sml2008-am01: Decoded Instruction Format

Adam Megacz

November 14, 2008

Abstract

This memo gives the decoded, 37-bit-wide, mostly-one-hot format used internally within the dock circuitry to represent an instruction.

Changes:

14-Nov	Inverted bits 14 and 13 in the move instruction forms
10-Nov	Swapped order of FlagA and FlagB fields
	Documented Set Flags truth table field
01-Nov	Changed `Rq` to `OS`
	Changed `Int` to $\overline{\texttt{Int}}$
	Swapped Z and !Z
31-Oct	Added encoding of `Predicate` field
30-Oct	Divided `move` instruction into subinstructions based on path latch
29-Oct	Added TAIL instruction
	Removed "done" bit, relocated infinity bit
23-Oct	Changed polarity of bit 20 on "Shift" and "Set Data Latch"
	Noted that "Immediate→ILC" must have bit 7 set to 0
	Labeled bits 9 and 7 on last two instruction forms
21-Aug	Initial Revision

Overview

FleetTwo Instructions in main memory occupy 37 bits. Of this, 11 bits give the path to the dock which is to execute the instruction; thus, only 26 of these bits are interpreted by the dock.

It is easiest to design the OD and EX stages of the dock if the control bits supplied there are mostly one-hot encoded. Moreover, due to layout considerations there is very little cost associated with making the instruction fifo 36 bits wide rather than 26 bits wide.

Due to these two considerations, all 26-bit instructions binary-coded-control instructions are expanded into 36-bit unary-coded-control instructions upon entry to the instruction fifo. This memo documents the 36-bit unary-coded-control format.

Predicate Field

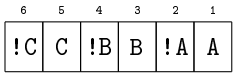
The `Predicate` field, common to many instructions, consists of a six-bit wide, one-hot encoded field. The instruction will be **skipped** (not executed) if **any** condition corresponding to a bit whose value is one is met.

The `Z` flag is an "imaginary" flag which is "set" iff the outer loop counter is zero.

For example, if bits 31 and 34 are set, the instruction will be skipped if either the `B` flag is cleared or the `A` flag is set. Equivalently, it will be executed iff the `B` flag is set and the `A` flag is cleared.

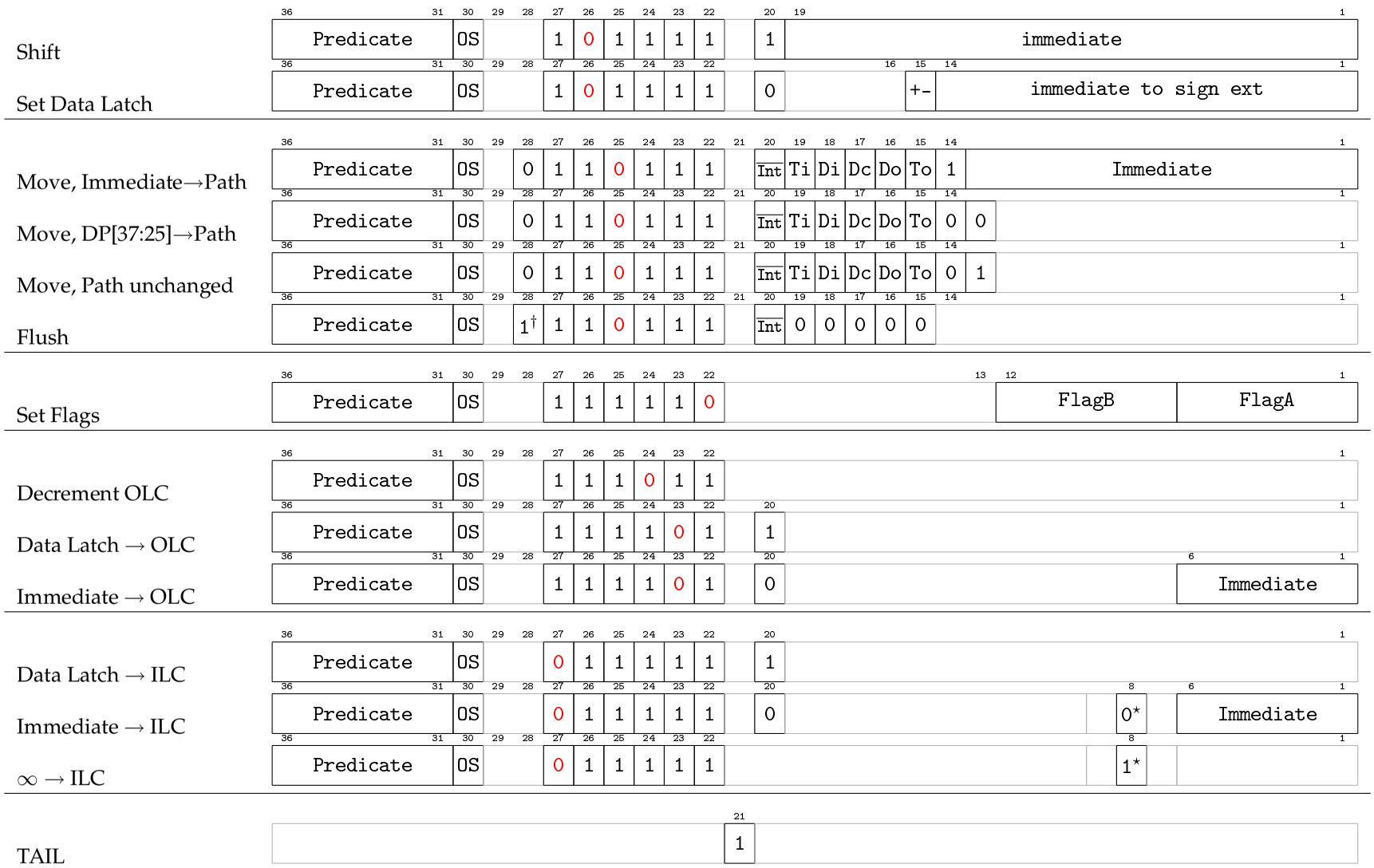
Set Flags

Each of the `FlagA` and `FlagB` fields in the Set Flags instruction gives a truth table; the new value of the flag is the logical OR of the inputs whose bits are set to `1`.



Legend

`OS` = One-Shot (0=Requeueing, 1=Not-Requeueing)
$\overline{\texttt{Int}}$ = Not Interruptible (0=Torpedoable, 1=Not-Torpedoable)



⋆ – bit 8 is the “infinity” bit
† – bit 28 is copied to the “flushing latch” whenever a move instruction occurs