



TK68HC11K Family

Technical Data

November 8, 2019



Address:

7901 E. Riverside Dr. Bldg. 2, Suite 150
Austin, TX 78744

Phone:

(512) 342-9871

Sales Inquiries:

sales@tekmos.com

List of Sections

Technical Data — List of Sections	3
Technical Data — Table of Contents	5
Technical Data — List of Figures	15
Technical Data — List of Tables	21
Section 1. General Description	23
Section 2. Pin Description	29
Section 3. Central Processor Unit (CPU)	45
Section 4. System Configuration	65
Section 5. Resets and Interrupts	101
Section 6. Parallel Input/Output	135
Section 7. Serial Communications Interface (SCI)	149
Section 8. Serial Peripheral Interface (SPI)	167
Section 9. Timing System	181
Section 10. Analog-to-Digital (A/D) Converter	235
Section 11. Memory Expansion and Chip Selects	245
Section 12. Electrical Characteristics	267
Section 13. Mechanical Data	293

Section 14. Ordering Information295

**Section 15. Differences Between Tekmos and Freescale
Devices297**

Table of Contents

Technical Data — List of Sections

Technical Data — Table of Contents

Technical Data — List of Figures

Technical Data — List of Tables

Section 1. General Description

1.1	Contents	23
1.2	Introduction	23
1.3	TK68HC11K Family Members	24
1.4	Features	25
1.5	Structure	27

Section 2. Pin Description

2.1	Contents	29
2.2	Introduction	30
2.3	Power Supply (V_{DD} , V_{SS} , AV_{DD} , and AV_{SS})	35
2.4	Reset ($\overline{\text{RESET}}$)	35
2.5	Crystal Driver and External Clock Input (XTAL and EXTAL)	36
2.6	XOUT	37
2.7	E-Clock Output (E)	37
2.8	Interrupt Request ($\overline{\text{IRQ}}$) and Non-Maskable Interrupt ($\overline{\text{XIRQ}}$)	38

Table of Contents

2.9	DCSAMode Selection, Instruction Cycle Reference, and Standby Power (MODA/ $\overline{\text{LIR}}$ and MODB/ V_{STBY})	39
2.10	V_{RH} and V_{RL}	41
2.11	Port Signals	41

Section 3. Central Processor Unit (CPU)

3.1	Contents	45
3.2	Introduction	46
3.3	CPU Registers	46
3.3.1	Accumulators A, B, and D (ACCA, ACCB, and ACCD)	47
3.3.2	Index Register X (IX)	47
3.3.3	Index Register Y (IY)	47
3.3.4	Stack Pointer (SP)	47
3.3.5	Program Counter (PC)	50
3.3.6	Condition Code Register (CCR)	50
3.3.6.1	Carry/Borrow (C)	50
3.3.6.2	Overflow (V)	50
3.3.6.3	Zero (Z)	50
3.3.6.4	Negative (N)	51
3.3.6.5	Interrupt Mask (I)	51
3.3.6.6	Half Carry (H)	51
3.3.6.7	Non-Maskable Interrupt (X)	51
3.3.6.8	Stop Disable (S)	52
3.4	Data Types	52
3.5	Opcodes and Operands	52
3.6	Addressing Modes	53
3.6.1	Immediate	53
3.6.2	Direct	53
3.6.3	Extended	53
3.6.4	Indexed	54
3.6.5	Inherent	54
3.6.6	Relative	54
3.7	Instruction Set	54

Section 4. System Configuration

4.1	Contents	65
4.2	Introduction	66
4.3	Control Registers	66
4.4	Operating Modes	77
4.4.1	Single-Chip Mode	77
4.4.2	Expanded Mode	77
4.4.3	Bootstrap Mode	78
4.4.4	Special Test Mode	78
4.4.5	Mode Selection	79
4.5	Memory Map	81
4.5.1	Control Registers and RAM	83
4.5.2	ROM or EPROM	84
4.5.3	EEPROM	85
4.5.4	Bootloader ROM	88
4.6	EPROM/OTPROM (TK68HC711K4,TK68HC711KS8, and TK68HC711KS2)88	
4.6.1	Programming the EPROM with Downloaded Data	88
4.6.2	Programming the EPROM from Memory	89
4.7	EEPROM and the CONFIG Register	91
4.7.1	EEPROM Registers	91
4.7.1.1	EEPROM Programming Control Register	92
4.7.1.2	Block Protect Register	94
4.7.1.3	System Configuration Options Register	95
4.7.2	EEPROM Programming	95
4.7.2.1	EEPROM Programming	96
4.7.2.2	EEPROM Bulk Erase	96
4.7.2.3	EEPROM Row Erase	97
4.7.2.4	EEPROM Byte Erase	97
4.7.3	CONFIG Register Programming	97
4.7.4	RAM and EEPROM Security	98

Section 5. Resets and Interrupts

5.1	Contents	101
-----	--------------------	-----

Table of Contents

5.2	Introduction	102
5.3	Sources of Resets	102
5.3.1	Power-On Reset (POR)	103
5.3.2	External Reset (RESET)	104
5.3.3	Computer Operating Properly (COP) System	105
5.3.3.1	System Configuration Register	106
5.3.3.2	System Configuration Options Register	107
5.3.3.3	Arm/Reset COP Timer Circuitry Register	108
5.3.4	Clock Monitor Reset	109
5.3.4.1	System Configuration Options Register	110
5.3.4.2	System Configuration Options Register 2	111
5.4	Effects of Reset	114
5.5	Interrupts	116
5.5.1	Non-Maskable Interrupts	119
5.5.1.1	Non-Maskable Interrupt Request (\overline{XIRQ})	119
5.5.1.2	Illegal Opcode Trap	119
5.5.1.3	Software Interrupt (SWI)	119
5.5.2	Maskable Interrupts	121
5.6	Reset and Interrupt Priority	121
5.7	Reset and Interrupt Processing	123
5.8	Low-Power Operation	129
5.8.1	Wait Mode	130
5.8.2	Stop Mode	131
5.8.3	Slow Mode	133

Section 6. Parallel Input/Output

6.1	Contents	135
6.2	Introduction	136
6.3	Port A	138
6.4	Port B	139
6.5	Port C	140
6.6	Port D	142

6.7	Port E	143
6.8	Port F	144
6.9	Port G	145
6.10	Port H	146
6.11	Internal Pullup Resistors	147

Section 7. Serial Communications Interface (SCI)

7.1	Contents	149
7.2	Introduction	149
7.3	Data Format	150
7.4	Transmit Operation	150
7.5	Receive Operation	153
7.6	Wakeup Feature	156
7.7	Short Mode Idle Line Detection	156
7.8	Baud Rate Selection	157
7.9	SCI Registers	158
7.9.1	SCI Baud Rate Control Register	158
7.9.2	Serial Communications Control Register 1	160
7.9.3	Serial Communications Control Register 2	161
7.9.4	Serial Communication Status Register 1	162
7.9.5	Serial Communication Status Register 2	164
7.9.6	Serial Communications Data Register	165

Section 8. Serial Peripheral Interface (SPI)

8.1	Contents	167
8.2	Introduction	167
8.3	SPI Functional Description	168
8.4	SPI Signal Descriptions	169
8.4.1	Master In Slave Out (MISO)	170

Table of Contents

8.4.2	Master Out Slave In (MOSI)	170
8.4.3	Serial Clock (SCK)	170
8.4.4	Slave Select (\overline{SS})	170
8.4.5	SPI Timing	170
8.5	SPI System Errors	172
8.5.1	Mode Fault Error	172
8.5.2	Write Collision Error	172
8.6	SPI Registers	173
8.6.1	Serial Peripheral Control Register	173
8.6.2	Serial Peripheral Status Register	176
8.6.3	Serial Peripheral Data Register	177
8.6.4	Port D Data Direction Register	178
8.6.5	System Configuration Options 2	179

Section 9. Timing System

9.1	Contents	181
9.2	Introduction	182
9.3	Timer Structure	183
9.4	Input Capture and Output Compare Overview	185
9.4.1	Timer Counter Register	188
9.4.2	Timer Interrupt Flag 2 Register	189
9.4.3	Timer Interrupt Mask 2 Register	190
9.4.4	Port A Data Direction Register	192
9.4.5	Pulse Accumulator Control Register	193
9.5	Input Capture (IC)	193
9.5.1	Timer Input Capture Registers	194
9.5.2	Timer Input Capture 4/Output Compare 5 Register	195
9.5.3	Timer Interrupt Flag 1 Register	196
9.5.4	Timer Interrupt Mask 1 Register	197
9.5.5	Timer Control 2 Register	198
9.6	Output Compare (OC)	198
9.6.1	Timer Output Compare Registers	200
9.6.2	Timer Input Capture 4/Output Compare 5 Register	202
9.6.3	Timer Interrupt Flag 1 Register	203

9.6.4	Timer Interrupt Mask 1 Register	204
9.6.5	Timer Control 1 Register	205
9.6.6	Timer Compare Force Register	206
9.6.7	Output Compare 1 Mask Register	207
9.6.8	Output Compare 1 Data Register	208
9.7	Pulse Accumulator (PA)	209
9.7.1	Port A Data Direction Register	211
9.7.2	Pulse Accumulator Control Register	212
9.7.3	Timer Interrupt Flag 2	214
9.7.4	Timer Interrupt Mask 2 Register	215
9.7.5	Pulse Accumulator Count Register	216
9.8	Real-Time Interrupt (RTI)	216
9.8.1	Timer Interrupt Flag 2 Register	217
9.8.2	Timer Interrupt Mask 2 Register	218
9.8.3	Pulse Accumulator Control Register	219
9.9	XOUT Pin Control	220
9.9.1	System Configuration Register	221
9.9.2	System Configuration Options 2 Register	222
9.10	Pulse-Width Modulator (PWM)	223
9.10.1	PWM System Description	224
9.10.2	Pulse-Width Modulation Control Registers	227
9.10.2.1	Pulse-Width Modulation Timer Clock Select Register	227
9.10.2.2	Pulse-Width Modulation Timer Polarity Register	229
9.10.2.3	Pulse-Width Modulation Timer Prescaler Register	229
9.10.2.4	Pulse-Width Modulation Timer Enable Register	230
9.10.2.5	Pulse-Width Modulation Timer Counters 1 to 4 Registers	231
9.10.2.6	Pulse-Width Modulation Timer Periods 1 to 4 Registers	232
9.10.2.7	Pulse-Width Modulation Timer Duty Cycle 1 to 4 Registers	233

Section 10. Analog-to-Digital (A/D) Converter

10.1	Contents	235
10.2	Introduction	235
10.3	Functional Description	236
10.3.1	Multiplexer	237

10.3.2	Analog Converter	237
10.3.3	Result Registers	237
10.3.4	Digital Control	238
10.4	A/D Control/Status Registers	239
10.4.1	System Configuration Options Register	240
10.4.2	A/D Control/Status Register	241
10.4.3	Analog-to-Digital Converter Result Registers	243
10.5	Design Considerations	244
10.5.1	A/D Input Pins	244
10.5.2	Operation in Stop and Wait Modes	244

Section 11. Memory Expansion and Chip Selects

11.1	Contents	245
11.2	Introduction	246
11.3	Memory Expansion	246
11.3.1	Memory Size and Address Line Allocation	246
11.3.2	Control Registers	248
11.3.2.1	Port G Assignment Register	248
11.3.2.2	Memory Mapping Size Register	249
11.3.2.3	Memory Mapping Window Base Register	250
11.3.2.4	Memory Mapping Window Control Registers	250
11.4	Chip Selects	252
11.4.1	Program Chip Select	254
11.4.2	Input/Output Chip Select	254
11.4.3	General-Purpose Chip Selects	256
11.4.3.1	Memory Mapping Size Register	257
11.4.3.2	General-Purpose Chip Select 1 Address Register	257
11.4.3.3	General-Purpose Chip Select 1 Control Register	258
11.4.3.4	General-Purpose Chip Select 2 Address Register	259
11.4.3.5	General-Purpose Chip Select 2 Control Register	259
11.4.4	One Chip Select Driving Another	260
11.4.4.1	General-Purpose Chip Select 1 Control Register	261
11.4.4.2	General-Purpose Chip Select 2 Control Register	261
11.4.5	Clock Stretching	262
11.5	Memory Expansion Examples	263

Section 12. Electrical Characteristics

12.1	Contents	267
12.2	Introduction	268
12.3	Maximum Ratings for Standard and Extended Voltage Devices	268
12.4	Functional Operating Range	269
12.5	Functional Operating Range, Extended Voltage Devices	269
12.6	Thermal Characteristics	270
12.7	Thermal Characteristics, Extended Voltage Devices	270
12.8	5.0-Volt Electrical Characteristics	271
12.9	5.0-Volt Power Dissipation Characteristics	272
12.10	Extended Voltage Electrical Characteristics	273
12.11	Extended Voltage Power Dissipation Characteristics	274
12.12	Control Timing	276
12.13	Peripheral Port Timing	280
12.14	Analog-to-Digital Converter Characteristics	282
12.15	Expansion Bus Timing	284
12.16	Expansion Bus Timing, Extended Voltage Devices	285
12.17	Serial Peripheral Interface Timing	287
12.18	Serial Peripheral Interface Timing, Extended Voltage Devices	288
12.19	EEPROM Characteristics	291
12.20	EEPROM Characteristics, Extended Voltage Devices	291

Section 13. Mechanical Data

13.1	Contents	293
13.2	Introduction	293

Section 14. Ordering Information

14.1	Contents	295
14.2	Part Designators	295
14.3	Part Number Construction	295

Section 15. Differences Between Tekmos and Freescale Devices

List of Figures

1-1	. TK68HC11K4 Family Block Diagram.	27
1-2	. TK68HC11KS Family Block Diagram.	28
2-1	. Pin Assignments for TK68HC11K 84-Pin PLCC/CLCC.	31
2-2	. Pin Assignments for TK6811K 80-Pin QFP.	32
2-3	. Pin Assignments for TK6811KS 68-Pin PLCC	33
2-4	. Pin Assignments for TK6811KA/KS 80-Pin TQFP.	34
2-5	. External Reset Circuit.	36
2-6	. Common Crystal Connections	37
2-7	. External Oscillator Connections.	37
2-8	. System Configuration Options 2 (OPT2)	39
2-9	. $\overline{\text{LIR}}$ Timing	40
2-10	. $\text{MODB}/V_{\text{STBY}}$ Connection.	40
3-1	. Programming Model.	46
3-2	. Stacking Operations	49
4-1	. Register and Control Bit Assignments	67
4-2	. Highest Priority I-Bit Interrupt and Miscellaneous Register (HPRIO)80	
4-3	. TK68HC11K4 Family Memory Map	81
4-4	. TK68HC11KS Family Memory Map	82
4-5	. RAM and I/O Mapping Register (INIT).	83
4-6	. System Configuration Register (CONFIG)	86
4-7	. EEPROM Mapping Register (INIT2).	87
4-8	. EPROM Programming Control Register (EPROG)	89
4-9	. EEPROM Programming Control Register (PPROG).	92
4-10	. Block Protect Register (BPROT)	94
4-11	. System Configuration Options Register (OPTION).	95
4-12	. Block Protect Register (BPROT)	97
4-13	. System Configuration Register (CONFIG)	98
5-1	. System Configuration Register (CONFIG)	106
5-2	. System Configuration Options Register (OPTION).	107
5-3	. Arm/Reset COP Timer Circuitry Register (COPRST).	108

List of Figures

5-4	. System Configuration Options Register (OPTION).....	110
5-5	. System Configuration Options Register 2 (OPT2).....	111
5-6	. System Configuration Options Register (OPTION).....	121
5-7	. Highest Priority I-Bit Interrupt and Miscellaneous Register (HPRIO)122	
5-8	. Processing Flow Out of Reset (Sheet 1 of 2)	125
5-9	. Interrupt Priority Resolution (Sheet 1 of 2)	127
5-10	. Interrupt Priority Resolution Within SCI System.....	129
5-11	. System Configuration Options Register (OPTION).....	132
5-12	. System Configuration Options 3 Register (OPT3).....	133
5-13	. Slow Mode Example.....	134
6-1	. Port A Data Register (PORTA)	138
6-2	. Port A Data Direction Register (DDRA)	138
6-3	. Port B Data Register (PORTB).....	139
6-4	. Port B Data Direction Register (DDRB)	139
6-5	. Port C Data Register (PORTC).....	140
6-6	. Port C Data Direction Register (DDRC)	141
6-7	. System Configuration Options 2 Register (OPT2).....	141
6-8	. Port D Data Register (PORTD)	142
6-9	. Port D Data Direction Register (DDRD)	142
6-10	. Port E Data Register (PORTE).....	143
6-11	. Port F Data Register (PORTF)	144
6-12	. Port F Data Direction Register (DDRF).....	144
6-13	. Port G Data Register (PORTG)	145
6-14	. Port G Data Direction Register (DDRG)	145
6-15	. Port H Data Register (PORTH)	146
6-16	. Port H Data Direction Register (DDRH)	146
6-17	. Port Pullup Assignment Register (PPAR)	147
6-18	. System Configuration Register (CONFIG)	147
7-1	. SCI Data Formats	150
7-2	. SCI Transmitter Block Diagram.....	152
7-3	. SCI Receiver Block Diagram	155
7-4	. SCI Baud Generator Circuit Diagram.....	157
7-5	. SCI Baud Rate Control Register High (SCBDH).....	158
7-6	. SCI Baud Rate Control Register Low (SCBDL)	158
7-7	. SCI Control Register 1 (SCCR1)	160
7-8	. SCI Control Register 2 (SCCR2)	161
7-9	. SCI Control Register 1 (SCCR1)	162

7-10	. SCI Control Register 2 (SCCR2)	164
7-11	. SCI Data Register (SCDR)	165
8-1	. SPI Block Diagram	169
8-2	. Data Clock Timing Diagram	171
8-3	. Serial Peripheral Control Register (SPCR)	173
8-4	. Serial Peripheral Status Register (SPSR)	176
8-5	. Serial Peripheral Data Register (SPDR)	177
8-6	. Port D Data Direction Register (DDRD)	178
8-7	. System Configuration Options 2 Register (OPT2)	179
9-1	. Timer Clock Divider Chains	183
9-2	. Capture/Compare Block Diagram	187
9-3	. Timer Counter Register (TCNT)	188
9-4	. Timer Interrupt Flag 2 (TFLG2)	189
9-5	. Timer Interrupt Mask 2 (TMSK2)	190
9-6	. Port A Data Direction Register (DDRA)	192
9-7	. Pulse Accumulator Control Register (PACTL)	193
9-8	. Timer Input Capture Registers (TIC1–TIC3)	194
9-9	. Timer Input Capture 4/Output Compare 5 Register (TI4/O5)195	
9-10	. Timer Interrupt Flag 1 Register (TFLG1)	196
9-11	. Timer Interrupt Mask 1 Register (TMSK1)	197
9-12	. Timer Control 2 Register (TCTL2)	198
9-13	. Timer Output Compare Registers (TOC1–TOC4)200	
9-14	. Timer Input Capture 4/Output Compare 5 Register (TI4/O5)202	
9-15	. Timer Interrupt Flag 1 Register (TFLG1)	203
9-16	. Timer Interrupt Mask 1 Register (TMSK1)	204
9-17	. Timer Control Register 1 (TCTL1)	205
9-18	. Timer Compare Force Register (CFORC)	206
9-19	. Output Compare 1 Mask Register (OC1M)	207
9-20	. Output Compare 1 Data Register (OC1D)	208
9-21	. Pulse Accumulator	210
9-22	. Port A Data Direction Register (DDRA)	211
9-23	. Pulse Accumulator Control Register (PACTL)	212
9-24	. Timer Interrupt Flag 2 (TFLG2)	214
9-25	. Timer Interrupt Mask 2 (TMSK2)	215
9-26	. Pulse Accumulator Count Register (PACNT)	216

List of Figures

9-27	. Timer Interrupt Flag 2 Register (TFLG2)	217
9-28	. Timer Interrupt Mask 2 Register (TMSK2)	218
9-29	. Pulse Accumulator Control Register (PACTL)	219
9-30	. System Configuration Register (CONFIG)	221
9-31	. System Configuration Options 2 Register (OPT2)	222
9-32	. Pulse-Width Modulation Timer Block Diagram	225
9-33	. Pulse-Width Modulation Timer Clock Select (PWCLK)	227
9-34	. Pulse-Width Modulation Timer Polarity Register (PWPOL)229	
9-35	. Pulse-Width Modulation Timer Prescaler Register (PWSCAL)229	
9-36	. Pulse-Width Modulation Timer Enable Register (PWEN)230	
9-37	. Pulse-Width Modulation Timer Counters 1 to 4 (PWCNT1 to PWCNT4)231	
9-38	. Pulse-Width Modulation Timer Periods 1 to 4 (PWPER1 to PWPER4)232	
9-39	. Pulse-Width Modulation Timer Duty Cycle 1 to 4 (PWDTY1 to PWDTY4)233	
10-1	. A/D Converter Block Diagram	236
10-2	. A/D Conversion Sequence	238
10-3	. System Configuration Options Register (OPTION)	241
10-4	. Analog-to-Digital Control/Status Register (ADCTL)	241
10-5	. Analog-to-Digital Result Registers (ADR1–ADR4)	243
10-6	. Electrical Model of an A/D Input Pin (Sample Mode)	244
11-1	. Port G Assignment Register (PGAR)	249
11-2	. Memory Mapping Size Register (MMSIZ)	249
11-3	. Memory Mapping Window Base Register (MMWBR)	250
11-4	. Memory Mapping Window Control Registers (MM1CR and MM2CR)251	
11-5	. Chip-Select Control Register (CSCTL)	254
11-6	. Chip-Select Control Register (CSCTL)	255
11-7	. Memory Mapping Size Register (MMSIZ)	257
11-8	. General-Purpose Chip Select 1 Address Register (GPCS1A)257	
11-9	. General-Purpose Chip Select 1 Control Register (GPCS1C)258	
11-10	. General-Purpose Chip Select 2	

Address Register (GPCS2A)	259
11-11 . General-Purpose Chip Select 2	
Control Register (GPCS2C)	259
11-12 . General-Purpose Chip Select 1	
Control Register (GPCS1C)	261
11-13 . General-Purpose Chip Select 2	
Control Register (GPCS2C)	261
11-14 . Chip Select Clock Stretch Register (CSCSTR)	263
11-15 . Memory Expansion Example 1 — Memory Map for a Single 8-Kbyte Window with Eight Banks of External Memory	264
11-16 . Memory Expansion Example 2 (Sheet 1 of 2) Memory Map for One 8-Kbyte Window with Eight Banks and One 16-Kbyte Window with 16 Banks of External Memory	265
12-1 . Test Methods	275
12-2 . Timer Inputs	276
12-3 . POR External Reset Timing Diagram	277
12-4 . STOP Recovery Timing Diagram	278
12-5 . WAIT Recovery from Inerrupt Timing Diagram	279
12-6 . Interrupt Timing Diagram	279
12-7 . Port Read Timing Diagram	281
12-8 . Port Write Timing Diagram	281
12-9 . Expansion Bus Timing	286
12-10 . SPI Timing Diagram (Sheet 1 of 2)	289
14-1 . TK68HC11 Part Numbering System	296

List of Figures

List of Tables

Table	Title	Page
1-1	. TK68HC11K Family Devices	24
2-1	. Package Types for the TK68HC11K Family Devices	30
2-2	. I/O Ports and Peripheral Functions.	42
2-3	. Port Signal Summary	43
3-1	. Instruction Set.	55
4-1	. Write Access-Limited Registers	67
4-2	. Hardware Mode Select Summary.	79
4-3	. TK68HC11K Family Memory Map Addresses	82
4-4	. RAM Mapping.	84
4-5	. Register Mapping	84
4-6	. EEPROM Map	87
4-7	. Scope of EEPROM Erase.	93
4-8	. EEPROM Block Protect	94
5-1	. Reset Vectors	102
5-2	. COP Timeout	107
5-3	. IRVNE Operation After Reset.	112
5-4	. Interrupt and Reset Vector Assignments	116
5-5	. Stacking Order on Entry to Interrupts.	118
5-6	. Highest Priority Interrupt Selection	124
6-1	. Port Configuration.	136
7-1	. SCI Receiver Flags.	153
7-2	. SCI+ Baud Rates	159
8-1	. SPI+ Baud Rates	175
9-1	. Main Timer Rates	184
9-2	. Timer Prescale	191
9-3	. Input Capture Edge Selection.	198
9-4	. Timer Output Compare Actions	205
9-5	. Pulse Accumulator Timing	209
9-6	. Pulse Accumulator Edge Control	213

List of Tables

9-7	. Real-Time Interrupt Rate versus RTR[1:0]	219
9-8	. XOUT Frequencies	222
9-9	. Clock A Prescaler	228
9-10	. Clock B Prescaler	228
10-1	. A/D Converter Channel Selection	239
11-1	. CPU Address and Address Expansion Signals	247
11-2	. Window Size Select	249
11-3	. Memory Expansion Window Base Address	250
11-4	. Chip Select Control Parameter Summary	253
11-5	. Program Chip Select Size	254
11-6	. General-Purpose Chip Select 1 Size Control	258
11-7	. General-Purpose Chip Select 2 Size Control	260
11-8	. One Chip Select Driving Another	262
11-9	. CSCSTR Bits Versus Clock Cycles	263
14-1	. TK68HC11K Family Devices	295

Section 1. General Description

1.1 Contents

1.2	Introduction	23
1.3	TK68HC11K Family Members	24
1.4	Features	25
1.5	Structure	26

1.2 Introduction

The TK68HC11K Family of high-performance microcontroller units (MCUs) offers a non-multiplexed expanded bus, high speed and low power consumption. The fully static design allows operation at frequencies from dc to 5 MHz.

This manual contains information concerning standard, custom-ROM (read-only memory), and extended-voltage devices. Standard devices include those replacing the ROM with:

- Disabled ROM
- Disabled EEPROM (electrically erasable, programmable read-only memory)
- EPROM (erasable, programmable read-only memory)
- OTPROM (one-time programmable read-only memory)

Custom-ROM devices have a ROM array that is programmed at the factory to customer specifications. Extended-voltage devices are guaranteed to operate down to 3.0 Vdc when running at lower frequencies.

1.3 TK68HC11K Family Members

TK68HC11K Family devices feature up to 62 input/output (I/O) lines distributed among eight ports, A through H. The KS Family removes seven pins from port G and four pins from port H for a total of 51 I/O lines. The KSx versions are derived from the KAx version and feature a slow mode for the clocks to allow power conservation. **Table 1-1** lists devices currently available in the K Family along with their distinguishing features.

NOTE: Tekmos no longer offers the KA2 and KA4 devices. These devices have been replaced by the pin-for-pin compatible KS2.

Table 1-1. TK68HC11K Family Devices

Device Number	ROM or EPROM (Bytes)	RAM (Bytes)	EEPROM (Bytes)	I/O (Bytes)	Chip Select	Slow Mode	Packages
TK68HC(L)11K0 TK68HC(L)11K1 TK68HC(L)11K3 TK68HC(L)11K4	0 0 24 K 24 K	768 768 768 768	0 640 0 640	37 37 62 62	Yes Yes Yes Yes	No No No No	84-pin PLCC ⁽¹⁾ 80-pin QFP ⁽²⁾
TK68HC711K4	24 K	768	640	62	Yes	No	84-pin CLCC ⁽³⁾ 84-pin PLCC 80-pin QFP
TK68HC11KS2 TK68HC11KS8	32 K 48 K	1 K 1.5 K	640 640	51 51	No No	Yes Yes	68-pin PLCC 68-pin PLCC and 80-pin TQFP ⁽⁴⁾
TK68HC711KS2 TK68HC711KS8	32 K 48 K	1 K 1.5 K	640 640	51 51	No No	Yes Yes	68-pin CLCC 68-pin PLCC and 80-pin TQFP

1. PLCC = Plastic leaded chip carrier
2. QFP = Quad flat pack
3. CLCC = Ceramic leaded chip carrier
4. TQFP = Thin quad flat pack

1.4 Features

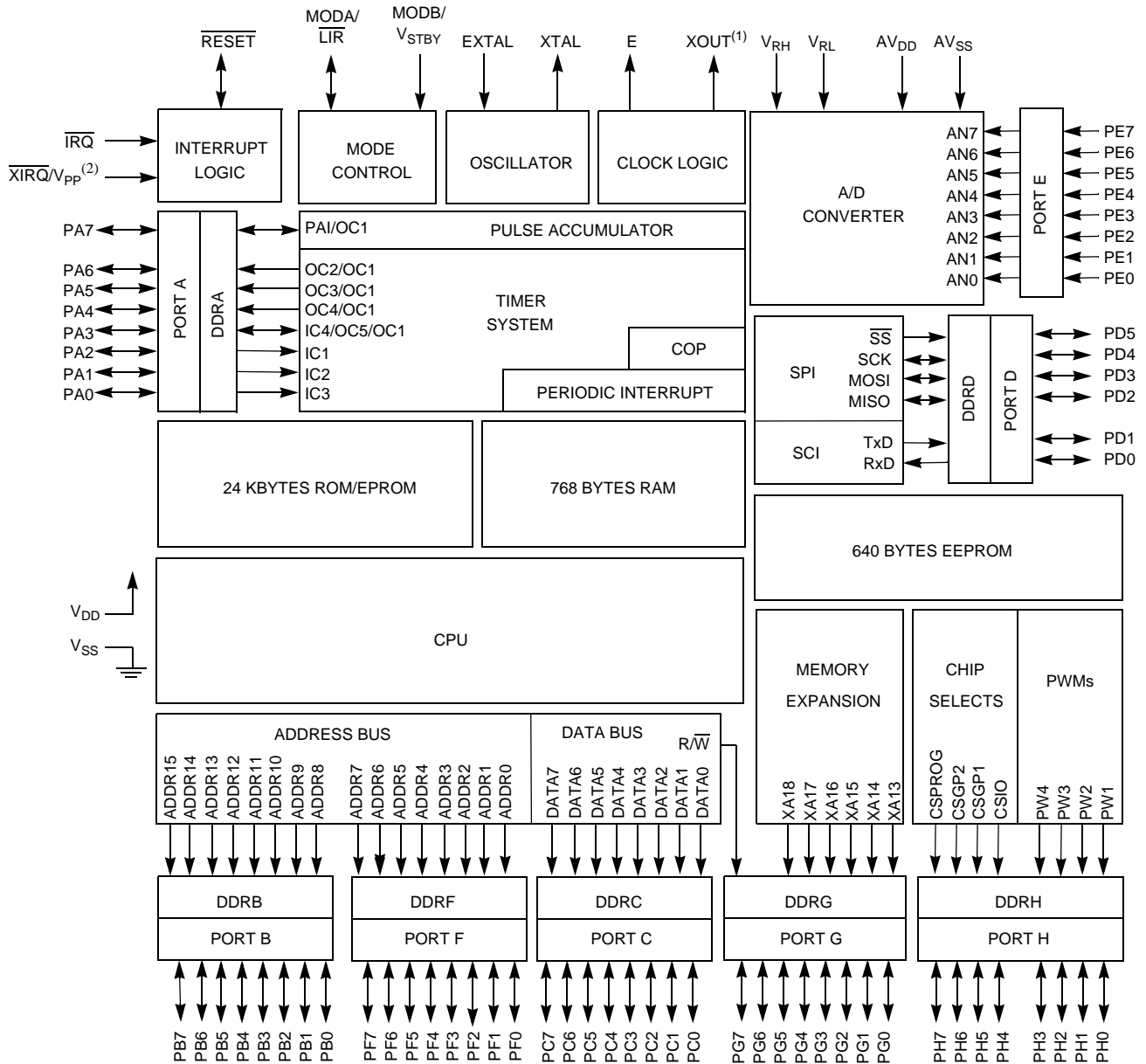
TK68HC11K Family features include:

- 8-bit opcodes and data
- 16-bit addressing
- Two 8-bit accumulators, which can be concatenated to form one 16-bit accumulator
- On-board memory:
 - 32 Kbytes or 48 Kbytes of ROM, EPROM, or OTPROM
 - 768 bytes, 1 Kbyte, or 1.5 Kbytes of static RAM (random-access memory)
 - 640 bytes of EEPROM
 - 128-byte register block
- Dual-function I/O lines — Any pins used for the microcontroller's peripheral functions can be configured as general-purpose I/O lines.
- Non-multiplexed address and data buses
- 68HC11K4 offers:
 - 1 Mbyte of address space, using on-chip memory mapping logic
 - Four programmable chip selects (expanded modes)
- 16-bit timer system:
 - Three input capture (IC) channels, record event timing by storing the value of the timing system's 16-bit free-running counter when an input signal transition occurs.
 - Four output compare (OC) channels, provide timed outputs by signaling when the free-running counter reaches a predetermined number.
 - One IC or OC channel (software selectable)
- 8-bit pulse accumulator
- Four 8-bit pulse-width modulation (PWM) outputs
- Enhanced asynchronous serial communications interface (SCI)
- Enhanced synchronous serial peripheral interface (SPI)
- 8-channel, 8-bit, analog-to-digital (A/D) converter
- Computer operating properly (COP) watchdog system to guard against infinite loops and other system problems
- Real-time interrupt timer

- Power-saving modes:
 - Slow mode reduces power consumption by slowing down internal operations.
 - Wait mode shuts down various system features selected by the user with power consumption typically dropping to 10–100 mW.
 - Stop mode also shuts down system clocks, typically reducing power consumption to about 1.5 mW.
- Package availability for ROM devices:
 - K versions:
 - 84-pin plastic leaded chip carrier (PLCC)
 - 80-pin quad flat pack (QFP)
 - KS versions:
 - 68-pin plastic leaded chip carrier (PLCC)
 - 80-pin thin quad flat pack (TQFP)
- Package availability for EPROM devices:
 - K versions:
 - 84-pin windowed ceramic leaded chip carrier (CLCC)
 - KS versions:
 - 68-pin windowed ceramic leaded chip carrier (CLCC)
 - 80-pin thin quad flat pack (TQFP)

1.5 Structure

Figure 1-1 is a block diagram of the TK68HC11K Family MCU.
Figure 1-2 is a block diagram of the TK68HC11KS devices.

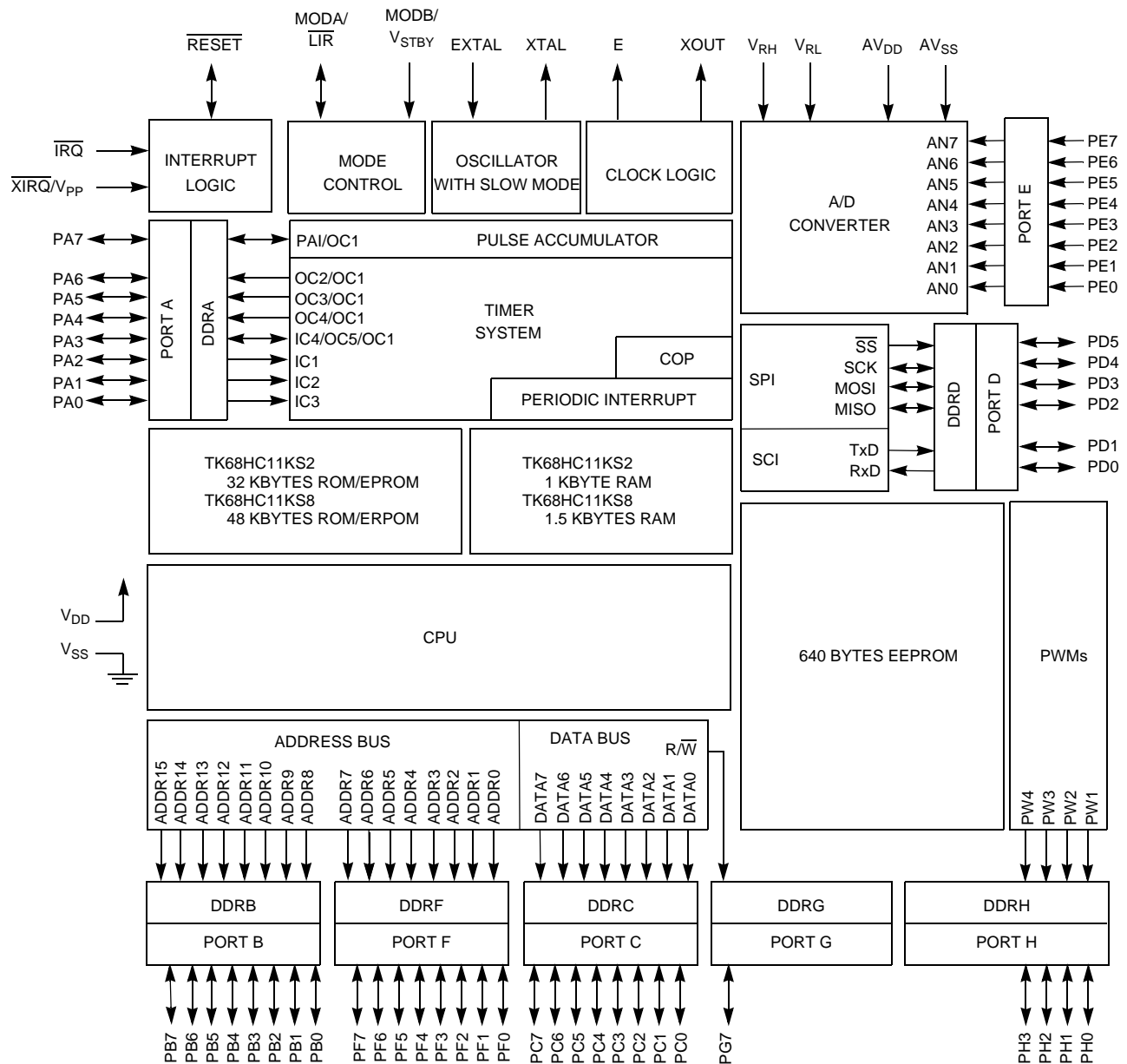


Notes:

1. XOUT pin omitted on 80-pin QFP
2. V_{PP} applies only to EPROM devices.

Figure 1-1. TK68HC11K4 Family Block Diagram

General Description



Note: The configuration shown in this diagram is the TK68HC11KS2 and TK68HC11KS8.

Figure 1-2. TK68HC11KS Family Block Diagram

Section 2. Pin Description

2.1 Contents

2.2	Introduction	29
2.3	Power Supply (V_{DD} , V_{SS} , AV_{DD} , and AV_{SS})	35
2.4	Reset (\overline{RESET})	35
2.5	Crystal Driver and External Clock Input (XTAL and EXTAL)	36
2.6	XOUT	37
2.7	E Clock Output (E)	37
2.8	Interrupt Request (\overline{IRQ}) and Non-Maskable Interrupt (XIRQ)	38
2.9	ABCDMode Selection, Instruction Cycle Reference, and Standby Power (MODA/LIR and MODB/ V_{STBY})	39
2.10	V_{RH} and V_{RL}	41
2.11	Port Signals	41

2.2 Introduction

The TK68HC11K Family is available in a variety of packages, as shown in **Table 2-1**. Most pins on this MCU serve two or more functions, as described in this section. Pin assignments for the various package types are shown in **Figure 2-1**, **Figure 2-2**, **Figure 2-3**, and **Figure 2-4**.

Table 2-1. Package Types for the TK68HC11K Family Devices

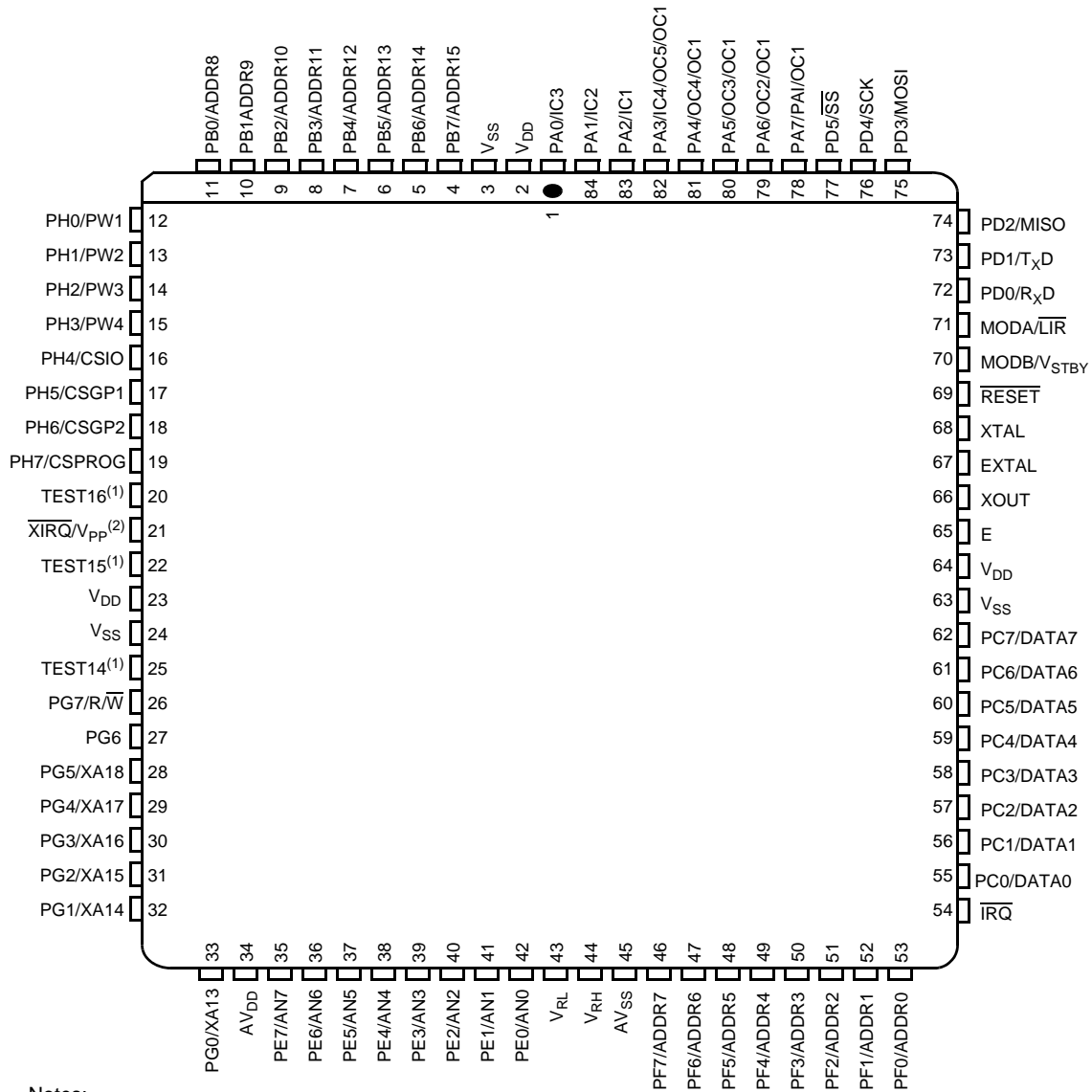
Device Number	Packages
TK68HC(L)11K0 TK68HC(L)11K1 TK68HC(L)11K3 TK68HC(L)11K4	84-pin PLCC ⁽¹⁾ 80-pin QFP ⁽²⁾
TK68HC711K4	84-pin CLCC ⁽³⁾ 84-pin PLCC 80-pin QFP
TK68HC11KS2 TK68HC11KS8	68-pin PLCC 68-pin PLCC and 80-pin TQFP ⁽⁴⁾
TK68HC711KS2 TK68HC711KS8	68-pin CLCC 68-pin PLCC and 80-pin TQFP

1. PLCC = Plastic leaded chip carrier, see **Figure 2-1** and **Figure 2-3**

2. QFP = Quad flat pack, see **Figure 2-2**

3. CLCC = Ceramic leaded chip carrier, **Figure 2-1**

4. TQFP = Thin quad flat pack, see **Figure 2-4**

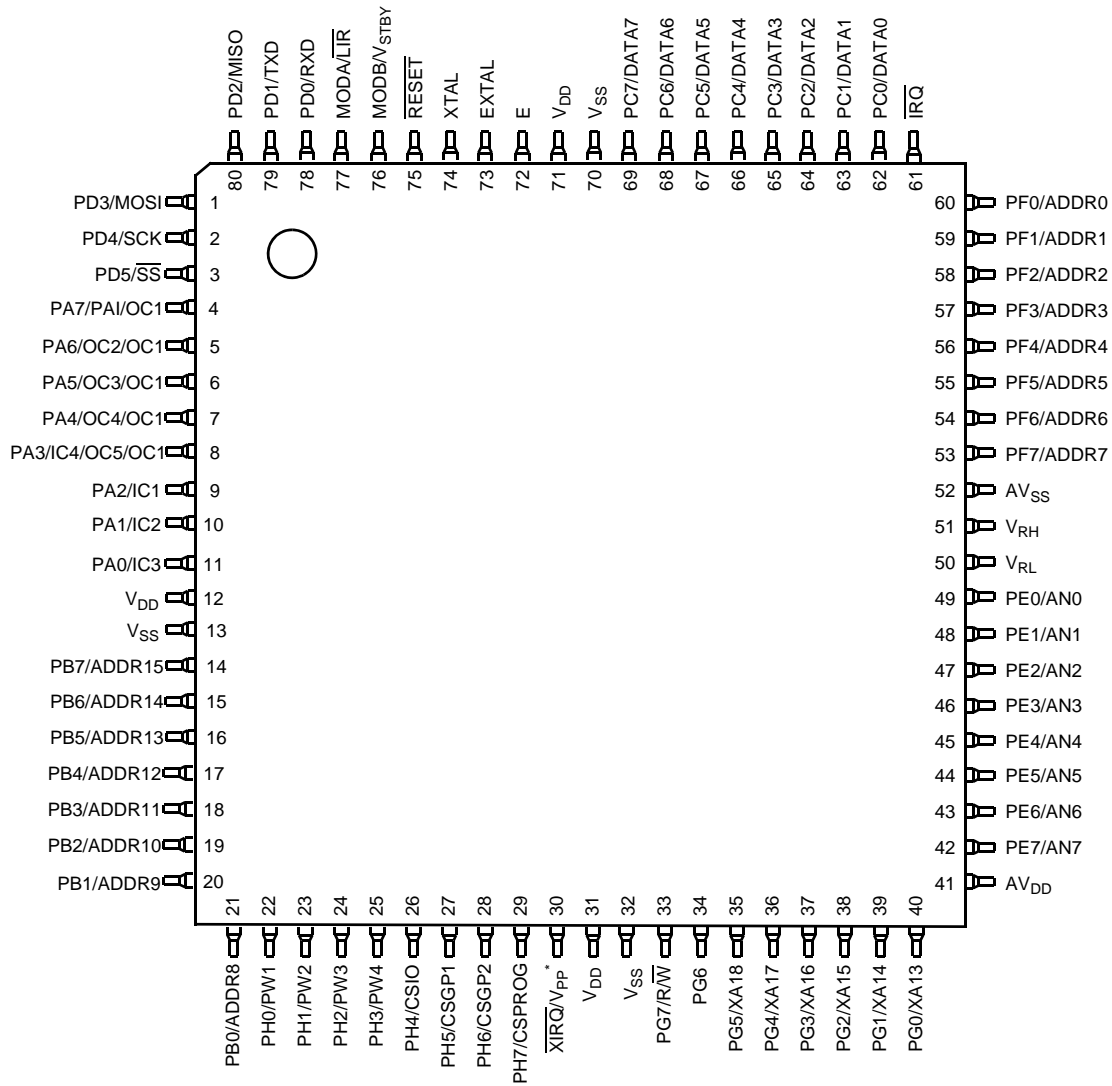


Notes:

1. Pins 20, 22, and 25 are used only during factory testing and should not be connected to external circuitry.
2. VPP applies only to EPROM devices.

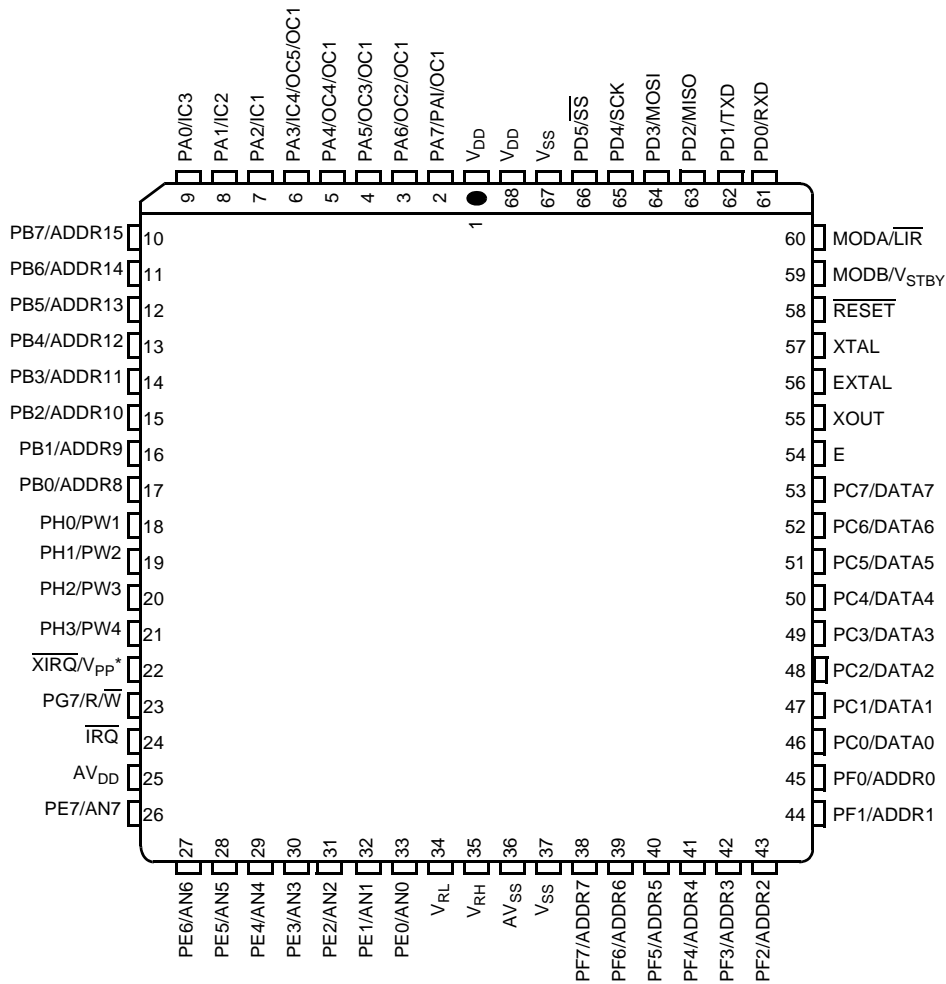
Figure 2-1. Pin Assignments for TK68HC11K 84-Pin PLCC/CLCC

Pin Description



* VPP applies only to EPROM devices.

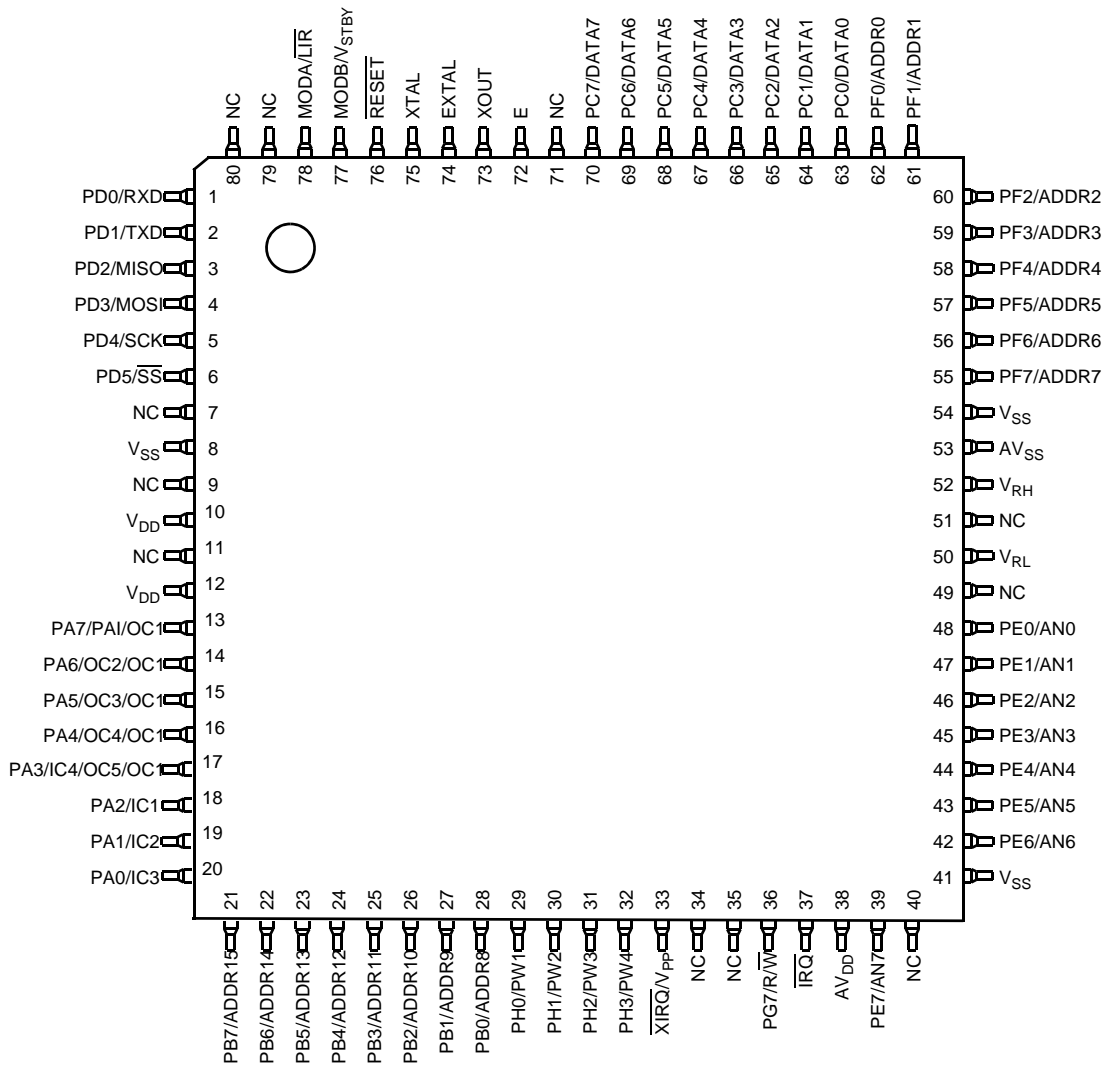
Figure 2-2. Pin Assignments for TK6811K 80-Pin QFP



* VPP applies only to EPROM devices.

Figure 2-3. Pin Assignments for TK6811KS 68-Pin PLCC

Pin Description



* VPP applies only to EPROM devices.

Figure 2-4. Pin Assignments for TK6811KA/KS 80-Pin TQFP

2.3 Power Supply (V_{DD}, V_{SS}, AV_{DD}, and AV_{SS})

The MCU operates from a single 5-volt (nominal) power supply. V_{DD} is the positive power input and V_{SS} is ground. There are three V_{DD}/V_{SS} pairs of pins on the K series devices and two sets on the KS devices. All devices contain a separate pair of power inputs, AV_{DD} and AV_{SS}, for the analog-to-digital (A/D) converter, so that the A/D circuitry can be bypassed independently.

Very fast signal transitions occur on the MCU pins. The short rise and fall times place high, short duration current demands on the power supply. To prevent noise problems, provide good power supply bypassing at the MCU. Also, use bypass capacitors that have good high-frequency characteristics and situate them as close to the MCU as possible. Bypass requirements vary, depending on how heavily the MCU pins are loaded.

2.4 Reset ($\overline{\text{RESET}}$)

This active-low, bidirectional control signal acts as an input to initialize the MCU to a known start-up state. It also serves as an open-drain output to indicate that an internal failure has been detected in either the clock monitor or computer operating properly (COP) watchdog circuit.

The CPU distinguishes between internal and external reset conditions by counting the number of E-clock cycles that occur between the start of reset and the presence of a logic 1 voltage level on the reset pin. Less than two cycles indicates an internal reset; greater than two, an external reset. To prevent the device from misinterpreting the kind of reset that occurs, do not connect an external resistor-capacitor (RC) power-up delay circuit directly to the reset pin.

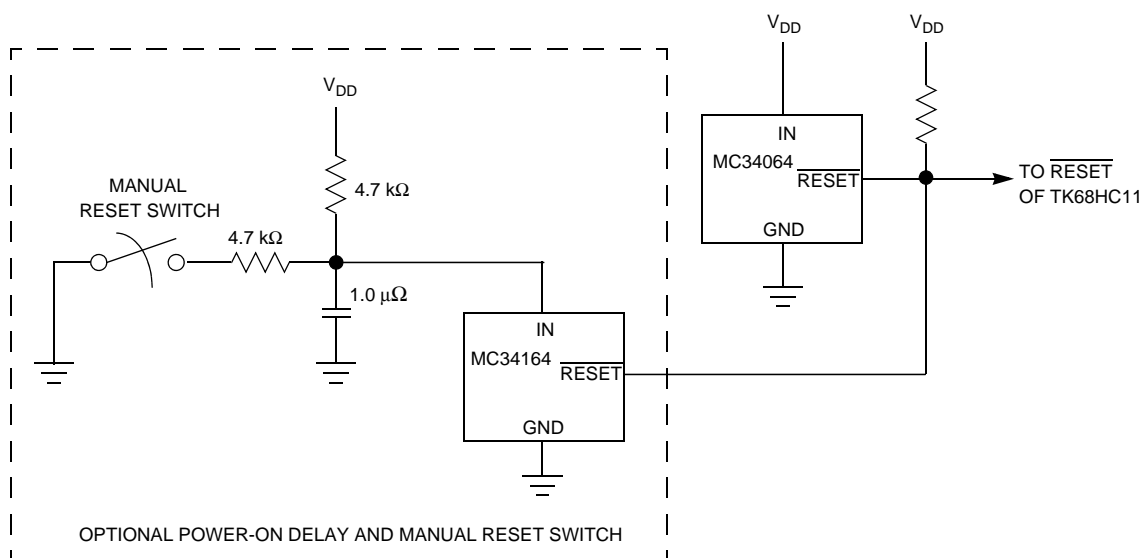


Figure 2-5. External Reset Circuit

It is important to protect the MCU against corruption of RAM and EEPROM during power transitions. This can be done with a low-voltage interrupt (LVI) circuit which holds the $\overline{\text{RESET}}$ pin low when V_{DD} drops below the minimum operating level. **Figure 2-5** shows a suggested reset circuit that incorporates two LVI devices and an external switch.

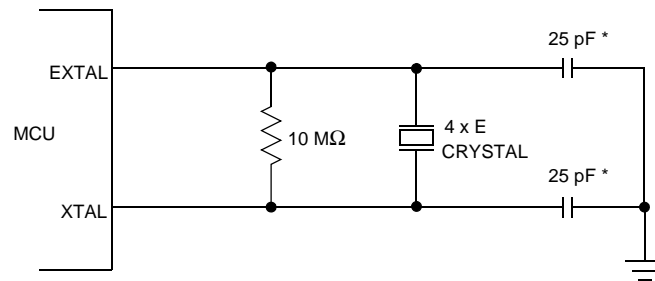
2.5 Crystal Driver and External Clock Input (XTAL and EXTAL)

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 is four times higher than the desired E-clock rate.

When an external CMOS-compatible clock input is connected to the EXTAL pin, the XTAL pin must be left unterminated. The XTAL output also can be used to drive the EXTAL input of another TK68HC11 device or as a general 4x clock reference through a high-impedance buffer.

CAUTION: *In all cases, use caution around the oscillator pins.*

Load capacitances shown in **Figure 2-6** and **Figure 2-7** include all stray layout capacitances.



* This value includes all stray capacitances.

Figure 2-6. Common Crystal Connections

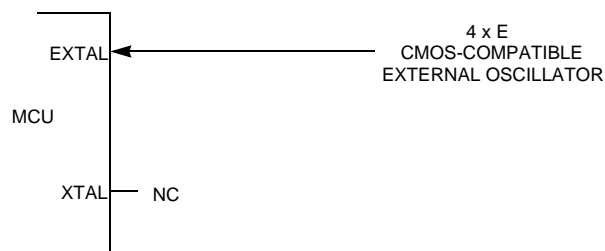


Figure 2-7. External Oscillator Connections

2.6 XOUT

The XOUT pin provides a buffered oscillator signal to synchronize external devices with the MCU. See **9.9 XOUT Pin Control**.

NOTE: This signal is not present on the 80-pin QFP package.

2.7 E-Clock Output (E)

The internally generated instruction cycle clock, or E clock, is available on the E pin as a timing reference. Its frequency is one fourth the input frequency at the XTAL and EXTAL pins. The E clock is low during internal processing and high during external data access. All clocks, including the E clock, are halted when the MCU is in stop mode. The E-pin driver can be turned off in single-chip modes to reduce radio frequency interference (RFI).

2.8 Interrupt Request ($\overline{\text{IRQ}}$) and Non-Maskable Interrupt ($\overline{\text{XIRQ}}$)

The MCU provides two pins for applying asynchronous interrupt requests. Interrupts applied to the $\overline{\text{IRQ}}$ pin can be masked by setting the I bit in the condition code register (CCR), which can be set or cleared by software at any time. Triggering is level sensitive by default, which is required for wire-OR configuration. Software can change the triggering to edge sensitive.

$\overline{\text{XIRQ}}$ interrupts can be non-maskable after reset initialization. Out of reset, the X bit in the CCR is set, masking $\overline{\text{XIRQ}}$ interrupts. Once software clears the X bit, it cannot be reset, and the $\overline{\text{XIRQ}}$ interrupts become non-maskable. The $\overline{\text{XIRQ}}$ input is level sensitive only. $\overline{\text{XIRQ}}$ is often used as a power-loss detect interrupt.

Whenever $\overline{\text{IRQ}}$ or $\overline{\text{XIRQ}}$ is used with multiple interrupt sources, each source must drive the interrupt input with an open-drain type of driver to avoid contention between outputs. There should be a single pullup resistor near the MCU interrupt pin (typically 4.7 k Ω). There must also be an interlock mechanism at each interrupt source which holds the interrupt line low until the MCU recognizes and acknowledges the interrupt request. If any interrupt sources are still pending after the MCU services a request, the interrupt line will remain low, interrupting the MCU again as soon as the I bit in the MCU is cleared (normally upon return from an interrupt). Interrupt mechanisms are explained further in **Section 5. Resets and Interrupts**.

On EPROM devices, the $\overline{\text{XIRQ}}$ pin also functions as the high-voltage supply, V_{PP} , during EPROM or OTPROM programming.

CAUTION: *Ensure that the voltage level at this pin is equal to V_{DD} during normal operation to avoid programming accidents.*

2.9 DCSAMode Selection, Instruction Cycle Reference, and Standby Power (MODA/LIR and MODB/V_{STBY})

During reset, MODA and MODB select one of four operating modes:

1. Single-chip
2. Expanded
3. Bootstrap
4. Special test

For full descriptions of these modes, refer to **4.4 Operating Modes**.

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

$\overline{\text{LIR}}$ can be driven high for a portion of each instruction cycle by setting the LIRDV bit in the system configuration options 2 (OPT2) register (see **Figure 2-8** and **Figure 2-9**). This feature can help detect consecutive instructions and prevent false triggering in high-speed applications.

Address: \$0038

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

*STRCH is not available on KS devices.

U = Undefined

Figure 2-8. System Configuration Options 2 (OPT2)

LIRDV — $\overline{\text{LIR}}$ Driven Bit

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

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

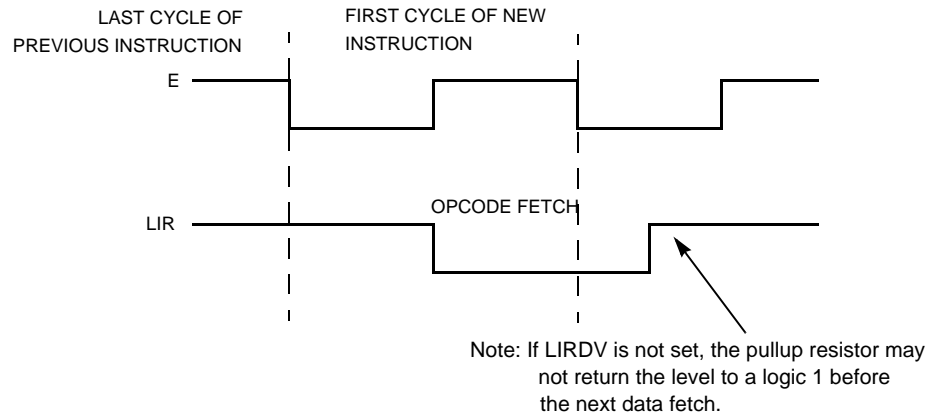


Figure 2-9. $\overline{\text{LIR}}$ Timing

The MODB pin is grounded to select special modes, and has no function after reset. To select the normal operating modes (single-chip and expanded) the MODB pin is pulled to a logic high level. Connecting MODB to a voltage source other than V_{DD} enables it to function as a battery backup input, V_{STBY} . When V_{DD} drops more than one MOS threshold (about 0.7 volts) below the voltage at V_{STBY} , the MCU's RAM and part of the reset logic are powered from V_{STBY} rather than V_{DD} . Reset must be driven low before V_{DD} is removed and must remain low until V_{DD} has been restored to a valid level. The extra hardware required to utilize V_{STBY} may be justified in certain applications where a significant amount of external circuitry operates from V_{DD} . **Figure 2-10** shows a suggested circuit employing the V_{STBY} pin.

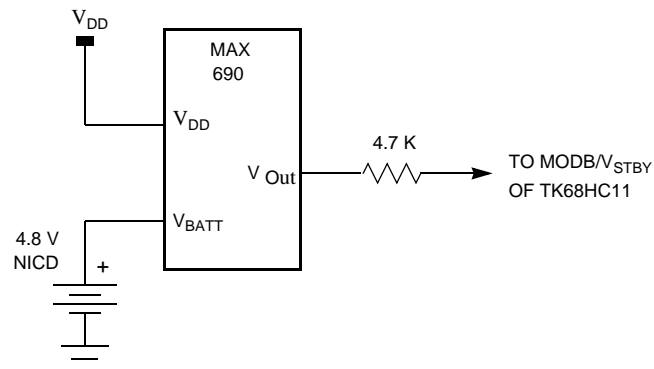


Figure 2-10. MODB/ V_{STBY} Connection

2.10 V_{RH} and V_{RL}

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

2.11 Port Signals

The K series contains 62 input/output lines arranged in eight ports, A through H; all ports are eight bits except port D, which is six bits. The KS series drops seven lines from port G and four from port H, for a total of 51 I/O lines. All ports are fully bidirectional except port E, which is input only.

Each port can serve as either general-purpose I/O or as part of the microcontroller's specialized functions, depending on the operating mode or peripheral functions selected. The functions of ports B, C, and F and port G bit 7 depend on the operating mode. They serve as general-purpose I/O lines in single-chip and bootstrap modes and provide the address and data buses in expanded and special test modes. The other ports serve as general-purpose I/O out of reset; writes to control registers enable their special functions.

Section 6. Parallel Input/Output describes general-purpose I/O operation in detail.

Table 2-2 summarizes the ports and references for peripheral functions.

Table 2-3 summarizes the port signals.

Table 2-2. I/O Ports and Peripheral Functions

I/O Port	Special Function(s)	Enabled by	Refer to
Port A	Timer and pulse accumulator	Control registers	Section 9. Timing System
Port B	High-order address lines	Expanded operating modes	Section 4. System Configuration
Port C	Data lines	Expanded operating modes	Section 4. System Configuration
Port D	Serial communication interface and Serial peripheral interface	Control registers	Section 7. Serial Communications Interface (SCI) and Section 8. Serial Peripheral Interface (SPI)
Port E	A/D converter	Control registers	Section 10. Analog-to-Digital (A/D) Converter
Port F	Low-order address lines	Expanded operating modes	Section 4. System Configuration
Port G bit 7 bits 6–0 ⁽¹⁾	R \bar{W} line and expansion address lines	Expanded operating modes and control registers ⁽²⁾	Section 4. System Configuration and Section 11. Memory Expansion and Chip Selects
Port H bits 7–4 ⁽¹⁾ bits 3–0	Chip-select lines and pulse-width modulator	Control registers	Section 9. Timing System and Section 11. Memory Expansion and Chip Selects

1. Not available on KS devices

2. Control registers can enable these functions only in expanded operating modes.

Table 2-3. Port Signal Summary

Port/Bit	Single-Chip and Bootstrap Modes	Expanded and Special Test Modes
PA0		PA0/IC3
PA1		PA1/IC2
PA2		PA2/IC1
PA3		PA3/OC5/IC4/and-or OC1
PA4		PA4/OC4/and-or OC1
PA5		PA5/OC3/and-or OC1
PA6		PA6OC2/and-or OC1
PA7		PA7/PAI/and-or OC1
PB[7:0]	PB[7:0]	ADDR[15:8]
PC[7:0]	PC[7:0]	DATA[7:0]
PD0		PD0/RxD
PD1		PD1/TxD
PD2		PD2/MISO
PD3		PD3/MOSI
PD4		PD4/SCK
PD5		PD5/ \overline{SS}
PE[7:0]	PE[7:0] PE[7:0]/AN[0:7]	
PF[7:0]	PF[7:0]	ADDR[7:0]
PG0	PG0	PG0/XA13
PG1	PG1	PG1/XA14
PG2	PG2	PG2/XA15
PG3	PG3	PG3/XA16
PG4	PG4	PG4/XA17
PG5	PG5	PG5/XA18
PG6	PG6	PG6
PG7	PG7	PG7/R \overline{W}
PH0		PH0/PW1
PH1		PH1/PW2
PH2		PH2/PW3
PH3		PH3/PW4
PH4	PH4	PH4/CSIO
PH5	PH5	PH5/CSGP1
PH6	PH6	PH6/CSGP2
PH7	PH7	PH7/CSPROG

Section 3. Central Processor Unit (CPU)

3.1 Contents

3.2	Introduction	46
3.3	CPU Registers	46
3.3.1	Accumulators A, B, and D (ACCA, ACCB, and ACCD)	47
3.3.2	Index Register X (IX)	47
3.3.3	Index Register Y (IY)	47
3.3.4	Stack Pointer (SP)	48
3.3.5	Program Counter (PC)	50
3.3.6	Condition Code Register (CCR)	50
3.3.6.1	Carry/Borrow (C)	50
3.3.6.2	Overflow (V)	50
3.3.6.3	Zero (Z)	50
3.3.6.4	Negative (N)	51
3.3.6.5	Interrupt Mask (I)	51
3.3.6.6	Half Carry (H)	51
3.3.6.7	Non-Maskable Interrupt (X)	52
3.3.6.8	Stop Disable (S)	52
3.4	Data Types	52
3.5	Opcodes and Operands	53
3.6	Addressing Modes	53
3.6.1	Immediate	54
3.6.2	Direct	54
3.6.3	Extended	54
3.6.4	Indexed	54
3.6.5	Inherent	55
3.6.6	Relative	55
3.7	Instruction Set	55

3.2 Introduction

This section presents information on TK68HC11 central processor unit (CPU) architecture, data types, addressing modes, the instruction set, and special operations, such as subroutine calls and interrupts.

The CPU employs memory-mapped input/output (I/O). There are no special instructions for I/O; all peripheral, I/O, and memory locations are simply addresses in the 64-Kbyte memory map. This architecture also enables access to operands from external memory locations with no execution time penalty.

3.3 CPU Registers

TK68HC11 CPU registers are an integral part of the CPU and are not addressed as memory locations. The seven registers are shown in **Figure 3-1**.

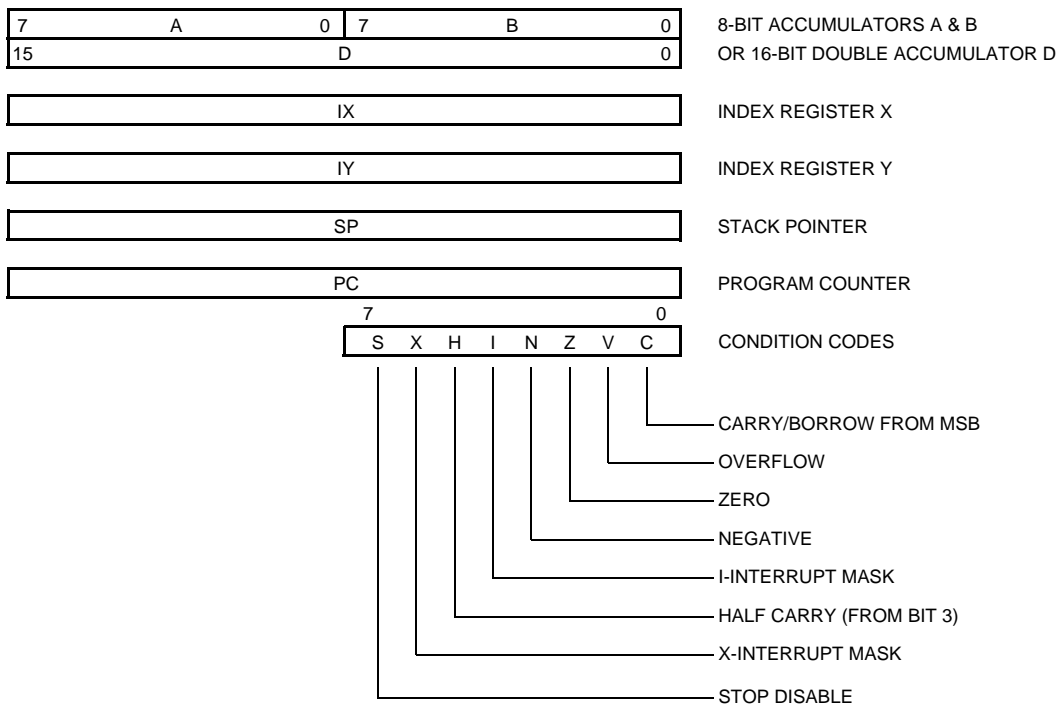


Figure 3-1. Programming Model

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

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

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

3.3.2 Index Register X (IX)

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

3.3.3 Index Register Y (IY)

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

3.3.4 Stack Pointer (SP)

The stack pointer holds the 16-bit address of the next free location in the TK68HC11 CPU's automatic program stack. This stack is a data structure that

grows downward from high memory to low memory. The stack can be located anywhere in the address space and can be any size up to the amount of memory available in the system. Most application programs initialize the SP at the beginning of an application program with a load stack (LDS) instruction. Thereafter, each time the CPU pushes a new byte onto the stack, it decrements the SP. To pull a byte from the stack, the CPU first increments the SP. **Figure 3-2** is a summary of SP operations.

A jump-to-subroutine (JSR) or branch-to-subroutine (BSR) instruction pushes the address of the instruction immediately after the JSR or BSR onto the stack, least significant byte first. The last instruction of the subroutine is a return-from-subroutine (RTS), which pulls the previously stored return address from the stack and loads it into the program counter. Execution then continues at this recovered return address.

When the processor recognizes an interrupt, it finishes the current instruction, pushes the return address (the current value in the program counter) onto the stack, pushes all of the CPU registers onto the stack, and continues at the address specified by the vector for the interrupt. The interrupt service routine ends with a return-from-interrupt (RTI) instruction, which pulls the saved registers off the stack in reverse order. Program execution resumes at the return address with all register contents restored.

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

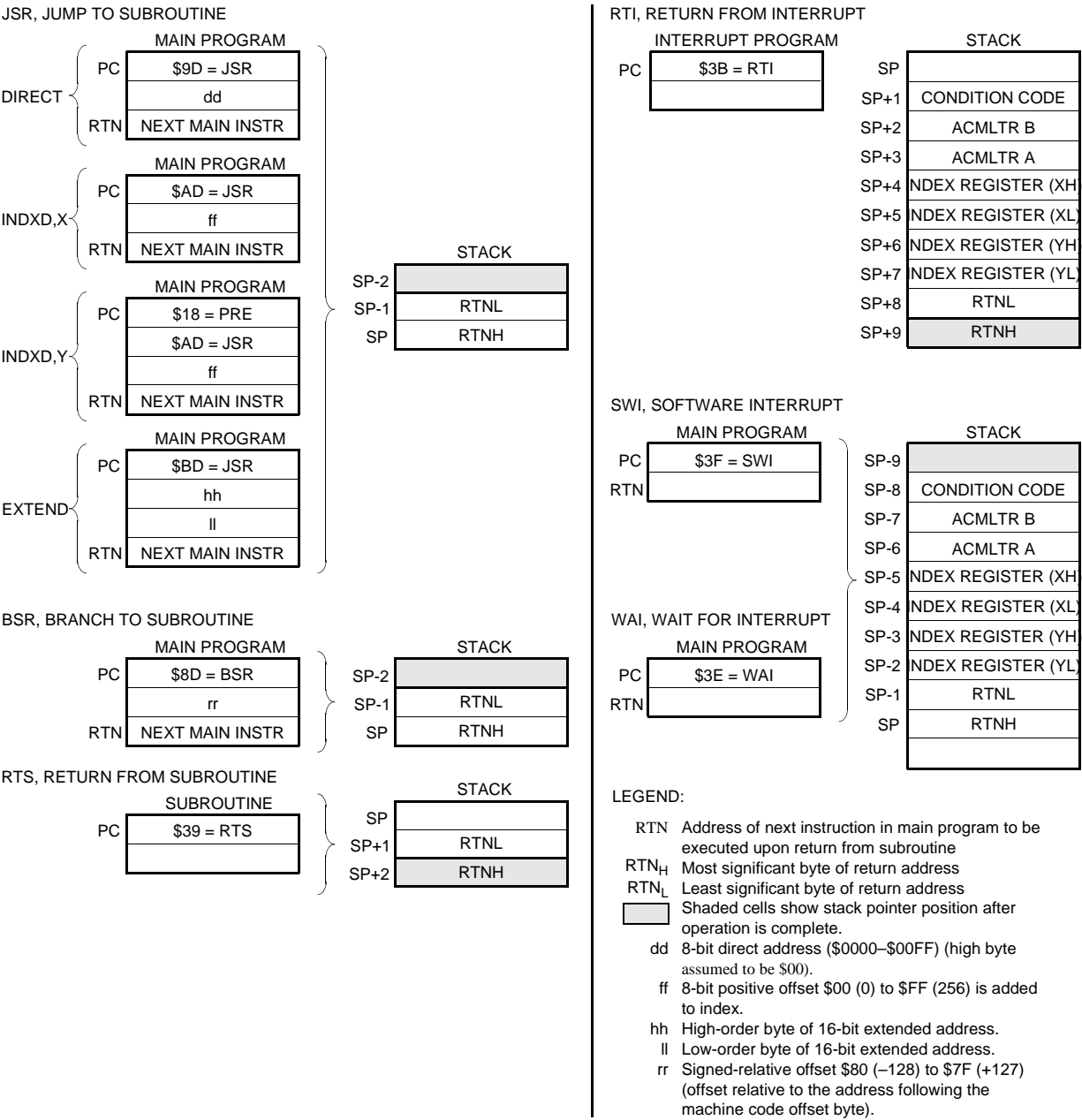


Figure 3-2. Stacking Operations

3.3.5 Program Counter (PC)

The 16-bit program counter contains the address of the next instruction to be executed. Its initial value after reset is fetched from one of six possible vectors, depending on operating mode and the cause of reset, as described in **5.3 Sources of Resets**.

3.3.6 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)
- A stop disable bit (S)

Most instructions update condition codes automatically, as described in the following paragraphs. Certain instructions, such as pushes, pulls, add B to X (ABX), add B to Y (ABY), and transfer/exchange instructions do not affect the condition codes. **Table 3-1** shows which condition codes are affected by each instruction.

3.3.6.1 Carry/Borrow (C)

The C bit is set if the CPU performs a carry or borrow during an arithmetic operation. This 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.

3.3.6.2 Overflow (V)

The overflow bit is set if an operation results in a two's complement overflow of the 8-bit signed range -128 to $+127$. Otherwise, the V bit is cleared.

3.3.6.3 Zero (Z)

The Z bit is set if the result of an arithmetic, logic, or data manipulation operation is 0. 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.

3.3.6.4 Negative (N)

The N bit is set if the result of an arithmetic, logic, or data manipulation operation is negative, meaning that the most significant bit (MSB) of the result is a 1. Otherwise, the N bit is cleared. To determine quickly if the MSB of a particular byte is set, load it into an accumulator and then check the status of the N bit.

3.3.6.5 Interrupt Mask (I)

When the interrupt mask bit is set, it disables all maskable interrupt requests (IRQs). The CPU continues to operate uninterrupted while interrupts remain pending until the I bit is cleared. Every reset sets the I bit by default and only a software instruction can clear it. When the processor recognizes an interrupt, it stacks the registers, sets the I bit, and then fetches the interrupt vector. The final instruction of an interrupt service routine is usually a return from interrupt (RTI), which restores the registers to the values that were present before the interrupt occurred and clears the I bit.

NOTE: *Although the I bit can be cleared earlier in the interrupt service routine, avoid nesting interrupts in this way without a clear understanding of latency and of the arbitration mechanism.*

Refer to **Section 5. Resets and Interrupts**.

3.3.6.6 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 binary-coded decimal (BCD) operations.

3.3.6.7 Non-Maskable Interrupt (X)

Setting the XIRQ mask (X) bit disables non-maskable interrupts from the $\overline{\text{XIRQ}}$ pin. Every reset sets the X bit by default and only a software instruction can clear it. When the processor recognizes a non-maskable interrupt, it stacks the registers, sets the X and I bits, and then fetches the interrupt vector. An interrupt service routine usually ends with a return from interrupt (RTI), which restores the registers to the values that were present before the interrupt occurred and clears the X bit. Only hardware or an acknowledge can set the X bit. Only software can clear the X bit (for example, the TAP instruction which transfers data from accumulator A to the condition code register). There is no hardware action for clearing X.

3.3.6.8 Stop Disable (S)

Setting the STOP disable (S) bit prevents the STOP instruction from putting the TK68HC11 into a low-power stop condition. If the S bit is set, the CPU treats a STOP instruction as if it were a no-operation (NOP) instruction and continues to the next instruction.

NOTE: *S is set by reset and STOP is disabled by default.*

The STOP instruction can be cleared by using the TAP instruction which transfers data from accumulator A to the condition code register.

3.4 Data Types

The TK68HC11 CPU supports these 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 TK68HC11 is an 8-bit CPU, there are no special requirements for alignment of instructions or operands.

3.5 Opcodes and Operands

The TK68HC11 Family of microcontrollers uses 8-bit opcodes. Every instruction requires a unique opcode for each of its addressing modes. The resulting number of opcodes exceeds the 256 available in an 8-bit binary number. A 4-page opcode map has been implemented to accommodate the extra 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 to three operands. The operands contain information the CPU needs for executing the instruction. Complete instructions can be from one to five bytes long.

3.6 Addressing Modes

Six addressing modes can be used to access memory:

1. Immediate
2. Direct
3. Extended
4. Indexed
5. Inherent
6. Relative

All modes except inherent mode use an effective address. The effective address is the memory address where 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.

3.6.1 Immediate

In the immediate addressing mode, the byte(s) immediately following the opcode contain the arguments. The number of bytes following the opcode matches the size of the register or memory location being used. Immediate instructions can be two, three, or (if a prebyte is required) four bytes.

3.6.2 Direct

In the direct addressing mode, the user specifies only the low-order byte of the effective address in a single byte following the opcode. The processor assumes the high-order byte of the address to be \$00. Thus, the CPU accesses addresses \$00–\$FF directly, using 2-byte instructions. This reduces execution time by eliminating the additional memory access required for the high-order address byte. Most applications reserve this 256-byte area for frequently referenced data, but various combinations of internal registers, RAM, or external memory can occupy these addresses.

3.6.3 Extended

In the extended addressing mode, the two bytes following the opcode byte contain the effective address of the argument. For this reason, instructions are three bytes, or they are four bytes if a prebyte is required.

3.6.4 Indexed

In the indexed addressing mode, the CPU computes the effective address of the argument by adding an 8-bit unsigned offset to the value contained in an index register (IX or IY). Any memory location in the 64-Kbyte address space can be accessed with this mode. The instructions are from two to five bytes.

3.6.5 Inherent

In the inherent addressing mode, the opcode contains all required information. The operands (if any) are registers, so no memory access is required. This mode includes:

- Control instructions with no arguments
- Operations that only involve the index registers or accumulators

These instructions are one or two bytes.

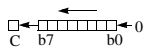
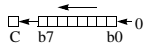
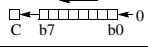
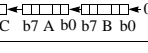
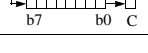
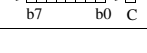
3.6.6 Relative

Only branch instructions use the relative addressing mode. If the branch condition is true, the CPU adds the 8-bit signed offset following the opcode to the contents of the program counter to form the effective branch address. Otherwise, control proceeds to the next instruction. These are usually 2-byte instructions.

3.7 Instruction Set

Table 3-1 presents a detailed listing of all the TK68HC11 instructions in all possible addressing modes.

Table 3-1. Instruction Set (Sheet 1 of 8)

Mnemonic	Operation	Description	Addressing Mode	Instruction			Condition Codes							
				Opcode	Operand	Cycles	S	X	H	I	N	Z	V	C
ABA	Add Accumulators	$A + B \Rightarrow A$	INH	1B	—	2	—	—	D	—	D	D	D	D
ABX	Add B to X	$IX + B \Rightarrow IX$	INH	3A	—	3	—	—	—	—	—	—	—	—
ABY	Add B to Y	$IY + B \Rightarrow IY$	INH	18 3A	—	4	—	—	—	—	—	—	—	—
ADCA (opr)	Add with Carry to A	$A + M + C \Rightarrow A$	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	—	—	D	—	D	D	D	D
ADCB (opr)	Add with Carry to B	$B + M + C \Rightarrow B$	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	—	—	D	—	D	D	D	D
ADDA (opr)	Add Memory to A	$A + M \Rightarrow A$	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	—	—	D	—	D	D	D	D
ADDB (opr)	Add Memory to B	$B + M \Rightarrow B$	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	—	—	D	—	D	D	D	D
ADDD (opr)	Add 16-Bit to D	$D + (M : M + 1) \Rightarrow D$	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	—	—	—	—	D	D	D	D
ANDA (opr)	AND A with Memory	$A \cdot M \Rightarrow A$	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	—	—	—	—	D	D	0	—
ANDB (opr)	AND B with Memory	$B \cdot M \Rightarrow B$	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	—	—	—	—	D	D	0	—
ASL (opr)	Arithmetic Shift Left		EXT IND,X IND,Y	78 68 18 68	hh ll ff ff	6 6 7	—	—	—	—	D	D	D	D
ASLA	Arithmetic Shift Left A		A INH	48	—	2	—	—	—	—	D	D	D	D
ASLB	Arithmetic Shift Left B		B INH	58	—	2	—	—	—	—	D	D	D	D
ASLD	Arithmetic Shift Left D		INH	05	—	3	—	—	—	—	D	D	D	D
ASR	Arithmetic Shift Right		EXT IND,X IND,Y	77 67 18 67	hh ll ff ff	6 6 7	—	—	—	—	D	D	D	D
ASRA	Arithmetic Shift Right A		A INH	47	—	2	—	—	—	—	D	D	D	D

Central Processor Unit (CPU)

Table 3-1. Instruction Set (Sheet 2 of 8)

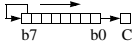
Mnemonic	Operation	Description	Addressing Mode	Instruction			Condition Codes							
				Opcode	Operand	Cycles	S	X	H	I	N	Z	V	C
ASRB	Arithmetic Shift Right B		B INH	57	—	2	—	—	—	—	D	D	D	D
BCC (rel)	Branch if Carry Clear	? C = 0	REL	24	rr	3	—	—	—	—	—	—	—	—
BCLR (opr) (msk)	Clear Bit(s)	M • (mm) ⇒ M	DIR IND,X IND,Y	15 1D 18 1D	dd mm ff mm ff mm	6 7 8	—	—	—	—	D	D	0	—
BCS (rel)	Branch if Carry Set	? C = 1	REL	25	rr	3	—	—	—	—	—	—	—	—
BEQ (rel)	Branch if = Zero	? Z = 1	REL	27	rr	3	—	—	—	—	—	—	—	—
BGE (rel)	Branch if Δ 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 (rel)	22	rr	3	—	—	—	—	—	—	—	—
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	85	ii	2	—	—	—	—	D	D	0	—
			A DIR	95	dd	3	—	—	—	—	—	—	—	—
			A EXT	B5	hh ll	4	—	—	—	—	—	—	—	—
			A IND,X	A5	ff	4	—	—	—	—	—	—	—	—
			A IND,Y	18 A5	ff	5	—	—	—	—	—	—	—	—
BITB (opr)	Bit(s) Test B with Memory	B • M	B IMM	C5	ii	2	—	—	—	—	D	D	0	—
			B DIR	D5	dd	3	—	—	—	—	—	—	—	—
			B EXT	F5	hh ll	4	—	—	—	—	—	—	—	—
			B IND,X	E5	ff	4	—	—	—	—	—	—	—	—
			B IND,Y	18 E5	ff	5	—	—	—	—	—	—	—	—
BLE (rel)	Branch if Δ 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 not = 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	13	dd mm rr	6	—	—	—	—	—	—	—	—
			IND,X	1F	ff mm rr	7	—	—	—	—	—	—	—	
			IND,Y	18 1F	ff mm rr	8	—	—	—	—	—	—	—	
BRN (rel)	Branch Never	? 1 = 0	REL	21	rr	3	—	—	—	—	—	—		
BRSET (opr) (msk) (rel)	Branch if Bit(s) Set	? (M̄) • mm = 0	DIR	12	dd mm rr	6	—	—	—	—	—	—	—	
			IND,X	1E	ff mm rr	7	—	—	—	—	—	—	—	
			IND,Y	18 1E	ff mm rr	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	—	—	—	—	D	D	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 to B	A – B	INH	11	—	2	—	—	—	—	D	D	D	D

Table 3-1. Instruction Set (Sheet 3 of 8)



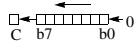
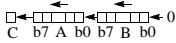
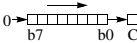
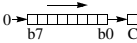
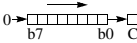
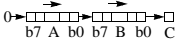
Mnemonic	Operation	Description	Addressing Mode	Instruction			Condition Codes							
				Opcode	Operand	Cycles	S	X	H	I	N	Z	V	C
CLC	Clear Carry Bit	0 ⇒ C	INH	0C	—	2	—	—	—	—	—	—	—	0
CLI	Clear Interrupt Mask	0 ⇒ I	INH	0E	—	2	—	—	—	0	—	—	—	—
CLR (opr)	Clear Memory Byte	0 ⇒ M	EXT IND,X IND,Y	7F 6F 6F	hh ll 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 to 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	—	—	—	—	D	D	D	D
CMPB (opr)	Compare B to 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	—	—	—	—	D	D	D	D
COM (opr)	One's Complement Memory Byte	\$FF – M ⇒ M	EXT IND,X IND,Y	73 63 18 63	hh ll ff ff	6 6 7	—	—	—	—	D	D	0	1
COMA	One's Complement A	\$FF – A ⇒ A	A INH	43	—	2	—	—	—	—	D	D	0	1
COMB	One's Complement B	\$FF – B ⇒ B	B INH	53	—	2	—	—	—	—	D	D	0	1
CPD (opr)	Compare D to 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	—	—	—	—	D	D	D	D
CPX (opr)	Compare X to 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	—	—	—	—	D	D	D	D
CPY (opr)	Compare Y to 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	—	—	—	—	D	D	D	D
DAA	Decimal Adjust A	Adjust Sum to BCD	INH	19	—	2	—	—	—	—	D	D	D	D
DEC (opr)	Decrement Memory Byte	M – 1 ⇒ M	EXT IND,X IND,Y	7A 6A 18 6A	hh ll ff ff	6 6 7	—	—	—	—	D	D	D	—
DECA	Decrement Accumulator A	A – 1 ⇒ A	A INH	4A	—	2	—	—	—	—	D	D	D	—
DECB	Decrement Accumulator B	B – 1 ⇒ B	B INH	5A	—	2	—	—	—	—	D	D	D	—
DES	Decrement Stack Pointer	SP – 1 ⇒ SP	INH	34	—	3	—	—	—	—	—	—	—	—

Central Processor Unit (CPU)

Table 3-1. Instruction Set (Sheet 4 of 8)

Mnemonic	Operation	Description	Addressing Mode	Instruction			Condition Codes								
				Opcode	Operand	Cycles	S	X	H	I	N	Z	V	C	
DEX	Decrement Index Register X	$IX - 1 \Rightarrow IX$	INH	09	—	3	—	—	—	—	—	—	D	—	—
DEY	Decrement Index Register Y	$IY - 1 \Rightarrow IY$	INH	18 09	—	4	—	—	—	—	—	—	D	—	—
EORA (opr)	Exclusive OR A with Memory	$A \oplus M \Rightarrow A$	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	—	—	—	—	—	—	D	D	0 —
EORB (opr)	Exclusive OR B with Memory	$B \oplus M \Rightarrow B$	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	—	—	—	—	—	—	D	D	0 —
FDIV	Fractional Divide 16 by 16	$D / IX \Rightarrow IX; r \Rightarrow D$	INH	03	—	41	—	—	—	—	—	—	D	D	D
IDIV	Integer Divide 16 by 16	$D / IX \Rightarrow IX; r \Rightarrow D$	INH	02	—	41	—	—	—	—	—	—	D	0	D
INC (opr)	Increment Memory Byte	$M + 1 \Rightarrow M$	EXT IND,X IND,Y	7C 6C 18 6C	hh ll ff ff	6 6 7	—	—	—	—	—	—	D	D	D —
INCA	Increment Accumulator A	$A + 1 \Rightarrow A$	A INH	4C	—	2	—	—	—	—	—	—	D	D	D —
INCB	Increment Accumulator B	$B + 1 \Rightarrow B$	B INH	5C	—	2	—	—	—	—	—	—	D	D	D —
INS	Increment Stack Pointer	$SP + 1 \Rightarrow SP$	INH	31	—	3	—	—	—	—	—	—	—	—	—
INX	Increment Index Register X	$IX + 1 \Rightarrow IX$	INH	08	—	3	—	—	—	—	—	—	D	—	—
INY	Increment Index Register Y	$IY + 1 \Rightarrow IY$	INH	18 08	—	4	—	—	—	—	—	—	D	—	—
JMP (opr)	Jump	Effective Address \Rightarrow PC	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	—	—	—	—	—	—	—	—	—
LDA (opr)	Load Accumulator A	$M \Rightarrow A$	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	—	—	—	—	—	—	D	D	0 —
LDAB (opr)	Load Accumulator B	$M \Rightarrow B$	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	—	—	—	—	—	—	D	D	0 —
LDD (opr)	Load Double Accumulator D	$M \Rightarrow A, M + 1 \Rightarrow B$	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	—	—	—	—	—	—	D	D	0 —

Table 3-1. Instruction Set (Sheet 5 of 8)

Mnemonic	Operation	Description	Addressing Mode	Instruction			Condition Codes							
				Opcode	Operand	Cycles	S	X	H	I	N	Z	V	C
LDS (opr)	Load Stack Pointer	$M : M + 1 \Rightarrow SP$	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	—	—	—	—	D	D	0	—
LDX (opr)	Load Index Register X	$M : M + 1 \Rightarrow IX$	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	—	—	—	—	D	D	0	—
LDY (opr)	Load Index Register Y	$M : M + 1 \Rightarrow IY$	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	—	—	—	—	D	D	0	—
LSL (opr)	Logical Shift Left		EXT IND,X IND,Y	78 68 18 68	hh ll ff ff	6 6 7	—	—	—	—	D	D	D	D
LSLA	Logical Shift Left A		A INH	48	—	2	—	—	—	—	D	D	D	D
LSLB	Logical Shift Left B		B INH	58	—	2	—	—	—	—	D	D	D	D
LSLD	Logical Shift Left Double		INH	05	—	3	—	—	—	—	D	D	D	D
LSR (opr)	Logical Shift Right		EXT IND,X IND,Y	74 64 18 64	hh ll ff ff	6 6 7	—	—	—	—	0	D	D	D
LSRA	Logical Shift Right A		A INH	44	—	2	—	—	—	—	0	D	D	D
LSRB	Logical Shift Right B		B INH	54	—	2	—	—	—	—	0	D	D	D
LSRD	Logical Shift Right Double		INH	04	—	3	—	—	—	—	0	D	D	D
MUL	Multiply 8 by 8	$A * B \Rightarrow D$	INH	3D	—	10	—	—	—	—	—	—	—	D
NEG (opr)	Two's Complement Memory Byte	$0 - M \Rightarrow M$	EXT IND,X IND,Y	70 60 18 60	hh ll ff ff	6 6 7	—	—	—	—	D	D	D	D
NEGA	Two's Complement A	$0 - A \Rightarrow A$	A INH	40	—	2	—	—	—	—	D	D	D	D
NEGB	Two's Complement B	$0 - B \Rightarrow B$	B INH	50	—	2	—	—	—	—	D	D	D	D
NOP	No operation	No Operation	INH	01	—	2	—	—	—	—	—	—	—	—
ORAA (opr)	OR Accumulator A (Inclusive)	$A + M \Rightarrow A$	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	—	—	—	—	D	D	0	—

Central Processor Unit (CPU)

Table 3-1. Instruction Set (Sheet 6 of 8)

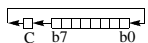
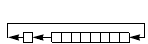
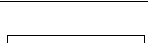
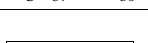
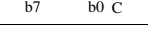
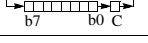
Mnemonic	Operation	Description	Addressing Mode	Instruction			Condition Codes							
				Opcode	Operand	Cycles	S	X	H	I	N	Z	V	C
ORAB (opr)	OR Accumulator B (Inclusive)	$B + M \Rightarrow B$	B IMM	CA	ii	2	—	—	—	—	D	D	0	—
			B DIR	DA	dd	3								
			B EXT	FA	hh ll	4								
			B IND,X	EA	ff	4								
			B IND,Y	EA	ff	5								
PSHA	Push A onto Stack	$A \Rightarrow \text{Stk};$ $SP = SP - 1$	A INH	36	—	3	—	—	—	—	—	—	—	—
PSHB	Push B onto Stack	$B \Rightarrow \text{Stk};$ $SP = SP - 1$	B INH	37	—	3	—	—	—	—	—	—	—	—
PSHX	Push X onto Stack (Lo First)	$IX \Rightarrow \text{Stk};$ $SP = SP - 2$	INH	3C	—	4	—	—	—	—	—	—	—	—
PSHY	Push Y onto Stack (Lo First)	$IY \Rightarrow \text{Stk};$ $SP = SP - 2$	INH	18 3C	—	5	—	—	—	—	—	—	—	—
PULA	Pull A from Stack	$SP = SP + 1;$ $A \Leftarrow \text{Stk}$	A INH	32	—	4	—	—	—	—	—	—	—	—
PULB	Pull B from Stack	$SP = SP + 1;$ $B \Leftarrow \text{Stk}$	B INH	33	—	4	—	—	—	—	—	—	—	—
PULX	Pull X From Stack (Hi First)	$SP = SP + 2;$ $IX \Leftarrow \text{Stk}$	INH	38	—	5	—	—	—	—	—	—	—	—
PULY	Pull Y from Stack (Hi First)	$SP = SP + 2;$ $IY \Leftarrow \text{Stk}$	INH	18 38	—	6	—	—	—	—	—	—	—	—
ROL (opr)	Rotate Left		EXT	79	hh ll	6	—	—	—	—	D	D	D	D
			IND,X	69	ff	6								
			IND,Y	18 69	ff	7								
ROLA	Rotate Left A		A INH	49	—	2	—	—	—	—	D	D	D	D
ROLB	Rotate Left B		B INH	59	—	2	—	—	—	—	D	D	D	D
ROR (opr)	Rotate Right		EXT	76	hh ll	6	—	—	—	—	D	D	D	D
			IND,X	66	ff	6								
			IND,Y	18 66	ff	7								
RORA	Rotate Right A		A INH	46	—	2	—	—	—	—	D	D	D	D
RORB	Rotate Right B		B INH	56	—	2	—	—	—	—	D	D	D	D
RTI	Return from Interrupt	See Figure 3-2	INH	3B	—	12	D	∅	D	D	D	D	D	D
RTS	Return from Subroutine	See Figure 3-2	INH	39	—	5	—	—	—	—	—	—	—	—
SBA	Subtract B from A	$A - B \Rightarrow A$	INH	10	—	2	—	—	—	—	D	D	D	D
SBCA (opr)	Subtract with Carry from A	$A - M - C \Rightarrow A$	A IMM	82	ii	2	—	—	—	—	D	D	D	D
			A DIR	92	dd	3								
			A EXT	B2	hh ll	4								
			A IND,X	A2	ff	4								
			A IND,Y	18 A2	ff	5								

Table 3-1. Instruction Set (Sheet 7 of 8)

Mnemonic	Operation	Description	Addressing Mode	Instruction			Condition Codes							
				Opcode	Operand	Cycles	S	X	H	I	N	Z	V	C
SBCB (opr)	Subtract with Carry from B	$B - M - C \Rightarrow B$	B IMM	C2	ii	2	—	—	—	—	D	D	D	D
			B DIR	D2	dd	3								
			B EXT	F2	hh ll	4								
			B IND,X	E2	ff	4								
			B IND,Y	18 E2	ff	5								
SEC	Set Carry	$1 \Rightarrow C$	INH	0D	—	2	—	—	—	—	—	—	1	
SEI	Set Interrupt Mask	$1 \Rightarrow I$	INH	0F	—	2	—	—	—	1	—	—	—	
SEV	Set Overflow Flag	$1 \Rightarrow V$	INH	0B	—	2	—	—	—	—	—	1	—	
STAA (opr)	Store Accumulator A	$A \Rightarrow M$	A DIR	97	dd	3	—	—	—	—	D	D	0	—
			A EXT	B7	hh ll	4								
			A IND,X	A7	ff	4								
			A IND,Y	18 A7	ff	5								
STAB (opr)	Store Accumulator B	$B \Rightarrow M$	B DIR	D7	dd	3	—	—	—	—	D	D	0	—
			B EXT	F7	hh ll	4								
			B IND,X	E7	ff	4								
			B IND,Y	18 E7	ff	5								
STD (opr)	Store Accumulator D	$A \Rightarrow M, B \Rightarrow M + 1$	DIR	DD	dd	4	—	—	—	—	D	D	0	—
			EXT	FD	hh ll	5								
			IND,X	ED	ff	5								
			IND,Y	18 ED	ff	6								
STOP	Stop Internal Clocks	—	INH	CF	—	2	—	—	—	—	—	—		
STS (opr)	Store Stack Pointer	$SP \Rightarrow M : M + 1$	DIR	9F	dd	4	—	—	—	—	D	D	0	—
			EXT	BF	hh ll	5								
			IND,X	AF	ff	5								
			IND,Y	18 AF	ff	6								
STX (opr)	Store Index Register X	$IX \Rightarrow M : M + 1$	DIR	DF	dd	4	—	—	—	—	D	D	0	—
			EXT	FF	hh ll	5								
			IND,X	EF	ff	5								
			IND,Y	CD EF	ff	6								
STY (opr)	Store Index Register Y	$IY \Rightarrow M : M + 1$	DIR	18 DF	dd	5	—	—	—	—	D	D	0	—
			EXT	18 FF	hh ll	6								
			IND,X	1A EF	ff	6								
			IND,Y	18 EF	ff	6								
SUBA (opr)	Subtract Memory from A	$A - M \Rightarrow A$	A IMM	80	ii	2	—	—	—	—	D	D	D	D
			A DIR	90	dd	3								
			A EXT	B0	hh ll	4								
			A IND,X	A0	ff	4								
			A IND,Y	18 A0	ff	5								
SUBB (opr)	Subtract Memory from B	$B - M \Rightarrow B$	A IMM	C0	ii	2	—	—	—	—	D	D	D	D
			A DIR	D0	dd	3								
			A EXT	F0	hh ll	4								
			A IND,X	E0	ff	4								
			A IND,Y	18 E0	ff	5								
SUBD (opr)	Subtract Memory from D	$D - M : M + 1 \Rightarrow D$	IMM	83	jj kk	4	—	—	—	—	D	D	D	D
			DIR	93	dd	5								
			EXT	B3	hh ll	6								
			IND,X	A3	ff	6								
			IND,Y	18 A3	ff	7								
SWI	Software Interrupt	See Figure 3-2	INH	3F	—	14	—	—	—	1	—	—	—	
TAB	Transfer A to B	$A \Rightarrow B$	INH	16	—	2	—	—	—	—	D	D	0	—
TAP	Transfer A to CC Register	$A \Rightarrow CCR$	INH	06	—	2	D	∅	D	D	D	D	D	
TBA	Transfer B to A	$B \Rightarrow A$	INH	17	—	2	—	—	—	—	D	D	0	—
TEST	TEST (Only in Test Modes)	Address Bus Counts	INH	00	—	*	—	—	—	—	—	—	—	

Central Processor Unit (CPU)

Table 3-1. Instruction Set (Sheet 8 of 8)

Mnemonic	Operation	Description	Addressing Mode	Instruction			Condition Codes									
				Opcode	Operand	Cycles	S	X	H	I	N	Z	V	C		
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	—	—	—	—	D	D	0	0		
TSTA	Test A for Zero or Minus	A – 0	A INH	4D	—	2	—	—	—	—	D	D	0	0		
TSTB	Test B for Zero or Minus	B – 0	B INH	5D	—	2	—	—	—	—	D	D	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 Regs & WAIT	INH	3E	—	**	—	—	—	—	—	—	—	—	—	—
XGDY	Exchange D with X	IX ⇒ D, D ⇒ IX	INH	8F	—	3	—	—	—	—	—	—	—	—	—	—
XGDY	Exchange D with Y	IY ⇒ D, D ⇒ IY	INH	18 8F	—	4	—	—	—	—	—	—	—	—	—	—

Cycle:

* Infinity 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 recognized. Finally, two additional cycles are used to fetch the appropriate interrupt vector (14 + n total).

Operands:

dd= 8-bit direct address (\$0000 –\$00FF) (high byte assumed to be \$00)

ff= 8-bit positive offset \$00 (0) to \$FF (255) (is added to index)

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 (-128) to \$7F (+127)

(offset relative to address following machine code offset byte)

Operators:

() Contents of register shown inside parentheses

⇐ Is transferred to

• Boolean AND

+ Arithmetic addition symbol except where used as inclusive-OR symbol in Boolean formula

⊕ Exclusive-OR

* Multiply

: Concatenation

– Arithmetic subtraction symbol or negation symbol (two's complement)

Condition Codes:

— Bit not changed

0 Bit always cleared

1 Bit always set

Δ Bit cleared or set, depending on operation

↓ Bit can be cleared, cannot become set

Section 4. System Configuration

4.1 Contents

4.2	Introduction	66
4.3	Control Registers	66
4.4	Operating Modes	78
4.4.1	Single-Chip Mode	78
4.4.2	Expanded Mode	78
4.4.3	Bootstrap Mode	79
4.4.4	Special Test Mode	80
4.4.5	Mode Selection	80
4.5	Memory Map	82
4.5.1	Control Registers and RAM	84
4.5.2	ROM or EPROM	86
4.5.3	EEPROM	86
4.5.4	Bootloader ROM	89
4.6	EPROM/OTPRM (TK68HC711K4, TK68HC711KS8, and TK68HC711KS2)	89
4.6.1	Programming the EPROM with Downloaded Data	89
4.6.2	Programming the EPROM from Memory	90
4.7	EEPROM and the CONFIG Register	92
4.7.1	EEPROM Registers	92
4.7.1.1	EEPROM Programming Control Register	93
4.7.1.2	Block Protect Register	95
4.7.1.3	System Configuration Options Register	96
4.7.2	EEPROM Programming	96
4.7.2.1	EEPROM Programming	97
4.7.2.2	EEPROM Bulk Erase	97
4.7.2.3	EEPROM Row Erase	98
4.7.2.4	EEPROM Byte Erase	98
4.7.3	CONFIG Register Programming	98
4.7.4	RAM and EEPROM Security	99

4.2 Introduction

This section presents the elements involved in configuring the TK68HC11K/KS Family microcontrollers (MCUs), including:

- A list of the control registers
- Description of the four operating modes and how they're selected
- Various memory maps in the K Family
- Information on programming EPROM (erasable, programmable read-only memory) and EEPROM (electrically erasable, programmable read-only memory)

4.3 Control Registers

The heart of the TK68HC11 Family of MCUs is a special register block which controls most of the peripheral functions. In the K Family, this block is 128 bytes. The default location of this block is the first 128 bytes of memory, but software can map it to any 4-Kbyte boundary (see **4.5.1 Control Registers and RAM**).

Certain bits that control initialization and the basic operation of the MCU are protected against writes in normal operating modes except under special circumstances. Some bits cannot be written at all; others can be written only once and/or within the first 64 bus cycles after any reset. The special operating modes override these restrictions.

Normal and special operating modes are discussed in **4.3 Control Registers**. The write-restricted registers and bits are summarized in **Table 4-1**.

Figure 4-1 lists the entire 128-byte register block in ascending order by address, using the default memory block assignment \$0000–\$007F.

NOTE: *Throughout this manual, the registers are discussed by function. In the event that not all bits in a register are referenced, the bits that are not discussed are shaded.*

Table 4-1. Write Access-Limited Registers

Register Address	Register Name	Affected Bits	Write Restrictions	See page
\$x024	TKSK2	[1:0]	Within 64 cycles, once only	190
\$x035	BPROT	[7:0]	Within 64 cycles ⁽¹⁾ , once only	94
\$x037	INIT2	[7:4]	Once only	87
\$x038		[4]	Once only	39
\$x039	OPTION	[5:4],[1:0]	Within 64 cycles, once only	110, 111, 121
\$x03D	INIT	[7:0]	Within 64 cycles	86

1. Only to clear bits. Bits can be set at any time.

Addr.	Register Name	Bit 7	6	5	4	3	2	1	Bit 0	
\$0000	Port A Data Register (PORTA)	Read:	PA7	PA6	PA5	PA4	PA3	PA2	PA1	PA0
		Write:								
		Reset:	Undefined after reset							
\$0001	Port A Data Direction Register (DDRA)	Read:	DDA7	DDA6	DDA5	DDA4	DDA3	DDA2	DDA1	DDA0
		Write:								
		Reset:	0	0	0	0	0	0	0	0
\$0002	Port B Data Direction Register (DDRB)	Read:	DDB7	DDB6	DDB5	DDB4	DDB3	DDB2	DDB1	DDB0
		Write:								
		Reset:	0	0	0	0	0	0	0	0
\$0003	Port F Data Direction Register (DDRF)	Read:	DDF7	DDF6	DDF5	DDF4	DDF3	DDF2	DDF1	DDF0
		Write:								
		Reset:	0	0	0	0	0	0	0	0
\$0004	Port B Data Register (PORTB)	Read:	PB7	PB6	PB5	PB4	PB3	PB2	PB1	PB0
		Write:								
		Reset:	Undefined after reset							
\$0005	Port F Data Register (PORTF)	Read:	PF7	PF6	PF5	PF4	PF3	PF2	PF1	PF0
		Write:								
		Reset:	Undefined after reset							

= Unimplemented
R = Reserved
U = Undefined

1. Present only in EPROM (711) devices
2. Not available on TK68HC11K4 devices
3. Not available on TK68HC11KS devices

Figure 4-1. Register and Control Bit Assignments (Sheet 1 of 10)

System Configuration

Addr.	Register Name	Bit 7	6	5	4	3	2	1	Bit 0	
\$0006	Port C Data Register (PORTC)	Read:	PC7	PC6	PC5	PC4	PC3	PC2	PC1	PC0
		Write:								
		Reset:	Undefined after reset							
\$0007	Port C Data Direction Register (DDRC)	Read:	0	0	DDC5	DDC4	DDC3	DDC2	DDC1	DDC0
		Write:								
		Reset:	0	0	0	0	0	0	0	0
\$0008	Port D Data Register (PORTD)	Read:	0	0	PD5	PD4	PD3	PD2	PD1	PD0
		Write:								
		Reset:	0	0	U	U	U	U	U	U
\$0009	Port D Data Direction Register (DDRD)	Read:	0	0	DDD5	DDD4	DDD3	DDD2	DDD1	DDD0
		Write:								
		Reset:	0	0	0	0	0	0	0	0
\$000A	Port E Data Register (PORTE)	Read:	PE7	PE6	PE5	PE4	PE3	PE2	PD1	PD0
		Write:								
		Reset:	Undefined after reset							
\$000B	Timer Compare Force Register (CFORC)	Read:	FOC1	FOC2	FOC3	FOC4	FOC5	0	0	0
		Write:								
		Reset:	0	0	0	0	0	0	0	0
\$000C	Output Compare 1 Mask Register (OC1M)	Read:	OC1M7	OC1M6	OC1M5	OC1M4	OC1M3	0	0	0
		Write:								
		Reset:	0	0	0	0	0	0	0	0
\$000D	Output Compare 1 Data Register (OC1D)	Read:	OC1D7	OC1D6	OC1D5	OC1D4	OC1D3	0	0	0
		Write:								
		Reset:	0	0	0	0	0	0	0	0
\$000E	Timer Counter Register High (TCNTH)	Read:	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8
		Write:								
		Reset:	0	0	0	0	0	0	0	0
\$000F	Timer Counter Register Low (TCNTL)	Read:	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
		Write:								
		Reset:	0	0	0	0	0	0	0	0
\$0010	Timer Input Capture 1 Register High (TIC1H)	Read:	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8
		Write:								
		Reset:	Unaffected by reset							

= Unimplemented
 R = Reserved
 U = Undefined

1. Present only in EPROM (711) devices
2. Not available on TK68HC11K4 devices
3. Not available on TK68HC11KS devices

Figure 4-1. Register and Control Bit Assignments (Sheet 2 of 10)

Addr.	Register Name	Bit 7	6	5	4	3	2	1	Bit 0	
\$0011	Timer Input Capture 1 Register Low (TIC1L)	Read:	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
		Write:								
		Reset:	Unaffected by reset							
\$0012	Timer Input Capture 2 Register High (TIC2H)	Read:	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8
		Write:								
		Reset:	Unaffected by reset							
\$0013	Timer Input Capture 2 Register Low (TIC2L)	Read:	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
		Write:								
		Reset:	Unaffected by reset							
\$0014	Timer Input Capture 3 Register High (TIC3H)	Read:	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8
		Write:								
		Reset:	Unaffected by reset							
\$0015	Timer Input Capture 3 Register Low (TIC3L)	Read:	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
		Write:								
		Reset:	Unaffected by reset							
\$0016	Timer Output Compare 1 High Register (TOC1H)	Read:	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8
		Write:								
		Reset:	1	1	1	1	1	1	1	1
\$0017	Timer Output Compare 1 Low Register (TOC1L)	Read:	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
		Write:								
		Reset:	1	1	1	1	1	1	1	1
\$0018	Timer Output Compare 2 High Register (TOC2H)	Read:	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8
		Write:								
		Reset:	1	1	1	1	1	1	1	1
\$0019	Timer Output Compare 2 Low Register (TOC2L)	Read:	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
		Write:								
		Reset:	1	1	1	1	1	1	1	1
\$001A	Timer Output Compare 3 High Register (TOC3H)	Read:	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8
		Write:								
		Reset:	1	1	1	1	1	1	1	1
\$001B	Timer Output Compare 3 Low Register (TOC3L)	Read:	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
		Write:								
		Reset:	1	1	1	1	1	1	1	1
\$001C	Timer Output Compare 4 High Register (TOC4H)	Read:	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8
		Write:								
		Reset:	1	1	1	1	1	1	1	1

= Unimplemented
 = Reserved
 U = Undefined

- 1. Present only in EPROM (711) devices
- 2. Not available on TK68HC11K4 devices
- 3. Not available on TK68HC11KS devices

Figure 4-1. Register and Control Bit Assignments (Sheet 3 of 10)

System Configuration

Addr.	Register Name	Bit 7	6	5	4	3	2	1	Bit 0	
\$001D	Timer Output Compare 4 Low Register (TOC4L)	Read:	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
		Write:								
		Reset:	1	1	1	1	1	1	1	1
\$001E	Timer Input Capture 4/ Output Compare 5 Reg. High (TI4H/O5H)	Read:	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8
		Write:								
		Reset:	1	1	1	1	1	1	1	1
\$001F	Timer Input Capture 4/ Output Compare 5 Reg. Low (TI4L/O5L)	Read:	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
		Write:								
		Reset:	1	1	1	1	1	1	1	1
\$0020	Timer Control 1 Register (TCTL1)	Read:	OM2	OL2	OM3	OL3	OM4	OL4	OM5	OL5
		Write:								
		Reset:	0	0	0	0	0	0	0	0
\$0021	Timer Control 2 Register (TCTL2)	Read:	EDG4B	EDG4A	EDG1B	EDG1A	EDG2B	EDG2A	EDG3B	EDG3A
		Write:								
		Reset:	0	0	0	0	0	0	0	0
\$0022	Timer Interrupt Mask 1 Register (TKSK1)	Read:	OC1I	OC2I	OC3I	OC4I	I4/O5I	IC1I	IC2I	IC3I
		Write:								
		Reset:	0	0	0	0	0	0	0	0
\$0023	Timer Interrupt Flag 1 Register (TFLG1)	Read:	OC1F	OC2F	OC3F	OC4F	I4/O5F	IC1F	IC2F	IC3F
		Write:								
		Reset:	0	0	0	0	0	0	0	0
\$0024	Timer Interrupt Mask 2 Register (TKSK2)	Read:	TOI	RTII	PAOVI	PAII	0	0	PR1	PR0
		Write:								
		Reset:	0	0	0	0	0	0	0	0
\$0025	Timer Interrupt Flag 2 (TFLG2)	Read:	TOF	RTIF	PAOVF	PAIF	0	0	0	0
		Write:								
		Reset:	0	0	0	0	0	0	0	0
\$0026	Pulse Accumulator Control Register (PACTL)	Read:	0	PAEN	PAMOD	PEDGE	0	I4/O5	RTR1	RTR0
		Write:								
		Reset:	0	0	0	0	0	0	0	0
\$0027	Pulse Accumulator Count Register (PACNT)	Read:	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
		Write:								
		Reset:	0	0	0	0	0	0	0	0
\$0028	Serial Peripheral Control Register (SPCR)	Read:	SPIE	SPE	DWOM	MSTR	CPOL	CPHA	SPR1	SPR2
		Write:								
		Reset:	0	0	0	0	0	1	U	U

 = Unimplemented
 R = Reserved
 U = Undefined

1. Present only in EPROM (711) devices
2. Not available on TK68HC11K4 devices
3. Not available on TK68HC11KS devices

Figure 4-1. Register and Control Bit Assignments (Sheet 4 of 10)

Addr.	Register Name	Bit 7	6	5	4	3	2	1	Bit 0	
\$0029	Serial Peripheral Status Register (SPSR)	Read:	SPIF	WCOL	0	MODF	0	0	0	0
		Write:								
		Reset:	0	0	0	0	0	0	0	0
\$002A	SPI Data Register (SPDR)	Read:	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
		Write:								
		Reset:	0	0	0	0	0	0	0	0
\$002B	EPROM Programming Control Register (EPROG) ⁽¹⁾	Read:	MBE	0	ELAT	EXCOL	EXROW	0	0	EPGM
		Write:								
		Reset:	0	0	0	0	0	0	0	0
\$002C	Port Pullup Assignment Register (PPAR)	Read:	0	0	0	0	HPPUE	GPPUE	FPPUE	BPPUE
		Write:								
		Reset:	0	0	0	0	1	1	1	1
\$002D	Port G Assignment Register (PGAR)	Read:	—	—	PGAR5	PGAR4	PGAR3	PGAR2	PGAR1	PGAR0
		Write:								
		Reset:	0	0	0	0	0	0	0	0
\$002E	System Configuration Options 3 Register (OPT3) ⁽²⁾	Read:		SM						
		Write:								
		Reset:	0	0	0	0	0	0	0	0
\$002F	Reserved	R	R	R	R	R	R	R	R	
\$0030	Analog-to-Digital Control/Status Register (ADCTL)	Read:	CCF	—	SCAN	MULT	CD	CC	CB	CA
		Write:								
		Reset:	—	0	—	—	—	—	—	—
\$0031	Analog-to-Digital Results Register 1 (ADR1)	Read:	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
		Write:								
		Reset:	Undefined after reset							
\$0032	Analog-to-Digital Results Register 2 (ADR2)	Read:	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
		Write:								
		Reset:	Undefined after reset							
\$0033	Analog-to-Digital Results Register 3 (ADR3)	Read:	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
		Write:								
		Reset:	Undefined after reset							
\$0034	Analog-to-Digital Results Register 4 (ADR4)	Read:	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
		Write:								
		Reset:	Undefined after reset							

= Unimplemented
 R = Reserved
 U = Undefined

1. Present only in EPROM (711) devices
2. Not available on TK68HC11K4 devices
3. Not available on TK68HC11KS devices

Figure 4-1. Register and Control Bit Assignments (Sheet 5 of 10)

System Configuration

Addr.	Register Name	Bit 7	6	5	4	3	2	1	Bit 0	
\$0035	Block Protect Register (BPROT)	Read:	BULKP	LVPEN	BPRT4	PTCON	BPRT3	BPRT2	BPRT1	BPRT0
		Write:								
		Reset:	1	1	1	1	1	1	1	1
\$0036	Reserved	R	R	R	R	R	R	R	R	
\$0037	EEPROM Mapping Register (INIT2)	Read:	EE3	EE2	EE1	EE0	0	0	0	0
		Write:								
		Reset:	0	0	0	0	0	0	0	0
\$0038	System Configuration Options 2 Register (OPT2)	Read:	LIRDV	CWOM	STRCH⁽³⁾	IRVNE	LSBF	SPR2	XDV1	XDV0
		Write:								
		Reset:	0	0	0	—	0	0	0	0
\$0039	System Configuration Options Register (OPTION)	Read:	ADPU	CSEL	IRQE	DLY	CME	FCME	CR1	CR0
		Write:								
		Reset:	0	0	0	1	0	0	0	0
\$003A	Arm/Reset COP Timer Circuitry Register (COPRST)	Read:	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
		Write:								
		Reset:	0	0	0	0	0	0	0	0
\$003B	EEPROM Programming Control Register (PPROG)	Read:	ODD	EVEN	LVPI	BYTE	ROW	ERASE	EELAT	EEPGM
		Write:								
		Reset:	0	0	0	0	0	0	0	0
\$003C	Highest Priority I-Bit Interrupt and Misc. Register (HPRIO)	Read:	RBOOT	SMOD	MDA	PSEL4	PSEL3	PSEL2	PSEL1	PSEL0
		Write:								
		Reset:	—	—	—	0	0	1	1	0
\$003D	RAM and I/O Mapping Register (INIT)	Read:	RAM3¹⁾	RAM2	RAM1	RAM0	REG3	REG2	REG1	REG0
		Write:								
		Reset:	0	0	0	0	0	0	0	0
\$003E	Test 1 Register (TEST1)	Read:	TILOP	0	OCCR	CBYP	DISR	FCM	FCOP	0
		Write:								
		Reset:	0	0	0	0	0	0	0	0
\$003F	System Configuration Register (CONFIG)	Read:	ROMAD	1	CLKX	PAREN	NOSEC	NOCOP	ROMON	EEON
		Write:								
		Reset:	—	1	—	—	1	—	—	—
\$0040	Reserved	R	R	R	R	R	R	R	R	
to										
\$0055	Reserved		R	R	R	R	R	R	R	
			= Unimplemented			R	= Reserved		U = Undefined	

1. Present only in EPROM (711) devices
2. Not available on TK68HC11K4 devices
3. Not available on TK68HC11KS devices

Figure 4-1. Register and Control Bit Assignments (Sheet 6 of 10)

Addr.	Register Name	Bit 7	6	5	4	3	2	1	Bit 0	
\$0056	Memory Mapping Size Register (MMSIZ) ⁽³⁾	Read:	MXGS2	MXGS1	W2SZ1	W2SZ0	0	0	W1SZ1	W1SZ0
		Write:								
		Reset:	0	0	0	0	0	0	0	0
\$0057	Memory Mapping Window Base Register (MMWBR) ⁽³⁾	Read:	W2A15	W2A14	W2A13	0	W1A15	W1A14	W1A13	0
		Write:								
		Reset:	0	0	0	0	0	0	0	0
\$0058	Memory Mapping Window 1 Control Register (MM1CR) ⁽³⁾	Read:	0	X1A18	X1A17	X1A16	X1A15	X1A14	X1A13	0
		Write:								
		Reset:	0	0	0	0	0	0	0	0
\$0059	Memory Mapping Window 2 Control Register (MM2CR) ⁽³⁾	Read:	0	X2A18	X2A17	X2A16	X2A15	X2A14	X2A13	0
		Write:								
		Reset:	0	0	0	0	0	0	0	0
\$005A	Chip Select Clock Stretch Register (CSCSTR) ⁽³⁾	Read:	IOSA	IOSB	GP1SA	GP1SB	GP2SA	GP2SB	PCSA	PCSB
		Write:								
		Reset:	0	0	0	0	0	0	0	0
\$005B	Chip Select Control Register (CSCTL) ⁽³⁾	Read:	IOEN	IOPL	IOCSA	IOSZ	GCSPR	PCSEN	PCSZA	PCSZB
		Write:								
		Reset:	0	0	0	0	0	1	0	0
\$005C	General-Purpose Chip Select 1 Address Register (GPCS1A) ⁽³⁾	Read:	G1A18	G1A17	G1A16	G1A15	G1A14	G1A13	G1A12	G1A11
		Write:								
		Reset:	0	0	0	0	0	0	0	0
\$005D	General-Purpose Chip Select 1 Control Register (GPCS1C) ⁽³⁾	Read:	G1DG2	G1DPC	G1POL	G1AV	G1SZA	G1SZB	G1SCC	G1SZD
		Write:								
		Reset:	0	0	0	0	0	0	0	0
\$005E	General-Purpose Chip Select 2 Address Register (GPCS2A) ⁽³⁾	Read:	G2A18	G2A17	G2A16	G2A15	G2A14	G2A13	G2A12	G2A11
		Write:								
		Reset:	0	0	0	0	0	0	0	0
\$005F	General-Purpose Chip Select 2 Control Register (GPCS2C) ⁽³⁾	Read:	0	G2DPC	G2POL	G2AV	G2SZA	G2SZB	G2SZC	G2SZD
		Write:								
		Reset:	0	0	0	0	0	0	0	0
\$0060	Pulse Width Modulation Timer Clock Select Register (PWCLK)	Read:	CON34	CON12	PCKA2	PCKA1	0	PCKB3	PCKB2	PCKB1
		Write:								
		Reset:	0	0	0	0	0	0	0	0

= Unimplemented
 R = Reserved
 U = Undefined

1. Present only in EPROM (711) devices
2. Not available on TK68HC11K4 devices
3. Not available on TK68HC11KS devices

Figure 4-1. Register and Control Bit Assignments (Sheet 7 of 10)

System Configuration


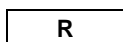
Addr.	Register Name	Bit 7	6	5	4	3	2	1	Bit 0	
\$0061	Pulse Width Modulation Timer Polarity Register (PWPOL)	Read:	PCLK4	PCLK3	PCLK2	PCLK1	PPOL4	PPOL3	PPOL2	PPOL1
		Write:								
		Reset:	0	0	0	0	0	0	0	0
\$0062	Pulse Width Modulation Timer Prescaler Register (PWSCAL)	Read:	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
		Write:								
		Reset:	0	0	0	0	0	0	0	0
\$0063	Pulse Width Modulation Timer Enable Register (PWEN)	Read:	TPWSL	DISCP	0	0	PWEN4	PWEN3	PWEN2	PWEN1
		Write:								
		Reset:	0	0	0	0	0	0	0	0
\$0064	Pulse Width Modulation Timer Counter 1 Register (PWCNT1)	Read:	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
		Write:								
		Reset:	0	0	0	0	0	0	0	0
\$0065	Pulse Width Modulation Timer Counter 2 Register (PWCNT2)	Read:	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
		Write:								
		Reset:	0	0	0	0	0	0	0	0
\$0066	Pulse Width Modulation Timer Counter 3 Register (PWCNT3)	Read:	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
		Write:								
		Reset:	0	0	0	0	0	0	0	0
\$0067	Pulse Width Modulation Timer Counter 4 Register (PWCNT4)	Read:	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
		Write:								
		Reset:	0	0	0	0	0	0	0	0
\$0068	Pulse Width Modulation Timer Period 1 Register (PWPER1)	Read:	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
		Write:								
		Reset:	0	0	0	0	0	0	0	0
\$0069	Pulse Width Modulation Timer Period 2 Register (PWPER2)	Read:	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
		Write:								
		Reset:	0	0	0	0	0	0	0	0
\$006A	Pulse Width Modulation Timer Period 3 Register (PWPER3)	Read:	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
		Write:								
		Reset:	0	0	0	0	0	0	0	0
\$006B	Pulse Width Modulation Timer Period 4 Register (PWPER4)	Read:	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
		Write:								
		Reset:	0	0	0	0	0	0	0	0
\$006C	Pulse Width Modulation Timer Duty Cycle 1 Register (PWDTY1)	Read:	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
		Write:								
		Reset:	0	0	0	0	0	0	0	0

 = Unimplemented
 R = Reserved
 U = Undefined

1. Present only in EPROM (711) devices
2. Not available on TK68HC11K4 devices
3. Not available on TK68HC11KS devices

Figure 4-1. Register and Control Bit Assignments (Sheet 8 of 10)

Addr.	Register Name	Bit 7	6	5	4	3	2	1	Bit 0	
\$006D	Pulse Width Modulation Timer Duty Cycle 2 Register (PWDTY2)	Read:	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
		Write:								
		Reset:	0	0	0	0	0	0	0	0
\$006E	Pulse Width Modulation Timer Duty Cycle 3 Register (PWDTY3)	Read:	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
		Write:								
		Reset:	0	0	0	0	0	0	0	0
\$006F	Pulse Width Modulation Timer Duty Cycle 4 Register (PWDTY4)	Read:	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
		Write:								
		Reset:	0	0	0	0	0	0	0	0
\$0070	SCI Baud Rate Control Register High (SCBDH)	Read:	BTST	BSPL	0	SBR12	SBR11	SBR10	SBR9	SBR8
		Write:								
		Reset:	0	0	0	0	0	U	U	U
\$0071	SCI Baud Rate Control Register Low (SCBDL)	Read:	SBR7	SBR6	SBR5	SBR4	SBR3	SBR2	SBR1	SBR0
		Write:								
		Reset:	0	0	0	0	0	U	U	U
\$0072	SCI Control Register 1 (SCCR1)	Read:	LOOPS	WOMS	0	M	WAKE	ILT	PE	PT
		Write:								
		Reset:	U	U	0	0	0	0	0	0
\$0073	SCI Control Register 2 (SCCR2)	Read:	TIE	TCIE	RIE	ILIE	TE	RE	RWU	SBK
		Write:								
		Reset:	0	0	0	0	0	0	0	0
\$0074	SCI Status Register 1 (SCSR1)	Read:	TDRE	TC	RDRF	IDLE	OR	NF	FE	PF
		Write:								
		Reset:	1	1	0	0	0	0	0	0
\$0075	SCI Status Register 2 (SCSR2)	Read:	0	0	0	0	0	0	0	RAF
		Write:								
		Reset:	1	1	0	0	0	0	0	0
\$0076	SCI Data Register (SCDR)	Read:	R8	T8	0	0	0	0	0	0
		Write:								
		Reset:	Undefined after reset							
\$0077	SCI Data Register (SCDR)	Read:	R7/T7	R6/T6	R5/T5	R4/T4	R3/T3	R2/T2	R1/T1	R0/T0
		Write:								
		Reset:	Undefined after reset							
\$0078	Reserved	Read:	R	R	R	R	R	R	R	R
		Write:								
		Reset:								

 = Unimplemented  = Reserved U = Undefined

1. Present only in EPROM (711) devices
2. Not available on TK68HC11K4 devices
3. Not available on TK68HC11KS devices

Figure 4-1. Register and Control Bit Assignments (Sheet 9 of 10)

System Configuration

Addr.	Register Name	Bit 7	6	5	4	3	2	1	Bit 0	
\$007B	Reserved	R	R	R	R	R	R	R	R	
\$007C	Port H Data Register (PORTH)	Read:	PH7 ⁽³⁾	PH6 ⁽³⁾	PH5 ⁽³⁾	PH4 ⁽³⁾	PH3	PH2	PH1	PH0
		Write:								
		Reset:	0	0	0	0	0	0	0	0
\$007D	Port H Data Direction Register (DDRH)	Read:	DDH7 ⁽³⁾	DDH6 ⁽³⁾	DDH5 ⁽³⁾	DDH4 ⁽³⁾	DDH3	DDH2	DDH1	DDH0
		Write:								
		Reset:	0	0	0	0	0	0	0	0
\$007E	Port G Data Register (PORTG)	Read:	PG7	PG6 ⁽³⁾	PG5 ⁽³⁾	PG4 ⁽³⁾	PG3 ⁽³⁾	PG2 ⁽³⁾	PG1 ⁽³⁾	PG0 ⁽³⁾
		Write:								
		Reset:	0	0	0	0	0	0	0	0
\$007F	Port G Data Direction Register (DDRG)	Read:	DDG7	DDG6 ⁽³⁾	DDG5 ⁽³⁾	DDG4 ⁽³⁾	DDG3 ⁽³⁾	DDG2 ⁽³⁾	DDG1 ⁽³⁾	DDG0 ⁽³⁾
		Write:								
		Reset:	0	0	0	0	0	0	0	0

= Unimplemented
 R = Reserved
 U = Undefined

1. Present only in EPROM (711) devices
2. Not available on TK68HC11K4 devices
3. Not available on TK68HC11KS devices

Figure 4-1. Register and Control Bit Assignments (Sheet 10 of 10)

4.4 Operating Modes

The two normal modes of operation in the TK68HC11K Family are:

- Single-chip mode — All port pins available for input/output (I/O); only on-board memory accessible
- Expanded mode — Access to external memory; 25 I/O pins used for interface

The two special modes of operation are:

- Bootstrap mode — A variation of single-chip mode; executes a bootloader program in an internal bootstrap read-only memory (ROM)
- Test mode — A variation of the expanded mode used in production testing; allows privileged access to internal resources

The logic levels applied at reset to input pins MODA and MODB determine the operating mode. See **4.4.5 Mode Selection**.

4.4.1 Single-Chip Mode

In single-chip mode, the MCU functions as a self-contained microcontroller. In this mode, all address and data activity occurs within the MCU. Ports B, C, F, G, and H are available for general-purpose I/O because the external address and data buses are not required.

4.4.2 Expanded Mode

In expanded mode, the MCU uses ports B, C, F, and G to access a 64-Kbyte address space. This includes:

- The same on-chip memory addresses used in single-chip mode
- External memory
- Peripheral devices

Port B provides the high-order address byte (Addr[15:8]), port F the low-order address byte (Addr[7:0]), port C the data bus (Data[7:0]), and port G pin 7 the read/write line ($\overline{R/W}$) which controls direction of data flow.

Expanded mode also enables these two special features available only on the K Family devices:

1. Memory expansion uses port G[5:0] to increase the available external

address space to 1 Mbyte.

2. Four chip-select lines on port H[7:4] simplify selection of external memory devices.

Both of these features are described in **Section 11. Memory Expansion and Chip Selects**.

4.4.3 Bootstrap Mode

Resetting the MCU in special bootstrap mode selects a reset vector to a special ROM bootloader program at addresses \$BE00–\$BFFF. With this program personal programs can be downloaded into on-chip random-access memory (RAM) through the SCI. To do this:

1. Send \$FF to the serial communications interface (SCI) receiver at a baud rate of either $E \text{ clock} \div 16$ or $E \text{ clock} \div 104$, which is 1200 baud with a 2-MHz E clock.
2. Download up to 768 bytes of program data, which the CPU places into RAM starting at \$0080 and also echoes back on the TxD signal. The bootloader program ends the download after 768 bytes or when the received data line is idle for at least four character times.

When loading is complete, the MCU jumps to location \$0080 and begins executing the code. Interrupt vectors are directed to RAM, which allows the use of interrupts through a jump table. The SCI transmitter requires an external pullup resistor since it is part of port D, which the bootloader configures for wired-OR operation.

For a detailed description of bootstrap mode, refer to the Tekmos application note entitled *TK68HC11 Bootstrap Mode*, document order number AN1060/D.

4.4.4 Special Test Mode

Special test mode, a variation of the expanded mode, is used primarily during Tekmos's internal production testing. However, for those devices containing EPROM, it can be used to program the EPROM for program calibration data in EEPROM and support emulation and debugging during development.

For more detailed information, refer to **4.6.1 Programming the EPROM with Downloaded Data**.

4.4.5 Mode Selection

The operating mode is selected by applying the appropriate logic states to the MODA and MODB pins during reset. MODA selects single-chip mode (0) or expanded mode (1). A logic high on MODB selects normal modes, and reset vectors are fetched from memory area \$FFFE–\$FFFF. A logic low on MODB selects special modes, and reset vectors are fetched from memory area \$BFFE–\$BFFF. Values reflecting the selected mode are latched into the RBOOT, SMOD, and MDA bits of the highest priority I-bit interrupt and miscellaneous register (HPRIO) on the rising edge of $\overline{\text{RESET}}$.

Table 4-2 summarizes the inputs, modes selected, and register bits latched. The HPRIO register is illustrated in **Figure 4-2**.

Table 4-2. Hardware Mode Select Summary

Inputs		Mode	Control Bits in HPRIO Latched at Reset		
MODB	MODA		RBOOT	SMOD	MDA
1	0	Single-chip	0	0	0
1	1	Expanded	0	0	1
0	0	Special bootstrap	1	1	0
0	1	Special test	0	1	1

Address: \$003C

	Bit 7	6	5	4	3	2	1	Bit 0
Read:	RBOOT ⁽¹⁾	SMOD ⁽¹⁾	MDA ⁽¹⁾	PSEL4	PSEL3	PSEL2	PSEL1	PSEL0
Write:								
Reset:	—	—	—	0	0	1	1	0

1. The values of the RBOOT, SMOD, and MDA bits at reset depend on the mode during initialization.

Figure 4-2. Highest Priority I-Bit Interrupt and Miscellaneous Register (HPRIO)

NOTE: To avoid race conditions, HPRIO is designed so that it can be written only while I-bit interrupts are inhibited (for example, the I bit is set).

RBOOT — Read Bootstrap ROM Bit

In special modes, this bit enables the bootloader ROM

0 = Bootloader ROM disabled and not in map

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

In normal modes this bit is clear and cannot be written.

SMOD — Special Mode Select Bit

This bit reflects the inverse of the MODB input pin at the rising edge of $\overline{\text{RESET}}$. If MODB is low during reset, SMOD is set; if MODB is high during reset, SMOD is cleared. Software can clear the SMOD bit, but cannot set it. Thus, it is possible for software to change the operating mode from special to normal, but not vice versa. To switch from a special mode to a normal mode, write to the access-limited registers (see **Table 4-1**) before clearing SMOD.

0 = Normal mode operation in effect

1 = Special mode operation in effect

MDA — Mode Select A Bit

The mode select A bit reflects the status of the MODA input pin at the rising edge of $\overline{\text{RESET}}$. Software can change the MDA bit only while the SMOD bit is set, effectively switching the operating mode between special bootstrap and special test modes. Once the SMOD bit is clear, the MODA bit is read-only and the operating mode cannot be changed without going through a reset sequence.

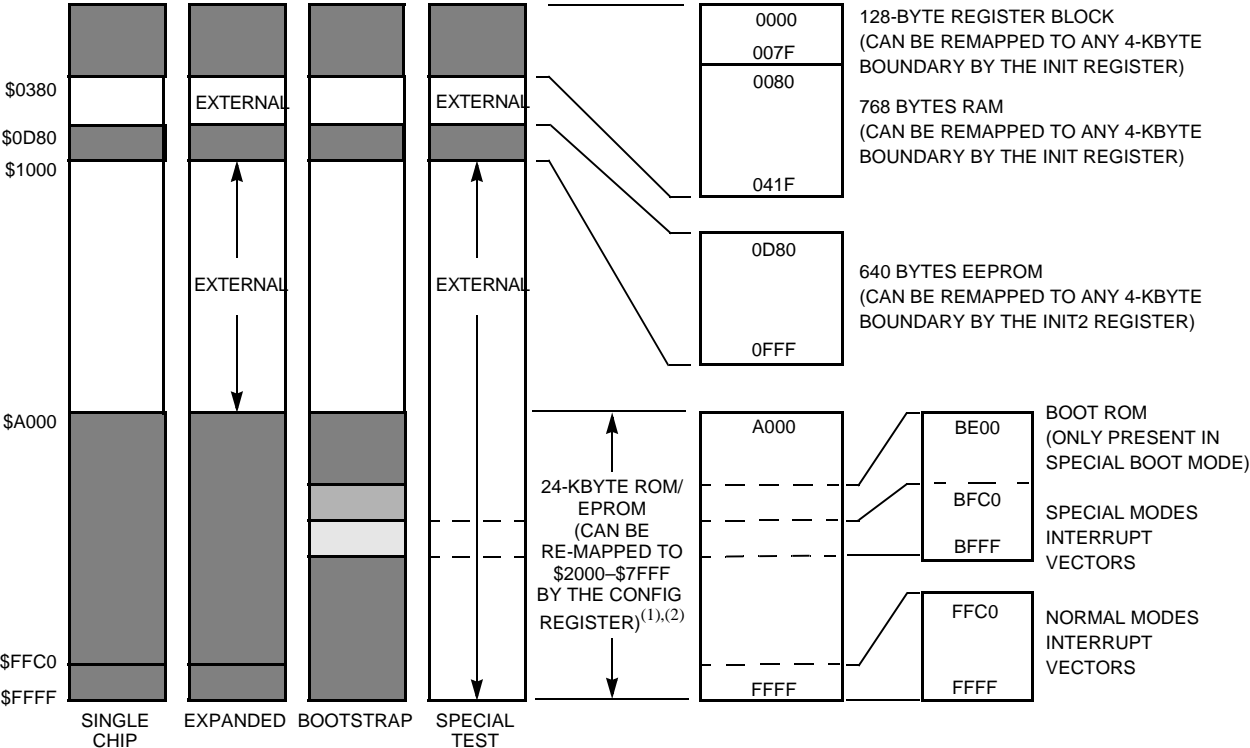
0 = Normal single-chip or special bootstrap mode in effect

1 = Normal expanded or special test mode in effect

After $\overline{\text{RESET}}$ is released, the mode select pins revert to their alternate functions, described in **2.9 DCSAMode Selection, Instruction Cycle Reference, and Standby Power (MODA/LIR and MODB/VSTBY)**, and no longer influence the MCU operating mode.

4.5 Memory Map

The operating mode determines memory mapping and whether memory is addressed on-chip or off-chip. **Figure 4-3** and **Figure 4-4** illustrate the TK68HC11K4 Family and TK68HC11KS Family memory maps for each of the four modes of operation. Memory locations for on-chip resources are the same for both expanded and single-chip modes.



- Notes:
1. EPROM can be enabled in special test mode by setting the ROMON bit in the CONFIG register after reset.
 2. The configuration shown in this diagram is the TK68HC11K4.

Figure 4-3. TK68HC11K4 Family Memory Map

System Configuration

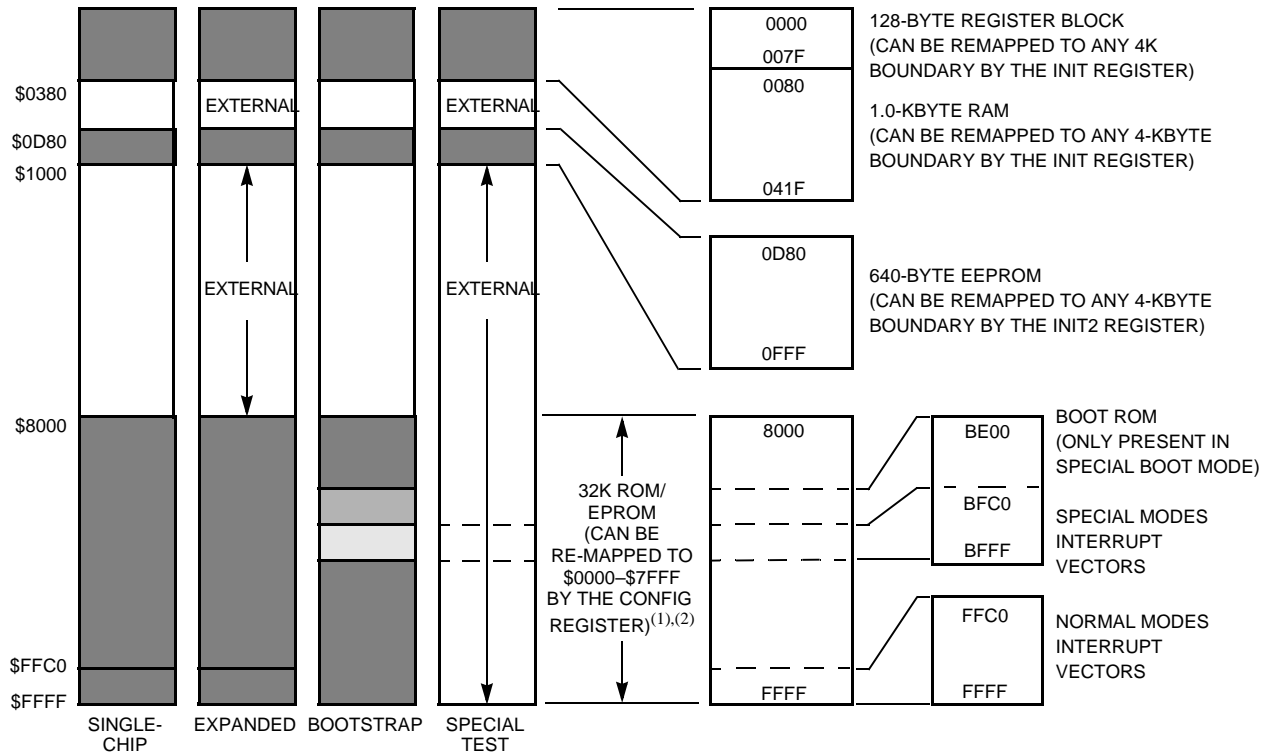


Figure 4-4. TK68HC11KS Family Memory Map

Table 4-3 shows the memory map addresses for the TK68HC11K Family devices.

Table 4-3. TK68HC11K Family Memory Map Addresses

Device	ROM/EPROM	RAM	EEPROM
68HC11K4	\$A000-\$FFFF	\$0080-\$037F	\$0D80-\$0FFF
68HC711K4	\$A000-\$FFFF	\$0080-\$037F	\$0D80-\$0FFF
68HC11KS2	\$8000-\$FFFF	\$0080-\$047F	\$0D80-\$0FFF
68HC711KS2	\$8000-\$FFFF	\$0080-\$047F	\$0D80-\$0FFF
68HC11KS8	\$4000-\$FFFF	\$0080-\$067F	\$0D80-\$0FFF
68HC711KS8	\$4000-\$FFFF	\$0080-\$067F	\$0D80-\$0FFF

4.5.1 Control Registers and RAM

Out of reset, the 128-byte register block is mapped to \$0000 and the 768-byte RAM (1 Kbyte and 1.5 Kbyte on the [7]11KS2 and [7]11KS8, respectively) is mapped to \$0080. Both the register block and the RAM can be placed at any other 4-Kbyte boundary (\$x000 and \$x080, respectively) by writing the appropriate value to the INIT register.

NOTE: RAM and I/O mapping register (INIT) is an 8-bit special-purpose register which can be written to only once within the first 64 E-clock cycles after a reset, and then it becomes a read-only register.

Address: \$003D

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

Figure 4-5. RAM and I/O Mapping Register (INIT)

RAM[3:0] — RAM Map Position Bits

These four bits determine the position of RAM in the memory map by specifying the upper hexadecimal digit of the RAM address. Refer to **Table 4-4**.

REG[3:0] — Register Block Position Bits

These four bits determine the position of the register block in memory by specifying the upper hexadecimal digit of the block address. Refer to **Table 4-5**.

Table 4-4. RAM Mapping

RAM[3:0]	Address ⁽¹⁾
0000	\$0000–\$02FF
0001	\$1000–\$12FF
0010	\$2000–\$22FF
0011	\$3000–\$32FF
0100	\$4000–\$42FF
0101	\$5000–\$52FF
0110	\$6000–\$62FF
0111	\$7000–\$72FF
1000	\$8000–\$82FF
1001	\$9000–\$92FF
1010	\$A000–\$A2FF
1011	\$B000–\$B2FF
1100	\$C000–\$C2FF
1101	\$D000–\$D2FF
1110	\$E000–\$E2FF
1111	\$F000–\$F2FF

Table 4-5. Register Mapping

REG[3:0]	Address
0000	\$0000–\$007F
0001	\$1000–\$107F
0010	\$2000–\$207F
0011	\$3000–\$307F
0100	\$4000–\$407F
0101	\$5000–\$507F
0110	\$6000–\$607F
0111	\$7000–\$707F
1000	\$8000–\$807F
1001	\$9000–\$907F
1010	\$A000–\$A07F
1011	\$B000–\$B07F
1100	\$C000–\$C07F
1101	\$D000–\$D07F
1110	\$E000–\$E07F
1111	\$F000–\$F07F

1. On the [7]11KS2, RAM is 1 K, so the address range is \$x000–\$x3FF.
 On the [7]11KS8, RAM is 1.5 K, so the address range is \$x000–\$x5FF.

Since the direct addressing mode accesses RAM more quickly and efficiently than other addressing modes, many applications will find the default locations of registers and on-board RAM at the bottom of memory to be the most advantageous.

If RAM and the registers are both mapped to the same 4-K boundary, the registers are mapped at \$x000–\$x07F, and RAM is moved up to \$x080–\$x37F (\$x080–x47F for the [7]11KS2 and \$X080–\$x67F for the [7]11KS8).

4.5.2 ROM or EPROM

The presence and location of the 24-Kbyte (EP)ROM on the [7]11K4 is determined by two bits in the system configuration register (CONFIG). The CONFIG register is a special EEPROM register (see **Figure 4-6**).

(EP)ROM is present in the memory map when the ROMON bit is set and removed from the memory map when the bit is cleared. The default location of

this memory is \$A000–\$FFFF, but it can be moved to \$2000–\$7FFF in expanded mode by clearing the ROMAD bit. Both bits are set out of reset in single-chip mode.

- On the [7]11KS2, (EP)ROM is 32 K, mapped to \$8000–\$FFFF by default, and moved to \$0000–\$7FFF by clearing the ROMAD bit.
- On the [7]11KS8, (EP)ROM is 48 K mapped to \$4000–\$FFFF by default, and moved to \$0000–\$BFFF by clearing the ROMAD bit.

In special test mode, the ROMON bit is forced to 0, removing (EP)ROM from the memory map.

4.5.3 EEPROM

The TK68HC11K Family devices contain 640 bytes of EEPROM. It is initially located at \$0D80 after reset if it is enabled by the EEON bit in the CONFIG register (see **Figure 4-6**). It can be relocated to any 4-K boundary (\$xD80) by writing to the EEPROM mapping register (INIT2) (see **Figure 4-7**).

NOTE: *On the TK68HC11KS devices, the EEPROM can be mapped to where it will contain the vector space.*

Address: \$003F

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

Figure 4-6. System Configuration Register (CONFIG)

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

ROMAD — ROM Address Mapping Control Bit

Set out of reset in single-chip mode

- 0 = (EP)ROM set at \$2000–\$7FFF;
\$0000–\$7FFF in [7]11KS2;
\$0000–\$BFFF in [7]11KS8
(expanded mode only)
- 1 = (EP)ROM set at \$A000–\$FFFF;
\$8000–\$FFFF in [7]11KS2;
\$4000–\$FFFF in [7]11KS8

ROMON — ROM/PROM Enable Bit

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

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

EEON — EEPROM Enable Bit

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

Address: \$0037

	Bit 7	6	5	4	3	2	1	Bit 0
Read:	EE3	EE2	EE1	EE0	0	0	0	0
Write:					0	0	0	0
Reset:	0	0	0	0	0	0	0	0

Figure 4-7. EEPROM Mapping Register (INIT2)

NOTE: *INIT2 is writable once in normal modes and writable at any time in special modes.*

EE[3:0] — EEPROM Map Position Bits

These four bits determine the most-significant hexadecimal digit in the address range of the EEPROM, as shown in **Table 4-6**.

Table 4-6. EEPROM Map

EE[3:0]	Location
0000	\$0D80–\$0FFF
0001	\$1D80–\$1FFF
0010	\$2D80–\$2FFF
0011	\$3D80–\$3FFF
0100	\$4D80–\$4FFF
0101	\$5D80–\$5FFF
0110	\$6D80–\$6FFF
0111	\$7D80–\$7FFF
1000	\$8D80–\$8FFF
1001	\$9D80–\$9FFF
1010	\$AD80–\$AFFF
1011	\$BD80–\$BFFF
1100	\$CD80–\$CFFF
1101	\$DD80–\$DFFF
1110	\$ED80–\$EFFF
1111	\$FD80–\$FFFF

4.5.4 Bootloader ROM

The bootloader program occupies 512 bytes of bootstrap ROM at addresses \$BE00–\$BFFF. It is active only in special modes when the RBOOT bit is set.

4.6 EPROM/OTPROM (TK68HC711K4,TK68HC711KS8, and TK68HC711KS2)

The TK68HC711K4 devices include 24 Kbytes of on-chip EPROM (OTPROM in non-windowed packages). The TK68HC711KS8 has 48 Kbytes of EPROM and the TK68HC711KS2 has 32 Kbytes of EPROM.

The two methods available to program the EPROM are:

- Downloading data through the serial communication interface (SCI) in bootstrap or special test mode
- Programming individual bytes from memory

Before proceeding with programming:

- Ensure that the CONFIG register ROMON bit is set.
- Ensure that the $\overline{\text{IRQ}}$ pin is pulled to a high level.
- Apply 12 volts to the $\overline{\text{XIRQ}}/V_{\text{PP}}$ pin.

Program the EPROM only at room temperature. Place an opaque label over the quartz window on windowed parts after programming.

4.6.1 Programming the EPROM with Downloaded Data

The MCU can download EPROM data through the SCI while in the special test or bootstrap modes. This can be done either with custom software, also downloaded through the SCI, or with a built-in utility program in bootstrap ROM. In either case, the 12-volt nominal programming voltage must be present on the $\overline{\text{XIRQ}}/V_{\text{PP}}$ pin.

To use the bootstrap ROM utility, download a 3-byte program consisting of a single jump instruction to \$BF00, the starting address of the resident EPROM programming utility. The utility program sets the X and Y index registers to default values, then receives data from an external host and programs it into the EPROM. The value in the X index register determines programming delay time. The value in the Y index register is a pointer to the first address in EPROM to be programmed. The default starting address is \$8000 for the TK68HC11KS2 and \$4000 for the TK68HC11KS8.

When the utility program is ready to receive programming data, it sends the host a \$FF character and waits for a reply. When the host sees the \$FF character, it sends the EPROM programming data, starting with the first location in the EPROM array. After the MCU receives the last byte to be programmed and returns the corresponding verification data, it terminates the programming operation by initiating a reset. Refer to the TekmosTekmos application note entitled *TK68HC11 Bootstrap Mode*, document order number AN1060/D.

4.6.2 Programming the EPROM from Memory

In this method, software programs the EPROM one byte at a time. Each byte is read from memory, then latched and programmed into the EPROM using the EPROM programming control register (EPROG). This procedure can be done in any operating mode.

Address: \$002B

	Bit 7	6	5	4	3	2	1	Bit 0
Read:	MBE	0	ELAT	EXCOL	EXROW	0	0	EPGM
Write:								
Reset:	0	0	0	0	0	0	0	0

Figure 4-8. EPROM Programming Control Register (EPROG)

MBE — Multiple-Byte Program Enable Bit

MBE is for factory use only and is accessible only in special test mode. When MBE is set, the MCU ignores address bit 5, so that bytes with ADDR5 = 0 and ADDR5 = 1 both get programmed with the same data.

- 0 = Normal programming
- 1 = Multiple-byte programming enabled

ELAT — EPROM Latch Control Bit

Setting ELAT latches the address and data of writes to the EPROM. The EPROM cannot be read. ELAT can be read at any time. ELAT can be written any time except when PGM = 1, which disables writes to ELAT.

- 0 = EPROM address and data bus configured for normal reads. EPROM cannot be programmed.
- 1 = EPROM address and data bus are configured for programming. Address and data of writes to EPROM are latched. EPROM cannot be read.

EXCOL — Select Extra Columns Bit

EXCOL is for factory use only and is accessible only in special test mode. When EXCOL equals 1, extra columns can be accessed at bit 7 and bit 0. Addresses use bits [11:5]. Bits [4:1] are ignored.

0 = User array selected

1 = Extra columns selected and user array disabled

EXROW — Select Extra Rows Bit

EXROW is for factory use only and is only accessible in special test mode. When EXROW equals 1, two extra rows are available. Addresses use bits [5:0]. Bits [11:6] are ignored.

0 = User array selected

1 = Extra rows selected and user array is disabled

EPGM — EPROM Programming Enable Bit

EPGM applies programming voltage (V_{PP}) to the EPROM. EPGM can be read at any time. EPGM can be written only when ELAT = 1.

0 = Programming voltage to EPROM array is disconnected

1 = Programming voltage to EPROM array is connected; ELAT cannot be changed.

This procedure programs one byte into EPROM. On entry, accumulator A contains the byte of data to be programmed and X contains the target EPROM address.

```

EPROG LDAB  #$20
      STAB  $002B  Set ELAT bit to enable EPROM latches.
                        (EPGM must be 0.)
      STAA  $0,X   Store data to EPROM address
      LDAB  #$21
      STAB  $002B  Set EPGM bit with ELAT=1
                        to enable EPROM programming voltage
      JSR   DLYEP  Delay 1-2 ms
      CLR  $002B  Turn off programming voltage and set to
                        READ mode

```

4.7 EEPROM and the CONFIG Register

The 640-byte on-board EEPROM is enabled by the EEON bit in the CONFIG register and located on a 4-K boundary determined by the INIT2 register (**4.5.3 EEPROM**). An internal charge pump supplies the programming voltage for the EEPROM, eliminating the need for an external high-voltage supply.

The two methods available to program the EEPROM are

- Downloading data through the serial communication interface (SCI) in bootstrap or special test mode
- programming individual bytes from memory

An extra row of 16 bytes of EEPROM located at \$0D60 is available for factory testing in special modes. Endurance and data retention specifications do not apply to this row.

4.7.1 EEPROM Registers

This section describes the EEPROM registers:

- EEPROM programming control register (PPROG)
- Block protect register (BPROT)
- System configuration options register (OPTION)

The EEPROM programming control register (PPROG) controls programming and erasing. The block protect register (BPROT) can prevent inadvertent writes to (or erases of) blocks of EEPROM and enables or disables a low-voltage inhibit circuit for further protection. The CSEL bit in the system configuration

options register (OPTION) selects an on-chip oscillator clock for programming and erasing when operating at frequencies below 1 MHz.

4.7.1.1 EEPROM Programming Control Register

Address: \$003B

	Bit 7	6	5	4	3	2	1	Bit 0
Read:			LVPI	BYTE	ROW	ERASE	EELAT	EEPGM
Write:	ODD	EVEN						
Reset:	0	0	0	0	0	0	0	0

= Unimplemented

Figure 4-9. EEPROM Programming Control Register (PPROG)

ODD — Program Odd Rows in Half of EEPROM Bit

This bit is accessible only in test mode.

EVEN — Program Even Rows in Half of EEPROM Bit

This bit is accessible only in test mode.

LVPI — Low-Voltage Programming Inhibit Bit

LVPI is a read-only bit which reflects the condition of the voltage detector enabled by the VPEN bit in the BPROT register.

0 = EEPROM programming allowed

1 = EEPROM programming inhibited due to detected low voltage

BYTE — Byte/Other EEPROM Erase Mode Bit

0 = Row or bulk erase mode used

1 = Erase only one byte of EEPROM

ROW — Row/All EEPROM Erase Mode Bit
 0 = All 640 bytes of EEPROM erased
 1 = Erase only one 16-byte row of EEPROM

NOTE: *ROW is valid only when BYTE = 0.*

The BYTE and ROW bits work together to determine the scope of erasing, as shown in **Table 4-7**.

Table 4-7. Scope of EEPROM Erase

BYTE	ROW	Action
0	0	Bulk erase; all 640 bytes
0	1	Row erase; 16 bytes
1	0	Byte erase
1	1	Byte erase

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

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

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

4.7.1.2 Block Protect Register

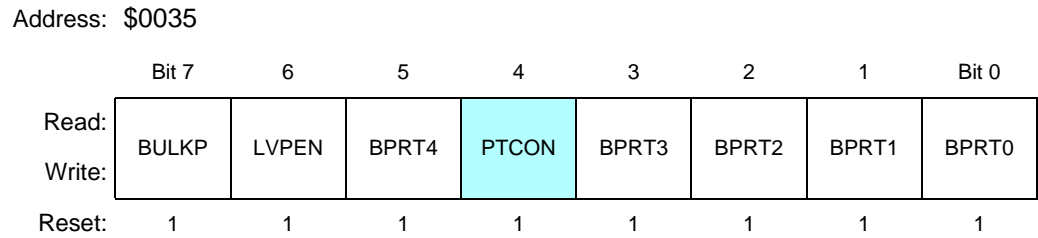


Figure 4-10. Block Protect Register (BPROT)

BULKP — Bulk Erase of EEPROM Protect Bit

0 = EEPROM can be bulk erased normally.

1 = EEPROM cannot be bulk or row erased.

LVPEN — Low-Voltage Programming Protect Enable Bit

Setting this bit enables a special-purpose on-board voltage detection circuit. If this circuit detects that V_{DD} has fallen below a safe operating voltage it sets the low-voltage programming inhibit bit (LVPI) in the PPROG register.

0 = Low-voltage programming protect for EEPROM disabled

1 = Low-voltage programming protect for EEPROM enabled

BPRT[4:0] — Block Protect Bits for EEPROM Bits, see **Table 4-8**

0 = Protection disabled for associated block

1 = Protection enabled for associated block

Table 4-8. EEPROM Block Protect

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

4.7.1.3 System Configuration Options Register

Address: \$0039

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

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

Figure 4-11. System Configuration Options Register (OPTION)

CSEL — Clock Select Bit

Selects the built-in RC clock source for on-chip EEPROM and A/D charge pumps. This clock should be used when the E clock falls below 1 MHz.

0 = A/D and EEPROM use system E clock.

1 = A/D and EEPROM use internal RC clock.

4.7.2 EEPROM Programming

To write to any EEPROM byte, it must first be erased, for instance, all of its bits must be set. A single byte, a row, or the entire EEPROM in a single procedure can be erased by adjusting the BYTE and ROW bits in PPROG. Once the targeted area has been erased, each byte can be individually written.

The procedures for both writing and erasing involve these five steps:

1. **Set the EELAT bit in PPROG.** If erasing, also set the ERASE bit and the appropriate BYTE and ROW bits.
2. **Write data to the appropriate EEPROM address.** If erasing, any data will work. To erase a row, write to any location in the row. To erase the entire EEPROM, write to any location in the array. This step is done before applying the programming voltage because setting the EEPGM bit inhibits writes to EEPROM addresses.
3. **Set the EEPGM bit in PPROG,** keeping EELAT set. If erasing, also set the ERASE bit and the appropriate BYTE and ROW bits.
4. **Delay for 10 ms.**
5. **Clear the PPROG register** to turn off the high voltage and reconfigure the EEPROM address and data buses for normal operation.

The following examples demonstrate programming a single EEPROM byte, erasing the entire EEPROM, erasing a row (16 bytes), and erasing a single byte.

4.7.2.1 EEPROM Programming

On entry, accumulator A contains the data to be written and X points to the address to be programmed.

```
EEPROG  LDAB  #$02
        STAB  $003B  Set EELAT bit to enable EEPROM
                    latches.
        STAA  $0,X  Store data to EPROM address
        LDAB  #$03
        STAB  $002B  Set EPGM bit with ELAT=1
                    to enable EEPROM programming voltage
        JSR   DLY10  Delay 10 ms
        CLR   $003B  Turn off programming voltage and set
                    to READ mode
```

4.7.2.2 EEPROM Bulk Erase

```
BULKE   LDAB  #$06
        STAB  $003B  Set EELAT and ERASE.
        STAA  $0,X  Store any data to any EEPROM address
        LDAB  #$07
        STAB  $002B  Set EEPGM bit as well
                    to enable EEPROM programming voltage
        JSR   DLY10  Delay 10 ms
        CLR   $003B  Turn off programming voltage and set
                    to READ mode
```

4.7.2.3 EEPROM Row Erase

ROWE	LDAB	#\$07	
	STAB	\$003B	Set EELAT, ERASE and ROW.
	STAA	\$0,X	Store any data to any EEPROM address in row
	LDAB	#\$07	
	STAB	\$002B	Set EEPGM bit as well to enable EEPROM programming voltage
	JSR	DLY10	Delay 10 ms
	CLR	\$003B	Turn off programming voltage and set to READ mode

4.7.2.4 EEPROM Byte Erase

BYTEE	LDAB	#\$16	
	STAB	\$003B	Set EELAT, ERASE and BYTE.
	STAA	\$0,X	Store any data to targeted EEPROM address
	LDAB	#\$17	
	STAB	\$002B	Set EEPGM bit as well to enable EEPROM programming voltage
	JSR	DLY10	Delay 10 ms
	CLR	\$003B	Turn off programming voltage and set to READ mode

4.7.3 CONFIG Register Programming

The CONFIG register is implemented with EEPROM cells, so EEPROM procedures are required to change it. CONFIG can be programmed or erased (including byte erase) while the MCU is operating in any mode, provided that PTCON in BPROT is clear.

Address: \$0035

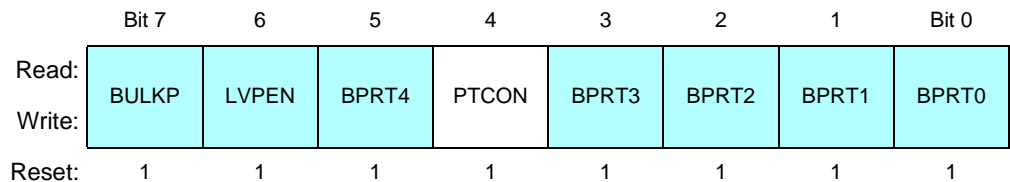


Figure 4-12. Block Protect Register (BPROT)

PTCON — Protect for CONFIG Bit

0 = CONFIG register can be programmed or erased normally.

1 = CONFIG register cannot be programmed or erased.

To change the value in the CONFIG register, complete this procedure. Do not initiate a reset until the procedure is complete.

- Erase the CONFIG register.
- Program the new value to the CONFIG address.
- Initiate reset.

4.7.4 RAM and EEPROM Security

The NOSEC bit in the CONFIG register enables and disables an optional security feature which protects the contents of EEPROM and RAM from unauthorized access. This is done by restricting operation to single-chip modes, preventing the memory locations from being monitored externally. Single-chip modes do not allow visibility of the internal address and data buses. Resident programs, however, have unlimited access to the internal EEPROM and RAM and can read, write, or transfer the contents of these memories.

Address: \$003F

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

Figure 4-13. System Configuration Register (CONFIG)

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

NOSEC — RAM and EPROM Security Disabled Bit

0 = Enable security

1 = Disable security

TK68HC11K Family devices are normally manufactured with NOSEC set and the security option unavailable. However, on special request, a mask option is selected during fabrication that enables the security mode. The secure mode can be invoked on these parts by clearing NOSEC. Contact a Tekmos representative for information on the availability of this feature.

The bootstrap program performs this sequence when the security feature is present, enabled, and bootstrap mode is selected:

1. Output \$FF, all 1s, on the SCI.
2. Clear the BPROT register by turning block protect off.
3. If the EEPROM is enabled, erase the EEPROM.
4. Verify that the EEPROM is erased. If EEPROM is not erased, begin sequence again.
5. Write \$FF, all 1s, to the entire block of RAM.
6. Erase the CONFIG register.

If all of the operations are successful, the bootload process continues as if the device were never secured.

Section 5. Resets and Interrupts

5.1 Contents

5.2	Introduction	102
5.3	Sources of Resets	102
5.3.1	Power-On Reset (POR)	103
5.3.2	External Reset (RESET)	103
5.3.3	Computer Operating Properly (COP) System	103
5.3.3.1	System Configuration Register	104
5.3.3.2	System Configuration Options Register	105
5.3.3.3	Arm/Reset COP Timer Circuitry Register	106
5.3.4	Clock Monitor Reset	106
5.3.4.1	System Configuration Options Register	107
5.3.4.2	System Configuration Options Register 2	108
5.4	Effects of Reset	111
5.5	Interrupts	113
5.5.1	Non-Maskable Interrupts	116
5.5.1.1	Non-Maskable Interrupt Request (XIRQ)	116
5.5.1.2	Illegal Opcode Trap	116
5.5.1.3	Software Interrupt (SWI)	117
5.5.2	Maskable Interrupts	117
5.6	Reset and Interrupt Priority	118
5.7	Reset and Interrupt Processing	119
5.8	Low-Power Operation	125
5.8.1	Wait Mode	126
5.8.2	Stop Mode	126
5.8.3	Slow Mode	128

5.2 Introduction

When a reset or interrupt occurs, the microcontroller (MCU) retrieves the starting address of a program or interrupt routine from a vector table in memory and loads it in the program counter. A reset immediately stops execution of the current instruction and reinitializes the control registers. An interrupt preserves the current program status, performs an interrupt service routine, and resumes operation as if there had been no interruption.

5.3 Sources of Resets

The four sources of reset are:

- Power-on reset (POR)
- External reset ($\overline{\text{RESET}}$)
- Computer operating properly (COP) system
- Clock monitor

NOTE: *Power-on reset and external reset share the same interrupt vectors.*

The CPU fetches a restart vector during the first three clock cycles after reset and begins executing instructions. Vector selection is based on the type of reset and operating mode, as shown in **Table 5-1**.

Table 5-1. Reset Vectors

Operating Mode	POR or $\overline{\text{RESET}}$	Clock Monitor	COP Watchdog
Normal	\$FFFE and \$FFFF	\$FFFC and \$FFFD	\$FFFA and \$FFFB
Test or bootstrap	\$BFFE and \$BFFF	\$BFFC and \$BFFD	\$BFFA and \$BFFB

5.3.1 Power-On Reset (POR)

A positive transition on v_{DD} generates a POR, which is used only for power-up conditions. POR cannot be used to detect drops in power supply voltages. The CPU delays 4064 internal clock cycles after the oscillator becomes active to allow the clock generator to stabilize, then checks the $\overline{\text{RESET}}$ pin. If $\overline{\text{RESET}}$ is at logical 0, the CPU remains in the reset condition until the $\overline{\text{RESET}}$ pin goes to logical 1.

5.3.2 External Reset ($\overline{\text{RESET}}$)

The CPU distinguishes between internal and external reset conditions by sensing whether the reset pin rises to a logic 1 in less than two E-clock cycles after an internal device releases reset. When a reset condition is sensed, the $\overline{\text{RESET}}$ pin is driven low by an internal device for four E-clock cycles, then released. Two E-clock cycles later, it is sampled. If the pin is still held low, the CPU assumes that an external reset has occurred. If the pin is high, it indicates that the reset was initiated internally by either the COP system or the clock monitor.

NOTE: *It is not advisable to connect an external resistor capacitor (RC) power-up delay circuit to the reset pin of TK68HC11 devices because the circuit charge time constant can cause the device to misinterpret the type of reset that occurred.*

5.3.3 Computer Operating Properly (COP) System

The MCU includes a COP system to help protect against software failures. When the COP is enabled, software periodically reinitializes a free-running watchdog timer before it times out and resets the system. Such a system reset indicates that a software error has occurred.

Three registers are involved in COP operation:

- The CONFIG register contains a bit which determines whether the COP system is enabled or disabled.
- The OPTION register contains two bits which determine the COP timeout period.
- The COPRST register must be written by software to reset the watchdog timer.

NOTE: Throughout this manual, the registers are discussed by function. In the event that not all bits in a register are referenced, the bits that are not discussed are shaded.

5.3.3.1 System Configuration Register

In normal modes, COP is enabled out of reset and does not depend on software action. To disable the COP system, set the NOCOP bit in the CONFIG register (see **Figure 5-1**). In special test and bootstrap operating modes, the COP system is initially inhibited by the disable resets (DISR) control bit in the TEST1 register. The DISR bit can subsequently be written to 0 to enable COP resets.

Address: \$003F

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

Figure 5-1. System Configuration Register (CONFIG)

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

NOCOP — COP System Disable Bit
 0 = COP enabled
 1 = COP disabled

5.3.3.2 System Configuration Options Register

Two bits in the OPTION register select one of four values for the COP timer.

Address: \$0039

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

Figure 5-2. System Configuration Options Register (OPTION)

CR[1:0] — COP Timer Rate Select Bits

The MCU derives the counter for the COP timer by dividing the system E clock by 2^{15} and applying a further scaling factor selected by CR[1:0] as shown in **Table 5-2**. After reset, these bits are 0, and that condition selects the fastest timeout period.

NOTE: *In normal operating modes, these bits can be written only once within 64 bus cycles after reset.*

Table 5-2. COP Timeout

EXTAL Frequencies						
EXTAL Freq.	8.0 MHz	12.0 MHz	16.0 MHz	20.0 MHz	24.0 MHz	Other EXTAL
E Clock Freq.	2.0 MHz	3.0 MHz	4.0 MHz	5.0 MHz	6.0 MHz	EXTAL ÷ 4
Control Bits SPR[2:0]	COP Timeout					Timeout
	0/+16.384 ms	0/+10.923 ms	0/+8.192 ms	0/+6.544 ms	0/+5.461 ms	0/+ $2^{15} \div E$
0 0	16.384 ms	10.923 ms	8.192 ms	6.554 ms	5.461 ms	$2^{15} \div E$
0 1	65.536 ms	43.691 ms	32.768 ms	26.214 ms	21.845 ms	$2^{17} \div E$
1 0	262.144 ms	174.763 ms	131.072 ms	104.858 ms	87.381 ms	$2^{19} \div E$
1 1	1.049 sec	699.051 ms	524.288 ms	419.430 ms	349.525 ms	$2^{21} \div E$

5.3.3.3 Arm/Reset COP Timer Circuitry Register

Address: \$003A

	Bit 7	6	5	4	3	2	1	Bit 0
Read:	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Write:	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Reset:	0	0	0	0	0	0	0	0

Figure 5-3. Arm/Reset COP Timer Circuitry Register (COPRST)

To prevent a COP reset, this sequence must be completed:

1. Write \$55 to COPRST to arm the COP timer clearing mechanism.
2. Write \$AA to COPRST to clear the COP timer.

NOTE: *Performing instructions between these two steps is possible as long as both steps are completed in the correct sequence before the timer times out.*

5.3.4 Clock Monitor Reset

The clock monitor can serve as a backup for the COP system. Its circuit is based on an internal RC time delay. If no MCU clock edges are detected within this RC time delay, the clock monitor generates a system reset. Because the COP needs a clock to function, it is disabled when the clocks stop. Thus, the clock monitor system can detect clock failures not detected by the COP system.

5.3.4.1 System Configuration Options Register

The clock monitor function is enabled or disabled by the CME control bit in the OPTION register (see **Figure 5-4**). The FCME bit in OPTION overrides CME and enables the clock monitor until the next reset.

Address: \$0030

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

Figure 5-4. System Configuration Options Register (OPTION)

NOTE: *In normal operating modes, these bits can be written only once within 64 bus cycles after reset.*

CME — Clock Monitor Enable Bit

This control bit can be read or written at any time and controls whether or not the internal clock monitor circuit triggers a reset sequence when the system clock is slow or absent. When it is clear, the clock monitor circuit is disabled. When it is set, the clock monitor circuit is enabled. Reset clears the CME bit.

0 = Clock monitor disabled

1 = Clock monitor enabled

FCME — Force Clock Monitor Enable Bit

0 = Clock monitor follows the state of the CME bit.

1 = Clock monitor is enabled until the next reset.

Semiconductor wafer processing causes variations of the RC timeout values between individual devices. An E-clock frequency below 10 kHz generates a clock monitor error. An E-clock frequency of 200 kHz or more prevents clock monitor errors. Using the clock monitor function when the E clock is below 200 kHz is not recommended.

5.3.4.2 System Configuration Options Register 2

Address: \$0038

	Bit 7	6	5	4	3	2	1	Bit 0
Read:	LIRDV	CWOM	STRCH ⁽¹⁾	IRVNE	LSBF	SPR2	EXT4X	DISE
Write:								
Reset:	0	0	0	—	0	0	0	0

1. Not available on TK68HC11KS devices

Figure 5-5. System Configuration Options Register 2 (OPT2)

LIRDV — LIR Driven Bit

This bit allows power savings in expanded modes by turning off the $\overline{\text{LIR}}$ output (it has no meaning in single-chip or bootstrap modes). The $\overline{\text{LIR}}$ pin is driven low to indicate that execution of an instruction has begun. To detect consecutive instructions in a high-speed application, this signal drives high for a quarter of a cycle to prevent false triggering. An external pullup is required in expanded modes, while a hardwired V_{SS} connection is possible in single-chip modes. LIRDV is reset to 0 in single-chip modes and to 1 in expanded modes.

- 1 = Enable $\overline{\text{LIR}}$ push-pull drive
- 0 = $\overline{\text{LIR}}$ not driven on MODA/ $\overline{\text{LIR}}$ pin

CWOM — Port C Wired-OR Mode Bit

For detailed information, refer to **Section 6. Parallel Input/Output**.

- 1 = Port C outputs are open drain.
- 0 = Port C operates normally.

STRCH — Stretch External Accesses Bit

When this bit is set, off-chip accesses of selected addresses are extended by one E-clock cycle to allow access to slow peripherals. The E clock stretches externally, but the internal clocks are not affected, so that timers and serial systems are not corrupted. The state of the ROMAD bit in the CONFIG register determines which address range is affected.

- 1 = Off-chip accesses are selectively extended by one E-clock cycle.
- 0 = Normal operation

NOTE: *STRCH is cleared on reset; therefore, a program cannot execute out of reset in a slow external ROM.*

To use the STRCH feature, ROMON must be set on reset so that the device starts with internal ROM included in the memory map. STRCH should then be set.

STRCH has no effect in single-chip and bootstrap modes.

NOTE: *STRCH is not available on TK68HC11KS devices.*

IRVNE — Internal Read Visibility/Not E Bit

IRVNE can be written once in any user mode. In expanded modes, IRVNE determines whether IRV is on or off (but has no meaning in user expanded secure mode, as IRV must be disabled). In special test mode, IRVNE is reset to 1. In normal modes, IRVNE is reset to 0.

1 = Data from internal reads is driven out of the external data bus.

0 = No visibility of internal reads on external bus

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

1 = E pin is driven low.

0 = E clock is driven out from the chip.

Refer to **Table 5-3** for a summary of the operation immediately following reset.

Table 5-3. IRVNE Operation After Reset

Mode	IRVNE after Reset	E Clock after Reset	IRV after Reset	IRVNE Affects Only	IRVNE Can Be Written
Single-chip	0	On	Off	E	Once
Expanded	0	On	Off	IRV	Once
Bootstrap	0	On	Off	E	Unlimited
Special test	1	On	On	IRV	Unlimited

LSBF — Least Significant Bit (LSB) First Enable Bit

For detailed information, refer to **Section 8. Serial Peripheral Interface (SPI)**.

1 = Data is transferred LSB first.

0 = Data is transferred MSB (most significant bit) first.

SPR2 — SPI Clock Rate Selected Bit

This bit adds a divide-by-four to the SPI clock chain. For detailed information, refer to **Section 8. Serial Peripheral Interface (SPI)**.

EXT4X — 4XCLK or EXTAL Clock Output Select Bit

This bit can be written once and can be read at any time.

This bit selects which clock is to be output on the 4XOUT pin, when enabled by the CLK4X bit in the CONFIG register. On reset, or when BCS = 0, 4XCLK (the PLL output) is the same as EXTALi. There is a phase delay between EXTALi and 4XOUT.

1 = EXTALi clock output on the 4XOUT pin

0 = 4XCLK clock output on the 4XOUT pin

NOTE: *The 4XOUT pin is not available on 84-pin packaged devices.*

DISE — E Clock Output Disable Bit

This bit can be written once and can be read at any time.

IRVNE allows E clock to be turned off in single-chip modes. DISE has been added for expanded modes, but can be used in every mode. Writing a 0 to this bit prevents accidental E clock turn-off in systems requiring this signal.

1 = No E clock output

0 = E clock is output normally.

5.4 Effects of Reset

When the MCU recognizes a reset condition, it forces the CPU registers and control bits to established initial states. These in turn force the on-chip peripheral systems to known startup states, as described here.

- Central processor unit (CPU)
 - The stack pointer and other CPU registers are indeterminate immediately after reset, except for three bits in the condition code register (CCR).
 - The X and I interrupt mask bits are set to mask any interrupt requests, and the S bit in the CCR is set to inhibit the stop mode.
- Memory map
 - The INIT register is initialized to \$00, putting the control registers at locations \$0000–\$007F.
 - The 1.5 Kbytes of RAM are at locations \$0080–\$067F except for the TK68HC11KS Family, which has 1 Kbytes of RAM at locations \$0080–\$047F.
 - The INIT2 register is \$00, locating the EEPROM at \$0D80–\$0FFF.
- Timer
 - The timing system is initialized to a count of \$0000.
 - The prescaler bits are cleared, and all output compare registers are initialized to \$FFFF.
 - All input capture registers are indeterminate after reset.
 - The output compare 1 mask (OC1M) register is cleared so that successful OC1 compares do not affect any input/output (I/O) pins. The other four output compares are configured so that they do not affect any I/O pins on successful compares.
 - All input capture edge-detector circuits are configured for capture disabled operation.
 - The timer overflow interrupt flag and all eight timer function interrupt flags are cleared.
 - All nine timer interrupts are disabled because their mask bits have been cleared.
 - The I4/O5 bit in the PACTL register is cleared to configure the I4/O5 function as OC5; however, the OM5:OL5 control bits in the TCTL1 register are clear so OC5 does not control the PA3 pin.
 -

- Real-time interrupt (RTI)
 - The RTI enable bit in TKSK2 is cleared, masking automatic hardware interrupts.
 - The rate control bits are cleared after reset and can be initialized by software before the RTI system is enabled.
- Pulse accumulator
 - The pulse accumulator system is disabled at reset.
 - The PAI input pin defaults to a general-purpose input pin (PA7).
- Computer operating properly (COP) watchdog system
 - The COP watchdog system is enabled if the NOCOP control bit in the CONFIG register is clear and disabled if NOCOP is set.
 - The OPTION register's CR[1:0] bits are cleared, setting the COP rate for the shortest duration timeout.
- Serial communications interface (SCI)
 - At reset, the SCI baud rate control register (**7.9.1 SCI Baud Rate Control Register**) is initialized to \$0004.
 - All transmit and receive interrupts are masked and both the transmitter and receiver are disabled so the port pins default to being general-purpose I/O lines.
 - The SCI frame format is initialized to an 8-bit character size.
 - The send break and receiver wake-up functions are disabled.
 - The TDRE and TC status bits in the SCI status register are both set, indicating that there is no transmit data in either the transmit data register or the transmit serial shift register.
 - The RDRF, IDLE, OR, NF, FE, PF, and RAF receive-related status bits are cleared.
- Serial peripheral interface (SPI)
 - The SPI system is disabled by reset.
 - The port pins associated with this function default to being general-purpose I/O lines.
- Analog-to-digital (A/D) converter
 - The ADPU bit in the OPTION register is cleared, disabling the A/D system.
 - The conversion complete flag in the ADCTL register is also cleared.
- System

- The external $\overline{\text{IRQ}}$ pin has the highest I-bit interrupt priority because PSEL[4:0] in the HPRIO register are initialized with the value %00110 (where % indicates a binary value).
- The RBOOT, SMOD, and MDA bits in the HPRIO register reflect the status of the MODB and MODA inputs at the rising edge of reset.
- The $\overline{\text{IRQ}}$ pin is configured for level-sensitive operation for wired-OR systems.
- The DLY control bit in the OPTION register is set, enabling oscillator startup delay after recovery from stop mode.
- The clock monitor system is disabled because the CME and FCME bits in the OPTION register are cleared.

5.5 Interrupts

The MCU has 18 interrupt vectors that support 22 interrupt sources. The 19 maskable interrupts are generated by on-chip peripheral systems. They are recognized when the I bit in the CCR is clear. The three non-maskable interrupt sources are illegal opcode trap, software interrupt, and $\overline{\text{XIRQ}}$ pin. **Table 5-4** lists the interrupt sources and vector assignments for each source.

Table 5-4. Interrupt and Reset Vector Assignments

Vector Address	Interrupt Source	CC Register Mask	Local Mask
FFC0, C1 — FFD4, D5	Reserved	—	—
FFD6, D7	SCI serial system: <ul style="list-style-type: none"> • SCI transmit complete • SCI transmit data register empty • SCI idle line detect • SCI receiver overrun • SCI receive data register full 	I bit	TCIE TIE ILIE RIE RIE
FFD8, D9	SPI serial transfer complete	I bit	SPIE
FFDA, DB	Pulse accumulator input edge	I bit	PAII
FFDC, DD	Pulse accumulator overflow	I bit	PAOVI
FFDE, DF	Timer overflow	I bit	TOI
FFE0, E1	Timer input capture 4/output compare 5	I bit	I4/O5I
FFE2, E3	Timer output compare 4	I bit	OC4I
FFE4, E5	Timer output compare 3	I bit	OC3I

Table 5-4. Interrupt and Reset Vector Assignments (Continued)

Vector Address	Interrupt Source	CC Register Mask	Local Mask
FFE6, E7	Timer output compare 2	I bit	OC2I
FFE8, E9	Timer output compare 1	I bit	OC1I
FFEA, EB	Timer input capture 3	I bit	IC3I
FFEC, ED	Timer input capture 2	I bit	IC2I
FFEE, EF	Timer input capture 1	I bit	IC1I
FFF0, F1	Real-time interrupt	I bit	RTI
FFF2, F3	$\overline{\text{IRQ}}$ (external pin)	I bit	None
FFF4, F5	$\overline{\text{XIRQ}}$ pin	X bit	None
FFF6, F7	Software interrupt	None	None
FFF8, F9	Illegal opcode trap	None	None
FFFA, FB	COP failure	None	NOCOP
FFFC, FD	Clock monitor fail	None	CME
FFFE, FF	RESET	None	None

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

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

Table 5-5. Stacking Order on Entry to Interrupts

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

5.5.1 Non-Maskable Interrupts

Non-maskable interrupts can interrupt CPU operations at any time. The most common use for such an interrupt is for serious system problems, such as program runaway or power failure. The three sources of non-maskable interrupt are:

- \overline{XIRQ} pin
- Illegal opcode trap
- Software interrupt instruction (SWI)

5.5.1.1 Non-Maskable Interrupt Request (\overline{XIRQ})

The \overline{XIRQ} input is an updated version of the non-maskable \overline{NMI} input of earlier MCUs. Upon reset, both the X bit and I bit of the CCR are set to inhibit all maskable interrupts and \overline{XIRQ} . After minimum system initialization, software can clear the X bit by a transfer from accumulator A to condition code register (TAP) instruction, enabling \overline{XIRQ} interrupts. Thereafter, software cannot set the X bit and the \overline{XIRQ} interrupt becomes non-maskable.

I bit-related interrupts do not affect the X bit, which has a higher priority than they do in the interrupt priority logic. When an I bit-related interrupt occurs, the CPU sets the I bit after stacking the CCR byte, but the X bit remains unaffected. When an X bit-related interrupt occurs, the CPU sets both the X and I bits after stacking the CCR. The RTI instruction restores the X and I bits to their pre-interrupt request state when it pulls the CCR from the stack.

5.5.1.2 Illegal Opcode Trap

The MCU includes an illegal opcode detection circuit to avoid attempting to process undefined opcodes or opcode sequences. This mechanism works for all unimplemented opcodes on all four opcode map pages. When the circuit detects an illegal opcode, it generates an interrupt. The CPU responds by pushing the current value of the program counter, which is actually the address of the first byte of the illegal opcode, on the stack. The illegal opcode service routine can use this stacked address as a pointer to the illegal opcode to correct it. To avoid repeated execution of the illegal opcode, which can lead to stack overflow, the service routine should reinitialize the stack pointer.

5.5.1.3 Software Interrupt (SWI)

SWI cannot be masked by virtue of the fact that it is a software instruction. It is not inhibited by the global mask bits in the CCR. Execution of SWI sets the I

mask bit, so other interrupts are inhibited until user software clears the I bit or SWI terminates with an RTI instruction.

5.5.2 Maskable Interrupts

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

Address: \$0039

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

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

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

IRQE — Configure IRQ for Edge-Sensitive Operation Bit

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

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

5.6 Reset and Interrupt Priority

A hardware priority scheme determines which reset or interrupt is serviced first when simultaneous requests occur.

The six highest-priority interrupt sources are not maskable. The priority arrangement for these sources is:

1. POR or $\overline{\text{RESET}}$ pin
2. Clock monitor reset
3. COP watchdog reset
4. $\overline{\text{XIRQ}}$ interrupt
5. Illegal opcode interrupt
6. Software interrupt (SWI)

The maskable interrupt sources have this priority arrangement:

1. IRQ
2. Real-time interrupt
3. Timer input capture 1
4. Timer input capture 2
5. Timer input capture 3
6. Timer output compare 1
7. Timer output compare 2
8. Timer output compare 3
9. Timer output compare 4
10. Timer input capture 4/output compare 5
11. Timer overflow
12. Pulse accumulator overflow
13. Pulse accumulator input edge
14. SPI transfer complete
15. SCI system

Any single maskable interrupt can be given priority over other maskable interrupts by writing the appropriate value to the PSEL bits in the HPRIO register (see **Figure 5-7**). An interrupt that is assigned highest priority is still subject to global masking by the I bit in the CCR or by any associated local bits. Interrupt vectors are not affected by priority assignment.

Address: \$003C

	Bit 7	6	5	4	3	2	1	Bit 0
Read:	RBOOT	SMOD	MDA	PSEL4	PSEL3	PSEL2	PSEL1	PSEL0
Write:								
Reset:	0	0	0	1	0	0	0	0

Figure 5-7. Highest Priority I-Bit Interrupt and Miscellaneous Register (HPRIO)

NOTE: To avoid race conditions, HPRIO is designed so that it can be written only while I-bit interrupts are inhibited (for example, the I bit is set).

PSEL[4:0] — Priority Select Bits

These bits select one interrupt source to have the highest priority, as explained in **Table 5-6**.

5.7 Reset and Interrupt Processing

This section presents flow diagrams of the reset and interrupt processes. **Figure 5-8** illustrates how the CPU begins from a reset and how interrupt detection relates to normal opcode fetches. **Figure 5-9** is an expansion of a block in **Figure 5-8** and illustrates interrupt priorities. **Figure 5-10** shows the resolution of interrupt sources within the SCI subsystem.

Table 5-6. Highest Priority Interrupt Selection

PSELx					Interrupt Source Promoted
4	3	2	1	0	
0	0	0	X	X	Reserved (default to $\overline{\text{IRQ}}$)
0	0	1	0	0	Reserved (default to $\overline{\text{IRQ}}$)
0	0	1	0	1	Reserved (default to $\overline{\text{IRQ}}$)
0	0	1	1	0	$\overline{\text{IRQ}}$
0	0	1	1	1	Real-time interrupt
0	1	0	0	0	Timer input capture 1
0	1	0	0	1	Timer input capture 2
0	1	0	1	0	Timer input capture 3
0	1	0	1	1	Timer output compare 1
0	1	1	0	0	Timer output compare 2
0	1	1	0	1	Timer output compare 3
0	1	1	1	0	Timer output compare 4
0	1	1	1	1	Timer output compare 5/input capture 4
1	0	0	0	0	Timer overflow
1	0	0	0	1	Pulse accumulator overflow
1	0	0	1	0	Pulse accumulator input edge
1	0	0	1	1	SPI serial transfer complete
1	0	1	0	0	SCI serial system
1	0	1	0	1	Reserved (default to $\overline{\text{IRQ}}$)
1	0	1	1	0	Reserved (default to $\overline{\text{IRQ}}$)
1	0	1	1	1	Reserved (default to $\overline{\text{IRQ}}$)
1	1	X	X	X	Reserved (default to $\overline{\text{IRQ}}$)

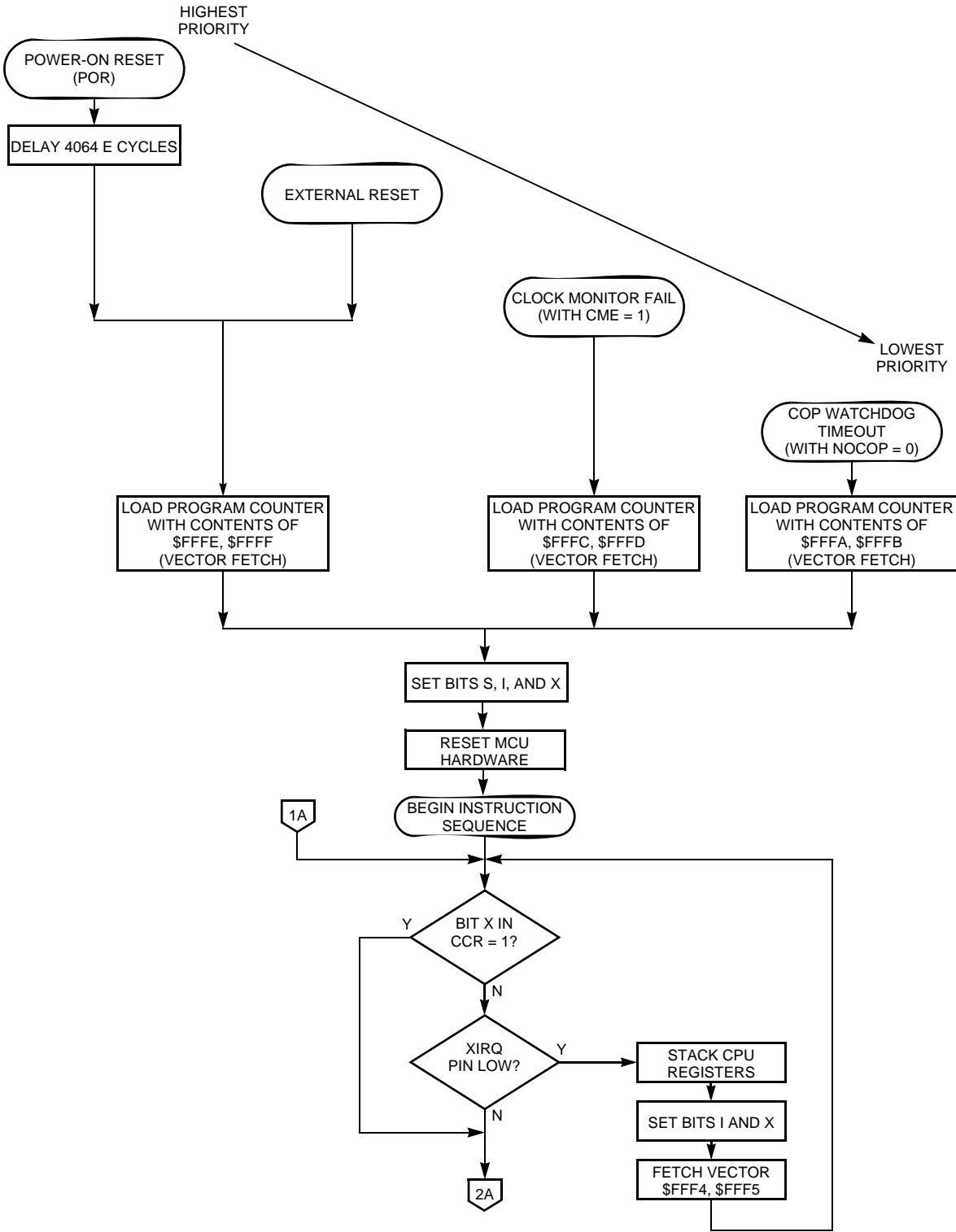


Figure 5-8. Processing Flow Out of Reset (Sheet 1 of 2)

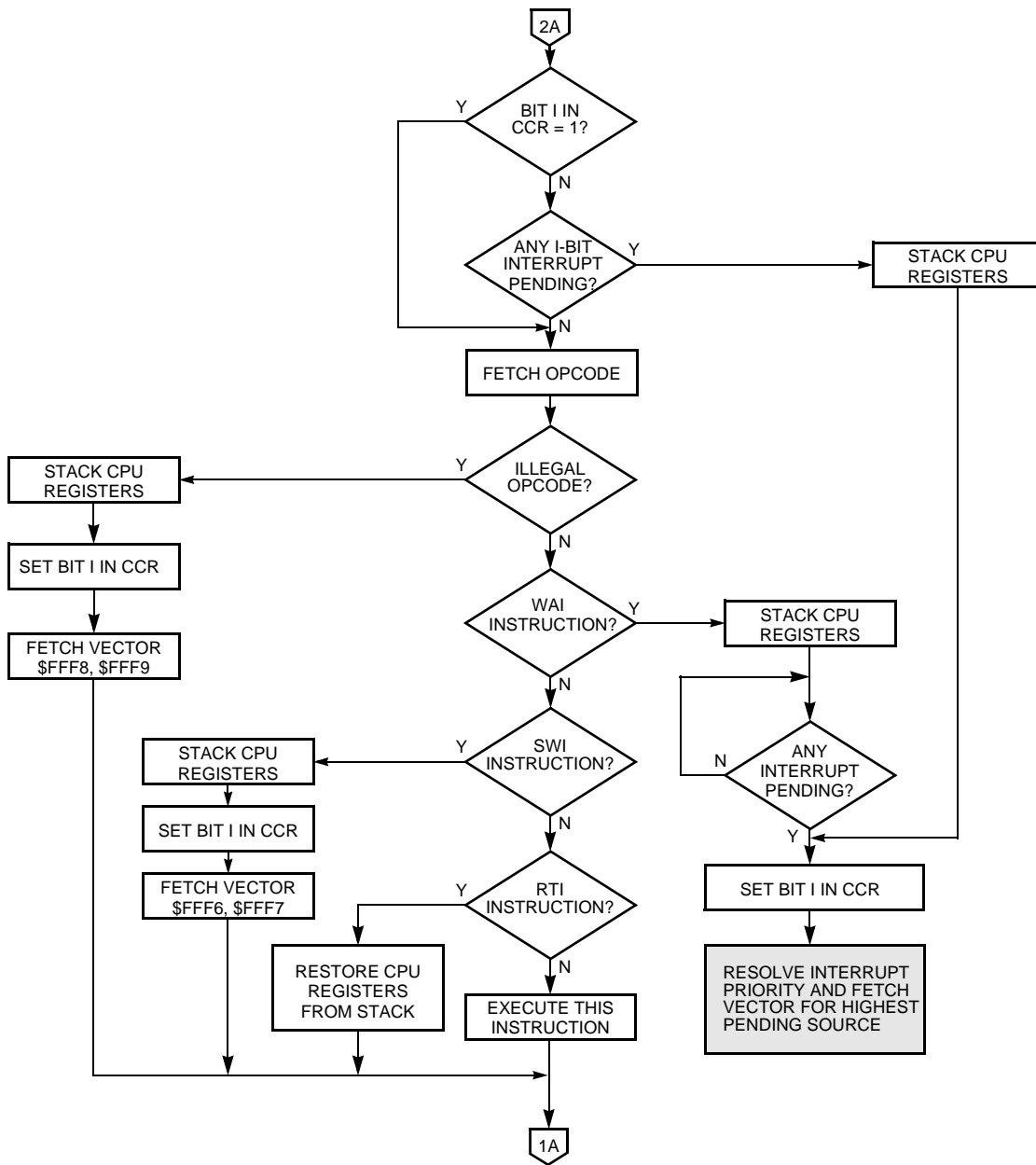


Figure 5-8. Processing Flow Out of Reset (Sheet 2 of 2)

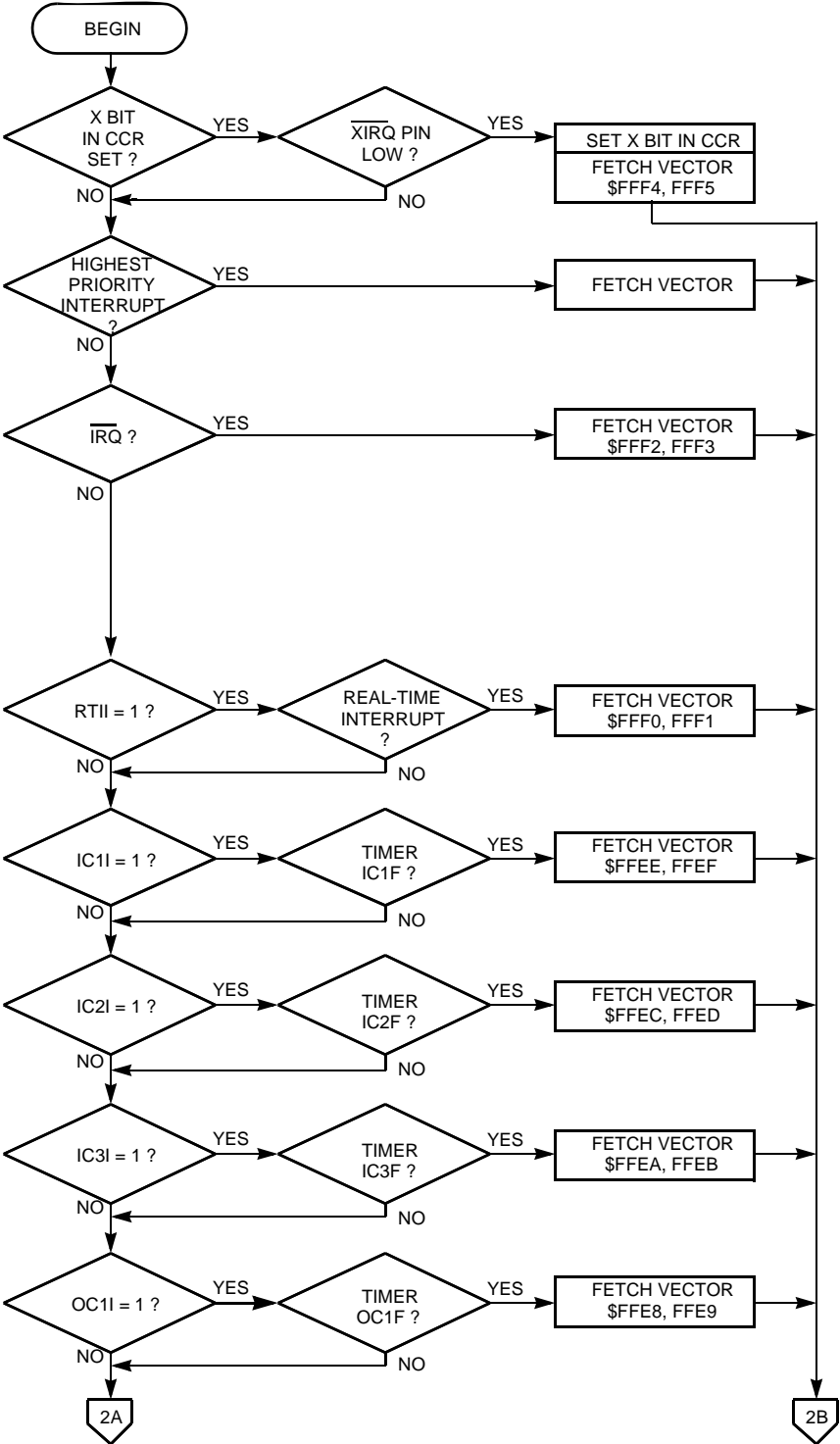


Figure 5-9. Interrupt Priority Resolution (Sheet 1 of 2)

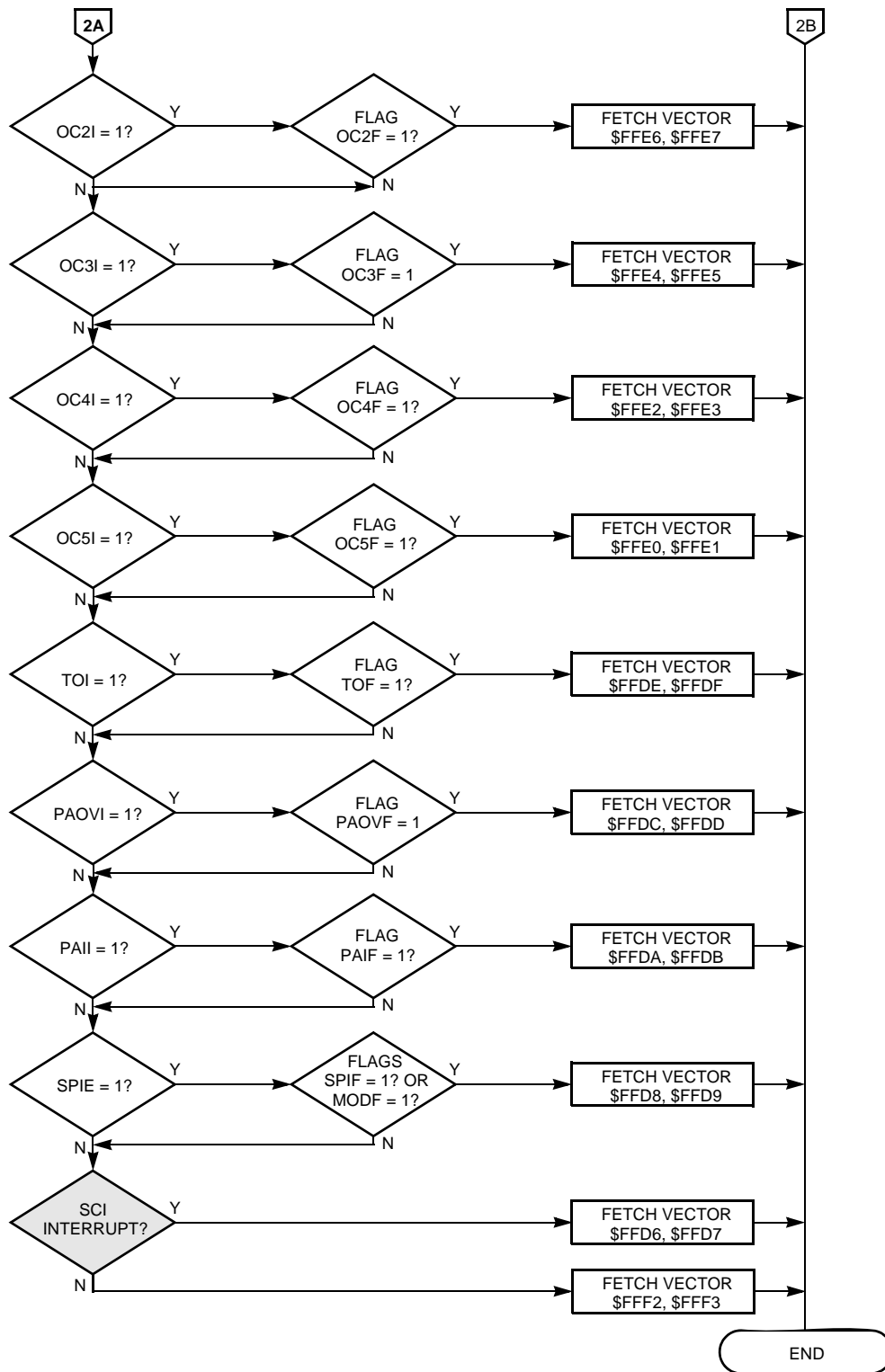


Figure 5-9. Interrupt Priority Resolution (Sheet 2 of 2)

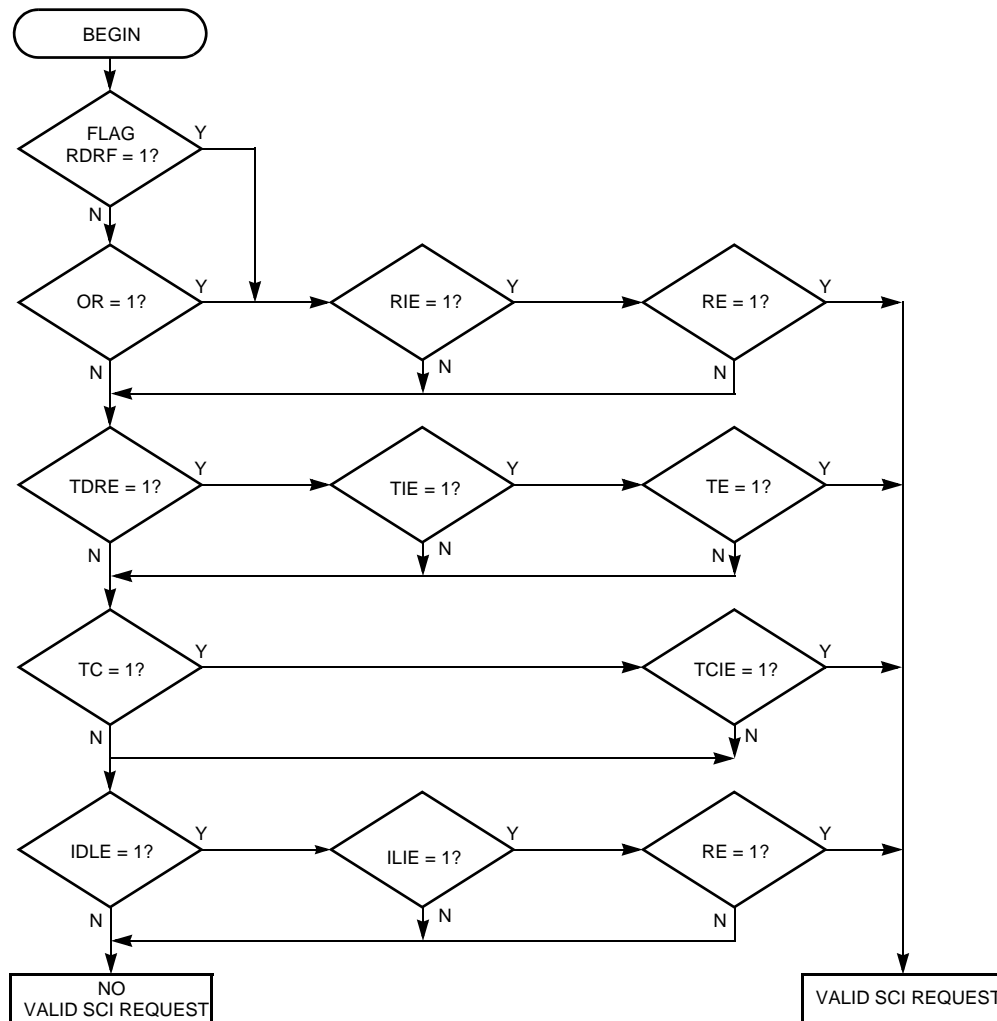


Figure 5-10. Interrupt Priority Resolution Within SCI System

5.8 Low-Power Operation

The MCU contains two software instructions, WAIT and STOP, to reduce power consumption when processing is not required. Both instructions suspend operation until a reset or interrupt occurs while retaining register and RAM contents. The wait condition suspends processing, reducing power consumption to an intermediate level. The stop condition turns off all on-chip clocks as well and reduces power consumption to an absolute minimum.

5.8.1 Wait Mode

The WAI opcode places the MCU in the wait condition, during which the CPU registers are stacked and CPU processing is suspended until a qualified interrupt is detected. The interrupt can be an external $\overline{\text{IRQ}}$, an $\overline{\text{XIRQ}}$, or any of the internally generated interrupts, such as the timer or serial interrupts. The on-chip crystal oscillator remains active throughout the wait standby period.

The reduction of power in the wait condition depends on how many internal clock signals driving on-chip peripheral functions can be shut down. The CPU is always shut down during WAIT. While in the wait state, the address/data bus repeatedly runs read cycles to the address where the CCR contents were stacked. The MCU leaves the wait state when it senses any interrupt that has not been masked.

The free-running timer system is shut down only if maskable interrupts are disabled (I bit is set) and the COP system is disabled (NOCOP is set). Other systems can be shut down through the software-controlled configuration control bits, including the SPI system (SPE control bit), the SCI transmitter (TE bit), and the SCI receiver (RE bit). Net power reduction in WAIT depends on which of these features is disabled.

5.8.2 Stop Mode

The STOP instruction halts all system clocks, including the crystal oscillator, thereby minimizing power consumption. The S bit in the CCR must be cleared to place the MCU in the stop condition; otherwise, the stop opcode is treated as a no-operation (NOP). To exit STOP and resume normal processing, a logic low level must be applied to one of the external interrupt pins ($\overline{\text{IRQ}}$ or $\overline{\text{XIRQ}}$) or to the $\overline{\text{RESET}}$ pin. A pending edge-triggered $\overline{\text{IRQ}}$ can also bring the CPU out of stop.

Because all clocks are stopped in this mode, all internal peripheral functions also stop. RAM and register contents are preserved as long as V_{DD} power is maintained. The CPU state and I/O pin levels are static and are not altered by STOP, so the MCU resumes processing seamlessly after the system is reactivated by an interrupt. However, if a reset is used to restart the system, a normal reset sequence results and all pins and registers are reinitialized.

To use the $\overline{\text{IRQ}}$ pin as a means of recovering from STOP, the I bit in the CCR must be clear ($\overline{\text{IRQ}}$ not masked). The $\overline{\text{XIRQ}}$ pin can be used to wake up the MCU from STOP regardless of the state of the X bit in the CCR, although the state of this bit does affect the recovery sequence. If X is clear ($\overline{\text{XIRQ}}$ not masked), the MCU executes a normal XIRQ service routine. If X is set ($\overline{\text{XIRQ}}$ masked or inhibited), then processing continues with the instruction that immediately follows the STOP instruction, and no $\overline{\text{XIRQ}}$ interrupt service is requested or pending.

Executing a STOP instruction requires special consideration when the clock monitor is enabled. Because the stop function halts all clocks, the clock monitor function will generate a reset sequence if it is enabled at the time the stop mode was initiated. To prevent this, clear the CME and FCME bits in the OPTION register before executing a STOP instruction to disable the clock monitor. After recovery from STOP, set the CME bit to enable the clock monitor. Systems using the internal oscillator require a delay after restart upon leaving STOP to allow the oscillator to stabilize. If a stable external oscillator is used, the DLY control bit in the OPTION register can be used to bypass this startup delay (see **Figure 5-11**). Reset sets the DLY control bit; it can be cleared during initialization. Do not use reset to recover from STOP if the DLY is to be bypassed, since reset sets the DLY bit again, causing the restart delay. This same delay will follow a power-on reset, regardless of the state of the DLY control bit, but does not apply to a reset while the clocks are running.

Address: \$0039

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

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

Figure 5-11. System Configuration Options Register (OPTION)

DLY — Enable Oscillator Startup Delay Bit

This bit is set during reset and can be written only once during the first 64 E-clock cycles after reset in normal modes. This bit can be used to inhibit the oscillator startup delay after reset when using an external clock source.

0 = No stabilization delay on exit from STOP

1 = Stabilization delay enabled on exit from STOP

5.8.3 Slow Mode

Slow mode is a software selectable mode that allows the user to connect, under software control, an extra divide-by-16 between the oscillator and the internal clock. This feature permits a slow down of all the internal operations reducing power consumption.

When WAI is used for power reduction, the slow mode helps further reduce the power. Control of slow mode is performed in the system configuration options 3 register (OPT3). See **Figure 5-12**.

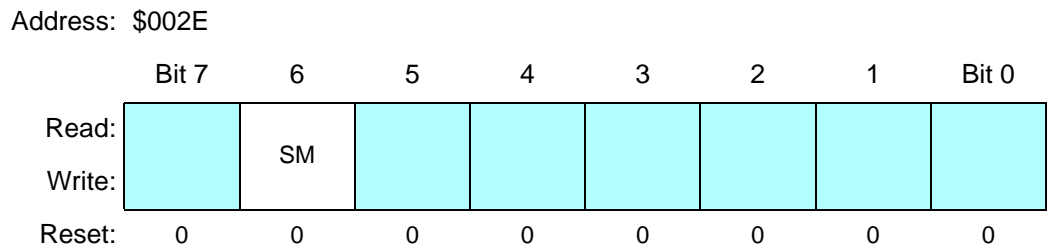


Figure 5-12. System Configuration Options 3 Register (OPT3)

SM — Slow-Mode Enable Bit

Read and write at any time

1 = When the SM bit is asserted, a 16-clock divider is connected between the oscillator divide-by-4 and the bus clock. This causes the system clock to run 16 times slower than normal. All modules of the MCU slow down, including the timer, SCI, SPI, and A/D. It is also cleared in hardware when entering stop mode or when reset, including POR, is asserted low.

0 = When the SM bit is negated, the divider is disconnected and the system runs at normal bus speed.

NOTE: *The slow mode function should not be enabled while using the A/D converter or during an erase/program operation of the EEPROM, unless the internal RC oscillator is turned on.*

The clock monitor function should not be used if the resultant E clock will be slower than 200 kHz.

Resets and Interrupts

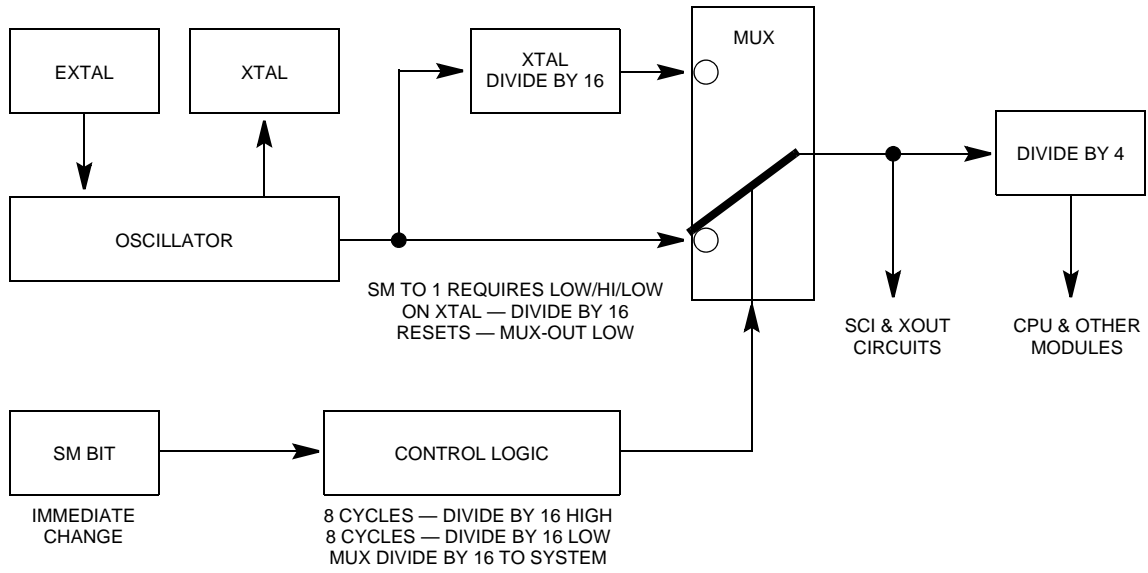


Figure 5-13. Slow Mode Example

Section 6. Parallel Input/Output

6.1 Contents

6.2	Introduction	132
6.3	Port A	134
6.4	Port B	135
6.5	Port C	136
6.6	Port D	138
6.7	Port E	139
6.8	Port F	140
6.9	Port G	141
6.10	Port H	142
6.11	Internal Pullup Resistors.	143

6.2 Introduction

The TK68HC11K series MCUs contain eight input/output (I/O) ports, A through H. All ports can provide general-purpose I/O (GPIO) as well as their specialized functions, as explained in **2.11 Port Signals** and summarized in **Table 6-1**.

Table 6-1. Port Configuration

Port	Input Pins	Output Pins	Bidirectional Pins	Shared Functions
Port A	—	—	8	Timer
Port B	—	—	8	High-order address
Port C	—	—	8	Data bus
Port D	—	—	6	SCI and SPI
Port E	8	—	—	A/D converter
Port F	—	—	8	Low-order address
Port G	—	—	g ⁽¹⁾	Memory expansion
Port H	—	—	g ⁽²⁾	Chip selects and PWM

1. KS devices do not contain port G[6:0], so they have only one bidirectional pin on this port.
2. KS devices do not contain port H[7:4], so they have only four bidirectional pins on this port.

Each of the ports has an associated data register (PORTx). Each port, except port E, also has an associated data direction register (DDRx). When a port is configured for GPIO, its DDR determines whether port pins function as inputs or outputs. A port's special functions override the DDR when they are enabled.

Writes to any port, except port E, are stored in internal latches. The latches drive the port pins only when they are configured as general-purpose outputs.

When software reads a port pin configured for GPIO, the MCU returns the physical pin level, not the port register value. This applies to both inputs and outputs. The only exception applies to ports C and D in wired-OR mode. When they are configured as outputs, a read returns the pin driver levels.

Ports B, F, G, and H contain on-chip pullup devices which are enabled by the port pullup assignment register (PPAR) described in **6.11 Internal Pullup Resistors**.

At reset, the ports are configured as high-impedance GPIO inputs (except for ports B, C, F, and port G pin 7 in expanded modes). The contents of the data latches is undefined. If any of the bidirectional pins are changed to outputs before writing to the associated data registers, the undefined contents will be

driven on the pins. This is indicated by the letter U in the register descriptions that follow.

NOTE: *Throughout this manual, the registers are discussed by function. In the event that not all bits in a register are referenced, the bits that are not discussed are shaded.*

6.3 Port A

Port A provides the I/O lines for the timer functions and pulse accumulator. The eight port A bits (PA[7:0]) are configured as high-impedance general-purpose inputs out of reset. Writes to DDRA can change any of the bits to outputs. Writes to timer registers enable the various timer functions (see **Section 9. Timing System**).

Address: \$0000

	Bit 7	6	5	4	3	2	1	Bit 0
Read:	PA7	PA6	PA5	PA4	PA3	PA2	PA1	PA0
Write:								
Reset:	Undefined after reset							
Alternate Pin Function:	PA1	OC2	OC3	OC4	IC4/OC5	IC1	IC2	IC3
And/or:	OC1	OC1	OC1	OC1	OC1	—	—	—

Figure 6-1. Port A Data Register (PORTA)

Address: \$0001

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

Figure 6-2. Port A Data Direction Register (DDRA)

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

6.4 Port B

The state of port B (PB[7:0]) at reset is mode dependent. In single-chip or bootstrap modes, port B pins are high-impedance inputs with selectable internal pullup resistors (see **6.11 Internal Pullup Resistors**). Writes to DDRB can change any of the bits to outputs. In expanded or test modes, port B pins provide the high-order address lines ADDR[15:8] for external memory devices.

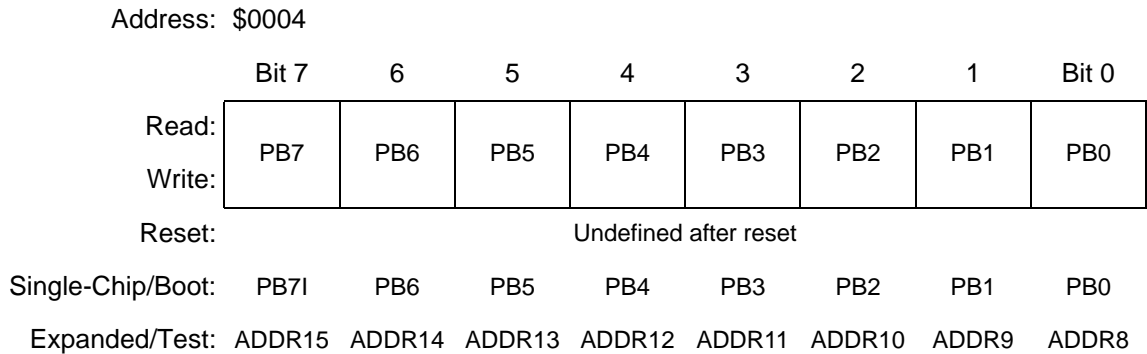


Figure 6-3. Port B Data Register (PORTB)

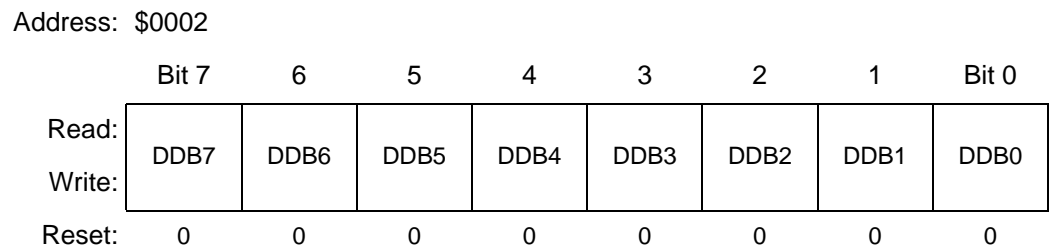


Figure 6-4. Port B Data Direction Register (DDRB)

DDB[7:0] — Data Direction for Port B Bits

0 = Input

1 = Output

6.5 Port C

The state of port C at reset is mode dependent. In single-chip or bootstrap modes, port C pins (PC[7:0]) are high-impedance inputs. Writes to DDRC can change any of the bits to outputs. In expanded or test modes, port C pins provide the data lines (DATA[7:0]) for external memory devices. The MCU's internal data bus can also be driven on port C by setting the IRVNE bit in the system configuration options register (OPT2). See **Figure 6-7**.

When port C functions as GPIO (single-chip mode), it can be configured for wired-OR operation by setting the CWOM bit in the OPT2 register. This disables port C's P-channel drivers, effectively generating open-drain-type outputs. To output a logic 0 on a wired-OR pin, the MCU turns on its N-channel driver. To generate a logic 1, both P- and N-channel drivers are turned off, presenting a high-impedance state which requires an external pullup resistor to apply the appropriate voltage level.

Address: \$0006

	Bit 7	6	5	4	3	2	1	Bit 0
Read:	PC7	PC6	PC5	PC4	PC3	PC2	PC1	PC0
Write:								
Reset:	Undefined after reset							
Single-Chip/Boot:	PC7I	PC6	PC5	PC4	PC3	PC2	PC1	PC0
Expanded/Test:	DATA7	DATA6	DATA5	DATA4	DATA3	DATA2	DATA1	DATA0

Figure 6-5. Port C Data Register (PORTC)

Address: \$0007

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

Figure 6-6. Port C Data Direction Register (DDRC)

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

Address: \$0038

	Bit 7	6	5	4	3	2	1	Bit 0
Read:	LIRDV	CWOM	STRCH ₃	IRVNE	LSBF	SPR2	EXT4X	DISE
Write:								
Reset:	0	0	0	—	0	0	0	0

Figure 6-7. System Configuration Options 2 Register (OPT2)

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

IRVNE — Internal Read Visibility/Not E Bit

In expanded modes, setting this bit drives MCU's internal data bus on port-C.

0 = No internal read visibility on external data bus
 1 = Data from internal reads is driven on port C.

In single-chip modes, setting this bit inhibits the E clock driver, and the E pin is pulled low

0 = E clock drives the E pin.
 1 = E pin is driven low.

NOTE: *IRVNE can be written only once after reset. The default value of IRVNE after reset is low.*

6.6 Port D

The six port D bits, PD[5:0] function as the serial communication interface (see **Section 7. Serial Communications Interface (SCI)**) and the serial peripheral interface (see **Section 8. Serial Peripheral Interface (SPI)**) when these functions are enabled by software. They are high-impedance general-purpose inputs out of reset; DDRD can be used to change any of the pins to outputs.

Address: \$0008

	Bit 7	6	5	4	3	2	1	Bit 0
Read:	0	0	PD5	PD4	PD3	PD2	PD1	PD0
Write:	0	0	PD5	PD4	PD3	PD2	PD1	PD0
Reset:	0	0	U	U	U	U	U	U
Alternate Pin Function:	—	—	\overline{SS}	SCK	MOSI	MISO	TxD	RxD

U = Undefined

Figure 6-8. Port D Data Register (PORTD)

Address: \$0009

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

Figure 6-9. Port D Data Direction Register (DDRD)

DDD[5:0] — Data Direction for Port D Bits

0 = Input

1 = Output

6.7 Port E

Port E, PE[7:0], is the only port that functions as input only, and its pins are configured as high-impedance inputs out of reset. It also serves as the analog input for the analog-to-digital converter when this function is enabled by software (see **Section 10. Analog-to-Digital (A/D) Converter**).

NOTE: *PORT E should not be read during the sample portion of an A/D conversion.*

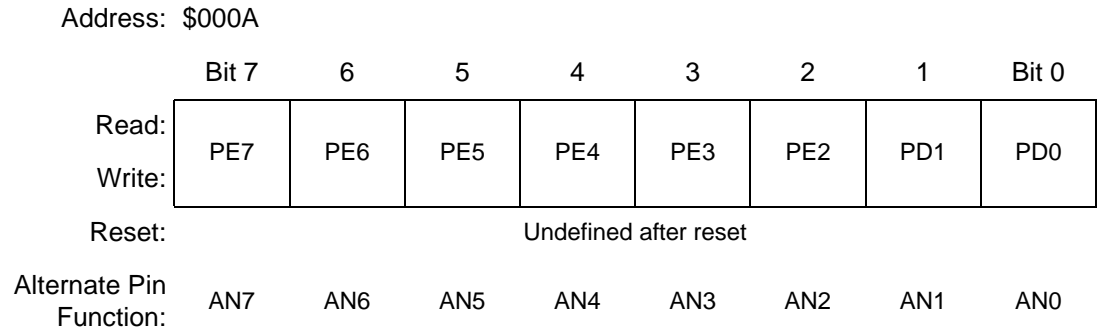


Figure 6-10. Port E Data Register (PORTE)

6.8 Port F

The state of port F (PF[7:0]) at reset is mode dependent. In single-chip or bootstrap modes, port F pins are high-impedance inputs with selectable internal pullup resistors (see **6.11 Internal Pullup Resistors**). Writes to DDRF can change any of the bits to outputs. In expanded or test modes, port F pins provide low-order address lines, ADDR[7:0], for external memory devices.

Address: \$0005

	Bit 7	6	5	4	3	2	1	Bit 0
Read:	PF7	PF6	PF5	PF4	PF3	PF2	PF1	PF0
Write:								
Reset:	Undefined after reset							
Single-Chip/Boot:	PF7	PF6	PF5	PF4	PF3	PF2	PF1	PF0
Expanded/Test:	AN7	AN6	AN5	AN4	AN3	AN2	AN1	AN0

Figure 6-11. Port F Data Register (PORTF)

Address: \$0003

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

Figure 6-12. Port F Data Direction Register (DDRF)

DDF[7:0] — Data Direction for Port F Bits

0 = Input

1 = Output

6.9 Port G

The state of port G pin 7 (PG7) at reset is mode dependent. In single-chip or bootstrap modes, it is a high-impedance input; its data direction can be changed through DDRG. In expanded and special test modes, PG7 functions as the $\overline{R/W}$ line to control the direction of data flow between the MCU and external memory devices.

Port G pins (PG[6:0]) reset to high-impedance inputs in any mode. Data direction can be changed through DDRG. Port G bits [5:0] can serve as memory expansion address lines (see **11.3 Memory Expansion**) in expanded and special test modes. TK68HC11KS devices do not contain these pins.

All eight port G pins have selectable internal pullup resistors (see **6.11 Internal Pullup Resistors**).

Address: \$007E

	Bit 7	6	5	4	3	2	1	Bit 0
Read:	PG7	PG6 ⁽¹⁾	PG5 ⁽¹⁾	PG4 ⁽¹⁾	PG3 ⁽¹⁾	PG2 ⁽¹⁾	PG1 ⁽¹⁾	PG0 ⁽¹⁾
Write:								
Reset:	0	0	0	0	0	0	0	0
Alternate Pin Function:	$\overline{R/W}$	—	XA18	XA17	XA16	XA15	XA14	XA13

1. Not available on KS devices

Figure 6-13. Port G Data Register (PORTG)

Address: \$007F

	Bit 7	6	5	4	3	2	1	Bit 0
Read:	DDG7	DDG6 ⁽¹⁾	DDG5 ⁽¹⁾	DDG4 ⁽¹⁾	DDG3 ⁽¹⁾	DDG2 ⁽¹⁾	DDG1 ⁽¹⁾	DDG0 ⁽¹⁾
Write:								
Reset:	0	0	0	0	0	0	0	0

1. Not available on KS devices

Figure 6-14. Port G Data Direction Register (DDRG)

DDG[7:0] — Data Direction for Port G Bits

0 = Input

1 = Output

6.10 Port H

The state of port H pin 7 (PH7) at reset is mode dependent. In single-chip or bootstrap modes, it is a high-impedance input; its data direction can be changed through DDRH. In expanded and special test modes PH7 is the program chip select line, $\overline{\text{CSPROG}}$ at reset, but can be reconfigured for GPIO (see **11.4 Chip Selects**).

Port H pins (PH[6:0]) reset to high-impedance inputs in any mode. Data direction can be changed through DDRH. Except for the TK68HC11KS devices, bits 6:4 can serve as chip select lines in expanded and special test modes (see **11.4 Chip Selects**). Pins 3:0 can be configured as pulse-width modulator outputs (see **9.10 Pulse-Width Modulator (PWM)**) in any mode.

All eight port H pins have selectable internal pullup resistors (see **6.11 Internal Pullup Resistors**).

Address: \$007C

	Bit 7	6	5	4	3	2	1	Bit 0
Read:	PH7 ⁽¹⁾	PH6 ⁽¹⁾	PH5 ⁽¹⁾	PH4 ⁽¹⁾	PH3	PH2	PH1	PH0
Write:								
Reset:	0	0	0	0	0	0	0	0
Alternate Pin Function:	CSPROG	CSPG2	CSPG1	CSIO	PW4	PS3	PS2	PS1

1. Not available on KS devices

Figure 6-15. Port H Data Register (PORTH)

Address: \$007D

	Bit 7	6	5	4	3	2	1	Bit 0
Read:	DDH7 ⁽¹⁾	DDH6 ⁽¹⁾	DDH5 ⁽¹⁾	DDH4 ⁽¹⁾	DDH3	DDH2	DDH1	DDH0
Write:								
Reset:	0	0	0	0	0	0	0	0

1. Not available on KS devices

Figure 6-16. Port H Data Direction Register (DDRH)

DDH[7:0] — Data Direction for Port H Bits

0 = Input

1 = Output

6.11 Internal Pullup Resistors

TK68HC11KS series devices contain selectable internal pullup resistors for ports B, F, G, and H. The resistors for each port are enabled by setting the corresponding bit in the PPAR register. PPAR itself must be enabled by setting the PAREN bit in the system configuration register (CONFIG). Refer to **Figure 6-17** and **Figure 6-18**.

Address: \$002C

	Bit 7	6	5	4	3	2	1	Bit 0
Read:	0	0	0	0	HPPUE	GPPUE	FPPUE	BPPUE
Write:								
Reset:	0	0	0	0	1	1	1	1

Figure 6-17. Port Pullup Assignment Register (PPAR)

xPPUE — Port x Pin Pullup Enable Bits

Only active when enabled by the PAREN bit in the CONFIG register

1 = Port x pin on-chip pullup devices disabled

0 = Port x pin on-chip pullup devices enabled

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

Address: \$003F

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

Figure 6-18. System Configuration Register (CONFIG)

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

PAREN — Pullup Assignment Register Enable Bit

0 = PPAR register disabled

1 = PPAR register enabled; pullups can be enabled through PPAR

Section 7. Serial Communications Interface (SCI)

7.1 Contents

7.2	Introduction	145
7.3	Data Format	146
7.4	Transmit Operation	147
7.5	Receive Operation	149
7.6	Wakeup Feature	152
7.7	Short Mode Idle Line Detection	153
7.8	Baud Rate Selection	153
7.9	SCI Registers	154
7.9.1	SCI Baud Rate Control Register	154
7.9.2	Serial Communications Control Register 1	156
7.9.3	Serial Communications Control Register 2	157
7.9.4	Serial Communication Status Register 1	158
7.9.5	Serial Communication Status Register 2	160
7.9.6	Serial Communications Data Register	161

7.2 Introduction

The serial communications interface (SCI) is a universal asynchronous receiver transmitter (UART) employing a standard non-return-to-zero (NRZ) format. Several baud rates are available. The SCI transmitter and receiver are independent, but they use the same data format and baud rate.

The TK68HC11K series offers several enhancements to the basic TK68HC11 SCI, including:

- 13-bit modulus prescaler in the baud generator
- Receiver-active flag

Serial Communications Interface (SCI)

- Transmitter and receiver hardware parity
- Accelerated idle line detection

7.3 Data Format

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

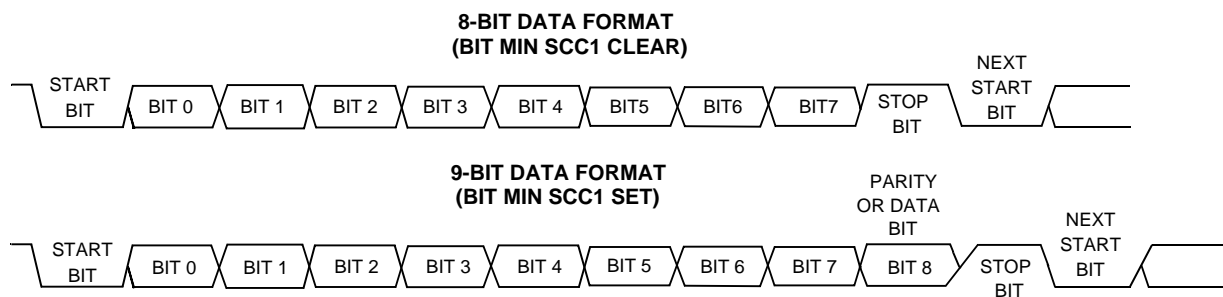


Figure 7-1. SCI Data Formats

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

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

7.4 Transmit Operation

Transmission starts by writing a data character to the 2-byte SCI data register (SCDRH and SCDRL). The MCU parallel-loads the character into a serial shift register which shifts the data out on the transmission pin. This double-buffered

operation allows transmission of the current character while the MCU loads the next one. The output of the serial shift register drives the TxD pin as long as the transmit enable (TE) bit of serial communication control register 2 (SCCR2) is set.

Two flags in serial communication status register 1 (SCSR1) alert the MCU of transmission status. The TDRE (transmit data register empty) flag is set when the SCDR loads its contents into the shift register; this flag can generate an interrupt if the TIE (transmit interrupt enable) bit in SCCR2 is set. The TC (transmit complete) flag is set when transmission is complete (line idle); this can also generate an interrupt if the TCIE (transmit complete interrupt enable) bit in SCSR1 is set. The TDRE and TC flags are normally set when software sets the TE bit to enable the transmitter. See **Figure 5-10. Interrupt Priority Resolution Within SCI System** for a flow diagram of SCI interrupts.

If interrupts are not enabled, the status flags can be read by software (polled) to determine when the corresponding conditions exist. Status flags are set automatically by hardware logic conditions, but must be cleared by software. The software clearing sequence for these flags is automatic. Functions that are normally performed in response to the status flags also satisfy the conditions of the clearing sequence.

When software clears the TE bit, the TxD pin reverts to its general-purpose I/O function (PD1). The transmitter completes transmission of a character in progress before actually shutting down; other characters waiting in the transmit queue are lost. The TC and TDRE flags are set at the completion of this last character, even though TE has been disabled. Only an MCU reset can abort transmission in midcharacter.

Figure 7-2 is a block diagram of the SCI transmitter.

Serial Communications Interface (SCI)

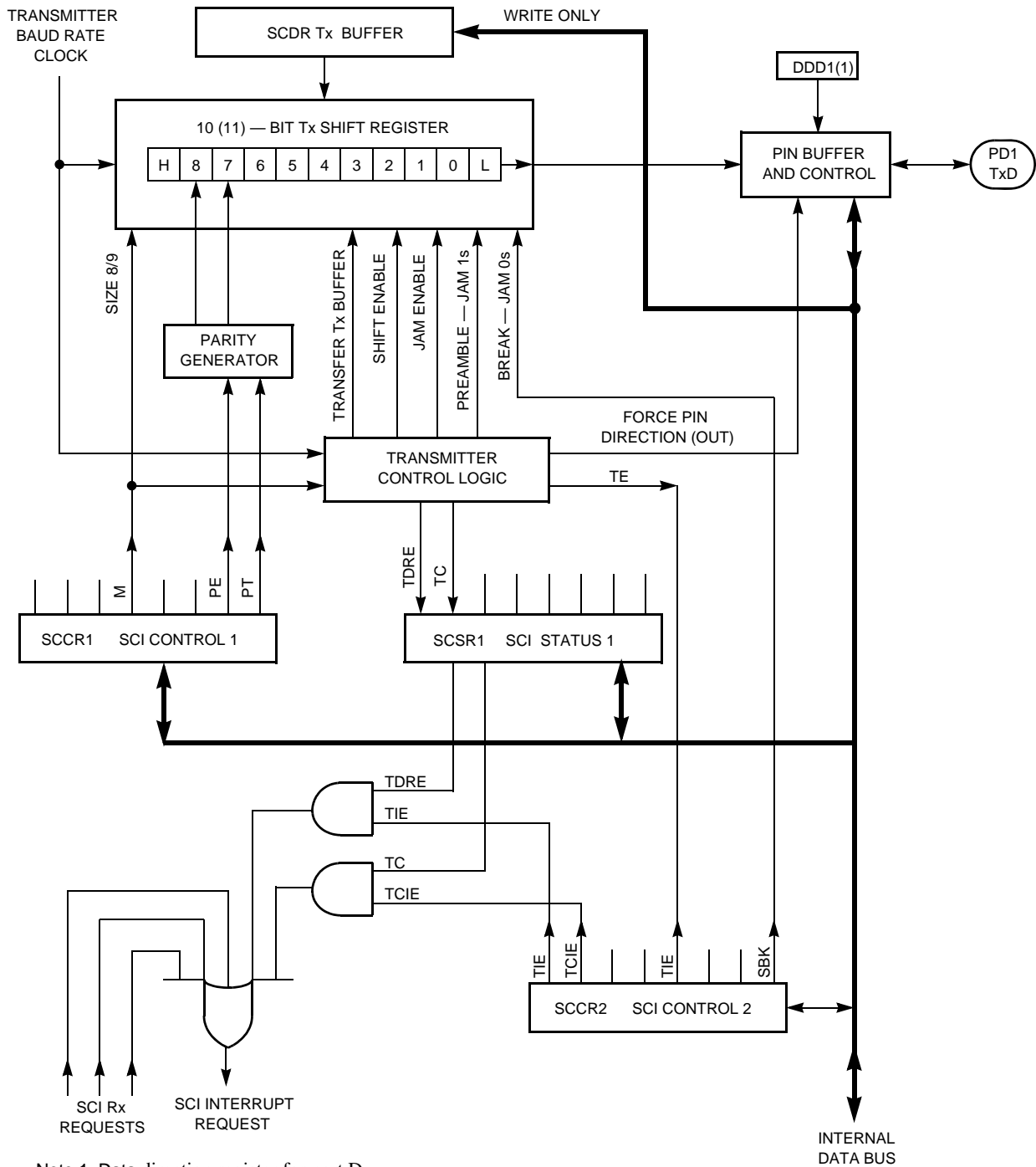


Figure 7-2. SCI Transmitter Block Diagram

7.5 Receive Operation

During receive operations, data from the TxD pin is shifted into the serial shift register. A completed word is parallel-loaded to a receive data register (RDR), which can be read through SCDRH/L. This double-buffered operation allows reception of the current character while the MCU reads the previous character.

The SCI receiver has seven status flags, summarized in **Table 7-1**.

Table 7-1. SCI Receiver Flags

Flag	Name	Set When	Interrupt Enable Bit
RDRF	Receive data register full	Character transferred from shift register to RDR	RIE
IDLE	Idle-line detected	Active transmit line goes idle	ILIE
OR	Overrun error	Character ready for RDR while previous character unread	RIE
NF	Noise error	Samples of data bit not unanimous	—
FE	Frame error	0 detected where stop bit expected	—
PF	Parity error	Calculated parity does not match data arity bit	—
RAF	Receiver active	A character is being received	—

Three of the flags can generate interrupt requests if the corresponding enable bits in SCCR2 are set. The status flags are set by the SCI logic in response to specific conditions in the receiver. These flags can be read (polled) at any time by software. Each bit except RAF is cleared by reading SCSR1 and SCDR sequentially.

- The receive data register full (RDRF) flag is set when the last bit of a character is received and data is transferred from the shift register to the RDR.
- The IDLE flag is set after a transition on the RxD line from an active state to an idle state. This prevents repeated interrupts during the time RxD remains idle.
- The overrun error (OR) flag is set instead of the RDRF bit when the next byte is ready to be transferred from the receive shift register to the RDR and the RDR is already full. The data in the shift register is lost and the data that was already in RDR is not disturbed.

Serial Communications Interface (SCI)

- The noise flag (NF) is set if there is noise on any of the received bits, including the start and stop bits. The data recovery circuit takes three samples of each bit and indicates noise if any set of three samples is not unanimous. NF is not set until the entire character is received and transferred to the RDR, when RDRF is set.
- The framing error (FE) flag is set when no stop bit is detected in the received data character. FE is set at the same time as RDRF. If the byte received causes both framing and overrun errors, the processor only recognizes the overrun error. The framing error flag inhibits further data transfer into the RDR until the flag is cleared.
- The parity error (PE) flag indicates that the parity bit of a received character does not match the parity calculated by hardware.
- The receiver active flag (RAF) is a read-only bit that is set during data reception and cleared when the line goes idle. This is the only flag cleared by hardware.

Figure 7-3 is a block diagram of the SCI receiver.



Figure 7-3. SCI Receiver Block Diagram

7.6 Wakeup Feature

The wakeup feature reduces SCI service overhead in multiple receiver systems. If a system generates address information at the beginning of every message, each receiver can determine whether or not it is the intended recipient of a message by evaluating the first character(s) through software.

If the message is intended for a different receiver, the SCI can be placed in a sleep mode so that the rest of the message will not generate requests for service. It does this by setting the RWU (receiver wakeup) bit in SCI control register 2 (SCCR2), which inhibits all receiver-related status flags (RDRF, IDLE, OR, NF, FE, PF, and RAF). A new message clears the receiver's RWU bit, enabling it to evaluate the new address information. Although RWU can be cleared by a software write to SCCR2, this is rarely done because hardware clears RWU automatically.

Two methods of wakeup are available:

- Idle line wakeup — Asleeping receiver wakes up as soon as the RxD line becomes idle (for example, in a logic 1 state for at least one frame time). A system using this type of wakeup must provide at least one character time of idle between messages to wake up sleeping receivers and must not allow any idle time between characters within a message.
- Address mark wakeup — Uses the most significant bit (MSB) to distinguish address characters (MSB = 1) from data characters (MSB = 0). A sleeping receiver wakes up whenever it receives an address character. Unlike the idle line method, address mark wakeup allows idle periods within messages and does not require idle time between messages. However, message processing is less efficient because the start bit of each character must be evaluated.

7.7 Short Mode Idle Line Detection

This feature can increase system communication speed by reducing the amount of time between messages. Setting the ILT bit in SCCR1 allows the SCI receiver to detect the consecutive 1s of an idle period before the stop bit of an incoming character is received. If the last few bits of the character are 1s, they are counted as the first high bits in the frame of 1s comprising the idle period following the character.

NOTE: *Extra care may be needed to prevent premature detection of an idle line condition.*

7.8 Baud Rate Selection

The baud rate generator for the SCI includes a 13-bit modulus prescaler driven by the system crystal clock (EXTAL). Writing to the SCI baud rate register (SCBDH/L) selects the prescaler value. See **Figure 7-4**.

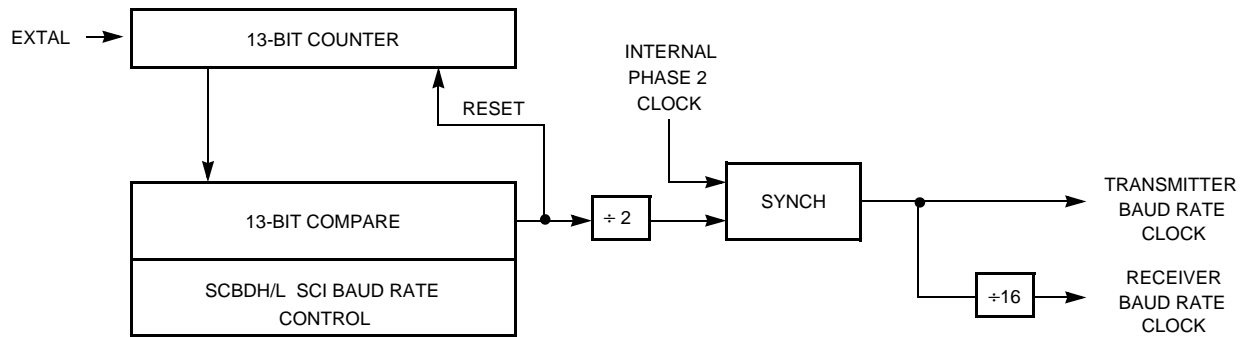


Figure 7-4. SCI Baud Generator Circuit Diagram

7.9 SCI Registers

The six addressable registers in the SCI are:

- SCI baud rate control register (SCBDH and SCBDL)
- Serial communications control register 1 (SCCR1)
- Serial communications control register 2 (SCCR2)
- Serial communication status register 1 (SCSR1)
- Serial communication status register 2 (SCSR2)
- Serial communications data register (SCDRH and SCDRL)

NOTE: Throughout this manual, the registers are discussed by function. In the event that not all bits in a register are referenced, the bits that are not discussed are shaded.

7.9.1 SCI Baud Rate Control Register

This register selects the 13-bit divisor shown in Figure 7-4 to generate the SCI baud rate. (See Table 7-2.) Normally, this register is written once during initialization, but it can be changed at any time.

Address: \$0070

	Bit 7	6	5	4	3	2	1	Bit 0
Read:	BTST	BSPL	0	SBR12	SBR11	SBR10	SBR9	SBR8
Write:								
Reset:	0	0	0	0	0	U	U	U

U = Undefined

Figure 7-5. SCI Baud Rate Control Register High (SCBDH)

Address: \$0071

	Bit 7	6	5	4	3	2	1	Bit 0
Read:	SBR7	SBR6	SBR5	SBR4	SBR3	SBR2	SBR1	SBR0
Write:								
Reset:	0	0	0	0	0	U	U	U

U = Undefined

Figure 7-6. SCI Baud Rate Control Register Low (SCBDL)

BTST — Baud Register Test Bit

BTST is for factory use only and is only accessible in special test mode.

BSPL — Baud Rate Counter Split Bit

BSOK is for factory use only and is only accessible in special test mode.

SBR[12:0] — SCI Baud Rate Select Bits

These bits represent the value BR in:

SCI baud rate control

$$\text{register value} = (\text{EXTAL}/32)/\text{target baud rate}$$

Table 7-2. SCI+ Baud Rates

EXTAL Frequencies										
EXTAL Freq.	8.0 MHz		12.0 MHz		16.0 MHz		20.0 MHz		24.0 MHz	
E Clock Freq.	2.0 MHz		3.0 MHz		4.0 MHz		5.0 MHz		6.0 MHz	
Target Baud Rate	SCI Baud Rate Control Register Values									
	Decimal	Hex	Decimal	Hex	Decimal	Hex	Decimal	Hex	Decimal	Hex
110 baud	2273	\$08E1	3409	\$0D51	4545	\$11C1	5682	\$1632	6818	\$1AA2
150 baud	1667	\$0683	2500	\$09C4	3333	\$0D05	4167	\$1047	5000	\$1388
300 baud	833	\$0341	1250	\$04E2	1667	\$0683	2083	\$0823	2500	\$09C4
600 baud	417	\$01A1	625	\$0271	833	\$0341	1042	\$0412	1250	\$04E2
1200 baud	208	\$00D0	313	\$0139	417	\$01A1	521	\$0209	625	\$0271
2400 baud	104	\$0068	156	\$009C	208	\$00D0	260	\$0104	313	\$0139
4800 baud	52	\$0034	78	\$004E	104	\$0068	130	\$0082	156	\$009C
9600 baud	26	\$001A	39	\$0027	52	\$0034	65	\$0041	78	\$004E
19200 baud	13	\$000D	20	\$0014	26	\$001A	33	\$0021	39	\$0027
38400 baud	—	—	—	—	13	\$000D	16	\$0010	20	\$0014
76800 baud	—	—	—	—	—	—	8	\$0008	10	\$000A

Serial Communications Interface (SCI)

7.9.2 Serial Communications Control Register 1

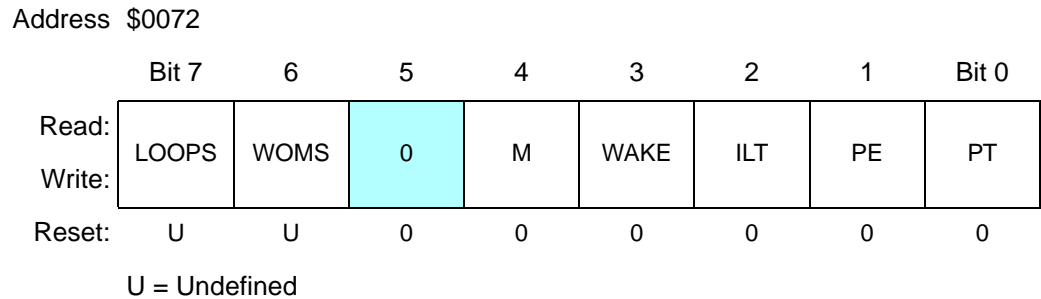


Figure 7-7. SCI Control Register 1 (SCCR1)

LOOPS — SCI Loop Mode Enable Bit

Both the transmitter and receiver must be enabled to use the loop mode. When the loop mode is enabled, the TxD pin is driven high (idle line state) if the transmitter is enabled.

0 = SCI transmit and receive operate normally.

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

WOMS — Wired-OR Mode for SCI Pins PD[1:0] Bits

See also 8.6.1 *Serial Peripheral Control Register* for a description of the DWOM (port D wired-OR mode) bit in the serial peripheral control register (SPCR).

0 = TxD and RxD operate normally.

1 = TxD and RxD are open drains if operating as outputs.

M — Mode (SCI Word Size) Bit

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

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

WAKE — Wakeup Mode Bit

0 = Wake up by idle line recognition

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

ILT — Idle Line Type Bit

0 = Short (SCI counts consecutive 1s after start bit.)

1 = Long (SCI counts one only after stop bit.)

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

PT — Parity Type Bit
0 = Parity even (even number of 1s causes parity bit to be 0, odd number of 1s causes parity bit to be 1)
1 = Parity odd (odd number of 1s causes parity bit to be 0, even number of 1s causes parity bit to be 1)

7.9.3 Serial Communications Control Register 2

Address \$0073

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

Figure 7-8. SCI Control Register 2 (SCCR2)

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

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

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

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

Serial Communications Interface (SCI)

TE — Transmitter Enable Bit

When TE goes from 0 to 1, one unit of idle character time (logic 1) is queued as a preamble.

0 = Transmitter disabled

1 = Transmitter enabled

RE — Receiver Enable Bit

0 = Receiver disabled

1 = Receiver enabled

RWU — Receiver Wakeup Control

0 = Normal SCI receiver operation

1 = Wakeup is enabled and receiver interrupts are inhibited.

SBK — Send Break Bit

At least one character time of break is queued and sent each time SBK is written to 1. Multiple breaks may be sent if the transmitter is idle at the time the SBK bit is toggled on and off, as the baud rate clock edge could occur between writing the 1 and writing the 0 to SBK.

0 = Break generator off

1 = Break codes generated as long as SBK = 1

7.9.4 Serial Communication Status Register 1

The SCSR provides flags for various SCI conditions which can be polled or used to generate SCI system interrupts. To clear any set flag, read SCSR while the flag is set and then write to SCDR.

Address \$0074

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

Figure 7-9. SCI Control Register 1 (SCCR1)

TDRE — Transmit Data Register Empty Flag

TDRE is set when the SCDR transfers its contents to the transmission shift register.

- 0 = SCDR is full.
- 1 = SCDR is empty.

TC — Transmit Complete Flag

TC is set when the final character in a message has been sent (no data, preamble, or break transmissions pending).

- 0 = Transmitter busy
- 1 = Transmitter idle

RDRF — Receive Data Register Full Flag

RDRF is set when the shift register has received a complete character and transferred it to the receive data register.

- 0 = RDR not full
- 1 = RDR full

IDLE — Idle Line Detected Flag

IDLE is set when a frame of all 1s is received after a message.

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

OR — Overrun Error Flag

OR is set if a new character is received before a previously received character is read from SCDR.

- 0 = No overrun
- 1 = Overrun detected

NF — Noise Error Flag

NF is set after the last bit in a frame is received if the samples in the receiver's data recovery circuit are not unanimous for any of the bits, including start and stop bits.

- 0 = No noise detected
- 1 = Noise detected

Serial Communications Interface (SCI)

FE — Framing Error Flag

FE is set when a 0 is detected where a stop bit (logic 1) was expected.

0 = Stop bit detected

1 = Logic 0 detected at the end of a character

PF — Parity Error Flag

PF is set if received data has incorrect parity. Clear PF by reading SCSR1.

0 = Parity disabled

1 = Parity enabled

7.9.5 Serial Communication Status Register 2

Address \$0075

	Bit 7	6	5	4	3	2	1	Bit 0
Read:	0	0	0	0	0	0	0	RAF
Write:								
Reset:	1	1	0	0	0	0	0	0


 = Unimplemented

Figure 7-10. SCI Control Register 2 (SCCR2)

RAF — Receiver Active Flag

RAF is a read-only bit.

0 = Receiver is inactive.

1 = A character is being received.

7.9.6 Serial Communications Data Register

The SCDR is a parallel register that performs two functions. Received data in the RDR is read from this address when the SCI is receiving, and data to be transmitted is written to this address when the SCI is transmitting.

Address \$0076

	Bit 7	6	5	4	3	2	1	Bit 0
Read:	R8	T8	0	0	0	0	0	0
Write:								
Reset:	Undefined after reset							

Address \$0077

	Bit 7	6	5	4	3	2	1	Bit 0
Read:	R7/T7	R6/T6	R5/T5	R4/T4	R3/T3	R2/T2	R1/T1	R0/T0
Write:								
Reset:	Undefined after reset							

Figure 7-11. SCI Data Register (SCDR)

R8 and T8 — Receiver Bit 8 and Transmitter Bit 8

Ninth data bit is received or transmitted when the system is configured for 8-bit data using mark address wakeup.

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

Serial Communications Interface (SCI)

Section 8. Serial Peripheral Interface (SPI)

8.1 Contents

8.2	Introduction	163
8.3	SPI Functional Description	164
8.4	SPI Signal Descriptions	166
8.4.1	Master In Slave Out (MISO)	166
8.4.2	Master Out Slave In (MOSI)	166
8.4.3	Serial Clock (SCK)	166
8.4.4	Slave Select (\overline{SS})	167
8.4.5	SPI Timing	167
8.5	SPI System Errors	168
8.5.1	Mode Fault Error	168
8.5.2	Write Collision Error	169
8.6	SPI Registers	169
8.6.1	Serial Peripheral Control Register	170
8.6.2	Serial Peripheral Status Register	172
8.6.3	Serial Peripheral Data Register	173
8.6.4	Port D Data Direction Register	174
8.6.5	System Configuration Options 2	175

8.2 Introduction

The serial peripheral interface (SPI) provides synchronous communication between the MCU and peripheral devices such as transistor-transistor logic (TTL) shift registers, liquid crystal display (LCD) drivers, analog-to-digital (A/D) converter subsystems, and other processors. Synchronous communication requires a clock and, in the TKHC11 series, a slave-select signal, but provides substantially faster communication than the asynchronous SCI, which does not require this extra hardware. The SPI system can send data at up to one half of

the E-clock rate when configured as master and the full E-clock rate when configured as a slave.

8.3 SPI Functional Description

The SPI is a 4-wire, full-duplex communication system. Characters are eight bits, transmitted most significant bit (MSB) first. One master device exchanges data with one or more slave devices. Each device selects its mode by writing either a 1 (master) or 0 (slave) to the MSTR bit in the serial peripheral control register (SPCR). As a master device transmits data to a slave device via the MOSI (master out slave in) line, the slave transmits data to the master via the MISO (master in slave out) line. The master produces a common synchronization clock signal and drives it on its SCK (serial clock) pin, which is configured as an output. The slave SCK pin is configured as an input to receive the clock. An external logic low signal is applied to the slave select pin (\overline{SS}) of each slave device for which a particular message is intended. Devices not selected (\overline{SS} high) ignore the transmission.

Received characters are double-buffered. Serial input bits are fed into a shift register; when the last bit is received, the completed character is parallel-loaded to a read data buffer. This allows the next message to be received while the current message is being read. As long as the buffer is read before the next received character is ready to be transferred to the buffer, no overrun condition occurs.

Transmitted characters are not double-buffered, they are written directly to the output shift register. This means that new data for transmission cannot be written to the shift register until the previous transmission is complete. An attempt to write during data transmission will not go through; the transmission in progress will proceed undisturbed, and the MCU will set the write collision (WCOL) status bit in the serial peripheral status register (SPSR). After the last bit of a character is shifted out, the SPI transfer complete flag (SPIF) of the SPSR is set. This will also generate an interrupt if the SPIE (SPI interrupt enable) bit in the SPCR is set.

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

Figure 8-1 shows the SPI block diagram.

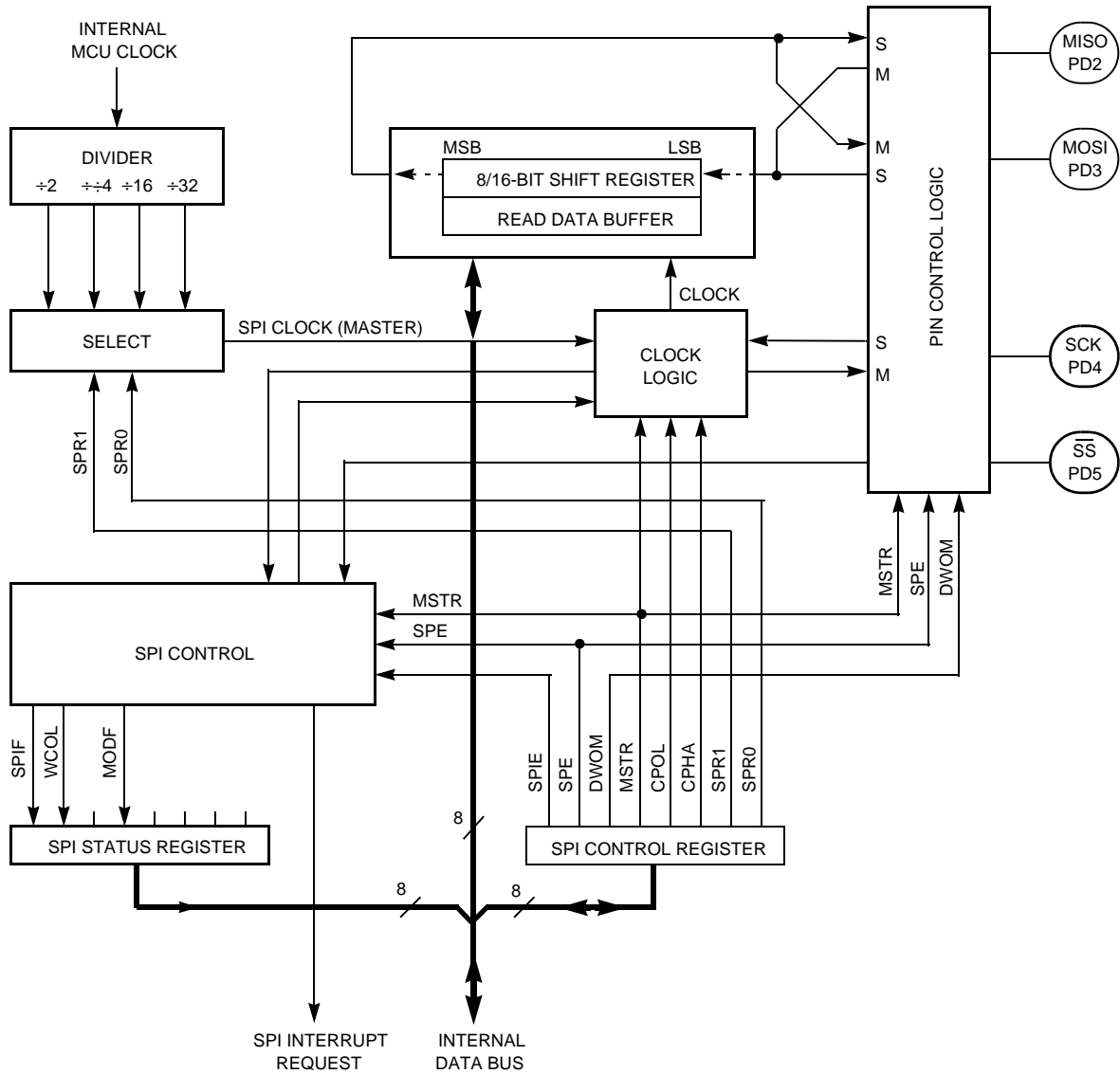


Figure 8-1. SPI Block Diagram

8.4 SPI Signal Descriptions

The four basic SPI signals (MISO, MOSI, SCK, and \overline{SS}) are discussed for both the master and slave modes in the following paragraphs.

Every SPI output line must have its corresponding port D data direction register (DDRD) bit set. If this bit is clear, the line is disconnected from the SPI logic and becomes a general-purpose input line. SPI input lines are not affected by the data direction register.

Serial Peripheral Interface (SPI)

8.4.1 Master In Slave Out (MISO)

The MISO is one of two unidirectional serial data lines in the SPI. It functions as an input in a master device and as an output in a slave device. The MISO line of a slave device is placed in the high-impedance state if the slave is not selected.

8.4.2 Master Out Slave In (MOSI)

This unidirectional serial data line is an output in a master device and an input in a slave device.

8.4.3 Serial Clock (SCK)

The serial clock (SCK) synchronizes data movement both in and out of all devices. Master and slave devices exchange a byte of information simultaneously during a sequence of eight clock cycles. SCK is generated by the master device so its SCK pin functions as an output. Slave devices receive this signal through their SCK pins, which are configured as inputs.

The SPI clock rate select bits in the master device determine the SCK clock rate. These bits are SPR[1:0] in the serial peripheral control register (SPCR) and SPR2 in the system configuration options 2 register (OPT2). These bits have no effect in a slave device.

8.4.4 Slave Select (\overline{SS})

The slave select (\overline{SS}) input is used to target specific devices in the SPI system. It must be pulled low on a targeted slave device prior to any communication with a master and must remain low for the duration of the transaction. \overline{SS} must always be high on any device in master mode. Pulling \overline{SS} low on a master mode device generates a mode fault error (see **8.5.1 Mode Fault Error**).

8.4.5 SPI Timing

Four possible timing relationships are available through control bits CPOL (clock polarity) and CPHA (clock phase) in the SPCR. These bits must be the same in both master and slave devices. The master device always places data on the MOSI line approximately a half-cycle before the SCK clock edge. This enables the slave device to latch the data. See **Figure 8-2**.

A write collision is normally a slave error because a slave has no control over when a master initiates a transfer. A master knows when a transfer is in progress, so there is no reason for a master to generate a write-collision error, although the SPI logic can detect write collisions in both master and slave devices.

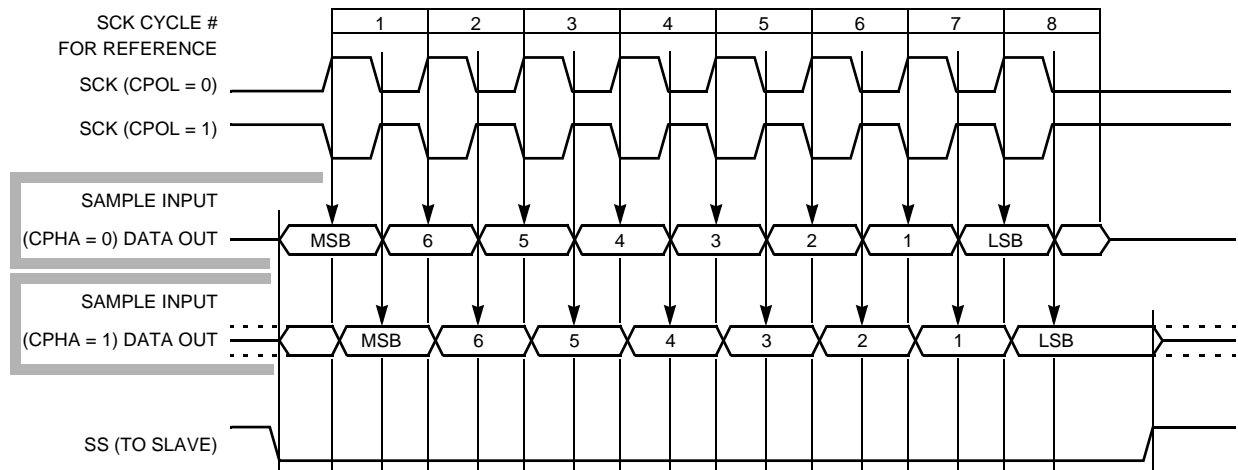


Figure 8-2. Data Clock Timing Diagram

CPOL selects an active high or low clock edge. CPHA selects one of two transfer formats. When CPHA is cleared, the shift clock is ORed with \overline{SS} . Each slave's \overline{SS} pin must be pulled high before it writes the next output byte to its data register. If a slave writes to its data register while \overline{SS} is low, a write collision error occurs. When CPHA is set, \overline{SS} may be left low for several SPI characters. When there is only one SPI slave MCU, its \overline{SS} line may be tied to V_{SS} if CPHA = 1 at all times.

The SPI configuration determines the characteristics of a transfer in progress. For a master, a transfer begins when data is written to SPDR and ends when SPIF is set. For a slave with CPHA cleared, a transfer starts when \overline{SS} goes low and ends when \overline{SS} returns high. In this case, SPIF is set at the middle of the eighth SCK cycle when data is transferred from the shifter to the parallel data register, but the transfer is still in progress until \overline{SS} goes high. For a slave with CPHA set, transfer begins when the SCK line goes to its active level, which is the edge at the beginning of the first SCK cycle. The transfer ends when SPIF is set. SCK in a slave must be inactive for at least two E-clock cycles between byte transfers.

8.5 SPI System Errors

Two types of errors can be detected by the SPI system:

- A mode fault error can occur when multiple devices attempt to act in master mode simultaneously.
- A write collision error results from an attempt to write data to the SPDR while a transmission is in progress.

8.5.1 Mode Fault Error

A mode fault error occurs when the \overline{SS} input line of an SPI system configured as a master goes to active low, usually because two devices have attempted to act as master at the same time. The resulting contention between push-pull CMOS pin drivers can cause them permanent damage. The mode fault disables the drivers in an attempt to protect them. The MSTR control bit in the SPCR and all four DDRD control bits associated with the SPI are cleared, effectively forcing the pins to be high-impedance inputs. The mode fault error flag (MODF) is set in the serial peripheral status register (SPSR). An interrupt is generated, subject to masking by the SPIE control bit and the I bit in the CCR. To disable the mode fault circuit, write a 1 to DDRD bit 5. This configures port D bit 5 as a general-purpose output rather than \overline{SS} .

Other precautions may be necessary to prevent driver damage. For instance, if two devices are made masters at the same time, mode fault does not help protect either one unless one of them selects the other as slave. The amount of damage possible depends on the length of time both devices attempt to act as master.

8.5.2 Write Collision Error

A write collision error occurs when the SPDR is written while a transmission is in progress. The SPDR is not double buffered in the transmit direction, so writes to the SPDR go directly into the SPI shift register, which would corrupt any transfer in progress. The MCU protects against this by preventing the write and generating the write collision error. The transmission continues undisturbed.

A write collision is normally a slave error because a slave has no control over when a master initiates a transfer. A master knows when a transfer is in progress, so there is no reason for a master to generate a write-collision error, although the SPI logic can detect write collisions in both master and slave devices.

8.6 SPI Registers

The three SPI registers provide control, status, and data storage functions respectively:

- Serial peripheral control register (SPCR)
- Serial peripheral status register (SPSR)
- Serial peripheral data register (SPDR)

NOTE: Throughout this manual, the registers are discussed by function. In the event that not all bits in a register are referenced, the bits that are not discussed are shaded.

8.6.1 Serial Peripheral Control Register

Address: \$0028

	Bit 7	6	5	4	3	2	1	Bit 0
Read:	SPIE	SPE	DWOM	MSTR	CPOL	CPHA	SPR1	SPR2
Write:								
Reset:	0	0	0	0	0	1	U	U

U = Undefined

Figure 8-3. Serial Peripheral Control Register (SPCR)

SPIE — Serial Peripheral Interrupt Enable Bit

0 = SPI interrupt disabled

1 = SPI interrupt is enabled each time the SPIF or MODF status flag in SPSR is set.

SPE — Serial Peripheral System Enable Bit

0 = SPI off

1 = SPI on — PD[5:2] function as SPI signals

DWOM — Port D Wired-OR Mode Bit

DWOM affects only the four SPI pins on port D, PD[5:2]. See also **7.9.2**

Serial Communications Control Register 1 for a discussion of the WOMS (wired-OR Mode for SCI pins) bit in the serial communications control register 1 (SCCR1).

0 = Normal CMOS outputs

1 = Open-drain outputs

Serial Peripheral Interface (SPI)

MSTR — Master Mode Select Bit

0 = Slave mode

1 = Master mode

CPOL — Clock Polarity Bit

When the clock polarity bit is cleared and data is not being transferred, the SCK pin of the master device has a steady state low value. When CPOL is set, SCK idles high.

CPHA — Clock Phase Bit

The clock phase bit, in conjunction with the CPOL bit, controls the clock-data relationship between master and slave. The CPHA bit selects one of two different clocking protocols.

SPR[1:0] — SPI Clock Rate Select Bits

On a master device, these two bits in conjunction with SPR2 in the OPT2 register select the baud rate to be used as SCK. See **Table 8-1**. These bits have no effect in slave mode.

Table 8-1. SPI+ Baud Rates

EXTAL Frequencies						
EXTAL Freq.	8.0 MHz	12.0 MHz	16.0 MHz	20.0 MHz	24.0 MHz	Other EXTAL
E Clock Freq.	2.0 MHz	3.0 MHz	4.0 MHz	5.0 MHz	6.0 MHz	EXTAL ÷ 4
Control Bits SPR[2:0]	SPI Baud Rate					E Clock Divide by
0 0 0	1.0 MHz	1.5 MHz	2.0 MHz	2.5 MHz	3.0 MHz	2
0 0 1	500 kHz	750 kHz	1.0 MHz	1.3 kHz	1.5 MHz	4
0 1 0	125 kHz	187.5 kHz	250 kHz	312.5 kHz	375.0 kHz	16
0 1 1	62.5 kHz	93.8 kHz	125 kHz	156.3 kHz	187.5 kHz	32
1 0 0	250 kHz	375 kHz	500 kHz	625 kHz	750.0 kHz	8
1 0 1	125 kHz	187.5 kHz	250 kHz	312.5 kHz	375.0 kHz	16
1 1 0	31.3 kHz	46.9 kHz	62.5 kHz	78.1 kHz	93.8 kHz	64
1 1 1	15.6 kHz	23.4 kHz	31.3 kHz	39.1 kHz	46.9 kHz	128

Serial Peripheral Interface (SPI)

8.6.2 Serial Peripheral Status Register

Address: \$0029

	Bit 7	6	5	4	3	2	1	Bit 0
Read:	SPIF	WCOL	0	MODF	0	0	0	0
Write:								
Reset:	0	0	0	0	0	0	0	0

Figure 8-4. Serial Peripheral Status Register (SPSR)

SPIF — SPI Transfer Complete Flag

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 Bit

Clearing the WCOL bit is accomplished by reading the SPSR (with WCOL set) followed by an access of SPDR.

- 0 = No write collision
- 1 = Write collision

MODF — Mode Fault Bit

To clear the MODF bit, read the SPSR (with MODF set), then write to the SPCR.

- 0 = No mode fault
- 1 = Mode fault

8.6.3 Serial Peripheral Data Register

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.

Address: \$002A

	Bit 7	6	5	4	3	2	1	Bit 0
Read:	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Write:	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Reset:	0	0	0	0	0	0	0	0

Figure 8-5. Serial Peripheral Data Register (SPDR)

A write to SPDR goes directly to the transmission shift register.

A read of the SPDR retrieves data from the read data buffer.

Serial Peripheral Interface (SPI)

8.6.4 Port D Data Direction Register

Address: \$0009

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

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

DDD5 Bit

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

0 = PD5 is an error-detect input to the SPI.

1 = PD5 is configured as a general-purpose output line.

DDD[4:2] Bits

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

8.6.5 System Configuration Options 2

Address: \$0038

	Bit 7	6	5	4	3	2	1	Bit 0
Read:	LIRDV	CWOM	STRCH ⁽¹⁾	IRVNE	LSBF	SPR2	EXT4X	DISE
Write:								
Reset:	0	0	0	—	0	0	0	0

1. Not available on TKHC11KS devices

Figure 8-7. System Configuration Options 2 Register (OPT2)

LSBF — Least Significant Bit First Enable Bit

Setting LSBF causes data to be transmitted LSB first (the default is MSB first). LSBF does not affect bit positions in the data register; reads and writes always have MSB in bit 7.

SPR2 — SPI Clock Rate Select Bit

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

Serial Peripheral Interface (SPI)

Section 9. Timing System

9.1 Contents

9.2	Introduction	178
9.3	Timer Structure	179
9.4	Input Capture and Output Compare Overview	181
9.4.1	Timer Counter Register	184
9.4.2	Timer Interrupt Flag 2 Register	185
9.4.3	Timer Interrupt Mask 2 Register	185
9.4.4	Port A Data Direction Register	186
9.4.5	Pulse Accumulator Control Register	187
9.5	Input Capture (IC)	187
9.5.1	Timer Input Capture Registers	188
9.5.2	Timer Input Capture 4/Output Compare 5 Register	189
9.5.3	Timer Interrupt Flag 1 Register	190
9.5.4	Timer Interrupt Mask 1 Register	190
9.5.5	Timer Control 2 Register	191
9.6	Output Compare (OC)	192
9.6.1	Timer Output Compare Registers	193
9.6.2	Timer Input Capture 4/Output Compare 5 Register	195
9.6.3	Timer Interrupt Flag 1 Register	195
9.6.4	Timer Interrupt Mask 1 Register	196
9.6.5	Timer Control 1 Register	196
9.6.6	Timer Compare Force Register	197
9.6.7	Output Compare 1 Mask Register	198
9.6.8	Output Compare 1 Data Register	198
9.7	Pulse Accumulator (PA)	199
9.7.1	Port A Data Direction Register	201
9.7.2	Pulse Accumulator Control Register	201
9.7.3	Timer Interrupt Flag 2	202

9.7.4	Timer Interrupt Mask 2 Register	203
9.7.5	Pulse Accumulator Count Register	204
9.8	Real-Time Interrupt (RTI)	204
9.8.1	Timer Interrupt Flag 2 Register	205
9.8.2	Timer Interrupt Mask 2 Register	205
9.8.3	Pulse Accumulator Control Register (PACTL)	206
9.9	XOUT Pin Control	207
9.9.1	System Configuration Register	207
9.9.2	System Configuration Options 2 Register	208
9.10	Pulse-Width Modulator (PWM)	209
9.10.1	PWM System Description	209
9.10.2	Pulse-Width Modulation Control Registers	211
9.10.2.1	Pulse-Width Modulation Timer Clock Select Register	211
9.10.2.2	Pulse-Width Modulation Timer Polarity Register	213
9.10.2.3	Pulse-Width Modulation Timer Prescaler Register	213
9.10.2.4	Pulse-Width Modulation Timer Enable Register	214
9.10.2.5	Pulse-Width Modulation Timer Counters 1 to 4 Registers	215
9.10.2.6	Pulse-Width Modulation Timer Periods 1 to 4 Registers	216
9.10.2.7	Pulse-Width Modulation Timer Duty Cycle1 to 4 Registers	217

9.2 Introduction

TK68HC11 microcontrollers contain an extensive timing system to support a wide variety of timer-related functions. This section discusses the nature of the timing system and presents details of timer-related functions including:

- Input capture/output compare (IC/OC)
- Real-time interrupt (RTI)
- Pulse accumulator (PA)
- Pulse width modulation (PWM)

9.3 Timer Structure

As **Figure 9-1** shows, the primary system clocks, including the E clock and the internal PH2 bus clock, are derived from the oscillator output divided by four.

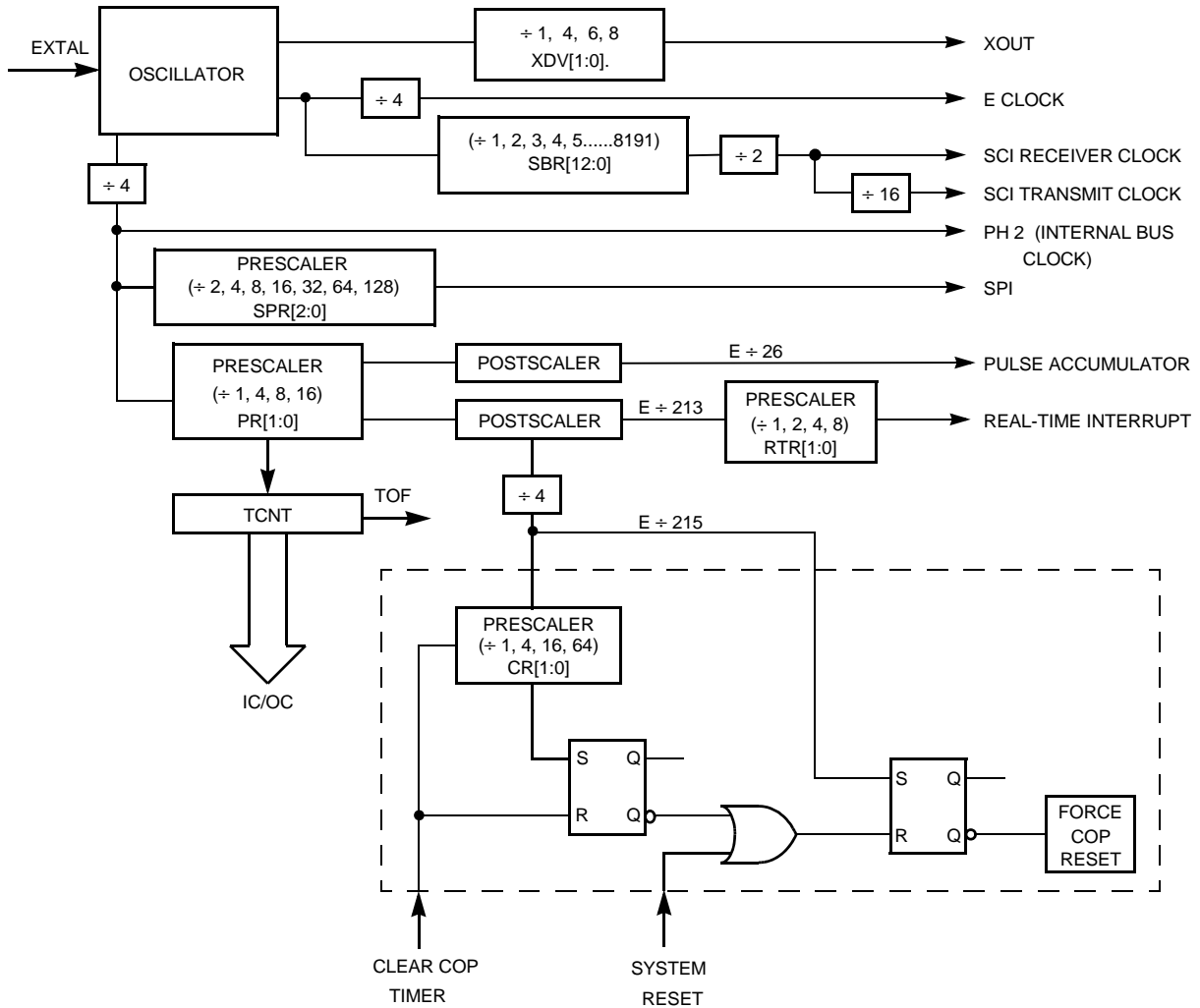


Figure 9-1. Timer Clock Divider Chains

The PH2 bus clock feeds four primary divider chains. The functions supplied by each of these chains are:

1. Serial peripheral interface (SPI)
2. Input capture/output compare (IC/OC)
3. Pulse accumulator (PA)
4. RTI and COP watchdog circuit

The SPI prescale factor is determined by bits SPR[2] in the system configuration options 2 (OPT2) register and SPR[1:0] in the serial peripheral control register (SPCR). See **8.6.1 Serial Peripheral Control Register** and **8.6.5 System Configuration Options 2**.

The input capture and output compare functions are based on a 16-bit free-running counter, which is driven by the PH2 clock divided by a programmable prescaler. Bits PR[1:0] of the timer interrupt mask 2 (TMSK2) register enable the user to select one of four divisors: 1, 4, 8, or 16. The output of this prescaler, referred to as the main timer, feeds the divider chains for the pulse accumulator, RTI, and COP circuits as well as the free-running counter. **Table 9-1** shows main timer frequencies and periods available from the most common crystal inputs.

Table 9-1. Main Timer Rates

EXTAL Frequencies						
EXTAL Freq.	8.0 MHz	12.0 MHz	16.0 MHz	20.0 MHz	20.0 MHz	Other EXTAL
E Clock Freq.	2.0 MHz	3.0 MHz	4.0 MHz	5.0 MHz	5.0 MHz	EXTAL/4
E Clock Period	500 ns	333 ns	250 ns	200 ns	200 ns	1/E
Control Bits PR[1:0]	Main Timer Period (1 Count/Timer Overflow)					1 Count Timer Overflow
0 0	500 ns 32.768 ms	333 ns 21.845 ms	250 ns 16.384 ms	200 ns 13.107 ms	167 ns 10.923 ms	$1 \div E$ $2^{16} \div E$
0 1	2.0 μ s 131.07 ms	1.3 μ s 87.381 ms	1.0 μ s 85.536 ms	800 ns 52.429 ms	667 ns 43.961 ms	$4 \div E$ $2^{18} \div E$
1 0	4.0 μ s 262.14 ms	2.667 μ s 174.76 ms	2.0 μ s 131.07 ms	1.6 μ s 104.86 ms	1.333 μ s 87.381 ms	$8 \div E$ $2^{19} \div E$
1 1	8.0 μ s 524.29 ms	5.333 μ s 349.53 ms	4.0 μ s 262.14 ms	3.2 μ s 209.72 ms	2.667 μ s 174.76 ms	$16 \div E$ $2^{20} \div E$

The free-running 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 the timer overflow flag (TOF) in the

timer interrupt flag 2 (TFLG2) register, and continues to increment. The value in this counter can be read in the timer counter (TCNT) register, but cannot be written or changed except by reset.

The pulse accumulator, described in **9.7 Pulse Accumulator (PA)**, derives its clock by post-scaling the main timer so that the output frequency is always E clock divided by 64. This clock drives an 8-bit counter while the pulse accumulator is operating in event counting mode.

RTI is a programmable periodic interrupt circuit that can be used to pace the execution of software routines, as described in **9.8 Real-Time Interrupt (RTI)**. The clock driving this function is also derived from the clock driving the free-running counter. The post-scaler output of this chain runs at a frequency of E clock divided by 2^{13} .

The COP watchdog timer (**5.3.3 Computer Operating Properly (COP) System**) further divides the RTI clock by four to drive its circuitry at a frequency of E clock divided by 2^{15} .

9.4 Input Capture and Output Compare Overview

The TK68HC11K series features:

- Three input capture channels
- Four output compare channels
- 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.

Figure 9-2 shows the capture/compare system block diagram. The port A pin control block includes logic for timer functions and for general-purpose I/O. This block contains the edge-detection logic for pins PA[2:0] as well as the control logic that enables edge selection for the input capture trigger.

- PA[2:0] can serve either as input capture pins IC[1:3] or as general-purpose input/output (GPIO).
- PA[6:4] can serve either as drivers for output compare functions OC[2:4] or GPIO.
- PA3 can be used for GPIO, input capture 4, output compare 5, or output compare 1.

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

Reading the port A register returns the actual pin level on any pin functioning as an input, and the logic level of the internal pin driver (NOT the pin level) on any pin functioning as an output. This is true whether the pins are configured for timer functions or GPIO. Writing to port A pins configured for timer functions has no visible effect; the writes are latched but do not drive the pins.

Registers common to both the input capture and output compare functions include:

- Timer counter register (TCNT)
- Timer interrupt flag 2 (TFLG2)
- Timer interrupt mask 2 (TMSK2)
- Data direction register A (DDRA)
- Pulse accumulator control register (PACTL)

NOTE: *Throughout this manual, the registers are discussed by function. In the event that not all bits in a register are referenced, the bits that are not discussed are shaded.*

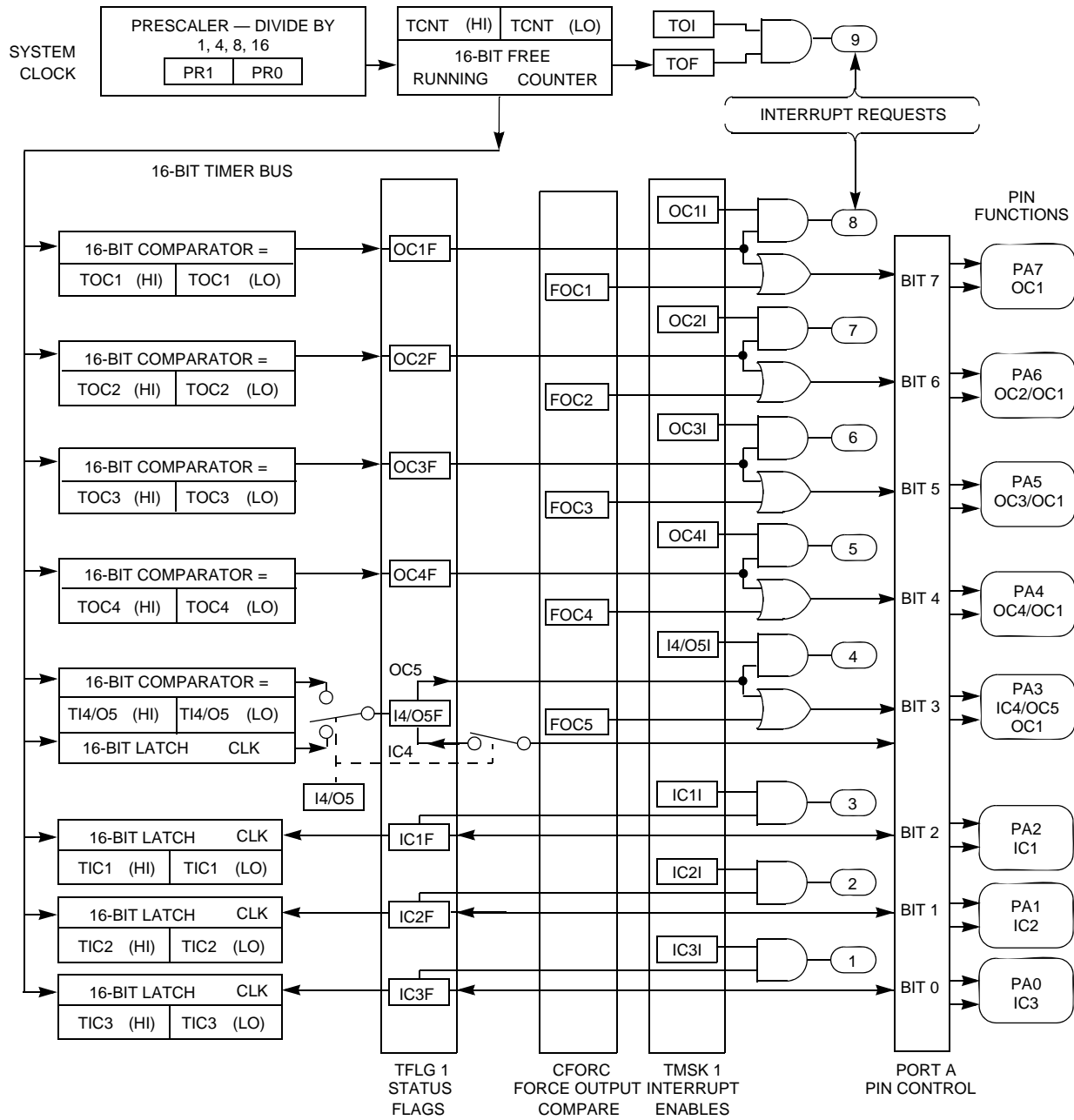


Figure 9-2. Capture/Compare Block Diagram

9.4.1 Timer Counter Register

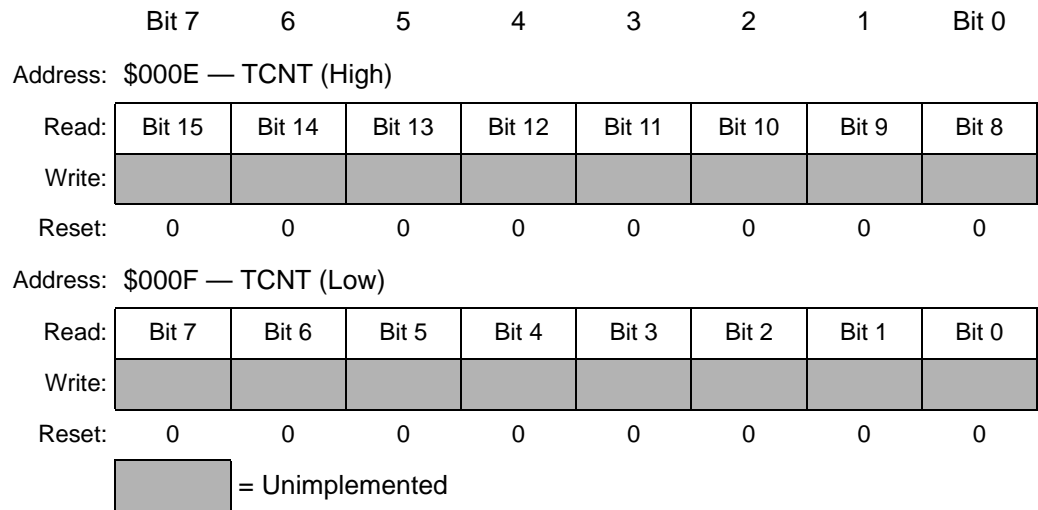


Figure 9-3. Timer Counter Register (TCNT)

TCNT reflects the current value in the free-running counter. Input capture functions use this number to mark the time of an external event, and output compare functions use it to determine the time at which to generate an event.

In normal modes, TCNT is a read-only register. Writes to TCNT in normal modes have no effect. TCNT can be read and written in special modes.

9.4.2 Timer Interrupt Flag 2 Register

Address: \$0025

	Bit 7	6	5	4	3	2	1	Bit 0
Read:	TOF	RTIF	PAOVF	PAIF	0	0	0	0
Write:								
Reset:	0	0	0	0	0	0	0	0

Figure 9-4. Timer Interrupt Flag 2 (TFLG2)

Clear each flag by writing a 1 to the corresponding bit position.

TOF — Timer Overflow Flag

Set when TCNT changes from \$FFFF to \$0000.

9.4.3 Timer Interrupt Mask 2 Register

Address: \$0024

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

Figure 9-5. Timer Interrupt Mask 2 (TMSK2)

Bits in TMSK2 correspond bit for bit with flag bits in TFLG2.

TOI — Timer Overflow Interrupt Enable Bit

0 = Timer overflow interrupt disabled

0 = An interrupt request is generated when TOF is set.

PR[1:0] — Timer Prescaler Select Bits

These bits determine the main timer prescale divisor, as shown in **Table 9-2**. The system bus (E clock) frequency is divided by this number to produce the clock which drives the free-running counter. Refer to **Table 9-1** for specific timing values.

In normal modes, PR[1:0] can be written only once, and the write must be within 64 cycles after reset.

Table 9-2. Timer Prescale

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

9.4.4 Port A Data Direction Register

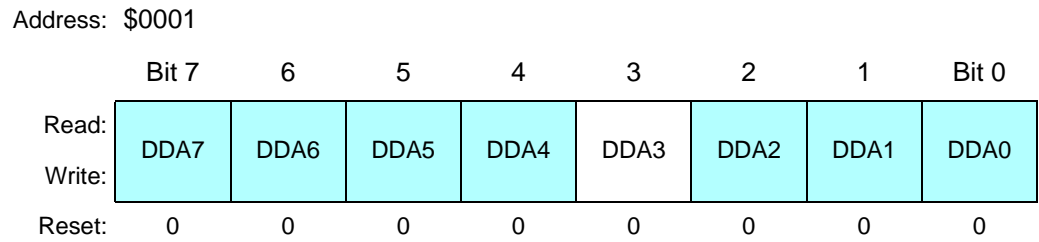


Figure 9-6. Port A Data Direction Register (DDRA)

DDA3 — Data Direction Control for Port A, Bit 3
0 = PA3 configured as an input
1 = PA3 configured as an output

9.4.5 Pulse Accumulator Control Register

Address: \$0026

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

Figure 9-7. Pulse Accumulator Control Register (PACTL)

I4/O5 — Input Capture 4/Output Compare 5 Bit
 0 = Configure PA3 as OC5
 1 = Configure PA3 as IC4

To configure PA3 as input compare 4, clear DDA3 and set I4/O5. To configure PA3 as output compare 5, set DDA3 and clear I4/O5. If the DDA3 bit is set (configuring PA3 as an output) and IC4 is enabled, writing a one to TI4/O5 causes an input capture. Writing to TI4/O5 has no effect when DDA3 is cleared and/or OC5 is enabled.

9.5 Input Capture (IC)

The input capture function records the time an external event occurs by latching the value of the free-running counter into one of the timer input capture (TIC) registers when a selected edge is detected at its associated timer input pin. Software can store latched values and use them to compute the period 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 the period, two successive edges of the same polarity are captured. To measure pulse width, two alternate polarity edges are captured.

Capture requests are latched on the opposite half cycle of PH2 from when the timer counter is being incremented. This synchronization process introduces a delay between edge occurrence and counter value detection. Because these delays offset each other when the time between two edges is measured, they can be ignored. There is a similar delay for output compare between the actual compare point and when the output pin changes state.

9.5.1 Timer Input Capture Registers

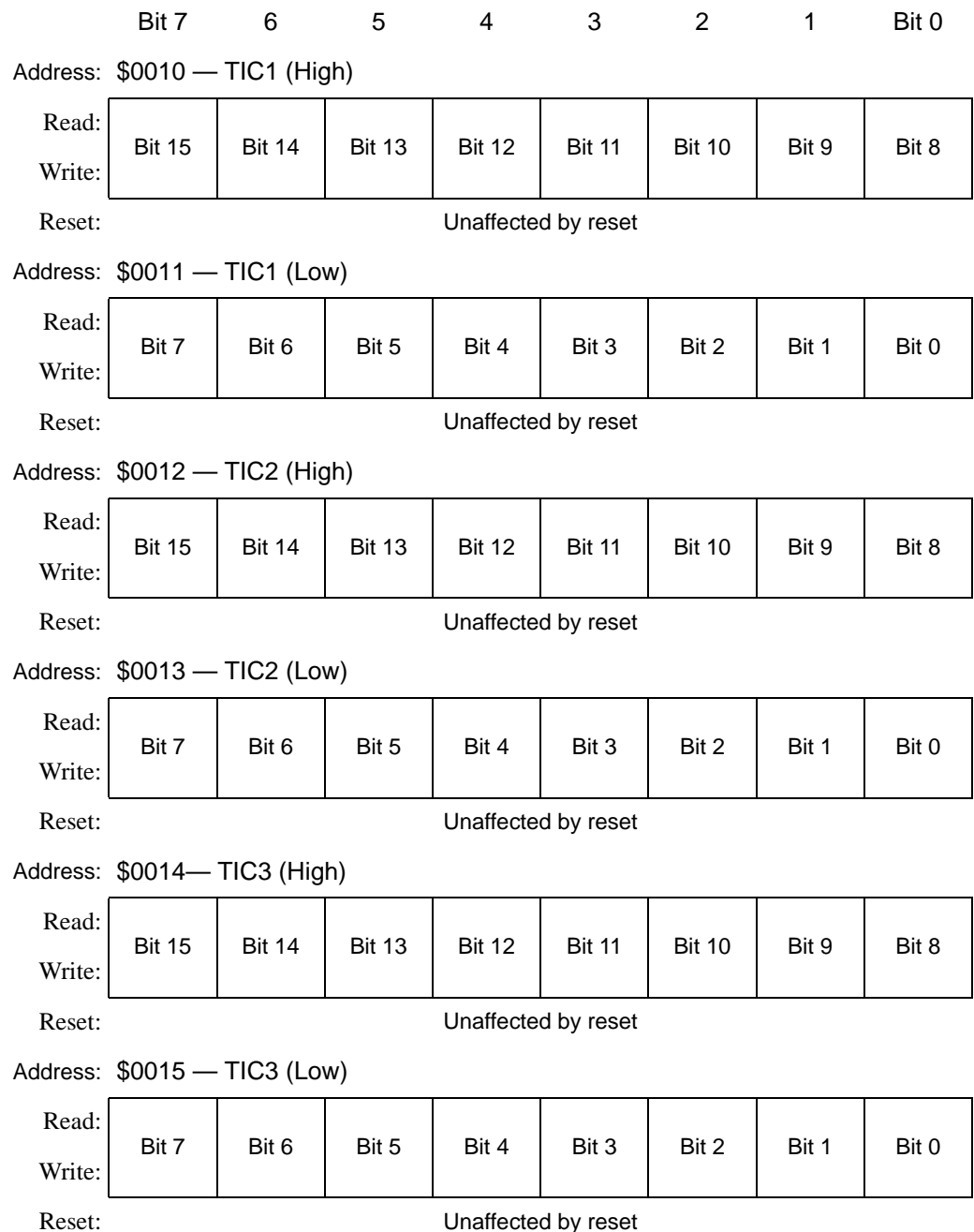


Figure 9-8. Timer Input Capture Registers (TIC1–TIC3)

9.5.2 Timer Input Capture 4/Output Compare 5 Register

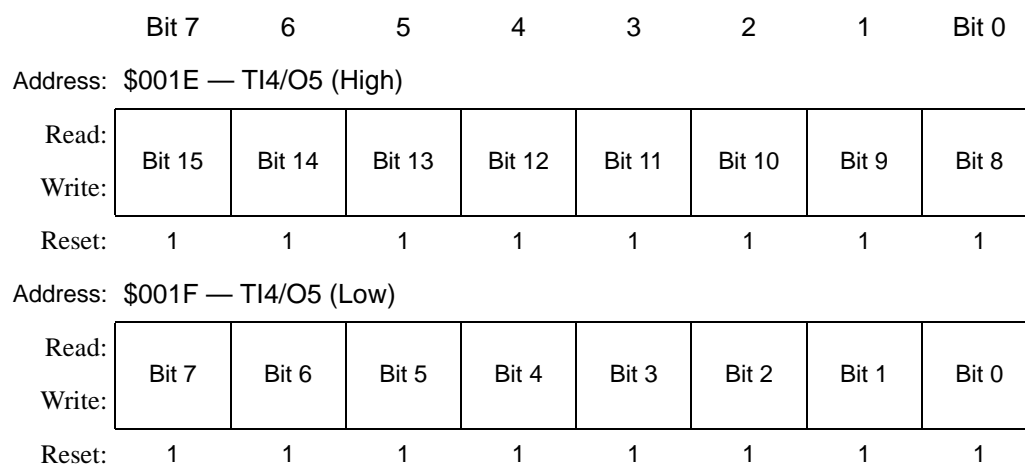


Figure 9-9. Timer Input Capture 4/Output Compare 5 Register (TI4/O5)

TI4/O5 functions as the input capture register for IC4 when PA3 is configured for input capture 4.

When an edge on an input capture pin has been detected and synchronized, the 16-bit free-running counter value is latched in the associated input capture register pair in a single 16-bit parallel transfer. The latch occurs on the opposite half-cycle of the phase two clock from when the timer counter is incremented. This ensures a stable count value whenever a capture occurs.

Input capture values can be retrieved from a TIC register with two successive 8-bit reads. Reading the high-order byte inhibits a new capture transfer for one bus cycle to ensure that the successive low-order byte read corresponds with it. If a new input capture occurs immediately after a high-order byte read, transfer is delayed for an additional cycle, but the new value is not lost. To assure coherency between the two bytes, use a double-byte read instruction such as LDD.

9.5.3 Timer Interrupt Flag 1 Register

Address: \$0023

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

Figure 9-10. Timer Interrupt Flag 1 Register (TFLG1)

Clear each flag by writing a 1 to the corresponding bit position.

ICxF — Input Capture x Flag

Set each time a selected active edge is detected on the corresponding input capture line.

I4/O5F — Input Capture 4/Output Compare 5 Flag

Set each time a selected active edge is detected on the IC4 line if IC4 is enabled.

9.5.4 Timer Interrupt Mask 1 Register

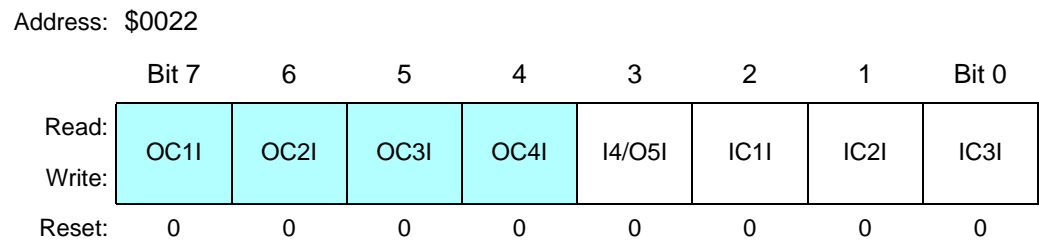


Figure 9-11. Timer Interrupt Mask 1 Register (TMSK1)

Bits in TMSK1 correspond bit for bit with flag bits in TFLG1.

ICxI — Input Capture Interrupt Enable Bit

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

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

If I4/O5I is set when IC4 is enabled and the I4/O5F flag bit is set, a hardware interrupt sequence is requested.

9.5.5 Timer Control 2 Register

Address: \$0021

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

Figure 9-12. Timer Control 2 Register (TCTL2)

EDGx[B:A] — Input Capture Edge Control Bits

These bit pairs determine the edge polarities on the input capture pins that trigger the corresponding input capture functions. 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.

Each EDGx bit pair is cleared (IC function disabled) by reset and must be encoded according to the values in **Table 9-3** to configure the corresponding input capture edge detector circuit. IC4 functions only if the I4/O5 bit in the PACTL register is set.

Table 9-3. Input Capture Edge Selection

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

9.6 Output Compare (OC)

The output compare (OC) function generates a programmed action when the 16-bit counter reaches a specified value. Each of the five output compare functions contains a separate 16-bit timer output compare (TOC) register and a dedicated 16-bit comparator. Each TOC register is set to \$FFFF on reset. When an OC channel is enabled, the value in its TOC register is compared to the free-running counter value during each E-clock cycle. When the values

match, the channel's output compare flag is set in timer interrupt flag 1 (TFLG1). If the channel's interrupt is enabled in the timer interrupt mask register 1 (TMSK1), an interrupt is generated. Also, the corresponding timer output pin is toggled or driven to a specified logic level. This pin activity occurs on each successful compare, whether or not the OCxF flag in the TFLG1 register was previously cleared.

The pin action for each of the OC channels [5:2] is controlled by a pair of bits (OMx and OLx) in the TCTL1 register and affects only the channel's associated pin. A successful OC1 compare can affect any or all five of the OC pins. The action taken when a match is found for OC1 is controlled by two 8-bit registers:

- Output compare 1 mask register (OC1M)
- Output compare 1 data register (OC1D)

OC1M specifies which port A outputs are to be used, and OC1D specifies the data placed on these port pins.

Although the TK68HC11K series devices have four built-in pulse-width modulation (PWM) channels (**9.10 Pulse-Width Modulator (PWM)**), the output compare function can be used to produce an additional pulse-width modulated waveform. 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.
- Use OC1D to select either a high or low output, depending on the polarity of the pulse being produced.
- After a match occurs, change the appropriate OC1D bit to the opposite polarity, then add a value representing the width of the pulse to the original value and write it 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 to the resolution of the free-running counter, independent of software latencies. To generate an output signal of a specific frequency and duty cycle, repeat this pulse-generating procedure.

9.6.1 Timer Output Compare Registers

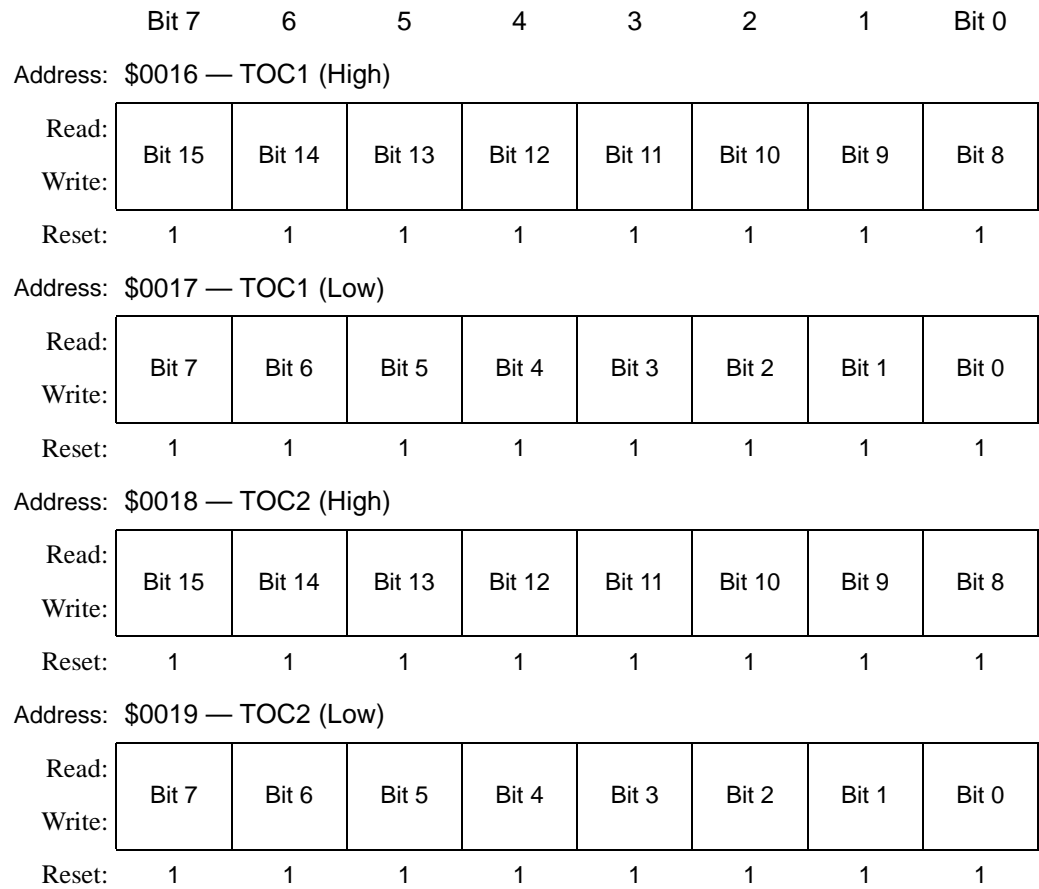


Figure 9-13. Timer Output Compare Registers (TOC1–TOC4)

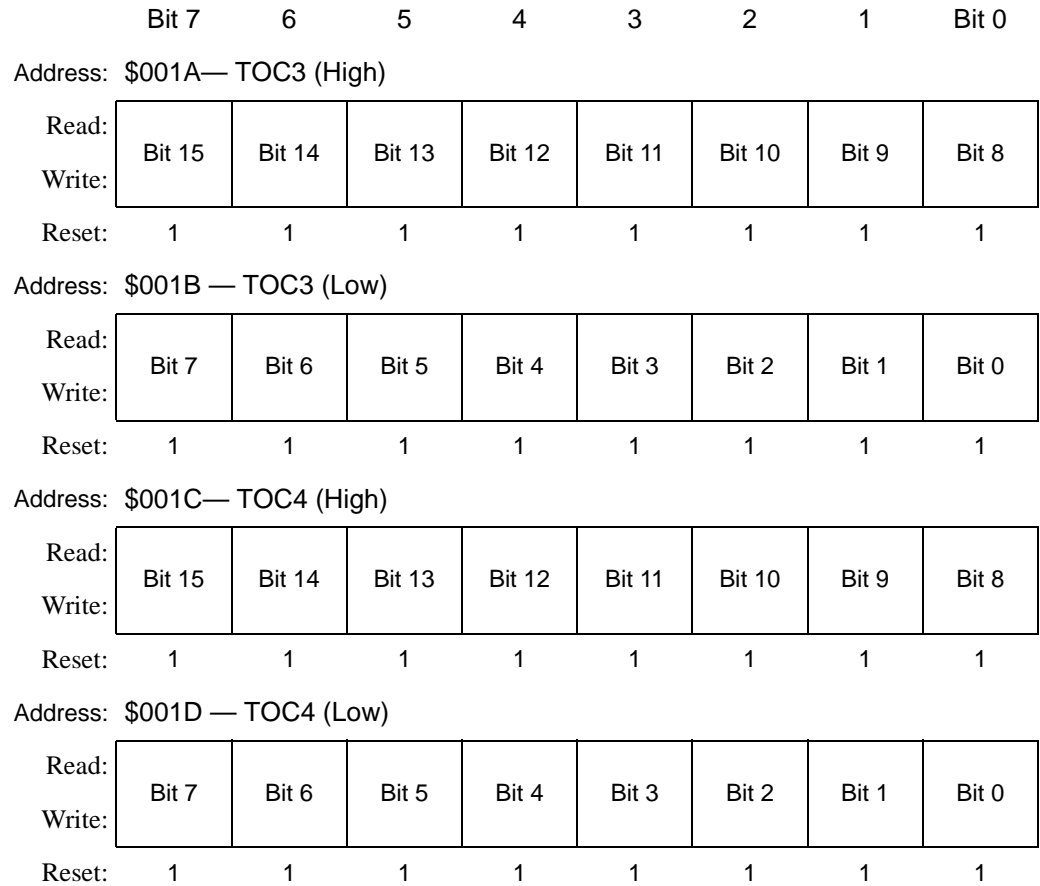


Figure 9-13. Timer Output Compare Registers (TOC1–TOC4) (Continued)

All output compare registers are 16-bit read-write. Any of these registers can be used as a storage location if it is not used for output compare or input capture.

9.6.2 Timer Input Capture 4/Output Compare 5 Register

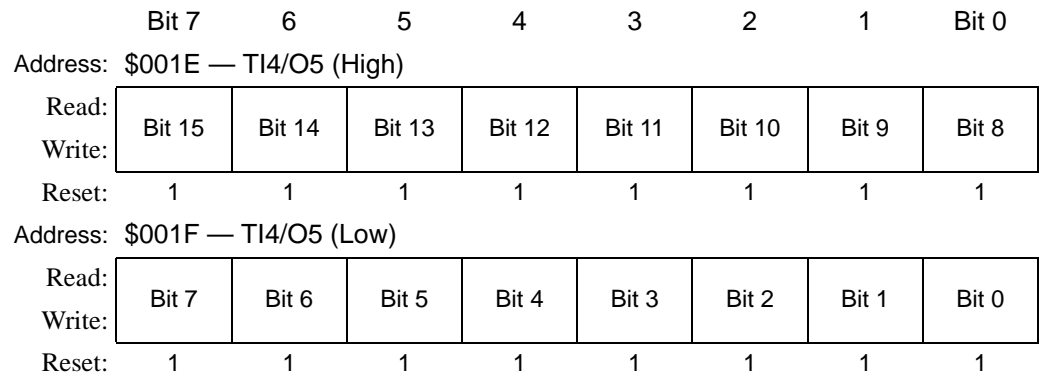


Figure 9-14. Timer Input Capture 4/Output Compare 5 Register (TI4/O5)

Functions as the output compare register for OC5 when PA3 is configured for output compare 5. This register is 16-bit read-write. It can be used as a storage location if it is not used for output compare or input capture.

9.6.3 Timer Interrupt Flag 1 Register

Address: \$0023

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

Figure 9-15. Timer Interrupt Flag 1 Register (TFLG1)

Clear each flag by writing a 1 to the corresponding bit position.

OCxF — Output Compare x Flag

Set each time the counter matches output compare x value.

I4/O5F — Input Capture 4/Output Compare 5 Flag

Set each time the counter matches output compare 5 value if OC5 is enabled.

9.6.4 Timer Interrupt Mask 1 Register

Address: \$0022

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

Figure 9-16. Timer Interrupt Mask 1 Register (TMSK1)

Bits in TMSK1 correspond bit for bit with flag bits in TFLG1.

OC1I–OC4I — Output Compare x Interrupt Enable Bits

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 or Output Compare 5 Interrupt Enable Bit

If I4/O5I is set when OC5 is enabled and the I4/O5F flag bit is set, a hardware interrupt sequence is requested.

9.6.5 Timer Control 1 Register

Address: \$0020

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

Figure 9-17. Timer Control Register 1 (TCTL1)

OM[2:5] and OL[2:5] — Output Mode and Output Level Bits

Use these bit pairs as indicated in **Table 9-4** to specify the action taken after a successful OCx compare.

Table 9-4. Timer Output Compare Actions

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

9.6.6 Timer Compare Force Register

Address: \$000B

	Bit 7	6	5	4	3	2	1	Bit 0
Read:	FOC1	FOC2	FOC3	FOC4	FOC5	0	0	0
Write:						0	0	0
Reset:	0	0	0	0	0	0	0	0

Figure 9-18. Timer Compare Force Register (CFORC)

The CFORC register allows forced early compares. Writing a 1 to any bit of FOC[1:5] forces the programmed pin actions for the corresponding OC channel to occur at the next timer count transition after the write to CFORC. The action taken as a result of a forced compare is identical to the action taken when a match between the OCx register and the free-running counter occurs, except that the corresponding interrupt and status flag bits are not set.

CFORC should not be applied to an output compare function that is programmed to toggle its output on a successful compare because a normal compare that occurs immediately before or after the force can result in an undesirable operation.

FOC[1:5] — Force Output Comparison Bits

0 = No action taken

1 = Output x action occurs at the next timer count transition

9.6.7 Output Compare 1 Mask Register

Address: \$000C

	Bit 7	6	5	4	3	2	1	Bit 0
Read:	OC1M7	OC1M6	OC1M5	OC1M4	OC1M3	0	0	0
Write:								
Reset:	0	0	0	0	0	0	0	0

Figure 9-19. Output Compare 1 Mask Register (OC1M)

OC1M specifies which bits of port A will respond to a successful compare for output capture 1. The bits of the OC1M[7:3] register correspond to PA[7:3].

OC1M[7:3] — Output Compare 1 Masks

0 = OC1 is disabled at the corresponding PA pin.

1 = OC1 is enabled at the corresponding PA pin.

9.6.8 Output Compare 1 Data Register

Address: \$000D

	Bit 7	6	5	4	3	2	1	Bit 0
Read:	OC1D7	OC1D6	OC1D5	OC1D4	OC1D3	0	0	0
Write:								
Reset:	0	0	0	0	0	0	0	0

Figure 9-20. Output Compare 1 Data Register (OC1D)

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

OC1D[7:3] — Output Compare Data Bits

0 = Corresponding port A pin is cleared on successful OC1 compare if the corresponding OC1M bit is set.

1 = Corresponding port A pin is set on successful OC1 compare if the corresponding OC1M bit is set.

9.7 Pulse Accumulator (PA)

The pulse accumulator can be used either to count events or measure the duration of a particular event. In event counting mode, the pulse accumulator's 8-bit counter increments each time a specified edge is detected on the pulse accumulator input pin, PA7. The maximum clocking rate for this mode is E clock divided by two. In gated time accumulation mode, an internal clock increments the 8-bit counter at a rate of E clock divided by 64 while the input at PA7 remains at a predetermined logic level. **Table 9-5** shows pulse accumulator clock periods for three common crystal frequencies.

Figure 9-21 is a block diagram of the pulse accumulator.

Table 9-5. Pulse Accumulator Timing

EXTAL Frequencies						
EXTAL Freq.	8.0 MHz	12.0 MHz	16.0 MHz	20.0 MHz	24.0 MHz	Other EXTAL
E Clock Freq.	2.0 MHz	3.0 MHz	4.0 MHz	5.0 MHz	6.0 MHz	EXTAL ÷
Maximum event counting frequency	1.0 MHz	1.5 MHz	2.0 MHz	2.5 MHz	3.0 MHz	E ÷ 2
Gated mode count resolution	32.0 μs	21.3 μs	16.0 μs	12.8 μs	10.7 μs	2 ⁶ ÷ E
Gated mode count overflow	8.192 μs	5.461 μs	4.096 μs	3.277 μs	2.731 μs	2 ¹⁴ ÷ E

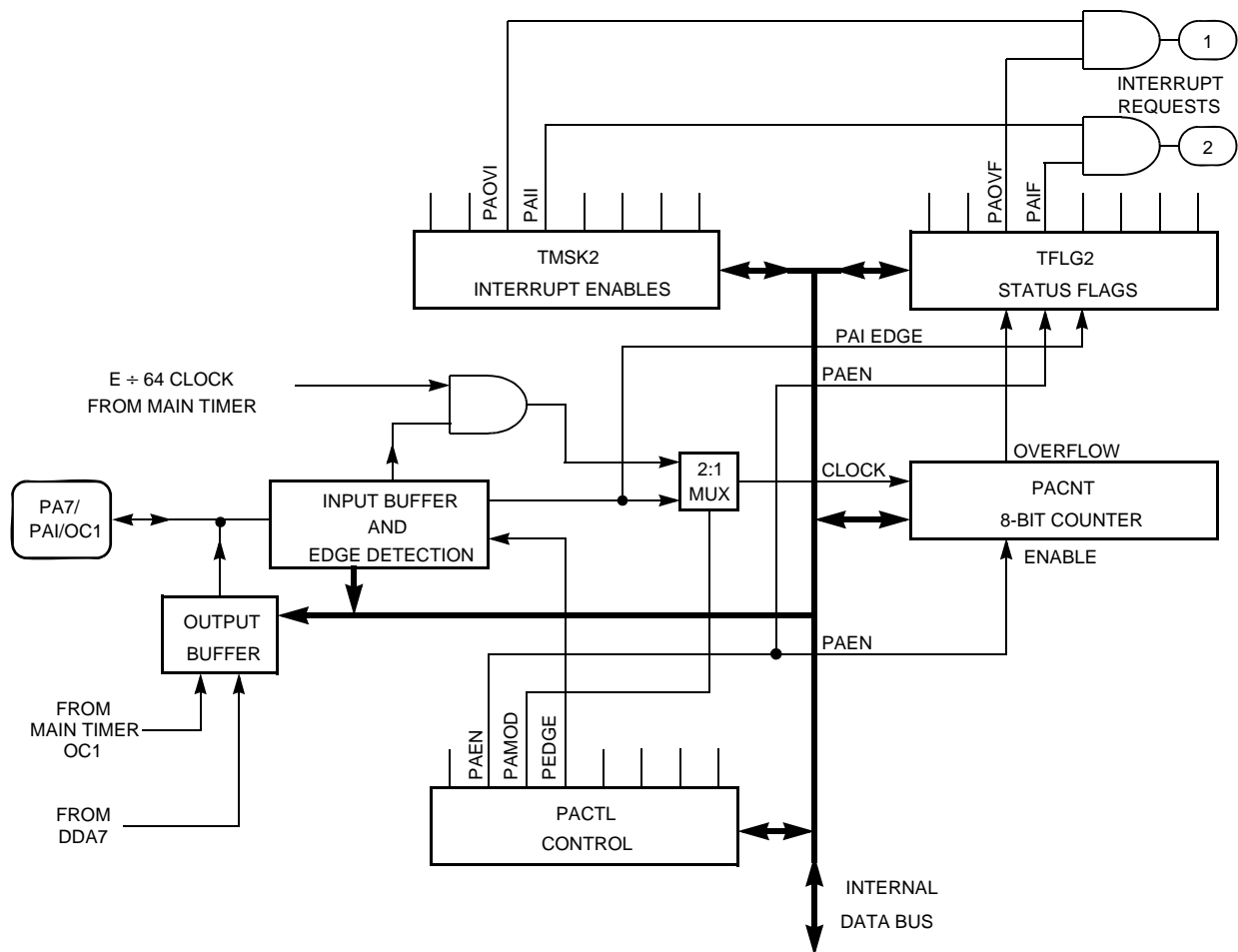


Figure 9-21. Pulse Accumulator

Registers involved in pulse accumulator operation include:

- Data direction register A (DDRA)
- Pulse accumulator control register (PACTL)
- Timer interrupt mask 2 register (TMSK2)
- Timer interrupt flag 2 (TFLG2)
- Pulse accumulator count register (PACNT)

9.7.1 Port A Data Direction Register

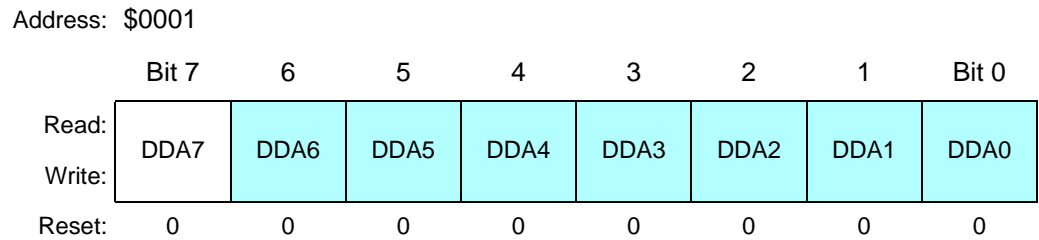


Figure 9-22. Port A Data Direction Register (DDRA)

The pulse accumulator uses port A, bit 7 as the PAI input, but the pin can also be used as general-purpose I/O or as an output compare.

NOTE: *Even when port A, bit 7 is configured as an output, the pin still drives the input to the pulse accumulator.*

DDA7 — Data Direction Control for Port A, Bit 7

0 = PA7 configured as an input

1 = PA7 configured as an output

9.7.2 Pulse Accumulator Control Register

Address: \$0026

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

Figure 9-23. Pulse Accumulator Control Register (PACTL)

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

PAMOD — Pulse Accumulator Mode Bit

0 = Event counter

1 = Gated time accumulation

PEDGE — Pulse Accumulator Edge Control Bit

In event counting mode, PEDGE selects either the rising or falling edge of the input at PA7 to increment the pulse accumulator counter. In gated accumulation mode, PEDGE determines which input level at PA7 inhibits counter increments from the internal clock. **Table 9-6** shows the relationship between PEDGE and PAMOD.

Table 9-6. Pulse Accumulator Edge Control

PAMOD	PEDGE	Action on Clock
0	0	PAI falling edge increments the counter.
0	1	PAI rising edge increments the counter.
1	0	A 0 on PAI inhibits counting.
1	1	A 1 on PAI inhibits counting.

9.7.3 Timer Interrupt Flag 2

Address: \$0025

	Bit 7	6	5	4	3	2	1	Bit 0
Read:	TOF	RTIF	PAOVF	PAIF	0	0	0	0
Write:								
Reset:	0	0	0	0	0	0	0	0

Figure 9-24. Timer Interrupt Flag 2 (TFLG2)

Clear each flag by writing a 1 to the corresponding bit position.

9.7.4 Timer Interrupt Mask 2 Register

Address: \$0024

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

Figure 9-25. Timer Interrupt Mask 2 (TMSK2)

Bits in TMSK2 correspond bit for bit with flag bits in TFLG2.

PAOVF — Pulse Accumulator Overflow Flag

The PAOVF status bit is set each time the pulse accumulator count rolls over from \$FF to \$00.

PAOVI — Interrupt Enable Bit

If PAOVI is set, an interrupt request is also generated. If PAOVI is cleared, pulse accumulator overflow interrupts are inhibited, and PAOVF must be polled by user software to determine when an overflow has occurred. In either case, software must clear PAOVF by writing a 1 to bit 5 in the TFLG2 register.

PAIF — Pulse Accumulator Input Edge Flag

The PAIF status bit is automatically set each time a selected edge is detected at the PA7 pin.

PAII — Interrupt Enable Bit

If PAII is set, an interrupt request is also generated. If PAII is cleared, pulse accumulator input interrupts are inhibited, and PAIF must be polled by user software to determine when an input edge has been detected. In either case, software must clear PAIF by writing a 1 to bit 5 in the TFLG2 register.

9.7.5 Pulse Accumulator Count Register

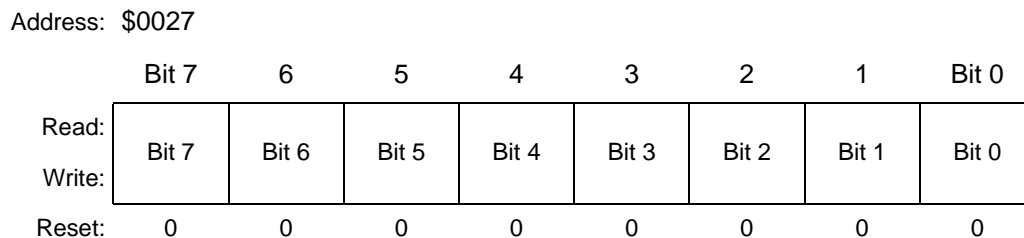


Figure 9-26. Pulse Accumulator Count Register (PACNT)

In event counting mode, PACNT contains the count of external input events at the PAI input. In gated accumulation mode, PACNT is incremented by the pulse accumulator's $E \div 64$ clock when the PAI input is at the selected level. Counting is synchronized to the internal PH2 clock so that incrementing and reading occur during opposite half cycles. The counter is not affected by reset and can be read or written to at any time.

9.8 Real-Time Interrupt (RTI)

The real-time interrupt (RTI) feature generates hardware interrupts at a fixed periodic rate. The rate is determined by bits RTR[1:0] in the PACTL register, which further divide a clock running at $E \div 2^{13}$ by 1, 2, 4 or 8. The resulting periods for various common crystal frequencies are shown in **Table 9-7**.

Every cycle of the RTI clock sets the RTIF bit in timer interrupt flag 2 (TFLG2) register. This flag can be polled to determine when RTI timeouts occur, or an interrupt can be generated if the RTII bit in the timer interrupt mask 2 (TMSK2) register is set. After reset, one entire real-time interrupt period elapses before the RTIF flag is set for the first time.

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

9.8.1 Timer Interrupt Flag 2 Register

Address: \$0025

	Bit 7	6	5	4	3	2	1	Bit 0
Read:	TOF	RTIF	PAOVF	PAIF	0	0	0	0
Write:								
Reset:	0	0	0	0	0	0	0	0

Figure 9-27. Timer Interrupt Flag 2 Register (TFLG2)

Clear each flag by writing a 1 to the corresponding bit position.

RTIF — Real-Time Interrupt Flag

The RTIF status bit is automatically set to 1 at the end of every RTI period.

9.8.2 Timer Interrupt Mask 2 Register

Address: \$0024

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

Figure 9-28. Timer Interrupt Mask 2 Register (TMSK2)

Bits in TMSK2 correspond bit for bit with flag bits in TFLG2.

RTII — Real-time Interrupt Enable Bit

0 = RTIF interrupts disabled

1 = Interrupt requested when RTIF is set to 1

9.8.3 Pulse Accumulator Control Register

Address: \$0026

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

Figure 9-29. Pulse Accumulator Control Register (PACTL)

RTR[1:0] — Real-Time Interrupt Rate Select Bits

These two bits select a divisor (1, 2, 4, or 8) for the $E \div 2^{13}$ RTI clock. Refer to **Table 9-7**.

Table 9-7. Real-Time Interrupt Rate versus RTR[1:0]

RTR[1:0]	Rate	XTAL = 12.0 MHz	XTAL = 2 ²³	XTAL = 8.0 MHz	XTAL = 4.9152 MHz	XTAL = 4.0 MHz	XTAL = 3.6864 MHz
0 0	$2^{13} \div E$	2.730 ms	3.91 ms	4.10 ms	6.67 ms	8.19 ms	8.89 ms
0 1	$2^{14} \div E$	5.461 ms	7.81 ms	8.19 ms	13.33 ms	16.38 ms	17.78 ms
1 0	$2^{15} \div E$	10.92 ms	15.62 ms	16.38 ms	26.67 ms	32.77 ms	35.56 ms
1 1	$2^{16} \div E$	21.84 ms	31.25 ms	32.77 ms	53.33 ms	65.54 ms	71.11 ms
	E =	3.0 MHz	2.1 MHz	2.0 MHz	1.2288 MHz	1.0 MHz	921.6 kHz

9.9 XOUT Pin Control

The XOUT pin provides a buffered XTAL signal to synchronize external devices with the MCU. It is enabled by the CLKX bit in the system configuration (CONFIG) register. The frequency of XOUT can be divided by one-of-four divisors selected by the XDV[1:0] bits in the system configuration options 2 (OPT2) register. The XOUT pin is not configured on all packages. Refer to the pin assignments in **Section 2. Pin Description**.

9.9.1 System Configuration Register

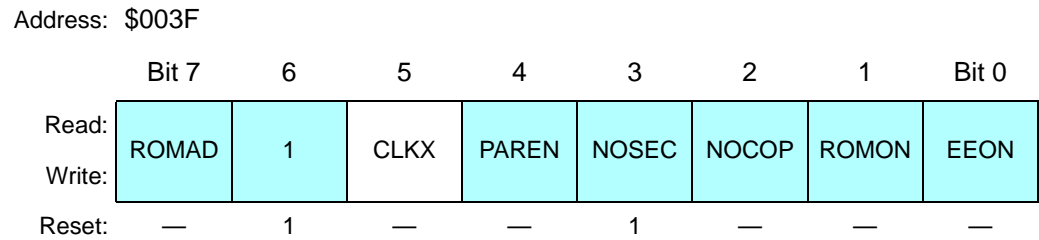


Figure 9-30. System Configuration Register (CONFIG)

Writable once in normal modes and writable at any time in special modes

CLKX — XOUT Clock Enable Bit

The CLKX bit is a switch that enables a buffered clock running at the same frequency as a referenced crystal. This buffered clock is intended to synchronize external devices with the MCU.

0 = The XOUT pin is disabled.

1 = The X clock is driven out on the XOUT pin.

9.9.2 System Configuration Options 2 Register

Address: \$0038

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

1. Not available on TK68HC11KS devices

Figure 9-31. System Configuration Options 2 Register (OPT2)

XDV[1:0] — XOUT Clock Divide Select Bits

These two bits select the divisor for the XOUT clock frequency, as shown in **Table 9-8**. The divisor is set to 1 out of reset (XOUT = XTAL). It takes a maximum of 16 cycles after writing these bits for XOUT to stabilize. The phase relationship between XOUT and XTAL cannot be predicted.

Table 9-8. XOUT Frequencies

XDV[1:0]	EXTAL Divided By	Frequency at EXTAL = 8 MHz	Frequency at EXTAL = 12 MHz	Frequency at EXTAL = 16 MHz	Frequency at EXTAL = 16 MHz
0 0	1	8 MHz	12 MHz	16 MHz	20 MHz
0 1	4	2 MHz	3 MHz	4 MHz	5 MHz
1 0	6	1.33 MHz	2 MHz	2.67 MHz	3.33 MHz
1 1	8	1 MHz	1.5 MHz	2 MHz	2.5 MHz

9.10 Pulse-Width Modulator (PWM)

Four 8-bit pulse-width modulation channels are available in the TK68HC11K Family devices. They are output on port H pins 3–0. Pairs of channels can be concatenated to produce 16-bit outputs. Three programmable clocks and a flexible clock selection scheme provide a wide range of frequencies.

The 8-bit mode with $E = 4$ MHz can produce waveforms from 40 kHz at 1 percent duty cycle resolution to less than 10 Hz at 0.4 percent duty cycle resolution. In 16-bit mode, a duty cycle resolution down to 15 parts per million can be achieved (at a frequency of 60 Hz). At 1 kHz, the duty cycle resolution is 250 ppm.

9.10.1 PWM System Description

Figure 9-32 shows a block diagram of the PWM system. Each of four channels is enabled by bit PWENx in the PWEN register. Each channel has an 8-bit counter (PWCNTx), a period register (PWPERx), and a duty cycle register (PWDTYx). The counter is driven by one of three user-scaled clock sources — clock A, B, or S — selected by the pulse-width channel select (PCLKx) bit in the pulse-width modulation timer polarity (PWPOL) register.

A pulse-width modulation period begins when the counter matches the value stored in the period register. When this happens, a logic value determined by the polarity bit (PPOLx) in the PWPOL register is driven on the associated port H output pin, and the counter is reset to 0. When the counter matches the number stored in the duty cycle register, the output reverses polarity. The period and duty cycle registers are double buffered so they can be changed without disturbing the current waveform. A new period or duty cycle can be forced by writing to the period (PWPERx) or duty cycle register (PWDTYx) and then to the counter (PWCNTx). Writing to the counter always resets it to 0.

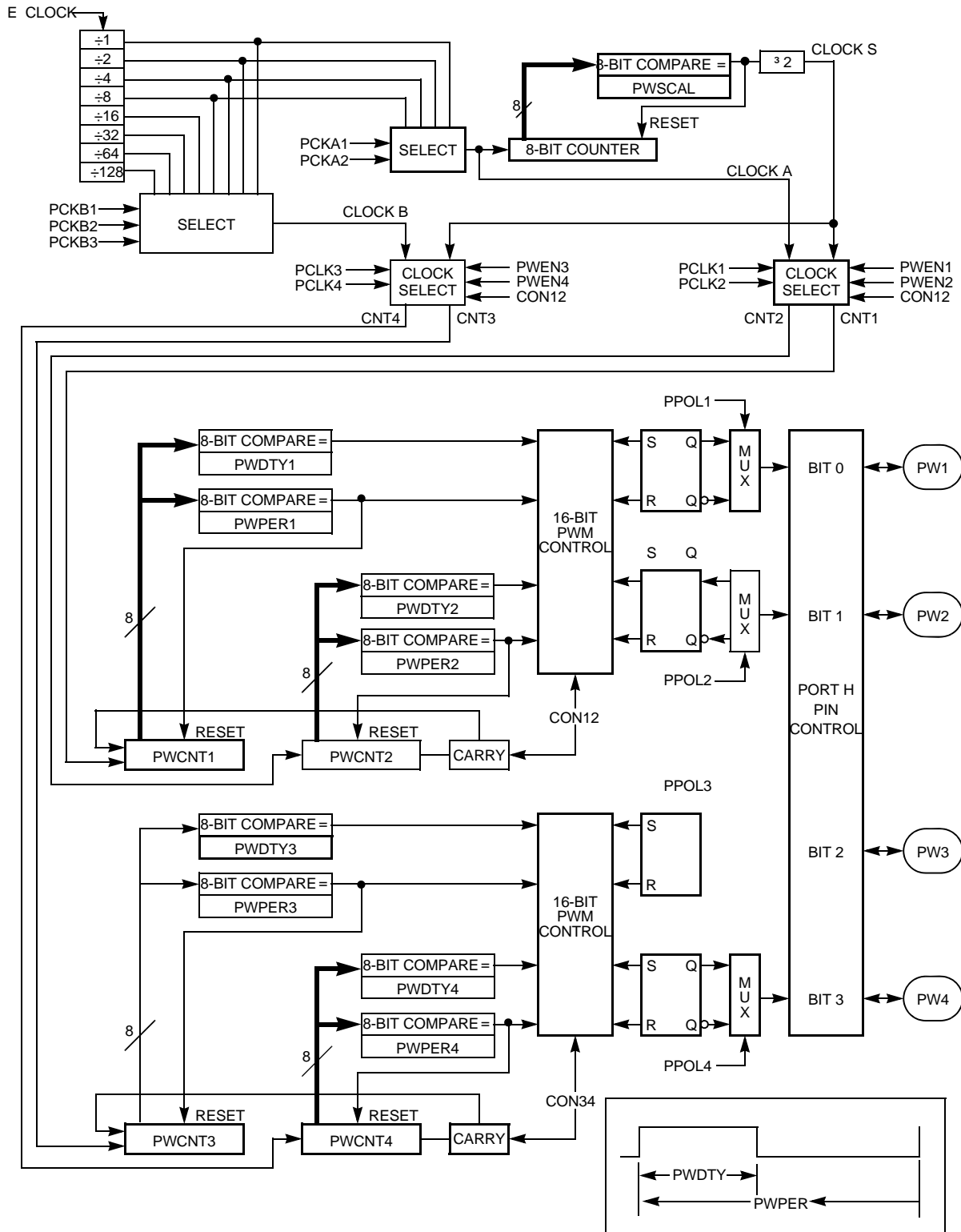


Figure 9-32. Pulse-Width Modulation Timer Block Diagram

The three clocks are derived from the E clock by writing to registers which determine their scaling factors. The clock A frequency is equal to E divided by 1, 2, 4, or 8, depending on which bits (PCKA[2:1]) in the PWCLK register are set. The clock B frequency is equal to the E clock divided by a power of two determined by bits PCKB[3:1] in the PWCLK register. Clock S is derived by dividing clock A by the integer (1 to 256) stored in the PWSCAL register, then by two.

Two channels can be concatenated by setting the appropriate bit (CON34 or CON12) in the PWCLK register. In this mode, the clock source is determined by the low-order channel, which is channel two in CON12 and channel four in CON34. The output is also placed on the pin associated with the low-order channels, so when two channels are concatenated the pin associated with the high-order channel (PH0 and/or PH2) can be used for GPIO. A read of the high-order byte causes the low-order byte to be latched for one cycle to guarantee that double-byte reads are accurate. A write to the low-order byte of the counter causes reset of the entire counter. A write to the high-order byte of the counter has no effect.

9.10.2 Pulse-Width Modulation Control Registers

The PWM control registers are described here.

9.10.2.1 Pulse-Width Modulation Timer Clock Select Register

Address: \$0060

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

Figure 9-33. Pulse-Width Modulation Timer Clock Select (PWCLK)

CON34 — Concatenate Channels 3 and 4 Bit

Channel 3 is the high-order byte, and channel 4 (port H, bit 3) is output.

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

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

CON12 — Concatenate Channels 1 and 2 Bit

Channel 1 is the high-order byte, and channel 2 (port H, bit 1) is output.

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

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

PCKA[2:1] — Prescaler for Clock A Bits

These bits select the frequency for clock A as shown in **Table 9-9**.

Table 9-9. Clock A Prescaler

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

PCKB[3:1] — Prescaler for Clock B Bits

These bits select the frequency for clock B as shown in **Table 9-10**.

Table 9-10. Clock B Prescaler

PCKB[3:1]	Clock B Frequency
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

9.10.2.2 Pulse-Width Modulation Timer Polarity Register

Address: \$0061

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

Figure 9-34. Pulse-Width Modulation Timer Polarity Register (PWPOL)

PCLK[4:3] — Pulse-Width Channel 4, 3 Clock Select Bits

- 0 = Clock B is source.
- 1 = Clock S is source.

PCLK[2:1] — Pulse-Width Channel 2, 1 Clock Select Bits

- 0 = Clock A is source.
- 1 = Clock S is source.

PPOL[4:1] — Pulse-Width Channel x Polarity Bits

- 0 = PWM channel x output is low at the beginning of the clock cycle and goes high when duty count is reached.
- 1 = PWM channel x output is high at the beginning of the clock cycle and goes low when duty count is reached.

9.10.2.3 Pulse-Width Modulation Timer Prescaler Register

Address: \$0062

	Bit 7	6	5	4	3	2	1	Bit 0
Read:	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Write:								
Reset:	0	0	0	0	0	0	0	0

Figure 9-35. Pulse-Width Modulation Timer Prescaler Register (PWSCAL)

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

9.10.2.4 Pulse-Width Modulation Timer Enable Register

Address: \$0063

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

Figure 9-36. Pulse-Width Modulation Timer Enable Register (PWEN)

TPWSL — PWM Scaled Clock Test Bit — factory use only; only accessible in special test mode

0 = Normal operation

1 = Clock S is output to PWSCAL register (test only)

DISCP — Disable Compare Scaled E-Clock Bit — factory use only; only accessible in special test mode

0 = Normal operation

1 = Match of period does not reset associated count register (test only)

PWEN[4:1] — Pulse-Width Enable for Channels [4:1] Bits

0 = Channel disabled

1 = Channel enabled at port H bits [3:0]

9.10.2.5 Pulse-Width Modulation Timer Counters 1 to 4 Registers

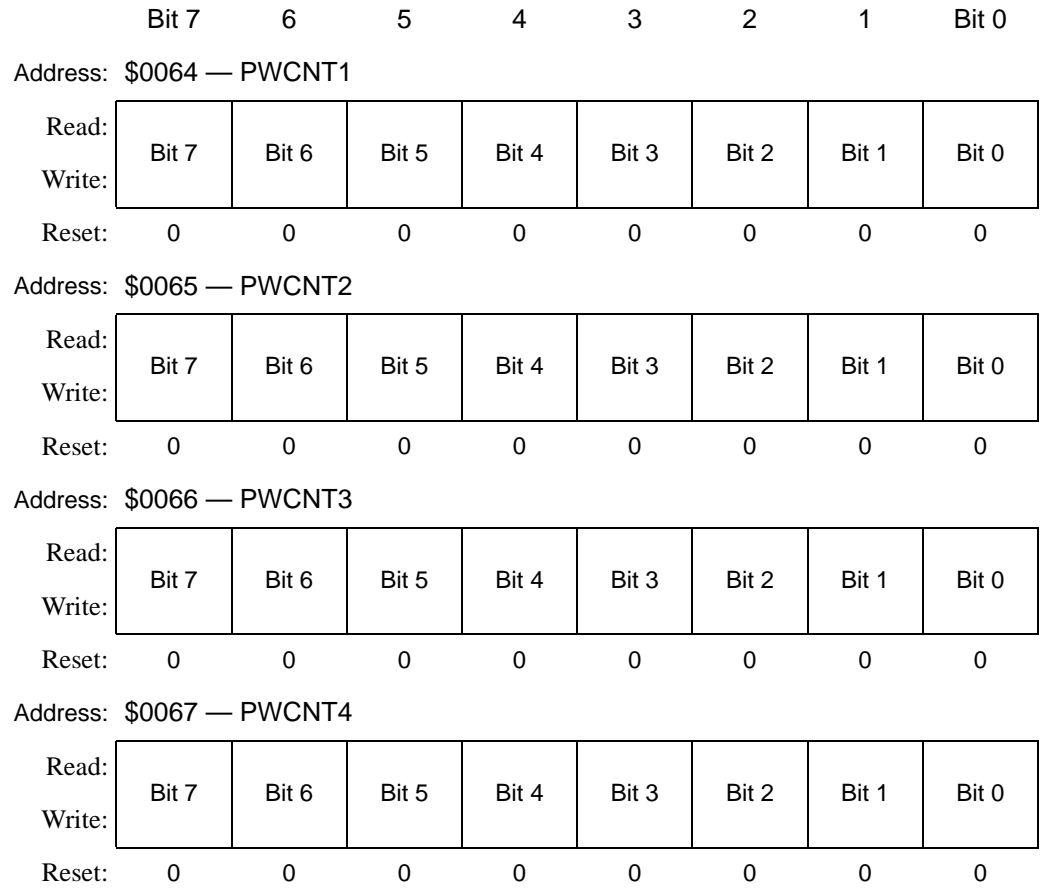


Figure 9-37. Pulse-Width Modulation Timer Counters 1 to 4 (PWCNT1 to PWCNT4)

Each counter resets to 0 when it is written and can be read at any time.

9.10.2.6 Pulse-Width Modulation Timer Periods 1 to 4 Registers

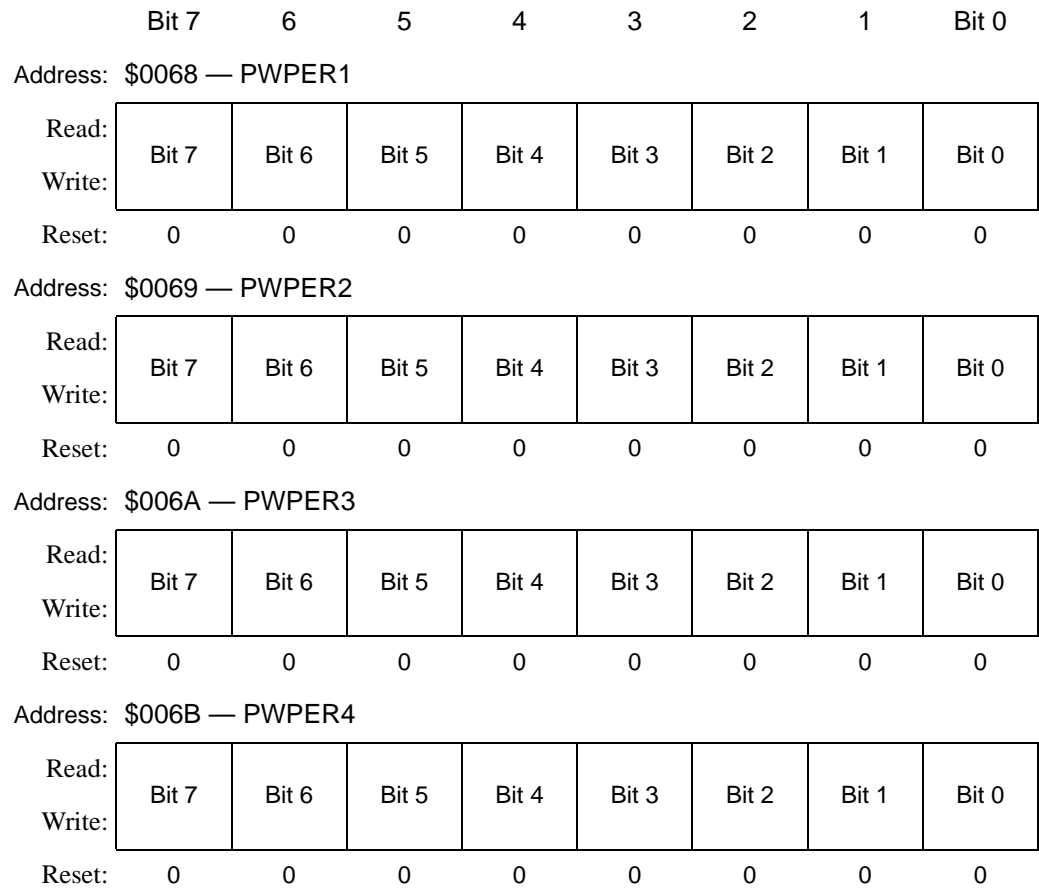
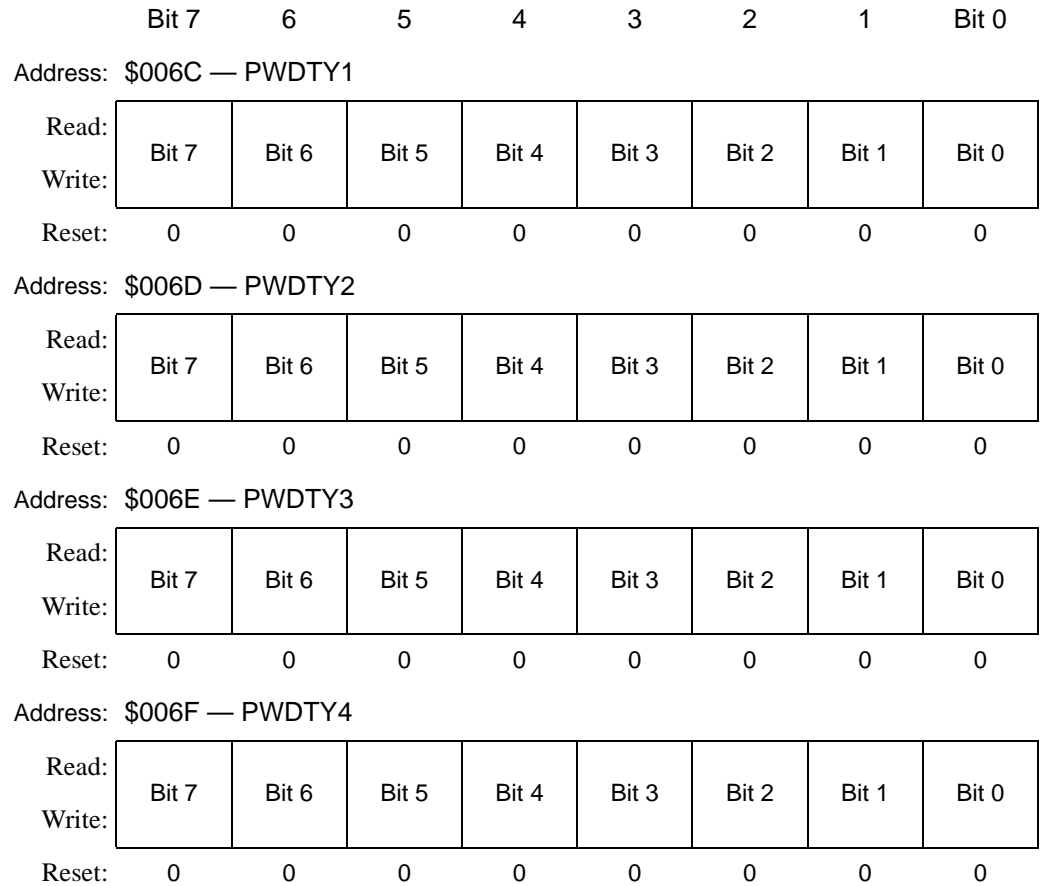


Figure 9-38. Pulse-Width Modulation Timer Periods 1 to 4 (PWPER1 to PWPER4)

Each period register can be read or written at any time. If it is written, the new period will not take effect until the associated counter is reset by a match or a write.

9.10.2.7 Pulse-Width Modulation Timer Duty Cycle 1 to 4 Registers



**Figure 9-39. Pulse-Width Modulation Timer
Duty Cycle 1 to 4 (PWDTY1 to PWDTY4)**

Each duty cycle register can be read or written at any time. If it is written, the new duty cycle will not take effect until the associated counter is reset by a match or a write.

Section 10. Analog-to-Digital (A/D) Converter

10.1 Contents

10.2	Introduction	219
10.3	Functional Description	220
10.3.1	Multiplexer	221
10.3.2	Analog Converter	221
10.3.3	Result Registers	221
10.3.4	Digital Control	222
10.4	A/D Control/Status Registers	224
10.4.1	System Configuration Options Register	224
10.4.2	A/D Control/Status Register	225
10.4.3	Analog-to-Digital Converter Result Registers	227
10.5	Design Considerations	228
10.5.1	A/D Input Pins	228
10.5.2	Operation in Stop and Wait Modes	228

10.2 Introduction

The analog-to-digital (A/D) system in TK68HC11 microcontrollers is an 8-channel, 8-bit, multiplexed input converter. It employs a successive approximation technique with an all-capacitive charge redistribution system that does not require external sample-and-hold circuits. A/D converter timing can be synchronized either to the E clock or an internal resistor-capacitor (RC) oscillator. Separate power supply inputs, AV_{DD} and AV_{SS} , allow independent bypassing for noise immunity.

Analog-to-Digital (A/D) Converter

10.3 Functional Description

The A/D converter system consists of four functional blocks as shown in **Figure 10-1**:

- Multiplexer
- Analog converter
- Result storage
- Digital control

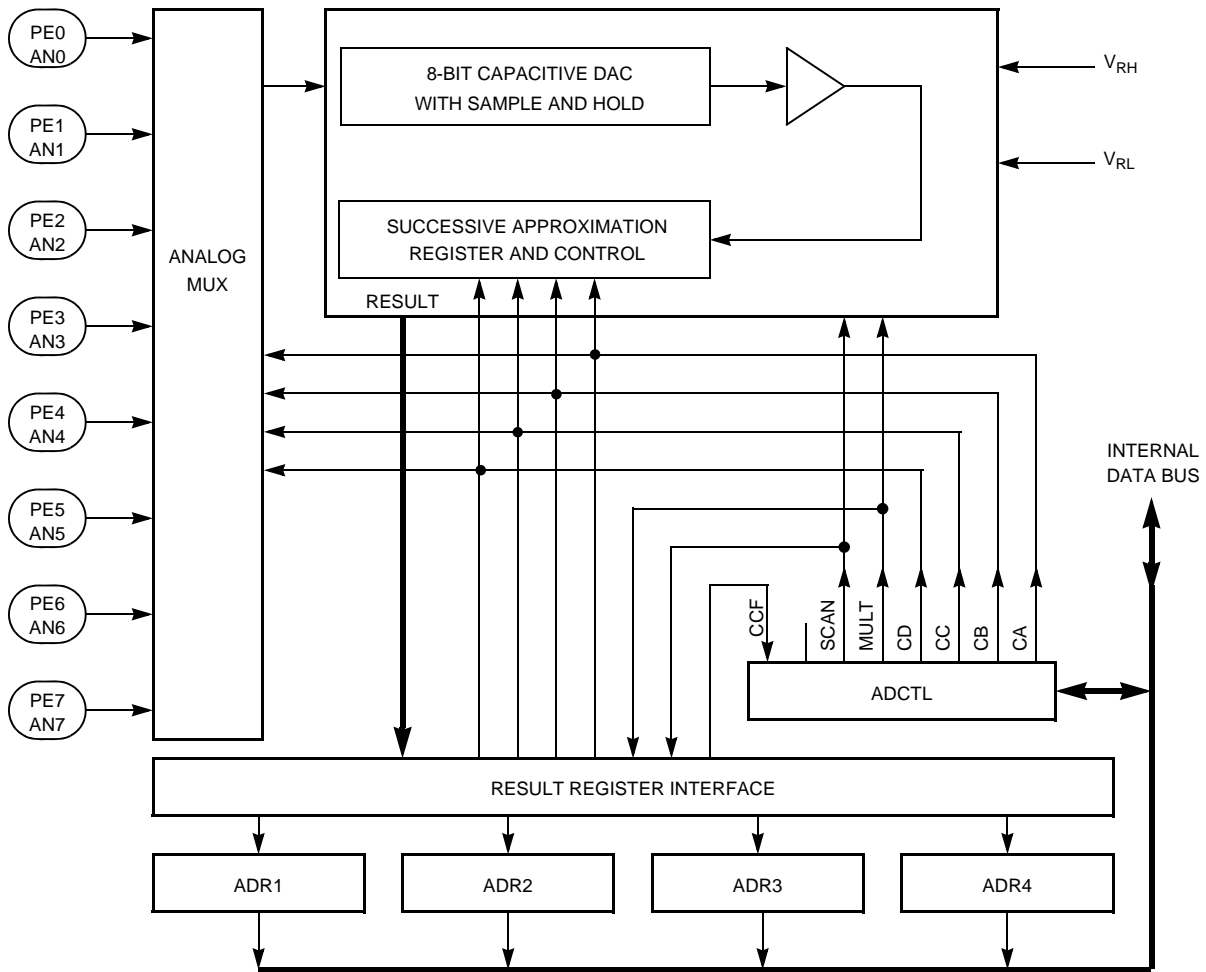


Figure 10-1. A/D Converter Block Diagram

10.3.1 Multiplexer

The multiplexer selects which of the eight analog inputs at port E will be converted. There are also three internal reference levels which can be multiplexed to the converter for system testing. Input selection is controlled by the value of bits CD:CA in the A/D control register (ADCTL).

The V_{RH} and V_{RL} pins provide inputs for the A/D system reference voltage. An input voltage equal to V_{RL} converts to \$00 and an input voltage equal to V_{RH} converts to \$FF (full scale), with no overflow indication. For ratiometric conversions of this type, the source of each analog input should use V_{RH} as the supply voltage and be referenced to V_{RL} .

10.3.2 Analog Converter

The conversion block contains a digital-to-analog capacitor (DAC) array, a comparator, and a successive approximation register (SAR). When an analog input is sampled for conversion, an analog switch connects the input to the DAC array. This series of scaled capacitors retains the sample for the duration of the conversion.

A conversion consists of a sequence of eight comparison operations. Each comparison determines one bit of the result, starting with the most significant bit (MSB). During each comparison, analog switches connect different elements of the DAC array to the comparator. The output of the comparator is stored in the next bit in the successive approximation register. When a conversion sequence is complete, the contents of the SAR are transferred to the appropriate result register.

10.3.3 Result Registers

The A/D conversion sequence begins one E-clock cycle after a write to the analog-to-digital control/status register (ADCTL). Converter operations are performed in sequences of four conversions each. Each conversion result in a sequence is stored in one of the four result registers, ADR[4:1]. The conversion complete flag (CCF) in the ADCTL register is set after the fourth conversion in a sequence to signal the availability of data in the result registers. The result registers are written during a portion of the system clock cycle when reads do not occur, so there is no conflict. A conversion sequence can repeat continuously or stop after one iteration. **Figure 10-2** shows the timing of a typical sequence. In this example, synchronization is referenced to the system E clock.

Analog-to-Digital (A/D) Converter

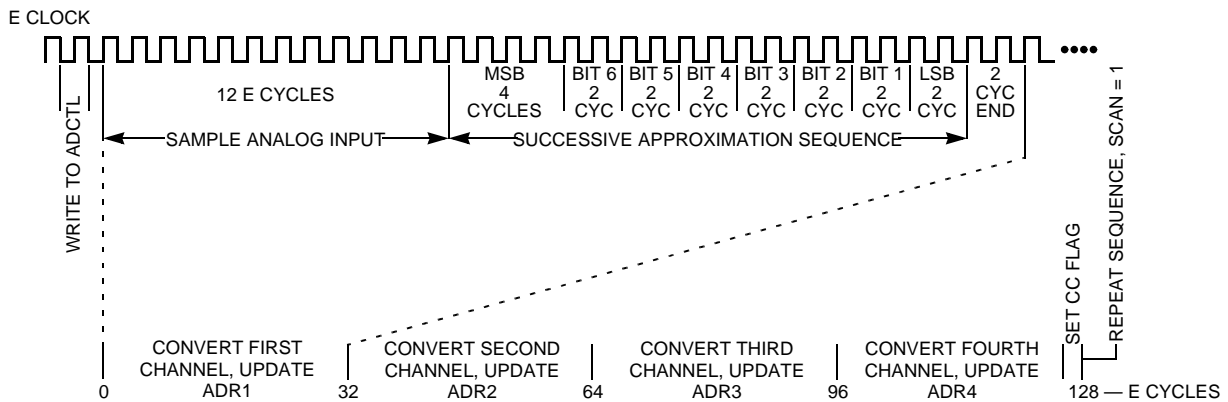


Figure 10-2. A/D Conversion Sequence

10.3.4 Digital Control

In addition to the conversion complete status flag, ADCTL bits select single or continuous conversions, whether conversions are performed on single or multiple channels, and the analog input(s) to be converted.

Single or continuous conversions are selected by the SCAN bit. Clearing the SCAN bit selects the single conversion option, in which results are written to each of the four result registers one time. The first result is stored in A/D result register 1 (ADR1), and the fourth result is stored in ADR4. All conversion activity is then halted until the ADCTL register is written again. In the continuous mode (SCAN = 1), conversion activity does not stop. The fifth conversion is stored in register ADR1 (overwriting the first conversion result), the sixth conversion overwrites ADR2, and so on.

The MULT bit in ADCTL determines whether one or four channels are converted. When MULT = 0, one channel is converted. The selected channel is sampled and the conversion results are written to ADR1; the same channel is sampled again, and the conversion results are stored in ADR2, and so on. In continuous mode, the same channel continues to be sampled and converted. Setting the MULT bit converts four different channels; each result register contains the conversion result of a different channel. In continuous mode, the same four channels are converted in each sequence.

ADCTL bits [3:0] select the particular channel(s) to be converted. See **Table 10-1**.

Table 10-1. A/D Converter Channel Selection

Channel Select Control Bits	Channel Signal	Result Location When MULT = 1
CD:CC:CB:CA		
0000	AN0	ADR1
0001	AN1	ADR2
0010	AN2	ADR3
0011	AN3	ADR4
0100	AN4	ADR1
0101	AN5	ADR2
0110	AN6	ADR3
0111	AN7	ADR4
10XX	Reserved	—
1100	$V_{RH}^{(1)}$	ADR1
1101	$V_{RL}^{(1)}$	ADR2
1110	$(V_{RH})/2^{(1)}$	ADR3
1111	Reserved ⁽¹⁾	ADR4

1. Used for factory testing

10.4 A/D Control/Status Registers

The registers involved in A/D operation include OPTION, ADCTL, and the four result registers ADR[1:4].

NOTE: Throughout this manual, the registers are discussed by function. In the event that not all bits in a register are referenced, the bits that are not discussed are shaded.

10.4.1 System Configuration Options Register

Bit 7 in the system configuration options register (OPTION), ADPU, enables the A/D converter system. Setting ADPU applies power to the A/D circuitry, including the charge pump that drives the analog switches. Clearing ADPU removes power from the A/D system.

The gates of analog switches in the multiplexer are driven by a charge pump that develops between seven and eight volts. The high gate voltage assures low source-to-drain impedance for the analog signals. Both the charge pump and the comparator circuits require up to 100 μ s to stabilize after setting the ADPU bit.

The CSEL bit (bit 6) determines whether the A/D converter uses the system E clock or an internal RC oscillator for synchronization. It is cleared out of reset, selecting the E clock. This is the preferred setting at normal operating frequencies because all switching and comparator operations are synchronized to the main MCU clocks. This allows the comparator output to be sampled at relatively quiet portions of the MCU clock cycles.

When the E clock frequency is less than 750 kHz, charge leakage in the capacitor array can cause errors. In this case, set the CSEL bit to select the internal oscillator, which usually runs at about 2 MHz. The additional error introduced by the asynchronous oscillator is about $\pm 1/2$ LSB (least significant bit), which is usually less than that incurred by a slow clock.

Address: \$0039

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

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

Figure 10-3. System Configuration Options Register (OPTION)

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

CSEL — Clock Select
0 = A/D and EEPROM use system E clock.
1 = A/D and EEPROM use internal RC clock.

10.4.2 A/D Control/Status Register

All bits in this register can be read or written except bit 7, which is a read-only status indicator, and bit 6, which always reads as 0. Writing to ADCTL initiates a conversion, aborting any conversion in progress.

Address: \$0030

	Bit 7	6	5	4	3	2	1	Bit 0
Read:	CCF	0	SCAN	MULT	CD	CC	CB	CA
Write:								
Reset:	—	0	—	—	—	—	—	—


 = Unimplemented

Figure 10-4. Analog-to-Digital Control/Status Register (ADCTL)

CCF — Conversions Complete Flag

Set when all four A/D result registers contain valid conversion results.

Cleared when the ADCTL register is overwritten, starting a new conversion sequence. In continuous mode, CCF is set at the end of the first conversion sequence.

SCAN — Continuous Scan Control Bit

0 = Conversion process stops after each of the four result registers is written.

1 = Conversions are performed continuously.

MULT — Multiple Channel/Single Channel Control Bit

0 = A single channel specified by the four channel select bits CD:CA is sampled and converted four times.

1 = Each of four channels is converted and the results written to a different result register.

NOTE: *When the multiple-channel continuous scan mode is used, extra care is needed in the design of circuitry driving the A/D inputs. The charge on the capacitive DAC array before the sample time is related to the voltage on the previously converted channel. A charge share situation exists between the internal DAC capacitance and the external circuit capacitance. Although the amount of charge involved is small, the rate at which it is repeated is every 64 μ s for an E clock of 2 MHz. The RC charging rate of the external circuit must be balanced against this charge sharing effect to avoid errors in accuracy. Refer to the TK68HC11 Reference Manual, Tekmos document order number TK68HC11RM/AD, for further information.*

CD:CA — Channel Selects D:A Bits

Refer to **Table 10-1**. In multiple channel mode (MULT = 1), the two least significant channel select bits (CB and CA) have no meaning and the CD and CC bits specify which group of four channels is to be converted.

10.4.3 Analog-to-Digital Converter Result Registers

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

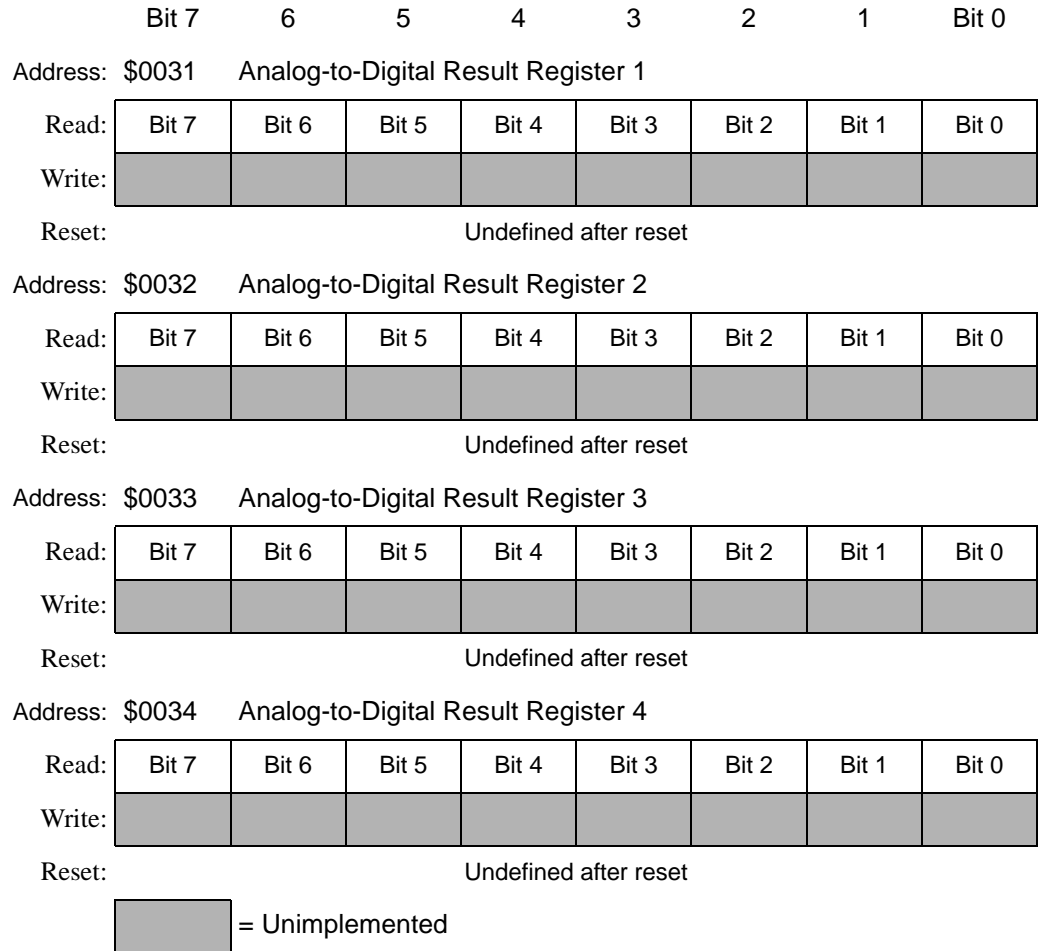


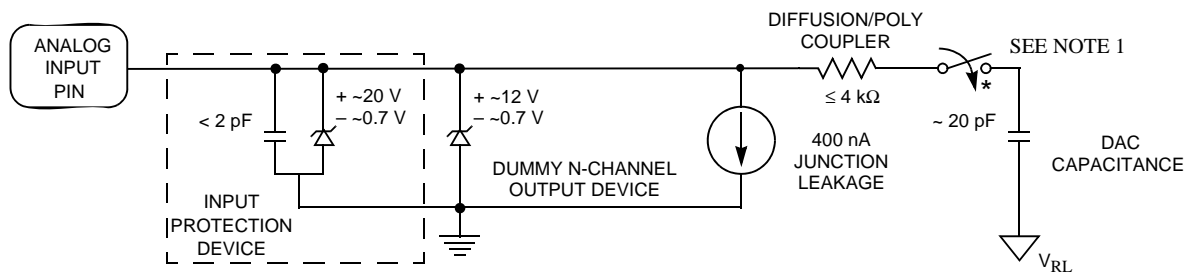
Figure 10-5. Analog-to-Digital Result Registers (ADR1–ADR4))

10.5 Design Considerations

This section discusses design considerations.

10.5.1 A/D Input Pins

Port E pins can also be used as general-purpose digital inputs. Digital reads of port E pins are not recommended during the sample portion of an A/D conversion cycle, when the gate signal to the N-channel input is on. No P-channel devices are directly connected to either input pins or reference voltage pins, so voltages above V_{DD} do not cause a latchup problem, although current should be limited according to maximum ratings. Refer to **Figure 10-3**.



Note 1. This analog switch is closed only during the 12-cycle sample time.

Figure 10-6. Electrical Model of an A/D Input Pin (Sample Mode)

10.5.2 Operation in Stop and Wait Modes

If a conversion sequence is in progress when either the stop or wait mode is entered, the conversion of the current channel is suspended. When the MCU resumes normal operation, that channel is resampled and the conversion sequence is resumed. 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, all analog bias currents are disabled and it is necessary to allow a stabilization period when leaving stop mode. If stop mode is exited with a delay (DLY = 1), there is enough time for these circuits to stabilize before the first conversion. If stop mode is exited with no delay (DLY bit in OPTION register = 0), allow 10 ms for the A/D circuitry to stabilize to avoid invalid results.

Section 11. Memory Expansion and Chip Selects

11.1 Contents

11.2	Introduction	229
11.3	Memory Expansion	230
11.3.1	Memory Size and Address Line Allocation	230
11.3.2	Control Registers	232
11.3.2.1	Port G Assignment Register	232
11.3.2.2	Memory Mapping Size Register	233
11.3.2.3	Memory Mapping Window Base Register	234
11.3.2.4	Memory Mapping Window Control Registers	235
11.4	Chip Selects	236
11.4.1	Program Chip Select	238
11.4.2	Input/Output Chip Select	239
11.4.3	General-Purpose Chip Selects	240
11.4.3.1	Memory Mapping Size Register	241
11.4.3.2	General-Purpose Chip Select 1 Address Register	241
11.4.3.3	General-Purpose Chip Select 1 Control Register	242
11.4.3.4	General-Purpose Chip Select 2 Address Register	243
11.4.3.5	General-Purpose Chip Select 2 Control Register	243
11.4.4	One Chip Select Driving Another	244
11.4.4.1	General-Purpose Chip Select 1 Control Register	245
11.4.4.2	General-Purpose Chip Select 2 Control Register	245
11.4.5	Clock Stretching	246
11.5	Memory Expansion Examples	247

11.2 Introduction

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

11.3 Memory Expansion

The TK68HC11K Family devices employ a register-based paging scheme to extend their address range beyond the physical 64-Kbyte limit of the 16 CPU address lines. Pages are selected using the expansion address lines XA[18:13] available on port G. This selection can be facilitated by the chip-select lines on port H, discussed in **11.4 Chip Selects**. The TK68HC11KS devices do not provide these features since they lack the required port G and port H lines. Refer to **Figure 1-2. TK68HC11KS Family Block Diagram**.

11.3.1 Memory Size and Address Line Allocation

To access expanded memory, the user first allocates portion(s) of the 64 Kbyte address space, or window(s), through which the CPU will view external memory. One or two windows can be designated, and the size of each window can be 0 (disabled), 8, 16, or 32 Kbytes.

Expanded memory is addressed with a combination of the CPU's normal address lines ADDR[15:0] and the expansion address lines XA[18:13]. The expansion address lines select a memory bank, and the CPU's normal address lines select a particular location within the bank. The size of the window(s) and the number of memory banks determine exactly which expansion address lines are used. The port G assignment register (PGAR) controls which port G pins function as expanded address lines. Any port G pins not allocated for memory expansion can serve as general-purpose input/output (GPIO). When a configuration uses any of the lower three expansion address lines XA[15:13] they replace the CPU's equivalent address lines (ADDR[15:13]). **Table 11-1** shows how address and expansion lines are allocated for various combinations of memory banks and window size.

Table 11-1. CPU Address and Address Expansion Signals

Number of Banks	Window Size			
	8 Kbytes	16 Kbytes	32 Kbytes	32 Kbytes (Window Based at \$4000)
2	ADDR[12:0] XA13	ADDR[13:0] XA14	ADDR[14:0] XA15	ADDR[13:0] XA[15:14]
4	ADDR[12:0] XA[14:13]	ADDR[13:0] XA[15:14]	ADDR[14:0] XA[16:15]	ADDR[13:0] XA[16:14]
8	ADDR[12:0] XA[15:13]	ADDR[13:0] XA[16:14]	ADDR[14:0] XA[17:15]	ADDR[13:0] XA[17:14]
16	ADDR[12:0] XA[16:13]	ADDR[13:0] XA[17:14]	ADDR[14:0] XA[18:15]	ADDR[13:0] XA[18:14]
32	ADDR[12:0] XA[17:13]	ADDR[13:0] XA[18:14]	— —	— —
64	ADDR[12:0] XA[18:13]	— —	— —	— —

The base address for each window must be an integer multiple of the window size, with one exception. When the window size is 32 Kbytes, the base address can be at \$4000 as well as the 32-Kbyte multiples \$0000 and \$8000.

This special case requires a modification in address line deployment. Normally, when the bank size is 32 Kbytes and the bank address is \$0000 or \$8000, CPU address lines ADDR[14:0] select individual bytes within the 32-Kbyte space and the ADDR[14:0] pins are connected to address lines A[14:0] of the memory device. When the base address is \$4000, the CPU address signal ADDR14 must be inverted to allow 32 Kbytes of contiguous memory. To do this, the CPU drives the inverted ADDR14 signal onto the XA14 pin when the window is active, and the non-inverted CPU ADDR14 signal onto the XA14 pin when the window is not active. Therefore, address 14 of the memory device must be connected to expansion line XA14 rather than normal address line ADDR14.

If the two memory windows overlap, window 1 has priority, and only the portion of window 2 that does not overlap window 1 remains active. If a window overlaps any portion of internal registers, RAM, or EEPROM, that portion is repeated in all banks associated with that window. If a window overlaps (EP)ROM, the (EP)ROM is present in all banks with XA[18:16] = 0:0:0.

The reset vector most commonly resides in on-chip (EP)ROM at address \$FFFE–\$FFFF. However, if the (EP)ROM is disabled or mapped at address \$2000–\$7FFF, the reset vector is fetched from external memory at

\$FFFE–\$FFFF. When expanded memory is enabled, the reset vector is fetched from external memory at \$7FFE–\$7FFF, regardless of the presence of on-chip (EP)ROM.

11.3.2 Control Registers

Expansion address lines are enabled by the port G assignment register (PGAR). The size and position of memory windows are controlled by the memory mapping size (MMSIZ) and memory mapping window base (MMWBR) registers, respectively. The memory mapping window control registers, MM1CR and MM2CR, select the particular bank or page of expanded memory present in the window(s) at a given time.

NOTE: *Throughout this manual, the registers are discussed by function. In the event that not all bits in a register are referenced, the bits that are not discussed are shaded.*

11.3.2.1 Port G Assignment Register

The port G assignment register (PGAR) sets each of port G pins 5:0 as either an input/output (I/O) pin or memory expansion address line. Clearing a bit configures the corresponding port G pin as GPIO; setting the bit configures the pin as an expansion address line. If neither bank uses a particular expansion address bit, the corresponding pin is available for GPIO.

NOTE: *A special case exists for the address lines that overlap the CPU address lines XA[15:13]. If these lines are selected as expansion address lines in PGAR, but are not used in either window, the corresponding CPU address line is still output on the appropriate pin.*

Address: \$002D

	Bit 7	6	5	4	3	2	1	Bit 0
Read:	—	—	PGAR5	PGAR4	PGAR3	PGAR2	PGAR1	PGAR0
Write:								
Reset:	0	0	0	0	0	0	0	0

Figure 11-1. Port G Assignment Register (PGAR)

PGAR[5:0] — Port G Pin Assignment Bits

0 = Corresponding port G pin is GPIO.

1 = Corresponding port G pin is expansion address line XA[18:13].

11.3.2.2 Memory Mapping Size Register

The memory mapping size register (MMSIZ) sets the size of the windows in use.

Address: \$0056

	Bit 7	6	5	4	3	2	1	Bit 0
Read:	MXGS2	MXGS1	W2SZ1	W2SZ0	0	0	W1SZ1	W1SZ0
Write:								
Reset:	0	0	0	0	0	0	0	0

Figure 11-2. Memory Mapping Size Register (MMSIZ)

W2SZ[1:0] — Window 2 Size Bit

W1SZ[1:0] — Window 1 Size Bit

These bits enable the memory windows and determine their size, as shown in **Table 11-2**.

Table 11-2. Window Size Select

WxSZ[1:0]	Window Size
00	Window disabled
01	8 K — Window can have up to 64 8-Kbyte banks
10	16 K — Window can have up to 32 16-Kbyte banks
11	32 K — Window can have up to 16 32-Kbyte banks

Memory Expansion and Chip Selects

11.3.2.3 Memory Mapping Window Base Register

The memory mapping window base register (MMWBR) defines the starting address of each of the two windows within the CPU 64-Kbyte address range. The windows normally begin at a boundary related to their size (an 8-Kbyte window can begin on any 8-Kbyte boundary, beginning at \$0000).

Address: \$0057

	Bit 7	6	5	4	3	2	1	Bit 0
Read:	W2A15	W2A14	W2A13	0	W1A15	W1A14	W1A13	0
Write:								
Reset:	0	0	0	0	0	0	0	0

Figure 11-3. Memory Mapping Window Base Register (MMWBR)

W2A[15:13] — Window 2 Base Address Bits

Selects the three most significant bits (MSB) of the base address for memory mapping window 2. Refer to **Table 11-3**.

W1A[15:13] — Window Base 1 Address Bits

Selects the three MSB of the base address for memory mapping window 1. Refer to **Table 11-3**.

Table 11-3. Memory Expansion Window Base Address

MSB Bits	Window Base Address		
	8 Kbytes	16 Kbytes	32 Kbytes
WxA[15:13]			
000	\$0000	\$0000	\$0000
001	\$2000	\$0000	\$0000
010	\$4000	\$4000	\$4000
011	\$6000	\$4000	\$4000
100	\$8000	\$8000	\$8000
101	\$A000	\$8000	\$8000
110	\$C000	\$C000	\$8000
111	\$E000	\$C000	\$8000

11.3.2.4 Memory Mapping Window Control Registers

Each of the memory mapping window control registers (MM1CR and MM2CR) determine the active memory bank for the corresponding window, containing

the value to be output on the expansion address lines when the CPU selects addresses within its extended memory window. To change banks, write the address of the new bank into the appropriate window register.

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

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

Address: \$0059 Memory Mapping Window 2 Control Register (MM2CR)

	Bit 7	6	5	4	3	2	1	Bit 0
Read:	0	X2A18	X2A17	X2A16	X2A15	X2A14	X2A13	0
Write:	0	X2A18	X2A17	X2A16	X2A15	X2A14	X2A13	0
Reset:	0	0	0	0	0	0	0	0

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

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

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

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

11.4 Chip Selects

The TK68HC11K Family microcontrollers contain a set of four software-configured chip select signals which can reduce the amount of external glue logic needed to interface the MCU to external devices in the memory map. Each chip select signal is asserted when the CPU accesses a memory location in its designated range, which is determined by writes to control registers. Polarity of most chip-select signals is programmable, as well as the segment of the address cycle during which the signal is asserted (high E clock or address valid). Some chip-select signals can be programmed to drive each other as well as their own designated sections of memory (see **11.4.4 One Chip Select Driving Another**), and the address cycle during which any chip-select signal is asserted can be “stretched” up to three clock cycles (see **11.4.5 Clock Stretching**).

The four chip-select signals operate only in expanded modes. The program chip select ($\overline{\text{CSPROG}}$) is used to enable external memory in the 64-Kbyte memory map that contains the reset vectors and program. The chip select for I/O (CSIO) operates only within the first eight Kbytes of the memory map. The two general-purpose chip selects, CSGP1 and CSGP2, can enable devices anywhere in the 1-Mbyte expanded memory space. Any port H pins 4-7 that are not used for chip-select functions can serve as GPIO pins.

The six chip select control registers are:

- CSCTL enables and controls most of the features in $\overline{\text{CSPROG}}$ and CSIO.
- GPCS1A, GPCS1C, GPCS2A, and GPCS2C define most of the operations of the two general-purpose chip selects.
- CSCSTR controls clock stretching for all four signals.

Table 11-4 summarizes the controls for each of the chip-select signals.

Table 11-4. Chip Select Control Parameter Summary

CSIO	Enable	IOEN in CSCTL	1 = enabled, 0 = disabled ⁽¹⁾	
	Valid	IOCSA in CSCTL	1 = address valid, 0 ⁽¹⁾ = E high	
	Polarity	IOPL in CSCTL	1 = active high, 0 = active low ⁽¹⁾	
	Size	IOSZ in CSCTL	1 = 4 K (\$1000–\$1FFF) 0 = 8 K (\$0000–\$1FFF) ⁽¹⁾	
	Start address	Fixed (see size)		
	Stretch	IO1S[A:B] in CSCSTR	0 ⁽¹⁾ , 1, 2, or 3 E clocks	
CSPROG	Enable	PCSEN in CSCTL	1 = enabled ⁽¹⁾ , 0 = disabled	
	Valid	Fixed (address valid)		
	Polarity	Fixed (active low)		
	Size	PCSZ[A:B] in CSCTL	0:0 = 64 K (\$0000–\$FFFF) ⁽¹⁾ 0:1 = 32 K (\$8000–\$FFFF) 1:0 = 16 K (\$C000–\$FFFF) 1:1 = 8 K (\$E000–\$FFFF)	
	Start address	Fixed (see size)		
	Stretch	PCS[A:B] in CSCSTR	0 ⁽¹⁾ , 1, 2, or 3 E clocks	
	Priority	GCSPR in CSCTL	1 = CSGPx above CSPROG 0 ⁽¹⁾ = CSPROG above CSGPx	
CSGP1,	Enable	Set size to 0K to disable		
CSGP2	Valid	G1AV in GPCS1C G2AV in GPCS2C	1 = address valid, 0 = E high ⁽¹⁾	
	Polarity	G1POL in GPS1C G2POL in GPS2C	1 = active high, 0 = active low ⁽¹⁾	
	Size	G1SZ[A:D] in GPCS1C G1SZ[A:D] in GPCS2C	2 K to 512 K in nine steps 0K = disabled ⁽¹⁾ can also follow memory expansion window 1 or window 2	
	Start address	GPCS1A GPCS2A		
	Stretch	CSCSTR	0 ⁽¹⁾ , 1, 2, or 3 E clocks	
	Other		G1DG2 in GPCS1C	Allows CSGP1 and CSGP2 to be logically ORed and driven out the CSGP2 pin
			G1DPC in GPCS1C	Allows CSGP1 and CSPROG to be logically ORed and driven out the CSPROG pin
			G2DPC in GPCS2C	Allows CSGP2 and CSPROG to be logically ORed and driven out the CSPROG pin.
		MXGS2 in MMSIZ	Allows CSGP2 to follow either 64 K CPU addresses or 512K expansion addresses	
	MXGS1 in MMSIZ	Allows CSGP1 to follow either 64 K CPU addresses or 512K expansion addresses		

1. Configuration at reset

11.4.1 Program Chip Select

The program chip select signal accesses external memory in the main program area within the MCU's 64-Kbyte memory map. Program chip select validity is fixed at address valid timing and polarity is fixed at active low. The chip-select control register (CSCTL) contains bits to enable $\overline{\text{CSPROG}}$, determine its priority over the general-purpose chip selects, and set its effective address range. Clock stretching can be set from zero to three cycles.

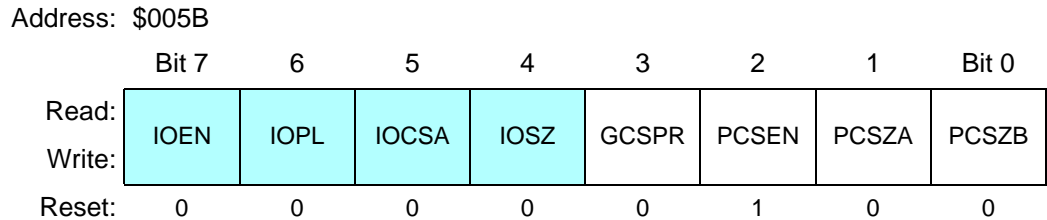


Figure 11-5. Chip-Select Control Register (CSCTL)

GCSPR — General-Purpose Chip Select Priority Bit

- 0 = Program chip select has priority over general-purpose chip selects
- 1 = General-purpose chip selects have priority over program chip select

PCSEN — Program Chip Select Enable Bit

- 0 = $\overline{\text{CSPROG}}$ disabled and port H bit 7 available as GPIO
- 1 = $\overline{\text{CSPROG}}$ enabled out of reset and uses port H bit 7 pin

PCSZA and PCSZB — Program Chip Select Size A Bit and Program Chip Select Size B Bit

These bits determine the address range of $\overline{\text{CSPROG}}$, as shown in **Table 11-5**.

Table 11-5. Program Chip Select Size

PCSZA	PCSZB	Size (Bytes)	Address Range
0	0	64 K	\$0000–\$FFFF
0	1	32 K	\$8000–\$FFFF
1	0	16 K	\$C000–\$FFFF
1	1	8 K	\$E000–\$FFFF

11.4.2 Input/Output Chip Select

The I/O chip select (CSIO) is programmable for a 4-Kbyte size located at addresses \$1000–\$1FFF or 8-Kbyte size located at addresses \$0000–\$1FFF. The default active-low polarity can be changed to active high by setting the

IOPL bit in CSCTL. Default validity during high E clock can be changed to address valid time by setting the IOCSA bit in CSCTL. Clock stretching can be set from zero to three cycles (See **11.4.5 Clock Stretching**). CSIO is disabled out of reset.

Address: \$005B

	Bit 7	6	5	4	3	2	1	Bit 0
Read:	IOEN	IOPL	IOCSA	IOSZ	GCSPR	PCSEN	PCSZA	PCSZB
Write:								
Reset:	0	0	0	0	0	1	0	0

Figure 11-6. Chip-Select Control Register (CSCTL)

IOEN — I/O Chip-Select Enable Bit

- 0 = CSIO is disabled and port H bit 4 is GPIO.
- 1 = CSIO is enabled and uses port H bit 4.

IOPL — I/O Chip-Select Polarity Select Bit

- 0 = CSIO active low
- 1 = CSIO active high

IOCSA — I/O Chip-Select Address Valid Bit

- 0 = CSIO is valid during E-clock high time.
- 1 = CSIO is valid during address valid time.

IOSZ — I/O Chip-Select Size Select Bit

- 0 = CSIO size is four Kbytes at \$1000–\$1FFF.
- 1 = CSIO size is eight Kbytes at \$0000–\$1FFF.

11.4.3 General-Purpose Chip Selects

The general-purpose chip selects are the most flexible and programmable of the chip-select signals. They can access any memory in the expanded 1-Mbyte address space. Polarity of active state, E valid or address valid, size, and starting address are all programmable. Clock stretching can be set from zero to three cycles. Both signals can be programmed to drive CSPROG, and GPCS1 can be configured to drive GPCS2. In addition, each signal can follow a window; for instance, be asserted whenever the CPU address falls within a selected memory expansion window regardless of the state of the expanded address lines.

There are two registers for each of the general-purpose chip select signals:

- The control register, GPS1C or GPS2C, determines the GPCS's active signal polarity, its valid time, which of the other chip-select signals it can drive, and either the size of the memory it enables or which window it follows.
- The address register, GPS1A or GPS2A, programs the chip-select's starting address; valid bits in this register are determined by the size of the address range selected by the control register.

In addition, the MMSIZ register contains a bit for each GPCS which determines whether it is driven by the CPU's 64-Kbyte address lines or the expansion address lines.

11.4.3.1 Memory Mapping Size Register

Address: \$0056

	Bit 7	6	5	4	3	2	1	Bit 0
Read:	MXGS2	MXGS1	W2SZ1	W2SZ0	0	0	W1SZ1	W1SZ0
Write:								
Reset:	0	0	0	0	0	0	0	0

Figure 11-7. Memory Mapping Size Register (MMSIZ)

MXGS2 — Memory Expansion Select for GPCS 2 Bit
 0 = GPCS 2 based on 64-Kbyte CPU address
 1 = GPCS 2 based on expansion address

MXGS1 — Memory Expansion Select for GPCS 1 Bit
 0 = GPCS 1 based on 64-Kbyte CPU address
 1 = GPCS 1 based on expansion address

11.4.3.2 General-Purpose Chip Select 1 Address Register

Address: \$005C

	Bit 7	6	5	4	3	2	1	Bit 0
Read:	G1A18	G1A17	G1A16	G1A15	G1A14	G1A13	G1A12	G1A11
Write:								
Reset:	0	0	0	0	0	0	0	0

Figure 11-8. General-Purpose Chip Select 1 Address Register (GPCS1A)

G1A[18:11] — General-Purpose Chip Select 1 Address Bits
 They select the starting address of GPCS1. Refer to **Table 11-6**.

Memory Expansion and Chip Selects

11.4.3.3 General-Purpose Chip Select 1 Control Register

Address: \$005D

	Bit 7	6	5	4	3	2	1	Bit 0
Read:	G1DG2	G1DPC	G1POL	G1AV	G1SZA	G1SZB	G1SCC	G1SZD
Write:								
Reset:	0	0	0	0	0	0	0	0

Figure 11-9. General-Purpose Chip Select 1 Control Register (GPCS1C)

G1POL — General-Purpose Chip Select 1 Polarity Select Bit

0 = CSGP1 active low

1 = CSGP1 active high

G1AV — General-Purpose Chip Select 1 Address Valid Select Bit

0 = CSGP1 is valid during E high time.

1 = CSGP1 is valid during address valid time.

G1SZ[A:D] — General-Purpose Chip Select 1 Size Bits

They select the range of GPCS1. Refer to **Table 11-6**.

Table 11-6. General-Purpose Chip Select 1 Size Control

G1SZ[A:D]	Size (Bytes)	Valid Bits (MXGS1 = 0)	Valid Bits (MXGS1 = 1)
0 0 0 0	Disabled	None	None
0 0 0 1	2 K	G1A[15:11]	G1A[18:11]
0 0 1 0	4 K	G1A[15:11]	G1A[18:12]
0 0 1 1	8 K	G1A[15:13]	G1A[18:13]
0 1 0 0	16 K	G1A[15:14]	G1A[18:14]
0 1 0 1	32 K	G1A[15]	G1A[18:15]
0 1 1 0	64 K	None	G1A[18:16]
0 1 1 1	128 K	None	G1A[18:17]
1 0 0 0	256 K	None	G1A18
1 0 0 1	512 K	None	None
1 0 1 0	Follow window 1	None	None
1 0 1 1	Follow window 2	None	None
1100–1111	Default to 512 K	None	None

11.4.3.4 General-Purpose Chip Select 2 Address Register

Address: \$005E

	Bit 7	6	5	4	3	2	1	Bit 0
Read:	G2A18	G2A17	G2A16	G2A15	G2A14	G2A13	G2A12	G2A11
Write:								
Reset:	0	0	0	0	0	0	0	0

Figure 11-10. General-Purpose Chip Select 2 Address Register (GPCS2A)

G2A[18:11] — General-Purpose Chip Select 2 Address Bits
They select the starting address of GPCS2. Refer to **Table 11-7**.

11.4.3.5 General-Purpose Chip Select 2 Control Register

Address: \$005F

	Bit 7	6	5	4	3	2	1	Bit 0
Read:	0	G2DPC	G2POL	G2AV	G2SZA	G2Szb	G2SZC	G2SZD
Write:								
Reset:	0	0	0	0	0	0	0	0

Figure 11-11. General-Purpose Chip Select 2 Control Register (GPCS2C)

G2POL — General-Purpose Chip Select 2 Polarity Select Bit
0 = CSGP2 active low
1 = CSGP2 active high

G2AV — General-Purpose Chip Select 2 Address Valid Select Bit
0 = CSGP2 is valid during E high time.
1 = CSGP2 is valid during address valid time.

G2SZ[A:D] — General-Purpose Chip Select 2 Size Bits
They select the range of GPCS2. Refer to **Table 11-7**.

Table 11-7. General-Purpose Chip Select 2 Size Control

G2SZ[A:D]	Size (Bytes)	Valid Bits (MXGS2 = 0)	Valid Bits (MXGS2 = 1)
0 0 0 0	Disabled	None	None
0 0 0 1	2 K	G2A[15:11]	G2A[18:11]
0 0 1 0	4 K	G2A[15:11]	G2A[18:12]
0 0 1 1	8 K	G2A[15:13]	G2A[18:13]
0 1 0 0	16 K	G2A[15:14]	G2A[18:14]
0 1 0 1	32 K	G2A[15]	G2A[18:15]
0 1 1 0	64 K	None	G2A[18:16]
0 1 1 1	128 K	None	G2A[18:17]
1 0 0 0	256 K	None	G2A18
1 0 0 1	512 K	None	None
1 0 1 0	Follow window 1	None	None
1 0 1 1	Follow window 2	None	None
1100–1111	Default to 512 K	None	None

11.4.4 One Chip Select Driving Another

The general-purpose chip selects can be programmed to drive other chip-select signals as well as their own memory areas. Although all of the eight possible combinations of the bits G1DG2, G1DPC, and G2DPC are allowed, some combinations cause operations which do not perform as one might expect. The results of all combinations are defined in **Table 11-8**. The priorities defined in the previous sections still apply. The table assumes that none of the chip-select ranges overlap.

11.4.4.1 General-Purpose Chip Select 1 Control Register

Address: \$005D

	Bit 7	6	5	4	3	2	1	Bit 0
Read:	G1DG2	G1DPC	G1POL	G1AV	G1SZA	G1SZB	G1SCC	G1SZD
Write:								
Reset:	0	0	0	0	0	0	0	0

Figure 11-12. General-Purpose Chip Select 1 Control Register (GPCS1C)

G1DG2 — GPCS 1 Drives GPCS 2 Bit

0 = CSGP1 does not affect CSGP2.

1 = CSGP1 and CSGP2 are ORed and driven out of the CSGP2.

G1DPC — General-Purpose Chip Select 1 Drives Program Chip Select Bit

0 = CSGP1 does not affect CSPROG.

1 = CSGP1 and CSPROG are ORed and driven out of the CSPROG.

11.4.4.2 General-Purpose Chip Select 2 Control Register

Address: \$005F

	Bit 7	6	5	4	3	2	1	Bit 0
Read:	0	G2DPC	G2POL	G2AV	G2SZA	G2SZB	G2SZC	G2SZD
Write:								
Reset:	0	0	0	0	0	0	0	0

Figure 11-13. General-Purpose Chip Select 2 Control Register (GPCS2C)

G2DPC — General-Purpose Chip Select 2 Drives Program Chip Select Bit

0 = Does not affect program chip select

1 = CSGP2 and CSPROG are ORed and driven out of the CSPROG pin.

Table 11-8. One Chip Select Driving Another

G1DG2	G1DPC	G2DPC	Program CS Pin is Asserted When Address is in:	General 2 CS Pin is Asserted When Address is in:	General 1 CS Pin is Asserted When Address is in:
0	0	0	A valid program area	A valid general 2 area	A valid general 1 area
0	0	1	A valid program or general 2 area	Never asserted	A valid general 1 area
0	1	0	A valid program or general 1 area	A valid general 2 area	Never asserted
0	1	1	A valid program or general 1 or 2 area	Never asserted	Never asserted
1	0	0	A valid program area	A valid general 2 or general 1 area	Never asserted
1	0	1	A valid program or general 2 area	Never asserted	A valid general 1 area
1	1	0	A valid program or general 1 area	A valid general 2 area	Never asserted
1	1	1	A valid program or general 1 or 2 area	Never asserted	Never asserted

11.4.5 Clock Stretching

Chip select and bus control signals are synchronized with the external E clock. To accommodate devices that are slower than the MCU, the E clock can be stretched when a chip select is asserted so that it remains high for one to three extra bus cycles. During this stretch, which can occur only during accesses to addresses in that chip select's address range, the other clocks continue running normally, maintaining the integrity of the timers and baud generators. Each chip select has two associated bits in the chip-select clock stretch (CSCSTR) register that set its clock stretching from zero (disabled) to three cycles.

Address: \$005A

	Bit 7	6	5	4	3	2	1	Bit 0
Read:	IOSA	IOSB	GP1SA	GP1SB	GP2SA	GP2SB	PCSA	PCSB
Write:								
Reset:	0	0	0	0	0	0	0	0

Figure 11-14. Chip Select Clock Stretch Register (CSCSTR)

IOS[A:B] — CSIO Stretch Select Bits

GP1S[A:B] — CSGP1 Stretch Select Bits

GP2S[A:B] — CSGP2 Stretch Select Bits

PCS[A:B] — CSPROG Stretch Select Bits

Each of these pairs of bits contain the binary number of cycles of clock stretch, as shown in **Table 11-9**.

Table 11-9. CSCSTR Bits Versus Clock Cycles

Bit [A:B]	Clock Stretch
0 0	None
0 1	1 cycle
1 0	2 cycles
1 1	3 cycles

11.5 Memory Expansion Examples

The first example, shown in **Figure 11-15** contains a system with 64 Kbytes of external memory to be accessed through a single 8-Kbyte window. To access eight Kbytes, or 2^{13} address locations, the CPU will need 13 address lines, ADDR[12:0]. The number of memory banks needed is the total memory, 64 Kbytes divided by the window size, eight Kbytes. This yields eight memory banks, or 2^3 . Thus, three expansion lines are required, so expansion address lines XA[15:13] replace CPU address lines ADDR[15:13]. **Figure 1-1** shows a memory map and schematic drawing of this system.

Memory Expansion and Chip Selects

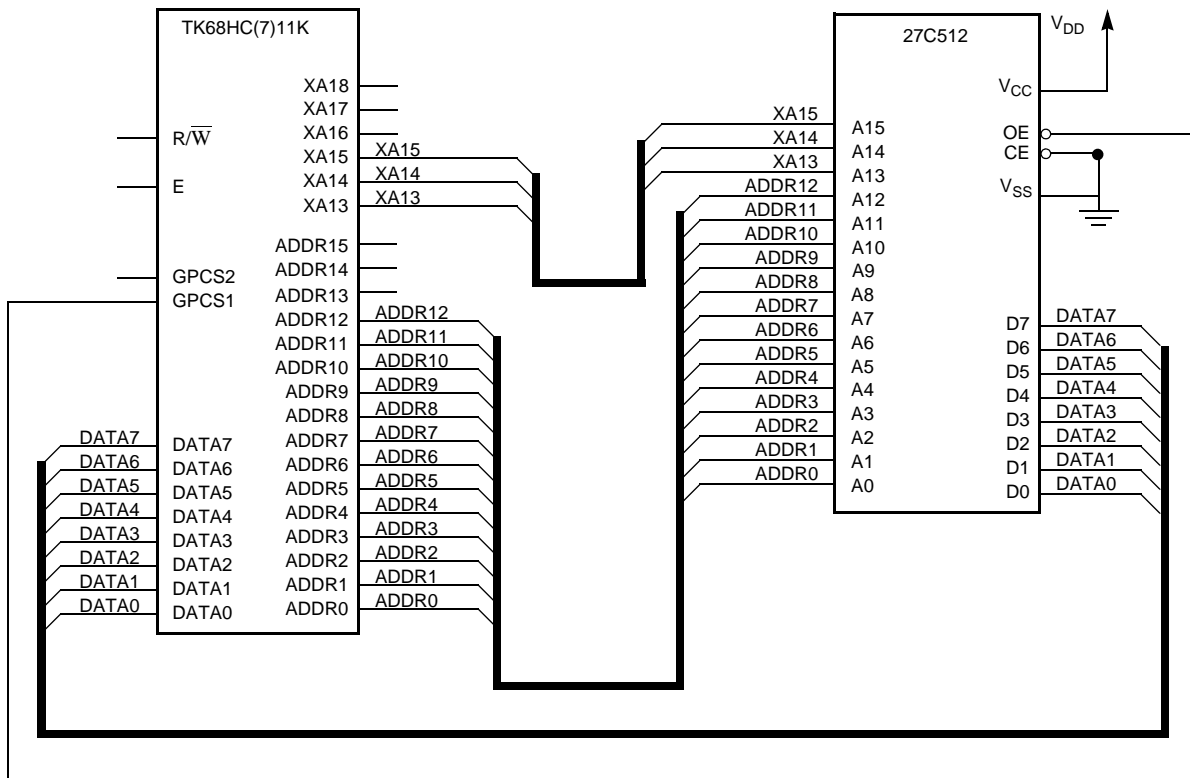
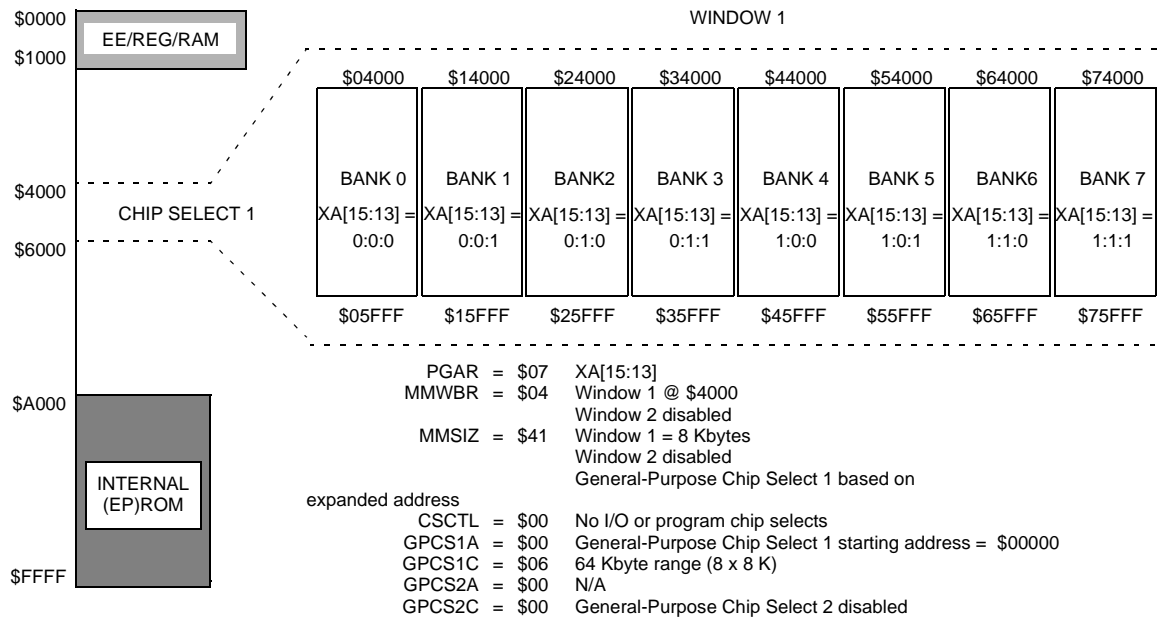
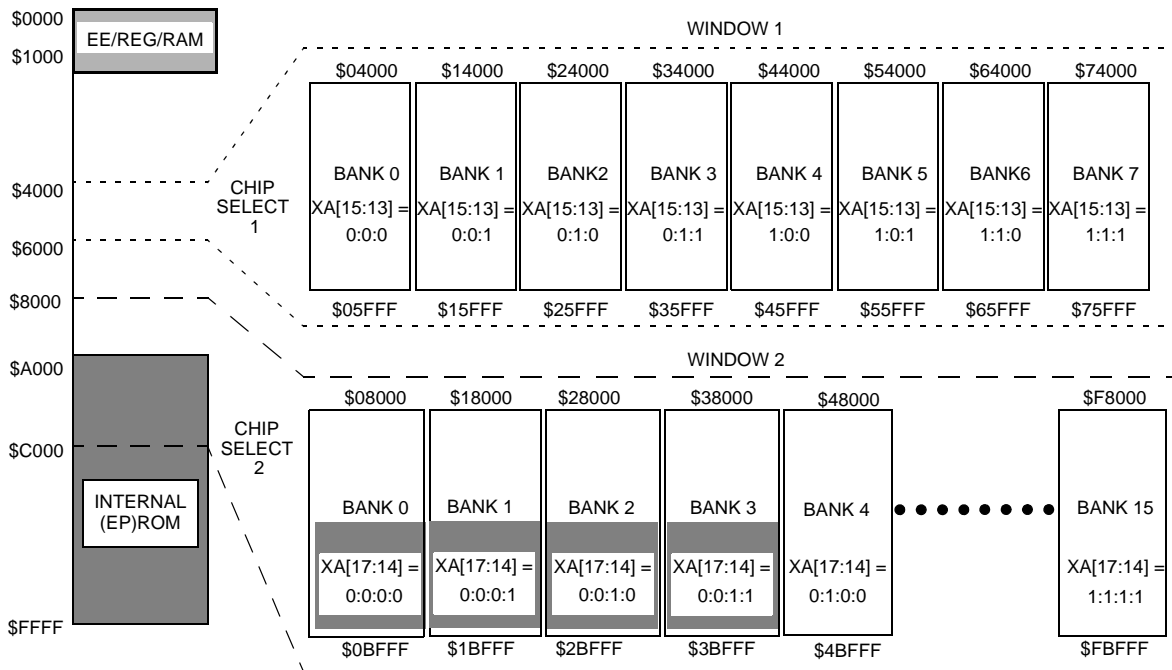


Figure 11-15. Memory Expansion Example 1 — Memory Map for a Single 8-Kbyte Window with Eight Banks of External Memory

The second example system shown in **Figure 11-16** contains two memory windows. The first window is organized as in the previous example, 8 banks of 8 Kbytes each. The second window accesses 256 Kbytes of memory in 16 banks of 16 Kbyte each. To access 16 Kbytes, or 2^{14} address locations, the CPU will need 14 address lines, ADDR[13:0]. Since ADDR13 is driven on XA13 in this example, XA13 replaces ADDR13 to drive the A13 line in the 6226 devices, but ADDR13 could be used as well. 16 (2^4) memory banks require four expansion address lines, A[17:14]. Refer to **Figure 11-16** for a memory map and schematic drawing of this system.



PGAR = \$1F	XA[17:13]	CCTL = \$00	No I/O or program chip selects
MMWBR = \$84	Window 1 @ \$4000	GPCS1A = \$00	General-purpose chip select 1 from \$00000
	Window 2 @ \$8000	GPCS1C = \$06	64 KByte range (8 x 8 K)
MMSIZ = \$E1	Window 1 = 8 Kbytes	GPCS2A = \$00	General-purpose chip select 2 from \$00000
	Window 2 = 16 Kbytes	GPCS2C = \$08	256 KByte range (16 x 16 K)
	GPCS1, GPCS2 based on expansion address		

Figure 11-16. Memory Expansion Example 2 (Sheet 1 of 2)
Memory Map for One 8-Kbyte Window with Eight Banks and
One 16-Kbyte Window with 16 Banks of External Memory

Memory Expansion and Chip Selects

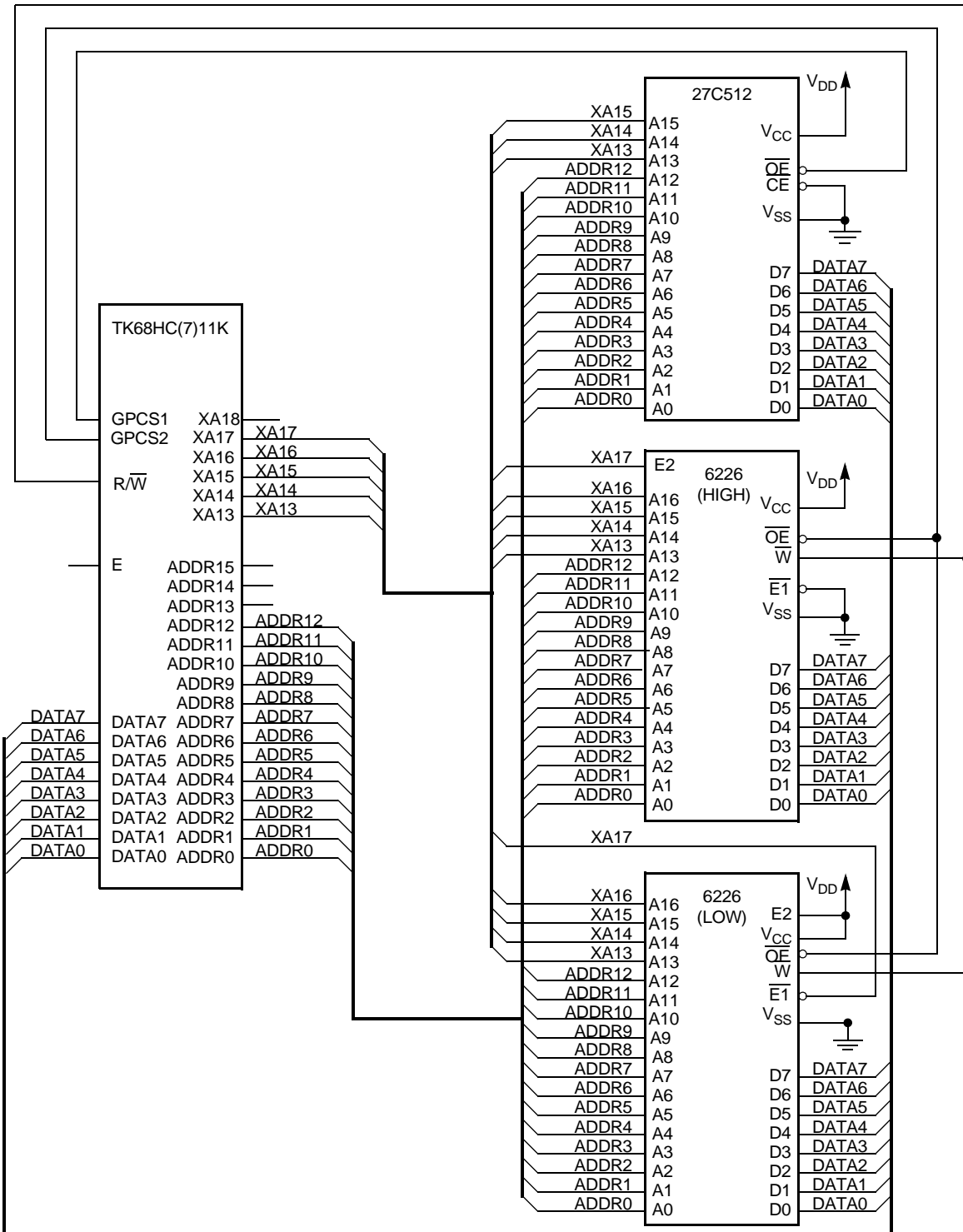


Figure 11-16. Memory Expansion Example 2 (Sheet 2 of 2)
Memory Map for One 8-Kbyte Window with Eight Banks and
One 16-Kbyte Window with 16 Banks of External Memory

Section 12. Electrical Characteristics

12.1 Contents

12.2	Introduction	252
12.3	Maximum Ratings for Standard and Extended Voltage Devices	252
12.4	Functional Operating Range	253
12.5	Functional Operating Range, Extended Voltage Devices . . .	253
12.6	Thermal Characteristics	254
12.7	Thermal Characteristics, Extended Voltage Devices	254
12.8	5.0-Volt Electrical Characteristics	255
12.9	5.0-Volt Power Dissipation Characteristics	256
12.10	Extended Voltage Electrical Characteristics	257
12.11	Extended Voltage Power Dissipation Characteristics	258
12.12	Control Timing	260
12.13	Peripheral Port Timing	264
12.14	Analog-to-Digital Converter Characteristics	266
12.15	Expansion Bus Timing	268
12.16	Expansion Bus Timing, Extended Voltage Devices	269
12.17	Serial Peripheral Interface Timing	271
12.18	Serial Peripheral Interface Timing, Extended Voltage Devices	272
12.19	EEPROM Characteristics	275
12.20	EEPROM Characteristics, Extended Voltage Devices	275

12.2 Introduction

This section contains electrical parameters for standard and extended voltage devices. When applicable, extended voltage parameters are shown separately. Diagrams apply to both standard and extended voltage devices.

12.3 Maximum Ratings for Standard and Extended Voltage Devices

Maximum ratings are the extreme limits to which the MCU can be exposed without permanently damaging it.

NOTE: *This device is not guaranteed to operate properly at the maximum ratings. Refer to **12.8 5.0-Volt Electrical Characteristics** and **12.10 Extended Voltage Electrical Characteristics** for guaranteed operating conditions.*

Rating	Symbol	Value	Unit
Supply voltage	V_{DD}	-0.3 to +7.0	V
Input voltage	V_{In}	-0.3 to +7.0	V
Current drain per pin ⁽¹⁾ excluding V_{DD} , V_{SS} , AV_{DD} , V_{RH} , and V_{RL}	I_D	25	mA
Storage temperature	T_{STG}	-55 to +150	°C

1. One pin at a time, observing maximum power dissipation limits

NOTE: *This device contains circuitry to protect the inputs against damage due to high static voltages or electric fields; however, it is advised that normal precautions be taken to avoid application of any voltage higher than maximum-rated voltages to this high-impedance circuit. For proper operation, it is recommended that V_{In} and V_{Out} be constrained to the range $V_{SS} \leq (V_{In} \text{ or } V_{Out}) \leq V_{DD}$. Reliability of operation is enhanced if unused inputs are connected to an appropriate logic voltage level (for example, either V_{SS} or V_{DD}).*

12.4 Functional Operating Range

Rating	Symbol	Value	Unit
Operating temperature range TK68HC(7)11KS TK68HC(7)11KV TK68HC(7)11 KM	T_A	T_L to T_H -40 to +85 -40 to +105 -40 to +125	°C
Operating voltage range	V_{DD}	$5.0 \pm 10\%$	V

12.5 Functional Operating Range, Extended Voltage Devices

Rating	Symbol	Value	Unit
Operating temperature range TK68L11KS	T_A	T_L to T_H -20 to +70	°C
Operating voltage range	V_{DD}	3.0 to 5.5	V

12.6 Thermal Characteristics

Characteristic	Symbol	Value	Unit
Average junction temperature	T_J	$T_A + (P_D \times \Theta_{JA})$	$^{\circ}\text{C}$
Ambient temperature	T_A	User-determined	$^{\circ}\text{C}$
Package thermal resistance (junction-to-ambient) 80-pin thin quad flat pack 68-pin plastic leaded chip carrier 68-pin windowed ceramic cerquad (EPROM)	Θ_{JA}	80 50 60	$^{\circ}\text{C}/\text{W}$
Total power dissipation ⁽¹⁾	P_D	$\frac{P_{INT} + P_{I/O}}{K / T_J + 273^{\circ}\text{C}}$	W
Device internal power dissipation	P_{INT}	$I_{DD} \times V_{DD}$	W
I/O pin power dissipation ⁽²⁾	$P_{I/O}$	User-determined	W
A constant ⁽³⁾	K	$P_D \times (T_A + 273^{\circ}\text{C}) + \Theta_{JA} \times P_D^2$	$\text{W}/^{\circ}\text{C}$

1. This is an approximate value, neglecting P_{I/O}.
2. For most applications, P_{I/O} ≤ P_{INT} and can be neglected.
3. K is a constant pertaining to the device. Solve for K with a known T_A and a measured P_D (at equilibrium). Use this value of K to solve for P_D and T_J iteratively for any value of T_A.

12.7 Thermal Characteristics, Extended Voltage Devices

Characteristic	Symbol	Value	Unit
Average junction temperature	T_J	$T_A + (P_D \times \Theta_{JA})$	$^{\circ}\text{C}$
Ambient temperature	T_A	User-determined	$^{\circ}\text{C}$
Package thermal resistance (junction-to-ambient) 84-pin plastic leaded chip carrier 80-pin thin quad flat pack	Θ_{JA}	50 80	$^{\circ}\text{C}/\text{W}$
Total power dissipation ⁽¹⁾	P_D	$\frac{P_{INT} + P_{I/O}}{K / T_J + 273^{\circ}\text{C}}$	W
Device internal power dissipation	P_{INT}	$I_{DD} \times V_{DD}$	W
I/O pin power dissipation ⁽²⁾	$P_{I/O}$	User-determined	W
A constant ⁽³⁾	K	$P_D \times (T_A + 273^{\circ}\text{C}) + \Theta_{JA} \times P_D^2$	$\text{W}/^{\circ}\text{C}$

1. This is an approximate value, neglecting P_{I/O}.
2. For most applications, P_{I/O} ≤ P_{INT} and can be neglected.
3. K is a constant pertaining to the device. Solve for K with a known T_A and a measured P_D (at equilibrium). Use this value of K to solve for P_D and T_J iteratively for any value of T_A.

12.8 5.0-Volt Electrical Characteristics

Characteristic ⁽¹⁾	Symbol	Min	Max	Unit
Output voltage, $I_{Load} = \pm 10.0 \mu A^{(2)}$ All outputs except XTAL All outputs except XTAL, \overline{RESET} , and MODA	V_{OL} V_{OH}	— $V_{DD} - 0.1$	0.1 —	V
Output high voltage, $I_{Load} = -0.8 \text{ mA}$, $V_{DD} = 4.5 \text{ V}^{(2)}$ All outputs except XTAL, \overline{RESET} , and MODA	V_{OH}	$V_{DD} - 0.8$	—	V
Output low voltage, $I_{Load} = 1.6 \text{ mA}$ All outputs except XTAL	V_{OL}	—	0.4	V
Input high voltage All inputs except \overline{RESET} \overline{RESET}	V_{IH}	$0.7 \times V_{DD}$ $0.8 \times V_{DD}$	$V_{DD} + 0.3$ $V_{DD} + 0.3$	V
Input low voltage All Inputs	V_{IL}	$V_{SS} - 0.3$	$0.2 \times V_{DD}$	V
I/O ports, three-state leakage, $V_{in} = V_{IH}$ or V_{IL} Ports A, B, C, D, F, G, H, MODA/ \overline{LIR} , \overline{RESET}	I_{OZ}	—	± 10	μA
Input leakage current, $V_{in} = V_{DD}$ or $V_{SS}^{(3)}$ \overline{IRQ} , \overline{XIRQ} on standard devices MODB/ V_{STBY} , \overline{XIRQ} on EPROM devices	I_{in}	— —	± 1 ± 10	μA
Input current with pullup resistors, $V_{in} = V_{IL}$ Ports B, F, G, and H	I_{IPR}	100	500	μA
RAM standby voltage, power down	V_{SB}	2.0	V_{DD}	V
RAM standby current, power down	I_{SB}	—	10	μA
Input capacitance PE[7:0], \overline{IRQ} , \overline{XIRQ} , EXTAL Ports A, B, C, D, F, G, H, MODA/ \overline{LIR} , \overline{RESET}	C_{in}	— —	8 12	pF
Output load capacitance All outputs except PD[4:1], XOUT, XTAL, MODA/ \overline{LIR} PD[4:1] XOUT	C_L	— — —	90 200 30	pF

1. $V_{DD} = 5.0 \text{ Vdc} \pm 10\%$, $V_{SS} = 0 \text{ Vdc}$, $T_A = T_L$ to T_H , unless otherwise noted

2. V_{OH} specification for \overline{RESET} and MODA is not applicable because they are open-drain pins. V_{OH} specification not applicable to ports C and D in wired-OR mode.

3. Refer to **12.14 Analog-to-Digital Converter Characteristics** for leakage current for port E.

12.9 5.0-Volt Power Dissipation Characteristics

Characteristic	Symbol	2 MHz	4 MHz	Unit
Maximum total supply current ⁽¹⁾ RUN: Single-chip mode Expanded mode Slow mode	I_{DD}	27 35 5.85	40 50 6.85	mA
WAIT: (All peripheral functions shut down) Single-chip mode Expanded mode Slow mode	WI_{DD}	10 12 3.5	20 22 4.5	mA
STOP: (No clocks) Single-chip mode	SI_{DD}	50	50	μ A
Maximum power dissipation Single-chip mode Expanded mode	P_D	149 193	220 275	mW

1. EXTAL is driven with a square wave; $t_{cyc} = 500$ ns for 2 MHz rating; $t_{cyc} = 250$ ns for 4 MHz rating;
 $V_{IL} \leq 0.2$ V; $V_{IH} \geq V_{DD} - 0.2$ V; no dc loads

12.10 Extended Voltage Electrical Characteristics

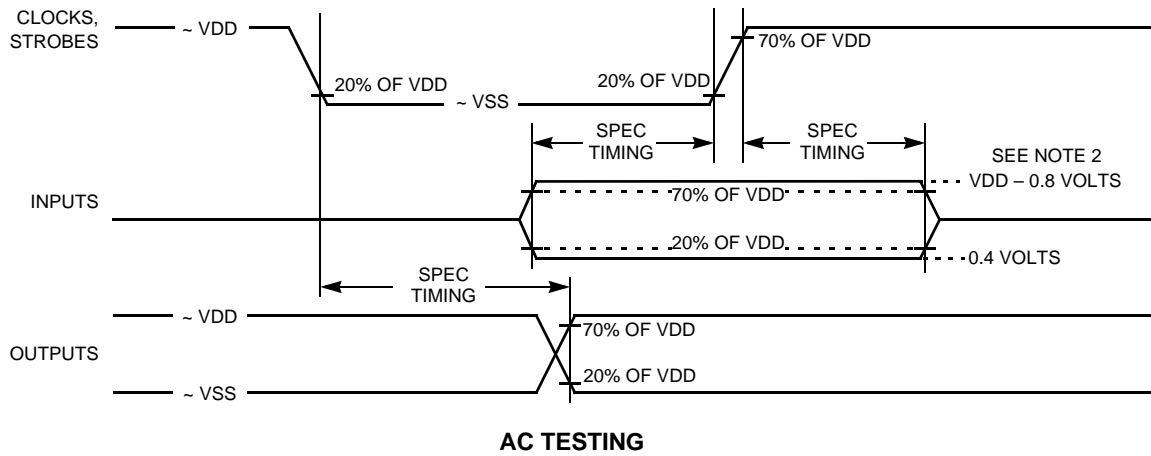
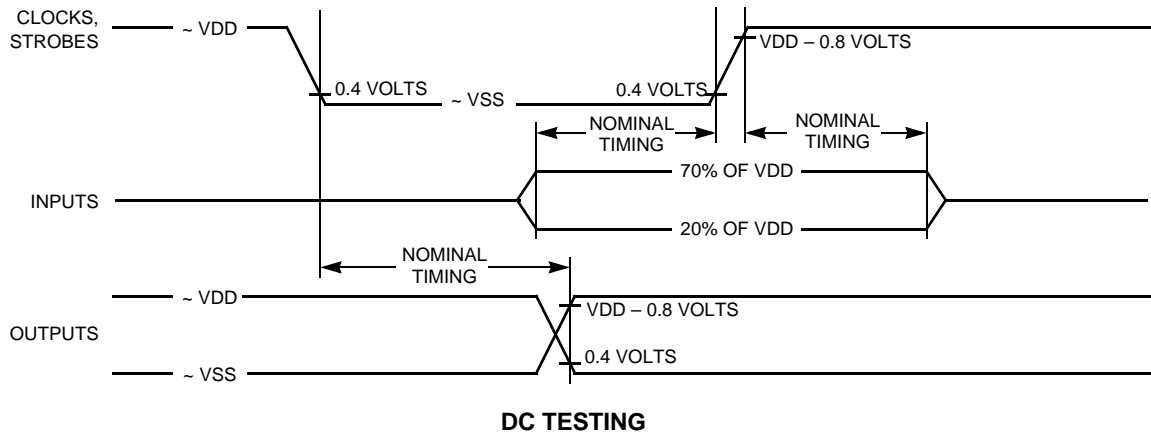
Characteristic ⁽¹⁾	Symbol	Min	Max	Unit
Output voltage, $I_{Load} = 10.0 \mu A$ ⁽²⁾ All outputs except XTAL All outputs except XTAL, \overline{RESET} , and MODA	V_{OL} V_{OH}	— $V_{DD} - 0.1$	0.1 —	V
Output high voltage, $I_{Load} = -0.5 \text{ mA}$, $V_{DD} = 3.0 \text{ V}$ ¹ $I_{Load} = -0.8 \text{ mA}$, $V_{DD} = 4.5 \text{ V}$ All outputs except XTAL, \overline{RESET} , and MODA	V_{OH}	$V_{DD} - 0.8$	—	V
Output low voltage, $I_{Load} = 1.6 \text{ mA}$, $V_{DD} = 5.0 \text{ V}$ $I_{Load} = 1.0 \text{ mA}$, $V_{DD} = 3.0 \text{ V}$ All outputs except XTAL	V_{OL}	—	0.4	V
Input high voltage All inputs except \overline{RESET} \overline{RESET}	V_{IH}	$0.7 \times V_{DD}$ $0.8 \times V_{DD}$	$V_{DD} + 0.3$ $V_{DD} + 0.3$	V
Input low voltage, all inputs	V_{IL}	$V_{SS} - 0.3$	$0.2 \times V_{DD}$	V
Three-state leakage, I/O ports, $V_{in} = V_{IH}$ or V_{IL} Ports A, B, C, D, F, G, H, MODA/ \overline{LIR} , \overline{RESET}	I_{OZ}	—	± 10	μA
Input leakage current, $V_{in} = V_{DD}$ or V_{SS} ⁽³⁾ \overline{IRQ} , \overline{XIRQ} MODB/ V_{STBY} , \overline{XIRQ} on EPROM devices	I_{In}	— —	± 1 ± 10	μA
Input current with pullup resistors, $V_{in} = V_{IL}$ Ports B, F, G, and H	I_{ipr}	25	500	μA
RAM standby voltage, ppower down	V_{SB}	2.0	V_{DD}	V
RAM standby current, power down	I_{SB}	—	10	μA
Input capacitance PE[7:0], \overline{IRQ} , \overline{XIRQ} , EXTAL Ports A, B, C, D, F, G, H, MODA/ \overline{LIR} , \overline{RESET}	C_{In}	— —	8 12	pF
Output load capacitance All outputs except PD[4:1], XOUT, XTAL, MODA/ \overline{LIR} PD[4:1] XOUT	C_L	— — —	90 100 30	pF

- $V_{DD} = 3.0 \text{ Vdc}$ to 5.5 Vdc , $V_{SS} = 0 \text{ Vdc}$, $T_A = T_L$ to T_H , unless otherwise noted
- V_{OH} specification for \overline{RESET} and MODA is not applicable because they are open-drain pins. V_{OH} specification is not applicable to ports C and D in wired-OR mode.
- Refer to **12.14 Analog-to-Digital Converter Characteristics** for leakage current for port E.

12.11 Extended Voltage Power Dissipation Characteristics

Characteristic	Symbol	1 MHz	3 MHz	Unit
Maximum total supply current⁽¹⁾ RUN: Single-chip mode $V_{DD} = 5.5\text{ V}$ $V_{DD} = 3.0\text{ V}$ Expanded mode $V_{DD} = 5.5\text{ V}$ $V_{DD} = 3.0\text{ V}$	I_{DD}	15 6	32 12	mA
WAIT: All peripheral functions shut down Single-chip mode $V_{DD} = 5.5\text{ V}$ $V_{DD} = 3.0\text{ V}$ Expanded mode $V_{DD} = 5.5\text{ V}$ $V_{DD} = 3.0\text{ V}$	$W_{I_{DD}}$	8 3.5	15 7	mA
STOP: no clocks Single-chip mode $V_{DD} = 5.5\text{ V}$ $V_{DD} = 3.0\text{ V}$	$S_{I_{DD}}$	50 25	50 25	μA
Maximum power dissipation Single-chip mode $V_{DD} = 5.5\text{ V}$ $V_{DD} = 3.0\text{ V}$ Expanded mode $V_{DD} = 5.5\text{ V}$ $V_{DD} = 3.0\text{ V}$	P_D	83 18	176 36	mW
		110 24	231 48	

1. EXTAL is driven with a square wave; $t_{cyc} = 1000\text{ ns}$ for 1 MHz rating; $t_{cyc} = 333\text{ ns}$ for 3 MHz rating;
 $V_{IL} \leq 0.2\text{ V}$; $V_{IH} \geq V_{DD} - 0.2\text{ V}$; no dc loads



Notes:

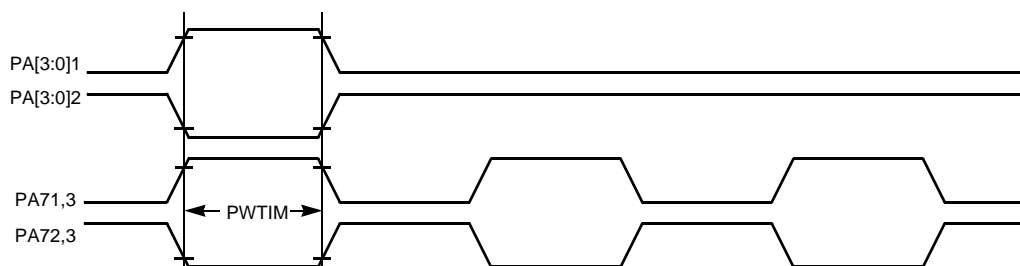
1. Full test loads are applied during all DC electrical tests and AC timing measurements.
2. During AC timing measurements, inputs are driven to 0.4 volts and $V_{DD} - 0.8$ volts while timing measurements are taken at the 20% and 70% of V_{DD} points.

Figure 12-1. Test Methods

12.12 Control Timing

Characteristic ⁽¹⁾	Symbol	1.0 MHz		2.0 MHz		3.0 MHz		4.0 MHz ⁽²⁾		Unit
		Min	Max	Min	Max	Min	Max	Min	Max	
Frequency of operation	f_o	dc	1.0	dc	2.0	dc	3.0	dc	4.0	MHz
E-clock period	t_{cyc}	1000	—	500	—	333	—	250	—	ns
Crystal frequency	f_{XTAL}	—	4.0	—	8.0	—	12.0	—	16.0	MHz
External oscillator frequency	$4 f_o$	dc	4.0	dc	8.0	dc	12.0	dc	16.0	MHz
Processor control setup time $t_{PCSU} = 1/4 t_{cyc} + 50 \text{ ns}$ $t_{PCSU} = 1/4 t_{cyc} + 75 \text{ ns}$ (extended voltage devices)	t_{PCSU}	300	—	175	—	133	—	112	—	ns
Reset input pulse width ⁽³⁾ To guarantee external reset vector Minimum input time ⁽⁴⁾	PW_{RSTL}	16 1	— —	16 1	— —	16 1	— —	16 1	— —	t_{cyc}
Mode programming setup time	t_{MPS}	2	—	2	—	2	—	2	—	t_{cyc}
Mode programming hold time	t_{MPH}	10	—	10	—	10	—	10	—	ns
Interrupt pulse width, IRQ edge-sensitive mode $PW_{IRQ} = t_{cyc} + 20 \text{ ns}$	PW_{IRQ}	1020	—	520	—	353	—	270	—	ns
Wait recovery startup time	t_{WRS}	—	4	—	4	—	4	—	4	t_{cyc}
Timer pulse width $PW_{TIM} = t_{cyc} + 20 \text{ ns}$ Input capture, pulse accumulator	PW_{TIM}	1020	—	520	—	353	—	270	—	ns

- $V_{DD} = 4.5$ to 5.5 Vdc for standard devices, $V_{DD} = 3.0$ to 5.5 Vdc for extended voltage devices, $V_{SS} = 0 \text{ Vdc}$, $T_A = T_L$ to T_H . All timing measurements refer to 20% V_{DD} and 70% V_{DD} , unless otherwise noted.
- Extended voltages device operation is not guaranteed above 3.0 MHz.
- 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 two cycles later to determine the source of the interrupt.
- Can be pre-empted by internal reset



Notes:

- Rising edge sensitive input
- Falling edge sensitive input
- Maximum pulse accumulator clocking rate is E-clock frequency divided by 2.

Figure 12-2. Timer Inputs

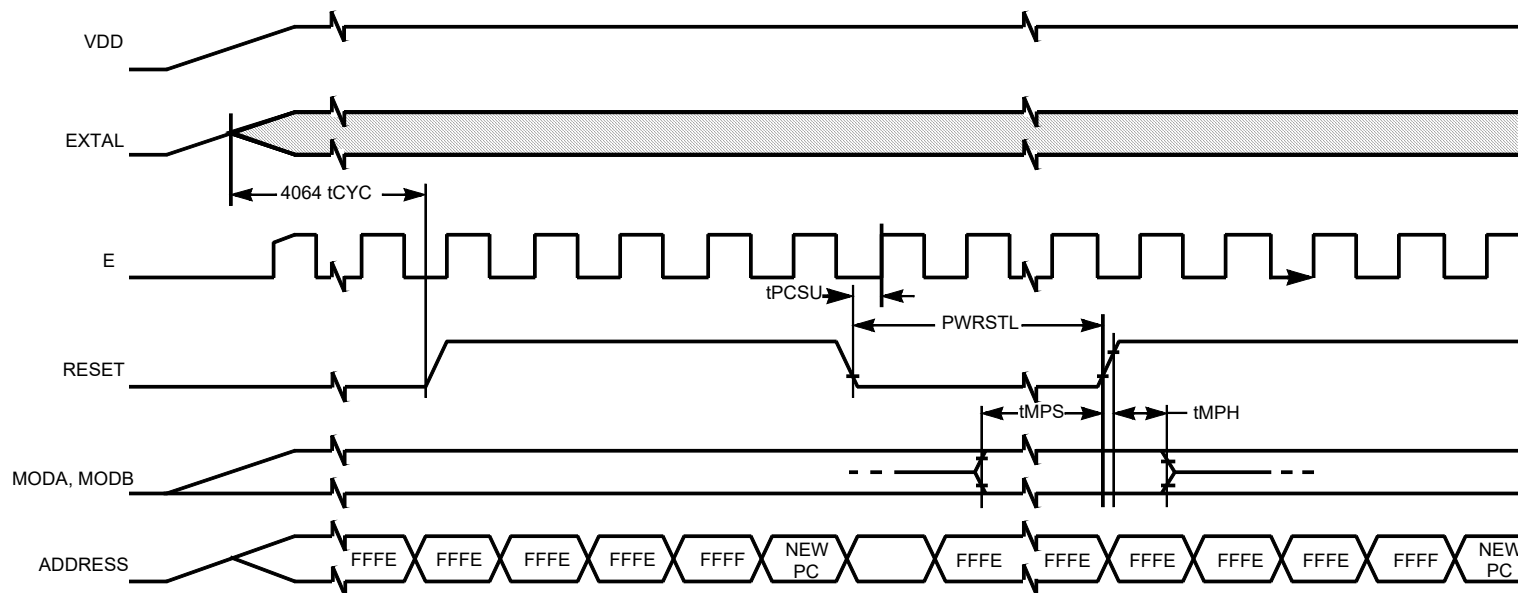
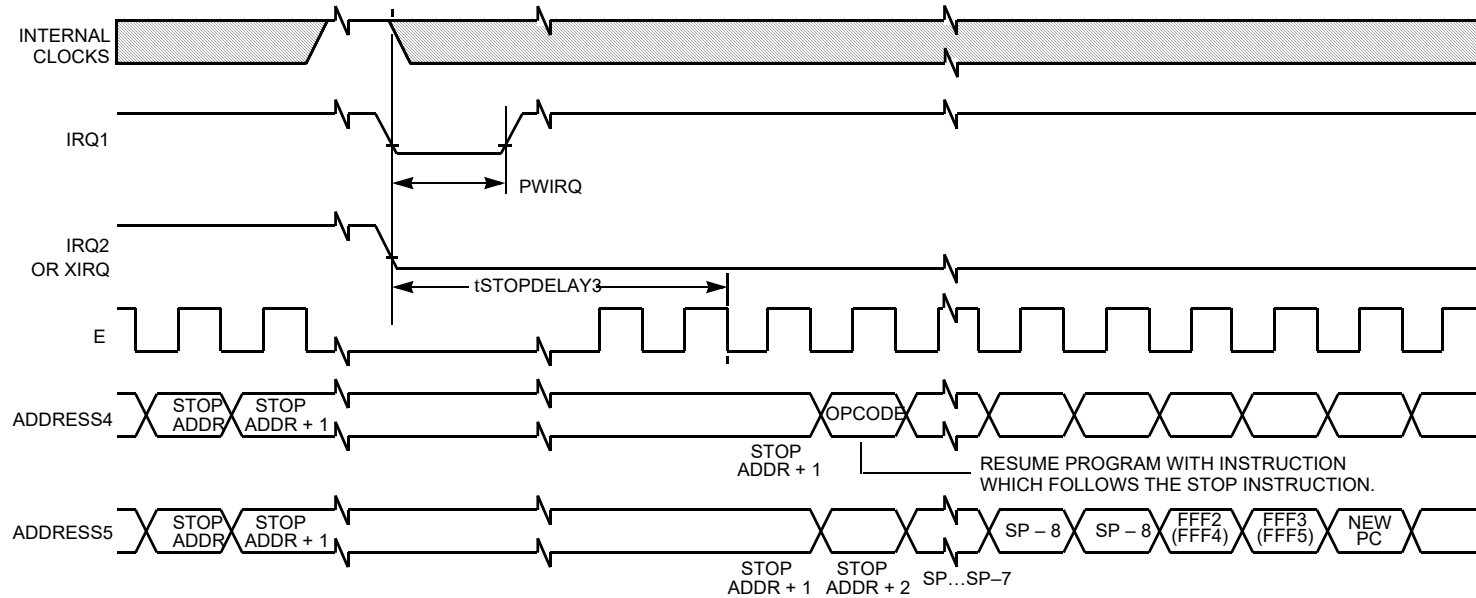


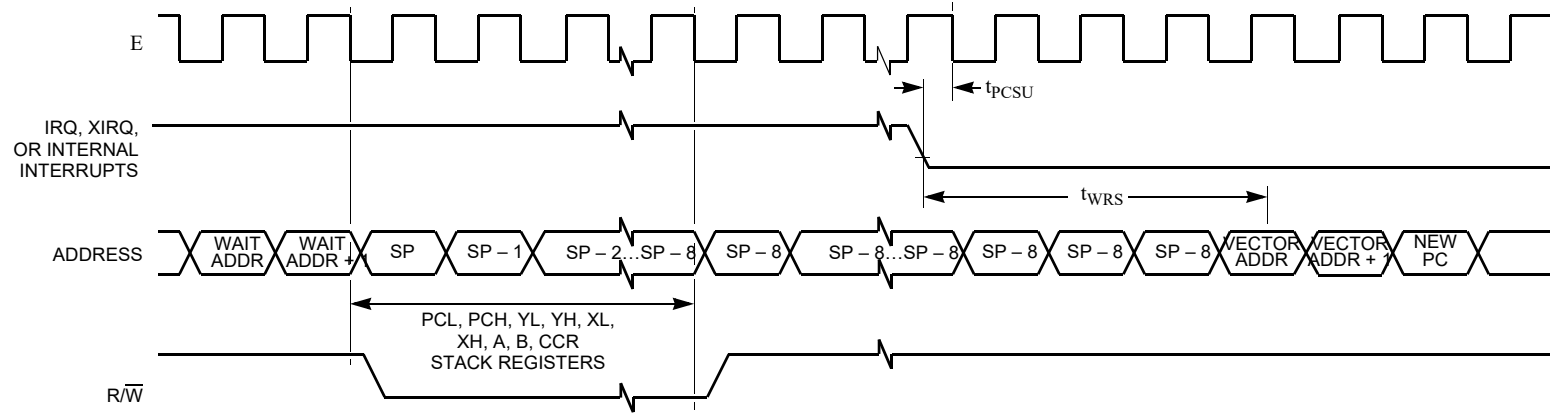
Figure 12-3. POR External Reset Timing Diagram



Notes:

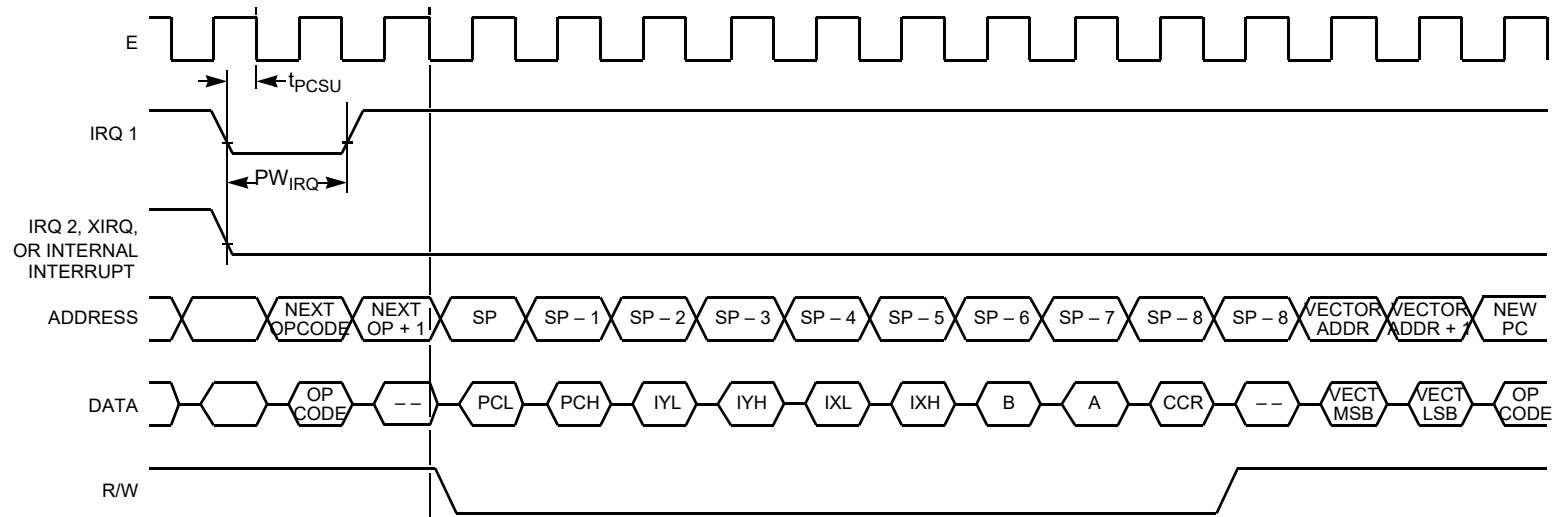
1. Edge-sensitive \overline{IRQ} pin (IRQE bit = 1)
2. Level-sensitive \overline{IRQ} pin (IRQE bit = 0)
3. $t_{STOPDELAY} = 4064 t_{cyc}$ if DLY bit = 1 or $4 t_{cyc}$ if DLY = 0
4. \overline{XIRQ} with X bit in CCR = 1
5. \overline{IRQ} or \overline{XIRQ} with X bit in CCR = 0

Figure 12-4. STOP Recovery Timing Diagram



Note: $\overline{\text{RESET}}$ also causes recovery from WAIT.

Figure 12-5. WAIT Recovery from Interrupt Timing Diagram



Notes:

1. Edge-sensitive $\overline{\text{IRQ}}$ pin (IRQE bit = 1)
2. Level-sensitive IRQ pin (IRQE bit = 0)

Figure 12-6. Interrupt Timing Diagram

12.13 Peripheral Port Timing

Characteristic ⁽¹⁾	Symbol	1.0 MHz		2.0 MHz		3.0 MHz		4.0 MHz ⁽²⁾		Unit
		Min	Max	Min	Max	Min	Max	Min	Max	
Frequency of operation (E clock)	f_o	dc	1.0	dc	2.0	dc	3.0	dc	4.0	MHz
E-clock period	t_{cyc}	1000	—	500	—	333	—	250	—	ns
Peripheral data setup time ⁽³⁾ MCU read of ports A, B, C, D, E, F, G, and H	t_{PDSU}	100	—	100	—	100	—	100	—	ns
Peripheral data hold time ² MCU read of ports A, B, C, D, E, F, G, and H	t_{PDH}	50	—	50	—	50	—	50	—	ns
Delay time, peripheral data write ² Standard devices MCU write to port A, B, G, and H MCU write to ports C, D, and F ($t_{PWD} = 1/4 t_{cyc} + 100$ ns) Extended voltage MCU write to port A, B, G, and H MCU write to ports C, D, and F ($t_{PWD} = 1/4 t_{cyc} + 150$ ns)	t_{PWD}	—	200	—	200	—	200	—	200	ns
		—	350	—	225	—	183	—	162	
		—	250	—	250	—	250	—	—	
		—	400	—	225	—	233	—	—	

- $V_{DD} = 4.5$ to 5.5 Vdc for standard devices, $V_{DD} = 3.0$ to 5.5 Vdc for extended voltage devices, $V_{SS} = 0$ Vdc, $T_A = T_L$ to T_H
All timing measurements refer to 20% V_{DD} and 70% V_{DD} , unless otherwise noted.
- Extended voltage device operation is not guaranteed above 3.0 MHz.
- Ports C and D timing is valid only in active drive mode. (CWOM and DWOM bits are cleared in OPT2 and SPCR registers, respectively.)

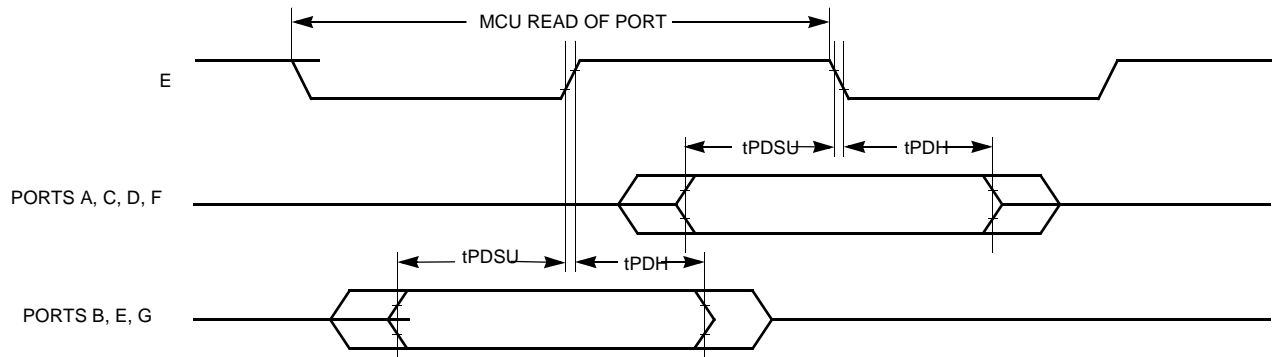


Figure 12-7. Port Read Timing Diagram

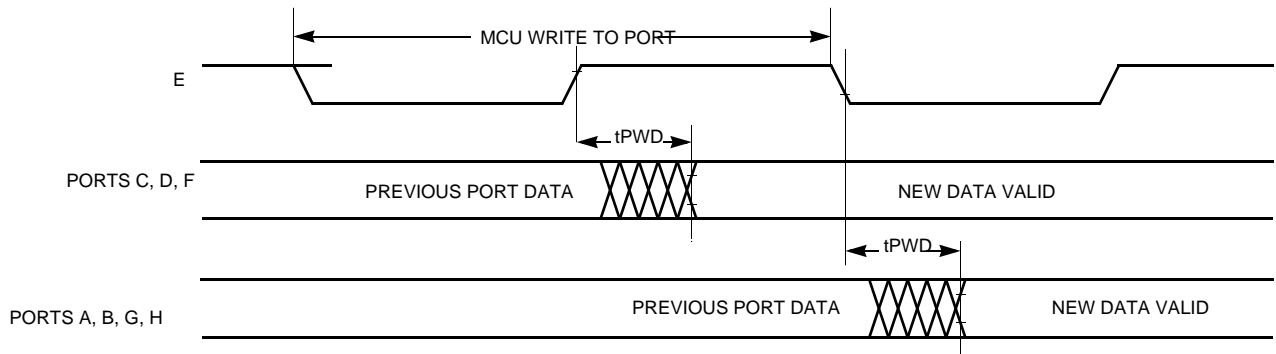


Figure 12-8. Port Write Timing Diagram

12.14 Analog-to-Digital Converter Characteristics

Characteristic ⁽¹⁾	Parameter	Min	Absolute	Max		Unit
				$f_o \leq 2.0$ MHz	$f_o > 2.0$ MHz ⁽²⁾	
Resolution	Number of bits resolved by A/D converter	—	8	—	—	Bits
Non-linearity	Maximum deviation from the ideal A/D transfer characteristics ⁽³⁾	—	—	$\pm 1/2$	± 1	LSB
Zero error	Difference between the output of an ideal and an actual for zero input voltage ⁽³⁾	—	—	$\pm 1/2$	± 1	LSB
Full scale error	Difference between the output of an ideal and an actual A/D for full-scale input voltage ⁽³⁾	—	—	$\pm 1/2$	± 1	LSB
Total unadjusted error	Maximum sum of non-linearity, zero error, and full-scale error ⁽³⁾	—	—	$\pm 1/2$	$\pm 1 1/2$	LSB
Quantization error	Uncertainty because of converter resolution	—	—	$\pm 1/2$	$\pm 1/2$	LSB
Absolute accuracy	Difference between the actual input voltage and the full-scale weighted equivalent of the binary output code, all error sources included ⁽³⁾	—	—	± 1	± 2	LSB
Conversion range	Analog input voltage range	V_{RL}	—	V_{RH}	V_{RH}	V
V_{RH}	Maximum analog reference voltage ⁽⁴⁾	V_{RL}	—	$V_{DD} + 0.1$	$V_{DD} + 0.1$	V
V_{RL}	Minimum analog reference voltage ⁽⁴⁾	$V_{SS} - 0.1$	—	V_{RH}	V_{RH}	V
ΔV_R	Minimum difference between V_{RH} and V_{RL} ⁽⁴⁾	3	—	—	—	V
Conversion time	Total time to perform a single analog-to-digital conversion: E clock Internal RC oscillator	— —	32 —	— $t_{cyc} + 32$	— $t_{cyc} + 32$	t_{cyc} μs
Monotonicity	Conversion result never decreases with an increase in input voltage and has no missing codes			Guaranteed		
Zero Input reading	Conversion result when $V_{in} = V_{RL}$	00	—	—	—	Hex
Full Scale reading	Conversion result when $V_{in} = V_{RH}$	—	—	FF	FF	Hex

Continued

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

1. $V_{DD} = 4.5$ to 5.5 Vdc for standard devices; $V_{DD} = 3.0$ to 5.5 Vdc for extended voltage devices; $V_{SS} = 0$ Vdc, $T_A = T_L$ to T_H
Source impedances greater than $10\text{ k}\Omega$ affect accuracy adversely because of input leakage.
All timing measurements refer to 20% V_{DD} and 70% V_{DD} , unless otherwise noted.
2. Up to 4.0 MHz for standard devices, 3.0 MHz for extended voltage devices.
3. For extended voltage devices, greater accuracy below 2.0 MHz is not guaranteed.
4. Performance is verified down to $2.5\text{ V } \Delta V_R$, but accuracy is tested and guaranteed at $\Delta V_R = 5\text{ V } \pm 10\%$.

12.15 Expansion Bus Timing

Num	Characteristic ⁽¹⁾	Symbol	2.0 MHz		3.0 MHz		4.0 MHz		Unit
			Min	Max	Min	Max	Min	Max	
	Frequency of operation (E clock) ⁽²⁾	f_o	dc	2.0	dc	3.0	dc	4.0	MHz
1	Cycle time, $t_{cyc} = 1/f_o$	t_{cyc}	500	—	333	—	250	—	ns
2	Pulse width, E low, $PW_{EL} = 1/2 t_{cyc} - 20$ ns	PW_{EL}	230	—	147	—	105	—	ns
3	Pulse width, E highs ⁽³⁾ $PW_{EH} = 1/2 t_{cyc} - 25$ n	PW_{EH}	225	—	142	—	100	—	ns
4A	E clock Rise time	t_r	—	20	—	20	—	20	ns
4B	Fall time	t_f	—	20	—	18	—	15	ns
9	Address hold time, $t_{AH} = 1/8 t_{cyc} - 10$ ns	t_{AH}	53	—	32	—	21	—	ns
11	Address Delay time, $t_{AD} = 1/8 t_{cyc} + 40$ ns	t_{AD}	—	103	—	82	—	71	ns
12	Address valid time to E rise $t_{AV} = PW_{EL} - t_{AD}$	t_{AV}	128	—	65	—	34	—	ns
17	Read data setup time	t_{DSR}	30	—	30	—	20	—	ns
18	Read data hold time	t_{DHR}	0	—	0	—	0	—	ns
19	Write data delay time	t_{DDW}	—	40	—	40	—	40	ns
21	Write data hold time, $t_{DHW} = 1/8 t_{cyc}$	t_{DHW}	63	—	42	—	31	—	ns
29	MPU address access time ⁽³⁾ $t_{ACCA} = t_{cyc} - t_f - t_{DSR} - t_{AD}$	t_{ACCA}	348	—	203	—	144	—	ns
39	Write data setup time ⁽³⁾ $t_{DSW} = PW_{EH} - t_{DDW}$	t_{DSW}	185	—	102	—	60	—	ns
50	E valid chip-select delay time	t_{ECSD}	—	40	—	40	—	40	ns
51	E valid chip-select access time ⁽³⁾ $t_{ECSA} = PW_{EH} - t_{ECSD} - t_{DSR}$	t_{ECSA}	155	—	72	—	40	—	ns
52	Chip select hold time	t_{CH}	0	20	0	20	0	20	ns
54	Address valid chip-select delay time $t_{ACSD} = 1/4 t_{cyc} + 40$ ns	t_{ACSD}	—	165	—	123	—	103	ns
55	Address valid chip-select access time $t_{ACSA} = t_{cyc} - t_f - t_{DSR} - t_{ACSD}$ ⁽³⁾	t_{ACSA}	285	—	162	—	113	—	ns
56	Address valid to chip-select time	t_{AVCS}	10	—	10	—	10	—	ns
57	Address valid to data three-state time	t_{AVDZ}	—	10	—	10	—	10	ns

1. $V_{DD} = 5.0 \pm 10\%$, $V_{SS} = 0$ Vdc, $T_A = T_L$ to T_H , unless otherwise noted

All timing measurements refer to 20% V_{DD} and 70% V_{DD} , unless otherwise noted.

2. Input clocks with duty cycles other than 50% affect bus performance.

3. This parameter is affected by clock stretching. Add $n(t_{cyc})$ to parameter value, where $n = 1, 2,$ or 3 depending on values written to CSCSTR register.

12.16 Expansion Bus Timing, Extended Voltage Devices

Num	Characteristic ⁽¹⁾	Symbol	1.0 MHz		3.0 MHz		Unit
			Min	Max	Min	Max	
	Frequency of operation (E clock) ⁽²⁾	f_o	dc	1.0	dc	3.0	MHz
1	Cycle time, $t_{cyc} = 1/f_o$	t_{cyc}	1000	—	333	—	ns
2	Pulse width, E low $PW_{EL} = 1/2 t_{cyc} - 20$ ns	PW_{EL}	475	—	142	—	ns
3	Pulse width, E highs ⁽³⁾ $PW_{EH} = 1/2 t_{cyc} - 25$ n	PW_{EH}	470	—	137	—	ns
4A	E clock Rise time	t_r	—	25	—	25	ns
4B	Fall time	t_f	—	20	—	20	ns
9	Address hold time, $t_{AH} = 1/8 t_{cyc} - 10$ ns	t_{AH}	105	—	22	—	ns
11	Address delay time, $t_{AD} = 1/8 t_{cyc} + 40$ ns	t_{AD}	—	185	—	102	ns
12	Address valid time to E rise, $t_{AV} = PW_{EL} - t_{AD}$	t_{AV}	290	—	40	—	ns
17	Read data setup time	t_{DSR}	30	—	30	—	ns
18	Read data hold time	t_{DHR}	0	—	0	—	ns
19	Write data delay time	t_{DDW}	—	60	—	60	ns
21	Write data hold time, $t_{DHW} = 1/8 t_{cyc}$	t_{DHW}	110	—	27	—	ns
29	MPU address access time ⁽³⁾ $t_{ACCA} = t_{cyc} - t_f - t_{DSR} - t_{AD}$	t_{ACCA}	765	—	181	—	ns
39	Write data setup time ⁽³⁾ , $t_{DSW} = PW_{EH} - t_{DDW}$	t_{DSW}	410	—	77	—	ns
50	E valid chip-select delay time	t_{ECSD}	—	60	—	60	ns
51	E valid chip-select access time ⁽³⁾ $t_{ECSA} = PW_{EH} - t_{ECSD} - t_{DSR}$	t_{ECSA}	380	—	47	—	ns
52	Chip-select hold time	t_{CH}	0	20	0	20	ns
54	Address valid chip-select delay time $t_{ACSD} = 1/4 t_{cyc} + 40$ ns	t_{ACSD}	—	310	—	143	ns
55	Address valid chip-select access time $t_{ACSA} = t_{cyc} - t_f - t_{DSR} - t_{ACSD}$ ⁽³⁾	t_{ACSA}	640	—	140	—	ns
56	Address valid to chip-select time	t_{AVCS}	10	—	10	—	ns
57	Address valid to data three-state time	t_{AVDZ}	—	10	—	10	ns

1. $V_{DD} = 3.0$ Vdc to 5.5 Vdc, $V_{SS} = 0$ Vdc, $T_A = T_L$ to T_H , unless otherwise noted

All timing measurements refer to 20% V_{DD} and 70% V_{DD} , unless otherwise noted.

2. Input clocks with duty cycles other than 50% affect bus performance.

3. This parameter is affected by clock stretching. Add $n(t_{cyc})$ to parameter value, where $n = 1, 2,$ or 3 depending on values written to CSCSTR register.

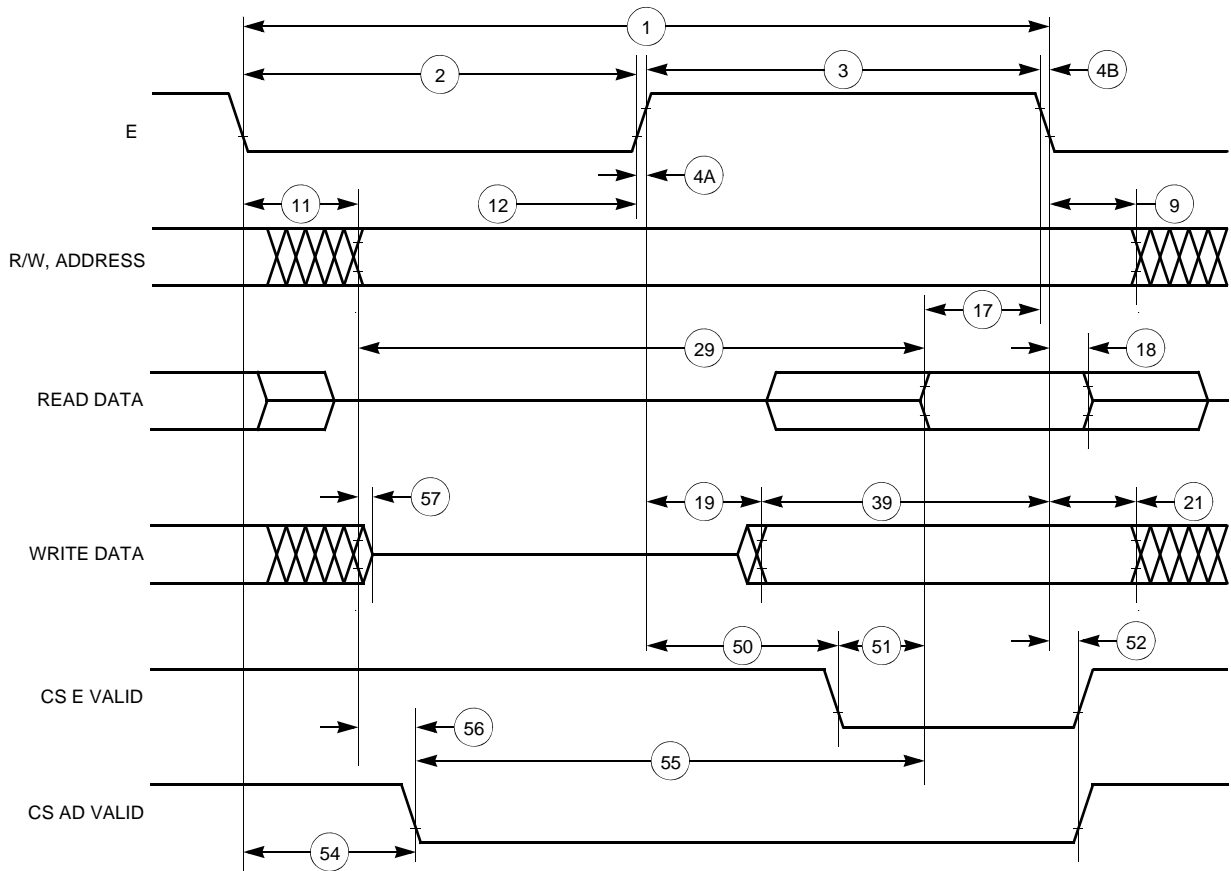


Figure 12-9. Expansion Bus Timing

12.17 Serial Peripheral Interface Timing

Num	Characteristic ⁽¹⁾	Symbol	1.0 MHz		2.0 MHz		3.0 MHz		4.0 MHz ⁽²⁾		Unit
			Min	Max	Min	Max	Min	Max	Min	Max	
	Operating frequency Master Slave	$f_{op(m)}$ $f_{op(s)}$	dc dc	0.5 1.0	dc dc	0.5 2.0	dc dc	0.5 3.0	dc dc	0.5 4.0	f_{op} MHz
1	Cycle time Master Slave	$t_{cyc(m)}$ $t_{cyc(s)}$	2.0 1000	— —	2.0 500	— —	2.0 333	— —	2.0 250	— —	t_{cyc} ns
2	Enable lead time Master ⁽³⁾ Slave	$t_{lead(m)}$ $t_{lead(s)}$	— 500	— —	— 250	— —	— 240	— —	— 200	— —	ns
3	Enable lag time Master ⁽³⁾ Slave	$t_{lag(m)}$ $t_{lag(s)}$	— 500	— —	— 250	— —	— 240	— —	— 200	— —	ns
4	Clock (SCK) high time Master Slave	$t_{w(SCKH)m}$ $t_{w(SCKH)s}$	680 380	— —	340 190	— —	227 127	— —	130 85	— —	ns
5	Clock (SCK) low time Master Slave	$t_{w(SCKL)m}$ $t_{w(SCKL)s}$	680 380	— —	340 190	— —	227 127	— —	130 85	— —	ns
6	Data setup time (inputs) Master Slave	$t_{su(m)}$ $t_{su(s)}$	100 100	— —	100 100	— —	100 100	— —	100 100	— —	ns
7	Data hold time (inputs) Master Slave	$t_{h(m)}$ $t_{h(s)}$	100 100	— —	100 100	— —	100 100	— —	100 100	— —	ns
8	Slave access time (time to data active from high-impedance state)	t_a	0	120	0	120	0	120	0	120	ns
9	Slave disable time (hold time to high-impedance state)	t_{dis}	—	240	—	240	—	167	—	125	ns
10	Data valid (after enable edge) ⁽⁴⁾	$t_{v(s)}$	—	240	—	240	—	167	—	125	ns
11	Data hold time (outputs) (after enable edge)	t_{ho}	0	—	0	—	0	—	0	—	ns
12	Rise time ⁽⁴⁾ SPI outputs (SCK, MOSI, and MISO) SPI inputs (SCK, MOSI, MISO, and SS)	t_{rm} t_{rs}	— —	100 2.0	— —	100 2.0	— —	100 2.0	— —	100 2.0	ns μ s
13	Fall time ⁽⁴⁾ SPI outputs (SCK, MOSI, and MISO) SPI inputs (SCK, MOSI, MISO, and SS)	t_{fm} t_{fs}	— —	100 2.0	— —	100 2.0	— —	100 2.0	— —	100 2.0	ns μ s

1. $V_{DD} = 4.5$ to 5.5 Vdc (standard devices), $V_{DD} = 3.0$ to 5.5 Vdc (extended voltage devices), $V_{SS} = 0$ Vdc, $T_A = T_L$ to T_H All timing measurements refer to 20% V_{DD} and 70% V_{DD} , unless otherwise noted.

2. Extended voltages device operation is not guaranteed above 3.0 MHz.

3. Signal production depends on software.

4. Capacitive load on all SPI pins is 200 pF on standard devices, 100 pF on extended voltage devices.

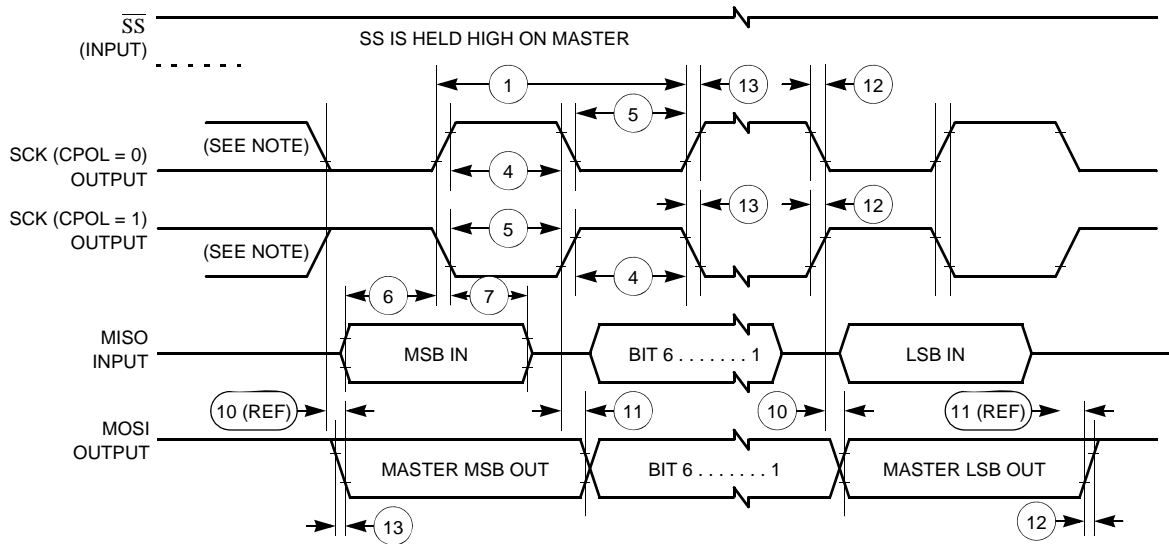
12.18 Serial Peripheral Interface Timing, Extended Voltage Devices

Num	Characteristic ⁽¹⁾	Symbol	1.0 MHz		3.0 MHz		Unit
			Min	Max	Min	Max	
	Operating frequency Master Slave	$f_{op(m)}$ $f_{op(s)}$	dc dc	0.5 1.0	dc dc	0.5 3.0	f_{op} MHz
1	Cycle time Master Slave	$t_{cyc(m)}$ $t_{cyc(s)}$	2.0 1000	— —	2.0 333	— —	t_{cyc} ns
2	Enable lead time Master ⁽²⁾ Slave	$t_{lead(m)}$ $t_{lead(s)}$	— 500	— —	— 240	— —	ns
3	Enable lag time Master ⁽²⁾ Slave	$t_{lag(m)}$ $t_{lag(s)}$	— 500	— —	— 240	— —	ns
4	Clock (SCK) high time Master Slave	$t_{w(SCKH)m}$ $t_{w(SCKH)s}$	680 380	— —	227 127	— —	ns
5	Clock (SCK) low time Master Slave	$t_{w(SCKL)m}$ $t_{w(SCKL)s}$	680 380	— —	227 127	— —	ns
6	Data setup time (inputs) Master Slave	$t_{su(m)}$ $t_{su(s)}$	100 100	— —	100 100	— —	ns
7	Data hold time (inputs) Master Slave	$t_{h(m)}$ $t_{h(s)}$	100 100	— —	100 100	— —	ns
8	Slave access time (time to data active from high-impedance state)	t_a	0	120	0	120	ns
9	Slave disable time (hold time to high-impedance state)	t_{dis}	—	240	—	167	ns
10	Data valid (after enable edge) ⁽³⁾	$t_{v(s)}$	—	240	—	167	ns
11	Data hold time (outputs) (after enable edge)	t_{ho}	0	—	0	—	ns
12	Rise time (20% V_{DD} to 70% V_{DD} , $C_L = 100$ pF) SPI outputs (SCK, MOSI, and MISO) SPI inputs (SCK, MOSI, MISO, and SS)	t_{rm} t_{rs}	— —	100 2.0	— —	100 2.0	ns μ s
13	Fall time (70% V_{DD} to 20% V_{DD} , $C_L = 100$ pF) SPI outputs (SCK, MOSI, and MISO) SPI inputs (SCK, MOSI, MISO, and SS)	t_{fm} t_{fs}	— —	100 2.0	— —	100 2.0	ns μ s

1. $V_{DD} = 3.0$ Vdc to 5.5 Vdc, $V_{SS} = 0$ Vdc, $T_A = T_L$ to T_H All timing is shown with respect to 20% V_{DD} and 70% V_{DD} , unless otherwise noted.

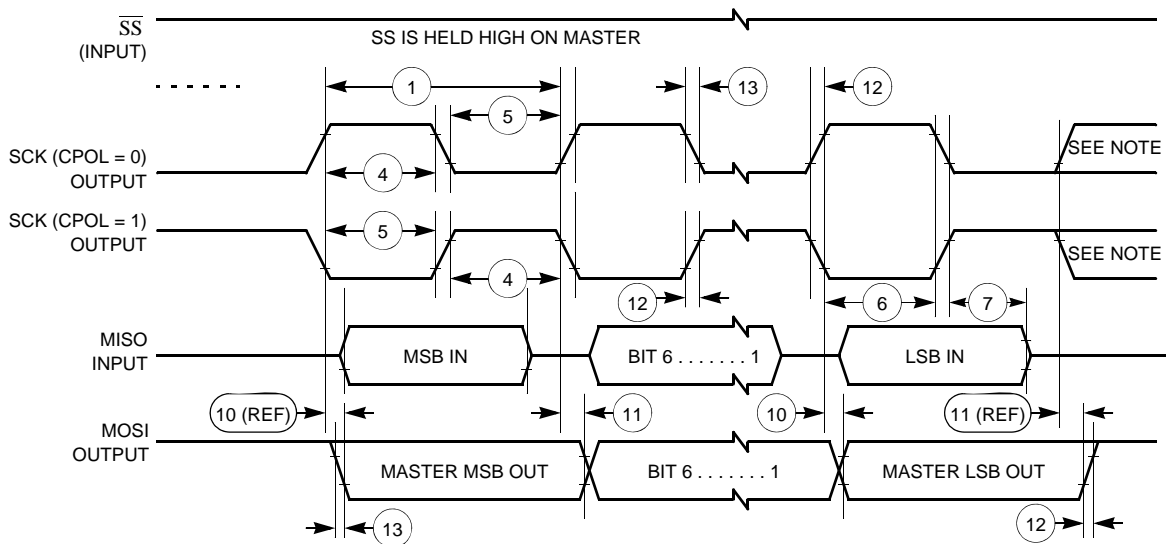
2. Signal production depends on software.

3. Assumes 100 pF load on all SPI pins



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

a) SPI Master Timing (CPHA = 0)

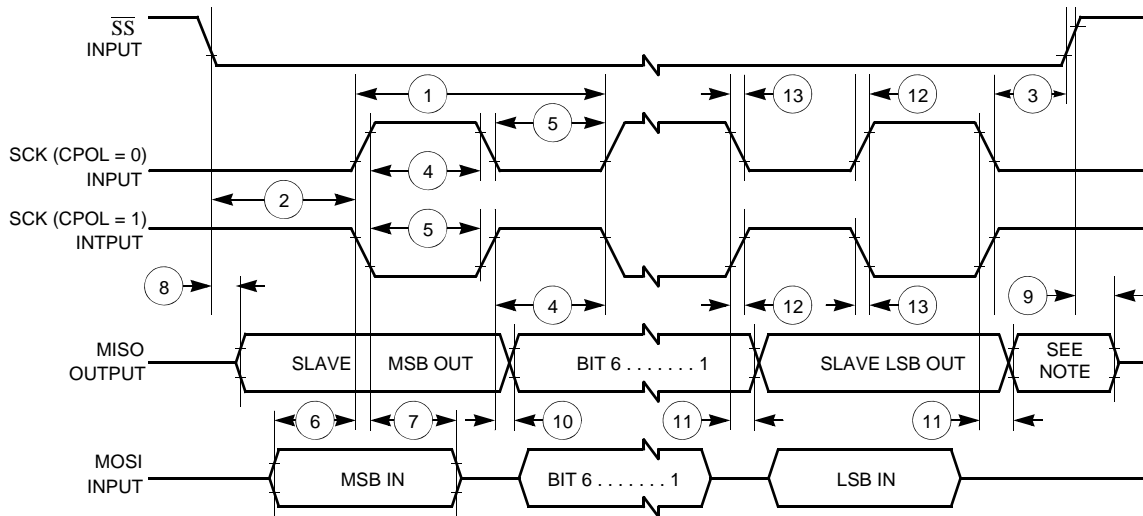


Note: This last clock edge is generated internally but is not seen at the SCK pin.

b) SPI Master Timing (CPHA = 1)

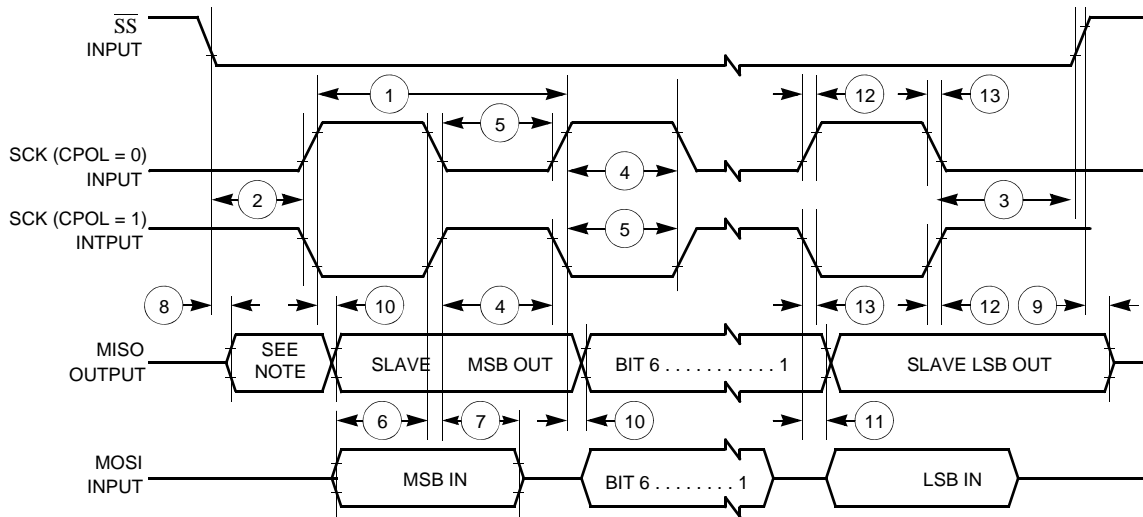
Figure 12-10. SPI Timing Diagram (Sheet 1 of 2)

Electrical Characteristics



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

a) SPI Slave Timing (CPHA = 0)



Note: Not defined, but normally LSB of character previously transmitted

b) SPI Slave Timing (CPHA = 1)

Figure 12-10. SPI Timing Diagram (Sheet 2 of 2)

12.19 EEPROM Characteristics

Characteristic ⁽¹⁾	Temperature Range			Unit
	–40 to 85 °C	–40 to 105 °C	–40 to 125 °C	
Programming time <1.0 MHz, RCO enabled ⁽²⁾ 1.0 to 2.0 MHz, RCO disabled ≥ 2.0 MHz (or anytime RCO enabled)	10 20 10	15 Must use RCO 15	20 Must use RCO 20	ms
Erase time ⁽²⁾ Byte, row, and bulk	10	10	10	ms
Write/erase endurance ⁽³⁾	10,000	10,000	10,000	Cycles
Data retention ⁽³⁾	10	10	10	Years

1. $V_{DD} = 5.0 \text{ Vdc} \pm 10\%$, $V_{SS} = 0 \text{ Vdc}$, $T_A = T_L$ to T_H

2. 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 below 1.0 MHz.

3. Refer to *Reliability Monitor Report* (current quarterly issue) for current failure rate information.

12.20 EEPROM Characteristics, Extended Voltage Devices

Characteristic ⁽¹⁾	Temperature Range –20 to 70 °C	Unit
Programming time 3 V, E ≤ 2.0 MHz, RCO enabled ⁽²⁾ 5 V, E ≤ 2.0 MHz, RCO enabled	25 10	ms
Erase time Byte, row, and bulk 3 V, E ≤ 2.0 MHz, RCO enabled ⁽²⁾ 5 V, E ≤ 2.0 MHz, RCO enabled	25 10	ms
Write/erase endurance ⁽³⁾	10,000	Cycles
Data retention ⁽³⁾	10	Years

1. $V_{DD} = 3.0 \text{ Vdc}$ to 5.5 Vdc , $V_{SS} = 0 \text{ Vdc}$, $T_A = T_L$ to T_H

2. The RC oscillator (RCO) must be enabled (by setting the CSEL bit in the OPTION register) for EEPROM programming and erasure.

3. Refer to *Reliability Monitor Report* (current quarterly issue) for current failure rate information.

Section 13. Mechanical Data

13.1 Contents

13.2	Introduction	277
13.3	84-Pin Plastic-Leaded Chip Carrier	279
13.4	84-Pin Ceramic-Leaded Chip Carrier	280
13.5	80-Pin Quad Flat Pack	281
13.6	80-Pin Thin Quad Flat Pack	282

13.2 Introduction

The TK68HC11K series microcontrollers are available in:

- 84-pin plastic-leaded chip carrier (PLCC)
- 84-pin ceramic-leaded chip carrier (CLCC)
- 80-pin quad flat pack (QFP)
- 80-pin thin quad flat pack (TQFP)

The diagrams will be added to this document.

For more information contact:

- Local Tekmos Sales Office
 - Phone 1-512-342-9871
 - EMAIL Sales@Tekmos.com;
 - <http://www.tekmos.co>

Section 14. Ordering Information

14.1 Contents

14.2	Part Designators	283
14.3	Part Number Construction	283

14.2 Part Designators

Use **Table 14-1** to determine part numbers when placing an order.

Table 14-1. TK68HC11K Family Devices

Device Number	ROM or EPROM	RAM	EEPROM	I/O	Chip Select	Slow Mode	Packages
TK68HC(L)11K0 TK68HC(L)11K1 TK68HC(L)11K3 TK68HC(L)11K4	0 0 24 K 24 K	768 768 768 768	0 640 0 640	37 37 62 62	Yes Yes Yes Yes	No No No No	84-pin PLCC ⁽¹⁾ 80-pin QFP ⁽²⁾
TK68HC711K4	24 K	768	640	62	Yes	No	84-pin CLCC ⁽³⁾ 84-pin PLCC 80-pin QFP
TK68HC11KS2 TK68HC11KS8	32 K 48 K	1 K 1.5 K	640 640	51 51	No No	Yes Yes	68-pin PLCC 68-pin PLCC and 80-pin TQFP ⁽⁴⁾
TK68HC711KS2 TK68HC711KS8	32 K 48 K	1 K 1.5 K	640 640	51 51	No No	Yes Yes	68-pin CLCC 68-pin PLCC and 80-pin TQFP

1. PLCC = Plastic leaded chip carrier
2. QFP = Quad flat pack
3. CLCC = Ceramic leaded chip carrier
4. TQFP = Thin quad flat pack

14.3 Part Number Construction

Figure 14-1 illustrates how Tekmos TK68HC11K part numbers are constructed.

Ordering Information

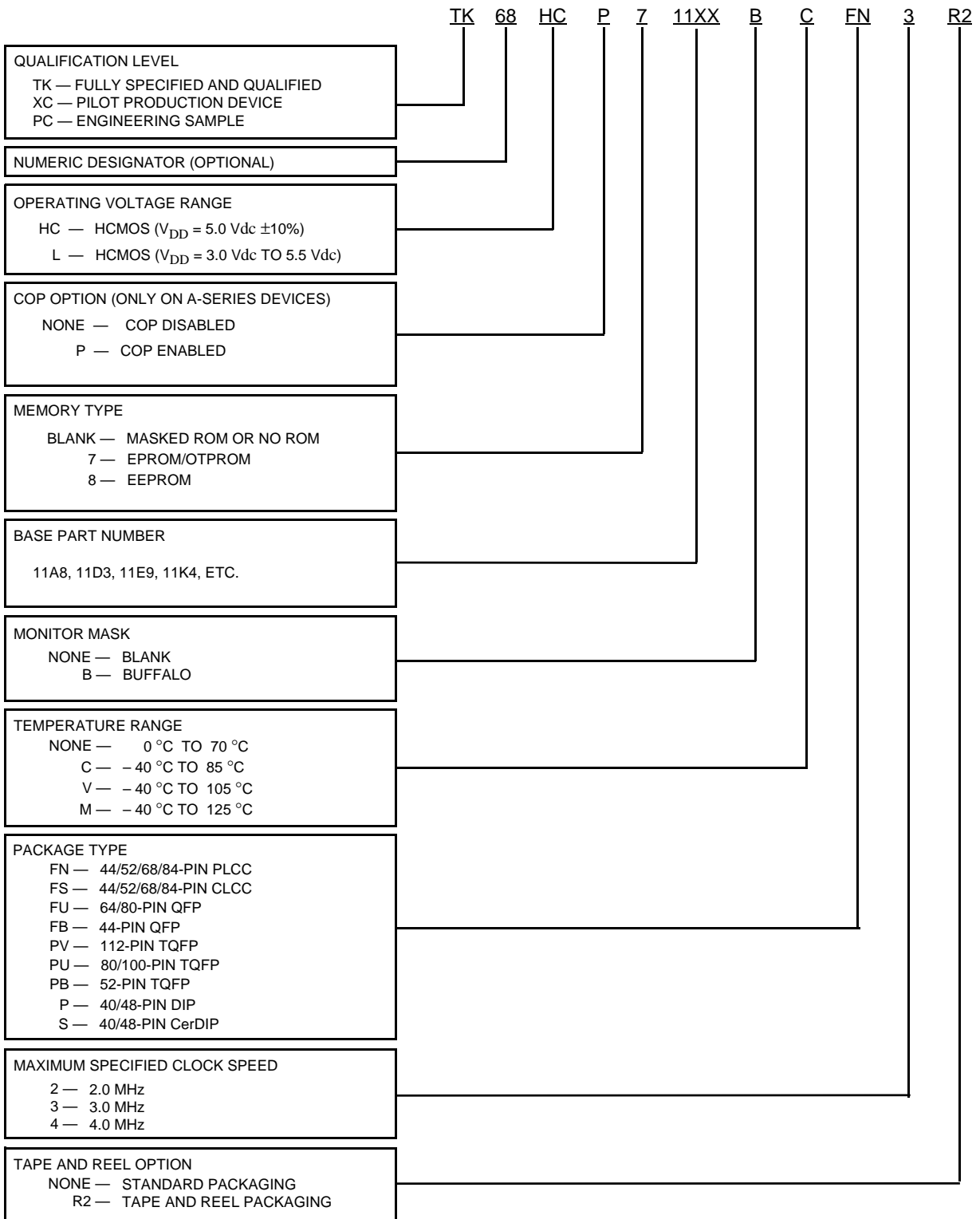


Figure 14-1. TK68HC11 Part Numbering System

Section 15. Differences Between Tekmos and Freescale Devices

The Tekmos TK68HC11K1 is a new design that was created by Tekmos. There are some differences between the Tekmos part and those manufactured by Freescale. These differences are listed below.

EEPROM

The Tekmos EEPROM does not directly support row erase. If row erase is activated, the Tekmos part will perform consecutive byte erases until the entire row is erased.

ADC

The Tekmos ADC uses a resistive DAC for the reference voltage. The original Freescale part uses a capacitive DAC. The Tekmos DAC is only connected to the reference pins when the ADC is enabled.

ESD Protection

Tekmos uses P-channel diodes on all pins for ESD protection. This limits the maximum overvoltage voltage on a pin to $V_{DD} + 0.3$ volts. The MODB pin is an exception since it is designed to provide a standby supply to the RAM during power down situations.

EPROM

The Freescale 68HC711 parts used OTP EPROM for program storage. The equivalent Tekmos parts use flash. This requires both a different programming algorithm and different programming voltages. Other than that, the differences are undetectable in the application.

Differences Between Tekmos and Freescale

Index

B

BAUD Baud Rate
SCP[1:0] SCI Baud Rate Prescaler Selects 159

C

CFORC Timer Compare Force
FOC[1:5] Force Output Comparison 206
CONFIG System Configuration
NOCOP ROM/PROM Enable 106
ROMON ROM/PROM Enable 86, 87

H

HPRIO Highest Priority I-Bit Interrupt and Miscellaneous
MDA Mode Select A 80
PSEL[3:0] Priority Select Bits 122
RBOOT Read Bootstrap ROM 80
SMOD Special Mode Select 80

I

INIT RAM and I/O Mapping Register 67
RAM[3:0] RAM Map Position 83
REG[3:0] Register Block Position 83

L

LIRDV LIR Driven 39

O

OC1D Output Compare 1 Data
OC1D[7:3] Output Compare Masks 209
OC1M Output Compare 1 Mask
OC1M[7:3] Output Compare 1 Masks 207
OPTION System Configuration Options 67
CME Clock Monitor Enable 110
CR[1:0] COP Timer Rate Select Bits 107
DLY Enable Oscillator Startup Delay 132
IRQE Configure IRQ for Edge-Sensitive Operation 121

P

PACTL Pulse Accumulator Control
I4/O5 Input Capture 4/Output Compare 5 193
PAEN Pulse Accumulator System Enable 212

PAMOD Pulse Accumulator Mode 213
RTR[1:0] Real Time Interrupt Rate Select 219
PPROG EPROM Programming Control
ELAT PROM Latch Control 91

S

SCCR1 SCI Control Register 1
M Mode (SCI Word Size) 160
WAKE Wakeup mode 160
SCCR2 SCI Control Register 2
ILIE Idle Line Interrupt Enable 161
RE Receiver Enable 162
RIE Receiver Interrupt Enable 161
RWU Receiver Wakeup Control 162
TCIE Transmit Complete Interrupt Enable 161
TE Transmitter Enable 162
TIE Transmit Interrupt Enable 161
SCI Control Register 2
SBK Send Break 162
SCSR SCI Status Register
FE Framing Error Flag 164
IDLE Idle Line Detected Flag 163
NF Noise Error Flag 163
OR Overrun Error Flag 163
RDRF Receive Data Register Full Flag 163
TC Transmit Complete Flag 163
TDRE Transmit Data Register Empty Flag 163
SPCR Serial Peripheral Control Register
CPHA Clock Phase 175
CPOL Clock Polarity 175
DWOM Port D Wired-OR Mode 173
MSTR Master Mode Select 174
SPE Serial Peripheral System Enable 173
SPR[1:0] SPI Clock Rate Selects 175
SPSR Serial Peripheral Status Register
MODF Mode Fault 176
SPIF SPI Transfer Complete Flag 176
WCOL Write Collision 176

T

TCTL1 Timer Control 1
OM[2:5] Output Mode 205
TCTL2 Timer Control 2

- EDGxB and EDGxA Input Capture Edge Control 198
- TFLG1 Timer Interrupt Flag 1
 - I4/O5F Input Capture 4/Output Compare 5 Flag 196, 203
 - IC1F–IC3F Input Capture x Flag 196
 - OC1F–OC4F Output Compare x Flag 203
- TFLG2 Timer Interrupt Flag 2
 - PAIF Pulse Accumulator Input Edge Flag 215
 - PAOVF Pulse Accumulator Overflow Flag 215
 - RTIF Real-Time Interrupt Flag 217
 - TOF Timer Overflow Interrupt Flag 189
- TMSK1 Timer Interrupt Mask 1
 - I4/O5I Input Capture 4 or Output Compare 5 Interrupt Enable 197, 204
 - IC1I–IC3I Input Capture x Interrupt Enable 197
 - OC1I–OC4I Output Compare x Interrupt Enable 204
- TMSK2 Timer Interrupt Mask 2 67
 - PR[1:0] Timer Prescaler Select 191
 - RTII Real-time Interrupt Enable 218
 - TOI Timer Overflow Interrupt Enable 190



TK68HC11K Family

Home Page:

<http://www.tekmos.com/>

Product Page:

<https://www.tekmos.com/68hc711-microcontrollers/tk68hc11k1-microcontroller>

Distributors:

<https://www.tekmos.com/contact/tekmos-distributors>

Sales Representatives:

<https://www.tekmos.com/contact/sales-representative>

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

Tekmos reserves the right to make changes without further notice to any products herein. Tekmos makes no warranty, representation or guarantee regarding the suitability of its products for any particular purpose, nor does Tekmos 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 which may be provided in Tekmos data sheets and/or specifications can and do vary in different applications and actual performance may vary over time. All operating parameters, including "Typicals" must be validated for each customer application by customer's technical experts. Tekmos does not convey any license under its patent rights nor the rights of others. Tekmos 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 Tekmos product could create a situation where personal injury or death may occur. Should Buyer purchase or use Tekmos products for any such unintended or unauthorized application, Buyer shall indemnify and hold Tekmos 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 Tekmos was negligent regarding the design or manufacture of the part. Tekmos is a registered trademarks of Tekmos, Inc. Tekmos, Inc. is an Equal Opportunity/Affirmative Action Employer.