PA-RISC 2.0 Architecture

This document is an excerpt from PA-RISC 2.0 ARCHITECTURE, by Gerry Kane, published by Prentice Hall PTR, isbn 0-13-182734-0. Print copies of the book can be ordered at www.hp.com/hpbooks.

This document contains:

Index
A

absolute accesses 3-1, G-2
absolute_address() E-2
access ID 3-13
access rights 3-13
access type
execute 3-12
read 3-12
write 3-12
ADD 7-2
ADD (see ADD)
ADD AND BRANCH 7-4
ADD IMMEDIATE AND BRANCH 7-6
ADD IMMEDIATE LEFT 7-7
ADD TO IMMEDIATE 7-5
ADDB (see ADD AND BRANCH)
ADDI (see ADD TO IMMEDIATE)
ADDIB (see ADD IMMEDIATE AND BRANCH)
ADDIL (see ADD IMMEDIATE LEFT)
address aliasing F-5
   equivalent aliases F-6
   many-reader/one-writer F-6
   non-equivalent aliases F-6
   read-only F-6
AND 7-8
AND (see AND)
AND COMPLEMENT 7-9
ANDCM (see AND COMPLEMENT)
assemble_12() E-1
assemble_16() E-1
assemble_16a() E-1
assemble_17() E-1
assemble_21() E-1
assemble_22() E-1
assemble_3() E-1
assemble_6() E-1
assist emulation trap 5-12
assist exception trap 5-9
assist processor 1-13

atomicity G-1

B

B (see BRANCH)
BB (see BRANCH ON BIT)
BCD, (see binary coded decimal)
BE (see BRANCH EXTERNAL)
   binary coded decimal 7-35
   binary coded decimal data type 2-19
   bit data type 2-18
   bits
      ignored 2-17
      nonexistent 2-17
      reserved 2-17
      undefined 2-17
   Block Copy cache control hint 6-10
   BLR (see BRANCH AND LINK REGISTER)
   BRANCH 2-8, 6-14, 7-10
   BRANCH AND LINK REGISTER 6-14, 7-16
   BRANCH EXTERNAL 6-15, 7-15
   BRANCH ON BIT 7-13
   BRANCH VECTORED 6-14, 7-18
   BRANCH VECTORED EXTERNAL 6-15, 7-19
   branches 4-1
      base relative 4-2
      conditional 4-2, 6-15
      delay slot 4-1
      delayed 4-1
      dynamic displacement 4-2
      external 6-15
      IA relative 4-2
      interspace 6-15
      intraspace 6-14
      local 6-14
      not-taken 4-2
      static displacement 4-2
      taken 4-2
      unconditional 4-2, 6-14
   BREAK 7-17
BREAK instruction trap 5-8
BV (see BRANCH VECTORED)
BVE (see BRANCH VECTORED EXTERNAL)
byte data type 2-18

C

cache
  clean 3-17
  dirty 3-17
  move-in F-8

cache control hints 6-9
  Block Copy 6-10
  Coherent Operation 6-11
  load 6-10
  semaphore 6-10
  Spacial Locality 6-10
  store 6-10

CALL 7-11
cat() E-1
CCR (see Coprocessor Configuration Register)
check 5-1

CLDD (see COPROCESSOR LOAD DOUBLE-WORD)

CLDW (see COPROCESSOR LOAD WORD)
CLEAR BRANCH TARGET STACK 7-25
CLRBTs (see CLEAR BRANCH TARGET STACK)
CMP (see COMPARE AND BRANCH)
CMPCLR (see COMPARE AND CLEAR)
CMPIB (see COMPARE IMMEDIATE AND BRANCH)
CMPICLR (see COMPARE IMMEDIATE AND CLEAR)

coherence_index() E-2
coherent input/output F-11
Coherent Operation cache control hint 6-11
coherent_system E-2

COMPARE AND BRANCH 7-26
COMPARE AND CLEAR 7-27

COMPARE IMMEDIATE AND BRANCH 7-28
COMPARE IMMEDIATE AND CLEAR 7-29
conditional trap 5-9

conditions
  arithmetic/logical D-1
  floating-point compare 8-16
  floating-point test 8-17
  shift/extract/deposit D-8

unit D-7
control registers 2-10

COPR (see COPROCESSOR OPERATION)

COPR,0,0 (see IDENTIFY COPROCESSOR)
coprocessor 1-13, 6-22
Coprocessor Configuration Register 2-12, 6-22
COPROCESSOR LOAD DOUBLEWORD 7-21
COPROCESSOR LOAD WORD 7-23
COPROCESSOR OPERATION 7-30
COPROCESSOR STORE DOUBLEWORD 7-31
COPROCESSOR STORE WORD 7-33

coprocessor_condition() E-2
coprocessor_op() E-2

COPY 7-82

CRs (see control registers)
CSTD (see COPROCESSOR STORE DOUBLEWORD)

CSTW (see COPROCESSOR STORE WORD)

D
data memory access rights trap 5-10
data memory break trap 5-11
data memory protection ID trap 5-10
data memory protection trap/unaligned data reference trap 5-11
data TLB miss fault/data page fault 5-9
data types
  binary coded decimal 2-19
  bit 2-18
  byte 2-18
  double-precision floating-point 2-19, 8-6
  integer 2-18
  quad-precision floating-point 8-6
  signed halfword 2-18
  signed word 2-18
  single-precision floating-point 2-19, 8-6
  unsigned halfword 2-18
  unsigned word 2-19

Dcache_flush() E-3
Dcache_flush_entries() E-3
Dcache_flush_or_purge() E-3

DCOR (see DECIMAL CORRECT)

DECIMAL CORRECT 7-35

DEPD (see DEPOSIT DOUBLEWORD)

DEPDI (see DEPOSIT DOUBLEWORD IMMEDIATE)
DEPOSIT DOUBLEWORD  7-37
DEPOSIT DOUBLEWORD IMMEDIATE  7-39
DEPOSIT WORD  7-41
DEPOSIT WORD IMMEDIATE  7-43
DEPW (see DEPOSIT WORD)
DEPWI (see DEPOSIT WORD IMMEDIATE)
DIAG (see DIAGNOSE)
DIAGNOSE  7-45
Direct I/O  1-13
Direct Memory Access I/O  1-13
DIVIDE STEP  2-8, 7-46
double-precision floating-point data type  2-19
DS (see DIVIDE STEP)
DTLB_alloc() E-3
DTLB_purge_broadcast() E-3
DTLB_purge_entries() E-3
DTLB_purge_local() E-3
DTLB_search() E-3

E
Eiem (see External Interrupt Enable Mask)
EIRR (see External Interrupt Request Register)
endian bit  2-19
equivalent aliases F-6
equivalently-mapped F-5
excepting instruction  10-1
EXCLUSIVE OR  7-154
external interrupt  5-6
External Interrupt Enable Mask  2-13
External Interrupt Request Register  2-16
EXTRACT DOUBLEWORD  7-47
EXTRACT WORD  7-49
EXTRD (see EXTRACT DOUBLEWORD)
EXTRW (see EXTRACT WORD)

F
FABS (see FLOATING-POINT ABSOLUTE VALUE)
FADD (see FLOATING-POINT ADD)
fault  5-1
FCMP (see FLOATING-POINT COMPARE)
FCNV (see FLOATING-POINT CONVERT)
FCPY (see FLOATING-POINT COPY)
FDC (see FLUSH DATA CACHE)
FDCX (see FLUSH DATA CACHE ENTRY)
FDIV (see FLOATING-POINT DIVIDE)

PA-RISC 2.0 Architecture

INDEX
CQ field 8-10
D-bit 8-10
Enables field 8-9
Flags field 8-9
I bits 8-10
model field 8-11
O bits 8-10
revision field 8-11
RM field 8-9
T-bit 8-10
U bits 8-10
V bits 8-10
Z bits 8-10
FLOATING-POINT STORE DOUBLEWORD 9-23
FLOATING-POINT STORE WORD 9-25
FLOATING-POINT SUBTRACT 9-27
FLOATING-POINT TEST 9-28
FLUSH DATA CACHE 7-51
FLUSH DATA CACHE ENTRY 7-53
FLUSH INSTRUCTION CACHE 7-54
FLUSH INSTRUCTION CACHE ENTRY 7-56
FMPY (see FLOATING-POINT MULTIPLY)
FMPYADD (see FLOATING-POINT MULTIPLY/ADD)
FMPYFADD (see FLOATING-POINT MULTIPLY FUSED ADD)
FMPYNFADD (see FLOATING-POINT MULTIPLY NEGATE FUSED ADD)
FMPYSUB (see FLOATING-POINT MULTIPLY/SUBTRACT)
FNEG (see FLOATING-POINT NEGATE)
FNEGABS (see FLOATING-POINT NEGATE ABSOLUTE VALUE)
FPSR (see Floating-Point Status Register)
FRND (see FLOATING-POINT ROUND TO INTEGRER)
FSQRT (see FLOATING-POINT SQUARE ROOT)
FSTD (see FLOATING-POINT STORE DOUBLEWORD)
FSTW (see FLOATING-POINT STORE WORD)
FSUB (see FLOATING-POINT SUBTRACT)
FTEST (see FLOATING-POINT TEST)

H
HADD (see HALFWORD PARALLEL ADD)
HALFWORD PARALLEL ADD 7-57
HALFWORD PARALLEL AVERAGE 7-58
HALFWORD PARALLEL SHIFT LEFT 7-59
HALFWORD PARALLEL SHIFT LEFT AND ADD 7-60
HALFWORD PARALLEL SHIFT RIGHT 7-61
HALFWORD PARALLEL SHIFT RIGHT AND ADD 7-59, 7-62
HALFWORD PARALLEL SUBTRACT 7-63
HAVG (see HALFWORD PARALLEL AVERAGE)
higher-privilege transfer trap 5-13
high-priority machine check 5-5
HSHL (see HALFWORD PARALLEL SHIFT LEFT)
HSHLADD (see HALFWORD PARALLEL SHIFT LEFT AND ADD)
HSHR (see HALFWORD PARALLEL SHIFT RIGHT)
HSHRAD (see HALFWORD PARALLEL SHIFT RIGHT AND ADD)
HSUB (see HALFWORD PARALLEL SUBTRACT)

I
I/O (see input/output)
IAOO (see Instruction Address Offset Queue)
IAQs (see Instruction Address Queues)
IASQ (see Instruction Address Space Queue)
Icache_flush() E-3
Icache_flush_entries() E-4
IDENTIFY COPROCESSOR 6-22, 7-30
IDENTIFY SFU 6-21, 7-125
IDTLBT (see INSERT DATA TLB TRANSFORMATION)
IEEE 754 8-1
ignored
bits 2-17
IIAQQ (see Interruption Instruction Address Offset Queue)
IIAQs (see Interruption Instruction Address Queues)
IIASQ (see Interruption Instruction Address Space Queue)
IIR (see Interruption Instruction Register)
IITLBT (see INSERT INSTRUCTION TLB TRANSFORMATION)
illegal instruction 6-24
illegal instruction trap 5-8
INCLUSIVE OR 7-101
INDIRECT CALL 7-20
input/output 1-12
INSERT DATA TLB TRANSLATION 7-64
INSERT INSTRUCTION TLB TRANSLATION 7-66
Instruction Address Offset Queue 2-4
Instruction Address Queues 2-4
Instruction Address Space Queue 2-4
instruction memory protection trap 5-7
instruction TLB miss fault/instruction page fault 5-7
instructions
immediate 6-12
integer data type 2-18
interrupt 5-1
Interuption Instruction Address Offset Queue 2-13
Interuption Instruction Address Queues 2-13
Interuption Instruction Address Space Queue 2-13
Interuption Instruction Register 2-15
Interuption Offset Register 2-15
Interuption ParameterRegisters 2-15
Interuption Processor Status Word 2-16
Interuption Space Register 2-15
Interuption Vector Address 2-13
interruptions
disabling 5-4
floating-point 10-4
group 1 5-5
group 2 4-7, 5-6
group 3 4-8, 5-7
group 4 4-9, 5-13
masking 5-4
performance monitor 11-1
priorities 5-4
Interval Timer 2-5
IO_EIR 2-17
IOR (see Interruption Offset Register)
IPRs (see Interruption Parameter Registers)
IPSW (see Interruption Processor Status Word)
ISR (see Interruption Space Register)
ITLB_alloc() E-4
ITLB_purge_local() E-4
ITLB_purge_broadcast() E-4
ITLB_purge_entries() E-4
IVA (see Interruption Vector Address)

L
LCI (see LOAD COHERENCE INDEX)
LDB (see LOAD BYTE)
LDCHD (see LOAD AND CLEAR DOUBLEWORD)
LDCW (see LOAD AND CLEAR WORD)
LDD (see LOAD DOUBLEWORD)
LDDA (see LOAD DOUBLEWORD ABSOLUTE)
LDH (see LOAD HALFWORD)
LDI (see LOAD IMMEDIATE)
LDIL (see LOAD IMMEDIATE LEFT)
LDO (see LOAD OFFSET)
LDSID (see LOAD SPACE IDENTIFIER)
LDW (see LOAD WORD)
LDWA (see LOAD WORD ABSOLUTE)
LOAD AND CLEAR DOUBLEWORD 7-71
LOAD AND CLEAR WORD 7-73
LOAD BYTE 7-69
LOAD COHERENCE INDEX 7-68
LOAD DOUBLEWORD 7-75
LOAD DOUBLEWORD ABSOLUTE 7-77
LOAD HALFWORD 7-79
LOAD IMMEDIATE 7-82
LOAD IMMEDIATE LEFT 7-81
LOAD OFFSET 7-82
LOAD PHYSICAL ADDRESS 7-88
LOAD SPACE IDENTIFIER 7-83
LOAD WORD 7-84
LOAD WORD ABSOLUTE 7-86
low_sign_ext() E-1
lower-privilege transfer trap 5-13
low-priority machine check 5-7
LPA (see LOAD PHYSICAL ADDRESS)
lshift() E-1

M
many-reader/one-writer aliasing F-6
mask
  system 2-7
measurement_enabled E-4
mem_load() 6-7
mem_store() 6-7

PA-RISC 2.0 Architecture

Index IN-5
memory-mapped input/output 1-12
MFCTL (see MOVE FROM CONTROL REGISTER)
MFIA (see MOVE FROM INSTRUCTION ADDRESS)
MFSP (see MOVE FROM SPACE REGISTER)
MIX HALFWORDS 7-92
MIX WORDS 7-93
MIXH (see MIX HALFWORDS)
MIXW (see MIX WORDS)
MOVB (see MOVE AND BRANCH)
MOVE AND BRANCH 7-94
MOVE FROM CONTROL REGISTER 7-89
MOVE FROM INSTRUCTION ADDRESS 7-90
MOVE FROM SPACE REGISTER 7-91
MOVE IMMEDIATE AND BRANCH 7-95
MOVE TO CONTROL REGISTER 7-96
MOVE TO SHIFT AMOUNT REGISTER 7-96
MOVE TO SPACE REGISTER 7-100
MOVE TO SYSTEM MASK 7-99
move-in
data cache F-8
instruction cache F-9
MOVIB (see MOVE IMMEDIATE AND BRANCH)
MTCTL (see MOVE TO CONTROL REGISTER)
MTSAR (see MOVE TO SHIFT AMOUNT REGISTER)
MTSARCM (see MOVE TO SHIFT AMOUNT REGISTER COMPLEMENT)
MTSM (see MOVE TO SYSTEM MASK)
MTSP (see MOVE TO SPACE REGISTER)
Multimedia Instructions 6-3
multiprocessor systems F-12

O
observed G-1
operation undefined 6-24
OR (see INCLUSIVE OR)
ordered G-1
ordering G-1
overflow
signed D-2
unsigned D-2
overflow trap 5-9

P
page
  cacheable 3-17
  uncacheable 3-17
page reference trap 5-12
page table 3-15
PDC (see PURGE DATA CACHE)
PDTLB (see PURGE DATA TLB)
PDTLBE (see PURGE DATA TLB ENTRY)
performance monitor
  interruptions 11-1
PERFORMANCE MONITOR DISABLE 11-3
PERFORMANCE MONITOR ENABLE 11-4
performance monitor interrupt 5-7
performed G-1
PERMH (see PERMUTE HALFWORDS)
PERMUTE HALFWORDS 7-106
phys_mem_load() E-4
phys_mem_store() E-5
PIDs (see Protection Identifiers)
PITLB (see PURGE INSTRUCTION TLB)
PITLBE (see PURGE INSTRUCTION TLB ENTRY)
PMDIS (see PERFORMANCE MONITOR DISABLE)
PMENB (see PERFORMANCE MONITOR ENABLE)
POP BRANCH TARGET STACK 7-110
POPBTS (see POP BRANCH TARGET STACK)
power failure interrupt 5-6
Privileged Software-Accessible Registers 2-7
privilege level 3-12
  changing 4-5, I-1
privileged operation trap 5-8

N
NaN (see floating-point, Not a Number)
NO OPERATION 7-101
non-access data TLB miss fault/non-access data page fault 5-10
non-access instruction TLB miss fault 5-9
non-equivalent aliases F-6
nonexistent bits 2-17
NOP (see NO OPERATION)
null instructions 6-25
nullification 4-7
privileged register trap 5-8
PROBE ACCESS 7-111
PROBE ACCESS IMMEDIATE 7-112
PROBE(see PROBE ACCESS)
PROBEI (see PROBE ACCESS IMMEDIATE)
Processor 2-7
Processor Status Word 2-7
B-bit 2-8, 4-7, 4-9
C/B bits 2-9
C-bit 2-8
D-bit 2-9
E-bit 2-8, 2-19, 5-2
F-bit 2-9
H-bit 2-8, 4-9
I-bit 2-9
L-bit 2-8, 4-9
M-bit 2-8, 4-7, 5-3
N-bit 2-8, 4-7, 4-9
O-bit 2-9
P-bit 2-9
Q-bit 2-9
R-bit 2-9, 2-11, 4-9
S-bit 2-8
T-bit 2-8
V-bit 2-8
W-bit 5-2
X-bit 2-8, 4-7, 4-9
Protection Identifiers 2-12, 3-12
PSW (see Processor Status Word)
PURGE DATA CACHE 7-102
PURGE DATA TLB 7-104
PURGE DATA TLB ENTRY 7-105
PURGE INSTRUCTION TLB ENTRY 7-109
PUSH BRANCH TARGET STACK 7-113
PUSH NOMINATED 7-114
PUSHBTS (see PUSH BRANCH TARGET STACK)
PUSHNOM (see PUSH NOMINATED)

R
read_access_allowed() E-5
read-only aliasing F-6
read-only translation F-6
Recovery Counter 2-11, 4-9
recovery counter trap 5-6
registers
control 2-10
floating-point 8-2
floating-point exception 10-1
general 2-2
reserved 2-17
shadow 2-9
space 2-3
relied-upon translation F-2
reserved
bits 2-17
instruction field values 6-25
instruction fields 6-24
registers 2-17
RESET SYSTEM MASK 7-117
RET (see RETURN)
RETURN 7-20
RETURN FROM INTERRUPTION 5-4, 7-115
RFI (see RETURN FROM INTERRUPTION)
rounding, floating-point 8-22
rshift() E-2
RSM (see RESET SYSTEM MASK)

S
SAR (see Shift Amount Register)
SCR (see SFU Configuration Register)
send_to_copr() E-2
SET SYSTEM MASK 7-128
SFU Configuration Register 2-12, 6-21
SFU Registers 2-6
sfu_condition0() E-5
sfu_condition1() E-5
sfu_condition2() E-5
sfu_condition3() E-5
sfu_operation0() E-5
sfu_operation1() E-5
sfu_operation2() E-5
sfu_operation3() E-5
shadow registers 2-9, 7-115
Shift Amount Register 2-5
SHIFT LEFT AND ADD 7-118
SHIFT LEFT DOUBLEWORD 7-38
SHIFT LEFT WORD 7-42
SHIFT RIGHT DOUBLEWORD 7-48
SHIFT RIGHT PAIR DOUBLEWORD 7-120
SHIFT RIGHT PAIR WORD 7-122
SHIFT RIGHT WORD 7-50
SHLADD (see SHIFT LEFT AND ADD)
SHLD (see SHIFT LEFT DOUBLEWORD)
SHLW (see SHIFT LEFT WORD)
SHRD (see SHIFT RIGHT DOUBLEWORD)
SHRPD (see SHIFT RIGHT PAIR DOUBLEWORD)
SHRPW (see SHIFT RIGHT PAIR WORD)
SHRs (see shadow registers)
SHRw (see SHIFT RIGHT WORD)
sign_ext() E-2
sign_ext_16() E-2
signed halfword data type 2-18
signed overflow D-2
signed word data type 2-18
single-precision floating-point data type 2-19
space registers 2-3
space_select() E-6
Spacial Locality cache control hint 6-10
special function unit 1-13, 6-20
SPECIAL OPERATION ONE 7-125
SPECIAL OPERATION THREE 7-127
SPECIAL OPERATION TWO 7-126
SPECIAL OPERATION ZERO 7-124
SPOP0 (see SPECIAL OPERATION ZERO)
SPOP1 (see SPECIAL OPERATION ONE)
SPOP1, sfu, 0 (see IDENTIFY SFU)
SPOP2 (see SPECIAL OPERATION TWO)
SPOP3 (see SPECIAL OPERATION THREE)
SRs (see space registers)
SSM (see SET SYSTEM MASK)
STB (see STORE BYTE)
STBY (see STORE BYTES)
STD (see STORE DOUBLEWORD)
STD A (see STORE DOUBLEWORD ABSOLUTE)
STD BY (see STORE DOUBLEWORD BYTES)
STH (see STORE HALFWORD)
STORE BYTE 7-129
STORE BYTES 7-131
STORE BYTES SHORT 6-12, G-1, H-6
STORE DOUBLEWORD 7-134
STORE DOUBLEWORD ABSOLUTE 7-136
STORE DOUBLEWORD BYTES 7-137
STORE HALFWORD 7-140
STORE WORD 7-142
STORE WORD ABSOLUTE 7-144
store_in_memory() E-2
strongly ordered G-1
STW (see STORE WORD)

STWA (see STORE WORD ABSOLUTE)
SUB (see SUBTRACT)
SUBI (see SUBTRACT FROM IMMEDIATE)
SUBTRACT 7-146
SUBTRACT FROM IMMEDIATE 7-148
SYNC (see SYNCHRONIZE CACHES)
SYNCDMA (see SYNCHRONIZE DMA)
SYNCHRONIZE CACHES 7-149
SYNCHRONIZE DMA 7-150
system mask 2-7

T
taken branch trap 5-13
temporary registers 2-17
TLB (see Translation Lookaside Buffer)
TLB dirty bit trap 5-12
translation
read-only F-6
write-capable F-6
Translation Lookaside Buffer 3-9
access identifier (ID) 3-13
access rights 3-13
B-bit 3-10
combined 3-9
D-bit 3-10
taken branch trap 5-13
temporary registers 2-17
TLB (see Translation Lookaside Buffer)
TLB dirty bit trap 5-12
translation
read-only F-6
write-capable F-6
Translation Lookaside Buffer 3-9
access identifier (ID) 3-13
access rights 3-13
B-bit 3-10
combined 3-9
D-bit 3-10
entry F-1
hardware miss handling F-3
O-bit 3-10
relied-upon translation F-2
slot F-1
software miss handling F-2
T-bit 3-10
U-bit 3-10
U

UADDCM (see UNIT ADD COMPLEMENT)
unaligned data reference trap 5-11
undefined
bits 2-17
instruction 6-24
undefined operation 6-24
UNIT ADD COMPLEMENT 7-151
UNIT XOR 7-153
unsigned halfword data type    2-18
unsigned overflow    D-2
unsigned word data type    2-19
UXOR (see UNIT XOR)

V
virt_mem_load()    E-6
virt_mem_store()    E-7
virtual accesses    3-1

W
WD bit    2-12
weakly ordered    G-1
write_access_allowed()    E-7
write-capable translation    F-6

X
XMPYU (see FIXED-POINT MULTIPLY UNSIGNED)
XOR (see EXCLUSIVE OR)
xor()    E-2

Z
zero_ext()    E-2