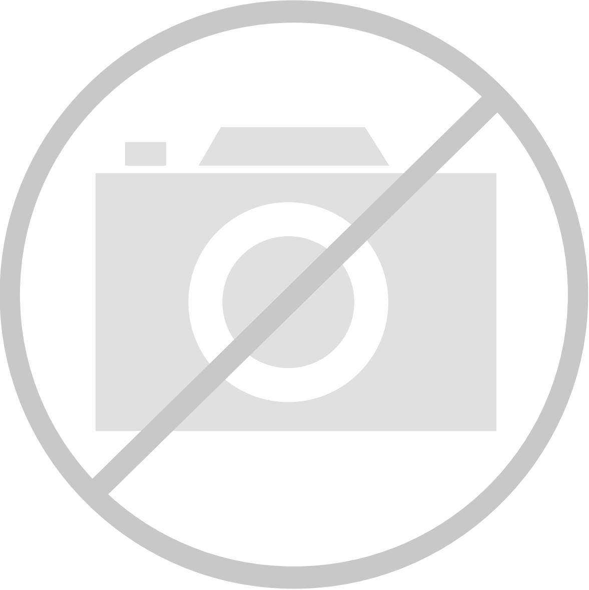


Welcome to E-XFL.COM

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	e200z4d
Core Size	32-Bit Single-Core
Speed	120MHz
Connectivity	CANbus, LINbus, SPI, UART/USART
Peripherals	DMA, POR, PWM, WDT
Number of I/O	-
Program Memory Size	1MB (1M x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	128K x 8
Voltage - Supply (Vcc/Vdd)	3V ~ 5.5V
Data Converters	A/D 32x12b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 105°C (TA)
Mounting Type	Surface Mount
Package / Case	257-LFBGA
Supplier Device Package	257-MAPBGA (14x14)
Purchase URL	https://www.e-xfl.com/pro/item?MUrl=&PartUrl=mpxs2010vmm120

Chapter 12
Clock Monitor Unit (CMU)

Chapter 13
Cross-Triggering Unit (CTU)

Chapter 34
Nexus Port Controller (NPC)

Chapter 35
Oscillators

Table 13-26. CTU_CNTRNG field descriptions

Field	Description
VALUE	Setting at one this bit you mask the same bit of the expected counter and of the internal counter. In this way you can set a range for the expected conversion time.

13.10.19 FIFO DMA control register (FDCR)

Address: Base + 0x006C

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	0	0	0	0	0	0	0	0	0	0	0	DE3	DE2	DE1	DE0
												R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Figure 13-31. FIFO DMA control register (FDCR)

Table 13-27. FDCR field description

Name	Description
0-11	Reserved
12-15 DEx	This bit enables DMA for the FIFOx

13.10.20 FIFO control register (FCR)

Address: Base + 0x0070

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

16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
OR_EN3	OF_EN3	EMPTY_EN3	FULL_EN3	OR_EN2	OF_EN2	EMPTY_EN2	FULL_EN2	OR_EN1	OF_EN1	EMPTY_EN1	FULL_EN1	OR_EN0	OF_EN0	EMPTY_EN0	FULL_EN0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Figure 13-32. FIFO control register (FCR)

Table 19-29. TCDn Word 2 (TCDn.nbytes) field description

Name	Description	Value
nbytes[0:31]	Inner “minor” byte transfer count	Number of bytes to be transferred in each service request of the channel. As a channel is activated, the contents of the appropriate TCD is loaded into the eDMA engine, and the appropriate reads and writes performed until the complete byte transfer count has been transferred. This is an indivisible operation and cannot be stalled or halted. After the minor count is exhausted, the current values of the saddr and daddr are written back into the local memory, the major iteration count is decremented and restored to the local memory. If the major iteration count is completed, additional processing is performed. The nbytes value 0x0000_0000 is interpreted as 0x1_0000_0000, thus specifying a 4 GB transfer.

When minor loop mapping (DMACR[EMLM] = 1) is enabled, TCD word2 is redefined as four fields: a source minor loop offset enable, a destination minor loop offset enable, a minor loop offset field and a nbytes field.

Register address: DMA_Offset + 0x1000 + (32 x n) + 0x08

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
R W	smloe	dmloe	mloff[0:13] or nbytes[0:13]													
RESET:	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-

	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
R W	mloff[14:19] or nbytes[14:19]						nbytes[20:29]									
RESET:	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-

▨ = Unimplemented

Figure 19-21. TCDn Word 2 (TCDn.nbytes) Field (DMACR[EMLM] = 1)

Table 19-30. TCDn Word 2 (TCDn.nbytes) field descriptions

Name	Description	Value
smloe	Source minor loop offset enable	This flag selects whether the minor loop offset is applied to the source address upon minor loop completion. 0 The minor loop offset is not applied to the saddr. 1 The minor loop offset is applied to the saddr.

2. Select the block, or blocks to be receive array integrity check by writing ones to the appropriate registers in LMS or HBS registers.

   NOTE

   Locked Blocks can be tested with Array Integrity if selected in LMS and HBS.

   It is not possible to do UTest operations on the shadow block.

   While Array Integrity is being executed, flash memory array accesses thru the BIU should not be requested.

3. If desired, set the UT0[AIS] bit to 1 for sequential addressing only.

   NOTE

   For normal integrity checks of the flash memory, sequential addressing is recommended.

   If it is required to more fully check the read path (in a diagnostic mode), it is recommend that AIS be left at 0, to use the address sequence that checks the read path more fully, and examine read transitions. This sequence takes more time.

4. Seed the MISR UM0 thru UM4 with desired values.
5. Set the UT0[AIE] bit.
   If desired, the Array Integrity operation may be aborted prior to UT0[AID] going high. This may be done by clearing the UT0[AIE] bit and then continuing to the next step. It should be noted that in the event of an aborted array integrity check the MISR registers will contain a signature for the portion of the operation that was completed prior to the abort, and will not be deterministic. Prior to doing another array integrity operation, the UM0, UM1, UM2 and UM3 registers may need to be initialized to the desired seed value by doing register writes.
6. Wait until the UT0[AID] bit goes high.
7. Read values in the MISR registers (UM0 through UM4) to ensure correct signature.
8. Write a logic 0 to the UT0[AIE] bit.

23.1.5.12 ECC logic check

ECC logic can be checked by providing data to be read in the UT0[DSI], UT1[DAI] and/or UT2[DAI] registers. Then array reads can be done, ensuring expected results. The ECC logic check consists of the following sequence of events:

1. Enable UTest mode.
2. Write UT0[EIE] to 1.
3. Write UT0[DSI], UT1[DAI] and/or UT2[DAI] bits to provide data and check bit values to be read. Single or Double bit detections/corrections can be simulated by properly choosing Data and Check Bit combinations.
4. Write double word address to receive the data inputted in step 3 into the ADR register.

Table 23-17. UM3 Field Descriptions

Field	Description
0-31 MISR[127:96]	See the description of the MISR in Table 23-14.

23.1.6.12.5 UM4 Register

The following field and bit descriptions fully define the UM4 register.

Offset 0x0058 Access: User read/write

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
R	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	MISR																
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Figure 23-17. UM4 register

MISR register functions are shown in Table 23-18.

Table 23-18. UM4 field descriptions

Field	Description
MISR[144:128]	See the description of the MISR in Table 23-14.

23.1.6.13 Nonvolatile Private Censorship Password 0 register (NVPWD0)

Offset 0x3DD8 Access: User read/write

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
R W	PWD[31:16]															
Reset	1	1	1	1	1	1	1	0	1	1	1	0	1	1	0	1

	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
R W	PWD[15:0]															
Reset	1	1	1	1	1	0	1	0	1	1	0	0	1	1	1	0

Figure 23-18. Nonvolatile Private Censorship Password 0 register (NVPWD0)

The Nonvolatile Private Censorship Password 0 register (NVPWD0) contains the 32 LSB of the password used to validate the censorship information contained in the NVSCI register.

Table 23-19. NVPWD0 field descriptions

Field	Description
PWD	These bits represent the private censorship password.

Offset 0x3DE0 Access: User read/write

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

	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
R W	CW															
Reset	0	1	0	1	0	1	0	1	1	0	1	0	1	0	1	0

Figure 23-21. Nonvolatile System Censoring Information register (NVSCI) for cut2/3

The Nonvolatile System Censoring Information register (NVSCI) stores the censorship control word of the device. It is read during the reset phase of the flash memory module and the protection mechanisms are activated consequently. The devices are delivered uncensored to the user.

Table 23-21. NVSCI field descriptions

Field	Description
SC	Serial Censorship control word. These bits represent the Serial Censorship Control Word (SCCW). If SC[15:0] = 0x55AA, the Public Access is disabled. If SC[15:0] ≠ 0x55AA, the Public Access is enabled.
CW	Censorship control Word. These bits represent the Censorship Control Word (CCW). If CW = 0x55AA, the Censored Mode is disabled. If CW ≠ 0x55AA, the Censored Mode is enabled.

23.1.7 User option bits

Table 23-22 describes the user option bits on this device. These are programmed in the BIU4 register (see Section 23.1.6.8, Bus Interface Unit 4 Register (BIU4)) and verified using the UOPS register in the SSCM (see Section 48.3.1.8, User Option Status Register (UOPS)).

Table 23-22. User option bits

Bit number	Function name	Description
31–20	FCCU_CFG	FCCU configuration Bits 26–31: FCCU_CFG.FOP Bits 23–25: FCCU_CFG.FOM Bit 22: FCCU_CFG.PS Bit 21: FCCU_CFG.SM Bit 20: FCCU_CFG.CM
19–10	Reserved	

- Maskable interrupts
- Independent of the transmission medium (an external transceiver is assumed)
- Short latency time due to an arbitration scheme for high-priority messages
- Low power modes, with programmable wake up on bus activity

NOTE

The individual Rx Mask per Message Buffer feature may not be available in low cost MCUs. Please consult the specific MCU documentation to find out if this feature is supported.

24.1.3 Modes of operation

The FlexCAN module has four functional modes: Normal Mode (User and Supervisor), Freeze Mode, Listen-Only Mode and Loop-Back Mode. There are also two low power modes: Disable Mode and Stop Mode.

24.1.3.1 Normal Mode (User or Supervisor)

In Normal Mode, the module operates receiving and/or transmitting message frames, errors are handled normally and all the CAN Protocol functions are enabled. User and Supervisor Modes differ in the access to some restricted control registers.

24.1.3.2 Freeze Mode

It is enabled when the FRZ bit in the MCR Register is asserted. If enabled, Freeze Mode is entered when the HALT bit in MCR is set or when Debug Mode is requested at MCU level. In this mode, no transmission or reception of frames is done and synchronicity to the CAN bus is lost. See Section 24.4.9.1, Freeze Mode, for more information.

24.1.3.3 Listen-Only Mode

The module enters this mode when the LOM bit in the Control Register is asserted. In this mode, transmission is disabled, all error counters are frozen and the module operates in a CAN Error Passive mode. Only messages acknowledged by another CAN station will be received. If FlexCAN detects a message that has not been acknowledged, it will flag a BIT0 error (without changing the REC), as if it was trying to acknowledge the message.

24.1.3.4 Loop-Back Mode

The module enters this mode when the LPB bit in the Control Register is asserted. In this mode, FlexCAN performs an internal loop back that can be used for self test operation. The bit stream output of the transmitter is internally fed back to the receiver input. The Rx CAN input pin is ignored and the Tx CAN output goes to the recessive state (logic ‘1’). FlexCAN behaves as it normally does when transmitting and treats its own transmitted message as a message received from a remote node. In this mode, FlexCAN ignores the bit sent during the ACK slot in the CAN frame acknowledge field to ensure proper reception of its own message. Both transmit and receive interrupts are generated.

- Time Segment 2: This segment represents the Phase Segment 2 of the CAN standard. It can be programmed by setting the PSEG2 field of the CTRL Register (plus 1) to be 2 to 8 time quanta long

$$Bit \quad Rate = \frac{f_{Tq}}{(number \quad of \quad Time \quad Quanta)}$$

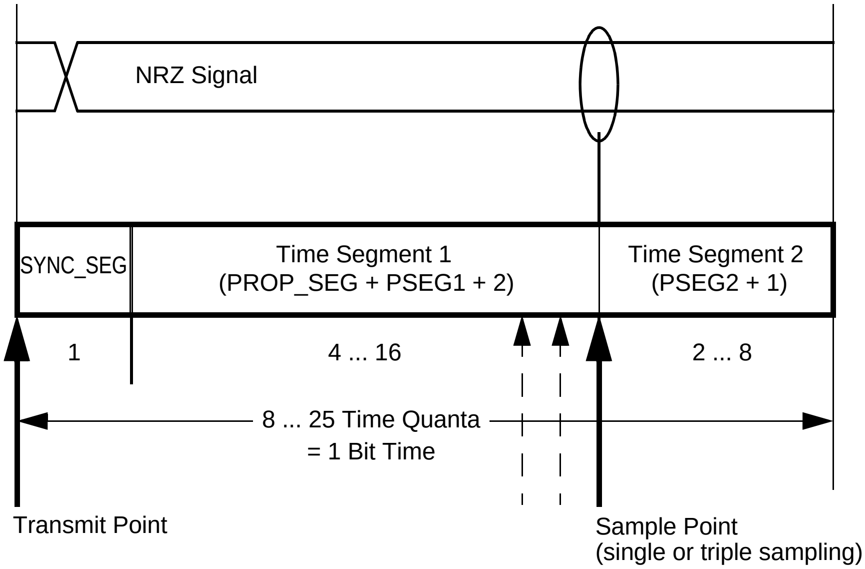


Figure 24-15. Segments within the Bit Time

Table 24-9. Time Segment Syntax

Syntax	Description
SYNC_SEG	System expects transitions to occur on the bus during this period.
Transmit Point	A node in transmit mode transfers a new value to the CAN bus at this point.
Sample Point	A node samples the bus at this point. If the three samples per bit option is selected, then this point marks the position of the third sample.

Table 24-10 gives an overview of the CAN compliant segment settings and the related parameter values.

Table 24-10. CAN Standard Compliant Bit Time Segment Settings

Time Segment 1	Time Segment 2	Re-synchronization Jump Width
5 .. 10	2	1 .. 2
4 .. 11	3	1 .. 3
5 .. 12	4	1 .. 4
6 .. 13	5	1 .. 4
7 .. 14	6	1 .. 4
8 .. 15	7	1 .. 4

26.5.2.65 Receive FIFO Range Filter Configuration Register (FR_RFRFCFR)

Base + 0x0098 16-bit write access required Write: WMD, IBD, SEL: Any Time
SID: *POC:config*

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
R	0	IBD	SEL		0	SID_A/SID_B										
W	WMD				▨											
Rese t	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Figure 26-66. Receive FIFO Range Filter Configuration Register (FR_RFRFCFR)

This register provides access to the four internal frame ID range filter boundary registers of the selected FIFO. For details on frame ID range filter see Section 26.6.9.9, FIFO Filtering.

Table 26-74. FR_RFRFCFR Field Descriptions

Field	Description
WMD	**Write Mode** — This control bit defines the write mode of this register. 0 Write to all fields in this register on write access. 1 Write to SEL and IBD field only on write access.
IBD	**Interval Boundary** — This control bit selects the interval boundary to be programmed with the SID value. 0 program lower interval boundary 1 program upper interval boundary
SEL	**Filter Selector** — This control field selects the frame ID range filter to be accessed. 00 select frame ID range filter 0. 01 select frame ID range filter 1. 10 select frame ID range filter 2. 11 select frame ID range filter 3.
SID_A SID_B	**Slot ID —** Defines the IBD-selected frame ID boundary value for the SEL-selected range filter.

26.5.2.66 Receive FIFO Range Filter Control Register (FR_RFRFCTR)

Base + 0x009A Write: Anytime

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
R	0	0	0	0	F3MD	F2MD	F1MD	F0MD	0	0	0	0	F3EN	F2EN	F1EN	F0EN
W	▨	▨	▨	▨					▨	▨	▨	▨				
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Figure 26-67. Receive FIFO Range Filter Control Register (FR_RFRFCTR)

This register is used to enable and disable each frame ID range filter and to define whether it is running as acceptance or rejection filter.

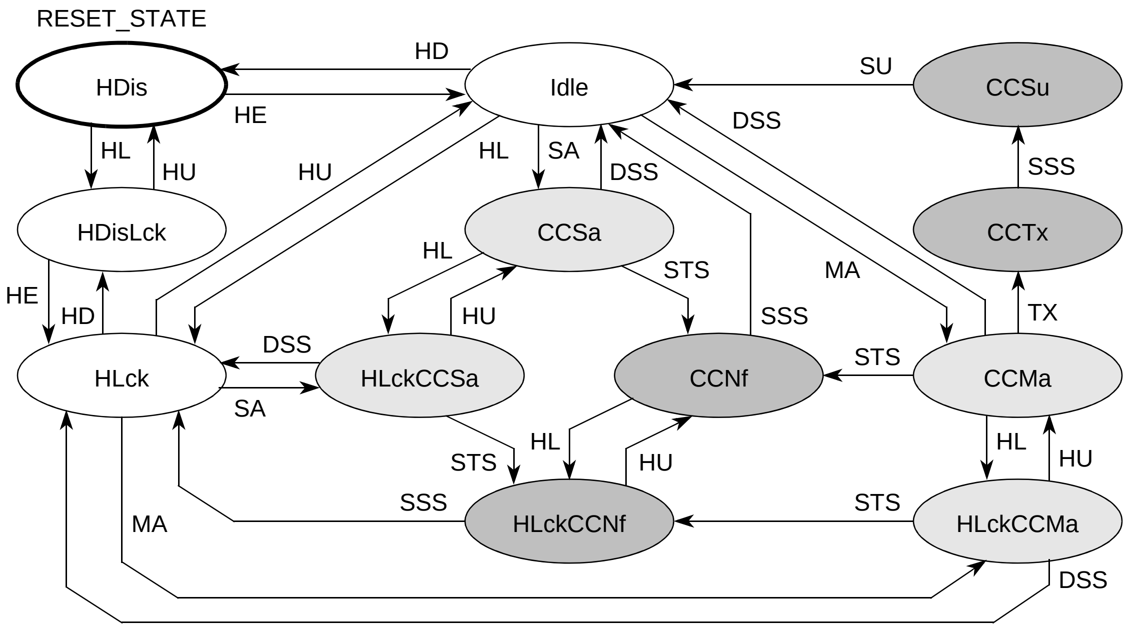


Figure 26-130. Single Transmit Message Buffer States

Table 26-106. Single Transmit Message Buffer State Description

State	FR_MBCCSRn		Access Region		Description
	EDS	**LCKS**	**Appl.**	**Module**	
Idle	1	0	–	CM, SR	**Idle** - Message Buffer is idle. Included in message buffer search.
HDis	0	0	CFG	–	**Disabled** - Message Buffer under configuration. Excluded from message buffer search.
HDisLck	0	1	CFG	–	**Disabled and Locked** - Message Buffer under configuration. Excluded from message buffer search.
HLck	1	1	MSG	SR	Locked - Applications access to data, control, and status. Included in message buffer search.
CCSa	1	0	–	–	Slot Assigned - Message buffer assigned to next static slot. Ready for Null Frame transmission.
HLckCCSa	1	1	MSG	–	Locked and Slot Assigned - Applications access to data, control, and status.Message buffer assigned to next static slot
CCNf	1	0	–	NF	Null Frame Transmission Header is used for null frame transmission.
HLckCCNf	1	1	MSG	NF	Locked and Null Frame Transmission - Applications access to data, control, and status. Header is used for null frame transmission.
CCMa	1	0	–	CM	Message Available - Message buffer is assigned to next slot and cycle counter filter matches.
HLckCCMa	1	1	MSG	–	Locked and Message Available - Applications access to data, control, and status. Message buffer is assigned to next slot and cycle counter filter matches.
CCTx	1	0	–	TX	Message Transmission - Message buffer data transmit. Payload data from buffer transmitted

28.5.1.3 Unique Vector for Each Interrupt Request Source

Each peripheral and software settable interrupt request is assigned a hardwired unique 9-bit vector. Software settable interrupts 0 - 7 are assigned vectors 0 - 7 respectively. The peripheral interrupt requests are assigned vectors 8 to as high as needed to cover all of the peripheral interrupt requests.

28.5.2 Priority Management

The asserted interrupt requests are compared to each other based on their PRI*x* values set in INTC_PSR*n*. The result of that comparison also is compared to PRI in the associated INTC_CPR_PRC0 register. The results of those comparisons are used to manage the priority of the ISR being executed by the associated processor. The associated LIFO also assists in managing that priority.

28.5.2.1 Current Priority and Preemption

The priority arbitrator, selector, encoder, and comparator subblocks shown in Figure 28-1 are used to compare the priority of the asserted interrupt requests to the current priority. If the priority of any asserted peripheral or software settable interrupt request is higher than the current priority for a given processor, then the interrupt request to the processor is asserted. Also, a unique vector for the preempting peripheral or software settable interrupt request is generated for the associated INTC_IACKR_PRC0 register, and if in hardware vector mode, for the interrupt vector provided to the processor.

28.5.2.1.1 Priority Arbitrator Subblock

The priority arbitrator subblock for each processor compares all the priorities of all of the asserted interrupt requests assigned to that processor, both peripheral and software settable. The output of the priority arbitrator subblock is the highest of those priorities assigned to a given processor. Also, any interrupt requests which have this highest priority are output as asserted interrupt requests to the associated request selector subblock.

28.5.2.1.2 Request Selector Subblock

If only one interrupt request from the associated priority arbitrator subblock is asserted, then it is passed as asserted to the associated vector encoder subblock. If multiple interrupt requests from the associated priority arbitrator subblock are asserted, then only the one with the lowest vector is passed as asserted to the associated vector encoder subblock. The lower vector is chosen regardless of the time order of the assertions of the peripheral or software settable interrupt requests.

28.5.2.1.3 Vector Encoder Subblock

The vector encoder subblock generates the unique 9-bit vector for the asserted interrupt request from the request selector subblock for the associated processor.

28.5.2.1.4 Priority Comparator Subblock

The priority comparator subblock compares the highest priority output from the associated priority arbitrator subblock with PRI in the associated INTC_CPR_PRC0. If the priority comparator subblock detects that this highest priority is higher than the current priority, then it asserts the interrupt request to

Table 31-14. LINIER field descriptions (continued)

Field	Description
HEIE	Header Error Interrupt Enable 0: No interrupt on Break Delimiter error, Synch Field error, ID field error 1: Interrupt generated on Break Delimiter error, Synch Field error, ID field error
FEIE	Framing Error Interrupt Enable 0: No interrupt on Framing error 1: Interrupt generated on Framing error
BOIE	Buffer Overrun Interrupt Enable 0: No interrupt on Buffer overrun 1: Interrupt generated on Buffer overrun
LSIE	LIN State Interrupt Enable 0: No interrupt on LIN state change 1: Interrupt generated on LIN state change This interrupt can be used for debugging purposes. It has no status flag but is reset when writing '1111' into the LIN state bits in the LINSR register.
WUIE	Wake-up Interrupt Enable 0: No interrupt when WUF bit in LINSR or UARTSR is set 1: Interrupt generated when WUF bit in LINSR or UARTSR is set
DBFIE	Data Buffer Full Interrupt Enable 0: No interrupt when buffer data register is full 1: Interrupt generated when data buffer register is full
DBEIETOIE	Data Buffer Empty Interrupt Enable / Timeout Interrupt Enable 0: No interrupt when buffer data register is empty 1: Interrupt generated when data buffer register is empty **Note:** An interrupt is generated if this bit is set and one of the following is true: LINFlexD is in LIN mode and LINSR[DBEF] is set LINFlexD is in UART mode and UARTSR[TO] is set
DRIE	Data Reception Complete Interrupt Enable 0: No interrupt when data reception is completed 1: Interrupt generated when data received flag (DRF) in LINSR or UARTSR is set
DTIE	Data Transmitted Interrupt Enable 0: No interrupt when data transmission is completed 1: Interrupt generated when data transmitted flag (DTF) is set in LINSR or UARTSR register
HRIE	Header Received Interrupt Enable 0: No interrupt when a valid LIN header has been received 1: Interrupt generated when a valid LIN header has been received, that is, HRF bit in LINSR register is set

Once a valid mode transition request is detected, the target mode configuration information is loaded from the corresponding ME_<*mode*>_MC register. The mode transition request may require a number of cycles depending on the programmed configuration, and software should check the S_CURRENT_MODE bit field and the S_MTRANS bit of the global status register to verify when the mode has been correctly entered and the transition process has completed. For a description of valid mode requests, please refer to Section 32.4.5, Mode Transition Interrupts.

Any modification of the mode configuration register of the currently selected mode will not be taken into account immediately but on the next request to enter this mode. This means that transition requests such as RUN0…3 Æ RUN0…3, DRUN Æ DRUN, SAFE Æ SAFE, and TEST Æ TEST are considered valid mode transition requests. As soon as the mode request is accepted as valid, the S_MTRANS bit is set till the status in the register matches the configuration programmed in the respective ME_<*mode*>_MC register.

NOTE

It is recommended that software poll the S_MTRANS bit in the register after requesting a transition to HALT0 or STOP0 modes.

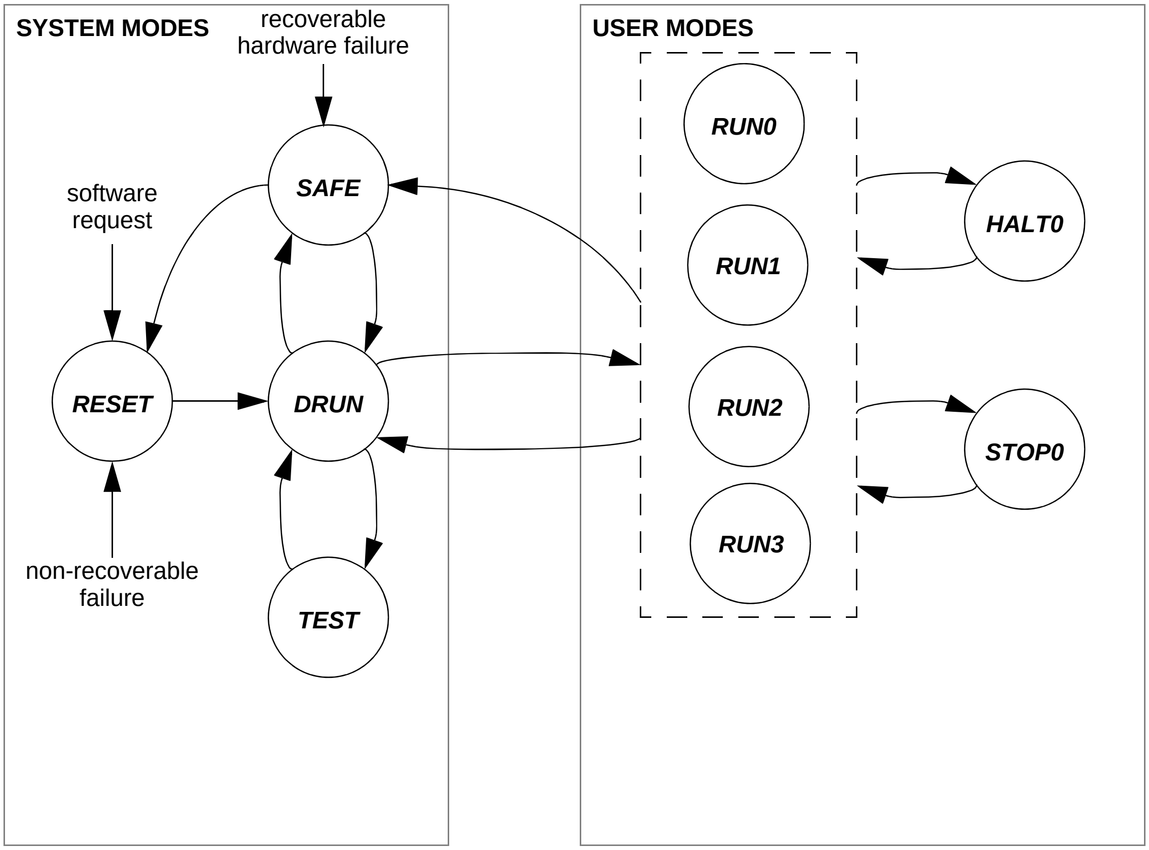


Figure 32-25. MC_ME Mode Diagram

Table 33-11. Data Trace Size (DSZ) encodings (TCODE = 5, 6, 13, 14)

DTM size encoding	Transfer size
000	Byte
001	Halfword (two bytes)
010	Word (four bytes)
011	Doubleword (eight bytes)
100–111	Reserved

33.6.2 Data Trace

This section deals with the data trace mechanism supported by the NXSS_0 and NXSS_1 modules. Data trace is implemented via data write messaging (DWM) and data read messaging (DRM).

33.6.3 Data Trace Messaging (DTM)

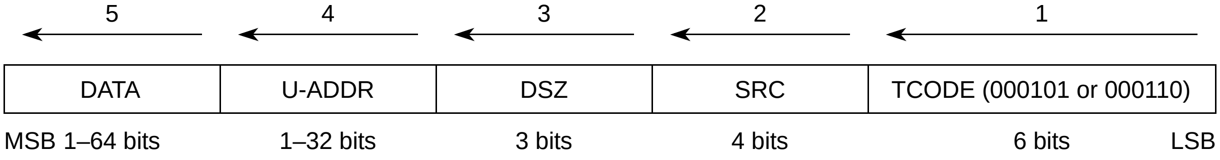
NXSS_0 and NXSS_1 data trace messaging is accomplished by snooping the NXSS_0 and NXSS_1 data bus, and storing the information for qualifying accesses (based on enabled features and matching AHB Master ID and target addresses). The NXSS module traces all data access that meet the selected range and attributes.

33.6.4 DTM Message Formats

The NXSS module supports five types of DTM Messages — data write, data read, data write synchronization, data read synchronization, and error messages.

33.6.4.1 Data Write and Data Read Messages

The data write and data read messages contain the data write/read value and the address of the write/read access, relative to the previous data trace message. Data write message and data read message information is messaged out in the following format:



Max length = 109 bits; Min length = 15 bits

Figure 33-11. Data Write/Read Message Format

33.6.4.2 DTM Overflow Error Messages

An error message occurs when a new message cannot be queued due to the message queue being full. The FIFO discards incoming messages until it has completely emptied the queue. After it is emptied, an error

Table 39-7. PMUCTRL_FAULT field descriptions

Field	Description
BB_LV	Bypass Ballast Low Voltage. This field monitors the NOT of nBYPASS_BALLAST_LV[3:0] bus. 0 BB_LV[n] = NOT (nBYPASS_BALLAST_LV[n] = ‘1’); where n goes from 3 down to 0 1 BB_LV[n] = NOT (nBYPASS_BALLAST_LV[n] = ‘0’); where n goes from 3 down to 0
FLNCF	Flash memory voltage monitor non-critical fault 0 No flash memory voltage monitor fault is active 1 Flash memory voltage monitor fault is active
IONCF	IO voltage monitor non-critical fault 0 No IO voltage monitor fault is active 1 IO voltage monitor fault is active
RENCF	Regulator voltage monitor non-critical fault 0 No regulator voltage monitor fault is active 1 Regulator voltage monitor fault is active
LHCF	Low high voltage detector critical fault 0 No low high voltage fault is active 1 Low high voltage fault is active
LNCF	Low voltage detector non-critical fault 0 No low voltage fault is active 1 Low voltage fault is active
HNCF	High voltage detector non-critical fault 0 No high voltage fault is active 1 High voltage fault is active

39.7.5 PMUCTRL interrupt request status register (PMUCTRL_IRQS)

This register allows you to read and clear the various PMU interrupt status bits. To clear these bits, write a ‘1’ to them.

Table 41-6. Destructive Event Reset Disable Register (RGM_DERD) Field Descriptions

Field	Description
D_SOFT_DEST (cut2/3)	Disable software 'destructive' reset 0 A software 'destructive' reset event triggers a reset sequence
D_LVD27_IO	**Disable 2.7 V low-voltage det. (I/O)** 0 A 2.7 V low-voltage det. (I/O) event triggers a reset sequence
D_LVD27_FLASH	**Disable 2.7 V low-voltage det. (flash)** 0 A 2.7 V low-voltage det. (flash) event triggers a reset sequence
D_LVD27_VREG	**Disable 2.7 V low-voltage det. (VREG)** 0 A 2.7 V low-voltage det. (VREG) event triggers a reset sequence
D_HVD12	**Disable 1.2 V high-voltage detected** 0 A 1.2 V high-voltage detected event triggers a reset sequence
D_LVD12	**Disable 1.2 V low-voltage detected** 0 A 1.2 V low-voltage detected event triggers a reset sequence

41.3.1.5 Functional Event Alternate Request Register (RGM_FEAR)

Address 0xC3FE_4010 Access: User read, Supervisor read/write, Test read/write

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
R	0	0	0	0	AR_CMU12_FHL	0	AR_PLL1	0	AR_FCCU_SAFE	AR_CMU0_FHL	AR_CMU0_OLR	AR_PLL0	AR_CWD	0	0	0
W	▨	▨	▨	▨		▨		▨	▨					▨	▨	▨
Reset[*]	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

[*] This register is reset if and only if one of thefollowing occurs:

- Power up
- 1.2 V low-voltage detection (i.e. when the core voltage drops below the point at which the flip-flops can reliably retain their value)

Figure 41-8. Functional Event Alternate Request Register (RGM_FEAR)

This register defines an alternate request to be generated when a reset on a functional event has been disabled. The alternate request can be either a ***SAFE*** mode request to MC_ME or an interrupt request to the system. It can be accessed in read/write in either supervisor mode or test mode. It can be accessed in read only in user mode.

The status flag associated with a given 'destructive' reset event (**RGM_DES.F_*<destructive reset>*** bit) is set when the 'destructive' reset is asserted and the power-on reset is not asserted. It is possible for multiple status bits to be set simultaneously, and it is software's responsibility to determine which reset source is the most critical for the application.

The device's low-voltage detector threshold ensures that, when 1.2 V low-voltage detected is enabled, the supply is sufficient to have the destructive event correctly propagated through the digital logic. Therefore, if a given 'destructive' reset is enabled, the MC_RGM ensures that the associated reset event will be correctly triggered to the full system. However, if the given 'destructive' reset is disabled and the voltage goes below the digital functional threshold, functionality can no longer be ensured, and the reset may or may not be asserted.

An enabled destructive reset will trigger a reset sequence starting from the beginning of ***PHASE0***.

41.4.3 External Reset

The MC_RGM manages the external reset coming from RESET_B. The detection of a falling edge on RESET_B will start the reset sequence from the beginning of ***PHASE1***.

The status flag associated with the external reset falling edge event (**RGM_FES.F_EXR** bit) is set when the external reset is asserted and the power-on reset is not asserted.

The external reset can optionally be disabled by writing bit **RGM_FERD.D_EXR**.

NOTE

The **RGM_FERD** register can be written only once between two power-on reset events.

An enabled external reset will normally trigger a reset sequence starting from the beginning of ***PHASE1***. Nevertheless, the **RGM_FESS** register enables the further configuring of the reset sequence triggered by the external reset. When **RGM_FESS.SS_EXR** is set, the external reset will trigger a reset sequence starting directly from the beginning of ***PHASE3***, skipping ***PHASE1*** and ***PHASE2***. This can be useful especially when an external reset should not reset the flash.

The MC_RGM may also assert the external reset if the reset sequence was triggered by one of the following:

- a power-on reset
- a 'destructive' reset event
- an external reset event
- a 'functional' reset event configured via the **RGM_FBRE** register to assert the external reset

In this case, the external reset is asserted until the end of ***PHASE3***.

41.4.4 Functional resets

A 'functional' reset indicates that an event has occurred after which it can be guaranteed that critical register and memory content is still intact.

47.3 Features

The System Integration Unit Lite supports these distinctive features:

- GPIO
  - — 121 pins which are user-configurable inputs and/or outputs
  - — 22 pins have user-configurable General Purpose Input (GPI) functionality
  - — 99 pins have user-configurable General Purpose Input/Ouput (GPIO) functionality
  - — Dedicated input and output registers for each GPIO pin
- External interrupts
  - — 4 system interrupt vectors for 32 interrupt sources
  - — 32 programmable digital glitch filters
  - — Independent interrupt mask
  - — Edge detection
- System configuration
  - — Pad configuration control

47.3.1 Register protection

The individual registers of System Integration Unit Lite are protected from accidental writes, see Chapter 40, Register Protection (REG_PROT).

47.4 External signal description

The pad configuration allows flexible, centralized control of the pin electrical characteristics of the MCU with the GPIO control providing centralized general purpose I/O for an MCU that multiplexes GPIO with other signals at the I/O pads. These other signals, or alternate functions, will normally be the peripherals functions. The internal multiplexing allows user selection of the input to chip-level signal multiplexors. Each GPIO port communicates via 16 I/O channels. In order to use the pad as a GPIO, the corresponding Pad Configuration Registers (PCR) for all pads used in the port must be configured as GPIO rather than as the alternate pad function.

Table 47-1 lists the external pins used by the SIUL.

Table 47-1. SIUL signal properties

Name	I/O Type	Function
System Configuration		
GPIO	I/O	General-Purpose I/O
External Interrupt		
EIRQ[0:8, 10:18, 22, 30:31, 35:38, 77:81, 96:97]	Input	External Interrupt Request Input

Table 47-10. PCR0 Field Descriptions

Field	Description
SMC	Safe Mode Control This bit supports the overriding of the automatic deactivation of the output buffer of the associated pad upon entering SAFE mode of the SoC. 1: In SoC SAFE mode, the output buffer remains functional. 0: In SoC SAFE mode, the output buffer of the pad is disabled.
APC	Analog Pad Control This bit enables the usage of the pad as analog input. 1: Analog input path switch can be enabled by the ADC. 0: Analog input path from the pad is gated and can not be used.
PA[1:0]	Pad Output Assignment This field is used to select the function that is allowed to drive the output of a multiplexed pad. The PA field size can vary from zero to two bits, depending on the number of output functions associated with this pad. 00: Alternative Mode 0: GPIO. 01: Alternative Mode 1 10: Alternative Mode 2 11: Alternative Mode 3 **Note:** Number of bit depending of the number of actual alternate function. Please refer to datasheet
OBE	Output Buffer Enable This bit enables the output buffer of the pad in case the pad is in GPIO mode. 1: Output Buffer of the pad is enabled when PA = 00. 0: Output Buffer of the pad is disabled when PA = 00.
IBE	Input Buffer Enable This bit enables the input buffer of the pad. 1: Input Buffer of the pad is enabled. 0: Input Buffer of the pad is disabled.
ODE	Open Drain Output Enable This bit controls output driver configuration for the pads connected to this signal. Either open drain or push/pull driver configurations can be selected. This feature applies to output pads only. 1: Open drain enable signal is asserted for the pad. 0: Open drain enable signal is negated for the pad.

SRC	Slew Rate Control
	SRC = 0 Slowest configuration
	SRC = 1 Fastest configuration

WPE	Weak Pull Up/Down Enable This bit controls whether the weak pull up/down devices are enabled/disabled for the pad connected to this signal. 1: Weak pull device enable signal is asserted for the pad. 0: Weak pull device enable signal is negated for the pad.
WPS	Weak Pull Up/Down Select This bit controls whether weak pull up or weak pull down devices are used for the pads connected to this signal when weak pull up/down devices are enabled. 1: The pull up enabled. 0: The pull down enabled.