



# Karnaugh Map



### G V V Sharma\*

1

1

#### **CONTENTS**

## 1 Incrementing Decoder

## 2 Karnaugh Map

Abstract—This manual explains Karnaugh maps (Kmap) by finding the logic functions for the incrementing decoder.

#### 1 Incrementing Decoder

The incrementing decoder takes the numbers 0, 1, ..., 9 in binary as inputs and generates the consecutive number as output. The corresponding truth table is available in Table. 0.

Z	Y	X	W	D	C	В	A
0	0	0	0	0	0	0	1
0	0	0	1	0	0	1	0
0	0	1	0	0	0	1	1
0	0	1	1	0	1	0	0
0	1	0	0	0	1	0	1
0	1	0	1	0	1	1	0
0	1	1	0	0	1	1	1
0	1	1	1	1	0	0	0
1	0	0	0	1	0	0	1
1	0	0	1	0	0	0	0

TABLE 0

\*The author is with the Department of Electrical Engineering, Indian Institute of Technology, Hyderabad 502285 India e-mail: gadepall@iith.ac.in. All content in this manual is released under GNU GPL. Free and open source.

### 2 KARNAUGH MAP

Using Boolean logic, output A in Table 0 can be expressed in terms of the inputs W, X, Y, Z as

$$A = W'X'Y'Z' + W'XY'Z' + W'X'YZ' + W'XYZ' + W'X'Y'Z'$$
(1)

1) K-Map for A: The expression in (1) can be minimized using the K-map in Fig. 1. In Fig. 1, the *implicants* in boxes 0,2,4,6 result in W'Z'. The implicants in boxes 0,8 result in W'X'Y'. Thus, after minimization using Fig. 2, (1) can be expressed as

$$A = W'Z' + W'X'Y' \tag{2}$$

Using the fact that

$$X + X' = 1$$
$$XX' = 0,$$
 (3)

derive (2) from (1) algebraically.

### **Solution:**

$$A = W'X'Y'Z' + W'XY'Z' + W'X'YZ' + W'XYZ' + W'X'Y'Z'$$
(4)

$$A = W'X'Y' + W'X'Z' + W'Y'Z' + W'XYZ'$$
 (5)

$$A = W'X'Y' + W'Z'(X' + Y' + XY)$$
 (6)

$$A = W'X'Y' + W'Z'((XY)' + XY)$$
 (7)

$$A = W'Z' + W'X'Y' \tag{8}$$

2) K-Map for *B*: From Table 0, using boolean logic,

$$B = WX'Y'Z' + W'XY'Z' + WX'YZ' + W'XYZ'$$
(9)

Show that (11) can be reduced to

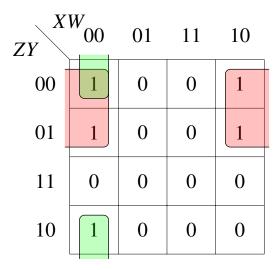


Fig. 1: K-map for A.

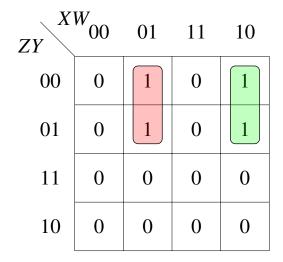


Fig. 2: K-map for *B*.

$$B = WX'Z' + W'XZ' \tag{10}$$

using Fig. 2.

3) Derive (10) from (11) algebraically using (3). **Solution:** 

$$B = WX'Y'Z' + W'XY'Z' + WX'YZ' + W'XYZ'$$
(11)

$$B = WX'Z' + W'XZ' \tag{12}$$

4) K-Map for C: From Table 0, using boolean

logic,

$$C = WXY'Z' + W'X'YZ' + WX'YZ' + W'XYZ'$$
(13)

Show that (13) can be reduced to

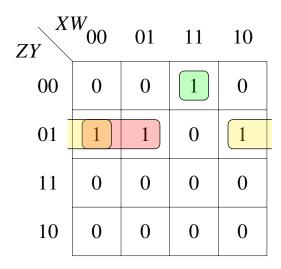


Fig. 4: K-map for *C*.

$$C = WXY'Z' + X'YZ' + W'YZ'$$
 (14)

using Fig. 4.

5) Derive (14) from (13) algebraically using (3). **Solution:** 

$$C = WXY'Z' + W'X'YZ' + WX'YZ' + W'XYZ'$$
(15)

$$C = WXY'Z' + X'YZ' + W'YZ' \tag{16}$$

6) K-Map for *D*: From Table 0, using boolean logic,

$$D = WXYZ' + W'X'Y'Z \tag{17}$$

- 7) Minimize (17) using Fig. 6.
- 8) Download the code in

and modify it using the K-Map equations for A,B,C and D. Execute and verify.

9) Display Decoder: Table 9 is the truth table for the display decoder. Use K-maps to obtain the minimized expressions for a, b, c, d, e, f, g in

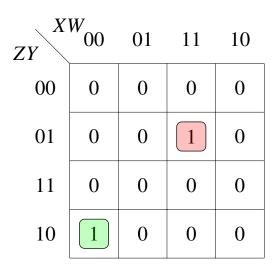


Fig. 6: K-map for *D*.

terms of A, B, C, D without don't care conditions.

D	С	В	A	a	b	c	d	e	f	g	Decimal
0	0	0	0	0	0	0	0	0	0	1	0
0	0	0	1	1	0	0	1	1	1	1	1
0	0	1	0	0	0	1	0	0	1	0	2
0	0	1	1	0	0	0	0	1	1	0	3
0	1	0	0	0	1	0	0	1	0	0	4
0	1	0	1	1	0	0	1	1	0	0	5
0	1	1	0	0	1	0	0	1	0	0	6
0	1	1	1	0	0	0	1	1	1	1	7
1	0	0	0	0	0	0	0	0	0	0	8
1	0	0	1	0	0	0	0	1	0	0	9

TABLE 9: Truth table for display decoder.

### **Solution:**

Without DON'T CARE:

from Fig. 9

$$a = D'C'B'A + D'CB'A' \tag{18}$$

from Fig. 9

$$b = D'CB'A + D'CBA' \tag{19}$$

from Fig. 9

$$c = D'C'BA' \tag{20}$$

from Fig. 9

$$d = D'CB'A' + D'CBA + C'B'A$$
 (21)

from Fig. 9

$$e = D'A + C'B'A + D'CB'$$
 (22)

from Fig. 9

$$f = D'BA + D'C'A + D'C'B$$
 (23)

from Fig. 9

$$g = D'C'B' + D'CBA \tag{24}$$

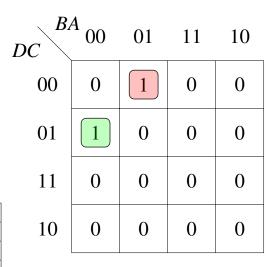


Fig. 9: K-map for a.

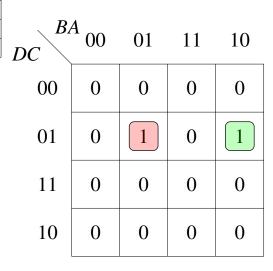
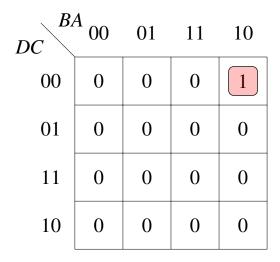


Fig. 9: K-map for b.



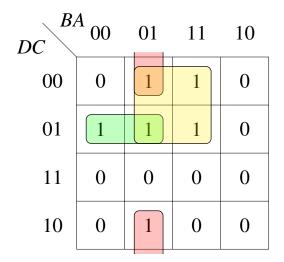


Fig. 9: K-map for c.

Fig. 9: K-map for e.

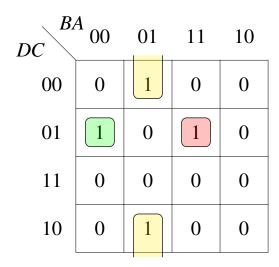


Fig. 9: K-map for d.

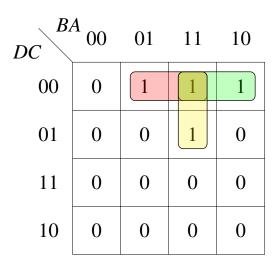


Fig. 9: K-map for f.

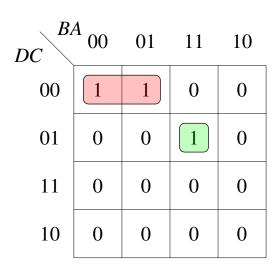


Fig. 9: K-map for g.