MOS INTEGRATED CIRCUIT

μPD1704C

SINGLE-CHIP MICROCOMPUTER WITH A DIGITAL PHASE-LOCKED-LOOP FREQUENCY SYNTHESIZER

μPD1704 is a CMOS 4-bit single-chip microcomputer that incorporates a digital phase-locked-loop (PLL) and a display driver for a digital tuning system. μPD1704 instruction set includes 4-bit parallel addition and subtraction instructions (such as AD and SU), logical operation instructions (such as EXL), bit test instructions (such as TMT), carry F/F set and reset instructions (such as STC), interrupt function, and timer function.

μPD1704 provides 11 I/O ports that are controlled with input and output instructions (such as IN and OUT) and a variable duty port (VDP). A simple D/A converter can be configured easily by adding a low pass filter (LPF) to the VDP output. This device is available in a space saving 42-pin DIP (Dual In-line Package).

FEATURES

- 4-bit microcomputer for digital tuning
- Digital PLL frequency synthesizer
- Single power supply of 5 V ± 10 %
- Low power consumption CMOS
- Low power: Data (RAM) retention mode (with CE pin low)
- Programmable memory (ROM): 16 bits x 1016 steps
- Data memory (RAM): 4 bits x 128 (= 2 x 64) words
- 78 kinds of 1-word (16-bit) instructions
- Instruction execution time: 33.3 μs (4.5 MHz crystal oscillator)
- Ample addition and subtraction instructions with conditional skip (12 addition instructions and 12 subtraction instructions)
- Various conditional skip instructions
- Memory-to-memory data transfer within the same row address (X0H to XFH)
- Instructions for indirect data transfer between registers (such as MVRD and MVRS)
- 16 general purpose registers
- 3 stack levels
- No pages or fields contained in programmable memory (ROM) (Jump and subroutine call can be executed at any ROM addresses.)
- Dedicated Port for dynamic display (6 digits 7 segments) and key input
- FIP (Fluorescent Indicator Panel) direct drive (segments only)
- Clock stop instruction (Power supply current is redued to less than 10 μA. CE=0)
- 11 I/O ports: Port A (PA_0 to PA_3), individual lines can be configured as input or output. Port B (PB_0 to PB_3), Port C (PC_2, PC_3), and Port D (PD_3), these lines can be set to either input or output by port.
- Built-in variable duty port (VDP) with 64 duties. (This can be used to implement a D/A converter.)
- Effective I/O instructions (such as IN and OUT)
- Input and output port bits test instructions (TPT and TPF)
- External edge triggered interrupt ($\overline{INT}$ pin)
- Built-in timer F/F (This F/F is set every 125 ms. Time of day clock can be easily implemented.)
- PLL lock state can be tested using TUL instruction

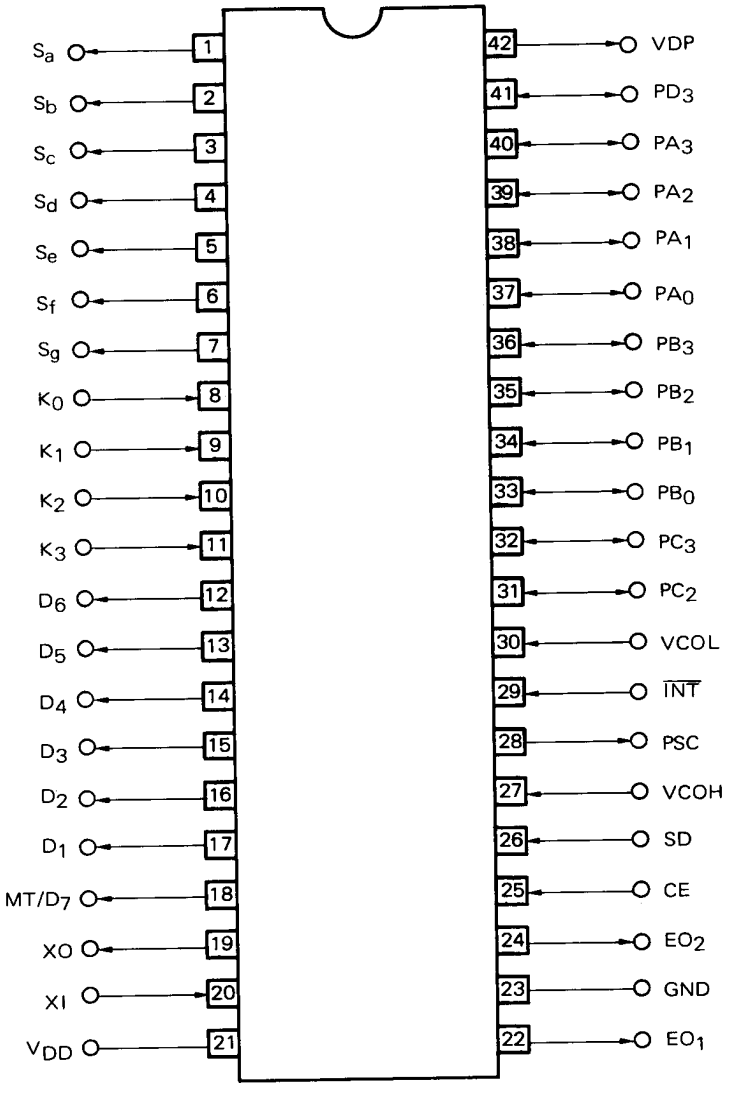
NEC cannot assume any responsibility for any circuits shown or represent that they are free from patent infringement.

© NEC Corporation 1986

This Material Copyrighted By Its Respective Manufacturer

- Transfer of frequency division ratio, frequency division method, and reference frequency data to PLL section with only one instruction, PLL
- Built-in programmable logic array (PLA) for segments and digits outputs (user mask programmable)
- Independent frequency input pins for VCOL and VCOH (maximum input frequency: 2.5 MHz for VCOL pin and 8.8 MHz for VCOH pin)
- Selection of pulse swallowing or direct frequency divisions by program
- Two Error Out pins, one for FM and the other for AM
- Direct connection to two-modulus prescaler, μPB553AC
- Selection of 6 reference frequencies by program (1, 5, 9, 10, 12.5, and 25 kHz)
- Easy FM IF frequency offset (N_F register)

PIN CONFIGURATION (Top View)



This Material Copyrighted By Its Respective Manufacturer

PIN DESCRIPTION

PIN NO.	SYMBOL	NAME	DESCRIPTION	OUTPUT TYPE
1 to 7	S_a to S_g	Segment Output	Display segment output pins. A SEG instruction unconditionally loads the contents of specified addresses in the data memory (RAM) to the programmable logic array (PLA). The selected data are output to these pins from PLA. (See the segment PLA.) These pins can also be used as a key return signal source for a key matrix.	P-ch Open drain
8 to 11	K_0 to K_3	Key-return Signal Input	Dedicated input port for key return signals from an external key matrix. The avaialble key return signal source is segment pins (S_a to S_g), digit pins (D_1 to D_6), or output ports. Executing a KIN instruction reads these pin states (0 or 1) to the memory (RAM) specified in its operand field. <Note> When the CE pin is low, do not input high-level signals to the K_0 to K_3 pins.	Input
12 to 17	D_6 to D_1	Digit Output	Display digit signal output pins. A DIG instruction unconditionally loads the contents of specified register in the data memory (RAM) to the programmable logic array (PLA). The Selected data are output to these pins from the PLA. (See the digit PLA.)	CMOS Push-pull
18	MT/D_7	Mute Signal Output	Output pin for mute signal. This pin is set or reset in a same manner as D_1 to D_6 with D16 instruction. When this pin is not used for mute output, it can be used for the display digit signal. It can be set arbitrarily during the digit PLA design.	CMOS Push-pull
19 20	XO XI	X'tal	Crystal oscillator connection pins. Connect a 4.5 MHz crystal oscillator to these pins.	Input (XI), CMOS Push-pull (XO)
21	V_{DD}	Power Supply	Device power supply pins. These pins supply 5 V ±10 % during device operation. The voltage can be dropped to 2.5 V to hold the internal data memory (RAM), after executing CKSTP instruction.	–
22 24	EO_1 EO_2	Error Output	PLL error output pins. When the divided oscillation frequency is higher than the reference frequency, these pins output a high-level signal; when it is lower, these pins output a low-level signal; and when it is the same, these pins float. The same signal is output to the EO_1 and EO_2 pins simultaneously.	CMOS 3-state

This Material Copyrighted By Its Respective Manufacturer

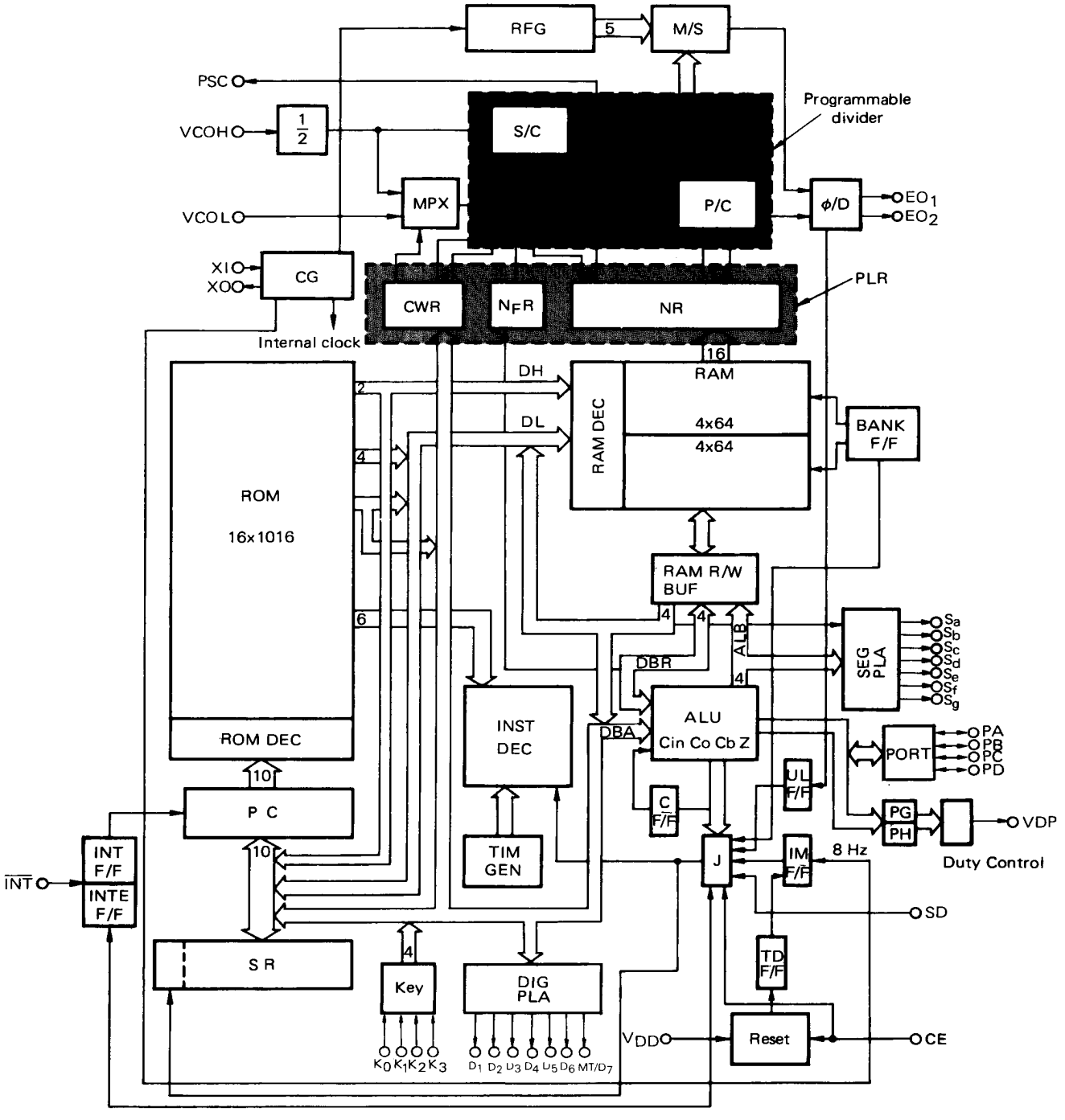
PIN NO.	SYMBOL	NAME	DESCRIPTION	OUTPUT TYPE
25	CE	Chip Enable	Device selection signal input pin. This pin must be high level to enable the device and low level to disable the device. An input of 134 μs or less is not accepted. When a CKSTP instruction in the program is executed while the CE pin is low level, the internal clock generator and CPU stop and the device can enter the stop mode requiring low power consumption (10 μA or less).	Input
26	SD	Station Detector	Pin for inputting a stop signal during automatic tuning. This pin state can be tested wiht a TSD instruction.	Input
28	PSC	Pulse Swallowing Control	Pin for outputting a frequency division ratio selection signal to two-modulus prescaler (μPB553AC) when the pulse swallow division method is used. Connect this pin to the PSC pin of μPB553AC. The frequency divsion ratio is selected between 1/16 or 1/17.	CMOS Push-pull
30	VCOL	VCO (Low) Signal Input	When a direct method (band L) is specified in the program, this pin generally inputs AM (MW) or LW local oscillation (VCO) output. The input frequency is 0.5 to 2.5 MHz.	Input
27	VCOH	VCO (High) Signal Input	When the pulse swallow method (band H) is specified in the program, this pin inputs the FM local oscillation (VCO) frequency divided in 1/16 or 1/17 by μPB553AC. The input frequency is 0.5 to 8.8 MHz.	Input
29	$\overline{INT}$	Interrupt	Interrupt request signal input pin. An interrupt request is issued at the trailing edge of the signal applied to this pin. When the interrupt request is accepted, control unconditionally jumps to adderss 1.	Input
37 to 40	PA_0 to PA_3	Port A	4-bit input/output port. The contents at address 1FH in data memory (RAM) bank 0 specify the input or output state of each bit at this port. (See note 1.)	CMOS Push-pull (I/O)
33 to 36	PB_0 to PB_3	Port B	4-bit input/output port. (See notes 1, 2, and 3.)	CMOS Push-pull (I/O)
31 32	PC_2 PC_3	Port C	2-bit input/output port. (See notes 1, 2 and 3.)	CMOS Push-pull (I/O)
41	PD_3	Port D	1-bit input/output port. (See notes 1, 2, and 3.)	CMOS Push-pull (I/O)
42	VDP	Variable Duty Port	Selection of variable duty port (VDP) or 1-bit output port (PG_2) can be selected by the program. When it is used as a VDP, this port continuously outputs 1.34 kHz pulses. Its duty can be changed in 64 steps.	CMOS Push-pull
23	GND	Ground	Ground pins. This pin is connected to ground of the system.	—

Notes:

1. PA_0 is associated with the least significant bit and PA_3 is associated with the most significant bit for port operation instructions (I/O, test, and set/reset instructions). This is the same also for PB, PC, PD, and PG. When PC, PD's condition is read by operating IN instruction, the contents of bits which are associated with not existed ports (PC_0, PC_1, PD_0 to PD_2) are not fixed.
2. When the device is reset (the CE pin goes from low to high), all input/output ports enters the input mode.
3. An IN instruction places all bits of ports B, C, or D in the input mode and OUT, SPB, and RPB instructions place them in the output mode. This is effective only for the first instruction after reset (CE pin low→high).

This Material Copyrighted By Its Respective Manufacturer

BLOCK DIAGRAM



This Material Copyrighted By Its Respective Manufacturer

CONTENTS

1. CPU 7
   - 1.1 PROGRAM COUNTER (PC) 7
   - 1.2 STACK REGISTER (SR) 7
   - 1.3 PROGRAMMABLE MEMORY (ROM) 7
   - 1.4 DATA MEMORY (RAM) 8
   - 1.5 TIMER F/F (TM F/F) 8
   - 1.6 UNLOCK F/F (UL F/F) 11
   - 1.7 CARRY F/F (C F/F) 11
   - 1.8 BANK F/F (B F/F) 11
   - 1.9 INT F/F, INTE F/F 11
   - 1.10 STATUS WORDS 12

2. PLL 13
   - 2.1 REFERENCE FREQUENCY GENERATOR (RFG) 13
   - 2.2 PHASE DETECTOR (ϕ-DET) 13
   - 2.3 PROGRAMMABLE DIVIDER (P/D) 13
   - 2.4 PLL REGISTER 14

3. PORTS 17
   - 3.1 PORT A, B, C, D 17
   - 3.2 VDP (VARIABLE DUTY PORT) 18

4. PLA 19
   - 4.1 PLA 19
   - 4.2 DIGIT PLA (DIG PLA) 24

5. INSTRUCTION 25
   - 5.1 INSTRUCTION SET 25
   - 5.2 INSTRUCTIONS 26

6. ELECTRICAL CHARACTERISTICS 29
   - 6.1 ABSOLUTE MAXIMUM RATINGS 29
   - 6.2 RECOMMENDED OPERATING CONDITIONS 29
   - 6.3 DC CHARACTERISTICS 30
   - 6.4 AC CHARACTERISTICS 30

7. PACKAGE DIMENSIONS 31

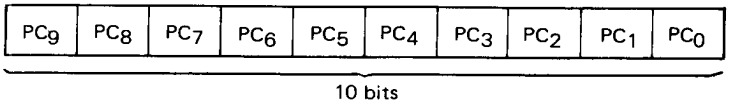
8. SUPPORT TOOLS 32

This Material Copyrighted By Its Respective Manufacturer

1. CPU

1.1 PROGRAM COUNTER (PC)

The program counter is a 10-bit binary counter used to address the programmable memory (ROM) that stores programs.



Usually, the counter is incremented by one each time an instruction is executed. When a jump or subroutine call instruction is executed, the address specified in the operand field is loaded to the counter. When an interrupt request is accepted, **current address+1 is unconditionally loaded to the counter.**

1.2 STACK REGISTER (SR)

The stack register is a 3 x 11 bit register which stores the program counter value plus 1, that is, the 10-bit return address, when a subroutine call instruction is executed or an interrupt request is accepted. In additoin, if an instruction having a skip function is executed when the interrupt request is being generated, the stack register also stores the 1-bit skip decision result. The stack register value is loaded to the program counter when a return instruction (RT or RTS) is executed and it returns to the source program flow.

The stack register can be used for both subroutine calls and interrupts. If one level is used for interrupts, the remaining two levels can be used for subroutine calls.

1.3 PROGRAMMABLE MEMORY (ROM)

The ROM consists of 16 bits x 1016 steps to store programs. The accessible ROM addresses range from 000H to 3F7H.

Since the ROM contains no pages or fields, JMP and CAL instructions can be executed at any address location to any part of program memory.

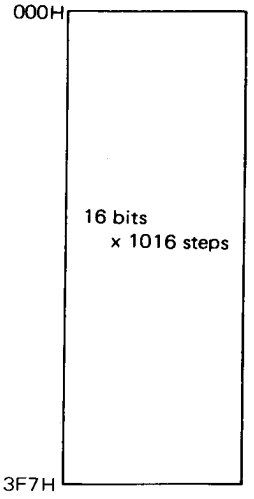


Fig. 1 ROM Configuration

This Material Copyrighted By Its Respective Manufacturer

1.4 DATA MEMORY (RAM)

The RAM consists of 4 bits x 128 words to store data. It is divided into banks 0 and 1 with 64 words each. For data processing in either bank, the bank must be specified previously with a BANK0 or BANK1 instruction.

Addresses 00H to 0FH in bank 0 compose general registers which can be used for calculation and data transfer to/from the memory. They can also be used as an ordinary memory. Bank specification is not necessary when locations 00 to 0FH are used as registers, but when it is used as a memory BANK0 or BANK1 must be specified.

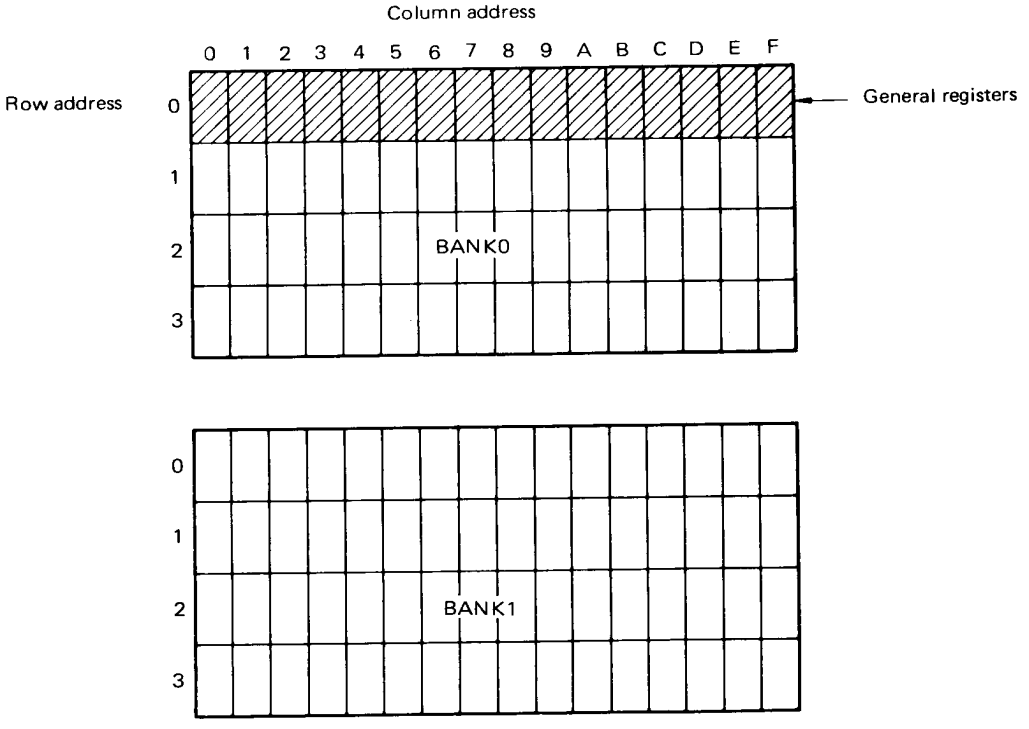


Fig. 2 RAM Configuration

1.5 TIMER F/F (TM F/F)

The timer F/F is set by an 8 Hz (125 ms) pulse that is output from the clock generator and reset by a test timer instruction (TTM).

Since the timer F/F is automatically set every 125 ms, it can be used for internal timer (8 counts per second) and mute time counter.

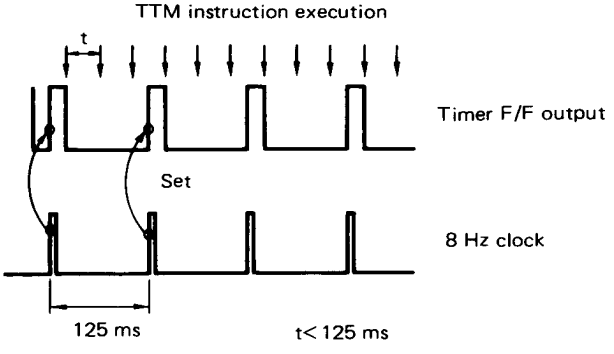
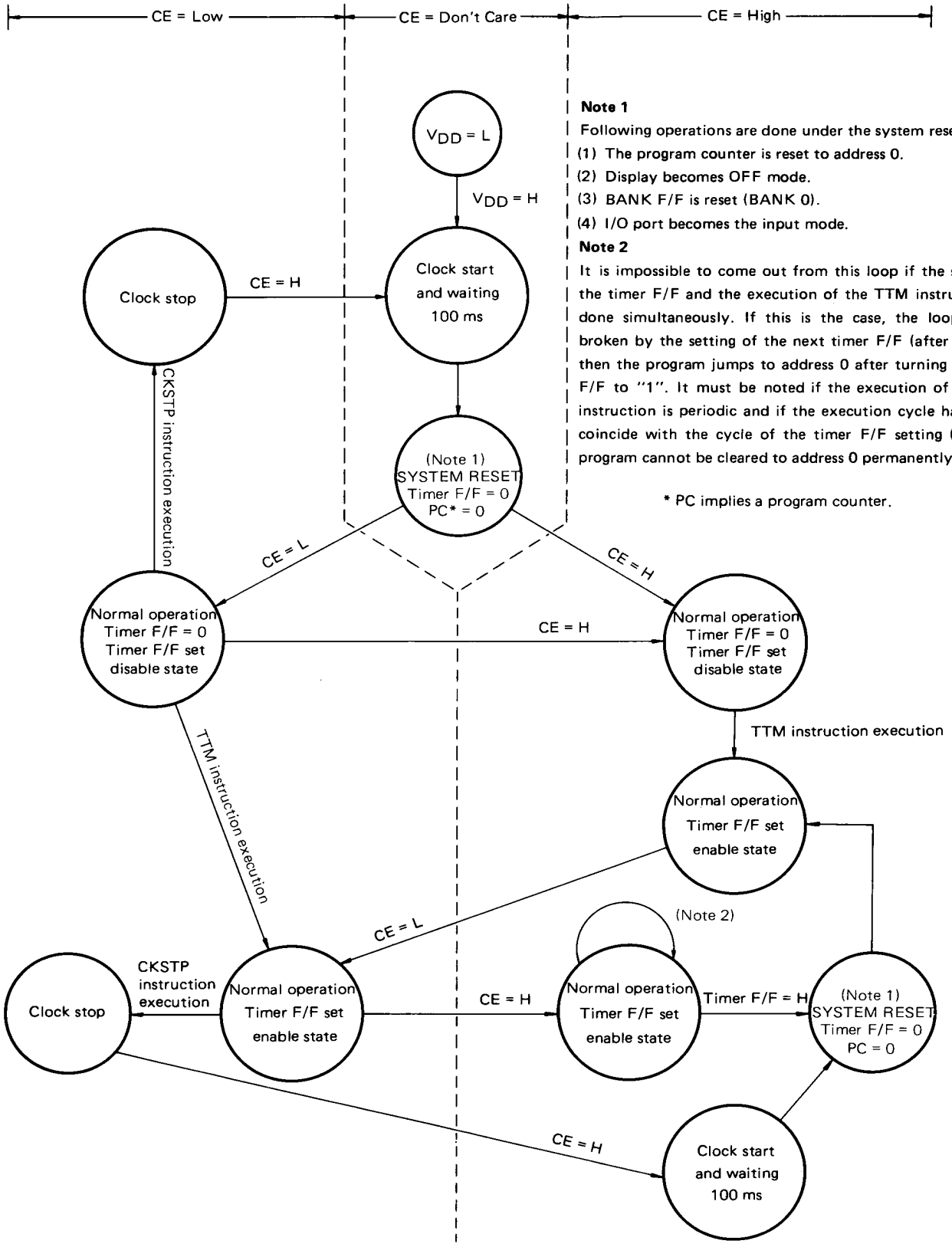


Fig. 3 TTM Instruction Execution Timings

This Material Copyrighted By Its Respective Manufacturer



Note 1

Following operations are done under the system reset status:

(1) The program counter is reset to address 0.

(2) Display becomes OFF mode.

(3) BANK F/F is reset (BANK 0).

(4) I/O port becomes the input mode.

Note 2

It is impossible to come out from this loop if the setting of the timer F/F and the execution of the TTM instruction are done simultaneously. If this is the case, the loop can be broken by the setting of the next timer F/F (after 125 ms), then the program jumps to address 0 after turning the timer F/F to "1". It must be noted if the execution of the TTM instruction is periodic and if the execution cycle happens to coincide with the cycle of the timer F/F setting (125 ms), program cannot be cleared to address 0 permanently.

* PC implies a program counter.

Fig. 4 CPU Status Transition Diagram Associated with CE Pin

This Material Copyrighted By Its Respective Manufacturer

As apparent from the above illustration, the program starts from the address 0 after the power is turned on (V_{DD} changes from Low to High), no matter what condition the CE pin is held, **while the timer F/F remains reset.** The timer F/F is not set again unless the TTM instruction was once executed (status unable to set the timer F/F). Once the TTM instruction was executed, however, the timer F/F can be set at any time at the intervals of 125 ms each.

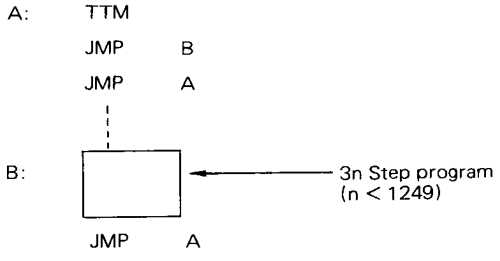
If the power is being fed (V_{DD}=high) and the CE pin changes from low to high, the program flow jumps to the address 0 immediately when the timer F/F is set. The program therefore starts from the address 0 **while the time F/F remains set.**

As you could understand well from the above explanation, the contents of Timer F/F vary between the time the power failure is recovered (V_{DD} changes from low to high) and the time when the power is continuously fed (V_{DD} =high and CE=low) or when the device is restored from the backed-up condition (but without when the device is restored from the clock stop condition.). Through testing the contents of this timer F/F (i.e. the execution of TTM instruction) it is possible to judge if it is restored from the power failure or from the non-power failure. In other words the power failure can be judged if the execution of TTM instruction, which is executed within 125 ms from the start of program from the address 0, results in 0 (false) or it can be determined as non-power failure (backed-up condition) if the result of test turns to 1 (true).

Care must be also taken to the programming when restoring from the non-power failure (V_{DD}=high and CE pin changes from low to high) while the clock function of a given program, if provided, needed to be operated (without using the CKSTP instruction) even if CE was low. The program flow in this case jumps to the address 0 immediately after the timer F/F is set. It is therefore necessary to update the clock after executing the TTM instruction to detect the power failure (the execution results in finding true). Otherwise the clock delays by 125 ms each whenever the CE pin changes from low to high.

Note 1: The program starts from the address 0 after the timer F/F was set if CE pin changed from low to high following the execution of CKSTP instruction in case of μPD1704.
The timer F/F is not used to detect the power failure in program using CKSTP instruction.
Therefore, we will advise to judge data of the RAM for detecting the power failure.

Note 2: Even if the CE pin changes from low to high level, the program flow does not move to address 0 when the setting of the timer F/F and the execution of the TTM instruction overlap. If this is the case, the system judges that the timer F/F is set by the TTM instruction execution and the timer F/F is reset. This point must be kept in mind when a power failure detection is done by the TTM instruction. That is, you should know that the TTM instruction has higher priority than the setting of the timer F/F when they overlap. Therefore, the clock does not become incorrect and a misjudging of a power failure does not occur. However, if the TTM instruction execution and the timer F/F setting happen to coincide in the following program, the program does not jump to address 0 forever. (The timer F/F will not be reset.)



In this example the program executes the normal TTM instruction, skips the next 'JMP B' instruction because the timer F/F is reset, and executes 'JMP A' instruction. Therefore, it cycles this loop. The cycle is 100 μs. (3 steps) The timer F/F is set once every 125 ms and the operations of B are performed. This operations take (3n + 3) steps (multiple of 100 μs). If the CE pin happens to change from low to high level during the TTM instruction execution in this program, operations in B are done due to the judgement that the timer F/F is set by the TTM instruction. However, the timer F/F is set again (125 ms later) when the next TTM instruction is executed because the time interval from the last TTM execution to the next TTM execution is a multiple of 100 μs. Therefore, the timer F/F will not be reset and the program cycles this endless loop. This problem occurs in a program where the TTM instruction is executed in every 125 ms. If this is the case, change the program so that the TTM instruction is not executed after 125 ms from the TTM execution.

This Material Copyrighted By Its Respective Manufacturer

1.6 UNLOCK F/F (UL F/F)

While the PLL is unlocked, that is, while the reference frequency (fr) does not match the divided frequency output from the VCO, the phase detector (ϕ-DET) outputs a pulse at the fr cycle. The unlock F/F is set by this pulse and reset by executing a TUL instruction. Therefore, a TUL instruction must be executed at a frequency greater than that of the reference frequency.

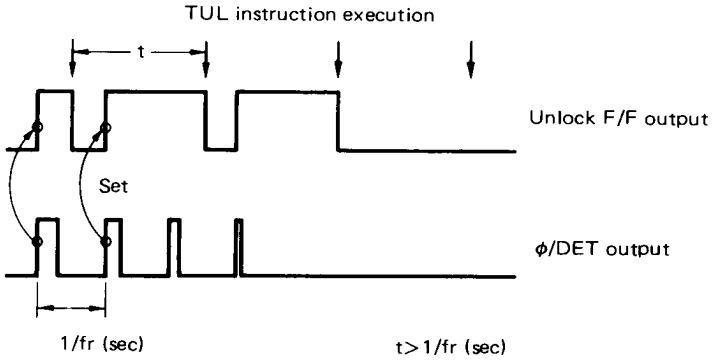


Fig. 5 TUL Instruction Execution Timings

1.7 CARRY F/F (C F/F)

Usually, when a carry or borrow is generated by executing an arithmetic instruction, the carry F/F is set. This F/F is reset when no carry or borrow is generated. The carry F/F can be set or reset directly by carry F/F set/reset instructions (STC and RSC) or status word operation instructions (SS and RS).

1.8 BANK F/F (B F/F)

This is a flip flop for specifying a BANK in the data memory (RAM). The RAM is divided into BANK 0 and BANK 1, with 64 words each. When the BANK F/F is reset, BANK 0 is specified; when the BANK F/F is set, BANK 1 is specified. This F/F can be set or reset by BANK0, BANK1, SS, and RS instructions.

1.9 INT F/F, INTE F/F

The INT F/F is set unconditionally by the trailing edge of a signal applied to the $\overline{\text{INT}}$ pin. If the internal INTE F/F is enabled (an EI instruction has been executed), an interrupt request is accepted; if it is disabled (no EI instruction has been executed), an interrupt request is not accepted. If the EI instruction is executed while the INT F/F is being set by external interrupt, the interrupt will be processed immediately after the EI instruction.

If the INTE F/F is being reset by instructions DI or RS while the INT F/F is being set by external interrupt, the interrupt will not be processed.

When an interrupt request is accepted, the DI state is set, INTE F/F reset, automatically and control jumps to address 1 (interrupt processing routine). At this time, the address of the instruction succeeding the generation of interrupt is stored in the stack register. If the instruction where the interrupt was generated had a skip function, the result of skip condition is also stored in the stack register. When an RT instruction is executed in the interrupt processing routine, the program counter and the skip status are restored by the contents stack register. Since the carry F/F and BANK F/F contents are not saved in the stack register, they must be saved by the interrupt routine, if they need to be.

Since the interrupt processing uses one stack level, stack level management is necessary.

This Material Copyrighted By Its Respective Manufacturer

1.10 STATUS WORDS

Internal device states which must be checked or specified for program execution are stored in 4-bit status words. They can be set or reset by programs.

External pins and Interrupt F/F are directly manipulated by status word 1 and 2.

(1)Status word 1 (write-only word)

Operation instructions: SS, RS, etc.

#3	#2	#1	#0
0	BANK F/F	Carry F/F	INTE F/F

Status word 1 is a write-only word which can be set and reset by instructions such as SS, RS, and EI.

(2)Status word 2 (read-only word)

Operation instructions: TST, TSF, etc.

#3	#2	#1	# 0
0	BANK F/F	CE pin	$\overline{\text{INT}}$ pin

Status word 2 is a read-only word whose contents can be judged by instructions such as TST, TSF, and SBK0.

This Material Copyrighted By Its Respective Manufacturer

2. PLL

2.1 REFERENCE FREQUENCY GENERATOR (RFG)

The external crystal oscillator (4.5 MHz) is used to generate six reference frequencies: 1 kHz, 5 kHz, 9 kHz, 12.5 kHz, and 25 kHz. One of these frequencies can be selected by the program.

2.2 PHASE DETECTOR (ϕ-DET)

The phase detector circuit detects phase difference between the reference frequency (fr) and VCO output divided by the programmable divider. The output of the ϕ-det is input to the internal charge pump which then outputs the pulses to the Error out pins.

EO_1 and EO_2 pins:

(1) $fr > f_{osc}/N$: Low level
(2) $fr < f_{osc}/N$: High level
(3) $fr = f_{osc}/N$: Floating

where, f_{osc} is the VCO oscillation frequency and N is the division ratio of the programmable divider.

2.3 PROGRAMMABLE DIVIDER (P/D)

The programmable divider consists of a swallow counter (S/C) and programmable counter (P/C) both are binary decrement counters.

The swallow counter is a 5-bit presettable counter. The contents of the 4-bit NR0 register and 1-bit N_F register that are loaded in PLL register are used to update this counter in the cycle of the reference frequency.

The programmable counter is a 12-bit counter. The contents of the NR1 to NR3 registers that are load from Data RAM by PLL instruction are used to update this counter at a period equal to $1/f_{ref}$. Both the programmable counter and the swallow counter are decremented simultaneously by the incoming signal via the VCOL and VCOH pins.

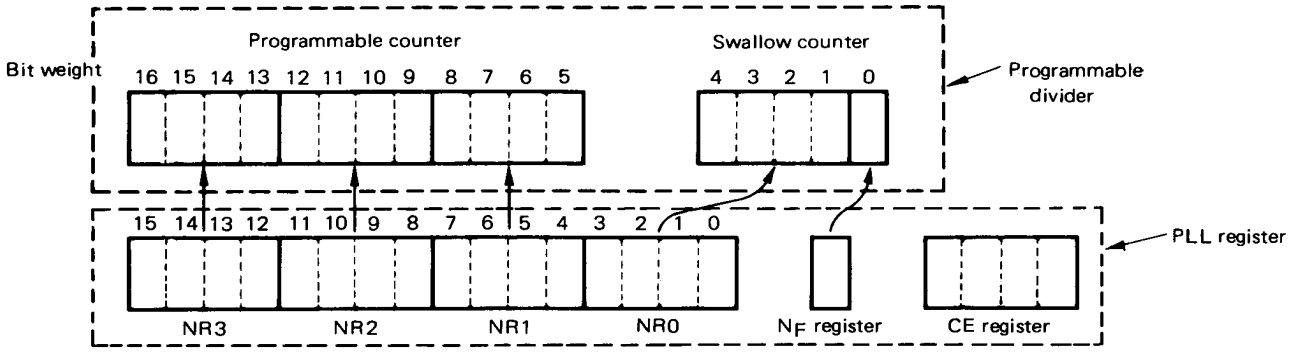


Fig. 6 Programmable Divider Configuration

When the direct method is used, the swallow counter contents are ignored, and only the programmable counter contents are used.

This Material Copyrighted By Its Respective Manufacturer

2.4 PLL REGISTER

To control the PLL, the following information must be stored in the PLL register:

(1) Division ratio (N)
(2) Reference frequency (fr)
(3) Division method (direct or pulse swallowing)

The PLL register consists of the 16-bit N register, 1-bit N_F register, and the 4-bit control word (CW) register. These registers are loaded by data in designated RAM location with a single instruction, PLL. The 16-bit N-registers are loaded by data in addresses 00H, 10H, 20H, and 30H of the RAM N0 being least significant nibble, the N_F bit is assigned to the most significant bit of specified general purpose register, and the CW is assigned to specified RAM location excluding the 4 locations assigned to N and the nibble containing the N_F bit. The CW and N_F locations are specified by operands of PLL instruction.

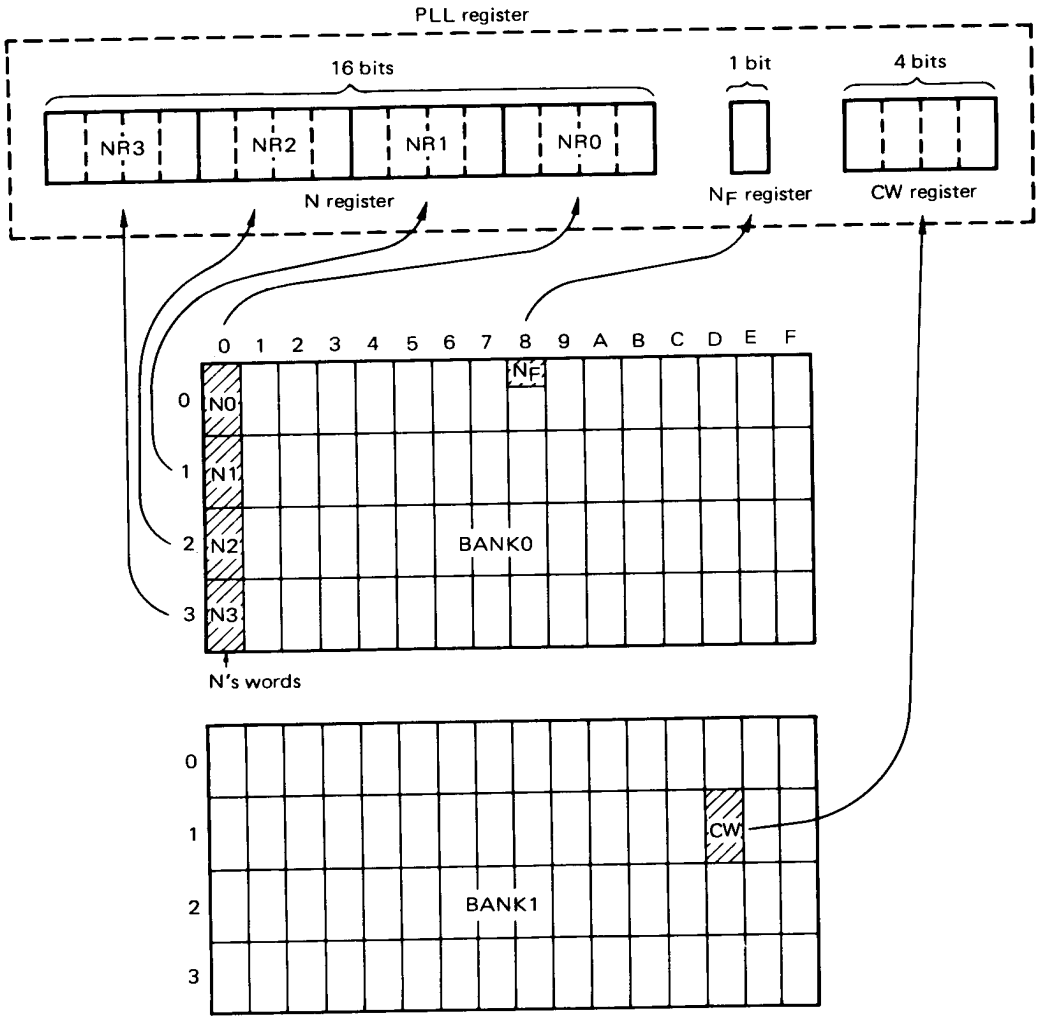


Fig. 7 PLL Instruction Functions

This Material Copyrighted By Its Respective Manufacturer

Table 1 Control Word Codes

Control word

#3	#2	#1	#0

0	Band L (direct type)
1	Band H (pulse swallow type)

	#2	#1	#0	fr(kHz)
	0	0	0	1
	0	1	0	5
	0	1	1	9
	1	0	0	10
	1	0	1	12.5
	1	1	0	25
0	1	1	1	Disable PLL

PLL Format Setting

The PLL format (division ratio, reference frequency, and division method) is set by the PLL instruction. Various methods are given below.

Example 1. Direct method

(Incoming frequency via VCOL pin: 1422 kHz, reference frequency: 9 kHz, intermediate frequency (IF): 450 kHz)

$$N = \frac{1422 + 450}{9} = 208$$

= 0D0H (H implies a hexadecimal code.)

N3	N2	N1	N0	N_F	CW
0 0 0 0	1 1 0 1	0 0 0 0	X X X X	X	0 0 1 1
0	D	0	Don't care		

Reference frequency (9 kHz)

Division method (direct)

In the direct method, the contents of N0 and N_F are ignored.

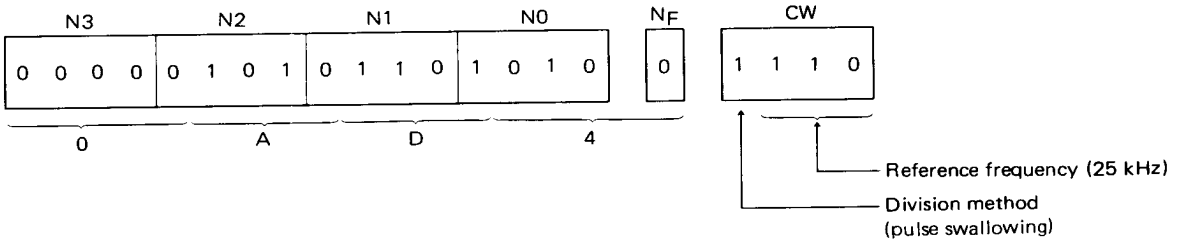
This Material Copyrighted By Its Respective Manufacturer

Example 2. Pulse swallowing method

(Incoming frequency via VCOH pin: 80.0 MHz, reference frequency: 25 kHz, IF: −10.7 MHz)

$$N = \frac{(80.0 - 10.7) \times 10^3}{25} = 2772$$

$$= 0AD4H$$



The VCO frequency can be changed by 25 kHz steps by incrementing or decrementing from NF bit. To change the VCO frequency in 50 kHz, 100 kHz, or 200 kHz steps, increment N0 register by 1, 2, or 4, respectively.

Usually, the N_F bit is used for IF fine tuning. When the N_F bit is set to 1 in the example above, division ratio N becomes 2773. Assuming that the receive frequency is fixed to 80.0 MHz, the IF (f_{IF}) is as follows:

$$\frac{(80.0 - f_{IF}) \times 10^3}{25} = 2773$$

$$f_{IF} = (69.325 - 80.0)$$

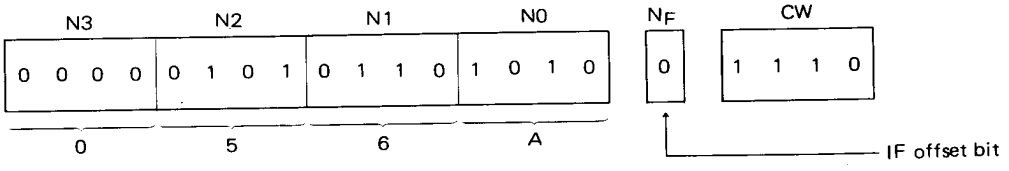
$$= -10.675 \text{ MHz}$$

This is equivalent to the IF changing by 25 kHz. Thus, the N_F bit can be used for IF correction.

The N value is calculated assuming that the N_F bit is the least significant bit in the above example; however, the program would be understood more easily by grouping every four bits from N0. That is, assume that the reference frequency is 50 kHz.

$$N = \frac{(80.0 - 10.7) \times 10^3}{50} = 1386$$

$$= 056AH$$



The contents of N-registers are the same as when the reference frequency is 25 kHz.

This Material Copyrighted By Its Respective Manufacturer

3. PORTS

The ports are addressed according to the two direct accessing bits and BANK F/F in the instruction operand. (See the table below.)

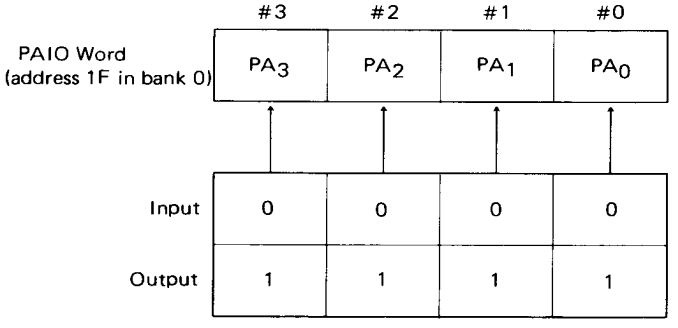
Table 2 Port Addressing

Direct Addr		BANK F/F	
#1	#0	BANK0	BANK1
0	0	PA	–
0	1	PB	–
1	0	PC	P_G (Note)
1	1	PD	P_H (Note)

Note: PG and PH are variable duty port (VDP) addresses.

3.1 PORT A, B, C, D

The I/O ports are grouped into Port A (PA_0 to PA_3), Port B (PB_0 to PB_3), Port C (PC_2 and PC_3), and Port D (PD_3). Port A (PA_0 to PA_3), the individual bits can be configured as input or output. This is determined by PAIO Word, address 1F in BANK 0 of the data memory (RAM). The input mode is set by writing a 0 and output mode by writing a 1 to the corresponding bit of PAIO Word.



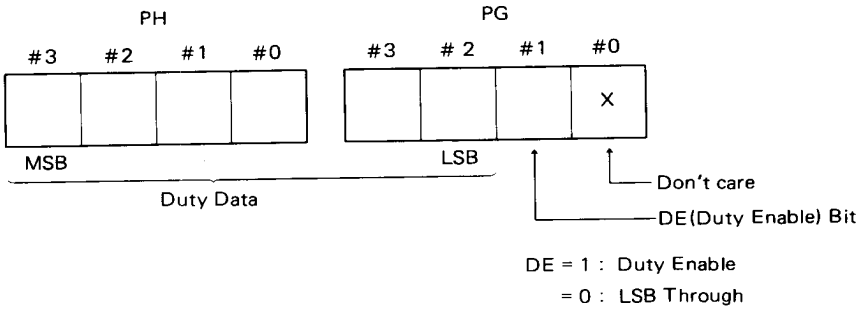
The other ports (Port B to Port D) can be set to the input or output mode by port units. Executing an input instruction (IN) automatically sets these ports in the input mode and executing an output instruction (OUT, SPB, or RPB) sets them in the output mode.

This Material Copyrighted By Its Respective Manufacturer

3.2 VDP (VARIABLE DUTY PORT)

The VDP continuously outputs a signal at 1.34 kHz with its duty cycle determined by Port H and Port G. 64 different duties can be set by the program.

Two internal ports (PG and PH) are allocated for duty control. They must be set as follows:



DE = 1 : Duty Enable
= 0 : LSB Through

When PG1 (duty enable bit) is set to 1, the VDP outputs 1.34 kHz signal with the six-bit data in PG3 to PH3, determining the duty cycle. The relationships between the duty data and the duty cycle of the signal output to the VDP pin are as follows:

$$\text{Duty} = \frac{\text{Duration of high level}}{\text{Cycle}} = \frac{(\text{duty data}) + 2}{67}$$

When PG1 (DE) goes to 0, the VDP outputs the PG2 contents. That is, when PG2 = 1, the VDP goes high; when PG2 = 0, the VDP goes low and it can be used as a 1-bit output port.

This Material Copyrighted By Its Respective Manufacturer

4. PLA

4.1 SEGMENT PLA (SEG PLA)

The segment PLA (SEG PLA) is a programmable logic array with five latched inputs and seven outputs. It can generate 32 segment patterns by using the 5-bit address.

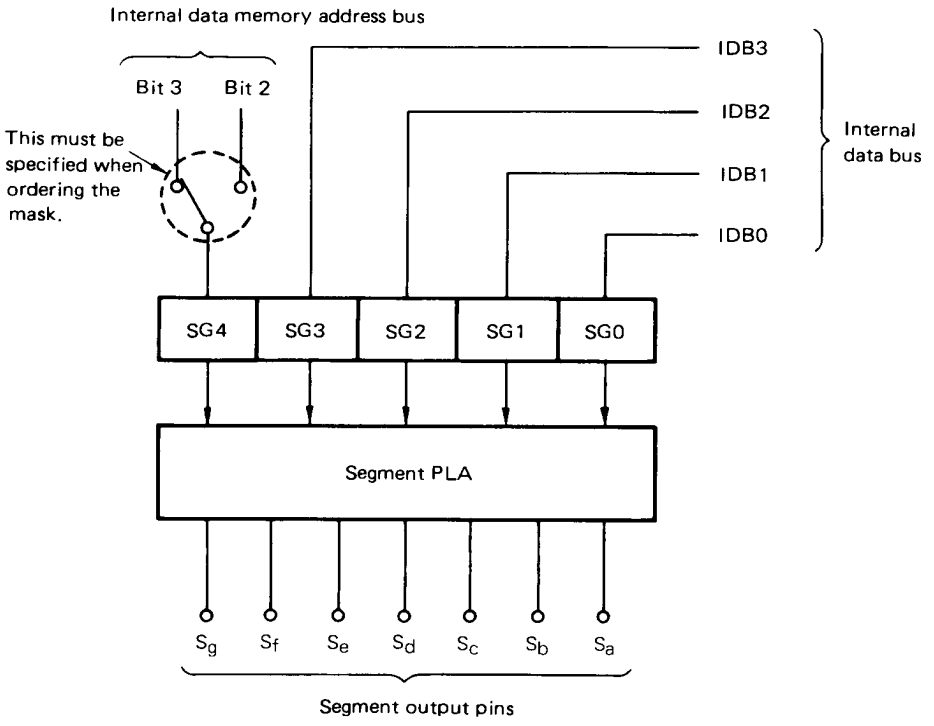


Fig. 8 Segment PLA Configuration

Of the five inputs, the four low-order bits (SG0 to SG3) are latched by the contents of the data memory specified in the SEG instruction operand. The most significant bit (SG4) latches bit 3 or bit 2 of the data memory address specified by the operand of SEG instruction.

Selection of which bit, either bit 2 or 3 of data memory address that is input to SG4, must be specified when ordering the mask.

Since the source bit of most significant bit of the 5-bit input to the segment PLA can be set during mask ordering, two different patterns can be generated according to the memory address storing the 4-bit data, even if the same data is input to the four low-order bits of segment PLA. Therefore, the segment PLA can be considered as two PLAs: each 4-bit inputs to the PLA generates 16 segment patterns with 7-bit output. One of these patterns can be selected by the most significant bit (SG4) value.

This Material Copyrighted By Its Respective Manufacturer

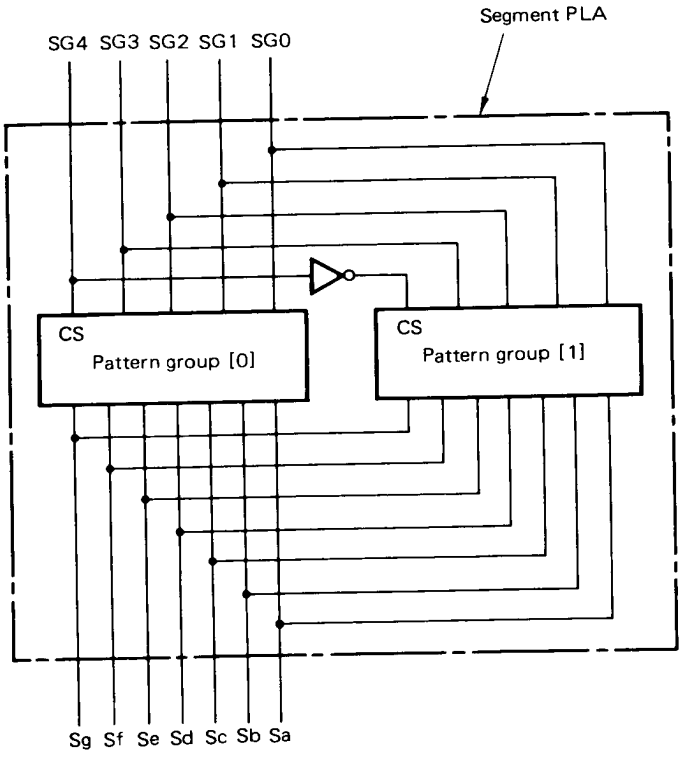


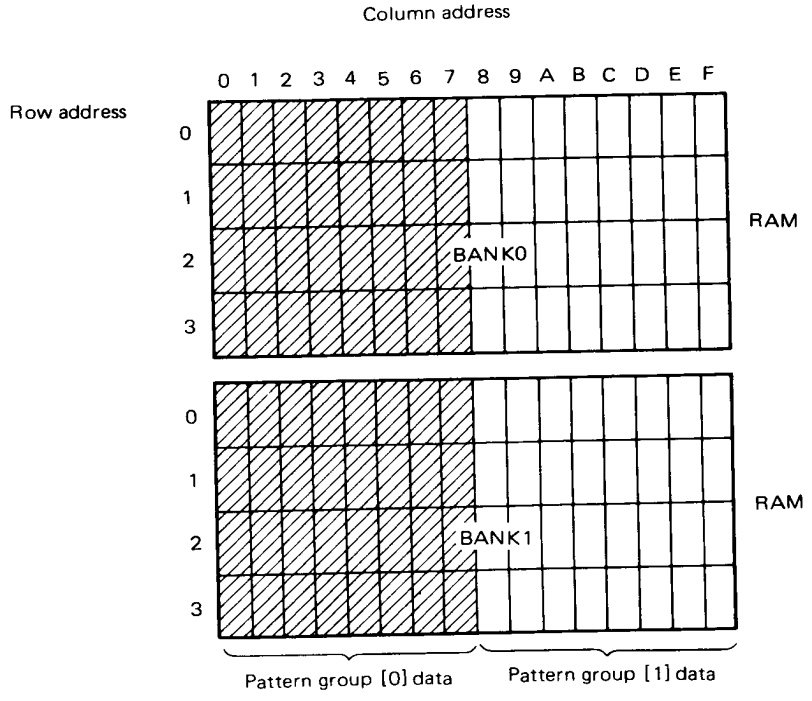
Fig. 9 Segment PLA Internal Configuration

The sixteen patterns generated when the SG4 bit is 0 are called pattern group [0] and those generated when the SG4 bit is 1 are called pattern group [1].

When data memory address bit 3 is selected to determine the SG4 value. The two PLA patterns are generated as follows;

Pattern [0] — generated with operand column address 0 to 7H

Pattern [1] — generated with operand column address 8 to 0FH

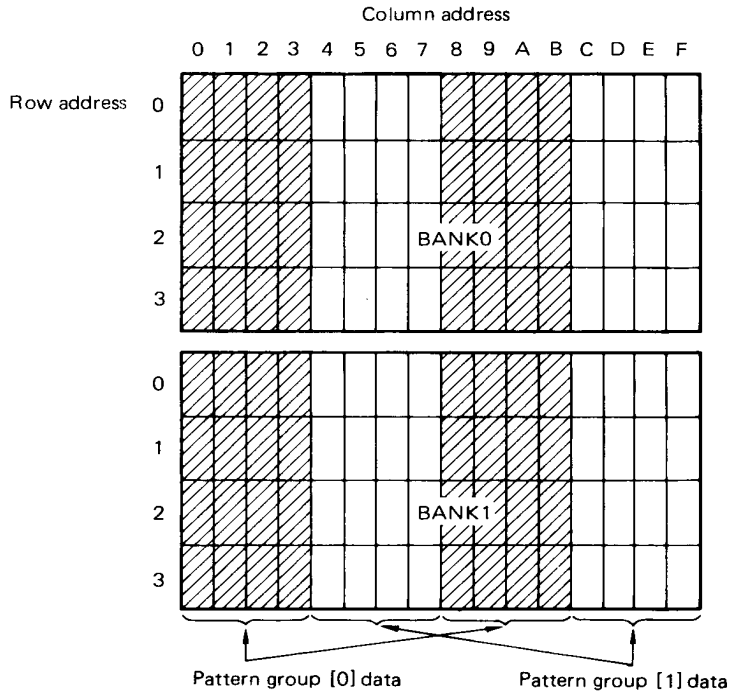


This Material Copyrighted By Its Respective Manufacturer

When data memory address bit 2 is selected to determine the SG4 value the two patterns are generated as follows:

Pattern [0] – generated with operand column address 0H to 3H and 8H to BH

Pattern [1] – generated with operand column address 4H to 7H and 0CH to 0FH



Tables 3 and 4 list the segment pattern examples of pattern groups [0] and [1] respectively.

This Material Copyrighted By Its Respective Manufacturer

Table 3 Segment Pattern Example

PATTERN GROUP [0]

SG4	Segment latch				Segment output							Remarks
	SG3	SG2	SG1	SG0	S_g	S_f	S_e	S_d	S_c	S_b	S_a	
0	0	0	0	0	0	1	1	1	1	1	1	0
	0	0	0	1	0	0	0	0	1	1	0	1
	0	0	1	0	1	0	1	1	0	1	1	2
	0	0	1	1	1	0	0	1	1	1	1	3
	0	1	0	0	1	1	0	0	1	1	0	4
	0	1	0	1	1	1	0	1	1	0	1	5
	0	1	1	0	1	1	1	1	1	0	1	6
	0	1	1	1	0	1	0	0	1	1	1	7
	1	0	0	0	1	1	1	1	1	1	1	8
	1	0	0	1	1	1	0	1	1	1	1	9
	1	0	1	0	0	0	0	0	0	0	0	Blank
	1	0	1	1	1	1	1	0	1	1	1	A
	1	1	0	0	1	0	0	0	0	0	0	-
	1	1	0	1	0	1	0	0	1	1	0	
	1	1	1	0	0	1	0	0	0	0	0	
	1	1	1	1	1	1	1	1	0	0	1	E

This Material Copyrighted By Its Respective Manufacturer

Table 4 Segment Pattern Example

PATTERN GROUP [1]

SG4	Segment latch				Segment output							Remarks
	SG3	SG2	SG1	SG0	S_g	S_f	S_e	S_d	S_c	S_b	S_a	
1	0	0	0	0	0	0	0	1	0	0	1	
	0	0	0	1	0	0	0	0	0	0	1	
	0	0	1	0	0	1	0	0	0	0	1	
	0	0	1	1	0	0	0	0	0	1	0	
	0	1	0	0	0	0	0	1	0	1	0	
	0	1	0	1	0	0	0	0	1	0	0	
	0	1	1	0	0	1	0	0	1	0	0	
	0	1	1	1	0	0	0	1	0	0	0	
	1	0	0	0	0	0	1	1	0	0	1	
	1	0	0	1	0	0	1	0	0	0	0	
	1	0	1	0	0	0	0	0	0	0	0	
	1	0	1	1	0	1	0	0	0	0	0	
	1	1	0	0	0	0	1	1	0	1	0	
	1	1	0	1	1	0	0	0	0	0	0	
	1	1	1	0	0	0	1	1	0	0	0	
	1	1	1	1	0	0	0	1	1	1	1	

Assume that data memory address bit 3 (column adderss most significant bit) has been specified as the SG4 input. When the column addresses storing the data to be input to SG0 to SG3 are 0 to 7H, a segment pattern is selected from those in group [0]; if the column addresses are 8 to FH, a segment pattern is selected from those in group [1].

This Material Copyrighted By Its Respective Manufacturer

4.2 DIGIT PLA (DIG PLA)

The digit PLA decodes the digit latch output to generate digit outputs D_1 to D_6 and MT/D_7.

The MT/D_7 pin can be used as a mute or digit output.

When the CE pin is low, D_1 to D_6 and MT/D_7 can be specified to be high or low individually during mask order.

Table 5 gives an example of digit control PLA input/output data. The digit latch bits (DL3 to DL0) have the following meanings. In this example, the most significant digit output is used as mute control data.

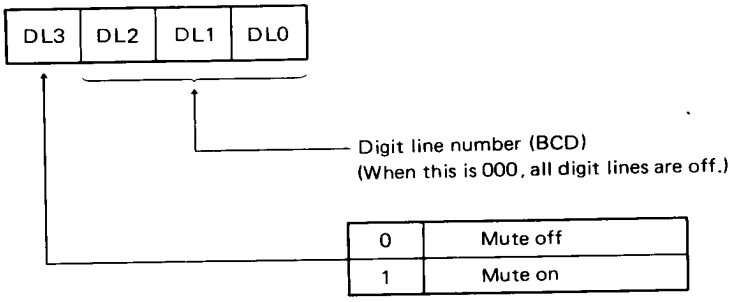


Table 5 Example of Digit Line Control PLA Input/Output Codes

INPUT				OUTPUT						
DL3	DL2	DL1	DL0	MT/D_7	D_6	D_5	D_4	D_3	D_2	D_1
0	0	0	0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	0	0	0	1
0	0	1	0	0	0	0	0	0	1	0
0	0	1	1	0	0	0	0	1	0	0
0	1	0	0	0	0	0	1	0	0	0
0	1	0	1	0	0	1	0	0	0	0
0	1	1	0	0	1	0	0	0	0	0
0	1	1	1	0	0	0	0	0	0	0
1	0	0	0	1	0	0	0	0	0	0
1	0	0	1	1	0	0	0	0	0	1
1	0	1	0	1	0	0	0	0	1	0
1	0	1	1	1	0	0	0	1	0	0
1	1	0	0	1	0	0	1	0	0	0
1	1	0	1	1	0	1	0	0	0	0
1	1	1	0	1	1	0	0	0	0	0
1	1	1	1	1	0	0	0	0	0	0

This Material Copyrighted By Its Respective Manufacturer

5. INSTRUCTION

5.1 INSTRUCTION SET

b_{13} b_{12} b_{11} b_{10}		b_{15} b_{14} = 0 0	0 1	1 0	1 1
		0	1	2	3
0 0 0 0	0	NOP	KIN M	DIG r	ST M, r
0 0 0 1	1	SPB P, N SS N_1 BANK1 EI STC	ORI M, I	SEG D_H, r	MVRS M, r
0 0 1 0	2	———	MVI M, I	OUT P, r	IN r, P
0 0 1 1	3	RPB P, N RS N_1 BANK0 DI RSC	ANI M, I	CKSTP	MVRD r, M
0 1 0 0	4	RT	AI M, I	MVSR M_1, M_2	AD r, M
0 1 0 1	5	RTS	SI M, I	EXL r, M	SU r, M
0 1 1 0	6	JMP ADDR	AIC M, I	LD r, M	AC r, M
0 1 1 1	7	CAL ADDR	SIB M, I	———	SB r, M
1 0 0 0	8	SBK0 TPF P, N TSF N_2 TCEF TITF	AIN M, I	TSD	ADN r, M
1 0 0 1	9	SBK1 TPT P, N TST N_2 TCET TITT	SIN M, I	TTM	SUN r, M
1 0 1 0	A	TMF M, N	AICN M, I	TUL	ACN r, M
1 0 1 1	B	TMT M, N	SIBN M, I	PLL M, r	SBN r, M
1 1 0 0	C	SLTI M, I	AIS M, I	SLT r, M	ADS r, M
1 1 0 1	D	SGEI M, I	SIS M, I	SGE r, M	SUS r, M
1 1 1 0	E	SEQI M, I	AICS M, I	SEQ r, M	ACS r, M
1 1 1 1	F	SNEI M, I	SIBS M, I	SNE r, M	SBS r, M

This Material Copyrighted By Its Respective Manufacturer

5.2 INSTRUCTIONS

Explanatory note

: D_H : Data memory address high (row address) [2 bits]
D_L : Data memory address low (column address) [4 bits]
Rn : Register number [4 bits]
I : Immediate data [4 bits]
N : Bit position [4 bits]
ADDR : Program memory address [10 bits]
—— : All "1"
r : General register
One of addresses 00–0FH of BANK0

M : Data memory address
One of data 00–3FH of BANK0 and 00–3FH of BANK1
P : Port $0 \leqq P \leqq 3$
N_1 : Bit position of status word 1 $0 \leqq N_1 \leqq 7$
N_2 : Bit position of status word 2 $0 \leqq N_2 \leqq 7$
( ) : Contents of register or memory
c : Carry
b : Borrow
( )n : Contents on bit n of register or memory
{ } : Decoded value of contents of register or memories

	Mnemonic	Operand 1ST	Operand 2ND	Function	Operation	Machine code: Operation code			
Addition	AD	r	M	Add memory to register	r←(r) + (M)	1 1 0 1 0 0	D_H	D_L	Rn
	ADS	r	M	Add memory to register, then skip if carry	r←(r) + (M) skip if carry	1 1 1 1 0 0	D_H	D_L	Rn
	ADN	r	M	Add memory to register, then skip if not carry	r←(r) + (M) skip if not carry	1 1 1 0 0 0	D_H	D_L	Rn
	AC	r	M	Add memory to register with carry	r←(r) + (M) + c	1 1 0 1 1 0	D_H	D_L	Rn
	ACS	r	M	Add memory to register with carry, then skip if carry	r←(r) + (M) + c skip if carry	1 1 1 1 1 0	D_H	D_L	Rn
	ACN	r	M	Add memory to register with carry, then skip if not carry	r←(r) + (M) + c skip if not carry	1 1 1 0 1 0	D_H	D_L	Rn
	AI	M	I	Add immediate data to memory	M←(M) + I	0 1 0 1 0 0	D_H	D_L	I
	AIS	M	I	Add immediate data to memory, then skip if carry	M←(M) + I skip if carry	0 1 1 1 0 0	D_H	D_L	I
	AIN	M	I	Add immediate data to memory, then skip if not carry	M←(M) + I skip if not carry	0 1 1 0 0 0	D_H	D_L	I
	AIC	M	I	Add immediate data to memory with carry	M←(M) + I + c	0 1 0 1 1 0	D_H	D_L	I
	AICS	M	I	Add immediate data to memory with carry, then skip if carry	M←(M) + I + c skip if carry	0 1 1 1 1 0	D_H	D_L	I
	AICN	M	I	Add immediate data to memory with carry, then skip if not carry	M←(M) + I + c skip if not carry	0 1 1 0 1 0	D_H	D_L	I
Subtraction	SU	r	M	Subtract memory from register	r←(r) − (M)	1 1 0 1 0 1	D_H	D_L	Rn
	SUS	r	M	Subtract memory from register, then skip if borrow	r←(r) − (M) skip if borrow	1 1 1 1 0 1	D_H	D_L	Rn
	SUN	r	M	Subtract memory from register, then skip if not borrow	r←(r) − (M) skip if not borrow	1 1 1 0 0 1	D_H	D_L	Rn
	SB	r	M	Subtract memory from register with borrow	r←(r) − (M) − b	1 1 0 1 1 1	D_H	D_L	Rn
	SBS	r	M	Subtract memory from register with borrow, then skip if borrow	r←(r) − (M) − b skip if borrow	1 1 1 1 1 1	D_H	D_L	Rn
	SBN	r	M	Subtract memory from register with borrow, then skip if not borrow	r←(r) − (M) − b skip if not borrow	1 1 1 0 1 1	D_H	D_L	Rn
	SI	M	I	Subtract immediate data from memory	M←(M) − I	0 1 0 1 0 1	D_H	D_L	I
	SIS	M	I	Subtract immediate data from memory, then skip if borrow	M←(M) − I skip if borrow	0 1 1 1 0 1	D_H	D_L	I
	SIN	M	I	Subtract immediate data from memory, then skip if not borrow	M←(M) − I skip if not borrow	0 1 1 0 0 1	D_H	D_L	I
	SIB	M	I	Subtract immediate data from memory with borrow	M←(M) − I − b	0 1 0 1 1 1	D_H	D_L	I
	SIBS	M	I	Subtract immediate data from memory with borrow, then skip if borrow	M←(M) − I − b skip if borrow	0 1 1 1 1 1	D_H	D_L	I
	SIBN	M	I	Subtract immediate data from memory with borrow, then skip if not borrow	M←(M) − I − b skip not borrow	0 1 1 0 1 1	D_H	D_L	I

This Material Copyrighted By Its Respective Manufacturer

	Mnemonic	Operand 1ST	Operand 2ND	Function	Operation	Machine code: Operation code			
Comparison	SEQ	r	M	Skip if register equals memory	r − M skip if zero	1 0 1 1 1 0	D_H	D_L	Rn
	SNE	r	M	Skip if register not equals memory	r − M skip if not zero	1 0 1 1 1 1	D_H	D_L	R_n
	SGE	r	M	Skip if register is greater than or equal to memory	r − M skip if not borrow (r) ≧ (M)	1 0 1 1 0 1	D_H	D_L	R_n
	SLT	r	M	Skip if register is less than memory	r − M skip if borrow (r) < (M)	1 0 1 1 0 0	D_H	D_L	R_n
	SEQI	M	I	Skip if memory equals immediate data	M − I skip if zero	0 0 1 1 1 0	D_H	D_L	I
	SNEI	M	I	Skip if memory not equals immediate data	M − I skip if not zero	0 0 1 1 1 1	D_H	D_L	I
	SGEI	M	I	Skip if memory is greater than or equal to immediate data	M − I skip if not borrow (M) ≧ I	0 0 1 1 0 1	D_H	D_L	I
	SLTI	M	I	Skip if memory is less than immediate data	M − I skip if borrow (M) < I	0 0 1 1 0 0	D_H	D_L	I
Logical operation	ANI	M	I	Logic AND of memory and immediate data	M←(M) ∧ I	0 1 0 0 1 1	D_H	D_L	$\bar{I}$
	ORI	M	I	Logic OR of memory and immediate data	M←(M) ∨ I	0 1 0 0 0 1	D_H	D_L	I
	EXL	r	M	Exclusive OR Logic of memory and register	r←(r)⊕(M)	1 0 0 1 0 1	D_H	D_L	R_n
Transfer	LD	r	M	Load memory to register	r←(M)	1 0 0 1 1 0	D_H	D_L	R_n
	ST	M	r	Store register to memory	M←(r)	1 1 0 0 0 0	D_H	D_L	R_n
	MVRD	r	M	Move memory to destination memory referring to register in the same row	[D_H, Rn]←(M)	1 1 0 0 1 1	D_H	D_L	R_n
	MVRS	M	r	Move source memory referring to register to memory in the same row	M←[D_H, Rn]	1 1 0 0 0 1	D_H	D_L	R_n
	MVSR	•M_1	M_2	Move memory to memory in the same row	[D_H, DL_1]←[D_H, DL_2]	1 0 0 1 0 0	D_H	D_{L1}	D_{L2}
	MVI	M	I	Move immediate data to memory	M← I	0 1 0 0 1 0	D_H	D_L	I
	PLL	M	r	Load N0~N3, N_F & memory to PLL registers	PLLR←(N0~N3), N_F & (M)	1 0 1 0 1 1	D_H	D_L	R_n
Bit test	TMT	M	N	Test memory bits, then skip if all bits specified are true	if M(N) =all "1", then skip	0 0 1 0 1 1	D_H	D_L	N
	TMF	M	N	Test memory bits, then skip if all bits specified are false	if M(N) =all "0", then skip	0 0 1 0 1 0	D_H	D_L	N
Jump	JMP	ADDR		Jump to the address specified	PC←ADDR	0 0 0 1 1 0	ADDR (10 bits)		
Subroutine	CAL	ADDR		Call subroutine	Stack←(PC) + 1	0 0 0 1 1 1	ADDR (10 bits)		
	RT			Return to main routine	PC←(stack)	0 0 0 1 0 0	—	—	—
	RTS			Return to main routine, then skip unconditionary	PC←(stack), and skip	0 0 0 1 0 1	—	—	—
Interrupt	EI			Enable interrupt	INTE F/F ← 1	0 0 0 0 0 1	—	0 0 0 1	—
	DI			Disable interrupt	INTE F/F ← 0	0 0 0 0 1 1	—	0 0 0 1	—
F/F test	TTM			Test and reset timer F/F, then skip if it has not been set	if Timer F/F = 1, then Timer F/F←0 if Timer F/F = 0, then skip	1 0 1 0 0 1	—	—	—
	TUL			Test and reset unlock F/F, then skip if it has not been set	if UL F/F = 1, then UL F/F←0 if UL F/F = 0, then skip	1 0 1 0 1 0	—	—	—

This Material Copyrighted By Its Respective Manufacturer

	Mnemonic	Operand 1ST	Operand 2ND	Function	Operation	Machine code: Operation code			
Status word and terminal test	SS	N_1		Set status word 1	(STATUS WORD 1)$_N$←1	0 0 0 0 0 1	—	0 N_1	—
	RS	N_1		Reset status word 1	(STATUS WORD 1)$_N$← 0	0 0 0 0 1 1	—	0 N_1	—
	TST	N_2		Test status word 2 true	if (STATUS WORD 2)$_N$=all "1", then skip	0 0 1 0 0 1	—	0 N_2	—
	TSF	N_2		Test status word 2 false	if (STATUS WORD 2)$_N$=all "0", then skip	0 0 1 0 0 0	—	0 N_2	—
	STC			Set carry F/F	carry F/F← 1	0 0 0 0 0 1	—	0 0 1 0	—
	RSC			Reset carry F/F	carry F/F← 0	0 0 0 0 1 1	—	0 0 1 0	—
	BANK0			Select BANK0	BANK F/F← 0	0 0 0 0 1 1	—	1 1 0 0	—
	BANK1			Select BANK1	BANK F/F← 1	0 0 0 0 0 1	—	0 1 0 0	—
	TITT			Test INT, skip if true	if $\overline{INT}$ = 0, then skip	0 0 1 0 0 1	—	0 0 0 1	—
	TITF			Test INT, skip if false	if $\overline{INT}$ = 1, then skip	0 0 1 0 0 0	—	0 0 0 1	—
	TCET			Test CE, skip if true	if CE= 1, then skip	0 0 1 0 0 1	—	0 0 1 0	—
	TCEF			Test CE, skip if false	if CE = 0, then skip	0 0 1 0 0 0	—	0 0 1 0	—
	SBK0			Skip if BANK0	if BANK F/F =0, then skip	0 0 1 0 0 0	—	1 1 0 0	—
	SBK1			Skip if BANK1	if BANK F/F =1, then skip	0 0 1 0 0 1	—	0 1 0 0	—
	TSD			Test station detector, then skip if it is not high level	Skip if SD = 0	1 0 1 0 0 0	—	—	—
Input / output	SEG	D_H	r	Output segment pattern based on the memory specified indirectly	$SG_{0\sim4}$←{D_H, (Rn)}, $S_{a\sim g}$←{$SG_{0\sim4}$}	1 0 0 0 0 1	D_H	—	Rn
	DIG	r		Output digit pattern based on the register	$DL_{0\sim3}$←{(Rn)$_{0\sim3}$} {$D_{1\sim6}$, MT/D_7}←$DL_{0\sim3}$	1 0 0 0 0 0	—	—	r
	KIN	M		Input key data to memory, then skip if data are zero	M←$K_{0\sim3}$, skip if (M) = 0	0 1 0 0 0 0	D_H	D_L	—
	IN	r	P	Input data on port to register	r←(Port (P))	1 1 0 0 1 0	P	—	Rn
	OUT	P	r	Output contents of register to port	(Port (P)) ← (r)	1 0 0 0 1 0	P	—	Rn
	SPB	P	N	Set port bits	(Port (P))$_N$←1	0 0 0 0 0 1	P	0 0 0 0	N
	RPB	P	N	Reset port bits	(Port (P))$_N$←0	0 0 0 0 1 1	P	0 0 0 0	N
	TPT	P	N	Test port bits, then skip if all bits specified are true	if (Port (P))$_N$=all 1s, then skip	0 0 1 0 0 1	P	0 0 0 0	N
	TPF	P	N	Test port bits, then skip if all bits specified are false	if (Port (P))$_N$ =all 0s, then skip	0 0 1 0 0 0	P	0 0 0 0	N
Others	CKSTP			Clock stop by CE	stop clock if CE = 0	1 0 0 0 1 1	—	1 1 1 0	1 1 1 0
	NOP			No operation		0 0 0 0 0 0	—	—	—

This Material Copyrighted By Its Respective Manufacturer

6. ELECTRICAL CHARACTERISTICS

6.1 ABSOLUTE MAXIMUM RATINGS

Supply Voltage	V_{DD}	−3.0 to +6.0	V
Input Voltage	V_I	−0.3 to $+V_{DD}$	V
Output Voltage	V_O	−0.3 to $+V_{DD}$	V
Output Sink Current	I_O	10	mA
Operating Temperature	T_a	−35 to +75	°C
Storage Temperature	T_{stg}	−55 to +125	°C
Output Breakdown Voltage	V_{BDS}	S_a to S_g pin (Voltage between source drains) −35	V

6.2 RECOMMENDED OPERATING CONDITIONS

CHARACTERISTIC	SYMBOL	MIN.	TYP.	MAX.	UNIT	CONDITION
Supply Voltage	V_{DD}	4.5	5.0	5.5	V	
Data Retention Voltage	V_{DR}	2.5		5.5	V	CE = 0, Clock Stop (CKSTP instruction execution)
Crystal Oscillator Stop Voltage	V_{DDS}		3.2	3.8	V	
Output Breakdown Voltage	V_{BDS}			−30	V	S_a to S_g pin (Voltage between source drains) I_{OH} = −5 μA
Supply Voltage Rise Time	T_{rise}			500	ms	V_{DD} = 0 → 4.5 V

This Material Copyrighted By Its Respective Manufacturer

6.3 DC CHARACTERISTICS (V_{DD} = +4.5 to 5.5 V, T_a = −35 to 75 °C)

CHARACTERISTICS	SYMBOL	MIN.	TYP.	MAX.	UNIT	CONDITION
High Level Input Voltage	V_{IH1}	0.8 V_{DD}		V_{DD}	V	SD and $\overline{INT}$ pins
High Level Input Voltage	V_{IH2}	0.7 V_{DD}		V_{DD}	V	I/O port and CE pin
High Level Input Voltage	V_{IH3}	0.6 V_{DD}		V_{DD}	V	K_0 to K_3 pins
Low Level Input Voltage	V_{IL1}	0		0.3 V_{DD}	V	I/O port and CE pin
Low Level Input Voltage	V_{IL2}	0		0.2 V_{DD}	V	K_0 to K_3, SD, and $\overline{INT}$ pins
High Level Output Voltage	V_{OH1}	4.0			V	EO_1 and EO_2 pins I_{OH}=−0.5 mA
High Level Output Voltage	V_{OH2}	4.0			V	D_1 to D_6, MT/D_7, I/O port I_{OH}=−0.2 mA
High Level Output Voltage	V_{OH3}	4.0			V	PSC pin I_{OH}=−0.1 mA
High Level Output Voltage	V_{OH4}	3.0			V	S_a to S_g pins I_{OH}=−0.5 mA
Low Level Output Voltage	V_{OL1}			0.5	V	EO_1 and EO_2 pins, I/O port I_{OL}=0.5 mA
Low Level Output Voltage	V_{OL2}			0.5	V	D_1 to D_6, MT/D_7, PSC pins I_{OL}=0.2 mA
High Level Input Current	$+I_{IH1}$	25	50	100	μA	K_0 to K_3 pins V_{IN}=V_{DD}=5.0 V
High Level Input Current	$+I_{IH2}$		300		μA	XI pin V_{IN}=V_{DD}=5.0 V
Low Level Input Current	$-I_{IL1}$		300		μA	VCOL and VCOH pins V_{DD}=5.0 V V_{IN}=0 V
Output Leakage Current	I_L		10^{-3}	1	μA	EO_1 and EO_2 pins T_a=25 °C
Supply Current	I_{DD1}		3		mA	Except the I/O current from I/O pins
Supply Current	I_{DD2}		0.6		mA	CE=0, PLL stoped, CPU operated only T_a=25 °C, V_{DD}=5.0 V
Data Retention Current	I_{DR}			10	μA	CE=0, CKSTP instruction execution T_a=25 °C, V_{DD}=5.0 V

Note I/O ports means the PA_0 to PA_3, PB_0 to PB_3, PC_2, PC_3 and PD_3 pins.

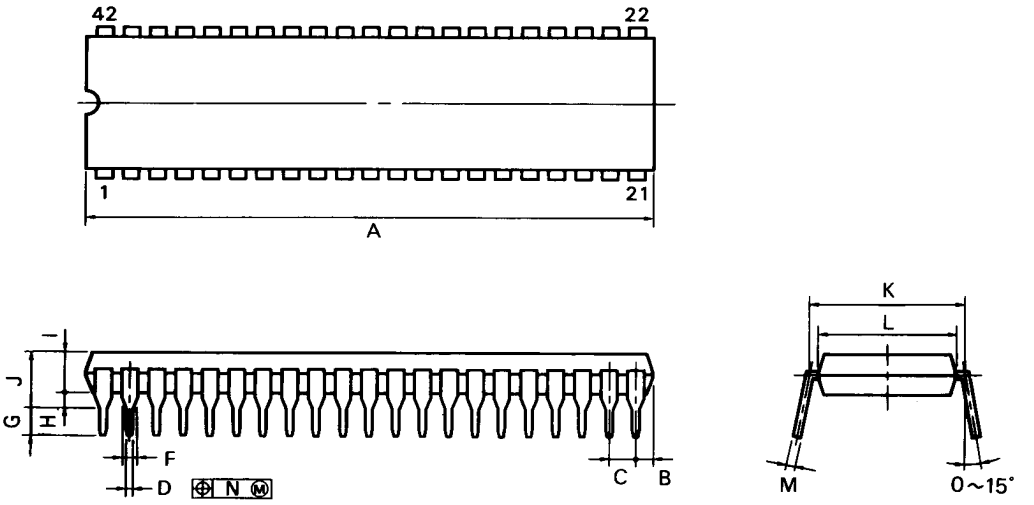
6.4 AC CHARACTERISTICS (V_{DD} = +4.5 to 5.5 V, T_a = −35 to 75 °C)

CHARACTERISTICS	SYMBOL	MIN.	TYP.	MAX.	UNIT	CONDITION
Operating Frequency	f_{in1}	0.5		2.5	MHz	VCOL pin V_{in}=1.0 V_{p-p} (MIN.), DC cut
Operating Frequency	f_{in2}	0.5		8.8	MHz	VCOH pin V_{in}=0.8 V_{p-p} (MIN.), DC cut

This Material Copyrighted By Its Respective Manufacturer

7. PACKAGE DIMENSIONS (UNIT: mm)

42 pin plastic DIP (600 mil)



P42C-100-600A,B

NOTES

1) Each lead centerline is located within 0.25 mm (0.01 inch) of its true position (T.P.) at maximum material condition.

2) Item "K" to center of leads when formed parallel.

ITEM	MILLIMETERS	INCHES
A	55.88 MAX.	2.200 MAX.
B	2.54 MAX.	0.100 MAX.
C	2.54 (T.P.)	0.100 (T.P.)
D	$0.50^{\pm 0.10}$	$0.020^{+0.004}_{-0.005}$
F	1.2 MIN.	0.047 MIN.
G	$3.6^{\pm 0.3}$	$0.142^{\pm 0.012}$
H	0.51 MIN.	0.020 MIN.
I	4.31 MAX.	0.170 MAX.
J	5.72 MAX.	0.226 MAX.
K	15.24 (T.P.)	0.600 (T.P.)
L	13.2	0.520
M	$0.25^{+0.10}_{-0.05}$	$0.010^{+0.004}_{-0.003}$
N	0.25	0.01

This Material Copyrighted By Its Respective Manufacturer

8. SUPPORT TOOLS

The following support tolls are available for developing systems using the μPD1704.

Hardware	EVAKIT-1700		This is an evaluation board usable in common in the μPD1700 series. It is operated on PROM base and it is possible that immediately adding and revising the program through the console.		
	SE-1704		This is a simulation board in which the program developed by EVAKIT-1700 is loaded and provided in a system instead of the μPD1704 to evaluate the system. SE-1704 is possible to evaluate the μPD1710 also.		
Software	μPD1700 Series Assembler	Absolute Assembler	Host	OS	Order name
			PDA-880	CP/M-80™	μS281AS1700
			PDA-800 + PDA-800FDD		
			MD-080 series		
			MD-086 series	MP/M-86™	μS171AS1700

Remarks 1. Supply format of software is 8-inch FD double-side double density.

2. Regarding CP/M version of OS of PDA-800 and PDA-880 systems, please contact our authorized agent.

3. CP/M-80™ and MP/M-86™ are the trademark of Digital Research Co., Ltd.

IC—1824
September 1986P
Printed in Japan

013718

This Material Copyrighted By Its Respective Manufacturer