# KLS Vishwanathrao Deshpande Institute of Technology

(Accredited by NAAC with "A" Grade)

(Approved by AICTE, New Delhi, Affiliated to VTU, Belagavi)

(Recognized Under Section 2(f) by UGC, New Delhi)

Udyog Vidya Nagar, Haliyal - 581 329, Dist.: Uttara Kannada

www.klsvdit.edu.in | principal@klsvdit.edu.in | hodece@klsvdit.edu.in

## DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

# University / Model Question Paper Scheme & Solution

| | | |
|---|---|---|
| Faculty Name | : | Nikhil Kulkarni |
| Course Name | : | Embedded System Design |
| Course Code | : | BEC601 |
| Year of Question Paper | : | June / July 2025 |
| Date of Submission | : | 13-02-2026 |

Faculty Member

HoD

Dean (Acad.)

Head of the Department
Dept. of Electronics & Communication Engg.
KLS V.D.I.T., HALIYAL (U.K.)

CBCS SCHEME - Make-Up Exam

USN | | | | | | | | | |

BEC/BTE/BVL601

# Sixth Semester B.E./B.Tech. Degree Examination, June/July 2025
## Embedded System Design

Time: 3 hrs. Max. Marks: 100

***Note: 1. Answer any FIVE full questions, choosing ONE full question from each module.***
***2. M : Marks , L: Bloom's level , C: Course outcomes.***

| | | Module – 1 | M | L | C |
|---|---|---|---|---|---|
| Q.1 | a. | Define Embedded Systems. Explain the purposes of embedded systems with an examples for each. | 10 | L2 | CO1 |
| | b. | What are the different types of memories used in embedded system design? Explain the role of each. | 10 | L2 | CO1 |
| | | **OR** | | | |
| Q.2 | a. | With the diagram of elements of embedded system, mention all the cores around which an embedded system is built. Discuss any two in detail | 10 | L2 | CO1 |
| | b. | Write a brief note on the following :<br>i) 7 segment display<br>ii) Brown out protection<br>iii) UART<br>iv) I2C | 10 | L2 | CO1 |
| | | **Module – 2** | | | |
| Q.3 | a. | Illustrate the domain specific aspect of embedded systems with automotive domain as an example. | 10 | L2 | CO2 |
| | b. | Design drives/passenger 'Seat Belt Warning' in an automotive using the FSM model. The system requirements are captured as<br>i) When the vehicle ignition is turned ON and the seat belt is not fastened within 10 seconds of ignition ON the system generates an alarm signal for 5 seconds.<br>ii) The Alarm is turned OFF when the alarm time (5 seconds) expires or if the driver/passenger, fastens the belt or if the ignition switch is turned off, whichever happens first. | 10 | L3 | CO2 |
| | | **OR** | | | |
| Q.4 | a. | Illustrate the application specific aspect of embedded systems with washing machine as an example. | 10 | L2 | CO2 |
| | b. | Design an automatic tea/coffee vending machine based on FSM model for the following requirement :<br>i) The tea/coffee vending is initiated by user inserting a 5 rupee coin<br>ii) After inserting the coin, the user can either select 'Coffee' or 'Tea' or press 'Cancel' to cancel the order and take back the coin. | 10 | L3 | CO2 |

| | | | | | |
|---|---|---|---|---|---|
| | | **Module – 3** | | | |
| **Q.5** | **a.** | Briefly explain the function of operating system with diagram. | **10** | **L2** | **CO3** |
| | **b.** | Explain preemptive SJF scheduling and illustrate with an example. | **10** | **L3** | **CO3** |
| | | **OR** | | | |
| **Q.6** | **a.** | Write a note on racing and deadlock in task synchronization. | **10** | **L2** | **CO3** |
| | **b.** | With a diagram, mention function of the components in an embedded system development environment. | **10** | **L2** | **CO3** |
| | | **Module – 4** | | | |
| **Q.7** | **a.** | With a block diagram, explain typical ARM based embedded system. | **10** | **L2** | **CO4** |
| | **b.** | What is pipeline in ARM? Explain the different pipeline stages of ARM processor. | **10** | **L2** | **CO4** |
| | | **OR** | | | |
| **Q.8** | **a.** | With a block diagram, explain ARM core dataflow model. | **10** | **L2** | **CO4** |
| | **b.** | Write a detailed note on : i) Vector table ii) Exception iii) Interrupts. | **10** | **L2** | **CO4** |
| | | **Module – 5** | | | |
| **Q.9** | **a.** | Explain the different branch instructions of ARM processors. | **10** | **L2** | **CO5** |
| | **b.** | Write an ALP with comments to add an array of 16 bit numbers and store the 32 bit result in internal RAM. | **10** | **L3** | **CO5** |
| | | **OR** | | | |
| **Q.10** | **a.** | Explain the different data processing instructions in ARM. | **10** | **L2** | **CO5** |
| | **b.** | Write an ALP with comments to sort the array of 32 bit numbers in ascending order using bubble sort method. | **10** | **L3** | **CO5** |

* * * * *

# MODULE – 1

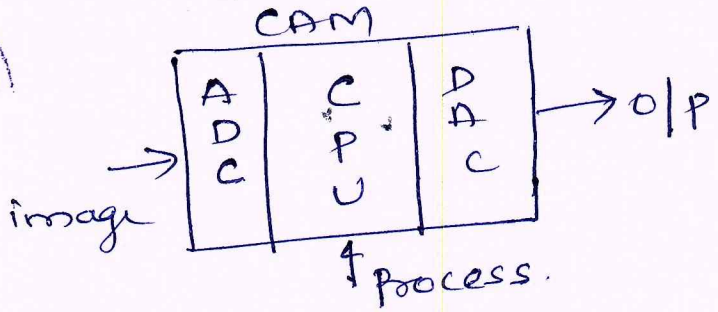
Q1a7. Embedded System is the electronic ckt having both hardware and software (firmware) designed on a single chip. – 2M

- Purpose Of ES

Embedded System can be used in many applications such as. (Any 4) 2 marks each

① Data Collection | Storage | Representation:

A typical Digital Camera is an example of such system. in which image to be taken can be considered as analog value which can be converted to Digital by ADC, further a process - or can be employed for image processing and an o/p side DAC can be employed to convert it to analog representation – 2M

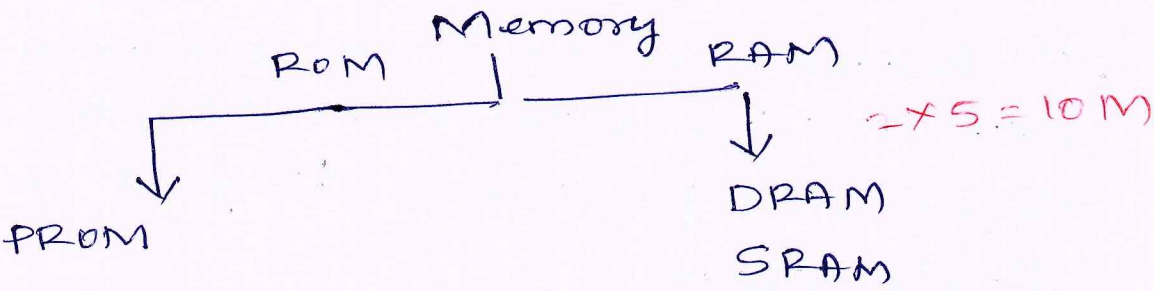



② Data Communication: A n/w router is an example, explanation – 2M

③ Data processing: A Digital hearing aid explanation – 2M

④ Monitor: A CRO, TV Screen (interactive) explanation – 2M

Q1b> Different types of memories used in ES are




-x5 = 10M

Any 5 can be explained.

PROM: Programmable ROM, (one time programmable), as the name indicates it can be programmed only once.
Ex: Digital Remote

MROM: Masked ROM, It is also a OTPROM.

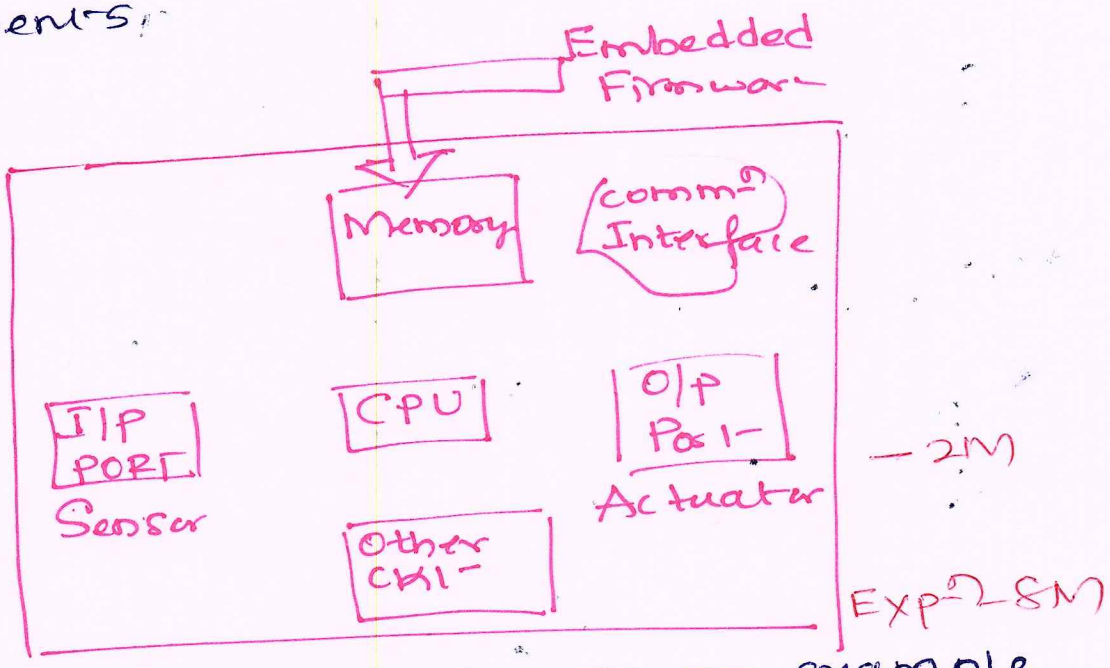
EPROM: Electrically Programmable ROM, which can be programmed through fuse

| DRAM | SRAM |
|---|---|
| made using one MOSFET | made using 6 MOSFET |
| Refreshing required | not required |
| Low capacity | High capacity |

MODULE-1

Q2a) Elements of an ES.

A Typical ES consists of the following elements:






Let us take a Laptop as an example in which following parts can be understood with elements of ES.

① CPU / core: It is the main processor associated with laptop like i3, i5, etc which are typically FPGA / ASIC / DSP / SoC

② I/P Port: Sensors, In ES Sensors are required to convert analog data to electrical signals., Ex: Face detection, Finger print i/p

③ O/p Port: Actuator, In ES actuators are required to convert Digital signal s to analog quantities, ex: Audio s/m in laptop;

④ Other CKT: These are modem IC, Graphics card of laptop

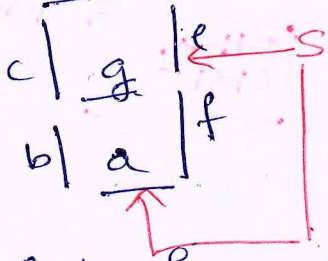
Q2a) Continued -

Memory: It refers to RAM, ROM of laptop.

Communication Interface: These are bluetooth, wifi, USB, HDMI etc.

Q2b) i) 7 Segment Display (3M)

These are the group of 7 LEDs arranged from a to e as shown below, where d




each Segment from a to f are LEDs which are typically connected either in common Anode or common cathode configurations.

In common Anode all Anodes of a to f are connected to VCC, ∴ to turn on a LED we need to load 0, In case of common cathode we need to load 1 to turn on LEDs.

Example to represent 1 or 1 we use following sequences

| | a | b | c | d | e | f | g | Display |
|---|---|---|---|---|---|---|---|---|
| CA | 1 | 1 | 1 | 1 | 0 | 0 | 1 | 1 |
| CC | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 1 |

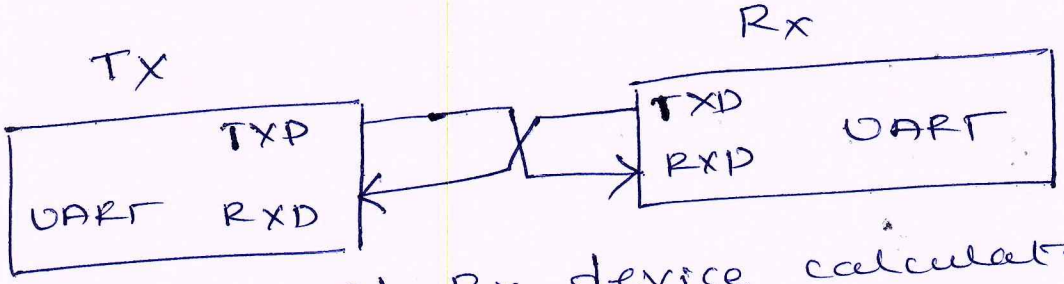
Q2b) continued...

ii) Brown out protection: — @ 1M

- It prevents processor from unexpected program execution behaviour when the supply voltage falls below the threshold level.
- In ckt - when $V_{CC} < V_{BE} + V_2$, The BJT stops working, and restarts the CPU.

iii) ~~XP~~ UART — 3M

- Universal Asynchronous Receiver Transmitter
- It is the serial data transmission method

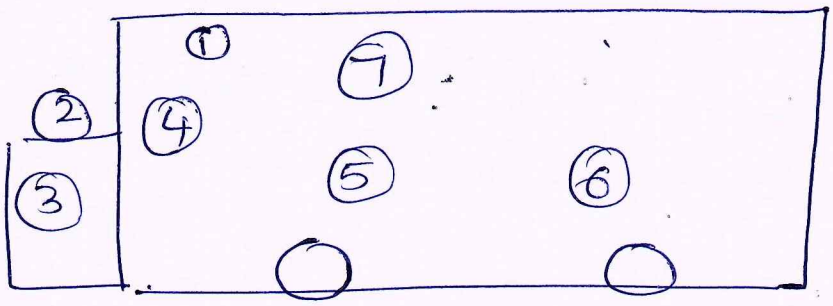



- The UART of Rx device calculates the parity of the bits received and compares it with the received parity bit for error checking.
- The UART of Rx. device discards the start, stop and Parity bit from the received bit stream and converts it to a word

iv) I2C : Block Diagram — 3M
Explanation

Q3a) Embedded System in automotive domain.

→ ES plays important role is automobiles also, Let us consider the car and its parts.

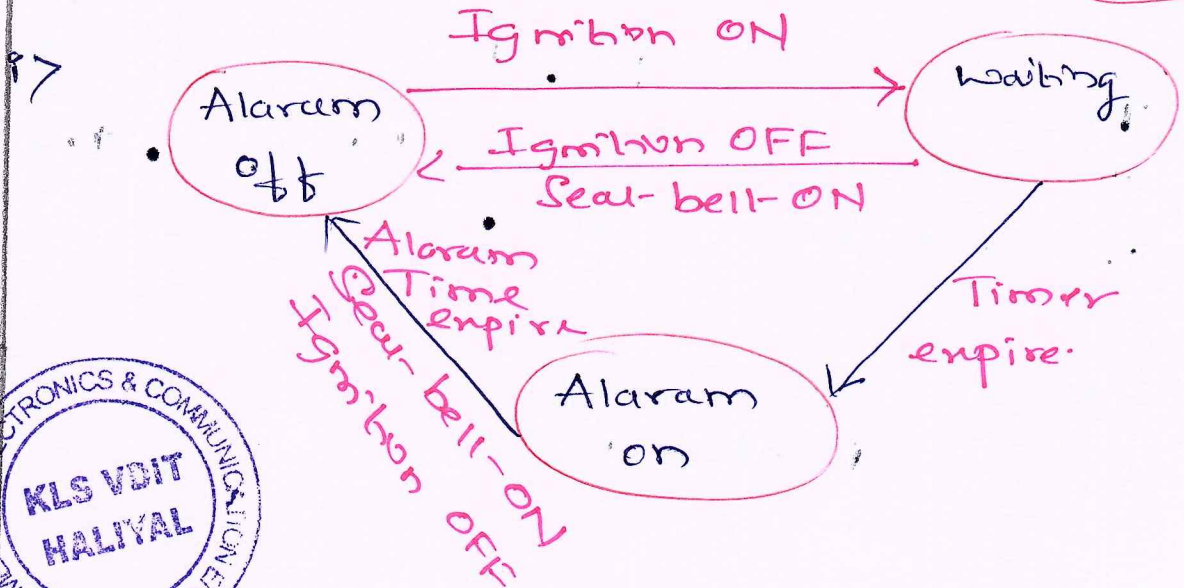



1. → Instrumentation
2. → Engine Control
3. → FAN control
4. Air Conditioner
5. Air bag control
6. Wiper / Power windows.

Explain any 5 Functions

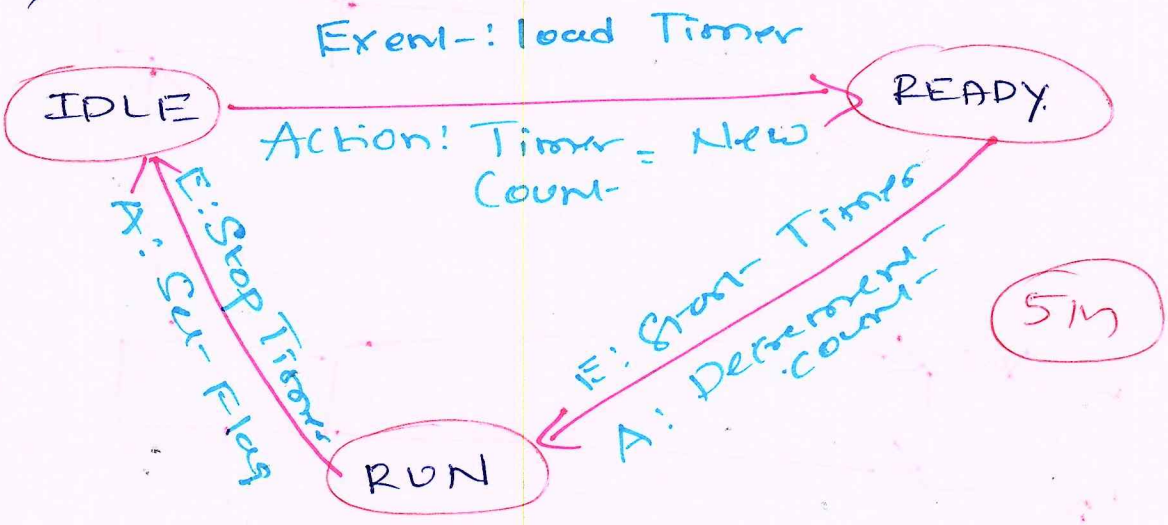
Q3b) FSM Model

5M



Q3b) continued..

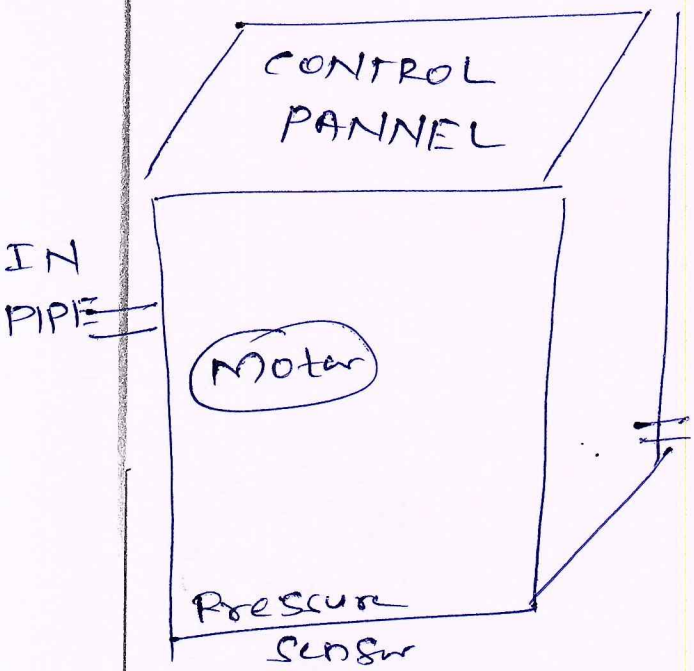
ii)



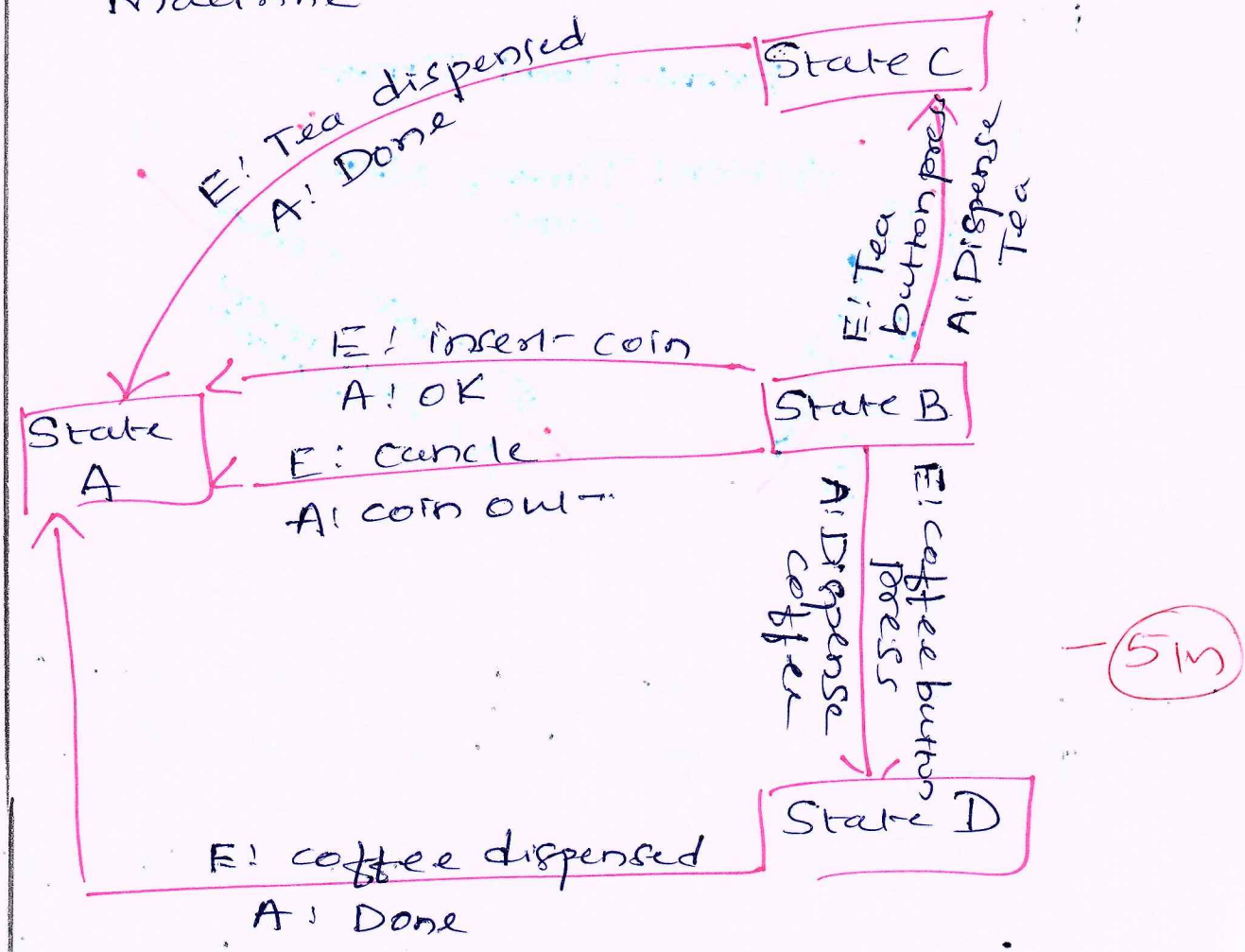

Q4a) Washing machine parts ES appln: 10M
Diagram - 5M, Explanation - 5M.

Explain the following

- Control Pannel for various wash cycle
- In pipe water level indicator
- motor rotation based on i/p



Q4b> FSM model for Tea/coffee Vending Machine




A → wait-for coin

B → wait-for user i/p

C → Dispense tea

D → Dispense coffee

SM

# Module-3

Q5a) Functions of OS.

Diagram - 5M

Explanation - 5M

Main Functions are

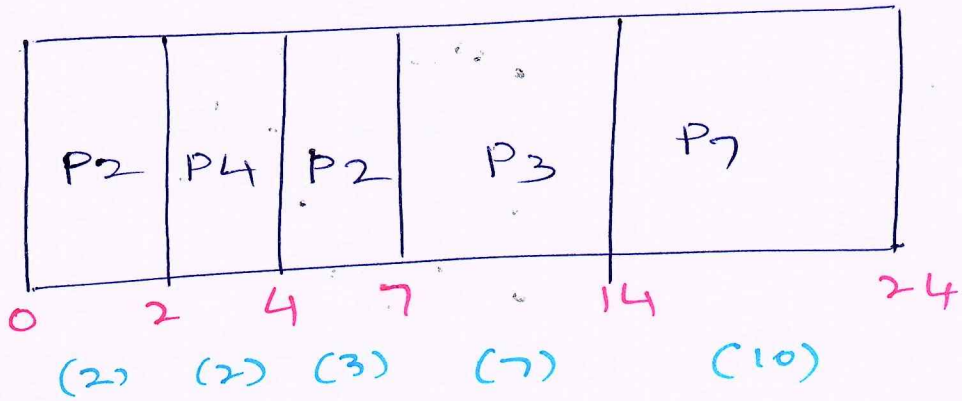
| Name | Function |
|---|---|
| (1) Task Management | • Create, Schedules, delete task<br>• Critical task run.<br>• Supports multitask |
| (2) Scheduling | • Determines which task to run next<br>• priority based event<br>• Deadline use met |
| (3) Memory Management | • Allocates and free memory<br>• prevents memory conflicts<br>• Optimized for limited memory resource |
| (4) Power management | |
| (5) IPC etc | |

Q5b) Example of preemptive SJF

→ Consider the 3 process P1, P2, P3 with 10, 5, 7 ms enter the queue

→ A new process P4 enters with 2ms estimated completion time after 2ms.

→




waiting time

- P2 = 2ms, P4 = 0ms, P3 = 7ms, P1 = 14ms

- Avg wait-time: 0+2+7+14 / 4 = 5.75msec

- TAT

P2 = 7ms, P4 = 2ms, P3 = 14ms, P1 = 24ms

# module-3

Q6a) Race condition:

- It occurs when 2 or more tasks access the shared data at same time
- In this cond° OS allocates the resource based on the priority assigned.
- Example:
  Task A reads A1, Task B reads A1
  ∴ result is A2 instead of A3.
- It can be controlled by mutex method

-5M

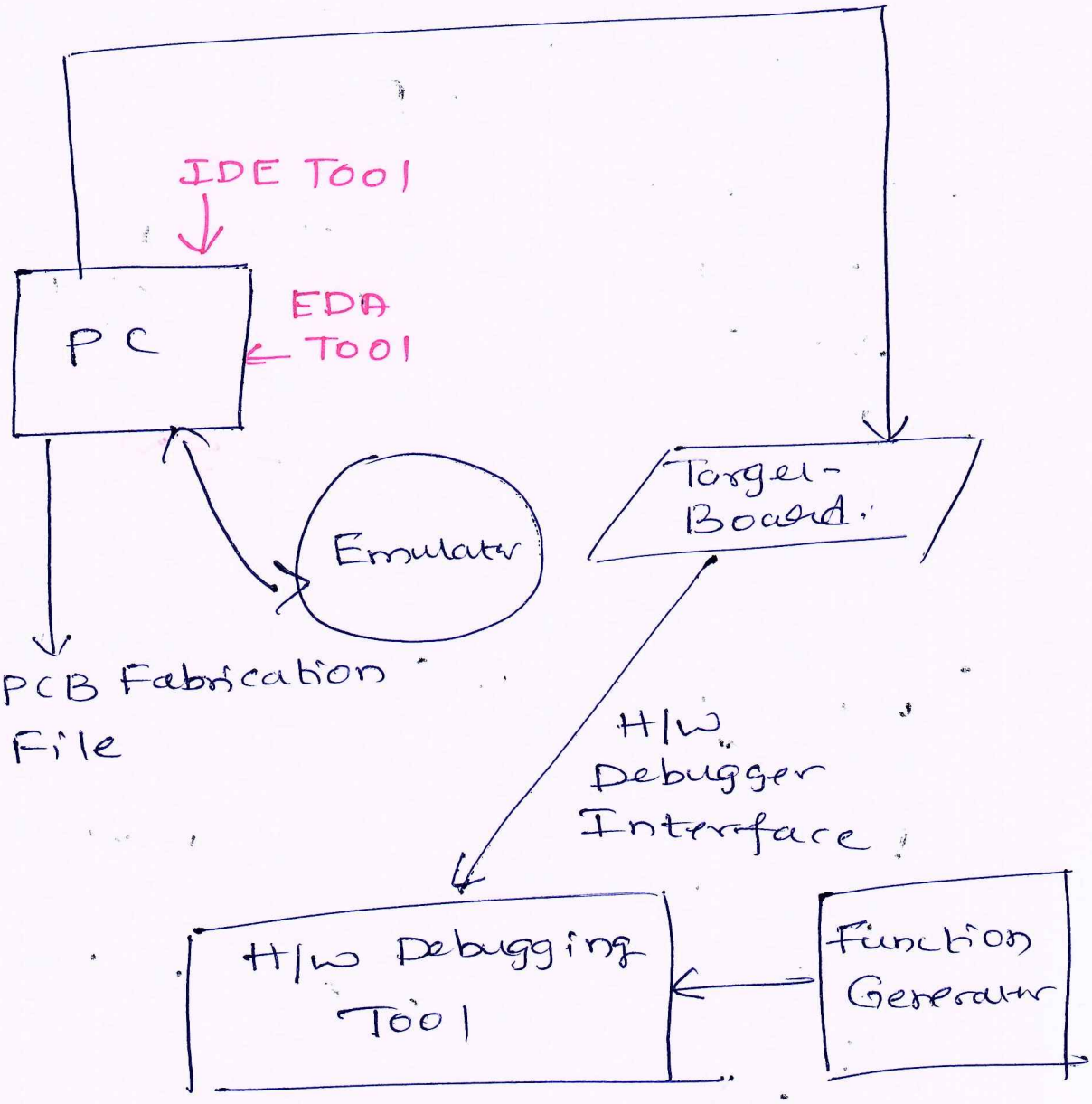
Deadlock

- It occurs when 2 or more resources wait for the resources held by each other so none can proceed
- Example:
  - Task A holds resource 1 waits for resource 2
  - Task B holds resource 2 wait for resource 1
- To avoid deadlock use time limit for smooth execution

-5M

module-3

Q6b) Diagram - 6M
Explanation - 4M.




ES Development Environment

Example : FPGA Board
Pi-Boards
etc

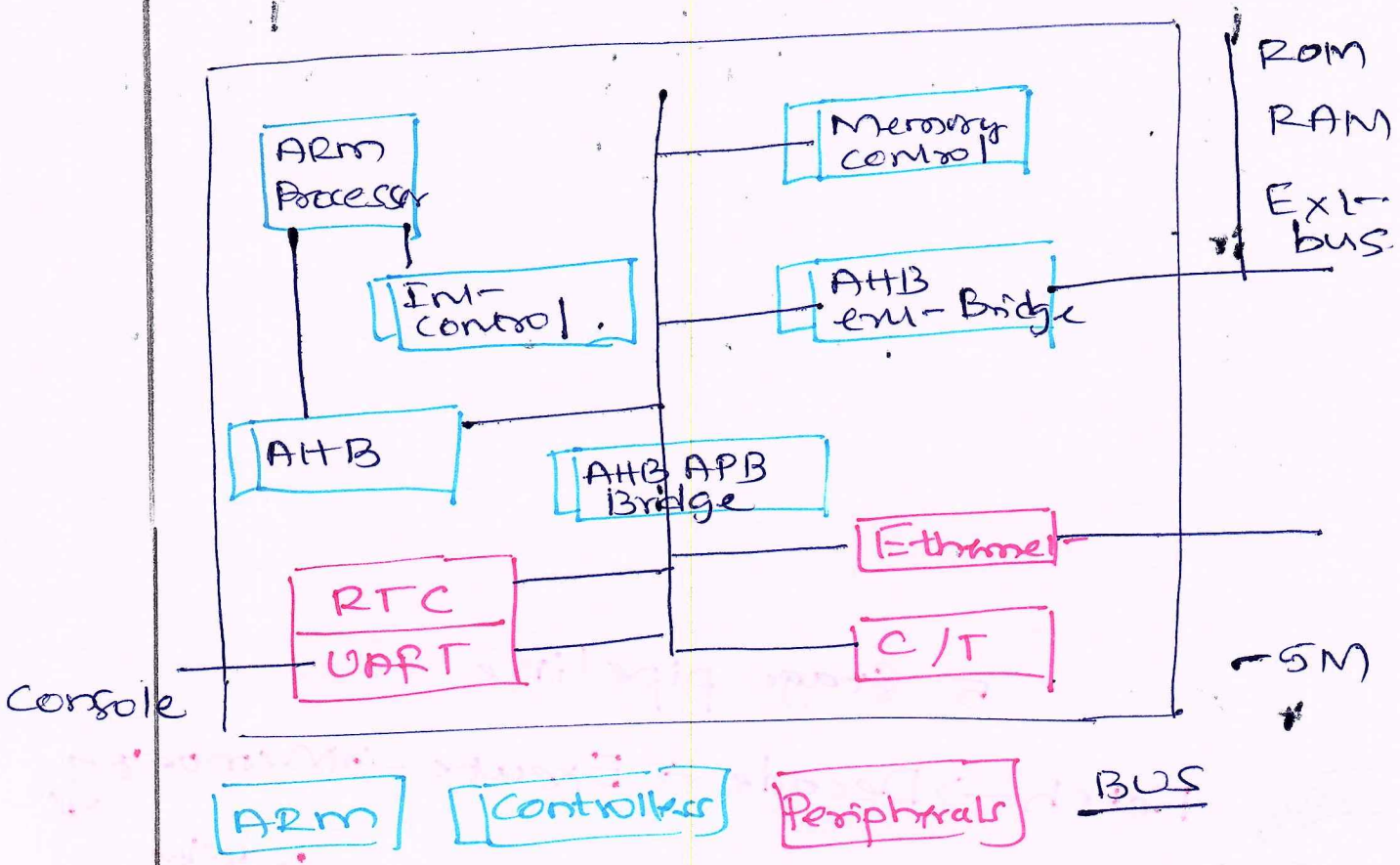
# Module - 4

Q4A) Typical ARM based ES 10M

Block Diagram - 5M

Explanation - 5M

Block Diagram of ARM based ES:




-5M

Explanation of
- ARM Processor
- AHB [ARM High Speed Bus]
- APB [ARM Physical Bus]. Peripheral

} 5M

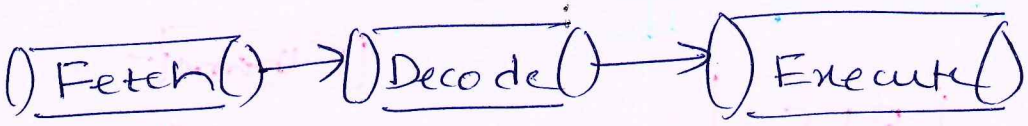
- Typical CPU size is 32 bit
- All Registers are orthogonal
- based on RISC Family

Q7b) Pipeline in ARM

- It is the process of CPU executing multiple task in a given cycle of execution.
- In one clock cycle, a CPU can handle Fetching(N) of present opcode, Decoding of previous opcode (N-1), decoding of opcode of previous (N-2), Execution of (N-3) instruction

- Different Pipeline Stages in ARM ase

* ARM7 3-Stage Pipeline




* ARM9 5 Stage pipeline

Fetch → Decode → Execute → Memory → write



Compared to conventional pipeline process it includes 2 more Stages memory access and writeback

* ARM10 Six-Stage Pipeline

Fetch → Issue → Decode → Execute → Memory → write

A new Stage called issue is added b/w Fetch and Decode to avoid overlapping and resource lock and dead lock.

module - 4

Q8a) ARM core data flow model

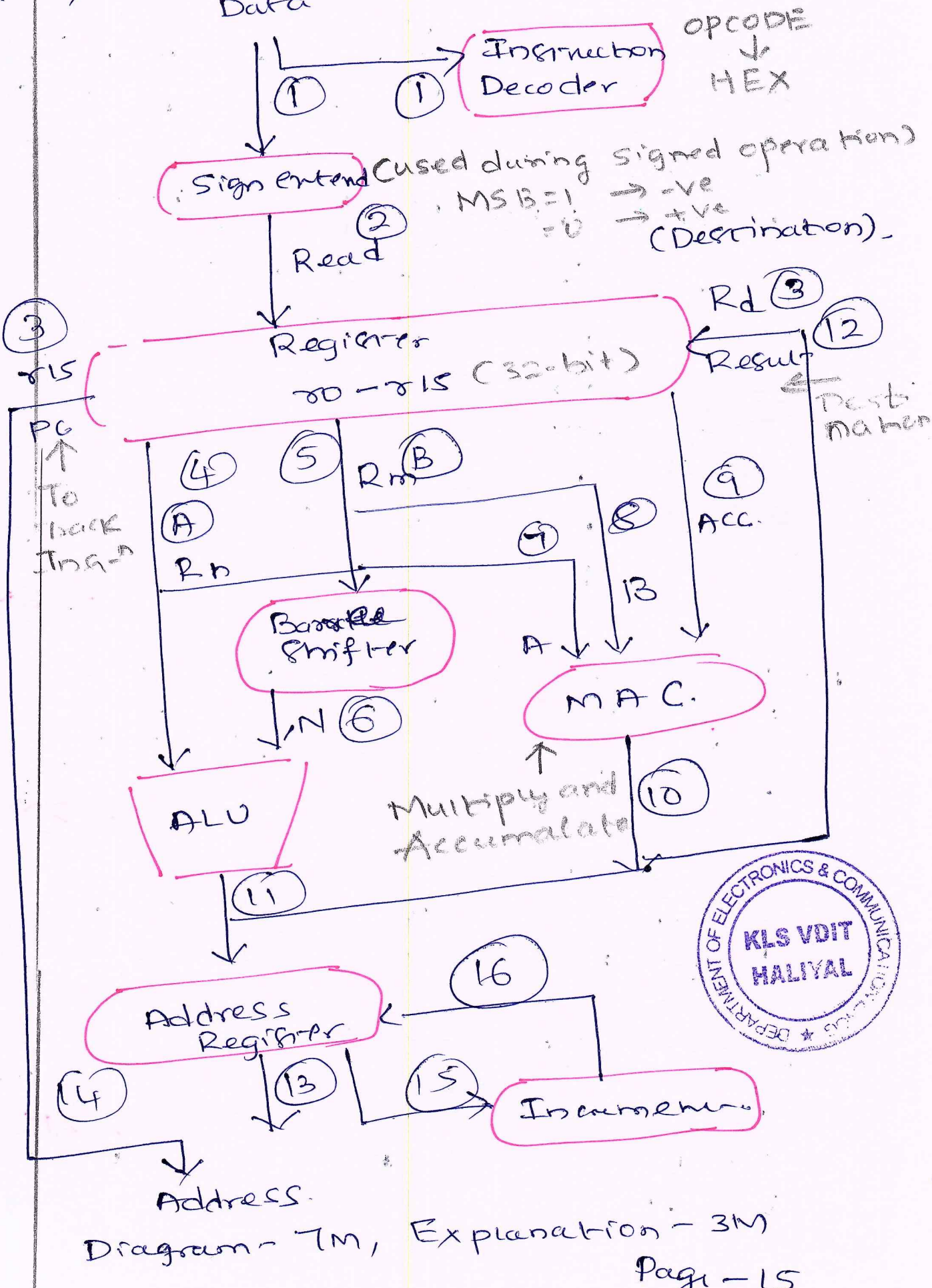



Diagram - 7M, Explanation - 3M

Q8b) Note on

i) Vector Table

When a interrupt occurs or service is requested by, then CPU stops the execution and saves its address in stack and jumps to the address of respective Exception/interrupt as mentioned below

| Exception / Interrupt | Address |
|---|---|
| Reset → | 0X00000000 |
| UNDEF → | 0X00000004 |
| SWI → | 0X00000008 |
| PABT → | 0X0000000C |
| DABT → | 0X00000010 |
| Reserved → | 0X00000014 |

— 6M



ii) Exception: It is the error which is allowed to happen in a execution cycle of a code, ex: divide by zero error.

— 2M

iii) Interrupts:

These are the service requested during the present execution of a code by CPU

Depending on maskable/non-maskable interrupt CPU gives the service or terminates the request — 2M

Ex: wrong pin in ATM during withdraw, but print error is not an maskable one

# Module-5

Q9a) Different Branch Instructions 10M

- It changes the flow of execution or it is used to call a routine
- Syntax

B{<cond>} Label

BL{<cond>} Label

BX{<cond>} Rm

BLX{<cond>} Label/Rm

B → Branch with L [RL Reg value]

BL → Branch with Link

BX → Branch exchange

BLX → Branch exchange with link

- Examples

```
    B   forward
    ADD r1,r2,#4
    ...
forward SUB r1,r2,#4
back  ADD r1,r2,#4
    ...
    B   back
```

Branches and executes SUB r1,r2,#4

Branches and executes ADD r1,r2,#4

# Module - 5

Q9b) Add Array of 16-bit numbers and Store in internal RAM.

```
        Area. Add, code, Readonly
        mov  R5, #3; 3 16 bit num
        mov  R0, #0;
        LDR  R1, =value1
Loop    LDR  R2, [R1], #2
        LDR  R3, mask
        AND  R2, R2, R3
        ADD  R0, R0, R2
        SUBS R5, R5, #1
        cmp  R5, #0
        BNE  Loop
        LDR  R4, =Result
        STR  R0, [R4]
        NOP
here    B here
mask    DCD 0x0000FFFF
value1  DCW 0x0001, 0x0002, 0x8003
        Area Data2, data, Readwrite
Result  DCD 0x0
        end
```

Result is R0 = 0x0001
+ 0x0002
+ 0x8003.
---
0x8006

## module-5

Q10a) Data Processing Instructions.

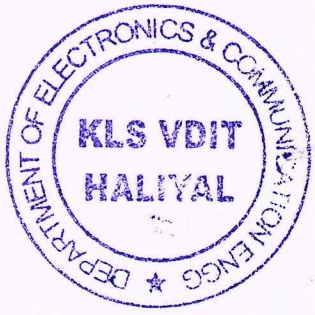
Explain any 5 from following types of DPIs

| DPIs Type | Instructions |
|---|---|
| i) Arithmetic | ADD<br>ADC<br>ADDW<br>SUB<br>SBC<br>SBCW<br>RSB<br>MUL etc |
| ii) 32bit types | SMULL<br>SMLAL |
| iii) Logical | AND, ORR, BIC, ORN<br>EOR |
| iv) Shift & rotate | ASR, LSL, LSR, ROR<br>RRX |

- Any 5 from each type - 2 marks each

Q10b)

```
        Area   sorting, code, readonly
        mov    r5, #4
loop1   mov    r6, r5
        ldr    r1, = 0x40000000
loop    ldr    r2, [r1], #4; fous numbers
        ldr    r3, [r1]
        cmp    r2, r3
        bls    loop2
        str    r2, [r1], #-4
        str    r3, [r1]
        add    r1, #4
loop2   subs   r6, r6, #1
        cmp    r6, #0
        bne    loop
        subs   r5, r5, #1.
        cmp    r5, #0
        bne    loop1
stop    B      stop
        end
```




r1 = 0x40000000

0x40000000 0x80
1 0x96
2 0x9
3 0x99

after

0x40000000 0x80
1 x91
2 x96
3 x99