CO Assignment 2

Ques 2:

Distinguish between different interrupts like external vs internal interrupts; vectored vs non-vectored interrupts; hardware vs software interrupts; and maskable vs non-maskable interrupts and compare their merits and demerits.

Ans2:

Hardware vs software interrupts:

Hardware Interrupts are the interrupts caused by the connected devices.
Whereas software interrupts are those which are introduced by software instructions deliberately to generate user defined exceptions.

External and Internal Interrupts:

Internal interrupt is when there's no interference from outside, i.e. peripherals,user or any external network. It is coded within the program to 'automatically' switch to another task. It generally avoids the user to make changes, it happens naturally as the program proceeds without any separate signal or instruction. Whereas the External interrupt occurs when there's some interference from the outside which maybe from the user or any external network.

Maskable and non-maskable:

A maskable interrupt is the one which can be temporarily 'masked','disabled',or 'delayed' by the programmer. Eg. INTR,RST5.5 etc. Whereas, non-maskable interrupt is the one which cannot be disabled by the programmer. Eg Reset IN.

Vector and non-vectored Interrupts:

In case of a vectored interrupt, the address of the ISR is known. Peripheral must provide the address of the ISR.It is common when microprocessor has multiple peripherals connected by a system bus .Whereas, in case of non-vectored interrupt the device never sends an interrupt vector. As soon as the interrupt received by the CPU it jumps the program counter to a fixed address in hardware.

Ques3:

What is the maximum and minimum floating point number that can be represented in IEEE-754 single and double precision format? Write an algorithm for implementation of addition and substraction of the same and provide the h/w implementation with the example.

Ans3.

ALGORITHM FOR FLOATING POINT ADDITION/SUBTRACTION

1.Compare exponents.Shift smaller no. to right till exponents match.
2.Add the significands.
3.Normalize the sum.
4.If overflow or underflow occurs throw an exception else goto step 5.
5.Round the sum to appropriate no. of significant digits.
6.Set sign bit to sign bit of no. bigger in magnitude.

Ques4:

Write an algorithm for implementation of floating point multiplication and division with the related H/W where the numbers are represented in IEEE-754 single precision format with the example of both the operands are negative numbers?

Ans4.
ALGORITHM FOR FLOATING POINT MULTIPLICATION/DIVISION

1.Add exponents.Subtract the bias.
2.Multiply significands.
3.Normalize product.
4.If overflow or underflow occurs throw an exception else goto step 5.
5.Round the product to appropriate no. of significant digits
6.Set sign bit 0 if both nos have same sign bit else set it to 1.