**LAB 4**

**Question #1:** Simplify the Product-Of-Sums Boolean expression below. Implement the resulting expression into the logic works.

http://sub.allaboutcircuits.com/images/14132.png

**K MAP:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **C’D’** | **C’D** | **CD** | **CD’** |
| **A’** |  | 1 |  |  |
| **A’** |  | 1 |  | 1 |
| **A** |  | 1 |  | 1 |
| **A** |  | 1 |  | 1 |

Out=πM(1,5,6,9,10,13,14)

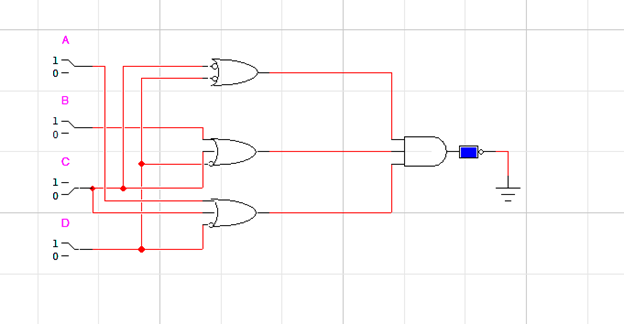
Out’=∑m(1,5,6,9,10,13,14)

**POS Expression:**

Out’/Out=(C’D+ACD’+BCD’)/(CD’+ACD’+BCD’)

**Out=(C+D’)(A’+C’+D)(B’+C’+D)**

**Circuit:**

****

**Question #2:** Find the product-of-sums for the following function:

F(A,B,C,D) = product.gif (95 bytes)M (3,5,7,8,10,11,12,13)

**K MAP:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **C’D’** | **C’D** | **CD** | **CD’** |
| **A’B’** |  |  | 1 |  |
| **A’B** |  | 1 | 1 |  |
| **AB** | 1 | 1 |  |  |
| **AB’** | 1 |  | 1 | 1 |

F’(A,B,C,D)=AC’D’+BC’D+A’CD+AB’C

**POS Expression:**

F(A,B,C,D)=(A’+C+D)(B’+C+D’)(A+C’+D’)(A’+B’+C)

**Question # 3** For the Boolean function do the following:

1. Find truth table

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **A** | **B** | **C** | **D** | **F** |
| 0 | 0 | 0 | 0 | **0** |
| 0 | 0 | 0 | 1 | **0** |
| 0 | 0 | 1 | 0 | **1** |
| 0 | 0 | 1 | 1 | **0** |
| 0 | 1 | 0 | 0 | **1** |
| 0 | 1 | 0 | 1 | **0** |
| 0 | 1 | 1 | 0 | **0** |
| 0 | 1 | 1 | 1 | **0** |
| 1 | 0 | 0 | 0 | **0** |
| 1 | 0 | 0 | 1 | **0** |
| 1 | 0 | 1 | 0 | **0** |
| 1 | 0 | 1 | 1 | **0** |
| 1 | 1 | 0 | 0 | **1** |
| 1 | 1 | 0 | 1 | **0** |
| 1 | 1 | 1 | 0 | **1** |
| 1 | 1 | 1 | 1 | **0** |

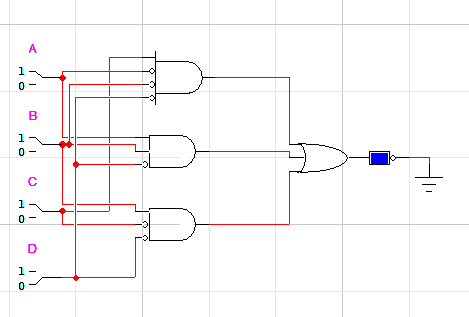
1. Find minimal SOP expression for Boolean function using K-map. Draw K-map in the space given below and implement it on logic works.

**K MAP:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **C’D’** | **C’D** | **CD** | **CD’** |
| **A’B’** |  |  |  | 1 |
| **A’B** | 1 |  |  |  |
| **AB** | 1 |  |  | 1 |
| **AB’** |  |  |  |  |

**F=BC’D’+ABD’+A’B’CD’**

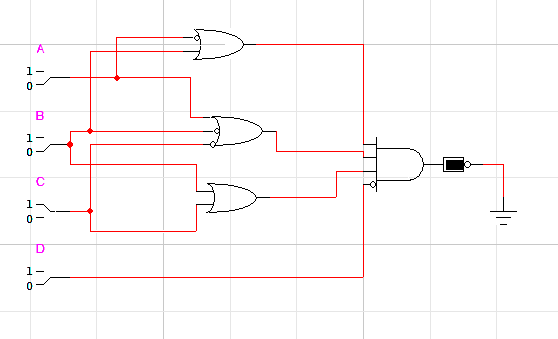
**Circuit:**

****

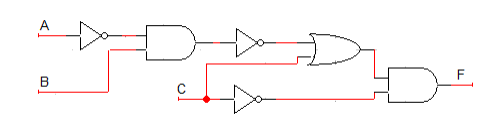
1. Find minimal POS expression for Boolean function using K-map. Draw K-map in the space given below and implement it on logic works.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **C’D’** | **C’D** | **CD** | **CD’** |
| **A’B’** | 0 | 0 | 0 |  |
| **A’B** |  | 0 | 0 | 0 |
| **AB** |  | 0 | 0 |  |
| **AB’** | 0 | 0 | 0 | 0 |

**Circuit:**

****

**Q4**. Find the output expression, truth-table. And also implement on Logic Works:



**Boolean expression:**

F = part1. part2

F = part. C’ part2=C’

F = (part1 + part2). C’

F = (part’ + C). C’ part2=C

F = [(part1.part2)’+C).C’

F = [(part1’.part2)’+C).C’

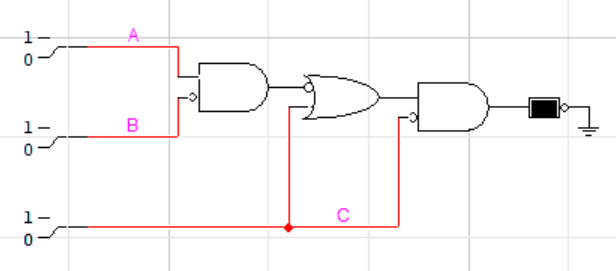
F = [(A’B)’+C)].C’ part1=A’ and part2=B

**F = [(A’B)’+C].C’**

**Truth Table:**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **A** | **B** | **C** | **A’** | **C’** | **A’B** | **(A’B)’** | **(A’B)’+C** | **F** |
| 0 | 0 | 0 | 1 | 1 | 0 | 1 | 1 | 1 |
| 0 | 0 | 1 | 1 | 0 | 0 | 1 | 1 | 0 |
| 0 | 1 | 0 | 1 | 1 | 1 | 0 | 0 | 0 |
| 0 | 1 | 1 | 1 | 0 | 1 | 0 | 1 | 0 |
| 1 | 0 | 0 | 0 | 1 | 0 | 1 | 1 | 1 |
| 1 | 0 | 1 | 0 | 0 | 0 | 1 | 1 | 0 |
| 1 | 1 | 0 | 0 | 1 | 0 | 1 | 1 | 1 |
| 1 | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 0 |

**Circuit:**

****

**Q5.** For the Boolean Functions **E** and **F**:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **X** | **Y** | **Z** | **E** | **F** |
| **0** | **0** | **0** | **0** | **1** |
| **0** | **0** | **1** | **1** | **0** |
| **0** | **1** | **0** | **1** | **1** |
| **0** | **1** | **1** | **0** | **0** |
| **1** | **0** | **0** | **1** | **1** |
| **1** | **0** | **1** | **0** | **0** |
| **1** | **1** | **0** | **1** | **0** |
| **1** | **1** | **1** | **0** | **1** |

1. List min-term and max-terms of each function.

**Min-terms of E:**

m1, m2, m4, m6

**Max-terms of E:**

M0, M3, M5, M7

**Min-terms of F:**

m0, m2, m4, m7

**Max-terms of F:**

M1, M3, M5, M6

1. Express E and F in sum-of-min-terms algebraic form.

**Expression of E:**

E = X’Y’Z+X’YZ’+XY’Z’+XYZ’

**Expression of F:**

F = X’Y’Z’+X’YZ’+XY’Z’+XYZ

1. Implement the **E** function on **Logic works.**

