The overhead for the entire process from accepting an interrupt to jumping to an interrupt processing program is 20 states.

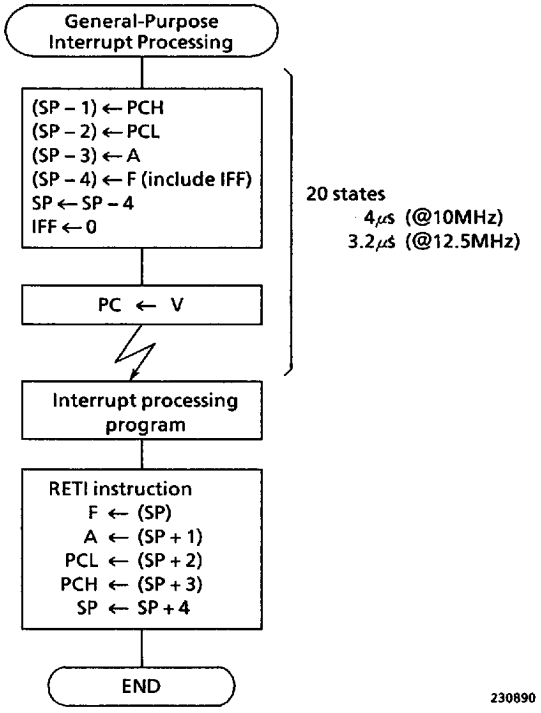

Figure 3.3 (2) General Purpose Interrupt Processing Flowchart

An interrupt (Maskable and Nonmaskable) processing program ends with a RETI instruction.

When this instruction is executed, the data previously stacked from the program counter PC and the register pair AF are restored. (Returns to the interrupt enable flag (IFF) before the interrupt.)

After the CPU reads out the interrupt vector, the interrupt source acknowledges that the CPU accepts the request, and clears the request.

A non-maskable interrupt cannot be disabled by programming. A maskable interrupt, on the other hand, can be enabled or disabled by programming. An interrupt enable flip flop (IFF) is provided on the bit 5 of Register F in the CPU. The interrupt is enabled or disabled by setting IFF to "1" by the EI instruction or to "0" by the DI instruction, respectively. IFF is reset to "0" by the reset operation or the acceptance of any interrupt (including non-maskable interrupt). The interrupt can be enabled after the subsequent instruction of EI instruction is executed.

Table 3.3 (1) lists the possible interrupt sources.

Table3.3 (1) Interrupt Sources

Priority order	Type	Interrupt sourece	Vector value ÷8	Vector value	Start address of general purpose interrupt processing	Start address of Mico DMA processing parameter
1	Non maskable	SWI instruction		10H	0010H	–
2		NMI (Input from $\overline{NMI}$ pin)		18H	0018H	–
3		INTWD (watchdog)		20H	0020H	–
4	Maskable	INT0 (External input 0)	05H	28H	0028H	FF28H
5		INTT0 (Timer 0)	06H	30H	0030H	FF30H
6		INTT1 (Timer 1)	07H	38H	0038H	FF38H
7		INTT2 (Timer 2)	08H	40H	0040H	FF40H
7		INTAD (A/D Converter)	08H	40H	0040H	FF40H
8		INTT3 (Timer 3)	09H	48H	0048H	FF48H
9		INTT4 (Timer 4)	0AH	50H	0050H	FF50H
10		INT1 (External input 1)	0BH	58H	0058H	FF58H
11		INTT5 (Timer 5)	0CH	60H	0060H	FF60H
12		INT2 (External input 2)	0DH	68H	0068H	FF68H
13		INTRX (End of serial receiving)	0EH	70H	0070H	FF70H
14		INTTX (End of serial transmission)	0FH	78H	0078H	FF78H

Note : Either INTT2 or INTAD is selected by INTEH<ADIS>. 200689

The "priority order" in the table shows the order of the interrupt source to be acknowledge by the CPU when more than one interrupt are requested at one time.

If interrupt of fourth and fifth orders are requested simultaneously, for example, an interrupt of the "5th" priority is acknowledged after a "4th" priority interrupt processing has been completed by a RETI instruction. However, a lower priority interrupt can be acknowledged immediately by executing an EI instruction in a program that processes a higher priority interrupt.

The internal interrupt controller merely determines the priority of the sources of interrupts to be acknowledged by the CPU when more than one interrupt are requested at a time. It is, therefore, unable to compare the priority of interrupt being executed with the one being requested.

To permit another interrupt during a certain interrupt operation, set the interrupt enabling flag for the source of the interrupt to be allowed, and execute the EI command.

3.3.2 Micro DMA Processing

Figure 3.3 (3) is a flowchart of the micro DMA processing. Parameters (addresses of source and destination, and transfer mode) for the data transfer between memories are loaded by the CPU from an address modified by an interrupt vector value. After the data transfer between memories according to these parameter, these parameters are updated and saved into the original locations. The CPU then decrements the number of transfers, and completes the micro DMA processing unless the result is "0".