Version AHG/2

EGT3
ENGINEERING TRIPOS PART IIB

Friday 5 May 2023 9.30 to 11.10

Module 4F14

COMPUTER SYSTEMS

*Answer not more than **two** questions.*

All questions carry the same number of marks.

*The **approximate** percentage of marks allocated to each part of a question is indicated in the right margin.*

*Write your candidate number **not** your name on the cover sheet.*

STATIONERY REQUIREMENTS
Single-sided script paper

SPECIAL REQUIREMENTS TO BE SUPPLIED FOR THIS EXAM
CUED approved calculator allowed
Engineering Data Book

10 minutes reading time is allowed for this paper at the start of the exam.

You may not start to read the questions printed on the subsequent pages of this question paper until instructed to do so.

You may not remove any stationery from the Examination Room.

1 (a) Distinguish between *accumulator* and *general purpose register, load-store* instruction set architectures. Illustrate your answer by showing, for each type of architecture, how two numbers in memory can be summed and the result stored back into memory. [20%]

(b) How many bits are used for the address fields in MIPS branch, jump and load/store instructions? What are the practical implications of these finite field widths? [40%]

(c) Consider a C++ program **A** which calls some functions in a standard library **B**. **A** and **B** were compiled independently. A program called a *linker* combines the machine code of **A** and **B** to produce an executable **C**. It then happens that **C** executes concurrently with some other program **D**. How might the linker and other elements of the computer system ensure correct execution of **C** and **D**, with no memory conflicts or unresolved references? [40%]

2 (a) In the context of a pipelined datapath, what is meant by a branch hazard? How might branch hazards be resolved? [20%]

(b) Figure 1 shows the standard pipelined MIPS datapath while Fig. 2 shows a modified design of the ID stage enabling a delayed branch with one delay slot and no flushing or stalling whether the branch is taken or not. Explain why the speed of the comparator (labelled "=" in Fig. 2) is critical. How could a fast comparator be implemented in practice? [20%]

(c) Write some MIPS code to copy either `$8` or `$9` into `$10`, as follows. If `$8<$9` copy `$8`, otherwise copy `$9`. Use the `slt` instruction

```
slt $11,$8,$9     # if ($8<$9) $11=1; else $11=0
```

in conjunction with a conditional branch. Optimize the code for the single branch delay slot pipeline in Fig. 2. [20%]

(d) `movn` and `movz` are instructions available in some MIPS extensions. For example

```
movz $10,$9,$11
```

copies the contents of `$9` into `$10` if the contents of `$11` are zero, otherwise it does nothing. `movn` is similar but copies if `$11` is not zero. Repeat part (c) of this question using the `movn` and/or `movz` instructions. What advantages do these instructions offer over conditional branches (i) in this example and (ii) more generally with pipelined datapaths? [20%]

(e) Explain how the `movz` instruction could be formatted in 32 bits, without breaking the regularity of the MIPS instruction set. List carefully all the changes you would need to make to the datapath and control in Fig. 1 to accommodate the `movn` instruction. You may introduce only multiplexors, connections and control signals. You may also assume that the register file can be modified to output the contents of three registers simultaneously. [20%]

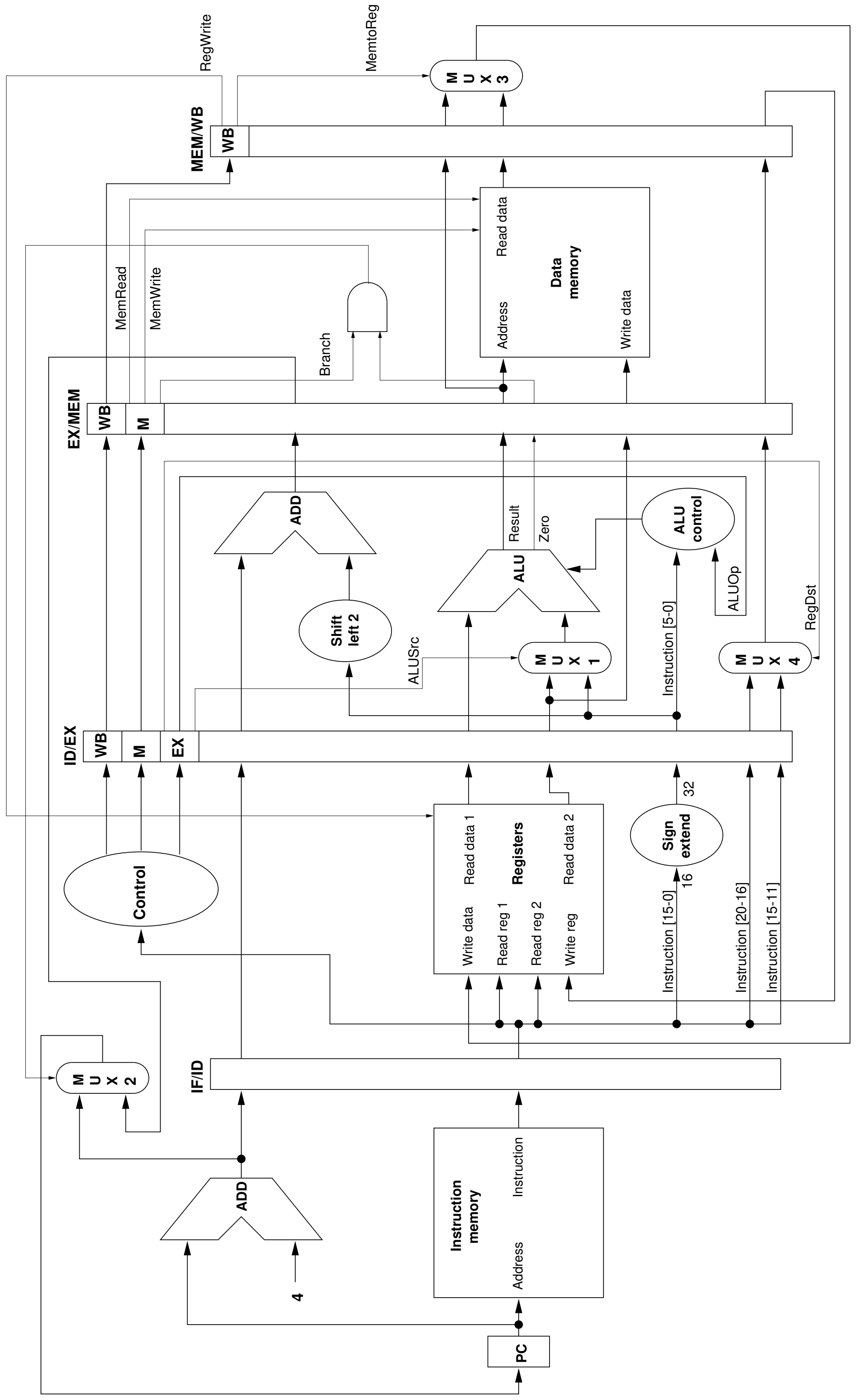
RegWrite
MemtoReg
M U X 3
MEM/WB
WB
Read data
Data memory
Address
Write data
MemRead
MemWrite
Branch
EX/MEM
WB
M
ADD
Result
Zero
ALU
ALU control
Instruction [5-0]
ALUOp
RegDst
M U X 4
Shift left 2
ALUSrc
M U X 1
ID/EX
WB
M
EX
32
Sign extend
16
Read data 1
Registers
Read data 2
Write data
Read reg 1
Read reg 2
Write reg
Control
Instruction [15-0]
Instruction [20-16]
Instruction [15-11]
IF/ID
M U X 2
ADD
4
Instruction memory
Instruction
Address
PC

Fig. 1

(cont.

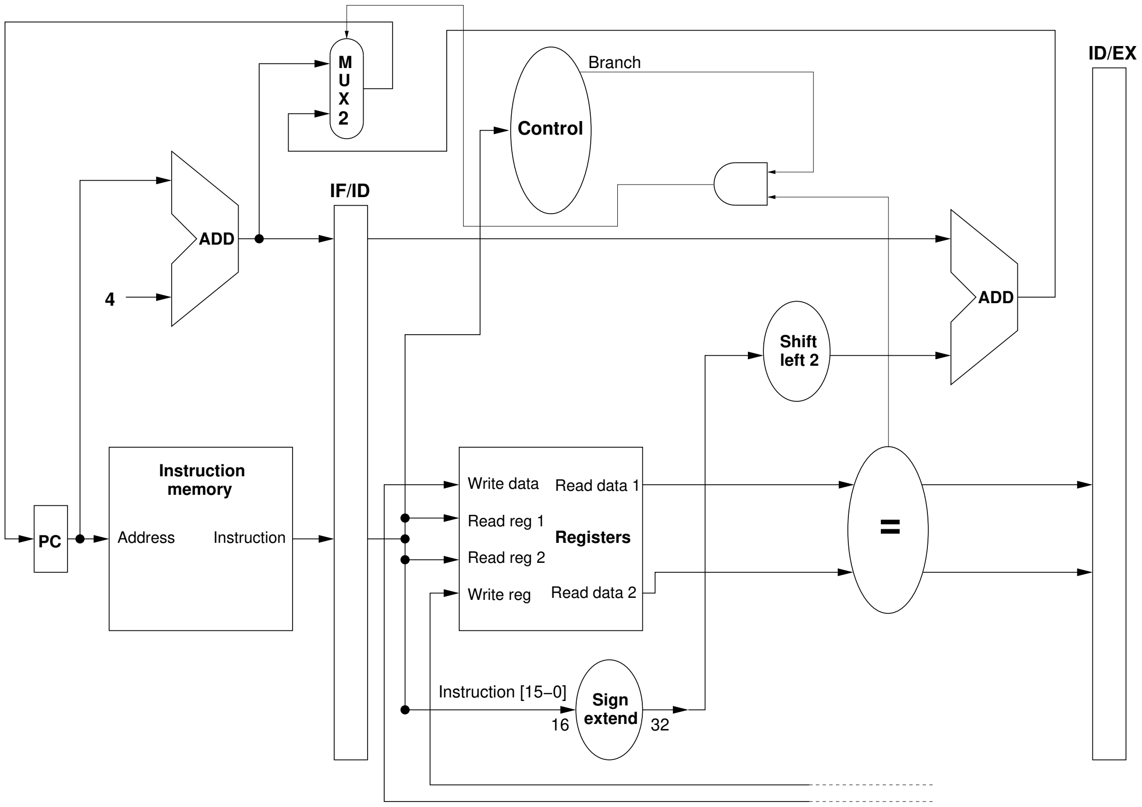
M
U
X
2
Branch
ID/EX
Control
IF/ID
ADD
4
ADD
Shift
left 2
Instruction
memory
PC
Address
Instruction
Write data
Read reg 1
Read reg 2
Write reg
Read data 1
Registers
Read data 2
=
Instruction [15–0]
Sign
extend
16
32

Fig. 2

3 (a) What are the reasons behind the general trend towards replacing parallel I/O buses with serial point-to-point networks? Give one specific example of such a transition in consumer PC hardware. [25%]

(b) Discuss the relative advantages and disadvantages of polling, interrupt-driven I/O and direct memory access (DMA). For each method, suggest one common I/O task for which it might be used. [25%]

(c) A process reads a large file from disk into memory. The operating system controls the disk through a device driver that uses DMA for the transfer. The driver allocates a portion of memory as follows

`status`	`buffer`

where `status` is an integer used to control and monitor the DMA operation, and `buffer` is the region of memory into which the disk file is transferred. The driver periodically reads and writes `status` during the DMA operation. The cache coherency protocol involves selectively flushing or invalidating cache blocks at the start of the DMA operation. Assuming a write-back cache, explain carefully why the coherency protocol fails if `status` and the first few words of `buffer` share a cache block. Suggest ways to solve this problem. [50%]

END OF PAPER

Part IIB 2023

Module 4F14: Computer Systems

Numerical Answers

1. (b) 16, 26 and 16 bits respectively