MC68HC11KW1/D

MC68HC11KW1

HC11

TECHNICAL DATA

MC68HC11KW1

TECHNICAL
DATA

MOTOROLA

MOTOROLA

MC68HC11KW1

High-density complementary metal oxide semiconductor (HCMOS) microcontroller unit

All Trade Marks recognized. This document contains information on new products. Specifications and information herein are subject to change without notice.

All products are sold on Motorola's Terms & Conditions of Supply. In ordering a product covered by this document the Customer agrees to be bound by those Terms & Conditions and nothing contained in this document constitutes or forms part of a contract (with the exception of the contents of this Notice). A copy of Motorola's Terms & Conditions of Supply is available on request.

Motorola reserves the right to make changes without further notice to any products herein. Motorola makes no warranty, representation or guarantee regarding the suitability of its products for any particular purpose, nor does Motorola assume any liability arising out of the application or use of any product or circuit, and specifically disclaims any and all liability, including without limitation consequential or incidental damages. "Typical" parameters can and do vary in different applications. All operating parameters, including "Typicals", must be validated for each customer application by customer's technical experts. Motorola does not convey any license under its patent rights nor the rights of others. Motorola products are not designed, intended, or authorized for use as components in systems intended for surgical implant into the body, or other applications intended to support or sustain life, or for any other application in which the failure of the Motorola product could create a situation where personal injury or death may occur. Should Buyer purchase or use Motorola products for any such unintended or unauthorized application, Buyer shall indemnify and hold Motorola and its officers, employees, subsidiaries, affiliates, and distributors harmless against all claims, costs, damages, and expenses, and reasonable attorney fees arising out of, directly or indirectly, any claim of personal injury or death associated with such unintended or unauthorized use, even if such claim alleges that Motorola was negligent regarding the design or manufacture of the part. Motorola and Ⓜ are registered trademarks of Motorola, Inc. Motorola, Inc. is an Equal Opportunity/Affirmative Action Employer.

The Customer should ensure that it has the most up to date version of the document by contacting its local Motorola office. This document supersedes any earlier documentation relating to the products referred to herein. The information contained in this document is current at the date of publication. It may subsequently be updated, revised or withdrawn.

© MOTOROLA LTD., 1997

Conventions

Where abbreviations are used in the text, an explanation can be found in the glossary, at the back of this manual. Register and bit mnemonics are defined in the paragraphs describing them.

An overbar is used to designate an active-low signal, eg: $\overline{\text{RESET}}$.

Because the bits in any one register are not necessarily linked by a common function, the description of a register may appear in several sections referring to different aspects of device operation. A full description of a bit is given only in a section in which it has relevance. Elsewhere, it appears shaded in the register diagram and is only briefly described.

When the state of a bit on reset is described as ‘x’, this means that its state depends on factors such as the operating mode selected. A ‘u’ indicates that the bit’s state on reset is undefined.

TABLE OF CONTENTS

Paragraph Number	TITLE	Page Number
	1 INTRODUCTION	
1.1	Features	1-1
1.2	Mask option	1-2
	2 PIN DESCRIPTIONS	
2.1	RESET	2-2
2.2	Crystal driver and external clock input (XTAL, EXTAL)	2-3
2.3	VDD and VSS	2-4
2.4	E clock output (E)	2-4
2.5	XOUT	2-4
2.6	Interrupt request ($\overline{IRQ}$)	2-4
2.7	Nonmaskable interrupt ($\overline{XIRQ}$)	2-5
2.8	MODA and MODB (MODA/$\overline{LIR}$ and MODB/VSTBY)	2-5
2.9	VRH and VRL	2-6
2.10	R/W	2-6
2.11	Port signals	2-6
2.11.1	Port A	2-6
2.11.2	Port B	2-8
2.11.3	Port C	2-8
2.11.4	Port D	2-8
2.11.5	Port E	2-9
2.11.6	Port F	2-9
2.11.7	Port G	2-9
2.11.8	Port H	2-10
2.11.9	Port J	2-10
2.11.10	Port K	2-10

3
CENTRAL PROCESSING UNIT

3.1 Registers 3-1
3.1.1 Accumulators A, B and D 3-2
3.1.2 Index register X (IX) 3-2
3.1.3 Index register Y (IY) 3-2
3.1.4 Stack pointer (SP) 3-2
3.1.5 Program counter (PC) 3-4
3.1.6 Condition code register (CCR) 3-4
3.1.6.1 Carry/borrow (C) 3-5
3.1.6.2 Overflow (V) 3-5
3.1.6.3 Zero (Z) 3-5
3.1.6.4 Negative (N) 3-5
3.1.6.5 Interrupt mask (I) 3-5
3.1.6.6 Half carry (H) 3-6
3.1.6.7 X interrupt mask (X) 3-6
3.1.6.8 Stop disable (S) 3-6
3.2 Data types 3-6
3.3 Opcodes and operands 3-7
3.4 Addressing modes 3-7
3.5 Immediate (IMM) 3-7
3.5.1 Direct (DIR) 3-7
3.5.2 Extended (EXT) 3-8
3.5.3 Indexed (IND, X; IND, Y) 3-8
3.5.4 Inherent (INH) 3-8
3.5.5 Relative (REL) 3-8
3.6 Instruction set 3-8

4
OPERATING MODES AND ON-CHIP MEMORY

4.1 Operating modes 4-1
4.1.1 Single chip operating mode 4-1
4.1.2 Expanded operating mode 4-1
4.1.3 Special test mode 4-2
4.1.4 Special bootstrap mode 4-2
4.2 On-chip memory 4-3
4.2.1 Mapping allocations 4-3
4.2.1.1 RAM 4-4
4.2.1.2 Bootloader ROM 4-4
4.2.2 Registers 4-4
4.3 System initialization 4-10
4.3.1 Mode selection 4-10
4.3.1.1 HPRIO — Highest priority I-bit interrupt & misc. register 4-11

4.3.2 Initialization 4-12
4.3.2.1 CONFIG — System configuration register 4-12
4.3.2.2 INIT — RAM and I/O mapping register 4-13
4.3.2.3 INIT2 — EEPROM mapping register....4-15
4.3.2.4 OPTION — System configuration options register 1....4-15
4.3.2.5 OPT2 — System configuration options register 2 4-17
4.3.2.6 BPROT — Block protect register....4-18
4.3.2.7 TMSK2 — Timer interrupt mask register 2....4-20
4.3.2.8 TCTL4 and TCTL6 — Timer 2 and 3 control registers 4-21
4.4 Memory expansion 4-22
4.4.1 Memory expansion logic 4-22
4.4.2 Extended addressing 4-23
4.4.3 Memory expansion examples 4-24
4.4.4 MMSIZ — Memory mapping window size register....4-29
4.4.5 MMWBR – Memory mapping window base register....4-30
4.4.6 MM1CR, MM2CR – Memory mapping window 1 and 2 control registers 4-31
4.4.7 PGAR — Port G assignment register 4-32
4.5 Chip selects 4-32
4.5.1 Chip select priorities....4-33
4.5.2 Program chip select 4-33
4.5.3 I/O chip select 4-33
4.5.4 CSCTL — Chip select control register....4-34
4.5.5 General-purpose chip selects 4-35
4.5.5.1 GPCS1A — General-purpose chip select 1 address register 4-35
4.5.5.2 GPCS1C — General-purpose chip select 1 control register 4-36
4.5.5.3 GPCS2A — General-purpose chip select 2 address register 4-37
4.5.5.4 GPCS2C — General-purpose chip select 2 control register 4-37
4.5.6 One chip select driving another 4-38
4.5.7 Clock stretching 4-39
4.5.7.1 CSCSTR — Chip select clock stretch register 4-39
4.6 EEPROM and CONFIG register 4-41
4.6.1 EEPROM 4-41
4.6.1.1 PPROG — EEPROM programming control register 4-41
4.6.1.2 EEPROM bulk erase 4-43
4.6.1.3 EEPROM row erase 4-43
4.6.1.4 EEPROM byte erase 4-44
4.6.2 CONFIG register programming....4-44
4.6.3 RAM and EEPROM security 4-45

5
RESETS AND INTERRUPTS

5.1 Resets 5-1
5.1.1 Power-on reset....5-1
5.1.2 External reset ($\overline{\text{RESET}}$) 5-2

5.1.3 COP reset 5-2
5.1.3.1 COPRST — Arm/reset COP timer circuitry register 5-3
5.1.4 Clock monitor reset 5-3
5.1.5 OPTION — System configuration options register 1 5-4
5.1.6 CONFIG — Configuration control register 5-5
5.2 Effects of reset 5-6
5.2.1 Central processing unit 5-7
5.2.2 Memory map 5-7
5.2.3 Parallel I/O 5-7
5.2.4 Timer 1 5-7
5.2.5 Timers 2 and 3 5-8
5.2.6 Real-time interrupt (RTI) 5-8
5.2.7 Pulse accumulator 5-8
5.2.8 Computer operating properly (COP) 5-8
5.2.9 Serial communications interface (SCI) 5-8
5.2.10 Serial peripheral interface (SPI) 5-9
5.2.11 Analog-to-digital converter 5-9
5.2.12 System 5-9
5.3 Reset and interrupt priority 5-9
5.3.1 HPRIO — Highest priority I-bit interrupt and misc. register 5-10
5.4 Interrupts 5-13
5.4.1 Interrupt recognition and register stacking 5-13
5.4.2 Nonmaskable interrupt request ($\overline{\text{XIRQ}}$) 5-14
5.4.3 Illegal opcode trap 5-14
5.4.4 Software interrupt 5-14
5.4.5 Maskable interrupts 5-15
5.4.6 Reset and interrupt processing 5-15
5.5 Low power operation 5-15
5.5.1 WAIT 5-15
5.5.2 STOP 5-16

6
PARALLEL INPUT/OUTPUT

6.1 Port A 6-2
6.1.1 PORTA — Port A data register 6-2
6.1.2 DDRA — Data direction register for port A 6-2
6.2 Port B 6-3
6.2.1 PORTB — Port B data register 6-3
6.2.2 DDRB — Data direction register for port B 6-3
6.3 Port C 6-4
6.3.1 PORTC — Port C data register 6-4
6.3.2 DDRC — Data direction register for port C 6-4
6.4 Port D 6-5
6.4.1 PORTD — Port D data register 6-5

6.4.2 DDRD — Data direction register for port D 6-5
6.5 Port E 6-6
6.5.1 PORTE — Port E data register 6-6
6.6 Port F 6-7
6.6.1 PORTF — Port F data register 6-7
6.6.2 DDRF — Data direction register for port F 6-7
6.7 Port G 6-8
6.7.1 PORTG — Port G data register 6-8
6.7.2 DDRG — Data direction register for port G 6-9
6.7.3 PGAR — Port G assignment register 6-9
6.8 Port H 6-10
6.8.1 PORTH — Port H data register 6-10
6.8.2 DDRH — Data direction register for port H 6-10
6.9 Port J 6-11
6.9.1 PORTJ — Port J data register 6-11
6.9.2 DDRJ — Data direction register for port J 6-11
6.10 Port K 6-12
6.10.1 PORTK — Port K data register 6-12
6.10.2 DDRK — Data direction register for port K 6-12
6.11 Internal pull-up resistors 6-13
6.11.1 PPAR — Port pull-up assignment register 6-13
6.12 System configuration 6-14
6.12.1 OPT2 — System configuration options register 2 6-14
6.12.2 CONFIG — System configuration register 6-15

7
SERIAL COMMUNICATIONS INTERFACE

7.1 Data format 7-2
7.2 Transmit operation 7-2
7.3 Receive operation 7-2
7.4 Wake-up feature 7-4
7.4.1 Idle-line wake-up 7-4
7.4.2 Address-mark wake-up 7-4
7.5 SCI error detection 7-5
7.6 SCI registers 7-5
7.6.1 SCBDH, SCBDL — SCI baud rate control registers 7-6
7.6.2 SCCR1 — SCI control register 1 7-7
7.6.3 SCCR2 — SCI control register 2 7-9
7.6.4 SCSR1 — SCI status register 1 7-10
7.6.5 SCSR2 — SCI status register 2 7-11
7.6.6 SCDRH, SCDRL — SCI data high/low registers 7-12
7.7 Status flags and interrupts 7-12
7.7.1 Receiver flags 7-13

8
SERIAL PERIPHERAL INTERFACE

8.1 Functional description 8-1
8.2 SPI transfer formats 8-2
8.2.1 Clock phase and polarity controls 8-3
8.3 SPI signals 8-3
8.3.1 Master in slave out 8-4
8.3.2 Master out slave in 8-4
8.3.3 Serial clock 8-4
8.3.4 Slave select 8-4
8.4 SPI system errors 8-5
8.5 SPI registers 8-5
8.5.1 SPCR — SPI control register 8-6
8.5.2 SPSR — SPI status register 8-8
8.5.3 SPDR — SPI data register 8-9
8.5.4 OPT2 — System configuration options register 2 8-9

9
TIMING SYSTEM

9.1 Timer 1 9-1
9.1.1 Timer 1 structure 9-3
9.1.2 Input capture 9-4
9.1.2.1 TCTL2 — Timer control register 2 9-6
9.1.2.2 TIC1–TIC3 — Timer input capture registers 9-7
9.1.2.3 TI4/O5 — Timer input capture 4/output compare 5 register 9-7
9.1.3 Output compare 9-8
9.1.3.1 TOC1–TOC4 — Timer output compare registers 9-9
9.1.3.2 CFORC — Timer compare force register 9-9
9.1.3.3 OC1M — Output compare 1 mask register 9-10
9.1.3.4 OC1D — Output compare 1 data register 9-10
9.1.3.5 TCNT — Timer counter register 9-11
9.1.3.6 TCTL1 — Timer control register 1 9-11
9.1.3.7 TMSK1 — Timer interrupt mask register 1 9-12
9.1.3.8 TFLG1 — Timer interrupt flag register 1 9-13
9.1.3.9 TMSK2 — Timer interrupt mask register 2 9-14
9.1.3.10 TFLG2 — Timer interrupt flag register 2 9-15
9.2 Timer 2 9-15
9.2.1 Output compare 9-18
9.2.2 Input capture 9-18
9.2.3 F23FRC — Compare force register for Timers 2 and 3. 9-18
9.2.4 T2C4 — Timer 2 channel 4 register 9-19
9.2.5 T2OC1–T2OC3 — Timer 2 output compare registers 9-19
9.2.6 TCNT2 — Timer 2 counter register 9-20

9.2.7 TCTL3 — Timer control register 3 (Timer 2) 9-20
9.2.8 TCTL4 — Timer control register 4 (Timer 2) 9-21
9.2.9 T2MSK — Timer 2 interrupt mask register....9-22
9.2.10 T2FLG — Timer 2 interrupt flag register 9-23
9.3 Timer 3 9-24
9.3.1 T3C4 — Timer 3 channel 4 register 9-24
9.3.2 T3OC1–T3OC3 — Timer 3 output compare registers 9-26
9.3.3 TCNT3 — Timer 3 counter register....9-26
9.3.4 TCTL5 — Timer control register 5 (Timer 3) 9-27
9.3.5 TCTL6 — Timer control register 6 (Timer 3) 9-27
9.3.6 T3MSK — Timer 3 interrupt mask register....9-29
9.3.7 T3FLG — Timer 3 interrupt flag register 9-30
9.4 Real-time interrupt 9-31
9.4.1 TMSK2 — Timer interrupt mask register 2....9-31
9.4.2 TFLG2 — Timer interrupt flag register 2 9-32
9.4.3 PACTL — Pulse accumulator control register 9-33
9.5 Computer operating properly watchdog function 9-33
9.6 Pulse accumulator 9-33
9.6.1 PACTL — Pulse accumulator control register 9-35
9.6.2 PACNT — Pulse accumulator count register 9-36
9.6.3 Pulse accumulator status and interrupt bits 9-36
9.6.3.1 TMSK2 — Timer interrupt mask 2 register....9-36
9.6.3.2 TFLG2 — Timer interrupt flag 2 register 9-36
9.7 Pulse-width modulation (PWM) timer 9-37
9.7.1 PWM timer block diagram 9-38
9.7.2 PWCLK — PWM clock prescaler and 16-bit select register....9-38
9.7.2.1 16-bit PWM function 9-38
9.7.2.2 Clock prescaler selection 9-40
9.7.3 PWPOL — PWM timer polarity & clock source select register 9-41
9.7.4 PWSCAL — PWM timer prescaler register....9-41
9.7.5 PWEN — PWM timer enable register 9-42
9.7.6 PWCNT1–4 — PWM timer counter registers 1 to 4....9-43
9.7.7 PWPER1–4 — PWM timer period registers 1 to 4....9-43
9.7.8 PWDTY1–4 — PWM timer duty cycle registers 1 to 4....9-44
9.7.9 Boundary cases 9-44

10
ANALOG-TO-DIGITAL CONVERTER

10.1 Conversion process 10-2
10.2 Channel assignments 10-2
10.3 Single channel operation 10-3
10.3.1 4-conversion, single scan....10-4
10.3.2 4-conversion, continuous scan....10-4
10.3.3 8-conversion, single scan....10-4

10.3.4 8-conversion, continuous scan 10-4
10.4 Multiple channel operation 10-4
10.4.1 4-channel single scan 10-5
10.4.2 4-channel continuous scan 10-5
10.4.3 8-channel single scan 10-5
10.4.4 8-channel continuous scan 10-5
10.5 Power-up and clock select 10-5
10.6 Operation in STOP and WAIT modes 10-6
10.7 Registers 10-6
10.7.1 ADCTL — A/D control and status register 10-6
10.7.2 ADFRQ — A/D converter frequency select register 10-7
10.7.3 ADR1 — ADR8 A/D result registers 10-8

A
ELECTRICAL SPECIFICATIONS

A.1 Maximum ratings A-1
A.2 Thermal characteristics and power considerations A-2
A.3 Test methods A-3
A.4 DC electrical characteristics A-4
A.5 Control timing A-5
A.5.1 Peripheral port timing A-8
A.5.2 Analog-to-digital converter characteristics A-10
A.5.3 Serial peripheral interface timing A-11
A.5.4 Non-multiplexed expansion bus timing A-14
A.6 EEPROM characteristics A-15

B
MECHANICAL DATA

B.1 Packaging B-1

C
DEVELOPMENT SYSTEMS

C.1 EVS — Evaluation system C-1
C.2 MMDS11 — Motorola modular development system C-2
C.3 SPGMR11 — Serial peripheral system C-2

LIST OF FIGURES

Figure Number	TITLE	Page Number
1-1	MC68HC11KW1 block diagram	1-3
2-1	MC68HC11KW1 100-pin TQFP	2-1
2-2	External reset circuitry	2-2
2-3	Oscillator connections	2-3
2-4	RAM stand-by connections	2-5
3-1	Programming model	3-1
3-2	Stacking operations	3-3
4-1	MC68HC11KW1 memory map	4-3
4-2	RAM and register overlap	4-14
4-3	Memory map example of memory expansion	4-25
4-4	Schematic example of memory expansion	4-26
4-5	Memory map example of memory expansion	4-27
4-6	Schematic example of memory expansion	4-28
5-1	Processing flow out of reset (1 of 2)	5-17
5-2	Processing flow out of reset (2 of 2)	5-18
5-3	Interrupt priority resolution (1 of 3)	5-19
5-4	Interrupt priority resolution (2 of 3)	5-20
5-5	Interrupt priority resolution (3 of 3)	5-21
5-6	Interrupt source resolution within the SCI subsystem	5-22
7-1	SCI baud rate generator circuit diagram	7-1
7-2	SCI block diagram	7-3
7-3	Interrupt source resolution within SCI	7-14
8-1	SPI block diagram	8-2
8-2	SPI transfer format	8-3
9-1	Timer clock divider chains	9-2
9-2	Timer 1 capture/compare block diagram	9-5
9-3	Timer 2 capture/compare block diagram	9-17
9-4	Timer 3 capture/compare block diagram	9-25
9-5	Pulse accumulator block diagram	9-34
9-6	PWM timer block diagram	9-39
9-7	PWM duty cycle	9-44
A-1	Test methods	A-3
A-2	Timer inputs	A-5

Figure Number	TITLE	Page Number
A-3	Reset timing	A-6
A-4	Interrupt timing	A-6
A-5	STOP recovery timing	A-7
A-6	WAIT recovery timing	A-7
A-7	Port read timing diagram	A-8
A-8	Port G control timing	A-8
A-9	Port write timing diagram	A-9
A-10	SPI master timing (CPHA = 0)	A-12
A-11	SPI master timing (CPHA = 1)	A-12
A-12	SPI slave timing (CPHA = 0)	A-13
A-13	SPI slave timing (CPHA = 1)	A-13
A-14	Expansion bus timing	A-15
B-1	100-pin TQFP	B-1
B-2	100-pin TQFP mechanical dimensions	B-2

LIST OF TABLES

Table Number	TITLE	Page Number
2-1	Port signal functions	2-7
3-1	Reset vector comparison	3-4
3-2	Instruction set	3-9
4-1	Example bootloader baud rates	4-2
4-2	Register and control bit assignments	4-5
4-3	Registers with limited write access	4-10
4-4	Hardware mode select summary	4-11
4-5	RAM and register remapping	4-14
4-6	EEPROM remapping	4-15
4-7	XCLK frequencies	4-18
4-8	EEPROM block protect	4-19
4-9	CPU address and address expansion signals	4-24
4-10	Window size select	4-29
4-11	Memory expansion window base address	4-30
4-12	Chip select priorities	4-33
4-13	Program chip select size	4-34
4-14	General purpose chip select priority	4-35
4-15	General-purpose chip select 1 size control	4-37
4-16	General-purpose chip select 2 size control	4-38
4-17	One chip select driving another	4-39
4-18	Chip select control parameter summary	4-40
4-19	Erase mode selection	4-42
5-1	COP timer rate select	5-2
5-2	Reset cause, reset vector and operating mode	5-6
5-3	Highest priority interrupt selection	5-11
5-4	Interrupt and reset vector assignments	5-12
5-5	Stacking order on entry to interrupts	5-13
6-1	Port configuration	6-1
7-1	Example SCI baud rate control values	7-7
8-1	SPI clock rates	8-7
9-1	Timer 1 resolution and capacity	9-3
9-2	RTI periodic rates	9-31
9-3	Pulse accumulator timing	9-34

9-4 Clock A and clock B prescalers 9-40
10-1 Channel assignments 10-3
C-1 M68HC11 development tools C-1

1
INTRODUCTION

The MC68HC11KW1 8-bit microcontroller is a member of the M68HC11 family of HCMOS microcontrollers. It has 640 bytes of EEPROM and 768 bytes of RAM. Making use of a 100-pin TQFP package, a non-multiplexed expanded bus is a feature of this device. The main timer system includes three input captures, four output compares and a software selectable input capture or output compare. There are two additional 16-bit timers, each with three output compares and one software selectable input capture or output compare. Other major features of this device are: a 10-channel, 10-bit resolution A/D converter, four PWM timer channels, an SPI (serial peripheral interface) and an enhanced SCI (serial communications interface). In common with other family members, the MC68HC11KW1 also includes an 8-bit pulse accumulator circuit, a real time interrupt facility, and a computer operating properly watchdog system. This device is intended for use in expanded memory applications.

1.1 Features

- Low power, high performance M68HC11 CPU core with 4 MHz internal bus frequency
- 768 bytes of RAM
- 640 bytes of byte-erasable EEPROM, with on-chip charge pump
- 448 bytes of boot ROM
- Up to 70 general purpose I/O lines, plus up to 10 input-only lines
- Non-multiplexed address and data buses, permitting direct access to the full 64K byte address map
- Memory expansion unit, with six address extension lines, allowing up to (for example) sixteen 32K byte banks of external memory to be addressed in either of two bank windows
- Four external chip selects
- 16-bit timer with 3/4 input captures and 5/4 output compares; pulse accumulator and COP watchdog timer
- Real-time interrupt circuit

- Two additional 16-bit timers, each with 3 output compares and one input capture or output compare (may be externally clocked, if required, for external event counter operation)
- SCI subsystem (NRZ type for compatibility with standard RS232 systems) with parity and a modulus prescaler
- SPI subsystem, with software selectable MSB/LSB first option and increased baud rate selection range
- 10-channel, 10-bit analog-to-digital converter
- Four 8-bit PWM timer channels
- Available in 100-pin TQFP package

1.2 Mask option

There is a single mask option on the MC68HC11KW1, which is programmed during manufacture and must be specified on the order form:

- Security option (available/unavailable). See Section 4.6.3

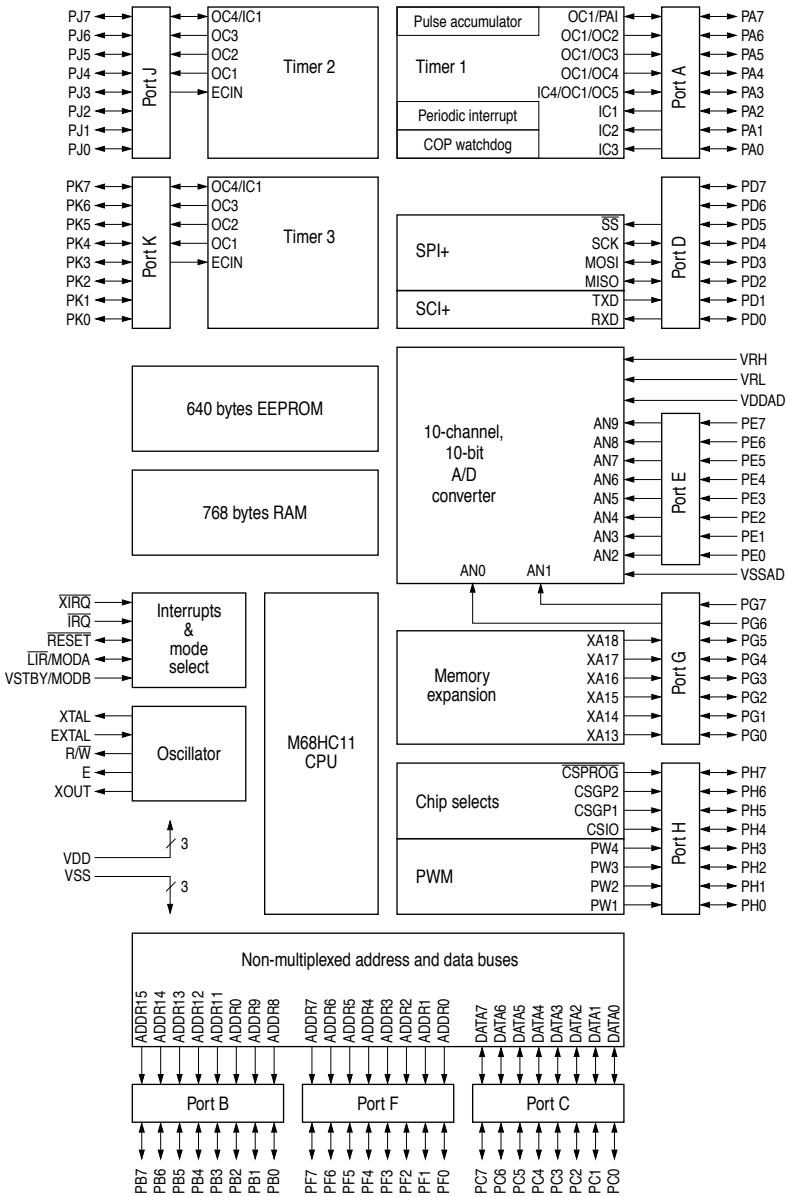


Figure 1-1 MC68HC11KW1 block diagram

THIS PAGE LEFT BLANK INTENTIONALLY

2
PIN DESCRIPTIONS

The MC68HC11KW1 is available packaged in a 100-pin thin quad flat pack (TQFP), as shown in Figure 2-1. Most pins on this MCU serve two or more functions, as described in the following paragraphs.

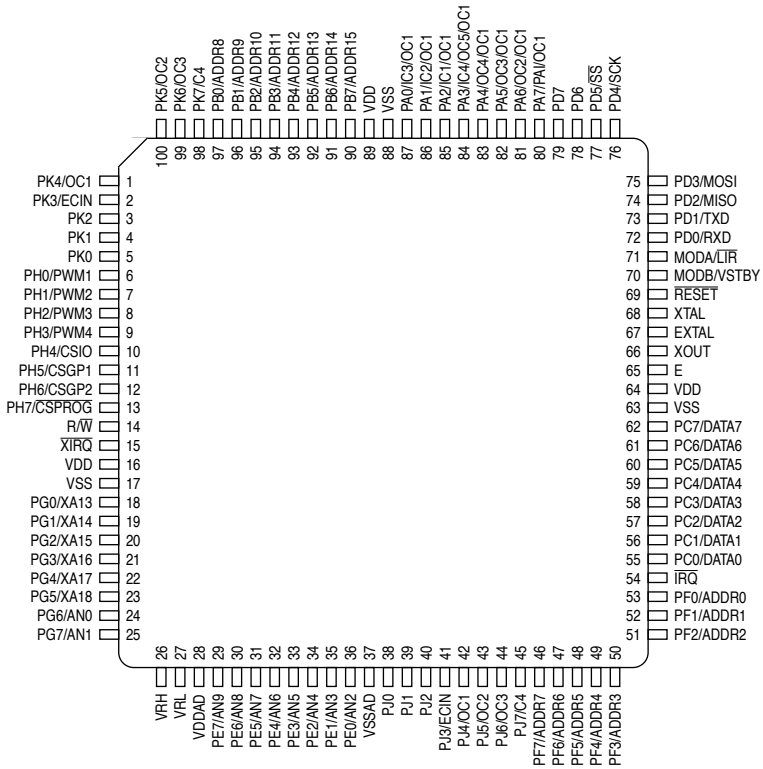


Figure 2-1 MC68HC11KW1 100-pin TQFP

2.1 $\overline{\text{RESET}}$

An active-low, bidirectional control signal, $\overline{\text{RESET}}$ acts as an input to initialize the MCU to a known start-up state. It also acts as an open-drain output to indicate that an internal failure has been detected in either the clock monitor or the COP watchdog circuit. The CPU distinguishes between internal and external reset conditions by sensing whether the reset pin rises to a logic one in less than four E clock cycles after an internal reset has been released. It is therefore not advisable to connect an external resistor-capacitor (RC) power-up delay circuit to the reset pin of M68HC11 devices because the circuit charge time constant can cause the device to misinterpret the type of reset that occurred. Refer to Section 5 for further information.

Figure 2-2 illustrates a typical reset circuit that includes an external switch together with a low voltage inhibit circuit, to prevent power transitions, or RAM or EEPROM corruption.

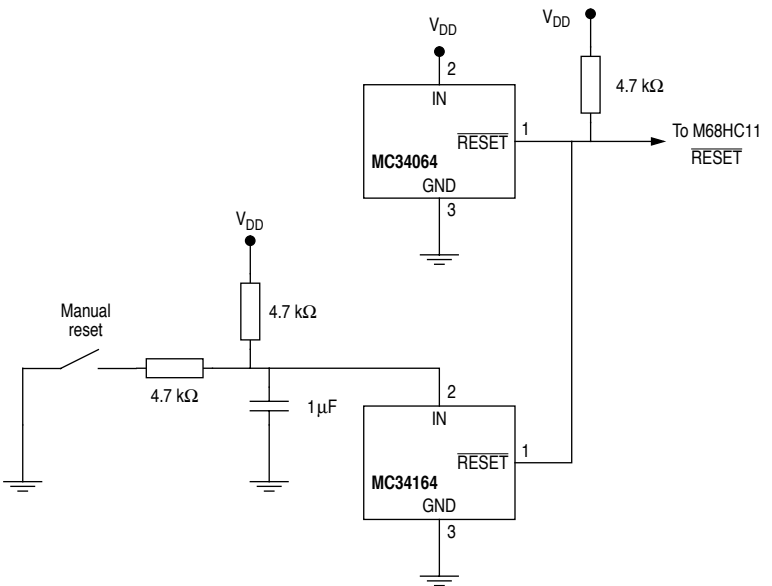


Figure 2-2 External reset circuitry

2.2 Crystal driver and external clock input (XTAL, EXTAL)

These two pins provide the interface for either a crystal or a CMOS compatible clock to control the internal clock generator circuitry. The frequency applied to these pins must be four times higher than the desired E clock rate. Refer to Figure 2-3.

The XTAL pin is normally left unconnected when an external CMOS compatible clock input is connected to the EXTAL pin. The XTAL output is normally intended to drive only a crystal. The XTAL output can be buffered with a high-impedance buffer, or it can be used to drive the EXTAL input of another M68HC11 family device.

In all cases, use caution when designing circuitry associated with the oscillator pins.

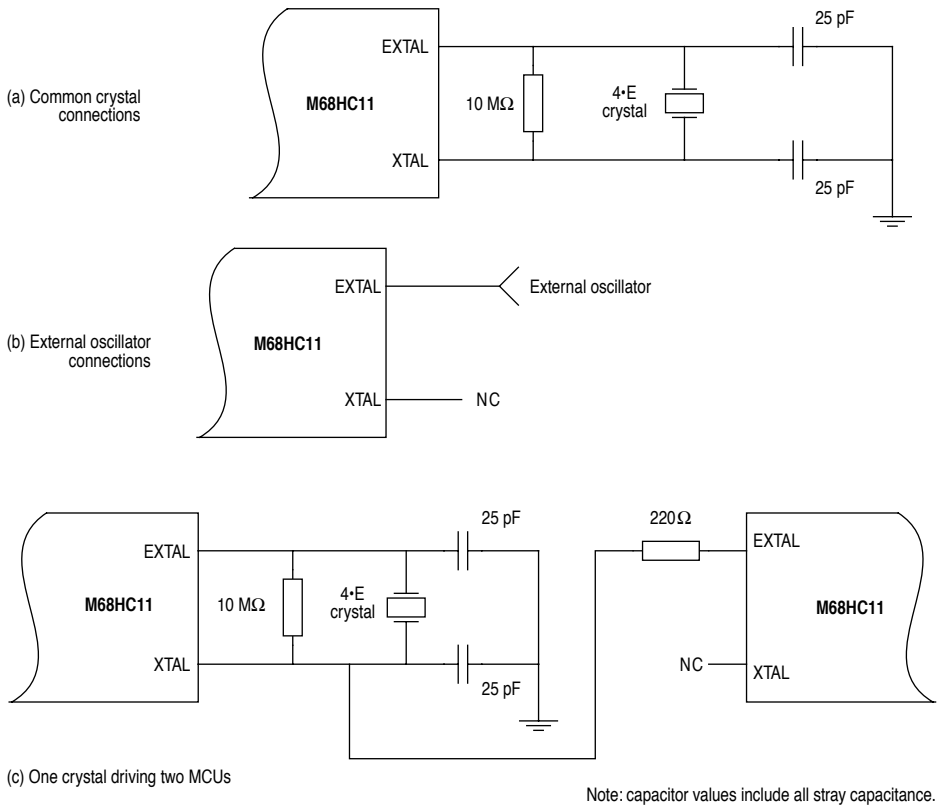


Figure 2-3 Oscillator connections

2.3 VDD and VSS

Power is supplied to the microcontroller via these pins. VDD is the positive supply and VSS is ground. The MCU operates from a 5V (nominal) power supply.

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 bypassing at the MCU. Bypass capacitors should have good high-frequency characteristics and be as close to the MCU as possible. Bypassing requirements vary, depending on how heavily the MCU pins are loaded.

The MC68HC11KW1 has four VDD pins and four VSS pins. One pair of these pins is reserved for supplying power to the analog-to-digital converter (VDDAD, VSSAD); the remaining pins are used for the internal logic, and to supply power to the port logic on either half of the chip.

2.4 E clock output (E)

E is the output connection for the internally generated E clock. The signal from E is used as a timing reference. The frequency of the E clock output is one quarter that of the input frequency at the XTAL and EXTAL pins. When E clock output is low, an internal process is taking place; when it is high, data is being accessed. All clocks, including the E clock, are halted when the MCU is in STOP mode. The E clock output can be turned off in single-chip modes to reduce the effects of RFI (see Section 4.3.2.5).

2.5 XOUT

The XOUT pin outputs the buffered CLKX signal, if enabled by the XCLK bit in the CONFIG register. The frequency of CLKX can be selected using two bits in the OPT2 register (XDV1 and XDV2). On reset, CLKX has the same frequency as EXTAL (4E). See Section 4.

Note that the phase relationship between CLKX and EXTAL cannot be predicted.

2.6 Interrupt request ($\overline{\text{IRQ}}$)

The $\overline{\text{IRQ}}$ input provides a means of applying asynchronous interrupt requests to the MCU. Either falling-edge-sensitive triggering or level-sensitive triggering is program selectable (OPTION register). $\overline{\text{IRQ}}$ is always configured to level-sensitive-triggering at reset.

Note: Connect an external pull-up resistor, typically 4.7 kΩ, to VDD when $\overline{\text{IRQ}}$ is used in a level sensitive wired-OR configuration. See also Section 2.7.

2.7 Nonmaskable interrupt ($\overline{\text{XIRQ}}$)

The $\overline{\text{XIRQ}}$ input provides a means of requesting a nonmaskable interrupt after reset initialization. During reset, the X bit in the condition code register (CCR) is set and any interrupt is masked until MCU software enables it. $\overline{\text{XIRQ}}$ is often used as a power loss detect interrupt.

Whenever $\overline{\text{XIRQ}}$ or $\overline{\text{IRQ}}$ is used with multiple interrupt sources ($\overline{\text{IRQ}}$ must be configured for level-sensitive operation if there is more than one source of interrupt), each source must drive the interrupt input with an open-drain type of driver to avoid contention between outputs. There should be a single pull-up resistor near the MCU interrupt input pin (typically 4.7 kΩ). There must also be an interlock mechanism at each interrupt source so that the source holds the interrupt line low until the MCU recognizes and acknowledges the interrupt request. If one or more interrupt source is still pending after the MCU services a request, the interrupt line will still be held low and the MCU will be interrupted again as soon as the interrupt mask bit in the MCU is cleared (normally upon return from an interrupt). Refer to Section 5.

2.8 MODA and MODB (MODA/$\overline{\text{LIR}}$ and MODB/VSTBY)

During reset, MODA and MODB select one of the four operating modes. Refer to Section 4.

After the operating mode has been selected, the $\overline{\text{LIR}}$ pin provides an open-drain output (driven low) to indicate that execution of an instruction has begun. In order to detect consecutive instructions in a high-speed application, this signal drives high for a short time to prevent false triggering. A series of E clock cycles occurs during execution of each instruction. The $\overline{\text{LIR}}$ signal goes low during the first E clock cycle of each instruction (opcode fetch). This output is provided for assistance in program debugging, and its operation is controlled by the LIRDV bit in the OPT2 register.

The VSTBY pin is used to input RAM stand-by power. The MCU is powered from the VDD pin unless the difference between the level of V_{STBY} and V_{DD} is greater than one MOS threshold (about 0.7 volts). When these voltages differ by more than 0.7 volts, the internal RAM and part of the reset logic are powered from V_{STBY} rather than V_{DD}. This allows RAM contents to be retained without VDD power applied to the MCU. Reset must be driven low before V_{DD} is removed and must remain low until V_{DD} has been restored to a valid level.

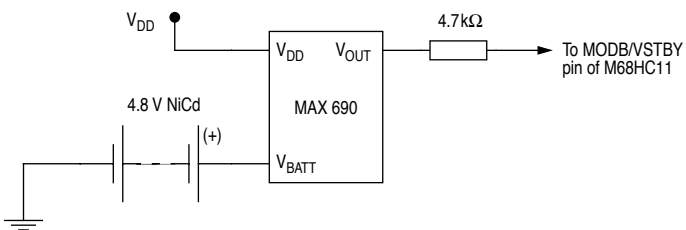


Figure 2-4 RAM stand-by connections

2.9 VRH and VRL

These pins provide the reference voltages for the analog-to-digital converter.

2.10 R/$\overline{W}$

In expanded and test modes, R/$\overline{W}$ performs the read/write function. R/$\overline{W}$ signals the direction of transfers on the external data bus. A high on this pin indicates that a read cycle is in progress.

In single chip mode the R/$\overline{W}$ signal is driven low.

2.11 Port signals

The MC68HC11KW1 includes 80 pins that are arranged into ten 8-bit ports (A, B, C, D, E, F, G, H, J and K). All the port pins are bidirectional, except for PG7, PG6 and port E pins [7:0]; these are input only. Most of the bidirectional ports serve a purpose other than I/O, depending on the operating mode or peripheral function selected. The input-only pins may be used as general-purpose inputs, or as inputs to the A/D converter. Note that ports B, C, and F are available for I/O functions only in single chip and bootstrap modes. Refer to Table 2-1 for details of the port signals' functions in different operating modes.

Note: When using the information about port functions, do not confuse pin function with the electrical state of the pin at reset. All general purpose I/O pins configured as inputs at reset are in a high-impedance state. Port data registers reflect the functional state of the port at reset. The pin function is mode dependent.

2.11.1 Port A

Port A is an 8-bit, general-purpose I/O port with a data register (PORTA) and a data direction register (DDRA). Port A pins share functions with the main 16-bit timer system, Timer 1 (see Section 9 for further information). PORTA can be read at any time and always returns the pin level. If written, PORTA stores the data in internal latches. The pins are driven only if they are configured as outputs. Writes to PORTA do not change the pin state when the pins are configured for timer output compares.

Out of reset, port A pins [7:0] are general purpose high-impedance inputs. When the functions associated with these pins are disabled, the bits in DDRA govern the I/O state of the associated pin. For further information, refer to Section 6.

Table 2-1 Port signal functions

Port/bit	Single chip and bootstrap mode	Expanded multiplexed and special test mode
PA7	PA7/PAI and/or OC1	
PA6	PA6/OC2 and/or OC1	
PA5	PA5/OC3 and/or OC1	
PA4	PA4/OC4 and/or OC1	
PA3	PA3/OC5/IC4 and/or OC1	
PA2	PA2/IC1	
PA1	PA1/IC2	
PA0	PA0/IC3	
PB[7:0]	PB[7:0]	ADDR[15:8]
PC[7:0]	PC[7:0]	DATA[7:0]
PD[7,6]	PD[7, 6]	
PD5	PD5/$\overline{SS}$	
PD4	PD4/SCK	
PD3	PD3/MOSI	
PD2	PD2/MISO	
PD1	PD1/TXD	
PD0	PD0/RXD	
PE[7:0]	Input only / analog inputs	
PF[7:0]	PF[7:0]	ADDR[7:0]
PG[7,6]	Input only / analog inputs	
PG[5:0]	PG[5:0]	PG[5:0] / XA[18:13]
PH7	PH7	PH7 / $\overline{CSPROG}$
PH6	PH6	PH6 / CSGP2
PH5	PH5	PH5 / CSGP1
PH4	PH4	PH4 / CSIO
PH[3:0]	PH[3:0] / PWM[4:1]	
PJ7	PJ7 / C4	
PJ6	PJ6 / OC3	
PJ5	PJ5 / OC2	
PJ4	PJ4 / OC1	
PJ3	PJ3 / ECIN	
PJ[2,0]	PJ[2,0]	
PK7	PK7 / C4	
PK6	PK6 / OC3	
PK5	PK5 / OC2	
PK4	PK4 / OC1	
PK3	PK3 / ECIN	
PK[2,0]	PK[2,0]	

2.11.2 Port B

Port B is an 8-bit, general-purpose I/O port with a data register (PORTB) and a data direction register (DDRB). In single chip mode, port B pins are general purpose I/O pins (PB[7:0]). In expanded mode, port B pins act as the high-order address lines (ADDR[15:8]) of the address bus.

PORTB can be read at any time and always returns the pin level. If PORTB is written, the data is stored in internal latches. The pins are driven only if they are configured as outputs in single chip or bootstrap mode. For further information, refer to Section 6.

Port B pins include on-chip pull-up devices which can be enabled or disabled via the port pull-up assignment register (PPAR).

2.11.3 Port C

Port C is an 8-bit, general-purpose I/O port with a data register (PORTC) and a data direction register (DDRC). In single chip mode, port C pins are general purpose I/O pins (PC[7:0]). In the expanded mode, port C pins are configured as data bus pins (DATA[7:0]).

PORTC can be read at any time; inputs return the pin level and outputs return the pin driver input level. If PORTC is written, the data is stored in internal latches. The pins are driven only if they are configured as outputs in single chip or bootstrap mode. Port C pins are general purpose inputs out of reset in single chip and bootstrap modes. In expanded and test modes, these pins are data bus lines out of reset.

The CWOM control bit in the OPT2 register disables port C's p-channel output drivers. Because the n-channel driver is not affected by CWOM, setting CWOM causes port C to become an open-drain-type output port suitable for wired-OR operation. In wired-OR mode (PORTC bits at logic level zero), the pins are actively driven low by the n-channel driver. When a port C bit is at logic level one, the associated pin is in a high impedance state as neither the n-channel nor the p-channel devices are active. It is customary to have an external pull-up resistor on lines that are driven by open-drain devices. Port C can only be configured for wired-OR operation when the MCU is in single chip mode. For further information, refer to Section 6.

2.11.4 Port D

Port D, an 8-bit, general-purpose I/O port, has a data register (PORTD) and a data direction register (DDRD). All the port D pins can be used for general purpose I/O, and pins [5:0] can also be used for the serial peripheral interface (SPI, pins [5:2]) and the serial communications interface (SCI, pins [1,0]).

PORTD can be read at any time; inputs return the pin level and outputs return the pin driver input level. If PORTD is written, the data is stored in internal latches. The pins are driven only if port D is configured for general purpose output.

The DWOM bit in SPCR disables the p-channel output drivers of pins D[5:2], and the WOMS bit in SCCR1 disables those of pins D[1,0]. Because the n-channel driver is not affected by DWOM or WOMS, setting either bit causes the corresponding port D pins to become open-drain-type outputs suitable for wired-OR operation. In wired-OR mode (PORTD bits at logic level zero), the pins are actively driven low by the n-channel driver. When a port D bit is at logic level one, the associated pin is in a high impedance state as neither the n-channel nor the p-channel devices are active. It is customary to have an external pull-up resistor on lines that are driven by open-drain devices. Port D can be configured for wired-OR operation when the MCU is in single chip or expanded mode.

For further information, refer to Section 6, Section 7 (SCI) and Section 8 (SPI).

2.11.5 Port E

Port E pins can be used as the analog inputs for the analog-to-digital converter, or as general-purpose inputs. For further information, refer to Section 6 and Section 10 (A/D).

2.11.6 Port F

Port F is an 8-bit, general-purpose I/O port with a data register (PORTF) and a data direction register (DDRF). In single chip mode, port F pins are general purpose I/O pins (PF[7:0]). In expanded mode, port F pins act as the low-order address lines (ADDR[7:0]) of the address bus.

PORTF can be read at any time and always returns the pin level. If PORTF is written, the data is stored in internal latches. The pins are driven only if they are configured as outputs in single chip or bootstrap mode.

Port F pins include on-chip pull-up devices that can be enabled or disabled via the port pull-up assignment register (PPAR). For further information, refer to Section 6.

2.11.7 Port G

In normal modes, Port G is an 8-bit general-purpose port with 6 I/O lines (PG[5:0]), and two input only lines (PG[7, 6]). Associated with port G are a data register (PORTG), a data direction register (DDRG) and an assignment register (PGAR). Pins [7, 6] can be used as general-purpose inputs, or as inputs to the analog-to-digital converter. The functions of pins [5:0] are controlled by bits in the port G assignment register (PGAR), which select whether the pins are used for general purpose I/O, or, in expanded mode, for the memory expansion lines XA[18:13].

PORTG can be read at any time and always returns the pin level. If PORTG is written, the data is stored into an internal latch. The pin is driven only if it is configured as an output.

Pins [5:0] have on-chip pull-up devices that can be enabled or disabled via the port pull-up assignment register (PPAR). Refer to Section 6, Section 10 (A/D) and Section 4.

2.11.8 Port H

Port H is an 8-bit general-purpose I/O port with a data register (PORTH) and a data direction register (DDRH). Port H lines can be used for general-purpose I/O, for chip select lines (PH[7:4]), and for the pulse width modulation timer (PWM, PH[3:0]).

PORTH can be read at any time and always returns the pin level. If PORTH is written, the data is stored into an internal latch. The pin is driven only if it is configured as an output.

Port H pins include on-chip pull-up devices that can be enabled or disabled via the port pull-up assignment register (PPAR). For further information, refer to Section 6, Section 9 (Timing system) and Section 4.

2.11.9 Port J

Port J is an 8-bit, general-purpose I/O port with a data register (PORTJ) and a data direction register (DDRJ). Port J lines can be used for general-purpose I/O, and pins [7:3] share functions with one of the 16-bit timers, Timer 2.

PORTJ can be read at any time and always returns the pin level. If written, PORTJ stores the data in internal latches. The pins are driven only if they are configured as outputs. Writes to PORTJ do not change the pin state when the pins are configured for timer output compares.

Out of reset, port J pins [7:0] are general purpose high-impedance inputs. When the functions associated with these pins are disabled, the bits in DDRJ govern the I/O state of the associated pin. For further information, refer to Section 6 and Section 9 (Timing system).

2.11.10 Port K

Port K is an 8-bit general-purpose I/O port with a data register (PORTK) and a data direction register (DDRK). Port K lines can be used for general-purpose I/O, and pins [7:3] share functions with one of the 16-bit timers, Timer 3.

PORTK can be read at any time and always returns the pin level. If written, PORTK stores the data in internal latches. The pins are driven only if they are configured as outputs. Writes to PORTK do not change the pin state when the pins are configured for timer output compares.

Out of reset, port K pins [7:0] are general purpose high-impedance inputs. When the functions associated with these pins are disabled, the bits in DDRK govern the I/O state of the associated pin. For further information, refer to Section 6 and Section 9 (Timing system).

3
CENTRAL PROCESSING UNIT

This section discusses the M68HC11 central processing unit (CPU) architecture, its addressing modes and the instruction set. For more detailed information on the instruction set, refer to the ***M68HC11 Reference Manual (M68HC11RM/AD)***.

The CPU is designed to treat all peripheral, I/O and memory locations identically, as addresses in the 64Kbyte memory map. This is referred to as memory-mapped I/O. There are no special instructions for I/O that are separate from those used for memory. This architecture also allows accessing an operand from an external memory location with no execution-time penalty.

3.1 Registers

M68HC11 CPU registers are an integral part of the CPU and are not addressed as if they were memory locations. The seven registers are shown in Figure 3-1 and are discussed in the following paragraphs.

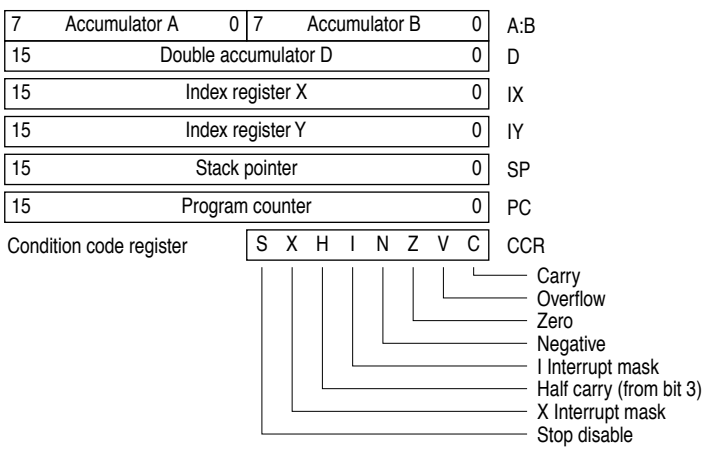


Figure 3-1 Programming model

3.1.1 Accumulators A, B and D

Accumulators A and B are general purpose 8-bit registers that hold operands and results of arithmetic calculations or data manipulations. For some instructions, these two accumulators are treated as a single double-byte (16-bit) accumulator called accumulator D. Although most operations can use accumulators A or B interchangeably, the following exceptions apply:

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

3.1.2 Index register X (IX)

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

3.1.3 Index register Y (IY)

The 16-bit IY register performs an indexed mode function similar to that of the IX register. However, most instructions using the IY register require an extra byte of machine code and an extra cycle of execution time because of the way the opcode map is implemented. Refer to Section 3.3 for further information.

3.1.4 Stack pointer (SP)

The M68HC11 CPU has an automatic program stack. This stack can be located anywhere in the address space and can be any size up to the amount of memory available in the system. Normally the SP is initialized by one of the first instructions in an application program. The stack is configured as a data structure that grows downward from high memory to low memory. Each time a new byte is pushed onto the stack, the SP is decremented. Each time a byte is pulled from the stack, the SP is incremented. At any given time, the SP holds the 16-bit address of the next free location in the stack. Figure 3-2 is a summary of SP operations.

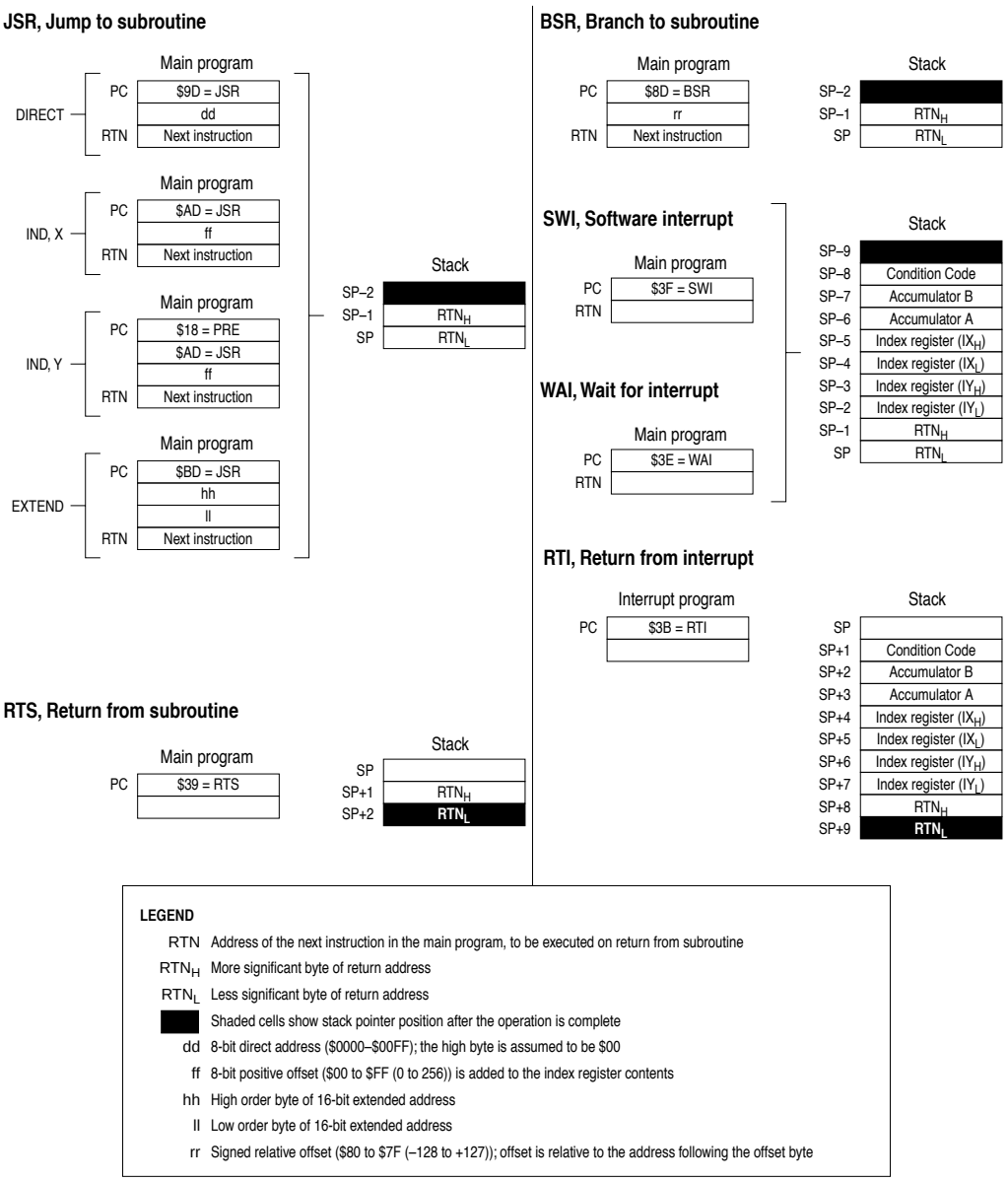


Figure 3-2 Stacking operations

When a subroutine is called by a jump to subroutine (JSR) or branch to subroutine (BSR) instruction, the address of the instruction after the JSR or BSR is automatically pushed onto the stack, less significant byte first. When the subroutine is finished, a return from subroutine (RTS) instruction is executed. The RTS pulls the previously stacked return address from the stack, and loads it into the program counter. Execution then continues at this recovered return address.

When an interrupt is recognized, the current instruction finishes normally, the return address (the current value in the program counter) is pushed onto the stack, all of the CPU registers are pushed onto the stack, and execution continues at the address specified by the vector for the interrupt. At the end of the interrupt service routine, an RTI instruction is executed. The RTI instruction causes the saved registers to be pulled off the stack in reverse order. Program execution resumes at the return address.

There are instructions that push and pull the A and B accumulators and the X and Y index registers. These instructions are often used to preserve program context. For example, pushing accumulator A onto the stack when entering a subroutine that uses accumulator A, and then pulling accumulator A off the stack just before leaving the subroutine, ensures that the contents of a register will be the same after returning from the subroutine as it was before starting the subroutine.

3.1.5 Program counter (PC)

The program counter, a 16-bit register, contains the address of the next instruction to be executed. After reset, the program counter is initialized from one of six possible vectors, depending on operating mode and the cause of reset.

Table 3-1 Reset vector comparison

	POR or RESET pin	Clock monitor	COP watchdog
Normal	$FFFE, $FFFF	$FFFC, $FFFD	$FFFA, $FFFB
Test or Boot	$BFFE, $BFFF	$BFFE, $BFFF	$BFFE, $BFFF

3.1.6 Condition code register (CCR)

This 8-bit register contains five condition code indicators (C, V, Z, N, and H), two interrupt masking bits, (IRQ and XIRQ) and a stop disable bit (S). In the M68HC11 CPU, condition codes are automatically updated by most instructions. For example, load accumulator A (LDAA) and store accumulator A (STAA) instructions automatically set or clear the N, Z, and V condition code flags. Pushes, pulls, add B to X (ABX), add B to Y (ABY), and transfer/exchange instructions do not affect the condition codes. Refer to Table 3-2, which shows the condition codes that are affected by a particular instruction.

3.1.6.1 Carry/borrow (C)

The C-bit is set if the arithmetic logic unit (ALU) performs a carry or borrow during an arithmetic operation. The C-bit also acts as an error flag for multiply and divide operations. Shift and rotate instructions operate with and through the carry bit to facilitate multiple-word shift operations.

3.1.6.2 Overflow (V)

The overflow bit is set if an operation causes an arithmetic overflow. Otherwise, the V-bit is cleared.

3.1.6.3 Zero (Z)

The Z-bit is set if the result of an arithmetic, logic, or data manipulation operation is zero. Otherwise, the Z-bit is cleared. Compare instructions do an internal implied subtraction and the condition codes, including Z, reflect the results of that subtraction. A few operations (INX, DEX, INY, and DEY) affect the Z-bit and no other condition flags. For these operations, only = and ≠ conditions can be determined.

3.1.6.4 Negative (N)

The N-bit is set if the result of an arithmetic, logic, or data manipulation operation is negative; otherwise, the N-bit is cleared. A result is said to be negative if its most significant bit (MSB) is set (MSB = 1). A quick way to test whether the contents of a memory location has the MSB set is to load it into an accumulator and then check the status of the N-bit.

3.1.6.5 Interrupt mask (I)

The interrupt request (IRQ) mask (I-bit) is a global mask that disables all maskable interrupt sources. While the I-bit is set, interrupts can become pending, but the operation of the CPU continues uninterrupted until the I-bit is cleared. After any reset, the I-bit is set by default and can only be cleared by a software instruction. When an interrupt is recognized, the I-bit is set after the registers are stacked, but before the interrupt vector is fetched. After the interrupt has been serviced, a return from interrupt instruction is normally executed, restoring the registers to the values that were present before the interrupt occurred. Normally, the I-bit is zero after a return from interrupt is executed. Although the I-bit can be cleared within an interrupt service routine, ‘nesting’ interrupts in this way should only be done when there is a clear understanding of latency and of the arbitration mechanism. Refer to Section 5.

3.1.6.6 Half carry (H)

The H-bit is set when a carry occurs between bits 3 and 4 of the arithmetic logic unit during an ADD, ABA, or ADC instruction. Otherwise, the H-bit is cleared. Half carry is used during BCD operations.

3.1.6.7 X interrupt mask (X)

The XIRQ mask (X) bit disables interrupts from the $\overline{\text{XIRQ}}$ pin. After any reset, X is set by default and must be cleared by a software instruction. When an $\overline{\text{XIRQ}}$ interrupt is recognized, the X- and I-bits are set after the registers are stacked, but before the interrupt vector is fetched. After the interrupt has been serviced, an RTI instruction is normally executed, causing the registers to be restored to the values that were present before the interrupt occurred. The X interrupt mask bit is set only by hardware $\overline{\text{RESET}}$ or $\overline{\text{XIRQ}}$ acknowledge). X is cleared only by program instruction (TAP, where the associated bit of A is 0; or RTI, where bit 6 of the value loaded into the CCR from the stack has been cleared). There is no hardware action for clearing X.

3.1.6.8 Stop disable (S)

Setting the STOP disable (S) bit prevents the STOP instruction from putting the M68HC11 into a low-power stop condition. If the STOP instruction is encountered by the CPU while the S-bit is set, it is treated as a no-operation (NOP) instruction, and processing continues to the next instruction. S is set by reset — STOP disabled by default.

3.2 Data types

The M68HC11 CPU supports the following data types:

- Bit data
- 8-bit and 16-bit signed and unsigned integers
- 16-bit unsigned fractions
- 16-bit addresses

A byte is eight bits wide and can be accessed at any byte location. A word is composed of two consecutive bytes with the most significant byte at the lower value address. Because the M68HC11 is an 8-bit CPU, there are no special requirements for alignment of instructions or operands.

3.3 Opcodes and operands

The M68HC11 family of microcontrollers uses 8-bit opcodes. Each opcode identifies a particular instruction and associated addressing mode to the CPU. Several opcodes are required to provide each instruction with a range of addressing capabilities. Only 256 opcodes would be available if the range of values were restricted to the number able to be expressed in 8-bit binary numbers.

A four-page opcode map has been implemented to expand the number of instructions. An additional byte, called a prebyte, directs the processor from page 0 of the opcode map to one of the other three pages. As its name implies, the additional byte precedes the opcode.

A complete instruction consists of a prebyte, if any, an opcode, and zero, one, two, or three operands. The operands contain information the CPU needs for executing the instruction. Complete instructions can be from one to five bytes long.

3.4 Addressing modes

Six addressing modes; immediate, direct, extended, indexed, inherent, and relative, detailed in the following paragraphs, can be used to access memory. All modes except inherent mode use an effective address. The effective address is the memory address from which the argument is fetched or stored, or the address from which execution is to proceed. The effective address can be specified within an instruction, or it can be calculated.

3.5 Immediate (IMM)

In the immediate addressing mode an argument is contained in the byte(s) immediately following the opcode. The number of bytes following the opcode matches the size of the register or memory location being operated on. There are two, three, and four (if prebyte is required) byte immediate instructions. The effective address is the address of the byte following the instruction.

3.5.1 Direct (DIR)

In the direct addressing mode, the low-order byte of the operand address is contained in a single byte following the opcode, and the high-order byte of the address is assumed to be $00. Addresses $00–$FF are thus accessed directly, using two-byte instructions. Execution time is reduced by eliminating the additional memory access required for the high-order address byte. In most applications, this 256-byte area is reserved for frequently referenced data. In M68HC11 MCUs, the memory map can be configured for combinations of internal registers, RAM, or external memory to occupy these addresses.

3.5.2 Extended (EXT)

In the extended addressing mode, the effective address of the argument is contained in two bytes following the opcode byte. These are three-byte instructions (or four-byte instructions if a prebyte is required). One or two bytes are needed for the opcode and two for the effective address.

3.5.3 Indexed (IND, X; IND, Y)

In the indexed addressing mode, an 8-bit unsigned offset contained in the instruction is added to the value contained in an index register (IX or IY) — the sum is the effective address. This addressing mode allows referencing any memory location in the 64Kbyte address space. These are two- to five-byte instructions, depending on whether or not a prebyte is required.

3.5.4 Inherent (INH)

In the inherent addressing mode, all the information necessary to execute the instruction is contained in the opcode. Operations that use only the index registers or accumulators, as well as control instructions with no arguments, are included in this addressing mode. These are one or two-byte instructions.

3.5.5 Relative (REL)

The relative addressing mode is used only for branch instructions. If the branch condition is true, an 8-bit signed offset included in the instruction is added to the contents of the program counter to form the effective branch address. Otherwise, control proceeds to the next instruction. These are usually two-byte instructions.

3.6 Instruction set

Refer to Table 3-2, which shows all the M68HC11 instructions in all possible addressing modes. For each instruction, the table shows the operand construction, the number of machine code bytes, and execution time in CPU E clock cycles.

Table 3-2 Instruction set (Page 1 of 6)

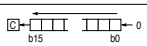
Mnemonic	Operation	Description	Addressing mode	Instruction			Condition codes						
				Opcode	Operand	Cycles	S	X	H	I	N	Z	V	C
ABA	Add accumulators	A + B ⇒ A	INH	1B	—	2	—	—	Δ	—	Δ	Δ	Δ	Δ
ABX	Add B to X	IX + (00:B) ⇒ IX	INH	3A	—	3	—	—	—	—	—	—	—	—
ABY	Add B to Y	IY + (00:B) ⇒ IY	INH	18 3A	—	4	—	—	—	—	—	—	—	—
ADCA (opr)	Add with carry to A	A + M + C ⇒ A	A IMM A DIR A EXT A IND, X A IND, Y	89 99 B9 A9 18 A9	ii dd hh ll ff ff	2 3 4 4 5	—	—	Δ	—	Δ	Δ	Δ	Δ
ADCB (opr)	Add with carry to B	B + M + C ⇒ B	B IMM B DIR B EXT B IND, X B IND, Y	C9 D9 F9 E9 18 E9	ii dd hh ll ff ff	2 3 4 4 5	—	—	Δ	—	Δ	Δ	Δ	Δ
ADDA (opr)	Add memory to A	A + M ⇒ A	A IMM A DIR A EXT A IND, X A IND, Y	8B 9B BB AB 18 AB	ii dd hh ll ff ff	2 3 4 4 5	—	—	Δ	—	Δ	Δ	Δ	Δ
ADDB (opr)	Add memory to B	B + M ⇒ B	B IMM B DIR B EXT B IND, X B IND, Y	CB DB FB EB 18 EB	ii dd hh ll ff ff	2 3 4 4 5	—	—	Δ	—	Δ	Δ	Δ	Δ
ADDD (opr)	Add 16-bit to D	D + (M:M+1) ⇒ D	IMM DIR EXT IND, X IND, Y	C3 D3 F3 E3 18 E3	jj kk dd hh ll ff ff	4 5 6 6 7	—	—	—	—	Δ	Δ	Δ	Δ
ANDA (opr)	AND A with memory	A • M ⇒ A	A IMM A DIR A EXT A IND, X A IND, Y	84 94 B4 A4 18 A4	ii dd hh ll ff ff	2 3 4 4 5	—	—	—	—	Δ	Δ	0	—
ANDB (opr)	AND B with memory	B • M ⇒ B	B IMM B DIR B EXT B IND, X B IND, Y	C4 D4 F4 E4 18 E4	ii dd hh ll ff ff	2 3 4 4 5	—	—	—	—	Δ	Δ	0	—
ASL (opr)	Arithmetic shift left		EXT IND, X IND, Y	78 68 18 68	hh ll ff ff	6 6 7	—	—	—	—	Δ	Δ	Δ	Δ
ASLA	Arithmetic shift left A		A INH	48	—	2	—	—	—	—	Δ	Δ	Δ	Δ
ASLB	Arithmetic shift left B		B INH	58	—	2	—	—	—	—	Δ	Δ	Δ	Δ
ASLD	Arithmetic shift left D		INH	05	—	3	—	—	—	—	Δ	Δ	Δ	Δ
ASR	Arithmetic shift right		EXT IND, X IND, Y	77 67 18 67	hh ll ff ff	6 6 7	—	—	—	—	Δ	Δ	Δ	Δ
ASRA	Arithmetic shift right A		A INH	47	—	2	—	—	—	—	Δ	Δ	Δ	Δ
ASRB	Arithmetic shift right B		B INH	57	—	2	—	—	—	—	Δ	Δ	Δ	Δ
BCC (rel)	Branch if carry clear	C = 0 ?	REL	24	rr	3	—	—	—	—	—	—	—	—
BCLR (opr) (msk)	Clear bit(s)	M • ($\overline{MM}$) ⇒ M	DIR IND, X IND, Y	15 1D 18 1D	dd mm ff mm ff mm	6 7 8	—	—	—	—	Δ	Δ	0	—
BCS (rel)	Branch if carry set	C = 1 ?	REL	25	rr	3	—	—	—	—	—	—	—	—
BEQ (rel)	Branch if equal to zero	Z = 1 ?	REL	27	rr	3	—	—	—	—	—	—	—	—
BGE (rel)	Branch if ≥ zero	N ⊕ V = 0 ?	REL	2C	rr	3	—	—	—	—	—	—	—	—
BGT (rel)	Branch if > zero	Z + (N ⊕ V) = 0 ?	REL	2E	rr	3	—	—	—	—	—	—	—	—
BHI (rel)	Branch if higher	C + Z = 0 ?	REL	22	rr	3	—	—	—	—	—	—	—	—

Table 3-2 Instruction set (Page 2 of 6)

Mnemonic	Operation	Description	Addressing mode	Instruction			Condition codes							
				Opcode	Operand	Cycles	S	X	H	I	N	Z	V	C
BHS (rel)	Branch if higher or same	C = 0 ?	REL	24	rr	3	–	–	–	–	–	–	–	–
BITA (opr)	Bit(s) test A with memory	A • M	A IMM A DIR A EXT A IND, X A IND, Y	85 95 B5 A5 18 A5	ii dd hh ll ff ff	2 3 4 4 5	–	–	–	–	Δ	Δ	0	–
BITB (opr)	Bit(s) test B with memory	B • M	B IMM B DIR B EXT B IND, X B IND, Y	C5 D5 F5 E5 18 E5	ii dd hh ll ff ff	2 3 4 4 5	–	–	–	–	Δ	Δ	0	–
BLE (rel)	Branch if ≤ zero	Z + (N ⊕ V) = 1 ?	REL	2F	rr	3	–	–	–	–	–	–	–	–
BLO (rel)	Branch if lower	C = 1 ?	REL	25	rr	3	–	–	–	–	–	–	–	–
BLS (rel)	Branch if lower or same	C + Z = 1 ?	REL	23	rr	3	–	–	–	–	–	–	–	–
BLT (rel)	Branch if < zero	N ⊕ V = 1 ?	REL	2D	rr	3	–	–	–	–	–	–	–	–
BMI (rel)	Branch if minus	N = 1 ?	REL	2B	rr	3	–	–	–	–	–	–	–	–
BNE (rel)	Branch if ≠ zero	Z = 0 ?	REL	26	rr	3	–	–	–	–	–	–	–	–
BPL(rel)	Branch if plus	N = 0 ?	REL	2A	rr	3	–	–	–	–	–	–	–	–
BRA (rel)	Branch always	1 = 1 ?	REL	20	rr	3	–	–	–	–	–	–	–	–
BRCLR(opr) (msk) (rel)	Branch if bit(s) clear	M • mm = 0 ?	DIR IND, X IND, Y	13 1F 18 1F	dd mm rr ff mm rr ff mm rr	6 7 8	–	–	–	–	–	–	–	–
BRN (rel)	Branch never	1 = 0 ?	REL	21	rr	3	–	–	–	–	–	–	–	–
BRSET(opr) (msk) (rel)	Branch if bit(s) set	$\overline{M}$ • mm = 0 ?	DIR IND, X IND, Y	12 1E 18 1E	dd mm rr ff mm rr ff mm rr	6 7 8	–	–	–	–	–	–	–	–
BSET (opr) (msk)	Set bit(s)	M + mm ⇒ M	DIR IND, X IND, Y	14 1C 18 1C	dd mm ff mm ff mm	6 7 8	–	–	–	–	Δ	Δ	0	–
BSR (rel)	Branch to subroutine	see Figure 3-2	REL	8D	rr	6	–	–	–	–	–	–	–	–
BVC (rel)	Branch if overflow clear	V = 0 ?	REL	28	rr	3	–	–	–	–	–	–	–	–
BVS (rel)	Branch if overflow set	V = 1 ?	REL	29	rr	3	–	–	–	–	–	–	–	–
CBA	Compare A with B	A – B	INH	11	–	2	–	–	–	–	Δ	Δ	Δ	Δ
CLC	Clear carry bit	0 ⇒ C	INH	0C	–	2	–	–	–	–	–	–	–	0
CLI	Clear interrupt mask	0 ⇒ I	INH	0E	–	2	–	–	–	0	–	–	–	–
CLR (opr)	Clear memory byte	0 ⇒ M	DIR IND, X IND, Y	7F 6F 18 6F	hh ll ff ff	6 6 7	–	–	–	–	0	1	0	0
CLRA	Clear accumulator A	0 ⇒ A	A INH	4F	–	2	–	–	–	–	0	1	0	0
CLRB	Clear accumulator B	0 ⇒ B	B INH	5F	–	2	–	–	–	–	0	1	0	0
CLV	Clear overflow flag	0 ⇒ V	INH	0A	–	2	–	–	–	–	–	–	0	–
CMPA (opr)	Compare A with memory	A – M	A IMM A DIR A EXT A IND, X A IND, Y	81 91 B1 A1 18 A1	ii dd hh ll ff ff	2 3 4 4 5	–	–	–	–	Δ	Δ	Δ	Δ
CMPB (opr)	Compare B with memory	B – M	B IMM B DIR B EXT B IND, X B IND, Y	C1 D1 F1 E1 18 E1	ii dd hh ll ff ff	2 3 4 4 5	–	–	–	–	Δ	Δ	Δ	Δ
COM (opr)	Ones complement memory byte	$FF – M ⇒ M	EXT IND, X IND, Y	73 63 18 63	hh ll ff ff	6 6 7	–	–	–	–	Δ	Δ	0	1
COMA	Ones complement A	$FF – A ⇒ A	A INH	43	–	2	–	–	–	–	Δ	Δ	0	1
COMB	Ones complement B	$FF – B ⇒ B	B INH	53	–	2	–	–	–	–	Δ	Δ	0	1

Table 3-2 Instruction set (Page 3 of 6)

Mnemonic	Operation	Description	Addressing mode	Instruction			Condition codes							
				Opcode	Operand	Cycles	S	X	H	I	N	Z	V	C
CPD (opr)	Compare D with memory (16-bit)	D – (M:M+1)	IMM DIR EXT IND, X IND, Y	1A 83 1A 93 1A B3 1A A3 CD A3	jj kk dd hh ll ff ff	5 6 7 7 7	–	–	–	–	Δ	Δ	Δ	Δ
CPX (opr)	Compare IX with memory (16-bit)	IX – (M:M+1)	IMM DIR EXT IND, X IND, Y	8C 9C BC AC CD AC	jj kk dd hh ll ff ff	4 5 6 6 7	–	–	–	–	Δ	Δ	Δ	Δ
CPY (opr)	Compare IY with memory (16-bit)	IY – (M:M+1)	IMM DIR EXT IND, X IND, Y	18 8C 18 9C 18 BC 1A AC 18 AC	jj kk dd hh ll ff ff	5 6 7 7 7	–	–	–	–	Δ	Δ	Δ	Δ
DAA	Decimal adjust A	adjust sum to BCD	INH	19	–	2	–	–	–	–	Δ	Δ	?	Δ
DEC (opr)	Decrement memory byte	M – 1 ⇒ M	EXT IND, X IND, Y	7A 6A 18 6A	hh ll ff ff	6 6 7	–	–	–	–	Δ	Δ	Δ	–
DECA	Decrement accumulator A	A – 1 ⇒ A	A INH	4A	–	2	–	–	–	–	Δ	Δ	Δ	–
DECB	Decrement accumulator B	B – 1 ⇒ B	B INH	5A	–	2	–	–	–	–	Δ	Δ	Δ	–
DES	Decrement stack pointer	SP – 1 ⇒ SP	INH	34	–	3	–	–	–	–	–	–	–	–
DEX	Decrement index register X	IX – 1 ⇒ IX	INH	09	–	3	–	–	–	–	–	Δ	–	–
DEY	Decrement index register Y	IY – 1 ⇒ IY	INH	18 09	–	4	–	–	–	–	–	Δ	–	–
EORA (opr)	Exclusive OR A with memory	A ⊕ M ⇒ A	A IMM A DIR A EXT A IND, X A IND, Y	88 98 B8 A8 18 A8	ii dd hh ll ff ff	2 3 4 4 5	–	–	–	–	Δ	Δ	0	–
EORB (opr)	Exclusive OR B with memory	B ⊕ M ⇒ A	B IMM B DIR B EXT B IND, X B IND, Y	C8 D8 F8 E8 18 E8	ii dd hh ll ff ff	2 3 4 4 5	–	–	–	–	Δ	Δ	0	–
FDIV	Fractional divide, 16 by 16	D / IX ⇒ IX; r ⇒ D	INH	03	–	41	–	–	–	–	–	Δ	Δ	Δ
IDIV	Integer divide, 16 by 16	D / IX ⇒ IX; r ⇒ D	INH	02	–	41	–	–	–	–	–	Δ	0	Δ
INC (opr)	Increment memory byte	M + 1 ⇒ M	EXT IND, X IND, Y	7C 6C 18 6C	hh ll ff ff	6 6 7	–	–	–	–	Δ	Δ	Δ	–
INCA	Increment accumulator A	A + 1 ⇒ A	A INH	4C	–	2	–	–	–	–	Δ	Δ	Δ	–
INCB	Increment accumulator B	B + 1 ⇒ B	B INH	5C	–	2	–	–	–	–	Δ	Δ	Δ	–
INS	Increment stack pointer	SP + 1 ⇒ SP	INH	31	–	3	–	–	–	–	–	–	–	–
INX	Increment index register X	IX + 1 ⇒ IX	INH	08	–	3	–	–	–	–	–	Δ	–	–
INY	Increment index register Y	IY + 1 ⇒ IY	INH	18 08	–	4	–	–	–	–	–	Δ	–	–
JMP (opr)	Jump	see Figure 3-2	EXT IND, X IND, Y	7E 6E 18 6E	hh ll ff ff	3 3 4	–	–	–	–	–	–	–	–
JSR (opr)	Jump to subroutine	see Figure 3-2	DIR EXT IND, X IND, Y	9D BD AD 18 AD	dd hh ll ff ff	5 6 6 7	–	–	–	–	–	–	–	–
LDAA (opr)	Load accumulator A	M ⇒ A	A IMM A DIR A EXT A IND, X A IND, Y	86 96 B6 A6 18 A6	ii dd hh ll ff ff	2 3 4 4 5	–	–	–	–	Δ	Δ	0	–

Table 3-2 Instruction set (Page 4 of 6)

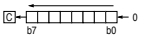
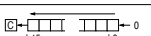
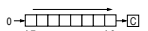
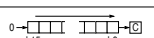
Mnemonic	Operation	Description	Addressing mode	Instruction			Condition codes							
				Opcode	Operand	Cycles	S	X	H	I	N	Z	V	C
LDAB (opr)	Load accumulator B	M ⇒ B	B IMM B DIR B EXT B IND, X B IND, Y	C6 D6 F6 E6 18 E6	ii dd hh ll ff ff	2 3 4 4 5	–	–	–	–	Δ	Δ	0	–
LDD (opr)	Load double accumulator D	M ⇒ A; M+1 ⇒ B	IMM DIR EXT IND, X IND, Y	CC DC FC EC 18 EC	jj kk dd hh ll ff ff	3 4 5 5 6	–	–	–	–	Δ	Δ	0	–
LDS (opr)	Load stack pointer	M:M+1 ⇒ SP	IMM DIR EXT IND, X IND, Y	8E 9E BE AE 18 AE	jj kk dd hh ll ff ff	3 4 5 5 6	–	–	–	–	Δ	Δ	0	–
LDX (opr)	Load index register X	M:M+1 ⇒ IX	IMM DIR EXT IND, X IND, Y	CE DE FE EE CD EE	jj kk dd hh ll ff ff	3 4 5 5 6	–	–	–	–	Δ	Δ	0	–
LDY (opr)	Load index register Y	M:M+1 ⇒ IY	IMM DIR EXT IND, X IND, Y	18 CE 18 DE 18 FE 1A EE 18 EE	jj kk dd hh ll ff ff	4 5 6 6 6	–	–	–	–	Δ	Δ	0	–
LSL (opr)	Logical shift left		EXT IND, X IND, Y	78 68 18 68	hh ll ff ff	6 6 7	–	–	–	–	Δ	Δ	Δ	Δ
LSLA	Logical shift left A		A INH	48	–	2	–	–	–	–	Δ	Δ	Δ	Δ
LSLB	Logical shift Left B		B INH	58	–	2	–	–	–	–	Δ	Δ	Δ	Δ
LSLD	Logical shift left D		INH	05	–	3	–	–	–	–	Δ	Δ	Δ	Δ
LSR (opr)	Logical shift right		EXT IND, X IND, Y	74 64 18 64	hh ll ff ff	6 6 7	–	–	–	–	0	Δ	Δ	Δ
LSRA	Logical shift right A		A INH	44	–	2	–	–	–	–	0	Δ	Δ	Δ
LSRB	Logical shift right B		B INH	54	–	2	–	–	–	–	0	Δ	Δ	Δ
LSRD	Logical shift right D		INH	04	–	3	–	–	–	–	0	Δ	Δ	Δ
MUL	Multiply, 8 x 8	A * B ⇒ D	INH	3D	–	10	–	–	–	–	–	–	–	Δ
NEG (opr)	Twos complement memory byte	0 – M ⇒ M	EXT IND, X IND, Y	70 60 18 60	hh ll ff ff	6 6 7	–	–	–	–	Δ	Δ	Δ	Δ
NEGA	Twos complement A	0 – A ⇒ A	A INH	40	–	2	–	–	–	–	Δ	Δ	Δ	Δ
NEGB	Twos complement B	0 – B ⇒ B	B INH	50	–	2	–	–	–	–	Δ	Δ	Δ	Δ
NOP	No operation	no operation	INH	01	–	2	–	–	–	–	–	–	–	–
ORAA	OR accumulator A (inclusive)	A + M ⇒ A	A IMM A DIR A EXT A IND, X A IND, Y	8A 9A BA AA 18 AA	ii dd hh ll ff ff	2 3 4 4 5	–	–	–	–	Δ	Δ	0	–
ORAB	OR accumulator B (inclusive)	B + M ⇒ B	B IMM B DIR B EXT B IND, X B IND, Y	CA DA FA EA 18 EA	ii dd hh ll ff ff	2 3 4 4 5	–	–	–	–	Δ	Δ	0	–
PSHA	Push A onto stack	A ⇒ Stack; SP = SP–1	A INH	36	–	3	–	–	–	–	–	–	–	–
PSHB	Push B onto stack	B ⇒ Stack; SP = SP–1	B INH	37	–	3	–	–	–	–	–	–	–	–
PSHX	Push IX onto stack (low first)	IX ⇒ Stack; SP = SP–2	INH	3C	–	4	–	–	–	–	–	–	–	–
PSHY	Push IY onto stack (low first)	IY ⇒ Stack; SP = SP–2	INH	18 3C	–	5	–	–	–	–	–	–	–	–

Table 3-2 Instruction set (Page 5 of 6)

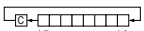
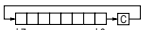
Mnemonic	Operation	Description	Addressing mode	Instruction			Condition codes							
				Opcode	Operand	Cycles	S	X	H	I	N	Z	V	C
PULA	Pull A from stack	SP = SP+1; Stack ⇒ A	A INH	32	–	4	–	–	–	–	–	–	–	–
PULB	Pull B from stack	SP = SP+1; Stack ⇒ B	B INH	33	–	4	–	–	–	–	–	–	–	–
PULX	Pull IX from stack (high first)	SP = SP+2; Stack ⇒ IX	INH	38	–	5	–	–	–	–	–	–	–	–
PULY	Pull IY from stack (high first)	SP = SP+2; Stack ⇒ IY	INH	18 38	–	6	–	–	–	–	–	–	–	–
ROL (opr)	Rotate left		EXT IND, X IND, Y	79 69 18 69	hh ll ff ff	6 6 7	–	–	–	–	Δ	Δ	Δ	Δ
ROLA	Rotate left A		A INH	49	–	2	–	–	–	–	Δ	Δ	Δ	Δ
ROLB	Rotate left B		B INH	59	–	2	–	–	–	–	Δ	Δ	Δ	Δ
ROR (opr)	Rotate right		EXT IND, X IND, Y	76 66 18 66	hh ll ff ff	6 6 7	–	–	–	–	Δ	Δ	Δ	Δ
RORA	Rotate right A		A INH	46	–	2	–	–	–	–	Δ	Δ	Δ	Δ
RORB	Rotate right B		B INH	56	–	2	–	–	–	–	Δ	Δ	Δ	Δ
RTI	Return from interrupt	see Figure 3-2	INH	3B	–	12	Δ	↓	Δ	Δ	Δ	Δ	Δ	Δ
RTS	Return from subroutine	see Figure 3-2	INH	39	–	5	–	–	–	–	–	–	–	–
SBA	Subtract B from A	A – B ⇒ A	INH	10	–	2	–	–	–	–	Δ	Δ	Δ	Δ
SBCA (opr)	Subtract with carry from A	A – M – C ⇒ A	A IMM A DIR A EXT A IND, X A IND, Y	82 92 B2 A2 18 A2	ii dd hh ll ff ff	2 3 4 4 5	–	–	–	–	Δ	Δ	Δ	Δ
SBCB (opr)	Subtract with carry from B	B – M – C ⇒ B	B IMM B DIR B EXT B IND, X B IND, Y	C2 D2 F2 E2 18 E2	ii dd hh ll ff ff	2 3 4 4 5	–	–	–	–	Δ	Δ	Δ	Δ
SEC	Set carry	1 ⇒ C	INH	0D	–	2	–	–	–	–	–	–	–	1
SEI	Set interrupt mask	1 ⇒ I	INH	0F	–	2	–	–	–	1	–	–	–	–
SEV	Set overflow flag	1 ⇒ V	INH	0B	–	2	–	–	–	–	–	–	1	–
STAA (opr)	Store accumulator A	A ⇒ M	A DIR A EXT A IND, X A IND, Y	97 B7 A7 18 A7	dd hh ll ff ff	3 4 4 5	–	–	–	–	Δ	Δ	0	–
STAB (opr)	Store accumulator B	B ⇒ M	B DIR B EXT B IND, X B IND, Y	D7 F7 E7 18 E7	dd hh ll ff ff	3 4 4 5	–	–	–	–	Δ	Δ	0	–
STD (opr)	Store accumulator D	A ⇒ M; B ⇒ M+1	DIR EXT IND, X IND, Y	DD FD ED 18 ED	dd hh ll ff ff	4 5 5 6	–	–	–	–	Δ	Δ	0	–
STOP	Stop internal clocks	–	INH	CF	–	2	–	–	–	–	–	–	–	–
STS (opr)	Store stack pointer	SP ⇒ M:M+1	DIR EXT IND, X IND, Y	9F BF AF 18 AF	dd hh ll ff ff	4 5 5 6	–	–	–	–	Δ	Δ	0	–
STX (opr)	Store index register X	IX ⇒ M:M+1	DIR EXT IND, X IND, Y	DF FF EF CD EF	dd hh ll ff ff	4 5 5 6	–	–	–	–	Δ	Δ	0	–
STY (opr)	Store index register Y	IY ⇒ M:M+1	DIR EXT IND, X IND, Y	18 DF 18 FF 1A EF 18 EF	dd hh ll ff ff	5 6 6 6	–	–	–	–	Δ	Δ	0	–

Table 3-2 Instruction set (Page 6 of 6)

Mnemonic	Operation	Description	Addressing mode	Instruction			Condition codes							
				Opcode	Operand	Cycles	S	X	H	I	N	Z	V	C
SUBA (opr)	Subtract memory from A	A – M ⇒ A	A IMM A DIR A EXT A IND, X A IND, Y	80 90 B0 A0 18 A0	ii dd hh ll ff ff	2 3 4 4 5	–	–	–	–	Δ	Δ	Δ	Δ
SUBB (opr)	Subtract memory from B	B – M ⇒ B	B IMM B DIR B EXT B IND, X B IND, Y	C0 D0 F0 E0 18 E0	ii dd hh ll ff ff	2 3 4 4 5	–	–	–	–	Δ	Δ	Δ	Δ
SUBD (opr)	Subtract memory from D	D – M:M+1 ⇒ D	IMM DIR EXT IND, X IND, Y	83 93 B3 A3 18 A3	jj kk dd hh ll ff ff	4 5 6 6 7	–	–	–	–	Δ	Δ	Δ	Δ
SWI	Software interrupt	see Figure 3-2	INH	3F	–	14	–	–	–	1	–	–	–	–
TAB	Transfer A to B	A ⇒ B	INH	16	–	2	–	–	–	–	Δ	Δ	0	–
TAP	Transfer A to CC register	A ⇒ CCR	INH	06	–	2	Δ	↓	Δ	Δ	Δ	Δ	Δ	Δ
TBA	Transfer B to A	B ⇒ A	INH	17	–	2	–	–	–	–	Δ	Δ	0	–
TEST	Test (only in test modes)	address bus increments	INH	00	–	†	–	–	–	–	–	–	–	–
TPA	Transfer CC register to A	CCR ⇒ A	INH	07	–	2	–	–	–	–	–	–	–	–
TST (opr)	Test for zero or minus	M – 0	EXT IND, X IND, Y	7D 6D 18 6D	hh ll ff ff	6 6 7	–	–	–	–	Δ	Δ	0	0
TSTA	Test A for zero or minus	A – 0	A INH	4D	–	2	–	–	–	–	Δ	Δ	0	0
TSTB	Test B for zero or minus	B – 0	B INH	5D	–	2	–	–	–	–	Δ	Δ	0	0
TSX	Transfer stack pointer to X	SP + 1 ⇒ IX	INH	30	–	3	–	–	–	–	–	–	–	–
TSY	Transfer stack pointer to Y	SP + 1 ⇒ IY	INH	18 30	–	4	–	–	–	–	–	–	–	–
TXS	Transfer X to stack pointer	IX – 1 ⇒ SP	INH	35	–	3	–	–	–	–	–	–	–	–
TYS	Transfer Y to stack pointer	IY – 1 ⇒ SP	INH	18 35	–	4	–	–	–	–	–	–	–	–
WAI	Wait for interrupt	stack registers & WAIT	INH	3E	–	‡	–	–	–	–	–	–	–	–
XGDX	Exchange D with X	IX ⇒ D; D ⇒ IX	INH	8F	–	3	–	–	–	–	–	–	–	–
XGDY	Exchange D with Y	IY ⇒ D; D ⇒ IY	INH	18 8F	–	4	–	–	–	–	–	–	–	–

Operators

- ⇒ Is transferred to
- • Boolean AND
- + Arithmetic addition, except where used as an inclusive-OR symbol in Boolean formulae
- ⊕ Exclusive-OR
- * Multiply
- : Concatenation
- – Arithmetic subtraction, or negation symbol (Twos complement)

Operands

- dd 8-bit direct address ($0000–$00FF); the high byte is assumed to be zero
- ff 8-bit positive offset ($00 to $FF (0 to 256)) is added to the contents of the index register
- hh High order byte of 16-bit extended address
- ii One byte of immediate data
- jj High order byte of 16-bit immediate data
- kk Low order byte of 16-bit immediate data
- ll Low order byte of 16-bit extended address
- mm 8-bit mask (set bits to be affected)
- rr Signed relative offset ($80 to $7F (–128 to +127)); offset is relative to the address following the offset byte

Cycles

- † Infinite, or until reset occurs
- ‡ 12 cycles are used, beginning with the opcode fetch. A wait state is entered, which remains in effect for an integer number of MPU E clock cycles (n) until an interrupt is recognised. Finally, two additional cycles are used to fetch the appropriate interrupt vector. (14 + n, total).

Condition Codes

- – Bit not changed
- 0 Bit always cleared
- 1 Bit always set
- Δ Bit set or cleared, depending on the operation
- ↓ Bit can be cleared, but cannot become set
- ? Not defined

4
OPERATING MODES AND ON-CHIP MEMORY

This section contains information about the modes that define MC68HC11KW1 operating conditions, and about the on-chip memory that allows the MCU to be configured for various applications.

4.1 Operating modes

The values of the mode select inputs MODB and MODA during reset determine the operating mode (See Table 4-4). Single chip and expanded modes are the normal modes. In single chip mode only on-board memory is available. Expanded mode, however, allows access to external memory. Each of these two normal modes is paired with a special mode. Bootstrap, a variation of the single chip mode, is a special mode that executes a bootloader program in an internal bootstrap ROM. Test is a special expanded mode that allows privileged access to internal resources.

4.1.1 Single chip operating mode

In single chip operating mode, the MC68HC11KW1 microcontroller has no external address or data bus. Ports B, C and F are available for general-purpose parallel I/O.

4.1.2 Expanded operating mode

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

The MC68HC11KW1 includes an additional memory expansion feature, available in expanded modes, which allows access to pages of memory in one or two windows within the 64K byte physical memory space. This can extend the memory space to more than 1M byte. See Section 4.4.

For information about the four programmable chip selects available in expanded modes, refer to Section 4.5. A security feature can protect EEPROM data when in expanded mode (see Section 4.6.3).

4.1.3 Special test mode

Special test, a variation of the expanded mode, is used during Motorola's internal production testing, and is not intended or recommended for any other purpose. Its specification is subject to change without notice.

4.1.4 Special bootstrap mode

When the MCU is reset in special bootstrap mode, a small on-chip ROM is enabled at address $BE40–$BFFF. The ROM contains a reset vector and a bootloader program. The MCU fetches the reset vector, then executes the bootloader.

For normal use of the bootloader program, send a synchronization byte $FF to the SCI receiver at either E clock ÷256, or E clock ÷1664 (15624 or 2400 baud respectively, for an E clock of 4 MHz). Then download up to 768 bytes of program data (which is put into RAM starting at $00A0). These characters are echoed through the transmitter. The bootloader program ends the download after a timeout of four character times or 768 bytes. When loading is complete, the program jumps to location $00A0 and begins executing the code. Use of an external pull-up resistor is required when using the SCI transmitter pin (TXD) because port D pins are configured for wired-OR operation by the bootloader. In bootstrap mode, the interrupt vectors point to RAM. This allows the use of interrupts through a jump table.

Further baud rate options are available on the MC68HC11KW1 by using a different value for the synchronization byte, as shown in Table 4-1.

Refer also to Motorola application note ***AN1060, M68HC11 Bootstrap Mode*** (the bootloader mode is similar to that used on the MC68HC11K4).

Table 4-1 Example bootloader baud rates

Sync. byte	Timeout delay	Baud rates for an E clock of 4.00 MHz
$FF	4 char.	15624
$FF	4	2400
$F0	4.9	19200
$FD	17.3	10416
$FD	13	7812

4.2 On-chip memory

The MC68HC11KW1 MCU includes 768 bytes of on-chip RAM and 640 bytes of EEPROM. The bootloader ROM occupies a 448 byte block of the memory map. The CONFIG register is implemented as a separate EEPROM byte.

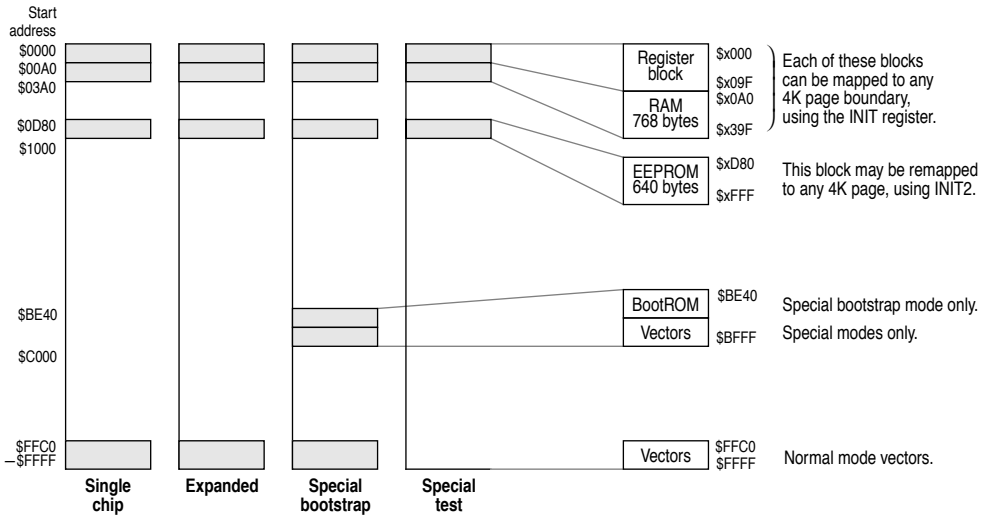


Figure 4-1 MC68HC11KW1 memory map

4.2.1 Mapping allocations

Memory locations for on-chip resources are the same for both expanded and single chip modes. The 160-byte register block originates at $0000 on reset and can be placed at any other 4K boundary ($x000) after reset by writing an appropriate value to the INIT register. Refer to Figure 4-1, which shows the memory map.

The on-board 768 byte block of RAM is initially located at $00A0 after reset. The RAM is divided into two sections of 160 bytes and 608 bytes. If RAM and registers are both mapped to the same 4K boundary, RAM starts at $x0A0 and 160 bytes are remapped at $x300–$x39F. Otherwise, RAM starts at $x000. See Figure 4-2.

Remapping is accomplished by writing appropriate values into the two nibbles of the INIT register. See Section 4.3.2.2.

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

In special bootstrap mode, a bootloader ROM is enabled at locations $BE40–$BFFF. The vectors for special bootstrap mode are contained in the bootloader program.

4.2.1.1 RAM

The MC68HC11KW1 has 768 bytes of fully static RAM that are used for storing instructions, variables and temporary data during program execution. RAM can be placed at any 4K boundary in the 64K byte address space by writing an appropriate value to the INIT register.

By default, RAM is initially located at $00A0 in the memory map. Direct addressing mode can access the first 96 locations of RAM using a one-byte address operand. Direct mode accesses save program memory space and execution time. Registers can be moved to other boundaries to allow 256 bytes of RAM to be located in direct addressing space. See Figure 4-2.

The on-chip RAM is a fully static memory. RAM contents can be preserved during periods of processor inactivity by either of two methods, both of which reduce power consumption:

1) During the software-based STOP mode, MCU clocks are stopped, but the MCU continues to draw power from V_{DD}. Power supply current is directly related to operating frequency in CMOS integrated circuits and there is very little leakage when the clocks are stopped. These two factors reduce power consumption while the MCU is in STOP mode.
2) To reduce power consumption to a minimum, V_{DD} can be turned off, and the MODB/VSTBY pin can be used to supply RAM power from either a battery back-up or a second power supply. Although this method requires external hardware, it is very effective. Refer to Section 2 for information about how to connect the stand-by RAM power supply and to Section 5 for a description of low power operation.

4.2.1.2 Bootloader ROM

The bootloader ROM is enabled at address $BE40–$BFFF during special bootstrap mode. The reset vector is fetched from this ROM and the MCU executes the bootloader firmware. In normal modes, the bootloader ROM is disabled.

4.2.2 Registers

In Table 4-2, a summary of registers and control bits, the registers are shown in ascending order within the 160-byte register block. The addresses shown are for default block mapping ($0000–$009F), however the INIT register remaps the block to any 4K page ($x000–$x09F). See Section 4.3.2.2.

Table 4-2 Register and control bit assignments (Page 1 of 5)

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	PA7	PA6	PA5	PA4	PA3	PA2	PA1	PA0	undefined
Data direction A (DDRA)	$0001	DDA7	DDA6	DDA5	DDA4	DDA3	DDA2	DDA1	DDA0	0000 0000
Data direction B (DDRB)	$0002	DDB7	DDB6	DDB5	DDB4	DDB3	DDB2	DDB1	DDB0	0000 0000
Data direction F (DDRF)	$0003	DDF7	DDF6	DDF5	DDF4	DDF3	DDF2	DDF1	DDF0	0000 0000
Port B data (PORTB)	$0004	PB7	PB6	PB5	PB4	PB3	PB2	PB1	PB0	undefined
Port F data (PORTF)	$0005	PF7	PF6	PF5	PF4	PF3	PF2	PF1	PF0	undefined
Port C data (PORTC)	$0006	PC7	PC6	PC5	PC4	PC3	PC2	PC1	PC0	undefined
Data direction C (DDRC)	$0007	DDC7	DDC6	DDC5	DDC4	DDC3	DDC2	DDC1	DDC0	0000 0000
Port D data (PORTD)	$0008	PD7	PD6	PD5	PD4	PD3	PD2	PD1	PD0	undefined
Data direction D (DDRD)	$0009	DDD7	DDD6	DDD5	DDD4	DDD3	DDD2	DDD1	DDD0	0000 0000
Port E data (PORTE)	$000A	PE7	PE6	PE5	PE4	PE3	PE2	PE1	PE0	undefined
Timer compare force (CFORC)	$000B	FOC1	FOC2	FOC3	FOC4	FOC5	0	0	0	0000 0000
Output compare 1 mask (OC1M)	$000C	OC1M7	OC1M6	OC1M5	OC1M4	OC1M3	0	0	0	0000 0000
Output compare 1 data (OC1D)	$000D	OC1D7	OC1D6	OC1D5	OC1D4	OC1D3	0	0	0	0000 0000
Timer count (TCNT) high	$000E	(bit 15)	(14)	(13)	(12)	(11)	(10)	(9)	(bit 8)	0000 0000
Timer count (TCNT) low	$000F	(bit 7)	(6)	(5)	(4)	(3)	(2)	(1)	(bit 0)	0000 0000
Timer input capture 1 (TIC1) high	$0010	(bit 15)	(14)	(13)	(12)	(11)	(10)	(9)	(bit 8)	undefined
Timer input capture 1 (TIC1) low	$0011	(bit 7)	(6)	(5)	(4)	(3)	(2)	(1)	(bit 0)	undefined
Timer input capture 2 (TIC2) high	$0012	(bit 15)	(14)	(13)	(12)	(11)	(10)	(9)	(bit 8)	undefined
Timer input capture 2 (TIC2) low	$0013	(bit 7)	(6)	(5)	(4)	(3)	(2)	(1)	(bit 0)	undefined
Timer input capture 3 (TIC3) high	$0014	(bit 15)	(14)	(13)	(12)	(11)	(10)	(9)	(bit 8)	undefined
Timer input capture 3 (TIC3) low	$0015	(bit 7)	(6)	(5)	(4)	(3)	(2)	(1)	(bit 0)	undefined
Timer output compare 1 (TOC1) high	$0016	(bit 15)	(14)	(13)	(12)	(11)	(10)	(9)	(bit 8)	1111 1111
Timer output compare 1 (TOC1) low	$0017	(bit 7)	(6)	(5)	(4)	(3)	(2)	(1)	(bit 0)	1111 1111
Timer output compare 2 (TOC2) high	$0018	(bit 15)	(14)	(13)	(12)	(11)	(10)	(9)	(bit 8)	1111 1111
Timer output compare 2 (TOC2) low	$0019	(bit 7)	(6)	(5)	(4)	(3)	(2)	(1)	(bit 0)	1111 1111
Timer output compare 3 (TOC3) high	$001A	(bit 15)	(14)	(13)	(12)	(11)	(10)	(9)	(bit 8)	1111 1111
Timer output compare 3 (TOC3) low	$001B	(bit 7)	(6)	(5)	(4)	(3)	(2)	(1)	(bit 0)	1111 1111
Timer output compare 4 (TOC4) high	$001C	(bit 15)	(14)	(13)	(12)	(11)	(10)	(9)	(bit 8)	1111 1111
Timer output compare 4 (TOC4) low	$001D	(bit 7)	(6)	(5)	(4)	(3)	(2)	(1)	(bit 0)	1111 1111
Capture 4/compare 5 (TI4/O5) high	$001E	(bit 15)	(14)	(13)	(12)	(11)	(10)	(9)	(bit 8)	1111 1111
Capture 4/compare 5 (TI4/O5) low	$001F	(bit 7)	(6)	(5)	(4)	(3)	(2)	(1)	(bit 0)	1111 1111
Timer control 1 (TCTL1)	$0020	OM2	OL2	OM3	OL3	OM4	OL4	OM5	OL5	0000 0000
Timer control 2 (TCTL2)	$0021	EDG4B	EDG4A	EDG1B	EDG1A	EDG2B	EDG2A	EDG3B	EDG3A	0000 0000
Timer interrupt mask 1 (TMSK1)	$0022	OC1I	OC2I	OC3I	OC4I	I4/O5I	IC1I	IC2I	IC3I	0000 0000

Table 4-2 Register and control bit assignments (Page 2 of 5)

Register name	Address	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	State on reset
Timer interrupt flag 1 (TFLG1)	$0023	OC1F	OC2F	OC3F	OC4F	I4/O5F	IC1F	IC2F	IC3F	0000 0000
Timer interrupt mask 2 (TMSK2)	$0024	TOI	RTII	PAOVI	PAII	0	0	PR1	PR0	0000 0000
Timer interrupt flag 2 (TFLG2)	$0025	TOF	RTIF	PAOVF	PAIF	0	0	0	0	0000 0000
Pulse accumulator control (PACTL)	$0026	0	PAEN	PAMOD	PEDGE	0	I4/O5	RTR1	RTR0	0000 0000
Pulse accumulator count (PACNT)	$0027	(bit 7)	(6)	(5)	(4)	(3)	(2)	(1)	(bit 0)	undefined
SPI control (SPCR)	$0028	SPIE	SPE	DWOM	MSTR	CPOL	CPHA	SPR1	SPR0	0000 01uu
SPI status (SPSR)	$0029	SPIF	WCOL	0	MODF	0	0	0	0	0000 0000
SPI data (SPDR)	$002A	(bit 7)	(6)	(5)	(4)	(3)	(2)	(1)	(bit 0)	undefined
Reserved	$002B									
Port pull-up assignment (PPAR)	$002C	0	0	0	0	HPPUE	GPPUE	FPPUE	BPPUE	0000 1111
Port G assignment (PGAR)	$002D	0	0	PGAR5	PGAR4	PGAR3	PGAR2	PGAR1	PGAR0	0000 0000
Reserved	$002E									
Reserved	$002F									
A/D control & status (ADCTL)	$0030	CCF	CONV8	SCAN	MULT	CD	CC	CB	CA	0000 0000
Compare force for timers 2 and 3 (F23FRC)	$0031	FT3C1	FT3C2	FT3C3	FT3C4	FT2C1	FT2C2	FT2C3	FT2C4	0000 0000
A/D frequency select (ADFRQ)	$0032	0	0	0	0	0	0	0	ADER	0000 0000
Reserved	$0033									
Reserved	$0034									
Block protect (BPROT)	$0035	BULKP	BIT6	BPRT4	PTCON	BPRT3	BPRT2	BPRT1	BPRT0	1111 1111
Reserved	$0036									
EEPROM mapping (INIT2)	$0037	EE3	EE2	EE1	EE0	0	0	0	0	0000 0000
System config. options 2 (OPT2)	$0038	LIRDV	CWOM	0	IRVNE	LSBF	SPR2	XDV1	XDV0	000x 0000
System config. options 1 (OPTION)	$0039	ADPU	CSEL	IRQE	DLY	CME	FCME	CR1	CR0	0001 0000
COP timer arm/reset (COPRST)	$003A	(bit 7)	(6)	(5)	(4)	(3)	(2)	(1)	(bit 0)	undefined
EEPROM programming (PPROG)	$003B	ODD	EVEN	0	BYTE	ROW	ERASE	EELAT	EEPGM	0000 0000
Highest priority interrupt (HPRIO)	$003C	RBOOT	SMOD	MDA	PSEL4	PSEL3	PSEL2	PSEL1	PSEL0	xxx0 0110
RAM & I/O mapping (INIT)	$003D	RAM3	RAM2	RAM1	RAM0	REG3	REG2	REG1	REG0	0000 0000
Factory test (TEST1)	$003E	TILOP	0	OCCR	CBYP	DISR	FCM	FCOP	0	0000 x000
Configuration control (CONFIG)	$003F	1	1	CLKX	PAREN	NOSEC	NOCOP	1	EEON	11xx xx1x
A/D result 1 (ADR1) high	$0040	(Bit 15)	(14)	(13)	(12)	(11)	(10)	(9)	(8)	undefined
A/D result 1 (ADR1) low	$0041	(7)	(6)	0	0	0	0	0	0	uu00 0000
A/D result 2 (ADR2) high	$0042	(Bit 15)	(14)	(13)	(12)	(11)	(10)	(9)	(8)	undefined
A/D result 2 (ADR2) low	$0043	(7)	(6)	0	0	0	0	0	0	uu00 0000
A/D result 3 (ADR3) high	$0044	(Bit 15)	(14)	(13)	(12)	(11)	(10)	(9)	(8)	undefined
A/D result 3 (ADR3) low	$0045	(7)	(6)	0	0	0	0	0	0	uu00 0000

Table 4-2 Register and control bit assignments (Page 3 of 5)

Register name	Address	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	State on reset
A/D result 4 (ADR4) high	$0046	(Bit 15)	(14)	(13)	(12)	(11)	(10)	(9)	(8)	undefined
A/D result 4 (ADR4) low	$0047	(7)	(6)	0	0	0	0	0	0	uu00 0000
A/D result 5 (ADR5) high	$0048	(Bit 15)	(14)	(13)	(12)	(11)	(10)	(9)	(8)	undefined
A/D result 5 (ADR5) low	$0049	(7)	(6)	0	0	0	0	0	0	uu00 0000
A/D result 6 (ADR6) high	$004A	(Bit 15)	(14)	(13)	(12)	(11)	(10)	(9)	(8)	undefined
A/D result 6 (ADR6) low	$004B	(7)	(6)	0	0	0	0	0	0	uu00 0000
A/D result 7 (ADR7) high	$004C	(Bit 15)	(14)	(13)	(12)	(11)	(10)	(9)	(8)	undefined
A/D result 7 (ADR7) low	$004D	(7)	(6)	0	0	0	0	0	0	uu00 0000
A/D result 8 (ADR8) high	$004E	(Bit 15)	(14)	(13)	(12)	(11)	(10)	(9)	(8)	undefined
A/D result 8 (ADR8) low	$004F	(7)	(6)	0	0	0	0	0	0	uu00 0000
Reserved	$0050									
Reserved	$0051									
Reserved	$0052									
Reserved	$0053									
Reserved	$0054									
Reserved	$0055									
Memory mapping window size (MMSIZ)	$0056	MXGS2	MXGS1	W2SZ1	W2SZ0	0	0	W1SZ1	W1SZ0	0000 0000
Memory mapping window base (MMWBR)	$0057	W2A15	W2A14	W2A13	0	W1A15	W1A14	W1A13	0	0000 0000
Memory mapping window 1 control (MM1CR)	$0058	0	X1A18	X1A17	X1A16	X1A15	X1A14	X1A13	0	0000 0000
Memory mapping window 2 control (MM2CR)	$0059	0	X2A18	X2A17	X2A16	X2A15	X2A14	X2A13	0	0000 0000
Chip select clock stretch (CSCSTR)	$005A	IOSA	IOSB	GP1SA	GP1SB	GP2SA	GP2SB	PCSA	PCSB	0000 000x
Chip select control (CSCTL)	$005B	IOEN	IOPL	IOCSA	IOSZ	GCSPR	PCSEN	PCSZA	PCSZB	0000 0100
Gen. purpose chip select 1 addr. (GPCS1A)	$005C	G1A18	G1A17	G1A16	G1A15	G1A14	G1A13	G1A12	G1A11	0000 0000
Gen. purpose chip select 1 con. (GPCS1C)	$005D	G1DG2	G1DPC	G1POL	G1AV	G1SZA	G1SZB	G1SZC	G1SZD	0000 0000
Gen. purpose chip select 2 addr. (GPCS2A)	$005E	G2A18	G2A17	G2A16	G2A15	G2A14	G2A13	G2A12	G2A11	0000 0000
Gen. purpose chip select 2 con. (GPCS2C)	$005F	0	G2DPC	G2POL	G2AV	G2SZA	G2SZB	G2SZC	G2SZD	0000 0000
Pulse width clock select (PWCLK)	$0060	CON34	CON12	PCKA2	PCKA1	0	PCKB3	PCKB2	PCKB1	0000 0000
Pulse width polarity select (PWPOL)	$0061	PCLK4	PCLK3	PCLK2	PCLK1	PPOL4	PPOL3	PPOL2	PPOL1	0000 0000
Pulse width scale (PWSCAL)	$0062	(bit 7)	(6)	(5)	(4)	(3)	(2)	(1)	(bit 0)	0000 0000
Pulse width enable (PWEN)	$0063	TPWSL	DISCP	0	0	PWEN4	PWEN3	PWEN2	PWEN1	0000 0000
Pulse width count 1 (PWCNT1)	$0064	(bit 7)	(6)	(5)	(4)	(3)	(2)	(1)	(bit 0)	0000 0000
Pulse width count 2 (PWCNT2)	$0065	(bit 7)	(6)	(5)	(4)	(3)	(2)	(1)	(bit 0)	0000 0000
Pulse width count 3 (PWCNT3)	$0066	(bit 7)	(6)	(5)	(4)	(3)	(2)	(1)	(bit 0)	0000 0000
Pulse width count 4 (PWCNT4)	$0067	(bit 7)	(6)	(5)	(4)	(3)	(2)	(1)	(bit 0)	0000 0000
Pulse width period 1 (PWPER1)	$0068	(bit 7)	(6)	(5)	(4)	(3)	(2)	(1)	(bit 0)	1111 1111

Table 4-2 Register and control bit assignments (Page 4 of 5)

Register name	Address	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	State on reset
Pulse width period 2 (PWPER2)	$0069	(bit 7)	(6)	(5)	(4)	(3)	(2)	(1)	(bit 0)	1111 1111
Pulse width period 3 (PWPER3)	$006A	(bit 7)	(6)	(5)	(4)	(3)	(2)	(1)	(bit 0)	1111 1111
Pulse width period 4 (PWPER4)	$006B	(bit 7)	(6)	(5)	(4)	(3)	(2)	(1)	(bit 0)	1111 1111
Pulse width duty 1 (PWDTY1)	$006C	(bit 7)	(6)	(5)	(4)	(3)	(2)	(1)	(bit 0)	1111 1111
Pulse width duty 2 (PWDTY2)	$006D	(bit 7)	(6)	(5)	(4)	(3)	(2)	(1)	(bit 0)	1111 1111
Pulse width duty 3 (PWDTY3)	$006E	(bit 7)	(6)	(5)	(4)	(3)	(2)	(1)	(bit 0)	1111 1111
Pulse width duty 4 (PWDTY4)	$006F	(bit 7)	(6)	(5)	(4)	(3)	(2)	(1)	(bit 0)	1111 1111
SCI baud rate high (SCBDH)	$0070	BTST	BSPL	SYNC	SBR12	SBR11	SBR10	SBR9	SBR8	0000 0000
SCI baud rate low (SCBDL)	$0071	SBR7	SBR6	SBR5	SBR4	SBR3	SBR2	SBR1	SBR0	0000 0100
SCI control 1 (SCCR1)	$0072	LOOPS	WOMS	0	M	WAKE	ILT	PE	PT	0000 0000
SCI control 2 (SCCR2)	$0073	TIE	TCIE	RIE	ILIE	TE	RE	RWU	SBK	0000 0000
SCI status 1 (SCSR1)	$0074	TDRE	TC	RDRF	IDLE	OR	NF	FE	PF	1100 0000
SCI status 2 (SCSR2)	$0075	0	0	0	0	0	0	0	RAF	0000 0000
SCI data high (SCDRH)	$0076	R8	T8	0	0	0	0	0	0	undefined
SCI data low (SCDRL)	$0077	R7T7	R6T6	R5T5	R4T4	R3T3	R2T2	R1T1	R0T0	undefined
Reserved	$0078									
Reserved	$0079									
Reserved	$007A									
Reserved	$007B									
Port H data (PORTH)	$007C	PH7	PH6	PH5	PH4	PH3	PH2	PH1	PH0	undefined
Data direction H (DDRH)	$007D	DDH7	DDH6	DDH5	DDH4	DDH3	DDH2	DDH1	DDH0	0000 0000
Port G data (PORTG)	$007E	PG7	PG6	PG5	PG4	PG3	PG2	PG1	PG0	undefined
Data direction G (DDRG)	$007F	0	0	DDG5	DDG4	DDG3	DDG2	DDG1	DDG0	0000 0000
Timer control register 3 (TCTL3)	$0080	OM1	OL1	OM2	OL2	OM3	OL3	OM4	OL4	0000 0000
Timer control register 4 (TCTL4)	$0081	EDGB	EDGA	PR2B	PR2A	ECEB	ECEA	T2STP	I1/04	0000 0000
Timer 2 counter register (TCNT2) high	$0082	(bit 15)	(14)	(13)	(12)	(11)	(10)	(9)	(bit 8)	0000 0000
Timer 2 counter register (TCNT2) low	$0083	(bit 7)	(6)	(5)	(4)	(3)	(2)	(1)	(bit 0)	0000 0000
Timer 2 output compare 1(T2OC1) high	$0084	(bit 15)	(14)	(13)	(12)	(11)	(10)	(9)	(bit 8)	1111 1111
Timer 2 output compare 1 (T2OC1) low	$0085	(bit 7)	(6)	(5)	(4)	(3)	(2)	(1)	(bit 0)	1111 1111
Timer 2 output comp. 2 (T2OC2) high	$0086	(bit 15)	(14)	(13)	(12)	(11)	(10)	(9)	(bit 8)	1111 1111
Timer 2 output comp. 2 (T2OC2) low	$0087	(bit 7)	(6)	(5)	(4)	(3)	(2)	(1)	(bit 0)	1111 1111
Timer 2 output comp. 3 (T20C3) high	$0088	(bit 15)	(14)	(13)	(12)	(11)	(10)	(9)	(bit 8)	1111 1111
Timer 2 output comp. 3(T20C3) low	$0089	(bit 7)	(6)	(5)	(4)	(3)	(2)	(1)	(bit 0)	1111 1111
Timer 2 channel 4 (T2C4) high	$008A	(bit 15)	(14)	(13)	(12)	(11)	(10)	(9)	(bit 8)	1111 1111
Timer 2 channel 4 (T2C4) low	$008B	(bit 7)	(6)	(5)	(4)	(3)	(2)	(1)	(bit 0)	1111 1111

Table 4-2 Register and control bit assignments (Page 5 of 5)

Register name	Address	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	State on reset
Timer 2 mask (T2MSK)	$008C	OC1I	OC2I	OC3I	C4I	TO2I	0	0	0	0000 0000
Timer 2 flag (T2FLG)	$008D	OC1F	OC2F	OC3F	C4F	TO2F	0	0	0	0000 0000
Port J data (PORTJ)	$008E	PJ7	PJ6	PJ5	PJ4	PJ3	PJ2	PJ1	PJ0	undefined
Data direction J (DDRJ)	$008F	DDJ7	DDJ6	DDJ5	DDJ4	DDJ3	DDJ2	DDJ1	DDJ0	0000 0000
Timer control register 5 (TCTL5)	$0090	OM1	OL1	OM2	OL2	OM3	OL3	OM4	OL4	0000 0000
Timer control register 6 (TCTL6)	$0091	EDGB	EDGA	PR3B	PR3A	ECEB	ECEA	T3STP	I1/O4	0000 0000
Timer 3 counter (TCNT3) high	$0092	(bit 15)	(14)	(13)	(12)	(11)	(10)	(9)	(bit 8)	0000 0000
Timer 3 counter (TCNT3) low	$0093	(bit 7)	(6)	(5)	(4)	(3)	(2)	(1)	(bit 0)	0000 0000
Timer 3 output compare 1 (T30C1) high	$0094	(bit 15)	(14)	(13)	(12)	(11)	(10)	(9)	(bit 8)	1111 1111
Timer 3 output compare 1 (T30C1) low	$0095	(bit 7)	(6)	(5)	(4)	(3)	(2)	(1)	(bit 0)	1111 1111
Timer 3 output compare 2 (T30C2) high	$0096	(bit 15)	(14)	(13)	(12)	(11)	(10)	(9)	(bit 8)	1111 1111
Timer 3 output compare 2 (T30C2) low	$0097	(bit 7)	(6)	(5)	(4)	(3)	(2)	(1)	(bit 0)	1111 1111
Timer 3 output comp. 3 (T3OC3) high	$0098	(bit 15)	(14)	(13)	(12)	(11)	(10)	(9)	(bit 8)	1111 1111
Timer 3 output comp. 3 (T3OC3) low	$0099	(bit 7)	(6)	(5)	(4)	(3)	(2)	(1)	(bit 0)	1111 1111
Timer 3 channel 4 (T3C4) high	$009A	(bit 15)	(14)	(13)	(12)	(11)	(10)	(9)	(bit 8)	1111 1111
Timer 3 channel 4 (T3C4) low	$009B	(bit 7)	(6)	(5)	(4)	(3)	(2)	(1)	(bit 0)	1111 1111
Timer 3 mask (T3MSK)	$009C	OC1I	OC2I	OC3I	C4I	TO3I	0	0	0	0000 0000
Timer 3 flag (T3FLG)	$009D	OC1F	OC2F	OC3F	C4F	TO3F	0	0	0	0000 0000
Port K data (PORTK)	$009E	PK7	PK6	PK5	PK4	PK3	PK2	PK1	PK0	undefined
Data direction K (DDRK)	$009F	DDK7	DDK6	DDK5	DDK4	DDK3	DDK2	DDK1	DDK0	0000 0000

KEY

x State on reset depends on mode selected

u State of bit on reset is undefined

4.3 System initialization

Registers and bits that control initialization and the basic operation of the MCU are protected against writes except under special circumstances. The following table lists registers that can be written only once after reset, or that must be written within the first 64 cycles after reset.

Table 4-3 Registers with limited write access

Register address	Register name	Must be written in first 64 cycles	Write once only
$x024	Timer interrupt mask register 2 (TMSK2)	(1)	—
$x035	Block protect register (BPROT)	(2)	—
$x037	EEPROM mapping register (INIT2)	No	Yes
$x038	System configuration options register 2 (OPT2)	No	(3)
$x039	System configuration options register (OPTION)	(4)	—
$x03D	RAM and I/O map register (INIT)	(5)	—
$x081	Timer control register 4 (Timer 2) TCTL4	No	(6)
$x091	Timer control register 6 (Timer 3) TCTL6	No	(6)

(1) When SMOD = 0, bits 1 and 0 can be written only once, during the first 64 cycles, after which they become read-only. When SMOD = 1, however, these bits can be written at any time. All other bits can be written at any time.

(2) Bits can be written to zero once and only in the first 64 cycles or in special modes. Bits can be set to one at any time.

(3) Bit 4 (IRVNE) can be written only once.

(4) When SMOD = 0, bits 5, 4, 2, 1, and 0 can be written once and only in the first 64 cycles. When SMOD = 1, however, bits 5, 4, 2, 1, and 0 can be written at any time. All other bits can be written at any time.

(5) When SMOD = 0, bits can be written only once, during the first 64 cycles, after which the register becomes read-only. When SMOD = 1, bits can be written at any time.

(6) Bits 5, 4, 3 and 2 can be written only once.

4.3.1 Mode selection

The four mode variations are selected by the logic states of the mode A (MODA) and mode B (MODB) pins during reset. The MODA and MODB logic levels determine the logic state of the special mode (SMOD) and mode A (MDA) control bits in the highest priority I-bit interrupt and miscellaneous (HPRIO) register.

After reset is released, the mode select pins no longer influence the MCU operating mode. In single chip operating mode, MODA pin is connected to a logic zero. In expanded mode, MODA is normally connected to V_{DD} through a pull-up resistor of 4.7 kΩ. The MODA pin also functions as the load instruction register ($\overline{LIR}$) pin when the MCU is not in reset. The open-drain active low $\overline{LIR}$ output pin drives low during the first E cycle of each instruction. The MODB pin also functions as the stand-by power input (VSTBY), which allows the RAM contents to be maintained in the absence of V_{DD}.

8.5.2 SPSR — SPI status register

	Address	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	State on reset
SPI status (SPSR)	$0029	SPIF	WCOL	0	MODF	0	0	0	0	0000 0000

This register can be read at any time, but writing to it has no effect.

SPIF — SPI interrupt complete flag

1 (set) – Data transfer to external device has been completed.

0 (clear) – No valid completion of data transfer.

SPIF is set upon completion of data transfer between the processor and the external device. If SPIF goes high, and if SPIE is set, a serial peripheral interrupt is generated. To clear the SPIF bit, read the SPSR with SPIF set, then access the SPDR. Unless SPSR is read (with SPIF set) first, attempts to write SPDR are inhibited.

WCOL — Write collision

1 (set) – Write collision.

0 (clear) – No write collision.

Clearing the WCOL bit is accomplished by reading the SPSR (with WCOL set) followed by an access of SPDR. Refer to Section 8.3.4 and Section 8.4.

MODF — Mode fault

1 (set) – Mode fault.

0 (clear) – No mode fault.

To clear the MODF bit, read the SPSR (with MODF set), then write to the SPCR. Refer to Section 8.3.4 and Section 8.4.

Bits [5, 3:0] — Not implemented; always read zero.

8.5.3 SPDR — SPI data register

	Address	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	State on reset
SPI data (SPDR)	$002A	(bit 7)	(6)	(5)	(4)	(3)	(2)	(1)	(bit 0)	undefined

The SPDR is used when transmitting or receiving data on the serial bus. Only a write to this register initiates transmission or reception of a byte, and this only occurs in the master device. At the completion of transferring a byte of data, the SPIF status bit is set in both the master and slave devices.

A read of the SPDR is actually a read of a buffer. To prevent an overrun and the loss of the byte that caused the overrun, the first SPIF must be cleared by the time a second transfer of data from the shift register to the read buffer is initiated.

SPI is double buffered in and single buffered out.

8.5.4 OPT2 — System configuration options register 2

	Address	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	State on reset
System config. options 2 (OPT2)	$0038	LIRDV	CWOM	0	IRVNE	LSBF	SPR2	XDV1	XDV0	000x 0000

LIRDV — $\overline{\text{LIR}}$ driven (refer to Section 4)

1 (set) – Enable $\overline{\text{LIR}}$ drive high pulse.

0 (clear) – $\overline{\text{LIR}}$ not driven high on MODA/$\overline{\text{LIR}}$ pin.

CWOM — Port C wired-OR mode (refer to Section 6)

1 (set) – Port C outputs are open-drain.

0 (clear) – Port C operates normally.

Bit 5 — Not implemented; always reads zero.

IRVNE — Internal read visibility/not E (refer to Section 4)

1 (set) – Data from internal reads is driven out of the external data bus.

0 (clear) – No visibility of internal reads on external bus.

In **single chip mode** this bit determines whether the E clock drives out from the chip.

1 (set) – E pin is driven low.

0 (clear) – E clock is driven out from the chip.

LSBF — LSB first enable

1 (set) – SPI data is transferred LSB first.

0 (clear) – SPI data is transferred MSB first.

If this bit is set, data, which is usually transferred MSB first, is transferred LSB first. LSBF does not affect the position of the MSB and LSB in the data register. Reads and writes of the data register always have MSB in bit 7.

SPR2 — SPI clock rate select

When set, SPR2 adds a divide-by-4 prescaler to the SPI clock chain. With the two bits in the SPCR, this bit specifies the SPI clock rate. Refer to Table 8-1.

XDV[1, 0] — XOUT clock divide select (refer to Section 4)

These two bits control the frequency of the XCLK signal, which is output on the XOUT pin if enabled by the CLKX bit in CONFIG.

9
TIMING SYSTEM

The MC68HC11KW1 contains three 16-bit timers. Figure 9-1 provides a diagram of the entire timing system. The main timer, Timer 1, is described in the following paragraphs; refer to Section 9.2 and Section 9.3 for descriptions of Timer 2 and Timer 3.

9.1 Timer 1

Timer 1 is the standard M68HC11 timing system, composed of several clock divider chains. The main clock divider chain includes a 16-bit free-running counter, driven by a programmable prescaler.

The prescaler output divides the system clock by 1, 4, 8, or 16. Taps from this main clocking chain drive-circuitry generate the slower clocks used by the pulse accumulator, the real-time interrupt (RTI), and the computer operating properly (COP) watchdog subsystems, which are also described in this section. Refer to Figure 9-1.

All main timer system activities are referenced to this free-running counter. The counter begins incrementing from $0000 as the MCU comes out of reset, and continues to the maximum count, $FFFF. At the maximum count, the counter rolls over to $0000, sets an overflow flag and continues to increment. As long as the MCU is running in a normal operating mode, there is no way to reset, change or interrupt the counting. The capture/compare subsystem features three input capture channels, four output compare channels and one channel that can be selected to perform either input capture or output compare. Each of the three input capture functions has its own 16-bit input capture register (time capture latch) and each of the output compare functions has its own 16-bit compare register. All timer functions, including the timer overflow and RTI, have their own interrupt controls and separate interrupt vectors.

The pulse accumulator contains an 8-bit counter and edge select logic. The pulse accumulator can operate in either event counting mode or gated time accumulation mode. During event counting mode, the pulse accumulator's 8-bit counter increments when a specified edge is detected on an input signal. During gated time accumulation mode, an internal clock source increments the 8-bit counter while an input signal has a predetermined logic level.

The real-time interrupt (RTI) is a programmable periodic interrupt circuit that permits pacing the execution of software routines by selecting one of four interrupt rates.

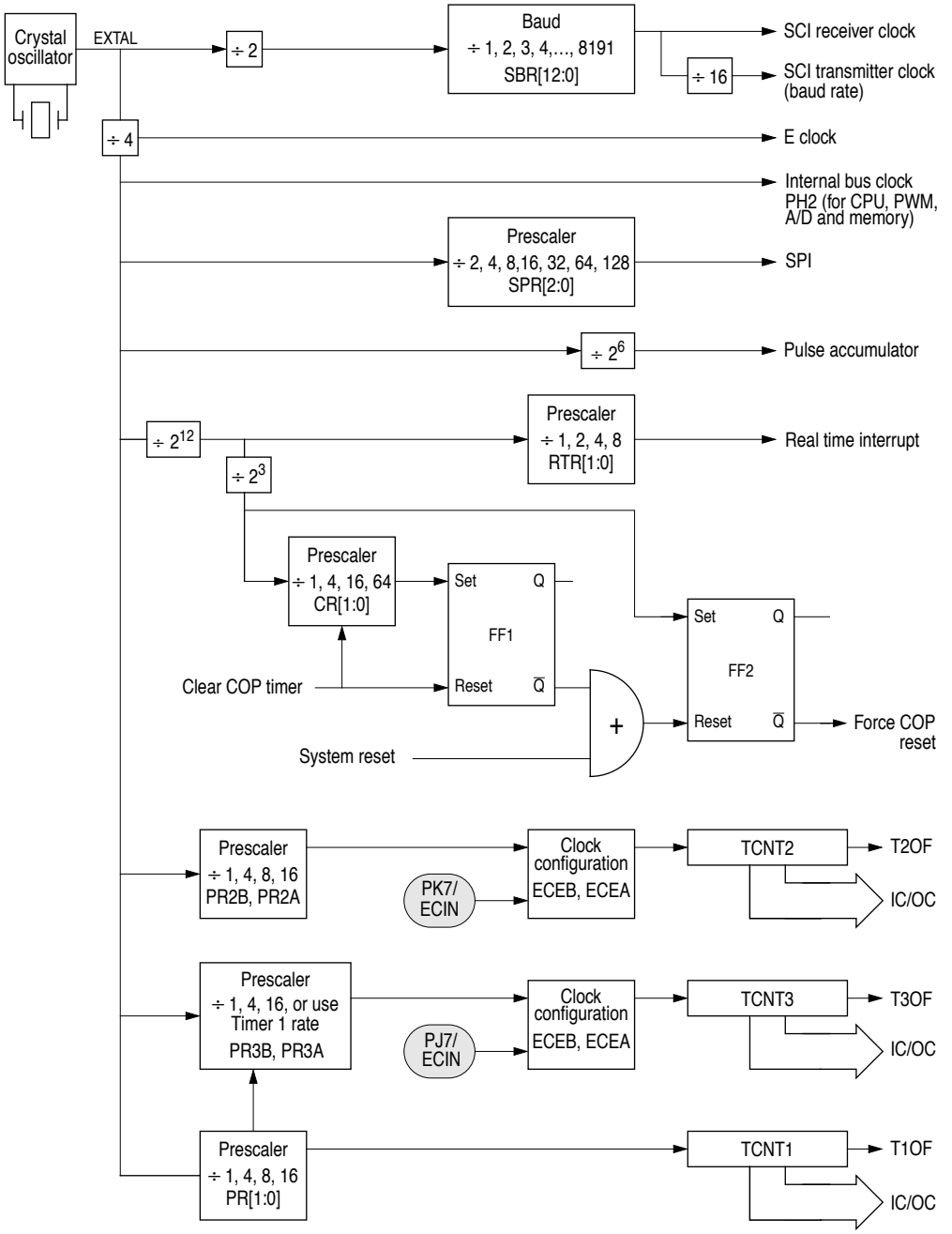


Figure 9-1 Timer clock divider chains

The COP watchdog clock input ($E/2^{15}$) is tapped off from the free-running counter chain. The COP automatically times out unless it is serviced within a specific time by a program reset sequence. If the COP is allowed to time out, a reset is generated, which drives the $\overline{\text{RESET}}$ pin low to reset the MCU and the external system. Refer to Table 9-1 for crystal related frequencies and periods.

Table 9-1 Timer 1 resolution and capacity

Control bits PR[1:0]	Clock: 16.0 MHz / 4.0 MHz / 250 ns	Clock: 4 E / E / 1/E	XTAL / E clock / Period
0 0	250 ns 16.384 ms	1/E 2^{16}/E	– resolution – overflow
0 1	1.0 µs 65.536 ms	4/E 2^{18}/E	– resolution – overflow
1 0	2.0 µs 131.07 ms	8/E 2^{19}/E	– resolution – overflow
1 1	4.0 µs 262.14 ms	16/E 2^{20}/E	– resolution – overflow

9.1.1 Timer 1 structure

The functions of Timer 1 share I/O with the pins of port A as follows:

Pin	Alternative function
PA0	IC3
PA1	IC2
PA2	IC1
PA3	OC5 and/or OC1, or IC4
PA4	OC4 and/or OC1
PA5	OC3 and/or OC1
PA6	OC2 and/or OC1
PA7	PAI and/or OC1

Figure 9-3 shows the Timer 1 capture/compare system block diagram. The port A pin control block includes logic for timer functions and for general-purpose I/O. For pins PA3, PA2, PA1 and PA0, this block contains both the edge-detection logic and the control logic that enables the selection of which edge triggers an input capture. The digital level on PA[3:0] can be read at any time (read PORTA register), even if the pin is being used for the input capture function. Pins PA[6:3] are used either for general-purpose I/O, or as output compare pins. When one of these pins is being used for an output compare function, it cannot be written directly as if it were a general-purpose output. Each of the output compare functions (OC[5:2]) is related to one of the port A output pins. Output

compare 1 (OC1) has extra control logic, allowing it optional control of any combination of the PA[7:3] pins. The PA7 pin can be used as a general-purpose I/O pin, as an input to the pulse accumulator or as an OC1 output pin.

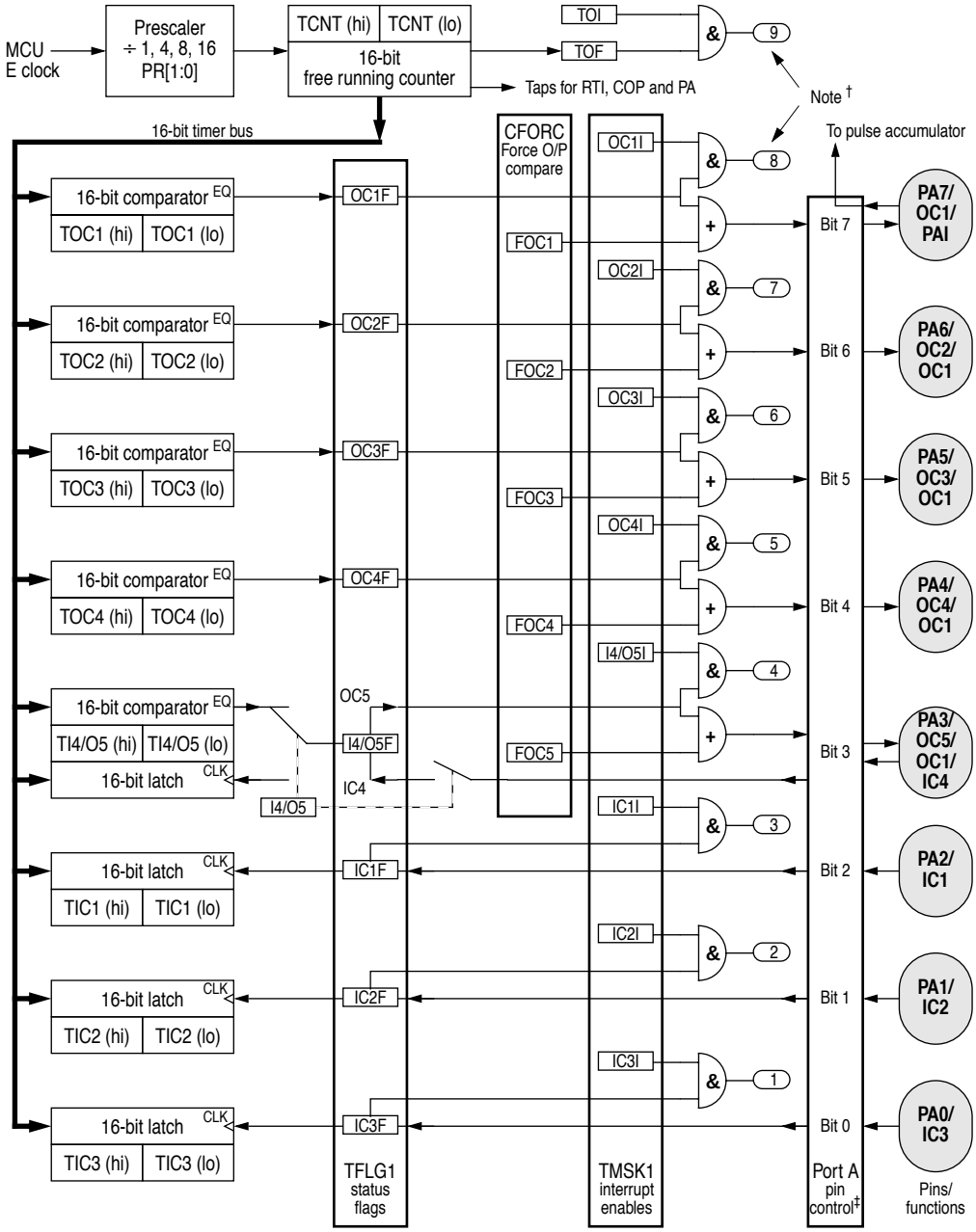
9.1.2 Input capture

The input capture function records the time an external event occurs by latching the value of the free-running counter when a selected edge is detected at the associated timer input pin. Software can store latched values and use them to compute the periodicity and duration of events. For example, by storing the times of successive edges of an incoming signal, software can determine the period and pulse width of a signal. To measure period, two successive edges of the same polarity are captured. To measure pulse width, two alternate polarity edges are captured.

In most cases, input capture edges are asynchronous with respect to the internal timer counter, which is clocked relative to an internal clock (PH2). These asynchronous capture requests are synchronized with PH2 so that latching occurs on the opposite half cycle of PH2 from when the timer counter is being incremented. This synchronization process introduces a delay from when the edge occurs to when the counter value is detected. Because these delays cancel out when the time between two edges is being measured, the delay can be ignored. When an input capture is being used with an output compare, there is a similar delay between the actual compare point and when the output pin changes state.

The control and status bits that implement the input capture functions are contained in the PACTL, TCTL2, TMSK1, and TFLG1 registers.

To configure port A bit 3 as an input capture, clear the DDA3 bit of the DDRA register. Note that this bit is cleared out of reset. To enable PA3 as the fourth input capture, set the I4/O5 bit in the PACTL register. Otherwise, PA3 is configured as a fifth output compare out of reset, with bit I4/O5 being cleared. If the DDA3 bit is set (configuring PA3 as an output), and IC4 is enabled, then writes to PA3 cause edges on the pin to result in input captures. Writing to TI4/O5 has no effect when the TI4/O5 register is acting as IC4.



† Interrupt requests 1–9 (these are further qualified by the I-bit in the CCR)

‡ Port A pin actions are controlled by OC1M, OC1D, PACTL, TCTL1 and TCTL2 registers

Figure 9-2 Timer 1 capture/compare block diagram

9.1.2.1 TCTL2 — Timer control register 2

	Address	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	State on reset
Timer control 2 (TCTL2)	$0021	EDG4B	EDG4A	EDG1B	EDG1A	EDG2B	EDG2A	EDG3B	EDG3A	0000 0000

Use the control bits of this register to program input capture functions to detect a particular edge polarity on the corresponding timer input pin. Each of the input capture functions can be independently configured to detect rising edges only, falling edges only, any edge (rising or falling), or to disable the input capture function. The input capture functions operate independently of each other and can capture the same TCNT value if the input edges are detected within the same timer count cycle.

EDGxB and EDGxA — Input capture edge control

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

There are four pairs of these bits. Each pair is cleared by reset and must be encoded to configure the corresponding input capture edge detector circuit. IC4 functions only if the I4/O5 bit in the PACTL register is set.

9.1.2.2 TIC1–TIC3 — Timer input capture registers

	Address	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	State on reset
Timer input capture 1 (TIC1) high	$0010	(bit 15)	(14)	(13)	(12)	(11)	(10)	(9)	(bit 8)	not affected
Timer input capture 1 (TIC1) low	$0011	(bit 7)	(6)	(5)	(4)	(3)	(2)	(1)	(bit 0)	not affected
Timer input capture 2 (TIC2) high	$0012	(bit 15)	(14)	(13)	(12)	(11)	(10)	(9)	(bit 8)	not affected
Timer input capture 2 (TIC2) low	$0013	(bit 7)	(6)	(5)	(4)	(3)	(2)	(1)	(bit 0)	not affected
Timer input capture 3 (TIC3) high	$0014	(bit 15)	(14)	(13)	(12)	(11)	(10)	(9)	(bit 8)	not affected
Timer input capture 3 (TIC3) low	$0015	(bit 7)	(6)	(5)	(4)	(3)	(2)	(1)	(bit 0)	not affected

When an edge has been detected and synchronized, the 16-bit free-running counter value is transferred into the input capture register pair as a single 16-bit parallel transfer. Timer counter value captures and timer counter incrementing occur on opposite half-cycles of the phase 2 clock so that the count value is stable whenever a capture occurs. Input capture values can be read from a pair of 8-bit read-only registers. A read of the high-order byte of an input capture register pair inhibits a new capture transfer for one bus cycle. If a double-byte read instruction, such as LDD, is used to read the captured value, coherency is assured. When a new input capture occurs immediately after a high-order byte read, transfer is delayed for an additional cycle but the value is not lost.

The TICx registers are not affected by reset.

9.1.2.3 TI4/O5 — Timer input capture 4/output compare 5 register

	Address	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	State on reset
Capture 4/compare 5 (TI4/O5) high	$001E	(bit 15)	(14)	(13)	(12)	(11)	(10)	(9)	(bit 8)	1111 1111
Capture 4/compare 5 (TI4/O5) low	$001F	(bit 7)	(6)	(5)	(4)	(3)	(2)	(1)	(bit 0)	1111 1111

Use TI4/O5 as either an input capture register or an output compare register, depending on the function chosen for the PA3 pin. To enable it as an input capture pin, set the I4/O5 bit in the pulse accumulator control register (PACTL) to logic level one. To use it as an output compare register, set the I4/O5 bit to a logic level zero. Refer to Section 9.6.1.

The TI4/O5 register pair resets to ones ($FFFF).

9.1.3 Output compare

Use the output compare (OC) function to program an action to occur at a specific time — when the 16-bit counter reaches a specified value. For each of the five output compare functions, there is a separate 16-bit compare register and a dedicated 16-bit comparator. The value in the compare register is compared to the value of the free-running counter on every bus cycle. When the compare register matches the counter value, an output compare status flag is set. The flag can be used to initiate the automatic actions for that output compare function.

To produce a pulse of a specific duration, write a value to the output compare register that represents the time the leading edge of the pulse is to occur. The output compare circuit is configured to set the appropriate output either high or low, depending on the polarity of the pulse being produced. After a match occurs, the output compare register is reprogrammed to change the output pin back to its inactive level at the next match. A value representing the width of the pulse is added to the original value, and then written to the output compare register. Because the pin state changes occur at specific values of the free-running counter, the pulse width can be controlled accurately at the resolution of the free-running counter, independent of software latency. To generate an output signal of a specific frequency and duty cycle, repeat this pulse-generating procedure.

There are four 16-bit read/write output compare registers: TOC1, TOC2, TOC3, and TOC4, and the TI4/O5 register, which functions under software control as either IC4 or OC5. Each of the OC registers is set to $FFFF on reset. A value written to an OC register is compared to the free-running counter value during each E clock cycle. If a match is found, the particular output compare flag is set in timer interrupt flag register 1 (TFLG1). If that particular interrupt is enabled in the timer interrupt mask register 1 (TMSK1), an interrupt is generated. In addition to an interrupt, a specified action can be initiated at one or more timer output pins. For OC[5:2], the pin action is controlled by pairs of bits (OMx and OLx) in the TCTL1 register. The output action is taken on each successful compare, regardless of whether or not the OCxF flag in the TFLG1 register was previously cleared.

OC1 is different from the other output compares in that a successful OC1 compare can affect any or all five of the OC pins. The OC1 output action taken when a match is found is controlled by two 8-bit registers with three bits unimplemented: the output compare 1 mask register, OC1M, and the output compare 1 data register, OC1D. OC1M specifies which port A outputs are to be used, and OC1D specifies what data is placed on these port pins.

9.1.3.1 TOC1–TOC4 — Timer output compare registers

	Address	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	State on reset
Timer output compare 1 (TOC1) high	$0016	(bit 15)	(14)	(13)	(12)	(11)	(10)	(9)	(bit 8)	1111 1111
Timer output compare 1 (TOC1) low	$0017	(bit 7)	(6)	(5)	(4)	(3)	(2)	(1)	(bit 0)	1111 1111
Timer output compare 2 (TOC2) high	$0018	(bit 15)	(14)	(13)	(12)	(11)	(10)	(9)	(bit 8)	1111 1111
Timer output compare 2 (TOC2) low	$0019	(bit 7)	(6)	(5)	(4)	(3)	(2)	(1)	(bit 0)	1111 1111
Timer output compare 3 (TOC3) high	$001A	(bit 15)	(14)	(13)	(12)	(11)	(10)	(9)	(bit 8)	1111 1111
Timer output compare 3 (TOC3) low	$001B	(bit 7)	(6)	(5)	(4)	(3)	(2)	(1)	(bit 0)	1111 1111
Timer output compare 4 (TOC4) high	$001C	(bit 15)	(14)	(13)	(12)	(11)	(10)	(9)	(bit 8)	1111 1111
Timer output compare 4 (TOC4) low	$001D	(bit 7)	(6)	(5)	(4)	(3)	(2)	(1)	(bit 0)	1111 1111

All output compare registers are 16-bit read-write. Each is initialized to $FFFF at reset. If an output compare register is not used for an output compare function, it can be used as a storage location. A write to the high-order byte of an output compare register pair inhibits the output compare function for one bus cycle. This inhibition prevents inappropriate subsequent comparisons. Coherency requires a complete 16-bit read or write. However, if coherency is not needed, byte accesses can be used.

For output compare functions, write a comparison value to output compare registers TOC1–TOC4 and TI4/O5. When TCNT value matches the comparison value, specified pin actions occur.

9.1.3.2 CFORC — Timer compare force register

	Address	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	State on reset
Timer compare force (CFORC)	$000B	FOC1	FOC2	FOC3	FOC4	FOC5	0	0	0	0000 0000

The CFORC register allows forced early compares. FOC[1:5] correspond to the five output compares. These bits are set for each output compare that is to be forced. The action taken as a result of a forced compare is the same as if there were a match between the OCx register and the free-running counter, except that the corresponding interrupt status flag bits are not set. The forced channels trigger their programmed pin actions to occur at the next timer count transition after the write to CFORC.

The CFORC bits should not normally be used on an output compare function that is programmed to toggle its output on a successful compare, because a normal compare occurring immediately before or after the force would produce a double toggle. This may be undesirable if it happens quickly, since the resulting output pulse would be very short.

FOC[1:5] — Force output compares

1 (set) – A forced output compare action will occur on the specified pin.

0 (clear) – No action.

Bits [2:0] — Not implemented; always read zero

9.1.3.3 OC1M — Output compare 1 mask register

	Address	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	State on reset
Output compare 1 mask (OC1M)	$000C	OC1M7	OC1M6	OC1M5	OC1M4	OC1M3	0	0	0	0000 0000

Use OC1M with OC1 to specify the bits of port A that are affected by a successful OC1 compare. The bits of the OC1M register correspond to PA7–PA3.

OC1M[7:3] — Output compare masks for OC1

1 (set) – OC1 is configured to control the corresponding pin of port A.

0 (clear) – OC1 will not affect the corresponding port A pin.

Bits [2:0] — Not implemented; always read zero.

9.1.3.4 OC1D — Output compare 1 data register

	Address	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	State on reset
Output compare 1 data (OC1D)	$000D	OC1D7	OC1D6	OC1D5	OC1D4	OC1D3	0	0	0	0000 0000

Use this register with OC1 to specify the data that is to be written to the affected pin of port A after a successful OC1 compare. When a successful OC1 compare occurs, a data bit in OC1D is written to the corresponding pin of port A for each bit that is set in OC1M.

OC1D[7:3] — Output compare data for OC1

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

Bits [2:0] — Not implemented; always read zero

9.1.3.5 TCNT — Timer counter register

	Address	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	State on reset
Timer count (TCNT) high	$000E	(bit 15)	(14)	(13)	(12)	(11)	(10)	(9)	(bit 8)	0000 0000
Timer count (TCNT) low	$000F	(bit 7)	(6)	(5)	(4)	(3)	(2)	(1)	(bit 0)	0000 0000

The 16-bit read-only TCNT register contains the prescaled value of the 16-bit timer. A full counter read addresses the more significant byte (MSB) first. A read of this address causes the less significant byte (LSB) to be latched into a buffer for the next CPU cycle so that a double-byte read returns the full 16-bit state of the counter at the time of the MSB read cycle.

TCNT resets to $0000.

9.1.3.6 TCTL1 — Timer control register 1

	Address	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	State on reset
Timer control 1 (TCTL1)	$0020	OM2	OL2	OM3	OL3	OM4	OL4	OM5	OL5	0000 0000

The bits of this register specify the action taken as a result of a successful OCx compare.

OM[2:5] — Output mode
OL[2:5] — Output level

OMx	OLx	Action taken on successful compare
0	0	Timer disconnected from OCx pin logic
0	1	Toggle OCx output line
1	0	Clear OCx output line to 0
1	1	Set OCx output line to 1

These control bit pairs are encoded to specify the action taken after a successful OCx compare. OC5 functions only if the I4/O5 bit in the PACTL register is clear.

9.1.3.7 TMSK1 — Timer interrupt mask register 1

	Address	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	State on reset
Timer interrupt mask 1 (TMSK1)	$0022	OC1I	OC2I	OC3I	OC4I	I4/O5I	IC1I	IC2I	IC3I	0000 0000

Use this 8-bit register to enable or inhibit the timer input capture and output compare interrupts.

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

OC1I–OC4I — Output compare x interrupt enable

1 (set) – OCx interrupt is enabled.

0 (clear) – OCx interrupt is disabled.

If the OCxI enable bit is set when the OCxF flag bit is set, a hardware interrupt sequence is requested.

I4/O5I — Input capture 4/output compare 5 interrupt enable

1 (set) – IC4/OC5 interrupt is enabled.

0 (clear) – IC4/OC5 interrupt is disabled.

When I4/O5 in PACTL is set, I4/O5I is the input capture 4 interrupt enable bit.
When I4/O5 in PACTL is zero, I4/O5I is the output compare 5 interrupt enable bit.

IC1I–IC3I — Input capture x interrupt enable

1 (set) – ICx interrupt is enabled.

0 (clear) – ICx interrupt is disabled.

If the ICxI enable bit is set when the ICxF flag bit is set, a hardware interrupt sequence is requested.

9.1.3.8 TFLG1 — Timer interrupt flag register 1

	Address	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	State on reset
Timer interrupt flag 1 (TFLG1)	$0023	OC1F	OC2F	OC3F	OC4F	I4/O5F	IC1F	IC2F	IC3F	0000 0000

Bits in this register indicate when timer system events have occurred. Coupled with the bits of TMSK1, the bits of TFLG1 allow the timer subsystem to operate in either a polled or interrupt driven system. Clear flags by writing a one to the corresponding bit position(s).

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

OC1F–OC4F — Output compare x flag

1 (set) – Counter has reached the preset output compare x value.

0 (clear) – Counter has not reached the preset output compare x value.

These flags are set each time the counter matches the corresponding output compare x values.

I4/O5F — Input capture 4/output compare 5 flag

Set by IC4 or OC5, depending on the function enabled by I4/O5 bit in PACTL

IC1F–IC3F — Input capture x flag

1 (set) – Selected edge has been detected on corresponding port pin.

0 (clear) – Selected edge has not been detected on corresponding port pin.

These flags are set each time a selected active edge is detected on the ICx input line

9.1.3.9 TMSK2 — Timer interrupt mask register 2

	Address	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	State on reset
Timer interrupt mask 2 (TMSK2)	$0024	TOI	RTII	PAOVI	PAII	0	0	PR1	PR0	0000 0000

Use this 8-bit register to enable or inhibit timer overflow and real-time interrupts. The timer prescaler control bits are included in this register.

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

TOI — Timer overflow interrupt enable

1 (set) – Timer overflow interrupt requested when TOF is set.

0 (clear) – TOF interrupts disabled.

RTII — Real-time interrupt enable (refer to Section 9.4)

PAOVI — Pulse accumulator overflow interrupt enable (refer to Section 9.6.3)

PAII — Pulse accumulator input edge interrupt enable (refer to Section 9.6.3)

PR[1:0] — Timer prescaler select

9

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

These bits are used to select the prescaler divide-by ratio. In normal modes, PR[1:0] can only be written once, and the write must be within 64 cycles after reset. See Table 9-1 for specific timing values. These two bits also specify the number of divide-by-two stages that are to be inserted between the E-clock and the timer free-running counter of Timer 3. This enables Timer 1-Timer 3 synchronization. This can, however, be overridden if a different prescale is required for Timer 3, by writing to the Timer 3 prescale bits in the register TCTL6. See Section 9.3.5. The default state is that the Timer 1 prescale rate is used for Timer 3.

9.1.3.10 TFLG2 — Timer interrupt flag register 2

	Address	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	State on reset
Timer interrupt flag 2 (TFLG2)	$0025	TOF	RTIF	PAOVF	PAIF	0	0	0	0	0000 0000

Bits in this register indicate when certain timer system events have occurred. Coupled with the four high-order bits of TMSK2, the bits of TFLG2 allow the timer subsystem to operate in either a polled or interrupt driven system. Clear flags by writing a one to the corresponding bit position(s).

Note: Bits in TFLG2 correspond bit for bit with flag bits in TMSK2. Ones in TMSK2 enable the corresponding interrupt sources.

TOF — Timer overflow interrupt flag

1 (set) – TCNT has overflowed from $FFFF to $0000.

0 (clear) – No timer overflow has occurred.

RTIF — Real time (periodic) interrupt flag (refer to Section 9.4)

PAOVF — Pulse accumulator overflow interrupt flag (refer to Section 9.6)

PAIF — Pulse accumulator input edge interrupt flag (refer to Section 9.6.)

Bits [3:0] — Not implemented; always read zero

9.2 Timer 2

Timer 2 comprises a 4-stage prescaler and a 16-bit counter. It has three associated 16-bit output compare registers along with a software-programmable input capture or output compare register.

The functions of Timer 2 share I/O with the pins of port J as follows:

Pin	Alternative function
PJ3	ECIN
PJ4	OC1
PJ5	OC2
PJ6	OC3
PJ7	C4

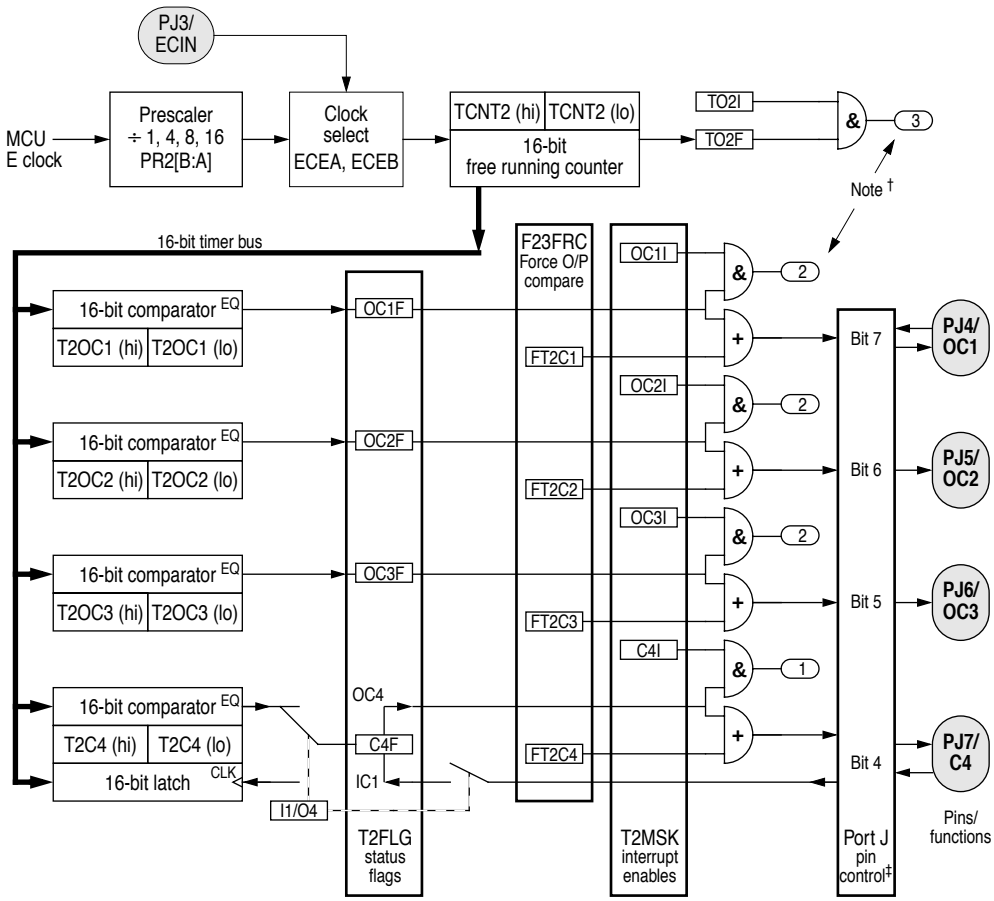
The Timer 2 prescaler is a 4 stage divider with the E clock as its input. Prescaling factors of 1, 4, 8 or 16 can be selected by the P2RA and P2RB bits in the TCTL4 register. Timer 2 also offers an

event counting mode of operation, in which the counter is clocked by an external source. In this case, the prescaler is bypassed and the external signal is used to clock the Timer 2 counter directly.

The input capture and output compare signals (OC1, OC2, OC3 and I1/OC4) are interfaced through port J pins [7:4]. If the timer is configured for event counting mode, then port J pin 3 is used as the input for the external clock. Refer to Figure 9-1 and Figure 9-3.

The Timer 2 counter can be read at any time, and can be reset to $FFFF by writing any value to it (in normal modes). In addition, the counter can be stopped and reset by setting the T2STP bit in the TCTL4 register (see Section 9.2.8).

Timer 2 can be switched to event counting mode by writing to the ECEA and ECEB bits in the TCTL4 register. When switching to event counting mode, the counter is not reset to $FFFF, but carries on from the state it was in at the time of switching. Immediately before the switch it was sourced by the internal E clock, so the first 16-stage count cycle has an offset equal to the number of E clock cycles after reset before the first write to TCTL4. The T2STP bit can be used to reset the counter to $FFFF, if required.



† Interrupt requests 1, 2 & 3 (these are further qualified by the I-bit in the CCR).

‡ Port J pin actions are controlled by the TCTL3 and TCTL4 registers.

Figure 9-3 Timer 2 capture/compare block diagram

9.2.1 Output compare

There are three dedicated output compare registers associated with Timer 2 (T20C[3:1]). In addition, a fourth register (T2C4) is used either for a fourth output compare function or for an input capture function, depending on the state of the I1/O4 bit in the TCTL4 register.

The output compare function operates in a similar fashion to that on Timer 1; a value written to an output compare register is compared to the free running counter value during each clock cycle. If a match is found, the appropriate output compare flag is set in the interrupt flag register (T2FLG). An interrupt is then generated if that particular interrupt is enabled in the interrupt mask register (T2MSK). Unlike Timer 1, which has separate interrupt requests for each interrupt, Timer 2 output compare interrupts 1,2 and 3 are associated with a single interrupt sequence. This means that flag polling is required if more than one interrupt is enabled. Refer to Figure 9-3.

In addition to an interrupt, a successful output compare can trigger a specified action at the associated output port pins PJ [7:4]. The type of action taken is defined by bits in timer control register TCTL3. This action may be forced by setting the appropriate bit in the F23FRC register (see Section 9.2.3).

9.2.2 Input capture

When configured as an input capture register, the 16-bit T2C4 register is used to latch the value of the counter when a selected transition at pin PJ7 is detected. The type of transition which triggers the capture is defined by the EDGA and EDGB bits in the TCTL4 register.

9.2.3 F23FRC — Compare force register for Timers 2 and 3.

	Address	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	State on reset
Compare force for Timers 2 and 3 (F23FRC)	$0031	FT3C1	FT3C2	FT3C3	FT3C4	FT2C1	FT2C3	FT2C3	FT2C4	0000 0000

The F23FRC register allows forced early compares for Timers 2 and 3. Bits [7:4] correspond to the four output compares of Timer 3, and bits [3:0] correspond to those of Timer 2. These bits are set for each output compare that is to be forced. The action taken as a result of a forced compare is the same as if there were a match between the OCx register and the free-running counter, except that the corresponding interrupt status flag bits are not set. The forced channels trigger their programmed pin actions to occur at the next timer count transition after the write to F23FRC.

The F23FRC bits should not normally be used on an output compare function that is programmed to toggle its output on a successful compare, because a normal compare occurring immediately before or after the force would produce a double toggle. This may be undesirable if it happens quickly, since the resulting output pulse would be very short.

FT3C[1:4] and FT2C[1:4] — Force output compares

1 (set) – A forced output compare action will occur on the appropriate pin.

0 (clear) – No action.

9.2.4 T2C4 — Timer 2 channel 4 register

	Address	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	State on reset
Timer 2 channel 4 (T2C4) high	$008A	(bit 15)	(14)	(13)	(12)	(11)	(10)	(9)	(bit 8)	1111 1111
Timer 2 channel 4 (T2C4) low	$008B	(bit 7)	(6)	(5)	(4)	(3)	(2)	(1)	(bit 0)	1111 1111

Use T2C4 as either an input capture register or an output compare register, depending on the function chosen for the PJ7 pin. To enable it as an input capture pin, set the I1/O4 bit in the timer control register 4 (TCTL4). To use it as an output compare pin, clear the I1/O4 bit. Refer to Section 9.2.8.

The T2C4 register pair resets to ones ($FFFF).

9.2.5 T2OC1–T2OC3 — Timer 2 output compare registers

	Address	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	State on reset
Timer 2 output compare 1 (T2OC1) high	$0084	(bit 15)	(14)	(13)	(12)	(11)	(10)	(9)	(bit 8)	1111 1111
Timer 2 output compare 1 (T2OC1) low	$0085	(bit 7)	(6)	(5)	(4)	(3)	(2)	(1)	(bit 0)	1111 1111
Timer 2 output compare 2 (T2OC2) high	$0086	(bit 15)	(14)	(13)	(12)	(11)	(10)	(9)	(bit 8)	1111 1111
Timer 2 output compare 2 (T2OC2) low	$0087	(bit 7)	(6)	(5)	(4)	(3)	(2)	(1)	(bit 0)	1111 1111
Timer 2 output compare 3 (T2OC3) high	$0088	(bit 15)	(14)	(13)	(12)	(11)	(10)	(9)	(bit 8)	1111 1111
Timer 2 output compare 3 (T2OC3) low	$0089	(bit 7)	(6)	(5)	(4)	(3)	(2)	(1)	(bit 0)	1111 1111

These three output compare registers are 16-bit read-write. Each is initialized to $FFFF at reset. If an output compare register is not used for an output compare function, it can be used as a storage location. A write to the high-order byte of an output compare register pair inhibits the output compare function for one bus cycle. This inhibition prevents inappropriate subsequent comparisons. Coherency requires a complete 16-bit read or write. However, if coherency is not needed, byte accesses can be used.

For output compare functions, write a comparison value to output compare registers T2OC1–T2OC3 and TI1/O4. When TCNT2 value matches the comparison value, the specified pin actions occur.

9.2.6 TCNT2 — Timer 2 counter register

	Address	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	State on reset
Timer 2 count (TCNT2) high	$0082	(bit 15)	(14)	(13)	(12)	(11)	(10)	(9)	(bit 8)	1111 1111
Timer 2 count (TCNT2) low	$0083	(bit 7)	(6)	(5)	(4)	(3)	(2)	(1)	(bit 0)	1111 1111

This 16-bit register can be read at any time. In normal modes (SMOD = 0), writing any value to the counter causes it to be reset to $FFFF. In special modes (SMOD = 1), any write to the most significant byte (MSB) causes the counter to be preset to $FFF8. This preset capability is intended for factory testing only. The counter can be stopped and reset by writing to the T2STP bit in the TCTL4 register (see Section 9.2.8).

The TCNT2 register contains the value of the 16-bit timer. A full counter read addresses the most significant byte (MSB) first. A read of this address causes the less significant byte (LSB) to be latched into a buffer for the next CPU cycle so that a double-byte read returns the full 16-bit state of the counter at the time of the MSB read cycle.

TCNT2 resets to $FFFF.

9.2.7 TCTL3 — Timer control register 3 (Timer 2)

	Address	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	State on reset
Timer control register 3 (TCTL3)	$0080	OM1	OL1	OM2	OL2	OM3	OL3	OM4	OL4	0000 0000

The bits of this register specify the action taken as a result of a successful Timer 2 OCx compare.

OM[1:4] — Output mode
OL[1:4] — Output level

OMx	OLx	Action taken on successful compare
0	0	Timer disconnected from OCx pin logic
0	1	Toggle OCx output line
1	0	Clear OCx output line to 0
1	1	Set OCx output line to 1

These control bit pairs are encoded to specify the action taken after a successful OCx compare. OC4 functions only if the I1/O4 bit in the TCTL4 register is clear.

9.2.8 TCTL4 — Timer control register 4 (Timer 2)

	Address	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	State on reset
Timer control register 4 (TCTL4)	$0081	EDGB	EDGA	PR2B	PR2A	ECEB	ECEA	T2STP	I1/04	0000 0000

EDGB and EDGA — Input capture edge control

This pair of bits configures the input capture edge detector circuits for IC1. IC1 functions only if the I1/O4 bit is set.

EDGB	EDGA	Configuration
0	0	IC1 disabled
0	1	IC1 captures on rising edges only
1	0	IC1 captures on falling edges only
1	1	IC1 captures on any edge

Note: The maximum frequency of the input clock must be less than E/2 when counting on one edge, and less that E/4 when counting on both edges.

PR2A and PR2B — Timer 2 prescaler select

These bits are used to select the prescaler divide-by ratio for Timer 2. They can only be written to once after reset.

PR2B	PR2A	Prescaler
0	0	1
0	1	4
1	0	8
1	1	16

ECEB and ECEA — Event counter edge control

These control bits configure the input clock source for the Timer 2 counter. They can be written to only once after reset.

ECEB	ECEA	Configuration
0	0	Timer 2 uses internal clock and prescaler
0	1	Count on rising edges of external clock only
1	0	Count on falling edges of external clock only
1	1	Count on any edge of external clock

T2STP — Stop Timer 2 counter

1 (set) – Timer 2 counter and prescaler are stopped and the counter is reset to $FFFF.

0 (clear) – Timer 2 counter operates normally.

I1/O4 — Input capture 1/output compare 4

1 (set) – Input capture 1 function is enabled (no OC4)

0 (clear) – Output compare 4 function is enabled (no IC1)

9.2.9 T2MSK — Timer 2 interrupt mask register

	Address	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	State on reset
Timer 2 interrupt mask (T2MSK)	$008C	OC1I	OC2I	OC3I	C4I	TO2I	0	0	0	0000 0000

Use this 8-bit register to enable or inhibit the timer input capture and output compare interrupts.

Note: Bits in T2MSK correspond bit for bit with flag bits in T2FLG. Ones in T2MSK enable the corresponding interrupt sources.

OC1I–OC3I — Output compare x interrupt enable

1 (set) – OC interrupt is enabled.

0 (clear) – OC interrupt is disabled.

If an OCxI enable bit is set when its associated OCxF flag bit is set, a hardware interrupt sequence is requested. All three interrupt enable bits are associated with a single Timer 2 output compare interrupt sequence; any successful output compare causes such an interrupt unless the corresponding OCxI bit is clear. Therefore, flag polling is required unless all but one of the interrupts have been disabled.

C4I— Input capture 1/output compare 4 interrupt enable

1 (set) – IC1/OC4 interrupt is enabled.

0 (clear) – IC1/OC4 interrupt is disabled.

When I1/O4 in TCTL4 is set, C4I is the input capture 1 interrupt enable bit.
When I1/O4 in TCTL4 is zero, C4I is the output compare 4 interrupt enable bit.

TO2I — Timer 2 overflow interrupt enable

1 (set) – Timer 2 overflow interrupt requested when T2OF is set.

0 (clear) – T2OF interrupts disabled.

Bits [2:0] — Not implemented; always read zero.

9.2.10 T2FLG — Timer 2 interrupt flag register

	Address	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	State on reset
Timer 2 interrupt flag (T2FLG)	$008D	OC1F	OC2F	OC3F	C4F	TO2F	0	0	0	0000 0000

Bits in this register indicate when timer system events have occurred. Clear flags by writing a one to the corresponding bit position(s).

Note: Bits in T2FLG correspond bit for bit with flag bits in T2MSK. Ones in T2MSK enable the corresponding interrupt sources.

OC1F–OC3F — Output compare x flag

1 (set) – Counter has reached the preset output compare x value.

0 (clear) – Counter has not reached the preset output compare x value.

These flags are set each time the counter matches the corresponding output compare x values.

C4F — Input capture 1/output compare 4 flag

This flag is set by IC1 or OC4, depending on the function enabled by I1/O4 bit in TCTL4. If C4 is configured as an input capture channel, then:

1 (set) – Selected edge has been detected on pin PJ7

0 (clear) – Selected edge has not been detected on pin PJ7.

TO2F — Timer 2 overflow flag

1 (set) – TCNT2 has overflowed from $FFFF to $0000.

0 (clear) – No Timer 2 overflow has occurred.

Bits [2:0] — Not implemented; always read zero.

9.3 Timer 3

Timer 3 is very similar in operation to Timer 2. Its functions share I/O with port K pins as follows:

Pin	Alternative function
PK3	ECIN
PK4	OC1
PK5	OC2
PK6	OC3
PK7	C4

When not in event counting mode, Timer 3 can be driven either by the E clock divided by 1, 4, or 16, or by the Timer 1 clock rate, as shown in Figure 9-1. See also Section 9.1.3.9.

Another difference between Timers 2 and 3 is that Timer 3 has only two interrupt request sequences associated with it (IC1/OC4/OC[3:1] and timer overflow), whereas Timer 2 has three (OC[3:1], IC1/OC4 and timer overflow). See Section 5 and Section 9.3.5.

Apart from these two differences, Timer 3 operates in the same way as Timer 2. The control and data registers associated with Timer 3 are described in the following paragraphs.

Note: Timer 3 output compares can be forced using bits in the F23FRC register, described in Section 9.2.3.

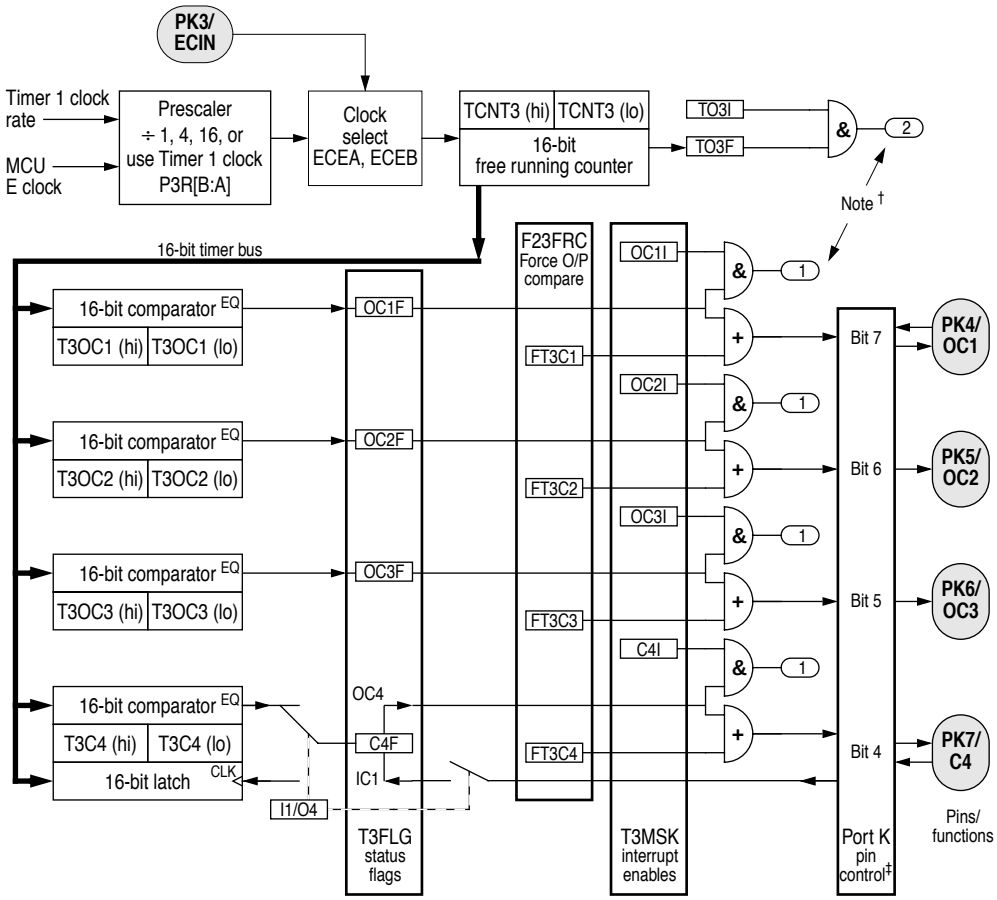
9

9.3.1 T3C4 — Timer 3 channel 4 register

	Address	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	State on reset
Timer 3 channel 4 (T2C4) high	$009A	(bit 15)	(14)	(13)	(12)	(11)	(10)	(9)	(bit 8)	1111 1111
Timer 3 channel 4(T2C4) low	$009B	(bit 7)	(6)	(5)	(4)	(3)	(2)	(1)	(bit 0)	1111 1111

Use T3C4 as either an input capture register or an output compare register, depending on the function chosen for the PK7 pin. To enable it as an input capture pin, set the I1/O4 bit in the timer control register 4 (TCTL6). To use it as an output compare pin, clear the I1/O4 bit. Refer to Section 9.3.5.

The T3C4 register pair resets to ones ($FFFF).



† Interrupt requests 1 and 2 (these are further qualified by the I-bit in the CCR).

‡ Port K pin actions are controlled by the TCTL5 and TCTL6 registers.

Figure 9-4 Timer 3 capture/compare block diagram

9.3.2 T3OC1–T3OC3 — Timer 3 output compare registers

	Address	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	State on reset
Timer 3 output compare 1 (T3OC1) high	$0094	(bit 15)	(14)	(13)	(12)	(11)	(10)	(9)	(bit 8)	1111 1111
Timer 3 output compare 1 (T3OC1) low	$0095	(bit 7)	(6)	(5)	(4)	(3)	(2)	(1)	(bit 0)	1111 1111
Timer 3 output compare 2 (T3OC2) high	$0096	(bit 15)	(14)	(13)	(12)	(11)	(10)	(9)	(bit 8)	1111 1111
Timer 3 output compare 2 (T3OC2) low	$0097	(bit 7)	(6)	(5)	(4)	(3)	(2)	(1)	(bit 0)	1111 1111
Timer 3 output compare 3 (T3OC3) high	$0098	(bit 15)	(14)	(13)	(12)	(11)	(10)	(9)	(bit 8)	1111 1111
Timer 3 output compare 3 (T3OC3) low	$0099	(bit 7)	(6)	(5)	(4)	(3)	(2)	(1)	(bit 0)	1111 1111

These three output compare registers are 16-bit read-write. Each is initialized to $FFFF at reset. If an output compare register is not used for an output compare function, it can be used as a storage location. A write to the high-order byte of an output compare register pair inhibits the output compare function for one bus cycle. This inhibition prevents inappropriate subsequent comparisons. Coherency requires a complete 16-bit read or write. However, if coherency is not needed, byte accesses can be used.

For output compare functions, write a comparison value to output compare registers T3OC1–T3OC3 and TI1/O4. When the TCNT3 value matches the comparison value, specified pin actions occur.

9.3.3 TCNT3 — Timer 3 counter register

	Address	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	State on reset
Timer 3 count (TCNT3) high	$0092	(bit 15)	(14)	(13)	(12)	(11)	(10)	(9)	(bit 8)	1111 1111
Timer 3 count (TCNT3) low	$0093	(bit 7)	(6)	(5)	(4)	(3)	(2)	(1)	(bit 0)	1111 1111

This register can be read at any time. In normal modes (SMOD = 0), writing any value to the counter causes it to be rest to $0000. In special modes (SMOD = 1), any write to the most significant byte (MSB) causes the counter to be preset to $FFF8.This preset capability is intended for factory testing only. The counter can be stopped and reset by writing to the T3STP bit in the TCTL6 register (see Section 9.3.5).

The 16-bit read-only TCNT3 register contains the value of the 16-bit timer. A full counter read addresses the most significant byte (MSB) first. A read of this address causes the less significant byte (LSB) to be latched into a buffer for the next CPU cycle so that a double-byte read returns the full 16-bit state of the counter at the time of the MSB read cycle.

TCNT3 resets to $FFFF.

9.3.4 TCTL5 — Timer control register 5 (Timer 3)

	Address	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	State on reset
Timer control register 5 (TCTL5)	$0090	OM1	OL1	OM2	OL2	OM3	OL3	OM4	OL4	0000 0000

The bits of this register specify the action taken as a result of a successful Timer 3 OCx compare.

OM[1:4] — Output mode
OL[1:4] — Output level

OMx	OLx	Action taken on successful compare
0	0	Timer disconnected from OCx pin logic
0	1	Toggle OCx output line
1	0	Clear OCx output line to 0
1	1	Set OCx output line to 1

These control bit pairs are encoded to specify the action taken after a successful OCx compare. OC4 functions only if the I1/O4 bit in the TCTL6 register is clear.

9.3.5 TCTL6 — Timer control register 6 (Timer 3)

	Address	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	State on reset
Timer control register 6 (TCTL6)	$0091	EDGB	EDGA	PR3B	PR3A	ECEB	ECEA	T3STP	I1/04	0000 0000

EDGB and EDGA — Input capture edge control

This pair of bits configures the input capture edge detector circuits for IC1. IC1 functions only if the I1/O4 bit is set.

EDGB	EDGA	Configuration
0	0	IC1 disabled
0	1	IC1 captures on rising edges only
1	0	IC1 captures on falling edges only
1	1	IC1 captures on any edge

Note: The maximum frequency of the input clock must be less than E/2 when counting on one edge, and less that E/4 when counting on both edges.

PR3A and PR3B — Timer 3 prescaler select

These bits are used to select the prescaler divide-by ratio for Timer 3. They can only be written to once after reset. If PR3B and PR3A are both cleared, then Timer 3 is synchronized to the prescaled Timer 1 rate, which is determined by the PR1 and PR0 bits in the TMSK2 register. See Section 9.1.3.9 and Figure 9-1.

PR3B	PR3A	Prescaler
0	0	Use Timer 1 rate
0	1	1
1	0	4
1	1	16

ECEB and ECEA — Event counter edge control

These control bits configure the input clock source for the Timer 3 counter. They can be written to only once after reset.

ECEB	ECEA	Configuration
0	0	Timer 2 uses internal clock and prescaler
0	1	Count on rising edges of external clock only
1	0	Count on falling edges of external clock only
1	1	Count on any edge of external clock

T3STP — Stop Timer 3 counter

1 (set) – Timer 3 counter and prescaler are stopped and the counter is reset to $FFFF.

0 (clear) – Timer 3 counter operates normally.

I1/O4 — Input capture 1/output compare 4

1 (set) – Input capture 1 function is enabled (no OC4)

0 (clear) – Output compare 4 function is enabled (no IC1)

9.3.6 T3MSK — Timer 3 interrupt mask register

	Address	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	State on reset
Timer 3 interrupt mask (T2MSK)	$009C	OC1I	OC2I	OC3I	C4I	TO2I	0	0	0	0000 0000

Use this 8-bit register to enable or inhibit the timer input capture and output compare interrupts.

Note: Bits in T3MSK correspond bit for bit with flag bits in T3FLG. Ones in T3MSK enable the corresponding interrupt sources.

OC1I–OC3I — Output compare x interrupt enable

1 (set) – OCx interrupt is enabled.

0 (clear) – OCx interrupt is disabled.

If an OCxI enable bit is set when its associated OCxF flag bit is set, a hardware interrupt sequence is requested. These three bits and the C4I bit are associated with a single Timer 3 output compare interrupt sequence; any successful output compare or input capture causes such an interrupt unless the corresponding OCxI or C4I bit is clear. Therefore flag polling is required if more than one interrupt is enabled.

C4I— Input capture 1/output compare 4 interrupt enable

1 (set) – IC1/OC4 interrupt is enabled.

0 (clear) – IC1/OC4 interrupt is disabled.

When I1/O4 in TCTL4 is set, C4I is the input capture 1 interrupt enable bit.
When I1/O4 in TCTL4 is zero, C4I is the output compare 4 interrupt enable bit.

TO3I — Timer 3 overflow interrupt enable

1 (set) – Timer 3 overflow interrupt requested when T3OF is set.

0 (clear) – T3OF interrupts disabled.

Bits [2:0] — Not implemented; always read zero.

9.3.7 T3FLG — Timer 3 interrupt flag register

	Address	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	State on reset
Timer 3 interrupt flag (T3FLG)	$009D	OC1F	OC2F	OC3F	C4F	TO2F	0	0	0	0000 0000

Bits in this register indicate when timer system events have occurred. Clear flags by writing a one to the corresponding bit position(s).

Note: Bits in T3FLG correspond bit for bit with flag bits in T3MSK. Ones in T3MSK enable the corresponding interrupt sources.

OC1F–OC3F — Output compare x flag

1 (set) – Counter has reached the preset output compare x value.

0 (clear) – Counter has not reached the preset output compare x value.

These flags are set each time the counter matches the corresponding output compare x values.

C4F — Input capture 1/output compare 4flag

This flag is set by IC1 or OC4, depending on the function enabled by the I1/O4 bit in TCTL6. If C4 is configured as an input capture channel, then:

1 (set) – Selected edge has been detected on pin PK7

0 (clear) – Selected edge has not been detected on pin PK7.

TO3F — Timer 3 overflow flag

1 (set) – TCNT3 has overflowed from $FFFF to $0000.

0 (clear) – No Timer 3 overflow has occurred.

Bits [2:0] — Not implemented; always read zero.

9.4 Real-time interrupt

The real-time interrupt (RTI) feature, used to generate hardware interrupts at a fixed periodic rate, is controlled and configured by two bits (RTR1 and RTR0) in the pulse accumulator control (PACTL) register. The RTII bit in the TMSK3 register enables the interrupt capability. The four different rates available are a product of the MCU oscillator frequency and the value of bits RTR[1:0]. Refer to Table 9-2, which shows the periodic real-time interrupt rates.

Table 9-2 RTI periodic rates

RTR[1:0]	E = 4 MHz	E = x MHz
0 0	1.02 ms	2^{12}/E
0 1	2.05 ms	2^{13}/E
1 0	4.09 ms	2^{14}/E
1 1	8.19 ms	2^{15}/E

The clock source for the RTI function is a free-running clock that cannot be stopped or interrupted except by reset. This clock causes the time between successive RTI timeouts to be a constant that is independent of the software latency associated with flag clearing and service. For this reason, an RTI period starts from the previous timeout, not from when RTIF is cleared.

Every timeout causes the RTIF bit in TFLG2 to be set, and if RTII is set, an interrupt request is generated. After reset, one entire RTI period elapses before the RTIF flag is set for the first time. Refer to the TMSK2, TFLG2, and PACTL registers.

9.4.1 TMSK2 — Timer interrupt mask register 2

	Address	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	State on reset
Timer interrupt mask 2 (TMSK2)	$0024	TOI	RTII	PAOVI	PAII	0	0	PR1	PR0	0000 0000

This register contains the real-time interrupt enable bit.

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

TOI — Timer overflow interrupt enable (refer to Section 9.1.3.9)

RTII — Real-time interrupt enable

1 (set) – Real time interrupt requested when RTIF is set.

0 (clear) – Real time interrupts disabled.

PAOVI — Pulse accumulator overflow interrupt enable (refer to Section 9.6)

PAII — Pulse accumulator input edge (refer to Section 9.6)

PR[1:0] — Timer prescaler select (refer to Section 9.1.3.9)

9.4.2 TFLG2 — Timer interrupt flag register 2

	Address	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	State on reset
Timer interrupt flag 2 (TFLG2)	$0025	TOF	RTIF	PAOVF	PAIF	0	0	0	0	0000 0000

Bits of this register indicate the occurrence of timer system events. Coupled with the four high-order bits of TMSK2, the bits of TFLG2 allow the timer subsystem to operate in either a polled or interrupt driven system. Clear flags by writing a one to the corresponding bit position(s).

Note: Bits in TFLG2 correspond bit for bit with flag bits in TMSK2. Ones in TMSK2 enable the corresponding interrupt sources.

TOF — Timer overflow interrupt flag

1 (set) – The timer has overflowed, from $FFFF to $0000.

0 (clear) – No timer overflow has occurred.

RTIF — Real-time interrupt flag

1 (set) – RTI period has elapsed.

0 (clear) – RTI flag has been cleared.

The RTIF status bit is automatically set to one at the end of every RTI period.

PAOVF — Pulse accumulator overflow interrupt flag (refer to Section 9.6)

PAIF — Pulse accumulator input edge interrupt flag (refer to Section 9.6)

Bits [3:0] — Not implemented; always read zero

9.4.3 PACTL — Pulse accumulator control register

	Address	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	State on reset
Pulse accumulator control (PACTL)	$0026	0	PAEN	PAMOD	PEDGE	0	I4/O5	RTR1	RTR0	0000 0000

Bits RTR[1:0] of this register select the rate for the RTI system. The remaining bits control the pulse accumulator and IC4/OC5 functions.

Bits 7, 3 — Not implemented; always read zero

PAEN — Pulse accumulator system enable (refer to Section 9.6)

PAMOD — Pulse accumulator mode (refer to Section 9.6)

PEDGE — Pulse accumulator edge control (refer to Section 9.6)

I4/O5 — Input capture 4/output compare (refer to Section 9.6)

RTR[1:0] — RTI interrupt rate select

These two bits determine the rate at which the RTI system requests interrupts. The RTI system is driven by an $E/2^{12}$ clock rate that is compensated so it is independent of the timer prescaler. These two control bits select an additional division factor. Refer to Table 9-2.

9.5 Computer operating properly watchdog function

The clocking chain for the COP function, tapped off from the main timer divider chain, is only superficially related to the main timer system. The CR[1:0] bits in the OPTION register and the NOCOP bit in the CONFIG register determine the status of the COP function. One additional register, COPRST, is used to arm and clear the COP watchdog reset system. Refer to Section 5 for a more detailed discussion of the COP function.

9.6 Pulse accumulator

The MC68HC11KW1 has an 8-bit counter that can be configured to operate either as a simple event counter, or for gated time accumulation, depending on the state of the PAMOD bit in the PACTL register. Refer to the pulse accumulator block diagram, Figure 9-5.

In the event counting mode, the 8-bit counter is clocked to increasing values by an external pin. The maximum clocking rate for the external event counting mode is the E clock divided by two. In gated time accumulation mode, a free-running E clock ÷ 64 signal drives the 8-bit counter, but only

while the external PAI pin is activated. Refer to Table 9-3. The pulse accumulator counter can be read or written at any time.

Table 9-3 Pulse accumulator timing

Crystal frequency	E clock	Cycle time	64/E	PACNT overflow
16.0 MHz	4.0 MHz	250 ns	16.0 µs	4.096 ms

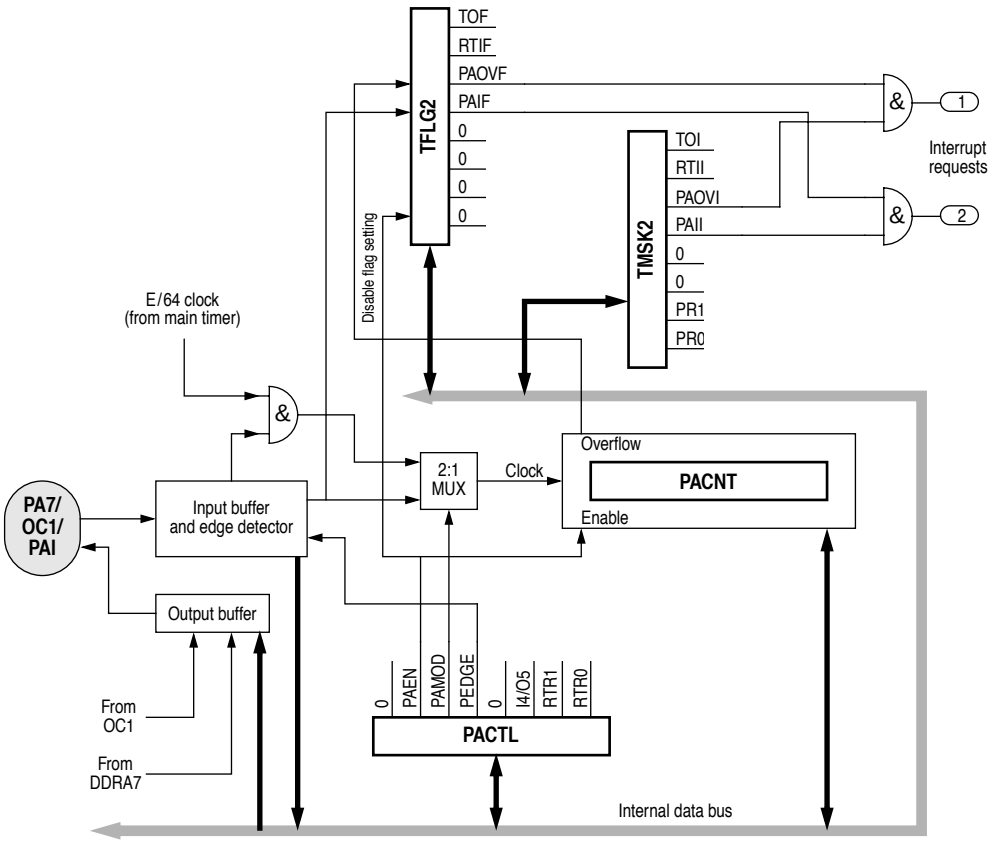


Figure 9-5 Pulse accumulator block diagram

Pulse accumulator control bits are also located within two timer registers, TMSK2 and TFLG2, as described in the following paragraphs.

9.6.1 PACTL — Pulse accumulator control register

	Address	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	State on reset
Pulse accumulator control (PACTL)	$0026	0	PAEN	PAMOD	PEDGE	0	I4/O5	RTR1	RTR0	0000 0000

Four of this register's bits control an 8-bit pulse accumulator system. Another bit enables either the OC5 function or the IC4 function, while two other bits select the rate for the real-time interrupt system.

Bits [7, 3] — Not implemented; always read zero

PAEN — Pulse accumulator system enable

1 (set) – Pulse accumulator enabled.

0 (clear) – Pulse accumulator disabled.

PAMOD — Pulse accumulator mode

1 (set) – Gated time accumulation mode.

0 (clear) – Event counter mode.

PEDGE — Pulse accumulator edge control

This bit has different meanings depending on the state of the PAMOD bit, as shown:

PAMOD	PEDGE	Action of clock
0	0	PAI falling edge increments the counter.
0	1	PAI rising edge increments the counter.
1	0	A zero on PAI inhibits counting.
1	1	A one on PAI inhibits counting.

I4/O5 — Input capture 4/output compare 5

1 (set) – Input capture 4 function is enabled (no OC5).

0 (clear) – Output compare 5 function is enabled (no IC4).

RTR[1:0] — RTI interrupt rate selects (refer to Section 9.4)

9.6.2 PACNT — Pulse accumulator count register

	Address	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	State on reset
Pulse accumulator count (PACNT)	$0027	(bit 7)	(6)	(5)	(4)	(3)	(2)	(1)	(bit 0)	undefined

This 8-bit read/write register contains the count of external input events at the PAI input, or the accumulated count. In gated time accumulation mode, PACNT is readable even if PAI is not active. The counter is not affected by reset and can be read or written at any time. Counting is synchronized to the internal PH2 clock so that incrementing and reading occur during opposite half cycles.

9.6.3 Pulse accumulator status and interrupt bits

The pulse accumulator control bits, PAOVI and PAII, PAOVF, and PAIF are located within timer registers TMSK2 and TFLG2.

9.6.3.1 TMSK2 — Timer interrupt mask 2 register

	Address	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	State on reset
Timer interrupt mask 2 (TMSK2)	$0024	TOI	RTII	PAOVI	PAII	0	0	PR1	PR0	0000 0000

9.6.3.2 TFLG2 — Timer interrupt flag 2 register

	Address	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	State on reset
Timer interrupt flag 2 (TFLG2)	$0025	TOF	RTIF	PAOVF	PAIF	0	0	0	0	0000 0000

PAOVI and PAOVF — Pulse accumulator interrupt enable and overflow flag

The PAOVF status bit is set each time the pulse accumulator count rolls over from $FF to $00. To clear this status bit, write a one in the corresponding data bit position (bit 5) of the TFLG2 register. The PAOVI control bit allows configuring the pulse accumulator overflow for polled or interrupt-driven operation and does not affect the state of PAOVF. When PAOVI is zero, pulse accumulator overflow interrupts are inhibited, and the system operates in a polled mode, which requires that PAOVF be polled by user software to determine when an overflow has occurred. When the PAOVI control bit is set, a hardware interrupt request is generated each time PAOVF is set. Before leaving the interrupt service routine, software must clear PAOVF by writing to the TFLG2 register.

PAII and PAIF — Pulse accumulator input edge interrupt enable and flag

The PAIF status bit is automatically set each time a selected edge is detected at the PA7/PAI/OC1 pin. To clear this status bit, write to the TFLG2 register with a one in the corresponding data bit position (bit 4). The PAII control bit allows configuring the pulse accumulator input edge detect for polled or interrupt-driven operation but does not affect setting or clearing the PAIF bit. When PAII is zero, pulse accumulator input interrupts are inhibited, and the system operates in a polled mode. In this mode, the PAIF bit must be polled by user software to determine when an edge has occurred. When the PAII control bit is set, a hardware interrupt request is generated each time PAIF is set. Before leaving the interrupt service routine, software must clear PAIF by writing to the TFLG register.

9.7 Pulse-width modulation (PWM) timer

The PWM timer subsystem provides up to four 8-bit pulse-width modulated waveforms on the port H pins. Channel pairs can be concatenated to create 16-bit PWM outputs. Three clock sources (A, B, and S) and a flexible clock select scheme give the PWM a wide range of frequencies.

Pin	Alternative function
PH0	PW1
PH1	PW2
PH2	PW3
PH3	PW4

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

Each channel also has a separate 8-bit counter, period register, and duty cycle register. The period and duty cycle registers are double buffered so that if they are changed while the channel is enabled, the change does not take effect until the counter rolls over or the channel is disabled. A new period or duty cycle can be forced into effect immediately by writing to the period or duty cycle register and then writing to the counter.

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

In 16-bit mode, duty cycle resolution of up to 15 parts per million can be achieved (at a PWM frequency of 60Hz). In the same system, a PWM frequency of 1kHz corresponds to a duty cycle resolution of 0.025%.

9.7.1 PWM timer block diagram

Figure 9-6 shows the block diagram of the PWM timer subsystem. Three different clock sources are selectable and provide inputs to the control registers. Each of the four channels has a counter, a period register, and a duty register. The waveform output is the result of a match between the period register (PWPERx) and the value in the counter (PWCNTx). The duty register (PWDTYx) changes the state of the output during the period to determine the duty cycle.

9.7.2 PWCLK — PWM clock prescaler and 16-bit select register

	Address	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	State on reset
Pulse width clock select (PWCLK)	$0060	CON34	CON12	PCKA2	PCKA1	0	PCKB3	PCKB2	PCKB1	0000 0000

This register contains bits for selecting the 16-bit PWM options and the prescaler values for the clocks.

9.7.2.1 16-bit PWM function

The PWCLK register contains two control bits, each of which is used to concatenate a pair of PWM channels into one 16-bit channel. Channels 3 and 4 are concatenated with the CON34 bit, and channels 1 and 2 are concatenated with the CON12 bit.

When the 16-bit concatenated mode is selected, the clock source is determined by the low order channel. Channel 2 is the low order channel when channels 1 and 2 are concatenated. Channel 4 is the low order channel when channels 3 and 4 are concatenated. The pins associated with channels 1 and 3 can be used for general-purpose I/O when 16-bit PWM mode is selected.

Channel 1 registers are the high order byte of the double-byte channel when channels 1 and 2 are concatenated. Channel 3 registers are the high order byte of the double-byte channel when channels 3 and 4 are concatenated. Reads of the high order byte cause the low order byte to be latched for one cycle to guarantee that double byte reads are accurate. Writes to the low byte of the counter cause reset of the entire counter. Writes to the upper bytes of the counter have no effect.

CON34 — Concatenate channels 3 and 4

1 (set) – Channels 3 and 4 are concatenated into one 16-bit PWM channel.

0 (clear) – Channels 3 and 4 are separate 8-bit PWMs.

When concatenated, channel 3 is the high-order byte and the channel 4 pin (PH3) is the output.

CON12 — Concatenate Channels 1 and 2

1 (set) – Channels 1 and 2 are concatenated into one 16-bit PWM channel.

0 (clear) – Channels 1 and 2 are separate 8-bit PWMs.

When concatenated, channel 1 is the high-order byte and the channel 2 pin (PH1) is the output.

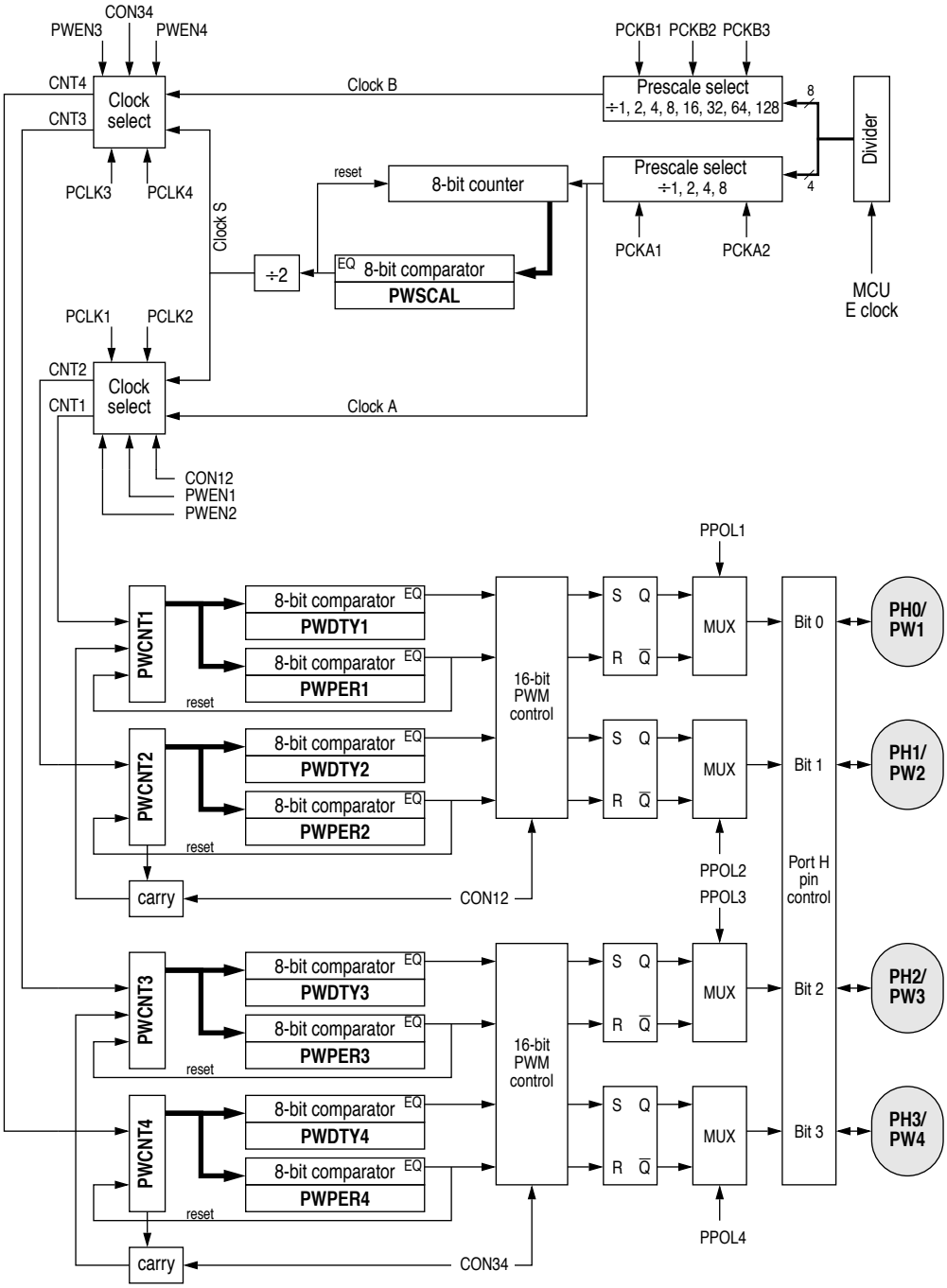


Figure 9-6 PWM timer block diagram

9.7.2.2 Clock prescaler selection

The three available clocks are clock A, clock B, and clock S (scaled). Clock A can be software selected to be E, E/2, E/4, or E/8. Clock B can be software selected to be E, E/2, E/4,..., E/128. The scaled clock (clock S) uses clock A as an input and divides it with a reloadable counter. The rates available are software selectable to be clock A/2, down to clock A /512.

The clock source portion of the block diagram shows the three clock sources and how the scaled clock is created. Clock A is an input to an 8-bit counter which is then compared to a user programmable scale value. When they match, this circuit has an output that is divided by two and the counter is reset.

Each PWM timer channel can be driven by one of two clocks. Refer to Figure 9-6.

PCKA[2:1] — Prescaler for clock A

Determines the frequency of clock A. Refer to Table 9-4.

Bit 3 — Not implemented; always reads zero

PCKB[3:1] — Prescaler for clock B

Determines the frequency of clock B. Refer to Table 9-4.

Table 9-4 Clock A and clock B prescalers

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

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

9.7.3 PWPOL — PWM timer polarity & clock source select register

	Address	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	State on reset
Pulse width polarity select (PWPOL)	$0061	PCLK4	PCLK3	PCLK2	PCLK1	PPOL4	PPOL3	PPOL2	PPOL1	0000 0000

PCLK[4:3] — Pulse width channel 4/3 clock select

1 (set) – Clock S is source.

0 (clear) – Clock B is source.

PCLK[2:1] — Pulse width channel 2/1 clock select

1 (set) – Clock S is source.

0 (clear) – Clock A is source.

PPOL[4:1] — Pulse width channel x polarity

1 (set) – PWM channel x output is high at the beginning of the clock cycle and goes low when duty count is reached.

0 (clear) – PWM channel x output is low at the beginning of the clock cycle and goes high when duty count is reached.

Each channel has a polarity bit that allows a cycle to start with either a high or a low level. This is shown on the block diagram, Figure 9-6, as a selection of either the Q output or the $\overline{Q}$ output of the PWM output flip flop. When one of the bits in the PWPOL register is set, the associated PWM channel output is high at the beginning of the clock cycle, then goes low when the duty count is reached.

9.7.4 PWSCAL — PWM timer prescaler register

	Address	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	State on reset
Pulse width scale (PWSCAL)	$0062	(bit 7)	(6)	(5)	(4)	(3)	(2)	(1)	(bit 0)	0000 0000

Scaled clock S is generated by dividing clock A by the value in PWSCAL, then dividing the result by two. If PWSCAL = $00, clock A is divided by 256, then divided by two to generate clock S.

9.7.5 PWEN — PWM timer enable register

	Address	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	State on reset
Pulse width enable (PWEN)	$0063	TPWSL	DISCP	0	0	PWEN4	PWEN3	PWEN2	PWEN1	0000 0000

Each timer has an enable bit to start its waveform output. Writing any of these PWENx bits to one causes the associated port line to become an output regardless of the state of the associated DDR bit. This does not change the state of the DDR bit and when PWENx returns to zero the DDR bit again controls I/O state. On the front end of the PWM timer the clock is connected to the PWM circuit by the PWENx enable bit being high. There is a synchronizing circuit to guarantee that the clock will only be enabled or disabled at an edge.

PWEN contains 4 PWM enable bits — one for each channel. When an enable bit is set to one, the pulse modulated signal becomes available at the associated port pin.

TPWSL — PWM scaled clock test bit (Test mode only)

1 (set) – Clock S output to PWSCAL register (Test only).

0 (clear) – Normal operation.

When TPWSL is one, clock S from the PWM timer is output to PWSCAL register. Normal writing to the PWSCAL register still functions.

DISCP — Disable compare scaled E clock (Test mode only)

1 (set) – Match of period does not reset associated count register (Test only).

0 (clear) – Normal operation.

Bits [5:4] — Not implemented; always read zero

PWEN[4:1] — Pulse width channels 4–1

1 (set) – Channel enabled on the associated port pin.

0 (clear) – Channel disabled.

9.7.6 PWCNT1–4 — PWM timer counter registers 1 to 4

	Address	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	State on reset
Pulse width count 1 (PWCNT1)	$0064	(bit 7)	(6)	(5)	(4)	(3)	(2)	(1)	(bit 0)	0000 0000
Pulse width count 2 (PWCNT2)	$0065	(bit 7)	(6)	(5)	(4)	(3)	(2)	(1)	(bit 0)	0000 0000
Pulse width count 3 (PWCNT3)	$0066	(bit 7)	(6)	(5)	(4)	(3)	(2)	(1)	(bit 0)	0000 0000
Pulse width count 4 (PWCNT4)	$0067	(bit 7)	(6)	(5)	(4)	(3)	(2)	(1)	(bit 0)	0000 0000

Each channel has its own counter which can be read at any time without affecting the count or the operation of the PWM channel. Writing to a counter causes it to be reset to $00; this is generally done before the counter is enabled. A counter may also be written to whilst it is enabled; this may cause a truncated PWM period.

9.7.7 PWPER1–4 — PWM timer period registers 1 to 4

	Address	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	State on reset
Pulse width period 1 (PWPER1)	$0068	(bit 7)	(6)	(5)	(4)	(3)	(2)	(1)	(bit 0)	1111 1111
Pulse width period 2 (PWPER2)	$0069	(bit 7)	(6)	(5)	(4)	(3)	(2)	(1)	(bit 0)	1111 1111
Pulse width period 3 (PWPER3)	$006A	(bit 7)	(6)	(5)	(4)	(3)	(2)	(1)	(bit 0)	1111 1111
Pulse width period 4 (PWPER4)	$006B	(bit 7)	(6)	(5)	(4)	(3)	(2)	(1)	(bit 0)	1111 1111

There is one period register for each channel. The value in this register determines the period of the associated PWM timer channel. PWPERx is connected internally to a buffer which compares directly with the counter register. The period value in PWPERx is loaded into the buffer when the counter is cleared by the termination of the previous period or by a write to the counter. This register can be written at any time, and the written value will take effect from the start of the next PWM timer cycle. Reads of this register return the most recent value written.

9.7.8 PWDTY1–4 — PWM timer duty cycle registers 1 to 4

	Address	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	State on reset
Pulse width duty 1 (PWDTY1)	$006C	(bit 7)	(6)	(5)	(4)	(3)	(2)	(1)	(bit 0)	1111 1111
Pulse width duty 2 (PWDTY2)	$006D	(bit 7)	(6)	(5)	(4)	(3)	(2)	(1)	(bit 0)	1111 1111
Pulse width duty 3 (PWDTY3)	$006E	(bit 7)	(6)	(5)	(4)	(3)	(2)	(1)	(bit 0)	1111 1111
Pulse width duty 4 (PWDTY4)	$006F	(bit 7)	(6)	(5)	(4)	(3)	(2)	(1)	(bit 0)	1111 1111

There is one duty register for each channel. The value in this register determines the duty cycle of the associated PWM timer channel. PWDTYx is compared to the counter contents and if they are equal, a match occurs and the output goes to the state defined by the associated polarity bit. If the register is written while the channel is enabled, then the new value is held in a buffer until the counter rolls over or the channel is disabled. Reads of this register return the most recent value written.

Note: If PWDTYx ≥ PWPERx then there will be no change of state due to the duty cycle value. In addition, if the duty register is set to $00, then the output will always be in the state which would normally be result from the duty change of state (see also Section 9.7.9).

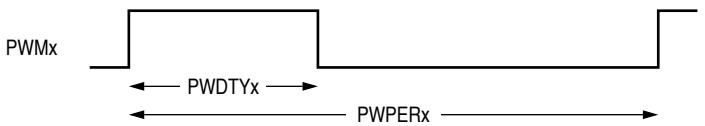


Figure 9-7 PWM duty cycle

9.7.9 Boundary cases

The following boundary conditions apply to the values stored in the PWDTYx and PWPERx registers and the PPOLx bits:

- If PWDTYx = $00, PWPERx > $00 and PPOLx = 0 then the output is always high.
- If PWDTYx = $00, PWPERx > $00 and PPOLx = 1 then the output is always low.
- If PWDTYx ≥ PWPERx and PPOLx = 0 then the output is always low.
- If PWDTYx ≥ PWPERx and PPOLx = 1 then the output is always high.
- If PWPERx = $00 and PPOLx = 0 then the output is always low.
- If PWPERx = $00 and PPOLx = 1 then the output is always high.

10
ANALOG-TO-DIGITAL CONVERTER

The analog-to-digital converter system consists of a single 10-bit successive approximation type converter and a 16-channel multiplexer. Ten of the channels are connected to pins on the MC68HC11KW1 (ports E and G), two are unused and the remaining four channels are dedicated to internal reference points or test functions. The A/D converter shares input pins with port E and port G as follows:

Pin	Alternative function
PG6	AN0
PG7	AN1
PE0	AN2
PE1	AN3
PE2	AN4
PE3	AN5
PE4	AN6
PE5	AN7
PE6	AN8
PE7	AN9

There are eight 10-bit result registers, and control logic allows for four or eight consecutive conversions before stopping, or for conversions to continue with the newest conversion overwriting the oldest result register. Also, conversions can be performed several times on a single selected channel, or consecutively on a selected group of four channels. In addition, the control logic can convert eight channels and then stop, or convert continuously.

Two dedicated lines (VRH and VRL) are provided for the reference voltage inputs. These pins may be connected to a separate or isolated power supply to ensure full accuracy of the A/D conversion. Furthermore, the A/D converter supply pins VDDAD and VSSAD allow the user to isolate the A/D voltage supply from the main VDD/VSS supply lines. This reduces the effects that noise from the CPU core has on the A/D converter, thus improving accuracy.

The 10-bit A/D converter accepts analog inputs ranging from V_{RL} to V_{RH}. Smaller input ranges can also be obtained by adjusting V_{RL} and V_{RH} to the desired upper and lower limits. Conversion is specified and tested for V_{RL} = 0 volts and V_{RH} = 5 volts. The A/D system can be operated with V_{RL} below V_{DD} and/or V_{RL} above V_{SS} as long as V_{RH} is above V_{RL} by enough to support the conversions (2.5 to 5.0 volts).

Each set of four conversions takes 144 cycles of the E-clock, provided that E is greater than or equal to 750 kHz. If E is less than 750 kHz, an internal R-C oscillator, which is nominally 1.5 MHz, must be used for the A/D conversion clock. When the internal R-C oscillator is being used as the conversion clock, the conversion complete flag (CCF) must be used to determine when a conversion sequence has been completed. When using the internal R-C oscillator for A/D conversions, the sample and conversion process runs at the nominal 1.5 MHz rate; however, the conversion results must be transferred to the MCU result registers synchronously with the MCU E-clock, so conversion time is limited to a maximum of one channel per E-clock cycle. Alternatively, if the R-C oscillator is not being used and E is greater than 2.1 MHz, the conversion frequency can be halved to E/2 under control of the ADER bit in the ADFRQ register. Note that in this operating mode, each set of four conversions takes 288 cycles of the E clock.

Two control bits in the OPTION register control the basic configuration of the A/D system. The A/D power-up bit (ADPU) allows the system to be disabled, resulting in reduced power consumption when the A/D system is not being used. Any conversion which is in process when ADPU is written to zero will be aborted. A delay of typically 100 microseconds is required after turning on the A/D (by writing ADPU from 0 to 1) for the analog and comparator sections to stabilize. The CSEL bit is used to select either the internal R-C oscillator or the MCU E-clock as the A/D system clock source.

10.1 Conversion process

The A/D converter is ratiometric. An input voltage equal to V_{RH} converts to $FFC0 (full scale) and an input voltage equal to V_{RL} converts to $0000. An input voltage greater than V_{RH} will convert to $FFC0 with no overflow indication. Note that the six least significant bits always read zero. For ratiometric conversions, the source of each analog input should use V_{RH} as the supply voltage and be referenced to V_{RL}.

The A/D reference inputs are applied to a precision internal digital-to-analog converter. Control logic drives this D/A and the analog output is successively compared to the selected analog input which was sampled at the beginning of the conversion time. The conversion process is monotonic with no missing codes.

10.2 Channel assignments

A multiplexer allows the single A/D converter to select one of sixteen analog signals. Ten of these channels are supported on Port E and G input pins. Of the six other channels, two are reserved for future use and four are for internal reference points and testing purposes. Table 10-1 shows the signals selected by the channel select bits (CD, CC, CB, CA) in the ADCTL register. The CONV8 bit selects either four or eight conversions. All “reserved” channels are connected to V_{RL}.

Table 10-1 Channel assignments

CD	CC	CB	CA	Channel signal	CONV8 = 0 Result in register if		CONV8 = 1 Result in register if	
					MULT = 1	MULT = 0	MULT = 1	MULT = 0
0	0	0	0	AN2 (on PE0)	ADR5	ADR5 – 8	ADR1	ADR1 – 8
0	0	0	1	AN3 (on PE1)	ADR6	ADR5 – 8	ADR2	ADR1 – 8
0	0	1	0	AN4 (on PE2)	ADR7	ADR5 – 8	ADR3	ADR1 – 8
0	0	1	1	AN5 (on PE3)	ADR8	ADR5 – 8	ADR4	ADR1 – 8
0	1	0	0	AN6 (on PE4)	ADR5	ADR5 – 8	ADR5	ADR1 – 8
0	1	0	1	AN7 (on PE5)	ADR6	ADR5 – 8	ADR6	ADR1 – 8
0	1	1	0	AN8 (on PE6)	ADR7	ADR5 – 8	ADR7	ADR1 – 8
0	1	1	1	AN9 (on PE7)	ADR8	ADR5 – 8	ADR8	ADR1 – 8
1	0	0	0	AN0 (on PG6)	ADR5	ADR5 – 8	ADR1	ADR1 – 8
1	0	0	1	AN1 (on PG7)	ADR6	ADR5 – 8	ADR2	ADR1 – 8
1	0	1	0	Reserved	ADR7	ADR5 – 8	ADR3	ADR1 – 8
1	0	1	1	Reserved	ADR8	ADR5 – 8	ADR4	ADR1 – 8
1	1	0	0	VRH(1)	ADR5	ADR5 – 8	ADR5	ADR1 – 8
1	1	0	1	VRL(1)	ADR6	ADR5 – 8	ADR6	ADR1 – 8
1	1	1	0	VRH/2(1)	ADR7	ADR5 – 8	ADR7	ADR1 – 8
1	1	1	1	Test (reserved)(1)	ADR8	ADR5 – 8	ADR8	ADR1 – 8

(1) Used for factory testing.

10.3 Single channel operation

Single channel operation is selected by writing a zero to the MULT bit in the A/D control and status register (ADCTL). This mode has four variations, which can be selected using the CONV8 and SCAN bits in the ADCTL register. In the first two variations, the CONV8 bit is clear and the single selected channel is converted four consecutive times. In the second two variations, the CONV8 bit is set and the single selected channel is converted eight consecutive times. The state of the SCAN bit determines whether continuous or single scanning is selected. The channel is selected by the CD – CA bits in the ADCTL register.

If channels eight and nine are selected, then the result registers previously used for two of the other channels become overwritten with channel eight and nine results.

10.3.1 4-conversion, single scan

Whichever port E or G pin is selected, the first result will be stored in the ADR5 result register and the fourth result will be stored in the ADR8 register. After the fourth conversion is complete all conversion activity is halted until a new conversion command is written to the ADCTL control register.

10.3.2 4-conversion, continuous scan

Conversions continue to be performed on the selected channel with the fifth conversion being stored in the ADR5 register (overwriting the first conversion result), the sixth conversion overwrites ADR6, the seventh overwrites ADR7, and so on continuously. Using this variation, the data in any result register is at most four conversion times old.

10.3.3 8-conversion, single scan

The result of the first conversion will be placed in result register ADR1, while the result of the eighth conversion will be placed in result register ADR8. After the eighth conversion is complete all conversion activity is halted until a new conversion command is written to the ADCTL control register.

10.3.4 8-conversion, continuous scan

Conversions continue to be performed on the selected channel with the ninth conversion being stored in the ADR1 register (overwriting the first conversion result), the tenth conversion overwrites ADR2, the eleventh overwrites ADR3, and so on continuously. Using this variation, the data in any result register is at most eight conversion times old.

10.4 Multiple channel operation

Multiple channel operation is selected by writing a one to the MULT bit in the A/D control and status register (ADCTL). This mode has four variations, which can be selected using the CONV8 and SCAN bits in the ADCTL register. In the first two variations, the CONV8 bit is clear and a group of four channels is selected. In this multiple channel mode, the group of channels selected depends only on the two most significant bits of the channel address (CD and CC). In the second two variations, the CONV8 bit is set and a group of eight channels is selected, depending on the state of the CD bit. The state of the SCAN bit determines whether continuous or single scanning is selected. Refer to Table 10-1.

10.4.1 4-channel single scan

A group of four channels is selected by the CD and CC bits in the ADCTL register (see Table 10-1). The first result is stored in the ADR5 result register, the second in ADR6, the third in ADR7 and the fourth in the ADR8 register. After the fourth conversion is complete, all conversion activity is halted until a new conversion command is written to the ADCTL control register.

10.4.2 4-channel continuous scan

Conversions continue to be performed on the selected group of channels with the fifth conversion being stored in the ADR5 register (replacing the earlier conversion result for the first channel in the group), the sixth conversion overwrites ADR6, the seventh overwrites ADR7, and so on, continuously. Using this second variation the data in any result register is, at most, four conversion times old.

10.4.3 8-channel single scan

When CONV8 is set and MULT is set, then a group of eight channels is converted. The group is selected by the CD bit. Each of the channels is converted and the result is placed in a separate result register. Port E pin 0 uses result register ADR1, Port E pin 1 uses result register ADR2 and so on. Each channel is converted once, then all conversion activity is halted until a new conversion command is written to the ADCTL control register.

10.4.4 8-channel continuous scan

Conversions continue to be performed on the eight channels with the ninth conversion being stored in the ADR1 register (replacing the earlier conversion result for the first channel in the group), the tenth conversion overwrites ADR2, the eleventh overwrites ADR3, and so on, continuously. Using this second variation the data in any result register is, at most, eight conversion times old.

10.5 Power-up and clock select

A/D power up is controlled by the ADPU bit in the OPTION register. When ADPU is cleared, power to the A/D system is removed. When ADPU is set, the A/D system is enabled. A delay of 100 microseconds is required after turning on the A/D converter, to allow the analog bias voltages to stabilize.

Clock select is controlled by the CSEL bit in the OPTION register. When CSEL is cleared, the A/D system uses the system E-clock. When CSEL is set, the A/D system uses an internal R-C clock

source, nominally 1.5 MHz, in which case the R-C internal clock should be selected. A delay of 10 milliseconds is required, after changing CSEL from zero to one, to allow the R-C oscillator to start and internal bias voltages to settle. If the E-clock frequency drops below 750kHz, then the internal R-C oscillator must be used.

When the A/D system is operating with the MCU E-clock, all switching and comparator operations are synchronized with the MCU clock. This allows the comparator results to be sampled at quiet clock times to minimise the effect of internal switching noise. As the internal R-C oscillator is asynchronous with respect to the MCU clock, internal switching noise is more likely to affect the overall accuracy of the A/D results, when using this oscillator, than when using the E-clock.

10.6 Operation in STOP and WAIT modes

If a conversion sequence is still in process when the MC68HC11KW1 enters the STOP or WAIT mode, the conversion of the current channel is suspended. When the MCU resumes normal operation, that channel is re-sampled and the conversion sequence resumes. As the MCU exits the WAIT mode, the A/D circuits are stable and valid results can be obtained on the first conversion. However, in STOP mode the comparator, charge pump and R-C oscillator are turned off. If the MC68HC11KW1 exits the STOP mode with a delay (as is normal), there will automatically be enough time for these circuits to stabilize before the first conversion. If the MC68HC11KW1 exits the STOP mode with no delay (DLY bit in OPTION register equal to zero) and a stable external clock supplied, the user must allow about 100 microseconds for the A/D circuitry to stabilize and to avoid invalid results.

10.7 Registers

10.7.1 ADCTL — A/D control and status register

	Address	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	State on reset
A/D control & status register (ADCTL)	$0030	CCF	CONV8	SCAN	MULT	CD	CC	CB	CA	undefined

This register can be read and written at any time. Note that a write to this register will always clear the CCF bit.

CCF — Conversions complete flag

This flag bit is set automatically after an A/D conversion cycle (four or eight conversions, depending on which conversion mode is selected). If a continuous scan mode is selected, the CCF flag will become set after the first time all four (or eight) registers have been updated, and it will remain set until the

ADCTL register is again written. Each time the ADCTL register is written, this bit is automatically cleared, any current conversion is aborted and a new conversion sequence is started.

CONV8 — Convert 8/convert 4 select bit

1 (set) – Convert 8 channels or one channel 8 times, (uses all 8 result registers).

0 (clear) – Convert 4 channels or one channel 4 times (uses 4 result registers).

SCAN — Continuous scan control

1 (set) – Convert continuously.

0 (clear) – Perform selected number of conversions (4 or 8) and stop.

MULT — Multiple channel/single channel control

1 (set) – Convert the channels in the selected group.

0 (clear) – Convert single channel selected.

CD, CC, CB, CA — Channel select bits

When 4-conversion (CONV8 = 0) and multiple channel (MULT=1) modes are selected, the CB and CA bits have no meaning or effect, and the CD and CC bits specify which of four groups of four channels are to be converted. When 8-conversion (CONV8 = 1) and multiple channel (MULT=1) modes are selected, the CC, CB and CA bits have no meaning or effect. Refer to Table 10-1 for a list of the A/D channel assignments.

10.7.2 ADFRQ — A/D converter frequency select register

	Address	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	State on reset
A/D frequency select register (ADFRQ)	$0032	0	0	0	0	0	0	0	ADER	0000 0000

This register can be read and written at any time.

Bits [7:1] — Not implemented; always read zero.

ADER — A/D frequency select

1 (set) – E/2 clock is used for A/D conversions.

0 (clear) – E clock is used for A/D conversions.

This bit improves the accuracy of conversion when the MC68HC11KW1 bus frequency is above 2.1MHz.

Note if the CSEK bit in the OPTION register is set, then the internal R-C oscillator (nominally 1.5MHz) is used for the A/D conversion and the ADER bit has no effect on the frequency.

10.7.3 ADR1 — ADR8 A/D result registers

	Address	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	State on reset
A/D result 1 (ADR1) high	$0040	(Bit 15)	(14)	(13)	(12)	(11)	(10)	(9)	(8)	undefined
A/D result 1 (ADR1) low	$0041	(7)	(6)	0	0	0	0	0	0	uu00 0000
A/D result 2 (ADR2) high	$0042	(Bit 15)	(14)	(13)	(12)	(11)	(10)	(9)	(8)	undefined
A/D result 2 (ADR2) low	$0043	(7)	(6)	0	0	0	0	0	0	uu00 0000
A/D result 3 (ADR3) high	$0044	(Bit 15)	(14)	(13)	(12)	(11)	(10)	(9)	(8)	undefined
A/D result 3 (ADR3) low	$0045	(7)	(6)	0	0	0	0	0	0	uu00 0000
A/D result 4 (ADR4) high	$0046	(Bit 15)	(14)	(13)	(12)	(11)	(10)	(9)	(8)	undefined
A/D result 4 (ADR4) low	$0047	(7)	(6)	0	0	0	0	0	0	uu00 0000
A/D result 5 (ADR5) high	$0048	(Bit 15)	(14)	(13)	(12)	(11)	(10)	(9)	(8)	undefined
A/D result 5 (ADR5) low	$0049	(7)	(6)	0	0	0	0	0	0	uu00 0000
A/D result 6 (ADR6) high	$004A	(Bit 15)	(14)	(13)	(12)	(11)	(10)	(9)	(8)	undefined
A/D result 6 (ADR6) low	$004B	(7)	(6)	0	0	0	0	0	0	uu00 0000
A/D result 7 (ADR7) high	$004C	(Bit 15)	(14)	(13)	(12)	(11)	(10)	(9)	(8)	undefined
A/D result 7 (ADR7) low	$004D	(7)	(6)	0	0	0	0	0	0	uu00 0000
A/D result 8 (ADR8) high	$004E	(Bit 15)	(14)	(13)	(12)	(11)	(10)	(9)	(8)	undefined
A/D result 8 (ADR8) low	$004F	(7)	(6)	0	0	0	0	0	0	uu00 0000

The eight 10-bit result registers are read-only; they can be read at any time, but a write will have no meaning or effect. In each result register, the eight high order bits are in one address location and the remaining two low order bits are in bit locations 6 and 7 of the following address. The six unused bits will always read as zeros. This allows a double byte read to be performed without having to adjust the result.

A
ELECTRICAL SPECIFICATIONS

This section contains the electrical specifications and associated timing information for the standard supply voltage (V_{DD} = 5V ± 5%) MC68HC11KW1.

A.1 Maximum ratings

Rating	Symbol	Value	Unit
Supply voltage (1)	V_{DD}	– 0.3 to +7.0	V
Input voltage (1)	V_{IN}	– 0.3 to +7.0	V
Operating temperature range	T_A	T_L to T_H –40 to +85	°C
Storage temperature range	T_{STG}	– 50 to +150	°C
Current drain per pin (2) – not VDD, VSS, VDDAD, VSSAD, VRH, VRL or ports H, K,	I_D	25	mA

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

(2) Maximum current drain per pin is for one pin at a time, observing maximum power dissipation limits.

Note: 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}.

A.2 Thermal characteristics and power considerations

The average chip junction temperature, T_J, in degrees Celsius 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 = Total power dissipation = $P_{INT} + P_{I/O}$ (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:

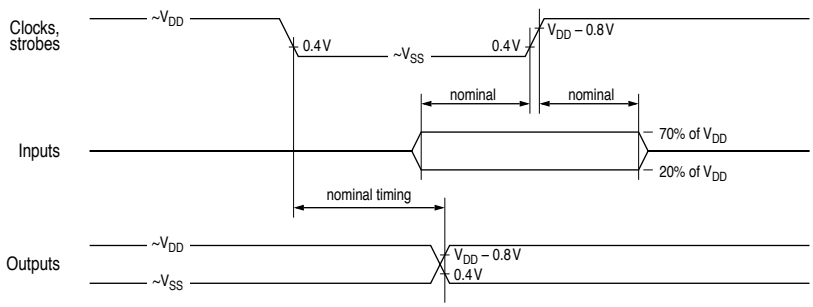
$$K = P_D \bullet (T_A + 273) + \theta_{JA} \bullet P_D^2 \quad [3]$$

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 below:

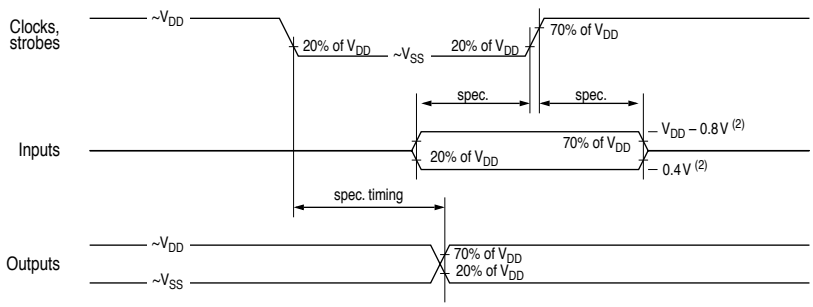
Characteristics	Symbol	Value	Unit
Thermal resistance – 100-pin TQFP package	θ_{JA}	54°	°C/W

A

A.3 Test methods



(b) DC testing



(c) AC testing

Notes:

(1) Full test loads are applied during all DC electrical tests and AC timing measurements.

(2) During AC timing measurements, inputs are driven to 0.4 V and V_{DD} – 0.8 V; timing measurements are taken at the 20% and 70% of V_{DD} points.

Figure A-1 Test methods

A

A.4 DC electrical characteristics

(V_{DD} = 5.0Vdc ± 5%, V_{SS} = 0Vdc, $T_A = T_L$ to T_H, unless otherwise noted)

Characteristic	Symbol	Min.	Max.	Unit
Output voltage[1] (I_{LOAD} = ± 10 µA)				
All outputs except XTAL	V_{OL}	–	0.1	V
All outputs except XTAL, $\overline{RESET}$ & MODA	V_{OH}	V_{DD} – 0.1	–	V
Output high voltage[1] (I_{LOAD} = –0.8mA, V_{DD} =4.5V)				
All outputs except XTAL, $\overline{RESET}$ & MODA	V_{OH}	V_{DD} – 0.8	–	V
Output low voltage (I_{LOAD} = +1.6mA)				
All outputs except XTAL	V_{OL}	–	0.4	V
Input high voltage	V_{IH}			V
All inputs except $\overline{RESET}$		0.7V_{DD}	V_{DD} + 0.3	
$\overline{RESET}$		0.8V_{DD}	V_{DD} + 0.3	
Input low voltage – all inputs	V_{IL}	V_{SS} – 0.3	0.2V_{DD}	V
I/O ports three-state leakage ($V_{IN} = V_{IH}$ or V_{IL})	I_{OZ}	–	±10	µA
Input leakage[2]	I_{IN}			
($V_{IN} = V_{DD}$ or V_{SS}) $\overline{IRQ}$, $\overline{XIRQ}$		–	±1	µA
($V_{IN} = V_{DD}$ or V_{SS}) MODB/V_{STBY}		–	±10	µA
Input current with pull-up resistors ($V_{IN} = V_{IL}$)				
Ports B, F, H, Port G[5:0]	I_{IPR}	100	500	µA
RAM stand-by voltage (power down)	V_{SB}	2.0	V_{DD}	V
RAM stand-by current (power down)	I_{SB}	–	20	µA
Input capacitance	C_{in}			
PE[7:0], PG[7,6], $\overline{IRQ}$, $\overline{XIRQ}$, EXTAL		–	8	pF
Ports A, B, C, D, F, G[5:0], H, J, K, MODA/$\overline{LIR}$, $\overline{RESET}$		–	12	pF
Output load capacitance	C_L			pF
All outputs except PD[4:1], XOUT, XTAL, MODA/LIR		–	90	
XOUT		–	30	
PD[4:1]		–	200	
Maximum total supply current[3]:				
RUN:	I_{DD}			
Single chip mode		–	40	mA
Expanded mode		–	50	mA
WAIT: (All peripheral functions shut down)	W_{IDD}			
Single chip mode		–	25	mA
Expanded mode		–	30	mA
STOP:	S_{IDD}			
Single chip mode, no clocks		–	100	µA
Maximum power dissipation	P_D			
Single chip mode		–	210	mW
Expanded mode		–	262.5	mW

(1) V_{OH} specification for $\overline{RESET}$ and MODA is not applicable as they are open-drain pins.

(2) Refer to A/D specification for the leakage current on A/D inputs.

(3) EXTAL is driven with a square wave. No dc loads. Expansion bus active. All other ports configured as inputs; bus frequency is f_0 = 4.0MHz.

A

A.5 Control timing

(V_{DD} = 5.0V ± 5%, V_{SS} = 0Vdc, $T_A = T_L$ to T_H unless otherwise noted)

Characteristic (1)	Symbol	4.0 MHz		Unit
		Min.	Max.	
Frequency of operation	f_{OP}	0	4.0	MHz
E clock period	t_{CYC}	250	—	ns
Crystal frequency	f_{XTAL}	—	16.0	MHz
External oscillator frequency	$4f_{OP}$	0	16.0	MHz
Processor control setup time (t_{PCSU} = 1/4 t_{CYC} + 50 ns)	t_{PCSU}	112	–	ns
Reset input pulse width (2)	PW_{RSTL} (3) PW_{RSTL} (4)	16 1	— —	t_{CYC}
Mode programming set-up time	t_{MPS}	2	—	t_{CYC}
Mode programming hold time	t_{MPH}	10	—	ns
Interrupt pulse width ($\overline{IRQ}$ edge sensitive mode)	PW_{IRQ}	270	—	ns
Interrupt pulse period(5)	t_{ILIH}	Note 5	—	t_{CYC}
WAIT recovery start-up time	t_{WRS}	—	4	t_{CYC}
Timer pulse width, input capture pulse accumulator input PW_{TIM} = t_{CYC} + 20 ns	PW_{TIM}	270	—	ns

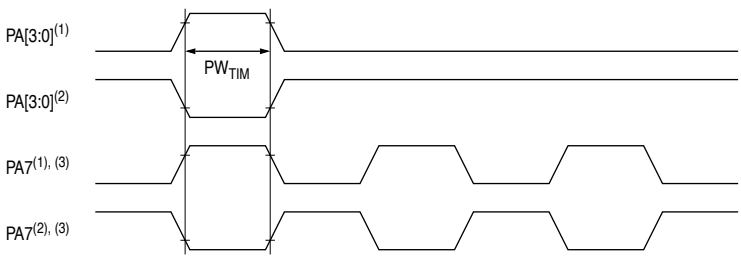
(1) All timing is given with respect to 20% and 70% of V_{DD}, unless otherwise noted.

(2) Reset is recognized during the first clock cycle it is held low. Internal circuitry then drives the pin low for eight clock cycles, releases the pin and samples the pin level four cycles later to determine the source of the interrupt. (See Section 5.)

(3) To guarantee an external reset vector.

(4) This is the minimum input time; it can be pre-empted by an internal reset.

(5) The minimum period t_{ILIH} should not be less than the number of cycles it takes to execute the interrupt service plus 21 t_{CYC}.



Notes
(1) Rising edge sensitive input.
(2) Falling edge sensitive input.
(3) Maximum pulse accumulator clocking rate is E clock frequency divided by two (E/2).

Figure A-2 Timer inputs

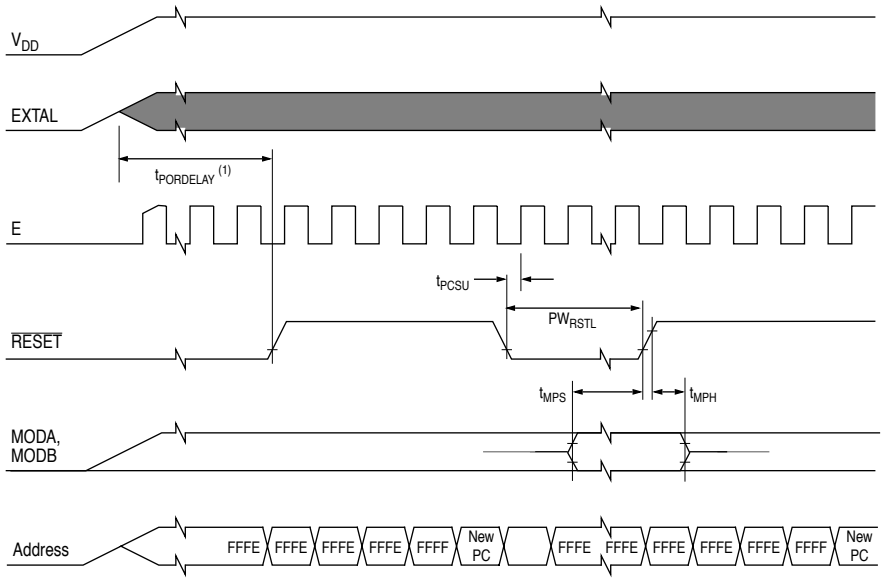


Figure A-3 Reset timing

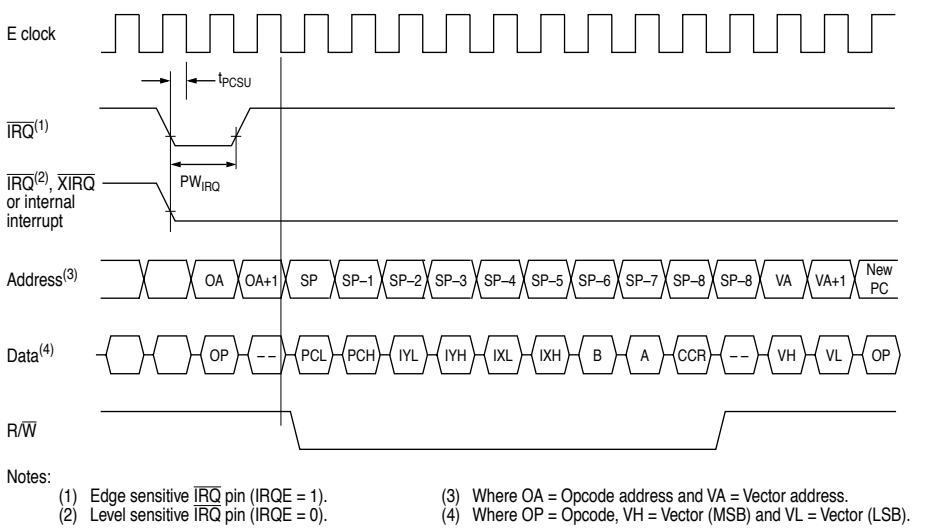
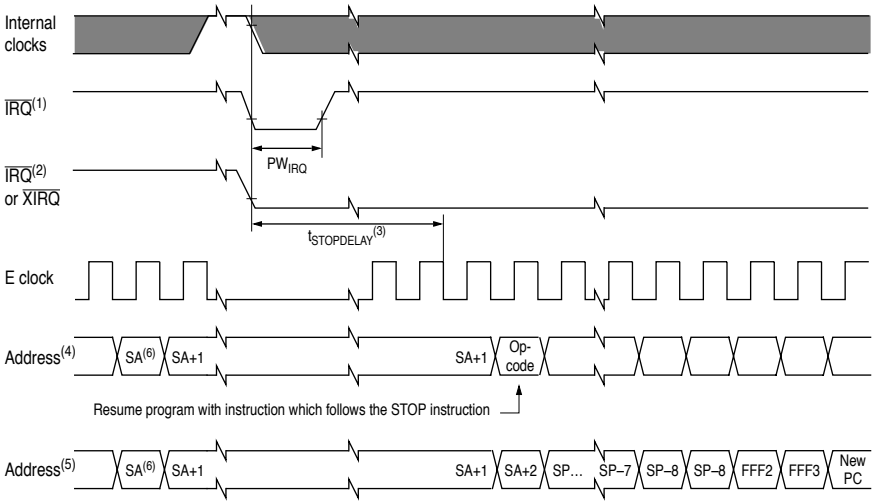
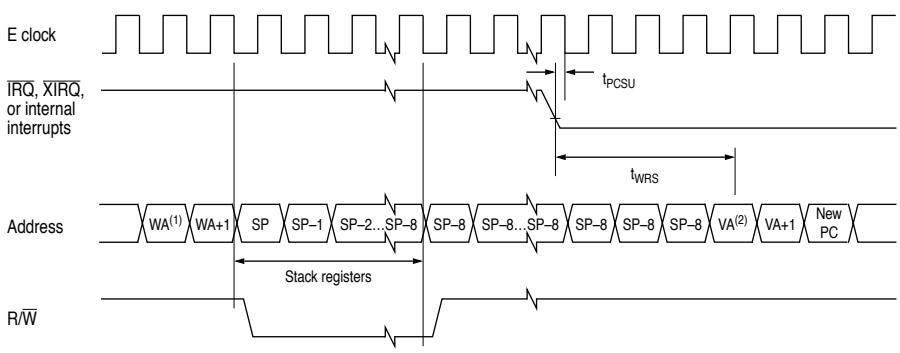


Figure A-4 Interrupt timing



Notes: (1) Edge sensitive $\overline{IRQ}$ pin (IRQE = 1).
(2) Level sensitive $\overline{IRQ}$ pin (IRQE = 0).
(3) If DLY = 1: $t_{STOPDELAY} = 4064\ t_{CYC}$
If DLY = 0: $t_{STOPDELAY} = 4\ t_{CYC}$
(4) $\overline{XIRQ}$ with X-bit in CCR = 1.
(5) $\overline{IRQ}$ (or $\overline{XIRQ}$, with X-bit = 0; in this case vector fetch will be $FFF4/5).
(6) SA = STOP address.

Figure A-5 STOP recovery timing



Notes:
$\overline{RESET}$ also causes recovery from WAIT. (1) WA = WAIT address. (2) VA = Vector address.

Figure A-6 WAIT recovery timing

A

A.5.1 Peripheral port timing

(V_{DD} = 5.0Vdc ± 5%, V_{SS} = 0Vdc, $T_A = T_L$ to T_H)

Characteristic (1)	Symbol	4.0 MHz		Unit
		Min.	Max.	
Frequency of operation (E clock frequency)	f_{OP}	0	4.0	MHz
E clock period	t_{CYC}	250	—	ns
Peripheral data set-up time, all ports (2)	t_{PDSU}	100	—	ns
Peripheral data hold time, all ports (2)	t_{PDH}	50	—	ns
Delay time, peripheral data write	t_{PWD}			
MCU write to port A, B, G[5:0], H, J or K		—	200	ns
MCU write to port C, D or F ($t_{PWD} = t_{CYC}/4 + 100$ ns)		—	162	ns

(1) All timing is given with respect to 20% and 70% of V_{DD}, unless otherwise noted.

(2) Port C, and D timing is valid for active drive (CWOM, DWOM, and WOMS bits clear).

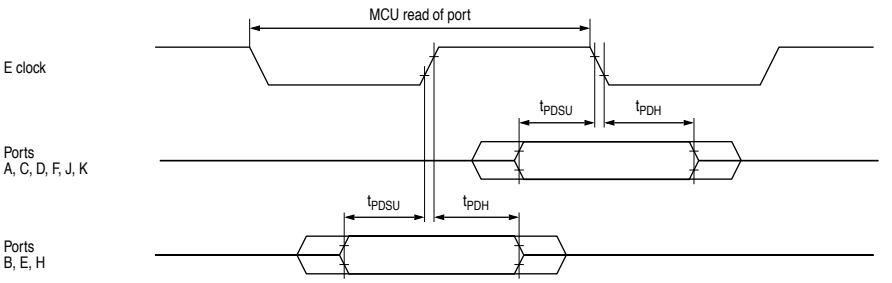


Figure A-7 Port read timing diagram

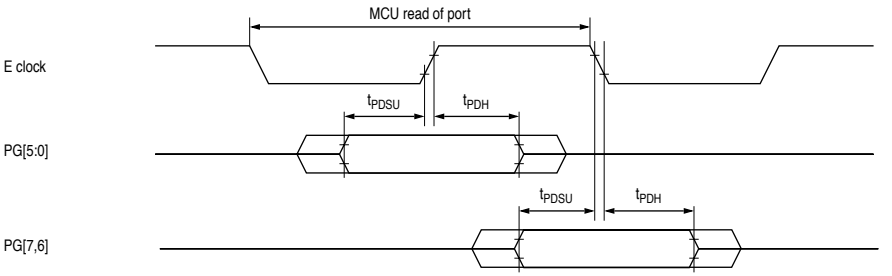


Figure A-8 Port G control timing

A

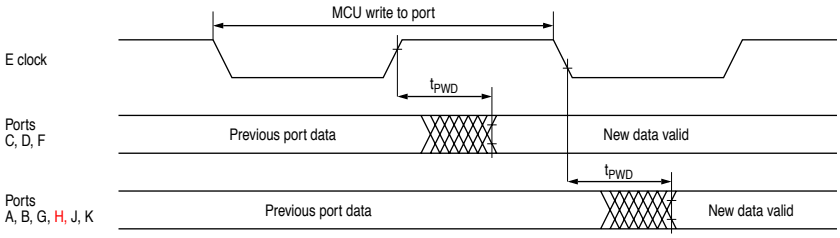


Figure A-9 Port write timing diagram

A

A.5.2 Analog-to-digital converter characteristics

(V_{DD} = 5.0V ± 5%, V_{SS} = 0V, V_{RH} = V_{DD}, V_{RL} = V_{SS}, T_A = T_L to T_H, 750kHz ≤ E ≤ 4MHz, unless otherwise noted(1))

Characteristic	Parameter	Min.	Absolute	Max.	Unit
Resolution	Number of bits resolved by ADC	—	10	—	bits
Non-linearity(2)	Maximum deviation from the best fitting ADC transfer characteristics	—	—	± 1.0	LSB
Offset error(2)	Maximum offset from ADC transfer characteristics	—	—	± 1.0	LSB
Total unadjusted error	Maximum deviation from the ideal ADC transfer characteristics	—	—	± 1.5	LSB
Quantization error	Uncertainty due to converter resolution	—	—	± 0.5	LSB
Absolute accuracy	Difference between the actual input voltage and the full-scale weighted equivalent of the binary output code, including all error sources	—	—	± 2.0	LSB
Conversion range	Analog input voltage range	V_{RL}	—	V_{RH}	V
Analog input voltage(3)	Maximum and minimum analog input voltage	V_{RL} – 0.3	—	1.125V_{RH}	V
V_{RH}	Maximum analog reference voltage	V_{RL}	—	V_{DD} + 0.1	V
V_{RL}	Minimum analog reference voltage	V_{SS} – 0.1	—	V_{RH}	V
ΔV_R	Minimum difference between V_{RH} and V_{RL}	4.5	—	—	V
Monotinicity	Conversion result never decreases with an increase in input voltage and has no missing codes		Guaranteed		
Zero input reading	Conversion result when V_{in} = V_{RL}	0000	—	—	Hex
Full scale reading	Conversion result when V_{in} = V_{RH}	—	—	FFC0	Hex
Input leakage(4)	Input leakage on A/D pins: PE[7:0], PG[7,6]	—	—	400	nA
Input leakage V_{rh} to V_{rl}	V_{rh} to V_{rl} divider chain current consumption			300	µA

(1) In order to attain these accuracies with the ECK > 2.1MHz, the ADER bit in ADFRQ must be set. Input clock duty cycles other than 50% will affect the A/D accuracies

(2) Non-linearity and offset error are characterised using the process window parameters affecting the ADC accuracy, but they are not tested

(3) Minimum analogue input voltage should not go below V_{RL}-0.3V
Maximum analogue input voltage should not exceed 1.125V_{rh}

(4) Source impedance should equal approximately 1 KΩ. Source impedances greater than 1 KΩ affect accuracy adversely because of input leakage.

A

A.5.3 Serial peripheral interface timing

(V_{DD} = 5.0Vdc ± 5%, V_{SS} = 0Vdc, $T_A = T_L$ to T_H)

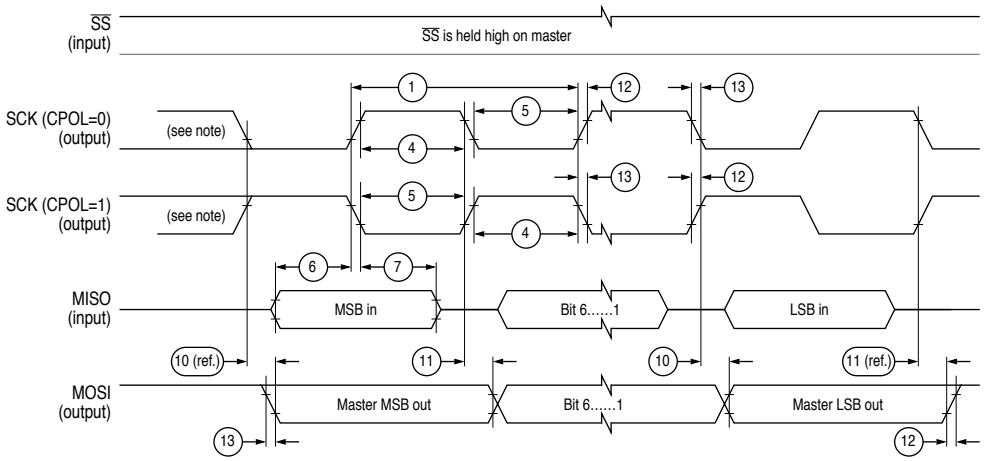
Num	Characteristic (1)		Symbol	4.0 MHz		Unit
				Min.	Max.	
	Operating frequency	Master	$f_{OP(M)}$	0	0.5	f_{OP}
		Slave	$f_{OP(S)}$	0	4.0	MHz
1	Cycle time	Master	$t_{CYC(M)}$	2.0	—	t_{CYC}
		Slave	$t_{CYC(S)}$	250	—	ns
2	Enable lead time (2)	Master	$t_{LEAD(M)}$	—	—	ns
		Slave	$t_{LEAD(S)}$	200	—	ns
3	Enable lag time (2)	Master	$t_{LAG(M)}$	—	—	ns
		Slave	$t_{LAG(S)}$	200	—	ns
4	Clock (SCK) high time	Master	$t_{W(SCKH)M}$	130	—	ns
		Slave	$t_{W(SCKH)S}$	85	—	ns
5	Clock (SCK) low time	Master	$t_{W(SCKL)M}$	130	—	ns
		Slave	$t_{W(SCKL)S}$	85	—	ns
6	Input data set-up time	Master	$t_{SU(M)}$	100	—	ns
		Slave	$t_{SU(S)}$	100	—	ns
7	Input data hold time	Master	$t_{H(M)}$	100	—	ns
		Slave	$t_{H(S)}$	100	—	ns
8	Access time (from high-z to data active)	Slave	t_A	0	120	ns
9	Disable time (hold time to high-z state)	Slave	t_{DIS}	—	125	ns
10	Data valid (after enable edge) (3)		$t_{V(S)}$	—	125	ns
11	Output data hold time (after enable edge)		t_{HO}	0	—	ns
12	Rise time (3)					
	SPI outputs (SCK, MOSI and MISO)		t_{RM}	—	100	ns
	SPI inputs (SCK, MOSI, MISO and $\overline{SS}$)		t_{RS}	—	2.0	µs
13	Fall time (3)					
	SPI outputs (SCK, MOSI and MISO)		t_{FM}	—	100	ns
	SPI inputs (SCK, MOSI, MISO and $\overline{SS}$)		t_{FS}	—	2.0	µs

(1) All timing is given with respect to 20% and 70% of V_{DD}, unless otherwise noted.

(2) Signal production depends on software.

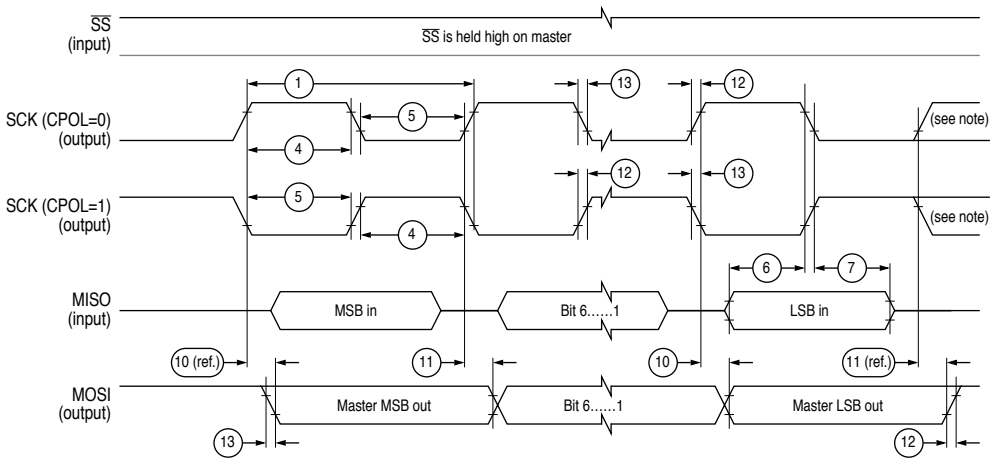
(3) Assumes 200pF load on all SPI pins.

A



Note: This first clock edge is generated internally, but is not seen at the SCK pin.

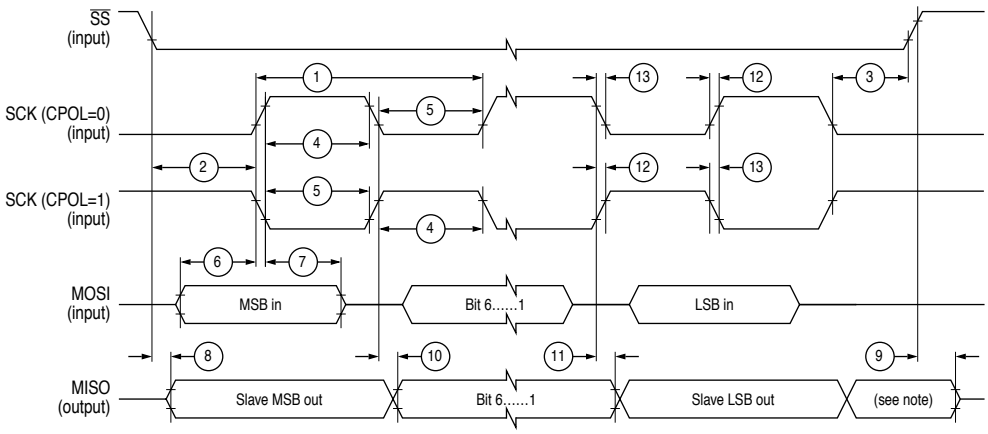
Figure A-10 SPI master timing (CPHA = 0)



Note: This last clock edge is generated internally, but is not seen at the SCK pin.

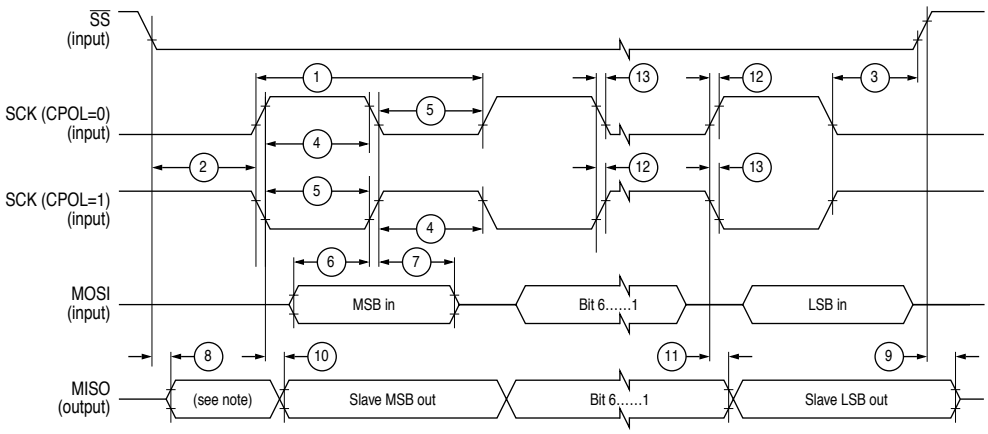
Figure A-11 SPI master timing (CPHA = 1)

A



Note: Not defined, but normally the MSB of character just received.

Figure A-12 SPI slave timing (CPHA = 0)



Note: Not defined, but normally the LSB of character last transmitted.

Figure A-13 SPI slave timing (CPHA = 1)

A

A.5.4 Non-multiplexed expansion bus timing

(V_{DD} = 5.0Vdc ± 5%, V_{SS} = 0Vdc, $T_A = T_L$ to T_H)

Num	Characteristic [1]	Symbol	4.0 MHz		Unit
			Min.	Max.	
	Frequency of operation (E clock frequency)	f_{OP}	0	4.0	MHz
1	E clock period	t_{CYC}	250	—	ns
2	Pulse width, E low [2], [3]	PW_{EL}	105	—	ns
3	Pulse width, E high [2], [3], [4]	PW_{EH}	100	—	ns
4A	E clock rise time	t_R	—	20	ns
4B	E clock fall time	t_F	—	15	ns
9	Address hold time [3]	t_{AH}	21	—	ns
11	Address delay time [3]	t_{AD}	—	71	ns
12	Address valid to E rise time [3]	t_{AV}	34	—	ns
17	Read data set-up time	t_{DSR}	20	—	ns
18	Read data hold time	t_{DHR}	0	—	ns
19	Write data delay time	t_{DDW}	—	40	ns
21	Write data hold time [3]	t_{DHW}	31	—	ns
29	MPU address access time [3], [4]	t_{ACCA}	144	—	ns
39	Write data set-up time [3], [4]	t_{DSW}	60	—	ns
50	E valid chip select delay time	t_{ECSD}	—	45	ns
51	E valid chip select access time [4]	t_{ECSA}	40	—	ns
52	Chip select hold time	t_{CH}	0	25	ns
54	Address valid chip select delay time	t_{ACSD}	—	103	ns
55	Address valid chip select access time [4]	t_{ACSA}	113	—	ns
56	Address valid to chip select time	t_{AVCS}	10	—	ns
57	Address valid to data three-state time	t_{AVDZ}	—	10	ns

(1) All timing is given with respect to 20% and 70% of V_{DD}, unless otherwise noted.

(2) Input clock duty cycles other than 50% will affect the bus performance.

(3) For $f_{OP} \leq 2$MHz the following formulae may be used to calculate parameter values:

$PW_{EL} = t_{CYC}/2 - 20$ns — $PW_{EH} = t_{CYC}/2 - 25$ns

$t_{AH} = t_{CYC}/8 - 10$ns — $t_{AD} = t_{CYC}/8 + 40$ns

$t_{AV} = PW_{EL} - t_{AD}$ — $t_{DHW} = t_{CYC}/8$

$t_{ACCA} = t_{CYC} - t_F - t_{DSR} - t_{AD}$ — $t_{DSW} = PW_{EH} - t_{DDW}$

$t_{ECSA} = PW_{EH} - t_{ECSD} - t_{DSR}$ — $t_{ACSA} = t_{CYC} - t_F - t_{DSR} - t_{ACSD}$

(4) Indicates a parameter affected by clock stretching. Add n(t_{CYC}) to the parameter value, where n = 1, 2 or 3, depending on the valued written to the CSCSTR register.

A

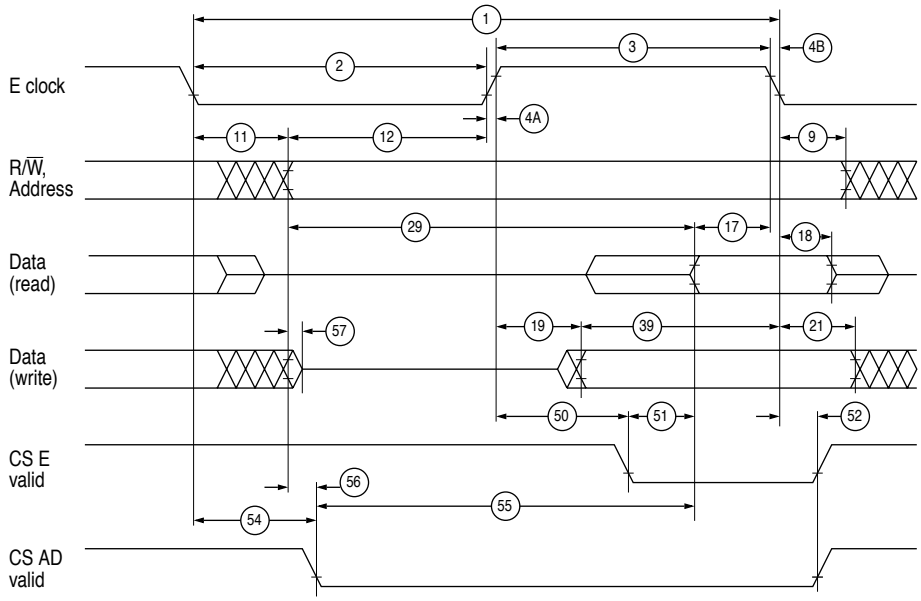


Figure A-14 Expansion bus timing

A.6 EEPROM characteristics

Characteristic	Temperature range −40 to +85°C	Unit
Programming time, t_{EEPROG} [1]		
<1 MHz, RCO enabled	10	ms
1–2 MHz, RCO disabled	20	
≥2 MHz & whenever RCO enabled	10	
Erase time: byte, row and bulk [1]	10	ms
Write/erase endurance [2]	10000	cycles
Data retention [2]	10	years

(1) The RC oscillator (RCO) must be enabled (by setting the CSEL bit in the OPTION register) for EEPROM programming and erasure when the E clock frequency is less than 1.0 MHz.

(2) Refer to the current issue of Motorola's quarterly Reliability Monitor Report for the latest failure rate information.

A

THIS PAGE LEFT BLANK INTENTIONALLY

B
MECHANICAL DATA

B.1 Packaging

The MC68HC11KW1 is available packaged in a 100-pin thin quad flat pack (TQFP).

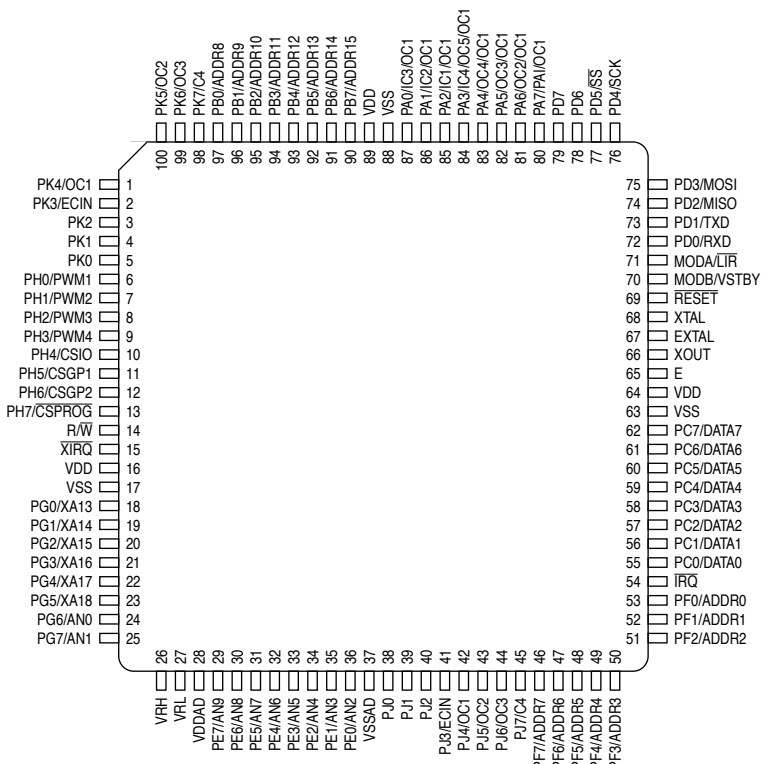
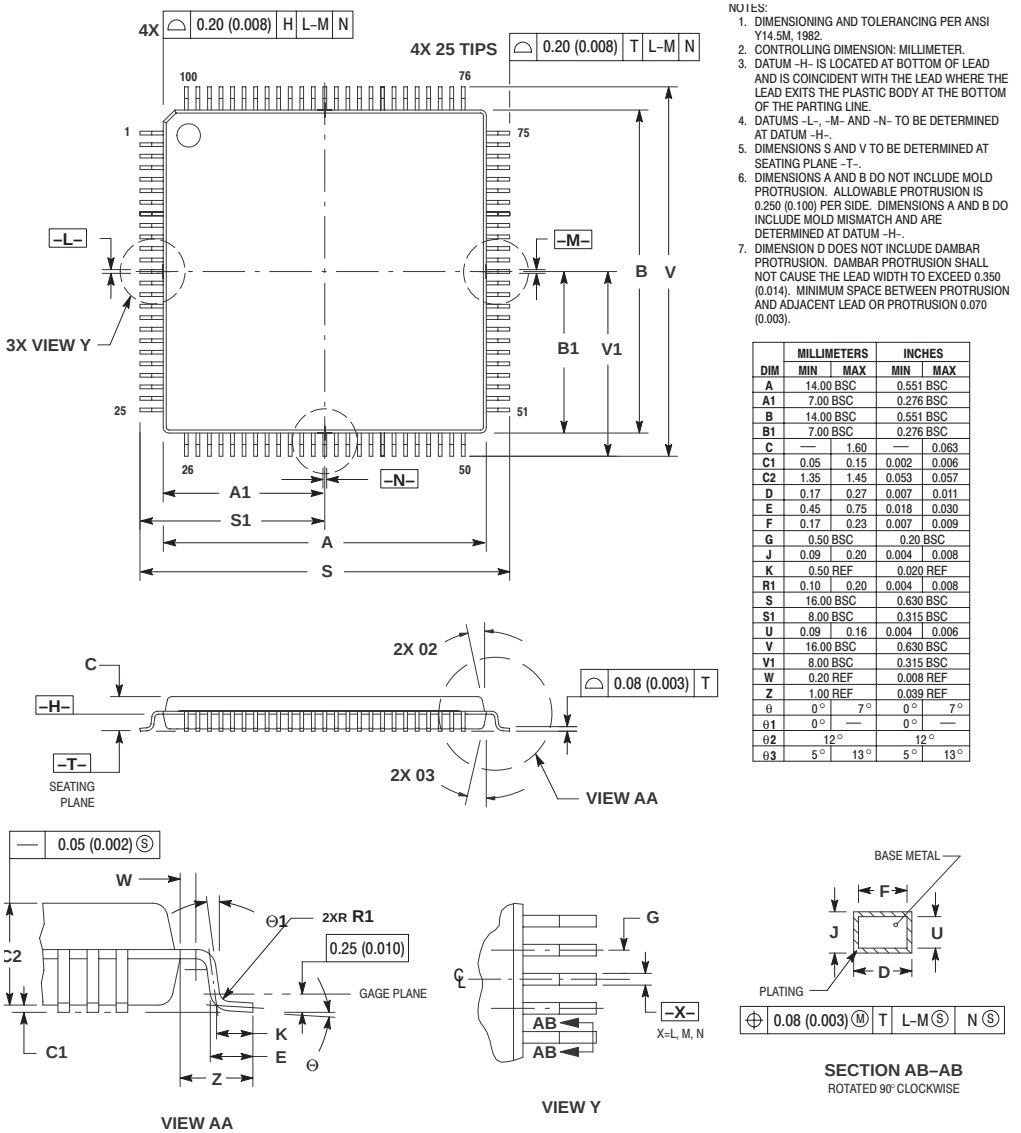


Figure B-1 100-pin TQFP



NOTES:
1. DIMENSIONING AND TOLERANCING PER ANSI Y14.5M, 1982.
2. CONTROLLING DIMENSION: MILLIMETER.
3. DATUM –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.
4. DATUMS –L–, –M– AND –N– TO BE DETERMINED AT DATUM –H–.
5. DIMENSIONS S AND V TO BE DETERMINED AT SEATING PLANE –T–.
6. DIMENSIONS A AND B DO NOT INCLUDE MOLD PROTRUSION. ALLOWABLE PROTRUSION IS 0.250 (0.100) PER SIDE. DIMENSIONS A AND B DO INCLUDE MOLD MISMATCH AND ARE DETERMINED AT DATUM –H–.
7. DIMENSION D DOES NOT INCLUDE DAMBAR PROTRUSION. DAMBAR PROTRUSION SHALL NOT CAUSE THE LEAD WIDTH TO EXCEED 0.350 (0.014). MINIMUM SPACE BETWEEN PROTRUSION AND ADJACENT LEAD OR PROTRUSION 0.070 (0.003).

	MILLIMETERS		INCHES	
DIM	MIN	MAX	MIN	MAX
A	14.00 BSC		0.551 BSC	
A1	7.00 BSC		0.276 BSC	
B	14.00 BSC		0.551 BSC	
B1	7.00 BSC		0.276 BSC	
C	—	1.60	—	0.063
C1	0.05	0.15	0.002	0.006
C2	1.35	1.45	0.053	0.057
D	0.17	0.27	0.007	0.011
E	0.45	0.75	0.018	0.030
F	0.17	0.23	0.007	0.009
G	0.50 BSC		0.20 BSC	
J	0.09	0.20	0.004	0.008
K	0.50 REF		0.020 REF	
R1	0.10	0.20	0.004	0.008
S	16.00 BSC		0.630 BSC	
S1	8.00 BSC		0.315 BSC	
U	0.09	0.16	0.004	0.006
V	16.00 BSC		0.630 BSC	
V1	8.00 BSC		0.315 BSC	
W	0.20 REF		0.008 REF	
Z	1.00 REF		0.039 REF	
θ	0°	7°	0°	7°
θ1	0°	—	0°	—
θ2	12°		12°	
θ3	5°	13°	5°	13°

Case 983-01

Figure B-2 100-pin TQFP mechanical dimensions

C
DEVELOPMENT SYSTEMS

The following information provides a reference to development tools for the M68HC11 family of microcontrollers. For more detailed information please refer to the appropriate system manual.

Table C-1 M68HC11 development tools

Devices	Evaluation boards	Evaluation modules	Evaluation systems/kits	Programmer boards
MC68HC11KW1	–	M68EM11KW1	–	M68SPGMR11

Note: Target cables for the evaluation module should be ordered separately.

C.1 EVS — Evaluation system

The EVS is an economical tool for designing, debugging and evaluating target systems based on the MC68HC11KW1 device. The two printed circuit boards that comprise the EVS are the M68EM11KW1 emulator module and the M68PFB11KIT platform board. The main features of the EVS are as follows:

- Monitor/debugger firmware
- Single-line assembler/disassembler
- Host computer download capability
- Dual memory maps:
  - 64K byte monitor map that includes 16K bytes of monitor EPROM
- MCU extension I/O port for single chip, expanded and special test operating modes
- RS-232C terminal and host I/O ports
- Logic analyser connector

C.2 MMDS11 — Motorola modular development system

The MMDS11 is an emulator system that provides a bus state analyser and real-time memory windows. The unit's integrated design environment includes an editor, an assembler, user interface and source-level debug. A complete MMDS11 consists of:

- A station module — the metal MMDS11 enclosure, containing the control board and the internal power supply. Most system cables connect to the MMDS11 station module. (The cable to an optional target system, however, runs through an aperture in the station module enclosure to connect directly to the emulator module).
- An emulator module (EM) — such as the EM11KW1: a printed circuit board that enables system functionality for a specific set of MCUs. The EM fits into the station module through a sliding panel in the enclosure top. The EM has a connector for the target cable.
- Two logic clip cable assemblies — twisted pair cables that connect the station module to your target system, a test fixture, a clock or any other circuitry useful for evaluation or analysis. One end of each cable assembly has a moulded connector, which fits into station module pod A or pod B. Leads at the other end of the cable terminate in female probe tips. Ball clips come with the cables.
- A 9-lead RS-232 serial cable — the cable that connects the station module to the host computer's RS-232 port.

C.3 SPGMR11 — Serial peripheral system

The SPGMR11 is an economical tool for programming M68HC11 MCUs. The system consists of the M68SPGMR11 unit and a programming module which adapts the SPGMR11 to the appropriate MCU and package type. For the MC68HC11KW1, the programming module can be ordered as M68PA11KW1PU100 for the 100-pin packaged device.

GLOSSARY

This section contains abbreviations and specialist words used in this data sheet and throughout the industry. Further information on many of the terms may be gleaned from Motorola's ***M68HC11 Reference Manual, M68HC11RM/AD***, or from a variety of standard electronics text books.

$xxxx	The digits following the '$' are in hexadecimal format.
%xxxx	The digits following the '%' are in binary format.
A/D, **ADC**	Analog-to-digital (converter).
Bootstrap mode	In this mode the device automatically loads its internal memory from an external source on reset and then allows this program to be executed.
Byte	Eight bits.
CCR	Condition codes register; an integral part of the CPU.
CERQUAD	A ceramic package type, principally used for EPROM and high temperature devices.
Clear	'0' — the logic zero state; the opposite of 'set'.
CMOS	Complementary metal oxide semiconductor. A semiconductor technology chosen for its low power consumption and good noise immunity.
COP	Computer operating properly. *aka* 'watchdog'. This circuit is used to detect device runaway and provide a means for restoring correct operation.
CPU	Central processing unit.
D/A, DAC	Digital-to-analog (converter).
EEPROM	Electrically erasable programmable read only memory. *aka* 'EEROM'.
EPROM	Erasable programmable read only memory. This type of memory requires exposure to ultra-violet wavelengths in order to erase previous data. *aka* 'PROM'.
ESD	Electrostatic discharge.
Expanded mode	In this mode the internal address and data bus lines are connected to external pins. This enables the device to be used in much more complex systems, where there is a need for external memory for example.

EVS	Evaluation system. One of the range of platforms provided by Motorola for evaluation and emulation of their devices.
HCMOS	High-density complementary metal oxide semiconductor. A semiconductor technology chosen for its low power consumption and good noise immunity.
I/O	Input/output; used to describe a bidirectional pin or function.
Input capture	(IC) This is a function provided by the timing system, whereby an external event is 'captured' by storing the value of a counter at the instant the event is detected.
Interrupt	This refers to an asynchronous external event and the handling of it by the MCU. The external event is detected by the MCU and causes a predetermined action to occur.
$\overline{\text{IRQ}}$	Interrupt request. The overline indicates that this is an active-low signal format.
K byte	A kilo-byte (of memory); 1024 bytes.
LCD	Liquid crystal display.
LSB	Least significant byte.
M68HC11	Motorola's family of advanced 8-bit MCUs.
MCU	Microcontroller unit.
MSB	Most significant byte.
Nibble	Half a byte; four bits.
NRZ	Non-return to zero.
Opcode	The opcode is a byte which identifies the particular instruction and operating mode to the CPU. See also: prebyte, operand.
Operand	The operand is a byte containing information the CPU needs to execute a particular instruction. There may be from 0 to 3 operands associated with an opcode. See also: opcode, prebyte.
Output compare	(OC) This is a function provided by the timing system, whereby an external event is generated when an internal counter value matches a predefined value.
PLCC	Plastic leaded chip carrier package.
PLL	Phase-locked loop circuit. This provides a method of frequency multiplication, to enable the use of a low frequency crystal in a high frequency circuit.
Prebyte	This byte is sometimes required to qualify an opcode, in order to fully specify a particular instruction. See also: opcode, operand.
Pull-down, pull-up	These terms refer to resistors, sometimes internal to the device, which are permanently connected to either ground or V_{DD}.

PWM	Pulse width modulation. This term is used to describe a technique where the width of the high and low periods of a waveform is varied, usually to enable a representation of an analog value.
QFP	Quad flat pack package.
RAM	Random access memory. Fast read and write, but contents are lost when the power is removed.
RFI	Radio frequency interference.
RTI	Real-time interrupt.
ROM	Read-only memory. This type of memory is programmed during device manufacture and cannot subsequently be altered.
RS-232C	A standard serial communications protocol.
SAR	Successive approximation register.
SCI	Serial communications interface.
Set	'1' — the logic one state; the opposite of 'clear'.
Silicon glen	An area in the central belt of Scotland, so called because of the concentration of semiconductor manufacturers and users found there.
Single chip mode	In this mode the device functions as a self contained unit, requiring only I/O devices to complete a system.
SPI	Serial peripheral interface.
Test mode	This mode is intended for factory testing.
TTL	Transistor-transistor logic.
UART	Universal asynchronous receiver transmitter.
VCO	Voltage controlled oscillator.
Watchdog	*see* 'COP'.
Wired-OR	A means of connecting outputs together such that the resulting composite output state is the logical OR of the state of the individual outputs.
Word	Two bytes; 16 bits.
$\overline{\text{XIRQ}}$	Non-maskable interrupt request. The overline indicates that this has an active-low signal format.

THIS PAGE LEFT BLANK INTENTIONALLY

INDEX

In this index numeric entries are placed first; page references in *italics* indicate that the reference is to a figure.

16-bit PWM 9-38

A

A/D 10-1
- accuracy of conversion 6-6
- ADCTL — A/D control and status register 10-6
- ADFRQ — A/D converter frequency select register 10-7
- ADR1—ADR8 — A/D results registers 10-8
- pins 10-1
- reset 5-9

accumulators 3-2
ADCTL — A/D control and status register 10-6
addressing modes 3-7
address-mark wakeup 7-4
ADER - bit in ADFRQ 10-7
ADFRQ — A/D converter frequency select register 10-7
ADR1—ADR8 — A/D results registers 10-8

B

baud rates
- bootloader 4-2
- SCI 7-6

block diagrams
- MC68HC11KW1 1-3
- pulse accumulator *9-34*
- PWM *9-39*
- SCI *7-3*
- SCI baud rate *7-1*
- SPI *8-2*
- timer *9-17*, *9-25*
- timer clock divider chains *9-2*
- timers *9-5*

bootloader 4-2, 4-4
boundary conditions, PWM 9-44
BPPUE - bit in PPAR 6-13
BPROT — Block protect reg. 4-18
BPRT[5:0] - bits in BPROT 4-19
BSPL - bit in SCBDH 7-6
BTST - bit in SCBDH 7-6
BULKP - bit in BPROT 4-19
bypassing 2-4
BYTE - bit in PPROG 4-41

C

C4F - bit in T2FLG 9-23
C4F - bit in T3FLG 9-30
C4I - bit in T2MSK 9-22
C4I - bit in T3MSK 9-29
C-bit in CCR 3-5
CCF - bit in ADCTL 10-6
CCR — condition code reg. 3-4
CD—CA - bits in ADCTL 10-7
CFORC — Timer compare force register 9-9
Chip selects
- CSTL — Chip select control register 4-34
- priorities 4-33

chip selects 4-32
- clock stretching 4-39
- CSCSTR — Chip select clock stretch register 4-39
- general-purpose 4-35
- GPCS1A — General-purpose chip select 1 address reg. 4-35
- GPCS1C — general-purpose chip select 1 control reg. 4-36
- GPCS2A — General-purpose chip select 2 address reg. 4-37
- GPCS2C — General-purpose chip select 2 control reg. 4-37
- I/O (CSIO) 4-33
- program ($\overline{\text{CSPROG}}$) 4-33

CLK4X - bit in CONFIG 4-12
clock monitor 5-3, 5-5
clocks
- CMOS compatible 2-3
- E 2-3, 4-17
- monitor reset 5-3, 5-5
- PWM 9-40
- SPI 8-4
- timer divider chains *9-2*

CME - bit in OPTION 5-4
coherency, timer 9-9, 9-19, 9-26
CON12 - bit in PWCLK 9-38

CON34 - bit in PWCLK 9-38
concatenation, of PWM 9-38
CONFIG — System configuration reg. 4-12
  programming 4-44
configuration 4-12
CONV8 - bit in ADCTL 10-7
COP 9-3, 9-33
  CONFIG — Configuration control reg. 5-5
  COPRST — Arm/reset COP timer circuitry reg. 5-3
  enable 5-6
  OPTION — System configuration options reg. 1 5-4
  rates 5-2, 5-5
  reset 5-2, 5-3, 5-8
  timeout 5-2
COPRST — Arm/reset COP timer circuitry reg. 5-3
corruption
  of A/D 6-6
  of memory 2-2
CPHA - bit in SPCR 8-3, 8-4, 8-7
CPOL - bit in SPCR 8-7
CPU
  accumulators (A, B and D) 3-2
  architecture 3-1
  CCR — condition code reg. 3-4
  index registers (IX, IY) 3-2
  program counter (PC) 3-4
  programming model *3-1*
  registers 3-1
  reset 5-7
CR[1:0] - bits in OPTION 5-5
CSCSTR — Chip select clock stretch register 4-39
CSTL — Chip select control register 4-34
CWOM - bit in OPT2 6-14

D

data format, SCI 7-2
data types 3-6
DDA[7:0] - bits in DDRA 6-2
DDB[7:0] - bits in DDRB 6-3
DDC[7:0] - bits in DDRC 6-4
DDD[5:0] - bits in DDRD 6-5
DDF[7:0] - bits in DDRF 6-7
DDG[7:0] - bits in DDRG 6-9
DDH[7:0] - bits in DDRH 6-10
DDJ[7:0] - bits in DDRJ 6-11
DDK[7:0] - bits in DDRK 6-12
DDRA — Data direction reg. for port A 6-2
DDRB — Data direction reg. for port B 6-3
DDRC — Data direction reg. for port C 6-4
DDRD — Data direction reg. for port D 6-5
DDRF — Data direction reg. for port F 6-7
DDRG — Data direction reg. for port G 6-9
DDRH — Data direction reg. for port H 6-10
DDRJ — Data direction reg. for port J 6-11
DDRK — Data direction reg. for port K 6-12
development tools C-1
DIR - direct addressing mode 3-7
DISCP - bit in PWEN 9-42
DLY - bit in OPTION 4-16
duty cycle, PWM 9-44
DWOM - bit in SPCR 8-6

E

E clock pin 2-4
ECEB, ECEA - bits in TCTL4 9-21
ECEB, ECEA - bits in TCTL6 9-28
EDGB, EDGA - bits in TCTL4 9-21
EDGB, EDGA - bits in TCTL6 9-27
EDGxA and EDGxB - bits in TCTL2 9-6
EELAT - bit in PPROG 4-42
EEON - bit in CONFIG 4-13
EEPGM - bit in PPROG 4-42
EEPROM 4-41—4-44
  erased state ($FF) 4-41
  erasing 4-43—4-44
  PPROG — EEPROM programming control reg. 4-41
  security 4-45
EEx - bits in INIT2 4-15
ERASE - bit in PPROG 4-42
erased state
  EEPROM ($FF) 4-41
error detection, SCI 7-5
ESD protection A-1
EVEN - bit in PPROG 4-41
event counter - see pulse accumulator
EVS — Evaluation system C-1
expansion address lines 4-23, 4-24
EXTAL pin 2-3

F

F23FRC — Compare force reg. for timers 2 and 3 9-18
FCME - bit in OPTION 5-5
FE - bit in SCSR1 7-11
FOC[1:5] - bits in CFORC 9-10
FPPUE - bit in PPAR 6-13
free-running counter 9-1
FT3Cx, FT2Cx - bits in F23FRC 9-19

G

G1A[18:11] - bits in GPCS1A 4-35
G1AV - bit in GPCS1C 4-36
G1DG2 - bit in GPCS1C 4-36
G1DPC - bit in GPCS1C 4-36
G1POL - bit in GPCS1C 4-36
G1SZA—G1SZD - bits in GPCS1C 4-36
G2A[18:11] - bits in GPCS2A 4-37
G2AV - bit in GPCS2C 4-38
G2DPC - bit in GPCS2C 4-37
G2POL - bit in GPCS2C 4-38
G2SZA—G2SZD - bits in GPCS2C 4-38
GCSPR - bit in CSCTL 4-34

general-purpose chip selects 4-35
GP1SA, GP1SB - bits in CSCSTR 4-39
GP2SA, GP2SB - bits in CSCSTR 4-39
GPCS1A — General-purpose chip select 1 address reg. 4-35
GPCS1C — general-purpose chip select 1 control reg. 4-36
GPCS2A — General-purpose chip select 2 address reg. 4-37
GPCS2C — General-purpose chip select 2 control reg. 4-37
GPPUE - bit in PPAR 6-13

H

H-bit in CCR 3-6
HPPUE - bit in PPAR 6-13
HPRIO — Highest priority I-bit interrupt & misc. reg. 4-11

I

I/O chip select (CSIO) 4-33
I/O, on reset 5-7
I1/O4 - bit in TCTL4 9-22
I1/O4 - bit in TCTL6 9-28
I4/O5F - bit in TFLG1 9-13
I4/O5I - bit in TMSK1 9-12
I-bit in CCR 3-5, 5-14
IC1F–IC3F - bits in TFLG1 9-13
IC1I–IC3I - bits in TMSK1 9-12
IDLE - bit in SCSR1 7-10
idle-line wakeup 7-4
ILIE - bit in SCCR2 7-9
illegal opcode trap 5-14
ILT - bit in SCCR1 7-8
IMM - immediate addressing mode 3-7
IND, X/Y - indexed addressing modes 3-8
index registers (IX, IY) 3-2
INH - inherent addressing mode 3-8
INIT — RAM and I/O mapping reg. 4-13
initialization 4-12
input capture 9-4
instruction set 3-8
internal oscillator 4-16, A-15
interrupts
  I-bit 3-5, 5-14
  illegal opcode trap 5-14
  $\overline{\text{IRQ}}$ 2-4
  maskable 5-15
  multiple sources 2-5
  non-maskable 5-14
  priorities 5-9
  priority resolution *5-19*
  SCI *5-22*, *7-14*
  sensitivity 2-4
  stacking 5-13
  SWI 5-14
  triggering 2-4
  types 5-13
  vectors 5-12
  wired-OR 2-4
  X-bit 3-6, 5-14
  $\overline{\text{XIRQ}}$ 2-5, 5-14
IOCSA - bit in CSCTL 4-34
IOEN - bit in CSCTL 4-34
IOPL - bit in CSCTL 4-34
IOSA, IOSB - bits in CSCSTR 4-39
IOSZ - bit in CSCTL 4-34
$\overline{\text{IRQ}}$ pin 2-4
IRQE - bit in OPTION 4-16
IRVNE - bit in OPT2 4-17

J

junction temperature, chip A-2

L

LCD driver interface 8-1
$\overline{\text{LIR}}$ pin 2-5
LIRDV - bit in OPT2 4-17
LOOPS - bit in SCCR1 7-7
low power modes
  RAM 4-4
  stand-by connections *2-5*
  stand-by voltage 2-5
  STOP 5-16
  WAIT 5-15
low voltage inhibit circuit 2-2
LSBF - bit in OPT2 8-10
LVI 2-2
LVPEN - bit in BPROT 4-19
LVPI - bit in PPROG 4-41

M

M - bit in SCCR1 7-7
mask options
  security 4-45
maximum ratings A-1
MDA - bit in HPRIO 4-11
memory
  corruption of 2-2
  EEPROM 4-41—4-44
  map *4-3*
  mapping 4-3, 4-13—4-15
  protection 4-18, 4-45
  RAM 4-4
  RAM stand-by connections *2-5*
  register map 4-4
memory expansion 4-22
  address lines 4-23, 4-24
  examples 4-24—??
  MM1CR, MM2CR — Memory mapping window 1 and 2

control registers 4-31
MMSIZ — Memory mapping window size register 4-29
MMWBR — Memory mapping window base register 4-30
PGAR — Port G assignment register 4-32
memory map, on reset 5-7
MISO 8-4
MM1CR, MM2CR — Memory mapping window 1 and 2 control registers 4-31
MMSIZ — Memory mapping window size register 4-29
MMWBR — Memory mapping window base register 4-30
MODA/$\overline{\text{LIR}}$ pin 2-5
MODB/VSTBY pin 2-5
MODF - bit in SPSR 8-8
MOSI 8-4
MSTR - bit in SPCR 8-5, 8-6
MULT — bit in ADCTL 10-7
MXGS[2:1] - bits in MMSIZ 4-29

N

N-bit in CCR 3-5
NF - bit in SCSR1 7-11
NMI 2-5, 5-14
NOCOP - bit in CONFIG 5-6
noise 2-4
non-maskable interrupt 2-5
NOSEC - bit in CONFIG 4-46

O

OC1D — Output compare 1 data register 9-10
OC1D[7:3] - bits in OC1D 9-10
OC1F–OC3F - bits in T2FLG 9-23
OC1F–OC3F - bits in T3FLG 9-30
OC1F–OC4F - bits in TFLG1 9-13
OC1I–OC3I - bits in T2MSK 9-22
OC1I–OC3I - bits in T3MSK 9-29
OC1I–OC4I - bits in TMSK1 9-12
OC1M — Output compare 1 mask register 9-10
OC1M[7:3] - bits in OC1M 9-10
ODD - bit in PPROG 4-41
OL[1:4] - bits in TCTL3 9-20
OL[1:4] - bits in TCTL5 9-27
OL[2:5] - bits in TCTL1 9-11
OM[1:4] - bits in TCTL3 9-20
OM[1:4] - bits in TCTL5 9-27
OM[2:5] - bits in TCTL1 9-11
operating modes 4-1
baud rates 4-2
bootstrap 4-2
expanded 4-1
HPRIO register 4-11
selection of 2-5, 4-10
single chip 4-1
STOP 4-4, 5-16
test 4-2
VSTBY 4-4
WAIT 5-15
OPT2 — System configuration options reg. 2 4-17
OPTION — System configuration options reg. 1 5-4
OR - bit in SCSR1 7-11
oscillator 2-3
connections *2-3*
output compare 9-8
overflow bit in CCR 3-5

P

packages
options 2-1, B-1
thermal characteristics A-1
PACNT — Pulse accumulator count reg. 9-36
PACTL — Pulse accumulator control reg. 9-35
PAEN - bit in PACTL 9-35
PAIF - bit in TFLG2 9-37
PAII - bit in TMSK2 9-37
PAMOD - bit in PACTL 9-35
PAOVF - bit in TFLG2 9-36
PAOVI - bit in TMSK2 9-36
PAREN - bit in CONFIG 6-15
PCKA[2:1] - bits in PWCLK 9-40
PCKB[3:1] - bits in PWCLK 9-40
PCLK[2:1] - bits in PWPOL 9-41
PCLK[4:3] - bits in PWPOL 9-41
PCSA, PCSB - bits in CSCSTR 4-39
PCSEN - bit in CSCTL 4-34
PCSZA, PCSZB - bits in CSCTL 4-34
PE - bit in SCCR1 7-8
PEDGE - bit in PACTL 9-35
PF - bit in SCSR1 7-11
PGAR — Port G assignment register 4-32
PGAR[5:0] - bits in PGAR 4-32
pins
E clock 2-4
EXTAL 2-3
$\overline{\text{IRQ}}$ 2-4
$\overline{\text{LIR}}$ 2-5
MODA/$\overline{\text{LIR}}$ 2-5
MODB/VSTBY 2-5
OC1, special features 9-4, 9-8
R/$\overline{\text{W}}$ 2-6
$\overline{\text{RESET}}$ 2-2, 5-2
VDD AD, VSS AD 2-4
VDD, VSS 2-4
VRH, VRL 2-6
VSTBY 2-5
$\overline{\text{XIRQ}}$ 2-5
XOUT 2-4
XTAL 2-3
POR 5-1
PORTA — Port A data reg. 6-2
PORTB — Port B data reg. 6-3
PORTC — Port C data reg. 6-4
PORTD — Port D data reg. 6-5
PORTE — Port E data reg. 6-6

PORTF — Port F data reg. 6-7
PORTG — Port G data reg. 6-8
PORTH — Port H data reg. 6-10
PORTJ — Port J data reg. 6-11
PORTK — Port K data reg. 6-12
ports
A (Timer 1) 2-6, 6-2
B (ADDR[15:8]) 2-8, 6-3
C (DATA[7:0]) 2-8, 6-4
D (SCI, SPI) 2-8, 6-5
DDRA — Data direction reg. for port A 6-2
DDRB — Data direction reg. for port B 6-3
DDRC — Data direction reg. for port C 6-4
DDRD — Data direction reg. for port D 6-5
DDRF — Data direction reg. for port F 6-7
DDRG — Data direction reg. for port G 6-9
DDRH — Data direction reg. for port H 6-10
DDRJ — Data direction reg. for port J 6-11
DDRK — Data direction reg. for port K 6-12
E (A/D) 2-9, 6-6
F (ADDR[7:0]) 2-9, 6-7
G (Memory expansion, A/D) 2-9, 6-8
H (Chip select, PWM) 2-10, 6-10
J (Timer 2) 2-10, 6-11
K (Timer 3) 2-10, 6-12
PGAR — Port G assignment register 4-32
PORTA — Port A data reg. 6-2
PORTB — Port B data reg. 6-3
PORTC — Port C data reg. 6-4
PORTD — Port D data reg. 6-5
PORTE — Port E data reg. 6-6
PORTF — Port F data reg. 6-7
PORTG — Port G data reg. 6-8
PORTH — Port H data reg. 6-10
PORTJ — Port J data reg. 6-11
PORTK — Port K data reg. 6-12
signals 2-6
power-on reset - see POR
PPAR — Port pull-up assignment reg. 6-13
PPOL[4:1] - bits in PWPOL 9-41
PPROG — EEPROM programming control reg. 4-41
PR[1:0] - bits in TMSK2 4-20, 9-14
PR2A, PR2B - bits in TCTL4 9-21
PR3A, PR3B - bits in TCTL6 9-28
prebyte 3-7
prescaler, PWM 9-40
priorities, resets and interrupts 5-9, 5-11
program chip select ($\overline{\text{CSPROG}}$) 4-33
program counter (PC) 3-4
programming
CONFIG 4-44
EEPROM 4-41
protection
of memory 4-18, 4-45
registers 4-10
PSEL[4:0] - bits in HPRIO 5-11
PT - bit in SCCR1 7-8
PTCON - bit in BPROT 4-19
pull-ups 6-13
pulse accumulator 9-1, 9-33
block diagram *9-34*
PACNT — Pulse accumulator count reg. 9-36
PACTL — Pulse accumulator control reg. 9-35
reset 5-8
TFLG2 — Timer interrupt flag 2 reg. 9-36
TMSK2 — Timer interrupt mask 2 reg. 9-36
pulse-width modulation - see PWM
PWCLK — PWM clock prescaler and 16-bit select reg. 9-38
PWCNT1–4 — PWM timer counter reg. 1 to 4 9-43
PWDTY1–4 — PWM timer duty cycle reg. 1 to 4 9-44
PWEN — PWM timer enable reg. 9-42
PWEN[4:1] - bits in PWEN 9-42
PWM 9-37
16-bit operation 9-38
block diagram *9-39*
boundary conditions 9-44
clock select 9-40
duty cycle 9-37, 9-44
periods 9-37
pins 9-37
PWCLK — PWM clock prescaler and 16-bit select reg. 9-38
PWCNT1–4 — PWM timer counter reg. 1 to 4 9-43
PWDTY1–4 — PWM timer duty cycle reg. 1 to 4 9-44
PWEN — PWM timer enable reg. 9-42
PWPER1–4 — PWM timer period reg. 1 to 4 9-43
PWPOL — PWM timer polarity & clock source select reg. 9-41
PWSCAL — PWM timer prescaler reg. 9-41
PWPER1–4 — PWM timer period reg. 1 to 4 9-43
PWPOL — PWM timer polarity & clock source select reg. 9-41
PWSCAL — PWM timer prescaler reg. 9-41

R

R/T[7:0] - bits in SCDRL 7-12
R/$\overline{\text{W}}$ pin 2-6
R8 - bit in SCDRH 7-12
RAF - bit in SCSR2 7-11
RAM 4-4
data retention 4-4
security 4-45
RAM[3:0] - bit in INIT 4-13
RBOOT - bit in HPRIO 4-11
RDRF - bit in SCSR1 7-10
RE - bit in SCCR2 7-9
real-time interrupt - see RTI
receiver flags, SCI 7-13
REG[3:0] - bit in INIT 4-13
REL - relative addressing mode 3-8
$\overline{\text{RESET}}$ pin 2-2
resets
circuit *2-2*
clock monitor 5-3, 5-5
COP 5-2, 5-3
effect on A/D 5-9
effect on COP 5-8
effect on CPU 5-7

effect on I/O 5-7
effect on memory map 5-7
effect on pulse accumulator 5-8
effect on RTI 5-8
effect on SCI 5-8
effect on SPI 5-9
effect on system 5-9
effect on timer 5-7
effects of 5-6
external 5-2
HPRIO — Highest priority I-bit interrupt and misc. reg. 5-10
power-on, POR 5-1
priorities 5-9
processing flow *5-17*
$\overline{\text{RESET}}$ pin 5-2
vectors 5-6, 5-12
resetting the COP watchdog 5-3
RFI 2-4
RIE - bit in SCCR2 7-9
ROW - bit in PPROG 4-42
RTI 9-1, 9-31
PACTL — Pulse accumulator control reg. 9-33
rates 9-31
reset 5-8
TFLG2 — Timer interrupt flag reg. 2 9-32
TMSK2 — Timer interrupt mask reg. 2 9-31
RTIF - bit in TFLG2 9-32
RTII - bit in TMSK2 9-31
RTR[1:0] - bits in PACTL 9-33
RWU - bit in SCCR2 7-4, 7-9

S

S-bit in CCR 3-6
SBK - bit in SCCR2 7-9
SBR[12:0] - bits in SCBDH/L 7-6
SCAN - bit in ADCTL 10-7
SCBDH, SCBDL — SCI baud rate control reg. 7-6
SCCR1 — SCI control reg. 1 7-7
SCCR2 — SCI control reg. 2 7-9
SCDRH, SCDRL — SCI data high/low reg. 7-12
SCI 7-1
baud rate *7-1*, 7-6
block diagram *7-3*
data format 7-2
error detection 7-5
interrupt source resolution *5-22*, *7-14*
pins 7-1
receive operation 7-2
reset 5-8
SCBDH, SCBDL — SCI baud rate control reg. 7-6
SCCR1 — SCI control reg. 1 7-7
SCCR2 — SCI control reg. 2 7-9
SCDRH, SCDRL — SCI data high/low reg. 7-12
SCSR1 — SCI status reg. 1 7-10
SCSR2 — SCI status reg. 2 7-11
status flags 7-12
transmit operation 7-2
wakeup 7-4
SCK 8-4
SCSR1 — SCI status reg. 1 7-10
SCSR2 — SCI status reg. 2 7-11
security 4-45
mask option 4-45
NOSEC bit 4-46
sensitivity, of interrupts 2-5, 4-16
serial communications interface - see SCI
serial peripheral interface - see SPI
slave select ($\overline{\text{SS}}$) 8-4
SMOD - bit in HPRIO 4-11
software interrupt (SWI) 5-14
SPCR — Serial peripheral control reg. 8-6
SPDR — SPI data reg. 8-9
SPE - bit in SPCR 8-6
SPI 8-1
block diagram *8-2*
buffering 8-1, 8-9
clock phase 8-3
clock polarity 8-7
clock rate 8-4, 8-7
errors 8-5
master mode 8-6
MISO 8-4
MOSI 8-4
OPT2 — System configuration options reg. 2 8-9
pins 8-1
polarity 8-3
reset 5-9
SCK 8-4
signals 8-3
SPCR — Serial peripheral control reg. 8-6
SPDR — SPI data reg. 8-9
SPSR — Serial peripheral status reg. 8-8
$\overline{\text{SS}}$ 8-4
transfer formats 8-2, *8-3*
SPIE - bit in SPCR 8-5, 8-6
SPIF - bit in SPSR 8-8
SPR1 and SPR0 - bits in SPCR 8-7
SPR2 - bit in OPT2 8-10
SPSR — Serial peripheral status reg. 8-8
stack pointer (SP) 3-2
stacking operations *3-3*
stand-by voltage 2-5
status flags, SCI 7-12
STOP mode 4-4, 5-16
disabling 3-6
SWI 5-14
SYNC - bit in OPT2 7-6
system reset 5-9

T

T2C4 — Timer 2 channel 4 register 9-19
T2FLG — Timer 2 interrupt flag register 9-23
T2MSK — Timer 2 interrupt mask register 9-22
T2OC1—T2OC3 — Timer 2 output compare registers 9-19
T2STP - bit in TCTL4 9-22

T3C4 — Timer 3 channel 4 register 9-24
T3FLG — Timer 3 interrupt flag register 9-30
T3MSK — Timer 3 interrupt mask register 9-29
T3OC1—T3OC3 — Timer 3 output compare registers 9-26
T3STP - bit in TCTL6 9-28
T8 - bit in SCDRH 7-12
TC - bit in SCSR1 7-10
TCIE - bit in SCCR2 7-9
TCNT — Timer counter register 9-11
TCNT2 — Timer 2 counter register 9-20
TCNT3 — Timer 3 counter register 9-26
TCTL1 — Timer control register 1 9-11
TCTL2 — Timer control register 2 9-6
TCTL3 — Timer control register 3 (timer 2) 9-20
TCTL4 — Timer control register 4 (Timer 2) 9-21
TCTL5 — Timer control register 5 (timer 3) 9-27
TCTL6 — Timer control register 6 (Timer 3) 9-27
TDRE - bit in SCSR1 7-10
TE - bit in SCCR2 7-9
test methods *A-3*
TFLG1 — Timer interrupt flag register 1 9-13
TFLG2 — Timer interrupt flag register 2 9-15
TI4/O5 — Timer input capture 4/output compare 5 register 9-7
TIC1–TIC3 — Timer input capture register 9-7
TIE - bit in SCCR2 7-9
time accumulation - see pulse accumulator
timers 9-1
  block diagrams *9-5*, *9-17*, *9-25*
  CFORC — Timer compare force register 9-9
  clock divider chains *9-2*
  coherency 9-9, 9-19, 9-26
  COP 9-33
  F23FRC — Compare force reg. for timers 2 and 3 9-18
  free-running counter 9-1
  input capture 9-4, 9-18
  OC1, special features 9-4, 9-8
  OC1D — Output compare 1 data register 9-10
  OC1M — Output compare 1 mask register 9-10
  output compare 9-8, 9-18
  pins 9-3, 9-15, 9-24
  reset 5-7
  T2C4 — Timer 2 channel 4 register 9-19
  T2FLG - Timer 2 interrupt flag register 9-23
  T2MSK - Timer 2 interrupt mask register 9-22
  T2OC1—T2OC3 — Timer 2 output compare registers 9-19
  T3C4 — Timer 3 channel 4 register 9-24
  T3FLG - Timer 3 interrupt flag register 9-30
  TCNT — Timer counter register 9-11
  TCNT2 — Timer 2 counter register 9-20
  TCNT3 — Timer 3 counter register 9-26
  TCTL1 — Timer control register 1 9-11
  TCTL2 — Timer control register 2 9-6
  TCTL3 — Timer control register 3 (timer 2) 9-20
  TCTL4 — Timer control register 4 (Timer 2) 9-21
  TCTL5 — Timer control register 5 (timer 3) 9-27
  TCTL6 — Timer control register 6 (Timer 3) 9-27
  TFLG1 — Timer interrupt flag register 1 9-13
  TFLG2 — Timer interrupt flag register 2 9-15
  TI4/O5 — Timer input capture 4/output compare 5 register 9-7
  TIC1–TIC3 — Timer input capture register 9-7
  timer 1 9-1
  timer 2 9-15
  Timer 3 9-24
  TMSK1 — Timer interrupt mask register 1 9-12
  TMSK2 — Timer interrupt mask reg. 2 4-20, 9-14
  TOC1–TOC4 — Timer output compare register 9-9
TMSK1 — Timer interrupt mask register 1 9-12
TMSK2 — Timer interrupt mask reg. 2 4-20, 9-14
TO2F - bit in T2FLG 9-23
TO2I - bit in T2MSK 9-23
TO3F - bit in T3FLG 9-30
TO3I - bit in T3MSK 9-29
TOC1–TOC4 — Timer output compare register 9-9
TOF - bit in TFLG2 9-15, 9-32
TOI - bit in TMSK2 9-14
TPWSL - bit in PWEN 9-42

U

UART 7-1

V

V-bit in CCR 3-5
VDD AD, VSS AD pins 2-4
VDD pin 2-4
vectors
  interrupt 5-12
  reset 5-6, 5-12
VRH, VRL pins 2-6
VSS pin 2-4
VSTBY pin 2-5

W

W1A[15:13] - bits in MMWBR 4-30
W1SZ[1:0] - bits in MMSIZ 4-29
W2A[15:13] - bits in MMWBR 4-30
W2SZ[1:0] - bits in MMSIZ 4-29
WAIT mode 5-15
WAKE - bit in SCCR1 7-8
wakeup, SCI 7-4
watchdog - see COP
WCOL - bit in SPSR 8-8
wired-OR 2-4, 2-8, 4-2, 6-14
WOMS - bit in SCCR1 7-7

X

X1A[18:13] - bits in MM1CR 4-31
X2A[18:13] - bits in MM2CR 4-31
X-bit in CCR 3-6, 5-14

XDV[1:0] - bits in OPT2 4-18
$\overline{\text{XIRQ}}$ 2-5, 5-14
XOUT pin 2-4
xPPUE - bits in PPAR 6-13
XTAL pin 2-3

Z

Z-bit in CCR 3-5

How to reach us:

Mfax: RMFAX0@email.sps.mot.com – TOUCHTONE (602) 244-6609

INTERNET: http://Design-NET.com

USA/EUROPE: Motorola Literature Distribution; P.O. Box 5405; Denver, Colorado 80217. 303-675-2140

JAPAN: Nippon Motorola Ltd.; Tatsumi-SPD-JLDC, Toshikatsu Otsuki, 6F Seibu-Butsuryu-Center, 3-14-2 Tatsumi Koto-Ku, Tokyo 135, Japan. 03-3521-8315

HONG KONG: Motorola Semiconductors H.K. Ltd.; 8B Tai Ping Industrial Park, 51 Ting Kok Road, Tai Po, N.T., Hong Kong. 852-26629298

MOTOROLA