Digital Clock Implementation using Arduino with Multiplexing and Editing Features

Dhawal

Department of Electrical Engineering
Indian Institute of Technology Hyderabad
Email: ee24btech11015@iith.ac.in

Electrical Engineering Department
Indian Institute of Technology, Hyderabad

Outline

Introduction

- Digital clock system with editing features using Arduino
- Multiplexing six 7-segment displays using minimal I/O pins
- Pause/play functionality and digit-by-digit editing
- Boolean logic for increment/decrement and constraints for each digit

Components List

Component	Value	Quantity
Arduino Uno		1
USB Cable	Type B	1
Seven Segment Display	Common Cathode	6
Push Buttons		4
IC 7447		1
Jumper Wires	M-M	16
Breadboard		1
Resistors	220Ω	7
Resistors	$10k\Omega$	4

Arduino Pin Connections

Item	Arduino Pin	Function
Button 1	D10	Edit Mode Toggle
Button 2	D11	Next Digit Selection
Button 3	D12	Increment Digit
Button 4	D13	Decrement Digit
IC 7447 Pin 7	D0	BCD Bit 0 (A)
IC 7447 Pin 1	D1	BCD Bit 1 (B)
IC 7447 Pin 2	D2	BCD Bit 2 (C)
IC 7447 Pin 6	D3	BCD Bit 3 (D)
Display 1	D4	Hours Tens Digit
Display 2	D5	Hours Units Digit
Display 3	D6	Minutes Tens Digit
Display 4	D7	Minutes Units Digit
Display 5	D8	Seconds Tens Digit
Display 6	D9	Seconds Units Digit

Multiplexing

- All BCD inputs shared among six 7-segment displays
- Arduino controls enable pins D4-D9
- Each digit displayed for 1ms $\rightarrow$ appears continuous
- Saves I/O pins, enables six-digit display

Editing System

1. PAUSE button toggles run/edit mode
2. NEXT button selects which digit to edit
3. INC button increments selected digit (rollover constraints)
4. DEC button decrements selected digit (rollunder constraints)
5. Selected digit blinks every 500ms

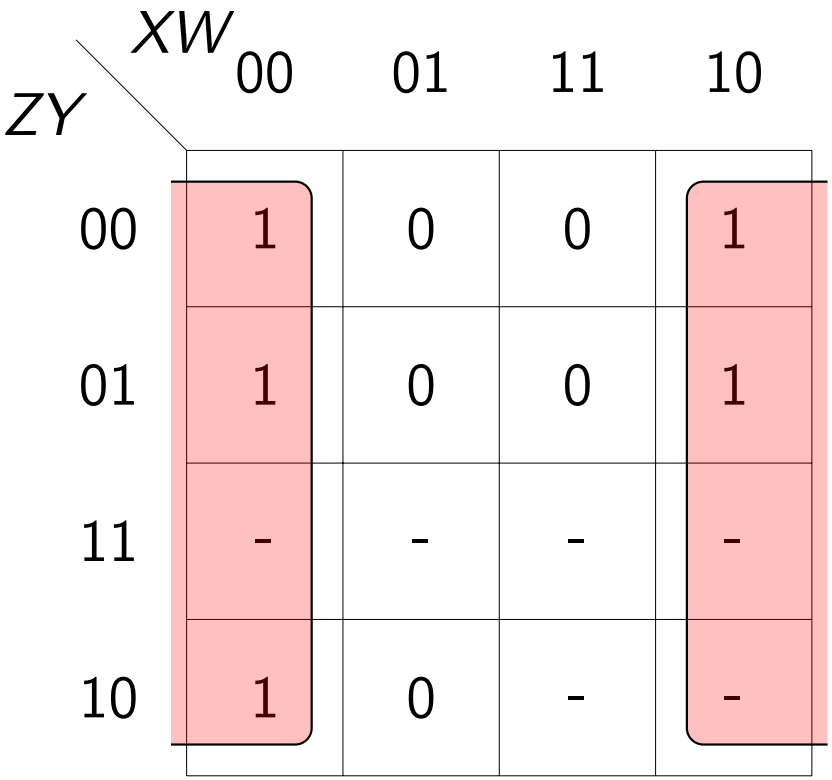
Digit Constraints

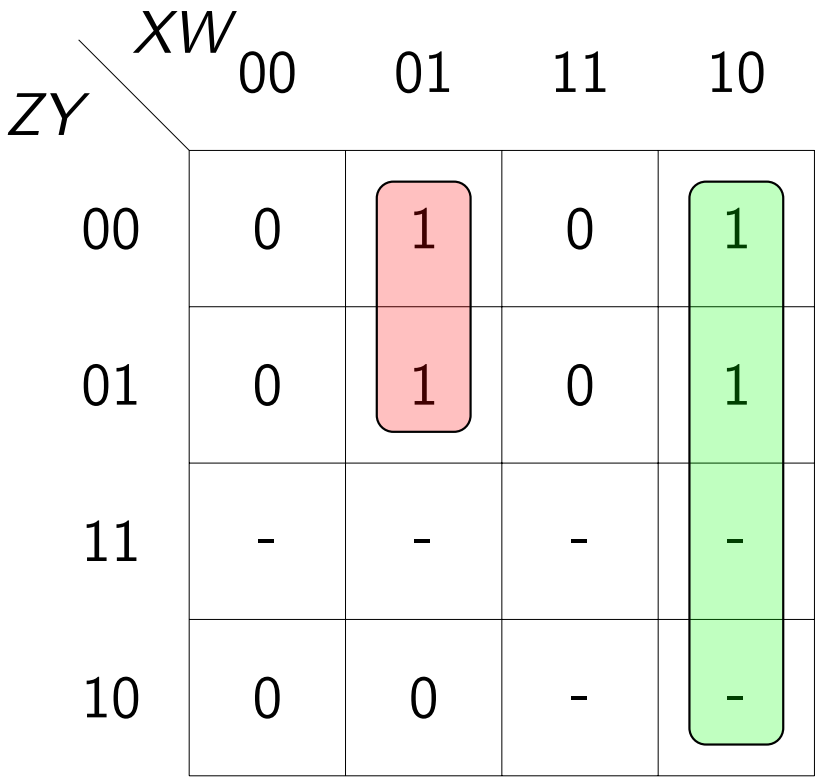
- Seconds/Minutes Ones: 0–9
- Seconds/Minutes Tens: 0–5
- Hours Ones: 0–9 (if tens = 0/1), 0–3 (if tens = 2)
- Hours Tens: 0–2

Seconds / Minutes / Hours (Tens= 0/1) Ones (0-9)

Z	Y	X	W	D	C	B	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

K-maps

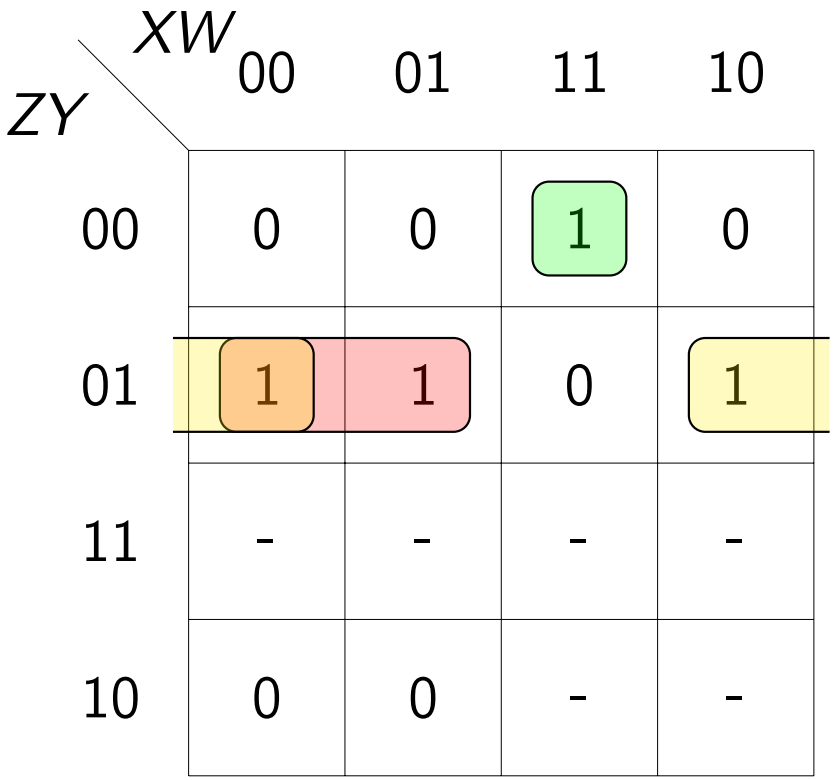


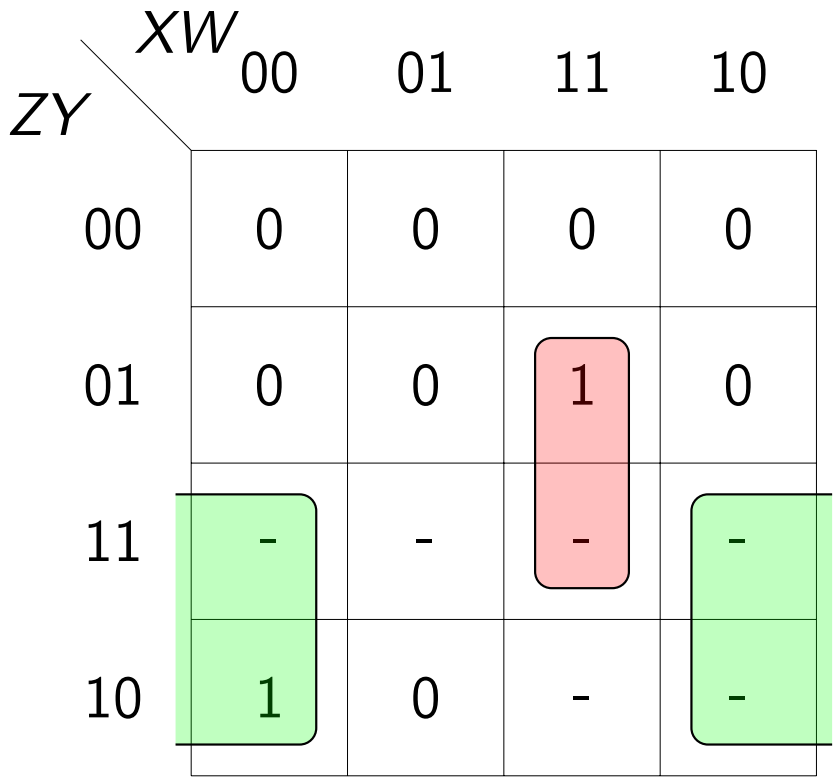


$$A = W_1'$$

$$B = (W_1 X_1' Z_1') + (W_1' X_1)$$

K-maps





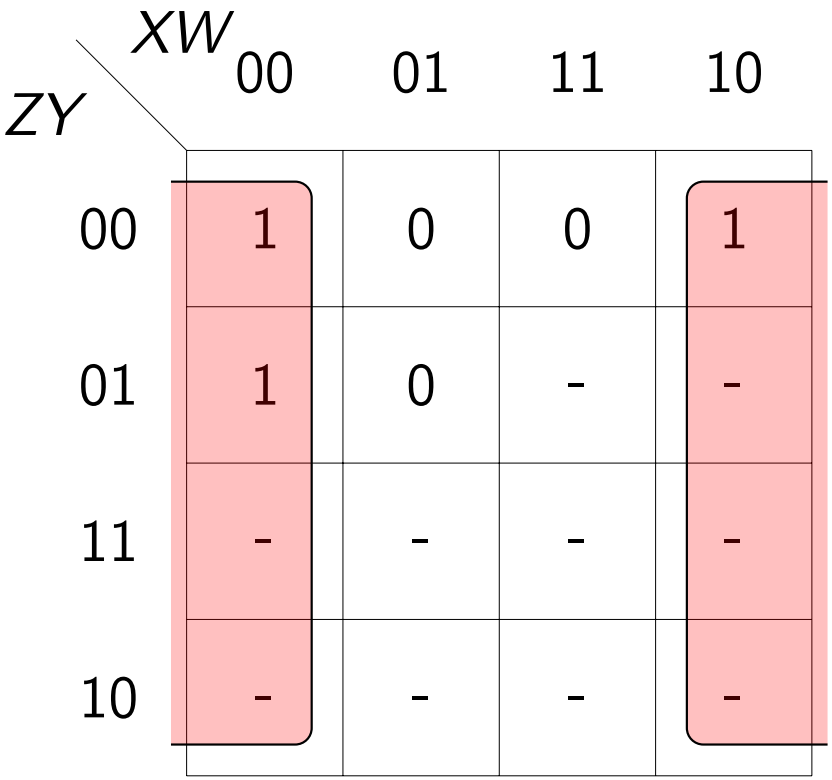
$$C = (X_1' Y_1) + (W_1' Y_1) + (W_1 X_1 Y_1')$$

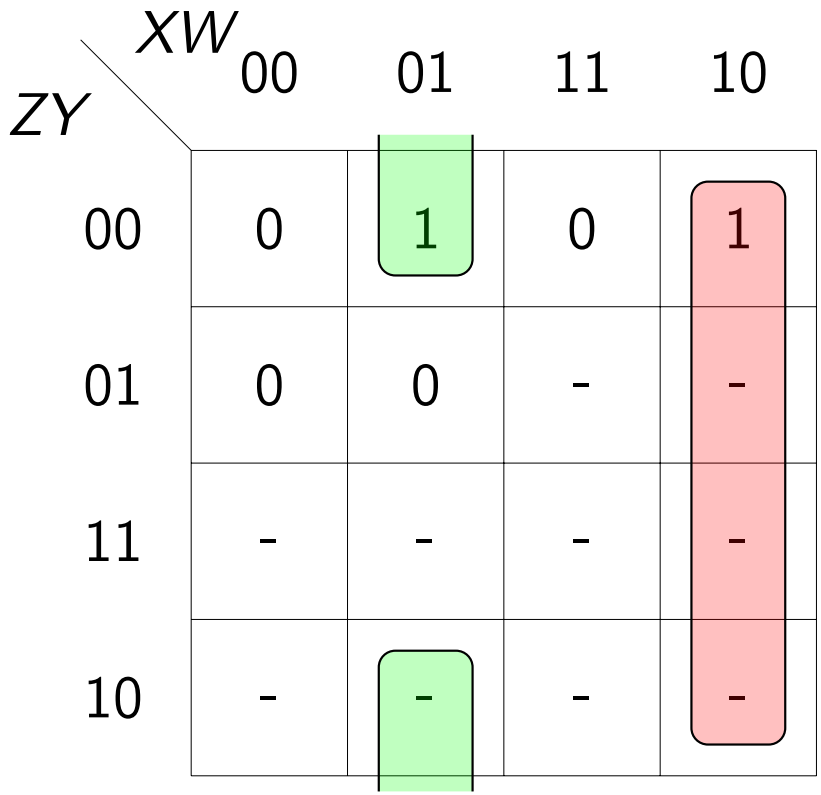
$$D = (W_1' Z_1) + (W_1 X_1 Y_1)$$

Seconds / Minutes Tens (0-5)

Z	Y	X	W	D	C	B	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	0	0	0

K-maps

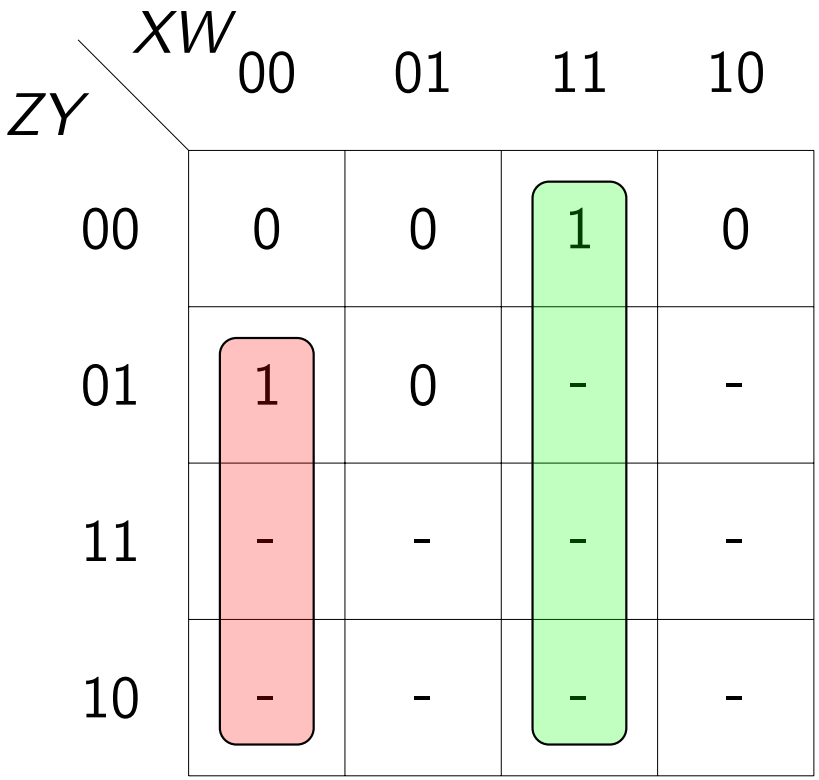




$$A = W_2'$$

$$B = (W_2 X_2' Y_2') + (W_2' X_2)$$

K-maps



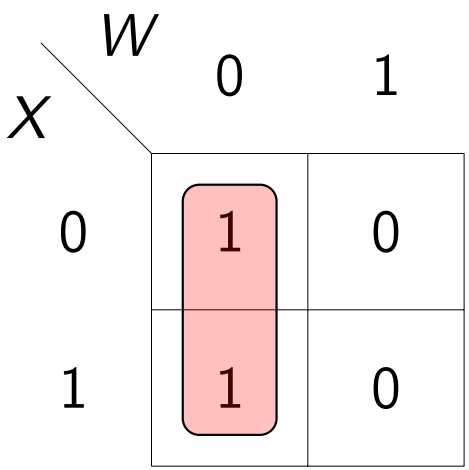
$D = 0$

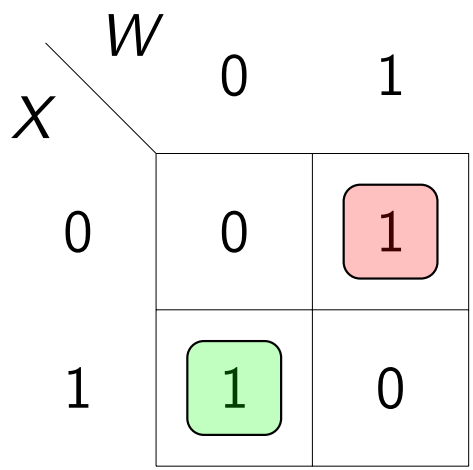
$$C = (W_2X_2) + (W_2'X_2'Y_2)$$

Hours Ones (Tens = 2 → 0-3)

X	W	D	C	B	A
0	0	0	0	0	1
0	1	0	0	1	0
1	0	0	0	1	1
1	1	0	0	0	0

K-maps





$$A = W_5'$$

$$B = (W_5 X_5') + (W_5' X_5)$$

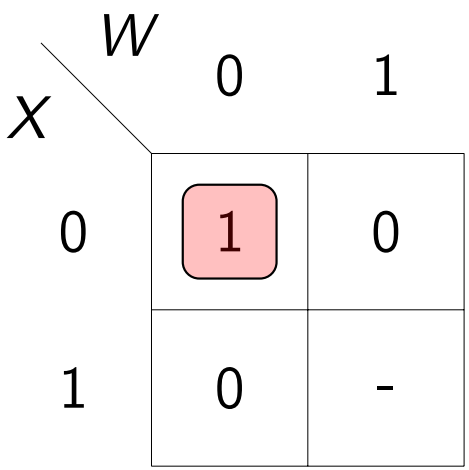
$$C = 0 \tag{1}$$

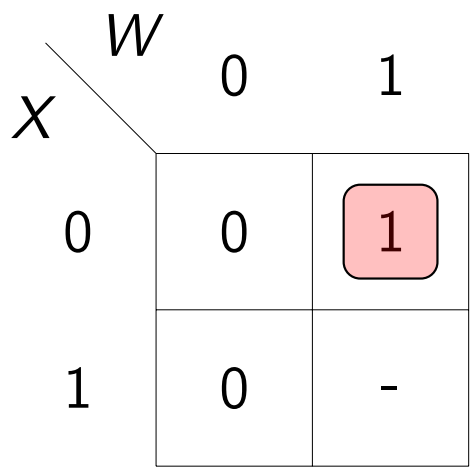
$$D = 0$$

Hours Tens (0-2)

X	W	D	C	B	A
0	0	0	0	0	1
0	1	0	0	0	0
1	0	0	0	0	0

K-maps





$$A = W_6' X_6'$$

$$B = W_6 X_6'$$

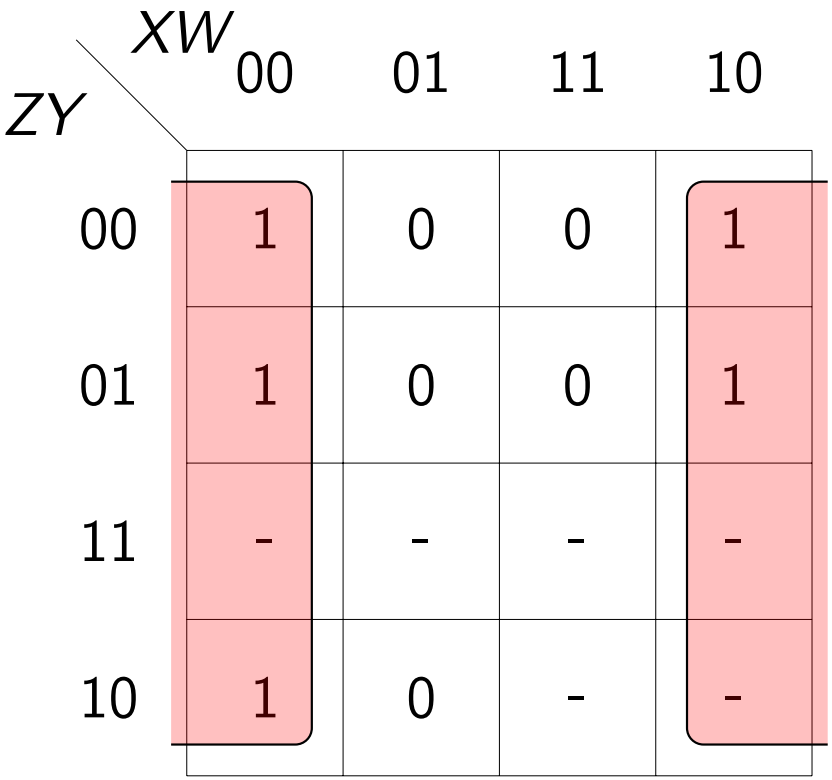
$$C = 0 \tag{2}$$

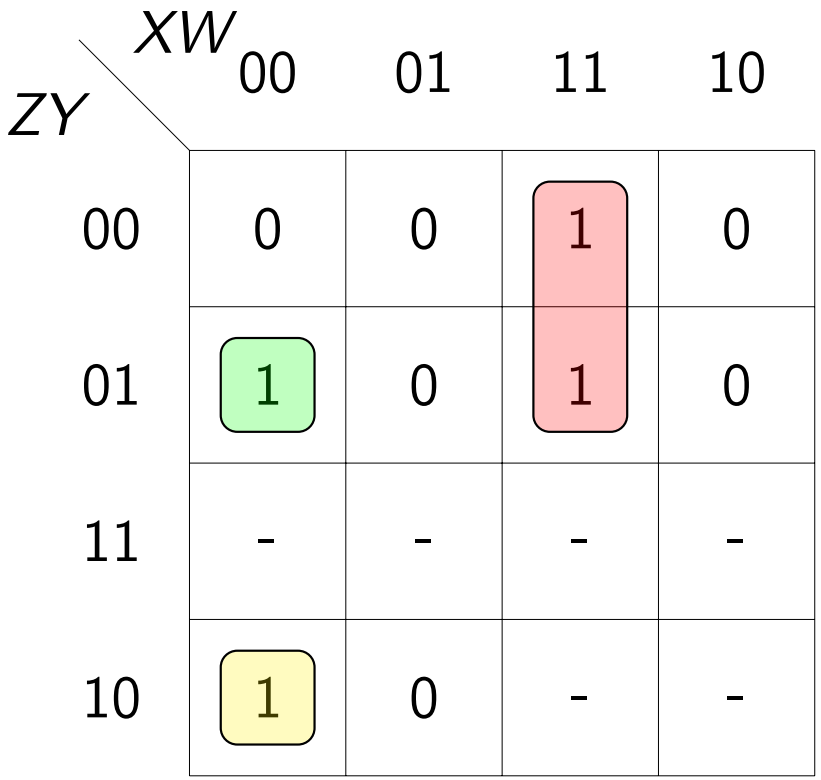
$$D = 0$$

Seconds / Minutes / Hours (Tens= 0/1) Ones (0-9)

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

K-maps





$$A = W_1'$$

$$B = (X_1' W_1' ((Z_1' Y_1) + (Z_1 Y_1'))) + (Z_1' W_1 X$$

K-maps

ZY \ XW	00	01	11	10
00	0	0	0	0
01	0	1	1	1
11	-	-	-	-
10	1	0	-	-

ZY \ XW	00	01	11	10
00	1	0	0	0
01	0	0	0	0
11	-	-	-	-
10	0	1	-	-

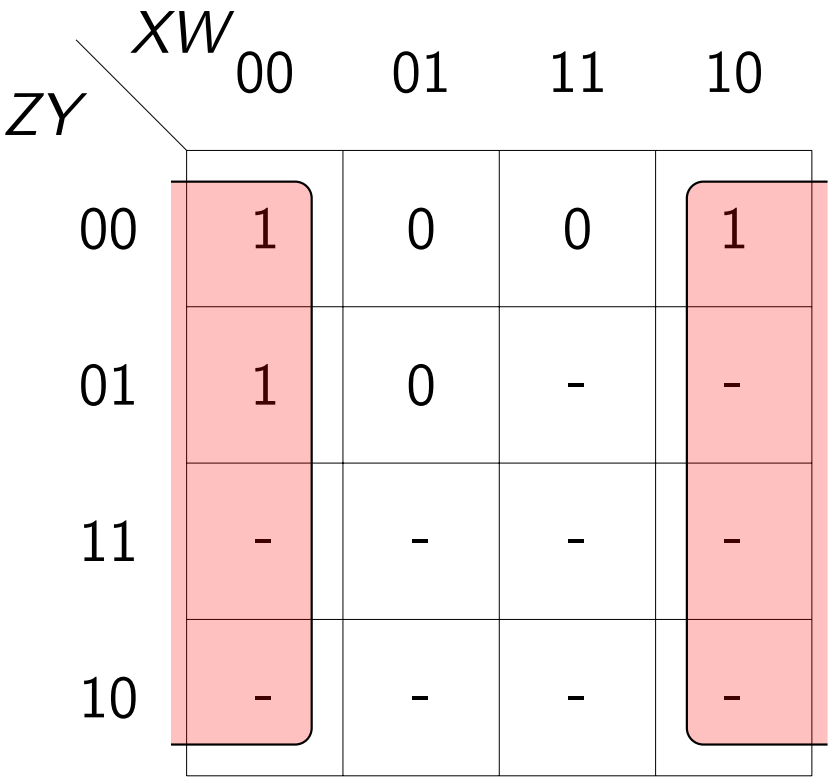
$$C = (Z_1' Y_1 (X_1 + W_1)) + (Z_1 X_1' W_1' Y_1')$$

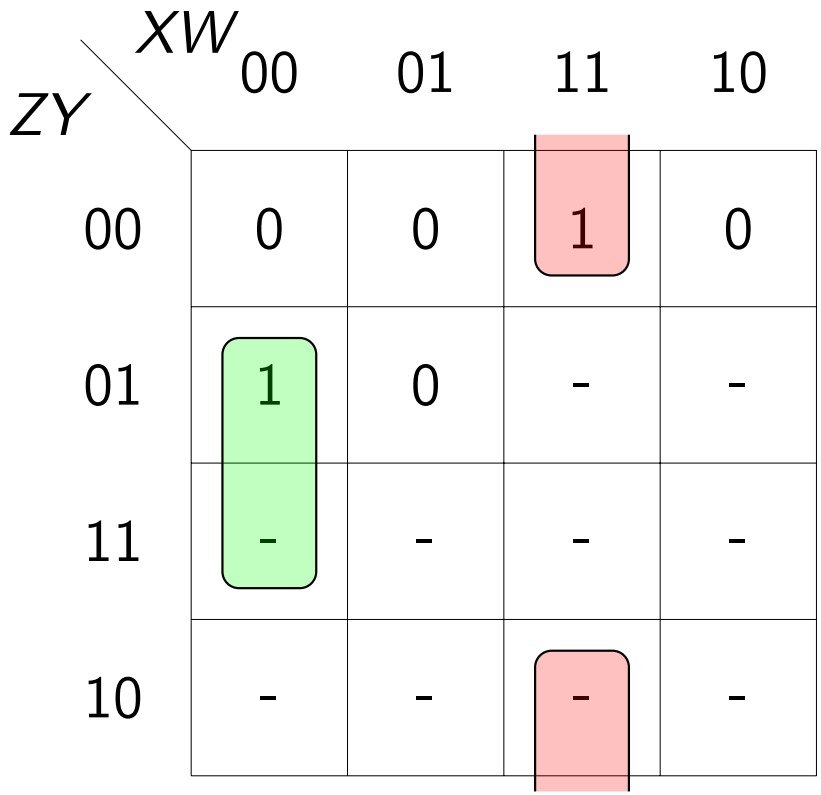
$$D = X_1' Y_1' ((Z_1 W_1) + (Z_1' W_1'))$$

Seconds / Minutes Tens (0-5)

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

K-maps

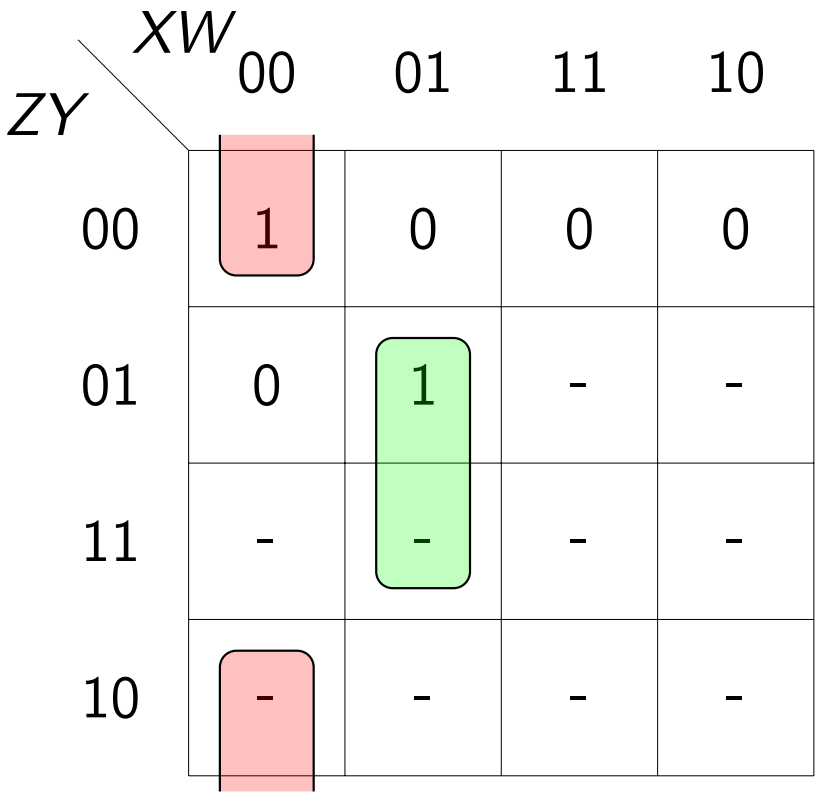




$$A = W_2'$$

$$B = (Y_2 X_2' W_2') + (Y_2' X_2 W_2)$$

K-maps



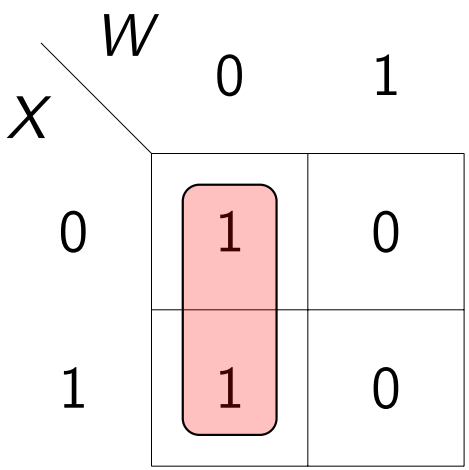
$$D = 0$$

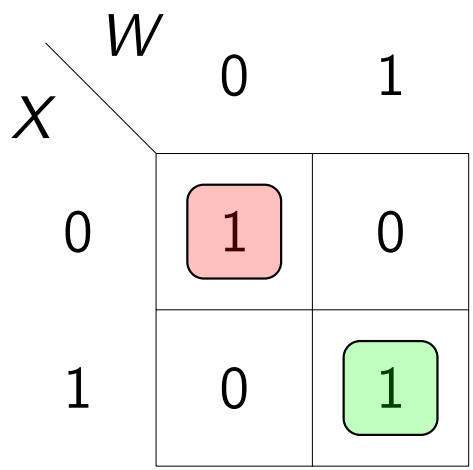
$$C = X_2'((Y_2W_2) + (Y_2'W_2'))$$

Hours Ones (Tens = 2 → 0-3)

X	W	D	C	B	A
0	0	0	0	1	1
0	1	0	0	0	0
1	0	0	0	0	1
1	1	0	0	1	0

K-maps





$$A = W_5'$$

$$B = (X_5 W_5) + (X_5' W_5')$$

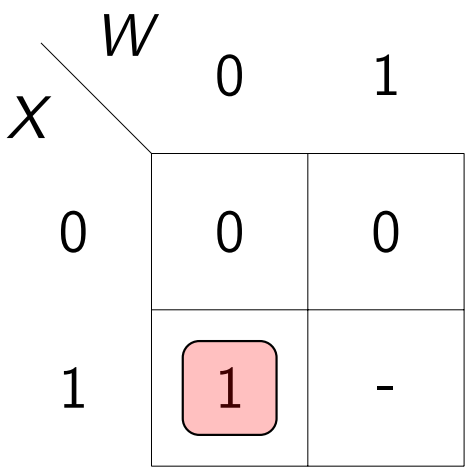
$$C = 0 \tag{3}$$

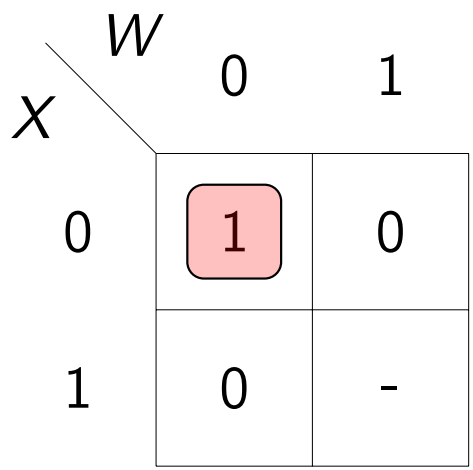
$$D = 0$$

Hours Tens (0-2)

X	W	D	C	B	A
0	0	0	0	1	0
0	1	0	0	0	0
1	0	0	0	0	1

K-maps





$$A = X_6 W_6'$$

$$B = X_6' W_6'$$

$$C = 0 \tag{4}$$

$$D = 0$$

Hardware Implementation

- Connect seven-segment displays to breadboard
- Connect all segment outputs together (through resistors)
- Make connections to IC7447 and buttons to Arduino
- Add current-limiting resistors for LEDs
- Add pull-down resistors for buttons

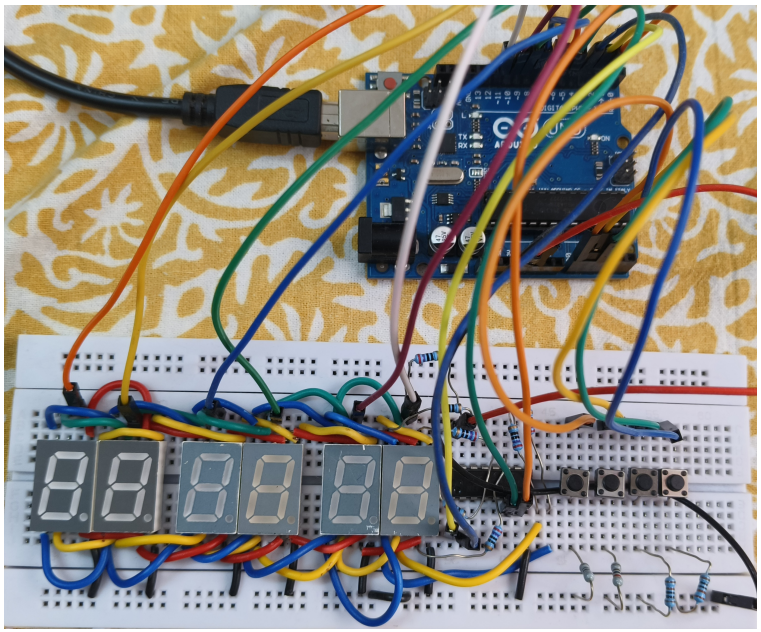
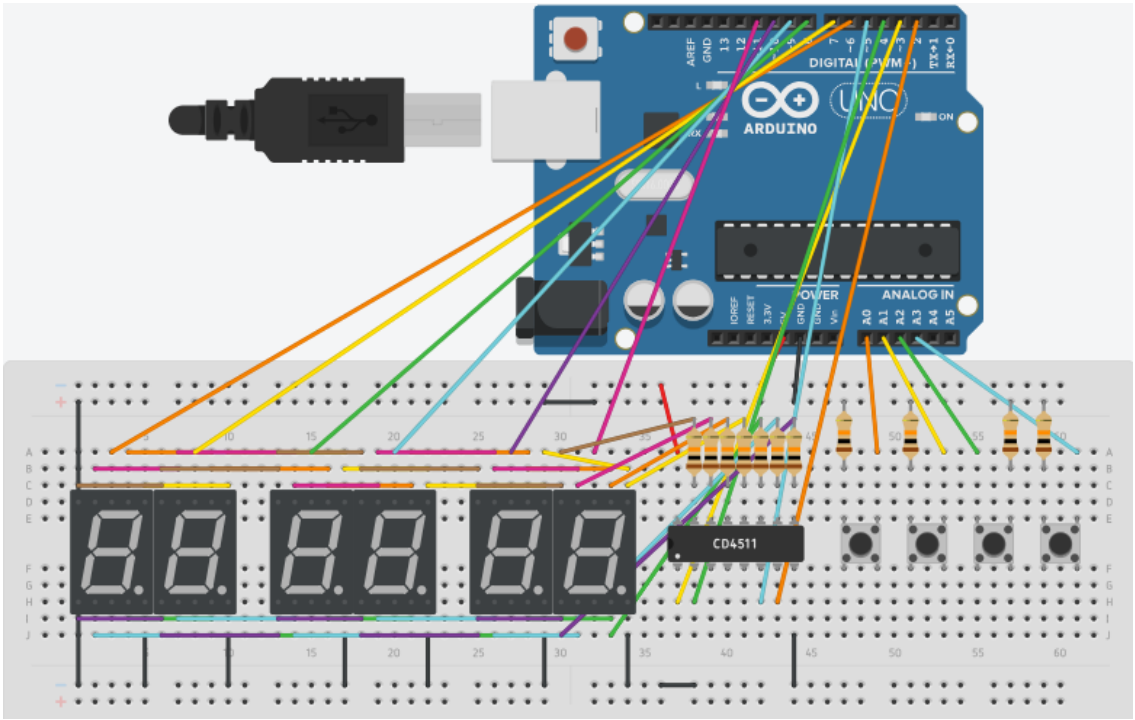


Figure: Final Arduino-based Clock Implementation

Tinkercad Simulation



Clock Tinkercad Simulation

Summary

- Successfully implemented digital clock with editing features
- Efficient multiplexing and minimal I/O usage
- Complete increment/decrement logic implemented via Boolean expressions
- Full digit-by-digit editing with constraints for hours, minutes, seconds

Acknowledgment

The complete source code and documentation can be found at:
`https://github.com/Dhawal24112006/projects.git`

Thank You!