0-99 Synchronous Up/Down Counter Design Report

1 Overview

This design creates a decimal counter (0-99) with:

- Two seven-segment displays
- Up/down counting functionality via push buttons
- LED indicator when count reaches 99
- Synchronous operation using JK flip-flops

2 System Design

2.1 Components Required

1. 8 JK flip-flops (4 for each decimal digit)
2. 2 seven-segment displays with BCD-to-7-segment decoders
3. 2 push buttons (up/down)
4. Logic gates (AND, OR, NOT)
5. Debounce circuits for push buttons
6. LED indicator with driver
7. Clock source

2.2 Counter Structure

We'll implement this as two cascaded BCD (Binary Coded Decimal) counters:

- First stage: Units digit (0-9)
- Second stage: Tens digit (0-9)

3 Circuit Design

3.1 Flip-Flop Configuration

For each decimal digit (0-9), we need 4 JK flip-flops to represent values in binary:

- Units digit: FF_0, FF_1, FF_2, FF_3 (representing 1, 2, 4, 8)
- Tens digit: FF_4, FF_5, FF_6, FF_7 (representing 10, 20, 40, 80)

3.2 Clock and Control Logic

3.2.1 Clock Source:

- Each push button will generate one clock pulse
- The UP and DOWN buttons will share the same clock line but affect the J/K inputs differently

3.2.2 Input Logic for UP Counter:

For the units digit (0-9):

$FF_0 : J_0 = 1, K_0 = 1$ (toggles with every clock)
$FF_1 : J_1 = Q_0, K_1 = Q_0$ (toggles when $Q_0 = 1$)
$FF_2 : J_2 = Q_0 \cdot Q_1, K_2 = Q_0 \cdot Q_1$ (toggles when $Q_0 = Q_1 = 1$)
$FF_3 : J_3 = Q_0 \cdot Q_1 \cdot Q_2, K_3 = Q_0 \cdot Q_1 \cdot Q_2$ (toggles when $Q_0 = Q_1 = Q_2 = 1$)

For the tens digit, connect to the units digit:

- Clock = Clock AND (Units=9) AND UP for increment

3.2.3 Input Logic for DOWN Counter:

For counting down, we'll need additional logic:

- FF_0: $J = 1, K = 1$ (still toggles)
- For other flip-flops, modify inputs to handle down-counting
- Clock = Clock AND (Units=0) AND DOWN for tens digit decrement

3.3 Overflow and Reset Logic

3.3.1 Detection Logic:

- 99 detection: $Q_7 \cdot Q_6 \cdot \overline{Q_5} \cdot \overline{Q_4} \cdot Q_3 \cdot Q_2 \cdot \overline{Q_1} \cdot \overline{Q_0}$ (1001 1001 in binary)
- 00 detection: $\overline{Q_7} \cdot \overline{Q_6} \cdot \overline{Q_5} \cdot \overline{Q_4} \cdot \overline{Q_3} \cdot \overline{Q_2} \cdot \overline{Q_1} \cdot \overline{Q_0}$

3.3.2 Indicator LED:

- LED = $Q_7 \cdot Q_6 \cdot \overline{Q_5} \cdot \overline{Q_4} \cdot Q_3 \cdot Q_2 \cdot \overline{Q_1} \cdot \overline{Q_0}$ (99 in BCD: 1001 1001)

4 Implementation Details

4.1 BCD Counter Logic

Each BCD counter needs to count from 0-9 only, so we need reset logic:

- When count reaches 1010 (10 in binary), reset to 0000
- For UP counter: When a digit reaches 9 and UP is pressed, reset digit to 0 and increment next digit
- For DOWN counter: When a digit reaches 0 and DOWN is pressed, set digit to 9 and decrement next digit

4.2 Seven-Segment Display Interface

For each 4-bit BCD digit, connect to a BCD-to-7-segment decoder:

- Units digit: Connect FF_0-FF_3 outputs
- Tens digit: Connect FF_4-FF_7 outputs

4.3 Push Button Interface

4.3.1 Debounce Circuit:

- Add RC filter and Schmitt trigger for each button
- Use edge-triggered operation to prevent multiple counts

4.3.2 Direction Control:

- UP Button: Activates UP logic paths
- DOWN Button: Activates DOWN logic paths

5 Circuit Implementation

5.1 Clock Generation

```
Push Button → Debounce Circuit → Edge Detector → Clock Signal
```

5.2 JK Flip-Flop Connections for Units Digit (UP)

$$\begin{aligned}
&\text{FF}_0 : J_0 = 1, K_0 = 1 \\
&\text{FF}_1 : J_1 = Q_0, K_1 = Q_0 \\
&\text{FF}_2 : J_2 = Q_0 \cdot Q_1, K_2 = Q_0 \cdot Q_1 \\
&\text{FF}_3 : J_3 = Q_0 \cdot Q_1 \cdot Q_2, K_3 = Q_0 \cdot Q_1 \cdot Q_2
\end{aligned}$$

5.3 JK Flip-Flop Connections for Units Digit (DOWN)

$$\begin{aligned}
&\text{FF}_0 : J_0 = 1, K_0 = 1 \\
&\text{FF}_1 : J_1 = \overline{Q_0}, K_1 = \overline{Q_0} \\
&\text{FF}_2 : J_2 = \overline{Q_0} + \overline{Q_1}, K_2 = \overline{Q_0} + \overline{Q_1} \\
&\text{FF}_3 : J_3 = \overline{Q_0} + \overline{Q_1} + \overline{Q_2}, K_3 = \overline{Q_0} + \overline{Q_1} + \overline{Q_2}
\end{aligned}$$

5.4 Reset Logic for Units Digit

$$\begin{aligned}
\text{Reset_Units_UP} &= Q_3 \cdot \overline{Q_2} \cdot \overline{Q_1} \cdot Q_0 \text{ (When count = 1001 or 9)} \\
\text{Reset_Units_DOWN} &= \overline{Q_3} \cdot \overline{Q_2} \cdot \overline{Q_1} \cdot \overline{Q_0} \text{ (When count = 0000 or 0)}
\end{aligned}$$

5.5 Tens Digit Clock

$$\begin{aligned}
\text{Tens_Clock_UP} &= \text{Main_Clock AND } Q_3 \cdot \overline{Q_2} \cdot \overline{Q_1} \cdot Q_0 \text{ AND UP_Button} \\
\text{Tens_Clock_DOWN} &= \text{Main_Clock AND } \overline{Q_3} \cdot \overline{Q_2} \cdot \overline{Q_1} \cdot \overline{Q_0} \text{ AND DOWN_Button}
\end{aligned}$$

5.6 99 Detection for LED

$$\text{LED} = (Q_7 \cdot Q_6 \cdot \overline{Q_5} \cdot \overline{Q_4}) \text{ AND } (Q_3 \cdot Q_2 \cdot \overline{Q_1} \cdot \overline{Q_0})$$

6 Testing Procedure

1. Power up the circuit and verify all displays show "00"
2. Press the UP button repeatedly and verify counting from 00→01→...→99
3. Confirm LED illuminates when counter reaches 99
4. Press UP button at 99 and verify counter wraps to 00
5. Press DOWN button repeatedly and verify decremental counting
6. Press DOWN button at 00 and verify counter wraps to 99

7 Detailed Circuit Operation

7.1 UP Operation

For UP counting, the circuit performs these operations:

1. When UP button is pressed, a clock pulse is generated
2. The units digit increments by 1
3. If units digit was 9, it rolls over to 0 and generates a carry pulse
4. The carry pulse increments the tens digit
5. When the counter reaches 99, the LED lights up
6. The next UP pulse rolls the counter to 00

7.2 DOWN Operation

For DOWN counting, the circuit performs these operations:

1. When DOWN button is pressed, a clock pulse is generated
2. The units digit decrements by 1
3. If units digit was 0, it rolls over to 9 and generates a borrow pulse
4. The borrow pulse decrements the tens digit
5. When the counter reaches 00, the next DOWN pulse rolls the counter to 99

7.3 Implementation Notes

1. Use synchronous design principles - all flip-flops change state based on the same clock signal
2. Add pull-down resistors to button inputs to prevent floating inputs
3. Consider adding a manual reset button to return the counter to 00
4. Ensure adequate power supply filtering to prevent erroneous triggering

Table 1: Complete Connection Table for 0-99 Up/Down Counter

Component	**Pin/Terminal**	**Connection Details**	**Notes**
FF_0	Clock	Main Clock AND (UP OR DOWN)	First flip-flop
	J	Logic HIGH (1)	Always set to 1
	K	Logic HIGH (1)	Always set to 1
	Q	Connects to J_1, K_1, and Seven-Segment Decoder Bit 0	LSB of Units
	$\overline{Q}$	Not used	
FF_1	Clock	Main Clock AND (UP OR DOWN)	
	J, K	UP mode: Q_0, Down mode: $\overline{Q}_0$	Multiplexed input
	Q	Connects to Seven-Segment Decoder Bit 1, J_2, K_2	
	$\overline{Q}$	Not used	
FF_2	Clock	Main Clock AND (UP OR DOWN)	
	J, K	UP mode: (Q_0 AND Q_1), Down mode: ($\overline{Q}_0$ OR $\overline{Q}_1$)	Multiplexed input
	Q	Connects to Seven-Segment Decoder Bit 2, J_3, K_3	
	$\overline{Q}$	Not used	
FF_3	Clock	Main Clock AND (UP OR DOWN)	
	J, K	UP mode: (Q_0 AND Q_1 AND Q_2), Down mode: ($\overline{Q}_0$ OR $\overline{Q}_1$ OR $\overline{Q}_2$)	Multiplexed input
	Q	Connects to Seven-Segment Decoder Bit 3	MSB of Units
	$\overline{Q}$	Connects to detection logic	
FF_4	Clock	($Q_3 \cdot \overline{Q}_2 \cdot \overline{Q}_1 \cdot Q_0$ AND UP) OR ($\overline{Q}_3 \cdot \overline{Q}_2 \cdot \overline{Q}_1 \cdot \overline{Q}_0$ AND DOWN)	Carries/Borrows
	J, K	Logic HIGH (1)	Always set to 1
	Q	Connects to J_5, K_5, and Seven-Segment Decoder Bit 0	LSB of Tens
	$\overline{Q}$	Not used	
UP Mode	-	UP Button Output	Sets count direction
DOWN Mode	-	DOWN Button Output	Sets count direction
Units Display	BCD inputs	Q_0, Q_1, Q_2, Q_3	Units digit decoder
Tens Display	BCD inputs	Q_4, Q_5, Q_6, Q_7	Tens digit decoder
99 Detector	Inputs	$Q_7 \cdot Q_6 \cdot \overline{Q}_5 \cdot \overline{Q}_4 \cdot Q_3 \cdot Q_2 \cdot \overline{Q}_1 \cdot \overline{Q}_0$	Detects 99 (1001 1001 in binary)
	Output	Connects to LED	Lights when count reaches 99
UP Debounce	Input	UP Button	Raw button input
	Output	UP Signal	Clean button signal
DOWN Debounce	Input	DOWN Button	Raw button input
	Output	DOWN Signal	Clean button signal

Continued on next page

Table 1 – *Continued from previous page*

Component	**Pin/Terminal**	**Connection Details**	**Notes**
Reset	Input	Reset Button	Optional
	Output	Connects to all FF CLR inputs	Active low reset