# Inst. Memory

The general instruction format is:

inst\_name: output, input1, input2… input N, flag/type

There are two types of instructions: 1) 32-bit for non-memory related instruction, such as prodvm, and 2) 64-bit for memory-related instructions, such as load, strore, dma.

Instruction width 32bit.

|  |  |  |
| --- | --- | --- |
| Instruction sub-types | Bit width | Remark |
| Opcode | 7 | Supporting 128 inst. maximum |
| RegID | 7 | Indexing 128 register entries in total. The four regfiles are indexed uniformly |
| Mem\_addr | 32 | Supporting 4GB virtually memory |
| Long Immediate data | 10 |  |
| Short immediate data | 5 |  |
| Flag | 2 | Dedicated for sign or direction |

The instruction bit-map is as follows:

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Inst | **31** | **30** | **29** | **28** | **27** | **26** | **25** | **24** | **23** | **22** | **21** | **20** | **19** | **18** | **17** | **16** | **15** | **14** | **13** | **12** | **11** | **10** | **9** | **8** | **7** | **6** | **5** | **4** | **3** | **2** | **1** | **0** |
| **Data moving(10)** | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| load0 | **1** | **0** | **0** | **0** | **0** | **0** | **0** | **Reg\_id(24-18)** | | | | | | |  | **Buf\_id(16-15)** | | **Length(14-11)** | | | |  |  |  |  |  |  |  |  |  |  |  |
| load1 | **1** | **0** | **0** | **0** | **0** | **0** | **1** | **Buffer\_addr(24-7)** | | | | | | | | | | | | | | | | | |  |  |  |  |  |  |  |
| store0 | **1** | **0** | **0** | **0** | **0** | **1** | **0** |  |  |  |  |  |  |  |  | **Buf\_id(16-15)** | | **Length(14-11)** | | | | **Reg\_id(10-4)** | | | | | | |  |  |  |  |
| store1 | **1** | **0** | **0** | **0** | **0** | **1** | **1** | **Buffer\_addr(24-7)** | | | | | | | | | | | | | | | | | |  |  |  |  |  |  |  |
| launch0 | **1** | **0** | **0** | **0** | **1** | **0** | **0** | **Layer\_addr(24-18)** | | | | | | |  | **Buf\_id(16-15)** | | **Row(14-11)** | | | |  |  |  |  |  |  |  | **Col(3-0)** | | | |
| launch1 | **1** | **0** | **0** | **0** | **1** | **0** | **1** | **Buffer\_addr(24-7)** | | | | | | | | | | | | | | | | | |  |  |  |  |  |  |  |
| wb0 | **1** | **0** | **0** | **0** | **1** | **1** | **0** |  |  |  |  |  |  |  |  | **Buf\_id(16-15)** | | **Row(14-11)** | | | | **Layer\_addr(10-4)** | | | | | | | **Col(3-0)** | | | |
| wb1 | **1** | **0** | **0** | **0** | **1** | **1** | **1** | **Buffer\_addr(24-7)** | | | | | | | | | | | | | | | | | |  |  |  |  |  |  |  |
| dmalr0 | **1** | **0** | **0** | **1** | **0** | **0** | **0** | **Mem\_addr\_h(24-0)** | | | | | | | | | | | | | | | | | | | | | | | | |
| dmalr1 | **1** | **0** | **0** | **1** | **0** | **0** | **1** | **Buffer\_addr(24-7)** | | | | | | | | | | | | | | | | | | **Mem\_addr\_l(6-0)** | | | | | | |
| dmalr2 | **1** | **0** | **0** | **1** | **0** | **1** | **0** | **Repetition(24-21)** | | | | **Length(20-17)** | | | | **Buf\_id(16-15)** | | **Stride(14-0)** | | | | | | | | | | | | | | |
| dmalc0 | **1** | **0** | **0** | **1** | **0** | **1** | **1** | **Mem\_addr\_h(24-0)** | | | | | | | | | | | | | | | | | | | | | | | | |
| dmalc1 | **1** | **0** | **0** | **1** | **1** | **0** | **0** | **Buffer\_addr(24-7)** | | | | | | | | | | | | | | | | | | **Mem\_addr\_l(6-0)** | | | | | | |
| dmalc2 | **1** | **0** | **0** | **1** | **1** | **0** | **1** | **Repetition(24-21)** | | | | **Length(20-17)** | | | | **Buf\_id(16-15)** | | **Stride(14-0)** | | | | | | | | | | | | | | |
| dmasr0 | **1** | **0** | **0** | **1** | **1** | **1** | **0** | **Mem\_addr\_h(24-0)** | | | | | | | | | | | | | | | | | | | | | | | | |
| dmasr1 | **1** | **0** | **0** | **1** | **1** | **1** | **1** | **Buffer\_addr(24-7)** | | | | | | | | | | | | | | | | | | **Mem\_addr\_l(6-0)** | | | | | | |
| dmasr2 | **1** | **0** | **1** | **0** | **0** | **0** | **0** | **Repetition(24-21)** | | | | **Length(20-17)** | | | | **Buf\_id(16-15)** | | **Stride(14-0)** | | | | | | | | | | | | | | |
| dmasc0 | **1** | **0** | **1** | **0** | **0** | **0** | **1** | **Mem\_addr\_h(24-0)** | | | | | | | | | | | | | | | | | | | | | | | | |
| dmasc1 | **1** | **0** | **1** | **0** | **0** | **1** | **0** | **Buffer\_addr(24-7)** | | | | | | | | | | | | | | | | | | **Mem\_addr\_l(6-0)** | | | | | | |
| dmalr2 | **1** | **0** | **1** | **0** | **0** | **1** | **1** | **Repetition(24-21)** | | | | **Length(20-17)** | | | | **Buf\_id(16-15)** | | **Stride(14-0)** | | | | | | | | | | | | | | |
| mov | **1** | **0** | **1** | **0** | **1** | **0** | **0** | **Reg\_id(24-18)** | | | | | | | **Reg\_id(17-11)** | | | | | | |  |  |  |  |  |  |  |  |  |  |  |
| exmov | **1** | **0** | **1** | **0** | **1** | **0** | **1** | **Reg\_id(24-18)** | | | | | | | **Reg\_id(17-11)** | | | | | | |  |  |  |  |  |  |  |  |  |  |  |
| **Neuro computation(16)** | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| prodvm | **0** | **1** | **0** | **0** | **0** | **0** | **0** | **Reg\_id(24-18)** | | | | | | | **Reg\_id(17-11)** | | | | | | | **Layer\_addr(10-4)** | | | | | | | **Ac** |  |  |  |
| prodvmp | **0** | **1** | **0** | **0** | **0** | **0** | **1** |  |  |  |  |  |  |  | **Reg\_id(17-11)** | | | | | | | **Layer\_addr(10-4)** | | | | | | | **Ac** |  |  |  |
| prodmv | **0** | **1** | **0** | **0** | **0** | **1** | **0** | **Reg\_id(24-18)** | | | | | | | **Reg\_id(17-11)** | | | | | | | **Layer\_addr(10-4)** | | | | | | | **Ac** |  |  |  |
| prodmvp | **0** | **1** | **0** | **0** | **0** | **1** | **1** |  |  |  |  |  |  |  | **Reg\_id(17-11)** | | | | | | | **Layer\_addr(10-4)** | | | | | | | **Ac** |  |  |  |
| prodvv | **0** | **1** | **0** | **0** | **1** | **0** | **0** | **Layer\_addr(24-18)** | | | | | | | **Reg\_id(17-11)** | | | | | | | **Reg\_id (10-4)** | | | | | | | **Ac** |  |  |  |
| prodvvp | **0** | **1** | **0** | **0** | **1** | **0** | **1** |  |  |  |  |  |  |  | **Reg\_id(17-11)** | | | | | | | **Reg\_id(10-4)** | | | | | | | **Ac** |  |  |  |
| prodvvd | **0** | **1** | **0** | **0** | **1** | **1** | **0** | **Reg\_id(24-18)** | | | | | | | **Reg\_id(17-11)** | | | | | | | **Reg\_id(10-4)** | | | | | | |  |  |  |  |
| diff | **0** | **1** | **0** | **0** | **1** | **1** | **1** | **Reg\_id(24-18)** | | | | | | | **Reg\_id(17-11)** | | | | | | |  |  |  |  |  |  |  |  |  |  |  |
| addv | **0** | **1** | **0** | **1** | **0** | **0** | **0** | **Reg\_id(24-18)** | | | | | | | **Reg\_id(17-11)** | | | | | | | **Reg\_id(10-4)** | | | | | | |  |  |  |  |
| addm | **0** | **1** | **0** | **1** | **0** | **0** | **1** | **Layer\_addr(24-18)** | | | | | | | **Layer\_addr(17-11)** | | | | | | | **Layer\_addr(10-4)** | | | | | | |  |  |  |  |
| subv | **0** | **1** | **0** | **1** | **0** | **1** | **0** | **Reg\_id(24-18)** | | | | | | | **Reg\_id(17-11)** | | | | | | | **Reg\_id(10-4)** | | | | | | |  |  |  |  |
| subm | **0** | **1** | **0** | **1** | **0** | **1** | **1** | **Layer\_addr(24-18)** | | | | | | | **Layer\_addr(17-11)** | | | | | | | **Layer\_addr(10-4)** | | | | | | |  |  |  |  |
| max | **0** | **1** | **0** | **1** | **1** | **0** | **0** | **Reg\_id(24-18)** | | | | | | | **Reg\_id(17-11)** | | | | | | | **Reg\_id(10-4)** | | | | | | |  |  |  |  |
| scale | **0** | **1** | **0** | **1** | **1** | **0** | **1** | **Reg\_id(24-18)** | | | | | | | **Reg\_id(17-11)** | | | | | | | **Scale\_fac(10-4)** | | | | | | |  |  |  |  |
| bias | **0** | **1** | **0** | **1** | **1** | **1** | **0** | **Reg\_id(24-18)** | | | | | | | **Reg\_id(17-11)** | | | | | | | **Bias\_fac(10-4)** | | | | | | |  |  |  |  |
| probcmp | **0** | **1** | **0** | **1** | **1** | **1** | **1** | **Reg\_id(24-18)** | | | | | | | **Reg\_id(17-11)** | | | | | | | **Reg\_id(10-4)** | | | | | | |  |  |  |  |
| **Mathematical function(4)** | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| exp | **1** | **1** | **0** | **0** | **0** | **0** | **0** | **Reg\_id(24-18)** | | | | | | | **Reg\_id(17-11)** | | | | | | |  |  |  |  |  |  |  |  |  |  |  |
| log | **1** | **1** | **0** | **0** | **0** | **0** | **1** | **Reg\_id(24-18)** | | | | | | | **Reg\_id(17-11)** | | | | | | |  |  |  |  |  |  |  |  |  |  |  |
| randgen | **1** | **1** | **0** | **0** | **0** | **1** | **0** | **Reg\_id(24-18)** | | | | | | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| act | **1** | **1** | **0** | **0** | **0** | **1** | **1** | **Reg\_id(24-18)** | | | | | | | **Reg\_id(17-11)** | | | | | | |  |  |  |  |  |  |  | **Act type(3-0)** | | | |

# Decode module

AC: accumul control register, it is used to prodvm, prodvmp, prodmv, prodmvp, prodvv, prodvv instructions.

eg. Prodvm:

if accumul=0, v(reg\_id1)=v(reg\_id2)\*m(layer\_addr)

if accumul=1, v(reg\_id1)=v(reg\_id2)\*m(layer\_addr)+v(Rbuffer) (the last time computation result).

Scale/bias factor: Scale / bias factor register (7bits), it is used to scale, bias instructions.

W\_Reg\_id/W\_Layer\_addr: the destination register id register or layer address register (7bits), it is used to prodvm, prodmv, prodvv, prodvvd, diff, addv, addm, subv, subm, sum, max, scale, bias instructions.

//add layer input control signal

case(reg0\_ins\_type)

//prodvm

4'b0000:begin add0\_sel=3'b000;/\*cmo\*/ add1\_sel=3'b000;/\*cmo\*/ end

//prodvmp

4'b0001:begin add0\_sel=3'b000;/\*cmo\*/ add1\_sel=3'b000;/\*cmo\*/ end

//prodmv

4'b0010:begin add0\_sel=3'b001;/\*rmo\*/ add1\_sel=3'b001;/\*rmo\*/ end

//prodmvp

4'b0011:begin add0\_sel=3'b001;/\*rmo\*/ add1\_sel=3'b001;/\*rmo\*/ end

//prodvv

4'b0100:begin add0\_sel=3'b000;/\*cmo\*/ add1\_sel=3'b000;/\*cmo\*/ end

//prodvvp

4'b0101:begin add0\_sel=3'b000;/\*cmo\*/ add1\_sel=3'b000;/\*cmo\*/ end

//prodvvd

4'b0110:begin add0\_sel=3'b000;/\*cmo\*/ add1\_sel=3'b000;/\*cmo\*/ end

//diff

4'b0111:begin add0\_sel=3'b010;/\*rrv0\*/ add1\_sel=3'b010;/\*rrmo\*/ end

//addv

4'b1000:begin add0\_sel=3'b011;/\*rrv0\*/ add1\_sel=3'b011;/\*rrv1\*/ end

//addm

4'b1001:begin add0\_sel=3'b100;/\*rm0\*/ add1\_sel=3'b100;/\*rm1\*/ end

//subv

4'b1010:begin add0\_sel=3'b101;/\*rrv0\*/ add1\_sel=3'b101;/\*~rrv1\*/ end

//subm

4'b1011:begin add0\_sel=3'b110;/\*rm0\*/ add1\_sel=3'b110;/\*~rm1\*/ end

//max

4'b1100:begin add0\_sel=3'b101;/\*rrv0\*/ add1\_sel=3'b101;/\*~rrv1\*/ end

//scale

4'b1101:begin add0\_sel=3'b000;/\*cmo\*/ add1\_sel=3'b000;/\*cmo\*/ end

//bias

4'b1110:begin add0\_sel=3'b011;/\*rrv0\*/ add1\_sel=3'b110;/\*rfactor\*/ end

//probcmp

4'b1111:begin add0\_sel=3'b101;/\*rrv0\*/ add1\_sel=3'b101;/\*~rrv1\*/ end

default:begin end

endcase

//output layer control signal

case(reg1\_ins\_type)

//prodvm

4'b0000:begin ro\_sel=3'b000; mo\_sel=3'b000; rvalid=1'b1; mvalid=1'b0; end

//prodvmp

4'b0001:begin ro\_sel=3'b000; mo\_sel=3'b000; rvalid=1'b0; mvalid=1'b0; end

//prodmv

4'b0010:begin ro\_sel=3'b001; mo\_sel=3'b000; rvalid=1'b1; mvalid=1'b0; end

//prodmvp

4'b0011:begin ro\_sel=3'b001; mo\_sel=3'b000; rvalid=1'b0; mvalid=1'b0; end

//prodvv

4'b0100:begin ro\_sel=3'b000; mo\_sel=3'b000; rvalid=1'b0; mvalid=1'b1; end

//prodvvp

4'b0101:begin ro\_sel=3'b000; mo\_sel=3'b000; rvalid=1'b0; mvalid=1'b0; end

//prodvvd

4'b0110:begin ro\_sel=3'b010; mo\_sel=3'b000; rvalid=1'b1; mvalid=1'b0; end

//diff

4'b0111:begin ro\_sel=3'b011; mo\_sel=3'b000; rvalid=1'b1; mvalid=1'b0; end

//addv

4'b1000:begin ro\_sel=3'b011; mo\_sel=3'b000; rvalid=1'b1; mvalid=1'b0; end

//addm

4'b1001:begin ro\_sel=3'b000; mo\_sel=3'b001; rvalid=1'b0; mvalid=1'b1; end

//subv

4'b1010:begin ro\_sel=3'b011; mo\_sel=3'b000; rvalid=1'b1; mvalid=1'b0; end

//subm

4'b1011:begin ro\_sel=3'b000; mo\_sel=3'b001; rvalid=1'b0; mvalid=1'b1; end

//max

4'b1100:begin ro\_sel=3'b100; mo\_sel=3'b000; rvalid=1'b1; mvalid=1'b0; end

//scale

4'b1101:begin ro\_sel=3'b010; mo\_sel=3'b000; rvalid=1'b1; mvalid=1'b0; end

//bias

4'b1110:begin ro\_sel=3'b011; mo\_sel=3'b000; rvalid=1'b1; mvalid=1'b0; end

//probcmp

4'b1111:begin ro\_sel=3'b101; mo\_sel=3'b000; rvalid=1'b1; mvalid=1'b0; end

default:begin ro\_sel=3'b000; mo\_sel=3'b000; rvalid=1'b1; mvalid=1'b0; end

endcase

end

case(reg1\_ins\_type)

//prodvm

4'b0000:begin ro\_sel=3'b000;/\*add\_reg15\*/ rvalid=1'b1; mvalid=1'b0; end

//prodvmp

4'b0001:begin ro\_sel=3'b000;/\*add\_reg15\*/ rvalid=1'b0; mvalid=1'b0; end

//prodmv

4'b0010:begin ro\_sel=3'b001;/\*add\_reg\_15\*/ rvalid=1'b1; mvalid=1'b0; end

//prodmvp

4'b0011:begin ro\_sel=3'b001;/\*add\_reg\_15\*/ rvalid=1'b0; mvalid=1'b0; end

//prodvv

4'b0100:begin ro\_sel=3'b000;/\*add\_reg15\*/ rvalid=1'b0; mvalid=1'b1; end

//prodvvp

4'b0101:begin ro\_sel=3'b000;/\*add\_reg15\*/ rvalid=1'b0; mvalid=1'b0; end

//prodvvd

4'b0110:begin ro\_sel=3'b010;/\*add\_reg0\*/ rvalid=1'b1; mvalid=1'b0; end

//diff

4'b0111:begin ro\_sel=3'b010;/\*add\_reg0\*/ rvalid=1'b1; mvalid=1'b0; end

//addv

4'b1000:begin ro\_sel=3'b010;/\*add\_reg0\*/ rvalid=1'b1; mvalid=1'b0; end

//addm

4'b1001:begin ro\_sel=3'b000;/\*add\_reg15\*/ rvalid=1'b0; mvalid=1'b1; end

//subv

4'b1010:begin ro\_sel=3'b010;/\*add\_reg0\*/ rvalid=1'b1; mvalid=1'b0; end

//subm

4'b1011:begin ro\_sel=3'b000;/\*add\_reg15\*/ rvalid=1'b0; mvalid=1'b1; end

//max

4'b1100:begin ro\_sel=3'b011;/\*max\_data\*/ rvalid=1'b1; mvalid=1'b0; end

//scale

4'b1101:begin ro\_sel=3'b010;/\*add\_reg0\*/ rvalid=1'b1; mvalid=1'b0; end

//bias

4'b1110:begin ro\_sel=3'b010;/\*add\_reg0\*/ rvalid=1'b1; mvalid=1'b0; end

//probcmp

4'b1111:begin ro\_sel=3'b100;/\*probcm\_data\*/ rvalid=1'b1; mvalid=1'b0; end

default:begin ro\_sel=3'b000; rvalid=1'b0; mvalid=1'b0; end

endcase

//prodvm

4'b0000:begin add0\_sel=4'b0000;/\*cmo\*/ /\*cmo\*/ add\_sel=3'b000;/\*cmo\*/ /\*cmo\*/ end

//prodvmp

4'b0001:begin add0\_sel=4'b0000;/\*cmo\*/ /\*cmo\*/ add\_sel=3'b000;/\*cmo\*/ /\*cmo\*/ end

//prodmv

4'b0010:begin add0\_sel=4'b0001;/\*rmo\*/ /\*rmo\*/ add\_sel=3'b001;/\*cmo\*/ /\*cmo\*/ end

//prodmvp

4'b0011:begin add0\_sel=4'b0001;/\*rmo\*/ /\*rmo\*/ add\_sel=3'b001;/\*cmo\*/ /\*cmo\*/ end

//prodvv

4'b0100:begin add0\_sel=4'b0010;/\*rmo\*/ /\*prodvv\_acdata\*/ add\_sel=3'b010;/\*rmo\*/ /\*prodvv\_acdata\*/ end

//prodvvp

4'b0101:begin add0\_sel=4'b0010;/\*rmo\*/ /\*prodvv\_acdata\*/ add\_sel=3'b010;/\*rmo\*/ /\*prodvv\_acdata\*/ end

//prodvvd

4'b0110:begin add0\_sel=4'b0011;/\*rmo\*/ /\*16'h0000\*/ add\_sel=3'b000;/\*cmo\*/ /\*cmo\*/end

//diff

4'b0111:begin add0\_sel=4'b0100;/\*rrv0\*/ /\*~rrmo\*/ add\_sel=3'b000;/\*cmo\*/ /\*cmo\*/end

//addv

4'b1000:begin add0\_sel=4'b0101;/\*rrv0\*/ /\*rrv1\*/ add\_sel=3'b000;/\*cmo\*/ /\*cmo\*/end

//addm

4'b1001:begin add0\_sel=4'b0110;/\*rm0\*/ /\*rm1\*/ add\_sel=3'b011;/\*rm0\*/ /\*rm1\*/end

//subv

4'b1010:begin add0\_sel=4'b0111;/\*rrv0\*/ /\*~rrv1\*/ add\_sel=3'b000;/\*cmo\*/ /\*cmo\*/end

//subm

4'b1011:begin add0\_sel=4'b1000;/\*rm0\*/ /\*~rm1\*/ add\_sel=3'b100; /\*rm0\*/ /\*~rm1\*/ end

//max

4'b1100:begin add0\_sel=4'b0111;/\*rrv0\*/ /\*~rrv1\*/ add\_sel=3'b000;/\*cmo\*/ /\*cmo\*/end

//scale

4'b1101:begin add0\_sel=4'b0011;/\*rmo\*/ /\*16'h0000\*/ add\_sel=3'b000;/\*cmo\*/ /\*cmo\*/end

//bias

4'b1110:begin add0\_sel=4'b1001;/\*rrv0\*/ /\*rfactor\*/ add\_sel=3'b000;/\*cmo\*/ /\*cmo\*/end

//probcmp

4'b1111:begin add0\_sel=4'b0111;/\*rrv0\*/ /\*~rrv1\*/ add\_sel=3'b000;/\*cmo\*/ /\*cmo\*/end

default:begin add0\_sel=4'b0000;/\*cmo\*/ /\*cmo\*/ add\_sel=3'b000;/\*cmo\*/ /\*cmo\*/ end