Review lecture

Module 1

- Execution time= Instr. Count x Cycles/instruction (CPI) x Clk period
- Performance = 1/execution time.
- CPI of a single cycle processor is 1
- Performance metrics in terms of FLOPS, MIPS and relative MIPS
- Overall speed up= performance of enhanced machine= T original
  performance of original machine T enhanced

  Amdhals law (workload constant)

  Gustafson law (total time taken constant)

SEMESTER 2 EXAMINATION 2018-2019

(a) Owing to an enhancement in the architecture of a processor, 60% of a program runs 12 times faster and the remaining 40% of the program runs 4 times faster. Find the overall speedup of the program due to this enhancement.

(5 marks)

Module 2 and 3

- ARMv8 ISA: A design example named as LEGv8
- Instruction format (R, D, I, B and CB format)
- Addressing modes (Register ,Immediate ,Base or displacement ,PC-relative addressing)
- Functionality of instruction
- Review of basic logic devices and register file
- Single-cycle datapath design (for all formats and addressing modes)

* 6. For branch (B offset) instruction with 26 bits in the address field, which is word address, can branch ± _____MB from the current PC.

Answer:

✔ 128

The correct answer is ±128.0

2^28= 256 MB. That means branch can move ±128 MB from the current PC

* 7a. Consider a modified version of LEGv8 processor with 32 bit of data and instruction bus. Find the maximum address of the instruction memory to which the control of execution of a LEGv8 code could be moved forward by the unconditional branch instruction of the form "B offset".

Choose one of the following answers.

- 0xFFFFFFFC
- 0xFFFFFFFF
- ✔ 0x07FFFFFC

The correct answer is 0x07FFFFFC
the maximum offset the Branch instruction can have is : 01 1111 1111 1111 1111 1111 1111
sign extended (offset) :
: 0000 0001 1111 1111 1111 1111 1111 1111
(as sign bit is zero)
Sign extended (offset) <<2 (byte address)
: 0000 0111 1111 1111 1111 1111 1111 1100
(0x07FFFFFC)

- 0x07FFFFFF

2. (a) Figure Q2a shows the block diagram of a single-cycle datapath for the implementation of R-format and I-format instructions including arithmetic and logic instructions, load, store and conditional branch instructions. Propagation delays of the datapath components at V=2.5 volt are given in Table Q2.

Table Q2

PCin→ PCout	ALU	Adder (for branch)	Shifter	Mux	AND gate	REG (R/W)	Sign-extend
100ps	250ps	200ps	50ps	10ps	5ps	200ps	50ps

PC++	Control logic	D-mem (R/W)	I-mem (R)
200ps	50ps	500ps	500ps

PC++ stands for the delay for PC increment. I-MEM (R) stands for the time for reading instruction memory, and D-MEM (R/W) stands for the time required for read or write operation on data memory. REG (R/W) stands for the time required for read or write operation on register file.

(i) Explain the operations of SW and BEQ instructions, respectively, by providing details on the operations resulting in the maximal latency. Use examples if needed.

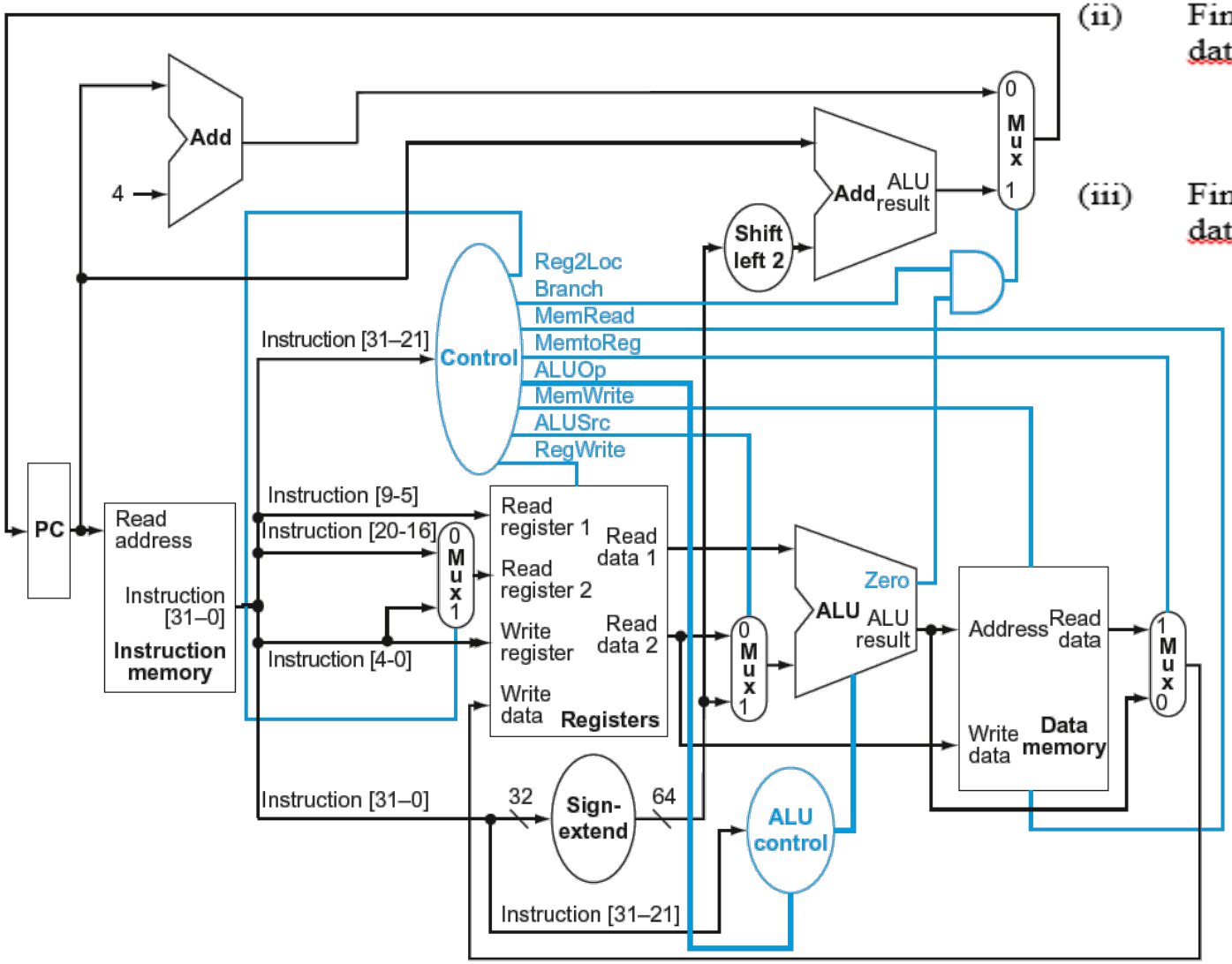
(8 marks)

(ii) Find the maximum possible frequency at which the given datapath can function if the operating voltage V=2.5 volt.

(3 marks)

(iii) Find the maximum possible operating frequency of the given datapath at the operating voltage V=1 volt.

(3 marks)



Module 4

- Concept of pipelining
- Data , control and structural hazards
- RAW, WAW and WAR (data hazards)
  - Detect and wait – easier to implement but more stalls
  - Forwarding through register (DEC and WB simultaneously)
  - Detect and forward- needs separate hardware but less stalls
- Control hazards
  - Conservative method (3 stalls)-update in memory stage
    - (2 stalls) update of PC in exe stage
    - Early evaluation of PC in DEC stage 1 stall)
  - Static prediction
    - Always not taken
    - Delayed branching
  - Dynamic prediction
    - One bit prediction
    - 2 bit prediction
- Instruction level parallelism (superscalar and VLIW processors)

Consider the following code segment which is intended to be executed in a 5-stage pipelined LEGv8 processor. Assume that write-back and register-read operations of different instructions can be performed in the same clock cycle.

Listing Q2a

```
I1              ADDI X4, X31, 0X36
I2  Loop:       LDUR X2, [X1, #0]
I3              XORI X2, X2, #10
I4              STUR X2, [X1, #0]
I5              ADDI X1, X1, #8
I6              SUBI X4, X4, #2
I7              CBNZ X4   Loop
```

- Calculate the number of stall cycles needed for the execution of the code in Listing Q2a, if full data forwarding is allowed. Assume that the program counter is updated with the branch target address at the memory stage.

- Show the data forwarding path and find the steady state CPI with full data forwarding. Also determine the total number of iterations of the loop,

```
I1            ADDI X4, X31, 0x36
I2  Loop:     LDUR X2, [X1, #0]
I3            XORI X2, X2, #10
I4            STUR X2, [X1, #0]
I5            ADDI X1, X1, #8
I6            SUBI X4, X4, #2
I7            CBNZ X4   Loop
```