

SHARP

Extension RAM

MODEL **MZ-1R12**

INSTRUCTION MANUAL



FOREWORD

Thank you for your purchase of the SHARP Extension RAM, MZ-1R12.

Please read this manual carefully for proper use. Also, be sure to keep this manual for later use.

This manual should be helpful during use or when problems arise.

PRECAUTIONS

- 1) The MZ-1R12 contains delicate electronic parts and components, including LSIs. Avoid excessive temperature variations, humidity, dust, and direct sunlight.
- 2) Do not tap it sharply or drop on the floor.
- 3) Static electricity can destroy integrated circuits. Therefore, be sure to touch a grounded water pipe (a steel pipe) or the like before touching any components, and do not touch the IC pins, etc.
- 4) Since the MZ-1R12's RAM is backed up by battery, voltage is always present. Therefore, do not place the MZ-1R12 on any surface which is electrically conductive.

CAUTION

- 1) Contents of this manual are subject to change without notice.
- 2) No part of this manual may be reproduced in any form without the express written permission of the Sharp Corporation.

◀TABLE OF CONTENTS▶

INTRODUCTION

page

- INTRODUCTION 2
- FEATURES 2
- CONNECTION 2
- SPECIFICATIONS 2
- INSTRUCTIONS 3
 - 1. Storage of the MZ-700 system programs 3
 - 2. To use the MZ-1R12 as an external memory of the MZ-700 5
 - 3. A sample application program 7

Note : Keep bit 7 of the DIP switch off unless the MZ-1R12 is used to store the system programs of the MZ-700.
See page 5 for details.

INTRODUCTION

The MZ-1R12 may be used to store the system programs of the MZ-700, allowing them to be bootstrapped from RAM instead of from magnetic cassette tape. It may also be used as external memory.

The MZ-1U06 (expansion unit) is required in order to install the MZ-1R12.

FEATURES

- (1) It is possible to start the system programs automatically turning on the MZ-700 power.
- (2) Since the CMOS STATIC RAM (S-RAM) is battery operated, it can be used as non-volatile memory (data is maintained even when the MZ-700's power is off).
- (3) Data may be written in and read from the MZ-1R12 via I/O ports.
- (4) The MZ-1R12 includes an address counter which increments automatically when data is read or written. This makes it easy to read or write consecutive addresses.

CONNECTION

First, make sure that the MZ-700 and the MZ-1U06 expansion unit are both turned off. Then install the MZ-1R12 to the MZ-1U06. Be sure to turn on power for the MZ-1U06 first, then for the MZ-700.

SPECIFICATIONS

- | | |
|----------------------------|---|
| (1) Memory capacity | 32K bytes |
| (2) ROM | 4K bytes |
| (3) Data preservation time | Approx. 1.5 months
(with battery charged 150%) |
| (4) Charging time | 100% 29 hours
150% 43 hours |
| (5) Supply voltage | 5V DC \pm 10% |
| (6) Devices | CMOS STATIC RAM (2K bytes) \times 16 |

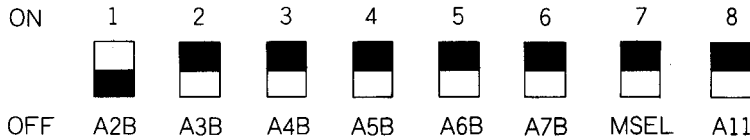
INSTRUCTIONS

1. Storage of the MZ-700 system programs

The BASIC interpreter may be loaded from the MZ-1R12 instead of from a cassette tape. This requires storing the BASIC interpreter (1Z-013B) into the MZ-1R12.

The I/O ports addressed as F8H, F9H, and FAH must be used for bootstrapping since the port addresses are initially so set in the ROM.

Specifically, the DIP switch provided on the MZ-1R12 board for port address setting must be set as shown below (this is the initial setting done before delivery).



Be sure to turn on power for the MZ-1U06 expansion unit first, then for the MZ-700. The message shown below will then appear.

Press R, W or M
R : read S-RAM
W : write S-RAM
M : monitor

Initial message

Depression of the R, W, or M keys.

R : Programs are loaded from the S-RAM with check sum examined. If a wrong check sum is detected, check sum error will be indicated before the initial message appears again.

(Example) The BASIC interpreter will be read from the S-RAM into the MZ-700.

W : The message shown below will appear.

S-RAM programming
Set master tape ↓ Play

Load an BASIC cassette tape or an application program written in the machine language, then depress the PLAY button.

The contents of the cassette tape will be transferred to the S-RAM together with check sum.

The initial message will appear after completion of transfer.

If the BASIC interpreter is stored in the S-RAM, you have only to depress the R key on the initial screen in order to load BASIC. It is time-saving and much easier to operate.

Note : The MZ-1R12 memory capacity is 32K bytes. If you attempt to store programs exceeding this limit, " File is too long " is displayed and the initial screen reappears.

M : Controls transfers to the MZ-700 ROM, monitor and stands by for command input.

Refer to the MZ-700 instruction manual for the monitor.

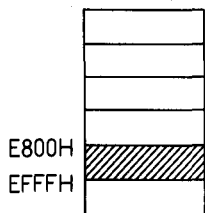
Note : Only one system program may be stored on one board. The board is mounted with a P-ROM (for MZ-700 bootstrapping) to which the E800 H~EFFFH address is assigned.

The address space assigned is 2K bytes while the P-ROM has a capacity of 4K bytes.

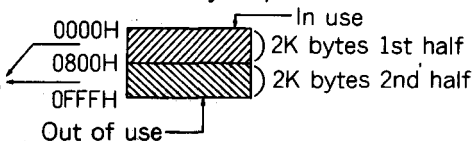
Bit 8 (All) of the DIP switch selects either the first or the second half of the P-ROM.

	ON	OFF
DIP switch bit 8	1st half selected	2nd half selected

MZ-700 memory map



P-ROM memory map



2. To use the MZ-1R12 as an external memory of the MZ-700

- (1) To write data MZ-700 RAM → S-RAM OUT command
 - (2) To read data MZ-700 RAM ← S-RAM INP command
- (Data here include numerical and character data.)

(i) Port address

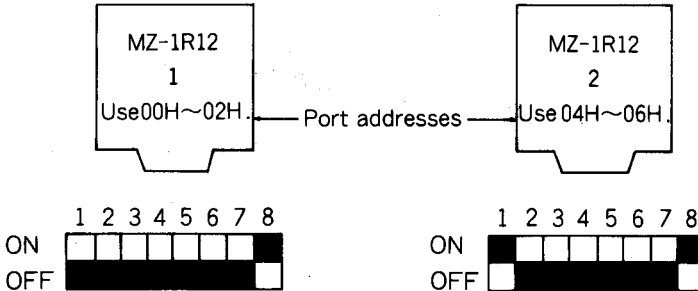
The MZ-1R12 uses I/O ports addressed as F8H, F9H, and FAH. Functions are assigned to the ports as follows.

Port address	OUT	INP
F8H	Sets high address order	Resets counter
F9H	Sets low address order	Reads data
FAH	Writes data	Out of use

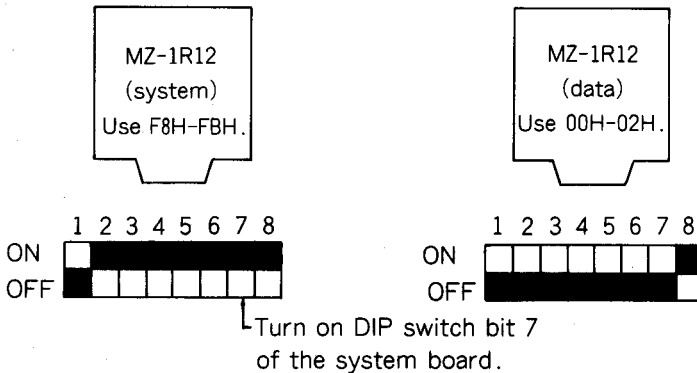
(Assignment at the time of shipment)

Notes :

- (1) Keep DIP switch bit 7 (MSEL) off to use the MZ-1R12 as an external memory.
- (2) If you install more than one MZ-1R12 board for storing data, set the DIP switches as shown below. (The port address should start from 00H.)



If board 1 is to be assigned to system and board 2 to data, set the DIP switches as follows.



(ii) Sample programs

The MZ-1R12 increments address automatically when reading and writing data. Therefore, it is enough to designate only the first address when reading or writing data from or into consecutive locations.

Sample program with BASIC [1Z-013B]

This program write numeric data for 0 to 255 S-RAM addresses 0000H~00FFH. (See the figure below for the S-RAM addresses.)

```
10 INP # $F8, A
20 FOR A= 0 TO 255
30 OUT # $FA, A
40 NEXT A
50 END
```

Any variable name may be used.
Set address counter to \$ 0000H.
Write data 0~255 (variable A value)

Note : If the start address is not 0000H, change line 10 as follows.

```
10 OUT # $F8, $01
15 OUT # $F9, $FF
```

If start address is \$ 01FFH
Set high address order
Set low address order

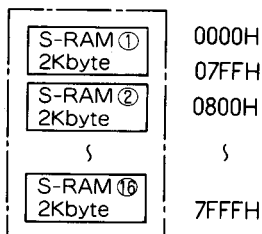
To display data held in locations 0000H~00FFH of S-RAM

```
10 INP # $F8, A
20 FOR I= 0 TO 255
30 INP # $F9, A
40 PRINT A,
50 NEXT I
60 END
```

Reset counter.
Read data and input variable A.
Display A value.

Note : The address counter increments each time line 30 is executed.

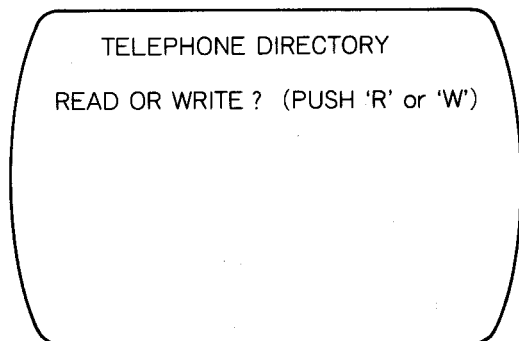
Addresses are assigned to the S-RAM as follows.



3. A sample application program

This program generates a telephone directory in the S-RAM, permitting new entry and reference.

When the program starts, an initial screen appears as shown below and a stands by for command input.



- * Depress W to enter name and telephone number. The initial screen appears again after data for five people have been entered.
- * Depress R to refer to the directory held in the S-RAM.

```
10 PRINT " ■ "
20 PRINT " ■ " : CLR
30 PRINT " ■ " : PRINT " TELEPHONE DIRECTORY "
40 PRINT " ■ " : INPUT " READ OR WRITE ? (PUSH 'R' or 'W') " : A $
50 IF A $ = " W " THEN 80
60 IF A $ = " R " THEN 270
70 GOTO 20
80 DIM SN $ ( 5 ) , ST $ ( 5 )
90 FOR I=1 TO 5
100 PRINT : INPUT " NAME ? : " ; SN $ ( I )
110 INPUT " TEL ? : " ; ST $ ( I )
120 NEXT I
130 OUT#$F8, $70 : OUT#$F9, $0
140 FOR I=1 TO 5
150 OUT#$FA, LEN(SN $ ( I ))
160 FOR J=1 TO LEN(SN $ ( I ))
170 A=ASC(MID $ ( SN $ ( I ) , J , 1 ))
180 OUT#$FA, A
190 NEXT J
```



```

200 OUT#$FA,LEN(ST$(I))
210 FOR J=1 TO LEN(ST$(I))
220 A=ASC(MID$(ST$(I),J,1))
230 OUT#$FA,A
240 NEXT J
250 NEXT I
260 GOTO 10
270 REM
280 DIM NN$(5),NT$(5)
290 OUT#$F8,$70:OUT#$F9,$0
300 FOR I=1 TO 5
310 INP#$F9,L
320 FOR J=1 TO L
330 INP#$F9,A
340 NN$(I)=NN$(I)+CHR$(A)
350 NEXT J
360 INP#$F9,L
370 FOR J=1 TO L
380 INP#$F9,A
390 NT$(I)=NT$(I)+CHR$(A)
400 NEXT J
410 NEXT I
420 REM
430 FOR I=1 TO 5
440 PRINT:PRINT"NAME:";NN$(I)
450 PRINT"TEL:";NT$(I)
460 NEXT I
470 GOTO 20
480 END

```

SHARP CORPORATION
OSAKA, JAPAN

Printed in Japan
Gedruckt in Japan
Imprimé au Japon
3L0.5 01S