SECTION A **ANSWER ALL Questions in this section**

QUESTION 1

a) What kind of problem arises if two devices attempt to use the system bus at the same time?

[1]

a) The problem that arises is Bus contention where two or more devices are sharing a common resource such as memory or disk.

b) Name the four general purpose registers found in the 80x86 microprocessor, each general purpose register also has special functions, state what they are.

[8]

b) EAX=Accumulator
EBX = Base Register

ECX= Counter register

EDX=Data Register

c) What do the square brackets mean when they appear in an operand? (e.g. MOV EAX,[3000])?

[2]

c) The square brackets implies that you get the value from the memory location 3000 and store it in AX i.e. CONTENTS OF.

d) What are the three conditions that need to be satisfied in the critical section problem of process synchronization.

[3]

d) Mutual Exclusion, Progress and Bounded Waiting.

e) What are the two problems with using semaphores?

[2]

e) Two problems with using semaphores are: Deadlock and starvation.

f) Write brief notes, with machine code examples, explaining the following addressing modes in 80x86 CPU's

i) Immediate Addressing.

i) This is the simplest addressing mode. The instruction contains *data* to be used by the instruction rather than the address of the data

E.g. “Load ECX with the value 1024 ”

MOV ECX,1024

The data is available when the instruction is fetched

Don’t need to get data from the memory

Useful for dealing with constants. But the constant may be limited by the size of the instruction.

ii) Direct Addressing.

ii) The instruction contains the address of the data to be used by the instruction

E.g. “Load register EAX with the data at memory location 1234”

MOV EAX, [1234]

The instruction will always access exactly the same memory location.

Need to know the desired address at compile-time. Useful for dealing with global variables

iii) Registered Addressing.

[9]

iii) Like direct addressing, but the data is in a register rather than a memory location

E.g. “Add registers BX and CX and put the result in register AX”

ADD BX, CX

E.g. “Load register EAX with the data in register EBX”

MOV EAX, EBX

g) What is meant by the term Instruction Set Architecture?

[4]

- How the machine appears to a machine-code programmer
- Interface between the software and the hardware
  - Programs are translated into ISA-level “machine code” instructions
  - Hardware executes these directly

h) In 80x86 assembler. Explain what DUP (?) does.

Keys BYTE 128 DUP (?), 0

[1]

DUP (?) says to set aside x amount of memory and leave the contents of the memory locations with their current value.

SECTION B **SELECT <u>ANY</u> TWO Questions from this section**

QUESTION 2

a) If register AL contains the value 20H and register CL contains the value 80H.

i) Convert 20H and 80H to Decimal values (Show your workings). [2]

ii) Multiple the two values together and convert the result back to Hexadecimal. (Show your workings) [2]

iii) What register is used to store the result of executing the instruction MUL CL as calculated above?

[1]

a) The decimal equivalents of 20H and 80H are 32 and 128 respectively

32 x 128 = 4096

4096 = 1000H

Upon completion of MUL CL. AX contains the result 1000H

They also need to show there workings.

b) Register AX contains the value 0A064H. Upon executing the instruction MUL AX a result of 647D2710H is obtained, this is too big to fit in register AX.

i) What other register is used to hold part of the result?

[2]

ii) Does the register that you identified hold the upper half or the lower half of the final value? [1]

iii) Show what value is stored in each register. [2]

b) The register used in conjunction with the AX register during 16-bit multiplication is the DX register. The upper half of the value 647D2710H is stored in DX the lower half in AX
DX = 647DH and AX = 2710H

This split will still occur even though the EAX register is 32 bit. As the assembler knows you are only doing 16-bit operations because you explicitly used the AX register.

c) Write brief notes on the 80x86 instructions LOOPE and LOOPZ. How do they differ from the basic LOOP instruction?

[4]

LOOPE Loop if equal and LOOPZ Loop if Zero.
These instructions are similar to LOOP but a secondary condition must also be met.
In addition to decrementing CX LOOPZ also examines the state of the zero flag. If set and CX not equal to zero LOOPZ will jump to the target.
If CX equals zero, or the zero flag gets cleared within the loop, the loop will terminate.
LOOPE is an alternate name for this instruction.

d) How many times does the NOP instruction execute in the following sequence? Explain your answer.

```
      MOV   CX,20H
XYX:  PUSH  CX
      MOV   CX,9
ABC:  NOP
      LOOP  ABC
      POP   CX
      LOOP XYZ
```

[7]

d) The outer loop (XYZ) is executed 32 times. (CX = 20H = 32)
The inner loop (ABC) is executed 9 times.

Therefore the NOP instruction will execute 9 x32 = 288 times

e) Outline the code required to perform a "for" loop in 80x86 assembler, using the counter register.

[4]

```
      mov    CX,10
Back: -------
      -------
      -------
      -------
loop  Back
```

This will loop 10 times before exiting the routine

[Total 25]

QUESTION 3

a) What does the term RAID stand for?

[1]

a) RAID stands for Redundant Array of Inexpensive Disks

b) What does the term Little endian mean?

b) Little endian refers to how INTEL processors store a value in memory, the bytes are swapped. Least significant bit goes to the lower memory location on left, most significant bit to the right.
i.e value 1234 is store 3412.

c) One structure that is created on a disk drive during initialisation is called the BOOT SECTOR. Describe the basic components of this structure.

[10]

- This is a vital sector. Disk will be unusable if this sector gets damaged
- Each partition on a hard drive has its own boot sector.
- Each operating system has its own boot sector format.
- A DOS/Windows formatted floppy/hard disk's Boot Sector contains
  - A jump and a NOP opcode
  - BPB (BIOS Parameter Block)
    - Sectors per cluster
    - Number of Root directory entries
    - Sectors per FAT (File Allocation Table)
    - Volume Label
- For Booting, Bootstrap Loader loads Boot Sector data in to a particular address of memory (0000:7C00h) and sets the PC. If OS not present then an error message is displayed
    - e.g. NON SYSTEM DISC or DISC ERROR REPLACE DISC AND STRIKE ANY KEY WHEN READY

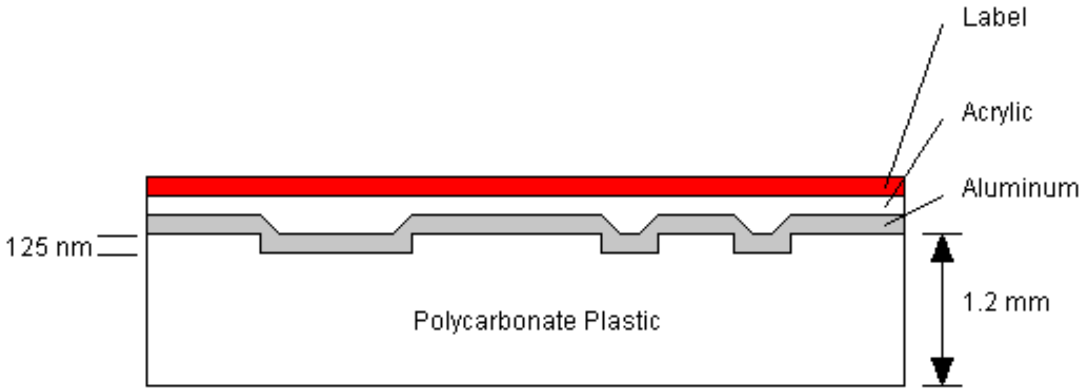
d) Outline the structure of a CD ROM.

[4]

A CDROM Drive uses a small plastic-encapsulated disk that can store data

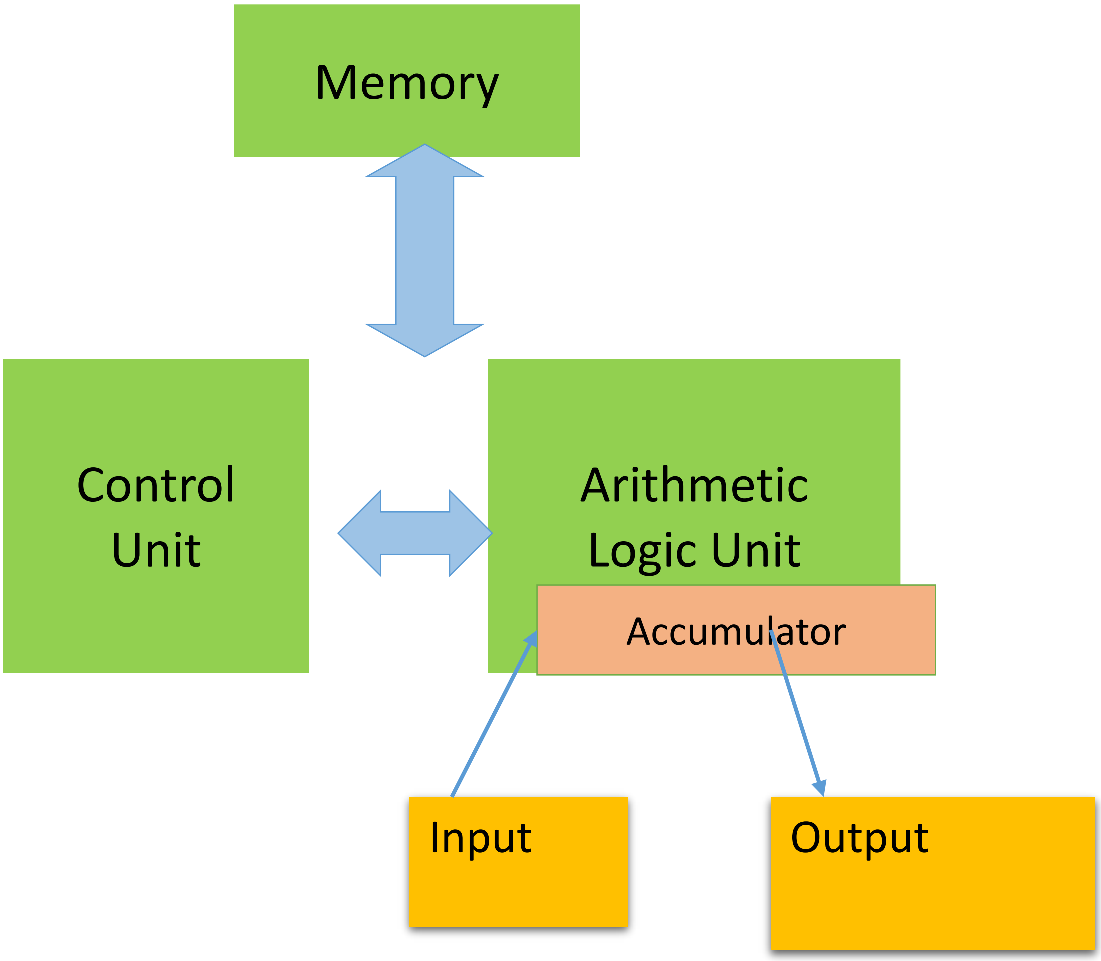
A CD consists of four layers

- The biggest part is clear polycarbonate (nominally 1.2mm)
- There is a very thin layer of reflective metal (usually aluminum) on top of the polycarbonate
- Then a thin layer of some protective material covering the reflective metal
- A label or some screened lettering on top of protective material



e) There are 5 major components that make up the Von Neumann architecture. List what the components are and draw a outline diagram

[9]



[

Question 4

a) What is meant by the term, CPU-I/O Burst cycle? How is it related to CPU-bound and I/O-bound processes? [4]

CPU-I/O burst. Process execution consists of a cycle of CPU execution and I/O wait. When one process does I/O, a scheduler will typically switch the CPU to another process.

- A typical process execution has
  - a large number of short CPU bursts
  - a small number of long CPU bursts
- A **CPU-bound** process – mostly long CPU bursts
- **I/O-bound** process – mostly short CPU bursts

b) What are the advantages and disadvantages of FIXED_PARTITION MEMORY MANAGEMENT?

[4]

Fixed partition memory

Advantage

Simple

Disadvantage

The degree of multiprogramming is constrained.

The size of each process is bounded.Suffers **internal fragmentation**

Memory that is internal to a partition but is not being used

c) When dealing with variable partition memory management, what are the three placement strategies that can be employed. ?

[6]

The three placement strategies are:

First-fit: allocate the first hole large enough.

Best-fit: allocate hole with the smallest leftover.

Worst-fit: allocate the largest hole.

d) In Round-Robin Scheduling, what can happen if the time quantum is set too high? [1]

[1]

- If the quantum time is too large, RR scheduling degenerates into First Come First Served.

e) What is a semaphore? [2]

1. A **semaphore** is a mechanism provided by the system. It is associated with a queue that stores the references to the processes that are waiting. Its a special integer variable **S** that, apart from initialization, is accessed only through two standard **atomic** operations.

f) What is the difference between Preemptive vs. Non-preemptive Scheduling? [8]

Non-preemptive Scheduling: Once the system has assigned a CPU to a process, the system cannot remove that CPU from that process
Simpler

Up to the process to release the CPU

Preemptive Scheduling: The system can remove the CPU from the running process.

Need extra hardware (timer)

What if the process is in the middle of updating some data?

[Total 25]