Reference Guide

Low-Energy Accelerator (LEA) Registers

ABSTRACT

This document describes the registers for the low-energy accelerator (LEA) module on MSP430™ microcontrollers (MCUs).

Table of Contents

Trademarks

MSP430™ is a trademark of Texas Instruments.
All trademarks are the property of their respective owners.

1 LEA Control and Configuration Registers

Table 1-2 lists the registers of the LEA module. Table 1-1 lists a common base address for the LEA register. Verify this address with the device-specific data sheet.

Note

Access to the LEA other than LEACNF0 registers is not possible when LEACNF0.LEASWRST=1.

Table 1-1. LEA Interface Base Address

Module	Base Address
LEA	0xD00 (see the device-specific data sheet)

Table 1-2. LEA Registers

Offset	Acronym	Register Name	Type	Access	Reset
00h	LEACAP	LEA Capability Register	Read only	32B	--
00h	LEACAPL	LEA Capability Register Low word	Read only	16B	
04h	LEACNF0	Configuration Register 0	Read/Write	32B	with PUC
04h	LEACNF0L	Configuration Register 0 low Word	Read/Write	16B	
06h	LEACNF0H	Configuration Register 0 high Word	Read/Write	16B	
08h	LEACNF1	Configuration Register 1	Read/Write	32B	with PUC
08h	LEACNF1L	Configuration Register 1 low Word	Read/Write	16B	
0Ah	LEACNF1H	Configuration Register 1 high Word	Read/Write	16B	
0Ch	LEACNF2	Configuration Register 2	Read/Write	32B	with PUC
0Ch	LEACNF2L	Stack Pointer	Read/Write	16B	
0Eh	LEACNF2H	Base Address Register	Read/Write	16B	
10h	LEAMB	Memory Bottom Register	Read/Write	32B	with PUC
10h	LEAMBL	Memory Bottom Register low Word	Read/Write	16B	
14h	LEAMT	Memory Top Register	Read/Write	32B	with PUC
14h	LEAMTL	Memory Top Register low Word	Read/Write	16B	
18h	LEACMA	Code Memory Access Register	Read/Write	32B	with PUC
18h	LEACMAL	Code Memory Access Reg. low Word	Read/Write	16B	
1Ah	LEACMAH	Code Memory Access Reg. high Word	Read/Write	16B	
1Ch	LEACMCTL	Code Memory Control Register	Read/Write	32B	with PUC
1Ch	LEACMCTLL	Code Memory Control Reg. low Word	Read/Write	16B	
1Eh	LEACMCTLH	Code Memory Control Reg. high Word	Read/Write	16B	
28h	LEACMDSTAT	LEA Command Status Register	Read only	32B	with PUC
28h	LEACMDSTATL	LEA Command Status Reg. low Word	Read only	16B	
2Ah	LEACMDSTATH	LEA Command Status Reg. high Word	Read only	16B	
2Ch	LEAS1STAT	LEA Source 1 Status Register	Read only	32B	with PUC
2Ch	LEAS1STATL	LEA Source 1 Status Reg. low Word	Read only	16B	
2Eh	LEAS1STATH	LEA Source 1 Status Reg. high Word	Read only	16B	
30h	LEAS0STAT	LEA Source 0 Status Register	Read only	32B	with PUC
30h	LEAS0STATL	LEA Source 0 Status Reg. low Word	Read only	16B	
32h	LEAS0STATL	LEA Source 0 Status Reg. high Word	Read only	16B	
34h	LEADSTSTAT	LEA Result Status Register	Read only	32B	with PUC
34h	LEADSTSTATL	LEA Result Status Register	Read only	16B	
36h	LEADSTSTATH	LEA Result Status Register	Read only	16B	
40h	LEAPMCTL	PM Control Register	Read/Write	32B	with PUC
40h	LEAPMCTLL	Control Register low Word	Read/Write	16B	

Table 1-2. LEA Registers (continued)

Offset	Acronym	Register Name	Type	Access	Reset
44h	LEAPMDST	PM Result Register	Read/Write	32B	with PUC
44h	LEAPMDSTL	Result Register low Word	Read/Write	16B	
48h	LEAPMDSTH	Result Register high Word	Read/Write	16B	
48h	LEAPMS1	PM Source 1 Register	Read/Write	32B	with PUC
48h	LEAPMS1L	Source 1 Register low Word	Read/Write	16B	
4Ah	LEAPMS1H	Source 1 Register high Word	Read/Write	16B	
4Ch	LEAPMS0	PM Source 0 Register	Read/Write	32B	with PUC
4Ch	LEAPMS0L	Source 0 Register low Word	Read/Write	16B	
4Eh	LEAPMS0H	Source 0 Register high Word	Read/Write	16B	
50h	LEAPMCB	PM Command Buffer Register	Read/Write	32B	with PUC
50h	LEAPMCBL	Command Buffer Register low Word	Read/Write	16B	
52h	LEAPMCBH	Command Buffer Register high Word	Read/Write	16B	
70h	LEAIFGSET	Interrupt Flag and Set Register	Read/Write	32B	with PUC
70h	LEAIFGSETL	Interrupt Flag and Set Reg. low Word	Read/Write	16B	
74h	LEAIE	Interrupt Enable Register	Read/Write	32B	with PUC
74h	LEAIEL	Interrupt Enable Register low Word	Read/Write	16B	
78h	LEAIFG	Interrupt Flag and Clear Register	Read/Write	32B	with PUC
78h	LEAIFGL	Interrupt Flag and Clear Reg. low Word	Read/Write	16B	
7Ch	LEAIV	Interrupt Vector Register	Read/Write	32B	with PUC
7Ch	LEAIVL	Interrupt Vector Register low Word	Read/Write	16B	

1.1 LEACAP, LEACAPL LEA Capability Register

With this register the Capabilities of the present LEA module are identified. Please check the actual datasheet of your device if more than the base capabilities are implemented.

Figure 1-1. LEACAP Register

31	30	29	28	27	26	25	24
Reserved							
r0	r0	r0	r0	r0	r0	r0	r0
23	22	21	20	19	18	17	16
Reserved							
r0	r0	r0	r0	r0	r0	r0	r0
15	14	13	12	11	10	9	8
Reserved							
r0	r0	r0	r0	r0	r0	r0	r0
7	6	5	4	3	2	1	0
Reserved				LEAMSIZ			
r0	r0	r0	r0	r0	r0	r0	r1

Table 1-3. LEACAP Register Field Descriptions

Bit	Field	Type	Reset	Description
31-0	Reserved	r	0	Reserved. Reads as 0.
3-0	LEAMSIZ	r	1	LEA Code Memory Size. This register identifies the size of available code RAM 0 : no code RAM 1 : 1kB Code RAM

1.2 LEACNF0, LEACNF0L, LEACNF0H LEA Configuration Register 0

This register controls the various command execution options and clock sources of LEA.

Figure 1-2. LEACNF0 Register

31	30	29	28	27	26	25	24
LEATISELx				Reserved		LEATEN	LEATRST
rw-0	rw-0	rw-0	rw-0	r0	r0	rw-0	rw-0
23	22	21	20	19	18	17	16
~MEMFLTS	~CFLTS		Reserved	Reserved		~FREES	~DONES
rw-0	rw-0	rw-0	r0	r0	r0	rw-0	rw-0
15	14	13	12	11	10	9	8
~MEMFLTE	Reserved		Reserved	LEAILB	LEAILPM	Reserved	LEALPR
rw-0	r0	rw-0	r0	rw-0	rw-0	r0	rw-0
7	6	5	4	3	2	1	0
Reserved			Reserved			~FLTHOLD	~SWRST
r0	r0	r0	r0	r0	r0	rw-0	rw-1

Table 1-4. LEACNF0 Register Field Descriptions

Bit	Field	Type	Reset	Description		
31-28	LEATISELx	rw	0h	LEA timer interval select. These bits select LEA timer interval. $t_{CLK} = 1/f_{CLK}$, $f_{CLK} = f_{MCLK}$		
				Value	**Selected Timeout Period**	**Selected Interval Period**
				0000	128 * t_{CLK} (16 µs at 8 MHz)	256 * t_{CLK} (32 µs at 8 MHz)
				0001	256 * t_{CLK} (32 µs at 8 MHz)	512 * t_{CLK} (64 µs at 8 MHz)
				0010	512 * t_{CLK} (64 µs at 8 MHz)	1024 * t_{CLK} (128 µs at 8 MHz)
				0011	1024 * t_{CLK} (128 µs at 8 MHz)	2048 * t_{CLK} (256 µs at 8 MHz)
				0100	2048 * t_{CLK} (256 µs at 8 MHz)	4096 * t_{CLK} (512 µs at 8 MHz)
				0101	4096 * t_{CLK} (512 µs at 8 MHz)	8192 * t_{CLK} (1 ms at 8 MHz)
				0110	8192 * t_{CLK} (1 ms at 8 MHz)	16384 * t_{CLK} (2 ms at 8 MHz)
				0111	16384 * t_{CLK} (2 ms at 8 MHz)	32768 * t_{CLK} (4 ms at 8 MHz)
				1000	32768 * t_{CLK} (4 ms at 8 MHz)	65536 * t_{CLK} (8 ms at 8 MHz)
				1001	65536 * t_{CLK} (8 ms at 8 MHz)	131072 * t_{CLK} (16 ms at 8 MHz)
				1010	131072 * t_{CLK} (16 ms at 8 MHz)	262144 * t_{CLK} (32 ms at 8 MHz)
				1011	524288 * t_{CLK} (65 ms at 8 MHz)	1048576 * t_{CLK} (131 ms at 8 MHz)
				1100	1048576 * t_{CLK} (131 ms at 8 MHz)	2097152 * t_{CLK} (262 ms at 8 MHz)
				1101	2097152* t_{CLK} (262 ms at 8 MHz)	4194304 * t_{CLK} (524 ms at 8 MHz)
				1110	4194304 * t_{CLK} (524 ms at 8 MHz)	8388608 * t_{CLK} (1.05 s at 8 MHz)
				1111	8388608 * t_{CLK} (1.05 s at 8 MHz)	16777216 * t_{CLK} (2.1 s at 8 MHz)
27-26	Reserved			Reserved. Reads as 0.		

Table 1-4. LEACNF0 Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
25	LEATEN	rw	0h	LEA timer enable; writing a one to this bit enables LEA timer operations.
24	LEATRST	rw	0h	LEA module timer reset. Setting this bit to one clears LEA module timer. This bit is self clearing and will always be read as zero.
23	LEAMEMFLTS	rw	0h	LEA memory fault indication and set flag. This bit indicates that a fault in the memory interface occurred. The exact fault reason may be identified by checking LEACNF1. LEAWRSTAT and LEACNF1.LEARDSTAT. This fault is also signaled to the SYSs-module as bus-error when enabled (~MEMFLTE=1). Only one fault condition is signaled until this bit is cleared. Leaving this bit set will not cause any further faults. This Fault may also be set by writing a one to it. Writing a zero has no effect. 0 = No memory fault occurred since this bit was cleared 1 = At least one memory fault since this bit was cleared
22	LEACFLTS	rw	0h	LEA command fault indication and set flag; This bits indicates that a command was invoked that is not implemented. This fault is also signaled to the SYS module as a “User-NMI” when enabled (~CFLTE=1). Only one fault condition is signaled until this bit is cleared. Leaving this bit set will not cause any further faults. This fault may also be set by writing a one to it. Writing a zero has no effect. 0 = No command fault occurred since this bit was cleared 1 = At least one command fault occurred since this bit was cleared
21	LEATIMFLTS	rw	0h	LEA timeout fault indication and set flag; This bits indicates that timer timeout occurred. This fault may also be set by writing a one to it. Writing a zero has no effect.
20-18	Reserved	r	0h	Reserved. Reads as 0.
17	LEAFREES	rw	0h	LEA free event indication and set flag. This bit indicated the free event for LEA. This bit can be set by writing a one to it. Writing a zero has no effect.
16	LEADONES	rw	0h	LEA done event indication and set flag. This bit indicated the done event for LEA. This bit can be set by writing a one to it. Writing a zero has no effect.
15	LEAMEMFLTE	rw	0h	Enable bit on memory faults. 0 = LEA memory faults are disabled 1 = LEA memory faults are enabled
14	LEACFLT	rw	0h	Enable bit on command faults 0 = LEACFLT is disabled 1 = LEACFLT is enabled
13	LEATIFLTE	rw	0h	LEA module timer fault enable 0: a LEA module timer timeout will not cause a fault indication 1: a LEA module timer timeout will cause a fault indication. Here LEA stops operation an enters “Ready-state”.
12	Reserved			Reserved. Reads as 0.
11	LEAILB	r	0h	LEA instruction loop buffer disable. Debugging function for LEA (leave it zero).
10	LEAILPM	rw	0h	This bit defines if a “Command done interrupt” shall be triggered in LPM mode 0 = Interrupt of LEA is suppressed in LPM mode until AM is entered then the LEA interrupt is triggered as well 1 = Interrupt of LEA is always triggered on completion of an LEA command
9	Reserved			Reserved. Reads as 0.
8	LEALPR	rw	0h	This bit defined if command execution shall be continued in LPM modes 0 = LEA command execution stops in deep low power modes 1 = LEA command execution continues in deep low power modes
7-5	Reserved	r	0h	Reserved. Reads as 0.

Table 1-4. LEACNF0 Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
1	LEAFTLHOLD	rw	0h	Hold on faults and NMIs for all pending LEA operations transfers. 0 = LEA transfers continue on faults/NMIs 1 = LEA transfers enter HOLD on faults/NMIs
0	LEASWRST	rw	1h	LEA module software restart. Setting this bit to one restarts the LEA module. As long this bit remains set to one the LEA is held in Restart. (The LEA accessible memory behaves as system RAM)

1.3 LEACNF1, LEACNF1L, LEACNF1H LEA Configuration Register 1

This register reflects the internal status of LEA.

Figure 1-3. LEACNF1 Register

31	30	29	28	27	26	25	24
LEAWRSTAT				LEARDSTAT			
r0	r-0	r-0	r-0	r0	r-0	r-0	r-0
23	22	21	20	19	18	17	16
~MEMFLTC	~CFLTC	~TIMFLTC	Reserved			~FREEC	~DONEC
rw-0	rw-0	rw-0	r0	r0	r0	rw-0	rw-0
15	14	13	12	11	10	9	8
LEAASST				LEAPWST			
r-0	r-0	r-0	r-0	r-0	r-0	r-0	r-0
7	6	5	4	3	2	1	0
LEAMODE				Reserved			LEABSY
r-0	r-0	r-0	r-0	r0	r0	r0	r-0

Table 1-5. LEACNF1 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-28	LEAWRSTAT	r	0h	Write Status. This bit field keeps the BUS write status lines from the last bus error condition.
27-24	LEARDSTAT	r	0h	Read Status. This bit field keeps the BUS read status lines from the last bus error condition.
23	LEAMEMFLTC	rw	0h	LEA memory fault indication and clear flag. This bit indicates that a fault in the memory interface occurred. The exact fault reason may be identified by checking LEAWRSTAT and LEARDSTAT. This fault is also signaled to the SYS-module as bus error when enabled (LEACNF0.LEAMEMFLTE=1). Only one fault condition is signaled until this bit is cleared. Leaving this bit set will not cause any further faults. This fault is cleared by writing a one to it. Writing a zero has no effect. 0 = No memory fault occurred since this bit was cleared 1 = At least one memory fault since this bit was cleared
22	LEAFLTC	rw	0h	LEA command fault indication and clear flag; This bits indicates that a command was invoked that is not implemented. This fault is also signaled to the SYS module as a “User-NMI” when enabled (LEACNF0.LEACFLTE=1). Only one fault condition is signaled until this bit is cleared. Leaving this bit set will not cause any further faults. This fault is cleared by writing a one to it. Writing a zero has no effect. 0 = No command fault occurred since this bit was cleared 1 = At least one command fault occurred since this bit was cleared
21	LEATIMFLTC	rw	0h	LEA timeout fault indication and clear flag; This bits indicates that a timer timeout occurred. This fault is cleared by writing a one to it. Writing a zero has no effect.
20-18	Reserved	r	0h	Reserved. Reads as 0.
17	LEAFREEC	rw	0h	LEA free event indication and clear flag. This bit indicated the free event for LEA. This bit is cleared by writing a one to it. Writing a zero has no effect.
16	LEADONEC	rw	0h	LEA done event indication and clear flag. This bit indicated the done event for LEA. This bit is cleared by writing a one to it. Writing a zero has no effect.
15-12	LEAASST	r	0h	These bits are used to store the internal state of the application specific processor (ASIP) inside the accelerator core. The specific meaning of those bit patterns is not shown in this document.

Table 1-5. LEACNF1 Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
11-8	LEAPWST	r	0h	These bits indicate the current power consumption as a relative value. The value zero indicated only static operation (usually clock less). This register might be read out for statistical power estimation of an application. These bits are also reflected in J-STATE when debugging
7-4	LEAMODE	r	0h	These bits indicate the operation mode of the LEA. 0000 = Off (implicit) 0001 = Ready 0010 = RunS (SUSPEND) 0011 = RunR (RESUME) 0100 = RunA (regular command operation) 0101 = Notify 0100 = Sleep 0111 = RunL other = reserved
3-1	Reserved	r	0h	Reserved. Reads as 0.
0	LEABSY	r	0h	This bit indicate if LEA is able to accept new Commands (SUSPEND is always accepted) 0: LEA is in Ready can accept new commands 1: LEA is busy right now and cannot accept any commands new:

1.4 LEACNF2, LEABA, LEASPTR, LEA Configuration Register 2

Figure 1-4. LEACNF2 Register

31	30	29	28	27	26	25	24
LEABADR							
r0	r0	r0	r0	r0	r0	r0	r0
23	22	21	20	19	18	17	16
LEABADR							
r0	rw-0	rw-0	rw-0	rw-0	rw-0	rw-0	rw-0
15	14	13	12	11	10	9	8
LEASPTR							
rw-0	rw-0	rw-0	rw-0	rw-0	rw-0	rw-0	rw-0
7	6	5	4	3	2	1	0
LEASPTR							
rw-0	rw-0	rw-0	rw-0	rw-0	rw-0	r0	r0

Table 1-6. LEACNF2 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-16	LEABADR	rw	0h	LEA data memory base address (upper 16 bit of an 32 bit value)
15-0	LEASPTR	rw	0h	LEA stack pointer value (byte units).

1.5 LEAMB, LEAMBL LEA Memory Bottom Address Boundary

This registers allows to set or verify the lower memory address boundary for LEA accessible memory. Whenever LEA accesses memory below this address an out of memory range interrupt is fired “LEAOOR”.

On devices that support programmable bottom address boundaries this register can be programmed to define the bottom address. See the data sheet of your device for the adress resolution.

On devices that support a fixed bottom address, this register can be read to find the bottom address.

Figure 1-5. LEAMB Register

31	30	29	28	27	26	25	24
Reserved							
r0	r0	r0	r0	r0	r0	r0	r0
23	22	21	20	19	18	17	16
Reserved							
r0	r0	r0	r0	r0	r0	r0	r0
15	14	13	12	11	10	9	8
LEAMB							
r	r	r	r	r	r	r	r
7	6	5	4	3	2	1	0
LEAMB							
r	r	r	r	r	r	r0	r0

Table 1-7. LEAMB Register Field Descriptions

Bit	Field	Type	Reset	Description
31-16	Reserved	r	0h	Reserved. Reads as 0.
15-0	LEAMB	r	0h	LEA memory bottom address boundary in byte address units

1.6 LEAMT, LEAMTL LEA Memory Top Address Boundary

This registers allows to set or verify the upper memory address boundary for LEA accessible memory. Whenever LEA accesses memory above this address an out of memory range interrupt is fired “LEAOOR”.

On devices that support programmable top address boundaries, this register can be programmed to define the top address. See the data sheet of your device for the address resolution.

On devices that support a fixed top address, this register can be read to find the top address.

Figure 1-6. LEAMT Register

31	30	29	28	27	26	25	24
Reserved							
r0	r0	r0	r0	r0	r0	r0	r0
23	22	21	20	19	18	17	16
Reserved							
r0	r0	r0	r0	r0	r0	r0	r0
15	14	13	12	11	10	9	8
LEAMT							
r	r	r	r	r	r	r	r
7	6	5	4	3	2	1	0
LEAMT							
r	r	r	r	r	r	r0	r0

Table 1-8. LEAMT Register Field Descriptions

Bit	Field	Type	Reset	Description
31-16	Reserved	r	0h	Reserved. Reads as 0.
15-0	LEAMT	r	0h	LEA memory top address boundary in byte address units

1.7 LEACMA, LEACMAL, LEACMAH LEA Code Memory Access Register

This registers contains the bit fields used to access LEA internal code memory. The LEA code memory is accessible only when the LEA is in Ready state and the code memory access enable LEACMAE is set to 1.

Figure 1-7. LEACMA Register

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

Table 1-9. LEACMA Register Field Descriptions

Bit	Field	Type	Reset	Description
31-0	LEACMDP	rw	0h	LEA code memory data port. Through this bit field the specific memory location is accessible LEACMAP is pointing at. Accesses may be read, write and read modify write. Data accesses are 32 bit wide for whole word and 16 bit wide for upper low word only. Accessing this location affects The LEAAP when LEACMCTL.LEADEC or LEACMCTL.LEAINC are set.

1.8 LEACMCTL, LEACMCTLL, LEACMCTLH LEA Code Memory Control Register

This registers controls the access flow to access LEA internal code memory.

Figure 1-8. LEACMCTL Register

31	30	29	28	27	26	25	24
LEAAP							
r0	r0	rw-0	rw-0	rw-0	rw-0	rw-0	rw-0
23	22	21	20	19	18	17	16
LEAAP							
rw-0	rw-0	rw-0	rw-0	rw-0	rw-0	rw-0	r0
15	14	13	12	11	10	9	8
Reserved							
r0	r0	r0	r0	r0	r0	r0	r0
7	6	5	4	3	2	1	0
Reserved		LEACROFFx		LEADEC	LEAINC	Reserved	LEAMAE
r0	r0	rw-1	rw-0	rw-0	rw-0	r0	rw-0

Table 1-10. LEACMCTL Register Field Descriptions

Bit	Field	Type	Reset	Description
31-16	LEACMAP	rw	0h	LEA code memory address port. This bit field points to the memory location that is accessible via LEACMDP
15-4	Reserved	r	0h	Reserved. Reads as 0.
5 -4	LEACROFF	rw	2h	Control bits only available if LEA code RAM is available. Otherwise reserved. 00b = Contents of LEA code RAM are retained in LPM3/LPM4. 01b = Turns off the LEA code RAM in LPM3/LPM4, re-activates it on wake-up. All data of the code RAM is lost after wakeup from LPM3/LPM4. See the device specific data sheet for presence and size of Code RAM. 10b = Turns off the code RAM entering LPM3/LPM4, the code RAM sector remains off after wake-up. All data of the code RAM is lost. See the device-specific data sheet for presence and size of Code RAM. 11b = Reserved
3	LEADEC	rw	0h	This bit when set causes the code memory address port field LEACMAP to decrement each time after an access to LEACMDP is performed. The decrement is by value two on 16B accesses on lower word of LEACMA. The decrement is by value two on 32B accesses on LEACMA. Note: A double decrement is performed when read modify write accesses are done on LEACMDP.
2	LEAINC	rw	0h	This bit when set causes the code memory address port field LEACMAP to increment each time after an access to LEACMDP is performed. The decrement is by value two on 16B accesses on lower word of LEACMA. The decrement is by value two on 32B accesses on LEACMA. Note: A double increment is performed when read modify write accesses are done on LEACMDP.
1	Reserved	r	0h	Reserved. Reads as 0.

Table 1-10. LEACMCTL Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
0	LEACMAE	Rw	0h	This bit controls access to LEA code memory 0 = code memory access disabled. Accesses to LEA code memory are not possible. LEA does accept commands for execution. Reads to LEA code memory will return zeroes and writes are ignored. 1 = code memory access enabled. Accesses to LEA code memory are possible. LEA does not accept commands during this mode (command is ignored). Coprocessor interface accesses by the CPU cause a Coprocessor not available indication.

1.9 LEACMDSTAT, LEACMDSTATL, LEACMDSTATH LEA Command Status Register

The data read from this register reflects the command and context that is currently being processed by LEA.

Figure 1-9. LEACMDSTAT Register

31	30	29	28	27	26	25	24
Reserved							
r0	r0	r0	r0	r0	r0	r0	r0
23	22	21	20	19	18	17	16
Reserved							
r0	r0	r0	r0	r0	r0	r0	r0
15	14	13	12	11	10	9	8
Reserved						LEACMD	
r0	r0	r0	r0	r0	r0	r-0	r-0
7	6	5	4	3	2	1	0
LEACMD						LEAITFLG	
r-0	r-0	r-0	r-0	r-0	r-0	r-0	r-0

Table 1-11. LEACMDSTAT Register Field Descriptions

Bit	Field	Type	Reset	Description
31-10	Reserved	r	0h	Reserved. Reads as 0.
9-2	LEACMD	r	0h	These bits represent the LEA command to be invoked. See also the command table
1-0	LEAITFLG	r	0h	LEA instruction handshake synchronization type flags 00 = LEA command without any further indication 01 = LEA command with explicit result update 10 = LEA command with interrupt upon completion 11 = LEA command with interrupt and explicit result update

1.10 LEAS0STAT, LEAS0STATL LEAS0STATH LEA Source 0 Status Buffer Register

The data read from this register reflects the source register content that is currently being processed by LEA.

Figure 1-10. LEAS0STAT Register

31	30	29	28	27	26	25	24
LEAS0VAL							
r-0	r-0	r-0	r-0	r-0	r-0	r-0	r-0
23	22	21	20	19	18	17	16
LEAS0VAL							
r-0	r-0	r-0	r-0	r-0	r-0	r-0	r-0
15	14	13	12	11	10	9	8
LEAS0VAL							
r-0	r-0	r-0	r-0	r-0	r-0	r-0	r-0
7	6	5	4	3	2	1	0
LEAS0VAL							
r-0	r-0	r-0	r-0	r-0	r-0	r-0	r-0

Table 1-12. LEAS0STAT Register Field Descriptions

Bit	Field	Type	Reset	Description
31-0	LEAS0VAL	rw	0h	LEA source register value

1.11 LEAS1STAT, LEAS1STATL LEAS1STATH LEA Source 1 Status Buffer Register

The data read from this register reflects the source register content that is currently being processed by LEA.

Figure 1-11. LEAS1STAT Register

31	30	29	28	27	26	25	24
LEAS1VAL							
r-0	r-0	r-0	r-0	r-0	r-0	r-0	r-0
23	22	21	20	19	18	17	16
LEAS1VAL							
r-0	r-0	r-0	r-0	r-0	r-0	r-0	r-0
15	14	13	12	11	10	9	8
LEAS1VAL							
r-0	r-0	r-0	r-0	r-0	r-0	r-0	r-0
7	6	5	4	3	2	1	0
LEAS1VAL							
r-0	r-0	r-0	r-0	r-0	r-0	r-0	r-0

Table 1-13. LEAS1STAT Register Field Descriptions

Bit	Field	Type	Reset	Description
31-0	LEAS1VAL	rw	0h	LEA source register value

1.12 LEAPMCTL, LEACTLL LEA Peripheral Mapped Control Registers

Figure 1-12. LEAPMCTL Register

31	30	29	28	27	26	25	24
Reserved							
r0	r0	r0	r0	r0	r0	r0	r0
23	22	21	20	19	18	17	16
Reserved							
r0	r0	r0	r0	r0	r0	r0	r0
15	14	13	12	11	10	9	8
Reserved							
r0	r0	r0	r0	r0	r0	r0	r0
7	6	5	4	3	2	1	0
TRG	Reserved						CMDEN
r	r0	r0	r0	r0	r0	r0	rw-0

Table 1-14. LEAPMCTL Register Field Descriptions

Bit	Field	Type	Reset	Description
31-8	Reserved	r	0h	Reserved. Reads as 0.
7	TRG	r	0h	This bit indicates a command trigger
6-1	Reserved	R	0h	Reserved. Reads as 0.
0	CMDEN	rw	0h	Peripheral Mapped Command enable 0 = Command triggering by writing to LEAPMCB is disabled 1 = Command triggering by writing to LEAPMCB is enabled

1.13 LEAPMDST, LEADSTL, LEADSTH LEA Peripheral Mapped Result Register

This register keeps the otional result of an LEA command. Reading of this register may cause an data inconcistency interrupt if the result of the command was not available at the time of the read operation. Reading LEADSTH sets the LEESCFREE flag. Writing LEADSTH clears the LEAFREE flag.

Figure 1-13. LEAPMDST Register

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

Table 1-15. LEAPMDST Register Field Descriptions

Bit	Field	Type	Reset	Description
31-0	LEAPMDST	rw	--	Result register of peripheral mapped commands. This register keeps result of an LEA operation that returns a result by value.

1.14 LEAPMS0, LEAS0L, LEAS0H LEA Peripheral Mapped Source 0 Register

Optional source register for arguments of an LEA command.

Figure 1-14. LEAPMS0 Register

31	30	29	28	27	26	25	24
LEAPMS0							
rw-0	rw-0	rw-0	rw-0	rw-0	rw-0	rw-0	rw-0
23	22	21	20	19	18	17	16
LEAPMS0							
rw-0	rw-0	rw-0	rw-0	rw-0	rw-0	rw-0	rw-0
15	14	13	12	11	10	9	8
LEAPMS0							
rw-0	rw-0	rw-0	rw-0	rw-0	rw-0	rw-0	rw-0
7	6	5	4	3	2	1	0
LEAPMS0							
rw-0	rw-0	rw-0	rw-0	rw-0	rw-0	rw-0	rw-0

Table 1-16. LEAPMS0 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-0	LEAPMS0	rw	0h	LEA source register value

1.15 LEAPMS1, LEAS1L, LEAS1H LEA Peripheral Mapped Source 1 Register

Optional source register for arguments of an LEA command.

Figure 1-15. LEAPMS1 Register

31	30	29	28	27	26	25	24
LEAPMS1							
rw-0	rw-0	rw-0	rw-0	rw-0	rw-0	rw-0	rw-0
23	22	21	20	19	18	17	16
LEAPMS1							
rw-0	rw-0	rw-0	rw-0	rw-0	rw-0	rw-0	rw-0
15	14	13	12	11	10	9	8
LEAPMS1							
rw-0	rw-0	rw-0	rw-0	rw-0	rw-0	rw-0	rw-0
7	6	5	4	3	2	1	0
LEAPMS1							
rw-0	rw-0	rw-0	rw-0	rw-0	rw-0	rw-0	rw-0

Table 1-17. LEAPMS1 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-0	LEAPMS1	rw	0h	LEA source register value

1.16 LEAPMCB, LEACBL, LEACBH LEA Peripheral Mapped Command Buffer Register

Writing to LEAPMCBL in 16-bit mode or writing to LEAPMCB in 32-bit mode invokes an LEA command when enabled. These write accesses are delayed if the LEA is still executing an previously invoked command, except that a SUSPEND command is immediately invoked. Writing to LEAPMCBH is always possible.

Figure 1-16. LEAPMCB Register

31	30	29	28	27	26	25	24
LEACTX							
rw-0	rw-0	rw-0	rw-0	rw-0	rw-0	rw-0	rw-0
23	22	21	20	19	18	17	16
LEACTX				Reserved			
rw-0	rw-0	rw-0	rw-0	r0	r0	r0	r0
15	14	13	12	11	10	9	8
Reserved						LEACMD	
r0	r0	r0	r0	r0	r0	rw-0	rw-0
7	6	5	4	3	2	1	0
LEACMD						LEAITFLG	
rw-0	rw-0	rw-0	rw-0	rw-0	rw-0	rw-0	rw-0

Table 1-18. LEAPMCB Register Field Descriptions

Bit	Field	Type	Reset	Description
31-20	LEACTX	rw	0h	Command context: This bit field may be set by the user together with invoking a command. This bit field is saved on SUSPEND and restored from the LEA stack on RESUME. This bit field is used by software to associate or synchronize LEA commands to a certain tasks and IDs.
19-10	Reserved	r	0h	Reserved. Reads as 0.
9-2	LEACMD	rw	0h	These bits represent the LEA command to be invoked. See also the command table
1-0	LEAITFLG	rw	0h	LEA instruction handshake synchronization type flag 00 = LEA command without any further indication 01 = LEA command with explicit result update 10 = LEA command with interrupt upon completion 11 = LEA command with interrupt and explicit result update

1.17 LEAIFGSET, LEAIFGSETL LEA Interrupt Flag and Set Registers

LEA interrupt flag set register

NOTE: To set an interrupt flag, write 1 to the corresponding interrupt flag bit.

Figure 1-17. LEAIFGSET Register

31	30	29	28	27	26	25	24
Reserved							
r0	r0	r0	r0	r0	r0	r0	r0
23	22	21	20	19	18	17	16
Reserved							
r0	r0	r0	r0	r0	r0	r0	r0
15	14	13	12	11	10	9	8
Reserved							
r0	r0	r0	r0	r0	r0	r0	r0
7	6	5	4	3	2	1	0
Reserved			CMDIS	LEASDIIS	LEAOORIS	LEATIS	LEACOVLIS
r0	r0	r0	rw-0	rw-0	rw-0	rw-0	rw-0

Table 1-19. LEAIFGSET Register Field Descriptions

Bit	Field	Type	Reset	Description
31-5	Reserved	r	0h	Reserved. Reads as 0.
4	CMDIS	rw	0h	Peripheral memory triggered Command completed interrupt flag. 0 = no interrupt pending 1 = interrupt pending
3	LEASDIIS	rw	0h	LEA scalar data inconsistency interrupt flag 0 = no interrupt pending 1 = interrupt pending
2	LEAOORIS	rw	0h	LEA out of address range interrupt flag. 0 = no interrupt pending 1 = interrupt pending
1	LEATIS	rw	0h	LEA timer interrupt flag 0 = no interrupt pending 1 = interrupt pending
0	LEACOVLIS	rw	0h	LEA command overflow interrupt flag 0 = no interrupt pending 1 = interrupt pending

1.18 LEAIE, LEAIEL LEA Interrupt Enable Register

LEA interrupt enable register

Figure 1-18. LEAIE Register

31	30	29	28	27	26	25	24
Reserved							
r0	r0	r0	r0	r0	r0	r0	r0
23	22	21	20	19	18	17	16
Reserved							
r0	r0	r0	r0	r0	r0	r0	r0
15	14	13	12	11	10	9	8
Reserved							
r0	r0	r0	r0	r0	r0	r0	r0
7	6	5	4	3	2	1	0
Reserved			CMDIE	LEASDIIE	LEAOORIE	LEATIE	LEACOVLIE
r0	r0	r0	rw-0	rw-0	rw-0	rw-0	rw-0

Table 1-20. LEAIE Register Field Descriptions

Bit	Field	Type	Reset	Description
31-5	Reserved	r	0h	Reserved. Reads as 0.
4	CMDIE	rw	0h	Peripheral memory triggered Command completed interrupt enable. 0 = interrupt disabled 1 = interrupt enabled
3	LEASDIIE	rw	0h	LEA scalar data inconsistency interrupt enable 0 = interrupt disabled 1 = interrupt enabled
2	LEAOORIE	rw	0h	LEA out of address range interrupt enable 0 = interrupt disabled 1 = interrupt enabled
1	LEATIE	rw	0h	LEA timer event interrupt enable 0 = interrupt disabled 1 = interrupt enabled
0	LEACOVLIE	rw	0h	LEA command overflow interrupt enable 0 = interrupt disabled 1 = interrupt enabled

1.19 LEAIFG, LEAIFGL LEA Interrupt Flag and Clear Registers

LEA interrupt flag register

NOTE: To clear an interrupt flag, write 1 to the corresponding interrupt flag bit.

Figure 1-19. LEAIFG Register

31	30	29	28	27	26	25	24
Reserved							
r0	r0	r0	r0	r0	r0	r0	r0
23	22	21	20	19	18	17	16
Reserved							
r0	r0	r0	r0	r0	r0	r0	r0
15	14	13	12	11	10	9	8
Reserved							
r0	r0	r0	r0	r0	r0	r0	r0
7	6	5	4	3	2	1	0
Reserved			CMDIFG	LEASDIIFG	LEAOORIFG	LEATIFG	LEACOVLIFG
r0	r0	r0	rw-0	rw-0	rw-0	rw-0	rw-0

Table 1-21. LEAIFG Register Field Descriptions

Bit	Field	Type	Reset	Description
31-5	Reserved	r	0h	Reserved. Reads as 0.
4	CMDIFG	rw	0h	Peripheral memory triggered Command completed interrupt flag. 0 = no interrupt pending 1 = interrupt pending
3	LEASDIIFG	rw	0h	LEA scalar data inconsistency interrupt flag 0 = no interrupt pending 1 = interrupt pending
2	LEAOORIFG	rw	0h	LEA out of address range interrupt flag 0 = no interrupt pending 1 = interrupt pending
1	LEATIFG	rw	0h	LEA timer interrupt flag 0 = no interrupt pending 1 = interrupt pending
0	LEACOVLIFG	rw	0h	LEA command overflow interrupt flag 0 = no interrupt pending 1 = interrupt pending

1.20 LEAIV, LEAIVL LEA Interrupt Vector Register

Figure 1-20. LEAIV Register

31	30	29	28	27	26	25	24
Reserved							
r0	r0	r0	r0	r0	r0	r0	r0
23	22	21	20	19	18	17	16
Reserved							
r0	r0	r0	r0	r0	r0	r0	r0
15	14	13	12	11	10	9	8
Reserved							
r0	r0	r0	r0	r0	r0	r0	r0
7	6	5	4	3	2	1	0
LEAIV							
r0	r0	r0	r-0	r-0	r-0	r-0	r0

Table 1-22. LEAIV Register Field Descriptions

Bit	Field	Type	Reset	Description
31-8	Reserved	r	0h	Reserved. Reads as 0.
7-0	LEAIV	rw	0h	LEA interrupt vector. This is a generated value that can be used as address offset for fast interrupt service routine handling. Reading this register clears the highest pending LEA interrupt (displaying this register with an IDE does not affect its content). Writing to this register clears al pending interrupt flags. This value is always aligned to a 20 bit address offset boundary 0000h = No interrupt pending 0002h = LEA command overflow 0004h = LEA timer interrupt 0006h = LEA out of range interrupt 0008h = LEA scalar data inconsistency 000Ah = PMCMD complete interrupt other = reserved

2 Revision History

NOTE: Page numbers for previous revisions may differ from page numbers in the current version.

DATE	REVISION	NOTES
March 2021	*	Initial Release

IMPORTANT NOTICE AND DISCLAIMER

TI PROVIDES TECHNICAL AND RELIABILITY DATA (INCLUDING DATA SHEETS), DESIGN RESOURCES (INCLUDING REFERENCE DESIGNS), APPLICATION OR OTHER DESIGN ADVICE, WEB TOOLS, SAFETY INFORMATION, AND OTHER RESOURCES “AS IS” AND WITH ALL FAULTS, AND DISCLAIMS ALL WARRANTIES, EXPRESS AND IMPLIED, INCLUDING WITHOUT LIMITATION ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT OF THIRD PARTY INTELLECTUAL PROPERTY RIGHTS.

These resources are intended for skilled developers designing with TI products. You are solely responsible for (1) selecting the appropriate TI products for your application, (2) designing, validating and testing your application, and (3) ensuring your application meets applicable standards, and any other safety, security, regulatory or other requirements.

These resources are subject to change without notice. TI grants you permission to use these resources only for development of an application that uses the TI products described in the resource. Other reproduction and display of these resources is prohibited. No license is granted to any other TI intellectual property right or to any third party intellectual property right. TI disclaims responsibility for, and you will fully indemnify TI and its representatives against, any claims, damages, costs, losses, and liabilities arising out of your use of these resources.

TI’s products are provided subject to TI’s Terms of Sale or other applicable terms available either on ti.com or provided in conjunction with such TI products. TI’s provision of these resources does not expand or otherwise alter TI’s applicable warranties or warranty disclaimers for TI products.

TI objects to and rejects any additional or different terms you may have proposed.

Mailing Address: Texas Instruments, Post Office Box 655303, Dallas, Texas 75265