Cortex-A57 Implementation Defined PMU Events

Processor Division

Document number:	ARM-EPM-042751	1.0
Date of Issue:	4 June 2013	
Author:	ARM	

Abstract

This document defines the implementation defined PMU Events in the Cortex-A57 processor.

Keywords

Cortex-A57, PMU

Reviewer list

Name	Function	Name	Function

Contents

1 ABOUT THIS DOCUMENT 3
1.1 Change control 3
1.2 References 3
1.3 Terms and abbreviations 3

2 SCOPE 4

3 INTRODUCTION 4

4 IMPLEMENTATION DEFINED EVENTS 4

1 ABOUT THIS DOCUMENT

1.1 Change control

Issue	Date	Change
0.1	4 June 2013	Initial version
1.0	28 May 2013	Cortex-A57 r0p0 LAC release

1.2 References

This document refers to the following documents.

Ref	Doc No	Author(s)	Title
1	DDI 0488	ARM	Cortex-A57 Technical Reference Manual
2	DDI 0406C	ARM	ARM v7-A Architecture Reference Manual
3	PRD03-PRDC-010486	ARM	ARM v8 Debug Architecture

1.3 Terms and abbreviations

This document uses the following terms and abbreviations.

Term	Meaning
ACP	Accelerator Coherency Port
AXI	Advanced eXtensible Interface
BTB	Branch Target Buffer
ECC	Error Correcting Code
IPA	Intermediate Physical Address
PA	Physical Address
PMU	Performance Monitoring Unit
ETM	Embedded Trace Module
VFP	Vector Floating Point
SIMD	Single Instruction Multiple Data
TLB	Translation Look aside Buffer

2 SCOPE

This document is the specification for implementation defined PMU events in Cortex-A57 processor.

3 INTRODUCTION

The Cortex-A57 processor includes logic to gather various statistics on the operation of the processor and memory system during runtime, based on PMUv3 architecture. These events provide useful information about the behavior of the processor that you can use when debugging or profiling code.

The processor PMU provides six counters. Each counter can count any of the events available in the processor. The absolute counts recorded might vary because of pipeline effects. This has negligible effect except in cases where the counters are enabled for a very short time. For more information see *Cortex-A57 Technical Reference Manual*.

4 IMPLEMENTATION DEFINED EVENTS

The list of common architectural and micro architectural PMU events implemented in Cortex-A57 is described in *Cortex-A57 Technical Reference Manual*.

Following Table 4-1 shows the implementation defined events that are generated in Cortex-A57 and the numbers that the PMU uses to reference the events. Event reference numbers that are not listed are reserved.

Table 4-1 PMU Implementation defined events

Event Number	Event Description
0x100	Number of micro BTB or BTB predicted branches
0x101	Number of predictable branches predicted by static predictor
0x102	Number of predictable branches predicted by BTB/micro BTB that are indirect
0x103	Number of micro BTB/BTB predicted conditional branches
0x104	Number of predictable branches that are conditional
0x105	Number of predictable branches that are taken
0x106	Number of directional mispredicted micro BTB/BTB predicted branches
0x107	Number of mispredicted branches predicted by BTB/micro BTB
0x108	Number of mispredicted indirect branch predicted by BTB/micro BTB
0x109	Mispredicted procedure returns
0x10A	Number of mispredicted indirect branches
0x10B	Number of unpredicted not taken branches
0x10C	Increment every cycle that the fetch queue contains no instructions
0x110	Increment every cycle when single micro operation is decoded/sequenced
0x111	Increment every cycle that the rename queue contains no micro operations
0x112	Increment every cycle when rename stalls due to lack of flags

Event Number	Event Description
0x113	Increment every cycle when rename stalls due to lack of registers
0x118	Count number of micro operations renamed
0x140	Count groups
0x141	Count ETM EXTOUT 0
0x142	Count ETM EXTOUT 1
0x143	Count ETM EXTOUT 2
0x144	Count ETM EXTOUT 3
0x150	Load store load restart count
0x120	Increment every cycle that the dispatch queue contains no micro operations
0x121	Dispatch stall due to issue queue full branch cluster
0x122	Dispatch stall due to issue queue full complex cluster1 or complex cluster 2
0x123	Dispatch stall due to issue queue full load store cluster
0x124	Dispatch stall due to issue queue full multi cycle cluster
0x125	Dispatch stall due to issue queue full special purpose cluster
0x126	Dispatch stall due to issue queue full single cycle cluster 1 or single cycle cluster 2
0x127	Flush caused due to the branch predictor incorrectly generating a prediction on an instruction that was decoded to not be a branch
0x128	Flushes triggered for micro architectural reasons from the Level 1 memory system. Three possible causes: 1) ECC error (single or double bit) in the Level 1 data cache, pipe flush while load store unit deals with the error. 2) The load store unit has speculatively completed two memory requests out of order and has determined that the younger load has potentially returned older data than the older load. This is an ARM memory ordering violation and the machine must be flushed. 3) Memory operation issued to device/strongly ordered memory with an extension register as the data source/destination. This causes a pipe flush and re-issue with different grouping rules to avoid a deadlock case.
0x129	Count flush events 0x127, 0x128, 0x12A and 0x12B
0x12A	Flush triggered when single word producer is followed by a double word consumer of that single word and the double word is not in the architecture register file.
0x12B	Flush caused due to incorrect prediction of advanced SIMD execution within an IT block.
0x12C	Dispatch stall due to single word producer double word consumer hazard
0x151	Load pipe hazard in Load Store unit
0x152	Store pipe hazard in Load Store unit
0x160	Bus read transaction
0x161	Bus write transaction
0x162	Bus access - snoop
0x163	Bus transaction - prefetches
0x164	ACP based AXI master port read data
0x165	ACP based AXI master port write data

Event Number	Event Description
0x166	Level 2 prefetcher request generated
0x167	Level 2 prefetcher request used (or demanded)
0x168	Level 2 prefetcher request issued
0x169	Level 2 prefetcher request pending (when demanded)
0x16A	Level 2 stalled due to no available entry in fill/evict queue
0x16B	Level 2 Cache Coherency Bus (CCB) transfer from another cpu's Level 1 data cache to this cpu
0x170	Level 2 Table walk descriptor read access
0x171	Level 2 IPA-PA cache access
0x172	Level 2 IPA-PA cache hit
0x173	Level 2 Stage 1 Level 2 PA cache hit
0x174	Level 2 Stage 1 walk cache hit
0x175	Level 2 TLB accesses made to the array during lookup
0x176	Level 2 TLB shootdowns
0x177	Level 2 TLB accesses for lookups
0x178	Level 2 TLB accesses for table prefetches
0x179	Level 2 TLB refills
0x17A	Level 2 TLB refills for instruction fetches
0x17B	Level 2 TLB refills for loads and stores
0x17C	Level 2 TLB refills for table walk prefetches