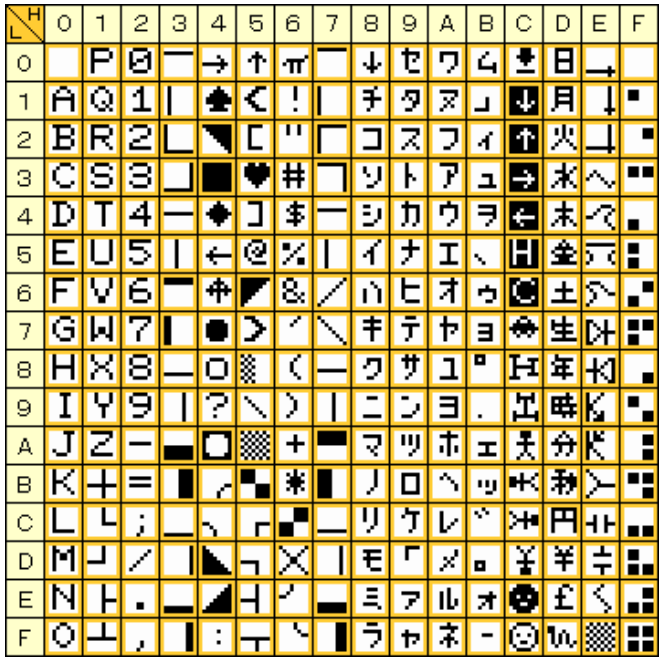
VRAM

<table>
<tr><th>address</th><th>R / W</th><th>Description</th></tr>
<tr><td>D000h
:
D3FFh</td><td>R / W</td><td>Text VRAM
<table>
<tr><td>行\桁</td><td>0</td><td>1</td><td>2</td><td></td><td>38</td><td>39</td></tr>
<tr><td>0</td><td>D000</td><td>D001</td><td>D002</td><td></td><td>D026</td><td>D027</td></tr>
<tr><td>1</td><td>D028</td><td>D029</td><td>D02A</td><td></td><td>D04E</td><td>D04F</td></tr>
<tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td>24</td><td>D3C0</td><td>D3C1</td><td>D3C2</td><td></td><td>D3E6</td><td>D3E7</td></tr>
</table>
<table>
<tr><th>Bit</th><td>7</td><td>6</td><td>Five</td><td>Four</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><th>function</th><td colspan="8">Display code</td></tr>
</table>
Specify the display code of the character to be displayed</td></tr>
<tr><td>D400h
:
D7FFh</td><td>R / W</td><td>PCG VRAM 1 (MZ-1500)
<table>
<tr><th>Bit</th><td>7</td><td>6</td><td>Five</td><td>Four</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><th>function</th><td colspan="8">PCGN [7: 0]</td></tr>
</table>
Specify the PCG number together with PCG VRAM 2</td></tr>
<tr><td>D800h
:
DBFFh</td><td>R / W</td><td>Attribute VRAM
<table>
<tr><th>Bit</th><td>7</td><td>6</td><td>Five</td><td>Four</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><th>function</th><td>ATB</td><td colspan="3">FC</td><td></td><td colspan="3">BC</td></tr>
</table>
<table>
<tr><th>ATB</th><td>Character set switching</td><td>0: Set 0 (uppercase letters / katakana)
1: Set 1 (lowercase letters / hiragana)</td></tr>
<tr><th>FC</th><td>Letter color</td><td rowspan="2">
<table>
<tr><th>FC, BC</th><th>color</th><th>FC, BC</th><th>color</th></tr>
<tr><td>000</td><td>■ Black</td><td>100</td><td>■ Green</td></tr>
<tr><td>001</td><td>■ Blue</td><td>101</td><td>■ Cyan</td></tr>
<tr><td>010</td><td>■ Red</td><td>110</td><td>■ Yellow</td></tr>
<tr><td>011</td><td>■ Magenta</td><td>111</td><td>□ White</td></tr>
</table>
</td></tr>
<tr><th>BC</th><td>Background color</td></tr>
</table>
</td></tr>
<tr><td>DC00h
:
DFFFh</td><td>R / W</td><td>PCG VRAM 2 (MZ-1500)
<table>
<tr><th>Bit</th><td>7</td><td>6</td><td>Five</td><td>Four</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><th>function</th><td colspan="2">PCGN [9: 8]</td><td colspan="2"></td><td>PCGE</td><td colspan="3"></td></tr>
</table>
</td></tr>
</table>

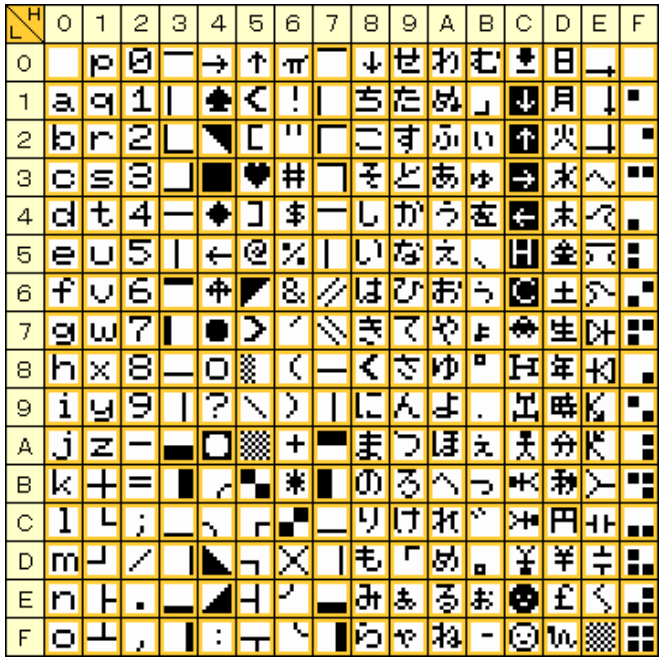
		PCGN	PCG number	0-1023
		PCGE	PCG display control	0: Do not show PCG at this position 1: Show PCG at this position
		Specify the PCG number together with PCG VRAM 1		

Display code table

Character set 0 (ATB = 0)



Character set 1 (ATB = 1)



Memory-mapped I / O

address	R / W	Description
E000h	W	8255 Port A

Bit	7	6	Five	Four	3	2	1	0
R / W	W				W			
signal	556RST				KEYSTROBE			

556RST	Reset the timer for blinking the cursor	0: Reset

			1: Do nothing
		KEYSTROBE	Keyboard matrix strobe output

E001h R

8255 Port B
keyboard matrix data entry

\ **Strobe Data **	**0**	**1**	**2**	**3**	**Four**	**Five**	**6**	**7**	**8**	**9**
D7	Kana	Y	Q	I	A	1	*	INST	BREAK	F1
D6	GRAPH	Z	R	J	B	2	+	DEL	CTRL	F2
D5	= =	@	S	K	C	3	---	↑		F3
D4	Alphabet	(	T	L	D	Four	SP	↓		F4
D3		)	U	M	E	Five	0	→		F5
D2	;		V	N	F	6	9	←		
D1	::		W	O	G	7	,	??		
D0	CR		X	P	H	8	..	/	SHIFT	

E002h R / W

8255 Port C

Bit	7	6	Five	Four	3	2	1	0
R / W	R	R	R	R	W	W	W	W
signal	$\overline{\text{VBLK}}$	556OUT	RDATA	MOTOR	M-ON	INTMSK	WDATA	SOUNDMSK

$\overline{\text{VBLK}}$	Vertical blanking signal	
556OUT	Timer for blinking cursor	
RDATA	Input from data recorder	
MOTOR	Data recorder motor status	0: OFF 1: ON
M-ON	Data recorder motor ON / OFF	Trigger operation at rising edge from 0 to 1
INTMSK	8253 Mask interrupts from Ch.2.	0: Interrupt disabled 1: Interrupt enabled
WDATA	Output to data recorder	
SOUNDMSK (MZ-1500)	8253 Mask the sound output of Ch.0	0: Mask 1: Do not mask

E003h W

8255 control

- Bitset / reset

Bit	7	6	Five	Four	3	2	1	0

signal	0	Port C bit number
		0: Reset 1: Set

- Mode set
  * Set to 8Ah for MZ-700 / 1500

Bit	7	6	Five	Four	3	2	1	0
signal	1	ModeA	PortA	PortCH	ModeB	PortB	PortCL	

ModeA	Port A operating mode	1x: Mode 2 01: Mode 1 00: Mode 0
ModeB	Port B operating mode	1: Mode 1 0: Mode 0
PortA	Port A input / output	0: Output 1: Input
PortB	Port B input / output	0: Output 1: Input
PortCH	Port C upper nibble input / output	0: Output 1: Input
PortCL	Port C lower nibble I / O	0: Output 1: Input

E004h	R / W	8253 Ch.0 Counter read / write
E005h	R / W	8253 Ch.1 Counter Read / Write
E006h	R / W	8253 Ch.2 Counter read / write
E007h	W	8253 Control

Bit	7	6	Five	Four	3	2	1	0
signal	SC1	SC0	RL1	RL0	M2	M1	M0	BCD

SC1, SC0	Counter selection	00: Ch.0 01: Ch.1 10: Ch.2 11: Invalid
RL1, RL0	Read / Load	00: Counter latch operation 01: Access the lower byte of the counter 10: Access the upper byte of the counter 11: Access the lower byte of the counter → the upper byte consecutively

		M2, M1, M0	Mode selection	000: Mode 0 (Interrupt on Terminal Count) 001: Mode 1 (Programmable One-Shot) x10: Mode 2 (Rate Generator) x11: Mode 3 (Square Wave Generator) 100: Mode 4 (Software Triggered Strobe) 101: Mode 5 (Hardware Triggered Strobe)
		BCD	Count mode	0: 16 Bit Binary 1: 4 Digit BCD

E008h

R

Bit	7	6	Five	Four	3	2	1	0
signal	$\overline{\text{HBLK}}$			JB2	JB1	JA2	JA1	TEMPO

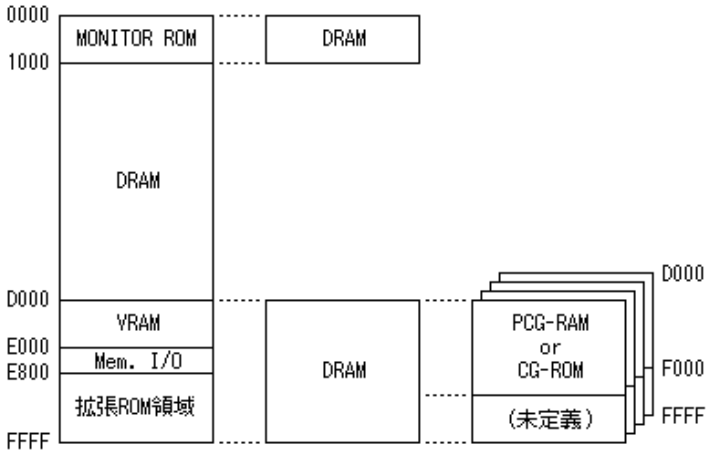
$\overline{\text{HBLK}}$	Horizontal blanking signal
JA2, JA1	Joystick A input
JB2, JB1	Joystick B input
TEMPO	Tempo timer input

W

Bit	7	6	Five	Four	3	2	1	0
signal								GATE

GATE 8253 To Ch.0 GATE input

Memory map



PCG bank is MZ-1500 only.
The extended ROM area is undefined in the MZ-700 (free area. ROM on the board such as MZ-1R12, MZ-1E05, MZ-1E14 is used), and the second monitor ROM 9Z-502M is

placed in the MZ-1500. There is.
The memory-mapped I / O area is actually reserved only for E000h-E00Fh, and E000h-E00Bh is used. E00Ch-E00Fh is undefined. E010h-E7FFh is undefined in MZ-700, but MZ-1500 can read FFh written in ROM.
PCG bank F000h-FFFFh is undefined. When the PCG bank is switched to CGROM, the actual contents of CGROM are in the range of D000h-DFFFh, and the range of E000h-EFFFh is the mirror image.

Memory bank switching

<table>
<tr><th rowspan="2">address</th><th rowspan="2">R / W</th><th colspan="2">Description</th></tr>
<tr><th>0000h-0FFFh</th><th>D000h-FFFFh</th></tr>
<tr><td>E0h</td><td rowspan="7">W</td><td>DRAM</td><td>---</td></tr>
<tr><td>E1h</td><td>---</td><td>DRAM</td></tr>
<tr><td>E2h</td><td>MONITOR ROM</td><td>---</td></tr>
<tr><td>E3h</td><td>---</td><td>VRAM / KEY / TIMER</td></tr>
<tr><td>E4h</td><td>MONITOR ROM</td><td>VRAM / KEY / TIMER</td></tr>
<tr><td>E5h</td><td>---</td><td>MZ-1500: PCG bank switching
<table>
<tr><th>Bit</th><td>7</td><td>6</td><td>Five</td><td>Four</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><th>signal</th><td colspan="5"></td><td colspan="3">BANK</td></tr>
</table>
<table>
<tr><th>BANK</th><th>Description</th></tr>
<tr><td>00</td><td>CGROM</td></tr>
<tr><td>01 01</td><td>PCG blue plane</td></tr>
<tr><td>Ten</td><td>PCG red plane</td></tr>
<tr><td>11</td><td>PCG green plane</td></tr>
</table></td></tr>
<tr><td>E6h</td><td>---</td><td>MZ-1500: Close PCG bank</td></tr>
</table>

* E1h and E3h do not close the PCG bank, but the DRAM bank and VRAM / KEY / TIMER bank are switched behind the scenes, which is reflected when the PCG bank is closed with E6h. For example, if D000h-FFFFh is in the DRAM bank state and OUT as E5h → E3h → E6h, it will be in the VRAM / KEY / TIMER bank state.
E4h closes the PCG bank.

Printer interface

MZ-700

address	R / W	Description
FEh	R / W	Printer control

		Bit 7 6 Five Four 3 2 1 0 **R / W** W W R R R R **signal** RDP IRT 0 SNS STA RDA * SNS is when "printer switching switch is built-in side" and "plotter printer is not connected" 1
FFh	W	Printer data output **Bit** 7 6 Five Four 3 2 1 0 **signal** Data

MZ-1500

address	R / W	Description
FCh	W	Z80PIO port A control
FDh	W	Z80PIO port B control
FEh	R / W	Printer control / timer interrupt (Z80PIO port A data) **Bit** 7 6 Five Four 3 2 1 0 **R / W** W W R R R R R R **signal** RDP IRT INT1 INT0 0 0 STA RDA **INT0** 8253 OUT0 output **INT1** 8253 OUT2 output RDP and IRT can be inverted with DIP SW
FFh	W	Printer data output (Z80PIO port B data) **Bit** 7 6 Five Four 3 2 1 0 **signal** Data

---

Text / PCG priority

address	R / W	Description
F0h	W	priority **Bit** 7 6 Five Four 3 2 1 0 **signal** Priority PCG display **Priority** 0: Text text color> PCG> Text background color 1: PCG> Text

		PCG display	0: Hide PCG 1: Show PCG

palette

address	R / W	Description
F1h	W	Palette settings

Bit	7	6	Five	Four	3	2	1	0
signal		Palette number				Color code		

PSG SN76489AN

address	R / W	Description
F2h (L-Ch.) F3h (R-Ch.) E9h (simultaneous)	W	1st byte

Bit	7	6	Five	Four	3	2	1	0
signal	1	R2	R1	R0	D3	D2	D1	D0

R2	**R1**	**R0**	**Control register allocation**
0	0	0	Tone 0 frequency
0	0	1	Tone 0 volume
0	1	0	Tone 1 frequency
0	1	1	Tone 1 volume
1	0	0	Tone 2 frequency
1	0	1	Tone 2 volume
1	1	0	Noise control
1	1	1	Noise volume

- Frequency (division ratio)
  1st byte

Bit	7	6	Five	Four	3	2	1	0
signal	1	R2	R1	R0	Division ratio n (lower 4 bits)			

  2nd byte

Bit	7	6	Five	Four	3	2	1	0

signal	0		Division ratio n (upper 6 bits)

Frequency f (Hz) = $3.579545 \times 10^{6} / (32 \times n)$

- Volume

Bit	7	6	Five	Four	3	2	1	0
signal	1	R2	R1	R0	Attenuation amount			

D3	**D2**	**D1**	**D0**	**Attenuation (dB)**	**Attenuation ratio (reference)**
0	0	0	0	0	1.000000
0	0	0	1	2	0.794328
0	0	1	0	Four	0.630957
0	0	1	1	6	0.501187
0	1	0	0	8	0.398107
0	1	0	1	Ten	0.316228
0	1	1	0	12	0.251189
0	1	1	1	14	0.199526
1	0	0	0	16	0.158489
1	0	0	1	18	0.125893
1	0	1	0	20	0.100000
1	0	1	1	twenty two	0.079433
1	1	0	0	twenty four	0.063096
1	1	0	1	26	0.050119
1	1	1	0	28	0.039811
1	1	1	1	OFF OFF	0.000000

- Noise control

Bit	7	6	Five	Four	3	2	1	0
signal	1	1	1	0		Type	frequency	

Type	0: Sync noise 1: White noise
frequency	00: 3.579545×10^{3} /512 (kHz)

		01: 3.579545×10^{3} /1024 (kHz) 10: 3.579545×10^{3} /2048 (kHz) 11: tone generator 2 outputs

MZ-1R18 RAM file (64K)

address	R / W	Description
EAh	R	Data read / address counter +1
	W	Data write / address counter +1
EBh	W	Address counter write upper byte = {A15-A8}, lower byte = {D7-D0}

MZ-1M08 Voice board

address	R / W	Description
E8h	W	Send to voice board **Bit** 7 6 Five Four 3 2 1 0 **signal** ACL / N1 BUSY DATA **ACL** Reset bit — 0: Normal operation, 1: Reset voice board **/ N1** ROM specification — 0: Internal ROM, 1: External ROM **BUSY** Handshake bit **DATA** Transmission data
	R	Acnoridge from voice board **Bit** 7 6 Five Four 3 2 1 0 **signal** ACK

How to send data

Follow the steps below to serially transfer voice numbers from Bit7 to Bit0. Do not change / N1 during transfer
Secure a reset (ACL = 1) period of 10 ms or more.

1. Check ACK == 0, set BUSY = 1 and send DATA
2. Check ACK == 1 and set BUSY = 0

* After sending Bit0, ACK == 0 is after the end of pronunciation. After ACK == 0, set BUSY = 0

Internal ROM

Lower \ Upper	0	1	2	3
0	Oh	Ten	Zero	Chime 1
1	One	Eleven	Time is up	Chime 2
2	Two	Twelve	to	Chime 3
3	Three	Thirteen	it's	Chime 4
Four	Four	Fourteen	it's now	256ms off
Five	Five	Fifteen	go to	16ms off
6	Six	Sixteen	Pun	Rest 192ms
7	Seven	Seventeen	Pip	
8	Eight	Eighteen	Pip pip	
9	Nine	Nineteen	Pup	
A	Hours	AM	Poon	
B	Hour	PM	Pip	
C	Minutes	Twenty	Pip pip pip pip	
D	Minute	Thirty	Alarm	
E	8ms off	Forty	Alarm set for	
F	8ms off	Fifty	(Melody)	

External ROM

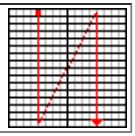
\ Upper and lower \	0	1	2	3	Four	Five	6	7	8	9	A	B	C	D	E	F
0	Used after zero 0 minutes	Huh	SP 128ms	0 zero	Queue **window**	P	Well it can be directly below	Add	D-Day	~ Progress	Bya	Time	Ta	Mi	Pya	De

1	Minute shit	Juu (A)	8ms off	1 Ichi	A	Q	laughter	Pull	~ 100 million	Nya	View	A	Ji	Mu	Pyu	Do
2	Minute pun	Juu (C)	16ms off	2 **D**	B	R	Game start	Itetet	Ca	New	Byo	I	Tsu	Me	Pyo	Pa
3	Pause large 200ms	Ju (Ten)	32ms off	3 service **down**	C	S	Game over	It!	Kyu	Nyo	Lya	C	Te	Mo	Moth	Jeong
Four	80ms during hibernation	~ Ju (Ten)	48ms off	4 **Yo** down	D	T	out	Car (all)	Kyo	a.m.	Ryu	D	To	Ya	Gi	Pu
Five	Pause small 40ms	Ten	Rest 64ms	5 **go**	E	U	GO	Car (start)	D (for clock "hour")	afternoon	Ryo	Oh	Na	Yu	Gu	Pe
6	Pup	~ Juu (A)	160ms off	6 Roku	F	V	hit	Car (running)	Lot (for watches)	Hya	Wo	Mosquitoes	D	Yo	Ge	Po
7	Pun	~ Juu (B)	256ms off	7 **Na** Na	G	W	Your favorite the key and press please	Car (ending)	Sha	Huh	Ga	Ki	Nu	La	Go	Pa
8	Pip	Hyaku	Holiday 320ms	8 bees	H	X	Of the Y or N key and press please	Melody funeral march	Shu	Hyo	Gyu	Ku	Ne	Ri	The	Bi
9	D (for clock "minutes")	~ Hyaku (A)	Is	9 **key** oxalic	I	Y	Highest score	Melody Dixieland	Show	Yo (hour)	Gyo	Ke	No	Ru	The	Bu
A	Go (for watches)	~ Hyaku (B)	Call	D	J	Z	Bonus points	Melody Hotaru no Hikari	She	Wah (hour)	Tee	Ko	Ha	Re	Zu	Pe
B	Go (for 50 minutes)	~ Pyak	plus	**Sa** down	K	Go	It's your turn	Melody Oh! Susanna	knee	Mya	Fa	Sa	Hi	B	Ze	Bo
C	~ Ju (for watches)	~ Byaku	E	Yo **down**	L	I 'm wrong	my	Melody Hometown People	Cha	Mu	Ja	Shi	Fu	Wa	Zo	SLOW
D	[CR]	~ Sen	minus	equal	M	Is the correct answer	It's a grammatical error	Blacksmith in Melody Village	Chu	Myo	Ju	Su	He	N	Da	

E	It	~ Zen	Marten	Go	N	OK	I am MZ	Pause 480ms	Cho	~ Used for time 2,4,5,9	Jo	Se	Ho	Pause 1120ms	Di	MIDD
F	Lot	~ Man	Divide	Na **na**	O	Once again challenge Are you sure you want to	point	Pause 340ms	Choi	~ Used for time 1,3,6,7,8	Je	So	Ma	Pause 290ms	Zu	FAST

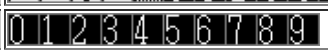
* 0xFC, 0xFE, 0xFF are control codes for changing the playback speed and are valid until reset.

MZ-1R23 Kanji ROM board / MZ-1R24 Dictionary ROM board

<table>
<tr><th>address</th><th>R / W</th><th>Description</th></tr>
<tr><td>B8h</td><td>W</td><td>Control port
<table>
<tr><th>Bit</th><td>7</td><td>6</td><td>Five</td><td>Four</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><th>signal</th><td>KANJI</td><td>ENDIAN</td><td colspan="4"></td><td colspan="2">BANK</td></tr>
</table>
<table>
<tr><td>KANJI</td><td>Kanji ROM / dictionary ROM selection</td><td>0: Dictionary ROM
1: Kanji ROM</td></tr>
<tr><td>ENDIAN</td><td>Bit sequence of the pattern to be read</td><td>0: Bit7 is on the left, Bit0 is on the right
1: Bit0 is on the left, Bit7 is on the right</td></tr>
<tr><td>BANK</td><td>Dictionary ROM bank selection</td><td>00-11</td></tr>
</table></td></tr>
<tr><td rowspan="2">B9h</td><td>W</td><td>Kanji pattern number / dictionary ROM address
Upper byte = {A15-A8}, Lower byte = {D7-D0}</td></tr>
<tr><td>R</td><td>The data read (auto increment)
pattern is read in the order of upper left → lower left, upper right → lower right (right figure).
  </td></tr>
</table>

Kanji pattern number = (JisH --0x21-((JisH> = 0x30)? 0x08: 0x00)) * 94 + (JisL --0x21)

Some are different from the standard JIS code. There is no pattern corresponding to 2821 ~ 2F7E

222F-2239	
2240-225F	
2276-227D	
2321 ~ 232F	
2475-247E	

2577-257E	
2659-265F	
2661-267A	Uppercase letters A to Z (half-width)
2746-274F	
2774-277E	
4F54-4F5F	譲 赳 Discipline Vehicle Congestion Congestion 隨 遽 邁 鄂 鈑
4F60-4F6F	錮 鈑 鑽 隘 鞏 頌 騒 Hair dust 麒 齟 齬 俯 會 僣 剎
4F70-4F7E	庞 Country dust vocabulary 怱 扣 抒 旁 旁 漱 view rattan rattan 羞 SHO glue
5021-5029	Scissors, pungent, pungent, pungent

MZ-1R12 CMOS Battery Backup Memory (32K)

address	R / W	Description
F8h	R	Reset address counter to 0000h
	W	Write high-order byte of address counter
F9h	R	Data read / address counter +1
	W	Write lower byte of address counter
FAh	W	Data write / address counter +1

* The base address can be specified by DIP SW. The address that can be used as a boot device on the MZ-700 is F8h. The address that can be used as a boot device on the MZ-1500 is F8h / A8h (only A8h when installed in the expansion unit MZ-1U08).

-MZ-1R12 automatic startup program storage format

offset	data
+00	Program storage size (L)
+01	Program storage size (H)
+02	Load address (L)
+03	Load address (H)
+04	Execution address (L)
+05	Execution address (H)
+06	Data part checksum (L) * 1
+07	Data part checksum (H) * 1

+08	Header checksum * 2
+ 09 ~	Program body (size specified by offset +00, +01)

* 1 The number of bits '1' is counted by 16 bits for each byte of the program body.
* 2 The number of bits '1' is counted by 8 bits for each byte of the header part (+00 to +08). thing

---

MZ-1E24 / MZ-8BI03 RS-232C board

address	R / W	Description
B0h	R / W	Z80SIO channel A data
B1h	R / W	Z80SIO Channel A Control
B2h	R / W	Z80SIO channel B data
B3h	R / W	Z80SIO Channel B Control

* The base address can be specified by DIP SW.
The owner's manual of MZ-1500 instructs
BASIC 5Z- to specify D0h when installing in the expansion slot of the main unit and B0h when installing in the expansion unit MZ- 1U08. 001 supports device names RS1 (B0h, Ch.A) RS2 (B0h, Ch.B) RS3 (D0h, Ch.A) RS4 (D0h, Ch.B)

QD I / F (Z80A-SIO)

address	R / W	Description
F4h	R / W	SIO channel A data
F5h	R / W	SIO channel B data
F6h	R / W	SIO channel A control
F7h	R / W	SIO channel B control

SIO signal	I / O	QD signal
$\overline{\text{DCDA}}$	I	Media switch L: Media is set H: Media is not set
$\overline{\text{CTSA}}$	I	Write protect L: Writable H: Write protected
$\overline{\text{RTSA}}$	O	WRITE GATE Output L: Write

$\overline{\text{DCDB}}$	I	$\overline{\text{HOME}}$ Signal L: Head is in read / write valid area
$\overline{\text{DTRB}}$	O	Motor on H → L: Motor on H: Motor off when the motor stop point is reached
$\overline{\text{RTSB}}$	O	MFM Demodulator Circuit VFO Enable L: Read

MZ-1E05 FDD I / F (MB8876)

address	R / W	Description
D8h	R	Status register
	W	Command register
D9h	R / W	Track register
DAh	R / W	Sector register
DBh	R / W	Data register
DCh	W	Motor / drive select control register **Bit** 7 6 Five Four 3 2 1 0 **signal** M-ON DSEN DS **M-ON** FDD motor control: 0: OFF, 1: ON **DSEN** Drive select enabled?: 0: Drive select disabled, 1: Select the drive with the number specified by DS **DS** Drive select?: Drive number to select
DDh	W	Head select control register **Bit** 7 6 Five Four 3 2 1 0 **signal** HS **HS** Head select: 0: Side 0, 1: Side 1

PCG-700 (HAL Laboratory)

<table>
<tr><th>address</th><th>R / W</th><th>Description</th></tr>
<tr><td>E010h</td><td>W</td><td>Pattern data to be written to PCG-RAM
<table>
<tr><th>Bit</th><td>7</td><td>6</td><td>Five</td><td>Four</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><th>signal</th><td colspan="8">Pattern data</td></tr>
</table></td></tr>
<tr><td>E011h</td><td>W</td><td>PCG-RAM address lower 8 bits
<table>
<tr><th>Bit</th><td>7</td><td>6</td><td>Five</td><td>Four</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><th>signal</th><td colspan="8">ADDR [7: 0]</td></tr>
</table></td></tr>
<tr><td>E012h</td><td>W</td><td>PCG-RAM address upper 3 bits / PCG control
<table>
<tr><th>Bit</th><td>7</td><td>6</td><td>Five</td><td>Four</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><th>signal</th><td></td><td colspan="2">COPY</td><td>WE</td><td>SSW</td><td colspan="2">ADDR [10: 8]</td></tr>
</table>
<table>
<tr><td>COPY</td><td>Copy from CGROM</td><td>0: When writing with WE, write the pattern data written in E010h
1: When writing with WE, write the pattern data from CGROM</td></tr>
<tr><td>WE</td><td>Write operation</td><td>Write data to PCG-RAM by toggle from 0 → 1 → 0</td></tr>
<tr><td>SSW</td><td>PCG selection</td><td>0: Use PCG
1: Do not use PCG</td></tr>
<tr><td>ADDR</td><td>Write address</td><td>000h-7FFh</td></tr>
</table></td></tr>
</table>

The range of display code 80h-FFh (128 x 2 = 256 characters including switching of attribute VRAM by Bit7) is displayed on PCG.
Note that even if the banks of D000h to FFFFh are in the DRAM state, they will react to writing to E010h to E012h (do not leave SSW at 1).

PIO-3034 320K EMM (IO DATA)

address	R / W	Description
00h	W	Address counter Bit [7: 0]
01h	W	Address counter Bit [15: 8]
02h	W	Address Counter Bit [18:16]
03h	R	Data read / address counter +1
	W	Data write / address counter +1

* Base address can be specified by DIP SW ・ HuBASIC supports 00h-03h

---

Celestite LAN / Memory Composite Board (Oh! Made by Mr. Ishi)

<table>
<tr><th>address</th><th>R / W</th><th>Description</th></tr>
<tr><td>60h</td><td>W</td><td>W5100 MR (Mode Register)</td></tr>
<tr><td>61h</td><td>W</td><td>W5100 IDM_AR0 (upper address register)</td></tr>
<tr><td>62h</td><td>W</td><td>W5100 IDM_AR1 (lower address register)</td></tr>
<tr><td>63h</td><td>R / W</td><td>W5100 IDM_DR (data read / write / address +1)</td></tr>
<tr><td>64h</td><td>R / W</td><td>Interrupt vector</td></tr>
<tr><td rowspan="2">65h</td><td>W</td><td>Interrupt control
<table>
<tr><th>Bit</th><td>7</td><td>6</td><td>Five</td><td>Four</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><th>signal</th><td colspan="5"></td><td colspan="3">ENABLE</td></tr>
</table>
<table>
<tr><td>ENABLE</td><td>Interrupt enabled</td><td>0: Prohibition
1: Allow</td></tr>
</table></td></tr>
<tr><td>R</td><td>Interrupt status
<table>
<tr><th>Bit</th><td colspan="2">7</td><td colspan="2">6</td><td>Five</td><td>Four</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><th>signal</th><td colspan="2">Interrupt input</td><td colspan="2">Interrupt reception</td><td colspan="4"></td><td>ENABLE</td></tr>
</table>
<table>
<tr><td>???</td><td>Interrupt input</td><td>0: Not in
interrupt request 1: In interrupt request</td></tr>
<tr><td>???</td><td>Interrupt reception</td><td>0: Interrupt acknowledge cycle not detected
1: Interrupt acknowledge cycle detected</td></tr>
<tr><td>ENABLE</td><td>Interrupt enabled state
(written value)</td><td>0: Prohibition
1: Allow</td></tr>
</table></td></tr>
<tr><td>66h</td><td>R / W</td><td>Memo register</td></tr>
<tr><td rowspan="2">68h</td><td>W</td><td>UFM (User Flash Memory) Address Register</td></tr>
<tr><td>R</td><td>UFM (User Flash Memory) Status
<table>
<tr><th>Bit</th><td>7</td><td>6</td><td>Five</td><td>Four</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><th>signal</th><td>WP</td><td colspan="4"></td><td>READY</td><td>BUSY</td></tr>
</table>
<table>
<tr><td>WP</td><td>Write-protect</td><td>0: Write-protected
1 : Write-protected</td></tr>
<tr><td>READY</td><td>UFM data valid</td><td>0: Data invalid
1: Valid data corresponding to the address is in 69h</td></tr>
<tr><td>BUSY</td><td>UFM busy state</td><td>0: UFM operation possible
1: UFM operation not possible</td></tr>
</table></td></tr>
</table>

<table>
<tr><td>69h</td><td>R / W</td><td>UFM (User Flash Memory) Data Register

Lead

1. Set UFM address (68h)
2. Wait for the UFM status (68h) READY to be 1
3. Read UFM data (69h)

Light

1. Write D1h, 57h to the unlock register (6Fh) to erase the entire UFM
2. Wait for the UFM status (68h) BUSY to reach 0
3. If UFM is in write protect state, write D1h, 0Fh in the unlock register (6Fh) to release write protection.
4. Set UFM address (68h)
5. Wait for the UFM status (68h) READY to be 1
6. Write data to UFM data (69h)
7. Wait for the UFM status (68h) BUSY to reach 0
8. Repeat the process from 4. for the number of data you want to write
9. Write D1h, 0Fh to the unlock register (6Fh) and apply write protection.</td></tr>
<tr><td>6Fh</td><td>W</td><td>Unlock (Board function control)
Each function is executed by writing the following keywords after D1h.
<table>
<tr><th>keyword</th><th>function</th><th>At reset</th></tr>
<tr><td>12h</td><td>Double the capacity of MZ-1R12 compatible CMOS RAM to 64KB</td><td>32KB</td></tr>
<tr><td>37h</td><td>Enable EMM for MZ-2500 (only when disabled by DIP-SW)</td><td>Follow the DIP-SW settings</td></tr>
<tr><td>05h</td><td>Remove write protection for the expansion ROM and FD ROM areas for the MZ-700</td><td>Write-protect</td></tr>
<tr><td>57h</td><td>Erase UFM area (Erase even when write protected)</td><td></td></tr>
<tr><td>0Fh</td><td>Invert write protection to UFM area</td><td>Write-protect</td></tr>
</table></td></tr>
</table>

MZ-1R37 640K EMM

address	R / W	Description
ACh	W	Address latch write address [19:16] = {A15-A8}, address [15: 8] = {D7-D0}
ADh	R	Data read address [7: 0] = {A15-A8}, data = {D7-D0}
	W	Data write address [7: 0] = {A15-A8}, data = {D7-D0}

* There is no automatic address increment

An option for the MZ-2500, but supported by Celestite LAN / Memory Composite Boards. Since the EMM area is used as the MZ-1R23 Kanji ROM area on the MZ-1500, these ports are mainly used for initializing the Kanji ROM area and the extended ROM area (E800h-FFFFh).

The extended ROM area is mapped to EMM 0FE800h to 0FFFFFh.

---

go back