|  |
| --- |
| Digital Logic Design Prqactice Questions |

**MCQ’s**

1. The binary equivalent of 0.15510 is
   1. 0.000001
   2. 0.001011
   3. 0.001000
   4. **0.001001**
2. The binary equivalent of 7018 is
   1. **111000001**
   2. 111000000
   3. 110000001
   4. 111100001
3. The 8-bit signed binary equivalent of -2810 is
   1. 1111 0100
   2. **1110 0100**
   3. 1110 1100
   4. 1111 0100
4. The octal equivalent of signed 8-bit binary number 1111 11002 is
   1. 274
   2. 374
   3. **774**
   4. 674
5. Applying the Boolean algebra techniques to simplify the expression AB + A(B + C) + B(B + C), the resultant simplified expression is:
   1. **B + AC**
   2. AB + B
   3. B + BC
   4. C + AB
6. Using the DeMorgan’s theorem to simplify , the resultant simplified expression is:
7. Given the following Karnaugh map, the minimum Boolean POS expression is:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| CD\AB | 00 | 01 | 11 | 10 |
| 00 | 0 | 0 | 1 | 0 |
| 01 | 1 | X | X | 1 |
| 11 | X | 1 | 1 | X |
| 10 | 0 | 0 | 1 | 0 |

* 1. (A+C)(B+C)
  2. **(A+D)(B+D)**
  3. (B+A)(B+D)
  4. (A+D)(C+D)

1. A circuit has four inputs (A,B,C,D) representing a single decimal number ranging from 0 to 15 and an output (Y) that denotes whether the decimal number is divisible by 5. The SOP canonical of Y is:

**Mixed**

1. Convert
2. 12.4510 to Base 2

1100.011100112

1. Convert 12.416 to Base 10

18.2510

1. What range of hexadecimal numbers fit in 4 bytes field

0x00000000 to 0xFFFFFFFF

1. Using 2’s complement conversion on operands, perform the following operations. Show all working in binary 8-bit numbers.

-13-12

0000 1101 (+13)

1111 0011 (-13)

0000 1100 (+12)

1111 0100 (-12)

1111 0011

+ 1111 0100

11110 0111

1. Compute the following using two’s complement

1100 11112 – 1100 00002

1100 0000

0011 1111 (1’s)

0100 0000 (2’s)

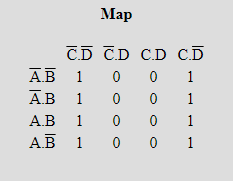
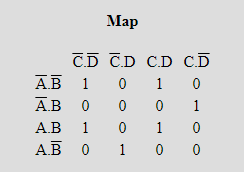
1100 1111

+ 0100 0000

10000 1111

1. A four-bit binary number {A,B,C,D}, where A is the most significant digit and D the least significant digit, appears on the input to a combinational logic circuit. Output X indicates whether the number is divisible by 2 without any remainder and output Y indicates if the number is divisible by 3 without remainder. Obtain the sum-of-products logic equations for X and Y, simplify using K-Maps and draw the logic diagram.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | A | B | C | D | **X** | **Y** |
| 0 | 0 | 0 | 0 | 0 | 1 | 1 |
| 1 | 0 | 0 | 0 | 1 | 0 | 0 |
| 2 | 0 | 0 | 1 | 0 | 1 | 0 |
| 3 | 0 | 0 | 1 | 1 | 0 | 1 |
| 4 | 0 | 1 | 0 | 0 | 1 | 0 |
| 5 | 0 | 1 | 0 | 1 | 0 | 0 |
| 6 | 0 | 1 | 1 | 0 | 1 | 1 |
| 7 | 0 | 1 | 1 | 1 | 0 | 0 |
| 8 | 1 | 0 | 0 | 0 | 1 | 0 |
| 9 | 1 | 0 | 0 | 1 | 0 | 1 |
| 10 | 1 | 0 | 1 | 0 | 1 | 0 |
| 11 | 1 | 0 | 1 | 1 | 0 | 0 |
| 12 | 1 | 1 | 0 | 0 | 1 | 1 |
| 13 | 1 | 1 | 0 | 1 | 0 | 0 |
| 14 | 1 | 1 | 1 | 0 | 1 | 0 |
| 15 | 1 | 1 | 1 | 1 | 0 | 1 |

1. 
2. D’
3. 
4. A'B'C'D' + A'B'CD + A'BCD' + AB'C'D + ABC'D' + ABCD

A diagram of a circuit

Description automatically generated