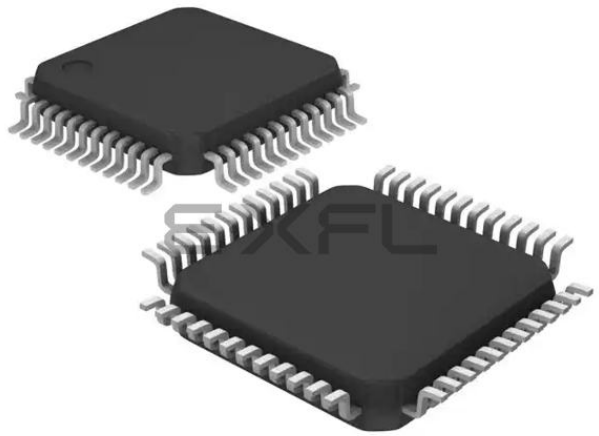


Welcome to E-XFL.COM

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	34
Program Memory Size	32KB (32K 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 10x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 105°C (TA)
Mounting Type	Surface Mount
Package / Case	48-LQFP
Supplier Device Package	48-LQFP (7x7)
Purchase URL	https://www.e-xfl.com/product-detail/nxp-semiconductors/s9s12zvl32f0vlf

Table 1-3. Module Register Address Ranges

Address	Module	Size (Bytes)
0x0700–0x0707	SCI0	8
0x0708–0x070F	Reserved	8
0x0710–0x0717	SCI1	8
0x0718–0x077F	Reserved	104
0x0780–0x0787	SPI0	8
0x0788–0x07BF	Reserved	56
0x07C0–0x07C7	IIC0	8
0x07C8–0x07FF	Reserved	56
0x0800–0x083F	CAN0	64
0x0840–0x097F	Reserved	320
0x0980–0x0987	LINPHY0	8
0x0988–0x0B3F	Reserved	440
0x0B40–0x0B47	PGA	8
0x0B48–0x0FFF	Reserved	1208

[1] Reading from the first 16 locations in this range returns undefined data

NOTE

Reserved register space shown above is not allocated to any module. This register space is reserved for future use. Writing to these locations has no effect. Read access to these locations returns zero.

Table 1-12. Interrupt Vector Locations (Sheet 3 of 4)

Vector Address[1]	Interrupt Source	CCR Mask	Local Enable	Wake up from STOP	Wake up from WAIT
Vector base + 0x16C to Vector base + 0x168	Reserved				
Vector base + 0x164	FLASH error	I bit	FERCNFG (SFDIE)	No	Yes
Vector base + 0x160	FLASH command	I bit	FCNFG (CCIE)	No	Yes
Vector base + 0x15C	CAN0 wake-up	I bit	CANRIER(WUPIE)	Yes	Yes
Vector base + 0x158	CAN0 error	I bit	CANRIER(CSCIE, OVRIE)	No	Yes
Vector base + 0x154	CAN0 receive	I bit	CANRIER(RXFIE)	No	Yes
Vector base + 0x150	CAN0 transmit	I bit	CANRIER(TXEIE[2:0])	No	Yes
Vector base + 0x14C to Vector base + 0x148	Reserved				
Vector base + 0x144	LINPHY over-current interrupt	I bit	LPIE (LPDTIE, LPOCIE)	No	Yes
Vector base + 0x140	BATS supply voltage monitor interrupt	I bit	BATIE (BVHIE,BVLIE)	No	Yes
Vector base + 0x13C to Vector base + 0x128	Reserved				
Vector base + 0x124	Port S interrupt	I bit	PIES(PIES[3:0])	Yes	Yes
Vector base + 0x120 to Vector base + 0x110	Reserved				
Vector base + 0x10C	Port P interrupt	I bit	PIEP(PIEP[7:0])	Yes	Yes
Vector base + 0x108	Port P over-current interrupt	I bit	OCIEP (OCIEP7,OCIEP5,OCIEP3,OCIEP1,)	No	Yes
Vector base + 0x104	Low-voltage interrupt (LVI)	I bit	CPMUCTRL (LVIE)	No	Yes
Vector base + 0x100	Autonomous periodical interrupt (API)	I bit	CPMUAPICTRL (APIE)	Yes	Yes
Vector base + 0x0FC	High temperature interrupt	I bit	CPMUHTCTL(HTIE)	No	Yes
Vector base + 0x0F8	Reserved				
Vector base + 0x0F4	Port AD interrupt	I bit	PIEADH(PIEADH[1:0]) PIEADL(PIEADL[7:0])	Yes	Yes
Vector base + 0x0F0 to Vector base + 0x0C4	Reserved				
Vector base + 0x0C0	Port L interrupt	I bit	PIEL(PIEL[0])	Yes	Yes
Vector base + 0x0BC to Vector base + 0x0B0	Reserved				

Global Address	Register Name		Bit 7	6	5	4	3	2	1	Bit 0
0x033F	Reserved	R	0	0	0	0	0	0	0	0
		W								
0x0340–0x037F	Reserved	R	0	0	0	0	0	0	0	0
		W								

2.3.2 PIM Registers 0x0200-0x020F

This section details the specific purposes of register implemented in address range 0x0200-0x020F. These registers serve for specific PIM related functions not part of the generic port registers.

- If not stated differently, writing to reserved bits has no effect and read returns zero.
- All register read accesses are synchronous to internal clocks.
- Register bits can be written at any time if not stated differently.

2.3.2.1 Module Routing Register 0 (MODRR0)

Address 0x0200 Access: User read/write[1]

	7	6	5	4	3	2	1	0
R	0	0	CAN0RR	IIC0RR	SCI1RR	S0L0RR2-0		
W								
	—	—	TXCAN0 RXCAN0	SDA0 SCL0	TXD1 RXD1	SCI0-LINPHY0 (see Figure 2-2)		
Reset	0	0	0	0	0	0	0	0

Figure 2-1. Module Routing Register 0 (MODRR0)

[1] Read: Anytime
Write: Once in normal, anytime in special mode

Table 2-2. MODRR0 Routing Register Field Descriptions

Field	Description
5 CAN0RR	**Module Routing Register** — CAN0 routing 1 RXCAN0 on PJ0; TXCAN0 on PJ1 0 RXCAN0 on PS0; TXCAN0 on PS2
4 IIC0RR	**Module Routing Register** — IIC0 routing 1 SDA0 on PT0; SCL0 on PT1 0 SDA0 on PJ0; SCL0 on PJ1

Table 2-3. Preferred Interface Configurations

S0L0RR[2:0]	Description
000	Default setting: SCI0 connects to LINPHY0, interface internal only
001	Direct control setting: LP0DR[LPDR1] register bit controls LPTXD0, interface internal only
100	Probe setting: SCI0 connects to LINPHY0, interface accessible on 2 external pins
110	Conformance test setting: Interface opened and all 4 signals routed externally

NOTE

For standalone usage of SCI0 on external pins set S0L0RR[2:0]=0b110 and disable LINPHY0 (LPCR[LPE]=0). This releases PT0 and PT1 to other associated functions and maintains TXD0 and RXD0 signals on PS1 and PS0, respectively, if no other function with higher priority takes precedence.

2.3.2.2 Module Routing Register 1 (MODRR1)

Address 0x0201 Access: User read/write[1]

	7	6	5	4	3	2	1	0
R	PWM7RR	PWM6RR	PWM5RR	PWM4RR	0	PWM2RR	0	PWM0RR
W								
	PWM opt. 7	PWM opt. 6	PWM opt. 5	PWM opt. 4	—	PWM opt. 2	—	PWM opt. 0
Reset	0	0	0	0	0	0	0	0

Figure 2-3. Module Routing Register 1 (MODRR1)

[1] Read: Anytime
Write: Once in normal, anytime in special mode

Table 2-4. MODRR1 Routing Register Field Descriptions

Field	Description
7 PWM7RR	**Module Routing Register** — PWM option 7 routing 1 PWM option 7 to PJ1 0 PWM option 7 to PP7
6 PWM6RR	**Module Routing Register** — PWM option 6 routing 1 PWM option 6 to PS1 0 PWM option 6 to PP6
5 PWM5RR	**Module Routing Register** — PWM option 5 routing 1 PWM option 5 to PJ0 0 PWM option 5 to PP5

Field	Description
7-4 (MMCECL) ACC[3:0]	**Access Type Field** — The ACC[3:0] bits capture the type of memory access, which caused the access violation. The access type is captured in form of a 4 bit value which is assigned as follows: 0:none (no error condition detected) 1:opcode fetch 2:vector fetch 3:data load 4:data store 5-15: reserved
3-0 (MMCECL) ERR[3:0]	**Error Type Field** — The EC[3:0] bits capture the type of the access violation. The type is captured in form of a 4 bit value which is assigned as follows: 0:none (no error condition detected) 1:access to an illegal address 2:uncorrectable ECC error 3-15:reserved

The MMCEC register captures debug information about access violations. It is set to a non-zero value if a S12ZCPU access violation or an uncorrectable ECC error has occurred. At the same time this register is set to a non-zero value, access information is captured in the MMCPCn and MMCCCRn registers. The MMCECn, the MMCPCn and the MMCCCRn registers are not updated if the MMCECn registers contain a non-zero value. The MMCECn registers are cleared by writing the value 0xFFFF.

4.3.2.3 Captured S12ZCPU Condition Code Register (MMCCCRH, MMCCCRL)

Address: 0x0082 (MMCCCRH)

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

Address: 0x0083 (MMCCCRL)

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

Figure 4-6. Captured S12ZCPU Condition Code Register (MMCCCRH, MMCCCRL)

Read: Anytime

Write: Never

4.4.2 Illegal Accesses

The S12ZDBG module monitors all memory traffic for illegal accesses. See Table 4-8 for a complete list of all illegal accesses.

Table 4-8. Illegal memory accesses

		S12ZCPU	S12ZBDC	ADC
Register space	Read access	ok	ok	illegal access
	Write access	ok	ok	illegal access
	Code execution	illegal access		
RAM	Read access	ok	ok	ok
	Write access	ok	ok	ok
	Code execution	ok		
EEPROM	Read access	ok[1]	ok[1]	ok[1]
	Write access	illegal access	illegal access	illegal access
	Code execution	ok[1]		
Reserved Space	Read access	ok	ok	illegal access
	Write access	only permitted in SS mode	ok	illegal access
	Code execution	illegal access		
Reserved Read-only Space	Read access	ok	ok	illegal access
	Write access	illegal access	illegal access	illegal access
	Code execution	illegal access		
NVM IFR	Read access	ok[1]	ok[1]	illegal access
	Write access	illegal access	illegal access	illegal access
	Code execution	illegal access		
Program NVM	Read access	ok[1]	ok[1]	ok[1]
	Write access	illegal access	illegal access	illegal access
	Code execution	ok[1]		
Unmapped Space	Read access	illegal access	illegal access	illegal access
	Write access	illegal access	illegal access	illegal access
	Code execution	illegal access		

1 Unsupported NVM accesses during NVM command execution (“collisions”), are treated as illegal accesses.

Illegal accesses are reported in several ways:

- All illegal accesses performed by the S12ZCPU trigger machine exceptions.
- All illegal accesses performed through the S12ZBDC interface, are captured in the ILLACC bit of the BDCCSRL register.

NOTE

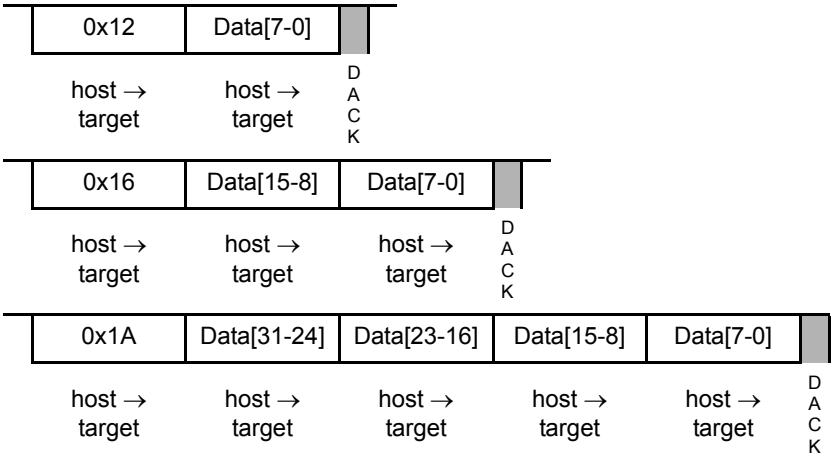
DUMP_MEM{_WS} is a valid command only when preceded by SYNC, NOP, READ_MEM{_WS}, or another DUMP_MEM{_WS} command. Otherwise, an illegal command response is returned, setting the ILLCMD bit. NOP can be used for inter-command padding without corrupting the address pointer.

The size field (sz) is examined each time a DUMP_MEM{_WS} command is processed, allowing the operand size to be dynamically altered. The examples show the DUMP_MEM.B{_WS}, DUMP_MEM.W{_WS} and DUMP_MEM.L{_WS} commands.

5.4.4.6 FILL_MEM.sz, FILL_MEM.sz_WS

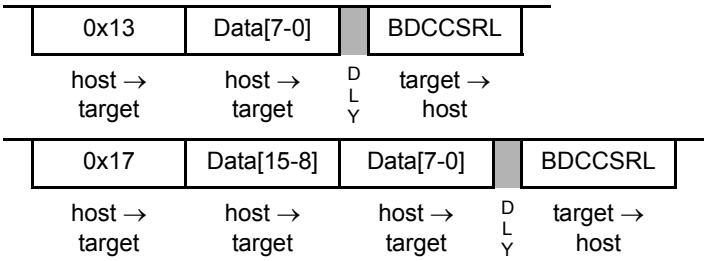
FILL_MEM.sz

Write memory specified by debug address register, then increment address **Non-intrusive**



FILL_MEM.sz_WS

Write memory specified by debug address register with status, then increment address **Non-intrusive**



7.3.2.14 Debug Comparator D Control Register (DBGDCTL)

Address: 0x0140

	7	6	5	4	3	2	1	0
R	0	0	INST	0	RW	RWE	reserved	COMPE
W								
Reset	0	0	0	0	0	0	0	0

= Unimplemented or Reserved

Figure 7-17. Debug Comparator D Control Register

Read: Anytime.

Write: If DBG not armed.

Table 7-24. DBGDCTL Field Descriptions

Field[1]	Description
5 INST	**Instruction Select** — This bit configures the comparator to compare PC or data access addresses. 0 Comparator compares addresses of data accesses 1 Comparator compares PC address
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 ignored if RWE is clear or INST is set. 0 Write cycle is matched 1 Read cycle is 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 INST is set. 0 Read/Write is not used in comparison 1 Read/Write is used in comparison
0 COMPE	**Enable Bit** — Determines if comparator is enabled 0 The comparator is not enabled 1 The comparator is enabled

1 If the CDCM field selects range mode comparisons, then DBGCCTL bits configure the comparison, DBGDCTL is ignored.

Table 7-25 shows the effect for RWE and RW on the comparison conditions. These bits are ignored if INST is set, because matches based on opcodes reaching the execution stage are data independent.

Table 7-25. 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 match
1	0	1	No match
1	1	0	No match
1	1	1	Read match

Upon detection of a status change (UPOSC) the OSCIF flag is set. Going into Full Stop Mode or disabling the oscillator can also cause a status change of UPOSC.

Any change in PLL configuration or any other event which causes the PLL lock status to be cleared leads to a loss of the oscillator status information as well (UPOSC=0).

Oscillator status change interrupts are locally enabled with the OSCIE bit.

NOTE

Loosing the oscillator status (UPOSC=0) affects the clock configuration of the system[1]. This needs to be dealt with in application software.

9.6.1.4 Low-Voltage Interrupt (LVI)

In FPM the input voltage VDDA is monitored. Whenever VDDA drops below level $V_{LVIA,}$ the status bit LVDS is set to 1. When VDDA rises above level V_{LVID} the status bit LVDS is cleared to 0. An interrupt, indicated by flag LVIF = 1, is triggered by any change of the status bit LVDS if interrupt enable bit LVIE = 1.

9.6.1.5 HTI - High Temperature Interrupt

In FPM the junction temperature T_J is monitored. Whenever T_J exceeds level T_{HTIA} the status bit HTDS is set to 1. Vice versa, HTDS is reset to 0 when T_J get below level T_{HTID}. An interrupt, indicated by flag HTIF = 1, is triggered by any change of the status bit HTDS, if interrupt enable bit HTIE = 1.

9.6.1.6 Autonomous Periodical Interrupt (API)

The API sub-block can generate periodical interrupts independent of the clock source of the MCU. To enable the timer, the bit APIFE needs to be set.

The API timer is either clocked by the Autonomous Clock (ACLK - trimmable internal RC oscillator) or the Bus Clock. Timer operation will freeze when MCU clock source is selected and Bus Clock is turned off. The clock source can be selected with bit APICLK. APICLK can only be written when APIFE is not set.

The APIR[15:0] bits determine the interrupt period. APIR[15:0] can only be written when APIFE is cleared. As soon as APIFE is set, the timer starts running for the period selected by APIR[15:0] bits. When the configured time has elapsed, the flag APIF is set. An interrupt, indicated by flag APIF = 1, is triggered if interrupt enable bit APIE = 1. The timer is re-started automatically again after it has set APIF.

The procedure to change APICLK or APIR[15:0] is first to clear APIFE, then write to APICLK or APIR[15:0], and afterwards set APIFE.

The API Trimming bits ACLKTR[5:0] must be set so the minimum period equals 0.2 ms if stable frequency is desired.

See Table 9-21 for the trimming effect of ACLKTR[5:0].

1. For details please refer to “<st-blue>9.4.6 System Clock Configurations”

– *When finished:*
This bit is cleared when the first conversion command of the sequence from top of active Sequence Command List is loaded

– *Mandatory Requirement:*
- In all ADC conversion flow control modes a Restart Event causes bit RSTA to be set. Bit SEQA is set simultaneously by ADC hardware if:
* ADC not idle (a conversion or conversion sequence is ongoing and current CSL not finished) and no Sequence Abort Event in progress (bit SEQA not already set or set simultaneously via internal interface or data bus)
* ADC idle but RVL done condition not reached
The RVL done condition is reached by one of the following:
* A “End Of List” command type has been executed
* A Sequence Abort Event is in progress or has been executed (bit SEQA already set or set simultaneously via internal interface or data bus)
The ADC executes the Sequence Abort Event followed by the Restart Event for the conditions described before or only a Restart Event.
- In ADC conversion flow control mode “Trigger Mode” a Restart Event causes bit TRIG being set automatically. Bit TRIG is set when no conversion or conversion sequence is ongoing (ADC idle) and the RVL done condition is reached by one of the following:
* A “End Of List” command type has been executed
* A Sequence Abort Event is in progress or has been executed
The ADC executes the Restart Event followed by the Trigger Event.
- In ADC conversion flow control mode “Trigger Mode” a Restart Event and a simultaneous Trigger Event via internal interface or data bus causes the TRIG_EIF bit being set and ADC cease operation.

- **Restart Event + CSL Exchange (Swap)**
Internal Interface Signals: Restart + LoadOK
Corresponding Bit Names: RSTA + LDOK

– *Function:*
Go to top of active CSL (clear index register for CSL) and switch to other offset register for address calculation if configured for double buffer mode (exchange the CSL list)
Requested by:
- Internal interface with the assertion of Interface Signal Restart the interface Signal LoadOK is evaluated and bit LDOK is set accordingly (bit LDOK set if Interface Signal LoadOK asserted when Interface Signal Restart asserts).
- Write Access via data bus to set control bit RSTA simultaneously with bit LDOK.

– *When finished:*
Bit LDOK can only be cleared if it was set as described before and both bits (LDOK, RSTA) are cleared when the first conversion command from top of active Sequence Command List is loaded

– *Mandatory Requirement:*
No ongoing conversion or conversion sequence
Details if using the internal interface:

NOTE

The CANTBSEL register is held in the reset state when the initialization mode is active (INITRQ = 1 and INITAK=1). This register is writable when not in initialization mode (INITRQ = 0 and INITAK = 0).

Table 13-17. CANTBSEL Register Field Descriptions

Field	Description
2-0 TX[2:0]	**Transmit Buffer Select** — The lowest numbered bit places the respective transmit buffer in the CANTXFG register space (e.g., TX1 = 1 and TX0 = 1 selects transmit buffer TX0; TX1 = 1 and TX0 = 0 selects transmit buffer TX1). Read and write accesses to the selected transmit buffer will be blocked, if the corresponding TXEx bit is cleared and the buffer is scheduled for transmission (see Section 13.3.2.7, “MSCAN Transmitter Flag Register (CANTFLG)”). 0 The associated message buffer is deselected 1 The associated message buffer is selected, if lowest numbered bit

The following gives a short programming example of the usage of the CANTBSEL register:

To get the next available transmit buffer, application software must read the CANTFLG register and write this value back into the CANTBSEL register. In this example Tx buffers TX1 and TX2 are available. The value read from CANTFLG is therefore 0b0000_0110. When writing this value back to CANTBSEL, the Tx buffer TX1 is selected in the CANTXFG because the lowest numbered bit set to 1 is at bit position 1. Reading back this value out of CANTBSEL results in 0b0000_0010, because only the lowest numbered bit position set to 1 is presented. This mechanism eases the application software’s selection of the next available Tx buffer.

- LDAA CANTFLG; value read is 0b0000_0110
- STAA CANTBSEL; value written is 0b0000_0110
- LDAA CANTBSEL; value read is 0b0000_0010

If all transmit message buffers are deselected, no accesses are allowed to the CANTXFG registers.

13.3.2.12 MSCAN Identifier Acceptance Control Register (CANIDAC)

The CANIDAC register is used for identifier acceptance control as described below.

Module Base + 0x000B — Access: User read/write[1]

	7	6	5	4	3	2	1	0
R	0	0	IDAM1	IDAM0	0	IDHIT2	IDHIT1	IDHIT0
W								
Reset:	0	0	0	0	0	0	0	0

= Unimplemented

Figure 13-15. MSCAN Identifier Acceptance Control Register (CANIDAC)

1 Read: Anytime
Write: Anytime in initialization mode (INITRQ = 1 and INITAK = 1), except bits IDHITx, which are read-only

Module Base + 0x000C — Access: User read/write[1]

	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

= Unimplemented

Figure 13-16. MSCAN Reserved Registers

1 Read: Always reads zero in normal system operation modes
Write: Unimplemented in normal system operation modes

NOTE

13.3.2.14 Writing to this register when in special system operating modes can alter the MSCAN functionality.**MSCAN Receive Error Counter (CANRXERR)**

This register reflects the status of the MSCAN receive error counter.

Module Base + 0x000E — Access: User read/write[1]

	7	6	5	4	3	2	1	0
R	RXERR7	RXERR6	RXERR5	RXERR4	RXERR3	RXERR2	RXERR1	RXERR0
W								
Reset:	0	0	0	0	0	0	0	0

= Unimplemented

Figure 13-17. MSCAN Receive Error Counter (CANRXERR)

1 Read: Only when in sleep mode (SLPRQ = 1 and SLPAK = 1) or initialization mode (INITRQ = 1 and INITAK = 1)
Write: Unimplemented

NOTE

Reading this register when in any other mode other than sleep or initialization mode may return an incorrect value. For MCUs with dual CPUs, this may result in a CPU fault condition.

Module Base + 0x00XF

Access: User read/write[1]

	7	6	5	4	3	2	1	0
R	TSR7	TSR6	TSR5	TSR4	TSR3	TSR2	TSR1	TSR0
W								
Reset:	x	x	x	x	x	x	x	x

Figure 13-37. Time Stamp Register — Low Byte (TSRL)

1 Read: or transmit buffers: Anytime when TXEx flag is set (see Section 13.3.2.7, “MSCAN Transmitter Flag Register (CANTFLG)”) and the corresponding transmit buffer is selected in CANTBSEL (see Section 13.3.2.11, “MSCAN Transmit Buffer Selection Register (CANTBSEL)”). For receive buffers: Anytime when RXF is set.
Write: Unimplemented

NOTE

The reserved bit SCIDRH[2:0] are designed for factory test purposes only, and are not intended for general user access. Writing to these bit is possible when in special mode and can alter the modules functionality.

Table 18-13. SCIDRH and SCIDRL Field Descriptions

Field	Description
SCIDRH 7 R8	**Received Bit 8** — R8 is the ninth data bit received when the SCI is configured for 9-bit data format (M = 1).
SCIDRH 6 T8	**Transmit Bit 8** — T8 is the ninth data bit transmitted when the SCI is configured for 9-bit data format (M = 1).
SCIDRL 7:0 R[7:0] T[7:0]	**R7:R0** — Received bits seven through zero for 9-bit or 8-bit data formats **T7:T0** — Transmit bits seven through zero for 9-bit or 8-bit formats

NOTE

If the value of T8 is the same as in the previous transmission, T8 does not have to be rewritten.The same value is transmitted until T8 is rewritten

In 8-bit data format, only SCI data register low (SCIDRL) needs to be accessed.

When transmitting in 9-bit data format and using 8-bit write instructions, write first to SCI data register high (SCIDRH), then SCIDRL.

18.4 Functional Description

This section provides a complete functional description of the SCI block, detailing the operation of the design from the end user perspective in a number of subsections.

Figure 18-14 shows the structure of the SCI module. The SCI allows full duplex, asynchronous, serial communication between the CPU and remote devices, including other CPUs. The SCI transmitter and receiver operate independently, although they use the same baud rate generator. The CPU monitors the status of the SCI, writes the data to be transmitted, and processes received data.

20.3.1.6 IIC Control Register 2(IBCR2)

Module Base + 0x0005

	7	6	5	4	3	2	1	0
R	GCEN	ADTYPE	0	0	0	ADR10	ADR9	ADR8
W								
Reset	0	0	0	0	0	0	0	0

Figure 20-9. IIC Bus Control Register 2(IBCR2)

This register contains the variables used in general call and in ten-bit address.

Read and write anytime

Table 20-10. IBCR2 Field Descriptions

Field	Description
7 GCEN	**General Call Enable**. 0 General call is disabled. The module dont receive any general call data and address. 1 enable general call. It indicates that the module can receive address and any data.
6 ADTYPE	**Address Type— This bit selects the address length. The variable must be configured correctly before IIC enters slave mode.** **0 7-bit address** **1 10-bit address**
5,4,3 RESERVED	**Reserved** — Bit 5,4 and 3 of the IBCR2 are reserved for future compatibility. These bits will always read 0.
2:0 ADR[10:8]	**Slave Address [10:8]** —These 3 bits represent the MSB of the 10-bit address when address type is asserted (ADTYPE = 1).

20.4 Functional Description

This section provides a complete functional description of the IICV3.

20.4.1 I-Bus Protocol

The IIC bus system uses a serial data line (SDA) and a serial clock line (SCL) for data transfer. All devices connected to it must have open drain or open collector outputs. Logic AND function is exercised on both lines with external pull-up resistors. The value of these resistors is system dependent.

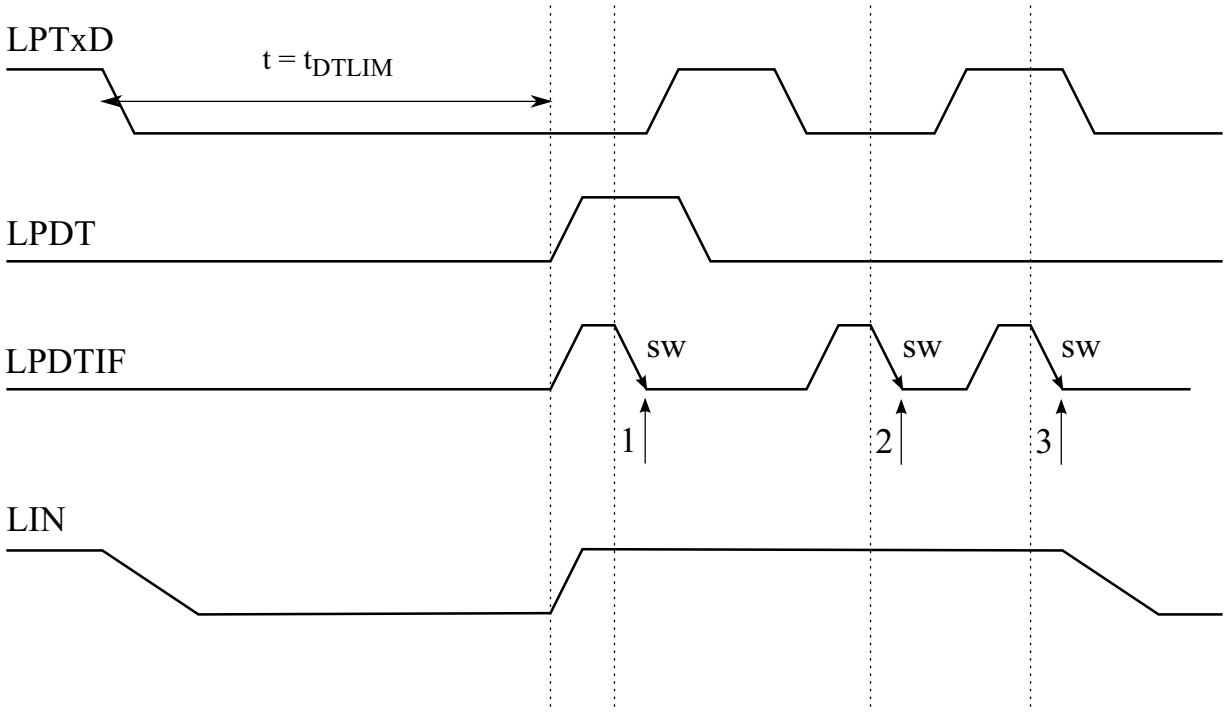
Normally, a standard communication is composed of four parts: START signal, slave address transmission, data transfer and STOP signal. They are described briefly in the following sections and illustrated in Figure 20-10.

After clearing LPDTIF, if the TxD-dominant timeout condition is still present or the LPTxD pin is dominant while being in normal mode, the transmitter remains disabled and the LPDTIF flag is set after a time again to indicate that the attempt to re-enable has failed. This time is equal to:

- minimum 1 IRC period (1 us) + 2 bus periods
- maximum 2 IRC periods (2 us) + 3 bus periods

If the bit LPDTIE is set in the LPIE register, an interrupt is requested.

Figure 21-13 shows the different scenarios of TxD-dominant timeout interrupt handling.



1: Flag cleared, transmitter re-enable not successful because TxD-dominant timeout condition is still present
2: Flag cleared, transmitter re-enable not successful because LPTxD is dominant
3: Flag cleared, transmitter re-enable successful

Figure 21-13. TxD-dominant timeout interrupt handling

B.2 IRC and OSC Electrical Specifications

Table B-3. IRC electrical characteristics

Num	Rating	Symbol	Min	Typ	Max	Unit
1	Junction Temperature - 40° to 150° Celsius Internal Reference Frequency, factory trimmed	f_{IRC1M_TRIM}	0.987	1.000	1.013	MHz
2	Junction Temperature 150° to 175° Celsius Internal Reference Frequency, factory trimmed	f_{IRC1M_TRIM}	0.9855	—	1.0145	MHz

Table B-4. OSC electrical characteristics

Num	Rating	Symbol	Min	Typ	Max	Unit
1	Nominal crystal or resonator frequency	f_{OSC}	4.0	—	20	MHz
2	Startup Current	i_{OSC}	100	—	—	μA
3a	Oscillator start-up time (4MHz)[1]	t_{UPOSC}	—	2	10	ms
3b	Oscillator start-up time (8MHz)[1]	t_{UPOSC}	—	1.6	8	ms
3c	Oscillator start-up time (16MHz)[1]	t_{UPOSC}	—	1	5	ms
3d	Oscillator start-up time (20MHz)[1]	t_{UPOSC}	—	1	4	ms
4	Clock Monitor Failure Assert Frequency	f_{CMFA}	200	450	1200	KHz
5	Input Capacitance (EXTAL, XTAL pins)	C_{IN}	—	7	—	pF
6	EXTAL Pin Input Hysteresis	$V_{HYS,EXTAL}$	—	120	—	mV
7	EXTAL Pin oscillation amplitude (loop controlled Pierce)	$V_{PP,EXTAL}$	—	1.0	—	V
8	EXTAL Pin oscillation required amplitude[2]	$V_{PP,EXTAL}$	0.8	—	1.5	V

1 These values apply for carefully designed PCB layouts with capacitors that match the crystal/resonator requirements.

2 Needs to be measured at room temperature on the application board using a probe with very low (<=5pF) input capacitance.

B.3 Phase Locked Loop

B.3.1 Jitter Information

With each transition of the feedback clock, the deviation from the reference clock is measured and the input voltage to the VCO is adjusted accordingly.The adjustment is done continuously with no abrupt changes in the VCOCLK frequency. Noise, voltage, temperature and other factors cause slight variations in the control loop resulting in a clock jitter. This jitter affects the real minimum and maximum clock periods as illustrated in Figure B-2.

Table C-2. ADC Electrical Characteristics

Supply voltage: MC9S12ZVL(S)32\16\8: 3.13V ≤ V_{DDX} ≤ 5.5V, MC9S12ZVL(A)128\96\64: 3.2V ≤ V_{DDX} ≤ 5.15V, -40°C < T_J < 175°C						
Num	**Rating**	**Symbol**	**Min**	**Typ**	**Max**	**Unit**
1	Max input source resistance[1]	R_S	—	—	1	KΩ
2	Total input capacitance Non sampling Total input capacitance Sampling	C_{INN} C_{INS}	— —	— —	10 16	pF
3	Input internal Resistance	R_{INA}	-	5	15	kΩ
4	Disruptive analog input current	I_{NA}	-2.5	—	2.5	mA
5	Coupling ratio positive current injection	K_p	—	—	1E-4	A/A
6	Coupling ratio negative current injection	K_n	—	—	5E-3	A/A

1 1 Refer to C.1.1.2 for further information concerning source resistance

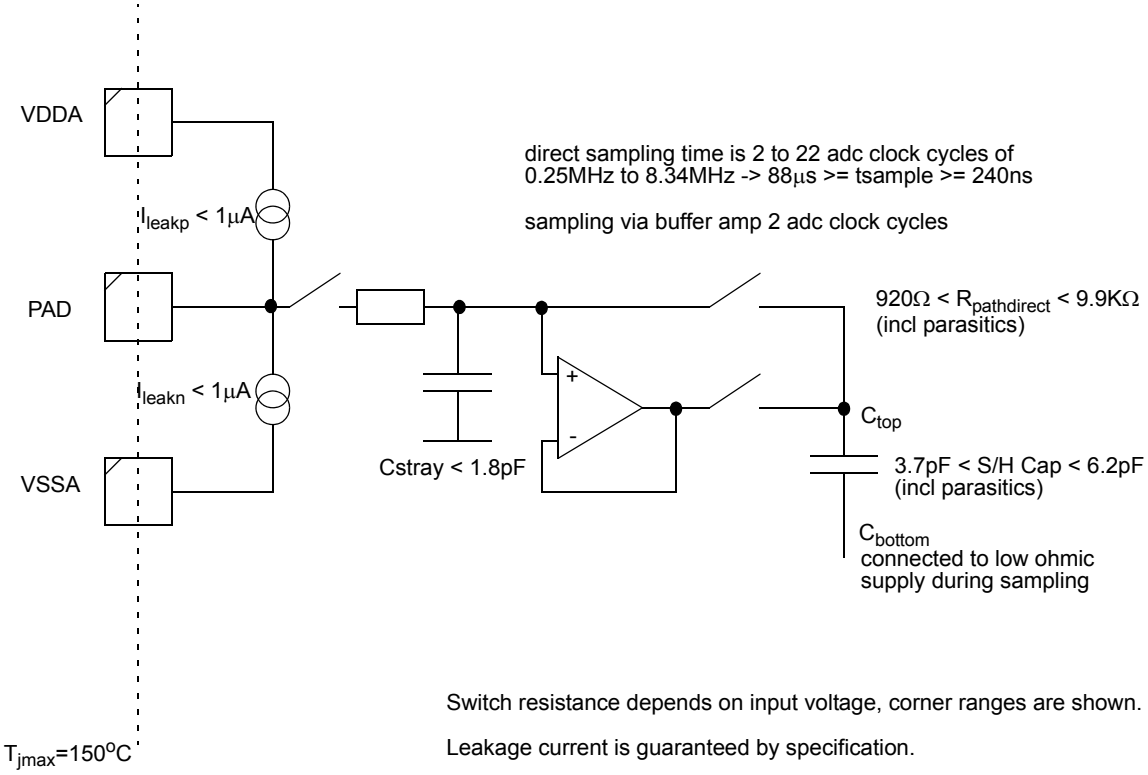
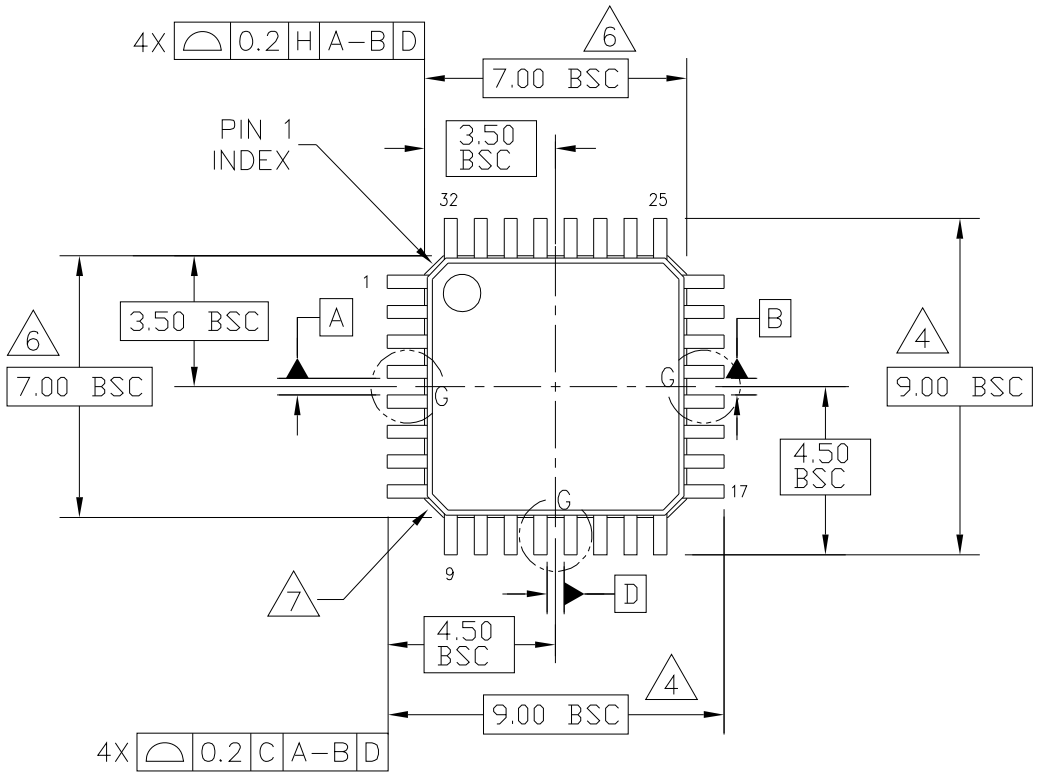
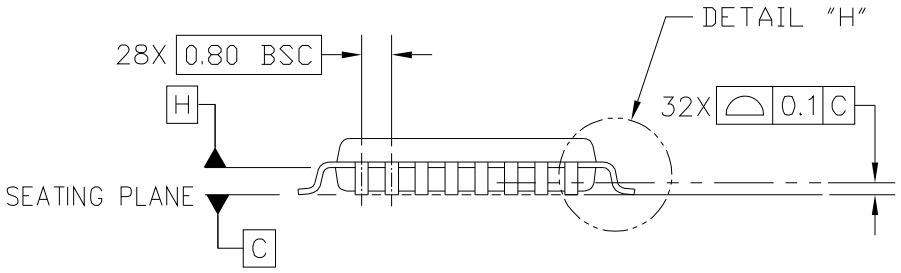


Figure C-1. ADC input logic





© FREESCALE SEMICONDUCTOR, INC. ALL RIGHTS RESERVED.	MECHANICAL OUTLINE	PRINT VERSION NOT TO SCALE
TITLE: LOW PROFILE QUAD FLAT PACK (LQFP) 32 LEAD, 0.8 PITCH (7 X 7 X 1.4)	DOCUMENT NO: 98ASH70029A	REV: D
	CASE NUMBER: 873A-03	19 MAY 2005
	STANDARD: JEDEC MS-026 BBA	