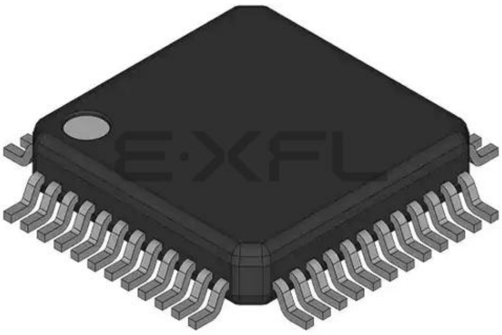

Welcome to E-XFL.COM

What is "Embedded - Microcontrollers"?

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

Applications of "Embedded - Microcontrollers"

Details	
Product Status	Active
Core Processor	S08
Core Size	8-Bit
Speed	40MHz
Connectivity	I²C, LINbus, SCI, SPI
Peripherals	LVD, POR, PWM, WDT
Number of I/O	39
Program Memory Size	32KB (32K x 8)
Program Memory Type	FLASH
EEPROM Size	1K x 8
RAM Size	1.5K x 8
Voltage - Supply (Vcc/Vdd)	2.7V ~ 5.5V
Data Converters	A/D 16x12b
Oscillator Type	External
Operating Temperature	-40°C ~ 105°C (TA)
Mounting Type	Surface Mount
Package / Case	48-LQFP
Supplier Device Package	48-LQFP (7x7)
Purchase URL	https://www.e-xfl.com/pro/item?MUrl=&PartUrl=mc9s08dn32avlf

Section Number	Title	Page

6.3.2 Edge and Level Sensitivity 82
6.3.3 Pull-up/Pull-down Resistors 82
6.3.4 Pin Interrupt Initialization 82
6.4 Pin Behavior in Stop Modes....82
6.5 Parallel I/O and Pin Control Registers 83
6.5.1 Port A Registers 84
6.5.2 Port B Registers 88
6.5.3 Port C Registers 92
6.5.4 Port D Registers 95
6.5.5 Port E Registers 99
6.5.6 Port F Registers 102
6.5.7 Port G Registers 105

Chapter 7
Central Processor Unit (S08CPUV3)

7.1 Introduction 109
7.1.1 Features 109
7.2 Programmer's Model and CPU Registers 110
7.2.1 Accumulator (A) 110
7.2.2 Index Register (H:X) 110
7.2.3 Stack Pointer (SP) 111
7.2.4 Program Counter (PC) 111
7.2.5 Condition Code Register (CCR) 111
7.3 Addressing Modes....113
7.3.1 Inherent Addressing Mode (INH) 113
7.3.2 Relative Addressing Mode (REL) 113
7.3.3 Immediate Addressing Mode (IMM) 113
7.3.4 Direct Addressing Mode (DIR) 113
7.3.5 Extended Addressing Mode (EXT) 114
7.3.6 Indexed Addressing Mode 114
7.4 Special Operations....115
7.4.1 Reset Sequence 115
7.4.2 Interrupt Sequence 115
7.4.3 Wait Mode Operation 116
7.4.4 Stop Mode Operation 116
7.4.5 BGND Instruction 117
7.5 HCS08 Instruction Set Summary 118

Chapter 8
Multi-Purpose Clock Generator (S08MCGV1)

8.1 Introduction 129
8.1.1 Features 131

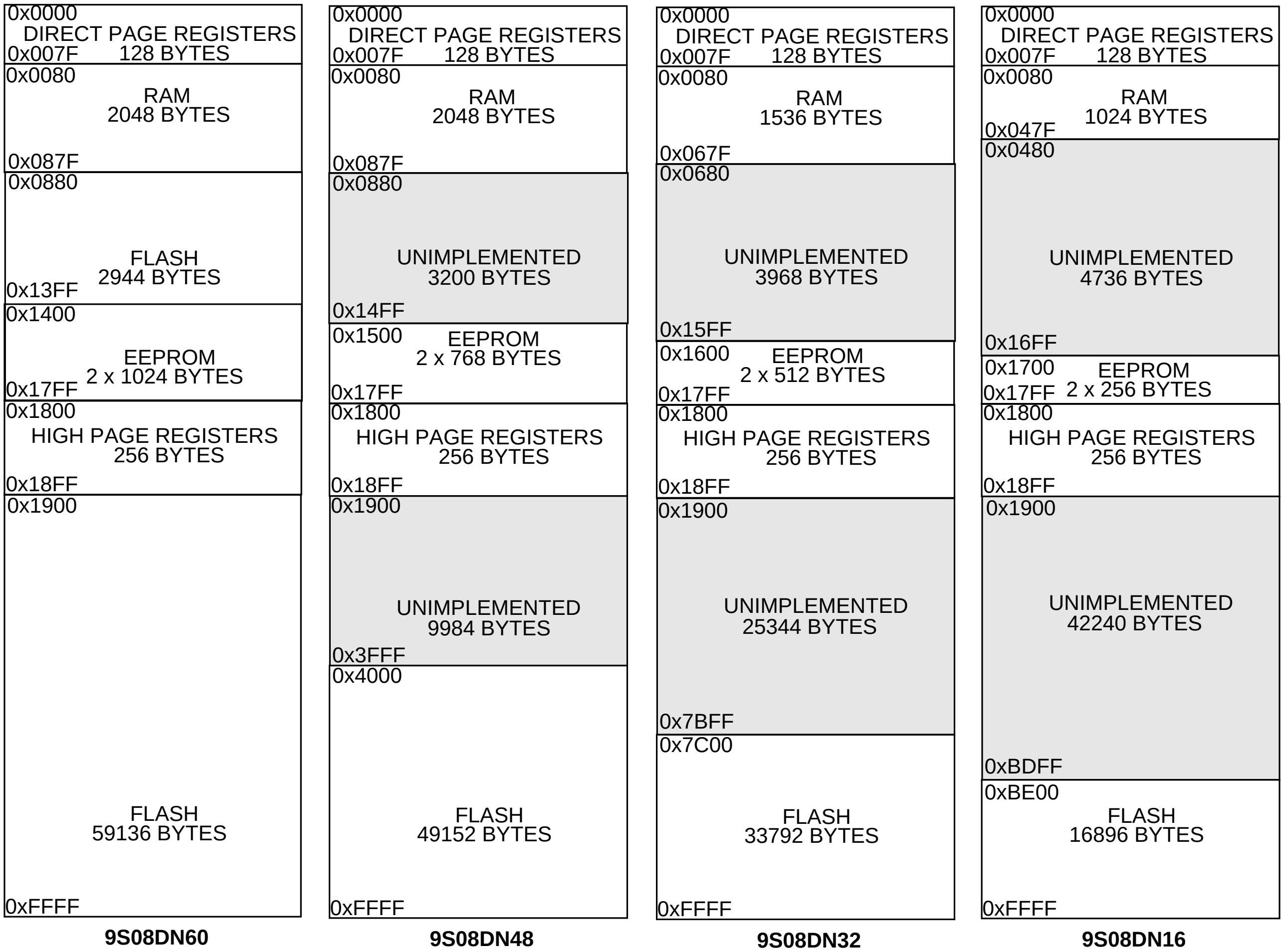

Figure 4-1. MC9S08DN60 Series Memory Map

4.2 Reset and Interrupt Vector Assignments

Table 4-1 shows address assignments for reset and interrupt vectors. The vector names shown in this table are the labels used in the MC9S08DN60 Series equate file provided by Freescale Semiconductor.

Table 4-1. Reset and Interrupt Vectors

Address (High/Low)	Vector	Vector Name
0xFFC0:0xFFC1	ACMP2	Vacmp2
0xFFC2:0xFFC3	ACMP1	Vacmp1
0xFFC4:0xFFCB	Reserved	—
0xFFCC:0xFFCD	RTC	Vrtc
0xFFCE:0xFFCF	IIC	Viic
0xFFD0:0xFFD1	ADC Conversion	Vadc

Table 4-4. Nonvolatile Register Summary

Address	Register Name	Bit 7	6	5	4	3	2	1	Bit 0
0xFFAE	Reserved for storage of FTRIM	0	0	0	0	0	0	0	FTRIM
0xFFAF	Res. for storage of MCGTRM	TRIM							
0xFFB0–0xFFB7	**NVBACKKEY**	8-Byte Comparison Key							
0xFFB8–0xFFBC	Reserved	— —	— —	— —	— —	— —	— —	— —	— —
0xFFBD	**NVPROT**	EPS		FPS					
0xFFBE	Reserved	—	—	—	—	—	—	—	—
0xFFBF	**NVOPT**	KEYEN	FNORED	EPGMOD	0	0	0	SEC	

Provided the key enable (KEYEN) bit is 1, the 8-byte comparison key can be used to temporarily disengage memory security. This key mechanism can be accessed only through user code running in secure memory. (A security key cannot be entered directly through background debug commands.) This security key can be disabled completely by programming the KEYEN bit to 0. If the security key is disabled, the only way to disengage security is by mass erasing the Flash if needed (normally through the background debug interface) and verifying that Flash is blank. To avoid returning to secure mode after the next reset, program the security bits (SEC) to the unsecured state (1:0).

Table 4-6. FCDIV Register Field Descriptions

Field	Description
7 DIVLD	**Divisor Loaded Status Flag** — When set, this read-only status flag indicates that the FCDIV register has been written since reset. Reset clears this bit and the first write to this register causes this bit to become set regardless of the data written. 0 FCDIV has not been written since reset; erase and program operations disabled for Flash and EEPROM. 1 FCDIV has been written since reset; erase and program operations enabled for Flash and EEPROM.
6 PRDIV8	**Prescale (Divide) Flash and EEPROM Clock by 8** (This bit is write once.) 0 Clock input to the Flash and EEPROM clock divider is the bus rate clock. 1 Clock input to the Flash and EEPROM clock divider is the bus rate clock divided by 8.
5:0 DIV	**Divisor for Flash and EEPROM Clock Divider** — These bits are write once. The Flash and EEPROM clock divider divides the bus rate clock (or the bus rate clock divided by 8 if PRDIV8 = 1) by the value in the 6-bit DIV field plus one. The resulting frequency of the internal Flash and EEPROM clock must fall within the range of 200 kHz to 150 kHz for proper Flash and EEPROM operations. Program/Erase timing pulses are one cycle of this internal Flash and EEPROM clock which corresponds to a range of 5 µs to 6.7 µs. The automated programming logic uses an integer number of these pulses to complete an erase or program operation. See Equation 4-1 and Equation 4-2.

$$\text{if PRDIV8} = 0 \text{ — } f_{FCLK} = f_{Bus} \div (DIV + 1) \qquad \textit{Eqn. 4-1}$$

$$\text{if PRDIV8} = 1 \text{ — } f_{FCLK} = f_{Bus} \div (8 \times (DIV + 1)) \qquad \textit{Eqn. 4-2}$$

Table 4-7 shows the appropriate values for PRDIV8 and DIV for selected bus frequencies.

Table 4-7. Flash and EEPROM Clock Divider Settings

f_{Bus}	PRDIV8 (Binary)	DIV (Decimal)	f_{FCLK}	Program/Erase Timing Pulse (5 µs Min, 6.7 µs Max)
20 MHz	1	12	192.3 kHz	5.2 µs
10 MHz	0	49	200 kHz	5 µs
8 MHz	0	39	200 kHz	5 µs
4 MHz	0	19	200 kHz	5 µs
2 MHz	0	9	200 kHz	5 µs
1 MHz	0	4	200 kHz	5 µs
200 kHz	0	0	200 kHz	5 µs
150 kHz	0	0	150 kHz	6.7 µs

4.5.11.2 Flash and EEPROM Options Register (FOPT and NVOPT)

During reset, the contents of the nonvolatile location NVOPT are copied from Flash into FOPT. To change the value in this register, erase and reprogram the NVOPT location in Flash memory as usual and then issue a new MCU reset.

Table 4-13. Flash Block Protection (continued)

FPS	Address Area Protected	Memory Size Protected (bytes)	Number of Sectors Protected
0x1B	0x2800–0xFFFF	54K	72
0x1A	0x2200–0xFFFF	55.5K	74
0x19	0x1C00–0xFFFF	57K	76
0x18–0x00	0x0000–0xFFFF	64K	86

4.5.11.5 Flash and EEPROM Status Register (FSTAT)

	7	6	5	4	3	2	1	0
R	FCBEF	FCCF	FPVIOL	FACCERR	0	FBLANK	0	0
W								
Reset	1	1	0	0	0	0	0	0

= Unimplemented or Reserved

Figure 4-9. Flash and EEPROM Status Register (FSTAT)

Table 4-14. FSTAT Register Field Descriptions

Field	Description
7 FCBEF	**Command Buffer Empty Flag** — The FCBEF bit is used to launch commands. It also indicates that the command buffer is empty so that a new command sequence can be executed when performing burst programming. The FCBEF bit is cleared by writing a 1 to it or when a burst program command is transferred to the array for programming. Only burst program commands can be buffered. 0 Command buffer is full (not ready for additional commands). 1 A new burst program command can be written to the command buffer.
6 FCCF	**Command Complete Flag** — FCCF is set automatically when the command buffer is empty and no command is being processed. FCCF is cleared automatically when a new command is started (by writing 1 to FCBEF to register a command). Writing to FCCF has no meaning or effect. 0 Command in progress 1 All commands complete
5 FPVIOL	**Protection Violation Flag** — FPVIOL is set automatically when a command that attempts to erase or program a location in a protected block is launched (the erroneous command is ignored). FPVIOL is cleared by writing a 1 to FPVIOL. 0 No protection violation. 1 An attempt was made to erase or program a protected location.

Table 5-1. Vector Summary[1]

Vector No.	Address (High/Low)	Vector Name	Module	Source	Enable	Description
31	0xFFC0/0xFFC1	Vacmp2	ACMP2	ACF	ACIE	Analog comparator 2
30	0xFFC2/0xFFC3	Vacmp1	ACMP1	ACF	ACIE	Analog comparator 1
29–26	0xFFC4/0xFFC5– 0xFFCA/0xFFCB					(Reserved)
25	0xFFCC/0xFFCD	Vrtc	RTC	RTIF	RTIE	Real-time interrupt
24	0xFFCE/0xFFCF	Viic	IIC	IICIS	IICIE	IIC control
23	0xFFD0/0xFFD1	Vadc	ADC	COCO	AIEN	ADC
22	0xFFD2/0xFFD3	Vport	Port A,B,D	PTAIF, PTBIF, PTDIF	PTAIE, PTBIE, PTDIE	Port Pins
21–19	0xFFD4/0xFFD5– 0xFFD8/0xFFD9					(Reserved)
18	0xFFDA/0xFFDB	Vsci1tx	SCI1	TDRE, TC	TIE, TCIE	SCI1 transmit
17	0xFFDC/0xFFDD	Vsci1rx	SCI1	IDLE, LBKDIF, RDRF, RXEDGIF	ILIE, LBKDIE, RIE, RXEDGIE	SCI1 receive
16	0xFFDE/0xFFDF	Vsci1err	SCI1	OR, NF, FE, PF	ORIE, NFIE, FEIE, PFIE	SCI1 error
15	0xFFE0/0xFFE1	Vspi	SPI	SPIF, MODF, SPTEF	SPIE, SPIE, SPTIE	SPI
14	0xFFE2/0xFFE3	Vtpm2ovf	TPM2	TOF	TOIE	TPM2 overflow
13	0xFFE4/0xFFE5	Vtpm2ch1	TPM2	CH1F	CH1IE	TPM2 channel 1
12	0xFFE6/0xFFE7	Vtpm2ch0	TPM2	CH0F	CH0IE	TPM2 channel 0
11	0xFFE8/0xFFE9	Vtpm1ovf	TPM1	TOF	TOIE	TPM1 overflow
10	0xFFEA/0xFFEB	Vtpm1ch5	TPM1	CH5F	CH5IE	TPM1 channel 5
9	0xFFEC/0xFFED	Vtpm1ch4	TPM1	CH4F	CH4IE	TPM1 channel 4
8	0xFFEE/0xFFEF	Vtpm1ch3	TPM1	CH3F	CH3IE	TPM1 channel 3
7	0xFFF0/0xFFF1	Vtpm1ch2	TPM1	CH2F	CH2IE	TPM1 channel 2
6	0xFFF2/0xFFF3	Vtpm1ch1	TPM1	CH1F	CH1IE	TPM1 channel 1
5	0xFFF4/0xFFF5	Vtpm1ch0	TPM1	CH0F	CH0IE	TPM1 channel 0
4	0xFFF6/0xFFF7	Vlol	MCG	LOLS	LOLIE	MCG loss of lock
3	0xFFF8/0xFFF9	Vlvd	System control	LVWF	LVWIE	Low-voltage warning
2	0xFFFA/0xFFFB	Virq	IRQ	IRQF	IRQIE	IRQ pin
1	0xFFFC/0xFFFD	Vswi	Core	SWI Instruction	—	Software interrupt
0	0xFFFE/0xFFFF	Vreset	System control	COP, LOC, LVD, $\overline{\text{RESET}}$, ILOP, ILAD, POR, BDFR	COPE CME LVDRE — — — — —	Watchdog timer Loss-of-clock Low-voltage detect External pin Illegal opcode Illegal address Power-on-reset BDM-forced reset

[1] Vector priority is shown from lowest (first row) to highest (last row). For example, Vreset is the highest priority vector.

5.6 Low-Voltage Detect (LVD) System

The MC9S08DN60 Series includes a system to protect against low-voltage conditions in order to protect memory contents and control MCU system states during supply voltage variations. The system is

In general, whenever a pin is shared with both an alternate digital function and an analog function, the analog function has priority such that if both the digital and analog functions are enabled, the analog function controls the pin.

It is a good programming practice to write to the port data register before changing the direction of a port pin to become an output. This ensures that the pin will not be driven momentarily with an old data value that happened to be in the port data register.

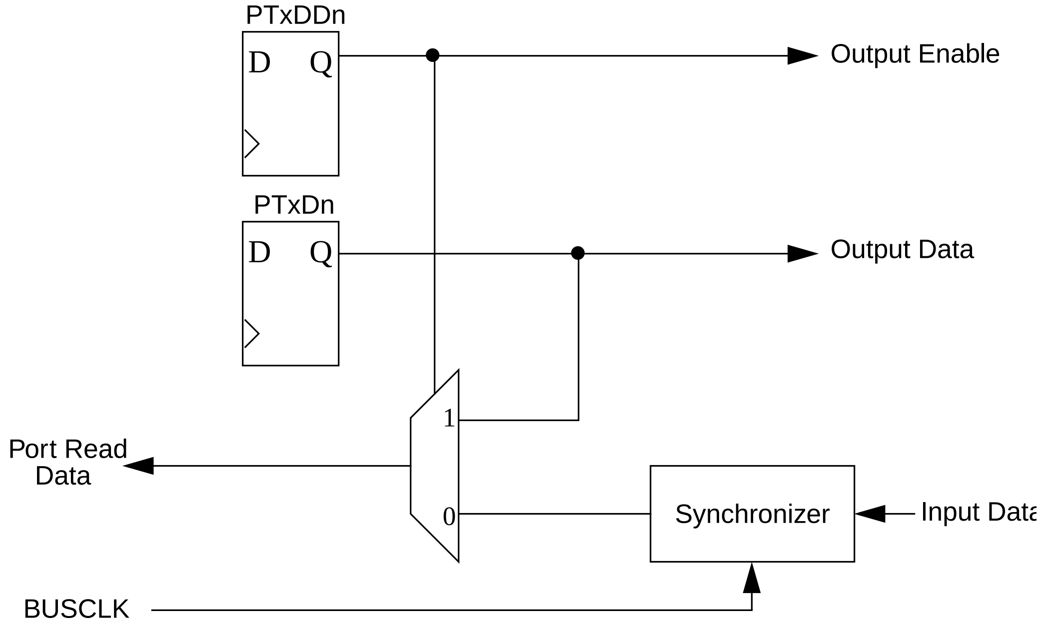

Figure 6-1. Parallel I/O Block Diagram

6.2 Pull-up, Slew Rate, and Drive Strength

Associated with the parallel I/O ports is a set of registers located in the high page register space that operate independently of the parallel I/O registers. These registers are used to control pull-ups, slew rate, and drive strength for the pins.

An internal pull-up device can be enabled for each port pin by setting the corresponding bit in the pull-up enable register (PTxPEn). The pull-up device is disabled if the pin is configured as an output by the parallel I/O control logic or any shared peripheral function regardless of the state of the corresponding pull-up enable register bit. The pull-up device is also disabled if the pin is controlled by an analog function.

Slew rate control can be enabled for each port pin by setting the corresponding bit in the slew rate control register (PTxSEn). When enabled, slew control limits the rate at which an output can transition in order to reduce EMC emissions. Slew rate control has no effect on pins that are configured as inputs.

NOTE

Slew rate reset default values may differ between engineering samples and final production parts. Always initialize slew rate control to the desired value to ensure correct operation.

6.5.1.7 Port A Interrupt Pin Select Register (PTAPS)

	7	6	5	4	3	2	1	0
R W	PTAPS7	PTAPS6	PTAPS5	PTAPS4	PTAPS3	PTAPS2	PTAPS1	PTAPS0
Reset:	0	0	0	0	0	0	0	0

Figure 6-9. Port A Interrupt Pin Select Register (PTAPS)

Table 6-7. PTAPS Register Field Descriptions

Field	Description
7:0 PTAPS[7:0]	**Port A Interrupt Pin Selects** — Each of the PTAPSn bits enable the corresponding port A interrupt pin. 0 Pin not enabled as interrupt. 1 Pin enabled as interrupt.

6.5.1.8 Port A Interrupt Edge Select Register (PTAES)

	7	6	5	4	3	2	1	0
R W	PTAES7	PTAES6	PTAES5	PTAES4	PTAES3	PTAES2	PTAES1	PTAES0
Reset:	0	0	0	0	0	0	0	0

Figure 6-10. Port A Edge Select Register (PTAES)

Table 6-8. PTAES Register Field Descriptions

Field	Description
7:0 PTAES[7:0]	**Port A Edge Selects** — Each of the PTAESn bits serves a dual purpose by selecting the polarity of the active interrupt edge as well as selecting a pull-up or pull-down device if enabled. 0 A pull-up device is connected to the associated pin and detects falling edge/low level for interrupt generation. 1 A pull-down device is connected to the associated pin and detects rising edge/high level for interrupt generation.

6.5.6.5 Port F Drive Strength Selection Register (PTFDS)

	7	6	5	4	3	2	1	0
R W	PTFDS7	PTFDS6	PTFDS5	PTFDS4	PTFDS3	PTFDS2	PTFDS1	PTFDS0
Reset:	0	0	0	0	0	0	0	0

Figure 6-41. Drive Strength Selection for Port F Register (PTFDS)

Table 6-39. PTFDS Register Field Descriptions

Field	Description
7:0 PTFDS[7:0]	**Output Drive Strength Selection for Port F Bits** — Each of these control bits selects between low and high output drive for the associated PTF pin. For port F pins that are configured as inputs, these bits have no effect. 0 Low output drive strength selected for port F bit n. 1 High output drive strength selected for port F bit n.

7.4.5 BGND Instruction

The BGND instruction is new to the HCS08 compared to the M68HC08. BGND would not be used in normal user programs because it forces the CPU to stop processing user instructions and enter the active background mode. The only way to resume execution of the user program is through reset or by a host debug system issuing a GO, TRACE1, or TAGGO serial command through the background debug interface.

Software-based breakpoints can be set by replacing an opcode at the desired breakpoint address with the BGND opcode. When the program reaches this breakpoint address, the CPU is forced to active background mode rather than continuing the user program.

The nine states of the MCG are shown as a state diagram and are described below. The arrows indicate the allowed movements between the states.

8.4.1.1 FLL Engaged Internal (FEI)

FLL engaged internal (FEI) is the default mode of operation and is entered when all the following conditions occur:

- CLKS bits are written to 00
- IREFS bit is written to 1
- PLLS bit is written to 0
- RDIV bits are written to 000. Since the internal reference clock frequency should already be in the range of 31.25 kHz to 39.0625 kHz after it is trimmed, no further frequency divide is necessary.

In FLL engaged internal mode, the MCGOUT clock is derived from the FLL clock, which is controlled by the internal reference clock. The FLL clock frequency locks to 1024 times the reference frequency, as selected by the RDIV bits. The MCGLCLK is derived from the FLL and the PLL is disabled in a low power state.

8.4.1.2 FLL Engaged External (FEE)

The FLL engaged external (FEE) mode is entered when all the following conditions occur:

- CLKS bits are written to 00
- IREFS bit is written to 0
- PLLS bit is written to 0
- RDIV bits are written to divide reference clock to be within the range of 31.25 kHz to 39.0625 kHz

In FLL engaged external mode, the MCGOUT clock is derived from the FLL clock which is controlled by the external reference clock. The external reference clock which is enabled can be an external crystal/resonator or it can be another external clock source.The FLL clock frequency locks to 1024 times the reference frequency, as selected by the RDIV bits. The MCGLCLK is derived from the FLL and the PLL is disabled in a low power state.

8.4.1.3 FLL Bypassed Internal (FBI)

In FLL bypassed internal (FBI) mode, the MCGOUT clock is derived from the internal reference clock and the FLL is operational but its output clock is not used. This mode is useful to allow the FLL to acquire its target frequency while the MCGOUT clock is driven from the internal reference clock.

The FLL bypassed internal mode is entered when all the following conditions occur:

- CLKS bits are written to 01
- IREFS bit is written to 1
- PLLS bit is written to 0
- RDIV bits are written to 000. Since the internal reference clock frequency should already be in the range of 31.25 kHz to 39.0625 kHz after it is trimmed, no further frequency divide is necessary.

- LP bit is written to 0

In FLL bypassed internal mode, the MCGOUT clock is derived from the internal reference clock. The FLL clock is controlled by the internal reference clock, and the FLL clock frequency locks to 1024 times the reference frequency, as selected by the RDIV bits. The MCGLCLK is derived from the FLL and the PLL is disabled in a low power state.

8.4.1.4 FLL Bypassed External (FBE)

In FLL bypassed external (FBE) mode, the MCGOUT clock is derived from the external reference clock and the FLL is operational but its output clock is not used. This mode is useful to allow the FLL to acquire its target frequency while the MCGOUT clock is driven from the external reference clock.

The FLL bypassed external mode is entered when all the following conditions occur:

- CLKS bits are written to 10
- IREFS bit is written to 0
- PLLS bit is written to 0
- RDIV bits are written to divide reference clock to be within the range of 31.25 kHz to 39.0625 kHz
- LP bit is written to 0

In FLL bypassed external mode, the MCGOUT clock is derived from the external reference clock. The external reference clock which is enabled can be an external crystal/resonator or it can be another external clock source.The FLL clock is controlled by the external reference clock, and the FLL clock frequency locks to 1024 times the reference frequency, as selected by the RDIV bits. The MCGLCLK is derived from the FLL and the PLL is disabled in a low power state.

NOTE

It is possible to briefly operate in FBE mode with an FLL reference clock frequency that is greater than the specified maximum frequency. This can be necessary in applications that operate in PEE mode using an external crystal with a frequency above 5 MHz. Please see 8.5.2.4, “Example # 4: Moving from FEI to PEE Mode: External Crystal = 8 MHz, Bus Frequency = 8 MHz for a detailed example.

8.4.1.5 PLL Engaged External (PEE)

The PLL engaged external (PEE) mode is entered when all the following conditions occur:

- CLKS bits are written to 00
- IREFS bit is written to 0
- PLLS bit is written to 1
- RDIV bits are written to divide reference clock to be within the range of 1 MHz to 2 MHz

In PLL engaged external mode, the MCGOUT clock is derived from the PLL clock which is controlled by the external reference clock. The external reference clock which is enabled can be an external crystal/resonator or it can be another external clock source The PLL clock frequency locks to a

Table 9-3. ACMPxSC Field Descriptions (continued)

Field	Description
3 ACO	Analog Comparator Output. Reading ACO returns the current value of the analog comparator output. ACO is reset to a 0 and reads as a 0 when the ACMP is disabled (ACME = 0).
2 ACOPE	Analog Comparator Output Pin Enable. Enables the comparator output to be placed onto the external pin, ACMPxO. 0 Analog comparator output not available on ACMPxO 1 Analog comparator output is driven out on ACMPxO
1:0 ACMOD	Analog Comparator Mode. ACMOD selects the type of compare event which sets ACF. 00 Encoding 0 — Comparator output falling edge 01 Encoding 1 — Comparator output rising edge 10 Encoding 2 — Comparator output falling edge 11 Encoding 3 — Comparator output rising or falling edge

9.4 Functional Description

The analog comparator can compare two analog input voltages applied to ACMPx+ and ACMPx–, or it can compare an analog input voltage applied to ACMPx– with an internal bandgap reference voltage. ACBGS selects between the bandgap reference voltage or the ACMPx+ pin as the input to the non-inverting input of the analog comparator. The comparator output is high when the non-inverting input is greater than the inverting input, and is low when the non-inverting input is less than the inverting input. ACMOD selects the condition that causes ACF to be set. ACF can be set on a rising edge of the comparator output, a falling edge of the comparator output, or a rising or a falling edge (toggle). The comparator output can be read directly through ACO. The comparator output can be driven onto the ACMPxO pin using ACOPE.

	7	6	5	4	3	2	1	0
R	COCO	AIEN	ADCO	ADCH				
W								
Reset:	0	0	0	1	1	1	1	1

Figure 10-3. Status and Control Register (ADCSC1)

Table 10-3. ADCSC1 Field Descriptions

Field	Description
7 COCO	Conversion Complete Flag. The COCO flag is a read-only bit set each time a conversion is completed when the compare function is disabled (ACFE = 0). When the compare function is enabled (ACFE = 1), the COCO flag is set upon completion of a conversion only if the compare result is true. This bit is cleared when ADCSC1 is written or when ADCRL is read. 0 Conversion not completed 1 Conversion completed
6 AIEN	Interrupt Enable AIEN enables conversion complete interrupts. When COCO becomes set while AIEN is high, an interrupt is asserted. 0 Conversion complete interrupt disabled 1 Conversion complete interrupt enabled
5 ADCO	Continuous Conversion Enable. ADCO enables continuous conversions. 0 One conversion following a write to the ADCSC1 when software triggered operation is selected, or one conversion following assertion of ADHWT when hardware triggered operation is selected. 1 Continuous conversions initiated following a write to ADCSC1 when software triggered operation is selected. Continuous conversions are initiated by an ADHWT event when hardware triggered operation is selected.
4:0 ADCH	Input Channel Select. The ADCH bits form a 5-bit field that selects one of the input channels. The input channels are detailed in Table 10-4. The successive approximation converter subsystem is turned off when the channel select bits are all set. This feature allows for explicit disabling of the ADC and isolation of the input channel from all sources. Terminating continuous conversions this way prevents an additional, single conversion from being performed. It is not necessary to set the channel select bits to all ones to place the ADC in a low-power state when continuous conversions are not enabled because the module automatically enters a low-power state when a conversion completes.

Table 10-4. Input Channel Select

ADCH	Input Select
00000–01111	AD0–15
10000–11011	AD16–27
11100	Reserved
11101	V_{REFH}
11110	V_{REFL}
11111	Module disabled

12.5 Functional Description

An SPI transfer is initiated by checking for the SPI transmit buffer empty flag (SPTEF = 1) and then writing a byte of data to the SPI data register (SPID) in the master SPI device. When the SPI shift register is available, this byte of data is moved from the transmit data buffer to the shifter, SPTEF is set to indicate there is room in the buffer to queue another transmit character if desired, and the SPI serial transfer starts.

During the SPI transfer, data is sampled (read) on the MISO pin at one SPSCK edge and shifted, changing the bit value on the MOSI pin, one-half SPSCK cycle later. After eight SPSCK cycles, the data that was in the shift register of the master has been shifted out the MOSI pin to the slave while eight bits of data were shifted in the MISO pin into the master's shift register. At the end of this transfer, the received data byte is moved from the shifter into the receive data buffer and SPRF is set to indicate the data can be read by reading SPID. If another byte of data is waiting in the transmit buffer at the end of a transfer, it is moved into the shifter, SPTEF is set, and a new transfer is started.

Normally, SPI data is transferred most significant bit (MSB) first. If the least significant bit first enable (LSBFE) bit is set, SPI data is shifted LSB first.

When the SPI is configured as a slave, its $\overline{SS}$ pin must be driven low before a transfer starts and $\overline{SS}$ must stay low throughout the transfer. If a clock format where CPHA = 0 is selected, $\overline{SS}$ must be driven to a logic 1 between successive transfers. If CPHA = 1, $\overline{SS}$ may remain low between successive transfers. See Section 12.5.1, "SPI Clock Formats" for more details.

Because the transmitter and receiver are double buffered, a second byte, in addition to the byte currently being shifted out, can be queued into the transmit data buffer, and a previously received character can be in the receive data buffer while a new character is being shifted in. The SPTEF flag indicates when the transmit buffer has room for a new character. The SPRF flag indicates when a received character is available in the receive data buffer. The received character must be read out of the receive buffer (read SPID) before the next transfer is finished or a receive overrun error results.

In the case of a receive overrun, the new data is lost because the receive buffer still held the previous character and was not ready to accept the new data. There is no indication for such an overrun condition so the application system designer must ensure that previous data has been read from the receive buffer before a new transfer is initiated.

12.5.1 SPI Clock Formats

To accommodate a wide variety of synchronous serial peripherals from different manufacturers, the SPI system has a clock polarity (CPOL) bit and a clock phase (CPHA) control bit to select one of four clock formats for data transfers. CPOL selectively inserts an inverter in series with the clock. CPHA chooses between two different clock phase relationships between the clock and data.

Figure 12-10 shows the clock formats when CPHA = 1. At the top of the figure, the eight bit times are shown for reference with bit 1 starting at the first SPSCK edge and bit 8 ending one-half SPSCK cycle after the sixteenth SPSCK edge. The MSB first and LSB first lines show the order of SPI data bits depending on the setting in LSBFE. Both variations of SPSCK polarity are shown, but only one of these waveforms applies for a specific transfer, depending on the value in CPOL. The SAMPLE IN waveform applies to the MOSI input of a slave or the MISO input of a master. The MOSI waveform applies to the MOSI output

Instead of hardware interrupts, software polling may be used to monitor the TDRE and TC status flags if the corresponding TIE or TCIE local interrupt masks are 0s.

When a program detects that the receive data register is full (RDRF = 1), it gets the data from the receive data register by reading SCI1D. The RDRF flag is cleared by reading SCI1S1 while RDRF = 1 and then reading SCI1D.

When polling is used, this sequence is naturally satisfied in the normal course of the user program. If hardware interrupts are used, SCI1S1 must be read in the interrupt service routine (ISR). Normally, this is done in the ISR anyway to check for receive errors, so the sequence is automatically satisfied.

The IDLE status flag includes logic that prevents it from getting set repeatedly when the RxD line remains idle for an extended period of time. IDLE is cleared by reading SCI1S1 while IDLE = 1 and then reading SCI1D. After IDLE has been cleared, it cannot become set again until the receiver has received at least one new character and has set RDRF.

If the associated error was detected in the received character that caused RDRF to be set, the error flags — noise flag (NF), framing error (FE), and parity error flag (PF) — get set at the same time as RDRF. These flags are not set in overrun cases.

If RDRF was already set when a new character is ready to be transferred from the receive shifter to the receive data buffer, the overrun (OR) flag gets set instead the data along with any associated NF, FE, or PF condition is lost.

At any time, an active edge on the RxD serial data input pin causes the RXEDGIF flag to set. The RXEDGIF flag is cleared by writing a “1” to it. This function does depend on the receiver being enabled (RE = 1).

13.3.5 Additional SCI Functions

The following sections describe additional SCI functions.

13.3.5.1 8- and 9-Bit Data Modes

The SCI system (transmitter and receiver) can be configured to operate in 9-bit data mode by setting the M control bit in SCI1C1. In 9-bit mode, there is a ninth data bit to the left of the MSB of the SCI data register. For the transmit data buffer, this bit is stored in T8 in SCI1C3. For the receiver, the ninth bit is held in R8 in SCI1C3.

For coherent writes to the transmit data buffer, write to the T8 bit before writing to SCI1D.

If the bit value to be transmitted as the ninth bit of a new character is the same as for the previous character, it is not necessary to write to T8 again. When data is transferred from the transmit data buffer to the transmit shifter, the value in T8 is copied at the same time data is transferred from SCI1D to the shifter.

9-bit data mode typically is used in conjunction with parity to allow eight bits of data plus the parity in the ninth bit. Or it is used with address-mark wakeup so the ninth data bit can serve as the wakeup bit. In custom protocols, the ninth bit can also serve as a software-controlled marker.

Table 15-7. TPM Clock Source Selection

CLKSB:CLKSA	TPM Clock Source to Prescaler Input
00	No clock selected (TPM counter disabled)
01	Bus rate clock
10	Fixed system clock
11	External source

The bus rate clock is the main system bus clock for the MCU. This clock source requires no synchronization because it is the clock that is used for all internal MCU activities including operation of the CPU and buses.

In MCUs that have no PLL and FLL or the PLL and FLL are not engaged, the fixed system clock source is the same as the bus-rate-clock source, and it does not go through a synchronizer. When a PLL or FLL is present and engaged, a synchronizer is required between the crystal divided-by two clock source and the timer counter so counter transitions will be properly aligned to bus-clock transitions. A synchronizer will be used at chip level to synchronize the crystal-related source clock to the bus clock.

The external clock source may be connected to any TPM channel pin. This clock source always has to pass through a synchronizer to assure that counter transitions are properly aligned to bus clock transitions. The bus-rate clock drives the synchronizer; therefore, to meet Nyquist criteria even with jitter, the frequency of the external clock source must not be faster than the bus rate divided-by four. With ideal clocks the external clock can be as fast as bus clock divided by four.

When the external clock source shares the TPM channel pin, this pin should not be used for other channel timing functions. For example, it would be ambiguous to configure channel 0 for input capture when the TPM channel 0 pin was also being used as the timer external clock source. (It is the user's responsibility to avoid such settings.) The TPM channel could still be used in output compare mode for software timing functions (pin controls set not to affect the TPM channel pin).

15.4.1.2 Counter Overflow and Modulo Reset

An interrupt flag and enable are associated with the 16-bit main counter. The flag (TOF) is a software-accessible indication that the timer counter has overflowed. The enable signal selects between software polling (TOIE=0) where no hardware interrupt is generated, or interrupt-driven operation (TOIE=1) where a static hardware interrupt is generated whenever the TOF flag is equal to one.

The conditions causing TOF to become set depend on whether the TPM is configured for center-aligned PWM (CPWMS=1). In the simplest mode, there is no modulus limit and the TPM is not in CPWMS=1 mode. In this case, the 16-bit timer counter counts from 0x0000 through 0xFFFF and overflows to 0x0000 on the next counting clock. TOF becomes set at the transition from 0xFFFF to 0x0000. When a modulus limit is set, TOF becomes set at the transition from the value set in the modulus register to 0x0000. When the TPM is in center-aligned PWM mode (CPWMS=1), the TOF flag gets set as the counter changes direction at the end of the count value set in the modulus register (that is, at the transition from the value set in the modulus register to the next lower count value). This corresponds to the end of a PWM period (the 0x0000 count value corresponds to the center of a period).

Figure 16-3 shows the host receiving a logic 1 from the target HCS08 MCU. Because the host is asynchronous to the target MCU, there is a 0-to-1 cycle delay from the host-generated falling edge on BKGD to the perceived start of the bit time in the target MCU. The host holds the BKGD pin low long enough for the target to recognize it (at least two target BDC cycles). The host must release the low drive before the target MCU drives a brief active-high speedup pulse seven cycles after the perceived start of the bit time. The host should sample the bit level about 10 cycles after it started the bit time.

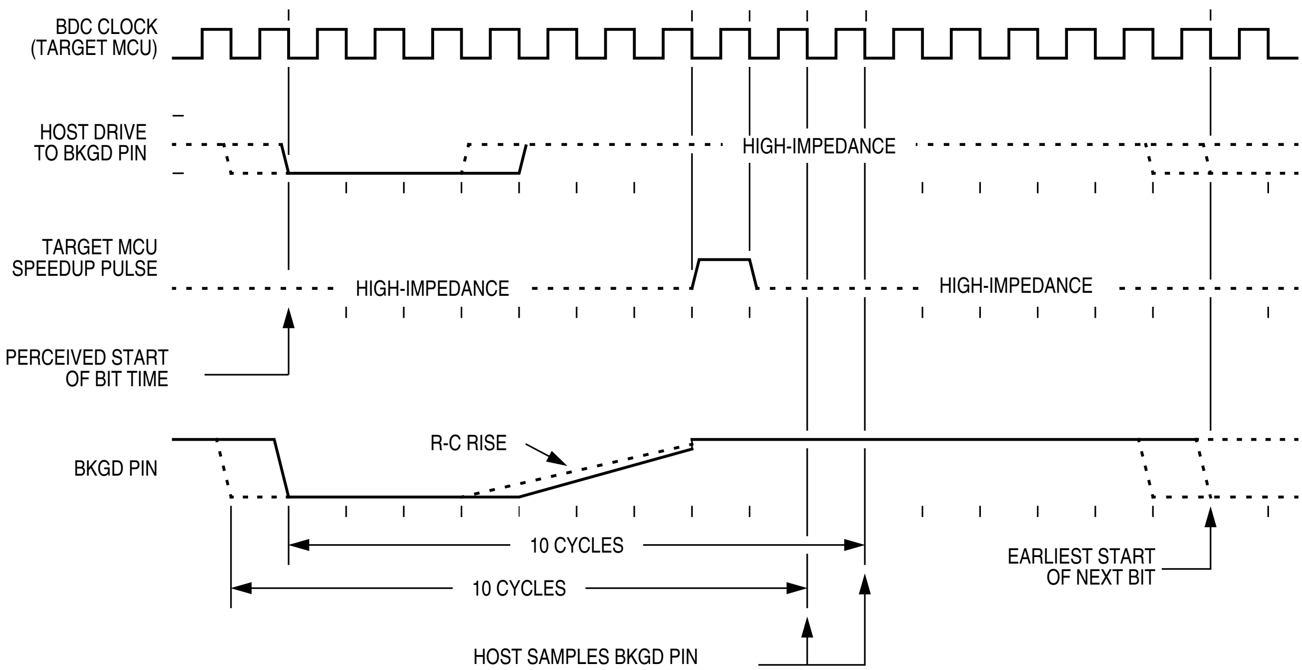

Figure 16-3. BDC Target-to-Host Serial Bit Timing (Logic 1)

A-Only — Trigger when the address matches the value in comparator A

A OR B — Trigger when the address matches either the value in comparator A or the value in comparator B

A Then B — Trigger when the address matches the value in comparator B but only after the address for another cycle matched the value in comparator A. There can be any number of cycles after the A match and before the B match.

A AND B Data (Full Mode) — This is called a full mode because address, data, and R/W (optionally) must match within the same bus cycle to cause a trigger event. Comparator A checks address, the low byte of comparator B checks data, and R/W is checked against RWA if RWAEN = 1. The high-order half of comparator B is not used.

In full trigger modes it is not useful to specify a tag-type CPU breakpoint (BRKEN = TAG = 1), but if you do, the comparator B data match is ignored for the purpose of issuing the tag request to the CPU and the CPU breakpoint is issued when the comparator A address matches.

A AND NOT B Data (Full Mode) — Address must match comparator A, data must not match the low half of comparator B, and R/W must match RWA if RWAEN = 1. All three conditions must be met within the same bus cycle to cause a trigger.

In full trigger modes it is not useful to specify a tag-type CPU breakpoint (BRKEN = TAG = 1), but if you do, the comparator B data match is ignored for the purpose of issuing the tag request to the CPU and the CPU breakpoint is issued when the comparator A address matches.

Event-Only B (Store Data) — Trigger events occur each time the address matches the value in comparator B. Trigger events cause the data to be captured into the FIFO. The debug run ends when the FIFO becomes full.

A Then Event-Only B (Store Data) — After the address has matched the value in comparator A, a trigger event occurs each time the address matches the value in comparator B. Trigger events cause the data to be captured into the FIFO. The debug run ends when the FIFO becomes full.

Inside Range (A ≤ Address ≤ B) — A trigger occurs when the address is greater than or equal to the value in comparator A and less than or equal to the value in comparator B at the same time.

Outside Range (Address < A or Address > B) — A trigger occurs when the address is either less than the value in comparator A or greater than the value in comparator B.

	7	6	5	4	3	2	1	0
R	Bit 7	6	5	4	3	2	1	Bit 0
W	Any write to TPMxCNTL clears the 16-bit counter.							
Reset	0	0	0	0	0	0	0	0

Figure B-4. Timer Counter Register Low (TPMxCNTL)

When background mode is active, the timer counter and the coherency mechanism are frozen such that the buffer latches remain in the state they were in when the background mode became active even if one or both bytes of the counter are read while background mode is active.

B.2.3 Timer Counter Modulo Registers (TPMxMODH:TPMxMODL)

The read/write TPM modulo registers contain the modulo value for the TPM counter. After the TPM counter reaches the modulo value, the TPM counter resumes counting from 0x0000 at the next clock (CPWMS = 0) or starts counting down (CPWMS = 1), and the overflow flag (TOF) becomes set. Writing to TPMxMODH or TPMxMODL inhibits TOF and overflow interrupts until the other byte is written. Reset sets the TPM counter modulo registers to 0x0000, which results in a free-running timer counter (modulo disabled).

	7	6	5	4	3	2	1	0
R W	Bit 15	14	13	12	11	10	9	Bit 8
Reset	0	0	0	0	0	0	0	0

Figure B-5. Timer Counter Modulo Register High (TPMxMODH)

	7	6	5	4	3	2	1	0
R W	Bit 7	6	5	4	3	2	1	Bit 0
Reset	0	0	0	0	0	0	0	0

Figure B-6. Timer Counter Modulo Register Low (TPMxMODL)

It is good practice to wait for an overflow interrupt so both bytes of the modulo register can be written well before a new overflow. An alternative approach is to reset the TPM counter before writing to the TPM modulo registers to avoid confusion about when the first counter overflow will occur.