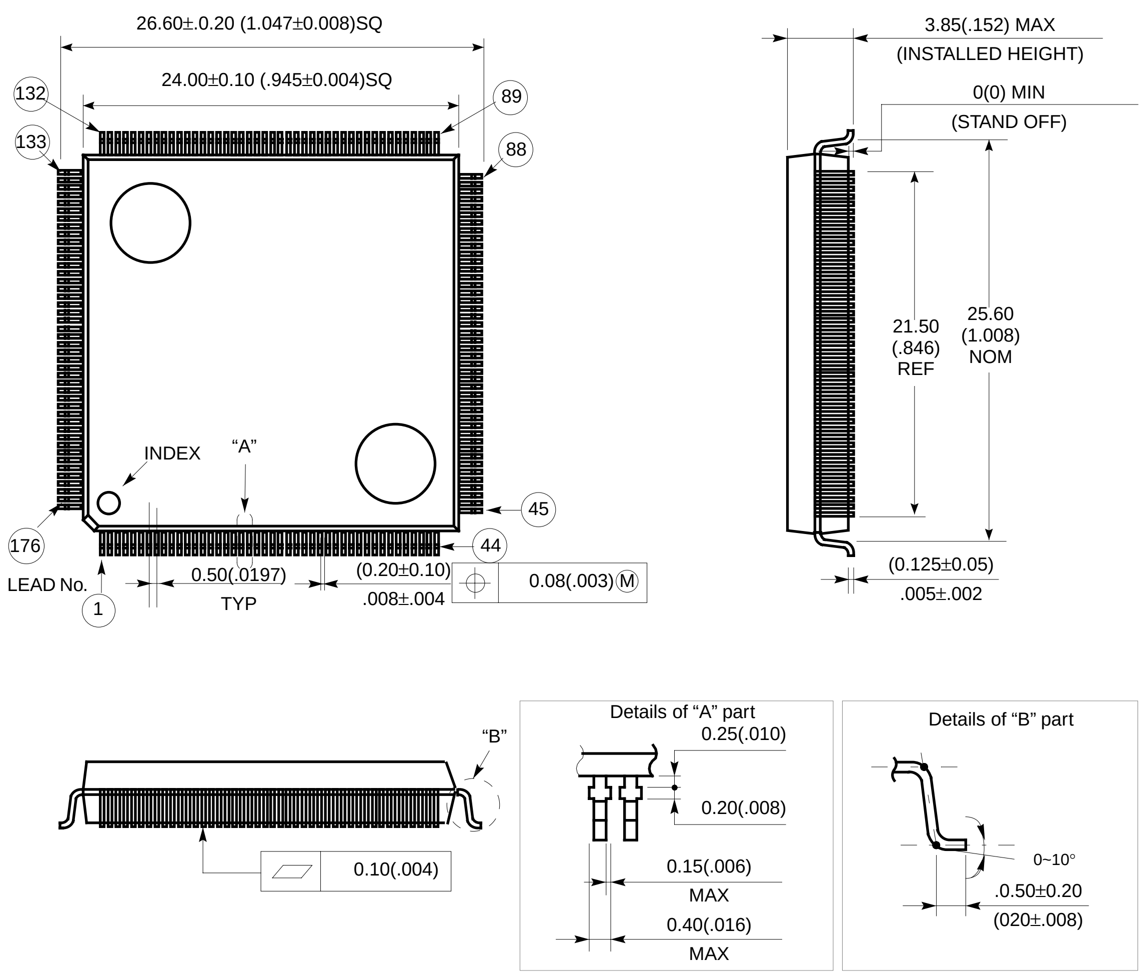


(FPT-176P-M01)

Lead Pitch	0.50mm
Package Width × Package Length	24 × 24mm
Lead Profile	Gull Wing
Sealing Method	Plastic Mold

ORDERING INFO: MB86832-66/80/100 PFV-G

176-LEAD PLASTIC QFP PACKAGE
(CASE No.: FPT-176P-M01)



Dimensions in inches (millimeters)

Printed in U.S.A.

EC-DS-20501-8/98

FUJITSU MICROELECTRONICS, INC.

Corporate Headquarters
3545 North First Street, San Jose, California 95134-1804
Tel: (800) 866-8608 Fax: (408) 922-9179
E-mail: fmicrc@fmi.fujitsu.com Internet: http://www.fujitsumicro.com