Order this document
by MC68HC11N4TS/D

MC68HC11N4

Technical Summary

8-Bit Microcontroller

The MC68HC11N4 high-performance microcontroller unit (MCU) is an enhanced, ROM-based derivative of the MC68HC11K4 and, as shown in the block diagram, includes many additional features. The MC68HC11N4 MCU with nonmultiplexed bus is a high-speed, low-power MCU capable of operating at speeds up to 4 MHz. Its fully static design allows it to operate at frequencies down to dc.

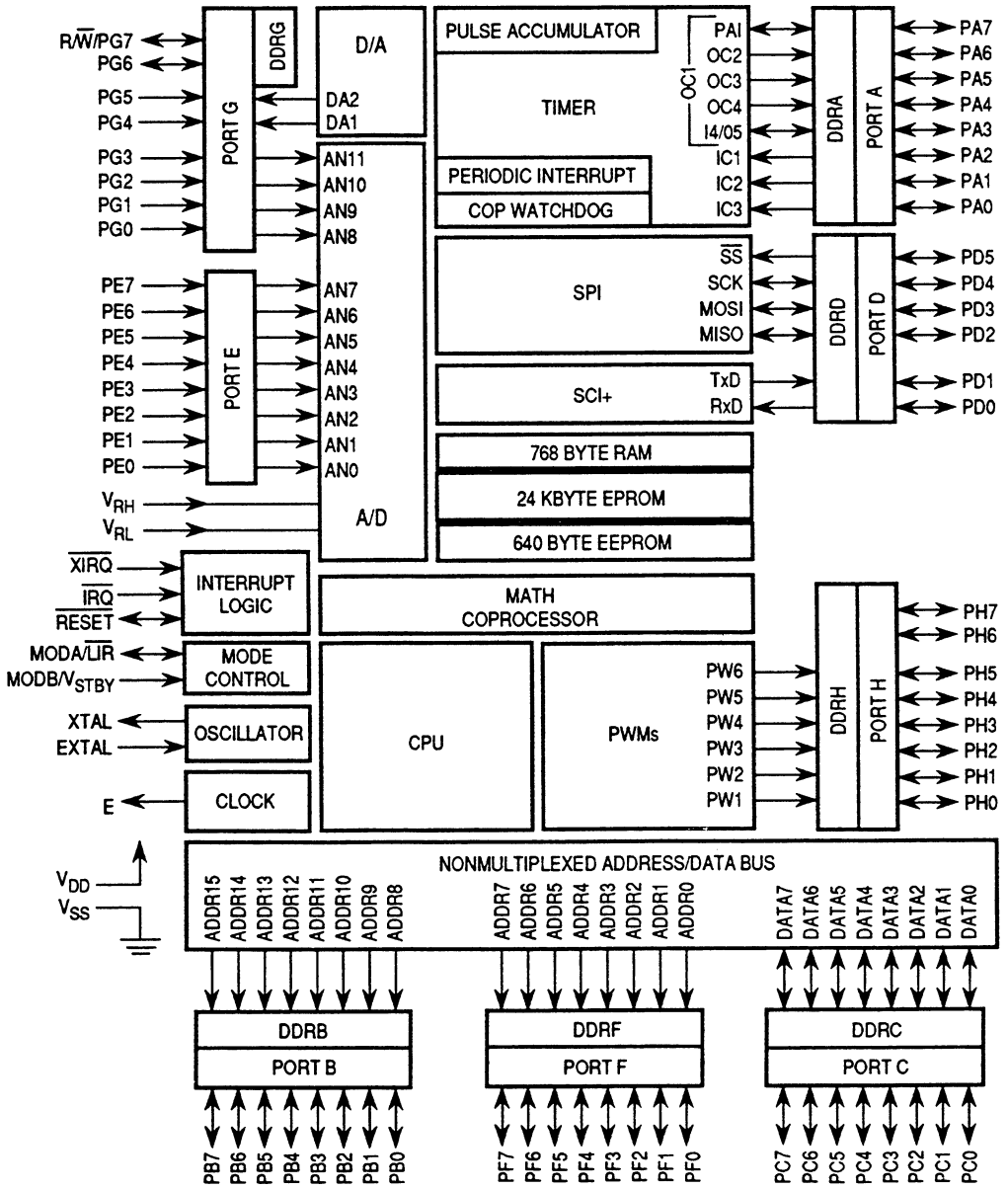
Features

- M68HC11 Central Processing Unit (CPU)
- On-chip 16-Bit Math Coprocessor
- Power Saving STOP and WAIT Modes
- 24 KBytes On-Chip ROM
- 640 Bytes Electrically Erasable Programmable ROM (EEPROM)
- 768 Bytes RAM (All Saved During Standby)
- Nonmultiplexed Address and Data Buses
- Enhanced 16-Bit Timer with Four-Stage Programmable Prescaler
  - — Three Input Capture (IC) Channels
  - — Four Output Compare (OC) Channels
  - — One Additional Channel, Selectable as Fourth IC or Fifth OC
- 8-Bit Pulse Accumulator
- Four 8-Bit or Two 16-Bit Pulse-Width Modulation (PWM) Timer Channels
- Two 12-Bit PWM Timer Channels
- Real-Time Interrupt Circuit
- Computer Operating Properly (COP) Watchdog
- Enhanced Asynchronous Nonreturn to Zero (NRZ) Serial Communications Interface (SCI)
- Enhanced Synchronous Serial Peripheral Interface (SPI)
- Twelve-Channel 8-Bit Analog-to-Digital (A/D) Converter
- Two-Channel 8-Bit Digital-to-Analog (A/D) Converter
- Eight Input/Output (I/O) Ports (62 Pins)
  - — 48 Bidirectional
  - — 14 Input Only
- Available in 84-Pin Plastic Leaded Chip Carrier (PLCC) or 80-Pin Quad Flat Pack (QFP)

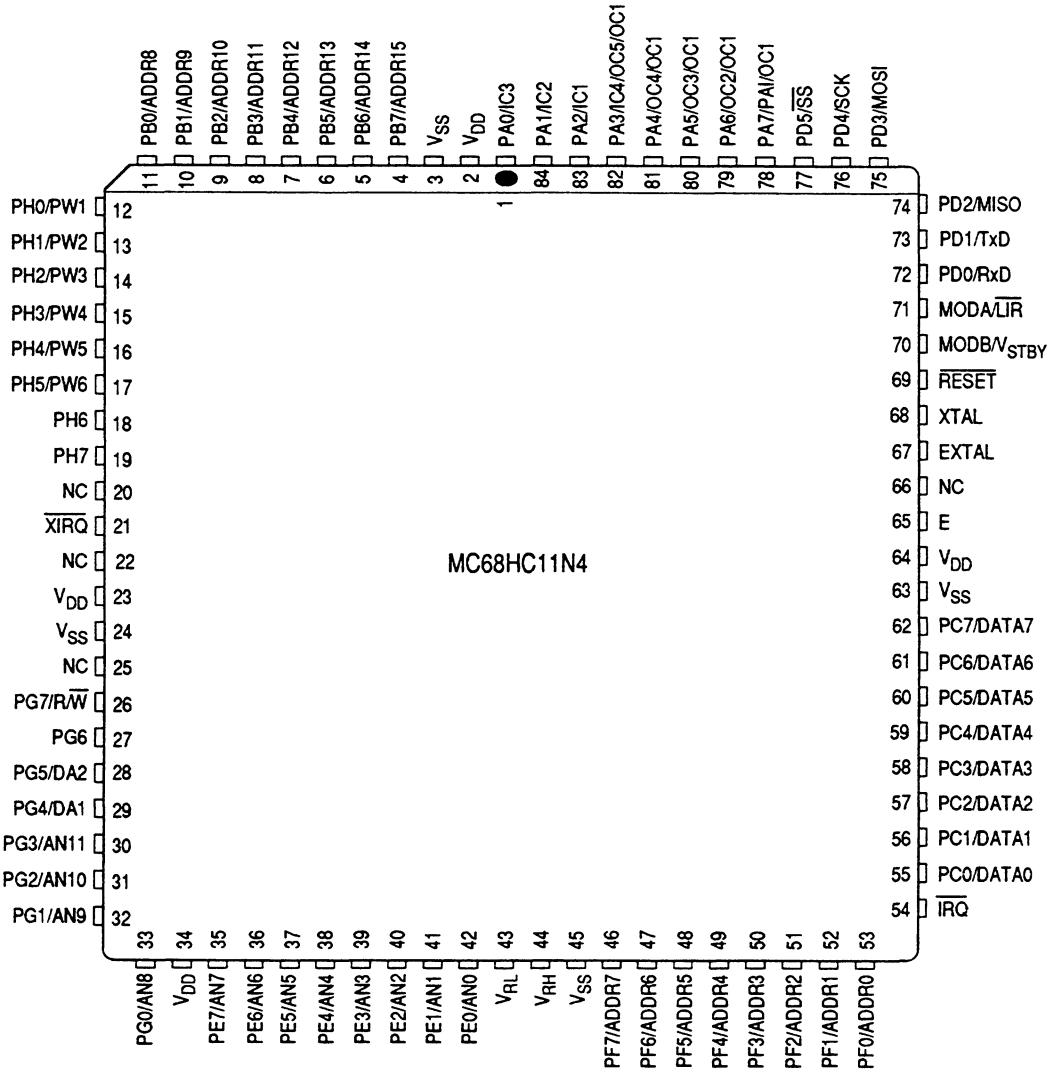
Ordering Information

Package	Temperature	Frequency	MC Order Number
84-Pin PLCC	– 40° to + 85° C	2 MHz	MC68HC11N4CFN2
		3 MHz	MC68HC11N4CFN3
		4 MHz	MC68HC11N4CFN4
80-Pin QFP	– 40° to + 85° C	2 MHz	MC68HC11N4CFU2
		3 MHz	MC68HC11N4CFU3
		4 MHz	MC68HC11N4CFU4

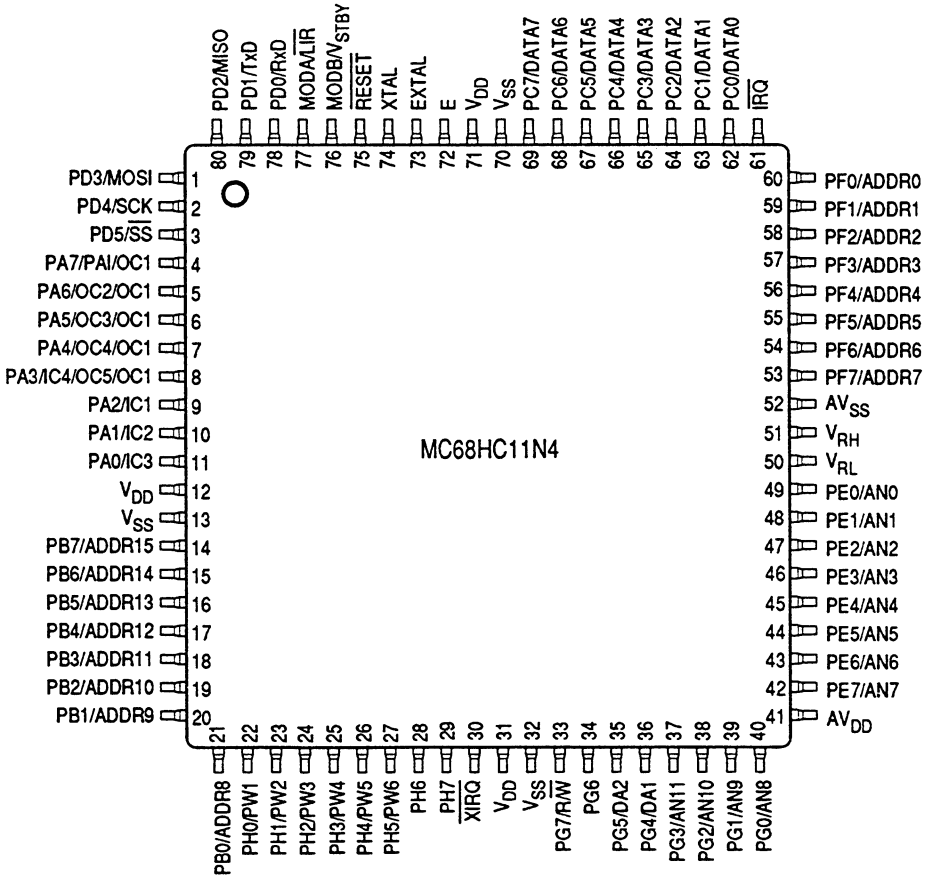
Table of Contents



MC68HC11N4 Block Diagram



Pin Assignments for 84-Pin PLCC/Cerquad



Pin Assignments for 80-Pin QFP

MC68HC11N4 Register and Control Bit Assignments (1 of 4)

	Bit 7	6	5	4	3	2	1	Bit 0	
$0000	PA7	PA6	PA5	PA4	PA3	PA2	PA1	PA0	PORTA
$0001	DDA7	DDA6	DDA5	DDA4	DDA3	DDA2	DDA1	DDA0	DDRA
$0002	DDB7	DDB6	DDB5	DDB4	DDB3	DDB2	DDB1	DDB0	DDRB
$0003	DDF7	DDF6	DDF5	DDF4	DDF3	DDF2	DDF1	DDF0	DDRF
$0004	PB7	PB6	PB5	PB4	PB3	PB2	PB1	PB0	PORTB
$0005	PF7	PF6	PF5	PF4	PF3	PF2	PF1	PF0	PORTF
$0006	PC7	PC6	PC5	PC4	PC3	PC2	PC1	PC0	PORTC
$0007	DDC7	DDC6	DDC5	DDC4	DDC3	DDC2	DDC1	DDC0	DDRC
$0008	0	0	PD5	PD4	PD3	PD2	PD1	PD0	PORTD
$0009	0	0	DDD5	DDD4	DDD3	DDD2	DDD1	DDD0	DDRD
$000A	PE7	PE6	PE5	PE4	PE3	PE2	PE1	PE0	PORTE
$000B	FOC1	FOC2	FOC3	FOC4	FOC5	0	0	0	CFORC
$000C	OC1M7	OC1M6	OC1M5	OC1M4	OC1M3	0	0	0	OC1M
$000D	OC1D7	OC1D6	OC1D5	OC1D4	OC1D3	0	0	0	OC1D
$000E	Bit 15	14	13	12	11	10	9	Bit 8	TCNT (High)
$000F	Bit 7	6	5	4	3	2	1	Bit 0	TCNT (Low)
$0010	Bit 15	14	13	12	11	10	9	Bit 8	TIC1 (High)
$0011	Bit 7	6	5	4	3	2	1	Bit 0	TIC1 (Low)
$0012	Bit 15	14	13	12	11	10	9	Bit 8	TIC2 (High)
$0013	Bit 7	6	5	4	3	2	1	Bit 0	TIC2 (Low)
$0014	Bit 15	14	13	12	11	10	9	Bit 8	TIC3 (High)
$0015	Bit 7	6	5	4	3	2	1	Bit 0	TIC3 (Low)
$0016	Bit 15	14	13	12	11	10	9	Bit 8	TOC1(High)
$0017	Bit 7	6	5	4	3	2	1	Bit 0	TOC1 (Low)
$0018	Bit 15	14	13	12	11	10	9	Bit 8	TOC2 (High)
$0019	Bit 7	6	5	4	3	2	1	Bit 0	TOC2 (Low)
$001A	Bit 15	14	13	12	11	10	9	Bit 8	TOC3 (High)
$001B	Bit 7	6	5	4	3	2	1	Bit 0	TOC3 (Low)
$001C	Bit 15	14	13	12	11	10	9	Bit 8	TOC4 (High)
$001D	Bit 7	6	5	4	3	2	1	Bit 0	TOC4 (Low)
$001E	Bit 15	14	13	12	11	10	9	Bit 8	TI4/O5 (High)
$001F	Bit 7	6	5	4	3	2	1	Bit 0	TI4/O5 (Low)

MC68HC11N4 Register and Control Bit Assignments (2 of 4)

	Bit 7	6	5	4	3	2	1	Bit 0	
$0020	OM2	OL2	OM3	OL3	OM4	OL4	OM5	OL5	TCTL1
$0021	EDG4B	EDG4A	EDG1B	EDG1A	EDG2B	EDG2A	EDG3B	EDG3A	TCTL2
$0022	OC1I	OC2I	OC3I	OC4I	I4O5I	IC1I	IC2I	IC3I	TMSK1
$0023	OC1F	OC2F	OC3F	OC4F	I4O5F	IC1F	IC2F	IC3F	TFLG1
$0024	TOI	RTII	PAOVI	PAII	0	0	PR1	PR0	TMSK2
$0025	TOF	RTIF	PAOVF	PAIF	0	0	0	0	TFLG2
$0026	0	PAEN	PAMOD	PEDGE	0	I4/O5	RTR1	RTR0	PACTL
$0027	Bit 7	6	5	4	3	2	1	Bit 0	PACNT
$0028	SPIE	SPE	DWOM	MSTR	CPOL	CPHA	SPR1	SPR0	SPCR
$0029	SPIF	WCOL	0	MODF	0	0	0	Bit 0	SPSR
$002A	Bit 7	6	5	4	3	2	1	Bit 0	SPDR
$002B	0	0	0	0	0	0	0	0	Reserved
$002C	0	0	0	0	HPPUE	GPPUE	FPPUE	BPPUE	PPAR
$002D	—	—	—	—	—	—	—	—	Reserved
$002E	—	—	—	—	—	—	—	—	Reserved
$002F	—	—	—	—	—	—	—	—	Reserved
$0030	CCF	0	SCAN	MULT	CD	CC	CB	CA	ADCTL
$0031	Bit 7	6	5	4	3	2	1	Bit 0	ADR1
$0032	Bit 7	6	5	4	3	2	1	Bit 0	ADR2
$0033	Bit 7	6	5	4	3	2	1	Bit 0	ADR3
$0034	Bit 7	6	5	4	3	2	1	Bit 0	ADR4
$0035	BULKP	0	BPRT4	PTCON	BPRT3	BPRT2	BPRT1	BPRT0	BPROT
$0036	—	—	—	—	—	—	—	—	Reserved
$0037	EE3	EE2	EE1	EE0	0	0	0	0	INIT2
$0038	LIRDV	CWOM	STRCH	IRVNE	LSBF	SPR2	0	0	OPT2
$0039	ADPU	CSEL	IRQE	DLY	CME	FCME	CR1	CR0	OPTION
$003A	Bit 7	6	5	4	3	2	1	Bit 0	COPRST
$003B	ODD	EVEN	0	BYTE	ROW	ERASE	EELAT	EEPGM	PPROG
$003C	RBOOT	SMOD	MDA	PSEL4	PSEL3	PSEL2	PSEL1	PSEL0	HPRIO
$003D	RAM3	RAM2	RAM1	RAM0	REG3	REG2	REG1	REG0	INIT
$003E	TILOP	0	OCCR	CBYP	DISR	FCM	FCOP	0	TEST1
$003F	ROMAD	1	1	PAREN	NOSEC	NOCOP	ROMON	EEON	CONFIG

MC68HC11N4 Register and Control Bit Assignments (3 of 4)

	Bit 7	6	5	4	3	2	1	Bit 0	
$0040	Bit 31	30	29	28	27	26	25	Bit 24	CREG (High)
$0041	Bit 23	22	21	20	19	18	17	Bit 16	CREG (Mid-High)
$0042	Bit 15	14	13	12	11	10	9	Bit 8	CREG (Mid-Low)
$0043	Bit 7	6	5	4	3	2	1	Bit 0	CREG (Low)
$0044	SIG	DIV	MAC	DCC	TRG	0	0	0	ALUC
$0045	Bit 15	14	13	12	11	10	9	Bit 8	AREG (High)
$0046	Bit 7	6	5	4	3	2	1	Bit 0	AREG (Low)
$0047	Bit 15	14	13	12	11	10	9	Bit 8	BREG (High)
$0048	Bit 7	6	5	4	3	2	1	Bit 0	BREG (Low)
$0049	NEG	RZF	0	0	0	OVF	DZF	ACF	ALUF
$004A	—	—	—	—	—	—	—	—	Reserved
$004B	—	—	—	—	—	—	—	—	Reserved
$004C	—	—	—	—	—	—	—	—	Reserved
$004D	0	0	0	0	0	0	DAE2	DAE1	DACON
$004E	Bit 7	6	5	4	3	2	1	Bit 0	DA1
$004F	Bit 7	6	5	4	3	2	1	Bit 0	DA2
$0050	0	0	PCKC2	PCKC1	0	0	PPOL6	PPOL5	PWCTL
$0051	0	0	0	0	PW6S2	PW6S1	PW5S2	PW5S1	PWSIZ
$0052	0	0	0	Bit 12	11	10	9	Bit 8	PWTDY5 (High)
$0053	Bit 7	6	5	4	3	2	1	Bit 0	PWTDY5 (Low)
$0054	0	0	0	Bit 12	11	10	9	Bit 8	PWTDY6 (High)
$0055	Bit 7	6	5	4	3	2	1	Bit 0	PWTDY6 (Low)
$0056	0	0	0	0	Bit 11	10	9	Bit 8	PWCNT5 (High)
$0057	Bit 7	6	5	4	3	2	1	Bit 0	PWCNT5 (Low)
$0058	—	—	—	—	—	—	—	—	Reserved
$0059	—	—	—	—	—	—	—	—	Reserved
$005A	—	—	—	—	—	—	—	—	Reserved
$005B	—	—	—	—	—	—	—	—	Reserved
$005C	—	—	—	—	—	—	—	—	Reserved
$005D	—	—	—	—	—	—	—	—	Reserved
$005E	—	—	—	—	—	—	—	—	Reserved
$005F	—	—	—	—	—	—	—	—	Reserved

MC68HC11N4 Register and Control Bit Assignments (4 of 4)

	Bit 7	6	5	4	3	2	1	Bit 0	
$0060	CON34	CON12	PCKA2	PCKA1	0	PCKB3	PCKB2	PCKB1	PWCLK
$0061	PCLK4	PCLK3	PCLK2	PCLK1	PPOL4	PPOL3	PPOL2	PPOL1	PWPOL
$0062	Bit 7	6	5	4	3	2	1	Bit 0	PWSCAL
$0063	TPWSL	DISCP	0	0	PWEN4	PWEN3	PWEN2	PWEN1	PWEN
$0064	Bit 7	6	5	4	3	2	1	Bit 0	PWCNT1
$0065	Bit 7	6	5	4	3	2	1	Bit 0	PWCNT2
$0066	Bit 7	6	5	4	3	2	1	Bit 0	PWCNT3
$0067	Bit 7	6	5	4	3	2	1	Bit 0	PWCNT4
$0068	Bit 7	6	5	4	3	2	1	Bit 0	PWPER1
$0069	Bit 7	6	5	4	3	2	1	Bit 0	PWPER2
$006A	Bit 7	6	5	4	3	2	1	Bit 0	PWPER3
$006B	Bit 7	6	5	4	3	2	1	Bit 0	PWPER4
$006C	Bit 7	6	5	4	3	2	1	Bit 0	PWDTY1
$006D	Bit 7	6	5	4	3	2	1	Bit 0	PWDTY2
$006E	Bit 7	6	5	4	3	2	1	Bit 0	PWDTY3
$006F	Bit 7	6	5	4	3	2	1	Bit 0	PWDTY4
$0070	BTST	BSPL	0	SBR12	SBR11	SBR10	SBR9	SBR8	SCBDH
$0071	SBR7	SBR6	SBR5	SBR4	SBR3	SBR2	SBR1	SBR0	SCBDL
$0072	LOOPS	WOMS	0	M	WAKE	ILT	PE	PT	SCCR1
$0073	TIE	TCIE	RIE	ILIE	TE	RE	RWU	SBK	SCCR2
$0074	TDRE	TC	RDRF	IDLE	OR	NF	FE	PF	SCSR1
$0075	0	0	0	0	0	0	0	RAF	SCSR2
$0076	R8	T8	0	0	0	0	0	0	SCDRH
$0077	R7/T7	R6/T6	R5/T5	R4/T4	R3/T3	R2/T2	R1/T1	R0/T0	SCDRL
$0078	—	—	—	—	—	—	—	—	Reserved
$0079	—	—	—	—	—	—	—	—	Reserved
$007A	—	—	—	—	—	—	—	—	Reserved
$007B	—	—	—	—	—	—	—	—	Reserved
$007C	PH7	PH6	PH5	PH4	PH3	PH2	PH1	PH0	PORTH
$007D	DDH7	DDH6	DDH5	DDH4	DDH3	DDH2	DDH1	DDH0	DDRH
$007E	PG7	PG6	PG5	PG4	PG3	PG2	PG1	PG0	PORTG
$007F	DDG7	DDG6	0	0	0	0	0	0	DDRG

Operating Modes

In single-chip operating mode, the MC68HC11N4 is a stand-alone microcontroller with no external address or data bus.

In expanded nonmultiplexed operating mode, the MCU can access a 64 Kbyte physical address space. This space includes the same on-chip memory addresses used for single-chip mode, in addition to addressing capabilities for external peripheral and memory devices. The expansion bus is made up of ports B, C, and F, and the R/$\overline{W}$ signal. In expanded operating mode, high order address bits are output on the port B pins, low order address bits on the port F pins, and the data bus on port C. The R/$\overline{W}$ pin controls the direction of data transfer on the port C bus.

Special bootstrap mode allows special-purpose programs to be entered into internal RAM. The bootloader program uses the serial communications interface (SCI) to read a program of up to 768 bytes into on-chip RAM. After a four-character delay, or after receiving the character for address $037F, control passes to the loaded program at $0080.

Special test mode is used primarily for factory testing.

On-Chip Memory

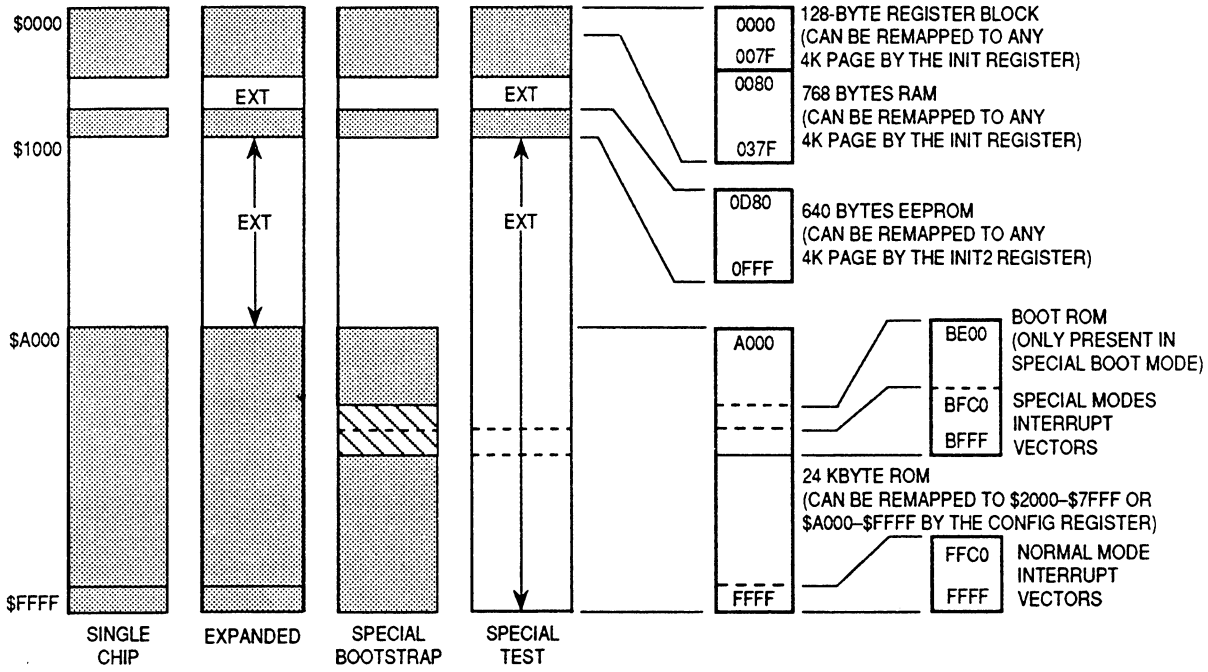
The INIT, INIT2, and CONFIG registers control the existence and locations of the registers, RAM, EEPROM, and ROM in the physical 64 Kbyte memory space.

The 128-byte register block originates at $0000 after reset and can be placed at any other 4K boundary ($x000) after reset by writing an appropriate value to the INIT register.

The 768-byte RAM is mappable to any 4K boundary in memory. The RAM is divided into two sections of 128 bytes and 640 bytes. For the MC68HC11N4, 128 bytes of the RAM are mapped at $0000–$007F unless the registers are mapped to this space. If the registers are located in that space, the same 128 bytes of RAM are located at $0300 to $037F. Remapping is accomplished by writing appropriate values into the two nibbles of the INIT register. Refer to the register and RAM mapping examples following the MC68HC11N4 memory map.

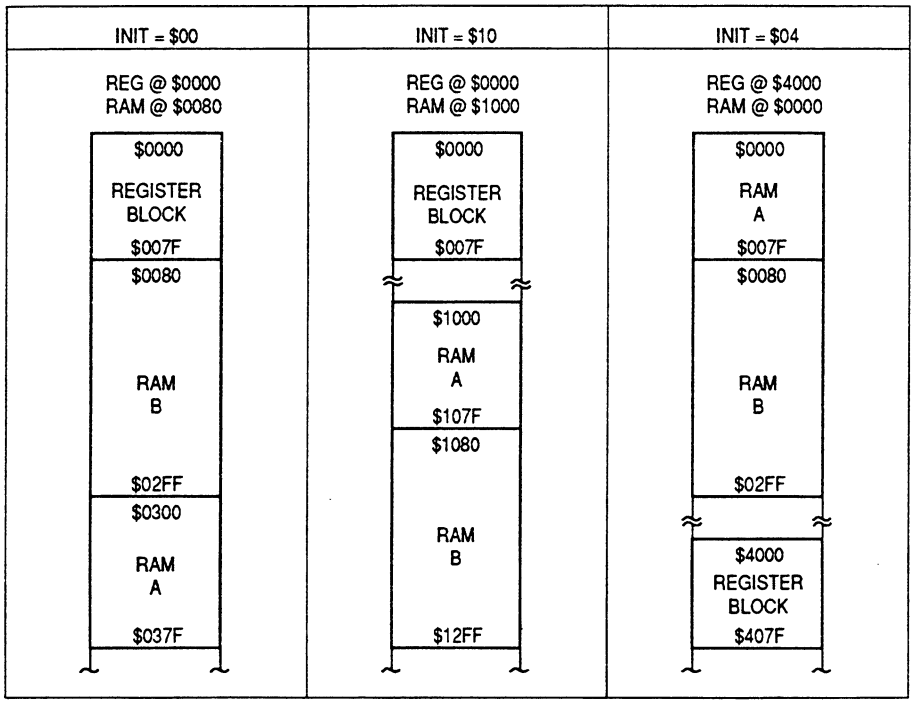
The 640-byte EEPROM is initially located at $0D80 after reset, assuming EEPROM is enabled in the memory map by the CONFIG register. EEPROM can be placed at any other 4K boundary ($xD80) by writing appropriate values to the INIT2 register.

The ROMAD and ROMON control bits in the CONFIG register control the position and presence of ROM in the memory map. In special test mode, the ROMON bit is forced to zero so that the ROM is initially removed from the memory map. In single-chip mode, the ROMAD and ROMON bits are forced to one, causing the ROM to be enabled at $A000–$FFFF. This ensures that, in single-chip mode, there will be ROM at the vector space.



NOTE: ROM CAN BE ENABLED IN SPECIAL TEST MODE BY SETTING ROMON BIT IN THE CONFIG REGISTER AFTER RESET.

Memory Map



RAM and Register Mapping

HPRIO — Highest Priority I-Bit Interrupt and Miscellaneous **$003C**

	Bit 7	6	5	4	3	2	1	Bit 0	
	RBOOT*	SMOD*	MDA*	PSEL4	PSEL3	PSEL2	PSEL1	PSEL0	
RESET:	0	0	0	0	0	1	1	0	Single Chip
	0	0	1	0	0	1	1	0	Expanded
	1	1	0	0	0	1	1	0	Bootstrap
	0	1	1	0	0	1	1	0	Special Test

*The reset values of RBOOT, SMOD, and MDA depend on the mode selected at power-up.

RBOOT — Read Bootstrap ROM

Valid only when SMOD is set to one (special bootstrap or special test mode). Can only be written in special mode.

0 = Bootloader ROM disabled and not in map

1 = Bootloader ROM enabled and in map at $BE00–$BFFF

SMOD and MDA — Special Mode Select and Mode Select A

These two bits can be read at any time, but can only be written in special mode.

Inputs		Mode	Latched at Reset	
MODB	MODA		SMOD	MDA
1	0	Single Chip	0	0
1	1	Expanded	0	1
0	0	Special Bootstrap	1	0
0	1	Special Test	1	1

PSEL[4:0] — Priority Select Bits 4 to 0

Refer to **Resets and Interrupts**.

INIT — RAM and I/O Register Mapping **$003D**

	Bit 7	6	5	4	3	2	1	Bit 0
	RAM3	RAM2	RAM1	RAM0	REG3	REG2	REG1	REG0
RESET:	0	0	0	0	0	0	0	0

Can be written only once in first 64 cycles out of reset in normal modes or at any time in special modes.

RAM[3:0] — Internal RAM Map Position

REG[3:0] — 128-Byte Register Block Map Position

CONFIG — COP, ROM Mapping, EEPROM Enables **$003F**

	Bit 7	6	5	4	3	2	1	Bit 0
	ROMAD	1	1	PAREN	NOSEC	NOCOP	ROMON	EEON
RESET:	—	1	1	—	—	—	—	—

CONFIG is made up of EEPROM cells and static working latches. The operation of the MCU is controlled directly by these latches and not the actual EEPROM byte. When the CONFIG register is programmed, the EEPROM byte is accessed. When the CONFIG register is read, the static latches are accessed.

These bits can be read at any time. The values read are those latched into the register from the EEPROM cells during the last reset sequence. A new value programmed into this register is not readable until after a subsequent reset sequence. Unused bits always read as ones.

If SMOD = 1, CONFIG bits can be written at any time. If SMOD = 0, CONFIG bits can only be written using the EEPROM programming sequence, and are neither readable nor active until latched via the next reset.

ROMAD — ROM Mapping Control

In single-chip mode ROMAD is forced to one out of reset.

0 = ROM located at $2000–$7FFF
1 = ROM located at $A000–$FFFF

Bits[6:5] — Not Implemented

Always read one

PAREN — Pull-up Assignment Register Enable

Refer to **Parallel Input/Output**.

NOSEC — Security Disable

NOSEC is invalid unless the security mask option is specified before the MCU is manufactured. If security mask option is omitted NOSEC always reads one.

0 = Security enabled
1 = Security disabled

NOCOP — COP System Disable

Resets to programmed value

0 = COP enabled (forces reset on timeout)
1 = COP disabled (does not force reset on timeout)

ROMON — ROM Enable

In single-chip mode, ROMON is forced to one out of reset. In special test mode, ROMON is forced to zero out of reset.

0 = ROM removed from memory map
1 = ROM present in memory map

EEON — EEPROM Enable

0 = EEPROM disabled from memory map
1 = EEPROM present in memory map with location depending on value specified in EE[3:0] in INIT2

OPT2 — System Configuration Options 2 **$0038**

	Bit 7	6	5	4	3	2	1	Bit 0
	LIRDV	CWOM	STRCH	IRVNE	LSBF	SPR2	0	0
RESET:	0	0	0	—	0	0	0	0

LIRDV— LIR Driven

In single-chip and bootstrap modes, this bit has no meaning or effect. The LIR pin is normally configured for wired-OR operation (only pulls low). In order to detect consecutive instructions in a high-speed application, this signal can be made to drive high for a short time to prevent false triggering.

0 = LIR not driven high out of reset
1 = LIR driven high for one quarter cycle to reduce transition time

CWOM — Port C Wired-OR Mode

Refer to **Parallel Input/Output**.

STRCH — Stretch

0 = Normal operation
1 = Off-chip accesses to $0000–$7FFF extended by one E-clock cycle

IRVNE — Internal Read Visibility/Not E

Can be written only once in any mode. In expanded mode, IRVNE determines whether IRV is on or off. In special test mode, IRVNE resets to 1. In all other modes, IRVNE is reset to 0.

0 = No internal read visibility on external bus
1 = Data from internal reads is driven out of the external data bus.

In single-chip modes, this bit determines whether the E clock is driven out from the chip.

0 = E is driven out from the chip
1 = E pin is driven low

E is always active out of reset and in expanded modes.

Mode	IRVNE Out of Reset	E Clock Out of Reset	IRV Out of Reset	IRVNE Affects Only	IRVNE Can Be Written
Single Chip	0	On	Off	E	Once
Expanded	0	On	Off	IRV	Once
Boot	0	On	Off	E	Once
Special Test	1	On	On	IRV	Once

LSBF — SPI LSB First Enable

Refer to **Serial Peripheral Interface**.

SPR2 — SPI Clock Rate Select

Refer to **Serial Peripheral Interface**.

Bits [1:0] — Not implemented

Always read zero

Electrically Erasable Programmable Read-Only Memory (EEPROM)

The 640-byte on-chip EEPROM is initially located from $0D80 to $0FFF after reset in all modes. It can be mapped to any other 4K boundary by writing to the INIT2 register. The EEPROM is enabled by the EEON bit in the CONFIG register. Programming and erasing is controlled by the PPROG register.

An internal oscillator clock-run charge pump supplies the programming voltage. Use of the block protect register (BPROT) prevents inadvertent writes to (or erases of) blocks of EEPROM. The CSEL bit in the OPTION register selects the on-chip oscillator clock for programming and erasing while operating at frequencies below 2 MHz.

In special mode there is an extra row of 16 bytes of EEPROM (located at $0D60), which is used for factory testing. Endurance and data retention specifications do not apply to this row.

The erased state of EEPROM is $FF (all ones).

To erase the EEPROM, ensure that the proper bits of the BPROT register are cleared, then complete the following steps using the PPROG register:

1. Write to PPROG with the ERASE, EELAT, and appropriate BYTE and ROW bits set.
2. Write to the appropriate EEPROM address with any data. Row erase only requires a write to any location in the row. Bulk erase is accomplished by writing to any location in the array.
3. Write to PPROG with ERASE, EELAT, EEPGM, and the appropriate BYTE and ROW bits set.
4. Delay for 10 ms or more, as appropriate.
5. Clear the EEPGM bit in PPROG to turn off the high voltage.
6. Clear the PPROG register to reconfigure the EEPROM address and data buses for normal operation.

To program the EEPROM, ensure the proper bits of the BPROT register are cleared, then complete the following steps using the PPROG register:

1. Write to PPROG with the EELAT bit set.
2. Write data to the desired address.
3. Write to PPROG with the EELAT and EEPGM bits set.
4. Delay for 10 ms or more, as appropriate.
5. Clear the EEPGM bit in PPROG to turn off the high voltage.
6. Clear the PPROG register to reconfigure the EEPROM address and data buses for normal operation.

BPROT — Block Protect **$0035**

	Bit 7	6	5	4	3	2	1	Bit 0
	BULKP	0	BPRT4	PTCON	BPRT3	BPRT2	BPRT1	BPRT0
RESET:	1	0	1	1	1	1	1	1

NOTE

Block protect register bits can be written to zero (protection disabled) only once within 64 cycles of a reset in normal modes, or at any time in special modes. Block protect register bits can be written to one (protection enabled) at any time.

BULKP — Bulk Erase of EEPROM Protect
0 = EEPROM can be bulk erased normally
1 = EEPROM cannot be bulk or row erased

Bit 6 — Not Implemented
Always reads zero

BPRT[4:0] — Block Protect Bits for EEPROM
0 = Protection disabled
1 = Protection enabled

Bit Name	Block Protected	Block Size
BPRT4	$xF80–$xFFF	128 Bytes
BPRT3	$xE60–$xF7F	288 Bytes
BPRT2	$xDE0–$xE5F	128 Bytes
BPRT1	$xDA0–$xDDF	64 Bytes
BPRT0	$xD80–$xD9F	32 Bytes

PTCON — Protect for CONFIG
0 = CONFIG register can be programmed or erased normally
1 = CONFIG register cannot be programmed or erased

INIT2 — EEPROM Mapping **$0037**

	Bit 7	6	5	4	3	2	1	Bit 0
	EE3	EE2	EE1	EE0	0	0	0	0
RESET:	0	0	0	0	0	0	0	0

INIT2 can be written only once in normal modes, any time in special modes.

EE[3:0] — EEPROM Map Position
EEPROM is at $xD80–$xFFF, where x is the hexadecimal digit represented by EE[3:0] bits.

Bits [3:0] — Not implemented
Always read zero

PPROG — EEPROM Programming Control **$003B**

	Bit 7	6	5	4	3	2	1	Bit 0
	ODD	EVEN	0	BYTE	ROW	ERASE	EELAT	EEPGM
RESET:	0	0	0	0	0	0	0	0

ODD — Program Odd Rows in Half of EEPROM (TEST)

EVEN — Program Even Rows in Half of EEPROM (TEST)

Bit 5 — Not Implemented
Always reads zero

BYTE — Byte/Other EEPROM Erase Mode
0 = Row or bulk erase mode used
1 = Erase only one byte of EEPROM

ROW — Row/All EEPROM Erase Mode (only valid when BYTE = 0)
0 = All 640 bytes of EEPROM erased
1 = Erase only one 16-byte row of EEPROM

BYTE	ROW	Action
0	0	Bulk Erase (All 640 Bytes)
0	1	Row Erase (16 Bytes)
1	0	Byte Erase
1	1	Byte Erase

ERASE — Erase/Normal Control for EEPROM
0 = Normal read or program mode
1 = Erase mode

EELAT — EEPROM Latch Control
0 = EEPROM address and data bus configured for normal reads
1 = EEPROM address and data bus configured for programming or erasing

EEPGM — EEPROM Program Command
0 = Program or erase voltage switched off to EEPROM array
1 = Program or erase voltage switched on to EEPROM array

Refer also to INIT2 register.

OPTION — System Configuration Options **$0039**

	Bit 7	6	5	4	3	2	1	Bit 0
	ADPU	CSEL	IRQE*	DLY*	CME	FCME*	CR1*	CR0*
RESET:	0	0	0	1	0	0	0	0

*Can be written only once in first 64 cycles out of reset in normal modes, any time in special mode.

ADPU — A/D Converter Power-Up
Refer to **Analog-to-Digital Converter**.

CSEL — Clock Select for A/D and EEPROM
0 = Use system E clock for EEPROM programming and A/D
1 = Use internal RC clock source for EEPROM programming and A/D

IRQE— IRQ Select Edge-Sensitive Only
Refer to **Resets and Interrupts**.

DLY — Enable Oscillator Startup Delay on Exit from Stop
Refer to **Resets and Interrupts**.

CME — Clock Monitor Enable
Refer to **Resets and Interrupts**.

FCME — Force Clock Monitor Enable
Refer to **Resets and Interrupts**.

CR1, CR0 — COP Timer Rate Select
Refer to **Main Timer**.

Resets and Interrupts

The MC68HC11N4 has 3 reset vectors and 18 interrupt vectors. The reset vectors are as follows:

- $\overline{\text{RESET}}$, or Power-On Reset
- Clock Monitor Fail
- COP Failure

The 18 interrupt vectors service 22 interrupt sources (3 nonmaskable, 19 maskable). The 3 non-maskable interrupt vectors are as follows:

- Illegal Opcode Trap
- Software Interrupt
- $\overline{\text{XIRQ}}$ Pin (X-Bit Interrupt)

On-chip peripheral systems generate maskable interrupts, which are recognized only if the global interrupt mask bit (I) in the condition code register (CCR) is clear. Maskable interrupts are prioritized according to a default arrangement; however, any one source can be elevated to the highest maskable priority position by a software-accessible control register (HPRIO). The HPRIO register can be written at any time, provided bit I in the CCR is set.

Nineteen interrupt sources in the MC68HC11N4 are subject to masking by the global interrupt mask bit (bit I in the CCR). In addition to the global bit I, all of these sources, except the external interrupt ($\overline{\text{IRQ}}$) pin, are controlled by local enable bits in control registers. Most interrupt sources in the M68HC11 have separate interrupt vectors; therefore, there is usually no need for software to poll control registers to determine the cause of an interrupt.

For some interrupt sources, such as the SCI interrupts, the flags are automatically cleared during the normal course of responding to the interrupt requests. For example, the RDRF flag in the SCI system is cleared by the automatic clearing mechanism consisting of a read of the SCI status register while RDRF is set, followed by a read of the SCI data register. The normal response to an RDRF interrupt request would be to read the SCI status register to check for receive errors, then to read the received data from the SCI data register. These two steps satisfy the automatic clearing mechanism without requiring any special instructions.

Refer to the following table for a list of interrupt and reset vector assignments.

Vector Address	Interrupt Source	CC Register Mask Bit	Local Mask
FFC0, C1 — FFD4, D5	Reserved	—	—
FFD6, D7	SCI Serial System	I	
	• SCI Receive Data Register Full		RIE
	• SCI Receiver Overrun		RIE
	• SCI Transmit Data Register Empty		TIE
	• SCI Transmit Complete		TCIE
	• SCI Idle Line Detect		ILIE
FFD8, D9	SPI Serial Transfer Complete	I	SPIE
FFDA, DB	Pulse Accumulator Input Edge	I	PAII
FFDC, DD	Pulse Accumulator Overflow	I	PAOVI
FFDE, DF	Timer Overflow	I	TOI
FFE0, E1	Timer Input Capture 4/Output Compare 5	I	I4/O5I
FFE2, E3	Timer Output Compare 4	I	OC4I
FFE4, E5	Timer Output Compare 3	I	OC3I
FFE6, E7	Timer Output Compare 2	I	OC2I
FFE8, E9	Timer Output Compare 1	I	OC1I
FFEA, EB	Timer Input Capture 3	I	IC3I
FFEC, ED	Timer Input Capture 2	I	IC2I
FFEE, EF	Timer Input Capture 1	I	IC1I
FFF0, F1	Real-Time Interrupt	I	RTII
FFF2, F3	$\overline{\text{IRQ}}$ (External Pin)	I	None
FFF4, F5	$\overline{\text{XIRQ}}$ Pin	X	None
FFF6, F7	Software Interrupt	None	None
FFF8, F9	Illegal Opcode Trap	None	None
FFFA, FB	COP Failure	None	NOCOP
FFFC, FD	Clock Monitor Fail	None	CME
FFFE, FF	$\overline{\text{RESET}}$	None	None

OPTION — System Configuration Options **$0039**

	Bit 7	6	5	4	3	2	1	Bit 0
	ADPU	CSEL	IRQE*	DLY*	CME	FCME*	CR1*	CR0*
RESET:	0	0	0	1	0	0	0	0

*Can be written only once in first 64 cycles out of reset in normal mode, or at any time in special mode.

ADPU — A/D Converter Power up
Refer to **Analog-to-Digital Converter**.

CSEL — Clock Select for A/D and EEPROM
Refer to **Electrically Erasable Programmable Read-Only Memory (EEPROM)**.

IRQE — $\overline{\text{IRQ}}$ Select Edge-Sensitive Only
0 = Low level recognition
1 = Falling edge recognition

DLY — Enable Oscillator Start-Up Delay on Exit from STOP
0 = No stabilization delay on exit from STOP
1 = Stabilization delay enabled on exit from STOP

CME — Clock Monitor Enable
0 = Clock monitor disabled; slow clocks can be used
1 = Slow or stopped clocks cause clock failure reset

FCME — Force Clock Monitor Enable
0 = Clock monitor follows the state of the CME bit
1 = Clock monitor circuit is enabled until next reset

CR[1:0] — COP Timer Rate Select

COP Timer Rate Select

CR[1:0]	Divide E/2^{15} By	XTAL = 8.0 MHz Timeout – 0/+16.4 ms	XTAL = 12.0 MHz Timeout – 0/+10.9 ms	XTAL = 16.0 MHz Timeout – 0/+8.2 ms
0 0	1	16.384 ms	10.923 ms	8.192 ms
0 1	4	65.536 ms	43.691 ms	32.768 ms
1 0	16	262.14 ms	174.76 ms	131.07 ms
1 1	64	1.049 sec	699.05 ms	524.29 ms
	E =	2.0 MHz	3.0 MHz	4.0 MHz

COPRST — Arm/Reset COP Timer Circuitry **$003A**

	Bit 7	6	5	4	3	2	1	Bit 0
	7	6	5	4	3	2	1	0
RESET:	0	0	0	0	0	0	0	0

Write $55 to COPRST to arm COP watchdog clearing mechanism.
Write $AA to COPRST to reset COP watchdog.

HPRIO — Highest Priority I-Bit Interrupt and Miscellaneous **$003C**

	Bit 7	6	5	4	3	2	1	Bit 0
	RBOOT*	SMOD*	MDA*	PSEL4	PSEL3	PSEL2	PSEL1	PSEL0
RESET:	—	—	—	0	0	1	1	0

*RBOOT, SMOD, and MDA reset depend on power-up initialization mode and can only be written in special mode.

RBOOT — Read Bootstrap ROM

Refer to **Operating Modes and On-Chip Memory**.

SMOD — Special Mode Select

Refer to **Operating Modes and On-Chip Memory**.

MDA — Mode Select A

Refer to **Operating Modes and On-Chip Memory**.

PSEL[4:0] — Priority Select Bit 4 through Bit 0

Can be written only while bit I in the CCR is set (interrupts disabled). These bits select one interrupt source to be elevated above all other I-bit related sources.

PSELx					
4	**3**	**2**	**1**	**0**	**Interrupt Source Promoted**
0	0	0	X	X	Reserved (Default to $\overline{IRQ}$)
0	0	1	0	0	Reserved (Default to $\overline{IRQ}$)
0	0	1	0	1	Reserved (Default to $\overline{IRQ}$)
0	0	1	1	0	$\overline{IRQ}$ (External Pin)
0	0	1	1	1	Real-Time Interrupt
0	1	0	0	0	Timer Input Capture 1
0	1	0	0	1	Timer Input Capture 2
0	1	0	1	0	Timer Input Capture 3
0	1	0	1	1	Timer Output Compare 1
0	1	1	0	0	Timer Output Compare 2
0	1	1	0	1	Timer Output Compare 3
0	1	1	1	0	Timer Output Compare 4
0	1	1	1	1	Timer Output Compare 5/Input Capture 4
1	0	0	0	0	Timer Overflow
1	0	0	0	1	Pulse Accumulator Overflow
1	0	0	1	0	Pulse Accumulator Input Edge
1	0	0	1	1	SPI Serial Transfer Complete
1	0	1	0	0	SCI Serial System
1	0	1	0	1	Reserved (Default to $\overline{IRQ}$)
1	0	1	1	0	Reserved (Default to $\overline{IRQ}$)
1	0	1	1	1	Reserved (Default to $\overline{IRQ}$)
1	1	X	X	X	Reserved (Default to $\overline{IRQ}$)

CONFIG — COP, ROM Mapping, EEPROM Enables **$003F**

	Bit 7	6	5	4	3	2	1	Bit 0
	ROMAD	1	1	PAREN	NOSEC	NOCOP	ROMON	EEON
RESET:	—	1	1	—	—	—	—	—

CONFIG is made up of EEPROM cells and static latches. The operation of the MCU is controlled directly by these latches and not the actual EEPROM byte. When programming the CONFIG register, the EEPROM byte is being accessed. When the CONFIG register is being read, the static latches are being accessed.

These bits can be read at any time. The value read is the one latched into the register from the EEPROM cells during the last reset sequence. A new value programmed into this register is not readable until after a subsequent reset sequence. Unused bits always read as ones.

If SMOD = 1, CONFIG bits can be written any time. If SMOD = 0, CONFIG bits can only be written using the EEPROM programming sequence, and are neither readable nor active until latched via the next reset.

ROMAD — ROM Mapping Control
Refer to **Operating Modes and On-Chip Memory**.

Bits [6:5] — Not implemented
Always read one

PAREN — Pull up Assignment Register Enable
Refer to **Parallel Input/Output**.

NOSEC — Security Disable
Refer to **Operating Modes and On-Chip Memory**.

NOCOP — COP System Disable
Resets to programmed value
0 = COP enabled (forces reset on timeout)
1 = COP disabled (does not force reset on timeout)

ROMON — ROM Enable
Refer to **Operating Modes and On-Chip Memory**.

EEON — EEPROM Enable
Refer to **Operating Modes and On-Chip Memory**.

Parallel Input/Output

The MC68HC11N4 has up to 62 input/output lines, depending on the operating mode. To enhance the I/O functions, the data bus of this microcontroller is nonmultiplexed. The following table is a summary of the configuration and features of each port.

Port	Input Pins	Output Pins	Bidirectional Pins	Shared Functions
Port A	—	—	8	Timer
Port B	—	—	8	High Order Address
Port C	—	—	8	Data Bus
Port D	—	—	6	SCI and SPI
Port E	8	—	—	A/D Converter
Port F	—	—	8	Low Order Address
Port G	—	—	8	Memory Expansion
Port H	—	—	8	PWM, Chip Select

CONFIG — COP, ROM Mapping, EEPROM Enables **$003F**

	Bit 7	6	5	4	3	2	1	Bit 0
	ROMAD	1	1	PAREN	NOSEC	NOCOP	ROMON	EEON
RESET:	—	1	1	—	—	—	—	—

ROMAD — ROM Mapping Control
Refer to **Operating Modes and On-Chip Memory**.

Bits [6:5] — Not implemented
Always read one

PAREN — Pull-up Assignment Register Enable
0 = Pull-ups always disabled regardless of state of bits in PPAR
1 = Pull-ups either enabled or disabled through PPAR

NOSEC — Security Disable
Refer to **Operating Modes and On-Chip Memory**.

NOCOP — COP System Disable
Refer to **Resets and Interrupts**.

ROMON — ROM Enable
Refer to **Operating Modes and On-Chip Memory**.

EEON — EEPROM Enable
Refer to **Operating Modes and On-Chip Memory**.

OPT2 — System Configuration Options 2 **$0038**

	Bit 7	6	5	4	3	2	1	Bit 0
	LIRDV	CWOM	STRCH	IRVNE	LSBF	SPR2	0	0
RESET:	0	0	0	—	0	0	0	0

LIRDV— LIR Driven
Refer to **Operating Modes and On-Chip Memory**.

CWOM — Port C Wired-OR Mode
0 = Port C operates normally.
1 = Port C outputs are open-drain.

STRCH — Stretch
0 = Normal operation
1 = Off-chip accesses to $0000–$7FFF extended by one E-clock cycle

IRVNE — Internal Read Visibility/Not E
Refer to **Operating Modes and On-Chip Memory**.

LSBF — SPI LSB First Enable
Refer to **Serial Peripheral Interface**.

SPR2 — SPI Clock (SCK) Rate Select
Refer to **Serial Peripheral Interface**.

Bits [1:0] — Not implemented
Always read zero

NOTE

Do not confuse pin function with the electrical state of the pin at reset. All general-purpose I/O pins configured as inputs at reset are in a high-impedance state and the contents of port data registers is undefined. In port descriptions, a "U" indicates this condition. The pin function is mode dependent.

PORTA — Port A Data **$0000**

	Bit 7	6	5	4	3	2	1	Bit 0
	PA7	PA6	PA5	PA4	PA3	PA2	PA1	PA0
RESET:	U	U	U	U	U	U	U	U
Alt. Pin Func.:	PAI	OC2	OC3	OC4	IC4/OC5	IC1	IC2	IC3
And/or:	OC1	OC1	OC1	OC1	OC1	—	—	—

NOTE

To enable PA3 as fourth input capture, set the I4/O5 bit in the PACTL register. Otherwise, PA3 is configured as a fifth output compare out of reset, with bit I4/O5 being cleared. If the DDA3 bit is set (configuring PA3 as an output), and IC4 is enabled, writes to PA3 cause edges on the pin to result in input captures. Writing to TI4/O5 has no effect when the TI4/O5 register is acting as IC4. PA7 drives the pulse accumulator input but also can be configured for general-purpose I/O, or output compare. Note that even when PA7 is configured as an output, the pin still drives the pulse accumulator input.

DDRA — Data Direction Register for Port A **$0001**

	Bit 7	6	5	4	3	2	1	Bit 0
	DDA7	DDA6	DDA5	DDA4	DDA3	DDA2	DDA1	DDA0
RESET:	0	0	0	0	0	0	0	0

DDA[7:0] — Data Direction for Port A
0 = Input
1 = Output

PORTB — Port B Data **$0004**

	Bit 7	6	5	4	3	2	1	Bit 0
	PB7	PB6	PB5	PB4	PB3	PB2	PB1	PB0
S. Chip or Boot:	PB7	PB6	PB5	PB4	PB3	PB2	PB1	PB0
RESET:	U	U	U	U	U	U	U	U
Expan. or Test:	ADDR15	ADDR14	ADDR13	ADDR12	ADDR11	ADDR10	ADDR9	ADDR8

Reset state is mode dependent. In single-chip or bootstrap modes, port B pins are high-impedance inputs with selectable internal pull-up resistors. In expanded or test modes, port B pins are high order address outputs and PORTB is not in the memory map.

DDRB — Data Direction Register for Port B **$0002**

	Bit 7	6	5	4	3	2	1	Bit 0
	DDB7	DDB6	DDB5	DDB4	DDB3	DDB2	DDB1	DDB0
RESET:	0	0	0	0	0	0	0	0

DDB[7:0] — Data Direction for Port B
0 = Input
1 = Output

DDRF — Data Direction Register for Port F **$0003**

	Bit 7	6	5	4	3	2	1	Bit 0
	DDF7	DDF6	DDF5	DDF4	DDF3	DDF2	DDF1	DDF0
RESET:	0	0	0	0	0	0	0	0

DDF[7:0] — Data Direction for Port F
0 = Input
1 = Output

PORTF — Port F Data **$0005**

	Bit 7	6	5	4	3	2	1	Bit 0
	PF7	PF6	PF5	PF4	PF3	PF2	PF1	PF0
S. Chip or Boot:	PF7	PF6	PF5	PF4	PF3	PF2	PF1	PF0
RESET:	U	U	U	U	U	U	U	U
Expan. or Test:	ADDR7	ADDR6	ADDR5	ADDR4	ADDR3	ADDR2	ADDR1	ADDR0

Reset state is mode dependent. In single-chip or bootstrap modes, port F is high-impedance input with selectable internal pull-up resistors. In expanded or test modes, port F pins are low order address outputs and PORTF is not in the memory map.

PORTC — Port C Data **$0006**

	Bit 7	6	5	4	3	2	1	Bit 0
	PC7	PC6	PC5	PC4	PC3	PC2	PC1	PC0
S. Chip or Boot:	PC7	PC6	PC5	PC4	PC3	PC2	PC1	PC0
RESET:	0	0	0	0	0	0	0	0
Expan. or Test:	DATA7	DATA6	DATA5	DATA4	DATA3	DATA2	DATA1	DATA0

Reset state is mode dependent. In single-chip or bootstrap modes, port C is high-impedance input with selectable internal pull-up resistors. In expanded or test modes, port C pins are data bus inputs and outputs and PORTC is not in the memory map.

DDRC — Data Direction Register for Port C **$0007**

	Bit 7	6	5	4	3	2	1	Bit 0
	DDC7	DDC6	DDC5	DDC4	DDC3	DDC2	DDC1	DDC0
RESET:	0	0	0	0	0	0	0	0

DDC[7:0] — Data Direction for Port C
0 = Input
1 = Output

PORTD — Port D Data **$0008**

	Bit 7	6	5	4	3	2	1	Bit 0
	0	0	PD5	PD4	PD3	PD2	PD1	PD0
RESET:	0	0	U	U	U	U	U	U
Alt. Pin Func.:	—	—	$\overline{SS}$	SCK	MOSI	MISO	TxD	RxD

DDRD — Data Direction Register for Port D **$0009**

	Bit 7	6	5	4	3	2	1	Bit 0
	0	0	DDD5	DDD4	DDD3	DDD2	DDD1	DDD0
RESET:	0	0	0	0	0	0	0	0

Bits [7:6] — Not implemented
Always read zero

DDD[5:0] — Data Direction for Port D
0 = Input
1 = Output

NOTE

When the SPI system is in slave mode, DDD5 has no meaning nor effect. When the SPI system is in master mode, DDD5 determines whether bit 5 of PORTD is an error detect input (DDD5 = 0) or a general-purpose output (DDD5 = 1). If the SPI system is enabled and expects any of bits [4:2] to be an input that bit will be an input regardless of the state of the associated DDR bit. If any of bits [4:2] are expected to be outputs that bit will be an output **only** if the associated DDR bit is set.

PORTE — Port E Data **$000A**

	Bit 7	6	5	4	3	2	1	Bit 0
	PE7	PE6	PE5	PE4	PE3	PE2	PE1	PE0
RESET:	U	U	U	U	U	U	U	U
Alt. Pin Func.:	AN7	AN6	AN5	AN4	AN3	AN2	AN1	AN0

PPAR — Port Pull-Up Assignment **$002C**

	Bit 7	6	5	4	3	2	1	Bit 0
	0	0	0	0	HPPUE	GPPUE	FPPUE	BPPUE
RESET:	0	0	0	0	1	1	1	1

Bits [7:4] — Not implemented
Always read zero

xPPUE — Port x Pin Pull-up Enable
Refer to PAREN bit in CONFIG register discussed in **Parallel Input/Output**.
0 = Port x pin on-chip pull-up devices disabled
1 = Port x pin on-chip pull-up devices enabled

NOTE

FPPUE and BPPUE do not apply in expanded mode because port F and B are address outputs.

PORTH — Port H Data **$007C**

	Bit 7	6	5	4	3	2	1	Bit 0
	PH7	PH6	PH5	PH4	PH3	PH2	PH1	PH0
RESET:	U	U	U	U	U	U	U	U
Alt. Pin Func.:	$\overline{\text{CSPROG}}$	CSGP2	CSGP1	CSIO	PW4	PW3	PW2	PW1

Port H pins reset to high-impedance inputs with selectable internal pull-up resistors. In expanded and special test modes, reset also causes PH7 to be configured as $\overline{\text{CSPROG}}$.

DDRH — Data Direction Register for Port H **$007D**

	Bit 7	6	5	4	3	2	1	Bit 0
	DDH7	DDH6	DDH5	DDH4	DDH3	DDH2	DDH1	DDH0
RESET:	0	0	0	0	1	1	1	1

DDH[7:0] — Data Direction for Port H

0 = Bits set to zero to configure corresponding I/O pin for input only

1 = Bits set to one to configure corresponding I/O pin for output

NOTE

PWM circuitry forces the I/O state to be an output for each port H line associated with an enabled pulse-width modulator channel. In these cases, data direction bits are not changed and have no effect on these lines. DDRH reverts to controlling the I/O state of a pin when the associated PWM function is disabled. Refer to **Pulse-Width Modulation (PWM) Timer** for further information.

PORTG — Port G Data **$007E**

	Bit 7	6	5	4	3	2	1	Bit 0
	PG7	PG6	PG5	PG4	PG3	PG2	PG1	PG0
RESET:	U	U	U	U	U	U	U	U
Alt. Pin Func.:	R/$\overline{W}$	—	DA2	DA1	AN11	AN10	AN9	AN8

Port G pins reset to high-impedance inputs with selectable internal pull-up resistors. In expanded and special test modes PG7 becomes R/$\overline{W}$.

DDRG — Data Direction Register for Port G **$007F**

	Bit 7	6	5	4	3	2	1	Bit 0
	DDG7	DDG6	0	0	0	0	0	0
RESET:	0	0	0	0	0	0	0	0

Only bits DDG[7:6] control data direction of corresponding Port G pins.

DDG[7:6] — Data Direction for Port G
0 = Configure corresponding I/O pin for input only
1 = Configure corresponding I/O pin for output

In expanded and test modes, bit 7 is configured for R/$\overline{W}$, forcing the state of this pin to be an output although the DDRG value remains 0.

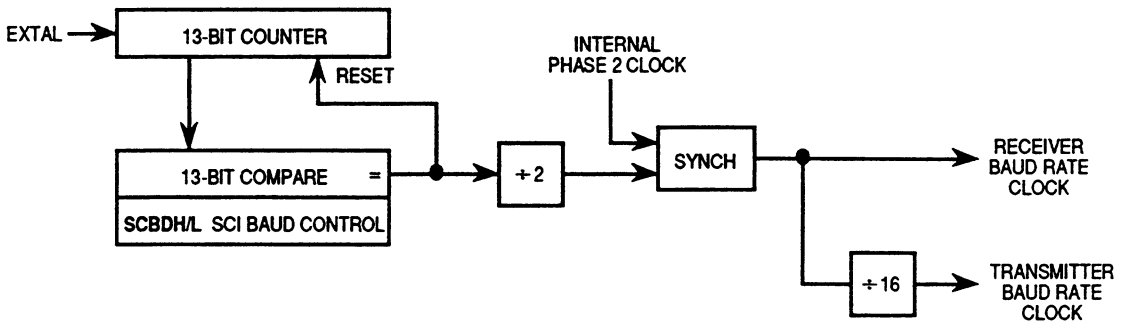
Bits [5:0] — Not implemented
Always read zero

Serial Communications Interface

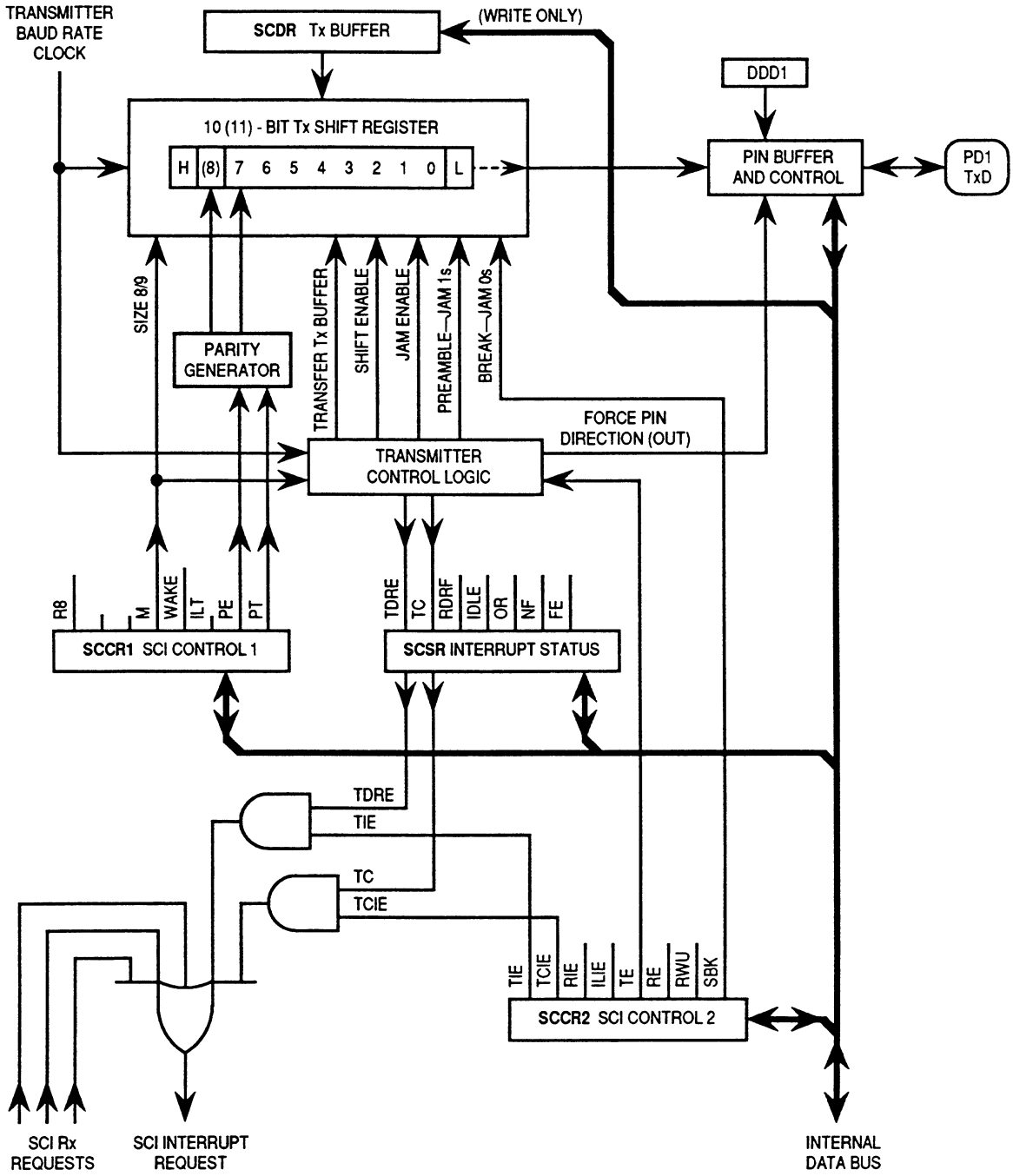
The SCI, a universal asynchronous receiver transmitter (UART) serial communications interface, is one of two independent serial I/O subsystems in the MC68HC11N4. Rearranging registers and control bits used in previous HC11 family devices has enhanced the existing SCI system and added new features, which include the following:

- A 13-bit modulus prescaler that allows greater baud rate control
- A new idle mode detect, independent of preceding serial data
- A receiver active flag
- Hardware parity for both transmitter and receiver

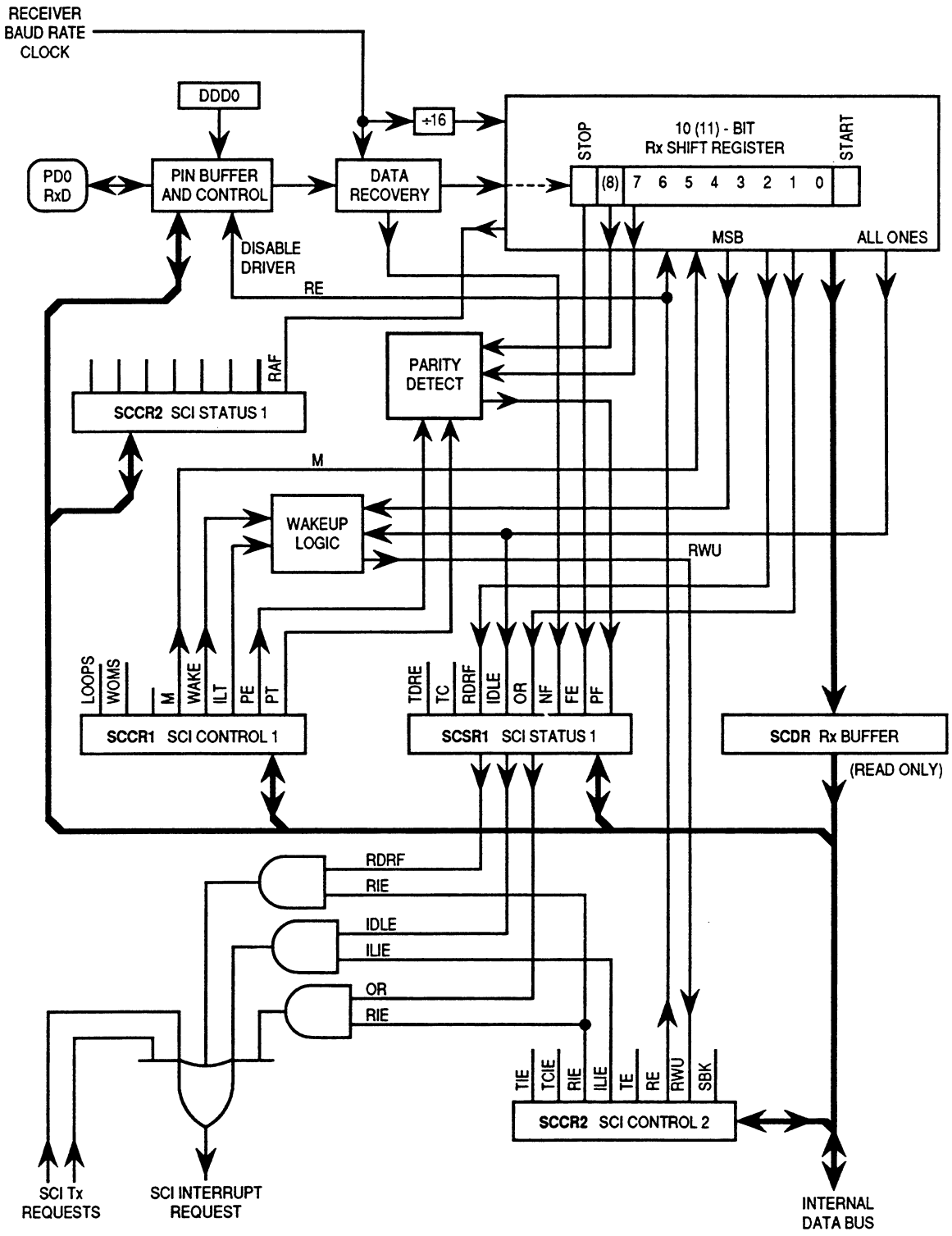
The enhanced baud rate generator is shown in the following diagram. Refer to the table of SCI baud rate control values for standard values.



SCI Baud Generator Circuit Diagram



SCI Transmitter Block Diagram



SCI Receiver Block Diagram

SPCR — Serial Peripheral Control **$0028**

	Bit 7	6	5	4	3	2	1	Bit 0
	SPIE	SPE	DWOM	MSTR	CPOL	CPHA	SPR1	SPR0
RESET:	0	0	0	0	0	0	0	0

SPIE — SPI Interrupt Enable
Refer to **Serial Peripheral Interface**.

SPE — SPI System Enable
Refer to **Serial Peripheral Interface**.

DWOM — Port D Wired-OR Mode Option for SPI Pins PD[5:2] (See also WOMS bit in SCCR1)
0 = Normal CMOS outputs
1 = Open-drain outputs

MSTR — Master/Slave Mode Select
Refer to **Serial Peripheral Interface**.

CPOL — Clock Polarity
Refer to **Serial Peripheral Interface**.

CPHA — Clock Phase
Refer to **Serial Peripheral Interface**.

SPR1, SPR0 — SPI Clock Rate Selects
Refer to **Serial Peripheral Interface**.

SCBDH/L — SCI Baud Rate Control High/Low **$0070, $0071**

	Bit 7	6	5	4	3	2	1	Bit 0	
$0070	BTST	BSPL	0	SBR12	SBR11	SBR10	SBR9	SBR8	High
RESET:	0	0	0	0	0	0	0	0	
$0071	SBR7	SBR6	SBR5	SBR4	SBR3	SBR2	SBR1	SBR0	Low
RESET:	0	0	0	0	0	1	0	0	

BTST — Baud Register Test (TEST)

BSPL — Baud Rate Counter Split (TEST)

Bit 5 — Not implemented
Always reads zero

SBR[12:0] — SCI Baud Rate Selects

Use the following formula to calculate SCI baud rate. Refer to the table of baud rate control values for example rates:

SCI baud rate = EXTAL ÷ [16 • (2 • BR)]

Where BR is the contents of SCBDH, L (BR = 1, 2, 3 ..., 8191).

BR = 0 disables the baud rate generator.

SCI Baud Rate Control Values

Target Baud Rate	Crystal Frequency (EXTAL)					
	8 MHz		12 MHz		16 MHz	
	Dec Value	Hex Value	Dec Value	Hex Value	Dec Value	Hex Value
110	2272	$08E0	3409	$0D51	4545	$11C1
150	1666	$0682	2500	$09C4	3333	$0D05
300	833	$0341	1250	$04E2	1666	$0682
600	416	$01A0	625	$0271	833	$0341
1200	208	$00D0	312	$0138	416	$01A0
2400	104	$0068	156	$009C	208	$00D0
4800	52	$0034	78	$004E	104	$0068
9600	26	$001A	39	$0027	52	$0034
19.2 K	13	$000D	20	$0014	26	$001A
38.4 K	—	—	—	—	13	$000D

SCCR1 — SCI Control 1 **$0072**

	Bit 7	6	5	4	3	2	1	Bit 0
	LOOPS	WOMS	0	M	WAKE	ILT	PE	PT
RESET:	0	0	0	0	0	0	0	0

LOOPS — SCI LOOP Mode Enable

0 = SCI transmit and receive operate normally

1 = SCI transmit and receive are disconnected from TxD and RxD pins, and transmitter output is fed back into the receiver input

WOMS — Wired-Or Mode for SCI Pins (PD1, PD0; see also DWOM bit in SPCR.)

0 = TxD and RxD operate normally

1 = TxD and RxD are open drains if operating as an output

Bit 5 — Not implemented

Always reads zero

M — Mode (Select Character Format)

0 = Start bit, 8 data bits, 1 stop bit

1 = Start bit, 9 data bits, 1 stop bit

WAKE — Wakeup by Address Mark/Idle

0 = Wakeup by IDLE line recognition

1 = Wakeup by address mark (most significant data bit set)

ILT — Idle Line Type
- 0 = Short (SCI counts consecutive ones after start bit)
- 1 = Long (SCI counts ones only after stop bit)

PE — Parity Enable
- 0 = Parity disabled
- 1 = Parity enabled

PT — Parity Type
- 0 = Parity even (even number of ones causes parity bit to be zero, odd number of ones causes parity bit to be one)
- 1 = Parity odd (odd number of ones causes parity bit to be zero, even number of ones causes parity bit to be one)

SCCR2 — SCI Control 2 **$0073**

	Bit 7	6	5	4	3	2	1	Bit 0
	TIE	TCIE	RIE	ILIE	TE	RE	RWU	SBK
RESET:	0	0	0	0	0	0	0	0

TIE — Transmit Interrupt Enable
- 0 = TDRE interrupts disabled
- 1 = SCI interrupt requested when TDRE status flag is set

TCIE — Transmit Complete Interrupt Enable
- 0 = TC interrupts disabled
- 1 = SCI interrupt requested when TC status flag is set

RIE — Receiver Interrupt Enable
- 0 = RDRF and OR interrupts disabled
- 1 = SCI interrupt requested when RDRF flag or the OR status flag is set

ILIE — Idle Line Interrupt Enable
- 0 = IDLE interrupts disabled
- 1 = SCI interrupt requested when IDLE status flag is set

TE — Transmitter Enable
- 0 = Transmitter disabled
- 1 = Transmitter enabled

RE — Receiver Enable
- 0 = Receiver disabled
- 1 = Receiver enabled

RWU — Receiver Wakeup Control
- 0 = Normal SCI receiver
- 1 = Wakeup enabled and receiver interrupts inhibited

SBK — Send Break
- 0 = Break generator off
- 1 = Break codes generated as long as SBK = 1

SCSR1 — SCI Status Register 1 **$0074**

	Bit 7	6	5	4	3	2	1	Bit 0
	TDRE	TC	RDRF	IDLE	OR	NF	FE	PF
RESET:	1	1	0	0	0	0	0	0

TDRE — Transmit Data Register Empty Flag

This flag is set when SCDR is empty. Clear the TDRE flag by reading SCSR1 with TDRE set and then writing to SCDR.

0 = SCDR busy
1 = SCDR empty

TC — Transmit Complete Flag

This flag is set when the transmitter is idle (no data, preamble, or break transmission in progress). Clear the TC flag by reading SCSR1 with TC set and then writing to SCDR.

0 = Transmitter busy
1 = Transmitter idle

RDRF — Receive Data Register Full Flag

Once cleared, IDLE is not set again until the RxD line has been active and becomes idle again. RDRF is set if a received character is ready to be read from SCDR. Clear the RDRF flag by reading SCSR1 with RDRF set and then reading SCDR.

0 = SCDR empty
1 = SCDR full

IDLE — Idle Line Detected Flag

This flag is set if the RxD line is idle. Once cleared, IDLE is not set again until the RxD line has been active and becomes idle again. The IDLE flag is inhibited when RWU = 1. Clear IDLE by reading SCSR1 with IDLE set and then reading SCDR.

0 = RxD line is active
1 = RxD line is idle

OR — Overrun Error Flag

OR is set if a new character is received before a previously received character is read from SCDR. Clear the OR flag by reading SCSR1 with OR set and then reading SCDR.

0 = No overrun
1 = Overrun detected

NF — Noise Error Flag

NF is set if majority sample logic detects anything other than a unanimous decision. Clear NF by reading SCSR1 with NF set and then reading SCDR.

0 = Unanimous decision
1 = Noise detected

FE — Framing Error

FE is set when a zero is detected where a stop bit was expected. Clear the FE flag by reading SCSR1 with FE set and then reading SCDR.

0 = Stop bit detected
1 = Zero detected

PF — Parity Error Flag

PF is set if received data has incorrect parity. Clear PF by reading SCSR1 with PE set and then reading SCDR.

0 = Parity correct

1 = Incorrect parity detected

SCSR2 — SCI Status Register 2 **$0075**

	Bit 7	6	5	4	3	2	1	Bit 0
	0	0	0	0	0	0	0	RAF
RESET:	0	0	0	0	0	0	0	0

Bits [7:1] — Not implemented

Always read zero

RAF — Receiver Active Flag (Read only)

0= A character is not being received

1 = A character is being received

SCDRH, SCDRL — SCI Data Register High/ Low **$0076, $0077**

	Bit 7	6	5	4	3	2	1	Bit 0	
$0076	R8	T8	0	0	0	0	0	0	SCDRH (High)
$0077	R7/T7	R6/T6	R5/T5	R4/T4	R3/T3	R2/T2	R1/T1	R0/T0	SCDRL (Low)

R8 — Receiver Bit 8

Ninth serial data bit received when SCI is configured for a nine data bit operation.

T8 — Transmitter Bit 8

Ninth serial data bit transmitted when SCI is configured for a nine data bit operation.

Bits [5:0] — Not implemented

Always read zero

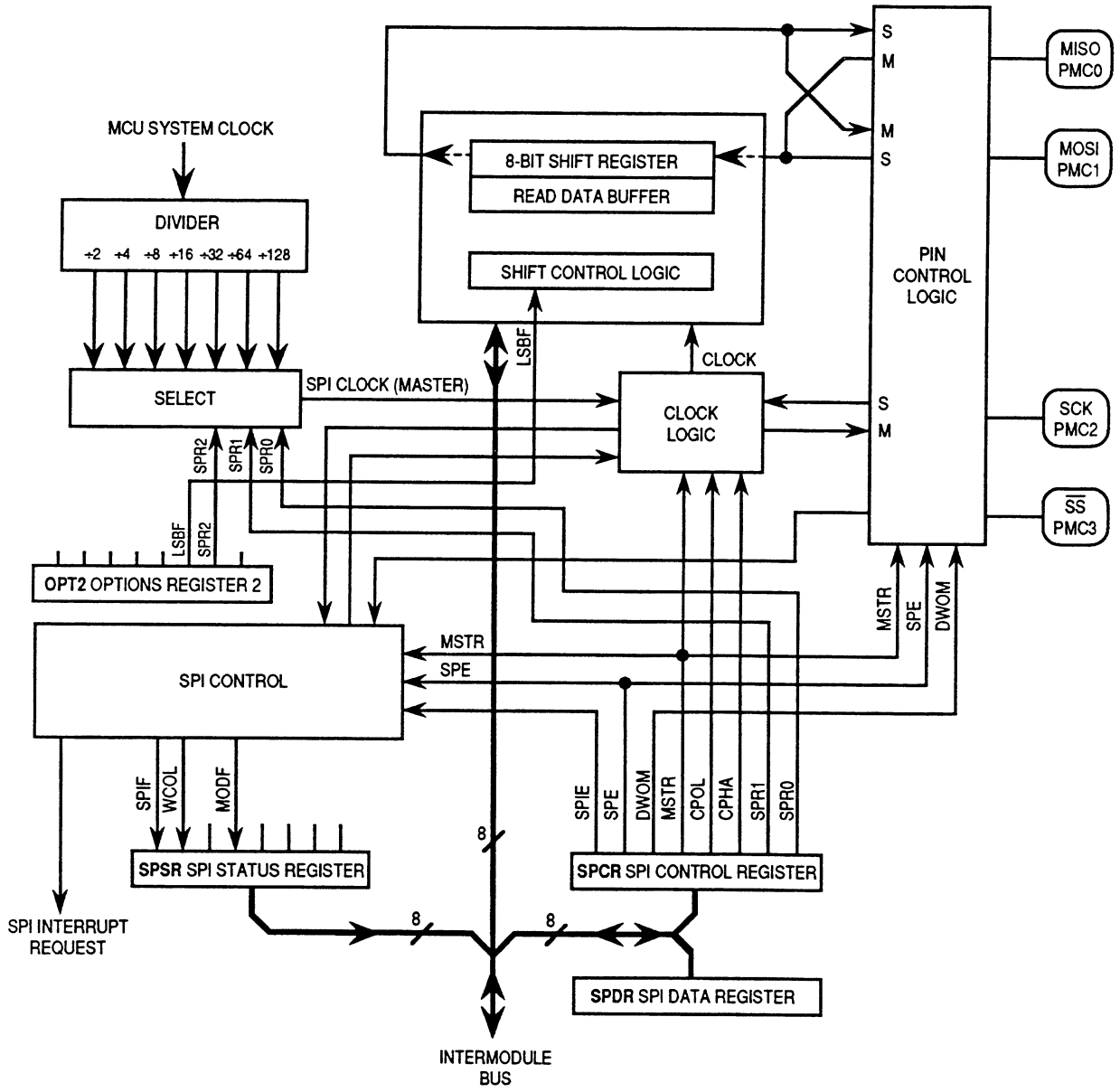
R[7:0]/T[7:0] — Receiver/Transmitter Data Bits 7 to 0

SCI data is double buffered in both directions.

Serial Peripheral Interface

The SPI allows the MCU to communicate synchronously with peripheral devices and other microprocessors. Data rates can be as high as 2 Megabits per second when configured as a master and 4 Megabits per second when configured as a slave (assuming 4 MHz bus speed).

Two control bits in OPT2 allow the transfer of data either MSB or LSB first and select an additional divide by four stage to be inserted before the SPI baud rate clock divider.



SPI Block Diagram

SPCR — Serial Peripheral Control Register **$0028**

	Bit 7	6	5	4	3	2	1	Bit 0
	SPIE	SPE	DWOM	MSTR	CPOL	CPHA	SPR1	SPR0
RESET:	0	0	0	0	0	1	U	U

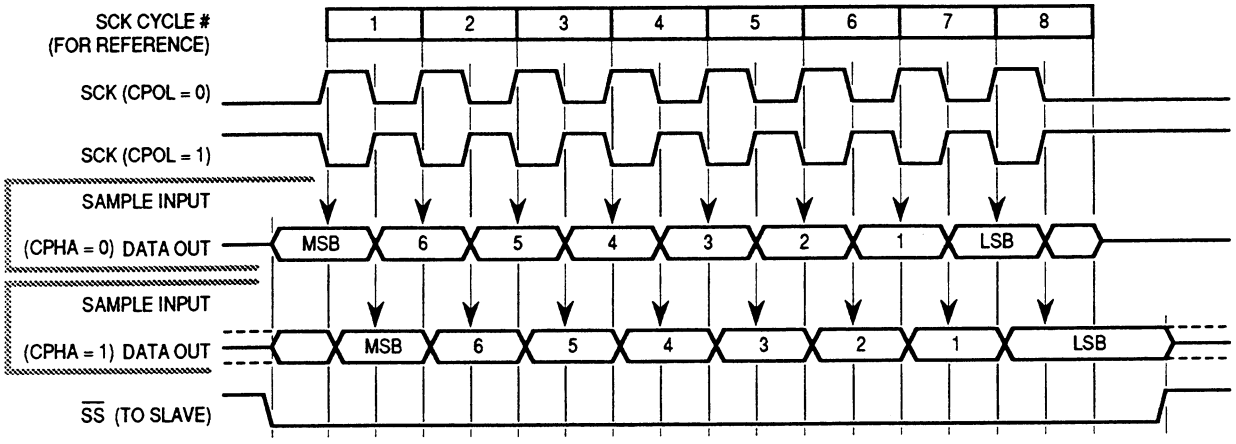
SPIE — Serial Peripheral Interrupt Enable
0 = SPI interrupts disabled
1 = SPI interrupts enabled

SPE — Serial Peripheral System Enable
0 = SPI off
1 = SPI on

DWOM — Port D Wired-OR Mode Option for SPI Pins PD[5:2] (See also WOMS bit in SCCR2.)
0 = Normal CMOS outputs
1 = Open-drain outputs

MSTR — Master Mode Select
0 = Slave mode
1 = Master mode

CPOL, CPHA — Clock Polarity, Clock Phase
Refer to the following figure, **SPI Transfer Format**.



SPI Transfer Format

NOTE

This figure shows transmission order when LSBF = 0 (default). If LSBF = 1, data is transferred in reverse order (LSB first).

SPR2, SPR1 and SPR0 — SPI Clock Rate Selects (SPR2 Is located in OPT2 register)

SPI Clock Rate Selects

SPR[2:0]	Divide E Clock By	Frequency at E = 2 MHz (Baud)
0 0 0	2	1.0 MHz
0 0 1	4	500 kHz
0 1 0	16	125 kHz
0 1 1	32	62.5 kHz
1 0 0	8	250 kHz
1 0 1	16	125 kHz
1 1 0	64	31.3 kHz
1 1 1	128	15.6 kHz

SPSR — Serial Peripheral Status Register **$0029**

	Bit 7	6	5	4	3	2	1	Bit 0
	SPIF	WCOL	0	MODF	0	0	0	0
RESET:	0	0	0	0	0	0	0	0

SPIF — SPI Transfer Complete Flag

SPIF is set when an SPI transfer is complete. This bit is cleared by reading SPSR with SPIF set, followed by SPDR access (read or write).

WCOL — Write Collision

WCOL is set when SPDR is written while transfer is in progress. This bit is cleared by reading SPSR with WCOL set, followed by SPDR access (read or write).

Bit 5 — Not implemented

Always reads zero

MODF — Mode Fault (Mode fault terminates SPI operation)

MODF is set when $\overline{SS}$ is pulled low while MSTR = 1. This bit is cleared by an SPCR read with MODF set, followed by SPCR write.

Bits [3:0] — Not implemented

Always read zero

SPDR — SPI Data **$002A**

Bit 7	6	5	4	3	2	1	Bit 0
Bit 7	6	5	4	3	2	1	Bit 0

SPI is double buffered in, single buffered out.

OPT2 — System Configuration Options 2 **$0038**

	Bit 7	6	5	4	3	2	1	Bit 0
	LIRDV	CWOM	STRCH	IRVNE	LSBF	SPR2	0	0
RESET:	0	0	0	—	0	0	0	0

LIRDV — LIR Driven
Refer to **Operating Modes and On-Chip Memory**.

CWOM — Port C Wired-OR Mode
Refer to **Parallel Input/Output**.

STRCH — Stretch
0 = Normal operation
1 = Off-chip accesses to $0000–$7FFF extended by one E-clock cycle

IRVNE — Internal Read Visibility/Not E
Refer to **Operating Modes and On-Chip Memory**.

LSBF — SPI LSB First Enable
0 = SPI data transferred MSB first
1 = SPI data transferred LSB first

SPR2 — SPI Clock (SCK) Rate Select
Adds a divide by four prescaler to SPI clock chain. Refer to SPCR register.

Bits [1:0] — Not implemented
Always read zero

Main Timer

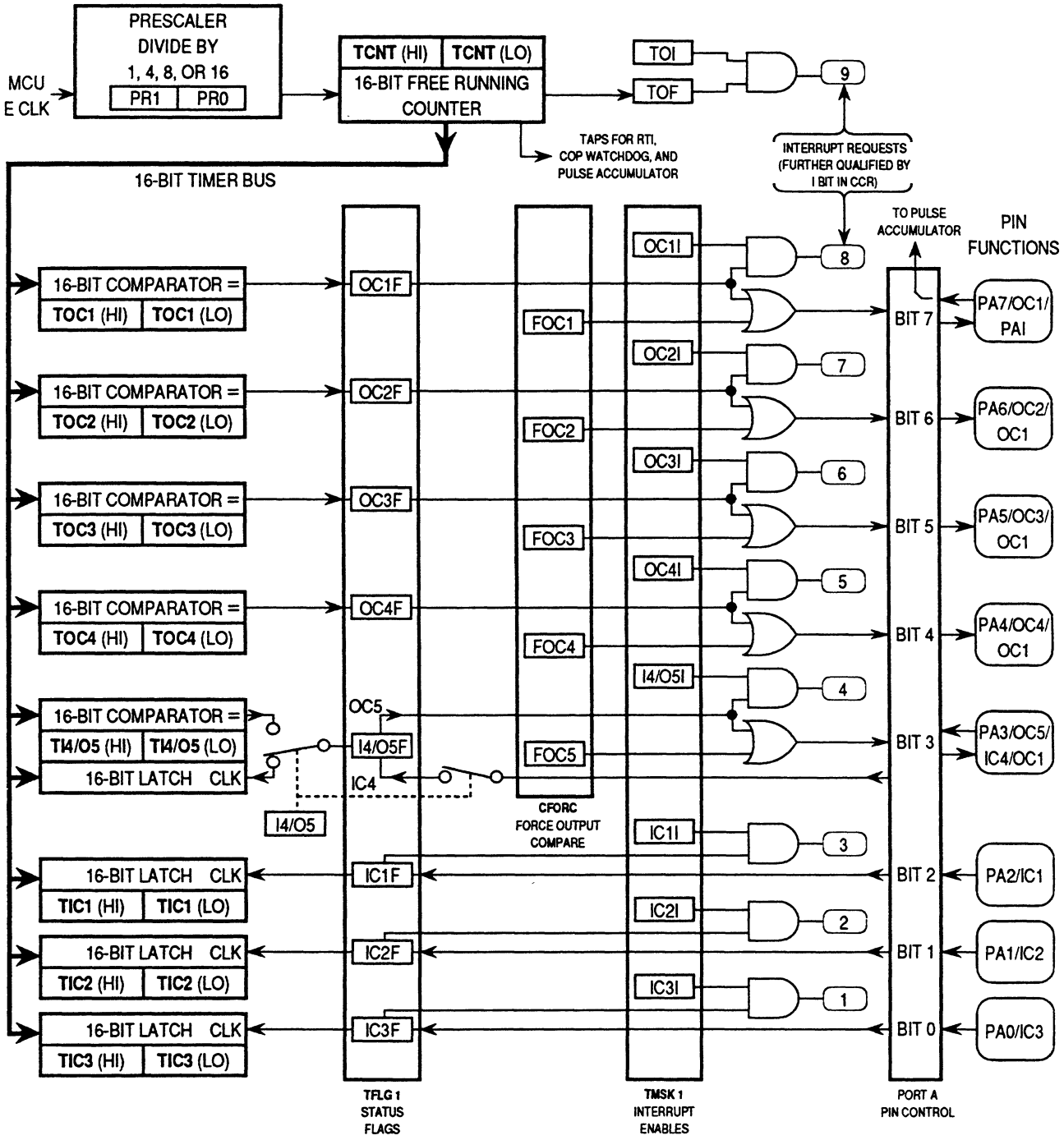
The timing system is based on a free-running 16-bit counter with a four-stage programmable prescaler. A timer overflow function allows software to extend the system's timing capability beyond the counter's 16-bit range.

The timer has three channels of input capture, four channels of output compare, and one channel that can be configured as a fourth input capture or a fifth output compare. In addition, the timing system includes pulse accumulator and real-time interrupt (RTI) functions, as well as a clock monitor function, which can be used to detect clock failures that are not detected by the COP.

Refer to **Pulse Accumulator** and **Real-Time Interrupt** for further information about these functions. Refer to the following table for a summary of the crystal-related frequencies and periods.

Timer Summary

Control Bits	XTAL Frequencies			
	8.0 MHz	12.0 MHz	16.0 MHz	Other Rates
	2.0 MHz	3.0 MHz	4.0 MHz	(E)
	500 ns	333 ns	250 ns	(1/E)
PR[1:0]	Main Timer Count Rates			
0 0 1 count — overflow —	 500 ns 32.768 ms	 333 ns 21.845 ms	 250 ns 16.384 ms	 (E/1) ($E/2^{16}$)
0 1 1 count — overflow —	 2.0 µs 131.07 ms	 1.333 µs 87.381 ms	 1.0 µs 65.536 ms	 (E/4) ($E/2^{18}$)
1 0 1 count — overflow —	 4.0 µs 262.14 ms	 2.667 µs 174.76 ms	 2.0 µs 131.07 ms	 (E/8) ($E/2^{19}$)
1 1 1 count — overflow —	 8.0 µs 524.29 ms	 5.333 µs 349.52 ms	 4.0 µs 262.14 ms	 (E/16) ($E/2^{20}$)
RTR[1:0]	Periodic (RTI) Interrupt Rates			
0 0	4.096 ms	2.731 ms	2.048 ms	($E/2^{13}$)
0 1	8.192 ms	5.461 ms	4.096 ms	($E/2^{14}$)
1 0	16.384 ms	10.923 ms	8.192 ms	($E/2^{15}$)
1 1	32.768 ms	21.845 ms	16.384 ms	($E/2^{16}$)
CR[1:0]	COP Watchdog Timeout Rates			
0 0	16.384 ms	10.923 ms	8.192 ms	($E/2^{15}$)
0 1	65.536 ms	43.691 ms	32.768 ms	($E/2^{17}$)
1 0	262.14 ms	174.76 ms	131.07 ms	($E/2^{19}$)
1 1	1.049 s	699.05 ms	524.28 ms	($E/2^{21}$)
Timeout Tolerance (– 0 ms/+...)	16.4 ms	10.9 ms	8.192 ms	($E/2^{15}$)



Timer Block Diagram

CFORC — Timer Compare Force **$000B**

	Bit 7	6	5	4	3	2	1	Bit 0
	FOC1	FOC2	FOC3	FOC4	FOC5	0	0	0
RESET:	0	0	0	0	0	0	0	0

FOC[5:1] — Write ones to force compare(s)
0 = Not affected
1 = Output x action occurs

Bits [2:0] — Not implemented
Always read zero

OC1M — Output Compare 1 Mask **$000C**

	Bit 7	6	5	4	3	2	1	Bit 0
	OC1M7	OC1M6	OC1M5	OC1M4	OC1M3	0	0	0
RESET:	0	0	0	0	0	0	0	0

Set bit(s) to enable OC1 to control corresponding pin(s) of port A

Bits [2:0] — Not implemented
Always read zero

OC1D — Output Compare 1 Data **$000D**

	Bit 7	6	5	4	3	2	1	Bit 0
	OC1D7	OC1D6	OC1D5	OC1D4	OC1D3	0	0	0
RESET:	0	0	0	0	0	0	0	0

If OC1Mx is set, data in OC1Dx is output to port A bit x on successful OC1 compares.

Bits [2:0] — Not implemented
Always read zero

TCNT — Timer Count **$000E, $000F**

$000E	Bit 15	14	13	12	11	10	9	Bit 8	High	**TCNT**
$000F	Bit 7	6	5	4	3	2	1	Bit 0	Low	

TCNT resets to $0000

In normal modes, TCNT is a read-only register.

TIC1–TIC3 — Timer Input Capture **$0010–$0015**

$0010	Bit 15	14	13	12	11	10	9	Bit 8	High	**TIC1**
$0011	Bit 7	6	5	4	3	2	1	Bit 0	Low	
$0012	Bit 15	14	13	12	11	10	9	Bit 8	High	**TIC2**
$0013	Bit 7	6	5	4	3	2	1	Bit 0	Low	
$0014	Bit 15	14	13	12	11	10	9	Bit 8	High	**TIC3**
$0015	Bit 7	6	5	4	3	2	1	Bit 0	Low	

TICx not affected by reset

TOC1–TOC4 — Timer Output Compare **$0016–$001D**

$0016	Bit 15	14	13	12	11	10	9	Bit 8	High	**TOC1**
$0017	Bit 7	6	5	4	3	2	1	Bit 0	Low	
$0018	Bit 15	14	13	12	11	10	9	Bit 8	High	**TOC2**
$0019	Bit 7	6	5	4	3	2	1	Bit 0	Low	
$001A	Bit 15	14	13	12	11	10	9	Bit 8	High	**TOC3**
$001B	Bit 7	6	5	4	3	2	1	Bit 0	Low	
$001C	Bit 15	14	13	12	11	10	9	Bit 8	High	**TOC4**
$001D	Bit 7	6	5	4	3	2	1	Bit 0	Low	

All TOCx register pairs reset to ones ($FFFF).

TI4/O5 — Timer Input Capture 4/Output Compare 5 **$001E, $001F**

$001E	Bit 15	14	13	12	11	10	9	Bit 8	High
$001F	Bit 7	6	5	4	3	2	1	Bit 0	Low

This is a shared register and is either input capture 4 or output compare 5 depending on the state of bit I4/O5 in PACTL. Writes to TI4/O5 have no effect when this register is configured as input capture 4. The TI4/O5 register pair resets to ones ($FFFF).

TCTL1 — Timer Control 1 **$0020**

	Bit 7	6	5	4	3	2	1	Bit 0
	OM2	OL2	OM3	OL3	OM4	OL4	OM5	OL5
RESET:	0	0	0	0	0	0	0	0

OM[5:2] — Output Mode

OL[5:2] — Output Level

OMx	OLx	Action Taken on Successful Compare
0	0	Timer disconnected from output pin logic
0	1	Toggle OCx output line
1	0	Clear OCx output line to zero
1	1	Set OCx output line to one

TCTL2 — Timer Control 2 **$0021**

	Bit 7	6	5	4	3	2	1	Bit 0
	EDG4B	EDG4A	EDG1B	EDG1A	EDG2B	EDG2A	EDG3B	EDG3A
RESET:	0	0	0	0	0	0	0	0

Timer Control Configuration

EDGxB	EDGxA	Configuration
0	0	Capture disabled
0	1	Capture on rising edges only
1	0	Capture on falling edges only
1	1	Capture on any edge

TMSK1 — Timer Interrupt Mask 1 **$0022**

	Bit 7	6	5	4	3	2	1	Bit 0
	OC1I	OC2I	OC3I	OC4I	I4/O5I	IC1I	IC2I	IC3I
RESET:	0	0	0	0	0	0	0	0

OC1I–OC4I — Output Compare x Interrupt Enable

I4/O5I — Input Capture 4/Output Compare 5 Interrupt Enable

IC1I–IC3I — Input Capture x Interrupt Enable

NOTE

Control bits in TMSK1 correspond bit for bit with flag bits in TFLG1. Ones in TMSK1 enable the corresponding interrupt sources.

TFLG1 — Timer Interrupt Flag 1 **$0023**

	Bit 7	6	5	4	3	2	1	Bit 0
	OC1F	OC2F	OC3F	OC4F	I4/O5F	IC1F	IC2F	IC3F
RESET:	0	0	0	0	0	0	0	0

Clear flags by writing a one to the corresponding bit position(s).

OC1F–OC5F — Output Compare x Flag
Set each time the counter matches output compare x value

I4/O5F — Input Capture 4/Output Compare 5 Flag
Set by IC4 or OC5, depending on which function was enabled by I4/O5 of PACTL

IC1F–IC3F — Input Capture x Flag
Set each time a selected active edge is detected on the ICx input line

TMSK2 — Timer Interrupt Mask 2 **$0024**

	Bit 7	6	5	4	3	2	1	Bit 0
	TOI	RTII	PAOVI	PAII	0	0	PR1	PR0
RESET:	0	0	0	0	0	0	0	0

TOI — Timer Overflow Interrupt Enable
0 = Timer overflow interrupt disabled
1 = Timer overflow interrupt enabled

RTII — Real-Time Interrupt Enable
0 = RTIF interrupts disabled
1 = Interrupt requested when RTIF is set to one

NOTE

Control bits [7:4] in TMSK2 correspond bit for bit with flag bits [7:4] in TFLG2. Ones in TMSK2 enable the corresponding interrupt sources.

PAOVI — Pulse Accumulator Overflow Interrupt Enable
Refer to **Pulse Accumulator**.

PAII — Pulse Accumulator Interrupt Enable
Refer to **Pulse Accumulator**.

Bits [3:2] — Not implemented
Always read zero

PR[1:0] — Timer Prescaler Select
In normal modes, PR1 and PR0 can only be written once, and the write must occur within 64 cycles after reset. Refer to **Timer Summary** for specific timing values.

PR[1:0]	Prescaler
0 0	1
0 1	4
1 0	8
1 1	16

TFLG2 — Timer Interrupt Flag 2 **$0025**

	Bit 7	6	5	4	3	2	1	Bit 0
	TOF	RTIF	PAOVF	PAIF	0	0	0	0
RESET:	0	0	0	0	0	0	0	0

Clear flags by writing a one to the corresponding bit position(s).

TOF — Timer Overflow Flag
Set when TCNT changes from $FFFF to $0000

RTIF — Real-Time (Periodic) Interrupt Flag
Set periodically. Refer to RTR[1:0] bits in PACTL register.

PAOVF — Pulse Accumulator Overflow Flag
Refer to **Pulse Accumulator**.

PAIF — Pulse Accumulator Input Edge Flag
Refer to **Pulse Accumulator**.

Bits [3:0] — Not implemented
Always read zero

PACTL — Pulse Accumulator Control **$0026**

	Bit 7	6	5	4	3	2	1	Bit 0
	0	PAEN	PAMOD	PEDGE	0	I4/O5	RTR1	RTR0
RESET:	0	0	0	0	0	0	0	0

Bit 7 — Not implemented
Always read zero

PAEN — Pulse Accumulator System Enable
Refer to **Pulse Accumulator**.

PAMOD — Pulse Accumulator Mode
Refer to **Pulse Accumulator**.

PEDGE — Pulse Accumulator Edge Control
Refer to **Pulse Accumulator**.

Bit 3 — Not implemented
Always reads zero

I4/O5 — Input Capture 4/Output Compare 5
Configure TI4/O5 for input capture or output compare
0 = OC5 enabled
1 = IC4 enabled

RTR[1:0] — Real-Time Interrupt (RTI) Rate
Refer to **Real-Time Interrupt**.

Real-Time Interrupt

The real-time interrupt (RTI) function can generate interrupts at different fixed periodic rates. These rates are a function of the MCU oscillator frequency and the value of the software-accessable control bits, RTR1 and RTR0. These bits determine the rate at which interrupts are requested by the RTI system. The RTI system is driven by an E divided by 2^{13} rate clock compensated so that it is independent of the timer prescaler. The RTR1 and RTR0 control bits select an additional division factor. RTI is set to its fastest rate by default out of reset and can be changed at any time. Refer to interrupt enable and flag bits in TMSK2 and TFLG2 registers respectively. Refer to interrupt enable and flag bits in TMSK2 and TFLG2 registers respectively.

Real-Time Interrupt Rates

RTR [1:0]	Divide E By	XTAL = 2^{23} MHz	XTAL = 8.0 MHz	XTAL = 12.0 MHz	XTAL = 16.0 MHz
0 0	2^{13}	3.91 ms	4.096 ms	2.731 ms	2.048 ms
0 1	2^{14}	7.81 ms	8.192 ms	5.461 ms	4.096 ms
1 0	2^{15}	15.62 ms	16.384 ms	10.923 ms	8.192 ms
1 1	2^{16}	31.25 ms	32.768 ms	21.845 ms	16.383 ms
	E =	2.1 MHz	2.0 MHz	3.0 MHz	4.0 MHz

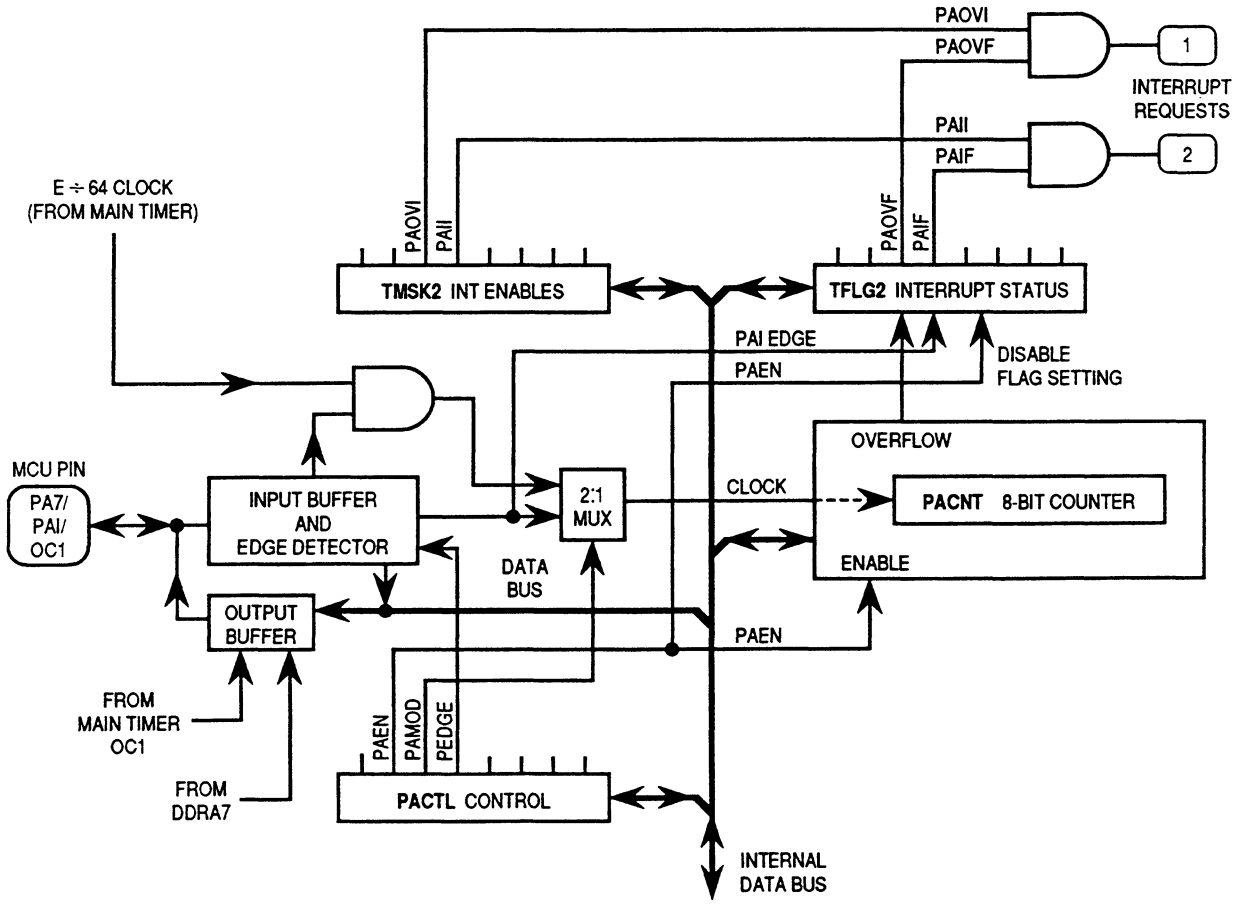
Refer to interrupt enable and flag bits in TMSK2 and TFLG2 registers.

Pulse Accumulator

The MC68HC11N4 has an 8-bit counter that can be configured as a simple event counter or for gated time accumulation. The counter can be read or written at any time.

The port A bit 7 I/O pin can be configured to act as a clock in event counting mode, or as a gate signal to enable a free-running clock (E divided by 64) to the 8-bit counter in gated time accumulation mode.

	Selected Crystal	Common XTAL Frequencies		
		8.0 MHz	12.0 MHz	16.0 MHz
CPU Clock	(E)	2.0 MHz	3.0 MHz	4.0 MHz
Cycle Time	(1/E)	500 ns	333 ns	250 ns
Pulse Accumulator (Gated Mode)				
$(E/2^6)$	1 count —	32.0 µs	21.330 µs	16.0 µs
$(E/2^{14})$	overflow —	8.192 ms	5.491 ms	4.096 ms



Pulse Accumulator System Block Diagram

TMSK2 — Timer Interrupt Mask 2 **$0024**

	Bit 7	6	5	4	3	2	1	Bit 0
	TOI	RTII	PAOVI	PAII	0	0	PR1	PR0
RESET:	0	0	0	0	0	0	0	0

TOI — Timer Overflow Interrupt Enable
Refer to **Main Timer**.

RTII — Real-Time Interrupt Enable
Refer to **Main Timer**.

PAOVI — Pulse Accumulator Overflow Interrupt Enable
0 = Pulse accumulator overflow interrupt disabled
1 = Pulse accumulator overflow interrupt enabled

PAII — Pulse Accumulator Input Interrupt Enable
0 = Pulse accumulator input interrupt disabled
1 = Pulse accumulator input interrupt enabled if PAIF bit in TFLG2 register is set

Bits [3:2] — Not implemented
Always read zero

PR[1:0] — Timer Prescaler Select
Refer to **Main Timer**.

NOTE

Control bits [7:4] in TMSK2 correspond bit for bit with flag bits [7:4] in TFLG2. Ones in TMSK2 enable the corresponding interrupt sources.

TFLG2 — Timer Interrupt Flag 2 **$0025**

	Bit 7	6	5	4	3	2	1	Bit 0
	TOF	RTIF	PAOVF	PAIF	0	0	0	0
RESET:	0	0	0	0	0	0	0	0

Clear flags by writing a one to the corresponding bit position(s).

TOF — Timer Overflow Enable
Refer to **Main Timer**.

RTIF — Real-Time Interrupt Flag
Refer to **Main Timer**.

PAOVF — Pulse Accumulator Overflow Flag
Set when PACNT changes from $FF to $00

PAIF — Pulse Accumulator Input Edge Flag
Set each time a selected active edge is detected on the PAI input line

Bits [3:0] — Not implemented
Always read zero

PACTL — Pulse Accumulator Control **$0026**

	Bit 7	6	5	4	3	2	1	Bit 0
	0	PAEN	PAMOD	PEDGE	0	I4/O5	RTR1	RTR0
RESET:	0	0	0	0	0	0	0	0

Bit 7 and 3 — Not implemented
Always read zero

PAEN — Pulse Accumulator System Enable
0 = Pulse accumulator disabled
1 = Pulse accumulator enabled

PAMOD — Pulse Accumulator Mode
0 = Event counter
1 = Gated time accumulation

PEDGE — Pulse Accumulator Edge Control
0 = In event mode, falling edges increment counter. In gated accumulation mode, high level enables accumulator and falling edge sets PAIF.
1 = In event mode, rising edges increment counter. In gated accumulation mode, low level enables accumulator and rising edge sets PAIF.

I4/O5 — Input Capture 4/Output Compare 5
Refer to **Main Timer**.

RTR[1:0] — Real-Time Interrupt Rate
Refer to **Real-Time Interrupt**.

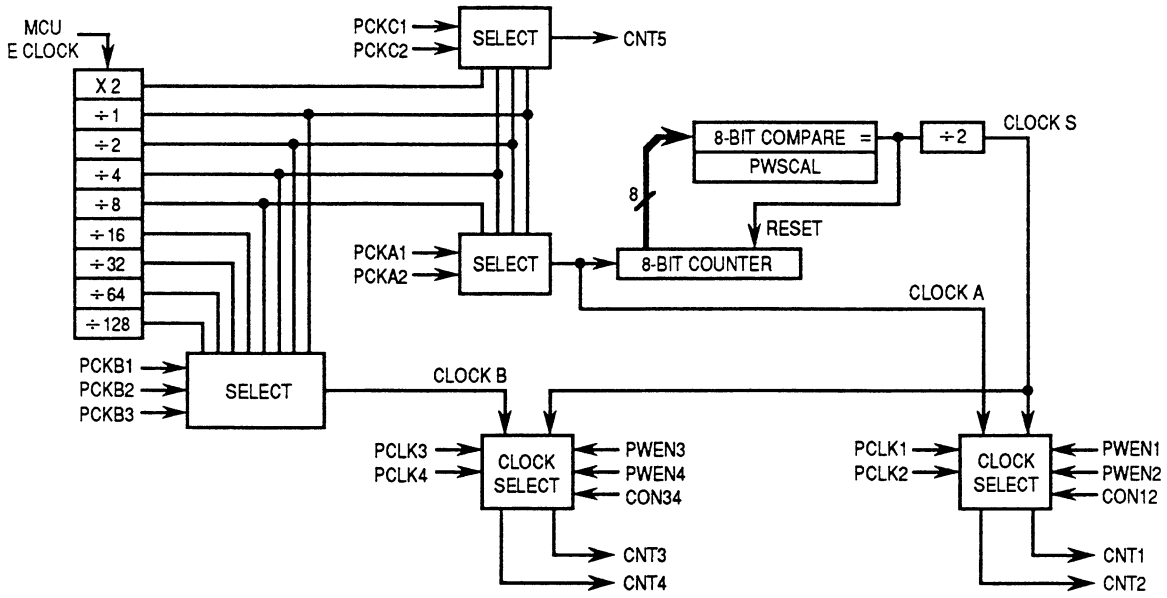
PACNT — Pulse Accumulator Counter **$0027**

Bit 7	6	5	4	3	2	1	Bit 0
Bit 7	6	5	4	3	2	1	Bit 0

Can be read and written

Pulse-Width Modulation (PWM) Timer

The MC68HC11N4 MCU contains a six-channel PWM timer that is composed of a two-channel 12-bit modulator and a four-channel 8-bit modulator. Each of the modulators can create independent continuous waveforms with software-selectable duty rates from 0% to 100%. The following diagram shows the clock system used by both the two-channel and four-channel modulators.



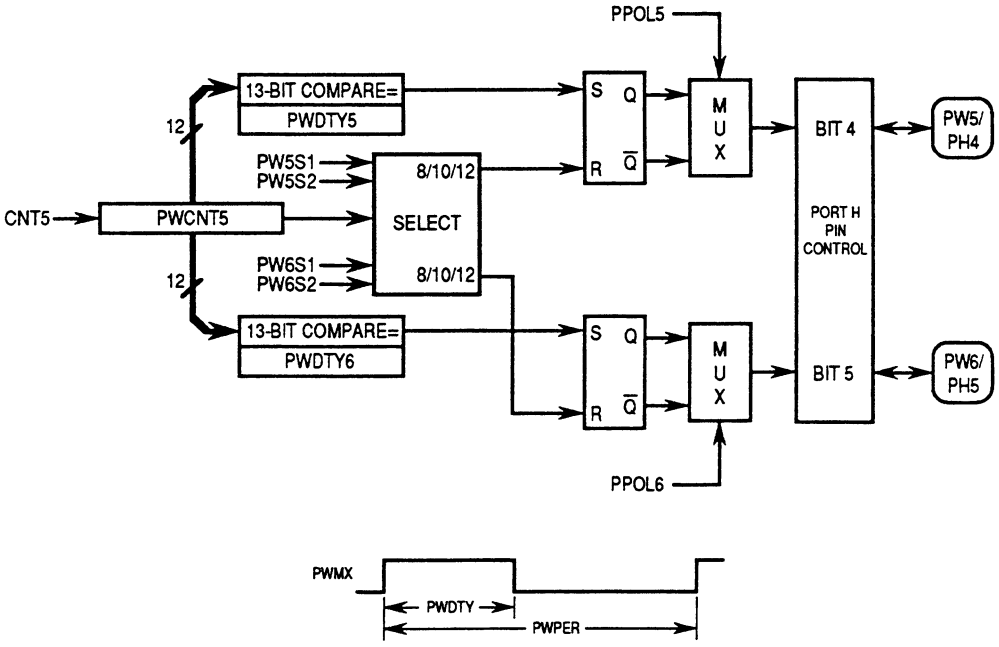
MC68HC11N4 PWM System Clock Source

Two-Channel Modulator

Two of the PWM channels, 5 and 6, share a 12-bit counter. The period of each PWM signal begins when the counter rolls over. Each channel contains a duty register that causes the output to change state when the content of the duty register matches the value of the counter. A control bit for each channel allows polarity selection. Also, the period for each channel can be selected as 8-, 10-, or 12- bit counts.

Each duty register is double buffered. When a channel is active, writes to the duty register are buffered until the counter rolls over or the channel is disabled. At this time the new value takes effect. This design requires the output to be either the old duty waveform or the new duty waveform. If the channel is not enabled, writes to the duty register go directly to both the latches and the buffer. Refer to the channel 5 and 6 block diagram.

Two control registers, PWSIZ and PWCTL, configure the outputs of channels 5 and 6. When the enable bit is set to one, the pulse modulated signal is available at the associated port H line. Channel 5 is port H bit 4. Channel 6 is port H bit 5. The following block diagram shows the two 12-bit channels of the MC68HC11N4 PWM system.



Two 12-Bit Channels of the PWM System

Four-Channel Modulator

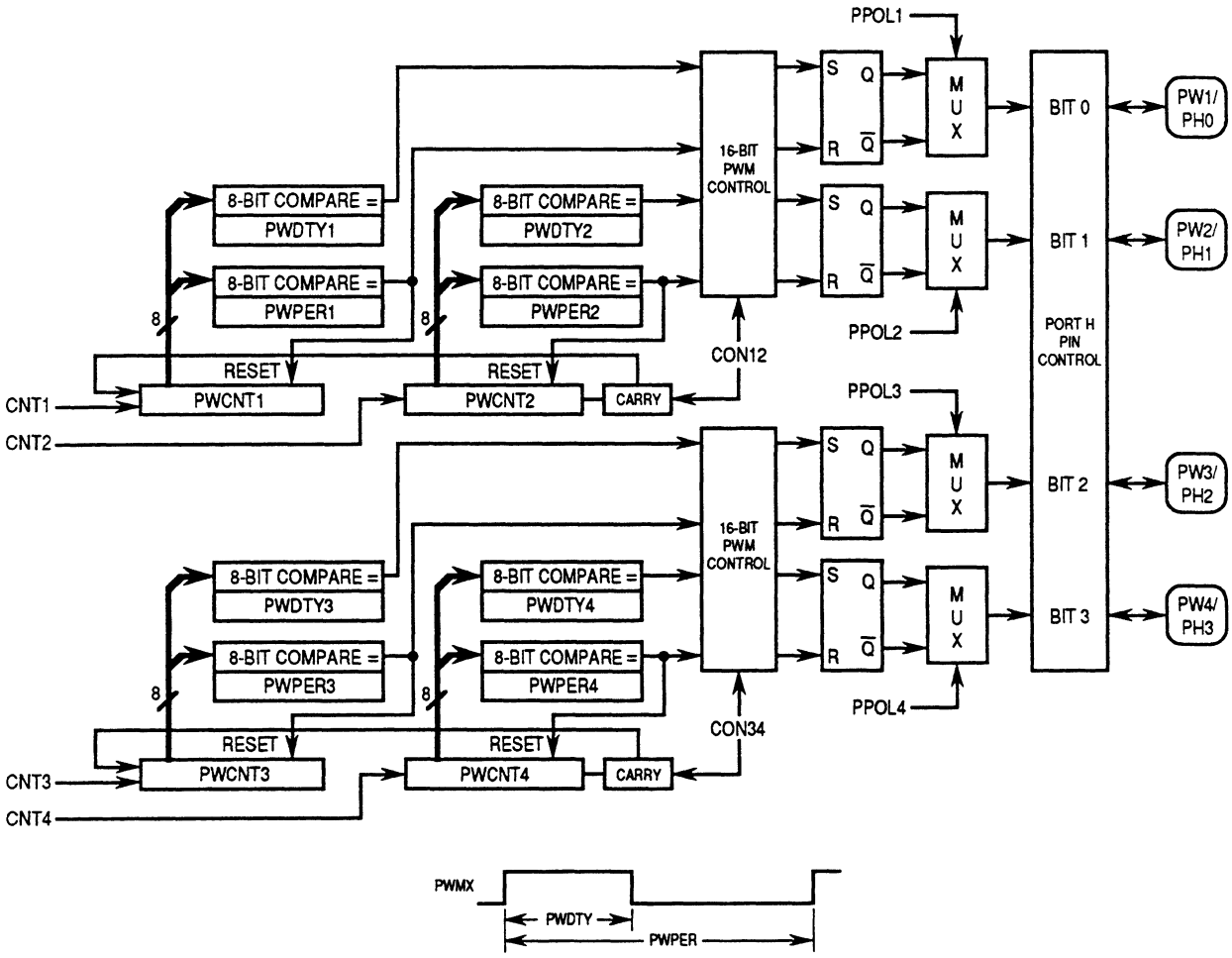
The four-channel modulator provides up to four pulse-width modulated waveforms on specific port H pins. Each channel has its own counter. Pairs of counters can be concatenated to create 16-bit PWM outputs based on 16-bit counts. Three clock sources (A, B, and S) give the PWM a wide range of frequencies.

Four control registers configure the PWM outputs — PWCLK, PWPOL, PWSCAL, and PWEN. The PWCLK register selects the prescale value for PWM clock sources and enables the 16-bit counters. The PWPOL register determines each channel's polarity and selects the clock source for each channel. The PWSCAL register derives a user-scaled clock, based on the A clock source, and the PWEN register enables the PWM channels.

Each channel has a separate 8-bit counter, period register, and duty cycle register. The period and duty cycle registers are double buffered so that if they are changed while the channel is enabled, the change does not take effect until the counter rolls over or the channel is disabled.

With channels configured for 8-bit mode and E = 4 MHz, PWM signals of 40 kHz (1% duty cycle resolution) to less than 10 Hz (approximately 0.4% duty cycle resolution) can be produced. By configuring the channels for 16-bit mode with E = 4 MHz, PWM periods greater than one minute are possible.

In 16-bit mode, duty cycle resolution of almost 15 parts per million can be achieved (at a PWM frequency of about 60 Hz). In the same system, a PWM frequency of 1 kHz corresponds to a duty cycle resolution of 0.025%. The following block diagram shows the four 8-bit channels of the MC68HC11N4 PWM system.



Four 8-Bit Channels of the PWM System

PWCTL — Pulse-Width Control for Channels 5 and 6 **$0050**

	Bit 7	6	5	4	3	2	1	Bit 0
	0	0	PCKC2	PCKC1	0	0	PPOL6	PPOL5
RESET:	0	0	0	0	0	0	0	0

Bits [7:6] — Not implemented
Always read zero

PCKC[2:1] — Prescaler for Clock C
Determine rate of clock C

PCKC[2:1]	Value of Clock C
0 0	2E
0 1	E
1 0	E/2
1 1	E/4

Bits [3:2] — Not implemented
Always read zero

PPOL6 — Pulse-Width Channel 6 Polarity
0 = PWM channel 6 output is low at start of clock cycle, then goes high when duty count is reached
1 = PWM channel 6 output is high at start of clock cycle, then goes low when duty count is reached

PPOL5 — Pulse-Width Channel 5 Polarity
0 = PWM channel 5 output is low at start of clock cycle, then goes high when duty count is reached
1 = PWM channel 5 output is high at start of clock cycle, then goes low when duty count is reached

PWSIZ — Pulse-Width Size and Enable **$0051**

	Bit 7	6	5	4	3	2	1	Bit 0
	0	0	0	0	PW6S2	PW6S1	PW5S2	PW5S1
RESET:	0	0	0	0	0	0	0	0

Bits [7:4] — Not implemented
Always read zero

PW6S2, PW6S1 — Pulse-Width Channel 6 Size Select and Enable
Period of pulse-width channel 6 is determined as shown in the following table

PW5S2, PW5S1 — Pulse-Width Channel 5 Size Select and Enable
Period of pulse-width channel 5 is determined as shown in the following table

PWxS2	PWxS1	State	Period Size	Counts
0	0	Disabled	—	—
0	1	Enabled	8 Bits	256
1	0	Enabled	10 Bits	1024
1	1	Enabled	12 Bits	4096

PWTDY5 — Pulse-Width Channel 5 Duty **$0052, $0053**

$0052	0	0	0	Bit 12	11	10	9	Bit 8	PWTDY5 High
$0053	Bit 7	6	5	4	3	2	1	Bit 0	PWTDY5 Low
RESET:	0	0	0	1	1	1	1	1	

Resets to $1FFF.

PWTDY6 — Pulse-Width Channel 6 Duty **$0054, $0055**

$0054	0	0	0	Bit 12	11	10	9	Bit 8	PWTDY6 High
$0055	Bit 7	6	5	4	3	2	1	Bit 0	PWTDY6 Low
RESET:	1	1	1	1	1	1	1	1	

Resets to $1FFF.

PWCNT5 — Pulse-Width Channel 5 and 6 Counter **$0056, $0057**

$0056	0	0	0	0	Bit 11	10	9	Bit 8	PWCNT5 High
$0057	Bit 7	6	5	4	3	2	1	Bit 0	PWCNT5 Low
RESET:	0	0	0	0	0	0	0	0	

Resets to $0000.

PWCLK — Pulse-Width Modulation Clock Select **$0060**

	Bit 7	6	5	4	3	2	1	Bit 0
	CON34	CON12	PCKA2	PCKA1	0	PCKB3	PCKB2	PCKB1
RESET:	0	0	0	0	0	0	0	0

CON34 — Concatenate Channels 3 and 4

Channel 3 is high-order byte, and channel 4 (port H, bit 3) is output. Clock source is determined by PCLK4.

0 = Channels 3 and 4 are separate 8-bit PWMs.

1 = Channels 3 and 4 concatenated to create one 16-bit PWM channel.

CON12 — Concatenate Channels One and Two

Channel 1 is high order byte, and channel 2 (port H, bit 1) is output. Clock source is determined by PCLK2.

0 = Channels 1 and 2 are separate 8-bit PWMs

1 = Channels 1 and 2 are concatenated to create one 16-bit PWM channel.

PCKA2–PCKA1 — Prescaler for Clock A (See also PWSCAL register)

Determines the rate of clock A

PCKA[2:1]	Value of Clock A
0 0	E
0 1	E/2
1 0	E/4
1 1	E/8

Bit 3— Not implemented

Always reads zero

PCKB3–PCKB1 — Prescaler for Clock B

Determines the rate for clock B

PCKB[3:1]	Value of Clock B
0 0 0	E
0 0 1	E/2
0 1 0	E/4
0 1 1	E/8
1 0 0	E/16
1 0 1	E/32
1 1 0	E/64
1 1 1	E/128

PWPOL — Pulse-Width Modulation Timer Polarity **$0061**

	Bit 7	6	5	4	3	2	1	Bit 0
	PCLK4	PCLK3	PCLK2	PCLK1	PPOL4	PPOL3	PPOL2	PPOL1
RESET:	0	0	0	0	0	0	0	0

PCLK4 — Pulse-Width Channel 4 Clock Select
0 = Clock B is source
1 = Clock S is source

PCLK3 — Pulse-Width Channel 3 Clock Select
0 = Clock B is source
1 = Clock S is source

PCLK2 — Pulse-Width Channel 2 Clock Select
0 = Clock A is source
1 = Clock S is source

PCLK1 — Pulse-Width Channel 1 Clock Select
0 = Clock A is source
1 = Clock S is source

PPOL[4:1] — Pulse-Width Channel x Polarity
0 = PWM channel x output is low at the beginning of the clock cycle and goes high when duty count is reached
1 = PWM channel x output is high at the beginning of the clock cycle and goes low when duty count is reached

PWSCAL — Pulse-Width Modulation Timer Prescaler **$0062**

	Bit 7	6	5	4	3	2	1	Bit 0
	7	6	5	4	3	2	1	0
RESET:	0	0	0	0	0	0	0	0

Scaled clock S is generated by dividing clock A by the value in PWSCAL, then dividing the result by 2. If PWSCAL = $00, divide clock A by 256, then divide the result by 2.

PWEN — Pulse-Width Modulation Timer Enable **$0063**

	Bit 7	6	5	4	3	2	1	Bit 0
	TPWSL	DISCP	0	0	PWEN4	PWEN3	PWEN2	PWEN1
RESET:	0	0	0	0	0	0	0	0

TPWSL — PWM Scaled Clock Test Bit (TEST)

DISCP — Disable Compare Scaled E Clock (TEST)

Bits [5:4] — Not implemented
Always read zero

PWEN[4:1] — Pulse-Width Channel 4–1
0 = Channel disabled
1 = Channel enabled

PWCNT1–4 — Pulse-Width Modulation Timer Counter 1 to 4 **$0064–$0067**

	Bit 7	6	5	4	3	2	1	Bit 0	
$0064	Bit 7	6	5	4	3	2	1	Bit 0	PWCNT1
$0065	Bit 7	6	5	4	3	2	1	Bit 0	PWCNT2
$0066	Bit 7	6	5	4	3	2	1	Bit 0	PWCNT3
$0067	Bit 7	6	5	4	3	2	1	Bit 0	PWCNT4
RESET:	0	0	0	0	0	0	0	0	

PWCNT1–4
Begins count using whichever clock was selected

PWPER1–4 — Pulse-Width Modulation Timer Period 1 to 4 **$0068–$006B**

	Bit 7	6	5	4	3	2	1	Bit 0	
$0068	Bit 7	6	5	4	3	2	1	Bit 0	PWPER1
$0069	Bit 7	6	5	4	3	2	1	Bit 0	PWPER2
$006A	Bit 7	6	5	4	3	2	1	Bit 0	PWPER3
$006B	Bit 7	6	5	4	3	2	1	Bit 0	PWPER4
RESET:	1	1	1	1	1	1	1	1	

PWPER1–4
Determines period of associated PWM channel

PWDTY1–4 — Pulse-Width Modulation Timer Duty Cycle 1 to 4 **$006C–$006F**

	Bit 7	6	5	4	3	2	1	Bit 0	
$006C	Bit 7	6	5	4	3	2	1	Bit 0	PWDTY1
$006D	Bit 7	6	5	4	3	2	1	Bit 0	PWDTY2
$006E	Bit 7	6	5	4	3	2	1	Bit 0	PWDTY3
$006F	Bit 7	6	5	4	3	2	1	Bit 0	PWDTY4
RESET:	1	1	1	1	1	1	1	1	

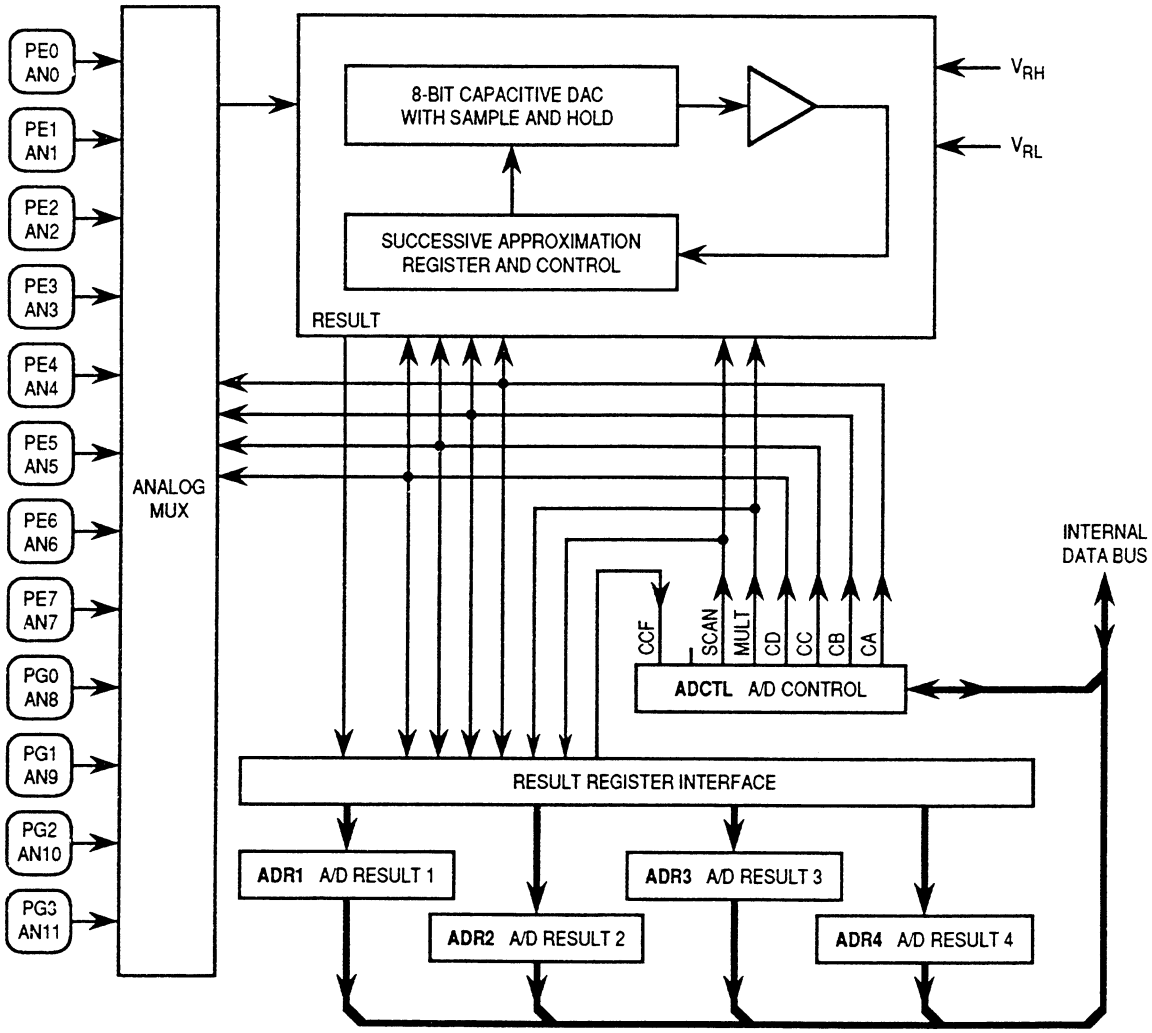
PWDTY1–4

Determines duty cycle of associated PWM channel

Analog-to-Digital Converter

The analog-to-digital (A/D) converter system uses an all-capacitive charge-redistribution technique to convert analog signals to digital values. The MC68HC11N4 A/D converter system is an 8-channel, 8-bit, multiplexed-input, successive-approximation converter. It does not require external sample and hold circuits.

The clock source for the A/D converter's charge pump, like the clock source for the EEPROM charge pump, is selected with the CSEL bit in the OPTION register. When the E clock is slower than 1 MHz, the CSEL bit must be set to ensure that the successive approximation sequence for the A/D converter will be completed before any charge loss occurs.



A/D Converter Block Diagram

The A/D converter can operate in single- or multiple-conversion modes. Multiple conversions are performed in sequences of four. Sequences can be performed on a single channel or on a group of channels.

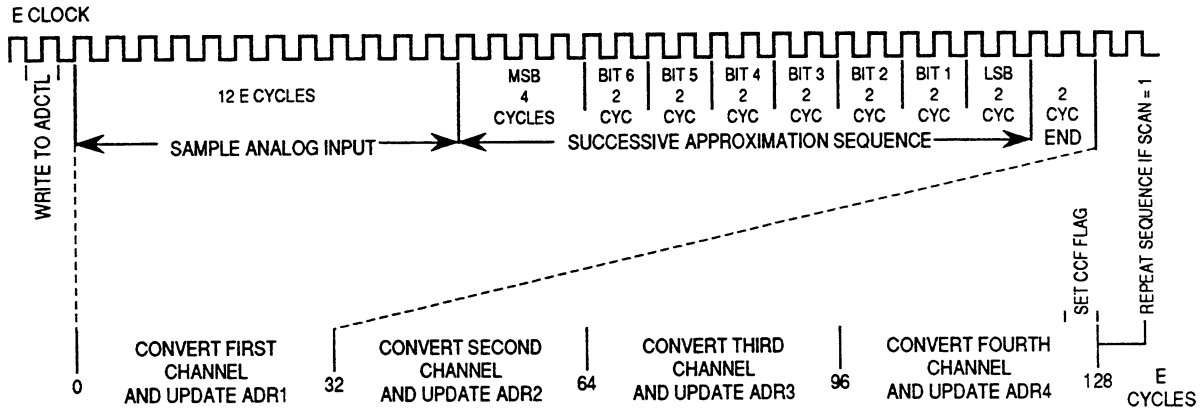
Dedicated lines V_{RH} and V_{RL} provide the reference supply voltage inputs.

A multiplexer allows the single A/D converter to select one of 16 analog input signals.

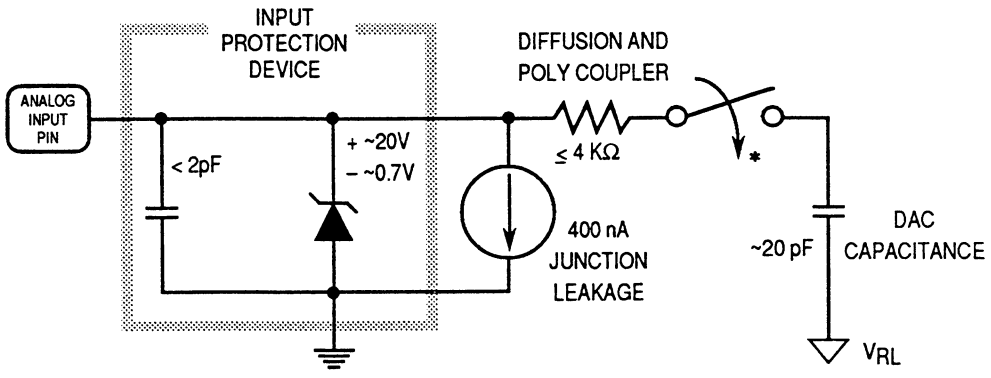
The A/D converter control logic implements automatic conversion sequences on a selected channel four times or on four channels once each. A write to the ADCTL register initiates conversions and, if made while a conversion is in process, a write to ADCTL also halts a conversion operation in progress.

When the SCAN bit is zero, four requested conversions are performed, once each, to fill the four result registers. When SCAN is one, conversions continue in a round-robin fashion with the result registers being updated as new data becomes available.

When the MULT bit is zero, the A/D converter system is configured to perform four consecutive conversions on the single channel specified by the four channel-select bits (CD–CA). When the MULT bit is one, the A/D system is configured to perform conversions on each channel in the group of four channels specified by the CD and CC channel select bits.



Timing Diagram for a Sequence of Four A/D Conversions



*THIS ANALOG SWITCH IS CLOSED ONLY DURING THE 12-CYCLE SAMPLE TIME.

Electrical Model of an Analog Input Pin (Sample Mode)

ADCTL — A/D Control/Status **$0030**

	Bit 7	6	5	4	3	2	1	Bit 0
	CCF	0	SCAN	MULT	CD	CC	CB	CA
RESET:	0	0	0	0	0	0	0	0

CCF — Conversions Complete Flag
CCF is set after an A/D conversion cycle. This bit is cleared when ADCTL is written.

Bit 6 — Not implemented
Always reads zero

SCAN — Continuous Scan Control
0 = Do four conversions and stop
1 = Convert four channels in selected group continuously

MULT — Multiple Channel/Single Channel Control
0 = Convert single channel selected
1 = Convert four channels in selected group

CD–CA — Channel Select D through A

A/D Converter Channel Assignments

Channel Select Control Bits				Channel Signal	Result in ADRx if MULT = 1
CD	CC	CB	CA		
0	0	0	0	AN0	ADR1
0	0	0	1	AN1	ADR2
0	0	1	0	AN2	ADR3
0	0	1	1	AN3	ADR4
0	1	0	0	AN4	ADR1
0	1	0	1	AN5	ADR2
0	1	1	0	AN6	ADR3
0	1	1	1	AN7	ADR4
1	0	0	0	AN8	ADR1
1	0	0	1	AN9	ADR2
1	0	1	0	AN10	ADR3
1	0	1	1	AN11	ADR4
1	1	0	0	V_{RH}*	ADR1
1	1	0	1	V_{RL}*	ADR2
1	1	1	0	$(V_{RH})/2$*	ADR3
1	1	1	1	Reserved*	ADR4

*Used for factory testing

ADR[4:1] — A/D Results **$0031–$0034**

$0031	Bit 7	6	5	4	3	2	1	Bit 0	ADR1
$0032	Bit 7	6	5	4	3	2	1	Bit 0	ADR2
$0033	Bit 7	6	5	4	3	2	1	Bit 0	ADR3
$0034	Bit 7	6	5	4	3	2	1	Bit 0	ADR4

OPTION — System Configuration Options **$0039**

	Bit 7	6	5	4	3	2	1	Bit 0
	ADPU	CSEL	IRQE*	DLY*	CME	FCME*	CR1*	CR0*
RESET:	0	0	0	1	0	0	0	0

*Can be written only once in first 64 cycles out of reset in normal modes, any time in special mode.

ADPU — A/D Converter Power-Up
0 = A/D converter powered down
1 = A/D converter powered up

CSEL — Clock Select for A/D and EEPROM
Refer to **Electrically Erasable Programmable Read-Only Memory (EEPROM)**.

IRQE— IRQ Select Edge Sensitive Only
Refer to **Resets and Interrupts**.

DLY — Enable Oscillator Startup Delay on Exit from Stop
Refer to **Resets and Interrupts**.

CME — Clock Monitor Enable
Refer to **Resets and Interrupts**.

FCME — Force Clock Monitor Enable
Refer to **Resets and Interrupts**.

CR1, CR0 — COP Timer Rate Select
Refer to **Main Timer**.

Digital-to-Analog Converter

The two D/A channels on the MC68HC11N4 provide an analog reference output for specialized applications. Each channel creates an analog voltage level based on a digital value between $00 and $FF. The D/A converter consists of two data registers and a control register. Each D/A converter channel generates an analog voltage in even steps between V_{SS} and V_{DD}, based on digital values in the associated data register. During stop mode, D/A outputs become open to prevent additional current drain.

DACON — D/A Converter Control **$004D**

	Bit 7	6	5	4	3	2	1	Bit 0
	0	0	0	0	0	0	DAE2	DAE1
RESET:	0	0	0	0	0	0	0	0

Bits [7:2] — Not implemented
Always read zero

DAE2 — D/A Channel 2 Enable
0 = D/A channel 2 disabled, port G bit 5 is general-purpose input
1 = Analog voltage based on digital value in DA2 output to port G bit 5

DAE1 — D/A Channel 1 Enable
0 = D/A channel 1 disabled, port G bit 4 is general-purpose input
1 = Analog voltage based on digital value in DA1 output to port G bit 4

NOTE

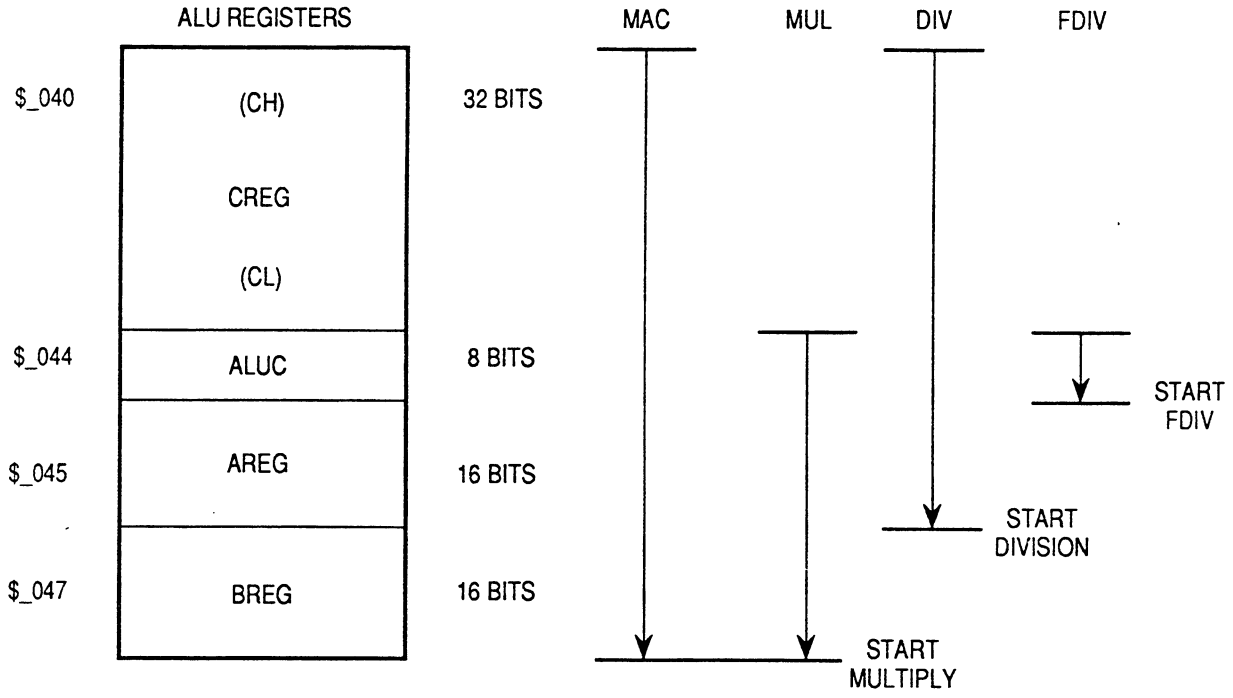
During STOP mode the D/A outputs become open to prevent additional current drain.

DA2, DA1 — D/A Converter Data **$004E, $004F**

	Bit 7	6	5	4	3	2	1	Bit 0
	Bit 7	6	5	4	3	2	1	Bit 0
RESET:	0	0	0	0	0	0	0	0

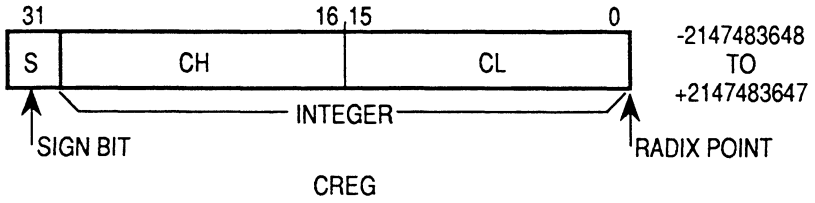
Math Coprocessor

The MC68HC11N4 arithmetic logic unit performs fast 16-bit integer multiplication and division as a coprocessor. It performs signed or unsigned multiplication, with or without accumulated product, as well as signed or unsigned fast division. Because the arithmetic operations are executed independently, the CPU is free to perform other operations. The five registers of the coprocessor are mapped in the peripheral register area and can be read from or written to by the CPU. Refer to the following diagram for the coprocessor structure.

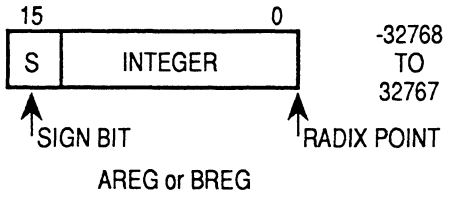


Coprocessor Map

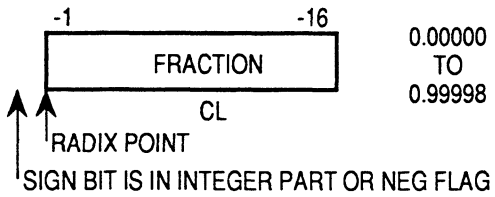
32-BIT SIGNED INTEGER NUMBER



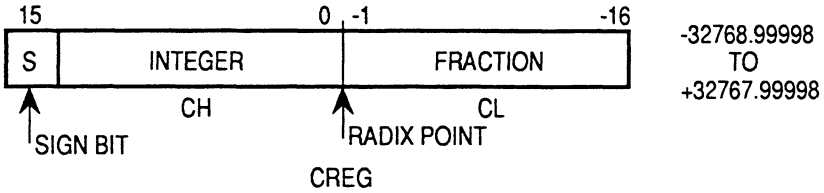
16-BIT SIGNED INTEGER NUMBER



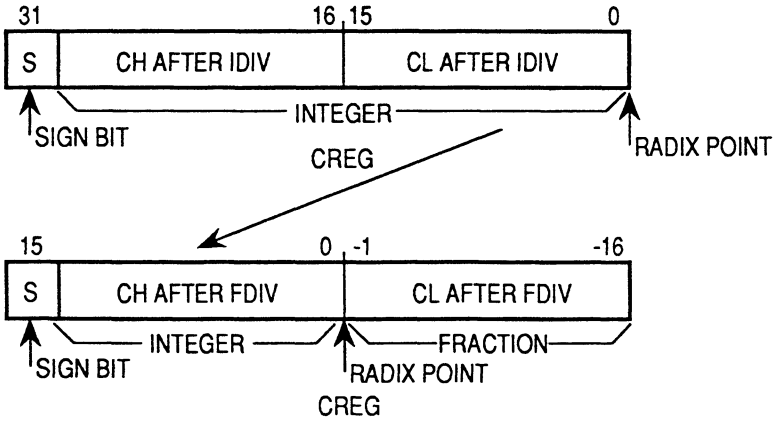
16-BIT FRACTION NUMBER AFTER FDIV



32-BIT SIGNED INTEGER AND FRACTION FOLLOWING AN FDIV



LONG WORD SIGNED RESULT AFTER FDIV FOLLOWING AN IDIV



ALU Operations

The following data registers hold the signed or unsigned integers. There is an implied fixed radix point at the right of bit 0. AREG holds the value of the multiplicand or the divisor. BREG holds the multiplier or remainder after division. CREG, considered two 16-bit registers (CH and CL), holds

the product or accumulated product after multiplication, or the numerator before division and the quotient after division.

During fractional division, the fractional number in CL does not include the sign bit. The quotient is placed in CL and the remainder is placed in BREG. The previous contents of CL are moved into CH after the fractional division. The signed data that is used for both input or output is twos complement binary integer format.

Control of the coprocessor is provided by the ALUC register. The ALUF register indicates the status of the operation just performed.

CREG — Data Register C **$0040–$0043**

$0040	Bit 31	30	29	28	27	26	25	Bit 24	CREG (High)
$0041	Bit 23	22	21	20	19	18	17	Bit 16	CREG (Mid-High)
$0042	Bit 15	14	13	12	11	10	9	Bit 8	CREG (Mid-Low)
$0043	Bit 7	6	5	4	3	2	1	Bit 0	CREG (Low)

CREG reset value is indeterminate.

ALUC — Arithmetic Logic Unit Control **$0044**

	Bit 7	6	5	4	3	2	1	Bit 0
	SIG	DIV	MAC	DCC	TRG	—	—	—
RESET:	0	0	0	0	0	0	0	0

SIG — Signed Number Enable
0 = AREG, BREG, and CREG contents are unsigned numbers
1 = AREG, BREG, and CREG contents are signed numbers

DIV — Division Enable
MAC — Multiply with Accumulated Product Enable
DCC — Division Compensation for Concatenated Quotient Enable

SIG	DIV	MAC	DCC	FUNCTION	START TRIGGERS
0	0	0	X	Unsigned MUL	Write BREG or set TRG
1	0	0	X	Signed MUL	Write BREG or set TRG
0	0	1	X	Unsigned MAC	Write BREG or set TRG
1	0	1	X	Signed MAC	Write BREG or set TRG
0	1	0	X	Unsigned IDIV	Write AREG or set TRG
1	1	0	0	Signed IDIV	Write AREG or set TRG
1	1	0	1	Signed IDIV DCC	Write AREG or set TRG
0	1	1	X	Unsigned FDIV	Set TRG
1	1	1	0	Signed FDIV	Set TRG
1	1	1	1	Signed FDIV DCC	Set TRG

TRG — Function Start Trigger Bit
Always reads zero.
0 = No effect
1 = Writing this bit to one starts the function

Bits [2:0] — Not implemented
Always read zero

AREG — Data Register A **$0045, $0046**

	Bit 7	6	5	4	3	2	1	Bit 0	
$0045	Bit 15	14	13	12	11	10	9	Bit 8	AREG (High)
$0046	Bit 7	6	5	4	3	2	1	Bit 0	AREG (Low)

AREG reset value is indeterminate.

BREG — Data Register B **$0047, $0048**

	Bit 7	6	5	4	3	2	1	Bit 0	
$0047	Bit 15	14	13	12	11	10	9	Bit 8	BREG (High)
$0048	Bit 7	6	5	4	3	2	1	Bit 0	BREG (Low)

BREG reset value is indeterminate.

ALUF — Arithmetic Logic Unit Status Flag **$0049**

	Bit 7	6	5	4	3	2	1	Bit 0
	NEG	RZF	0	0	0	OVF	DZF	ACF
RESET:	0	0	0	0	0	0	0	0

NEG — Negative Result
NEG is set if the result is a negative value. This is a read-only bit. Writes to this bit do not affect the value.

RZF — Remainder Equals Zero Flag
RZF is set if the remainder is zero.

Bits [5:3] — Not implemented
Always read zero

OVF — Overflow Flag
OVF is set if overflow from MSB on CREG is detected. Cleared automatically by a write to this register with bit 2 set.

DZF — Divide by Zero Flag

DZF is set if a divide by zero condition is detected. DZF is cleared automatically by a write to this register with bit 1 set.

ACF — Arithmetic Completion Flag

ACF is set by completion of the arithmetic operation. ACF is cleared automatically by a write to this register with bit 0 set.

Home Page:
www.freescale.com
email:
support@freescale.com
USA/Europe or Locations Not Listed:
Freescale Semiconductor
Technical Information Center, CH370
1300 N. Alma School Road
Chandler, Arizona 85224
(800) 521-6274
480-768-2130
support@freescale.com
Europe, Middle East, and Africa:
Freescale Halbleiter Deutschland GmbH
Technical Information Center
Schatzbogen 7
81829 Muenchen, Germany
+44 1296 380 456 (English)
+46 8 52200080 (English)
+49 89 92103 559 (German)
+33 1 69 35 48 48 (French)
support@freescale.com
Japan:
Freescale Semiconductor Japan Ltd.
Headquarters
ARCO Tower 15F
1-8-1, Shimo-Meguro, Meguro-ku
Tokyo 153-0064, Japan
0120 191014
+81 2666 8080
support.japan@freescale.com
Asia/Pacific:
Freescale Semiconductor Hong Kong Ltd.
Technical Information Center
2 Dai King Street
Tai Po Industrial Estate,
Tai Po, N.T., Hong Kong
+800 2666 8080
support.asia@freescale.com
For Literature Requests Only:
Freescale Semiconductor
Literature Distribution Center
P.O. Box 5405
Denver, Colorado 80217
(800) 441-2447
303-675-2140
Fax: 303-675-2150
LDCForFreescaleSemiconductor
@hibbertgroup.com

RoHS-compliant and/or Pb- free versions of Freescale products have the functionality and electrical characteristics of their non-RoHS-compliant and/or non-Pb- free counterparts. For further information, see http://www.freescale.com or contact your Freescale sales representative.

For information on Freescale.s Environmental Products program, go to http://www.freescale.com/epp.

MC68HC11N4TS/D
1ATX31108-1 PRINTED IN USA 11/97 IMPERIAL LITHO 30,125 530 MTG902