

บททดลองที่ 9 การทดลองวงจรนับ (Counter)

วัตถุประสงค์

1. เพื่อศึกษาการทำงานของวงจรนับทางดิจิทัลในแบบต่างๆ
2. เพื่อให้เห็นประโยชน์ของ JK Flip Flops ที่สามารถนำมาสร้างเป็นวงจรนับ
3. เรียนรู้ถึงการออกแบบวงจรนับ และสามารถนำวงจรนับไปประยุกต์ใช้งาน

อุปกรณ์ที่ใช้ในการทดลอง

IC 7473	2	ตัว,	IC 7400	1	ตัว,	IC 7408	1	ตัว
IC 555	1	ตัว,	IC7493	1	ตัว,	IC 7485	1	ตัว
R 100 k Ω	1	ตัว,	R 10 k Ω	1	ตัว,	C 0.01 μ F	1	ตัว
LED								
Switch กดติดปล่อยดับ 1 ตัว								
LED 7-Segment display								

ทฤษฎีย่อ

วงจรนับ ก็คือวงจรที่นำ FF หลายๆตัว มาใช้งานรวมกัน โดย FF แต่ละตัวที่มาต่อกันจะประพฤติตัวอยู่ในสถานะของ T Flip Flops ซึ่งจากการทดลองที่ 7 ในเรื่องของ Flip Flops เราพบว่า JK FF สามารถทำให้เป็น T FF ได้ง่ายและสะดวกที่สุด (เพียง Set ให้ขา J และ K มีสถานะเป็น 1 เท่านั้น) โดยพื้นฐานสถานะของวงจรนับจะมีการเปลี่ยนแปลง หรือเริ่มทำการนับก็ต่อเมื่อมี Pulse จ่ายเข้าเป็น Input ให้กับวงจร Counter

เมื่อ Counter ได้รับ Pulse ตัว FF ที่เชื่อมกันอยู่ภายในจะเริ่มทำงานในแบบของ T FF เพื่อทำการนับจำนวน Pulse ที่จ่ายให้กับตัว Counter ตัวอย่างเช่น ถ้ามี Pulse จำนวน 5 ลูกจ่ายให้กับวงจร Counter ที่ประกอบด้วย JK FF จำนวน 3 ตัว ค่า Output ที่ได้จาก FF แต่ละตัวจะรวมเป็น Output ของ Counter และมีค่าเป็น '101' และในเวลาต่อมาเมื่อมี Pulse จ่ายให้ Counter อีก 1 ลูกค่า Output ของ Counter ก็จะมีค่าเป็น '110'

จากเนื้อหาที่กล่าวมาเราทำให้ Counter นับจำนวน Pulse ที่จ่ายให้กับ Counter ได้ ด้วยหลักการดังกล่าวเราสามารถนำ Counter มาประยุกต์ใช้งานได้หลายอย่าง เช่น ใช้เป็นวงจร Frequency divider, วงจร Timing delay เป็นต้น

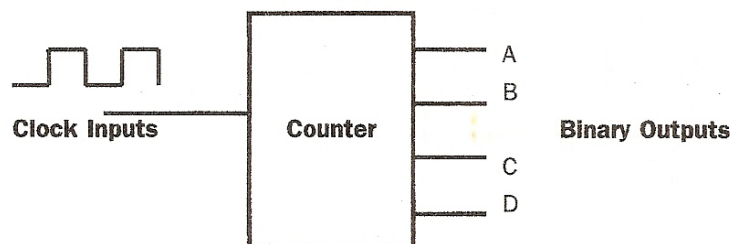


Figure 9.1 ลักษณะพื้นฐานของวงจร Counter

การทดลอง

9.1 ทดสอบหน้าที่การทำงานของวงจรนับเลขไบนารี (Ripple Counter)

Ripple Counter เป็น Counter แบบแรกๆ ไม่มีความยุ่งยากในการออกแบบ เพราะวงจรดังกล่าวถูกสร้างขึ้นโดยการนำ T Flip Flops จำนวนหลายๆ ตัวมาทำการต่ออนุกรมเข้าด้วยกันเท่านั้นการทำงานของ Counter ตัวดังกล่าวคือจะทำการนับจำนวน Pulse หรือ Clock ที่จ่ายเข้าสู่ Counter แล้วผลที่นับได้จะออกมาอยู่ในรูปของตัวเลข Binary เช่น ถ้า Counter ตัวหนึ่งมี Output จำนวน 4 เส้น (D, C, B และ A) เราสามารถจะทำการนับจำนวน Pulse ได้ตั้งแต่ '0000' ไปจนถึง '1111' (0 ถึง 15 ในเลข Decimal) โดยเราจะกำหนดว่า ในตำแหน่งของ A ให้เป็นตำแหน่งที่ 1S ของ Binary หรือเราจะเรียกตำแหน่งนี้อีกอย่างหนึ่งว่า "Least Significant Bit, LSB" ดังรูป 9.2

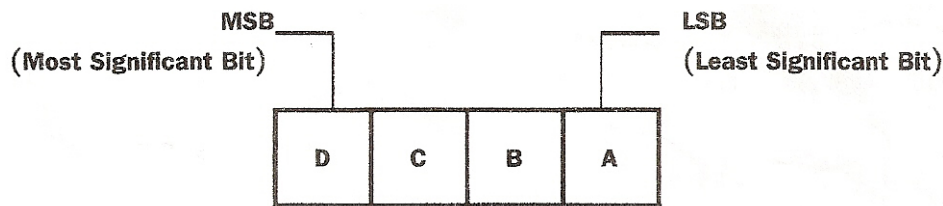


Figure 9.2 อธิบายถึงตำแหน่งของ MSB และ LSB

ในกรณีที่ Counter ตัวดังกล่าวมี Output จำนวน 4 ตัวนั้นแสดงว่าสามารถนับได้ตั้งแต่ '0000' ถึง '1111' ทำให้เกิดความแตกต่างทั้งหมด 16 แบบ เราจึงเรียก Counter แบบนี้ว่า "Modulo 16 counter" หรือเรียกสั้นๆ ว่า "MOD-16 Counter"

MOD-16 Counter นั้นสร้างขึ้นจาก JK Flip Flop จำนวน 4 ตัว โดย JK Flip Flop แต่ละตัวจะถูกทำให้อยู่ในสถานะของ T FF ($J=K=1$) ดังรูปที่ 9.3

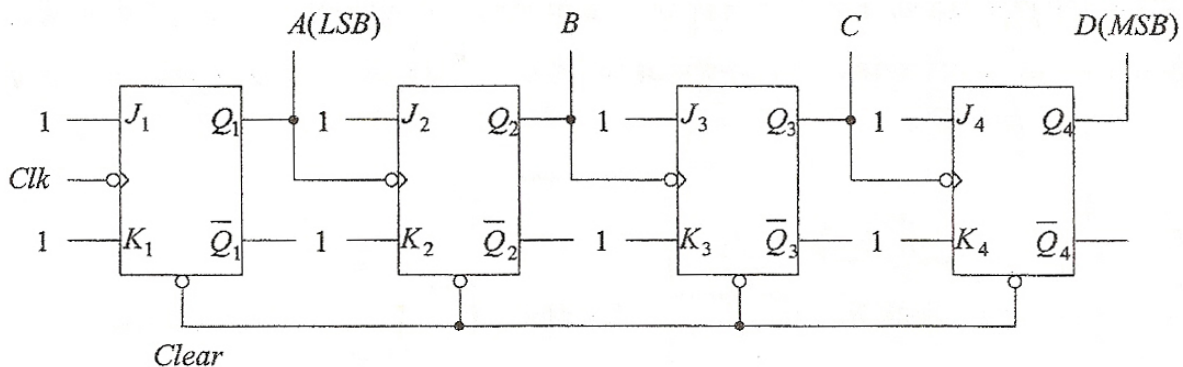


Figure 9.3 วงจรนับเลขไบนารี (Ripple Counter) แบบ MOD-16

ถ้าเราสมมุติว่าในสถานะเริ่มต้น Output ที่ได้จาก FF ทุกตัว เป็น '0' (Counter Output = '0000') เมื่อ Clock ลูกที่ 1 เข้ามายังวงจรนับ FF1 จะทำการ Toggle อีกครั้งทำให้ Output (Q) ของ FF1 เป็น '1' ผลคือ Output ของ Counter เป็น

‘0001’ ในเวลาต่อมาเมื่อ Clock ลูกที่ 2 เข้ามา FF1 จะ Toggle อีกครั้ง Output ที่ได้จาก FF1 จะกลายเป็น ‘0’ ในขณะที่ FF2 ก็ จะเกิดการ Toggle ขึ้นเช่นกัน ส่งผลใน Output ของ FF2 เปลี่ยนจาก ‘0’ เป็น ‘1’ ผลคือ Output ของ Counter จะเป็น ‘0010’ การทำงานจะเป็นลักษณะเช่นนี้ไปเรื่อยๆ จนถึง ‘1111’ จากนั้น Output ของ Counter ก็จะวนกลับมาเป็น ‘0000’ อีก ครั้ง

- ให้นักศึกษาทำการต่อวงจรตามรูปที่ 9.3 โดยใช้ IC 7473
- เมื่อต่อวงจรเป็นที่เรียบร้อยแล้วให้ทำการ Clear ข้อมูลของ FF ทุกตัวเพื่อให้ได้ Output เป็น ‘0’
- เริ่มทำการนับโดยการป้อน Clock ที่ละ 1 Pulse
- ตรวจสอบค่าที่อ่านได้จาก LED แล้วบันทึกผลลงในตารางที่ 9.1

จำนวนของ Clock	ค่าที่อ่านได้จาก Flip Flops				ค่าที่อ่านได้จาก 7-Segment
	D (MSB)	C	B	A (LSB)	
0					
1					
2					
3					
4					
5					
6					
7					
8					
9					
10					
11					
12					
13					
14					
15					

Table 9.1

9.2 ทดสอบหน้าที่การทำงานของวงจรนับสิบ (Decade Counter หรือ MOD-10 Counter)

วงจร MOD-10 Counter นั้นไม่ว่าจะเป็นรูปแบบการทำงาน หรือลักษณะพื้นฐานของวงจรเหมือนกับวงจร MOD-16 Counter เพียงแต่วงจร MOD-10 Counter ต้องการลดจำนวนของสถานะการเปลี่ยนแปลงของ Output จาก '0000' ไปจนถึง '1111' ในวงจร MOD-16 Counter ให้เหลือเพียง '0000' ไปจนถึง '1001' เท่านั้น

ดังนั้นเราจึงสามารถนำวงจร MOD-16 ดัดแปลงให้ได้วงจร MOD-10 ได้โดยการยึดหลักที่ว่าสถานะที่ต่อไปจากสถานะสูงสุด MOD-10 (สถานะสูงสุดของ MOD-10 คือ '1001') คือ '1010' ตามลักษณะการทำงานของ MOD-16 นั่นคือสำหรับ MOD-10 Counter แล้ว สถานะที่ '1010' จะต้องไม่เกิดขึ้นแต่ควรจะเป็น '0000' แทนเพื่อจะได้เริ่มทำการนับรอบใหม่ต่อไป

จากหลักการข้างต้นเราสามารถนำมอดออกมาเป็นวงจร MOD-10 ได้โดยการเพิ่ม NAND Gate เข้าไปในวงจร MOD-16 โดย Output ของ NAND Gate ต่อกับขา Clear ของ FF ทั้ง 4 ตัว ส่วนทางด้าน Input ต่ออยู่กับ Output ของ FF ที่ตำแหน่ง B และ D ของ Counter ตามรูปที่ 9.2 เพื่อคอยตรวจสอบ Output ของ Bit ทั้งสองซึ่งหากเป็น '1' ทั้งคู่ NAND จะทำการ Clear ค่าใน FF ทุกตัว ให้เริ่มทำการนับที่ '0000' ใหม่อีกครั้ง ดังนั้นวงจรดังกล่าวจึงสามารถนับได้จาก '0000' ไปจนถึง '1001' เท่านั้น ตามความต้องการของวงจร MOD-10 Counter

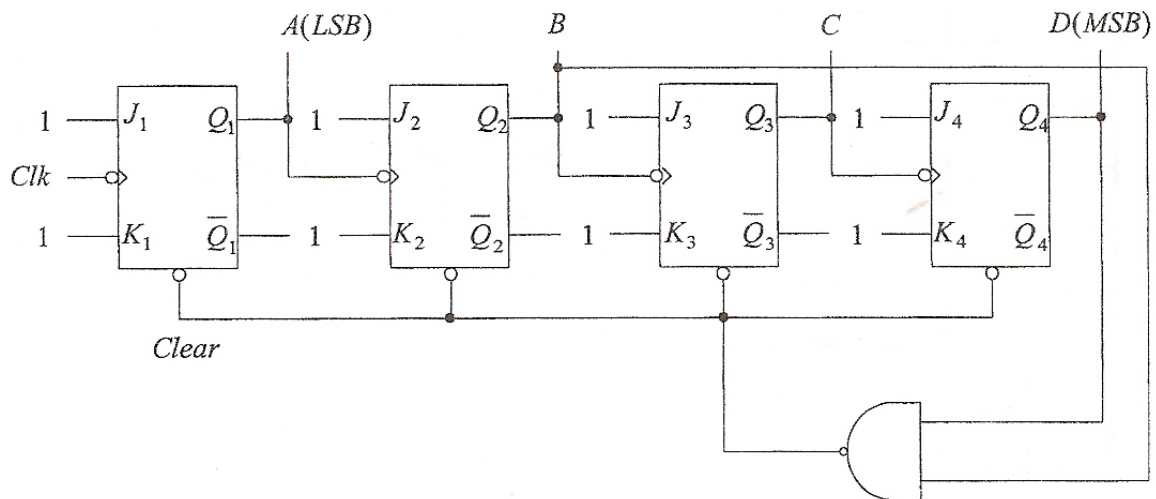


Figure 9.4 วงจร Ripple Counter แบบ MOD-10 (Decade Counter)

- ทำการต่อวงจรตามรูปที่ 9.4 โดยใช้ IC 7400 และ 7473
- ต่อวงจร encode outputs ของ FFs เพื่อแสดงผลบน 7-Segment display
- เมื่อต่อวงจรเป็นที่เรียบร้อยแล้วให้ทำการ Clear ข้อมูลของ FF ทุกตัวเพื่อให้ได้ Output เป็น "0"
- เริ่มทำการนับโดยการป้อน Clock ที่ละ 1 Pulse
- ตรวจสอบค่าที่อ่านได้จาก LED แล้วบันทึกผลลงในตารางที่ 9.2

จำนวนของ Clock	ค่าที่อ่านได้จาก Flip Flops				ค่าที่อ่านได้จาก 7-Segment
	D (MSB)	C	B	A (LSB)	
0					
1					
2					
3					
4					
5					
6					
7					
8					
9					
10					

Table 9.2

9.3 ทดสอบ Synchronous Counters

เราได้ศึกษา Ripper Counter มาถึง 2 การทดลองแล้ว ซึ่งวงจรดังกล่าวก็สามารถจะทำงานได้ แต่ในการใช้งานจริงที่ความถี่สูงๆ (High Frequency) นั้นวงจร Ripper Counter จะไม่สามารถทำงานได้อย่างมีประสิทธิภาพ เนื่องจาก Ripper Counter อาศัยการ Toggle กันเป็นทอดๆ จาก FF1 ไปยัง FF2 เรื่อยไปจนถึง FF ตัวสุดท้าย ซึ่งในความเป็นจริง FF แต่ละตัวจะใช้เวลาในการ Trigger และเริ่มทำงานที่ไม่เท่ากัน (ขึ้นกับ propagation delay time ของ FFs ตัวก่อนหน้า) และเวลาไม่คงที่ ดังนั้นเราเรียก Ripper Counter ว่าเป็น Counter แบบ Asynchronous Counter ปัญหาที่เกิดขึ้นก็คือที่ความถี่สูงๆ clock ลูกถัดไป อาจจะเข้ามาถึง FF ตัวแรก ก่อนที่ output จาก FF ตัวท้ายๆ ที่เกิดจาก clock ลูกก่อนหน้าจะปรากฏออกมา ทำให้การนับผิดพลาดได้

ทางแก้ปัญหาก็คือ เราต้องทำการออกแบบวงจรให้ FF ทุกตัวได้รับ clock จากแหล่งเดียวกัน ในเวลาที่พร้อมๆ กัน เราจึงเรียก Counter แบบใหม่นี้ว่า Synchronous Counter ซึ่ง Clock ที่จ่ายให้วงจรจะมีลักษณะการจ่ายแบบ Parallel โดยในการทดลองนี้จะทดลองวงจร Synchronous Counter แบบ MOD-8 ดังรูปที่ 9.5

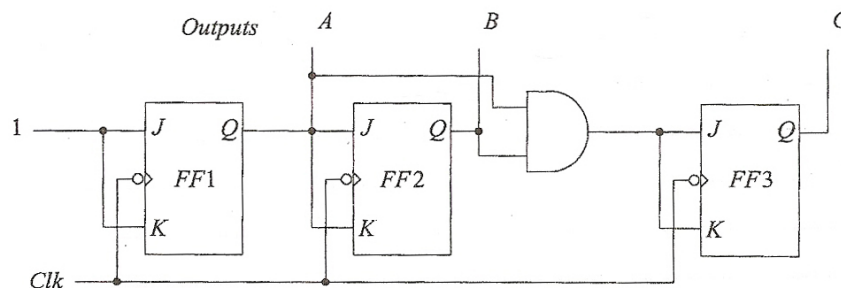


Fig 9.5 วงจร Synchronous Counter แบบ MOD-8

- ให้นักศึกษาทำการต่อวงจรตามรูปที่ 9.5 โดยใช้ IC 7473
- ให้นักศึกษาทำการต่อเชื่อมต่อ Clear ของ FF ทั้ง 3 ตัวเข้าด้วยกัน จากนั้นทำการ Clear ข้อมูลของ FF ทุกตัวให้เป็น “0”
- เริ่มทำการป้อน Clock ที่ละ 1 Pulse
- ตรวจสอบ Output ที่ได้จาก LED แล้วบันทึกผลลงในตารางที่ 9.3
- ให้นักศึกษาทำการเขียนอธิบายการทำงานของ Synchronous Counter เมื่อได้รับการป้อน Pulse ที่ละลูก

Number of Clock Pulses	Output Binary		
	C (MSB)	B	A (LSB)
0			
1			
2			
3			
4			
5			
6			
7			
8			
9			

Table 9.3

อธิบายหลักการทำงานของ Synchronous Counter แบบ MOD-8

9.4 ทดสอบวงจร Counter โดยใช้ IC สำเร็จรูป IC 7493 (4 Bits Binary Counter)

- ให้นักศึกษาใช้ Data Sheet ของ 7493 เพื่อออกแบบวงจร 4 Bits Counter
- จากนั้นทำการป้อน Clock ทีละ 1 Pulse
- ทำการตรวจสอบ Output ที่ได้จาก LED และ 7-Segment display แล้วบันทึกผลลงในตารางที่ 9.4

จำนวนของ Clock	ค่าที่อ่านได้จาก Flip Flops				ค่าที่อ่านได้จาก 7-Segment
	D (MSB)	C	B	A (LSB)	
0					
1					
2					
3					
4					
5					
6					
7					
8					
9					
10					
11					
12					
13					
14					
15					

Table 9.4

คำถามท้ายการทดลอง

1. จงเขียน Timing Diagram ของ Q_d Q_c Q_b และ Q_a ของวงจร MOD-16 Counter เมื่อเทียบกับ Clock โดยกำหนดให้ bit 'd' เป็น MSB และ bit 'a' เป็น LSB ตามลำดับ
2. จากวงจรข้อ 9.4 ถ้าต้องการเปลี่ยนเป็นวงจร MOD-6 ต้องต่อ 7493 อย่างไร
3. จงออกแบบวงจร MOD-12 แบบ Asynchronous Counter พร้อมทั้งอธิบายการทำงานของวงจร
4. จงออกแบบวงจร MOD-10 แบบ Synchronous Counter พร้อมทั้งอธิบายการทำงานของวงจร

ส่งคำตอบของคำถามท้ายการทดลองทั้ง 4 ข้อ มาที่อีเมล: kanat@egco.org