PIN DESCRIPTION

Pin	Name	Input/ Output	Functions	Internal reset state	Pin structure
V_{DD} V_{SS}	Supply voltage		Power supply inputs 5V±10% to V_{DD}, and 0V to V_{SS}.		
OSC IN	Input for oscillating circuit	Input	These are I/O pins of internal clock oscillating circuit. Depending on the reference signal, connect either a quartz crystal resonator (4MHz) and capacitors to these pins.		Hysteresis characteristics exists
OSC OUT	Output for oscillating circuit	Output			CMOS output
F_0~F_3	Port F	Output	This port outputs data from the register A.	"L"	CMOS output
J_0~J_3	Port J	Output	This port outputs data from the register A.	"H"	CMOS output
K_0~K_3	Port K	Output	This port outputs data from the register A.	"H"	N-channel open-drain high-voltage output (12V)
H_0~H_3	Port H	Output	①This port outputs data from the register A. ②This port can be connected to LED directly.	"H"	N-channel open-drain high-voltage output (12V)
M_0~M_3	Port M	Output	This port outputs data from the lower 3 bits of the register A.	"H"	N-channel open-drain high-voltage output (12V)
E_0~E_3	Port E	Input	External applied signal's level is read into the register A.		Built-in pull-up resistor (option)
G_0~G_3	Port G	Output	①This port outputs data from the register A.	"H"	Built-in pull-up resistor (option)
		Input	②External applied signal's level is read into the register A.		
L_0~L_3	Port L	Output	①This port outputs data from the register A.	"H"	Built-in pull-up resistor (option)
		Input	②External applied signal's level is read into the register A.		
			③L_0 can be used as an input pin for the HS counter.		
D	Port D	Output	①This port outputs data from the most significant bit of the register A.	Input state	CMOS tri-state I/O
		Input	②External applied signal's level is read into the carry flag.		
A-D	Comparative voltage input	Input	This pin is comparative voltage input for the 3-bit A-D converter.		
TEST	Input for test	Input	This is usually connected to V_{SS} and supply "L" (0V).		

$\overline{RESET}$	Reset input	Input	If a "L" (0V) input is applied, the reset state is entered. If the input is change from "L" to "H", the internal reset continues for at least 3 machine cycles. For normal use, apply a "H" input. Also, when the power is initially applied, an external circuit is required for reset.		Built-in pull-up resistor (option), and hysteresis characteristics exists
$\overline{D\text{-}A}$	14-bit PWM output	Output	①According to the 14-bit digital information, the PWM signal is output. The repeated frequency is 122Hz, and the minimum pulse width is 500ns. And, the input polarity can be changed by selecting the option. ②This pin can be used as an 1-bit output port.	The output state is undefined until the data is input into D-A latch.	CMOS output
VDP_0 VDP_1 VDP_2	7-bit PWM output	Output	①According to the 7-bit latched data, the PWM signal (which has 65 levels) are output. The frequency is 1kHz, and the minimum pulse width is 16μs. And, the input polarity can be changed by selecting the option. ②This pin can be used as 1-bit output port.	VDP_0 is not active state, the other state is undefined until the data is input into D-A latch.	N-channel open-drain high-voltage output (12V)
$\overline{INT}$	Interrupt input	Input	This pin used when an external interrupt is applied to the CPU.		Built-in pull-up resistor (option)
Hsync	Horizontal synchronizing signal input for display	Input	This is the horizontal synchronizing signal input pin for TV applications. The input polarity can be changed by selecting an option.		
Vsync	Vertical synchronizing signal input for display	Input	This is the vertical synchronizing signal input pin for TV applications. The input polarity can be changed by selecting an option.		
OSC1	Oscillator input for display	Input	A display clock can be obtained by connecting a variable resistor (described later) and a capacitor, or both a coil and capacitors, between these pins.		
OSC2	Oscillator output for display	Output			CMOS output
R, G, B	R, G, B output	Output	These pins control the color display data. The output polarity can be changed by selecting an option.	Not an active state.	CMOS output
OUT	OUT output	Output	This is a logical OR output for the R,G,B pins. The output polarity can be changed by selecting an option.	Not an active state.	CMOS output

BASIC FUNCTION BLCOKS

14-BIT PWM OUTPUT

(1) $\overline{\text{D-A}}$ port

The 14-bit PWM output port ($\overline{\text{D-A}}$) outputs the pulse width modulated (PWM) signals. It can also be used as a tuning-voltage output port.

As shown in Figure 1, the $\overline{\text{D-A}}$ port has a 14-bit D-A latch and a 15-bit D-A latch. Altogether, the 15-bits of data (addresses M(0,0), M(0,1), M(0,2), and except for the most significant bit of address M (0,3)) are transfered to the D-A latch by the DATA instruction. According to the lower 14-bit digital information, the pulse width modulated signals are output from the $\overline{\text{D-A}}$ port. By performing D-A conversion on these output rectangle waves (via the integration circuit), a tuning voltage can be generated.

And, the $\overline{\text{D-A}}$ port can be used as a 1-bit output by controlling the upper 1-bit.

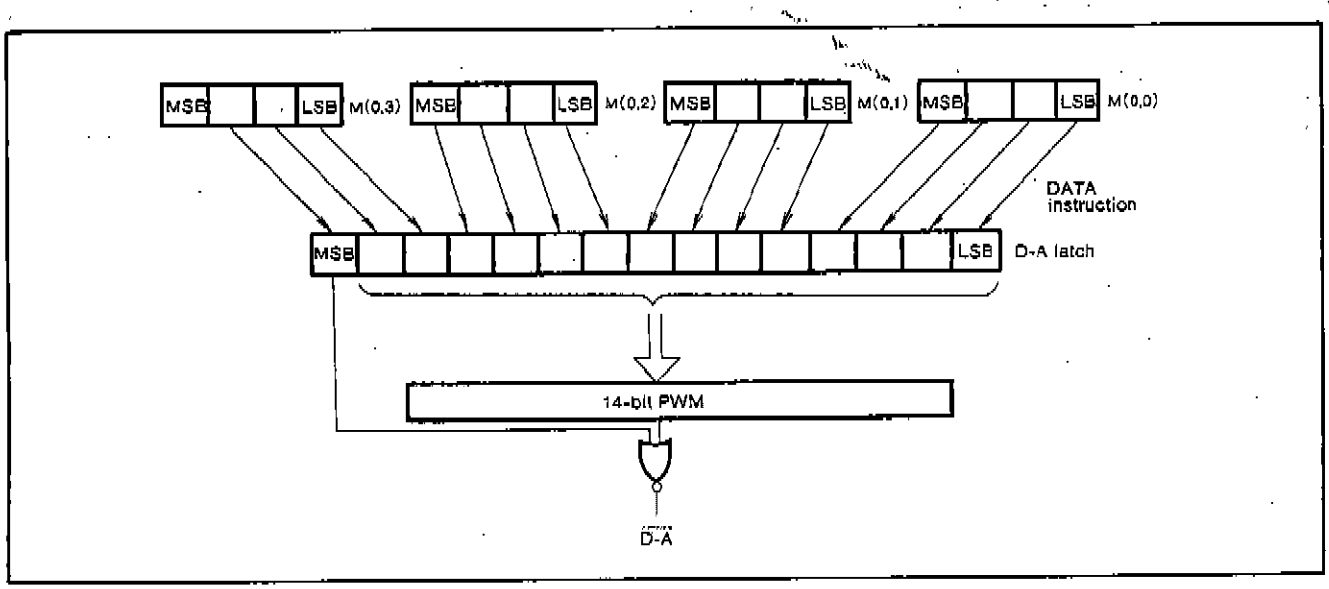

Fig.1 RAM, D-A lacth, 14-bit PWM, and D-A port

(2) $\overline{\text{D-A}}$ port output signals

The periodic interval (T_0, about 122 Hz) can be divided into 2^{14} minimum pulse widths (t_0, 500ns), and the pulse width can be modulated by that t_0 unit depending on the 14-bit data.

Also, by creating a small periodic interval (T_S, about 7.8kHz with $64T_S = T_0$), which is 256 times t_0, pulses of almost equal width can be output with a period of T_S.

T_m ($m = 1 \sim 64$), defined as the signal duration in 64 small intervals, is determined as follows: First, the 14-bit data is split into two parts: the lower 6 bits and the upper 8 bits. When the number designated by the lower 6 bits reaches 64, it is moved up one place. So T_m ($1 \sim 64$) = (number indicated by upper 8 bits) $\times$ t_0 in the 64 small intervals. Furthermore, as many t_0's as are indicated by the lower 6 bits are added one by one in the 64 small intervals. The relationship between this lower 6-bit data and T_m is shown in Table 1.

Table 1. Relationship between the Lower 6-bit Data and Tm

Lower 6-bit data (LSB)	Interval longer (by t_0) than other Tm's (m=1~64)
0 0 0 0 0 0	None
0 0 0 0 0 1	m=32
0 0 0 0 1 0	m=16, 48
0 0 0 1 0 0	m=8, 24, 40, 56
0 0 1 0 0 0	m=4, 12, 20, 28, 36, 44, 52, 60
0 1 0 0 0 0	m=2, 6, 10, ·····58, 62
1 0 0 0 0 0	m=1, 3, 5, ·····61, 63

The 64 cases from 000000 to 111111 can be expressed by the seven combinations shown in the Table 1. Also, by combining the upper 8 bits, 2^{14} sets from 00000000000000 to 11111111111111 can be expressed.

If the upper 8 bits of the 14-bit are 00000011, and the lower 6 bits are 000101, (for example, T_8, T_{24}, T_{32}, T_{40}, and T_{56}, $= 4t_0$) and all the other T_m's become "L" during $3t_0$. Figure 2 shows this example.

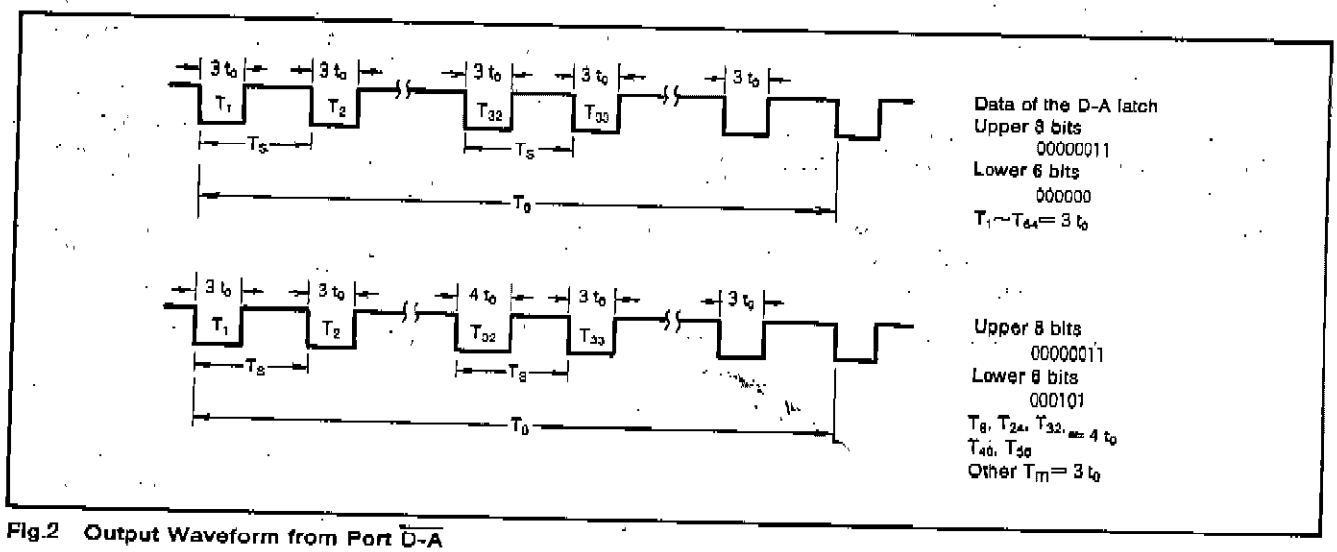

Fig.2 Output Waveform from Port D-A

Thus, those duration times of T_m's (of the level in each small interval) can coincide or become almost equal with at most t_0 difference. This periodic time of the small interval (T_S, about 7.8kHz) approximates the repeated frequency.

After the RAM data at the specifc address is set, the output pin status will still be unstable untill the DATA instruction is executed.

7-BIT PWM OUTPUT

(1) VDP port

The 7-bit PWM output port (VDP) is used to output the pulse width modulated (PWM) signals. It can also be used as an output port for various analog data control by using lowpass filter. A 7-bit analog latch is connected brough PWM block to the VDP port. The lower 3 bits of register B and the 4-bit data of register A are transfered to the analog latch by the $VDP_{0\sim2}$ instructions. According to this 7-bit digital information, the pulse width modulated signals (changeable to 64 levels) are output from the VDP port thus allowing various analog data to be controlled.

(2) Output waveform from the VDP port

According to the 7-bit analog latch data, the VDP port outputs pulses, whose widths have been modulated by every minimum pulse width (t) (16μs), in every periodic time interval (T) (about 1kHz of repeated frequency). T can be divided into 64 sets of t. An example of this is shown in Table 2. Thus, Pulse-width-modulated signals from 0000000 to 1XXXXXX can be obtained.

When the VDP port output needs to be changed, enter the corresponding data of the desired pulse width output into register A and B. Then transfer this data to the analog latch using the $VDP_{0\sim2}$ instructions.

Table 2. Relationship between the 7-bit Data and the Output Waveform from Port VDP

7-bit data (LSB)	tm (m=1~64) is become "H" in the period T (=64t)
0 0 0 0 0 0 0	None (all "L")
0 0 0 0 0 0 1	m=32
0 0 0 0 0 1 0	m=16, 48
0 0 0 0 1 0 0	m=8, 24, 40, 56
0 0 0 1 0 0 0	m=4, 12, 20, 28, 36, 44, 52, 60
0 0 1 0 0 0 0	m=2, 6, 10, ·····58, 62
0 1 0 0 0 0 0	m=1, 3, 5, ·····61, 63
1 X X X X X X	m=1~64(all "H")

INTERRUPT

(1) IF flag

The IF (Interrupt Flag) consists of four bits. ED1 sets the polarity of the input signal input to the $\overline{INT}$ pin. ED2 selects two kinds of reference clocks that determine the pulse interval of the input signal to the $\overline{INT}$ pin.

EDF is a bit indicating the EI/DI state A "0" inhibits the interrupt, while a "1" allows the interrupt. The EI/DI state is set by the EI or DI instruction. INTF is a bit indicating the presence or absence of an external interrupt input. This bit is set to "1" when an external interrupt occurs. When in the EI state and an interrupt is accepted, the INTF signal is cleared and set to "0". The EDF signal is automatically set to "0".

The data of the lower two bits of the IF flag is moved from register A by the TIFA instruction and the IF flag is read and transferred to register A by the TAIF instruction.

Table 3. Structure of IF Flag

Flag	Data	Description
ED1 (bit 0)	0	Interrupt occurs at falling edge of INT pin
	1	Interrupt occurs at rising edge of INT pin
ED2 (bit 1)	0	Select 64μs standard clock
	1	Select 128μs standard clock
EDF (bit 2)	0	External interrupt is disabled
	1	External interrupt is enabled
INTF (bit 3)	0	Indicate no existence of interrupt request
	1	Indicate existence of interrupt request

(2) Acceptance of an interrupt

Recognition of an interrupt is made by sensing the pulse edge. About 68μs~128μs (changeable to 136μs to 256μs by the ED2 flag) after the signal applied to the $\overline{INT}$ pin drops, the signal can be recognized as an interrupt. The interrupt request signal (INT) is then generated to set the interrupt flag (INTF/F). Whether this interrupt is accepted or not depends upon the condition of EI/DI flag. If the EI/DI flag is set to "1", an interrupt request can be accepted. On the other hand, if the EI/DI flag is "0", the interrupt request is disabled. At this time, so INTE flag is set, an interrupt is accepted as soon as EI/DI flag is set to "1".

(3) Interrupt control circuit

The interrupt control circuit, shown in Figure 5, is composed of the INTF/F (interrupt flag) and EI/DI flag.

The interrupt request signal (INT) described above, is input to INTF/F. This signal acknowledge the existence of an input. When an interrupt is accepted and jumping to the interrupt processing routine is executed, the INTF/F is automatically reset.

The logical product of outputs (from INTF/F and EI/DI flag) produces an interrupt signal (INT). Once an interrupt is appepted, the program execution automatically jump to address 3, and the next program address to be executed is pushed on the stack register. The DI state is then automatically set.

At reset, the EI/DI flag enters the DI state, INTF/F enters the reset state, and no other interrupt can be accepted.

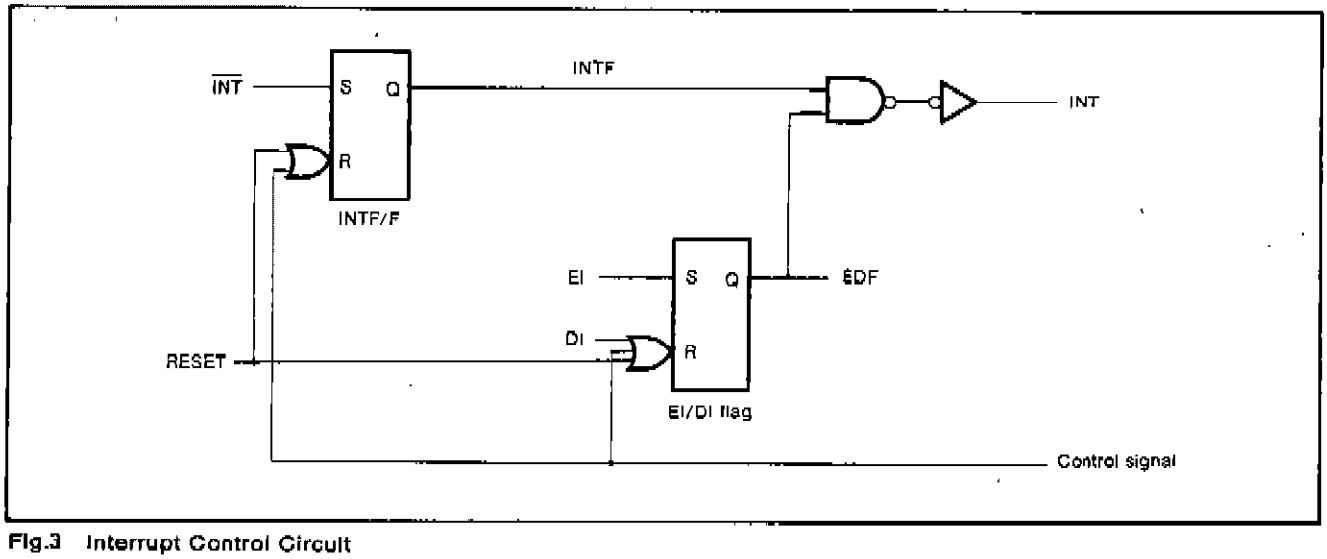

Fig.3 Interrupt Control Circuit

REGISTER I

Register I is a 3-bit read-only register which stores bit data and word-end data of the interrupt request signals. It also stores the input status of the $\overline{INT}$ terminal.

The INT input circuit includes a counter which checks the interval between interrupts. When using remote control, this signal can be input to the $\overline{INT}$ pin.

There are two sets of standard clocks to measure the $\overline{INT}$ pin pulse intervals; one with a 64μs cycle, and one with a 128μs cycle. Those clocks can be switched using the ED2.

By using one of these clocks, the following two decisions can be made; a bit judgement (bit data), expressed as a "1" or "0", indicating whether the pulse interrupt (between the falling or rising edges of the input signals) is longer or shorter than the standard interval (about 1.54ms), respectively. The other is a word decision (word-end) which determines whether one word has been reached. This is done by comparing the pulse interval with the standard time interval of about 3.2ms. If using the 128μs cycle clock, both the time interval described above and that described in "Interval" needs to be doubled (See Table 4).

The bit data described above is stored in bit 0 of register I. If the pulse interval is shorter than the standard time interval, this bit is set to "0". Otherwise, it is set to "1". The word decision data is stored in bit 1 also. Bit 2 of register I stores the input status of the $\overline{INT}$ pin.

The entire data of register I can be read into register A by executing the TAI instruction. Refer to Figure 4.

Table 4. Judgment of Bit Data, Word-end (at 4MHz frequency)

	Clock period	Pulse Interval (T_P)	Bit data	Word -end
Standard time (fast)	64μs	T_P<1.54ms±64μs	0	0
		(1.54ms±64μs)≦T_P<3.2ms±64μs	1	0
		(3.2ms±64μs)≦T_P	1	1
Standard time (slow)	128μs	T_P<3.07ms±128μs	0	0
		(3.07ms±128μs)≦T_P<6.4ms±128μs	1	0
		(6.4ms±128μs)≦T_P	1	1

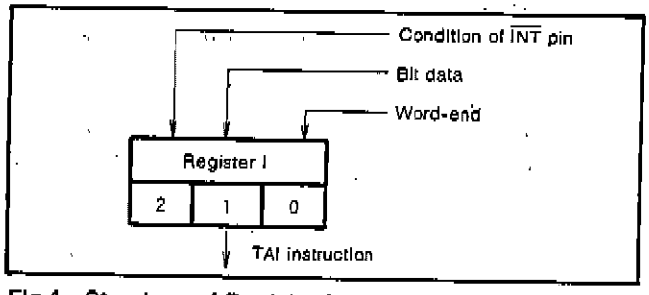

Fig.4 Structure of Register I

HS COUNTER

The HS counter is composed of a 5-bit counter which inputs signals from L_0 of port L and a latch which stores the upper 4 bits of the counter (refer to Figure 5). Using this counter, the existence of video signals can be determined.

The counter is first reset and then begins counting for a period about 1.02ms (255 machine cycles) the number of pulses entered at L_0. It stores this count in the latch during the next 0.004ms (1 machine cycle).

Then, after it is reset, it again starts counting the pulses input to L_0. If the number of pulses entered into L_0 exceeded 32, the counter value becomes "0". While the counter data is being transfered into the latch and the counter being reset, the pulse input to L_0 is disabled.

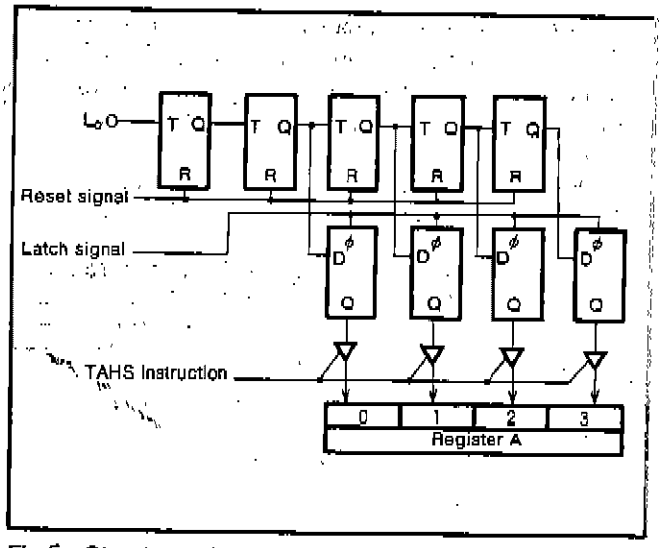

Fig.5 Structure of HS counter

A-D CONVERTER AND REGISTER R

The A-D converter is composed of a 3-bit D-A converter and a comparator. Data is placed in the D-A converter through register R in such a way that the lower 3 bits of register A are transfered to register R by the TRA instruction. After that, the D-A reference voltage (V_{ref}) is set. When the TACM instruction is executed the V_{ref} data, compared with the comparative voltage of the A-D port, is transfered to the least significant bit of register A.

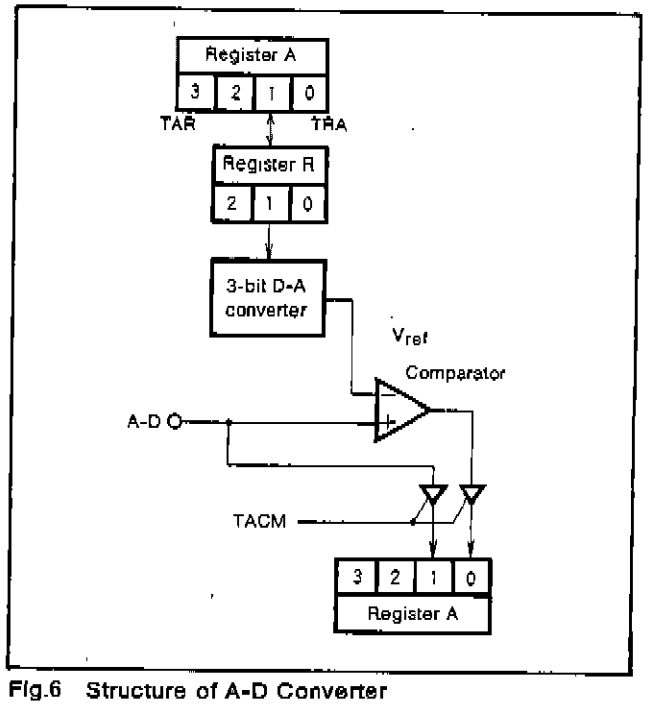

Fig.6 Structure of A-D Converter

The D-A converter can produce 8 different reference voltages (V_{ref}, voltages independent of V_{DD}) depending on the contents of register R. The relationship between the analog voltage (comparative voltage) supplied to the A-D pin and V_{ref} is as follows:

"1", if the comparative voltage > the reference voltage
"0", if the comparative voltage < the reference voltage

Table 5. The Contents of register R and Reference Voltage (Vref)

Register R		Reference voltage
DEC	Binary	Vref(V_{DD}= 5 V)
0	0 0 0	1.12V
1	0 0 1	1.56V
2	0 1 0	2.00V
3	0 1 1	2.45V
4	1 0 0	2.89V
5	1 0 1	3.34V
6	1 1 0	3.79V
7	1 1 1	4.23V

8-BIT TIMER COUNTER

The 8.192ms clock is counted down with an 8-bit counter. When the count reaches zero, the timer flip-flop (TIMF/F) is set and the content of the 8-bit timer latch is again loaded into the counter.

(1) Timer counter control register (EVC)

This register is used to control the clock input to the counter. Executing the TEFA instruction causes the lower 2 bits in register A to be transferred to EVC. Execution of the TAEF instruction causes that control register information to be moved back into register A.

The cotrol register operation is shown in Table 6.

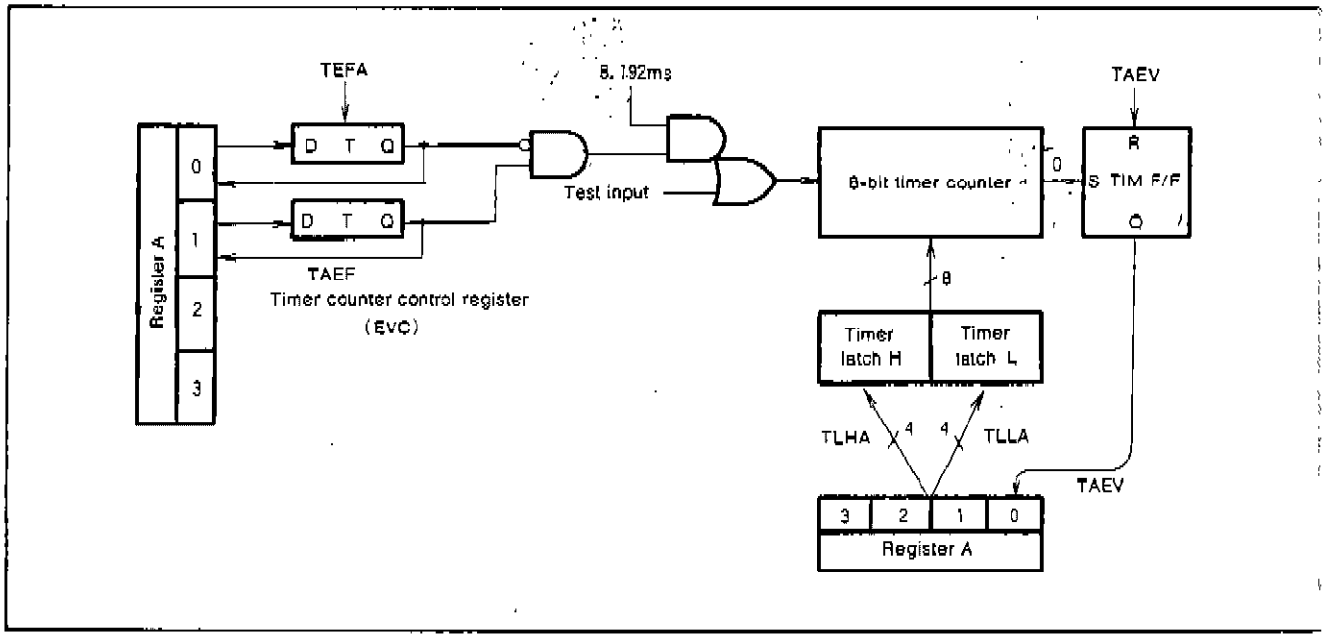

Fig.7 Structure of Timer Counter

(2) Timer latch

This is an 8-bit latch. The contents of register A are moved into the lower 4 bits of the timer latch by the execution of the TLLA instruction, and into the upper 4 bits of the event latch by the TLHA instruction.

(3) Timer counter

The clock input to the L_1 pin or the internal clock (selected by EVC) is counted down. When the count reaches zero, the timer flip-flop (TIMF/F) is set, and the contents of the timer latch are raloaded into the timer counter.

(4) Timer flip-flop (TIMF/F)

When the contents of the timer counter reaches zero, the timer flip flop is set, and then reset, after the flip flop contents are moved into the A register by the TAEV instruction.

Table 6. Event Counter Control Register (EVC)

Control Register				Description
3	2	1	0	
—	—	0	0	Stop the counter
		0	1	
—	0	1	0	Select the 8.192ms internal clock
—	0	1	1	Not select internal clock, and the contents of counter are indefinite
—	1	1	0	Select the 8.192ms internal clock after resetting the counter
—	1	1	1	Not select internal clock, and the contents of counter are indefinite

CRT DISPLAY

The M50436-XXXSP is capable of screen display and can directly control the screen display by executing the appropriate instructions.

(1) Number of display characters

These are two display blocks, each block being 1 line by 16 characters.

As shown in Table 7, the 4-th bit of the CRT control (CC) register allows the 1 line by 16-character mode. Also shown in the table 7, the on/off of the display of each block can be controlled separately by the CC register.

Data is transfered from register A to register CC by executing the TCCA instruction.

Table 7. Control Mode of Register CC

	Bit	Data	Functions
Register CC	0	0	All display OFF
		1	All display ON
	1	0	Block 1 display OFF
		1	Block 1 display ON
	2	0	Block 2 display OFF
		1	Block 2 display ON
	3	0	Block 3 display OFF
		1	Block 3 display ON

If reset, all the bits of register CC are set to "0".

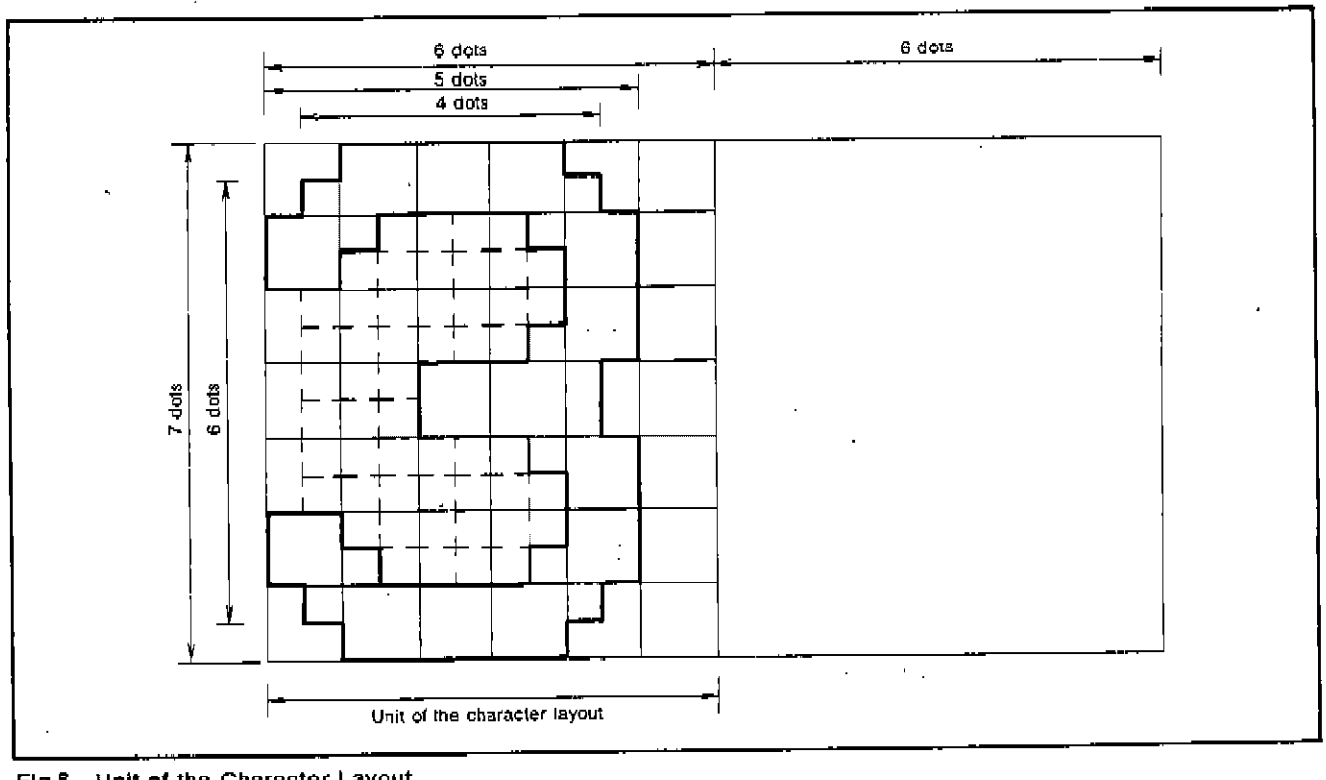

Fig.8 Unit of the Character Layout

(2) Character layout

A displayed character consists of a matrix of 6×7 dots. If a one-dot space is allowed between adjacent characters (in the horizontal direction), one character will be displayed from 5×7 dots.

Also, within the 5×7 dot matrix, is rounding feature of 4×6 dots which can be shifted by half a dot. This is to enable the display of a smooth chracter pattern. Refer to Figure 8.

For kanji character display, a 12×14 dot display can be enabled by specifying block 1 and block 2 of vertically adjacent positions.

Refer to Figure 9.

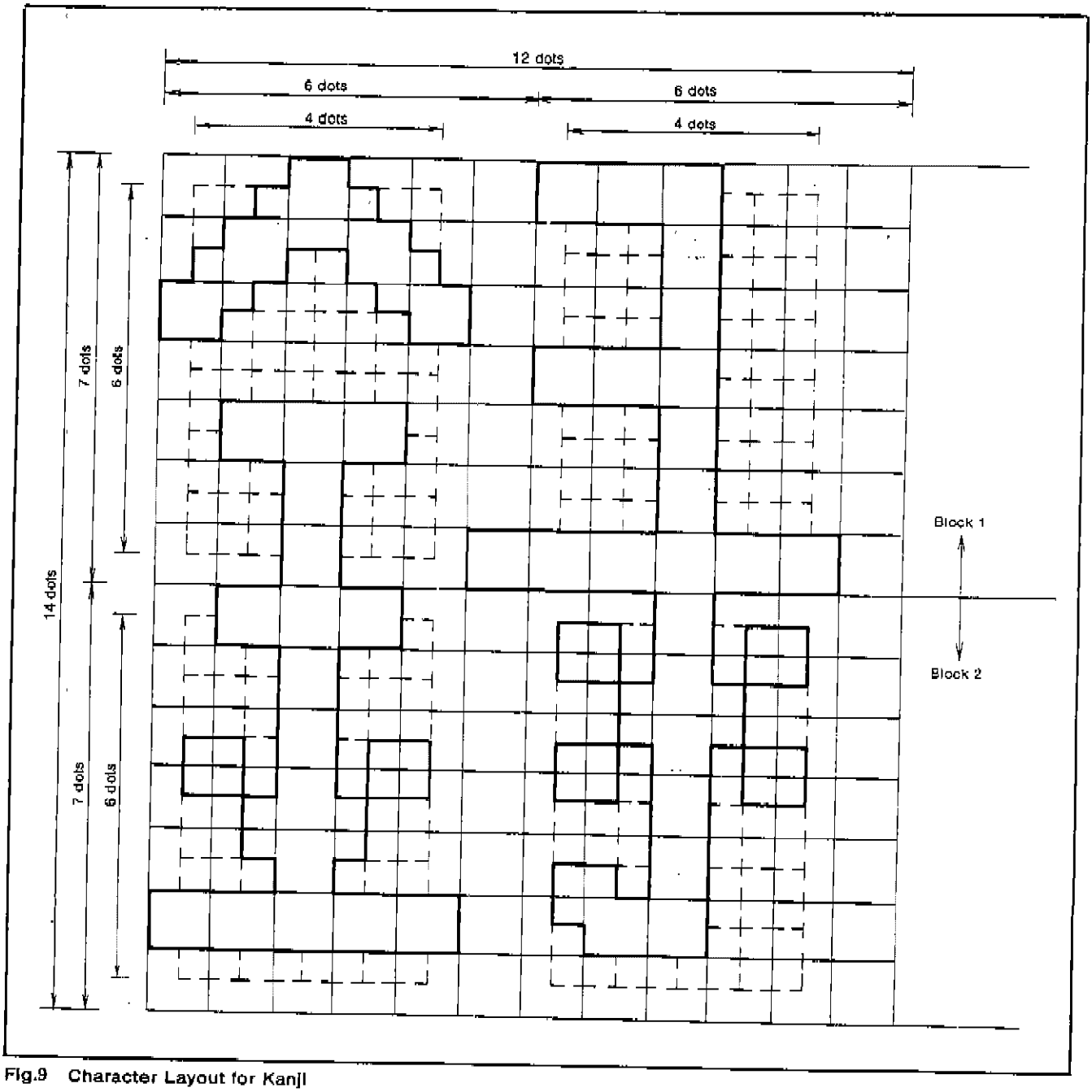

Fig.9 Character Layout for Kanji

(3) Display characters

Sixty four sets of character patterns, including those with the rounding feature, can be given in the character ROM. Table 8 shows the displayed characters.

Table 8. Display Characters

Cr_3~Cr_0 \ Cr_5~Cr_4	0	1	2	3
0	A	R	0	メ
1	B	S	1	イ
2	C	T	2	ン
3	D	U	3	サ
4	E	V	4	ヲ
5	F	W	5	ス
6	G	X	6	テ
7	H	Y	7	レ
8	I	Z	8	オ
9	J	.	9	フ
A	K	[	:	タ
B	L	]	▮▮	マ
C	M	/	▮▬	ヨ
D	N	ヒ	▬▪	ヤ
E	P	デ	ー	グ
F	Q	オ	Blank	Blank

One character portion of the display memory is composed of a 2-bit color specification and a 6-bit character code which corresponds to each character shown in Table 8. Figure 10 shows the structure of display memory.

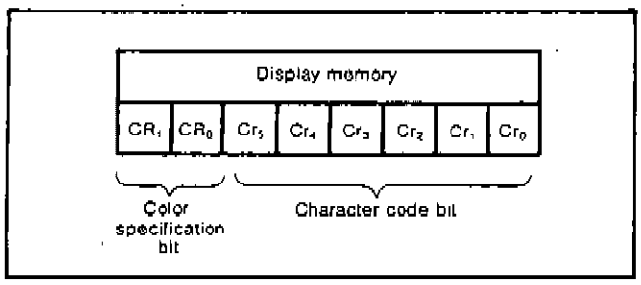

Fig.10 Structure of Display Memory

(4) Color designation

The display colors are determined by selecting color registers 0 to 3 which are pointed by the two bits of the display memory. Since there are only four kinds of color registers that can be specified by two bits, only four colors can be specified at a time. The R, G, B output enables seven different colors. The color registers are set from register A data by the TCnA (n=0~3) instruction. This is shown in Figure 11.

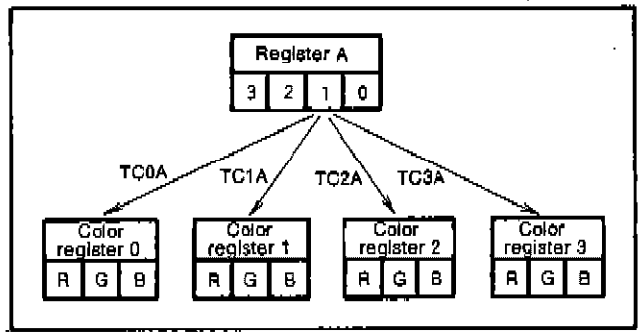

Fig.11 Setting of Color register

A color designation bit (R, G, and B) outputs nothing if it is "0" but becomes an active output if it is "1". If all bits for the R, G, and B are set to "0", nothing is output even through the character code is specified. Table 9 shows the correspondence between the color-designated bit data and actual output colors.

The OUT pin is the logical sum output of the R, G, and B pins and can be used as a background colors cut off when characters are output.

A set of three pins, R, G, and B, and the output polarity of the OUT pin can be specified when the ROM code is determined.

Table 9. Color Specification Bits and Output Colors

Color specification bit			Color
R	G	B	
0	0	0	—
0	0	1	Blue
0	1	0	Green
0	1	1	Cyan
1	0	0	Red
1	0	1	Magenta
1	1	0	Yellow
1	1	1	White

(5) Display location

Block 1 can select 64 levels of display location both in horizontal and vertical directions. The horizontal direction of block 2 and block 3 are is the same location as block1, and only the vertical location of block 2 and block 3 can be selected from the 64 levels of the display location, starting from the end location of block 1, and block 2, respectively.

Figure 12 shows the display locations of blocks 1, 2, and 3 on the screen.

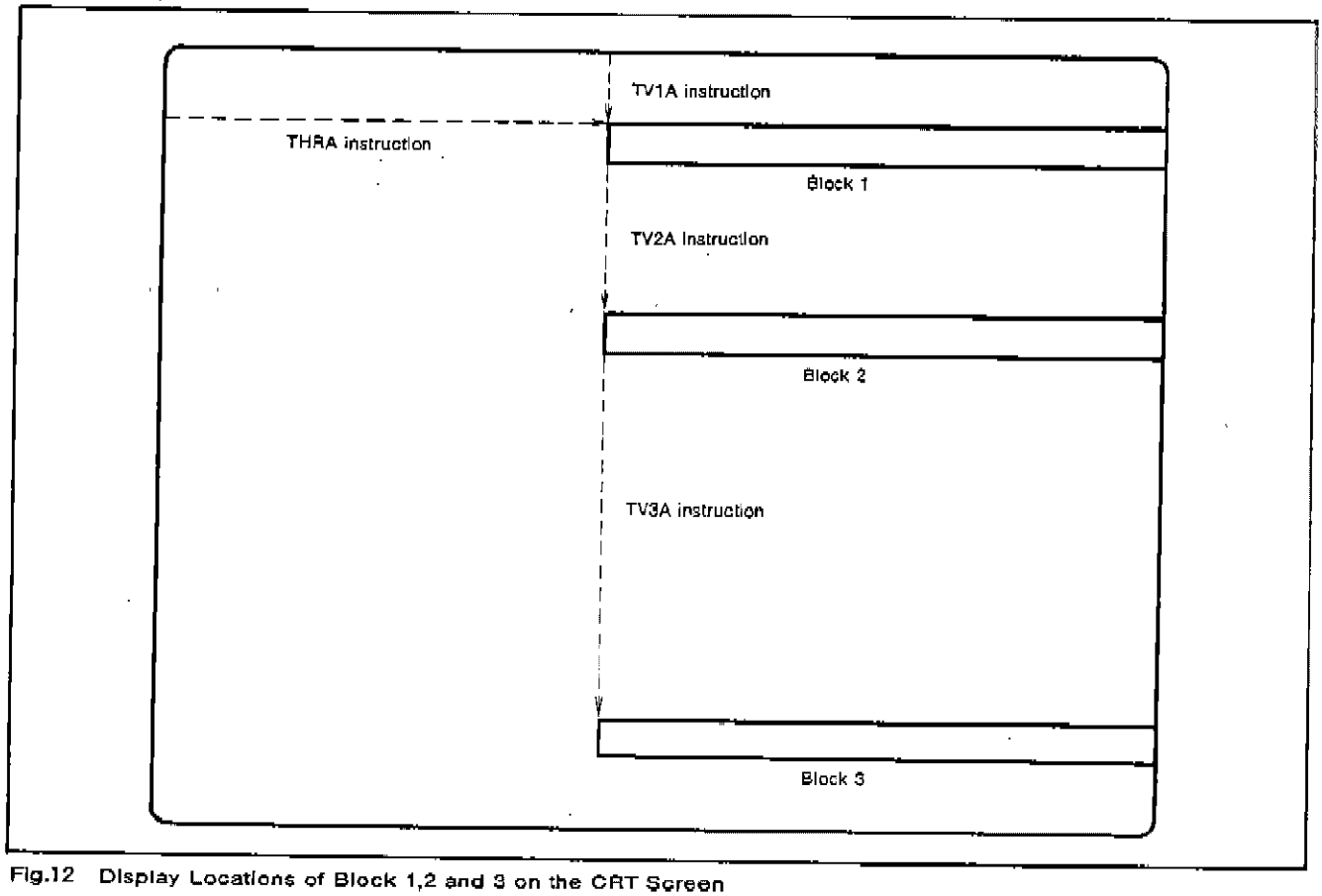

Fig.12 Display Locations of Block 1,2 and 3 on the CRT Screen

The size of one level is 4Tc (Tc: Display oscillation frequency) in the horizontal direction, and 4 scan lines in the vetical direction.

(6) Display character size

Four sets of display character sizes can be selected. The display character size can be set when the upper 2 bits of register C are transfered (as character size data) to the character size register by executing the TVnA instruction. The relationship between the upper 2 bits of register C and the character size display are shown in Table 10.

Table 10. Upper 2 Bits of Register C and Display Character Size

Register C		Display character size	Horizontal direction (display oscillation frequency cycle)	Vertical direction (number of screen lines)
C_3	C_2			
0	0	Small	2Tc	2
0	1	Medium	4Tc	4
1	0	Large	6Tc	6
1	1	Special large	8Tc	8

Size of 1 dot

(7) Display oscillating circuit

The clock signals for display can be obtained by connecting a resistor and a capacitor, or a coil and capacitors between the I/O pins for the oscillating circuit (OSC1 and OSC2). (Figure 13 shows an example of the circuit).

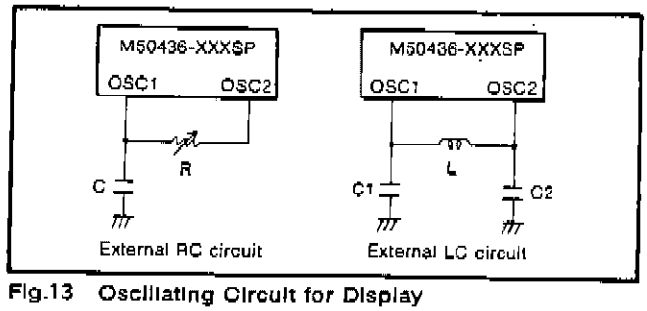

Fig.13 Oscillating Circuit for Display

OSCILLATING CIRCUIT

Since the CMOS inverter and feed-back pull-up resistor are built into the IC, the reference signal can be obtained by connecting a ceramic resonator between the oscillating circuit input pin and the output pin (OSC IN and OSC OUT). (Figure 14 shows an example of the circuit).

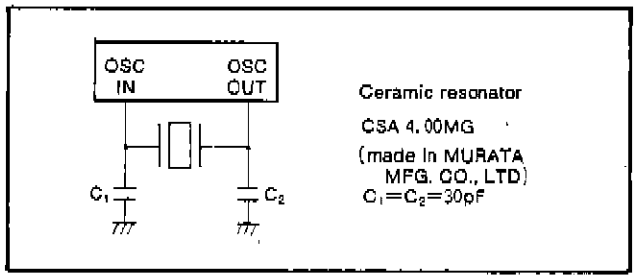

Fig.14 External Ceramic Resonator Circuit

RESET FUNCTION

By connecting a circuit to the $\overline{\text{RESET}}$ pin as shown in Figure 15, a reset can be implemented when the power is applied. The $\overline{\text{RESET}}$ pin contains a Schmitt circuit, therefore a reset is accomplished only when the voltage (V_C) of the $\overline{\text{RESET}}$ pin exceeds the threshold voltage V_{TH1}. Once reset, a second reset can not be made until V_C falls below the threshold Voltage V_{TH2}. Furthermore, to ensure that the reset is made when power is applied, the time interval T_C (which is the time between the power supply voltage V_{DD} reaching 4.5V and V_C) V_{TH1} must be less than 1 ms. Figure 16 describes this.

The reset function initializes the program counter to 0.

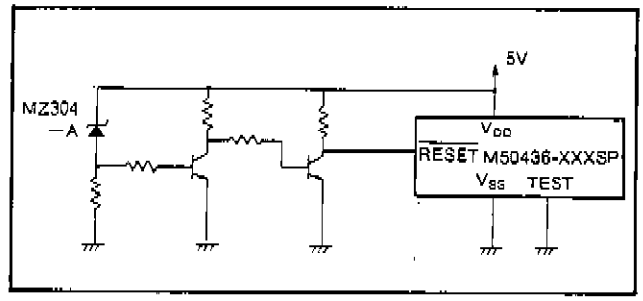

Fig.15 Power-on Clear Circuit

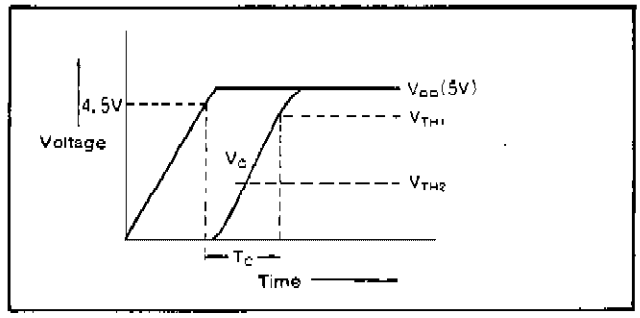

Fig.16 Relationship between the Voltage and the Auto Clear Function

M50436-XXXSP INSTRUCTION CODE TABLE

D_7~D_4 / Hexadecimal notation / D_3~D_0		0000	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111
		0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0000	0	NOP	RB 0	LY 0	JMP	LA 0	AD 0	TMA	TAM	EI	RC	LCMI 0	CAL	LX 0	LC 0	SNEA 0	SEA 0
0001	1	DATA	RB 1	LY 1	JMP	LA 1	AD 1		XAM	DI	SC	LCMI 1	CAL	LX 1	LC 1	SNEA 1	SEA 1
0010	2	MUT	RB 2	LY 2	JMP	LA 2	AD 2	TMAX	TAMX	TACM	CMC	LCMI 2	CAL	LX 2	LC 2	SNEA 2	SEA 2
0011	3	NMUT	RB 3	LY 3	JMP	LA 3	AD 3	TMAY	TAMY	ODA	ICD	LCMI 3	CAL	LX 3	LC 3	SNEA 3	SEA 3
0100	4	SZC SNZBO	SB 0	LY 4	JMP	LA 4	AD 4	INX	RAR	VDP 0	RT	LCMI 4	CAL	LX 4	LC 4	SNEA 4	SEA 4
0101	5	SNZC SZBO	SB 1	LY 5	JMP	LA 5	AD 5	INY	RAL	VDP 1	RTI	LCMI 5	CAL	LX 5	LC 5	SNEA 5	SEA 5
0110	6	SEV	SB 2	LY 6	JMP	LA 6	AD 6	PHXY	TAI	VDP 2	RTS	LCMI 6	CAL	LX 6	LC 6	SNEA 6	SEA 6
0111	7	SNEV	SB 3	LY 7	JMP	LA 7	AD 7	PLXY	CMA	*The second word	SKIP	LCMI 7	CAL	LX 7	LC 7	SNEA 7	SEA 7
1000	8	TC0A	SZB 0	LY 8	JMP	LA 8	AD 8	TYA	TAY	TCCA	TCMAI	LCMI 8	CAL	LX 8	LC 8	SNEA 8	SEA 8
1001	9	TC1A	SZB 1	LY 9	JMP	LA 9	AD 9	TXA	TAX	TAC	TCA	LCMI 9	CAL	LX 9	LC 9	SNEA 9	SEA 9
1010	A	TC2A	SZB 2	LY A	JMP	LA A	AD A	TBA	TAB	TAZ	TZA	LCMI A	CAL	LX A	LC A	SNEA A	SEA A
1011	B	TC3A	SZB 3	LY B	JMP	LA B	AD B	TIFA	TAIF	TAZZ	TZZA	LCMI B	CAL	LX B	LC B	SNEA B	SEA B
1100	C	OJA	SNZB 0	LY C	JMP	LA C	AD C	OFA	IAE	AM	TV1A	LCMI C	CAL	LX C	LC C	SNEA C	SEA C
1101	D	OKA	SNZB 1	LY D	JMP	LA D	AD D	OGA	IAG	AMC	TV2A	LCMI D	CAL	LX D	LC D	SNEA D	SEA D
1110	E	OLA	SNZB 2	LY E	JMP	LA E	AD E	OHA	TAHS	SNEAM	TV3A	LCMI E	CAL	LX E	LC E	SNEA E	SEA E
1111	F	—	SNZB 3	LY F	JMP	LA F	AD F	OMA	IAL	SEAM	THRA	LCMI F	CAL	LX F	LC F	SNEA F	SEA F

2-word instruction

3-word instruction

D_7~D_4 / Hexadecimal notation / D_3~D_0		0000	0001	0010	0011	0100	0101
		0	1	2	3	4	5
0000	0	SU 0	OR 0	AND 0	TAR	TRA	—
0001	1	SU 1	OR 1	AND 1	—	—	—
0010	2	SU 2	OR 2	AND 2	SUM	—	—
0011	3	SU 3	OR 3	AND 3	SUMB	—	—
0100	4	SU 4	OR 4	AND 4	ORM	TLLA	—
0101	5	SU 5	OR 5	AND 5	ANDM	TLHA	—
0110	6	SU 6	OR 6	AND 6	TAEF	TEFA	—
0111	7	SU 7	OR 7	AND 7	TAEV	—	—
1000	8	SU 8	OR 8	AND 8	TAC0	—	—
1001	9	SU 9	OR 9	AND 9	TAC1	—	—
1010	A	SU A	OR A	AND A	TAC2	—	—
1011	B	SU B	OR B	AND B	TAC3	—	—
1100	C	SU C	OR C	AND C	TACML	—	—
1101	D	SU D	OR D	AND D	TACMH	—	—
1110	E	SU E	OR E	AND E	—	—	—
1111	F	SU F	OR F	AND F	—	—	—

D_7~D_4 / Hexadecimal notation / D_3~D_0		0000	0001	0010	0011	1000	1001	1010	1011
		0	1	2	3	8	9	A	B
0000	0	[illegible]	[illegible]	—	—	[illegible]	[illegible]	—	—
0001	1	[illegible]	[illegible]	—	—	[illegible]	[illegible]	—	—
0010	2	[illegible]	[illegible]	—	—	[illegible]	[illegible]	—	—
0011	3	[illegible]	[illegible]	—	—	[illegible]	[illegible]	—	—
0100	4	[illegible]	[illegible]	—	—	[illegible]	[illegible]	—	—
0101	5	[illegible]	[illegible]	—	—	[illegible]	[illegible]	—	—
0110	6	[illegible]	[illegible]	—	—	[illegible]	[illegible]	—	—
0111	7	[illegible]	[illegible]	—	—	[illegible]	[illegible]	—	—
1000	8	[illegible]	—	—	—	[illegible]	—	—	—
1001	9	[illegible]	—	—	—	[illegible]	—	—	—
1010	A	[illegible]	—	—	—	[illegible]	—	—	—
1011	B	[illegible]	—	—	—	[illegible]	—	—	—
1100	C	[illegible]	—	—	—	[illegible]	—	—	—
1101	D	[illegible]	—	—	—	[illegible]	—	—	—
1110	E	[illegible]	—	—	—	[illegible]	—	—	—
1111	F	[illegible]	—	—	—	[illegible]	—	—	—

SYMBOLS

The following notation are used for the following descriptions.

Symbol	Contents	Symbol	Contents
A	Register A(4 bits)	INTE	Interrupt enable flag
B	Register B(4 bits)	INTF	Interrupt request flip-flop
C	Register C(4 bits)	ED/ID	ED/ID flag
CC	Register CC(4 bits)	ELL	The lower 4 bits of event latch
CO_0	Color register(3 bits)	ELH	The upper 4 bits of event latch
CO_1	Color register(3 bits)	EVC	Event counter control register
CO_2	Color register(3 bits)	EVF	Event flip-flop
CO_3	Color register(3 bits)	MUTF	Mute flag
I	Register I(3 bits)	∧	AND
R	Register R(3 bits)	∨	OR
D-AL	D-A latch(14 bits)	M(DP)	RAM address which is specified by data pointer
X	Register X(4 bits)		
Y	Register Y(4 bits)	←	Direction in which data is transferred
Z	Register Z(4 bits)	( )	The contents of register, memory, etc.
ZZ	Register ZZ(2 bits)	x	1-bit binary variable
DP	Data pointer (8 bits)	n_1n_0	2-bit binary variable
	(consisting of the registers X and Y)	j_1j_0	2-bit binary variable
PC	Program counter(13 bits)	$x_3x_2x_1x_0$	4-bit binary variable
SK	Stack register(13 bits)	$y_3y_2y_1y_0$	4-bit binary variable
SKX	Stack register X(4 bits)	$n_3n_2n_1n_0$	4-bit binary variable
SKY	Stack register Y(4 bits)		
CY	Carry flag(1 bit)	a	Label to show the address of
D	Port D(1 bit)		a_{11} a_{10} a_9 a_8 a_7 a_6 a_5 a_4 a_3 a_2 a_1 a_0
E	Port E(4 bits)	—	Negation or condition of the flag is not change after the instruction is executed
F	Port F(4 bits)		
G	Port G(4 bits)		
H	Port H(4 bits)		
J	Port J(4 bits)		
K	Port K(4 bits)		
L	Port L(4 bits)		
M	Port M(3 bits)		

Note 1 : The M50436-XXXSP performs a skip by ignoring the next instruction, and by not executing the instruction at the address pointed to by the contents of the program counter + 2.
Therefore, the cycle number does not change, regardless of whether a skip is generated or not.

MACHINE INSTRUCTIONS

Type of instructions \ Parameter	Mnemonic	Instruction code								Hexadecimal notation		Number of words	Number of cycles	Functions
		D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0					
Register to register transfers	TAB	0	1	1	1	1	0	1	0	7	A	1	1	(A)←(B)
	TBA	0	1	1	0	1	0	1	0	6	A	1	1	(B)←(A)
	TAC	1	0	0	0	1	0	0	1	8	9	1	1	(A)←(C)
	TCA	1	0	0	1	1	0	0	1	9	9	1	1	(C)←(A)
	TAX	0	1	1	1	1	0	0	1	7	9	1	1	(A)←(X)
	TXA	0	1	1	0	1	0	0	1	6	9	1	1	(X)←(A)
	TAY	0	1	1	1	1	0	0	0	7	8	1	1	(A)←(Y)
	TYA	0	1	1	0	1	0	0	0	6	8	1	1	(Y)←(A)
	TAR	1	0	0	0	0	1	1	1	8	7	2	2	(A_3, A_2, A_1, A_0)←(0, R_2, R_1, R_0)
		0	0	1	1	0	0	0	0	3	0			
	TRA	1	0	0	0	0	1	1	1	8	7	2	2	(R_2, R_1, R_0)←(A_2, A_1, A_0)
		0	1	0	0	0	0	0	0	4	0			
	TACM	1	0	0	0	0	0	1	0	8	2	2	2	(A_3, A_2, A_1, A_0)←(0, 0, A/D, COMP)
		0	0	0	0	0	0	0	0	0	0			
	TAHS	0	1	1	1	1	1	1	0	7	E	1	1	(A)←(HS)
	TAI	0	1	1	1	0	1	1	0	7	6	1	1	(A_3, A_2, A_1, A_0)←(0, INT, BIT, GP)
	TAIF	0	1	1	1	1	0	1	1	7	B	1	1	(A_3, A_2, A_1, A_0)←(INTF, EDF, ED_1, ED_0)
	TIFA	0	1	1	0	1	0	1	1	6	B	1	1	(ED_1, ED_0)←(A_1, A_0)
Register control operations	LX x	1	1	0	0	x_3	x_2	x_1	x_0	C	x	1	1	(X)←x, where x= 0 ~15
	LY y	0	0	1	0	y_3	y_2	y_1	y_0	2	y	1	1	(Y)←y, where y= 0 ~15
	LC n	1	1	0	1	n_3	n_2	n_1	n_0	D	n	1	1	(C)←n, where n= 0 ~15
	INY	0	1	1	0	0	1	0	1	6	5	1	1	(Y)←(Y)+1
	INX	0	1	1	0	0	1	0	0	6	4	1	1	(X)←(X)+1
	PHXY	0	1	1	0	0	1	1	0	6	6	1	1	(SKY)←(X), (SKY)←(Y)
	PLXY	0	1	1	0	0	1	1	1	6	7	1	1	(X)←(SKX), (Y)←(SKY)

Skip conditions	Carry flag	Detailed description
—	—	The contents of register B are transferred to register A.
—	—	The contents of register A are transferred to register B.
—	—	The contents of register C are transferred to register A.
—	—	The contents of register A are transferred to register C.
—	—	The contents of register X are transferred to register A.
—	—	The contents of register A are transferred to register X.
—	—	The contents of register Y are transferred to register A.
—	—	The contents of register A are transferred to register Y.
—	—	The contents of register R are transferred to register A.
—	—	The contents of register A are transferred to register R.
—	—	The result of comparison (COMP) and the value of A-D pin (A/D) are transferred to register A.
—	—	The contents of HS counter are transferred to register A.
—	—	The result of judging interrupt are transferred to register A.
—	—	The contents of IF flag are transferred to register A.
—	—	The contents of register A are transferred to IF flag.
continuous description	—	The immediate field value x is loaded into register X. If a continuous description of LX instructions are written and are being executed, only the first LX instruction is executed, the following LX instructions are all skipped.
continuous description	—	The immediate field value y is loaded into register Y. If a continuous description of LY instructions are written and are being executed, only the first LY instruction is executed, the following LY instructions are all skipped.
continuous description	—	The immediate field value n is loaded into register C. If a continuous description of LC instructions are written and are being executed, only the first LC instruction is executed, the following LC instructions are all skipped.
—	—	The contents of register X are incremented by 1.
—	—	The contents of register Y are incremented by 1. As a result, if the contents of register Y are "0", the next instruction is skipped.
—	—	The contents of data pointer (register X and register Y) are transferred to stack X and stack Y.
—	—	The contents of stack X and stack Y are transferred to data pointer.

Type of instructions	Mnemonic	Instruction code D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0	Hexadecimal notation		Number of words	Number of cycles	Functions
RAM addresses	TAM	0	1	1	1	0	0	0	0	7	0	1	1	(A)←(M(DP))
	TAMX	0	1	1	1	0	0	1	0	7	2	1	1	(B)←(M(DP)), (X)←(X)+1
	TAMY	0	1	1	1	0	0	1	1	7	3	1	1	(A)←(M(DP)), (Y)←(Y)+1
	TMA	0	1	1	0	0	0	0	0	6	0	1	1	(M(DP))←(A)
	TMAX	0	1	1	0	0	0	1	0	6	2	1	1	(M(DP))←(A), (X)←(X)+1
	TXAY	0	1	1	0	0	0	1	1	6	3	1	1	(M(DP))←(A), (Y)←(Y)+1
	XAM	0	1	1	1	0	0	0	1	7	1	1	1	(A)↔(M(DP))
Arithmetic operations	AM	1	0	0	0	1	1	0	0	8	C	1	1	(A)←(A)+(M(DP)), (CY)←carry
	AMC	1	0	0	0	1	1	0	1	8	D	1	1	(A)←(A)+(M(DP))+(CY), (CY)←carry
	SUM	1 0	0 0	0 1	0 1	0 0	1 0	1 1	1 0	8 3	7 2	2	2	(A)←(A)−(M(DP)), (C)←$\overline{\text{borrow}}$
	SUMB	1 0	0 0	0 1	0 1	0 0	1 0	1 1	1 1	8 3	7 3	2	2	(A)←(A)−(M(DP))−(borrow) (CY)←$\overline{\text{borrow}}$
	ANDM	1 0	0 0	0 1	0 1	0 0	1 1	1 0	1 1	8 3	7 5	2	2	(A)←(A)∧(M(DP))
	ORM	1 0	0 0	0 1	0 1	0 0	1 1	1 0	1 0	8 3	7 4	2	2	(A)←(A)∨(M(DP))
	LA n	0	1	0	0	n_3	n_2	n_1	n_0	4	n	1	1	(A)←n, where n=0~15
	AD n	0	1	0	1	n_3	n_2	n_1	n_0	5	n	1	1	(A)←(A)+n, where n=0~15, (CY)←carry
	AND n	1 0	0 0	0 1	0 0	0 n_3	1 n_2	1 n_1	1 n_0	8 2	7 n	2	2	(A)←(A)∧n, where n=0~15
	OR n	1 0	0 0	0 0	0 1	0 n_3	1 n_2	1 n_1	1 n_0	8 1	7 n	2	2	(A)←(A)∨n, where n=0~15
	SU n	1 0	0 0	0 0	0 0	0 n_3	1 n_2	1 n_1	1 n_0	8 0	7 n	2	2	(A)←(A)−n, where n=0~15, (CY)←$\overline{\text{borrow}}$
	CMA	0	1	1	1	0	1	1	1	7	7	1	1	(A)←($\overline{\text{A}}$)
	RAL	0	1	1	1	0	1	0	1	7	5	1	1	CY ← $A_3A_2A_1A_0$ ←
	RAR	0	1	1	1	0	1	0	0	7	4	1	1	→ CY → $A_3A_2A_1A_0$
	SC	1	0	0	1	0	0	0	1	9	1	1	1	(CY)←1
	RC	1	0	0	1	0	0	0	0	9	0	1	1	(CY)←0

Skip conditions	Carry flag	Detailed description
—	—	The contents of M(DP) are transferred to register A.
—	—	The contents of M(DP) are transferred to register A, and the contents of register X are incremented by 1.
—	—	The contents of register A are transferred to M(DP). And if the contents of register Y are incremetented by 1 and the result is "0", then the next instruction is skipped.
—	—	The contents of register A are transferred to M(DP).
—	—	The contents of register A are transferred to M(DP), and the contents of register X are incremented by 1.
—	—	The contents of register A are transferred to M(DP). And if the contents of register Y are incremetented by 1 and the result is "0", then the next instruction is skipped.
—	—	The contents of M(DP) are exchanged with register A.
—	—	The contents of M(DP) are added to register A, and the result is stored into register A and in the carry flag CY.
—	—	The contents of M(DP) and carry flag CY are added to register A and the result is stored into register A and into carry flag CY.
—	—	The contents of M(DP) are subtract from register A, and the result is stored into register A and in the carry flag CY.
—	—	The contents of M(DP) and borrow are subtract from register A, and the result is stored into register A and in the carry flag CY.
—	—	A logical AND is performed between the contents of register A and the contents of M(DP), and the result is stored into register A.
—	—	A logical OR is performed between the contents of register A and the contents of M(DP), and the result is stored into register A.
—	—	The immediate field value n is loaded into register A. If a continuous description of LA instructions are written and being executed, only the first LA instruction is executed, the following LA instructions are all skipped.
—	—	The immediate field value n is added to register A, and the result is stored into register A and in the carry flag CY.
—	—	A logical AND is performed between the contents of register A and the immediate field value n, and the result is stored into register A.
—	—	A logical OR is performed between the contents of register A and the immediate field value n, and the result is stored into register A.
—	—	The immediate field value n is subtract from register A, and the result is stored into register A and in the carry flag CY.
—	—	The one's complement of register A's contents are stored into register A.
—	A_3	The register A, including carry flag CY, is rotated 1 bit to the left.
—	A_0	The register A, including carry flag CY, is rotated 1 bit to the right.
—	1	The carry flag CY is set (1).
—	0	The carry flag CY is reset (0).

Type of instructions / Parameter	Mnemonic	Instruction code D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0	Hexadecimal notation		Number of words	Number of cycles	Functions
Arithmetic operations	CMC	1	0	0	1	0	0	1	0	9	2	1	1	$(CY) \leftarrow (\overline{CY})$
	SZC	1	0	0	0	0	1	0	0	8	4	1	1	(CY)=0 ?
	SNZBO													
	SNZC	0	0	0	0	0	1	0	1	0	5	1	1	(CY)≠0 ?
	SZBO													
Bit operations	SB j	0	0	0	1	0	1	j_1	j_0	1	4+j	1	1	(MJ(DP))←1, where j=0∼3
	RB j	0	0	0	1	0	0	j_1	j_0	1	J	1	1	(MJ(DP))←0, where j=0∼3
	SZB j	0	0	0	1	1	0	j_1	j_0	1	8+j	1	1	(MJ(DP))←0 ?, where j=0∼3
	SNZB j	0	0	0	1	1	1	j_1	j_0	1	C+j	1	1	(MJ(DP))←1 ?, where j=0∼3
Comparisons	SEA n	1	1	1	1	n_3	n_2	n_1	n_0	F	n	1	1	(A)=n ?, where n=0∼15
	SNEA n	1	1	1	0	n_3	n_2	n_1	n_0	E	n	1	1	(A)≠n ?, where n=0∼15
	SEAM	1	0	0	0	1	1	1	1	8	F	1	1	(A)=(M(DP)) ?
	SNEAM	1	0	0	0	1	1	1	0	8	E	1	1	(A)≠(M(DP)) ?
Interrupt operations	DI	1	0	0	0	0	0	0	1	8	1	1	1	(EI/DI)←0
	EI	1	0	0	0	0	0	0	0	8	0	1	1	(EI/DI)←1
Event counter control operations	TAEV	1	0	0	0	0	1	1	1	8	7	2	2	$(A_3, A_2, A_1, A_0) \leftarrow (0, 0, 0, EVF)$
		0	0	1	1	0	1	1	1	3	7			
	TAEF	1	0	0	0	0	1	1	1	8	7	2	2	$(A_3, A_2, A_1, A_0) \leftarrow (0, 0, EVC_1, EVC_0)$
		0	0	1	1	0	1	1	0	3	6			
	TEFA	1	0	0	0	0	1	1	1	8	7	2	2	$(EVCF, EVC_1, EVC_0) \leftarrow (A_2, A_1, A_0)$
		0	1	0	0	0	1	1	0	4	6			
	TLLA	1	0	0	0	0	1	1	1	8	7	2	2	(ELL)←(A)
		0	1	0	0	0	1	0	0	4	4			
	TLHA	1	0	0	0	0	1	1	1	8	7	2	2	(ELH)←(A)
		0	1	0	0	0	1	0	1	4	5			
	SEV	0	0	0	0	0	1	1	0	0	6	1	1	(EVF)=1 ?
	SNEV	0	0	0	0	0	1	1	1	0	7	1	1	(EVF)=0 ?

Skip conditions	Carry flag	Detailed description
$(\overline{CY})$	—	The one's complement of carry flag CY's contents are stored into carry flag CY.
(CY) = 0	—	If the contents of carry flag CY are "0", the next instruction is skipped.
(CY) ≠ 0	—	If the contents of carry flag CY are "1", the next instruction is skipped.
—	—	The j-th bit of the contents of M(DP), which is the bit specified by the immediate field value j, is set to 1.
—	—	The j-th bit of the contents of M(DP), which is the bit specified by the immediate field value j, is reset to 0.
—	—	If the j-th bit of the contents of M(DP), which is the bit specified by the immediate field value j, is "0", the next instruction is skipped.
—	—	If the j-th bit of the contents of M(DP), which is the bit specified by the immediate field value j, is "1", the next instruction is skipped.
(A) = n	—	If the contents of register A are equal to the immediate field value n, the next instruction is skipped.
(A) ≠ n	—	If the contents of register A are not equal to the immediate field value n, the next instruction is skipped.
(A) = (M(DP))	—	If the contents of register A are equal to the contents of M(DP), the next instruction is skipped.
(A) ≠ (M(DP))	—	If the contents of register A are not equal to the contents of M(DP), the next instruction is skipped.
—	—	The EI/DI flag is reset (0) to change the state in which an interrupt is disabled.
—	—	The EI/DI flag is set (1) to change the state in which an interrupt is enabled.
—	—	The contents of event flip-flop are transferred to register A and the flip-flop is reset (0).
—	—	The contents of event control register are transferred to the lower 2 bits of register A.
—	—	The lower 3 bits of register A are transferred to the event control register.
—	—	The contents of register A are transferred to the lower 4 bits of event latch.
—	—	The contents of register A are transferred to the upper 4 bits of event latch.
(EVF) = 1	—	If the contents of event flip-flop are "1", the next instruction is skipped and the flip-flop is reset (0).
(EVF) = 0	—	If the contents of event flip-flop are "0", the next instruction is skipped and the flip-flop is reset (0).

Type of instructions / Parameter	Mnemonic	Instruction code D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0	Hexadecimal notation		Number of words	Number of cycles	Functions
Input/Output operations	ODA	1	0	0	0	0	0	1	1	8	3	1	1	(D)←(A_3)
	OFA	0	1	1	0	1	1	0	0	6	C	1	1	(F)—(A)
	OGA	0	1	1	0	1	1	0	1	6	D	1	1	(G)←(A)
	OHA	0	1	1	0	1	1	1	0	6	E	1	1	(H)←(A)
	OJA	0	0	0	0	1	1	0	0	0	C	1	1	(J)←(A)
	OKA	0	0	0	0	1	1	0	1	0	D	1	1	(K)←(A)
	OLA	0	0	0	0	1	1	1	0	0	E	1	1	(L)←(A)
	OMA	0	1	1	0	1	1	1	1	6	F	1	1	(M)—(A_2, A_1, A_0)
	IAE	0	1	1	1	1	1	0	0	7	C	1	1	(A)←(E)
	IAG	0	1	1	1	1	1	0	1	7	D	1	1	(A)←(G)
	IAL	0	1	1	1	1	1	1	1	7	F	1	1	(A)←(L)
	ICD	1	0	0	1	0	0	1	1	9	3	1	1	(CY)←(D)
D-A convertor control operations	VDP n	1	0	0	0	0	1	n_1	n_0	8	4 + n	1	1	(VDPn)←(B_2, B_1, B_0, A_3, A_2, A_1, A_0) where n = 0 ~ 2
	DATA	0	0	0	0	0	0	0	1	0	1	1	1	(D-A L)←(M(0, 3)~M(0, 0))
	MUT	0	0	0	0	0	0	1	0	0	2	1	1	(MUTF)—1
	NMUT	0	0	0	0	0	0	1	1	0	3	1	1	(MUTF)←0
Jump and subroutine call operations	JMP a	0 a_7	0 a_6	1 a_5	1 a_4	a_3 a_3	a_2 a_2	a_1 a_1	a_0 a_0	3 a	a a	2	2	(PC)←a_{11}~a_0
	CAL a	1 a_7	0 a_6	1 a_5	1 a_4	a_3 a_3	a_2 a_2	a_1 a_1	a_0 a_0	B a	a a	2	2	(SK)←(PC_{next}) (PC)—a_{11}~a_0
	JMPL aa	0 0 a_7	1 0 a_6	1 0 a_5	0 a_4 a_4	0 a_3 a_3	0 a_2 a_2	0 a_1 a_1	1 a_0 a_0	6 a a	1 a a	3	3	(PC)—a_{12}~a_0
	CALL aa	0 1 a_7	1 0 a_6	1 0 a_5	0 a_4 a_4	0 a_3 a_3	0 a_2 a_2	0 a_1 a_1	1 a_0 a_0	6 8 + a a	1 a a	3	3	(SK)—(PC_{next}) (PC)—a_{12}~a_0
	RT	1	0	0	1	0	1	0	0	9	4	1	1	(PC)←(SK)
	RTI	1	0	0	1	7	1	0	1	9	5	1	1	(PC)—(SK), (SKIP)—(SKS)
	RTS a	1	0	0	1	0	1	1	0	9	6	1	1	(PC)←(SK), (SKIP)←1
	SKIP	1	0	0	1	0	1	1	1	9	7	1	1	(SKIP)—1
	NOP	1	0	0	1	0	1	1	1	0	0	1	1	(PC)—(PC)+1

Skip conditions	Carry flag	Detailed description
—	—	The most significant bit of register A is output to the port D.
—	—	The contents of register A are output to the port F.
—	—	The contents of register A are output to the port G.
—	—	The contents of register A are output to the port H.
—	—	The contents of register A are output to the port J.
—	—	The contents of register A are output to the port K.
—	—	The contents of register A are output to the port L.
—	—	The lower 3 bits of register A are output to the port M.
—	—	The input from the port E are transferred to register A.
—	—	The input from the port G are transferred to register A.
—	—	The input from the port L are transferred to register A.
—	—	The input from the port D are transferred to carry flag CY.
—	—	The 7-bit D-A date (the lower 3 bits of register B and register A) are transferred to VDP latch.
—	—	The 15-bit D-A data (M(0,0), M(0,1), M(0,2), and except for the upper 1 bit of M(03)) are transffered to the D-A latch.
—	—	The MUTE flag is set (1). The VDP port becomes "L" level.
—	—	The MUTE flag is set (0). Mute state is canceled.
—	—	Jump to an address which is specified by a_{11}~a_0 unconditionally.
—	—	The return address is stored into stack register, and jump to an address which is specified by a_{11}~a_0 unconditionally.
—	—	Jump to an address which is specified by a_{12}~a_0 unconditionally.
—	—	The return address is stored into stack register, and jump to an address which is specified by a_{12}~a_0 unconditionally.
—	—	Control is then returned from the subroutine to the routine which is called the subroutine.
(SKS) = 1	—	Control is then returned from the interrupt handing routine to the main routine.
unconditional skip	—	Control is then returned from the subroutine to the routine which is called the subroutine, and the next instruction is unconditionally skipped.
unconditional skip	—	The next instruction is skipped after the skip flag is set to "1".
—	—	No operation.

Parameter / Type of instructions	Mnemonic	Instruction code D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0	Hexadecimal notation		Number of words	Number of cycles	Functions
Display control operation	TCCA	1	0	0	0	1	0	0	0	8	8	1	1	(CC)←(A)
	TAZ	1	0	0	0	1	0	1	0	8	A	1	1	(A)←(Z)
	TZA	1	0	0	1	1	0	1	0	9	A	1	1	(Z)—(A)
	TAZZ	1	0	0	0	1	0	1	1	8	B	1	1	(A_3, A_2, A_1, A_0)←$(0, 0, ZZ_1, ZZ_0)$
	TZZA	1	0	0	1	1	0	1	1	9	B	1	1	(ZZ_1, ZZ_0)←(A_1, A_0)
	TCnA	0	0	0	0	1	0	0	0	0	8+n	1	1	(Cn_2, Cn_1, Cn_0)←(A_2, A_1, A_0) where $n=0\sim3$
	TAC n	1	0	0	0	0	1	1	1	8	7	2	2	(A_3, A_2, A_1, A_0)←$(0, Cn_2, Cn_1, Cn_0)$
		0	0	1	1	1	0	n_1	n_0	3	8+n			where $n=0\sim3$
	TVnA	1	0	0	1	1	1	m_1	m_0	9	C+m	1	1	$(SZ_1, SZ_0, Vn_5, Vn_4, Vn_3, Vn_2, Vn_1, Vn_0)$ ←$(C_3, C_2, C_1, C_0, A_3, A_2, A_1, A_0)$ $m=(n-1)$, where $n=0\sim2$
	THRA	1	0	0	1	1	1	1	1	9	F	1	1	$(H_3, H_4, H_3, H_2, H_1, H_0)$ ←$(C_1, C_0, A_3, A_2, A_1, A_0)$
	LCMI n	1	0	1	0	n_3	n_2	n_1	n_0	A	n	1	1	(CM(ZZ, Z))—(C), n, (Z)—(Z)+1
	TCMAI	1	0	0	1	1	0	0	0	9	8	1	1	(CM(ZZ,Z))—(C),(A), (Z)—(Z)+1
	TACML	1	0	0	0	0	1	1	1	8	7	2	2	(A)←$(CM(ZZ, Z))_{3\sim0}$
		0	0	1	1	1	1	0	0	3	C			
	TACMH	1	0	0	0	0	1	1	1	8	7	2	2	(A)—$(CM(ZZ, Z))_{7\sim4}$
		0	0	1	1	1	1	0	1	3	D			

Skip conditions	Carry flag	Detailed description
—	—	The contents of register A are transferred to register CC.
—	—	The contents of pointer register Z for CRTRAM are transferred to register A.
—	—	The contents of register A are transferred to the pointer register Z for CRTRAM.
—	—	The contents of selector register ZZ for CRTRAM are transferred to register A.
—	—	The contents of register A are transferred to the selector register ZZ for CRTRAM.
—	—	The contents of register A are transferred to the color register Cn.
—	—	The contents of color register Cn are transferred to register A.
—	—	The contents of register A and register C are transferred to the character size register SZn and the vertical location register Vn.
—	—	The contents of the lower 2 bits of register C and register A are transferred to register HR.
—	—	After the immediate field value n and the contents of register C are transferred to M(Z), the contents of register Z are incremented by 1.
—	—	The contents of registers A and C are transferred to M(Z), the contents of register Z are incremented by 1.
—	—	The lower 4 bits of CRTRAM are transferred to register A.
—	—	The upper 4 bits of CRTRAM are transferred to register A.

ABSOLUTE MAXIMUM RATINGS (T_a=25°C, except the limits of temperature)

Symbol	Parameter	Test conditions	Ratings	Unit
V_{DD}	Supply voltage		−0.3~6	V
V_I	Input voltage Ports E_0~E_3, G_0~G_3, L_0~L_3 HSYNC, VSYNC, $\overline{INT}$, $\overline{RESET}$	With respect to V_{SS}	−0.3~V_{DD}+0.3	V
V_O	Output voltage		−0.3~V_{DD}+0.3	V
V_O	Output voltage VDP_0~VDP_2, Ports M_0~M_3, H_0~H_3 K_0~K_3	With respect to V_{SS} Output transistors cut-off	−0.3~13	V
$\|I_{OH}\|$	Current in circuit		0~1(Note 1)	mA
I_{OL}	Current in circuit		0~2(Note 2)	mA
I_{OL}	Current in circuit Port H_0~H_3		0~10(Note 3)	mA
P_d	Power dissipation	T_a=25°C	600	mW
T_{opr}	Operating temperature		−10~70	°C
T_{stg}	Storage temperature		−40~125	°C

Note 1 : The total amount current flowing out of the IC must not exceed 15mA.
2 : The total amount current flowing into the IC must not exceed 20mA except port H.
3 : The total amount of port H current flowing into the IC must not exceed 40mA.

RECOMMENDED OPERATING CONDITIONS (T_a=−10~70°C, V_{DD}=5 V±10%, unless otherwise noted)

Symbol	Parameter	Limits			Unit
		Min	Nom	Max	
V_{DD}	Supply voltage	4.5	5	5.5	V
V_{SS}	Supply voltage		0		V
V_{IH}	"H" input voltage Ports E_0~E_3, D, $\overline{INT}$, L_2~L_3 G_0~G_3, HSYNC, VSYNC	0.7V_{DD}	V_{DD}	V_{DD}	V
V_{IH}	"H" input voltage Ports L_0, L_1	0.8V_{DD}	V_{DD}	V_{DD}	V
V_{IH}	"H" input voltage $\overline{RESET}$	0.9V_{DD}	V_{DD}	V_{DD}	V
V_{IL}	"L" input voltage Ports E_0~E_3, D, $\overline{INT}$, L_2~L_3 G_0~G_3, HSYNC, VSYNC, $\overline{RESET}$	0		0.3V_{DD}	V
V_{IL}	"L" input voltage Ports L_0, L_1	0	0	0.2V_{DD}	V
V_O	"H" output current Port H_0~H_3	0		12	V
I_O	"L" output current Port H_0~H_3			10	mA
f_{CPU}	Clock oscillating frequency (in CPU section)(Note 4)	3.6	4	4.4	MHz
f_{CRT}	Clock oscillating frequency (in CRT section)	4	5	6	MHz

Note 4 : Use a quartz crystal oscillator or a ceramic resonator for the CPU oscillating circuit.

ELECTRICAL CHARACTERISTICS (T_a=−10~70°C, V_{DD}=5 V±10%, f_{CPU}=4 MHz, unless otherwise noted)

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
V_{DD}	Supply voltage	f_{CPU}=4MHz	4.5	5	5.5	V
I_{DD}	Supply current	V_{DD}=5.5V f_{CPU}=4MHz at display off		1.5	6	mA
I_{DD}	Supply current	V_{DD}=5.5V f_{CPU}=4MHz f_{CRT}=5MHz at display on		4	15	mA
I_{OH}	"H" output current Ports F_0~F_3, J_0~J_3, D, R, G, B OUT, D-A	V_{DD}=4.5V V_{OH}=2.4V	−0.5			mA
I_{OL}	"L" output current D-A, D, Ports M_0~M_3, VDP_0~VDP_2 F_0~F_3, J_0~J_3, K_0~K_3, OUT	V_{DD}=4.5V V_{OL}=0.4V	0.5			mA
I_{OL}	"L" output current Ports G_0~G_3, L_0~L_3	V_{DD}=4.5V V_{OL}=0.4V	1			mA
I_{OL}	"L" output current R, G, B	V_{DD}=4.5V V_{OL}=0.4V	1			mA
I_{OL}	"L" output current Port H_0~H_3	V_{DD}=4.5V V_{OL}=3V	10			mA
V_{T+}−V_{T-}	Hysteresis Ports L_0, L_1	V_{DD}=5V at using as counter input	0.5	1	1.5	V
V_{T+}−V_{T-}	Hysteresis $\overline{RESET}$	V_{DD}=5V	0.5	1	1.5	V
R_U	Pull-up transistor Ports L_0~L_3, E_0~E_3, G_0~G_3 (Note 5)	V_{DD}=5V V_I=0V	15	30	60	kΩ
R_U	Pull-up transistor $\overline{INT}$, $\overline{RESET}$ (Note 5)	V_{DD}=5V V_I=0V	25	50	100	kΩ
I_{OZL}	Output leak current Ports G_0~G_3, L_0~L_3	V_O=5V			10	μA
I_{OZL}	Output leak current VDP_0~VDP_2, Ports M_0~M_3 K_0~K_3, H_0~H_3	V_O=12V			10	μA
V_O	Pressure-resistant output voltage Ports M_0~M_3, VDP_0 ~VDP_2, K_0~K_3, H_0~H_3				12	V

Note 5 : Pull-up transistor is an option.