rfPhoenix

Register File

There are 64 general purpose registers.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Reg | Mnen. |  |  | Reg |  |  |
| 0 | Zero |  |  | 32 | VM0 | Vector Mask |
| 1 | A0 | Return Value |  | 33 | VM1 |  |
| 2 | A1 |  |  | 34 | VM2 |  |
| 3 | T0 | Temporaries |  | 35 | VM3 |  |
| 4 | T1 |  |  | 36 | VM4 |  |
| 5 | T2 |  |  | 37 | VM5 |  |
| 6 | T3 |  |  | 38 | VM6 |  |
| 7 | T4 |  |  | 39 | VM7 |  |
| 8 | T5 |  |  | 40 | LC | Loop Counter |
| 9 | T6 |  |  | 41 | LR1 | Link Registers |
| 10 | T7 |  |  | 42 | LR2 |  |
| 11 | S0 | Register Vars |  | 43 | R43 |  |
| 12 | S1 |  |  | 44 | SSP |  |
| 13 | S2 |  |  | 45 | HSP |  |
| 14 | S3 |  |  | 46 | MSP |  |
| 15 | S4 |  |  | 47 | ISP |  |
| 16 | S5 |  |  | 48 | EIP0 |  |
| 17 | S6 |  |  | 49 | EIP1 |  |
| 18 | S7 |  |  | 50 | EIP2 |  |
| 19 | S8 |  |  | 51 | EIP3 |  |
| 20 | S9 |  |  | 52 | EIP4 |  |
| 21 | A2 | Arguments |  | 53 | EIP5 |  |
| 22 | A3 |  |  | 54 | EIP6 |  |
| 23 | A4 |  |  | 55 | EIP7 |  |
| 24 | A5 |  |  | 56 | R56 |  |
| 25 | A6 |  |  | 57 | R57 |  |
| 26 | A7 |  |  | 58 | R58 |  |
| 27 | GP3 |  |  | 59 | R59 |  |
| 28 | GP2 |  |  | 60 | R60 |  |
| 29 | GP1 |  |  | 61 | R61 |  |
| 30 | FP | Frame Pointer |  | 62 | R62 |  |
| 31 | SP | Stack Pointer |  | 63 | R63 |  |

Register Format

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 39 | 3837 | 36 | 25 30 | 29 | 28 23 | 22 20 | 19 | 18 13 | 12 | 11 6 | 5 0 |
| m | ~2 | ~ | Func6 | Tb | Rb6 | Mask3 | Ta | Ra6 | Tt | Rt6 | Opcode6 |

Float Register Format

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 39 | 3837 | 36 | 25 30 | 29 | 28 23 | 22 20 | 19 | 18 13 | 12 | 11 6 | 5 0 |
| m | Rm2 | Tc | Rc6 | Tb | Rb6 | Mask3 | Ta | Ra6 | Tt | Rt6 | Opcode6 |

Immediate Format

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 39 | 38 23 | 22 20 | 19 | 18 13 | 12 | 11 6 | 5 0 |
| m | Immediate16 | Mask3 | Ta | Ra6 | Tt | Rt6 | Opcode6 |

Branch

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 39 23 | 22 20 | 19 | 18 13 | 12 | 11 6 | 5 0 |
| Displacement17 | Cnd3 | 0 | Ra6 | 0 | Rb6 | Opcode6 |

Call

|  |  |  |
| --- | --- | --- |
| 39 8 | 7 6 | 5 0 |
| Target32 | Rt2 | Opcode6 |

Load / Store Format

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 39 | 38 23 | 22 20 | 19 | 18 13 | 12 | 11 6 | 5 0 |
| m | Displacement16 | Mask3 | Ta | Ra6 | Tt | Rt6 | Opcode6 |

Load / Store Indexed Format

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 39 | 3837 | 36 | 25 30 | 29 | 28 23 | 22 20 | 19 | 18 13 | 12 | 11 6 | 5 0 |
| m | ~2 | ~ | Func6 | Tb | Rb6 | Mask3 | Ta | Ra6 | Tt | Rt6 | Opcode6 |

Prefix

|  |  |  |
| --- | --- | --- |
| 39 8 | 7 6 | 5 0 |
| Immediate32 | Sh2 | Opcode6 |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| 0x | 0  BRK | 1  PFX | 2  R2 | 3 | 4  ADD | 5  SUBF | 6  MUL | 7 |
| 1x | 8  AND | 9  OR | 10  XOR | 11 | 12 | 13 | 14  CMP EQ | 15  CMP NE |
| 2x | 16  CMP LT | 17  CMP GE | 18  CMP LE | 19  CMP GT | 20  CMP LTU | 21  CMP GEU | 22  CMP LEU | 23  CMP GTU |
| 3x | 24  CALL abs | 25  CALL rel | 26  JMP abs | 27  JMP rel | 28  Bcc | 29  FBcc | 30  FCMP EQ | 31  FCMP NE |
| 4x | 32 | 33 | 34 | 35 | 36  FCMP LT | 37  FCMP GE | 38  FCMP LE | 39  FCMP GT |
| 5x | 40  ITOF | 41  FTOI | 42 | 43 | 44  FMA | 45  FMS | 46  FNMA | 47  FNMS |
| 6x | 48  LDB | 49  LDBU | 50  LDW | 51  LDWU | 52  LDT | 53 | 54 | 55  LD ndx |
| 7x | 56  STB | 57  STW | 58  STT | 59 | 60 | 61 | 62 | 63  ST ndx |

### ADD - Register-Register

**Description:**

Add two registers and place the sum in the target register. If the instruction is a vector addition then Ra and Rt are vector registers. Rb may be either a vector or a scalar register. The mask register is ignored for scalar instructions. All registers are integer registers.

**Instruction Format:** R2

Results are sign extended to 64-bits.

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 31 30 | 29 27 | 26 | 25 | 24 | 23 19 | 18 14 | 13 9 | 8 | 7 0 |
| Sz2 | m3 | z | ~ | Tb | Rb5 | Ra5 | Rt5 | v | 19h8 |

**Operation:**

Rt = Ra + Rb

**Instruction Format:** R3

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 47 41 | 4038 | 37 | 3635 | 34 | 33 28 | 27 | 26 21 | 20 15 | 14 9 | 8 | 7 0 |
| 04h7 | m3 | z | Sz2 | Tc | Rc6 | Tb | Rb6 | Ra6 | Rt6 | v | 03h8 |

**Operation:**

Rt = Ra + Rb + Rc

**Clock Cycles:** 1

**Execution Units:** All Integer ALU’s

**Exceptions:** none

**Notes:**