CS 343, Spring 2022

# Laboratory Project

# BEQ, BNE,J MIPS OPERATIONS -GRADED!

*Instructor: Professor Izidor Gertner*

**April 11, 2022, Due April 24, 2022 by 11:00 PM**

## Objective:

- **Learn importance of comparator lab and important use of REGISTER File, Registers**
- **Learn MIPS instructions where EQUAL operation is used**
- **Design and implement in VHDL MIPS instructions: BEQ, BNE, J**

- Branch to a labeled instruction if a condition is true
  - Otherwise, continue sequentially
- `beq rs, rt, L1`
  - if (rs == rt) branch to instruction labeled L1;
- `bne rs, rt, L1`
  - if (rs != rt) branch to instruction labeled L1;
- `j L1`
  - unconditional jump to instruction labeled L1

## 1. Design and verify BEQ instruction, I Type format

- `You will also need` **PC**`- Program Counter 32 bit register`

```
beq   rs, rt, imm16
```

Please refer to green pages in the textbook for more information

You will need to add another register-Instruction Pointer which is named **PC**

```
Equal <= (R[rs] == R[rt])     Calculate the branch condition
if (Equal)                    Calculate the next instruction's address
        PC  <=  PC + 4 + { SignExt(imm16) , 2b00 }
Else  PC  <=  PC + 4
```

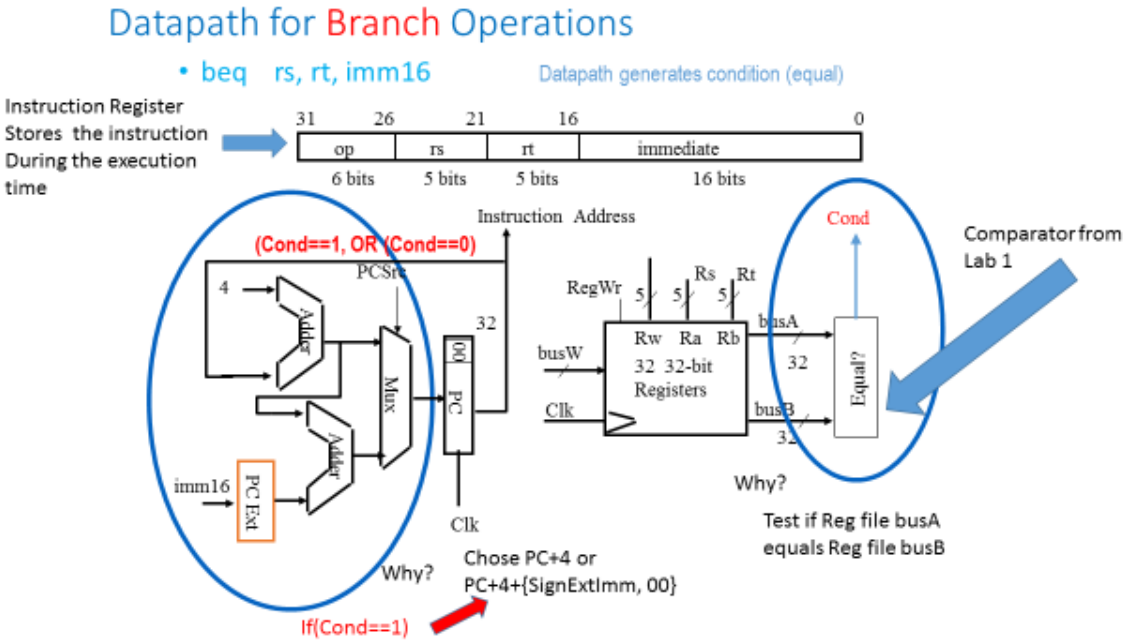
CS 343, Spring 2022

# Laboratory Project

# BEQ, BNE,J MIPS OPERATIONS -GRADED!

*Instructor: Professor Izidor Gertner*

**April 11, 2022, Due April 24, 2022 by 11:00 PM**




In this lab you will to need to integrate ***Comparator lab*** and ***Registers from Latches-FlipFlops, Midterm Memory lab.***

**Part A.**

***Design and implement the following registers:***

***1. 16 bit offset stored in IMMEDIATE FIELD of the BEQ instruction in INSTRUCTION REGISTER***

***2. 32 bit register RS to store integer as operand 1 is in REGISTER FILE, The index is in IR***

***3. 32 bit register RT to store integer as oprand 2 is in REGISTER FILE, The index is in IR***

***4. 32 bit program counter register PC (EIP on I7) to store the address of the next instruction.***

NOTE: You need to use INSTRUCTION REGISTER IR and REGISTER FILE in this lab!. You can implement Register File as an VHDL array or using LPM.

CS 343, Spring 2022

# Laboratory Project

# BEQ, BNE,J MIPS OPERATIONS -GRADED!

*Instructor: Professor Izidor Gertner*

**April 11, 2022, Due April 24, 2022 by 11:00 PM**

***Operations to be performed:***

1. Unit ***Equal***

***( is your comparator lab modified to accept to operands stored in 32 bit registers RS and RT).***

The output of *Unit Equal* is signal Cond. Cond <=1 if RS==RT else Cond <=0;

2. Next Address Logic. Unit (NAL)

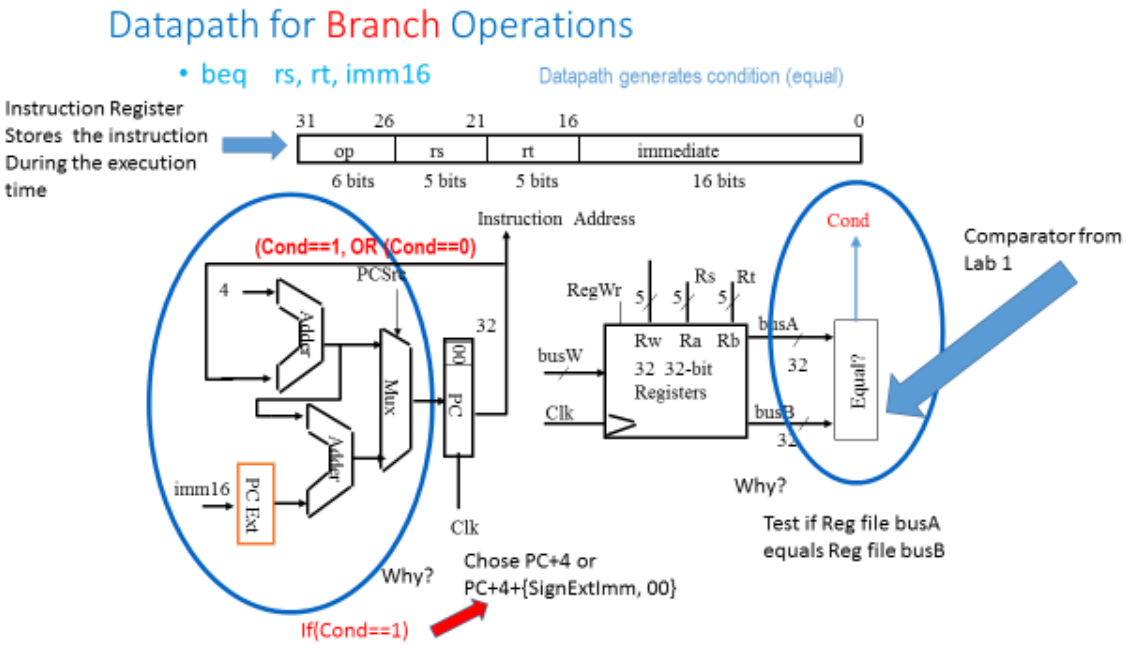
Operation of NAL Unit.

2.1 If **Cond =0 then PC<=PC + 4**

Computes the address of the next instruction by adding 4 to PC register ( since instruction length is fixed 4 bytes)

2.2 If **Cond =1 then PC<=PC + 4+ sign extended imm16.**

2.3 Use 2:1 MUX to control 2.1 and 2.2.



CS 343, Spring 2022

# Laboratory Project

# BEQ, BNE,J MIPS OPERATIONS -GRADED!

*Instructor: Professor Izidor Gertner*

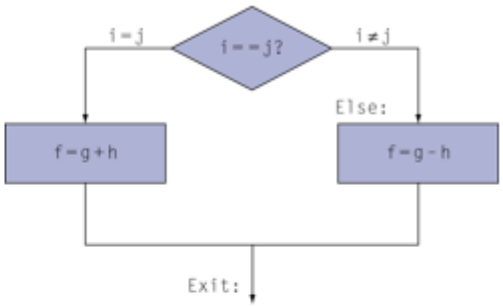
**April 11, 2022, Due April 24, 2022 by 11:00 PM**

Another application of BEQ, BNE, and J instructions.

- C code:

```
if (i==j) f = g+h;
else f = g-h;
```

  - f, g, … in $s0, $s1, …




- Compiled MIPS code:

```
        bne $s3, $s4, Else
        add $s0, $s1, $s2
        j   Exit
Else:   sub $s0, $s1, $s2
Exit:   …
```

Assembler calculates addresses

3. Implement all independent instructions shown above: bne, add, J , sub.

What to submit

1. Detailed report with verification in ModelSim Using waveforms. Write test bench vhdl code to verify your design.
2. Make a 2 min presentation on your design , in the form of 2 min video. Make sure your title page includes all course information including your talking face video for identification.

3. Have archived project files submitted to me. Please have prepared READMEFIRST file included on how to use your software. YOUR LAST NAME IS ALWAYS IN FIRST PLACE.