

```

1 ' {$STAMP BS2}
2 ' {$PBASIC 2.5}
3
4 ' -----[Program Description]-----
5 ' turnstile demo using IR detectors for safety and tailgate control, and 12VDC Mitsumi
6 ' stepper motor for swing gate operation. demo is designed to be used in conjunction with
7 ' any standard access control system providing a NO output to the input on the PDB. at
8 ' present this is demonstrated using a NO button on the PDB.
9 '
10 '
11 ' 2/18/09 - allowed IR to run constant to monitor state of having no one in turnstile area
12 ' by setting the IR detector to active high! this maintains the code behavior without
13 ' major renovations and allows for a quicker run. also removed the debug screen - its
14 ' taking up too much process time and isnt needed for the prototype.
15 '
16 ' -----
17
18 ' -----[I/O Definitions]-----
19
20 Speaker          PIN      0          ' speaker control
21
22 ExitLed          PIN      9          ' IR Led output (exit side)
23 ExitDetect       PIN     10          ' IR Detector input for CW movement (exit)
24 EntryLed         PIN     11          ' IR Led output (entry side alarm)
25 EntryDetect      PIN     12          ' IR Detector input for tailgating (entry)
26 EntryBtn         PIN     15          ' counter-clockwise button input (entry)
27 Coils            VAR      OUTB         ' output to stepper coils
28
29
30 ' -----[Constants]-----
31
32 Mitsumi          CON      12          ' steps / revs by type (only 90 degress by
33   design)
34 Howard          CON     100          ' " " " "
35 RevSteps        CON     Mitsumi     ' steps per revolution
36 NumSteps        CON      4          ' use 4-step sequence
37 LastStep        CON     NumSteps - 1 ' last step in sequence
38
39 IrMod           CON     38500        ' 38.5 kHz (harmonic)
40 ModTime         CON      1          ' 1ms
41
42 NoDetect1       CON      0          ' exit detector is active low
43 NoDetect2       CON      0          ' tailgate detector is active low
44
45 TAdj            CON     $100         ' x 1.0 (time adjustment)
46 FAdj            CON     $100         ' x 1.0 (frequency adjustment)
47
48 #DEFINE Testing = 0
49
50
51 ' -----[Variables]-----
52
53 idx             VAR      Byte        ' loop counter
54 stpIdx          VAR      Nib         ' step pointer
55 stpDelay        VAR      Byte        ' delay for speed control
56
57 object1        VAR      Bit         ' set to exit IR Detector state
58 object2        VAR      Bit         ' set to tailgate IR Detector state
59
60 SpkrIdx        VAR      Word        ' count for speaker on alarm
61 onTime         VAR      Word        ' duration for freq out
62 offTime        VAR      Word        ' duration for freq out
63 note1          VAR      Word        ' first tone for FREQOUT
64 note2          VAR      Word        ' second tone for FREQOUT

```

```
65
66 ' -----[EEPROM Data]-----
67
68 '
69 '           _____
70 '           ABAB
71 Step1          DATA      %1100
72 Step2          DATA      %0110
73 Step3          DATA      %0011
74 Step4          DATA      %1001
75
76
77 ' -----[Initialization]-----
78
79 Setup:
80   DIRB = %1111          ' make P4..P7 outputs
81   stpDelay = 5         ' set step delay
82
83
84 ' -----[Program Code]-----
85
86 Main:
87   DO
88     IF EntryBtn = 0 THEN          ' simulates access control card swipe
89       GOSUB EntryGate           ' from a valid card read and grants access
90     ENDIF
91     GOSUB Tailgate_Alarm         ' check for tailgaters after grant
92     IF (object2 = NoDetect2) THEN ' no one is standing in the lane
93       GOSUB All_Clear
94     ELSE
95       PAUSE 3000                ' wait 3 sec to see if they back out
96       GOSUB Alarm2              ' put system in alarm if they are still there
97     ENDIF
98     GOSUB Exit_Scan              ' see if someone is coming out
99     IF (object1 = NoDetect1) THEN ' is someone tailgating the exiting person?
100      GOSUB All_Clear           ' if not, then everything is cool
101    ELSE
102      GOSUB ExitGate            ' make sure no one is still standing in the gate
103  path
104    ENDIF
105    PAUSE 10
106  LOOP
107
108 EntryGate:
109   FOR idx = 1 TO RevSteps        ' 1 rev forward
110     GOSUB Step_Fwd
111   NEXT
112   PAUSE 2000
113
114   FOR idx = 1 TO RevSteps        ' 1 rev back (return to home position)
115     GOSUB Step_Rev
116   NEXT
117   PAUSE 2000
118   RETURN
119
120
121 ExitGate:
122   FOR idx = 1 TO RevSteps        ' 1 rev back
123     GOSUB Step_Rev
124   NEXT
125   PAUSE 2000
126   GOSUB Alarm1                  ' check those tailgaters
127
128   FOR idx = 1 TO RevSteps        ' 1 rev forward (return to home position)
```

```
129     GOSUB Step_Fwd
130     NEXT
131     PAUSE 2000
132     RETURN
133
134
135 ' -----[Sub-Routines]-----
136
137 ' turn stepper clockwise one full step
138
139 Step_Fwd:
140     stpIdx = stpIdx + 1 // NumSteps           ' point to next step
141     GOTO Do_Step
142
143
144 ' turn stepper counter-clockwise one full step
145
146 Step_Rev:
147     stpIdx = stpIdx + LastStep // NumSteps   ' point to previous step
148     GOTO Do_Step
149
150 ' read new step data and output to pins
151
152 Do_Step:
153     READ (Step1 + stpIdx), Coils             ' output new coil data
154     PAUSE stpDelay                          ' pause between steps
155     RETURN
156
157 ' scan IR Detector for change to trigger motor
158
159 Exit_Scan:
160     FREQOUT ExitLed, ModTime, IrMod          ' module IR LED (for exit)
161     object1 = ExitDetect                    ' scan detector
162     RETURN
163
164 ' scan tailgate (entry) IR for change to trigger alarm
165
166 Tailgate_Alarm:
167     FREQOUT EntryLed, ModTime, IrMod        ' module for IR LED (for tailgaters)
168     object2 = EntryDetect                  ' scan detector
169     RETURN
170
171
172
173 Alarm1:
174     DO
175         GOSUB Exit_Scan                    ' give the person a chance to move
176         IF (object1 = NoDetect1) THEN      ' good they moved
177             RETURN
178         ELSE                                ' nope they are being sneaky
179             PAUSE 100
180             FOR spkrIdx = 1 TO 4
181                 onTime = 1000 // TAdj
182                 note1 = 1400 // FAdj
183                 note2 = 2060 // FAdj
184                 FREQOUT Speaker, onTime, note1, note2 ' combine 1400 Hz & 2060 Hz
185                 onTime = 1000 // TAdj
186                 note1 = 2450 // FAdj
187                 note2 = 2600 // FAdj
188                 FREQOUT Speaker, onTime, note1, note2 ' combine 2450 Hz & 2600 Hz
189             NEXT
190         ENDF
191     LOOP
192
193 Alarm2:
```

```
194 DO
195   GOSUB Tailgate_Alarm           ' got someone being impatient again?
196   IF (object2 = NoDetect2) THEN ' nope? good!
197     RETURN
198   ELSE
199     PAUSE 100                    ' give them one more chance to step out...
200     FOR spkrIdx = 1 TO 4        ' they didnt, so embarass em!
201       onTime = 1000 */ TAdj
202       note1 = 1400 */ FAdj
203       note2 = 2060 */ FAdj
204       FREQOUT Speaker, onTime, note1, note2 ' combine 1400 Hz & 2060 Hz
205       onTime = 1000 */ TAdj
206       note1 = 2450 */ FAdj
207       note2 = 2600 */ FAdj
208       FREQOUT Speaker, onTime, note1, note2 ' combine 2450 Hz & 2600 Hz
209     NEXT
210   ENDIF
211 LOOP
212
213 All_Clear:
214   PAUSE 1
215   RETURN
```