MIDE instruction reference data

Core Instruction Set

R type

Name	Mnemonic	Operation	Sintaxis	OPcode	ALUop
ADD	add	Rd = Rs1 + Rs2	add Rd, Rs1, Rs2	00000	000
Substract	sub	Rd = Rs1 - Rs2	sub Rd, Rs1,Rs2	00000	001
Multiply	mul	Rd = Rs1[15:0]*Rs2[15:0]	mul Rd, Rs1,Rs2	00000	010
Shift Left	sl	Rd = Rs1 << Rs2	sl Rd, Rs1, Rs2	00000	100
And	and	Rd = Rs1 & Rs2	and Rd, Rs1, Rs2	00000	101
Shift Right	sr	Rd = Rs1 >> Rs2	sr Rd, Rs1, Rs2	00000	110
Load Word	ldw	Rd = Mem[Rs1[15:0]]	ldw Rd, Rs1	00010	-
Load Half-Word	ldh	Rd = {16'b0,M[Rs1](15:0)}	ldh Rd, Rs1	00011	-
Store Word	stw	M[Rs2[15:0]] = Rs1	stw Rs1, Rs2	00100	-
Store Half word	sth	M[Rs2[15:0]] = Rs1	sth Rs1, Rs2	00101	-
Jump Register	jr	PC = Rs1	jr Rs1	00110	-

I type

Name	Mnemonic	Operation	Sintaxis	OPcode	ALUop
Add Immediate	addi	Rd = Rs1 + Imm	addi Rd, Rs1, Imm	01000	000
Substract Immediate	subi	Rd = Rs1 - Imm	subi Rd, Rs, Imm	01001	001
Branch on equal	be	if(Rs1 == Rs2) PC + 4 + Imm	be Rs1, Rs2, "label"	01010	-
Branch on greater than	bgt	if(Rs1 > Rs2) PC + 4 + Imm	bgt Rs1, Rs2, "label"	01011	-

J type

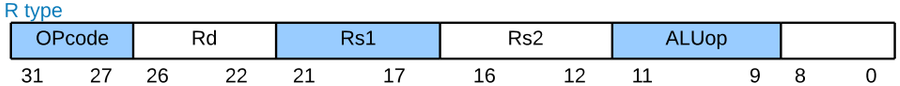
Name	Mnemonic	Operation	Sintaxis	OPcode	ALUop
Jump	jmp	PC = Address	jmp "label"	10000	-
Call function	call	RA = PC; PC = Address	call "label"	10001	-

V type

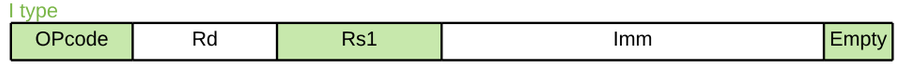
Name	Mnemonic	Operation	sintaxis	OPcode	ALUop
Vector Load Byte	vldh	Vd = {24'b0, M[RVs1](7:0)}	vldh Vd, Rs	11011	-
Vector Store Byte	vstb	M[Rs2[15:0]]={24'b0, VRs1(7:0)}	vstb Vd, RVs1	11101	-
Nearest neighbor interpolation 1	vnni1	Vd = {Vs1[0], Vs1[0], Vs1[1], Vs1[1]}	vnni1 Vd, RVs1	11000	001
Nearest neighbor interpolation 2	vnni2	Vd = {Vs1[2], Vs1[2], Vs1[3], Vs1[3]}	vnni2 Vd, RVs1	11000	010
Bilinear interpolation vertical	vbiv	Vd = 2/3*Vs1 + 1/3*Vs2	vbiv Vd, VRs1, VRs2	11000	100
Bilinear interpolation horizontal 1	vbih1	Vd = {Vs[0], Vs[0]*2/3 + Vs[1]*1/3, Vs[1]*2/3+Vs[0]*1/3, Vs[1]}	vbih1 Vd, VRs1	11000	101
Bilinear interpolation horizontal 2	vbih2	Vd = {Vs[1]*2/3+Vs[2]*1/3, Vs[2]*2/3+Vs[1]*1/3, Vs[2], Vs[2]}	vbih2 Vd, VRs1	11000	110
Bilinear interpolation horizontal 3	vbih3	Vd = {Vs1[3]*2/3+Vs1[2]*1/3, Vs1[3], Vs1[3]*2/3+Vs2[0]*1/3, Vs2[0]*2/3+Vs1[3]*1/3}	vbih3 Vd, VRs1, VRs2	11000	111

Formats

R type



I type



J type

V type

Registers		
Name	Number	Use
R0	0	Constant 0
R1-R10	1-10	Scalar General Purpose
VP	11	VRAM Pointer
IC	12	Image Counter
IP	13	Image Pointer
RA	14	Return Address
SP	15	Stack Pointer
V1 - V8	16-23	Vectorial General Purpose

Memory Map	
8 bit Hexa	Section name
160000-291072	VROM Full
1036-91036	VRAM Cuadrant
408	Cuadrant
404	Interpolación
400	Start Button
0 - 9F	Stack - Instructions

Addressing Modes		
Instruction	Type	Mode
be	I	PC relative.
bgt	I	PC relative.
jr	R	Register.
jmp	J	Immediate.
call	J	Immediate.

Size and data type	
Word	32 bits
Half Word	16 bits
Byte	8 bits
Vectorial Register	Composed by 8 independent words.

Created and implemented by

Jasson Rodríguez Méndez.

Marco Herrera Valverde.

Kenneth Hernández Salazar.

Edgar Chaves González.