1 Essä 8 poäng ISA

Write the MIPS assembly code for a function that looks through an array of integers and returns the smallest integer value in the array.

The function should take two arguments: memory address of beginning of array, number of integers in array.
Function should return one value: minimum integer in the array.

1. **[3 points]** Write a pseudo-code (Or code in C, python, or other high level languages) for the function.
2. **[5 points]** Write the function using MIPS assembly.

2 Essä 6 poäng ISA-2

In this problem make sure your code from the previous question follows the register calling convention.

1. **[3 points]** Make sure to use the correct registers for the arguments and the return values. List the registers in your previous answer that are used for the arguments and return values. Then, list which registers you would use to conform to the register calling convention.
   (If you used the conforming registers from the beginning, then just list the registers that are used for arguments and return).
2. **[3 points]** Correctly save/restore registers for the temporary registers in your function. (Remember that question 1 requires you to write a 'function'). If you don't need to have any save/restoring of registers, state why.

3 Essä 9 poäng ISA

Assume we want to add a new branch equal immediate instruction (beqi). This instruction would compare the value stored in a register with an immediate value. If the comparison results in equality, the CPU will branch.

Assume that we have a 32-register ISA (MIPS has 32-registers), and the 6-bits are reserved for the opcode, and the instructions are always 32-bits wide.

1. **[3 points]** Before going on to the beqi instruction, first list the components of the beq instruction. How are the 32-bits of the beq instruction divided up?
2. **[3 points]** Now, how would the 32 bits of the beqi instruction be divided up? How many bits will be available for the branch offset + immediate?
3. **[3 points]** Assume that we want to use 4 bits for the immediate. How many bits would be available for the branch offset? How many instructions would these bits allow us to jump (Answer will be a range [min, max])?

4 Flera val 2 poäng Arithmetic

Why do we use two's compliment notation?

○ There is only one zero

○ Subtraction is just invert addition with the carry in set to 1

○ We can easily determine if the number is negative by looking at the MSB

● All of the above

5 Flera val 2 poäng Arithmetic

What is the largest positive value you can represent with a 6-bit (-1)^S*(0.FF)*(2^EEE) floating point format, where FF is unsigned and EEE is two's complement?

● 6

○ 14

○ 96

○ 224

6 Flera val 2 poäng Arithmetic

When adding a positive and a negative two's complement number, can you cause an overflow?

○ No, because addition of two's complement values cannot an overflow

● No, because addition of opposite signed two's complement values cannot cause an overflow

○ Yes, if the positive number is large enough

○ Yes, if the negative number is large enough

7 Flera val 2 poäng Logic

What kinds of circuits are needed for a counter?

○ Combinational

○ State register

○ Full adder

● All of the above

8 Flera val 2 poäng Logic

Why is DRAM so much cheaper than SRAM?

- ○ DRAM cells are made of two inverters so they are smaller
- ● DRAM cells are made of a single capacitor so they are smaller
- ○ DRAM cells are faster so you need fewer of them
- ○ DRAM cells are larger so they are easier to make

9 Flera val 2 poäng Logic

What does a latch do?

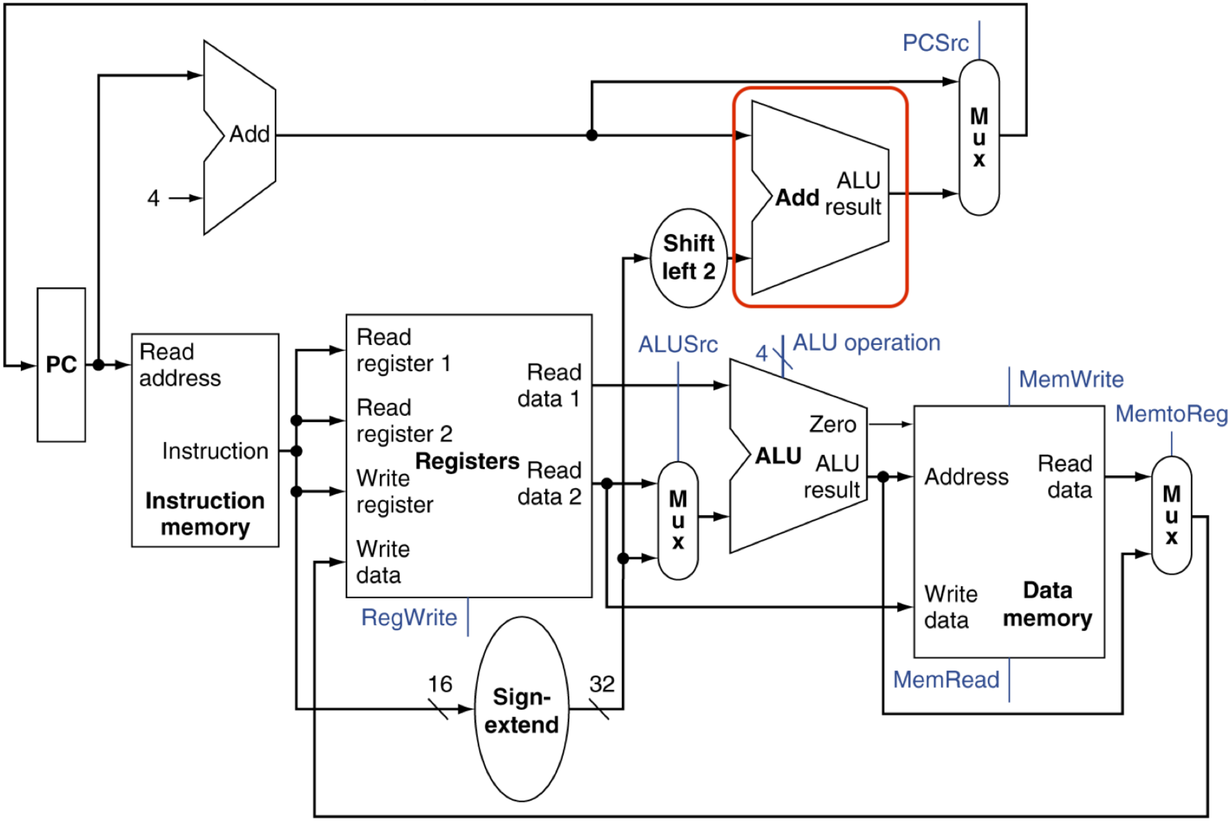
- ○ Store a value
- ○ Update the stored value on the clock signal
- ○ Break the feedback loop in a sequential circuit
- ● All of the above

10 Essä 9 poäng Datapath

1. **[5 points]** Explain what each of the following components do in the datapath (one sentence per component)

- PC
- Instruction memory
- Registers
- ALU
- Data Memory

2. **[4 points]** Explain when the Adder in the red box of the image is used. In the case where we need to use the Adder? Why cannot we use the ALU to do the addition?



11 Essä 7 poäng Pipelining

A single-cycle processor takes 50ns for each instruction. The processor can be pipelined up to 5 stages, and takes 11ns for each stage. Furthermore, the pipeline register adds another 1ns per stage.

1. **[2 points]** Calculate the latency (time to complete one instruction) of the pipelined processor.
2. **[3 points]** Calculate the throughput (how many instructions complete in a given number of cycles) of a single-cycle processor, and a 5 stage pipelined processor in a 500ns window.
3. **[2 points]** Based on your prior two answers, what is the advantage and disadvantage of the two types of processors (single-cycle vs. pipelined).

12 Essä 8 poäng Pipeline Hazards

Pipelining improves CPU throughput (as you demonstrated from the above), but it also introduces many problems. The three problems covered in the course were:

1. Structural Hazard
2. Data Hazard
3. Control Hazard

Pick two hazards and explain the problem (with an example) and a solution for each hazard type.

13 Essä 8 poäng Branch prediction

1. **[5 points]** Assuming the 5-stage pipelined processor we learned in the course, which stage would you place the branch predictor? Justify your answer. Make sure to include how many bubbles or nops cycles are required on a branch prediction hit.
2. **[3 points]** How would you connect the branch predictor to the datapath (Where would the output of your branch predictor send the data to? What input would your branch predictor take?)

14 Flera val 2 poäng Branch Prediction

Why do we need a branch target buffer (BTB)?

○ To predict whether the branch is taken or not

○ To keep track of which branch the prediction is associated with

● To avoid having to wait for the branch address calculation

○ To determine if the prediction is valid

15 Essä 8 poäng I/O

1. **[5 points]** Once the disk reads the requested data from the underlying media (HDD or SSD), the data must be copied onto the main memory before the data can be processed by the CPU. Explain how you could use DMA and interrupts to copy the data into main memory. How can using DMA improve the system performance?
2. **[3 points]** With I/O, what is the benefit of polling over interrupt? What is the benefit of interrupt over polling?

16 Essä 10 poäng Caches

1. **[6 points]** Give an example of why a direct-mapped cache can result in conflicts. What is the alternative, and how does it overcome the conflicts of the direct-mapped cache (demonstrate that the alternative overcomes the problem the direct-mapped cache had with the same example).
2. **[4 points]** What is the benefit of having a smaller cache block size? What is the disadvantage of having a smaller cache block size?

17 Essä 7 poäng Virtual Memory

1. **[3 points]** What is the benefit of introducing virtual memory? List at least three benefits that VM provides.
2. **[4 points]** How does the OS and HW keep track of which virtual address maps to which physical address? What data structure does the OS use? How does the HW use the OS data structure to translate from virtual to physical memory?

18 Flera val 2 poäng Parallelism

Which was **not** an example of using parallelism to improve performance?

○ Double-pumped register file access

○ Pipelining

○ VIPT

● Moving the branch calculation earlier

19 Flera val 2 poäng Parallelism

A program takes 100s to run on a single CPU and 50% of it can be run in parallel. A multicore CPU has 4 processors. How long will it take to run on the multicore CPU?

○ 25s

○ 37.5s

○ 50s

● 62.5s

20 Flera val 2 poäng Parallelism

What do modern processors have to be able to do to execute more than 1 instruction per cycle?

○ Find independent instructions

○ Have multiple functional units (e.g., ALUs) to execute multiple instructions at the same time

○ Load multiple instructions at the same time

● All of the above