El - Megha Lohar - 49398

PAGE:
DATE: / /

Sunbeam Institute of Information Technology, Pune

Microcontrollers and Interfacing

Rapid Fire Questions

Question 1

What is difference between microcontroller and microprocessor?

Microprocessor (μP)	Microcontroller (μC)
① μP is a general purpose computing machine / chip.	μC is a dedicated processing chip.
② μP contains CPU [ALU, registers] cache, and / or MMU.	μC contains CPU, RAM, ROM, peripherals (timers, gpio, adc, dac, spi, i2c, can etc). on same chip.
③ Usually works on high clock speed and hence higher power cosumption.	Usually works on lower clock speed and hence lower power cosumption.
④ It requires more space & cost.	It requires less space & cost.
⑤ System design is flexible. Designer can choose required amount of RAM, ROM & peripherals	μC can not customize a controller, but can choose from variety of controllers as per

EL Megha Lohar - 49398

PAGE:
DATE: / /

Question 2

What is difference between I/O mapped I/O and Memory mapped I/O ?

	Memory mapped IO	IO mapped IO
①	Memory devices (registers) are mapped as memory addresses (in memory addr space).	IO devices (io registers) are mapped as separate IO addresses (in diff. addr space).
②	IO regs are accessed with same instructions as of memory. e.g. MOV, LD/ST.	IO regs are accessed with special instructions. e.g. IN/OUT.
③	There is no differentiation between memory locations and IO regs -- except addrs.	Memory locations and IO regs are differentiated by different buses or control signal. e.g. IO/M.
④	Example ARM	Example x86 arch.

Question 3

What is pipeline ? What are limitations ? Explain pipeline in ARM architecture ?

E1_ Megha Lohar _ 49398

PAGE:
DATE: / /

Pipeline

- The pipeline is used to increase the processing speed of our processor.
- Instruction pipeline contains multiple independents units that process instruction partially. All these units can execute simultaneously and thus multiple instructions can be processed parallely. This is called as "instruction level parallelism".
- Example:
  - 2 stage : Fetch → Execute.
    e.g. AVR & PIC
  - 3 stage : Fetch → Decode → Execute.
    e.g. ARM7 & ARM-CM3.
  - 5 stage : Fetch → Decode → Execute → Memory → Write Back.
    : ARM 9.

Limitations

- Designing of the pipelined processor is complex.
- Instruction latency increases in pipelined processors.
- The throughput of a pipelined processor is difficult to predict.
- The longer the pipeline, worse the problem of Hazard for branch instructions.

Pipeline Problems / Hazards.

1) Control Hazards
2) Data Hazards
3) Structural Hazards.

Pipeline in ARM architecture :-

PAGE:
DATE: / /

ARM processors up to the ARM7 employ a simple 3-stage pipeline with the following pipeline stages.

1	fetch	decode	execute
2		fetch	decode	execute
3			fetch	decode	execute

Ans. **Question 4.**

What is ARM7 pipeline? What are limitations of pipeline?

ARM7 Core has a three-stage pipeline that increases instruction flow through the processor up to three times

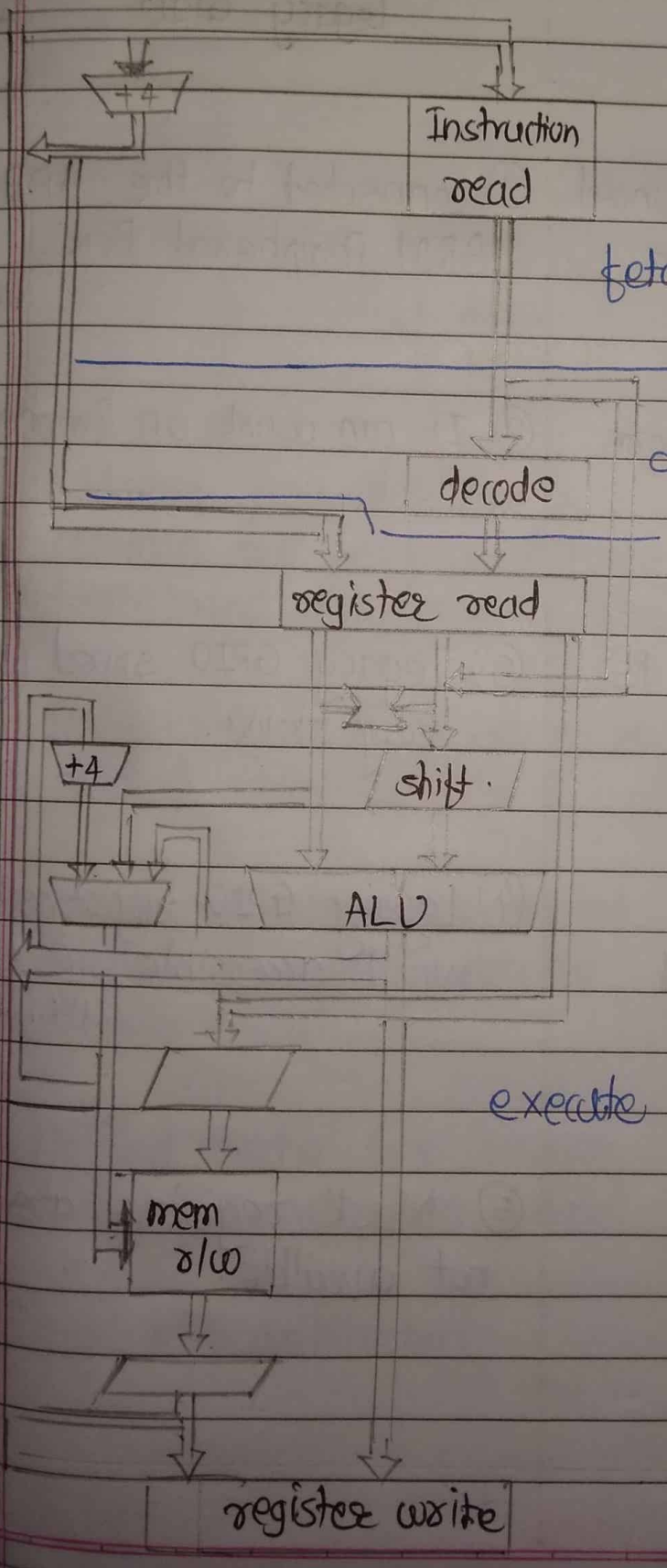
	Cycle 1	Cycle 2	Cycle 3	Cycle 4	Cycle 5
Instruction 1	Fetch	Decode	Execute		
Instruction 2		Fetch	Decode	Execute	
Instruction 3			Fetch	Decode	Execute

3 Stage Pipeline organization.

- Principal components
  - The register bank
  - The barrel shifter
    - Can shift or rotate one operand by any number of bits.
  - The ALU

PAGE:
DATE: / /

- The address register and incrementer
  - Select and hold all memory address and generate sequential addresses.
- The data registers.
- The instruction decoder and associated control logic



- Fetch - The instruction is fetched from memory & placed in the instruction pipeline.
- Decode - The instruction is decoded & the data path control signals prepared for the next cycle.
- Execute - The register bank is read, an operand shifted, the ALU result generated & written back into destination register.

Explorer EI_Megha Lohar_49398

PAGE:
DATE: / /

Question 5

What is Difference between Fast GPIO and Legacy GPIO ?

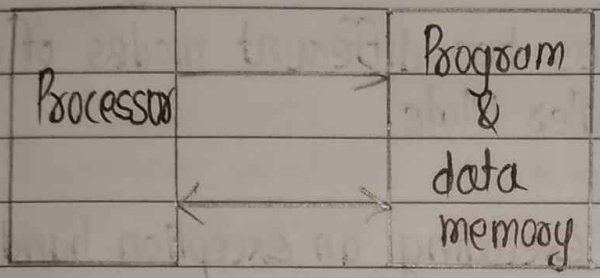
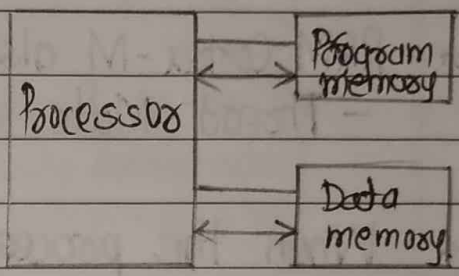
Fast GPIO	Legacy GPIO
① Connect to the Advanced High Performance Bus (AHB)	① Connected to the APB. ARM Peripheral Bus.
② It can work on Core clock.	② It can work on Peripheral clock.
③ Faist GPIO speed is very fast.	③ legacy GPIO speed is very slow.
④ GPIO registers are word, half-word & byte accessible	④ Legacy GPIO registers are word accessible.
⑤ Mask registers are available.	⑤ Mask registers are not availble.

EL_Megha Lohar-49398

PAGE:
DATE: / /

Question 6

What is Difference between Von-Neumann and Harvard?

Von-Neumann	Harvard.
	
① address and data bus is common for program memory (ROM) and data memory (RAM)	Separate address & data buses for program memory (ROM) & data memory (RAM).
② Instruction and data cannot be fetched simultaneously.	Instruction and data can be fetched simultaneously.
③ Op code and operands are not fetched in single cycle. So slower execution.	Opcode and operands are fetched in single cycle. So faster execution.
④ Unified Cache. (i.e. common cache for instruction & data).	Separate cache of instruction & data.
⑤ e.g. x86 architecture.	e.g. AVR.

PAGE:
DATE: / /

Question 7

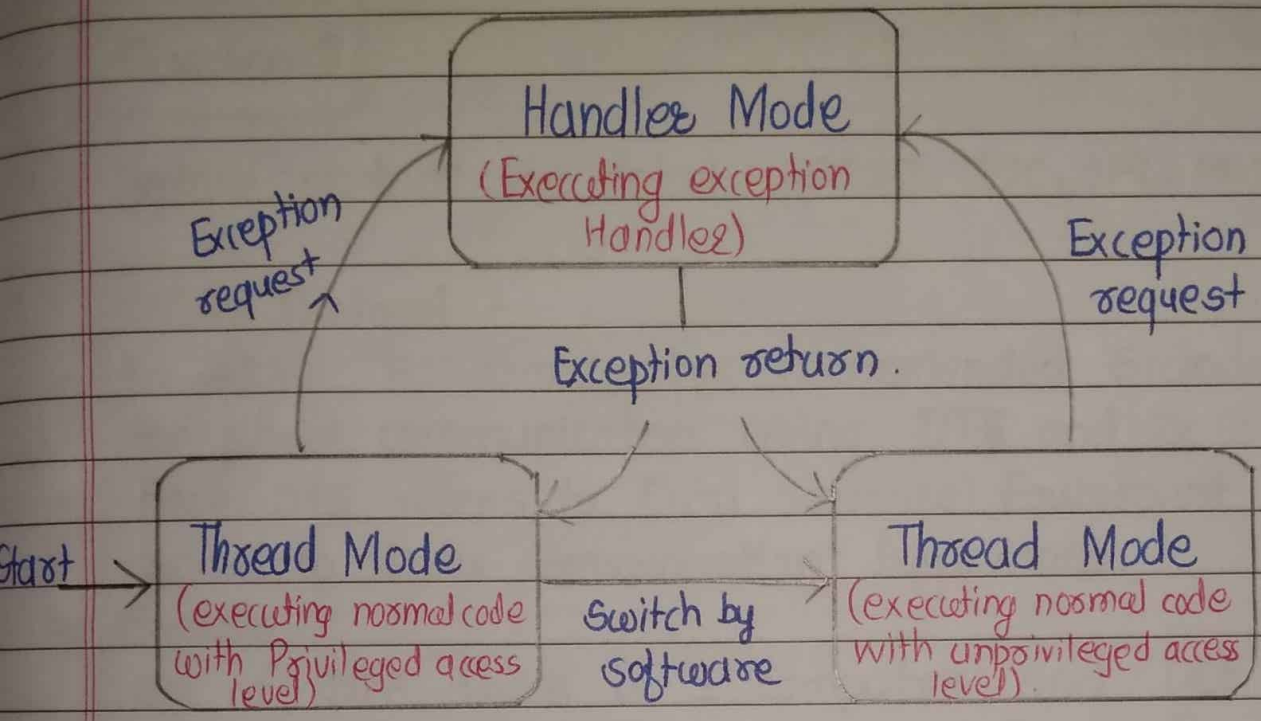
Explain ARM Cortex-M modes? Explain register banking. Why FIQ is faster than IRQ?

ARM Cortex-M modes:

* ARM Cortex-M also provides two different modes of operation - Thread Mode and Handler Mode.
* When the processor is executing an exception handler, it is in Handler mode. In all other cases it is in Thread mode. Handler mode always uses privileged access level.
* Thus the handler mode can manipulate system resources & protected memory regions.
* Application code uses the Thread mode.
* It can use either privileged or unprivileged access level.
* Thread mode with privileged access level is the default when processor starts up.
* Most applications do not need strong separation between user tasks and system tasks.

EL_Megha Lohar_49398

PAGE:
DATE: / /



El. Megha Lohar - 49398

Question 8

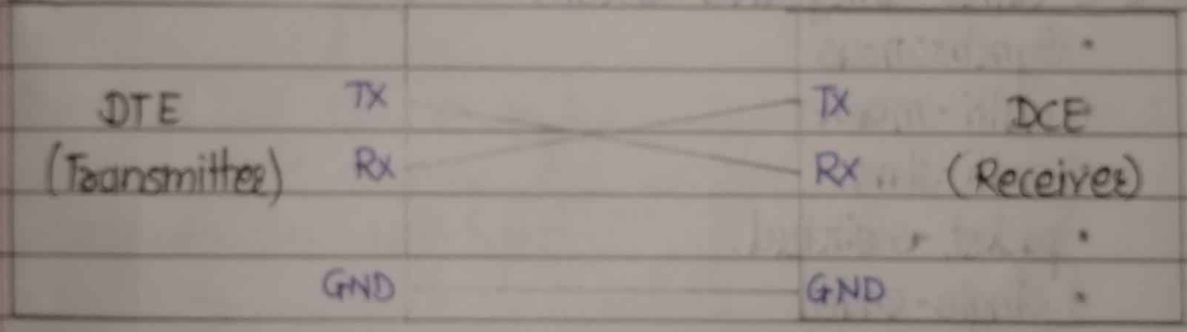
Explain protocols in detail : RS232, I2C, SPI, PS2

RS232 Protocol :-

- In RS232, 'RS' stands for Recommended Standard. It defines the serial communication using DTE and DCE signals.
- Here, DTE refers to Data Terminal Equipment and DCE refers to Data Communication Equipment.
- Example

eg of DTE device is a computer and DCE is a modem.

RS-232 Protocol

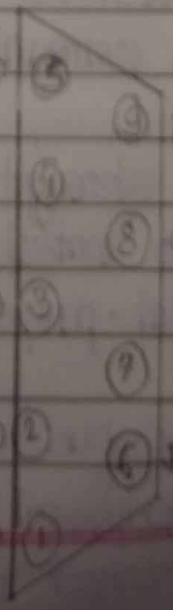


RS232 Pins.

RS-232 Pinout on DB25

2 - Transmit Data (TxD)
3 - Receive Data (RxD)
4 - Request to Send (RTS)
5 - Clear to send (CTS)
6 - Dataset ready (DSR)
7 - Signal Ground
8 - Data Carrier Detect (DCD)
15 - Transmit Clock
17 - Receive clock
20 - Data Terminal Ready
24 - Auxiliary Clock (DTR)

RS-232 DB-9 Male Pinout



Pin 1 : Data Carrier Detect
2 : Receive data
3 : Transmit data
4 : Data Terminal Ready
5 : Signal Ground
6 : Data set ready.
7 : Request to send
8 : Clear to Send
9 : Ring indicator

PAGE:
DATE: / /

Limitations of the standard.

- The limited transmission speed, relatively large voltage swing & large standard connectors motivated development of the universal serial bus (USB) which has displaced RS-232.

- Multi-drop connection among more than two devices is not defined.

- Also multi-drop have limitations in speed and compatibi

I2C Protocol:

* I^2C (Inter-Integrated Circuit) is a
  - Synchronous
  - multi-master
  - multi-slave
  - packet-switched.
  - single-ended.

serial computer bus invented in 1982 by Philips Semiconduct

* It is widely used for attaching lower-speed peripheral ICs to processors and microcontrollers in short-distance, intra-board communication.

* A particular strength of I^2C is the capability of a micro-controller to control a network of device chips with just two general-purpose I/O pins and software.

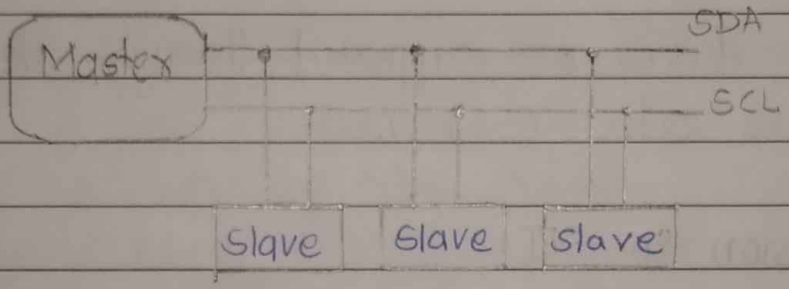
* Many other bus technologies used in similar appl^{ns}, suc as Serial Peripheral Interface Bus, require more pins and signals to connect multiple devices.

PAGE:
DATE: / /

* I^2C uses only two bidirectional open-drain lines, Serial Data Line (SDA) and Serial Clock Line (SCL), pulled up with resistors.

* Typical voltages used are +5V and +3.3V, although systems with other voltages are permitted.

Design



The bus has two roles for nodes : master & slave.
Master node :- node that generates the clock and initiates communication with slaves.

Slave node :- node that receives the clock and responds when addressed by the master.

* The bus is a multi-master bus. which means that any number of master nodes can be present.

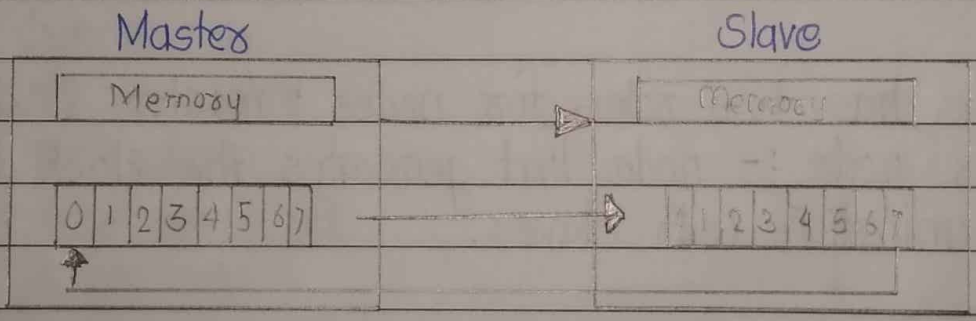
* Additionaly, master and slave roles may be changed between messages (after a STOP is sent).

PAGE:
DATE: / /

SPI

* The Serial Peripheral Interface bus (SPI) is synchronous.
* Serial communication interface specification used for short distance communication.
* The interface was developed by Motorola in 1980s.
* SPI devices communicate in full duplex mode using a master-slave-architecture with a single master.
* The master device originates the frame for reading and writing.
* Multiple slave devices are supported through selection with individual slave select (SS) lines.

Data Transmission in SPI



* Transmissions normally involve two shift registers of some given word size (may be 8 bits), one the master & one in the slave.

* They are connected in a virtual ring topology.

* Data is usually shifted out with the most-significant bit first. On the clock edge, both master & slave shift out a bit & output it on the transmission line to the counterpart.

EI_Megha Lohar_49398

PAGE:
DATE: / /

PS2 Protocol.

* The PS2 port is used for connecting keyboards and mice to a PC. This standard was introduced in 1987 by IBM with main purpose of replacing the serial keyboard & mouse.

* In our days ps/2 port was replaced by usb port, which is more easy to implement on a PC even though ps2 offers more capability and a greater speed.

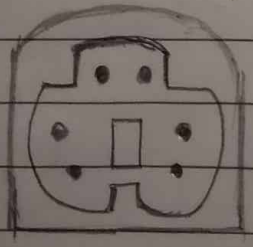
* When this standard was released the keyboard & mouse port had the same color, black like the connector or white like the connecting cable. Modern ps2 ports are colored different event thought the pinout configuration is the same.

* PS2 connector color code.

* Purple - Keyboard

* Green - Mouse

* Ps2 Mini-DIN Connector has 6 pins and carries a serial signal at frequencies starting from 10 up to 16 kHz with one start bit, one stop bit and one parity bit.



Pin 1 - Data
Pin 2 - Not connected
Pin 3 - Ground
Pin 4 - VDC +5V at 275 mA
Pin 5 - Clock
Pin 6 - Not connected

EL_Megha Lohar_49398

PAGE:
DATE: / /

Question 9

Q9. Which programmer you have used for ARM?

⇒ We use ST-LINK/V2 is an in-circuit debugger & programmer for STM32 microcontrollers.

⇒ The single-wire interface module (SWIM) and JTAG / serial wire debugging (SWD) interfaces are used to communicate with STM32 uc located on an application board.

PAGE:
DATE: 11

Question 10

What is wired AND in I2C ? What is clock stretching ? What is bus arbitration ?

Wired AND

- Wired AND is a connection method in which the o/p of two AND gates is connected together directly & this connection acts like a AND gate.
- In wired AND, a resistor is tied to VCC. The o/p is the bottom of the register. It will normally be high. But if you connect a transistor to the resistor & ground & activate it. The o/p will go low. Now just keep adding transistor.

Clock Stretching

- I2C devices can slow down comm^n by stretching SCL. During an SCL low phase, any I2C device on the bus may additionally hold down SCL to prevent it from rising again, enabling it to slow down the SCL clock rate or to stop. I2C comm^n for a while. This is also referred to as clock synchronization

Arbitration

- Several I2C multi-master can be connected to the same I2C bus & operate concurrently.
- By constantly monitoring SDA & SCL for start & stop cond^{ns}, they can determine whether the bus is currently idle or not. If the bus is busy, master delay pending I2C transfers until a stop condition indicates that the bus is free again.
- However, it may happen that two masters start a transfer at the same time. During the transfer, the masters constantly monitor SDA & SCL. If one of them detects that SDA is low when it actually be high. It assumes that another master is active & immediately stops transfer, is called Arbitration.

El_Megha Lohar - 49398

PAGE:
DATE: / /

Question 10.

What are ARM Processor states?

ARM Processor States
1. ARM State
2. Thumb State.

- ARM processors always start in ARM state.
- A processor that is executing Thumb instructions is operating in Thumb state.

ARM7TDMI processor has two operating states:
1) ARM state which executes 32-bit, word aligned ARM instructions

2) Thumb state which can execute 16-bit, halfword aligned THUMB instructions.

El-Megha Lohar - 49398

PAGE:
DATE: / /

Question 12.

Compare ARM, AVR, PIC & 8051 controllers.

	8051	PIC	AVR	ARM
Bus width.	8-bit for standard core	8/16/32 bit	8/32 bit	32-bit mostly also available in 64 bit
Communication Protocols	UART, USART, SPI, I2C	PIC, UART, USART, LIN, CAN, Ethernet, SPI, I2S	UART, USART, SPI, I2C (Special Purpose AVR support CAN, USB, Ethernet)	UART, USART, LIN, I2C, SPI, CAN, USB, Ethernet, I2S, DSP, SAI, IrDA
Speed	12 clock / instruction cycle.	4 clock / instruction cycle	1 clock / instruction cycle.	1 Clock / instruction cycle.
Memory	ROM, SRAM, FLASH,	SRAM, FLASH	Flash, SRAM, EEPROM.	Flash, SDRAM, EEPROM.
ISA	CISC	Some feature of RISC	RISC	RISC
Memory Architecture	Von-Neumann architecture	Harvard architecture	Modified.	Modified Harvard archi.
Power Consumption	Average	Low	Low	Low

PAGE:
DATE: / /

	8051	PIC	AVR	ARM
Families	8051 variants	PIC 16, PIC 17, PIC 18, PIC 24, PIC 32	Tiny, Atmega, Xmega, special purpose AVR	ARMv4-5,6,7 & series.
Community	Vast	Very Good	Very Good	Vast.
Manufacturers	NXP, Atmel, Silicon Labs, Dallas, Cypress, Infineon etc	Microchip Average	Atmel	Apple, Nvidia, Qualcomm, Samsung Electronics & TI etc.
Cost (as compared to features provided)	Very low	Average	Average	Low
Other Feature	known for its standard	Cheap	Cheap, effective	High speed operation vast.
Popular Microcontrollers	AT89C51, P89V51, etc	PIC 18XX8 PIC 16188X. PIC 32 MXX	Atmega8, 16 32 Arduino Community.	LPC2148, ARM Cortex M0 to ARM Cortex-M, etc

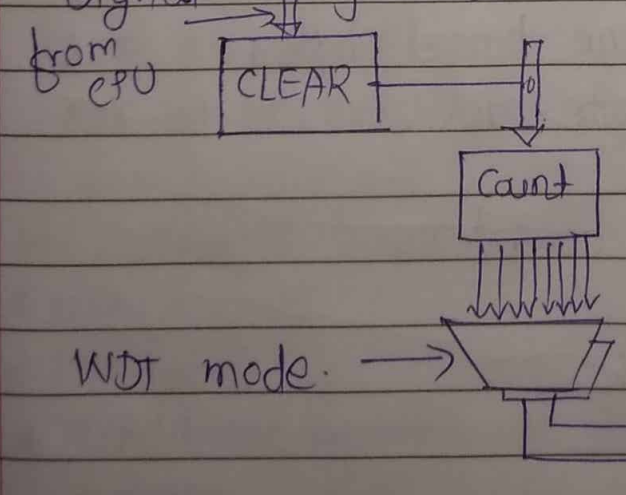
PAGE:
DATE: / /

Question 14

Explain watch dog timer and brown out detection.

Watchdog Timer (WDT)

* A Watchdog timer (WDT) is an embedded timing device that automatically prompts corrective action upon system malfunction detection.
* If software hangs or is lost, a WDT resets the system microcontroller via a 16-bit counter.
* A **WDT** is also known as a computer operating properly (COP) timer.
* The Watch Dog Timer (WDT) allows a Cortex® + M0 + based microcontroller to recover from catastrophic software errors such as a run-away code or a deadlock condition. Once enabled, the WDT requires the appln software to write a bit pattern to a WDT register within an allocated time period. If the WDT [CLEAR] register is not updated by the appln is not updated by the appn within the allotted time, a sim reset signal is generated.



Brown out detection:- Important fun^n to increase the reliability of a uc after start-up. Normally used to solve problems with the power supply.

El. Megha Lohar - 49398

PAGE:
DATE: / /

Question 15

Explain the Instruction Set for ARM in detail.

ARM Instruction Set

- The ARM has a load store architecture. meaning that all arithmetic and logical instructions take only register operands.
- They cannot directly operate on operands to memory.
- Separate instructions load and store instructions are used for moving data between registers & memory.

1. Data transfer instructions.
2. Arithmetic instructions.
3. Logical instructions.
4. Conditional branching and if-then instruction.
5. Barrel shifter.
6. Load - store instructions
   - Post increment
   - Pre increment
   - Pre increment + Inline barrel shifter.
   - Pre increment with write-back.
7. Load - store multiple.
8. Function call and stack operations
9. DSP instructions.
10. Miscellaneous instructions

El_Megha Lohar - 49398

PAGE:
DATE: / /

① Data transfer

mov rd, #K
mvn rd, #K
mrs rd, special-register
msr special-register, rs

② Arithmetic instructions

add rd, rm, rn
add rd, rs
sub rd, rm, rn
subs rd, rm, rn
mul rd, rm, rn
mla rd, rm, rm, ra
udiv rd, rn, rm.

③ Logical instructions

and rd, rn, rm
bic rd, rn, rm
tst rm, rn
orr rd, rn, rm
orn rd, rn, rm
eor rd, rn, rm.

④ conditional branching

* cmp rm, rn
* bxx label
* Conditional execution of ARM instructions
  - movxx rd, rs
  - addxx rd, rm, rn

EL_Megha Lohar_49398

* Thumb-2 if-then instruction
  - cmp rm, rn
  - ite gt
  - movgt rd, rm
  - movle rd, rn

⑤ Barrel shifter
LSL, ASR, LSR, ROR, RRX.
- Inline barrel shifter
  - mov rd, rs, lsl #k
  - mov rd, rs, lsr #k

⑥ Load store instructions
- Global variables
- ldr ra, =addr
- ldr rd, [ra]
- str rs, [ra]
- ldrb, ldrh, ldr.
- ldrsb, ldrsh.

Load / Store Pre / Post-increment
Pre-indexed ⇒ STR r0, [r1, #12]
Post-indexed ⇒ STR r0, [r1], #12.

⑦ Load store multiple.
LDMxx r10, {r0, r1, r4}
STMxx r10, {r0, r1, r4}

⑧ Function call and stack operation
- b label

- bl func_label
- stmfd sp!, {lr}
- push {lr}
- mov pc, lr
- ldmfd sp! {pc}
- pop {pc}.

⑨ DSP instructions

- Saturated Math
  - mov r0, #10
  - usat r1, #5, r0, lsl #1
  - usat r1, #5, r0, lsl #2

- SIMD instructions
  - ldr r1, = 0x11223344
  - ldr r2, = 0x44332211
  - qadd8 r0, r1, r2.

⑩ Miscellaneous instructions

- rev instruction
- sign extend
- bit-field extrac
- clear / insert bits

Question 16

Give overview of CAN protocol ? Where CAN find its main usage.

El_Megha Lohar_49398

CAN

- CAN or Controller area network is two wired half duplex high speed serial network technology. It is basically used in communication.
- Among different devices in a low radius region such as in an automobile.
- A CAN protocol is a CSMA-CD / ASM protocol or carrier sense multiple access collision detection arbitrations on message priority protocol.
- Collision detection ensures that the collision is avoided by selecting the messages based on their prescribed priority.
- It provides a signaling rate from 125 kbps to 1Mbps.

CAN protocol main usage:

* The protocol or CAN protocol eliminates the need for excessive writing by allowing electronic devices to communicate with each other along a single multiplex wire that connects each node in the network to the main dashboard.

- Examples:

Devices include engine controller (ECU),
transmission,
ABS,
lights,
power windows,
power steering,
instrument panel
& so on..

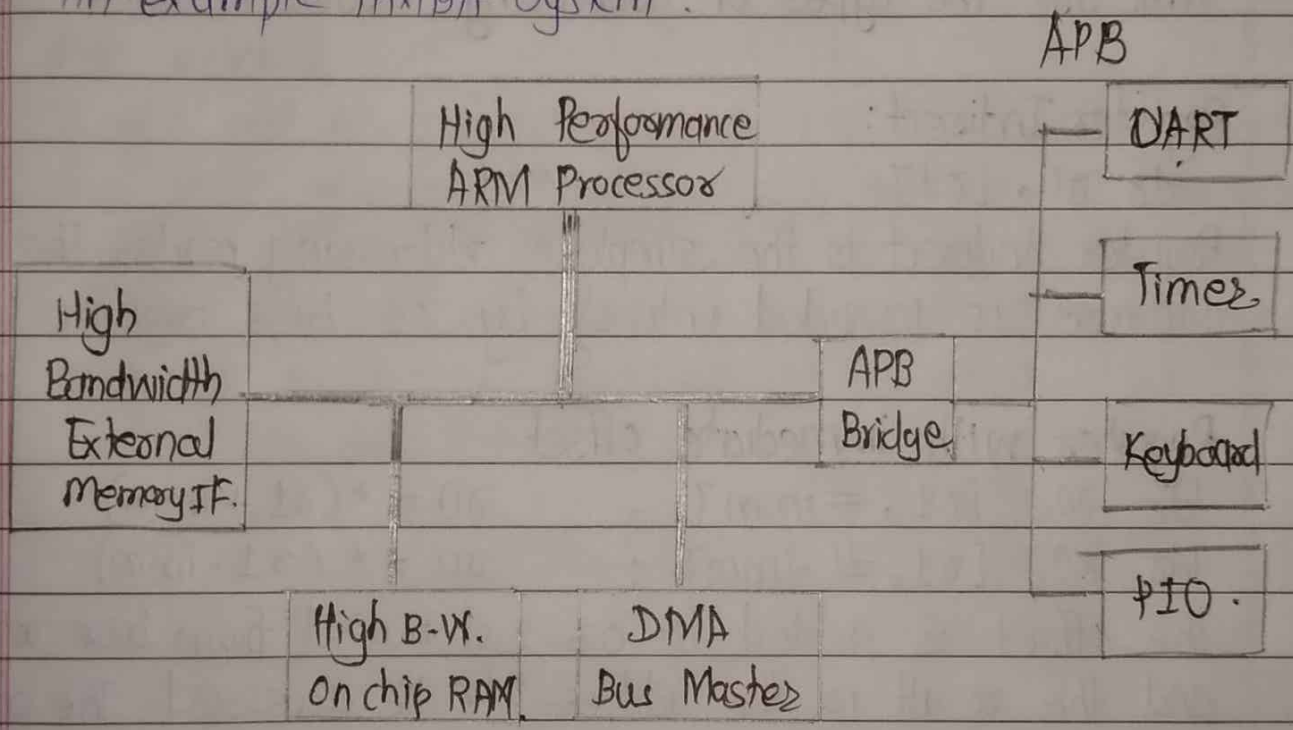
PAGE:
DATE: / /

Question 17

What is the use of the AMBA interface and explain in detail.

AMBA

- AMBA (Advanced Microcontroller Bus Architecture) is a freely-available, open standard for the connection and management of functional blocks in a system-on-chip (SoC).
- It facilitates right-first-time development of multi-processor designs, with large numbers of controllers & peripherals.
- An example AMBA System.



High Performance
Pipelined
Burst support.
Multiple Bus Masters.

Low Power.
Non-pipelined
Simple Interface.

PAGE:
DATE: / /

~~Features~~
~~Unaligned data transfers using byte.~~

Question 18.
What are the types of addressing modes in ARM?

Register Indirect:
ldr r0, [r1] ; r0 = *r1
Register indirect is the simplest addressing mode. The address is provided entirely by the base register

Register with immediate offset
ldr r0, [r1, #imm] ; r0 = *(r1 + imm)
ldr r0, [r1, #-imm] ; r0 = *(r1 - imm)
The offset is added to or subtracted from base register and the result is the address to be accessed. The offset can be in the range -255 - - + 4095, with small positive offsets offering the possibility of a 16 bit encoding.

Register with register offset
ldr r0, [r1, r2] ; r0 = *(r1 + r2)
ldr r0, [r1, -r2] ; r0 = *(r1 - r2)
The value of the offset register is added to or subtracted from the base register to form the

effective address.

Register with scaled register offset

ldr r0, [r1, r2, LSL #2] ; r0 = *(r1 + (r2<<2))

ldr r0, [r1, -r2, ASR #1] ; r0 = *(r1 - (r2>>1)) signed shift.

The scale is an operation performed by the barrel shifter on the offset before it is combined with the base register.

Pre-indexed

If you put an exclamation point after the close-bracket, then the base register is updated to contain the resulting effective address. This is called preindexed because the update occurs before the dereference. It corresponds roughly to the C preincrement operator.

ldr r0, [r1, #4]! ; r1 = r1+4
r0 = *r1

ldr r0, [r1, r2, lsl #2]! ; r1 = r1 + (r2<<2)
; r0 = *r1.

Post indexed

If you put the offset & scale outside the close-bracket, then the base register is used as the effective address, but the base register is then updated by the amount specified by the offset & scale. This is called post-indexed because the update occurs after the dereference. It corresponds roughly to the C postincrement operator.

ldr r0, [r1], #4 ; r0 = *r1
; r1 = r1+4

ldr r0, [r1], r2, lsl #2 ; r0 = *r1
; r1 = r1+(r2<<2).

El-Megha Lohar - 49398

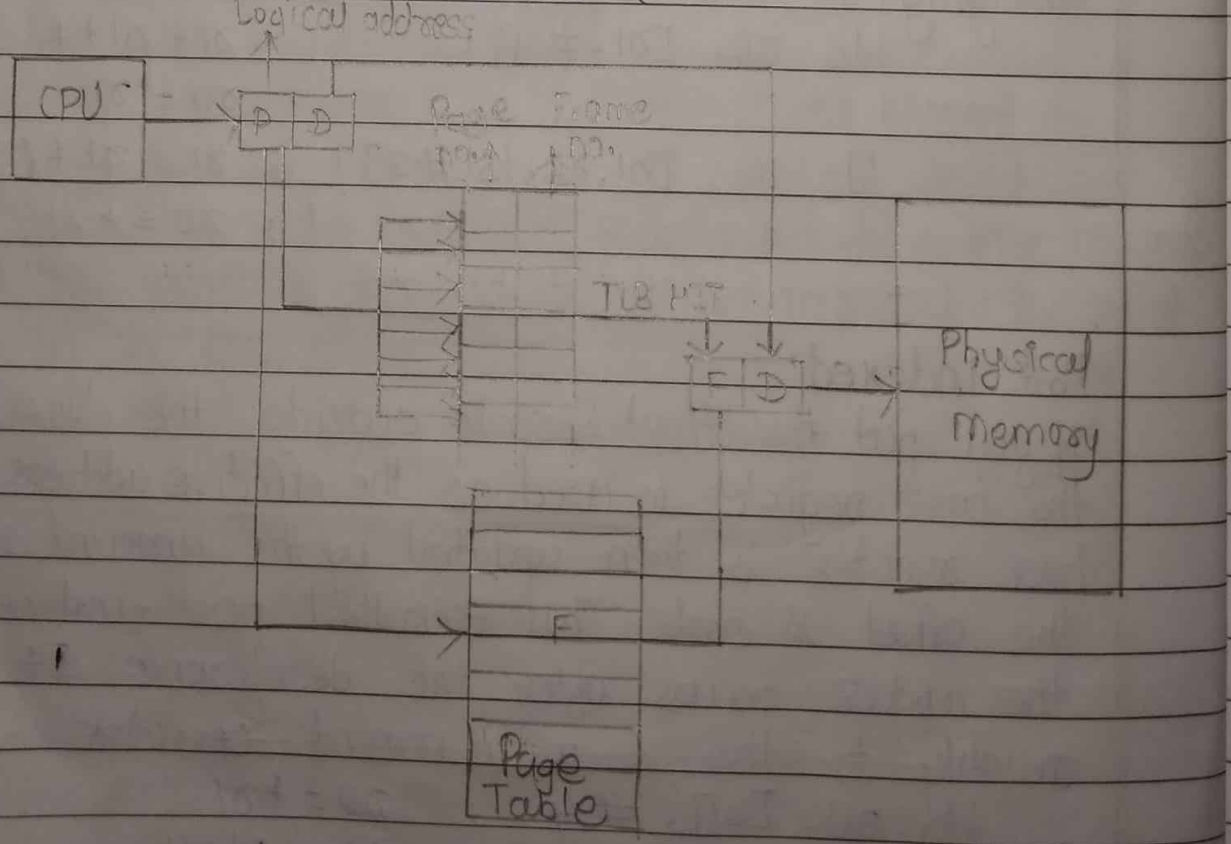
PAGE:
DATE: / /

Question 19

What is Translation Lookaside Buffer (TLB)?

Translation Lookaside Buffer (TLB)

- A translation lookaside Buffer (TLB) is a memory cache that is used to reduce the time taken to access a user memory location.
- It is a part of the chip's memory-management unit (MMU).
- The TLB stores the recent translations of virtual memory to physical memory and can be called an address-translation cache.
- A TLB may reside between the CPU & the CPU-cache, bet? CPU cache and the main memory or between the different levels of the multi-level cache.



El-Megha Lohar - 49398

Question 20

What is single issue multiple data (SIMD) processing?

- Single instruction, multiple data (SIMD) is a type of parallel processing in Flynn's taxonomy.
- ARM Neon technology is an advanced single instruction multiple data (SIMD) architecture extension for the Arm Cortex-A & Cortex-R series processors.
- Neon registers are considered as vectors of elements of the same data type, with Neon instructions operating on multiple elements simultaneously.
- ~~Neon~~ NEON is a wide SIMD data processing architecture.
  - Extension of the ARM instruction set (v7-A).
  - 32 x 64 bit wide registers.
- NEON instruction perform "Packed SIMD" processing.

Question 21

How will you allow Thumb C code to call the ARM assembly code?

ARM - Thumb interworking is the name given to the method of linking ARM & Thumb code together for both assembly & C/C++.

PAGE:
DATE: / /

- It handles the transmission between the two states. Extra code, called veneer, is sometimes needed to carry out the transition. ATPCS defines the ARM & Thumb procedure call standards.
- To call Thumb routine from ARM routine, the core has to change state. This state change is shown in the T bit of the cpsr.
- The Bx & BLX instructions cause a switch between ARM & Thumb state while branching to routine.
- The Bx lr instruction returns. From routine, also with a state switch if necessary.
- The BLX Instruction was introduced in ARMvST. On ARMv4T cores the linker uses or veneer to switch state on subroutine call.
- Instead of calling the routine directly, the linker calls the veneer, which switches to Thumb state using the BX instruction.
- There are 2 versions of the BX & BLX instructions : an ARM instruction & Thumb equivalent.
- The ARM Bx instruction enters Thumb state only if bit 0 of the address in Rn is set to binary 1; otherwise it enters ARM state.
- The Thumb BX instruction does the same

Syntax : Bx Rm
Bx Rm | label

BX	Thumb version branch exchange	PC = Rn & 0xfffffffe T = Rn[0]
BLX	Thumb version of branch exchange with link	lr = (instru. addr after BLX) + 1 PC = label, T = 0 PC = Rm & 0xfffffffe, T = Rm[0]

PAGE:
DATE: / /

Question 22

What is the use of 'SWI' in ARM assembly?

The SWI instruction causes a SWI exception. This means that the processor state changes to ARM, the processor mode changes to Supervisor, the CPSR is saved to the Supervisor Mode SPSR, and execution branches to the SWI vector.

Software Interrupt

Syntax

SWI {cond} immed

cond - is an optional condition code.

immed - is an expression evaluating to an integer in the range.

- 0 to $2^{24}-1$ (a 24 bit value) in an ARM instruction.
- 0 to 255 (an 8 bit value) in a 16-bit Thumb instruction.

Question 23.

Tell about the Exception Handling in ARM processors. What does the ARM Core do automatically for every exception?

Exception types:

Reset
Non-maskable Interrupts (NMI)
Faults
PendSV
SVCall
External Interrupt
SysTick Interrupt.

PAGE:
DATE: 11

Question 24.

Explain memory map in ARM architecture.

- ARM Memory Organization. The Cortex-M3 & Cortex-M4 have a predefined memory map. This allows the built-in peripherals, such as the interrupt controller and the debug components, to be accessed by simple memory access instructions. Thus, most system features are accessible in program code.

- The predefined memory map also allows the Cortex-M3 processor to be highly optimized for speed and ease of integration in system-on-a-chip (SoC) designs.

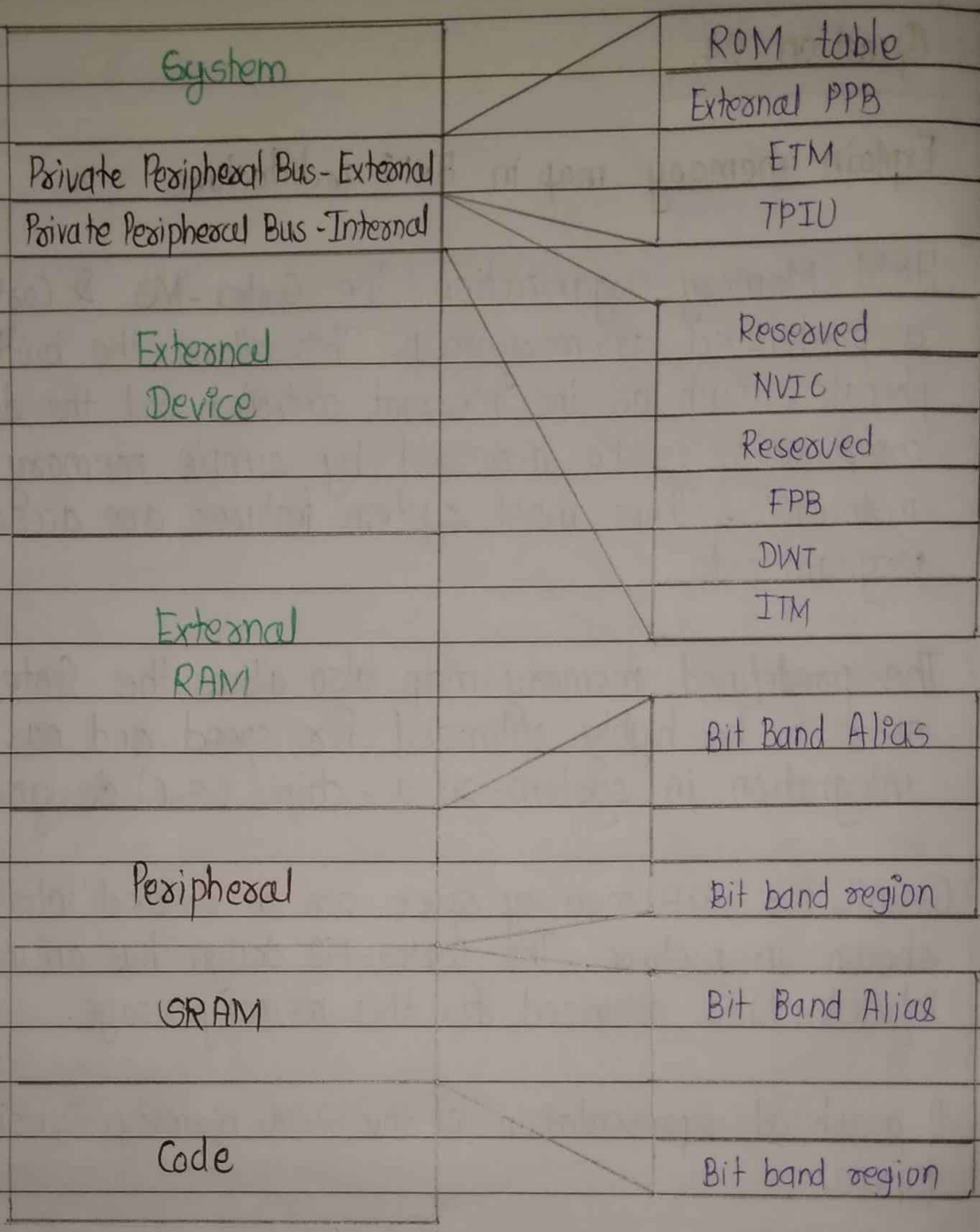
- Overall, the 4GB memory space can be divided into ranges as shown in picture. The Cortex-M3 design has an integrated bus infrastructure optimized for this memory usage.

- A graphical representation of the ARM memory is shown in figure.

- System -

- Private Peripheral Bus - External ⇒ Provides access to :-
  - the Trace Port Interface Unit (TPIU).
  - the Embedded Trace Macrocell (ETM).
  - the ROM table.
  - Implementation-specific areas of the PPB memory map.

PAGE:
DATE: / /



ARM Memory Map

- Private Peripheral Bus - External
  Provides access to:
  - the Instrumentation Trace Macrocell (ITM),
  - the Data Watchpoint and Trace (DWT),
  - the Flashpatch and Breakpoint (FPB),
  - the System Control Space (SCS), including the MPU & the Nested Vectored Interrupt Controller (NVIC).

PAGE:
DATE: 11

- **External Device:**
  This region is used for external device memory.

- **Peripheral:**
  This region includes bit band and bit band alias areas.
  - Peripheral Bit-band alias
    Direct accesses to this memory range behave as peripheral memory accesses, but this region is also bit addressable through bit-band alias.
  - Peripheral Bit-band region
    Data accesses to this region are remapped to bit band region. A write operation is performed as read-modify-write.

- **SRAM:-**
  - This executable region is for data storage. Code can also be stored here. This region includes bit band and bit band alias areas.
    - SRAM Bit-band alias
      Direct accesses to this memory range behave as SRAM memory accesses, but this region is also bit addressable through bit-band alias.
    - SRAM-bit-band region.
      Data accesses to this region are remapped to bit band region. A write operation is performed as read-modify-write.

- **Code:-**
  - This executable region is for program code. Data can also be stored here.

PAGE:
DATE: / /

Question 25

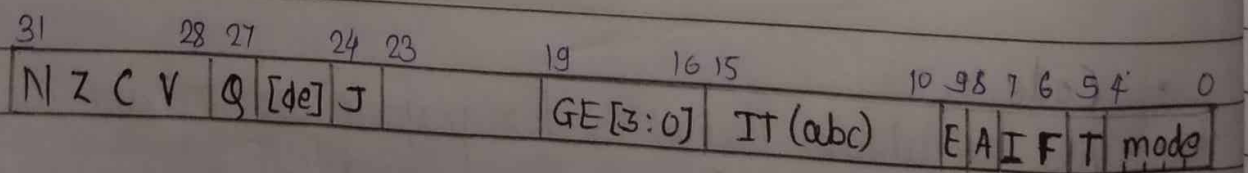
Explain difference between ARM or Thumb Mode?

	ARM Mode	Thumb Mode
①	ARM mode is status that the T-bit of CPSR (Current Program Status Register) is 0.	① Thumb mode is status that the T bit of CPSR (Current Program Status Register is 1.
②	ARM mode provides 32 bit instruction set.	② Thumb mode fetchs instructions by 2 bytes.
③	ARM mode fetchs instructions by 4 bytes	③ Thumb mode fetchs instructions by 2 bytes.

Question 26

Explain about the bits in CPSR register.

Program Status Register.



PAGE:
DATE: / /

Condition code Flags.

N = Negative result from ALU
Z = Zero result from ALU
C = ALU operation Carried Out.
V = ALU operation oVerflowed.

Stickly Overflow flag - Q flag

Indicates if saturation has occurred.

SIMD Condition code bits - GE [3:0]

Used by some SIMD instructions.

IF THEN status bits - IT [abcde]

Controls conditional execution of Thumb instructions.

T bit

T = 0 : Processor in ARM state.
T = 1 : Processor in Thumb state.

J bit

J = 1 : Processor in Jazelle state.

Mode bits

Specify the processor mode.

Interrupt Disable bits

I = 1 : Disables IRQ
F = 1 : Disables FIQ

PAGE:
DATE: / /

E bit

E = 0: Data load / store is little endian

E = 1: Data load / store is bigendian.

A bit

A = 1: Disable imprecise data aborts.

Question 27.

What is an interrupt ? Write a C program to handle external interrupt for an ARM chip.

* Interrupt :- An interrupt as the Name suggest Interrupt the uC from whatever it is doing and draws the attention to perform a special task.

* An interrupt is the Automatic transfer of software execution in response to the hardware event that is asynchronous with the current software execution.

* An interrupt is a computer or Arch signal that tells the computer to stop running the current program to the New one can be started or a circuit that corners such signal.

Header file

```
#ifndef EXTI_H_
# define EXTI_H_
# include "stm32f4xx.h"
```

EI- Megha Lohar - 49398

```
#define SWITCH_EXTI        0
#define SWITCH_EXTI_IRQn   EXTI0_IRQn
#define SWITCH_CLK_EN      RCC_AHB1ENR_GPIOAEN_Pn
#define SWITCH_GPIO        GPIOA
#define SWITCH_PIN         0

// SWITCH EXTICR [3:0] = 000
#define SWITCH_EXTICR      SYSEFG -> EXTICR[0]
#define SWITCH_EXTIIR_MSK  (BV(3) | BV(2) | BV(1) | BV(0))

void switchExtiInit (void);

#endif
```

Source file :

```
#include "Exti.h"
#include "led.h"
void switchExtiInit (void) {
  // configure gpio (PA.0) for switch
    RCC -> AHB1ENR |= BV (switch_clk_EN);

  // set switch pin as Input pin.
SWITCH_GPIO -> MODER &= ~(BV(switchpin*2+1) |
                          BV(switchpin*2));

  // set low speed
SWITCH_GPIO -> PUPDR &= ~(BV(switchpin*2+1) |
                          BV(switchpin*2));
```

PAGE:
DATE: / /

```
//configure no pull up.
SWITCH_GPIO -> PUPDR &= ~(BV(switchpin2*1) |
                          BV(switchpin*2));

//SYSCFG EXTI CONFIG --> PA
//enable clock for syscfg
RCC -> APB2 |= BV ( RCC_APB2ENR_SYSCFG EN_Pos)

// switch_ExticR &=~ SWITCH EXTIR_Msk.

//ext_intr controller config.
EXTI -> FTSR |= BV (SWITCH_EXTI);
EXTI -> RTSR &= ~BV (SWITCH_EXTI);

// unmask exti 0 from IMR
EXTI -> IMR |= BW (SWITCH_EXTI);

//ext intr NVIC config.
NVIC_Enable IRQ (switch_EXTI_IRQn);

}

Void EXTI0_IRQ Handler (void) {
 EXTI -> PR |= BV (SWITCH_EXTI);
}.
```

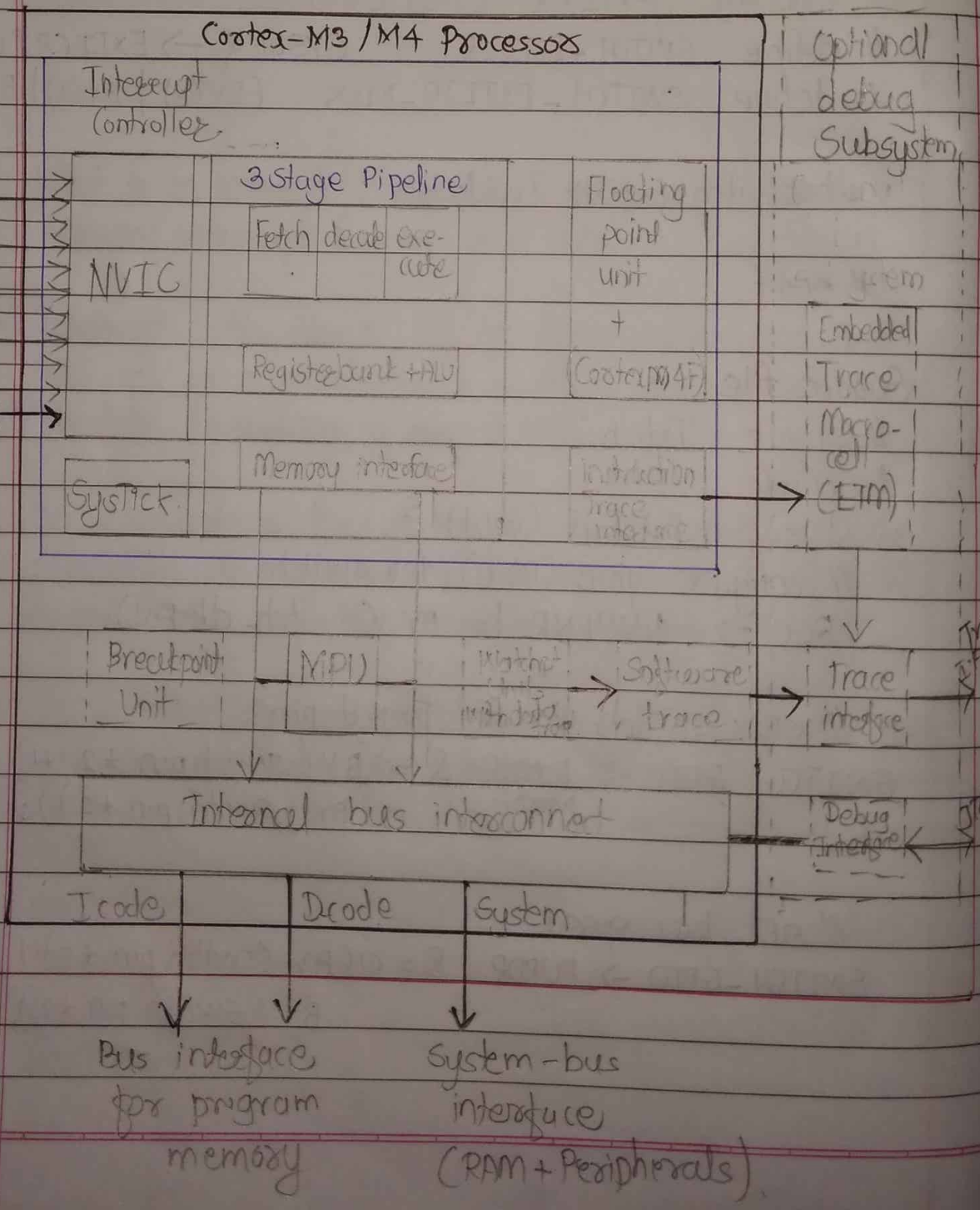
El_Megha Lohar_49398

PAGE:
DATE: / /

Question 28

Explain ARM Cortex-M3 architecture.

Cortex-M3/M4 Integration (Processor System)



PAGE:
DATE: / /

Cortex-M3 architecture:

* Cortex M3 use a 32 bit architecture.
* The ISA in the Cortex-M processor is called the Thumb ISA & is based on Thumb-2 Technology which supports a mixture of 16-bit & 32-bit architectures.

* 3-stage Pipeline
* Harvard bus architecture
* 32-bit addressing, supporting 4GB of memory space.
* On chip bus interface based on ARM AMBA
* AHB-Lite bus interface.
* Fixed memory map.
* 4-240 interrupts.
  → Configurable priority levels
  → Non-Maskable Interrupt support.
  → Debug & sleep Control.

* Serial wire or JTAG debug.
* Optional ETM.