

2017 Architecture

# Project 1 – Instruction simulator



# Outline

- ▣ Workstation setting : MobaXterm
- ▣ Project

# Outline

- ▣ Workstation setting : MobaXterm
  - Please check “ project\_setup.pdf ”
- ▣ Project

# Outline

- ▣ Workstation setting : MobaXterm
- ▣ Project

# Project Goal

- ▣ Implement chosen instructions
- ▣ We provide you:
  - **Simulator.v**
    - ▣ Already a prototype
  - **Test\_Bench.v**
    - ▣ Read instruction from testcase
    - ▣ Generate clk/rst signal and simulator instance
    - ▣ Print out simulator's register value
  - **3 example testcase files**

# How to run simulation

- ▣ \$ ncverilog *all .v files you need*
  - i.e. \$ncverilog Simulator.v Test\_Bench.v

```
2. nthucad.cs.nthu.edu.tw
Initial blocks:      2      2
Pseudo assignments: 1      1
Simulation timescale: lps
Writing initial simulation snapshot: worklib.Test_Bench.v
Loading snapshot worklib.Test_Bench.v ..... Done
*Verdi3* Loading libsscore_ius141.so
*Verdi3* : Enable Parallel Dumping.
ncsim> source /usr/cad/cadence/INCISIV/cur/tools/inca/files/ncsimrc
ncsim> run
$0: 0
$1: 0
$2: 0
$3: 0
$4: 0
$5: 0
$6: 0
$7: 0
$8: 0
$9: 0
$10: 0
$11: 0
$12: 0
$13: 0
$14: 0
$15: 0
$16: 0
$17: 0
$18: 0
$19: 0
$20: 0
$21: 0
$22: 0
$23: 0
$24: 0
$25: 0
$26: 0
$27: 0
$28: 0
$29: 0
$30: 0
$31: 0
Simulation complete via $finish(1) at time 2010 NS + 0
./Test_Bench.v:45      #(`CYCLE_TIME*`END_CYCLE) $finish;
ncsim> exit
```

# R-type instruction

Instruction	Example	Meaning	Op field	Function field
<b>ADD</b> (Addition)	add r1, r2, r3	$r1 = r2 + r3$	0x00	32(0x20)
<b>SUB</b> (Subtraction)	sub r1, r2, r3	$r1 = r2 - r3$	0x00	34(0x22)
<b>AND</b> (Logic And)	and r1, r2, r3	$r1 = r2 \& r3$	0x00	36(0x24)
<b>OR</b> (Logic Or)	or r1, r2, r3	$r1 = r2   r3$	0x00	37(0x25)
<b>SLT</b> (Set on Less Than) signed comparison	slt r1, r2, r3	if ( $r2 < r3$ ) $r1 = 1$ else $r1 = 0$	0x00	42(0x2a)

# I-type instruction

Instruction	Example	Meaning	Op field
<b>ADDI</b> (Add Immediate)	addi r1, r2, 100	$r1 = r2 + 100$	0x08
<b>LW</b> (Load word)	lw r1, 0(s1)	R1 = 4 byte from Mem[s1+0]	0x23
<b>SW</b> (Store word)	sw r1, 0(s1)	4 byte from Mem[s1+0] = r1	0x2B
<b>SLTI</b> (Set on Less Than Immediate)	slti r1, r2, 10	if( $r2 < 10$ ) $r1 = 1$ else $r1 = 0$	0x0A
<b>BEQ</b> (Branch On Equal)	beq r1, r2, 25	if ( $r1 == r2$ ) go to PC+4+100	0x04

Immediate is signed for these instructions



# Initial status

- Data memory and register should be set to 0 at the beginning
- PC starts with 0
- Be able to reset program by rst\_i

# Hand in your project

- ▣ Please upload all your .v file to iLMS
  - No file I/O
  - No clk and rst setting
  - You don't have to upload Test\_Bench.v
  - **DO NOT** compress your file
  
- ▣ There shouldn't be \$stop in your code

**Violation: -30 points**

# Notice

- ▣ Don't modify name, usage, size
  - Top module Simulator.v Name: Simulator
  - Register file : [32-1:0] Reg\_File [0:32-1]
    - ▣ Store value of register
    - ▣ Simulator.Reg\_File
  - Instruction memory: [32-1:0] Instr\_Mem [0:256-1]
    - ▣ Store value of instruction memory
    - ▣ Simulator.Instr\_Mem
  - Data memory: [32-1:0] Data\_Mem [0:256-1]

**Violation: -30 points**

# Notice

- ❑ Copy or you let other copy **Get 0 point**
- ❑ Discussion is encouraged
  - You have to do your own work
- ❑ Take good care of your code
  - Some online storage space might not be safe

# Test case

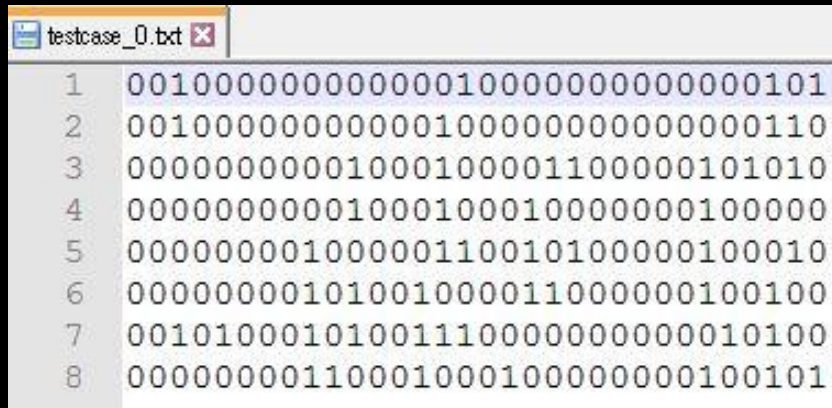
## ■ 3 types of test cases

Type	It will cover
1	ADD, SUB, AND, OR, SLT, ADDI, SLTI
2	type 1 + BEQ
3	type2 + LW,SW

# Example testcase

## ■ Testcase 0

- Format : Each line = one assembly code



```
testcase_0.txt
1 001000000000000010000000000000101
2 001000000000000010000000000000110
3 000000000001000100001100000101010
4 000000000001000100010000000100000
5 000000000100000110010100000100010
6 000000000101001000011000000100100
7 00101000101001110000000000010100
8 000000000110001000100000000100101
```

### Testcase\_0.txt

```
addi $1, $0, 5
addi $2, $0, 6
slt $3, $1, $2
add $4, $1, $2
sub $5, $4, $3
and $6, $5, $4
slti $7, $5, 20
or $8, $6, $4
```

### Result

```
$1=5,$2=6,$3=1,
$4=11,$5=10,$6=10
$7=1,$8=11
```

# Example testcase

## Testcase\_1.txt

```
addi $1, $0, 19
addi $2, $0, 8
beq $0, $0, 2
slt $3, $2, $1
add $4, $1, $2
sub $5, $4, $3
and $6, $5, $4
slti $7, $5, 20
or $8, $1, $2
```

## Result

\$1=19,\$2=8,\$3=0,  
\$4=0,\$5=0,\$6=0  
\$7=1,\$8=27

## Testcase\_2.txt

```
addi $1, $0, 19
addi $2, $0, 8
sw $1, 0($10)
beq $0, $0, 1
slt $3, $2, $1
add $4, $1, $2
sub $5, $4, $3
beq $2, $0, 3
and $6, $5, $4
slti $7, $5, 20
lw $8, 0($10)
or $9, $1, $3
addi $1, $1, -4
```

## Result

\$1=15,\$2=8,\$3=0,  
\$4=27,\$5=27,\$6=27  
\$7=0,\$8=19,\$9=19

# Deadline

- Deadline : 4/3(Mon), 23:59
- No late submission is allowed