FUJITSU

GENERAL PURPOSE DIGITAL SIGNAL PROCESSOR

MB 8764

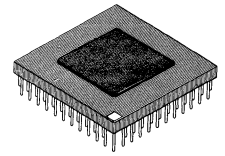
December 1985
Edition 2.0

GENERAL PURPOSE DIGITAL SIGNAL PROCESSOR

The Fujitsu MB 8764 is a general purpose silicon-gate CMOS digital signal processor (DSP) integrated circuit. The MB 8764 features a high-speed pipelined multiplier, supports concurrent operations with compound instructions and multiple data paths, offers flexible and expandable memory options and has an on-chip DMA channel.

With its high-speed operation, the MB 8764 gives high throughput in various applications, such as telecommunications, signal processing and image processing.

Being packaged in the 88-pin pin grid array, the MB 8764 allows a complex system to be built with the external program ROM and data RAM accessed through dedicated address and data buses.

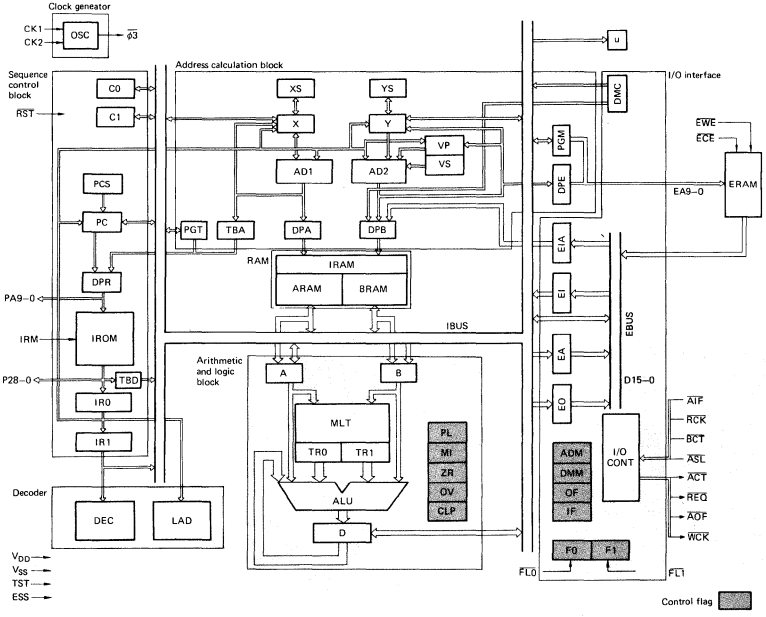


CERAMIC PACKAGE
RIT-88C-A01

This device contains circuitry to protect the inputs against damage due to high static voltages or electric fields. However, it is advised that normal precautions be taken to avoid application of any voltage higher than maximum rated voltages to this high impedance circuit.

- General purpose high-speed digital signal processing
- High speed operation
  - 100ns cycle time
- Parallel pipelined multiply function
  - 16 bits x 16 bits → 26 bits
- Divide function
  - 26 bits ÷ 16 bits → 16 bits
- Program ROM
  - 1024 words x 24 bits
  - Internal (mask-programmed) and external ROM selectable
- Part of the program ROM can be used for constant data storage
- Two built-in 128 x 16 bits RAMs
- Expansion RAM function
  - Expandable up to 1024 words x 16 bits
  - Two access speed rates can be selected
- Numerous I/O functions
  - 16-bit parallel interface
  - Three input modes and two output modes including DMA
- Powerful instruction set using compound instructions
  - One level of subroutine nesting (multi-level nesting can be programmed)
  - Two levels of loop nesting (multi-level nesting can be programmed)
  - Compound instructions (for example, an arithmetic/logic instruction combined with a move instruction) enable concurrent processing
  - 15 arithmetic/logic instructions
- Addressing
  - Direct addressing
  - Indexed addressing
  - Immediate addressing
  - Virtual shift addressing
- Silicon-gate CMOS process
- Single 5 volt power supply, TTL I/O interface (except pins for clock signals)
- 88-pin space-saving pin grid array package
- Support tool, including cross-assembly software and evaluation board

Fig. 1 – BLOCK DIAGRAM



FUNCTION OF BLOCK

Block	Function
Clock generator	This block generates a cycle clock ($\overline{\phi 3}$) used for internal operations. The clock pulses can be generated by supplying a clock signal from an external circuit through external pins CK1 and CK2, or by a crystal resonator and capacitors connected to CK1 and CK2. The master clock (MCLK) obtained by either of the above two methods has the same frequency as that of the CK1/CK2 clock and generates a cycle clock $\overline{\phi 3}$ having the frequency of the machine cycle (which is half the MCLK frequency). All internal operations are timed by the cycle clock $\overline{\phi 3}$.
Sequence control block	This block controls the DSP instruction execution sequence. The program counter (PC) is reset to address 0 by the $\overline{RST}$ pulse, and is incremented by 1 at each leading edge of $\overline{\phi 3}$ after $\overline{RST}$ is turned off. The PC output is connected to the address input of the internal microinstruction ROM (IROM) via the ROM pointer (DPR), and the ROM data is read out sequentially according to the PC value. The DPR value is also output through PA9 to PA0 to the outside to permit access to an external ROM (EROM). Data from the EROM is input to the MB 8764 through P23 to P0. At any given time, either the IROM or EROM can be used, and the choice is controlled by the IRM input. The IROM is a mask ROM with a capacity of 1,024 words x 24 bits. The ROM that has the same organization can be used for the EROM. The IROM and EROM are functionally identical. The ROM output data is transferred to the instruction register IRO at the beginning of a cycle (that is, at the leading edge of $\overline{\phi 3}$), moved to the instruction register IR1 at the beginning of the following cycle, then decoded and executed. To perform a branch instruction, address can be loaded into PC through IR0 and the IBUS, and the PC value can be saved in RAM or in another register through the IBUS. PCS is single PC stack used for subroutine execution. Two loop counters, C0 and C1, are provided to facilitate the handling of loops. This block also has a cycle counter (CYC) that controls execution of multi-cycle instructions. This counter automatically stops incrementing PC during execution of a multi-cycle instruction.

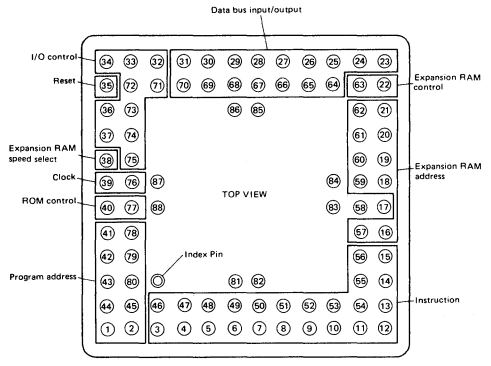
6

FUNCTION OF BLOCK (Cont'd)

Block	Function
Decoder	Instruction codes fetched from the instruction ROM and transferred to instruction registers IR0 and IR1 at the beginning of each cycle are moved to the look-ahead decoder (LAD) and decoder (DEC), respectively, then interpreted and executed. Execution of an instruction (the execution cycle) usually takes place while the instruction is stored in IR1. The DEC output controls the enable lines of the registers required for execution. Before an instruction is executed, LAD controls calculation of the effective address in RAM, interprets operations to be performed in the arithmetic and logic block, and decodes the number of cycles required for the instruction. The number of cycles required for an instruction is the number of machine cycles during which the instruction is stored in IR1.
Address calculation block	This block calculates the effective (execution) address in RAM (IRAM/ERAM) or ROM (table ROM). The address calculation block consists of index registers X and Y, stacks XS and YS for index registers X and Y, a 7-bit adder (AD1), an 8-bit adder (AD2), the virtual shift pointer (VP), and the virtual shift mode register (VS). An effective address is calculated in the LAD cycle, and the result is used as the execution address in the following execution cycle. An address in the table ROM is first calculated in AD1, then used to read table data through the table address register (TBA) and ROM pointer (DPR). To access IRAM by an instruction having one address, the effective address is first calculated in AD2, then the result is used to access IRAM through the RAM pointer (DPB). To access IRAM by an instruction having two addresses, the effective address in ARAM is calculated in AD1, the effective address in BRAM is calculated in AD2, and the results are used to access ARAM and BRAM through DPA and DPB. An address in ERAM is calculated by AD2 and the result is used to access ERAM through the ERAM pointer (DPE). Note that the table ROM is accessed by adding the value of page register PGT as the MSB element of the address, and the ROM data (16-bit) is output to IBUS through TBD. ERAM is accessed by adding the value of page register PGM as the MSB element of the address.
RAM	This device has two 128-word x 16-bit RAM areas called ARAM and BRAM. ARAM and BRAM can be used as two independent RAMs, or as a single RAM (IRAM) having a continuous address space. If the internal RAM is not sufficient, an external RAM (ERAM) can be connected to the chip. The ERAM can be used as an extension of BRAM or IRAM, but its address space is independent of BRAM or IRAM.
Arithmetic and logic block	Arithmetic and logic instructions are executed in this block. Execution of an instruction is timed by the machine cycle. This block consists of input registers A and B, an accumulator D that receives the operation result, a multiplier MLT, and an arithmetic and logic unit ALU. Multiplication is performed by a two-stage parallel multiplier in which MLT and ALU functions are pipelined. MLT multiplies the values of A and B unconditionally at each instruction and stores the intermediate results in the temporary registers TR0 and TR1. The final result of multiplication is obtained by having the ALU add the values of TR0 and TR1 according to a subsequent multiply instruction. Since the multiplier has a two-stage pipeline structure, it takes two cycles to obtain the multiplication result in D after data have been loaded into A and B. Operations other than multiplication are performed by the ALU alone, and the result is stored directly in D. The arithmetic and logic block also includes operation flags (PL, MI, ZR, and OV) that can be used to indicate conditions for conditional branch instructions. Register D has a longer bit length than the internal bus (IBUS), so a control register CLP is provided to output clipped data when the D value overflows the IBUS.
I/O interface	The I/O interface is used to exchange data between the DSP chip and an external circuit. It consists of I/O registers, an I/O controller and flags. The I/O controller controls data transfer to/from the external circuit independently of the execution of instructions. Data can be input from an external circuit through EI with or without address information through EIA. There are three input modes: the P, D, and A modes. These modes are distinguished by values set by instructions in the mode registers ADM and DMM. When data is set in EI, the input flag IF is set. In the P mode, the EI value is transferred to another register or to RAM by the program. In the D or A mode, the EI value is transferred to IRAM by cycle stealing. In the D mode, DMC is selected as the IRAM address, while in the A mode, EIA is selected. IF is reset when the EI contents are transferred to another location. Data is output to an external circuit through EA and EO. There are two output modes, and they are distinguished by the instruction data placed in EA. OF is set when data is placed in EA, and is reset when data output to the external circuit is completed. The data exchange between the DSP and an external circuit as explained above is performed through I/O control pins for synchronization with the external circuit. The I/O interface also includes the F0 and F1 flags. These are set by external input signals and used for program control or synchronization.

PIN ASSIGNMENT

Fig. 2 – PIN ASSIGNMENT



No.	Name	No.	Name	No.	Name	No.	Name	No.	Name	No.	Name
1	PA1	16	EA0	31	D_{14}	46	P23	61	EA7	76	CK2
2	PA0	17	–	32	$\overline{REQ}$	47	P21	62	EA9	77	TST
3	P22	18	EA3	33	$\overline{BCT}$	48	P18	63	$\overline{ECE}$	78	PA9
4	P20	19	EA5	34	$\overline{RCK}$	49	P16	64	D_2	79	PA6
5	P19	20	EA6	35	$\overline{RST}$	50	P15	65	D_4	80	PA4
6	P17	21	EA8	36	$\overline{FLO}$	51	P13	66	D_7	81	GND
7	P14	22	$\overline{EWE}$	37	$\overline{WCK}$	52	P10	67	D_9	82	V_{CC}
8	P12	23	D_0	38	ESS	53	P8	68	D_{10}	83	V_{CC}
9	P11	24	D_1	39	CK1	54	P6	69	D_{12}	84	GND
10	P9	25	D_3	40	IRM	55	P3	70	D_{15}	85	GND
11	P7	26	D_5	41	PA8	56	P1	71	$\overline{ACT}$	86	V_{CC}
12	P5	27	D_6	42	PA7	57	EA1	72	$\overline{AIF}$	87	V_{CC}
13	P4	28	D_8	43	PA5	58	–	73	$\overline{FL1}$	88	GND
14	P2	29	D_{11}	44	PA3	59	EA2	74	$\overline{AOF}$		
15	P0	30	D_{13}	45	PA2	60	EA4	75	$\overline{ASL}$		

No.	Name	I/O	Function
78	PA9	Output	Program address MSB
41	PA8	Output	Program address BIT8
42	PA7	Output	Program address BIT7
79	PA6	Output	Program address BIT6
43	PA5	Output	Program address BIT5
80	PA4	Output	Program address BIT4
44	PA3	Output	Program address BIT3
45	PA2	Output	Program address BIT2
1	PA1	Output	Program address BIT1
2	PA0	Output	Program address LSB
46	P23	I/O	Instruction MSB
3	P22	I/O	Instruction BIT22
47	P21	I/O	Instruction BIT21
4	P20	I/O	Instruction BIT20
5	P19	I/O	Instruction BIT19
48	P18	I/O	Instruction BIT18
6	P17	I/O	Instruction BIT17
49	P16	I/O	Instruction BIT16
50	P15	I/O	Instruction BIT15
7	P14	I/O	Instruction BIT14
51	P13	I/O	Instruction BIT13
8	P12	I/O	Instruction BIT12
9	P11	I/O	Instruction BIT11
52	P10	I/O	Instruction BIT10
10	P9	I/O	Instruction BIT9
53	P8	I/O	Instruction BIT8
11	P7	I/O	Instruction BIT7
54	P6	I/O	Instruction BIT6
12	P5	I/O	Instruction BIT5
13	P4	I/O	Instruction BIT4

No.	Name	I/O	Function
55	P3	I/O	Instruction BIT3
14	P2	I/O	Instruction BIT2
56	P1	I/O	Instruction BIT1
15	P0	I/O	Instruction LSB
39	CK1	Input	Master clock input pin 1
76	CK2	Input	Master clock input pin 2
35	$\overline{RST}$	Input	Initialization
40	IRM	Input	Internal/external ROM switching
77	TST	Input	Internal ROM test mode
62	EA9	Output	Expansion RAM address MSB
21	EA8	Output	Expansion RAM address BIT8
61	EA7	Output	Expansion RAM address BIT7
20	EA6	Output	Expansion RAM address BIT6
19	EA5	Output	Expansion RAM address BIT5
60	EA4	Output	Expansion RAM address BIT4
18	EA3	Output	Expansion RAM address BIT3
59	EA2	Output	Expansion RAM address BIT2
57	EA1	Output	Expansion RAM address BIT1
16	EAO	Output	Expansion RAM address LSB
70	D15	I/O	Data bus I/O MSB
31	D14	I/O	Data bus I/O BIT14

No.	Name	I/O	Function
30	D13	I/O	Data bus I/O BIT13
69	D12	I/O	Data bus I/O BIT12
29	D11	I/O	Data bus I/O BIT11
68	D10	I/O	Data bus I/O BIT10
67	D9	I/O	Data bus I/O BIT9
28	D8	I/O	Data bus I/O BIT8
66	D7	I/O	Data bus I/O BIT7
27	D6	I/O	Data bus I/O BIT6
26	D5	I/O	Data bus I/O BIT5
65	D4	I/O	Data bus I/O BIT4
25	D3	I/O	Data bus I/O BIT3
64	D2	I/O	Data bus I/O BIT2
24	D1	I/O	Data bus I/O BIT1
23	D0	I/O	Data bus I/O LSB
34	$\overline{RCK}$	Input	Data read clock
33	$\overline{BCT}$	Input	Data bus output enable
72	$\overline{AIF}$	Input	Data input request
36	$\overline{FLO}$	Input	Flag input
73	$\overline{FL1}$	Input	Flag input
75	$\overline{ASL}$	Input	Data output type specification in E mode
37	$\overline{WCK}$	Output	Data write clock
74	$\overline{AOF}$	Output	Output data type specification in I mode
71	$\overline{ACT}$	Output	Input enable
32	$\overline{REQ}$	Output	Data bus request
22	$\overline{EWE}$	Output	ERAM write clock
63	$\overline{ECE}$	Output	ERAM chip enable
38	ESS	Input	ERAM speed select

6

ABSOLUTE MAXIMUM RATINGS*1

Parameter	Symbol	Rating		Unit
		Min	Max	
Power supply voltage	V_{CC}	-0.3*2	7.0	V
Input voltage	V_I	-0.3*2	V_{CC} + 0.3*2	V
Output voltage	V_O	-0.3*2	V_{CC} + 0.3*2	V
Operating temperature	T_{OP}	0	85	°C
Storage temperature	T_{STG}	-55	150	°C

Note: *1 Permanent device damage may occur if the absolute maximum ratings are exceeded. Functional operation should be restricted to the conditions as detailed in the operational sections of this data sheet. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

*2 This value applies in a steady condition. It may be 0.5 V in a transient condition (for 20 to 30 ns).

RECOMMENDED OPERATING CONDITIONS

Parameter	Symbol	Condition			Unit
		Min	Typ	Max	
Power supply voltage	V_{CC}	4.5	5.0	5.5	V
Operating temperature	T_{OP}	0		85	°C

DC CHARACTERISTICS

(Recommended Operating Conditions unless otherwise specified.)

Parameter	Symbol	Condition	Min	Typ	Max	Unit
Input high voltage	V_{IH}	Other than CK1, CK2	2.0		V_{CC} +0.3	V
	V_{IHCK}	CK1, CK2	4.0		V_{CC} +0.3	V
Input low voltage	V_{IL}	Other than CK1, CK2	-0.3		0.8	V
	V_{ILCK}	CK1, CK2	-0.3		0.6	V
Output high voltage	V_{OH}	I_{OH} = -0.4 mA	2.7		V_{CC}	V
Output low voltage	V_{OL}	I_{OL} = 2 mA			0.4	V
Input leakage current	I_{LI}	V_I = 0 to 5.5 V	-25		25	μA
Input leakage current (Three-state pin input)	I_{LZ}	V_I = 0 to 5.5 V	-40		40	μA
Static power supply current	I_{CCS}			1		mA
Power supply current	I_{CC}	f_{OP} = 8 MHz		60		mA

CAPACITANCE

(V_{CC} = V_I = 0 V, f_M = 8 MHz)

Parameter	Symbol	Min	Typ	Max	Unit
Input pin	C_{IN}			5	pF
Output pin	C_{OUT}			5	pF
I/O pin	$C_{I/O}$			8	pF

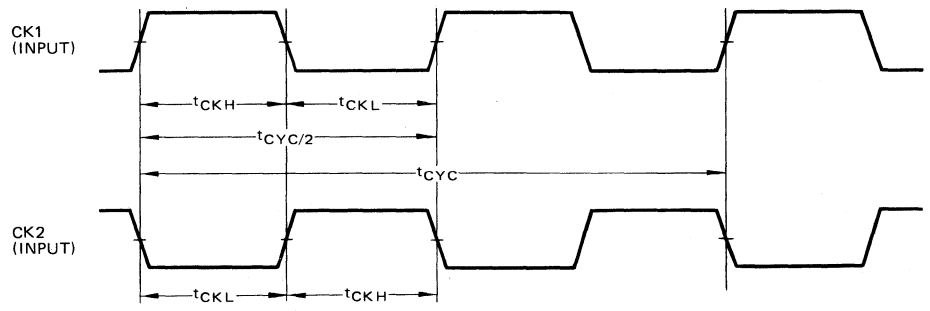
AC CHARACTERISTICS

EXTERNAL CLOCK TIMING

Parameter	Symbol	Min	Typ	Max	Unit
Cycle time *1	t_{CYC}	100			ns
High voltage pulse width	t_{CKH}	20			ns
Low voltage pulse width	t_{CKL}	20			ns

Note: *1 Value when ERAM (extended RAM) is not used. When ERAM is used, follow the specifications for the ERAM interface AC characteristics. This note also applies to the following AC characteristics.

Fig. 3 – EXTERNAL CLOCK TIMING DIAGRAM



INTERNAL OSCILLATOR (Crystal oscillator connected)

Parameter	Symbol	Min	Typ	Max	Unit
Cycle time	t_{CYC}	100			ns
Crystal frequency	f_{CYC}		16	20	MHz

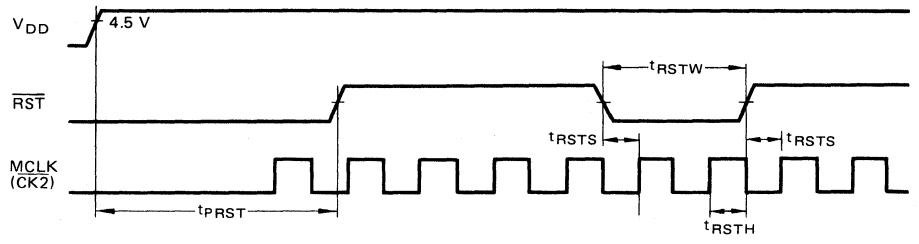
RESET INPUT TIMING

Parameter	Symbol	Min	Typ	Max	Unit
Power-on reset *1	t_{PRST}		1		ms
MCLK setup *2	t_{RSTS}	20			ns
MCLK hold *2	t_{RSTH}	15			ns
Reset input pulse width	t_{RSTW}	t_{CYC} +35			ns

Note: *1 The time specification for power-on reset applies to the internal oscillation mode.
In the external clock mode, the reset pulse must be entered so that the leading edge of MCLK ($\overline{CK2}$) can be produced while $\overline{RST}$ = 0.

*2 In the external clock mode, MCLK is considered to be $\overline{CK2}$ (the inversion of the clock input from CK2).
This note also applies to the following AC characteristics.

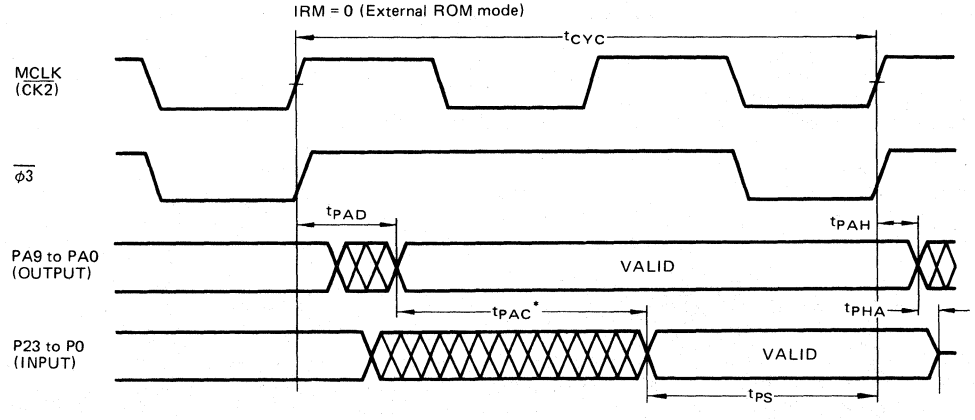
Fig. 4 – RESET INPUT TIMING DIAGRAM



EXTERNAL ROM INTERFACE TIMING

Parameter	Symbol	Condition	Min	Typ	Max	Unit
Address output delay (from MCLK pulse)	t_{PAD}	C_L = 50pF		60	75	ns
Address output hold (from MCLK pulse)	t_{PAH}	C_L = 50pF	20			ns
Data hold time (to address)	t_{PHA}	C_L = 50pF	0			ns
Data setup (before MCLK pulse)	t_{PS}	C_L = 50pF	10	10		ns

Fig. 5 – EXTERNAL ROM INTERFACE TIMING DIAGRAM



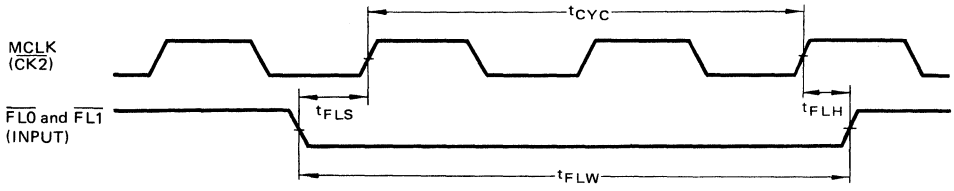
Note: *t_{PAC} is the address access time provided by the specification of the external ROM.

FLAG ($\overline{FL0}$ and $\overline{FL1}$) INPUT TIMING

Parameter	Symbol	Min	Typ	Max	Unit
Setup time	t_{FLS}	15			ns
Hold time	t_{FLH}	30			ns
Pulse width* 1	t_{FLW}	t_{CYC} + 45			ns

Note: *1 t_{FLW} (Min) = 2 x t_{CYC} + 45 when ERAM is used with ESS = 1.

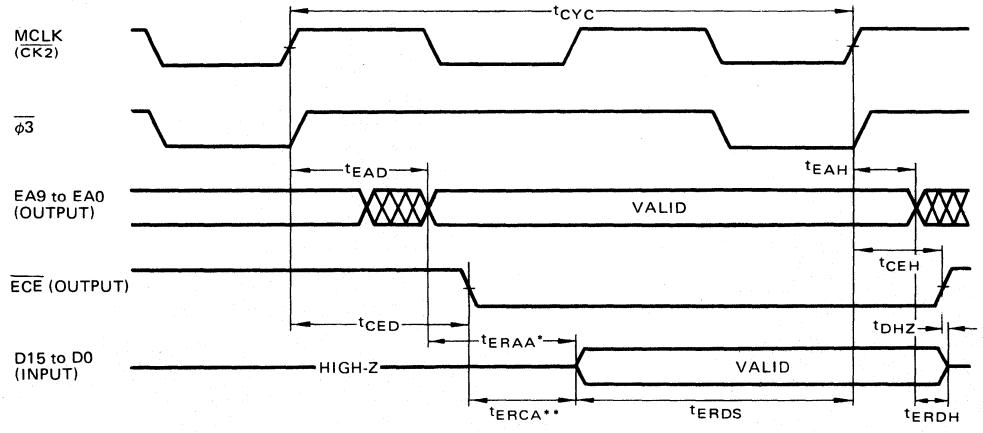
Fig. 6 – FLAG INPUT TIMING DIAGRAM



EXPANSION RAM INTERFACE TIMING (ESS = 0, Read Cycle)

Parameter	Symbol	Condition	Min	Typ	Max	Unit
Address output delay	t_{EAD}	C_L = 50pF		50	60	ns
Address output hold	t_{EAH}	C_L = 50pF	10	13		ns
Chip enable output delay	t_{CED}	C_L = 50pF		57	70	ns
Chip enable output hold	t_{CEH}	C_L = 50pF	17	19		ns
Output disable	t_{DHZ}	C_L = 50pF	0			ns
Data input setup time	t_{ERDS}	C_L = 50pF	30	25		ns
Data input hold time	t_{ERDH}	C_L = 50pF	0			ns

Fig. 7 – EXPANSION RAM INTERFACE TIMING DIAGRAM (ESS = 0, Read Cycle)



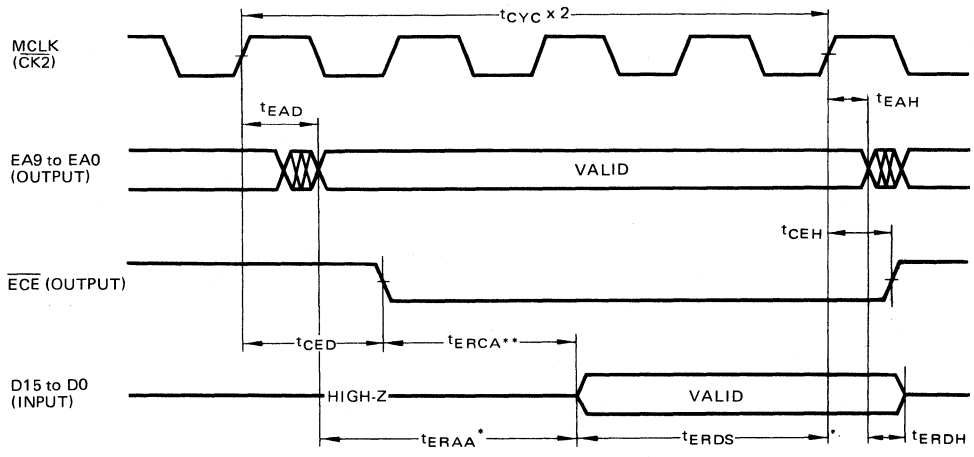
Note: *t_{ERAA} is the address access time provided by the specification of the expansion RAM.
**t_{ERCA} is the chip select access time provided by the specification of the expansion RAM.

6

EXPANSION RAM INTERFACE TIMING (ESS = 1, Read Cycle)

Parameter	Symbol	Condition	Min	Typ	Max	Unit
Address output delay	t_{EAD}	C_L = 50pF		50	60	ns
Address output hold	t_{EAH}	C_L = 50pF	10	13		ns
Chip enable output delay	t_{CED}	C_L = 50pF		57	70	ns
Chip enable output hold	t_{CEH}	C_L = 50pF	17	19		ns
Data input setup time	t_{ERDS}	C_L = 50pF	30	25		ns
Data input hold time	t_{ERDH}	C_L = 50pF	0			ns

Fig. 8 – EXPANSION RAM INTERFACE TIMING DIAGRAM (ESS = 1, Read Cycle)

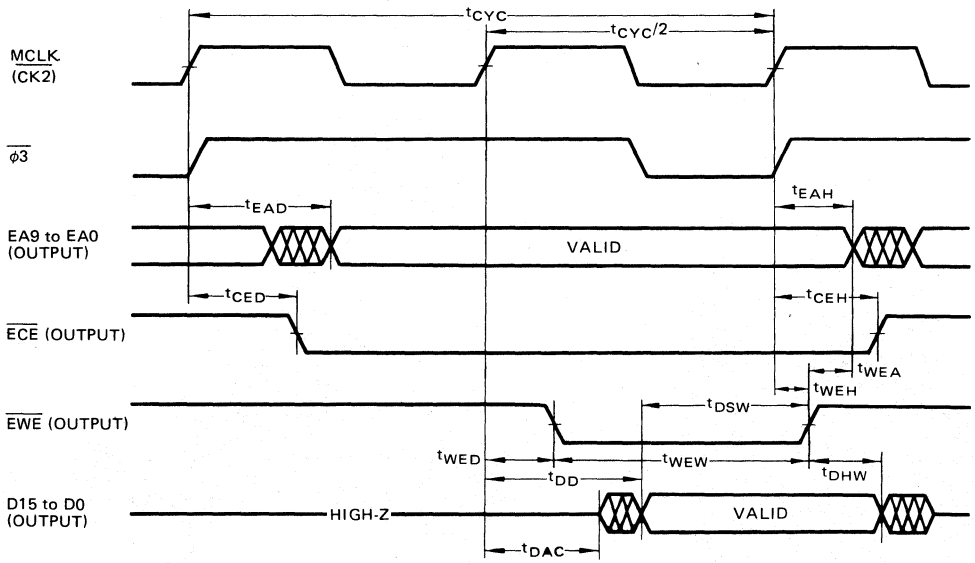


Note: $^{*}t_{ERAA}$ is the address access time provided by the specification of the expansion RAM.
$^{**}t_{ERCA}$ is the chip select access time provided by the specification of the expansion RAM.

EXPANSION RAM INTERFACE TIMING (ESS = 0, Write Cycle)

Parameter	Symbol	Condition	Min	Typ	Max	Unit
Address output delay	t_{EAD}	C_L = 50pF		50	60	ns
Address output hold	t_{EAH}	C_L = 50pF	10	13		ns
Address hold (after $\overline{EWE}$)	t_{WEA}	C_L = 50pF	5			ns
Chip enable output delay	t_{CED}	C_L = 50pF		57	70	ns
Chip enable output hold	t_{CEH}	C_L = 50pF	17	19		ns
Write enable output delay	t_{WED}	C_L = 50pF		40	50	ns
Write enable output hold	t_{WEH}	C_L = 50pF	5		35	ns
Write enable pulse width	t_{WEW}	C_L = 50pF	$\frac{t_{CYC}}{2} - 30$			ns
Data output delay	t_{DD}	C_L = 50pF + 1TTL		52	70	ns
Data setup (before $\overline{EWE}$)	t_{DSW}	C_L = 50pF + 1TTL	$\frac{t_{CYC}}{2} - 50$			ns
Data hold (after $\overline{EWE}$)	t_{DHW}	C_L = 50pF + 1TTL	5			ns
Data output active delay	t_{DAC}	C_L = 50pF + 1TTL		52	70	ns

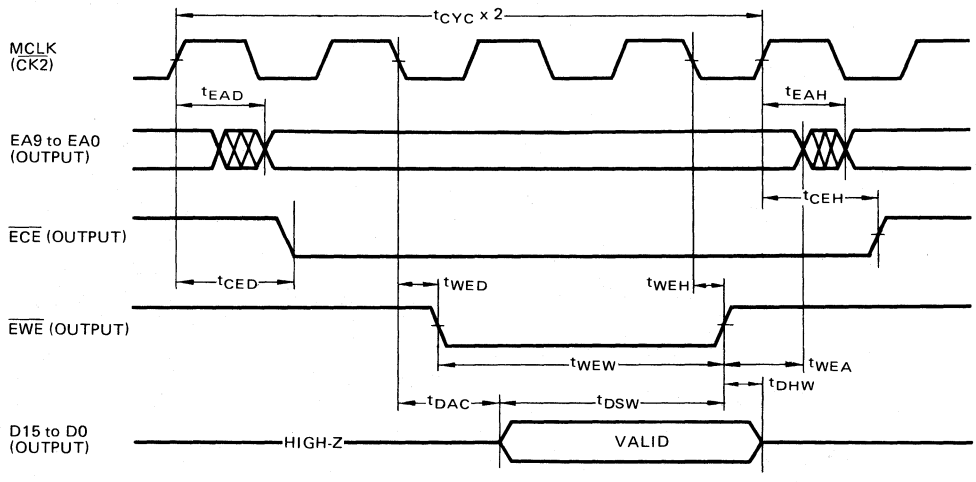
Fig. 9 – EXPANSION RAM INTERFACE TIMING DIAGRAM (ESS = 0, Write Cycle)



EXPANSION RAM INTERFACE TIMING (ESS = 1, Write Cycle)

Parameter	Symbol	Condition	Min	Typ	Max	Unit
Address output delay	t_{EAD}	C_L = 50pF		50	60	ns
Address output hold	t_{EAH}	C_L = 50pF	10	13		ns
Address output hold (after $\overline{EWE}$)	t_{WEA}	C_L = 50pF		25		ns
Chip enable output delay	t_{CED}	C_L = 50pF		57	70	ns
Chip enable output hold	t_{CEH}	C_L = 50pF	17	19		ns
Write enable output delay	t_{WED}	C_L = 50pF			50	ns
Write enable output hold	t_{WEH}	C_L = 50pF	10		35	ns
Write enable pulse width	t_{WEW}	C_L = 50pF	$t_{CYC} - 40$			ns
Data output active delay	t_{DAC}	C_L = 50pF + 1TTL		57	75	ns
Data setup (during $\overline{EWE}$)	t_{DSW}	C_L = 50pF + 1TTL	$t_{CYC} - 65$			ns
Data hold (after $\overline{EWE}$)	t_{DHW}	C_L = 50pF + 1TTL	5			ns

Fig. 10 – EXPANSION RAM INTERFACE TIMING DIAGRAM (ESS = 1, Write Cycle)

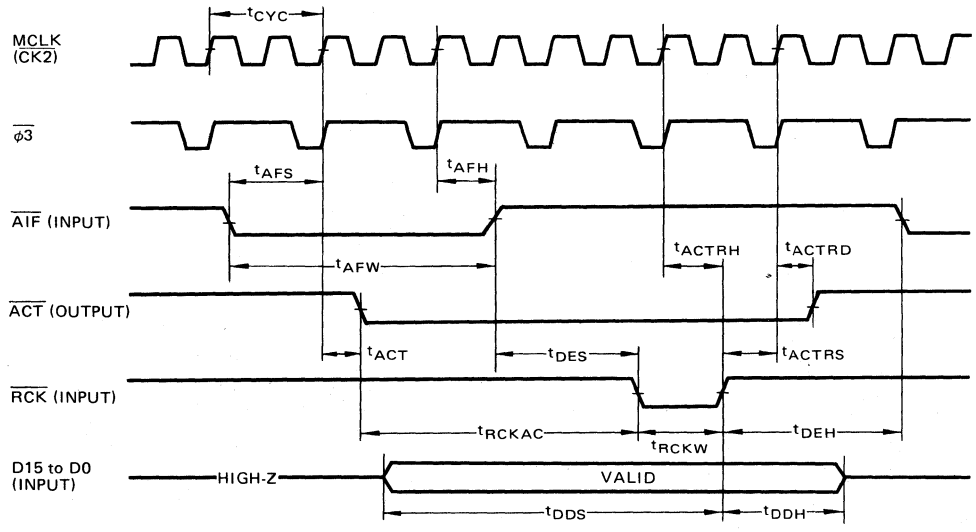


P MODE AND D MODE INPUT TIMING

Parameter	Symbol	Condition	Min	Typ	Max	Unit
$\overline{AIF}$ setup	t_{AFS}		30			ns
$\overline{AIF}$ hold	t_{AFH}		20			ns
$\overline{AIF}$ pulse width *1	t_{AFW}		t_{CYC} +50			ns
$\overline{ACT}$ fall delay	t_{ACT}	C_L = 50pF + 1TTL			70	ns
$\overline{ACT}$ reset delay	t_{ACTRD}	C_L = 50pF + 1TTL			70	ns
$\overline{RCK}$ input enable	t_{RCKAC}		0			ns
$\overline{RCK}$ pulse width	t_{RCKW}		40			ns
$\overline{RCK}$ enable setup	t_{DES}		35			ns
$\overline{RCK}$ enable hold	t_{DEH}		25			ns
Data setup	t_{DDS}		25			ns
Data hold	t_{DDH}		25			ns
$\overline{ACT}$ reset setup	t_{ACTRS}		60			ns
$\overline{ACT}$ reset hold	t_{ACTRH}		10			ns

Note: *1 t_{AFW} (Min) =2 x t_{CYC} + 50 when ERAM is used with ESS = 1.

Fig. 11 – P MODE AND D MODE INPUT TIMING DIAGRAM

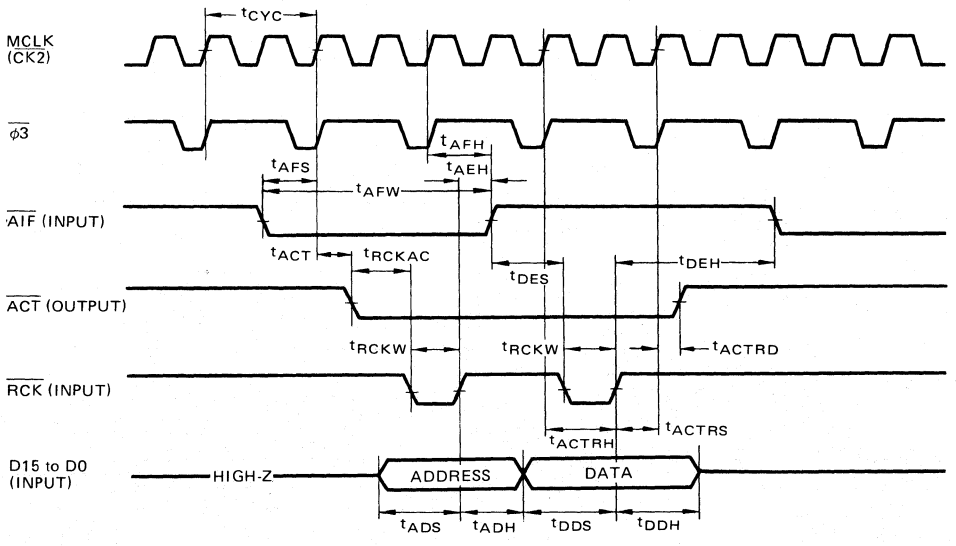


A MODE INPUT TIMING

Parameter	Symbol	Condition	Min	Typ	Max	Unit
$\overline{AIF}$ setup	t_{AFS}		30			ns
$\overline{AIF}$ hold	t_{AFH}		20			ns
$\overline{AIF}$ pulse width *1	t_{AFW}		t_{CYC} + 50			ns
$\overline{ACT}$ fall delay	t_{ACT}	C_L = 50pF + 1TTL			70	ns
$\overline{ACT}$ reset delay	t_{ACTRD}	C_L = 50pF + 1TTL			70	ns
$\overline{RCK}$ input enable	t_{RCKAC}		0			ns
$\overline{RCK}$ pulse width	t_{RCKW}		40			ns
$\overline{RCK}$ enable hold	t_{AEH}		25			ns
$\overline{RCK}$ enable setup	t_{DES}		35			ns
$\overline{RCK}$ enable hold	t_{DEH}		25			ns
Address setup	t_{ADS}		25			ns
Address hold	t_{ADH}		25			ns
Data setup	t_{DDS}		25			ns
Data hold	t_{DDH}		25			ns
$\overline{ACT}$ reset setup	t_{ACTRS}		60			ns
$\overline{ACT}$ reset hold	t_{ACTRH}		10			ns

Note: *1 t_{AFW} (Min) = 2 x t_{CYC} + 50 when ERAM is used with ESS = 1.

Fig. 12 – A MODE INPUT TIMING DIAGRAM

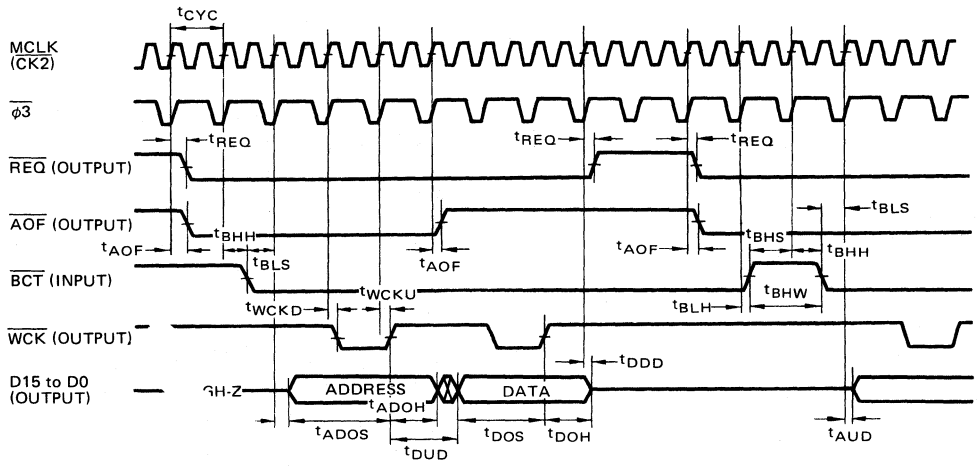


I MODE OUTPUT TIMING

Parameter	Symbol	Condition	Min	Typ	Max	Unit
$\overline{REQ}$ output delay	t_{REQ}	C_L = 50pF + 1TTL			75	ns
$\overline{AOF}$ output delay	t_{AOF}	C_L = 50pF + 1TTL			65	ns
$\overline{BCT}$ level 0 setup	t_{BLS}		40			ns
$\overline{BCT}$ level 0 hold	t_{BLH}		15			ns
$\overline{BCT}$ level 1 setup	t_{BHS}		40			ns
$\overline{BCT}$ level 1 hold	t_{BHH}		15			ns
$\overline{BCT}$ level 1 pulse width *1	t_{BHW}		t_{CYC} + 55			ns
$\overline{WCK}$ fall delay	t_{WCKD}	C_L = 50pF + 1TTL			65	ns
$\overline{WCK}$ rise delay	t_{WCKU}	C_L = 50pF + 1TTL			65	ns
Address output delay	t_{AUD}	C_L = 50pF + 1TTL			85	ns
Data output delay	t_{DUD}	C_L = 50pF + 1TTL			80	ns
Data output disable	t_{DDD}	C_L = 50pF + 1TTL			70	ns
Address setup	t_{ADOS}	C_L = 50pF + 1TTL	170			ns
Address hold	t_{ADOH}	C_L = 50pF + 1TTL	65			ns
Data setup	t_{DOS}	C_L = 50pF + 1TTL	170			ns
Data hold	t_{DOH}	C_L = 50pF + 1TTL	65			ns

Note: *1 t_{BHW} (Min) = 2 x t_{CYC} + 55 when ERAM is used with ESS = 1.

Fig. 13 – I MODE OUTPUT TIMING DIAGRAM

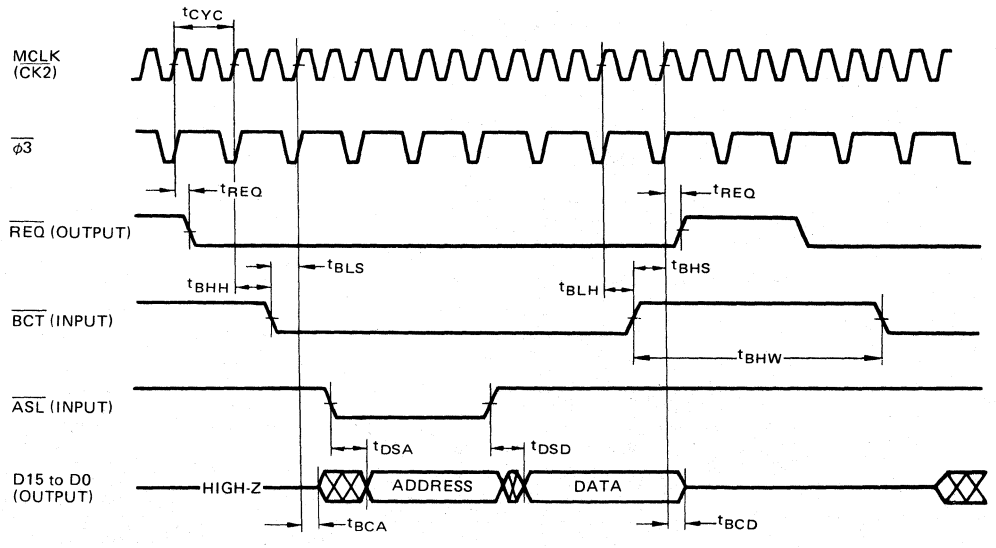


E MODE OUTPUT TIMING

Parameter	Symbol	Condition	Min	Typ	Max	Unit
$\overline{REQ}$ output delay	t_{REQ}	C_L = 50pF + 1TTL			75	ns
$\overline{BCT}$ level 0 setup	t_{BLS}		40			ns
$\overline{BCT}$ level 0 hold	t_{BLH}		15			ns
$\overline{BCT}$ level 1 setup	t_{BHS}		40			ns
$\overline{BCT}$ level 1 hold	t_{BHH}		15			ns
$\overline{BCT}$ level 1 pulse width *1	t_{BHW}		t_{CYC} +55			ns
Output active delay	t_{BCA}	C_L = 50pF + 1TTL			85	ns
Address output from fall of $\overline{ASL}$	t_{DSA}	C_L = 50pF + 1TTL			85	ns
Data output from rise of $\overline{ASL}$	t_{DSD}	C_L = 50pF + 1TTL			85	ns
Output inactive	t_{BCD}	C_L = 50pF + 1TTL			70	ns

Note: *1 t_{BHW} (Min) = 2 x t_{CYC} + 55 when ERAM is used with ESS = 1.

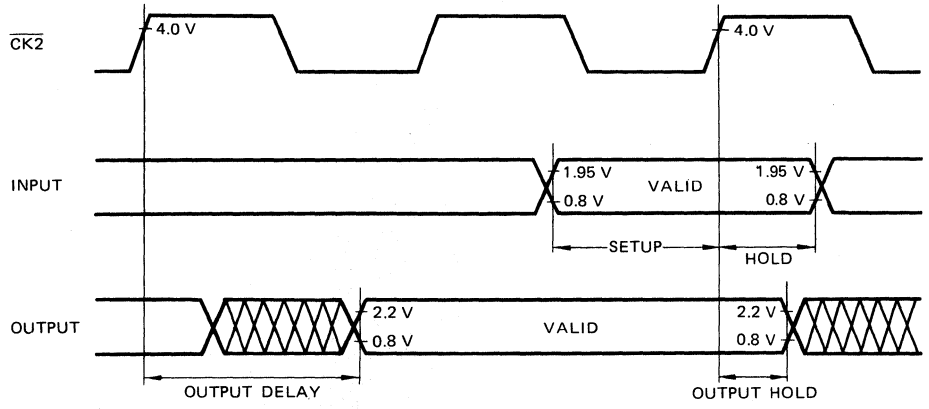
Fig. 14 – E MODE OUTPUT TIMING DIAGRAM



AC CHARACTERISTICS MEASUREMENT CONDITIONS

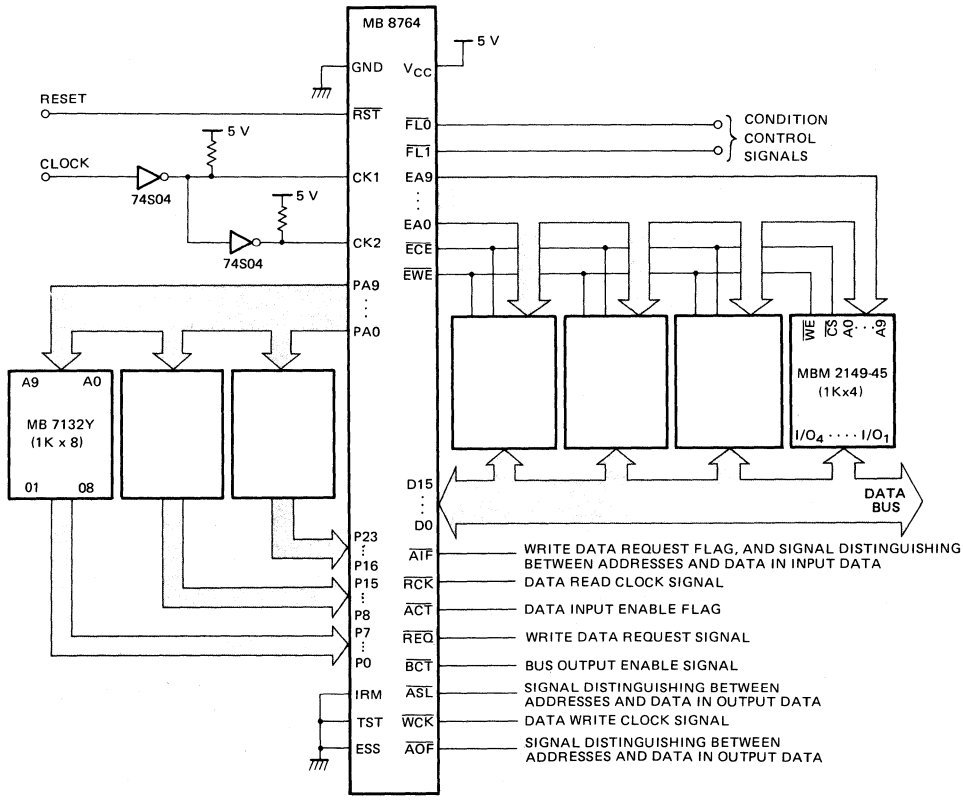
Parameter	Symbol	Condition
Power supply voltage	V_{DD}	5 V ± 10%
Ambient temperature	T_A	0 to 85°C

Fig. 15 – AC CHARACTERISTICS MEASUREMENT WAVEFORMS



APPLICATION INFORMATION

Fig. 16 – EXAMPLE OF MB 8764 APPLICATION CIRCUIT (with 1K-word External ROM and 1K-word External RAM)



INSTRUCTION SET

ARITHMETIC AND LOGIC INSTRUCTIONS

Mnemonic	Processing performed	Mnemonic	Processing performed
NOP	No operation	ABS	$\|D\| \rightarrow D$
ADD	$A + B \rightarrow D$	NEG	$-D \rightarrow D$
MLT	$A \times B \rightarrow D$	SRA	Shift D right arithmetic $\rightarrow D$
SUB	$B - A \rightarrow D$	SLA	Shift D left arithmetic $\rightarrow D$
MSM	$D + A \times B \rightarrow D$	AND	$D \cap A \rightarrow D$
MRD	$D - A \times B \rightarrow D$	ORA	$D \cup A \rightarrow D$
SUM	$D + A \rightarrow D$	DIV	$D \div A \rightarrow D$
RED	$D - A \rightarrow D$	COM	$\overline{D} \rightarrow D$

TRANSFER INSTRUCTIONS

Mnemonic		Processing performed
LTB: (Arithmetic/logic instruction)	$a, $b	ROMT → A, BRAM/ERAM → B
LAB: (Arithmetic/logic instruction)	$a, $b	ARAM → A, BRAM/ERAM → B
MAB: (Arithmetic/logic instruction)	$a, $b	ARAM → BRAM/ERAM
MBA: (Arithmetic/logic instruction)	$a, $b	BRAM/ERAM → ARAM
MOV: (Arithmetic/logic instruction)	$a, Reg [:Reg . .]	IRAM/ERAM → Register
MOV: (Arithmetic/logic instruction)	#$d, Reg [:Reg . .]	Immediate data (d) → Register
MOV: (Arithmetic/logic instruction)	Reg, Reg [:Reg . .]	Register → Register
LDI: (Arithmetic/logic instruction)	#$d	d → A
LIB: (Arithmetic/logic instruction)	#$d	d → A, BRAM → B

JUMP INSTRUCTIONS

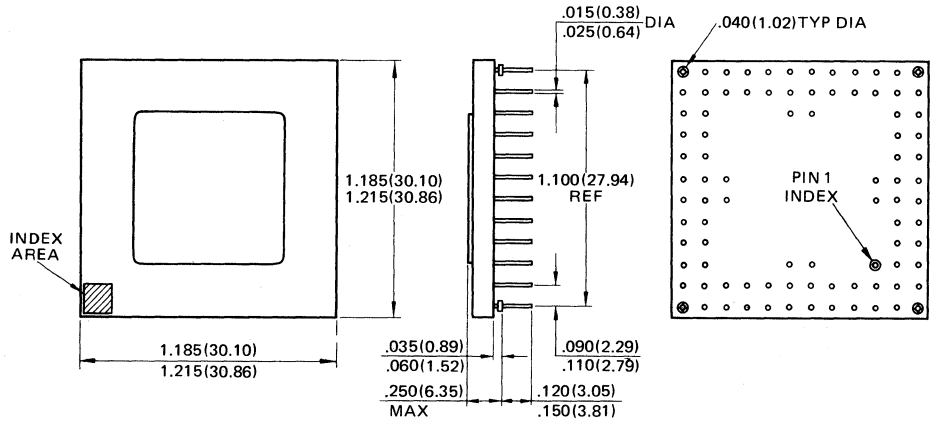
Mnemonic		Processing performed
JMP: (Arithmetic/logic instruction)	#$d	Unconditional jump (d → PC)
JOC: (Arithmetic/logic instruction)	#$d, flag	Conditional jump (d→PC)
JOC: (Arithmetic/logic instruction)	$a, flag	Conditional jump (IRAM/ERAM → PC)
JSR: (Arithmetic/logic instruction)	#$d	Jump to subroutine (PC → PCS, d → PC)
RTS: (Arithmetic/logic instruction)		Return from subroutine

MISCELLANEOUS INSTRUCTIONS

Mnemonic		Processing performed
CLR: [Reg [:Reg . . .] . . .]		Clear register
SET: [Reg [:Reg . . .] . . .]		Set register
MXY: (Arithmetic/logic instruction)	#d_1$, #Sd$_2$	$X + d_1 \rightarrow X$, $Y + d_2 \rightarrow Y$
LIY: (Arithmetic/logic instruction)	#$d	d → A,BRAM → B, Y + 1 → Y
AVP: (Arithmetic/logic instruction)	#$d	VP + d → VP
LVP: (Arithmetic/logic instruction)	#$d	d → VP
ADY: (Arithmetic/logic instruction)		Y+YS → Y
GXY: (Arithmetic/logic instruction)		XS → X, YS → Y
SXY: (Arithmetic/logic instruction)	#$d	d → C1, X → XS, Y → YS, O → X, O → Y
NOP: (Arithmetic/logic instruction)		No operation

PACKAGE DIMENSIONS

88-LEAD CERAMIC (METAL SEAL) REPEATED QUAD IN-LINE PACKAGE
(CASE No.: RIT-88C-A01)



Dimensions in
inches (millimeters)