NEC

MOS INTEGRATED CIRCUIT
μPD6133, 6134

4-BIT SINGLE-CHIP MICROCONTROLLER
FOR INFRARED REMOTE CONTROL TRANSMISSION

DESCRIPTION

Equipped with low-voltage 1.8V operation, a carrier generation circuit for infrared remote control transmission, a standby release function through key entry, and a programmable timer, the μPD6133 and 6134 are suitable for infrared remote control transmitters.

For the μPD6133 and 6134, we have made available the one-time PROM product μPD61P34B for program evaluation or small-quantity production.

FEATURES

- Program memory (ROM)
  - · μPD6133: 512 × 10 bits
  - · μPD6134: 1002 × 10 bits
- Data memory (RAM): 32 × 4 bits
- Built-in carrier generation circuit for infrared remote control
- 9-bit programmable timer : 1 channel
- Command execution time : 8 μs (when operating at f_X = 1 MHz: ceramic oscillation)
- Stack level : 1 level (Stack RAM is for data memory RF as well.)
- I/O pins ($K_{I/O}$) : 8 units
- Input pins (K_I) : 4 units
- Sense input pin (S_0) : 1 unit
- $S_1/\overline{LED}$ pin (I/O) : 1 unit (When in output mode, this is the remore control transmission display pin.)
- Power supply voltage : V_{DD} = 1.8 to 3.6 V (when operating at f_X = 500 kHz)
  V_{DD} = 2.2 to 3.6 V (when operating at f_X = 1 MHz)
- Operating ambient temperature : T_A = –40 to +85 ˚C
- Oscillation frequency : f_X = 300 kHz to 1 MHz
- POC circuit (Mask option)

APPLICATION

Infrared remote control transmitter (for AV and household electric appliances)

Unless otherwise stated, the μPD6133 is taken as a representative product in this document.

Document No. U10454EJ6V0DS00 (6th edition)
Date Published June 1999 N CP(K)
Printed in Japan

The mark ★ shows major revised points.

ORDERING INFORMATION

	Part Number	Package
	μPD6133GS-×××	20-pin plastic SOP (300 mil)
	μPD6134GS-×××	20-pin plastic SOP (300 mil)
★	μPD6134MC-×××-5A4	20-pin plastic SSOP (300 mil)

Remark ××× indicates ROM code suffix.

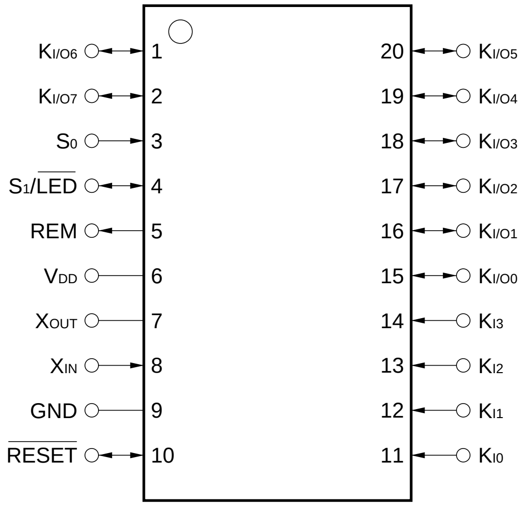
PIN CONFIGURATION (TOP VIEW)

20-pin Plastic SOP (300 mil)

- μ**PD6133GS-**×××
- μ**PD6134GS-**×××

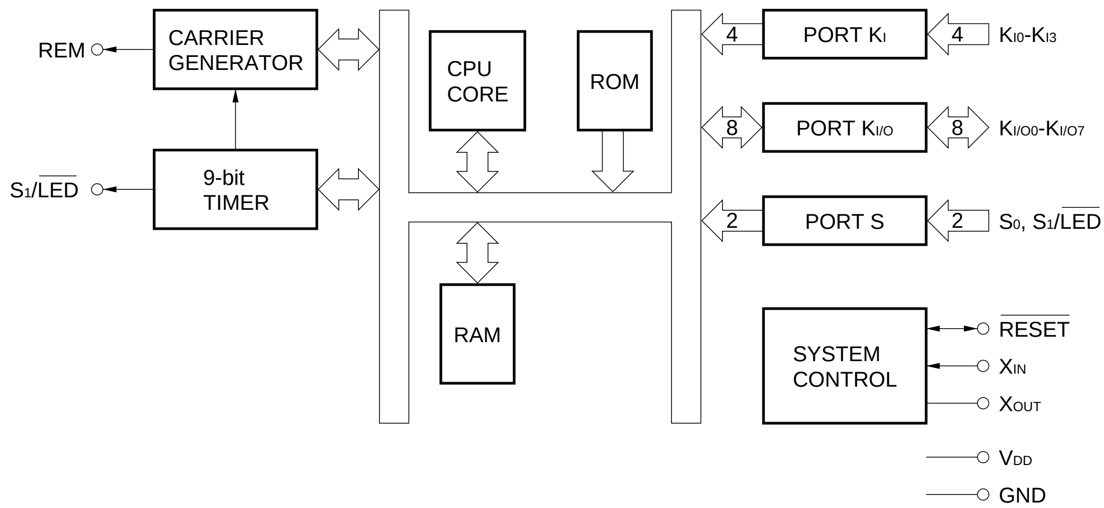
★ **20-pin Plastic SSOP (300 mil)**

- μ**PD6134MC-**×××**-5A4**



Caution The pin numbers of K_I and $K_{I/O}$ are in the reverse order of the μPD6600A and 6124A.

BLOCK DIAGRAM



LIST OF FUNCTIONS

Item	μPD6133	μPD6134	μPD61P34B
ROM capacity	512 × 10 bit	1002 × 10 bit	1002 × 10 bits
	Mask ROM		One-time PROM
RAM capacity	32 × 4 bits		
Stack	1 level (multiplexed with RF of RAM)		
I/O pins	• Key input (K_I) : 4 • Key I/O ($K_{I/O}$) : 8 • Key extended input (S_0, S_1) : 2 • Remote control transmission display output ($\overline{LED}$) : 1 (multiplexed with S_1 pin)		
Number of keys	• 32 keys • 48 keys (when extended by key extension input) • 96 keys (when extended by key extension input and diode)		
Clock frequency	Ceramic oscillation • f_X = 300 kHz to 1 MHz • f_X = 300 to 500 kHz (with POC circuit)		
Instruction execution time	8 μs (fx = 1 MHz)		
Carrier frequency	f_X, $f_X/2$, $f_X/8$, $f_X/12$, $f_X/16$, $f_X/24$, no carrier (high level)		
Timer	9-bit programmable timer: 1 channel		
POC circuit	Mask option		Internal
★ Supply voltage	V_{DD} = 1.8 to 3.6 V		V_{DD} = 2.2 to 3.6 V
Operating ambient temperature	• T_A = –40 to +85 °C • T_A = –20 to +70 °C (with POC circuit)		
★ Package	• 20-pin plastic SOP (300 mil)	• 20-pin plastic SOP (300 mil) • 20-pin plastic SSOP (300 mil)	

TABLE OF CONTENTS

1. PIN FUNCTIONS

1.1 List of Pin Functions

Pin No.	Symbol	Function	Output Format	When Reset
1 2 15-20	$K_{I/O0}$-$K_{I/O7}$	These pins refer to the 8-bit I/O ports. I/O switching can be made in 8-bit units. In INPUT mode, a pull-down resistor is added. In OUTPUT mode, they can be used as the key scan output of the key matrix.	CMOS push-pull[Note 1]	High-level output
3	S_0	Refers to the input port. Can also be used as the key return input of the key matrix. In INPUT mode, the availability of the pull-down resistor of the S_0 and S_1 ports can be specified by software in terms in 2-bit units. If INPUT mode is canceled by software, this pin is placed in OFF mode and enters the high-impedance state.	—	High-impedance (OFF mode)
4	$S_1/\overline{LED}$	Refers to the I/O port. In INPUT mode (S_1), this pin can also be used as the key return input of the key matrix. The availability of the pull-down resistor of the S_0 and S_1 ports can be specified by software in 2-bit units. In OUTPUT mode ($\overline{LED}$), it becomes the remote control transmission display output (active low). When the remote control carrier is output from the REM output, this pin outputs the low level from the $\overline{LED}$ output synchronously with the REM signal.	CMOS push-pull	High-level output ($\overline{LED}$)
5	REM	Refers to the infrared remote control transmission output. The output is active high. Carrier frequency: f_X, $f_X/8$, $f_X/12$, high-level, $f_X/2$, $f_X/16$, $f_X/24$ (usable on software)	CMOS push-pull	Low-level output
6	V_{DD}	Refers to the power supply.	—	—
7 8	X_{OUT} X_{IN}	These pins are connected to system clock ceramic resonators.	—	Low level (oscillation stopped)
9	GND	Refers to the ground.	—	—
10	$\overline{RESET}$	Normally, this pin is a system reset input. By inputting a low level, the CPU can be reset. When resetting with the POC circuit (mask option) a low level is output. A pull-up resistor is incorporated.	—	—
11-14	K_{I0}-K_{I3}[Note 2]	These pins refer to the 4-bit input ports. They can be used as the key return input of the key matrix. The use of the pull-down resistor can be specified by software in 4-bit units.	—	Input (low-level)

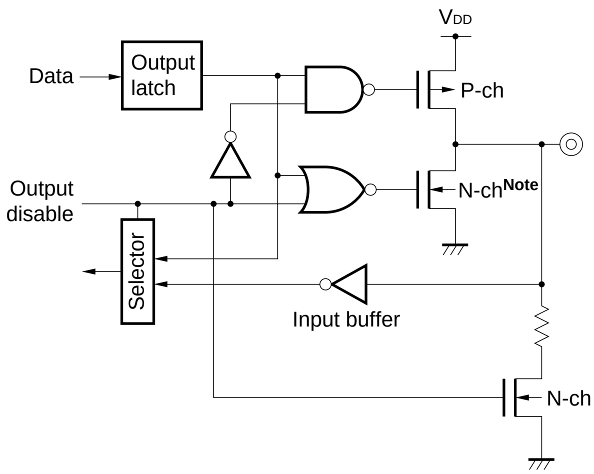
Notes 1. Be careful about this because the drive capability of the low-level output side is held low.

2. In order to prevent malfunction, be sure to input a low level to more than one of pins K_{I0} to K_{I3} when reset is released (when $\overline{RESET}$ pin changes from low level to high level, or POC is released due to supply voltage startup).

1.2 INPUT/OUTPUT Circuits of Pins

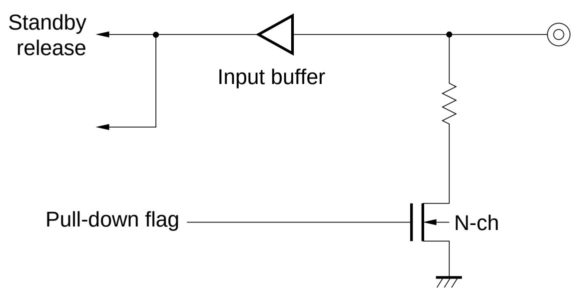
The input/output circuits of the μPD6133 pins are shown in partially simplified forms below.

(1) $K_{I/O0}$-$K_{I/O7}$

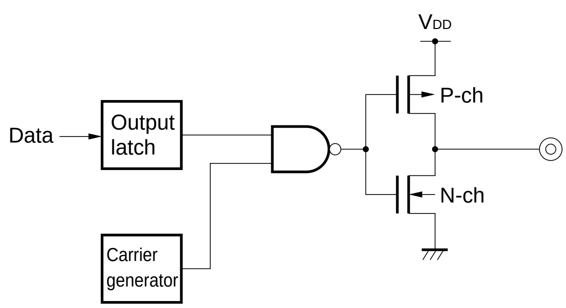


Note The drive capability is held low.

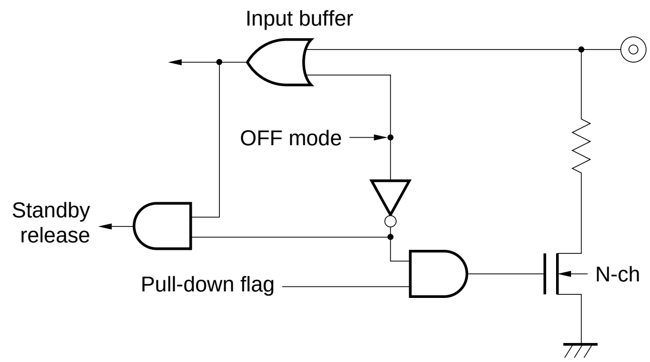
(2) K_{I0}-K_{I3}



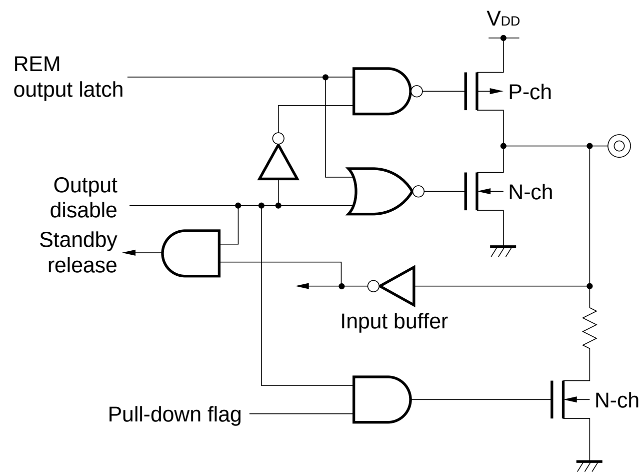
(3) REM



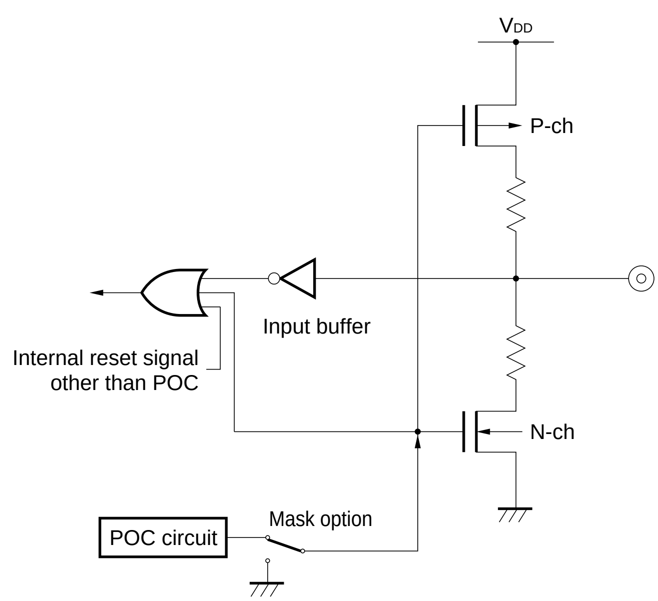
(4) S_0



(5) S_1/$\overline{LED}$



(6) $\overline{RESET}$



1.3 Dealing with Unused Pins

The following connections are recommended for unused pins.

Table 1-1. Connections for Unused Pins

Pin		Connection	
		Inside the microcontroller	Outside the microcontroller
$K_{I/O}$	INPUT mode	—	Open
	OUTPUT mode	High-level output	
REM		—	
$S_1/\overline{LED}$		OUTPUT mode ($\overline{LED}$) setting	
S_0		OFF mode setting	Directly connected to GND
K_I		—	
$\overline{RESET}$[Note]		Built-in POC circuit	Open

Note If the circuit is an applied one requiring high reliability, be sure to design it in such a manner that the $\overline{RESET}$ signal is entered externally.

Caution The I/O mode and the terminal output level are recommended to be fixed by setting them repeatedly in each loop of the program.

2. INTERNAL CPU FUNCTIONS

2.1 Program Counter (PC): 10 Bits

Refers to the binary counter that holds the address information of the program memory.

Figure 2-1. Program Counter Organization

PC	PC9	PC8	PC7	PC6	PC5	PC4	PC3	PC2	PC1	PC0

The program counter contains the address of the instruction that should be executed next. Normally, the counter contents are automatically incremented in accordance with the instruction length (byte count) each time an instruction is executed.

However, when executing JUMP instructions (JMP, JC, JNC, JF, JNF), the program counter contains the jump destination address written in the operand.

When executing the subroutine call instruction (CALL), the call destination address written in the operand is entered in the PC after the PC contents at the time are saved in the address stack register (ASR). If the return instruction (RET) is executed after the CALL instruction is executed, the address saved in the ASR is restored to the PC.

When reset, the value of the program counter becomes “000H”.

2.2 Stack Pointer (SP): 1 Bit

Refers to the 1-bit register which holds the status of the address stack register.

The stack pointer contents are incremented when the call instruction (CALL) is executed; they are decremented when the return instruction (RET) is executed.

When reset, the stack pointer contents are cleared to “0”.

When the stack pointer overflows (stack level 2 or more) or underflows, the CPU is hung up thus a system reset signal is generated and the PC becoming “000H”.

As no instruction is available to set a value directly for the stack pointer, it is not possible to operate the pointer by means of a program.

2.3 Address Stack Register (ASR (RF)): 10 Bits

The address stack register saves the return address of the program after a subroutine call instruction is executed.

The low-order 8 bits are arranged in the RF of the data memory as a dual-function RAM. The register holds the ASR value even after the RET is executed.

When reset, it holds the previous data (undefined when turning on the power).

Caution If the RF is accessed as the data memory, the high-order 2 bits of the ASR become undefined.

Figure 2-2. Address Stack Register Organization

			RF							
ASR	ASR9	ASR8	ASR7	ASR6	ASR5	ASR4	ASR3	ASR2	ASR1	ASR0

(RF spans ASR7 to ASR0.)

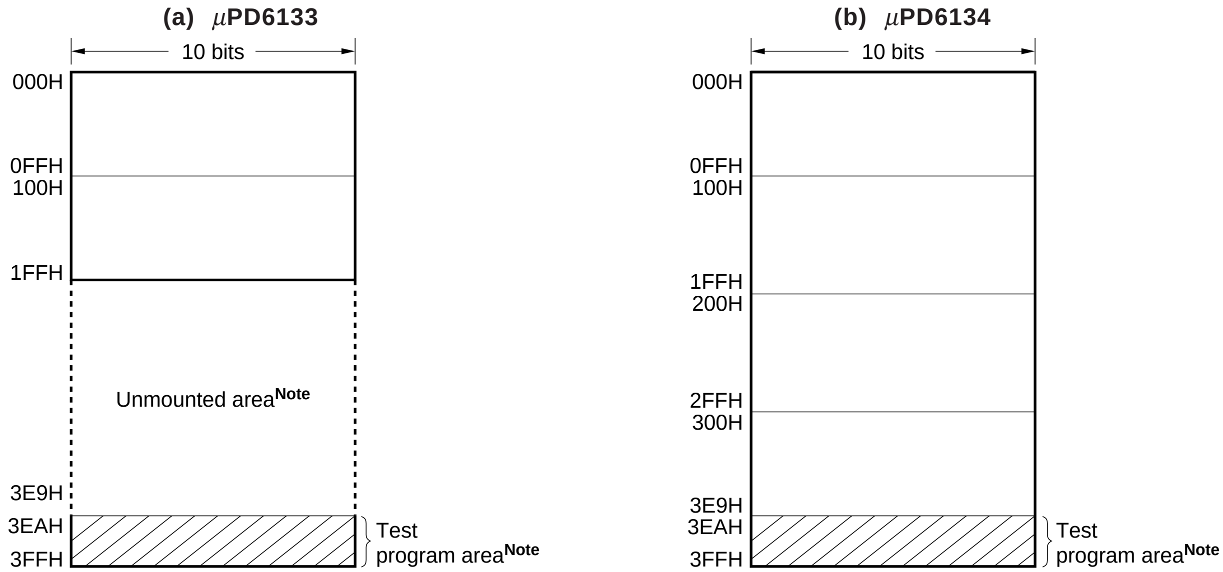
2.4 Program Memory (ROM): 512 steps × 10 bits (μPD6133)
1002 steps × 10 bits (μPD6134)

The ROM consists of 10 bits per step, and is addressed by the program counter.

The program memory stores programs and table data, etc.

The 22 steps from 3EAH to 3FFH cannot be used in the test program area.

Figure 2-3. Program Memory Map



Note The unmounted area and the test program area are so designed that a program or data placed in either of them by mistake is returned to the 000H address.

2.5 Data Memory (RAM): 32 × 4 Bits

The data memory, which is a static RAM consisting of 32 × 4 bits, is used to retain processed data. The data memory is sometimes processed in 8-bit units. R0 can be used as the ROM data pointer.

RF is also used as the ASR.

When reset, R0 is cleared to “00H” and R1 to RF retain the previous data (undefined when turning on the power).

Figure 2-4. Data Memory Organization

Register	R_{1n} (high-order 4 bits)	R_{0n} (low-order 4 bits)	
R0	R_{10}	R_{00}	→DP (Refer to **2.6 Data Pointer**.)
R1	R_{11}	R_{01}	
R2	R_{12}	R_{02}	
R3	R_{13}	R_{03}	
R4	R_{14}	R_{04}	
R5	R_{15}	R_{05}	
R6	R_{16}	R_{06}	
R7	R_{17}	R_{07}	
R8	R_{18}	R_{08}	
R9	R_{19}	R_{09}	
RA	R_{1A}	R_{0A}	
RB	R_{1B}	R_{0B}	
RC	R_{1C}	R_{0C}	
RD	R_{1D}	R_{0D}	
RE	R_{1E}	R_{0E}	
RF	R_{1F}	R_{0F}	→ASR (Refer to **2.3 Address Stack Register**.)

2.6 Data Pointer (DP): 10 Bits

The ROM data table can be referenced by setting the ROM address in the data pointer to call the ROM contents.

The low-order 8 bits of the ROM address are specified by R0 of the data memory; and the high-order 2 bits by bits 4 and 5 of the P3 register (CR0).

When reset, the pointer contents become “000H”.

Figure 2-5. Data Pointer Organization

	P3 Register b_5	P3 Register b_4	R_{10}				R_{00}				
P3	DP_9	DP_8	DP_7	DP_6	DP_5	DP_4	DP_3	DP_2	DP_1	DP_0	R0

2.7 Accumulator (A): 4 Bits

The accumulator, which refers to a register consisting of 4 bits, plays a leading role in performing various operations.

When reset, the accumulator contents are left undefined.

Figure 2-6. Accumulator Organization

A_3	A_2	A_1	A_0	A

2.8 Arithmetic and Logic Unit (ALU): 4 Bits

The arithmetic and logic unit (ALU), which refers to an arithmetic circuit consisting of 4 bits, executes simple manipulations with priority given to logical operations.

2.9 Flags

2.9.1 Status flag (F)

Pin and timer statuses can be checked by executing the STTS instruction to check the status flag.

The status flag is set (to 1) in the following cases.

- If the condition specified with the operand is met when the STTS instruction has been executed
- When STANDBY mode is canceled.
- When the cancelation condition is met at the point of executing the HALT instruction. (In this case, the system is not placed in STANDBY mode.)

Conversely, the status flag is cleared (to 0) in the following cases:

- If the condition specified with the operand is not met when the STTS instruction has been executed.
- When the status flag has been set (to 1), the HALT instruction executed, but the cancelation condition is not met at the point of executing the HALT instruction. (In this case, the system is not placed in STANDBY mode.)

Table 2-1. Conditions for Status Flag (F) to be Set by STTS Instruction

Operand Value of STTS Instruction				Condition for Status Flag (F) to be Set
b_3	b_2	b_1	b_0	
0	0	0	0	High level is input to at least one of K_I pins.
	0	1	1	High level is input to at least one of K_I pins.
	1	1	0	High level is input to at least one of K_I pins.
	1	0	1	The down counter of the timer is 0.
1	Either of the combinations of b_2, b_1, and b_0 above.			[The following condition is added in addition to the above.] High level is input to at least one of S_0 and S_1 pins.

2.9.2 Carry flag (CY)

The carry flag is set (to 1) in the following cases:

- If the ANL instruction or the XRL instruction is executed when bit 3 of the accumulator is “1” and bit 3 of the operand is “1”.
- If the RL instruction or the RLZ instruction is executed when bit 3 of the accumulator is “1”.
- If the INC instruction or the SCAF instruction is executed when the value of the accumulator is 0FH.

The carry flag is cleared (to 0) in the following cases:

- If the ANL instruction or the XRL instruction is executed when at least either bit 3 of the accumulator or bit 3 of the operand is “0”.
- If the RL instruction or the RLZ instruction is executed when bit 3 of the accumulator is “0”.
- If the INC instruction or the SCAF instruction is executed when the value of the accumulator is other than 0FH.
- If the ORL instruction is executed.
- When Data is written to the accumulator by the MOV instruction or the IN instruction.

3. PORT REGISTERS (PX)

The $K_{I/O}$ port, the K_I port, the special ports (S_0, $S_1/\overline{LED}$), and the control register are treated as port registers. At reset, port register values are shown below.

Figure 3-1. Port Register Organization

Port Register								At Reset
P0								FFH
P_{10}				P_{00}				
$K_{I/O7}$	$K_{I/O6}$	$K_{I/O5}$	$K_{I/O4}$	$K_{I/O3}$	$K_{I/O2}$	$K_{I/O1}$	$K_{I/O0}$	
P1								×FH[Note]
P_{11}				P_{01}				
K_{I3}	K_{I2}	K_{I1}	K_{I0}	$S_1/\overline{LED}$	S_0	1	1	
P3 (Control register 0)								03H
P_{13}				P_{03}				
0	0	DP_9	DP_8	TCTL	CARY	MOD_1	MOD_0	
P4 (Control register 1)								26H
P_{14}				P_{04}				
0	0	K_I pull-down	S_0/S_1 pull-down	0	$S_1/\overline{LED}$mode	$K_{I/O}$ mode	S_0 mode	

Note ×: Refers to the value based on the K_I pin state.

Table 3-1. Relationship between Ports and their Read/Write

Port Name	INPUT Mode		OUTPUT Mode	
	Read	Write	Read	Write
$K_{I/O}$	Pin state	Output latch	Output latch	Output latch
K_I	Pin state	—	—	—
S_0	Pin state	—	**Note**	—
$S_1/\overline{LED}$	Pin state	—	Pin state	—

Note When in OFF mode, “1” is normally read.

3.1 $K_{I/O}$ Port (P0)

The $K_{I/O}$ port is an 8-bit input/output port for key scan output.

INPUT/OUTPUT mode is set by bit 1 of the P4 register.

If a read instruction is executed, the pin state can be read in INPUT mode, whereas the output latch contents can be read in OUTPUT mode.

If the write instruction is executed, data can be written to the output latch regardless of INPUT or OUTPUT mode.

When reset, the port is placed in OUTPUT mode; and the value of the output latch (P0) becomes 1111 1111B.

The $K_{I/O}$ port contains the pull-down resistor, allowing pull-down in INPUT mode only.

Caution During double pressing of a key, a high-level output and a low-level output may coincide with each other at the $K_{I/O}$ port. To avoid this, the low-level output current of the $K_{I/O}$ port is held low. Therefore, be careful when using the $K_{I/O}$ port for purposes other than key scan output. The $K_{I/O}$ port is so designed that, even when connected directly to V_{DD} within the normal supply voltage range (V_{DD} = 1.8 to 3.6 V), no problem may occur.

Table 3-2. $K_{I/O}$ Port (P0)

Bit	b_7	b_6	b_5	b_4	b_3	b_2	b_1	b_0
Name	$K_{I/O7}$	$K_{I/O6}$	$K_{I/O5}$	$K_{I/O4}$	$K_{I/O3}$	$K_{I/O2}$	$K_{I/O1}$	$K_{I/O0}$

b_0-b_7 : In reading : In INPUT mode, the $K_{I/O}$ pin's state is read.
In OUTPUT mode, the $K_{I/O}$ pin's output latch contents are read.
In writing : Data is written to the $K_{I/O}$ pin's output latch regardless of INPUT or OUTPUT mode.

3.2 K_I Port/Special Ports (P1)

3.2.1 K_I port (P_{11}: bits 4-7 of P1)

The K_I port is to the 4-bit input port for key entry.

The pin state can be read.

Software can be used to set the availability of the pull-down resistor of the K_I port in 4-bit units by means of bit 5 of the P4 register.

When reset, the pull-down resistor is connected.

Table 3-3. K_I/Special Port Register (P1)

Bit	b_7	b_6	b_5	b_4	b_3	b_2	b_1	b_0
Name	K_{I3}	K_{I2}	K_{I1}	K_{I0}	$S_1/\overline{LED}$	S_0	(Fixed to "1")	

b_2 : In INPUT mode, state of the S_0 pin is read (Read only).
In OFF mode, this bit is fixed to "1".

b_3 : The state of the $S_1/\overline{LED}$ pin is read regardless of INPUT/OUTPUT mode (Read only).

b_4-b_7 : The state of the K_I pin is read (Read only).

Caution In order to prevent malfunction, be sure to input a low level to more than one of pins K_{I0} to K_{I3} when reset is released (when $\overline{RESET}$ pin changes from low level to high level, or POC is released due to supply voltage startup).

3.2.2 S_0 port (bit 2 of P1)

The S_0 port is the INPUT/OFF mode port.

The pin state can be read by setting this port to INPUT mode with bit 0 of the P4 register.

In INPUT mode, software can be used to set the availability of the pull-down resistor of the S_0 and $S_1/\overline{LED}$ port in 2-bit units by means of bit 4 of the P4 register.

If INPUT mode is canceled (thus set to OFF mode), the pin becomes high-impedance but it also makes that the through current does not flow internally. In OFF mode, "1" can be read regardless of the pin state.

When reset, it is set to OFF mode, thus becoming high-impedance.

3.2.3 $S_1/\overline{LED}$ (bit 3 of P1)

The $S_1/\overline{LED}$ port is the input/output port.

It uses bit 2 of the P4 register to set INPUT or OUTPUT mode. The pin state can be read in both INPUT mode and OUTPUT mode.

When in INPUT mode, software can be used to set the availability of the pull-down resistor of the S_0 and $S_1/\overline{LED}$ ports in 2-bit units by means of bit 4 of the P4 register.

When in OUTPUT mode, the pull-down resistor is automatically disconnected thus becoming the remote transmission display pin (refer to **4. TIMER**).

When reset, it is placed in OUTPUT mode, and high level is output.

3.3 Control Register 0 (P3)

Control register 0 consists of 8 bits. The contents that can be controlled are as shown below.
When reset, the register becomes 0000 0011B.

Table 3-4. Control Register 0 (P3)

Bit		b_7	b_6	b_5	b_4	b_3	b_2	b_1	b_0
Name		—	—	DP (Data pointer)		TCTL	CARY	MOD_1	MOD_0
				DP_9	DP_8				
Set value	0	Fixed to “0”	Fixed to “0”	0	0	1/1	ON	Refer to **Table 3-5**.	
	1			1	1	1/2	OFF		
When reset		0	0	0	0	0	0	1	1

b_0, b_1 : These bits specify the carrier frequency and duty ratio of the REM output.

b_2 : This bit specifies the availability of the carrier of the frequency specified by b_0 and b_1.
“0” = ON (with carrier); “1” = OFF (without carrier; high level)

b_3 : This bit changes the carrier frequency and the timer clock’s frequency division ratio.
“0” = 1/1 (carrier frequency: the specified value of b_0 and b_1; timer clock: $f_X/8$)
“1” = 1/2 (carrier frequency: half of the specified value of b_0 and b_1; timer clock: $f_X/16$)

Table 3-5. Timer Clock and Carrier Frequency Setup

b_3	b_2	b_1	b_0	Timer Clock	Carrier Frequency (Duty Ratio)
0	0	0	0	$f_X/8$	f_X (Duty 1/2)
		0	1		$f_X/8$ (Duty 1/2)
		1	0		$f_X/12$ (Duty 1/2)
		1	1		$f_X/12$ (Duty 1/3)
	1	×	×		Without carrier (high level)
1	0	0	0	$f_X/16$	$f_X/2$ (Duty 1/2)
		0	1		$f_X/16$ (Duty 1/2)
		1	0		$f_X/24$ (Duty 1/2)
		1	1		$f_X/24$ (Duty 1/3)
	1	×	×		Without carrier (high level)

b_4 and b_5 : These bits specify the high-order 2 bits (DP_8 and DP_9) of ROM’s data pointer.

Remark ×: don’t care

3.4 Control Register 1 (P4)

Control register 1 consists of 8 bits. The contents that can be controlled are as shown below.
When reset, the register becomes 0010 0110B.

Table 3-6. Control Register 1 (P4)

Bit		b_7	b_6	b_5	b_4	b_3	b_2	b_1	b_0
Name		—	—	K_I Pull-down	S_0/S_1 Pull-down	—	$S_1/\overline{LED}$ mode	$K_{I/O}$ mode	S_0 mode
Set value	0	Fixed to “0”	Fixed to “0”	OFF	OFF	Fixed to “0”	S_1	IN	OFF
	1			ON	ON		$\overline{LED}$	OUT	IN
When reset		0	0	1	0	0	1	1	0

b_0 : Specifies the input mode of the S_0 port. “0” = OFF mode (high impedance); “1” = IN (INPUT mode).

b_1 : Specifies the I/O mode of the $K_{I/O}$ port.
“0” = IN (INPUT mode); “1” = OUT (OUTPUT mode).

b_2 : Specifies the I/O mode of the $S_1/\overline{LED}$ port. “0” = S_1 (INPUT mode); “1” = $\overline{LED}$ (output mode).

b_4 : Specifies the availability of the pull-down resistor in S_0/S_1 port INPUT mode. “0” = OFF (unavailable); “1” = ON (available)

b_5 : Specifies the availability of the pull-down resistor in K_I port. “0” = OFF (unavailable); “1” = ON (available).

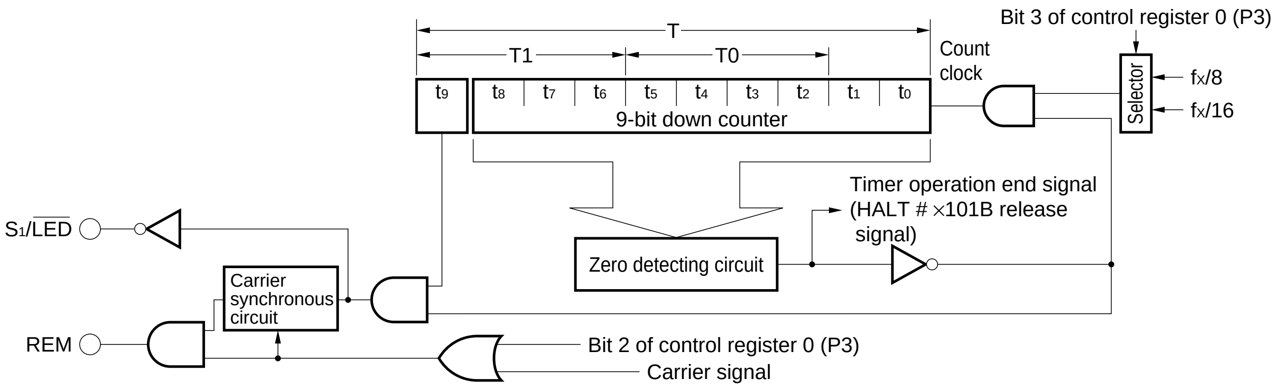
Remark In OUTPUT mode or in OFF mode, all the pull-down resistors are automatically disconnected.

4. TIMER

4.1 Timer Configuration

The timer is the block used for creating a remote control transmission pattern. As shown in Figure 4-1, it consists of a 9-bit down counter (t_8 to t_0), a flag (t_9) permitting the 1-bit timer output, and a zero detecting circuit.

Figure 4-1. Timer Configuration



4.2 Timer Operation

The timer starts (counting down) when a value other than 0 is set for the down counter with a timer operation instruction. The timer operation instructions for making the timer start operation are shown below:

```
MOV T0, A
MOV T1, A
MOV T, #data10
MOV T, @R0
```

The down counter is decremented (–1) in the cycle of $8/f_X$ or $16/f_X$ Note. If the value of the down counter becomes 0, the zero detecting circuit generates the timer operation end signal to stop the timer operation. At this time, if the timer is in HALT mode (HALT # × 101B) waiting for the timer to stop its operation, the HALT mode is canceled and the instruction following the HALT instruction is executed. The output of the timer operation end signal is continued while the down counter is 0 and the timer is stopped. There is the following relational expression between the timer's time and the down counter's set value.

Timer time = (Set value + 1) × $8/f_X$ (or $16/f_X$ Note)

Note This becomes $16/f_X$ if bit 3 of the control register is set (to 1).

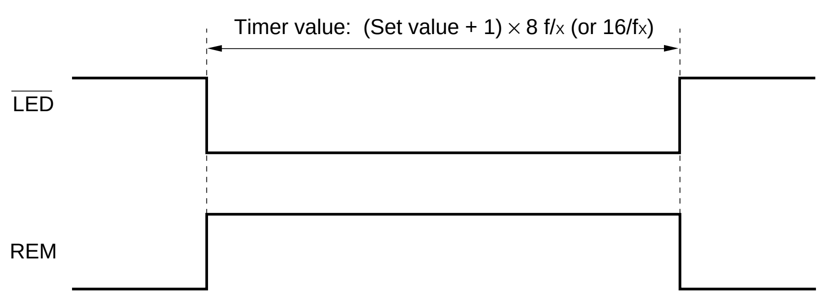
By setting 1 for the flag (t_9) which enables the timer output, the timer can output its operation status from the $S_1/\overline{LED}$ pin and the REM pin. The REM pin can also output the carrier while the timer is in operation.

Table 4-1. Timer Output (at t_9 = 1)

	$S_1/\overline{LED}$ Pin	REM Pin
Timer operating	L	H (or carrier output Note)
Timer halting	H	L

Note The carrier output results if bit 2 of the control register 0 is cleared (to 0).

Figure 4-2. Timer Output (When Carrier Is Not Output)

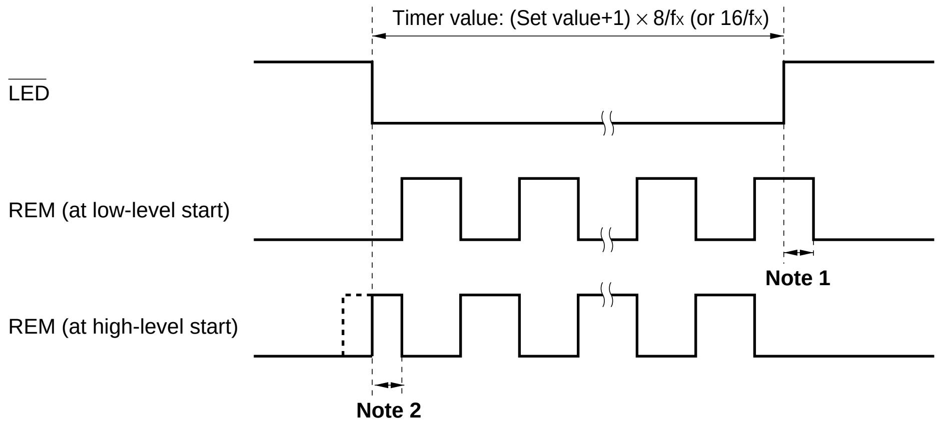


4.3 Carrier Output

The carrier for remote-controlled transmission can be output from the REM pin by clearing (to 0) bit 2 of the control register 0.

As shown in Figure 4-3, in the case where the timer stops when the carrier is at a high level, the carrier continues to be output until its next fall and then stops due to the function of the carrier synchronous circuit . When the timer starts operation, however, the high-level width of the first carrier may become shorter than the specified width.

Figure 4-3. Timer Output (When Carrier Is Output)



Notes 1. Error when the REM output ends: Lead by "the carrier's low-level width" to lag by "the carrier's high-level width"

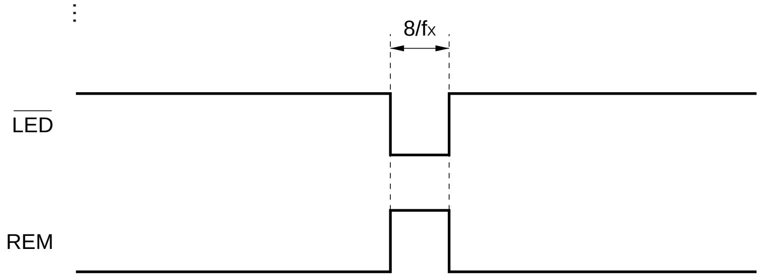
2. Error of the carrier's high-level width: 0 to "the carrier's high-level width"

4.4 Software Control of Timer Output

The timer output can be controlled by software. As shown in Figure 4-4, the pulse with a minimum width of 1-instruction cycle (8/fx) can be output.

Figure 4-4. Pulse Output of 1-Instruction Cycle Width

```
⋮
MOV T, #0000000000B; low-level output from the REM pin
⋮
MOV T, #1000000000B; high-level output from the REM pin
MOV T, #0000000000B; low-level output from the REM pin
⋮
```



5. STANDBY FUNCTION

5.1 Outline of Standby Function

To save current consumption, two types of standby modes, i.e., HALT mode and STOP mode, are made available.

In STOP mode, the system clock stops oscillation. At this time, the X_{IN} and X_{OUT} pins are fixed at a low level.

In HALT mode, CPU operation halts, while the system clock continues oscillation. When in HALT mode, the timer (including REM output and $\overline{LED}$ output) operates.

In either STOP mode or HALT mode, the statuses of the data memory, accumulator, and port register, etc. immediately before the standby mode is set are retained. Therefore, make sure to set the port status for the system so that the current consumption of the whole system is suppressed before the standby mode is set.

Table 5-1. Statuses During Standby Mode

			STOP Mode	HALT Mode
Setting instruction			HALT instruction	
Clock oscillator			Oscillation stopped	Oscillation continued
Operation statuses	CPU		• Operation halted	
	Data memory		• Immediately preceding status retained	
	Accumulator		• Immediately preceding status retained	
	Flag	F	• 0 (When 1, the flag is not placed in the standby mode.)	
		CY	• Immediately preceding status retained	
	Port register		• Immediately preceding status retained	
	Timer		• Operation halted (The count value is reset to “0”)	• Operable

Cautions 1. Write the NOP instruction as the first instruction after STOP mode is canceled.

2. When standby mode is canceled, the status flag (F) is set (to 1).

3. If, at the point the standby mode has been set, its cancelation condition is met, then the system is not placed in the standby mode. However, the status flag (F) is set (1).

5.2 Standby Mode Setup and Cancelation

The standby mode is set with the HALT #$b_3b_2b_1b_0$B instruction for both STOP mode and HALT mode. For the standby mode to be set, the status flag (F) is required to have been cleared (to 0).

The standby mode is canceled by the cancelation condition specified with the $\overline{\text{RESET}}$ ($\overline{\text{RESET}}$ input; POC) or the operand of HALT instruction. If the standby mode is canceled, the status flag (F) is set (to 1).

Even when the HALT instruction is executed in the state that the status flag (F) has been set (to 1), the standby mode is not set. If the cancelation condition is not met at this time, the status flag is cleared (to 0). If the cancelation condition is met, the status flag remains set (to 1).

Even in the case when the cancelation condition has been already met at the point that the HALT instruction is executed, the standby mode is not set. Here, also, the status flag (F) is set (to 1).

Caution Depending on the status of the status flag (F), the HALT instruction may not be executed. Be careful about this. For example, when setting HALT mode after checking the key status with the STTS instruction, the system does not enter HALT mode as long as the status flag (F) remains set (to 1) thus sometimes performing an unintended operation. In this case, the intended operation can be realized by executing the STTS instruction immediately after timer setting to clear (to 0) the status flag.

Example

```
STTS    #03H      ;To check the KI pin status.
   :
MOV     T, #0xxH  ;To set the timer
STTS    #05H      ;To clear the status flag
   :    (During this time, be sure not to execute an instruction that may set the status flag.)
HALT    #05H      ;To set HALT mode
```

Table 5-2. Addresses Executed After Standby Mode Release

Cancelation Condition	Address Executed After Cancelation
Reset	0 address
Cancelation condition shown in Table 5-3	The address following the HALT instruction

Table 5-3. Standby Mode Setup (HALT #$b_3b_2b_1b_0$B) and Release Conditions

Operand Value of HALT Instruction				Setting Mode	Precondition for Setup	Release Condition
b_3	b_2	b_1	b_0			
0	0	0	0	STOP	All $K_{I/O}$ pins are high-level output.	High level is input to at least one of K_I pins.
	0	1	1	STOP	All $K_{I/O}$ pins are high-level output.	High level is input to at least one of K_I pins.
	1	1	0	STOP Note 1	The $K_{I/O0}$ pin is high-level output.	High level is input to at least one of K_I pins.
1	Any of the combinations of $b_2b_1b_0$ above			STOP or HALT	[The following condition is added in addition to the above.] —	High level is input to at least one of S_0 and S_1 pins Note 2.
0/1	1	0	1	HALT	—	When the timer's down counter is 0

Notes 1. When setting HALT #×110B, configure a key matrix by using the $K_{I/O0}$ pin and the K_I pin so that an internal reset takes effect at the time of program hang-up.

2. At least one of the S_0 and S_1 pins (the pin used for releasing the standby) must be in INPUT mode. (The internal reset does not take effect even when both pins are in OUTPUT mode.)

Cautions 1. The internal reset takes effect when the HALT instruction is executed with an operand value other than that above or when the precondition has not been satisfied when executing the HALT instruction.

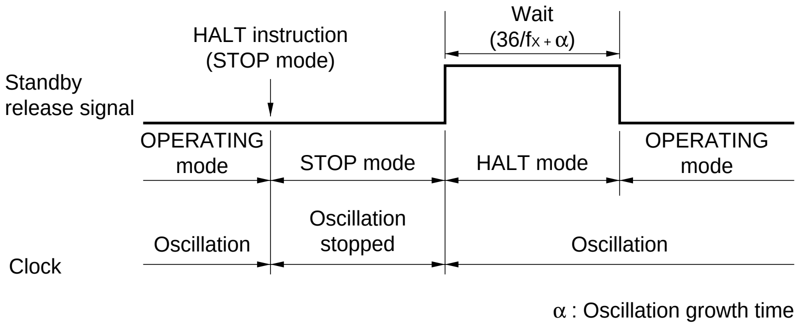
2. If STOP mode is set when the timer's down counter is not 0 (timer operating), the system is placed in STOP mode only after all the 10 bits of the timer's down counter and the timer output permit flag are cleared to 0.

3. Write the NOP instruction as the first instruction after STOP mode is released.

5.3 Standby Mode Release Timing

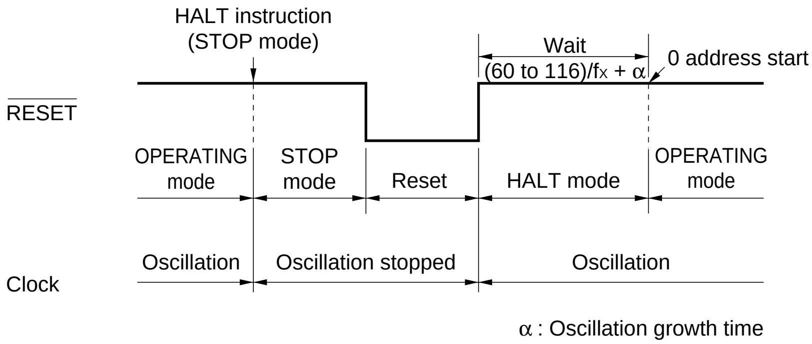
(1) STOP Mode Release Timing

Figure 5-1. STOP Mode Cancelation by Release Condition



Caution When a release condition is established in the STOP mode, the device is released from the STOP mode and goes into a wait status. At this time, if the release condition is not held, the device mode and goes into the STOP mode again after the wait time has elapsed. Therefore, when releasing the STOP mode, it is necessary to hold the release condition longer than the wait time.

Figure 5-2. STOP Mode Release by $\overline{\text{RESET}}$ Input



(2) HALT Mode Release Timing

Figure 5-3. HALT Mode Release by Cancelation Condition

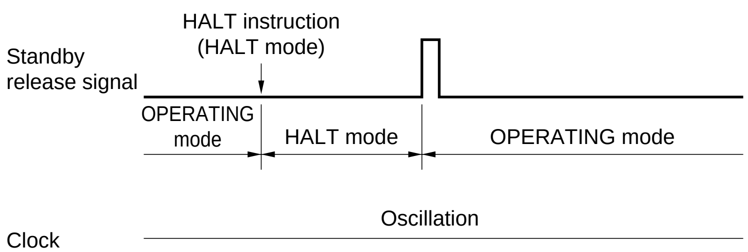
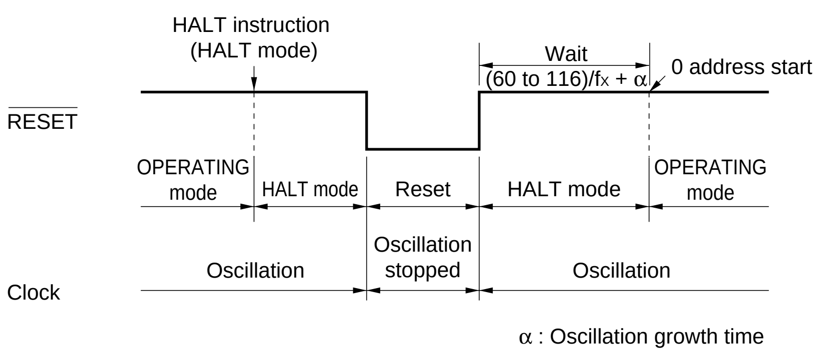


Figure 5-4. HALT Mode Release by $\overline{\text{RESET}}$ Input



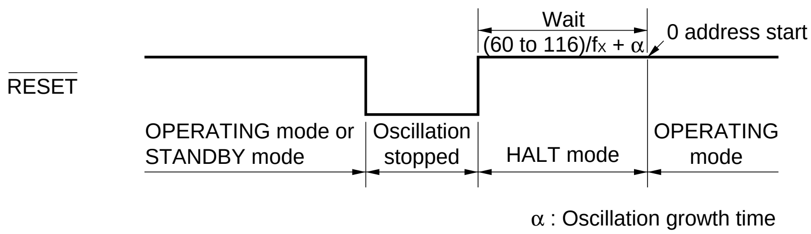
6. $\overline{\text{RESET}}$ PIN

The system reset takes effect by inputting low level to the $\overline{\text{RESET}}$ pin.

While the $\overline{\text{RESET}}$ pin is at low level, the system clock oscillator is stopped and the X_{IN} and X_{OUT} pins are fixed to the GND.

If the $\overline{\text{RESET}}$ pin is raised from low level to high level, it executes the program from the 0 address after counting 60 to 116 of the system clock (f_X).

Figure 6-1. Reset Operation by $\overline{\text{RESET}}$ Input



The $\overline{\text{RESET}}$ pin outputs low level when the POC circuit (mask option) is in operation.

Caution When connecting a reset IC to the $\overline{\text{RESET}}$ pin, ensure that the IC is of the N-ch open drain output type.

Table 6-1. Hardware Statuses After Reset

<table>
<tr><th colspan="2">Hardware</th><th>• $\overline{\text{RESET}}$ Input in Operation
• Resetting by Internal POC Circuit in Operation
• Resetting by Other Factors[Note 1]</th><th>• $\overline{\text{RESET}}$ Input During STANDBY Mode
• Resetting by the Internal POC Circuit During STANDBY Mode</th></tr>
<tr><td colspan="2">PC (10 bits)</td><td colspan="2">000H</td></tr>
<tr><td colspan="2">SP (1 bit)</td><td colspan="2">0B</td></tr>
<tr><td rowspan="2">Data memory</td><td>R0 = DP</td><td colspan="2">000H</td></tr>
<tr><td>R1-RF</td><td>Undefined</td><td>Previous status retained</td></tr>
<tr><td colspan="2">Accumulator (A)</td><td colspan="2">Undefined</td></tr>
<tr><td colspan="2">Status flag (F)</td><td colspan="2">0B</td></tr>
<tr><td colspan="2">Carry flag (CY)</td><td colspan="2">0B</td></tr>
<tr><td colspan="2">Timer (10 bits)</td><td colspan="2">000H</td></tr>
<tr><td rowspan="2">Port register</td><td>P0</td><td colspan="2">FFH</td></tr>
<tr><td>P1</td><td colspan="2">×FH[Note 2]</td></tr>
<tr><td rowspan="2">Control register</td><td>P3</td><td colspan="2">03H</td></tr>
<tr><td>P4</td><td colspan="2">26H</td></tr>
</table>

Notes 1. The following resets are available.
- Reset when executing the HALT instruction (when the operand value is illegal or does not satisfy the precondition)
- Reset when executing the RLZ instruction (when A = 0)
- Reset by stack pointer's overflow or underflow

2. Refers to the value by the K_I pin status.
In order to prevent malfunction, be sure to input a low level to more than one of pins K_{I0} to K_{I3} when reset is released (when $\overline{\text{RESET}}$ pin changes from low level to high level, or POC is released due to supply voltage startup).

7. POC CIRCUIT (MASK OPTION)

The POC circuit monitors the power supply voltage and applies an internal reset in the microcontroller at the time of battery replacement. If the applied circuit satisfies the following conditions, the POC circuit can be incorporated by the mask option.

- High reliability is not required.
- Clock frequency f_X = 300 to 500 kHz
- Operating ambient temperature T_A = –20 to +70 ˚C

Cautions 1. The one-time PROM product (μPD61P34B) originally contains the POC circuit.

2. There are cases in which the POC circuit cannot detect a low power supply voltage of less than 1 ms. Therefore, if the power supply voltage has become low for a period of less than 1 ms, the POC circuit may malfunction because it does not generate an internal reset signal.

3. Clock oscillation is stopped by the resonator due to low power supply voltage before the POC circuit generates the internal reset signal. In this case, malfunction may result, for example when the power supply voltage is recovered after the oscillation is stopped. This type of phenomenon takes place because the POC circuit does not generate an internal reset signal (because the power supply voltage recovers before the low power supply voltage is detected) even though the clock has stopped. If, by any chance, a malfunction has taken place, remove the battery for a short time and put it back. In most cases, normal operation will be resumed.

4. If the applied circuit does not satisfy the conditions above, design the applied circuit in such a manner that the reset takes effect without failure within the power supply voltage range by means of an external reset circuit.

5. In order to prevent malfunction, be sure to input a low level to more than one of pins K_{I0} to K_{I3} when reset is released (when $\overline{RESET}$ pin changes from low level to high level, or POC is released due to supply voltage startup).

Remarks 1. It is recommended that a POC circuit should be incorporated if applied circuits are infrared remote-control transmitters for household appliances.

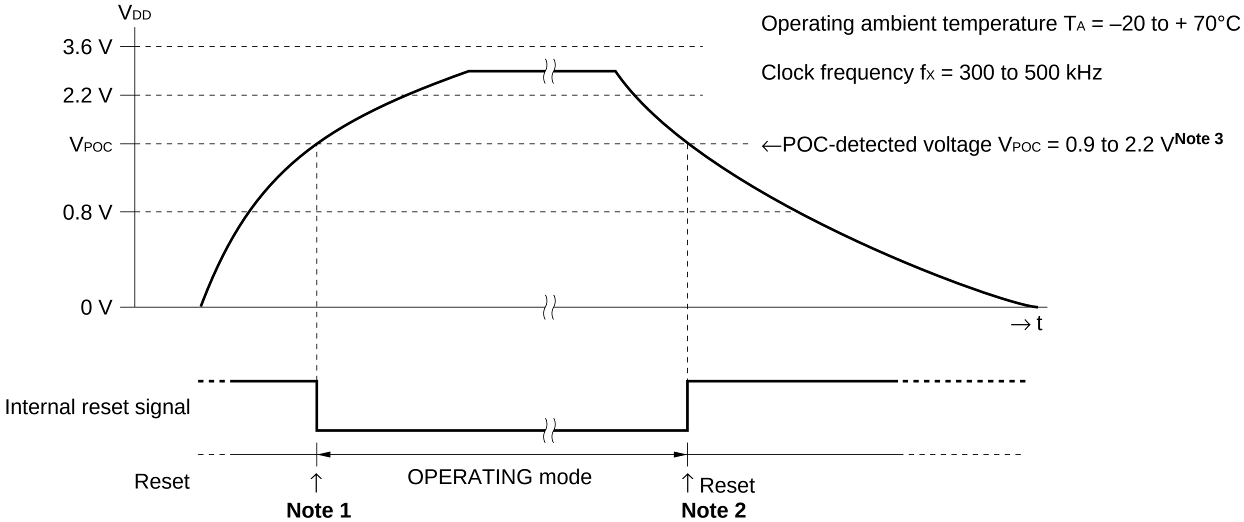
2. Even when a POC circuit is incorporated, the externally entered $\overline{RESET}$ input is valid with the OR condition; therefore, the POC circuit and the $\overline{RESET}$ input can be used at the same time. However, if the POC circuit detects a low power supply voltage, the $\overline{RESET}$ pin will be forced to low level; therefore, use an N-ch open drain output or NPN open collector output for the external reset circuit.

7.1 Functions of POC Circuit

The POC circuit has the following functions:

- Generates an internal reset signal when $V_{DD} \leq V_{POC}$.
- Cancels an internal reset signal when $V_{DD} > V_{POC}$.

Here, V_{DD}: power supply voltage, V_{POC}: POC-detected voltage.



Notes 1. In reality, there is the oscillation stabilization wait time until the circuit is switched to OPERATING mode. The oscillation stabilization wait time is about $60/f_X$ to $116/f_X$ (when about 130 to 250 μs; f_X = 455 kHz).

2. For the POC circuit to generate an internal reset signal when the power supply voltage has fallen, it is necessary for the power supply voltage to be kept less than the V_{POC} for the period of 1 ms or more. Therefore, in reality, there is the time lag of up to 1 ms until the reset takes effect.

3. The POC-detected voltage (V_{POC}) varies between 0.9 to 2.2 V; thus, the resetting may be canceled at a power supply voltage smaller than the assured range (V_{DD} = 1.8 to 3.6 V). However, as long as the conditions for operating the POC circuit are met, the actual lowest operating power supply voltage becomes lower than the POC-detected voltage. Therefore, there is no malfunction occurring due to the shortage of power supply voltage. However, malfunction for such reasons as the clock not oscillating due to low power supply voltage may occur (refer to **Cautions 3**. in **7. POC CIRCUIT**).

7.2 Oscillation Check at Low Supply Voltage

A reliable resetting operation can be expected of the POC circuit if it satisfies the condition that the clock can oscillate even at low power supply voltage (the oscillation start voltage of the resonator being even lower than the POC-detected voltage). Whether this condition is being met or not can be checked by measuring the oscillation status on a product which actually contains a POC circuit, as follows.

<1> Connect a storage oscilloscope to the X_{OUT} pin so that the oscillation status can be measured.

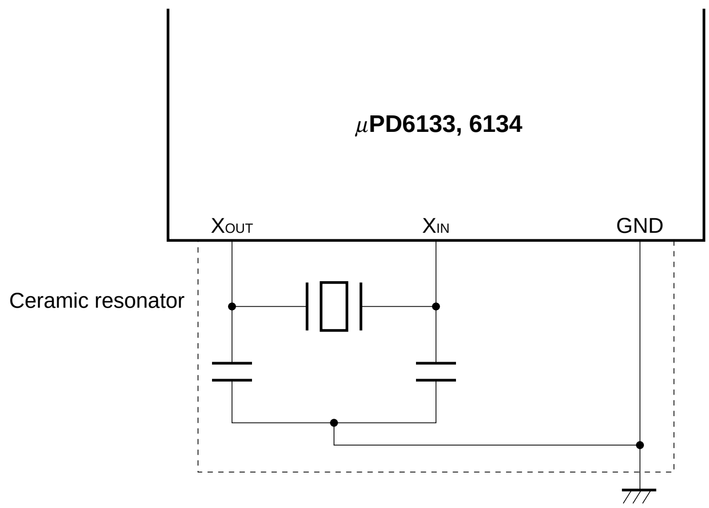
<2> Connect a power supply whose output voltage can be varied and then gradually raise the power supply voltage V_{DD} from 0 V (making sure to avoid V_{DD} > 3.6V).

At first (during V_{DD} < 0.9 V), the X_{OUT} pin is 0 V regardless of the V_{DD}. However, at the point that V_{DD} reaches the POC-detected voltage (voltage somewhere between V_{POC} = 0.9 to 2.2 V), the voltage of the X_{OUT} pin jumps to about 0.5 V_{DD}. Maintain this power supply voltage for a while to measure the waveform of the X_{OUT} pin. If, by any chance, the oscillation start voltage of the resonator is lower than the POC-detected voltage, the growing oscillation of the X_{OUT} pin can be confirmed within several ms after the V_{DD} has reached the V_{POC}.

8. SYSTEM CLOCK OSCILLATOR

The system clock oscillator consists of oscillators for ceramic resonators (f_X = 300 kHz to 1 MHz).

Figure 8-1. System Clock



The system clock oscillator stops its oscillation when reset or in STOP mode.

Caution When using the system clock oscillator, wire area indicated by the dotted-line in the diagram as follows to reduce the effects of the wiring capacitance, etc.

- **Make the wiring as short as possible.**
- **Do not allow the wiring to intersect other signal lines. Do not wire close to lines through which large fluctuating currents flow.**
- **Make sure that the point where the oscillator capacitor is installed is always at the same electric potential as the ground. Never earth with a ground pattern through which large currents flow.**
- **Do not extract signals from the oscillator.**

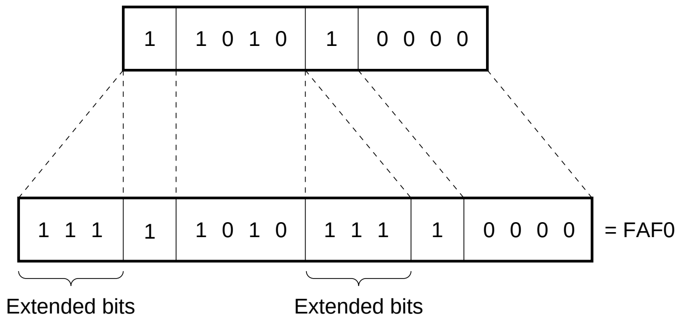
9. INSTRUCTION SET

9.1 Machine Language Output by Assembler

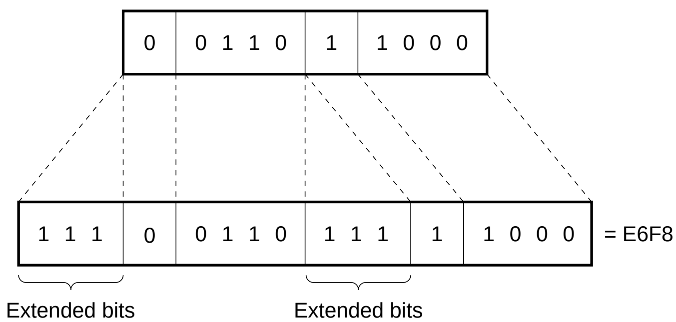
The bit length of the machine language of this product is 10 bits per word.However, the machine language that is output by the assembler is extended to 16 bits per word. As shown in the example below, the expansion is made by inserting 3-bit extended bits (111) in two locations.

Figure 9-1. Example of Assembler Output (10 bits extended to 16 bits)

<1> In the case of "ANL A, @R0H"



<2> In the case of "OUT P0, #data8"



9.2 Circuit Symbol Description

A	: Accumulator
ASR	: Address Stack Register
addr	: Program memory address
CY	: Carry flag
data4	: 4-bit immediate data
data8	: 8-bit immediate data
data10	: 10-bit immediate data
F	: Status flag
PC	: Program Counter
Pn	: Port register pair (n = 0, 1, 3, 4)
P0n	: Port register (low-order 4 bits)
P1n	: Port register (high-order 4 bits)
ROMn	: Bit n of the program memory's (n = 0-9)
Rn	: Register pair
R0n	: Data memory (General-purpose register; n = 0-F)
R1n	: Data memory (General-purpose register; n = 0-F)
SP	: Stack Pointer
T	: Timer register
T0	: Timer register (low-order 4 bits)
T1	: Timer register (high-order 4 bits)
(×)	: Content addressed with ×

9.3 Mnemonic to/from Machine Language (Assembler Output) Contrast Table

Accumulator Operation Instructions

Mnemonic	Operand	Instruction Code 1st Word	Instruction Code 2nd Word	Instruction Code 3rd Word	Operation	Instruction Length	Instruction Cycle
ANL	A, R0n	FBEn			$(A) \leftarrow (A) \wedge (Rmn)$ m = 0, 1 n = 0-F	1	1
	A, R1n	FAEn			$CY \leftarrow A_3 \bullet Rmn_3$		
	A, @R0H	FAF0			$(A) \leftarrow (A) \wedge ((P13), (R0))_{7\text{-}4}$ $CY \leftarrow A_3 \bullet ROM_7$		
	A, @R0L	FBF0			$(A) \leftarrow (A) \wedge ((P13), (R0))_{3\text{-}0}$ $CY \leftarrow A_3 \bullet ROM_3$		
	A, #data4	FBF1	data4		$(A) \leftarrow (A) \wedge data4$ $CY \leftarrow A_3 \bullet data4_3$	2	
ORL	A, R0n	FDEn			$(A) \leftarrow (A) \vee (Rmn)$ m = 0, 1 n = 0-F	1	
	A, R1n	FCEn			$CY \leftarrow 0$		
	A, @R0H	FCF0			$(A) \leftarrow (A) \vee ((P13), (R0))_{7\text{-}4}$ $CY \leftarrow 0$		
	A, @R0L	FDF0			$(A) \leftarrow (A) \vee ((P13), (R0))_{3\text{-}0}$ $CY \leftarrow 0$		
	A, #data4	FDF1	data4		$(A) \leftarrow (A) \vee data4$ $CY \leftarrow 0$	2	
XRL	A, R0n	F5En			$(A) \leftarrow (A) \veebar (Rmn)$ m = 0, 1 n = 0-F	1	
	A, R1n	F4En			$CY \leftarrow A_3 \bullet Rmn_3$		
	A, @R0H	F4F0			$(A) \leftarrow (A) \veebar ((P13), (R0))_{7\text{-}4}$ $CY \leftarrow A_3 \bullet ROM_7$		
	A, @R0L	F5F0			$(A) \leftarrow (A) \veebar ((P13), (R0))_{3\text{-}0}$ $CY \leftarrow A_3 \bullet ROM_3$		
	A, #data4	F5F1	data4		$(A) \leftarrow (A) \veebar data4$ $CY \leftarrow A_3 \bullet data4_3$	2	
INC	A	F4F3			$(A) \leftarrow (A) + 1$ if (A) = 0 $CY \leftarrow 1$ else $CY \leftarrow 1$	1	
RL	A	FCF3			$(A_{n+1}) \leftarrow (A_n), (A_0) \leftarrow (A_3)$ $CY \leftarrow A_3$		
RLZ	A	FEF3			if A = 0 reset else $(A_{n+1}) \leftarrow (A_n), (A_0) \leftarrow (A_3)$ $CY \leftarrow A_3$		

Input/output Instructions

Mnemonic	Operand	Instruction Code			Operation	Instruction Length	Instruction Cycle
		1st Word	2nd Word	3rd Word			
IN	A, P0n	FFF8 + n	—	—	(A) ← (Pmn) m = 0, 1 n = 0, 1, 3, 4	1	1
	A, P1n	FEF8 + n	—	—	CY ← 0		
OUT	P0n, A	E5F8 + n	—	—	(Pmn) ← (A) m = 0, 1 n = 0, 1, 3, 4		
	P1n, A	E4F8 + n	—	—			
ANL	A, P0n	FBF8 + n	—	—	(A) ← (A) ∧ (Pmn) m = 0, 1 n = 0, 1, 3, 4		
	A, P1n	FAF8 + n	—	—	CY ← A_3 • Pmn_3		
ORL	A, P0n	FDF8 + n	—	—	(A) ← (A) ∨ (Pmn) m = 0, 1 n = 0, 1, 3, 4		
	A, P1n	FCF8 + n	—	—	CY ← 0		
XRL	A, P0n	F5F8 + n	—	—	(A) ← (A) ⊻ (Pmn) m = 0, 1 n = 0, 1, 3, 4		
	A, P1n	F4F8 + n	—	—	CY ← A_3 • Pmn_3		

Mnemonic	Operand	Instruction Code			Operation	Instruction Length	Instruction Cycle
		1st Word	2nd Word	3rd Word			
OUT	Pn, #data8	E6F8 + n	data8		(Pn) ← data8 n = 0, 1, 3, 4	2	1

Remark Pn: P1n-P0n are dealt with in pairs.

Data Transfer Instruction

Mnemonic	Operand	Instruction Code			Operation	Instruction Length	Instruction Cycle
		1st Word	2nd Word	3rd Word			
MOV	A, R0n	FFEn			(A) ← (Rmn) m = 0, 1 n = 0-F	1	1
	A, R1n	FEEn			CY ← 0		
	A, @R0H	FEF0			(A) ← $((P13), (R0))_{7\text{-}4}$ CY ← 0		
	A, @R0L	FFF0			(A) ← $((P13), (R0))_{7\text{-}4}$ CY ← 0		
	A, #data4	FFF1	data4		(A) ← data4 CY ← 0	2	
	R0n, A	E5En			(Rmn) ← (A) m = 0, 1 n = 0-F	1	
	R1n, A	E4En					

Mnemonic	Operand	Instruction Code			Operation	Instruction Length	Instruction Cycle
		1st Word	2nd Word	3rd Word			
MOV	Rn, #data8	E6En	data8	—	(R1n-R0n) ← data8 n = 0-F	2	1
	Rn, @R0	E7En	—	—	(R1n-R0n) ← ((P13), (R0)) n = 1-F	1	

Remark Rn: R1n-R0n are dealt with in pairs.

Branch Instructions

Mnemonic	Operand	Instruction Code			Operation	Instruction Length	Instruction Cycle
		1st Word	2nd Word	3rd Word			
JMP	addr (Page 0)	E8F1	addr		PC ← addr	2	1
	addr (Page 1)	E9F1	addr				
JC	addr (Page 0)	ECF1	addr		if CY = 1 PC ← addr		
	addr (Page 1)	EAF1	addr		else PC ← PC + 2		
JNC	addr (Page 0)	EDF1	addr		if CY = 0 PC ← addr		
	addr (Page 1)	EBF1	addr		else PC ← PC + 2		
JF	addr (Page 0)	EEF1	addr		if F = 1 PC ← addr		
	addr (Page 1)	F0F1	addr		else PC ← PC + 2		
JNF	addr (Page 0)	EFF1	addr		if F = 0 PC ← addr		
	addr (Page 1)	F1F1	addr		else PC ← PC + 2		

Caution 0 and 1, which refer to PAGE0 and 1, are not written when describing mnemonics.

Subroutine Instructions

Mnemonic	Operand	Instruction Code			Operation	Instruction Length	Instruction Cycle
		1st Word	2nd Word	3rd Word			
CALL	addr (Page 0)	E6F2	E8F1	addr	SP ← SP + 1, ASR ← PC, PC ← addr	3	2
	addr (Page 1)	E6F2	E9F1	addr			
RET		E8F2			PC ← ASR, SP ← SP – 1	1	1

Caution 0 and 1, which refer to PAGE0 and 1, are not written when describing mnemonics.

Timer Operation Instructions

Mnemonic	Operand	Instruction Code			Operation	Instruction Length	Instruction Cycle
		1st Word	2nd Word	3rd Word			
MOV	A, T0	FFFF			(A) ← (Tn) n = 0, 1	1	1
	A, T1	FEFF			CY ← 0		
	T0, A	E5FF			(Tn) ← (A) n = 0, 1		
	T1, A	F4FF			(T) n ← 0		

Mnemonic	Operand	Instruction Code			Operation	Instruction Length	Instruction Cycle
		1st Word	2nd Word	3rd Word			
MOV	T, #data10	E6FF	data10		(T) ← data10	1	1
	T, @R0	F4FF			(T) ← ((P13), (R0))		

Others

Mnemonic	Operand	Instruction Code			Operation	Instruction Length	Instruction Cycle
		1st Word	2nd Word	3rd Word			
HALT	#data4	E2F1	data4		Standby mode	2	1
STTS	#data4	E3F1	data4		if statuses match F ← 1 else F ← 0		
	R0n	E3En			if statuses match F ← 1 else F ← 0 n = 0-F	1	
SCAF		FAF3			if A = 0FH CY ← 1 else CY ← 0		
NOP		E0E0			PC ← PC + 1		

9.4 Accumulator Operation Instructions

ANL A, R0n

ANL A, R1n

<1> Instruction code :

1	1	0	1	R_4	0	R_3	R_2	R_1	R_0

<2> Cycle count : 1

<3> Function : (A) ← (A) ∧ (Rmn)   m = 0, 1   n = 0 to F

CY ← A_3 • Rmn_3

The accumulator contents and the register Rmn contents are ANDed and the results are entered in the accumulator.

ANL A, @R0H

ANL A, @R0L

<1> Instruction code :

1	1	0	1	0/1	1	0	0	0	0

<2> Cycle count : 1

<3> Function : (A) ← (A) ∧ ((P13), (R0))$_{7\text{-}4}$ (in the case of ANL A, @R0H)

CY ← A_3 • ROM_7

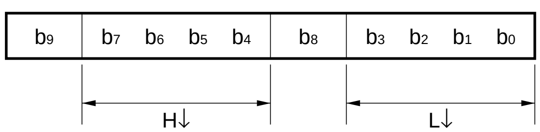
(A) ← (A) ∧ ((P13), (R0))$_{3\text{-}0}$ (in the case of ANL A, @R0L)

CY ← A_3 • ROM_3

The accumulator contents and the program memory contents specified with the control register P13 and register pair R_{10}-R_{00} are ANDed and the results are entered in the accumulator.

If H is specified, b_7, b_6, b_5 and b_4 take effect. If L is specified, b_3, b_2, b_1 and b_0 take effect.

- **Program memory (ROM) organization**



Valid bits at the time of accumulator operation

ANL A, #data4

<1> Instruction code :

1	1	0	1	1	1	0	0	0	1
0	0	0	0	0	0	d_3	d_2	d_1	d_0

<2> Cycle count : 1

<3> Function : (A) ← (A) ∧ data4

CY ← A_3 • $data4_3$

The accumulator contents and the immediate data are ANDed and the results are entered in the accumulator.

ORL A, R0n

ORL A, R1n

<1> Instruction code :

1	1 1 0 R_4	0	R_3 R_2 R_1 R_0

<2> Cycle count : 1

<3> Function : (A) ← (A) ∨ (Rmn) m = 0, 1 n = 0 to F

CY ← 0

The accumulator contents and the register Rmn contents are ORed and the results are entered in the accumulator.

ORL A, @R0H

ORL A, @R0L

<1> Instruction code :

1	1 1 0 0/1	1	0 0 0 0

<2> Cycle count : 1

<3> Function : (A) ← (A) ∨ (P13), $(R0))_{7\text{-}4}$ (in the case of ORL A, @R0H)

(A) ← (A) ∨ (P13), $(R0))_{3\text{-}0}$ (in the case of ORL A, @R0L)

CY ← 0

The accumulator contents and the program memory contents specified with the control register P13 and register pair R_{10}-R_{00} are ORed and the results are entered in the accumulator.

If H is specified, b_7, b_6, b_5 and b_4 take effect. If L is specified, b_3, b_2, b_1 and b_0 take effect.

ORL A, #data4

<1> Instruction code :

1	1 1 0 1	1	0 0 0 1
0	0 0 0 0	0	d_3 d_2 d_1 d_0

<2> Cycle count : 1

<3> Function : (A) ← (A) ∨ data4

CY ← 0

The accumulator contents and the immediate data are exclusive-ORed and the results are entered in the accumulator.

XRL A, R0n

XRL A, R1n

<1> Instruction code :

1	0 1 0 R_4	0	R_3 R_2 R_1 R_0

<2> Cycle count : 1

<3> Function : (A) ← (A) ⊻ (Rmn) m = 0, 1 n = 0 to F

CY ← A_3 • Rmn_3

The accumulator contents and the register Rmn contents are ORed and the results are entered in the accumulator.

XRL A, @R0H

XRL A, @R0L

<1> Instruction code :

1	0 1 0 0/1	1	0 0 0 0

<2> Cycle count : 1

<3> Function : (A) ← (A) $\veebar$ (P13), (R0))$_{7\text{-}4}$ (in the case of XRL A, @R0H)

CY ← A_3 • ROM_7

(A) ← (A) $\veebar$ (P13), (R0))$_{3\text{-}0}$ (in the case of XRL A, @R0L)

CY ← A_3 • ROM_3

The accumulator contents and the program memory contents specified with the control register P13 and register pair R_{10}-R_{00} are exclusive-ORed and the results are entered in the accumulator.

If H is specified, b_7, b_6, b_5, and b_4 take effect. If L is specified, b_3, b_2, b_1, and b_0 take effect.

XRL A, #data4

<1> Instruction code :

1	0 1 0 1	1	0 0 0 1
0	0 0 0 0	0	d_3 d_2 d_1 d_0

<2> Cycle count : 1

<3> Function : (A) ← (A) $\veebar$ data4

CY ← A_3 • $data4_3$

The accumulator contents and the immediate data are exclusive-ORed and the results are entered in the accumulator.

INC A

<1> Instruction code :

1	0 1 0 0	1	0 0 1 1

<2> Cycle count : 1

<3> Function : (A) ← (A) + 1

if A = 0 CY ← 1

else CY ← 0

The accumulator contents are incremented (+1).

RL A

<1> Instruction code :

1	1 1 0 0	1	0 0 1 1

<2> Cycle count : 1

<3> Function : (A_{n+1}) ← (An), (A_0) ← (A_3)

CY ← A_3

The accumulator contents are rotated anticlockwise bit by bit.

RLZ A

<1> Instruction code :

1	1 1 1 0	1	0 0 1 1

<2> Cycle count : 1

<3> Function : if A = 0 reset

else (A_{n+1}) ← (An), (A_0) ←(A_3)

CY ← A_3

The accumulator contents are rotated anticlockwise bit by bit.

If A = 0H at the time of command execution, an internal reset takes effect.

9.5 Input/Output Instructions

IN A, P0n
IN A, P1n

<1> Instruction code : | 1 | 1 1 1 P_4 | 1 | 1 P_2 P_1 P_0 |

<2> Cycle count : 1

<3> Function : (A) ← (Pmn) m = 0, 1 n = 0, 1, 3, 4
CY ← 0

The port Pmn data is loaded (read) onto the accumulator.

OUT P0n, A
OUT P1n, A

<1> Instruction code : | 0 | 0 1 0 P_4 | 1 | 1 P_2 P_1 P_0 |

<2> Cycle count : 1

<3> Function : (Pmn) ← (A) m = 0, 1 n = 0, 1, 3, 4

The accumulator contents are transferred to port Pmn to be latched.

ANL A, P0n
ANL A, P1n

<1> Instruction code : | 1 | 1 0 1 P_4 | 1 | 1 P_2 P_1 P_0 |

<2> Cycle count : 1

<3> Function : (A) ← (A) $\wedge$ (Pmn) m = 0, 1 n = 0, 1, 3, 4
CY ← A_3 • Pmn

The accumulator contents and the port Pmn contents are ANDed and the results are entered in the accumulator.

ORL A, P0n
ORL A, P1n

<1> Instruction code : | 1 | 1 1 0 P_4 | 1 | 1 P_2 P_1 P_0 |

<2> Cycle count : 1

<3> Function : (A) ← (A) $\vee$ (Pmn) m = 0, 1 n = 0, 1, 3, 4
CY ← 0

The accumulator contents and the port Pmn contents are ORed and the results are entered in the accumulator.

XRL A, P0n
XRL A, P1n

<1> Instruction code : | 1 | 0 1 0 P_4 | 1 | 1 P_2 P_1 P_0 |

<2> Cycle count : 1

<3> Function : (A) ← (A) $\veebar$ (Pmn) m = 0, 1 n = 0, 1, 3, 4
CY ← A_3 • Pmn

The accumulator contents and the port Pmn contents are exclusive-ORed and the results are entered in the accumulator.

OUT Pn, #data8

<1> Instruction code :

0	0 1 1 0	1	1 P_2 P_1 P_0
0	d_7 d_6 d_5 d_4	0	d_3 d_2 d_1 d_0

<2> Cycle count : 1

<3> Function : (Pn) ← data8   n = 0, 1, 3, 4

The immediate data is transferred to port Pn. In this case, port Pn refers to P_{1n}-P_{0n} operating in pairs.

9.6 Data Transfer Instructions

MOV A, R0n
MOV A, R1n

<1> Instruction code :

1	1 1 1 R_4	0	R_3 R_2 R_1 R_0

<2> Cycle count : 1

<3> Function : (A) ← (Rmn)   m = 0, 1   n = 0 to F
CY ← 0

The register Rmn contents are transferred to the accumulator.

MOV A, @R0H

<1> Instruction code :

1	1 1 1 0	1	0 0 0 0

<2> Cycle count : 1

<3> Function : (A) ← $((P13), (R0))_{7\text{-}4}$
CY ← 0

The high-order 4 bits (b_7 b_6 b_5 b_4) of the program memory specified with control register P13 and register pair R_{10}-R_{00} are transferred to the accumulator. b_9 is ignored.

MOV A, @R0L

<1> Instruction code :

1	1 1 1 1	1	0 0 0 0

<2> Cycle count : 1

<3> Function : (A) ← $((P13), (R0))_{3\text{-}0}$
CY ← 0

The low-order 4 bits (b_3 b_2 b_1 b_0) of the program memory specified with control register P13 and register pair R_{10}-R_{00} are transferred to the accumulator. b_8 is ignored.

- **Program memory (ROM) contents**

	@R_0H		@R_0L
b_9	b_7 b_6 b_5 b_4	b_8	b_3 b_2 b_1 b_0

MOV A, #data4

<1> Instruction code :

1	1 1 1 1	1	0 0 0 1
0	0 0 0 0	0	d_3 d_2 d_1 d_0

<2> Cycle count : 1

<3> Function : (A) ← data4
CY ← 0

The immediate data is transferred to the accumulator.

MOV R0n, A

MOV R1n, A

<1> Instruction code :

0	0 1 0 R_4	0	R_3 R_2 R_1 R_0

<2> Cycle count : 1

<3> Function : (Rmn) ← (A)   m = 0, 1   n = 0 to F

The accumulator contents are transferred to register Rmn.

MOV Rn, #data8

<1> Instruction code :

0	0 1 1 0	0	R_3 R_2 R_1 R_0
0	d_7 d_6 d_5 d_4	0	d_3 d_2 d_1 d_0

<2> Cycle count : 1

<3> Function : (R1n-R0n) ← data8   n = 0-F

The immediate data is transferred to the register. Using this instruction, registers operate as register pairs.

The pair combinations are as follows:

R_0 : R_{10} - R_{00}

R_1 : R_{11} - R_{01}

:

R_E : R_{1E} - R_{0E}

R_F : R_{1F} - R_{0F}

Lower column

Higher column

MOV Rn, @R0

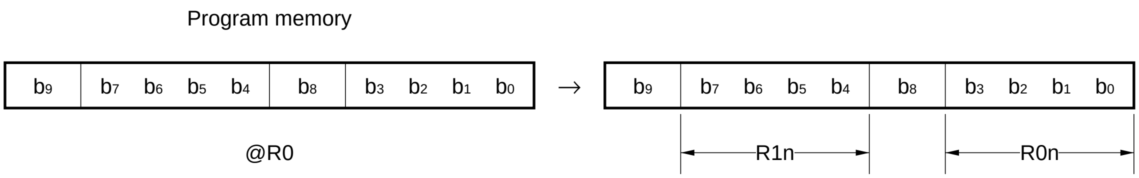
<1> Instruction code :

0	0 1 1 1	0	R_3 R_2 R_1 R_0

<2> Cycle count : 1

<3> Function : (R1n-R0n) ← ((P13), R0))   n = 1 to F

The program memory contents specified with control register P13 and register pair R_{10}-R_{00} are transferred to register pair R1n-R0n. The program memory consists of 10 bits and has the following state after the transfer to the register.



The high-order 2 bits of the program memory address is specified with the control register (P13).

9.7 Branch Instructions

The program memory consists of pages in steps of 1K (000H to 3FFH). However, as the assembler automatically performs page optimization, it is unnecessary to designate pages. The pages allowed for each product are as follows.

μPD6133 (ROM: 0.5K steps) : page 0
μPD6134 (ROM: 1K steps) : page 0
μPD61P34B (ROM: 1K steps): page 0

JMP addr

<1> Instruction code : page 0 ; page 1

page 0	0	1 0 0 0	1	0 0 0 1
page 1	0	1 0 0 1	1	0 0 0 1
	a_9	a_7 a_6 a_5 a_4	a_8	a_3 a_2 a_1 a_0

<2> Cycle count : 1

<3> Function : PC ← addr

The 10 bits (PC_{9-0}) of the program counter are replaced directly by the specified address addr (a_9 to a_0).

JC addr

<1> Instruction code : page 0 ; page 1

page 0	0	1 1 0 0	1	0 0 0 1
page 1	0	1 0 1 0	1	0 0 0 1
	a_9	a_7 a_6 a_5 a_4	a_8	a_3 a_2 a_1 a_0

<2> Cycle count : 1

<3> Function : if CY = 1 PC ← addr
else PC ← PC + 2

If the carry flag CY is set (to 1), a jump is made to the address specified with addr (a_9 to a_0).

JNC addr

<1> Instruction code : page 0 ; page 1

page 0	0	1 1 0 1	1	0 0 0 1
page 1	0	1 0 1 1	1	0 0 0 1
	a_9	a_7 a_6 a_5 a_4	a_8	a_3 a_2 a_1 a_0

<2> Cycle count : 1

<3> Function : if CY = 0 PC ← addr
else PC ← PC + 2

If the carry flag CY is cleared (to 0), a jump is made to the address specified with addr (a9 to a0).

JF addr

<1> Instruction code : page 0 ; page 1

page 0	0	1 1 1 0	1	0 0 0 1
page 1	1	0 0 0 0	1	0 0 0 1
	a_9	a_7 a_6 a_5 a_4	a_8	a_3 a_2 a_1 a_0

<2> Cycle count : 1

<3> Function : if F = 1 PC ← addr
else PC ← PC + 2

If the status flag F is set (to 1), a jump is made to the address specified with addr (a_9 to a_0).

JNF addr

<1> Instruction code : page 0 ; page 1

page 0	0	1 1 1 1	1	0 0 0 1
page 1	1	0 0 0 1	1	0 0 0 1
	a_9	a_7 a_6 a_5 a_4	a_8	a_3 a_2 a_1 a_0

<2> Cycle count : 1

<3> Function : if F = 0 PC ← addr
else PC ← PC + 2

If the status flag F is cleared (to 0), a jump is made to the address specified with addr (a_9 to a_0).

9.8 Subroutine Instructions

The program memory consists of pages in steps of 1K (000H to 3FFH). However, as the assembler automatically performs page optimization, it is unnecessary to designate pages. The pages allowed for each product are as follows.

μPD6133 (ROM: 0.5K steps) : page 0
μPD6134 (ROM: 1K steps) : page 0
μPD61P34B (ROM: 1K steps): page 0

CALL addr

<1> Instruction code :

	0	0 1 1 0	1	0 0 1 0						
page 0	0	1 0 0 0	1	0 0 0 1	; page 1	0	1 0 0 1	1	0 0 0 1
	a_9	a_7 a_6 a_5 a_4	a_8	a_3 a_2 a_1 a_0						

<2> Cycle count : 2

<3> Function : SP ← SP + 1
ASR ← PC
PC ← addr

Increments (+1) the stack pointer value and saves the program counter value in the address stack register. Then, enters the address specified with the operand addr (a_9 to a_0) into the program counter. If a carry is generated when the stack pointer value is incremented (+1), an internal reset takes effect.

RET

<1> Instruction code :

0	1 0 0 0	1	0 0 1 0

<2> Cycle count : 1

<3> Function : PC ← ASR
SP ← SP – 1

Restores the value saved in the address stack register to the program counter. Then, decrements (–1) the stack pointer.
If a borrow is generated when the stack pointer value is decremented (–1), an internal reset takes effect.

9.9 Timer Operation Instructions

MOV A, T0
MOV A, T1

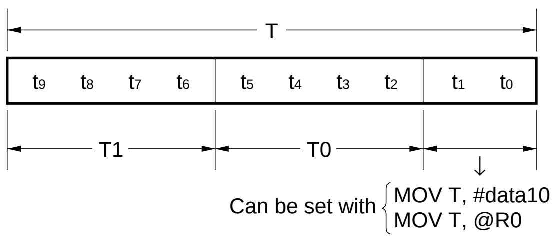
<1> Instruction code :

1	1	1	1	0/1	1	1	1	1	1

<2> Cycle count : 1

<3> Function : (A) ← (Tn) n = 0, 1
CY ← 0

The timer Tn contents are transferred to the accumulator. T1 corresponds to (t_9, t_8, t_7, t_6); T0 corresponds to (t_5, t_4, t_3, t_2).



MOV T0, A
MOV T1, A

<1> Instruction code :

0	0	1	0	0/1	1	1	1	1	1

<2> Cycle count : 1

<3> Function : (Tn) ← (A) n = 0, 1

The accumulator contents are transferred to the timer register Tn. T1 corresponds to (t_9, t_8, t_7, t_6); T0 corresponds to (t_5, t_4, t_3, t_2). After executing this instruction, if data is transferred to T1, t_1 becomes 0; if data is transferred to T0, t_0 becomes 0.

MOV T, #data10

<1> Instruction code :

0	0	1	1	0	1	1	1	1	1
t_1	t_9	t_8	t_7	t_6	t_0	t_5	t_4	t_3	t_2

<2> Cycle count : 1

<3> Function : (T) ← data10

The immediate data is transferred to the timer register T (t_9-t_0).

Remark The timer time is set with (set value + 1) × 8/f_X (or 16/f_X).

MOV T, @R0

<1> Instruction code :

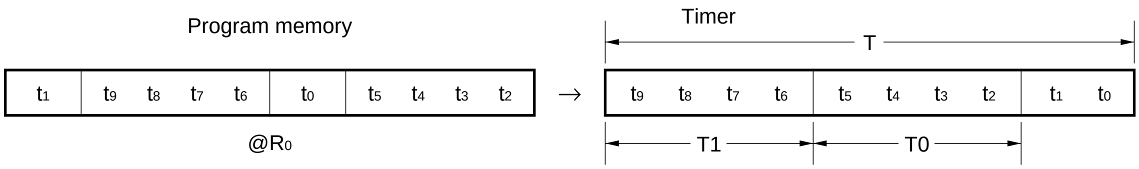
0	0	1	1	1	1	1	1	1	1

<2> Cycle count : 1

<3> Function : (T) ← ((P13), (R0))

Transfers the program memory contents to the timer register T (t_9 to t_0) specified with the control register P13 and the register pair R_{10}-R_{00}.

The program memory, which consists of 10 bits, is placed in the following state after the transfer to the register.



The high-order 2 bits of the program memory address are specified with the control register (P13).

Caution When setting a timer value in the program memory, ensure to use the DT directive.

9.10 Others

HALT #data4

<1> Instruction code :

0	0	0	1	0	1	0	0	0	1
0	0	0	0	0	0	d_3	d_2	d_1	d_0

<2> Cycle count : 1

<3> Function : Sandby mode

Places the CPU in standby mode.

The condition for having the standby mode (HALT/STOP mode) canceled is specified with the immediate data.

STTS R0n

<1> Instruction code :

0	0	0	1	1	0	R_3	R_2	R_1	R_0

<2> Cycle count : 1

<3> Function : if statuses match F ← 1

else F ← 0 n = 0 to F

Compares the S_0, S_1, $K_{I/O}$, K_I, and TIMER statuses with the register R_{0n} contents. If at least one of the statuses coincides with the bits that have been set, the status flag F is set (to 1).

If none of them coincide, the status flag F is cleared (to 0).

STTS #data4

<1> Instruction code : | 0 | 0 0 1 1 | 1 | 0 0 0 1 |
: | 0 | 0 0 0 0 | 0 | d_3 d_2 d_1 d_0 |

<2> Cycle count : 1

<3> Function : if statuses match F ← 1
else F ← 0

Compares the S_0, S_1, $K_{I/O}$, K_I, and TIMER statuses with the immediate data contents. If at least one of the statuses coincides with the bits that have been set, the status flag F is set (to 1).
If none of them coincide, the status flag F is cleared (to 0).

SCAF (Set Carry If A_{CC} = F_H)

<1> Instruction code : | 1 | 1 0 1 0 | 1 | 0 0 1 1 |

<2> Cycle count : 1

<3> Function : if A = 0FH CY ← 1
else CY ← 0

Sets the carry flag CY (to 1) if the accumulator contents are FH.
The accumulator values after executing the SCAF instruction are as follows:

Accumulator Value		Carry Flag
Before execution	After execution	
×××0	0000	0 (clear)
××01	0001	0 (clear)
×011	0011	0 (clear)
0111	0111	0 (clear)
1111	1111	1 (set)

Remark ×: don't care

NOP

<1> Instruction code : | 0 | 0 0 0 0 | 0 | 0 0 0 0 |

<2> Cycle count : 1

<3> Function : PC ← PC + 1

No operation

10. ASSEMBLER RESERVED WORDS

10.1 Mask Option Directives

When creating the μPD6133 program, it is necessary to use a mask option directive in the assembler's source program to specify a mask option.

10.1.1 OPTION and ENDOP directives

From the OPTION directive on to the ENDOP directive are called the mask option definition block. The format of the mask option definition block is as follows:

Format

Symbol field	Mnemonic field	Operand field	Comment field
[Label:]	OPTION		[; Comment]
	:		
	:		
	ENDOP		

10.1.2 Mask option definition directives

The directives that can be used in the mask option definition block are listed in Table 10-1.

An example of the mask option definition is shown below.

Example

Symbol field	Mnemonic field	Operand field	Comment field
	OPTION		
	USEPOC		; POC circuit incorporated
	ENDOP		

Table 10-1. List of Mask Option Definition Directives

Name	Mask Option Definition Directive	PRO File	
		Address value	Data value
POC	USEPOC (POC circuit incorporated)	2044H	01
	NOUSEPOC (Without POC circuit)		00

11. ELECTRICAL SPECIFICATIONS

Absolute Maximum Ratings (T_A = +25 °C)

Parameter	Symbol	Test Conditions		Rating	Unit
Power supply voltage	V_{DD}			–0.3 to +5.0	V
Input voltage	V_I	$K_{I/O}$, K_I, S_0, S_1, $\overline{RESET}$		–0.3 to V_{DD} + 0.3	V
Output voltage	V_O			–0.3 to V_{DD} + 0.3	V
High-level output current	I_{OH} Note	REM	Peak value	–30	mA
			rms	–20	mA
		$\overline{LED}$	Peak value	–7.5	mA
			rms	–5	mA
		One $K_{I/O}$ pin	Peak value	–13.5	mA
			rms	–9	mA
		Total of $\overline{LED}$ and $K_{I/O}$ pins	Peak value	–18	mA
			rms	–12	mA
Low-level output current	I_{OL} Note	REM	Peak value	7.5	mA
			rms	5	mA
		$\overline{LED}$	Peak value	7.5	mA
			rms	5	mA
Operating ambient temperature	T_A			–40 to +85	°C
Storage temperature	T_{stg}			–65 to +150	°C

Note Work out the rms with: [rms] = [Peak value] $\times \sqrt{\text{Duty}}$.

Caution Product quality may suffer if the absolute rating is exceeded for any parameter, even momentarily. In other words, an absolute maxumum rating is a value at which the possibility of psysical damage to the product cannnot be ruled out. Care must therefore be taken to ensure that the these ratings are not exceeded during use of the product.

Recommended Power Supply Voltage Range (T_A = –40 to +85 °C)

Parameter	Symbol	Test Conditions	MIN.	TYP.	MAX.	Unit
Power supply voltage	V_{DD}	f_X = 300 to 500 kHz	1.8	3.0	3.6	V
		f_X = 500 kHz to 1 MHz	2.2	3.0	3.6	V
		When using the POC circuit (mask option) T_A = –20 to +70 °C f_X = 300 to 500 kHz	2.2	3.0	3.6	V

DC Characteristics (T_A = –40 to +85 ˚C, V_{DD} = 1.8 to 3.6 V)

Parameter	Symbol	Test Conditions		MIN.	TYP.	MAX.	Unit
High-level input voltage	V_{IH1}	$\overline{RESET}$		0.8 V_{DD}		V_{DD}	V
	V_{IH2}	$K_{I/O}$		0.65 V_{DD}		V_{DD}	V
	V_{IH3}	K_I, S_0, S_1		0.65 V_{DD}		V_{DD}	V
Low-level input voltage	V_{IL1}	$\overline{RESET}$		0		0.2 V_{DD}	V
	V_{IL2}	$K_{I/O}$		0		0.3 V_{DD}	V
	V_{IL3}	K_I, S_0, S_1		0		0.15 V_{DD}	V
High-level input leakage current	I_{LIH1}	K_I V_I = V_{DD}, pull-down resistor not incorporated				3	μA
	I_{LIH2}	S_0, S_1 V_I = V_{DD}, pull-down resistor not incorporated				3	μA
Low-level input leakage current	I_{LIL1}	K_I V_I = 0 V				–3	μA
	I_{LIL2}	$K_{I/O}$ V_I = 0 V				–3	μA
	I_{LIL3}	S_0, S_1 V_I = 0 V				–3	μA
High-level output voltage	V_{OH1}	REM, $\overline{LED}$, $K_{I/O}$	I_{OH} = –0.3 mA	0.8 V_{DD}			V
Low-level output voltage	V_{OL1}	REM, $\overline{LED}$	I_{OL} = 0.3 mA			0.3	V
	V_{OL2}	$K_{I/O}$	I_{OL} = 15 μA			0.4	V
High-level output current	I_{OH1}	REM	V_{DD} = 3.0 V, V_{OH} = 1.0 V	–5	–9		mA
	I_{OH2}	$K_{I/O}$	V_{DD} = 3.0 V, V_{OH} = 2.2 V	–2.5	–5		mA
Low-level output current	I_{OL1}	$K_{I/O}$	V_{DD} = 3.0 V, V_{OL} = 0.4 V	30	70		μA
			V_{DD} = 3.0 V, V_{OL} = 2.2 V	100	220		μA
Built-in pull-up resistor	R_1	$\overline{RESET}$		25	50	100	kΩ
Built-in pull-down resistor	R_2	$\overline{RESET}$		2.5	5	15	kΩ
	R_3	K_I, S_0, S_1		75	150	300	kΩ
	R_4	$K_{I/O}$		130	250	500	kΩ
Data hold power supply voltage	V_{DDDR}	In STOP mode		0.9		3.6	V
Supply current[Note]	I_{DD1}	OPERATING mode	f_X = 1.0 MHz, V_{DD} = 3 V ± 10 %		0.4	0.8	mA
			f_X = 455 kHz, V_{DD} = 3 V ± 10 %		0.35	0.7	mA
	I_{DD2}	HALT mode	f_X = 1.0 MHz, V_{DD} = 3 V ± 10 %		0.35	0.7	mA
			f_X = 455 kHz, V_{DD} = 3 V ± 10 %		0.3	0.6	mA
	I_{DD3}	STOP mode	V_{DD} = 3 V ± 10 %		1.0	8.0	μA
			V_{DD} = 3 V ± 10 %, T_A = 25 ˚C		0.1	1.0	μA

Note The POC circuit current and the current flowing in the built-in pull-up resistor are not included.

AC Characteristics (T_A = –40 to +85 °C, V_{DD} = 1.8 to 3.6 V)

Parameter	Symbol	Test Conditions		MIN.	TYP.	MAX.	Unit
Command execution time	t_{CY}	V_{DD} = 2.2 to 3.6 V		7.9		27	μs
				15.9		27	μs
K_I, S_0, S_1 high-level width	t_H			10			μs
		When releasing Standby mode	in HALT mode	10			μs
			in STOP mode	**Note**			μs
$\overline{\text{RESET}}$ low-level width	t_{RSL}			10			μs

Note 10 + 36/f_X + oscillation growth time

Remark t_{CY} = 8/f_X (f_X: System clock oscillator frequency)

POC Circuit (mask option[Note 1]) (T_A = –20 to +70 °C)

Parameter	Symbol	Test Conditions	MIN.	TYP.	MAX.	Unit
POC-detected voltage[Note 2]	V_{POC}		0.9	1.6	2.2	V
POC circuit current	I_{POC}			0.9	1.0	μA

★ **Notes** **1.** Operates effectively under the conditions of f_X = 300 to 500 kHz.

2. Refers to the voltage with which the POC circuit cancels an internal reset. If $V_{POC} < V_{DD}$, the internal reset is released.

From the time of $V_{POC} \geq V_{DD}$ until the internal reset takes effect, lag of up to 1 ms occurs. When the period of $V_{POC} \geq V_{DD}$ lasts less than 1 ms, the internal reset may not take effect.

System Clock Oscillator Characteristics (T_A = –40 to +85 °C, V_{DD} = 1.8 to 3.6 V)

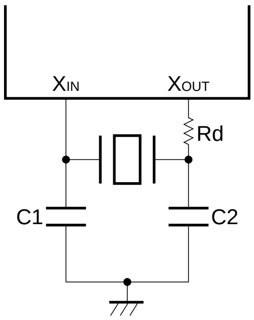
Parameter	Symbol	Test Conditions	MIN.	TYP.	MAX.	Unit
Oscillator frequency (ceramic resonator)	f_X		300	455	500	kHz
		V_{DD} = 2.2 to 3.6 V	300	455	1000	kHz

Recommended Ceramic Resonator (T_A = –40 to +85 °C)

Manufacturer (Order Disregarded)	Part Number	Recommended Constant			Power Supply Voltage [V]Note		Remark
		C1 [pF]	C1 [pF]	Rd [kΩ]	MIN.	MAX.	
Murata Mfg. Co., Ltd	CSB455E	150	150	0	1.8	3.6	
	CSB600P	150	150	0	2.2	3.6	
	CSB910J	100	100	0	2.2	3.6	
Kyocera Corp.	KBR-455BK	220	220	0	1.8	3.6	
	KBR-455BY	220	220	0	1.8	3.6	
	KBR-1000F	100	100	0	2.2	3.6	
	KBR-1000Y	100	100	0	2.2	3.6	
TDK Corp.	FCR400K3	220	220	0	1.8	3.6	
	FCR455K3	220	220	0	1.8	3.6	
	FCR500K3	220	220	0	1.8	3.6	
Matsushita Electronics Components Co., Ltd.	EFOA440K06B	470	470	0	1.8	3.6	
	EFOA455K06B	470	470	0	1.8	3.6	
	EFOA480K06B	470	470	0	1.8	3.6	

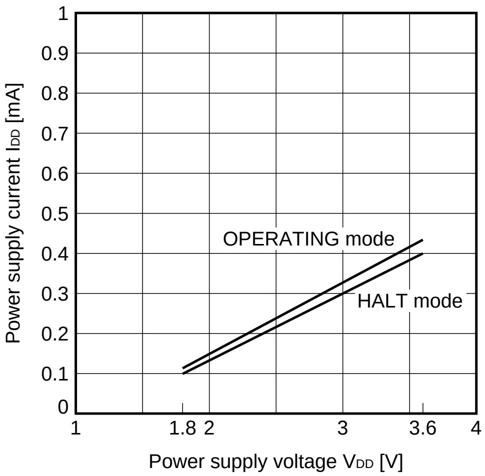
Note When a POC circuit (mask option) is not incorporated

An external circuit example

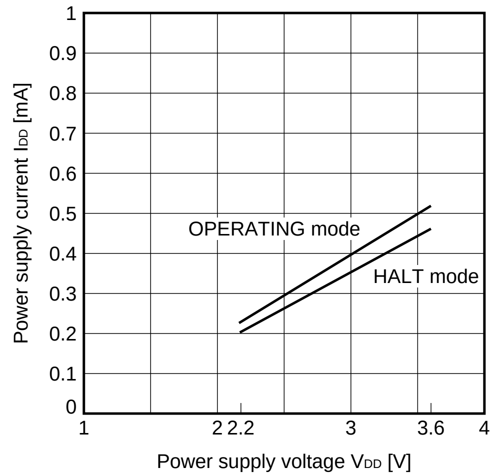


12. CHARACTERISTIC CURVE (REFERENCE VALUES) (common to μPD6133 and 6134)

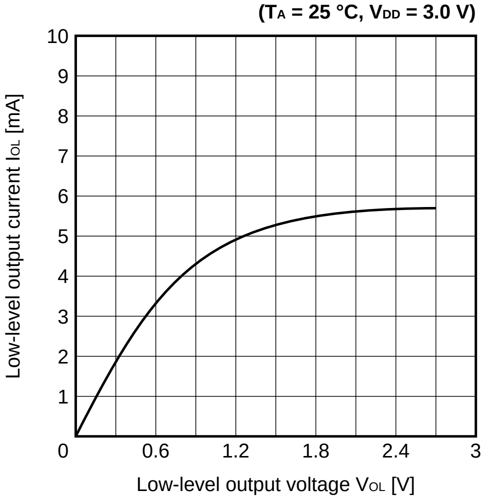
I_{DD} vs V_{DD} (f_X = 455 kHz)
(T_A = 25 °C , V_{DD} = 1.8 to 3.6 V)



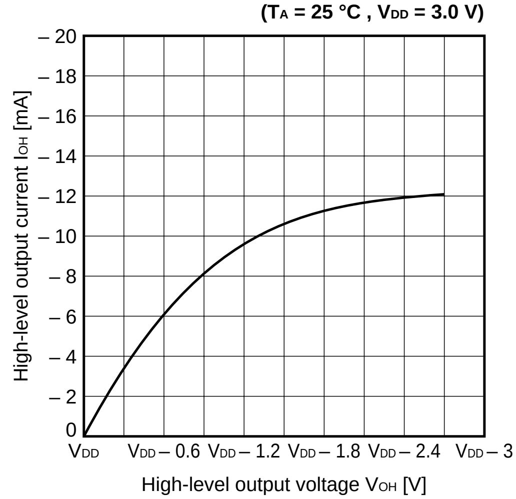
I_{DD} vs V_{DD} (f_X = 1 MHz)
(T_A = 25 °C , V_{DD} = 2.2 to 3.6 V)



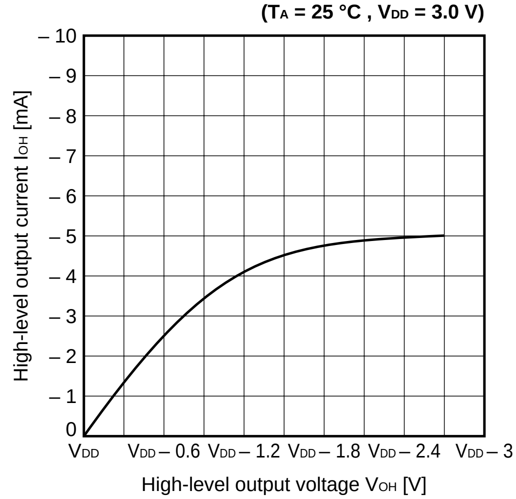
I_{OL} vs V_{OL} (REM, $\overline{\text{LED}}$)
(T_A = 25 °C, V_{DD} = 3.0 V)



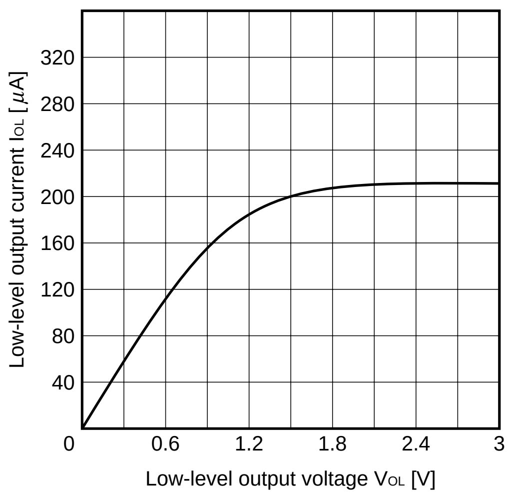
I_{OH} vs V_{OH} (REM)
(T_A = 25 °C , V_{DD} = 3.0 V)



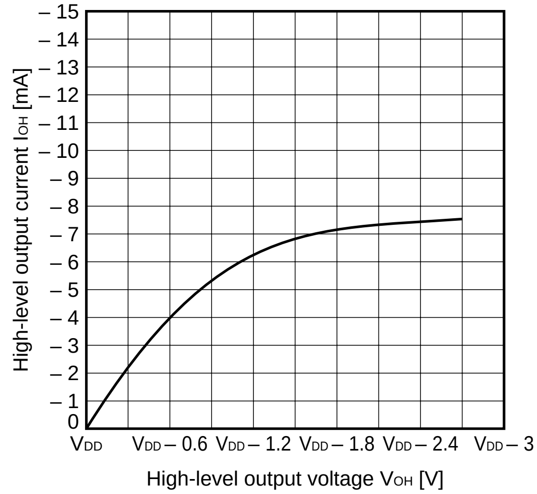
I_{OH} vs V_{OH} ($\overline{\text{LED}}$)
(T_A = 25 °C , V_{DD} = 3.0 V)



I_{OL} vs V_{OL} ($K_{I/O}$)
(T_A = 25 °C, V_{DD} = 3.0 V)



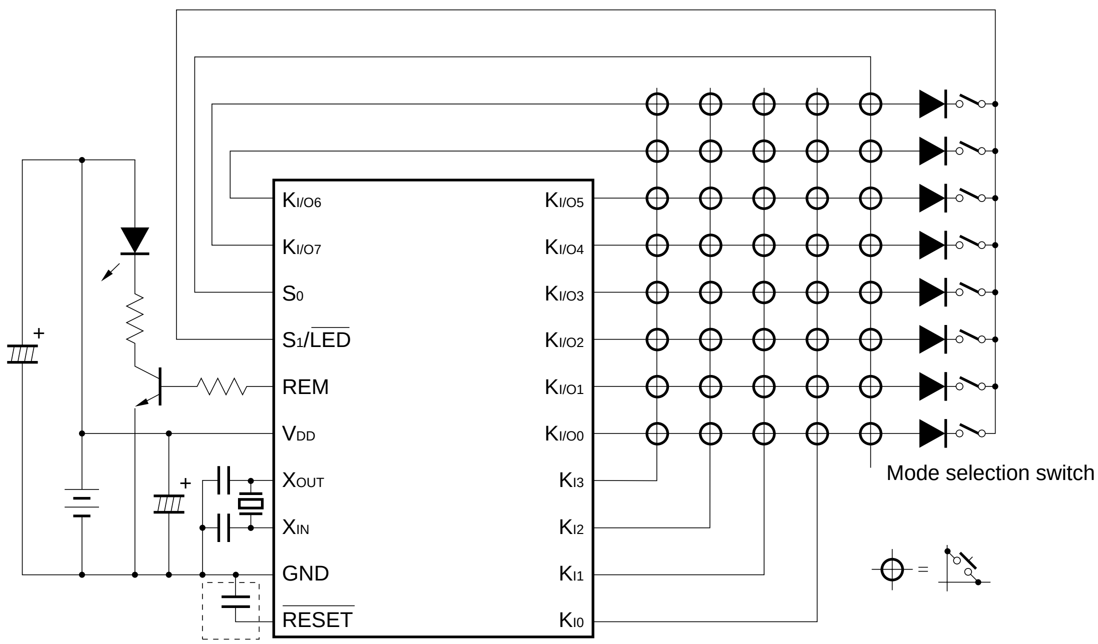
I_{OH} vs V_{OH} ($K_{I/O}$)
(T_A = 25 °C, V_{DD} = 3.0 V)



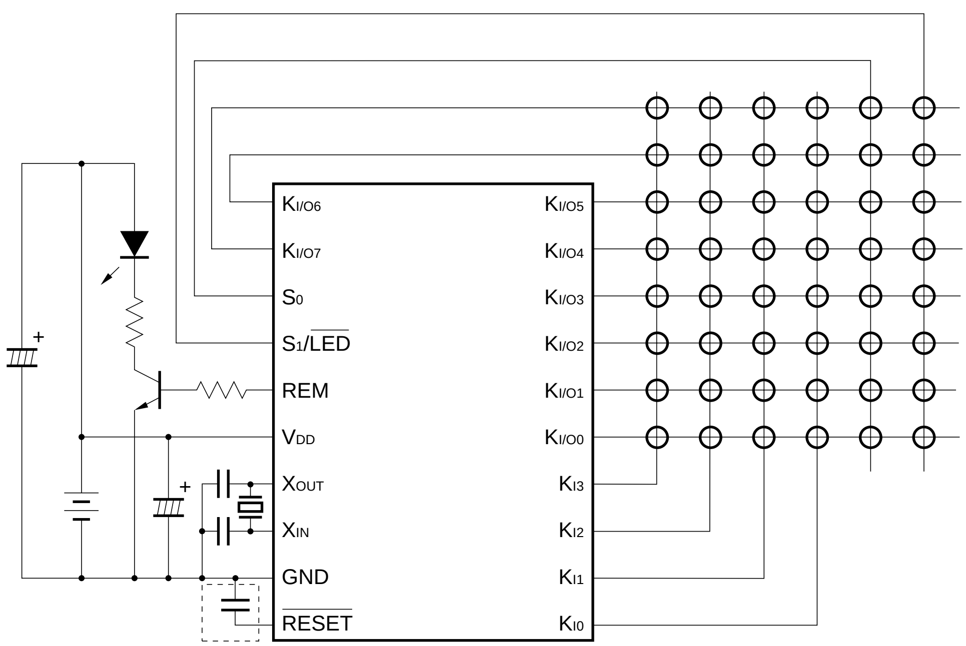
13. APPLIED CIRCUIT EXAMPLE

Example of Application to System

· **Remote-control transmitter (40 keys; mode selection switch accommodated)**



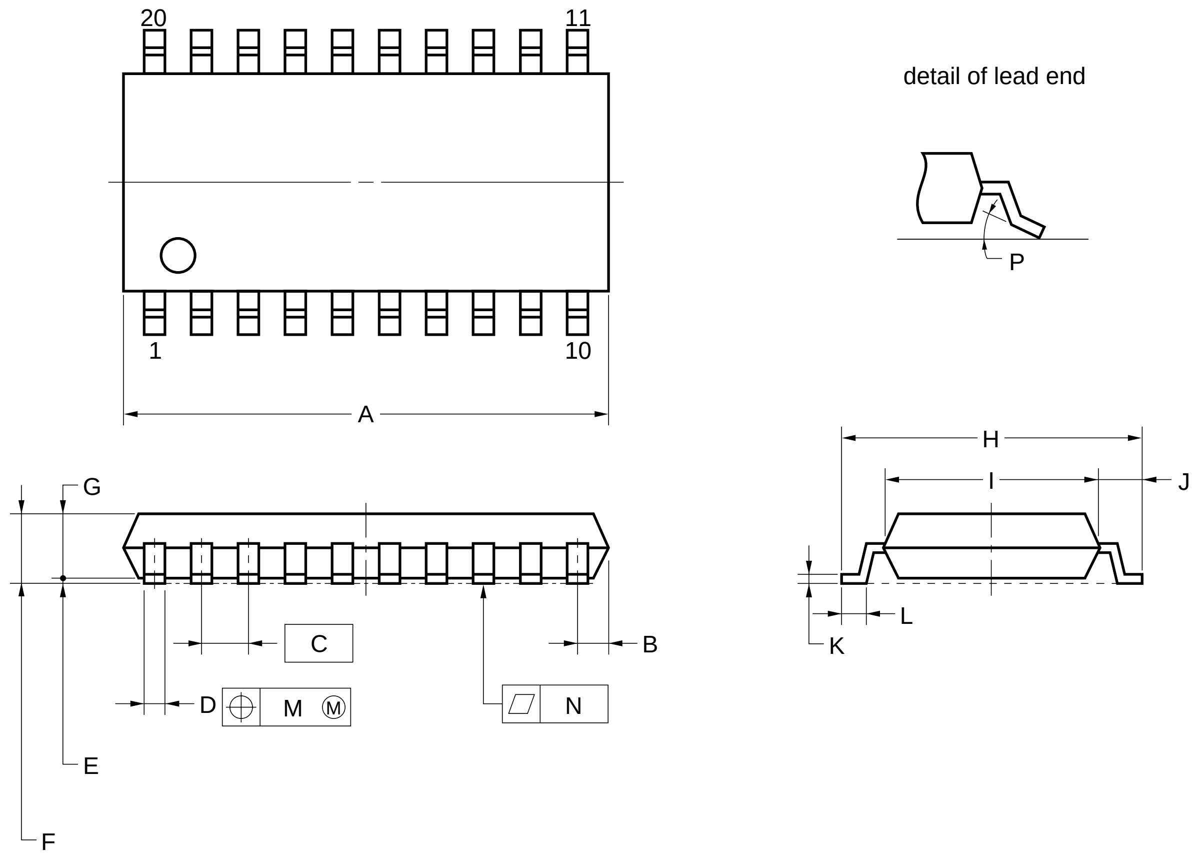
· **Remote-control transmitter (48 keys accommodated)**



Remark When the POC circuit of the mask option is used effectively, it is not necessary to connect the capacitor enclosed in the dotted lines.

14. PACKAGE DRAWINGS

20 PIN PLASTIC SOP (300 mil)



NOTE

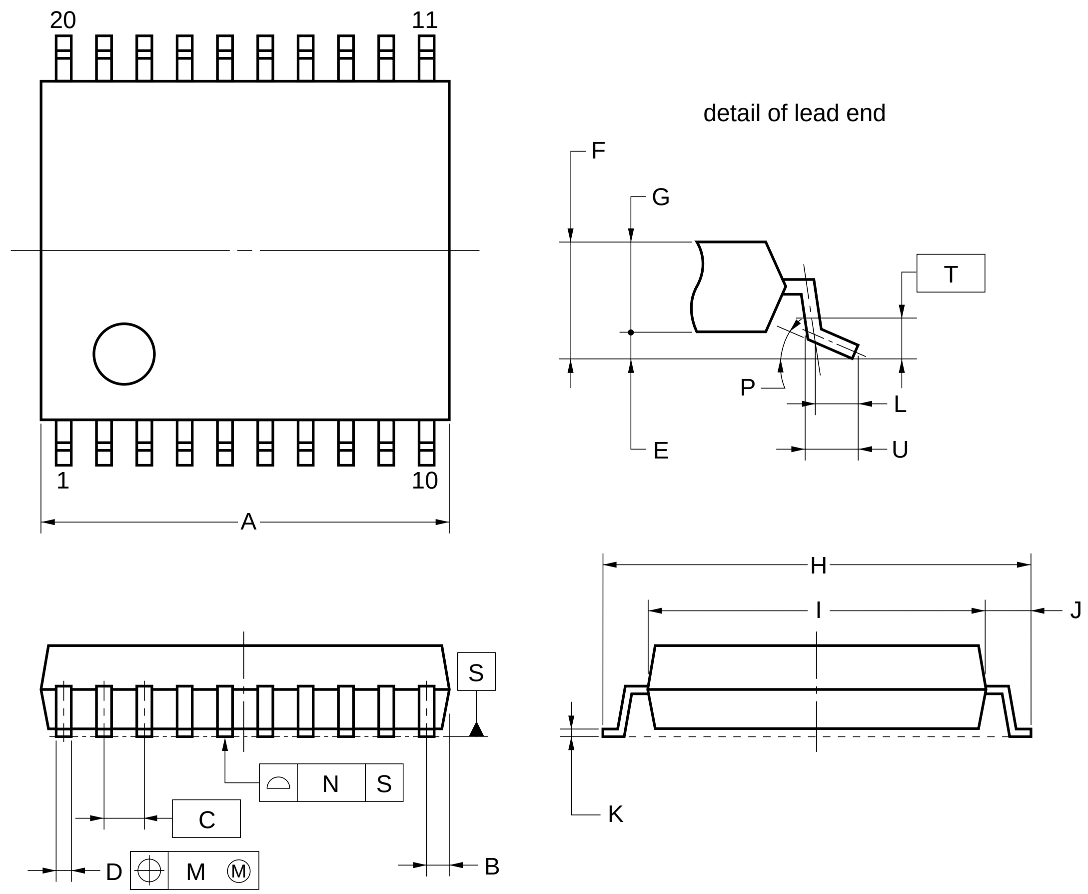
Each lead centerline is located within 0.12 mm (0.005 inch) of its true position (T.P.) at maximum material condition.

ITEM	MILLIMETERS	INCHES
A	12.7±0.3	0.500±0.012
B	0.78 MAX.	0.031 MAX.
C	1.27 (T.P.)	0.050 (T.P.)
D	$0.42^{+0.08}_{-0.07}$	$0.017^{+0.003}_{-0.004}$
E	0.1±0.1	0.004±0.004
F	1.8 MAX.	0.071 MAX.
G	1.55±0.05	0.061±0.002
H	7.7±0.3	0.303±0.012
I	5.6±0.2	$0.220^{+0.009}_{-0.008}$
J	1.1	0.043
K	$0.22^{+0.08}_{-0.07}$	$0.009^{+0.003}_{-0.004}$
L	0.6±0.2	$0.024^{+0.008}_{-0.009}$
M	0.12	0.005
N	0.10	0.004
P	$3°^{+7°}_{-3°}$	$3°^{+7°}_{-3°}$

P20GM-50-300B, C-5

Remark The dimensions and materials of the ES model are the same as those of mass production model.

★ 20 PIN PLASTIC SSOP (300 mil)



NOTE

Each lead centerline is located within 0.12 mm of its true position (T.P.) at maximum material condition.

ITEM	MILLIMETERS
A	6.65±0.15
B	0.475 MAX.
C	0.65 (T.P.)
D	$0.24^{+0.08}_{-0.07}$
E	0.1±0.05
F	1.3±0.1
G	1.2
H	8.1±0.2
I	6.1±0.2
J	1.0±0.2
K	0.17±0.03
L	0.5
M	0.13
N	0.10
P	$3°^{+5°}_{-3°}$
T	0.25
U	0.6±0.15

S20MC-65-5A4-1

Remark The dimensions and materials of the ES model are the same as those of mass production model.

15. RECOMMENDED SOLDERING CONDITIONS

Carry out the soldered packaging of this product under the following recommended conditions.

For details of the soldering conditions, refer to information material **Semiconductor Device Mounting Technology Manual (C10535E)**.

For soldering methods and conditions other than the recommended conditions, please consult one of our NEC sales representatives.

Table 15-1. Soldering Conditions for Surface-Mount Type

(1) μPD6133GS-×××: 20-pin plastic SOP (300 mil)
μPD6134GS-×××: 20-pin plastic SOP (300 mil)

Soldering Method	Soldering Condition	Recommended Condition Symbol
Infrared reflow	Package peak temperature: 235 °C; time: 30 secs. max. (210 °C min.); count: twice max.	IR35-00-2
VPS	Package peak temperature: 215 °C; time: 40 secs. max. (200 °C min.); count: twice max.	VP15-00-2
Wave soldering	Solder bath temperature: 260 °C max.; time: 10 secs. max.; count: once; Preliminary heat temperature: 120 °C max. (Package surface temperature)	WS60-00-1
Partial heating	Pin temperature: 300 ˚C or less ; time: 3 secs. max. (for each side of the device)	—

Caution Using more than one soldering method should be avoided (except in the case of partial heating).

★ **(2) μPD6134MC-×××-5A4: 20-pin plastic SSOP (300 mil)**

Soldering Method	Soldering Condition	Recommended Condition Symbol
Infrared reflow	Package peak temperature: 235 °C; time: 30 secs. max. (210 °C min.); count: three times max.	IR35-00-3
VPS	Package peak temperature: 215 °C; time: 40 secs. max. (200 °C min.); count: three times max.	VP15-00-3
Wave soldering	Solder bath temperature: 260 °C max.; time: 10 secs. max.; count: once; Preliminary heat temperature: 120 °C max. (Package surface temperature)	WS60-00-1
Partial heating	Pin temperature: 300 ˚C or less ; time: 3 secs. max. (for each side of the device)	—

Caution Using more than one soldering method should be avoided (except in the case of partial heating).

APPENDIX A. DEVELOPMENT TOOLS

An emulator is provided for the μPD6133.

Hardware

- **Emulator (EB-6133[Note])**
  It is used to emulate the μPD6133.

Note This is a product of Naito Densei Machida Mfg. Co., Ltd. For details, consult Naito Densei Machida Mfg. Co., Ltd. (044-822-3813).

Software

- **Assembler (AS6133)**
  - This is a development tool for remote control transmitter software.

Part Number List of AS6133

Host Machine	OS	Supply Medium	Part Number
PC-9800 series (CPU: 80386 or more)	MS-DOS™ (Ver. 5.0 to Ver. 6.2)	3.5-inch 2HD	μS5A13AS6133
IBM PC/AT™ compatible	MS-DOS (Ver. 6.0 to Ver. 6.22)	3.5-inch 2HC	μS7B13AS6133
	PC DOS™ (Ver. 6.1 to Ver. 6.3)		

Caution Although Ver.5.0 or later has a task swap function, this function cannot be used with this software.

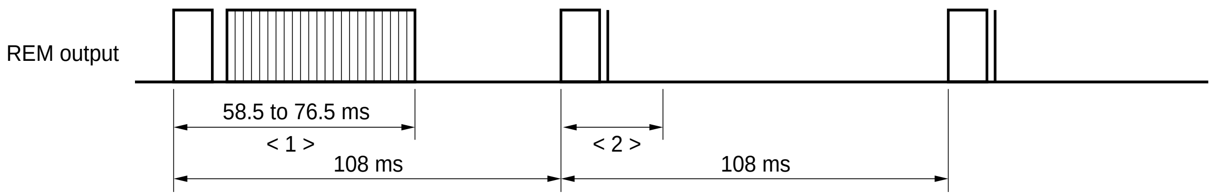
APPENDIX B. FUNCTIONAL COMPARISON BETWEEN μPD6133 SUBSERIES AND OTHER SUBSERIES

Item		μPD6133	μPD6134	μPD6600A	μPD6124A
ROM capacity		512 × 10 bits	1002 × 10 bits	512 × 10 bits	1002 × 10 bits
RAM capacity		32 × 4 bits		32 × 5 bits	
Stack		1 level (multiplexed with RF of RAM)		3 levels (multiplexed with RAM)	
Key matrix		8 × 6 = 48 keys		8 × 4 = 32 keys	
S_0 (S-IN) input		Read by P_{01} register (with function to release standby mode)		Read by left shift instruction	
$S_1/\overline{LED}$ (S-OUT)		I/O (with function to release standby mode)		Output	
Clock frequency		Ceramic oscillation • f_X = 300 kHz to 1 MHz • f_X = 300 to 500 kHz (with POC circuit)		Ceramic oscillation • f_X = 400 to 500 kHz	
Timer	Clock	$f_X/8$, $f_X/16$		$f_X/8$	
	Count start	Writing count value		Writing count value and P1 register value	
Carrier	Frequency	• f_X, $f_X/8$, $f_X/12$ (timer clock: $f_X/8$) • $f_X/2$, $f_X/16$, $f_X/24$ (timer clock: $f_X/16$) • No carrier		• $f_X/8$, $f_X/12$	
	Output start	Synchronized with timer		Not synchronized with timer	
Instruction execution time		8 μs (f_X = 1 MHz)		16 μs (f_X = 500 kHz)	
Relative branch instruction		None		Provided	
Left shift instrucion		None		Provided	
"MOV Rn, @R0" instrucion		n = 1 to F		n = 0 to F	
Standby mode (HALT instruction)		HALT mode for timer only. STOP mode for only releasing K_I ($K_{I/O}$ high-level output or $K_{I/O0}$ high-level output)		HALT/STOP mode set by P1 register value	
Relation between HALT instruction execution and status flag (F)		HALT instruction not executed when F = 1		HALT instrucion executed regardless of status of F	
Reset function by charging/ discharging capacitor		None		Provided	
POC circuit		Mask option Low level output to $\overline{RESET}$ pin on detection		Provided (low-voltage detection circuit) Low level ouput to S-OUT pin on detection	
Mask option		POC circuit only (Circuits other than POC circuit are set by software.)		• Pull-down resistor • Variable duty • Hang-up detection	
Supply voltage		V_{DD} = 1.8 to 3.6 V		• V_{DD} = 2.2 to 3.6 V	• V_{DD} = 2.2 to 5.5 V
Operating temperature		• T_A = –40 to +85 °C • T_A = –20 to +70 °C (with POC circuit)		• T_A = –20 to +70 °C	
★ Package		• 20-pin plastic SOP	• 20-pin plastic SOP • 20-pin plastic SSOP	• 20-pin plastic SOC • 20-pin plastic shrink DIP	
One-time PROM product		μPD61P34B		μPD61P24	

APPENDIX C. EXAMPLE OF REMOTE-CONTROL TRANSMISSION FORMAT
(in the case of NEC transmission format in command one-shot transmission mode)

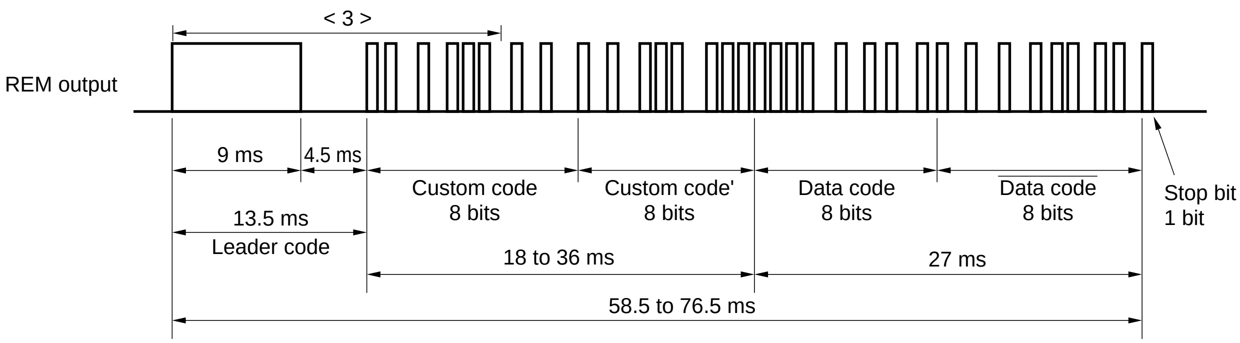
Caution **When using the NEC transmission format, please apply for a custom code at NEC.**

(1) REM output waveform (From <2> on, the output is made only when the key is kept pressed.)

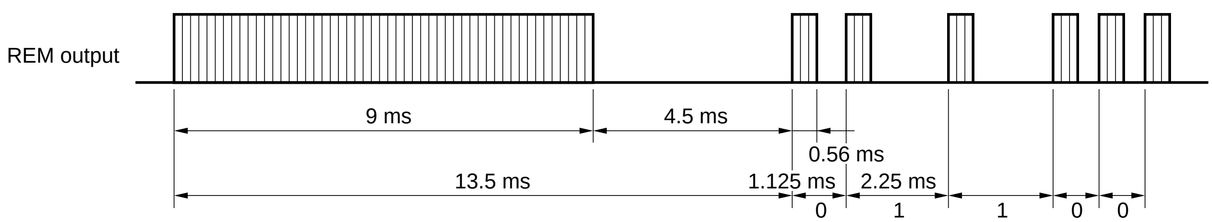


Remark If the key is repeatedly pressed, the power consumption of the infrared light-emitting diode (LED) can be reduced by sending the reader code and the stop bit from the second time.

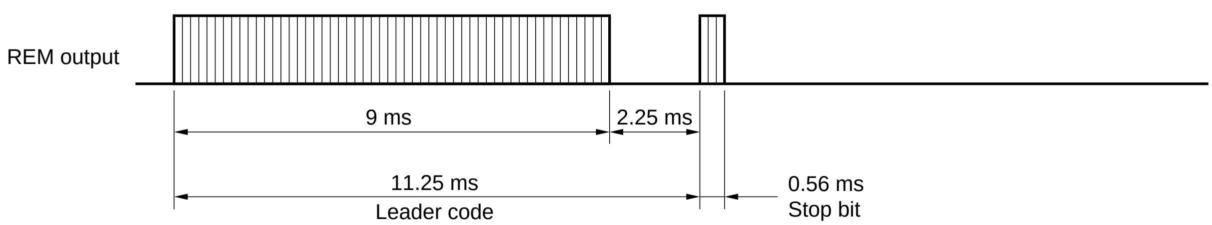
(2) Enlarged waveform of <1>



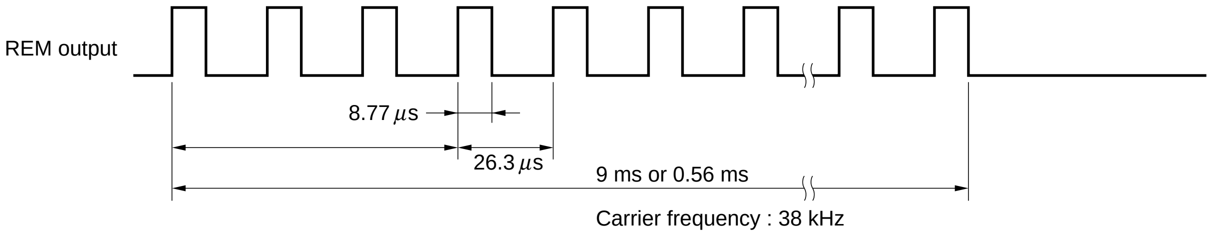
(3) Enlarged waveform of <3>



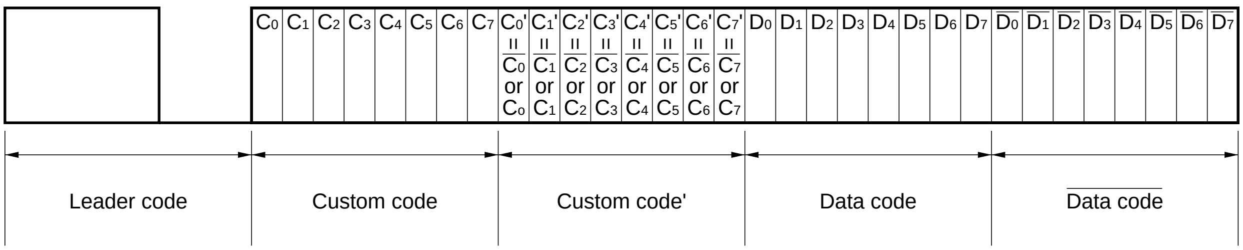
(4) Enlarged waveform of <2>



(5) Carrier waveform (Enlarged waveform of each code's high period)



(6) Bit array of each code



Caution **To prevent malfunction with other systems when receiving data in the NEC transmission format, not only fully decode (make sure to check $\overline{\text{Data Code}}$ as well) the total 32 bits of the 16-bit custom codes (Custom Code, Custom Code') and the 16-bit data codes (Data Code, $\overline{\text{Data Code}}$) but also check to make sure that no signals are present.**