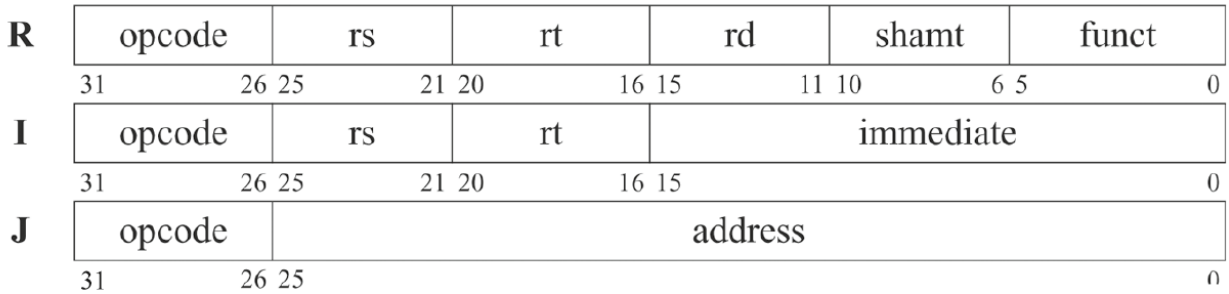
Homework 4 – Pipelined MIPS CPU

Due Dec. 13, 2016

Refer to the MIPS instruction reference data for details on the MIPS instructions and their encoding. MIPS has three instruction formats: R, I, J, as shown below.



Below are some MIPS instructions (add, addi, beq, lw, slt, sw) and their instruction encodings.

CORE INSTRUCTION SET

NAME, MNEMONIC		FOR-MAT	OPERATION (in Verilog)		OPCODE / FUNCT (Hex)
Add	add	R	R[rd] = R[rs] + R[rt]	(1)	$0 / 20_{hex}$
Add Immediate	addi	I	R[rt] = R[rs] + SignExtImm	(1,2)	8_{hex}
Branch On Equal	beq	I	if(R[rs]==R[rt]) PC=PC+4+BranchAddr	(4)	4_{hex}
Load Word	lw	I	R[rt] = M[R[rs]+SignExtImm]	(2)	23_{hex}
Set Less Than	slt	R	R[rd] = (R[rs] < R[rt]) ? 1 : 0		$0 / 2a_{hex}$
Store Word	sw	I	M[R[rs]+SignExtImm] = R[rt]	(2)	$2b_{hex}$

The following shows examples of machine codes for some MIPS instructions.

MIPS machine language

Name	Format	Example						Comments
add	R	0	18	19	17	0	32	add $s1,$s2,$s3
sub	R	0	18	19	17	0	34	sub $s1,$s2,$s3
addi	I	8	18	17	100			addi $s1,$s2,100
lw	I	35	18	17	100			lw $s1,100($s2)
sw	I	43	18	17	100			sw $s1,100($s2)
Field size		6 bits	5 bits	5 bits	5 bits	5 bits	6 bits	All MIPS instructions are 32 bits long
R-format	R	op	rs	rt	rd	shamt	funct	Arithmetic instruction format
I-format	I	op	rs	rt	address			Data transfer format

1. Use Verilog to model the standard 5-stage MIPS pipeline (IF, ID, EX, ME, WB) for the above MIPS instructions.
2. Write testbench to verify the RTL code.
3. Use Synopsys Design Compiler to synthesize the RTL code into gate-level netlist based on TSMC 90nm standard cell library.
4. Re-simulate the synthesized gate-level netlist using the same testbench as in the RTL simulation.
5. Identify the critical path and determine the maximum working frequency.

6. (optional for undergraduate) Redo the synthesis and gate-level simulation using FPGA tools.

References:

1. D. A. Patterson and J. L. Hennessy, *Computer Organization and Design, The Hardware/Software Interfac*, , 5th ed., Elsevier, 2013.