**Birla Institute of Technology & Science – Pilani**

**Hyderabad Campus**

**I Semester 2013-2014**

**Computer Architecture (CS F342) – Test I**

Date: 28.09.2013 Weightage: 20% Duration: 60 Min. Type: Closed Book

1. Answer **all** questions and parts of a question if any consecutively
2. Assume that there are no typographical errors
3. No partial marking for any question or part of a question
4. Total no. of pages in the question paper: 2
5. Answer the following 10x2 = 20**M**
6. When two equal numbers are subtracted, the result would be --------- and not ---------
7. (-27) can be represented as --------- in a signed magnitude format (16 bits)
8. The mathematical representation for 2’’s complement number is ----------
9. Write the equivalent MIPS instructions for the pseudo instruction given below,

blt $s0, $s1, Ll

1. Using MIPS assembly, how can you load a 32 bit number (0000 0000 0010 1011 0000 0001 0110 1001) into a register $s0?
2. The division operation (-17)÷(-5) results in ----- reminder, ------ quotient (binary)
3. When Booth’s algorithm is used, how many ADD and SUB operations are required for **Mx1100111011100111**, where M is multiplicand?
4. The addressing modes used for following MIPS instructions are -----, -----

beq $t0, $t1, L1

j Exit

1. The range of positive numbers that can be represented using IEEE 754 are ------ to ----- (single precision)
2. A digital computer represents its floating numbers using 128 bits. 1 bit is reserved for sign, 14 bits are reserved for exponent and remaining bits are reserved for mantissa. Assuming that the device follows biased notation for exponents, what is the range of true exponents it can support?
3. Represent the following decimal numbers into their equivalent binary representation (single precision) 5x2 = **10M**
   1. +105.625
   2. -(1/16)
4. Perform the division of -145/13, using “restoring two’s complement division” method. Your answer should contain all the steps involved in the division. **10M**

contd...

1. Carefully examine the following MIPS procedure and convert the same to high level “C function”. You should not deviate from parameter passing and return conventions of MIPS assembly. Also, name the function appropriately based on the task it performs. **10M**

**XXX\_F1**:

**beq** $a0, $a1, L2

**slt** $v0, $a1, $a0

**bne** $v0, $zero, L1

**subu** $a1, $a1, $a0

**b XXX\_F1**

L1:

**subu** $a0, $a0, $a1

**b XXX\_F1**

L2:

**move** $v0, $a0

**j** $ra

1. Convert the following C code to its equivalent MIPS assembly. **10M**

**int** s, \*i, a[10];

**for** (s=0, i = a+9 ; i >= a ; i--)

s += \*i;

########### \*\*\*\*\* ###########