Comprehensive PMU Event Reference

POWER7

Alex Mericas, Brad Elkin, Venkat Rajeev Indukuru

Release 1.0

August 30, 2011

IBM Systems and Technology Group

1. Performance Monitoring Events

The POWER7 processor has a built in performance monitoring unit (PMU) for each hardware thread, which provides instrumentation to aid in performance monitoring, workload characterization, system characterization and code analysis.

There are 6 thread-level Performance Monitor Counters (PMC) in a PMU. PMC1 – PMC4 are programmable, PMC5 counts non idle completed instructions and PMC6 counts non idle cycles.

The thread level and core level instrumentation have access to a rich set of performance events (close to 550) that cover essential statistics such as miss rates, unit utilization, thread balance, hazard conditions, translation related misses, stall analysis, instruction mix, L1 I cache and D cache reload source, effective cache counts and memory latency counts.

This document covers all of the performance monitoring events supported by the POWER7 PMU. These events can be measured using tools like hpmcount (AIX) and perf (Linux).

Appendix A lists all of the performance monitoring events for POWER7, sorted alphabetically by event name (a mnemonic). Each event entry includes the event code used to program the PMU, a short description, and the domain that the event covers. The event domain can be at the thread, core or chip level.

Appendix B provides details on each event and how it is triggered. Each entry lists the event name, a brief event description and a detailed description

Appendix A. POWER7 Performance Monitoring Event Codes

Table 1-1. PMU Events

PMC1	PMC2	PMC3	PMC4	Event	Short Description	Doman
00F2				PM_1PLUS_PPC_CMPL	1 or more ppc insts finished	Thread
			00F2	PM_1PLUS_PPC_DISP	Cycles at least one Instr Dispatched,	Thread
		0062		PM_1THRD_CON_RUN_INSTR	1 thread Concurrent Run Instructions	Thread
00FA				PM_ANY_THRD_RUN_CYC	One of threads in run_cycles	Thread
			00F6	PM_BR_MPRED	Number of Branch Mispredicts	Thread
	0004			PM_BR_TAKEN	Branch Taken	Thread
0068				PM_BRU_FIN	Branch Instruction Finished	Thread
			000A	PM_CMPLU_STALL	No groups completed, GCT not empty	Thread
			004E	PM_CMPLU_STALL_BRU	Completion stall due to BRU	Thread
	0016			PM_CMPLU_STALL_DCACHE_MISS	Completion stall caused by D cache miss	Thread
	003C			PM_CMPLU_STALL_DFU	Completion stall caused by Decimal Floating Point Unit	Thread
			0014	PM_CMPLU_STALL_DIV	Completion stall caused by DIV instruction	Thread
0028				PM_CMPLU_STALL_END_GCT_NOSLOT	Count ended because GCT went empty	Thread
			0018	PM_CMPLU_STALL_ERAT_MISS	Completion stall caused by ERAT miss	Thread
	0014			PM_CMPLU_STALL_FXU	Completion stall caused by FXU instruction	Thread
			004C	PM_CMPLU_STALL_IFU	Completion stall due to IFU	Thread
	0012			PM_CMPLU_STALL_LSU	Completion stall caused by LSU instruction	Thread
			0016	PM_CMPLU_STALL_REJECT	Completion stall caused by reject	Thread
			0012	PM_CMPLU_STALL_SCALAR	Completion stall caused by FPU instruction	Thread
	0018			PM_CMPLU_STALL_SCALAR_LONG	Completion stall caused by long latency scalar instruction	Thread
	004A			PM_CMPLU_STALL_STORE	Completion stall due to store instruc-tion	Thread
001C				PM_CMPLU_STALL_THRD	Completion Stalled due to thread con-flict. Group ready to complete but it was another thread's turn	Thread
	001C			PM_CMPLU_STALL_VECTOR	Completion stall caused by Vector instruction	Thread
			004A	PM_CMPLU_STALL_VECTOR_LONG	completion stall due to long latency vector instruction	Thread
001E	001E	001E	001E	PM_CYC	Cycles	Thread
		C04C		PM_DATA_FROM_DL2L3_MOD	Data loaded from distant L2 or L3 modified	Thread

Table 1-1. PMU Events

PMC1	PMC2	PMC3	PMC4	Event	Short Description	Doman
			C044	PM_DATA_FROM_DL2L3_MOD	Data loaded from distant L2 or L3 modified	Thread
		C044		PM_DATA_FROM_DL2L3_SHR	Data loaded from distant L2 or L3 shared	Thread
C04A				PM_DATA_FROM_DMEM	Data loaded from distant memory	Thread
	C042			PM_DATA_FROM_DMEM	Data loaded from distant memory	Thread
C040				PM_DATA_FROM_L2	Data loaded from L2	Thread
		C046		PM_DATA_FROM_L21_MOD	Data loaded from another L2 on same chip modified	Thread
		C04E		PM_DATA_FROM_L21_SHR	Data loaded from another L2 on same chip shared	Thread
			C046	PM_DATA_FROM_L21_SHR	Data loaded from another L2 on same chip shared	Thread
	00FE			PM_DATA_FROM_L2MISS	Demand LD - L2 Miss (not L2 hit)	Thread
			C048	PM_DATA_FROM_L2MISS	Data loaded missed L2	Thread
C048				PM_DATA_FROM_L3	Data loaded from L3	Thread
	C040			PM_DATA_FROM_L3	Data loaded from L3	Thread
C044				PM_DATA_FROM_L31_MOD	Data loaded from another L3 on same chip modified	Thread
C04E				PM_DATA_FROM_L31_SHR	Data loaded from another L3 on same chip shared	Thread
	C046			PM_DATA_FROM_L31_SHR	Data loaded from another L3 on same chip shared	Thread
		00FE		PM_DATA_FROM_L3MISS	Demand LD - L3 Miss (not L2 hit and not L3 hit)	Thread
	C048			PM_DATA_FROM_L3MISS	Data loaded from private L3 miss	Thread
		C04A		PM_DATA_FROM_LMEM	Data loaded from local memory	Thread
			C042	PM_DATA_FROM_LMEM	Data loaded from local memory	Thread
C042				PM_DATA_FROM_RL2L3_MOD	Data loaded from remote L2 or L3 modified	Thread
C04C				PM_DATA_FROM_RL2L3_SHR	Data loaded from remote L2 or L3 shared	Thread
	C044			PM_DATA_FROM_RL2L3_SHR	Data loaded from remote L2 or L3 shared	Thread
		C042		PM_DATA_FROM_RMEM	Data loaded from remote memory	Thread
		001A		PM_DATA_TABLEWALK_CYC	Data Tablewalk Active	Thread
			C05C	PM_DERAT_MISS_16G	DERAT misses for 16G page	Thread
		C05C		PM_DERAT_MISS_16M	DERAT misses for 16M page	Thread
C05C				PM_DERAT_MISS_4K	DERAT misses for 4K page	Thread
	C05C			PM_DERAT_MISS_64K	DERAT misses for 64K page	Thread
0006				PM_DISP_HELD	Dispatch Held	Thread
		0006		PM_DISP_HELD_THERMAL	Dispatch Held due to Thermal	Thread

Table 1-1. PMU Events

PMC1	PMC2	PMC3	PMC4	Event	Short Description	Doman
		0008		PM_DISP_WT	Dispatched Starved (not held, nothing to dispatch)	Thread
	0006			PM_DPU_HELD_POWER	Dispatch Held due to Power Manage-ment	Thread
		00FC		PM_DTLB_MISS	TLB reload valid	Thread
C05E				PM_DTLB_MISS_16G	Data TLB miss for 16G page	Thread
			C05E	PM_DTLB_MISS_16M	Data TLB miss for 16M page	Thread
	C05A			PM_DTLB_MISS_4K	Data TLB miss for 4K page	Thread
	C05E			PM_DTLB_MISS_4K	Data TLB miss for 4K page	Thread
		C05E		PM_DTLB_MISS_64K	Data TLB miss for 64K page	Thread
	00F8			PM_EXT_INT	external interrupt	Thread
00F4				PM_FLOP	Floating Point Operation Finished	Thread
			00F8	PM_FLUSH	Flush (any type)	Thread
		0012		PM_FLUSH_COMPLETION	Completion Flush	Thread
		000C		PM_FREQ_DOWN	Frequency is being slewed down due to Power Management	Thread
			000C	PM_FREQ_UP	Power Management: Above Thresh-old A	Thread
	000E			PM_FXU_BUSY	fxu0 busy and fxu1 busy.	Thread
000E				PM_FXU_IDLE	fxu0 idle and fxu1 idle	Thread
		000E		PM_FXU0_BUSY_FXU1_IDLE	fxu0 busy and fxu1 idle	Thread
0004				PM_FXU0_FIN	FXU0 Finished	Thread
			000E	PM_FXU1_BUSY_FXU0_IDLE	fxu0 idle and fxu1 busy.	Thread
			0004	PM_FXU1_FIN	FXU1 Finished	Thread
	0008			PM_GCT_EMPTY_CYC	GCT empty, all threads	Thread
			001A	PM_GCT_NOSLOT_BR_MPRED	GCT empty by branch mispredict	Thread
			001C	PM_GCT_NOSLOT_BR_MPRED_IC_MISS	GCT empty by branch mispredict + IC miss	Thread
00F8				PM_GCT_NOSLOT_CYC	No itags assigned	Thread
	001A			PM_GCT_NOSLOT_IC_MISS	GCT empty by I cache miss	Thread
000A				PM_GRP_BR_MPRED_NONSPEC	Group experienced non-speculative branch redirect	Thread
		0004		PM_GRP_CMPL	group completed	Thread
		000A		PM_GRP_DISP	dispatch_success (Group Dispatched)	Thread
000C				PM_GRP_IC_MISS_NONSPEC	Group experienced non-speculative I cache miss	Thread
0031				PM_GRP_MRK	IDU Marked Instruction	Thread
0030				PM_GRP_MRK_CYC	cycles IDU marked instruction before dispatch	Thread
	000A			PM_HV_CYC	cycles in hypervisor mode	Thread

Table 1-1. PMU Events

PMC1	PMC2	PMC3	PMC4	Event	Short Description	Doman
0018				PM_IC_DEMAND_CYC	Cycles when a demand ifetch was pending	Thread
00F6				PM_IERAT_MISS	IERAT Miss	Thread
			0066	PM_IFU_FIN	IFU Finished a (non-branch) instruction	Thread
0002	0002	0002	0002	PM_INST_CMPL	# PPC Instructions completed	Thread
	00F2			PM_INST_DISP	# PPC Dispatched	Thread
		00F2		PM_INST_DISP	Instructions dispatched	Thread
		404C		PM_INST_FROM_DL2L3_MOD	Instruction fetched from distant L2 or L3 modified	Thread
			4044	PM_INST_FROM_DL2L3_MOD	Instruction fetched from distant L2 or L3 modified	Thread
		4044		PM_INST_FROM_DL2L3_SHR	Instruction fetched from distant L2 or L3 shared	Thread
404A				PM_INST_FROM_DMEM	Instruction fetched from distant memory	Thread
	4042			PM_INST_FROM_DMEM	Instruction fetched from distant memory	Thread
4040				PM_INST_FROM_L2	Instruction fetched from L2	Thread
		4046		PM_INST_FROM_L21_MOD	Instruction fetched from another L2 on same chip modified	Thread
		404E		PM_INST_FROM_L21_SHR	Instruction fetched from another L2 on same chip shared	Thread
			4046	PM_INST_FROM_L21_SHR	Instruction fetched from another L2 on same chip shared	Thread
			4048	PM_INST_FROM_L2MISS	Instruction fetched missed L2	Thread
4048				PM_INST_FROM_L3	Instruction fetched from L3	Thread
	4040			PM_INST_FROM_L3	Instruction fetched from L3	Thread
4044				PM_INST_FROM_L31_MOD	Instruction fetched from another L3 on same chip modified	Thread
404E				PM_INST_FROM_L31_SHR	Instruction fetched from another L3 on same chip shared	Thread
	4046			PM_INST_FROM_L31_SHR	Instruction fetched from another L3 on same chip shared	Thread
	4048			PM_INST_FROM_L3MISS	Instruction fetched missed L3	Thread
		404A		PM_INST_FROM_LMEM	Instruction fetched from local memory	Thread
			4042	PM_INST_FROM_LMEM	Instruction fetched from local memory	Thread
4046				PM_INST_FROM_PREF	Instruction fetched from prefetch	Thread
4042				PM_INST_FROM_RL2L3_MOD	Instruction fetched from remote L2 or L3 modified	Thread
404C				PM_INST_FROM_RL2L3_SHR	Instruction fetched from remote L2 or L3 shared	Thread
	4044			PM_INST_FROM_RL2L3_SHR	Instruction fetched from remote L2 or L3 shared	Thread

Table 1-1. PMU Events

PMC1	PMC2	PMC3	PMC4	Event	Short Description	Doman
		4042		PM_INST_FROM_RMEM	Instruction fetched from remote memory	Thread
00F0				PM_INST_IMC_MATCH_CMPL	IMC Match Count	Thread
		0016		PM_INST_IMC_MATCH_DISP	IMC Matches dispatched	Thread
			E054	PM_INST_PTEG_FROM_DL2L3_MOD	Instruction PTEG loaded from distant L2 or L3 modified	Thread
		E054		PM_INST_PTEG_FROM_DL2L3_SHR	Instruction PTEG loaded from remote L2 or L3 shared	Thread
	E052			PM_INST_PTEG_FROM_DMEM	Instruction PTEG loaded from distant memory	Thread
E050				PM_INST_PTEG_FROM_L2	Instruction PTEG loaded from L2	Thread
		E056		PM_INST_PTEG_FROM_L21_MOD	Instruction PTEG loaded from another L2 on same chip modified	Thread
			E056	PM_INST_PTEG_FROM_L21_SHR	Instruction PTEG loaded from another L2 on same chip shared	Thread
			E058	PM_INST_PTEG_FROM_L2MISS	Instruction PTEG loaded from L2 miss	Thread
	E050			PM_INST_PTEG_FROM_L3	Instruction PTEG loaded from L3	Thread
E054				PM_INST_PTEG_FROM_L31_MOD	Instruction PTEG loaded from another L3 on same chip modified	Thread
	E056			PM_INST_PTEG_FROM_L31_SHR	Instruction PTEG loaded from another L3 on same chip shared	Thread
	E058			PM_INST_PTEG_FROM_L3MISS	Instruction PTEG loaded from L3 miss	Thread
			E052	PM_INST_PTEG_FROM_LMEM	Instruction PTEG loaded from local memory	Thread
E052				PM_INST_PTEG_FROM_RL2L3_MOD	Instruction PTEG loaded from remote L2 or L3 modified	Thread
	E054			PM_INST_PTEG_FROM_RL2L3_SHR	Instruction PTEG loaded from remote L2 or L3 shared	Thread
		E052		PM_INST_PTEG_FROM_RMEM	Instruction PTEG loaded from remote memory	Thread
0014				PM_IOPS_CMPL	Internal Operations completed	Thread
		0014		PM_IOPS_DISP	IOPS dispatched	Thread
			00FC	PM_ITLB_MISS	ITLB Reloaded (always zero on POWER6)	Thread
		00F6		PM_L1_DCACHE_RELOAD_VALID	L1 reload data source valid	Thread
	00FC			PM_L1_ICACHE_MISS	Demand iCache Miss	Thread
			00F0	PM_LD_MISS_L1	Load Missed L1	Thread
	00F6			PM_LSU_DERAT_MISS	DERAT Reloaded due to a DERAT miss	Thread
		C05A		PM_LSU_DERAT_MISS	DERAT Misses	Thread
		0066		PM_LSU_FIN	LSU Finished an instruction (up to 2 per cycle)	Thread
0066				PM_LSU_FX_FIN	LSU Finished a FX operation (up to 2 per cycle)	Core

Table 1-1. PMU Events

PMC1	PMC2	PMC3	PMC4	Event	Short Description	Doman
		001C		PM_LSU_LMQ_SRQ_EMPTY_ALL_CYC	ALL threads lsu empty (lmq and srq empty)	Core
	003E			PM_LSU_LMQ_SRQ_EMPTY_CYC	LSU empty (lmq and srq empty)	Thread
0064				PM_LSU_REJECT	LSU Reject (up to 2 per cycle)	Thread
	0064			PM_LSU_REJECT_ERAT_MISS	LSU Reject due to ERAT (up to 2 per cycles)	Thread
			0008	PM_LSU_SRQ_EMPTY_CYC	ALL threads srq empty	Core
001A				PM_LSU_SRQ_FULL_CYC	Storage Queue is full and is blocking dispatch	Thread
		0036		PM_MRK_BR_MPRED	Marked Branch Mispredicted	Thread
0036				PM_MRK_BR_TAKEN	Marked Branch Taken	Thread
	003A			PM_MRK_BRU_FIN	bru marked instr finish	Thread
		D04C		PM_MRK_DATA_FROM_DL2L3_MOD	Marked data loaded from distant L2 or L3 modified	Thread
			D044	PM_MRK_DATA_FROM_DL2L3_MOD	Marked data loaded from distant L2 or L3 modified	Thread
			002A	PM_MRK_DATA_FROM_DL2L3_MOD_CYC	Marked ld latency Data source 1011 (L2.75/L3.75 M different 4 chip node)	Thread
		D044		PM_MRK_DATA_FROM_DL2L3_SHR	Marked data loaded from distant L2 or L3 shared	Thread
	002A			PM_MRK_DATA_FROM_DL2L3_SHR_CYC	Marked ld latency Data Source 1010 (Distant L2.75/L3.75 S)	Thread
D04A				PM_MRK_DATA_FROM_DMEM	Marked data loaded from distant memory	Thread
	D042			PM_MRK_DATA_FROM_DMEM	Marked data loaded from distant memory	Thread
	002E			PM_MRK_DATA_FROM_DMEM_CYC	Marked ld latency Data Source 1110 (Distant Memory)	Thread
D040				PM_MRK_DATA_FROM_L2	Marked data loaded from L2	Thread
	0020			PM_MRK_DATA_FROM_L2_CYC	Marked ld latency Data source 0000 (L2 hit)	Thread
		D046		PM_MRK_DATA_FROM_L21_MOD	Marked data loaded from another L2 on same chip modified	Thread
			0024	PM_MRK_DATA_FROM_L21_MOD_CYC	Marked ld latency Data source 0101 (L2.1 M same chip)	Thread
		D04E		PM_MRK_DATA_FROM_L21_SHR	Marked data loaded from another L2 on same chip shared	Thread
			D046	PM_MRK_DATA_FROM_L21_SHR	Marked data loaded from another L2 on same chip shared	Thread
	0024			PM_MRK_DATA_FROM_L21_SHR_CYC	Marked ld latency Data source 0100 (L2.1 S)	Thread
			D048	PM_MRK_DATA_FROM_L2MISS	Marked data loaded missed L2	Thread
D048				PM_MRK_DATA_FROM_L3	Marked data loaded from L3	Thread
	D040			PM_MRK_DATA_FROM_L3	Marked data loaded from L3	Thread

Table 1-1. PMU Events

PMC1	PMC2	PMC3	PMC4	Event	Short Description	Doman
			0020	PM_MRK_DATA_FROM_L3_CYC	Marked ld latency Data source 0001 (L3)	Thread
D044				PM_MRK_DATA_FROM_L31_MOD	Marked data loaded from another L3 on same chip modified	Thread
			0026	PM_MRK_DATA_FROM_L31_MOD_CYC	Marked ld latency Data source 0111 (L3.1 M same chip)	Thread
D04E				PM_MRK_DATA_FROM_L31_SHR	Marked data loaded from another L3 on same chip shared	Thread
	D046			PM_MRK_DATA_FROM_L31_SHR	Marked data loaded from another L3 on same chip shared	Thread
	0026			PM_MRK_DATA_FROM_L31_SHR_CYC	Marked ld latency Data source 0110 (L3.1 S)	Thread
	D048			PM_MRK_DATA_FROM_L3MISS	Marked data loaded from L3 miss	Thread
		D04A		PM_MRK_DATA_FROM_LMEM	Marked data loaded from local memory	Thread
			D042	PM_MRK_DATA_FROM_LMEM	Marked data loaded from local memory	Thread
	002C			PM_MRK_DATA_FROM_LMEM_CYC	Marked ld latency Data Source 1100 (Local Memory)	Thread
D042				PM_MRK_DATA_FROM_RL2L3_MOD	Marked data loaded from remote L2 or L3 modified	Thread
			0028	PM_MRK_DATA_FROM_RL2L3_MOD_CYC	Marked ld latency Data source 1001 (L2.5/L3.5 M same 4 chip node)	Thread
D04C				PM_MRK_DATA_FROM_RL2L3_SHR	Marked data loaded from remote L2 or L3 shared	Thread
	D044			PM_MRK_DATA_FROM_RL2L3_SHR	Marked data loaded from remote L2 or L3 shared	Thread
	0028			PM_MRK_DATA_FROM_RL2L3_SHR_CYC	Marked ld latency Data Source 1000 (Remote L2.5/L3.5 S)	Thread
		D042		PM_MRK_DATA_FROM_RMEM	Marked data loaded from remote memory	Thread
			002C	PM_MRK_DATA_FROM_RMEM_CYC	Marked ld latency Data source 1101 (Memory same 4 chip node)	Thread
			D05C	PM_MRK_DERAT_MISS_16G	Marked DERAT misses for 16G page	Thread
		D05C		PM_MRK_DERAT_MISS_16M	Marked DERAT misses for 16M page	Thread
D05C				PM_MRK_DERAT_MISS_4K	Marked DERAT misses for 4K page	Thread
	D05C			PM_MRK_DERAT_MISS_64K	Marked DERAT misses for 64K page	Thread
	0032			PM_MRK_DFU_FIN	Decimal Unit marked Instruction Finish	Thread
D05E				PM_MRK_DTLB_MISS_16G	Marked Data TLB misses for 16G page	Thread
			D05E	PM_MRK_DTLB_MISS_16M	Marked Data TLB misses for 16M page	Thread
	D05A			PM_MRK_DTLB_MISS_4K	Marked Data TLB misses for 4K page	Thread
	D05E			PM_MRK_DTLB_MISS_4K	Marked Data TLB misses for 4K page	Thread

Table 1-1. PMU Events

PMC1	PMC2	PMC3	PMC4	Event	Short Description	Doman
		D05E		PM_MRK_DTLB_MISS_64K	Marked Data TLB misses for 64K page	Thread
003C				PM_MRK_FIN_STALL_CYC	Marked instruction Finish Stall cycles (marked finish after NTC)	Thread
003D				PM_MRK_FIN_STALL_CYC_COUNT	Marked instruction Finish Stall cycles (marked finish after NTC) (use edge detect to count #)	Thread
	0038			PM_MRK_FXU_FIN	fxu marked instr finish	Thread
			0030	PM_MRK_GRP_CMPL	Marked group complete	Thread
			0038	PM_MRK_GRP_IC_MISS	Marked group experienced I cache miss	Thread
		003A		PM_MRK_IFU_FIN	IFU non-branch marked instruction finished	Thread
	0030			PM_MRK_INST_DISP	marked instruction dispatch	Thread
		0030		PM_MRK_INST_FIN	marked instr finish any unit	Thread
0032				PM_MRK_INST_ISSUED	Marked instruction issued	Thread
			0034	PM_MRK_INST_TIMEO	marked Instruction finish timeout	Thread
003E				PM_MRK_LD_MISS_EXPOSED_CYC	Marked Load exposed Miss	Thread
003F				PM_MRK_LD_MISS_EXPOSED_CYC_COUNT	Marked Load exposed Miss (use edge detect to count #)	Thread
	0036			PM_MRK_LD_MISS_L1	Marked DL1 Demand Miss	Thread
			003E	PM_MRK_LD_MISS_L1_CYC	L1 data load miss cycles	Thread
		D05A		PM_MRK_LSU_DERAT_MISS	Marked DERAT Miss	Thread
			0032	PM_MRK_LSU_FIN	Marked LSU instruction finished	Thread
			0064	PM_MRK_LSU_REJECT	LSU marked reject	Thread
		0064		PM_MRK_LSU_REJECT_ERAT_MISS	LSU marked reject due to ERAT	Thread
			D054	PM_MRK_PTEG_FROM_DL2L3_MOD	Marked PTEG loaded from distant L2 or L3 modified	Thread
		D054		PM_MRK_PTEG_FROM_DL2L3_SHR	Marked PTEG loaded from remote L2 or L3 shared	Thread
	D052			PM_MRK_PTEG_FROM_DMEM	Marked PTEG loaded from distant memory	Thread
D050				PM_MRK_PTEG_FROM_L2	Marked PTEG loaded from L2	Thread
		D056		PM_MRK_PTEG_FROM_L21_MOD	Marked PTEG loaded from another L2 on same chip modified	Thread
			D056	PM_MRK_PTEG_FROM_L21_SHR	Marked PTEG loaded from another L2 on same chip shared	Thread
			D058	PM_MRK_PTEG_FROM_L2MISS	Marked PTEG loaded from L2 miss	Thread
	D050			PM_MRK_PTEG_FROM_L3	Marked PTEG loaded from L3	Thread
D054				PM_MRK_PTEG_FROM_L31_MOD	Marked PTEG loaded from another L3 on same chip modified	Thread
	D056			PM_MRK_PTEG_FROM_L31_SHR	Marked PTEG loaded from another L3 on same chip shared	Thread

Table 1-1. PMU Events

PMC1	PMC2	PMC3	PMC4	Event	Short Description	Doman
	D058			PM_MRK_PTEG_FROM_L3MISS	Marked PTEG loaded from L3 miss	Thread
			D052	PM_MRK_PTEG_FROM_LMEM	Marked PTEG loaded from local memory	Thread
D052				PM_MRK_PTEG_FROM_RL2L3_MOD	Marked PTEG loaded from remote L2 or L3 modified	Thread
	D054			PM_MRK_PTEG_FROM_RL2L3_SHR	Marked PTEG loaded from remote L2 or L3 shared	Thread
		D052		PM_MRK_PTEG_FROM_RMEM	Marked PTEG loaded from remote memory	Thread
0034				PM_MRK_ST_CMPL	marked store finished (was complete)	Thread
		0034		PM_MRK_ST_CMPL_INT	marked store complete (data home) with intervention	Thread
	0034			PM_MRK_ST_NEST	marked store sent to Nest	Thread
		003E		PM_MRK_STALL_CMPLU_CYC	Marked Group Completion Stall cycles	Thread
		003F		PM_MRK_STALL_CMPLU_CYC_COUNT	Marked Group Completion Stall cycles (use edge detect to count #)	Thread
		0032		PM_MRK_VSU_FIN	vsu (fpu) marked instr finish	Thread
	0010			PM_PMC1_OVERFLOW	Overflow from counter 1	Thread
		0010		PM_PMC2_OVERFLOW	Overflow from counter 2	Thread
		0020		PM_PMC2_REWIND	PMC2 Rewind Event (did not match condition)	Thread
0022				PM_PMC2_SAVED	PMC2 Rewind Value saved	Thread
			0010	PM_PMC3_OVERFLOW	Overflow from counter 3	Thread
0010				PM_PMC4_OVERFLOW	Overflow from counter 4	Thread
0020				PM_PMC4_REWIND	PMC4 Rewind Event	Thread
		0022		PM_PMC4_SAVED	PMC4 Rewind Value saved (matched condition)	Thread
0024				PM_PMC5_OVERFLOW	Overflow from counter 5	Thread
		0024		PM_PMC6_OVERFLOW	Overflow from counter 6	Thread
			C054	PM_PTEG_FROM_DL2L3_MOD	PTEG loaded from distant L2 or L3 modified	Thread
		C054		PM_PTEG_FROM_DL2L3_SHR	PTEG loaded from remote L2 or L3 shared	Thread
	C052			PM_PTEG_FROM_DMEM	PTEG loaded from distant memory	Thread
C050				PM_PTEG_FROM_L2	PTEG loaded from L2	Thread
		C056		PM_PTEG_FROM_L21_MOD	PTEG loaded from another L2 on same chip modified	Thread
			C056	PM_PTEG_FROM_L21_SHR	PTEG loaded from another L2 on same chip shared	Thread
			C058	PM_PTEG_FROM_L2MISS	PTEG loaded from L2 miss	Thread
	C050			PM_PTEG_FROM_L3	PTEG loaded from L3	Thread

Table 1-1. PMU Events

PMC1	PMC2	PMC3	PMC4	Event	Short Description	Doman
C054				PM_PTEG_FROM_L31_MOD	PTEG loaded from another L3 on same chip modified	Thread
	C056			PM_PTEG_FROM_L31_SHR	PTEG loaded from another L3 on same chip shared	Thread
	C058			PM_PTEG_FROM_L3MISS	PTEG loaded from L3 miss	Thread
			C052	PM_PTEG_FROM_LMEM	PTEG loaded from local memory	Thread
C052				PM_PTEG_FROM_RL2L3_MOD	PTEG loaded from remote L2 or L3 modified	Thread
	C054			PM_PTEG_FROM_RL2L3_SHR	PTEG loaded from remote L2 or L3 shared	Thread
		C052		PM_PTEG_FROM_RMEM	PTEG loaded from remote memory	Thread
	00F4			PM_RUN_CYC	Run_cycles	Thread
			00FA	PM_RUN_INST_CMPL	Run_Instructions	Thread
			00F4	PM_RUN_PURR	Run_PURR	Thread
0008				PM_RUN_SPURR	Run SPURR	Thread
	00F0			PM_ST_FIN	Store Instructions Finished	Thread
		00F0		PM_ST_MISS_L1	L1 D cache store misses	Thread
0000	0000	0000	0000	PM_SUSPENDED	Counter OFF	Thread
0026				PM_TABLEWALK_CYC	Cycles when a tablewalk (I or D) is active	Thread
		00F8		PM_TB_BIT_TRANS	Time Base bit transition	Thread
			0006	PM_THERMAL_MAX	Processor In Thermal MAX	Core
0016				PM_THERMAL_WARN	Processor in Thermal Warning	Core
0060				PM_THRD_1_RUN_CYC	1 thread in Run Cycles	Thread
			0062	PM_THRD_2_CONC_RUN_INSTR	2 thread Concurrent Run Instructions	Core
	0060			PM_THRD_2_RUN_CYC	2 thread in Run Cycles	Core
0062				PM_THRD_3_CONC_RUN_INST	3 thread Concurrent Run Instructions	Thread
		0060		PM_THRD_3_RUN_CYC	3 thread in Run Cycles	Core
	0062			PM_THRD_4_CONC_RUN_INST	4 thread Concurrent Run Instructions	Thread
			0060	PM_THRD_4_RUN_CYC	4 thread in Run Cycles	Core
	000C			PM_THRD_ALL_RUN_CYC	All Threads in run_cycles	Core
		00F4		PM_THRD_CONC_RUN_INST	Concurrent Run Instructions	Thread
0012				PM_THRD_GRP_CMPL_BOTH_CYC	Cycles group completed by both threads	Core
0038				PM_THRESH_TIMEO	Threshold timeout event	Thread
	0066			PM_TLB_MISS	TLB Miss (I + D)	Thread
40B8	40B8	40B8	40B8	PM_BC_PLUS_8_CONV	BC+8 Converted	Thread
40BA	40BA	40BA	40BA	PM_BC_PLUS_8_RSLV_TAKEN	BC+8 Resolve outcome was Taken, resulting in the conditional instruction being canceled	Thread

Table 1-1. PMU Events

PMC1	PMC2	PMC3	PMC4	Event	Short Description	Doman
40A4	40A4	40A4	40A4	PM_BR_MPRED_CCACHE	Branch Mispredict due to Count Cache prediction	Thread
40AC	40AC	40AC	40AC	PM_BR_MPRED_CR	Branch mispredict - taken/not taken	Thread
48AE	48AE	48AE	48AE	PM_BR_MPRED_CR_TA	Branch mispredict - taken/not taken and target	Thread
40A6	40A6	40A6	40A6	PM_BR_MPRED_LSTACK	Branch Mispredict due to Link Stack	Thread
40AE	40AE	40AE	40AE	PM_BR_MPRED_TA	Branch mispredict - target address	Thread
409C	409C	409C	409C	PM_BR_PRED	Branch Predictions made	Thread
40A0	40A0	40A0	40A0	PM_BR_PRED_CCACHE	Count Cache Predictions	Thread
40A8	40A8	40A8	40A8	PM_BR_PRED_CR	Branch predict - taken/not taken	Thread
48AA	48AA	48AA	48AA	PM_BR_PRED_CR_TA	Branch predict - taken/not taken and target	Thread
40A2	40A2	40A2	40A2	PM_BR_PRED_LSTACK	Link Stack Predictions	Thread
40AA	40AA	40AA	40AA	PM_BR_PRED_TA	Branch predict - target address	Thread
409E	409E	409E	409E	PM_BR_UNCOND	Unconditional Branch	Thread
508A	508A	508A	508A	PM_BTAC_HIT	BTAC Correct Prediction	Thread
5088	5088	5088	5088	PM_BTAC_MISS	BTAC Mispredicted	Thread
D0B0	D0B0	D0B0	D0B0	PM_DC_PREF_DST	Data Stream Touch	Thread
2090	2090	2090	2090	PM_DISP_CLB_HELD	CLB Hold: Any Reason	Thread
2092	2092	2092	2092	PM_DISP_CLB_HELD_BAL	Dispatch/CLB Hold: Balance	Thread
2094	2094	2094	2094	PM_DISP_CLB_HELD_RES	Dispatch/CLB Hold: Resource	Thread
20A8	20A8	20A8	20A8	PM_DISP_CLB_HELD_SB	Dispatch/CLB Hold: Scoreboard	Thread
2098	2098	2098	2098	PM_DISP_CLB_HELD_SYNC	Dispatch/CLB Hold: Sync type instruction	Thread
2096	2096	2096	2096	PM_DISP_CLB_HELD_TLBIE	Dispatch Hold: Due to TLBIE	Thread
20A6	20A6	20A6	20A6	PM_DSEG	DSEG Exception	Thread
D090	D090	D090	D090	PM_DSLB_MISS	Data SLB Miss - Total of all segment sizes	Thread
2080	2080	2080	2080	PM_EE_OFF_EXT_INT	ee off and external interrupt	Thread
2084	2084	2084	2084	PM_FLUSH_BR_MPRED	Flush caused by branch mispredict	Thread
2082	2082	2082	2082	PM_FLUSH_DISP	Dispatch flush	Thread
208C	208C	208C	208C	PM_FLUSH_DISP_SB	Dispatch Flush: Scoreboard	Thread
			F080	PM_L3_PREF_BUSY	Prefetch machines >= threshold (8,16,20,24)	Core
2088	2088	2088	2088	PM_FLUSH_DISP_SYNC	Dispatch Flush: Sync	Thread
208A	208A	208A	208A	PM_FLUSH_DISP_TLBIE	Dispatch Flush: TLBIE	Thread
			F082	PM_L3_RD_BUSY	Rd machines busy >= threshold (2,4,6,8)	Thread
2086	2086	2086	2086	PM_FLUSH_PARTIAL	Partial flush	Thread
4086	4086	4086	4086	PM_GCT_FULL_CYC	Cycles No room in EAT	Thread

Table 1-1. PMU Events

PMC1	PMC2	PMC3	PMC4	Event	Short Description	Doman
209C	209C	209C	209C	PM_GCT_UTIL_1to2_SLOTS	GCT Utilization 1-2 entries	Thread
20A2	20A2	20A2	20A2	PM_GCT_UTIL_11plus_SLOTS	GCT Utilization 11+ entries	Thread
209E	209E	209E	209E	PM_GCT_UTIL_3to6_SLOTS	GCT Utilization 3-6 entries	Thread
20A0	20A0	20A0	20A0	PM_GCT_UTIL_7to10_SLOTS	GCT Utilization 7-10 entries	Thread
4084	4084	4084	4084	PM_IBUF_FULL_CYC	Cycles No room in ibuff	Thread
4082	4082	4082	4082	PM_IC_BANK_CONFLICT	Read blocked due to interleave conflict.	Thread
4098	4098	4098	4098	PM_IC_DEMAND_L2_BHT_REDIRECT	L2 I cache demand request due to BHT redirect	Thread
4898	4898	4898	4898	PM_IC_DEMAND_L2_BR_ALL	L2 I cache demand request due to BHT or redirect	Thread
409A	409A	409A	409A	PM_IC_DEMAND_L2_BR_REDIRECT	L2 I cache demand request due to branch redirect	Thread
4088	4088	4088	4088	PM_IC_DEMAND_REQ	Demand Instruction fetch request	Thread
4890	4890	4890	4890	PM_IC_PREF_CANCEL_ALL	Prefetch Canceled due to page boundary or icache hit	Thread
4092	4092	4092	4092	PM_IC_PREF_CANCEL_HIT	Prefetch Canceled due to icache hit	Thread
4094	4094	4094	4094	PM_IC_PREF_CANCEL_L2	L2 Squashed request	Thread
4090	4090	4090	4090	PM_IC_PREF_CANCEL_PAGE	Prefetch Canceled due to page boundary	Thread
408A	408A	408A	408A	PM_IC_PREF_REQ	Instruction prefetch requests	Thread
408E	408E	408E	408E	PM_IC_PREF_WRITE	Instruction prefetch written into IL1	Thread
4096	4096	4096	4096	PM_IC_RELOAD_SHR	Reloading line to be shared between the threads	Thread
4888	4888	4888	4888	PM_IC_REQ_ALL	Icache requests, prefetch + demand	Thread
488C	488C	488C	488C	PM_IC_WRITE_ALL	Icache sectors written, prefetch + demand	Thread
40BE	40BE	40BE	40BE	PM_IERAT_WR_64K	large page 64k	Thread
40BC	40BC	40BC	40BC	PM_IERAT_XLATE_WR_16MPLUS	large page 16M+	Thread
4080	4080	4080	4080	PM_INST_FROM_L1	Instruction fetches from L1	Thread
0881				PM_NEST_PAIR0_ADD	Nest events (MC0/MC1/PB/GX), Pair0 ADD	Thread
	0881			PM_NEST_PAIR1_ADD	Nest events (MC0/MC1/PB/GX), Pair1 ADD	Thread
		0881		PM_NEST_PAIR2_ADD	Nest events (MC0/MC1/PB/GX), Pair2 ADD	Thread
			0881	PM_NEST_PAIR3_ADD	Nest events (MC0/MC1/PB/GX), Pair3 ADD	Thread
0883				PM_NEST_PAIR0_AND	Nest events (MC0/MC1/PB/GX), Pair0 AND	Thread
	0883			PM_NEST_PAIR1_AND	Nest events (MC0/MC1/PB/GX), Pair1 AND	Thread

Table 1-1. PMU Events

PMC1	PMC2	PMC3	PMC4	Event	Short Description	Doman
		0883		PM_NEST_PAIR2_AND	Nest events (MC0/MC1/PB/GX), Pair2 AND	Thread
			0883	PM_NEST_PAIR3_AND	Nest events (MC0/MC1/PB/GX), Pair3 AND	Thread
20A4	20A4	20A4	20A4	PM_ISEG	ISEG Exception	Thread
D092	D092	D092	D092	PM_ISLB_MISS	Instruction SLB Miss - Tota of all segment sizes	Thread
408C	408C	408C	408C	PM_L1_DEMAND_WRITE	Instruction Demand sectors wriittent into IL1	Thread
D8B8	D8B8	D8B8	D8B8	PM_L1_PREF	L1 Prefetches	Thread
6180				PM_L2_CASTOUT_MOD	L2 Castouts - Modified (M, Mu, Me)	Core
6182				PM_L2_CASTOUT_SHR	L2 Castouts - Shared (T, Te, Si, S)	Core
6382				PM_L2_CO_FAIL_BUSY	L2 RC Cast Out dispatch attempt failed due to all CO machines busy	Core
	6182			PM_L2_DC_INV	Dcache invalidates from L2	Core
			6080	PM_L2_DISP_ALL	All successful LD/ST dispatches for this thread(i+d)	Thread
6482				PM_L2_GLOB_GUESS_CORRECT	L2 guess glb and guess was correct (ie data remote)	Core
	6482			PM_L2_GLOB_GUESS_WRONG	L2 guess glb and guess was not correct (ie data local)	Core
	6180			PM_L2_IC_INV	Icache Invalidates from L2	Core
		6080		PM_L2_INST	Instruction Load Count	Thread
		6082		PM_L2_INST_MISS	Instruction Load Misses	Thread
6080				PM_L2_LD	Data Load Count	Thread
		6180		PM_L2_LD_DISP	All successful load dispatches	Core
		6182		PM_L2_LD_HIT	All successful load dispatches that were L2 hits	Core
	6080			PM_L2_LD_MISS	Data Load Miss	Thread
6880				PM_L2_LDST	Data Load+Store Count	Core
	6880			PM_L2_LDST_MISS	Data Load+Store Miss	Thread
6480				PM_L2_LOC_GUESS_CORRECT	L2 guess loc and guess was correct (ie data local)	Core
	6480			PM_L2_LOC_GUESS_WRONG	L2 guess loc and guess was not correct (ie data remote)	Core
		6480		PM_L2_NODE_PUMP	RC req that was a local (aka node) pump attempt	Core
		6380		PM_L2_RC_ST_DONE	RC did st to line that was Tx or Sx	Core
			6282	PM_L2_RCLD_BUSY_RC_FULL	L2 activated Busy to the core for loads due to all RC full	Core
6280				PM_L2_RCLD_DISP	L2 RC load dispatch attempt	Core

Table 1-1. PMU Events

PMC1	PMC2	PMC3	PMC4	Event	Short Description	Doman
6282				PM_L2_RCLD_DISP_FAIL_ADDR	L2 RC load dispatch attempt failed due to address collision with RC/CO/SN/SQ	Core
	6280			PM_L2_RCLD_DISP_FAIL_OTHER	L2 RC load dispatch attempt failed due to other reasons	Core
	6282			PM_L2_RCST_BUSY_RC_FULL	L2 activated Busy to the core for stores due to all RC full	Core
		6280		PM_L2_RCST_DISP	L2 RC store dispatch attempt	Core
		6282		PM_L2_RCST_DISP_FAIL_ADDR	L2 RC store dispatch attempt failed due to address collision with RC/CO/SN/SQ	Core
			6280	PM_L2_RCST_DISP_FAIL_OTHER	L2 RC store dispatch attempt failed due to other reasons	Core
			6380	PM_L2_SN_M_RD_DONE	SNP dispatched for a read and was M	Core
			6382	PM_L2_SN_M_WR_DONE	SNP dispatched for a write and was M	Core
		6382		PM_L2_SN_SX_I_DONE	SNP dispatched and went from Sx or Tx to Ix	Core
6082				PM_L2_ST	Data Store Count	Thread
			6180	PM_L2_ST_DISP	All successful store dispatches	Core
			6182	PM_L2_ST_HIT	All successful store dispatches that were L2Hits	Core
	6082			PM_L2_ST_MISS	Data Store Miss	Core
		6482		PM_L2_SYS_PUMP	RC req that was a global (aka system) pump attempt	Core
F080				PM_L3_HIT	L3 Hits	Core
	F080			PM_L3_LD_HIT	L3 demand LD Hits	Core
	F082			PM_L3_LD_MISS	L3 demand LD Miss	Core
F082				PM_L3_MISS	L3 Misses	Core
		F080		PM_L3_PREF_HIT	L3 Prefetch Directory Hit	Core
D0AC	D0AC	D0AC	D0AC	PM_L3_PREF_LD	L3 cache LD prefetches	Thread
D8AC	D8AC	D8AC	D8AC	PM_L3_PREF_LDST	L3 cache prefetches LD + ST	Thread
		F082		PM_L3_PREF_MISS	L3 Prefetch Directory Miss	Core
D0AE	D0AE	D0AE	D0AE	PM_L3_PREF_ST	L3 cache ST prefetches	Thread
C894	C894	C894	C894	PM_LARX_LSU	Larx Finished	Thread
C094	C094	C094	C094	PM_LARX_LSU0	ls0 Larx Finished	Thread
C096	C096	C096	C096	PM_LARX_LSU1	ls1 Larx Finished	Thread
C880	C880	C880	C880	PM_LD_REF_L1	L1 D cache load references counted at finish	Thread
C080	C080	C080	C080	PM_LD_REF_L1_LSU0	LS0 L1 D cache load references counted at finish	Thread
C082	C082	C082	C082	PM_LD_REF_L1_LSU1	LS1 L1 D cache load references counted at finish	Thread

Table 1-1. PMU Events

PMC1	PMC2	PMC3	PMC4	Event	Short Description	Doman
D8A8	D8A8	D8A8	D8A8	PM_LSU_DC_PREF_STREAM_ALLOC	D cache new prefetch stream allocated	Thread
D8B4	D8B4	D8B4	D8B4	PM_LSU_DC_PREF_STREAM_CONFIRM	Dcache new prefetch stream confirmed	Thread
D8BC	D8BC	D8BC	D8BC	PM_LSU_DC_PREF_STRIDED_STREAM_CONFIRM	Dcache Strided prefetch stream confirmed (software + hardware)	Thread
D0A2	D0A2	D0A2	D0A2	PM_LSU_DCACHE_RELOAD_VALID	count per sector of lines reloaded in L1 (demand + prefetch)	Thread
208E	208E	208E	208E	PM_LSU_FLUSH	Flush initiated by LSU	Thread
C8B8	C8B8	C8B8	C8B8	PM_LSU_FLUSH_LRQ	Flush: LRQ	Thread
C8BC	C8BC	C8BC	C8BC	PM_LSU_FLUSH_SRQ	Flush: SRQ	Thread
C8B0	C8B0	C8B0	C8B0	PM_LSU_FLUSH_ULD	Flush: Unaligned Load	Thread
C8B4	C8B4	C8B4	C8B4	PM_LSU_FLUSH_UST	Flush: Unaligned Store	Thread
C884	C884	C884	C884	PM_LSU_LDF	All Scalar Loads	Thread
C888	C888	C888	C888	PM_LSU_LDX	All Vector loads (vsx vector + vmx vector)	Thread
D0A4	D0A4	D0A4	D0A4	PM_LSU_LMQ_FULL_CYC	LMQ full	Thread
D0A1	D0A1	D0A1	D0A1	PM_LSU_LMQ_S0_ALLOC	Slot 0 of LMQ valid	Thread
D0A0	D0A0	D0A0	D0A0	PM_LSU_LMQ_S0_VALID	Slot 0 of LMQ valid	Thread
D09F	D09F	D09F	D09F	PM_LSU_LRQ_S0_ALLOC	Slot 0 of LRQ valid	Thread
D09E	D09E	D09E	D09E	PM_LSU_LRQ_S0_VALID	Slot 0 of LRQ valid	Thread
C88C	C88C	C88C	C88C	PM_LSU_NCLD	Non-cachable Loads counted at finish	Thread
C090	C090	C090	C090	PM_LSU_NCST	Non-cachable Stores sent to nest	Thread
C0AA	C0AA	C0AA	C0AA	PM_LSU_PARTIAL_CDF	A partial cacheline was returned from the L3	Thread
C8AC	C8AC	C8AC	C8AC	PM_LSU_REJECT_LHS	Reject: Load Hit Store	Thread
C8A4	C8A4	C8A4	C8A4	PM_LSU_REJECT_LMQ_FULL	Reject: LMQ Full (LHR)	Thread
C8A8	C8A8	C8A8	C8A8	PM_LSU_REJECT_SET_MPRED	Reject: Set Predict Wrong	Thread
C0A8	C0A8	C0A8	C0A8	PM_LSU_SET_MPRED	Line already in cache at reload time	Thread
D09D	D09D	D09D	D09D	PM_LSU_SRQ_S0_ALLOC	Slot 0 of SRQ valid	Thread
D09C	D09C	D09C	D09C	PM_LSU_SRQ_S0_VALID	Slot 0 of SRQ valid	Thread
C8A0	C8A0	C8A0	C8A0	PM_LSU_SRQ_STFWD	Load got data from a store	Thread
D097	D097	D097	D097	PM_LSU_SRQ_SYNC_COUNT	SRQ sync count (edge of PM_LSU_SRQ_SYNC_CYC)	Thread
D096	D096	D096	D096	PM_LSU_SRQ_SYNC_CYC	A sync is in the SRQ	Thread
D0A6	D0A6	D0A6	D0A6	PM_LSU_TWO_TABLEWALK_CYC	Cycles when two tablewalks pending on this thread	Thread
D0A8	D0A8	D0A8	D0A8	PM_LSU0_DC_PREF_STREAM_ALLOC	LS0 D cache new prefetch stream allocated	Thread
D0B4	D0B4	D0B4	D0B4	PM_LSU0_DC_PREF_STREAM_CONFIRM	LS0 Dcache prefetch stream confirmed	Thread

Table 1-1. PMU Events

PMC1	PMC2	PMC3	PMC4	Event	Short Description	Doman
D0BC	D0BC	D0BC	D0BC	PM_LSU0_DC_PREF_STREAM_CONFIRM_STRIDE	LS0 Dcache Strided prefetch stream confirmed	Thread
C0B8	C0B8	C0B8	C0B8	PM_LSU0_FLUSH_LRQ	LS0 Flush: LRQ	Thread
C0BC	C0BC	C0BC	C0BC	PM_LSU0_FLUSH_SRQ	LS0 Flush: SRQ	Thread
C0B0	C0B0	C0B0	C0B0	PM_LSU0_FLUSH_ULD	LS0 Flush: Unaligned Load	Thread
C0B4	C0B4	C0B4	C0B4	PM_LSU0_FLUSH_UST	LS0 Flush: Unaligned Store	Thread
D0B8	D0B8	D0B8	D0B8	PM_LSU0_L1_PREF	LS0 L1 cache data prefetches	Thread
C09C	C09C	C09C	C09C	PM_LSU0_L1_SW_PREF	LSU0 Software L1 Prefetches, including SW Transient Prefetches	Thread
C084	C084	C084	C084	PM_LSU0_LDF	LS0 Scalar Loads	Thread
C088	C088	C088	C088	PM_LSU0_LDX	LS0 Vector Loads	Thread
D098	D098	D098	D098	PM_LSU0_LMQ_LHR_MERGE	LS0 Load Merged with another cacheline request	Thread
C08C	C08C	C08C	C08C	PM_LSU0_NCLD	LS0 Non-cachable Loads counted at finish	Thread
C0AC	C0AC	C0AC	C0AC	PM_LSU0_REJECT_LHS	LS0 Reject: Load Hit Store	Thread
C0A4	C0A4	C0A4	C0A4	PM_LSU0_REJECT_LMQ_FULL	LS0 Reject: LMQ Full (LHR)	Thread
C0A0	C0A0	C0A0	C0A0	PM_LSU0_SRQ_STFWD	LS0 SRQ forwarded data to a load	Thread
D0AA	D0AA	D0AA	D0AA	PM_LSU1_DC_PREF_STREAM_ALLOC	LS 1 D cache new prefetch stream allocated	Thread
D0B6	D0B6	D0B6	D0B6	PM_LSU1_DC_PREF_STREAM_CONFIRM	LS1 'Dcache prefetch stream confirmed	Thread
D0BE	D0BE	D0BE	D0BE	PM_LSU1_DC_PREF_STREAM_CONFIRM_STRIDE	LS1 Dcache Strided prefetch stream confirmed	Thread
C0BA	C0BA	C0BA	C0BA	PM_LSU1_FLUSH_LRQ	LS1 Flush: LRQ	Thread
C0BE	C0BE	C0BE	C0BE	PM_LSU1_FLUSH_SRQ	LS1 Flush: SRQ	Thread
C0B2	C0B2	C0B2	C0B2	PM_LSU1_FLUSH_ULD	LS 1 Flush: Unaligned Load	Thread
C0B6	C0B6	C0B6	C0B6	PM_LSU1_FLUSH_UST	LS1 Flush: Unaligned Store	Thread
D0BA	D0BA	D0BA	D0BA	PM_LSU1_L1_PREF	LS1 L1 cache data prefetches	Thread
C09E	C09E	C09E	C09E	PM_LSU1_L1_SW_PREF	LSU1 Software L1 Prefetches, including SW Transient Prefetches	Thread
C086	C086	C086	C086	PM_LSU1_LDF	LS1 Scalar Loads	Thread
C08A	C08A	C08A	C08A	PM_LSU1_LDX	LS1 Vector Loads	Thread
D09A	D09A	D09A	D09A	PM_LSU1_LMQ_LHR_MERGE	LS1 Load Merge with another cacheline request	Thread
C08E	C08E	C08E	C08E	PM_LSU1_NCLD	LS1 Non-cachable Loads counted at finish	Thread
C0AE	C0AE	C0AE	C0AE	PM_LSU1_REJECT_LHS	LS1 Reject: Load Hit Store	Thread
C0A6	C0A6	C0A6	C0A6	PM_LSU1_REJECT_LMQ_FULL	LS1 Reject: LMQ Full (LHR)	Thread
C0A2	C0A2	C0A2	C0A2	PM_LSU1_SRQ_STFWD	LS1 SRQ forwarded data to a load	Thread
D094	D094	D094	D094	PM_LWSYNC	lwsync count (easier to use than IMC)	Thread

Table 1-1. PMU Events

PMC1	PMC2	PMC3	PMC4	Event	Short Description	Doman
209A	209A	209A	209A	PM_LWSYNC_HELD	LWSYNC held at dispatch	Thread
		0083		PM_MEM0_PREFETCH_DISP	MC0 prefetch read dispatched	Chip
			0081	PM_MEM0_RD_CANCEL_TOTAL	MC0 speculative reads cancelled	Chip
		0081		PM_MEM0_RQ_DISP	MC0 read dispatched	Chip
			0083	PM_MEM0_WQ_DISP	MC0 writes dispatched	Chip
D08C	D08C	D08C	D08C	PM_MRK_LSU_FLUSH	Flush: (marked) : All Cases	Thread
D088	D088	D088	D088	PM_MRK_LSU_FLUSH_LRQ	Flush: (marked) LRQ	Thread
D08A	D08A	D08A	D08A	PM_MRK_LSU_FLUSH_SRQ	Flush: (marked) SRQ	Thread
D084	D084	D084	D084	PM_MRK_LSU_FLUSH_ULD	Flush: (marked) Unaligned Load	Thread
D086	D086	D086	D086	PM_MRK_LSU_FLUSH_UST	Flush: (marked) Unaligned Store	Thread
D080	D080	D080	D080	PM_MRK_LSU_PARTIAL_CDF	A partial cacheline was returned from the L3 for a marked load	Thread
D082	D082	D082	D082	PM_MRK_LSU_REJECT_LHS	Reject(marked): Load Hit Store	Thread
D08E	D08E	D08E	D08E	PM_MRK_STCX_FAIL	Marked STCX failed	Thread
0081				PM_PB_NODE_PUMP	Power Bus node pump	Chip
	0081			PM_PB_RETRY_NODE_PUMP	Power bus node pump retries	Chip
	0083			PM_PB_RETRY_SYS_PUMP	Power bus system pump retries	Chip
0083				PM_PB_SYS_PUMP	Power bus system pumps	Chip
28A4	28A4	28A4	28A4	PM_SEG_EXCEPTION	ISEG + DSEG Exception	Thread
5082	5082	5082	5082	PM_SHL_CREATED	SHL table entry Created	Thread
5080	5080	5080	5080	PM_SHL_DEALLOCATED	SHL Table entry deallocated	Thread
5086	5086	5086	5086	PM_SHL_MATCH	SHL Table Match	Thread
5084	5084	5084	5084	PM_SHL_MERGED	SHL table entry merged with existing	Thread
D890	D890	D890	D890	PM_SLB_MISS	Data + Instruction SLB Miss - Total of all segment sizes	Thread
D0B2	D0B2	D0B2	D0B2	PM_SNOOP_TLBIE	TLBIE snoop	Core
C098	C098	C098	C098	PM_STCX_CMPL	STCX executed	Thread
C09A	C09A	C09A	C09A	PM_STCX_FAIL	STCX failed	Thread
40B0	40B0	40B0	40B0	PM_THRD_PRIO_0_1_CYC	Cycles thread running at priority level 0 or 1	Thread
40B2	40B2	40B2	40B2	PM_THRD_PRIO_2_3_CYC	Cycles thread running at priority level 2 or 3	Thread
40B4	40B4	40B4	40B4	PM_THRD_PRIO_4_5_CYC	Cycles thread running at priority level 4 or 5	Thread
40B6	40B6	40B6	40B6	PM_THRD_PRIO_6_7_CYC	Cycles thread running at priority level 6 or 7	Thread
B0A0	B0A0	B0A0	B0A0	PM_VMX_RESULT_SAT_1	Valid result with sat=1	Thread
A880	A880	A880	A880	PM_VSU_1FLOP	one flop (fadd, fmul, fsub, fcmp, fsel, fabs, fnabs, fres, fsqrte, fneg) operation finished	Thread

Table 1-1. PMU Events

PMC1	PMC2	PMC3	PMC4	Event	Short Description	Doman
A898	A898	A898	A898	PM_VSU_2FLOP	two flops operation (scalar fmadd, fnmadd, fmsub, fnmsub and DP vector versions of single flop instructions)	Thread
A88C	A88C	A88C	A88C	PM_VSU_2FLOP_DOUBLE	DP vector version of fmul, fsub, fcmp, fsel, fabs, fnabs, fres ,fsqrte, fneg	Thread
A89C	A89C	A89C	A89C	PM_VSU_4FLOP	four flops operation (scalar fdiv, fsqrt; DP vector version of fmadd, fnmadd, fmsub, fnmsub; SP vector versions of single flop instructions)	Thread
A8A0	A8A0	A8A0	A8A0	PM_VSU_8FLOP	eight flops operation (DP vector versions of fdiv,fsqrt and SP vector versions of fmadd,fnmadd,fmsub,fnmsub)	Thread
A8AC	A8AC	A8AC	A8AC	PM_VSU_DENORM	Vector or Scalar denorm operand	Thread
A8B0	A8B0	A8B0	A8B0	PM_VSU_FCONV	Convert instruction executed	Thread
A8B8	A8B8	A8B8	A8B8	PM_VSU_FEST	Estimate instruction executed	Thread
A8BC	A8BC	A8BC	A8BC	PM_VSU_FIN	VSU0 Finished an instruction	Thread
A884	A884	A884	A884	PM_VSU_FMA	two flops operation (fmadd, fnmadd, fmsub, fnmsub) Scalar instructions only!	Thread
A890	A890	A890	A890	PM_VSU_FMA_DOUBLE	DP vector version of fmadd,fnmadd,fmsub,fnmsub	Thread
A8B4	A8B4	A8B4	A8B4	PM_VSU_FRSP	Round to single precision instruction executed	Thread
A888	A888	A888	A888	PM_VSU_FSQRT_FDIV	four flops operation (fdiv,fsqrt) Scalar Instructions only!	Thread
A894	A894	A894	A894	PM_VSU_FSQRT_FDIV_DOUBLE	DP vector versions of fdiv,fsqrt	Thread
B888	B888	B888	B888	PM_VSU_SCALAR_DOUBLE_ISSUED	Double Precision scalar instruction issued on Pipe0	Thread
B884	B884	B884	B884	PM_VSU_SCALAR_SINGLE_ISSUED	Single Precision scalar instruction issued on Pipe0	Thread
B894	B894	B894	B894	PM_VSU_SIMPLE_ISSUED	Simple VMX instruction issued	Thread
A8A8	A8A8	A8A8	A8A8	PM_VSU_SINGLE	Vector or Scalar single precision	Thread
B88C	B88C	B88C	B88C	PM_VSU_STF	FPU store (SP or DP) issued on Pipe0	Thread
B880	B880	B880	B880	PM_VSU_VECTOR_DOUBLE_ISSUED	Double Precision vector instruction issued on Pipe0	Thread
B890	B890	B890	B890	PM_VSU_VECTOR_SINGLE_ISSUED	Single Precision vector instruction issued (executed)	Thread
A0A4	A0A4	A0A4	A0A4	PM_VSU0_16FLOP	Sixteen flops operation (SP vector versions of fdiv,fsqrt)	Thread
A080	A080	A080	A080	PM_VSU0_1FLOP	one flop (fadd, fmul, fsub, fcmp, fsel, fabs, fnabs, fres, fsqrte, fneg, xsadd, xsmul, xssub, xscmp, xssel, xsabs, xsnabs, xsre, xssqrte, xsneg) operation finished	Thread
A098	A098	A098	A098	PM_VSU0_2FLOP	two flops operation (scalar fmadd, fnmadd, fmsub, fnmsub and DP vector versions of single flop instructions)	Thread

Table 1-1. PMU Events

PMC1	PMC2	PMC3	PMC4	Event	Short Description	Doman
A08C	A08C	A08C	A08C	PM_VSU0_2FLOP_DOUBLE	two flop DP vector operation (xvadddp, xvmuldp, xvsubdp, xvcmpdp, xvseldp, xvabsdp, xvnabsdp, xvredp ,xvsqrtedp, vxnegdp)	Thread
A09C	A09C	A09C	A09C	PM_VSU0_4FLOP	four flops operation (scalar fdiv, fsqrt; DP vector version of fmadd, fnmadd, fmsub, fnmsub; SP vector versions of single flop instructions)	Thread
A0A0	A0A0	A0A0	A0A0	PM_VSU0_8FLOP	eight flops operation (DP vector versions of fdiv,fsqrt and SP vector versions of fmadd,fnmadd,fmsub,fnmsub)	Thread
B096	B096	B096	B096	PM_VSU0_COMPLEX_ISSUED	Complex VMX instruction issued	Thread
A0AC	A0AC	A0AC	A0AC	PM_VSU0_DENORM	FPU denorm operand	Thread
A0B0	A0B0	A0B0	A0B0	PM_VSU0_FCONV	Convert instruction executed	Thread
A0B8	A0B8	A0B8	A0B8	PM_VSU0_FEST	Estimate instruction executed	Thread
A0BC	A0BC	A0BC	A0BC	PM_VSU0_FIN	VSU0 Finished an instruction	Thread
A084	A084	A084	A084	PM_VSU0_FMA	two flops operation (fmadd, fnmadd, fmsub, fnmsub, xsmadd, xsnmadd, xsmsub, xsnmsub) Scalar instructions only!	Thread
A090	A090	A090	A090	PM_VSU0_FMA_DOUBLE	four flop DP vector operations (xvmadddp, xvnmadddp, xvmsubdp, xvmsubdp)	Thread
B09C	B09C	B09C	B09C	PM_VSU0_FPSCR	Move to/from FPSCR type instruction issued on Pipe 0	Thread
A0B4	A0B4	A0B4	A0B4	PM_VSU0_FRSP	Round to single precision instruction executed	Thread
A088	A088	A088	A088	PM_VSU0_FSQRT_FDIV	four flops operation (fdiv,fsqrt,xsdiv,xssqrt) Scalar Instructions only!	Thread
A094	A094	A094	A094	PM_VSU0_FSQRT_FDIV_DOUBLE	eight flop DP vector operations (xvfdivdp, xvsqrtdp	Thread
B088	B088	B088	B088	PM_VSU0_SCAL_DOUBLE_ISSUED	Double Precision scalar instruction issued on Pipe0	Thread
B084	B084	B084	B084	PM_VSU0_SCAL_SINGLE_ISSUED	Single Precision scalar instruction issued on Pipe0	Thread
B094	B094	B094	B094	PM_VSU0_SIMPLE_ISSUED	Simple VMX instruction issued	Thread
A0A8	A0A8	A0A8	A0A8	PM_VSU0_SINGLE	FPU single precision	Thread
B08C	B08C	B08C	B08C	PM_VSU0_STF	FPU store (SP or DP) issued on Pipe0	Thread
B080	B080	B080	B080	PM_VSU0_VECT_DOUBLE_ISSUED	Double Precision vector instruction issued on Pipe0	Thread
B090	B090	B090	B090	PM_VSU0_VECTOR_SP_ISSUED	Single Precision vector instruction issued (executed)	Thread
A082	A082	A082	A082	PM_VSU1_1FLOP	one flop (fadd, fmul, fsub, fcmp, fsel, fabs, fnabs, fres, fsqrte, fneg, xsadd, xsmul, xssub, xscmp, xssel, xsabs, xsnabs, xsre, xssqrte, xsneg) operation finished	Thread

Table 1-1. PMU Events

PMC1	PMC2	PMC3	PMC4	Event	Short Description	Doman
A09A	A09A	A09A	A09A	PM_VSU1_2FLOP	two flops operation (scalar fmadd, fnmadd, fmsub, fnmsub and DP vector versions of single flop instructions)	Thread
A08E	A08E	A08E	A08E	PM_VSU1_2FLOP_DOUBLE	two flop DP vector operation (xvadddp, xvmuldp, xvsubdp, xvcmpdp, xvseldp, xvabsdp, xvnabsdp, xvredp ,xvsqrtedp, vxnegdp)	Thread
A09E	A09E	A09E	A09E	PM_VSU1_4FLOP	four flops operation (scalar fdiv, fsqrt; DP vector version of fmadd, fnmadd, fmsub, fnmsub; SP vector versions of single flop instructions)	Thread
A0A2	A0A2	A0A2	A0A2	PM_VSU1_8FLOP	eight flops operation (DP vector versions of fdiv,fsqrt and SP vector versions of fmadd,fnmadd,fmsub,fnmsub)	Thread
B098	B098	B098	B098	PM_VSU1_DD_ISSUED	64BIT Decimal Issued on Pipe1	Thread
A0AE	A0AE	A0AE	A0AE	PM_VSU1_DENORM	FPU denorm operand	Thread
B09A	B09A	B09A	B09A	PM_VSU1_DQ_ISSUED	128BIT Decimal Issued on Pipe1	Thread
A0B2	A0B2	A0B2	A0B2	PM_VSU1_FCONV	Convert instruction executed	Thread
A0BA	A0BA	A0BA	A0BA	PM_VSU1_FEST	Estimate instruction executed	Thread
A0BE	A0BE	A0BE	A0BE	PM_VSU1_FIN	VSU1 Finished an instruction	Thread
A086	A086	A086	A086	PM_VSU1_FMA	two flops operation (fmadd, fnmadd, fmsub, fnmsub, xsmadd, xsnmadd, xsmsub, xsnmsub) Scalar instructions only!	Thread
A092	A092	A092	A092	PM_VSU1_FMA_DOUBLE	four flop DP vector operations (xvmadddp, xvnmadddp, xvmsubdp, xvmsubdp)	Thread
A0B6	A0B6	A0B6	A0B6	PM_VSU1_FRSP	Round to single precision instruction executed	Thread
A08A	A08A	A08A	A08A	PM_VSU1_FSQRT_FDIV	four flops operation (fdiv,fsqrt,xsdiv,xssqrt) Scalar Instructions only!	Thread
A096	A096	A096	A096	PM_VSU1_FSQRT_FDIV_DOUBLE	eight flop DP vector operations (xvfdivdp, xvsqrtdp	Thread
B092	B092	B092	B092	PM_VSU1_PERMUTE_ISSUED	Permute VMX Instruction Issued	Thread
B08A	B08A	B08A	B08A	PM_VSU1_SCAL_DOUBLE_ISSUED	Double Precision scalar instruction issued on Pipe1	Thread
B086	B086	B086	B086	PM_VSU1_SCAL_SINGLE_ISSUED	Single Precision scalar instruction issued on Pipe1	Thread
A0AA	A0AA	A0AA	A0AA	PM_VSU1_SINGLE	FPU single precision	Thread
B09E	B09E	B09E	B09E	PM_VSU1_SQ	Store Vector Issued on Pipe1	Thread
B08E	B08E	B08E	B08E	PM_VSU1_STF	FPU store (SP or DP) issued on Pipe1	Thread
B082	B082	B082	B082	PM_VSU1_VECT_DOUBLE_ISSUED	Double Precision vector instruction issued on Pipe1	Thread

Appendix B. Detailed Event Descriptions

Event Name	Event Description	Detailed Description
PM_1PLUS_PPC_CMPL	1 or more ppc insts finished	A group containing at least one PowerPC instruction completed. For microcoded instructions that span multiple groups, this will only occur once.
PM_1PLUS_PPC_DISP	Cycles at least one Instr Dispatched,	A group containing at least one PowerPC instruction dispatched. For microcoded instructions that span multiple groups, this will only occur once.
PM_1THRD_CON_RUN_INSTR	1 thread Concurrent Run Instructions	Cycles where were 1 hread had its run latch set, and instructions completed for this thread
PM_ANY_THRD_RUN_CYC	One of threads in run_cycles	At least one thread has set its run latch. Operating systems use the run latch to indicate when they are doing useful work. The run latch is typically cleared in the OS idle loop. This event does not respect FCWAIT.
PM_BC_PLUS_8_CONV	BC+8 Converted	A BC+8 instruction was cracked into a Resolve instruction to evaluate the condition (which never redirects) and a conditional instruction that will cancel its execution based on the Resolve outcome.
PM_BC_PLUS_8_RSLV_TAKEN	BC+8 Resolve outcome was Taken, resulting in the conditional instruction being canceled	Resolve outcome was Taken, resulting in the conditional instruction being canceled
PM_BR_MPRED	Number of Branch Mispredicts	A branch instruction was predicted. This could have been a target prediction a condition prediction or both
PM_BR_MPRED_CCACHE	Branch Mispredict due to Count Cache prediction	A branch instruction target was incorrectly predicted due to count cache misprediction. This will result in a branch mispredict flush unless a flush is detected from an older instruction.
PM_BR_MPRED_CR	Branch mispredict - taken/not taken	A conditional branch instruction was incorrectly predicted as taken or not taken. The branch execution unit detects a branch mispredict because the CR value is opposite of the predicted value. This will result in a branch redirect flush if not overfidden by a flush of an older instruction.
PM_BR_MPRED_CR_TA	Branch mispredict - taken/not taken and target	A conditional branch instruction was incorrectly predicted as taken or not taken and target address was also predicted.
PM_BR_MPRED_LSTACK	Branch Mispredict due to Link Stack	A branch instruction target was incorrectly predicted due to Link stack Misprediction. This will result in a branch mispredict flush unless a flush is detected from an older instruction.
PM_BR_MPRED_TA	Branch mispredict - target address	A branch instruction target was incorrectly predicted. This will result in a branch mispredict flush unless a flush is detected from an older instruction.
PM_BR_PRED	Branch Predictions made	A branch prediction was made, counted at branch execute time.
PM_BR_PRED_CCACHE	Count Cache Predictions	A branch instruction target prediction was made using the count cache, counted at branch execute time
PM_BR_PRED_CR	Branch predict - taken/not taken	A conditional branch instruction was predicted as taken or not taken,counted at branch execute time
PM_BR_PRED_CR_TA	Branch predict - taken/not taken and target	A conditional branch instruction was predicted as taken or not taken and target address was also predicted, counted at branch execute time
PM_BR_PRED_LSTACK	Link Stack Predictions	A branch instruction target prediction was made using the Link stack, counted at branch execute time
PM_BR_PRED_TA	Branch predict - target address	A branch instruction target was predicted, counted at branch execute time
PM_BR_TAKEN	Branch Taken	A branch instruction was taken. This could have been aconditional branch or an unconditional branch, counted at branch execute time

Event Name	Event Description	Detailed Description
PM_BR_UNCOND	Unconditional Branch	A unconditional branch was executed
PM_BRU_FIN	Branch Instruction Finished	The branch unit finished a branch type instruction.
PM_BTAC_HIT	BTAC Correct Prediction	The branch target address cache correctly predicted a taken branch
PM_BTAC_MISS	BTAC Mispredicted	Brach Target address Cache Mispredicted This event can come on for two cases. The first case the BTAC can be wrong is if the branch is predicted taken, but there is no taken branch in the fetch group. When this occurs there will be 3 bubble cycles between the fetch of the group in question and the fetch of the next sequential group. The next case is when the BTAC predicts a branch and there is a branch in that fetch group, but that target address is wrong. This can occur because a different branch in the fetch group was taken, or because of aliasing .When the address is wrong the branch is always fetched and the instructions after the branch are discarded. The target will be fetched 4 cycles after the branch instruction in this case(3 bubble cycles), This is one cycle worse then if the BTAC had not predicted the branch.
PM_CMPLU_STALL	No groups completed, GCT not empty	Cycles where a thread was not completing any groups , when the group completion table had entries for that thread.
PM_CMPLU_STALL_BRU	Completion stall due to BRU	Following a completion stall (any period when no groups completed, while group completion table was not empty for that thread) the last instruction to finish before completion resumes was from the Branch Unit. This mostly occurs where the branch has dependencies on a long latency instruction such as a load.
PM_CMPLU_STALL_DCACHE_MISS	Completion stall caused by D cache miss	Following a completion stall (any period when no groups completed, while group completion table was not empty for that thread) the last instruction to finish before completion resumes suffered a Data Cache Miss. Data Cache Miss has higher priority than any other Load/Store delay, so if an instruction encounters multiple delays only the Data Cache Miss will be reported and the entire delay period will be charged to Data Cache Miss.
PM_CMPLU_STALL_DFU	Completion stall caused by Decimal Floating Point Unit	Following a completion stall (any period when no groups completed, while group completion table was not empty for that thread) the last instruction to finish before completion resumes was from the Decimal Floating Point Unit.
PM_CMPLU_STALL_DIV	Completion stall caused by DIV instruction	Following a completion stall (any period when no groups completed, while group completion table was not empty for that thread) the last instruction to finish before completion resumes was a fixed point divide instruction.
PM_CMPLU_STALL_END_GCT_NOSLOT	Count ended because GCT went empty	cycles where a completion stall ended because the group completion table went empty.
PM_CMPLU_STALL_ERAT_MISS	Completion stall caused by ERAT miss	Following a completion stall (any period when no groups completed, while group completion table was not empty for that thread)) the last instruction to finish before completion resumes suffered an ERAT miss.
PM_CMPLU_STALL_FXU	Completion stall caused by FXU instruction	Following a completion stall (any period when no groups completed, while group completion table was not empty for that thread) the last instruction to finish before completion resumes was from the Fixed Point Unit.
PM_CMPLU_STALL_IFU	Completion stall due to IFU	Following a completion stall (any period when no groups completed, while group completion table was not empty for that thread) the last instruction to finish before completion resumes was from the Instruction fetch unit (either Branch Unit or CR unit).

Event Name	Event Description	Detailed Description
PM_CMPLU_STALL_LSU	Completion stall caused by LSU instruction	Following a completion stall (any period when no groups completed, while group completion table was not empty for that thread) the last instruction to finish before completion resumes was from the Load store Unit.
PM_CMPLU_STALL_REJECT	Completion stall caused by reject	Following a completion stall (any period when no groups completed, while group completion table was not empty for that thread) the last instruction to finish before completion resumes suffered a load/store reject.
PM_CMPLU_STALL_SCALAR	Completion stall caused by FPU instruction	Following a completion stall (any period when no groups completed, while group completion table was not empty for that thread) the last instruction to finish before completion resumes was a scalar floating point instruction.
PM_CMPLU_STALL_SCALAR_LONG	Completion stall caused by long latency scalar instruction	Following a completion stall (any period when no groups completed, while group completion table was not empty for that thread) the last instruction to finish before completion resumes was a floating point divide or square root instruction.
PM_CMPLU_STALL_STORE	Completion stall due to store instruction	Following a completion stall (any period when no groups completed, while group completion table was not empty for that thread) the last instruction to finish before completion resumes was a store. This generally happens when we run out of real SRQ entries, which prevents stores from issuing.
PM_CMPLU_STALL_THRD	Completion Stalled due to thread conflict. Group ready to complete but it was another thread's turn	Following a completion stall (any period when no groups completed, while group completion table was not empty for that thread) the thread could not complete a group because the completion port its sharing was being used by another thread. In SMT4 mode Thread0 and thread2 share a completion port and Thread1 and Thread3 share another completion port
PM_CMPLU_STALL_VECTOR	Completion stall caused by Vector instruction	Following a completion stall (any period when no groups completed, while group completion table was not empty for that thread) the last instruction to finish before completion resumes was a vector instruction.
PM_CMPLU_STALL_VECTOR_LONG	completion stall due to long latency vector instruction	Following a completion stall (any period when no groups completed, while group completion table was not empty for that thread) the last instruction to finish before completion resumes was a long latency vector instruction.
PM_CYC	Cycles	Processor cycles
PM_DATA_FROM_DL2L3_MOD	Data loaded from distant L2 or L3 modified	The processor's Data Cache was reloaded with Modified (M) data from a chip's L2 or L3 on a different Node (Distant) due to a demand load .
PM_DATA_FROM_DL2L3_MOD	Data loaded from distant L2 or L3 modified	The processor's Data Cache was reloaded with Modified (M) data from a chip's L2 or L3 on a different Node (Distant) due to a demand load .
PM_DATA_FROM_DL2L3_SHR	Data loaded from distant L2 or L3 shared	The processor's Data Cache was reloaded with Shared (S) data from a chip's L2 or L3 on a different Node (Distant) due to a demand load .
PM_DATA_FROM_DMEM	Data loaded from distant memory	The processor's Data Cache was reloaded with data from a chip's memory on a different Node (Distant) due to a demand load .
PM_DATA_FROM_DMEM	Data loaded from distant memory	The processor's Data Cache was reloaded with data from a chip's memory on a different Node (Distant) due to a demand load .
PM_DATA_FROM_L2	Data loaded from L2	The processor's Data Cache was reloaded with data from the local chiplet's L2 cache due to a demand load.
PM_DATA_FROM_L21_MOD	Data loaded from another L2 on same chip modified	The processor's Data Cache was reloaded with Modified (M) data from another L2 on the same chip due to a demand load .

Event Name	Event Description	Detailed Description
PM_DATA_FROM_L21_SHR	Data loaded from another L2 on same chip shared	The processor's Data Cache was reloaded with Shared (S) data from another L2 on the same chip due to a demand load .
PM_DATA_FROM_L21_SHR	Data loaded from another L2 on same chip shared	The processor's Data Cache was reloaded with Shared (S) data from another L2 on the same chip due to a demand load .
PM_DATA_FROM_L2MISS	Demand LD - L2 Miss (not L2 hit)	The processor's Data Cache was reloaded but not from the local L2.
PM_DATA_FROM_L2MISS	Data loaded missed L2	The processor's Data Cache was reloaded but not from the local L2 due to a demand load.
PM_DATA_FROM_L3	Data loaded from L3	The processor's Data Cache was reloaded with data from the local chiplet's L3 cache due to a demand load.
PM_DATA_FROM_L3	Data loaded from L3	The processor's Data Cache was reloaded with data from the local chiplet's L3 cache due to a demand load.
PM_DATA_FROM_L31_MOD	Data loaded from another L3 on same chip modified	The processor's Data Cache was reloaded with Modified (M) data from another L3 on the same chip due to a demand load .
PM_DATA_FROM_L31_SHR	Data loaded from another L3 on same chip shared	The processor's Data Cache was reloaded with Shared (S) data from another L3 on the same chip due to a demand load .
PM_DATA_FROM_L31_SHR	Data loaded from another L3 on same chip shared	The processor's Data Cache was reloaded with Shared (S) data from another L3 on the same chip due to a demand load .
PM_DATA_FROM_L3MISS	Demand LD - L3 Miss (not L2 hit and not L3 hit)	The processor's Data Cache was reloaded but not from the local L3
PM_DATA_FROM_L3MISS	Data loaded from private L3 miss	The processor's Data Cache was reloaded but not from the local L3 due to a demand load.
PM_DATA_FROM_LMEM	Data loaded from local memory	The processor's Data Cache was reloaded with data from the local chip's memory due to a demand load.
PM_DATA_FROM_LMEM	Data loaded from local memory	The processor's Data Cache was reloaded with data from the local chip's memory due to a demand load.
PM_DATA_FROM_RL2L3_MOD	Data loaded from remote L2 or L3 modified	The processor's Data Cache was reloaded with Modified (M) data from a chip's L2 or L3 on the same Node (Remote) due to a demand load .
PM_DATA_FROM_RL2L3_SHR	Data loaded from remote L2 or L3 shared	The processor's Data Cache was reloaded with Shared (S) data from a chip's L2 or L3 on the same Node (Remote) due to a demand load .
PM_DATA_FROM_RL2L3_SHR	Data loaded from remote L2 or L3 shared	The processor's Data Cache was reloaded with Shared (S) data from a chip's L2 or L3 on the same Node (Remote) due to a demand load .
PM_DATA_FROM_RMEM	Data loaded from remote memory	The processor's Data Cache was reloaded with data from another chip's memory on the same node due to a demand load.
PM_DATA_TABLEWALK_CYC	Data Tablewalk Active	Cycles a translation tablewalk is active. While a tablewalk is active any request attempting to access the TLB will be rejected and retried.
PM_DC_PREF_DST	Data Stream Touch	A Data Stream Touch was initiated by software
PM_DERAT_MISS_16G	DERAT misses for 16G page	The source page size of a DERAT reload is 16G
PM_DERAT_MISS_16M	DERAT misses for 16M page	The source page size of a DERAT reload is 16M
PM_DERAT_MISS_4K	DERAT misses for 4K page	The source page size of a DERAT reload is 4K
PM_DERAT_MISS_64K	DERAT misses for 64K page	The source page size of a DERAT reload is 64K
PM_DISP_CLB_HELD	CLB Hold: Any Reason	An instruction group is held in the Ibuffer for reasons such as dispatch flush settings in TSCR.

Event Name	Event Description	Detailed Description
PM_DISP_CLB_HELD_BAL	Dispatch/CLB Hold: Balance	An instruction group is held at dispatch in the ibuffer, due to thread balance. This includes waiting for balance flush and waiting for thread balance conditions to clear such as an outstanding TLB/L2 miss. Thread balance is controlled in TSCR
PM_DISP_CLB_HELD_RES	Dispatch/CLB Hold: Resource	An instruction group is held at dispatch in the ibuffer, due to any resource full condition including BRQ Full, CR Issue Q full, Unified Issue Q Full, Mapper full, LRQ Full, STQ full, or other resource holds such as syncs tlbie
PM_DISP_CLB_HELD_SB	Dispatch/CLB Hold: Scoreboard	An instruction group is held at dispatch in the ibuffer, due to scoreboard full
PM_DISP_CLB_HELD_SYNC	Dispatch/CLB Hold: Sync type instruction	An instruction group is held at dispatch in the ibuffer due to sync type instruction
PM_DISP_CLB_HELD_TLBIE	Dispatch Hold: Due to TLBIE	An instruction group is held at dispatch in the ibuffer due to TLBIE
PM_DISP_HELD	Dispatch Held	Cycles dispatch to pipeline was held
PM_DISP_HELD_THERMAL	Dispatch Held due to Thermal	Cycles dispatch to pipeline was held due to a thermal condition
PM_DISP_WT	Dispatched Starved (not held, nothing to dispatch)	Cycles dispatch to pipeline was not held and there were no groups to dispatch
PM_DPU_HELD_POWER	Dispatch Held due to Power Management	Cycles dispatch to pipeline was held due to power management
PM_DSEG	DSEG Exception	A DSEG exception because of a DSLB miss has been triggred. The OS handler is invoked to resolve the SLB miss
PM_DSLB_MISS	Data SLB Miss - Total of all segment sizes	A SLB miss for a data request occurred. SLB misses trap to the operating system to resolve. This is a total count for all segment sizes.
PM_DTLB_MISS	TLB reload valid	Data TLB misses, all page sizes.
PM_DTLB_MISS_16G	Data TLB miss for 16G page	Data TLB references to 16G pages that missed the TLB. Page size is determined at TLB reload time.
PM_DTLB_MISS_16M	Data TLB miss for 16M page	Data TLB references to 16MB pages that missed the TLB. Page size is determined at TLB reload time.
PM_DTLB_MISS_4K	Data TLB miss for 4K page	Data TLB references to 4K pages that missed the TLB. Page size is determined at TLB reload time.
PM_DTLB_MISS_4K	Data TLB miss for 4K page	Data TLB references to 4K pages that missed the TLB. Page size is determined at TLB reload time.
PM_DTLB_MISS_64K	Data TLB miss for 64K page	Data TLB references to 64K pages that missed the TLB. Page size is determined at TLB reload time.
PM_EE_OFF_EXT_INT	ee off and external interrupt	Cycles when an interrupt due to an external exception is pending but external exceptions were masked
PM_EXT_INT	external interrupt	An interrupt due to an external exception occurred
PM_FLOP	Floating Point Operation Finished	The vector scalar unit has finished a floating point operation
PM_FLUSH	Flush (any type)	A flush has occurred, this could includes all types of flushes including branch mispredicts, load store unit flushes,partial flushes,completion flushes
PM_FLUSH_BR_MPRED	Flush caused by branch mispredict	A flush was caused by a branch mispredict
PM_FLUSH_COMPLETION	Completion Flush	A completion flush has occurred
PM_FLUSH_DISP	Dispatch flush	A dispatch flush occurred. A dispatch flush is a low latency flush that flushes the decode pipe. Sync, lwsync, ptesync, tlbsync, tlbie, and instructions with the scoreboard bit set can cause a dispatch flush on P7. Also, if enabled, a thread that was balanced flushed will be dispatch flushed if the chosen thread is stalled at dispatch

Event Name	Event Description	Detailed Description
PM_FLUSH_DISP_SB	Dispatch Flush: Scoreboard	An instruction with scoreboard bit set caused a dispatch flush to flush the decode pipe
PM_FLUSH_DISP_SYNC	Dispatch Flush: Sync	A sync/lwsync/ptesync/tlbsync caused a dispatch flush to flush the decode pipe
PM_FLUSH_DISP_TLBIE	Dispatch Flush: TLBIE	A TLBIE caused a dispatch flush to flush the decode pipe
PM_FLUSH_PARTIAL	Partial flush	A partial flush for a group for branch misprediction (one per cycle), since the first branch may be older than other instructions in the group
PM_FREQ_DOWN	Frequency is being slewed down due to Power Management	Frequency of the core was slewed up due to power management
PM_FREQ_UP	Power Management: Above Threshold A	Frequency of the core was slewed down due to power management
PM_FXU_BUSY	fxu0 busy and fxu1 busy.	Cycles when both fixed point uints are both busy
PM_FXU_IDLE	fxu0 idle and fxu1 idle	FXU0 and FXU1 are both idle
PM_FXU0_BUSY_FXU1_IDLE	fxu0 busy and fxu1 idle	Cycles when FXU0 busy and FXU1 idle
PM_FXU0_FIN	FXU0 Finished	The fixed point unit Unit 0 finished an instruction. Instructions that finish may not necessary complete.
PM_FXU1_BUSY_FXU0_IDLE	fxu0 idle and fxu1 busy.	Cycles when FXU0 idle and FXU1 busy
PM_FXU1_FIN	FXU1 Finished	The fixed point unit Unit 1 finished an instruction. Instructions that finish may not necessary complete.
PM_GCT_EMPTY_CYC	GCT empty, all threads	The Global Completion Table is completely empty for all threads
PM_GCT_FULL_CYC	Cycles No room in EAT	Cycles the Effective Address Table is full. When EAT is full for a thread, fetch is stalled. Thus, no new instructions for that thread will be fetched until an Address Group has been completed to free up a new entry.
PM_GCT_NOSLOT_BR_MPRED	GCT empty by branch mispredict	Cycles when the Global Completion Table has no slots from this thread because of a branch misprediction
PM_GCT_NOSLOT_BR_MPRED_IC_MISS	GCT empty by branch mispredict + IC miss	Cycles when the Global Completion Table has no slots from this thread because of a branch misprediction and Instruction cache Miss
PM_GCT_NOSLOT_CYC	No itags assigned	Cycles when the Global Completion Table has no slots from this thread.
PM_GCT_NOSLOT_IC_MISS	GCT empty by I cache miss	Cycles when the Global Completion Table has no slots from this thread because of an Instruction Cache miss.
PM_GCT_UTIL_1to2_SLOTS	GCT Utilization 1-2 entries	Cycles when the Global Completion Table has between 1-2 of its slots used for this thread. The GCT has 20 entries shared between threds.
PM_GCT_UTIL_11plus_SLOTS	GCT Utilization 11+ entries	Cycles when the Global Completion Table has 11 or more slots used for this thread. The GCT has 20 entries shared between threds.
PM_GCT_UTIL_3to6_SLOTS	GCT Utilization 3-6 entries	Cycles when the Global Completion Table has between 3-6 of its slots used for this thread. The GCT has 20 entries shared between threds.
PM_GCT_UTIL_7to10_SLOTS	GCT Utilization 7-10 entries	Cycles when the Global Completion Table has between 7-10 of its slots used for this thread. The GCT has 20 entries shared between threds.
PM_GRP_BR_MPRED_NONSPEC	Group experienced non-speculative branch redirect	Number of groups, counted at completion, that have encountered a branch redirect.
PM_GRP_CMPL	group completed	A group completed. Microcoded instructions that span multiple groups will generate this event once per group.

Event Name	Event Description	Detailed Description
PM_GRP_DISP	dispatch_success (Group Dispatched)	A group dispatched. Microcoded instructions that span multiple groups will generate this event once per group.
PM_GRP_IC_MISS_NONSPEC	Group experienced non-speculative I cache miss	Number of groups, counted at completion, that have encountered a Icache Miss.
PM_GRP_MRK	IDU Marked Instruction	A group was sampled (marked). The group is called a marked group. One instruction within the group is tagged for detailed monitoring. The sampled instruction is called a marked instructions. Events associated with the marked instruction are annotated with the marked term.
PM_GRP_MRK_CYC	cycles IDU marked instruction before dispatch	Cycles instruction dispatch unit marked an instruction
PM_HV_CYC	cycles in hypervisor mode	Cycles when the processor is executing in Hypervisor (MSR[HV] = 1 and MSR[PR]=0)
PM_IBUF_FULL_CYC	Cycles No room in ibuff	Cycles The Instruction fetch buffer is full .
PM_IC_BANK_CONFLICT	Read blocked due to interleave conflict.	Counts the number of reads that were blocked due to Icache interleave conflict. The icache is 8 way interleaved. This will allow reads and write to the cache at the same time as long as they are not to the same interleave. When a conflict occurs between a read and a write the write succeeds, but the read gets the wrong data. The Instruction fetch address register will detect an interleave conflict and kill the data that was read that cycle and will also kill the ifar increment.
PM_IC_DEMAND_CYC	Cycles when a demand ifetch was pending	Cycles where a demand instruction fetch was pending
PM_IC_DEMAND_L2_BHT_REDIRECT	L2 I cache demand request due to BHT redirect	An instruction cache demand request was sent to the L2, because of a BHT redirect. The branch history table predicted that the fetch group contains a branch that is taken
PM_IC_DEMAND_L2_BR_ALL	L2 I cache demand request due to BHT or redirect	Icache reques to L2 due to BHT redirect or branch flush
PM_IC_DEMAND_L2_BR_REDIRECT	L2 I cache demand request due to branch redirect	An instruction cache demand request was sent to the L2, because of a branch flush.
PM_IC_DEMAND_REQ	Demand Instruction fetch request	A demand instruction fetch request to the L2 , this is pulsed for every request.
PM_IC_PREF_CANCEL_ALL	Prefetch Canceled due to page boundary or icache hit	An instruction prefetch request was cancelled due to icache hit or cache line crossing
PM_IC_PREF_CANCEL_HIT	Prefetch Canceled due to icache hit	A request to prefetch a line to the icache was cancelled because of an icache hit. Instruction prefetch is disabled in SMT4 mode.
PM_IC_PREF_CANCEL_L2	L2 Squashed request	The L2 squashed either a demand or prefetch request that was no longer needed by the Instruction fetch unit.
PM_IC_PREF_CANCEL_PAGE	Prefetch Canceled due to page boundary	A request to prefetch a line to the icache was cancelled because of a page boundary crossing
PM_IC_PREF_REQ	Instruction prefetch requests	An instruction prefetch request was made to the L2, not valid in SMT4 mode.
PM_IC_PREF_WRITE	Instruction prefetch written into IL1	Count of instruction prefetch sectors written in the L1. Writes are always written as LRU and subsequent read makes it MRU
PM_IC_RELOAD_SHR	Reloading line to be shared between the threads	Count of lines that were reloaded that are valid for multiple threads. This indicates multiple threads are accessing the same line.
PM_IC_REQ_ALL	Icache requests, prefetch + demand	An instruction prefetch request or demand was made to the L2
PM_IC_WRITE_ALL	Icache sectors written, prefetch + demand	Count of instruction prefetch or demand sectors written in the L1. Writes are always written as LRU and subsequent read makes it MRU

Event Name	Event Description	Detailed Description
PM_IERAT_MISS	IERAT Miss (Not implemented as DI on POWER6)	An entry was written into the IERAT as a result of an IERAT miss.
PM_IERAT_WR_64K	large page 64k	An entry was written into the IERAT as a result of an IERAT miss for a 64k page. This event can be used to count IERAT misses to 64k pages.
PM_IERAT_XLATE_WR_16MPLUS	large page 16M+	An entry was written into the IERAT as a result of an IERAT miss for a 16M+ page size. This event can be used to count IERAT misses to 16M+ pages.
PM_IFU_FIN	IFU Finished a (non-branch) instruction	The Instruction fetch unit finished a non branch instruction
PM_INST_CMPL	# PPC Instructions completed	Number of PowerPC Instructions that completed.
PM_INST_DISP	# PPC Dispatched	Number of PowerPC instructions successfully dispatched
PM_INST_DISP	Instructions dispatched	Number of PowerPC instructions successfully dispatched
PM_INST_FROM_DL2L3_MOD	Instruction fetched from distant L2 or L3 modified	The processor's Instruction Cache was reloaded with Modified (M) data from a chip's L2 or L3 on a different Node (Distant) due to a demand load .
PM_INST_FROM_DL2L3_MOD	Instruction fetched from distant L2 or L3 modified	The processor's Instruction Cache was reloaded with Modified (M) data from a chip's L2 or L3 on a different Node (Distant) due to a demand load .
PM_INST_FROM_DL2L3_SHR	Instruction fetched from distant L2 or L3 shared	The processor's Instruction Cache was reloaded with Shared (S) data from a chip's L2 or L3 on a different Node (Distant) due to a demand load .
PM_INST_FROM_DMEM	Instruction fetched from distant memory	The processor's Instruction Cache was reloaded with data from a chip's memory on a different Node (Distant) due to a demand load .
PM_INST_FROM_DMEM	Instruction fetched from distant memory	The processor's Instruction Cache was reloaded with data from a chip's memory on a different Node (Distant) due to a demand load .
PM_INST_FROM_L1	Instruction fetches from L1	An instruction fetch group was fetched from L1. Fetch Groups can contain up to 8 instructions
PM_INST_FROM_L2	Instruction fetched from L2	The processor's Instruction Cache was reloaded with data from the local chiplet's L2 cache due to a demand load.
PM_INST_FROM_L21_MOD	Instruction fetched from another L2 on same chip modified	The processor's Instruction Cache was reloaded with Modified (M) data from another L2 on the same chip due to a demand load .
PM_INST_FROM_L21_SHR	Instruction fetched from another L2 on same chip shared	The processor's Instruction Cache was reloaded with Shared (S) data from another L2 on the same chip due to a demand load .
PM_INST_FROM_L21_SHR	Instruction fetched from another L2 on same chip shared	The processor's Instruction Cache was reloaded with Shared (S) data from another L2 on the same chip due to a demand load .
PM_INST_FROM_L2MISS	Instruction fetched missed L2	The processor's Instruction Cache was reloaded but not from the local L2 due to a demand load.
PM_INST_FROM_L3	Instruction fetched from L3	The processor's Instruction Cache was reloaded with data from the local chiplet's L3 cache due to a demand load.
PM_INST_FROM_L3	Instruction fetched from L3	The processor's Instruction Cache was reloaded with data from the local chiplet's L3 cache due to a demand load.
PM_INST_FROM_L31_MOD	Instruction fetched from another L3 on same chip modified	The processor's Instruction Cache was reloaded with Modified (M) data from another L3 on the same chip due to a demand load .
PM_INST_FROM_L31_SHR	Instruction fetched from another L3 on same chip shared	The processor's Instruction Cache was reloaded with Shared (S) data from another L3 on the same chip due to a demand load .

Event Name	Event Description	Detailed Description
PM_INST_FROM_L31_SHR	Instruction fetched from another L3 on same chip shared	The processor's Instruction Cache was reloaded with Shared (S) data from another L3 on the same chip due to a demand load .
PM_INST_FROM_L3MISS	Instruction fetched missed L3	The processor's Instruction Cache was reloaded but not from the local L3 due to a demand load.
PM_INST_FROM_LMEM	Instruction fetched from local memory	The processor's Instruction Cache was reloaded with data from the local chip's memory due to a demand load.
PM_INST_FROM_LMEM	Instruction fetched from local memory	The processor's Instruction Cache was reloaded with data from the local chip's memory due to a demand load.
PM_INST_FROM_PREF	Instruction fetched from prefetch	The processor's Instruction Cache was reloaded with data because on an instruction prefetch.
PM_INST_FROM_RL2L3_MOD	Instruction fetched from remote L2 or L3 modified	The processor's Instruction Cache was reloaded with Modified (M) data from a chip's L2 or L3 on the same Node (Remote) due to a demand load .
PM_INST_FROM_RL2L3_SHR	Instruction fetched from remote L2 or L3 shared	The processor's Instruction Cache was reloaded with Shared (S) data from a chip's L2 or L3 on the same Node (Remote) due to a demand load .
PM_INST_FROM_RL2L3_SHR	Instruction fetched from remote L2 or L3 shared	The processor's Instruction Cache was reloaded with Shared (S) data from a chip's L2 or L3 on the same Node (Remote) due to a demand load .
PM_INST_FROM_RMEM	Instruction fetched from remote memory	The processor's Instruction Cache was reloaded with data from another chip's memory on the same node due to a demand load.
PM_INST_IMC_MATCH_CMPL	IMC Match Count	Number of PowerPC instructions completed that matched the Instruction Match CAM
PM_INST_IMC_MATCH_DISP	IMC Matches dispatched	PowerPC instructions that were dispatched that matched the Instruction Match CAM
PM_INST_PTEG_FROM_DL2L3_MOD	Instruction PTEG loaded from distant L2 or L3 modified	A Page Table Entry was loaded into the TLB with modified (M) data from a chip's L2 or L3 on a different Node (Distant) due to a instruction side request.
PM_INST_PTEG_FROM_DL2L3_SHR	Instruction PTEG loaded from remote L2 or L3 shared	A Page Table Entry was loaded into the TLB with Shared (S) data from a chip's L2 or L3 on a different Node (Distant) due to a instruction side request.
PM_INST_PTEG_FROM_DMEM	Instruction PTEG loaded from distant memory	A Page Table Entry was loaded into the TLB with data from a chip's memory on a different Node due to a instruction side request.
PM_INST_PTEG_FROM_L2	Instruction PTEG loaded from L2	A Page Table Entry was loaded into the TLB with data from local chiplet's L2 due to a instruction side request.
PM_INST_PTEG_FROM_L21_MOD	Instruction PTEG loaded from another L2 on same chip modified	A Page Table Entry was loaded into the TLB with Modified (M) data from another L2 on the same chip due to a instruction side request.
PM_INST_PTEG_FROM_L21_SHR	Instruction PTEG loaded from another L2 on same chip shared	A Page Table Entry was loaded into the TLB with Shared (S) data from another L2 on the same chip due to a instruction side request.
PM_INST_PTEG_FROM_L2MISS	Instruction PTEG loaded from L2 miss	A Page Table Entry was loaded into the TLB with data not from local chiplet's L2 due to a instruction side request.
PM_INST_PTEG_FROM_L3	Instruction PTEG loaded from L3	A Page Table Entry was loaded into the TLB with data from local chiplet's L3 due to a instruction side request.
PM_INST_PTEG_FROM_L31_MOD	Instruction PTEG loaded from another L3 on same chip modified	A Page Table Entry was loaded into the TLB with Modified (M) data from another L3 on the same chip due to a instruction side request.
PM_INST_PTEG_FROM_L31_SHR	Instruction PTEG loaded from another L3 on same chip shared	A Page Table Entry was loaded into the TLB with Shared (S) data from another L3 on the same chip due to a instruction side request.

Event Name	Event Description	Detailed Description
PM_INST_PTEG_FROM_L3MISS	Instruction PTEG loaded from L3 miss	A Page Table Entry was loaded into the TLB with data not from local chiplet's L3 due to a instruction side request.
PM_INST_PTEG_FROM_LMEM	Instruction PTEG loaded from local memory	A Page Table Entry was loaded into the TLB with data from local chip's memory due to a instruction side request.
PM_INST_PTEG_FROM_RL2L3_MOD	Instruction PTEG loaded from remote L2 or L3 modified	A Page Table Entry was loaded into the TLB with modified (M) data from a chip's L2 or L3 on same Node (Remote) due to a instruction side request.
PM_INST_PTEG_FROM_RL2L3_SHR	Instruction PTEG loaded from remote L2 or L3 shared	A Page Table Entry was loaded into the TLB with Shared (S) data from a chip's L2 or L3 on same Node (Remote) due to a instruction side request.
PM_INST_PTEG_FROM_RMEM	Instruction PTEG loaded from remote memory	A Page Table Entry was loaded into the TLB with data from another chip's memory on same node due to a instruction side request.
PM_IOPS_CMPL	Internal Operations completed	Number of internal operations that completed.
PM_IOPS_DISP	IOPS dispatched	Number of internal operations that dispatched
PM_ISEG	ISEG Exception	A ISEG exception because of a ISLB miss has been triggred. The OS handler is invoked to resolve the SLB miss
PM_ISLB_MISS	Instruction SLB Miss - Tota of all segment sizes	A SLB miss for an instruction fetch has occurred.SLB misses trap to the operating system to resolve. This is a total count for all segment sizes.
PM_ITLB_MISS	ITLB Reloaded (always zero on POWER6)	A TLB miss for an Instruction Fetch has occurred
PM_L1_DCACHE_RELOAD_VALID	L1 reload data source valid	The data source information is valid, the data cache has been reloaded for demand loads , reported once per cacheline
PM_L1_DEMAND_WRITE	Instruction Demand sectors wriittent into IL1	Count of instruction demand sectors written in the L1. Writes are always written as LRU and subsequent read makes it MRU
PM_L1_ICACHE_MISS	Demand iCache Miss	An instruction fetch missed the L1 Instruction cache
PM_L1_PREF	L1 Prefetches	A request to prefetch a line into the L1 was made from LSU0 or LSU1
PM_L2_CASTOUT_MOD	L2 Castouts - Modified (M, Mu, Me)	Total L2 Castouts in Modified (M,Mu,Me) state performed by the L2 castout machine on a per core basis.This event is delivered from the L2 domain, so must be scaled accordingly (divide by 2)
PM_L2_CASTOUT_SHR	L2 Castouts - Shared (T, Te, Si, S)	Total L2 Castouts in Shared (T,Te,Sl,S) performed by the L2 castout machine on a per core basis.This event is delivered from the L2 domain, so must be scaled accordingly (divide by 2)
PM_L2_CO_FAIL_BUSY	L2 RC Cast Out dispatch attempt failed due to all CO machines busy	Cycles when when Castout dispatch failed because of castout machines being busy. This event is delivered from the L2 domain, so must be scaled accordingly (divide by 2)
PM_L2_DC_INV	Dcache invalidates from L2	Total L1 dcache Invalidates sent by L2 over reload bus on a per core basis. The L2 is inclusive of L1 icache and L1 dcache, so it has to invalidate the caches when it rolls over. This event is delivered from the L2 domain, so must be scaled accordingly (divide by 2)
PM_L2_DISP_ALL	All successful LD/ST dispatches for this thread(i+d)	Total number of L2 Read Claim machine dispatches for loads and stores (instruction + data) on a per thread basis. This event is delivered from the L2 domain, so must be scaled accordingly (divide by 2)
PM_L2_GLOB_GUESS_CORRECT	L2 guess glb and guess was correct (ie data remote)	L2 pump prediction mechanism , predicted a global pump and the guess was correct , ie data was remote to chip or node. This event is delivered from the L2 domain, so must be scaled accordingly (divide by 2)

Event Name	Event Description	Detailed Description
PM_L2_GLOB_GUESS_WRONG	L2 guess glb and guess was not correct (ie data local)	L2 pump prediction mechanism , predicted a global pump and the guess was Incorrect , ie data was local to chip or node. This event is delivered from the L2 domain, so must be scaled accordingly (divide by 2)
PM_L2_IC_INV	Icache Invalidates from L2	Total L1 Icache Invalidates sent by L2 over reload bus on a per core basis. The L2 is inclusive of L1 icache and L1 dcache, so it has to invalidate the caches when it rolls over. This event is delivered from the L2 domain, so must be scaled accordingly (divide by 2)
PM_L2_INST	Instruction Load Count	Total number of L2 Read Claim machine dispatches for Instructions on a per thread basis. This event is delivered from the L2 domain, so must be scaled accordingly (divide by 2)
PM_L2_INST_MISS	Instruction Load Misses	Total number of L2 Read Claim machine dispatches for Instructions that were L2 misses on a per thread basis. This event is delivered from the L2 domain, so must be scaled accordingly (divide by 2)
PM_L2_LD	Data Load Count	Total number of L2 Read Claim machine dispatches for loads on a per thread basis. This event is delivered from the L2 domain, so must be scaled accordingly (divide by 2)
PM_L2_LD_DISP	All successful load dispatches	Total number of L2 Read Claim machine dispatches for loads on a per core basis. This event is delivered from the L2 domain, so must be scaled accordingly (divide by 2)
PM_L2_LD_HIT	All successful load dispatches that were L2 hits	Total number of L2 Read Claim machine dispatches for loads that were L2 hits on a per core basis. This event is delivered from the L2 domain, so must be scaled accordingly (divide by 2)
PM_L2_LD_MISS	Data Load Miss	Total number of L2 Read Claim machine dispatches for loads that were L2 misses on a per thread basis. This event is delivered from the L2 domain, so must be scaled accordingly (divide by 2)
PM_L2_LDST	Data Load+Store Count	Total number of L2 Read Claim machine dispatches for loads or stores on a per thread basis. This event is delivered from the L2 domain, so must be scaled accordingly (divide by 2)
PM_L2_LDST_MISS	Data Load+Store Miss	Total number of L2 Read Claim machine dispatches for loads or stores that missed the L2 on a per thread basis. This event is delivered from the L2 domain, so must be scaled accordingly (divide by 2)
PM_L2_LOC_GUESS_CORRECT	L2 guess loc and guess was correct (ie data local)	L2 pump prediction mechanism , predicted a local pump and the guess was correct , ie data was local to chip or node. This event is delivered from the L2 domain, so must be scaled accordingly (divide by 2)
PM_L2_LOC_GUESS_WRONG	L2 guess loc and guess was not correct (ie data remote)	L2 pump prediction mechanism , predicted a local pump and the guess was Incorrect , ie data was remote to chip or node. This event is delivered from the L2 domain, so must be scaled accordingly (divide by 2)
PM_L2_NODE_PUMP	RC req that was a local (aka node) pump attempt	The L2 Read Claim machine performed a local (node /chip) pump attempt. This event is delivered from the L2 domain, so must be scaled accordingly (divide by 2)
PM_L2_RC_ST_DONE	RC did st to line that was Tx or Sx	A Read claim machine working on a store, did a store to a line that was in Sx or Tx state on a per core basis. This event is delivered from the L2 domain, so must be scaled accordingly (divide by 2)
PM_L2_RCLD_BUSY_RC_FULL	L2 activated Busy to the core for loads due to all RC full	Cycles when when all Read Claim machines that can do a load are currently full. This event is delivered from the L2 domain, so must be scaled accordingly (divide by 2)
PM_L2_RCLD_DISP	L2 RC load dispatch attempt	Total Read Claim Load dispatch attempts on a per core basis. This event is delivered from the L2 domain, so must be scaled accordingly (divide by 2)

Event Name	Event Description	Detailed Description
PM_L2_RCLD_DISP_FAIL_ADDR	L2 RC load dispatch attempt failed due to address collision with RC/CO/SN/SQ	Total Read Claim Load dispatch attempts that failed due to an address collision with another machine on a per core basis. This event is delivered from the L2 domain, so must be scaled accordingly (divide by 2)
PM_L2_RCLD_DISP_FAIL_OTHER	L2 RC load dispatch attempt failed due to other reasons	Total Read Claim Load dispatch attempts that failed due to other reasons on a per core basis. This event is delivered from the L2 domain, so must be scaled accordingly (divide by 2)
PM_L2_RCST_BUSY_RC_FULL	L2 activated Busy to the core for stores due to all RC full	Cycles when when all Read Claim machines that can do a store are currently full. This event is delivered from the L2 domain, so must be scaled accordingly (divide by 2)
PM_L2_RCST_DISP	L2 RC store dispatch attempt	Total Read Claim store dispatch attempts on a per core basis. This event is delivered from the L2 domain, so must be scaled accordingly (divide by 2)
PM_L2_RCST_DISP_FAIL_ADDR	L2 RC store dispatch attempt failed due to address collision with RC/CO/SN/SQ	Total Read Claim store dispatch attempts that failed due to an address collision with another machine on a per core basis. This event is delivered from the L2 domain, so must be scaled accordingly (divide by 2)
PM_L2_RCST_DISP_FAIL_OTHER	L2 RC store dispatch attempt failed due to other reasons	Total Read Claim store dispatch attempts that failed due to other reasons on a per core basis. This event is delivered from the L2 domain, so must be scaled accordingly (divide by 2)
PM_L2_SN_M_RD_DONE	SNP dispatched for a read and was M	A snoop machine was dispatched for a read and line was in Modified (M) state. This event is reported in 1:1 clk domain and does not require any scaling.
PM_L2_SN_M_WR_DONE	SNP dispatched for a write and was M	A snoop machine was dispatched for a rwrite and line was in Modified (M) state. This event is reported in 1:1 clk domain and does not require any scaling.
PM_L2_SN_SX_I_DONE	SNP dispatched and went from Sx or Tx to Ix	A snoop caused the transition of a line in L2 from Sx /Tx to Ix , (rwitm,dcl,dcbz,bk,dma_wr,dcbf can cause this). This event is reported in 1:1 clk domain and does not require any scaling.
PM_L2_ST	Data Store Count	Total number of L2 Read Claim machine dispatches for stores on a per thread basis. This event is delivered from the L2 domain, so must be scaled accordingly (divide by 2)
PM_L2_ST_DISP	All successful store dispatches	Total number of L2 Read Claim machine dispatches for stores on a per core basis. This event is delivered from the L2 domain, so must be scaled accordingly (divide by 2)
PM_L2_ST_HIT	All successful store dispatches that were L2Hits	Total number of L2 Read Claim machine dispatches for stores that were L2 hits on a per core basis. This event is delivered from the L2 domain, so must be scaled accordingly (divide by 2)
PM_L2_ST_MISS	Data Store Miss	Total number of L2 Read Claim machine dispatches for stores that were L2 misses on a per thread basis. This event is delivered from the L2 domain, so must be scaled accordingly (divide by 2)
PM_L2_SYS_PUMP	RC req that was a global (aka system) pump attempt	The L2 Read Claim machine performed a global (system) pump attempt. This event is delivered from the L2 domain, so must be scaled accordingly (divide by 2)
PM_L3_CO_L3.1	L3 Castouts to L3.1	Total L3 castouts per core that went to another L3 on the same chip as a part of lateral cast out
PM_L3_CO_MEM	L3 Castouts to Memory	Total L3 castouts per core that went to memory
PM_L3_HIT	L3 Hits	Total L3 reference hits on a per core basis.This event is delivered from the L2 domain, so must be scaled accordingly (divide by 2)
PM_L3_LD_HIT	L3 demand LD Hits	Total L3 Loads that hit the L3 on a per core basis.This event is delivered from the L2 domain, so must be scaled accordingly (divide by 2)
PM_L3_LD_MISS	L3 demand LD Miss	Total L3 Loads that miss the L3 on a per core basis.This event is delivered from the L2 domain, so must be scaled accordingly (divide by 2)

Event Name	Event Description	Detailed Description
PM_L3_MISS	L3 Misses	Total L3 references that miss the L3 on a per core basis.This event is delivered from the L2 domain, so must be scaled accordingly (divide by 2)
PM_L3_PREF_HIT	L3 Prefetch Directory Hit	Total L3 prefetches sent from core that hit L3 prefetch directory
PM_L3_PREF_LD	L3 cache LD prefetches	The LSU hardware prefetch mechanism sent a request to a prefetch a line to the L3 cache for a load
PM_L3_PREF_LDST	L3 cache prefetches LD + ST	Total L3 prefetches sent by the LSU (load + store)
PM_L3_PREF_MISS	L3 Prefetch Directory Miss	Total L3 prefetches sent from core that miss L3 prefetch directory
PM_L3_PREF_ST	L3 cache ST prefetches	The LSU hardware prefetch mechanism sent a request to a prefetch a line to the L3 cache for a store
PM_LARX_LSU	Larx Finished	Larx instruction finished in LSU0 or LSU1 (does not included rejects). The larx will subsequently be dispatched in L2, and the reservation bit for that thread will be set, when the RC for that dispatch goes valid.
PM_LARX_LSU0	ls0 Larx Finished	Larx instruction finished in LSU0 (does not included rejects). The larx will subsequently be dispatched in L2, and the reservation bit for that thread will be set, when the RC for that dispatch goes valid.
PM_LARX_LSU1	ls1 Larx Finished	Larx instruction finished in LSU1 (does not included rejects). The larx will subsequently be dispatched in L2, and the reservation bit for that thread will be set, when the RC for that dispatch goes valid.
PM_LD_MISS_L1	Load Missed L1	Load references that miss the Level 1 Data cache.
PM_LD_REF_L1	L1 D cache load references counted at finish	A Load type instruction was issued on LSU0 or LSU1 , referenced the L1 D cache counted at exec time. If the instruction is rejected this will come on multiple times
PM_LD_REF_L1_LSU0	LS0 L1 D cache load references counted at finish	A Load type instruction was issued on LSU0, referenced the L1 D cache counted at counted at exec time. If the instruction is rejected this will come on multiple times
PM_LD_REF_L1_LSU1	LS1 L1 D cache load references counted at finish	A Load type instruction was issued on LSU1, referenced the L1 D cache counted counted at exec time. If the instruction is rejected this will come on multiple times
PM_LSU_DC_PREF_STREAM_ALLOC	D cache new prefetch stream allocated	A new dcache prefetch stream was allocated.The LSU0 or LSU1 stream could have been allocated through the hardware prefetch mechanism or through software
PM_LSU_DC_PREF_STREAM_CONFIRM	Dcache new prefetch stream confirmed	A demand load issued on LSU0 or LSU1 referenced a line in an active prefetch stream. The stream could have been allocated through the hardware prefetch mechanism or through software. Combine up + down
PM_LSU_DC_PREF_STRIDED_STREAM_CONFIRM	Dcache Strided prefetch stream confirmed (software + hardware)	A demand load issued on LSU0 or LSU1 referenced a line in an active strided prefetch stream. The stream could have been allocated through the hardware prefetch mechanism or through software. Combine up + down
PM_LSU_DCACHE_RELOAD_VALID	count per sector of lines reloaded in L1 (demand + prefetch)	Counts once every time a 32B sector is written into the L1 Dcache (includes demand loads and prefetches).
PM_LSU_DERAT_MISS	DERAT Reloaded due to a DERAT miss	Total D-ERAT Misses. Requests that miss the Derat are rejected and retried until the request hits in the Erat. This may result in multiple erat misses for the same instruction. Combined Unit 0 + 1.
PM_LSU_DERAT_MISS	DERAT Misses	A Load/store instruction suffered a miss in the Effective to real address mapping cache.

Event Name	Event Description	Detailed Description
PM_LSU_FIN	LSU Finished an instruction (up to 2 per cycle)	The Load store unit finished an instruction. Instructions that finish may not necessary complete. As there are two LSU units , there could be 2 finishes per cycle
PM_LSU_FLUSH	Flush initiated by LSU	A flush was initiated by the Load store unit
PM_LSU_FLUSH_LRQ	Flush: LRQ	A load issued on LSU0 or LSU1 was flushed because a younger load executed before an older store executed and they had overlapping data OR two loads executed out of order and they have byte overlap and there was a snoop in between to an overlapped byte.
PM_LSU_FLUSH_SRQ	Flush: SRQ	A store issued on LSU0 or LSU1 was flushed because younger load hits and older store that is already in the SRQ or in the same group.
PM_LSU_FLUSH_ULD	Flush: Unaligned Load	A load issued on LSU1 or LSU0 was flushed because it was unaligned (crossed a 64byte boundary, or 32 byte if it missed the L1).This does not include unligned flushes due to DABR
PM_LSU_FLUSH_UST	Flush: Unaligned Store	A store issued on LSU0 or LSU1 was flushed because it was unaligned (crossed a 4K boundary). Does not include unaligned flushes due to DABR
PM_LSU_FX_FIN	LSU Finished a FX operation (up to 2 per cycle)	The Load store unit finished a fixed point instruction. Instructions that finish may not necessary complete. As there are two LSU units , there could be 2 finishes per cycle
PM_LSU_LDF	All Scalar Loads	A scalar floating point load finished in LSU0 or LSU1
PM_LSU_LDX	All Vector loads (vsx vector + vmx vector)	A vector load finished in LSU0 or LSU1
PM_LSU_LMQ_FULL_CYC	LMQ full	Number of cycles the Load Miss Queue was full (event is shared and not per thread)
PM_LSU_LMQ_S0_ALLOC	Slot 0 of LMQ valid	On a per thread basis this event counts the number of times SLOT0 of the LMQ was valid. By instrumenting a single slot we can calculate average service time for that slot.
PM_LSU_LMQ_S0_VALID	Slot 0 of LMQ valid	On a per thread basis this event counts the number of cycles SLOT0 of the LMQ was valid. By instrumenting a single slot we can calculate average service time for that slot.
PM_LSU_LMQ_SRQ_EMPTY_ALL_CYC	ALL threads lsu empty (lmq and srq empty)	Cycles the Load Miss Queue and Store Reoder Queue were completely empty for all threads.
PM_LSU_LMQ_SRQ_EMPTY_CYC	LSU empty (lmq and srq empty)	Cycles the Load Miss Queue and Store Reoder Queue were completely empty for this threads
PM_LSU_LRQ_S0_ALLOC	Slot 0 of LRQ valid	On a per thread basis this event counts the number of times SLOT0 of the LRQ was valid. By instrumenting a single slot we can calculate average service time for that slot.
PM_LSU_LRQ_S0_VALID	Slot 0 of LRQ valid	On a per thread basis this event counts the number of cycles SLOT0 of the LRQ was valid. By instrumenting a single slot we can calculate average service time for that slot.
PM_LSU_NCLD	Non-cachable Loads counted at finish	A Non cacheable load finished on LSU0 or LSU1 (non cacheable loads are issued as single instruction group and at next to complete time). This signal comes on after all exception/alignment checks are done
PM_LSU_NCST	Non-cachable Stores sent to nest	A non cacheable store was sent to L2 (no record of Issue pipe is maintained so this is for both LSU0 and LSU1)
PM_LSU_PARTIAL_CDF	A partial cacheline was returned from the L3	A partial cacheline was returned from the L3. L3 hit on a line that was target of a transient data touch operation (dcbtp or dcbtstp). Data is forwarded to the requesting load is not writen in the dcache
PM_LSU_REJECT	LSU Reject (up to 2 per cycle)	Total cycles the Load store unit was busy rejecting instructions, as there two LSU units there could be 2 rejects per cycle

Event Name	Event Description	Detailed Description
PM_LSU_REJECT_ERAT_MISS	LSU Reject due to ERAT (up to 2 per cycles)	Total cycles the Load store unit was busy rejecting instructions which suffered an ERAT miss, as there two LSU units there could be 2 rejects per cycle
PM_LSU_REJECT_LHS	Reject: Load Hit Store	LSU0 or LSU1 rejected an instruction because of Load Hit Store condition. Loads are rejected when data is needed from a previous store instruction but store forwarding is not possible because the data is not fully contained in the Store Data Queue or is not yet available in the Store Data Queue.
PM_LSU_REJECT_LMQ_FULL	Reject: LMQ Full (LHR)	Total cycles LSU0 or LSU1 is busy rejecting instructions because the Load Miss Queue was full. The LMQ has eight entries. If all the eight entries are full, subsequent load instructions are rejected. This also includes Load hit reaload reject, that is, an individual LMQ entry was full and a load that wanted to merge onto an existing entry was not able to.
PM_LSU_REJECT_SET_MPRED	Reject: Set Predict Wrong	cycles LSU was rejecting after detecting setp mispredict on LSU0 or LSU1
PM_LSU_SET_MPRED	Line already in cache at reload time	LSU detected setp mispredict at reload time because before the hash was written a hash read was done to read out all 8 sets. If a match is found the line is invalidated. This event will also include the alias case since the LSU doesn't have any way to separate the two (mispredicts and alias cases). Assuming aliases don't occur that often, this signal should be fairly accurate. Note: This event is not split per pipe, but it could happen on both pipes on the same cycle. The only way that it would happen on both pipes on the same cycle is if both recycled LSU0 and LSU1 critical data forward were using the same sector of the same l2 request. This situation really only reflects one of the 2 ops having a setp mispredict and fetching the L2 (since the other one merged).
PM_LSU_SRQ_EMPTY_CYC	ALL threads srq empty	Cycles the SRQ (store re order queue) was empty for all threads
PM_LSU_SRQ_FULL_CYC	Storage Queue is full and is blocking dispatch	Cycles where the SRQ has run out of virtual SRQ enties, and dispatch is blocked
PM_LSU_SRQ_S0_ALLOC	Slot 0 of SRQ valid	On a per thread basis this event counts the number of times SLOT0 of the SRQ was valid. By instrumenting a single slot we can calculate average service time for that slot.
PM_LSU_SRQ_S0_VALID	Slot 0 of SRQ valid	On a per thread basis this event counts the number of cycles SLOT0 of the SRQ was valid. By instrumenting a single slot we can calculate average service time for that slot.
PM_LSU_SRQ_STFWD	Load got data from a store	A load issued on LSU0 or LSU1 had an address match with an older store in SRQ , and data was forwarded to load from SDQ.
PM_LSU_SRQ_SYNC_COUNT	SRQ sync count (edge of PM_LSU_SRQ_SYNC_CYC)	This event counts the number of heavyweight syncs issued to the LSU. The event SRQ_SYNC_CYC divided by this count gives the average lifetime of a heavyweight sync.
PM_LSU_SRQ_SYNC_CYC	A sync is in the SRQ	number of cycles a heavyweight sync is in the SRQ. After the hwsync issues, it sits in srq until it becomes next to complete. when it becomes next to complete it is eligible to send out to L2. When the sync becomes the oldest store in the srq, and the L2 stq and the NCU stq have an entry available, the sync is sent to both the L2 and the NCU. when the sync has cleared, the nest responds with a sync_ack. upon receiving the sync_ack, the hwsync is finished and then the ISU completes the group with the hwsync and the LSU deallocates the heavyweight sync from the SRQ.

Event Name	Event Description	Detailed Description
PM_LSU_TWO_TABLEWALK_CYC	Cycles when two tablewalks pending on this thread	Cycles when two tablewalks were pending on this thread. Indicates that two tablewalks are pending simultaneously for the respective thread. In order for this event to come on in a 2 thread test requires that instructions from the respective thread are being issued to both pipes, LS0 and LS1. Alternatively, it would also be possible for this event to come on when an I-side tablewalk is active in addition to 1 LSU tablewalk.For SMT4 mode, the only way to observe two tablewalks pending for a particular thread is to to have both iside and dside pending for that thread
PM_LSU0_DC_PREF_STREAM_ALLOC	LS0 D cache new prefetch stream allocated	A new dcache prefetch stream was allocated.The LSU0 stream could have been allocated through the hardware prefetch mechanism or through software
PM_LSU0_DC_PREF_STREAM_CONFIRM	LS0 Dcache prefetch stream confirmed	A demand load issued on LSU0 referenced a line in an active prefetch stream. The stream could have been allocated through the hardware prefetch mechanism or through software. Combine up + down
PM_LSU0_DC_PREF_STREAM_CONFIRM_STRIDE	LS0 Dcache Strided prefetch stream confirmed	A demand load issued on LSU0 referenced a line in an active strided prefetch stream. The stream could have been allocated through the hardware prefetch mechanism or through software. Combine up + down
PM_LSU0_FLUSH_LRQ	LS0 Flush: LRQ	A load issued on LSU0 was flushed because a younger load executed before an older store executed and they had overlapping data OR two loads executed out of order and they have byte overlap and there was a snoop in between to an overlapped byte.
PM_LSU0_FLUSH_SRQ	LS0 Flush: SRQ	A store issued on LSU0 was flushed because younger load hits and older store that is already in the SRQ or in the same group.
PM_LSU0_FLUSH_ULD	LS0 Flush: Unaligned Load	A load issued on LSU0 was flushed because it was unaligned (crossed a 64byte boundary, or 32 byte if it missed the L1).This does not include unligned flushes due to DABR
PM_LSU0_FLUSH_UST	LS0 Flush: Unaligned Store	A store issued on LSU0 was flushed because it was unaligned (crossed a 4K boundary). Does not include unaligned flushes due to DABR
PM_LSU0_L1_PREF	LS0 L1 cache data prefetches	A request to prefetch a line into the L1 was made from LSU0
PM_LSU0_L1_SW_PREF	LSU0 Software L1 Prefetches, including SW Transient Prefetches	A software initiated L1 prefetch (includes transient prefetches) finished in LSU0.
PM_LSU0_LDF	LS0 Scalar Loads	A scalar floating point load finished in LSU0
PM_LSU0_LDX	LS0 Vector Loads	A vector load finished in LSU0
PM_LSU0_LMQ_LHR_MERGE	LS0 Load Merged with another cacheline request	A data cache miss occured for a load issued on LSU0 to the same real cache line address as an earlier request already in the Load Miss Queue and was merged into the LMQ entry.
PM_LSU0_NCLD	LS0 Non-cachable Loads counted at finish	A Non cacheable load finished on LSU0 (non cacheable loads are issued as single instruction group and at next to complete time). This signal comes on after all exception/alignment checks are done
PM_LSU0_REJECT_LHS	LS0 Reject: Load Hit Store	LSU0 rejected an instruction because of Load Hit Store condition. Loads are rejected when data is needed from a previous store instruction but store forwarding is not possible because the data is not fully contained in the Store Data Queue or is not yet available in the Store Data Queue.
PM_LSU0_REJECT_LMQ_FULL	LS0 Reject: LMQ Full (LHR)	Total cycles LSU0 is busy rejecting instructions because the Load Miss Queue was full. If all the LMQ entries are full, subsequent load instructions are rejected. This also includes Load hit reaload reject, that is, an individual LMQ entry was full and a load that wanted to merge onto an existing entry was not able to.

Event Name	Event Description	Detailed Description
PM_LSU0_SRQ_STFWD	LS0 SRQ forwarded data to a load	A load issued on LSU0 had an address match with an older store in SRQ , and data was forwarded to load from SDQ.
PM_LSU1_DC_PREF_STREAM_ALLOC	LS 1 D cache new prefetch stream allocated	A new dcache prefetch stream was allocated.The LSU1 stream could have been allocated through the hardware prefetch mechanism or through software
PM_LSU1_DC_PREF_STREAM_CONFIRM	LS1 'Dcache prefetch stream confirmed	A demand load issued on LSU1 referenced a line in an active prefetch stream. The stream could have been allocated through the hardware prefetch mechanism or through software. Combine up + down
PM_LSU1_DC_PREF_STREAM_CONFIRM_STRIDE	LS1 Dcache Strided prefetch stream confirmed	A demand load issued on LSU1 referenced a line in an active strided prefetch stream. The stream could have been allocated through the hardware prefetch mechanism or through software. Combine up + down
PM_LSU1_FLUSH_LRQ	LS1 Flush: LRQ	A load issued on LSU1 was flushed because a younger load executed before an older store executed and they had overlapping data OR two loads executed out of order and they have byte overlap and there was a snoop in between to an overlapped byte.
PM_LSU1_FLUSH_SRQ	LS1 Flush: SRQ	A store issued on LSU1 was flushed because younger load hits and older store that is already in the SRQ or in the same group.
PM_LSU1_FLUSH_ULD	LS 1 Flush: Unaligned Load	A load issued on LSU1 was flushed because it was unaligned (crossed a 64byte boundary, or 32 byte if it missed the L1).This does not include unligned flushes due to DABR
PM_LSU1_FLUSH_UST	LS1 Flush: Unaligned Store	A store issued on LSU1 was flushed because it was unaligned (crossed a 4K boundary). Does not include unaligned flushes due to DABR
PM_LSU1_L1_PREF	LS1 L1 cache data prefetches	A request to prefetch a line into the L1 was made from LSU1
PM_LSU1_L1_SW_PREF	LSU1 Software L1 Prefetches, including SW Transient Prefetches	A software initiated L1 prefetch (includes transient prefetches) finished in LSU1.
PM_LSU1_LDF	LS1 Scalar Loads	A scalar floating point load finished in LSU1
PM_LSU1_LDX	LS1 Vector Loads	A vector load finished in LSU1
PM_LSU1_LMQ_LHR_MERGE	LS1 Load Merge with another cacheline request	A data cache miss occured for a load issued on LSU1 to the same real cache line address as an earlier request already in the Load Miss Queue and was merged into the LMQ entry.
PM_LSU1_NCLD	LS1 Non-cachable Loads counted at finish	A Non cacheable load finished on LSU1(non cacheable loads are issued as single instruction group and at next to complete time). This signal comes on after all exception/alignment checks are done
PM_LSU1_REJECT_LHS	LS1 Reject: Load Hit Store	LSU0 rejected an instruction because of Load Hit Store condition. Loads are rejected when data is needed from a previous store instruction but store forwarding is not possible because the data is not fully contained in the Store Data Queue or is not yet available in the Store Data Queue.
PM_LSU1_REJECT_LMQ_FULL	LS1 Reject: LMQ Full (LHR)	Total cycles LSU1 is busy rejecting instructions because the Load Miss Queue was full. The LMQ has eight entries. If all the eight entries are full, subsequent load instructions are rejected. This also includes Load hit reaload reject, that is, an individual LMQ entry was full and a load that wanted to merge onto an existing entry was not able to.
PM_LSU1_SRQ_STFWD	LS1 SRQ forwarded data to a load	A load issued on LSU1 had an address match with an older store in SRQ , and data was forwarded to load from SDQ.
PM_LWSYNC	lwsync count (easier to use than IMC)	A lwsync issued to the LSU was sent to the L2. A lwsync doesn't require a sync_ack back from the L2, so the store can be completed in the core and SRQ entry can be released like other stores.

Event Name	Event Description	Detailed Description
PM_LWSYNC_HELD	LWSYNC held at dispatch	Cycles a LWSYNC instruction was held at dispatch. LWSYNC instructions are held at dispatch until all previous loads are done and all previous stores have issued. LWSYNC enters the Store Request Queue and is sent to the storage subsystem but does not wait for a response.
PM_MEM0_PREFETCH_DISP	MC0 prefetch read dispatched	A memory read for a prefetch was dispatched in Memory controller 0 . This event also needs to prescaled (multiply by 8) appropriately because of asynchronous clock domain crossing.
PM_MEM0_RD_CANCEL_TOTAL	MC0 speculative reads cancelled	Total memory reads that were cancelled due to speculation in memory controller 0. The memory controller can dispatch a read speculatively to have data before combined response, but sometimes has to cancel it because some other cache had the data. This event also needs to prescaled (multiply by 8) appropriately because of asynchronous clock domain crossing.
PM_MEM0_RQ_DISP	MC0 read dispatched	A memory read was dispatched in Memory controller 0 . This event needs to prescaled (multiply by 8) appropriately because of asynchronous clock domain crossing.
PM_MEM0_WQ_DISP	MC0 writes dispatched	A memory write was dispatched in Memory controller 0 . This event needs to prescaled (multiply by 8) appropriately because of asynchronous clock domain crossing.
PM_MRK_BR_MPRED	Marked Branch Mispredicted	A marked (sampled) instruction suffered a branch mispredict.
PM_MRK_BR_TAKEN	Marked Branch Taken	A marked (sampled) instruction was a taken branch.
PM_MRK_BRU_FIN	bru marked instr finish	The Branch unit unit finished a marked instruction that was a branch. Instructions that finish may not necessary complete
PM_MRK_DATA_FROM_DL2L3_MOD	Marked data loaded from distant L2 or L3 modified	The processor's Data Cache was reloaded with Modified (M) data from a chip's L2 or L3 on a different Node (Distant) due to a marked demand load .
PM_MRK_DATA_FROM_DL2L3_MOD	Marked data loaded from distant L2 or L3 modified	The processor's Data Cache was reloaded with Modified (M) data from a chip's L2 or L3 on a different Node (Distant) due to a marked demand load .
PM_MRK_DATA_FROM_DL2L3_MOD_CYC	Marked ld latency Data source 1011 (L2.75/L3.75 M different 4 chip node)	Cycles a marked load waited for Modified(M) data from a chip's memory on a different Node (Distant) . Counting begins when a marked load misses the data cache and ends when the data is reloaded into the data cache. To calculate average latency divide this count by the number of marked misses to the same level.
PM_MRK_DATA_FROM_DL2L3_SHR	Marked data loaded from distant L2 or L3 shared	The processor's Data Cache was reloaded with Shared (S) data from a chip's L2 or L3 on a different Node (Distant) due to a marked demand load .
PM_MRK_DATA_FROM_DL2L3_SHR_CYC	Marked ld latency Data Source 1010 (Distant L2.75/L3.75 S)	Cycles a marked load waited for Shared(S) data from a chip's L2 or L3 on a different Node (Distant) . Counting begins when a marked load misses the data cache and ends when the data is reloaded into the data cache. To calculate average latency divide this count by the number of marked misses to the same level.
PM_MRK_DATA_FROM_DMEM	Marked data loaded from distant memory	The processor's Data Cache was reloaded with data from a chip's memory on a different Node (Distant) due to a marked demand load .
PM_MRK_DATA_FROM_DMEM	Marked data loaded from distant memory	The processor's Data Cache was reloaded with data from a chip's memory on a different Node (Distant) due to a marked demand load .
PM_MRK_DATA_FROM_DMEM_CYC	Marked ld latency Data Source 1110 (Distant Memory)	Cycles a marked load waited for Shared(S) data from a chip's memory on a different Node (Distant) . Counting begins when a marked load misses the data cache and ends when the data is reloaded into the data cache. To calculate average latency divide this count by the number of marked misses to the same level.

Event Name	Event Description	Detailed Description
PM_MRK_DATA_FROM_L2	Marked data loaded from L2	The processor's Data Cache was reloaded with data from the local chiplet's L2 cache due to a marked demand load.
PM_MRK_DATA_FROM_L2_CYC	Marked ld latency Data source 0000 (L2 hit)	Cycles a marked load waited for data from a local chiplet's L2 . Counting begins when a marked load misses the data cache and ends when the data is reloaded into the data cache. To calculate average latency divide this count by the number of marked misses to the same level.
PM_MRK_DATA_FROM_L21_MOD	Marked data loaded from another L2 on same chip modified	The processor's Data Cache was reloaded with Modified (M) data from another L2 on the same chip due to a marked demand load .
PM_MRK_DATA_FROM_L21_MOD_CYC	Marked ld latency Data source 0101 (L2.1 M same chip)	Cycles a marked load waited for Modified(M) data from from another L2 on the same chip . Counting begins when a marked load misses the data cache and ends when the data is reloaded into the data cache. To calculate average latency divide this count by the number of marked misses to the same level.
PM_MRK_DATA_FROM_L21_SHR	Marked data loaded from another L2 on same chip shared	The processor's Data Cache was reloaded with Shared (S) data from another L2 on the same chip due to a marked demand load .
PM_MRK_DATA_FROM_L21_SHR	Marked data loaded from another L2 on same chip shared	The processor's Data Cache was reloaded with Shared (S) data from another L2 on the same chip due to a marked demand load .
PM_MRK_DATA_FROM_L21_SHR_CYC	Marked ld latency Data source 0100 (L2.1 S)	Cycles a marked load waited for Shared(S) data from from another L2 on the same chip . Counting begins when a marked load misses the data cache and ends when the data is reloaded into the data cache. To calculate average latency divide this count by the number of marked misses to the same level.
PM_MRK_DATA_FROM_L2MISS	Marked data loaded missed L2	The processor's Data Cache was reloaded but not from the local L2 due to a marked demand load.
PM_MRK_DATA_FROM_L3	Marked data loaded from L3	The processor's Data Cache was reloaded with data from the local chiplet's L3 cache due to a marked demand load.
PM_MRK_DATA_FROM_L3	Marked data loaded from L3	The processor's Data Cache was reloaded with data from the local chiplet's L3 cache due to a marked demand load.
PM_MRK_DATA_FROM_L3_CYC	Marked ld latency Data source 0001 (L3)	Cycles a marked load waited for data from a local chiplet's L3 . Counting begins when a marked load misses the data cache and ends when the data is reloaded into the data cache. To calculate average latency divide this count by the number of marked misses to the same level.
PM_MRK_DATA_FROM_L31_MOD	Marked data loaded from another L3 on same chip modified	The processor's Data Cache was reloaded with Modified (M) data from another L3 on the same chip due to a marked demand load .
PM_MRK_DATA_FROM_L31_MOD_CYC	Marked ld latency Data source 0111 (L3.1 M same chip)	Cycles a marked load waited for Modified(M) data from from another L3 on the same chip . Counting begins when a marked load misses the data cache and ends when the data is reloaded into the data cache. To calculate average latency divide this count by the number of marked misses to the same level.
PM_MRK_DATA_FROM_L31_SHR	Marked data loaded from another L3 on same chip shared	The processor's Data Cache was reloaded with Shared (S) data from another L3 on the same chip due to a marked demand load .
PM_MRK_DATA_FROM_L31_SHR	Marked data loaded from another L3 on same chip shared	The processor's Data Cache was reloaded with Shared (S) data from another L3 on the same chip due to a marked demand load .
PM_MRK_DATA_FROM_L31_SHR_CYC	Marked ld latency Data source 0110 (L3.1 S)	Cycles a marked load waited for Shared(S) data from from another L3 on the same chip . Counting begins when a marked load misses the data cache and ends when the data is reloaded into the data cache. To calculate average latency divide this count by the number of marked misses to the same level.
PM_MRK_DATA_FROM_L3MISS	Marked data loaded from L3 miss	The processor's Data Cache was reloaded but not from the local L3 due to a marked demand load.

Event Name	Event Description	Detailed Description
PM_MRK_DATA_FROM_LMEM	Marked data loaded from local memory	The processor's Data Cache was reloaded with data from the local chip's memory due to a marked demand load.
PM_MRK_DATA_FROM_LMEM	Marked data loaded from local memory	The processor's Data Cache was reloaded with data from the local chip's memory due to a marked demand load.
PM_MRK_DATA_FROM_LMEM_CYC	Marked ld latency Data Source 1100 (Local Memory)	Cycles a marked load waited for data from a local chip's memory . Counting begins when a marked load misses the data cache and ends when the data is reloaded into the data cache. To calculate average latency divide this count by the number of marked misses to the same level.
PM_MRK_DATA_FROM_RL2L3_MOD	Marked data loaded from remote L2 or L3 modified	The processor's Data Cache was reloaded with Modified (M) data from a chip's L2 or L3 on the same Node (Remote) due to a marked demand load .
PM_MRK_DATA_FROM_RL2L3_MOD_CYC	Marked ld latency Data source 1001 (L2.5/L3.5 M same 4 chip node)	Cycles a marked load waited for Modified(M) data from a chip's memory on same Node . Counting begins when a marked load misses the data cache and ends when the data is reloaded into the data cache. To calculate average latency divide this count by the number of marked misses to the same level.
PM_MRK_DATA_FROM_RL2L3_SHR	Marked data loaded from remote L2 or L3 shared	The processor's Data Cache was reloaded with Shared (S) data from a chip's L2 or L3 on the same Node (Remote) due to a marked demand load .
PM_MRK_DATA_FROM_RL2L3_SHR	Marked data loaded from remote L2 or L3 shared	The processor's Data Cache was reloaded with Shared (S) data from a chip's L2 or L3 on the same Node (Remote) due to a marked demand load .
PM_MRK_DATA_FROM_RL2L3_SHR_CYC	Marked ld latency Data Source 1000 (Remote L2.5/L3.5 S)	Cycles a marked load waited for Shared(S) data from a chip's memory on same Node . Counting begins when a marked load misses the data cache and ends when the data is reloaded into the data cache. To calculate average latency divide this count by the number of marked misses to the same level.
PM_MRK_DATA_FROM_RMEM	Marked data loaded from remote memory	The processor's Data Cache was reloaded with data from another chip's memory on the same node due to a marked demand load.
PM_MRK_DATA_FROM_RMEM_CYC	Marked ld latency Data source 1101 (Memory same 4 chip node)	Cycles a marked load waited for data from another chip's memory on same node . Counting begins when a marked load misses the data cache and ends when the data is reloaded into the data cache. To calculate average latency divide this count by the number of marked misses to the same level.
PM_MRK_DERAT_MISS_16G	Marked DERAT misses for 16G page	The source page size of a DERAT reload for a marked instruction is 16G
PM_MRK_DERAT_MISS_16M	Marked DERAT misses for 16M page	The source page size of a DERAT reload for a marked instruction is 16M
PM_MRK_DERAT_MISS_4K	Marked DERAT misses for 4K page	The source page size of a DERAT reload for a marked instruction is 4K
PM_MRK_DERAT_MISS_64K	Marked DERAT misses for 64K page	The source page size of a DERAT reload for a marked instruction is 64K
PM_MRK_DFU_FIN	Decimal Unit marked Instruction Finish	The decimal unit finished a marked instruction. Instructions that finish may not necessary complete
PM_MRK_DTLB_MISS_16G	Marked Data TLB misses for 16G page	Data TLB references to 16GB pages by a marked instruction that missed the TLB. Page size is determined at TLB reload time.
PM_MRK_DTLB_MISS_16M	Marked Data TLB misses for 16M page	Data TLB references to 16MB pages by a marked instruction that missed the TLB. Page size is determined at TLB reload time.
PM_MRK_DTLB_MISS_4K	Marked Data TLB misses for 4K page	Data TLB references to 4K pages by a marked instruction that missed the TLB. Page size is determined at TLB reload time.
PM_MRK_DTLB_MISS_4K	Marked Data TLB misses for 4K page	Data TLB references to 4K pages by a marked instruction that missed the TLB. Page size is determined at TLB reload time.

Event Name	Event Description	Detailed Description
PM_MRK_DTLB_MISS_64K	Marked Data TLB misses for 64K page	Data TLB references to 64K pages by a marked instruction that missed the TLB. Page size is determined at TLB reload time.
PM_MRK_FIN_STALL_CYC	Marked instruction Finish Stall cycles (marked finish after NTC)	Cycles a marked instruction was active (not finished) while the marked group was next to complete.
PM_MRK_FIN_STALL_CYC_COUNT	Marked instruction Finish Stall cycles (marked finish after NTC) (use edge detect to count #)	Number of times a marked instruction active (not finished) while the marked group was next to complete.
PM_MRK_FXU_FIN	fxu marked instr finish	The fixed point unit finished a marked instruction. Instructions that finish may not necessary complete
PM_MRK_GRP_CMPL	Marked group complete	A group containing a sampled instruction completed. Microcoded instructions that span multiple groups will generate this event once per group.
PM_MRK_GRP_IC_MISS	Marked group experienced I cache miss	A group containing a sampled instruction experienced an icache miss. This event can be used to accurately profile on icache misses.
PM_MRK_IFU_FIN	IFU non-branch marked instruction finished	The Instruction fetch unit unit finished a non-branch marked instruction. Instructions that finish may not necessary complete
PM_MRK_INST_DISP	marked instruction dispatch	A group containing a sampled instruction was disaptched .
PM_MRK_INST_FIN	marked instr finish any unit	One of the execution units finished a marked instruction. Instructions that finish may not necessary complete
PM_MRK_INST_ISSUED	Marked instruction issued	A sampled instruction was issued to any one of the execution units.
PM_MRK_INST_TIMEO	marked Instruction finish timeout	A sampled instruction was lost from tracking mechanism, either due to a flush or other asynchronous events.
PM_MRK_LD_MISS_EXPOSED_CYC	Marked Load exposed Miss	Cycles a marked demand load had data cache miss and was waiting on a reload, and the marked group was next to complete.
PM_MRK_LD_MISS_EXPOSED_CYC_COUNT	Marked Load exposed Miss (use edge detect to count #)	Number of times a marked demand load had data cache miss and was waiting on a reload, and the marked group was next to complete.
PM_MRK_LD_MISS_L1	Marked DL1 Demand Miss	Marked Load references that missed the Level 1 Data cache
PM_MRK_LD_MISS_L1_CYC	L1 data load miss cycles	Cycles a marked load that suffered an dcache miss was waiting on a reload from the memory subsystem.
PM_MRK_LSU_DERAT_MISS	Marked DERAT Miss	A marked (sampled) instruction suffered a DERAT miss
PM_MRK_LSU_FIN	Marked LSU instruction finished	The Load store unit finished a marked instruction. Instructions that finish may not necessary complete
PM_MRK_LSU_FLUSH	Flush: (marked) : All Cases	A marked instruction issued on LSU0/LSU1 was flushed for any reason (includes unaligned/SRQ/LRQ and other reasons for which we don't have specific events)
PM_MRK_LSU_FLUSH_LRQ	Flush: (marked) LRQ	A marked load issued on LSU0/LSU1 was flushed because a younger load executed before an older store executed and they had overlapping data OR two loads executed out of order and they have byte overlap and there was a snoop in between to an overlapped byte.
PM_MRK_LSU_FLUSH_SRQ	Flush: (marked) SRQ	A marked store issued on LSU0/LSU1 was flushed because younger load hits and older store that is already in the SRQ or in the same group.
PM_MRK_LSU_FLUSH_ULD	Flush: (marked) Unaligned Load	A marked load issued on LSU0 or LSU1 was flushed because it was unaligned (crossed a 64byte boundary, or 32 byte if it missed the L1).This does not include unligned flushes due to DABR

Event Name	Event Description	Detailed Description
PM_MRK_LSU_FLUSH_UST	Flush: (marked) Unaligned Store	A marked store issued on LSU0 or LSU1 was flushed because it was unaligned (crossed a 4K boundary). Does not include unaligned flushes due to DABR
PM_MRK_LSU_PARTIAL_CDF	A partial cacheline was returned from the L3 for a marked load	A partial cacheline was returned from the L3. L3 hit on a line that was target of a transient data touch operation (dcbtp or dcbtstp). Data is forwarded to the requesting load is not writen in the cache. For a marked load."
PM_MRK_LSU_REJECT	LSU marked reject	A sampled (marked) instruction suffered a LSU reject.
PM_MRK_LSU_REJECT_ERAT_MISS	LSU marked reject due to ERAT	A Marked (sampled) instruction was rejected in the Load store unit due to an ERAT miss .
PM_MRK_LSU_REJECT_LHS	Reject(marked): Load Hit Store	A marked instruction issued on LSU0/LSU1 was rejected because of Load Hit Store condition. Loads are rejected when data is needed from a previous store instruction but store forwarding is not possible because the data is not fully contained in the Store Data Queue or is not yet available in the Store Data Queue.
PM_MRK_PTEG_FROM_DL2L3_MOD	Marked PTEG loaded from distant L2 or L3 modified	A Page Table Entry was loaded into the TLB with modified (M) data from a chip's L2 or L3 on a different Node (Distant) due to a data side request from a marked instruction
PM_MRK_PTEG_FROM_DL2L3_SHR	Marked PTEG loaded from remote L2 or L3 shared	A Page Table Entry was loaded into the TLB with Shared (S) data from a chip's L2 or L3 on a different Node (Distant) due to a data side request from a marked instruction
PM_MRK_PTEG_FROM_DMEM	Marked PTEG loaded from distant memory	A Page Table Entry was loaded into the TLB with data from a chip's memory on a different Node due to a data side request from a marked instruction
PM_MRK_PTEG_FROM_L2	Marked PTEG loaded from L2	A Page Table Entry was loaded into the TLB with data from local chiplet's L2 due to a data side request from a marked instruction
PM_MRK_PTEG_FROM_L21_MOD	Marked PTEG loaded from another L2 on same chip modified	A Page Table Entry was loaded into the TLB with Modified (M) data from another L2 on the same chip due to a data side request from a marked instruction
PM_MRK_PTEG_FROM_L21_SHR	Marked PTEG loaded from another L2 on same chip shared	A Page Table Entry was loaded into the TLB with Shared (S) data from another L2 on the same chip due to a data side request from a marked instruction
PM_MRK_PTEG_FROM_L2MISS	Marked PTEG loaded from L2 miss	A Page Table Entry was loaded into the TLB with data not from local chiplet's L2 due to a data side request from a marked instruction
PM_MRK_PTEG_FROM_L3	Marked PTEG loaded from L3	A Page Table Entry was loaded into the TLB with data from local chiplet's L3 due to a data side request from a marked instruction
PM_MRK_PTEG_FROM_L31_MOD	Marked PTEG loaded from another L3 on same chip modified	A Page Table Entry was loaded into the TLB with Modified (M) data from another L3 on the same chip due to a data side request from a marked instruction
PM_MRK_PTEG_FROM_L31_SHR	Marked PTEG loaded from another L3 on same chip shared	A Page Table Entry was loaded into the TLB with Shared (S) data from another L3 on the same chip due to a data side request from a marked instruction
PM_MRK_PTEG_FROM_L3MISS	Marked PTEG loaded from L3 miss	A Page Table Entry was loaded into the TLB with data not from local chiplet's L3 due to a data side request from a marked instruction
PM_MRK_PTEG_FROM_LMEM	Marked PTEG loaded from local memory	A Page Table Entry was loaded into the TLB with data from local chip's memory due to a data side request from a marked instruction
PM_MRK_PTEG_FROM_RL2L3_MOD	Marked PTEG loaded from remote L2 or L3 modified	A Page Table Entry was loaded into the TLB with modified (M) data from a chip's L2 or L3 on same Node (Remote) due to a data side request from a marked instruction

Event Name	Event Description	Detailed Description
PM_MRK_PTEG_FROM_RL2L3_SHR	Marked PTEG loaded from remote L2 or L3 shared	A Page Table Entry was loaded into the TLB with Shared (S) data from a chip's L2 or L3 on same Node (Remote) due to a data side request from a marked instruction
PM_MRK_PTEG_FROM_RMEM	Marked PTEG loaded from remote memory	A Page Table Entry was loaded into the TLB with data from another chip's memory on same node due to a data side request from a marked instruction
PM_MRK_ST_CMPL	marked store finished (was complete)	A sampled store previously sent to the L2 has completed (data home).
PM_MRK_ST_CMPL_INT	marked store complete (data home) with intervention	A marked store previously sent to the L2 completed (data home) after requiring intervention.
PM_MRK_ST_NEST	marked store sent to Nest	A marked store instruction was sent to the L2.
PM_MRK_STALL_CMPLU_CYC	Marked Group Completion Stall cycles	Cycles a marked instruction was next to complete and has not yet completed
PM_MRK_STALL_CMPLU_CYC_COUNT	Marked Group Completion Stall cycles (use edge detect to count #)	Count of when marked instructions was next to complete and has not yet completed
PM_MRK_STCX_FAIL	Marked STCX failed	A marked stcx issued on LSU0 or LSU1 got a fail response from L2. For a stcx, if reservation is not set or the stcx address does not match the reservation address at L2 RC dispatch, then the L2 sends a fail indication to the core)
PM_MRK_VSU_FIN	vsu (fpu) marked instr finish	The Vector scalar unit finished a marked instruction. Instructions that finish may not necessary complete
PM_PB_NODE_PUMP	Power Bus node pump	Locally mastered node pumps for this chip, this event is shared across all threads on a chip. This event also needs to prescaled (multiply by 8) approprately because of asynchronous clock domain crossing. This event only looks at snoop port0
PM_PB_RETRY_NODE_PUMP	Power bus node pump retries	Locally mastered node pump that were retried for this chip, this event is shared across all threads on a chip. This event also needs to prescaled (multiply by 8) approprately because of asynchronous clock domain crossing. This event only looks at snoop port0
PM_PB_RETRY_SYS_PUMP	Power bus system pump retries	Locally mastered system pumps that were retried for this chip, this event is shared across all threads on a chip. This event also needs to prescaled (multiply by 8) approprately because of asynchronous clock domain crossing. This event only looks at snoop port0
PM_PB_SYS_PUMP	Power bus system pumps	Locally mastered system pumps for this chip, this event is shared across all threads on a chip. This event also needs to prescaled (multiply by 8) approprately because of asynchronous clock domain crossing. This event only looks at snoop port0
PM_PMC1_OVERFLOW	Overflow from counter 1	Overflows from PMC1 are counted. This effectively widens the PMC. The Overflow from the original PMC will not trigger an exception even if the PMU is configured to generate exceptions on overflow.
PM_PMC2_OVERFLOW	Overflow from counter 2	Overflows from PMC2 are counted. This effectively widens the PMC. The Overflow from the original PMC will not trigger an exception even if the PMU is configured to generate exceptions on overflow.
PM_PMC2_REWIND	PMC2 Rewind Event (did not match condition)	PMC2 was counting speculatively and the confition it was configured to match does not
PM_PMC2_SAVED	PMC2 Rewind Value saved	PMC2 was counting speculatively and the confition it was configured to match does.
PM_PMC3_OVERFLOW	Overflow from counter 3	Overflows from PMC3 are counted. This effectively widens the PMC. The Overflow from the original PMC will not trigger an exception even if the PMU is configured to generate exceptions on overflow.
PM_PMC4_OVERFLOW	Overflow from counter 4	Overflows from PMC4 are counted. This effectively widens the PMC. The Overflow from the original PMC will not trigger an exception even if the PMU is configured to generate exceptions on overflow.

Event Name	Event Description	Detailed Description
PM_PMC4_REWIND	PMC4 Rewind Event	PMC4 was counting speculatively and the confition it was configured to match does not
PM_PMC4_SAVED	PMC4 Rewind Value saved (matched condition)	PMC4 was counting speculatively and the confition it was configured to match does.
PM_PMC5_OVERFLOW	Overflow from counter 5	Overflows from PMC5 are counted. This effectively widens the PMC. The Overflow from the original PMC will not trigger an exception even if the PMU is configured to generate exceptions on overflow.
PM_PMC6_OVERFLOW	Overflow from counter 6	Overflows from PMC6 are counted. This effectively widens the PMC. The Overflow from the original PMC will not trigger an exception even if the PMU is configured to generate exceptions on overflow.
PM_PTEG_FROM_DL2L3_MOD	PTEG loaded from distant L2 or L3 modified	A Page Table Entry was loaded into the TLB with modified (M) data from a chip's L2 or L3 on a different Node (Distant) due to a data side request.
PM_PTEG_FROM_DL2L3_SHR	PTEG loaded from remote L2 or L3 shared	A Page Table Entry was loaded into the TLB with Shared (S) data from a chip's L2 or L3 on a different Node (Distant) due to a data side request.
PM_PTEG_FROM_DMEM	PTEG loaded from distant memory	A Page Table Entry was loaded into the TLB with data from a chip's memory on a different Node due to a data side request.
PM_PTEG_FROM_L2	PTEG loaded from L2	A Page Table Entry was loaded into the TLB with data from local chiplet's L2 due to a data side request.
PM_PTEG_FROM_L21_MOD	PTEG loaded from another L2 on same chip modified	A Page Table Entry was loaded into the TLB with Modified (M) data from another L2 on the same chip due to a data side request.
PM_PTEG_FROM_L21_SHR	PTEG loaded from another L2 on same chip shared	A Page Table Entry was loaded into the TLB with Shared (S) data from another L2 on the same chip due to a data side request.
PM_PTEG_FROM_L2MISS	PTEG loaded from L2 miss	A Page Table Entry was loaded into the TLB with data not from local chiplet's L2 due to a data side request.
PM_PTEG_FROM_L3	PTEG loaded from L3	A Page Table Entry was loaded into the TLB with data from local chiplet's L3 due to a data side request.
PM_PTEG_FROM_L31_MOD	PTEG loaded from another L3 on same chip modified	A Page Table Entry was loaded into the TLB with Modified (M) data from another L3 on the same chip due to a data side request.
PM_PTEG_FROM_L31_SHR	PTEG loaded from another L3 on same chip shared	A Page Table Entry was loaded into the TLB with Shared (S) data from another L3 on the same chip due to a data side request.
PM_PTEG_FROM_L3MISS	PTEG loaded from L3 miss	A Page Table Entry was loaded into the TLB with data not from local chiplet's L3 due to a data side request.
PM_PTEG_FROM_LMEM	PTEG loaded from local memory	A Page Table Entry was loaded into the TLB with data from local chip's memory due to a data side request.
PM_PTEG_FROM_RL2L3_MOD	PTEG loaded from remote L2 or L3 modified	A Page Table Entry was loaded into the TLB with modified (M) data from a chip's L2 or L3 on same Node (Remote) due to a data side request.
PM_PTEG_FROM_RL2L3_SHR	PTEG loaded from remote L2 or L3 shared	A Page Table Entry was loaded into the TLB with Shared (S) data from a chip's L2 or L3 on same Node (Remote) due to a data side request.
PM_PTEG_FROM_RMEM	PTEG loaded from remote memory	A Page Table Entry was loaded into the TLB with data from another chip's memory on same node due to a data side request.
PM_RUN_CYC	Run_cycles	Processor Cycles gated by the run latch. Operating systems use the run latch to indicate when they are doing useful work. The run latch is typically cleared in the OS idle loop. Gating by the run latch filters out the idle loop.

Event Name	Event Description	Detailed Description
PM_RUN_INST_CMPL	Run_Instructions	Number of PowerPC instructions completed, gated by the run latch.
PM_RUN_PURR	Run_PURR	The Processor Utilization of Resources Register was incremented while the run latch was set.
PM_RUN_SPURR	Run SPURR	The Scaled Processor Utilization of Resources Register was incremented while the run latch was set.
PM_SEG_EXCEPTION	ISEG + DSEG Exception	A DSEG exception because of a DSLB miss has been triggred or ISEG exception because of a ISLB miss has been triggered. The OS handler is invoked to resolve the SLB miss
PM_SHL_CREATED	SHL table entry Created	The Load store unit indicated a store hit load flush , but entry was not found in the Table , so a new entry was created. SHL flushes occur when a younger Load Instruction is issued before an older Store instruction to the same address, and the Load did not return the new Store data. A flush occurs in the LSU when the Store is issued and the Load Reorder Queue (LRQ) shows that a younger load to that same address has already been issued. The flush is needed since the Load has returned stale data.The goal of this mechanism is to prevent further out-of-order issuing of the Store and Load when a SHL flush occurs. If the Store would have been issued first then this flush would not have occurred. The purpose of the SHL Table is then to detect “hazardous loads”, and notify the ISU that they should change the issue policy for this load in order to avoid a potential SHL Flush.
PM_SHL_DEALLOCATED	SHL Table entry deallocated	A SHL Table entry was removed because the ICache was reloaded for that same EA.
PM_SHL_MATCH	SHL Table Match	A Valid Fetch Effective address matched a SHL Table Entry. This indicates that the issue queues should be toggled
PM_SHL_MERGED	SHL table entry merged with existing	The Load store unit indicated a store hit load flush and entry was found in the Table, which caused a merge onto existing entry
PM_SLB_MISS	Data + Instruction SLB Miss - Total of all segment sizes	A SLB miss for a data request or instruction fetch has occurred. SLB misses trap to the operating system to resolve. This is a total count for all segment sizes.
PM_SNOOP_TLBIE	TLBIE snoop	A tlbie was snooped from another processor. This is a per core event, when a snooped tlbie is granted all threads events are turned on.
PM_ST_FIN	Store Instructions Finished	A store finished in the load store unit,
PM_ST_MISS_L1	L1 D cache store misses	A store missed the L1 data cache.
PM_STCX_CMPL	STCX executed	counts when stwcx/stdcx is sent to L2 (no record of issue pipe is maintained so this combined for LSU0 and LSU1). This does not mean the STCX completed successfully
PM_STCX_FAIL	STCX failed	A stcx issued on LSU0 or LSU1 got a fail response from L2. For a stcx, if reservation is not set or the stcx address does not match the reservation address at L2 RC dispatch, then the L2 sends a fail indication to the core)
PM_SUSPENDED	Counter OFF	The counter is suspended (does not count)
PM_TABLEWALK_CYC	Cycles when a tablewalk (I or D) is active	Cycles when we had either a instruction or data tablewalk pending for that thread
PM_TB_BIT_TRANS	Time Base bit transition	When the selected time base bit (as specified in MMCR0[TBSEL])transitions from 0 to 1
PM_THERMAL_MAX	Processor In Thermal MAX	Procssor cycles where thermal max has been reached
PM_THERMAL_WARN	Processor in Thermal Warning	Procssor cycles where thermal warning has occurred
PM_THRD_1_RUN_CYC	1 thread in Run Cycles	Cycles where 1 threads had their run latch

Event Name	Event Description	Detailed Description
PM_THRD_2_CONC_RUN_INSTR	2 thread Concurrent Run Instructions	Cycles where were two threads had their run latch set, and instructions completed for this thread
PM_THRD_2_RUN_CYC	2 thread in Run Cycles	Cycles where 2 threads had their run latch
PM_THRD_3_CONC_RUN_INST	3 thread Concurrent Run Instructions	Cycles where 3 threads had their run latch and instructions completed for this thread
PM_THRD_3_RUN_CYC	3 thread in Run Cycles	Cycles where 3 threads had their run latch
PM_THRD_4_CONC_RUN_INST	4 thread Concurrent Run Instructions	Cycles where all 4 threads had their run latch and instructions completed for this thread
PM_THRD_4_RUN_CYC	4 thread in Run Cycles	All 4 threads had set its run latch. Operating systems use the run latch to indicate when they are doing useful work. The run latch is typically cleared in the OS idle loop. This event does not respect FCWAIT.
PM_THRD_ALL_RUN_CYC	All Threads in run_cycles	Cycles all 4 threads had their run latch set
PM_THRD_CONC_RUN_INST	Concurrent Run Instructions	Cycles where were all threads had their run latch set, and instructions completed for this thread
PM_THRD_GRP_CMPL_BOTH_CYC	Cycles group completed by both threads	Cycles a group was completed by both threads, as we have two completion ports. Core level event
PM_THRD_PRIO_0_1_CYC	Cycles thread running at priority level 0 or 1	Cycles thread running at priority level 0 or 1. Priority is set by writing to TSR , ie or 31,31,31 will trigger this event
PM_THRD_PRIO_2_3_CYC	Cycles thread running at priority level 2 or 3	Cycles thread running at priority level 2 or 3. Priority is set by writing to TSR , (or 1,1,1 OR or 6,6,6 will trigger this event)
PM_THRD_PRIO_4_5_CYC	Cycles thread running at priority level 4 or 5	Cycles thread running at priority level 4 or 5. Priority is set by writing to TSR , (or 2,2,2 OR or 5,5,5 will trigger this event)
PM_THRD_PRIO_6_7_CYC	Cycles thread running at priority level 6 or 7	Cycles thread running at priority level 6 or 7. Priority is set by writing to TSR , (or 3,3,3 OR or 7,7,7 will trigger this event)
PM_THRESH_TIMEO	Threshold timeout event	The threshold timer expired
PM_TLB_MISS	TLB Miss (I + D)	A TLB reload has occurred due to instruction request or data request
PM_VMX_RESULT_SAT_1	Valid result with sat=1	Valid result with saturation=1
PM_VSU_1FLOP	one flop (fadd, fmul, fsub, fcmp, fsel, fabs, fnabs, fres, fsqrte, fneg) operation finished	The vector scalar has finished one flop (fadd, fmul, fsub, fcmp, fsel, fabs, fnabs, fres, fsqrte, fneg, xsadd, xsmul, xssub, xscmp, xssel, xsabs, xsnabs, xsre, xssqrte, xsneg)
PM_VSU_2FLOP	two flops operation (scalar fmadd, fnmadd, fmsub, fnmsub and DP vector versions of single flop instructions)	The vector scalar has finished two flops operation (scalar fmadd, fnmadd, fmsub, fnmsub and DP vector versions of single flop instructions)
PM_VSU_2FLOP_DOUBLE	DP vector version of fmul, fsub, fcmp, fsel, fabs, fnabs, fres ,fsqrte, fneg	The vector scalar has finished two flop DP vector operation (xvadddp, xvmuldp, xvsubdp, xvcmpdp, xvseldp, xvabsdp, xvnabsdp, xvredp ,xvsqrtedp, vxnegdp)
PM_VSU_4FLOP	four flops operation (scalar fdiv, fsqrt; DP vector version of fmadd, fnmadd, fmsub, fnmsub; SP vector versions of single flop instructions)	The vector scalar has finished four flops operation (scalar fdiv, fsqrt; DP vector version of fmadd, fnmadd, fmsub, fnmsub; SP vector versions of single flop instructions)
PM_VSU_8FLOP	eight flops operation (DP vector versions of fdiv,fsqrt and SP vector versions of fmadd,fnmadd,fmsub,fnmsub)	The vector scalar has finished eight flops operation (DP vector versions of fdiv,fsqrt and SP vector versions of fmadd,fnmadd,fmsub,fnmsub)
PM_VSU_DENORM	Vector or Scalar denorm operand	The vector scalar has finished A FPU denorm operand
PM_VSU_FCONV	Convert instruction executed	The vector scalar unit has executed Convert instruction
PM_VSU_FEST	Estimate instruction executed	The vector scalar unit has executed Estimate instruction

Event Name	Event Description	Detailed Description
PM_VSU_FIN	VSU0 Finished an instruction	The Vector scalar unit finished a instruction. Instructions that finish may not necessary complete
PM_VSU_FMA	two flops operation (fmadd, fnmadd, fmsub, fnmsub) Scalar instructions only!	The vector scalar unit has finished two flops operation (fmadd, fnmadd, fmsub, fnmsub, xsmadd, xsnmadd, xsmsub, xsnmsub) Scalar instructions only!
PM_VSU_FMA_DOUBLE	DP vector version of fmadd,fnmadd,fmsub,fnmsub	The vector scalar unit has finished four flop DP vector operations (xvmadddp, xvnmadddp, xvmsubdp, xvmsubdp)
PM_VSU_FRSP	Round to single precision instruction executed	The vector scalar unit has executed Round to single precision instruction
PM_VSU_FSQRT_FDIV	four flops operation (fdiv,fsqrt) Scalar Instructions only!	The vector scalar unit has finished four flops operation (fdiv,fsqrt,xsdiv,xssqrt) Scalar Instructions only!
PM_VSU_FSQRT_FDIV_DOUBLE	DP vector versions of fdiv,fsqrt	The vector scalar unit has finished eight flop DP vector operations (xvfdivdp, xvsqrtdp
PM_VSU_SCALAR_DOUBLE_ISSUED	Double Precision scalar instruction issued on Pipe0	The vector scalar unit has issued Double Precision scalar instruction
PM_VSU_SCALAR_SINGLE_ISSUED	Single Precision scalar instruction issued on Pipe0	The vector scalar unit has issued Single Precision scalar instruction
PM_VSU_SIMPLE_ISSUED	Simple VMX instruction issued	The vector scalar unit has issued Simple VMX instruction
PM_VSU_SINGLE	Vector or Scalar single precision	The vector scalar unit has finished FPU single precision instruction
PM_VSU_STF	FPU store (SP or DP) issued on Pipe0	The vector scalar unit has issued FPU store (SP or DP)
PM_VSU_VECTOR_DOUBLE_ISSUED	Double Precision vector instruction issued on Pipe0	The vector scalar unit has issued Double Precision vector instruction
PM_VSU_VECTOR_SINGLE_ISSUED	Single Precision vector instruction issued (executed)	The vector scalar unit has issued Double Precision vector instruction
PM_VSU0_16FLOP	Sixteen flops operation (SP vector versions of fdiv,fsqrt)	The vector scalar unit 0 (VSU0) has finished Sixteen flop operation (SP vector versions of fdiv,fsqrt).
PM_VSU0_1FLOP	one flop (fadd, fmul, fsub, fcmp, fsel, fabs, fnabs, fres, fsqrte, fneg, xsadd, xsmul, xssub, xscmp, xssel, xsabs, xsnabs, xsre, xssqrte, xsneg) operation finished	The vector scalar unit 0 (VSU0) has finished one flop (fadd, fmul, fsub, fcmp, fsel, fabs, fnabs, fres, fsqrte, fneg, xsadd, xsmul, xssub, xscmp, xssel, xsabs, xsnabs, xsre, xssqrte, xsneg)
PM_VSU0_2FLOP	two flops operation (scalar fmadd, fnmadd, fmsub, fnmsub and DP vector versions of single flop instructions)	The vector scalar unit 0 (VSU0) has finished two flops operation (scalar fmadd, fnmadd, fmsub, fnmsub and DP vector versions of single flop instructions)
PM_VSU0_2FLOP_DOUBLE	two flop DP vector operation (xvadddp, xvmuldp, xvsubdp, xvcmpdp, xvseldp, xvabsdp, xvnabsdp, xvredp ,xvsqrtedp, vxnegdp)	The vector scalar unit 0 (VSU0) has finished two flop DP vector operation (xvadddp, xvmuldp, xvsubdp, xvcmpdp, xvseldp, xvabsdp, xvnabsdp, xvredp ,xvsqrtedp, vxnegdp)
PM_VSU0_4FLOP	four flops operation (scalar fdiv, fsqrt; DP vector version of fmadd, fnmadd, fmsub, fnmsub; SP vector versions of single flop instructions)	The vector scalar unit 0 (VSU0) has finished four flops operation (scalar fdiv, fsqrt; DP vector version of fmadd, fnmadd, fmsub, fnmsub; SP vector versions of single flop instructions)
PM_VSU0_8FLOP	eight flops operation (DP vector versions of fdiv,fsqrt and SP vector versions of fmadd,fnmadd,fmsub,fnmsub)	The vector scalar unit 0 (VSU0) has finished eight flops operation (DP vector versions of fdiv,fsqrt and SP vector versions of fmadd,fnmadd,fmsub,fnmsub)
PM_VSU0_COMPLEX_ISSUED	Complex VMX instruction issued	The vector scalar unit 0 (VSU0) has issued a Complex VMX instruction
PM_VSU0_DENORM	FPU denorm operand	The vector scalar unit 0 (VSU0) has finished A FPU denorm operand
PM_VSU0_FCONV	Convert instruction executed	The vector scalar unit 0 (VSU0) has executed Convert instruction
PM_VSU0_FEST	Estimate instruction executed	The vector scalar unit 0 (VSU0) has executed Estimate instruction
PM_VSU0_FIN	VSU0 Finished an instruction	The vector scalar unit 0 (VSU0) has Finished an instruction

Event Name	Event Description	Detailed Description
PM_VSU0_FMA	two flops operation (fmadd, fnmadd, fmsub, fnmsub, xsmadd, xsnmadd, xsmsub, xsnmsub) Scalar instructions only!	The vector scalar unit 0 (VSU0) has finished two flops operation (fmadd, fnmadd, fmsub, fnmsub, xsmadd, xsnmadd, xsmsub, xsnmsub) Scalar instructions only!
PM_VSU0_FMA_DOUBLE	four flop DP vector operations (xvmadddp, xvnmadddp, xvmsubdp, xvmsubdp)	The vector scalar unit 0 (VSU0) has finished four flop DP vector operations (xvmadddp, xvnmadddp, xvmsubdp, xvmsubdp)
PM_VSU0_FPSCR	Move to/from FPSCR type instruction issued on Pipe 0	The vector scalar unit 0 (VSU0) has issued Move to/from FPSCR type instruction
PM_VSU0_FRSP	Round to single precision instruction executed	The vector scalar unit 0 (VSU0) has executed Round to single precision instruction
PM_VSU0_FSQRT_FDIV	four flops operation (fdiv,fsqrt,xsdiv,xssqrt) Scalar Instructions only!	The vector scalar unit 0 (VSU0) has finished four flops operation (fdiv,fsqrt,xsdiv,xssqrt) Scalar Instructions only!
PM_VSU0_FSQRT_FDIV_DOUBLE	eight flop DP vector operations (xvfdivdp, xvsqrtdp	The vector scalar unit 0 (VSU0) has finished eight flop DP vector operations (xvfdivdp, xvsqrtdp
PM_VSU0_SCAL_DOUBLE_ISSUED	Double Precision scalar instruction issued on Pipe0	The vector scalar unit 0 (VSU0) has issued Double Precision scalar instruction
PM_VSU0_SCAL_SINGLE_ISSUED	Single Precision scalar instruction issued on Pipe0	The vector scalar unit 0 (VSU0) has issued Single Precision scalar instruction
PM_VSU0_SIMPLE_ISSUED	Simple VMX instruction issued	The vector scalar unit 0 (VSU0) has issued Simple VMX instruction
PM_VSU0_SINGLE	FPU single precision	The vector scalar unit 0 (VSU0) has finished FPU single precision instruction
PM_VSU0_STF	FPU store (SP or DP) issued on Pipe0	The vector scalar unit 0 (VSU0) has issued FPU store (SP or DP) issued
PM_VSU0_VECT_DOUBLE_ISSUED	Double Precision vector instruction issued on Pipe0	The vector scalar unit 0 (VSU0) has issued Double Precision vector instruction
PM_VSU0_VECTOR_SP_ISSUED	Single Precision vector instruction issued (executed)	The vector scalar unit 0 (VSU0) has issued Single Precision vector instruction
PM_VSU1_1FLOP	one flop (fadd, fmul, fsub, fcmp, fsel, fabs, fnabs, fres, fsqrte, fneg, xsadd, xsmul, xssub, xscmp, xssel, xsabs, xsnabs, xsre, xssqrte, xsneg) operation finished	The vector scalar unit 1 (VSU1) has finished one flop (fadd, fmul, fsub, fcmp, fsel, fabs, fnabs, fres, fsqrte, fneg, xsadd, xsmul, xssub, xscmp, xssel, xsabs, xsnabs, xsre, xssqrte, xsneg) operation
PM_VSU1_2FLOP	two flops operation (scalar fmadd, fnmadd, fmsub, fnmsub and DP vector versions of single flop instructions)	The vector scalar unit 1 (VSU1) has finished two flops operation (scalar fmadd, fnmadd, fmsub, fnmsub and DP vector versions of single flop instructions)
PM_VSU1_2FLOP_DOUBLE	two flop DP vector operation (xvadddp, xvmuldp, xvsubdp, xvcmpdp, xvseldp, xvabsdp, xvnabsdp, xvredp ,xvsqrtedp, vxnegdp)	The vector scalar unit 1 (VSU1) has finished two flop DP vector operation (xvadddp, xvmuldp, xvsubdp, xvcmpdp, xvseldp, xvabsdp, xvnabsdp, xvredp ,xvsqrtedp, vxnegdp)
PM_VSU1_4FLOP	four flops operation (scalar fdiv, fsqrt; DP vector version of fmadd, fnmadd, fmsub, fnmsub; SP vector versions of single flop instructions)	The vector scalar unit 1 (VSU1) has finished four flops operation (scalar fdiv, fsqrt; DP vector version of fmadd, fnmadd, fmsub, fnmsub; SP vector versions of single flop instructions)
PM_VSU1_8FLOP	eight flops operation (DP vector versions of fdiv,fsqrt and SP vector versions of fmadd,fnmadd,fmsub,fnmsub)	The vector scalar unit 1 (VSU1) has finished eight flops operation (DP vector versions of fdiv,fsqrt and SP vector versions of fmadd,fnmadd,fmsub,fnmsub)
PM_VSU1_DD_ISSUED	64BIT Decimal Issued on Pipe1	The vector scalar unit 1 (VSU1) has issued 64BIT Decimal
PM_VSU1_DENORM	FPU denorm operand	The vector scalar unit 1 (VSU1) has finished A FPU denorm operand
PM_VSU1_DQ_ISSUED	128BIT Decimal Issued on Pipe1	The vector scalar unit 1 (VSU1) has issued 128BIT Decimal Issued
PM_VSU1_FCONV	Convert instruction executed	The vector scalar unit 1 (VSU1) has executed Convert instruction
PM_VSU1_FEST	Estimate instruction executed	The vector scalar unit 1 (VSU1) has executed Estimate instruction

Event Name	Event Description	Detailed Description
PM_VSU1_FIN	VSU1 Finished an instruction	The vector scalar unit 1 (VSU1) has Finished an instruction
PM_VSU1_FMA	two flops operation (fmadd, fnmadd, fmsub, fnmsub, xsmadd, xsnmadd, xsmsub, xsnmsub) Scalar instructions only!	The vector scalar unit 1 (VSU1) has finished two flops operation (fmadd, fnmadd, fmsub, fnmsub, xsmadd, xsnmadd, xsmsub, xsnmsub) Scalar instructions only!
PM_VSU1_FMA_DOUBLE	four flop DP vector operations (xvmadddp, xvnmadddp, xvmsubdp, xvmsubdp)	The vector scalar unit 1 (VSU1) has finished four flop DP vector operations (xvmadddp, xvnmadddp, xvmsubdp, xvmsubdp)
PM_VSU1_FRSP	Round to single precision instruction executed	The vector scalar unit 1 (VSU1) has executed Round to single precision instruction
PM_VSU1_FSQRT_FDIV	four flops operation (fdiv,fsqrt,xsdiv,xssqrt) Scalar Instructions only!	The vector scalar unit 1 (VSU1) has finished four flops operation (fdiv,fsqrt,xsdiv,xssqrt) Scalar Instructions only!
PM_VSU1_FSQRT_FDIV_DOUBLE	eight flop DP vector operations (xvfdivdp, xvsqrtdp	The vector scalar unit 1 (VSU1) has finished eight flop DP vector operations (xvfdivdp, xvsqrtdp
PM_VSU1_PERMUTE_ISSUED	Permute VMX Instruction Issued	The vector scalar unit 1 (VSU1) has issued Permute VMX Instruction Issued
PM_VSU1_SCAL_DOUBLE_ISSUED	Double Precision scalar instruction issued on Pipe1	The vector scalar unit 1 (VSU1) has issued Double Precision scalar instruction
PM_VSU1_SCAL_SINGLE_ISSUED	Single Precision scalar instruction issued on Pipe1	The vector scalar unit 1 (VSU1) has issued Single Precision scalar instruction
PM_VSU1_SINGLE	FPU single precision	The vector scalar unit 1 (VSU1) has finished FPU single precision instruction
PM_VSU1_SQ	Store Vector Issued on Pipe1	The vector scalar unit 1 (VSU1) has issued Store Vector
PM_VSU1_STF	FPU store (SP or DP) issued on Pipe1	The vector scalar unit 1 (VSU1) has issued FPU store (SP or DP)
PM_VSU1_VECT_DOUBLE_ISSUED	Double Precision vector instruction issued on Pipe1	The vector scalar unit 1 (VSU1) has issued Double Precision vector instruction

Appendix C. Abbreviations Used

BRU BRanch execution Unit
CRL Condition Register Logical pipeline
DFU Decimal Floating-point Unit
FPU Floating-Point Unit
FXU Fixed-Point Unit
IFU Instruction Fetch Unit
LSU Load/Store Unit
PMC Performance Monitoring Counter
PMU Performance Monitoring Unit
VSU Vector-Scalar Unit

Appendix D. Notices

IBM Corporation
Marketing Communications, Systems Group
Route 100, Somers, New York 10589
Produced in the United States of America
March 2010, All Rights Reserved

those products.
This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.
COPYRIGHT LICENSE:
This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs.
More details can be found at the IBM Power Systems home page

Appendix E. Trademarks

IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both. These and other IBM trademarked terms are marked on their first occurrence in this information with the appropriate symbol (® or ™), indicating US registered or common law trademarks owned by IBM at the time this information was published. Such trademarks may also be registered or common law trademarks in other countries. A current list of IBM trademarks is available on the Web at http://www.ibm.com/legal/copytrade.shtml.
The following terms are trademarks of the International Business Machines Corporation in the United States, other countries, or both:

1350™
AIX 5L™
AIX®
alphaWorks®
Ascendant®
BetaWorks™
BladeCenter®
CICS®
Cool Blue™
DB2®
developerWorks®
Domino®
EnergyScale™
Enterprise Storage Server®
Enterprise Workload Manager™
eServer™
Express Portfolio™
FlashCopy®
GDPS®
General Parallel File System™
Geographically Dispersed Parallel
Sysplex™
Global Innovation Outlook™
GPFS™
HACMP™
HiperSockets™
HyperSwap™
i5/OS®
IBM Process Reference Model for

IT™
IBM Systems Director Active Energy Manager™
IBM®
IntelliStation®
Lotus Notes®
Lotus®
MQSeries®
MVS™
Netfinity®
Notes®
OS/390®
Parallel Sysplex®
PartnerWorld®
POWER™
POWER®
POWER4™
POWER5™
POWER6™
POWER7™
PowerExecutive™
Power Systems™
PowerPC®
PowerVM™
PR/SM™
pSeries®
QuickPlace®
RACF®
Rational Summit®
Rational Unified Process®
Rational®
Redbooks®
Redbooks (logo) ®
RS/6000®
RUP®
S/390®
Sametime®
Summit Ascendant™
Summit®
System i™
System p™

System Storage™
System x™
System z™
System z10™
System/360™
System/370™
Tivoli®
TotalStorage®
VM/ESA®
VSE/ESA™
WebSphere®
Workplace™
Workplace Messaging®
X-Architecture®
xSeries®
z/OS®
z/VM®
z10™
zSeries®

The following terms are trademarks of other companies:
•AltiVec™ is a trademark of Freescale Semiconductor, Inc.
•AMD™, AMD Opteron™, the AMD Arrow logo, and combinations thereof, are trademarks of Advanced Micro Devices, Inc.
•InfiniBand™, and the InfiniBand design marks are trademarks and/or service marks of the InfiniBand Trade Association.
•ITIL® is a registered trademark, and a registered community trademark of the Office of Government Commerce, and is registered in the U.S. Patent and Trademark Office.
•IT Infrastructure Library® is a registered trademark of the Central Computer and Telecommunications Agency which is now part of the Office of Government Commerce.
•Novell®, SUSE®, the Novell logo, and the N logo are registered trademarks of Novell, Inc. in the United States and other countries.
•Oracle®, JD Edwards®, PeopleSoft®, Siebel®, and TopLink® are registered trademarks of Oracle Corporation and/or its affiliates.
•SAP NetWeaver®, SAP R/3®, SAP®, and SAP logos are trademarks or registered trademarks of SAP AG in Germany and in several other countries.
•IQ™, J2EE™, Java™, JDBC™, Netra™, Solaris™, Sun™, and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.
•Microsoft™, Windows™, Windows NT™, Outlook™, SQL Server™, Windows Server™, Windows™, and the Windows logo are trademarks of Microsoft Corporation in the United States, other

countries, or both.
•Intel Xeon™, Intel™, Itanium™, Intel logo, Intel Inside logo, and Intel Centrino logo are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States, other countries, or both.
•QLogic™, the QLogic logo are the trademarks or registered trademarks of QLogic Corporation.
•SilverStorm™ is a trademark of QLogic Corporation.
•SPEC® is a registered trademark of Standard Performance Evaluation Corporation.
•SPEC MPI® is a registered trademark of Standard Performance Evaluation Corporation.
•UNIX® is a registered trademark of The Open Group in the United States and other countries.
•Linux™ is a trademark of Linus Torvalds in the United States, other countries, or both.
Other company, product, or service names may be trademarks or service marks of others.