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	HCS12
Core Size	16-Bit
Speed	32MHz
Connectivity	EBI/EMI, I²C, IrDA, LINbus, SCI, SPI
Peripherals	LCD, Motor control PWM, POR, PWM, WDT
Number of I/O	80
Program Memory Size	32KB (32K x 8)
Program Memory Type	FLASH
EEPROM Size	4K x 8
RAM Size	2K x 8
Voltage - Supply (Vcc/Vdd)	4.5V ~ 5.5V
Data Converters	A/D 8x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	100-LQFP
Supplier Device Package	100-LQFP (14x14)
Purchase URL	https://www.e-xfl.com/pro/item?MUrl=&PartUrl=s9s12ha32j0cll

Table 2-2. Block Memory Map (continued)

Port	Offset or Address	Register	Access	Reset Value	Section/Page
S	0x0248	PTS—Port S Data Register	R/W	0x00	2.3.22/2-84
	0x0249	PTIS—Port S Input Register	R	[3]	2.3.23/2-86
	0x024A	DDRS—Port S Data Direction Register	R/W	0x00	2.3.24/2-87
	0x024B	RDRS—Port S Reduced Drive Register	R/W	0x00	2.3.25/2-88
	0x024C	PERS—Port S Pull Device Enable Register	R/W	0xFF	2.3.26/2-89
	0x024D	PPSS—Port S Polarity Select Register	R/W	0x00	2.3.27/2-89
	0x024E	WOMS—Port S Wired-Or Mode Register	R/W	0x00	2.3.28/2-90
	0x024F	PTSRR Port S Routing Register	R/W	0x00	2.3.29/2-90
	0x0250 : 0x0257	PIM Reserved	R	0x00	2.3.30/2-91
P	0x0258	PTP—Port P Data Register	R/W	0x00	2.3.31/2-91
	0x0259	PTIP—Port P Input Register	R	[3]	2.3.32/2-92
	0x025A	DDRP—Port P Data Direction Register	R/W	0x00	2.3.33/2-92
	0x025B	RDRP—Port P Reduced Drive Register	R/W	0x00	2.3.34/2-93
	0x025C	PERP—Port P Pull Device Enable Register	R/W	0xFF	2.3.35/2-94
	0x025D	PPSP—Port P Polarity Select Register	R/W	0xFF	2.3.36/2-94
	0x025E	PTPRRH Port P Routing Register High	R/W	0x00	2.3.37/2-95
	0x025F	PTPRRL Port P Routing Register Low	R/W	0x00	2.3.38/2-95
H	0x0260	PTH—Port H Data Register	R/W	0x00	2.3.39/2-96
	0x0261	PTIH—Port H Input Register	R	[3]	2.3.40/2-98
	0x0262	DDRH—Port H Data Direction Register	R/W	0x00	2.3.41/2-98
	0x0263	RDRH—Port H Reduced Drive Register	R/W	0x00	2.3.42/2-100
	0x0264	PERH—Port H Pull Device Enable Register	R/W	0xFF	2.3.43/2-100
	0x0265	PPSH—Port H Polarity Select Register	R/W	0xFF	2.3.44/2-101
	0x0266	WOMH—Port H Wired-Or Mode Register	R/W	0x00	2.3.45/2-101
	0x0267	PIM Reserved	R	0x00	2.3.46/2-102
	0x0268 : 0x026F	PIM Reserved	R	0x00	2.3.47/2-102

5.4.9 SYNC — Request Timed Reference Pulse

The SYNC command is unlike other BDM commands because the host does not necessarily know the correct communication speed to use for BDM communications until after it has analyzed the response to the SYNC command. To issue a SYNC command, the host should perform the following steps:

1. Drive the BKGD pin low for at least 128 cycles at the lowest possible BDM serial communication frequency
2. Drive BKGD high for a brief speedup pulse to get a fast rise time (this speedup pulse is typically one cycle of the host clock.)
3. Remove all drive to the BKGD pin so it reverts to high impedance.
4. Listen to the BKGD pin for the sync response pulse.

Upon detecting the SYNC request from the host, the target performs the following steps:

1. Discards any incomplete command received or bit retrieved.
2. Waits for BKGD to return to a logic one.
3. Delays 16 cycles to allow the host to stop driving the high speedup pulse.
4. Drives BKGD low for 128 cycles at the current BDM serial communication frequency.
5. Drives a one-cycle high speedup pulse to force a fast rise time on BKGD.
6. Removes all drive to the BKGD pin so it reverts to high impedance.

The host measures the low time of this 128 cycle SYNC response pulse and determines the correct speed for subsequent BDM communications. Typically, the host can determine the correct communication speed within a few percent of the actual target speed and the communication protocol can easily tolerate speed errors of several percent.

As soon as the SYNC request is detected by the target, any partially received command or bit retrieved is discarded. This is referred to as a soft-reset, equivalent to a time-out in the serial communication. After the SYNC response, the target will consider the next negative edge (issued by the host) as the start of a new BDM command or the start of new SYNC request.

Another use of the SYNC command pulse is to abort a pending ACK pulse. The behavior is exactly the same as in a regular SYNC command. Note that one of the possible causes for a command to not be acknowledged by the target is a host-target synchronization problem. In this case, the command may not have been understood by the target and so an ACK response pulse will not be issued.

5.4.10 Instruction Tracing

When a TRACE1 command is issued to the BDM in active BDM, the CPU exits the standard BDM firmware and executes a single instruction in the user code. Once this has occurred, the CPU is forced to return to the standard BDM firmware and the BDM is active and ready to receive a new command. If the TRACE1 command is issued again, the next user instruction will be executed. This facilitates stepping or tracing through the user code one instruction at a time.

If an interrupt is pending when a TRACE1 command is issued, the interrupt stacking operation occurs but no user instruction is executed. Once back in standard BDM firmware execution, the program counter points to the first instruction in the interrupt service routine.

7.1.2.3 Stop Mode

This mode is entered by executing the CPU STOP instruction.

The voltage regulator is in Reduced Power Mode (RPM).

The API is available.

The Phase Locked Loop (PLL) is off.

The Internal Reference Clock (IRC1M) is off.

Core Clock, Bus Clock and BDM Clock are stopped.

Depending on the setting of the PSTP and the OSCE bit, Stop Mode can be differentiated between Full Stop Mode (PSTP = 0 or OSCE=0) and Pseudo Stop Mode (PSTP = 1 and OSCE=1).

- **Full Stop Mode (PSTP=0 or OSCE=0)**
  The external oscillator (OSCLCP) is disabled.
  After wake-up from Full Stop Mode the Core Clock and Bus Clock are running on PLLCLK (PLLSEL=1). After wake-up from Full Stop Mode COP and RTI are running on IRCCLK (COPOSCSEL=0, RTIOSCSEL=0).
- **Pseudo Stop Mode (PSTP=1 and OSCE=1)**
  The external oscillator (OSCLCP) continues torun. If the respective enable bits are set the COP and RTI will continue to run.
  The clock configuration bits PLLSEL, COPOSCSEL, RTIOSCSEL are unchanged.

NOTE

When starting up the external oscillator (either by programming OSCE bit to 1 or on exit from Full Stop Mode with OSCE bit already 1) the software must wait for a minimum time equivalent to the startup-time of the external oscillator t_{UPOSC} before entering Pseudo Stop Mode.

8.6 Interrupts

The interrupts requested by the ADC12B8C are listed in Table 8-23. Refer to MCU specification for related vector address and priority.

Table 8-23. ATD Interrupt Vectors

Interrupt Source	CCR Mask	Local Enable
Sequence Complete Interrupt	I bit	ASCIE in ATDCTL2
Compare Interrupt	I bit	ACMPIE in ATDCTL2

See Section 8.3.2, “Register Descriptions” for further details.

1. Read: Anytime
   Write: Anytime when not in initialization mode, except RSTAT[1:0] and TSTAT[1:0] flags which are read-only; write of 1 clears flag; write of 0 is ignored

NOTE

The CANRFLG register is held in the reset state[1] when the initialization mode is active (INITRQ = 1 and INITAK = 1). This register is writable again as soon as the initialization mode is exited (INITRQ = 0 and INITAK = 0).

Table 9-11. CANRFLG Register Field Descriptions

Field	Description
7 WUPIF	**Wake-Up Interrupt Flag** — If the MSCAN detects CAN bus activity while in sleep mode (see Section 9.4.5.5, "MSCAN Sleep Mode,") and WUPE = 1 in CANTCTL0 (see Section 9.3.2.1, "MSCAN Control Register 0 (CANCTL0)"), the module will set WUPIF. If not masked, a wake-up interrupt is pending while this flag is set. 0 No wake-up activity observed while in sleep mode 1 MSCAN detected activity on the CAN bus and requested wake-up
6 CSCIF	**CAN Status Change Interrupt Flag** — This flag is set when the MSCAN changes its current CAN bus status due to the actual value of the transmit error counter (TEC) and the receive error counter (REC). An additional 4-bit (RSTAT[1:0], TSTAT[1:0]) status register, which is split into separate sections for TEC/REC, informs the system on the actual CAN bus status (see Section 9.3.2.6, "MSCAN Receiver Interrupt Enable Register (CANRIER)"). If not masked, an error interrupt is pending while this flag is set. CSCIF provides a blocking interrupt. That guarantees that the receiver/transmitter status bits (RSTAT/TSTAT) are only updated when no CAN status change interrupt is pending. If the TECs/RECs change their current value after the CSCIF is asserted, which would cause an additional state change in the RSTAT/TSTAT bits, these bits keep their status until the current CSCIF interrupt is cleared again. 0 No change in CAN bus status occurred since last interrupt 1 MSCAN changed current CAN bus status
5-4 RSTAT[1:0]	**Receiver Status Bits** — The values of the error counters control the actual CAN bus status of the MSCAN. As soon as the status change interrupt flag (CSCIF) is set, these bits indicate the appropriate receiver related CAN bus status of the MSCAN. The coding for the bits RSTAT1, RSTAT0 is: 00 RxOK: 0 ≤ receive error counter ≤ 96 01 RxWRN: 96 < receive error counter ≤ 127 10 RxERR: 127 < receive error counter 11 Bus-off(1): transmit error counter > 255
3-2 TSTAT[1:0]	**Transmitter Status Bits** — The values of the error counters control the actual CAN bus status of the MSCAN. As soon as the status change interrupt flag (CSCIF) is set, these bits indicate the appropriate transmitter related CAN bus status of the MSCAN. The coding for the bits TSTAT1, TSTAT0 is: 00 TxOK: 0 ≤ transmit error counter ≤ 96 01 TxWRN: 96 < transmit error counter ≤ 127 10 TxERR: 127 < transmit error counter ≤ 255 11 Bus-Off: transmit error counter > 255

1. The RSTAT[1:0], TSTAT[1:0] bits are not affected by initialization mode.

9.3.2.10 MSCAN Transmitter Message Abort Acknowledge Register (CANTAAK)

The CANTAAK register indicates the successful abort of a queued message, if requested by the appropriate bits in the CANTARQ register.

Module Base + 0x0009 Access: User read/write(1)

	7	6	5	4	3	2	1	0
R	0	0	0	0	0	ABTAK2	ABTAK1	ABTAK0
W								
Reset:	0	0	0	0	0	0	0	0

= Unimplemented

Figure 9-13. MSCAN Transmitter Message Abort Acknowledge Register (CANTAAK)

1. Read: Anytime
   Write: Unimplemented

NOTE

The CANTAAK register is held in the reset state when the initialization mode is active (INITRQ = 1 and INITAK = 1).

Table 9-16. CANTAAK Register Field Descriptions

Field	Description
2-0 ABTAK[2:0]	**Abort Acknowledge** — This flag acknowledges that a message was aborted due to a pending abort request from the CPU. After a particular message buffer is flagged empty, this flag can be used by the application software to identify whether the message was aborted successfully or was sent anyway. The ABTAKx flag is cleared whenever the corresponding TXE flag is cleared. 0 The message was not aborted. 1 The message was aborted.

9.3.2.11 MSCAN Transmit Buffer Selection Register (CANTBSEL)

The CANTBSEL register allows the selection of the actual transmit message buffer, which then will be accessible in the CANTXFG register space.

Module Base + 0x000A Access: User read/write(1)

	7	6	5	4	3	2	1	0
R	0	0	0	0	0	TX2	TX1	TX0
W								
Reset:	0	0	0	0	0	0	0	0

= Unimplemented

Figure 9-14. MSCAN Transmit Buffer Selection Register (CANTBSEL)

1. Read: Find the lowest ordered bit set to 1, all other bits will be read as 0
   Write: Anytime when not in initialization mode

9.4 Functional Description

9.4.1 General

This section provides a complete functional description of the MSCAN.

9.4.2 Message Storage

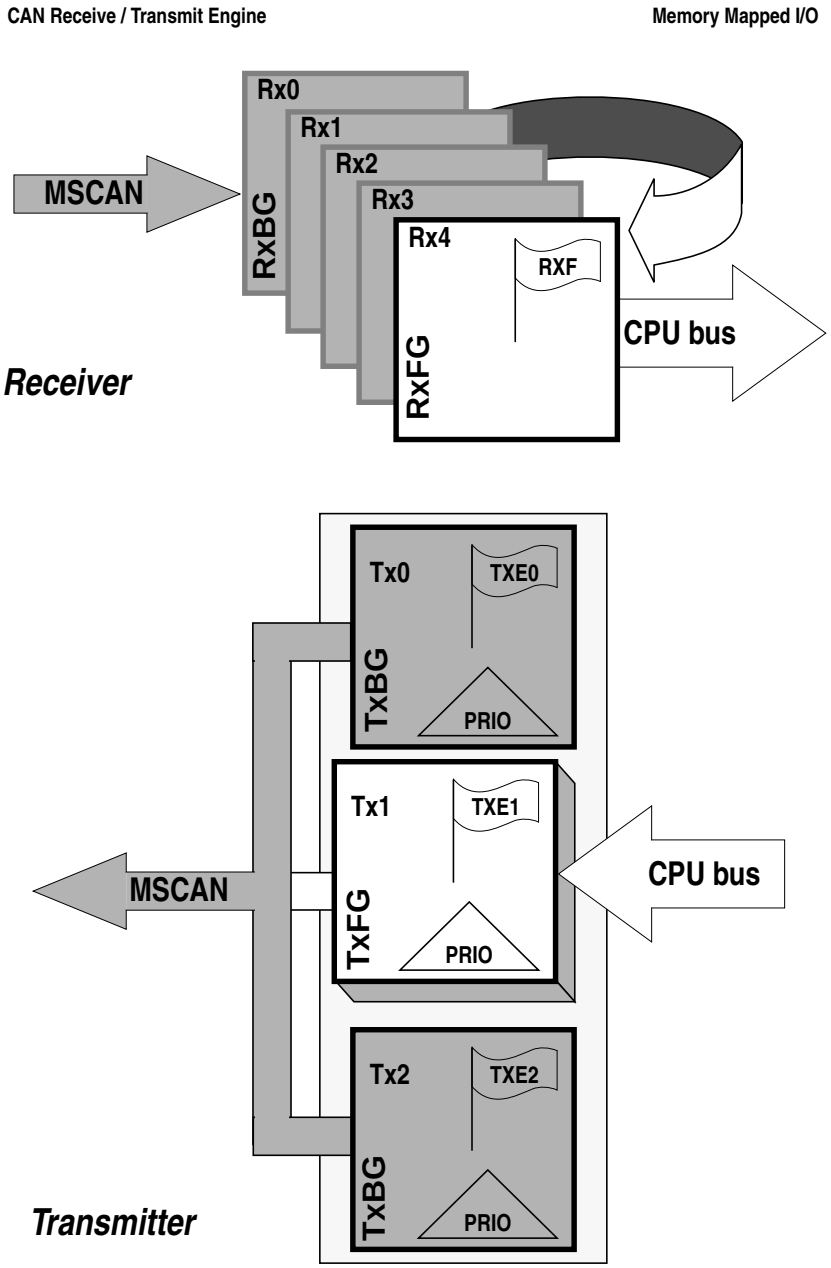

Figure 9-39. User Model for Message Buffer Organization

10.1.2 Modes of Operation

The IIC functions the same in normal, special, and emulation modes. It has two low power modes: wait and stop modes.

10.1.3 Block Diagram

The block diagram of the IIC module is shown in Figure 10-1.

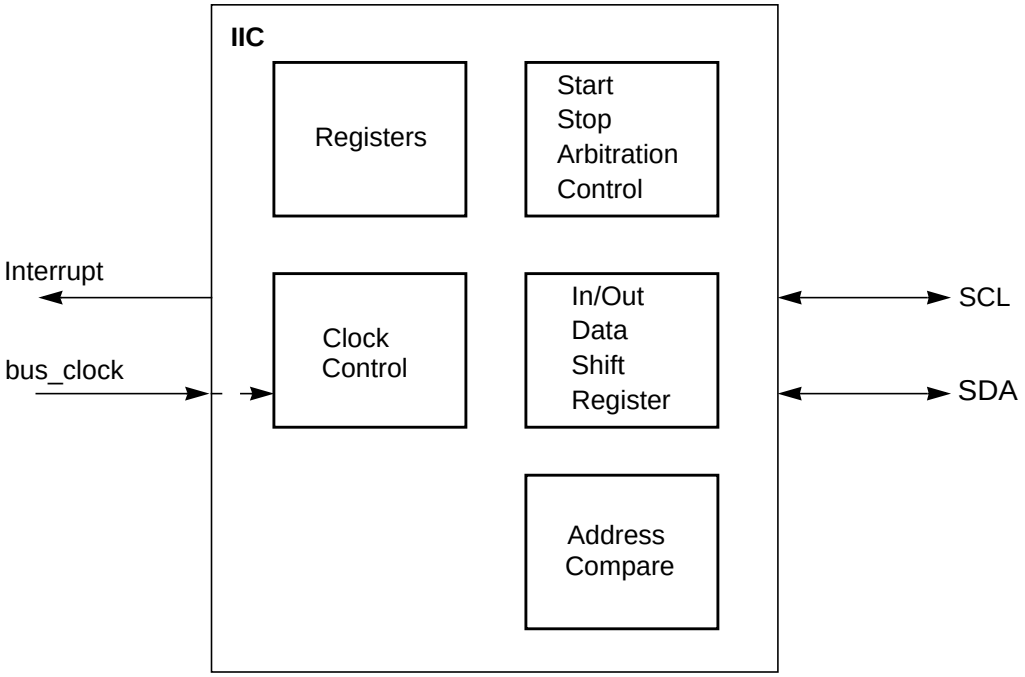

Figure 10-1. IIC Block Diagram

11.3.2.14 PWM Channel Duty Registers (PWMDTYx)

There is a dedicated duty register for each channel. The value in this register determines the duty of the associated PWM channel. The duty value is compared to the counter and if it is equal to the counter value a match occurs and the output changes state.

The duty registers for each channel are double buffered so that if they change while the channel is enabled, the change will NOT take effect until one of the following occurs:

- The effective period ends
- The counter is written (counter resets to $00)
- The channel is disabled

In this way, the output of the PWM will always be either the old duty waveform or the new duty waveform, not some variation in between. If the channel is not enabled, then writes to the duty register will go directly to the latches as well as the buffer.

NOTE

Reads of this register return the most recent value written. Reads do not necessarily return the value of the currently active duty due to the double buffering scheme.

See Section 11.4.2.3, “PWM Period and Duty” for more information.

NOTE

Depending on the polarity bit, the duty registers will contain the count of either the high time or the low time. If the polarity bit is one, the output starts high and then goes low when the duty count is reached, so the duty registers contain a count of the high time. If the polarity bit is zero, the output starts low and then goes high when the duty count is reached, so the duty registers contain a count of the low time.

To calculate the output duty cycle (high time as a% of period) for a particular channel:

- Polarity = 0 (PPOL x =0)

  Duty Cycle = [(PWMPERx-PWMDTYx)/PWMPERx] * 100%
- Polarity = 1 (PPOLx = 1)

  Duty Cycle = [PWMDTYx / PWMPERx] * 100%

For boundary case programming values, please refer to Section 11.4.2.8, “PWM Boundary Cases”.

Module Base + 0x001C = PWMDTY0, 0x001D = PWMDTY1, 0x001E = PWMDTY2, 0x001F = PWMDTY3
Module Base + 0x0020 = PWMDTY4, 0x0021 = PWMDTY5, 0x0022 = PWMDTY6, 0x0023 = PWMDTY7

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

Figure 11-16. PWM Channel Duty Registers (PWMDTYx)

Read: Anytime

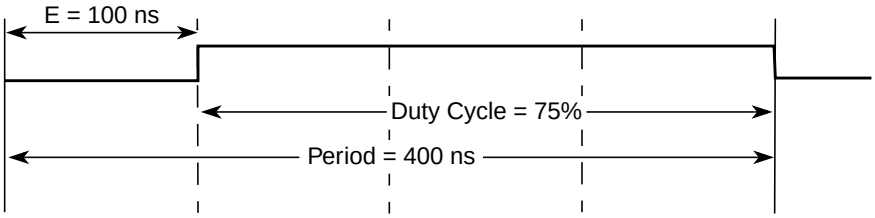

Figure 11-21. PWM Left Aligned Output Example Waveform

11.4.2.6 Center Aligned Outputs

For center aligned output mode selection, set the CAEx bit (CAEx = 1) in the PWMCAE register and the corresponding PWM output will be center aligned.

The 8-bit counter operates as an up/down counter in this mode and is set to up whenever the counter is equal to $00. The counter compares to two registers, a duty register and a period register as shown in the block diagram in Figure 11-19. When the PWM counter matches the duty register, the output flip-flop changes state, causing the PWM waveform to also change state. A match between the PWM counter and the period register changes the counter direction from an up-count to a down-count. When the PWM counter decrements and matches the duty register again, the output flip-flop changes state causing the PWM output to also change state. When the PWM counter decrements and reaches zero, the counter direction changes from a down-count back to an up-count and a load from the double buffer period and duty registers to the associated registers is performed, as described in Section 11.4.2.3, “PWM Period and Duty”. The counter counts from 0 up to the value in the period register and then back down to 0. Thus the effective period is PWMPERx*2.

NOTE

Changing the PWM output mode from left aligned to center aligned output (or vice versa) while channels are operating can cause irregularities in the PWM output. It is recommended to program the output mode before enabling the PWM channel.

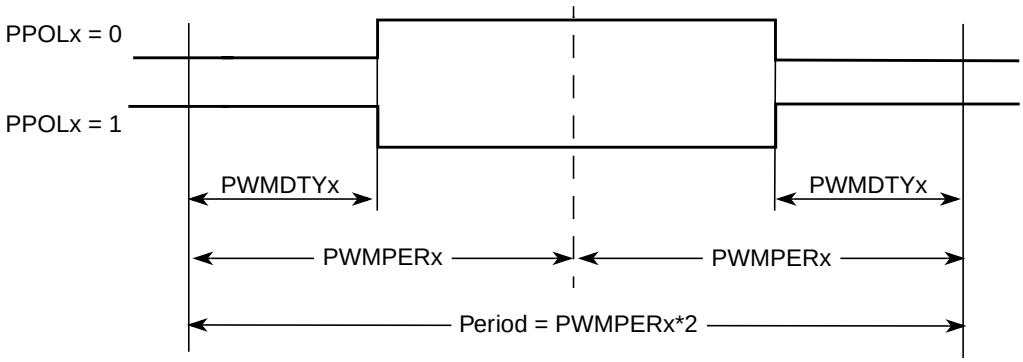

Figure 11-22. PWM Center Aligned Output Waveform

12.3.2.8 SCI Status Register 2 (SCISR2)

Module Base + 0x0005

	7	6	5	4	3	2	1	0
R	AMAP	0	0	TXPOL	RXPOL	BRK13	TXDIR	RAF
W								
Reset	0	0	0	0	0	0	0	0

= Unimplemented or Reserved

Figure 12-11. SCI Status Register 2 (SCISR2)

Read: Anytime

Write: Anytime

Table 12-12. SCISR2 Field Descriptions

Field	Description
7 AMAP	**Alternative Map** — This bit controls which registers sharing the same address space are accessible. In the reset condition the SCI behaves as previous versions. Setting AMAP=1 allows the access to another set of control and status registers and hides the baud rate and SCI control Register 1. 0 The registers labelled SCIBDH (0x0000),SCIBDL (0x0001), SCICR1 (0x0002) are accessible 1 The registers labelled SCIASR1 (0x0000),SCIACR1 (0x0001), SCIACR2 (0x00002) are accessible
4 TXPOL	**Transmit Polarity** — This bit control the polarity of the transmitted data. In NRZ format, a one is represented by a mark and a zero is represented by a space for normal polarity, and the opposite for inverted polarity. In IrDA format, a zero is represented by short high pulse in the middle of a bit time remaining idle low for a one for normal polarity, and a zero is represented by short low pulse in the middle of a bit time remaining idle high for a one for inverted polarity. 0 Normal polarity 1 Inverted polarity
3 RXPOL	**Receive Polarity** — This bit control the polarity of the received data. In NRZ format, a one is represented by a mark and a zero is represented by a space for normal polarity, and the opposite for inverted polarity. In IrDA format, a zero is represented by short high pulse in the middle of a bit time remaining idle low for a one for normal polarity, and a zero is represented by short low pulse in the middle of a bit time remaining idle high for a one for inverted polarity. 0 Normal polarity 1 Inverted polarity
2 BRK13	**Break Transmit Character Length** — This bit determines whether the transmit break character is 10 or 11 bit respectively 13 or 14 bits long. The detection of a framing error is not affected by this bit. 0 Break character is 10 or 11 bit long 1 Break character is 13 or 14 bit long
1 TXDIR	**Transmitter Pin Data Direction in Single-Wire Mode** — This bit determines whether the TXD pin is going to be used as an input or output, in the single-wire mode of operation. This bit is only relevant in the single-wire mode of operation. 0 TXD pin to be used as an input in single-wire mode 1 TXD pin to be used as an output in single-wire mode
0 RAF	**Receiver Active Flag** — RAF is set when the receiver detects a logic 0 during the RT1 time period of the start bit search. RAF is cleared when the receiver detects an idle character. 0 No reception in progress 1 Reception in progress

In Figure 12-22 the verification samples RT3 and RT5 determine that the first low detected was noise and not the beginning of a start bit. The RT clock is reset and the start bit search begins again. The noise flag is not set because the noise occurred before the start bit was found.

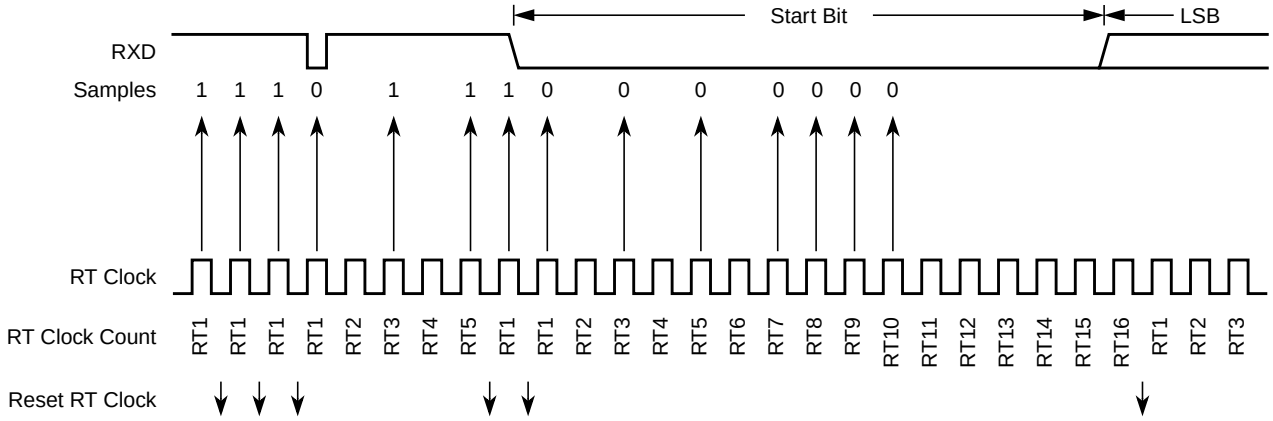

Figure 12-22. Start Bit Search Example 1

In Figure 12-23, verification sample at RT3 is high. The RT3 sample sets the noise flag. Although the perceived bit time is misaligned, the data samples RT8, RT9, and RT10 are within the bit time and data recovery is successful.

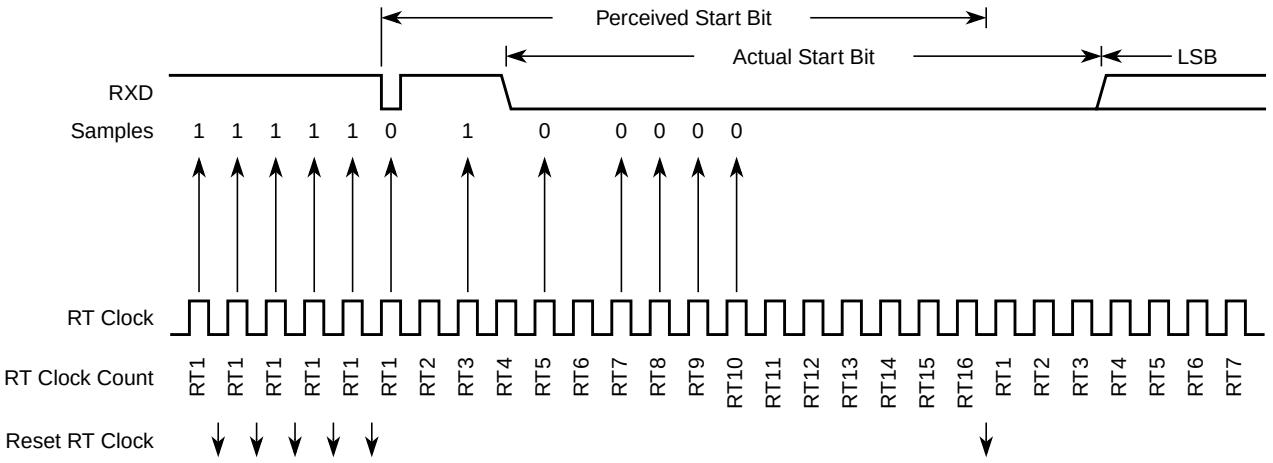

Figure 12-23. Start Bit Search Example 2

The main element of the SPI system is the SPI data register. The n-bit[1] data register in the master and the n-bit[1] data register in the slave are linked by the MOSI and MISO pins to form a distributed 2n-bit[1] register. When a data transfer operation is performed, this 2n-bit[1] register is serially shifted n[1] bit positions by the S-clock from the master, so data is exchanged between the master and the slave. Data written to the master SPI data register becomes the output data for the slave, and data read from the master SPI data register after a transfer operation is the input data from the slave.

A read of SPISR with SPTEF = 1 followed by a write to SPIDR puts data into the transmit data register. When a transfer is complete and SPIF is cleared, received data is moved into the receive data register. This data register acts as the SPI receive data register for reads and as the SPI transmit data register for writes. A common SPI data register address is shared for reading data from the read data buffer and for writing data to the transmit data register.

The clock phase control bit (CPHA) and a clock polarity control bit (CPOL) in the SPI control register 1 (SPICR1) select one of four possible clock formats to be used by the SPI system. The CPOL bit simply selects a non-inverted or inverted clock. The CPHA bit is used to accommodate two fundamentally different protocols by sampling data on odd numbered SCK edges or on even numbered SCK edges (see Section 13.4.3, “Transmission Formats”).

The SPI can be configured to operate as a master or as a slave. When the MSTR bit in SPI control register1 is set, master mode is selected, when the MSTR bit is clear, slave mode is selected.

NOTE

A change of CPOL or MSTR bit while there is a received byte pending in the receive shift register will destroy the received byte and must be avoided.

13.4.1 Master Mode

The SPI operates in master mode when the MSTR bit is set. Only a master SPI module can initiate transmissions. A transmission begins by writing to the master SPI data register. If the shift register is empty, data immediately transfers to the shift register. Data begins shifting out on the MOSI pin under the control of the serial clock.

- Serial clock

  The SPR2, SPR1, and SPR0 baud rate selection bits, in conjunction with the SPPR2, SPPR1, and SPPR0 baud rate preselection bits in the SPI baud rate register, control the baud rate generator and determine the speed of the transmission. The SCK pin is the SPI clock output. Through the SCK pin, the baud rate generator of the master controls the shift register of the slave peripheral.
- MOSI, MISO pin

  In master mode, the function of the serial data output pin (MOSI) and the serial data input pin (MISO) is determined by the SPC0 and BIDIROE control bits.
- $\overline{SS}$ pin

  If MODFEN and SSOE are set, the $\overline{SS}$ pin is configured as slave select output. The $\overline{SS}$ output becomes low during each transmission and is high when the SPI is in idle state.

1. n depends on the selected transfer width, please refer to Section 13.3.2.2, “SPI Control Register 2 (SPICR2)

Read: Anytime but will always return 0x0000 (1 state is transient)

Write: Anytime

Table 14-3. CFORC Field Descriptions

Field	Description
7:0 FOC[7:0]	**Force Output Compare Action for Channel 7:0** — A write to this register with the corresponding data bit(s) set causes the action which is programmed for output compare “x” to occur immediately. The action taken is the same as if a successful comparison had just taken place with the TCx register except the interrupt flag does not get set. **Note:** A channel 7 event, which can be a counter overflow when TTOV[7] is set or a successful output compare on channel 7, overrides any channel 6:0 compares. If forced output compare on any channel occurs at the same time as the successful output compare then forced output compare action will take precedence and interrupt flag won’t get set.

14.3.2.3 Output Compare 7 Mask Register (OC7M)

Module Base + 0x0002

	7	6	5	4	3	2	1	0
R W	OC7M7	OC7M6	OC7M5	OC7M4	OC7M3	OC7M2	OC7M1	OC7M0
Reset	0	0	0	0	0	0	0	0

Figure 14-8. Output Compare 7 Mask Register (OC7M)

Read: Anytime

Write: Anytime

Table 14-4. OC7M Field Descriptions

Field	Description
7:0 OC7M[7:0]	**Output Compare 7 Mask** — A channel 7 event, which can be a counter overflow when TTOV[7] is set or a successful output compare on channel 7, overrides any channel 6:0 compares. For each OC7M bit that is set, the output compare action reflects the corresponding OC7D bit. 0 The corresponding OC7Dx bit in the output compare 7 data register will not be transferred to the timer port on a channel 7 event, even if the corresponding pin is setup for output compare. 1 The corresponding OC7Dx bit in the output compare 7 data register will be transferred to the timer port on a channel 7 event. **Note:** The corresponding channel must also be setup for output compare (IOSx = 1 and OCPDx = 0) for data to be transferred from the output compare 7 data register to the timer port.

Table 15-19. P-Flash Protection Lower Address Range

FPLS[1:0]	Global Address Range	Protected Size
00	0x3_8000–0x3_83FF	1 Kbyte
01	0x3_8000–0x3_87FF	2 Kbytes
10	0x3_8000–0x3_8FFF	4 Kbytes
11	0x3_8000–0x3_9FFF	8 Kbytes

All possible P-Flash protection scenarios are shown in Figure 15-14. Although the protection scheme is loaded from the Flash memory at global address 0x3_FF0C during the reset sequence, it can be changed by the user. The P-Flash protection scheme can be used by applications requiring reprogramming in single chip mode while providing as much protection as possible if reprogramming is not required.

return CCIF to 1 and the FCCOB register will be used to communicate any results. The flow for a generic command write sequence is shown in Figure 16-26.

Table 17-15. FERSTAT Field Descriptions

Field	Description
1 DFDIF	**Double Bit Fault Detect Interrupt Flag** — The setting of the DFDIF flag indicates that a double bit fault was detected in the stored parity and data bits during a Flash array read operation or that a Flash array read operation was attempted on a Flash block that was under a Flash command operation.[1] The DFDIF flag is cleared by writing a 1 to DFDIF. Writing a 0 to DFDIF has no effect on DFDIF. 0 No double bit fault detected 1 Double bit fault detected or an invalid Flash array read operation attempted
0 SFDIF	**Single Bit Fault Detect Interrupt Flag** — With the IGNSF bit in the FCNFG register clear, the SFDIF flag indicates that a single bit fault was detected in the stored parity and data bits during a Flash array read operation or that a Flash array read operation was attempted on a Flash block that was under a Flash command operation.[1] The SFDIF flag is cleared by writing a 1 to SFDIF. Writing a 0 to SFDIF has no effect on SFDIF. 0 No single bit fault detected 1 Single bit fault detected and corrected or an invalid Flash array read operation attempted

[1] The single bit fault and double bit fault flags are mutually exclusive for parity errors (an ECC fault occurrence can be either single fault or double fault but never both). A simultaneous access collision (read attempted while command running) is indicated when both SFDIF and DFDIF flags are high.

17.3.2.9 P-Flash Protection Register (FPROT)

The FPROT register defines which P-Flash sectors are protected against program and erase operations.

Offset Module Base + 0x0008

	7	6	5	4	3	2	1	0
R	FPOPEN	RNV6	FPHDIS	FPHS[1:0]		FPLDIS	FPLS[1:0]	
W								
Reset	F	F	F	F	F	F	F	F

= Unimplemented or Reserved

Figure 17-13. Flash Protection Register (FPROT)

The (unreserved) bits of the FPROT register are writable with the restriction that the size of the protected region can only be increased (see Section 17.3.2.9.1, “P-Flash Protection Restrictions,” and Table 17-20).

During the reset sequence, the FPROT register is loaded with the contents of the P-Flash protection byte in the Flash configuration field at global address 0x3_FF0C located in P-Flash memory (see Table 17-3) as indicated by reset condition ‘F’ in Figure 17-13. To change the P-Flash protection that will be loaded during the reset sequence, the upper sector of the P-Flash memory must be unprotected, then the P-Flash protection byte must be reprogrammed. If a double bit fault is detected while reading the P-Flash phrase containing the P-Flash protection byte during the reset sequence, the FPOPEN bit will be cleared and remaining bits in the FPROT register will be set to leave the P-Flash memory fully protected.

Trying to alter data in any protected area in the P-Flash memory will result in a protection violation error and the FPVIOL bit will be set in the FSTAT register. The block erase of a P-Flash block is not possible if any of the P-Flash sectors contained in the same P-Flash block are protected.

18.4.4.1 1/1 Duty Multiplexed with 1/1 Bias Mode

Duty = 1/1:DUTY1 = 0, DUTY0 = 1

Bias = 1/1:BIAS = 0 or BIAS = 1

$V_0 = V_1 = VSSX$, $V_2 = V_3 = VLCD$

- BP1, BP2, and BP3 are not used, a maximum of 40 segments are displayed.

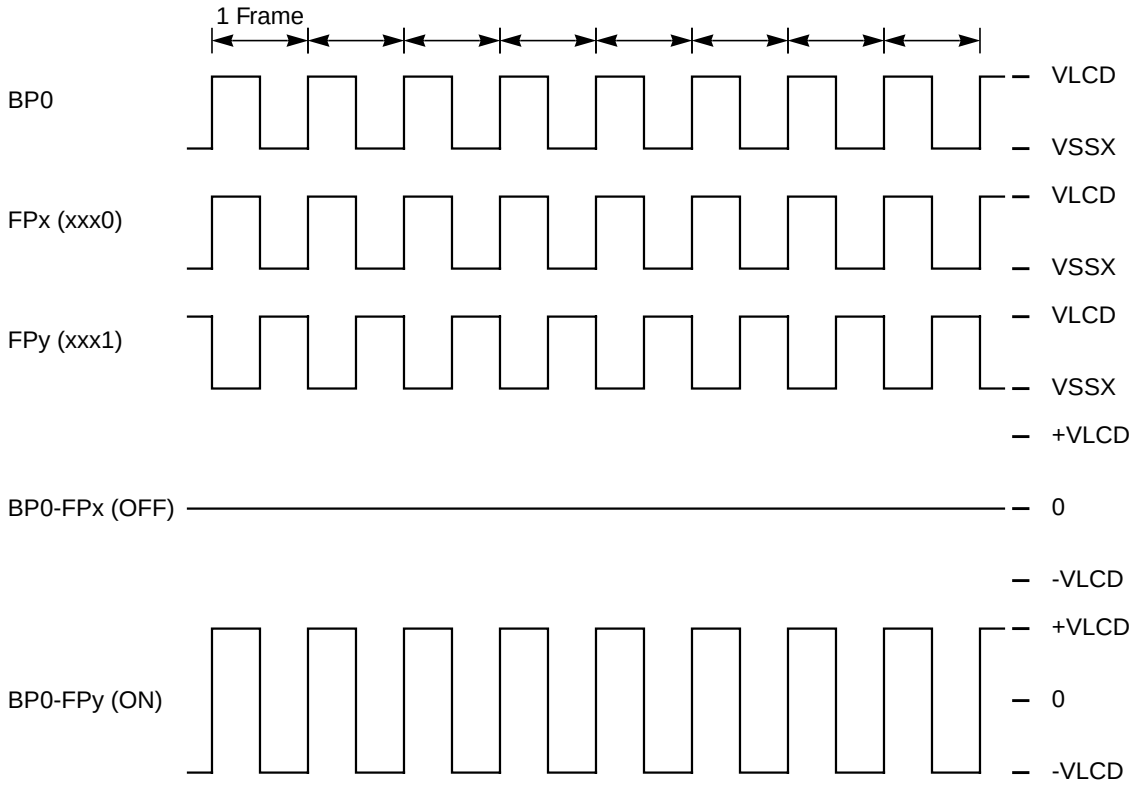

Figure 18-10. 1/1 Duty and 1/1 Bias

A.3.1.13 Set Field Margin Level (FCMD=0x0E)

The maximum set field margin level time is given by:

$$t = 350 \cdot \frac{1}{f_{NVMBUS}}$$

A.3.1.14 Erase Verify D-Flash Section (FCMD=0x10)

The time required to Erase Verify D-Flash for a given number of words N_W is given by:

$$t_{dcheck} \approx (450 + N_W) \cdot \frac{1}{f_{NVMBUS}}$$

A.3.1.15 Program D-Flash (FCMD=0x11)

D-Flash programming time is dependent on the number of words being programmed and their location with respect to a row boundary since programming across a row boundary requires extra steps. The D-Flash programming time is specified for different cases: 1,2,3,4 words and 4 words across a row boundary.

The typical D-Flash programming time is given by the following equation, where N_W denotes the number of words; BC=0 if no row boundary is crossed and BC=1 if a row boundary is crossed:

$$t_{dpgm} \approx \left((14 + (54 \cdot N_W) + (14 \cdot BC)) \cdot \frac{1}{f_{NVMOP}}\right) + \left((500 + (525 \cdot N_W) + (100 \cdot BC)) \cdot \frac{1}{f_{NVMBUS}}\right)$$

The maximum D-Flash programming time is given by:

$$t_{dpgm} \approx \left((14 + (54 \cdot N_W) + (14 \cdot BC)) \cdot \frac{1}{f_{NVMOP}}\right) + \left((500 + (750 \cdot N_W) + (100 \cdot BC)) \cdot \frac{1}{f_{NVMBUS}}\right)$$

A.3.1.16 Erase D-Flash Sector (FCMD=0x12)

Typical D-Flash sector erase times, expected on a new device where no margin verify fails occur, is given by:

$$t_{dera} \approx 5025 \cdot \frac{1}{f_{NVMOP}} + 700 \cdot \frac{1}{f_{NVMBUS}}$$

Maximum D-Flash sector erase times is given by:

$$t_{dera} \approx 20100 \cdot \frac{1}{f_{NVMOP}} + 3400 \cdot \frac{1}{f_{NVMBUS}}$$

The D-Flash sector erase time is ~ 5 ms on a new device and can extend to ~ 20 ms as the flash is cycled.

Address	Name		Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0x00AA	PWMSCNTA	R	0	0	0	0	0	0	0	0
		W								
0x00AB	PWMSCNTB	R	0	0	0	0	0	0	0	0
		W								
0x00AC	PWMCNT0	R	Bit 7	6	5	4	3	2	1	Bit 0
		W	0	0	0	0	0	0	0	0
0x00AD	PWMCNT1	R	Bit 7	6	5	4	3	2	1	Bit 0
		W	0	0	0	0	0	0	0	0
0x00AE	PWMCNT2	R	Bit 7	6	5	4	3	2	1	Bit 0
		W	0	0	0	0	0	0	0	0
0x00AF	PWMCNT3	R	Bit 7	6	5	4	3	2	1	Bit 0
		W	0	0	0	0	0	0	0	0
0x00B0	PWMCNT4	R	Bit 7	6	5	4	3	2	1	Bit 0
		W	0	0	0	0	0	0	0	0
0x00B1	PWMCNT5	R	Bit 7	6	5	4	3	2	1	Bit 0
		W	0	0	0	0	0	0	0	0
0x00B2	PWMCNT6	R	Bit 7	6	5	4	3	2	1	Bit 0
		W	0	0	0	0	0	0	0	0
0x00B3	PWMCNT7	R	Bit 7	6	5	4	3	2	1	Bit 0
		W	0	0	0	0	0	0	0	0
0x00B4	PWMPER0	R W	Bit 7	6	5	4	3	2	1	Bit 0
0x00B5	PWMPER1	R W	Bit 7	6	5	4	3	2	1	Bit 0
0x00B6	PWMPER2	R W	Bit 7	6	5	4	3	2	1	Bit 0
0x00B7	PWMPER3	R W	Bit 7	6	5	4	3	2	1	Bit 0
0x00B8	PWMPER4	R W	Bit 7	6	5	4	3	2	1	Bit 0
0x00B9	PWMPER5	R W	Bit 7	6	5	4	3	2	1	Bit 0
0x00BA	PWMPER6	R W	Bit 7	6	5	4	3	2	1	Bit 0
0x00BB	PWMPER7	R W	Bit 7	6	5	4	3	2	1	Bit 0
0x00BC	PWMDTY0	R W	Bit 7	6	5	4	3	2	1	Bit 0
0x00BD	PWMDTY1	R W	Bit 7	6	5	4	3	2	1	Bit 0