BANGLADESH UNIVERSITY OF ENGINEERING AND TECHNOLOGY, DHAKA

L-3/T-1   B. Sc. Engineering Examinations 2022-2023

Sub : **CSE 305** (Computer Architecture)

Full Marks : 210   Time : 3 Hours

The figures in the margin indicate full marks.

USE SEPARATE SCRIPTS FOR EACH SECTION

___

SECTION – A

There are **FOUR** questions in this section. Answer **Question No. 1 (Mandatory)** and any **TWO** other questions.

1. (a) Assume pipelined execution for the following sequence of instructions.

     ins 1: sub $2, $1, $3

     ins 2: and $2, $2, $5

     ins 3: or $6, $2, $7

   Assume that **no data forwarding mechanism** is available to you. Construct the execution pipeline for the given sequence along with the stalls required to produce correct output. Evaluate the total number of clock cycles required to execute all three instructions.   **(12)**

   (b) Adapt **data forwarding mechanism** for the instruction set in Question 1(a) to enhance parallelism. Reconstruct the execution pipeline for the given sequence showing all pipelined registers and the forwarding lines. Revise the evaluation of the total number of clock cycles required to execute all three instructions. Explain double-data hazard with the help of your constructed pipeline.   **(16)**

   (c) Formulate the data forwarding condition between ins 1 and ins 2 in your constructed pipeline for 1 (b).   **(7)**

2. (a) Assume that there are two small caches, each consisting of eight (8) blocks. One cache is direct-mapped, and the other is a two-way set-associative cache. For each cache organization.

   (i) Construct the following table for each cache organization for the following sequence of memory block addresses: 32, 36, 35, 32, 48, 32, 51 and 35. Note that you are allowed to subdivide the column "Cache content after access" as needed. Also, when it is necessary to replace an existing cache entry on a cache miss, follow the least recently used (LRU) replacement policy.   **(20)**

Address of Memory Block Accessed	Cache Index	Hit or Miss	Cache content after access

Table 1: Table for Question 2(a)

   (ii) Compute the hit ratio from the constructed tables in (i) for each cache organization.

   (b) How many total bits are required for an **eight-way set associative cache** of 4K blocks, a 4-word block size (i.e., 16 bytes), assuming a 32-bit address? [Show set number and tag bit calculations.]   **(15)**

CSE 305

3. (a) Suppose we have a processor with a base CPI of 2.0, assuming all references hit in the primary cache, and a clock cycle length of 0.5 ns. Assume a main memory access time of 150 ns, including all the miss handling. Suppose the miss rate per instruction at the primary cache is 5%.

(i) Calculate the total CPI for one-level of caching. How much faster the ideal CPU is? **(25)**

(ii) Now assume that we add a secondary cache that has a 15 ns access time for either a hit or a miss and is large enough to reduce the miss rate to 0.5%. The miss rate for primary cache remains 5%. Calculate the total CPI for two-level of caching. [Note that this secondary cache is accessed when there is a miss in the primary cache]

(iii) How much faster will the processor be with a two-level of caching compared to the processor with just one-level of caching?

(b) What is difference between a virtual address and a physical address of instructions of a program? What is the purpose of the virtual addressing of instructions of a program? **(5)**

(c) Consider a mapping of a 64-bit virtual address and to a 30-bit physical address. How many entries are there in the page table if the page size is 8 KB. **(5)**

4. (a) Consider the following branch type instruction format given in Figure for Question 4(a) assuming 32 bit addressing. Construct the formula to calculate branch target address. Draw the single cycle datapath for branch-on-equal instruction assuming 32 bit addressing. [Include only the relevant components and control lines in your design. For example, you can exclude data memory from your design.] **(16)**

Branch	4	-rs	rt	address
	31:26	25:21	20:16	15:0

Figure for Question 4(a): Branch Instruction Format in MIPS

(b) Four hardware components have been added in the ID stage of the pipelined datapath design as shown in Figure for Question 4(b) to reduce branch stalls. Write down the names of these four newly added components. Explain how the branch decision and the target address becomes available by the end of the ID stage of the *beq* instruction. **(12)**

Figure for Question 4(b): Branch Stall Reduction Mechanism

CSE 305
Contd ... Q. No. 4

(c) Briefly explain the following concepts: (7)

   i. Shared Memory Multiprocessor (SMP)

   ii. SISD vs MIMD

SECTION - B

There are **FOUR** questions in this section. **Question No. 5 is mandatory.** Answer any **TWO** questions from Questions No. 6, 7, and 8.
All relevant tables and figures are provided at the end of the question

5. **[MANDATORY QUESTION]**

(a) Rachel has a basic MIPS instruction set without support for multiple languages. In an attempt to add such support, she introduces four new instructions: lh (load halfword), lhu (load halfword unsigned), sh (store halfword), and shu (store halfword unsigned). However, Ross argued that she doesn't need all these instructions to achieve multi-language support. Do you agree with Ross? Justify your answer. (7)

(b) Howard wants to demonstrate his engineering skills, but Sheldon sets a challenging constraint for him. Howard is tasked with designing a 4-bit Arithmetic Logic Unit (ALU), and the constraint is that he cannot use any parallel adder. Instead, he is only allowed to use parallel subtractors and logic gates for the ALU design. The ALU should have three selection variables, denoted as S0, S1, and S2. These selection variables control the ALU to perform various operations on two 4-bit inputs, Labelled as A and B. Your task is to help Howard design the 4-bit Arithmetic Logic Unit (ALU) so that it can do the following operations: (14)

S_2	S_1	S_0	Operation
0	0	0	A - B
0	0	1	A + B
0	1	0	-A -B - 2
0	1	1	B - A
1	0	0	A XOR B
1	1	0	A AND B

(c) Assemble the two object files of Figure for Question 5(c)-1 and Figure for Question 5(c) - 2. Find the updated addresses of the first few instructions of the completed executable file. Note that all the necessary addresses and symbols must be updated in the link process. (14)
[Use Figure for Question 5(c)-3 for reference]

6. (a) Joey claims that the number of cycles is identical to the number of instructions. Do you agree with Joey's statement? Justify your answer. (5)

(b) What support does the MIPS instruction-set architecture provide for synchronization? Explain with an example. (7)
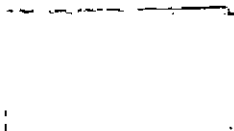
CSE 305
Contd ... Q. No. 6

(c) Write down the equivalent MIPS code for the following C code, considering registers $t1 and $t2 for variables x and y, respectively: **(8)**

```
if (x > y + 131075)
        x = 8;
```

(d) Consider two different implementations (P1 and P2) of the same instruction-set architecture. The instructions can be divided into four classes according to their CPI (class A, B, C, and D) P1 with a clock rate of 2.7 GHz and CPIs of 2, 1, 3 and 2, and P2 with a clock rate of 3.2 GHz and CPIs of 2, 3, 2, and 3.   **(5+5+5) =15**

Leonard develops a program with a dynamic instruction count of 1.2 E6 instructions, which is divided into the following classes: 20% class A, 25% class B, 40% class C, and 15% class D.

(i) Find the number of clock cycles required in both cases.

(ii) Which implementation is faster and how much faster compared to the other one?

(iii) If Sheldon has improved 40 percent of the code for the program, determine the maximum possible improvement factor Sheldon can obtain for that program.

7. (a) Define response time and throughput. Are these two terms synonymous?   **(5)**

(b) Chandler has used 40 variables in his program, whereas the MIPS architecture has only 32 registers. What does the compiler do in such a scenario?   **(7)**

(c) Translate function f (defined below) into MIPS assembly language. If you need to use registers $t0 through $t7, use the lower-numbered registers first. Assume the function declaration for func is "int func(int a, int b);". The code for function f is as follows: **(8)**

```
int f (int a, int b, int c, int d) {
        return func(func(a,b), c+d);
}
```

(d) Write about the main difference between static linking and dynamic linking. With the necessary diagrams, explain how dynamically linked libraries (DLLs) are called.   **(15)**

8. (a) Write down equivalent efficient sequence of MIPS instructions for the following C code fragment. Assume appropriate registers for each of the variables.

```
float a, b, c;
double w, x, y, z;
if (a+b>c)
        x=(y*z)/ w;
```
(7)

(b) Consider a 32-bit system where 19 bits are reserved for fraction, one bit for sign, and the rest for exponent. Answer the following questions: **(2+3+3)=8**

(i) What is the precision range of this system?

(ii) What are the smallest and the largest absolute values that can be stored in this system?

(iii) What is the smallest denormalized number that can be stored using this system?

(c) Consider two normalized floating point numbers f1 and f2 that are represented as IEEE single precision floats. This defines two 32-bit strings.   **(10+10)=20**

(i) Suppose f1 is less than f2, that is, f1 < f2. If the same two 32-bit strings were treated as unsigned integers (call them i1 and i2, respectively), can we conclude that i1 < i2?

(ii) Similar to (i), if the two 32-bit strings were treated as signed integers j1 and j2, can we conclude that j1 < j2?

Object file header			
	Name	Procedure A	
	Text size	150$_{hex}$	
	Data size	40$_{hex}$	
Text segment	Address	Instruction	
	0	lw Sa0, 0(Sgp)	
	4	jal 0	
	...	...	
Data segment	0	(X)	
	...	...	
Relocation information	Address	Instruction type	Dependency
	0	lw	X
	4	jal	B
Symbol table	Label	Address	
	X	-	
	B	-	

Figure for Question. 5(c)-1: Object File Header for Procedure A

Object file header			
	Name	Procedure B	
	Text size	350$_{hex}$	
	Data size	50$_{hex}$	
Text segment	Address	Instruction	
	0	sw Sa1, 0(Sgp)	
	4	jal 0	
	...	...	
Data segment	0	(Y)	
	...	...	
Relocation information	Address	Instruction type	Dependency
	0	sw	Y
	4	jal	A
Symbol table	Label	Address	
	Y	-	
	A	-	

Figure for Question. 5(c)-2: Object File Header for Procedure B
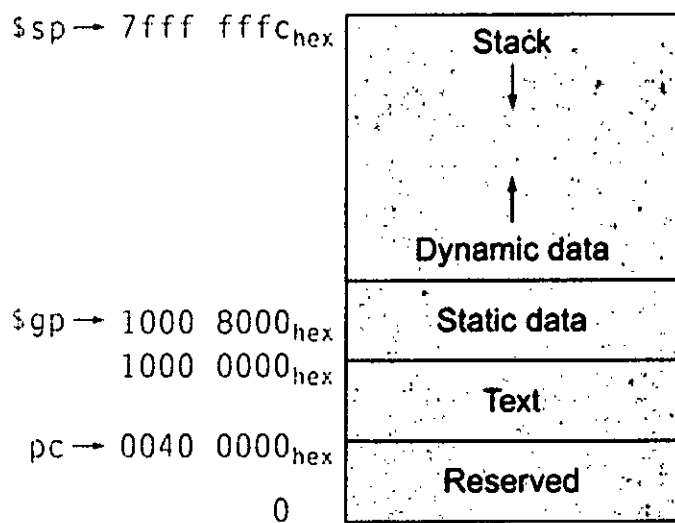
Figure for Question. 5(c)-3: The MIPS memory allocation for program and data.

13|5|2024

L-3/T-1/CSE   Date: ~~31/03/2024~~

BANGLADESH UNIVERSITY OF ENGINEERING AND TECHNOLOGY, DHAKA
L-3/T-1   B. Sc. Engineering Examinations 2022-2023
Sub: CSE 309 (Compiler)

Full Marks: 210   Time: 3 Hours

The figures in the margin indicate full marks.

USE SEPARATE SCRIPTS FOR EACH SECTION

SECTION – A

There are **FOUR** questions in this section. Answer to **Question no. 1 (one) is Compulsory.**
Answer any **TWO** questions from Questions 2-4.

1. **[COMPULSORY]**

   (a) Appraise the differences between bytecode and machine language.   **(3)**

   (b) The partial contents of a buffer in a lexical analyzer is   **(6)**

   cse123--++12.45==␣1␣

   Where ␣ indicates a space. At the current moment, both `lexemeBegin` and `forward` pointers are on c of cse (indicated as position 0). From the perspective of C programming language, write down the positions of the pointers in the table below when this string is scanned. Also, indicate any token identified in the third row. Not all the cells in this row will get populated. There may be more columns than needed in this table.

lexemeBegin														
forward														
Token														

   ⋮

lexemeBegin														
forward														
Token														

   (c) Given below are several statements/entity names related to the linker and loader. Determine which of the statements are related to the linker and which of the statements are related to loader. Only the serial numbers will suffice. You can assign a serial number to either the linker or loader, but not to both.   **(5)**

   (i) Object files.

   (ii) Accountable for loading programs and libraries.

   (iii) Load executable files to memory.

   (iv) Produce executable files.

   (v) Allocate address to executable files.

   (vi) Accountable for managing objects in the program's space.

   (vii) Setting up references that are utilized in the program.

   Partial solutions provided below as examples.

   Linker: i.

   Loader: ii.

CSE 309

Contd ... Q. No. 1

(d) When you carry out left factoring of a grammar, argue carefully that the grammar produced generates the same language as the original grammar. **(6)**

(e) Write a procedure based predictive parser for the following grammar keeping the grammar unaltered: **(5)**

$$E \to E + F \mid E\text{-}F \mid F$$

$$F \to \textbf{NUM} \mid \textbf{ID}(E)$$

Point out any possible flaw of the parser you have constructed.

(f) What will be the general appearances of the LL(0) grammars? Provide relevant examples. Comment on the applicability of these grammars for useful parsing. **(5)**

(g) For a certain context free grammar, there is exactly one production rule for each nonterminal. What kind of parser do you suggest for this grammar keeping optimality in mind? Explain clearly. **(5)**

2. (a) Why are the universal parsing methods not used in production compilers? **(3)**

(b) Elucidate carefully how the SLR(0) parsing table will be different from an SLR(1) parsing table. **(5)**

(c) Eliminate left recursion from the grammar: $A \to Ba \mid Aa \mid c, B \to Bb \mid Ab \mid d$. **(10)**

(d) Explain how the separation of the analysis portion of a compiler into lexical analysis and parsing (syntax analysis) phases, contributes to the adoption of a new language to an already existing compiler. **(5)**

(e) For the grammar given below, find the FOLLOW sets of the non-terminals. **(12)**

$$S \to aABe$$

$$A \to B \mid \varepsilon$$

$$B \to bB \mid Cd$$

$$C \to cC \mid \varepsilon$$

$$D \to dD \mid \varepsilon$$

3. (a) In a certain cell of a predictive parsing table, there are two entries of the form $A \to \alpha$ and $A \to \epsilon$. Explain what this means from the context of the FIRST and FOLLOW sets. **(7)**

(b) Consider the context free grammar given below: **(10)**

$$S \to Aab \mid aS \mid bc$$

$$A \to aacd \mid bef$$

$$B \to bef \mid \epsilon$$

If we consider the above grammar to be LL(k), what will be the value of k? Answer with necessary analysis of each of the productions.

(c) Explain why LL(1) parsing is associated with left-most derivations of parse trees. **(6)**

(d) Show that the grammar: $S \rightarrow AaAb \mid BbBa, A \rightarrow \epsilon, B \rightarrow \epsilon$, is LL(1) but not SLR(1). **(12)**

4. (a) Explain how empty production rules (epsilon productions) are handled in procedure based predictive parsers. **(5)**

(b) For the grammar $S \rightarrow SS+ \mid SS* \mid \alpha$, identify the handles for parsing the right-sentential form aaa*a++. Show your computations. **(10)**

(c) We say that a configuration $[A \rightarrow \beta_1 \cdot \beta_2, \alpha]$ is valid for a viable prefix $\alpha\beta_1$ if there is a derivation $S' \overset{*}{\underset{rm}{\Rightarrow}} \alpha Aw \underset{rm}{\Rightarrow} \alpha\beta_1\beta_2 w$. Why? **(10)**

(b) We have an LR(1) grammar. If we formulate an LALR(1) grammar from this, do you think that the resulting union will have a parsing-action conflict? Justify your answer. **(10)**

SECTION – B

There are **FOUR** questions in this section. Answer to **Question no. 5 (five) is Compulsory**.
Answer any **TWO** questions from Questions 6-8.
(All the symbols have their usual meanings in the context of compilers unless explicitly mentioned)

5. (a) Consider the three-address code below and answer the following questions. **(10+15)**

(i) Identify the basic blocks and construct the flow graph for the given code. Give each block an identifier and replace the label references in the statements with the corresponding identifiers.

(ii) Optimize the code by discovering all possible semantic preserving transformations. For each of the transformations used, specify the name of the transformation.

```
1    L1:  i = 0
2    L3:  t1 = n - 1
3         iffalse i < t1 goto L2
4    L4:  j = i + 1
5    L5:  iffalse j < n goto L6
6    L7:  t2 = i * 8
7         t3 = a [ t2 ]
8         t4 = j * 8
9         t5 = a [ t4 ]
10        iffalse t3 > t5 goto L8
11   L9:  t6 = i * 8
12        p = a [ t6 ]
13   L10: t7 = i * 8.
14        t8 = j * 8
15        t9 = a [ t8 ]
16        a [ t7 ] = t9
17   L11: t10 = j * 8
18        a [ t10 ] = p
19   L8:  j = j + 1
20        goto L5
21   L6:  i = i + 1
22        goto L3
23   L2:
```

CSE 309

Contd ... Q. No. 5

(b) Consider a 'C' code fragment and the corresponding assembly code generated by some 'gcc' compiler using –O2 optimization switch. Now, identify and expain the optimizations that have been performed by the compiler for the given 'C' code. **(10)**

Program written in C	Assembly code generated by gcc
| ```c
#include<stdio.h>
int a[50], b[50],x,y,z,k,n;
int factorial(int n)
{
    int fact;
    for(int i=n;i>=0; i--)
        fact=fact*i;
    return fact;
}
int main(){
    int t;
    t=y;
    y=z;
    z=t;

    x=y^y;

    if(x) n=10;
    else n=20;

    k=factorial(n);

    for (int j=0;j<50;j++)
        for(int i=0;i<100;i++)
            a[j]*=b[j];
    return 0;
    for (int i=0;i<50;i++)
        a[i]=b[i];
}
``` | ```asm
factorial(int):
    xor     eax, eax
    ret
main:
    mov     eax, DWORD PTR y[rip]
    mov     edx, DWORD PTR z[rip]
    xor     esi, esi
    mov     DWORD PTR x[rip], 0
    mov     DWORD PTR n[rip], 20
    mov     DWORD PTR y[rip], edx
    mov     DWORD PTR z[rip], eax
    mov     DWORD PTR k[rip], 0
.L4:
    mov     eax, DWORD PTR a[rsi]
    mov     ecx, DWORD PTR b[rsi]
    mov     edx, 100
.L5:
    imul    eax, ecx
    imul    eax, ecx
    sub     edx, 2
    jne     .L5
    mov     DWORD PTR a[rsi], eax
    add     rsi, 4
    cmp     rsi, 200
    jne     .L4
    xor     eax, eax
    ret
``` |

6. (a) Below is a grammar for simple expressions involving only numbers and the operators '+' and '*' such as 2*(3+4)*5. Write a postfix Syntax-Directed Translation Scheme (SDT) to compute the DAG (Directed Acyclic Graph) for an expression represented by the given grammar. Explain how your SDT would detect common subexpressions. **(12)**

$E \rightarrow E + T$

$E \rightarrow T$

$T \rightarrow T * F$

$T \rightarrow F$

$F \rightarrow (E)$

$F \rightarrow id$

$F \rightarrow num$

CSE 309

<u>Contd ... Q. No. 6</u>

(b) Consider the following E-productions from an SDT for translating infix expressions to postfix notation. Transform the SDT such that it can be implemented during top-down parsing. Show the correctness of your SDT by computing the output for the expression 2+3+4.   (5+4)

$E \rightarrow E_1 + T$ {print('+'); }

$E \rightarrow T$

(c) What is the use of a φ-function is SSA (Static Single-Assignment) form. Explain with a suitable example.   (8)

(d) Write the three-address code for the statement   (6)

```
x = a[b[c[j]]]
```

Assume that- (i) the arrays a, b and c are declared as, int a[5], b[3], c[3]; types and widths of the arrays have correctly been saved in the symbol table, (ii) i and x are integers, and (iii) the size of an integer is 2 byte.

7. (a) Give an outline of syntax-directed translations of *switch*-statements of the following form. Accordingly, show the three-address translation of a *switch*-statement. Briefly discuss how the n-way branch in a *switch*-statement can be evaluated.   (10+5)

```
switch ( E ) {
        case V₁: S₁
        case V₂: S₂
        . . .
        case Vₙ₋₁: Sₙ₋₁
        default: Sₙ
}
```

(b) Write down the three-address code for the statement below avoiding redundant gotos. Assume &&, || are left associative and precedence order is && then ||.   (5)

```
if ( x > 20 && (x < 10 || x ! = y ))
    x = 0 ,
else x = 1;
```

(c) Illustrate, using an example scenario, the difference between basic *mark-and-compact* and *Cheney's copying collector* algorithms for garbage collection. Specify the running-times of both the algorithms.   (12+3)

CSE 309

8. (a) Consider that the following code computes the binomial coefficient recursively. **(10)**

```
int binCoeff(int n, int r)
{
    if (r == 0 || r == n) //base cases
        return 1;

    return binCoeff(n - 1, r - 1) + binCoeff(n - 1, r);
}

int main()
{
    cout << binCoeff(5, 3);
    return 0;
}
```

Now, answer the questions below.

(i) Show the complete activation tree for it.

(ii) What is the maximum number of activation records that will reside in the stack during execution of the program? Explain your answer using your activation tree.

(iii) What does the control stack and its activation records look like when the base cases is satisfied for the first time and the function is about to return. Show only the arguments in an activation record.

(b) Determine the liveliness and next-use information for each statement in the basic block below. Assume t, u and v are temporary variables and the remaining variables are live on exit from the block. **(8)**

```
t = a - c
u = d - b
t = a + b
a = u + t
d = t + v
```

(c) Illustrate the 'linear scan' register allocation algorithm using the following three-address code as an example. Assume three physical registers are available. Make any other justified assumptions as required. What is the advantage of this algorithm? **(15+2)**

```
1          a=b+c
2          d=d-b
3          e=a+f
4          ifFalse e goto L1
5          f=a-d
6          goto L2
7     L1:
8          b=d+f
9          e=a-c
10    L2:
11         b=d+c
```

20/05/2024

BANGLADESH UNIVERSITY OF ENGINEERING AND TECHNOLOGY, DHAKA

L-3/T-1   B. Sc. Engineering Examinations 2022-2023

Sub: **CSE 311** (Data Communication)

Full Marks: 210   Time: 3 Hours

The figures in the margin indicate full marks.

USE SEPARATE SCRIPTS FOR EACH SECTION

___

SECTION – A

There are **FOUR** questions in this section. Answer to **Question no. 1 (one) is Compulsory.**

Answer any **TWO** questions from Questions 2-4.

1. (a) Demonstrate that the coefficients (D_n) of the exponential Fourier series of an even symmetric periodic signal are real whereas those of an odd symmetric periodic signal are imaginary. Now justify the statement for the signal $\cos 2\pi f_0 t$. The symbols carry their usual meanings. **(17)**

   (b) Derive and draw the Fourier spectrum of the standard rectangular signal $\Pi\left(\dfrac{t}{2B}\right)$. Comment on the nature of the frequency spectrum of practical signals. Now analyze why and how sampling process changes the frequency band of a practical signal. The symbols carry their usual meanings. **(18)**

2. (a) The filtering of a signal G(f) by a filter H(f) in frequency domain results in G(f)H(f). Derive its time domain equivalent using necessary transformation. Prove that the same output is obtained when the corresponding time domain signal $g(t) = \Im^{-1}(G(f))$ is passed through a linear time invariant system with impulse response $h(t) = \Im^{-1}(H(f))$. Use only time domain analysis for the later proof. The symbols carry their usual meanings. **(20)**

   (b) Explain the differences between a distortionless system and an all-pass system. How will you determine that a system is distortionless? **(15)**

3. (a) Explain the necessity of different categories of bit stuffing in digital communication. Explain the DM1/2 multiplexer format shown in **Figure for 3(a)** and describe how positive bit stuffing is done in DM1/2 multiplexer. Identify the **exact bit position** if data of channel B and D are to be stuffed. **(15)**

M_0	[48]	C_A	[48]	F_0	[48]	C_A	[48]	C_A	[48]	F_1	[48]
M_1	[48]	C_B	[48]	F_0	[48]	C_B	[48]	C_B	[48]	F_1	[48]
M_1	[48]	C_C	[48]	F_0	[48]	C_C	[48]	C_C	[48]	F_1	[48]
M_1	[48]	C_D	[48]	F_0	[48]	C_D	[48]	C_D	[48]	F_1	[48]

Figure for 3(a)

CSE 311

Contd ... Q. No. 3

(b) With appropriate mathematical derivation, prove that the signal-to-noise ration of a quantized signal is proportional to the squared number of quantization levels (L). Explain the effect of increase of L on the bit rate of the signal. Delta modulation (DM) uses only 2 levels of quantization. Explain what happens to the quantization error in a Delta modulated signal.

(20)

4. (a) Differential pulse code modulation (DPCM) predicts the signal value at a future instance. Show how this *future telling* is mathematically sound. Explain a tapped delay implementation of the predictor function of DPCM. Though delta modulation (DM) is called a 1-bit DPCM, explain how DM replaces DPCM's predictor module simply by an integrator.

(18)

(b) The input stream to a 4B/5B block coder is 1100 0000 0000 0000 0011. What is the output stream? What are the lengths of the longest consecutive sequence of 0's in the input and the output streams? What will be the results of scrambling the input stream using B8ZS, HDB3 and MLT-3 techniques? Assume that the last signal level just before the starting of the given stream has been positive and the number of non-zero pulses is odd after the last substitution. (Please see **Figure for 4(b) and Table for 4(b).**)

(17)

4B/5B mapping codes

Data Sequence	Encoded Sequence	Control Sequence	Encoded Sequence
0000	11110	Q (Quiet)	00000
0001	01001	I (Idle)	11111
0010	10100	H (Halt)	00100
0011	10101	J (Start delimiter)	11000
0100	01010	K (Start delimiter)	10001
0101	01011	T (End delimiter)	01101
0110	01110	S (Set)	11001
0111	01111	R (Reset)	00111
1000	10010		
1001	10011		
1010	10110		
1011	10111		
1100	11010		
1101	11011		
1110	11100		
1111	11101		

Table for 4(b)

CSE 311

Contd ... Q. No. 4(b)

Bipolar 8 Zeros (*B8ZS*) replacement rule: 0 0 0 V B 0 V B

High Definition Bipolar 3-Zero (*HDB3*):

0 0 0 V if No. of nonzero pulses after last substitution is ODD.

B 0 0 V if No. of nonzero pulses after last substitution is EVEN.
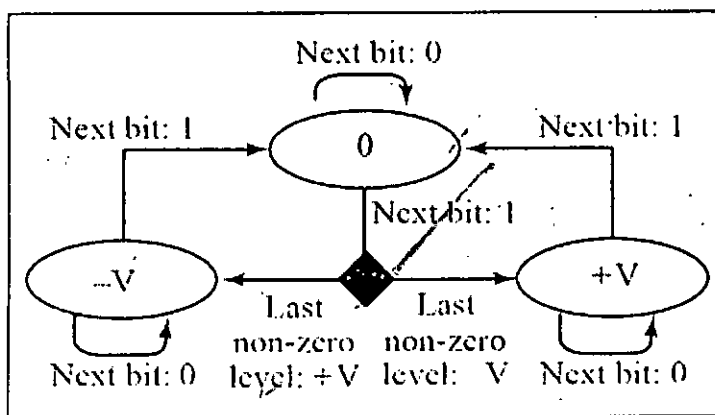
MLT-3 Transition states

Figure for 4(b)

SECTION – B

There are **FOUR** questions in this section. Answer to **Question no. 5 (five) is Compulsory**.

Answer any **TWO** questions from Questions 6-8.

5. (a) Differentiate between analog and digital modes of signal modulation, in your own words. Also, discuss elaborately the **three** advantages of modulation in signal transmission. **(15)**

    (b) "The deviation ratio is 5 in the FM broadcasting system." – Support this statement with proper reasoning. "The **percent modulation** of a frequency-modulated (FM) signal should not be greater than 1, but its **modulation index** can have a value greater than 1 in the FM broadcasting system." – Defend this assertion by distinguishing between the two measures mentioned. **(15)**

    (c) "The bipolar RZ line-coding scheme is better than the bipolar NRZ scheme for binary data transmission." – Judge this proposition based on criteria set by the desirable properties of a line-coding scheme. Also, compare the unipolar NRZ and bipolar NRZ line-coding schemes on the basis of different signal levels deployed. **(15)**

6. (a) An amplitude-modulated (AM) signal is represented by the expression: **(10)**

    $$s(t) = \{6 + 2 \times \cos(3236t)\} \times \cos(2\pi \times 10^6 t) volts$$

    Determine (i) percent modulation index, (ii) modulating frequency, (iii) period of the sinusoidal carrier signal, (iv) peak instantaneous value, and (v) percent transmission efficiency of the AM signal.

CSE 311

Contd ... Q. No. 6

(b) A multiple-tone amplitude-modulated (AM) signal is generated by modulating a sinusoidal carrier signal with three sinusoidal baseband signals. The percent modulation index is 58%, 78% and 38%, respectively, for the three baseband signals. Can you identify any problem with this AM transmission? Consider that the carrier signal has a peak amplitude of 5 *volts*. If there indeed is some problem with the transmission, then state the reason behind it and propose a new peak amplitude for the second baseband signal with a 78% modulation index so that the problem gets resolved. **(10)**

(c) Show that for 100% amplitude modulation, the total power transmitted by an antenna is 1.5 times the carrier power, and each sideband contains around 16.67% of the total transmitted power. **(10)**

7. (a) Derive the expression of instantaneous frequency from the expression of phase angle for a phase-modulated (PM) signal. Likewise, derive the expression of phase angle from the expression of instantaneous frequency for a frequency-modulated (FM) signal. Also, describe how we can use a phase modulator to generate an FM signal and a frequency modulator to generate a PM signal, with appropriate diagrams. **(15)**

(b) A radio station transmits a frequency-modulated (FM) audio signal, where the modulating (baseband) audio signal has a 980 H_z frequency and a 3 *volts* peak amplitude. The frequency deviation in the FM audio signal is *5 kH$_z$*. Again, the radio station transmits a different FM audio signal, where the modulating audio signal has a 1150 H_z frequency and a 4.5 *volts* peak amplitude. Compute the modulation index as well as the bandwidth of the FM audio signal using **Carson's rule** in both cases. Also, explain the motivation behind using Carson's rule in this scenario to calculate bandwidth. **(15)**

8. (a) Briefly discuss how amplitude shift-keying (ASK) and phase shift-keying (PSK) can be implemented to modulate a sinusoidal carrier signal with a digital baseband signal in digital signal modulation. **(5)**

(b) Consider a piece of binary data represented by a bit sequence of 101101001011. Draw the waveform of this binary data when the corresponding digital signal is generated using (i) Manchester encoding **(IEEE802.3 convention)** and (ii) differential Manchester encoding schemes. **(10)**

(c) The musical alphabet is a system of seven letters that represent the seven tones in a scale. This alphabet consists of seven English characters: *A, B, C, D, E, F* and *G*. Design a simple encoding scheme for converting these characters into bit strings. Also develop a line-coding scheme to generate digital signals from strings consisting of these musical alphabets. Show the waveform for a digital signal generated from a string *FEDE* using your proposed scheme. **(15)**