Shane Cincotta

3/18/19

Lab 1

ISA Design

Op-code	Instruction
00	Add
01	Sub
10	LI
11	Compare
000001	Print

Add:

7 6	5 4	3 2	1 0
Op-code	rd	rs	rt
2	2	2	2

Format: Add rd, rs, rt

Purpose: To add 2-bit integers

Description: rd = rs + rt

The 8-bit value in register *rs* is added to the 8-bit value in register *rt* to produce a 8-bit result.

- If the addition results in an integer overflow, the destination register is not modified, an exception occurs.
- If the addition does not result in an overflow, the 8-bit result is placed into register *rd*.

Sub:

7 6	5 4	3 2	1 0
Op-code	rd	rs	rt
2	2	2	2

Format: Sub rd, rs, rt

Purpose: To subtract 8-bit integers

Description: rd = rs – rt

The 8-bit value in register *rt* is subtracted from the 8-bit value in register *rs* to produce an 8-bit result.

- If the subtraction results in an integer overflow, the destination register is not modified, an exception occurs.
- If the subtraction does not result in an overflow, the 8-bit result is placed into register *rd*.

LI:

7 6	5 4	3 0
Op-code	rd	immediate
2	2	4

Format: LI rd, immediate

Purpose: To load an 4-bit immediate value into an 8-bit register

Description: rd = rd + immediate

- A 4-bit *immediate* value is sign extended and placed in the 8-bit register *rd*. The sign extension is performed by using *ori* which puts a constant in the least significant bits of *rd*.
- If an overflow occurs, an exception is thrown

Compare:

7 6	5 4	3 2	1 0
Op-code	rd	rs	rt
2	2	2	2

Format: compare rd, rs, rt

Purpose: Compare two registers. If they are not equal, execute the next instruction. If equal, the choice exists of either skipping either the next 1 or the next 2 instructions

Description: rd = rs and rt

Places the result of using instruction *and* on registers *rs* & *rt* into register *rd*.

Print:

7 2	1 0
Op-code	rd
6	2

Format: print rd

Purpose: Display a registers content to the console.

Description:

Checks the 8-bit value stored in register *rd* and prints the value to the console