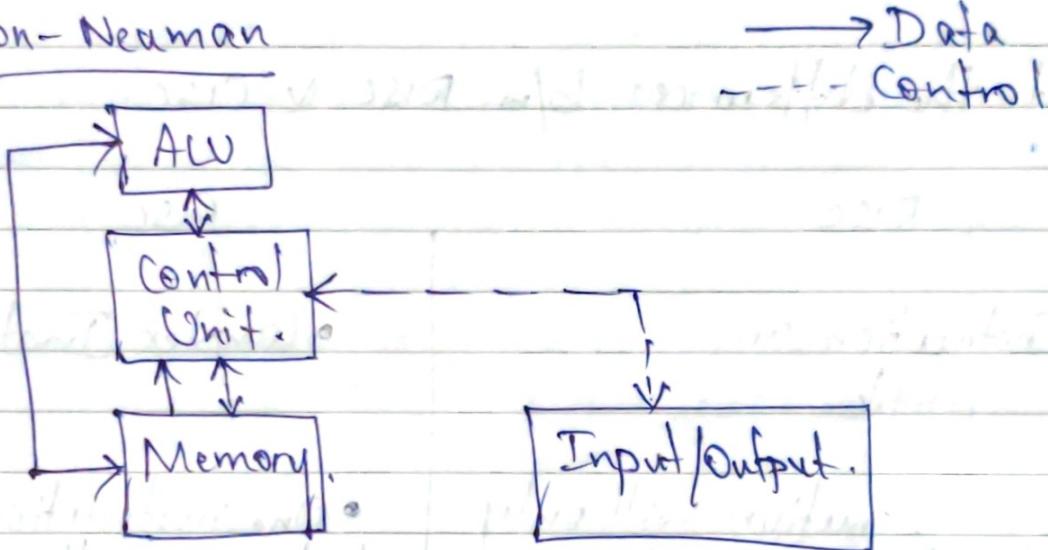


MCA

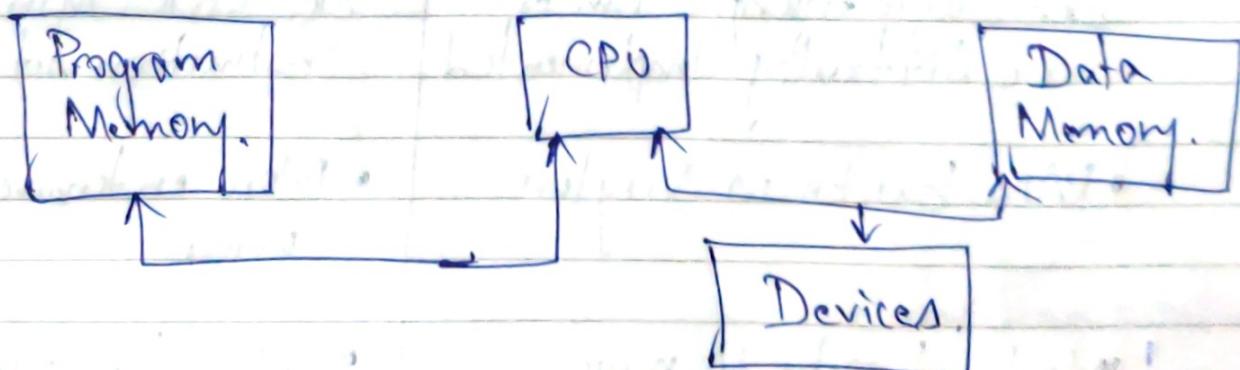
Assignment - 1

- ① With a neat diagram Explain Von-Neuman & Harvard Architecture.

Sol:

Von-Neuman

- Same memory space is shared b/w program & data in Von-Neuman
- Single bus is sufficient since data & programs cannot be fetched at a time
- Cost effective architecture.

Harvard Architecture.

- Separate memory space for code and data
- Additional bus is required.

- Cost is more due to separate bus for code and data
- free data in code [program is only specific to code / data menu respectively]

② list the differences b/w RISC & CISC.

Ans:-

RISC

- Instruction are simple.

- One instruction will only take place once clock pulse (1.5 CPI) AV.

- Instruction length is fixed.

- More burden on compiler

Ex: Same example has to be difficultly implemented

- RAM burden is very less

- Code density is more

CISC

- Complex Instruction

- One instruction will take more than one clock pulses to execute (2 to 15 CPI) AV.

- Variable length instruction

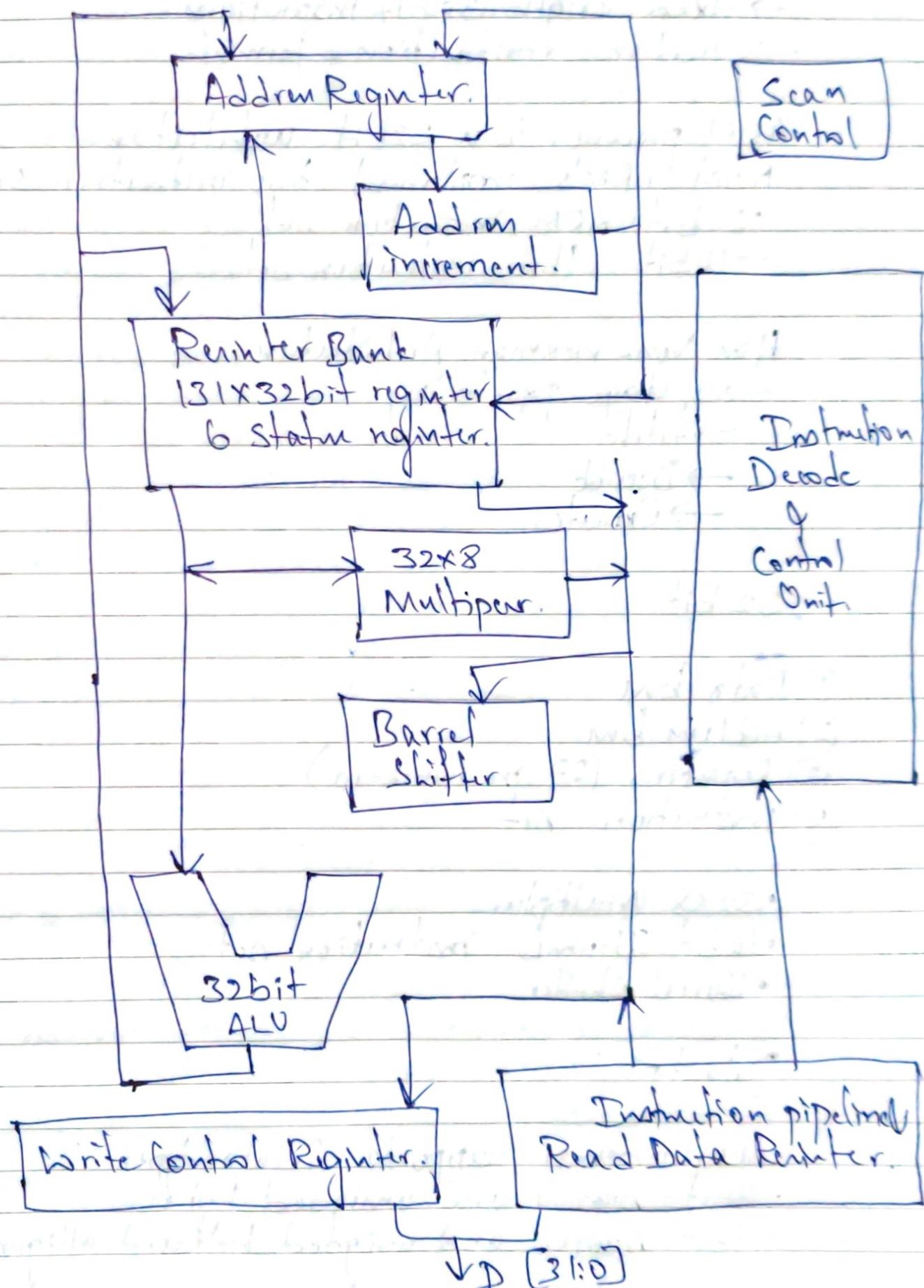
- Less burden on compiler.

Ex: \sqrt{M} instruction is only.

- More extensive RAM usage.

- Code density is less

③ With a neat diagram explain the ARM7 architecture



Features incl.

- Load/Store Architecture
- fixed length 32 bit instructions
- 3 address instruction format.

ARM processor is a 32bit architecture.

Most ARM's implement two instruction set

- 32bit ARM instruction set
- 16bit Thumb instruction set.

Has Von-neumann Architecture.

- 3 stage pipelining
- Fetch
- Decode
- Execute.

32 bit

- ① Data bus
- ② Address bus
- ③ Registers (37 in number)
- ④ Instruction Set

- 32x8 Multiplier.
- 16 bit Thumb instruction set
- Barrel Shifter.

Data type.

ARM processor supports 6 data types

- 8bit Signed and unsigned byte.
- 16bit Signed and unsigned halfword aligned

on 2 byte boundaries.

• 32bit signed and unsigned half words

ARM instruction are all 32bit words aligned
thumb instructions are half words aligned on 2byte
boundaries.

Internally all ARM operations are on 32bit
operands. the shorter data type are only supported
by data transfer instruction. When a byte is
loaded from memory it is zero or unsigned to 32bit

ARM co-processors support floating point values.

④ programming model for ARM7

• Each instruction can be viewed as performing a state
transformation of the state.

- Visible registers
- Invisible registers
- System memory
- User memory

Processor mode.

→ ARM has some basic operating modes

→ mode change by software control or external
interrupt

PSR (4:0) Mode User | Supervisor | Registers

0000	User	Normal user code	User
0001	FIQ	Pending fast interrupt	FIQ
0010	IRQ	Pending std interrupt	IRQ
0011	SVC	Pending software interrupt	SVC
0111	Abort	Handling memory fault	ABT
1011	Unde	Handling Undefined instruction	UND
0111	System	Running privileged OS.	User

- Most programs operate in user mode ARM has other privileged operating modes which are used to handle exceptions supervisor call & system mode

- More user right to memory systems & co-operation
- Current operating mode is defined by CPSR (4:0)

Privileged Modus

Supervisor modus

- Have some protective privileges
- Systems level functions can be accessed through specified supervisor call.
- usually implemented by software interrupt

ARM has 37 registers all of which are 32 bits

→ 1 program counter.

→ 1 current program status register.

→ 5 dedicated saved program status registers

- 30 general purpose registers.
- Each mode can access a particular set of R0-R12 registers.
- Stack pointer R13-link register.
- program counter r15 (PC)
- The current program status register CPSR.

Privileged modes can be access a particular SPSR (Saved program status register).

R0 to R15 + CPSR → User mode

F10	D10	SPNC	Undef	About
r8				
r9				
r10				
r11				
r12				
r13(SP)	r13(SP)	r13(rSP)	r13(SP)	r13(SP)
r14(lr)	r14(lsr)	r14(lr)	r14(lr)	r14(lr)
Spsr	Spsr	Spsr	Spsr	Spsr

⑥ With a neat diagram explain 3 stage pipeline of ARM

→ pipelining is the mechanism used by RISC & CISC processor to execute instructions.

→ By speeding up the execution by fetching the instruction while other instruction are being decoded & executed simultaneously.

→ pipelining is a design technique or process which plays an important role in increasing the efficiency of data processing in the processor of a computer & etc.

→ The ARM7 has three stage pipeline.

- Fetch: The instruction is fetched from the memory
- Decode: The instruction's opcode & quads are decoded to determine what function to perform
- Execute: The decoded instruction is executed.

Each of these operations requires one clock cycle for typical instructions. Thus a normal instruction requires three clock cycles to completely execute known as the latency of instruction execution.

because the pipeline has three stages of instruction execution completed in every clock cycle. In other words the pipeline has a throughput of one instruction per cycle.

	Fetch	Decode	Execute
Time cycle 1	ADD	ADD	
cycle 2	SWS	ADD	
cycle 3	ImmD	ADD	ADD

⑦ With the neat diagram. Explain CPSR Register.

N Z C V T undefined IF T mode.

Condition code flags

- N : Negative result from ALU
- Z : Zero result from ALU
- V : ALU operation overflowed
- C : ALU operation carried out.

Sticky Overflow flag - Q flag

- Architecture STE only
- indicates if saturation has occurred during certain operation

Interrupt Enable bit

- $\Rightarrow T=1$ Enables the IRQ
- $\Rightarrow T=0$ Disables the FIQ.

T bit

- Architecture XT only
- $T \Rightarrow 0$ processor in ARM state
- $T \Rightarrow 1$ processor in Thumb state

Mode bits

- used to specify the processor mode

CPSR \rightarrow current processor status register. Holds the information about the current state of the processor.

SPSR → Saved previous status register holds the information on the previous state before the system changed to this mode i.e. previous state just before an exception.

Q) Explain the seven different mode in ARM

The ARM & Thumb processor has seven modes of operation!

→ User mode is the usual ARM program execution state and is used for executing most application programs.

→ Fast interrupt (FIQ) mode supports a data transfer or channel program.

→ Interrupt (IRQ) mode is used for general purpose interrupt handling.

→ Supervisor mode is a protected mode for the operating systems.

→ Abort mode is entered after a data or instruction or prefetch Abort.

→ System mode is a privileged user mode for the operating system.

We can only enter system mode from another privileged mode by modifying the mode bit of the current program status register (PSR).

undefined mode M entered when an undefined instruction is executed

Mode other than user mode are collectively known as privileged mode. Privileged modes are used by device interrupt or exceptions or to access protected resources.

Mode

Mode Identifier

User

User

Fast interrupt

fiq

Interrupt

ird

Supervisor

soe

Abrt

abt

System

sys

undefined

und

Q) Explain the nomenclature in ARM

ARM was originally from Acorn Computer Ltd first RISC processor for commercial use.

ARM7TDMI processor.

32 bit processor Advanced machine

T → Thumb Architecture extension

D → Debug extension

M → Enhanced extension

I → Intrinsic Emulation

ARM {x} {y}. {z} TDMI {E} {J} {C} {S}

x → serial

y → Memory Management Unit

z → cache

T → Thumb 16 bit word.

D → JTAG Debugger

M → Fast Multiplier

I → Embedded ICE (In-Circuit Emulation)

E → Enhanced Instruction by Tazelle

F → floating point

S → synthesizable version

⑩ What is JTAG? Explain JTAG state Diagram

JTAG has become a standard in Embedded System and it is available in nearly every microcontroller and FPGA on the market

If we have programmed a microcontroller there is strong chance that we have used JTAG or one of the related standards.

JTAG i.e., Joint Test Action Group, is an industry standard for verifying designs and testing printed circuit boards after manufacture.

JTAG implement standard for onchip instruction in electronic design automation (EDA) as a complementary tool to digital simulation.

It specifies that we have a dedicated debug port implementing a serial communication interface for low-overhead access without requiring direct external access to the system address and control

buses. The interface connects to an on-chip fast access port (TAP) that implements a stateful protocol to access a set of fast registers.

fast logic reset

run fast idle

Select DR Scan

Capture DR

Shift DR

Exit DR

Pause DR

exit 2DR

Update DR

Select DRScan

Capture DR

Shift DR

Exit DR

Pause DR

exit 2DR

update DR

(ii) What is single Tasking? Give examples of popular applications.

Single tasking means doing one task at a time with no little distraction and interruption as possible. Microcontroller are known as computer on chip they are designed to perform a single task only because it is consuming power as well as memory is not suitable for installing and OS.

⑫ What is MMU? Why MMU is required? Give eg of MMU support.

The memory can be defined as a collection of data in a specified format. It is used to store instructions and processed data. The memory comprises a large array of or group of words or bytes, each with its own location. The primary motive of a computer system is to execute programs. These programs along with the information by array should be in the main memory during execution. The CPU fetches the instruction from memory according to the value of the program counter.

The main memory is central to the operation of a computer. Main memory is a large array of words or bytes ranging in size from hundreds to thousands to billions. Main memory is a repository of rapidly available information shared by the CPU and I/O devices.

Main memory is the place where programs & information are kept when the processor is effectively utilizing them. Main memory is associated with the processor. It is extremely fast.

Main memory is also known as RAM (Random Access Memory). This memory is a volatile memory. RAM loses its data when a power interruption occurs.

Memory Management.

In a multiprogramming computer the operating system resides in a part of memory and run n med by multiple process. The task of subdividing the memory among different process is called memory management. Memory management is a method in OS to manage/manage operations b/w main memory and Int during program execution. The main aim of memory management is to achieve efficient utilization of memory.

(B) Why Memory management is required.

- Allocate & de-allocate the memory before and after the program execution.
- To keep track of used memory space by program.
- To minimize fragmentation issues.
- To proper utilization of main memory.
- To maintain data integrity while executing program.

Eg:- IBM system/360 Model 67, IBM system/370

ARM

ARM architecture based application processor implement an MMU defined by ARM's virtual memory system architecture. The current architecture defines PTEs for describing 4KB and 8KB pages, 1MB sections and 16MB supersections legacy remap also defined in 1KB tiny page.

- ⑯ Write a C-program to find the endianess of given number

```
#include <stdio.h>
int main()
{
    unsigned int x = 0x76543210;
    char *c = (char*)&x;
    if (*c == 0x10)
    {
        printf("underlying architecture is little endian\n");
    }
    else
    {
        printf("underlying architecture is big endian\n");
    }
    return 0;
}
```

- ⑰ Explain following

① Bit - A bit is smallest unit of information that can be stored in a computer. Bits in computer are grouped to form a longer unit of information

② Byte - A byte is a combination of eight bits. Eight bits represent a character and is called a byte

③ Nibble - A nibble is a combination of four bits in other words a nibble is half a byte.

④ Word - a word is a combination of 16 bits, 32

bit or 64bit depending on the computer i.e., M known as quad word.

⑯ Explain the word align and Half word align in ARM memory.

Different processor have different definition of words for 32bit processor, a word M 32bit (4bytes) or the number implies, a halfword 16bit for a 16bit processor, a word M 16bit (2bytes), for 8bit processor word M 8bit

Word alignment: The stored address are adjacent and can be divided by 4, the last two digits are 00.

Half-word alignment: That in the stored address are adjacent and divisible by 2 that in the last bit M 0.

ARM architecture requires 32bit ARM instructions that must be word aligned and stored in memory and 16bit thumb instructions require Halfword aligned & stored. Therefore in ARM state the value of R15 is always divisible by 4, that in lowest 2bits of the R15 register are always divisible by 4, that in lowest 2bit of the R15 register are always 00.

In the thumb state, the value of R15 is always divisible by 2, when the lowest bit of the R15 register always 0. One word consist of one or more bytes i.e., usually integer bit of bytes.

classmate
Date _____

Q) Explain the following addressing modes in ARM

- (a) Three address
- (b) Two address
- (c) One address instruction using ARM

Ans:- Sequence of instructions form a program to perform a specific task

→ OP code field → specifies low level data manipulation

2 components →

→ Address field → specifies the data location.

When data to be read from or stored in two or more address field many have one or more than one address

- * Three address Instructions,
- * Two address instructions.
- * One address instruction.
- * Zero address instructions.

Program can execute an instruction only if it represented in binary sequence. Unique binary sequence pattern must be assigned this process is called op-code encoding

One address instruction:

This has an implied accumulator register for data manipulation and the other is in the registered memory location. Implied mean that

the CPU already knows that one operand is in the accumulator so there is no need to specify it

Eg: LDR addr

$Acc \leftarrow (addr)$

Two address instruction.

There has address can be specified in the instruction. In the one address instruction, the result was stored in the accumulator, then the result can be stored in different locations i.e., Register or memory location. But require more numbers of bit to represent the address.

Eg MOV R₁, R₂

$R_1 \leftarrow (R_2)$

Three address Instruction.

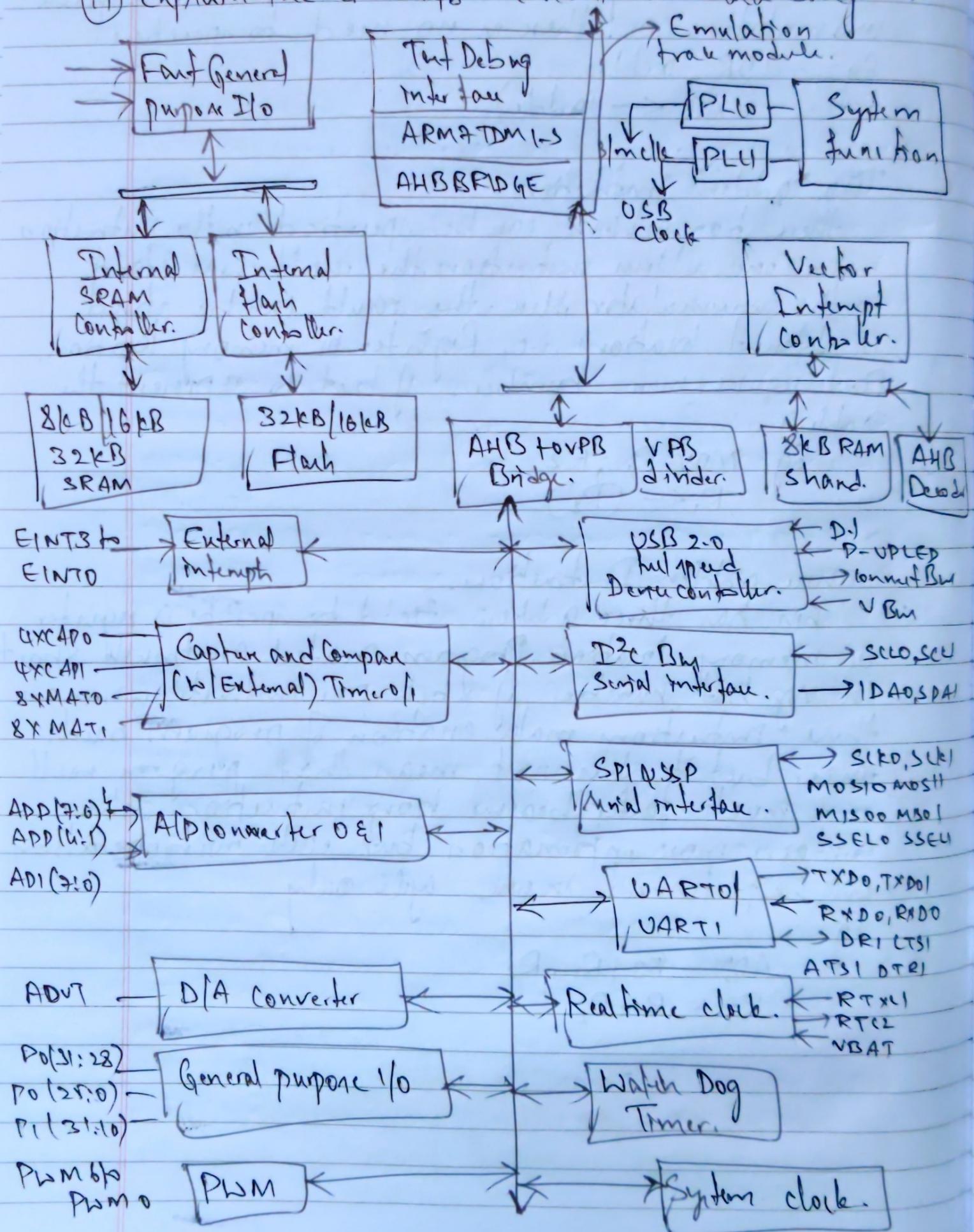
This has three address field to specify a register or memory location. Program created are much short in size but number of bit per instruction increases. These instructions make creation of programs much easier but it does not mean that program will run much faster because now instruction only contain more information but each micro-operation will be performed in one cycle only.

Eg:

ADD R₃; R₁, R₂

$$R_3 = R_1 + R_2$$

Q9 Explain the LPC 2148 Microcontroller Block diagram



- LPC 2148 features.
- 16 bit / 32 bit ARM7 TDM1-5 microcontroller.
- 8 kB to 40 kB of on chip static RAM
- 32 kB to 512 kB of on chip Flash Memory.
- 128 bit wide interface / accumulator enabled 60MHz high speed operations.
- In system programming / In Application programming via onchip boot loader software.
- USB 2.0 full speed device controller with 2 kB of end point RAM
- Single 10 bit DAC provides variable analog output
- Two 32 bit timers / external event counter, PWM unit and watchdog timer.
- low power Real time clock (RTC) with independent power of 32 kHz clock input
- up to 21 external interrupt pins available
- The onchip integrated oscillator operates with an external crystal from 1 MHz to 25 MHz
- Single power supply with POR, BOD circuits CPU operating voltage range of 3.0V to 3.6V

(20) Explain the LPC2148 microcontroller GPIO pins.

GPIO Pin → General purpose input/output.

A 32 bit register used to select the functions of pins in which the user needs it to operate there are four functions for each pin of the controller, where the first function is GPIO. It means that the pin can either act as an input or output with no specific functions.

There are totally three PIN register in LPC2148 controller in order to control the function of the pins in the respective port. the classification is given below

PIN SEL0 - Controls functions of Port 0.0 - Port 0.

PIN SEL1 - Controls functions of Port 0.16 - Port 0.31

PIN SEL2 - Controls functions of Port 1.16 - Port 0.31

When the 00 of 32 bit registers are GPIO configuration

LPC2148 has two 32bit general purpose I/O port A total of 30 I/O and a single output only pinout of 32Pin are available on PORT0. PORT1 has upto 16 pins available for GPIO functions PORT0 & PORT1 are connected to two groups of 4 registers.

* IOPIN

* IODSET

* IODDIR

* IODCLR

IOPIN

This register provides the value of port pins that are configured to perform only digital functions. The register will give the logic value of the pin, regard less of whether the pin is configured for I/P or output or as GPIO or an alternate digital function.

IOSET

This register is used to produce a HIGH level output at the port pin configured as GPIO in an OUTPUT mode. Writing 1 produces a HIGH level at the corresponding port pin or configured as an I/P or a secondary function, writing 1 to the corresponding bit in the IOSET has no effect. Reading the IOSET register returns the value of this register as determined by previous writes to IOSET register returns the IOSET register, as determined by previous writes to IOSET or DDIR. This value does not reflect the effect of any outside world influence on the I/O pins.

IDDIR

This word addressable register is used to control the direction of the pins when they are configured as GPIO port pins. Direction bit for any pin must be set according to the pin functionality.

DDIR

This register is used to produce a low-level output at port pins configured as GPIO in an Output mode. Writing 1 produces low level at the corresponding port pin & clear the corresponding bit in IOSET.

register. Writing 0 has no effect. If any pin configured as an input or a secondary function writing to TCCR has no effect.

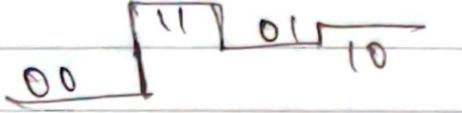
- Q2) With a neat diagram explain baud rate & bit rate. Explain the calculations.

Baud - How many times a signal change per second
 Bit - How many bits can be sent per time unit (usually per second)

Bit rate is controlled by baud and number of signal levels.

Bandwidth

- number of times lines changed per second
- Let Band rate be 4 (4 changes per second)
- If bit per line change be 2
- Bit rate = 8 bits per second
- Bit rate = $\times 2$ Band rate in this example



One second

- Band rate defines the switching speed of a signal
- Bit rate defines the rate of whole information flow across a data link measured in bits/second
- for a binary two level signal a data rate of one bit per second is equivalent to one baud

An analog signal carries 4 bit in each signal unit if 1000 signal units are sent per second, find the baud rate and the bit rate.

1 bit \rightarrow 1 symbol
bit rate = baud rate

$$\frac{1000}{\text{band/sec.}} = \text{Band rate}$$

$$\text{bit rate} = 1000 \times 4 = 4000 \text{ bps}$$

if bit rate (or data rate) is "b"
baud rate (or symbol) is "s"

General formula

$$b = s \times n$$

$\Rightarrow b$ = Data Rate (bit per second)

$\rightarrow s$ = Symbol rate (symbol/sec.)

$\rightarrow n$ = number of bits per second.

if $n=1$, Baud rate = bit rate

$n=4$, Bit rate = $4 \times$ band rate

22 With the neat diagram, Explain the working features of SPI protocol.

The serial peripheral interface (SPI) is a synchronous serial communication interface specification used for short distance communication primarily in embedded systems. The interface specification was developed by Motorola.

SPI device communicates in full duplex using a master-slave architecture usually with single master. The architecture

The Master device originates the frame for supported through selection (with individual chip select) some times called slave selected (SSP) bus.

SPI is called a four wire serial bus, contrasting with three, two or one wire serial bus. The SPI may be accurately described as a synchronous serial interface, but it is different from the synchronous serial interface (SSI) protocol, which is also a four-wire synchronous serial communication protocol.

The SPI bus specifies four logic signals:

SCLK : Serial clock (output from master)

MOSI : Master Out Slave In (Data output from slave)

CS/SS : Chip Select

(Often active low o/p from master to indicate the data is being sent)