North South University

Department of Electrical & Computer Engineering

PROJECT REPORT

Course Code: CSE231L.5

Course Title: Digital Logic Design

Faculty: Dr. S.M. Mahfuz Alam

Project Title:

Combinational and Sequential Seven-Segment Display Circuits

Date of Submission: 18 / 11 / 23

Section: 5

Group Number: 1

Submitted To: Jannatul Ferdaous

Submitted By:

Serial No.	Student Name	ID
41	Faiyad Hossain	2222258642
21	Kazi Estiaq Jacy	2211448042
13	Sohan Mondal	2132443642
3	Riven Khan	2011908042

Score

Table of Contents

Introduction

The project report serves as a documentation for the logic, design and implementation of the various forms of a seven-segment display circuit. The circuit is expected to generate the sequence 2S-10321DLDD, with each character corresponding to one of 12 different inputs or states.
The project is divided into two parts based on the fundamental circuit logic types: combinational logic, and sequential logic. For each part, numerous alternative designs of the circuit are presented. From among them only one is chosen for the hardware implementation, whereas all of them are implemented using software such as Logisim.
Therefore, as an implication, the purpose of this project is to test, assess and apply the culmination of one's skills and knowledge of digital logic design acquired through the course and lab works.

PHASE 1: COMBINATIONAL PART

Truth Table

Ref.	Inputs				Outputs						
	w	*x*	*y*	*z*	*A*	*B*	*C*	*D*	*E*	*F*	*G*
0	0	0	0	0	1	1	0	1	1	0	1
1	0	0	0	1	1	0	1	1	0	1	1
2	0	0	1	0	0	0	0	0	0	0	1
3	0	0	1	1	0	1	1	0	0	0	0
4	0	1	0	0	1	1	1	1	1	1	0
5	0	1	0	1	1	1	1	1	0	0	1
6	0	1	1	0	1	1	0	1	1	0	1
7	0	1	1	1	0	1	1	0	0	0	0
8	1	0	0	0	0	1	1	1	1	0	1
9	1	0	0	1	0	0	0	1	1	1	0
10	1	0	1	0	0	1	1	1	1	0	1
11	1	0	1	1	0	1	1	1	1	0	1
12	1	1	0	0	x	x	x	x	x	x	x
13	1	1	0	1	x	x	x	x	x	x	x
14	1	1	1	0	x	x	x	x	x	x	x
15	1	1	1	1	x	x	x	x	x	x	x

1st Canonical Forms

$$A\ (w, x, y, z) = \sum (0, 1, 4, 5, 6)$$
$$= \bar{w}\bar{x}\bar{y}\bar{z} + \bar{w}\bar{x}\bar{y}z + \bar{w}x\bar{y}\bar{z} + \bar{w}x\bar{y}z + \bar{w}xy\bar{z}$$

$$B\ (w, x, y, z) = \sum (0, 3, 4, 5, 6, 7, 8, 10, 11)$$
$$= \bar{w}\bar{x}\bar{y}\bar{z} + \bar{w}\bar{x}yz + \bar{w}x\bar{y}\bar{z} + \bar{w}x\bar{y}z + \bar{w}xy\bar{z} + \bar{w}xyz + w\bar{x}\bar{y}\bar{z} + w\bar{x}y\bar{z} + w\bar{x}yz$$

$$C\ (w, x, y, z) = \sum (1, 3, 4, 5, 7, 8, 10, 11)$$
$$= \bar{w}\bar{x}\bar{y}z + \bar{w}\bar{x}yz + \bar{w}x\bar{y}\bar{z} + \bar{w}x\bar{y}z + \bar{w}xyz + w\bar{x}\bar{y}\bar{z} + w\bar{y}x\bar{z} + w\bar{x}yz$$

$$D\ (w, x, y, z) = \sum (0, 1, 4, 5, 6, 8, 9, 10, 11)$$
$$= \bar{w}\bar{x}\bar{y}\bar{z} + \bar{w}\bar{x}\bar{y}z + \bar{w}x\bar{y}\bar{z} + \bar{w}x\bar{y}z + \bar{w}xy\bar{z} + w\bar{x}\bar{y}\bar{z} + w\bar{x}\bar{y}z + w\bar{x}y\bar{z} + w\bar{x}yz$$

$$E\ (w, x, y, z) = \sum (0, 4, 6, 8, 9, 10, 11)$$
$$= \bar{w}\bar{x}\bar{y}\bar{z} + \bar{w}x\bar{y}\bar{z} + \bar{w}xy\bar{z} + w\bar{x}\bar{y}\bar{z} + w\bar{x}\bar{y}z + w\bar{x}y\bar{z} + w\bar{x}yz$$

$$F\ (w, x, y, z) = \sum (1, 4, 9) = \bar{w}\bar{x}\bar{y}z + \bar{w}x\bar{y}\bar{z} + w\bar{x}\bar{y}z$$

$$G\ (w, x, y, z) = \sum (0, 1, 2, 5, 6, 8, 10, 11)$$
$$= \bar{w}\bar{x}\bar{y}\bar{z} + \bar{w}\bar{x}\bar{y}z + \bar{w}\bar{x}y\bar{z} + \bar{w}x\bar{y}z + \bar{w}xy\bar{z} + w\bar{x}\bar{y}\bar{z} + w\bar{x}y\bar{z} + w\bar{x}yz$$

2nd Canonical Forms

$$A\ (w, x, y, z) = \prod (2, 3, 7, 8, 9, 10, 11)$$
$$= (w + x + \bar{y} + z)(w + x + \bar{y} + \bar{z})(w + \bar{x} + \bar{y} + \bar{z})(\bar{w} + x + y + z)(\bar{w} + x + y + \bar{z})(\bar{w} + x + \bar{y} + z)(\bar{w} + x + \bar{y} + \bar{z})$$

$$B\ (w, x, y, z) = \prod (1, 2, 9)$$
$$= (w + x + y + \bar{z})(w + x + \bar{y} + z)(\bar{w} + x + y + \bar{z})$$

$$C\ (w, x, y, z) = \prod (0, 2, 6, 9)$$
$$= (w + x + y + z)(w + x + \bar{y} + z)(w + \bar{x} + \bar{y} + z)(\bar{w} + x + y + \bar{z})$$

$$D\ (w, x, y, z) = \prod (2, 3, 7)$$
$$= (w + x + \bar{y} + z)(w + x + \bar{y} + \bar{z})(w + \bar{x} + \bar{y} + \bar{z})$$

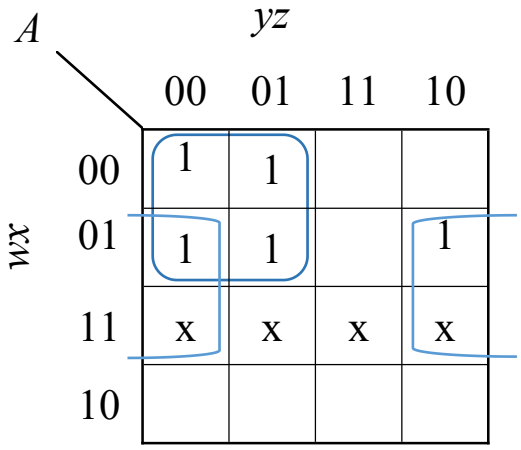
$$E\ (w, x, y, z) = \prod (1, 2, 3, 5, 7)$$
$$= (w + x + y + \bar{z})(w + x + \bar{y} + z)(w + x + \bar{y} + \bar{z})(w + \bar{x} + y + \bar{z})(w + \bar{x} + \bar{y} + \bar{z})$$

$$F\ (w, x, y, z) = \prod (0, 2, 3, 5, 6, 7, 8, 10, 11)$$
$$= (w + x + y + z)(w + x + \bar{y} + z)(w + x + \bar{y} + \bar{z})(w + \bar{x} + y + \bar{z})(w + \bar{x} + \bar{y} + z)(w + \bar{x} + \bar{y} + \bar{z})(\bar{w} + x + y + z)(\bar{w} + x + \bar{y} + z)(\bar{w} + x + \bar{y} + \bar{z})$$

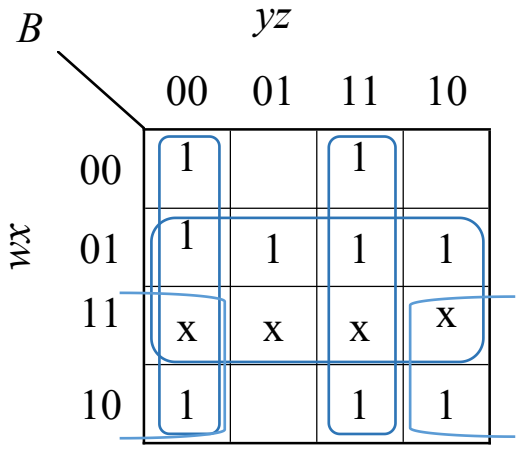
$$G\ (w, x, y, z) = \prod (3, 4, 7, 9)$$
$$= (w + x + \bar{y} + \bar{z})(w + \bar{x} + y + z)(w + \bar{x} + \bar{y} + \bar{z})(\bar{w} + x + y + \bar{z})$$

Sum of Product Forms (SOP)

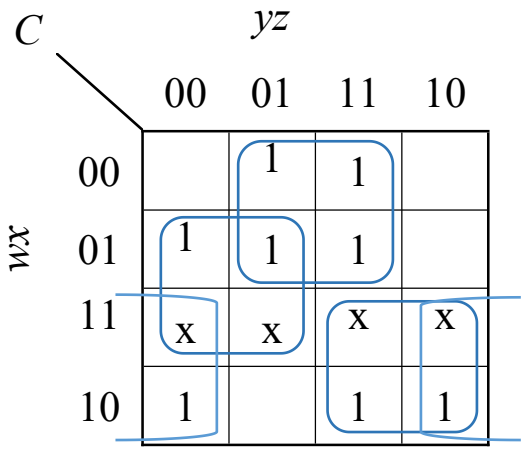
Boolean Equations with K-Maps



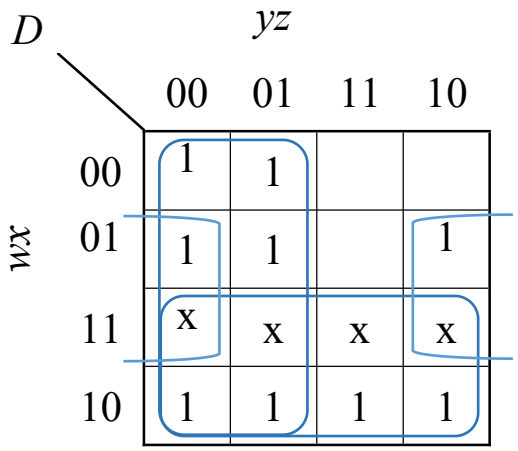
$A\ (w, x, y, z) = \bar{w}\bar{y} + x\bar{z}$



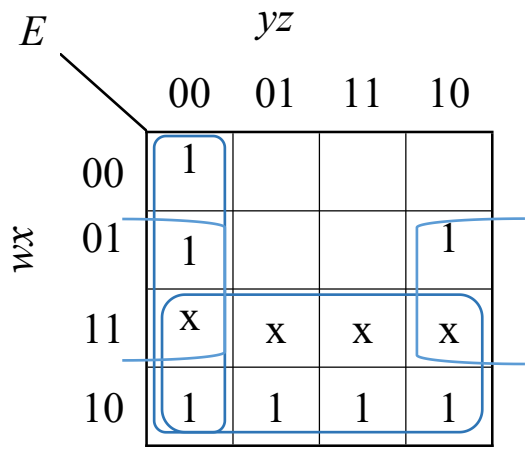
$B\ (w, x, y, z) = x + w\bar{z} + yz + \bar{y}\bar{z}$



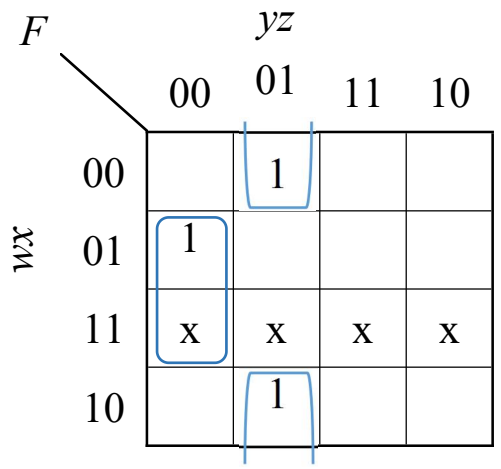
$C\ (w, x, y, z) = wy + w\bar{z} + \bar{w}z + x\bar{y}$



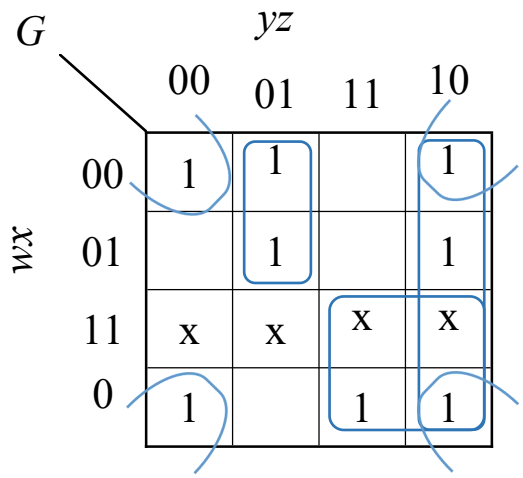
$D\ (w, x, y, z) = w + \bar{y} + x\bar{z}$



$E\ (w, x, y, z) = w + x\bar{z} + \bar{y}\bar{z}$

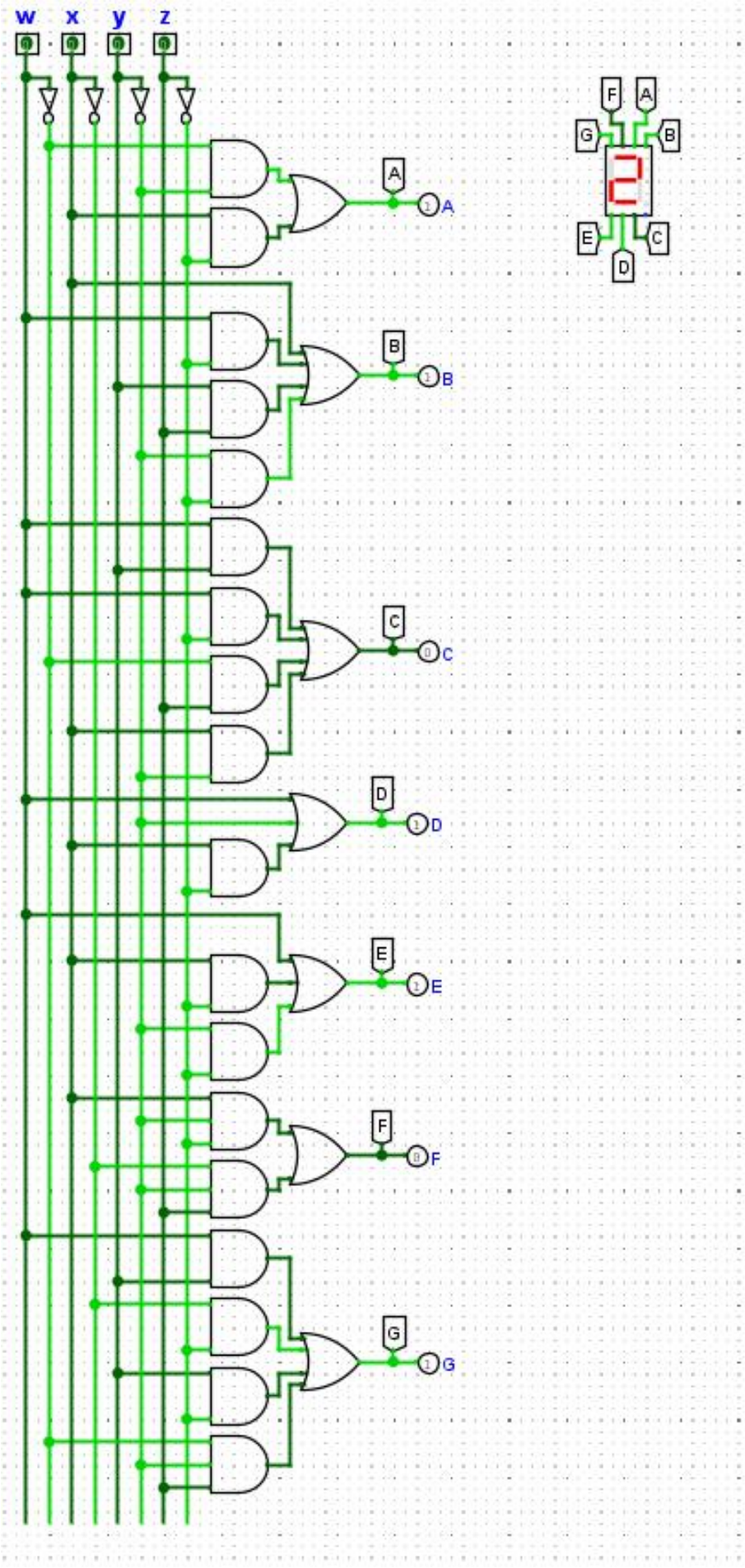


$F\ (w, x, y, z) = x\bar{y}\bar{z} + \bar{x}\bar{y}z$



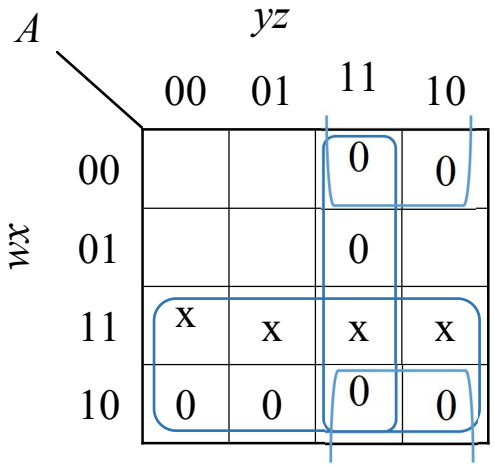
$G\ (w, x, y, z) = wy + \bar{x}\bar{z} + y\bar{z} + \bar{w}\bar{y}z$

Circuit Diagram

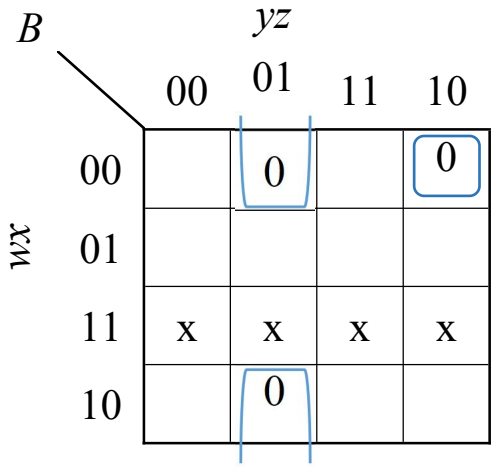


Product of Sum Forms (POS)

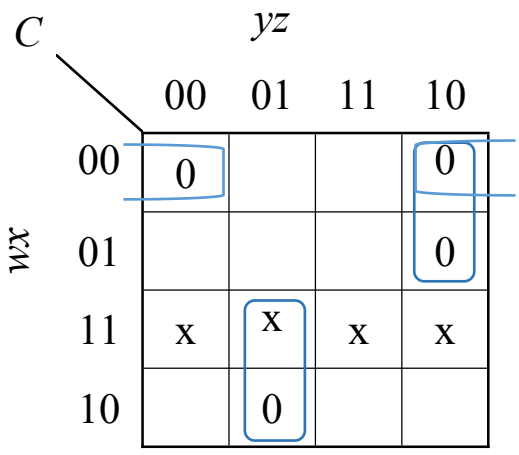
Boolean Equations with K-Maps



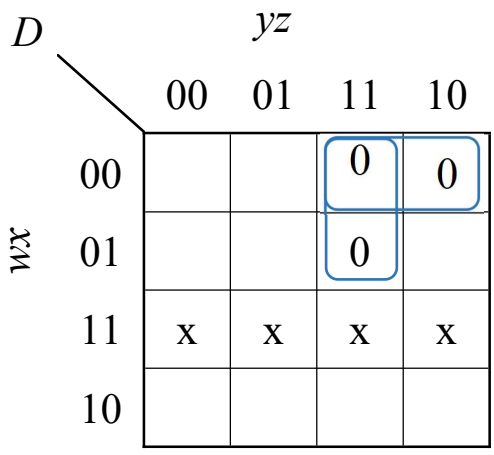
$A\ (w, x, y, z) = \bar{w}(x + \bar{y})(\bar{y} + \bar{z})$



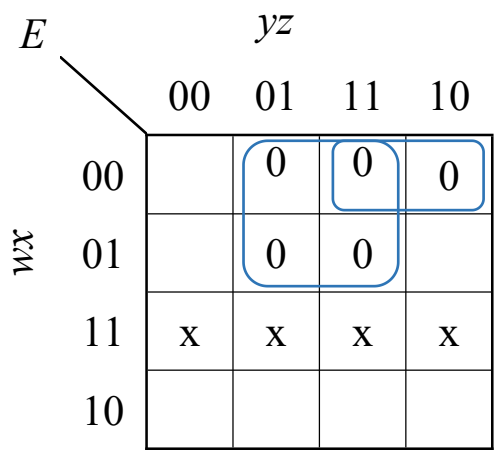
$B\ (w, x, y, z) = (x + y + \bar{z})$
$(w + x + \bar{y} + z)$



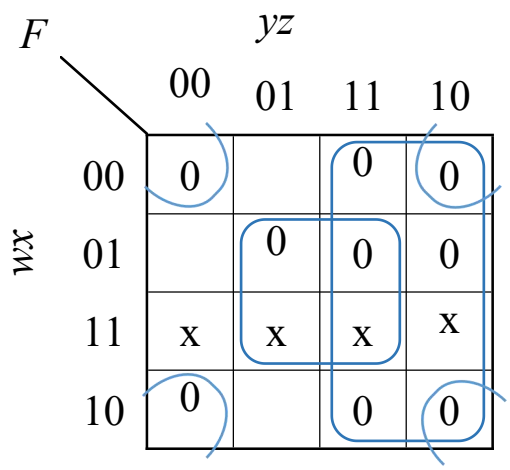
$C\ (w, x, y, z) = (w + x + z)$
$(w + \bar{y} + z)(\bar{w} + y + \bar{z})$



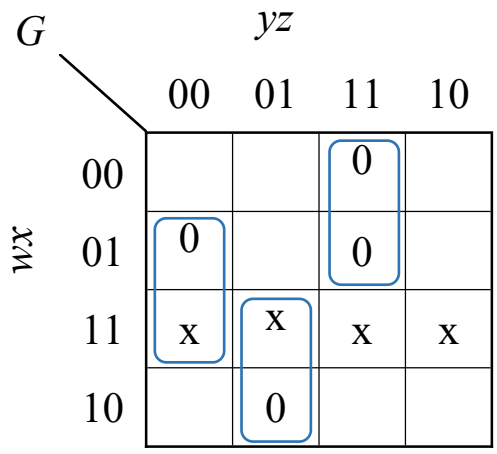
$D\ (w, x, y, z) = (w + x + \bar{y})$
$(w + \bar{y} + \bar{z})$



$E\ (w, x, y, z) = (w + \bar{z})(w + x + \bar{y})$

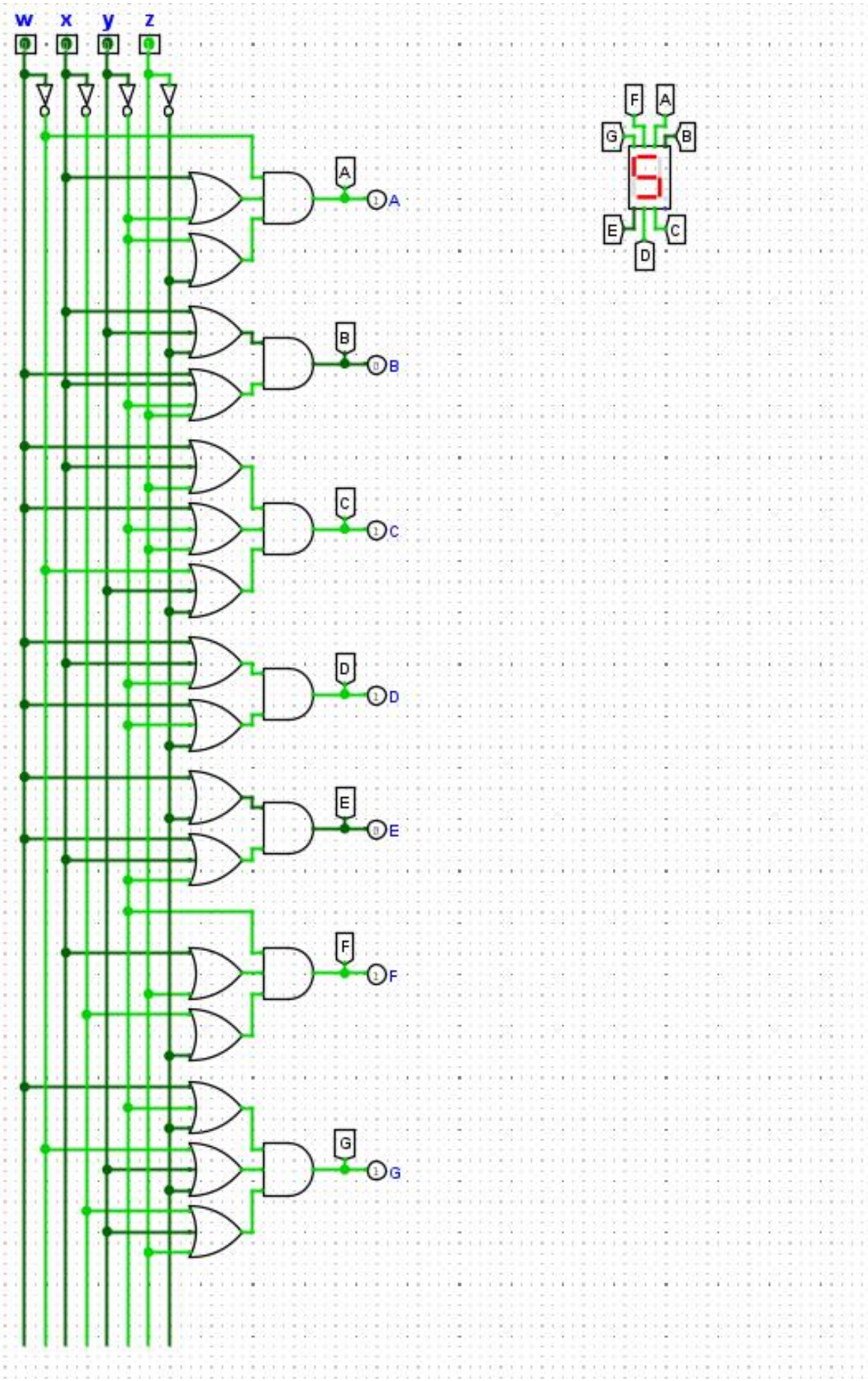


$F\ (w, x, y, z) = \bar{y}(x + z)(\bar{x} + \bar{z})$



$G\ (w, x, y, z) = (w + \bar{y} + \bar{z})(\bar{w} + y + \bar{z})(\bar{x} + y + z)$

Circuit Diagram



NAND Gate Implementation

Explanation

One way to figure out an implementation of the seven-segment display circuit using only NAND gates is by transformation each output function in SOP form so that the only operators involved are the NOT and AND operators (which are combined to represent the NAND operator). Then, on the basis of these alternative expressions the circuit is built.

Boolean Equations

$$A\ (w, x, y, z) = \bar{w}\bar{y} + x\bar{z} = \overline{\overline{\bar{w}\bar{y} + x\bar{z}}} = \overline{\overline{\bar{w}\bar{y}}\ \overline{x\bar{z}}}$$

$$B\ (w, x, y, z) = x + w\bar{z} + yz + \bar{y}\bar{z} = \overline{\overline{x + w\bar{z} + yz + \bar{y}\bar{z}}} = \overline{\bar{x}\ \overline{w\bar{z}}\ \overline{yz}\ \overline{\bar{y}\bar{z}}}$$

$$C\ (w, x, y, z) = wy + w\bar{z} + \bar{w}z + x\bar{y} = \overline{\overline{wy + w\bar{z} + \bar{w}z + x\bar{y}}} = \overline{\overline{wy}\ \overline{w\bar{z}}\ \overline{\bar{w}z}\ \overline{x\bar{y}}}$$

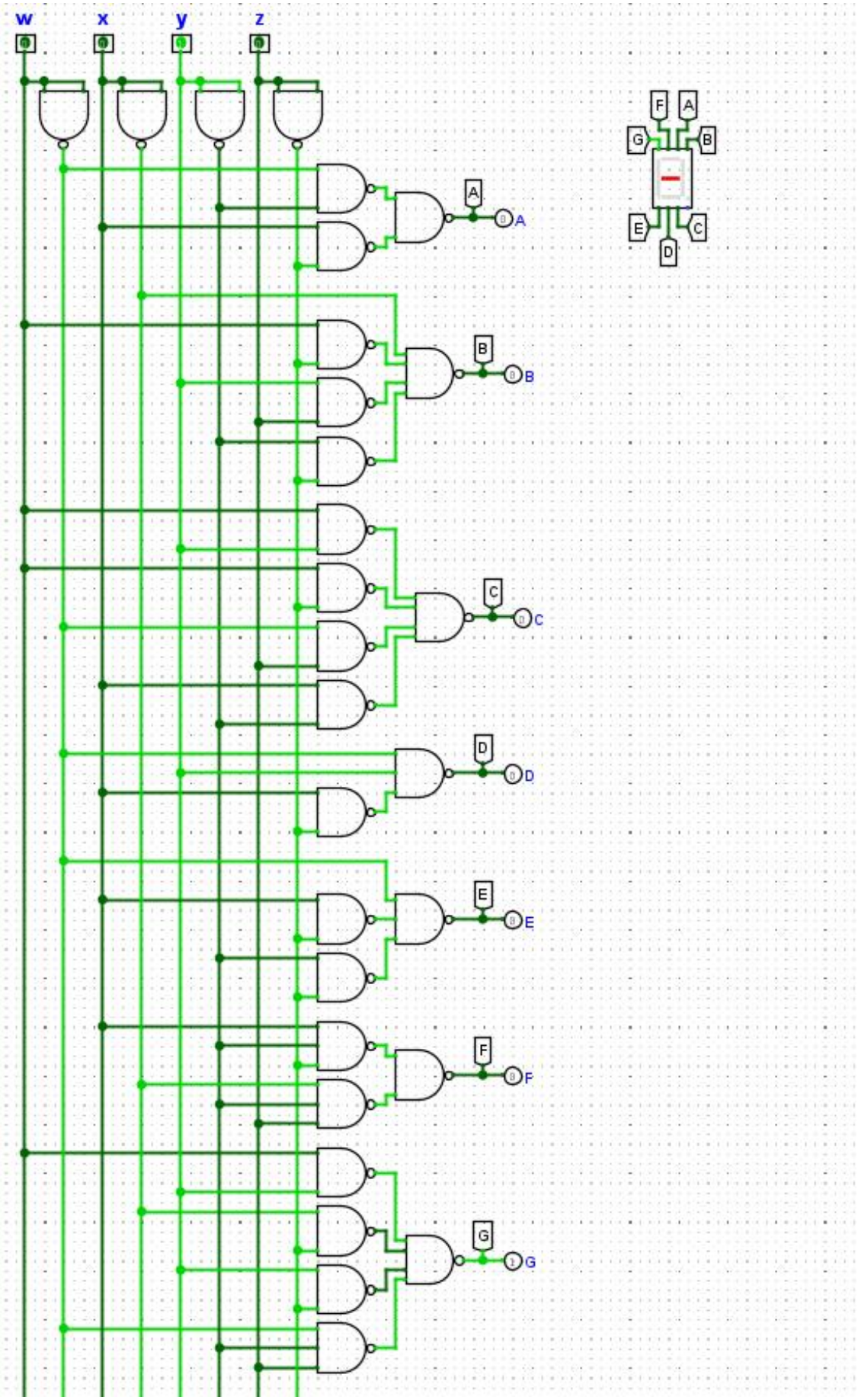
$$D\ (w, x, y, z) = w + \bar{y} + x\bar{z} = \overline{\overline{w + \bar{y} + x\bar{z}}} = \overline{\bar{w}\ y\ \overline{x\bar{z}}}$$

$$E\ (w, x, y, z) = w + x\bar{z} + \bar{y}\bar{z} = \overline{\overline{w + x\bar{z} + \bar{y}\bar{z}}} = \overline{\bar{w}\ \overline{x\bar{z}}\ \overline{\bar{y}\bar{z}}}$$

$$F\ (w, x, y, z) = x\bar{y}\bar{z} + \bar{x}\bar{y}z = \overline{\overline{x\bar{y}\bar{z} + \bar{x}\bar{y}z}} = \overline{\overline{x\bar{y}\bar{z}}\ \overline{\bar{x}\bar{y}z}}$$

$$G\ (w, x, y, z) = wy + \bar{x}\bar{z} + y\bar{z} + \bar{w}\bar{y}z = \overline{\overline{wy + \bar{x}\bar{z} + y\bar{z} + \bar{w}\bar{y}z}} = \overline{\overline{wy}\ \overline{\bar{x}\bar{z}}\ \overline{y\bar{z}}\ \overline{\bar{w}\bar{y}z}}$$

Circuit Diagram



NOR Gate Implementation

Explanation

One way to figure out an implementation of the seven-segment display circuit using only NOR gates is by transformation each output function in POS form so that the only operators involved are the NOT and OR operators (which are combined to represent the NOR operator). Then, on the basis of these alternative expressions the circuit is built.

Boolean Expressions

$$A\ (w, x, y, z) = \overline{w}(x + \overline{y})(\overline{y} + \overline{z}) = \overline{\overline{w}(x + \overline{y})(\overline{y} + \overline{z})} = \overline{w + \overline{(x + \overline{y})} + \overline{(\overline{y} + \overline{z})}}$$

$$\begin{aligned} B\ (w, x, y, z) &= (x + y + \overline{z})(w + x + \overline{y} + z) = \overline{\overline{(x + y + \overline{z})(w + x + \overline{y} + z)}} \\ &= \overline{(x + y + \overline{z})} + \overline{(w + x + \overline{y} + z)} \end{aligned}$$

$$\begin{aligned} C\ (w, x, y, z) &= (w + x + z)(w + \overline{y} + z)(\overline{w} + y + \overline{z}) \\ &= \overline{\overline{(w + x + z)(w + \overline{y} + z)(\overline{w} + y + \overline{z})}} \\ &= \overline{(w + x + z)} + \overline{(w + \overline{y} + z)} + \overline{(\overline{w} + y + \overline{z})} \end{aligned}$$

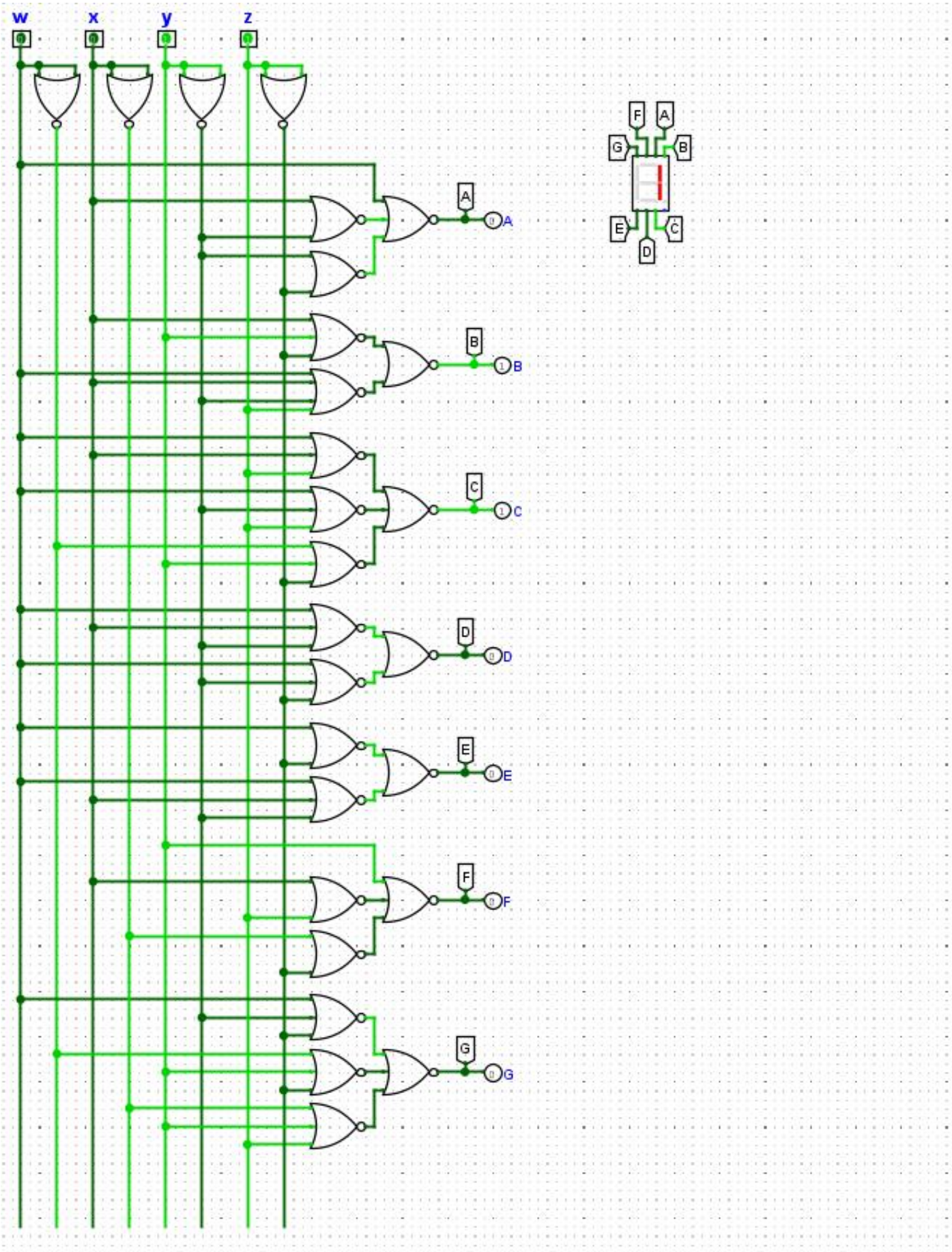
$$\begin{aligned} D\ (w, x, y, z) &= (w + x + \overline{y})(w + \overline{y} + \overline{z}) = \overline{\overline{(w + x + \overline{y})(w + \overline{y} + \overline{z})}} \\ &= \overline{(w + x + \overline{y})} + \overline{(w + \overline{y} + \overline{z})} \end{aligned}$$

$$\begin{aligned} E\ (w, x, y, z) &= (w + \overline{z})(w + x + \overline{y}) = \overline{\overline{(w + \overline{z})(w + x + \overline{y})}} \\ &= \overline{(w + \overline{z})} + \overline{(w + x + \overline{y})} \end{aligned}$$

$$F\ (w, x, y, z) = \overline{y}(x + z)(\overline{x} + \overline{z}) = \overline{\overline{y}(x + z)(\overline{x} + \overline{z})} = \overline{y + \overline{(x + z)} + \overline{(\overline{x} + \overline{z})}}$$

$$\begin{aligned} G\ (w, x, y, z) &= (w + \overline{y} + \overline{z})(\overline{w} + y + \overline{z})(\overline{x} + y + z) \\ &= \overline{\overline{(w + \overline{y} + \overline{z})(\overline{w} + y + \overline{z})(\overline{x} + y + z)}} \\ &= \overline{(w + \overline{y} + \overline{z})} + \overline{(\overline{w} + y + \overline{z})} + \overline{(\overline{x} + y + z)} \end{aligned}$$

Circuit Diagram

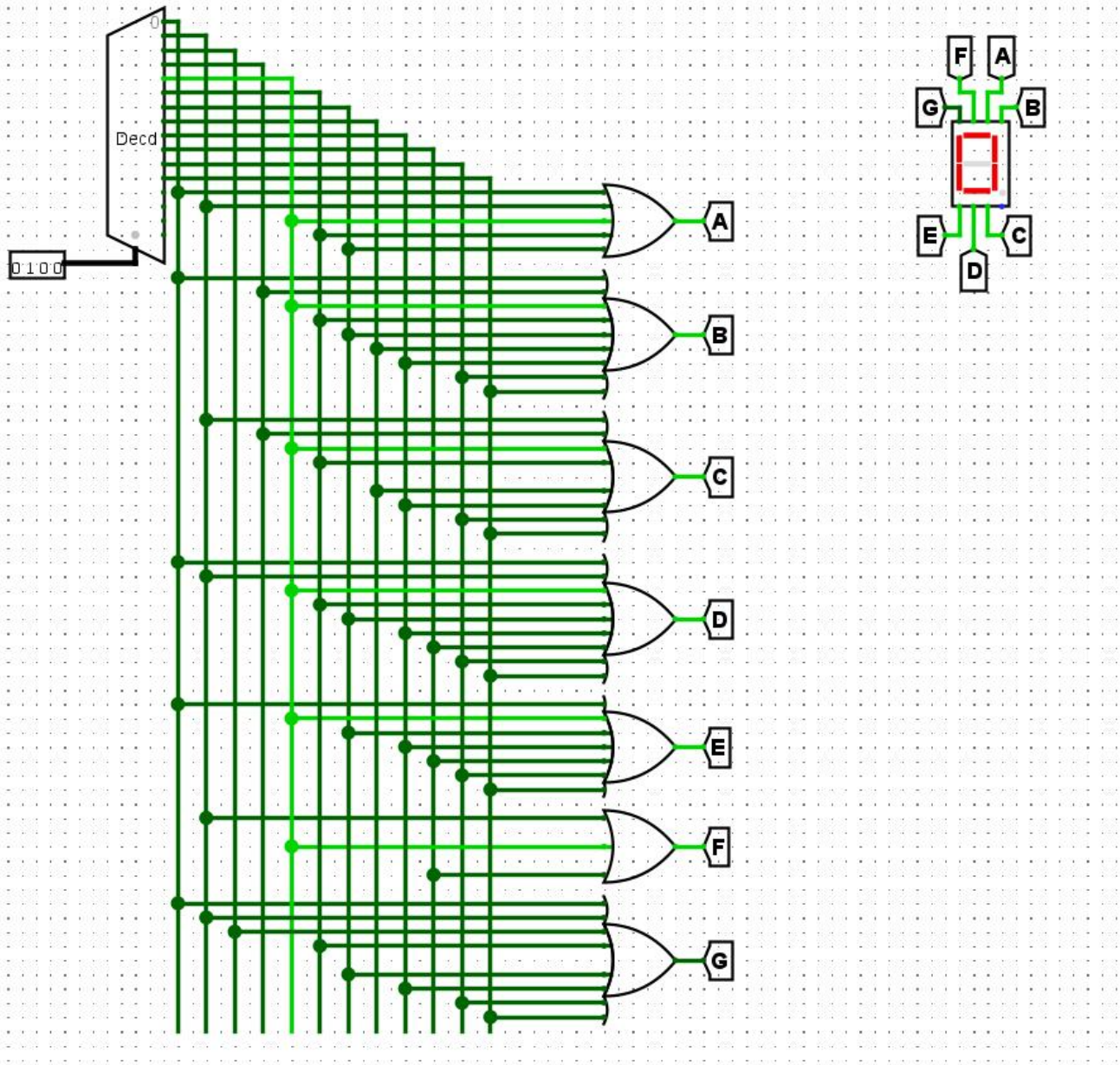


Decoder Implementation

Explanation

In constructing a seven-segment display circuit with an active-high 4-to-16 line decoder, the output activated by a specific select input combination serves the purpose of illuminating the segments essential for forming the related character. Each segment is turned on for more than one binary code, so the input to it is controlled by a multi-input OR gate that connects all the decoder outputs activated by the corresponding binary input combinations. Note that the decoder outputs for inputs 1100 to 1111 are not driven to any of the segments.

Circuit Diagram

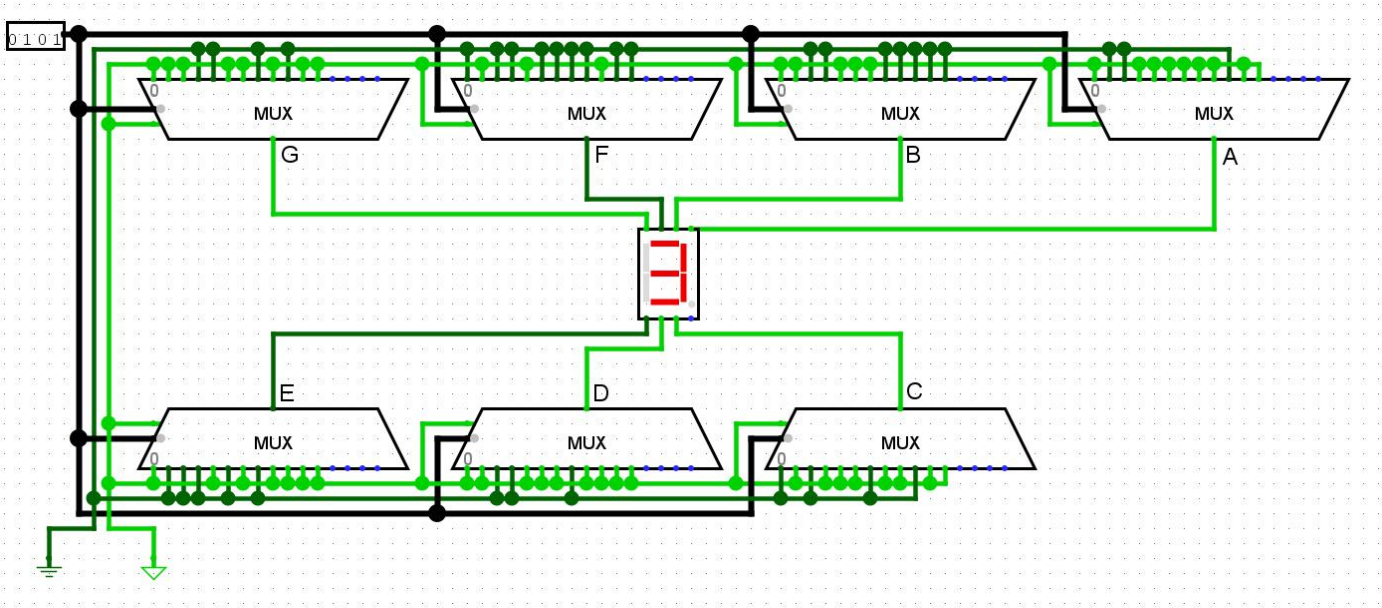


MUX Implementation

Explanation

Linked to each segment of the display is the output of a 16-to-1 multiplexer. Each data input of the MUX is connected to either the power supply (HIGH) or ground (LOW), depending on whether or not it activates the segment due to a certain combination of selection bits. This means that the data inputs directly evaluates to the output of the function, which is consistent with the truth table. However, as a consequence of this fact, all the multiplexers share the characteristics that the data inputs from I_{12} to I_{15} do not provide any relevant information, representing the don't care conditions.

Circuit Diagram



Hardware Implementation

Choice of Implementation

The choice of implementation depends on four factors to varying degrees: cost, complexity, and availability of materials. The MUX and decoder implementations were ignored, since the required ICs are difficult to come by. Moreover, using multiplexers and decoders with lesser input/output lines vastly complicates the circuit design by greatly increasing the number of connections. Among the rest of the implementations, the SOP form and the NAND gate ones stand out as optimal due to the fact that they require the least number of gates, thus minimizing cost and complexity of the circuit. However, the former of the two was chosen due to following reasons: (a) multi-input AND and OR gates can be more easily decomposed into a collection of interconnected two-input gates (b) as a result, repeated terms between equations (e.g. $x\bar{z}$ in D and E) would only need to be implemented once and shared among the corresponding circuit segments. Thus, the circuit can be further simplified using only ICs containing gates with no more than two inputs, reducing costs.
The materials for hardware implementation is listed in the following section.

Cost of Implementation

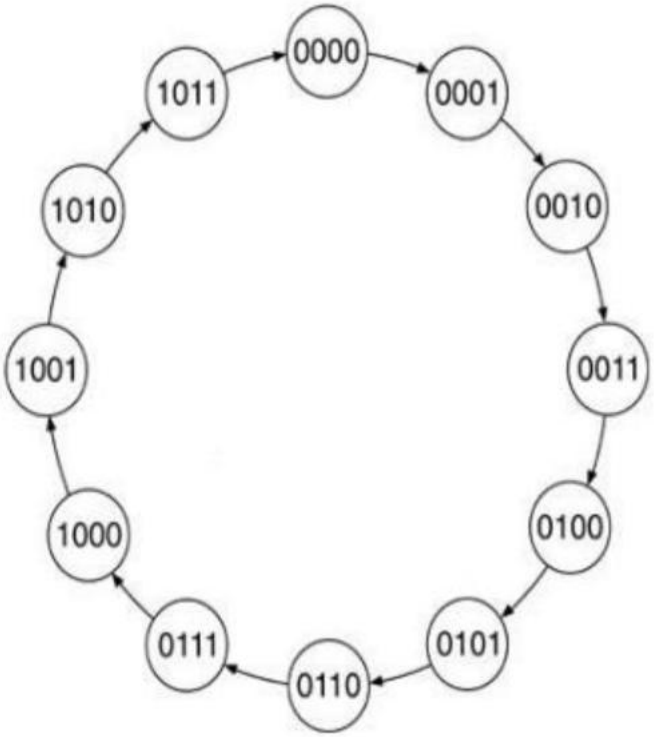
Index	Name	Cost Per Unit	Quantity	Total Cost
1	Breadboard (Large)	140	3	420
2	Male-to-Male Connectors x40	120	3	360
3	9V Battery	80	1	80
4	Battery Connector	15	1	15
5	5V Voltage Regulator	20	1	20
6	Seven Segment Display (Common Cathode)	15	1	15
7	Push Down Switch	15	4	60
8	IC 7404 Hex Inverters NOT Gates	30	1	30
9	IC 7408 Quad Two-Input AND Gates	30	4	120
10	IC 7432 Quad Two-Input OR Gates	30	4	120
CUMULATIVE TOTAL				1240

PHASE 2: SEQUENTIAL PART

Preface

As a sequential circuit, the seven-segment display should display each character in sequence between constant time intervals. This can be achieved in many ways. In this project, we have decided to design a synchronous MOD 12 up-counter, using the JK, D, and T flip-flops, that counts from 0000 to 1011 incrementally with every clock pulse. This means, the counter circuit requires 4 flip-flops, representing the bits of the binary code. The output of each flip-flop is then directed to the corresponding input of the SOP form of the combinational circuit connected to the seven-segment display. In this way, the characters in the sequence are displayed one-by-one as long as the counter is powered or the clock pulses are delivered.

State Diagram

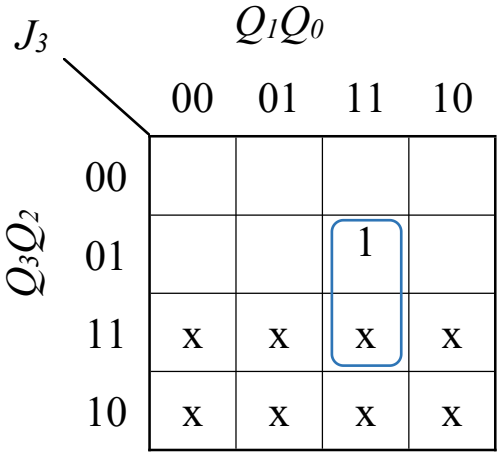


JK Flip-Flop Implementation

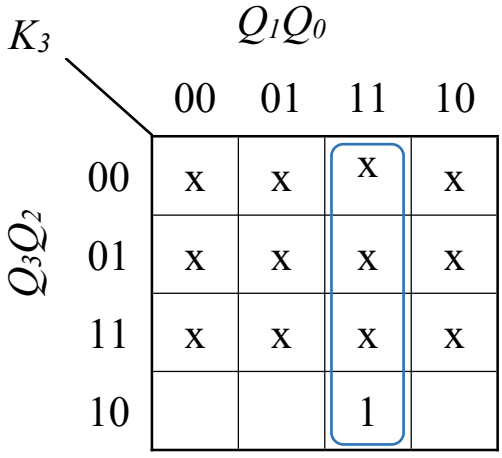
State Table

Present State				Next State				Flip Flop Inputs							
Q_3	Q_2	Q_1	Q_0	Q_3	Q_2	Q_1	Q_0	J_3	K_3	J_2	K_2	J_1	K_1	J_0	K_0
0	0	0	0	0	0	0	1	0	x	0	x	0	x	1	x
0	0	0	1	0	0	1	0	0	x	0	x	1	x	x	1
0	0	1	0	0	0	1	1	0	x	0	x	x	0	1	x
0	0	1	1	0	1	0	0	0	x	1	x	x	1	x	1
0	1	0	0	0	1	0	1	0	x	x	0	0	x	1	x
0	1	0	1	0	1	1	0	0	x	x	0	1	x	x	1
0	1	1	0	0	1	1	1	0	x	x	0	x	0	1	x
0	1	1	1	1	0	0	0	1	x	x	1	x	1	x	1
1	0	0	0	1	0	0	1	x	0	0	x	0	x	1	x
1	0	0	1	1	0	1	0	x	0	0	x	1	x	x	1
1	0	1	0	1	0	1	1	x	0	0	x	x	0	1	x
1	0	1	1	0	0	0	0	x	1	0	x	x	1	x	1
1	1	0	0	x	x	x	x	x	x	x	x	x	x	x	x
1	1	0	1	x	x	x	x	x	x	x	x	x	x	x	x
1	1	1	0	x	x	x	x	x	x	x	x	x	x	x	x

Flip-Flop Input Equations with K-Maps



$$J_3\,(Q_3, Q_2, Q_1, Q_0) = Q_2Q_1Q_0$$



$$K_3\,(Q_3, Q_2, Q_1, Q_0) = Q_1Q_0$$

J_2 Q_1Q_0

K_2 Q_1Q_0

Q_3Q_2 \ Q_1Q_0	00	01	11	10
00			1	
01	x	x	x	x
11	x	x	x	x
10				

$J_2\ (Q_3, Q_2, Q_1, Q_0) = \overline{Q_3} Q_1 Q_0$

Q_3Q_2 \ Q_1Q_0	00	01	11	10
00	x	x	x	x
01			1	
11	x	x	x	x
10	x	x	x	x

$K_2\ (Q_3, Q_2, Q_1, Q_0) = Q_1 Q_0$

J_1

Q_3Q_2 \ Q_1Q_0	00	01	11	10
00		1	x	x
01		1	x	x
11	x	x	x	x
10		1	x	x

$J_1\ (Q_3, Q_2, Q_1, Q_0) = Q_0$

K_1

Q_3Q_2 \ Q_1Q_0	00	01	11	10
00	x	x	1	
01	x	x	1	
11	x	x	x	x
10	x	x	1	

$K_1\ (Q_3, Q_2, Q_1, Q_0) = Q_0$

J_0

Q_3Q_2 \ Q_1Q_0	00	01	11	10
00	1	x	x	1
01	1	x	x	1
11	x	x	x	x
10	1	x	x	1

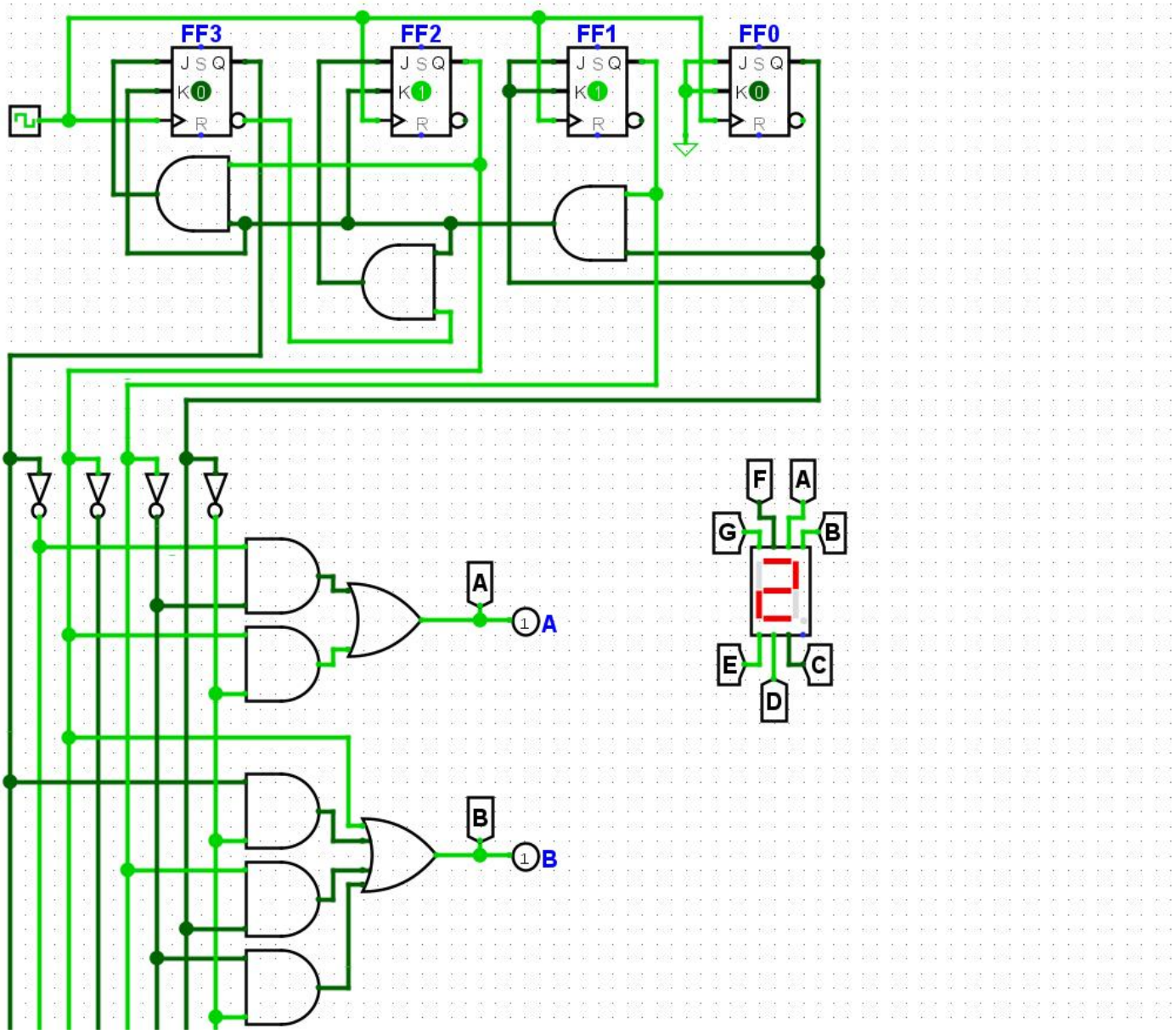
$J_0\ (Q_3, Q_2, Q_1, Q_0) = 1$

K_0

Q_3Q_2 \ Q_1Q_0	00	01	11	10
00	x	1	1	x
01	x	1	1	x
11	x	x	x	x
10	x	1	1	x

$K_0\ (Q_3, Q_2, Q_1, Q_0) = 1$

Circuit Diagram

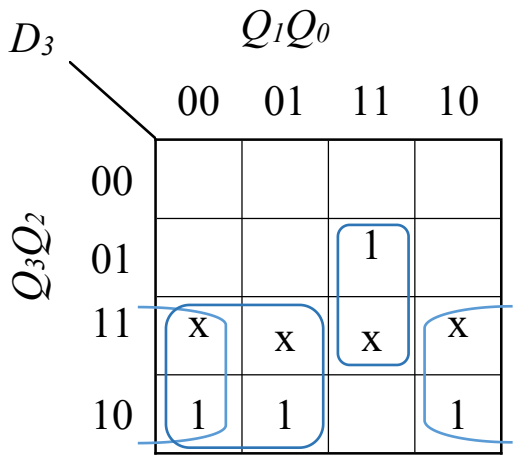


D Flip-Flop Implementation

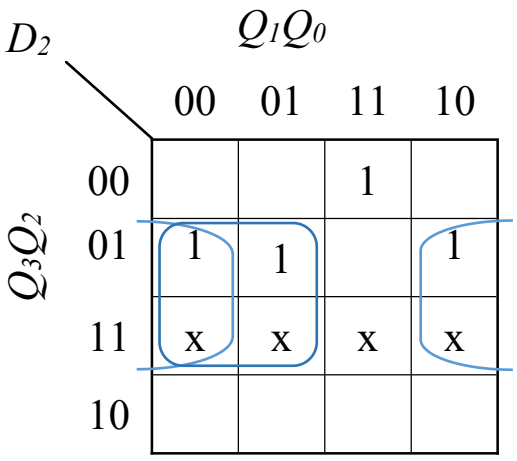
State Table

Present State				Next State				Flip Flop Inputs			
Q_3	Q_2	Q_1	Q_0	Q_3	Q_2	Q_1	Q_0	D_3	D_2	D_1	D_0
0	0	0	0	0	0	0	1	0	0	0	1
0	0	0	1	0	0	1	0	0	0	1	0
0	0	1	0	0	0	1	1	0	0	1	1
0	0	1	1	0	1	0	0	0	1	0	0
0	1	0	0	0	1	0	1	0	1	0	1
0	1	0	1	0	1	1	0	0	1	1	0
0	1	1	0	0	1	1	1	0	1	1	1
0	1	1	1	1	0	0	0	1	0	0	0
1	0	0	0	1	0	0	1	1	0	0	1
1	0	0	1	1	0	1	0	1	0	1	0
1	0	1	0	1	0	1	1	1	0	1	1
1	0	1	1	0	0	0	0	0	0	0	0
1	1	0	0	x	x	x	x	x	x	x	x
1	1	0	1	x	x	x	x	x	x	x	x
1	1	1	0	x	x	x	x	x	x	x	x
1	1	1	1	x	x	x	x	x	x	x	x

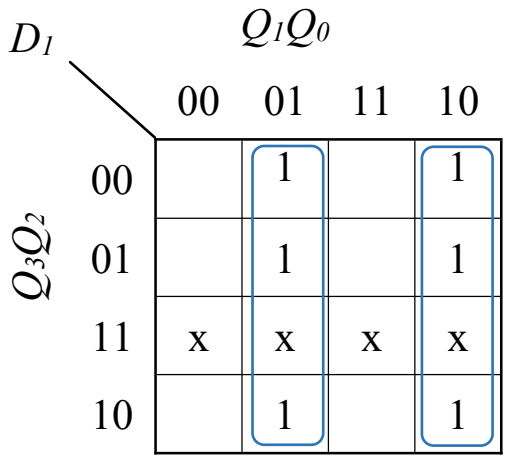
Flip-Flop Input Equations with K-Maps



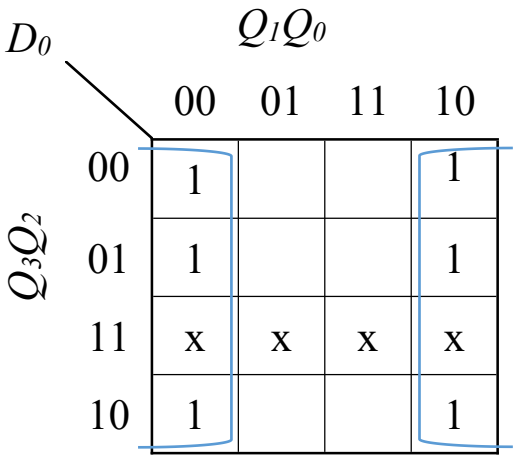
$$D_3\,(Q_3, Q_2, Q_1, Q_0) = Q_3\overline{Q_1} + Q_3\overline{Q_0} + Q_2Q_1Q_0 = Q_3(\overline{Q_1} + \overline{Q_0}) + Q_2Q_1Q_0$$



$$D_2\,(Q_3, Q_2, Q_1, Q_0) = Q_2\overline{Q_1} + Q_2\overline{Q_0} + \overline{Q_3}\,\overline{Q_2}Q_1Q_0 = Q_2(\overline{Q_1} + \overline{Q_0}) + \overline{Q_3}\,\overline{Q_2}Q_1Q_0$$

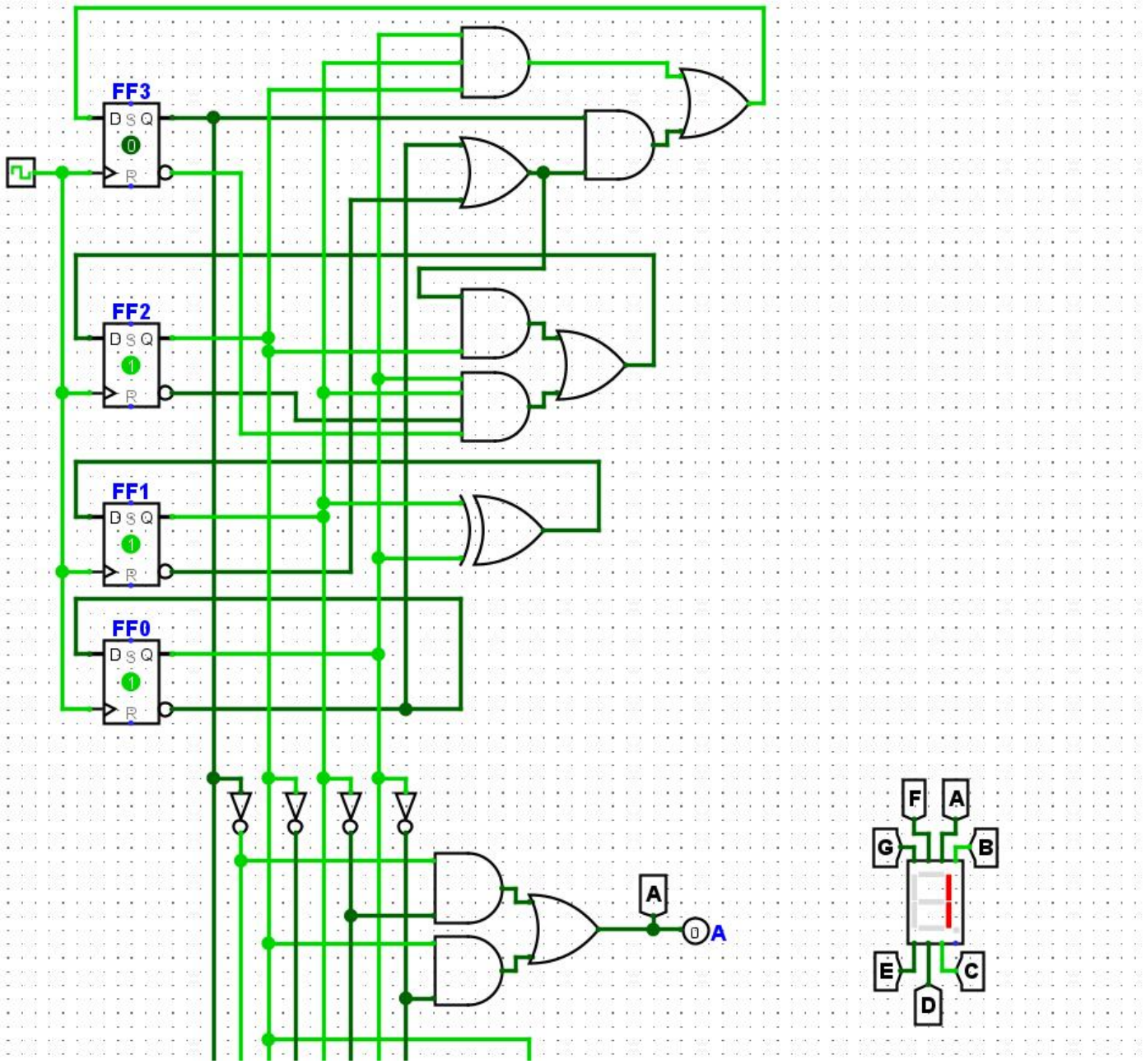


$$D_1\,(Q_3, Q_2, Q_1, Q_0) = \overline{Q_1}Q_0 + Q_1\overline{Q_0} = Q_1 \oplus Q_0$$



$$D_0\,(Q_3, Q_2, Q_1, Q_0) = \overline{Q_0}$$

Circuit Diagram

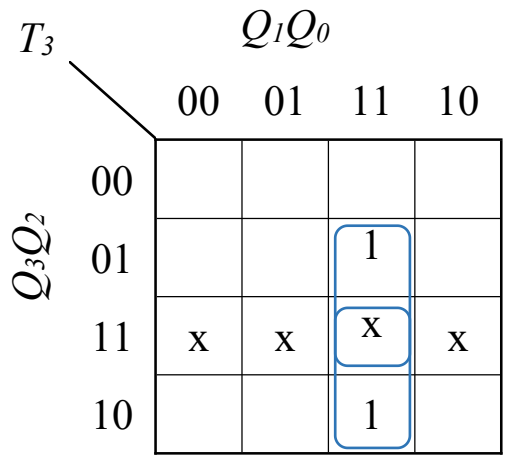


T Flip-Flop Implementation

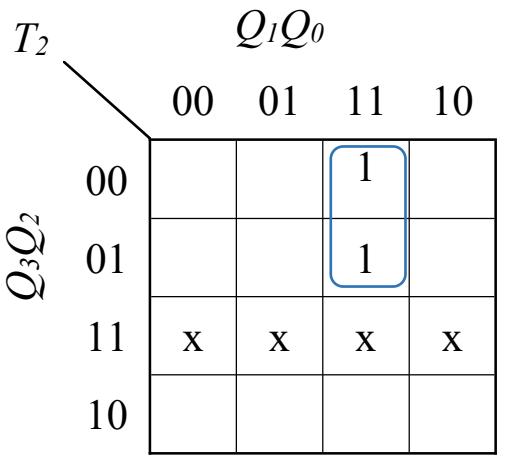
State Table

Present State				Next State				Flip Flop Inputs			
Q_3	Q_2	Q_1	Q_0	Q_3	Q_2	Q_1	Q_0	T_3	T_2	T_1	T_0
0	0	0	0	0	0	0	1	0	0	0	1
0	0	0	1	0	0	1	0	0	0	1	1
0	0	1	0	0	0	1	1	0	0	0	1
0	0	1	1	0	1	0	0	0	1	1	1
0	1	0	0	0	1	0	1	0	0	0	1
0	1	0	1	0	1	1	0	0	0	1	1
0	1	1	0	0	1	1	1	0	0	0	1
0	1	1	1	1	0	0	0	1	1	1	1
1	0	0	0	1	0	0	1	0	0	0	1
1	0	0	1	1	0	1	0	0	0	1	1
1	0	1	0	1	0	1	1	0	0	0	1
1	0	1	1	0	0	0	0	1	0	1	1
1	1	0	0	x	x	x	x	x	x	x	x
1	1	0	1	x	x	x	x	x	x	x	x
1	1	1	0	x	x	x	x	x	x	x	x
1	1	1	1	x	x	x	x	x	x	x	x

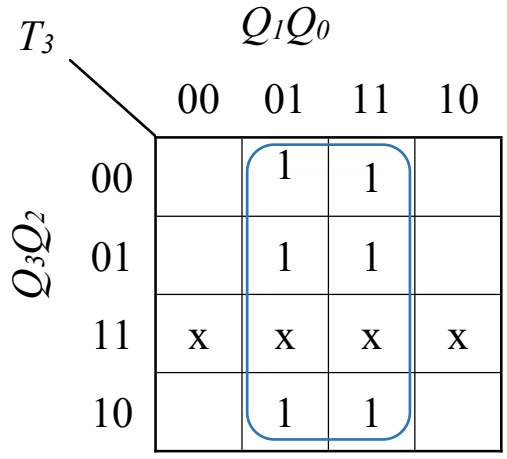
Flip-Flop Input Equations with K-Maps



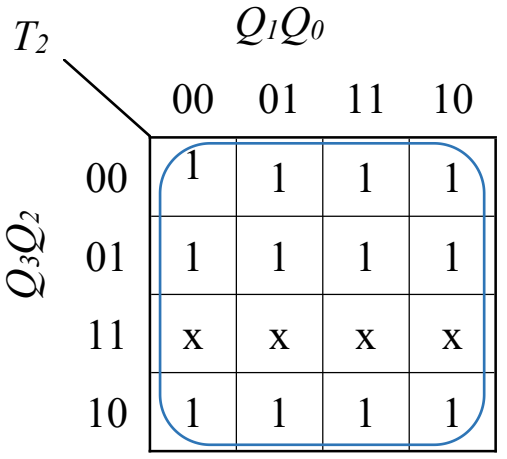
$$T_3\,(Q_3, Q_2, Q_1, Q_0) = Q_3Q_1Q_0 + Q_2Q_1Q_0 = Q_1Q_0(Q_3 + Q_2)$$



$$T_2\,(Q_3, Q_2, Q_1, Q_0) = \overline{Q_3}Q_1Q_0$$

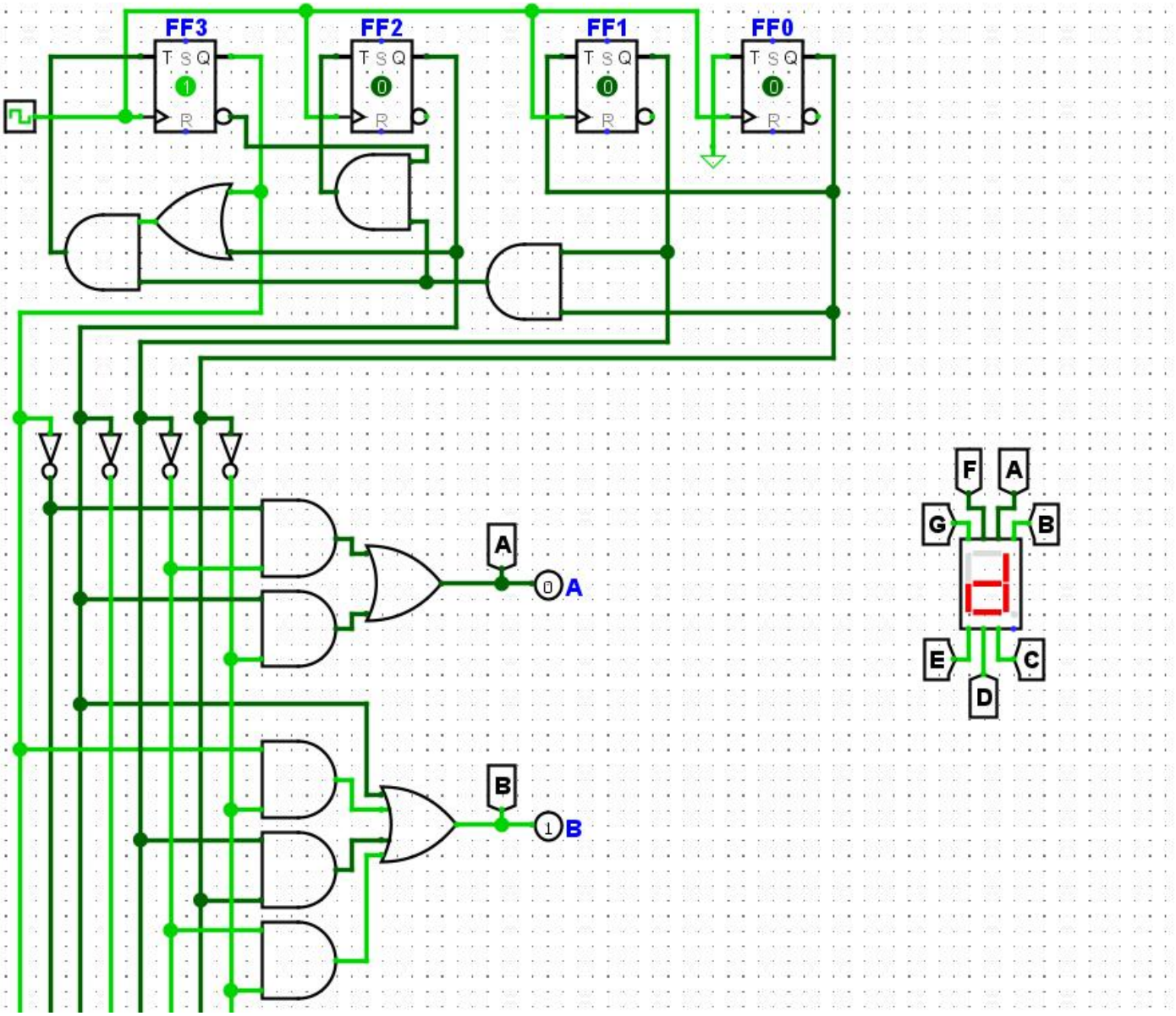


$$T_1\,(Q_3, Q_2, Q_1, Q_0) = Q_0$$



$$T_0\,(Q_3, Q_2, Q_1, Q_0) = 1$$

Circuit Diagram



Hardware Implementation

Choice of Implementation

Based on the same factors used to select for the hardware implementation of the combinational circuit, we choose to implement the sequential circuit using JK flip-flops as the counter uses the least number of gates in addition to the 4 flip-flops.
The materials for hardware implementation is listed in the following section.

Cost of Implementation

Index	Name	Cost Per Unit	Quantity	Total Cost
1	Breadboard (Large)	140	1	140
2	Male-to-Male Connectors x40	120	1	120
3	IC 74107 Dual JK Flip-Flops	40	2	80
4	4-pin Push Button Switch	10	1	10
5	IC 7408 Quad Two-Input AND Gates	30	1	30
6	Combinational Part	1240	1	1240
CUMULATIVE TOTAL				1620

Limitations

Over the course of the project, few hurdles were encountered, especially with regards to the hardware implementations of the seven segment display.

- The initial build of the combinational part provided incorrect outputs to different inputs. Therefore, the circuit had to be rebuilt and tested function by function to ensure that the final result was correct. Employing this step-by-step approach of verifying outputs towards completion would have reduced the time and effort taken.
- For the sequential part, the primary issue lay in deciding and implementing a mechanism for sending clock pulses to the flip-flops. We settled for a push-button switch mechanism that transitions a clock signal from HIGH to LOW on activation. At first, there were some inconsistencies in how characters appeared on the display, but by connecting a resistor between the switch and the VCC, it ensured that the transitions were smooth and stable. Nevertheless, the circuit would have benefited greatly by integrating a clock generator, that could have been built using a 555 Timer IC.

Conclusion

The project encompassing this report, the software implementations, and the hardware implementations compels students to involve themselves independently of any supervisor in the process of designing, developing, and implementing a digital system. In this manner, students get to evaluate their own capabilities and learn how to collaborate and leverage each other's strengths in order to deliver a complete and purposeful product. By overcoming the challenges and pitfalls along the way, the team gains practical insights that may enable them to become better digital designers and engineers as they apply themselves across the various fields that require competent hands offering fresh perspectives. Although the construction of a seven-segment display circuit in both its combinational and sequential forms has become one of the most common undertaking as a project in digital logic design, it is nonetheless the rudimentary leap that one might need to excel in the discipline.