

```

1 ' -----[ Title ]-----
2 'Robotics with the Boe-Bot - IrRoamingAndEscapeCornersAndPhotoAndTempStore.bs2
3 'Modification of FastIrRoaming (added pulse command).
4 '
5 ' -----[ Notes ]-----
6 '
7 'Used rechargeable 1.25V batteries.
8 '
9 'The program change from FastIrRoamingAndPulsing.bs2 to the one now shown and is a
10 'significant improvement. The second problem is now solved.
11 '
12 '1st. The Boe-Bot has flipped over in the past. It gets beside at wall
13 '(angled into it) AND a wheel will drive up the side of a wall.
14 '
15 '2nd. The Boe-Bot cannot get out of corners.
16 '
17 '3rd. It keeps resetting itself. It keeps doing this because there is not enough
18 'energy TO power the Red LEDs, servos, emitters, and detectors. Solved this
19 'problem by turning off the Red LEDs.
20
21 ' -----[ Program ]-----
22
23 '{$STAMP BS2}                'Stamp directive.
24 '{$PBASIC 2.5}              'PBASIC directive.
25
26 DEBUG "Program Running!"
27
28 ' -----[ Variables/Constants/Pins ]-----
29
30 ' Stuff for IRroaming.
31 IrDetectorLeft    VAR    Bit
32 IrDetectorRight  VAR    Bit
33 PulseLeft        VAR    Word
34 PulseRight       VAR    Word
35 LoopCount        VAR    Byte
36 PulseCount       VAR    Byte
37
38 FreqDetectable   CON    38500
39
40 Piezospeaker     PIN    4           'Speaker
41 WheelRight       PIN    12          'Wheels left/right.
42 WheelLeft        PIN    13
43 IrDetectLeft     PIN    9           'Left emitter/detector pair
44 IrLedLeft        PIN    8
45 IrDetectRight    PIN    0           'Right emitter/detector pair
46 IrLedRight       PIN    2
47
48 ' Leaving corners
49 CounterRight     VAR    Nib
50 CounterLeft      VAR    Nib
51
52 ' Stuff for data collection
53 PRRight          PIN    3
54 TempProbe        PIN    15
55
56 Kal              CON    15300
57 rct              VAR    Word        ' A word variable.
58 TK              VAR    Word        ' Kelvin temperature.
59 TC              VAR    Word        ' Degrees Celsius.
60
61 TimeRight        VAR    Word
62
63 GetSensors       VAR    Word
64 counter          VAR    Byte
65
66 CurLoc           VAR    Word        'EEPROM location

```

```

67
68
69
70 ' -----[ Initialization ]-----
71
72 FREQOUT Piezospeaker, 2000, FreqDetectable      'Signal program start/reset.
73 CounterLeft = 1
74 CounterRight = 1
75 GetSensors = 0
76 CurLoc = 0
77
78 ' -----[ Main Routine ]-----
79
80 DO 'Main loop
81   GOSUB IrSensing
82   GOSUB Movements
83
84   '----- Get out of corners -----
85
86   IF (CounterLeft > 4) AND (CounterRight > 4) THEN
87     GOSUB Turn_Around
88     CounterLeft = 1
89     CounterRight = 1
90   ENDIF '(CounterLeft > 4) AND (CounterRight > 4)
91
92   '----- Sensors -----
93
94   GetSensors = GetSensors + 1
95   IF (GetSensors = 300) OR (GetSensors = 600) OR (GetSensors = 900) THEN
96     GOSUB DataLogging
97   ELSEIF (GetSensors = 902) THEN
98     GOSUB Ending
99   ENDIF
100
101 LOOP 'Main loop
102
103 Ending:
104   END
105
106 '-----[ Subroutines ]-----
107
108
109 '----- IR Sensing -----
110
111 IrSensing:
112   FREQOUT IrLedLeft, 1, FreqDetectable      'Store IR detection values.
113   IrDetectorLeft = IN9
114
115   FREQOUT IrLedRight, 1, FreqDetectable
116   IrDetectorRight = IN0
117   RETURN
118
119
120 '----- Movements -----
121
122 Movements:
123   IF (IrDetectorLeft = 0 ) AND (IrDetectorRight = 0) THEN
124     PulseCount = 15
125     PulseLeft = 650                'Back up.
126     PulseRight = 850
127   ELSEIF (IrDetectorLeft = 0) THEN   'Left LED detected.
128     CounterLeft = CounterLeft + 1
129     PulseCount = 10
130     PulseLeft = 850                'Turn Right.
131     PulseRight = 850
132   ELSEIF (IrDetectorRight = 0) THEN  'Right LED detected.

```

```

133 CounterRight = counterRight + 1
134 PulseCount = 10
135 PulseLeft = 650 'Turn Left.
136 PulseRight = 650
137 ELSE 'Go forward.
138 PulseCount = 1
139 PulseLeft = 850 'No IR detected, go forward.
140 PulseRight = 650
141 ENDIF '(IrDetectorLeft = 0 ) AND (IrDetectorRight = 0)
142
143 FOR LoopCount = 1 TO PulseCount
144 PULSOUT WheelLeft, PulseLeft
145 PULSOUT WheelRight, PulseRight
146 PAUSE 15
147 NEXT 'LoopCount = 1 TO PulseCount
148 RETURN
149
150
151 '----- Escape Corners -----
152
153 Turn_Around:
154 GOSUB Back_Up
155
156 FREQOUT Piezospeaker, 2000, 1000 ' Angerey noise.
157
158 GOSUB Turn_Left
159 GOSUB Turn_Left
160 RETURN 'IF (counter > 4)
161
162 Back_Up: ' Back up.
163 FOR PulseCount = 0 TO 40
164 PULSOUT WheelLeft, 650
165 PULSOUT WheelRight, 850
166 PAUSE 20
167 NEXT
168 RETURN 'Turn_Around:
169
170 Turn_Left: ' Left turn, about 90-degrees.
171 FOR PulseCount = 0 TO 22
172 PULSOUT WheelLeft, 650
173 PULSOUT WheelRight, 650
174 PAUSE 20
175 NEXT
176 RETURN 'Turn_Around:
177
178
179 '----- Data Logging -----
180
181 DataLogging:
182
183 DEBUG CR, "Start Data Loging", CR ' Used to figure out duration of data logging
184 FREQOUT Piezospeaker, 500, FreqDetectable
185
186 FOR counter = 1 TO 30 ' Main routine.
187 GOSUB PhotoReadings
188 GOSUB TempReadings
189 GOSUB SaveReadings
190 PAUSE 960
191 NEXT
192
193 DEBUG "Stop Data Loging", CR ' Used to figure out duration of data logging
194 RETURN
195
196
197 '----- Photo Data -----
198

```

```

199 PhotoReadings:
200
201   HIGH PRRight           ' Right RC time measurement.
202   PAUSE 2
203   RCTIME PRRight, 1, TimeRight
204
205 RETURN
206
207
208 '----- Temperature Data -----
209
210 TempReadings:
211   LOW TempProbe         ' Discharge the capacitor.
212   RCTIME TempProbe, 0, rct  ' Time for the volts to rise to 1.3 V.
213
214   TK = Kal/rct*10 + (Kal//rct*10/rct) ' Calculate Kelvin
215   TC = TK - 273           ' and Celsius.
216 RETURN
217
218
219 '----- Log Data -----
220
221 SaveReadings:
222   CurLoc = CurLoc + 1     'Increment EEPROM storage location
223   WRITE CurLoc, (GetSensors/300) 'Store location
224   CurLoc = CurLoc + 1     'Increment EEPROM storage location
225   WRITE CurLoc, TimeRight 'Store light readings
226   CurLoc = CurLoc + 1     'Increment EEPROM storage location
227   WRITE CurLoc, TC       'Store temperature readings
228 RETURN
229

```