**(3 Hours)**     **Total Marks: 80**

**NB:**

1) Question no. 1 is compulsory

2) Solve any three from the remaining five questions.

3) Assume suitable additional data if necessary.

Q1     Answer the following questions.     (20)

a) Justify the need for brown-out detection circuit in embedded systems environment and the mechanism of implementing the same.

b) With respect to power, performance and cost state and explain the associated design metrics for an embedded system.

c)  Explain the structure of typical C source program for ARM based target processor.  Typically list the various data types along-with memory size supported by a C compiler.

d) What are interrupts and explain the factors that contribute to interrupt response time in a system.

Q2 a) With regards to Cortex – M3 architecture, explain the various states and its modes of operation.     (10)

b) Explain the utilisation bound in task scheduling in light of Rate Monotonic Scheduling algorithm.     (10)

Q3 a) What is a task and various states that a task can lie in for an embedded environment.     (10)

b) Explain briefly the memory and bus structure in Cortex-M3 architecture.   (10)

Q4  a) Explain briefly the serial communication protocol RS 232. What are the advantages of RS – 485 over RS -232 communicaction.     (10)

b) Explain the operation and significance of following MicroC/OS – II functions

    a) OSSemCreate(); OSSemPend(); OSSemPost(); b) OSInit(); OSStart();   (10)

Q5  a) Compare the features of Cortex – A8 and Cortex - R4 architectures.     (10)

  b) Explain the various inter-process/task communication and synchronisation

    tools like semaphores, mutex, mailbox and pipe used by an RTOS  environment.

                                                                             (10)

Q6)    Write short notes on (Any two) (10 x 2)                 (20)

a) Problem of priority inversion and mechanism to prevent the same.

b) MSP-430 architecture and its low power capability.

c) Design metrics for a typical embedded system.

d) Black-box and White-box testing

****************