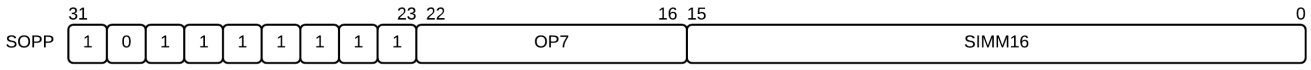
Miaow - Implemented Instructions Specification

November 28, 2012

1 Supported Instruction Formats

In this section the supported instruction formats are presented. A brief discription of each field is given. Refer to the ISA specification for more detailed information.

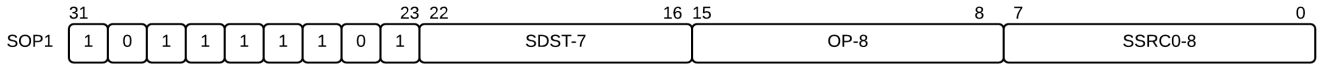
1.1 SOPP



OP[7]: Opcode for instruction.

SIMM[16]: 16 bit immediate value. Signedness is determined by opcode.

1.2 SOP1

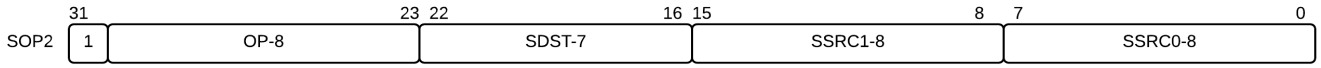


OP[8]: Opcode for instruction.

SDST[7]: Destination for the instruction. Must be a scalar 32 bit register. Can addres SGPRs, trap regisgers, VCC, M0 and EXEC.

SSRC0[8]: Source of instruction. Must be a scalar 32 bit register. Support all values of SDST and some aditional constants.

1.3 SOP2



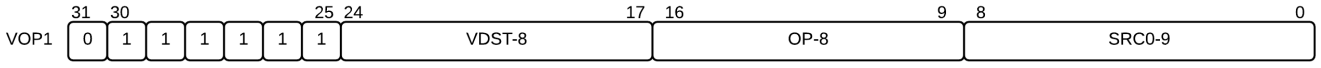
OP[8]: Opcode for instruction.

SDST[7]: Destination for the instruction. Must be a scalar 32 bit register. Can addres SGPRs, trap regisgers, VCC, M0 and EXEC.

SSRC0[8]: Source of instruction. Must be a scalar 32 bit register. Support all values of SDST and some aditional constants.

SSRC1[8]: Source of instruction. Must be a scalar 32 bit register. Support all values of SDST and some aditional constants.

1.4 VOP1

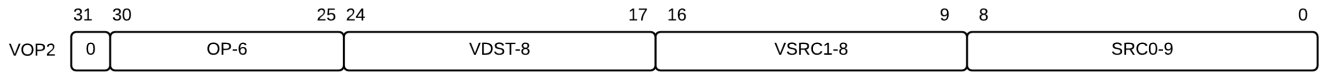


OP[8]: Opcode for instruction.

VDST[8]: Destination of the instruction. Can address all VGPRs.

SRC0[9]: Source for the instruction. Can address all scalar values and VGPRs

1.5 VOP2



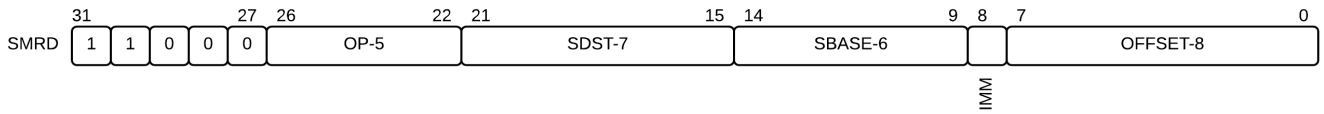
OP[6]: Opcode for instruction.

VDST[8]: Destination of the instruction. Can address all VGPRs.

SRC0[9]: Source for the instruction. Can address all scalar values and VGPRs

VSRC1[8]: Source for the instrucion. Can address all VGPRs

1.6 SMRD



OP[5]: Opcode for instruction.

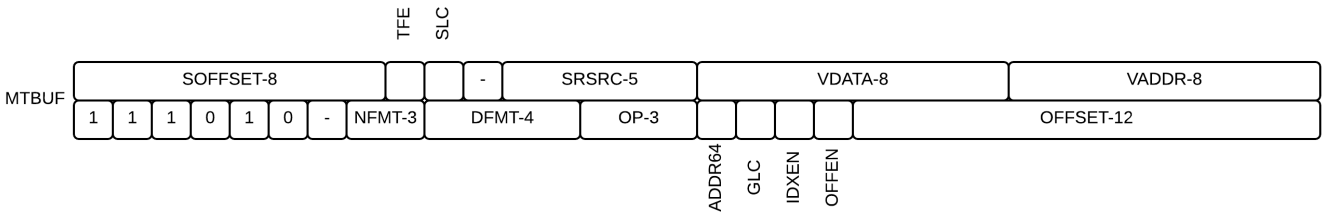
SDST[7]: Destination for the instruction. Must be a scalar 32 bit register. Can addres SGPRs, trap regisgers, VCC, M0 and EXEC.

SBASE[6]: Bits [6:1] of an aligned pair of SGPRs specifying size[16], base[48], where base and size are in Dword units. The low-order bits are in the first SGPR.

IMM[1]: If 1, OFFSET specifies a immediate offset, otherwise, offset specifies a sgpra that has the 8 bit offset.

OFFSET[8]: 8 bits unsigned immediate to be added to SBASE or address to SGPR that has the 8 bit offset according to IMM flag

1.7 MTBUF



OFFSET[12]: Unsigned bit offset for ADDR. Used only when offen is 0

OFFEN[1]: If this is 0, offset is specified by OFFSET otherwise VADDR is an offset

IDXEN[1]: If set, VADDR is an index otherwise index is 0.

GLC[1]: If set, operation is globally coherent

ADDR64[1]: If set, buffer addr is 64 bits.

OP[3]: Opcode for instruction.

DFMT[4]: Data format for typed buffer

NFMT[3]: Number format for typed buffer

VADDR[8]: Address for the operation. Can carry a offset or a index

VDATA[8]: VGPR to read/write result to

SRSRC[5]: ADDR of the 4 SGPRs that specify resource constant.

SLC[1]: System level coherent

TFE[1]: Texture fail enable

SOFFSET[8]: Specifies the scalar offset to be added to memory addr.

2 Instructions

2.1 Scalar ALU

Mnemonic	Format	Opcode	Discription
S_ENDPGM	SOPP	0x1	Terminate wavefront.
S_ADD_U32/_I32	SOP2	0x0/0x2	Unsigned or signed scalar add operation. SCC is set if carry out/overflow occurs.
S_SUB_I32	SOP2	0x3	Unsigned or signed scalar sub operation. SCC is set if carry out/overflow occurs.
S_AND_B32	SOP2	0xE	Bitwise AND scalar operation. SCC is set if result is non-zero.
S_OR_B32	SOP2	0x10	Bitwise OR scalar operation. SCC is set if result is non-zero.
S_MOV_B32	SOP1	0x3	Move scalar source to scalar destination.
S_NOT_B32	SOP1	0x7	Bitwise NOT scalar operation. SCC is set if result is non-zero.

2.2 Vector ALU

Mnemonic	Format	Opcode	Discription
V_MOV_B32	VOP1	0x1	Move vector value from source to vector destination. If a scalar source is given, the value is copied for each wave item.
V_ADD_I32	VOP2	0x25	$D.u = S0.u + S1.u$Integer signed/unsigned ADD. Carry out is written to VCC. If SRC0 is a scalar source, the value is copied for each wave item.
V_SUB_I32	VOP2	0x26	$D.u = S0.u - S1.u$ Integer sined/unsigned SUB. Borrow out is written to VCC. If SRC0 is a scalar source, the value is copied for each wave item.
V_AND_B32	VOP2	0x1B	Logical bitwise AND. If SRC0 is a scalar source, the value is copied for each wave item.
V_OR_B32	VOP2	0x1C	Logical bitwise OR. If SRC0 is a scalar source, the value is copied for each wave item.
V_LSHLREV_B32	VOP2	0x1A	$D.u = S1.u << S0.u[4:0]$. Scalar logical shift left. If SRC0 is a scalar source, the value iscopied for each wave item.
V_LSHRREV_B32	VOP2	0x16	$D.u = S1.u >> S0.u[4:0]$. Scalar logical shift right. If SRC0 is a scalar source, the value is copied for each wave item.

2.3 Load store unit

2.3.1 S_BUFFER_LOAD

Format: SMRD

Opcode: 0x8

Discription: Read a Dword from read-only buffer.

```
m_offset = IMM ? OFFSET : SGPR[OFFSET]
m_base = { SGPR[SBASE +1][15:0], SGPR[SBASE] }
m_stride = SGPR[SBASE +1][31:16]
m_num_records = SGPR[SBASE+2]
m_size = (m_stride == 0) ? 1 : m_num_records
m_addr = (SGPR[SBASE] + m_offset) & ~0x3
SGPR[SDST] = read_dword_from_kcache(m_base, m_offset, m_size)
```

2.3.2 T_BUFFER_LOAD_FORMAT_X

Format: MTBUF

Opcode: 0x00

Discription: Typed load from buffer memory with format conversion. Addres is calculates according to:

```
ADDR = Base + baseOffset + Ioffset + Voffset + Stride * (Vindex + TID)
          T#       SGPR        Instr     VGPR        T#    VGPR   0..63
```

T# is located at buffer resource constant, which in turn, is located at scalar registers addressed by SRSRC.

See ISA spec, pag 8-9, table 8.5 for details. s

Voffset is ignored when instruction bit OFFEN == 0

Vindex is ignored when instructino bit IDXEN == 0

TID is a constant value (0..63) unique to each thread in the wave. It is ignored when resource bit ADD_TID_ENABLE==0

2.3.3 T_BUFFER_STORE_FORMAT_X

Format: MTBUF

Opcode: 0x04

Discription: Typed store from buffer memory with format conversion. Address calculation is the same as T_BUFFER_LOAD_FORMAT_X

3 Wishlist

Mnemonic	Format	Opcode	Discription
V_CMP_*	VOPC		
S_MIN_U32	SOP2		
S_MAX_U32	SOP2		

- MULTIPLY
- WAITCNT
- MIAOW COMPILER!