Appendix: List of events which Linux Kernel State Tracer records on IA32

	Appendix: List of events which Linux Kernel State Tracer records on IA32 Version 2.02 Version 2.02								rights reserved		
Event type [hex]		Mnemonic	Descriptio	on of events	where to hook	filename	data recorded as "log arg1"	data recorded as "log arg2"	data recorded as "log arg3"	data recorded as "log arg4"	remarks
						/hornol/ochod o	address of the task_struct	address of the task_struct	prev. process state (value after	prev. process count (value before	from log_arg3, log_arg4, can determain
01	Process	PROCESS_CONTEXTSWITC	Process context switching		schedule()	./kernel/sched.c	of "prev"	of "next"	switch)	switch)	why processes were switched
02		PROCESS_WAKEUP	WAKEUP		try_to_wake_up()		value of "p" in the function	synchronous			
03	management	PROCESS_SIGSEND	sending signal		specific_send_sig_info()	./kernel/signal.c	value of "sig" in the function	value of "t" in the function	pointer to info (info)		
04		PROCESS_LTHREADGEN		•	kernel_thread()	./arch/i386/kernel/process.c	value of "fn" in the function	pointer to argument of kernel thread (arg			
10		INT_HARDWARE_ENTRY	hardware	entrance	do_IRQ()	./arch/i386/kernel/irq.c	value of "irq" in the function	interrupt status (status)	pointer to register stack		
12	Interrupts	INT_TASKLETHI_ENTRY		entrance	tasklet_hi_action()	./kernel/softirq.c	value of "t->func" in the function				
14		INT_TASKLET_ENTRY	software	entrance entrance	tasklet_action()	-	value of "t->func" in the function	address of action (bb. bass)			
16		INT_BH_ENTRY	do	entrance	bh_action()		value of "nr" in the function	address of action (bh_base)			
20	Exceptions	invalid_TSS segment_not_pri stack_segment alignment_check coprocessor_erri	overflow bounds invalid_op double_fault Coprocessor_segment_overrun invalid_TSS segment_not_present	entrance error_code device_not_available nmi device_not_available nmi nmi	error_code	./arch/i386/kernel/entry.S	handler address (edi)	error code (esi)	exception occurred address (eip)		
21			debug general_protection page_fault machine_check sprious_interrupt_bug device_not_available nmi device_not_available		nmi		the number of this exception handler address			-	
21		EXCEPTION_EXIT	nmi exit exceptions other than above two		nmi error code	4	the number of this exception handler address (edi)	1	+	4	
	1			1							recording arguments of system calls is
30		SYSCALL_ENTRY	entrance		beginning of system_call()	./arch/i386/kernel/entry.S	the number of this system call		1		optional feature
31		SYSCALL_EXIT	exit		ending of system_call()	./arch/i386/kernel/entry.S	the number of this system call	errno	<u> </u>	1	
	System calls		1						1	1	recording arguments of system calls is
32		SYSCALL_SYSENTER	sysenter instruction entrance		beginning of sysenter_entry()	./arch/i386/kernel/entry.S	the number of this system call				optional feature
33	1	SYSCALL_SYSEXIT	sysexit instruction exit		ending of sysenter_entry()	./arch/i386/kernel/entry.S	the number of this system call	errno			
50		MEM_SWAPOUT	swap out	exit exit	try_to_swap_out()	./mm/vmscan.c	pointer to page swapped out (page)				
51	_	MEM_SWAPIN	swap in		do_swap_page()	./mm/memory.c	pointer to page swapped in (page)				
52		MEM_DO_NOPAGE	mem_do_nopage	exit	do_no_page()	./mm/memory.c	pointer to page allocated (new_page)				
53	4	MEM_DO_WPPAGE	mem_do_wppage		do_wp_page()	./mm/memory.c	pointer to page (new page)				
54		MEM_WAIT_PAGE	mem_wait_page	entrance	wait_on_page()	./mm/filemap.c	pointer to page (page)				
55		MEM_GET_FREEPAGE	mem_get_freepage	exit	get_free_page()	./mm/page_alloc.c	pointer to page (paddr)	type of page (gfp_mask)	the number of page (order)	call address	
56	Memory	MEM_GET_ZEROPAGE	mem_get_zeropage	exit	get_zeroed_page()	./mm/page_alloc.c	pointer to page (address)	type of page (gfp_mask)	call address		
57	Management	MEM_FREEPAGE	mem_freepage	entrance	free_pages()	./mm/page_alloc.c	pointer to (addr)	the number of page (order)	call address call address		
58	Junior	MEM_VMALLOC MEM_VFREE	mem_vmalloc	exit entrance	vmalloc()	./mm/vmalloc.h ./mm/vmalloc.c	address (addr) address (addr)	size	call address		
59 5a	_	MEM_VEREE	mem_vfree mem cache create	exit	vfree() kmem cache create()	./mm/slab.c	name	size	cachep	-	
5b	-	MEM_CACHE_CREATE	mem_cache_alloc	exit	kmem cache alloc()	./mm/slab.c	cachep	flags	objp	call address	
50 50	-	MEM_CACHE_ALLOC	mem malloc	exit	kmalloc()	./mm/slab.c	cachep	flags	objp	call address	
5d	-	MEM CACHE FREE	mem cache free	entrance	kmem cache free()	./mm/slab.c	cachep	objp	call address		
5e	_	MEM_GREE	mem free	entrance	kfree()	./mm/slab.c	objp	call address			
60		NET_PKTSEND	sending packets	entrance	dev queue xmit()	./net/core/dev.c	skb				
61		NET_PKTSENDI	interrupt on sending packets	entrance	net tx action()	./net/core/dev.c	h				
62	Networking	NET PKTRECV	receiving packets	entrance	netif_rx()	./net/core/dev.c	skb				
63	ů	NET_PKTRECVI	interrupt on receiving packets	entrance	net_rx_action()	./net/core/dev.c	h				
64		NET_SOCKETIF	socket()	entrance	sys_socketcall	./net/socket.c	call	args			exit is recorded as exit of system call.
70		SYSV_IPC_SEMOP			sys_semop()		semid	tsops	nsops		
71		SYSV_IPC_SEMGET			sys_semget()	./ipc/sem.c	key	nsems	semflg		
72	_	SYSV_IPC_SEMCTL			sys_semctl()		semid	semnum	cmd	argument for the function	
73 74	_	SYSV_IPC_MSGSEND	-		sys_msgsend()	-	msqid msqid/msgflg	msgp	msgsz msgsz	msgflg	
74	SysV IPC	SYSV_IPC_MSGRCV SYSV_IPC_MSGGET	IPC functions	entrance	sys_msgrcv() sys_msgget()	./ipc/msg.c	key	msgp msqflq	IIISysz	msgtyp	
76	Sysv IFC	SYSV IPC MSGGET	IF C IUNCIONS	entrance	sys_msgctl()	-	msqid	cmd	buf		
77	-	SYSV IPC SHMAT			sys shmat()		shmid	shmaddr	shmflg	raddr	
78	1	SYSV_IPC_SHMDT	1		sys_shmdt()	1 /m = / = h = = =	shmaddr		9		1
79		SYSV IPC SHMGET	1		sys_shmget()	./ipc/shm.c	key	size	shmflg		
7a		SYSV_IPC_SHMCTL			sys_shmctl()		shmid	cmd	buf		
80		LK_SPINLOCK		lock	spin_lock()		address where it was called	lock			inline
81	4	LK_SPINTRYLOCK	spin lock	try lock (exit)	spin_trylock()	4	address where it was called	lock	return value		inline
82	_	LK SPINUNLOCK								1	inline
				unlock	spin_unlock()	-	address where it was called	lock			
83	Locks	LK_WRLOCK	1	write lock	write_lock()	./include/asm-i386/spinlock.h	address where it was called	rwlock			inline
84	Locks	LK_WRLOCK LK_WRTRYLOCK	read/write lock	write lock write try lock (exit)	write_lock() write_trylock()	./include/asm-i386/spinlock.h	address where it was called address where it was called	rwlock rwlock	return value		inline inline
84 85	Locks	LK_WRLOCK LK_WRTRYLOCK LK_WRUNLOCK	read/write lock	write lock write try lock (exit) write unlock	write_lock() write_trylock() write_unlock()	./include/asm-i386/spinlock.h	address where it was called	rwlock rwlock rwlock	return value		inline inline define
84	Locks	LK_WRLOCK LK_WRTRYLOCK LK_WRUNLOCK LK_RDLOCK	read/write lock	write lock write try lock (exit)	write_lock() write_trylock()	/include/asm-i386/spinlock.h	address where it was called address where it was called address where it was called	rwlock rwlock	return value		inline inline
84 85 86	Locks	LK_WRLOCK LK_WRTRYLOCK LK_WRUNLOCK LK_RDLOCK LK_RDUNLOCK TIMER_RUN	read/write lock	write lock write try lock (exit) write unlock read lock	write_lock() write_trylock() write_unlock() read_lock() read_unlock() run_timer_list()	./include/asm-i386/spinlock.h	address where it was called address where it was called address where it was called address where it was called	rwłock rwłock rwłock rwłock rwłock	return value		inline inline define inline
84 85 86 87 a0 a1	Locks	LK_WRLOCK LK_WRTRYLOCK LK_WRUNLOCK LK_RDLOCK LK_RDUNLOCK TIMER_RUN TIMER_ADD	run timer list add to timer list	write lock write try lock (exit) write unlock read lock	write_lock() write_trylock() write_unlock() read_lock() read_unlock() run_timer_list() add_timer()		address where it was called address where it was called address where it was called address where it was called address where it was called function address(fn) pointer to timer list (timer)	rwlock rwlock rwlock rwlock rwlock argument for the function(data) unexpired term (timer->expires)	function address (timer->function)	argument for the function (timer-	inline inline define inline
84 85 86 87 a0 a1 a2	Locks Timer	LK_WRLOCK LK_WRTRYLOCK LK_BUNLOCK LK_RDUNLOCK LK_RDUNLOCK TIMER_RUN TIMER_ADD TIMER_MOD	run timer list add to timer list modify timer list	write lock write try lock (exit) write unlock read lock	write_lock() write_trylock() write_unlock() fread_lock() read_unlock() run_timer_list() add_timer() mod_timer()	./include/asm-i386/spinlock.h	address where it was called address where it was called address where it was called address where it was called address where it was called function address(fn) pointer to timer list (timer) pointer to timer list (timer)	rwlock rwlock rwlock rwlock rwlock argument for the function(data) unexpired term (timer->expires) unexpired term (timer->expires)	function address (timer->function) function address (timer->function)	argument for the function (timer-	inline inline define inline
84 85 86 87 a0 a1 a2 a3	-	LK WRLOCK LK WRTRYLOCK LK WRUNLOCK LK RDLOCK LK RDUNLOCK TIMER_RUN TIMER_ADD TIMER_ADD TIMER_MOD TIMER_DEL	run timer list add to timer list modify timer list delete from timer list	write lock write try lock (exit) write unlock read lock read unlock	write_lock() write_trylock() write_unlock() read_lock() read_unlock() run_timer_list() add_timer() mod_timer() del_timer()		address where it was called address where it was called address where it was called address where it was called address where it was called function address(fin) pointer to timer list (timer) pointer to timer list (timer)	rwlock rwlock rwlock rwlock argument for the function(data) unexpired term (timer->expires) unexpired term (timer->expires)	function address (timer->function) function address (timer->function) function address (timer->function)	argument for the function (timer- argument for the function (timer-	inline inline define inline
84 85 86 87 a0 a1 a2 a3 a4	-	LK_WRLOCK LK_WRTRYLOCK LK_WRUNLOCK LK_RDUNLOCK LK_RDUNLOCK TIMER_RUN TIMER_ADD TIMER_MOD TIMER_DEL TIMER_DEL TIMER_DEL_SYNC	run timer list add to timer list modify timer list delete from timer list with synchro	write lock write try lock (exit) write unlock read lock read unlock	write_lock() write_trylock() write_unlock() read_lock() read_unlock() run_timer_list() add_timer() mod_timer() del_timer_sync()	/kernel/timer.c	address where it was called address where it was called address where it was called address where it was called function address(fn) pointer to timer list (timer) pointer to timer list (timer) pointer to timer list (timer) pointer to timer list (timer)	rwlock rwlock rwlock rwlock argument for the function(data) unexpired term (timer->expires) unexpired term (timer->expires) unexpired term (timer->expires)	function address (timer->function) function address (timer->function) function address (timer->function) function address (timer->function)	argument for the function (timer- argument for the function (timer-	inline inline define inline
84 85 86 87 a0 a1 a2 a3 a4 b0	-	LK_WRLOCK LK_WRUNLOCK LK_RDLOCK LK_RDLOCK TIMER_RUN TIMER_ADD TIMER_MOD TIMER_DEL TIMER_DEL_SYNC OOPS_PGFAULT	run timer list add to timer list modify timer list delete from timer list with synchro delete from timer list with synchro goops in page fault handler	write lock write try lock (exit) write unlock read lock read unlock onous just before the cops operation	write_lock() write_trylock() write_unlock() read_lock() read_unlock() run_timer_list() add_timer() del_timer() del_timer_sync() do_page_fault()	/kernel/timer.c ./arch/i386/mm/fault.c	address where it was called address where it was called address where it was called address where it was called address where it was called function address(fn) pointer to timer list (timer) pointer to timer list (timer) pointer to timer list (timer) pointer to timer list (timer) address where it was accessed	rwlock rwlock rwlock rwlock argument for the function(data) unexpired term (timer->expires) unexpired term (timer->expires)	function address (timer->function) function address (timer->function) function address (timer->function)	argument for the function (timer- argument for the function (timer-	inline inline define inline
84 85 86 87 a0 a1 a2 a3 a4 b0 b1	Timer	LK_WRLOCK LK_WRUNLOCK LK_RDLOCK LK_RDLOCK TIMER_RUN TIMER_ADD TIMER_MOD TIMER_DEL TIMER_DEL SYNC OOPS_PGFAULT OOPS_NMIWDOG	run timer list add to timer list modify timer list delete from timer list with synchro	write lock write try lock (exit) write unlock read lock read unlock	write_lock() write_trylock() write_unlock() read_lock() read_unlock() run_timer_list() add_timer() dot_timer() del_timer() del_timer() del_timer() del_timer() del_timer() del_timer()	/kernel/timer.c	address where it was called address where it was called address where it was called address where it was called function address(fn) pointer to timer list (timer) pointer to timer list (timer) pointer to timer list (timer) pointer to timer list (timer)	rwlock rwlock rwlock rwlock argument for the function(data) unexpired term (timer->expires) unexpired term (timer->expires) unexpired term (timer->expires)	function address (timer->function) function address (timer->function) function address (timer->function) function address (timer->function)	argument for the function (timer- argument for the function (timer-	inline inline define inline
84 85 86 87 a0 a1 a2 a3 a4 b0	Timer	LK_WRLOCK LK_WRUNLOCK LK_RDLOCK LK_RDLOCK TIMER_RUN TIMER_ADD TIMER_MOD TIMER_DEL TIMER_DEL_SYNC OOPS_PGFAULT	run timer list add to timer list modify timer list delete from timer list delete from timer list with synchro oops in page fault handler oops in nmi watchdog timer	write lock write try lock (exit) write unlock read lock read unlock pread unlock pread unlock procession just before the oops operation just before the oops operation	write_lock() write_trylock() write_unlock() read_lock() read_unlock() run_timer_list() add_timer() del_timer() del_timer_sync() do_page_fault() nmi_watchdog_tick() OUT() or betweenOUT1() and	/kernel/timer.c //arch/i386/mm/fault.c /arch/i386/kernel/nmi.c	address where it was called address where it was called address where it was called address where it was called function address(fn) pointer to timer list (timer) pointer to timer list (timer) pointer to timer list (timer) pointer to timer list (timer) address where it was accessed address where it was running	rwlock rwlock rwlock rwlock argument for the function(data) unexpired term (timer->expires) unexpired term (timer->expires) unexpired term (timer->expires) address where exception occurred	function address (timer->function) function address (timer->function) function address (timer->function) function address (timer->function) exception error code	argument for the function (timer- argument for the function (timer-	inline inline define inline
84 85 86 87 a0 a1 a2 a3 a4 b0 b1	Timer	LK_WRLOCK LK_WRUNLOCK LK_RDLOCK LK_RDLOCK TIMER_RUN TIMER_ADD TIMER_MOD TIMER_DEL TIMER_DEL SYNC OOPS_PGFAULT OOPS_NMIWDOG	run timer list add to timer list modify timer list delete from timer list with synchro delete from timer list with synchro goops in page fault handler	write lock write try lock (exit) write unlock read lock read unlock onous just before the cops operation	write_lock() write_trylock() write_unlock() read_lock() read_unlock() run_timer_list() add_timer() dot_timer() del_timer() del_timer() del_timer() del_timer() del_timer() del_timer()	/kernel/timer.c ./arch/i386/mm/fault.c	address where it was called address where it was called address where it was called address where it was called address where it was called function address(fn) pointer to timer list (timer) pointer to timer list (timer) pointer to timer list (timer) pointer to timer list (timer) address where it was accessed	rwlock rwlock rwlock rwlock argument for the function(data) unexpired term (timer->expires) unexpired term (timer->expires) unexpired term (timer->expires)	function address (timer->function) function address (timer->function) function address (timer->function) function address (timer->function)	argument for the function (timer- argument for the function (timer-	inline inline define inline define
84 85 86 87 a0 a1 a2 a3 a4 b0 b1 90	Timer Oops	LK_WRLOCK LK_WRTRYLOCK LK_WRUNLOCK LK_RDUNLOCK IMER_RUN TIMER_ADD TIMER_ADD TIMER_DEL TIMER_DEL TIMER_DEL TIMER_DEL OOPS_PGFAULT OOPS_NMIWDOG 0_PORTIN 0_PORTOUT 0_PANIC	run timer list add to timer list modify timer list delete from timer list delete from timer list with synchro oops in page fault handler oops in nmi watchdog timer	write lock write try lock (exit) write unlock read lock read unlock just before the oops operation just before the oops operation port output	write_lock() write_trylock() write_unlock() read_lock() read_unlock() run_timer_list() add_timer() del_timer() del_timer() del_timer() del_timer() del_timer_sync() do_page_fault() nmi_watchdog_tick() _OUT() or betweenOUT1() andOUT2()	/kernel/timer.c /arch/i386/mm/fault.c /arch/i386/kernel/nmi.c /include/asm-i386/io.h /kernel/panic.c	address where it was called address where it was called address where it was called address where it was called address where it was called function address(fn) pointer to timer list (timer) pointer to timer list (timer) pointer to timer list (timer) address where it was accessed address where it was running port address/byte width	rwlock rwlock rwlock rwlock rwlock argument for the function(data) unexpired term (timer->expires) unexpired term (timer->expires) unexpired term (timer->expires) address where exception occurred value to output value to input address where it was called	function address (timer->function) function address (timer->function) function address (timer->function) function address (timer->function) exception error code address where it was called	argument for the function (timer- argument for the function (timer-	inline inline define inline define inline inline inline inline inline inline inline inline inline
84 85 86 87 a0 a1 a2 a3 a4 b0 b1 90 91 92 93	Timer Oops Others	LK WRLOCK LK WRTRYLOCK LK WRUNLOCK LK RDUNLOCK LK RDUNLOCK TIMER RUN TIMER ADD TIMER_DEL TIMER_DEL TIMER_DEL TIMER_DEL_SYNC OOPS_NMIWDOG 0_PORTIN 0_PORTOUT 0_PRINTK	run timer list add to timer list modify timer list delete from timer list with synchro oops in page fault handler oops in nmi watchdog timer io commands panic printk	write lock write try lock (exit) write unlock read lock read unlock just before the oops operation just before the oops operation just before the oops operation port output port input	write_lock() write_trylock() write_unlock() read_lock() read_unlock() run_timer_list() add_timer() del_timer() del_timer() del_timer_sync() do_page_fault() nmi_watchdog_tick() _OUT2() tail ofIN()	/kernel/timer.c /arch/i386/mm/fault.c /arch/i386/kernel/nmi.c /include/asm-i386/io.h /kernel/panic.c /kernel/printk.c	address where it was called address where it was called address where it was called address where it was called address where it was called function address(fn) pointer to timer list (timer) pointer to timer list (timer) pointer to timer list (timer) address where it was accessed address where it was accessed address where it was running port address/byte width port address/byte width address of argument	rwlock rwlock rwlock rwlock rwlock argument for the function(data) unexpired term (timer->expires) unexpired term (timer->expires) unexpired term (timer->expires) unexpired term (timer->expires) address where exception occurred value to output value to input	function address (timer->function) function address (timer->function) function address (timer->function) function address (timer->function) exception error code address where it was called	argument for the function (timer- argument for the function (timer-	inline inline define inline define inline inline inline inline inline inline inline inline inline
84 85 86 87 a0 a1 a2 a3 a4 b0 b1 b1 90 91 92	Timer Oops	LK_WRLOCK LK_WRTRYLOCK LK_WRUNLOCK LK_RDUNLOCK IMER_RUN TIMER_ADD TIMER_ADD TIMER_DEL TIMER_DEL TIMER_DEL TIMER_DEL OOPS_PGFAULT OOPS_NMIWDOG 0_PORTIN 0_PORTOUT 0_PANIC	run timer list add to timer list modify timer list delete from timer list with synchro oops in page fault handler oops in nmi watchdog timer io commands panic	write lock write try lock (exit) write unlock read lock read unlock just before the oops operation just before the oops operation just before the oops operation port output port input	write_lock() write_trylock() write_unlock() read_lock() read_unlock() run_timer_list() add_timer() del_timer() del_timer() del_timer() del_timer() del_timer_sync() do_page_fault() nmi_watchdog_tick() _OUT() or betweenOUT1() andOUT2()	/kernel/timer.c /arch/i386/mm/fault.c /arch/i386/kernel/nmi.c /include/asm-i386/io.h /kernel/panic.c	address where it was called address where it was called address where it was called address where it was called function address(n) pointer to timer list (timer) pointer to timer list (timer) pointer to timer list (timer) pointer to timer list (timer) address where it was accessed address where it was running port address/byte width port address/byte width address of argument	rwlock rwlock rwlock rwlock rwlock argument for the function(data) unexpired term (timer->expires) unexpired term (timer->expires) unexpired term (timer->expires) address where exception occurred value to output value to input address where it was called	function address (timer->function) function address (timer->function) function address (timer->function) function address (timer->function) exception error code address where it was called	argument for the function (timer- argument for the function (timer-	inline inline define inline inline inline inline inline inline
84 85 86 87 a0 a1 a2 a3 a4 b0 b1 90 91 92 93 f00	Timer Oops Others	LK_WRLOCK LK_WRTRYLOCK LK_WRUNLOCK LK_RDUNLOCK LK_RDUNLOCK IMER_RUN TIMER_ADD TIMER_ADD TIMER_DEL TIMER_DEL TIMER_DEL TIMER_DEL_SYNC OOPS_PGFAULT OOPS_NMIWDOG O_PORTIN O_PORTOUT O_PANIC O_PRINTK LKST_INIT	run timer list add to timer list modify timer list delete from timer list delete from timer list with synchro oops in page fault handler oops in nmi watchdog timer io commands panic printk Progress of LKST initialization pr	write lock write try lock (exit) write unlock read lock read unlock just before the oops operation just before the oops operation just before the oops operation port output port input	write_lock() write_trylock() write_unlock() read_lock() read_unlock() add_timer() add_timer() del_timer() del_timer() del_timer() dot_timer() del_timer() dot_timer() dot_timer() del_timer() dot_timer() <	/kernel/timer.c /arch/i386/mm/fault.c /arch/i386/kernel/nmi.c /include/asm-i386/io.h /kernel/panic.c /kernel/printk.c /driver/lkst/lkst_core.c	address where it was called address where it was called address where it was called address where it was called address where it was called function address(n) pointer to timer list (timer) pointer to timer list (timer) pointer to timer list (timer) address where it was accessed address where it was accessed address where it was running port address/byte width address of argument address of argument address of argument	rwlock rwlock rwlock rwlock rwlock argument for the function(data) unexpired term (timer->expires) unexpired term (timer->expires) unexpired term (timer->expires) address where exception occurred value to output value to output address where it was called address where it was called	function address (timer->function) function address (timer->function) function address (timer->function) function address (timer->function) exception error code address where it was called	argument for the function (timer- argument for the function (timer-	inline inline define inline define inline inline inline This event is embeded in LKST. User
84 85 86 87 a0 a1 a2 a3 a4 b0 b1 90 91 92 93 f00 f01	Timer Oops Others	LK WRLOCK LK WRTRYLOCK LK WRUNLOCK LK RDUNLOCK LK RDUNLOCK TIMER_RUN TIMER_ADD TIMER_DEL TIMER_DEL TIMER_DEL TIMER_DEL OOPS_PGFAULT OOPS_NMIWDOG O_PORTIN O_PORTOUT O_PRINTK LKST_INIT LKST_KERNEL_DUMP	run timer list add to timer list modify timer list delete from timer list with synchro oops in page fault handler oops in nmi watchdog timer io commands panic printk Progress of LKST initialization pr kernel dump event	write lock write try lock (exit) write unlock read lock read unlock just before the oops operation just before the oops operation just before the oops operation port output port input	write_lock() write_trylock() write_unlock() read_lock() read_unlock() read_unlock() immer_list() add_timer() del_timer() del_timer() del_timer() del_timer_sync() do_page_fault() nmi_watchdog_tick() OUT2() tail ofIN() Ikst_init_stage[0-1]() Ikst_dump_notify_handler()	/kernel/timer.c /arch/i386/mm/fault.c /arch/i386/kernel/nmi.c /include/asm-i386/io.h /kernel/panic.c /kernel/pinitk.c /driver/lkst/lkst_core.c	address where it was called address where it was called address where it was called address where it was called address where it was called function address(fin) pointer to timer list (timer) pointer to timer list (timer) pointer to timer list (timer) address where it was accessed address where it was accessed address where it was running port address/byte width port address/byte width address of argument initialization status dump state	rwlock rwlock rwlock rwlock rwlock argument for the function(data) unexpired term (timer->expires) unexpired term (timer->expires) unexpired term (timer->expires) address where exception occurred value to output value to output value to input address where it was called address where it was called dump device	function address (timer->function) function address (timer->function) function address (timer->function) function address (timer->function) exception error code address where it was called address where it was called	argument for the function (timer- argument for the function (timer- argument for the function (timer-	inline inline define inline inline define inline inline inline inline inline can't handle it.
84 85 86 87 a0 a1 a2 a3 a4 b0 b1 90 91 92 93 f00 f01 f08	Timer Oops Others	LK WRLOCK LK WRTRYLOCK LK WRUNLOCK LK RDLOCK LK RDUNLOCK IMBER RUN TIMER RUN TIMER ADD TIMER_DEL TIMER_DEL TIMER DEL TIMER DEL TIMER DEL OOPS_NMIWDOG O_PORTIN O_PORTOUT O_PANIC O_PRINTK LKST_INIT LKST_KERNEL_DUMP LKST_MSET_XCHG	run timer list add to timer list modify timer list delete from timer list with synchro oops in page fault handler oops in nmi watchdog timer io commands panic printk Progress of LKST initialization pr kernel dump event LKST switches the masksets	write lock write try lock (exit) write unlock read lock read unlock just before the oops operation just before the oops operation just before the oops operation port output port input	write_lock() write_trylock() write_unlock() read_lock() read_unlock() read_unlock() read_unlock() add_timer() add_timer() del_timer() del_timer() del_timer() del_timer() del_timer() dol_timer() del_timer() lkst_dump_notify_handler() lkst_evhandlerprim_maskset_xchg_init	/kernel/timer.c /arch/i386/mm/fault.c /arch/i386/kernel/nmi.c /include/asm-i386/io.h /kernel/panic.c /kernel/printk.c /driver/lkst/lkst_core.c /driver/lkst/lkst_core.c	address where it was called address where it was called address where it was called address where it was called address where it was called function address(fn) pointer to timer list (timer) pointer to timer list (timer) pointer to timer list (timer) address where it was accessed address where it was accessed address where it was running port address/byte width address of argument address of argument address of argument dump state old maskset ID	rwlock rwlock rwlock rwlock rwlock argument for the function(data) unexpired term (timer->expires) unexpired term (timer->expires) unexpired term (timer->expires) address where exception occurred value to output value to output value to input address where it was called address where it was called dump device new maskset ID	function address (timer->function) function address (timer->function) function address (timer->function) function address (timer->function) exception error code address where it was called address where it was called pointer to old maskset	argument for the function (timer- argument for the function (timer- argument for the function (timer-	inline inline define inline define inline inline inline inline inline inline Recorded 2 times; before/after
84 85 86 87 a0 a1 a2 a3 a4 b0 b1 90 91 92 93 f00 f01 f08 f10	Timer Oops Others	LK WRLOCK LK WRTRYLOCK LK WRUNLOCK LK RDUNLOCK LK RDUNLOCK TIMER_RUN TIMER_ADD TIMER_DEL TIMER_DEL TIMER_DEL TIMER_DEL OOPS_PGFAULT OOPS_NMIWDOG O_PORTIN O_PORTOUT O_PRINTK LKST_INIT LKST_KERNEL_DUMP	run timer list add to timer list modify timer list delete from timer list with synchro oops in page fault handler oops in nmi watchdog timer io commands panic printk Progress of LKST initialization pr kernel dump event	write lock write try lock (exit) write unlock read lock read unlock just before the oops operation just before the oops operation just before the oops operation port output port input	write_lock() write_trylock() write_unlock() read_lock() read_unlock() read_unlock() immer_list() add_timer() del_timer() del_timer() del_timer() del_timer_sync() do_page_fault() nmi_watchdog_tick() OUT2() tail ofIN() Ikst_init_stage[0-1]() Ikst_dump_notify_handler()	/kernel/timer.c /arch/i386/mm/fault.c /arch/i386/kernel/nmi.c /include/asm-i386/io.h /kernel/panic.c /kernel/printk.c /driver/lkst/lkst_core.c /driver/lkst/lkst_core.c	address where it was called address where it was called address where it was called address where it was called address where it was called function address(fin) pointer to timer list (timer) pointer to timer list (timer) pointer to timer list (timer) address where it was accessed address where it was accessed address where it was running port address/byte width port address/byte width address of argument initialization status dump state	rwlock rwlock rwlock rwlock rwlock argument for the function(data) unexpired term (timer->expires) unexpired term (timer->expires) unexpired term (timer->expires) address where exception occurred value to output value to output value to input address where it was called address where it was called dump device	function address (timer->function) function address (timer->function) function address (timer->function) function address (timer->function) exception error code address where it was called address where it was called	argument for the function (timer- argument for the function (timer- argument for the function (timer-	inline inline define inline define inline inline inline This event is embeded in LKST. User can't handle it. Recorded 2 times; before/after Recorded 2 times; before/after
84 85 86 87 a0 a1 a2 a3 a4 b0 b1 90 91 92 93 f00 f01 f08	Timer Oops Others	LK WRLOCK LK WRTRYLOCK LK WRUNLOCK LK RDUNLOCK LK RDUNLOCK TIMER_RUN TIMER_ADD TIMER_DEL TIMER_DEL TIMER_DEL TIMER_DEL OOPS_PGFAULT OOPS_NMIWDOG O_PORTIN O_PORTOUT O_PRINTK LKST_INIT LKST_KERNEL_DUMP LKST_BUFF_SHIFT	run timer list add to timer list modify timer list delete from timer list with synchro oops in page fault handler oops in nmi watchdog timer io commands panic printk Progress of LKST initialization pr kernel dump event LKST switches the masksets LKST shifts the buffers	write lock write try lock (exit) write unlock read lock read unlock just before the oops operation just before the oops operation port output port input	write_lock() write_trylock() write_unlock() read_lock() read_unlock() read_unlock() imer_list() add_timer() del_timer() lkst_init_stage[0-1]() lkst_dump_notify_handler() lkst_evhandlerprim_maskset_xchg_inli lkst_evhandlerprim_buffer_shift_inline()	/kernel/timer.c /arch/i386/mm/fault.c /arch/i386/kernel/nmi.c /include/asm-i386/io.h /kernel/panic.c /kernel/printk.c /driver/lkst/lkst_core.c /driver/lkst/lkst_core.c /driver/lkst/lkst_core.c /driver/lkst/lkst_core.c	address where it was called address where it was called address where it was called address where it was called address where it was called function address(fin) pointer to timer list (timer) pointer to timer list (timer) pointer to timer list (timer) address where it was accessed address where it was running port address/byte width port address/byte width address of argument address of argument initialization status dump state old maskset ID old buffer ID	rwlock rwlock rwlock rwlock rwlock argument for the function(data) unexpired term (timer->expires) unexpired term (timer->expires) unexpired term (timer->expires) address where exception occurred value to output value to output value to input address where it was called address where it was called dump device new maskset ID	function address (timer->function) function address (timer->function) function address (timer->function) function address (timer->function) exception error code address where it was called address where it was called pointer to old maskset	argument for the function (timer- argument for the function (timer- argument for the function (timer-	inline inline define inline define inline define inline control to the set of the set o
84 85 86 87 a0 a1 a2 a3 a4 b0 b1 90 91 92 93 f00 f01 f08 f10 f11	Timer Oops Others	LK WRLOCK LK WRTRYLOCK LK WRUNLOCK LK RDUNLOCK LK RDUNLOCK IMBER RUN TIMER RUN TIMER ADD TIMER_DEL LKST_INIT LKST_KERNEL_DUMP LKST_MSET_XCHG LKST_BUFF_SHIFT LKST_BUFF_OVFLOW	run timer list add to timer list modify timer list delete from timer list with synchro oops in page fault handler oops in nmi watchdog timer io commands panic printk Progress of LKST initialization pr kernel dump event LKST switches the masksets LKST shifts the buffers overrun occurred in the current b	write lock write try lock (exit) write unlock read lock read unlock just before the oops operation just before the oops operation port output port input	write_lock() write_trylock() write_unlock() read_lock() read_unlock() read_unlock() irun_timer[ist() add_timer() del_timer() del_timer() del_timer() del_timer() del_timer() del_timer() ids_timer() del_timer() ids_timer() del_timer() ids_timer() ids_timer() ids_timer() ids_timer() ids_timer() ids_timer() ids_timer() ids_timer() ilkst_init_stage[0-1]() ilkst_dump_notify_handler() ilkst_evhandlerprim_maskset_xchg_init ikst_evhandlerprim_entry_next()	/kernel/timer.c /arch/i386/mm/fault.c /arch/i386/kernel/nmi.c /include/asm-i386/io.h /kernel/panic.c /kernel/printk.c /driver/lkst/lkst_core.c /driver/lkst/lkst_core.c /driver/lkst/lkst_core.c /driver/lkst/lkst_core.c	address where it was called address where it was called address where it was called address where it was called address where it was called function address(fn) pointer to timer list (timer) pointer to timer list (timer) pointer to timer list (timer) address where it was accessed address where it was accessed address where it was nunning port address/byte width port address/byte width address of argument address of argument address of argument old maskset ID old buffer ID pointer to the buffer	rwlock rwlock rwlock rwlock rwlock argument for the function(data) unexpired term (timer->expires) unexpired term (timer->expires) unexpired term (timer->expires) address where exception occurred value to output value to output value to input address where it was called address where it was called dump device new maskset ID	function address (timer->function) function address (timer->function) function address (timer->function) function address (timer->function) exception error code address where it was called address where it was called pointer to old maskset pointer to old buffer	argument for the function (timer- argument for the function (timer- argument for the function (timer-	Inline Inline Inline define Inline Idefine Inline Idefine Inline
84 85 86 87 a0 a1 a2 a3 a4 b0 91 92 93 f00 f01 f08 f10 f11 f19	Timer Oops Others	LK WRLOCK LK WRTRYLOCK LK WRTRYLOCK LK WRUNLOCK LK RDUNLOCK IK RDUNLOCK TIMER_RUN TIMER_ADD TIMER_ADD TIMER_DEL TIMER_DEL TIMER_DEL_SYNC OOPS_PGFAULT OOPS_NMWDOG O_PORTIN O_PORTOUT O_PANIC O_PRINTK LKST_KERNEL_DUMP LKST_KERNEL_DUMP LKST_BUFF_SHIFT LKST_BUFF_OVFLOW LKST_SYNC_UID	run timer list add to timer list modify timer list delete from timer list with synchro oops in page fault handler oops in nmi watchdog timer io commands panic printk Progress of LKST initialization pr kernel dump event LKST switches the masksets LKST shifts the buffers overrun occurred in the current b Synchronization with UID	write lock write try lock (exit) write unlock read lock read unlock just before the oops operation just before the oops operation port output port input	write_lock() write_unlock() read_lock() read_unlock() read_unlock() read_unlock() add_timer() add_timer() del_timer() del_timer() del_timer() idel_timer() del_timer() idel_timer() ikst_evhandlerprim_maskset_xchg_inli ikst_evhandlerprim_entry_next() sys_*uid(), set_user()	/kernel/timer.c /arch/i386/mm/fault.c /arch/i386/kernel/nmi.c /include/asm-i386/io.h /kernel/panic.c /kernel/printk.c /driver/lkst/lkst_core.c /driver/lkst/lkst_core.c /driver/lkst/lkst_core.c /driver/lkst/lkst_private.h /kernel/timer.c, sys.c	address where it was called address where it was called address where it was called address where it was called address where it was called function address(n) pointer to timer list (timer) pointer to timer list (timer) pointer to timer list (timer) address where it was accessed address where it was accessed address where it was nunning port address/byte width port address/byte width address of argument address of argument initialization status dump state old maskset ID old buffer ID	rwlock rwlock rwlock rwlock rwlock argument for the function(data) unexpired term (timer->expires) unexpired term (timer->expires) unexpired term (timer->expires) address where exception occurred value to output value to output value to input address where it was called address where it was called dump device new maskset ID	function address (timer->function) function address (timer->function) function address (timer->function) function address (timer->function) exception error code address where it was called address where it was called address where it was called pointer to old maskset pointer to old buffer	argument for the function (timer- argument for the function (timer- argument for the function (timer-	inline inline define inline define inline define This event is embeded in LKST. User can't handle it. Recorded 2 times; before/after Recorded 2 times; before/after ISecorded ISECO
84 85 86 87 a0 a1 a2 a3 a4 b0 b1 90 91 92 93 f00 f11 f19 f1a	Timer Oops Others	LK WRLOCK LK WRTRYLOCK LK WRUNLOCK LK WRUNLOCK LK RDUNLOCK TIMER_RUN TIMER_ADD TIMER_DEL TIMER_DEL TIMER_DEL SYNC OOPS_PGFAULT OOPS_NMIWDOG O_PORTIN O_PORTOUT O_PRINTK LKST_KERNEL_DUMP LKST_SHIFT LKST_BUFF_OVFLOW LKST_SYNC_UD LKST_SYNC_UD LKST_SYNC_GID	run timer list add to timer list modify timer list delete from timer list with synchro oops in page fault handler oops in nmi watchdog timer io commands panic printk Progress of LKST initialization pr kernel dump event LKST switches the masksets LKST shifts the buffers overrun occurred in the current b Synchronization with GID	write lock write try lock (exit) write unlock read lock read unlock just before the oops operation just before the oops operation port output port input	write_lock() write_trylock() write_unlock() read_lock() read_lock() read_unlock() imer_list() add_timer() del_timer() ids_totk()	/kernel/timer.c /arch/i386/mm/fault.c /arch/i386/kernel/nmi.c /include/asm-i386/io.h /kernel/pantc.c /kernel/printk.c /driver/lkst/lkst_core.c /driver/lkst/lkst_core.c /driver/lkst/lkst_core.c /driver/lkst/lkst_core.c /driver/lkst/lkst_core.c /driver/lkst/lkst_core.c /kernel/timer.c, sys.c	address where it was called address where it was called address where it was called address where it was called address where it was called function address(fn) pointer to timer list (timer) pointer to timer list (timer) pointer to timer list (timer) address where it was accessed address where it was running port address/byte width port address/byte width address of argument address of argument initialization status dump state old maskset ID old buffer ID pointer to the buffer UID GID	rwlock rwlock rwlock rwlock rwlock argument for the function(data) unexpired term (timer->expires) unexpired term (timer->expires) unexpired term (timer->expires) address where exception occurred value to output value to output value to input address where it was called address where it was called dump device new maskset ID	function address (timer->function) function address (timer->function) function address (timer->function) function address (timer->function) exception error code address where it was called address where it was called address where it was called pointer to old maskset pointer to old buffer	argument for the function (timer- argument for the function (timer- argument for the function (timer- argument for the function (timer- definition (timer- argument for the function (timer- definition (timer- argument for the function (timer- ar	inline inline define inline define inline define inline inline inline inline inline inline inline This event is embeded in LKST. User can't handle it. Recorded 2 times; before/after Used for automatically shifting buffer. If masked, LKST stops it. for compensation of dropped log data for compensation of dropped log data
84 85 86 87 a0 a1 a2 a3 a4 b0 b1 90 91 92 93 f00 f11 f19 f1a f1b	Timer Oops Others	LK WRLOCK LK WRTRYLOCK LK WRTRYLOCK LK WRUNLOCK LK RDUNLOCK IK RDUNLOCK TIMER_RUN TIMER_ADD TIMER_DEL TIMER_DEL TIMER_DEL TIMER_DEL TIMER_DELSYNC OOPS_PGFAULT OOPS_NMIWDOG O_PORTIN O_PORTOUT O_PORTOUT O_PANIC O_PORTOUT LKST_INIT LKST_KERNEL_DUMP LKST_SYNC_UID LKST_SYNC_DGID LKST_SYNC_PGID	run timer list add to timer list modify timer list delete from timer list with synchro oops in page fault handler oops in nmi watchdog timer io commands panic printk Progress of LKST initialization pr kernel dump event LKST switches the masksets LKST shifts the buffers overrun occurred in the current b Synchronization with GID Synchronization with PGID	write lock write try lock (exit) write unlock read lock read unlock just before the oops operation just before the oops operation port output port input	write_lock() write_trylock() write_unlock() read_lock() read_unlock() read_unlock() add_timer() add_timer() dde_timer() dde_timer() dde_timer() dde_timer() dde_timer() ids_page_fault() nnmi_watchdog_tick() OUT() or between _OUT1() and _OUT2() itail ofIN() ikst_dump_notify_handler() ikst_dump_notify_handler() ikst_evhandlerprim_maskset_xchg_inii ikst_evhandlerprim_entry_next() sys_*uid(), set_user() sys_*gid()	/kernel/timer.c /arch/i386/mm/fault.c /arch/i386/kernel/nmi.c /include/asm-i386/io.h /kernel/panic.c /kernel/pintk.c /driver/lkst/lkst_core.c /driver/lkst/lkst_core.c /driver/lkst/lkst_core.c /driver/lkst/lkst_core.c /driver/lkst/lkst_core.c /kernel/timer.c, sys.c /kernel/timer.c, sys.c	address where it was called address where it was called address where it was called address where it was called address where it was called function address(fn) pointer to timer list (timer) pointer to timer list (timer) pointer to timer list (timer) address where it was accessed address where it was accessed address where it was nunning port address/byte width port address/byte width address of argument address of argument address of argument old maskset ID old buffer ID pointer to the buffer UID GID PID	rwlock rwlock rwlock rwlock rwlock argument for the function(data) unexpired term (timer->expires) unexpired term (timer->expires) unexpired term (timer->expires) address where exception occurred value to output value to output value to output value to input address where it was called address where it was called dump device new maskset ID new buffer ID	function address (timer->function) function address (timer->function) function address (timer->function) function address (timer->function) exception error code address where it was called address where it was called address where it was called pointer to old maskset pointer to old buffer pointer to the process table pointer to the process table pointer to the process table	argument for the function (timer- argument for the function (timer- argument for the function (timer-	Inline Inline Inline define Inline Inline Idefine Inline Idefine Inline
84 85 86 87 a0 a1 a2 a3 a4 b0 91 92 93 f00 f01 f08 f10 f11 f19 f1b f1c	Timer Oops Others	LK WRLOCK LK WRTRYLOCK LK WRUNLOCK LK RDUNLOCK LK RDUNLOCK TIMER_RUN TIMER_ADD TIMER_DEL TIMER_DEL TIMER_DEL SYNC OOPS_PGFAULT OOPS_NMWDOG O_PORTIN O_PORTOUT O_PORTOUT O_PORTOUT O_PORTOUT O_PORTOUT O_PORTOUT O_PORTOUT O_PORTOUT UKST_SYNC_DID LKST_SYNC_PGID LKST_SYNC_TID	run timer list add to timer list modify timer list delete from timer list with synchro oops in page fault handler oops in nmi watchdog timer io commands panic printk Progress of LKST initialization pr kernel dump event LKST switches the masksets LKST shifts the buffers overrun occurred in the current b Synchronization with GID Synchronization with TID	write lock write try lock (exit) write unlock read lock read unlock just before the oops operation just before the oops operation port output port input	write_lock() write_trylock() write_unlock() read_lock() read_lock() read_unlock() imer_list() add_timer() del_timer() ids_totk()	/kernel/timer.c /arch/i386/mm/fault.c /arch/i386/kernel/nmi.c /include/asm-i386/io.h /kernel/pantc.c /kernel/printk.c /driver/lkst/lkst_core.c /driver/lkst/lkst_core.c /driver/lkst/lkst_core.c /driver/lkst/lkst_core.c /driver/lkst/lkst_core.c /driver/lkst/lkst_core.c /kernel/timer.c, sys.c	address where it was called address where it was called address where it was called address where it was called address where it was called function address(fn) pointer to timer list (timer) pointer to timer list (timer) pointer to timer list (timer) address where it was accessed address where it was accessed address where it was running port address/byte width address of argument address of argument initialization status dump state old maskset ID old buffer ID pointer to the buffer UID GID PID TID(pid)	rwlock rwlock rwlock rwlock rwlock argument for the function(data) unexpired term (timer->expires) unexpired term (timer->expires) unexpired term (timer->expires) address where exception occurred value to output value to output address where it was called address where it was called dump device new maskset ID new buffer ID PGRP	function address (timer->function) function address (timer->function) function address (timer->function) function address (timer->function) exception error code address where it was called address where it was called address where it was called pointer to old maskset pointer to old buffer pointer to the process table pointer to the process table pointer to the process table pointer to the process table pointer to the process table	argument for the function (timer- argument for the function (timer- argument for the function (timer- argument for the function (timer- definition (timer- argument for the function (timer- be) pointer to new maskset pointer to new buffer	inline inline define inline define inline define inline inline inline inline inline inline inline This event is embeded in LKST. User can't handle it. Recorded 2 times; before/after Used for automatically shifting buffer. If masked, LKST stops it. for compensation of dropped log data for compensation of dropped log data
84 85 86 87 a0 a1 a2 a3 a4 b0 b1 90 91 92 93 f00 f01 f08 f10 f11 f19 f1a f1b f1c f1c	Timer Oops Others	LK WRLOCK LK WRTRYLOCK LK WRUNLOCK LK WRUNLOCK LK RDUNLOCK TIMER RUN TIMER ADD TIMER_MOD TIMER_DEL TIMER_DEL TIMER_DEL OOPS_NMIWDOG O_PORTIN O_PORTOUT O_PORTOUT O_PRINTK LKST_KERNEL_DUMP LKST_STNC_TID LKST_SYNC_GID LKST_SYNC_GID LKST_SYNC_GID LKST_SYNC_TID LKST_EXTEND	run timer list add to timer list modify timer list delete from timer list delete from timer list with synchro oops in page fault handler oops in nmi watchdog timer io commands panic printk Progress of LKST initialization pr kernel dump event LKST switches the masksets LKST shifts the buffers overrun occurred in the current b Synchronization with UID Synchronization with UID Synchronization with UID Synchronization with UID Synchronization with UID Synchronization with UID Extra arguments	write lock write try lock (exit) write unlock read lock read unlock just before the oops operation just before the oops operation port output port input	write_lock() write_trylock() write_unlock() read_lock() read_unlock() read_unlock() add_timer() add_timer() dde_timer() dde_timer() dde_timer() dde_timer() dde_timer() ids_page_fault() nnmi_watchdog_tick() OUT() or between _OUT1() and _OUT2() itail ofIN() ikst_dump_notify_handler() ikst_dump_notify_handler() ikst_evhandlerprim_maskset_xchg_inii ikst_evhandlerprim_entry_next() sys_*uid(), set_user() sys_*gid()	/kernel/timer.c /arch/i386/mm/fault.c /arch/i386/kernel/nmi.c /include/asm-i386/io.h /kernel/panic.c /kernel/pintk.c /driver/lkst/lkst_core.c /driver/lkst/lkst_core.c /driver/lkst/lkst_core.c /driver/lkst/lkst_core.c /driver/lkst/lkst_core.c /kernel/timer.c, sys.c /kernel/timer.c, sys.c	address where it was called address where it was called address where it was called address where it was called address where it was called function address(fn) pointer to timer list (timer) pointer to timer list (timer) pointer to timer list (timer) address where it was accessed address where it was accessed address where it was accessed address where it was running port address/byte width port address/byte width address of argument address of argument initialization status dump state old maskset ID old buffer ID pointer to the buffer UID GID PID TID(pid) extra argument 1	rwlock rwlock rwlock rwlock rwlock rwlock argument for the function(data) unexpired term (timer->expires) unexpired term (timer->expires) unexpired term (timer->expires) address where exception occurred value to output value to output value to input address where it was called address where it was called dump device new maskset ID new buffer ID PGRP extra argument 2	function address (timer->function) function address (timer->function) function address (timer->function) function address (timer->function) exception error code address where it was called address where it was called address where it was called pointer to old maskset pointer to old buffer pointer to the process table pointer to the process table	argument for the function (timer- argument for the function (timer- argument for the function (timer- argument for the function (timer- definition (timer- argument for the function (timer- argument for the func	Inline Inline Inline define Inline Inline Idefine Inline Idefine Inline
84 85 86 87 a0 a1 a2 a3 a4 b0 b1 90 91 92 93 f00 f01 f08 f10 f11 f19 f1a f1b f1c ffc ffd	Timer Oops Others	LK WRLOCK LK WRTRYLOCK LK WRUNLOCK LK WRUNLOCK LK RDUNLOCK TIMER_RUN TIMER_ADD TIMER_DEL TIMER_DEL_SYNC OOPS_PGFAULT OOPS_NMIWDOG O_PORTIN O_PORTOUT O_PRINTK LKST_INIT LKST_KERNEL_DUMP LKST_SYNC_TID LKST_SYNC_DID LKST_SYNC_TID LKST_SYNC_PGID LKST_EXTENDE	run timer list add to timer list modify timer list delete from timer list with synchro oops in page fault handler oops in nmi watchdog timer io commands panic printk Progress of LKST initialization pr kernel dump event LKST switches the masksets LKST shifts the buffers overrun occurred in the current b Synchronization with GID Synchronization with GID Synchronization with TID Extra arguments	write lock write unlock tread lock read lock read unlock read unlock read unlock issue the oops operation just before the oops operation port output port input occess uffer.	write_lock() write_trylock() write_unlock() read_unlock() read_unlock() imer_lisit() add_timer() del_timer_sync() del_timer() likst_ent_stage[0-1]() likst_evhandlerprim_maskset_xchg_inli likst_evhandlerprim_buffer_shift_inline(likst_evhandlerprim_entry_next() sys_*gid() sys_*gid() sys_gettid()	/kernel/timer.c /arch/i386/mm/fault.c /arch/i386/kernel/nmi.c /include/asm-i386/io.h /kernel/panic.c /kernel/printk.c /driver/lkst/lkst_core.c /driver/lkst/lkst_core.c /driver/lkst/lkst_core.c /driver/lkst/lkst_core.c /driver/lkst/lkst_core.c /driver/lkst/lkst_core.c /driver/lkst/lkst_core.c /driver/lkst/lkst_core.c /driver/lkst/lkst_core.c /driver/lkst/lkst_core.c /driver/lkst/lkst_core.c /driver/lkst/lkst_core.c /driver/lkst/lkst_core.c /driver/lkst/lkst_core.c /driver/lkst/lkst_core.c	address where it was called address where it was called address where it was called address where it was called address where it was called function address(fn) pointer to timer list (timer) pointer to timer list (timer) pointer to timer list (timer) address where it was accessed address where it was accessed address where it was running port address/byte width address of argument address of argument initialization status dump state old maskset ID old buffer ID pointer to the buffer UID GID PID TID(pid)	rwlock rwlock rwlock rwlock rwlock argument for the function(data) unexpired term (timer->expires) unexpired term (timer->expires) unexpired term (timer->expires) address where exception occurred value to output value to output address where it was called address where it was called dump device new maskset ID new buffer ID PGRP	function address (timer->function) function address (timer->function) function address (timer->function) function address (timer->function) exception error code address where it was called address where it was called address where it was called pointer to old maskset pointer to old buffer pointer to the process table pointer to the process table pointer to the process table pointer to the process table pointer to the process table	argument for the function (timer- argument for the function (timer- argument for the function (timer- argument for the function (timer- definition (timer- argument for the function (timer- be) pointer to new maskset pointer to new buffer	Inline Inline Inline define Inline define Inline In
84 85 86 87 a0 a1 a2 a3 a4 b0 91 92 93 f00 f11 f19 f1a f1b f1c ffc ffd ffe	Timer Oops Others	LK WRLOCK LK WRTRYLOCK LK WRUNLOCK LK RDUNLOCK LK RDUNLOCK TIMER RUN TIMER RUN TIMER DEL TIMER DEL TI	run timer list add to timer list modify timer list delete from timer list with synchro oops in page fault handler oops in nmi watchdog timer io commands panic printk Progress of LKST initialization pr kernel dump event LKST switches the masksets LKST shifts the buffers overrun occurred in the current b Synchronization with UID Synchronization with GID Synchronization with PGID Synchronization with PGID Synchronization with TID Extra arguments End of Extra arguments Overwritten occured while readin	write lock write unlock tread lock read lock read unlock read unlock read unlock issue the oops operation just before the oops operation port output port input occess uffer.	write_lock() write_trylock() write_unlock() read_lock() read_unlock() read_unlock() add_timer() add_timer() dde_timer() dde_timer() dde_timer() dde_timer() dde_timer() ids_page_fault() nnmi_watchdog_tick() OUT() or between _OUT1() and _OUT2() itail ofIN() ikst_dump_notify_handler() ikst_dump_notify_handler() ikst_evhandlerprim_maskset_xchg_inii ikst_evhandlerprim_entry_next() sys_*uid(), set_user() sys_*gid()	/kernel/timer.c /arch/i386/mm/fault.c /arch/i386/kernel/nmi.c /include/asm-i386/io.h /kernel/panic.c /kernel/pintk.c /driver/lkst/lkst_core.c /driver/lkst/lkst_core.c /driver/lkst/lkst_core.c /driver/lkst/lkst_core.c /driver/lkst/lkst_core.c /kernel/timer.c, sys.c /kernel/timer.c, sys.c	address where it was called address where it was called address where it was called address where it was called address where it was called function address(fn) pointer to timer list (timer) pointer to timer list (timer) pointer to timer list (timer) address where it was accessed address where it was accessed address where it was accessed address where it was running port address/byte width port address/byte width address of argument address of argument initialization status dump state old maskset ID old buffer ID pointer to the buffer UID GID PID TID(pid) extra argument 1	rwlock rwlock rwlock rwlock rwlock rwlock argument for the function(data) unexpired term (timer->expires) unexpired term (timer->expires) unexpired term (timer->expires) address where exception occurred value to output value to output value to input address where it was called address where it was called dump device new maskset ID new buffer ID PGRP extra argument 2	function address (timer->function) function address (timer->function) function address (timer->function) function address (timer->function) exception error code address where it was called address where it was called address where it was called pointer to old maskset pointer to old buffer pointer to the process table pointer to the process table	argument for the function (timer- argument for the function (timer- argument for the function (timer- argument for the function (timer- definition (timer- argument for the function (timer- argument for the func	Inline Inline Inline Inline define Inline Inline Idefine Inline Idefine Inline
84 85 86 87 a0 a1 a2 a3 a4 b0 b1 90 91 92 93 f00 f01 f08 f10 f11 f19 f1a f1b f1c ffc ffd	Timer Oops Others	LK WRLOCK LK WRTRYLOCK LK WRUNLOCK LK WRUNLOCK LK RDUNLOCK TIMER_RUN TIMER_ADD TIMER_DEL TIMER_DEL_SYNC OOPS_PGFAULT OOPS_NMIWDOG O_PORTIN O_PORTOUT O_PRINTK LKST_INIT LKST_KERNEL_DUMP LKST_SYNC_TID LKST_SYNC_DID LKST_SYNC_TID LKST_SYNC_PGID LKST_EXTENDE	run timer list add to timer list modify timer list delete from timer list with synchro oops in page fault handler oops in nmi watchdog timer io commands panic printk Progress of LKST initialization pr kernel dump event LKST switches the masksets LKST shifts the buffers overrun occurred in the current b Synchronization with GID Synchronization with GID Synchronization with TID Extra arguments	write lock write unlock tread lock read lock read unlock read unlock read unlock issue the oops operation just before the oops operation port output port input occess uffer.	write_lock() write_trylock() write_unlock() read_unlock() read_unlock() imer_lisit() add_timer() del_timer_sync() del_timer() likst_ent_stage[0-1]() likst_evhandlerprim_maskset_xchg_inli likst_evhandlerprim_buffer_shift_inline(likst_evhandlerprim_entry_next() sys_*gid() sys_*gid() sys_gettid()	/kernel/timer.c /arch/i386/mm/fault.c /arch/i386/kernel/nmi.c /include/asm-i386/io.h /kernel/panic.c /kernel/printk.c /driver/lkst/lkst_core.c /driver/lkst/lkst_core.c /driver/lkst/lkst_core.c /driver/lkst/lkst_core.c /driver/lkst/lkst_core.c /driver/lkst/lkst_core.c /driver/lkst/lkst_core.c /driver/lkst/lkst_core.c /driver/lkst/lkst_core.c /driver/lkst/lkst_core.c /driver/lkst/lkst_core.c /driver/lkst/lkst_core.c /driver/lkst/lkst_core.c /driver/lkst/lkst_core.c /driver/lkst/lkst_core.c	address where it was called address where it was called address where it was called address where it was called address where it was called function address(fn) pointer to timer list (timer) pointer to timer list (timer) pointer to timer list (timer) address where it was accessed address where it was accessed address where it was accessed address where it was running port address/byte width port address/byte width address of argument address of argument initialization status dump state old maskset ID old buffer ID pointer to the buffer UID GID PID TID(pid) extra argument 1	rwlock rwlock rwlock rwlock rwlock rwlock argument for the function(data) unexpired term (timer->expires) unexpired term (timer->expires) unexpired term (timer->expires) address where exception occurred value to output value to output value to input address where it was called address where it was called dump device new maskset ID new buffer ID PGRP extra argument 2	function address (timer->function) function address (timer->function) function address (timer->function) function address (timer->function) exception error code address where it was called address where it was called address where it was called pointer to old maskset pointer to old buffer pointer to the process table pointer to the process table	argument for the function (timer- argument for the function (timer- argument for the function (timer- argument for the function (timer- definition (timer- argument for the function (timer- argument for the func	Inline Inline Inline define Inline In

Event type [hex]											
	Version 2.02 Categoly	Mnemonic	Descript	ion of events	where to hook	filename	data recorded as "log_arg1"	data recorded as "log_arg2"	Copyright (data recorded as "log_arg3"	C) Hitachi, Ltd., 2002-2003. All r data recorded as "log_arg4"	rights reserved. remarks
01		PROCESS CONTEXTSWITC			schedule()	./kernel/sched.c	address of the task_struct	address of the task_struct	prev. process state (value after	prev. process count (value before	from log_arg3, log_arg4, can determain
02	Process management	PROCESS_WAKEUP WAKEUP			try to wake up()	-	of "prev" value of "p" in the function	of "next" synchronous	switch)	switch)	why processes were switched
03	management	PROCESS_SIGSEND	sending signal		specific_send_sig_info()	./kernel/signal.c	value of "sig" in the function value of "fn" in the function	value of "t" in the function pointer to argument of kernel thread	pointer to info (info)		
04 10		INT_HARDWARE_ENTRY	creating a kernel thread hardware entrance		kernel_thread() do_IRQ()	./arch/ia64/kernel/process.c ./arch/ia64/kernel/irg.c	value of "irq" in the function	interrupt status (status)	tlag pointer to register stack		
12 14	Interrupts	INT_TASKLETHI_ENTRY	activara	entrance	tasklet_hi_action()	./kernel/softirq.c	value of "t->func" in the function value of "t->func" in the function				
14		INT_TASKLET_ENTRY		entrance entrance	tasklet_action() bh_action()		value of "r->runc" in the function	address of action (bh_base)			
20		EXCEPT_PGFLT_ENTRY	vhpt_miss	entrance							
20		EXCEPT_PGFLT_ENTRY	tlb_miss htlb_miss	entrance		lanah lia C Alanan Karulta			·		
			alt_itlb_miss alt_dtlb_miss nested dtlb_miss	exit	—ia64_do_page_fault()	./arch/ia64/mm/fault.c	fault address(ifa)	isr	ipsr	lip	
21		EXCEPT_PGFLT_EXIT									
22		EXCEPT_ILLOP_ENTRY		entrance	ia64_illegal_op_fault()		ec			iip	
23 24	-	EXCEPT_ILLOP_EAT		exit entrance		-					
25		EXCEPT_BADBRK_EXIT	bleak_Instruction	exit	ia64_bad_break()		break number(iim)		ipsr	lip	
			general_exception disabled_fp_reg	entrance	ia64_fault()		fault vector number				
			instruction_key_miss								
26	Exceptions	EXCEPT_FAULT_ENTRY									
			debug_vector			./arch/ia64/kernel/traps.c					
			unsupported_data_reference fp_fault					i§r	ipsr	iip	
			fp_trap	exit							
			lower_privilege_transfer_trap taken branch trap								
27		Single_step_trap	single_step_trap								
			ia32_exception ia32_intercept								
		ia32_i	ia32_interrupt								
28	-										
29		EXCEPT_UNALIGN_EXIT	unalighed_access	exit	- ia64_handle_unaligned()	./arch/ia64/kernel/unaligned.c	ITA		ipsr	пр	
<u> </u>	System calls	SYSCALL_ENTRY SYSCALL EXIT	entrance exit		beginning of system_call() ending of system_call()	./arch/ia64/kernel/ivt.S	system call function address system call function address	the number of this system call errno	-		recording arguments of system calls is optional feature
50	_	MEM_SWAPOUT	swap out	exit	try_to_swap_out()	./mm/vmscan.c	pointer to page swapped out (page)				
<u>51</u> 52	-	MEM_SWAPIN MEM_DO_NOPAGE	swap in mem_do_nopage	exit exit	do_swap_page() do_no_page()	./mm/memory.c ./mm/memory.c	pointer to page swapped in (page) pointer to page allocated (new_page)				
53		MEM_DO_WPPAGE	mem_do_wppage		do_wp_page()	./mm/memory.c	pointer to page (new page)				
<u>54</u> 55	Memory	MEM_WAIT_PAGE MEM_GET_FREEPAGE	mem_wait_page mem_get_freepage	entrance exit	wait_on_page() get_free_page()	./mm/filemap.c ./mm/page_alloc.c	pointer to page (page) pointer to page (paddr)	type of page (gfp_mask)	the number of page (order)	call address	
56		MEM_GET_ZEROPAGE	mem_get_zeropage	exit	get_zeroed_page()	./mm/page_alloc.c	pointer to page (address)	type of page (gfp_mask)	call address		
<u> </u>	Management	MEM_FREEPAGE MEM_VMALLOC	mem_freepage mem_vmalloc	entrance exit	free_pages() vmalloc()	./mm/page_alloc.c ./mm/vmalloc.h	pointer to (addr) address (addr)	the number of page (order) size	call address call address		
59		MEM_VFREE	mem_vfree	entrance	vfree()	./mm/vmalloc.c	address (addr)				
5a 5b	_	MEM_CACHE_CREATE MEM_CACHE_ALLOC	mem_cache_create mem_cache_alloc	exit exit	kmem_cache_create() kmem_cache_alloc()	./mm/slab.c ./mm/slab.c	name cachep	size flags	cachep obip	call address	
50 50		MEM_MALLOC	mem_malloc	exit	kmalloc()	./mm/slab.c	cachep	flags	objp	call address	
5d 5e	_	MEM_CACHE_FREE MEM_FREE	mem_cache_free mem free	entrance entrance	kmem_cache_free() kfree()	./mm/slab.c ./mm/slab.c	cachep objp	objp call address	call address		
60	<u> </u>	NET_PKTSEND	sending packets	entrance	dev_queue_xmit()	./net/core/dev.c	skb				
61 62		NET_PKTSENDI NET_PKTRECV	interrupt on sending packets receiving packets	entrance entrance	net_tx_action() netif_rx()	./net/core/dev.c ./net/core/dev.c	h skb				
63	Retworking	NET PKTRECVI	interrupt on receiving packets	entrance	net_rx_action()	./net/core/dev.c	h				
<u>64</u> 70		NET_SOCKETIF SYSV_IPC_SEMOP	socket()	entrance	sys_socketcall sys_semop()	./net/socket.c ./ipc/sem.c	call semid	args tsops	nsops		exit is recorded as exit of system call.
71		SYSV_IPC_SEMGET			sys_semget()		key	nsems	semflg		
72 73		SYSV_IPC_SEMCTL			sys_semctl()						
74		SVSV IPC MSGSEND			svs_msgsend()		semid	semnum	cmd msgsz	argument for the function	
		SYSV_IPC_MSGSEND SYSV_IPC_MSGRCV	-		sys_msgsend() sys_msgrcv()	line/mea.c	semid msqid msqid/msgflg	msgp msgp	cmd msgsz msgsz	argument for the function msgflg msgtyp	
75	SysV IPC	SYSV_IPC_MSGRCV SYSV_IPC_MSGGET	IPC functions	entrance	sys_msgrcv() sys_msgget()	./ipc/msg.c	msqid msqid/msgflg key	msgp msgp msgflg	msgsz msgsz	msgflg	
	SysV IPC	SYSV_IPC_MSGRCV	IPC functions	entrance	sys_msgrcv()	./ipc/msg.c	msqid msqid/msgflg	msgp msgp	msgsz	msgflg	
75 76 77 78	SysV IPC	SYSV_IPC_MSGRCV SYSV_IPC_MSGGET SYSV_IPC_MSGCTL SYSV_IPC_SHMAT SYSV_IPC_SHMDT	IPC functions	entrance	sys_msgcv() sys_msgget() sys_msgctl() sys_shmat() sys_shmat()	./ipc/msg.c ./ipc/shm.c	msqid msqid/msgfig key msqid shmid shmaddr	msgp msgp msgfig cmd shmaddr	msgsz msgsz buf shmfig	msgflg	
75 76 77 78 79 7a	SysV IPC	SYSV_IPC_MSGRCV SYSV_IPC_MSGGET SYSV_IPC_MSGCTL SYSV_IPC_SHMAT SYSV_IPC_SHMDT SYSV_IPC_SHMGET SYSV_IPC_SHMGET	IPC functions	entrance	sys_msgcv() sys_msgget() sys_msgct() sys_ssct() sys_shmat() sys_shmdt() sys_shmgt() sys_shmgt()		msqid msqid/msgfig key msqid shmid shmaddr key shmid	msgp msgp cmd shmaddr size cmd	msgsz msgsz	msgflg	
75 76 77 78 79 79 7a 80	SysV IPC	SYSV. IPC MSGRCV SYSV. IPC MSGET SYSV. IPC MSGCTL SYSV. IPC SHMAT SYSV. IPC SHMDT SYSV. IPC SHMGET SYSV. IPC SHMGET SYSV. IPC SHMCTL LK, SPINLOCK		lock	sys_msgrcv() sys_msgget() sys_msgctl() sys_shmat() sys_shmat() sys_shmat() sys_shmat() sys_shmat() sys_shmat() sys_shmat() sys_shmat()		msqid msqid/msqfig key msqid shmid shmid shmid key shmid address where it was called	msgp msgp cmd shmaddr size cmd lock	msgsz msgsz buf shmflg shmflg buf	msgflg	inline
75 76 77 78 79 7a 80 81 82	SysV IPC	SYSV_IPC_MSGRCV SYSV_IPC_MSGGET SYSV_IPC_MSGCTL SYSV_IPC_SHMAT SYSV_IPC_SHMDT SYSV_IPC_SHMGET SYSV_IPC_SHMCTL LK_SPINLOCK LK_SPINLOCK LK_SPINUNLOCK	IPC functions	lock try lock (exit) unlock	sys_msgccv() sys_msgget() sys_msgct() sys_msgct() sys_shmat() spin_ulock() spin_ulock()		msqid msqid/msgfig key msqid shmid shmid shmid shmid address where it was called address where it was called address where it was called	msgp msgp msgfig cmd shmaddr size cmd lock lock lock lock	msgsz msgsz buf shmflg shmflg	msgflg	inline inline
75 76 77 78 79 7a 80 81 82 83	SysV IPC	SYSV. IPC MSGRCV SYSV. IPC MSGGET SYSV. IPC MSGCTL SYSV. IPC SHMAT SYSV. IPC SHMDT SYSV. IPC SHMGET SYSV. IPC SHMGET SYSV. IPC SHMGTL LK SPINLOCK LK SPINLOCK LK SPINUNLOCK LK WRLOCK		lock try lock (exit) unlock write lock	sys_msgrcv() sys_msggel() sys_msgcll() sys_shmat() sys_shmat() sys_shmgel() sys_shmgel() sys_shmgel() sys_shmgel() sys_shmgel() sys_shmgel() sys_shmgel() sys_shmgel() sys_obsch() spin_lock() spin_unlock() write_lock()	/ipc/shm.c	msqid msqid/msgfig key msqid shmid shmid shmaddr key shmid address where it was called address where it was called address where it was called address where it was called	msgp msgp msgflg cmd shmaddr size cmd lock lock lock lock lock rvlock	msgsz msgsz buf shmflg shmflg buf return value	msgflg	inline inline inline
75 76 77 78 79 7a 80 81 82		SYSV. IPC MSGRCV SYSV. IPC MSGGET SYSV. IPC MSGCTL SYSV. IPC SHMAT SYSV. IPC SHMDT SYSV. IPC SHMDT SYSV. IPC SHMDT LK SPINLOCK LK SPINLOCK LK WRLOCK LK WRINLOCK LK WRTRYLOCK		lock try lock (exit) unlock write lock write try lock (exit) write unlock	sys_msgrcv() sys_msgget() sys_msgget() sys_msgctl() sys_shmat() spin_unlock() write_lock() write_unlock() write_unlock()	/ipc/shm.c	msqid msqid/msqfig key msqid shmid shmid address where it was called address where it was called	msgp msgp msgfig cmd shmaddr size cmd lock lock lock lock rwlock rwlock rwlock	msgsz msgsz buf shmflg shmflg buf	msgflg	inline inline inline inline define
75 76 77 78 79 7a 80 81 82 83 83 84 85 86		SYSV IPC MSGRCV SYSV IPC MSGGET SYSV IPC MSGCTL SYSV IPC SHMAT SYSV IPC SHMDT SYSV IPC SHMDT SYSV IPC SHMCTL LK SPINLOCK LK SPINLOCK LK WRINLOCK LK WRTRYLOCK LK WRUNLOCK LK WRUNLOCK LK WRUNLOCK	spin lock	lock try lock (exit) unlock write lock write try lock (exit) write unlock read lock	sys_msgrcv() sys_msgget() sys_msgget() sys_msgct() sys_shmat() sys_shmget() sys_shmget() <td>/ipc/shm.c</td> <td>msqid msqid/msgfig key msqid shmid shmid shmid address where it was called address where it was called</td> <td>msgp msgflg cmd shmaddr size cmd lock lock lock lock rwlock rwlock rwlock rwlock rwlock rwlock</td> <td>msgsz msgsz buf shmflg shmflg buf return value</td> <td>msgflg</td> <td>inline inline inline inline define inline</td>	/ipc/shm.c	msqid msqid/msgfig key msqid shmid shmid shmid address where it was called address where it was called	msgp msgflg cmd shmaddr size cmd lock lock lock lock rwlock rwlock rwlock rwlock rwlock rwlock	msgsz msgsz buf shmflg shmflg buf return value	msgflg	inline inline inline inline define inline
75 76 77 78 79 7a 80 81 82 83 84 85 85 86 87 a0		SYSV. IPC MSGRCV SYSV. IPC MSGGET SYSV. IPC SMGCTL SYSV. IPC SHMAT SYSV. IPC SHMDT SYSV. IPC SHMGET SYSV. IPC SHMGTL LK SPINLOCK LK SPINLOCK LK WRITRYLOCK LK WRITRYLOCK LK WRIDNLOCK LK WRIDNLOCK LK RDLOCK LK RDLOCK LK RDUNLOCK LK RDUNLOCK LK RDUNLOCK	spin lock read/write lock run timer list	lock try lock (exit) unlock write lock write try lock (exit) write unlock	sys_msgrcv() sys_msgget() sys_msgget() sys_msgget() sys_shmat() sys_lock() spin_lock() write_lock() write_lock() read_lock() read_lock() read_unlock() trun_timer_list()	/ipc/shm.c	msqid msqid/msqfig key msqid shmid shmid shmid address where it was called address where it was called function address(fn)	msgp msgp msgfig cmd shmaddr size cmd lock lock lock lock rwlock rwlock rwlock rwlock rwlock rwlock rwlock rwlock rwlock rwlock rwlock rwlock rwlock	msgsz msgsz buf shmflg shmflg buf return value return value	msgflg msgtyp raddr	inline inline inline inline define
75 76 77 78 79 7a 80 81 82 83 84 85 86 86 87 a0 a1	Locks	SYSV_IPC_MSGRCV SYSV_IPC_MSGGET SYSV_IPC_MSGCTL SYSV_IPC_SHMAT SYSV_IPC_SHMDT SYSV_IPC_SHMCTL LK_SPINLOCK LK_SPINUNLOCK LK_WRINLOCK LK_WRUNLOCK LK_WRUNLOCK LK_RDUNLOCK LK_RDUNLOCK LK_RDUNLOCK TIMER_RUN TIMER_ADD	spin lock read/write lock run timer list add to timer list	lock try lock (exit) unlock write lock write try lock (exit) write unlock read lock	sys_msgrcv() sys_msgget() sys_msgget() sys_msgget() sys_msgct() sys_shmat() sys_shmget() sys_shmget() <td>/ipc/shm.c</td> <td>msqid msqid/msgfig key msqid shmid shmid shmid address where it was called address (n) pointer to timer list (timer)</td> <td>msgp msgflg cmd shmaddr size cmd lock lock lock vick wick rwlock rwlock >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>></td> <td>msgsz msgsz buf shmflg buf return value return value return value function address (timer->function)</td> <td>msglyp msglyp raddr raddr raddr raddr raddr raddr raddr</td> <td>inline inline inline inline define inline</td>	/ipc/shm.c	msqid msqid/msgfig key msqid shmid shmid shmid address where it was called address (n) pointer to timer list (timer)	msgp msgflg cmd shmaddr size cmd lock lock lock vick wick rwlock rwlock >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>	msgsz msgsz buf shmflg buf return value return value return value function address (timer->function)	msglyp msglyp raddr raddr raddr raddr raddr raddr raddr	inline inline inline inline define inline
75 76 77 78 79 7a 80 81 82 83 84 85 85 86 85 86 87 a0 a1 a2 a3		SYSV_IPC_MSGRCV SYSV_IPC_MSGGET SYSV_IPC_MSGCTL SYSV_IPC_SHMAT SYSV_IPC_SHMDT SYSV_IPC_SHMDT SYSV_IPC_SHMCTL LK_SPINLOCK LK_SPINLOCK LK_SPINLOCK LK_WRUNLOCK LK_WRUNLOCK LK_WRUNLOCK LK_WRUNLOCK LK_RDUNLOCK LK_RDUNLOCK LK_RDUNLOCK LK_RDUNLOCK LK_RDUNLOCK IMER_RUN TIMER_ADD TIMER_MOD TIMER_MOD TIMER_MOL	spin lock read/write lock run timer list add to timer list modify timer list delete from timer list	lock try lock (exit) unlock write lock write unlock read lock read lock read unlock	sys_msgrcv() sys_msggel() sys_msggel() sys_msgcl() sys_shmat() sys_	/ipc/shm.c	msqid msqid/msqfig key msqid shmid shmid shmid address where it was called address (hor it was called)	msgp msgp msgfig cmd shmaddr size cmd lock lock lock lock lock rw	msgsz msgsz buf shmflg shmflg buf return value return value return value function address (timer-sfunction) function address (timer-sfunction) function address (timer-sfunction)	msgflg msgtyp raddr addr argument for the function (timer- argument for the function (timer- argument for the function (timer- argument for the function (timer-	inline inline inline inline define inline
75 76 77 78 80 81 82 83 84 85 86 87 a0 a1 a2	Locks	SYSV. IPC MSGRCV SYSV. IPC MSGGET SYSV. IPC MSGCTL SYSV. IPC SHMAT SYSV. IPC SHMGET SYSV. IPC SHMGET SYSV. IPC SHMGET LK SPINLOCK LK SPINLOCK LK WRILOCK LK WRILOCK LK WRILOCK LK WRILOCK LK RDLOCK LK RDLOCK LK RDLOCK TIMER_RUN TIMER_ADD TIMER_MOD	spin lock read/write lock run timer list add to timer list modify timer list	lock try lock (exit) unlock write lock write unlock read lock read lock read unlock	sys_msgrcv() sys_msgget() sys_msgget() sys_msgget() sys_msgget() sys_shmat() sys_shmget() write_lock() read_unlock() read_unlock() read_unlock() read_unlock() mod_timer()	/ipc/shm.c	msqid msqid/msqfig key msqid shmid shmid address where it was called address (n) pointer to timer list (timer) pointer to timer list (timer)	msgp msgp msgfig cmd shmaddr size cmd lock lock lock lock rwlock rwlock rwlock rwlock rwlock rwlock rwlock argument for the function(data) unexpired term (timer-sexpires)	msgsz msgsz buf shmflg buf return value return value function address (timer->function) function address (timer->function)	msgflg msgtyp raddr addr argument for the function (timer- argument for the function (timer- argument for the function (timer- argument for the function (timer-	inline inline inline inline define inline
75 76 77 78 79 7a 80 81 82 83 84 85 85 86 85 86 87 a0 a1 a2 a3	Locks	SYSV_IPC_MSGRCV SYSV_IPC_MSGGET SYSV_IPC_MSGCTL SYSV_IPC_SHMAT SYSV_IPC_SHMDT SYSV_IPC_SHMDT SYSV_IPC_SHMCTL LK_SPINLOCK LK_SPINLOCK LK_SPINLOCK LK_WRUNLOCK LK_WRUNLOCK LK_WRUNLOCK LK_WRUNLOCK LK_RDUNLOCK LK_RDUNLOCK LK_RDUNLOCK LK_RDUNLOCK LK_RDUNLOCK IK_RDUNLOCK IMER_RUN TIMER_ADD TIMER_MOD TIMER_MOD TIMER_MOL	spin lock read/write lock run timer list add to timer list modify timer list delete from timer list	lock try lock (exit) unlock write lock write unlock read lock read lock read unlock	sys_msgrcv() sys_msgget() sys_msgget() sys_msgget() sys_msgget() sys_shmat() sys_shmget() sys_s_shmget()<	/ipc/shm.c	msqid msqid/msqfig key msqid shmid shmid shmid address where it was called address (hor it was called)	msgp msgp msgfig cmd shmaddr size cmd lock lock lock lock lock rw	msgsz msgsz buf shmflg shmflg buf return value return value return value function address (timer-sfunction) function address (timer-sfunction) function address (timer-sfunction)	msgflg msgtyp raddr addr argument for the function (timer- argument for the function (timer- argument for the function (timer- argument for the function (timer-	inline inline inline inline define inline
75 76 77 78 79 7a 80 81 82 83 84 85 85 86 85 86 87 a0 a1 a2 a3	Locks	SYSV_IPC_MSGRCV SYSV_IPC_MSGGET SYSV_IPC_MSGCTL SYSV_IPC_SHMAT SYSV_IPC_SHMDT SYSV_IPC_SHMDT SYSV_IPC_SHMCTL LK_SPINLOCK LK_SPINLOCK LK_SPINLOCK LK_WRUNLOCK LK_WRUNLOCK LK_WRUNLOCK LK_WRUNLOCK LK_RDUNLOCK LK_RDUNLOCK LK_RDUNLOCK LK_RDUNLOCK LK_RDUNLOCK IK_RDUNLOCK IMER_RUN TIMER_ADD TIMER_MOD TIMER_MOD TIMER_MOL	spin lock read/write lock run timer list add to timer list modify timer list delete from timer list	lock try lock (exit) unlock write lock write unlock read lock read lock read unlock	sys_msgrcv() sys_msgget() sys_msgget() sys_msgget() sys_shmat() spin_unlock() read_unlock() run_timer() del_timer() del_timer, sync() ia64_inb() ia64_inl()	/ipc/shm.c	msqid msqid/msqfig key msqid shmid shmid shmid address where it was called address (hor it was called)	msgp msgp msgfig cmd shmaddr size cmd lock lock lock lock lock rw	msgsz msgsz buf shmflg shmflg buf return value return value return value function address (timer-sfunction) function address (timer-sfunction) function address (timer-sfunction)	msgflg msgtyp raddr addr argument for the function (timer- argument for the function (timer- argument for the function (timer- argument for the function (timer-	inline inline inline inline define inline
75 76 77 78 80 81 82 83 84 85 86 87 a0 a1 a2 a3 a4	Locks	SYSV. IPC MSGRCV SYSV. IPC MSGGET SYSV. IPC MSGCTL SYSV. IPC SHMAT SYSV. IPC SHMDT SYSV. IPC SHMGET SYSV. IPC SHMGET SYSV. IPC SHMCTL LK SPINLOCK LK SPINLOCK LK WRTRYLOCK LK WRTRYLOCK LK WRUNLOCK LK RDUNLOCK LK RDUNLOCK LK RDUNLOCK LK RDUNLOCK LK RDUNLOCK LK RDUNLOCK LK RDUNLOCK TIMER_ADD TIMER_ADD TIMER_MOD TIMER_DEL TIMER_DEL_SYNC	spin lock read/write lock run timer list add to timer list modify timer list delete from timer list	lock try lock (exit) unlock write lock write try lock (exit) write unlock read lock read unlock	sys_msgrcv() sys_msgget() sys_msgget() sys_msgget() sys_msgget() sys_msgget() sys_msgget() sys_shmat() sys_shmat() sys_shmget() sys_s_shmget() </td <td>/ipc/shm.c</td> <td>msqid msqid/msqfig key msqid shmid shmaddr key address where it was called address where it was called tunction address(n) pointer to timer list (timer) pointer to timer list (timer) pointer to timer list (timer)</td> <td>msgp msgp msgfig cmd shmaddr size cmd lock lock lock lock rwlock rwlock rwlock rwlock rwlock rwlock rwlock rwlock rwlock rwlock rwlock unexpired term (timer-sexpires) unexpired term (timer-sexpires) unexpired term (timer-sexpires)</td> <td>msgsz msgsz buf shmflg buf return value return value return value function address (timer->function) function address (timer->function) function address (timer->function) function address (timer->function)</td> <td>msgflg msgtyp raddr addr argument for the function (timer- argument for the function (timer- argument for the function (timer- argument for the function (timer-</td> <td>inline inline inline inline inline define define define</td>	/ipc/shm.c	msqid msqid/msqfig key msqid shmid shmaddr key address where it was called address where it was called tunction address(n) pointer to timer list (timer) pointer to timer list (timer) pointer to timer list (timer)	msgp msgp msgfig cmd shmaddr size cmd lock lock lock lock rwlock rwlock rwlock rwlock rwlock rwlock rwlock rwlock rwlock rwlock rwlock unexpired term (timer-sexpires) unexpired term (timer-sexpires) unexpired term (timer-sexpires)	msgsz msgsz buf shmflg buf return value return value return value function address (timer->function) function address (timer->function) function address (timer->function) function address (timer->function)	msgflg msgtyp raddr addr argument for the function (timer- argument for the function (timer- argument for the function (timer- argument for the function (timer-	inline inline inline inline inline define define define
75 76 77 78 80 81 82 83 84 85 86 87 a0 a1 a2 a3 a4 90	Locks	SYSV. IPC MSGRCV SYSV. IPC MSGGET SYSV. IPC MSGCTL SYSV. IPC SHMAT SYSV. IPC SHMDT SYSV. IPC SHMGET SYSV. IPC SHMGET SYSV. IPC SHMCTL LK SPINLOCK LK SPINLOCK LK WRTRYLOCK LK WRTRYLOCK LK WRUNLOCK LK RDUNLOCK LK RDUNLOCK LK RDUNLOCK LK RDUNLOCK LK RDUNLOCK LK RDUNLOCK LK RDUNLOCK TIMER_ADD TIMER_ADD TIMER_MOD TIMER_DEL TIMER_DEL_SYNC	spin lock read/write lock run timer list add to timer list modify timer list delete from timer list	lock try lock (exit) unlock write lock write try lock (exit) write unlock read lock read unlock	sys_msgrcv() sys_msgget() sys_msgget() sys_msgget() sys_msgget() sys_shmat() spin_unlock() write_unlock() read_unlock() run_timer_list() add_timer() del_timer_sync() ia64_inb() ia64_inb() ia64_ins() ia64_ins() ia64_ins()	/ipc/shm.c	msqid msqid/msqfig key msqid shmid shmaddr key address where it was called address where it was called tunction address(n) pointer to timer list (timer) pointer to timer list (timer) pointer to timer list (timer)	msgp msgp msgfig cmd shmaddr size cmd lock lock lock lock rwlock rwlock rwlock rwlock rwlock rwlock rwlock rwlock rwlock rwlock rwlock unexpired term (timer-sexpires) unexpired term (timer-sexpires) unexpired term (timer-sexpires)	msgsz msgsz buf shmflg buf return value return value return value function address (timer->function) function address (timer->function) function address (timer->function) function address (timer->function)	msgflg msgtyp raddr addr argument for the function (timer- argument for the function (timer- argument for the function (timer- argument for the function (timer-	inline inline inline inline inline define define define
75 76 77 78 80 81 82 83 84 85 86 87 a0 a1 a2 a3 a4 90	Locks	SYSV. IPC MSGRCV SYSV. IPC MSGGET SYSV. IPC MSGCTL SYSV. IPC SHMAT SYSV. IPC SHMDT SYSV. IPC SHMGET SYSV. IPC SHMGET SYSV. IPC SHMCTL LK SPINLOCK LK SPINLOCK LK WRTRYLOCK LK WRTRYLOCK LK WRUNLOCK LK RDUNLOCK LK RDUNLOCK LK RDUNLOCK LK RDUNLOCK LK RDUNLOCK LK RDUNLOCK LK RDUNLOCK TIMER_ADD TIMER_ADD TIMER_MOD TIMER_DEL TIMER_DEL_SYNC	spin lock read/write lock run timer list add to timer list modify timer list delete from timer list delete from timer list with synch	lock try lock (exit) unlock write lock write try lock (exit) write unlock read lock read unlock	sys_msgrcv() sys_msgget() sys_msgget() sys_msgget() sys_msgget() sys_msgget() sys_msgget() sys_msgget() sys_shmat() sys_shmget() sys_shmget() <td>/ipc/shm.c /include/asm-ia64/spinlock.h /kernel/timer.c</td> <td>msqid msqid/msqfig key msqid shmid shmaddr key address where it was called address where it was called tunction address(n) pointer to timer list (timer) pointer to timer list (timer) pointer to timer list (timer)</td> <td>msgp msgp msgfig cmd shmaddr size cmd lock lock lock lock rwlock rwlock rwlock rwlock rwlock rwlock rwlock rwlock rwlock rwlock rwlock unexpired term (timer-sexpires) unexpired term (timer-sexpires) unexpired term (timer-sexpires)</td> <td>msgsz msgsz buf shmflg buf return value return value return value function address (timer->function) function address (timer->function) function address (timer->function) function address (timer->function)</td> <td>msgflg msgtyp raddr addr argument for the function (timer- argument for the function (timer- argument for the function (timer- argument for the function (timer-</td> <td>inline inline inline inline inline define define define</td>	/ipc/shm.c /include/asm-ia64/spinlock.h /kernel/timer.c	msqid msqid/msqfig key msqid shmid shmaddr key address where it was called address where it was called tunction address(n) pointer to timer list (timer) pointer to timer list (timer) pointer to timer list (timer)	msgp msgp msgfig cmd shmaddr size cmd lock lock lock lock rwlock rwlock rwlock rwlock rwlock rwlock rwlock rwlock rwlock rwlock rwlock unexpired term (timer-sexpires) unexpired term (timer-sexpires) unexpired term (timer-sexpires)	msgsz msgsz buf shmflg buf return value return value return value function address (timer->function) function address (timer->function) function address (timer->function) function address (timer->function)	msgflg msgtyp raddr addr argument for the function (timer- argument for the function (timer- argument for the function (timer- argument for the function (timer-	inline inline inline inline inline define define define
75 76 77 78 80 81 82 83 84 85 86 87 a0 a1 a2 a3 a4 90	Locks	SYSV. IPC MSGRCV SYSV. IPC MSGGET SYSV. IPC MSGCTL SYSV. IPC SHMAT SYSV. IPC SHMDT SYSV. IPC SHMGET SYSV. IPC SHMGET SYSV. IPC SHMCTL LK SPINLOCK LK SPINLOCK LK WRTRYLOCK LK WRTRYLOCK LK WRUNLOCK LK RDUNLOCK LK RDUNLOCK LK RDUNLOCK LK RDUNLOCK LK RDUNLOCK LK RDUNLOCK LK RDUNLOCK TIMER_ADD TIMER_ADD TIMER_MOD TIMER_DEL TIMER_DEL_SYNC	spin lock read/write lock run timer list add to timer list modify timer list delete from timer list delete from timer list with synch	lock try lock (exit) unlock write lock write try lock (exit) write unlock read lock read unlock	sys_msgrcv() sys_msgget() sys_msgget() sys_msgget() sys_msgget() sys_shmat() spin_lock() write_unlock() trun_timer_list() add_timer() del_timer() del_timer() ia64_inb() ia64_inb() ia64_inst() ia64_inst() ia64_outb() ia64_outb() ia64_outb()	/ipc/shm.c /include/asm-ia64/spinlock.h /kernel/timer.c	msqid msqid/msqfig key msqid shmid shmaddr key address where it was called address where it was called tunction address(n) pointer to timer list (timer) pointer to timer list (timer) pointer to timer list (timer)	msgp msgp msgfig cmd shmaddr size cmd lock lock lock lock rwlock rwlock rwlock rwlock rwlock rwlock rwlock rwlock rwlock rwlock rwlock unexpired term (timer-sexpires) unexpired term (timer-sexpires) unexpired term (timer-sexpires)	msgsz msgsz buf shmflg buf return value return value return value function address (timer->function) function address (timer->function) function address (timer->function) function address (timer->function)	msgflg msgtyp raddr addr argument for the function (timer- argument for the function (timer- argument for the function (timer- argument for the function (timer-	inline inline inline inline inline define define define
75 76 77 78 79 7a 80 81 82 83 84 85 86 87 a0 a1 a2 a3 a4 90	Locks	SYSV_IPC_MSGRCV SYSV_IPC_MSGGET SYSV_IPC_MSGCTL SYSV_IPC_SHMAT SYSV_IPC_SHMDT SYSV_IPC_SHMGET SYSV_IPC_SHMCTL LK_SPINLOCK LK_SPINLOCK LK_SPINLOCK LK_WRUNLOCK LK_WRUNLOCK LK_WRUNLOCK LK_WRUNLOCK LK_WRUNLOCK LK_RDUNLOCK LK_RDUNLOCK LK_RDUNLOCK LK_RDUNLOCK LK_RDUNLOCK TIMER_RUN TIMER_ADD TIMER_ADD TIMER_DEL TIMER_DEL TIMER_DEL TIMER_DEL	spin lock read/write lock run timer list add to timer list modify timer list delete from timer list delete from timer list with synch	lock try lock (exit) unlock write lock write try lock (exit) write unlock read lock read unlock read unlock port input	sys_msgrcv() sys_msgget() sys_msgget() sys_msgget() sys_msgget() sys_shmat() spin_unlock() write_unlock() read_unlock() read_unlock() run_timer_list() add_timer() mod_timer_sync() ia64_inb() ia64_inb() ia64_inst() ia64_inst() ia64_out() ia64_out() ia64_out() ia64_out() ia64_out() ia64_out()	/ipc/shm.c /include/asm-ia64/spinlock.h /kernel/timer.c	msqid msqid/msqfig key msqid shmid shmid address where it was called address where it	msgp msgp msgftg cmd shmaddr size cmd lock lock lock lock rwlock rwlock rwlock rwlock rwlock rwlock rwlock rwlock rwlock rwlock rwlock rwlock rwlock unexpired term (timer-sexpires) unexpired term (timer-sexpires)	msgsz msgsz buf shmflg buf return value return value return value function address (timer->function) function address (timer->function) function address (timer->function) function address (timer->function) function address (timer->function) address where it was called	msgflg msgtyp raddr addr argument for the function (timer- argument for the function (timer- argument for the function (timer- argument for the function (timer-	inline inline inline define inline define inline define inline define inline
75 76 77 78 80 81 82 83 84 85 86 87 a0 a1 a2 a3 a4 90	Locks	SYSV. IPC. MSGRCV SYSV. IPC. MSGGET SYSV. IPC. MSGCTL SYSV. IPC. SHMAT SYSV. IPC. SHMGET SYSV. IPC. SHMGET SYSV. IPC. SHMGET SYSV. IPC. SHMGET LK. SPINLOCK LK. SPINLOCK LK. WRTRYLOCK LK. WRUNLOCK LK. WRUNLOCK LK. RDUNLOCK LK. RDUNLOCK LK. RDUNLOCK LK. RDUNLOCK IK. RDUNLOCK IK. RDUNLOCK IK. RDUNLOCK IK. RDUNLOCK IK. MRINN TIMER_ADD TIMER_DEL TIMER_DEL TIMER_DEL SYMENDEL O_PORTIN	spin lock read/write lock run timer list add to timer list modify timer list delete from timer list delete from timer list delete from timer list with synch io commands	lock try lock (exit) unlock write lock write try lock (exit) write unlock read lock read unlock read unlock port input	sys_msgrcv() sys_msggel() sys_msggel() sys_msggel() sys_msggel() sys_msggel() sys_shmat() spin_lock() write_unlock() trun_timer_list() add_timer() del_timer() del_timer() ia64_inb() ia64_inb() ia64_inst() ia64_inst() ia64_outs() ia64_outs() ia64_outs()	/ipc/shm.c /include/asm-ia64/spinlock.h /kernel/timer.c	msqid msqid/msqfig key msqid shmid shmid address where it was called address where it was called function address(n) pointer to timer list (timer) pointer to timer list (timer)	msgp msgp msgfg cmd shmaddr size cmd lock lock lock lock rwlock r	msgsz msgsz buf shmflg buf return value return value return value function address (timer->function) function address (timer->function) function address (timer->function) function address (timer->function) function address (timer->function) address where it was called	msgflg msgtyp raddr addr argument for the function (timer- argument for the function (timer- argument for the function (timer- argument for the function (timer-	inline inline inline define inline define inline define inline define inline
75 76 77 78 80 81 82 83 84 85 86 87 a0 a1 a2 a3 a4 90 91 91	Locks	SYSV. IPC MSGRCV SYSV. IPC MSGGET SYSV. IPC MSGCTL SYSV. IPC SHMAT SYSV. IPC SHMDT LK SPINLOCK LK SPINLOCK LK WRTRYLOCK LK WRUNLOCK LK WRUNLOCK LK RDUNLOCK LK RDUNLOCK IK RDUNLOCK IK RDUNLOCK IK RDUNLOCK IK RDUNLOCK IMER_ADD TIMER_MOD TIMER_DEL TIMER_DEL_SYNC O_PORTIN O_PORTOUT O_PORTOUT	spin lock read/write lock run timer list add to timer list delete from timer list delete fr	lock try lock (exit) unlock write lock write lock write unlock read lock read lock read unlock	sys_msgrcv() sys_msgget() sys_msgget() sys_msgget() sys_msgget() sys_msgget() sys_shmat() spin_unlock() (IA32 only write_unlock() trun_timer_list() add_timer() del_timer_sync() ia64_inb() ia64_inb() ia64_inb() ia64_out() i	/ipc/shm.c /include/asm-ia64/spinlock.h /kernel/timer.c ./include/asm-ia64/io.h	msqid msqid/msqfig key msqid shmid shmid address where it was called address where it was called pointer to timer list (timer) pointer to timer list (timer) port address/byte width address of argument address of argument	msgp msgp msgfg cmd shmaddr size cmd lock lock lock lock rwlock rwlock rwlock rwlock rwlock argument for the function(data) unexpired term (timer->expires) unexpired term (timer->expires)	msgsz msgsz buf shmflg buf return value return value function address (timer->function) function address (timer->function) function address (timer->function) function address (timer->function) address where it was called	msgflg msgtyp raddr addr argument for the function (timer- argument for the function (timer- argument for the function (timer- argument for the function (timer-	inline inline inline define inline define inline define inline define inline
75 76 77 78 80 81 82 83 84 85 86 86 87 a0 a1 a2 a3 a4 90 90 91 91 92 93 b0	Locks	SYSV_IPC_MSGRCV SYSV_IPC_MSGGET SYSV_IPC_SIMAT SYSV_IPC_SHMAT SYSV_IPC_SHMET SYSV_IPC_SHMET SYSV_IPC_SHMET SYSV_IPC_SHMET LK_SPINLOCK LK_SPINLOCK LK_WRINLOCK LK_WRUNLOCK LK_RDUNLOCK LK_RDUNLOCK LK_RDUNLOCK LK_RDUNLOCK IKR_ROD TIMER_MOD TIMER_DEL TIMER_DEL TIMER_DEL TIMER_DEL O_PORTIN O_PORTOUT O_PRINTK OOPS_PGFAULT	spin lock read/write lock run timer list add to timer list delete from timer list delete from timer list with synch io commands panic printk oops in page fault handler	lock try lock (exit) unlock write lock write try lock (exit) write unlock read lock read lock read unlock read unlock port input port input just before the oops operation	sys_msgrcv() sys_msggel() sys_msggel() sys_msggel() sys_msggel() sys_msggel() sys_shmat() sys_shmgel() indel, innlock() ina64_ins() ina64_ins() ina64_ins() ina64_outb() ina64_outb() ina64_outb() ina64_outb() ina64_outb() ina64_outb() ina64_outb() ina64_outb() </td <td>/ipc/shm.c /include/asm-ia64/spinlock.h /kernel/timer.c /include/asm-ia64/io.h /kernel/panic.c /kernel/panitk.c /arch/ia64/imm/fault.c</td> <td>msqid msqid/msqfig key msqid shmid shmid address where it was called address where it was called function address(n) pointer to timer list (timer) pointer to timer list (timer) pointer to timer list (timer) pointer to timer list (timer) point address/byte width address of argument address of argument address where it was accessed</td> <td>msgp msgp msgfig cmd shmaddr size cmd lock lock lock lock lock rwlock rwlock rwlock rwlock rwlock rwlock rwlock rwlock rwlock rwlock ind rwlock rwloc</td> <td>msgsz msgsz buf shmflg buf return value return value return value function address (timer->function) function address (timer->function) function address (timer->function) function address (timer->function) function address (timer->function) address where it was called</td> <td>msgflg msgtyp raddr addr argument for the function (timer- argument for the function (timer- argument for the function (timer- argument for the function (timer-</td> <td>inline inline inline define inline define inline define inline define inline</td>	/ipc/shm.c /include/asm-ia64/spinlock.h /kernel/timer.c /include/asm-ia64/io.h /kernel/panic.c /kernel/panitk.c /arch/ia64/imm/fault.c	msqid msqid/msqfig key msqid shmid shmid address where it was called address where it was called function address(n) pointer to timer list (timer) pointer to timer list (timer) pointer to timer list (timer) pointer to timer list (timer) point address/byte width address of argument address of argument address where it was accessed	msgp msgp msgfig cmd shmaddr size cmd lock lock lock lock lock rwlock rwlock rwlock rwlock rwlock rwlock rwlock rwlock rwlock rwlock ind rwlock rwloc	msgsz msgsz buf shmflg buf return value return value return value function address (timer->function) function address (timer->function) function address (timer->function) function address (timer->function) function address (timer->function) address where it was called	msgflg msgtyp raddr addr argument for the function (timer- argument for the function (timer- argument for the function (timer- argument for the function (timer-	inline inline inline define inline define inline define inline define inline
75 76 77 78 80 81 82 83 84 85 86 87 a0 a1 a2 a3 a4 90 91 91	Locks	SYSV. IPC MSGRCV SYSV. IPC MSGGET SYSV. IPC MSGCTL SYSV. IPC SHMAT SYSV. IPC SHMDT LK SPINLOCK LK SPINLOCK LK WRTRYLOCK LK WRUNLOCK LK WRUNLOCK LK RDUNLOCK LK RDUNLOCK IK RDUNLOCK IK RDUNLOCK IK RDUNLOCK IK RDUNLOCK IMER_ADD TIMER_MOD TIMER_DEL TIMER_DEL_SYNC O_PORTIN O_PORTOUT O_PORTOUT	spin lock read/write lock run timer list add to timer list delete from timer list delete fr	lock try lock (exit) unlock write lock write try lock (exit) write unlock read lock read lock read unlock read unlock port input port input just before the oops operation	sys_msgrcv() sys_msgget() sys_msgget() sys_msgget() sys_msgget() sys_msgget() sys_shmat() spin_unlock() (IA32 only write_unlock() trun_timer_list() add_timer() del_timer_sync() ia64_inb() ia64_inb() ia64_inb() ia64_out() i	/ipc/shm.c /include/asm-ia64/spinlock.h /kernel/timer.c /include/asm-ia64/io.h /kernel/panic.c /kernel/panic.c /arch/ia64/mn/fault.c /arch/ia64/st/kst_core.c	msqid msqid/msqfig key msqid shmid shmid address where it was called address where it was called pointer to timer list (timer) pointer to timer list (timer) port address/byte width address of argument address of argument	msgp msgp msgfg cmd shmaddr size cmd lock lock lock lock rwlock rwlock rwlock rwlock rwlock argument for the function(data) unexpired term (timer->expires) unexpired term (timer->expires)	msgsz msgsz buf shmflg buf return value return value function address (timer->function) function address (timer->function) function address (timer->function) function address (timer->function) address where it was called	msgflg msgtyp raddr addr argument for the function (timer- argument for the function (timer- argument for the function (timer- argument for the function (timer-	inline inline inline define inline define inline define inline define inline
75 76 77 78 79 7a 80 81 82 83 84 85 86 87 a0 a1 a2 a3 a4 90 91 92 93 b0 f00	Locks	SYSV. IPC MSGRCV SYSV. IPC MSGGET SYSV. IPC MSGCTL SYSV. IPC SHMAT SYSV. IPC SHMGET LK SPINTRYLOCK LK SPINUNLOCK LK WRLOCK LK WRLOCK LK RDUNLOCK LK RDUNLOCK LK RDUNLOCK LK RDUNLOCK TIMER_RUN TIMER_MOD TIMER_DEL TIMER_DEL TIMER_DEL TIMER_DEL_SYNC	spin lock read/write lock run timer list add to timer list modify timer list delete from timer list delete from timer list with syncf io commands panic printk oops in page fault handler Progress of LKST initialization	lock try lock (exit) unlock write lock write try lock (exit) write unlock read lock read lock read unlock read unlock port input port input just before the oops operation	sys_msgrcv() sys_msgget() sys_msgget() sys_msgget() sys_msgget() sys_shmat() spin_unlock() read_unlock() read_unlock() read_unlock() run_timer_list() add_timer() del_timer, sync()	/ipc/shm.c /include/asm-ia64/spinlock.h /kernel/timer.c ./include/asm-ia64/io.h ./kernel/panic.c /kernel/panic.c /kernel/pintk.c /driver/lkst/kst_core.c	msqid msqid/msqfig key msqid shmid shmaddr key address where it was called address where it was called function address(n) pointer to timer list (timer) pointer to timer list (timer) port address/byte width address of argument address of argument address where it was accessed initialization status	msgp msgp msgfig cmd shmaddr size cmd lock lock lock lock rwlock	msgsz msgsz buf shmflg buf return value return value function address (timer->function) function address (timer->function) function address (timer->function) function address (timer->function) function address (timer->function) address where it was called address where it was called	msgfig msgtyp raddr raddr argument for the function (timer- argument for the function (timer-	inline inline inline define inline define inline define inline define Recorded 2 times; before/after Recorded 2 times; before/after
75 76 77 78 80 81 82 83 84 85 86 87 a0 a1 a2 a3 a4 90 90 91 91 91 92 93 b0 f00 f08	Locks	SYSV. IPC MSGRCV SYSV. IPC MSGGET SYSV. IPC MSGCTL SYSV. IPC SHMAT SYSV. IPC SHMDT LK. SPINLOCK LK. SPINULOCK LK. WRTRYLOCK LK. WRUNULOCK LK. WRUNULOCK LK. WRUNULOCK LK. RDUOCK TIMER_ADD TIMER_ADD TIMER_MOD TIMER_DEL TIMER_DEL_SYNC O_PORTIN O_PORTOUT O_PORTOUT LKST_INIT LKST_INIT LKST_MSET_XCHG	spin lock read/write lock run timer list add to timer list delete from tinte delete from timer list delete from timer list delete from ti	lock try lock (exit) unlock write lock write try lock (exit) write unlock read lock read lock read unlock port input port input just before the oops operation process	sys_msgrcv() sys_msgget() sys_msgget() sys_msgget() sys_msgget() sys_msgget() sys_msgget() sys_shmat() spin_unlock() (IA32 onl) write_unlock() trun_timer_list() add_timer() del_timer_sync() ia64_inb() ia64_inb() ia64_inb() ia64_out()	/ipc/shm.c /include/asm-ia64/spinlock.h /kernel/timer.c ./include/asm-ia64/io.h ./kernel/panic.c /kernel/panic.c /kernel/pintk.c /driver/lkst/kst_core.c	msqid msqid/msqfig key msqid shmid shmid address where it was called address where it was called function address(n) pointer to timer list (timer) pointer to timer list (timer) pointer to timer list (timer) pointer to timer list (timer) port address/byte width address of argument address of argument address vhere it was accessed initialization status old maskset ID	msgp msgp msgfg cmd shmaddr size cmd lock lock lock lock rwlock rwlock rwlock rwlock argument for the function(data) unexpired term (timer->expires) unexpired	msgsz msgsz buf shmflg buf return value return value function address (timer->function) function address (timer->function) function address (timer->function) function address (timer->function) address where it was called address where it was called exception error code pointer to old maskset	msgflg msgtyp raddr addr argument for the function (timer- argument for th	inline inline inline define inline inline inline define inline Recorded 2 times; before/after
75 76 77 78 80 81 82 83 84 85 86 87 a0 a1 a2 a3 a4 90 91 92 93 b0 f00 f08 f10 f11	Locks Timer Others	SYSV_IPC_MSGRCV SYSV_IPC_MSGGET SYSV_IPC_SIMAT SYSV_IPC_SHMAT SYSV_IPC_SHMDT LK_SPINULOCK LK_RDUNLOCK LK_RDUNLOCK TIMER_ADD TIMER_DEL TIMER_DEL TIMER_DEL_SYNC O_PORTIN O_PORTOUT O_PORTOUT LKST_INIT LKST_BUFF_SHIFT LKST_BUFF_OVFLOW LKST_SYNO_UID	spin lock read/write lock run timer list add to timer list delete from timer list delete from timer list delete from timer list delete from timer list timer list delete from timer list timer list delete from timer list delete from timer list synchronization with UID	lock try lock (exit) unlock write lock write try lock (exit) write unlock read lock read lock read unlock port input port input just before the oops operation process	sys_msgrcv() sys_msggel() sys_msggel() sys_msggel() sys_msggel() sys_msggel() sys_shmat() spin_lock() write_unlock() write_unlock() trun_timer_list() add_timer() del_timer() del_timer() del_timer() ia64_inst() ia64_inst() ia64_outb()	/ipc/shm.c /include/asm-ia64/spinlock.h /kernel/timer.c /kernel/panic.c /kernel/panick.c /arch/ia64/inm/fault.c /driver/lkst/lkst_core.c /driver/lkst/lkst_core.c /driver/lkst/lkst_core.c /driver/lkst/lkst_core.c	msqid msqid/msqflg key shmid shmid shmid address where it was called address of argument address of argument address of argument address of atsus old buffer ID pointer to the buffer UID	msgp msgp msgfg cmd shmaddr size cmd lock lock lock lock rwlock rwlock rwlock rwlock argument for the function(data) unexpired term (timer->expires) unexpired	msgsz msgsz buf shmflg buf shmflg buf return value return value function address (timer->function) function address (timer->function) function address (timer->function) function address (timer->function) address where it was called exception error code pointer to old maskset pointer to the process table	msgflg msgtyp raddr addr argument for the function (timer- argument for th	inline inline inline define inline define inline define inline define Recorded 2 times; before/after Recorded 2 times; before/after Recorded 2 times; before/after If masked, LKST stops it. for compensation of dropped log data
75 76 77 78 79 7a 80 81 82 83 84 85 86 87 a0 a1 a2 a3 a4 90 91 92 93 b0 f00 f08 f11 f19 f14	Locks	SYSV. IPC MSGRCV SYSV. IPC MSGGET SYSV. IPC SHMAT SYSV. IPC SHMDT SYSV. IPC SHMOET SYSV. IPC SHMOTL LK. SPINLOCK LK. SPINLOCK LK. WRILOCK LK. WRUNLOCK LK. WRUNLOCK LK. RDUNLOCK LK. RDUNLOCK LK. RDUNLOCK LK. RDUNLOCK LK RDUNLOCK LK RDUNLOCK LK RDUNLOCK LK RDUNLOCK LK RDUNLOCK LK RDUNLOCK UK ROUNLOCK LK RDUNLOCK UK ROUNLOCK UK ROUNCK	spin lock read/write lock run timer list add to timer list delete from timer list delete from timer list with syncr io commands panic printk oops in page fault handler Progress of LKST initialization LKST switches the masksets LKST shifts the buffers overrun occurred in the current Synchronization with GID	lock try lock (exit) unlock write lock write try lock (exit) write unlock read lock read lock read unlock port input port input just before the oops operation process	sys_msgrcv() sys_msgget() sys_msgget() sys_msgget() sys_msgget() sys_shmat() spin_unlock() write_unlock() read_unlock() read_unlock() read_unlock() read_unlock() read_unlock() ia64_insl() ia64_insl() ia64_insl() ia64_insl() ia64_outs() ia64_outsl() jaa64_outsl() jaa64_outsl() jaa64_outsl() jaa64_outsl() jaa64_outsl() jaa64_outsl() jaa64_outsl()	/ipc/shm.c /include/asm-ia64/spinlock.h /kernel/timer.c /include/asm-ia64/io.h /kernel/panic.c /kernel/panic.c /kernel/panic.c /driver/ikst/kst_core.c /driver/ikst/kst_core.c /driver/ikst/kst_core.c /driver/ikst/kst_core.c /driver/ikst/kst_core.c /driver/ikst/kst_core.c /kernel/timer.c, sys.c	msqid msqid/msqflg key msqid shmid shmaddr key address where it was called address where it was called function address(n) pointer to timer list (timer) pointer to timer list (timer) pointer to timer list (timer) pointer to timer list (timer) pointer to timer list (timer) port address/byte width address of argument address of argument address where it was accessed initialization status old maskset ID old buffer ID pointer to the buffer UID GID	msgp msgp msgp cmd shmaddr size cmd lock lock lock rwlock	msgsz msgsz buf shmflg buf return value return value function address (timer->function) function address (timer->function) function address (timer->function) function address (timer->function) function address (timer->function) address where it was called address where it was called exception error code pointer to old maskset pointer to the process table pointer to the process table	msgflg msgtyp raddr raddr 	inline inline inline define inline define inline define inline define Recorded 2 times; before/after Recorded 2 times; before/after Used for automatically shifting buffer. If masked, LKST stops it. for compensation of dropped log data for compensation of dropped log data
75 76 77 78 80 81 82 83 84 85 86 87 a0 a1 a2 a3 a4 90 91 92 93 b0 f00 f08 f10 f11	Locks Timer Others Oops	SYSV_IPC MSGRCV SYSV_IPC MSGGET SYSV_IPC SHMAT SYSV_IPC SHMDT SYSV_IPC SHMDT SYSV_IPC SHMDT SYSV_IPC SHMGET SYSV_IPC SHMGET SYSV_IPC SHMOT LK SPINLOCK LK WRITRYLOCK LK WRUNLOCK LK WRUNLOCK LK RDUNLOCK TIMER_ADD TIMER_ADD TIMER_DEL_SYNC 0_PORTIN 0_PORTIN 0_PORTOUT 0_PORTOUT LKST_INIT LKST_SINT LKST_SUFF_SHIFT LKST_SYNC_UID LKST_SYNC_GID	spin lock read/write lock run timer list add to timer list delete from timer list delete from timer list delete from timer list delete from timer list timer list delete from timer list timer list delete from timer list delete from timer list synchronization with UID	lock try lock (exit) unlock write lock write try lock (exit) write unlock read lock read lock read unlock port input port input just before the oops operation process	sys_msgrcv() sys_msgget() sys_msgget() sys_msgget() sys_msgget() sys_msgget() sys_shmat() spin_unlock() (Inter_tist() read_unlock() (Inter_tist() add_timer() del_timer_sync() ia64_inb() ia64_inb() ia64_inb() ia64_out() ia	/ipc/shm.c /include/asm-ia64/spinlock.h /kernel/timer.c /include/asm-ia64/io.h /kernel/panic.c /kernel/panic.c /kernel/panic.c /driver/lkst/lkst_core.c /driver/lkst/lkst_core.c /driver/lkst/lkst_core.c /driver/lkst/lkst_core.c /kernel/timer.c, sys.c /kernel/timer.c, sys.c	msqid msqid/msqfig key shmid shmid shmid address where it was called function address(n) pointer to timer list (timer) pointer to timer list (timer) pointer to timer list (timer) port address/byte width address of argument address of argument address of argument address vere it was accessed initialization status old maskset ID old buffer ID pointer to the buffer <td>msgp msgp msgfg cmd shmaddr size cmd lock lock lock lock rwlock rwlock rwlock rwlock argument for the function(data) unexpired term (timer->expires) unexpired term (timer->expires) unexpired</td> <td>msgsz msgsz buf shmflg buf return value return value function address (timer->function) function address (timer->function) function address (timer->function) function address (timer->function) function address (timer->function) address where it was called address where it was called exception error code pointer to old maskset pointer to the process table pointer to the process table pointer to the process table</td> <td>msgflg msgtyp raddr addr argument for the function (timer- argument for th</td> <td>inline inline inline define inline define inline inline define Recorded 2 times; before/after Recorded 2 times; before/after Recorded 2 times; before/after If masked, LKST stops it. for compensation of dropped log data for compensation of dropped log data</td>	msgp msgp msgfg cmd shmaddr size cmd lock lock lock lock rwlock rwlock rwlock rwlock argument for the function(data) unexpired term (timer->expires) unexpired	msgsz msgsz buf shmflg buf return value return value function address (timer->function) function address (timer->function) function address (timer->function) function address (timer->function) function address (timer->function) address where it was called address where it was called exception error code pointer to old maskset pointer to the process table pointer to the process table pointer to the process table	msgflg msgtyp raddr addr argument for the function (timer- argument for th	inline inline inline define inline define inline inline define Recorded 2 times; before/after Recorded 2 times; before/after Recorded 2 times; before/after If masked, LKST stops it. for compensation of dropped log data for compensation of dropped log data
75 76 77 78 79 7a 80 81 82 83 84 85 86 87 a0 a1 a2 a3 a4 90 91 92 93 b0 f00 f08 f11 f19 f1a f1b f1c	Locks Timer Others Oops	SYSV. IPC MSGRCV SYSV. IPC MSGGET SYSV. IPC SHMAT SYSV. IPC SHMDT SYSV. IPC SHMOTL LK. SPINLOCK LK. SPINLOCK LK. SPINUNLOCK LK. WRINUNLOCK LK. RDUNLOCK LK. RDUNLOCK LK. RDUNLOCK LK STERD O_PORTIN O_PORTIN O_PORTIN LKST_MOTIN	spin lock read/write lock read/write lock add to timer list modify timer list delete from timer list delete from timer list with synch io commands panic printk oops in page fault handler Progress of LKST initialization LKST switches the masksets LKST shifts the buffers overrun occurred in the current Synchronization with UID Synchronization with TID Extra arguments	lock try lock (exit) unlock write lock write try lock (exit) write unlock read lock read lock read unlock port input port input just before the oops operation process	sys_msgrcv() sys_msgget() sys_msgget() sys_msgget() sys_msgget() sys_shmat() spin_unlock() write_unlock() read_unlock() read_unlock() read_unlock() read_unlock() read_unlock() ia64_insl() ia64_insl() ia64_insl() ia64_insl() ia64_outs() ia64_outsl() jaa64_outsl() jaa64_outsl() jaa64_outsl() jaa64_outsl() jaa64_outsl() jaa64_outsl() jaa64_outsl()	/ipc/shm.c /include/asm-ia64/spinlock.h /kernel/timer.c /include/asm-ia64/io.h /kernel/panic.c /kernel/panic.c /kernel/panic.c /driver/ikst/kst_core.c /driver/ikst/kst_core.c /driver/ikst/kst_core.c /driver/ikst/kst_core.c /driver/ikst/kst_core.c /driver/ikst/kst_core.c /kernel/timer.c, sys.c	msqid msqid/msqflg key shmid shmid shmid address where it was called function address(n) pointer to timer list (timer) pointer to timer list (timer) pointer to timer list (timer) port address/byte width address of argument address of argument address of argument address of argument old buffer ID old buffer ID pointer to the buffer UID GID <td>msgp msgp msgp msgfig cmd shmaddr size cmd lock lock lock lock lock lock rwlock rwlock rwlock rwlock rwlock rwlock rwlock rwlock rwlock rwlock rwlock rwlock rwlock ind unexpired term (timer->expires) unexpired term (timer-</td> <td>msgsz msgsz msgsz buf shmflg buf shmflg buf return value return value function address (timer->function) function address (timer->function) function address (timer->function) function address (timer->function) address where it was called address where it was called pointer to old maskset pointer to old buffer pointer to the process table pointer to the process table pointer to the process table pointer to the process table</td> <td>msgfig msgtyp raddr raddr </td> <td>inline inline inline define inline define inline define inline define Recorded 2 times; before/after Recorded 2 times; before/after Used for automatically shifting buffer. If masked, LKST stops it. for compensation of dropped log data for compensation of dropped log data</td>	msgp msgp msgp msgfig cmd shmaddr size cmd lock lock lock lock lock lock rwlock rwlock rwlock rwlock rwlock rwlock rwlock rwlock rwlock rwlock rwlock rwlock rwlock ind unexpired term (timer->expires) unexpired term (timer-	msgsz msgsz msgsz buf shmflg buf shmflg buf return value return value function address (timer->function) function address (timer->function) function address (timer->function) function address (timer->function) address where it was called address where it was called pointer to old maskset pointer to old buffer pointer to the process table pointer to the process table pointer to the process table pointer to the process table	msgfig msgtyp raddr raddr 	inline inline inline define inline define inline define inline define Recorded 2 times; before/after Recorded 2 times; before/after Used for automatically shifting buffer. If masked, LKST stops it. for compensation of dropped log data for compensation of dropped log data
75 76 77 78 79 7a 80 81 82 83 84 85 86 87 a0 a1 a2 a3 a4 90 91 92 93 b0 f00 f08 f10 f11 f19 f1a f1b f1c f1d	Locks Timer Others Oops	SYSV. IPC MSGRCV SYSV. IPC MSGGET SYSV. IPC SIMGET SYSV. IPC SHMAT SYSV. IPC SHMDT SYSV. IPC SHMGET LK SPINLOCK LK SPINULOCK LK WRTRYLOCK LK WRUNLOCK LK WRUNLOCK LK RDUNLOCK LK RDUNLOCK TIMER, ADD TIMER, ADD TIMER, MOD TIMER, MOD TIMER, DEL_SYNC O_PORTIN O_PORTOUT USST_SINT LKST_MSET_XCHG LKST_BUFF_SHIFT LKST_SYNC_GID LKST_SYNC_GID LKST_SYNC_GID LKST_SYNC_TOD LKST_SYNC_TOD LKST_EXTENDE	spin lock read/write lock run timer list add to timer list delete from timer list delete from timer list delete from timer list delete from timer list with synch io commands panic printk oops in page fault handler Progress of LKST initialization LKST switches the masksets LKST shifts the buffers overrun occurred in the current Synchronization with GID Synchronization with PGID Synchronization with PGID Synchronization with TID Extra arguments End of Extra arguments	lock try lock (exit) unlock write lock write lock read lock read lock read unlock read unlock port input just before the cops operation process buffer.	sys_msgrcv() sys_msgget() sys_msgget() sys_msgget() sys_msgget() sys_msgget() sys_msgget() sys_shmat() spin_unlock() (Inter_tist() read_unlock() (Inter_tist() add_timer() del_timer_sync() ia64_inb() ia64_inb() ia64_out() i	/ipc/shm.c /include/asm-ia64/spinlock.h /kernel/timer.c /include/asm-ia64/io.h /kernel/panic.c /kernel/panic.c /kernel/panic.c /driver/ikst/kst_core.c /driver/ikst/kst_core.c /driver/ikst/kst_core.c /driver/ikst/kst_core.c /kernel/timer.c, sys.c /kernel/timer.c, sys.c /kernel/timer.c, sys.c	msqid msqid msqid shmid shmid shmid shmid shmid address where it was called address of argument address of argument address of argument address value old buffer ID pointer to the buffer UID GID PID TID(pid)	msgp msgp msgp msgfg cmd shmaddr size cmd lock lock lock lock rwl	msgsz msgsz msgsz buf shmflg buf shmflg buf return value return value function address (timer->function) function address (timer->function) function address (timer->function) function address (timer->function) address where it was called exception error code pointer to old maskset pointer to the process table pointer to the process table pointer to the process table	msglig msglyp raddr raddr 	inline inline inline define inline define inline define inline define Recorded 2 times; before/after Recorded 2 times; before/after Recorded 2 times; before/after If masked, LKST stops it. for compensation of dropped log data for compensation of dropped log data for compensation of dropped log data
75 76 77 78 79 7a 80 81 82 83 84 85 86 87 a0 a1 a2 a3 a4 90 91 92 93 b0 f00 f08 f11 f18 f1b f1c	Locks Timer Others Oops	SYSV. IPC MSGRCV SYSV. IPC MSGGET SYSV. IPC SHMAT SYSV. IPC SHMDT SYSV. IPC SHMOTL LK. SPINLOCK LK. SPINLOCK LK. SPINUNLOCK LK. WRINUNLOCK LK. RDUNLOCK LK. RDUNLOCK LK. RDUNLOCK LK STERD O_PORTIN O_PORTIN O_PORTIN LKST_MOTIN	spin lock read/write lock read/write lock add to timer list modify timer list delete from timer list delete from timer list with synch io commands panic printk oops in page fault handler Progress of LKST initialization LKST switches the masksets LKST shifts the buffers overrun occurred in the current Synchronization with UID Synchronization with TID Extra arguments	lock try lock (exit) unlock write lock write lock read lock read lock read unlock read unlock port input just before the cops operation process buffer.	sys_msgrcv() sys_msgget() sys_msgget() sys_msgget() sys_msgget() sys_msgget() sys_shmat() spin_unlock() (Inter_tist() read_unlock() (Inter_tist() add_timer() del_timer_sync() ia64_inb() ia64_inb() ia64_inb() ia64_out() ia	/ipc/shm.c /include/asm-ia64/spinlock.h /kernel/timer.c /include/asm-ia64/io.h /kernel/panic.c /kernel/panic.c /kernel/panic.c /driver/lkst/lkst_core.c /driver/lkst/lkst_core.c /driver/lkst/lkst_core.c /driver/lkst/lkst_core.c /kernel/timer.c, sys.c /kernel/timer.c, sys.c	msqid msqid/msqflg key shmid shmid shmid address where it was called function address(n) pointer to timer list (timer) pointer to timer list (timer) pointer to timer list (timer) port address/byte width address of argument address of argument address of argument address of argument old buffer ID old buffer ID pointer to the buffer UID GID <td>msgp msgp msgp msgfig cmd shmaddr size cmd lock lock lock lock lock lock rwlock rwlock rwlock rwlock rwlock rwlock rwlock rwlock rwlock rwlock rwlock rwlock rwlock ind unexpired term (timer->expires) unexpired term (timer-</td> <td>msgsz msgsz msgsz buf shmflg buf shmflg buf return value return value function address (timer->function) function address (timer->function) function address (timer->function) function address (timer->function) address where it was called address where it was called pointer to old maskset pointer to old buffer pointer to the process table pointer to the process table pointer to the process table pointer to the process table</td> <td>msgfig msgtyp raddr raddr </td> <td>inline inline inline define inline define inline inline define Recorded 2 times; before/after Recorded 2 times; before/after Recorded 2 times; before/after If masked, LKST stops it. for compensation of dropped log data for compensation of dropped log data</td>	msgp msgp msgp msgfig cmd shmaddr size cmd lock lock lock lock lock lock rwlock rwlock rwlock rwlock rwlock rwlock rwlock rwlock rwlock rwlock rwlock rwlock rwlock ind unexpired term (timer->expires) unexpired term (timer-	msgsz msgsz msgsz buf shmflg buf shmflg buf return value return value function address (timer->function) function address (timer->function) function address (timer->function) function address (timer->function) address where it was called address where it was called pointer to old maskset pointer to old buffer pointer to the process table pointer to the process table pointer to the process table pointer to the process table	msgfig msgtyp raddr raddr 	inline inline inline define inline define inline inline define Recorded 2 times; before/after Recorded 2 times; before/after Recorded 2 times; before/after If masked, LKST stops it. for compensation of dropped log data for compensation of dropped log data