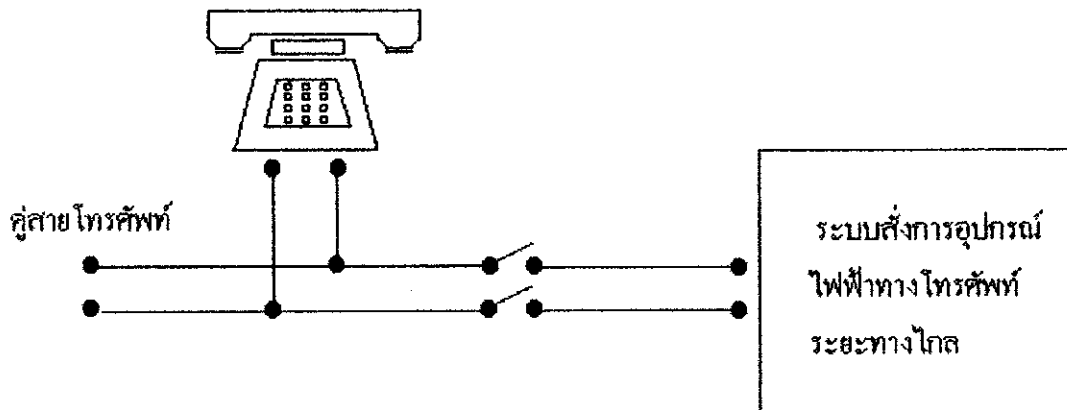


บทที่ 4

วงจรรวมและโปรแกรม

ในบทนี้จะแสดงส่วนประกอบของวงจรทั้งหมดของระบบการส่งอุปกรณ์ไฟฟ้าทางโทรศัพท์ระยะทางไกล

4.1 ส่วนของสายสัญญาณโทรศัพท์[3]

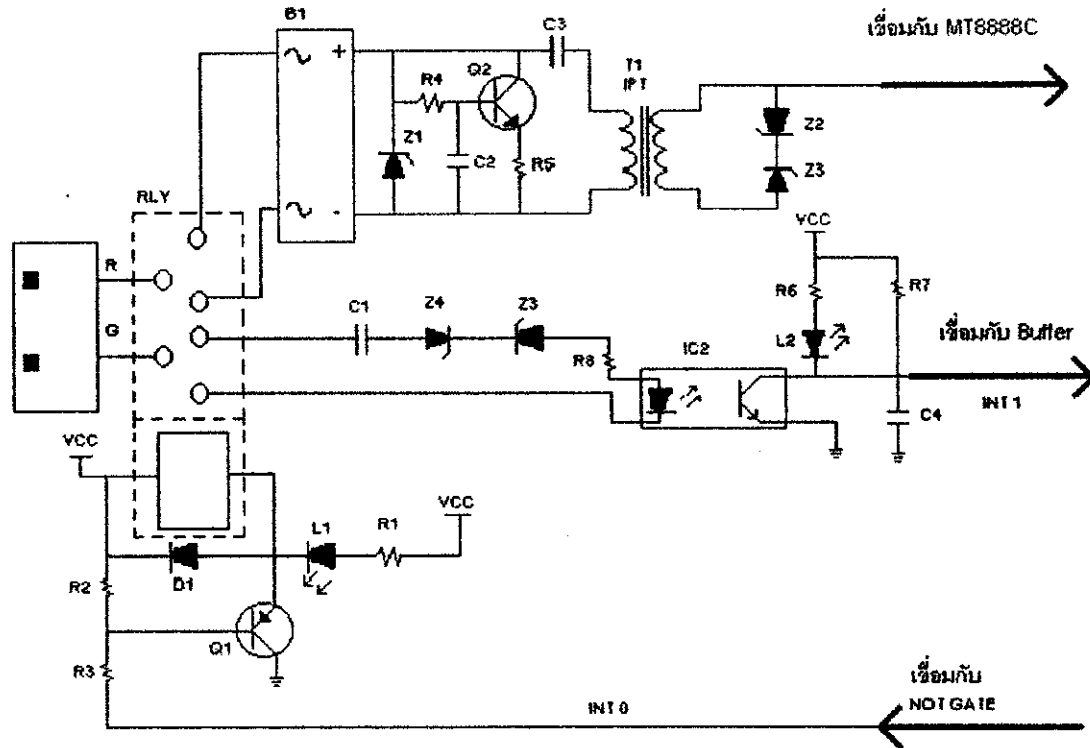


รูปที่ 4.1 แสดงสายสัญญาณโทรศัพท์ภายนอก

จากรูปที่ 4.1 แสดงการเชื่อมต่อกับสายสัญญาณโทรศัพท์ เริ่มจากนำสายสัญญาณของชุมสายโทรศัพท์ที่นำมาต่อกับโทรศัพท์บ้านตามปกติทั่วไป นำมาต่อขนานกับระบบการส่งการอุปกรณ์ไฟฟ้าทางโทรศัพท์ระยะทางไกลและจะมีสวิชต์กันไว้ระหว่างโทรศัพท์บ้านปกติกับระบบการส่งการอุปกรณ์ไฟฟ้าทางโทรศัพท์ระยะทางไกล สวิตซ์นี้ใช้สำหรับเวลาปกติที่เราไม่ต้องการใช้ระบบการส่งการอุปกรณ์ไฟฟ้าทางโทรศัพท์ระยะทางไกล

4.2 ส่วนของการแปลงสัญญาณโทรศัพท์[3]

ในส่วนของการแปลงสัญญาณโทรศัพท์นี้ เป็นการแปลงสัญญาณ DTMF จากสายสัญญาณของโทรศัพท์ เพื่อนำมาใช้กับการควบคุมในวงจร รวมถึงการใช้สัญญาณ DTMF ได้คอบจากระบบการส่งการอุปกรณ์ไฟฟ้าทางโทรศัพท์ระยะทางไกล ไปยังผู้ส่งงานระบบนี้ เป็นการรายงานสถานะของระบบไปยังผู้ส่งงาน จากรูปที่ 4.2 ในส่วนนี้จะมีการนับเสียงเรียกเข้าของสัญญาณโทรศัพท์แสดงออกทางไดโอดเปล่งแสง(L1) มีการแสดงการรับสายทางไดโอดเปล่งแสง(L2)



รูปที่ 4.2 แสดงวงจรการแปลงสัญญาณ DTMF

RLY = 5V

R1, R3, R6 = 470Ω

R8 = 3K

C1 = 1/250

C3 = 4.7/50

Z1 = 15

Z4, Z5 = 15V

B1 = 1A

R2, R4, R7 = 10KΩ

D1 = 4148

C2 = 22/50

C4 = 10/50

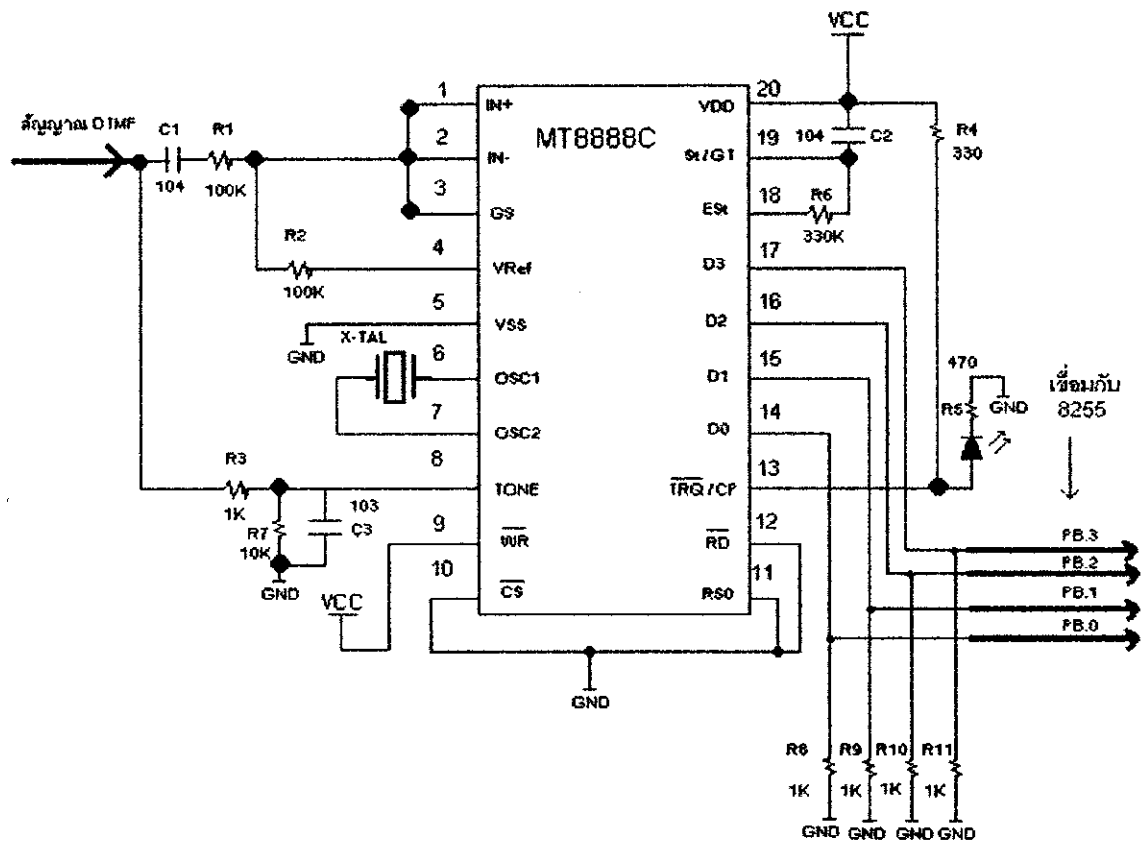
Z2, Z3 = 3.3

จากรูปที่ 4.2 แสดงวงจรในการแปลงสัญญาณ DTMF แต่วงจรนี้ต้องไปเชื่อมต่อกับตัว MT8888C ในรูปที่ 4.3 ในรูปที่ 4.2 นี้จะแบ่งออกเป็น 3 ส่วน คือ ส่วนบนสุด จะเป็นการแปลงสัญญาณ DTMF โดยสัญญาณที่ได้จะถูกจัดการโดยตัว MT8888C (รูปที่ 4.3) ในส่วนที่สอง เป็นส่วนของการนับเสียงสัญญาณของกริ่งโทรศัพท์ โดยสัญญาณที่ได้ออกมาจะถูกเชื่อมต่อเข้า Not Gate เพื่อจะนำสัญญาณที่ได้มาทำการนับเสียงกริ่ง แล้วเลือกว่าจะรับสายที่สัญญาณกริ่งที่เท่าไร ในส่วนที่สาม เป็นส่วนของการรับสาย โดยเมื่อสัญญาณของการนับเสียงกริ่ง ดังครบตามที่เรา

เลือกไว้แล้ว ไมโครคอนโทรลเลอร์จะสั่งให้รับสายจะแสดงการรับสายที่ได้ออกแปลงแสง(L1 ในรูปที่ 4.2)

4.3 ส่วนของการแปลงสัญญาณโทรศัพท์ด้วย MT8888C

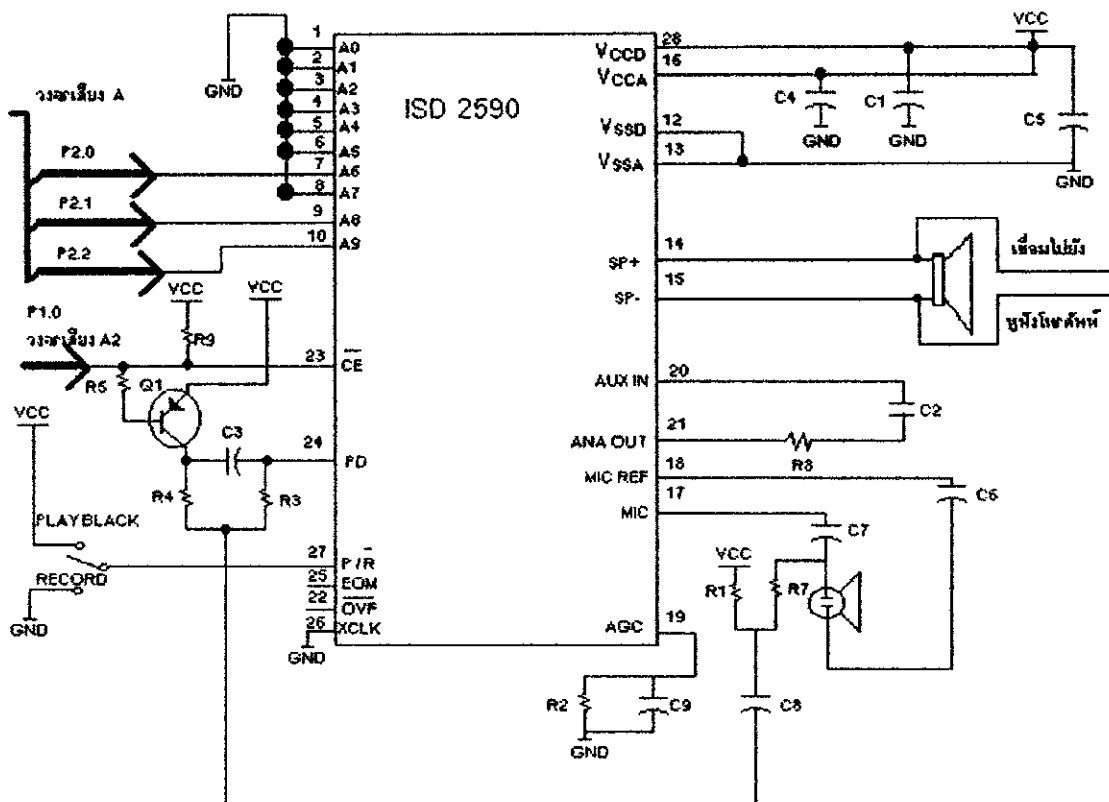
ในส่วนนี้จะรับสัญญาณจากการแปลงสัญญาณในช่วงแรก(รูปที่4.2) จะนำมาประมวลผลเลือกช่วงความถี่ของเสียง รวมถึงการกำจัดสัญญาณรบกวนต่าง แล้วถอดรหัสออกมาเป็นเลขฐาน 2 เพื่อแสดงว่าการกดโทรศัพท์นั้นเป็นการกดปุ่มไหน เช่น 1, 2, *, #, 7 เป็นต้น รหัสเลขฐาน 2 ที่ได้จะแสดงออกทางขา D0 – D3 แล้วจะนำไปเชื่อมต่อกับ 8255 เพื่อทำการรับค่าที่ได้มาประมวลผลต่อไป



รูปที่ 4.3 แสดงการถอดรหัสด้วย MT8888C

4.6 วงจรเสียง

วงจรเสียงนี้ประกอบขึ้นโดยใช้ตัว ISD2590 เป็น ไอซีที่สามารถบันทึกเสียงได้ ในวงจรนี้เราใช้ไมโครคอนโทรลเลอร์ มาควบคุมในการเลือกช่องสัญญาณเสียงและในการเล่นเสียงเข้าสู่ระบบ



รูปที่ 4.6 วงจรเสียงรูปที่ 1

$$R1 = 2.2K\Omega$$

$$R2 = 470K\Omega$$

$$R3 = 220K\Omega$$

$$R4, R9 = 47K\Omega$$

$$R5 = 1K\Omega$$

$$R7 = 10K\Omega$$

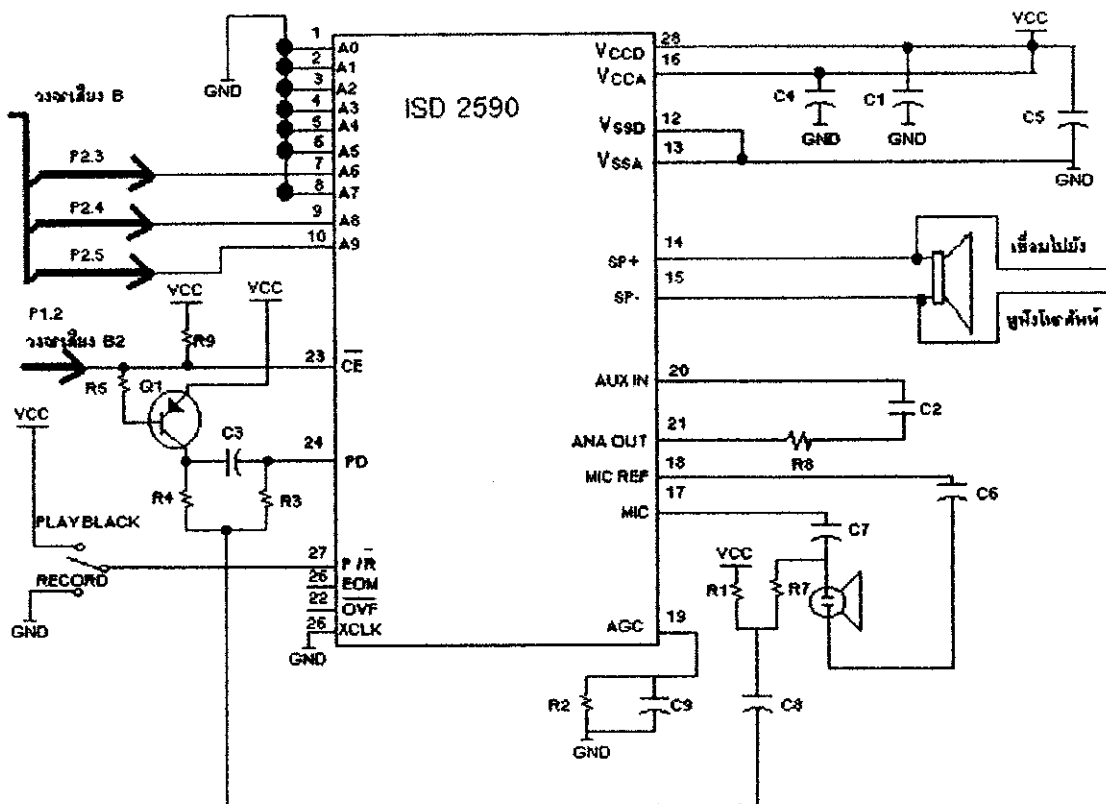
$$R8 = 5.1K\Omega$$

$$Q1 = 2N4403$$

$$C1, C2, C3, C4, C5, C6, C7 = 0.1\mu F$$

$$C8 = 2.2\mu F$$

$$C9 = 4.7\mu F$$



รูปที่ 4.7 วงจรเสียงรูปที่ 2

$R1 = 2.2K\Omega$

$R2 = 470K\Omega$

$R3 = 220K\Omega$

$R4, R9 = 47K\Omega$

$R5 = 1K\Omega$

$R7 = 10K\Omega$

$R8 = 5.1K\Omega$

$Q1 = 2N4403$

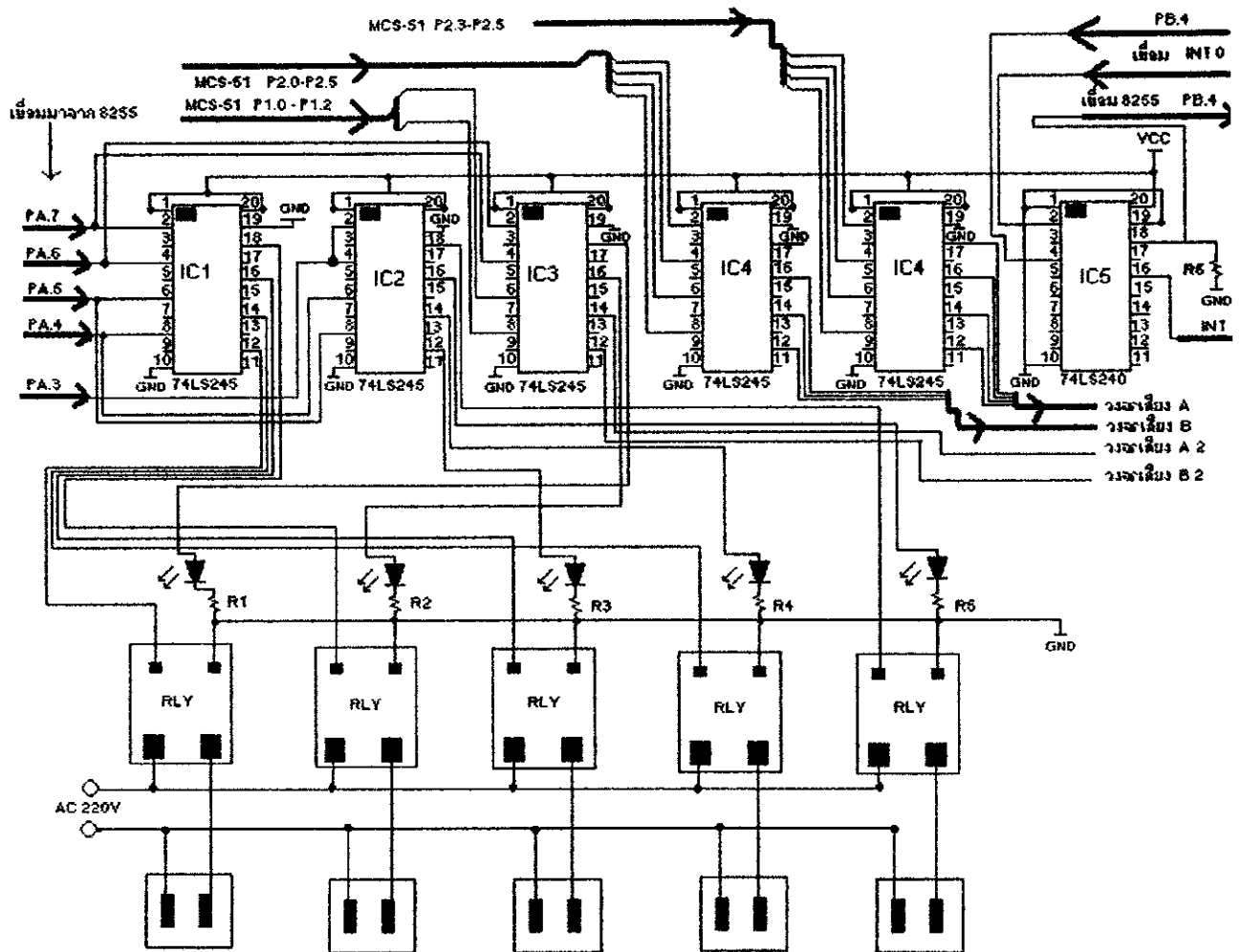
$C1, C2, C3, C4, C5, C6, C7 = 0.1\mu F$

$C8 = 2.2\mu F$

$C9 = 4.7\mu F$

4.7 วงจรบัฟเฟอร์และการต่อเอาต์พุต[3]

การที่เรานำสายสัญญาณมาต่อเข้ากับบัฟเฟอร์ เพื่อให้แรงดันไฟฟ้ามีค่าคงที่เท่ากับ 5 โวลต์ เพื่อที่จะสามารถเปิดรีเลย์ให้สามารถทำงานได้เต็มประสิทธิภาพและใช้แสดงผลผ่านไดโอดเปล่งแสง รวมถึงยังใช้ในการเพิ่มแรงดันในการควบคุมวงจรเสียงอีกด้วย



เค้าโครงวงจรสำหรับการทดสอบ

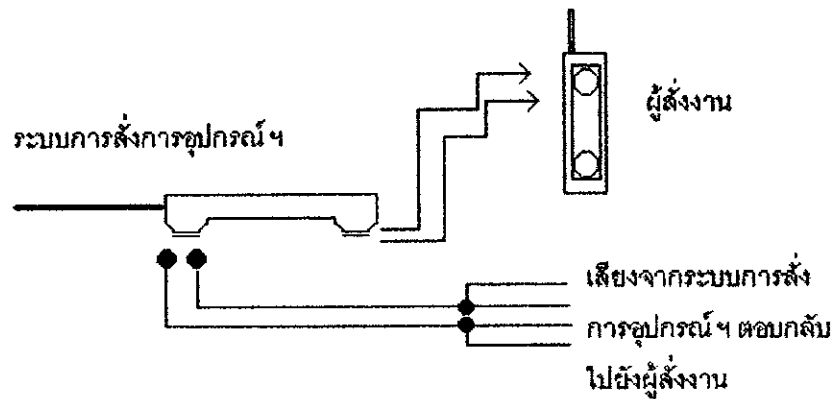
รูปที่ 4.8 วงจรบัฟเฟอร์และการต่อเอาต์พุต

R1, R2, R3, R4, R5 = 100Ω

R6 = 1KΩ

4.8 การตอบเสียงกลับไปยังผู้ใช้งาน

การตอบกลับไปยังผู้ใช้งานเป็นการรายงานสถานะของอุปกรณ์ไฟฟ้า เพื่อให้ผู้ใช้งานสามารถรับรู้สถานะปัจจุบันของระบบได้ ในวงจรจะมีการต่อสายสัญญาณเสียงออกจากวงจรเสียง (รูปที่ 4.6 และ รูปที่ 4.7) ผ่านมายังไมโครโฟนของวงจรโทรศัพท์ ทำให้ผู้ฟังสามารถได้ยินเสียงการรายงานสถานะได้



รูปที่ 4.9 แสดงการรายงานสถานะไปยังผู้ใช้งาน

4.9 โปรแกรมภาษาแอสเซมบลีที่ใช้ควบคุมการทำงานของวงจร

1.	ORG	0000H		; เริ่มต้นโปรแกรม
2.	CTR_L	EQU	8003H	; บรรทัดที่ 2-5 กือหมาย
3.	P_A	EQU	8000H	; เลขพอร์ตของไอซี 8255
4.	P_B	EQU	8001H	
5.	P_C	EQU	8002H	
6.	RE_1	EQU	20H	; บรรทัดที่ 6-11
7.	RE_2	EQU	21H	; กำหนดตำแหน่งของ
8.	RE_3	EQU	22H	; รีจิสเตอร์บัฟเฟอร์
9.	RE_4	EQU	23H	
10.	RE_5	EQU	24H	
11.	SHO_W	EQU	2FH	
12.		MOV	R3,#08H	; บรรทัดที่ 12-29
13.		MOV	R2,#00H	; เริ่มให้ไมโครคอน
14.	INI_TIAL1:	MOV	DPTR,#DATA_INI	; โทลเลอร์ทำงาน
15.		MOV	A,R2	; ที่พอร์ต P1 และ P2
16.		MOVC	A,@A+DPTR	; และ 8255 จะทำงาน
17.		MOV	P1,A	; พร้อมกันเลยโดยไม
18.		ACALL	DELAY	; ต้องไปจับปีเปอร์ชา
19.		INC	R2	; EA
20.		DJNZ	R3,INI_TIAL1	
21.		MOV	R3,#08H	
22.		MOV	R2,#00H	
23.	INI_TIAL2:	MOV	DPTR,#DATA_INI	
24.		MOV	A,R2	
25.		MOVC	A,@A+DPTR	
26.		MOV	P2,A	
27.		ACALL	DELAY	
28.		INC	R2	

```

29.          DJNZ R3,INI_TIAL2
30.  INI_TIAL3: MOV DPTR,#CTR_L      ; บรรทัดที่ 30-32
31.          MOV A,#82H             ; กำหนดค่าพอร์ตให้ออซี
32.          MOVX @DPTR,A           ; 8255 ทำงาน โดยให้
                                     ; พอร์ต A เป็นการแสดง
                                     ; ผล ส่วนพอร์ต B เป็น
                                     ; ส่วนรับข้อมูลจากไอซี
                                     ; MT8888CE และวงจร
                                     ; นับสัญญาณเสียงเรียก
                                     ; เข้า และพอร์ต C เป็น
                                     ; ส่วนของการรับสาย

33.          MOV RE_1,#00H         ; บรรทัดที่ 33-38 รีเซ็ต
34.          MOV RE_2,#00H         ; ค่ารีเซ็ตออร์บิทัลเฟอ์
35.          MOV RE_3,#00H         ; ให้เป็น 0 ทั้งหมด
36.          MOV RE_4,#00H
37.          MOV RE_5,#00H
38.          MOV SHO_W,#00H

39.          MOV A,#00000011B      ; บรรทัดที่ 39-40 รีเซ็ต
40.          MOV P1,A              ; วงจรบันทึกเสียงไม่ให้
                                     ; ทำงาน โดยจ่ายไฟไป
                                     ; ให้ขา CE 5 V.

```

โปรแกรมหลัก

```

41.  MAIN:    CALL START_1          ; เรียกโปรแกรมย่อย
                                     ; START_1 เพื่อตรวจ
                                     ; สอบว่ามีสัญญาณเรียก
                                     ; เข้ามาหรือไม่หรือมีการ
                                     ; รีเซ็ตหรือไม่
42.          CALL REPEA_T          ; เรียกโปรแกรมย่อย

```

```

; REPEA_T เพื่อรีเซ็ต
; ไอซี MT8888CE
43.          CALL PASS_W1          ; เรียกโปรแกรมย่อยเพื่อ
; ตรวจสอบรหัสผ่าน
44.          JMP  ORDE_RR          ; โปรแกรมรับคำสั่งว่าจะ
; สั่งงานอะไรบ้าง
;*****
45.  START_1:  MOV  R2,#45D          ; นับสัญญาณเสียงเรียกเข้า
46.  OPER_RI:  MOV  DPTR,#P_B        ; บรรทัดที่ 45-50 ตรวจสอบ
47.          MOVX A,@DPTR          ; สอบสัญญาณครั้งที่เรียก
48.          JB   ACC.4,INCR_1      ; เข้าและตรวจว่ามีการกด
49.          JB   ACC.5,STO_P       ; รีเซ็ตหรือไม่
50.          JMP  OPER_RI
;*****
51.  INCR_1:   MOV  DPTR,#P_B        ; บรรทัด 51-54 มีกริ่ง
52.          MOVX A,@DPTR          ; เรียกเข้าแล้วตรวจสอบ
53.          JB   ACC.4,INCR_1      ; จำนวนครั้งที่ถูกต้อง
54.          DJNZ R2,OPER_RI        ; หรือไม่
;*****
55.  SUCES_RI: MOV  DPTR,#P_C        ; บรรทัดที่ 55-61
56.          MOV  A,#01H            ; นับเสียงกริ่งครบแล้ว
57.          MOVX @DPTR,A          ; รับสาย และเลือก
58.          ACALL DELAY            ; ตำแหน่งของเสียงที่
59.          MOV  A,#00010000B      ; บันทึกไว้ในไอซีบันทึก
60.          CALL SOUN_D            ; เสียงคำพูดต้อนรับ
61.          RET                    ; เรียกโปรแกรมเล่นเสียง
; ของวงจรที่ 2
;*****
62.  STO_P:    MOV  DPTR,#P_B        ; บรรทัดที่ 62-96
63.          MOVX A,@DPTR          ; มีการกดรีเซ็ตที่ปุ่มรีเซ็ต
64.          JB   ACC.5,STO_P       ; ระบบจะรายงานว่ามี
65.          CALL CHEC_K            ; อุปกรณ์ไหนเปิดหรือปิด

```

```

66. S_BACK:   MOV  DPTR,#P_B           ; อยู่ข้างหลังจากนั้น
67.           MOVX A,@DPTR           ; จะทำการรีเซ็ตรหัสผ่าน
68.           ANL  A,#11100000B      ; โดยถ้าโยกคิปสวิทช์
69.           JB   ACC.5,G_TO1       ; หลัก 1 ขึ้น และยก
70.           CJNE A,#10000000B,S_GO1 ; หลักที่ 2 ขึ้นและโยกสวิทช์หลัก 1
71.           JMP  MAIN              ; ลง แล้วกด สวิทช์กด 1 ครั้ง ระบบ
72. G_TO1:    MOV  DPTR,#P_B           ; จะทำการรีเซ็ตระบบ แต่
73.           MOVX A,@DPTR           ; ถ้ามีการกดรีเซ็ตแล้ว
74.           ANL  A,#11100000B      ; ระบบรีเซ็ตเริ่มทำงาน
75.           JB   ACC.5,G_TO1       ; คือรายงานสถานะแล้ว
76.           JMP  MAIN              ; ไปกดปุ่มรีเซ็ตหรือโยก
77. S_GO1:    CJNE A,#01000000B,S_BACK; สวิทช์หลักที่ 2 ก่อนหลัก
78. S_BACK1:  MOV  DPTR,#P_B           ; ที่ 1 ระบบจะเริ่มรับค่า
79.           MOVX A,@DPTR           ; ใหม่คือจะไม่ทำการรีเซ็ต
80.           ANL  A,#11100000B      ; และกดรีเซ็ตแล้ว โยกคิป
81.           JB   ACC.5,G_TO1       ; สวิทช์หลักที่ 1 แล้วผ่าน
82.           CJNE A,#10000000B,S_BACK1; แล้วถ้าไปกดปุ่มรีเซ็ต
83. S_BACK2:  MOV  DPTR,#P_B           ; อีกครั้งหนึ่งก็จะไม่รีเซ็ต
84.           MOVX A,@DPTR           ; ให้อีกคือในระหว่างการ
85.           ANL  A,#11100000B      ; โยกสวิทช์จนครบถ้า
86.           JB   ACC.5,G_TO1       ; ไปกดปุ่มรีเซ็ตค้นก่อน
87.           CJNE A,#01000000B,FINA_L ; ระบบก็จะไม่รีเซ็ตให้
88.           JMP  S_BACK2           ; แต่ถ้าโยกมาถึงตำแหน่ง
89. FINA_L:   MOV  DPTR,#P_B           ; สุดท้ายแล้วต้องกดเพื่อ
90.           MOVX A,@DPTR           ; ระบบรีเซ็ตก็จะรีเซ็ตได้
91.           JB   ACC.5,FINA_L1
92.           JMP  FINA_L
93. FINA_L1:  MOV  DPTR,#P_B
94.           MOVX A,@DPTR
95.           JB   ACC.5,FINA_L1
96.           JMP  RES_ET

```

```

;*****

```

```

97.  PASS_W1:  MOV  DPTR,#P_B      ; บรรทัดที่ 97-133 จะทำ
98.                MOVX A,@DPTR      ; การตรวจสอบรหัสผ่าน
99.                ANL  A,#00001111B  ; โดยหลักนี้เลข 2 จะเป็น
100. DIS_1:    CJNE A,#00000001B,DIS_2 ; รหัสที่ถูกต้องถ้าคเลข
101.                CALL FAI_L        ; ที่ไม่ใช่เลข 2 ระบบจะ
102.                JMP  MAIN         ; ทำการวางสายให้โดย
103. DIS_2:    CJNE A,#00000011B,DIS_3 ; อัค โนมติและเริ่ม
104.                CALL FAI_L        ; โปรแกรมให้ใหม่แต่ถ้า
105.                JMP  MAIN         ; กดถูกค็อง จะไปตรวจ
106. DIS_3:    CJNE A,#00000100B,DIS_4 ; หลักค่อไปอีก 2 หลัก
107.                CALL FAI_L        ; และถ้ากด # จะวนรับ
108.                JMP  MAIN         ; ค่าใหม่เพราะกำหนด
109. DIS_4:    CJNE A,#00000101B,DIS_5 ; ให้ # เป็นการกรีเซ็ต
110.                CALL FAI_L        ; ระบบจึงสงวนไว้ 1 หลัก
111.                JMP  MAIN
112. DIS_5:    CJNE A,#00000110B,DIS_6
113.                CALL FAI_L
114.                JMP  MAIN
115. DIS_6:    CJNE A,#00000111B,DIS_7
116.                CALL FAI_L
117.                JMP  MAIN
118. DIS_7:    CJNE A,#00001000B,DIS_8
119.                CALL FAI_L
120.                JMP  MAIN
121. DIS_8:    CJNE A,#00001001B,DIS_9
122.                CALL FAI_L
123.                JMP  MAIN
124. DIS_9:    CJNE A,#00001010B,DIS_10
125.                CALL FAI_L
126.                JMP  MAIN
127. DIS_10:   CJNE A,#00001011B,DIS_11

```

```

128.          CALL  FAI_L
129.          JMP   MAIN
130.  DIS_11:   CJNE  A,#00001100B,PAS_1
131.  PAS_1:    CJNE  A,#02H,PAS_BACK1
132.          JMP   PASS_W2
133.  PAS_BACK1: JMP   PASS_W1
;-----
134.  PASS_W2:  MOV   DPTR,#P_B          ; บรรทัดที่ 134-169
135.          MOVX  A,@DPTR          ; เป็นการตรวจสอบรหัส
136.          ANL   A,#00001111B     ; ผ่านหลัก 2 ลักษณะการ
137.          CJNE  A,#00000001B,DIS_22 ; ทำงานจะทำเหมือนหลัก
138.          CALL  FAI_L          ; แรกแต่เปลี่ยนรหัสที่ถูก
139.          JMP   MAIN          ; เป็น 8 และถ้ากด 2 จะ
140.  DIS_22:   CJNE  A,#00000011B,DIS_33 ; เป็นการวางสายและเริ่ม
141.          CALL  FAI_L          ; ดันโปรแกรมใหม่แทน
142.          JMP   MAIN
143.  DIS_33:   CJNE  A,#00000100B,DIS_44
144.          CALL  FAI_L
145.          JMP   MAIN
146.  DIS_44:   CJNE  A,#00000101B,DIS_55
147.          CALL  FAI_L
148.          JMP   MAIN
149.  DIS_55:   CJNE  A,#00000110B,DIS_66
150.          CALL  FAI_L
151.          JMP   MAIN
152.  DIS_66:   CJNE  A,#00000111B,DIS_77
153.          CALL  FAI_L
154.          JMP   MAIN
155.  DIS_77:   CJNE  A,#00000010B,DIS_88
156.          JMP   PASS_W2
157.  DIS_88:   CJNE  A,#00001001B,DIS_99
158.          CALL  FAI_L

```

```

159.          JMP    MAIN
160.  DIS_99:   CJNE  A,#00001010B,DIS_100
161.          CALL  FAI_L
162.          JMP    MAIN
163.  DIS_100:  CJNE  A,#00001011B,DIS_111
164.          CALL  FAI_L
165.          JMP    MAIN
166.  DIS_111:  CJNE  A,#00001100B,PAS_2
167.  PAS_2:    CJNE  A,#00001000B,PAS_BACK2
168.          JMP    PASS_W3
169.  PAS_BACK2: JMP    PASS_W2
;-----
170.  PASS_W3:  MOV   DPTR,#P_B          ;บรรทัดที่ 170-209
171.          MOVX  A,@DPTR        ;เป็นการตรวจสอบรหัส
172.          ANL  A,#00001111B    ;ผ่านหลัก 3 ลักษณะการ
173.          CJNE A,#00000001B,DIS_222 ;ทำงานจะทำเหมือนหลัก
174.          CALL  FAI_L          ;สองแต่เปลี่ยนรหัสที่ถูก
175.          JMP   MAIN           ;เป็น 7 และถ้าค 8 จะ
176.  DIS_222:  CJNE  A,#00000011B,DIS_333 ;เป็นการวางสายและเริ่ม
177.          CALL  FAI_L          ;ต้นโปรแกรมใหม่แทน
178.          JMP   MAIN           ;ถ้าคมาถึงตรงนี้แล้วถูก
179.  DIS_333:  CJNE  A,#00000100B,DIS_444 ;แสดงว่ารหัสผ่านถูกต้อง
180.          CALL  FAI_L          ;หมด แล้วระบบจะรอ
181.          JMP   MAIN           ;งานการต้อนรับและบอก
182.  DIS_444:  CJNE  A,#00000101B,DIS_555 ;การทำงานว่าจะสั่งอะไร
183.          CALL  FAI_L          ;ให้กดอะไรบ้าง โดย
184.          JMP   MAIN           ;ทำการเลือกตำแหน่งของ
185.  DIS_555:  CJNE  A,#00000110B,DIS_666 ;เสียงที่บันทึกการรายงาน
186.          CALL  FAI_L          ;แล้วเรียกโปรแกรมเล่น
187.          JMP   MAIN           ;เสียงอันที่สองเพราะ
188.  DIS_666:  CJNE  A,#00000010B,DIS_777 ;บันทึกไว้ที่วงจรที่สอง
189.          CALL  FAI_L          ;โดยตำแหน่งที่บันทึก

```



```

190.          JMP    MAIN          ; ไว้คือตำแหน่งที่
191.  DIS_777:  CJNE  A,#00001000B,DIS_888; 00011000
192.  DIS_888:  CJNE  A,#00001001B,DIS_999
193.          CALL  FAI_L
194.          JMP    MAIN
195.  DIS_999:  CJNE  A,#00001010B,DIS_1000
196.          CALL  FAI_L
197.          JMP    MAIN
198.  DIS_1000: CJNE  A,#00001011B,DIS_1111
199.          CALL  FAI_L
200.          JMP    MAIN
201.  DIS_1111: CJNE  A,#00001100B,PAS_3
202.  PAS_3:    CJNE  A,#00000111B,PAS_BACK3
203.          JMP    SUCP_W
204.  PAS_BACK3: JMP  PASS_W3
205.  SUCP_W:   ACALL DELAY
206.          MOV   A,#00011000B
207.          CALL  SOUN_D
208.          ACALL DE_L
209.          RET

;*****
210.  ORDE_RR:  CALL  CHEC_K          ; บรรทัดที่ 210-280
211.  ORDE_R:   MOV   DPTR,#P_B      ; จะทำการรับสัญญาณที่
212.          MOVX  A,@DPTR          ; แปลงสัญญาณการกด
213.          ANL  A,#00001111B      ; จากโทรศัพท์จากไอซี
214.  ORDE_R1:  CJNE  A,#00000001B,ORDE_R2; MT8888CE ถักกด 1
215.          MOV   A,#00000010B     ; จะเลือกตำแหน่งที่บันทึก
216.          CALL  SOUN_D1          ; เสียง 00000010 แล้ว
217.          ACALL DE_Y            ; เรียกวงจรเสียงที่ 1 ให้
218.          MOV   RE_1,#10000000B  ; ทำงาน หลังจากนั้น
219.          JMP   ORDE_R           ; เก็บค่ารีจิสเตอร์ตำแหน่ง
220.  ORDE_R2:  CJNE  A,#00000010B,ORDE_R3; 10000000 เพื่อนำไป

```

```

221.          MOV  A,#00000001B      ; บวกกับค่ารีจิสเตอร์
222.          CALL SOUN_D1           ; ทั้งหมดเพื่อได้ผลลัพธ์
223.          ACALL DE_Y             ; คือ ถ้าเปิดตำแหน่งนั้น
224.          MOV  RE_1,#00000000B   ; จะเป็น 1 ถ้าไม่เปิดจะ
225.          JMP  ORDE_R            ; เป็น 0 เช่น ถ้าปิดคือ
226. ORDE_R3:  CJNE A,#00000011B,ORDE_R4; กด 2 ค่ารีจิสเตอร์ก็จะ
227.          MOV  A,#00000011B      ; เป็น 0 กด 3 และ 4 ก็
228.          CALL SOUN_D1           ; จะคล้ายกับกด 1 และ 2
229.          ACALL DE_Y             ; แต่ถ้ากด 5 การทำงานจะ
230.          MOV  RE_2,#01000000B   ; กล้าย ๆ กัน แต่จะเก็บ
231.          JMP  ORDE_R            ; รีจิสเตอร์ค่า RE_3
232. ORDE_R4:  CJNE A,#00000100B,ORDE_R5; 00100000 และ RE_4
233.          MOV  A,#00000100B      ; จะเก็บ 00010000
234.          CALL SOUN_D1           ; รีจิสเตอร์ RE_5เท่ากับ 0
235.          ACALL DE_Y             ; เพราะ RE_3 คือการเปิด
236.          MOV  RE_2,#00000000B   ; แอร์ที่แฟนคอยและคอม
237.          JMP  ORDE_R            ; เพรสเซอร์และ RE_4
238. ORDE_R5:  CJNE A,#00000101B,ODDR_R; คือพัลลวมสปีกที่ 1
239.          MOV  A,#00110000B      ; และ RE_5 ต้องเป็น
240.          CALL SOUN_D            ; 0 เพราะพัลลวมสปีก
241.          ACALL DE_Y             ; ต่าง ๆ จะเปิดพร้อมกัน
242.          MOV  RE_3,#00100000B   ; ไม่ได้ และกด 6 คือ
243.          MOV  RE_4,#00010000B   ; เก็บค่า 0 ของ RE_3,4,5
244.          MOV  RE_5,#00H         ; เพื่อปิดแอร์
245.          JMP  AIR_1             ; ถ้าทำรายการแค่นี้ไม่เพิ่มสปีกที่สอง
246. ODDR_R:   JMP  ORDE_R6         ; ของ สปีกพัลลวมแอร์ให้กด
                                           ; * เพื่อตกลงและจบ
;-----
247. AIR_1:   MOV  DPTR,#P_B        ; ส่วนตรงนี้คือถ้าต้องการ
248.          MOVX A,@DPTR          ; เพิ่มสปีกพัลลวมแอร์ให้
249.          ANL  A,#00001111B     ; กด 2 และระบบจะจบ
250.          CJNE A,#00000010B,AIR_2 ; เองและกลับไปวนรับค่าใหม่

```

```

251.          MOV  A,#00001000B
252.          CALL SOUN_D
253.          ACALL DE_Y
254.          MOV  RE_4,#00H
255.          MOV  RE_5,#00001000B
256.          JMP  OUT_1
257.  AIR_2:    CJNE A,#00001011B,AIR_1
258.          JMP  OUT_1
;-----
259.  ORDE_R6:  CJNE A,#00000110B,ORDE_R7
260.          MOV  RE_3,#00H
261.          MOV  RE_4,#00H
262.          MOV  RE_5,#00H
263.          MOV  A,#00000110B
264.          CALL SOUN_D1
265.          ACALL DE_Y
266.          JMP  ORDE_R
267.  ORDE_R7:  CJNE A,#00000111B,ORDE_R8 ; ถ้ากดตัวอื่น
268.  ORDE_R8:  CJNE A,#00001000B,ORDE_R9 ; นอกจากที่กล่าว
269.          CALL FAI_L           ; มาข้างต้นจะออกจาก
270.          JMP  MAIN           ; ระบบและเริ่มต้น
271.  ORDE_R9:  CJNE A,#00001001B,ORDE_R10 ; โปรแกรมใหม่
272.          CALL FAI_L           ; แต่กด 7 จะวนเพราะ
273.          JMP  MAIN           ; 7 เป็นรหัสผ่านตัวสุดท้าย
274.  ORDE_R10: CJNE A,#00001010B,ORDE_R11 ; มันจะวิ่งมาทำตรงนี้
275.          CALL FAI_L           ; เลขคือพอร์รหัสผ่านแล้ว
276.          JMP  MAIN           ; ถ้าเรากำหนดตัวนี้ให้
277.  ORDE_R11: CJNE A,#00001011B,ORDE_R12 ; เรียก FAI_L มัน
278.          JMP  OUT_1          ; ก็จะวิ่งออกเลขคือกลับ
279.  ORDE_R12: CJNE A,#00001100B,OOOR_R ; ไปเริ่มต้นโปรแกรม
280.  OOOR_R:   JMP  ORDE_R           ; ใหม่เลขจึงกำหนด
; ให้ กด 7 แล้ววนรับค่า

```

```

;*****
; บรรทคที่ 281-288
281.  SOUN_D:  MOV P2,A           ; นำค่า A ที่ได้มาจากการ
282.          MOV A,#00000001B   ; เลือกตามงานต่าง ๆ
283.          ACALL DELAY        ; มาใช้ แล้วนำไปออก
284.          MOV P1,A           ; พอร์ต P2 เพื่อเลือก
285.          ACALL DELAY        ; ตำแหน่ง จากนั้น
286.          MOV A,#00000011B   ; กำหนดค่า CE คือ
287.          MOV P1,A           ; ตำแหน่งที่ 1 ให้เป็น 0
288.          RET                ; คือ 00000001 และให้
                                   ; CE และ กลับมาเป็น 1
                                   ; โดยให้ 00000011 วงจร
                                   ; จึงทำงาน

```

```

;*****
; บรรทคที่ 289-296
289.  SOUN_D1: MOV P2,A           ; นำค่า A ที่ได้มาจากการ
290.          MOV A,#00000010B   ; เลือกตามงานต่าง ๆ
291.          ACALL DELAY        ; มาใช้ แล้วนำไปออก
292.          MOV P1,A           ; พอร์ต P2 เพื่อเลือก
293.          ACALL DELAY        ; ตำแหน่ง จากนั้น
294.          MOV A,#00000011B   ; กำหนดค่า CE คือ
295.          MOV P1,A           ; ตำแหน่งที่ 0 ให้เป็น 0
296.          RET                ; คือ 00000010 และให้
                                   ; CE และ กลับมาเป็น 1
                                   ; โดยให้ 00000011 วงจร
                                   ; จึงทำงาน

```

```

;*****
297.  FAI_L:   MOV A,#00101000B   ; บรรทคที่ 297-303
298.          CALL SOUN_D         ; รายงานว่าให้ติดต่อกลับ
299.          ACALL DE_L         ; มาใหม่
300.          MOV DPTR,#P_C       ; และเลือกใช้งานที่พอร์ต

```

```

301.          MOV  A,#00H          ; C เพื่อวางสาย โดยให้
302.          MOVX @DPTR,A        ; พอร์ต C เป็น 0
303.          RET

;*****
304. REPEA_T:  MOV  DPTR,#P_B      ; บรรทัดที่ 304-308
305.          MOVX A,@DPTR        ; รีเซ็ต MT8888CE ให้
306.          ANL  A,#00001111B    ; การกดครั้งสุดท้าย
307.          CJNE A,#00001100B,REPEA_T ; เป็นการกด #
308.          RET

;*****
309. OUT_1:    MOV  A,RE_1        ; บรรทัดที่ 309-322
310.          ADD  A,RE_2        ; ทำการบวกรีจิสเตอร์
311.          ADD  A,RE_3        ; RE_1,2,3,4,5 เข้าด้วยกัน
312.          ADD  A,RE_4        ; แล้วแสดงผลตามค่าที่
313.          ADD  A,RE_5        ; บวกได้ตามตำแหน่ง
314.          MOV  SHO_W,A
315.          MOV  DPTR,#P_A
316.          MOV  A,SHO_W
317.          MOVX @DPTR,A
318.          ACALL DELAY
319.          MOV  DPTR,#P_C
320.          MOV  A,#00H
321.          MOVX @DPTR,A
322.          JMP  MAIN

;*****
323. CHEC_K:   MOV  A,RE_1        ; บรรทัดที่ 323-356
324.          CJNE A,#10000000B,CH_K1 ; คือ นำค่ารีจิสเตอร์มา
325.          MOV  A,#00000010B    ; เปรียบเทียบ ถ้า
326.          CALL SOUN_D1        ; RE_1 = 10000000 จะ
327.          ACALL DE_Y         ; รายงานสถานะตาม
328.          JMP  GOO_1         ; งานของรีจิสเตอร์ RE_1
329. CH_K1:    MOV  A,#00000001B   ; คือ พัดลมเปิด ถ้า RE_1

```

```

330.          CALL SOUN_D1          ; เป็น 0 ก็แสดงว่าพัลลภ
331.          ACALL DE_Y            ; ปิด ถ้า RE_2 เป็น
332.  GOO_1:   MOV  A,RE_2          ; 01000000 แล้วแสดง
333.          CJNE A,#01000000B,CH_K2 ; ว่าไฟเปิด ถ้าเป็น 0
334.          MOV  A,#00000011B     ; ไฟเปิด ถ้า RE_4
335.          CALL SOUN_D1          ; เป็น 00010000 แสดงว่า
336.          ACALL DE_Y            ; แอร์สปีดที่ 1 เปิด ถ้า
337.          JMP  GOO_2            ; เป็น 0 ให้ตรวจสอบต่อ
338.  CH_K2:   MOV  A,#00000100B     ; ถ้า RE_5 เป็น 00001000
339.          CALL SOUN_D1          ; แสดงว่าแอร์สปีดที่ 2 ติด
340.          ACALL DE_Y            ; ถ้าทั้งสองอัน เป็น 0
341.  GOO_2:   MOV  A,RE_4          ; แสดงว่าแอร์เปิด
342.          CJNE A,#00010000B,CH_K3
343.          MOV  A,#00000101B
344.          CALL SOUN_D1
345.          ACALL DE_Y
346.          JMP  CH_SUC
347.  CH_K3:   MOV  A,RE_5
348.          CJNE A,#00001000B,CH_K4
349.          MOV  A,#00001000B
350.          CALL SOUN_D
351.          ACALL DE_Y
352.          JMP  CH_SUC
353.  CH_K4:   MOV  A,#00000110B
354.          CALL SOUN_D1
355.          ACALL DE_Y
356.  CH_SUC:  RET

;*****
357.  RES_ET:  MOV  RE_1,#00H
358.          MOV  RE_2,#00H
359.          MOV  RE_3,#00H
360.          MOV  RE_4,#00H

```

```

361.          MOV  RE_5,#00H
362.          JMP  OUT_1;
-----
363.  DELAY:   MOV  R1,#01H          ; บรรทัดที่ 363-383
364.  DELAY1:  MOV  R6,#002H        ; เป็นโปรแกรมหน่วงเวลา
365.  DELAY2:  MOV  R7,#250D        ; มี 3 โปรแกรมคือ
366.  DELAY3:  DJNZ R7,DELAY3       ; DELAY,DE_Y และ
367.          DJNZ R6,DELAY2       ; DE_L ที่มี 3 อันเพราะ
368.          DJNZ R1,DELAY1       ; แต่ละอันมีการหน่วงเวลาไม่
369.          RET                   ; เท่ากัน
370.  DE_Y:    MOV  27H,#20D
371.  DE_Y1:   MOV  28H,#250D
372.  DE_Y2:   MOV  29H,#0FFH
373.  DE_Y3:   DJNZ 29H,DE_Y3
374.          DJNZ 28H,DE_Y2
375.          DJNZ 27H,DE_Y1
376.          RET
377.  DE_L:    MOV  2AH,#210D
378.  DE_1:    MOV  2BH,#255D
379.  DE_2:    MOV  2CH,#0FFH
380.  DE_3:    DJNZ 2CH,DE_3
381.          DJNZ 2BH,DE_2
382.          DJNZ 2AH,DE_1
383.          RET
384.  DATA_INI: DB  01H,02H,04H,08H ; ข้อมูลที่ใช้ในการเริ่ม
385.          DB  16D,32D,64D,128D   ; ให้ไมโครคอนโทรลเลอร์
                                     ; ทำงาน
386.          END                   ; จบโปรแกรม

```