

เนื้อหา

Algorithmic State Machine

Algorithm = วิธีการแก้ปัญหาโดยอธิบายอย่างเป็นขั้นตอน สามารถอธิบายโดยใช้ตั้งแต่ภาษา
ระดับสูงถึงระดับต่ำหรืออาจเขียนเป็นลักษณะของรูปภาพ คือ Flowchart ซึ่งสามารถนำมาใช้
อธิบายวงจรได้เช่นกัน

State Machine = คือวงจร Synchronous Sequential นั่นเอง เป็นวงจรที่มี State เป็นตัวกำหนด
การทำงานการเปลี่ยน State และการสร้าง O/P ขึ้นอยู่กับ State และ I/P ของ วงจร

Algorithmic State Machine = State Machine ที่มีการทำงานอย่างเป็นขั้นตอนวิธีและเงื่อนไข
สามารถอธิบายการทำงาน โดยใช้ Algorithm

Algorithmic State Machine Method

Algorithmic State Machine นั้น จะมีการทำงานอย่างมีเงื่อนไขและขั้นตอนวิธี และจะ
อธิบายการทำงาน โดยใช้ Algorithm

ระบบชื่อสัญญาณของ ASM

ชื่อสัญญาณ I/P,O/P ของ ASM จะใช้ตัวอักษร 3-4 ตัวอักษร ซึ่งอาจสื่อความหมายได้ เช่น

RST สำหรับการ Reset เครื่อง

CLR สำหรับการ Clear เครื่อง

และอาจมีการบอก Active Logic (โดยใช้ H, Y, 1 และ L, N, 0) นำหน้าชื่อสัญญาณ เช่น

H.RST หมายถึง ทำการ Reset เมื่อ RST = 1

L.RST หมายถึง ทำการ Reset เมื่อ RST = 0

N.RED หมายถึง สัญญาณ RED Active ที่ลอจิก 0

Y.RED หมายถึง สัญญาณ RED Active ที่ลอจิก 1

O/P line อาจมี I นำหน้า หมายถึง Immediate Output ก็จะมีการเปลี่ยนแปลงได้ใน
ช่วงเวลาของ State ตามเงื่อนไขของ I/P หากมีการเปลี่ยน I/P อาจทำให้ค่าของ Immediate O/P นี้
เปลี่ยนแปลงได้

ถ้า O/P ใดไม่มี I นำหน้า หมายถึง O/P นั้นขึ้นอยู่กับ State เพียงอย่างเดียว การเปลี่ยนค่า
O/P นั้นเกิดขึ้นเมื่อมีการเปลี่ยน State เท่านั้น

การเขียน ASM Chart

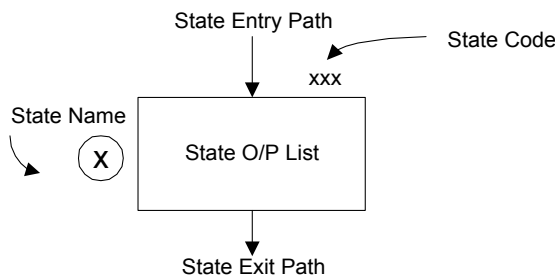
การเขียน Algorithm สำหรับ State Machine นั้น แบบที่ง่ายที่สุดคือ ASM chart ซึ่งมีลักษณะคล้าย Flowchart ต่างกันตรงที่ FlowChart จะเป็นการทำงานแบบต่อเนื่อง ตามลำดับ ทีละ Block จนจบ หาก Block หนึ่งมีหลายคำสั่ง จะทำตามลำดับก่อนหลัง แต่ ASM chart นั้น จะเป็นการทำงานแบบไม่ต่อเนื่อง (Digital) ช่วงเวลาของ State หนึ่งๆ เครื่องจะอยู่ที่ Block เดียว และจะเปลี่ยน Block ใหม่ทุกช่วงเวลา และทุกอย่างใน Block จะทำพร้อมกันหมด

ASM chart จะเป็น Diagram ที่แสดงถึง State Transition Function และ O/P Function ของ State Machine ซึ่งจะนำไปใช้ในการออกแบบวงจรและเป็นเอกสารส่วนหนึ่งในการอธิบายวงจร

ASM chart จะประกอบไปด้วยหลายๆ ASM block ซึ่ง 1 ASM Block จะแสดงการทำงานใน 1 State ทุกอย่างใน Block นั้นถือว่าทำงานพร้อมกันหมดในแต่ละ ASM Block ประกอบไปด้วย

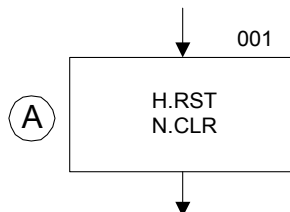
1. State Box
2. Decision Box
3. Condition O/P Box

1. State Box มีสัญลักษณ์ดังรูปที่ 1



รูปที่ 1 State Box

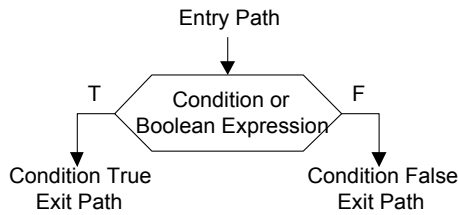
ชื่อ State มักเป็นตัวอักษรหรือตัวเลข ส่วนรหัสสถานะ เป็นเลขไบนารีที่กำหนดขึ้นใช้แทน State นั้นๆ ภายในจะมีชื่อสัญญาณ O/P ที่ Active ดังรูปที่ 2



รูปที่ 2 ตัวอย่าง State Box

จากรูปที่ 2 ที่ State A นี้ H.RST = 1 และ N.CLR = 0

2. Decision Box มีลักษณะดังรูปที่ 3

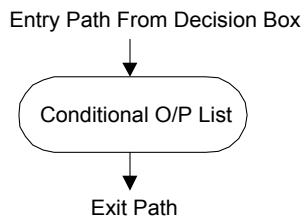


รูปที่ 3 Decision Box

Decision จะมี 1 ทางเข้า ภายใน Box จะเป็นเงื่อนไข (Condition) ในรูปของ Boolean Expression ส่วนทางออกจะมี 2 ทาง ซึ่งกำกับด้วย T, Y, 1 และ F, N, 0

- ทางออกซึ่งกำกับด้วย T, Y, 1 จะเป็นเส้นทางเมื่อเงื่อนไขเป็นจริง
- ทางออกซึ่งกำกับด้วย F, N, 0 จะเป็นเส้นทางเมื่อเงื่อนไขเป็นเท็จ

3. Conditional O/P Box มีลักษณะดังรูปที่ 4

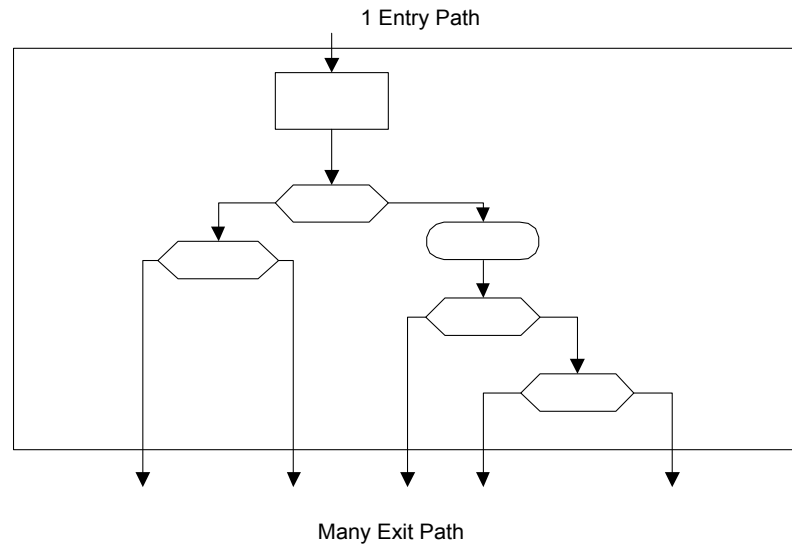


รูปที่ 4 Conditional O/P Box

Conditional O/P Box จะมีทางเข้าซึ่งออกมาจาก Decision Box เท่านั้น ภายใน Box จะเป็นรายชื่อ O/P ที่ Active

เนื่องจาก ทางเข้าของ Conditional O/P Box มาจาก Decision Box นั่นคือ รายชื่อ O/P ใน Conditional O/P box นี้จะ Active เมื่อได้รับผลอย่างใดอย่างหนึ่ง (จริง หรือ เท็จ) จาก Decision Box นั้น นั่นก็หมายถึง O/P เหล่านี้จะขึ้นอยู่กับทั้ง State และ I/P ที่ใช้ทดสอบเงื่อนไขใน Decision Box นั้น

ภายใน 1 ASM Block ต้องประกอบไปด้วย 1 State Box เท่านั้น และจะมี decision Box และ Conditional O/P Box ก็ Box ก็ได้ หรืออาจไม่มีก็ได้ ดังรูปที่ 5



รูปที่ 5 แสดง ASM Block

State Box จะเป็นตัวชี้ ASM Block นั่นคือ State Box จะเป็นตัวแบ่ง แต่ละ ASM Block ออกจากกันนั่นเอง

1 ASM Block จะมี 1 ทางเข้า หลายทางออก แต่ละทางออกจะต้องไปสู่ ASM Block อื่น คือไปยังทางเข้าของ State Box อื่นนั่นเอง แต่ละเส้นทางที่ไปสู่ State อื่นนี้ เรียกว่า Link Path

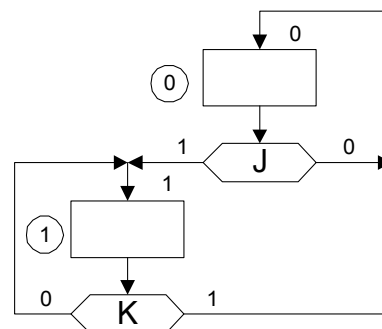
ตัวอย่างการใช้ ASM Chart กับวงจร Flip-Flop

Flip-Flop เป็นอุปกรณ์ที่ใช้เป็น Memory ให้กับ State Machine หรือวงจร Sequential ตัว Flip-Flop เองก็เป็นวงจร Sequential เช่นกัน โดยจะมี 2 สภาวะ คือ State 1 กับ State 0

JK Flip-Flop มี Function Table ดังรูปที่ 6a และมี ASM Chart ดังรูปที่ 6b

J	K	$Q_n + 1$
0	0	Q_n
0	1	0
1	0	1
1	1	Q_n

(a)



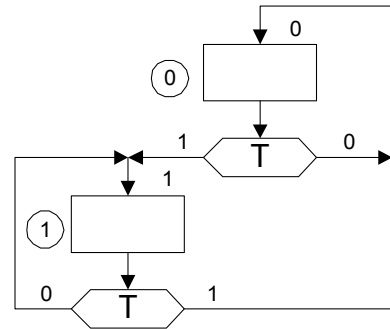
(b)

รูปที่ 6 JK Flip-Flop (a) Function Table (b) ASM Chart

T Flip-Flop มี Function Table ดังรูปที่ 7a และมี ASM Chart ดังรูปที่ 7b

T	Q_{n+1}
0	Q_n
1	$\overline{Q_n}$

(a)



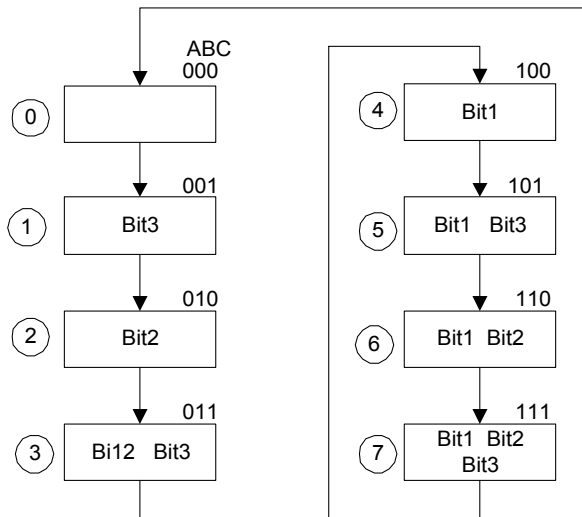
(b)

รูปที่ 7 T Flip-Flop (a) Function Table (b) ASM Chart

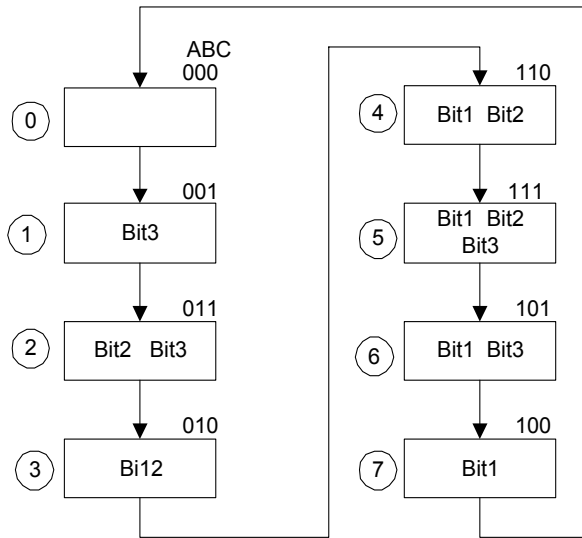
ตัวอย่างการใช้ ASM Chart กับวงจรนับ

ตัวอย่างที่ 1 3 Bit Binary Counter

เป็นวงจรที่ไม่ขึ้นอยู่กับ I/P ใดเลย O/P ขึ้นอยู่กับ State เท่านั้น เมื่อมีการเปลี่ยน State, State ถัดไปจะเป็น



รูปที่ 8 ASM Chart ของวงจร 3 Bit Binary Counter



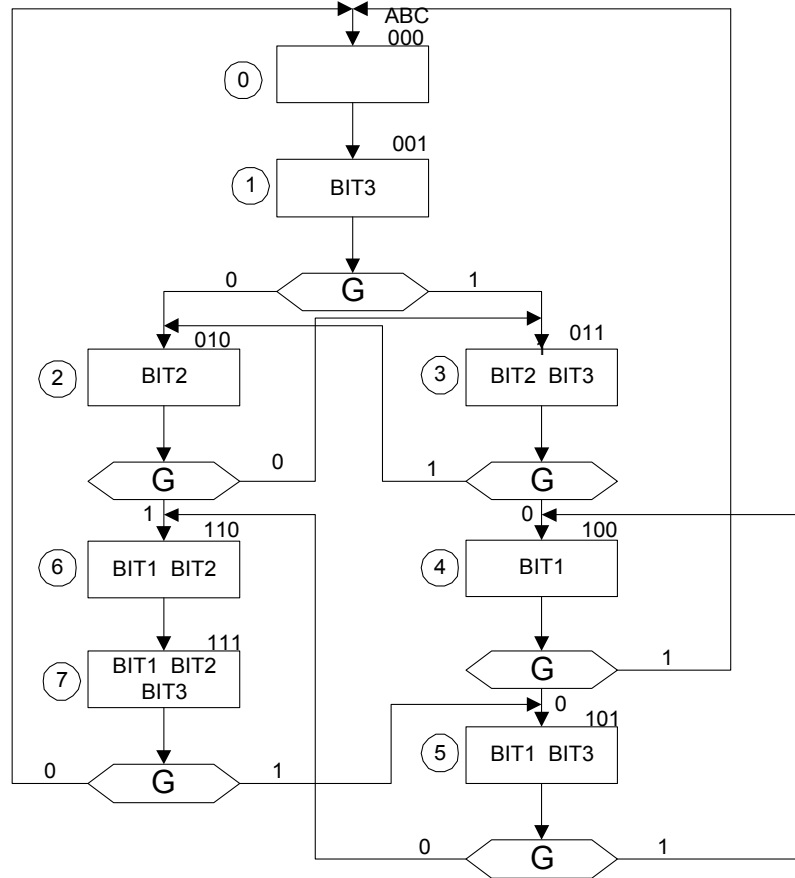
รูปที่ 9 ASM Chart ของวงจร 3 Bit Gray-Code Counter

วงจรนี้ก็เช่นเดียวกัน เป็น State Machine แบบ Moore ที่ O/P ขึ้นอยู่กับ State เท่านั้น

ตัวอย่างที่ 2 3 bit Binary/Gray code Counter

วงจรนี้จะเริ่มมีเงื่อนไข ซึ่งจะเป็นตัวกำหนดว่าวงจรจะนับ Binary หรือ Gray ซึ่งจะทำให้ ASM Chart ต้องมี Decision Box เข้ามาเป็นตัวตัดสินใจ

ถ้าเงื่อนไขคือ G=0 นับ Binary, ถ้า G=1 นับ Gray Code จะได้ ASM Chart ดังรูปที่ 10



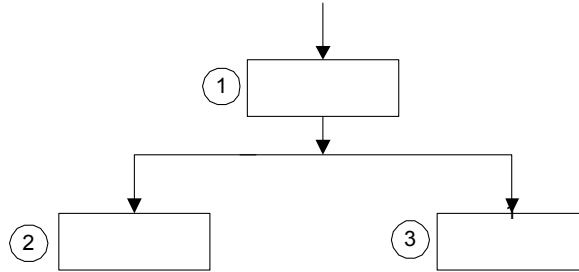
รูปที่ 10 ASM Chart ของวงจร 3-bit Binary/Gray code Counter

จะเห็นว่าวงจรนี้ Next State จะขึ้นอยู่กับ I/P G เช่นที่ State 1 O/P = 001 ถ้า G=0 นับ Binary จะทำให้ Next State เป็น State 2 O/P = 010 แต่หาก G=1 จะทำให้ Next State เป็น State 3 O/P = 011

เราให้เห็นตัวอย่างในการเขียน ASM Chart อย่างง่าย ๆ ไม่ซับซ้อนมาเล็กน้อยแล้ว ก่อนที่จะเริ่มต้นการออกแบบ ASM จริงๆ เราควรจะเรียนรู้วิธีการเขียน ASM Chart ที่ถูกต้อง ซึ่งมีรายละเอียดบางประการที่ควรระมัดระวังในการเขียน ASM Chart เช่น ในแต่ละ State จะเปลี่ยนไปยัง Next State ได้เพียง 1 State เท่านั้นใน 1 เงื่อนไข

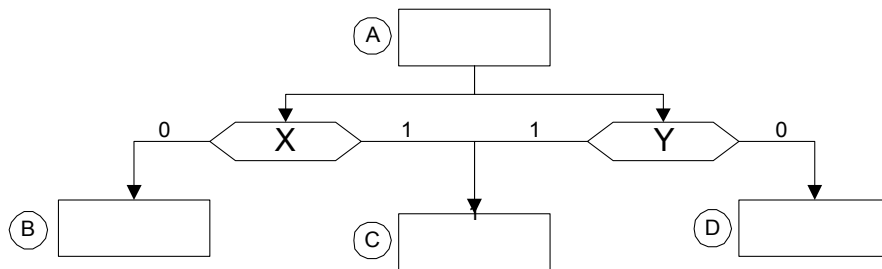
ตัวอย่าง ASM Chart ที่ผิด

ตัวอย่างที่ 1 ผิดที่มีการวิ่งไป 2 Next State



รูปที่ 11 ตัวอย่าง ASM Chart ที่ผิด

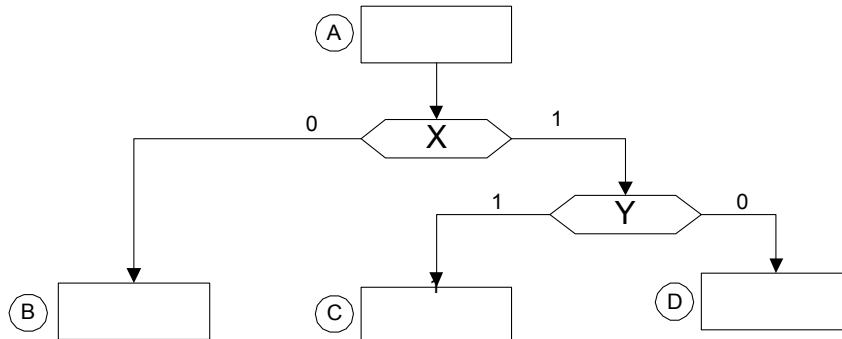
ตัวอย่างที่ 2 ผิดที่มีการวิ่งไป 2 Next State



รูปที่ 12 ตัวอย่าง ASM Chart ที่ผิด

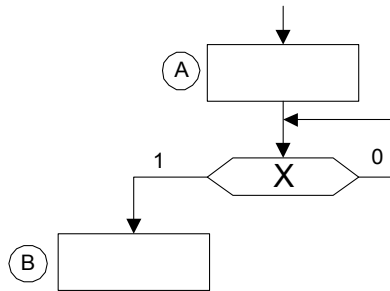
จะเห็นว่าการเช็คเงื่อนไขโดยใช้ X และ Y หาก X=0 และ Y=0 ในเวลาเดียวกัน จะมีการวิ่งไป 2 State คือ B และ D หรือ X=0, Y=1 ก็วิ่งไป 2 State คือ B และ C

ความจริง ผู้ออกแบบอาจต้องการให้เป็นอย่างรูปที่ 13



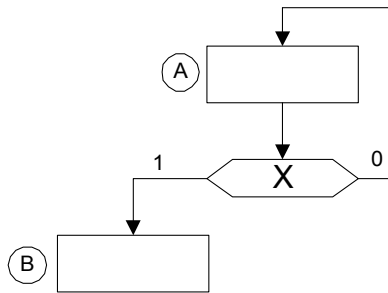
รูปที่ 13 ตัวอย่าง ASM Chart ที่ถูก

ตัวอย่างที่ 3 ผิดเพราะมีวงรอบใน ASM Block



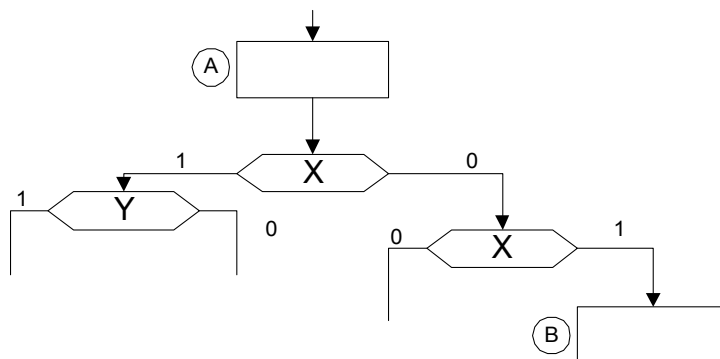
รูปที่ 14 ตัวอย่าง ASM Chart ที่ผิด

หากต้องการให้มีวงรอบมาที่ State A หาก $x=0$ ต้องเขียนดังรูปที่ 15



รูปที่ 15 ตัวอย่าง ASM Chart ที่ถูก

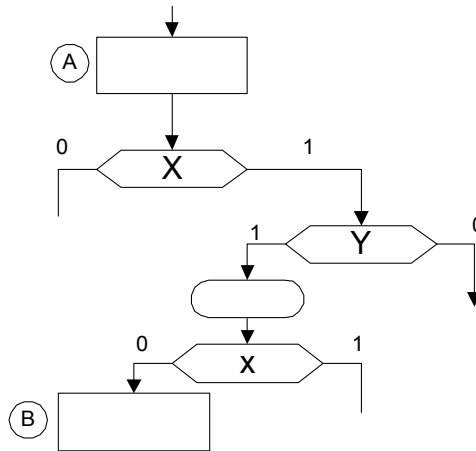
ตัวอย่างที่ 4 ผิดเพราะมีเส้นทางที่ไม่สามารถวิ่งไปได้



รูปที่ 16 ตัวอย่าง ASM Chart ที่ผิด

จะเห็นว่า Link Path จาก State A ไป B ผ่าน Condition แรกคือ $X=0$ แต่ Condition ที่ 2 คือ $X=1$ ซึ่งเป็นไปไม่ได้ที่ X จะเป็น 0, 1 ในเวลาเดียวกัน

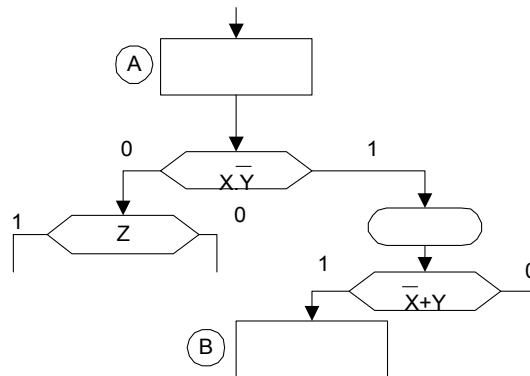
ตัวอย่างที่ 5 มีเส้นทางที่ไม่สามารถวิ่งไปได้



รูปที่ 17 ตัวอย่าง ASM Chart ที่ผิด

Link Path จาก A ไป B เงื่อนไขคือ $x=1, y=1, x=0$ ซึ่งไม่สามารถเป็นไปได้

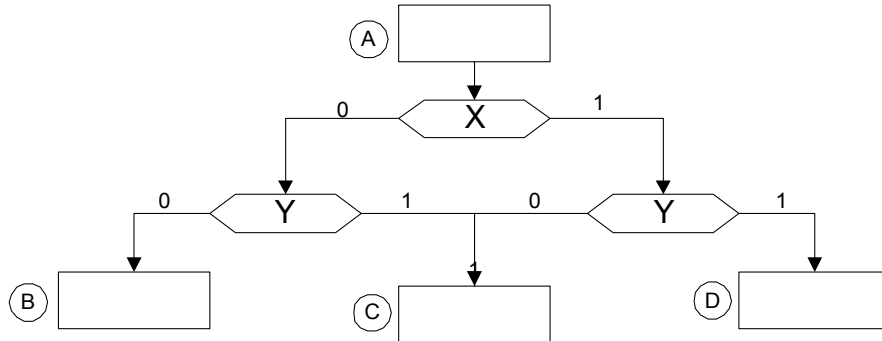
ตัวอย่างที่ 6 มีเส้นทางที่ไม่สามารถวิ่งไปได้



รูปที่ 18 ตัวอย่าง ASM Chart ที่ผิด

Link Path จาก A ไป B ผ่าน Condition ที่ 1 คือ $X \cdot Y = 1$ คือ $X=1, Y=0$ และ Condition ที่ 2 คือ $X + Y = 1$ ซึ่ง $X=0, Y=1$ ซึ่งเป็นไปไม่ได้ เพราะ Condition ทั้ง 2 ไม่สามารถเป็นจริงในเวลาเดียวกัน

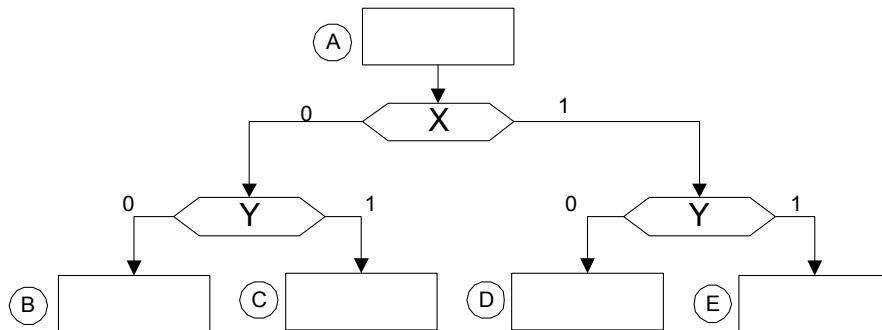
ตัวอย่างการใช้ตัวแปรหลายทีในโครงข่ายการตัดสินใจอย่างถูกต้อง
ตัวอย่างที่ 1



รูปที่ 19 ตัวอย่าง ASM Chart

ถึงจะมีตัวแปร Y ใน 2 Decision Box แต่ Box ซ้ายมือจะใช้กรณี X=0, ขวามือใช้กรณี X=1

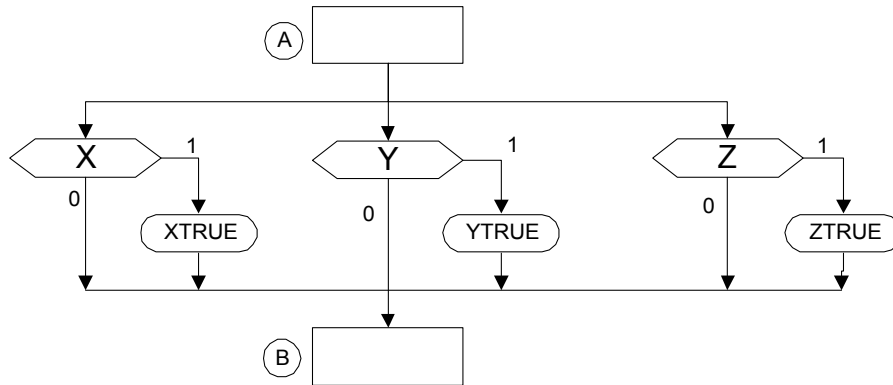
ตัวอย่างที่ 2



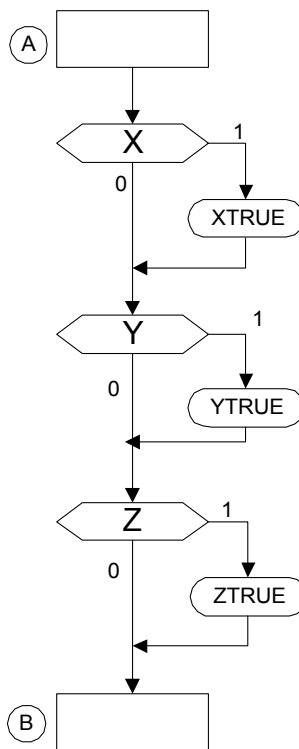
รูปที่ 20 ตัวอย่าง ASM Chart

โครงสร้างแบบขนานและอนุกรม

ในแต่ละ ASM Block โครงข่ายการตัดสินใจไม่มีข้อผูกมัดทางเวลา การทำงานของ Decision ทุก Box จะพร้อมกันหมด ไม่ว่าในการเขียนจะอยู่ในลำดับก่อนหรือหลัง ดังนั้นโครงสร้างแบบขนานหรือแบบอนุกรมก็สามารุใช้แทนกันได้



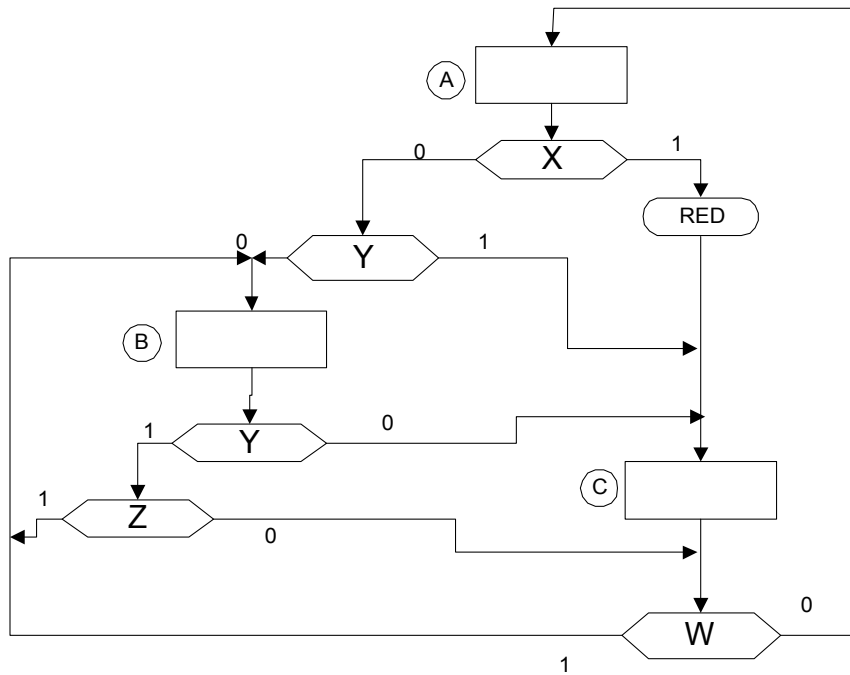
รูปที่ 21 ตัวอย่าง โครงข่ายการตัดสินใจแบบขนาน



รูปที่ 22 ตัวอย่าง โครงข่ายการตัดสินใจแบบอนุกรม

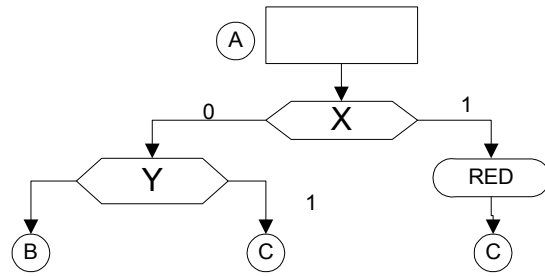
การใช้ Decision Box ร่วมกันของหลาย ASM Block

บางครั้งเพื่อความกระชับขอ ASM Chart อาจมีการใช้โครงข่ายการตัดสินใจร่วมกันของหลาย ASM Block เช่นรูปที่ 23



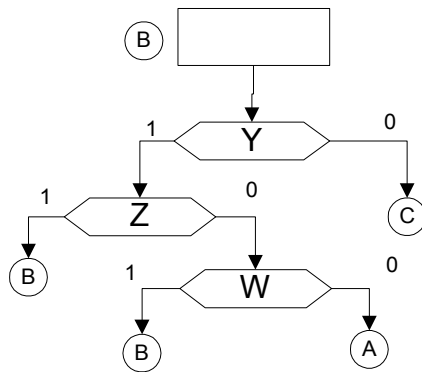
รูปที่ 23 ASM Chart ที่มีหลาย State ใช้ Decision Box ร่วมกัน

ใน ASM Chart นี้ State B, C ใช้ State Box ต่างสุดร่วมกัน ถึงแม้จะมีการใช้ Decision Box ร่วมกัน เรายังสามารถเข้าใจ Link Path ทั้งหมดได้ ดังรูปที่ 24



$$A \rightarrow B = \bar{X}Y$$

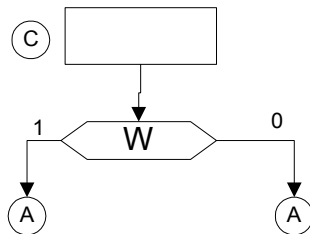
$$A \rightarrow C = X + \bar{X}Y$$



$$B \rightarrow B = YZ + Y\bar{Z}W$$

$$B \rightarrow A = Y\bar{Z}\bar{W}$$

$$B \rightarrow C = \bar{Y}$$



$$C \rightarrow B = W$$

$$C \rightarrow A = \bar{W}$$

รูปที่ 24 Link Path ในแต่ละ ASM Block ของรูปที่ 23

ตัวอย่างการออกแบบ ASM Chart

ตัวอย่างที่ 1

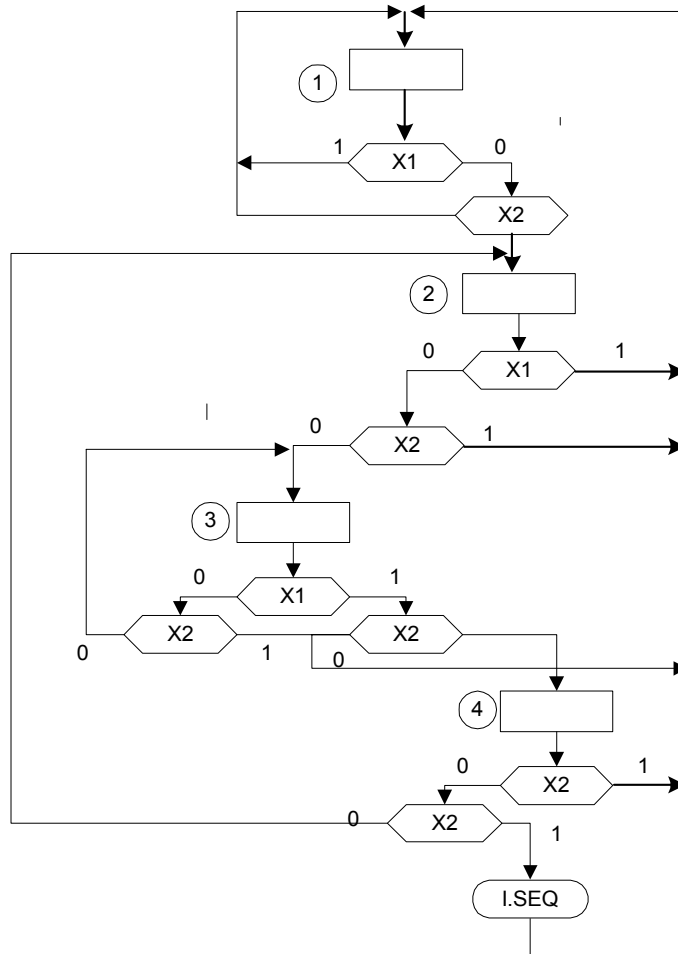
จงออกแบบวงจร Sequence Detector ของข้อมูล 2 บิต วงจรมี 2 I/P คือข้อมูล 2 บิตนั้น และ 1 O/P ซึ่งจะเป็น 1 เมื่อ I/P มีลำดับเป็น 00, 00, 11, 10 นอกจากนี้ O/P จะเป็น 0 เช่น

I/P Sequence x1 = 0 0 0 1 1 0 0 1 0 0 1 1

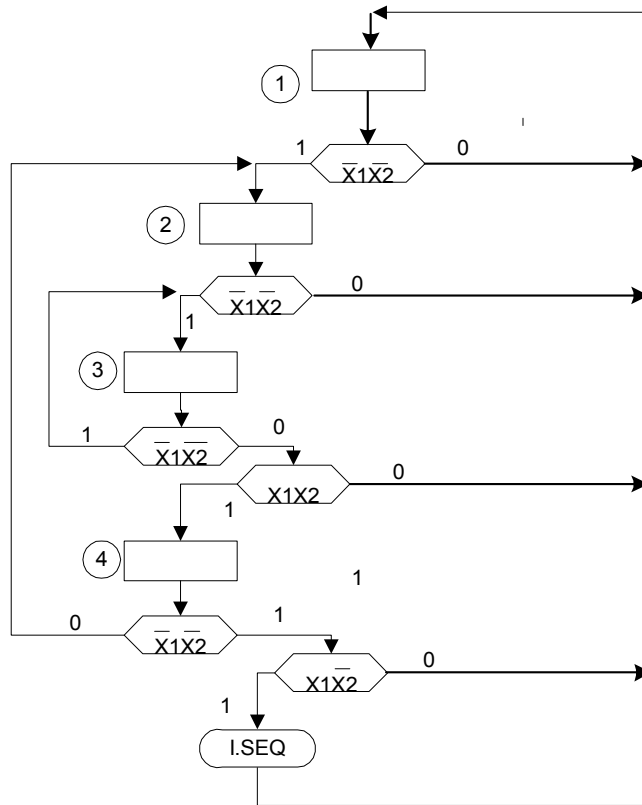
x2 = 0 1 0 0 1 0 0 0 0 0 0 1 0

O/P Sequence SEQ = 0 0 0 0 0 1 0 0 0 0 0 0 0

เขียนเป็น ASM Chart ได้ดังรูปที่ 25



รูปที่ 25 ASM Chart ของวงจร 00, 00, 01, 10 Sequence Detector หรือแบบรูปที่ 26 ก็ได้

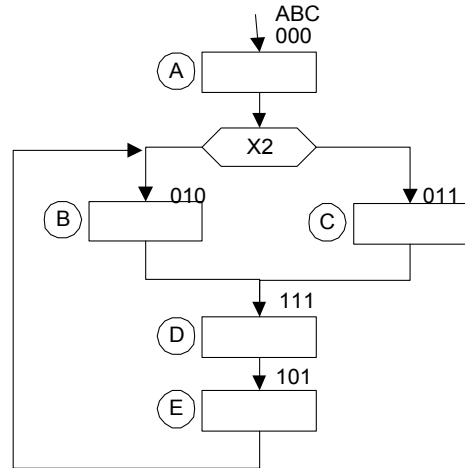


รูปที่ 26 ASM Chart ของวงจร 00, 00, 01, 10 Sequence Detector

State Assignment

การกำหนด State code มีหลักสำคัญคือ กำหนดให้แต่ละ State มี code ของตนเอง โดยไม่ซ้ำกับ State อื่น ดูตัวอย่างการทำ State Assignment ดังรูปที่ 27

State Name	State Code		
	A	B	C
A	0	0	0
B	0	1	0
C	0	1	1
D	1	1	1
E	1	0	1



(a)

(b)

รูปที่ 27 (a) การกำหนดรหัส State (b) ASM Chart

การกำหนดรหัสของ State ในแต่ละแบบ จะให้ NS Function และ O/P Function ที่ต่างกัน และมีขนาดและความซับซ้อนต่างกันด้วย ผู้ออกแบบย่อมต้องการให้วงจรที่ได้มีขนาดเล็ก มีฟังก์ชันที่ไม่ซับซ้อน

วิธีการกำหนดรหัสของ State ให้ได้ฟังก์ชันที่มีความซับซ้อนน้อยนั้น ได้กล่าวถึงไปแล้วในเรื่องของการออกแบบวงจร Sequential แต่ในส่วนของ ASM นี้จะเป็นวงจรที่มีขนาดใหญ่ จะบอกถึงวิธีการกำหนดรหัสที่ง่ายอีกแบบหนึ่ง คือวิธี Minimum State Locus

วิธีทำก็คือ เขียนลำดับของ Link Path ทั้งหมดออกมา แล้วทดลองแทนชื่อ State ด้วย State Code แบบต่างๆ แล้วหาจำนวนบิตที่เปลี่ยนค่าในแต่ละ Link Path แล้วนำมาบวกกัน เพื่อหา State Locus แล้วเลือกเอาแบบที่มี State Locus น้อยที่สุด

เช่น จากตัวอย่างก่อนหน้านี มีทั้งหมด 6 Link Path ซึ่งจะได้ State Locus ดังนี้

A → B	=	000 →	010	=	1 บิต	
A → C	=	000 →	011	=	2 บิต	
B → D	=	010 →	111	=	2 บิต	
C → D	=	011 →	111	=	1 บิต	
D → E	=	111 →	101	=	1 บิต	
E → B	=	101 →	010	=	3 บิต	
State Locus					=	10 บิต

ที่นี้ลองกำหนด State Code อีกแบบหนึ่ง ดังรูปที่ 28

State Name	State Code		
	A	B	C
A	0	0	0
B	0	1	0
C	1	0	0
D	1	1	0
E	1	1	1

แบบที่ 2 นี้ จะได้ State Locus ดังนี้

A → B	=	000 →	010	=	1 บิต	
A → C	=	000 →	100	=	1 บิต	
B → D	=	010 →	110	=	1 บิต	
C → D	=	100 →	110	=	1 บิต	
D → E	=	110 →	111	=	1 บิต	
E → B	=	111 →	010	=	2 บิต	
State Locus					=	7 บิต

ถือว่าเป็น Minimum State Locus

การสร้างตาราง ASM และการหา NS Function และ O/P Function

ในการออกแบบ ASM นั้น หลังจากเขียน ASM Chart แล้ว ขั้นตอนต่อไปคือเขียน ASM

Table

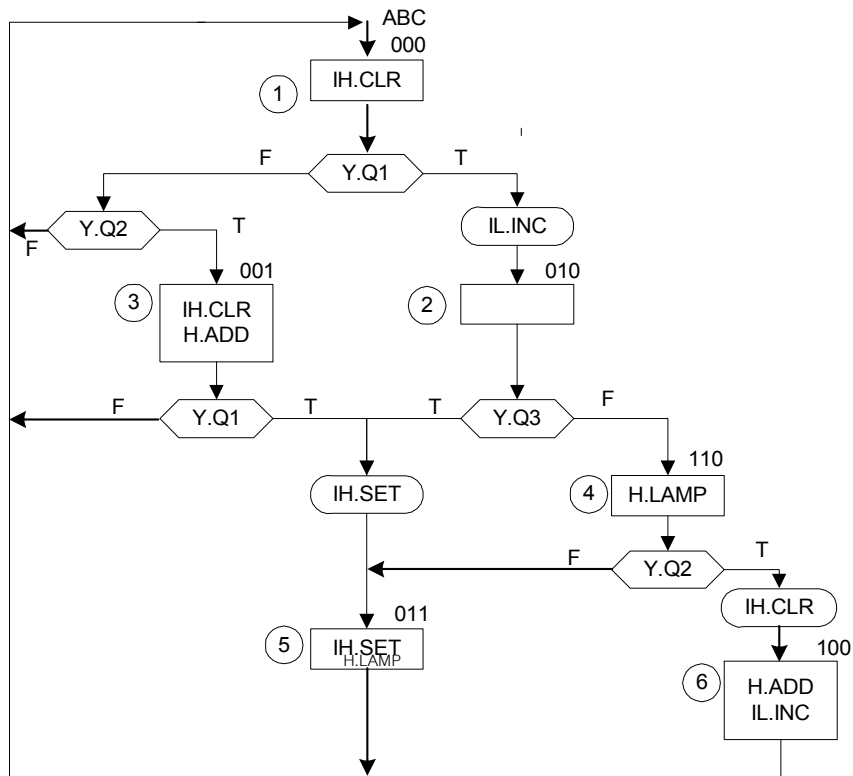
ASM Table นั้น จะมี 3 Table คือ

1. Next State Table หรือ State Transition Table
2. State O/P Table
3. Conditional O/P Table

แต่ละ State มี 2 รูปแบบคือ Symbolic Form และ Assigned Form

ตัวอย่าง

มี ASM Chart ดังรูปที่ 29



รูปที่ 29 ASM Chart

วงจรมี 3 I/P คือ Q1, Q2, Q3 ซึ่ง Active High ทั้งหมด

5 O/P คือ CLR, SET เป็น immediate และ Active High

INC เป็น immediate และ Active Low

ADD, LAMP เป็น State O/P และ Active High

วงจรมีใช้ A, B, C เป็น State Variable

ก่อนอื่นเราต้องหา Link Path ทั้งหมดใน ASM Chart

ให้ B = ASM Block และ L = Link Path เราจะเห็นว่าวงจรมี 6 ASM Block และ 11

Link Path

B1L1	:	1	→	1	โดย	Q1Q2
B1L2	:	1	→	2	โดย	Q1
B1L3	:	1	→	3	โดย	Q1Q2
B2L4	:	2	→	4	โดย	Q3
B2L5	:	2	→	5	โดย	Q3
B3L6	:	3	→	1	โดย	Q1
B3L7	:	3	→	5	โดย	Q1
B4L8	:	4	→	5	โดย	Q2
B4L9	:	4	→	6	โดย	Q2
B5L10	:	5	→	1	โดย	-
B6L11	:	6	→	1	โดย	-

จากนั้นเราก็เริ่มสร้างตารางแต่ละตาราง

1. Next State Table

มี 3 ส่วน คือ

- ส่วนที่ 1 เป็น I/P ของวงจร
- ส่วนที่ 2 เป็น Present State
- ส่วนที่ 3 เป็น Next State

จำนวนแถวจะมีเท่ากับจำนวน Link Path แต่ละแถวหมายถึงเงื่อนไขในแต่ละ Link Path

จาก ASM Chart รูปที่ 29 สามารถเขียน NS Table ได้ดังตารางที่ 1

ตารางที่ 1 Next State Table

(a) Symbolic Form

(b) Assigned Form

I/P			PS	NS	Comment
Y.Q1	Y.Q2	Y.Q3			
0	0	-	1	1	B1L1
1	-	-	1	2	B1L2
0	1	-	1	3	B1L3
-	-	0	2	4	B2L4
-	-	1	2	5	B2L5
0	-	-	3	1	B3L1
1	-	-	3	5	B3L5
-	0	-	4	5	B4L5
-	1	-	4	6	B4L6
-	-	-	5	1	B5L1
-	-	-	6	1	B6L1

I/P			PS			NS		
Q1	Q2	Q3	A	B	C	NA	NB	NC
0	0	-	0	0	0	0	0	0
1	-	-	0	0	0	0	1	0
0	1	-	0	0	0	0	0	1
-	-	0	0	1	0	1	1	0
-	-	1	0	1	0	0	1	1
0	-	-	0	0	1	0	0	0
1	-	-	0	0	1	0	1	1
-	0	-	1	1	0	0	1	1
-	1	-	1	1	0	1	0	0
-	-	-	0	1	1	0	0	0
-	-	-	1	0	0	0	0	0

NS Table ในรูปของ Symbolic form นั้นจะแทนค่า I/P ด้วยสัญลักษณ์ และ PS, NS เขียนโดยใช้ชื่อ State ส่วนในรูปของ Assign form นั้น จะแทนค่า I/P ต่างๆด้วยตัวเลข และเขียน PS, NS ด้วย State code

NS ในรูปของ Assign form นั้น สามารถที่จะนำไปสร้าง NS excitation table เพื่อที่จะหา NS Function ได้ดังนี้

เช่น

-ถ้าใช้ D flip-flop เขียน Excitation Table ได้ดังตารางที่ 2

ตารางที่ 2 Excitation Table

I/P			PS			NS			Memory I/P		
Q1	Q2	Q3	A	B	C	NA	NB	NC	DA	DB	DC
0	0	-	0	0	0	0	0	0	0	0	0
1	-	-	0	0	0	0	1	0	0	1	0
0	1	-	0	0	0	0	0	1	0	0	1
-	-	0	0	1	0	1	1	0	1	1	0
-	-	1	0	1	0	0	1	1	0	1	1
0	-	-	0	0	1	0	0	0	0	0	0
1	-	-	0	0	1	0	1	1	0	1	1
-	0	-	1	1	0	0	1	1	0	1	1
-	1	-	1	1	0	1	0	0	1	0	0
-	-	-	0	1	1	0	0	0	0	0	0
-	-	-	1	0	0	0	0	0	0	0	0

หา NS Function ได้ดังรูปที่ 30

		BC			
		A	00	01	11
DA	0	0	0	0	$\overline{Q3}$
	1	0	-	-	$Q2$

$$DA = ABQ2 + \overline{ABC}Q3$$

		BC			
		A	00	01	11
DB	0	$\overline{Q1}$	$\overline{Q1}$	0	1
	1	0	-	-	$\overline{Q2}$

$$DB = \overline{AB}Q1 + \overline{ABC} + ABQ3$$

DC		BC			
		A	00	01	11
0	1	$\overline{Q1}Q2$	$Q1$	0	$Q3$
		0	-	-	$\overline{Q2}$

$$DC = \overline{A}\overline{B}\overline{C}Q1Q2 + \overline{B}CQ1 + \overline{A}B\overline{C}Q3 + AB\overline{Q2}$$

รูปที่ 30 NS Map และ NS Function

- ถ้าให้ JK Flip-Flop เขียน Excitation Table ได้ดังตารางที่ 3

ตารางที่ 3 Excitation Table

I/P			PS			NS			Memory I/P					
Q1	Q2	Q3	A	B	C	NA	NB	NC	JA	KA	JB	KB	JC	KC
0	0	-	0	0	0	0	0	0	0	-	0	-	0	-
1	-	-	0	0	0	0	1	0	0	-	1	-	0	-
0	1	-	0	0	0	0	0	1	0	-	0	-	1	-
-	-	0	0	1	0	1	1	0	1	-	-	0	0	-
-	-	1	0	1	0	0	1	1	0	-	-	0	1	-
0	-	-	0	0	1	0	0	0	0	-	0	-	-	1
1	-	-	0	0	1	0	1	1	0	-	1	-	-	0
-	0	-	1	1	0	0	1	1	-	1	-	0	1	-
-	1	-	1	1	0	1	0	0	-	0	-	1	0	-
-	-	-	0	1	1	0	0	0	0	-	-	1	-	1
-	-	-	1	0	0	0	0	0	-	1	0	-	-	0

หา NS Function ได้ดังรูปที่ 31

JA

	BC	00	01	11	10
A	0	0	0	0	$\overline{Q3}$
	1	-	-	-	-

KA

	BC	00	01	11	10
A	0	-	-	-	-
	1	-	-	-	$\overline{Q2}$

JB

	BC	00	01	11	10
A	0	$Q1$	$Q1$	-	-
	1	0	-	-	-

KB

	BC	00	01	11	10
A	0	-	-	1	0
	1	-	-	-	$Q2$

JC

	BC	00	01	11	10
A	0	$\overline{Q1}Q2$	-	-	$Q2$
	1	-	-	-	$\overline{Q2}$

KC

	BC	00	01	11	10
A	0	-	$\overline{Q1}$	1	-
	1	0	-	-	-

$$JA = BC\overline{Q3}$$

$$JB = \overline{A}Q1$$

$$JC = \overline{A}Q1Q2 + \overline{A}BQ2 + A\overline{Q2}$$

$$KA = \overline{B} + \overline{Q2}$$

$$KB = C + A\overline{Q2}$$

$$KC = B + \overline{A}Q1$$

รูปที่ 31 NS Map และ NS Function

2. State Output Table

Table นี้จะกล่าวถึงเฉพาะ O/P ที่ขึ้นอยู่กับ State เพียงอย่างเดียว (ไม่ขึ้นกับ I/P) เป็น O/P ซึ่งปรากฏอยู่ใน State Box เท่านั้น

จำนวนแถวของตารางจะเท่ากับจำนวน State ตารางมี 2 ส่วน ส่วนแรกเป็น PS ส่วนที่สอง เป็น State O/P

State O/P Table ของ ASM Chart รูปที่ 29 แสดงดังตารางที่ 4

ตารางที่ 4 State O/P Table

(a) Symbolic Form

PS	O/P	
	H.ADD	H.LAMP
1		
2		
3	A	
4		A
5		A
6	A	

(b) Assigned Form

PS			O/P	
A	B	C	H.ADD	H.LAMP
0	0	0	0	0
0	1	0	0	0
0	0	1	1	0
1	1	0	0	1
0	1	1	0	1
1	0	0	1	0

จาก O/P State Table เราสามารถหา State O/P Function ได้ดังรูปที่ 32

H.ADD

A	BC			
	00	01	11	10
0	0	1	0	0
1	1	-	-	0

$$H.ADD = \overline{B}C + A\overline{B}$$

H.LAMP

A	BC			
	00	01	11	10
0	0	0	1	0
1	0	-	-	1

$$H.LAMP = BC + AB$$

3. Conditional Output Table

Table นี้จะแสดง O/P ที่ Active ตามเงื่อนไขของ I/P ซึ่งก็คือ O/P ที่มีปรากฏอยู่ใน Conditional O/P Box (บางครั้งอาจปรากฏใน State Box ด้วย)

จำนวนแถวของตารางจะเท่ากับจำนวน Link Path และมี 3 Column คือ

- Section ที่ 1 เป็น I/P ของวงจร
- Section ที่ 2 เป็น PS
- Section ที่ 3 เป็น Conditional O/P

Conditional O/P Table ของรูปที่ 29 แสดงดังตารางที่ 5

ตารางที่ 5 Conditional O/P Table

(a) Symbolic Form

I/P			PS	Conditional O/P		
Y.Q ₁	Y.Q ₂	Y.Q ₃		IL.INC	IH.SET	IH.CLR
0	0	-	1			A
1	-	-	1	A		A
0	1	-	1			A
-	-	0	2			
-	-	1	2		A	
0	-	-	3			A
1	-	-	3		A	A
-	0	-	4			
-	1	-	4			A
-	-	-	5		A	
-	-	-	6	A		

(b) Assigned Form

I/P			PS			O/P		
Q ₁	Q ₂	Q ₃	A	B	C	INC	SET	CLR
0	0	-	0	0	0	1	0	1
1	-	-	0	0	0	0	0	1
0	1	-	0	0	0	1	0	1
-	-	0	0	1	0	1	0	0
-	-	1	0	1	0	1	1	0
0	-	-	0	0	1	1	0	1
1	-	-	0	0	1	1	1	1
-	0	-	1	1	0	1	0	0
-	1	-	1	1	0	1	0	1
-	-	-	0	1	1	1	1	0
-	-	-	1	0	0	0	0	0

เราสามารถนำมาหา Condition O/P Function ได้ดังรูปที่ 33

IL.INC		BC				
		A	00	01	11	
0	1	$\overline{Q1}$	1	1	1	IL.INC = $\overline{AQ2} + B + C$
	1	0	-	-	1	

IH.SET		BC				
		A	00	01	11	
0	1	0	$Q1$	1	$Q3$	IH.SET = $ABQ1 + ABC + ABQ3$
	1	0	-	-	0	

IH.CLR		BC				
		A	00	01	11	
0	1	1	1	0	0	IH.CLR = $\overline{AB} + ABQ2$
	1	0	-	-	$Q2$	

รูปที่ 33 Conditional O/P Map และ Function

จะเห็นว่า ASM Table ทั้ง 3 ตาราง สามารถจะนำมาหาฟังก์ชันต่าง ๆ ที่จะนำมาสร้างวงจรได้ คือ

- NS Table นำมาหา NS Function
- State O/P Table และ Conditional O/P Table นำมาหา State O/P Function

ผู้ออกแบบอาจเขียน ASM table โดยรวมดังตารางที่ 6 ก็ได้

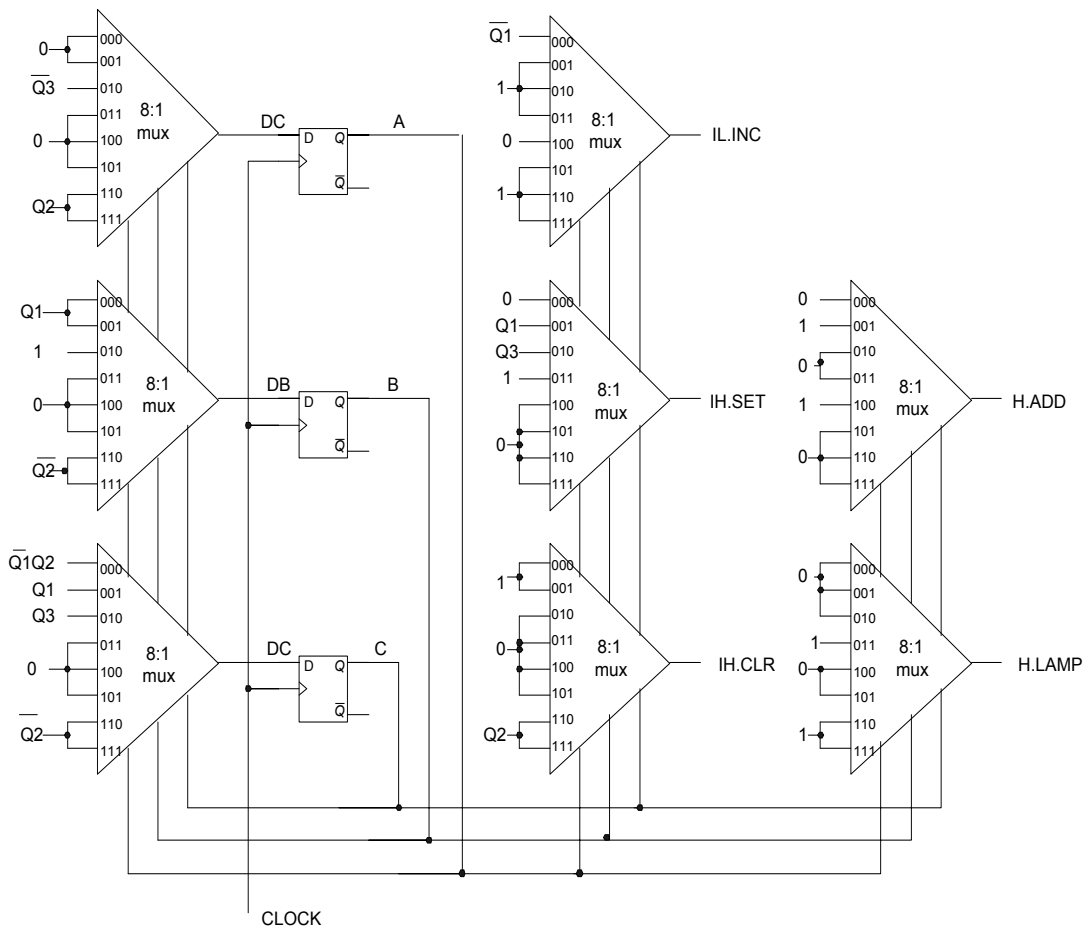
ตารางที่ 6 ASM Table

Link Path	I/P			PS			NS			O/P				
	Q ₁	Q ₂	Q ₃	A	B	C	NA	NB	NC	State		Condition		
										H.ADD	H.LAMP	IL.INC	IH.SET	IH.CLR
1	0	0	-	0	0	0	0	0	0	0	0	1	0	1
2	1	-	-	0	0	0	0	1	0	0	0	0	0	1
3	0	1	-	0	0	0	0	0	1	0	0	1	0	1
4	-	-	0	0	1	0	1	1	0	0	0	1	0	0
5	-	-	1	0	1	0	0	1	1	0	0	1	1	0
6	0	-	-	0	0	1	0	0	0	1	0	1	0	1
7	1	-	-	0	0	1	0	1	1	1	0	1	1	1
8	-	0	-	1	1	0	0	1	1	0	1	1	0	0
9	-	1	-	1	1	0	1	0	0	0	1	1	0	1
10	-	-	-	0	1	1	0	0	0	0	1	1	1	0
11	-	-	-	1	0	0	0	0	0	1	0	0	0	0

เราอาจเขียน ASM Table ดังตารางที่ 6 แล้วค่อยหาฟังก์ชันแต่ละส่วนก็ได้ แต่อาจทำให้สับสนและผิดพลาดได้

สร้างวงจร

หลังจากที่เราได้หา NS Function และ O/P Function แล้ว เราสามารถนำมาเขียนวงจรได้
รูปที่ 34 เป็นวงจรจาก ASM Chart รูปที่ 29 ที่สร้างโดยใช้ D flip-flop และรูปที่ 35 เป็นวงจรที่
สร้างโดยใช้ JK flip-flop



รูปที่ 34 วงจรจาก ASM Chart รูปที่ 29 ที่สร้างโดยใช้ D – Flip Flop