Welcome to E-XFL.COM

What is "Embedded - Microcontrollers"?

"Embedded - Microcontrollers" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" because they are embedded within electronic devices to control various functions, rather than serving as standalone computers. Microcontrollers are crucial in modern electronics, providing the intelligence and control needed for a wide range of applications.

Applications of "Embedded - Microcontrollers"

Details	
Product Status	Obsolete
Core Processor	740
Core Size	8-Bit
Speed	4MHz
Connectivity	-
Peripherals	LED, LVD, POR, WDT
Number of I/O	25
Program Memory Size	16KB (16K x 8)
Program Memory Type	QzROM
EEPROM Size	-
RAM Size	512 x 8
Voltage - Supply (Vcc/Vdd)	1.8V ~ 3.6V
Data Converters	-
Oscillator Type	External, Internal
Operating Temperature	-20°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	32-LQFP
Supplier Device Package	32-LQFP (7x7)
Purchase URL	https://www.e-xfl.com/product-detail/renesas-electronics-america/m37545g4gp-u0

Notice

1. All information included in this document is current as of the date this document is issued. Such information, however, is subject to change without any prior notice. Before purchasing or using any Renesas Electronics products listed herein, please confirm the latest product information with a Renesas Electronics sales office. Also, please pay regular and careful attention to additional and different information to be disclosed by Renesas Electronics such as that disclosed through our website.
2. Renesas Electronics does not assume any liability for infringement of patents, copyrights, or other intellectual property rights of third parties by or arising from the use of Renesas Electronics products or technical information described in this document. No license, express, implied or otherwise, is granted hereby under any patents, copyrights or other intellectual property rights of Renesas Electronics or others.
3. You should not alter, modify, copy, or otherwise misappropriate any Renesas Electronics product, whether in whole or in part.
4. Descriptions of circuits, software and other related information in this document are provided only to illustrate the operation of semiconductor products and application examples. You are fully responsible for the incorporation of these circuits, software, and information in the design of your equipment. Renesas Electronics assumes no responsibility for any losses incurred by you or third parties arising from the use of these circuits, software, or information.
5. When exporting the products or technology described in this document, you should comply with the applicable export control laws and regulations and follow the procedures required by such laws and regulations. You should not use Renesas Electronics products or the technology described in this document for any purpose relating to military applications or use by the military, including but not limited to the development of weapons of mass destruction. Renesas Electronics products and technology may not be used for or incorporated into any products or systems whose manufacture, use, or sale is prohibited under any applicable domestic or foreign laws or regulations.
6. Renesas Electronics has used reasonable care in preparing the information included in this document, but Renesas Electronics does not warrant that such information is error free. Renesas Electronics assumes no liability whatsoever for any damages incurred by you resulting from errors in or omissions from the information included herein.
7. Renesas Electronics products are classified according to the following three quality grades: "Standard", "High Quality", and "Specific". The recommended applications for each Renesas Electronics product depends on the product's quality grade, as indicated below. You must check the quality grade of each Renesas Electronics product before using it in a particular application. You may not use any Renesas Electronics product for any application categorized as "Specific" without the prior written consent of Renesas Electronics. Further, you may not use any Renesas Electronics product for any application for which it is not intended without the prior written consent of Renesas Electronics. Renesas Electronics shall not be in any way liable for any damages or losses incurred by you or third parties arising from the use of any Renesas Electronics product for an application categorized as "Specific" or for which the product is not intended where you have failed to obtain the prior written consent of Renesas Electronics. The quality grade of each Renesas Electronics product is "Standard" unless otherwise expressly specified in a Renesas Electronics data sheets or data books, etc.

   "Standard": Computers; office equipment; communications equipment; test and measurement equipment; audio and visual equipment; home electronic appliances; machine tools; personal electronic equipment; and industrial robots.

   "High Quality": Transportation equipment (automobiles, trains, ships, etc.); traffic control systems; anti-disaster systems; anti-crime systems; safety equipment; and medical equipment not specifically designed for life support.

   "Specific": Aircraft; aerospace equipment; submersible repeaters; nuclear reactor control systems; medical equipment or systems for life support (e.g. artificial life support devices or systems), surgical implantations, or healthcare intervention (e.g. excision, etc.), and any other applications or purposes that pose a direct threat to human life.
8. You should use the Renesas Electronics products described in this document within the range specified by Renesas Electronics, especially with respect to the maximum rating, operating supply voltage range, movement power voltage range, heat radiation characteristics, installation and other product characteristics. Renesas Electronics shall have no liability for malfunctions or damages arising out of the use of Renesas Electronics products beyond such specified ranges.
9. Although Renesas Electronics endeavors to improve the quality and reliability of its products, semiconductor products have specific characteristics such as the occurrence of failure at a certain rate and malfunctions under certain use conditions. Further, Renesas Electronics products are not subject to radiation resistance design. Please be sure to implement safety measures to guard them against the possibility of physical injury, and injury or damage caused by fire in the event of the failure of a Renesas Electronics product, such as safety design for hardware and software including but not limited to redundancy, fire control and malfunction prevention, appropriate treatment for aging degradation or any other appropriate measures. Because the evaluation of microcomputer software alone is very difficult, please evaluate the safety of the final products or system manufactured by you.
10. Please contact a Renesas Electronics sales office for details as to environmental matters such as the environmental compatibility of each Renesas Electronics product. Please use Renesas Electronics products in compliance with all applicable laws and regulations that regulate the inclusion or use of controlled substances, including without limitation, the EU RoHS Directive. Renesas Electronics assumes no liability for damages or losses occurring as a result of your noncompliance with applicable laws and regulations.
11. This document may not be reproduced or duplicated, in any form, in whole or in part, without prior written consent of Renesas Electronics.
12. Please contact a Renesas Electronics sales office if you have any questions regarding the information contained in this document or Renesas Electronics products, or if you have any other inquiries.

(Note 1) "Renesas Electronics" as used in this document means Renesas Electronics Corporation and also includes its majority-owned subsidiaries.

(Note 2) "Renesas Electronics product(s)" means any product developed or manufactured by or for Renesas Electronics.

Table 1 Performance overview (1)

Parameter			Function
Number of basic instructions			71
Instruction execution time			2.00 μs (Minimum instruction)
Memory sizes	ROM	M37545G1	4096 bytes × 8 bits
		M37545G2	8192 bytes × 8 bits
		M37545G4	16384 bytes × 8 bits
		M37545G6	24576 bytes × 8 bits
		M37545G8	32768 bytes × 8 bits
		M37545GC	49152 bytes × 8 bits
		M37545GF	61440 bytes × 8 bits
	RAM	M37545G1/G2	RAM1: 240 bytes × 8 bits, RAM2: 16 bytes × 8 bits
		M37545G4/G6/G8/GC/GF	RAM1: 384 bytes × 8 bits, RAM2: 128 bytes × 8 bits
I/O port	$P0_0$–$P0_7$	I/O	• 1-bit × 8 • CMOS compatible input level • CMOS 3-state output structure • Whether the pull-up function/key-on wakeup function is to be used or not can be determined by program.
	$P1_0$, $P1_1$	I/O (RLSS-only pin)	• 1-bit × 2 • CMOS compatible input level • The output structure can be switched to N-channel open-drain or CMOS by software.
	$P2_0$–$P2_7$	I/O	• 1-bit × 8 • CMOS compatible input level • The output structure can be switched to N-channel open-drain or CMOS by software. • P2 can output a large current for driving LED. • $P2_0$ and $P2_1$ are also used as INT_0 and INT_1, respectively.
	$P3_0$–$P3_7$	I/O	• 1-bit × 8 • CMOS compatible input level • The output structure can be switched to N-channel open-drain or CMOS by software.
	$P4_0$, $P4_1$	I/O (RLSS-only pin)	• 1-bit × 2 • CMOS compatible input level • CMOS 3-state output structure
	$P4_2$	I/O	• 1-bit × 1 • CMOS compatible input level • CMOS 3-state output structure • Carrier wave output pin for remote-control transmitter
Timer	Timer 1		8-bit timer with timer 1 latch Count source is Prescaler output.
	Timer 2		8-bit timer with timer 2 primary latch and timer 2 secondary latch Count source can be selected from $f(X_{IN})/16$, $f(X_{IN})/8$, $f(X_{IN})/2$ or $f(X_{IN})/1$.
	Timer 3		8-bit timer with timer 3 latch Count source can be selected from $f(X_{IN})/16$, $f(X_{IN})/8$ or $f(X_{IN})/2$ or carrier wave output.
Carrier wave generating circuit			Remote-control waveform is generated by using timer 2 and timer 3. 455 kHz carrier wave generating mode is available.
Watchdog timer			16-bit × 1
Power-on reset circuit			Built-in
Voltage drop detection circuit (Not available for RLSS)			Typ. 1.75 V (Ta=25 °C)
Interrupt	Source		7 sources (External × 3, Timer × 3, Software)
Function set ROM area	Function set ROM		Function set ROM is assigned to address $FFDA_{16}$. Enable/disable of watchdog timer and STP instruction can be selected. Valid/invalid of voltage drop detection circuit can be selected.
	ROM code protect		ROM code protect is assigned to address $FFDB_{16}$. Read/write the built-in QzROM by serial programmer is disabled by setting "00" to ROM code protect.
Device structure			CMOS silicon gate
Package			32-pin plastic molded LQFP (PLQP0032GB-A) 32-pin plastic molded SSOP (PLSP0032JB-A)
Operating temperature range			−20 to 85 °C
Power source voltage	$f(X_{IN})$ = 4 MHz		1.8 to 3.6 V

FUNCTIONAL DESCRIPTION

Central Processing Unit (CPU)

The MCU uses the standard 740 family instruction set. Refer to the table of 740 family addressing modes and machine-language instructions or the SERIES 740 <SOFTWARE> USER'S MANUAL for details on each instruction set.
Machine-resident 740 family instructions are as follows:

1. The FST and SLW instructions cannot be used.
2. The MUL and DIV instructions can be used.
3. The WIT instruction can be used.
4. The STP instruction can be used.

This instruction cannot be used while CPU operates by an on-chip oscillator.

[Accumulator (A)]

The accumulator is an 8-bit register. Data operations such as data transfer, etc., are executed mainly through the accumulator.

[Index register X (X), Index register Y (Y)]

Both index register X and index register Y are 8-bit registers. In the index addressing modes, the value of the OPERAND is added to the contents of register X or register Y and specifies the real address.
When the T flag in the processor status register is set to "1", the value contained in index register X becomes the address for the second OPERAND.

[Stack pointer (S)]

The stack pointer is an 8-bit register used during subroutine calls and interrupts. The stack is used to store the current address data and processor status when branching to subroutines or interrupt routines.
The lower eight bits of the stack address are determined by the contents of the stack pointer. The upper eight bits of the stack address are determined by the Stack Page Selection Bit. If the Stack Page Selection Bit is "0", then the RAM in the zero page is used as the stack area. If the Stack Page Selection Bit is "1", then RAM in page 1 is used as the stack area.
The Stack Page Selection Bit is located in the SFR area in the zero page. Note that the initial value of the Stack Page Selection Bit varies with each microcomputer type. Also some microcomputer types have no Stack Page Selection Bit and the upper eight bits of the stack address are fixed. The operations of pushing register contents onto the stack and popping them from the stack are shown in Figure 8.

[Program counter (PC)]

The program counter is a 16-bit counter consisting of two 8-bit registers PC_H and PC_L. It is used to indicate the address of the next instruction to be executed.

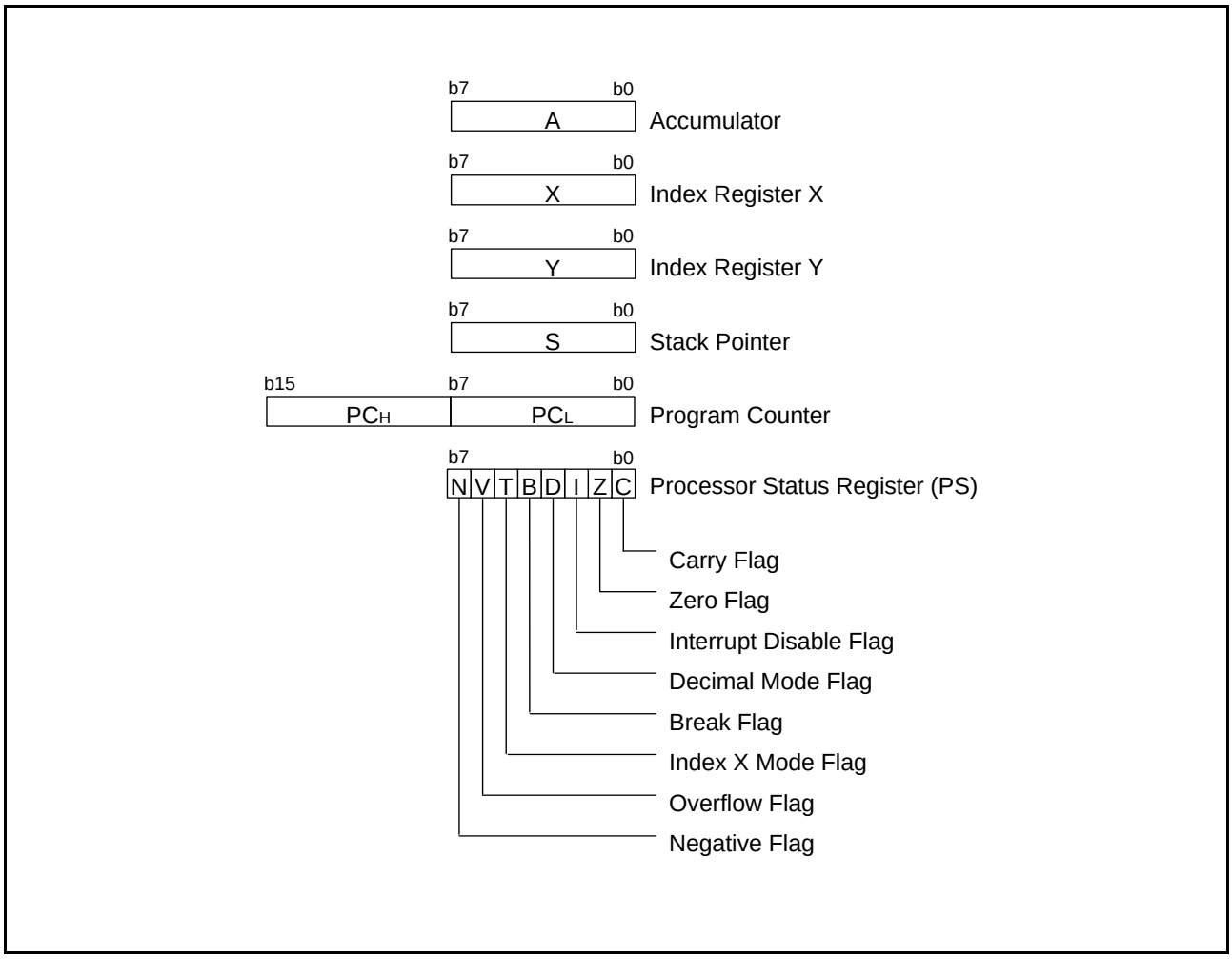


Fig. 7 740 Family CPU register structure

[CPU mode register (CPUM)]

The CPU mode register contains the stack page selection bit.
This register is allocated at address 003B16.
For this product, the clock speed of CPU is always f(XIN)/4.

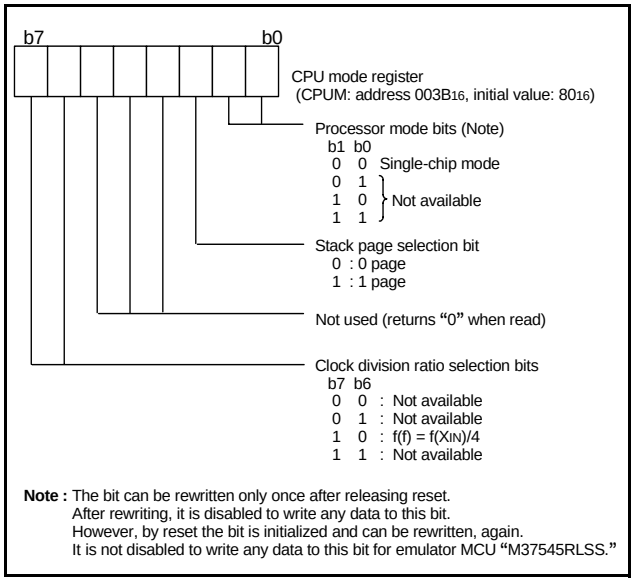


Fig. 9 Structure of CPU mode register

MEMORY

Special Function Register (SFR) Area

The SFR area in the zero page contains control registers such as I/O ports and timers.

RAM

RAM is used for data storage and for a stack area of subroutine calls and interrupts. RAM consists of RAM1 and RAM2. The power source for RAM1 is supplied from V_{CC} pin. The power source for RAM2 is supplied from V_{DDR} pin.

Note: When the V_{DDR} pin is used, connect an approximately 0.1 μF bypass capacitor across the V_{SS} line and the V_{DDR} line. When not used, connect it to V_{SS}.

ROM

The first 128 bytes and the last 2 bytes of ROM are reserved for device testing and the rest is a user area for storing programs.

Interrupt Vector Area

The interrupt vector area contains reset and interrupt vectors.

Zero Page

The 256 bytes from addresses 0000_{16} to $00FF_{16}$ are called the zero page area. The internal RAM and the special function registers (SFR) are allocated to this area.
The zero page addressing mode can be used to specify memory and register addresses in the zero page area. Access to this area with only 2 bytes is possible in the zero page addressing mode.

Special Page

The 256 bytes from addresses $FF00_{16}$ to $FFFF_{16}$ are called the special page area. The special page addressing mode can be used to specify memory addresses in the special page area. Access to this area with only 2 bytes is possible in the special page addressing mode.

Function Set ROM Area

[Renesas shipment test area]

Figure 10 shows the Assignment of Function set ROM area.
The random data are set to the Renesas shipment test areas (addresses $FFD4_{16}$ to address $FFD9_{16}$).
Do not rewrite the data of these areas.
When the checksum is included in the user program, avoid assigning it to these areas.

[Function set ROM data] FSROM

Function set ROM data (address $FFDA_{16}$) is used to set modes of peripheral functions. By setting this area, the operation mode of each peripheral function are set after system is released from reset.
Refer to the descriptions of peripheral functions for the details of operation of peripheral functions.

- Watchdog timer
- Low voltage detection circuit

This mode setting of peripheral functions cannot be changed by program after system is released from reset.

ROM Code Protect Address (address $FFDB_{16}$)

Address $FFDB_{16}$, which is the reserved ROM area of QzROM, is the ROM code protect address. "00_{16}" is written into this address when selecting the protect bit write by using a serial programmer or selecting protect enabled for writing shipment by Renesas Technology corp.. When "00_{16}" is set to the ROM code protect address, the protect function is enabled, so that reading or writing from/to QzROM is disabled by a serial programmer.
As for the QzROM product in blank, the ROM code is protected by selecting the protect bit write at ROM writing with a serial programmer.
As for the QzROM product shipped after writing, "00_{16}" (protect enabled) or "FF_{16}" (protect disabled) is written into the ROM code protect address when Renesas Technology corp. performs writing.
The writing of "00_{16}" or "FF_{16}" can be selected as the ROM option setup (referred to as "Mask option setup" in MM) when ordering.

<Notes>

1. Because the contents of RAM are indefinite at reset, set initial values before using.
2. Do not access to the reserved area.
3. Random data is written into the Renesas shipment test area and the reserved ROM area. Do not rewrite the data in these areas. Data of these area may be changed without notice. Accordingly, do not include these areas into programs such as checksum of all ROM areas.
4. The QzROM values in function set ROM data set the operating modes of the various peripheral functions after an MCU reset is released. Do not fail to set the value for the selected function. Bits designated with a fixed value of 1 or 0 must be set to the designated value.
5. Emulator MCU: As for M37545RLSS, set "$010000XX_2$" to Function set ROM data (address $FFDA_{16}$). Also, set "FF_{16}" to ROM code protect (address $FFDB_{16}$).

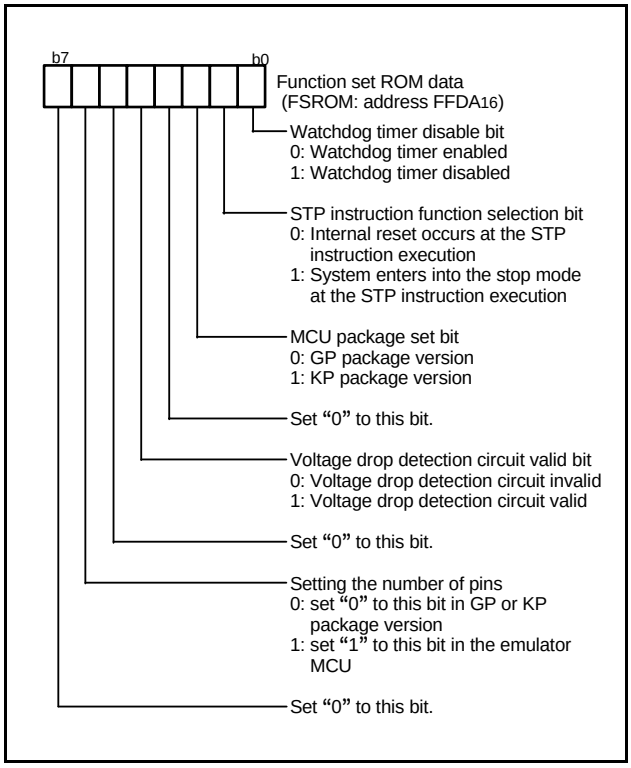


Fig. 12 Structure of Function set ROM area

I/O PORTS

[Direction registers] PiD

The I/O ports have direction registers which determine the input/output direction of each pin. Each bit in a direction register corresponds to one pin, and each pin can be set to be input or output.

When "1" is set to the bit corresponding to a pin, this pin becomes an output port. When "0" is set to the bit, the pin becomes an input port.

When data is read from a pin set to output, not the value of the pin itself but the value of port latch is read. Pins set to input are floating, and permit reading pin values.

If a pin set to input is written to, only the port latch is written to and the pin remains floating.

[Pull-up control register] Pull

By setting the pull-up control register (address 0016_{16}), port P0 can exert pull-up control by program. However, pins set to output are disconnected from this control and cannot exert pull-up control.

[Port output mode selection register] PMOD

By setting the port output mode selection register (address 0017_{16}), CMOS output or N-channel open-drain can be selected for ports P1, P2, P3 by program.

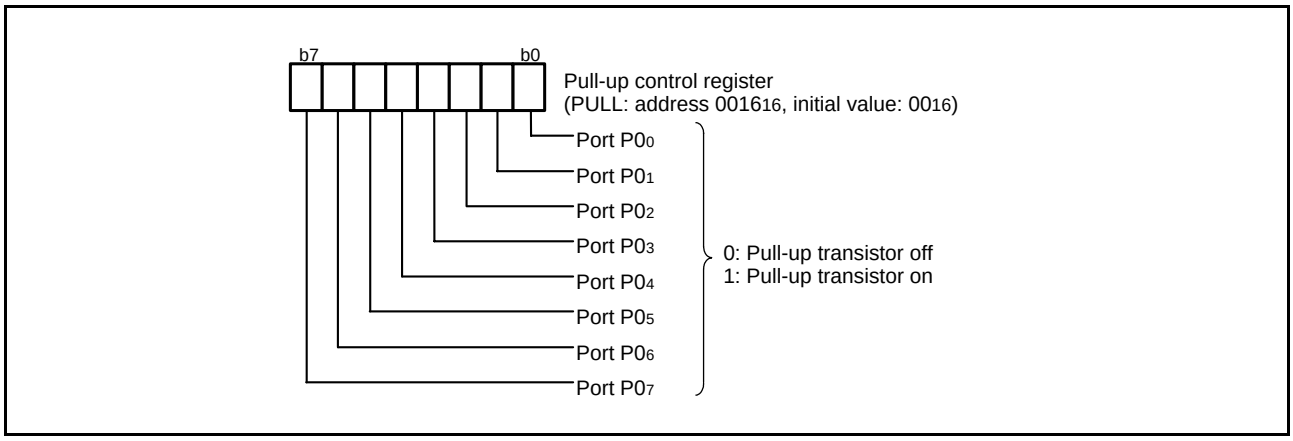


Fig. 13 Structure of pull-up control register

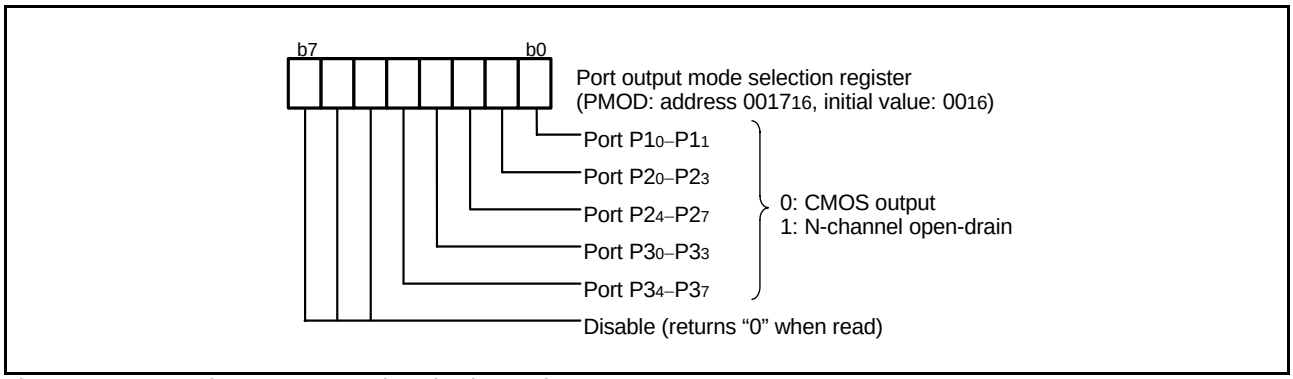


Fig. 14 Structure of port output mode selection register

Table 7 I/O port function table

Pin	Name	Input/Output	I/O format	Non-port function	Related SFRs	Diagram No.
P00–P07	Port P0	I/O individual bits	• CMOS compatible input level • CMOS 3-state output	Key input interrupt	Pull-up control register Key-on wakeup pin selection register Key-on wakeup edge selection register	(1)
P10–P11	Port P1		• CMOS compatible input level • CMOS 3-state output or N-channel open drain	RLSS-only pin	Port output mode selection register	(2)
P20/INT0 P21/INT1	Port P2			External interrupt input	Interrupt edge selection register Port output mode selection register	(3)
P22–P27					Port output mode selection register	(2)
P30–P37	Port P3				Port output mode selection register	(2)
P40, P41	Port P4		• CMOS compatible input level • CMOS 3-state output	RLSS-only pin		(4)
P42/CARR				Carrier wave output for remote-control transmitter	Carrier wave control register	(5)

Interrupts

The 7545 group interrupts are vector interrupts with a fixed priority scheme, and generated by 7 sources 3 external, 3 internal, and 1 software.
The interrupt sources, vector addresses(1), and interrupt priority are shown in Table 9.

Each interrupt except the BRK instruction interrupt has the interrupt request bit and the interrupt enable bit. These bits and the interrupt disable flag (I flag) control the acceptance of interrupt requests. Figure 16 shows an interrupt control diagram.

An interrupt requests is accepted when all of the following conditions are satisfied:

- Interrupt disable flag ..... "0"
- Interrupt request bit ..... "1"
- Interrupt enable bit ..... "1"

Though the interrupt priority is determined by hardware, priority processing can be performed by software using the above bits and flag.

Table 9 Interrupt vector address and priority

Interrupt source	Priority	Vector addresses(1)		Interrupt request generating conditions	Remarks
		High-order	Low-order		
Reset (2)	1	$FFFD_{16}$	$FFFC_{16}$	At reset input	Non-maskable
Key-on wakeup	2	$FFFB_{16}$	$FFFA_{16}$	AND operation of input logic level of port P0 (input)	External interrupt
INT_0	3	$FFF9_{16}$	$FFF8_{16}$	At detection of either rising or falling edge of INT_0 input	External interrupt (active edge selectable)
INT_1	4	$FFF7_{16}$	$FFF6_{16}$	At detection of either rising or falling edge of INT_1 input	External interrupt (active edge selectable)
Timer 2	5	$FFF5_{16}$	$FFF4_{16}$	At timer 2 underflow	
Timer 3	6	$FFF3_{16}$	$FFF2_{16}$	At timer 3 underflow	
Timer 1	7	$FFF1_{16}$	$FFF0_{16}$	At timer 1 underflow	STP release timer underflow
BRK instruction	8	$FFDD_{16}$	$FFDC_{16}$	At BRK instruction execution	Non-maskable software interrupt

NOTES:
1. Vector addressed contain interrupt jump destination addresses.
2. Reset function in the same way as an interrupt with the highest priority.

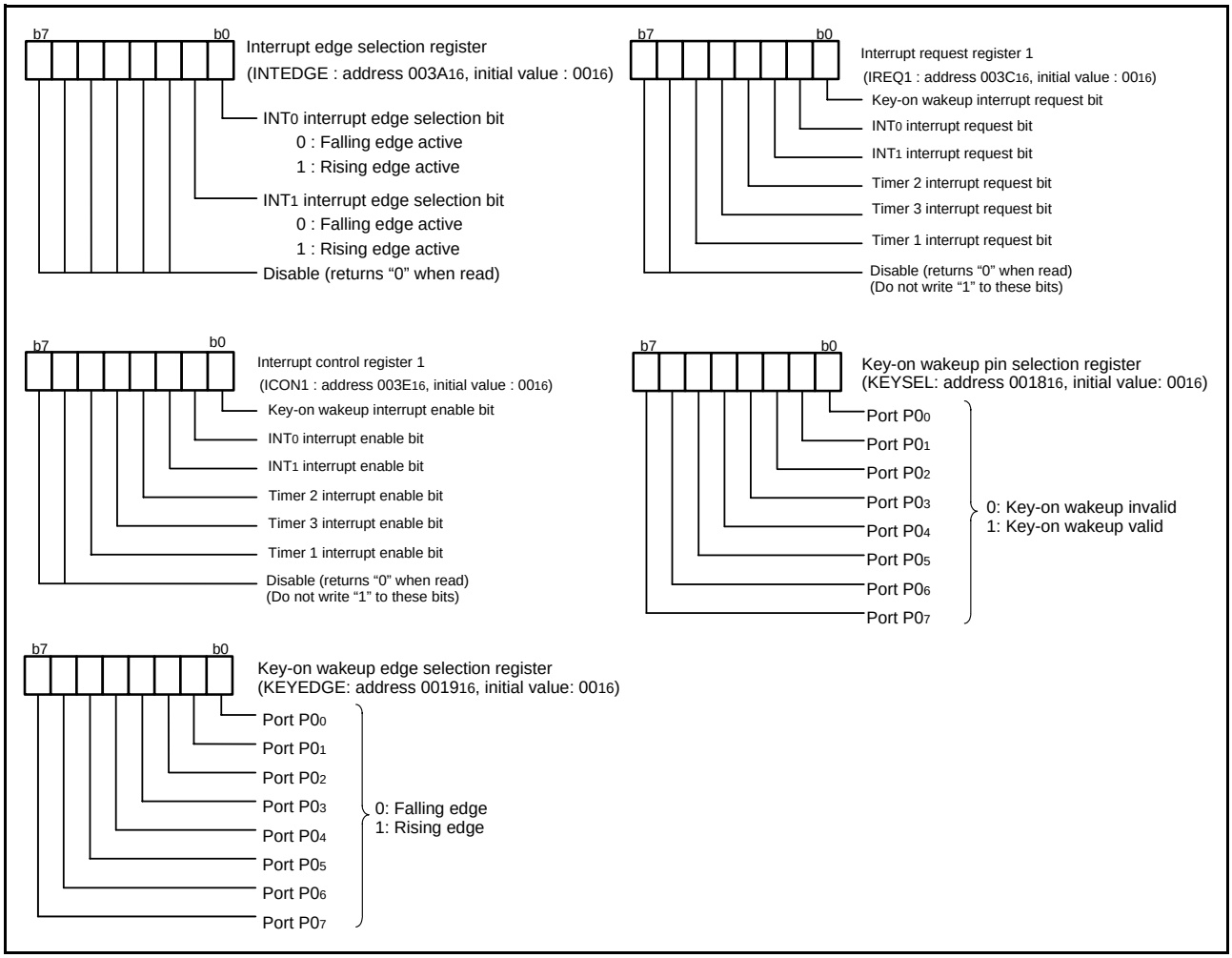


Fig. 17 Structure of interrupt-related registers

• Interrupt Request Generation, Acceptance, and Handling

Interrupts have the following three phases.

(i) Interrupt Request Generation
An interrupt request is generated by an interrupt source (external interrupt signal input, timer underflow, etc.) and the corresponding request bit is set to "1".

(ii) Interrupt Request Acceptance
Based on the interrupt acceptance timing in each instruction cycle, the interrupt control circuit determines acceptance conditions (interrupt request bit, interrupt enable bit, and interrupt disable flag) and interrupt priority levels for accepting interrupt requests. When two or more interrupt requests are generated simultaneously, the highest priority interrupt is accepted. The value of interrupt request bit for an unaccepted interrupt remains the same and acceptance is determined at the next interrupt acceptance timing point.

(iii) Handling of Accepted Interrupt Request
The accepted interrupt request is processed.

Figure 18 shows the time up to execution in the interrupt processing routine, and Figure 19 shows the interrupt sequence. Figure 20 shows the timing of interrupt request generation, interrupt request bit, and interrupt request acceptance.

• Interrupt Handling Execution

When interrupt handling is executed, the following operations are performed automatically.

(1) Once the currently executing instruction is completed, an interrupt request is accepted.

(2) The contents of the program counters and the processor status register at this point are pushed onto the stack area in order from 1 to 3.
1. High-order bits of program counter (PC_H)
2. Low-order bits of program counter (PC_L)
3. Processor status register (PS)

(3) Concurrently with the push operation, the jump address of the corresponding interrupt (the start address of the interrupt processing routine) is transferred from the interrupt vector to the program counter.

(4) The interrupt request bit for the corresponding interrupt is set to "0". Also, the interrupt disable flag is set to "1" and multiple interrupts are disabled.

(5) The interrupt routine is executed.

(6) When the RTI instruction is executed, the contents of the registers pushed onto the stack area are popped off in the order from 3 to 1. Then, the routine that was before running interrupt processing resumes.

As described above, it is necessary to set the stack pointer and the jump address in the vector area corresponding to each interrupt to execute the interrupt processing routine.

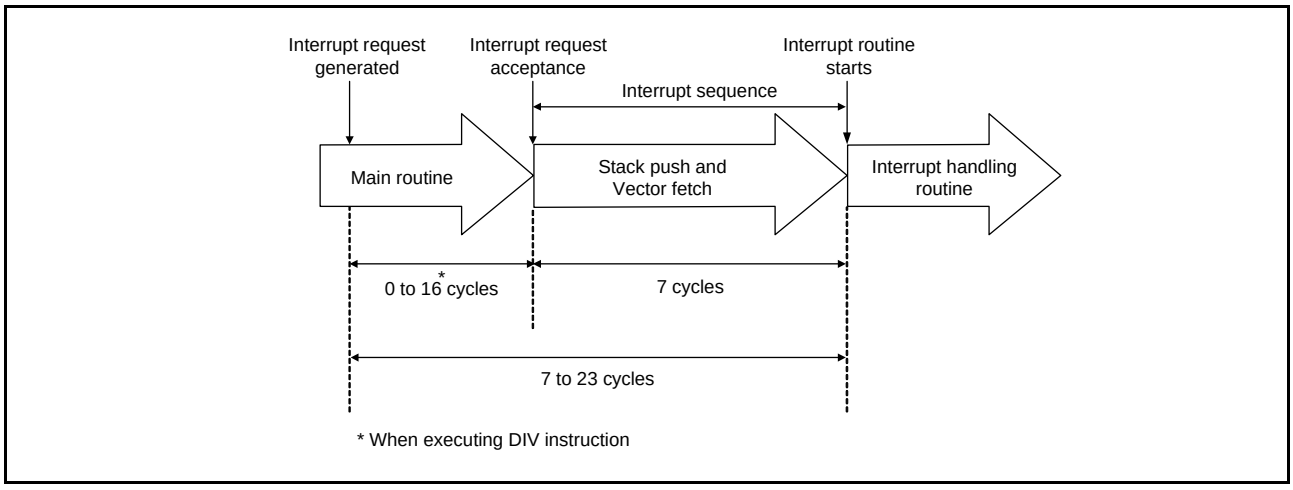
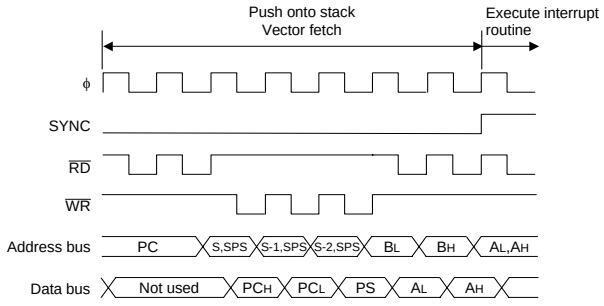
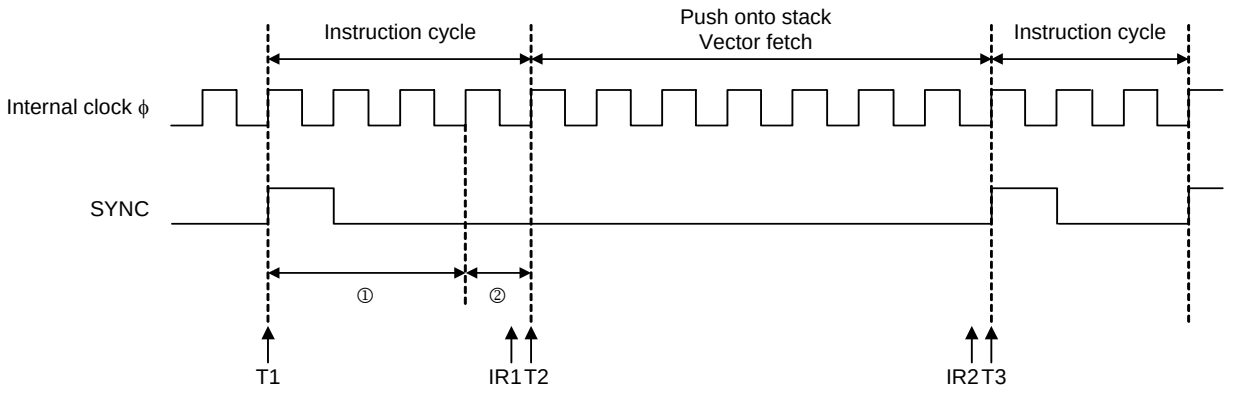


Fig. 18 Time up to execution in interrupt routine



SYNC : CPU operation code fetch cycle
(This is an internal signal that cannot be observed from the external unit.)
BL, BH: Vector address of each interrupt
AL, AH: Jump destination address of each interrupt
SPS : "00$_{16}$" or "01$_{16}$"
([SPS] is a page selected by the stack page selection bit of CPU mode register.)

Fig. 19 Interrupt sequence



T1 T2 T3 : Interrupt acceptance timing points
IR1 IR2 : Timings points at which the interrupt request bit is set to "1".

Note : Period 2 indicates the last ϕ cycle during one instruction cycle.

(1) The interrupt request bit for an interrupt request generated during period 1 is set to "1" at timing point IR1.
(2) The interrupt request bit for an interrupt request generated during period 2 is set to "1" at timing point IR1 or IR2. The timing point at which the bit is set to "1" varies depending on conditions. When two or more interrupt requests are generated during the period 2, each request bit may be set to "1" at timing point IR1 or IR2 separately.

Fig. 20 Timing of interrupt request generation, interrupt request bit, and interrupt acceptance

<Notes>

When setting the followings, the interrupt request bit may be set to "1" .

<When setting the external interrupt active edge>

- INT$_0$ interrupt edge selection bit (bit 0 of Interrupt edge selection register (address 3A$_{16}$))
- INT$_1$ interrupt edge selection bit (bit 1 of Interrupt edge selection register)
- Key-on wakeup edge selection register (address 19$_{16}$)

If it is not necessary to generate an interrupt synchronized with these settings, take the following sequence.

(1) Set the corresponding enable bit to "0" (disabled).
(2) Set the interrupt edge selection bit (the active edge switch bit) or the interrupt source bit.
(3) Set the corresponding interrupt request bit to "0" after one or more instructions have been executed.
(4) Set the corresponding interrupt enable bit to "1" (enabled).

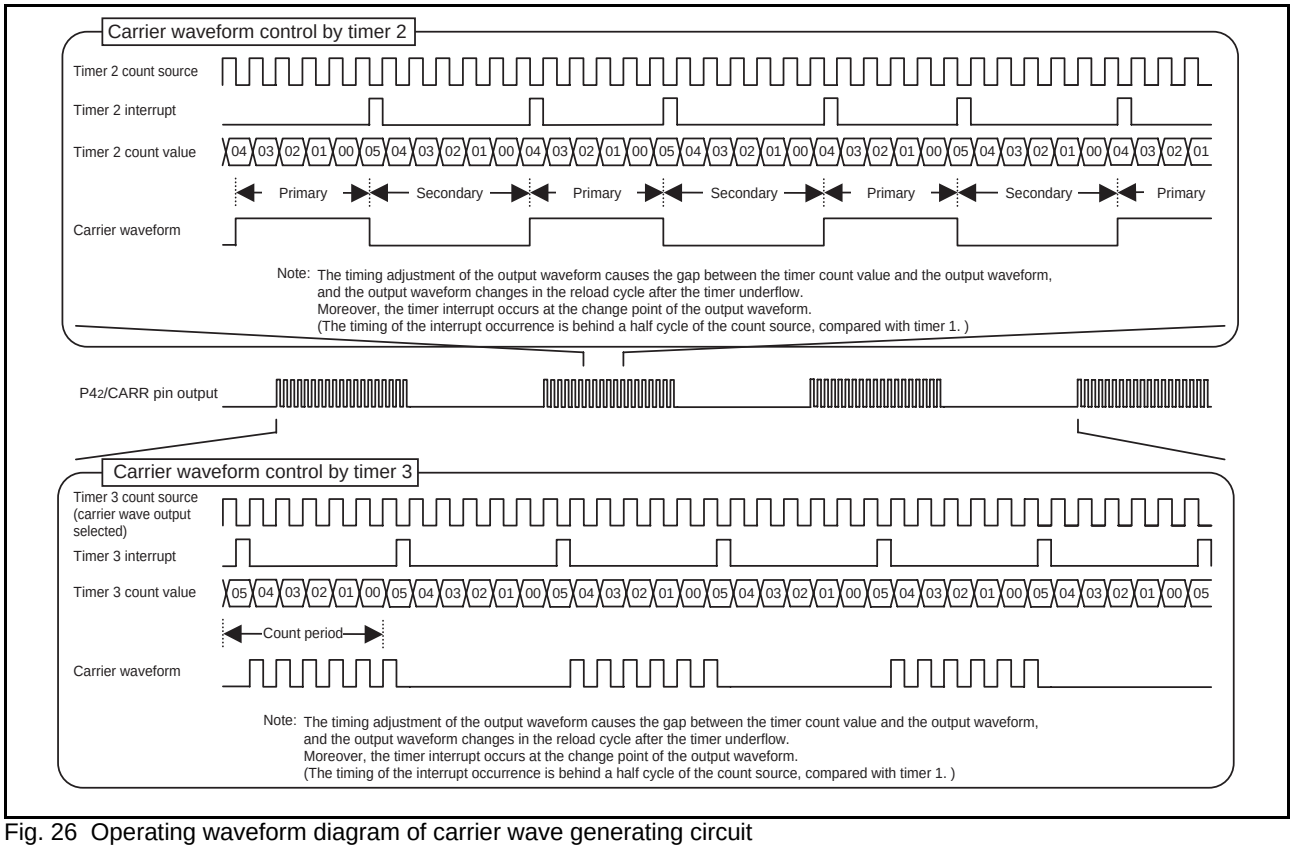


Fig. 26 Operating waveform diagram of carrier wave generating circuit

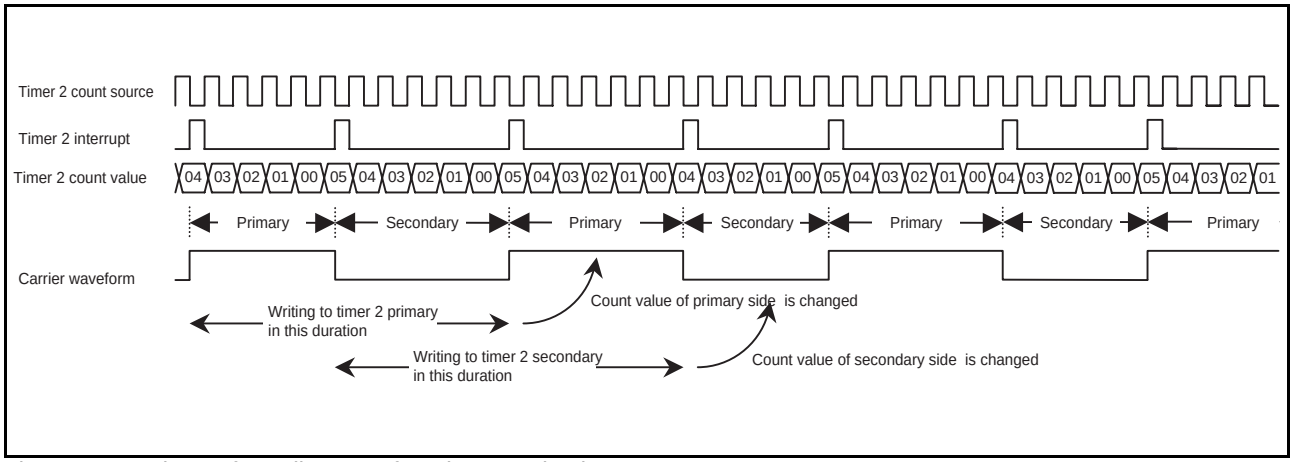


Fig. 27 Control waveform diagram of carrier wave by timer 2

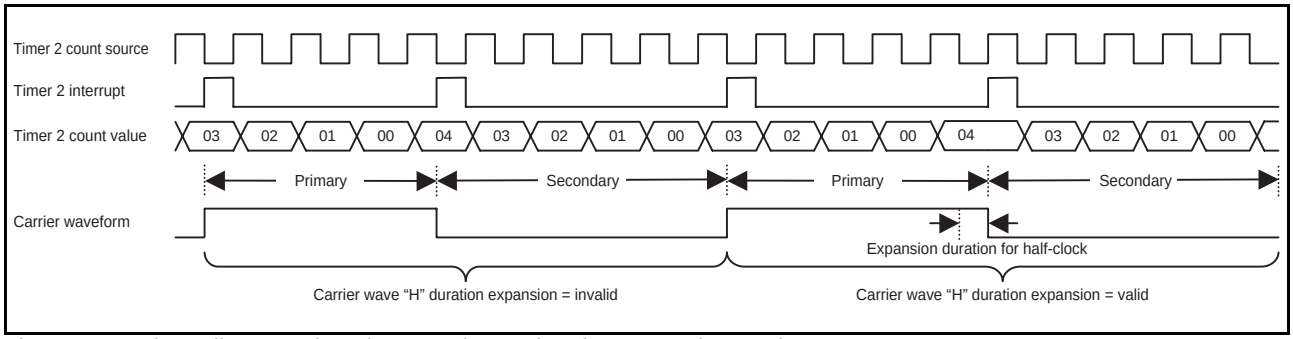


Fig. 28 Waveform diagram of carrier wave in "H" duration expansion mode

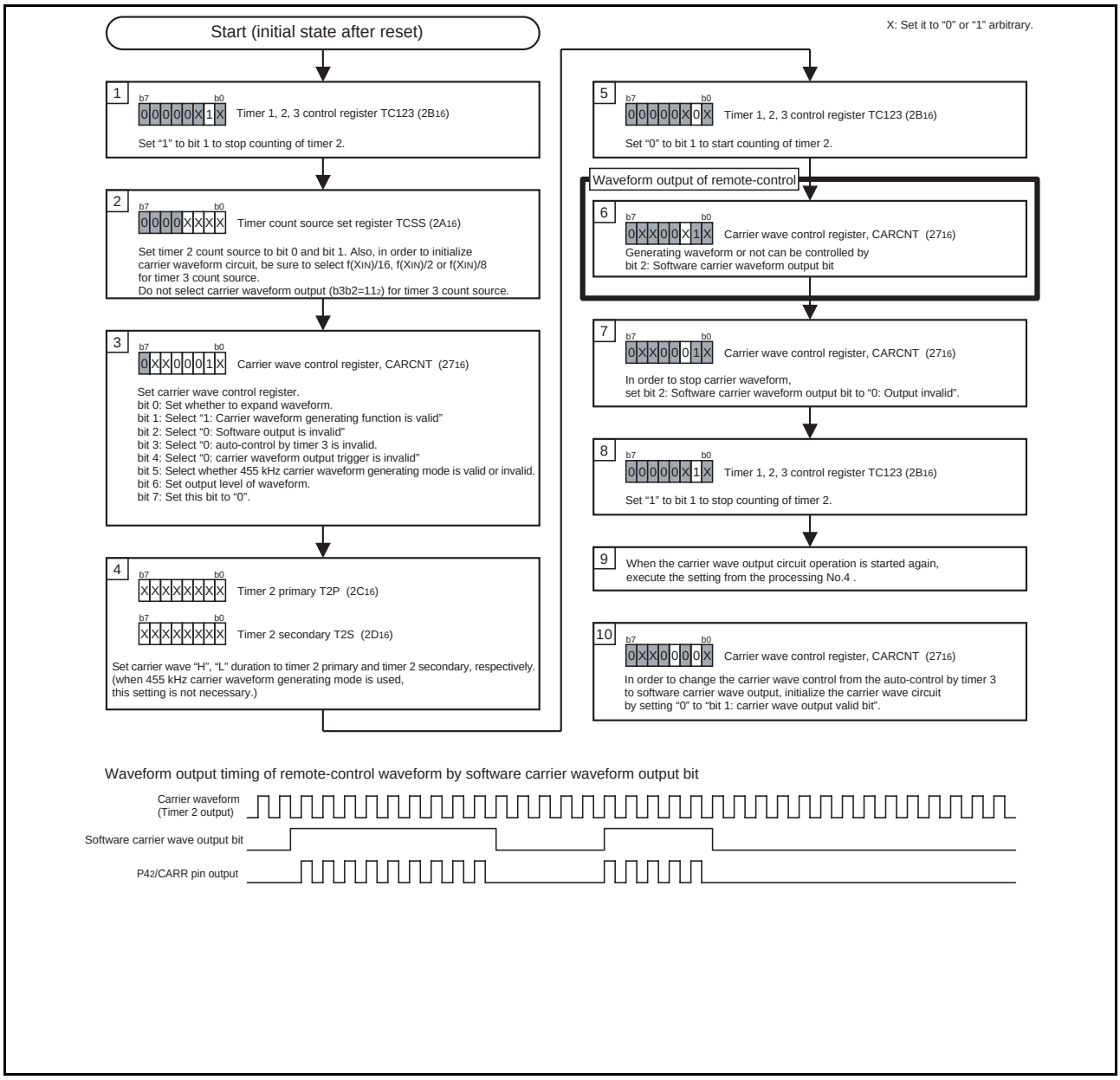


Fig. 32 Setting of carrier wave control by software

Watchdog Timer

The watchdog timer gives a means for returning to a reset status when the program fails to run on its normal loop due to a runaway.
The watchdog timer consists of an 8-bit watchdog timer H and an 8-bit watchdog timer L, being a 16-bit counter.

1. Standard operation of watchdog timer

The watchdog timer is valid by setting "0" to bit 0 of the Function set ROM data (address FFDA16) of the built-in QzROM.
When an internal clock is supplied after waiting the oscillation stabilizing time by timer 1 after system is released from reset, the watchdog timer starts operation. When the watchdog timer H underflows, an internal reset occurs. Accordingly, it is programmed that the watchdog timer control register (address 003916) can be set before an underflow occurs.
When the watchdog timer control register (address 003916) is read, the values of the high-order 6-bit of the watchdog timer H and watchdog timer H count source selection bit are read.

2. Initial value of watchdog timer

By a reset or writing to the watchdog timer control register (address 003916), the watchdog timer H is set to "FF16" and the watchdog timer L is set to "FF16".

3. Operation of watchdog timer H count source selection bit

A watchdog timer H count source can be selected by bit 7 of the watchdog timer control register (address 003916). When this bit is "0", the count source becomes a watchdog timer L underflow signal. The detection time is 262.144 ms at f(XIN) = 4 MHz.
When this bit is "1", the count source becomes f(XIN)/16. In this case, the detection time is 1024 μs at f(XIN) = 4 MHz.
This bit is cleared to "0" after reset.

4. STP instruction function selection bit

The function of the STP instruction can be selected by the bit 1 in FSROM. This bit cannot be used for rewriting by executing the STP instruction.

- When this bit is set to "0", internal reset occurs by executing the STP instruction.
- When this bit is set to "1", stop mode is entered by executing the STP instruction.

<Notes on Watchdog Timer>

1. The watchdog timer is operating during the wait mode. Write data to the watchdog timer control register to prevent timer underflow.
2. The watchdog timer stops during the stop mode. However, the watchdog timer is running during the oscillation stabilizing time after the STP instruction is released. In order to avoid the underflow of the watchdog timer, the watchdog timer H count source selection bit (bit 7 of watchdog timer control register (address 003916)) must be set to "0" just before executing the STP instruction.

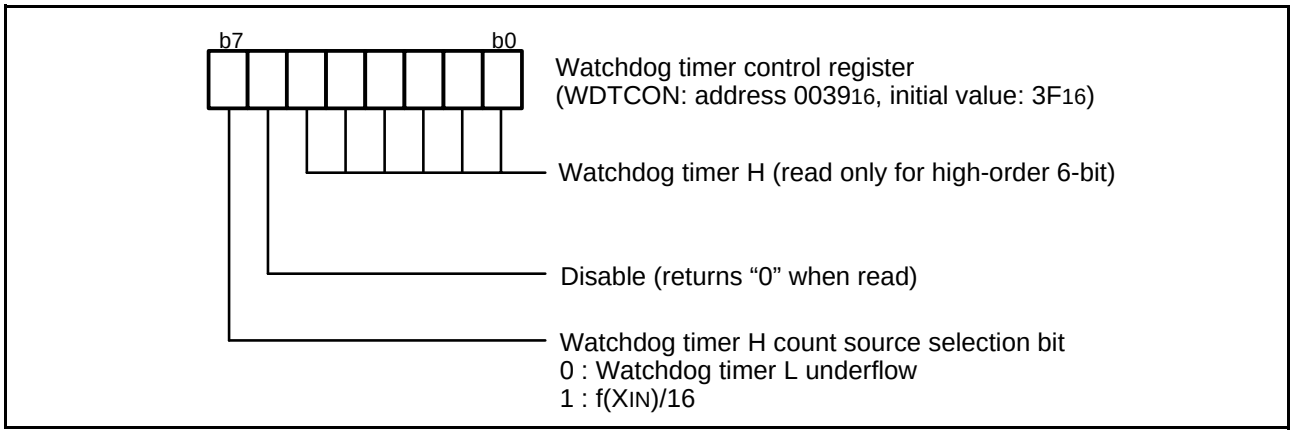


Fig. 33 Structure of watchdog timer control register

MISRG

The 7545 Group has two power source supply pins. One is the V_{CC} pin, and the other is the V_{DDR} pin only for RAM2. A potential difference between V_{CC} and V_{DDR} may cause some failures in reading from RAM2 or writing to RAM2. Accordingly, if there is a potential difference between V_{CC} and V_{DDR} at power-on, confirm the bit 1 (RAM2 status flag) of MISRG (address 0038_{16}) before reading from RAM2 or writing to RAM2.

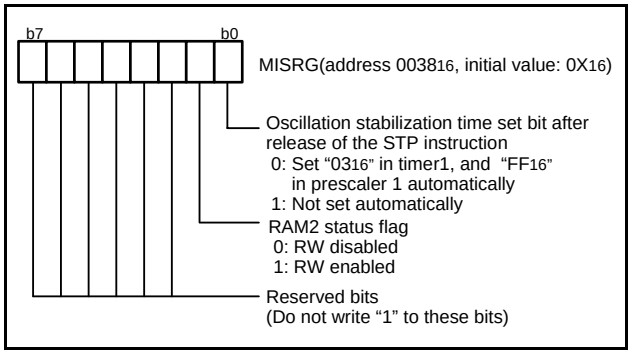


Fig. 39 Structure of MISRG

(1) Port P0 direction register (P0D)
(2) Port P1 direction register (P1D)
(3) Port P2 direction register (P2D)
(4) Port P3 direction register (P3D)
(5) Port P4 direction register (P4D)
(6) Pull-up control register (PULL)
(7) Port output mode switch register (PMOD)
(8) Key-on wakeup pin selection register (KEYSEL)
(9) Key-on wakeup edge selection register (KEYEDGE)
(10) Carrier wave control register (CARCNT)
(11) Prescaler 1 (PRE1)
(12) Timer 1 (T1)
(13) Timer count source set register (TCSS)
(14) Timer 1, 2, 3 control register (TC123)
(15) Timer 2 primary (T2P)
(16) Timer 2 secondary (T2S)
17) Timer 3 (T3)
(18) MISRG
(19) Watchdog timer control register (WDTCON)
(20) Interrupt edge selection register (INTEDGE)
(21) CPU mode register (CPUM)
(22) Interrupt request register 1 (IREQ1)
(23) Interrupt control register 1 (ICON1)
(24) Processor status register
(25) Program counter

Address	Register contents
0001_{16}	00_{16}
0003_{16}	X X X X X X 0 0
0005_{16}	00_{16}
0007_{16}	00_{16}
0009_{16}	X X X X X 0 0 0
0016_{16}	00_{16}
0017_{16}	00_{16}
0018_{16}	00_{16}
0019_{16}	00_{16}
0027_{16}	00_{16}
0028_{16}	FF_{16}
0029_{16}	03_{16}
$002A_{16}$	00_{16}
$002B_{16}$	06_{16}
$002C_{16}$	FF_{16}
$002D_{16}$	FF_{16}
$002E_{16}$	FF_{16}
0038_{16}	0 0 0 0 0 0 X 0
0039_{16}	0 0 1 1 1 1 1 1
$003A_{16}$	0 0 0 0 0 0 0 0
$003B_{16}$	1 0 0 0 0 0 0 0
$003C_{16}$	00_{16}
$003E_{16}$	00_{16}
(PS)	X X X X X 1 X X
(PCH)	Contents of address $FFFD_{16}$
(PCL)	Contents of address $FFFC_{16}$

X : Undefined

The content of other registers and RAM are undefined when the microcomputer is reset.
The initial values must be surely set before you use it.

Fig. 40 Internal status of microcomputer at reset

CLOCK GENERATING CIRCUIT

An oscillation circuit can be formed by connecting a resonator between XIN and XOUT.
Use the circuit constants in accordance with the resonator manufacturer's recommended values.
No external resistor is needed between XIN and XOUT since a feed-back resistor exists on-chip. (An external feed-back resistor may be needed depending on conditions.)

<Ceramic resonator/quartz-crystal oscillator>
When the ceramic resonator/quartz-crystal oscillator is used for the main clock, connect the ceramic resonator/quartz-crystal oscillator and the external circuit to pins XIN and XOUT at the shortest distance. A feedback resistor is built in between pins XIN and XOUT. (An external feed-back resistor may be needed depending on conditions.)

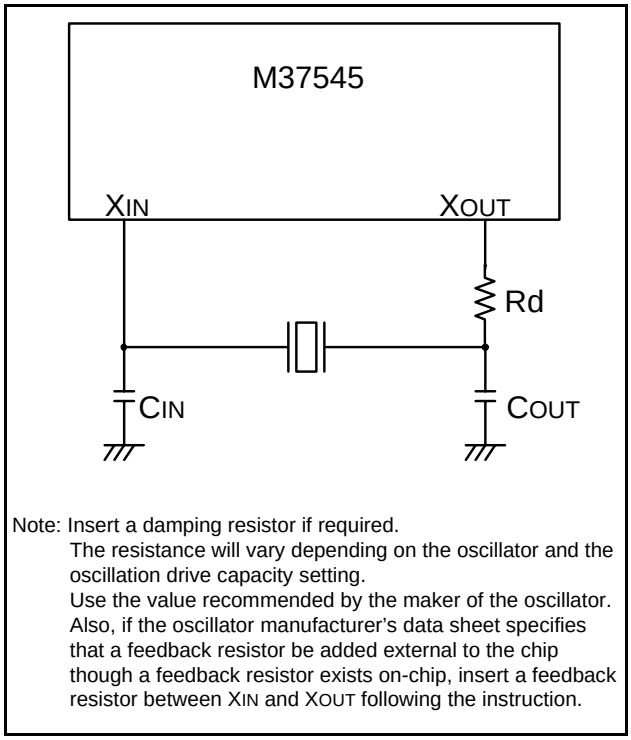
Oscillation Control

1. Stop mode

When the STP instruction is executed, the internal clock ϕ stops at an "H" level and the XIN oscillator stops. At this time, timer 1 is set to "03$_{16}$" and prescaler 1 is set to "FF$_{16}$"when the oscillation stabilization time set bit after release of the STP instruction is "0". On the other hand, timer 1 and prescaler 1 are not set when the above bit is "1". Accordingly, set the wait time fit for the oscillation stabilization time of the oscillator to be used. When an external interrupt is accepted, oscillation is restarted but the internal clock ϕ remains at "H" until timer 1 underflows. As soon as timer 1 underflows, the internal clock ϕ is supplied. This is because when a ceramic resonator is used, some time is required until a start of oscillation.
In the stop mode, the voltage drop detection circuit is stopped, so that the power dissipation is reduced.

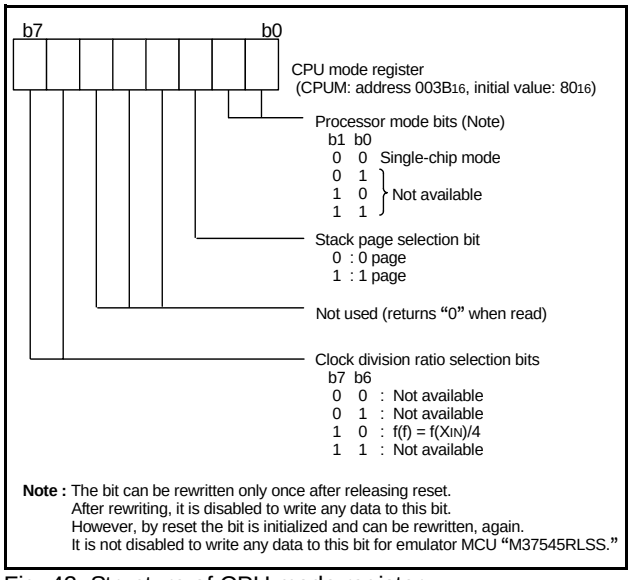
2. Wait mode

If the WIT instruction is executed, the internal clock ϕ stops at an "H" level, but the oscillator does not stop. The internal clock restarts if a reset occurs or when an interrupt is accepted. Since the oscillator does not stop, normal operation can be started immediately after the clock is restarted. To ensure that an interrupt will be accepted to release the STP or WIT state, the corresponding interrupt enable bit must be set to "1" before the STP or WIT instruction is executed.



Note: Insert a damping resistor if required.
The resistance will vary depending on the oscillator and the oscillation drive capacity setting.
Use the value recommended by the maker of the oscillator.
Also, if the oscillator manufacturer's data sheet specifies that a feedback resistor be added external to the chip though a feedback resistor exists on-chip, insert a feedback resistor between XIN and XOUT following the instruction.

Fig. 41 External circuit of ceramic resonator/quartz-crystal oscillator



Note : The bit can be rewritten only once after releasing reset.
After rewriting, it is disabled to write any data to this bit.
However, by reset the bit is initialized and can be rewritten, again.
It is not disabled to write any data to this bit for emulator MCU "M37545RLSS."

Fig. 42 Structure of CPU mode register

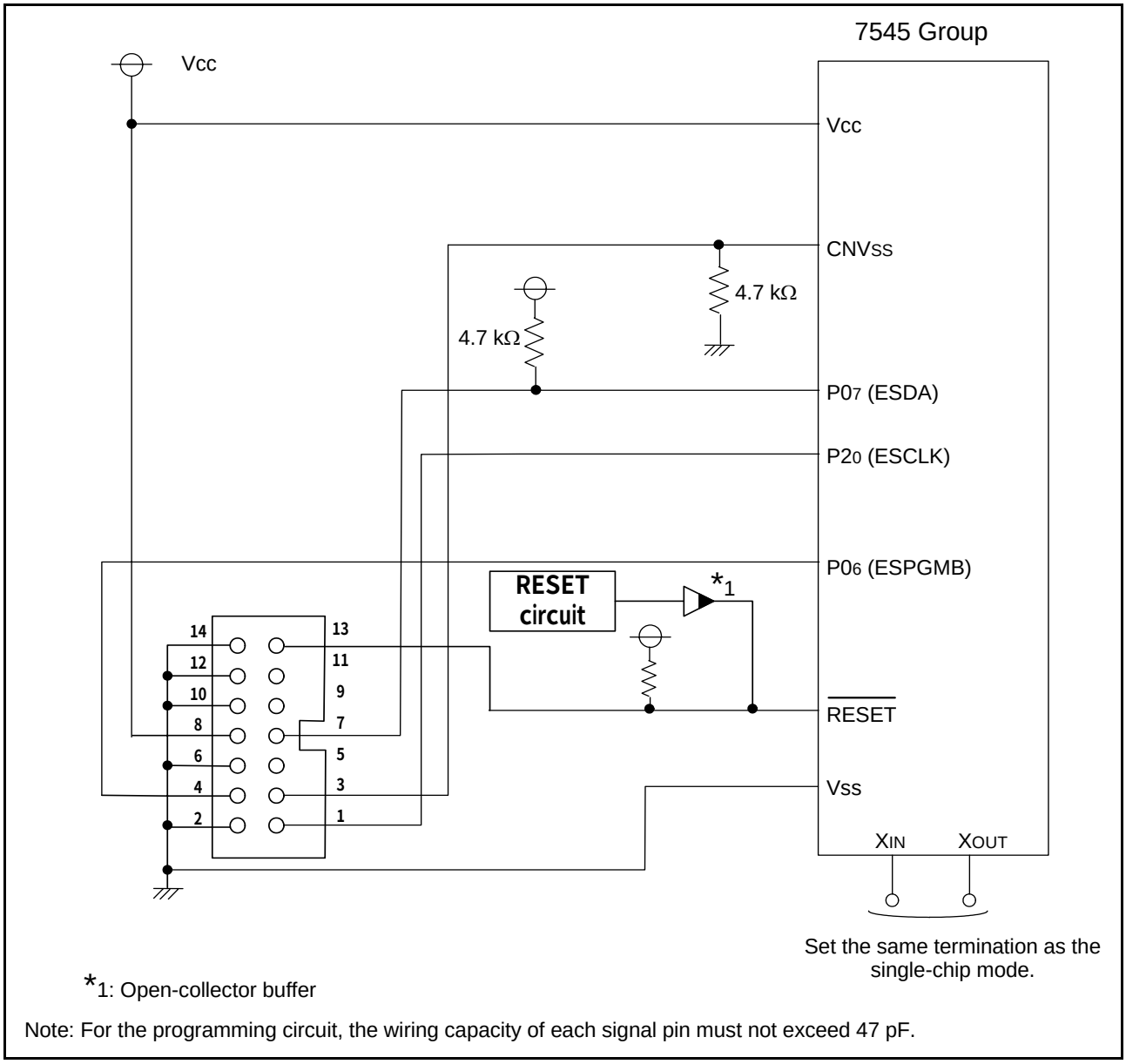


Fig. 46 When using E8 programmer, connection example

2. Connection of bypass capacitor

(1) Connection of bypass capacitor across Vss line and Vcc line

Connect an approximately 0.1 µF bypass capacitor across the VSS line and the VCC line as follows:

- Connect a bypass capacitor across the VSS pin and the VCC pin at equal length.
- Connect a bypass capacitor across the VSS pin and the VCC pin with the shortest possible wiring.
- Use lines with a larger diameter than other signal lines for VSS line and VCC line.
- Connect the power source wiring via a bypass capacitor to the VSS pin and the VCC pin.

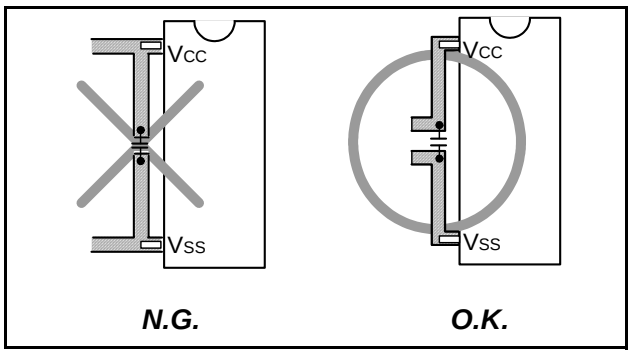


Fig. 52 Bypass capacitor across the VSS line and the VCC line

(2) Connection of bypass capacitor across Vss line and VDDR line

Connect an approximately 0.1 µF bypass capacitor across the VSS line and the VDDR line as follows:

- Connect a bypass capacitor across the VSS pin and the VDDR pin at equal length.
- Connect a bypass capacitor across the VSS pin and the VDDR pin with the shortest possible wiring.
- Use lines with a larger diameter than other signal lines for VSS line and VDDR line.
- Connect the power source wiring via a bypass capacitor to the VSS pin and the VDDR pin.

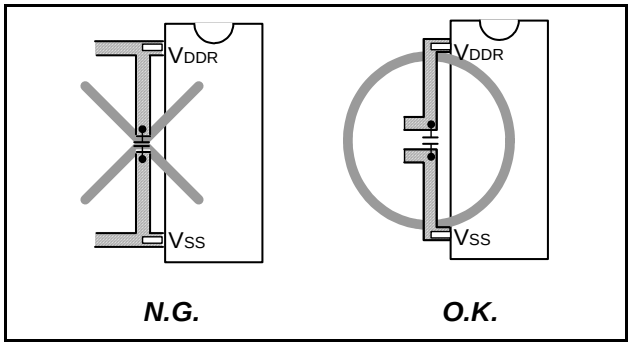


Fig. 53 Bypass capacitor across the VSS line and the VDDR line

3. Oscillator concerns

So that the product obtains the stabilized operation clock on the user system and its condition, contact the resonator manufacturer and select the resonator and oscillation circuit constants.
Be careful especially when range of voltage and temperature is wide.
Take care to prevent an oscillator that generates clocks for a microcomputer operation from being affected by other signals.

(1) Keeping oscillator away from large current signal lines

Install a microcomputer (and especially an oscillator) as far as possible from signal lines where a current larger than the tolerance of current value flows.
<Reason>
In the system using a microcomputer, there are signal lines for controlling motors, LEDs, and thermal heads or others. When a large current flows through those signal lines, strong noise occurs because of mutual inductance.

(2) Installing oscillator away from signal lines where potential levels change frequently

Install an oscillator and a connecting pattern of an oscillator away from signal lines where potential levels change frequently. Also, do not cross such signal lines over the clock lines or the signal lines which are sensitive to noise.
<Reason>
Signal lines where potential levels change frequently (such as the CARR pin signal line) may affect other lines at signal rising edge or falling edge. If such lines cross over a clock line, clock waveforms may be deformed, which causes a microcomputer failure or a program runaway.

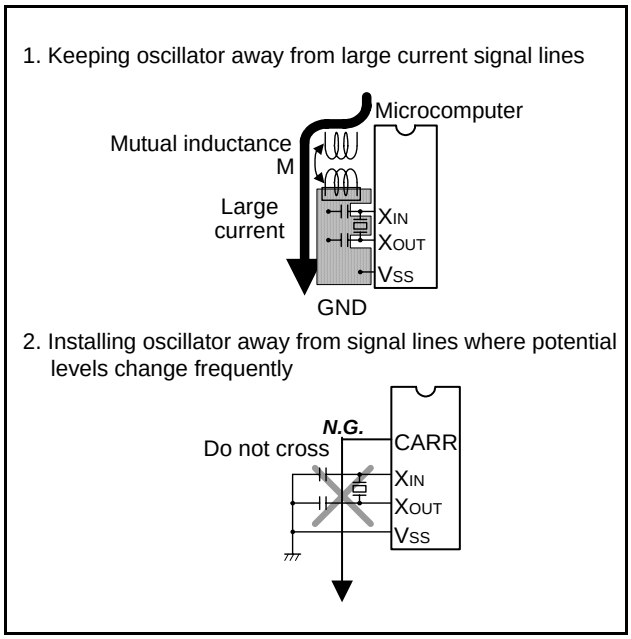


Fig. 54 Wiring for a large current signal line/Writing of signal lines where potential levels change frequently

(3) Oscillator protection using Vss pattern

As for a two-sided printed circuit board, print a VSS pattern on the underside (soldering side) of the position (on the component side) where an oscillator is mounted.
Connect the VSS pattern to the microcomputer VSS pin with the shortest possible wiring. Besides, separate this VSS pattern from other VSS patterns.

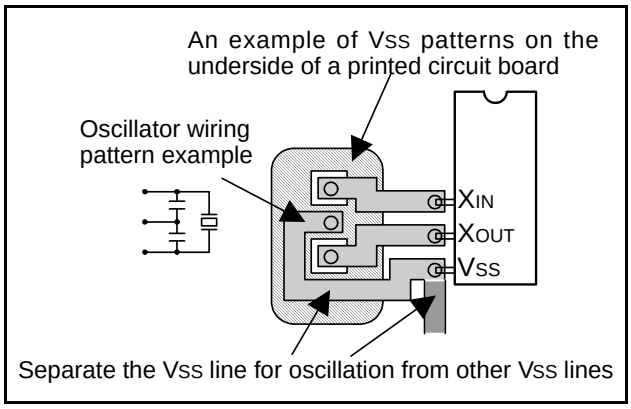


Fig. 55 Vss pattern on the underside of an oscillator

Electrical Characteristics

Table 13 Electrical characteristics (1) (V_{CC} = 1.8 to 3.6 V, Ta = –20 to 85 °C, unless otherwise noted)

Symbol	Parameter	Test conditions	Limits			Unit
			Min.	Typ.	Max.	
V_{OH}	"H" output voltage $P0_0$–$P0_7$, $P1_0$–$P1_1$, $P2_0$–$P2_7$, $P3_0$–$P3_7$ (1) $P4_0$–$P4_1$	I_{OH} = –2.0 mA V_{CC} = 3.0 V	2.1			V
V_{OH}	"H" output voltage $P4_2$	I_{OH} = –10 mA V_{CC} = 3.0 V	1.0			V
V_{OL}	"L" output voltage $P0_0$–$P0_7$, $P1_0$–$P1_1$, $P3_0$–$P3_7$	I_{OL} = 2 mA V_{CC} = 3.0 V			0.9	V
V_{OL}	"L" output voltage $P2_0$–$P2_7$, $P4_0$–$P4_2$	I_{OL} = 12 mA V_{CC} = 3.0 V			1.5	V
V_{T+}–V_{T-}	Hysteresis INT_0, INT_1, $P0_0$–$P0_7$ (2)	V_{CC} = 3.0 V		0.3		V
V_{T+}–V_{T-}	Hysteresis $\overline{RESET}$	V_{CC} = 3.0 V		0.45		V
I_{IH}	"H" input current $P0_0$–$P0_7$, $P1_0$–$P1_1$, $P2_0$–$P2_7$, $P3_0$–$P3_7$, $P4_0$–$P4_2$	V_I = V_{CC} (Pin floating. Pull up transistors "off")			5.0	µA
I_{IH}	"H" input current $\overline{RESET}$	V_I = V_{CC}			5.0	µA
I_{IL}	"L" input current $P0_0$–$P0_7$, $P1_0$–$P1_1$, $P2_0$–$P2_7$, $P3_0$–$P3_7$, $P4_0$–$P4_2$	V_I = V_{SS} (Pin floating. Pull up transistors "off")			–5.0	µA
R_{FB}	Feed-back resistor value between X_{IN}-X_{OUT}	V_{CC} = 3.0 V, V_I = 3.0 V	700		3200	kΩ
R_{PH}	Pull-up resistor value $P0_0$–$P0_7$	V_{CC} = 3.0 V, V_I = 0 V	50	120	250	kΩ
R_{PH}	Pull-up resistor value $\overline{RESET}$	V_{CC} = 3.0 V, V_I = 0 V	25	60	130	kΩ
R_{PL}	Pull-down resistor value $\overline{RESET}$	V_{CC} = 3.0 V, V_I = 3.0 V		7.0		kΩ
V_{RAM1}	RAM1 hold voltage (V_{CC})	When clock stopped	1.1		3.6	V
V_{RAM2}	RAM2 hold voltage (V_{DDR})	When clock stopped and reset by voltage drop detection	1.1			V

NOTES:
1. In this case, CMOS output is selected by the port output mode selection register.
2. It is available only when operating key-on wake up.