Welcome to E-XFL.COM

What is "Embedded - Microcontrollers"?

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

Applications of "Embedded - Microcontrollers"

Details	
Product Status	Obsolete
Core Processor	HC11
Core Size	8-Bit
Speed	4MHz
Connectivity	SCI, SPI
Peripherals	POR, PWM, WDT
Number of I/O	51
Program Memory Size	32KB (32K x 8)
Program Memory Type	OTP
EEPROM Size	640 x 8
RAM Size	1K x 8
Voltage - Supply (Vcc/Vdd)	4.5V ~ 5.5V
Data Converters	A/D 8x8b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 125°C (TA)
Mounting Type	Surface Mount
Package / Case	68-LCC (J-Lead)
Supplier Device Package	68-PLCC (24.21x24.21)
Purchase URL	https://www.e-xfl.com/product-detail/nxp-semiconductors/mchc711ks2mfne4

1.3 M68HC11K Family Members

M68HC11K Family devices feature up to 62 input/output (I/O) lines distributed among eight ports, A through H. The KS Family removes seven pins from port G and four pins from port H for a total of 51 I/O lines. The KSx versions feature a slow mode for the clocks to allow power conservation. **Table 1-1** lists devices currently available in the K Family along with their distinguishing features.

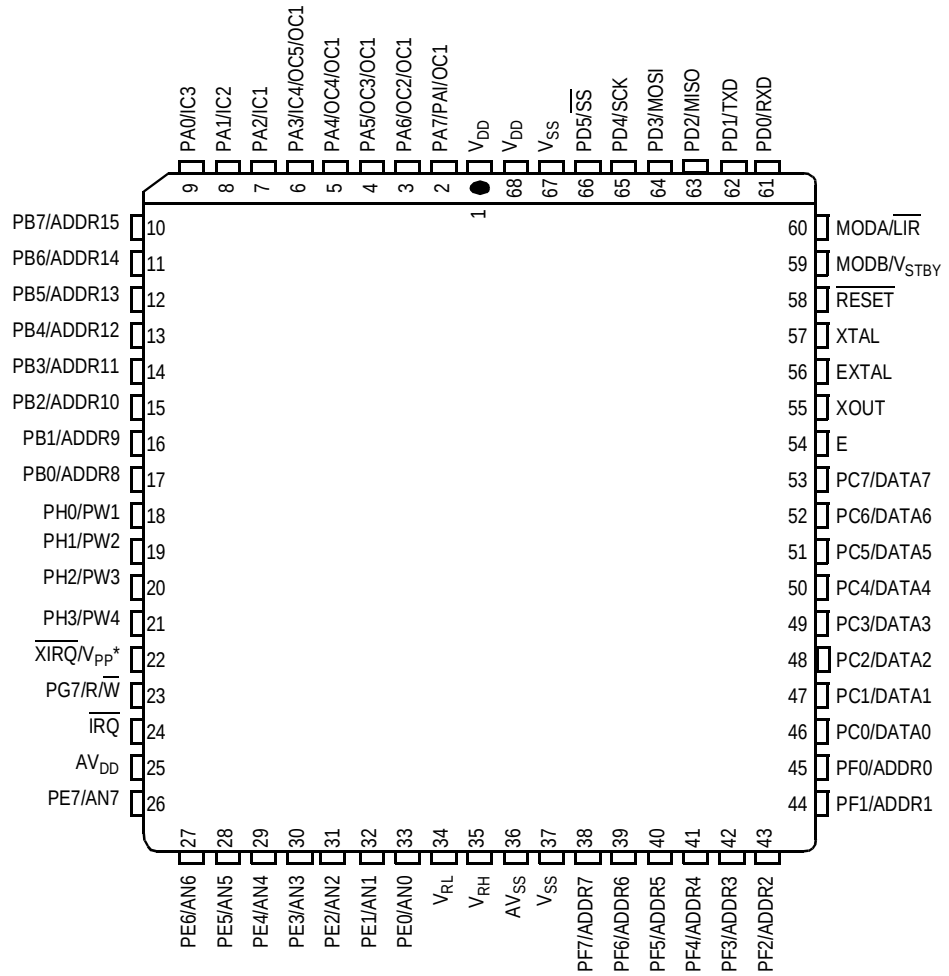
NOTE: *The KA2 and KA4 devices have been replaced by the pin-for-pin compatible KS2.*

Table 1-1. M68HC11K Family Devices

Device Number	ROM or *EPROM* (Bytes)(1)	RAM (Bytes)	EEPROM (Bytes)	I/O (Pins)	Chip Select	Slow Mode	Packages
MC68HC(L)11K0 MC68HC(L)11K1 MC68HC(L)11K4	0 0 24 K	768 768 768	0 640 640	37 37 62	Yes Yes Yes	No No No	84-pin PLCC(2) 80-pin QFP(3)
MC68HC711K4	*24 K*	768	640	62	Yes	No	84-pin J-cerquad(4) 84-pin PLCC 80-pin QFP
MC68HC11KS2	32 K	1 K	640	51	No	Yes	68-pin PLCC and 80-pin LQFP(5)
MC68HC711KS2	*32 K*	1 K	640	51	No	Yes	68-pin J-cerquad, 68-pin PLCC, and 80-pin LQFP

1. Where applicable, EPROM bytes appear in italics.
2. PLCC = Plastic leaded chip carrier
3. QFP = Quad flat pack
4. J-cerquad = Ceramic windowed version of PLCC
5. LQFP = Low-profile quad flat pack

Pin Description



* V_{PP} applies only to EPROM devices.

Figure 2-3. Pin Assignments for M6811KS 68-Pin PLCC/J-Cerquad

In single-chip and bootstrap modes, the MODA pin typically is grounded and has no function after reset. In expanded and special test modes, MODA is normally connected to V_{DD} through a 4.7-kΩ pullup resistor and functions as the load instruction register ($\overline{LIR}$) pin after reset. The open-drain, active-low $\overline{LIR}$ output drives low during the first E-clock cycle of each instruction (opcode fetch), providing a useful signal for system debugging.

$\overline{LIR}$ can be driven high for a portion of each instruction cycle by setting the LIRDV bit in the system configuration options 2 (OPT2) register (see **Figure 2-7** and **Figure 2-8**). This feature can help detect consecutive instructions and prevent false triggering in high-speed applications.

Address: $0038

	Bit 7	6	5	4	3	2	1	Bit 0
Read: Write:	LIRDV	CWOM	STRCH(1)	IRVNE	LSBF	SPR2	XDV1	XDV0
Reset:	0	0	0	—	0	0	0	0

1. STRCH is not available on K devices.

Figure 2-7. System Configuration Options 2 (OPT2)

LIRDV — $\overline{LIR}$ Driven Bit

0 = $\overline{LIR}$ not driven high

1 = $\overline{LIR}$ driven high for one quarter cycle to reduce transition time

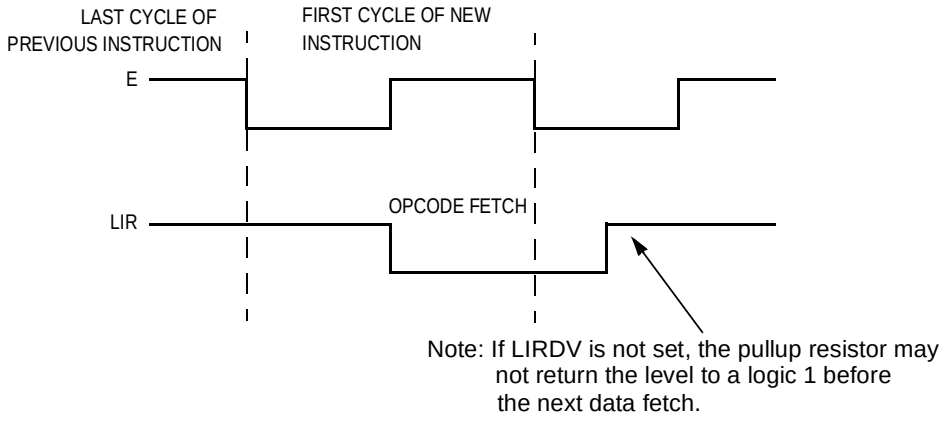


Figure 2-8. $\overline{LIR}$ Timing

3.3.1 Accumulators A, B, and D (ACCA, ACCB, and ACCD)

Accumulators A and B are general-purpose 8-bit registers that hold operands and results of arithmetic calculations or data manipulations. Some instructions treat these two accumulators as a single double-byte (16-bit) accumulator called accumulator D. Most operations can use either accumulator A or B, with these exceptions:

- The ABX and ABY instructions add the contents of 8-bit accumulator B to the contents of 16-bit register X or Y, but there are no equivalent instructions that use A instead of B.
- The TAP and TPA instructions transfer data from accumulator A to the condition code register or from the condition code register to accumulator A. However, there are no equivalent instructions that use B rather than A.
- The DAA instruction adjusts accumulator A after binary-coded decimal (BCD) arithmetic operations, but there is no equivalent BCD instruction to adjust accumulator B.
- The add, subtract, and compare instructions associated with both A and B (ABA, SBA, and CBA) only operate in one direction, making planning ahead important to ensure the correct operand is in the correct accumulator.

3.3.2 Index Register X (IX)

The IX register provides a 16-bit indexing value that can be added to the 8-bit offset provided in an instruction to create an effective address. The IX register can be used also as a counter or as a temporary storage register.

3.3.3 Index Register Y (IY)

The IY register provides a 16-bit indexed mode function similar to that of the IX register. Instructions using the IY register require an extra byte of machine code and an extra cycle of execution time because of the way the opcode map is implemented.

Table 4-5. RAM Mapping

RAM[3:0]	Address(1)	Address(2)
0000	$0080–$037F(3)	$0000–$02FF
0001	$1080–$137F	$1000–$12FF
0010	$2080–$237F	$2000–$22FF
0011	$3080–$337F	$3000–$32FF
0100	$4080–$437F	$4000–$42FF
0101	$5080–$537F	$5000–$52FF
0110	$6080–$637F	$6000–$62FF
0111	$7080–$737F	$7000–$72FF
1000	$8080–$837F	$8000–$82FF
1001	$9080–$937F	$9000–$92FF
1010	$A080–$A37F	$A000–$A2FF
1011	$B080–$B37F	$B000–$B2FF
1100	$C080–$C37F	$C000–$C2FF
1101	$D080–$D37F	$D000–$D2FF
1110	$E080–$E37F	$E000–$E2FF
1111	$F080–$F37F	$F000–$F2FF

1. RAM[3:0] = REG[3:0]: On the [7]11KS2, RAM address range is $x080–$x47F.
2. RAM[3:0] ≠ REG[3:0]: On the [7]11KS2, RAM address range is $x000–$x37F.
3. Default locations out of reset

Address: $003F

	Bit 7	6	5	4	3	2	1	Bit 0
Read: Write:	ROMAD	1	CLKX	PAREN	NOSEC	NOCOP	ROMON	EEON
Reset:	—	1	—	—	—	—	—	—

Figure 4-6. System Configuration Register (CONFIG)

NOTE: *CONFIG is writable once in normal modes and writable at any time in special modes.*

ROMAD — ROM Address Mapping Control Bit

Set out of reset in single-chip mode

0 = (EP)ROM set at $2000–$7FFF;
$0000–$7FFF in [7]11KS2;
$0000–$BFFF in [7]11KS8
(expanded mode only)

1 = (EP)ROM set at $A000–$FFFF;
$8000–$FFFF in [7]11KS2;
$4000–$FFFF in [7]11KS8

ROMON — ROM/PROM Enable Bit

Set by reset in single-chip mode; cleared by reset in special test mode

0 = (EP)ROM removed from the memory map
1 = (EP)ROM present in the memory map

EEON — EEPROM Enable Bit

0 = 640-byte EEPROM disabled
1 = 640-byte EEPROM enabled

Table 5-5. Interrupt and Reset Vector Assignments

Vector Address	Interrupt Source	CC Register Mask	Local Mask
FFC0, C1 — FFD4, D5	Reserved	—	—
FFD6, D7	SCI serial system: • SCI transmit complete • SCI transmit data register empty • SCI idle line detect • SCI receiver overrun • SCI receive data register full	I bit	 TCIE TIE ILIE RIE RIE
FFD8, D9	SPI serial transfer complete	I bit	SPIE
FFDA, DB	Pulse accumulator input edge	I bit	PAII
FFDC, DD	Pulse accumulator overflow	I bit	PAOVI
FFDE, DF	Timer overflow	I bit	TOI
FFE0, E1	Timer input capture 4/output compare 5	I bit	I4/O5I
FFE2, E3	Timer output compare 4	I bit	OC4I
FFE4, E5	Timer output compare 3	I bit	OC3I
FFE6, E7	Timer output compare 2	I bit	OC2I
FFE8, E9	Timer output compare 1	I bit	OC1I
FFEA, EB	Timer input capture 3	I bit	IC3I
FFEC, ED	Timer input capture 2	I bit	IC2I
FFEE, EF	Timer input capture 1	I bit	IC1I
FFF0, F1	Real-time interrupt	I bit	RTII
FFF2, F3	$\overline{\text{IRQ}}$ (external pin)	I bit	None
FFF4, F5	$\overline{\text{XIRQ}}$ pin	X bit	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	RESET	None	None

Many interrupt sources set associated flag bits when interrupts occur. These flags are usually cleared during the course of normal interrupt service. For example, the normal response to an RDRF interrupt request in the SCI is to read the SCI status register to check for receive errors, then read the received data from the SCI data register. It is precisely these two steps which clear RDRF, so no extra steps are required.

An interrupt can be recognized at any time after it is enabled by its local mask, if any, and by the global mask bit in the CCR. The CPU responds to an interrupt at the completion of the instruction being executed. Since the number of clock cycles in the instruction varies, so does interrupt latency. The CPU pushes the contents of its registers onto the stack in the order shown in **Table 5-6**. After the CCR value is stacked, the I bit is set (and the X bit as well if $\overline{\text{XIRQ}}$ is pending) to inhibit further interrupts. The CPU fetches the interrupt vector for the highest priority pending source, and execution continues at the address specified by the vector. The interrupt service routine ends with the return-from-interrupt (RTI) instruction, which tells the CPU to pull the saved registers from the stack in reverse order so that normal program execution can resume.

Table 5-6. Stacking Order on Entry to Interrupts

Memory Location	CPU Registers
SP	PCL
SP – 1	PCH
SP –2	IYL
SP – 3	IYH
SP – 4	IXL
SP – 5	IXH
SP – 6	ACCA
SP – 7	ACCB
SP – 8	CCR

execution of the illegal opcode, which can lead to stack overflow, the service routine should reinitialize the stack pointer.

5.5.1.3 Software Interrupt (SWI)

SWI cannot be masked by virtue of the fact that it is a software instruction. It is not inhibited by the global mask bits in the CCR. Execution of SWI sets the I mask bit, so other interrupts are inhibited until user software clears the I bit or SWI terminates with an RTI instruction.

5.5.2 Maskable Interrupts

All maskable interrupts are generated by on-chip peripherals, with the exception of the $\overline{\text{IRQ}}$ pin. This input can be connected through a wired-OR network to external devices. When one of these devices pulls $\overline{\text{IRQ}}$ low, a software accessible interrupt flag is set. When enabled, this flag causes a constant request for interrupt service. After the flag is cleared, the service request is released. $\overline{\text{IRQ}}$ is low-level sensitive by default, but can be set for falling-edge sensitivity by the IRQE bit in the OPTION register (see **Figure 5-6**).

Address: $0039

	Bit 7	6	5	4	3	2	1	Bit 0
Read: Write:	ADPU	CSEL	IRQE(1)	DLY(1)	CME	FCME(1)	CR1(1)	CR0(1)
Reset:	0	0	0	1	0	0	0	0

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

Figure 5-6. System Configuration Options Register (OPTION)

IRQE — Configure IRQ for Edge-Sensitive Operation Bit

This bit can be written only once during the first 64 E-clock cycles after reset in normal modes.

0 = Low-level recognition
1 = Falling-edge recognition

The M68HC11K series offers several enhancements to the basic MC68HC11 SCI, including:

- 13-bit modulus prescaler in the baud generator
- Receiver-active flag
- Transmitter and receiver hardware parity
- Accelerated idle line detection

7.3 Data Format

The SCI uses the standard non-return to zero mark/space data format illustrated in **Figure 7-1**.

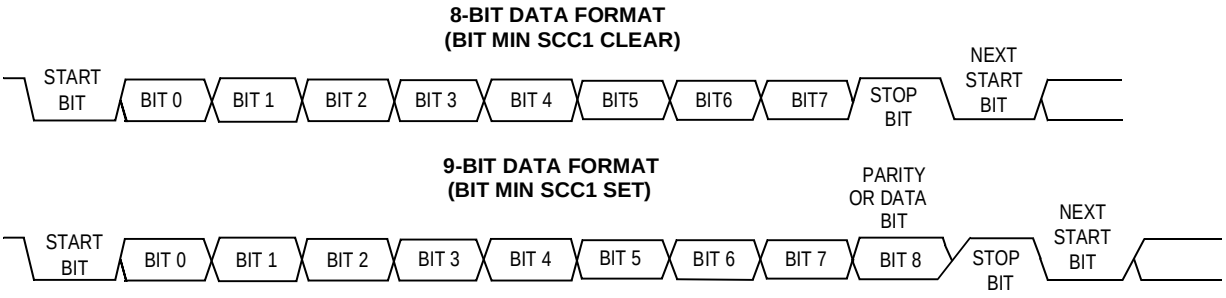


Figure 7-1. SCI Data Formats

Data is transmitted in frames consisting of a start bit, a word of eight or nine data bits, and a stop bit. The step-by-step transmission procedure is:

1. The transmission line is idle before a message is transmitted. This means that the line is in a logic 1 state for at least one frame time.
2. A start bit, logic 0, is transmitted, indicating the start of a frame.
3. An 8-bit or 9-bit word is transmitted, least significant bit (LSB) first.
4. A stop bit, logic 1, is transmitted to indicate the end of a frame.
5. An optional number of breaks can be transmitted. A break is the transmission of a logic low state for one frame time. After the last break character is sent, the line goes high for at least one bit time.

A single MCU register, the serial peripheral data register (SPDR) is used both to read input data from the read buffer and to write output data to the transmit shift register.

Figure 8-1 shows the SPI block diagram.

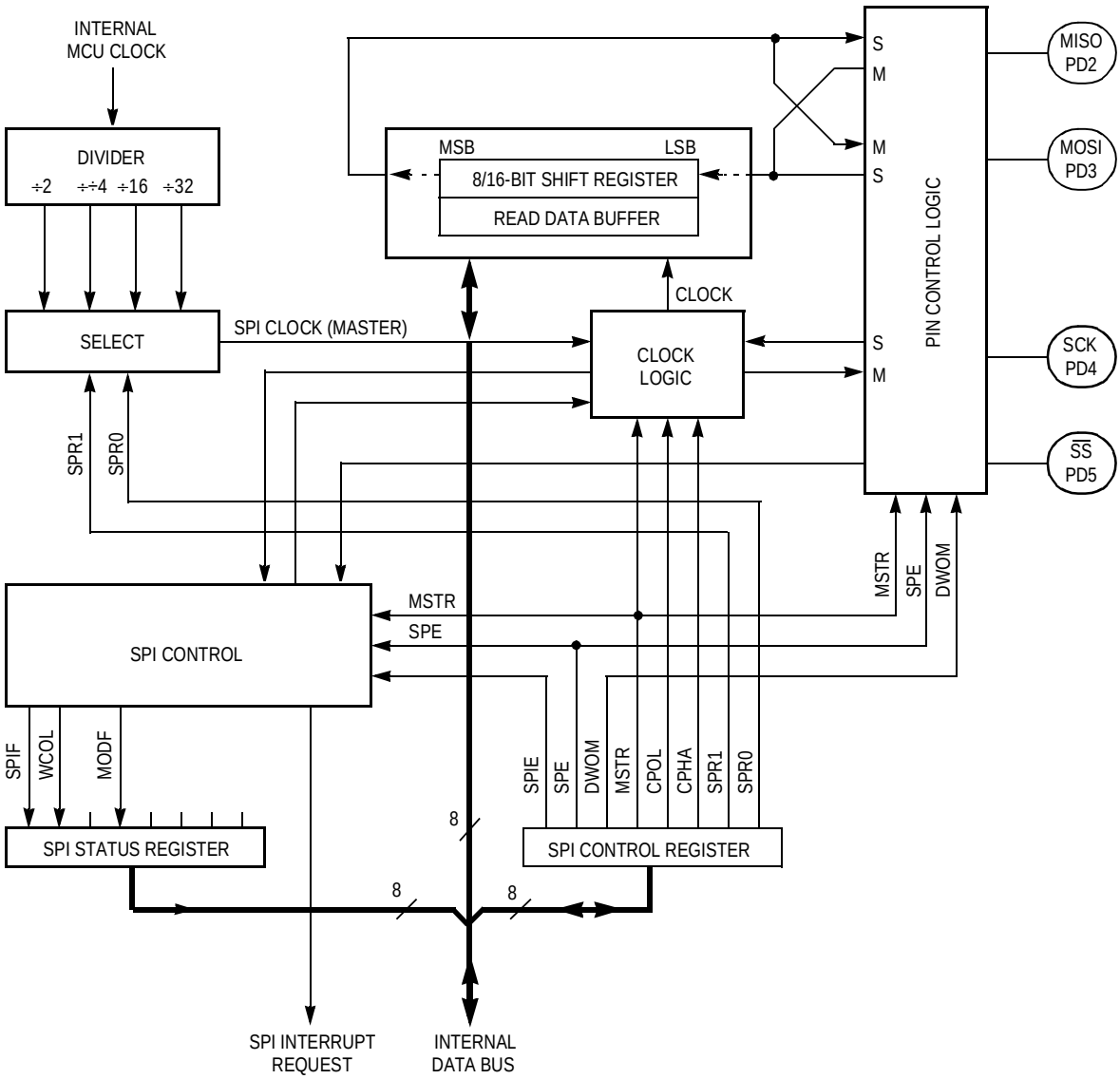


Figure 8-1. SPI Block Diagram

8.6.4 Port D Data Direction Register

Address: $0009

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

Figure 8-6. Port D Data Direction Register (DDRD)

DDD5 Bit

Bit 5 of the port D data register (PD5) is dedicated as the slave select ($\overline{SS}$) input. In SPI slave mode, DDD5 has no meaning or effect. In SPI master mode, DDD5 affects PD5 as follows:

0 = PD5 is an error-detect input to the SPI.
1 = PD5 is configured as a general-purpose output line.

DDD[4:2] Bits

When the SPI is enabled, SPI input pins remain functioning regardless of the state of the corresponding DDD[4:2] bits. For SPI output pins, however, the corresponding DDD[4:2] bits must be set or the pins will function as general-purpose inputs.

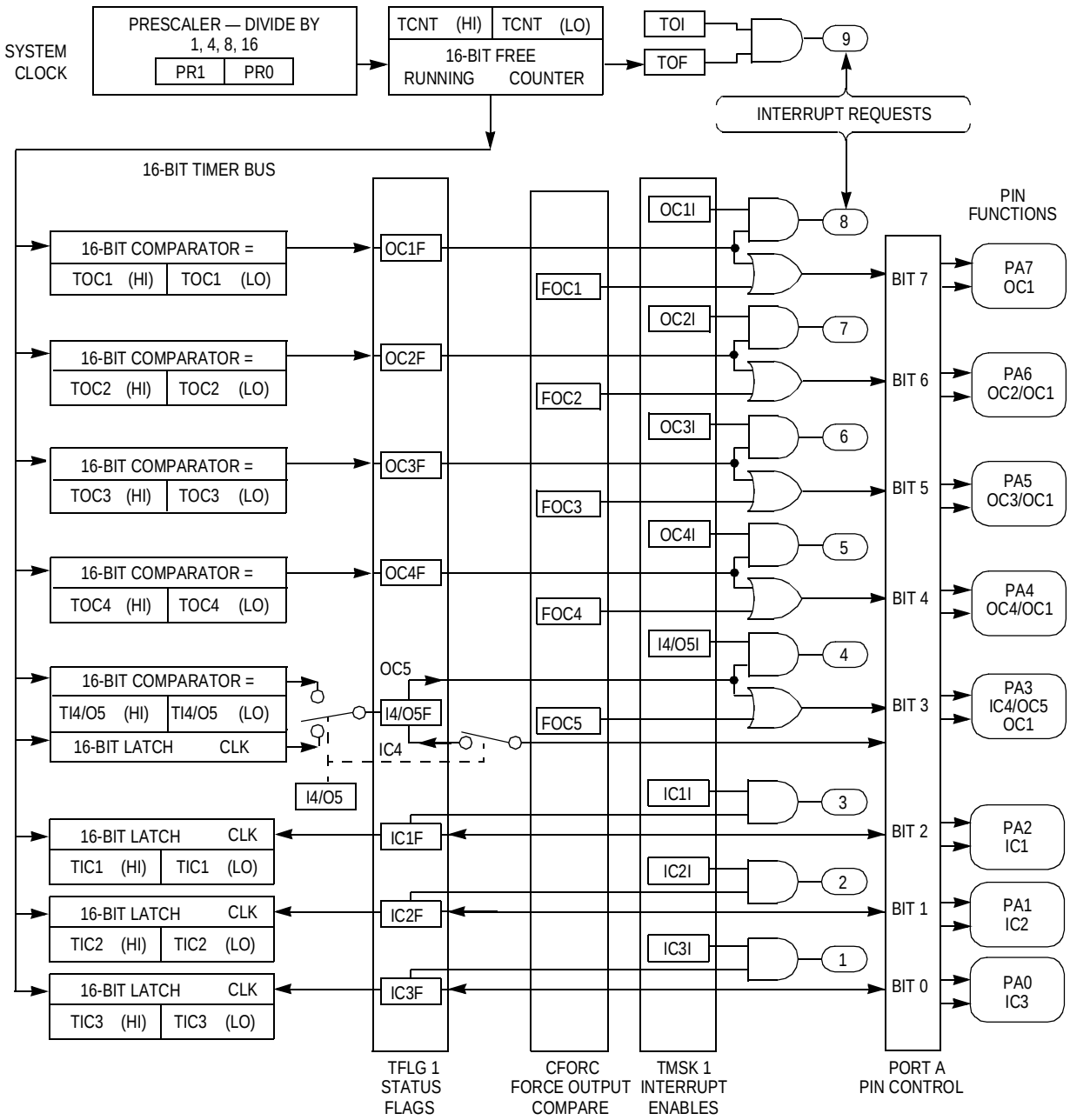


Figure 9-2. Capture/Compare Block Diagram

10.4.3 Analog-to-Digital Converter Result Registers

These read-only registers hold an 8-bit conversion result. Writes to these registers have no effect. Data in the A/D converter result registers is valid when the CCF flag in the ADCTL register is set, indicating a conversion sequence is complete. If conversion results are needed sooner, refer to **Figure 10-2**, which shows the A/D conversion sequence diagram.

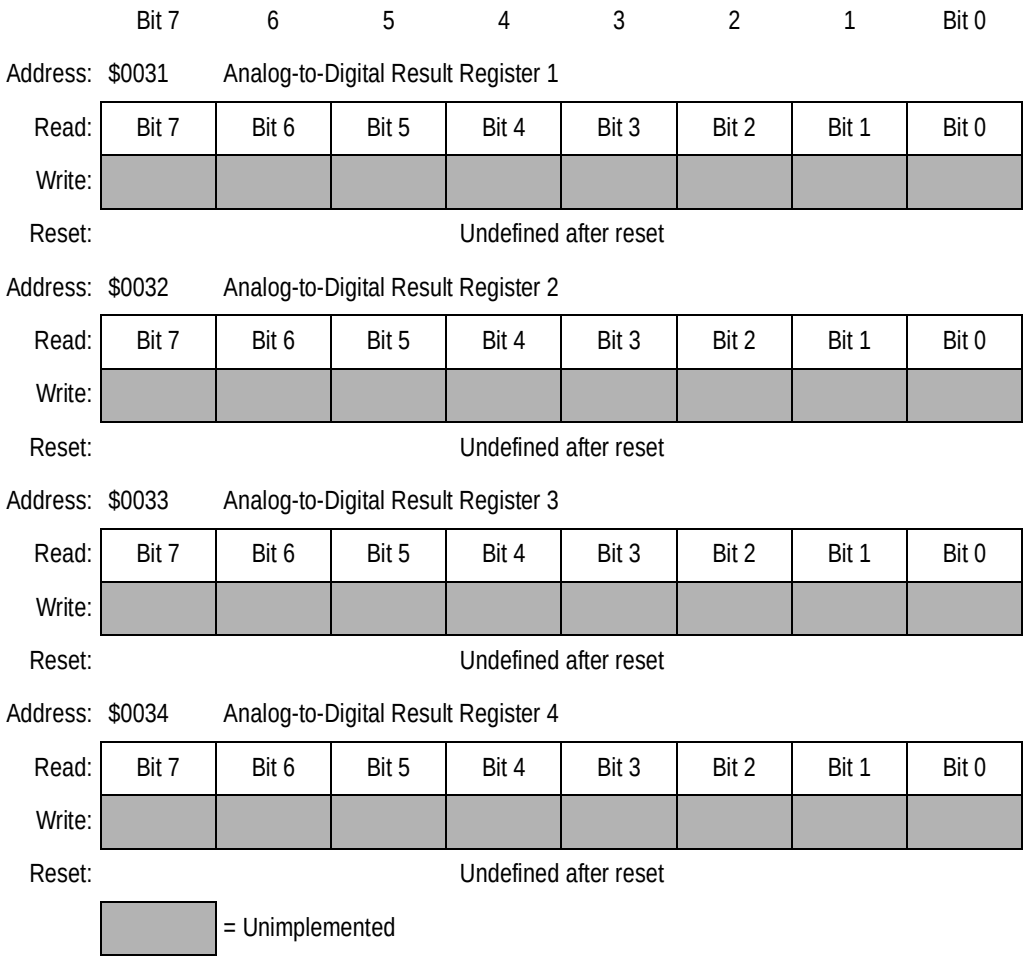


Figure 10-5. Analog-to-Digital Result Registers (ADR1–ADR4))

Section 11. Memory Expansion and Chip Selects

11.1 Contents

11.2 Introduction

This section provides descriptions of the expanded memory and the chip selects.

11.3.2.4 Memory Mapping Window Control Registers

Each of the memory mapping window control registers (MM1CR and MM2CR) determine the active memory bank for the corresponding window, containing the value to be output on the expansion address lines when the CPU selects addresses within its extended memory window. To change banks, write the address of the new bank into the appropriate window register.

Address: $0058 Memory Mapping Window 1 Control Register (MM1CR)

	Bit 7	6	5	4	3	2	1	Bit 0
Read: Write:	0	X1A18	X1A17	X1A16	X1A15	X1A14	X1A13	0
Reset:	0	0	0	0	0	0	0	0

Address: $0059 Memory Mapping Window 2 Control Register (MM1CR)

Read: Write:	0	X2A18	X2A17	X2A16	X2A15	X2A14	X2A13	0
Reset:	0	0	0	0	0	0	0	0

Figure 11-4. Memory Mapping Window Control Registers (MM1CR and MM2CR)

X1A[18:13] — Memory Mapping Window 1 Expansion Address Line Select Bits

X2A[18:13] — Memory Mapping Window 2 Expansion Address Line Select Bits

Each bit value written to the MMxCR registers is driven on the corresponding port G expansion address line (if enabled by PGAR) to enable the specified bank in the window.

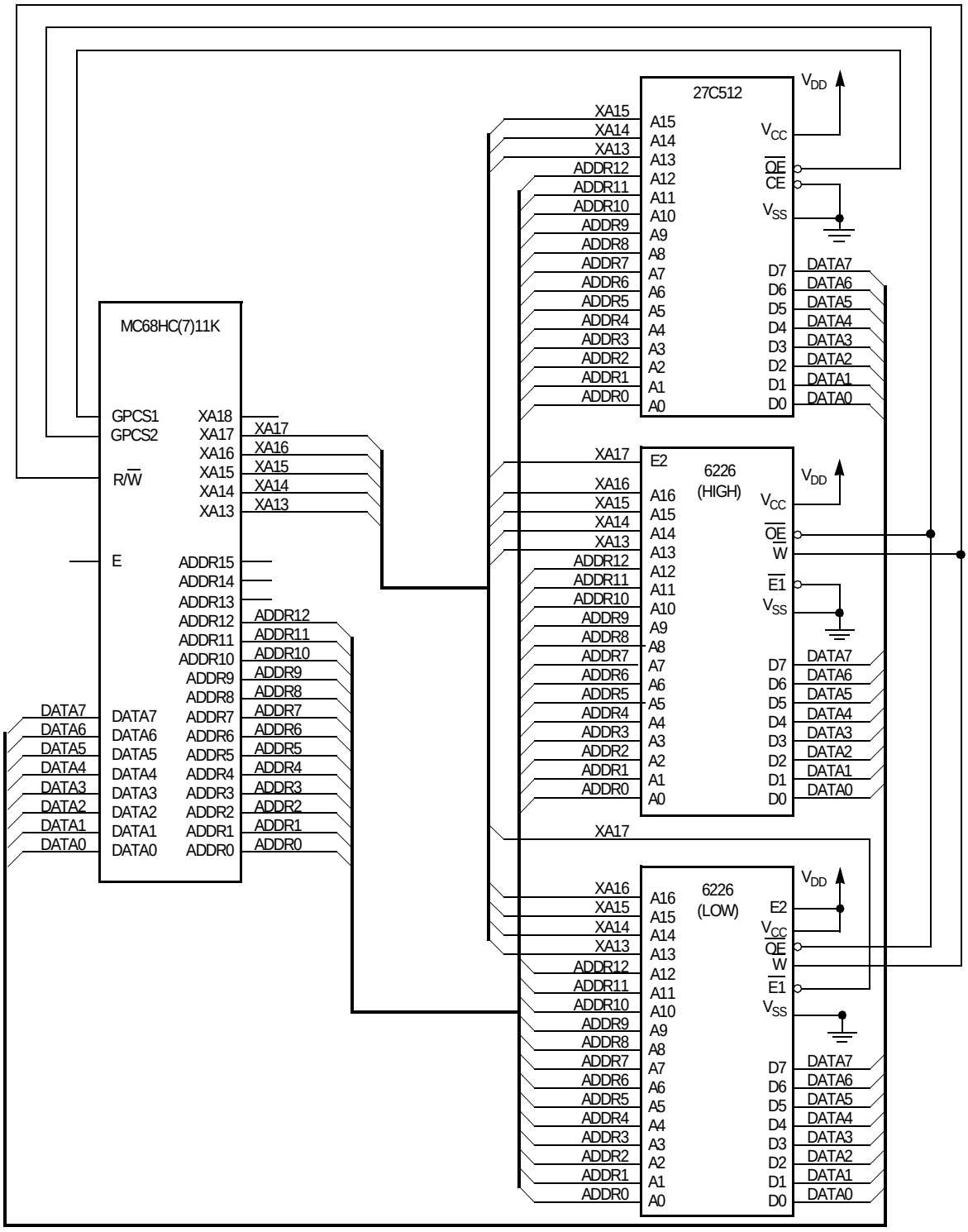


Figure 11-16. Memory Expansion Example 2 (Sheet 2 of 2)
Memory Map for One 8-Kbyte Window with Eight Banks and One 16-Kbyte Window with 16 Banks of External Memory

Electrical Characteristics

Characteristic[1]	Parameter	Min	Absolute	Max		Unit
				$f_o \leq 2.0$ MHz	$f_o > 2.0$ MHz[2]	
Sample acquisition time	Analog input acquisition sampling time: E Clock Internal RC Oscillator	 — —	 12 —	 — 12	 — 12	 t_{cyc} µs
Sample/hold capacitance	Input capacitance during sample PE[7:0]	—	20 (Typ)	—	—	pF
Input leakage	Input leakage on A/D pins PE[7:0] V_{RL}, V_{RH}	 — —	 — —	 400 1.0	 400 1.0	 nA µA

1. V_{DD} = 4.5 to 5.5 Vdc for standard devices; V_{SS} = 0 Vdc, $T_A = T_L$ to T_H Source impedances greater than 10 kΩ affect accuracy adversely because of input leakage.
   All timing measurements refer to 20% V_{DD} and 70% V_{DD}, unless otherwise noted.
2. Up to 4.0 MHz for standard devices.
3. Performance is verified down to 2.5 V ΔV_R, but accuracy is tested and guaranteed at ΔV_R = 5 V ±10%.