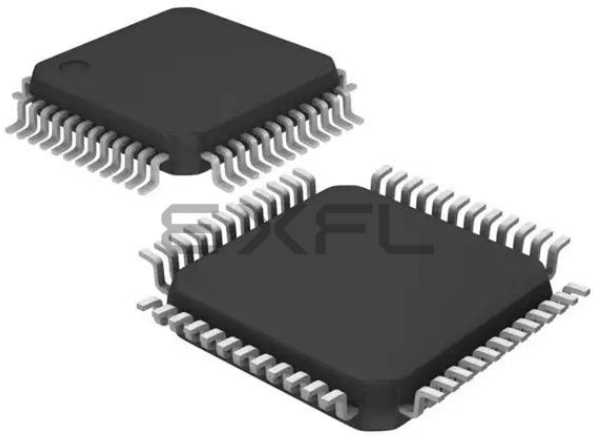

Welcome to E-XFL.COM

Embedded - Microcontrollers - Application Specific: Tailored Solutions for Precision and Performance

Embedded - Microcontrollers - Application Specific represents a category of microcontrollers designed with unique features and capabilities tailored to specific application needs. Unlike general-purpose microcontrollers, application-specific microcontrollers are optimized for particular tasks, offering enhanced performance, efficiency, and functionality to meet the demands of specialized applications.

What Are Embedded - Microcontrollers - Application Specific?

Application-specific microcontrollers are engineered to

Details	
Product Status	Active
Applications	Automotive
Core Processor	S12
Program Memory Type	FLASH (32KB)
Controller Series	HCS12
RAM Size	2K x 8
Interface	LIN, SCI
Number of I/O	9
Voltage - Supply	2.25V ~ 5.5V
Operating Temperature	-40°C ~ 105°C
Mounting Type	Surface Mount
Package / Case	48-LQFP
Supplier Device Package	48-LQFP (7x7)
Purchase URL	https://www.e-xfl.com/product-detail/nxp-semiconductors/mm912f634dv2apr2

Table 43 lists the timing characteristics for master mode.

Table 43. SPI Master Mode Timing Characteristics

Characteristic	Symbol	Min	Typ	Max	Unit
SCK frequency	f_{SCK}	1/2048	-	1/2	f_{BUS}
SCK period	t_{SCK}	2.0	-	2048	t_{BUS}
Enable lead time	t_{LEAD}	-	1/2	-	t_{SCK}
Enable lag time	t_{LAG}	-	1/2	-	t_{SCK}
Clock (SCK) high or low time	t_{WSCK}	-	1/2	-	t_{SCK}
Data setup time (inputs)	t_{SU}	8.0	-	-	ns
Data hold time (inputs)	t_{HI}	8.0	-	-	ns
Data valid after SCK edge	t_{VSCK}	-	-	29	ns
Data valid after $\overline{SS}$ fall (CPHA = 0)	t_{VSS}	-	-	15	ns
Data hold time (outputs)	t_{HO}	20	-	-	ns
Rise and fall time inputs	t_{RFI}	-	-	8.0	ns
Rise and fall time outputs	t_{RFO}	-	-	8.0	ns

Table 65. Analog die Registers[60] - 0x0200–0x02FF D2D Blocking Access (D2DI) 2 of 3/ 0x0300–0x03FF D2D Non Blocking Access (D2DI) 3 of 3 (continued)

Offset	Name		7	6	5	4	3	2	1	0
0x97	ADR8 (lo)	R	adr8 1	adr8 0	0	0	0	0	0	0
	ADC Data Result Register 8	W								
0x98	ADR9 (hi)	R	adr9 9	adr9 8	adr9 7	adr9 6	adr9 5	adr9 4	adr9 3	adr9 2
	ADC Data Result Register 9	W								
0x99	ADR9 (lo)	R	adr9 1	adr9 0	0	0	0	0	0	0
	ADC Data Result Register 9	W								
0x9A	ADR10 (hi)	R	adr10 9	adr10 8	adr10 7	adr10 6	adr10 5	adr10 4	adr10 3	adr10 2
	ADC Data Result Reg 10	W								
0x9B	ADR10 (lo)	R	adr10 1	adr10 0	0	0	0	0	0	0
	ADC Data Result Reg 10	W								
0x9C	ADR11 (hi)	R	adr11 9	adr11 8	adr11 7	adr11 6	adr11 5	adr11 4	adr11 3	adr11 2
	ADC Data Result Reg 11	W								
0x9D	ADR11 (lo)	R	adr11 1	adr11 0	0	0	0	0	0	0
	ADC Data Result Reg 11	W								
0x9E	ADR12 (hi)	R	adr12 9	adr12 8	adr12 7	adr12 6	adr12 5	adr12 4	adr12 3	adr12 2
	ADC Data Result Reg 12	W								
0x9F	ADR12 (lo)	R	adr12 1	adr12 0	0	0	0	0	0	0
	ADC Data Result Reg 12	W								
0xA2	ADR14 (hi)	R	adr14 9	adr14 8	adr14 7	adr14 6	adr14 5	adr14 4	adr14 3	adr14 2
	ADC Data Result Reg 14	W								
0xA3	ADR14 (lo)	R	adr14 1	adr14 0	0	0	0	0	0	0
	ADC Data Result Reg 14	W								
0xA4	ADR15 (hi)	R	adr15 9	adr15 8	adr15 7	adr15 6	adr15 5	adr15 4	adr15 3	adr15 2
	ADC Data Result Reg 15	W								
0xA5	ADR15 (lo)	R	adr15 1	adr15 0	0	0	0	0	0	0
	ADC Data Result Reg 15	W								
0xC0	TIOS	R	0	0	0	0	IOS3	IOS2	IOS1	IOS0
	TIM InCap/OutComp Select	W								
0xC1	CFORC	R	0	0	0	0	0	0	0	0
	Timer Compare Force Reg	W					FOC3	FOC2	FOC1	FOC0
0xC2	OC3M	R	0	0	0	0	OC3M3	OC3M2	OC3M1	OC3M0
	Output Comp 3 Mask Reg	W								
0xC3	OC3D	R	0	0	0	0	OC3D3	OC3D2	OC3D1	OC3D0
	Output Comp 3 Data Reg	W								
0xC4	TCNT (hi)	R	tcnt 15	tcnt 14	tcnt 13	tcnt 12	tcnt 11	tcnt 10	tcnt 9	tcnt 8
	Timer Count Register	W								
0xC5	TCNT (lo)	R	tcnt 7	tcnt 6	tcnt 5	tcnt 4	tcnt 3	tcnt 2	tcnt 1	tcnt 0
	Timer Count Register	W								
0xC6	TSCR1	R	TEN	0	0	TFFCA	0	0	0	0
	Timer System Control Reg 1	W								
0xC7	TTOV	R	0	0	0	0	TOV3	TOV2	TOV1	TOV0
	Timer Toggle Overflow Reg	W								

Table 79. ISR - Register Field Descriptions

Field	Description
0 - VSI	VSI - Voltage Status Interrupt combining the following sources: • Low Battery Interrupt • Low Voltage Interrupt • High Voltage Interrupt • Voltage Regulator Over-voltage Interrupt • Voltage Regulator High Temperature Interrupt
1 - CH0	CH0 - TIM Channel 0 Interrupt
2 - CH1	CH1 - TIM Channel 1 Interrupt
3 - CH2	CH2 - TIM Channel 2 Interrupt
4 - CH3	CH3 - TIM Channel 3 Interrupt
5 - TOV	TOV - Timer Overflow Interrupt
6 - ERR	ERR - SCI Error Interrupt
7 - TX	TX - SCI Transmit Interrupt
8 - RX	RX - SCI Receive Interrupt
9 - SCI	SCI - ADC Sequence Complete Interrupt
10 - LINOT	LINOT - LIN Driver Over-temperature Interrupt
11 - HSOT	HSOT - High Side Over-temperature Interrupt
12 - LSOT	LSOT - Low Side Over-temperature Interrupt
13 - HOT	HOT - HSUP Over-temperature Interrupt

4.6.1.2 Interrupt Vector Emulation by Priority

To allow a vector based interrupt handling by the MCU, the number of the highest prioritized interrupt pending is returned in the Interrupt Vector Register. To allow an offset based vector table, the result is pre-shifted (multiple of 2). Reading this register will not acknowledge an interrupt. An additional D2D access is necessary to serve the specific module.

4.6.1.2.1 Interrupt Vector Register (IVR)

Table 80. Interrupt Vector Register (IVR)

Offset[69] 0x02 Access: User read

	7	6	5	4	3	2	1	0
R	0	0	IRQ					
W								

Note:
69. Offset related to 0x0200 for blocking access and 0x300 for non blocking access within the global address space.

Table 81. IVR - Register Field Descriptions

Field	Description
5:0 IRQ	Represents the highest prioritized interrupt pending. See Table 82 In case no interrupt is pending, the result will be 0.

The following table is listing all MM912F634 analog die interrupt sources with the corresponding priority.

4.9 Window Watchdog

The MM912F634 analog die includes a configurable window watchdog, which is active in Normal mode. The watchdog module is based on a separate clock source (f_{BASE}) operating independent from the MCU based D2DCLK clock. The watchdog timeout (t_{WDTO}) can be configured between 10 ms and 1280 ms (typ.) using the Watchdog Register (WDR).

During Low Power mode, the watchdog feature is not active, a D2D read during Stop mode will have the WDOFF bit set.

To clear the watchdog counter, a alternating write must be performed to the Watchdog Service Register (WDSR). The first write after the RESET_A has been released has to be 0xAA. The next one must be 0x55.

After the RESET_A has been released, there will be a standard (non-window) watchdog active with a fixed timeout of t_{IWDTO}. The Watchdog Window Open (WDWO) bit is set during that time and the window watchdog can be configured (WDR) without changing the initial timeout, and can be trimmed using the trim value given in the MCU trimming Flash section. See Section 4.25, “MM912F634 - Analog Die Trimming".

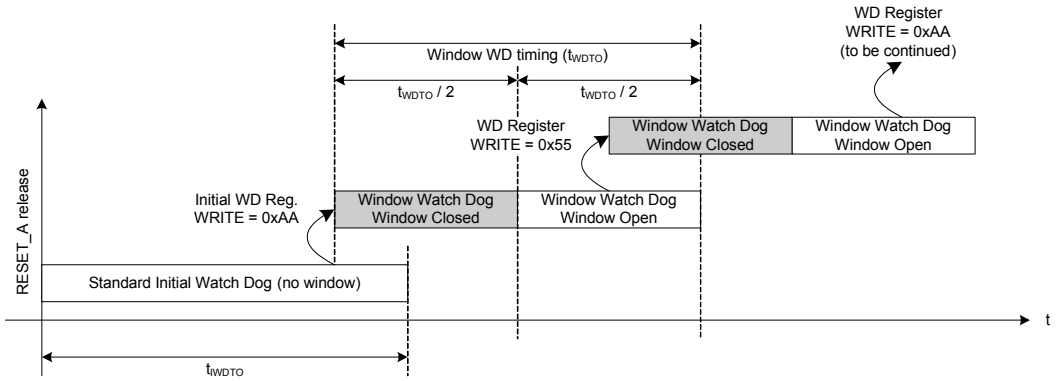

Figure 21. MM912F634 Analog Die Watchdog Operation

To enable the window watchdog, the initial counter reset has to be performed by writing 0xAA to the Watchdog Service Register (WDSR) before t_{IWDTO} is reached.

If the t_{IWDTO} timeout is reached with no counter reset or a value different from 0xAA was written to the WDSR, a watchdog reset will occur.

Once entering Window Watchdog mode, the first half of the time t_{WDTO} forbids a counter reset. To reset the watchdog counter, an alternating write of 0x55 and 0xAA must be performed within the second half of the t_{WDTO}. A Window Open (WDWO) flag will indicate the current status of the window. A timeout or wrong value written to the WDSR will force a watchdog reset.

For debug purpose, the watchdog can be completely disabled by applying V_{TST} to the TCLK pin while TEST_A is grounded. The watchdog will be disabled as long as V_{TST} is present. The watchdog is guaranteed functional for V_{TSTEN}. The WDOFF bit will indicate the watchdog being disabled. The WDSR register will reset to default once the watchdog is disabled. Once the watchdog is re-enabled, the initial watchdog sequence has to be performed.

During Low Power mode, the Watchdog clock is halted and the Watchdog Service Register (WDSR) is reset to the default state.

4.12.4 Functional Description

The Low Side switches are controlled by the bits LS1:2 in the Low Side Control Register (LSCR). In order to control the Low Sides, the LSCEN register has to be correctly written once after RESET or VROV.

To protect the device against over-voltage when an inductive load (relay) is turned off an active clamp circuit is implemented.

4.12.4.1 Voltage Regulator Over-voltage Protection

NOTE

The over-voltage threshold has to be trimmed at system power up. Please refer to Section 4.25.1.2.3, “Trimming Register 2 (CTR2)" for details. The default trim is worst case and may have disabled the LS function already. An initial LS enable would be needed.

To protect the application for an unintentional activation of the drivers in case of a voltage regulator over-voltage failure, the Low Side Drivers will automatically shut down in case of an over-voltage on one of the two regulators.

The shutdown is fully handled in the analog section of the driver. This will secure the feature in case the digital logic is damaged due to the over-voltage condition.

Once an over-voltage condition on one of the voltage regulators occurs, the LSx control bits in the Low Side Control Register (LSCR) will be reset to 0. The Voltage Regulator Over-voltage Condition Bit (VROVC) in the Voltage Status Register (VSR) will stay set as long as the condition is present. If the Voltage Regulator Over-voltage Interrupt was enabled (VROVIE=1), the VROV-Interrupt will be issued. Reading the Voltage Regulator Over-voltage Condition Bit (VROVC) in the Voltage Status Register (VSR) will clear the interrupt. To issue another VROV - Interrupt, the condition has to vanish and be present again.

To re-enable the Low Side Drivers after a Voltage Regulator Over-voltage condition occurred, first the LSCEN register has to be written with “0x05” - this information is processed through the main digital blocks, and would secure a minimum functionality before enabling the LS drivers again. In a second step, the LSx Control Bits in the Low Side Control Register (LSCR) must be enabled again after the over-voltage condition has vanished (VROVC=0).

4.12.4.2 Open Load Detection

Each Low Side driver signals an open load condition if the current through the Low Side is below the open load current threshold. The open load condition is indicated with the bits LS1OL and LS2OL in the Low Side Status Register (LSSR).

When the Low Side is in OFF state, the Open Load Detection function is not operating. When reading the LSSR register while the Low Side is operating in PWM and is in the OFF state, the LS1OL and LS2OL bits will not indicate Open Load.

4.12.4.3 Current Limitation

Each Low Side driver has a current limitation. In combination with the over-temperature shutdown, the Low Side drivers are protected against over-current and short-circuit failures.

The driver operates in current limitation, and is indicated with the bits LS1CL and LS2CL in the Low Side Status Register (LSSR).

Note: If the drivers is operating in current limitation mode excessive power might be dissipated.

4.12.4.4 Over-temperature Protection (LS Interrupt)

Both Low Side drivers are protected against over-temperature. In case of an over-temperature condition, both Low Side drivers are shut down and the event is latched in the Interrupt Control Module. The shutdown is indicated as LS Interrupt in the Interrupt Source Register (ISR).

If the bit LSM is set in the Interrupt Mask Register (IMR) than an Interrupt (IRQ) is generated.

A write to the Low Side Control Register (LSCR) will re-enable the Low Side drivers when the over-temperature condition is gone.

4.15.1.3 Block Diagram

Figure 31 shows the transmitter portion of the SCI.

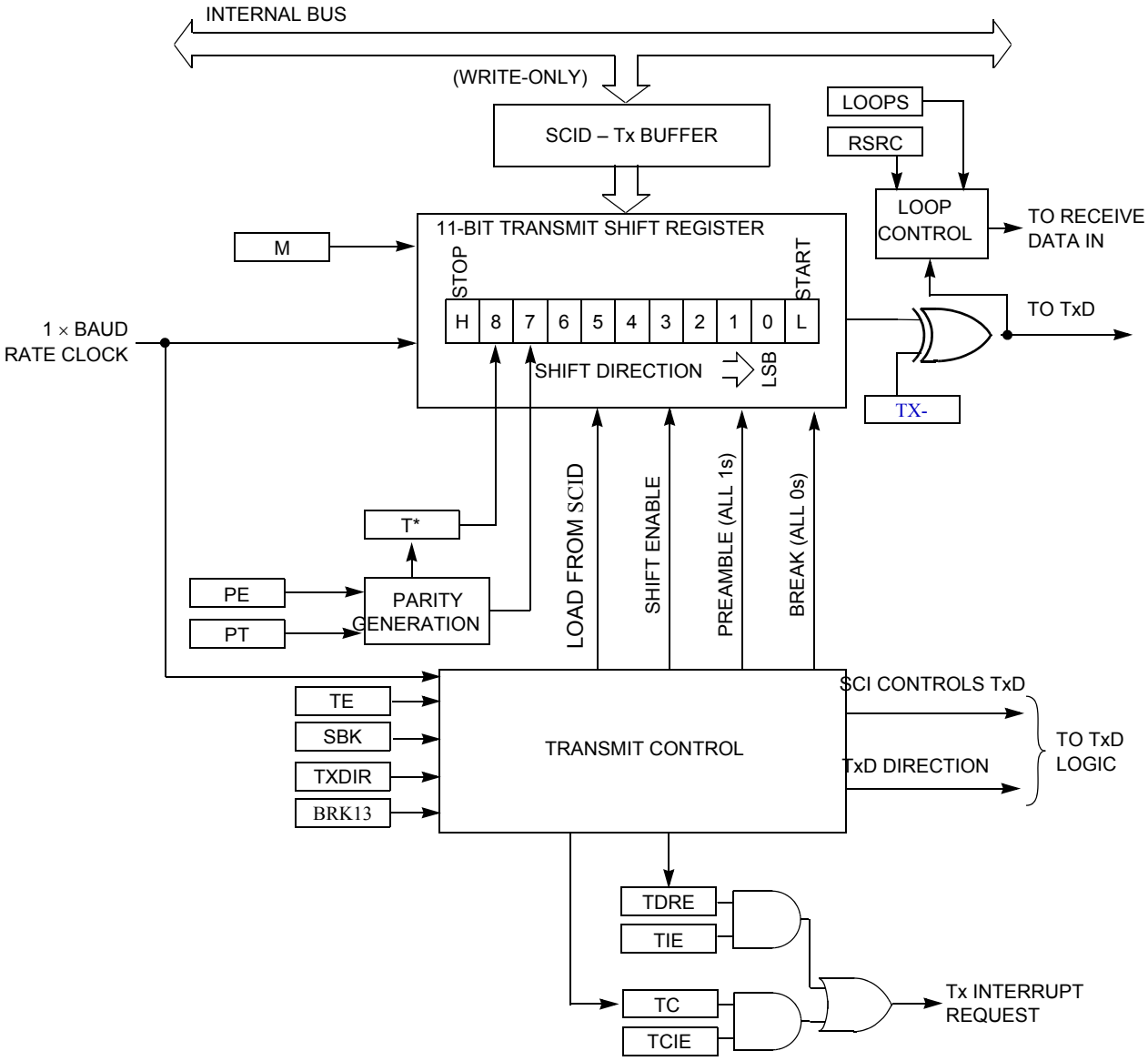

Figure 31. SCI Transmitter Block Diagram

Table 163. TSCR1 - Register Field Descriptions

Field	Description
7 TEN	Timer Enable 1 = Enables the timer. 0 = Disables the timer. (Used for reducing power consumption).
4 TFFCA	Timer Fast Flag Clear All 1 = For TFLG1 register, a read from an input capture or a write to the output compare channel [TC 3:0] causes the corresponding channel flag, CnF, to be cleared.For TFLG2 register, any access to the TCNT register clears the TOF flag. Any access to the PACNT registers clears the PAOVF and PAIF bits in the PAFLG register. This has the advantage of eliminating software overhead in a separate clear sequence. Extra care is required to avoid accidental flag clearing due to unintended accesses. 0 = Allows the timer flag clearing.

4.18.3.3.7 Timer Toggle On Overflow Register 1 (TTOV)

NOTE

TOVn toggles output compare pin on overflow. This feature only takes effect when the corresponding channel is configured for an output compare mode. When set, an overflow toggle on the output compare pin takes precedence over forced output compare events.

Table 164. Timer Toggle On Overflow Register 1 (TTOV)

Offset[121] 0xC7 Access: User read/write

	7	6	5	4	3	2	1	0
R	0	0	0	0	TOV3	TOV2	TOV1	TOV0
W								
Reset	0	0	0	0	0	0	0	0

Note:
121. Offset related to 0x0200 for blocking access and 0x300 for non blocking access within the global address space.

Table 165. TTOV - Register Field Descriptions

Field	Description
3-0 TOV[3-0]	Toggle On Overflow Bits 1 = Toggle output compare pin on overflow feature enabled. 0 = Toggle output compare pin on overflow feature disabled.

4.18.3.3.8 Timer Control Register 1 (TCTL1)

NOTE

These four pairs of control bits are encoded to specify the output action to be taken as a result of a successful Output Compare on “n” channel. When either OMn or OLn, the pin associated with the corresponding channel becomes an output tied to its IOC. To enable output action by the OMn and OLn bits on a timer port, the corresponding bit in OC3M should be cleared.

Table 166. Timer Control Register 1 (TCTL1)

Offset[122] 0xC8 Access: User read/write

	7	6	5	4	3	2	1	0
R W	OM3	OL3	OM2	OL2	OM1	OL1	OM0	OL0
Reset	0	0	0	0	0	0	0	0

Note:
122. Offset related to 0x0200 for blocking access and 0x300 for non blocking access within the global address space.

Table 167. TCTL1 - Register Field Descriptions

Field	Description
7,5,3,1 OMn	Output Mode bit
6,4,2,0 OLn	Output Level bit

Table 168. Compare Result Output Action

OMn	OLn	Action
0	0	Timer disconnected from output pin logic
0	1	Toggle OCn output line
1	0	Clear OCn output line to zero
1	1	Set OCn output line to one

4.18.3.3.9 Timer Control Register 2 (TCTL2)

Table 169. Timer Control Register 2 (TCTL2)

Offset 0xC9 Access: User read/write

	7	6	5	4	3	2	1	0
R W	EDG3B	EDG3A	EDG2B	EDG2A	EDG1B	EDG1A	EDG0B	EDG0A
Reset	0	0	0	0	0	0	0	0

Note:
123. [123]Offset related to 0x0200 for blocking access and 0x300 for non blocking access within the global address space.

Table 170. TCTL2 - Register Field Descriptions

Field	Description
EDGnB,EDGnA	Input Capture Edge Control

These four pairs of control bits configure the input capture edge detector circuits.

Table 247. Global Implemented Memory Space

Internal Resource	Bottom Address	Top Address
Registers	0x0_0000	0x0_03FF
IFR (if MMCCTL1.IFRON == 1'b1)	0x0_0400	0x0_047F
Unimplemented (if MMCCTL1.IFRON == 1'b0)		
Unimplemented	0x0_0480	0x0_07FF
RAM	0x0_0800	RAM_HIGH = 0x0_07FF plus RAMSIZE(169)
FLASH	FLASH_LOW = 0x4_0000 minus FLASHSIZE(170)	0x3_FFFF

Note:
169. RAMSIZE is the hexadecimal value of RAM SIZE in bytes.
170. FLASHSIZE is the hexadecimal value of FLASH SIZE in bytes.

In single-chip modes accesses by the CPU (except for firmware commands) to any of the unimplemented areas (see Figure 44) will result in an illegal access reset (system reset). BDM accesses to the unimplemented areas are allowed but the data will be undefined.

No misaligned word access from the BDM module will occur; these accesses are blocked in the BDM module (Refer to BDM Block Guide).

Figure 64 shows the host receiving a logic 0 from the target. Since the host is asynchronous to the target, there is up to a one clock-cycle delay from the host-generated falling edge on BKGD to the start of the bit time as perceived by the target. The host initiates the bit time but the target finishes it. Since the target wants the host to receive a logic 0, it drives the BKGD pin low for 13 target clock cycles then briefly drives it high to speed up the rising edge. The host samples the bit level about 10 target clock cycles after starting the bit time.

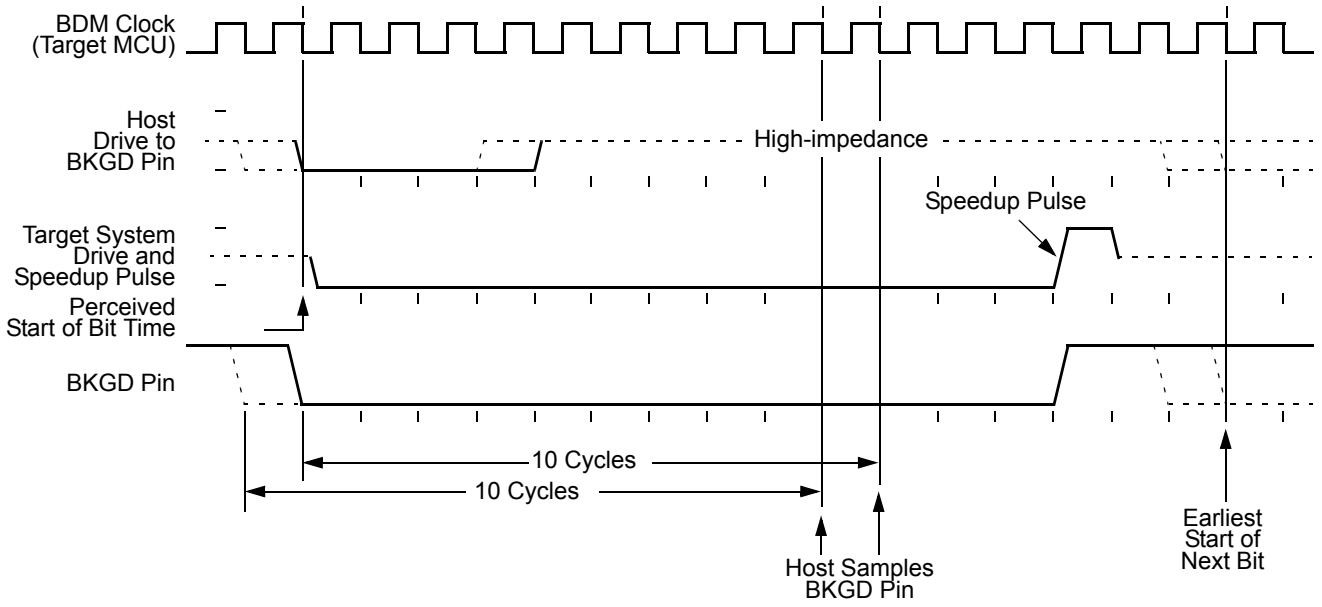

Figure 64. BDM Target-to-Host Serial Bit Timing (Logic 0)

4.30.4.7 Serial Interface Hardware Handshake Protocol

NOTE

If the ACK pulse was issued by the target, the host assumes the previous command was executed. If the CPU enters wait or stop prior to executing a hardware command, the ACK pulse will not be issued meaning that the BDM command was not executed. After entering wait or stop mode, the BDM command is no longer pending.

BDM commands that require CPU execution are ultimately treated at the MCU bus rate. Since the BDM clock source can be modified relative to the bus clock, when modifying DCO clock or the bus clock divider, it is very helpful to provide a handshake protocol in which the host could determine when an issued command is executed by the CPU. The alternative is to always wait the amount of time equal to the appropriate number of cycles at the slowest possible rate the clock could be running. This sub-section will describe the hardware handshake protocol.

The hardware handshake protocol signals to the host controller when an issued command was successfully executed by the target. This protocol is implemented by a 16 serial clock cycle low pulse followed by a brief speedup pulse in the BKGD pin. This pulse is generated by the target MCU when a command, issued by the host, has been successfully executed (see Figure 65). This pulse is referred to as the ACK pulse. After the ACK pulse has finished: the host can start the bit retrieval if the last issued command was a read command, or start a new command if the last command was a write command or a control command (BACKGROUND, GO, GO_UNTIL[169] or TRACE1). The ACK pulse is not issued earlier than 32 serial clock cycles after the BDM command was issued. The end of the BDM command is assumed to be the 16th tick of the last bit. This minimum delay assures enough time for the host to perceive the ACK pulse. Note also that, there is no upper limit for the delay between the command and the related ACK pulse, since the command execution depends upon the CPU bus, which in some cases could be very slow due to long accesses taking place.This protocol allows a great flexibility for the POD designers, since it does not rely on any accurate time measurement or short response time to any event in the serial communication.

Table 268. TRCMOD Trace Mode Bit Encoding (continued)

TRCMOD	Description
01	Loop1
10	Detail
11	Pure PC

4.31.3.2.4 Debug Control Register2 (DBGC2)

Table 269. Debug Control Register2 (DBGC2)

Address: 0x0023

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

Read: Anytime

Write: Anytime the module is disarmed.

This register configures the comparators for range matching.

Table 270. DBGC2 Field Descriptions

Field	Description
1–0 ABCM[1:0]	**A and B Comparator Match Control** — These bits determine the A and B comparator match mapping as described in Table 271.

Table 271. ABCM Encoding

ABCM	Description
00	Match0 mapped to comparator A match: Match1 mapped to comparator B match.
01	Match 0 mapped to comparator A/B inside range: Match1 disabled.
10	Match 0 mapped to comparator A/B outside range: Match1 disabled.
11	Reserved(181)

Note:
181. Currently defaults to Comparator A, Comparator B disabled.

4.31.3.2.5 Debug Trace Buffer Register (DBGTBH:DBGTBL)

Table 272. Debug Trace Buffer Register (DBGTB)

Address: 0x0024, 0x0025

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
W																
POR	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
Other Resets	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—

Read: Only when unlocked AND unsecured AND not armed AND TSOURCE set.

4.31.3.2.7.4 Debug Match Flag Register (DBGMFR)

Table 287. Debug Match Flag Register (DBGMFR)

Address: 0x0027

	7	6	5	4	3	2	1	0
R	0	0	0	0	0	MC2	MC1	MC0
W								
Reset	0	0	0	0	0	0	0	0

Read: If COMRV[1:0] = 11

Write: Never

DBGMFR is visible at 0x0027 only with COMRV[1:0] = 11. It features 3 flag bits each mapped directly to a channel. Should a match occur on the channel during the debug session, then the corresponding flag is set and remains set until the next time the module is armed by writing to the ARM bit. Thus the contents are retained after a debug session for evaluation purposes. These flags cannot be cleared by software, they are cleared only when arming the module. A set flag does not inhibit the setting of other flags. Once a flag is set, further comparator matches on the same channel in the same session have no affect on that flag.

4.31.3.2.8 Comparator Register Descriptions

Each comparator has a bank of registers that are visible through an 8-byte window in the DBG module register address map. Comparator A consists of 8 register bytes (3 address bus compare registers, two data bus compare registers, two data bus mask registers and a control register). Comparator B consists of four register bytes (three address bus compare registers and a control register). Comparator C consists of four register bytes (three address bus compare registers and a control register).

Each set of comparator registers can be accessed using the COMRV bits in the DBGC1 register. Unimplemented registers (e.g. Comparator B data bus and data bus masking) read as zero and cannot be written. The control register for comparator B differs from those of comparators A and C.

Table 288. Comparator Register Layout

0x0028	CONTROL	Read/Write	Comparators A,B and C
0x0029	ADDRESS HIGH	Read/Write	Comparators A,B and C
0x002A	ADDRESS MEDIUM	Read/Write	Comparators A,B and C
0x002B	ADDRESS LOW	Read/Write	Comparators A,B and C
0x002C	DATA HIGH COMPARATOR	Read/Write	Comparator A only
0x002D	DATA LOW COMPARATOR	Read/Write	Comparator A only
0x002E	DATA HIGH MASK	Read/Write	Comparator A only
0x002F	DATA LOW MASK	Read/Write	Comparator A only

Table 292. DBGXCTL Field Descriptions (continued)

Field	Description
5 TAG	**Tag Select**— This bit controls whether the comparator match has immediate effect, causing an immediate state sequencer transition or tag the opcode at the matched address. Tagged opcodes trigger only if they reach the execution stage of the instruction queue. 0 Allow state sequencer transition immediately on match 1 On match, tag the opcode. If the opcode is about to be executed allow a state sequencer transition
4 BRK	**Break**— This bit controls whether a comparator match terminates a debug session immediately, independent of state sequencer state. To generate an immediate breakpoint the module breakpoints must be enabled using the DBGC1 bit DBGBRK. 0 The debug session termination is dependent upon the state sequencer and trigger conditions. 1 A match on this channel terminates the debug session immediately; breakpoints if active are generated, tracing, if active, is terminated and the module disarmed.
3 RW	**Read/Write Comparator Value Bit** — The RW bit controls whether read or write is used in compare for the associated comparator. The RW bit is not used if RWE = 0. This bit is ignored if the TAG bit in the same register is set. 0 Write cycle will be matched 1 Read cycle will be matched
2 RWE	**Read/Write Enable Bit** — The RWE bit controls whether read or write comparison is enabled for the associated comparator.This bit is ignored if the TAG bit in the same register is set 0 Read/Write is not used in comparison 1 Read/Write is used in comparison
0 COMPE	Determines if comparator is enabled 0 The comparator is not enabled 1 The comparator is enabled

Table 293 shows the effect for RWE and RW on the comparison conditions. These bits are ignored if the corresponding TAG bit is set since the match occurs based on the tagged opcode reaching the execution stage of the instruction queue.

Table 293. Read or Write Comparison Logic Table

RWE Bit	RW Bit	RW Signal	Comment
0	x	0	RW not used in comparison
0	x	1	RW not used in comparison
1	0	0	Write data bus
1	0	1	No match
1	1	0	No match
1	1	1	Read data bus

4.31.3.2.8.2 Debug Comparator Address High Register (DBGXAH)

Table 294. Debug Comparator Address High Register (DBGXAH)

Address: 0x0029

	7	6	5	4	3	2	1	0
R	0	0	0	0	0	0	Bit 17	Bit 16
W								
Reset	0	0	0	0	0	0	0	0

The DBGC1_COMRV bits determine which comparator address registers are visible in the 8-byte window from 0x0028 to 0x002F.

Table 295. Comparator Address Register Visibility

COMRV	Visible Comparator
00	DBGAAH, DBGAAM, DBGAAL

4.31.4.5.3.1 Loop1 Mode

NOTE

In certain very tight loops, the source address will have already been fetched again before the background comparator is updated. This results in the source address being stored twice before further duplicate entries are suppressed. This condition occurs with branch-on-bit instructions when the branch is fetched by the first P-cycle of the branch or with loop-construct instructions in which the branch is fetched with the first or second P cycle. See examples below:

Loop1 mode, similarly to Normal mode also stores only COF address information to the trace buffer, it however allows the filtering out of redundant information.

The intent of Loop1 mode is to prevent the Trace Buffer from being filled entirely with duplicate information from a looping construct such as delays using the DBNE instruction or polling loops using BRSET/BRCLR instructions. Immediately after address information is placed in the Trace Buffer, the DBG module writes this value into a background register. This prevents consecutive duplicate address entries in the Trace Buffer resulting from repeated branches.

Loop1 mode only inhibits consecutive duplicate source address entries that would typically be stored in most tight looping constructs. It does not inhibit repeated entries of destination addresses or vector addresses, since repeated entries of these would most likely indicate a bug in the user's code that the DBG module is designed to help find.

```
LOOP    INX                     ; 1-byte instruction fetched by 1st P-cycle of BRCLR
        BRCLR       CMPTMP,#$0c,LOOP ; the BRCLR instruction also will be fetched by 1st
                                ; P-cycle of BRCLR

LOOP2   BRN     *               ; 2-byte instruction fetched by 1st P-cycle of DBNE
        NOP                     ; 1-byte instruction fetched by 2nd P-cycle of DBNE
        DBNE        A,LOOP2     ; this instruction also fetched by 2nd P-cycle of DBNE
```

4.31.4.5.3.2 Detail Mode

In Detail Mode, address and data for all memory and register accesses is stored in the trace buffer. This mode is intended to supply additional information on indexed, indirect addressing modes, where storing only the destination address would not provide all information required for a user to determine where the code is in error. This mode also features information bit storage to the trace buffer, for each address byte storage. The information bits indicates the size of access (word or byte) and the type of access (read or write).

When tracing in Detail mode, all cycles are traced except those when the CPU is either in a free or opcode fetch cycle.

4.31.4.5.3.3 Pure PC Mode

NOTE:

When tracing is terminated using forced breakpoints, latency in breakpoint generation means that opcodes following the opcode causing the breakpoint can be stored to the trace buffer. The number of opcodes is dependent on program flow. This should be avoided by using tagged breakpoints.

In Pure PC mode, tracing from the CPU the PC addresses of all executed opcodes, including illegal opcodes are stored.

4.32.5.2 Description of Reset Operation

NOTE

External circuitry connected to the RESET pin should not include a large capacitance that would interfere with the ability of this signal to rise to a valid logic one within 256 DCO Clock cycles after the low drive is released.

The reset sequence is initiated by any of the following events:

- Low level is detected at the RESET pin (External Reset).
- Power-on is detected.
- Illegal Address Access is detected (see MMC Block Guide for details).
- COP watchdog times out.
- Oscillator monitor failure is detected.

Upon detection of any reset event, an internal circuit drives the RESET pin low for 516 DCO Clock cycles. Depending on internal synchronization latency, it can also be 517 DCO Clock cycles (see Figure 77). Since entry into reset is asynchronous, it does not require a running DCO Clock. However, the internal reset circuit of the 9S12I32PIMV1 cannot sequence out of current reset condition without a running DCO Clock. After 516 DCO Clock cycles, the RESET pin is released. The reset generator of the 9S12I32PIMV1 waits for additional 256 DCO Clock cycles and then samples the RESET pin to determine the originating source. Table 334 shows which vector will be fetched.

Table 334. Reset Vector Selection

Sampled RESET Pin (256 cycles after release)	Oscillator monitor fail pending	COP timeout pending	Vector Fetch
1	0	0	POR /Illegal Address Access/External pin RESET
1	1	X	Oscillator Monitor Fail
1	0	1	COP time out
0	X	X	POR /Illegal Address Access/ External pin RESET

The internal reset of the MCU remains asserted while the reset generator completes the 768 DCO Clock long reset sequence. In case the RESET pin is externally driven low for more than these 768 DCO Clock cycles (External Reset), the internal reset remains asserted longer.

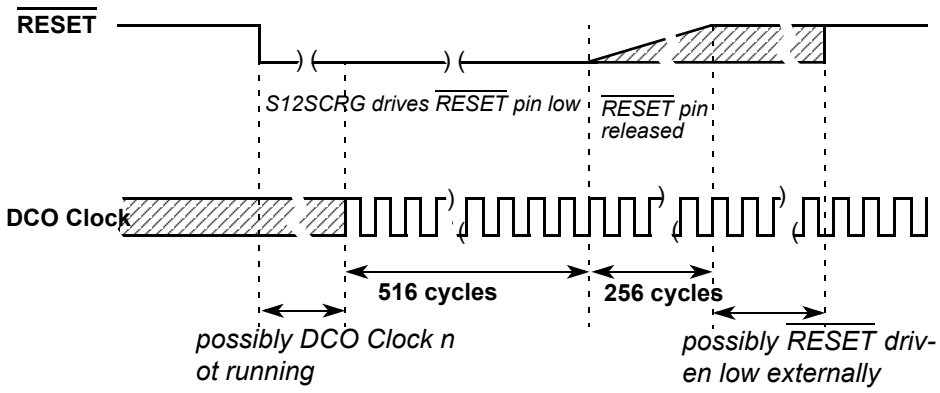

Figure 77. RESET Timing

4.32.5.2.1 Oscillator Monitor Reset

In case of loss of clock, or the oscillator frequency is below the failure assert frequency f_{OMFA} (see device electrical characteristics for values), the 9S12I32PIMV1 generates a Oscillator Monitor Reset.

4.34 Real Time Interrupt (S12SRTIV1)

4.34.1 Introduction

This section describes the functionality of the Real Time Interrupt module (RTI), a sub-block of the HCS12S core platform. The RTI (free running real time interrupt) enables the user to generate a hardware interrupt at a fixed periodic rate. If RTI is enabled, the interrupt will occur at the rate selected by the RTICTL and RTICNT register.

The RTI counter is clocked by the internal reference clock. At the end of the RTI timeout period the RTIF flag is set to one and a new RTI timeout period starts immediately.

The RTI contains two asynchronous clock domains (one for the Modulus Down Counter/Prescaler and one for the register bank). Information exchange between both clock domains is fully synchronized. Therefore modification of the RTI timeout period must be done in appliance to the write protection rules.

4.34.2 Overview

A block diagram of the RTI is shown in Figure 81

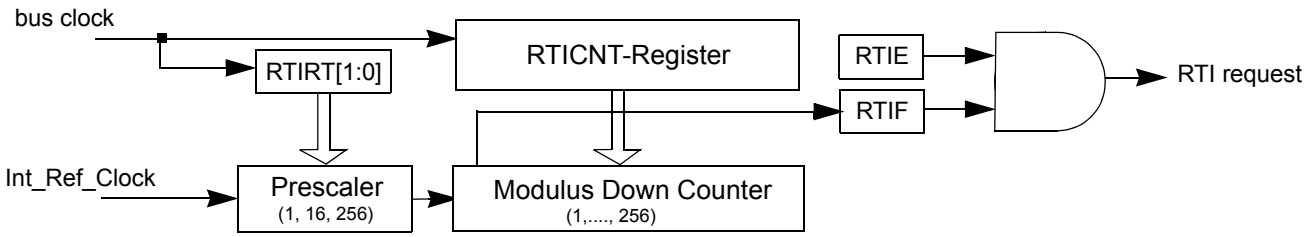

Figure 81. Block Diagram

4.34.3 Features

The RTI includes these distinctive features:

- Generate hardware interrupt at a fixed periodic rate
- Software selectable RTI operation in WAIT and STOP mode
- Software selectable RTI freeze during BDM active mode

4.34.4 Modes of Operation

- Run Mode
  If RTI functionality is required, the individual bits (RTIRT) of the associated rate select registers (RTICTL) have to be set to a non-zero value. In addition, to generate RTI requests, the RTI must be enabled (RTIE bit set). The RTI counter is stopped if all rate select bits in the RTICTL register are zero. Interrupt requests will be disabled if the corresponding bit (RTIE) is cleared.
- Wait mode
  If the respective enable bit (RTISWAI) is cleared, the RTI will continue to run, else RTI will remain frozen.
- Stop mode
  If the respective enable bit (RTIRSTP) is set, the RTI will continue to run, else RTI will remain frozen.

4.34.5 External Signal Description

There are no external signals associated with this module.

Table 374. Flash Address Low Register (FADDRLO - Special Mode) (continued)

Reset	0	0	0	0	0	0	0	0

All FADDR bits read 0 and are not writable in normal mode.

All assigned FADDR bits are readable and writable in special mode.

4.36.3.4.4 Flash Data Registers (FDATA)

The FDATA registers are the Flash data registers.

Table 375. Flash Data High Register (FDATAHI - Normal Mode)

0x010A

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

Table 376. Flash Data Low Register (FDATALO - Normal Mode)

0x010B

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

Table 377. Flash Data High Register (FDATAHI - Special Mode)

0x010A

	7	6	5	4	3	2	1	0
R W	FD[15:8]							
Reset	0	0	0	0	0	0	0	0

Table 378. Flash Data Low Register (FDATALO - Special Mode)

0x010B

	7	6	5	4	3	2	1	0
R W	FD[7:0]							
Reset	0	0	0	0	0	0	0	0

All FDATA bits read 0 and are not writable in normal mode.

All FDATA bits are readable and writable in special mode. The FDATA bits are indirectly written to when writing to an address within the Flash block, as part of a command write sequence.

Table 409. Example SPI Baud Rate Selection (20 MHz Bus Clock) (continued)

SPPR2	SPPR1	SPPR0	SPR2	SPR1	SPR0	Baud Rate Divisor	Baud Rate
0	0	0	0	1	1	16	1.25 MHz
0	0	0	1	0	0	32	625.00 kHz
0	0	0	1	0	1	64	312.50 kHz
0	0	0	1	1	0	128	156.25 kHz
0	0	0	1	1	1	256	78.13 kHz
0	0	1	0	0	0	4	5.00 MHz
0	0	1	0	0	1	8	2.50 MHz
0	0	1	0	1	0	16	1.25 MHz
0	0	1	0	1	1	32	625.00 kHz
0	0	1	1	0	0	64	312.50 kHz
0	0	1	1	0	1	128	156.25 kHz
0	0	1	1	1	0	256	78.13 kHz
0	0	1	1	1	1	512	39.06 kHz
0	1	0	0	0	0	6	3.33 MHz
0	1	0	0	0	1	12	1.66 MHz
0	1	0	0	1	0	24	833.33 kHz
0	1	0	0	1	1	48	416.67 kHz
0	1	0	1	0	0	96	208.33 kHz
0	1	0	1	0	1	192	104.17 kHz
0	1	0	1	1	0	384	52.08 kHz
0	1	0	1	1	1	768	26.04 kHz
0	1	1	0	0	0	8	2.50 MHz
0	1	1	0	0	1	16	1.25 MHz
0	1	1	0	1	0	32	625.00 kHz
0	1	1	0	1	1	64	312.50 kHz
0	1	1	1	0	0	128	156.25 kHz
0	1	1	1	0	1	256	78.13 kHz
0	1	1	1	1	0	512	39.06 kHz
0	1	1	1	1	1	1024	19.53 kHz
1	0	0	0	0	0	10	2.00 MHz
1	0	0	0	0	1	20	1.00 MHz
1	0	0	0	1	0	40	500.00 kHz
1	0	0	0	1	1	80	250.00 kHz
1	0	0	1	0	0	160	125.00 kHz
1	0	0	1	0	1	320	62.50 kHz
1	0	0	1	1	0	640	31.25 kHz
1	0	0	1	1	1	1280	15.63 kHz
1	0	1	0	0	0	12	1.66 kHz
1	0	1	0	0	1	24	833.33 kHz
1	0	1	0	1	0	48	416.67 kHz
1	0	1	0	1	1	96	208.33 kHz
1	0	1	1	0	0	192	104.17 kHz

4.38.4.2 Slave Mode

NOTE

When peripherals with duplex capability are used, take care not to simultaneously enable two receivers whose serial outputs drive the same system slave's serial data output line.

The SPI operates in slave mode when the MSTR bit in SPI control register 1 is clear.

- Serial clock
  In slave mode, SCK is the SPI clock input from the master.
- MISO, MOSI pin
  In slave mode, the function of the serial data output pin (MISO) and serial data input pin (MOSI) is determined by the SPC0 bit and BIDIROE bit in SPI control register 2.
- $\overline{SS}$ pin
  The $\overline{SS}$ pin is the slave select input. Before a data transmission occurs, the $\overline{SS}$ pin of the slave SPI must be low. $\overline{SS}$ must remain low until the transmission is complete. If $\overline{SS}$ goes high, the SPI is forced into idle state.
  The $\overline{SS}$ input also controls the serial data output pin, if $\overline{SS}$ is high (not selected), the serial data output pin is high impedance, and, if $\overline{SS}$ is low, the first bit in the SPI data register is driven out of the serial data output pin. Also, if the slave is not selected ($\overline{SS}$ is high), then the SCK input is ignored and no internal shifting of the SPI shift register occurs.
  Although the SPI is capable of duplex operation, some SPI peripherals are capable of only receiving SPI data in a slave mode. For these simpler devices, there is no serial data out pin.

As long as no more than one slave device drives the system slave's serial data output line, it is possible for several slaves to receive the same transmission from a master, although the master would not receive return information from all of the receiving slaves.

If the CPHA bit in SPI control register 1 is clear, odd numbered edges on the SCK input cause the data at the serial data input pin to be latched. Even numbered edges cause the value previously latched from the serial data input pin to shift into the LSB or MSB of the SPI shift register, depending on the LSBFE bit.

If the CPHA bit is set, even numbered edges on the SCK input cause the data at the serial data input pin to be latched. Odd numbered edges cause the value previously latched from the serial data input pin to shift into the LSB or MSB of the SPI shift register, depending on the LSBFE bit.

NOTE

A change of the CPOL, CPHA, SSOE, LSBFE, MODFEN, SPC0, or BIDIROE bits with SPC0 set in slave mode will corrupt a transmission in progress and must be avoided.

When CPHA is set, the first edge is used to get the first data bit onto the serial data output pin. When CPHA is clear and the $\overline{SS}$ input is low (slave selected), the first bit of the SPI data is driven out of the serial data output pin. After the eighth shift, the transfer is considered complete and the received data is transferred into the SPI data register. To indicate transfer is complete, the SPIF flag in the SPI status register is set.

4.38.4.3 Transmission Formats

During a SPI transmission, data is transmitted (shifted out serially) and received (shifted in serially) simultaneously. The serial clock (SCK) synchronizes shifting and sampling of the information on the two serial data lines. A slave select line allows selection of an individual slave SPI device; slave devices that are not selected do not interfere with SPI bus activities. Optionally, on a master SPI device, the slave select line can be used to indicate multiple-master bus contention.

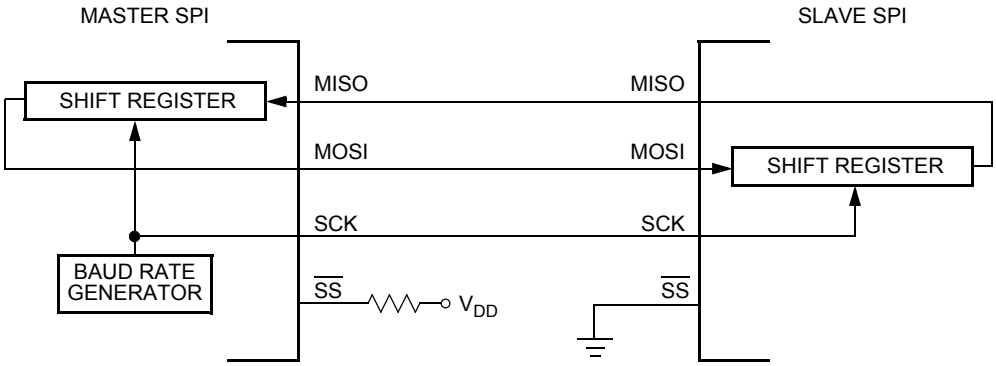

Figure 102. Master/Slave Transfer Block Diagram

4.38.4.3.1 Clock Phase and Polarity Controls

Using two bits in the SPI control register 1, software selects one of four combinations of serial clock phase and polarity.

The CPOL clock polarity control bit specifies an active high or low clock and has no significant effect on the transmission format.

The CPHA clock phase control bit selects one of two fundamentally different transmission formats.

Clock phase and polarity should be identical for the master SPI device and the communicating slave device. In some cases, the phase and polarity are changed between transmissions to allow a master device to communicate with peripheral slaves having different requirements.

4.38.4.3.2 CPHA = 0 Transfer Format

The first edge on the SCK line is used to clock the first data bit of the slave into the master, and the first data bit of the master into the slave. In some peripherals, the first bit of the slave's data is available at the slave's data out pin as soon as the slave is selected. In this format, the first SCK edge is issued a half cycle after $\overline{SS}$ has become low.

A half SCK cycle later, the second edge appears on the SCK line. When this second edge occurs, the value previously latched from the serial data input pin is shifted into the LSB or MSB of the shift register, depending on LSBFE bit.

After this second edge, the next bit of the SPI master data is transmitted out of the serial data output pin of the master to the serial input pin on the slave. This process continues for a total of 16 edges on the SCK line, with data being latched on odd numbered edges and shifted on even numbered edges.

Data reception is double buffered. Data is shifted serially into the SPI shift register during the transfer, and is transferred to the parallel SPI data register after the last bit is shifted in.

After the 16th (last) SCK edge:

- Data that was previously in the master SPI data register should now be in the slave data register, and the data that was in the slave data register should be in the master.
- The SPIF flag in the SPI status register is set, indicating that the transfer is complete.