MC68HC05L4/D

MC68HC05L4

HC05

TECHNICAL DATA

MC68HC05L4
MC68HC705L4

TECHNICAL
DATA

!MOTOROLA

!MOTOROLA

MC68HC05L4
MC68HC705L4

High-density Complementary Metal Oxide Semiconductor (HCMOS) Microcomputer Unit

Table of Contents

Table of Contents (continued)

Table of Contents (continued)

Table of Contents (continued)

Table of Contents (continued)

Table of Contents (continued)

LIST OF FIGURES

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF TABLES

THIS PAGE INTENTIONALLY LEFT BLANK

SECTION 1
INTRODUCTION

The MC68HC05L4, with 8 kbytes of mask-programmable ROM, is designed around the industry standard M68HC05 CPU core with its familiar and efficient instruction set. The device features LCD driver circuitry, two 16-bit timers, a 15-stage multi-purpose ripple counter/core timer and a Computer Operating Properly (COP) watchdog timer. The LCD driver circuitry provides up to 24 frontplane drivers with 2, 3 or 4-way multiplexing, giving a maximum of 96 segments.

This data sheet covers both the MC68HC05L4 ROM based device and the equivalent MC68HC705L4 EPROM device. All references in the text to the MC68HC05L4 apply equally to the MC68HC705L4, unless otherwise stated. *References specific to the MC68HC705L4 are italicised in the text.*

1.1 FEATURES

- Fully static design featuring the industry standard M68HC05 family CPU core
- On-chip oscillator with divide by 2 mechanism for internal bus frequency
- 8032 bytes of user ROM (MC68HC05L4); *8032 bytes of user EPROM (MC68HC705L4),* plus 16 bytes of user vectors
- 224 bytes of RAM
- Power saving STOP, WAIT and data retention modes
- Programmable LCD driver (up to 24 frontplane drivers and 2, 3 or 4 backplane drivers)
- 15-stage multi-purpose ripple counter with timer overflow, power-on-reset, COP watchdog and real time Interrupt functions
- Computer Operating Properly (COP) watchdog timer
- Two 16-bit free running timers, each with input capture and output compare functions
- One interrupt request input plus 5 on-board hardware interrupt sources
- Keyboard wake-up feature on 8 input lines
- Three 8-bit parallel I/O ports and one 8-bit input only port

- Packages available:-
  64-pin plastic shrink dual-in-line package (PSDIP)
  64-pin plastic quad flat pack (PQFP)
  68-pin plastic leaded chip carrier (PLCC)
  64-pin windowed ceramic shrink dual-in-line package (CSDIP)
  68-pin windowed ceramic leaded chip carrier (Cerquad)

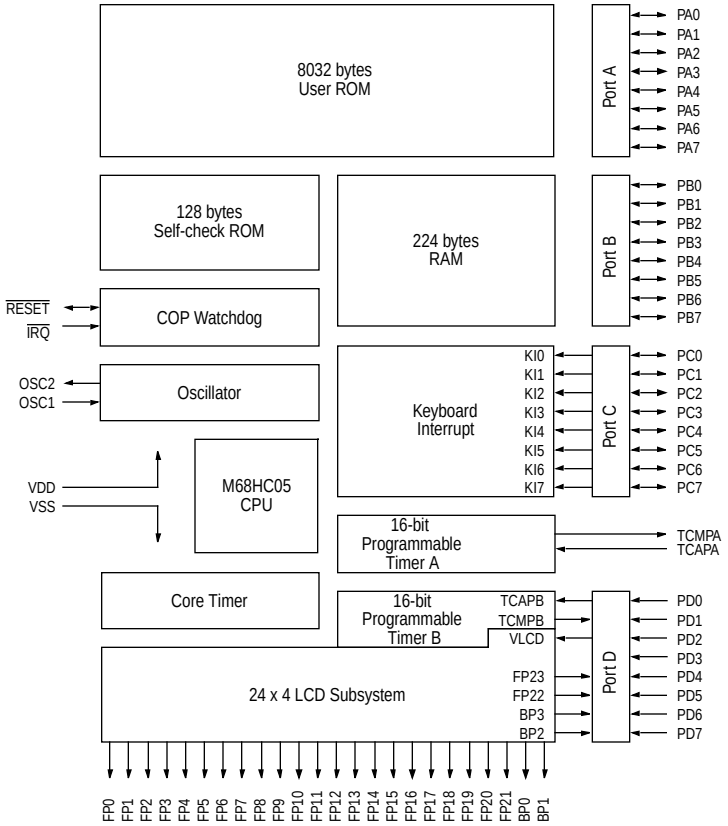


Figure 1-1 Functional block diagram

1.2 MASK OPTIONS ON THE MC68HC05L4

There are four mask options on the MC68HC05L4. These options are programmed during manufacture and must be specified on the order form.

- STOP instruction enable/disable
- COP Watchdog enable/disable
- Edge or edge-and-level sensitive interrupt triggering
- LCD clock select from 2 MHz or 16 kHz bus speed

1.3 MASK OPTIONS ON THE MC68HC705L4

With the exception of the STOP instruction, which is permanently enabled, the same mask options as for the MC68HC05L4 are available on the MC68HC705L4. These options must be programmed into the mask option register (MOR), at EPROM address $3FE0, via the bootloader mode prior to operating the device in single chip mode. The MOR is latched in at reset in single chip mode to allow emulation of the masked ROM part.

1.3.1 Mask Option Register (MOR)

This register may be written to only in Bootloader mode.

Address	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	State on Reset
$3FE0			LCD				IRQ	COP	unaffected

1.3.1.1 LCD — LCD clock option

1 (set) – LCD clock selected to run from bus speed of 16 kHz.

0 (clear) – LCD clock selected to run from bus speed of 2 MHz.

1.3.1.2 IRQ — Interrupt triggering sensitivity

1 (set) – $\overline{IRQ}$ interrupt trigger is edge-and-level sensitive.

0 (clear) – $\overline{IRQ}$ interrupt trigger is edge sensitive only.

1.3.1.3 COP — Computer Operating Properly watchdog enable/disable

1 (set) – The COP is enabled.

0 (clear) – The COP is disabled.

THIS PAGE INTENTIONALLY LEFT BLANK

SECTION 2
MODES OF OPERATION AND PIN DESCRIPTIONS

2.1 MODES OF OPERATION

The MC68HC05L4 has one mode of operation, namely single chip mode. *In addition, the MC68HC705L4 has two bootloader modes of operation, (EPROM and RAM).* Table 2-1 shows the conditions required to enter each mode on the rising edge of $\overline{\text{RESET}}$.

Table 2-1 Operating mode entry conditions

$\overline{\text{IRQ}}$/VPP	PB1	PB2	PB3	PB4	Mode	
V_{SS} to V_{DD}	x†	x	x	x	Single chip	
2 x V_{DD}	*1*	*0*	*1*	*1*	*EPROM bootloader*	*Program/Verify*
2 x V_{DD}	*1*	*0*	*0*	*1*		*Verify only*
2 x V_{DD}	*1*	*1*	*0*	x	*RAM bootloader*	*Jump to RAM ($51)*
2 x V_{DD}	*1*	*1*	*1*	x		*Load/execute RAM*

† x = Don't care

2.1.1 Single chip mode

This is the normal operating mode of the MC68HC05L4. In this mode the device functions as a self-contained microcomputer (MCU) with all on-board peripherals, including the three 8-bit I/O ports and the 8-bit input-only port, available to the user. All address and data activity occurs within the MCU.

Single chip mode is entered on the rising edge of $\overline{\text{RESET}}$ if the voltage level on the $\overline{\text{IRQ}}$ pin is within the normal operating range.

Warning: *In the MC68HC705L4, all vectors are fetched from EPROM in single chip mode; therefore, the EPROM must be programmed (via the bootloader mode) before the device is powered up in single chip mode. The mask option register is loaded from the EPROM ($3FE0) on reset so that emulation of mask programmable features is possible.*

2.1.2 EPROM bootloader mode for the MC68HC705L4

This mode is used for programming the on-board EPROM and mask option register. In bootloader mode the operation of the device is the same as in single chip mode, except that the vectors are fetched from a reserved area of ROM at locations $3FE4 to $3FEF, instead of the EPROM. The pin assignments are identical to that of single chip mode shown earlier. A recommended programming circuit is shown in Figure 2-1.

Because the addresses in the MC68HC705L4 and the EPROM containing the user code are incremented independently, it is essential that the data layout in the 27128 EPROM conforms exactly to the MC68HC705L4 memory map. The bootloader uses an external 14-bit counter to address the memory device containing the code to be copied. This counter requires a clock and a reset function.

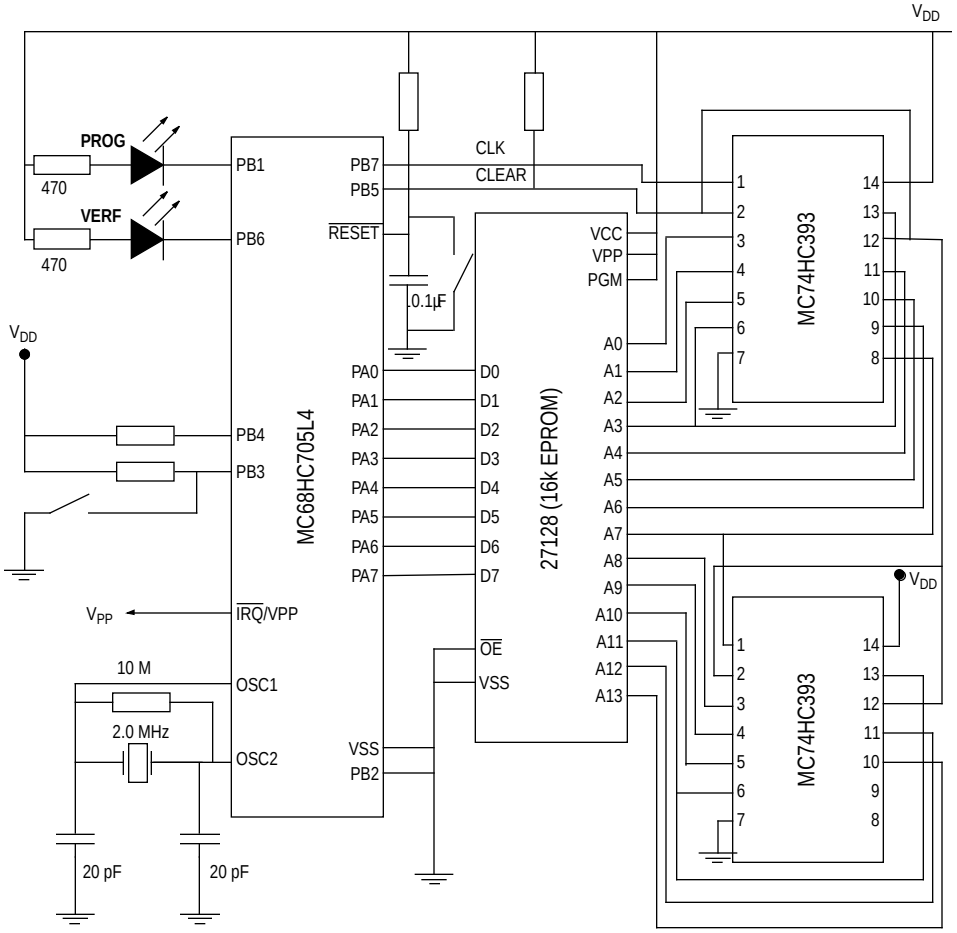
In this mode all interrupt vectors are mapped to pseudo-vectors in RAM (refer to Table 2-2). This allows programmers to use their own service routine addresses. Each pseudo-vector is allowed three bytes of space, rather than the two bytes for normal vectors, because an explicit jump (JMP) opcode is needed to cause the desired jump to the user's service routine address.

2.1.2.1 Bootloader functions

The bootloader code deals with the copying of user code from an external EPROM into the on-chip EPROM. The bootloader function can only be used with an external EPROM. The bootloader performs a programming pass and then does a verify pass.

Pins PB3 and PB4 are used to select various bootloader functions, including the programming mode (see Figure 2-1). Two other pins, PB1 and PB6 are used to drive the PROG and VERF LED outputs. While the EPROM is being programmed the PROG LED lights up; when programming is complete the internal EPROM contents are compared to that of the external EPROM and, if they match exactly, the VERF LED lights up.

Note: The EPROM must be erased before performing a program cycle.



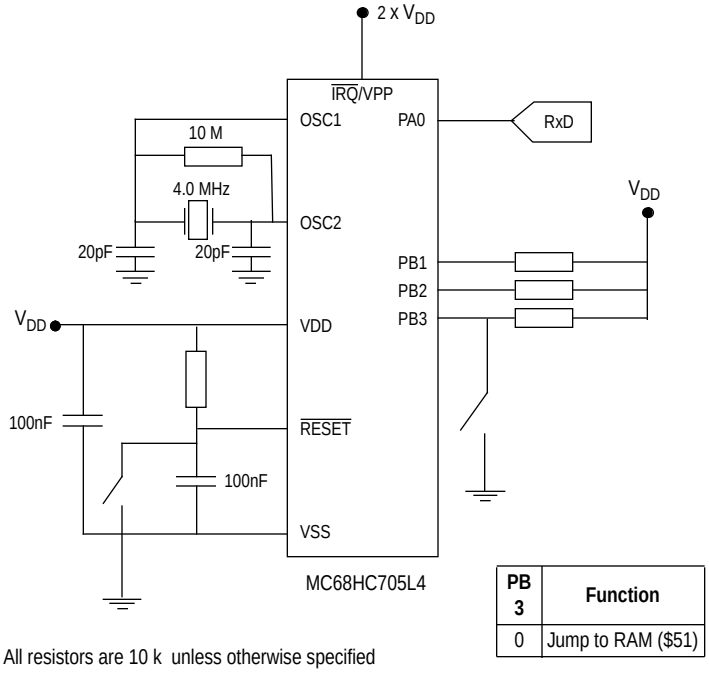
All resistors are 10 k unless otherwise specified

PB3	MODE
1	Program/Verify
0	Verify only

Figure 2-1 *MC68HC705L4 EPROM programming circuit*

2.1.3 RAM bootloader mode for the MC68HC705L4

In addition to the EPROM bootloader mode on the MC68HC705L4, there is a RAM bootloader mode that allows the user to perform simple load and execute instructions in ROM. To make use of this feature a circuit board should be constructed as shown in Figure 2-2. It is then possible, by correctly configuring port pins PB2 and PB3, to load a user program into RAM and then execute.



PB 3	Function
0	Jump to RAM ($51)

Figure 2-2 *RAM bootloader circuit*

The RAM bootloader is selected when the device is put into bootloader mode with PB2 held high. If PB3 is low, the program counter is set to $51 and a previously loaded RAM program can be executed. If PB3 is high at reset a program is serially loaded from PA0 into the RAM and executed from $51.

The first byte to be loaded is the count byte which should hold the program length plus the count byte. Therefore, for a program length of $30, the count should equal $31. The maximum program size including the count byte is 172 bytes ($AC), since 4 bytes must be left for the stack during download.

For a 4 MHz crystal, the serial data format is 9600 baud, low start bit, 8 data bits, high stop bit.

Note: In this mode, the COP must be disabled, otherwise it could interrupt the serial download.

In the RAM bootloader mode all interrupt vectors are mapped to pseudo-vectors in RAM (refer to Table 2-2). This allows programmers to use their own service routine addresses. Each pseudo-vector is allowed three bytes of space, rather than the two bytes for normal vectors, because an explicit jump (JMP) opcode is needed to cause the desired jump to the user's service routine address.

Table 2-2 *EPROM/RAM bootloader mode jump vectors*

Address	Pseudo-vector
005F	Timer B interrupt
005C	Timer A interrupt
0059	Core Timer interrupt
0056	IRQ/Keyboard interrupt
0053	Software interrupt

2.2 PIN DESCRIPTIONS

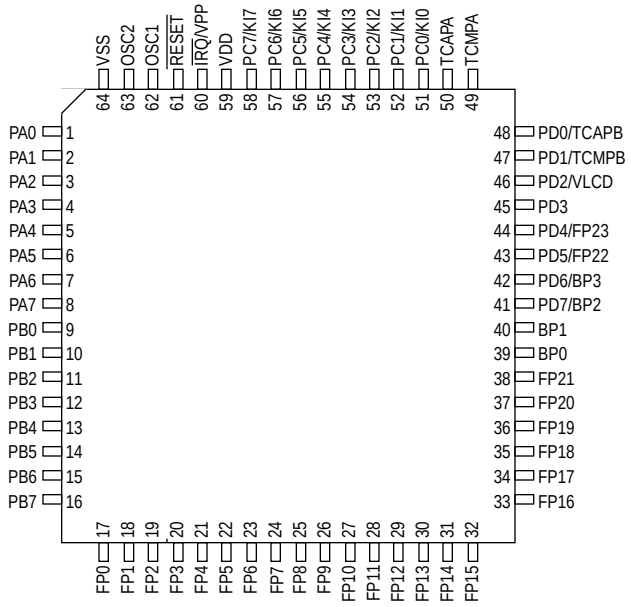


Figure 2-3 64-pin QFP single chip and bootloader mode pin assignments

Note: Pin assignments for all other package types are shown in Section 11.

2.2.1 VDD and VSS

Power is supplied to the microcontroller using these two pins. VDD is the positive supply and VSS is ground.

It is in the nature of CMOS designs that very fast signal transitions occur on the MCU pins. These short rise and fall times place very high short-duration current demands on the power supply. To prevent noise problems, special care must be taken to provide good power supply by-passing at the MCU. By-pass capacitors should have good high-frequency characteristics and be as close to the MCU as possible. By-passing requirements vary, depending on how heavily the MCU pins are loaded.

2.2.2 $\overline{\text{IRQ}}$/*VPP*

This is an input-only pin for external interrupt sources. Interrupt triggering can be either edge sensitive or edge-and-level sensitive, see Section 1.2 and *Section 1.3*. The $\overline{\text{IRQ}}$ pin contains an internal Schmitt trigger as part of its input to improve noise immunity. *This pin also serves as the input pin for the EPROM programming voltage.*

2.2.3 OSC1, OSC2

These pins provide control input for an on-chip oscillator circuit. A crystal, ceramic resonator or external clock signal connected to these pins supplies the oscillator clock. The oscillator frequency (f_{osc}) is divided by two to give the internal bus frequency (f_{op}).

2.2.3.1 Crystal

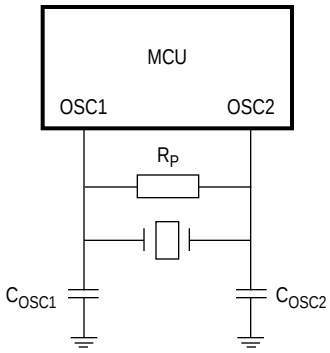
The circuit shown in Figure 2-4(a) is recommended when using either a crystal or a ceramic resonator. Figure 2-4(d) lists the recommended capacitance and feedback resistance values. The internal oscillator is designed to interface with an AT-cut parallel-resonant quartz crystal resonator in the frequency range specified for f_{osc} (see Section 10.4). Use of an external CMOS oscillator is recommended when crystals outside the specified ranges are to be used. The crystal and associated components should be mounted as close as possible to the input pins to minimise output distortion and start-up stabilisation time. The manufacturer of the particular crystal being considered should be consulted for specific information.

2.2.3.2 Ceramic Resonator

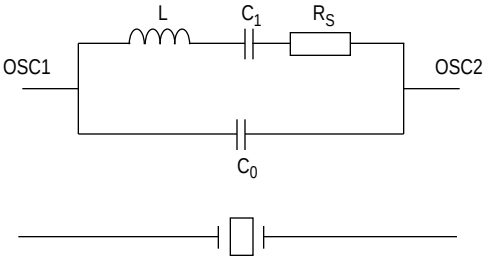
A ceramic resonator may be used instead of a crystal in cost sensitive applications. The circuit shown in Figure 2-4(a) is recommended when using either a crystal or a ceramic resonator. Figure 2-4(d) lists the recommended capacitance and feedback resistance values. The manufacturer of the particular ceramic resonator being considered should be consulted for specific information.

2.2.3.3 External Clock

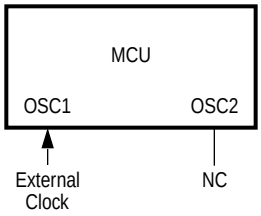
An external clock should be applied to the OSC1 input, with the OSC2 pin left unconnected, as shown in Figure 2-4(c). The t_{OXOV} specification (see Section 10.4) does not apply when using an external clock input. The equivalent specification of the external clock source should be used in lieu of t_{OXOV}.



(a) Crystal/Ceramic Resonator Oscillator Connections



(b) Crystal Equivalent Circuit



(c) External Clock Source Connections

Crystal

	2MHz	4MHz	Unit
R_S(max)	400	75	
C_0	5	7	pF
C_1	8	12	nF
C_{OSC1}	15 – 40	15 – 30	pF
C_{OSC2}	15 – 30	15 – 25	pF
R_P	10	10	M
Q	30 000	40 000	—

Ceramic Resonator

	2 – 4MHz	Unit
R_S(typ)	10	
C_0	40	pF
C_1	4.3	pF
C_{OSC1}	30	pF
C_{OSC2}	30	pF
R_P	1 – 10	M
Q	1250	—

(d) Crystal and Ceramic Resonator Parameters

Figure 2-4 Oscillator Connections

2.2.4 $\overline{\text{RESET}}$

This active low input pin is used to reset the MCU. Applying a logic zero to this pin forces the device to a known start-up state. An external RC-circuit can be connected to this pin to generate a power-on-reset (POR) if required. In this case, the time constant must be great enough (at least 100ms) to allow the oscillator circuit to stabilise. This input has an internal Schmitt trigger to improve noise immunity. When a reset condition occurs internally, i.e. from the COP watchdog, the

$\overline{\text{RESET}}$ pin provides an active-low open drain output signal which may be used to reset external hardware.

2.2.5 PA0–PA7/PB0–PB7

These 16 I/O lines comprise ports A and B. The state of any pin is software programmable, and all the pins are configured as inputs during power-on or reset.

2.2.6 PC0–PC7

On reset this 8-bit port behaves like ports A and B, all its pins being configured as inputs. However, by writing ones to the bits in the Keyboard Interrupt Mask register (KBI) at location $0020, the corresponding port C pins are configured as inputs and each will respond to a high-to-low input transition by generating keyboard interrupt requests, provided the KBIE bit in $1E is set. The interrupt vector is shared with that for $\overline{\text{IRQ}}$. A keyboard interrupt will bring the MCU out of STOP or WAIT mode.

2.2.7 PD0–PD7

This 8-bit input-only port shares its pins with other subsystems on the MCU and is controlled using the port D control register (COND). On reset, COND is cleared which forces the port D pins to be input port pins. Writing a ‘1’ to any bit in COND connects the subsystem function to the corresponding port D pin.

2.2.7.1 PD0/TCAPB, PD1/TCMPB

The timer B input capture and output compare functions are available on PD0 and PD1 when the corresponding bits of the port D control register are set.

2.2.7.2 PD2/VLCD

PD2 becomes the input for the analog LCD supply voltage when bit 2 of the port D control register is set.

2.2.7.3 PD3

PD3 is an input-only port pin.

2.2.7.4 PD4/FP23, PD5/FP22

LCD frontplane drivers FP23 and FP22 are available on PD4 and PD5 respectively when the corresponding bits of the port D control register are set.

2.2.7.5 PD6/BP3, PD7/BP2

LCD backplane drivers BP3 and BP2 are available on PD6 and PD7 respectively when the corresponding bits of the port D control register are set.

2.2.8 BP0, BP1

These are the two remaining backplane drivers. These pins are dedicated to the LCD subsystem.

2.2.9 FP21–FP0

These are the remaining frontplane drivers of the device. These pins are dedicated to the LCD subsystem.

2.2.10 TCAPA, TCMPA

These pins are dedicated to the timer A input capture and output compare functions.

2.3 LOW POWER MODES

2.3.1 STOP

The STOP instruction places the MCU in its lowest power consumption mode. In STOP mode, the internal oscillator is turned off, halting all internal processing, including timer (and COP watchdog timer) operation.

During the STOP mode, the core timer interrupt flags (CTOF and RTIF) and interrupt enable bits (CTOFE and RTIE) in the CTCSR, as well as the timer flags for timers A and B in the respective TSR registers, and interrupt enable bits in the TCR registers, are cleared by internal hardware. This removes any pending timer interrupt requests and disables any further timer interrupts. The timer prescalers are cleared. The I-bit in the CCR is cleared to enable external interrupts. All other registers, the remaining bits in the CTCSR, and memory contents remain unaltered. All input/output lines remain unchanged. The processor can be brought out of the STOP mode only by an external interrupt, a keyboard interrupt or a reset (see Figure 2-5).

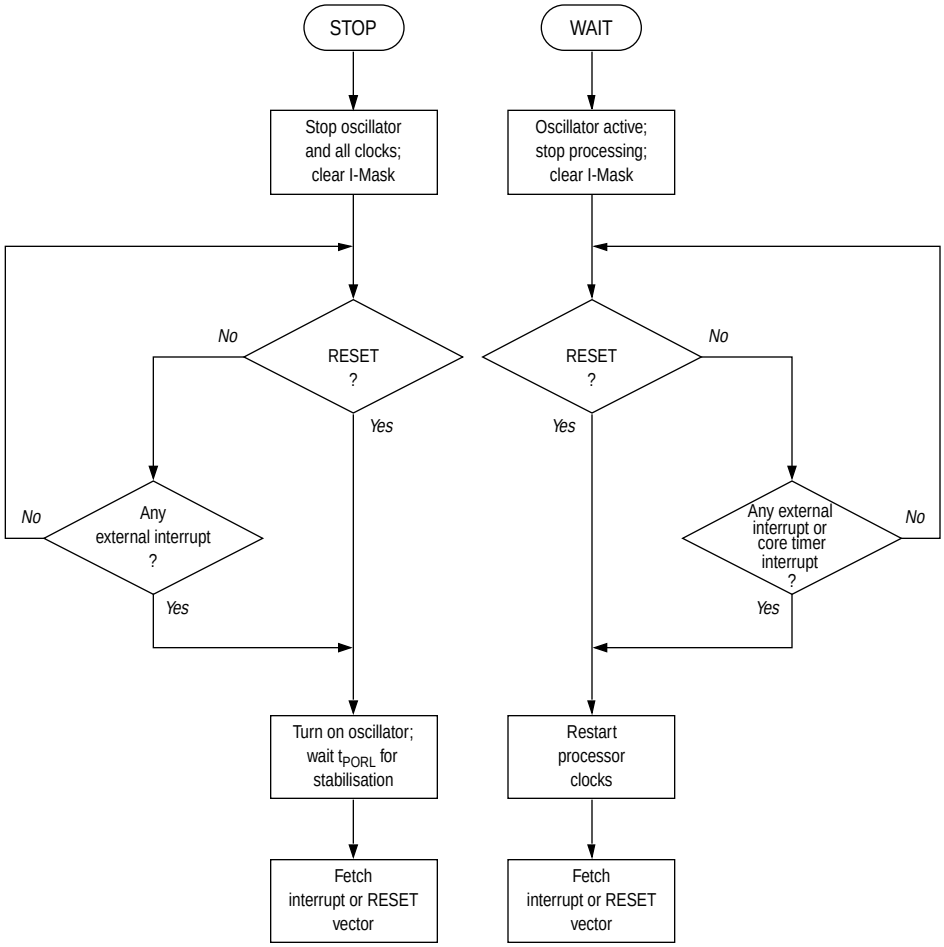


Figure 2-5 STOP and WAIT flowcharts

2.3.2 WAIT

The WAIT instruction places the MCU in a low power consumption mode, but the WAIT mode consumes more power than the STOP mode. All CPU action is suspended, but the core timer remains active. An interrupt from the core timer, if enabled, will cause the MCU to exit the WAIT mode.

During the WAIT mode, the I-bit in the CCR is cleared to enable interrupts. All other registers, memory and input/output lines remain in their previous state. The Core Timer may be enabled to allow a periodic exit from the WAIT mode (see Figure 2-5).

2.3.3 Data Retention

The contents of the RAM are retained at supply voltages as low as 2.0Vdc. This is called the Data Retention Mode, in which data is maintained but the device is not guaranteed to operate.

For lowest power consumption in data retention mode the device should be put into STOP mode before reducing the supply voltage, to ensure that all the clocks are stopped. If the device is not in STOP mode then it is recommended that $\overline{\text{RESET}}$ be held low whilst the power supply is outwith the normal operating range, to ensure that processing is suspended in an orderly manner.

Recovery from Data Retention Mode, after the power supply has been restored, is by an external interrupt, or by pulling the $\overline{\text{RESET}}$ line high.

SECTION 3
MEMORY AND REGISTERS

The MC68HC05L4 has a 16 kbyte memory map consisting of registers (for I/O, control and status), LCD RAM, RAM, User ROM *or EPROM*, *Bootloader ROM* and reset and interrupt vectors as shown in Figure 3-1.

3.1 REGISTERS

All the I/O, control and status registers of the MC68HC05L4 are contained within the first 48-byte block of the memory map, as detailed in Table 3-1.

3.2 LCD RAM

The 12 bytes of LCD RAM can be accessed at two different locations in the memory map ($0030-$003B and $0130-$013B). This RAM is used to store the data needed for the LCD . See Section 7.1 for further details of the organisation of these bits.

3.3 RAM

The user RAM consists of 224 bytes of memory, from $0050 to $0130. This is shared with a 64 byte stack area. The stack begins at $00FF and may extend down to $00C0.

Note: Using the stack area for data storage or temporary work locations requires care to prevent the data from being overwritten due to stacking from an interrupt or subroutine call.

3.4 NON-VOLATILE MEMORY (NVM)

The NVM consists of 8032 bytes of ROM (MC68HC05L4) *or EPROM (MC68HC705L4)* from $2000 to $3F60 and 16 bytes of user vectors from $3FF0 to $3FFF.

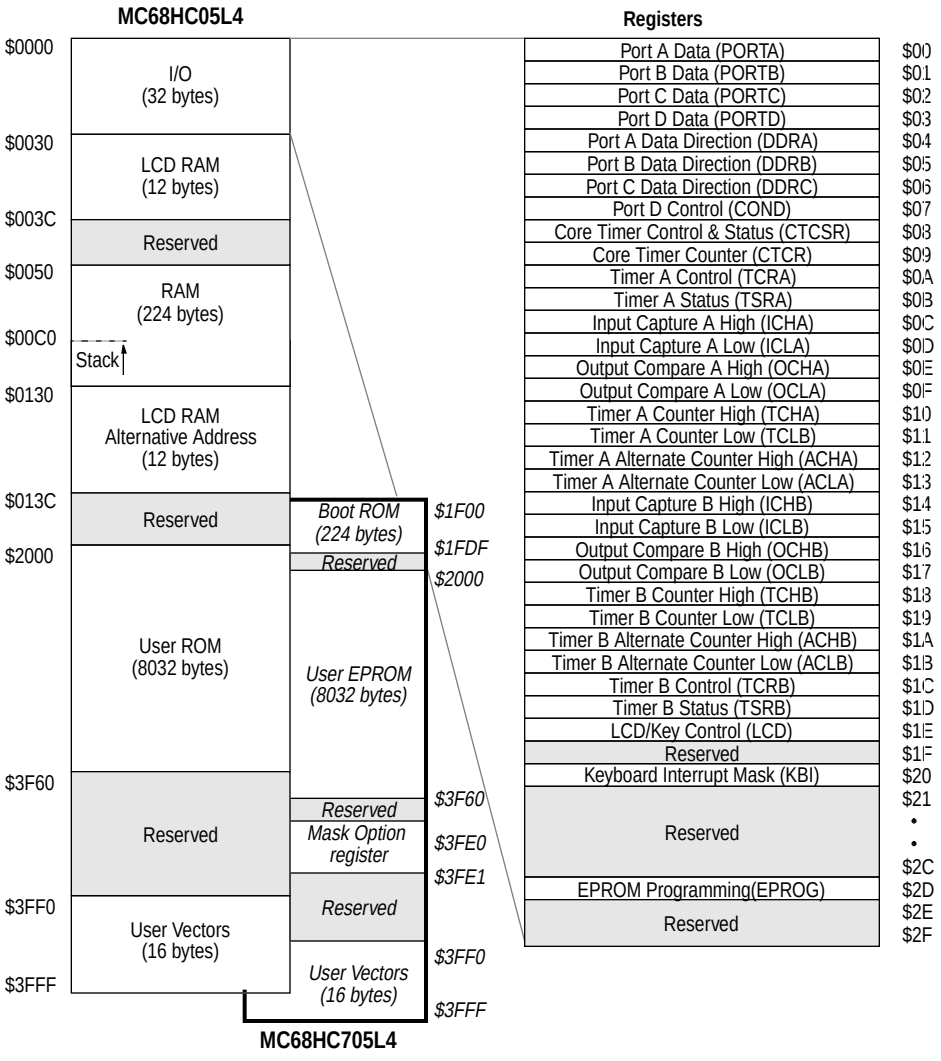


Figure 3-1 Memory map of the MC68HC05L4 and MC68HC705L4

3.4.1 EPROM Programming register (EPROG)

Address	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	State on Reset
$2D						LATCH	0	EPGM	- - - - -000

3.4.1.1 LATCH — EPROM Latch Control

1 (set) – EPROM address and data buses configured for programming. Causes address and data buses to be latched when a write to EPROM is carried out. EPROM cannot be read if LATCH = 1.

0 (clear) – EPROM address and data buses configured for normal reads.

3.4.1.2 EPGM — EPROM Program Control

1 (set) – Programming power connected to the EPROM array. EPGM can only be set if LATCH is set. EPGM is automatically cleared when LATCH = 0.

0 (clear) – Programming power disconnected from the EPROM array.

Note: LATCH and EPGM cannot be set on the same write operation.

Take the following steps to program a byte of EPROM:

1. *Apply the programming voltage V_{PP} to the VPP pin.*
2. *Set the LATCH bit.*
3. *Write to the EPROM address.*
4. *Set the EPGM bit for a time t_{PROG} to apply the programming voltage.*
5. *Clear the LATCH bit.*

Note: The erased state of the EPROM is $00.

Table 3-1 Register outline

Register Name	Address	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	State on Reset
Port A Data (PORTA)	$0000									unaffected
Port B Data (PORTB)	$0001									unaffected
Port C Data (PORTC)	$0002									unaffected
Port D Data (PORTD)	$0003									unaffected
Port A Data Direction (DDRA)	$0004									0000 0000
Port B Data Direction (DDRB)	$0005									0000 0000
Port C Data Direction (DDRC)	$0006									0000 0000
Port D Control (COND)	$0007	PD7/ BP2	PD6/ BP1	PD5/ FP22	PD4/ FP23	PD3	PD2/ VLCD	PD1/ TCMPB	PD0/ TCAPB	0000 0000
Core Timer Control/Status (CTCSR)	$0008	CTOF	RTIF	CTOFE	RTIE			RT1	RT0	0000 0011
Core Timer Counter (CTCR)	$0009									0000 0000
Timer A Control (TCRA)	$000A	ICIE	OCIE	TOIE				IEDG	OLVL	0000 00U0
Timer A Status (TSRA)	$000B	ICE	OCF	TOF						uuu0 0000
Input Capture A High (ICHA)	$000C	(bit 15)							(bit 8)	unaffected
Input Capture A Low (ICLA)	$000D									unaffected
Output Compare A High (OCHA)	$000E	(bit 15)							(bit 8)	unaffected
Output Compare A Low (OCLA)	$000F									unaffected
Timer A Counter High (TCHA)	$0010	(bit 15)							(bit 8)	1111 1111
Timer A Counter Low (TCLA)	$0011									1111 1100
Alternate A Counter High (ACHA)	$0012	(bit 15)							(bit 8)	1111 1111
Alternate A Counter Low (ACLA)	$0013									1111 1100
Input Capture B High (ICHB)	$0014	(bit 15)							(bit 8)	unaffected
Input Capture B Low (ICLB)	$0015									unaffected
Output Compare B High (OCHB)	$0016	(bit 15)							(bit 8)	unaffected
Output Compare B Low (OCLB)	$0017									unaffected
Timer B Counter High (TCHB)	$0018	(bit 15)							(bit 8)	1111 1111
Timer B Counter Low (TCLB)	$0019									1111 1100
Alternate B Counter High (ACHB)	$001A	(bit 15)							(bit 8)	1111 1111
Alternate B Counter Low (ACLB)	$001B									1111 1100
Timer B Control (TCRB)	$001C	ICIE	OCIE	TOIE				IEDG	OLVL	0000 00U0
Timer B Status (TSRB)	$001D	ICE	OCF	TOF						uuu0 0000
LCD/Key Control (LCD)	$001E	KSF	KBIE	FC	LC	FDISP	MUX4	MUX3	DISON	0000 0000
Reserved	$001F									
Keyboard Interrupt Mask (KBI)	$0020									0000 0000
Reserved	$0021									
EPROM Programing (EPROG)	$002D	0	0	0	0	0	LATCH	0	EPGM	0000 0000
Reserved	$2E–$2F									

SECTION 4
INPUT/OUTPUT PORTS

In single-chip mode, the MC68HC05L4 has a total of 24 I/O lines, arranged as three 8-bit ports (A, B and C), and eight input-only lines, arranged as one 8-bit port (D). Each I/O line is individually programmable as either input or output, under the software control of the data direction registers. The 8-bit port C can also be configured to respond to keyboard interrupts. The 8-bit input-only port shares various I/O configurations with the timer and LCD subsystems.

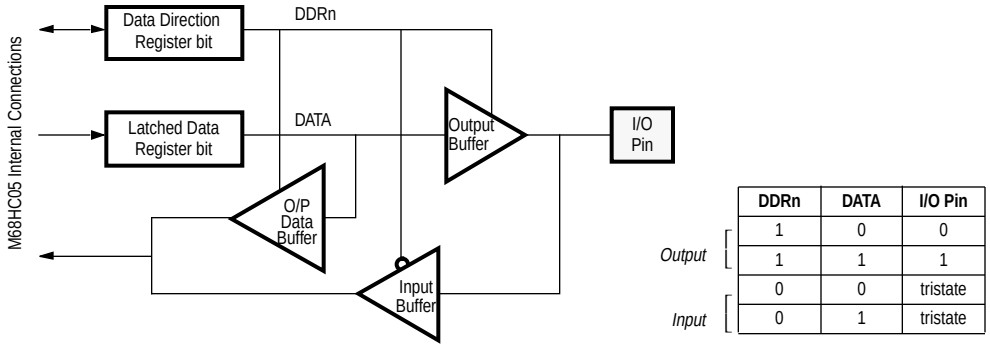
To avoid glitches on the output pins, data should be written to the I/O port data register before writing ones to the corresponding data direction register bits to set the pins to output mode.

4.1 INPUT/OUTPUT PROGRAMMING

The bidirectional port lines may be programmed as inputs or outputs under software control. The direction of each pin is determined by the state of the corresponding bit in the port data direction register (DDR). Each port has an associated DDR. Any I/O port pin is configured as an output if its corresponding DDR bit is set to a logic one. A pin is configured as an input if its corresponding DDR bit is cleared to a logic zero.

At power-on or reset, all DDRs are cleared, thus configuring all port pins as inputs. The data direction registers can be written to or read by the MCU. During the programmed output state, a

read of the data register actually reads the value of the output data latch and not the I/O pin. The operation of the standard port hardware is shown schematically in Figure 4-1.



DDRn	DATA	I/O Pin
1	0	0
1	1	1
0	0	tristate
0	1	tristate

Figure 4-1 Standard I/O port structure

This is further summarised in Table 4-1, which shows the effect of reading from, or writing to an I/O pin in various circumstances. Note that the read/write signal shown is internal and not available to the user.

Table 4-1 I/O pin states

R/$\overline{W}$	DDRn	Action of MCU write to/read of data bit
0	0	The I/O pin is in input mode. Data is written into the output data latch.
0	1	Data is written into the output data latch, and output to the I/O pin.
1	0	The state of the I/O pin is read.
1	1	The I/O pin is in output mode. The output data latch is read.

4.2 PORTS A AND B

These ports are standard M68HC05 bidirectional I/O ports, each comprising a data register and a data direction register.

Reset does not affect the state of the data register, but clears the data direction register, thereby returning all port pins to input mode. Writing a ‘1’ to any DDR bit sets the corresponding port pin to output mode.

4.3 PORT C

In addition to the standard port functions, this 8-bit port has a programmable keyboard interrupt feature, with internal pull-up resistors. (This pull-up is present whenever the port is in input mode.) On reset, this port is configured as a standard I/O port, comprising a data register and a data direction register.

Reset does not affect the state of the data register, but clears the data direction register, thereby returning all port pins to input mode. Writing a ‘1’ to any DDR bit sets the corresponding port pin to output mode.

Provided that the interrupt mask bit of the condition code register is cleared, the keyboard interrupt facility is enabled by setting the keyboard interrupt enable bit (KBIE) in the LCD/key control register at location $1E. Which port C inputs can generate the keyboard interrupt is selected by writing a ‘1’ to the corresponding bit in the keyboard interrupt mask register (KBI) at location $0020. For further details see Section 4.5.2.

Setting KBIE will force each of the I/O lines of the port, selected by $20, to be an input, with an internal pull-up resistor, providing the corresponding bit in the port D control register is set (see Table 10-3 and Table 10-5). The structure of the port pins is shown diagrammatically in Figure 4-2. When a high-to-low transition is detected on any of this port’s pins, selected by the KBI register, then a keyboard interrupt request is generated and the keyboard status flag (KSF) is set. The address of the interrupt service routine is specified by the contents of memory locations $3FFA and $3FFB. Since this interrupt vector is shared with the IRQ external interrupt function the interrupt service routine should check KSF to determine the interrupt source. KSF should be cleared by software in the interrupt service routine. The keyboard interrupt is edge and level-sensitive. Care must be taken to allow adequate time for switch debounce before clearing the flag.

A keyboard interrupt will force the MCU out of STOP or WAIT mode.

On reset, all bits in the LCD register are cleared, hence disabling all keyboard interrupts.

4.4 PORT D

Port D is an 8-bit port which shares its pins with the timer B subsystem and the LCD driver outputs. All the port D pins can be configured as input-only lines and PD0 – PD2 and PD4 – PD7 can be used by other systems within the MCU. Setting the port D control register bits to zero selects the corresponding port D pin as an input, while a ‘1’ selects the MCU subsystem. Writing a ‘1’ to bit 3 has no effect since this bit has no shared function. For details of the alternate function of each port D pin see Section 2.2.7.

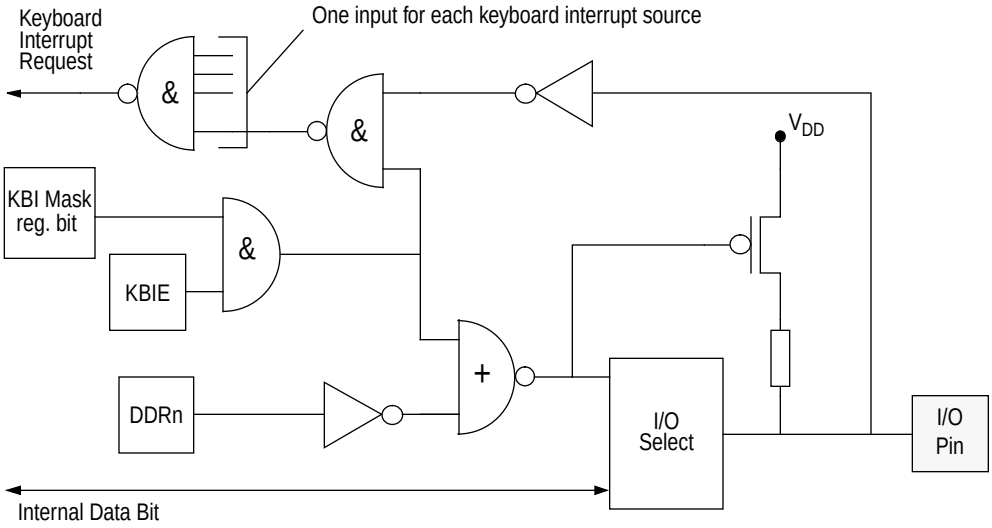


Figure 4-2 Structure of port with keyboard interrupt

4.5 PORT REGISTERS

The following sections explain in detail the individual bits in the data and control registers associated with the ports.

4.5.1 Port Data Registers (PORTA, PORTB and PORTC)

Address	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	State on Reset
$0000									unaffected
$0001									unaffected
$0002									unaffected

Each bit can be configured as input or output via the corresponding data direction bit in the port data direction register (DDRx).

Reset does not affect the state of the port data registers.

In addition to the normal port functions, port C is equipped with a software maskable keyboard interrupt capability as described in Section 4.3. Described below are the registers associated with this feature.

4.5.2 Keyboard Interrupt Mask Register (KBI)

Address	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	State on Reset
$0020									0000 0000

Writing a ‘1’ to a bit in the KBI register provides the corresponding port C pin with keyboard interrupt capability provided KBIE is set.

Reset clears this register, thus returning all port C pins to normal I/O lines.

4.5.3 LCD/key Control Register (LCD)

Address	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	State on Reset
$001E	KSF	KBIE	FC	LC	FDISP	MUX4	MUX3	DISON	0000 0000

Bits 0 - 5 of the LCD Control register are dedicated to the LCD subsystem and as such are described in Section 7.4.2.

4.5.3.1 KSF — Keyboard Interrupt Status Flag

1 (set) – A keyboard interrupt has been generated.

0 (clear) – No keyboard interrupt has been generated.

The keyboard interrupt status flag is cleared by writing a zero to it. It has no automatic clearing mechanism.

4.5.3.2 KBIE — Keyboard Interrupt Enable

1 (set) – Keyboard interrupt enabled.

0 (clear) – Keyboard interrupt disabled.

4.5.4 Port D Data Register

Address	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	State on Reset
$0003									Unaffected

With the exception of bit 3, which is an input-only bit, each of the port D bits is shared with another MCU subsystem. The configuration of this register is determined by the setting of individual bits in the port D control register. See Section 4.5.6.

4.5.5 Data Direction Registers (DDRA, DDRB and DDRC)

Address	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	State on Reset
$0004									0000 0000
$0005									0000 0000
$0006									0000 0000

Writing a ‘1’ to any bit configures the corresponding port pin as an output; conversely, writing any bit to ‘0’ configures the corresponding port pin as an input.

Reset clears these registers, thus configuring all ports as inputs.

4.5.6 Port D Control Register (COND)

Address	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	State on Reset
$0007	PD7/ BP2	PD6/ BP1	PD5/ FP22	PD4/ FP23	PD3	PD2/ VLCD	PD1/ TCMPB	PD0/ TCAPB	0000 0000

4.5.6.1 PD7/BP2

1 (set) – Pin PD7 is configured as backplane 2 of the LCD subsystem.

0 (clear) – Pin PD7 is an input-only pin.

4.5.6.2 PD6/BP3

1 (set) – Pin 6 is configured as backplane 3 of the LCD subsystem.

0 (clear) – Pin 6 is an input-only pin.

4.5.6.3 PD5/FP22

1 (set) – Pin 5 is configured as frontplane 22 of the LCD subsystem.

0 (clear) – Pin 5 is an input-only pin.

4.5.6.4 PD4/FP23

1 (set) – Pin 4 is configured as frontplane 23 of the LCD subsystem.

0 (clear) – Pin 4 is an input-only pin.

4.5.6.5 PD3

1 (set) – Writing a ‘1’ to this location has no meaning or effect.

0 (clear) – Bit 3 is an input-only pin.

4.5.6.6 PD2/VLCD

1 (set) – Bit 2 is configured as the voltage input pin for the LCD subsystem.

0 (clear) – Bit 2 is an input-only pin.

4.5.6.7 PD1/TCMPB

1 (set) – Bit 1 is configured as the output compare for timer B.

0 (clear) – Bit 1 is an input-only pin.

4.5.6.8 PD0/TCAPB

1 (set) – Bit 0 is configured as the input capture for timer B.

0 (clear) – Bit 0 is an input-only pin.

THIS PAGE INTENTIONALLY LEFT BLANK

SECTION 5
CORE TIMER

The MC68HC05L4 has a 15-stage ripple counter called the Core Timer (CTIMER). Features of this timer are: Timer Overflow; Power-On Reset (POR); Real Time interrupt (RTI), with four selectable interrupt rates; and a Computer Operating Properly (COP) watchdog timer.

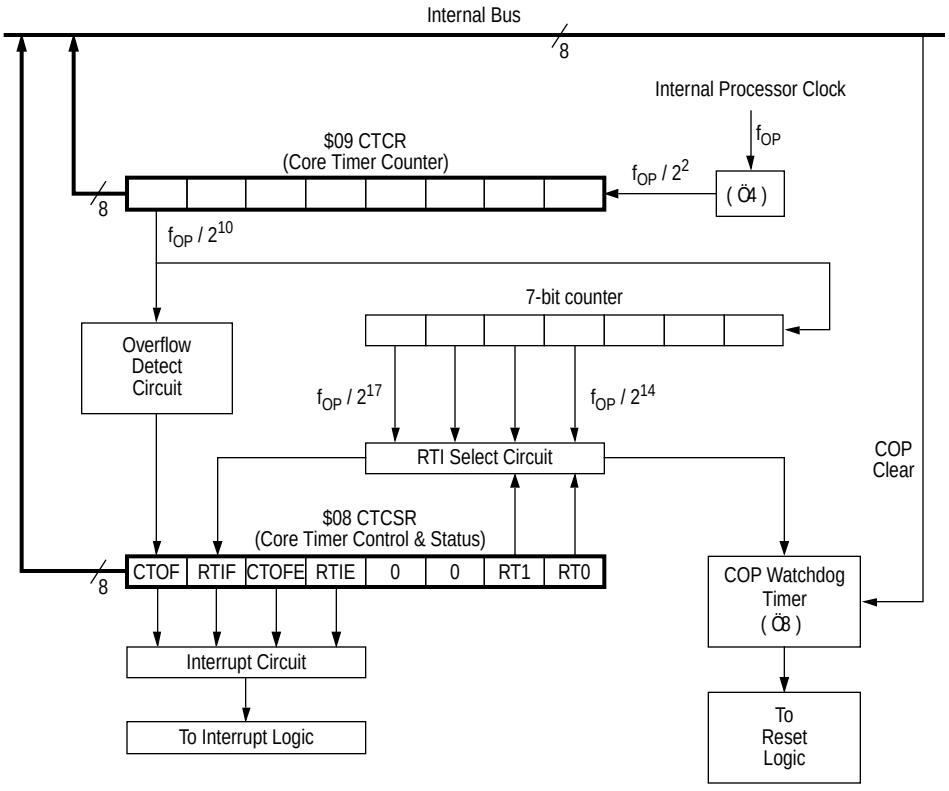


Figure 5-1 Core timer block diagram

As seen in Figure 5-1, the timer is driven by the internal bus clock divided by four with a fixed prescaler. This signal drives an 8-bit ripple counter. The value of this 8-bit ripple counter can be read by the CPU at any time, by accessing the CTIMER Counter Register (CTCR) at address $09. A timer overflow function is implemented on the last stage of this counter, giving a possible interrupt at the rate of $f_{OP}/1024$. (The POR signal (t_{PORL}) is also derived from this register, at $f_{OP}/4064$.) The Counter Register circuit is followed by four more stages, with the resulting clock ($f_{OP}/16384$) driving the Real Time Interrupt circuit. The RTI circuit consists of three divider stages with a 1-of-4 selector. The output of the RTI circuit is further divided by 8 to drive the COP Watchdog Timer circuit. The RTI rate selector bits, and the RTI and CTIMER overflow enable bits and flags, are located in the CTIMER Control and Status Register (CTCSR) at location $08.

CTOF (Core Timer Overflow Flag) is a clearable, read-only status bit and is set when the 8-bit ripple counter rolls over from $FF to $00. A CPU interrupt request will be generated if CTOFE is set. Clearing the CTOF is done by writing a ‘0’ to it. Writing a ‘1’ to CTOF has no effect on the bit’s value. Reset clears CTOF.

When CTOFE (Core Timer Overflow Enable) is set, a CPU interrupt request is generated when the CTOF bit is set. Reset clears CTOFE.

The Core Timer Counter Register (CTCR) is a read-only register that contains the current value of the 8-bit ripple counter at the beginning of the timer chain. This counter is clocked at $f_{OP}/4$ and can be used for various functions including a software input capture. Extended time periods can be attained using the CTIMER overflow function to increment a temporary RAM storage location thereby simulating a 16-bit (or more) counter.

The power-on cycle clears the entire counter chain and begins clocking the counter. After t_{PORL} cycles, the power-on reset circuit is released, which again clears the counter chain and allows the device to come out of reset. At this point, if $\overline{\text{RESET}}$ is not asserted, the timer will start counting up from zero and normal device operation will begin. When $\overline{\text{RESET}}$ is asserted at any time during operation (other than POR), the counter chain will be cleared.

5.1 REAL TIME INTERRUPTS (RTI)

The Real Time Interrupt circuit consists of a three stage divider and a 1-of-4 selector. The clock frequency that drives the RTI circuit is $f_{OP}/2^{14}$ (or $f_{OP}/16384$), with three additional divider stages, giving a maximum interrupt period of 4 seconds at a bus frequency (f_{OP}) of 32kHz. Register details are given in Section 5.3.

5.2 COMPUTER OPERATING PROPERLY (COP) WATCHDOG TIMER

The COP watchdog timer function is implemented by taking the output of the RTI circuit and further dividing it by eight, as shown in Figure 5-1. Note that the minimum COP timeout period is seven times the RTI period. This is because the COP will be cleared asynchronously with respect

to the value in the Core Timer counter register/RTI divider, hence the actual COP timeout period will vary between 7x and 8x the RTI period.

The COP function is a mask option, selectable via the Mask Option register during device manufacture.

If the COP circuit times out, an internal reset is generated and the normal reset vector is fetched. Preventing a COP timeout is done by writing a ‘0’ to bit 0 of address $0FF0. When the COP is cleared, only the final divide-by-eight stage is cleared (see Figure 5-1).

5.3 CORE TIMER REGISTERS

5.3.1 Core Timer Control and Status Register (CTCSR)

Address	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	State on Reset
$0008	CTOF	RTIF	CTOFE	RTIE	0	0	RT1	RT0	0000 0011

5.3.1.1 CTOF — Core Timer Overflow

1 (set) – Core Timer Overflow has occurred.

0 (clear) – No Core Timer Overflow interrupt has been generated.

This bit is set when the Core Timer Counter register rolls over from $FF to $00; an interrupt request will be generated if CTOFE is set. When set, the bit may be cleared by writing a ‘0’ to it.

5.3.1.2 RTIF — Real Time Interrupt Flag

1 (set) – A Real Time interrupt has occurred.

0 (clear) – No Real Time interrupt has been generated.

This bit is set when the output of the chosen stage becomes active; an interrupt request will be generated if RTIE is set. When set, the bit may be cleared by writing a ‘0’ to it.

5.3.1.3 CTOFE — Core Timer Overflow Enable

1 (set) – Core Timer Overflow interrupt is enabled.

0 (clear) – Core Timer Overflow interrupt is disabled.

Setting this bit enables the Core Timer Overflow Interrupt. A CPU interrupt request will then be generated whenever the CTOF bit becomes set. Clearing this bit disables the Core Timer Overflow interrupt capability.

5.3.1.4 RTIE — Real Time Interrupt Enable

1 (set) – Real Time interrupt is enabled.

0 (clear) – Real Time interrupt is disabled.

Setting this bit enables the Real Time Interrupt. A CPU interrupt request will then be generated whenever the RTIF bit becomes set. Clearing this bit disables the Real Time interrupt capability.

5.3.1.5 RT1:RT0 — Real Time Interrupt Rate Select

These two bits select one of four taps from the Real Time Interrupt circuitry. Reset sets both RT0 and RT1 to one, selecting the lowest periodic rate and therefore the maximum time in which to alter them if necessary. The COP reset times are also determined by these two bits. Care should be taken when altering RT0 and RT1 if a timeout is imminent, or the timeout period is uncertain. If the selected tap is modified during a cycle in which the counter is switching, an RTIF could be missed or an additional one could be generated. To avoid problems, the COP should be cleared before changing the RTI taps. SeeTable 5-1 for some example RTI periods.

Table 5-1 Example RTI periods

			Bus frequency f_{OP} = 2 MHz	
RT1	**RT0**	**Division Ratio**	**RTI Period**	**Minimum COP Period**
0	0	2^{14}	8.2ms	57.3ms
0	1	2^{15}	16.4ms	114.7ms
1	0	2^{16}	32.8ms	229.4ms
1	1	2^{17}	65.5ms	458.8ms

5.3.2 Core Timer Counter Register (CTCR)

Address	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	State on Reset
$0009									0000 0000

The Core Timer Counter register is a read-only register, which contains the current value of the 8-bit ripple counter at the beginning of the timer chain.

Reset clears this register.

5.4 CORE TIMER DURING WAIT

The CPU clock halts during the WAIT mode, but the core timer remains active. If the CTIMER interrupts are enabled, then a CTIMER interrupt will cause the processor to exit the WAIT mode.

5.5 CORE TIMER DURING STOP

The timer is cleared when going into STOP mode. When STOP is exited by an external interrupt or an external reset, the internal oscillator will restart, followed by an internal processor stabilisation delay (t_{PORL}). The timer is then cleared and operation resumes.

THIS PAGE INTENTIONALLY LEFT BLANK

SECTION 6
PROGRAMMABLE TIMERS A AND B

Besides the Core Timer the MC68HC05L4 has two 16-bit programmable timers, namely timer A and timer B. These timers are identical in operation and therefore, for the purposes of this manual, only timer A shall be described. Any references to address locations for timer B will be shown in parentheses after the timer A address.

The programmable timer consists of a 16-bit read-only free-running counter, with a fixed divide-by-four prescaler, plus the input capture/output compare circuitry. Selected input edges cause the current counter value to be latched into a 16-bit input capture register so that software can later read this value to determine when the edge occurred. When the free running counter value matches the value in the output compare registers, the programmed pin action takes place. Refer to Figure 6-1 for a block diagram of the timer. The input capture and output compare functions on timer B can only be enabled by setting bit 0 and bit 1 of the Port D control register as described in Section 4.5.6.

The timer has a 16-bit architecture, hence each specific functional segment is represented by two 8-bit registers. These registers contain the high and low byte of that functional segment. Accessing the low byte of a specific timer function allows full control of that function; however, an access of the high byte inhibits that specific timer function until the low byte is also accessed.

Note: The I-bit in the CCR should be set while manipulating both the high and low byte register of a specific timer function to ensure that an interrupt does not occur.

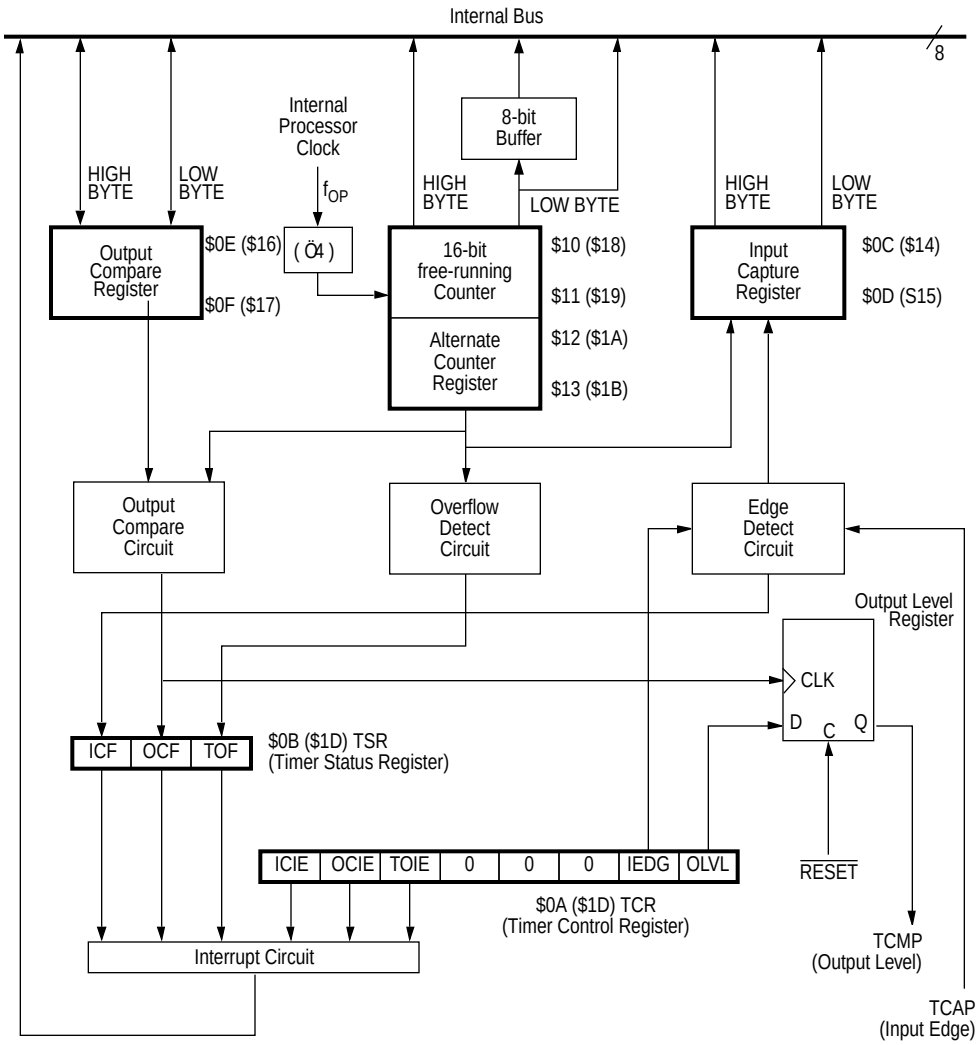


Figure 6-1 16-bit programmable timer block diagram

6.1 COUNTER

The key element in the programmable timer is a 16-bit, free-running counter, or counter register, preceded by a prescaler that divides the internal processor clock by four. The prescaler gives the timer a resolution of 2µs if the internal bus clock is 2MHz. The counter is incremented during the low portion of the internal bus clock. Software can read the counter at any time without affecting its value.

6.1.1 Counter High Register – $0010 ($0018)
Counter Low Register – $0011 ($0019)
Alternate Counter High Register – $0012 ($001A)
Alternate Counter Low Register – $0013 ($001B)

Address	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	State on Reset
$10 ($18)	(bit 15)							(bit 8)	$FF
$11 ($19)									$FC
$12 ($1A)	(bit 15)							(bit 8)	$FF
$13 ($1B)									$FC

The double-byte, free-running counter can be read from either of two locations, the counter register at $10 – $11 ($18 – $19) or the alternate counter register at $12 – $13 ($1A – $1B). A read from only the less significant byte (LSB) of the free-running counter, $11 or $13 ($19 or $1B), receives the count value at the time of the read. If a read of the free-running counter or alternate counter register first addresses the more significant byte (MSB), $10 or $12 ($18 or $1A), the LSB is transferred to a buffer. This buffer value remains fixed after the first MSB read, even if the user reads the MSB several times. This buffer is accessed when reading the free-running counter or alternate counter register LSB and thus completes a read sequence of the total counter value. In reading either the free-running counter or alternate counter register, if the MSB is read, the LSB must also be read to complete the sequence. If the timer overflow flag (TOF) is set when the counter register LSB is read, then a read of the TSR will clear the flag.

The Alternate Counter register differs from the Counter register only in that a read of the LSB does not clear TOF. Therefore, to avoid the possibility of missing timer overflow interrupts due to clearing of TOF, the Alternate Counter register should be used where this is a critical issue.

The free-running counter is set to $FFFC during reset and is always a read-only register. During a power-on reset, the counter is also preset to $FFFC and begins running after the oscillator start-up delay. Because the free-running counter is 16 bits preceded by a fixed divide-by-four prescaler, the value in the free-running counter repeats every 262144 internal bus clock cycles. TOF is set when the counter overflows (from $FFFF to $0000); this will cause an interrupt if TOIE is set.

6.1.1.1 Bits 8 – 15 — MSB of Counter/Alternate Counter Register

A read of only the more significant byte (MSB) transfers the LSB to a buffer, which remains fixed after the first MSB read, until the LSB is also read.

6.1.1.2 Bits 0 – 7 — LSB of Counter/Alternate Counter Register

A read of only the less significant byte (LSB) receives the count value at the time of reading.

6.2 TIMER FUNCTIONS

The 16-bit programmable timer is monitored and controlled by a group of ten registers, full details of which are contained in the following paragraphs. An explanation of the timer functions is also given.

6.2.1 Timer Control Register – TCRA (TCRB)

The Timer Control register at location $0A ($12) is used to enable the Input Capture (ICIE), Output Compare (OCIE), and Timer Overflow (TOIE) interrupt enable functions as well as selecting input edge sensitivity (IEDG) and output level polarity (OLVL).

Address	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	State on Reset
$0A ($12)	ICIE	OCIE	TOIE				IEDG	OLVL	000 - - - U0

6.2.1.1 ICIE — Input Capture Interrupt Enable

1 (set) – Input Capture interrupt enabled.

0 (clear) – Input Capture interrupt disabled.

6.2.1.2 OCIE — Output Compare Interrupt Enable

1 (set) – Output Compare interrupt enabled.

0 (clear) – Output Compare interrupt disabled.

6.2.1.3 TOIE — Timer Overflow Interrupt Enable

1 (set) – Timer Overflow interrupt enabled.

0 (clear) – Timer Overflow interrupt disabled.

6.2.1.4 IEDG — Input Edge

1 (set) – TCAP is positive-going edge sensitive.

0 (clear) – TCAP is negative-going edge sensitive.

When IEDG is set, a positive-going edge on the TCAP pin will trigger a transfer of the free-running counter value to the input capture register. When clear, a negative-going edge triggers the transfer.

6.2.1.5 OLVL — Output Level

1 (set) – A high output level will appear on the TCMP pin.

0 (clear) – A low output level will appear on the TCMP pin.

When OLVL is set, a high output level will be clocked into the output level register by the next successful output compare, and will appear on the TCMP pin. When clear, it will be a low level that will appear on the TCMP pin.

6.2.2 Timer Status Register – TSRA (TSRB)

The Timer Status register ($13) contains the status bits for the above three interrupt conditions — ICF, OCF, TOF.

Accessing the timer status register satisfies the first condition required to clear the status bits. The remaining step is to access the register corresponding to the status bit.

Address	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	State on Reset
$0B ($1D)	ICF	OCF	TOF						uuu- - - -

6.2.2.1 ICF — Input Capture Flag

1 (set) – A valid input capture has occurred.

0 (clear) – No input capture has occurred.

This bit is set when the selected polarity of edge is detected by the input capture edge detector; an input capture interrupt will be generated, if ICIE is set. ICF is cleared by reading the TSR and then the Input Capture Low register at $0D ($15).

6.2.2.2 OCF — Output Compare Flag

1 (set) – A valid output compare has occurred.

0 (clear) – No output compare has occurred.

This bit is set when the output compare register contents match those of the free-running counter; an output compare interrupt will be generated, if OCIE is set. OCF is cleared by reading the TSR and then the Output Compare Low register at $0F ($17).

6.2.2.3 TOF — Timer Overflow Flag

1 (set) – Timer Overflow has occurred.

0 (clear) – No timer overflow has occurred.

This bit is set when the free-running counter overflows from $FFFF to $0000; a timer overflow interrupt will occur, if TOIE is set. TOF is cleared by reading the TSR and the counter low register, $11 ($19).

When using the timer overflow function and reading the free-running counter at random times to measure an elapsed time, a problem may occur whereby the timer overflow flag is unintentionally cleared if:

- 1) the timer status register is read or written when TOF is set and
- 2) the LSB of the free-running counter is read, but not for the purpose of servicing the flag.

Reading the alternate counter register instead of the counter register will avoid this potential problem.

6.2.3 Input Capture Function

‘Input Capture’ is a technique whereby an external signal (connected to the TCAP pin) is used to trigger a read of the free-running counter. In this way it is possible to relate the timing of an external signal to the internal counter value, and hence to elapsed time.

6.2.4 Input Capture High Register – $000C ($0014)
Input Capture Low Register – $000D ($0015)

Address	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	State on Reset
$0C ($14)	(bit 15)							(bit 8)	Unaffected
$0D ($15)									Unaffected

The two 8-bit registers that make up the 16-bit input capture register are read-only, and are used to latch the value of the free-running counter after the input capture edge detector senses a valid transition. The level transition that triggers the counter transfer is defined by the input edge bit (IEDG). The most significant 8 bits are stored in the Input Capture High register at $0C ($14), the least significant in the Input Capture Low register at $0D ($15).

The result obtained from an input capture will be one greater than the value of the free-running counter on the rising edge of the internal bus clock preceding the external transition. This delay is required for internal synchronisation. Resolution is one count of the free-running counter, which is four internal bus clock cycles. The free-running counter contents are transferred to the input capture register on each valid signal transition whether the input capture flag (ICF) is set or clear. The input capture register always contains the free-running counter value that corresponds to the most recent input capture. After a read of the input capture register MSB ($0C ($14)), the counter transfer is inhibited until the LSB ($0D ($15)) is also read. This characteristic causes the time used in the input capture software routine and its interaction with the main program to determine the minimum pulse period. A read of the input capture register LSB ($0D ($15)) does not inhibit the free-running counter transfer since the two actions occur on opposite edges of the internal bus clock.

Reset does not affect the contents of the Input Capture register, except when exiting STOP mode.

6.2.5 Output Compare Function

‘Output Compare’ is a technique that may be used, for example, to generate an output waveform, or to signal when a specific time period has elapsed, by presetting the Output Compare register to the appropriate value.

6.2.6 Output Compare High Register – $000E ($0016)
Output Compare Low Register – $000F ($0017)

Address	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	State on Reset
$0E ($16)	(bit 15)							(bit 8)	Unaffected
$0F ($17)									Unaffected

The 16-bit output compare register is made up of two 8-bit registers at locations $0E ($16) (MSB) and $0F ($17) (LSB). The contents of the output compare register are continually compared with the contents of the free-running counter and, if a match is found, the output compare flag (OCF) in the Timer Status register is set and the output level (OLVL) bit clocked to the output level register. The output compare register values and the output level bit should be changed after each successful comparison to establish a new elapsed timeout. An interrupt can also accompany a successful output compare provided the corresponding interrupt enable bit (OCIE) is set. (The free-running counter is updated every four internal bus clock cycles.)

After a processor write cycle to the output compare register containing the MSB ($0E ($16)), the output compare function is inhibited until the LSB ($0F ($17)) is also written. The user must write both bytes (locations) if the MSB is written first. A write made only to the LSB will not inhibit the compare function. The processor can write to either byte of the output compare register without affecting the other byte. The output level (OLVL) bit is clocked to the output level register whether the output compare flag (OCF) is set or clear. The minimum time required to update the output compare register is a function of the program rather than the internal hardware. Because the output compare flag and the output compare register are not defined at power on, and not affected by reset, care must be taken when initialising output compare functions with software. The following procedure is recommended:

- 1) write to Output Compare High to inhibit further compares;
- 2) read the Timer Status register to clear OCF (if set);
- 3) write to Output Compare Low to enable the output compare function.

All bits of the Output Compare register are readable and writable and are not altered by the timer hardware or reset. If the compare function is not needed, the two bytes of the output compare register can be used as storage locations.

6.3 TIMER DURING WAIT MODE

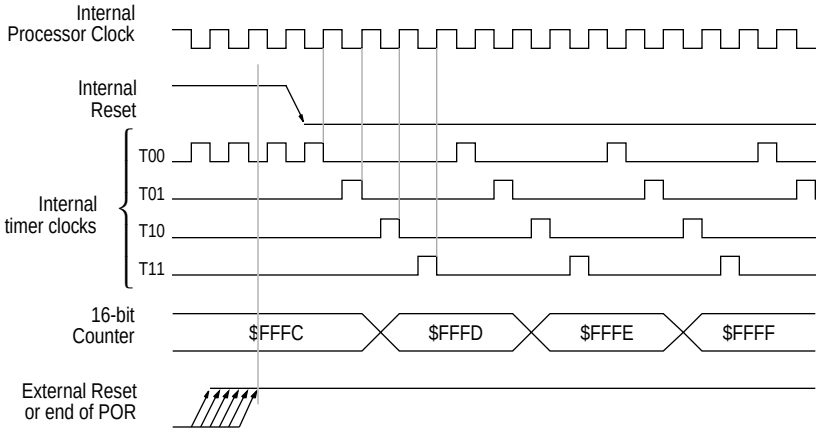
In WAIT mode all CPU action is suspended. Timers A and B stop and hold the last count value. On exit from WAIT mode, the timers will resume counting from where they had stopped. If $\overline{\text{RESET}}$ is used to exit WAIT mode, the counters are forced to $FFFC.

6.4 TIMER DURING STOP MODE

In the STOP mode all MCU clocks are stopped, hence the timers stop counting. If STOP is exited by an interrupt the counters retain the last count value. If the device is reset, then the counters are forced to $FFFC. During STOP, if at least one valid input capture edge occurs at the TCAP pin, the input capture detect circuit is armed. This does not set any timer flags nor wake up the MCU. When the MCU does wake up, however, there is an active input capture flag and data from the first valid edge that occurred during the STOP period. If the device is reset to exit STOP mode, then no input capture flag or data remains, even if a valid input capture edge occurred.

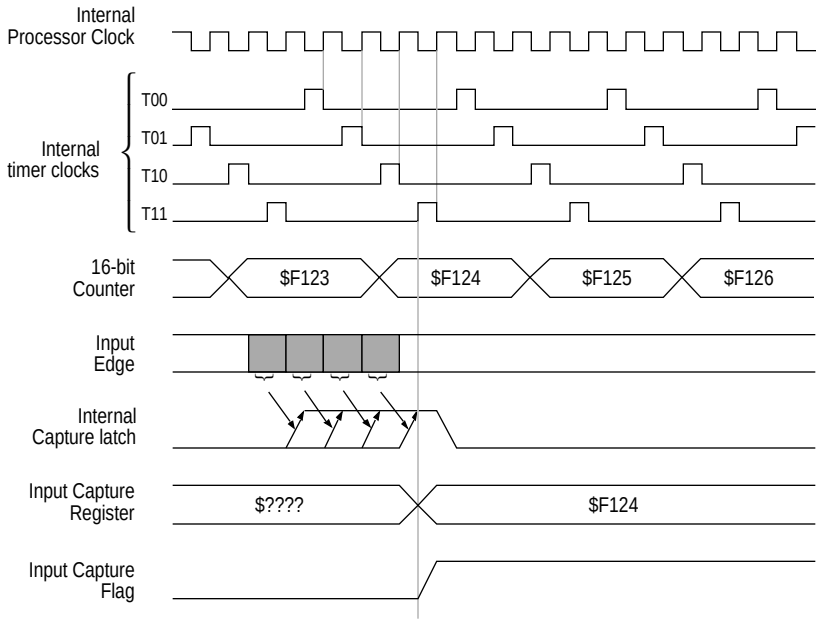
6.5 TIMER STATE DIAGRAMS

The relationships between the internal clock signals, the counter contents and the status of the flag bits are shown in the following diagrams. It should be noted that the signals labelled ‘internal’ (Processor Clock, Timer Clocks and Reset) are not available to the user.



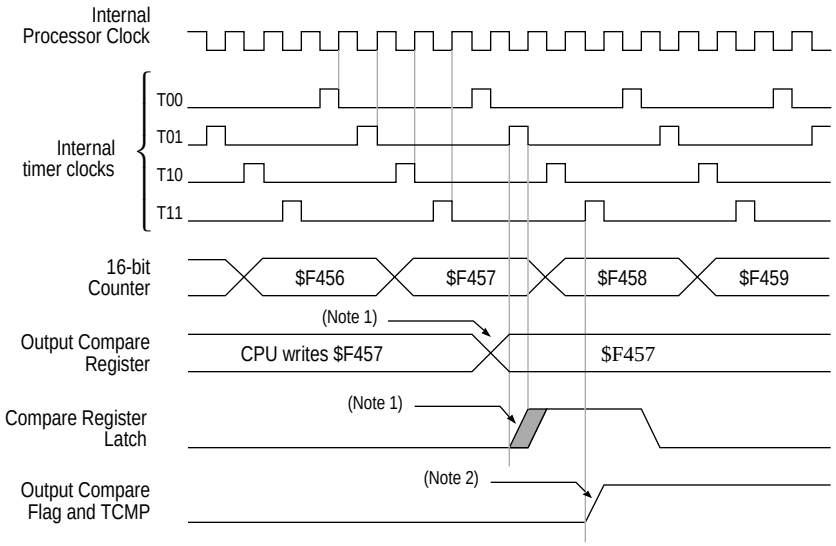
Note: The Counter and Timer Control Registers are the only ones affected by power-on or external reset.

Figure 6-2 Timer state timing diagram for reset



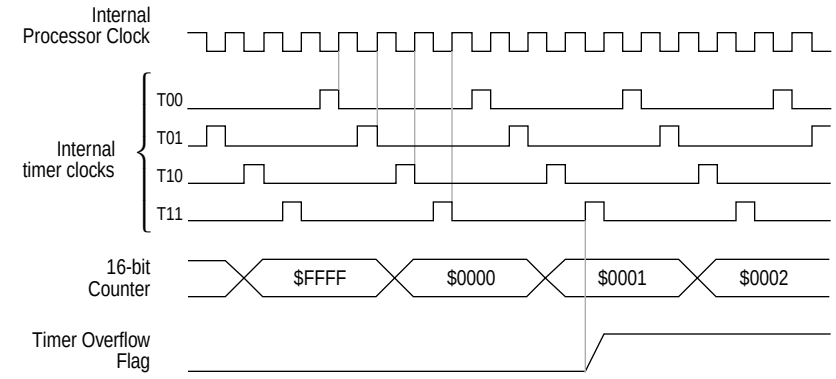
Note: If the input edge occurs in the shaded area from one timer state T10 to the next timer state T10, then the input capture flag will be set during the next T11 state.

Figure 6-3 Timer state timing diagram for input capture



Note: (1) The CPU write to the compare registers may take place at any time, but a compare only occurs at timer state T01. Thus a four cycle difference may exist between the write to the compare register and the actual compare.
(2) The Output Compare Flag is set at the timer state T11 that follows the comparison match ($F457 in this example).

Figure 6-4 Timer state timing diagram for output compare



Note: The Timer Overflow Flag is set at timer state T11 (transition of counter from $FFFF to $0000). It is cleared by a read of the Timer Status Register during the internal processor clock high time, followed by a read of the counter low register.

Figure 6-5 Timer state timing diagram for timer overflow

THIS PAGE INTENTIONALLY LEFT BLANK

SECTION 7
LIQUID CRYSTAL DISPLAY DRIVER MODULE

The LCD driver module on the MC68HC05L4 can be configured with up to 24 frontplane drivers and up to 4 backplane drivers. This allows a maximum of 96 LCD segments to be driven. Each segment is controlled by a corresponding bit in the LCD RAM. At reset or on power-up, the drivers are configured in the default duplex mode, 1/2 bias with 2 backplanes and 22 frontplanes. At this stage all Port D pins that are shared with the LCD subsystem are configured as input only pins. Also at power-up or reset the ON/OFF control for the display, the DISON bit in the LCD/Key Control (LCD) register, is cleared thus disabling the LCD drivers. Figure 7-1 shows a block diagram of the LCD system.

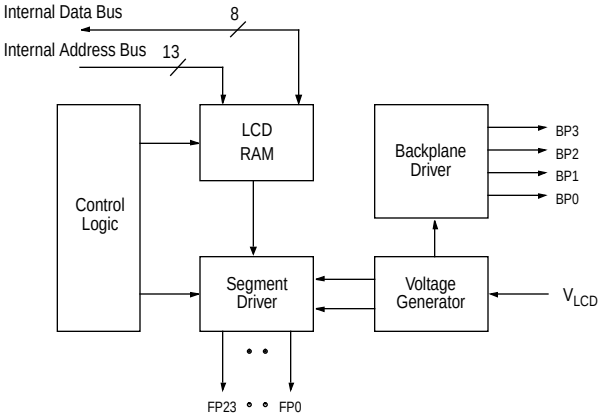


Figure 7-1 LCD block diagram

7.1 LCD DISPLAY RAM

Data to be displayed on the LCD must be written into the LCD RAM. The LCD RAM is comprised of 12 bytes of RAM (in the MC68HC05L4's memory map) at $0030-$003B or $0130-$013B. The 96 bits in the LCD RAM correspond to the 96 segments that can be driven by the frontplane/backplane drivers. Table 7-1 shows how the LCD RAM is organised. Writing a "1" to a

given location will result in the corresponding display segment being activated when the DISON bit is set. The LCD RAM is a dual port RAM that interfaces with the internal address and data buses of the MCU. It is possible to read from LCD RAM locations for scrolling purposes. When DISON = 0, the LCD RAM can be used as main on-chip RAM.

Note: The same 12 bytes of LCD RAM can be accessed at addresses $0030-$003B or $0130-$013B.

Table 7-1 LCD RAM organisation

LCD RAM Address	Data							
	7	6	5	4	3	2	1	0
$30/$130	FP1–BP3	FP1–BP2	FP1–BP1	FP1–BP0	FP0–BP3	FP0–BP2	FP0–BP1	FP0–BP0
$31/$131	FP3–BP3	FP3–BP2	FP3–BP1	FP3–BP0	FP2–BP3	FP2–BP2	FP2–BP1	FP2–BP0
$32/$132	FP5–BP3	FP5–BP2	FP5–BP1	FP5–BP0	FP4–BP3	FP4–BP2	FP4–BP1	FP4–BP0
$33/$133	FP7–BP3	FP7–BP2	FP7–BP1	FP7–BP0	FP6–BP3	FP6–BP2	FP6–BP1	FP6–BP0
$34/$134	FP9–BP3	FP9–BP2	FP9–BP1	FP9–BP0	FP8–BP3	FP8–BP2	FP8–BP1	FP8–BP0
$35/$135	FP11–BP3	FP11–BP2	FP11–BP1	FP11–BP0	FP10–BP3	FP10–BP2	FP10–BP1	FP10–BP0
$36/$136	FP13–BP3	FP13–BP2	FP13–BP1	FP13–BP0	FP12–BP3	FP12–BP2	FP12–BP1	FP12–BP0
$37/$137	FP15–BP3	FP15–BP2	FP15–BP1	FP15–BP0	FP14–BP3	FP14–BP2	FP14–BP1	FP14–BP0
$38/$138	FP17–BP3	FP17–BP2	FP17–BP1	FP17–BP0	FP16–BP3	FP16–BP2	FP16–BP1	FP16–BP0
$39/$139	FP19–BP3	FP19–BP2	FP19–BP1	FP19–BP0	FP18–BP3	FP18–BP2	FP18–BP1	FP18–BP0
$3A/$13A	FP21–BP3	FP21–BP2	FP21–BP1	FP21–BP0	FP20–BP3	FP20–BP2	FP20–BP1	FP20–BP0
$3B/$13B	FP23–BP3	FP23–BP2	FP23–BP1	FP23–BP0	FP22–BP3	FP22–BP2	FP22–BP1	FP22–BP0

7.2 LCD OPERATION

The LCD driver module can operate in four modes providing different multiplex ratios and number of backplanes as follows:

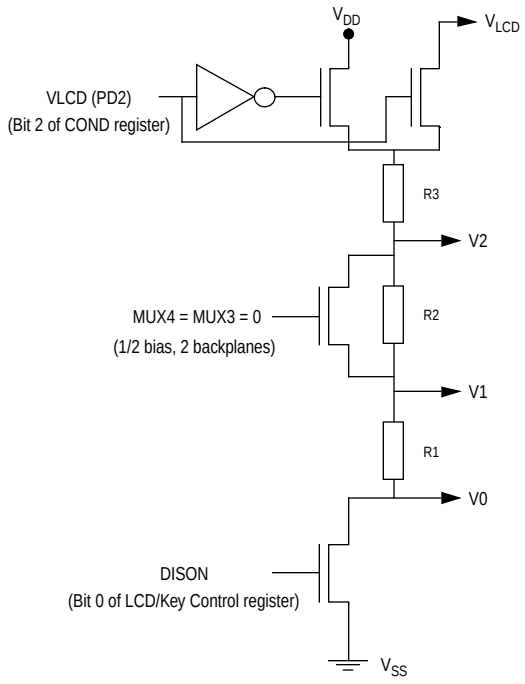
- 1/2 bias, 2 backplanes
- 1/3 bias, 2 backplanes
- 1/3 bias, 3 backplanes
- 1/4 bias, 4 backplanes

The operating mode is selected using the multiplex ratio bits (MUX3 and MUX4) in the LCD/Key control register as shown in Table 7-3.

It is recommended that the DISON bit in the LCD register is not set (to activate the display) until the multiplex rate is selected, to inhibit the LCD drivers. The voltage levels required for the different multiplex rates are generated internally by a resistive divider chain between V_{DD} and V_{SS}. If bit 2

of the Port D Control register (COND) is set, the display voltage is taken from V_{LCD} and not V_{DD} (see note). This particular option can be used when a display requires specific voltage thresholds. The 2-way multiplex with 1/3 bias and the 3 and 4-way multiplex options require four voltage levels, whereas the 2 way multiplex with 1/2 bias needs only three levels. Figure 7-2 shows the resistive divider chain network that is used to produce the various LCD waveforms outlined in Section 7.3.

Note: V_{LCD} may not exceed the positive power supply voltage V_{DD}.



Note: In the 1/2 bias, 2 backplane mode, R2 is shorted out and V2 = V1.

Figure 7-2 Voltage level selection

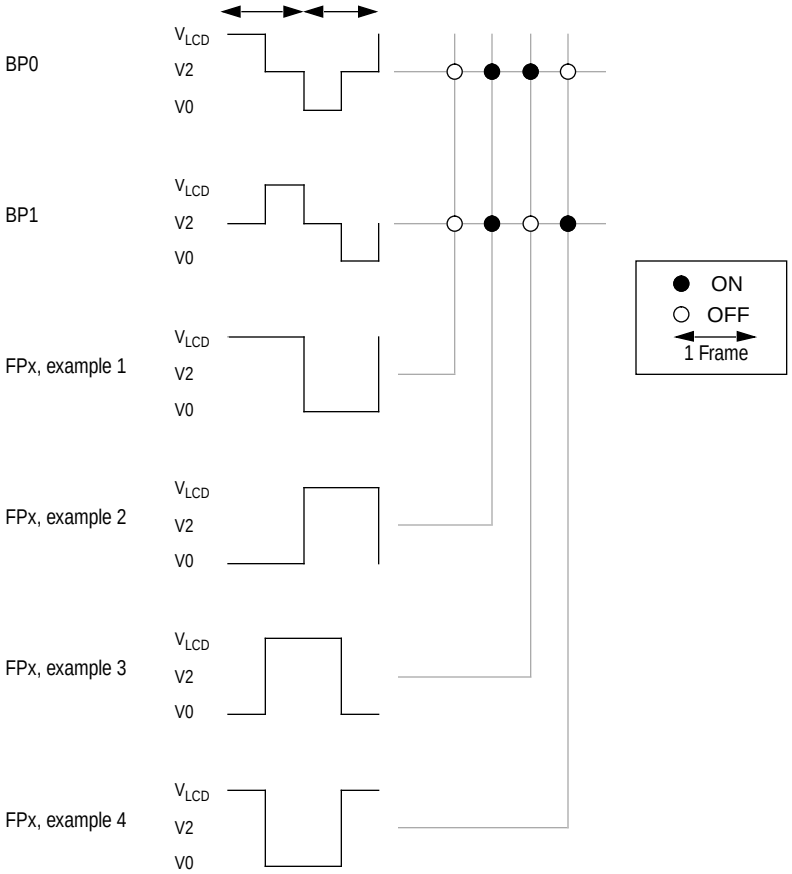
7.3 TIMING SIGNALS AND LCD VOLTAGE WAVEFORMS

The LCD timing signals are all derived from the main system clock; with a bus frequency of 2 MHz (f_{osc} = 4 MHz) the frame rate will be 61 Hz for 2 and 4-way multiplexing and 91 Hz for 3-way multiplexing (see Table 7-3). An extra divide by two stage can be included in the LCD clock generator by setting FDISP in the LCD register. This will result in the frame rate being halved. For example, when 3-way multiplexing is being used, a frame rate of 45.5 Hz instead of 91 Hz can be obtained. See Section 7.4.2.

Figure 7-3 to Figure 7-6 show the backplane waveforms and some examples of frontplane waveforms for each of the operating modes.

The backplane waveforms are continuous and repetitive (every 2 frames); they are fixed within each operating mode and are not affected by the data in the LCD RAM.

The frontplane waveforms are dependent on the LCD segments to be driven as defined in the LCD RAM. Each “on” segment must have a differential driving voltage (BP-FP) applied to it once in each frame; the LCD driver module hardware uses the data in the LCD RAM to construct the frontplane waveform to meet this criterion.



Note: In this mode V1=V2

Figure 7-3 LCD waveform with 2 backplanes, 1/2 Bias

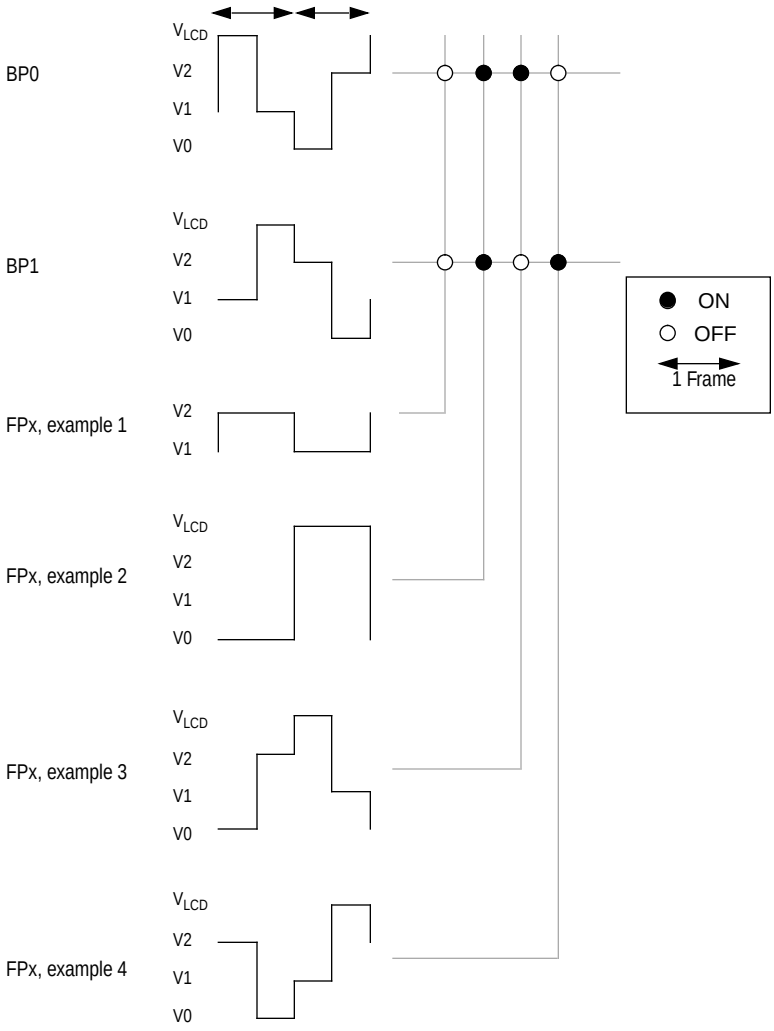


Figure 7-4 LCD waveform with 2 backplanes, 1/3 bias

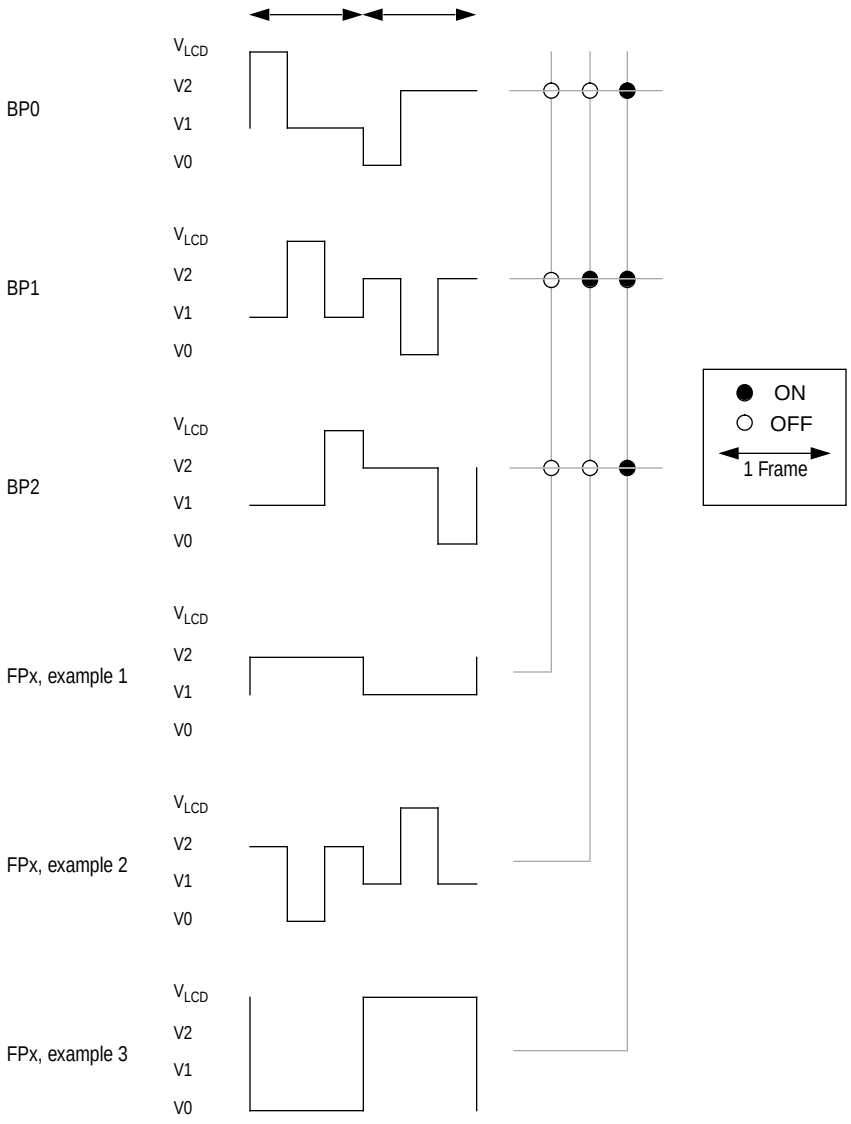


Figure 7-5 LCD waveform with 3 backplanes

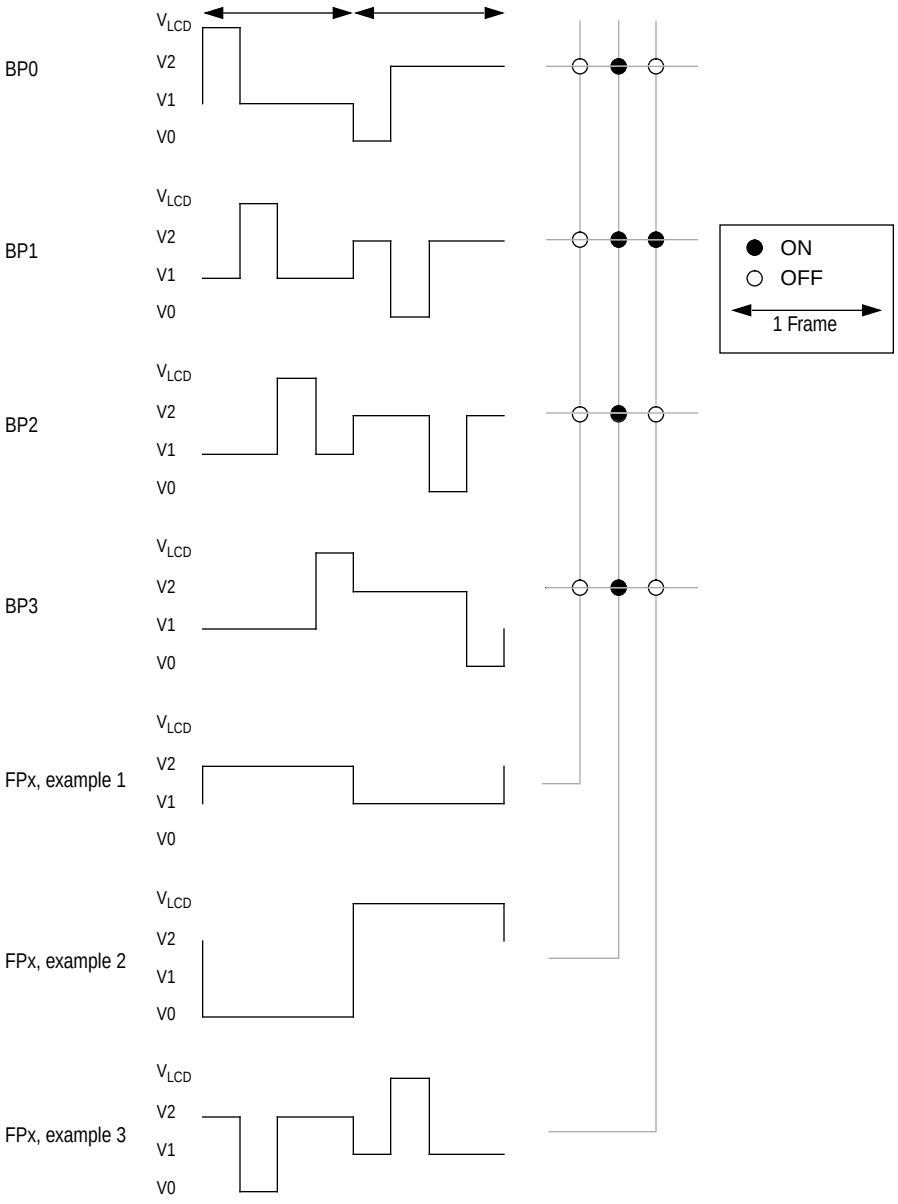


Figure 7-6 LCD waveform with 4 backplanes

7.4 LCD REGISTERS

7.4.1 Port D Control register (COND)

Address	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	State on Reset
$0007	PD7/ BP2	PD6/ BP1	PD5/ FP22	PD4/ FP23	PD3	PD2/ VLCD	PD1/ TCMPB	PD0/ TCAPB	0000 0000

7.4.1.1 VLCD — LCD voltage input

1 (set) – Bit 2 of port D is configured as the voltage input pin for the LCD subsystem.

0 (clear) – Bit 2 is an input-only pin.

7.4.2 LCD/Key control register

Address	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	State on Reset
$001E	KSF	KBIE	FC	LC	FDISP	MUX4	MUX3	DISON	0000 0000

7.4.2.1 KSF — Keyboard Interrupt Status Flag

1 (set) – A keyboard interrupt has been generated.

0 (clear) – No keyboard interrupt has been generated.

The keyboard interrupt status flag is cleared by writing a zero to it. It has no automatic clearing mechanism.

7.4.2.2 KBIE — Keyboard Interrupt Enable

1 (set) – Keyboard interrupt enabled.

0 (clear) – Keyboard interrupt disabled.

7.4.2.3 FC, LC — Fast Charge, Low Current

If the quality of the display is not critical, the LCD block on the device can be put into a low current mode using the FC and LC bits in the LCD register. By selecting the appropriate values for each bit, as shown in Table 7-2, an extra resistor can be added to the divider chain, hence reducing the current consumption. When normal quality is required for the display, the Fast Charge option

should be used which resumes default resistor values for a period of LCD CLK/128 in each time slot. The default value of these resistors is approximately 30 k .

Table 7-2 Resistor chain values

FC	LC	Action
0	0	Default value of approximately 30 k is used.
1	0	No action.
0	1	Resistor value of approximately 200 k is used.
1	1	Resistor values reduced to default for a period of LCD CLK/128 in each time slot.

7.4.2.4 FDISP — Display Frequency

1 (set) – An extra divide by two stage is included in the LCD clock generator to give a reduced frame rate. For example, in the 3-way multiplexing mode, a frame rate of 45.5 Hz instead of 91 Hz can be achieved.

0 (clear) – Default frame rate is used.

7.4.2.5 MUX4, MUX3 — Multiplex ratio

These two bits select the multiplex ratio to be 2, 3 or 4 backplanes.

Table 7-3 Multiplex ratio/backplane selection

MUX4	MUX3	BACKPLANES	BIAS	FREQUENCY
0	0	2	1/2	61 Hz
0	1	3	1/3	91 Hz
1	0	4	1/3	61 Hz
1	1	2	1/3	61 Hz

7.4.2.6 DISON — LCD Display ON/OFF

1 (set) – LCD is ON.

0 (clear) – LCD is OFF.

7.5 LCD DURING WAIT MODE

The LCD does not function during WAIT mode.

SECTION 8
RESETS AND INTERRUPTS

8.1 RESETS

The MC68HC05L4 can be reset in four ways: by the initial power-on reset function, by an active low input to the $\overline{\text{RESET}}$ pin, by an opcode fetch from an illegal address and by a COP watchdog timer reset, if the watchdog timer is enabled.

8.1.1 Power-on Reset

A power-on reset occurs when a positive transition is detected on VDD. The power-on reset function is strictly for power turn-on conditions and should not be used to detect drops in the power supply voltage. The power-on circuitry provides a stabilisation delay (t_{PORL}) from when the oscillator becomes active. If the external $\overline{\text{RESET}}$ pin is low at the end of this delay then the processor remains in the reset state until $\overline{\text{RESET}}$ goes high. The user must ensure that the voltage on VDD has risen to a point where the MCU can operate properly by the time t_{PORL} has elapsed. If there is doubt, the external $\overline{\text{RESET}}$ pin should remain low until the voltage on VDD has reached the specified minimum operating voltage. This may be accomplished by connecting an external RC-circuit to this pin to generate a power-on reset (POR). In this case, the time constant must be great enough (at least 100 ms) to allow the oscillator circuit to stabilise.

At power-up, the $\overline{\text{RESET}}$ pin is pulled active low by an internal open drain N-channel device driven from the power-on reset signal. This pin can be used as a reset output.

8.1.2 $\overline{\text{RESET}}$ Pin

When the oscillator is running in a stable state, the MCU is reset when a logic zero is applied to the $\overline{\text{RESET}}$ input for a minimum period of 1.5 machine cycles (t_{CYC}). This pin contains an internal Schmitt Trigger as part of its input to improve noise immunity. When a reset condition occurs internally, i.e. from the COP watchdog, the $\overline{\text{RESET}}$ pin provides an active-low open drain output signal which may be used to reset external hardware.

8.1.3 Illegal Address Reset

When an opcode fetch occurs from an address that is not part of the RAM ($0050 – $012F) or of the ROM ($2000 – $3E5F and $3FF0– $3FFF), then the device is automatically reset.

8.1.4 Computer Operating Properly (COP) Reset

The MCU contains a watchdog timer that automatically times out if not reset (cleared) within a specific time by a program reset sequence.

Note: COP timeout is prevented by periodically writing a ‘0’ to bit 0 of address $3FF0.

If the COP watchdog timer is allowed to timeout, an internal reset is generated to reset the MCU. Because the internal reset signal is used, the MCU comes out of a COP reset in the same operating mode it was in when the COP timeout was generated.

The COP reset function is enabled or disabled by a mask option.

Refer to Section 5.2 for more information on the COP watchdog timer.

8.2 INTERRUPTS

The MCU can be interrupted by six different sources, five maskable hardware interrupts and one non-maskable software interrupt:

- External signal on the $\overline{\text{IRQ}}$ pin
- Keyboard interrupt
- Core timer
- 16-bit programmable timer A
- 16-bit programmable timer B
- Software Interrupt Instruction (SWI)

Interrupts cause the processor to save the register contents on the stack and to set the interrupt mask (I-bit) to prevent additional interrupts. The RTI instruction (ReTurn from Interrupt) causes the register contents to be recovered from the stack and normal processing to resume.

Unlike reset, hardware interrupts do not cause the current instruction execution to be halted, but are considered pending until the current instruction is complete. The current instruction is the one already fetched and being operated on. When the current instruction is complete, the processor checks all pending hardware interrupts. If interrupts are not masked (CCR I-bit clear) and the

corresponding interrupt enable bit is set, the processor proceeds with interrupt processing; otherwise, the next instruction is fetched and executed.

If both an external interrupt and a timer interrupt are pending after an instruction execution, the external interrupt is serviced first.

Table 8-1 shows the relative priority of all the possible interrupt sources. Figure 8-1 shows the Interrupt Processing flow.

Table 8-1 Interrupt priorities

Source	Register	Flags	Vector Address	Priority
Reset	—	—	$3FFE, $3FFF	highest
Software Interrupt (SWI)	—	—	$3FFC, $3FFD	
External Interrupt ($\overline{IRQ}$) / Keyboard Interrupt	— LCD	— KSF	$3FFA, $3FFB	
Core Timer	CTCSR	CTOF, RTIF	$3FF8, $3FF9	
Programmable Timer A	TSRA	ICF, OCF, TOF	$3FF6, $3FF7	
Programmable Timer B	TSRB	ICF, OCF, TOF	$3FF4, $3FF5	lowest

8.2.1 Non-Maskable Software Interrupt (SWI)

The software interrupt (SWI) is an executable instruction and a non-maskable interrupt: it is executed regardless of the state of the I-bit in the CCR. If the I-bit is zero (interrupts enabled), SWI is executed after interrupts that were pending when the SWI was fetched, but before interrupts generated after the SWI was fetched. The SWI interrupt service routine address is specified by the contents of memory locations $3FFC and $3FFD.

8.2.2 Maskable Hardware Interrupts

If the interrupt mask bit (I-bit) of the CCR is set, all maskable interrupts (internal and external) are masked. Clearing the I-bit allows interrupt processing to occur.

Note: The internal interrupt latch is cleared in the first part of the interrupt service routine; therefore, one external interrupt pulse could be latched and serviced as soon as the I-bit is cleared.

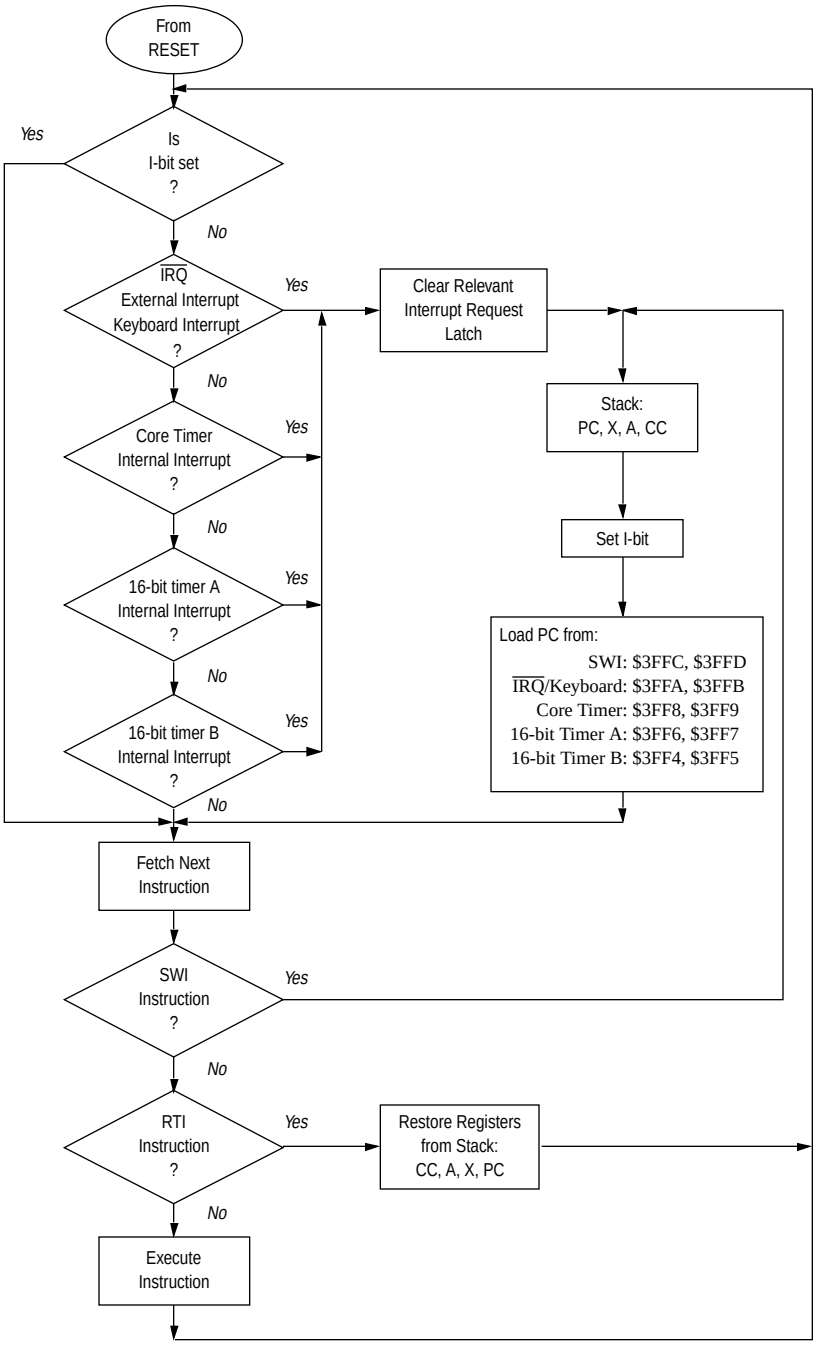


Figure 8-1 Interrupt flow chart

8.2.2.1 External Interrupt ($\overline{\text{IRQ}}$ or Keyboard)

These external interrupt sources will vector to the same interrupt service routine, whose start address is contained in memory locations $3FFA and $3FFB. Further details on the keyboard interrupt facility can be found in Section 4.3.

8.2.2.2 Real Time and Core Timer (CTIMER) Interrupts

There are two core timer interrupt flags that cause a CTIMER interrupt whenever an interrupt is enabled and its flag becomes set (RTIF and CTOF). The interrupt flags and enable bits are located in the CTIMER control and status register (CTCSR). These interrupts vector to the same interrupt service routine, whose start address is contained in memory locations $3FF8 and $3FF9 (see Section 5.3.1 and Figure 5-1).

To make use of the real time interrupt the RTIE bit must first be set. The RTIF bit will then be set after the specified number of counts.

To make use of the core timer overflow interrupt, the CTOFE bit must first be set. The CTOF bit will then be set when the core timer counter register overflows from $FF to $00.

8.2.2.3 Programmable 16-bit Timer A Interrupt

There are three different timer interrupt flags (ICF, OCF, TOF) that cause a timer interrupt whenever they are set and enabled. The timer interrupt enable bits (ICIE, OCIE, TOIE) are located in the timer control register (TCRA) and the timer interrupt flag is located in the timer status register (TSRA). All three interrupts vector to the same service routine, whose start address is contained in memory locations $3FF6 and $3FF7.

8.2.2.4 Programmable 16-bit Timer B Interrupt

There are three different timer interrupt flags (ICF, OCF, TOF) that cause a timer interrupt whenever they are set and enabled. The timer interrupt enable bits (ICIE, OCIE, TOIE) are located in the timer control register (TCRB) and the timer interrupt flag is located in the timer status register (TSRB). All three interrupts vector to the same service routine, whose start address is contained in memory locations $3FF4 and $3FF5.

8.2.3 Hardware Controlled Interrupt Sequence

The following three functions (RESET, STOP, and WAIT) are not in the strictest sense interrupts. However, they are acted upon in a similar manner. Flowcharts for STOP and WAIT are shown in Figure 2-5.

RESET: A reset condition causes the program to vector to its starting address, which is contained in memory locations $3FFE (MSB) and $3FFF (LSB). The I-bit in the condition code register is also set, to disable maskable interrupts.

STOP: The STOP instruction causes the oscillator to be turned off and the processor to ‘sleep’ until an external interrupt ($\overline{IRQ}$) or a keyboard interrupt occurs, or the device is reset.

WAIT: The WAIT instruction causes all processor clocks to stop, but leaves the core timer clock running. This ‘rest’ state of the processor can be cleared by reset, an external interrupt ($\overline{IRQ}$), a keyboard interrupt, or a core timer interrupt. There are no special WAIT vectors for these interrupts.

SECTION 9
CPU CORE AND INSTRUCTION SET

This section provides a description of the CPU core registers, the instruction set and the addressing modes of the MC68HC05L4.

9.1 REGISTERS

The MCU contains five registers, as shown in the programming model of Figure 9-1. The interrupt stacking order is shown in Figure 9-2.

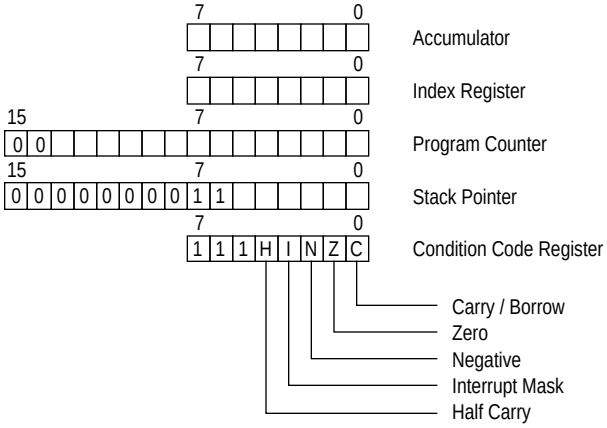


Figure 9-1 Programming model

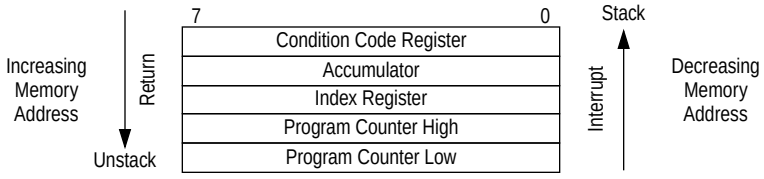


Figure 9-2 Stacking order

9.1.1 Accumulator (A)

The accumulator is a general purpose 8-bit register used to hold operands and results of arithmetic calculations or data manipulations.

9.1.2 Index Register (X)

The index register is an 8-bit register, which can contain the indexed addressing value used to create an effective address. The index register may also be used as a temporary storage area.

9.1.3 Program Counter (PC)

The program counter is a 16-bit register, which contains the address of the next byte to be fetched. Although the M68HC05 CPU core can address 64 kbytes of memory, the actual address range of the MC68HC05L4 is limited to 16 kbytes. The two most significant bits of the Program Counter are therefore not used and are permanently set to zero.

9.1.4 Stack Pointer (SP)

The stack pointer is a 16-bit register, which contains the address of the next free location on the stack. During an MCU reset or the reset stack pointer (RSP) instruction, the stack pointer is set to location $00FF. The stack pointer is then decremented as data is pushed onto the stack and incremented as data is pulled from the stack.

When accessing memory, the ten most significant bits are permanently set to 0000000011. These ten bits are appended to the six least significant register bits to produce an address within the range of $00C0 to $00FF. Subroutines and interrupts may use up to 64 (decimal) locations. If 64 locations are exceeded, the stack pointer wraps around and overwrites the previously stored information. A subroutine call occupies two locations on the stack; an interrupt uses five locations.

9.1.5 Condition Code Register (CCR)

The CCR is a 5-bit register in which four bits are used to indicate the results of the instruction just executed, and the fifth bit indicates whether interrupts are masked. These bits can be individually tested by a program, and specific actions can be taken as a result of their state. Each bit is explained in the following paragraphs.

9.1.5.1 Half carry (H)

This bit is set during ADD and ADC operations to indicate that a carry occurred between bits 3 and 4.

9.1.5.2 Interrupt (I)

When this bit is set all maskable interrupts are masked. If an interrupt occurs while this bit is set, the interrupt is latched and remains pending until the interrupt bit is cleared.

9.1.5.3 Negative (N)

When set, this bit indicates that the result of the last arithmetic, logical, or data manipulation was negative.

9.1.5.4 Zero (Z)

When set, this bit indicates that the result of the last arithmetic, logical, or data manipulation was zero.

9.1.5.5 Carry/Borrow (C)

When set, this bit indicates that a carry or borrow out of the arithmetic logical unit (ALU) occurred during the last arithmetic operation. This bit is also affected during bit test and branch instructions and during shifts and rotates.

9.2 INSTRUCTION SET

The MCU has a set of 62 basic instructions. They can be grouped into five different types as follows:

- Register/memory
- Read-modify-write
- Branch
- Bit manipulation
- Control

The following paragraphs briefly explain each type. All the instructions within a given type are presented in individual tables.

This MCU uses all the instructions available in the M146805 CMOS Family plus one more: the unsigned multiply (MUL) instruction. This instruction allows unsigned multiplication of the contents of the accumulator (A) and the index register (X). The high-order product is then stored in the index register, and the low-order product is stored in the accumulator. A detailed definition of the MUL instruction is shown in Table 9-1.

9.2.1 Register/Memory Instructions

Most of these instructions use two operands. The first operand is either the accumulator or the index register. The second operand is obtained from memory using one of the addressing modes. The jump unconditional (JMP) and jump to subroutine (JSR) instructions have no register operand. Refer to Table 9-2 for a complete list of register/memory instructions.

9.2.2 Branch Instructions

These instructions cause the program to branch if a particular condition is met; otherwise, no operation is performed. Branch instructions are two-byte instructions. Refer to Table 9-3.

9.2.3 Bit Manipulation Instructions

The MCU can set or clear any writable bit that resides in the first 256 bytes of the memory space (Page 0). All port data and data direction registers, timer and serial interface registers and, control/status registers and a portion of the on-chip RAM reside in Page 0. An additional feature allows the software to test and branch on the state of any bit within these locations. The bit set, bit clear, bit test, and branch functions are all implemented with single instructions. For the test and branch instructions, the value of the bit tested is also placed in the carry bit of the condition code register. Refer to Table 9-4.

9.2.4 Read/Modify/Write Instructions

These instructions read a memory location or a register, modify or test its contents, and write the modified value back to memory or to the register. The test for negative or zero (TST) instruction is an exception to this sequence of reading, modifying and writing, since it does not modify the value. Refer to Table 9-5 for a complete list of read/modify/write instructions.

9.2.5 Control Instructions

These instructions are register reference instructions and are used to control processor operation during program execution. Refer to Table 9-6 for a complete list of control instructions.

9.2.6 Tables

Tables for all the instruction types listed above follow. In addition there is a complete alphabetical listing of all the instructions (see Table 9-7 and Table 9-8), and an opcode map for the instruction set of the M68HC05 MCU family (see Table 9-9).

Table 9-1 MUL instruction

Operation	**X:A ← X*A**			
Description	Multiplies the eight bits in the index register by the eight bits in the accumulator and places the 16-bit result in the concatenated accumulator and index register.			
Condition Codes	H : Cleared I : Not affected N : Not affected Z : Not affected C : Cleared			
Source	MUL			
Form	Addressing Mode Inherent	Cycles 11	Bytes 1	Opcode $42

Table 9-2 Register/memory instructions

		Addressing Modes																	
		Immediate			**Direct**			**Extended**			**Indexed (No Offset)**			**Indexed (8-bit Offset)**			**Indexed (16-bit Offset)**		
Function	**Mnemonic**	**Opcode**	**# Bytes**	**# Cycles**	**Opcode**	**# Bytes**	**# Cycles**	**Opcode**	**# Bytes**	**# Cycles**	**Opcode**	**# Bytes**	**# Cycles**	**Opcode**	**# Bytes**	**# Cycles**	**Opcode**	**# Bytes**	**# Cycles**
Load A from Memory	LDA	A6	2	2	B6	2	3	C6	3	4	F6	1	3	E6	2	4	D6	3	5
Load X from Memory	LDX	AE	2	2	BE	2	3	CE	3	4	FE	1	3	EE	2	4	DE	3	5
Store A in Memory	STA				B7	2	4	C7	3	5	F7	1	4	E7	2	5	D7	3	6
Store X in Memory	STX				BF	2	4	CF	3	5	FF	1	4	EF	2	5	DF	3	6
Add Memory to A	ADD	AB	2	2	BB	2	3	CB	3	4	FB	1	3	EB	2	4	DB	3	5
Add Memory and Carry to A	ADC	A9	2	2	B9	2	3	C9	3	4	F9	1	3	E9	2	4	D9	3	5
Subtract Memory	SUB	A0	2	2	B0	2	3	C0	3	4	F0	1	3	E0	2	4	D0	3	5
Subtract Memory from A with Borrow	SBC	A2	2	2	B2	2	3	C2	3	4	F2	1	3	E2	2	4	D2	3	5
AND Memory with A	AND	A4	2	2	B4	2	3	C4	3	4	F4	1	3	E4	2	4	D4	3	5
OR Memory with A	ORA	AA	2	2	BA	2	3	CA	3	4	FA	1	3	EA	2	4	DA	3	5
Exclusive OR Memory with A	EOR	A8	2	2	B8	2	3	C8	3	4	F8	1	3	E8	2	4	D8	3	5
Arithmetic Compare A with Memory	CMP	A1	2	2	B1	2	3	C1	3	4	F1	1	3	E1	2	4	D1	3	5
Arithmetic Compare X with Memory	CPX	A3	2	2	B3	2	3	C3	3	4	F3	1	3	E3	2	4	D3	3	5
Bit Test Memory with A (Logical Compare)	BIT	A5	2	2	B5	2	3	C5	3	4	F5	1	3	E5	2	4	D5	3	5
Jump Unconditional	JMP				BC	2	2	CC	3	3	FC	1	2	EC	2	3	DC	3	4
Jump to Subroutine	JSR				BD	2	5	CD	3	6	FD	1	5	ED	2	6	DD	3	7

Table 9-3 Branch instructions

		Relative Addressing Mode		
Function	**Mnemonic**	**Opcode**	**# Bytes**	**# Cycles**
Branch Always	BRA	20	2	3
Branch Never	BRN	21	2	3
Branch if Higher	BHI	22	2	3
Branch if Lower or Same	BLS	23	2	3
Branch if Carry Clear	BCC	24	2	3
(Branch if Higher or Same)	(BHS)	24	2	3
Branch if Carry Set	BCS	25	2	3
(Branch if Lower)	(BLO)	25	2	3
Branch if Not Equal	BNE	26	2	3
Branch if Equal	BEQ	27	2	3
Branch if Half Carry Clear	BHCC	28	2	3
Branch if Half Carry Set	BHCS	29	2	3
Branch if Plus	BPL	2A	2	3
Branch if Minus	BMI	2B	2	3
Branch if Interrupt Mask Bit is Clear	BMC	2C	2	3
Branch if Interrupt Mask Bit is Set	BMS	2D	2	3
Branch if Interrupt Line is Low	BIL	2E	2	3
Branch if Interrupt Line is High	BIH	2F	2	3
Branch to Subroutine	BSR	AD	2	6

Table 9-4 Bit manipulation instructions

		Addressing Modes					
		Bit Set/Clear			Bit Test and Branch		
Function	**Mnemonic**	**Opcode**	**# Bytes**	**# Cycles**	**Opcode**	**# Bytes**	**# Cycles**
Branch if Bit n is Set	BRSET n (n=0–7)				2•n	3	5
Branch if Bit n is Clear	BRCLR n (n=0–7)				01+2•n	3	5
Set Bit n	BSET n (n=0–7)	10+2•n	2	5			
Clear Bit n	BCLR n (n=0–7)	11+2•n	2	5			

Table 9-5 Read/modify/write instructions

		Addressing Modes														
		Inherent (A)			Inherent (X)			Direct			Indexed (No Offset)			Indexed (8-bit Offset)		
Function	Mnemonic	Opcode	# Bytes	# Cycles	Opcode	# Bytes	# Cycles	Opcode	# Bytes	# Cycles	Opcode	# Bytes	# Cycles	Opcode	# Bytes	# Cycles
Increment	INC	4C	1	3	5C	1	3	3C	2	5	7C	1	5	6C	2	6
Decrement	DEC	4A	1	3	5A	1	3	3A	2	5	7A	1	5	6A	2	6
Clear	CLR	4F	1	3	5F	1	3	3F	2	5	7F	1	5	6F	2	6
Complement	COM	43	1	3	53	1	3	33	2	5	73	1	5	63	2	6
Negate (Two's Complement)	NEG	40	1	3	50	1	3	30	2	5	70	1	5	60	2	6
Rotate Left Through Carry	ROL	49	1	3	59	1	3	39	2	5	79	1	5	69	2	6
Rotate Right Through Carry	ROR	46	1	3	56	1	3	36	2	5	76	1	5	66	2	6
Logical Shift Left	LSL	48	1	3	58	1	3	38	2	5	78	1	5	68	2	6
Logical Shift Right	LSR	44	1	3	54	1	3	34	2	5	74	1	5	64	2	6
Arithmetic Shift Right	ASR	47	1	3	57	1	3	37	2	5	77	1	5	67	2	6
Test for Negative or Zero	TST	4D	1	3	5D	1	3	3D	2	4	7D	1	4	6D	2	5
Multiply	MUL	42	1	11												

Table 9-6 Control instructions

		Inherent Addressing Mode		
Function	Mnemonic	Opcode	# Bytes	# Cycles
Transfer A to X	TAX	97	1	2
Transfer X to A	TXA	9F	1	2
Set Carry Bit	SEC	99	1	2
Clear Carry Bit	CLC	98	1	2
Set Interrupt Mask Bit	SEI	9B	1	2
Clear Interrupt Mask Bit	CLI	9A	1	2
Software Interrupt	SWI	83	1	10
Return from Subroutine	RTS	81	1	6
Return from Interrupt	RTI	80	1	9
Reset Stack Pointer	RSP	9C	1	2
No-Operation	NOP	9D	1	2
Stop	STOP	8E	1	2
Wait	WAIT	8F	1	2

Table 9-7 Instruction set (1 of 2)

Mnemonic	Addressing Modes										Condition Codes				
	INH	IMM	DIR	EXT	REL	IX	IX1	IX2	BSC	BTB	H	I	N	Z	C
ADC												•			
ADD												•			
AND											•	•			•
ASL											•	•			
ASR											•	•			
BCC											•	•	•	•	•
BCLR											•	•	•	•	•
BCS											•	•	•	•	•
BEQ											•	•	•	•	•
BHCC											•	•	•	•	•
BHCS											•	•	•	•	•
BHI											•	•	•	•	•
BHS											•	•	•	•	•
BIH											•	•	•	•	•
BIL											•	•	•	•	•
BIT											•	•			•
BLO											•	•	•	•	•
BLS											•	•	•	•	•
BMC											•	•	•	•	•
BMI											•	•	•	•	•
BMS											•	•	•	•	•
BNE											•	•	•	•	•
BPL											•	•	•	•	•
BRA											•	•	•	•	•
BRN											•	•	•	•	•
BRCLR											•	•	•	•	
BRSET											•	•	•	•	
BSET											•	•	•	•	•
BSR											•	•	•	•	•
CLC											•	•	•	•	0
CLI											•	0	•	•	•
CLR											•	•	0	1	•
CMP											•	•			

Address Mode Abbreviations

BSC	Bit Set/Clear	IMM	Immediate
BTB	Bit Test & Branch	IX	Indexed (No Offset)
DIR	Direct	IX1	Indexed, 1 byte Offset
EXT	Extended	IX2	Indexed, 2 byte Offset
INH	Inherent	REL	Relative

Not Implemented

Condition Code Symbols

H	Half Carry (from Bit 3)		Tested and Set if True, Cleared otherwise
I	Interrupt Mask	•	Not Affected
N	Negate (Sign Bit)	?	Load CCR from Stack
Z	Zero	0	Cleared
C	Carry/Borrow	1	Set

Table 9-8 Instruction set (2 of 2)

Mnemonic	Addressing Modes										Condition Codes				
	INH	IMM	DIR	EXT	REL	IX	IX1	IX2	BSC	BTB	H	I	N	Z	C
COM											•	•			1
CPX											•	•			
DEC											•	•			•
EOR											•	•			•
INC											•	•			•
JMP											•	•	•	•	•
JSR											•	•	•	•	•
LDA											•	•			•
LDX											•	•			•
LSL											•	•			
LSR											•	•	0		
MUL											0	•	•	•	0
NEG											•	•			
NOP											•	•	•	•	•
ORA											•	•			•
ROL											•	•			
ROR											•	•			
RSP											•	•	•	•	•
RTI											?	?	?	?	?
RTS											•	•	•	•	•
SBC											•	•			
SEC											•	•	•	•	1
SEI											•	1	•	•	•
STA											•	•			•
STOP											•	0	•	•	•
STX											•	•			•
SUB											•	•			
SWI											•	1	•	•	•
TAX											•	•	•	•	•
TST											•	•			•
TXA											•	•	•	•	•
WAIT											•	0	•	•	•

Address Mode Abbreviations

BSC	Bit Set/Clear	IMM	Immediate
BTB	Bit Test & Branch	IX	Indexed (No Offset)
DIR	Direct	IX1	Indexed, 1 byte Offset
EXT	Extended	IX2	Indexed, 2 byte Offset
INH	Inherent	REL	Relative

Not Implemented

Condition Code Symbols

H	Half Carry (from Bit 3)		Tested and Set if True, Cleared otherwise
I	Interrupt Mask	•	Not Affected
N	Negate (Sign Bit)	?	Load CCR from Stack
Z	Zero	0	Cleared
C	Carry/Borrow	1	Set

Table 9-9 M68HC05 opcode map

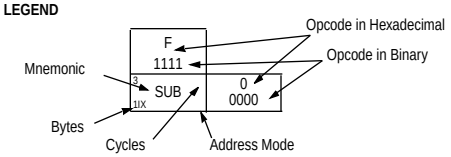
	Bit Manipulation		Branch	Read/Modify/Write					Control		Register/Memory						
	BTB	BSC	REL	DIR	INH	INH	IX1	IX	INH	INH	IMM	DIR	EXT	IX2	IX1	IX	
High / Low	0 0000	1 0001	2 0010	3 0011	4 0100	5 0101	6 0110	7 0111	8 1000	9 1001	A 1010	B 1011	C 1100	D 1101	E 1110	F 1111	High / Low
0 0000	5 BRSET0 3BTB	5 BSET0 2BSC	3 BRA 2REL	5 NEG 2DIR	3 NEGA 1INH	3 NEGX 1INH	6 NEG 2IX1	5 NEG 1IX	9 RTI 1INH		2 SUB 2IMM	3 SUB 2DIR	4 SUB 3EXT	5 SUB 3IX2	4 SUB 2IX1	3 SUB 1IX	0 0000
1 0001	5 BRCLR0 3BTB	5 BCLR0 2BSC	3 BRN 2REL						6 RTS 1INH		2 CMP 2IMM	3 CMP 2DIR	4 CMP 3EXT	5 CMP 3IX2	4 CMP 2IX1	3 CMP 1IX	1 0001
2 0010	5 BRSET1 3BTB	5 BSET1 2BSC	3 BHI 2REL		11 MUL 1INH						2 SBC 2IMM	3 SBC 2DIR	4 SBC 3EXT	5 SBC 3IX2	4 SBC 2IX1	3 SBC 1IX	2 0010
3 0011	5 BRCLR1 3BTB	5 BCLR1 2BSC	3 BLS 2REL	5 COM 2DIR	3 COMA 1INH	3 COMX 1INH	6 COM 2IX1	5 COM 1IX	10 SWI 1INH		2 CPX 2IMM	3 CPX 2DIR	4 CPX 3EXT	5 CPX 3IX2	4 CPX 2IX1	3 CPX 1IX	3 0011
4 0100	5 BRSET2 3BTB	5 BSET2 2BSC	3 BCC 2REL	5 LSR 2DIR	3 LSRA 1INH	3 LSRX 1INH	6 LSR 2IX1	5 LSR 1IX			2 AND 2IMM	3 AND 2DIR	4 AND 3EXT	5 AND 3IX2	4 AND 2IX1	3 AND 1IX	4 0100
5 0101	5 BRCLR2 3BTB	5 BCLR2 2BSC	3 BCS 2REL								2 BIT 2IMM	3 BIT 2DIR	4 BIT 3EXT	5 BIT 3IX2	4 BIT 2IX1	3 BIT 1IX	5 0101
6 0110	5 BRSET3 3BTB	5 BSET3 2BSC	3 BNE 2REL	5 ROR 2DIR	3 RORA 1INH	3 RORX 1INH	6 ROR 2IX1	5 ROR 1IX			2 LDA 2IMM	3 LDA 2DIR	4 LDA 3EXT	5 LDA 3IX2	4 LDA 2IX1	3 LDA 1IX	6 0110
7 0111	5 BRCLR3 3BTB	5 BCLR3 2BSC	3 BEQ 2REL	5 ASR 2DIR	3 ASRA 1INH	3 ASRX 1INH	6 ASR 2IX1	5 ASR 1IX		2 TAX 1INH		4 STA 2DIR	5 STA 3EXT	6 STA 3IX2	5 STA 2IX1	4 STA 1IX	7 0111
8 1000	5 BRSET4 3BTB	5 BSET4 2BSC	3 BHCC 2REL	5 LSL 2DIR	3 LSLA 1INH	3 LSLX 1INH	6 LSL 2IX1	5 LSL 1IX		2 CLC 1INH	2 EOR 2IMM	3 EOR 2DIR	4 EOR 3EXT	5 EOR 3IX2	4 EOR 2IX1	3 EOR 1IX	8 1000
9 1001	5 BRCLR4 3BTB	5 BCLR4 2BSC	3 BHCS 2REL	5 ROL 2DIR	3 ROLA 1INH	3 ROLX 1INH	6 ROL 2IX1	5 ROL 1IX		2 SEC 1INH	2 ADC 2IMM	3 ADC 2DIR	4 ADC 3EXT	5 ADC 3IX2	4 ADC 2IX1	3 ADC 1IX	9 1001
A 1010	5 BRSET5 3BTB	5 BSET5 2BSC	3 BPL 2REL	5 DEC 2DIR	3 DECA 1INH	3 DECX 1INH	6 DEC 2IX1	5 DEC 1IX		2 CLI 1INH	2 ORA 2IMM	3 ORA 2DIR	4 ORA 3EXT	5 ORA 3IX2	4 ORA 2IX1	3 ORA 1IX	A 1010
B 1011	5 BRCLR5 3BTB	5 BCLR5 2BSC	3 BMI 2REL							2 SEI 1INH	2 ADD 2IMM	3 ADD 2DIR	4 ADD 3EXT	5 ADD 3IX2	4 ADD 2IX1	3 ADD 1IX	B 1011
C 1100	5 BRSET6 3BTB	5 BSET6 2BSC	3 BMC 2REL	5 INC 2DIR	3 INCA 1INH	3 INCX 1INH	6 INC 2IX1	5 INC 1IX		2 RSP 1INH		2 JMP 2DIR	3 JMP 3EXT	4 JMP 3IX2	3 JMP 2IX1	2 JMP 1IX	C 1100
D 1101	5 BRCLR6 3BTB	5 BCLR6 2BSC	3 BMS 2REL	4 TST 2DIR	3 TSTA 1INH	3 TSTX 1INH	5 TST 2IX1	4 TST 1IX		2 NOP 1INH	6 BSR 2REL	5 JSR 2DIR	6 JSR 3EXT	7 JSR 3IX2	6 JSR 2IX1	5 JSR 1IX	D 1101
E 1110	5 BRSET7 3BTB	5 BSET7 2BSC	3 BIL 2REL						2 STOP 1INH		2 LDX 2IMM	3 LDX 2DIR	4 LDX 3EXT	5 LDX 3IX2	4 LDX 2IX1	3 LDX 1IX	E 1110
F 1111	5 BRCLR7 3BTB	5 BCLR7 2BSC	3 BIH 2REL	5 CLR 2DIR	3 CLRA 1INH	3 CLRX 1INH	6 CLR 2IX1	5 CLR 1IX	2 WAIT 1INH	2 TXA 1INH		4 STX 2DIR	5 STX 3EXT	6 STX 3IX2	5 STX 2IX1	4 STX 1IX	F 1111

Abbreviations for Address Modes and Registers

BSC	Bit Set/Clear	IX	Indexed (No Offset)
BTB	Bit Test and Branch	IX1	Indexed, 1 byte (8-bit) Offset
DIR	Direct	IX2	Indexed, 2 byte (16-bit) Offset
EXT	Extended	REL	Relative
INH	Inherent	A	Accumulator
IMM	Immediate	X	Index Register

Not Implemented

LEGEND



9.3 ADDRESSING MODES

Ten different addressing modes provide programmers with the flexibility to optimise their code for all situations. The various indexed addressing modes make it possible to locate data tables, code conversion tables, and scaling tables anywhere in the memory space. Short indexed accesses are single byte instructions; the longest instructions (three bytes) permit accessing tables throughout memory. Short absolute (direct) and long absolute (extended) addressing is also included. One or two byte direct addressing instructions access all data bytes in most applications. Extended addressing permits jump instructions to reach all memory locations.

The term ‘effective address’ (EA) is used in describing the various addressing modes. Effective address is defined as the address from which the argument for an instruction is fetched or stored. The ten addressing modes of the processor are described below. Parentheses are used to indicate ‘contents of’ the location or register referred to. For example, (PC) indicates the contents of the location pointed to by the PC (Program Counter). An arrow indicates ‘is replaced by’, and a colon indicates concatenation of two bytes. For additional details and graphical illustrations, refer to the *M6805 HMOS/M146805 CMOS Family Microcomputer/Microprocessor User's Manual* or to the *M68HC05 Applications Guide.*

9.3.1 Inherent

In the inherent addressing mode, all the information necessary to execute the instruction is contained in the opcode. Operations specifying only the index register or accumulator as well as the control instruction with no other arguments are included in this mode. These instructions are one byte long.

9.3.2 Immediate

In the immediate addressing mode, the operand is contained in the byte immediately following the opcode. The immediate addressing mode is used to access constants that do not change during program execution (e.g., a constant used to initialise a loop counter).

$$EA = PC+1;\ PC \leftarrow PC+2$$

9.3.3 Direct

In the direct addressing mode, the effective address of the argument is contained in a single byte following the opcode byte. Direct addressing allows the user to directly address the lowest 256 bytes in memory with a single two-byte instruction.

$$EA = (PC+1);\ PC \leftarrow PC+2$$
$$\text{Address Bus High} \leftarrow 0;\ \text{Address Bus Low} \leftarrow (PC+1)$$

9.3.4 Extended

In the extended addressing mode, the effective address of the argument is contained in the two bytes following the opcode byte. Instructions with extended addressing mode are capable of referencing arguments anywhere in memory with a single three-byte instruction. When using the Motorola assembler, the user need not specify whether an instruction uses direct or extended addressing. The assembler automatically selects the short form of the instruction.

EA = (PC+1):(PC+2); PC ← PC+3
Address Bus High ← (PC+1); Address Bus Low ← (PC+2)

9.3.5 Indexed, No Offset

In the indexed, no offset addressing mode, the effective address of the argument is contained in the 8-bit index register. This addressing mode can access the first 256 memory locations. These instructions are only one byte long. This mode is often used to move a pointer through a table or to hold the address of a frequently referenced RAM or I/O location.

EA = X; PC ← PC+1
Address Bus High ← 0; Address Bus Low ← X

9.3.6 Indexed, 8-bit Offset

In the indexed, 8-bit offset addressing mode, the effective address is the sum of the contents of the unsigned 8-bit index register and the unsigned byte following the opcode. Therefore the operand can be located anywhere within the lowest 511 memory locations. This addressing mode is useful for selecting the mth element in an n element table.

EA = X+(PC+1); PC ← PC+2
Address Bus High ← K; Address Bus Low ← X+(PC+1)
where K = the carry from the addition of X and (PC+1)

9.3.7 Indexed, 16-bit Offset

In the indexed, 16-bit offset addressing mode, the effective address is the sum of the contents of the unsigned 8-bit index register and the two unsigned bytes following the opcode. This address mode can be used in a manner similar to indexed, 8-bit offset except that this three-byte instruction allows tables to be anywhere in memory. As with direct and extended addressing, the Motorola assembler determines the shortest form of indexed addressing.

EA = X+[(PC+1):(PC+2)]; PC ← PC+3
Address Bus High ← (PC+1)+K; Address Bus Low ← X+(PC+2)
where K = the carry from the addition of X and (PC+2)

9.3.8 Relative

The relative addressing mode is only used in branch instructions. In relative addressing, the contents of the 8-bit signed byte (the offset) following the opcode is added to the PC if, and only if, the branch conditions are true. Otherwise, control proceeds to the next instruction. The span of relative addressing is from –126 to +129 from the opcode address. The programmer need not calculate the offset when using the Motorola assembler, since it calculates the proper offset and checks to see that it is within the span of the branch.

EA = PC+2+(PC+1); PC ← EA if branch taken;
otherwise EA = PC ← PC+2

9.3.9 Bit Set/Clear

In the bit set/clear addressing mode, the bit to be set or cleared is part of the opcode. The byte following the opcode specifies the address of the byte in which the specified bit is to be set or cleared. Any read/write bit in the first 256 locations of memory, including I/O, can be selectively set or cleared with a single two-byte instruction.

EA = (PC+1); PC ← PC+2
Address Bus High ← 0; Address Bus Low ← (PC+1)

9.3.10 Bit Test and Branch

The bit test and branch addressing mode is a combination of direct addressing and relative addressing. The bit to be tested and its condition (set or clear) is included in the opcode. The address of the byte to be tested is in the single byte immediately following the opcode byte (EA1). The signed relative 8-bit offset in the third byte (EA2) is added to the PC if the specified bit is set or cleared in the specified memory location. This single three-byte instruction allows the program to branch based on the condition of any readable bit in the first 256 locations of memory. The span of branch is from –125 to +130 from the opcode address. The state of the tested bit is also transferred to the carry bit of the condition code register.

EA1 = (PC+1); PC ← PC+2
Address Bus High ← 0; Address Bus Low ← (PC+1)
EA2 = PC+3+(PC+2); PC ← EA2 if branch taken;
otherwise PC ← PC+3

THIS PAGE INTENTIONALLY LEFT BLANK

SECTION 10
ELECTRICAL SPECIFICATIONS

This section contains the electrical specifications and associated timing information for the MC68HC05L4.

10.1 MAXIMUM RATINGS

Table 10-1 Maximum ratings

Rating	Symbol	Value	Unit
Supply Voltage	V_{DD}	– 0.3 to +7.0	V
Input Voltage:	V_{in}	V_{SS} – 0.3 to V_{DD} + 0.3	V
Input Voltage: *$\overline{IRQ}$/VPP – MC68HC705L4* *Bootloader mode*	V_{in}	V_{SS} – 0.3 to $2V_{DD}$ + 0.3	V
Operating Temperature Range	T_A	T_L to T_H 0 to +70	C
Storage Temperature Range	T_{stg}	– 65 to +150	C
Current Drain per pin (note 2) – excluding VDD and VSS	I_D	25	mA

Note 1) All voltages are with respect to V_{SS}.

Note 2) Maximum current drain per pin is for one pin at a time, limited by an external resistor.

Note 3) This device contains circuitry designed to protect against damage due to high electrostatic voltages or electric fields. However, it is recommended that normal precautions be taken to avoid the application of any voltages higher than those given in the Maximum Ratings table to this high impedance circuit. For maximum reliability all unused inputs should be tied to either V_{SS} or V_{DD}.

10.2 THERMAL CHARACTERISTICS AND POWER CONSIDERATIONS

Table 10-2 Package thermal characteristics

Characteristics	Symbol	Value	Unit
Thermal Resistance			
– Plastic 64 pin Shrink DIL package	θ_{JA}	40	C/W
– Plastic 64 pin QFP package	θ_{JA}	50	C/W
– Plastic 68 pin PLCC package	θ_{JA}	50	C/W
– Ceramic 64 pin Shrink DIL package	θ_{JA}	40	C/W
– Ceramic 68 pin Cerquad package	θ_{JA}	45	C/W

The average chip junction temperature, T_J, in degrees Celcius can be obtained from the following equation:

$$T_J = T_A + (P_D \bullet \theta_{JA}) \quad [1]$$

where:

T_A = Ambient Temperature (C)

θ_{JA} = Package Thermal Resistance, Junction-to-ambient (C/W)

$P_D = P_{INT} + P_{I/O}$ (W)

P_{INT} = Internal Chip Power = $I_{DD} \bullet V_{DD}$ (W)

$P_{I/O}$ = Power Dissipation on Input and Output pins (User determined)

An approximate relationship between P_D and T_J (if $P_{I/O}$ is neglected) is:

$$P_D = \frac{K}{T_J + 273} \quad [2]$$

Solving equations [1] and [2] for K gives:

$$K = P_D \bullet (T_A + 273) + \theta_{JA} \bullet P_D^2 \quad [3]$$

where K is a constant for a particular part. K can be determined by measuring P_D (at equilibrium) for a known T_A. Using this value of K, the values of P_D and T_J can be obtained for any value of T_A by solving the above equations. The package thermal characteristics are shown in Table 10-2.

Pins	R1	R2	C
PA0–7, PB0–7, PC0–7	3.26k	2.38k	50pF

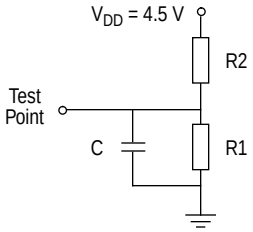


Figure 10-1 Equivalent test load

10.3 DC ELECTRICAL CHARACTERISTICS FOR 5V OPERATION

Table 10-3 DC electrical characteristics

(V_{DD} = 5.0 Vdc ± 10%, V_{SS} = 0 Vdc, $T_A = T_L$ to T_H)

Characteristic	Symbol	Min	Typ	Max	Unit
Output Voltage					
I_{LOAD} = – 10 µA	V_{OH}	V_{DD} – 0.1	—	—	V
I_{LOAD} = +10 µA	V_{OL}	—	—	0.1	V
Output High Voltage (I_{LOAD} = – 0.8 mA) PA0–7, PB0–7, PC0–7	V_{OH}	V_{DD} – 0.8	—	—	V
Output Low Voltage (I_{LOAD} = +1.6 mA) PA0–7, PB0–7, PC0–7	V_{OL}	—	—	0.4	V
Input High Voltage PA0–7, PB0–7, PC0–7, PD0–7, OSC1 $\overline{IRQ}$, $\overline{RESET}$	V_{IH}	0.7V_{DD}	—	V_{DD}	V
Input Low Voltage PA0–7, PB0–7, PC0–7, PD0–7, OSC1 $\overline{IRQ}$, $\overline{RESET}$	V_{IL}	V_{SS}	—	0.2V_{DD}	V
Supply Current (see Notes)	I_{DD}				
RUN		—	TBD	TBD	mA
WAIT		—	TBD	TBD	mA
STOP		—	TBD	TBD	µA
High-Z Leakage Current	I_{IL}				
PA0–7, PB0–7, PD0–7		—	—	1	µA
PC0–7		—	—	20	µA
Input Current	I_{IN}				
$\overline{RESET}$		—	—	20	µA
$\overline{IRQ}$, OSC1, TCAPA, TCAPB		—	—	1	µA
Port C pull-up resistors PC0–7	—	300k	—	1.5M	
Capacitance					
Ports (as input or output)	C_{OUT}	—	—	12	pF
$\overline{RESET}$, $\overline{IRQ}$, TCAPA	C_{IN}	—	—	8	pF
LCD Voltage Input	V_{LCD}	V_{SS}	—	V_{DD}	V
MC68HC705L4 EPROM:					
Programming Voltage	V_{PP}	*15*	*16*	*17*	*V*
Programming Current	I_{PP}	—	*TBD*	*TBD*	*mA*
Programming Time	t_{PROG}	—	*TBD*	*TBD*	*ms*

Note 1) All I_{DD} measurements taken with suitable decoupling capacitors across the power supply to suppress the transient switching currents inherent in CMOS designs (see Section 2.2.1).

Note 2) Typical values are at mid point of voltage range and at 25 C only

Note 3) RUN and WAIT I_{DD}: measured using an external square-wave clock source (f_{OSC} = 4.2MHz); all inputs 0.2 V from rail; no DC loads; maximum load on outputs 50pF (20pF on OSC2).

Note 4) STOP and WAIT I_{DD}: all ports configured as inputs; V_{IL} = 0.2 V and $V_{IH} = V_{DD} – 0.2$ V.

Note 5) STOP I_{DD}: measured with OSC1 = V_{DD}.

Note 6) WAIT I_{DD} is affected linearly by the OSC2 capacitance.

10.4 AC ELECTRICAL CHARACTERISTICS FOR 5V OPERATION

Table 10-4 AC electrical characteristics

(V_{DD} = 5.0 Vdc 10%, V_{SS} = 0 Vdc, $T_A = T_L$ to T_H)

Characteristic	Symbol	Min	Max	Unit
Frequency of Operation Oscillator frequency	 f_{OSC}	 0	 4.2	 MHz
Internal Operating Frequency Crystal External clock	 f_{OP} f_{OP}	 dc dc	 2.1 2.1	 MHz MHz
Processor cycle time	t_{CYC}	480	—	ns
Oscillator Clock pulse width	t_{OH}, t_{OL}	90	—	ns
Interrupt Pulse Width Low (Edge-Triggered)	t_{ILIH}	125	—	ns
Interrupt pulse period	t_{ILIL}	†	—	t_{cyc}
STOP Recovery start-up time(Crystal Oscillator)	t_{ILCH}	—	100	ms
$\overline{\text{RESET}}$ pulse width	t_{RL}	1.5	—	t_{CYC}
Power-on Reset delay	t_{PORL}	4064	4064	t_{CYC}
Crystal Oscillator start-up time	t_{OXOV}	—	100	ms

† The minimum period T_{ILIL} should not be less than the number of cycle times it takes to execute the interrupt service routine plus 21 t_{cyc}.

10.5 DC ELECTRICAL CHARACTERISTICS FOR 3.3V OPERATION

Table 10-5 DC electrical characteristics

(V_{DD} = 3.3 Vdc 10%, V_{SS} = 0 Vdc, $T_A = T_L$ to T_H)

Characteristic	Symbol	Min	Typ	Max	Unit
Output Voltage I_{LOAD} = – 10 µA I_{LOAD} = +10 µA	 V_{OH} V_{OL}	 V_{DD} – 0.1 —	 — —	 — 0.1	 V V
Output High Voltage (I_{LOAD} = – 0.8 mA) PA0–7, PB0–7, PC0–7	V_{OH}	V_{DD} – 0.3	—	—	V
Output Low Voltage (I_{LOAD} = +1.6 mA) PA0–7, PB0–7, PC0–7	V_{OL}	—	—	0.3	V
Input High Voltage PA0–7, PB0–7, PC0–7, PD0–7, OSC1 $\overline{IRQ}$, $\overline{RESET}$	V_{IH}	0.7V_{DD}	—	V_{DD}	V
Input Low Voltage PA0–7, PB0–7, PC0–7, PD0–7, OSC1 $\overline{IRQ}$, $\overline{RESET}$	V_{IL}	V_{SS}	—	0.2V_{DD}	V
Supply Current (see Notes) RUN WAIT STOP	I_{DD}	 — — —	 TBD TBD TBD	 TBD TBD TBD	 mA mA µA
High-Z Leakage Current PA0–7, PB0–7, PD0–7 PC0–7	I_{IL}	 — —	 — —	 1 20	 µA µA
Input Current $\overline{RESET}$ $\overline{IRQ}$, OSC1, TCAPA, TCAPB	I_{IN}	 — —	 — —	 20 1	 µA µA
Port C pull-up resistors PC0–7	—	300k	—	1.5M	
Capacitance Ports (as input or output) $\overline{RESET}$, $\overline{IRQ}$, TCAPA	 C_{OUT} C_{IN}	 — —	 — —	 12 8	 pF pF
LCD Voltage Input	V_{LCD}	V_{SS}	—	V_{DD}	V

Note 1) All I_{DD} measurements taken with suitable decoupling capacitors across the power supply to suppress the transient switching currents inherent in CMOS designs (see Section 2.2.1).

Note 2) Typical values are at mid point of voltage range and at 25 C only

Note 3) RUN and WAIT I_{DD}: measured using an external square-wave clock source (f_{OSC} = 2.0 MHz); all inputs 0.2 V from rail; no DC loads; maximum load on outputs 50pF (20pF on OSC2).

Note 4) STOP and WAIT I_{DD}: all ports configured as inputs; V_{IL} = 0.2 V and $V_{IH} = V_{DD}$ – 0.2 V.

Note 5) STOP I_{DD}: measured with OSC1 = V_{DD}.

Note 6) WAIT I_{DD} is affected linearly by the OSC2 capacitance.

10.6 AC ELECTRICAL CHARACTERISTICS FOR 3.3V OPERATION

Table 10-6 AC electrical characteristics

(V_{DD} = 3.3 Vdc 10%, V_{SS} = 0 Vdc, $T_A = T_L$ to T_H)

Characteristic	Symbol	Min	Max	Unit
Frequency of Operation Oscillator frequency	f_{OSC}	0	2.0	MHz
Internal Operating Frequency Crystal External clock	 f_{OP} f_{OP}	 dc dc	 1.0 1.0	 MHz MHz
Processor cycle time	t_{CYC}	1000	—	ns
Oscillator Clock pulse width	t_{OH}, t_{OL}	200	—	ns
Interrupt Pulse Width Low (Edge-Triggered)	t_{ILIH}	250	—	ns
Interrupt pulse period	t_{ILIL}	†	—	t_{cyc}
STOP Recovery start-up time(Crystal Oscillator)	t_{ILCH}	—	100	ms
$\overline{\text{RESET}}$ pulse width	t_{RL}	1.5	—	t_{CYC}
Power-on Reset delay	t_{PORL}	4064	4064	t_{CYC}
Crystal Oscillator start-up time	t_{OXOV}	—	100	ms

† The minimum period T_{ILIL} should not be less than the number of cycle times it takes to execute the interrupt service routine plus 21 t_{cyc}.

THIS PAGE INTENTIONALLY LEFT BLANK

SECTION 11
MECHANICAL DATA

11.1 MC68HC05L4 PIN CONFIGURATIONS

11.1.1 68-pin Plastic/Ceramic Leaded Chip Carrier (PLCC or CERQUAD)

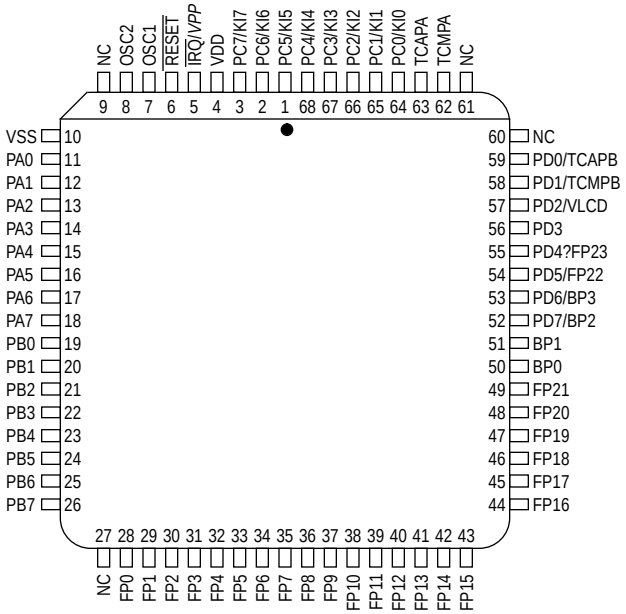


Figure 11-1 68-pin PLCC or Cerquad pin-out

11.1.2 64-pin Plastic Quad Flat Pack (PQFP)

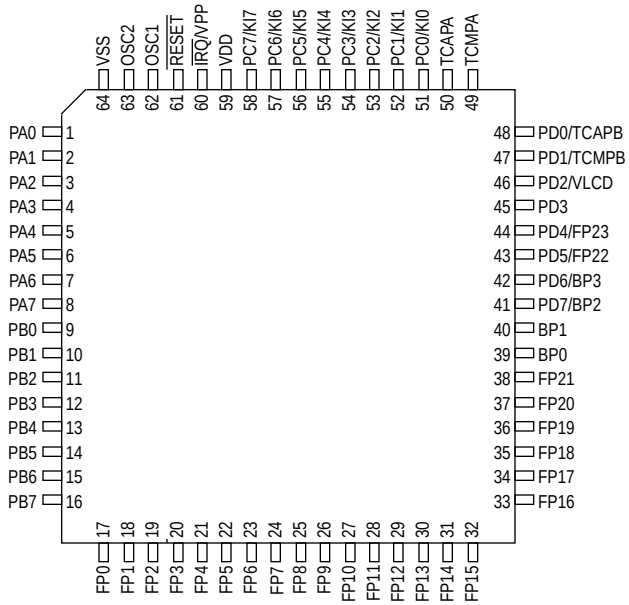


Figure 11-2 64-pin PQFP pin-out

11.1.3 64-pin Shrink Dual-in-Line Plastic/Ceramic (PSDIP or CSDIP)

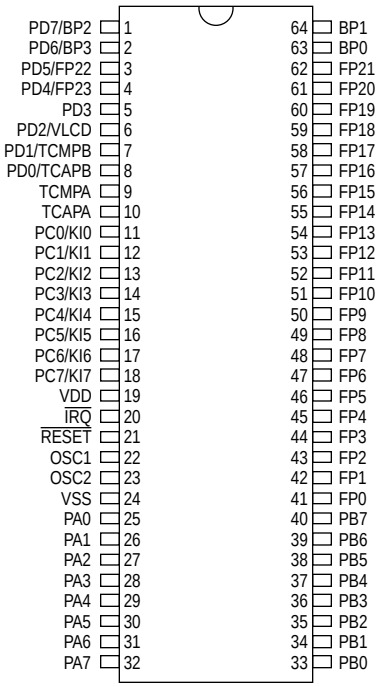
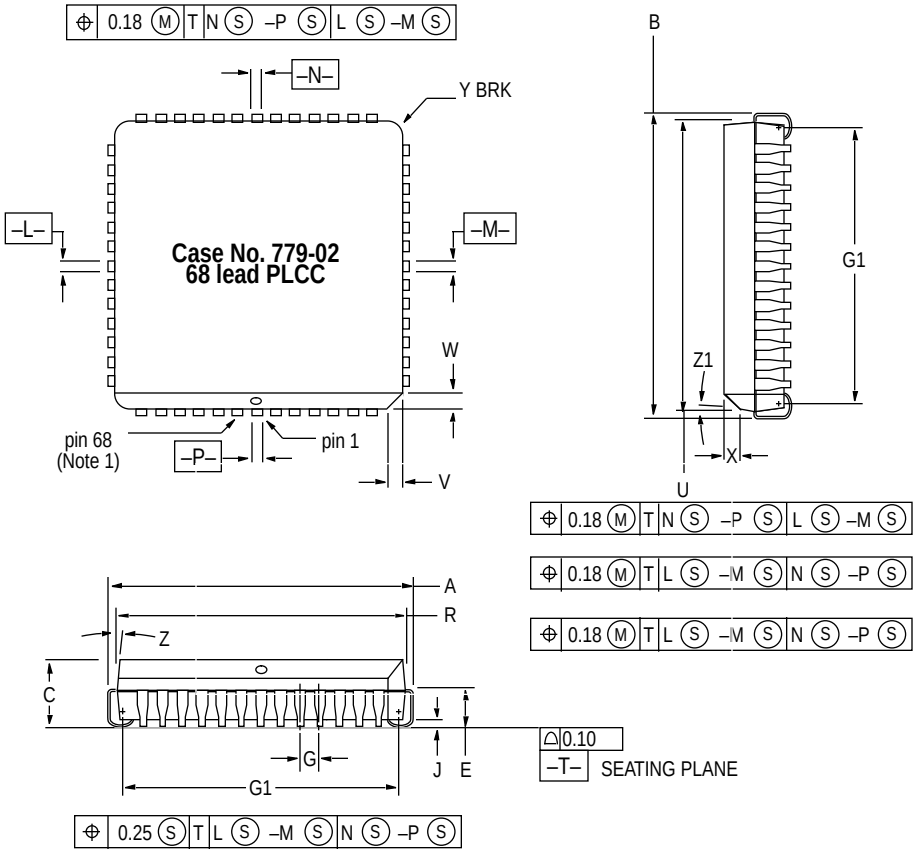


Figure 11-3 64-pin PSDIP or CSDIP pin-out

11.2 MC68HC05L4 MECHANICAL DIMENSIONS

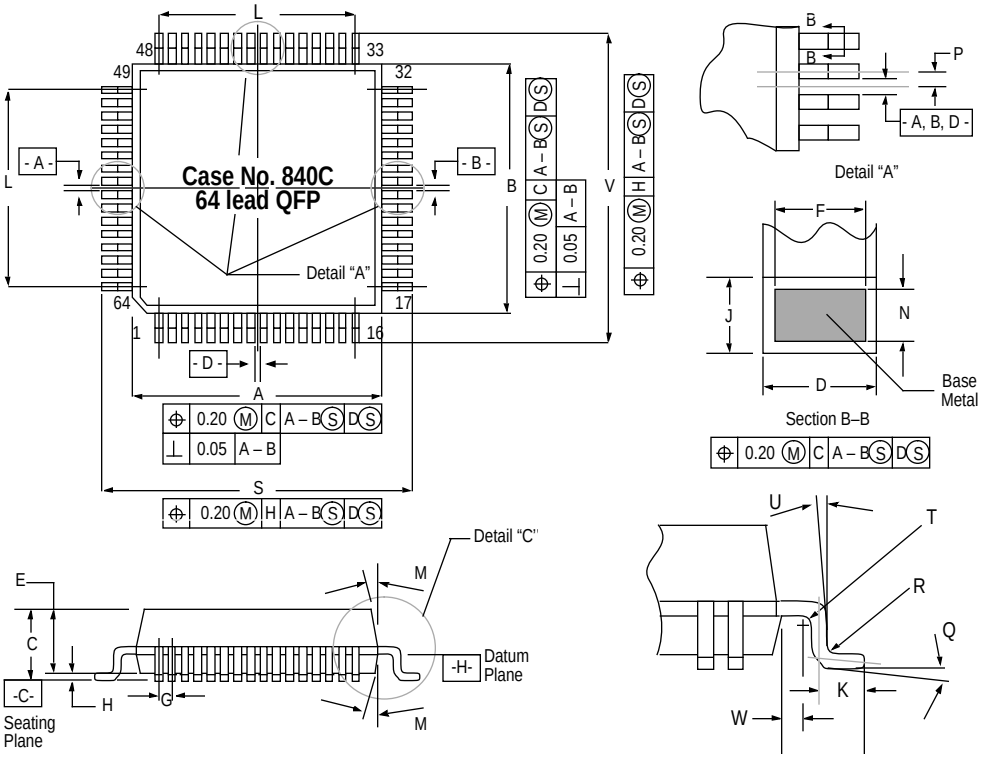
11.2.1 68-pin Plastic Leaded Chip Carrier (PLCC)



Dim.	Min.	Max.	Notes	Dim.	Min.	Max.
A	25.02	25.27		U	24.13	24.28
B	25.02	25.27	1. **Due to space limitations, this case shall be represented by a general case outline, rather than one showing all the leads.**	V	1.07	1.21
C	4.20	4.57		W	1.07	1.21
E	2.29	2.79	2. Datums –L–, –M–, –N– and –P– are determined where top of lead shoulder exits plastic body at mould parting line.	X	1.07	1.42
F	0.33	0.48	3. Dimension G1, true position to be measured at datum –T– (seating plane).	Y	—	0.50
G	1.27 BSC		4. Dimensions R and U do not include mould protrusion. Allowable mould protrusion is 0.25mm per side.	Z	2	10
H	0.66	0.81	5. Dimensions and tolerancing per ANSI Y 14.5M, 1982.	G1	23.12	23.62
J	0.51	—	6. All dimensions in mm.	K1	1.02	—
K	0.64	—		Z1	2	10
R	24.13	24.28				

Figure 11-4 Mechanical dimensions for 68-pin PLCC

11.2.2 64-pin Plastic Quad Flat Pack (PQFP)



Dim.	Min.	Max.
A	13.90	14.10
B	13.90	14.10
C	2.067	2.457
D	0.30	0.45
E	2.00	2.40
F	0.30	—
G	0.80 BSC	
H	0.067	0.250
J	0.130	0.230
K	0.50	0.66
L	12.00 REF	

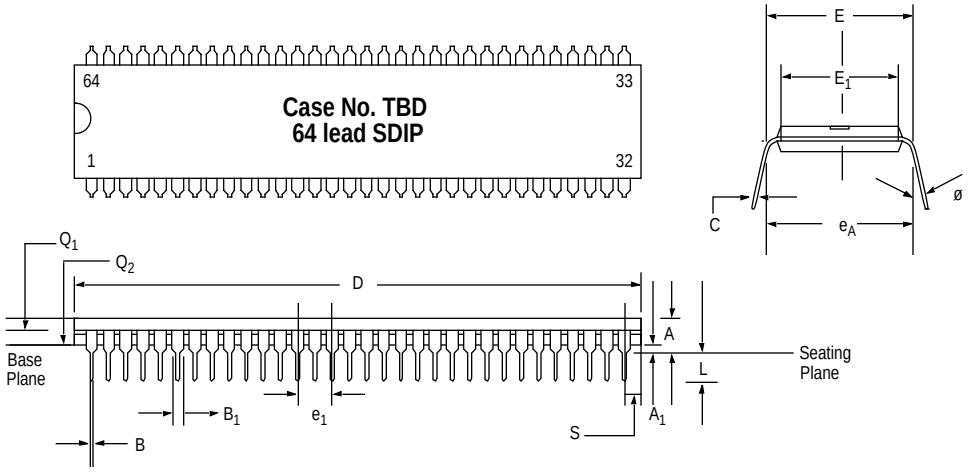
Notes

1. Datum Plane –H– is located at bottom of lead and is coincident with the lead where the lead exits the plastic body at the bottom of the parting line.
2. Datums A–B and –D to be determined at Datum Plane –H–.
3. Dimensions S and V to be determined at seating plane –C–.
4. Dimensions A and B do not include mould protrusion. Allowable mould protrusion is 0.25mm per side. Dimensions A and B do include mould mismatch and are determined at Datum Plane –H–.
5. Dimension D does not include dambar protrusion. Allowable dambar protrusion shall be 0.08 total in excess of the D dimension at maximum material condition. Dambar cannot be located on the lower radius or the foot.
6. Dimensions and tolerancing per ANSI Y 14.5M, 1982.
7. All dimensions in mm.

Dim.	Min.	Max.
M	5	10
N	0.130	0.170
P	0.40 BSC	
Q	2	8
R	0.13	0.30
S	16.20	16.60
T	0.20 REF	
U	9	15
V	16.20	16.60
W	0.042 NOM	
X	1.10	1.30

Figure 11-5 Mechanical dimensions for 64-pin QFP

11.2.3 64-pin Plastic Shrink Dual-in-Line (PSDIP)



Dim.	Min.	Max.
A	–	5.08
A_1	0.51	–
B	0.38	0.56
B_1	0.76	1.27
C	0.20	0.30
D	57.40	57.91
E	19.05	19.69
E_1	16.76	17.27

Notes

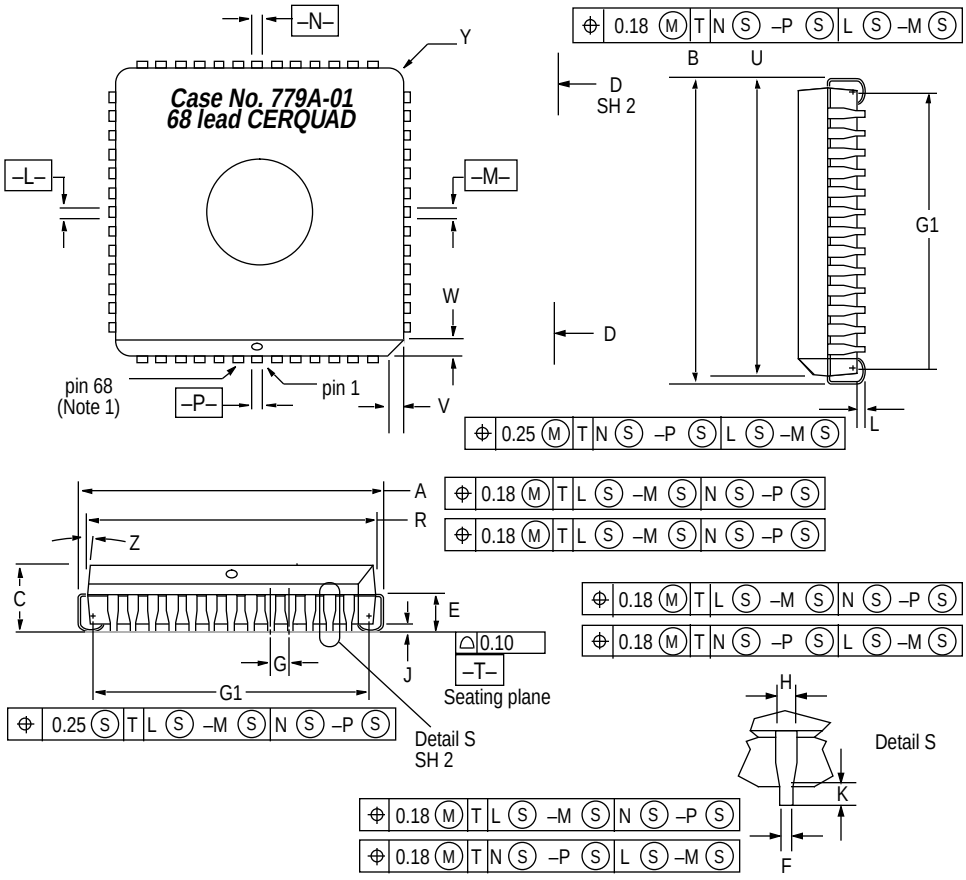
1. Dimensions and tolerancing per ANSI Y 14.5 1982.
2. Controlling dimension is mm.
3. ø applies to spread leads prior to installation.
4. Dimensions D and E_1 are to be measured at maximum material condition but do not include mould flash. Allowable mould flash is 0.254 mm.
5. Dimensions A, A1 and L are measured with the package seated in Jedec seating plane gauge GS-3.
6. This package conforms to EIAJ reference SC-553-64B.

Dim.	Min.	Max.
e_1	1.78 BSC	
e_A	19.05 BSC	
L	3.05	3.43
ø	0	15
Q_1	1.65	1.91
Q_2	3.81	
S	1.19	1.45

Figure 11-6 Mechanical dimensions for 64-pin PSDIP

11.3 MC68HC705L4 MECHANICAL DIMENSIONS

11.3.1 68-pin windowed ceramic leaded chip carrier (Cerquad)



Dim.	Min.	Max.
A	25.02	25.27
B	25.02	25.27
C	3.94	5.08
E	2.29	3.05
F	0.43	0.48
G	1.27 BSC	
H	0.66	0.81
J	0.51	—

Notes

1. **Due to space limitations, this case shall be represented by a general case outline, rather than one showing all the leads.**
2. Datums –L–, –M–, –N– and –P– are determined where top of lead shoulder exits plastic body at mould parting line.
3. Dimension G1, true position to be measured at datum –T– (seating plane).
4. Dimensions R and U do not include mould protrusion. Allowable mould protrusion is 0.25 mm per side.
5. Dimensions and tolerancing per ANSI Y 14.5M, 1982.
6. All dimensions in mm.

Dim.	Min.	Max.
K	1.27 BSC	
L	0.08	—
R	23.62	24.33
U	23.62	24.33
V	0.91	1.12
W	0.91	1.12
G1	22.61	23.62

Figure 11-7 Mechanical dimensions for 68-pin Cerquad

THIS PAGE INTENTIONALLY LEFT BLANK

SECTION 12
ORDERING INFORMATION

This section describes the information needed to order the MC68HC05L4 or *MC68HC705L4*.

To initiate a ROM pattern for the MCU, it is necessary to contact your local field service office, local sales person or Motorola representative. Please note that you will need to supply details such as: mask option selections; temperature range; oscillator frequency; package type; electrical test requirements; and device marking details so that an order can be processed, and a customer specific part number allocated. Refer to Table 12-1 for appropriate part numbers.

Table 12-1 MC order numbers

Device Title	Package Type	Temperature	Part Number
MC68HC05L4	68-pin PLCC	0 to +70 C	MC68HC05L4FN
	64-pin QFP	0 to +70 C	MC68HC05L4FU
	64-pin SDIP	0 to +70 C	MC68HC05L4B
MC68HC705L4	*68-pin windowed Cerquad*	*0 to +70 C*	*MC68HC705L4FS*
	64-pin windowed ceramic SDIP	*0 to +70 C*	*MC68HC705L4K*
	68-pin PLCC (OTP)	*0 to +70 C*	*MC68HC705L4FN*
	64-pin QFP (OTP)	*0 to +70 C*	*MC68HC705L4FU*
	64-pin SDIP (OTP)	*0 to +70 C*	*MC68HC705L4B*

12.1 EPROMS

A 16 kbyte EPROM programmed with the customer's software (positive logic for address and data) should be submitted for pattern generation. All unused bytes should be programmed to $00.

The EPROM should be clearly labelled, placed in a conductive IC carrier and securely packed.

12.2 VERIFICATION MEDIA

All original pattern media (EPROMs) are filed for contractual purposes and are not returned. A computer listing of the ROM code will be generated and returned with a listing verification form.

The listing should be thoroughly checked and the verification form completed, signed and returned to Motorola. The signed verification form constitutes the contractual agreement for creation of the custom mask. If desired, Motorola will program blank EPROMs (supplied by the customer) from the data file used to create the custom mask, to aid in the verification process.

12.3 ROM VERIFICATION UNITS (RVU)

Ten MCUs containing the customer's ROM pattern will be provided for program verification. These units will have been made using the custom mask but are for ROM verification only. For expediency, they are usually unmarked and are tested only at room temperature (25 C) and at 5 Volts. These RVUs are included in the mask charge and are not production parts. They are neither backed nor guaranteed by Motorola Quality Assurance.

– *Cut along this line to remove* –

CUSTOMER FEEDBACK QUESTIONNAIRE (MC68HC05L4)

Motorola wishes to continue to improve the quality of its documentation. We would welcome your feedback on the publication you have just received. Having used the document, please complete this card (or a photocopy of it, if you prefer).

1. How would you rate the quality of the document? Check one box in each category.

	Excellent			Poor		Excellent			Poor
Organisation	☐	☐	☐	☐	Tables	☐	☐	☐	☐
Readability	☐	☐	☐	☐	Table of contents	☐	☐	☐	☐
Understandability	☐	☐	☐	☐	Index	☐	☐	☐	☐
Accuracy	☐	☐	☐	☐	Page size/binding	☐	☐	☐	☐
Illustrations	☐	☐	☐	☐	Overall impression	☐	☐	☐	☐

Comments: _____

2. What is your intended use for this document? If more than one option applies, please rank them (1, 2, 3).

Selection of device for new application ☐
System design ☐
Training purposes ☐
Other ☐ Please specify: _____

3. How well does this manual enable you to perform the task(s) outlined in question 2?

Completely ☐ ☐ ☐ ☐ Not at all
Comments: _____

4. How easy is it to find the information you are looking for?

Easy ☐ ☐ ☐ ☐ Difficult
Comments: _____

5. Is the level of technical detail in the following sections sufficient to allow you to understand how the device functions?

		Too little detail				Too much detail
SECTION 1	INTRODUCTION	☐	☐	☐	☐	☐
SECTION 2	MODES OF OPERATION AND PIN DESCRIPTIONS	☐	☐	☐	☐	☐
SECTION 3	MEMORY AND REGISTERS	☐	☐	☐	☐	☐
SECTION 4	INPUT/OUTPUT PORTS	☐	☐	☐	☐	☐
SECTION 5	CORE TIMER	☐	☐	☐	☐	☐
SECTION 6	PROGRAMMABLE TIMERS A AND B	☐	☐	☐	☐	☐
SECTION 7	LIQUID CRYSTAL DISPLAY DRIVERS	☐	☐	☐	☐	☐
SECTION 8	RESETS AND INTERRUPTS	☐	☐	☐	☐	☐
SECTION 9	CPU CORE AND INSTRUCTION SET	☐	☐	☐	☐	☐
SECTION 10	ELECTRICAL SPECIFICATIONS	☐	☐	☐	☐	☐
SECTION 11	MECHANICAL DATA	☐	☐	☐	☐	☐
SECTION 12	ORDERING INFORMATION	☐	☐	☐	☐	☐

Comments: _____

6. Have you found any errors? If so, please comment: _____

7. From your point of view, is anything missing from the document? If so, please say what: _____

8. How could we improve this document? _____

_____

_____

9. How would you rate Motorola's documentation?

	Excellent			Poor	
– In general	☐	☐	☐	☐	
– Against other semiconductor suppliers	☐	☐	☐	☐	

10. Which semiconductor manufacturer provides the best technical documentation? _____

11. Which company (in any field) provides the best technical documentation? _____

12. How many years have you worked with microprocessors?

Less than 1 year ☐ 1–3 years ☐ 3–5 years ☐ More than 5 years ☐

– Second fold back along this line –

By air mail
Par avion

IBRS NUMBER PHQ-B/207/G
CCRI NUMERO PHQ-B/207/G

NE PAS AFFRANCHIR

NO STAMP REQUIRED

REPONSE PAYEE
GRANDE-BRETAGNE

Motorola Ltd.,
Colvilles Road,
Kelvin Industrial Estate,
EAST KILBRIDE,
G75 8BR.
GREAT BRITAIN.

F.A.O. Technical Publications Manager
(re: MC68HC05L4/D)

!

MOTOROLA LTD.
Semiconductor Products Sector

– First fold back along this line –

– Cut along this line to remove –

– Third fold back along this line –

13. Currently there is some discussion in the semiconductor industry regarding a move towards providing data sheets in electronic form. If you have any opinion on this subject, please comment. _____

_____

14. We would be grateful if you would supply the following information (at your discretion), or attach your card.

Name: _____ Phone No: _____

Position: _____ FAX No: _____

Department: _____

Company: _____

Address: _____

_____

Thank you for helping us improve our documentation,
Graham Livey, Technical Publications Manager, Motorola Ltd., Scotland.

– Last tuck this edge into opposite flap –

Literature Distribution Centres:

EUROPE: Motorola Ltd., European Literature Centre, 88 Tanners Drive, Blakelands, Milton Keynes, MK14 5BP, England.

ASIA PACIFIC: Motorola Semiconductors (H.K.) Ltd., Silicon Harbour Center, No. 2, Dai King Street, Tai Po Industrial Estate, Tai Po, N.T., Hong Kong.

JAPAN: Nippon Motorola Ltd., 4-32-1, Nishi-Gotanda, Shinagawa-ku, Tokyo 141, Japan.

USA: Motorola Literature Distribution, P.O. Box 20912, Phoenix, Arizona 85036.

!MOTOROLA

MC68HC05L4/D