

1-177-464/465/466/467



MEMORY CARD TYPES

□ SRAM

● Full CMOS utilising full CMOS die

Type	Capacity (KBytes)	Write protect switch	Battery	Battery life	Pin assignment	Dimensions
BS8D1-A	8	No	BR2016 lithium battery	5Y	1-1	Fig. 3
BS16D1-A	16			5Y		
BS24D1-A	24			5Y		
BS32D1-A	32			5Y		
BS8E1-A	8	Yes	BR2016 lithium battery	5Y	1-1	Fig. 3
BS16E1-A	16			5Y		
BS24E1-A	24			5Y		
BS32E1-A	32			5Y		

Chip enable access time = 250ns. Output enable access time = 100ns.

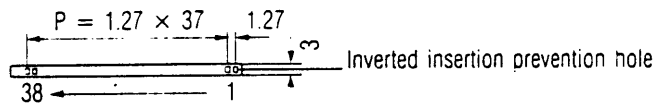
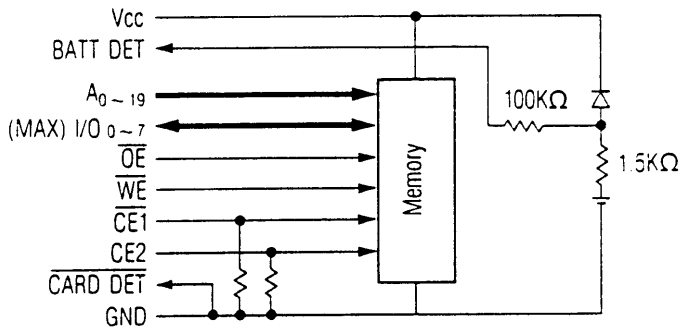
● Mix CMOS utilising mix CMOS die

Type	Capacity (KBytes)	Memory	Write protect switch	Battery	Battery life	Pin assignment	Dimensions
BS32D1-B	32	L Version	No	BR2016 lithium battery	1.3Y	1-1	Fig. 8
BS64D1-B	64				8M		
BS96D1-B	96				6M		
BS128D1-B	128				4M		
BS160D1-B	160				3M		
BS192D1-B	192				2.5M		
BS32D1-C	32	LL Version	No	BR2016 lithium battery	5Y	1-1	Fig. 8
BS64D1-C	64				3.9Y		
BS96D1-C	96				2.7Y		
BS128D1-C	128				2Y		
BS160D1-C	160				1.6Y		
BS192D1-C	192				1.4Y		
BS32E1-B	32	L Version	Yes	BR2016 lithium battery	1.3Y	1-1	Fig. 9
BS64E1-B	64				8M		
BS96E1-B	96				6M		
BS128E1-B	128				4M		
BS32E1-C	32	LL Version	Yes	BR2016 lithium battery	5Y	1-1	Fig. 9
BS64E1-C	64				3.9Y		
BS96E1-C	96				2.7Y		
BS128E1-C	128				2Y		
BS32F1-C	32	LL Version	No	BR2325 lithium battery	5Y	1-1	Fig. 10
BS64F1-C	64				5Y		
BS96F1-C	96				5Y		
BS128F1-C	128				4.3Y		
BS160F1-C	160				3.5Y		
BS192F1-C	192				2.9Y		
BS256F1-C	256				2.2Y		
BS512F1-C	512				1.1Y		
BS32G1-C	32	LL Version	Yes	BR2016 lithium battery	5Y	1-1	Fig. 11
BS64G1-C	64				5Y		
BS96G1-C	96				5Y		
BS128G1-C	128				4.3Y		
BS160G1-C	160				3.5Y		
BS192G1-C	192				2.9Y		
BS256G1-C	256				2.2Y		
BS512G1-C	512				1.1Y		

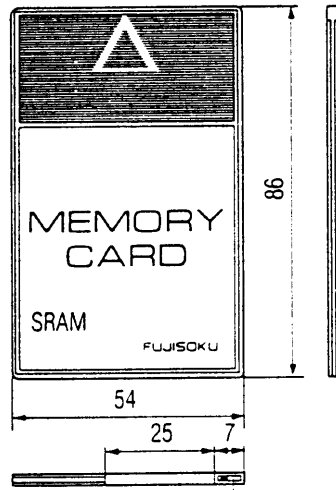
Chip enable access time = 200ns. Output enable access time = 70ns.

*DRAM CARDS of 512K and 1M byte (without built-in battery) are also available. Y = Years and M = months in the table.

SRAM BLOCK DIAGRAM AND CARD DIMENSIONS

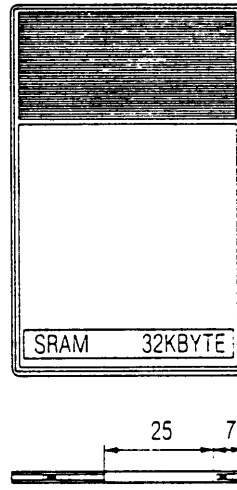


- SRAM using BR2016 battery
BS[] D1-A
BS[] D1-B
BS[] D1-C



Battery holder Battery lock mechanism

- SRAM using BR2016 battery with write protect switch
BS[] E1-A
BS[] E1-B
BS[] E1-C

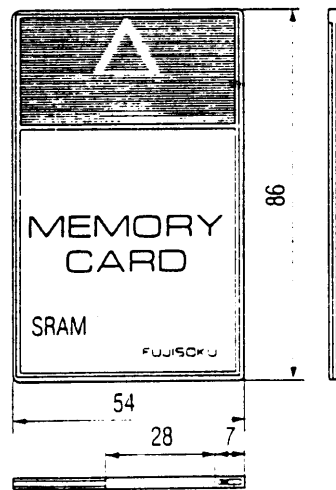
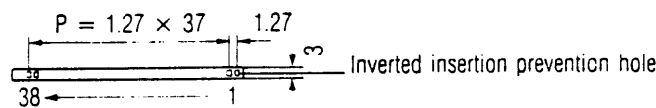


Protect switch Battery holder Battery lock mechanism

Fig. 8

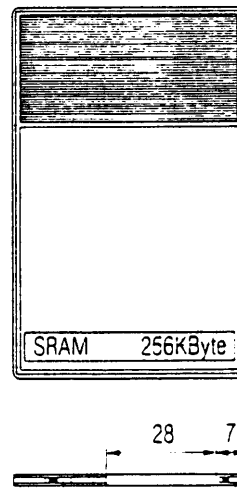
Fig. 9

- SRAM using BR2325 battery
BS[] F1-C



Battery holder Battery lock mechanism

- SRAM using BR2325 battery with write protect switch
- BS[] G1-C



Protect switch Battery holder

Fig. 10

Fig. 11



PIN ASSIGNMENT TABLE

Type of MEMORY		SRAM		OTP/EPPROM		EEPROM	MASKROM
Pin No	Table No	1-1	1-2	1-3	1-4	1-5	1-6
1		GND	GND	GND	GND	GND	GND
2		BATT DET	NC	NC	NC	NC	NC
3		Vcc	Vcc	Vcc	Vcc	Vcc	Vcc
4		$\overline{CE1}$	\overline{CE}	\overline{CE}	\overline{CE}	\overline{CE}	\overline{CE}
5		CE2	NC	NC	PGM	NC	NC
6		\overline{WE}	Vpp	NC	Vpp	WE	NC
7		\overline{OE}	\overline{OE}	\overline{OE}/Vpp	\overline{OE}	\overline{OE}	\overline{OE}
8		I/O ₀	I/O ₀	I/O ₀	I/O ₀	I/O ₀	I/O ₀
9		I/O ₁	I/O ₁	I/O ₁	I/O ₁	I/O ₁	I/O ₁
10		I/O ₂	I/O ₂	I/O ₂	I/O ₂	I/O ₂	I/O ₂
11		I/O ₃	I/O ₃	I/O ₃	I/O ₃	I/O ₃	I/O ₃
12		I/O ₄	I/O ₄	I/O ₄	I/O ₄	I/O ₄	I/O ₄
13		I/O ₅	I/O ₅	I/O ₅	I/O ₅	I/O ₅	I/O ₅
14		I/O ₆	I/O ₆	I/O ₆	I/O ₆	I/O ₆	I/O ₆
15		I/O ₇	I/O ₇	I/O ₇	I/O ₇	I/O ₇	I/O ₇
16		A ₀	A ₀	A ₀	A ₀	A ₀	A ₀
17		A ₁	A ₁	A ₁	A ₁	A ₁	A ₁
18		A ₂	A ₂	A ₂	A ₂	A ₂	A ₂
19		A ₃	A ₃	A ₃	A ₃	A ₃	A ₃
20		A ₄	A ₄	A ₄	A ₄	A ₄	A ₄
21		A ₅	A ₅	A ₅	A ₅	A ₅	A ₅
22		A ₆	A ₆	A ₆	A ₆	A ₆	A ₆
23		A ₇	A ₇	A ₇	A ₇	A ₇	A ₇
24		A ₈	A ₈	A ₈	A ₈	A ₈	A ₈
25		A ₉	A ₉	A ₉	A ₉	A ₉	A ₉
26		A ₁₀	A ₁₀	A ₁₀	A ₁₀	A ₁₀	A ₁₀
27		A ₁₁	A ₁₁	A ₁₁	A ₁₁	A ₁₁	A ₁₁
28		A ₁₂	A ₁₂	A ₁₂	A ₁₂	A ₁₂	A ₁₂
29		A ₁₃	A ₁₃	A ₁₃	A ₁₃	A ₁₃	A ₁₃
30		A ₁₄	A ₁₄	A ₁₄	A ₁₄	A ₁₄	A ₁₄
31		A ₁₅	A ₁₅	A ₁₅	A ₁₅	A ₁₅	A ₁₅
32		A ₁₆	A ₁₆	A ₁₆	A ₁₆	A ₁₆	A ₁₆
33		A ₁₇	A ₁₇	A ₁₇	A ₁₇	A ₁₇	A ₁₇
34		A ₁₈	A ₁₈	A ₁₈	A ₁₈	A ₁₈	A ₁₈
35		A ₁₉	A ₁₉	A ₁₉	A ₁₉	A ₁₉	A ₁₉
36		Vcc	Vcc	Vcc	Vcc	Vcc	Vcc
37		$\overline{CARD DET}$	$\overline{CARD DET}$	$\overline{CARD DET}$	$\overline{CARD DET}$	$\overline{CARD DET}$	$\overline{CARD DET}$
38		GND	GND	GND	GND	GND	GND

Either \overline{OE} or \overline{CE} will be selected for 128 kbytes of the masked ROM. *