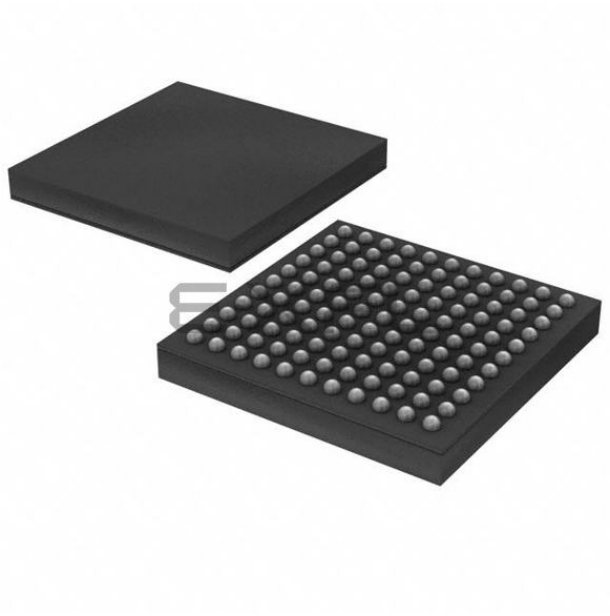


Welcome to E-XFL.COM

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"

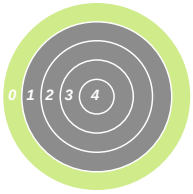
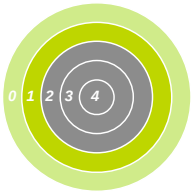
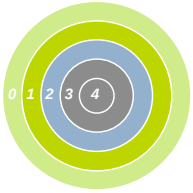
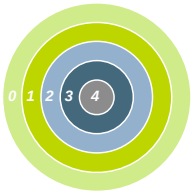
Details	
Product Status	Discontinued at Digi-Key
Core Processor	ARM® Cortex®-M3
Core Size	32-Bit Single-Core
Speed	48MHz
Connectivity	EBI/EMI, I²C, IrDA, SmartCard, SPI, UART/USART, USB
Peripherals	Brown-out Detect/Reset, DMA, LCD, POR, PWM, WDT
Number of I/O	93
Program Memory Size	1MB (1M x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	128K x 8
Voltage - Supply (Vcc/Vdd)	1.85V ~ 3.8V
Data Converters	A/D 8x12b; D/A 2x12b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	120-VFBGA
Supplier Device Package	120-BGA (7x7)
Purchase URL	https://www.e-xfl.com/product-detail/silicon-labs/efm32gg995f1024-bga120t

wake-up time, makes it attractive to remain in low energy modes for long periods and thus saving energy consumption.

Tip

Throughout this document, the first figure in every module description contains an Energy Mode Indicator showing which energy mode(s) the module can operate (see Table 3.1 (p. 8)).

Table 3.1. Energy Mode Description

Energy Mode	Name	Description
	EM0 – Energy Mode 0 (Run mode)	In EM0, the CPU is running and consuming as little as 219 µA/MHz, when running code from flash. All peripherals can be active.
	EM1 – Energy Mode 1 (Sleep Mode)	In EM1, the CPU is sleeping and the power consumption is only 80 µA/MHz. All peripherals, including DMA, PRS and memory system, are still available.
	EM2 – Energy Mode 2 (Deep Sleep Mode)	In EM2 the high frequency oscillator is turned off, but with the 32.768 kHz oscillator running, selected low energy peripherals (LCD, RTC, LETIMER, PCNT, LEUART, I^2C, LESENSE, OPAMP, USB, WDOG and ACMP) are still available. This gives a high degree of autonomous operation with a current consumption as low as 1.1 µA with RTC enabled. Power-on Reset, Brown-out Detection and full RAM and CPU retention is also included.
	EM3 - Energy Mode 3 (Stop Mode)	In EM3, the low-frequency oscillator is disabled, but there is still full CPU and RAM retention, as well as Power-on Reset, Pin reset, EM4 wake-up and Brown-out Detection, with a consumption of only 0.8 µA. The low-power ACMP, asynchronous external interrupt, PCNT, and I^2C can wake-up the device. Even in this mode, the wake-up time is a few microseconds.
	EM4 – Energy Mode 4 (Shutoff Mode)	In EM4, the current is down to 20 nA and all chip functionality is turned off except the pin reset, GPIO pin wake-up, GPIO pin retention, Backup RTC (including retention RAM) and the Power-On Reset. All pins are put into their reset state.

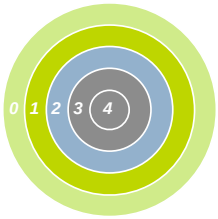
3.5 Product Overview

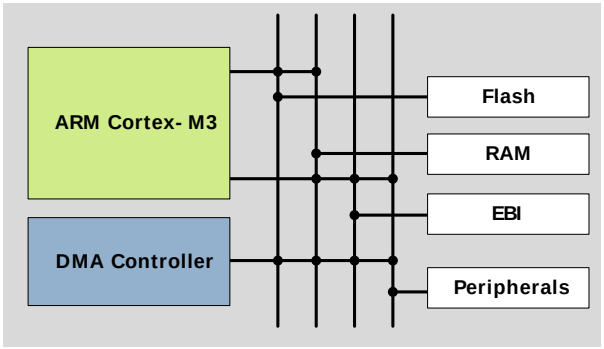
Table 3.2 (p. 8) shows a device overview of the EFM32GG Microcontroller Series, including peripheral functionality. For more information, the reader is referred to the device specific datasheets.

Table 3.2. EFM32GG Microcontroller Series

EFM32GG Part #	Flash	RAM	GPIO(pins)	USB	LCD	USART+UART	LEUART	I^2C	Timer(PWM)	LETIMER	RTC	PCNT	Watchdog	ADC(pins)	DAC(pins)	ACMP(pins)	AES	EBI	LESENSE	Op-Amps	Package
230F512	512	128	56	-	-	3	2	2	4 (12)	1	1	3	1	1 (8)	2 (2)	2 (16)	Y	-	Y	3	QFN64

5 Memory and Bus System





Quick Facts

What?

A low latency memory system, including low energy flash and RAM with data retention, makes extended use of low-power energy-modes possible.

Why?

RAM retention reduces the need for storing data in flash and enables frequent use of the ultra low energy modes EM2 and EM3 with as little as 0.8 µA current consumption.

How?

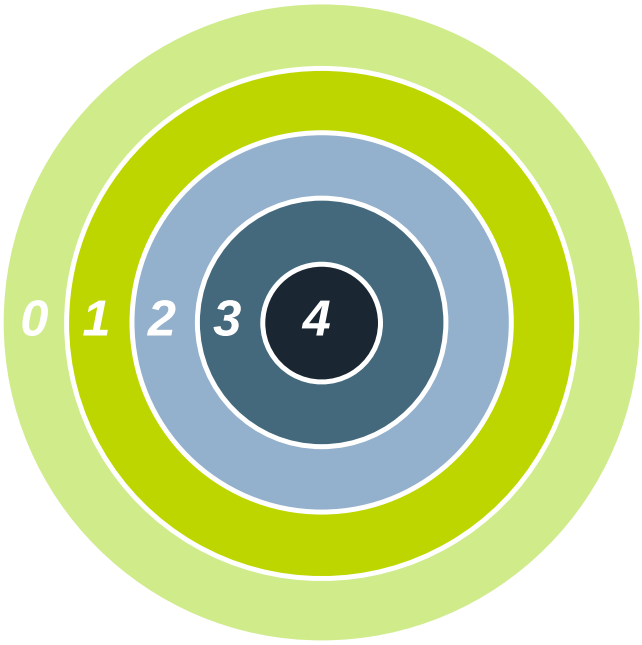
Low energy and non-volatile flash memory stores program and application data in all energy modes and can easily be reprogrammed in system. Low leakage RAM, with data retention in EM0 to EM3, removes the data restore time penalty, and the DMA ensures fast autonomous transfers with predictable response time.

5.1 Introduction

The EFM32GG contains an AMBA AHB Bus system allowing bus masters to access the memory mapped address space. A multilayer AHB bus matrix, using a Round-robin arbitration scheme, connects the master bus interfaces to the AHB slaves (Figure 5.1 (p. 16)). The bus matrix allows several AHB slaves to be accessed simultaneously. An AMBA APB interface is used for the peripherals, which are accessed through an AHB-to-APB bridge connected to the AHB bus matrix. The AHB bus masters are:

- **Cortex-M3 ICode:** Used for instruction fetches from Code memory (0x00000000 - 0x1FFFFFFF).
- **Cortex-M3 DCode:** Used for debug and data access to Code memory (0x00000000 - 0x1FFFFFFF).
- **Cortex-M3 System:** Used for instruction fetches, data and debug access to system space (0x20000000 - 0xDFFFFFFF).
- **DMA:** Can access EBI, SRAM, Flash and peripherals (0x00000000 - 0xDFFFFFFF).
- **USB DMA:** Can access EBI, SRAM and Flash (0x80000000 - 0xDFFFFFFF, 0x00000000 - 0x3FFFFFFF), and the AHB-peripherals: USB and AES.

10 EMU - Energy Management Unit



Quick Facts

What?

The EMU (Energy Management Unit) handles the different low energy modes in the EFM32GG microcontrollers.

Why?

The need for performance and peripheral functions varies over time in most applications. By efficiently scaling the available resources in real-time to match the demands of the application, the energy consumption can be kept at a minimum.

How?

With a broad selection of energy modes, a high number of low-energy peripherals available even in EM2, and short wake-up time (2 µs from EM2 and EM3), applications can dynamically minimize energy consumption during program execution.

10.1 Introduction

The Energy Management Unit (EMU) manages all the low energy modes (EM) in EFM32GG microcontrollers. Each energy mode manages if the CPU and the various peripherals are available. The energy modes range from EM0 to EM4, where EM0, also called run mode, enables the CPU and all peripherals. The lowest recoverable energy mode, EM3, disables the CPU and most peripherals while maintaining wake-up and RAM functionality. EM4 disables everything except the POR, pin reset and optionally Backup RTC, 512 byte data retention, GPIO state retention, and EM4 reset wakeup request.

The various energy modes differ in:

- Energy consumption
- CPU activity
- Reaction time
- Wake-up triggers
- Active peripherals
- Available clock sources

Low energy modes EM1 to EM4 are enabled through the application software. In EM1-EM3, a range of wake-up triggers return the microcontroller back to EM0. EM4 can only return to EM0 by power on reset, external pin reset, EM4 GPIO wakeup request, or Backup RTC interrupt.

The EMU can also be used to turn off the power to unused SRAM blocks.

10.2 Features

- Energy Mode control from software
- Flexible wakeup from low energy modes

Figure 14.2. EBI Non-multiplexed 8-bit Data, 8-bit Address Read Operation

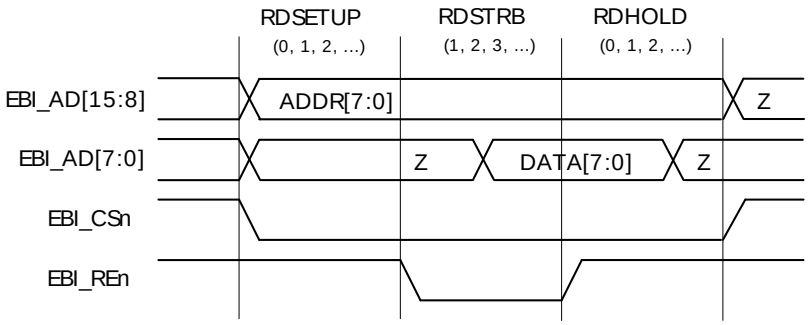
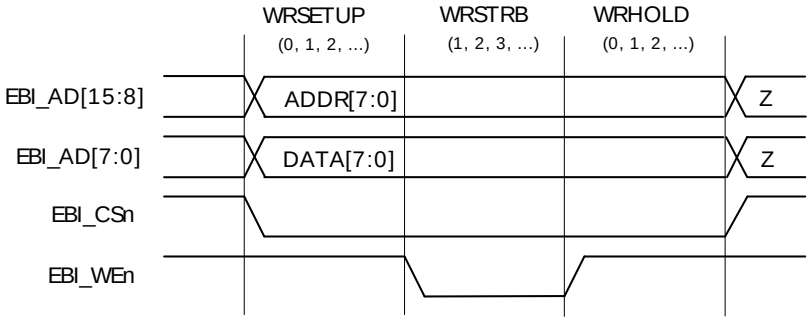


Figure 14.3. EBI Non-multiplexed 8-bit Data, 8-bit Address Write Operation



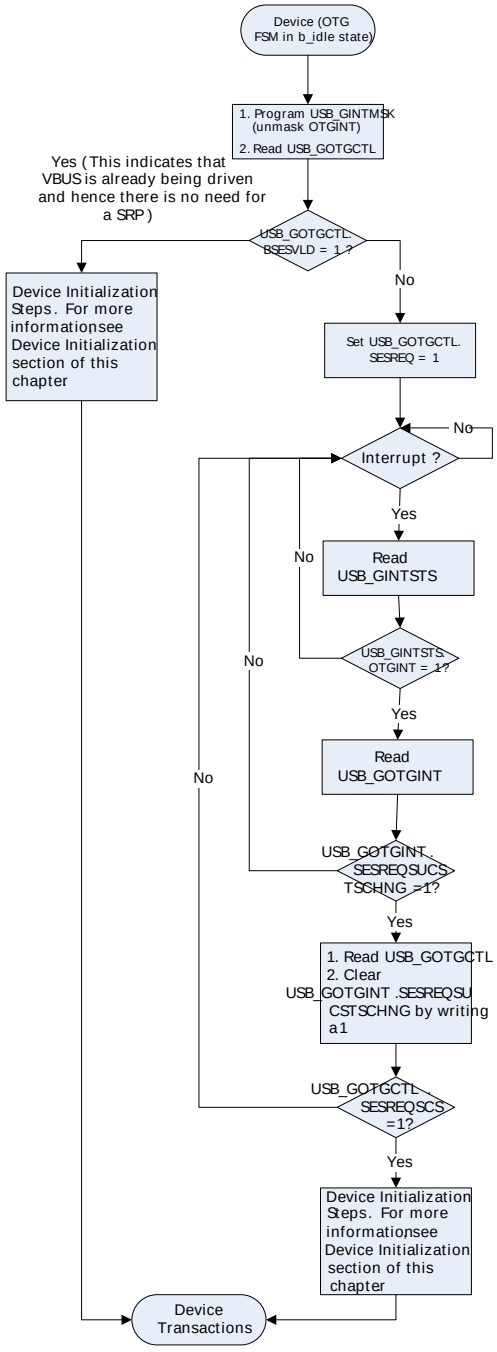
14.3.2 Multiplexed 16-bit Data, 16-bit Address Mode

In this mode, 16-bit address and 16-bit data is supported, but the utilization of an external latch is required. The 16-bit address and 16-bit data bits are multiplexed on the EBI_AD lines. An illustration of such a setup is shown in Figure 14.4 (p. 179) . This mode is set by programming the MODE field in the EBI_CTRL register to D16A16ALE.

Note

In this mode the 16-bit address is organized in 2-byte chunks at memory addresses aligned to 2-byte offsets. Consequently, the LSB of the 16-bit address will always be 0. In order to double the address space, the 16-bit address is internally shifted one bit to the right so that the LSB of the address driven into the EBI_AD bus, i.e. the EBI_AD[0]-bit, corresponds to the second least significant bit of the address, i.e. ADDR[1]. At the external device, the LSB of the address must be tied either low or high in order to create a full address.

Figure 15.33. SRP Initiation by the Core When Acting as a B-Device



Note

The programming flow illustrated in Figure 15.33 (p. 328) is similar to OTG revision 1.3. This is because the presence or absence of VBUS pulsing is transparent to the application.

15.4.6.2 OTG Revision 2.0 Host Negotiation Protocol

When the core is operating as A-device, the application must execute a GetStatus() operation to the B-device with a frequency of THOST_REQ_POLL to determine the state of the host request flag in the B-device. If the host request flag is set in B-device it must program the core to change its role within THOST_REQ_SUSP.

Figure 15.34 (p. 329) shows the programming steps that need to be performed by A-device's application (core as A-device) in order to change its role to device. In Figure 15.34 (p. 329) , the A-device performs a role change, becomes a B-device and then reverts back to host (A-device) mode of operation.

Bit	Name	Reset	Access	Description
31:25	*Reserved*	*To ensure compatibility with future devices, always write bits to 0. More information in Section 2.1 (p. 3)*		
24:21	FN	0x0	R	**Frame Number device only**
	This is the least significant 4 bits of the Frame number in which the packet is received on the USB.			
20:17	PKTSTS	0x0	R	**Packet Status (host or device)**
	Indicates the status of the received packet.			

Value	Mode	Description
1	GOUTNAK	Device mode: Global OUT NAK (triggers an interrupt).
2	PKTRCV	Host mode: IN data packet received. Device mode: OUT data packet received.
3	XFERCOMPL	Host mode: IN transfer completed (triggers an interrupt). Device mode: OUT transfer completed (triggers an interrupt).
4	SETUPCOMPL	Device mode: SETUP transaction completed (triggers an interrupt).
5	TGLERR	Host mode: Data toggle error (triggers an interrupt).
6	SETUPRCV	Device mode: SETUP data packet received.
7	CHLT	Host mode: Channel halted (triggers an interrupt).

16:15	DPID	0x0	R	**Data PID (host or device)**
	Host mode: Indicates the Data PID of the received packet. Device mode: Indicates the Data PID of the received OUT data packet.			

Value	Mode	Description
0	DATA0	DATA0 PID.
1	DATA1	DATA1 PID.
2	DATA2	DATA2 PID.
3	MDATA	MDATA PID.

14:4	BCNT	0x000	R	**Byte Count (host or device)**
	Host mode: Indicates the byte count of the received IN data packet. Device mode: Indicates the byte count of the received data packet.			
3:0	CHEPNUM	0x0	R	**Channel Number host only / Endpoint Number device only**
	Host mode: Indicates the channel number to which the current received packet belongs. Device mode: Indicates the endpoint number to which the current received packet belongs.			

15.6.16 USB_GRXSTSP - Receive Status Read and Pop Register

A read to the Receive Status Read and Pop register returns the contents of the top of the Receive FIFO and pops the top data entry out of the RxFIFO. The receive status contents must be interpreted differently in Host and Device modes. The core ignores the receive status pop/read when the receive FIFO is empty and returns a value of 0x00000000. The application must only pop the Receive Status FIFO when the Receive FIFO Non-Empty bit of the Core Interrupt register (USB_GINTSTS.RXFLVL) is asserted.

Offset	Bit Position																													
0x3C020	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reset								0x0				0x0				0x0		0x000											0x0			
Access								R				R				R		R											R			
Name								FN				PKTSTS				DPID		BCNT											CHEPNUM			

Bit	Name	Reset	Access	Description
31:25	*Reserved*	*To ensure compatibility with future devices, always write bits to 0. More information in Section 2.1 (p. 3)*		

Bit	Name	Reset	Access	Description
	The core sets this bit when a device connection is detected to trigger an interrupt to the application using the Host Port Interrupt bit of the Core Interrupt register (USB_GINTSTS.PRTINT). This bit can be set only by the core and the application should write 1 to clear it. The application must write a 1 to this bit to clear the interrupt.			
0	PRTCONNSTS	0	R	**Port Connect Status**
	When this bit is 1 a device is attached to the port.			

15.6.35 USB_HCx_CHAR - Host Channel x Characteristics Register

Offset	Bit Position																													
0x3C500	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reset	0	0	0	0x00							0x0		0x0		0		0	0x0				0x000										
Access	RW1H	RW1H	RW	RW							RW		RW		RW		RW	RW				RW										
Name	CHENA	CHDIS	ODDFRM	DEVADDR							MC		EPTYPE		LSPDDEV		EPDIR	EPNUM				MPS										

Bit	Name	Reset	Access	Description
31	CHENA	0	RW1H	**Channel Enable**
	This field is set by the application and cleared by the core. The state of this bit reflects the channel enable status.			
30	CHDIS	0	RW1H	**Channel Disable**
	The application sets this bit to stop transmitting/receiving data on a channel, even before the transfer for that channel is complete. The application must wait for the Channel Disabled interrupt before treating the channel as disabled.			
29	ODDFRM	0	RW	**Odd Frame**
	This field is set (reset) by the application to indicate that the OTG host must perform a transfer in an odd frame. This field is applicable for only periodic (isochronous and interrupt) transactions.			
28:22	DEVADDR	0x00	RW	**Device Address**
	This field selects the specific device serving as the data source or sink.			
21:20	MC	0x0	RW	**Multi Count**
	For periodic transfers this field indicates to the host the number of transactions that must be executed per frame for this periodic endpoint. For non-periodic transfers, this field is used only in DMA mode, and specifies the number packets to be fetched for this channel before the internal DMA engine changes arbitration.			
19:18	EPTYPE	0x0	RW	**Endpoint Type**
	Indicates the transfer type selected.			

Value	Mode	Description
0	CONTROL	Control endpoint.
1	ISO	Isochronous endpoint.
2	BULK	Bulk endpoint.
3	INT	Interrupt endpoint.

17	LSPDDEV	0	RW	**Low-Speed Device**
	This field is set by the application to indicate that this channel is communicating to a low-speed device.			
16	*Reserved*	*To ensure compatibility with future devices, always write bits to 0. More information in Section 2.1 (p. 3)*		
15	EPDIR	0	RW	**Endpoint Direction**
	Indicates whether the transaction is IN or OUT.			

Value	Mode	Description
0	OUT	Direction is OUT.

Bit	Name	Reset	Access	Description
31:16	*Reserved*	*To ensure compatibility with future devices, always write bits to 0. More information in Section 2.1 (p. 3)*		
15:0	SPCAVAIL	0x0200	R	**TxFIFO Space Available**
	Indicates the amount of free space available in the Endpoint TxFIFO. Values are in terms of 32-bit words.			

15.6.60 USB_DOEP0CTL - Device OUT Endpoint 0 Control Register

The application uses this register to control the behavior of each logical endpoint other than endpoint 0.

Offset	Bit Position																													
0x3CB00	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reset	0	0			0	0					0	0	0x0		0		1														0x0	
Access	RW1H	R			W1	W1					RW1H	RW	R		R		R														R	
Name	EPENA	EPDIS			SNAK	CNAK					STALL	SNP	EPTYPE		NAKSTS		USBACTEP														MPS	

Bit	Name	Reset	Access	Description
31	EPENA	0	RW1H	**Endpoint Enable**
	In DMA mode this bit indicates that the application has allocated the memory to start receiving data from the USB. The core clears this bit before setting any of the following interrupts on this endpoint: SETUP Phase Done, Endpoint Disabled, Transfer Completed. In DMA mode, this bit must be set for the core to transfer SETUP data packets into memory.			
30	EPDIS	0	R	**Endpoint Disable**
	This bit is always 0. The application cannot disable control OUT endpoint 0.			
29:28	*Reserved*	*To ensure compatibility with future devices, always write bits to 0. More information in Section 2.1 (p. 3)*		
27	SNAK	0	W1	**Set NAK**
	A write to this bit sets the NAK bit for the endpoint. Using this bit, the application can control the transmission of NAK handshakes on an endpoint. The core can also set bit on a Transfer Completed interrupt, or after a SETUP is received on the endpoint.			
26	CNAK	0	W1	**Clear NAK**
	A write to this bit clears the NAK bit for the endpoint.			
25:22	*Reserved*	*To ensure compatibility with future devices, always write bits to 0. More information in Section 2.1 (p. 3)*		
21	STALL	0	RW1H	**Handshake**
	The application can only set this bit, and the core clears it, when a SETUP token is received for this endpoint. If a NAK bit or Global OUT NAK is set along with this bit, the STALL bit takes priority. Irrespective of this bit's setting, the core always responds to SETUP data packets with an ACK handshake.			
20	SNP	0	RW	**Snoop Mode**
	This bit configures the endpoint to Snoop mode. In Snoop mode, the core does not check the correctness of OUT packets before transferring them to application memory.			
19:18	EPTYPE	0x0	R	**Endpoint Type**
	Hardcoded to 0. Endpoint 0 is always a control endpoint.			
17	NAKSTS	0	R	**NAK Status**
	When this bit is 0 the core is transmitting non-NAK handshakes based on the FIFO status. When this bit is 1 the core is transmitting NAK handshakes on this endpoint. When either the application or the core sets this bit, the core stops receiving data, even if there is space in the RxFIFO to accommodate the incoming packet. Irrespective of this bit's setting, the core always responds to SETUP data packets with an ACK handshake.			
16	*Reserved*	*To ensure compatibility with future devices, always write bits to 0. More information in Section 2.1 (p. 3)*		
15	USBACTEP	1	R	**USB Active Endpoint**
	This bit is always 1, indicating that a control endpoint 0 is always active in all configurations and interfaces.			

Offset	Bit Position																														
0x49000	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reset	0xXXXXXXXX																															
Access	RW																															
Name	FIFO12D																															

Bit	Name	Reset	Access	Description
31:0	FIFO12D	0xXXXXXXXX	RW	**Host Channel 12 FIFO**
	FIFO 12 push/pop region. Used in slave mode.			

15.6.82 USB_FIFO13Dx - Host Channel 13 FIFO

This register, available in Host mode, is used to read or write the FIFO space for channel 13, in a given direction. If a host channel is of type IN, the FIFO can only be read on the channel. Similarly, if a host channel is of type OUT, the FIFO can only be written on the channel.

Offset	Bit Position																														
0x4A000	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reset	0xXXXXXXXX																															
Access	RW																															
Name	FIFO13D																															

Bit	Name	Reset	Access	Description
31:0	FIFO13D	0xXXXXXXXX	RW	**Host Channel 13 FIFO**
	FIFO 13 push/pop region. Used in slave mode.			

When enabling the I^2C, the ABORT command or the Bus Idle Timeout feature must be applied prior to use even if the BUSY flag is not set.

16.3.3 Safely Disabling and Changing Slave Configuration

The I^2C slave is partially asynchronous, and some precautions are necessary to always ensure a safe slave disable or slave configuration change. These measures should be taken, if (while the slave is enabled) the user cannot guarantee that an address match will not occur at the exact time of slave disable or slave configuration change.

Worst case consequences for an address match while disabling slave or changing configuration is that the slave may end up in an undefined state. To reset the slave back to a known state, the EN bit in I2Cn_CTRL must be reset. This should be done regardless of whether the slave is going to be re-enabled or not.

16.3.4 Clock Generation

The SCL signal generated by the I^2C master determines the maximum transmission rate on the bus. The clock is generated as a division of the peripheral clock, and is given by Equation 16.2 (p. 420) :

I^2C Maximum Transmission Rate

$$f_{SCL} = 1/(T_{low} + T_{high}), \quad (16.2)$$

where

T_{low} and T_{high} is the low and high periods of the clock signal respectively, given below. When the clock is not streched, the low and high periods of the clock signal are:

I^2C High and Low Cycles Equations

$$T_{high} = (N_{high} \times (CLKDIV + 1))/f_{HFPERCLK},$$
$$T_{low} = (N_{low} \times (CLKDIV + 1))/f_{HFPERCLK}. \quad (16.3)$$

Equation 16.3 (p. 420) and Equation 16.2 (p. 420) does not apply for low clock division factors (0, 1 and 2) because of synchronization. For these clock division factors, the formulas for computing high and low periods of the clock signal are given in Table 16.2 (p. 420) .

Table 16.2. I^2C High and Low Periods for Low CLKDIV

CLKDIV	Standard (4:4)		Asymmetric (6:3)		Fast (11:6)	
	T_{low}	T_{high}	T_{low}	T_{high}	T_{low}	T_{high}
0	$7/f_{HFPERCLK}$	$7/f_{HFPERCLK}$	$9/f_{HFPERCLK}$	$6/f_{HFPERCLK}$	$14/f_{HFPERCLK}$	$9/f_{HFPERCLK}$
1	$10/f_{HFPERCLK}$	$10/f_{HFPERCLK}$	$14/f_{HFPERCLK}$	$8/f_{HFPERCLK}$	$24/f_{HFPERCLK}$	$14/f_{HFPERCLK}$
2	$15/f_{HFPERCLK}$	$15/f_{HFPERCLK}$	$21/f_{HFPERCLK}$	$12/f_{HFPERCLK}$	$36/f_{HFPERCLK}$	$21/f_{HFPERCLK}$

The values of N_{low} and N_{high} and thus the ratio between the high and low parts of the clock signal is controlled by CLHR in the I2Cn_CTRL register. The available modes are summarized in Table 16.3 (p. 421) along with the highest I^2C-bus frequencies in the given modes that can be achieved without violating the timing specifications of the I^2C-bus. The maximum data hold time is dependent on the DIV and is given by:

Maximum Data Hold Time

$$t_{HD,DAT\text{-}max} = (4+DIV)/f_{HFPERCLK}. \quad (16.4)$$

Note

DIV must be set to 1 or higher during slave mode operation.

Bit	Name	Reset	Access	Description

When set, the bus automatically goes idle on a bus idle timeout, allowing new transfers to be initiated.

Value	Description
0	A bus idle timeout has no effect on the bus state.
1	A bus idle timeout tells the I^2C module that the bus is idle, allowing new transfers to be initiated.

14 Reserved — To ensure compatibility with future devices, always write bits to 0. More information in Section 2.1 (p. 3)

13:12 BITO 0x0 RW **Bus Idle Timeout**

Use to generate a timeout when SCL has been high for a given amount time between a START and STOP condition. When in a bus transaction, i.e. the BUSY flag is set, a timer is started whenever SCL goes high. When the timer reaches the value defined by BITO, it sets the BITO interrupt flag. The BITO interrupt flag will then be set periodically as long as SCL remains high. The bus idle timeout is active as long as BUSY is set. It is thus stopped automatically on a timeout if GIBITO is set. It is also stopped a STOP condition is detected and when the ABORT command is issued. The timeout is activated whenever the bus goes BUSY, i.e. a START condition is detected.

Value	Mode	Description
0	OFF	Timeout disabled
1	40PCC	Timeout after 40 prescaled clock cycles. In standard mode at 100 kHz, this results in a 50us timeout.
2	80PCC	Timeout after 80 prescaled clock cycles. In standard mode at 100 kHz, this results in a 100us timeout.
3	160PCC	Timeout after 160 prescaled clock cycles. In standard mode at 100 kHz, this results in a 200us timeout.

11:10 Reserved — To ensure compatibility with future devices, always write bits to 0. More information in Section 2.1 (p. 3)

9:8 CLHR 0x0 RW **Clock Low High Ratio**

Determines the ratio between the low and high parts of the clock signal generated on SCL as master.

Value	Mode	Description
0	STANDARD	The ratio between low period and high period counters (N_{low}:N_{high}) is 4:4
1	ASYMMETRIC	The ratio between low period and high period counters (N_{low}:N_{high}) is 6:3
2	FAST	The ratio between low period and high period counters (N_{low}:N_{high}) is 11:6

7 Reserved — To ensure compatibility with future devices, always write bits to 0. More information in Section 2.1 (p. 3)

6 GCAMEN 0 RW **General Call Address Match Enable**

Set to enable address match on general call in addition to the programmed slave address.

Value	Description
0	General call address will be NACK'ed if it is not included by the slave address and address mask.
1	When a general call address is received, a software response is required.

5 ARBDIS 0 RW **Arbitration Disable**

A master or slave will not release the bus upon losing arbitration.

Value	Description
0	When a device loses arbitration, the ARB interrupt flag is set and the bus is released.
1	When a device loses arbitration, the ARB interrupt flag is set, but communication proceeds.

4 AUTOSN 0 RW **Automatic STOP on NACK**

Write to 1 to make a master transmitter send a STOP when a NACK is received from a slave.

Value	Description
0	Stop is not automatically sent if a NACK is received from a slave.
1	The master automatically sends a STOP if a NACK is received from a slave.

3 AUTOSE 0 RW **Automatic STOP when Empty**

Write to 1 to make a master transmitter send a STOP when no more data is available for transmission.

Value	Description
0	A stop must be sent manually when no more data is to be transmitted.
1	The master automatically sends a STOP when no more data is available for transmission.

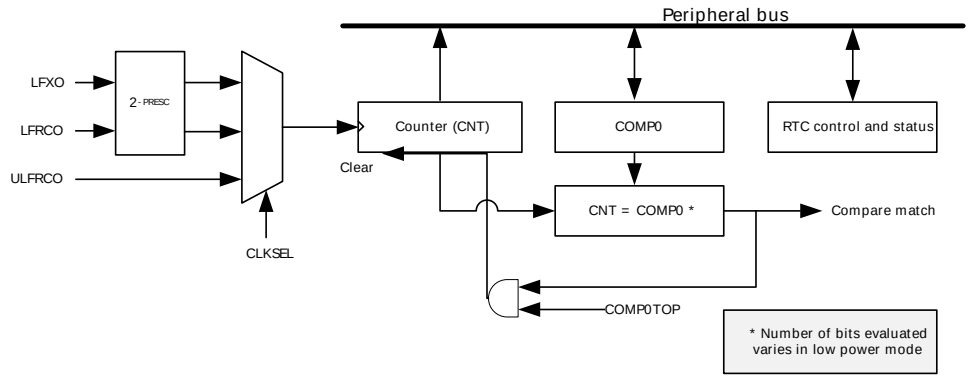
2 AUTOACK 0 RW **Automatic Acknowledge**

Set to enable automatic acknowledges.

22.3 Functional Description

The Backup RTC is a 32-bit counter with one compare channel. The Backup RTC resides in a power domain which can be configured to always be on, in EM0 through EM4. This domain also has the possibility to be powered by a backup battery. For further details on the backup power domain, refer to Section 10.3.4 (p. 111) . Available in all energy modes, the Backup RTC is ideal for applications where keeping track of time in combination with extremely low energy consumption is essential. An overview of the backup RTC is shown in Figure 22.1 (p. 571) .

Figure 22.1. BURTC Overview



22.3.1 Counter

The Backup RTC is enabled by configuring MODE in the BURTC_CTRL register. This configuration of MODE determines in which energy modes the backup RTC is operational. It will always be operational in EM0-EM2, and optionally in EM3 and EM4. The Backup RTC is available when the system is in backup mode if MODE is set to EM4EN. The counter is cleared by setting RSTEN in the control register. A system reset will not clear the counter. The counter value can be read through the CNT register.

22.3.2 Clock source

The Backup RTC is clocked by LFXO, LFRCO, or ULFRCO, depending on the configuration of CLKSEL in BURTC_CTRL. The PRESC bit-field in BURTC_CTRL controls the clock prescaling factor. Prescaler is only available for LFXO and LFRCO. When using the ULFRCO as clock source, only two frequency options are available; 2kHz and 1kHz. The 2kHz clock is selected when PRESC in BURTC_CTRL is set to DIV1, and the 1kHz clock is selected when PRESC is set to any other value. Available frequencies when using LFXO or LFRCO are given in Equation 22.1 (p. 571) . CLKSEL should not be changed while the backup RTC is running.

BURTC Frequency Equation

$$f_{BURTC} = 32768/2^{PRESC}\ \text{Hz, PRESC} = 0..7 \qquad (22.1)$$

When the LFXO or LFRCO is enabled, the Backup RTC will not use the clock until the timeout defined in the CMU has run out, i.e. the LFXORDY/LFRCORDY flag in CMU_STATUS is set. When an oscillator first has been enabled and is used by the Backup RTC, the Backup RTC will keep the selected clock source enabled, independent of both energy mode and CMU settings.

22.3.3 Compare channel

The backup RTC has one compare channel. The compare value is set by writing to the COMP0 register. When the value of CNT equals the value of COMP0, the COMP0 interrupt flag is set. If COMP0TOP in CTRL is set, the counter will wrap around when reaching the value in the compare register, COMP.

25.3.4 Sensor interaction

Many sensor types require some type of excitation in order to work. LESENSE can generate a variety of sensor stimuli, both on the same pin as the measurement is to be made on, and on alternative pins.

By default, excitation is performed on the pin associated with the channel, i.e. excitation and sensor measurement is performed on the same pin. The mode of the pin during the excitation phase is configured in EXMODE in CHx_INTERACT. The available modes during the excite phase are:

- DISABLED: The pin is disabled.
- HIGH: The pin is driven high.
- LOW: The pin is driven low.
- DACOUT: The pin is connected to the output of a DAC channel.

Note

Excitation with DAC output is only available on channels 0, 1, 2, and 3 (DAC0_CH0) and channels 12, 13, 14, and 15 (DAC0_CH1).

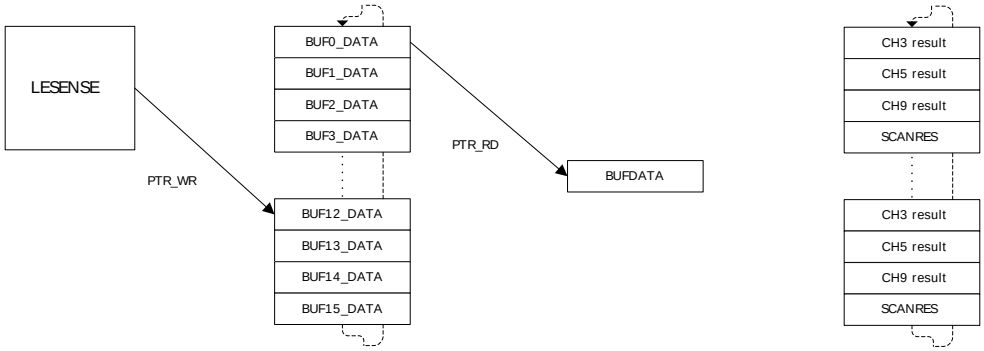
If the DAC is in opamp-mode, setting EXMODE to DACOUT will result in excitation with output from the opamp.

LESENSE is able to perform sensor excitation on another pin than the one to be measured. When ALTEX in CHx_INTERACT is set, the excitation will occur on the alternative excite pin associated with the given channel. All LESENSE channels mapped to ACMP0 have their alternative channel mapped to the corresponding channel on ACMP1, and vice versa. Alternatively, the alternative excite pins can be routed to the LES_ALTEX pins. Mapping of the alternative excite pins is configured in ALTEXMAP in CTRL. Table 25.2 (p. 627) summarizes the mapping of excitation pins for different configurations.

Table 25.2. LESENSE excitation pin mapping

LESENSE channel	ALTEX = 0	ALTEX = 1	
		ALTEXMAP = ACMP	ALTEXMAP = ALTEX
0	ACMP0_CH0	ACMP1_CH0	LES_ALTEX0
1	ACMP0_CH1	ACMP1_CH1	LES_ALTEX1
2	ACMP0_CH2	ACMP1_CH2	LES_ALTEX2
3	ACMP0_CH3	ACMP1_CH3	LES_ALTEX3
4	ACMP0_CH4	ACMP1_CH4	LES_ALTEX4
5	ACMP0_CH5	ACMP1_CH5	LES_ALTEX5
6	ACMP0_CH6	ACMP1_CH6	LES_ALTEX6
7	ACMP0_CH7	ACMP1_CH7	LES_ALTEX7
8	ACMP1_CH0	ACMP0_CH0	LES_ALTEX0
9	ACMP1_CH1	ACMP0_CH1	LES_ALTEX1
10	ACMP1_CH2	ACMP0_CH2	LES_ALTEX2
11	ACMP1_CH3	ACMP0_CH3	LES_ALTEX3
12	ACMP1_CH4	ACMP0_CH4	LES_ALTEX4
13	ACMP1_CH5	ACMP0_CH5	LES_ALTEX5
14	ACMP1_CH6	ACMP0_CH6	LES_ALTEX6
15	ACMP1_CH7	ACMP0_CH7	LES_ALTEX7

Figure 25.9. Circular result buffer



The right hand side of Figure 25.9 (p. 633) illustrates how the result buffer would be filled when channels 3,5, and 9 are enabled and have STRSAMPLE in CHx_EVAL set, in addition to STRSCANRES in CTRL. The measurement result from the three channels will be sequentially written during the scan, while SCANRES is written to the result buffer upon scan completion.

25.3.8 DAC interface

LESENSE is able to drive the DAC for generation of accurate reference voltages. DAC channels 0 and 1 are individually configured in the PERCTRL register. The conversion mode can be set to either continuous, sample/hold or sample/off. For further details about these modes, refer to Section 29.3.1 (p. 713) . Both DAC channels are refreshed prior to each sensor measurement, as depicted in Figure 25.3 (p. 626) . The conversion data is either taken from the data registers in the EFM32GG DAC interface (DAC0_CH0DATA and DAC0_CH1DATA) or from the ACMPTHRES bit-field in the CHx_INTERACT register for the active LESENSE channel. DAC data used is configured in DACCHxDATA in PERCTRL.

The DAC interface runs on AUXHFRCO and will enable this when it is needed. The DACPRESC bit-field in PERCTRL is used to prescale the AUXHFRCO to achieve wanted clock frequency for the LESENSE DAC interface. The frequency should not exceed 500kHz, i.e. DACPRESC has to be set to at least 1. The prescaler may also be used to tune how long the DAC should drive its outputs in sample/off mode.

Bias configuration, calibration and reference selection is done in the EFM32GG DAC module and LESENSE will not override these configurations. If a bandgap reference is selected for the DAC, the DACREF bit in PERCTRL should be set to BANDGAP.

LESENSE has the possibility to control switches that connect the DAC outputs to the pins associated with ACMP0_CH0-3 and ACMP1_CH12-15. This makes LESENSE able to excite sensors with output from the DAC channels.

The DAC may be chosen as reference to the analog comparators for accurate reference generation. If the DAC is configured in continuous or sample/hold mode this does not require any external components. If sample/off mode is used, an external capacitor is needed to keep the voltage in between samples. To connect the input from the DAC to the ACMP to this external capacitor, connect the capacitor to the DAC pin for the given channel and set OPAxSHORT in DAC_OPACTRL.

Note

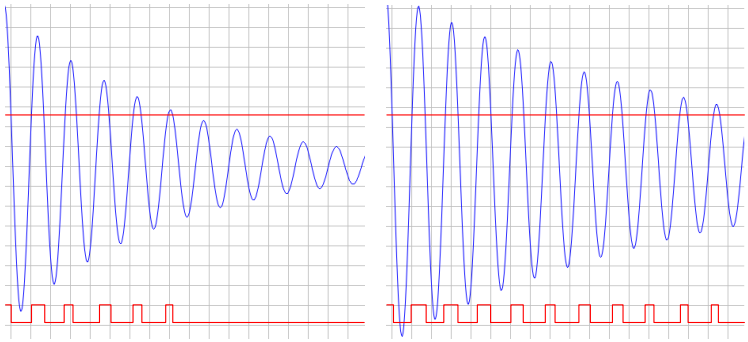
The DAC mode should not be altered while DACACTIVE in STATUS is set

25.3.9 ACMP interface

The ACMPs are used to measure the sensors, and have to be configured according to the application in order for LESENSE to work properly. Depending on the configuration in the ACMP0MODE and

exceeds a certain level. These pulses are counted using an asynchronous counter and compared with the threshold in COMPTHRES in the CHx_EVAL register. If the number of pulses exceeds the threshold level, the sensor is said to be active, otherwise it is inactive. Figure 25.12 (p. 636) illustrates how the output pulses from the ACMP correspond to damping of the oscillations. The results from sensor evaluation can automatically be fed into the decoder in order to keep track of rotations.

Figure 25.12. LC sensor oscillations



The following steps show how to configure LESENSE to scan through the four LC sensors 100 times per second.

1. Assuming $LFACLK_{LESENSE}$ is 32kHz, set PCPRESC to 3 and PCTOP to 39 in CTRL. This will make the LESENSE scan frequency 100Hz.
2. Enable the DAC and configure it to produce a voltage of Vdd/2.
3. Enable channels 0 through 3 in CHEN. Set IDLECONF for the active channels to DACOUT. The channel pins should be set to Vdd/2 in the idle phase to damp the oscillations.
4. Configure the ACMP to use scaled Vdd as negative input, refer to ACMP chapter for details.
5. Enable and configure PCNT and asynchronous PRS.
6. Configure the GPIOs used as PUSHPULL.
7. Configure the following bit fields in CHx_CONF, for channels 0 through 3:
   a. Set EXCLK to AUXHFRCO. AUXHFRCO is needed to achieve short excitation time.
   b. Set EXTIME to an appropriate value. Excitation will last for EXTIME/AUXHFRCO seconds (prescalar value in AUXPRESC in TIMCTRL is 0).
   c. Set EXMODE to LOW. The LC sensors are excited by pulling the excitation pin low.
   d. Set SAMPLE to COUNTER and COMP to LESS. Status of each sensor is evaluated based on the number of pulses generated by the ACMP. If they are less than the threshold value, the sensor is said to be active.
   e. Set SAMPLEDLY to an appropriate value, each sensor will be measured for SAMPLEDLY/ $LFACLK_{LESENSE}$ seconds.
8. Set CTRTHRESHOLD to an appropriate value. If the sensor is active, the counter value after the measurement phase should be less than the threshold. If it inactive, the counter value should be greater than the threshold.
9. Start scan sequence by writing a 1 to START in CMD.

25.3.14.3 LESENSE decoder 1

The example below illustrates how the LESENSE module can be used for decoding using three sensors

1. Configure STx_TCONFA and STx_TCONFB as described in Table 25.4 (p. 638) .

Table 25.4. LESENSE decoder configuration

Register	NEXTSTATE	COMP	MASK	CHAIN
ST0_TCONFA	8	0b1000	0b0111	1
ST0_TCONFB	2	0b0001	0b1000	-
ST1_TCONFA	6	0b0010	0b1000	0
ST1_TCONFB	6	0b0010	0b1000	-
ST2_TCONFA	8	0b1000	0b0111	1
ST2_TCONFB	4	0b0011	0b1000	-
ST3_TCONFA	0	0b0000	0b1000	0
ST3_TCONFB	0	0b0000	0b1000	-
ST4_TCONFA	8	0b1000	0b0111	1
ST4_TCONFB	6	0b0010	0b1000	-
ST5_TCONFA	2	0b0001	0b1000	0
ST5_TCONFB	2	0b0001	0b1000	-
ST6_TCONFA	8	0b1000	0b0111	1
ST6_TCONFB	0	0b0000	0b1000	-
ST7_TCONFA	4	0b0011	0b1000	0
ST7_TCONFB	4	0b0011	0b1000	-

2. To initialize the decoder, run one scan, and read the present sensor status from SENSORSTATE. Then write the index of this state to DECSTATE.
3. Write to START in CMD to start scanning of sensors and decoding.

Bit	Name	Reset	Access	Description

Value	Mode	Description
2	LOW	CH10 output is low in idle phase

19:18 CH9 0x0 RW **Channel 9 idle phase configuration**

Value	Mode	Description
0	DISABLE	CH9 output is disabled in idle phase
1	HIGH	CH9 output is high in idle phase
2	LOW	CH9 output is low in idle phase

17:16 CH8 0x0 RW **Channel 8 idle phase configuration**

Value	Mode	Description
0	DISABLE	CH8 output is disabled in idle phase
1	HIGH	CH8 output is high in idle phase
2	LOW	CH8 output is low in idle phase

15:14 CH7 0x0 RW **Channel 7 idle phase configuration**

Value	Mode	Description
0	DISABLE	CH7 output is disabled in idle phase
1	HIGH	CH7 output is high in idle phase
2	LOW	CH7 output is low in idle phase

13:12 CH6 0x0 RW **Channel 6 idle phase configuration**

Value	Mode	Description
0	DISABLE	CH6 output is disabled in idle phase
1	HIGH	CH6 output is high in idle phase
2	LOW	CH6 output is low in idle phase

11:10 CH5 0x0 RW **Channel 5 idle phase configuration**

Value	Mode	Description
0	DISABLE	CH5 output is disabled in idle phase
1	HIGH	CH5 output is high in idle phase
2	LOW	CH5 output is low in idle phase

9:8 CH4 0x0 RW **Channel 4 idle phase configuration**

Value	Mode	Description
0	DISABLE	CH4 output is disabled in idle phase
1	HIGH	CH4 output is high in idle phase
2	LOW	CH4 output is low in idle phase

7:6 CH3 0x0 RW **Channel 3 idle phase configuration**

Value	Mode	Description
0	DISABLE	CH3 output is disabled in idle phase
1	HIGH	CH3 output is high in idle phase
2	LOW	CH3 output is low in idle phase
3	DACCH0	CH3 output is connected to DAC CH0 output in idle phase

5:4 CH2 0x0 RW **Channel 2 idle phase configuration**

Bit	Name	Reset	Access	Description

Value	Mode	Description
0	DISABLE	CH2 output is disabled in idle phase
1	HIGH	CH2 output is high in idle phase
2	LOW	CH2 output is low in idle phase
3	DACCH0	CH2 output is connected to DAC CH0 output in idle phase

3:2	CH1	0x0	RW	**Channel 1 idle phase configuration**

Value	Mode	Description
0	DISABLE	CH1 output is disabled in idle phase
1	HIGH	CH1 output is high in idle phase
2	LOW	CH1 output is low in idle phase
3	DACCH0	CH1 output is connected to DAC CH0 output in idle phase

1:0	CH0	0x0	RW	**Channel 0 idle phase configuration**

Value	Mode	Description
0	DISABLE	CH0 output is disabled in idle phase
1	HIGH	CH0 output is high in idle phase
2	LOW	CH0 output is low in idle phase
3	DACCH0	CH0 output is connected to DAC CH0 output in idle phase

25.5.16 LESENSE_ALTEXCONF - Alternative excite pin configuration (Async Reg)

For more information about Asynchronous Registers please see Section 5.3 (p. 20) .

Offset	Bit Position																													
0x03C	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reset									0	0	0	0	0	0	0	0	0x0		0x0		0x0		0x0		0x0		0x0		0x0	
Access									RW	RW	RW	RW	RW	RW	RW	RW	RW		RW		RW		RW		RW		RW		RW	
Name									AEX7	AEX6	AEX5	AEX4	AEX3	AEX2	AEX1	AEX0	IDLECONF7		IDLECONF6		IDLECONF5		IDLECONF4		IDLECONF3		IDLECONF2		IDLECONF1		IDLECONF0	

Bit	Name	Reset	Access	Description
31:24	*Reserved*	*To ensure compatibility with future devices, always write bits to 0. More information in Section 2.1 (p. 3)*		
23	AEX7	0	RW	**ALTEX7 always excite enable**
22	AEX6	0	RW	**ALTEX6 always excite enable**
21	AEX5	0	RW	**ALTEX5 always excite enable**
20	AEX4	0	RW	**ALTEX4 always excite enable**
19	AEX3	0	RW	**ALTEX3 always excite enable**
18	AEX2	0	RW	**ALTEX2 always excite enable**

Bit	Name	Reset	Access	Description

Value	Mode	Description
0	DISABLE	ALTEX0 output is disabled in idle phase
1	HIGH	ALTEX0 output is high in idle phase
2	LOW	ALTEX0 output is low in idle phase

25.5.17 LESENSE_IF - Interrupt Flag Register

Offset	Bit Position																															
0x040	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reset										0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Access										R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Name										CNTOF	BUFOF	BUFLEVEL	BUFDATAV	DECERR	DEC	SCANCOMPLETE	CH15	CH14	CH13	CH12	CH11	CH10	CH9	CH8	CH7	CH6	CH5	CH4	CH3	CH2	CH1	CH0

Bit	Name	Reset	Access	Description
31:23	*Reserved*	*To ensure compatibility with future devices, always write bits to 0. More information in Section 2.1 (p. 3)*		
22	CNTOF	0	R	
	Set when the LESENSE counter overflows.			
21	BUFOF	0	R	
	Set when the result buffer overflows			
20	BUFLEVEL	0	R	
	Set when the data buffer is full.			
19	BUFDATAV	0	R	
	Set when data is available in the result buffer.			
18	DECERR	0	R	
	Set when the decoder detects an error			
17	DEC	0	R	
	Set when the decoder has issued and interrupt request			
16	SCANCOMPLETE	0	R	
	Set when a scan sequence is completed			
15	CH15	0	R	
	Set when channel 15 triggers			
14	CH14	0	R	
	Set when channel 14 triggers			
13	CH13	0	R	
	Set when channel 13 triggers			
12	CH12	0	R	
	Set when channel 12 triggers			
11	CH11	0	R	
	Set when channel 11 triggers			
10	CH10	0	R	
	Set when channel 10 triggers			
9	CH9	0	R	
	Set when channel 9 triggers			

26.4 Register Map

The offset register address is relative to the registers base address.

Offset	Name	Type	Description
0x000	ACMPn_CTRL	RW	Control Register
0x004	ACMPn_INPUTSEL	RW	Input Selection Register
0x008	ACMPn_STATUS	R	Status Register
0x00C	ACMPn_IEN	RW	Interrupt Enable Register
0x010	ACMPn_IF	R	Interrupt Flag Register
0x014	ACMPn_IFS	W1	Interrupt Flag Set Register
0x018	ACMPn_IFC	W1	Interrupt Flag Clear Register
0x01C	ACMPn_ROUTE	RW	I/O Routing Register

26.5 Register Description

26.5.1 ACMPn_CTRL - Control Register

Offset	Bit Position																													
0x000	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reset	0	1			0x7										0	0						0x0				0x0			0	0	0	0
Access	RW	RW			RW										RW	RW						RW				RW			RW	RW	RW	RW
Name	FULLBIAS	HALFBIAS			BIASPROG										IFALL	IRISE						WARMTIME				HYSTSEL			GPIOINV	INACTVAL	MUXEN	EN

Bit	Name	Reset	Access	Description
31	FULLBIAS	0	RW	**Full Bias Current**
	Set this bit to 1 for full bias current in accordance with Table 26.1 (p. 671) .			
30	HALFBIAS	1	RW	**Half Bias Current**
	Set this bit to 1 to halve the bias current in accordance with Table 26.1 (p. 671) .			
29:28	*Reserved*	*To ensure compatibility with future devices, always write bits to 0. More information in Section 2.1 (p. 3)*		
27:24	BIASPROG	0x7	RW	**Bias Configuration**
	These bits control the bias current level in accordance with Table 26.1 (p. 671) .			
23:18	*Reserved*	*To ensure compatibility with future devices, always write bits to 0. More information in Section 2.1 (p. 3)*		
17	IFALL	0	RW	**Falling Edge Interrupt Sense**
	Set this bit to 1 to set the EDGE interrupt flag on falling edges of comparator output.			

Value	Mode	Description
0	DISABLED	Interrupt flag is not set on falling edges.
1	ENABLED	Interrupt flag is set on falling edges.

16	IRISE	0	RW	**Rising Edge Interrupt Sense**
	Set this bit to 1 to set the EDGE interrupt flag on rising edges of comparator output.			

Value	Mode	Description
0	DISABLED	Interrupt flag is not set on rising edges.
1	ENABLED	Interrupt flag is set on rising edges.

15:11	*Reserved*	*To ensure compatibility with future devices, always write bits to 0. More information in Section 2.1 (p. 3)*		
10:8	WARMTIME	0x0	RW	**Warm-up Time**
	Set analog comparator warm-up time.			