rtfItanium

Bundle Format

Bundle Format:

|  |  |  |  |
| --- | --- | --- | --- |
| 127 120 | 119 80 | 79 40 | 39 0 |
| Template8 | Slot2 | Slot1 | Slot0 |

Instruction Formats:

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Immediate7 | | | | | A | R | Immed15 | | | | Rs16 | Opcode4 | Rd6 | | | | ML |
| Immediate7 | | | | | A | R | Immed9 | | Rs26 | | Rs16 | Opcode4 | Immed6 | | | | MS |
| Funct5 | | | | Immed2 | A | R | Sc3 | Rs36 | Immed6 | | Rs16 | Opcode4 | Rd6 | | | | MLX |
| Funct5 | | | | Immed2 | A | R | Sc3 | Rs36 | Rs26 | | Rs16 | Opcode4 | Immed6 | | | | MSX |
| Immediate7 | | | | | Op2 | | Immed15 | | | | Rs16 | Opcode4 | Rd6 | | | | RI |
| Funct5 | | | | Funct22 | Op2 | | Sz3 | Rs36 | Rs26 | | Rs16 | Opcode4 | Rd6 | | | | R3 |
| Funct5 | | | | Rg2 | o | w | ~3 | Rs36 | Bw6 | | Bo6 | Opcode4 | Rd6 | | | | BF |
| Funct5 | | | | Rg2 | o | Bw4 | | Rs36 | Rs26 | | Bo6 | Opcode4 | Rd6 | | | | BFI |
| Td21..13 | | | | | | | To12..4 | | Rs26 | | Rs16 | Opcode4 | To3..1 | | Cond3 | | Bcc |
| Td21..13 | | | | | | | To12..4 | | Bitno6..1 | | Rs16 | Opcode4 | To3..1 | | B0 | Cn2 | BBS |
| Td21..13 | | | | | | | To12..4 | | Immed6 | | Rs16 | Opcode4 | To3..1 | | Imm3 | | BEQI |
| ~9 | | | | | | | ~3 | Rs36 | Rs26 | | Rs16 | Opcode4 | ~2 | Cond4 | | | BRg |
| Address30 | | | | | | | | | | | | Opcode4 | Address6 | | | | CALL |
| Immediate18 | | | | | | | | | 616 | | 636 | Opcode4 | 636 | | | | RET |
| H | ~11 | | | | | | | ~6 | Cause6 | | Rs16 | Opcode4 | C2 | Imask4 | | | BRK |
| ~12 | | | | | | | | ~6 | ~6 | | Rs16 | Opcode4 | Sema6 | | | | RTI |
| Immediate7 | | | | | ~ | ~ | Immed9 | | Rs26 | | Rs16 | Opcode4 | Immed6 | | | | CHKI |
| ~12 | | | | | | | | Rs36 | Rs26 | | Rs16 | Opcode4 | ~6 | | | | CHK |
| Funct5 | | | | Prec4 | | | Rm3 | Rs36 | Rs26 | | Rs16 | Opcode4 | Rd6 | | | | FLT |
| Op2 | | OL2 | ~5 | | | | ~3 | Regno12 | | | Rs16 | Opcode4 | Rd6 | | | | CSR |
| Funct5 | | | | Cmd4 | | | ~11 | | | Tn4 | Rs16 | Opcode4 | Rd6 | | | | TLB |

|  |  |  |
| --- | --- | --- |
| Unit |  |  |
| 1 | Branch |  |
| 2 | Integer |  |
| 3 | Floating Point |  |
| 4 | Memory Load |  |
| 5 | Memory Store |  |
| 6 |  |  |
| 7 |  |  |

Memory Loads

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | xx11 | xx10 | xx01 | xx00 |
| 00xx | LDB | LDC | LDP | LDD |
| 01xx | LDBU | LDCU | LDPU | LDDR |
| 10xx | LDT | LDO |  |  |
| 11xx | LDTU |  | LEA | {MLX} |

Memory Stores

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | xx11 | xx10 | xx01 | xx00 |
| 00xx | STB | STC | STP | STD |
| 01xx |  |  |  | STDC |
| 10xx | STT | STO | CAS | PUSHC |
| 11xx |  | PUSH | CACHE | {MSX} |

Flow Control

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | xx11 | xx10 | xx01 | xx00 |
| 00xx | Bcc | BLcc | BRcc | NOP |
| 01xx | FBcc |  | BEQ # | BNE # |
| 10xx | JAL | JMP | CALL | RET |
| 11xx | CHK # | CHK | RTI | BRK |

ALU

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | x0 | x1 | x2 | x3 | x4 | x5 | x6 | x7 | x8 | x9 | xA | xB | xC | xD | xE | xF |
| 0x |  |  | {R3} | {R3} | ADD |  | CMP | CMPU | AND | OR | XOR |  | BLEND |  |  |  |
| 1x | SLT | SGE | SLE | SGT | SLTU | SGEU | SLEU | SGTU | SEQ | SNE |  |  |  |  |  |  |
| 2x | MUL | MULU | DIV | DIVU | MOD | MODU |  |  | MADF |  |  |  |  |  |  |  |
| 3x | FXMUL |  |  |  | ADDS1 | ADDS2 | ADDS3 |  | ANDS1 | ANDS2 | ANDS3 |  | ORS1 | ORS2 | ORS3 |  |

ALU – R3 Format

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | x0 | x1 | x2 | x3 | x4 | x5 | x6 | x7 | x8 | x9 | xA | xB | xC | xD | xE | xF |
| 0x | MULH | MULUH | ADDV | SUBV | ADD | SUB | CMP | CMPU | AND | OR | XOR |  | NAND | NOR | XNOR |  |
| 1x | SLT | SGE | SLE | SGT | SLTU | SGEU | SLEU | SGTU | SEQ | SNE | MOV | CMOVNZ | MIN | MAX | PTRDIF |  |
| 2x | MUL | MULU | DIV | DIVU | MOD | MODU |  |  | MADF | MUX |  |  |  |  |  |  |
| 3x | FXMUL | FXDIV | SHL | ASL | SHR | ASR | ROL | ROR | SHL # | ASL # | SHR # | ASR # | ROL # | ROR # | BMM |  |

|  |  |  |  |
| --- | --- | --- | --- |
| Template | Slot0 | Slot1 | Slot2 |
| 00 | Mem Unit | Int Unit | Int Unit |
| 01 | Mem Unit | Int Unit | Int Unit |
| 02 | Mem Unit | Int Unit | Int Unit |
| 03 | Mem Unit | Int Unit | Int Unit |
| 04 | Mem Unit |  |  |
| 05 | Mem Unit |  |  |
| 06 |  |  |  |
| 07 |  |  |  |
| 08 | Mem Unit | Mem Unit | Int Unit |
| 09 | Mem Unit | Mem Unit | Int Unit |
| 0A | Mem Unit | Mem Unit | Int Unit |
| 0B | Mem Unit | Mem Unit | Int Unit |
| 0C | Mem Unit | FP Unit | Int Unit |
| 0D | Mem Unit | FP Unit | Int Unit |
| 0E | Mem Unit | Mem Unit | FP Unit |
| 0F | Mem Unit | Mem Unit | FP Unit |
| 10 | Mem Unit | Int Unit | Branch Unit |
| 11 | Mem Unit | Int Unit | Branch Unit |
| 12 | Mem Unit | Branch Unit | Branch Unit |
| 13 | Mem Unit | Branch Unit | Branch Unit |
| 14 |  |  |  |
| 15 |  |  |  |
| 16 | Branch Unit | Branch Unit | Branch Unit |
| 17 | Branch Unit | Branch Unit | Branch Unit |
| 18 | Mem Unit | Mem Unit | Branch Unit |
| 19 | Mem Unit | Mem Unit | Branch Unit |
| 1A |  |  |  |
| 1B |  |  |  |
| 1C | Mem Unit | FP Unit | Branch Unit |
| 1D | Mem Unit | FP Unit | Branch Unit |
| 1E |  |  |  |
| 1F |  |  |  |

Templates

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | x0 | x1 | x2 | x3 | x4 | x5 | x6 | x7 | x8 | x9 | xA | xB | xC | xD | xE | xF |
| 0x |  |  |  | {R3} | ADD | CSR |  |  | AND | OR | XOR |  | BLEND | REX | LEA | {FLOAT} |
| 1x | SLT | SGE | SLE | SGT | SLTU | SGEU | SLEU | SGTU | SEQ | SNE |  |  |  |  |  |  |
| 2x |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 3x | Bcc | BLcc | FBcc | BBc | BRcc |  | BEQ # | BNE # | JMP | JAL | CALL | RET | SYS |  |  |  |
| 4x | LB | LC | LH | LW | LWR |  |  | {MLX} | LFT |  |  |  |  |  |  |  |
| 5x | SB | SC | SH | SW | SWC | PUSHC |  | {MSX} | SFT |  |  |  |  |  |  |  |
| 6x |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 7x | FORS1 # | FORS2 # | FORS3 # | FOR # | ADDS1 # | ADDS2 # | ADDS3 # |  | ANDS1 # | ANDS2 # | ANDS3 # |  | ORS1 # | ORS2 # | ORS3 # |  |

LDI – Load Immediate Constant

Description:

This instruction loads an immediate constant into one of six registers. Registers numbered 48,49,50, 52,53, and 54 may be used as the target. If the constant is loaded into register 48 or 52, the low order 32 bits of the register are set to the constant, the upper 64 bits of the register are zeroed out. If the constant is loaded into register 49 or 53, the constant is loaded into bits 32 to 63 of the register and the remaining bits are zeroed out. If the constant is loaded into register 50 or 54, the constant is placed in bits 64 to 95 of the register, the low order 64 bits of the constant are zeroed out. Loading constants in this manner allows a full 96 bit constant to be loaded using just four instructions – three constant load instructions plus an OR3.

Notes:

Having two sets of registers available allows building two different constants at the same time. Since many constants are small, they will normally fit into the 21-bit constant field of an immediate operate instruction. 64 and 96-bit constants are used only occasionally so two sets of available constant building registers should be adequate. There is little value to being able to load a constant into any register.

In order to pack as many bits as possible into the 40-bit instruction some bits have be used from the target register and opcode field.