Seat No.: _____ Enrolment No._____

# GUJARAT TECHNOLOGICAL UNIVERSITY

**BE - SEMESTER–III • EXAMINATION – WINTER 2013**

**Subject Code: 130704 Date: 26-11-2013**

**Subject Name: Computer Organization and Architecture**

**Time: 02.30 pm - 05.00 pm Total Marks: 70**

**Instructions:**

1. **Attempt all questions.**
2. **Make suitable assumptions wherever necessary.**
3. **Figures to the right indicate full marks.**

| | | | |
|---|---|---|---|
| **Q.1** | **(a)** | Define register transfer language. Draw and explain the block diagram for transfer of data from R1 to R2 when control p = 1. | **07** |
| | **(b)** | Draw and explain the 4-bit binary adder-subtractor circuit. | **07** |
| **Q.2** | **(a)** | List and explain functionalities of the basic computer registers and memory. | **07** |
| | **(b)** | What is an Instruction Cycle? Draw its flow chart. | **07** |
| | | **OR** | |
| | **(b)** | List memory reference instructions and explain any one with example. | **07** |
| **Q.3** | **(a)** | Write an ALP for swapping two numbers. Indicate the comments also. | **07** |
| | **(b)** | Explain the difference between hard wired control and micro programmed control. Give microinstruction format. | **07** |
| | | **OR** | |
| **Q.3** | **(a)** | What is the basic functionality of an assembler? Explain its first pass. | **07** |
| | **(b)** | Explain Booth multiplication algorithm. | **07** |
| **Q.4** | **(a)** | What is stack organization? Explain push and pop micro-operations. | **07** |
| | **(b)** | What are status register bits? Draw and explain the block diagram showing all status registers. | **07** |
| | | **OR** | |
| **Q.4** | **(a)** | List the addressing modes. Explain any one with example. | **07** |
| | **(b)** | What is overlapped register window? How window size and register file size is computed? | **07** |
| **Q.5** | | Attempt ANY FOUR | **14** |
| | **(a)** | Characteristics of RISC | |
| | **(b)** | SIMD and MIMD | |
| | **(c)** | Vector operation | |
| | **(d)** | Gray code | |
| | **(e)** | Subroutine call and return with micro-operations. | |
| | **(f)** | Zero address instructions. | |
| | **(g)** | Convert (A + B) * [C * (D + E) +F] into prefix and postfix notation. | |
| | **(h)** | Perform (-35) + (-40) in binary with negative numbers in 2's complement format. | |

**********