Version AHG/3

EGT3
ENGINEERING TRIPOS PART IIB

Monday 12 May 2025 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) 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. How are these modifications intended to improve the performance of the datapath? What precautions must be taken if these improvements are to be realised? [20%]

(c) The MIPS pseudo-instruction `ble $a,$b,dest` is translated by the assembler into the two real instructions `slt $1,$b,$a, beq $1,$0,dest`. Assuming the modified pipeline in Fig. 2, and counting from the start of the `slt` instruction as shown in Fig. 3, at which clock cycle might it be known whether the branch is taken or not? In the light of your answer, suggest why `beq` and `bne` are implemented directly in MIPS hardware, whereas other conditional branches like `ble`, `bgt`, `bge` and `blt` are pseudo-instructions. See below for definitions of the various instructions and pseudo-instructions. [30%]

```
slt $1,$b,$a     # if $b<$a $1=1 else $1=0
beq $a,$b,dest   # goto dest if $a=$b
bne $a,$b,dest   # goto dest if $a≠$b
ble $a,$b,dest   # goto dest if $a≤$b
blt $a,$b,dest   # goto dest if $a<$b
bge $a,$b,dest   # goto dest if $a≥$b
bgt $a,$b,dest   # goto dest if $a>$b
```

(d) Other instruction set architectures handle conditional branches by means of a *condition codes register* (CCR). For example, there might be bits N and Z in the CCR to indicate whether the result of an instruction is negative (N) or zero (Z). All ALU and data transfer instructions automatically set the bits of the CCR to 0 or 1, as appropriate, with subsequent branch instructions then branching (or not) depending on the state of the bits. Discuss the relative advantages and disadvantages of the MIPS and CCR approaches in the context of a pipelined datapath. In your discussion, be sure to consider: at which pipeline stages the CCR should be written and read; any implications for hazards; any implications for superscalar operation and dynamic pipeline scheduling. [30%]

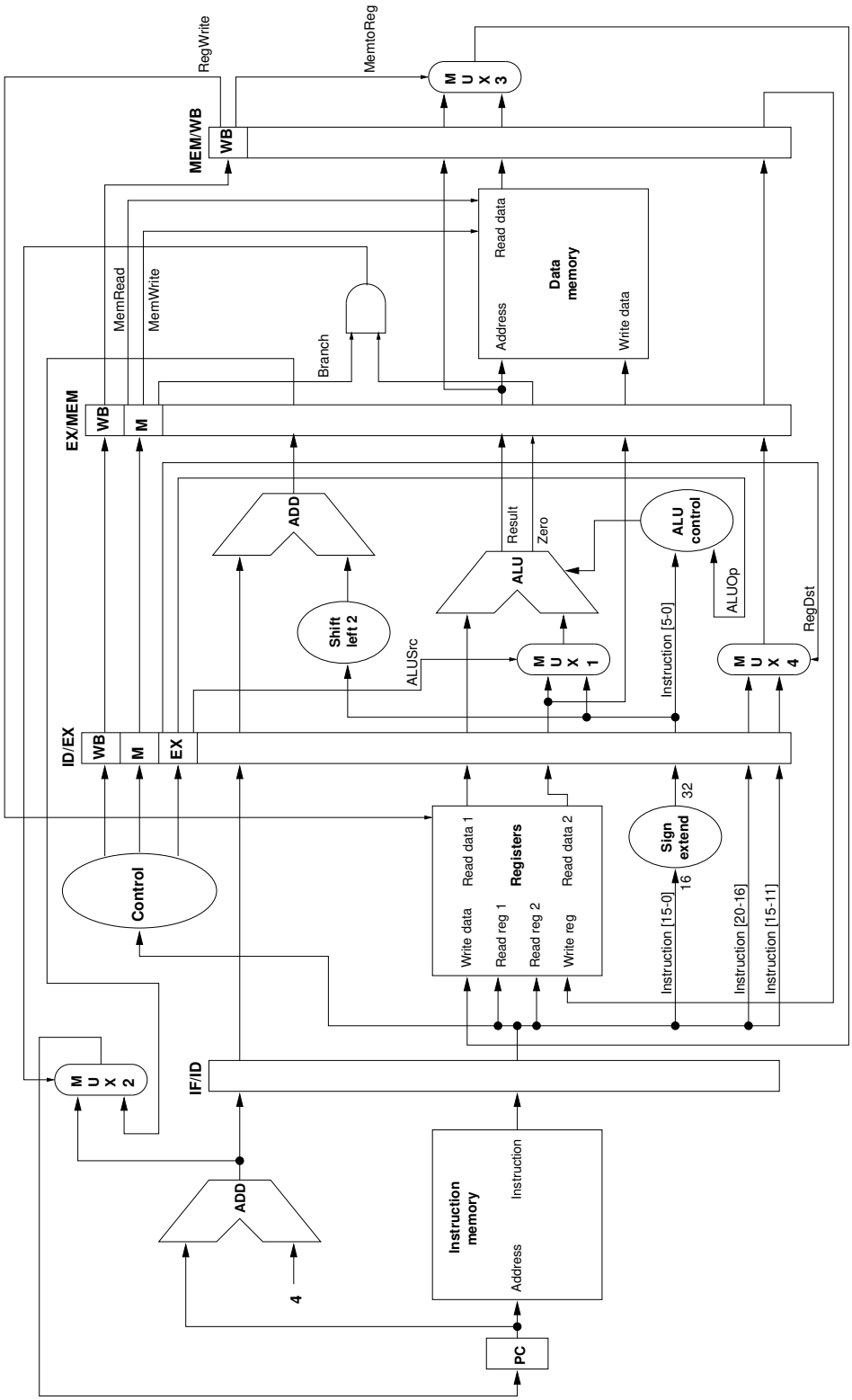
RegWrite
MemtoReg
M U X 3
MEM/WB
WB
Data memory
Read data
Address
Write data
MemRead
MemWrite
Branch
EX/MEM
WB
M
ADD
Result
Zero
ALU
ALU control
ALUOp
Shift left 2
ALUSrc
M U X 1
M U X 4
RegDst
Instruction [5-0]
ID/EX
WB
M
EX
32
Sign extend
16
Control
Registers
Read data 1
Read data 2
Write data
Read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

Fig. 1

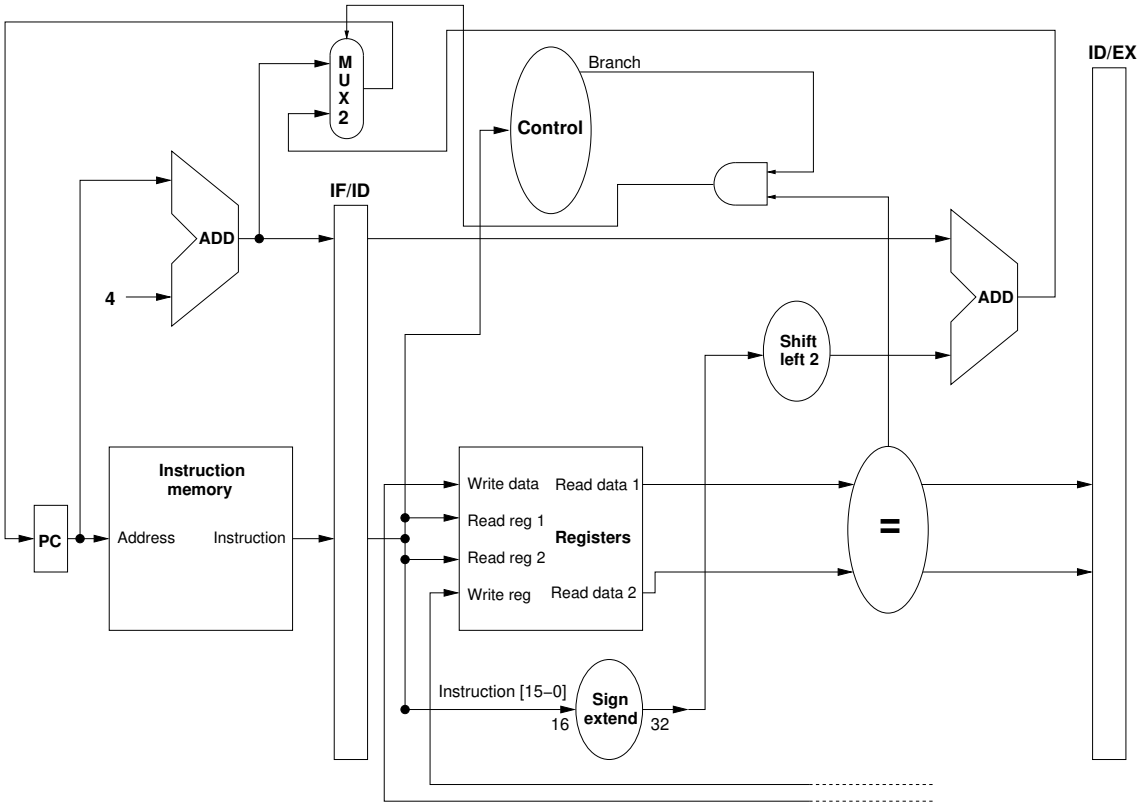
M
U
X
2
Branch
Control
ID/EX
IF/ID
ADD
4
ADD
Shift
left 2
Instruction
memory
PC
Address
Instruction
Write data
Read data 1
Read reg 1
Registers
Read reg 2
Write reg
Read data 2
=
Instruction [15–0]
Sign
extend
16
32

Fig. 2

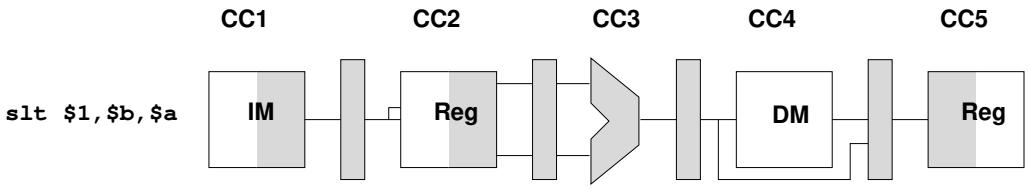
CC1
CC2
CC3
CC4
CC5
slt $1,$b,$a
IM
Reg
DM
Reg

Fig. 3

2 (a) What requirements of a modern computer system motivate the adoption of a virtual memory system? What is the purpose of the translation lookaside buffer (TLB)? [25%]

(b) A certain computer has a byte-addressable memory with 4-byte words. Both the virtual and physical addresses are 32 bits wide and the page size is 4 KiB. The TLB is direct-mapped with 64 rows and a block size of two page table entries (i.e. the TLB stores 128 page table entries).

(i) Assuming a TLB hit, sketch a hardware schematic showing how a virtual address is translated into a physical address. Annotate your sketch to show clearly how the addresses are divided into different fields and give the width of each field. How is the translation performed in the event of a TLB miss? [20%]

(ii) A process running on this computer copies sequential words from array `A` to array `B`. `A` is stored in memory starting at virtual address 0x80000000 while `B` starts at 0x90000000. Establish the pattern of TLB accesses and hence comment on the performance of the computer when executing this process. How might the computer's design be modified to improve this performance? [25%]

(c) Although a TLB is essentially a cache, different design decisions are often taken for TLBs compared with instruction/data caches. Identify the core differences and explain how they come about. How would the pattern of memory accesses in part (b)(ii) perform with a typical data cache? [30%]

3 (a) Distinguish between *accumulator*, *stack* 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.

Assuming 16-bit addresses, estimate the number of bytes required to store the instructions in your examples. Given your estimates, explain why reduced instruction set computers employing general purpose register, load-store instruction set architectures are nevertheless competitive when it comes to performance. [50%]

(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) What precautions must be taken when using DMA in conjunction with caches and virtual memory systems? [25%]

END OF PAPER