

บทที่ 5

การทดสอบและผลการทดสอบ

บทนำ

ขั้นตอนการทดสอบของการทำวิจัย เนื่องจากการวิจัยเป็นการค้นหาวิธีการใหม่เพื่อนำมาใช้ในการออกแบบโปรแกรมควบคุมพีแอลซี ดังนั้นการทดสอบจึงเป็นการนำเสนอไขการควบคุมที่กำหนดขึ้นมาแล้วเขียนเป็นเงื่อนไขป้อนเข้าสู่ระบบซอฟต์แวร์ทำการสังเคราะห์วงจรตรรก และแปลงออกมาเป็นรหัสนิโมติกส์ และรหัสคำสั่งของพีแอลซี CMU-1 รหัสนิโมติกส์จะสามารถนำไปป้อนให้กับพีแอลซีทั่วไปทำงานได้แต่มีข้อแม้ว่าจะต้องเปลี่ยนหมายเลขของอุปกรณ์อินพุต เอาต์พุต ตัวตั้งเวลา และตัวนับให้มีรูปแบบตามข้อกำหนดของพีแอลซีแต่ละรุ่นที่นำไปใช้ ทั้งนี้เนื่องจาก รหัสนิโมติกส์เป็นโปรแกรมที่บอกให้พีแอลซีเชื่อมต่ออุปกรณ์ภายในแต่ละชนิดที่ได้อ้างถึงเท่านั้น

5.1 การทดสอบการสังเคราะห์วงจรและผลการทดสอบ

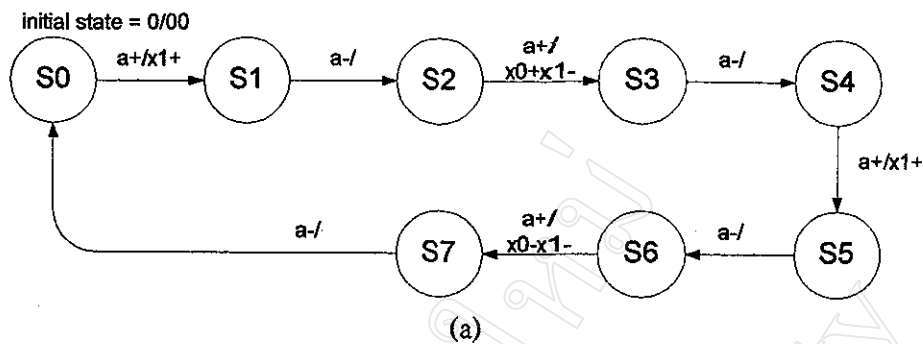
การทำงาน เริ่มต้นจากการกำหนดพฤติกรรมของระบบควบคุม ซึ่งเขียนในรูปของ ภาษาอธิบายการเปลี่ยนแปลงสถานะ (State Transition Description Language) เมื่อนำเงื่อนไขในรูปแบบนี้ไปป้อนเข้าสู่ซอฟต์แวร์ที่สร้างขึ้นเพื่อทดสอบ จะได้เอาต์พุตออกมาเป็นโปรแกรมสำหรับพีแอลซีในรูปแบบของรหัสนิโมติกส์ และรหัสคำสั่งของพีแอลซีรุ่น CMU-1 เราสามารถตรวจสอบความถูกต้องของซอฟต์แวร์ก่อนกำเนิดได้โดยการนำโปรแกรมสำหรับพีแอลซีที่เป็นรหัสคำสั่งนั้นไปรัน (Run) พีแอลซี และตรวจสอบผลการทำงานว่าถูกต้องตามพฤติกรรมของระบบควบคุมที่ต้องการหรือไม่

จากผลการทดลองรัน โปรแกรมหลายๆ แบบพบว่าพีแอลซีทำงานได้ถูกต้อง ตามที่กำหนดในขั้นตอนการกำหนดพฤติกรรมการควบคุม

5.2 ตัวอย่างการทดสอบ

ต่อไปนี้เป็นตัวอย่างการทดสอบการทำงานของตัวก่อกำเนิดโปรแกรมสำหรับระบบควบคุมที่ใช้พีแอลซีเป็นตัวควบคุมหลัก แต่ละตัวอย่างของการทดสอบเป็นการสมมติเงื่อนไขขึ้นมา และสร้างการอธิบายพฤติกรรมของระบบควบคุมตามข้อกำหนดและข้อจำกัดของการกำหนด

ตัวอย่างที่ 1 พฤติกรรมการควบคุมดังแสดงในรูปที่ 5.2 พฤติกรรมของการทำงานคือ เมื่อมีอินพุตเปลี่ยนแปลงจาก 0 เป็น 1 และเป็น 0 เอาต์พุตจะทำการนับครั้งละ 1 โดยแสดงเป็นไบนารีขนาด 2 บิตเมื่อครบ 3 ก็จะกลับมาเป็น 0 ใหม่



Input (a)
 Output (x0,x1)
 Timer ()
 Counter ()
 State (s0,s1,s2,s3,s4,s5,s6,s7)
 Initial state s0(0/00)
 S0,a+/x1+,S1
 S1,a-/ ,S2
 S2,a+/x0+x1-,S3
 S3,a-/ ,S4
 S4,a+/x1+,S5
 S5,a-/ ,S6
 S6,a+/x0-x1-,S7
 S7,a-/ ,S0

(b)

รูปที่ 5.1 แสดงตัวอย่างกราฟอธิบายการเปลี่ยนแปลงสถานะ(a) ที่ใช้ทดสอบการทำงาน และภาษาอธิบายการเปลี่ยนแปลงสถานะ (b)

ผลจากการออกแบบโปรแกรมพีแอลซีของพฤติกรรมของระบบควบคุมที่กำหนดในรูปที่ 5.1 โดยการใช้ตัวก่อกำเนิดโปรแกรมก่อกำเนิดโปรแกรมมีผลลัพธ์ดังนี้

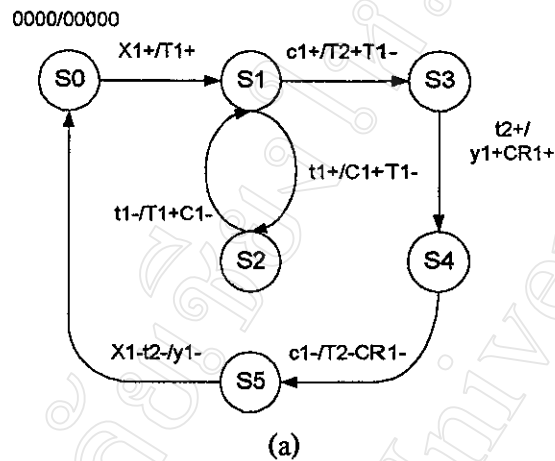
PLC-PROG-GEN-V1.0-CMU-1		
INPUT a = X100; OUTPUT X0 = Y100; OUTPUT X1 = Y101; AUX RELAY = AR100; AUX RELAY = AR101; TIMER ; COUNTER ; MNEMONICS CODE.		
ADDRESS	INSTRUCTION	DATA
0	LD	AR101
1	LD	Y100
2	LD	AR100
3	ORN	AR100
4	ANDL	
5	LDN	AR100
6	AND	Y100
7	AND	AR100
8	ORL	
9	ANDL	
10	LDN	X100
11	AND	AR100
12	AND	Y100
13	AND	AR100
14	ORL	
15	OUT	Y100
16	LD	X100
17	LD	AR100
18	AND	AR101

19	AND	Y100
20	LDN	AR100
21	ANDN	AR101
22	ANDN	Y100
23	ORL	
24	ANDL	
25	LD	AR100
26	LD	X100
27	ANDN	AR100
28	AND	Y100
29	LDN	AR100
30	ANDN	AR101
31	ANDN	Y100
32	ORL	
33	ANDL	
34	ORL	
35	OUT	Y101
36	LD	AR100
37	LD	Y100
38	LD	AR100
39	OR	X100
40	ANDL	
41	LD	X100
42	ANDN	AR101
43	ORL	
44	ANDL	
45	LDN	X100
46	AND	AR101
47	LD	Y100
48	ANDN	AR100
49	ORL	
50	OUT	AR100
51	LD	AR101
52	LD	Y100
53	LDN	AR100

54	LD	X100
55	AND	AR100
56	ORL	
57	ANDL	
58	LDN	AR100
59	ANDN	Y100
60	AND	AR100
61	ORL	
62	LDN	X100
63	ANDN	AR100
64	ANDN	Y100
65	AND	AR100
66	ORL	
67	OUT	AR101
68	END	

รูปที่ 5.2 แสดงรหัสนิโมติกส์จากผลการทำงานของซอฟต์แวร์

ตัวอย่างที่ 2 เป็นตัวอย่างการกำหนดเงื่อนไขที่มีตัวตั้งเวลาและตัวนับภายในพีแอลซี เนื่องจากตัวตั้งเวลาและตัวนับเมื่อค่าของเวลาและค่าของการนับตรงตามที่กำหนดก็จะสร้างอินพุตภายในขึ้นมาเป็นเงื่อนไขให้กับระบบควบคุมได้



Input (x1)
 Output (y1)
 Timer (t1,10,t2,20)
 Counter (c1,3)
 State (s0,s1,s2,s3,s4,s5)
 Initial state s0(0000/00000)
 S0,x1+/T1+,S1
 S1,t+/C1+T1-,S2
 S1,c1+/T2+T1-,S3
 S2,t1-/T1+C1--,S1
 S3,t2+/Y1+CR1+,S4
 S4,c1-/T2-CR1-,S5
 S5,c1-t2-/Y1-,S0

(b)

รูปที่ 5.3 แสดงตัวอย่างกราฟอธิบายการเปลี่ยนแปลงสถานะ(a)
 และภาษาอธิบายการเปลี่ยนแปลงสถานะ (b)

ผลจากการออกแบบ โปรแกรมพีแอลซีของพฤติกรรมของระบบควบคุมที่กำหนดในรูปที่ 5.3
 โดยการใช้ตัวก่อกำเนิด โปรแกรมก่อกำเนิด โปรแกรมมีผลลัพธ์ดังนี้

PLC-PROG-GEN-V1.0-CMU-1

 INPUT x1 = X100;
 OUTPUT Y1 = Y100;
 AUX RELAY = AR100
 AUX RELAY = AR101
 AUX RELAY = AR102
 AUX RELAY = AR103
 TIMER T1 = T100,10, AR100;
 TIMER T2 = T101,20, AR101;
 COUNTER C1 = C100,3, AR102, CR1 = AR103;

MNEMONICS CODE.

ADDRESS	INSTRUCTION	DATA
0	LDN	T100
1	LDN	AR100
2	LDN	AR102
3	LD	X100
4	LD	T101
5	LDN	C100
6	AND	Y100
7	LD	C100
8	AND	AR101
9	ORL	
10	ORL	
11	ANDL	
12	ANDL	
13	LD	T101
14	ANDN	C100
15	AND	Y100
16	ANDN	AR101

17	AND	AR103
18	ORL	
19	ANDL	
20	ANDL	
21	ANDL	
22	OUT	Y100
23	LD	X100
24	ANDN	T100
25	ANDN	T101
26	ANDN	C100
27	ANDN	Y100
28	ANDN	AR101
29	ANDN	AR103
30	OUT	AR100
31	OUT	T100
32		10
33	LD	X100
34	LDN	T100
35	LDN	C100
36	LDN	AR102
37	LDN	T101
38	ANDN	AR100
39	ANDN	AR103
40	LDN	T101
41	ANDN	Y100
42	ANDN	AR103
43	ORL	
44	ANDL	
45	OUT	AR101
46	OUT	T101
47		20
48	LD	X100
49	AND	T100
50	ANDN	T101
51	ANDN	C100

52	ANDN	Y100
53	ANDN	AR101
54	ANDN	AR103
55	OUT	AR102
56	OUT	C100
57		3
58	LD	X100
59	ANDN	T100
60	AND	T101
61	AND	C100
62	ANDN	AR100
63	AND	AR101
64	ANDN	AR102
65	OUT	AR103
66	OUT	C100
67	END	

รูปที่ 5.4 แสดงรหัสรีเลย์จากผลการทำงานของซอฟต์แวร์

จากผลการสังเคราะห์โปรแกรม ในเงื่อนไขที่ประกอบด้วยตัวตั้งเวลาและตัวนับ รีเลย์ช่วยจะเป็นเอาต์พุตที่คู่กับตัวตั้งเวลาและตัวนับทั้งนี้เนื่องจากตัวตั้งเวลาและตัวนับไม่สามารถเป็นสัญญาณป้อนกลับมาเป็นอินพุตให้กับวงจร ได้ดังนั้นจึงใช้วิธีการเพิ่มรีเลย์ช่วยจับคู่กับตัวตั้งเวลาและตัวนับเพื่อนำเอาเอาต์พุตของรีเลย์ช่วยมาเป็นสัญญาณป้อนกลับแทน (มีผลในกระบวนการสร้างตารางสถานะของซอฟต์แวร์ที่กำหนด)