

# Computer Architecture

## MIPS Design HW Final

### 목표

- Verilog를 이용하여 Pipelined MIPS Datapath를 구현합니다.
- 8가지(Add, Sub, And, Or, Slt, Sw, Lw, Beq)의 Instruction을 구현 및 검증합니다.
- 주어진 Instruction Memory와 Data Memory를 알맞게 연결 후, 출력 값을 확인합니다.

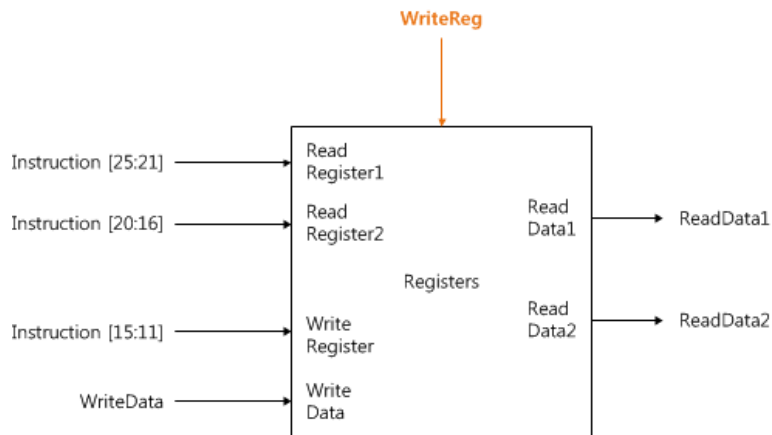
### 할일

#### - Fetch Stage

- 32Bits PC Register, 32Bits 2to1 Mux, 32Bits Adder를 구현합니다.
- 구현한 Module과 주어진 Instruction Memory를 알맞게 연결합니다.

#### - Decode Stage

- SignExtension을 구현하고, 주어진 Registers Module과 알맞게 연결합니다.
- Registers의 Input은 그림과 같이 넣습니다.  
단, Pipeline의 경우 WriteReg와 WriteRegister, WriteData는 ReadRegister1, ReadRegister2와 같은 시기의 Instruction이 아닙니다.



- SignExtension의 Input은 Instruction의 하위 16Bits이고, Output은 Sign Extend된 32Bits 값입니다.
- **Execute Stage**
  - 32Bits ALU, 32Bits 2to1 Mux, 32Bits Adder, 5Bits 2to1 Mux, ShiftLeft2, ALU Control를 구현합니다.
  - 32Bits ALU는 최소 5가지 연산(Add, Sub, And, Or, Slt)이 가능해야 합니다.  
또한 Output Zero는 Result값이 0일 때, 1입니다.  
ALU 같은 경우는 Control신호에 의해 다른 출력을 가지므로 Case문을 추천합니다.
  - ShiftLeft2은 Input을 왼쪽으로 2Bits Shift한 값을 Output으로 가집니다.
  - ALU Control은 Function Code를 가지고, ALU를 Control합니다. (Tip : case문 사용)

- **Memory Stage**
  - Branch 신호와 ALU에서 나온 Zero를 And Gate를 사용하여 묶어주어야 합니다.
  - 주어진 DataMemory를 가지고, 이전 Stage에서 온 신호를 잘 연결합니다.
  - Write Register Address도 온전히 전달하여 줍니다.
  - DataMemory의 TestPort를 밖으로 빼내는 것을 고려합니다.
- **Write Back Stage**
  - 32Bits 2to1 Mux를 이용하여 Memory Data와 ALU Result를 연결합니다.
- **Control Unit**
  - OpCode를 가지고, 알맞은 Control Signal을 생성하여 주고, 이때, Combinational Circuit으로 만듭니다. (Tip : case문 사용, always문 clk X)
  - Control Signal는 WriteReg, MemToReg, Branch, ReadMem, WriteMem, DstReg, ALUSrc, ALU\_OP[1:0]입니다.
- **Pipeline**
  - MIPS Datapath 그림을 보면, 각 Stage마다 세로의 긴 사각형이 보입니다. 그것이 Pipeline Register(FlipFlop)입니다.
  - Control Signal에 대한 Register도 고려합니다.
- **TestBench**
  - 주어진 tb\_MIPS\_Datapath을 가지고, Timing Graph를 뽑아보면, cnt값 100을 전후로 TestPort로 캡처된 것과 같은 값이 출력되어야 합니다.

## 평가

- 주어진 InstructionMemory와 비슷한 방식의 다른 InstructionMemory를 사용하여 평가합니다.
- TestPort를 보고 평가하기 때문에, TestPort 출력에 유의해주시기 바랍니다.
- 완성에 대한 점수는 이전에 공지한 것과 같습니다.
- 구체적인 디자인 없이 결과만을 위한 값 출력은 0점입니다.  
(Ex : Initial 구문을 이용한 결과값 도출)
- 동일한 소스 발견 시, 동일 소스 제출자 모두 0점이니 유의하시기 바랍니다.

## 제출

- 제출날짜는 6월 19일까지입니다.
- 각각의 모듈파일(\*.v)과 tb파일을 E-Mail로 제출합니다.  
(빠진 모듈의 있을 경우, 채점 때 불이익이 있을 수 있습니다. 꼭 확인하시기 바랍니다.)
- 메일 제목은 "[MIPS]\_학번\_이름" 으로 보내주셔야 필터링이 동작합니다.  
(Ex : "[MIPS]\_2006000000\_송진우")  
제목은 지켜주지 않으면, 메일을 확인하지 못할 수 있으니 유의해주시기 바랍니다.
- E-Mail : bluerose7112@gmail.com