

Assignment I

Simplifying Boolean expression using Kmap

Manideep Parusha - FWC22004

CONTENTS

| | | |
|------------|--------------------------------------|----------|
| I | Problem | 1 |
| II | Solution | 1 |
| II-A | Truth Table | 1 |
| II-B | K-map | 1 |
| II-C | Rules to simplify K-maps . . | 1 |
| II-D | Simplification | 2 |
| III | Verification & Conclusion | 2 |

I. PROBLEM

Reduce the following Boolean expression in the simplest form using Kmap. The Expression with Sum of Products (SOP) is as follows:

$$F(P, Q, R, S) = \sum(0, 1, 2, 3, 5, 6, 7, 10, 14, 15)$$

II. SOLUTION

A. Truth Table

Truth table for the SOP given:

| | P | Q | R | S | F(P,Q,R,S) |
|----|---|---|---|---|------------|
| 0 | 0 | 0 | 0 | 0 | 1 |
| 1 | 0 | 0 | 0 | 1 | 1 |
| 2 | 0 | 0 | 1 | 0 | 1 |
| 3 | 0 | 0 | 1 | 1 | 1 |
| 4 | 0 | 1 | 0 | 0 | 0 |
| 5 | 0 | 1 | 0 | 1 | 1 |
| 6 | 0 | 1 | 1 | 0 | 1 |
| 7 | 0 | 1 | 1 | 1 | 1 |
| 8 | 1 | 0 | 0 | 0 | 0 |
| 9 | 1 | 0 | 0 | 1 | 0 |
| 10 | 1 | 0 | 1 | 0 | 1 |
| 11 | 1 | 0 | 1 | 1 | 0 |
| 12 | 1 | 1 | 0 | 0 | 0 |
| 13 | 1 | 1 | 0 | 1 | 0 |
| 14 | 1 | 1 | 1 | 0 | 1 |
| 15 | 1 | 1 | 1 | 1 | 1 |

TABLE I
TRUTH TABLE FOR GIVEN BOOLEAN EXPRESSION

B. K-map

K-map for the above truth table:

| PQ \ RS | 00 | 01 | 11 | 10 |
|---------|----|----|----|----|
| 00 | 1 | 1 | 1 | 1 |
| 01 | 0 | 1 | 1 | 1 |
| 11 | 0 | 0 | 1 | 1 |
| 10 | 0 | 0 | 0 | 1 |

TABLE II
K-MAP FROM THE TRUTH TABLE

C. Rules to simplify K-maps

The Karnaugh map uses the following rules for the simplification of expressions by grouping together adjacent cells containing ones

- 1) Groups may not include any cell containing a zero
- 2) Groups may be horizontal or vertical, but not diagonal
- 3) Groups must contain 1, 2, 4, 8, or in general 2^n cells
- 4) Each group should be as large as possible.
- 5) Each cell containing a one must be in at least one group
- 6) Groups may overlap
- 7) Groups may wrap around the table. The leftmost cell in a row may be grouped with the rightmost cell and the top cell in a column may be grouped with the bottom cell

- 8) There should be as few groups as possible, as long as this does not contradict any of the previous rules

D. Simplification

| $PQ \backslash RS$ | | PQ | | | |
|--------------------|----|------|----|----|----|
| | | 00 | 01 | 11 | 10 |
| RS | 00 | 1 | 1 | 1 | 1 |
| | 01 | 0 | 1 | 1 | 1 |
| | 11 | 0 | 0 | 1 | 1 |
| | 10 | 0 | 0 | 0 | 1 |

TABLE III
GROUPED K-MAP

Simplified Boolean expression using Kmap without don't care will be:

$$F(P, Q, R, S) = P'Q' + P'S + RS' + QR$$

III. VERIFICATION & CONCLUSION

The simplified Boolean expression can be implemented using

```
wget https://raw.githubusercontent.com/parusamanideep/FWC/main/assignment1/src/main.cpp
```

The file in the above link is a C program to implement the logic that is simplified using the K-map.

The below link has the assembly program to implement the simplified logic using K-map.

```
wget https://raw.githubusercontent.com/parusamanideep/FWC/main/assignment1/asm/Assignment1.asm
```

P, Q, R, S are given as inputs to 2, 3, 4, 5 pins respectively from the 5V and GND lines.

The given Boolean expression is simplified and verified for functionality.