OSOROM Instruction Set Reference

The Moroso Project

June 11, 2014

Contents

Chapter 1

CPU Architecture

1.1 Instructions

Each instruction is a 32-bit word, stored in little-endian order. The top 2 bits of the instruction select one of four predicate registers, and the third bit from the top optionally inverts it. The instruction is only executed if the resulting predicate is true.

1.2 VLIW

At each time step, the CPU fetches a "packet" of 4 instructions located contiguously in memory and executes them in parallel. These packets must be aligned to their 16-byte size, and branch targets must also be so aligned.

There are three types of instructions: Control, Memory, and ALU instructions. Only one control instruction may be executed in any given packet; it must occupy the first (lowest-address) slot in its packet. Likewise, only the first two slots in a packet may execute memory instructions. ALU instructions may be located in any slot.

ALU instructions in the first three slots may optionally specify a "long immediate" operand. In this case, the 32 bits following that instruction are interpreted as an operand, and a no-op is scheduled in the following slot of that packet.

1.3 Registers

There are 32 32-bit general-purpose registers, R0 through R31. R31 is used as the link register for BL instructions. The program counter is stored separately, and may only be modified through branch instructions and read through BL instructions.

There are three one-bit predicate registers that may be written to by compare instructions and used to conditionally execute any instruction.

1.4 Shifting

The ALU is preceded by a shifter for its second operand

1.5 Virtual Memory

Virtual memory is accomplished through a hardware-filled TLB, with 4KB pages and a 2-level page-table structure that looks like x86's without any of the fancy bits. (more detail to come in this section; in particular, we want to boot with paging on but it's not clear how the world looks then)

1.6 Exceptions

Chapter 2

Instruction Listing

2.1 ADD - 32-bit Addition

2.1.1 Encoding

3	3	2	2	2	2	2	2	2	2	2	2	1	1	1	1	1	1	1	1	1	1											
1	0	9	8	7	6	5	4	3	2	1	0	9	8	7	6	5	4	3	2	1	0	9	8	7	6	5	4	3	2	1	0	
PRED			0	CONSTANT										ROTATE				0	0	0	0	RD					RS					
PRED			1	0	1	SHIFTAMT					SHF		RT					0	0	0	0	RD					RS					
PRED			1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	RD					RS					LIM

2.1.2 Syntax

- `ADD[PN] Rd, Rs, #Imm`
- `ADD[PN] Rd, Rs, Rt [SHF #Imm]`

2.1.3 Type

ADD is an ALU operation. It may be placed in any slot of a packet. If the long immediate form is used it must not be located in the last slot in a packet, and no instruction may be specified in the following slot.

2.1.4 Behavior

Adds the two operands together into the destination register. If two register operands are specified, the second may optionally be shifted by an immediate value before the addition is performed.

TODO optional overflow exceptions???

2.2 AND

2.2.1 Encoding

3	3	2	2	2	2	2	2	2	2	2	2	1	1	1	1	1	1	1	1	1	1											
1	0	9	8	7	6	5	4	3	2	1	0	9	8	7	6	5	4	3	2	1	0	9	8	7	6	5	4	3	2	1	0	
PRED			0	CONSTANT										ROTATE				0	0	0	1	RD					RS					
PRED			1	0	1	SHIFTAMT					SHF		RT					0	0	0	1	RD					RS					
PRED			1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	RD					RS					LIM

2.3 B

2.3.1 Encoding

3	3	2	2	2	2	2	2	2	2	2	2	1	1	1	1	1	1	1	1	1	1										
1	0	9	8	7	6	5	4	3	2	1	0	9	8	7	6	5	4	3	2	1	0	9	8	7	6	5	4	3	2	1	0
PRED			1	1	0	0	OFFSET																								
PRED			1	1	1	0	OFFSET																					RS				

2.4 BL

2.4.1 Encoding

3	3	2	2	2	2	2	2	2	2	2	2	1	1	1	1	1	1	1	1	1	1										
1	0	9	8	7	6	5	4	3	2	1	0	9	8	7	6	5	4	3	2	1	0	9	8	7	6	5	4	3	2	1	0
PRED			1	1	0	1	OFFSET																								
PRED			1	1	1	1	OFFSET																				RS				

2.5 BREAK

2.5.1 Encoding

3	3	2	2	2	2	2	2	2	2	2	2	1	1	1	1	1	1	1	1	1	1										
1	0	9	8	7	6	5	4	3	2	1	0	9	8	7	6	5	4	3	2	1	0	9	8	7	6	5	4	3	2	1	0
PRED			1	0	0	0	1	0	0	0	1	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X

2.6 CMP

2.6.1 Encoding

3	3	2	2	2	2	2	2	2	2	2	2	1	1	1	1	1	1	1	1	1	1											
1	0	9	8	7	6	5	4	3	2	1	0	9	8	7	6	5	4	3	2	1	0	9	8	7	6	5	4	3	2	1	0	
PRED			0	CONSTANT										ROTATE				0	1	1	1	CMPTP			PD		RS					
PRED			1	0	1	SHIFTAMT					SHF		RT					0	1	1	1	CMPTP			PD		RS					
PRED			1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	CMPTP			PD		RS					LIM

TODO do we want one entry for all compares that lists the types, or an entry for each one?

2.7 CPOP

2.7.1 Encoding

3	3	2	2	2	2	2	2	2	2	2	2	1	1	1	1	1	1	1	1	1	1										
1	0	9	8	7	6	5	4	3	2	1	0	9	8	7	6	5	4	3	2	1	0	9	8	7	6	5	4	3	2	1	0
PRED			1	0	0	0	1	0	1	0	1	C	X	X	X	X	X	CPOPC				X	X	X	X	X	RS				

2.8 DIV

2.8.1 Encoding

3	3	2	2	2	2	2	2	2	2	2	2	1	1	1	1	1	1	1	1	1	1										
1	0	9	8	7	6	5	4	3	2	1	0	9	8	7	6	5	4	3	2	1	0	9	8	7	6	5	4	3	2	1	0
PRED			1	0	0	0	1	1	0	0	1	X	RT					X	X	X	X	RD					RS				

2.9 ERET

2.9.1 Encoding

3	3	2	2	2	2	2	2	2	2	2	2	1	1	1	1	1	1	1	1	1	1										
1	0	9	8	7	6	5	4	3	2	1	0	9	8	7	6	5	4	3	2	1	0	9	8	7	6	5	4	3	2	1	0
PRED			1	0	0	0	1	0	1	0	0	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X

2.10 FENCE

2.10.1 Encoding

3	3	2	2	2	2	2	2	2	2	2	2	1	1	1	1	1	1	1	1	1	1										
1	0	9	8	7	6	5	4	3	2	1	0	9	8	7	6	5	4	3	2	1	0	9	8	7	6	5	4	3	2	1	0
PRED			1	0	0	0	1	0	0	1	1	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X

2.11 LB

2.11.1 Encoding

3	3	2	2	2	2	2	2	2	2	2	2	1	1	1	1	1	1	1	1	1	1										
1	0	9	8	7	6	5	4	3	2	1	0	9	8	7	6	5	4	3	2	1	0	9	8	7	6	5	4	3	2	1	0
PRED			1	0	0	1	OFFSET												0	0	0	RD					RS				

2.12 LH

2.12.1 Encoding

3	3	2	2	2	2	2	2	2	2	2	2	1	1	1	1	1	1	1	1	1	1										
1	0	9	8	7	6	5	4	3	2	1	0	9	8	7	6	5	4	3	2	1	0	9	8	7	6	5	4	3	2	1	0
PRED			1	0	0	1	OFFSET												0	0	1	RD					RS				

2.13 LL

2.13.1 Encoding

3	3	2	2	2	2	2	2	2	2	2	2	1	1	1	1	1	1	1	1	1	1										
1	0	9	8	7	6	5	4	3	2	1	0	9	8	7	6	5	4	3	2	1	0	9	8	7	6	5	4	3	2	1	0
PRED			1	0	0	1	OFFSET												0	1	1	RD					RS				

2.14 LW

2.14.1 Encoding

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
PRED			1	0	0	1	OFFSET												0	1	0	RD					RS				

2.15 MFC

2.15.1 Encoding

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
PRED			1	0	0	0	1	0	1	1	0	X	X	X	X	X	X	X	X	X	X	RD					CPRS				

2.16 MFHI

2.16.1 Encoding

3	3	2	2	2	2	2	2	2	2	2	2	1	1	1	1	1	1	1	1	1	1										
1	0	9	8	7	6	5	4	3	2	1	0	9	8	7	6	5	4	3	2	1	0	9	8	7	6	5	4	3	2	1	0
PRED			1	0	0	0	1	0	1	1	1	X	X	X	X	X	X	X	X	X	X	RD					X	X	X	X	X

2.17 MOV

2.17.1 Encoding

3	3	2	2	2	2	2	2	2	2	2	2	1	1	1	1	1	1	1	1	1	1											
1	0	9	8	7	6	5	4	3	2	1	0	9	8	7	6	5	4	3	2	1	0	9	8	7	6	5	4	3	2	1	0	
PRED			0	CONSTHIGH													R	1	0	0	0	RD					CONSTLOW					
PRED			1	0	1	SHIFTAMT					SHF		X	X	X	X	X	1	0	0	0	RD					RS					
PRED			1	0	0	0	0	0	0	1	SHF		RT					1	0	0	0	RD					RS					
PRED			1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	RD					RS					LIM

2.18 MTC

2.18.1 Encoding

3	3	2	2	2	2	2	2	2	2	2	2	1	1	1	1	1	1	1	1	1	1										
1	0	9	8	7	6	5	4	3	2	1	0	9	8	7	6	5	4	3	2	1	0	9	8	7	6	5	4	3	2	1	0
PRED			1	0	0	0	1	0	1	1	1	X	X	X	X	X	X	X	X	X	X	CPRD					RS				

2.19 MTHI

2.19.1 Encoding

3	3	2	2	2	2	2	2	2	2	2	2	1	1	1	1	1	1	1	1	1	1										
1	0	9	8	7	6	5	4	3	2	1	0	9	8	7	6	5	4	3	2	1	0	9	8	7	6	5	4	3	2	1	0
PRED			1	0	0	0	1	0	1	1	1	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	RS				

2.20 MVN

2.20.1 Encoding

3	3	2	2	2	2	2	2	2	2	2	2	1	1	1	1	1	1	1	1	1	1											
1	0	9	8	7	6	5	4	3	2	1	0	9	8	7	6	5	4	3	2	1	0	9	8	7	6	5	4	3	2	1	0	
PRED			0	CONSTHIGH													R	1	0	0	1	RD					CONSTLOW					
PRED			1	0	1	SHIFTAMT					SHF		X	X	X	X	X	1	0	0	1	RD					RS					
PRED			1	0	0	0	0	0	0	1	SHF		RT					1	0	0	1	RD					RS					
PRED			1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	RD					RS					LIM

2.21 MULT

2.21.1 Encoding

3	3	2	2	2	2	2	2	2	2	2	2	1	1	1	1	1	1	1	1	1	1										
1	0	9	8	7	6	5	4	3	2	1	0	9	8	7	6	5	4	3	2	1	0	9	8	7	6	5	4	3	2	1	0
PRED			1	0	0	0	1	1	0	0	0	X	RT					X	X	X	X	RD					RS				

2.22 NOR

2.22.1 Encoding

3	3	2	2	2	2	2	2	2	2	2	2	1	1	1	1	1	1	1	1	1	1											
1	0	9	8	7	6	5	4	3	2	1	0	9	8	7	6	5	4	3	2	1	0	9	8	7	6	5	4	3	2	1	0	
PRED			0	CONSTANT										ROTATE				0	0	1	0	RD					RS					
PRED			1	0	1	SHIFTAMT					SHF		RT					0	0	1	0	RD					RS					
PRED			1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	RD					RS					LIM

2.23 OR

2.23.1 Encoding

3	3	2	2	2	2	2	2	2	2	2	2	1	1	1	1	1	1	1	1	1	1											
1	0	9	8	7	6	5	4	3	2	1	0	9	8	7	6	5	4	3	2	1	0	9	8	7	6	5	4	3	2	1	0	
PRED			0	CONSTANT										ROTATE				0	0	1	1	RD					RS					
PRED			1	0	1	SHIFTAMT					SHF		RT					0	0	1	1	RD					RS					
PRED			1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	RD					RS					LIM

2.24 RSB

2.24.1 Encoding

3	3	2	2	2	2	2	2	2	2	2	2	1	1	1	1	1	1	1	1	1	1											
1	0	9	8	7	6	5	4	3	2	1	0	9	8	7	6	5	4	3	2	1	0	9	8	7	6	5	4	3	2	1	0	
PRED			0	CONSTANT										ROTATE				0	1	0	1	RD					RS					
PRED			1	0	1	SHIFTAMT					SHF		RT					0	1	0	1	RD					RS					
PRED			1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	RD					RS					LIM

2.25 SB

2.25.1 Encoding

3	3	2	2	2	2	2	2	2	2	2	2	1	1	1	1	1	1	1	1	1	1										
1	0	9	8	7	6	5	4	3	2	1	0	9	8	7	6	5	4	3	2	1	0	9	8	7	6	5	4	3	2	1	0
PRED			1	0	0	1	OFFSET												1	0	0	RD					RS				

2.26 SC

2.26.1 Encoding

3	3	2	2	2	2	2	2	2	2	2	2	1	1	1	1	1	1	1	1	1	1										
1	0	9	8	7	6	5	4	3	2	1	0	9	8	7	6	5	4	3	2	1	0	9	8	7	6	5	4	3	2	1	0
PRED			1	0	0	1	OFFSET												1	1	1	RD					RS				

2.27 SH

2.27.1 Encoding

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
PRED			1	0	0	1	OFFSET												1	0	1	RD					RS				

2.28 SUB

2.28.1 Encoding

3	3	2	2	2	2	2	2	2	2	2	2	1	1	1	1	1	1	1	1	1	1											
1	0	9	8	7	6	5	4	3	2	1	0	9	8	7	6	5	4	3	2	1	0	9	8	7	6	5	4	3	2	1	0	
PRED			0	CONSTANT										ROTATE				0	1	0	0	RD					RS					
PRED			1	0	1	SHIFTAMT					SHF		RT					0	1	0	0	RD					RS					
PRED			1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	RD					RS					LIM

2.29 SW

2.29.1 Encoding

3	3	2	2	2	2	2	2	2	2	2	2	1	1	1	1	1	1	1	1	1	1										
1	0	9	8	7	6	5	4	3	2	1	0	9	8	7	6	5	4	3	2	1	0	9	8	7	6	5	4	3	2	1	0
PRED			1	0	0	1	OFFSET												1	1	0	RD					RS				

2.30 SXB

2.30.1 Encoding

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	
PRED			0	CONSTHIGH													R	1	0	1	0	RD					CONSTLOW					
PRED			1	0	1	SHIFTAMT					SHF		X	X	X	X	X	1	0	0	1	RD					RS					
PRED			1	0	0	0	0	0	0	1	SHF		RT					1	0	1	0	RD					RS					
PRED			1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	RD					RS					LIM

2.31 SXH

2.31.1 Encoding

3	3	2	2	2	2	2	2	2	2	2	2	1	1	1	1	1	1	1	1	1	1											
1	0	9	8	7	6	5	4	3	2	1	0	9	8	7	6	5	4	3	2	1	0	9	8	7	6	5	4	3	2	1	0	
PRED			0	CONSTHIGH													R	1	0	1	1	RD					CONSTLOW					
PRED			1	0	1	SHIFTAMT					SHF		X	X	X	X	X	1	0	1	1	RD					RS					
PRED			1	0	0	0	0	0	0	1	SHF		RT					1	0	1	1	RD					RS					
PRED			1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	1	RD					RS					LIM

2.32 SYSCALL

2.32.1 Encoding

3	3	2	2	2	2	2	2	2	2	2	2	1	1	1	1	1	1	1	1	1	1										
1	0	9	8	7	6	5	4	3	2	1	0	9	8	7	6	5	4	3	2	1	0	9	8	7	6	5	4	3	2	1	0
PRED			1	0	0	0	1	0	0	1	0	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X

2.33 XOR

2.33.1 Encoding

3	3	2	2	2	2	2	2	2	2	2	2	1	1	1	1	1	1	1	1	1	1											
1	0	9	8	7	6	5	4	3	2	1	0	9	8	7	6	5	4	3	2	1	0	9	8	7	6	5	4	3	2	1	0	
PRED			0	CONSTANT										ROTATE				0	1	1	0	RD					RS					
PRED			1	0	1	SHIFTAMT					SHF		RT					0	1	1	0	RD					RS					
PRED			1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	RD					RS					LIM