

Welcome to E-XFL.COM

#### What is "Embedded - Microcontrollers"?

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

#### Applications of "<u>Embedded -</u> <u>Microcontrollers</u>"

#### Details

E·XFl

Product Status	Active
Core Processor	S08
Core Size	8-Bit
Speed	40MHz
Connectivity	I²C, SCI, SPI
Peripherals	LVD, POR, PWM, WDT
Number of I/O	34
Program Memory Size	32KB (32K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	2K x 8
Voltage - Supply (Vcc/Vdd)	2.7V ~ 5.5V
Data Converters	A/D 8x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 125°C (TA)
Mounting Type	Surface Mount
Package / Case	44-LQFP
Supplier Device Package	44-LQFP (10x10)
Purchase URL	https://www.e-xfl.com/pro/item?MUrl=&PartUrl=s9s08aw32e5mfge

Email: info@E-XFL.COM

Address: Room A, 16/F, Full Win Commercial Centre, 573 Nathan Road, Mongkok, Hong Kong



**Chapter 2 Pins and Connections** 

# 2.3.1 Power ( $V_{DD}$ , 2 x $V_{SS}$ , $V_{DDAD}$ , $V_{SSAD}$ )

 $V_{DD}$  and  $V_{SS}$  are the primary power supply pins for the MCU. This voltage source supplies power to all I/O buffer circuitry and to an internal voltage regulator. The internal voltage regulator provides regulated lower-voltage source to the CPU and other internal circuitry of the MCU.

Typically, application systems have two separate capacitors across the power pins. In this case, there should be a bulk electrolytic capacitor, such as a 10- $\mu$ F tantalum capacitor, to provide bulk charge storage for the overall system and a 0.1- $\mu$ F ceramic bypass capacitor located as near to the paired V<sub>DD</sub> and V<sub>SS</sub> power pins as practical to suppress high-frequency noise. The MC9S08AW60 has a second V<sub>SS</sub> pin. This pin should be connected to the system ground plane or to the primary V<sub>SS</sub> pin through a low-impedance connection.

 $V_{DDAD}$  and  $V_{SSAD}$  are the analog power supply pins for the MCU. This voltage source supplies power to the ADC module. A 0.1- $\mu$ F ceramic bypass capacitor should be located as near to the analog power pins as practical to suppress high-frequency noise.

# 2.3.2 Oscillator (XTAL, EXTAL)

Out of reset, the MCU uses an internally generated clock (self-clocked mode —  $f_{Self\_reset}$ ) equivalent to about 8-MHz crystal rate. This frequency source is used during reset startup and can be enabled as the clock source for stop recovery to avoid the need for a long crystal startup delay. This MCU also contains a trimmable internal clock generator (ICG) module that can be used to run the MCU. For more information on the ICG, see the Chapter 8, "Internal Clock Generator (S08ICGV4)."

The oscillator amplitude on XTAL and EXTAL is gain limited for low-power oscillation. Typically, these pins have a 1-V peak-to-peak signal. For noisy environments, the high gain output (HGO) bit can be set to enable rail-to-rail oscillation.

The oscillator in this MCU is a Pierce oscillator that can accommodate a crystal or ceramic resonator in either of two frequency ranges selected by the RANGE bit in the ICGC1 register. Rather than a crystal or ceramic resonator, an external oscillator can be connected to the EXTAL input pin.

Refer to Figure 2-4 for the following discussion.  $R_S$  (when used) and  $R_F$  should be low-inductance resistors such as carbon composition resistors. Wire-wound resistors, and some metal film resistors, have too much inductance. C1 and C2 normally should be high-quality ceramic capacitors that are specifically designed for high-frequency applications.

 $R_F$  is used to provide a bias path to keep the EXTAL input in its linear range during crystal startup and its value is not generally critical. Typical systems use 1 M $\Omega$  to 10 M $\Omega$ . Higher values are sensitive to humidity and lower values reduce gain and (in extreme cases) could prevent startup.

C1 and C2 are typically in the 5-pF to 25-pF range and are chosen to match the requirements of a specific crystal or resonator. Be sure to take into account printed circuit board (PCB) capacitance and MCU pin capacitance when sizing C1 and C2. The crystal manufacturer typically specifies a load capacitance which is the series combination of C1 and C2 which are usually the same size. As a first-order approximation, use 10 pF as an estimate of combined pin and PCB capacitance for each oscillator pin (EXTAL and XTAL).

#### **Chapter 4 Memory**





Figure 4-2. MC9S08AW32 and MC9S08AW16 Memory Map



Chapter 4 Memory

Address	Register Name	Bit 7	6	5	4	3	2	1	Bit 0
\$1846	PTBDS	PTBDS7	PTBDS6	PTBDS5	PTBDS4	PTBDS3	PTBDS2	PTBDS1	PTBDS0
\$1847	Reserved	_	—	_	—	—	_	_	_
\$1848	PTCPE	0	PTCPE6	PTCPE5	PTCPE4	PTCPE3	PTCPE2	PTCPE1	PTCPE0
\$1849	PTCSE	0	PTCSE6	PTCSE5	PTCSE4	PTCSE3	PTCSE2	PTCSE1	PTCSE0
\$184A	PTCDS	0	PTCDS6	PTCDS5	PTCDS4	PTCDS3	PTCDS2	PTCDS1	PTCDS0
\$184B	Reserved	_	_	—	—	_	—	—	—
\$184C	PTDPE	PTDPE7	PTDPE6	PTDPE5	PTDPE4	PTDPE3	PTDPE2	PTDPE1	PTDPE0
\$184D	PTDSE	PTDSE7	PTDSE6	PTDSE5	PTDSE4	PTDSE3	PTDSE2	PTDSE1	PTDSE0
\$184E	PTDDS	PTDDS7	PTDDS6	PTDDS5	PTDDS4	PTDDS3	PTDDS2	PTDDS1	PTDDS0
\$184F	Reserved	_		_	_		_	_	—
\$1850	PTEPE	PTEPE7	PTEPE6	PTEPE5	PTEPE4	PTEPE3	PTEPE2	PTEPE1	PTEPE0
\$1851	PTESE	PTESE7	PTESE6	PTESE5	PTESE4	PTESE3	PTESE2	PTESE1	PTESE0
\$1852	PTEDS	PTEDS7	PTEDS6	PTEDS5	PTEDS4	PTEDS3	PTEDS2	PTEDS1	PTEDS0
\$1853	Reserved		_		—				
\$1854	PTFPE	PTFPE7	PTFPE6	PTFPE5	PTFPE4	PTFPE3	PTFPE2	PTFPE1	PTFPE0
\$1855	PTFSE	PTFSE7	PTFSE6	PTFSE5	PTFSE4	PTFSE3	PTFSE2	PTFSE1	PTFSE0
\$1856	PTFDS	PTFDS7	PTFDS6	PTFDS5	PTFDS4	PTFDS3	PTFDS2	PTFDS1	PTFDS0
\$1857	Reserved				—				
\$1858	PTGPE	0	PTGPE6	PTGPE5	PTGPE4	PTGPE3	PTGPE2	PTGPE1	PTGPE0
\$1859	PTGSE	0	PTGSE6	PTGSE5	PTGSE4	PTGSE3	PTGSE2	PTGSE1	PTGSE0
\$185A	PTGDS	0	PTGDS6	PTGDS5	PTGDS4	PTGDS3	PTGDS2	PTGDS1	PTGDS0
\$185B– \$185F	Reserved	_	_	_		_	_	_	_

Table 1 1	) Linh Dono	Dogiator	C	(Cheat )	of 0)
Table 4-3	s. пign-Page	Register	Summary	(Sneet 2	012)

<sup>1</sup> This reserved bit must always be written to 0.

Nonvolatile FLASH registers, shown in Table 4-4, are located in the FLASH memory. These registers include an 8-byte backdoor key which optionally can be used to gain access to secure memory resources. During reset events, the contents of NVPROT and NVOPT in the nonvolatile register area of the FLASH memory are transferred into corresponding FPROT and FOPT working registers in the high-page registers to control security and block protection options.



## 4.6.4 FLASH Protection Register (FPROT and NVPROT)

During reset, the contents of the nonvolatile location NVPROT are copied from FLASH into FPROT. This register may be read at any time.

- If FPDIS = 0, then protection can be increased (in other words, a smaller value of FPS can be written).
- If FPDIS = 1, then writes do not change protection.

	7	6	5	4	3	2	1	0
R	FPS7	FPS6	FPS5	FPS4	FPS3	FPS2	FPS1	FPDIS
w	(1)	(1)	(1)	(1)	(1)	(1)	(1)	(1)

Reset

This register is loaded from nonvolatile location NVPROT during reset.

<sup>1</sup> Background commands can be used to change the contents of these bits in FPROT.

### Figure 4-9. FLASH Protection Register (FPROT)

#### Table 4-11. FPROT Register Field Descriptions

Field	Description
7:1 FPS[7:1]	<b>FLASH Protect Select Bits</b> — When FPDIS = 0, this 7-bit field determines the ending address of unprotected FLASH locations at the high address end of the FLASH. Protected FLASH locations cannot be erased or programmed.
0 FPDIS	<ul> <li>FLASH Protection Disable</li> <li>0 FLASH block specified by FPS[7:1] is block protected (program and erase not allowed).</li> <li>1 No FLASH block is protected.</li> </ul>

## 4.6.5 FLASH Status Register (FSTAT)

Bits 3, 1, and 0 always read 0 and writes have no meaning or effect. The remaining five bits are status bits that can be read at any time. Writes to these bits have special meanings that are discussed in the bit descriptions.



Figure 4-10. FLASH Status Register (FSTAT)



I bit in the CCR is 0 to allow interrupts. The global interrupt mask (I bit) in the CCR is initially set after reset which masks (prevents) all maskable interrupt sources. The user program initializes the stack pointer and performs other system setup before clearing the I bit to allow the CPU to respond to interrupts.

When the CPU receives a qualified interrupt request, it completes the current instruction before responding to the interrupt. The interrupt sequence obeys the same cycle-by-cycle sequence as the SWI instruction and consists of:

- Saving the CPU registers on the stack
- Setting the I bit in the CCR to mask further interrupts
- Fetching the interrupt vector for the highest-priority interrupt that is currently pending
- Filling the instruction queue with the first three bytes of program information starting from the address fetched from the interrupt vector locations

While the CPU is responding to the interrupt, the I bit is automatically set to avoid the possibility of another interrupt interrupting the ISR itself (this is called nesting of interrupts). Normally, the I bit is restored to 0 when the CCR is restored from the value stacked on entry to the ISR. In rare cases, the I bit may be cleared inside an ISR (after clearing the status flag that generated the interrupt) so that other interrupts can be serviced without waiting for the first service routine to finish. This practice is not recommended for anyone other than the most experienced programmers because it can lead to subtle program errors that are difficult to debug.

The interrupt service routine ends with a return-from-interrupt (RTI) instruction which restores the CCR, A, X, and PC registers to their pre-interrupt values by reading the previously saved information off the stack.

### NOTE

For compatibility with the M68HC08, the H register is not automatically saved and restored. It is good programming practice to push H onto the stack at the start of the interrupt service routine (ISR) and restore it immediately before the RTI that is used to return from the ISR.

When two or more interrupts are pending when the I bit is cleared, the highest priority source is serviced first (see Table 5-1).

### 5.5.1 Interrupt Stack Frame

Figure 5-1 shows the contents and organization of a stack frame. Before the interrupt, the stack pointer (SP) points at the next available byte location on the stack. The current values of CPU registers are stored on the stack starting with the low-order byte of the program counter (PCL) and ending with the CCR. After stacking, the SP points at the next available location on the stack which is the address that is one less than the address where the CCR was saved. The PC value that is stacked is the address of the instruction in the main program that would have executed next if the interrupt had not occurred.



#### Chapter 6 Parallel Input/Output

_	7	6	5	4	3	2	1	0
R W	PTBDS7	PTBDS6	PTBDS5	PTBDS4	PTBDS3	PTBDS2	PTBDS1	PTBDS0
Reset	0	0	0	0	0	0	0	0

### Figure 6-18. Output Drive Strength Selection for Port B (PTBDS)

### Table 6-11. PTBDS Register Field Descriptions

Field	Description
7:0 PTBDS[7:0]	<ul> <li>Output Drive Strength Selection for Port B Bits — Each of these control bits selects between low and high output drive for the associated PTB pin.</li> <li>0 Low output drive enabled for port B bit n.</li> <li>1 High output drive enabled for port B bit n.</li> </ul>



#### Chapter 7 Central Processor Unit (S08CPUV2)

interrupt service routine, this would allow nesting of interrupts (which is not recommended because it leads to programs that are difficult to debug and maintain).

For compatibility with the earlier M68HC05 MCUs, the high-order half of the H:X index register pair (H) is not saved on the stack as part of the interrupt sequence. The user must use a PSHH instruction at the beginning of the service routine to save H and then use a PULH instruction just before the RTI that ends the interrupt service routine. It is not necessary to save H if you are certain that the interrupt service routine does not use any instructions or auto-increment addressing modes that might change the value of H.

The software interrupt (SWI) instruction is like a hardware interrupt except that it is not masked by the global I bit in the CCR and it is associated with an instruction opcode within the program so it is not asynchronous to program execution.

### 7.4.3 Wait Mode Operation

The WAIT instruction enables interrupts by clearing the I bit in the CCR. It then halts the clocks to the CPU to reduce overall power consumption while the CPU is waiting for the interrupt or reset event that will wake the CPU from wait mode. When an interrupt or reset event occurs, the CPU clocks will resume and the interrupt or reset event will be processed normally.

If a serial BACKGROUND command is issued to the MCU through the background debug interface while the CPU is in wait mode, CPU clocks will resume and the CPU will enter active background mode where other serial background commands can be processed. This ensures that a host development system can still gain access to a target MCU even if it is in wait mode.

## 7.4.4 Stop Mode Operation

Usually, all system clocks, including the crystal oscillator (when used), are halted during stop mode to minimize power consumption. In such systems, external circuitry is needed to control the time spent in stop mode and to issue a signal to wake up the target MCU when it is time to resume processing. Unlike the earlier M68HC05 and M68HC08 MCUs, the HCS08 can be configured to keep a minimum set of clocks running in stop mode. This optionally allows an internal periodic signal to wake the target MCU from stop mode.

When a host debug system is connected to the background debug pin (BKGD) and the ENBDM control bit has been set by a serial command through the background interface (or because the MCU was reset into active background mode), the oscillator is forced to remain active when the MCU enters stop mode. In this case, if a serial BACKGROUND command is issued to the MCU through the background debug interface while the CPU is in stop mode, CPU clocks will resume and the CPU will enter active background mode where other serial background commands can be processed. This ensures that a host development system can still gain access to a target MCU even if it is in stop mode.

Recovery from stop mode depends on the particular HCS08 and whether the oscillator was stopped in stop mode. Refer to the Modes of Operation chapter for more details.



#### Chapter 7 Central Processor Unit (S08CPUV2)

Bit-Manip	ulation	Branch	Read-M	lodify-Write	Control			Register	/Memory		
				9E60 6 NEG 3 SP1					9ED0 5 SUB 4 SP2	9EE0 4 SUB 3 SP1	
				9E61 6 CBEQ 4 SP1					9ED1 5 CMP 4 SP2	9EE1 4 CMP 3 SP1	
									9ED2 5 SBC 4 SP2	9EE2 4 SBC 3 SP1	
				9E63 6 COM 3 SP1					9ED3 5 CPX 4 SP2	9EE3 4 CPX 3 SP1	9EF3 6 CPHX 3 SP1
				9E64 6 LSR 3 SP1					9ED4 5 AND 4 SP2	9EE4 4 AND 3 SP1	
									9ED5 5 BIT 4 SP2	9EE5 4 BIT 3 SP1	
				9E66 6 ROR 3 SP1					9ED6 5 LDA 4 SP2	9EE6 4 LDA 3 SP1	
				9E67 6 ASR 3 SP1					9ED7 5 STA 4 SP2	9EE7 4 STA 3 SP1	
				9E68 6 LSL 3 SP1					9ED8 5 EOR 4 SP2	9EE8 4 EOR 3 SP1	
				9E69 6 ROL 3 SP1					9ED9 5 ADC 4 SP2	9EE9 4 ADC 3 SP1	
				9E6A 6 DEC 3 SP1					9EDA 5 ORA 4 SP2	9EEA 4 ORA 3 SP1	
				9E6B 8 DBNZ 4 SP1					9EDB 5 ADD 4 SP2	9EEB 4 ADD 3 SP1	
				9E6C 6 INC 3 SP1							
				9E6D 5 TST 3 SP1							
						9EAE 5 LDHX 2 IX	9EBE 6 LDHX 4 IX2	9ECE 5 LDHX 3 IX1	9EDE 5 LDX 4 SP2	9EEE 4 LDX 3 SP1	9EFE 5 LDHX 3 SP1
				9E6F 6 CLR 3 SP1					9EDF 5 STX 4 SP2	9EEF 4 STX 3 SP1	9EFF 5 STHX 3 SP1

### Table 7-3. Opcode Map (Sheet 2 of 2)

Inherent Immediate Direct Extended DIR to DIR IX+ to DIR REL IX IX1 IX2 IMD DIX+ INH IMM DIR EXT DD IX+D

Relative Indexed, No Offset Indexed, 8-Bit Offset Indexed, 16-Bit Offset IMM to DIR DIR to IX+

Stack Pointer, 8-Bit Offset Stack Pointer, 16-Bit Offset Indexed, No Offset with Post Increment Indexed, 1-Byte Offset with Post Increment

SP1 SP2 IX+

IX1+

Note: All Sheet 2 Opcodes are Preceded by the Page 2 Prebyte (9E)

Prebyte (9E) and Opcode in Hexadecimal 9E60 6 NEG Number of Bytes 3 SP1 Addressing Mode

MC9S08AW60 Data Sheet, Rev 2



## 8.1 Introduction

The ICG provides multiple options for clock sources. This offers a user great flexibility when making choices between cost, precision, current draw, and performance. As seen in Figure 8-3, the ICG consists of four functional blocks. Each of these is briefly described here and then in more detail in a later section.

- Oscillator block The oscillator block provides means for connecting an external crystal or resonator. Two frequency ranges are software selectable to allow optimal startup and stability. Alternatively, the oscillator block can be used to route an external square wave to the system clock. External sources can provide a very precise clock source. The oscillator is capable of being configured for low power mode or high amplitude mode as selected by HGO.
- **Internal reference generator** The internal reference generator consists of two controlled clock sources. One is designed to be approximately 8 MHz and can be selected as a local clock for the background debug controller. The other internal reference clock source is typically 243 kHz and can be trimmed for finer accuracy via software when a precise timed event is input to the MCU. This provides a highly reliable, low-cost clock source.
- **Frequency-locked loop** A frequency-locked loop (FLL) stage takes either the internal or external clock source and multiplies it to a higher frequency. Status bits provide information when the circuit has achieved lock and when it falls out of lock. Additionally, this block can monitor the external reference clock and signals whether the clock is valid or not.
- **Clock select block** The clock select block provides several switch options for connecting different clock sources to the system clock tree. ICGDCLK is the multiplied clock frequency out of the FLL, ICGERCLK is the reference clock frequency from the crystal or external clock source, and FFE (fixed frequency enable) is a control signal used to control the system fixed frequency clock (XCLK). ICGLCLK is the clock source for the background debug controller (BDC).

### 8.1.1 Features

The module is intended to be very user friendly with many of the features occurring automatically without user intervention. To quickly configure the module, go to Section 8.5, "Initialization/Application Information" and pick an example that best suits the application needs.

Features of the ICG and clock distribution system:

- Several options for the primary clock source allow a wide range of cost, frequency, and precision choices:
  - 32 kHz-100 kHz crystal or resonator
  - 1 MHz–16 MHz crystal or resonator
  - External clock
  - Internal reference generator
- Defaults to self-clocked mode to minimize startup delays
- Frequency-locked loop (FLL) generates 8 MHz to 40 MHz (for bus rates up to 20 MHz)
   Uses external or internal clock as reference frequency
- Automatic lockout of non-running clock sources
- Reset or interrupt on loss of clock or loss of FLL lock



Chapter 9 Keyboard Interrupt (S08KBIV1)

### 9.3 Features

The keyboard interrupt (KBI) module features include:

- Four falling edge/low level sensitive
- Four falling edge/low level or rising edge/high level sensitive
- Choice of edge-only or edge-and-level sensitivity
- Common interrupt flag and interrupt enable control
- Capable of waking up the MCU from stop3 or wait mode



### 10.2.1 Features

The TPM has the following features:

- Each TPM may be configured for buffered, center-aligned pulse-width modulation (CPWM) on all channels
- Clock sources independently selectable per TPM (multiple TPMs device)
- Selectable clock sources (device dependent): bus clock, fixed system clock, external pin
- Clock prescaler taps for divide by 1, 2, 4, 8, 16, 32, 64, or 128
- 16-bit free-running or up/down (CPWM) count operation
- 16-bit modulus register to control counter range
- Timer system enable
- One interrupt per channel plus a terminal count interrupt for each TPM module (multiple TPMs device)
- Channel features:
  - Each channel may be input capture, output compare, or buffered edge-aligned PWM
  - Rising-edge, falling-edge, or any-edge input capture trigger
  - Set, clear, or toggle output compare action
  - Selectable polarity on PWM outputs

### 10.2.2 Block Diagram

Figure 10-2 shows the structure of a TPM. Some MCUs include more than one TPM, with various numbers of channels.



Chapter 11 Serial Communications Interface (S08SCIV2)

# 11.2.3 SCI Control Register 2 (SCIxC2)

This register can be read or written at any time.



### Figure 11-7. SCI Control Register 2 (SCIxC2)

### Table 11-4. SCIxC2 Register Field Descriptions

Field	Description
7 TIE	Transmit Interrupt Enable (for TDRE)0Hardware interrupts from TDRE disabled (use polling).1Hardware interrupt requested when TDRE flag is 1.
6 TCIE	Transmission Complete Interrupt Enable (for TC)0Hardware interrupts from TC disabled (use polling).1Hardware interrupt requested when TC flag is 1.
5 RIE	Receiver Interrupt Enable (for RDRF)0Hardware interrupts from RDRF disabled (use polling).1Hardware interrupt requested when RDRF flag is 1.
4 ILIE	Idle Line Interrupt Enable (for IDLE)00111Hardware interrupt requested when IDLE flag is 1.
3 TE	Transmitter Enable0Transmitter off.1Transmitter on.TE must be 1 in order to use the SCI transmitter. Normally, when TE = 1, the SCI forces the TxD pin to act as an output for the SCI system. If LOOPS = 1 and RSRC = 0, the TxD pin reverts to being a port B general-purpose I/O pin even if TE = 1.When the SCI is configured for single-wire operation (LOOPS = RSRC = 1), TXDIR controls the direction of traffic on the single SCI communication line (TxD pin).TE also can be used to queue an idle character by writing TE = 0 then TE = 1 while a transmission is in progress.Refer to Section 11.3.2.1, "Send Break and Queued Idle" for more details.When TE is written to 0, the transmitter keeps control of the port TxD pin until any data, queued idle, or queued break character finishes transmitting before allowing the pin to revert to a general-purpose I/O pin.
2 RE	Receiver Enable — When the SCI receiver is off, the RxD pin reverts to being a general-purpose port I/O pin.         0       Receiver off.         1       Receiver on.



#### IICChapter 13 Inter-Integrated Circuit (S08IICV1)



 Pins PTD7, PTD3, PTD2, and PTG4 contain both pullup and pulldown devices. Pulldown enabled when KBI is enabled (KBIPEn = 1) and rising edge is selected (KBEDGn = 1).

#### Figure 13-1. Block Diagram Highlighting the IIC Module

MC9S08AW60 Data Sheet, Rev 2



ADICLK	Selected Clock Source
00	Bus clock
01	Bus clock divided by 2
10	Alternate clock (ALTCLK)
11	Asynchronous clock (ADACK)

#### Table 14-8. Input Clock Select

### 14.4.8 Pin Control 1 Register (APCTL1)

The pin control registers are used to disable the I/O port control of MCU pins used as analog inputs. APCTL1 is used to control the pins associated with channels 0–7 of the ADC module.



Figure 14-11. Pin Control 1 Register (APCTL1)

Description
<ul> <li>ADC Pin Control 7 — ADPC7 is used to control the pin associated with channel AD7.</li> <li>0 AD7 pin I/O control enabled</li> <li>1 AD7 pin I/O control disabled</li> </ul>
<ul> <li>ADC Pin Control 6 — ADPC6 is used to control the pin associated with channel AD6.</li> <li>0 AD6 pin I/O control enabled</li> <li>1 AD6 pin I/O control disabled</li> </ul>
<ul> <li>ADC Pin Control 5 — ADPC5 is used to control the pin associated with channel AD5.</li> <li>0 AD5 pin I/O control enabled</li> <li>1 AD5 pin I/O control disabled</li> </ul>
<ul> <li>ADC Pin Control 4 — ADPC4 is used to control the pin associated with channel AD4.</li> <li>0 AD4 pin I/O control enabled</li> <li>1 AD4 pin I/O control disabled</li> </ul>
<ul> <li>ADC Pin Control 3 — ADPC3 is used to control the pin associated with channel AD3.</li> <li>0 AD3 pin I/O control enabled</li> <li>1 AD3 pin I/O control disabled</li> </ul>
<ul> <li>ADC Pin Control 2 — ADPC2 is used to control the pin associated with channel AD2.</li> <li>0 AD2 pin I/O control enabled</li> <li>1 AD2 pin I/O control disabled</li> </ul>

### Table 14-9. APCTL1 Register Field Descriptions



Field	Description
1 ADPC17	<ul> <li>ADC Pin Control 17 — ADPC17 is used to control the pin associated with channel AD17.</li> <li>0 AD17 pin I/O control enabled</li> <li>1 AD17 pin I/O control disabled</li> </ul>
0 ADPC16	<ul> <li>ADC Pin Control 16 — ADPC16 is used to control the pin associated with channel AD16.</li> <li>0 AD16 pin I/O control enabled</li> <li>1 AD16 pin I/O control disabled</li> </ul>

#### Table 14-11. APCTL3 Register Field Descriptions (continued)

# 14.5 Functional Description

The ADC module is disabled during reset or when the ADCH bits are all high. The module is idle when a conversion has completed and another conversion has not been initiated. When idle, the module is in its lowest power state.

The ADC can perform an analog-to-digital conversion on any of the software selectable channels. The selected channel voltage is converted by a successive approximation algorithm into an 11-bit digital result. In 8-bit mode, the selected channel voltage is converted by a successive approximation algorithm into a 9-bit digital result.

When the conversion is completed, the result is placed in the data registers (ADC1RH and ADC1RL).In 10-bit mode, the result is rounded to 10 bits and placed in ADC1RH and ADC1RL. In 8-bit mode, the result is rounded to 8 bits and placed in ADC1RL. The conversion complete flag (COCO) is then set and an interrupt is generated if the conversion complete interrupt has been enabled (AIEN = 1).

The ADC module has the capability of automatically comparing the result of a conversion with the contents of its compare registers. The compare function is enabled by setting the ACFE bit and operates in conjunction with any of the conversion modes and configurations.

## 14.5.1 Clock Select and Divide Control

One of four clock sources can be selected as the clock source for the ADC module. This clock source is then divided by a configurable value to generate the input clock to the converter (ADCK). The clock is selected from one of the following sources by means of the ADICLK bits.

- The bus clock, which is equal to the frequency at which software is executed. This is the default selection following reset.
- The bus clock divided by 2. For higher bus clock rates, this allows a maximum divide by 16 of the bus clock.
- ALTCLK, as defined for this MCU (See module section introduction).
- The asynchronous clock (ADACK) This clock is generated from a clock source within the ADC module. When selected as the clock source this clock remains active while the MCU is in wait or stop3 mode and allows conversions in these modes for lower noise operation.

Whichever clock is selected, its frequency must fall within the specified frequency range for ADCK. If the available clocks are too slow, the ADC will not perform according to specifications. If the available clocks



**Chapter 15 Development Support** 

## 15.1.1 Features

Features of the BDC module include:

- Single pin for mode selection and background communications
- BDC registers are not located in the memory map
- SYNC command to determine target communications rate
- Non-intrusive commands for memory access
- Active background mode commands for CPU register access
- GO and TRACE1 commands
- BACKGROUND command can wake CPU from stop or wait modes
- One hardware address breakpoint built into BDC
- Oscillator runs in stop mode, if BDC enabled
- COP watchdog disabled while in active background mode

Features of the ICE system include:

- Two trigger comparators: Two address + read/write (R/W) or one full address + data + R/W
- Flexible 8-word by 16-bit FIFO (first-in, first-out) buffer for capture information:
  - Change-of-flow addresses or
  - Event-only data
- Two types of breakpoints:
  - Tag breakpoints for instruction opcodes
  - Force breakpoints for any address access
- Nine trigger modes:
  - Basic: A-only, A OR B
  - Sequence: A then B
  - Full: A AND B data, A AND NOT B data
  - Event (store data): Event-only B, A then event-only B
  - Range: Inside range ( $A \le address \le B$ ), outside range (address < A or address > B)

# 15.2 Background Debug Controller (BDC)

All MCUs in the HCS08 family contain a single-wire background debug interface that supports in-circuit programming of on-chip nonvolatile memory and sophisticated non-intrusive debug capabilities. Unlike debug interfaces on earlier 8-bit MCUs, this system does not interfere with normal application resources. It does not use any user memory or locations in the memory map and does not share any on-chip peripherals.

BDC commands are divided into two groups:

• Active background mode commands require that the target MCU is in active background mode (the user program is not running). Active background mode commands allow the CPU registers to be read or written, and allow the user to trace one user instruction at a time, or GO to the user program from active background mode.



• Non-intrusive commands can be executed at any time even while the user's program is running. Non-intrusive commands allow a user to read or write MCU memory locations or access status and control registers within the background debug controller.

Typically, a relatively simple interface pod is used to translate commands from a host computer into commands for the custom serial interface to the single-wire background debug system. Depending on the development tool vendor, this interface pod may use a standard RS-232 serial port, a parallel printer port, or some other type of communications such as a universal serial bus (USB) to communicate between the host PC and the pod. The pod typically connects to the target system with ground, the BKGD pin, RESET, and sometimes  $V_{DD}$ . An open-drain connection to reset allows the host to force a target system reset, which is useful to regain control of a lost target system or to control startup of a target system before the on-chip nonvolatile memory has been programmed. Sometimes  $V_{DD}$  can be used to allow the pod to use power from the target system to avoid the need for a separate power supply. However, if the pod is powered separately, it can be connected to a running target system without forcing a target system reset or otherwise disturbing the running application program.



Figure 15-1. BDM Tool Connector

### 15.2.1 BKGD Pin Description

BKGD is the single-wire background debug interface pin. The primary function of this pin is for bidirectional serial communication of active background mode commands and data. During reset, this pin is used to select between starting in active background mode or starting the user's application program. This pin is also used to request a timed sync response pulse to allow a host development tool to determine the correct clock frequency for background debug serial communications.

BDC serial communications use a custom serial protocol first introduced on the M68HC12 family of microcontrollers. This protocol assumes the host knows the communication clock rate that is determined by the target BDC clock rate. All communication is initiated and controlled by the host that drives a high-to-low edge to signal the beginning of each bit time. Commands and data are sent most significant bit first (MSB first). For a detailed description of the communications protocol, refer to Section 15.2.2, "Communication Details."

If a host is attempting to communicate with a target MCU that has an unknown BDC clock rate, a SYNC command may be sent to the target MCU to request a timed sync response signal from which the host can determine the correct communication speed.

BKGD is a pseudo-open-drain pin and there is an on-chip pullup so no external pullup resistor is required. Unlike typical open-drain pins, the external RC time constant on this pin, which is influenced by external capacitance, plays almost no role in signal rise time. The custom protocol provides for brief, actively driven speedup pulses to force rapid rise times on this pin without risking harmful drive level conflicts. Refer to Section 15.2.2, "Communication Details," for more detail.



Num	С	Parameter	Symbol	Min	Typ <sup>1</sup>	Max	Unit
9	Ρ	High Impedance (off-state) leakage current <sup>2</sup>	I <sub>oz</sub>	—	0.01	1	μA
10	Ρ	Internal pullup resistors <sup>3</sup>	R <sub>PU</sub>	20	45	65	kΩ
11	Ρ	Internal pulldown resistors <sup>4</sup>		20	45	65	kΩ
12	С	Input Capacitance; all non-supply pins	C <sub>In</sub>	—	—	8	pF
13	Ρ	POR rearm voltage	V <sub>POR</sub>	0.9	1.4	2.0	V
14	D	POR rearm time	t <sub>POR</sub>	10	—	—	μs
15	Р	Low-voltage detection threshold — high range V <sub>DD</sub> falling V <sub>DD</sub> rising	V <sub>LVDH</sub>	4.2 4.3	4.3 4.4	4.4 4.5	v
16	Р	Low-voltage detection threshold — low range V <sub>DD</sub> falling V <sub>DD</sub> rising	V <sub>LVDL</sub>	2.48 2.54	2.56 2.62	2.64 2.7	v
17	Р	Low-voltage warning threshold — high range V <sub>DD</sub> falling V <sub>DD</sub> rising	V <sub>LVWH</sub>	4.2 4.3	4.3 4.4	4.4 4.5	v
18	Р	Low-voltage warning threshold — low range V <sub>DD</sub> falling V <sub>DD</sub> rising	V <sub>LVWL</sub>	2.48 2.54	2.56 2.62	2.64 2.7	v
19	Р	Low-voltage inhibit reset/recover hysteresis 5V 3V	V <sub>hys</sub>	_	100 60	_	mV
20	Ρ	Bandgap Voltage Reference Factory trimmed at V <sub>DD</sub> = 5.0 V Temp = 25 °C	V <sub>BG</sub>	1.185	1.20	1.215	V
21	D	dc injection current <sup>5, 6, 7, 8</sup> DC Injection Current Single pin limit $V_{IN} > V_{DD}$ $V_{IN} < V_{SS}$ Total MCU limit, includes sum of all stressed pins $V_{IN} > V_{DD}$ $V_{IN} < V_{SS}$	II <sub>IC</sub> I	0 0 0 0		2 -0.2 25 -5	mA mA mA mA

Table A-7. DC Characteristics (continued)

<sup>1</sup> Typical values are based on characterization data at 25°C unless otherwise stated.

- <sup>2</sup> Measured with  $V_{In} = V_{DD}$  or  $V_{SS}$ .
- <sup>3</sup> Measured with  $V_{In} = V_{SS}$ .
- <sup>4</sup> Measured with  $V_{In} = V_{DD}$ .
- <sup>5</sup> Power supply must maintain regulation within operating  $V_{DD}$  range during instantaneous and operating maximum current conditions. If positive injection current ( $V_{In} > V_{DD}$ ) is greater than  $I_{DD}$ , the injection current may flow out of  $V_{DD}$  and could result in external power supply going out of regulation. Ensure external  $V_{DD}$  load will shunt current greater than maximum injection current. This will be the greatest risk when the MCU is not consuming power. Examples are: if no system clock is present, or if clock rate is very low which (would reduce overall power consumption).
- $^{6}\,$  All functional non-supply pins are internally clamped to  $V_{SS}$  and  $V_{DD}$

<sup>7</sup> Input must be current limited to the value specified. To determine the value of the required current-limiting resistor, calculate resistance values for positive and negative clamp voltages, then use the larger of the two values.



# A.11 SPI Characteristics

Table A-15 and Figure A-16 through Figure A-19 describe the timing requirements for the SPI system.

Num <sup>1</sup>	С	Characteristic <sup>2</sup>		Symbol	Min	Max	Unit
		Operating frequency <sup>3</sup> Ma S	aster Slave	f <sub>op</sub> f <sub>op</sub>	f <sub>Bus</sub> /2048 dc	f <sub>Bus</sub> /2 f <sub>Bus</sub> /4	Hz
1		Cycle time Ma S	aster Slave	t <sub>SCK</sub> t <sub>SCK</sub>	2 4	2048 —	t <sub>cyc</sub> t <sub>cyc</sub>
2		Enable lead time Ma S	aster Slave	t <sub>Lead</sub> t <sub>Lead</sub>	 1/2	1/2	t <sub>SCK</sub> t <sub>SCK</sub>
3		Enable lag time Ma S	aster Slave	t <sub>Lag</sub> t <sub>Lag</sub>	 1/2	1/2	t <sub>SCK</sub> t <sub>SCK</sub>
4		Clock (SPSCK) high time Master and Slave		t <sub>scкн</sub>	1/2 t <sub>SCK</sub> – 25	_	ns
5		Clock (SPSCK) low time Ma and Slave	aster	t <sub>SCKL</sub>	1/2 t <sub>SCK</sub> – 25	_	ns
6		Data setup time (inputs) Ma S	aster Slave	t <sub>SI(M)</sub> t <sub>SI(S)</sub>	30 30		ns ns
7		Data hold time (inputs) Ma S	aster Slave	t <sub>HI(M)</sub> t <sub>HI(S)</sub>	30 30		ns ns
8		Access time, slave <sup>4</sup>		t <sub>A</sub>	0	40	ns
9		Disable time, slave <sup>5</sup>		t <sub>dis</sub>	—	40	ns
10		Data setup time (outputs) Ma S	aster Slave	t <sub>SO</sub> t <sub>SO</sub>	25 25		ns ns
11		Data hold time (outputs) Ma S	aster Slave	t <sub>HO</sub> t <sub>HO</sub>	-10 -10	_	ns ns

**Table A-15. SPI Electrical Characteristic** 

<sup>1</sup> Refer to Figure A-16 through Figure A-19.

 $^2\,$  All timing is shown with respect to 20%  $V_{DD}$  and 70%  $V_{DD}$ , unless noted; 100 pF load on all SPI pins. All timing assumes slew rate control disabled and high drive strength enabled for SPI output pins.

<sup>3</sup> Maximum baud rate must be limited to 5 MHz due to pad input characteristics.

<sup>4</sup> Time to data active from high-impedance state.

<sup>5</sup> Hold time to high-impedance state.











VIEW AA

© FREESCALE SEMICONDUCTOR, INC. ALL RIGHTS RESERVED.	MECHANICA	AL OUTLINE PRINT VERSION NOT TO SCALE			
TITLE: 64LD LQFP,	DOCUMENT NO: 98ASS23234W		REV: E		
10 X 10 X 1.4 PI	CASE NUMBER	2: 840F-02	11 AUG 2006		
O.5 PITCH, CASE OU	STANDARD: JEDEC MS-026 BCD				