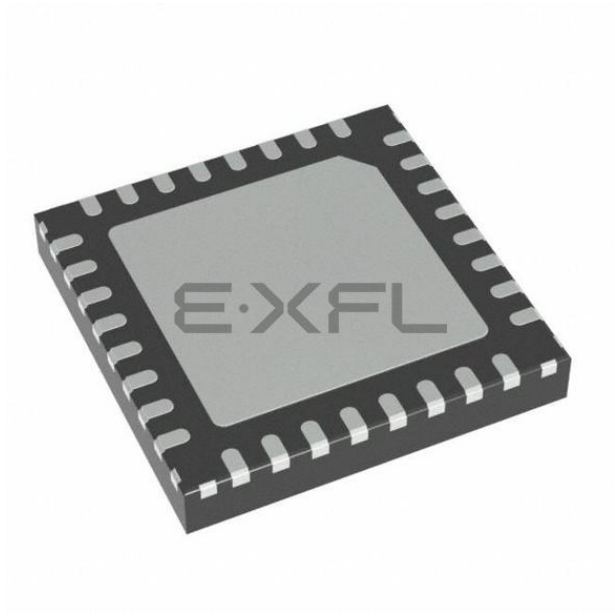

Welcome to E-XFL.COM

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	S12Z
Core Size	16-Bit
Speed	32MHz
Connectivity	I²C, IrDA, LINbus, SCI, SPI, UART/USART
Peripherals	LVD, POR, PWM, WDT
Number of I/O	19
Program Memory Size	16KB (16K x 8)
Program Memory Type	FLASH
EEPROM Size	128 x 8
RAM Size	1K x 8
Voltage - Supply (Vcc/Vdd)	5.5V ~ 18V
Data Converters	A/D 6x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 105°C (TA)
Mounting Type	Surface Mount, Wettable Flank
Package / Case	32-VFQFN Exposed Pad
Supplier Device Package	32-QFN-EP (5x5)
Purchase URL	https://www.e-xfl.com/product-detail/nxp-semiconductors/s9s12zvls1f0vfmr

Chapter 1
Device Overview MC9S12ZVL-Family

Table 1-1. Revision History

Version Number	Revision Date	Description of Changes
1.0	28. May 2013	• added feedback from shared review • changed IFR mapping table conditional text to make the ADC reference conversion visible for the customer
1.1	26. Aug.2013	• added feedback from shared review
1.2	29. Aug.2013	• update table 1-12, replaced for SCI0/1 EX with RX
1.3	19.Sep. 2013	• added chapter 1.13.2, “BDC Command Restriction”
1.4	2 April 2014	• fix findings from the shared review
1.5	5 Aug. 2014	• fix typo PMW -> PWM
1.6	24 Sep. 2014	• added the VL128 device
1.7	21 Oct. 2014	• added PWM channel muxing • added PGA - DAC - ACMP - ADC analog connections
1.8	03 Jun. 2015	• correct alignment in table 1-15
1.9	27 Oct. 2015	• correct table 1-3
2.0	08 Aug. 2017	• correct CTRL register name for PAD6-PAD9, table 1-6 • correct 32-pin QFN-EP availability, table 1-2

1.1 Introduction

The MC9S12ZVL-Family is an automotive 16-bit microcontroller family using the 180nm NVM + UHV technology that offers the capability to integrate 40V analog components. This family reuses many features from the existing S12 portfolio. The particular differentiating features of this family are the enhanced S12Z core and the integration of “high-voltage” analog modules, including the voltage regulator (VREG) and a Local Interconnect Network (LIN) physical layer.

The MC9S12ZVL-Family includes error correction code (ECC) on RAM, FLASH and EEPROM for diagnostic or data storage, a fast analog-to-digital converter (ADC) and a frequency modulated phase locked loop (IPLL) that improves the EMC performance. The MC9S12ZVL-Family delivers an optimized solution with the integration of several key system components into a single device, optimizing system architecture and achieving significant space savings. The MC9S12ZVL-Family delivers all the advantages and efficiencies of a 16-bit MCU while retaining the low cost, power consumption, EMC, and code-size efficiency advantages currently enjoyed by users of existing S12 families. The MC9S12ZVL-Family is available in 48-pin, 32-pin LQFP and 32-pin QFN-EP. In addition to the I/O ports available in each module, further I/O ports are available with interrupt capability allowing wake-up from stop or wait modes.

Global Address	Register Name		Bit 7	6	5	4	3	2	1	Bit 0
0x0314	PPSJ	R	0	0	0	0	0	0	PPSJ1	PPSJ0
		W								
0x0315–0x031E	Reserved	R	0	0	0	0	0	0	0	0
		W								
0x031F	WOMJ	R	0	0	0	0	0	0	WOMJ1	WOMJ0
		W								
0x0320–0x032F	Reserved	R	0	0	0	0	0	0	0	0
		W								
0x0330	Reserved	R	0	0	0	0	0	0	0	0
		W								
0x0331	PTIL	R	0	0	0	0	0	0	0	PTIL0
		W								
0x0332–0x0333	Reserved	R	0	0	0	0	0	0	0	0
		W								
0x0334	PPSL	R	0	0	0	0	0	0	0	PPSL0
		W								
0x0335	Reserved	R	0	0	0	0	0	0	0	0
		W								
0x0336	PIEL	R	0	0	0	0	0	0	0	PIEL0
		W								
0x0337	PIFL	R	0	0	0	0	0	0	0	PIFL0
		W								
0x0338–0x033B	Reserved	R	0	0	0	0	0	0	0	0
		W								
0x033C	DIENL	R	0	0	0	0	0	0	0	DIENL0
		W								
0x033D	PTAL	R	PTTEL	PTPSL	PTABYPL	PTADIRL	PTAENL	0	0	0
		W								
0x033E	PIRL	R	0	0	0	0	0	0	PIRL0[1:0]	
		W								

PP[5,3,1] connect the loads to the digital ground VSSX.

Similar protection mechanisms as for EVDD1 apply for PP[5,3,1] accordingly in an inverse way.

2.5.5 Open Input Detection on HVI

The connection of an external pull device on a high-voltage input can be validated by using the built-in pull functionality of the HVI. Depending on the application type an external pull-down circuit can be detected with the internal pull-up device whereas an external pull-up circuit can be detected with the internal pull-down device which is part of the input voltage divider.

Note that the following procedures make use of a function that overrides the automatic disable mechanism of the digital input buffer when using the HVI in analog mode. Make sure to switch off the override function when using the HVI in analog mode after the check has been completed.

External pull-down device (Figure 2-33):

1. Enable analog function on HVI in non-direct mode (PTAL[PTAENL]=1, PTAL[PTADIRL]=0)
2. Select internal pull-up device on HVI (PTAL[PTPSL]=1)
3. Enable function to force input buffer active on HVI in analog mode (PTAL[PTTEL]=1)
4. Verify PTIL=0 for a connected external pull-down device; read PTIL=1 for an open input

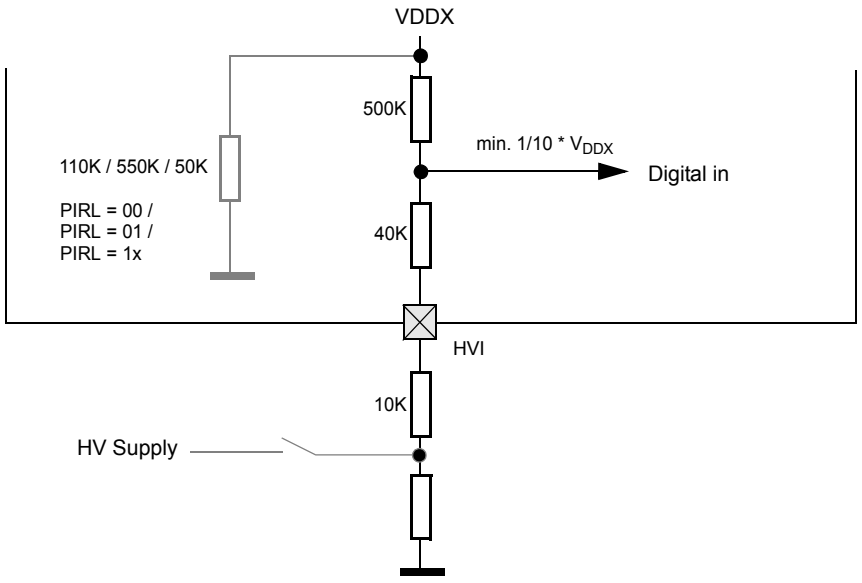

Figure 2-33. Digital Input Read with Pull-up Enabled

External pull-up device (Figure 2-34):

1. Enable analog function on HVI in non-direct mode (PTAL[PTAENL]=1, PTAL[PTADIRL]=0)
2. Select internal pulldown device on HVI (PTAL[PTPSL]=0)
3. Enable function to force input buffer active on HVI in analog mode (PTAL[PTTEL]=1)

Accesses to the internal memory map are not possible when the internal device clocks are disabled. Thus attempted accesses to memory mapped resources are suppressed and the NORESP flag is set. Resources can be accessed again by the next command received following exit from Stop mode.

A BACKGROUND command issued whilst in stop mode remains pending internally until the device leaves stop mode. This means that subsequent active BDM commands, issued whilst BACKGROUND is pending, set the ILLCMD flag because the device is not yet in active BDM.

If ACK handshaking is enabled, then the first ACK, following a stop mode entry is long to indicate a stop exception. The BDC indicates a stop mode occurrence by setting the BDCCSR bit STOP. If the host attempts further communication before the ACK pulse generation then the OVRUN bit is set.

STOP Mode With BDC Enabled And BDCCIS Set

If the BDC is enabled and BDCCIS is set, then the BDC prevents core clocks being disabled in stop mode. This allows BDC communication, for access of internal memory mapped resources, but not CPU registers, to continue throughout stop mode.

A BACKGROUND command issued whilst in stop mode remains pending internally until the device leaves stop mode. This means that subsequent active BDM commands, issued whilst BACKGROUND is pending, set the ILLCMD flag because the device is not yet in active BDM.

If ACK handshaking is enabled, then the first ACK, following a stop mode entry is long to indicate a stop exception. The BDC indicates a stop mode occurrence by setting the BDCCSR bit STOP. If the host attempts further communication before the ACK pulse generation then the OVRUN bit is set.

5.1.3.3.2 Wait Mode

The device enters wait mode when the CPU starts to execute the WAI instruction. The second part of the WAI instruction (return from wait mode) can only be performed when an interrupt occurs. Thus on entering wait mode the CPU is in the middle of the WAI instruction and cannot permit access to CPU internal resources, nor allow entry to active BDM. Thus only commands classified as Non-Intrusive or Always-Available are possible in wait mode.

On entering wait mode, the WAIT flag in BDCCSR is set. If the ACK handshake protocol is enabled then the first ACK generated after WAIT has been set is a long-ACK pulse. Thus the host can recognize a wait mode occurrence. The WAIT flag remains set and cannot be cleared whilst the device remains in wait mode. After the device leaves wait mode the WAIT flag can be cleared by writing a “1” to it.

A BACKGROUND command issued whilst in wait mode sets the NORESP bit and the BDM active request remains pending internally until the CPU leaves wait mode due to an interrupt. The device then enters BDM with the PC pointing to the address of the first instruction of the ISR.

With ACK disabled, further Non-Intrusive or Always-Available commands are possible, in this pending state, but attempted Active-Background commands set NORESP and ILLCMD because the BDC is not in active BDM state.

With ACK enabled, if the host attempts further communication before the ACK pulse generation then the OVRUN bit is set.

Read: Anytime.

Write: Never

DBGEFR contains flag bits each mapped to events whilst armed. Should an event occur, then the corresponding flag is set. With the exception of TRIGF, the bits can only be set when the ARM bit is set. The TRIGF bit is set if a TRIG event occurs when ARM is already set, or if the TRIG event occurs simultaneous to setting the ARM bit.All other flags can only be cleared by arming the DBG module. Thus the contents are retained after a debug session for evaluation purposes.

A set flag does not inhibit the setting of other flags.

Table 7-13. DBGEFR Field Descriptions

Field	Description
6 TRIGF	**TRIG Flag** — Indicates the occurrence of a TRIG event during the debug session. 0 No TRIG event 1 TRIG event
4 EEVF	**External Event Flag** — Indicates the occurrence of an external event during the debug session. 0 No external event 1 External event
3–0 ME[3:0]	**Match Event[3:0]**— Indicates a comparator match event on the corresponding comparator channel**.**

7.3.2.7 Debug Status Register (DBGSR)

Address: 0x010B

	7	6	5	4	3	2	1	0
R	0	0	0	0	0	SSF2	SSF1	SSF0
W								
Reset	—	0	0	0	0	0	0	0
POR	0	0	0	0	0	0	0	0

= Unimplemented or Reserved

Figure 7-10. Debug Status Register (DBGSR)

Read: Anytime.

Write: Never.

Table 7-14. DBGSR Field Descriptions

Field	Description
2–0 SSF[2:0]	**State Sequencer Flag Bits** — The SSF bits indicate the current State Sequencer state. During a debug session on each transition to a new state these bits are updated. If the debug session is ended by software clearing the ARM bit, then these bits retain their value to reflect the last state of the state sequencer before disarming. If a debug session is ended by an internal event, then the state sequencer returns to State0 and these bits are cleared to indicate that State0 was entered during the session. On arming the module the state sequencer enters State1 and these bits are forced to SSF[2:0] = 001. See Table 7-15**.**

9.3.2.15 S12CPMU_UHV COP Timer Arm/Reset Register (CPMUARMCOP)

This register is used to restart the COP time-out period.

Module Base + 0x000F

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

Figure 9-18. S12CPMU_UHV CPMUARMCOP Register

Read: Always reads $00

Write: Anytime

When the COP is disabled (CR[2:0] = “000”) writing to this register has no effect.

When the COP is enabled by setting CR[2:0] nonzero, the following applies:

Writing any value other than $55 or $AA causes a COP reset. To restart the COP time-out period write $55 followed by a write of $AA. These writes do not need to occur back-to-back, but the sequence ($55, $AA) must be completed prior to COP end of time-out period to avoid a COP reset. Sequences of $55 writes are allowed. When the WCOP bit is set, $55 and $AA writes must be done in the last 25% of the selected time-out period; writing any value in the first 75% of the selected period will cause a COP reset.

9.3.2.16 High Temperature Control Register (CPMUHTCTL)

The CPMUHTCTL register configures the temperature sense features.

Module Base + 0x0010

	7	6	5	4	3	2	1	0
R	0	0	VSEL	0	HTE	HTDS	HTIE	HTIF
W								
Reset	0	0	0	0	0	0	0	0

= Unimplemented or Reserved

Figure 9-19. High Temperature Control Register (CPMUHTCTL)

Read: Anytime

Write: VSEL, HTE, HTIE and HTIF are write anytime, HTDS is read only

12.3.2.2 PGA Control Register (PGACNTL)

Module Base + 0x0001 Access: User read/write[1]

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

Figure 12-4. PGA Control Register (PGACNTL)

1 Read: Anytime
Write: Anytime

Table 12-4. PGACNTL Field Description

Field	Description
5:4 PGAREFSEL[1:0]	**PGA reference voltage selection** — If PGAEN=1 these register bits select the source for the reference voltage (PGAREF, minus input of both amplifier stages). 00 Internally generated V_{DDA} / 2 is selected as reference voltage (PGAREF) 01 Reserved 10 External PGA_REF0 input is selected as reference voltage (PGAREF). 11 External PGA_REF1 input is selected as reference voltage (PGAREF).
1:0 PGAINSEL[1:0]	**PGA input voltages selection** — This register bit defines the source for the plus input voltage of the amplifier. 00 no input voltage selected (PGAIN). 01 input voltage selection controlled by external modules, please see SoC level connection for more details. If the external control signals enables both inputs, then PGA_IN0 is selected. 10 PGA_IN0 is selected as input voltage (PGAIN). 11 PGA_IN1 is selected as input voltage (PGAIN).

12.3.2.3 PGA Gain Register (PGAGAIN)

Module Base + 0x0002 Access: User read/write[1]

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

Figure 12-5. PGA Gain Register (PGAGAIN)

1 Read: Anytime
Write: Anytime

Table 12-5. PGAGAIN Field Description

Field	Description
3:0 PGAGAIN[3:0]	**PGA1 gain**— These register bits select the gain A_{PGA} (amplification factor) for the PGA stage, see Table 12-6., “Amplifier Gain

If the WUPE bit in CANCTL0 is not asserted, the MSCAN will mask any activity it detects on CAN. RXCAN is therefore held internally in a recessive state. This locks the MSCAN in sleep mode. WUPE must be set before entering sleep mode to take effect.

The MSCAN is able to leave sleep mode (wake up) only when:

- CAN bus activity occurs and WUPE = 1

  or
- the CPU clears the SLPRQ bit

NOTE

The CPU cannot clear the SLPRQ bit before sleep mode (SLPRQ = 1 and SLPAK = 1) is active.

After wake-up, the MSCAN waits for 11 consecutive recessive bits to synchronize to the CAN bus. As a consequence, if the MSCAN is woken-up by a CAN frame, this frame is not received.

The receive message buffers (RxFG and RxBG) contain messages if they were received before sleep mode was entered. All pending actions will be executed upon wake-up; copying of RxBG into RxFG, message aborts and message transmissions. If the MSCAN remains in bus-off state after sleep mode was exited, it continues counting the 128 occurrences of 11 consecutive recessive bits.

13.4.5.6 MSCAN Power Down Mode

The MSCAN is in power down mode (Table 13-37) when

- CPU is in stop mode

  or
- CPU is in wait mode and the CSWAI bit is set

When entering the power down mode, the MSCAN immediately stops all ongoing transmissions and receptions, potentially causing CAN protocol violations. To protect the CAN bus system from fatal consequences of violations to the above rule, the MSCAN immediately drives TXCAN into a recessive state.

NOTE

The user is responsible for ensuring that the MSCAN is not active when power down mode is entered. The recommended procedure is to bring the MSCAN into Sleep mode before the STOP or WAI instruction (if CSWAI is set) is executed. Otherwise, the abort of an ongoing message can cause an error condition and impact other CAN bus devices.

In power down mode, all clocks are stopped and no registers can be accessed. If the MSCAN was not in sleep mode before power down mode became active, the module performs an internal recovery cycle after powering up. This causes some fixed delay before the module enters normal mode again.

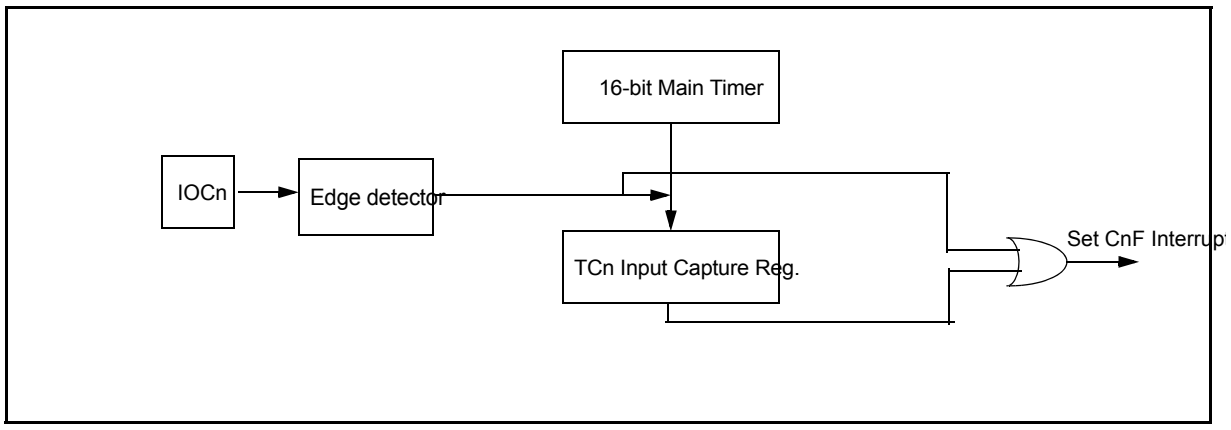

Figure 15-2. Interrupt Flag Setting

15.2 External Signal Description

The TIM16B6CV3 module has a selected number of external pins. Refer to device specification for exact number.

15.2.1 IOC5 - IOC0 — Input Capture and Output Compare Channel 5-0

Those pins serve as input capture or output compare for TIM16B6CV3 channel .

NOTE

For the description of interrupts see Section 15.6, “Interrupts”.

15.3 Memory Map and Register Definition

This section provides a detailed description of all memory and registers.

15.3.1 Module Memory Map

The memory map for the TIM16B6CV3 module is given below in Figure 15-3. The address listed for each register is the address offset. The total address for each register is the sum of the base address for the TIM16B6CV3 module and the address offset for each register.

15.3.2 Register Descriptions

This section consists of register descriptions in address order. Each description includes a standard register diagram with an associated figure number. Details of register bit and field function follow the register diagrams, in bit order.

17.4.2 PWM Channel Timers

The main part of the PWM module are the actual timers. Each of the timer channels has a counter, a period register and a duty register (each are 8-bit). The waveform output period is controlled by a match between the period register and the value in the counter. The duty is controlled by a match between the duty register and the counter value and causes the state of the output to change during the period. The starting polarity of the output is also selectable on a per channel basis. Shown below in Figure 17-16 is the block diagram for the PWM timer.

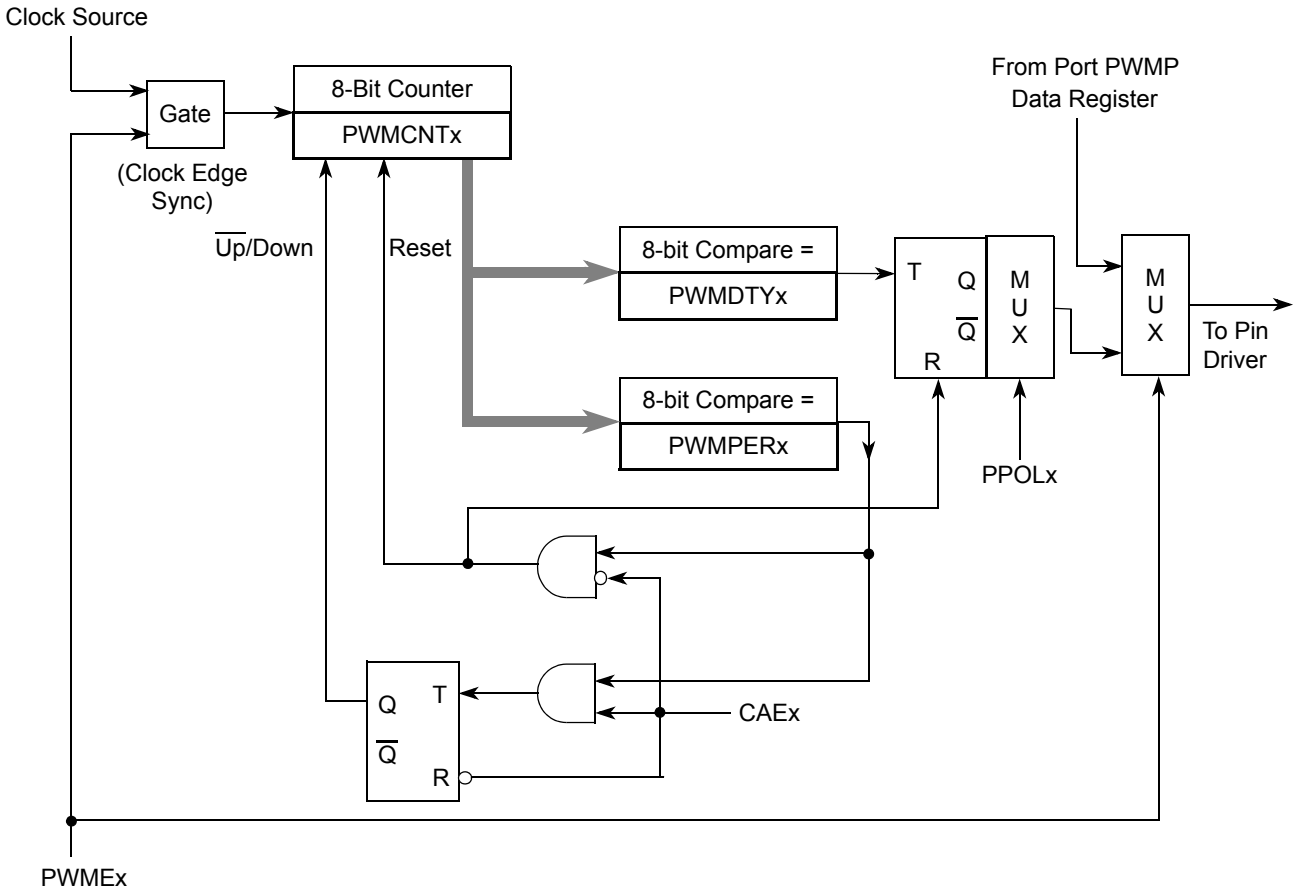

Figure 17-16. PWM Timer Channel Block Diagram

17.4.2.1 PWM Enable

Each PWM channel has an enable bit (PWMEx) to start its waveform output. When any of the PWMEx bits are set (PWMEx = 1), the associated PWM output signal is enabled immediately. However, the actual PWM waveform is not available on the associated PWM output until its clock source begins its next cycle due to the synchronization of PWMEx and the clock source. An exception to this is when channels are concatenated. Refer to Section 17.4.2.7, “PWM 16-Bit Functions” for more detail.

NOTE

The first PWM cycle after enabling the channel can be irregular.

- For channels 0, 1, 4, and 5 the clock choices are clock A.
- For channels 2, 3, 6, and 7 the clock choices are clock B.

17.6 Interrupts

The PWM module has no interrupt.

18.4.5.5 LIN Transmit Collision Detection

This module allows to check for collisions on the LIN bus.

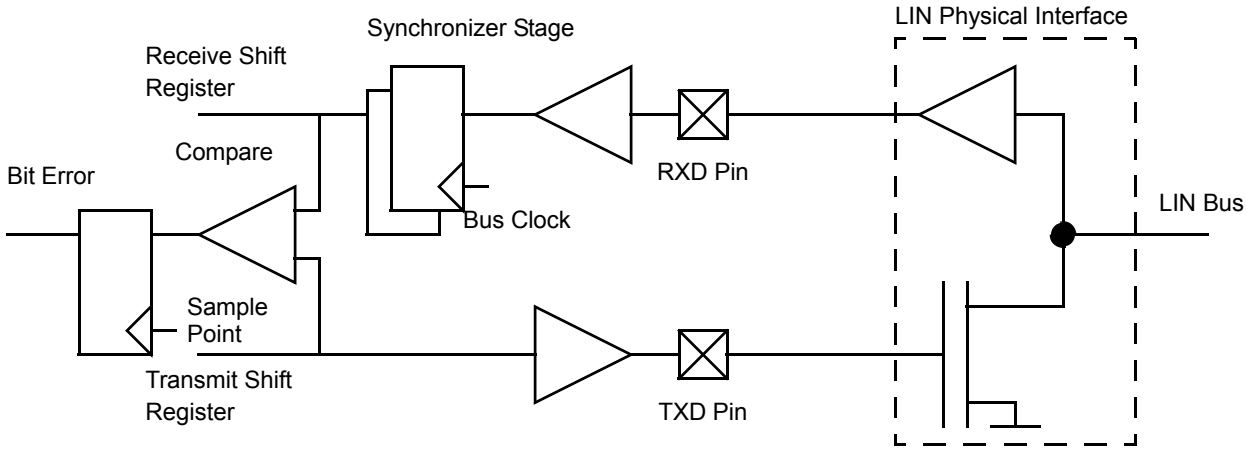

Figure 18-18. Collision Detect Principle

If the bit error circuit is enabled (BERRM[1:0] = 0:1 or = 1:0]), the error detect circuit will compare the transmitted and the received data stream at a point in time and flag any mismatch. The timing checks run when transmitter is active (not idle). As soon as a mismatch between the transmitted data and the received data is detected the following happens:

- The next bit transmitted will have a high level (TXPOL = 0) or low level (TXPOL = 1)
- The transmission is aborted and the byte in transmit buffer is discarded.
- the transmit data register empty and the transmission complete flag will be set
- The bit error interrupt flag, BERRIF, will be set.
- No further transmissions will take place until the BERRIF is cleared.

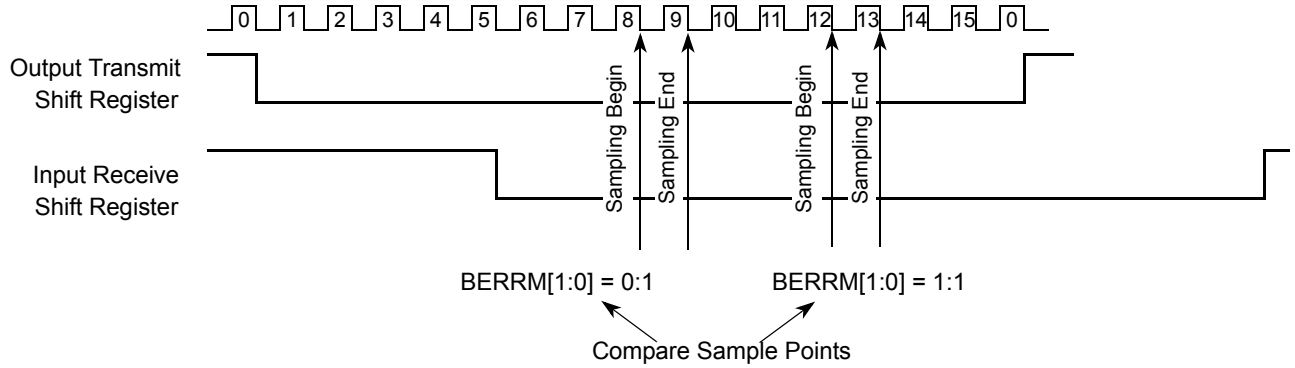

Figure 18-19. Timing Diagram Bit Error Detection

If the bit error detect feature is disabled, the bit error interrupt flag is cleared.

NOTE

The RXPOL and TXPOL bit should be set the same when transmission collision detect feature is enabled, otherwise the bit error interrupt flag may be set incorrectly.

Chapter 20
Inter-Integrated Circuit (IICV3)

Table 20-1. Revision History

Revision Number	Revision Date	Sections Affected	Description of Changes
V01.03	28 Jul 2006	20.7.1.7/20-592	- Update flow-chart of interrupt routine for 10-bit address
V01.04	17 Nov 2006	20.3.1.2/20-572	- Revise Table1-5
V01.05	14 Aug 2007	20.3.1.1/20-572	- Backward compatible for IBAD bit name

20.1 Introduction

The inter-IC bus (IIC) is a two-wire, bidirectional serial bus that provides a simple, efficient method of data exchange between devices. Being a two-wire device, the IIC bus minimizes the need for large numbers of connections between devices, and eliminates the need for an address decoder.

This bus is suitable for applications requiring occasional communications over a short distance between a number of devices. It also provides flexibility, allowing additional devices to be connected to the bus for further expansion and system development.

The interface is designed to operate up to 100 kbps with maximum bus loading and timing. The device is capable of operating at higher baud rates, up to a maximum of clock/20, with reduced bus loading. The maximum communication length and the number of devices that can be connected are limited by a maximum bus capacitance of 400 pF.

20.1.1 Features

The IIC module has the following key features:

- Compatible with I2C bus standard
- Multi-master operation
- Software programmable for one of 256 different serial clock frequencies
- Software selectable acknowledge bit
- Interrupt driven byte-by-byte data transfer
- Arbitration lost interrupt with automatic mode switching from master to slave
- Calling address identification interrupt
- Start and stop signal generation/detection
- Repeated start signal generation

resistor of 1 kΩ must be placed in parallel between VLINSUP and the LIN Bus pin, with a diode between VLINSUP and the resistor. The fall time from recessive to dominant and the rise time from dominant to recessive is selectable and controlled to guarantee communication quality and reduce EMC emissions. The symmetry between both slopes is guaranteed.

21.1.2 Modes of Operation

The LIN Physical Layer can operate in the following four modes:

1. Shutdown Mode

   The LIN Physical Layer is fully disabled. No wake-up functionality is available. The internal pullup resistor is replaced by a high ohmic one (330 kΩ) to maintain the LIN Bus pin in the recessive state. All registers are accessible.
2. Normal Mode

   The full functionality is available. Both receiver and transmitter are enabled.
3. Receive Only Mode

   The transmitter is disabled and the receiver is running in full performance mode.
4. Standby Mode

   The transmitter of the LIN Physical Layer is disabled. If the wake-up feature is enabled, the internal pullup resistor can be selected (330 kΩ or 34 kΩ). The receiver enters a low power mode and optionally it can pass wake-up events to the Serial Communication Interface (SCI). If the wake-up feature is enabled and if the LIN Bus pin is driven with a dominant level longer than t_{WUFR} followed by a rising edge, the LIN Physical Layer sends a wake-up pulse to the SCI, which requests a wake-up interrupt. (This feature is only available if the LIN Physical Layer is routed to the SCI).

21.1.3 Block Diagram

Figure 21-1 shows the block diagram of the LIN Physical Layer. The module consists of a receiver with wake-up control, a transmitter with slope and timeout control, a current sensor with overcurrent protection as well as a registers control block.

Table 22-3. Flash Configuration Field

Global Address	Size (Bytes)	Description
0xFF_FE00-0xFF_FE07	8	Backdoor Comparison Key Refer to Section 22.4.7.11, “Verify Backdoor Access Key Command,” and Section 22.5.1, “Unsecuring the MCU using Backdoor Key Access”
0xFF_FE08-0xFF_FE09[1]	2	Protection Override Comparison Key. Refer to Section 22.4.7.17, “Protection Override Command”
0xFF_FE0A-0xFF_FE0B[1]	2	Reserved
0xFF_FE0C[1]	1	P-Flash Protection byte. Refer to Section 22.3.2.9, “P-Flash Protection Register (FPROT)”
0xFF_FE0D[1]	1	EEPROM Protection byte. Refer to Section 22.3.2.10, “EEPROM Protection Register (DFPROT)”
0xFF_FE0E[1]	1	Flash Nonvolatile byte Refer to Section 22.3.2.11, “Flash Option Register (FOPT)”
0xFF_FE0F[1]	1	Flash Security byte Refer to Section 22.3.2.2, “Flash Security Register (FSEC)”

1 0xFF_FE08-0xFF_FE0F form a Flash phrase and must be programmed in a single command write sequence. Each byte in the 0xFF_FE0A - 0xFF_FE0B reserved field should be programmed to 0xFF.

Offset Module Base + 0x0011

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

Figure 22-23. Flash Common Command Object 2 Low Register (FCCOB2LO)

Offset Module Base + 0x0012

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

Figure 22-24. Flash Common Command Object 3 High Register (FCCOB3HI)

Offset Module Base + 0x0013

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

Figure 22-25. Flash Common Command Object 3 Low Register (FCCOB3LO)

Offset Module Base + 0x0014

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

Figure 22-26. Flash Common Command Object 4 High Register (FCCOB4HI)

Offset Module Base + 0x0015

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

Figure 22-27. Flash Common Command Object 4 Low Register (FCCOB4LO)

22.4.7.3 Erase Verify P-Flash Section Command

The Erase Verify P-Flash Section command will verify that a section of code in the P-Flash memory is erased. The Erase Verify P-Flash Section command defines the starting point of the code to be verified and the number of phrases.

Table 22-36. Erase Verify P-Flash Section Command FCCOB Requirements

Register	FCCOB Parameters	
FCCOB0	0x03	Global address [23:16] of a P-Flash block
FCCOB1	Global address [15:0] of the first phrase to be verified	
FCCOB2	Number of phrases to be verified	

Upon clearing CCIF to launch the Erase Verify P-Flash Section command, the Memory Controller will verify the selected section of Flash memory is erased. The CCIF flag will set after the Erase Verify P-Flash Section operation has completed. If the section is not erased, it means blank check failed, both MGSTAT bits will be set.

Table 22-37. Erase Verify P-Flash Section Command Error Handling

Register	Error Bit	Error Condition
FSTAT	ACCERR	Set if CCOBIX[2:0] != 010 at command launch
		Set if command not available in current mode (see Table 22-28)
		Set if an invalid global address [23:0] is supplied see Table 22-2)
		Set if a misaligned phrase address is supplied (global address [2:0] != 000)
		Set if the requested section crosses a the P-Flash address boundary
	FPVIOL	None
	MGSTAT1	Set if any errors have been encountered during the read or if blank check failed.
	MGSTAT0	Set if any non-correctable errors have been encountered during the read or if blank check failed.

22.4.7.4 Read Once Command

The Read Once command provides read access to a reserved 64 byte field (8 phrases) located in the nonvolatile information register of P-Flash. The Read Once field is programmed using the Program Once command described in Section 22.4.7.6. The Read Once command must not be executed from the Flash block containing the Program Once reserved field to avoid code runaway.

Table 22-38. Read Once Command FCCOB Requirements

Register	FCCOB Parameters	
FCCOB0	0x04	Not Required
FCCOB1	Read Once phrase index (0x0000 - 0x0007)	
FCCOB2	Read Once word 0 value	

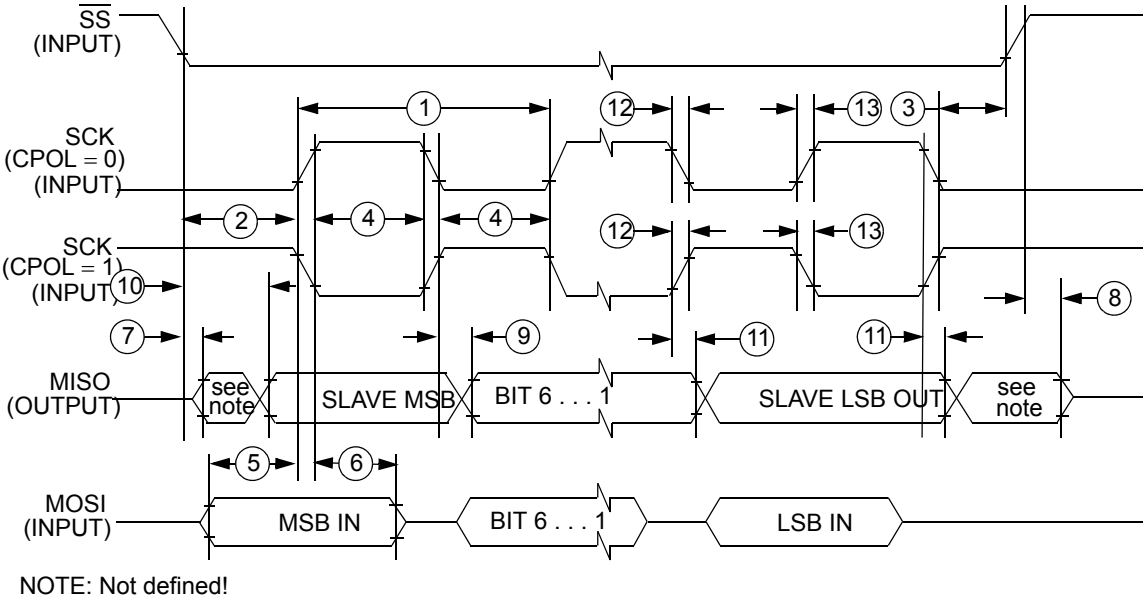

Figure K-3. SPI Slave Timing (CPHA=0)

In Figure K-4 the timing diagram for slave mode with transmission format CPHA=1 is depicted.

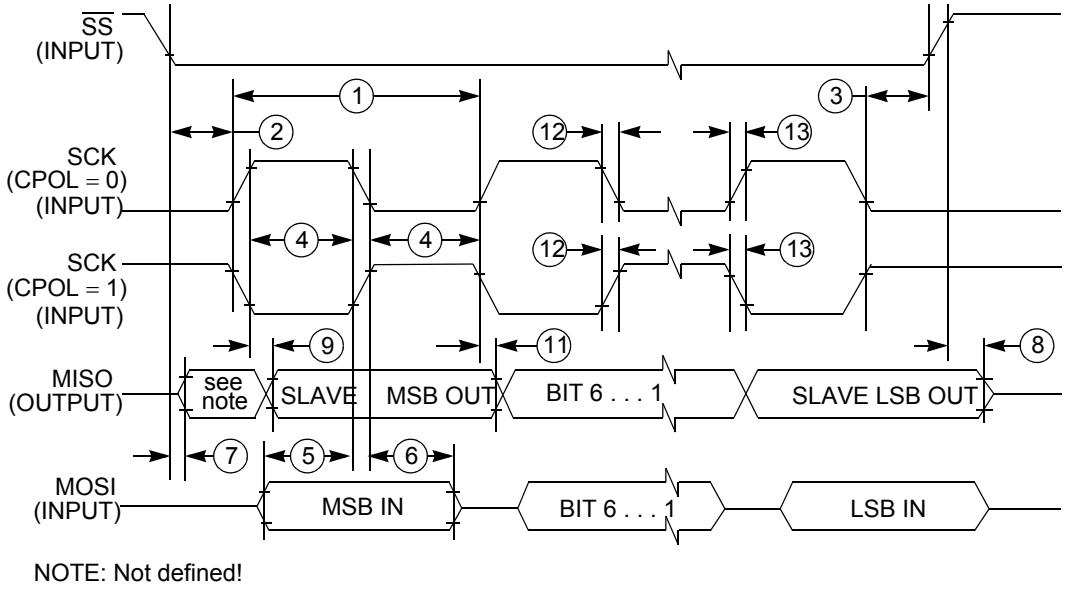

Figure K-4. SPI Slave Timing (CPHA=1)

O.2 0x0010–0x001F S12ZINT

Address	Name		Bit 7	6	5	4	3	2	1	Bit 0
0x001A	INT_CFDATA2	R	0	0	0	0	0	PRIOLVL[2:0]		
		W								
0x001B	INT_CFDATA3	R	0	0	0	0	0	PRIOLVL[2:0]		
		W								
0x001C	INT_CFDATA4	R	0	0	0	0	0	PRIOLVL[2:0]		
		W								
0x001D	INT_CFDATA5	R	0	0	0	0	0	PRIOLVL[2:0]		
		W								
0x001E	INT_CFDATA6	R	0	0	0	0	0	PRIOLVL[2:0]		
		W								
0x001F	INT_CFDATA7	R	0	0	0	0	0	PRIOLVL[2:0]		
		W								

O.6 0x0380-0x039F FTMRZ (continued)

Address	Name		7	6	5	4	3	2	1	0
0x0394	FCCOB4HI	R	CCOB15	CCOB14	CCOB13	CCOB12	CCOB11	CCOB10	CCOB9	CCOB8
		W								
0x0395	FCCOB4LO	R	CCOB7	CCOB6	CCOB5	CCOB4	CCOB3	CCOB2	CCOB1	CCOB0
		W								
0x0396	FCCOB5HI	R	CCOB15	CCOB14	CCOB13	CCOB12	CCOB11	CCOB10	CCOB9	CCOB8
		W								
0x0397	FCCOB5LO	R	CCOB7	CCOB6	CCOB5	CCOB4	CCOB3	CCOB2	CCOB1	CCOB0
		W								

O.7 0x03C0-0x03CF SRAM_ECC_32D7P

Address	Name		Bit 7	6	5	4	3	2	1	Bit 0
0x03C0	ECCSTAT	R	0	0	0	0	0	0	0	RDY
		W								
0x03C1	ECCIE	R	0	0	0	0	0	0	0	SBEEIE
		W								
0x03C2	ECCIF	R	0	0	0	0	0	0	0	SBEEIF
		W								
0x03C3 - 0x03C6	Reserved	R	0	0	0	0	0	0	0	0
		W								
0x03C7	ECCDPTRH	R	DPTR[23:16]							
		W								
0x03C8	ECCDPTRM	R	DPTR[15:8]							
		W								
0x03C9	ECCDPTRL	R	DPTR[7:1]							0
		W								
0x03CA - 0x03CB	Reserved	R	0	0	0	0	0	0	0	0
		W								
0x03CC	ECCDDH	R	DDATA[15:8]							
		W								
0x03CD	ECCDDL	R	DDATA[7:0]							
		W								
0x03CE	ECCDE	R	0	0	DECC[5:0]					
		W								
0x03CF	ECCDCMD	R	ECCDRR	0	0	0	0	0	ECCDW	ECCDR
		W								