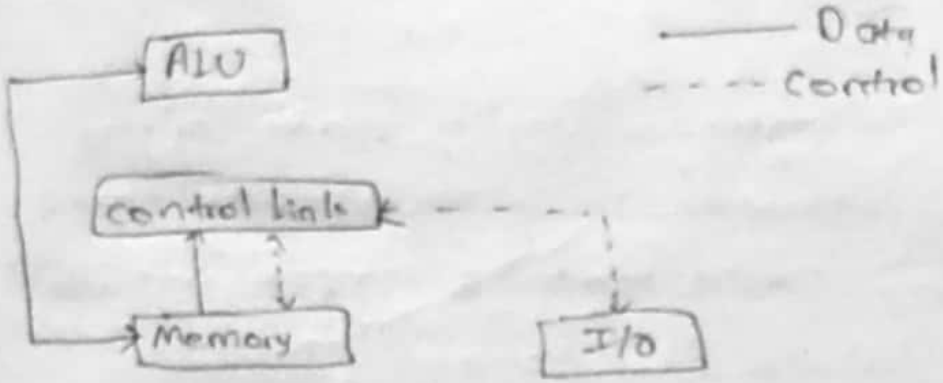
MCA

ASSIGNMENT - 1

① Von-Numan Architecture



All parts are controlled by controll unit CPU and connected together by Bus. Data can pass through bus in half Duplex mode to and from CPU. Memory holds programs & data known as stored program Concept.

Memory is split to small cells with the same size then original numbers are called address number.

Advantages

control unit gets data & instruction in the same way from one memory. It simplifies design & development of the control Unit.

Disadvantages

→ Serial instructions processing does not allow parallel execution. Parallel execution are simulated later by the O.S

→ One Bus is a Bottle neck.

→ Instructions are stored in same memory as the data can be accidently rewritten by an Error in a program

② CISC

CISC was developed to make compiler development easier & simple CISC is complex. Instruction set compute.

RISC

RISC is designed to perform a smaller number of and types of computer instructions. It is micro processor that is designed to perform smaller number of computer instructions RISC is reduced instruction is set computer.

CISC	RISC
→ More set of instructions	→ Lesser set of instructions
→ More addressing mode	→ Lesser addressing mode
→ Minimum amount of RAM	→ More amount of RAM
→ Focus is on Hardware.	→ Focus is on software to optimise.
→ High power consumption	→ Low power consumption.
→ CISC has variable instruction format.	→ RISC has fixed instruction format.
→ Single register set	→ Multiple register set.
→ Less pipelined	→ Highly pipelined.

(3) There are four main types of number systems

* Binary Number System (Base-2)
* Octal number system (Base-8)
* Decimal Number system (Base-10)
* Hexadecimal number system (Base-16)

The Binary number system is used to store the data in Computer.

The advantage of the octal number system is that it has fewer digits when compared to several other systems, hence there would be fewer computation errors.

The decimal number system is the system that we can use in daily life.

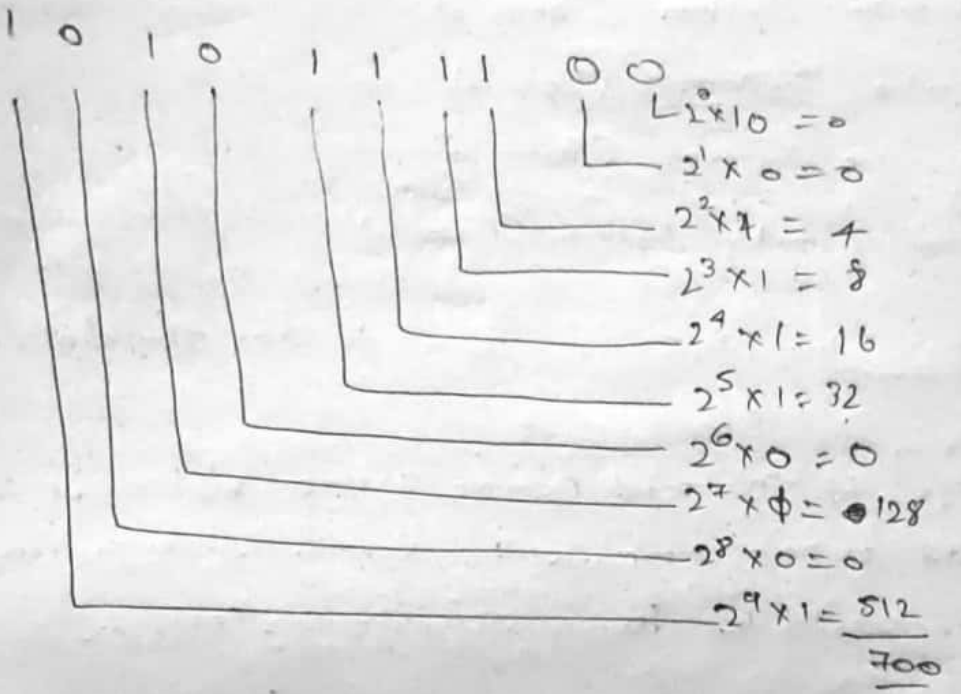
The hexadecimal number system is used in computer to reduce the large sized strings of the binary system

Number System helps in representing the numbers in a small symbol set Binary numbers are mostly used in computer that use digits like 0 and 1 to calculating simple problems. the numbers systems also help in converting one number system to another.

conversion from one number system to another number system

To convert a number from one of the binary/octal/ hexadecimal/ system to one of the other systems, we first convert it into decimal system & then we convert is to the required systems by using the below process

Eg: $1010\ 111100_2$ to decimal



$1010\ 111100_2 = 700_{10}$

16 | 700
16 | 43 – 12 → C
2 – 11 → B

$700_{10} = 2BC_{(16)}$

8 | 700
8 | 87 – 4
8 | 10 – 7
1 – 2

$= (1274)_8$

2 | 700
2 | 350 – 0
2 | 175 – 1
2 | 87 – 1
2 | 43 – 1
2 | 21 – 1
2 | 10 – 1
2 | 5 – 0
2 | 2 – 1
1 – 0

$= (1010\ 111100)_2$

⑤ Arithmetic operators are used to perform mathematical Calculations. Logical operators are used to perform logical Operations & include AND, OR, NOT. Boolean operators include AND, OR, XOR & NOT and can have one of two value true or false.

The arithmetic operators are addition (+) subtraction(-) Multiplication (*) & division (/) exponential (^).

The 3 common three logical operators are

* AND
* OR
* NOT

The logical operators are often used to help create a that expression that controls program flow this type of expressions is also known as Boolean expression because they create a boolean answer or value when evaluated.

Truth table

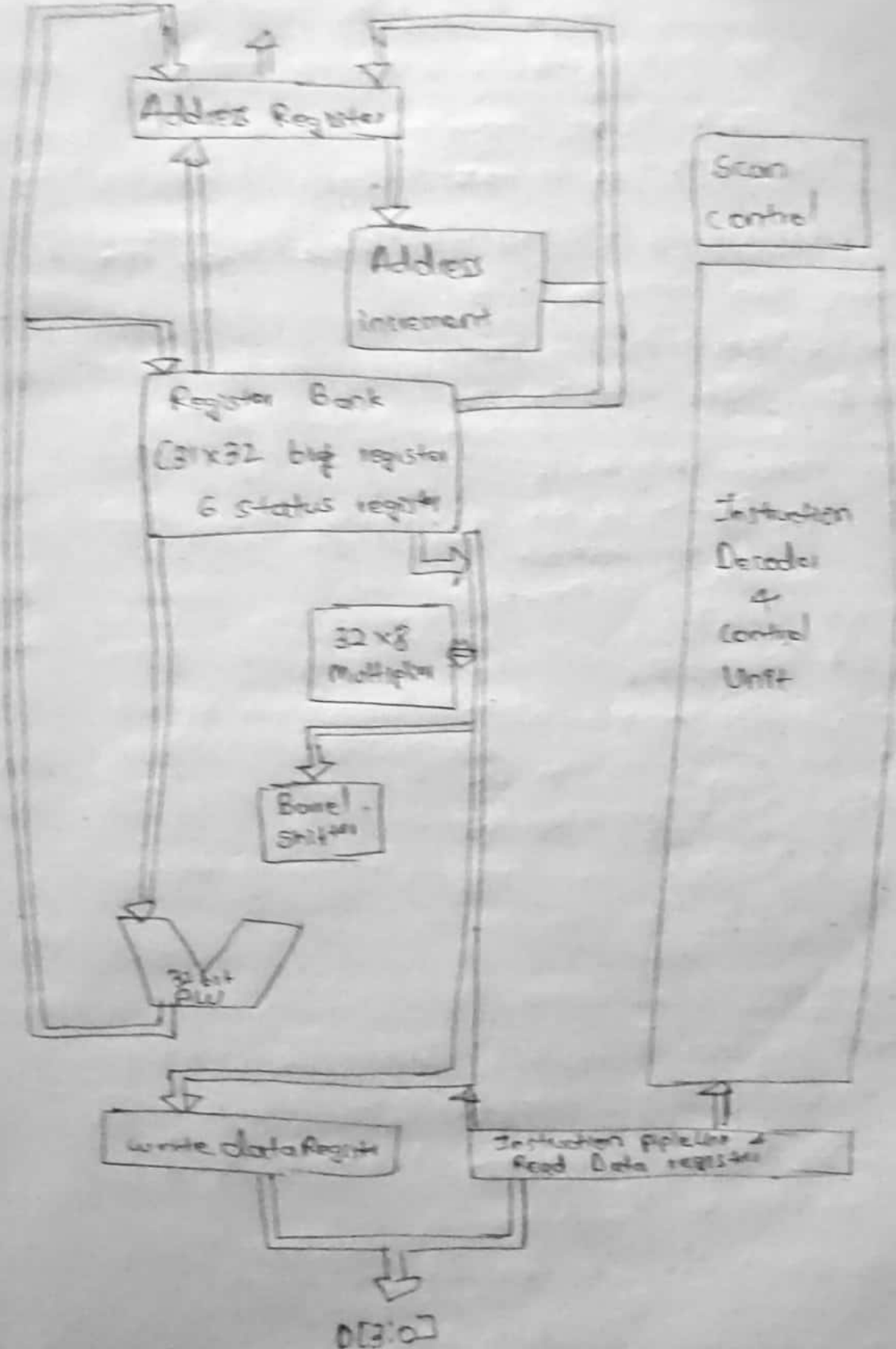
The common way to show logical relationships is in truth table.

x	y	x and y	x or y
false	false	false	false
false	true	false	true
true	false	false	true
true	false	true	true

X	Not X
false	true
true	false

Operator	Meaning	Example	Result
&&	Logical AND	(5>2) && (5>1)	false
||	Logical OR	(5>2) || (5>1)	true
!	Logical Not	!(5>2)	true

(3) Write a neat diagram, explain the ARM7 Architecture



Features used

→ Load / store Architecture
→ Fixed Length 32 bit instructions
→ 3 address instruction formats.

ARM processor is a 32 bit architecture most ARMs implement two instruction sets

→ 32 bit ARM instruction set.
→ 16 bit thumb instruction set.

→ Von-Neuman Architecture.
→ 3 stage Pipeline → fetch
→ decode
→ execute
→ 32 bit Data bus
→ 32 bit Address Bus.
→ 37 32 bit register
→ 32 bit ARM instruction set.
→ 16 bit thumb instruction set.
→ 32 x 8 Multiplier.
→ Barrel Shifter.

Data types

ARM processor supports 6 data types
→ 8 bit signed & unsigned bytes
→ 16 bit signed & unsigned half word aligned.
→ on 2 byte boundaries
→ 32 bit signed and unsigned half words

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

Internally all ARM operations are on 32-bit operands the shorter data type are only supported by data transfer instructions. when a byte is loaded from memory it is zero or signed extended to 32-bit.

ARM co-processor supports floating point values.

④ Programming model for ARM7

Each instruction can be viewed as performing a defined transformation of the states.

* Visible Register
* Invisible register
* System memory.
* User memory.

Processor modes

-> ARM has seven basic operating modes.
-> modes changes by software control or external interrupts.

CPSR[4:0]	Mode	Use	Register
0000	User	Normal user code	User
0001	FIQ	processing fast interrupt	fiq
0010	IRQ	processing slow Interrupt	irq
0011	SVC	processing software interrupt	- svc
0100	Abort	processing memory fault	- abt
1011	Undef	Handling undefined Instn type	- und
1111	System	Running privileged OS	_user

→ Most programs operate in user mode APM has other privileged operating modes which can used to handle exceptions, supervisor calls & system mode.

→ More access rights to memory systems & co-processor

→ Current operating mode is defined by CPSR[4:0].

Privileged modes

· Supervisor modes:-

→ Having some protective previliges

→ System Level function can be accessed through specified supervisor calls.

→ Usually implemented by software interrupt.

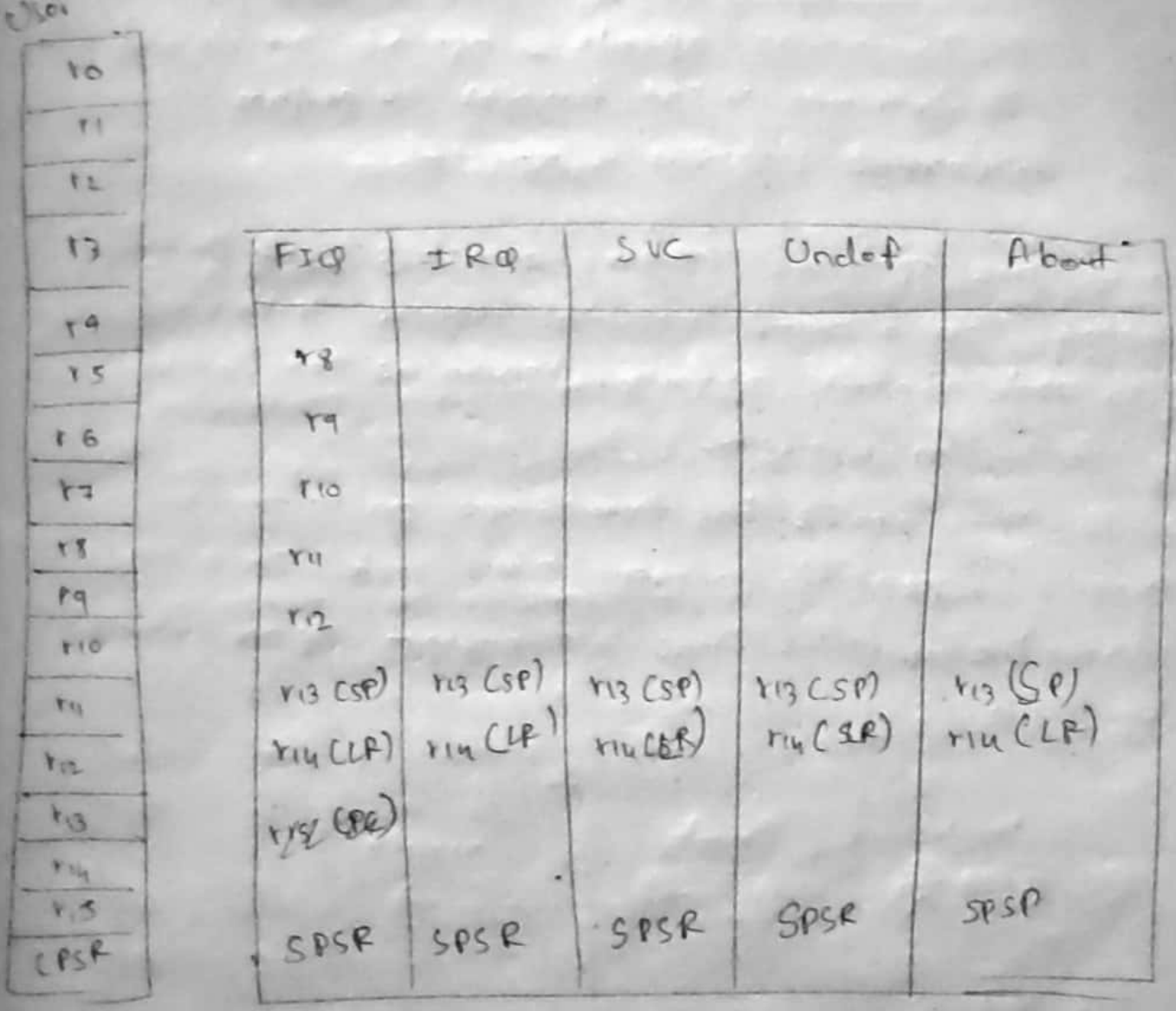
ARM has 37 register all of which are 32 bits

→ Program counter.
→ current program status register.
→ 5 dedicated saved program status register.
→ 30 general purpose register.

Each modes can access

→ as particular set of R_0 - R_{12} register.
→ stack pointer R_{14} - Link register.
→ Program counter (PC)
→ the current program status register (CPSR)

Privileged modes can access a particular SPSR saved program status register



(6) With a neat diagram, explain three stage pipeline of ARM

→ A pipeline is the mechanism used by RISC processor to execute instructions

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

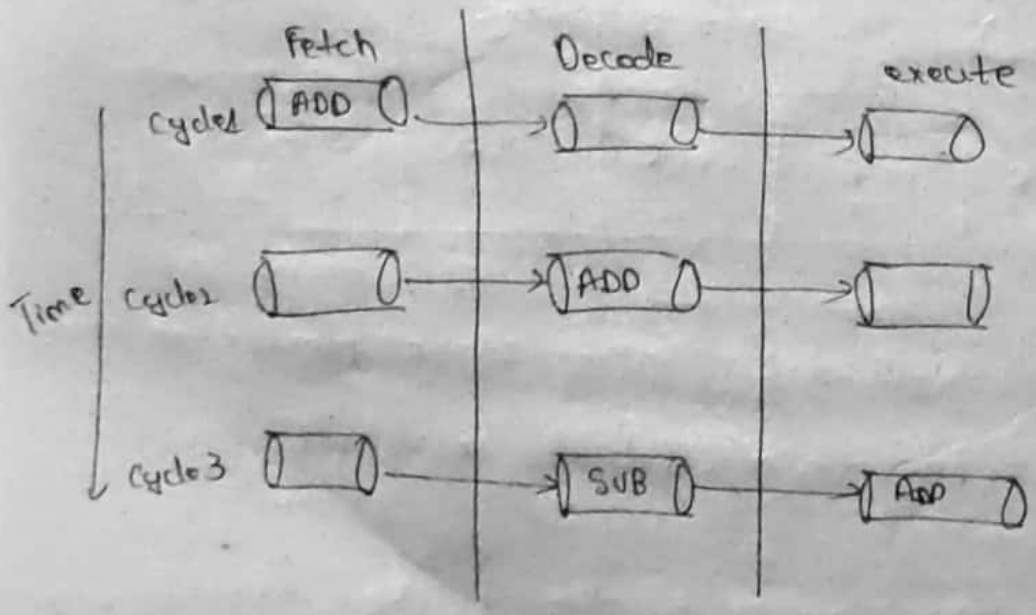
→ pipelining is a design technique a process which plays an important role in increasing the efficiency of data processing in the process of a computer & ...

The ARM 7 has three stage pipeline:

- Fetch: The instruction is fetched from memory.
- Decode: ~~The instruction is fetched from memory~~ The instructions opcode & operands are decoded to determine what function to perform.
- Execute: The decoded instruction is executed.

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



(7) With the neat diagram, explain CPSR register.

[Conditional flag]

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 5TE only

→ indicates if saturation has occured during certain operations

Interrupt Disable bits:-

→ I = 1 Disables the IRQ

→ F = 0 disables the FIQ

T bit

→ Architecture T only

→ T = 0 processor in ARM state.

T = 1 processor in thumb state

Mode Bits

→ Specify the processor mode.

CPSR → Current processor status register holds the information about the current status of the processor.

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

(8) Explain the seven different modes in ARM.

The ARM7TDMI processor has seven modes of operation

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

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

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

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

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

→ System mode is ~~entered~~ after 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 (CPSR).

Undefined mode is entered when an undefined instruction is executed.

Modes other than user mode are collectively known as privileged mode. Privileged modes are used to service interrupts or exceptions or to access protected resources.

Mode	Mode identifier
User	usr
Fast Interrupt	Fiq
Interrupt	IRq
Supervisor	Svc
Abort	abt
System	sys
undefined	und

(9) Explain the nomenclature in ARM

ARM was originally from Arm computer Ltd they RISC processor in commercial use.

ARM7TDMI processor

32 bit processor Advanced machine

T -> thumb Architecture extension.

D -> Debug extension.

M -> Enhanced extension.

I -> In circuit emulator.

ARM {x}{y}-{z} TDMI {F} {J} {F} {S}

x → series

y → Memory management Unit

z → cache

T → thumb 16 bit decoder

D → @JTAG Debugger

M → Fast Multiplier.

I → Embedded ICE) (In Circuit Emulator)

E → Enchanced instruction for DSP

J → JAVA acceleration @ Jazelle.

F → Floating point

S → synthesizable version.

(10) What is JTAG? Explain JTAG State Diagram.

JTAG has become a standard in embedded systems and it is available in nearly every microcontroller and FPGA on the market.

If we have programmed a microcontroller their a strong chance that we have used JTAG or @ the related standard.

JTAG is Joint test Action group is an industry standard for verifying designs & testing printed circuit boards after manufacture.

JTAG is implements standards for on chip instruction is electronics design automation (EDA) as a complement tool to digital simulation. It specifies the use of a dedicated debug port implementing serial comm interface for low-overhead access without to the system address & direct buses. The interface connects to an chip. Test access port (TAP) that implements a stateful protocol to access a set of test registers.

test logic reset

run test idle

select DR scan

capture DR

shift DR

exit 1 DR

pause DR

exit 2 DR

update DR

select IR scan

capture IR

shift IR

exit 1 IR

pause IR

exit 2 IR

update IR

(11) What is single tasking? Give examples of processor application

Single tasking means doing one task at a time with as little distraction & interruption as possible.

Micro controller are known as computer on chip They are designed to perform a single task only because its processing power as well as memory is not suitable to installing an O.S.

(12) What is MMU? why MMU is required? Give ex of MMU support.

The memory can be defined as a collection of data in a specific format. It is used to store instructions & processed data. The memory comprises a large array of 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 access, should be in the main memory during execution. The CPU fetches the instructions 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 & 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 is extremely fast.

Main memory is also known as RAM. This memory is a volatile memory. RAM lasts its when a power interruption occurs.

Memory Management

In a multiprogramming computer the operating system resides in a part of memory and rest is used by multiple process. The task of subdividing the memory among different processes is called memory management. Memory management is a method in O.S to manage operations blw main memory & disk memory during process execution. The main aim of memory management is to achieve efficient utilization of memory.

Why memory management is required?

-> Allocate & de-allocate the memory before & after the process execution.

-> To keep track of used memory space by process

-> To minimize fragmentation issues.

-> To proper utilization of main memory.

-> To maintain data integrity while executing of process.

Eg: IBM system/360 model 67, IBM System 370

ARM

ARM architecture based application processes implement an MMU designed by ARM's virtual memory system architecture. The current architecture defines PTE's for describing 4kB and 64kB pages, 1MB sections and 16MB super sections. Legacy versions also defined 1kB tiny pages

(16) write a C program to find the endianness of given number

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

(15) Explain following

→ Bit: A bit is the smallest unit of information that can be stored in a computer. Bits in computer are grouped to form a large unit of information.

→ Byte: A byte is a combination of eight bits. Eight bits represent a character and is called a byte.

→ nibble: A nibble is a combination of four bits, in other words a nibble is half byte.

→ wad: A wad is a combination of 16 bits, 32 bits or 64 bits depending on the computer. It is known as quad wad.

Length	Name	Example
1	Bit	0
4	Nibble	1011
8	byte	1011 0101
16	Halfwad	1011 0101 1001 001

(16) Explain the wad align & Half wad align in ARM memory

Different processors have different definition of words. In 32 bit processors, a word is 32 bit (4 bytes). As the name implies, a half word is 16 bit. In a 16 bit processor a word is 16 bit (2 bytes), in 8 bit processor word is 8 bits.

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

Half word alignment : That is the stored addresses are adjacent & divisible by 2, that is the last bit is 0.

ARM architecture requires 32 bit ARM instructions that must be word aligned & stored in memory & 16 bit thumb instructions requires Half word aligned & stored. Therefore in ARM state the value of R_{15} is always divisible by 4, that is lowest 2 bits of the R_{15} register are always 00. In the thumb state the value of R_{15} is always divisible by 2. which is the lowest bit of the R_{15} register always 0. One word consists of one or more bytes i.e usually integer bits of bytes.

(18) Explain the following addressing modes in ARM.

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

Ans Sequence of instructions forms a program to perform a specific tasks

2 components — Op-code field → Specifies how data manipulated
Address field → Specifies the data location

When data to be read from or store in two a more address field may have one or more than one address

* Three address Instruction
* Two address Instructions
* One address Instructions
* Zero address Instructions

Processor can execute an instruction only if it is represented in Binary Sequence.

Unique binary sequence pattern must be assigned. This process is called op-code encoding.

One address Instructions

This uses an implied Accumulate register to data manipulation and the other is in the register memory location Implied means that the CPU already knowns that one operand is in the accumulator so there is no need to specify it.

Eg: LDR addi
ACC ← (addr)

Two address instructions

Here two address can be specified in the instruction. In the one address instruction, the result was stored in the accumulator, Here the result can be stored in different locations, i.e register or memory locations But requires more numbers of bit to represent the address

Ex: MOV R_1, R_2
$R_1 \leftarrow [R_2]$

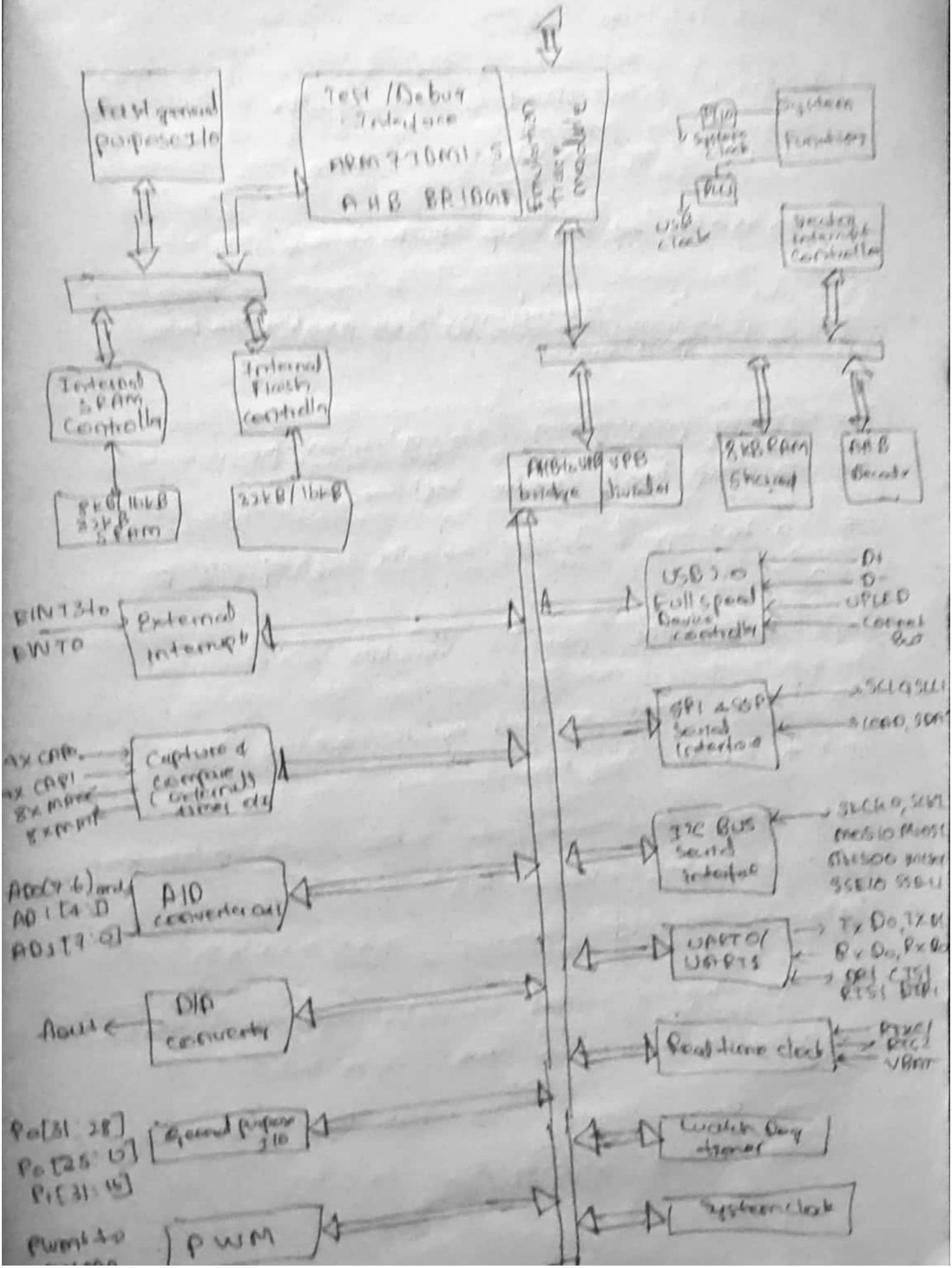
Three address Instructions

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

Eg: ADD R_3, R_1, R_2

$R_3 = R_1 + R_2$

(19) Explain the LPC2148 Microcontroller Block diagram.



LPC 2148 Features

- 16 bit / 32 bit ARM7TDMI-S microcontroller
- 8kb to 40kB of on chip static RAM.
- 32kb to 512kb of on chip flash memory.
- 128 bit bit wide Interface/ accelerator enables 60 MHz high speed operations
- In System programming / In application programming via on chip boot Loader software.
- USB 2.0 Full speed device controller 2kB of end point RAM.
- Single 10 bit DAC provides variables analog output
- Two 32 bit timers / external event counters, PWM unit + Watch dog timer.
- Low power Real time clock (RTC) with independent power & 32 MHz clock input.
- upto 21 external interrupt pins available.
- The on chip integrated oscillator operates with an external systems from 1MHz to 25 MHz.
- Single Power supply with POR, BOD circuits CPU operating voltage range of 3.0V to 3.6V.

(20) Explain the LPC 2148 microcontroller GPIO Pins

Ans GPIO → General purpose Input/output. A 32bit register used to select the functions of pins in which the user needs to operate. There are four functions to each pin of the controller which the first function is GPIO. It means that the pin can either act as an input or output with no specific function.

There is totally three PIN registers in LPC 2148 controller in order to control the functions of the Pins in the respective ports. The classification is given below.

PIN SEL0 → controlls functions of Port 0.0 - Port 0.15

PIN SEL1 - controls the functions of Port 0.16 - Port 0.31

PIN SEL2 - controls functions of Port 1.0 - Port 0.31

where 00 of 32 bits registers are GPIO configuration

LPC 2148 has two 32-bit General purpose I/o ports A total of 30 I/o and a single output only pinout of 32 pins are available to GPIO function PORT0 & PORT1 are controlled via two groups of 4 registers

* IO PIN
* IODIR
* IO SET
* IO CLR

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 regardless will give the logic value of the pin regardless of whether the pin is configured to input or output as a GPIO or an alternate digital functions

As an example: that particular port pin may have GPIO input, output, UART receive, PWM output as selectable functions Any configuration of that pin will allow its current Logic state tobe read from the IOPIN register. If a pin as an analog function as one of its option the pin state cannot be read if the analog configure is selected. while the pin as an A/D input disconnects the digital features of the pin, the pin value read in the IOPIN register is not valid. writing to the IO register stores the value in the port output register, bypassing the need to use both the IOSET & IOCLR register to obtain the entire written value. This feature should be used carefully in an application since it affects the entire port.

IOSET

This register is used to produce a High level output at the port pins configured as GPIO in an output mode. Writing 1 produces a HIGH level at the corresponding port pins. writing 0 has no effect. If any pin is configured as an IIP 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 write to IOSET register returns the value of this register, as determined by previous write to IOSET & IOCLR. This value does not reflect the effect of any outside world influence on the IO pins.

IODIR

This word accessible register is used to control the direction of the pins when they are configured as GPIO port pins ~~for~~ output mode. writing 1 produces a low level at the corresponding port pin & clear the corresponding bit in IOSET register. writing 0 has no effect. If any pin is configured as an input or a secondary function, writing to IOCLR has no effect.

IOCLR

This register is used to produce a low-level output on port pins configured as GPIO in an output mode. writing a 1 produces a low level and the corresponding port pin and clears the corresponding bit in IOSET register. writing 0 has no effect. If any pin is configured as an input or a secondary function, writing to IOCLR has no effect.

21) write a neat diagram explain Baud rate & Bit rate. Explain the calculations.

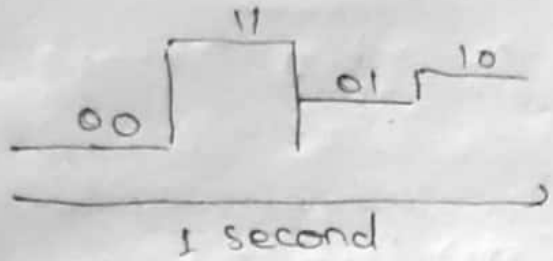
Baud : How many time a signal changes per second

Bit: How many bits can be sent per time unit. Usually (per second).

Bit rate is controlled by baud and number of signal levels

Baud rate

-> Number of time Line changed per second.
-> Let Baud rate be 4 (4 changes per second)
-> Let bits per line change be 2.
-> Bit rate = 8 bits per second.
-> Bit rate = x2 Baud rate in this example.



→ Baud rate defines the switching speed of a signal.

→ Bit rate defines the rate at which information flows 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 bits in each signal unit If 1000 signal units are sent per second, find the Baud rate & the bit rate.

1 bit → 1 symbol

Bit rate = Baud rate

1000

1000 = Baud rate

Bit rate = 1000 × 4 = 4000 bps

if bit rate (or data rate) is "b"

Baud rate (or symbol rate) is "s"

General formula

$$b = s \times n$$

b = Data Rates (bits per second)

s = Symbol rate / symbol / sec)

n = number of bits per second

if n=1, Baud rate = Bit rate

n=4, Bit rate = 4 × Baud rate

(22) With a 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 system. The interface specification was developed by Motorola.

SPI devices communications in full duplex using a master-slave architecture usually single master. The master device originate the frame for reading a writing. Multiple slave device may be supported through selection with indirect chip select (CS). sometimes called slave select (SS) lines.

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

The SPI Bus specifies four logic signals SCLK:
Serial Clock (o/p from master)
MOSI: Master out Slave In (data from master)
MISO: Master In Slave out (data output from slave).
CS/SS: Chip / slave Select.
(often active Low o/p from master to indicate the data is being sent).

SPI Master

SClk
MOSI
MISO
$\overline{SS}$

SClk
MOSI
MISO
$\overline{SS}$

SPI slave

(22) In SPI with a neat timing diagram explain (CPHASE, CPOL).

CPOL determines the polarity of the clock the polarities can be converted with a simple Inverter.

CPOL=0 is a clock which idles at 0, and each cycle consists of a pulse of 1. That is the leading edge is the rising edge and the trailing edge is a falling edge.

CPOL=1 is a clock which idles at 1 and each cycle consists of a pulse of 0, That is the ready edge is a falling edge and trailing edge is the rising edge.

CPHA determine the timing (i,e) of the data bits relative to the clock pulses. Conversion blw the two forms is non-trivial.

For CPHA=0 the "out side" changes the data on the trailing edge of the precedence clock cycle, while the "in" side capture the data on the leading edge of the clock cycle.

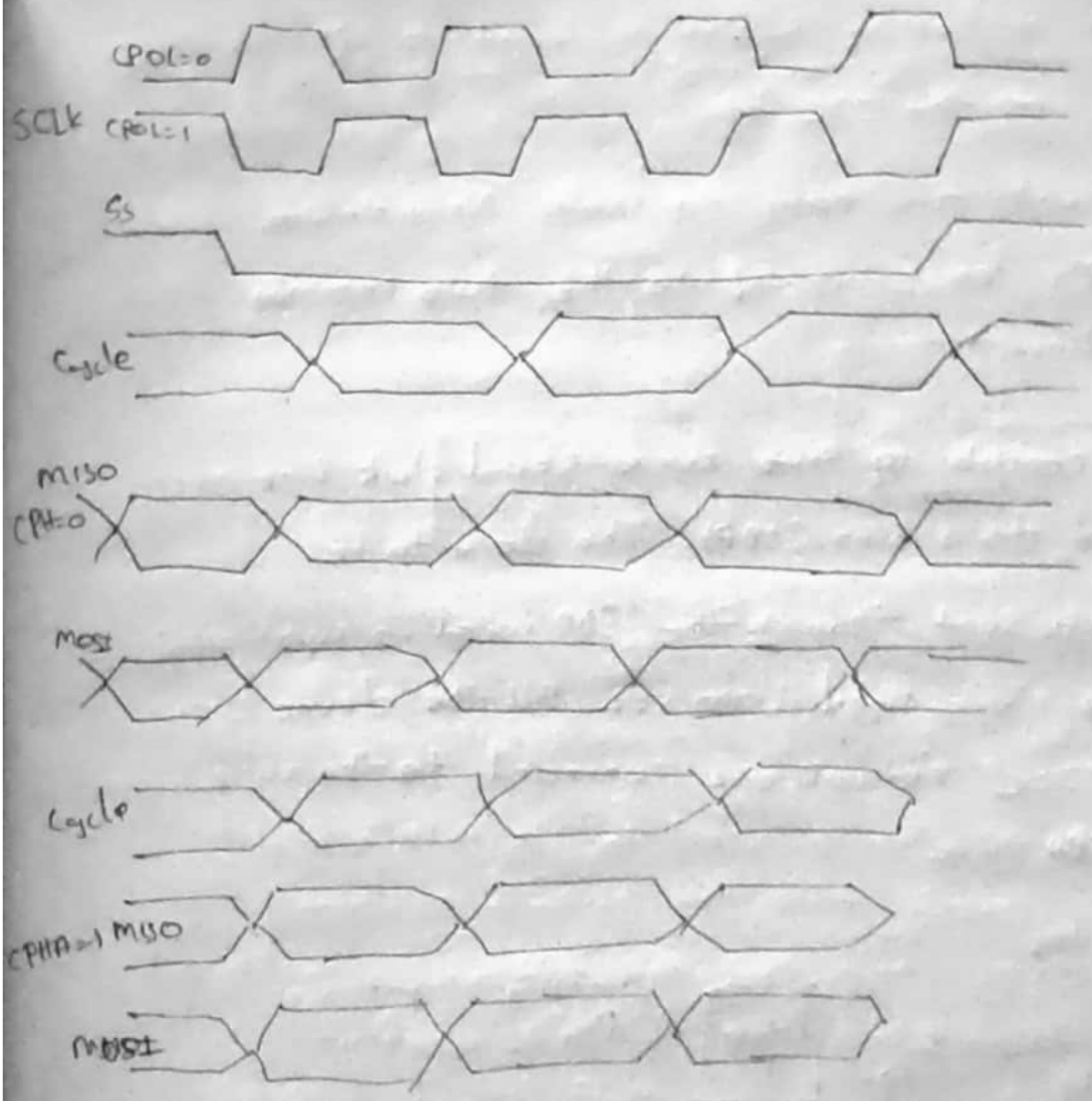
The out side holds the data valid untill the trailing edge of the current clock cycle. For the first cycle, the first bit must be on the MOSI line before the leading clock edge. An alternative way of considering it is to

that a CPHA=0 cycle consists of a half cycle with the clock idle, followed by a half cycle in the clock asserted.

For CPHA=1 the "out side" changes the data on the leading edge of the clock cycle, while the "in" side capture the data on the trailing edge of the clock cycle. The outside holds the data valid untill the Leading edge of the following clock, cycle. In the last cycle. the slave holds the miso line valid untily the slave select is diserted. An alternative way of Considering it is to say that CPHA= 1 cycle consists of the half cycle with clock asserted, followed by a half cycle with the clock idle.

The MOSI and MISO signals are usually stable for the half cycle untill the next clock transition SPI master and slave device may will sample data at different points in that half cycle.

P.T.O



(21) With the neat diagram explain the Features of I2C & its working

Ans → Half duplex - serial communication.

→ Synchronous communication protocol.

→ only two common bus line (wire) are required to control any device /IC on the I2C network.

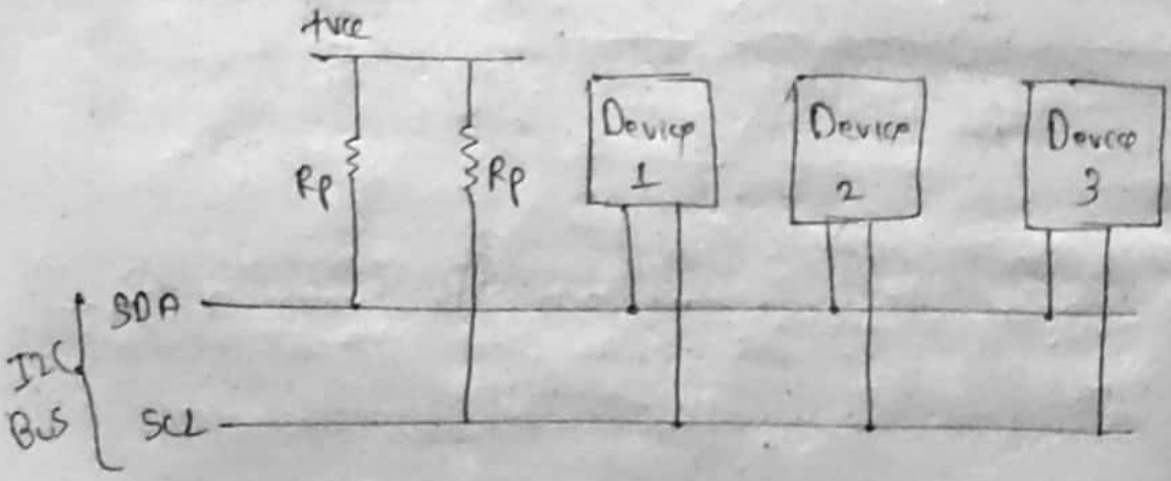
→ Data transfer speed can be adjusted whenever required

→ Simple mechanism for validation of data transfer

→ Uses 7 bit addressing system to target a specific device/IC on the I2C bus

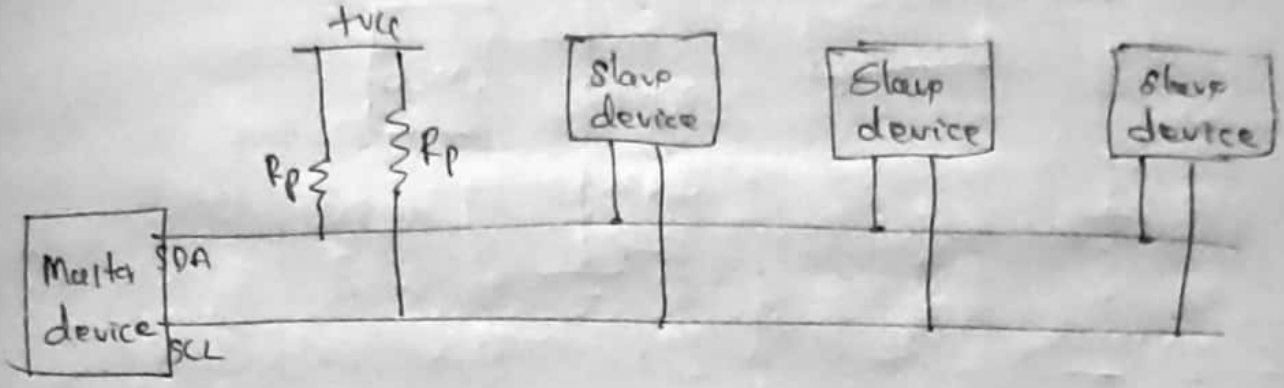
→ I2C network are easy to scale. New devices can simply be connected to the two common I2C bus lines

I2C Bus consists of two wires serial clock line (SCL) and serial data line (SDA). The data to be transited is sent through the SDA wired in synchronised with the clock signal from SCL All the device ICs on the I2C network are connected to the same SCL & SDA Lines.



Both the I2C bus lines (SDA, SCL) are operated in open drain drivers. It means that any device on the I2C bus can drive SDA and SCL Line but they cannot drive them high, so a pull up resistor is used for each bus Line to keep them highly by default.

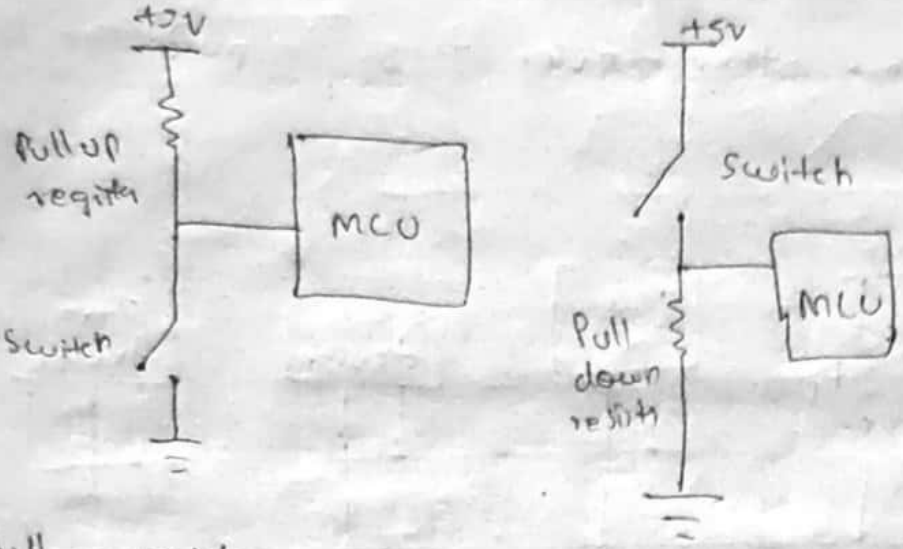
The reason to using an open drain system is only that there will be no chances of shorting, which might happen when one device tries to pull the line high and some other device tries to pull the line low.



The device are connected to the I2C bus are categorised as either masters a slaves. At any instant of time only a single master stays active on the I2C bus. It control the SCL clock line & decides what operation is to be done on the SDA data line.

All the devices that responds to instruction from this master device are slaves. For differentiating blw multiple Slave devices connected to the same I2C bus, each Slave device is physically. when a master device wants to transfer data on or from a slave device, it specifies this particular slave device address on the SDA line & then proceeds with the transfer. So effectively communication takes place blw the master device & a particular Slave.

(25) With a neat diagram, explain the pull up & pull down resistor.



Pull up resistor

A pull up resistor is used to establish an addition Loop over the critical components while making sure that the voltage is well-defined even when the switch is open

It is used to ensure that a wire is pulled to a high Logical Level in the absence of an input signal. Pull up resistor with a fixed value was used to connect the voltage supply and a particular pin in the digital circuit.

Pull down resistor

A pull down resistor is used to ensure that I/P to the Logic systems settle at expected Logic Level whenever the external device are disconnected or a high impedance.

It ensures that the wire is at a defined low logic level when there are no active connections with other device. The pull down resistor holds the logic signal near to zero volts when no other active device is connected.

(28) With a neat diagram explain the concept of arbitration in I2C

→ I2C is designed for multi master purpose that means that more than one device can initate transfer.

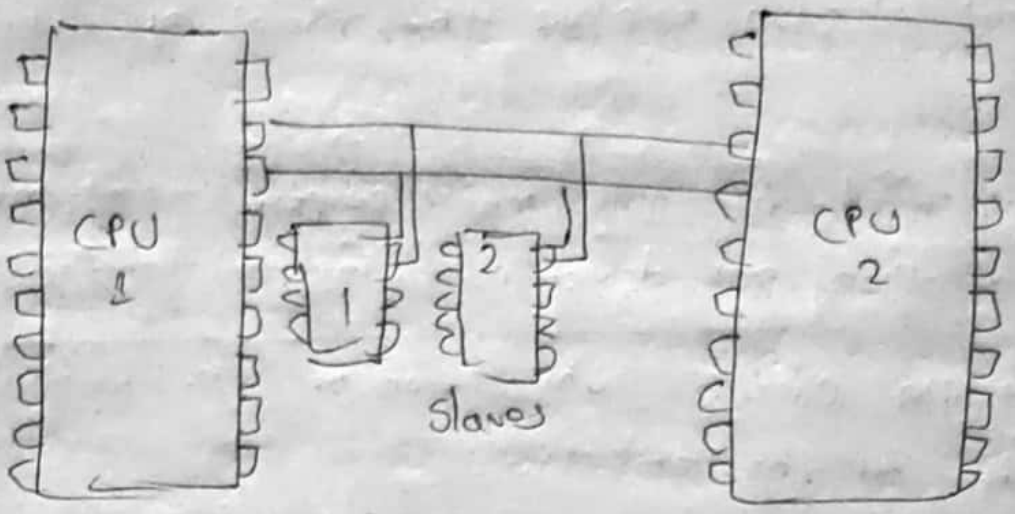
→ But arbitration occurs when two or more masters start a transfer at the same time.

→ The I2C bus was originally developed as a multi master bus. This means that more than one device initiating transfer can be active in the system.

→ when using only one master on the bus thus no real risk of captured data except it a slow down is malfonctoning or if there is a fault condition involving in the SDA/SCL bus.

→ when MCU1 issues a start condition & sends the address the slave will listen. If the address does not match the address of CPU2, this device by to hold back any activity until the bus becomes idle again after a stop condition.

As long as the two MCU's monitor what is going on the bus and as long as they are aware that a transaction is going on because that last issued command was not a STOP, there is no problem



(27) What is clock stretching. Explain clock states in I2C

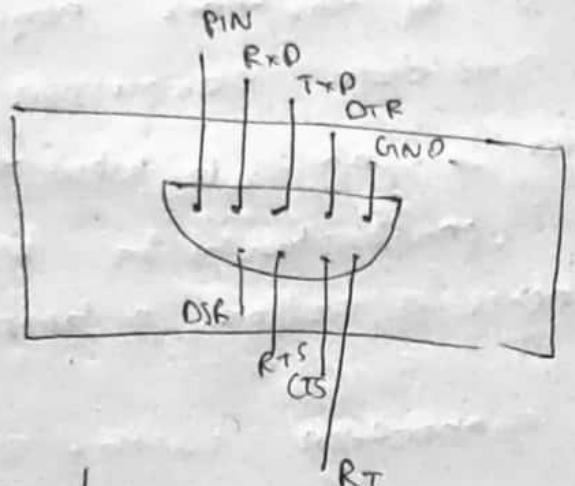
Ans clock stretching allows an I2C slave device to force the master device into a wait state. A slave device may perform clock stretching when it need more time to manage data such as store recei data a prepare the tranimit another byte a

clock streetching in I2C devices can slow down communication by stretching SCL during an SCL phase any I2C device on the bus may rise again enabling, enabling them to slow down the SCL-Clock rate a to stop I2C communicate for a while. This also referred to as Clock

Synchronisation.

In an I2C communication the master device determine the clock speed which relieve master and slave from synchronization exactly to a predefined baud rate.

(8) Explain the working of DB9 pins and handshaking with the modem



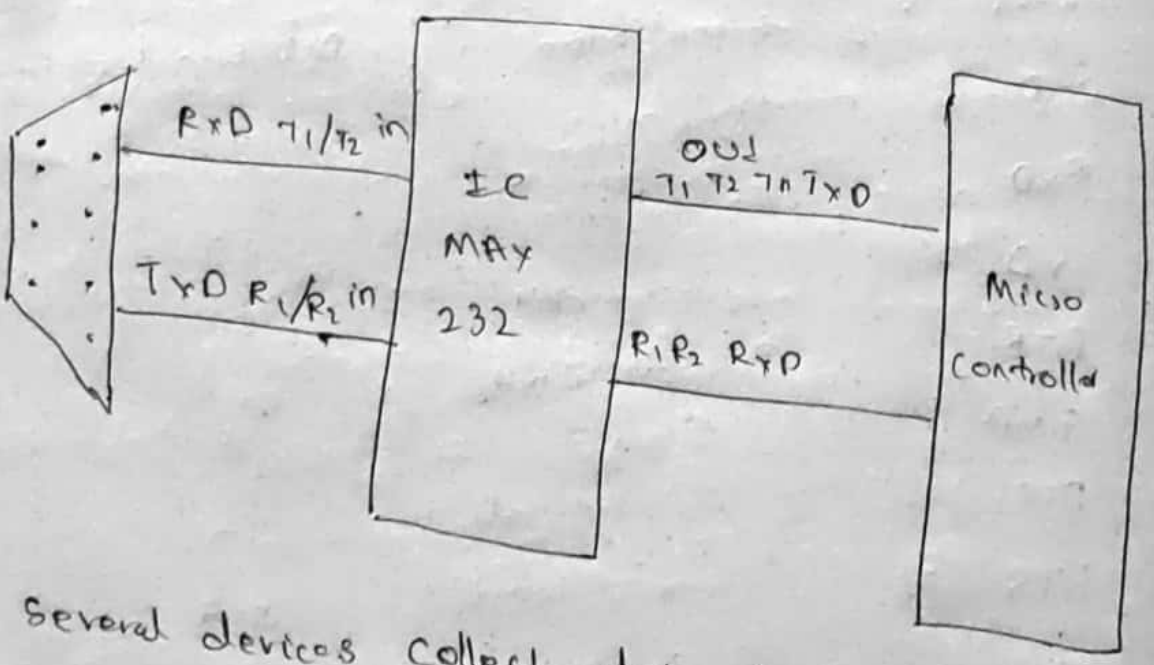
Pin	Signal	Signal name	DTE Signal direction
1	DCD	Data carrier detect	In
2	RxD	Receive data	In
3	TxD	Transmit data	Out
4	DTR	Data terminal ready	out
5	GND	Ground	—
6	DSR	Data set ready	In
7	RTS	Request to send	out
8	CTS	Clear to send	In
9	RI	Ring indicator	In

Handshaking modem

A modem handshake is what occurs when the receiving modem answer the phone call and the two modems ~~answer~~ begin to communicate.

Before anything else happens the modem must calculate the quality of the time negotiate error control protocols and data compression that they can both recognise & walk to what the most suitable connection speed should be based on the conditions. This process is called a handshake.

(29) Explain the RS232 connection with a microcontroller



Several devices collect data from sensor and need to send it to another until Like a computer for further processing. Data transfer / communication is generally used for Long distance communication In serial communication the data is sent one bit at a time.

An important parameter considered while interfacing signal post is the baud rate which is the speed at which data is transmitted serially. Microcontroller can be set to transfer and receive signal data at different baud rate using software instructions.

(36) Explain the frame format in UART communication.

Baud rate: Baud rate is a data transmission refers to the number of symbols transferred per second. A symbol is a group of a fixed number of bits.

Data Framing