A64FX®

PMU Events

Revision History

Change Date	Edition	Description of Change
2/28/2020	1.1	First Release
4/28/2020	1.2	Correct typos

Introduction

The A64FX processor (called A64FX, below) is a superscalar processor of the out-of-order execution type. The A64FX is designed for high-performance computing (HPC) and complies with the ARMv8-A architecture profile and the Scalable Vector Extension for ARMv8-A. The processor integrates 52 processor cores including redundant cores; a memory controller supporting HBM2; a Tofu-D interconnect controller; and a root complex supporting PCI-Express Gen3.

Events

ARMv8 Common Events

0x0000, SW_INCR

This event counts on writes to the PMSWINC register.

0x0001, L1I_CACHE_REFILL

This event counts operations that cause a refill of the L1I cache.
See L1I_CACHE_REFILL of ARMv8 Reference Manual for more information.

0x0002, L1I_TLB_REFILL

This event counts operations that cause a TLB refill of the L1I TLB.
See L1I_TLB_REFILL of ARMv8 Reference Manual for more information.

0x0003, L1D_CACHE_REFILL

This event counts operations that cause a refill of the L1D cache.
See L1D_CACHE_REFILL of ARMv8 Reference Manual for more information.

0x0004, L1D_CACHE

This event counts operations that cause a cache access to the L1D cache.
See L1D_CACHE of ARMv8 Reference Manual for more information.

0x0005, L1D_TLB_REFILL

This event counts operations that cause a TLB refill of the L1D TLB.
See L1D_TLB_REFILL of ARMv8 Reference Manual for more information.

0x0008, INST_RETIRED

This event counts every architecturally executed instruction.

0x0009, EXC_TAKEN

This event counts each exception taken.

0x000a, EXC_RETURN

This event counts each executed exception return instruction.

0x000b, CID_WRITE_RETIRED

This event counts every write to CONTEXTIDR.

0x0010, BR_MIS_PRED

This event counts each correction to the predicted program flow that occurs because of a misprediction from, or no prediction from, the branch prediction resources and that relates to instructions that the branch prediction resources are capable of predicting.

0x0011, CPU_CYCLES

This event counts every cycle.

0x0012, BR_PRED

This event counts every branch or other change in the program flow that the branch prediction resources are capable of predicting.

0x0014, L1I_CACHE

This event counts operations that cause a cache access to the L1I cache.
See L1I_CACHE of ARMv8 Reference Manual for more information.

0x0015, L1D_CACHE_WB

This event counts every write-back of data from the L1D cache.
See L1D_CACHE_WB of ARMv8 Reference Manual for more information.

0x0016, L2D_CACHE

This event counts operations that cause a cache access to the L2 cache.
See L2D_CACHE of ARMv8 Reference Manual for more information.

0x0017, L2D_CACHE_REFILL

This event counts operations that cause a refill of the L2 cache.
See L2D_CACHE_REFILL of ARMv8 Reference Manual for more information.

0x0018, L2D_CACHE_WB

This event counts every write-back of data from the L2 cache.
See L2D_CACHE_WB of ARMv8 Reference Manual for more information.

0x001b, INST_SPEC

This event counts every architecturally executed instruction.

0x0023, STALL_FRONTEND

This event counts every cycle counted by the CPU_CYCLES event on that no operation was issued because there are no operations available to issue for this PE from the frontend.

0x0024, STALL_BACKEND

This event counts every cycle counted by the CPU_CYCLES event on that no operation was issued because the backend is unable to accept any operations.

0x002d, L2D_TLB_REFILL

This event counts operations that cause a TLB refill of the L2D TLB.
See L2D_TLB_REFILL of ARMv8 Reference Manual for more information.

0x002e, L2I_TLB_REFILL

This event counts operations that cause a TLB refill of the L2I TLB.
See L2I_TLB_REFILL of ARMv8 Reference Manual for more information.

0x002f, L2D_TLB

This event counts operations that cause a TLB access to the L2D TLB.
See L2D_TLB of ARMv8 Reference Manual for more information.

0x0030, L2I_TLB

This event counts operations that cause a TLB access to the L2I TLB.
See L2I_TLB of ARMv8 Reference Manual for more information.

0x0049, L1D_CACHE_REFILL_PRF

This event counts L1D_CACHE_REFILL caused by software or hardware prefetch.

0x0059, L2D_CACHE_REFILL_PRF

This event counts L2D_CACHE_REFILL caused by software or hardware prefetch.

0x006c, LDREX_SPEC

This event counts architecturally executed load-exclusive instructions.

0x006f, STREX_SPEC

This event counts architecturally executed store-exclusive instructions.

0x0070, LD_SPEC

This event counts architecturally executed memory-reading instructions, as defined by the LD_RETIRED event.

0x0071, ST_SPEC

This event counts architecturally executed memory-writing instructions, as defined by the ST_RETIRED event. This event counts DCZVA as a store operation.

0x0072, LDST_SPEC

This event counts architecturally executed memory-reading instructions and memory-writing instructions, as defined by the LD_RETIRED and ST_RETIRED events.

0x0073, DP_SPEC

This event counts architecturally executed integer data-processing instructions. See DP_SPEC of ARMv8 Reference Manual for more information.

0x0074, ASE_SPEC

This event counts architecturally executed Advanced SIMD data-processing instructions.

0x0075, VFP_SPEC

This event counts architecturally executed floating-point data-processing instructions.

0x0076, PC_WRITE_SPEC

This event counts only software changes of the PC that defined by the instruction architecturally executed, condition code check pass, software change of the PC event.

0x0077, CRYPTO_SPEC

This event counts architecturally executed cryptographic instructions, except PMULL and VMULL.

0x0078, BR_IMMED_SPEC

This event counts architecturally executed immediate branch instructions.

0x0079, BR_RETURN_SPEC

This event counts architecturally executed procedure return operations that defined by the BR_RETURN_RETIRED event.

0x007a, BR_INDIRECT_SPEC

This event counts architecturally executed indirect branch instructions that includes software change of the PC other than exception-generating instructions and immediate branch instructions.

0x007c, ISB_SPEC

This event counts architecturally executed Instruction Synchronization Barrier instructions.

0x007d, DSB_SPEC

This event counts architecturally executed Data Synchronization Barrier instructions.

0x007e, DMB_SPEC

This event counts architecturally executed Data Memory Barrier instructions, excluding the implied barrier operations of load/store operations with release consistency semantics.

0x0081, EXC_UNDEF

This event counts only other synchronous exceptions that are taken locally.

0x0082, EXC_SVC

This event counts only Supervisor Call exceptions that are taken locally.

0x0083, EXC_PABORT

This event counts only Instruction Abort exceptions that are taken locally.

0x0084, EXC_DABORT

This event counts only Data Abort or SError interrupt exceptions that are taken locally.

0x0086, EXC_IRQ

This event counts only IRQ exceptions that are taken locally, including Virtual IRQ exceptions.

0x0087, EXC_FIQ

This event counts only FIQ exceptions that are taken locally, including Virtual FIQ exceptions.

0x0088, EXC_SMC

"This event counts only Secure Monitor Call exceptions.
The counter does not increment on SMC instructions trapped as a Hyp Trap exception."

0x008a, EXC_HVC

This event counts for both Hypervisor Call exceptions taken locally in the hypervisor and those taken as an exception from Non-secure EL1.

0x009f, DCZVA_SPEC

This event counts architecturally executed zero blocking operations due to the "DC ZVA" instruction.

A64FX Specific Events

0x0105, FP_MV_SPEC

This event counts architecturally executed floating-point move operations.

0x0108, PRD_SPEC

This event counts architecturally executed operations that using predicate register.

0x0109, IEL_SPEC

This event counts architecturally executed inter-element manipulation operations.

0x010a, IREG_SPEC

This event counts architecturally executed inter-register manipulation operations.

0x0112, FP_LD_SPEC

This event counts architecturally executed NOSIMD load operations that using SIMD&FP registers.

0x0113, FP_ST_SPEC

This event counts architecturally executed NOSIMD store operations that using SIMD&FP registers.

0x011a, BC_LD_SPEC

This event counts architecturally executed SIMD broadcast floating-point load operations.

0x0121, EFFECTIVE_INST_SPEC

This event counts architecturally executed instructions, excluding the MOVPRFX instruction.

0x0123, PRE_INDEX_SPEC

This event counts architecturally executed operations that uses "pre-index" as its addressing mode.

0x0124, POST_INDEX_SPEC

This event counts architecturally executed operations that uses "post-index" as its addressing mode.

0x0139, UOP_SPLIT

This event counts the occurrence count of the micro-operation split.

0x0180, LD_COMP_WAIT_L2_MISS

This event counts every cycle that no operation was committed because the oldest and uncommitted load/store/prefetch operation waits for memory access.

0x0181, LD_COMP_WAIT_L2_MISS_EX

This event counts every cycle that no instruction was committed because the oldest and uncommitted integer load operation waits for memory access.

0x0182, LD_COMP_WAIT_L1_MISS

This event counts every cycle that no instruction was committed because the oldest and uncommitted load/store/prefetch operation waits for L2 cache access.

0x0183, LD_COMP_WAIT_L1_MISS_EX

This event counts every cycle that no instruction was committed because the oldest and uncommitted integer load operation waits for L2 cache access.

0x0184, LD_COMP_WAIT

This event counts every cycle that no instruction was committed because the oldest and uncommitted load/store/prefetch operation waits for L1D cache, L2 cache and memory access.

0x0185, LD_COMP_WAIT_EX

This event counts every cycle that no instruction was committed because the oldest and uncommitted integer load operation waits for L1D cache, L2 cache and memory access.

0x0186, LD_COMP_WAIT_PFP_BUSY

This event counts every cycle that no instruction was committed due to the lack of an available prefetch port.

0x0187, LD_COMP_WAIT_PFP_BUSY_EX

This event counts the LD_COMP_WAIT_PFP_BUSY caused by an integer load operation.

0x0188, LD_COMP_WAIT_PFP_BUSY_SWPF

This event counts the LD_COMP_WAIT_PFP_BUSY caused by a software prefetch instruction.

0x0189, EU_COMP_WAIT

This event counts every cycle that no instruction was committed and the oldest and uncommitted instruction is an integer or floating-point/SIMD instruction.

0x018a, FL_COMP_WAIT

This event counts every cycle that no instruction was committed and the oldest and uncommitted instruction is a floating-point/SIMD instruction.

0x018b, BR_COMP_WAIT

This event counts every cycle that no instruction was committed and the oldest and uncommitted instruction is a branch instruction.

0x018c, ROB_EMPTY

This event counts every cycle that no instruction was committed because the CSE is empty.

0x018d, ROB_EMPTY_STQ_BUSY

This event counts every cycle that no instruction was committed because the CSE is empty and the store port (SP) is full.

0x018e, WFE_WFI_CYCLE

This event counts every cycle that the instruction unit is halted by the WFE/WFI instruction.

0x0190, 0INST_COMMIT

This event counts every cycle that no instruction was committed, but counts at the time when commits MOVPRFX only.

0x0191, 1INST_COMMIT

This event counts every cycle that one instruction is committed.

0x0192, 2INST_COMMIT

This event counts every cycle that two instructions are committed.

0x0193, 3INST_COMMIT

This event counts every cycle that three instructions are committed.

0x0194, 4INST_COMMIT

This event counts every cycle that four instructions are committed.

0x0198, UOP_ONLY_COMMIT

This event counts every cycle that only any micro-operations are committed.

0x0199, SINGLE_MOVPRFX_COMMIT

This event counts every cycle that only the MOVPRFX instruction is committed.

0x01a0, EAGA_VAL

This event counts valid cycles of EAGA pipeline.

0x01a1, EAGB_VAL

This event counts valid cycles of EAGB pipeline.

0x01a2, EXA_VAL

This event counts valid cycles of EXA pipeline.

0x01a3, EXB_VAL

This event counts valid cycles of EXB pipeline.

0x01a4, FLA_VAL

This event counts valid cycles of FLA pipeline.

0x01a5, FLB_VAL

This event counts valid cycles of FLB pipeline.

0x01a6, PRX_VAL

This event counts valid cycles of PRX pipeline.

0x01b4, FLA_VAL_PRD_CNT

This event counts the number of 1's in the predicate bits of request in FLA pipeline, where it is corrected so that it becomes 16 when all bits are 1.

0x01b5, FLB_VAL_PRD_CNT

This event counts the number of 1's in the predicate bits of request in FLB pipeline, where it is corrected so that it becomes 16 when all bits are 1.

0x01e0, EA_CORE

This event counts energy consumption per cycle of core.

0x0200, L1D_CACHE_REFILL_DM

This event counts L1D_CACHE_REFILL caused by demand access.

0x0202, L1D_CACHE_REFILL_HWPRF

This event counts L1D_CACHE_REFILL caused by hardware prefetch.

0x0208, L1_MISS_WAIT

This event counts outstanding L1D cache miss requests per cycle.

0x0209, L1I_MISS_WAIT

This event counts outstanding L1I cache miss requests per cycle.

0x0230, L1HWPF_STREAM_PF

This event counts streaming prefetch requests to L1D cache generated by hardware prefetcher.

0x0231, L1HWPF_INJ_ALLOC_PF

This event counts allocation type prefetch injection requests to L1D cache generated by hardware prefetcher.

0x0232, L1HWPF_INJ_NOALLOC_PF

This event counts non-allocation type prefetch injection requests to L1D cache generated by hardware prefetcher.

0x0233, L2HWPF_STREAM_PF

This event counts streaming prefetch requests to L2 cache generated by hardware prefecher.

0x0234, L2HWPF_INJ_ALLOC_PF

This event counts allocation type prefetch injection requests to L2 cache generated by hardware prefetcher.

0x0235, L2HWPF_INJ_NOALLOC_PF

This event counts non-allocation type prefetch injection requests to L2 cache generated by hardware prefetcher.

0x0236, L2HWPF_OTHER

This event counts prefetch requests to L2 cache generated by the other causes.

0x0240, L1_PIPE0_VAL

This event counts valid cycles of L1D cache pipeline#0.

0x0241, L1_PIPE1_VAL

This event counts valid cycles of L1D cache pipeline#1.

0x0250, L1_PIPE0_VAL_IU_TAG_ADRS_SCE

This event counts requests in L1D cache pipeline#0 that its sce bit of tagged address is 1.

0x0251, L1_PIPE0_VAL_IU_TAG_ADRS_PFE

This event counts requests in L1D cache pipeline#0 that its pfe bit of tagged address is 1.

0x0252, L1_PIPE1_VAL_IU_TAG_ADRS_SCE

This event counts requests in L1D cache pipeline#1 that its sce bit of tagged address is 1.

0x0253, L1_PIPE1_VAL_IU_TAG_ADRS_PFE

This event counts requests in L1D cache pipeline#1 that its pfe bit of tagged address is 1.

0x0260, L1_PIPE0_COMP

This event counts completed requests in L1D cache pipeline#0.

0x0261, L1_PIPE1_COMP

This event counts completed requests in L1D cache pipeline#1.

0x0268, L1I_PIPE_COMP

This event counts completed requests in L1I cache pipeline.

0x0269, L1I_PIPE_VAL

This event counts valid cycles of L1I cache pipeline.

0x0274, L1_PIPE_ABORT_STLD_INTLK

This event counts aborted requests in L1D pipelines that due to store-load interlock.

0x02a0, L1_PIPE0_VAL_IU_NOT_SEC0

This event counts requests in L1D cache pipeline#0 that its sector cache ID is not 0.

0x02a1, L1_PIPE1_VAL_IU_NOT_SEC0

This event counts requests in L1D cache pipeline#1 that its sector cache ID is not 0.

0x02b0, L1_PIPE_COMP_GATHER_2FLOW

This event counts the number of times where 2 elements of the gather instructions became 2 flows because 2 elements could not be combined.

0x02b1, L1_PIPE_COMP_GATHER_1FLOW

This event counts the number of times where 2 elements of the gather instructions became 1 flow because 2 elements could be combined.

0x02b2, L1_PIPE_COMP_GATHER_0FLOW

This event counts the number of times where 2 elements of the gather instructions became 0 flow because both predicate values are 0.

0x02b3, L1_PIPE_COMP_SCATTER_1FLOW

This event counts the number of flows of the scatter instructions.

0x02b8, L1_PIPE0_COMP_PRD_CNT

This event counts the number of 1's in the predicate bits of request in L1D cache pipeline#0, where it is corrected so that it becomes 16 when all bits are 1.

0x02b9, L1_PIPE1_COMP_PRD_CNT

This event counts the number of 1's in the predicate bits of request in L1D cache pipeline#1, where it is corrected so that it becomes 16 when all bits are 1.

0x0300, L2D_CACHE_REFILL_DM

This event counts L2D_CACHE_REFILL caused by demand access.

0x0302, L2D_CACHE_REFILL_HWPRF

This event counts L2D_CACHE_REFILL caused by hardware prefetch.

0x0308, L2_MISS_WAIT

This event counts outstanding L2 cache miss requests per cycle.
It counts all events caused in measured CMG regardless of measured PE.

0x0309, L2_MISS_COUNT

This event counts the number of times of L2 cache miss.
It counts all events caused in measured CMG regardless of measured PE.

0x0314, BUS_READ_TOTAL_TOFU

This event counts read transactions from tofu controller to measured CMG.
It counts all events caused in measured CMG regardless of measured PE.

0x0315, BUS_READ_TOTAL_PCI

This event counts read transactions from PCI controller to measured CMG.
It counts all events caused in measured CMG regardless of measured PE.

0x0316, BUS_READ_TOTAL_MEM

This event counts read transactions from measured CMG local memory to measured CMG.
It counts all events caused in measured CMG regardless of measured PE.

0x0318, BUS_WRITE_TOTAL_CMG0

This event counts write transactions from measured CMG to CMG0, if measured CMG is not CMG0.
Otherwise, this event counts write transactions from measured CMG to CMG0 local memory.
It counts all events caused in measured CMG regardless of measured PE.

0x0319, BUS_WRITE_TOTAL_CMG1

This event counts write transactions from measured CMG to CMG1, if measured CMG is not CMG1.
Otherwise, this event counts write transactions from measured CMG to CMG1 local memory.
It counts all events caused in measured CMG regardless of measured PE.

0x031a, BUS_WRITE_TOTAL_CMG2

This event counts write transactions from measured CMG to CMG2, if measured CMG is not CMG2.
Otherwise, this event counts write transactions from measured CMG to CMG2 local memory.
It counts all events caused in measured CMG regardless of measured PE.

0x031b, BUS_WRITE_TOTAL_CMG3

This event counts write transactions from measured CMG to CMG3, if measured CMG is not CMG3.
Otherwise, this event counts write transactions from measured CMG to CMG3 local memory.
It counts all events caused in measured CMG regardless of measured PE.

0x031c, BUS_WRITE_TOTAL_TOFU

This event counts write transactions from measured CMG to tofu controller.
It counts all events caused in measured CMG regardless of measured PE.

0x031d, BUS_WRITE_TOTAL_PCI

This event counts write transactions from measured CMG to PCI controller.
It counts all events caused in measured CMG regardless of measured PE.

0x031e, BUS_WRITE_TOTAL_MEM

This event counts write transactions from measured CMG to measured CMG local memory.
It counts all events caused in measured CMG regardless of measured PE.

0x0325, L2D_SWAP_DM

This event counts operations where demand access hits an L2 cache refill buffer allocated by software or hardware prefetch.

0x0326, L2D_CACHE_MIBMCH_PRF

This event counts operations where software or hardware prefetch hits an L2 cache refill buffer allocated by demand access.

0x0330, L2_PIPE_VAL

This event counts valid cycles of L2 cache pipeline.
It counts all events caused in measured CMG regardless of measured PE.

0x0350, L2_PIPE_COMP_ALL

This event counts completed requests in L2 cache pipeline.
It counts all events caused in measured CMG regardless of measured PE.

0x0370, L2_PIPE_COMP_PF_L2MIB_MCH

This event counts operations where software or hardware prefetch hits an L2 cache refill buffer allocated by demand access.
It counts all events caused in measured CMG regardless of measured PE.

0x0396, L2D_CACHE_SWAP_LOCAL

This event counts operations where demand access hits an L2 cache refill buffer allocated by software or hardware prefetch.
It counts all events caused in measured CMG regardless of measured PE.

0x03e0, EA_L2

This event counts energy consumption per cycle of L2 cache.
It counts all events caused in measured CMG regardless of measured PE.

0x03e8, EA_MEMORY

This event counts energy consumption per cycle of CMG local memory.
It counts all events caused in measured CMG regardless of measured PE.

SVE Common Events

0x8000, SIMD_INST_RETIRED

This event counts architecturally executed SIMD instructions, excluding the Advanced SIMD scalar instructions and the instructions listed in Non-SIMD SVE instructions section of SVE Reference Manual.

0x8002, SVE_INST_RETIRED

This event counts architecturally executed SVE instructions, including the instructions listed in Non-SIMD SVE instructions section of SVE Reference Manual.

0x8008, UOP_SPEC

This event counts all architecturally executed micro-operations.

0x800e, SVE_MATH_SPEC

This event counts architecturally executed math function operations due to the SVE FTSMUL, FTMAD, FTSSEL, and FEXPA instructions.

0x8010, FP_SPEC

This event counts architecturally executed operations due to scalar, Advanced SIMD, and SVE instructions listed in Floating-point instructions section of SVE Reference Manual.

0x8028, FP_FMA_SPEC

This event counts architecturally executed floating-point fused multiply-add and multiply-subtract operations.

0x8034, FP_RECPE_SPEC

This event counts architecturally executed floating-point reciprocal estimate operations due to the Advanced SIMD scalar, Advanced SIMD vector, and SVE FRECPE and FRSQRTE instructions.

0x8038, FP_CVT_SPEC

This event counts architecturally executed floating-point convert operations due to the scalar, Advanced SIMD, and SVE floating-point conversion instructions listed in Floating-point conversions section of SVE Reference Manual.

0x8043, ASE_SVE_INT_SPEC

This event counts architecturally executed integer arithmetic operations due to Advanced SIMD and SVE data-processing instructions listed in Integer instructions section of SVE Reference Manual.

0x8074, SVE_PRED_SPEC

This event counts architecturally executed SIMD data-processing and load/store operations due to SVE instructions with a Governing predicate operand that determines the Active elements.

0x807c, SVE_MOVPRFX_SPEC

This event counts architecturally executed operations due to MOVPRFX instructions, whether or not they were fused with the prefixed instruction.

0x807f, SVE_MOVPRFX_U_SPEC

This event counts architecturally executed operations due to MOVPRFX instructions that were not fused with the prefixed instruction.

0x8085, ASE_SVE_LD_SPEC

This event counts architecturally executed operations that read from memory due to SVE and Advanced SIMD load instructions.

0x8086, ASE_SVE_ST_SPEC

This event counts architecturally executed operations that write to memory due to SVE and Advanced SIMD store instructions.

0x8087, PRF_SPEC

This event counts architecturally executed prefetch operations due to scalar PRFM and SVE PRF instructions.

0x8089, BASE_LD_REG_SPEC

This event counts architecturally executed operations that read from memory due to an instruction that loads a general-purpose register.

0x808a, BASE_ST_REG_SPEC

This event counts architecturally executed operations that write to memory due to an instruction that stores a general-purpose register, excluding the "DC ZVA" instruction.

0x8091, SVE_LDR_REG_SPEC

This event counts architecturally executed operations that read from memory due to an SVE LDR instruction.

0x8092, SVE_STR_REG_SPEC

This event counts architecturally executed operations that write to memory due to an SVE STR instruction.

0x8095, SVE_LDR_PREG_SPEC

This event counts architecturally executed operations that read from memory due to an SVE LDR (predicate) instruction.

0x8096, SVE_STR_PREG_SPEC

This event counts architecturally executed operations that write to memory due to an SVE STR (predicate) instruction.

0x809f, SVE_PRF_CONTIG_SPEC

This event counts architecturally executed operations that prefetch memory due to an SVE predicated single contiguous element prefetch instruction.

0x80a5, ASE_SVE_LD_MULTI_SPEC

This event counts architecturally executed operations that read from memory due to SVE and Advanced SIMD multiple vector contiguous structure load instructions.

0x80a6, ASE_SVE_ST_MULTI_SPEC

This event counts architecturally executed operations that write to memory due to SVE and Advanced SIMD multiple vector contiguous structure store instructions.

0x80ad, SVE_LD_GATHER_SPEC

This event counts architecturally executed operations that read from memory due to SVE non-contiguous gather-load instructions.

0x80ae, SVE_ST_SCATTER_SPEC

This event counts architecturally executed operations that write to memory due to SVE non-contiguous scatter-store instructions.

0x80af, SVE_PRF_GATHER_SPEC

This event counts architecturally executed operations that prefetch memory due to SVE non-contiguous gather-prefetch instructions.

0x80bc, SVE_LDFF_SPEC

This event counts architecturally executed memory read operations due to SVE First-fault and Non-fault load instructions.

0x80c0, FP_SCALE_OPS_SPEC

"This event counts architecturally executed SVE arithmetic operations.
See FP_SCALE_OPS_SPEC of SVE Reference Manual for more information.
This event counter is incremented by (128 / CSIZE) and by twice that amount for operations that would also be counted by SVE_FP_FMA_SPEC."

0x80c1, FP_FIXED_OPS_SPEC

"This event counts architecturally executed v8SIMD&FP arithmetic operations.
See FP_FIXED_OPS_SPEC of SVE Reference Manual for more information.
The event counter is incremented by the specified number of elements for Advanced SIMD operations or by 1 for scalar operations, and by twice those amounts for operations that would also be counted by FP_FMA_SPEC."

0x80c2, FP_HP_SCALE_OPS_SPEC

"This event counts architecturally executed SVE half-precision arithmetic operations.
See FP_HP_SCALE_OPS_SPEC of SVE Reference Manual for more information.
This event counter is incremented by 8, or by 16 for operations that would also be counted by SVE_FP_FMA_SPEC."

0x80c3, FP_HP_FIXED_OPS_SPEC

"This event counts architecturally executed v8SIMD&FP half-precision arithmetic operations.
See FP_HP_FIXED_OPS_SPEC of SVE Reference Manual for more information.
This event counter is incremented by the number of 16-bit elements for Advanced SIMD operations, or by 1 for scalar operations, and by twice those amounts for operations that would also be counted by FP_FMA_SPEC."

0x80c4, FP_SP_SCALE_OPS_SPEC

"This event counts architecturally executed SVE single-precision arithmetic operations.
See FP_SP_SCALE_OPS_SPEC of SVE Reference Manual for more information.
This event counter is incremented by 4, or by 8 for operations that would also be counted by SVE_FP_FMA_SPEC."

0x80c5, FP_SP_FIXED_OPS_SPEC

"This event counts architecturally executed v8SIMD&FP single-precision arithmetic operations.
See FP_SP_FIXED_OPS_SPEC of SVE Reference Manual for more information.
This event counter is incremented by the number of 32-bit elements for Advanced SIMD operations, or by 1 for scalar operations, and by twice those amounts for operations that would also be counted by FP_FMA_SPEC."

0x80c6, FP_DP_SCALE_OPS_SPEC

"This event counts architecturally executed SVE double-precision arithmetic operations.
See FP_DP_SCALE_OPS_SPEC of SVE Reference Manual for more information.
This event counter is incremented by 2, or by 4 for operations that would also be counted by SVE_FP_FMA_SPEC."

0x80c7, FP_DP_FIXED_OPS_SPEC

"This event counts architecturally executed v8SIMD&FP double-precision arithmetic operations.
See FP_DP_FIXED_OPS_SPEC of SVE Reference Manual for more information.
This event counter is incremented by 2 for Advanced SIMD operations, or by 1 for scalar operations, and by twice those amounts for operations that would also be counted by FP_FMA_SPEC."