NEC
NEC Electronics Inc.

μPD78054 Family (μPD78052/053/054/P054) 8-Bit, K-Series Microcontrollers With UART, A/D and D/A Converters

Preliminary

September 1993

Description

The μPD78052, μPD78053, μPD78054, and μPD78P054 are members of the K-Series® of microcontrollers featuring an A/D and a D/A converter, UART, 8-bit hardware multiply and divide instructions, bit manipulation instructions, four banks of main registers, an advanced interrupt handling facility, and a powerful set of memory-mapped on-chip peripherals.

Timing is generated by two oscillators. The main oscillator normally drives the CPU amd most peripherals. The 32.768-kHz subsystem oscillator provides time keeping when the main oscillator is turned off. Since CMOS power dissipation is proportional to clock rate, the 78054 family provides a software selectable instruction cycle time from 0.40 μs to122 μs. The STOP and HALT modes turn off parts of the microcontroller for additional power savings. The data retention mode keeps RAM contents valid down to 2.0 volts.

These devices are ideally suited for applications in portable battery-powered equipment, office automation, communications, consumer electronics, home appliances, and fitness equipment.

K-Series is a registered trademark of NEC Electronics, Inc.

Features

- Eight-channel 8-bit A/D converter
- Two-channel 8-bit D/A converter
  - Real-time output capability
- Three-channel serial communication interface
  - 8-bit clock-synchronous interface 0
    Full-duplex, three-wire mode
    Half-duplex, two-wire mode
    NEC serial bus interface (SBI) mode
  - 8-bit clock-synchronous interface 1
    Full-duplex, three-wire mode
    Automatic transfer, full-duplex, three-wire mode
  - Serial interface 2
    Full-duplex, three-wire mode
    UART mode
- Timers: five channels
  - Watchdog timer
  - 16-bit timer/event counter
  - Two 8-bit timer/event counters usable as one 16-bit timer/event counter
  - Watch (clock) timer
- 69 I/O lines
  - Two CMOS input-only lines
  - 63 CMOS bidirectional I/O lines
  - One real-time output port operable in one 8-bit or two 4-bit units
  - Four n-channel, open-drain I/O lines at 15 V maximum
  - Software selectable pullup resistors on 63 lines
  - Mask option pullup resistors on four lines available on ROM versions
- External memory expansion
  - 64K bytes total memory space
- Powerful instruction set
  - 8-bit unsigned multiply and divide
  - 16-bit arithmetic and data transfer instructions
  - 1-bit and 8-bit logic instructions
- Minimum instruction execution times:
  - 0.4/0.8/1.6/3.2/6.4 μs program selectable using 5-MHz main system clock
  - 122 μs using 32.768-kHz subsystem clock
- Memory-mapped on-chip peripherals
  - Special function registers
- Programmable priority, vectored-interrupt controller (two levels)
- Programmable buzzer and clock outputs
- Power saving and battery back up
  - Variable CPU clock rate
  - STOP mode
  - HALT mode
  - 2-V data retention mode
- CMOS operation; V_{DD} from 2.7 to 6.0 V

Internal High-Capacity ROM and RAM

	78052	78053	78054	78P054
ROM	16K bytes	24K bytes	32K bytes	—
PROM	—	—	—	32K bytes
High-speed RAM	512 bytes	1024 bytes	1024 bytes	1024 bytes
Serial buffer RAM	32 bytes	32 bytes	32 bytes	32 bytes

3f

Ordering Information

Part Number	ROM	Package	Package Drawing
μPD78052GC-xxx-3B9	16K mask ROM	80-pin plastic QFP	S80GC-65-3B9-1
μPD78053GC-xxx-3B9	24K mask ROM		
μPD78054GC-xxx-3B9	32K mask ROM		
μPD78P054GC-3B9	32K OTP ROM		
μPD78052GK-xxx-BE9	16K mask ROM	80-pin plastic TQFP	P80GK-50-BE9-1
μPD78053GK-xxx-BE9	24K mask ROM		
μPD78054GK-xxx-BE9	32K mask ROM		
μPD78P054GK-BE9	32K OTP ROM		
μPD78P054KK-T	32K UV EPROM	80-pin ceramic LCC w/window	X80KW-65A

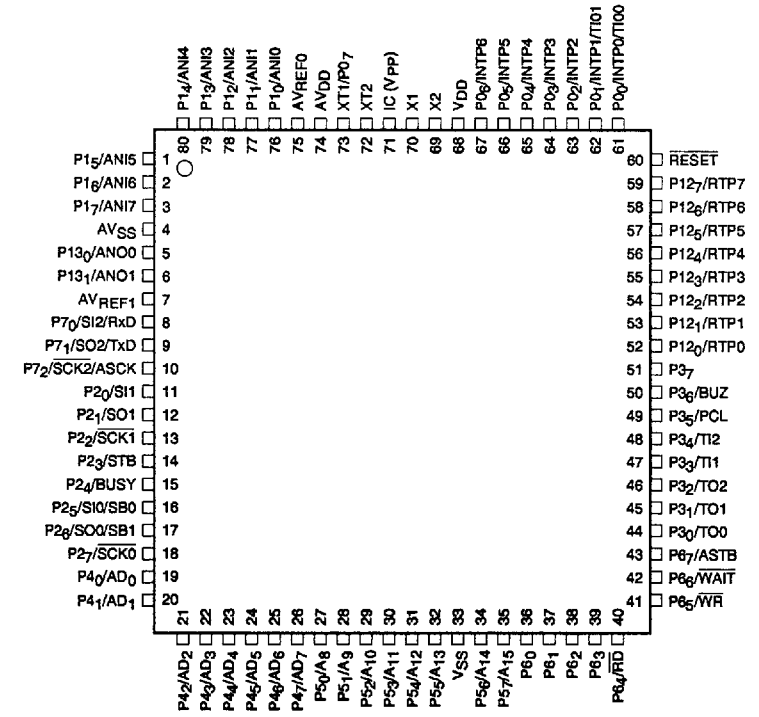
Notes:

(1) xxx indicates ROM code suffix

(2) All devices listed are standard quality grade

Pin Configurations

80-Pin Plastic QFP, Plastic TQFP, or Ceramic LCC



Notes:
(1) Connect IC (internally connected) pin (V_{PP} on μPD78P054) to V_{SS}
(2) AV_{DD} should be connected to V_{DD}.
(3) AV_{SS} should be connected to V_{SS}.

83RC-9548B (9/93)

3f

Pin Functions; Normal Operating Mode

Symbol	First Function	Symbol	Alternate Function
$P0_0$	Port 0; 8-bit, bit selectable I/O port (Bits 0 and 7 are input only)	INTP0 TI0	External maskable interrupt External count clock input to timer 0 or timer 0 capture trigger to capture registers CR00 and CR01
$P0_1$		INTP1 TI01	External maskable interrupt Timer 0 capture trigger to capture register CR00
$P0_2$ $P0_3$ $P0_4$ $P0_5$ $P0_6$		INTP2 INTP3 INTP4 INTP5 INTP6	External maskable interrupt
$P0_7$		XT1	Crystal oscillator or external clock input for subsystem clock
$P1_0$ - $P1_7$	Port 1; 8-bit, bit-selectable I/O port	ANI0-ANI7	Analog input to A/D converter
$P2_0$	Port 2; 8-bit, bit-selectable I/O port	SI1	Serial data input three-wire serial I/O mode
$P2_1$		SO1	Serial data output three-wire serial I/O mode
$P2_2$		$\overline{SCK1}$	Serial clock I/O for serial interface 1
$P2_3$		STB	Serial interface automatic transmit/receive strobe output
$P2_4$		BUSY	Serial interface automatic transmit/receive busy input
$P2_5$		SI0 SB0	Serial data input three-wire serial I/O mode 2/3-wire serial I/O mode
$P2_6$		SO0 SB1	Serial data output three-wire serial I/O mode 2/3-wire serial I/O mode
$P2_7$		$\overline{SCK0}$	Serial clock I/O for serial interface 0
$P3_0$	Port 3; 8-bit, bit-selectable I/O port	TO0	Timer output from timer 0
$P3_1$		TO1	Timer output from timer 1
$P3_2$		TO2	Timer output from timer 2
$P3_3$		TI1	External count clock input to timer 1
$P3_4$		TI2	External count clock input to timer 2
$P3_5$		PCL	Programmable clock output
$P3_6$		BUZ	Programmable buzzer output
$P3_7$		—	
$P4_0$ - $P4_7$	Port 4; 8-bit I/O port	AD_0 - AD_7	Low-order 8-bit multiplexed address/data bus for external memory
$P5_0$ - $P5_7$	Port 5; 8-bit, bit selectable I/O port	A_8 - A_{15}	High-order 8-bit address bus for external memory
$P6_0$ - $P6_3$	Port 6; 8-bit, bit selectable ($P6_0$ to $P6_3$ n-channel, open-drain I/O with mask option pullup resistors; $P6_4$ - $P6_7$ I/O). See note.	—	
$P6_4$		$\overline{RD}$	External memory read strobe
$P6_5$		$\overline{WR}$	External memory write strobe
$P6_6$		$\overline{WAIT}$	External memory wait signal input
$P6_7$		ASTB	Address strobe used to latch address for external memory

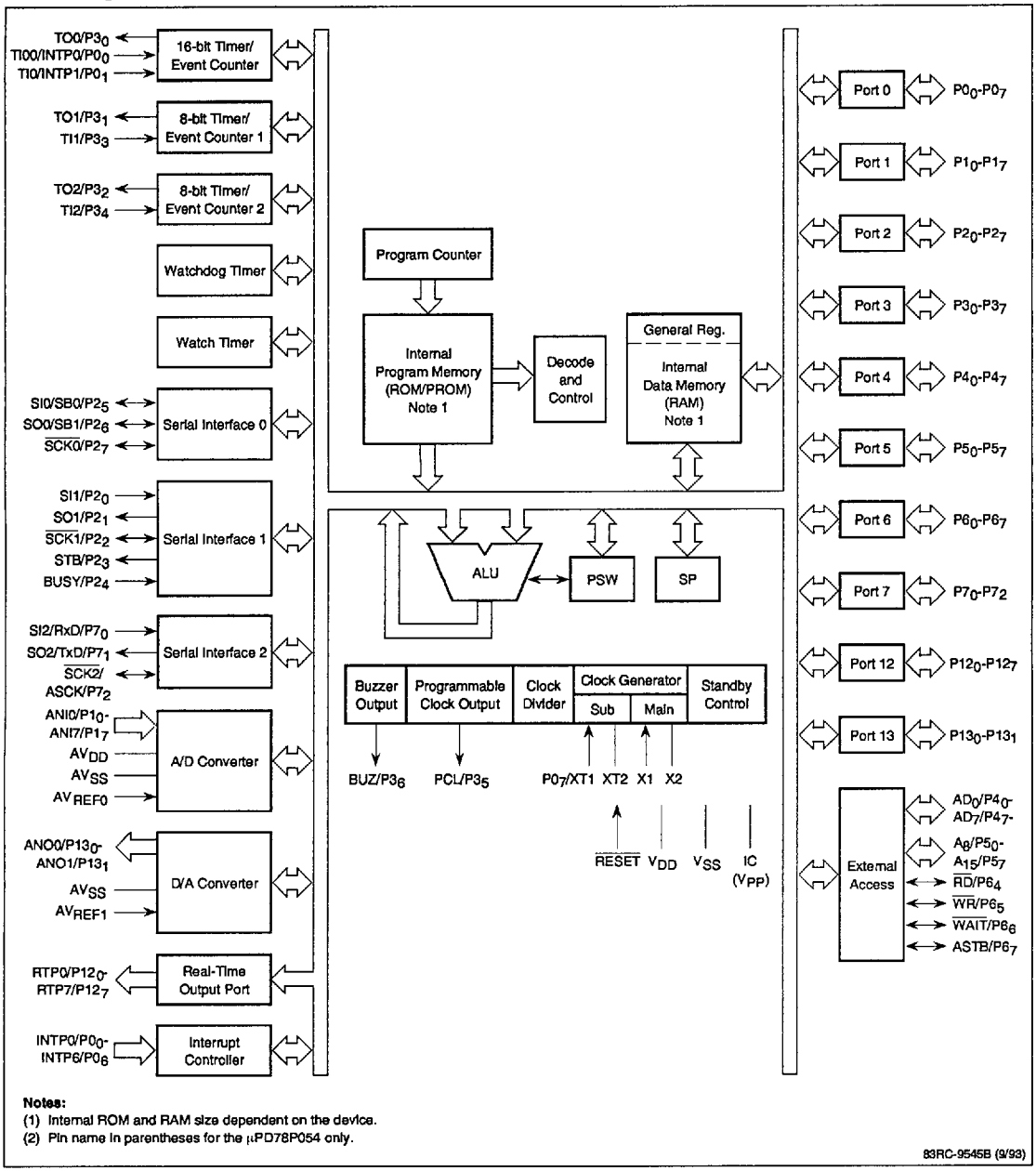
Pin Functions; Normal Operating Mode (cont)

Symbol	First Function	Symbol	Alternate Function
$P7_0$	Port 7; 3-bit , bit-selectable I/O port	SI2 RxD	Serial data input three-wire serial I/O mode Asynchronous serial data input
$P7_1$		SO2 TxD	Serial data output three-wire serial I/O mode Asynchronous serial data output
$P7_2$		$\overline{SCK2}$ ASCK	Serial clock I/O for serial interface 2 Asynchronous serial clock input
$P12_0$- $P12_7$	Port 12; 8-bit selectable I/O port	RTP0- RTP7	Real-time port
$P13_0$, $P13_1$	Port 13; 2-bit selectable I/O port	ANO0, ANO1	Analog output for D/A converter
$\overline{RESET}$	External system reset input		
X1	Crystal/ceramic resonator connection or external clock input for main system clock		
X2	Crystal/ceramic resonator connection or inverse of external clock for main system clock		
XT2	Crystal oscillator or left open when not using the subsystem clock		
AV_{REF0}	A/D converter reference voltage		
AV_{REF1}	D/A converter reference voltage		
AV_{DD}	A/D converter power supply input		
AV_{SS}	A/D and D/A converter ground		
V_{DD}	Power-supply input		
V_{PP}	μPD78P054 PROM programming power-supply input		
V_{SS}	Power-supply ground		
IC	Internal connection		

Note: See table 2 and figure 4 for details.

3f

Block Diagram



Notes:
(1) Internal ROM and RAM size dependent on the device.
(2) Pin name in parentheses for the μPD78P054 only.

83RC-9545B (9/93)

FUNCTIONAL DESCRIPTION

Central Processing Unit

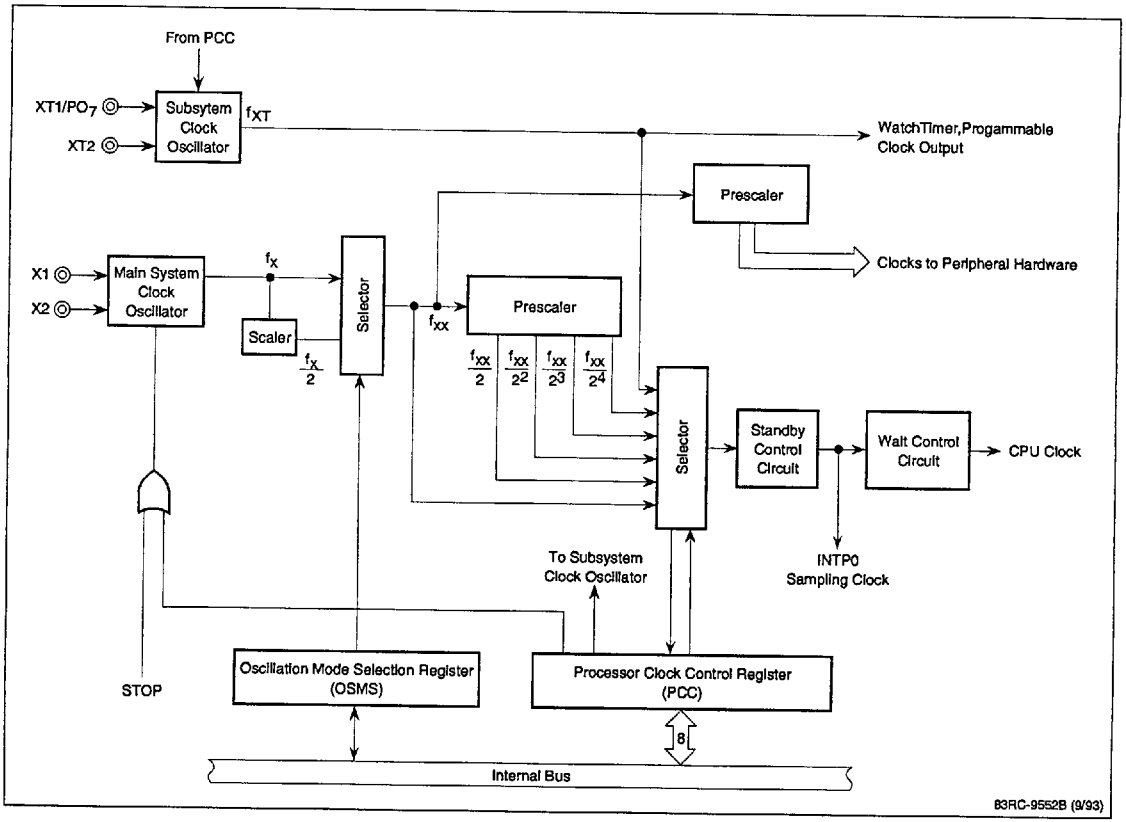
The central processing unit (CPU) of the μPD78054 family features 8- and 16-bit arithmetic including an 8 x 8-bit unsigned multiply and 16 x 8-bit unsigned divide (producing a 16-bit quotient and an 8-bit remainder). The multiply executes in 3.2 μs and the divide in 5 μs using the fastest clock cycle with a main system clock of 5.0 MHz.

A CALLT vector table and a CALLF area decrease the number of bytes in the call instructions for commonly used subroutines. A 1-byte call instruction can access up to 32 subroutines through their addresses contained in the CALLT vector table (40H to 7FH). A 2-byte call instruction can access any routine beginning in a specific CALLF area (0800H to 0FFFH).

Internal System Clock Generator

The internal system clocks of the μPD78054 family are derived from either the main system or the subsystem oscillator. See figure 1. The clocks for the watch timer and programmable clock output are derived from either the subsystem clock (f_{XT}) or the main system clock. The clocks for all other peripheral hardware are derived from the main system clock.

Figure 1. Internal System Clock Generator



3f

The CPU clock (φ) can be supplied from either the main system clock (f_X) or the subsystem clock (f_{XT}). A selector, which is controlled by the oscillation mode selection register (OSMS), determines whether the main system clock (f_X) or the scaled main system clock ($f_X/2$) is provided to the prescalar (f_{XX}). Using the processor clock control register (PCC), a CPU clock frequency equal to f_{XX}, $f_{XX}/2$, $f_{XX}/4$, $f_{XX}/8$, $f_{XX}/16$ or the subsystem clock f_{XT} can be selected. The CPU clock selected should be based on the power supply voltage available and the desired power consumption. On power up, the CPU clock defaults to the lowest speed from the main system clock ($f_{XX}/16$ with $f_{XX} = f_x/2$) and can be changed while the microcomputer is running.

Since the shortest instruction takes two CPU clocks to execute, the fastest minimum instruction execution time (t_{CY}) of 0.4 μs is achieved with a main system clock at 5.0 MHz ($f_{XX} = f_X$) and a V_{DD} of 4.5 to 6.0 volts. However, if the watch timer must generate an interrupt every 0.5 or 0.25 seconds, t_{CY} is 0.48 μs at 4.19 MHz with $f_{XX} = f_X$. The fastest minimum instruction execution time available across the full voltage range of 2.7 to 6.0 volts is 0.96 μs with a 4.19 MHz main system clock ($f_{XX} = f_X$). For the lowest power consumption, the CPU can be operated from the subsystem clock and the minimum instruction execution time is 122 μs at 32.768 kHz.

Memory Space

The μPD78054 family has a 64K-byte address space (see figure 2). This address space (0000H-FFFFH) can be used as both program and data memory.

Internal Program Memory

All devices in the μPD78054 family have internal program memory. The μPD78052/053/054 contain 16K, 24K, and 32K bytes of internal ROM, respectively. The μPD78P054 contains 32K bytes of UV EPROM or one time programmable ROM. To allow the μPD78P054 to emulate the mask ROM devices, the amount of internal program memory available in the μPD78P054 can be selected using the memory size switching register (IMS).

Internal RAM

The μPD78052 has 544 bytes and the μPD78053/054/P054 have 1056 bytes of Internal RAM. This Internal RAM consists of two types: high-speed Internal RAM and buffer RAM.

The μPD78052 contains 512 bytes (FD00H to FEFFH) while the μPD78053/054/P054 contain 1024 bytes (FB00H to FEFFH) of high-speed Internal RAM. The high-speed Internal RAM contains the general register banks and the stack. The remainder of the high-speed Internal RAM and any unused register bank locations are available for general storage.

All devices also contain 32 bytes of buffer RAM (FAC0H to FADFH). The buffer RAM is used for the automatic transfer mode of serial interface 1 or for general storage.

To allow the μPD78P054 to emulate the mask ROM devices, the amount of high-speed Internal RAM available in the μPD78P054 can also be selected using the IMS.

External Memory

The μPD78054 family can access 0, 256, 4K, 16K or all available bytes of external memory. The μPD78054 family has an 8-bit wide external data bus and a 16-bit wide external address bus. The low-order 8 bits of the address bus are multiplexed and also provide the 8-bit data bus and are supplied by port 4. The high-order address bits of the 16-bit address bus are taken from port 5 as required. The address latch, read, and write strobes, and the external $\overline{WAIT}$ signal are supplied by port 6.

The memory expansion mode register (MM) controls the size of external memory. It can be programmed to use 0, 4, 6, or 8 bits from port 5 for the high-order address. Any remaining port 5 bits can be used for I/O. The MM register also can be used to specify one additional wait state or the use of the external $\overline{WAIT}$ signal for low-speed external memory or external peripheral devices.

When only internal ROM and RAM are used and no external memory is required, ports 4, 5 and 6 are available as general purpose I/O ports.

CPU Control Registers

Program Counter. The program counter is a 16-bit binary counter register that holds the address of the next instruction to be executed. During reset, the program counter is loaded with the address stored in locations 0000H and 0001H.

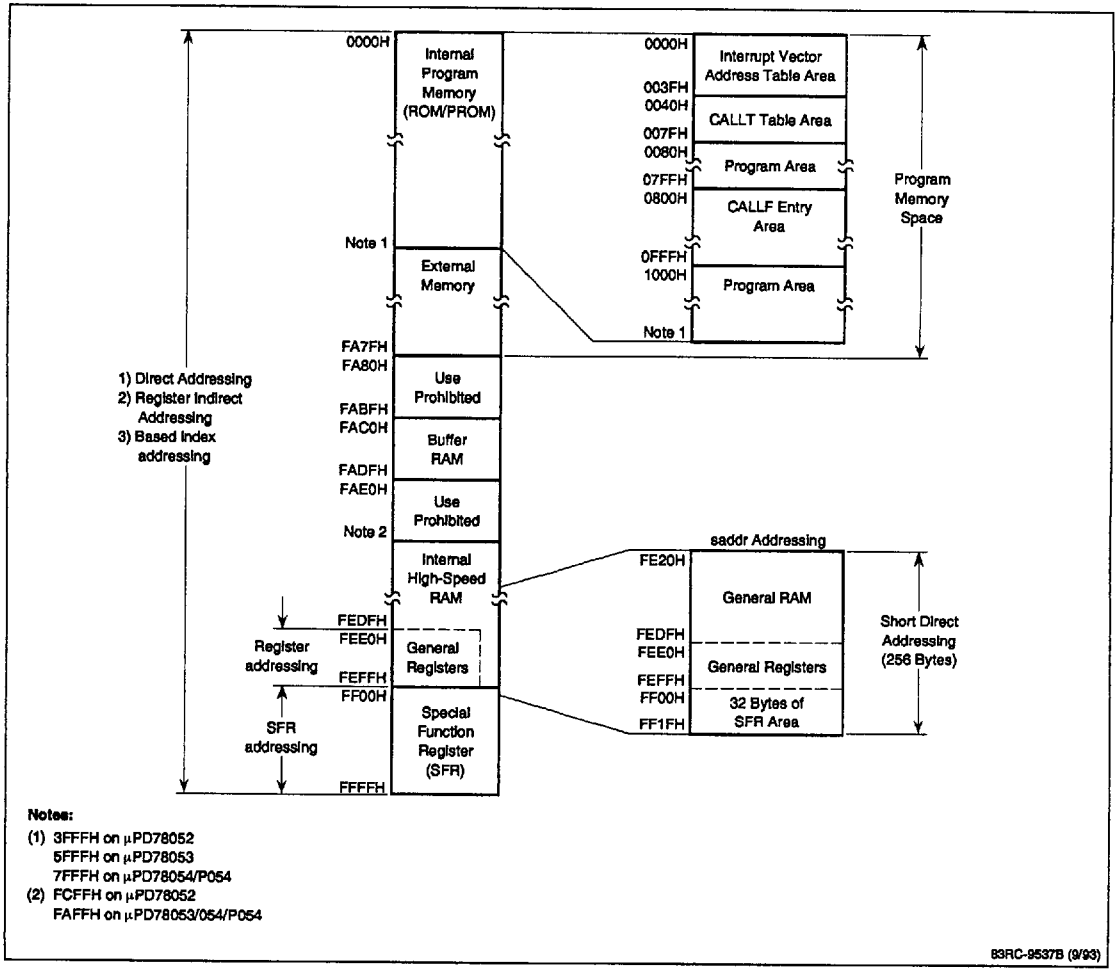
Stack Pointer. The stack pointer is a 16-bit register that holds the address of the last item pushed onto the stack. It is decremented before new data is pushed onto the stack and incremented after data is popped off the stack.

Program Status Word. The program status word (PSW) is an 8-bit register that contains flags that are set or reset depending on the results of an instruction. This register can be written to or read from 8 bits at a time. The individual flags can also be manipulated on a bit-by-bit basis. The assignment of PSW bits follows.

7							0
IE	Z	RBS1	AC	RBS0	0	ISP	CY

CY	Carry flag
ISP	In-service (interrupt) priority flag
RBS0, RBS1	Register bank selection flags
AC	Auxiliary carry flag
Z	Zero flag
IE	Interrupt request enable flag

Figure 2. Memory Map



Notes:

(1) 3FFFH on μPD78052
5FFFH on μPD78053
7FFFH on μPD78054/P054

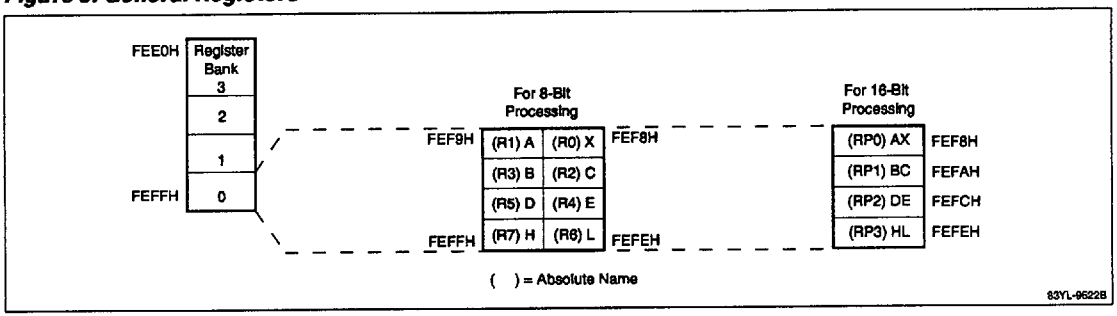
(2) FCFFH on μPD78052
FAFFH on μPD78053/054/P054

83RC-9537B (9/93)

3f

General Registers

Figure 3. General Registers



The general-purpose registers (figure 3) consist of four banks of registers located at addresses FEE0H to FEFFH in Internal RAM. Each bank consists of eight 8-bit general registers that can also be used in pairs to function as four 16-bit registers. Two bits in the PSW (RBS0 and RBS1) specify which of the register banks is active at any time and are set under program control.

Registers have both functional names (A, X, B, C, D, E, H, or L for 8-bit registers and AX, BC, DE, and HL for 16-bit registers) and absolute names (R1, R0, R3, R2, R5, R4, R7, or R6 for 8-bit registers and RP0, RP1, RP2, or RP3 for 16-bit registers). Either the functional or absolute register names can be used in instructions that use the operand identifiers r and rp.

Addressing

The program memory addressing (ROM) modes provided are relative, immediate, table indirect and register addressing. The operand addressing modes provided are implied, register, direct, short direct (saddr), special function (SFR), register indirect, based, based indexed, and stack addressing.

The SFR addressing and saddr addressing modes use direct addressing and require only 1 byte in the instruction to address RAM. Normally a 65K byte address space requires 2 bytes to address it. One-byte addressing results in faster access times, since the instructions are shorter. SFR addressing addresses the entire 256-byte SFR address space from FF00H to FFFFH. Saddr addressing (see figure 2) addresses the 256-byte address space FE20H to FF1FH. FE20H to FEFFH are composed of 224 bytes of internal high speed RAM; FF00H to FF1FH contain the first 32 bytes in the special function register area.

One-byte addressing is accomplished by using the first byte of the instruction for the opcode (and one operand if register A or AX is used) and the second byte of the instruction as an address (offset) into the 256-byte area. If register A or AX is used, the instructions are 2 bytes long, thereby providing fast access times. If immediate data is used, the instruction will be 3 or 4 bytes long depending upon whether the immediate data is a byte or a word. Many 16-bit SFRs are in the space FF00H to FF1FH. Using AX as an operand to these SFRs will provide fast access, since the instructions will be only 2 bytes long.

Special Function Registers

The input/output ports, timers, capture and compare registers, and mode and control registers for both the peripherals and CPU are collectively known as special function registers. They are all memory-mapped between FF00H and FFFFH and can be accessed either by main memory addressing or by SFR addressing. FF00H to FF1H can also be accessed using saddr addressing. They are either 8 or 16 bits as required, and many of the 8-bit registers are bit addressable. Table 1 lists the special function registers.

Input/Output Ports

Each device in the μPD78054 family has 69 port lines. Table 2 lists the features of each port and figure 4 shows the structure of each port pin.

Table 1. Special Function Registers

Address	Register (SFR)	Symbol	R/W	Access Units (Bits) 1	8	16	State After Reset
FF00H	Port 0	P0	R/W	x	x	—	00H
FF01H	Port 1	P1	R/W	x	x	—	00H
FF02H	Port 2	P2	R/W	x	x	—	00H
FF03H	Port 3	P3	R/W	x	x	—	00H
FF04H	Port 4	P4	R/W	x	x	—	Undefined
FF05H	Port 5	P5	R/W	x	x	—	Undefined
FF06H	Port 6	P6	R/W	x	x	—	Undefined
FF07H	Port 7	P7	R/W	x	x	—	00H
FF0CH	Port 12	P12	R/W	x	x	—	00H
FF0DH	Port 13	P13	R/W	x	x	—	00H
FF10H-FF11H	Capture/compare register 00	CR00	R/W	—	—	x	Undefined
FF12H-FF13H	Capture/compare register 01	CR01	R/W	—	—	x	Undefined
FF14H-FF15H	16-bit timer register	TM0	R	—	—	x	00H
FF16H	Compare register 10	CR10	R/W	—	x	—	Undefined
FF17H	Compare register 20	CR20	R/W	—	x	—	Undefined
FF18H	8-bit timer register 1	TM1	R	x	x	—	00H
FF19H	8-bit timer register 2	TM2	R	x	x	—	00H
FF18H-FF19H	16-bit timer register	TMS	R	—	—	x	0000H
FF1AH	Serial I/O shift register 0	SIO0	R/W	—	x	—	Undefined
FF1BH	Serial I/O shift register 1	SIO1	R/W	—	x	—	Undefined
FF1FH	A/D conversion result register	ADCR	R	—	x	—	Undefined
FF20H	Port mode register 0	PM0	R/W	x	x	—	FFH
FF21H	Port mode register 1	PM1	R/W	x	x	—	FFH
FF22H	Port mode register 2	PM2	R/W	x	x	—	FFH
FF23H	Port mode register 3	PM3	R/W	x	x	—	FFH
FF25H	Port mode register 5	PM5	R/W	x	x	—	FFH
FF26H	Port mode register 6	PM6	R/W	x	x	—	FFH
FF27H	Port mode register 7	PM7	R/W	x	x	—	FFH
FF2CH	Port mode register 12	PM12	R/W	x	x	—	FFH
FF2DH	Port mode register 13	PM13	R/W	x	x	—	FFH
FF30H	Real-time output port buffer register L	RTBL	R/W	—	x	—	00H
FF31H	Real-time output port buffer register H	RTBH	R/W	—	x	—	00H
FF34H	Real-time output port mode register	RTPM	R/W	x	x	—	00H
FF36H	Real-time output port control register	RTPC	R/W	x	x	—	00H
FF40H	Timer clock select register 0	TCL0	R/W	x	x	—	00H
FF41H	Timer clock select register 1	TCL1	R/W	—	x	—	00H
FF42H	Timer clock select register 2	TCL2	R/W	—	x	—	00H
FF43H	Timer clock select register 3	TCL3	R/W	—	x	—	88H
FF47H	Sampling clock select register	SCS	R/W	—	x	—	00H
FF48H	16-bit timer mode control register	TMC0	R/W	x	x	—	00H

3f

Table 1. Special Function Registers (cont)

Address	Register (SFR)	Symbol	R/W	Access Units (Bits) 1	8	16	State After Reset
FF49H	8-bit timer mode control register	TMC1	R/W	x	x	—	00H
FF4AH	Watch timer mode control register	TMC2	R/W	x	x	—	00H
FF4CH	Capture/compare control regiter 0	CRC0	R/W	x	x	—	04H
FF4EH	16-bit timer output control register	TOC0	R/W	x	x	—	00H
FF4FH	8-bit timer output control register	TOC1	R/W	x	x	—	00H
FF60H	Serial operating mode register 0	CSIM0	R/W	x	x	—	00H
FF61H	Serial bus interface control register	SBIC	R/W	x	x	—	00H
FF62H	Slave address register	SVA	R/W	—	x	—	Undefined
FF63H	Interrupt timing specify register	SINT	R/W	x	x	—	00H
FF68H	Serial operation mode register 1	CSIM1	R/W	x	x	—	00H
FF69H	Automatic data transmit/receive control register	ADTC	R/W	x	x	—	00H
FF6AH	Automatic data transmit/receive address pointer register	ADTP	R/W	—	x	—	00H
FF6BH	Automatic data transmission/reception interval specification register	ADT1	R/W	x	x	—	00H
FF70H	Asynchronous serial interface mode register	ASIM	R/W	x	x	—	00H
FF71H	Asynchronous serial interface status register	ASIS	R	x	x	—	00H
FF72H	Serial interface operating mode register 2	CSIM2	R/W	x	x	—	00H
FF73H	Baud rate generator control register	BRGC	R/W	—	x	—	00H
FF74H (Note 1)	Transmit shift register Serial I/O shift register Receive buffer register	TXS SIO2 RXB	W R/W R	— — —	x x x	— — —	FFH FFH FFH
FF80H	A/D converter mode register	ADM	R/W	x	x	—	01H
FF84H	A/D converter input select register	ADIS	R/W	—	x	—	00H
FF90H	D/A converter data register 0	DACS0	R/W	—	x	—	00H
FF91H	D/A converter data register 1	DACS1	R/W	—	x	—	00H
FF98H	D/A converter mode register	DAM	R/W	x	x	—	00H
FFD0H-FFDFH	External SFR access area (Note 2)	—	R/W	x	x	—	Undefined
FFE0H	Interrupt flag register L	IF0L	R/W	x	x	—	00H
FFE1H	Interrupt flag register H	IF0H	R/W	x	x	—	00H
FFE0H-FFE1H	Interrupt flag register	IF0	R/W	—	—	x	0000H
FFE2H	Interrupt flag register 1L	IF1L	R/W	x	x	—	00H
FFE4H	Interrupt mask flag register L	MK0L	R/W	x	x	—	FFH
FFE5H	Interrupt mask flag register H	MK0H	R/W	x	x	—	FFH
FFE4H-FFE5H	Interrrupt mask flag register	MK0	R/W	—	—	x	FFFFH
FFE6H	Interrupt mask flag register 1L	MK1L	R/W	x	x	—	FFH
FFE8H	Priority order specify flag register L	PR0L	R/W	x	x	—	FFH
FFE9H	Priority order specify flag register H	PR0H	R/W	x	x	—	FFH
FFE8H-FFE9H	Priority order specify flag register	PR0	R/W	—	—	x	FFFFH
FFEAH	Priority order specify flag register 1L	PR1L	R/W	x	x	—	FFH

Table 1. Special Function Registers (cont)

Address	Register (SFR)	Symbol	R/W	Access Units (Bits) 1	8	16	State After Reset
FFECH	External interrupt mode register 0	INTM0	R/W	—	x	—	00H
FFEDH	External interrupt mode register 1	INTM1	R/W	—	x	—	00H
FFF0H	Memory size switch register	IMS	W	—	x	—	(Note 3)
FFF2H	Oscillation mode select register	OSMS	R/W	x	x	—	00H
FFF3H	Pullup resistor option register H	PU0H	R/W	x	x	—	00H
FFF6H	Key return mode register	KRM	R/W	x	x	—	02H
FFF7H	Pullup resistor option register L	PU0L	R/W	x	x	—	00H
FFF8H	Memory expanded mode register	MM	R/W	x	x	—	10H
FFF9H	Watchdog timer mode register	WDTM	R/W	x	x	—	00H
FFFAH	Oscillation stabilization time select register	OSTS	R/W	—	x	—	04H
FFFBH	Processor clock control register	PCC	R/W	x	x	—	04H

Notes:

(1) SIO2 can be used instead of TXS and RXB. SIO2 is not a register; it is another symbol which can be used to reference the TXS and RXB registers. A write to SIO2 causes the CPU to write to the TXS register, and a read from SIO2 causes the CPU to read the RXB register

(2) The external access area cannot be accessed using SFR addressing. It can only be accessed using main memory addressing

(3) Value after reset depends on device
μPD78052: 44H
μPD78053 C6H
μPD78054 C8H
μPD78P054 C8H

3f

Table 2. Digital Port Functions

Port	Operational Features	Configuration	Direct Drive Capability	Software Pullup Resistor Connection (Note 1)
Port 0 (Note 3)	8-bit input or output	Bit selectable		Byte selectable, input bits only
Port 1 (Note 2)	8-bit input or output	Bit selectable		Byte selectable, input bits only
Port 2	8-bit input or output	Bit selectable		Byte selectable, input bits only
Port 3	8-bit input or output	Bit selectable		Byte selectable, input bits only
Port 4	8-bit input or output	Byte selectable		Byte selectable, input bits only
Port 5	8-bit input or output	Bit selectable	LED	Byte selectable, input bits only
Port 6	8-bit input or output ($P6_0$ - $P6_3$ n-channel)	Bit selectable	LED ($P6_0$-$P6_3$)	Byte selectable, input bits only $P6_0$ - $P6_3$ - mask option only (Note 4) $P6_4$ - $P6_7$ - software
Port 7	3-bit input or output	Bit selectable		Byte selectable, input bits only
Port 12	8-bit input or output	Bit selectable		Byte selectable, input bits only
Port 13	2-bit input or output	Bit selectable		Byte selectable, input bits only

Notes:

(1) Software pullup resistors can be internally connected (only on a port-by-port basis) to port bits set to input mode. Pullup resistors are not connected to port bits set to output mode

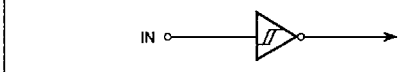
(2) Pullup resistors are automatically disconnected on pins used for A/D converter analog inputs

(3) $P0_0$ and $P0_7$ are input only and do not have a software pullup resistor When using $P0_7$ as an input, the feedback resistor for the subsystem clock should be disconnected with bit 6 (FRC) of the processor control register (PCC).

(4) All devices except μPD78P054

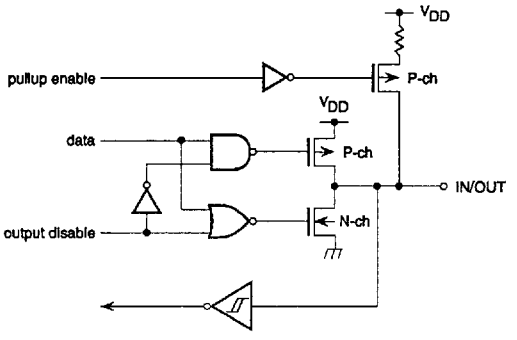
Figure 4. Pin Input/Output Circuits

Type 2 ($P0_0$, $\overline{RESET}$)

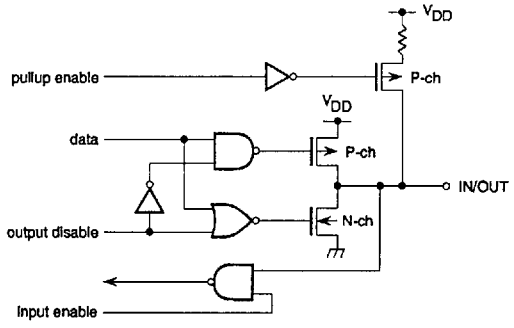


Schmitt trigger input with hysteresis characteristics

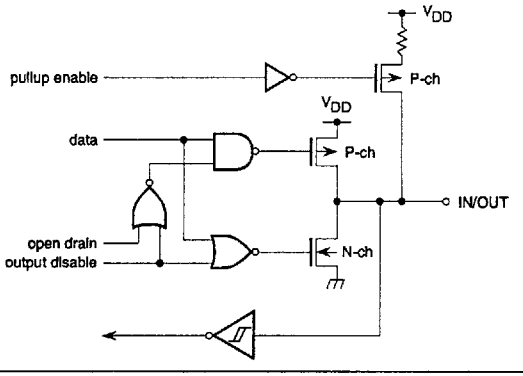
Type 8-A ($P0_1$ - $P0_6$, $P2_0$, $P2_2$, $P2_4$, $P3_3$- $P3_4$, $P7_0$, $P7_2$)



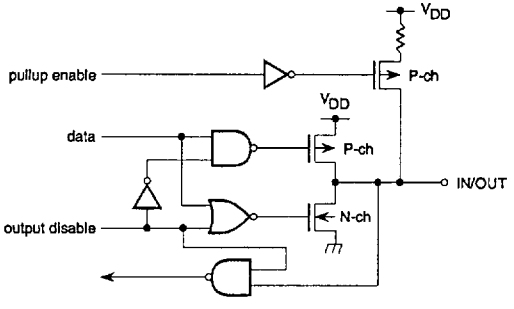
Type 5-A ($P2_1$, $P2_3$, $P3_0$- $P3_2$, $P3_5$- $P3_7$, $P5_0$- $P5_7$, $P6_4$- $P6_7$, $P7_1$, $P12_0$- $P12_7$)



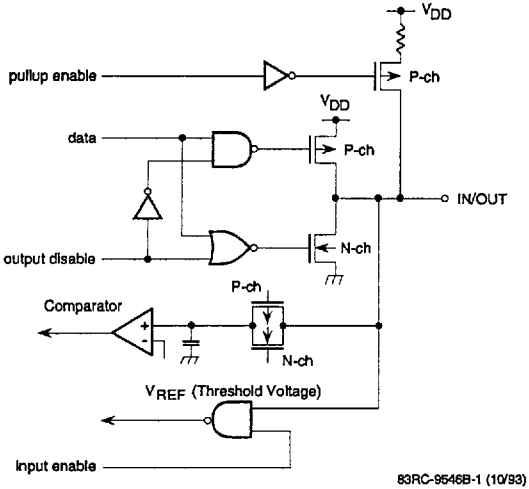
Type 10-A ($P2_5$- $P2_7$)



Type 5-E ($P4_0$- $P4_7$)

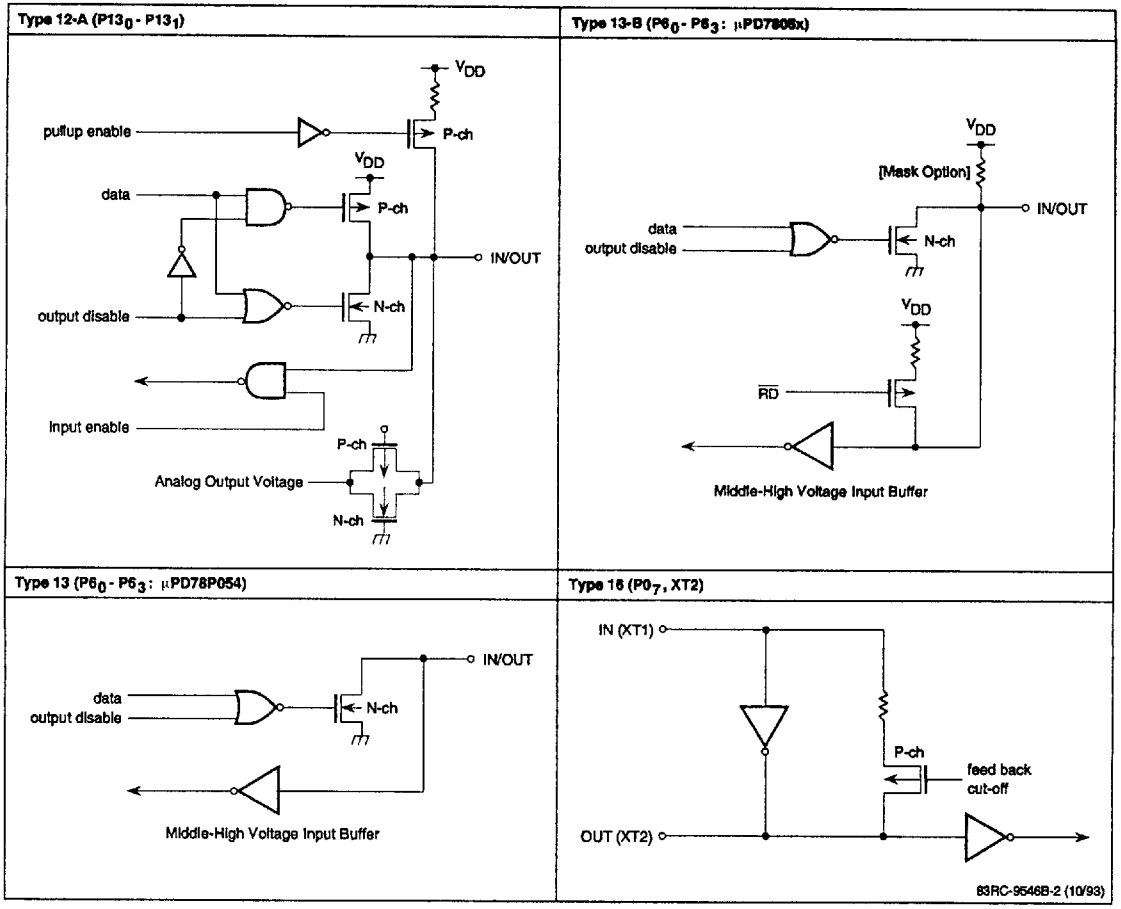


Type 11 ($P1_0$ - $P1_7$)



83RC-9546B-1 (10/93)

Figure 4. Pin Input/Output Circuits (cont)



3f

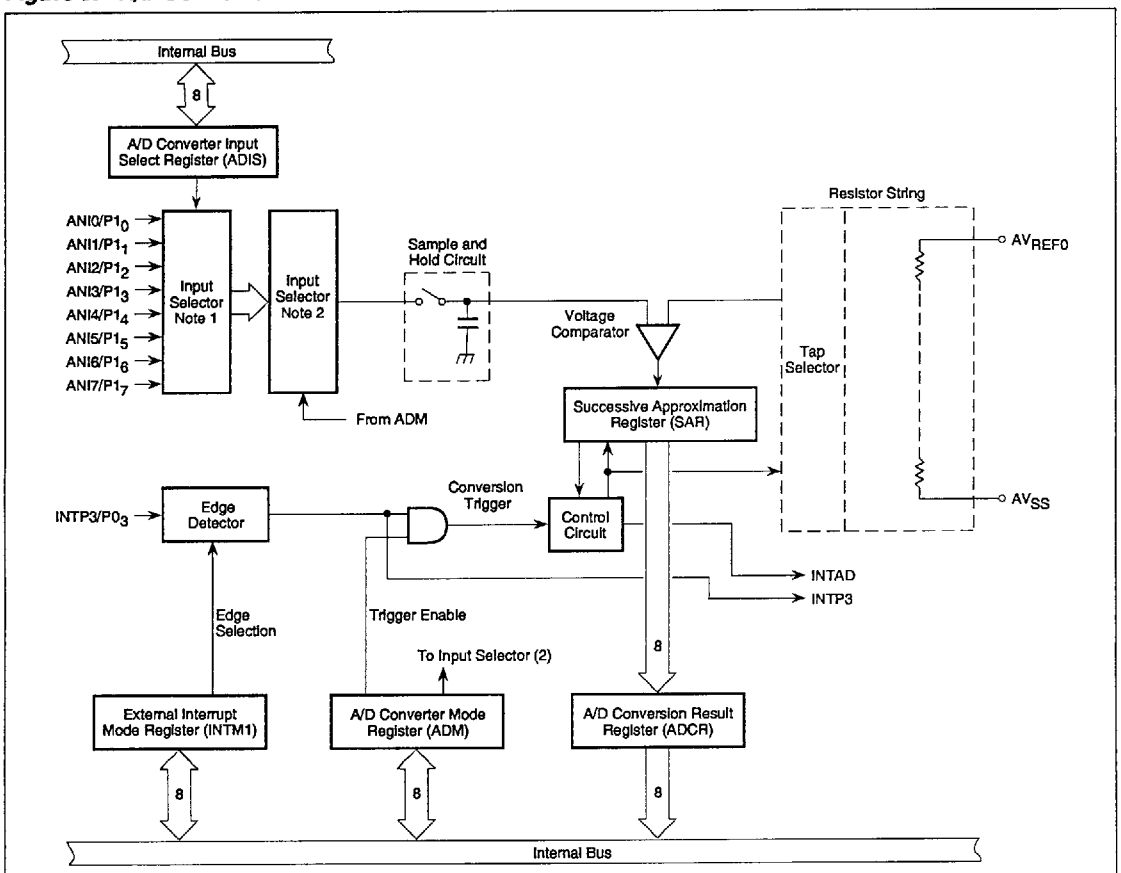
A/D Converter

The μPD78054 family analog-to-digital (A/D) converter (see figure 5) uses the successive-approximation method for converting one of eight multiplexed analog inputs into 8-bit digital data. The conversion time per input is 19.1 μs.

The A/D converter input select register (ADIS) selects the number of inputs that are used in A/D conversion. The remaining inputs are used as ports. The analog input to be converted is selected by programming the A/D converter mode register (ADM). Also, the ADM register is used to select the A/D conversion time. A/D conversion is started by external interrupt INTP3, or by writing to the ADM register. When the conversion is completed, the results are stored in the A/D conversion result register (ADCR) and an INTAD interrupt is generated.

If the A/D converter was started by an external interrupt, the A/D converter stops after the interrupt is generated. If the A/D converter was started by software, the A/D converter repeats the conversion until new data is written to the ADM register.

Figure 5. A/D Converter



Notes:
(1) Selects number of Port 1 inputs to be used for A/D conversion.
(2) Selects the channel for A/D conversion.

83RC-9543B (10/93)

D/A Converter

The μPD78054 family digital-to-analog (D/A) converter (see figure 6) uses an R-2R resistor ladder method for converting the 8-bit digital data to an analog voltage for each of the two independent D/A channels.

The D/A converter mode register (DAM) is used to start and stop D/A conversion as well as selecting "normal mode" or "real-time output mode" for each channel. The 8-bit data to be converted to an analog voltage is written into one of the two D/A conversion value set registers (DACS0 or DACS1). Before D/A conversion is enabled, the respective channel output is in a high-impedance state. The analog output voltage is determined by the following expression where n is 0 or 1 for the respective channel:

$$\text{ANOn (output voltage)} = AV_{REF1} \times \frac{\text{DACSn}}{256} \text{ (volts)}$$

In the normal mode with D/A conversion enabled, the D/A circuit continuously outputs the analog voltage on the ANOn pin representative of the 8-bit value in the DACSn register. When a different 8-bit value is written to the DACSn register, the D/A circuit immediately adjusts the voltage on the ANOn pin to represent the new value. When the D/A conversion operation stops, the output for the respective channel goes to high impedance.

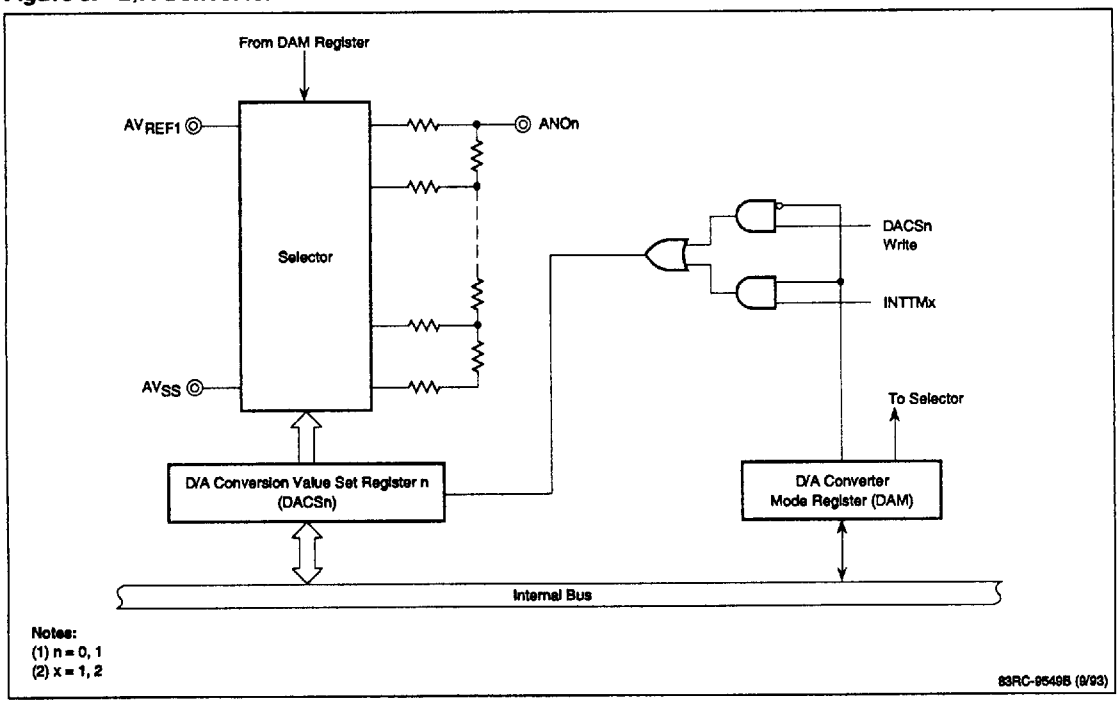
In the real-time output mode when D/A conversion is enabled for a respective channel, the D/A circuit output remains in a high-impedance state. After the interrupt (INTTMx, x = 1 and 2 for D/A channels 0 and 1, respectively) occurs, the D/A circuit outputs the analog voltage on the ANOn pin representative of the 8-bit value in the DACSn register.

When a different 8-bit value is written to the DACSn register, the D/A circuit will adjust the voltage on the ANOn pin to represent the new value after the next interrupt (INTTMx) occurs.

If the data is changed in the DACSn register and the register is read before the next interrupt occurs, the data read will be data previously written before the last interrupt occurred. When the D/A conversion operation stops, the output for the respective channel goes to a high-impedance state.

3f

Figure 6. D/A Converter

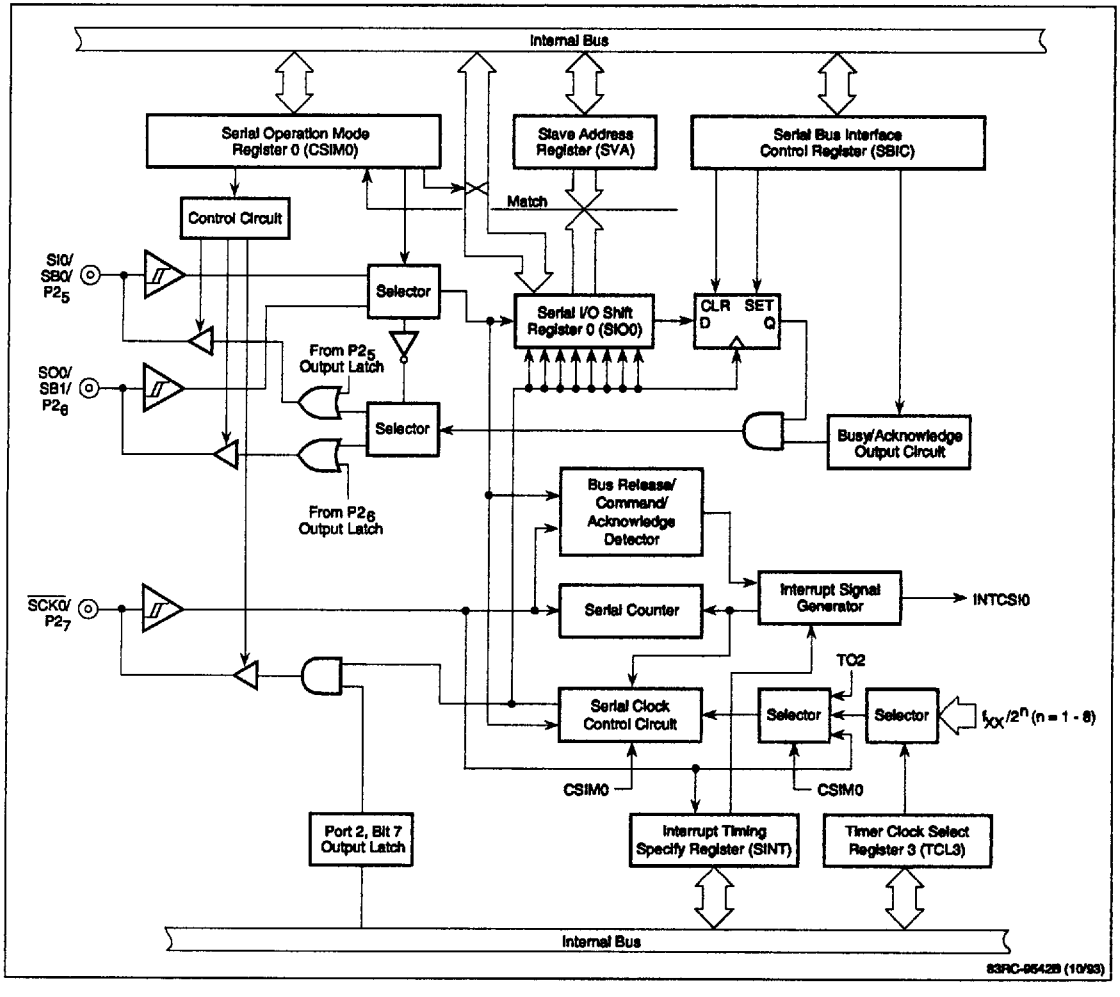


Serial Interfaces

The μPD78054 family has three independent serial interfaces: serial interface 0, serial interface 1, and serial interface 2.

Serial Interface 0. Serial interface 0 is an 8-bit clock synchronous serial interface (figure 7). It can be operated in either a three-wire serial I/O mode, NEC serial bus interface (SBI) mode, or two-wire serial I/O mode. The serial clock can be provided from one of eight internal clocks, the output of 8-bit timer register 2, or the external clock line $\overline{SCK0}$.

Figure 7. Serial Interface 0



In the three-wire serial I/O mode, the 8-bit shift register (SIO0) is loaded with a byte of data and eight clock pulses are generated. The falling edge of these eight pulses shifts the byte of data out of the SO0 line (either MSB or LSB first) while the rising edge of these pulses shifts the data in from the SI0 line providing full-duplex operation. The INTCSI0 interrupt is generated after each 8-bit transfer.

The NEC SBI mode is a two-wire high-speed proprietary serial interface available on most devices in the NEC μPD75xxx and μPD78xxx product lines. Devices are connected in a master/slave configuration (see figure 8). There is only one master device at a time; all others are slaves. The master sends addresses, commands, and data over one of the serial bus lines (SB0 or SB1) using a fixed hardware protocol synchronized with the $\overline{SCK0}$ line . Each slave device of the μPD78054 family can be programmed to respond in hardware to any one of 256 addresses set in its slave address register (SVA). There are also 256 commands and 256 data types. Since all commands are user definable, many software protocols, simple or complex, can be defined. It is even possible to develop commands to change a slave into a master and the previous master into a slave.

The two-wire serial I/O mode provides half-duplex operation using either the SB0 or SB1 line and the $\overline{SCK0}$ line. Communication format and handshaking can be handled in software by controlling the output levels of the data and clock lines between transfers. For data transmission, the 8-bit shift register (SIO0) is loaded with a byte of data and eight clock pulses are generated. The falling edge of these eight pulses shifts the byte of data out of either the SB0 or SB1 line (MSB first). In addition, this byte of data is also shifted back into SIO0 on the rising edge of these pulses providing a way of verifying that the transmission was correct.

For data reception, the SIO0 register is preloaded with the value FFH. As this data value is shifted out on the falling edge of the serial clock, it disables the n-channel open-drain driver. This allows the receive data to be driven on to the serial line and shifted into the SIO0 register on the rising edge of the serial clock. The INTCSI0 interrupt is generated after each 8-bit transfer.

Serial Interface 1. Serial interface 1 is also an 8-bit clock synchronous serial interface (figure 9). It can be operated in either a three-wire serial I/O mode, or three-wire serial I/O mode with automatic transmit/receive. The serial clock can also be provided from one of eight internal clocks (common clock for both interfaces), the output of 8-bit timer register 2, or the external clock line $\overline{SCK1}$.

In the three-wire serial I/O mode, the 8-bit shift register (SIO1) is loaded with a byte of data and eight clock pulses are generated. The falling edge of these eight pulses shifts the byte of data out of the SO1 line (either MSB or LSB first) while the rising edge of these pulses shifts the data in from the SI1 line providing full-duplex operation. The INTCSI1 interrupt is generated after each 8-bit transfer.

In the three-wire serial I/O mode with automatic transmit/receive, up to 32 bytes of data can be transferred with minimal CPU overhead. The data to be transmitted and received is stored in the buffer RAM. Handshaking using either the BUSY input line, the strobe (STB) output line, or both, can be selected by the program. Error detection of bit drift due to noise is available for each byte transferred when using the BUSY input line. This automatic transmit/receive mode is ideally suited for transferring data to/from external peripheral devices such as onscreen display (OSD) and LCD controller/driver devices.

3f

While in three-wire serial interface mode with automatic data transfer, the interface can be operated as either a full-duplex interface or a transmit-only interface in single or repetitive operation mode. In the full-duplex mode, a byte of data is transferred from the first location in the buffer RAM and shifted out of the SO1 line (either MSB or LSB first) while the received data is shifted into the SI1 line and stored back in the first buffer location. After the preset number of bytes has been transferred, the INTCSI1 interrupt is generated.

Figure 8. SBI Mode Master/Slave Configuration

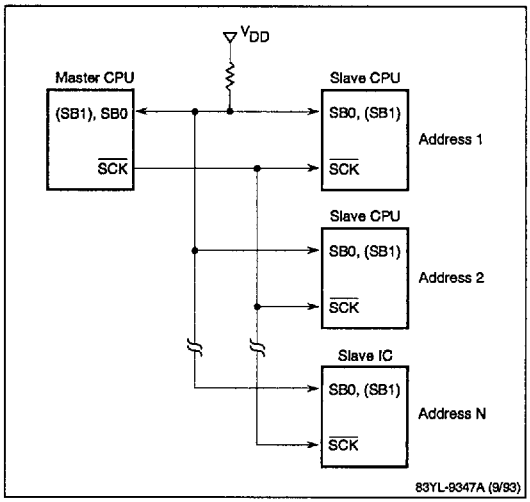
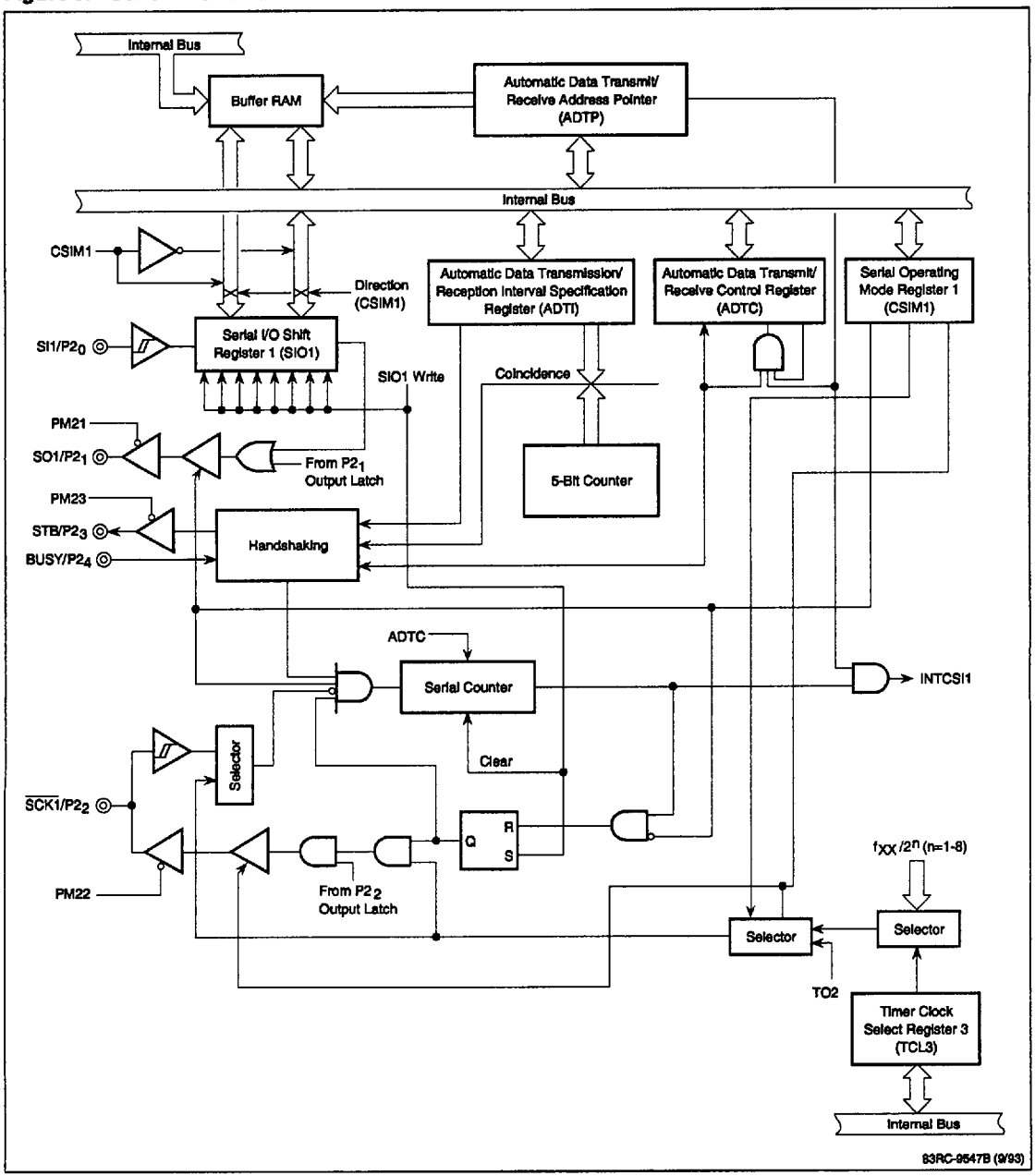


Figure 9. Serial Interface 1



In single-operation transmit mode, the preset number of bytes from the buffer RAM are transmitted out of the SO1 line (either MSB or LSB first) and the INTCSI1 interrupt is generated after all bytes are transferred. In the repetitive operation transmit mode, data in the buffer is transmitted repeatedly.

Serial Interface 2. Serial interface 2 is an 8-bit serial interface (figure 10) that can be operated in either a UART mode or a three-wire serial I/O mode. The internal baud rate generator circuit (figure 11) scales an internal clock to provide standard baud rates from 75 to 38400 bps. For non-standard baud rates, the internal baud rate generator circuit scales an external clock input on the ASCK pin. The output of the baud rate generator circuit is used as the data transmission and sampling clock in the UART mode or the data clock in the three-wire serial I/O mode.

In the UART mode, half or full-duplex operation with various protocols is programmable. The asynchronous serial interface mode register (ASIM) is used to specify the number of stop bits (1 or 2), data character length (7 or 8 bits), parity (none, even, odd, or 0), receive operation control (enable or disable), and transmit operation control (enable or disable). The ASIM register is also used to enable or disable the generation of an interrupt when a reception error occurs and whether an internal or external clock will be supplied to the baud rate generator circuit.

A transmit operation is started by writing a data character to the transmit shift register (TXS) register. The start bit, parity bit, and stop bit(s) are automatically added by the hardware to the data character in the TXS register. The data in the TXS register is shifted out of the TxD line and when the TXS register is empty, a transmission complete interrupt (INTST) is generated.

When the receive operation control is enabled, the RxD line is sampled using the clock specified by the ASIM register. When the RxD line is detected low, sampling starts at the midpoint of each bit. If the first sample yields a low, it is identified as a start bit. The RxD line continues to be sampled at the midpoint of each bit. Reception of one frame of data is complete when the data character bits, parity bit (if being transmitted), and one stop bit are detected after the start bit. Even if the protocol is set for two stop bits, only one stop bit is used for the end of reception detection.

When one frame of data has been received, the received data in the shift register is transferred to the receive buffer (RXB) and a reception complete vectored interrupt (INTSR) is generated even if an error (parity and/or framing) is detected. The data must be read from the RXB register before another frame is received or an overrun error will be generated. If an error occurs, the appropriate flag is set in the asynchronous serial interface status register (ASIS) and a receive error interrupt (INTSER) is generated.

In the three-wire I/O mode, the TXS register is loaded with a byte of data and eight clock pulses are generated. The falling edge of these eight pulses shifts the byte of data out of the SO2 line (either MSB or LSB first) while the rising edge of these pulses shifts the data in from the SI2 into the RXS register line providing full-duplex operation. The INTCSI2 interrupt is generated after each 8-bit transfer.

Timers

The μPD78054 family has one 16-bit timer/event counter, two 8-bit timer/event counters that can be combined for use as a 16-bit timer/event counter, a watch timer and a watchdog timer. All of these can be programmed to count a number of prescaled values of the main system clock. In addition, the watch timer can also count the subsystem clock. All of the timer/event counters can count external events.

16-Bit Timer/Event Counter 0. Timer/event counter 0 (figure 12) consists of a 16-bit counter (TM0), two 16-bit capture registers (CR00, CR01), control registers TMC0, TOC0, and CRC0, clock select register CLC0, and a timer output (TO0). Timer 0 can be used as an interval timer, to count external events on the timer input (TI00) pin, to output a programmable square wave, a 14-bit pulse-width modulated output, a one-shot pulse, or to measure pulse widths.

Figure 10. Serial Interface 2

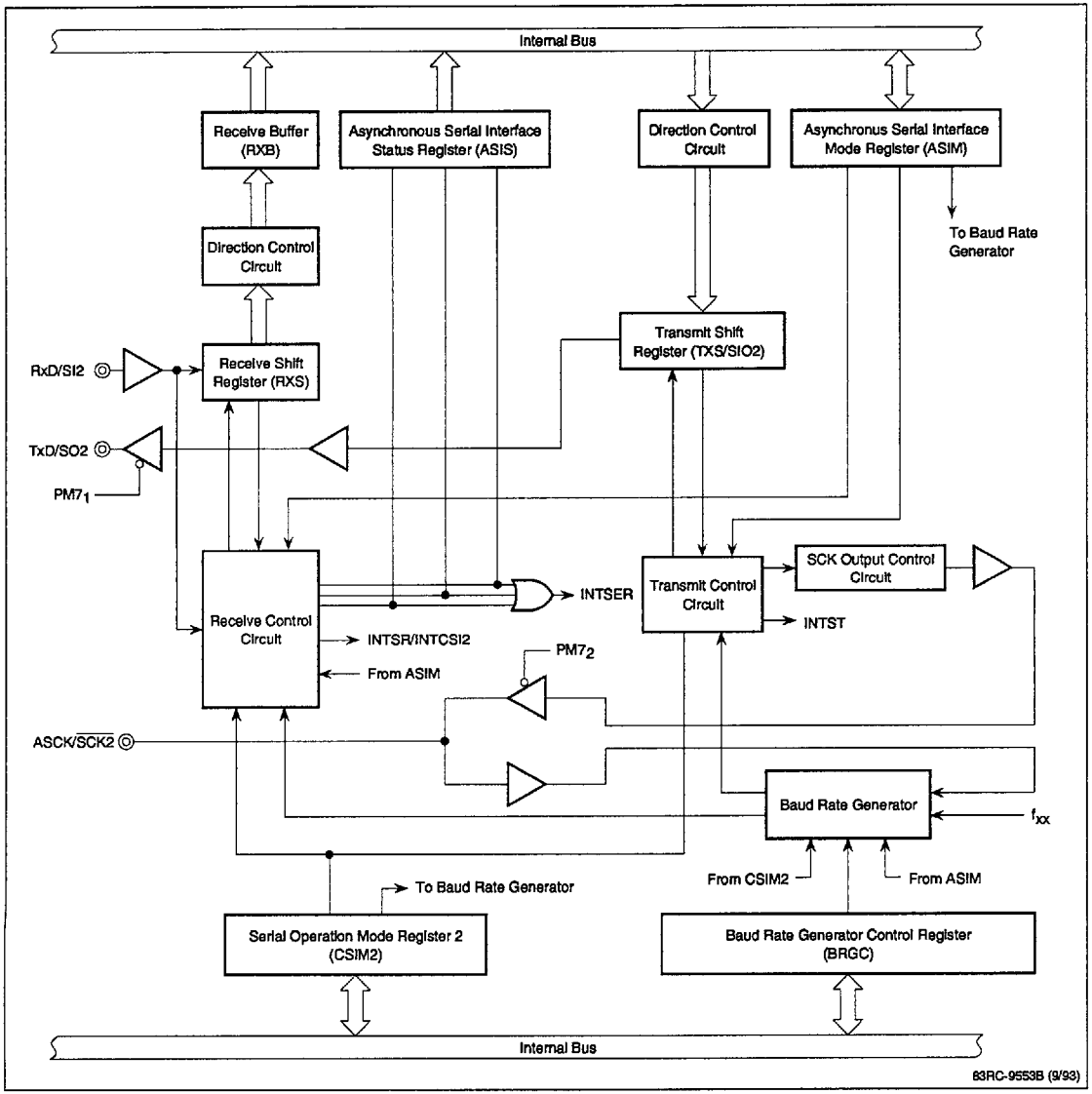
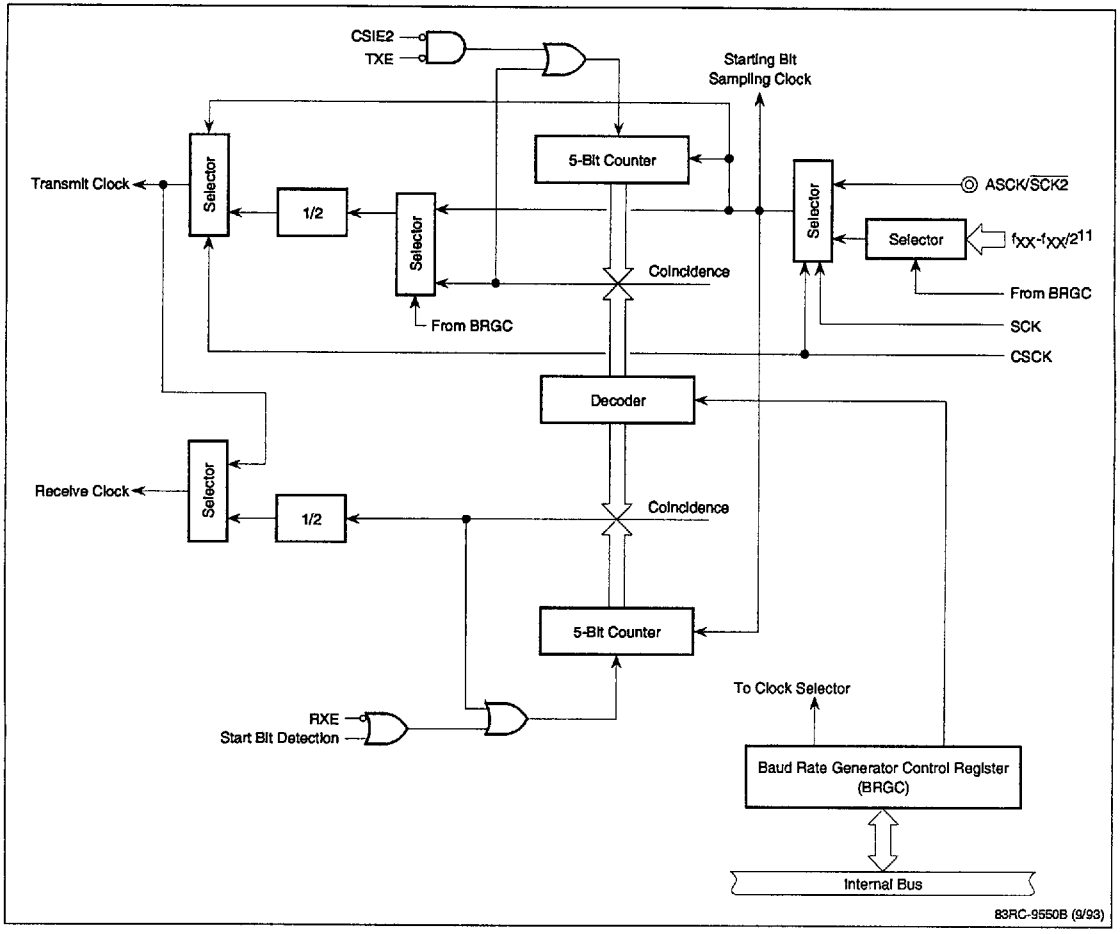
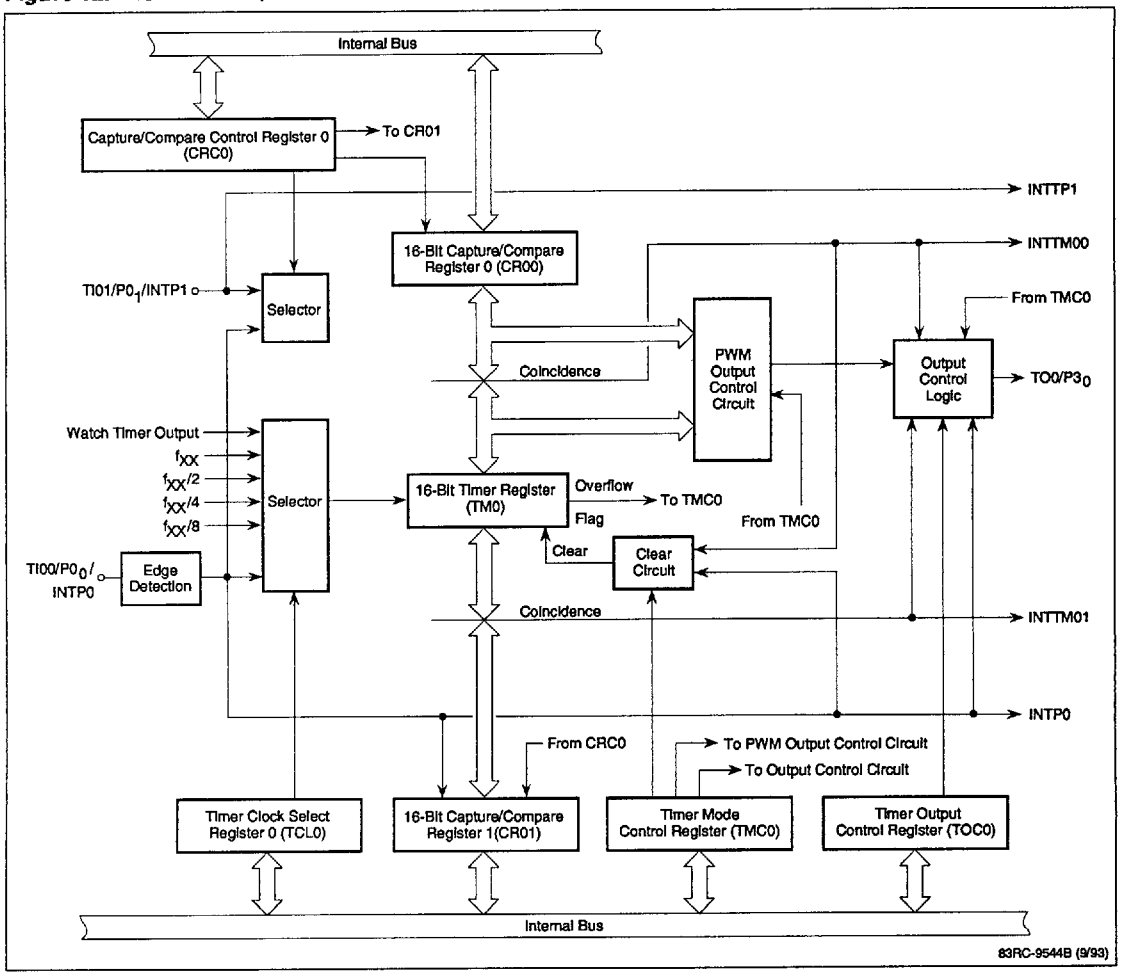


Figure 11. Internal Baud Rate Generator



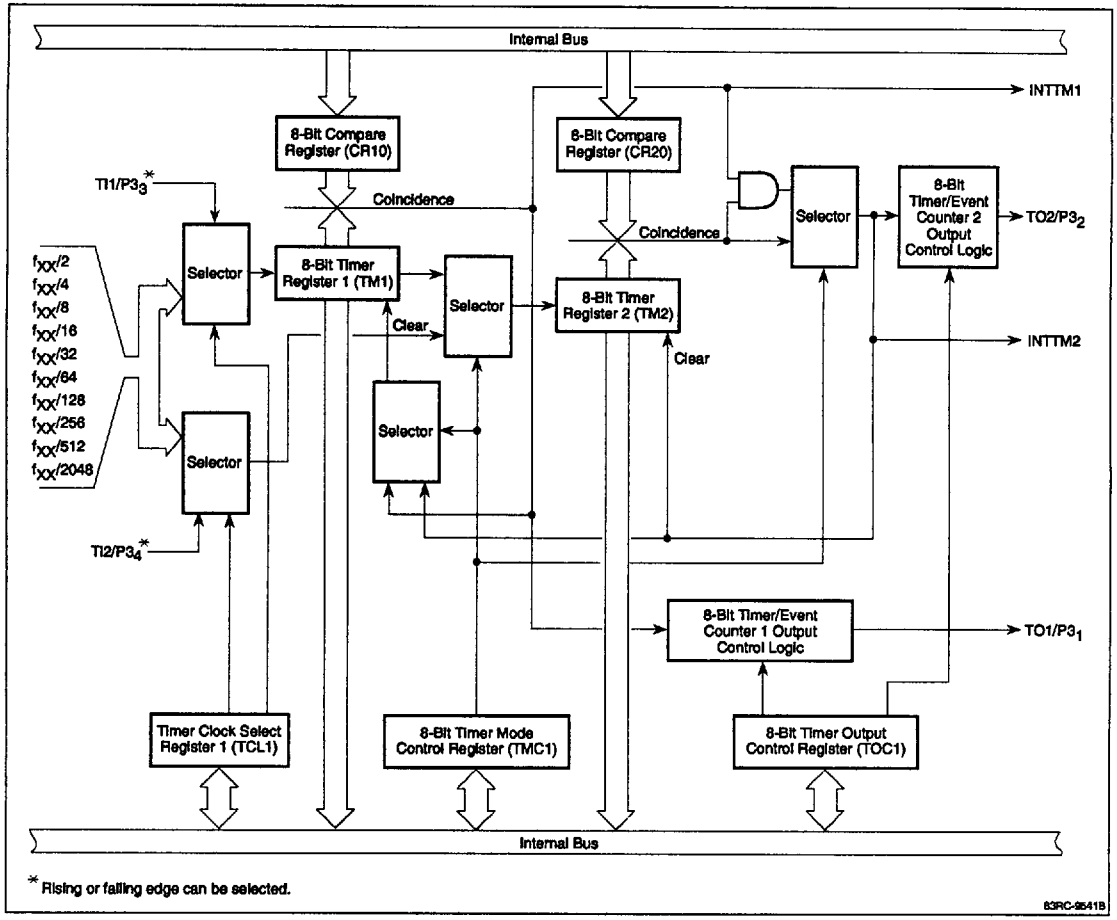
3f

Figure 12. 16-Bit Timer/Event Counter 0



8-Bit Timer/Event Counters 1 and 2. Timer/event counters 1 and 2 (figure 13) each consist of an 8-bit timer (TM1 or TM2), an 8-bit compare register (CR10 or CR20), and a timer output (TO1 or TO2). The timers are controlled by registers TCL1, TMC1, TOC1 via five selectors. Timer/event counters 1 and 2 can each be used as an 8-bit interval timer, to count external events on the timer input pins (TI1 or TI2), or to output a programmable square wave. In addition, timers 1 and 2 also can be combined as a 16-bit timer/event counter and used as a 16-bit interval timer, to count external events on TI1, or to output a programmable square wave on TO2.

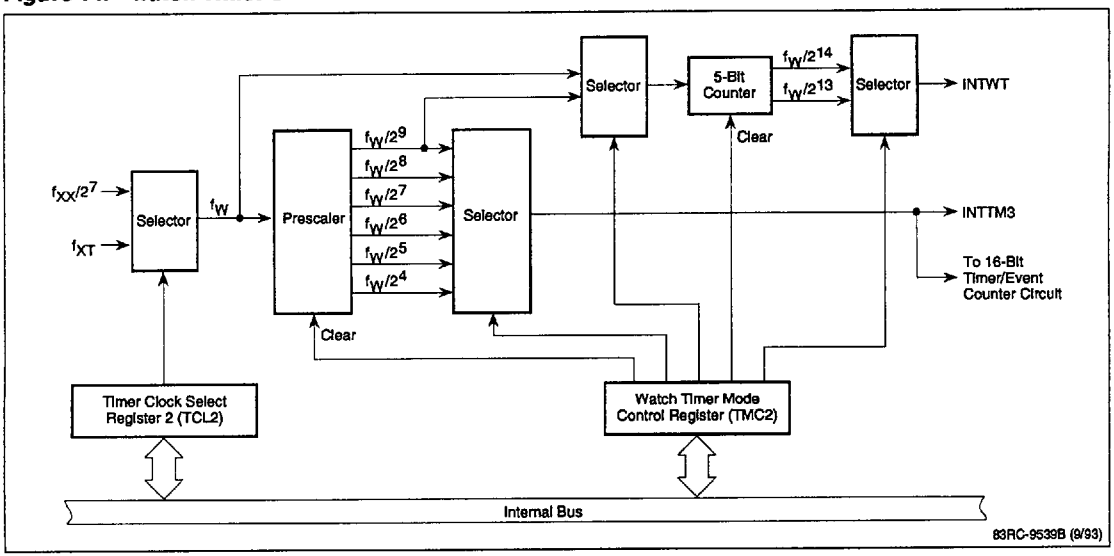
Figure 13. 8-Bit Timer/Event Counters 1 and 2



Watch Timer 3. Watch timer 3 (figure 14) is a 5-bit timer that can be used as a time source to keep track of time of day, to release the STOP or HALT modes at regular intervals, or to initiate any other task that must be performed at regular intervals. When driven by the subsystem clock, the watch timer continues to operate in the STOP mode.

The watch timer can function as both a watch timer and an interval timer simultaneously. When used as a watch timer, interrupt request INTWT can be generated using the main or subsystem clock of 0.5 or 0.25 seconds. When used as an interval timer, vectored interrupt request INTTM3 is generated at preselected time intervals. With a main system clock of 4.19 MHz and $f_{XX} = f_X$ or if using the subsystem clock of 32.768 kHz, the following time intervals can be selected: 489μ s, 978 μs, 1.96 ms, 3.91 ms, 7.82 ms or 15.6 ms.

Figure 14. Watch Timer 3



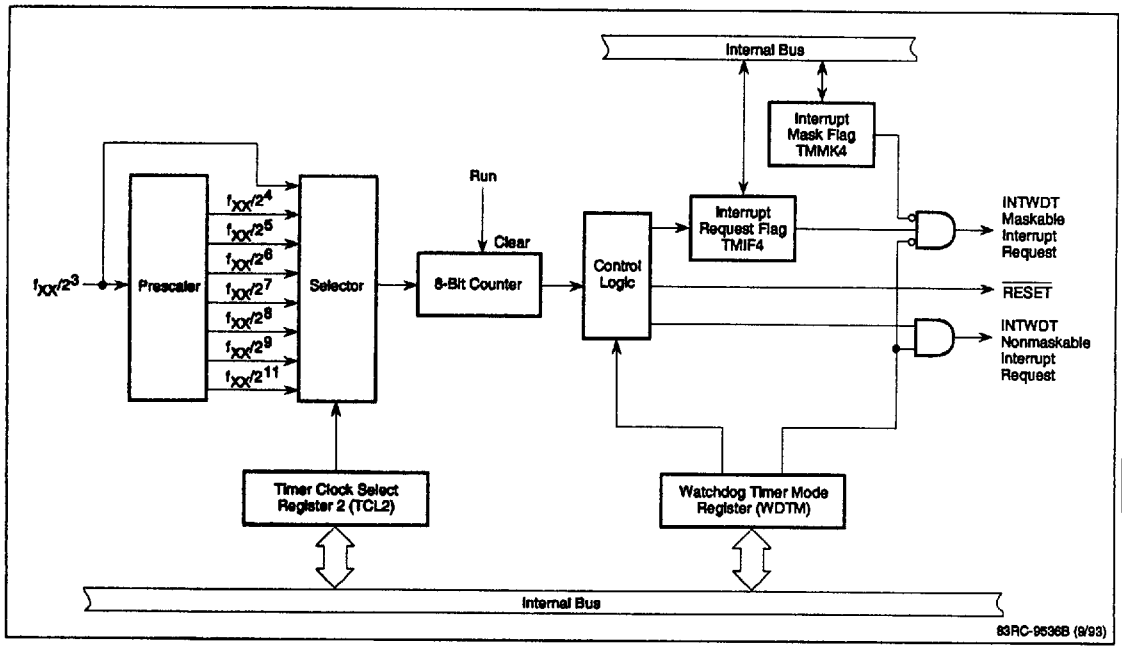
Watchdog Timer. The watchdog timer (figure 15) can be used as either a watchdog timer or an interval timer. When used as a watchdog timer it protects against program runaway. It can be selected to generate a nonmaskable interrupt (INTWDT), which vectors to address 0004H, or to generate an internal reset signal, which vectors to the restart address 0000H if the timer is not cleared by the program before it overflows. Eight program-selectable intervals based on the main system clock are available. With a main system clock of 4.19 MHz and $f_{XX} = f_X$, they are 0.489, 0.978, 1.96, 3.91, 7.82, 15.6, 31.3, and 125 ms. With a main system clock of 4.19 MHz and $f_{XX} = f_X/2$, they are 0.978, 1.96, 3.91, 7.82, 15.6, 31.3, 62.6, and 250 ms. Once the watchdog timer is initialized and started, the timer's mode cannot be changed and the timer can only be stopped by a reset.

When used as an interval timer, maskable interrupts (INTWDT) which vector to address 0004H are generated repeatedly at a preset interval. The time intervals available are the same as in the watchdog timer mode.

Programmable Clock Output

The μPD78054 family has a programmable clock output (PCL) that can be used for carrier output for remote controlled transmissions or as a clock output for peripheral devices. The main system clock (f_{XX}) divided by 1, 2, 4, 8, 16, 32, 64, or 128 or the subsystem clock (f_{XT}) can be output on the PCL pin. See figure 16. If the main system clock is 4.19 MHz and $f_{XX} = f_X$, the following frequencies are available: 4.19 MHz, 2.1 MHz, 1.05 MHz, 524 kHz, 262 kHz, 131 kHz, 65.5 kHz, and 32.7 kHz. With a main system clock of 4.19 MHz and $f_{XX} = f_X/2$, the following frequencies are available: 2.1 MHz, 1.05 MHz, 524 kHz, 262 kHz, 131 kHz, 65.5 kHz, 32.7 kHz and 16.4 kHz. With a subsystem clock of 32.768 kHz, 32.768 kHz is also available.

Figure 15. Watchdog Timer



3f

Figure 16. Programmable Clock Output

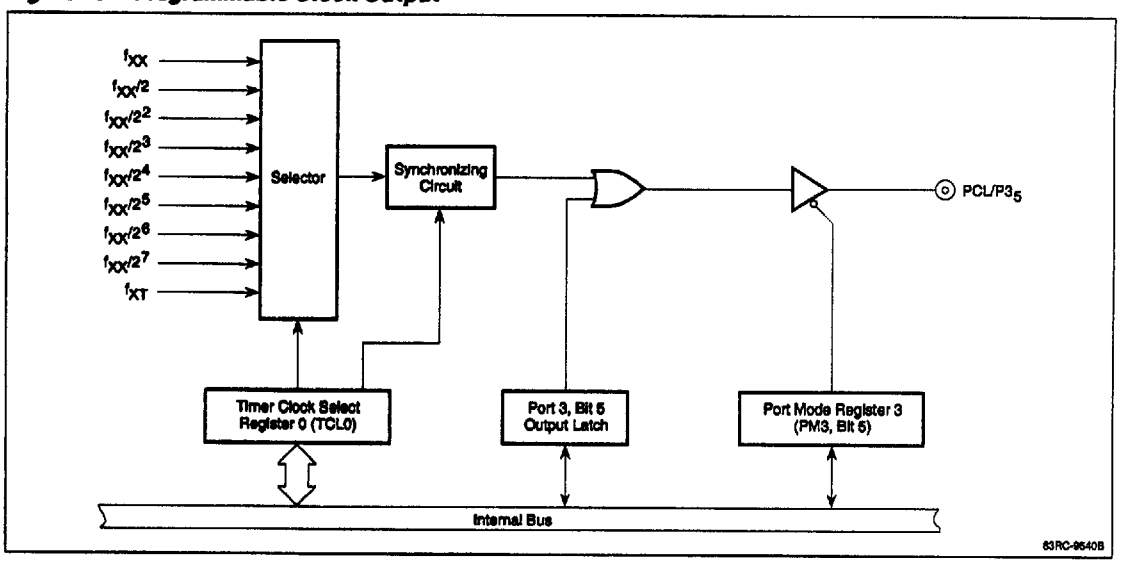


Figure 17. Buzzer Output

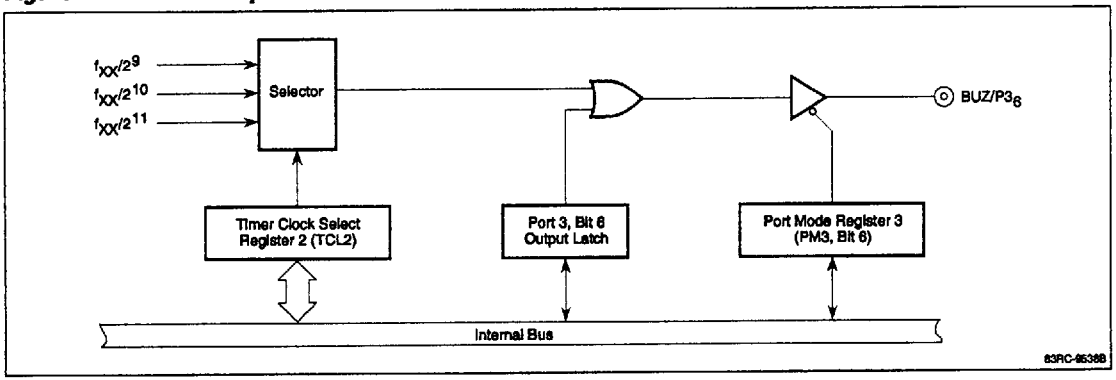
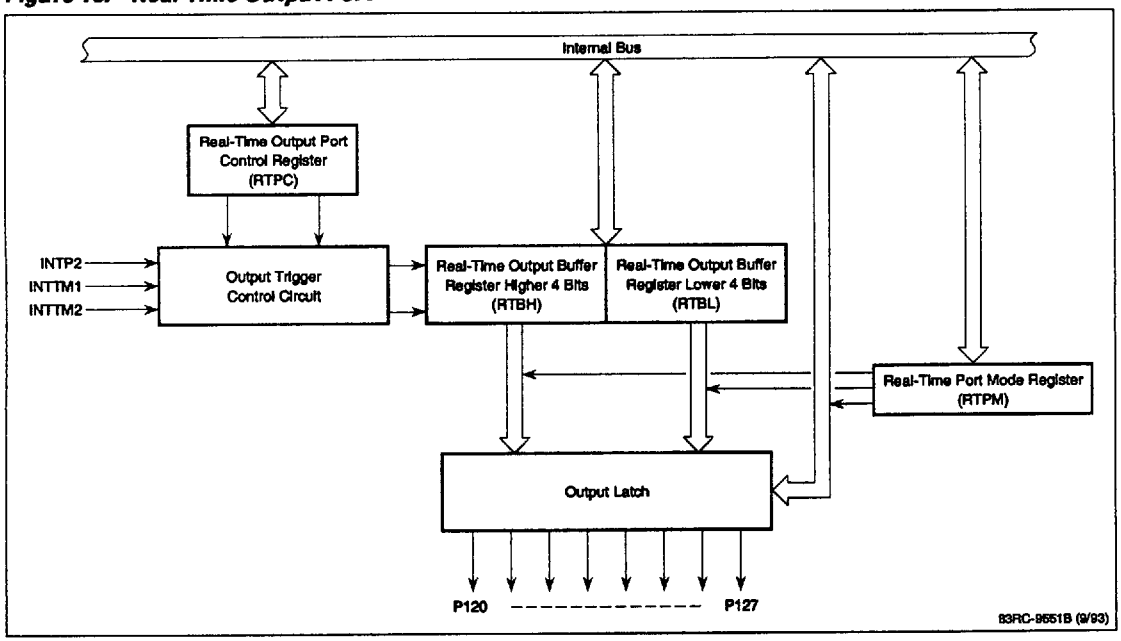


Figure 18. Real-Time Output Port



Buzzer Output

The μPD78054 family also has a programmable buzzer output (BUZ). The buzzer output frequency can be programmed to be equal to the main system clock (f_{XX}) divided by 512, 1024, or 2048. With a main system clock of 4.19 MHz and $f_{XX} = f_X$, the buzzer can be set to 8.2, 4.1 or 2.0 kHz. With a main system clock of 4.19 MHz and $f_{XX} = f_X/2$, the buzzer can be set to 4.1, 2.0, or 1.0 kHz. See figure 17.

Real-Time Output Port

The real-time output port (figure 18) shares pins with port 12. Each bit of port 12 is specified by the real-time output port mode register (RTPM) to be used in the port mode or real-time output port mode. If the real-time output port mode is selected, the real-time output port control register (RTPC) is used to specify the high and low nibbles to be treated separately or together. In the real-time output port mode, the previously written data in the real-time output buffer registers (RTBH, RTBL) is transferred to the output latch simultaneously with the generation of either a timer interrupt (INTTM1, INTTM2) or external interrupt (INTP2).

Interrupts

The μPD78054 family has 21 maskable hardware interrupt sources; 8 are external and 13 are internal. Of these 21 interrupt sources, 19 cause a vectored interrupt while the 2 testable inputs only generate an interrupt request. All of the 21 maskable interrupts can be used to release the HALT mode except INTP0. INTP0 cannot be used to release the STOP mode and cannot release the HALT mode when SCS = 0. In addition, there is one nonmaskable interrupt from the watchdog timer, one software interrupt, and a reset interrupt. The watchdog timer overflow interrupt (interrupt vector table address 0004H) can be initialized to be a nonmaskable interrupt or the highest default priority maskable interrupt. The software interrupt, generated by the BRK instruction, is not maskable. See table 3 and figure 19.

3f

Table 3. Interrupt Sources and Vector Addresses

Type of Request	Default Priority	Signal Name	Interrupt Source	Location	Vector Address	Interrupt* Configuration
Restart	—	RESET	$\overline{\text{RESET}}$ input pin	External	0000H	—
		INTWDT	Watchdog timer overflow (when reset mode selected)	Internal		
Nonmaskable	—	INTWDT	Watchdog timer overflow (when nonmaskable interrupt selected)	Internal	0004H	A
Maskable	0	INTWDT	Watchdog timer overflow (when interval timer selected)	Internal	0004H	B
	1	INTP0	External interrupt edge detection	External	0006H	C
	2	INTP1	External interrupt edge detection	External	0008H	D
	3	INTP2	External interrupt edge detection	External	000AH	D
	4	INTP3	External interrupt edge detection	External	000CH	D
	5	INTP4	External interrupt edge detection	External	000EH	D
	6	INTP5	External interrupt edge detection	External	0010H	D
	7	INTP6	External interrupt edge detection	External	0012H	D
	8	INTCSI0	End of clocked serial interface 0 transfer	Internal	0014H	B
	9	INTCSI1	End of clocked serial interface 1 transfer	Internal	0016H	B
	10	INTSER	Serial interface 2 UART reception error	Internal	0018H	B
	11	INTSR	End of serial interface 2 UART reception	Internal	001AH	B
		INTCSI2	End of serial interface 2 three-wire transfer			
	12	INTST	End of serial interface 2 UART transmission	Internal	001CH	B
	13	INTTM3	Watch timer reference time interval signal	Internal	001EH	B
	14	INTTM00	16-bit timer/event counter capture/compare (CR00) coincidence signal	Internal	0020H	B

Table 3. Interrupt Sources and Vector Addresses (cont)

Type of Request	Default Priority	Signal Name	Interrupt Source	Location	Vector Address	Interrupt* Configuration
Maskable	15	INTTM01	16-bit timer/event counter capture/compare (CR01) coincidence signal	Internal	0022H	B
	16	INTTM1	8-bit timer/event counter 1 coincidence signal	Internal	0024H	B
	17	INTTM2	8-bit timer/event counter 2 coincidence signal	Internal	0026H	B
	18	INTAD	End of A/D conversion	Internal	0028H	B
Software	—	—	BRK instruction	Internal	003EH	E
Test input	—	INTWT	Watch timer overflow	Internal	—	F
	—	INTPT4	Port 4 falling edge detection	External	—	F

* See figure 19

Interrupt Servicing. The μPD78054 family provides two levels of programmable hardware priority control and services all interrupt requests, except the two testable interrupts (INTWT and INTPT4) using vectored interrupts. The programmer can choose the priority of servicing each maskable interrupt by using the interrupt control registers.

Interrupt Control Registers. The μPD78054 family has three 3-byte interrupt control registers. The interrupt request flag registers (IF0L, IF0H, and IF1L) contain an interrupt request flag for each interrupt. The interrupt mask registers (MK0L, MK0H, and MK1L) are used to enable or disable any individual interrupt. The priority flag registers (PR0L, PR0H, and PR1L) can be used to specify a high or a low priority level for each interrupt except the two testable interrupts (INTWT and INTPT4).

Five other 8-bit registers are associated with interrupt processing. The key return mode register (KRM) contains the KRIF interrupt request flag associated with falling-edge detection on port 4 and the KRMK mask flag used to enable or disable clearing of the standby mode if a falling edge is detected on port 4. The external interrupt mode registers (INTM0 and INTM1) are used to select a rising, falling, or both edges as the valid edge for each of the external interrupts INTP0 to INTP6. The sampling clock select register (SCS) is used to select a sampling clock for the noise eliminator circuit on external interrupt INTP0.

The IE and the ISP bit of the program status word are also used to control interrupts. If the IE bit is 0, all maskable interrupts are disabled. The IE bit can be set or cleared using the EI and DI instructions, respectively, or by directly writing to the PSW. The IE bit is cleared each time an interrupt is accepted. The ISP bit is used by hardware to hold the priority level flag of the interrupt being serviced.

Interrupt Priority. If the watchdog timer overflow interrupt (INTWDT) has been initialized to be a nonmaskable interrupt, it has priority over all other interrupts. Two hardware-controlled priority levels are available for all maskable interrupts that generate a vectored interrupt (i.e., all except the two testable interrupts). Either a high or a low priority level can be assigned by software to each of the maskable interrupts.

Interrupt requests of the same priority or a priority higher than the processor's current priority level are held pending until interrupts in the current service routine are enabled by software or until one instruction has been executed after returning from the current service routine. Interrupt requests of a lower priority are always held pending until one instruction has been executed after returning from the current service routine.

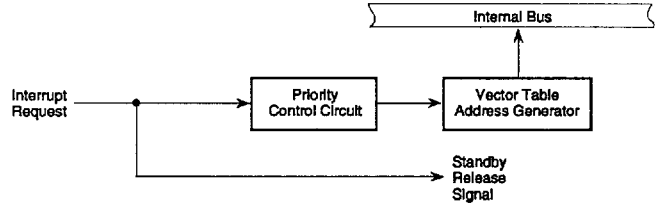
The default priorities listed in table 3 are fixed by hardware and are effective only when it is necessary to choose between two interrupt requests of the same software-assigned priority. For example, the default priorities would be used after the completion of a high-priority routine, if two interrupts of the same software priority were pending.

The software interrupt, initiated by the BRK instruction, is executed regardless of the processor's priority level and the state of the IE bit. It does not alter the processor's priority level.

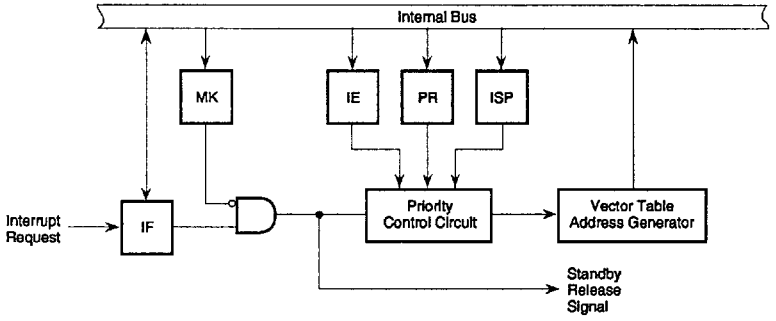
Vectored Interrupt Servicing. When a vectored interrupt is acknowledged, the program status word and the program counter are saved on the stack, the processor's priority is set to that specified for the interrupt, the IE bit in the PSW is set to zero, and the routine whose address is in the interrupt vector table is entered. At the completion of the service routine, the RETI instruction (RETB instruction for the software interrupt) reverses the process and the μPD78054 family microcomputer resumes the interrupted routine.

Figure 19. Interrupt Configurations

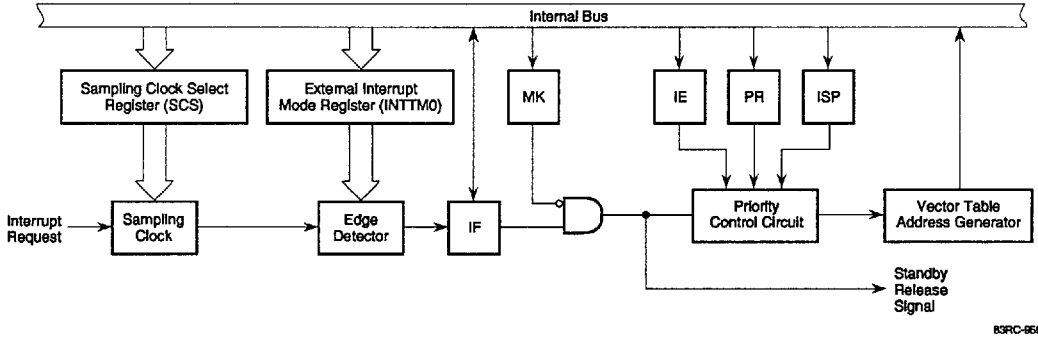
Type A: Internal nonmaskable interrupt



Type B: Internal maskable interrupt



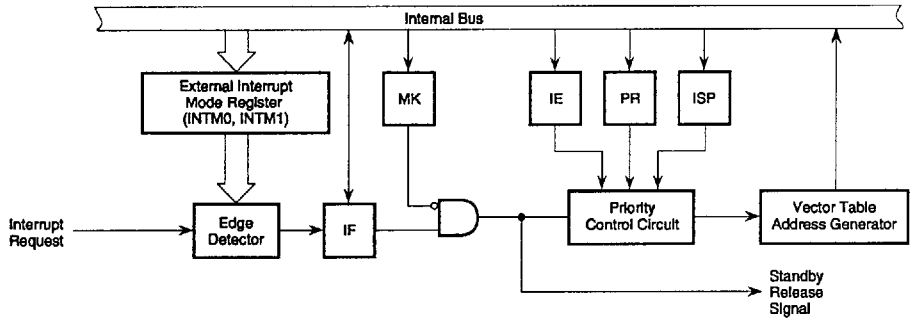
Type C: External maskable interrupt (INTP0)



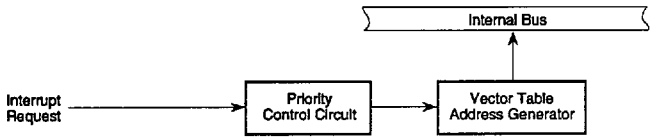
3f

Figure 19. Interrupt Configurations (cont)

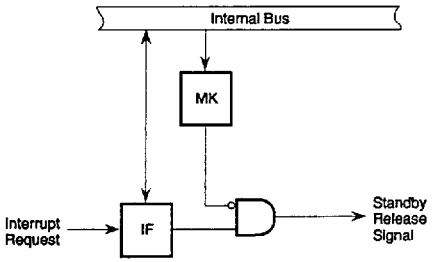
Type D: External maskable interrupt (except INTP0)



Type E: Software Interrupt



Type F: Test Input



Abbreviations:

IF: Interrupt request flag
IE: Interrupt enable flag
ISP: In-service priority flag
MK: Interrupt mask flag
PR: Priority specify flag

83RC-9591B-2

Standby Modes

HALT, STOP, and data retention modes are provided to reduce power consumption when CPU action is not required.

The HALT mode is entered by executing a HALT instruction while the CPU is operating from the main system or subsystem clock. In HALT mode, the CPU clock is stopped while the main system and the subsystem clock continue to run. The HALT mode is released by any unmasked interrupt request, a nonmaskable interrupt request, an unmasked test input, or an external reset pulse.

Power consumption may be further reduced by using the STOP mode. The STOP mode is entered by executing a STOP instruction while operating from the main system clock. In STOP mode, the main system clock input pin X1 is internally grounded stopping both the CPU and the peripheral hardware clock. The STOP mode is released by any unmasked interrupt request (except INTP0 if register SCS = 0), a nonmaskable interrupt request, an unmasked test input, or an external reset pulse. Any peripheral using the main oscillator as a clock source will also be disabled in the STOP mode and interrupts from such a peripheral can not be used to exit the STOP mode. Table 4 summarizes both the HALT and STOP standby modes.

When exiting the STOP mode, a wait time occurs before the CPU begins code execution to allow the main system clock oscillator circuit to stabilize. The oscillation stabilization time is selected by programming the OSTS register with one of five values before entering the STOP mode; the values range from 0.8 msec to 52.4 msec at f_X = 5 MHz.

Table 4. Standby Mode Operation Status

Item	HALT Mode	STOP Mode
Setting instruction	HALT instruction	STOP instruction
System clock when setting	Main system or subsystem clock	Main system clock
Clock oscillator	Main system and subsystem clocks can oscillate; CPU clock is stopped.	Subsystem clock can oscillate; CPU clock and main system clock are stopped.
CPU	Operation stopped	Operation stopped
Ports	Maintain previous state	Maintain previous state
16-bit timer/event counter	Operational from main system clock, or with watch timer output, or TI00 selected as the count clock	Operational only with watch timer output or TI00 selected as count clock.
8-bit timer/event counters	Operational from main system clock or with TI1 and TI2 selected as the count clock	Operational only with TI1 and TI2 selected as count clock
Watch timer	Operational from main system clock or with f_{XT} as count clock	Operational only with f_{XT} as count clock
Watchdog timer	Operational from main system clock	Operation stopped
Serial interface 0	Operational from main system clock or with external clock	Operational only with external clock
Serial interface 1	Operational from main system clock; no automatic transmit/ receive mode	Operational only with external clock; no automatic transmit/ receive mode
Serial interface 2	Operational from main system clock or with external clock	Operation stopped
A/D converter	Operational from main system clock	Operation stopped
D/A converter	Operational	Operational
Real-time output port	Operational	Operational with external trigger or when TI1 and TI2 count clocks are selected
External interrupts	Operational except for INTP0 when its sampling clock is based on the CPU clock	INTP0 not operational; INTP1 to INTP6 operational

3f

Once in the STOP mode, power consumption can be further minimized by lowering the power supply voltage V_{DD} to 2 volts. This places the device in the data retention mode. The contents of Internal RAM and the registers are retained. This mode is released by first raising V_{DD} to the proper operating range and then releasing the STOP mode.

External Reset

The μPD78054 family is reset by taking the $\overline{\text{RESET}}$ pin low or by an overflow of the watchdog timer (if enabled). The $\overline{\text{RESET}}$ input pin is a schmitt-trigger input with hysteresis characteristics to protect against spurious system resets caused by noise. On power-up, the $\overline{\text{RESET}}$ pin must remain low for a minimum of 10 μs after the power supply reaches its operating voltage.

There is no functional difference between an external reset and an internal reset caused by the overflow of the watchdog timer. In both cases, the main system clock oscillation is stopped and the subsystem clock oscillation continues. During reset, the program counter is loaded with the address contained in the reset vector (addresses 0000H, 0001H). Once the reset is cleared and the oscillation stabilization time of $2^{16}/f_X$ has elapsed, program execution starts at that address.