EE6008 MICROCONTROLLER BASED SYSTEM DESIGNL T P C

3 0 0 3

OBJECTIVES:

- To introduce the architecture of PIC microcontroller
- To educate on use of interrupts and timers
- To educate on the peripheral devices for data communication and transfer
- To introduce the functional blocks of ARM processor
- To educate on the architecture of ARM processors

UNIT I INTRODUCTION TO PIC MICROCONTROLLER 9

Introduction to PIC Microcontroller–PIC 16C6x and PIC16C7x Architecture–PIC16cxx–- Pipelining - Program Memory considerations – Register File Structure - Instruction Set - Addressing modes – Simple Operations.

UNIT II INTERRUPTS AND TIMER 9

PIC micro controller Interrupts- External Interrupts-Interrupt Programming–Loop time subroutine - Timers-Timer Programming– Front panel I/O-Soft Keys– State machines and key switches– Display of Constant and Variable strings. 98

UNIT III PERIPHERALS AND INTERFACING 9

I2C Bus for Peripherals Chip Access– Bus operation-Bus subroutines– Serial EEPROM—Analog to Digital Converter–UART-Baud rate selection–Data handling circuit–Initialization - LCD and keyboard Interfacing -ADC, DAC, and Sensor Interfacing.

UNIT IVINTRODUCTION TO ARM PROCESSOR 9

ARM Architecture –ARM programmer‘s model –ARM Development tools- Memory Hierarchy – ARM Assembly Language Programming–Simple Examples–Architectural Support for Operating systems.

UNIT V ARM ORGANIZATION 9

3-Stage Pipeline ARM Organization– 5-Stage Pipeline ARM Organization–ARM Instruction Execution- ARM Implementation– ARM Instruction Set– ARM coprocessor interface– Architectural support for High Level Languages – Embedded ARM Applications.

TOTAL: 45 PERIODS

OUTCOMES:

- To understand and apply computing platform and software for engineering problems.
- To understand ethical issues, environmental impact and acquire management skills.

TEXT BOOKS:

1. Peatman,J.B., ―Design with PIC Micro Controllers‖PearsonEducation,3rdEdition, 2004.
2. Furber,S., ―ARM System on Chip Architecture‖ Addison Wesley trade Computer Publication, 2000.

REFERENCE:

1. Mazidi, M.A.,―PIC Microcontroller‖ Rollin Mckinlay, Danny causey Printice Hall of India, 2007.

UNIT I

INTRODUCTION TO PIC MICROCONTROLLER

Introduction to PIC Microcontroller–PIC 16C6x and PIC16C7x Architecture– PIC16cxx–- Pipelining – Program Memory considerations – Register File Structure – Instruction Set – Addressing modes – Simple Operations.

PART-A

1. What is PIC microcontroller?

PIC stands for Peripheral Interface Controller coined by Microchip Technology to identify its single chip microcontrollers. These devices have been phenomenally successful in 8-bit microcontroller market. The main reason is that Microchip Technology has constantly upgraded the device architecture and added needed peripherals to the microcontroller to 'suit customers' requirements.

2. What are the PIC16C6X Microcontroller Core Features?

High performance RISC CPU

• Only 35 single word instructions to learn

• All single cycle instructions except for program branches which are two-cycle

• Operating speed: DC - 20 MHz clock input DC - 200 ns instruction cycle

• Interrupt capability

• Eight level deep hardware stack

• Direct, indirect, and relative addressing modes

• Power-on Reset (POR)

• Power-up Timer (PWRT) and Oscillator Start-up Timer (OST)

• Watchdog Timer (WDT) with its own on-chip RC oscillator for reliable operation

• Programmable code-protection

• Power saving SLEEP mode

• Selectable oscillator options

- Low-power, high-speed CMOS EPROM/ROM technology
- Fully static design
- Wide operating voltage range: 2.5V to 6.0V
- Commercial, Industrial, and Extended temperature ranges
- Low-power consumption:

  < 2 mA @ 5V, 4 MHz

  15mA typical @ 3V, 32 kHz

  < 1mA typical standby current

3.What are the PIC16C6X Microcontroller Peripheral Features

- Timer0: 8-bit timer/counter with 8-bit prescaler
- Timer1: 16-bit timer/counter with prescaler,can be incremented during sleep via external crystal/clock
- Timer2: 8-bit timer/counter with 8-bit period register, prescaler and postscaler
- Capture/Compare/PWM (CCP) module(s)
- Capture is 16-bit, max resolution is 12.5 ns, Compare is 16-bit, max resolution is 200 ns,PWM max resolution is 10-bit.
- Synchronous Serial Port (SSP) with SPI and I^2C
- Universal Synchronous Asynchronous ReceiverTransmitter (USART/SCI)
- Parallel Slave Port (PSP) 8-bits wide, with external RD, WR and CS controls.
- Brown-out detection circuitry for Brown-out Reset (BOR)

4.What are the PIC 16c6x family device?

For the PIC16C6X family of devices, there are four device ―types‖ as indicated in the device number:

1. C, as in PIC16**C**64. These devices have EPROM type memory and operate over the standard voltage range.

2. **LC**, as in PIC16**LC**64. These devices have EPROM type memory and operate over an extended voltage range.

3. **CR**, as in PIC16**CR**64. These devices have ROM program memory and operate over the standard voltage range.

4. **LCR**, as in PIC16**LCR**64. These devices have ROM program memory and operate over an extended voltage range.

5. What is the Low-end Architectures ?

Microchip PIC microcontrollers are available in various types. When PIC − M icroMCU first became available from General Instruments in early 1980‘s, the microcontroller consisted of a very simple processor executing 12-bit wide instructions with basic I/O functions. These devices are known as low-end architectures. Some of the low-end device past numbers are 12C5XX, 16C5X, and 16C505

6.**What is Mid-range Architectures?**

Mid-range Architectures are built by upgrading low-end architecture with more number of peripherals, more numbers of register and more data memory. Some of the mid-range devices are 16C6X 16C7X, 16F87X ↑Program memory type.

7.What is Brown Out delay

When the power supply drops below a certain voltage (4 v in case of PIC) it causes PIC to reset .

8.How many ports in the PIC 16c6xx

There are five ports are available in the PIC 16c6x

9. Addressing modes of the PIC microcontroller

The PIC microcontrollers support only TWO addressing modes .They are

(i)Direct Addressing Mode (ii) Indirect Addressing mode

10. Instruction set of the PIC microcontroller

The instruction set of PIC is divided into Three basic categories.

They are

- Byte oriented Instructions
- Bit oriented Instructions
- Literal and Control Instructions

11. Classify instruction set of the PIC microcontroller

All the instructions of the PIC microcontroller are classified into nearly 9 groups. They are given below with examples

(i) Arithmetic Operations
(ii) Logical Instructions
(iii)Increment/Decrement Instructions
(iv)Data Transfer instructions :
(v) Clear Instructions
(vi)Rotate Instructions
(vii) Branch Instructions
(viii) Miscellaneous
(ix)Bit manipulation.

12. How many modules in the Memory of the PIC micocontroller

The memory module of the PICcontroller has three memory blocks.

(i) Program memory (ii) Data memory (iii) Stack

13. How to select the memory bank in the PIC microcontroller

RP1 and RP0 bits are used for selecting the bits of the memory bank

RP1:RP0 (STATUS) = 00 → Bank0
= 01 → Bank1
= 10 → Bank2
= 11 → Bank3

14. Explain Pipelining with example?

These devices are known as low-end architectures. Some of the low-end device past numbers are 12C5XX, 16C5X, and 16C505

15. What are the CPU REGISTERS

(i) Working Register-W (Similar to Accumulator)

(ii) Status Register

(iii)FSR – File Select Register (Indirect Data memory address pointer)

(iv) INDF

(v) Program Counter

16. What is Working Register?

Working Register is used by many instructions as the source of an operand. It also serves as the destination for the result of instruction execution and it is similar to accumulator in other cs and ps.

17. Status Register:

This is an 8-bit register which denotes the status of ALU after any arithmetic operation and also RESET status and the bank select bits for the data memory.

C: $\overline{\text{Carry}}$/borrow bit

DC: Digit $\overline{\text{carry}}$/borrow bit

Z: Zero bit

NOT_PD : Reset Status bit (Power-down mode bit)

NOT_TO : Reset Status bit (tme- out bit)

RPO: Register bank Select

The bits 7 and 6 of Status Register are unused by 16c6x/7x. The _C‘ bit is set when two 8-bit operands are added together and a 9-bit result occurs. This 9-bit is placed in the carry bit. The DC or Digit carry bit indicates that a carry from the lower 4 bits occurred during an 8-bit addition.

Example: 0011 1000

0011 1000

0111 0000

Here DC=1 as a result of the carry from the bit 3 to the bit 4 position.

The Z or zero bits is affected by the execution of arithmetic or logic instructions.

18. What is FSR – (File Select Register):

It is the pointer used for indirect addressing. In the indirect addressing mode the 8-bit register file address is first written into FSR. It is a special purpose register that serves as an address pointer to any address throughout the entire register file.

19. What isINDF – (Indirect File):

It is not a physical register addressing but this INDF will cause indirect addressing. Any instruction using the INDF register actually access the register pointed to by the FSR.

20. What is PROGRAM COUNTER

PIC has a 13 bit program counter in which PCL is the lower 8-bits of the PC and PCLATH is the write buffer for the upper 5 bits of the PC.

PCLATH (program counter Latch can be read or from or written to without affecting the Program Counter(PC).The upper 3 bits of PCLATH remain zero.It is only when PCL is written to that PCLATH is automatically written into the PC at the same time.

21. What is the PARALLEL I/O Ports of PIC Microcontroller?

Most of the PIC16cx/7x family controllers have 33 I/O lines and five I/O ports They are PORT A, PORT B, PORT C , PORT D and PORT E.

PORT A:

Port A is a 6-bit wide bi-directional port. Its data direction register is TRISA setting TRISA bit to 1 will make the corresponding PORT A Pin an input. Clearing a TRIS a bit will make the corresponding pin as an output.

PORT B:

Port B is an 8-bit wide, bi-directional port. Four of the PORT B pins $RB_7 – RB_4$ have an interrupt-on- change feature. Only the pins configured as inputs can cause this interrupt to occur.

PORT C:

Port C is an 8-bit wide, bidirectional port. Bits of the TRISC Register determine the function of its pins. Similar to other ports, a logic one 1 in the TRISC Register configures the appropriate port pin as an input.

PORT D:

Port D is an 8-bit wide bi-directional port. In addition to I/O port, Port D also works as 8-bit parallel slave port or microprocessor port. When control bit PSPMODE (TRISE:4) is set.

PORT E:

It is a 3-bit bi-directional port. Port E bits are multiplexed with analog inputs of ADC and they serve as control signals (RD , WR, CS) for parallel slave port mode of operation.

22. What is the Program memory?

The PIC has 4k x14 program memory space (0000H-0FFFH).It has a 13 bit Program counter(PC) to access any address (2^{13}=4k). This PIC family uses 13-bit program counter allowing the controllers to an 8k-program memory without changing the CPU structure.

23. What is Data Memory?

The data memory of PIC is partitioned into multiple banks which contain the general purpose registers and the Special function Registers.(SFRs).The bits RP1 and RP0 bits of the status register are used to select these banks.Each bank extends upto 7FH(128 Bytes).The lower bytes of the each bank are reserved for the Special Function Registers.Above the SFRs are general purpose registers implemented as static RAM.

24. Write the status register of 16c6x.

0	**0**	**RPO**	$\overline{TO}$	$\overline{PD}$	**Z**	**DC**	**C**

Status bit	**Description**	**Remark**
RPO	Register bank select	To select 2 banks one bit RPO is sufficient
$\overline{TO}$	Reset status bit(Time-out bit)	Only readable
$\overline{PD}$	Reset status bit(Power down bit)	Only readable
Z	Zero bit	
DC	Digit carry/Borrow bit	
C	carry/Borrow bit	

PART-B

1. Explain What is an instruction pipelining in PIC?
2. Explain What is the brown out feature in PIC microcontroller. How do PIC microcontrollers support the power saving options

3. Explain the Register file structure of PIC 16c6x.
4. Explain the Architecture of the PIC 16c6x with neat diagram.
5. What are the various addressing modes in PIC microcontroller? What is the role of INDF in indirect addressing mode?
6. Explain the instruction set of PIC16c6x?
7. Justify the statement ― Once the watchdog timer is enabled(disable), it is not possib le to make it off(on)‖.
8. Explain the memory organization of the PIC microcontroller.

UNIT – II
INTERRUPTS AND TIMER

PIC micro controller Interrupts- External Interrupts-Interrupt Programming–Loop time subroutine – Timers-Timer Programming– Front panel I/O-Soft Keys– State machines and key switches– Display of Constant and Variable strings.

PART-A

1.What are the Two papermeters of interrupt source i

- The minimum time interval between interrupts from source, denoted by TP_i
- The maximum time it takes the CPU to execute the interrupt source's handler subroutine and its call from within **Intservice** , denoted by T_i.

The minimum time interval between interrupt for a given interrupts source is determined by the application

9600Bd UART , Each 8-bit character is framed between a start bit and stop bit. When a bit time of 1/9600, each 10 bit frame (i.e., each character) can arrive 10/9600 seconds apart:
TP_{UART}= 10/9600 second = 10,000,000/9600 □*s* =1042 □*s* .

The maximum time interval (Capture/Compare /PWM)facility must be dealt with before another CCP interrupt occurs.
TPCCP= 1/4000 second = 1,000,000/4000 □*s* =250 □*s*

2.What is GIE ?

Global interrupt enable bit, GIE (INTCON) enables all un-masked interrupts (if GIE set) or disables (if GIE cleared) all interrupts. When bit GIE is enabled, and an interrupt flag bit and mask bit are set, the interrupt will vector immediately. Individual interrupts can be disabled through their corresponding enable bits in the INTCON register. GIE is cleared on reset.

3. What is WATCH DOG TIMER (WDT):

The Watchdog Timer is a free running on-chip RC oscillator which does not require any external components. This RC oscillator is separate from the RC oscillator of the OSC1/CLKIN pin. That means that the WDT will run, even if the clock on the OSC1/CLKIN and OSC2/CLKOUT pins of the device has been stopped, for example, by execution of a SLEEP instruction. During normal operation, a WDT time-out generates a device reset. If the device is in SLEEP mode, a WDT time-out causes the device to wake-up and continue with normal operation. The WDT can be permanently disabled by clearing configuration bit WDTE.

4.Write the WATCH DOG TIMER REGISTER

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
2007h	Config. bits	(1)	BODEN(1)	CP1	CP0	PWRTE(1)	WDTE	FOSC1	FOSC0
81h,181h	OPTION	$\overline{RBPU}$	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0

5.Write the interrupts Which wake up the peripheral interrupts from SLEEP.

The following peripheral interrupts can wake the device from SLEEP:

1. TMR1 interrupt. Timer1 must be operating as an asynchronous counter.
2. SSP (Start/Stop) bit detect interrupt.
3. SSP transmit or receive in slave mode (SPI/I2C).
4. CCP capture mode interrupt.
5. Parallel Slave Port read or write.
6. USART TX or RX (synchronous slave mode).

6.What is TIMER 0 ?

- The Timer0 module is a simple 8-bit overflow counter.
- The clock source can be either the internal system clock or an external clock
- When the clock source is an external clock, the Timer0 module can be selected to increment on either the rising or falling edge.
- The Timer0 module also has a programmable prescaler option.
- This prescaler can be assigned to either the Timer0 module or the Watchdog Timer.
- Bit PSA (OPTION) assigns the prescaler, and bits PS2:PS0 (OPTION) determine the prescaler value.
- TMR0 can increment at the following rates: 1:1
- when the prescaler is assigned to Watchdog Timer, 1:2, 1:4, 1:8, 1:16, 1:32, 1:64, 1:128, and 1:256. Synchronization of the external clock occurs after the prescaler.
- When the prescaler is used, the external clock frequency may be higher then the device‘s frequency.
- The maximum frequency is 50 MHz, given the high and low time requirements of the clock.

7. What is the TIMER 1?

- Timer1 can operate in one of two modes:
  - As a timer
  - As a counter.
- The clock source can be either the internal system clock (Fosc/4), an external clock, or an external crystal.
- Timer1 can operate as either a timer or a counter.
- When operating as a counter , the counter can either operate synchronized to the device or asynchronously to the device.
- Asynchronous operation allows Timer1 to operate during sleep, which is useful for applications that require a real-time clock as well as the power savings of SLEEP mode.
- TImer1 also has a presale option which allows TMR1 to increment at the following rates: 1:1, 1:2, 1:4, and 1:8. TMR1 can be used in conjunction with the Capture/ Compare/PWM module.
- When used with a CCP module, Timer1 is the time-base for 16-bit capture or 16-bit compare and must be synchronized to the device.

9. What is the RBIF register?

RBIF-Register B interrupt flag

Port B will set the RBIF bit in the INTCON register. If the interrupts have been enabled by the setting of INTCON‘s RBIE and GIE bit then the CPU will be interrupted.

10. What are the two steps for two-step process?

1. Read PORT B to copy the upper four bits of PORTB into the hardware copy thereby removing the mismatch condition.
2. bcf INTCON ,RBIF

11. What are the two parameters for display ?

The conversion of number entered digit by digit from a keybad into its binary equivalen
The display of fixed and variable strings of characters

12. What is the sequence for KEYSTATE?

The following steps are used for KEYSTATE

- Debounce the key switch
- Determine which key is pressed
- Take appropriate action once for the press of the key
- Wait for the release of that key

13. Write the display string format?

- Cursor-positioning code
- ASCII string of characters to be displayed
- End-of-string designator.

14. What is the RPG

For the instrument design the display plus either key switches or a rotary pulse generator (RPG) in their design of the front panel.
The dispay serves to display measurement
The display combines key switches ot RPG for the entry of setup parameters

15. What is the use of key switch

The key switches are not changed very fast, they can be checked once each time around the mainline loop in a Key switch subroutine. Recall that a looptime of 10ms was selected because the maximum key bounce time of the most mechanical key switches is less than 10 ms

16. How the constant string uses?

The labels associated with softkeys represent one application
The units associated with a variable represent another application.

17. What is CCP

Capture/compare/PWM module
Pic chip hacing two CCP modules if both modules are being used for either a compare function or capture function, they will share TMR1. In this case TMR1 should never a changed by writing to it.

18. Write the structure of OPTIO-REG

If key switch detects that a key is newly pressed, it can be assured that the next time it is called, 10ms later, any erratic bouncing of the key contacts will have settled out, with the contact firmly closed.

19. Write the calculating Count,Fount, and TMR0 values

$$fout = \frac{fclk}{4*Prescaler*(256-TMR0)*Count} = 2Hz(the\ needed\ frequency) \rightarrow \left(T = \frac{1}{2Hz} = 0.5\sec\right)$$

$$Count = \frac{fclk}{4*Prescaler*(256-TMR0)*fout} = \frac{4MHz}{4*256*(256-0)*2Hz} = 7.6294 \cong 8$$

$$\boxed{Count = 8\ (this\ is\ the\ value\ need)}$$

Formula to calculate Cout using Timer0

20. What is the function of capture mode?

The Basic function of capture mode by CCP1 or CCP2 module is that the exact point of time of occurrence of that input edge change can be detected. For this purpose, timer1 must be running in timer mode

21. What are the three interrupt of PIC

16c6x? External interrupt INT at pinRB0

Timer interrupt

Port-B

PART-B

1.Discuss the interrupt structure in PIC microcontrollers. List out the various interrupt sources inPIC 16c7x

2.Explain the timer of the PIC microcontroller

3.Explain the prescalar timer0.

4.Explain the Watchdog timer of PIC microcontroller.

5.Explain RB0 /INT external interrupt input

6.Explain the Compare mode

7.Explain the Capture mode

8.xplain the Timer1 CCPperiodscaler.

9.Explain the PortB change interrupts(RB7:RB4)

10. Write about Display of variable strings

11. Write a program for display of constant strings

UNIT-III
PERIPHERALS AND INTERFACING OF PIC MICROCONTROLLER

I2C Bus for Peripherals Chip Access– Bus operation-Bus subroutines– Serial EEPROM—Analog to Digital Converter–UART-Baud rate selection–Data handling circuit–Initialization – LCD and keyboard Interfacing -ADC, DAC, and Sensor Interfacing.

PART-A

1. **What are the different capture modes available in the capture module of PIC microcontroller? (June'12)**
   Three modules are there.
   (i) CCP1
   (ii) CCP2
   (iii) CCP3

2. **Write the function of the bits EEPGD and WRERR bits in the EECONI register in PIC Microcontroller. (June'12)**
   EEPGD-Flash Program or Data EEPROM Memory select bit
   1=Access Program flash memory; 0= Access data EEPROM memory

   WRERR – Write Error Flag bit
   1=Write operation is prematurely terminate; 0=the write operation is completed.

3. **Calculate the resolution of 10bit ADC having Max. analog value +10.0 volts. (June / July 2013)**
   Resolution of a converter determines the degree of accuracy in conversion. It is equal to $1/2^n$so, $1/2^{10}$=0.000976

4. **What are the interrupts available in PIC? (Jan'14)**

Interrupt Source	Enabled by	Completion Status
External interrupt from INT	INTE = 1	INTF = 1
TMR0 interrupt	T0IE = 1	T0IF = 1
RB4–RB7 state change	RBIE = 1	RBIF = 1
EEPROM write complete	EEIE = 1	–

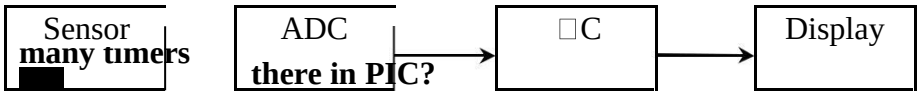
5. **How EEPROM memory stores the information? (Jan'15, Jan'14)**
   A EEPROM is a memory that allows storing the variables, as a result of burning the written program.

6. **What is flash memory? (Jan'15/Jan'13)**
   Erasure of the entire contents takes less than a second or one might say in a flash, hence its name flash memory. Flash memory's contents are erased (or written to), the entire device is erased.

7. **Using PIC micro controller how is analog signal converted into digital. (Jan'13)**

   ADC to translate the analog signals to digital numbers. So, that the microcontroller can read

and process them.



8. How many timers there in PIC?

PIC may have 3 to 5 timers depending on the series of the PIC. Timer0,1 and 2 etc.,.

9. Brief timer0.

It is an 8 bit wide timer. Internal clock is fosc/4 and external clock is given at RA4 pin.It is used for timing and generating time delay.

10. How do you calculate the timer0 delay?

The delay for Timer0 is given by
Timer0count x prescaler value x 4/fosc.

11. How do you calculate timer0 preload count?

The preload count for Timer0 is given by
256-(timer0delay x fosc)/ (prescalervalue x 4)

12. What are Port A pins?

The Port pins of PIC are RA0,RA1,RA2,RA3 and RA4.

13. What is the function of TRISA pin?

Setting TRISA bit will configure portA as input and resetting will configure as output port.

14. Write a program to initialize portA.

```
Org0
BcfSTATUS.RP0
clrf PORTA
bsf STATUS.RP0
movlw 00010000H
movwf TRISA. End
```

15. What is the status of ADON?

When ADON=0 then AD is off, when ADON=1 then AD is turned ON..

16. What are the bit postions of ADCON?

D0-ADON,D1-ADIF,D2-Go/Done,D3-CHSO,D4-CHS1,D5-undefined,D6-ADSC0,D7-ADSC1.

17. Explain about UART?

Universal asynchronous receiver transmitter. UART is useful for receiving and transmission of datas in asynchoronous mode.

18. What is synchronous and asynchronous transmission.

Asynchronous – start and stop bit allowed for transmission of data.
Synchronous – no start and stop bit only block header data

19. What is baud rate in asynchronous mode?

The baud rate in asynchoronous mode is given by B.R = Fosc/64.(x+1) for low speed,and Fosc/16(x+1) for high speed.

20. How do you configure the ports as input and output?

Any ports can be made as input by setting the port bits and they can be set as output by resetting the port bits.

21. How USART can be configured?

The Universal Synchronous Asynchronous Receiver Transmitter (USART) module is one of the two serial I/O modules. USART is also known as a Serial Communications Interface or SCI). The USART can be configured as

I. Asynchronous (full duplex)
II. Synchronous - Master (half duplex)
III. Synchronous - Slave (half duplex)

22. What do you mean by I^2C Bus?

At the low end of the spectrum of communication options for "inside the box" communication is I^2C The name I^2C is shorthand for a standard Inter-IC (integrated circuit) bus

I^2C provides good support for communication with various slow, on-board peripheral devices that are accessed intermittently, while being extremely modest in its hardware resource needs. It is a simple, low-bandwidth, short-distance protocol. Most available I^2C devices operate at speeds up to 400Kbps

23. What do you mean by Baud Rate Generator (BRG)?

The BRG supports both the Asynchronous and Synchronous modes of the USART. It is a dedicated 8-bit baud rate generator. The SPBRG register controls the period of a free running 8-bit timer the desired baud rate and Fosc, the nearest integer value for the PBRG register can be calculated using the formula

SYNC	BRGH = 0 (Low Speed)	BRGH = 1 (High Speed)
0	(Asynchronous) Baud Rate = FOSC /(64(X+1))	Baud Rate= FOSC/(16(X+1))
1	(Synchronous) Baud Rate = FOSC/(4(X+1))	NA

24. **What are the nodes of I^2C bus**

The bus has two roles for nodes: master and 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 any number of master nodes can be present. Additionally, master and slave roles may be changed between messages

25. Give four potential modes of operation of I^2C bus

- master transmit — master node is sending data to a slave
- master receive — master node is receiving data from a slave
- slave transmit — slave node is sending data to the master
- slave receive — slave node is receiving data from the master

26. Give the message protocols of I^2C bus?

I²C defines basic types of messages, each of which begins with a START and ends with a STOP:

- Single message where a master writes data to a slave;

- Single message where a master reads data from a slave;
- Combined messages, where a master issues at least two reads and/or writes to one or more slaves.

PART – B

1. Discuss the flash type ADC and its applications. Also explain how to interface it with PIC.

2. Discuss briefly how serial interfacing is accomplished in PIC. (Jan‘15)

3. Discuss in detail about the following (i) Timers and (ii) Interrupt (Jan‘13)

4. (i).Discuss in detail of I^2C in PIC microcontroller. (Jan‘13)
(ii). Briefly explain about UART in PIC microcontroller.(Jan‘13)

5. Explain the I/O ports in PIC microcontroller.

6. Briefly explain about flash memory in PIC with necessary diagram.

7. Discus briefly about LCD & key board interfacing in PIC?

UNIT IV
INTRODUCTION TO ARM PROCESSOR

ARM Architecture –ARM programmer‘s model –ARM Development tools- Memory Hierarchy –ARM Assembly Language Programming–Simple Examples–Architectural Support for Operating systems.

PART – A

1. **What is ARM Processor?**

   An ARM processor is one of a family of CPUs based on the RISC (reduced instruction set computer) architecture developed by ***Advanced RISC Machines (ARM).***ARM makes 32-bit and 64-bit RISC multi-core processors. RISC processors are designed to perform a smaller number of types of computer instructions so that they can operate at a higher speed, performing more millions of instructions per second (MIPS).

2. **What are the features of ARM processor?**
   - Load/store architecture.
   - An orthogonal instruction set.
   - Mostly single-cycle execution.
   - Enhanced power-saving design.
   - 64 and 32-bit execution states for scalable high performance.
   - Hardware virtualization support.

3. **What are the applications of ARM processor?**

   ARM processors are extensively used in consumer electronic devices such as smartphones,tablets, multimedia players and other mobile devices, such as wearables. Because of their reduced instruction set, they require fewer transistors, which enables a smaller die size for the integrated circuitry (IC). The ARM processor‘s smaller size, reduced complexity and lower power consumption makes them suitable for increasingly miniaturized devices.

4. **What is meant by Pipelining?**

   To improve the utilization of the hardware resources, and also the processor throughput, would be to start the next instruction before the current one has finished. This technique is called pipelining.

5. **What are the performance of Reduced Instruction Set Computer (RISC)?**
   - Pipelining – pipelining is the simplest form of concurrency to implement in a processor and delivers around two to three times speed up.
   - A high clock rate with single cycle execution – 3MHz for random accesses and 6Mhz for sequential accesses.

6. **What are the drawbacks of RISC processor?**
   - RISCs generally have poor code density (means consequence of fixed – length instruction set)compared with CISCs
   - RISCs don‘t execute x86 code
   - It is hard to fix , though PC emulation software is available for many RISC platforms.

7. **What is a software development tool of ARM processor?**

The ARM is supported by a toolkit which includes an instruction set emulator for hardware modelling and software testing and benchmarking, an assembler, C and C++compilers, a linker and a symbolic debugger.

8. What is the example of RISC architecture?

- Berkeley RISC I and II
- standford MIPS.

9. What are the feature were rejected by the ARM designers?

- Register windows –Procedure entry and exit instruction moved the visible _ window _ to give each procedure access to new registers.
- Delayed branches – branches cause pipeline problems since they interrupt the smooth flow of instruction. The problem with delayed branches is that they remove the atomicity of individuals instructions
- Single cycle execution of all instructions – ARM executes most data processing instructions in a single clock cycle, many other instructions take multiple clock cycles.

10. What are the registers of ARM processor?

User –level programs: The 15 general purpose 32-bit registers(r0 to r14), the Program Counter (r15) and the Current Program Status Register(CPSR) and the remaining registers are used only for System level programs.

11. What is the use of CPSR?

The CPSR is used in user level programs to store the condition code bits.

12. What are the condition code flags?

- N : Negative; the last ALU operation which changed the flags produced a negative result
- Z: Zero; the last ALU operation which changed the flags produced a zero result
- C: Carry; the last ALU operation which changed the flags generated a carry bout either as a result of an arithmetic operation in the ALU or from the shifter
- V: oVerflow; the last arithmetic ALU operation which changed the flags generated an overflow into the sign bit.

13. What are the three categories of ARM instruction?

- Data processing instruction
- Data transfer instruction
- Control flow instruction

14. What are the features of the ARM instructions?

- The load – store architecture
- 3-address data processing instruction
- Conditional execution of every instruction
- Very powerful load and store multiple register instructions
- Ability to perform a general shift operation and a general ALU operation in a single instruction.
- Open instruction set extension through the coprocessor instruction set
- A very dense 16-bit compressed representation of the instruction set.

15. What is the function of assembler?

The ARM assembler is a full assembler which produces ARM object format output that can be linked with output from the compiler.

16. What is the function of linker?

The linker takes one or more object files and combines them into and executable program. It resolves symbolic references between the object files and extracts object modules from libraries as needed by the program.

17. What is meant by ARMsd?

The ARM symbolic debugger is a front end interface to assist in debugging programs running either under emulation (on the ARMulator) or remotely on a target system such as the ARM development board.

18. What is meant by ARMulator? And list the various levels of accuracy.

The ARMulator (ARM emulator) is a suite of programs that models the behavior of various ARM processor core in software on a host system.

Various levels of accuracy:

- Instruction accurate
- Cycle accurate
- Timing accurate

19. List the various instruction set of ARM processor?

a) Arithmetic operations
b) Bitwise logical operations
c) Register movement operations (Data transfer)
d) Comparison operations
e) Shift register operations

20. What are the four formats of stack?

- Full ascending
- Empty ascending
- Full descending
- Empty descending

21. What are the levels of memory hierarchy?

- The RISC processor will typically have around thirty –two 32 bit registers making a total of 128 bytes.
- On chip cache memory will have a capacity of eight to 32 kbytes.
- High performance desktop system may have a second level off chip cache with a capacity of a few hundred Kbytes.
- Main memory will be Megabytes to tens of megabytes.
- Back up store, usually on a hard disk, will be hundred of Mbytes up to a Gbytes

22. What are the advantages of On-chip RAM?

It enables the programmer to allocate space in it using knowledge of the future processing load.

23. State the processor organization of the cache memory.

- A unifined cache

- Separate instruction cache and data caches.

24. Define cahce miss and cache hit.

An access to an memory item which is not in the cache memory is called ***cache miss***. An access to an memory item which is in the cache memory is called ***cache hit.*** The proportion of all the memory accesses that are satisfied by the cache is the ***cache hit rate*** and the proportions that are not is the ***miss rate.***

25. What are the type of cache memory organization?

- Direct mapped
- Two way - Set Associative
- Fully associative

26. What are the principals of memory management?

- Segmentation - it allows a program to have its own private view of memory and to co exist transparently with other programs in the same memory space
- Paging – Most processor incorporate a memory mapping scheme based on fixed – size chunks of memory called pages.

27. Give th**e CPSR format for ARM**

- N: Negative; the last ALU operation which changed the flags produced a negativeresult (the top bit of the 32-bit result was a one).
- Z: Zero; the last ALU operation which changed the flags produced a zero result(every bit of the 32-bit result was zero).
- C: Carry; the last ALU operation which changed the flags generated a carry-out,either as a result of an arithmetic operation in the ALU or from the shifter.
- V: overflow; the last arithmetic ALU operation which changed the flags generatedan overflow into the sign bit

28. Explain **about the memory organisation of ARM.**

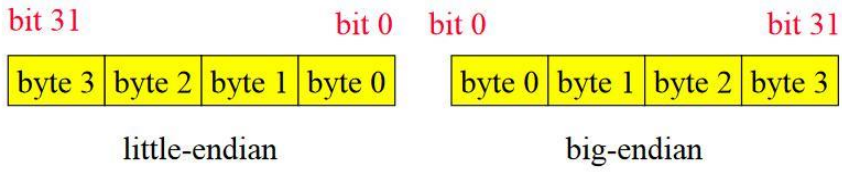
Memory may be viewed as a linear array of bytes numbered from zero up to 232-l. Data items may be 8-bit bytes, 16-bit half-words or 32-bit words. Words are always aligned on 4-byte boundaries (that is, the two least significant address bits are zero)and half-words are aligned on even byte boundaries.

29. List out the ARM Data Type

ARM processor support the following data types o Byte: 8 bits
o Halfword: 16 bits
Halfwordsmust be aligned to two-byte boundaries o Word: 32 bits
Words must be aligned to four byte bounries.

30. Write the Byte Organization of ARM

word Little-endian mode
The lowest order byte residing in the low-order bits of the word Big endian mode
The lowest –order byte stored in the highest bits of the word



31. What are the ARM Processor Modes.

User mode: a normal program execution state
FIQ (Fast Interrupt Request) mode :for supporting a specific data transfer or channel processing
IRQ (Interrupt ReQuest) mode:for general purpose interrupt handling
Supervisor mode:a protected mode for the operating system
Abort mode:entered when a data or instruction pre-fetch is aborted
System mode:a privileged user mode for the operating system, for example, implement reentrant interrupt handler
Undefined mode:entered when an undefined instruction is executed.
Operating mode changes :Can be controlled by software.
Can also be caused by external interrupts or exception Processing
Exception mode
System mode

32. Write the ARM Operating States?

ARM state :Executing 32-bit, word-aligned, ARM instructions
THUMB state :Executing 16-bit, half-word aligned THUMB instructions These states can be switched by software or by exception processing

33. What do you meant by exception?

Exceptions are usually used to handle unexpected events which arise during the execution of aprogram, such as interrupts or memory faults

34. What are the types of exception?

Exceptions generated as the direct effect of executing an instruction.Exceptions generated as a side-effect of an instruction.Exceptions generated externally, unrelated to the instruction flow.

35. Explain the priorities for exception.

Since multiple exceptions can arise at the same time it is necessary to define a priorityorder to determine the order in which the exceptions are handled. On ARM this is:1. Reset (highest priority);2. Data abort;3. FIQ;4. IRQ;5. Prefect abort;6. SWI, undefined instruction (including absent coprocessor). These are mutually exclusiveinstruction encodings.

36. What is Banked and Un banked registers? Unbankedregisters (R0-R7)
Each of the registers refers to the same 32-bit physical registers in all processor modes Bankedregisters (R8-R14)

37. List the Register organization in ARM state.

The physical register referred to by each of them depends on the current processor modeIf a particular register is intended without depending on the current processor mode
For example, R8_usr v.s. R8_fiq

38. What is Program Status word in ARM processor The format of CPSR and SPSR are the same

Called PSR in general

One CPSR (Current Program Status Register)

Copies of the Arithmetic Logic Unit(ALU) status flags The current processor mode
Interrupt disable flags

Five SPSR (Saved Program Status Registers)

Used to store the CPSU when an exception is taken
One SPSU is accessible in each of the exception-handling mode
User mode and System modedo not have an SPSR because they are not exception handling modes

PART – B

1. Explain in detail about ARM architecture
2. Explain in detail about ARM development tools
3. Explain in detail about Instruction set of ARM processor
4. Write ARM assembly language program to display _ Hello World‘
5. Write ARM assembly language program to load and store to copy the table.
6. Explain in detail about ARM MMU architecture.
7. Explain in detail about ARM protection unit Registers and Protection unit.
8. Explain the Register organization in ARM state.
9. Explain the ARM programmer‘s module in ARM.
10. Explain in detail the ARM processor Modes.

UNIT V
ARM ORGANIZATION

3-Stage Pipeline ARM Organization– 5-Stage Pipeline ARM Organization–ARM Instruction Execution- ARM Implementation– ARM Instruction Set– ARM coprocessor interface– Architectural support for High Level Languages – Embedded ARM Applications.

PART - A

1. **What are the principal components of 3 stage pipelining?**
   - The register bank, which stores the processor states
   - The barrel shifter, which can shift or rotate one operand by any number of bits
   - The ALU, which performs the arithmetic and logic functions required by the instruction set.
   - The address register and incrementer, which select and hold all memory address.
   - The data registers, which hold data passing to and from memory.
   - The instruction decoder and associated control logic.

2. **What are the pipe lines stages in three stage pipe lining?**
   - Fetch:The instruction is fetched from memory and placed in the instruction pipeline.
   - Decode:The instruction is decoded and the datapath control signals prepared for the nextCycle. In this stage the instruction 'owns' the decode logic but not the datapath.
   - Execute:The instruction 'owns' the datapath; the register bank is read, an operand shifted,the ALU result generated and written back into a destination register.

3. **What are the ways to view breaks in the ARM pipeline?**
   - All instructions occupy the data path for one or more adjacent cycles.
   - For each cycle that an instruction occupies the data path.
   - During the first data path cycle each instruction issues a fetch for the next instruction.
   - Branch instructions flush and refill the instruction pipeline.

4. **What are the pipeline stages in 5 stage pipeline?**
   - Fetch
   - Decode
   - Execute
   - Buffer/data

- Write-back

5. What are the ways to improve the performance of 5 stage pipeline?

- Increase the clock rate, fclk.
- Reduce the average number of clock cycles per instruction,CPI.

6. What are the minimum data path cycle time?

- The register read time
- Shifter delay
- ALU delay
- Register write set-up time
- The phase 2 to phase 1 non over lap time.

7. What are the 3 structural components of ARM codes?

- Instruction decoder PLA
- Distributed secondary control
- Decentralized control unit

8. What is mean by ARM co-processor interface?

It is based on BUS watching (other ARM cores use different techniques).As each co-processor instruction begins execution there is a handshake between the ARM and the co-processor to confirm that they are both ready to execute it.

9. What are the handshake signals used by the ARM co-processor interface?

- Cpi (from ARM to all co-processor)
- Cpa (from the co-processor to ARM)
- Cpb (from the co-processor to ARM)

10. What is the use of breakpoint instruction (BKPT)?

It is used for software debugging purposes; they cause the processor to break from normal instruction execution and enter appropriate debugging procedures.

11. What is the use of Software Interrupt Instruction?

The Software Interrupt Instruction (SWI) is used for calls to the operating system and is often called a ‗supervisory call'. It puts the processor in supervisory mode and begins executing instructions from address 0x08.

12. How ARM architecture will support for high level language?

A High level language allows the programmer to think in terms of abstraction that are above the machine level; indeed, the programmer may not even know on which machine the program will ultimately run. Parameter such as number of registers vary from architecture to architecture, so clearly these must not be reflected in the design of the language.

13. Write, compile and run a ' Hello World' program written in C.

The following program has the required

```
function /* Hello World in C */
# include <stio.h>
```

```
Int main ()
{
Printf ( — Hello World\n‖) ;
return
(   0   ); }
```

14. Write the number 2001 in 32 bit binary, binary-coded decimal, ASCII and single-precision floating point notation.

Binary : 2001 = 1024 + 512 + 256 + 128 + 64 + 16 + 1
= $00000000000000000000011111010001_2$

BCD : 2001 = 0010 0000 0000 0001

ASCII : 2001 = 00110010 00110000 00110000 00110001

F-P: 2001 = $1.1111010001 * 2^{1010}$
= 01001001 11110100 01000000 00000000

15. List the various power management modes in chip.

- On-line – all circuits are clocked at full speed
- Command – the ARM core runs with 1 to 64 wait states but all other circuitry runs full speed. An interrupt switches the system into on line mode.
- Sleep – all circuitry is stopped apart from the timers and oscillators. Particular interrupts return the system to online mode.
- Stopped – all circuits (including the oscillators) are stopped. Particular interrupts return the system to online mode.

16. What are the Embedded ARM applications?

- The VLSI Ruby II Advanced Communication Processor
- The VLSI ISDN Subscriber Processor
- The OneC VWS22100 GSM chip
- The Ericsson –VLSI Bluetooth Baseband Controller
- The ARM7500 and ARM7500FE
- The SA - 1100

17. When padding will be used in C compiler?

Where several data items of different types are declared at the same time, the compiler will introduce padding where necessary to achieve this alignment:

This structure will occupy three words of memory as shown in Struct SI { char c; int x; short s;} example;

1. What the features are of ARM 7processor?

- The ARM7, a 3 volt compatible rework of the ARM6 32-bit integer core, with the Thumb 16-bitcompressed instruction set;
- On-chip Debug support, enabling the processor to halt in response to a debug request;• An enhanced Multiplier, with higher performance than its predecessors and yielding a full 64-bitresult;

- Embeddable hardware to give on-chip breakpoint and watch point support.

2. **What are the additional features provided by ARM 9 TDMI?**
   - Hardware single-stepping is supported.
   - Breakpoints can be set on exceptions in addition to the address/data/control conditions supportedby ARM7TDMI

3. **How the ARM 9 TDMI supports the co-processor support?**
   The ARM9TDMI has a coprocessor interface which allows on-chip coprocessors for floating-point,digital signal processing or other special-purpose hardware acceleration requirements to besupported.

4. **What are the advantages of ARM instruction set?**
   - All instructions are 32 bits long.
   - Most instructions are executed in one single cycle.
   - Every instruction can be conditionally executed.

5. **What are the advantages of Thumb instruction set?**
   - All instructions are exactly 16 bits long to improve code density over other 32-bitarchitecture.
   - Long branch range
   - Powerful arithmetic operations
   - Large address space

6. **List the Arithmetic operations?**
   - ADD
   - ADC
   - SUB
   - SBC
   - RSB
   - RSC

7. **What is the use of barrel shifter in ARM?**

The ARM does not have actual shift instruction instead it has a barrel shifter which provides a mechanism to carry out shifts as a part of other instructions.

8. **Why ARM Architecture is a load or store Architecture?**

Because it does not support memory to memory data processing operations.

Also it must move the data values into registers before using them

9. **Mention three sets of instructions which interact with main memory?**

   Single register data transfer(LDR/SDR)2. Block data transfer (LDM/STM).3. Single Data Swap(SWP).

10. Wh**at are the typ**es of l**oad and** store in**struc**tions .
    Load or store the value of a single register*Load and store multiple Load and Store multiple instructions perform a block transfer of anynumber of the general purpose registers to or from memory Location.

11. Define ARM 7TDMI processor.

The ARM7TDMI processor is a member of the Advanced RISC machine family of general purpose32-bit microprocessor.

12. What are the two operating states in ARM7TDMI?

ARM state which executes 32-bit, word aligned ARM instructions–THUMB state which can execute 16-bit, half word aligned THUMB instructions

13. List the seven modes of operation of ARM7TDMI.

User (USR)–
FIQ (FIQ)–
IRQ (IRQ)–
Supervisor (SVC)–
Abort mode (ABT)–
System (SYS)–
Undefined (UND)

14. How many registers are available in ARM7TDMI? There are totally 4.There are totally 37 registers.

•31 general-purpose 32-bit registers
•6 status registers•

15. What is a load instruction?

A load instruction provides a way to branch anywhere in the 4Gbyte addressspace. A 32-bit value is loaded directly from memory into the PC, causing a branch.

16. What are the two classes of multiply instruction inARM?

normal, 32-bit result
long, 64-bit result

17. What are the types of load and store instructions

. *Load or store the value of a single register
*Load and store multiple Load and Store multiple instructions perform a block transfer of anynumber of the general purpose registers to or from memory Location.

18. What are the addressing modes provided in ARM7TDM1?

- pre-increment
- post-increment
- pre-decrement
- post-decrement

19. **Define thumb instruction set?**

The Thumb instruction set is a subset of the ARM instruction set, optimized for codedensity.

20. What are the data types accepted by ARM processor?

ARM processors support six data types:• 8-bit signed and unsigned bytes.• 16-bit signed and unsigned half-words; these are aligned on 2-byte boundaries.• 32-bit signed and unsigned words; these are aligned on 4-byte boundaries.
and therefore cannot occur simultaneously.

21. Explain about the conditional execution.

An unusual feature of the ARM instruction set is that every instruction (with the exception of certain v5T instructions) is conditionally executed. Conditional branches are a standard feature of most instruction sets, but

ARM extends the conditional execution to all of its instructions, includingsupervisor calls and coprocessor instructions. The condition field occupies the top four bits of the32-bit instruction field. Explain about the branch instruction.
Branch and Branch with Link instructions are the standard way to cause a switch in the sequence of instruction execution. The ARM normally executes instructions from sequential word addresses inmemory, using conditional execution to skip over individual instructions where required.

22. Explain about the Branch, Branch with Link and exchange.
These instructions are available on ARM chips which support the Thumb (16-bit) instruction set,and are a mechanism for switching the processor to execute Thumb instructions or for returningsymmetrically to ARM and Thumb calling routines.

23. Explain about software interrupt.
The software interrupt instruction is used for calls to the operating system and is often called a'supervisor call'. It puts the processor into supervisor mode and begins executing instructions fromaddress 0x08.

24. Explain the processor operations for the execution of software interrupt
If the condition is passed the instruction enters supervisor mode using the standard ARM exceptionentry sequence. In detail, the processor actions are:1. Save the address of the instruction after the SWI in r14 svc.2. Save the CPSR in SPSR svc.3. Enter supervisor mode and disable IRQs (but not FIQs) by setting CPSR [4:0] to100112 and CPSR [7tol].4. Set the PC to 08 and begin executing the instructions there.

25. What is the use of data processing instruction?
The ARM data processing instructions are used to modify data values in registers. The operationsthat are supported include arithmetic and bit-wise logical combinations of 32-bit data types. Oneoperand may be shifted or rotated en route

to the ALU, allowing, for example, shift and add in asingle instruction

26. Explain about single word and unsigned byte data transfer instruction.
These instructions are the most flexible way to transfer single bytes or words of data betweenARM's registers and memory. Transferring large blocks of data is usually better done using themultiple register transfer instructions, and recent ARM processors also support instructions fortransferring half-words and signed bytes.

27. Explain about half word and signed byte data transfer.
These instructions are not supported by some early ARM processors. As a result of their lateaddition to the architecture they are somewhat 'shoe-horned' into the instruction space as indicatedby the split immediate field.

28. Explain about multiple register transfer instruction.
The ARM multiple register transfer instructions allow any subset (or all) of the 16 registers visiblein the current operating mode to be loaded from or stored to memory. A form of the instruction alsoallows the operating system to load or store the user-mode registers to save or restore the userprocess state, and another form allows the CPSR to be restored from the SPSR as part of a returnfrom an exception handler.

29. Explain about the swap instruction.
Swap instructions combine a load and a store of a word or an unsigned byte in a single instruction.

30. What are the data types supported by ARM?

- Signed and unsigned characters of at least eight bits.
- Signed and unsigned short integers of at least 16 bits.
- Signed and unsigned integers of at least 16 bits.

- Signed and unsigned long integers of at least 32 bits.
- Floating-point, double and long double floating-point numbers.
- Enumerated types.
- Bit fields.

The ARM C compiler adopts the minimum sizes for each of these types except the standard integer.

31. Explain about the thumb instruction.

The Thumb instruction set can be incorporated into a 3-stage pipeline ARM processor macro cellwith relatively minor changes to most of the processor logic (the 5-stage pipeline implementationsare trickier). The biggest addition is the Thumb instruction decompressor in the instruction pipeline;this logic translates a Thumb instruction into its equivalent ARM instruction.

32. Give the properties of thumb instruction.

- The Thumb code requires 70% of the space of the ARM code.
- The Thumb code uses 40% more instructions than the ARM code.
- With 32-bit memory, the ARM code is 40% faster than the Thumb code.
- With 16-bit memory, the Thumb code is 45% faster than the ARM code.
- Thumb code uses 30% less external memory power than ARM code.

33. What is the application of thumb instruction.

- A high-end 32-bit ARM system may use Thumb code for certain non-critical routines to savepower or memory requirements.
- A low-end 16-bit system may have a small amount of on-chip 32-bit RAM for critical routinesrunning ARM code, but use off-chip Thumb code for all non-critical routines.

PART – B

1. Explain in detail about 3 stage pipelining ARM organization.
2. Explain in detail about 5 stage pipelining ARM organization.
3. Explain in detail about ARM instruction execution unit.
4. Explain in detail about Branch instruction Software Interrupt instruction(SWI)
5. Write short notes for the following coprocessor instruction, data operation, data transfer, register transfer operation.
6. Explain in detail about ARM floating point architecture .
7. Describe the hierarchical program structure.