

204225 สถาปัตยกรรมคอมพิวเตอร์

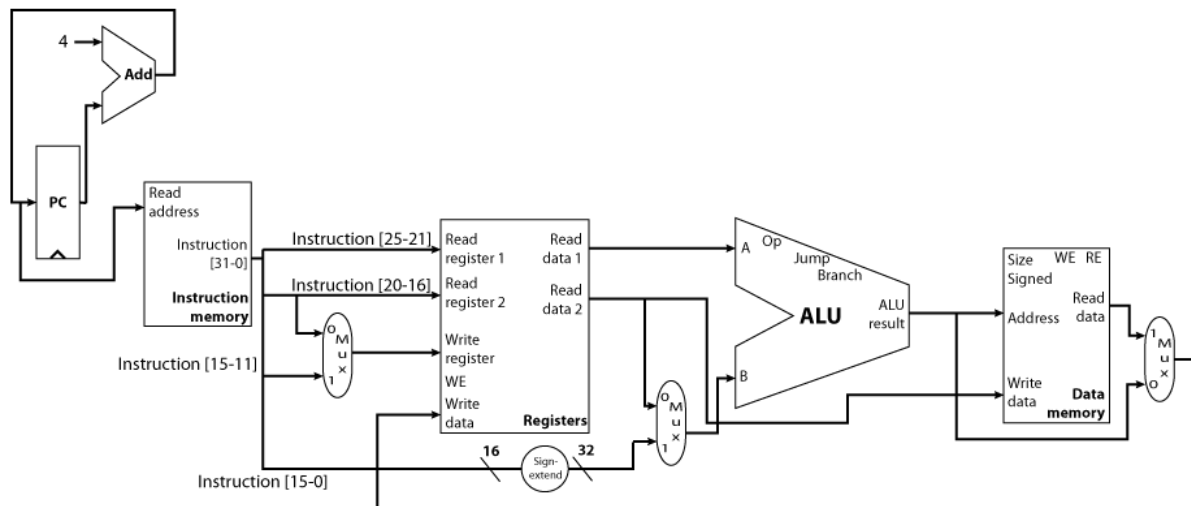
โครงการกลุ่มครั้งที่ 2

กำหนดส่ง 15 พฤษภาคม 2558 (ก่อนเวลา 23.50 น.)

Single-Cycle MIPS Control

ในครั้งนี้ จะให้จับกลุ่มเป็นกลุ่มละ 2 คน

implement จำลองการทำงานของ MIPS CPU ที่มีโครงสร้างดังภาพ ด้วย verilog



แต่ทั้งนี้ ได้แต่มีการ implement แต่ละ component ของ CPU ไว้เรียบร้อยแล้ว ยังขาดเพียง 6 ส่วน คือ

A register file (ไฟล์ register.v)

A Program Counter (PC) register (เป็น register หนึ่งใน processor)

A 2-input mux parameterized to support multiple widths (ไฟล์ mux.v)

An adder (ไฟล์ adder.v)

A sign-extender (ไฟล์ signextender.v)

A controller (ไฟล์ controller.v)

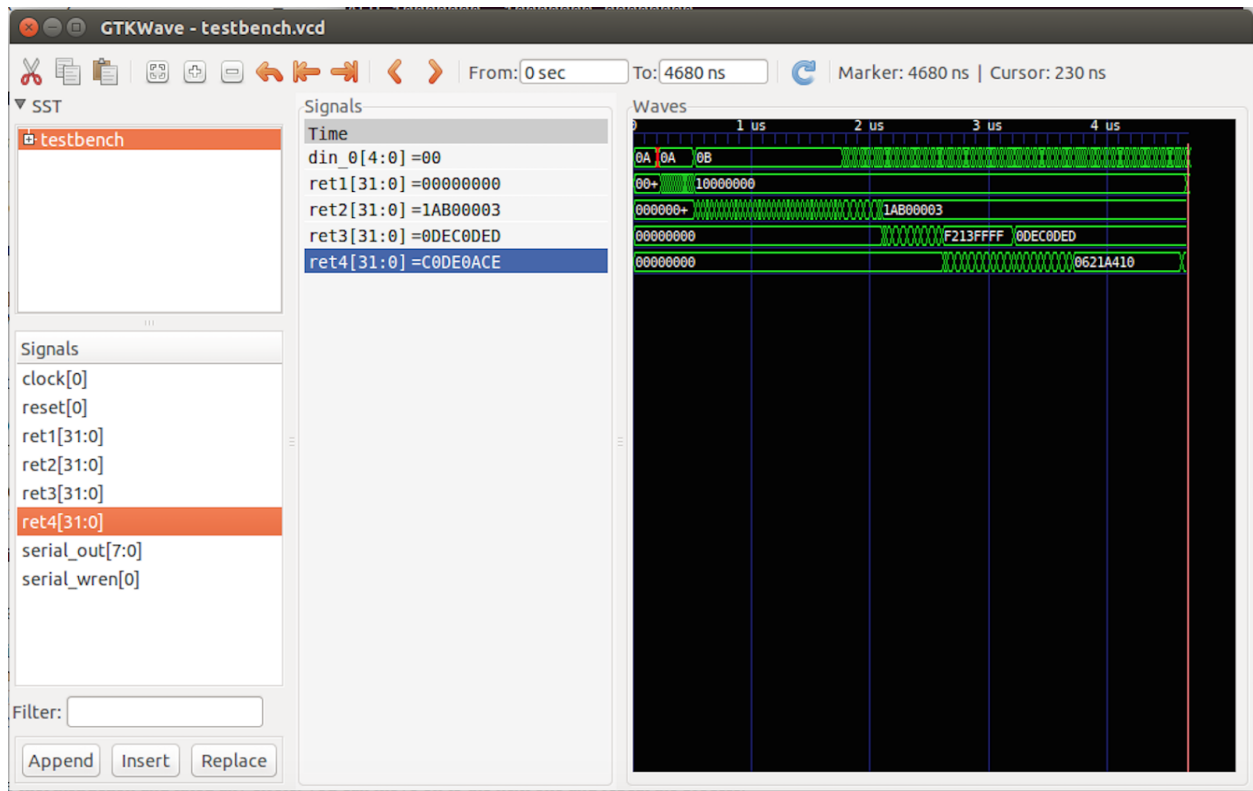
โดย จะเขียนต่อ component ทั้งหมดเข้าด้วยกันในไฟล์ processor.v

และ เขียน control เพื่อใช้ในการควบคุม path ของคำสั่งแต่ละคำสั่ง ในไฟล์ controller.v

โดยทั้ง inst_rom.v และ data_memory.v จะประกอบไปด้วย instruction และ data ซึ่งให้ ข้อมูล instruction ในทุกๆ clock

แนวทางการทดสอบ

เป้าหมาย เพื่อให้ สามารถรันไฟล์ testbench.v ได้ แล้วได้ผลลัพธ์สุดท้าย ดังรูป



ซึ่งโค้ด MIPS ที่จะรันนี้สามารถศึกษาได้ที่ไฟล์ test/lab3-test.dis และ test/ lab3-test.spim.s

**** เพิ่มเติม**

จะต้อง initial Memory ก่อน เพื่อโหลด MIPS เข้า memory โดย

- เปลี่ยน path ในไฟล์ data_memory.v ในบรรทัดที่ 45-48 ให้เป็น absolute path ไปยัง ไฟล์ data_ramX.memh ที่แนบไปใน folder test

- เปลี่ยน path ในไฟล์ inst_rom.v ในบรรทัดที่ 22 ให้เป็น absolute path ไปยัง ไฟล์ inst_rom.memh ที่แนบไปใน folder test

วิธีการรันคำสั่ง ใน Ubuntu

```
iverilog testbench.v -o processor  
vvp processor -lxt2  
gtkwave testbench.vcd  
rm processor
```

กำหนด format ของ module processor ดังนี้

```
processor(  
    clock,  
    reset,  
    serial_in,  
    serial_valid_in,  
    serial_ready_in,  
    serial_rden_out,  
    serial_out,s  
    serial_wren_out  
);
```

โดย ไม่จำเป็นต้องสนใจ serial*

หากมีข้อสงสัย สามารถส่งคำถามมาได้ ที่ facebook Pakawat Nakwijit และ Wipada Wannasiwaporn
หรือ เมลล์ kramatk@gmail.com, global.wipada@gmail.com