

```
' =====  
'  
' File..... RoamingWithPING.bs2  
' Purpose... Roam And Scan Using PING))) Mounting Bracket  
' Author.... Parallax, Inc.  
' E-mail.... support@parallax.com  
' Started...  
' Updated... 12-21-2005  
'  
' {$STAMP BS2}  
' {$PBASIC 2.5}
```

```
' =====  
'  
' -----[ Program Description ]-----
```

```
' This programs demonstrates roaming with the Parallax PING))) Mou  
' Bracket, which allows you to pan-scan with your Parallax PING)))  
'
```

```
' This program assumes you have already set your BOE-Bot up for ro  
' with IR sensors. You must also have the PING))) Mounting bracke  
' installed as per the instructions, with the servo connected to P  
' the PING))) sensor connected to P15.  
'
```

```
' Due to the way the PING))) bracket mounts you may have to trim y  
' detectors down so that they don't stick up in the way of the mov  
' PING))) bracket as it rotates. It's best to angle them off at 4  
' degree angles so that the BOE-Bot doesn't run into anything whil  
' moving along a wall at a narrow angle.  
'
```

```
' The BOE-Bot will now roam around and when it senses an object, i  
' scan within a 180 degree field of view for the clearest path, wh  
' will then navigate scanning for an object again. The IR Sensors  
' hitting a wall when coming at it from a narrow angle where the P  
' might have trouble seeing it due to the lack of reflection of so  
'
```

```
' Calibration is important to this code working properly. If you  
' have the correct values for your servos then you may get strange  
' from your BOE-Bot. The necessary calibration information is lis  
' each section of the code where it is required.  
'
```

```
' -----[ Revision History ]-----
```

```
' This code is basically a modified version of the Roaming With IR
' from the Robotics With The BOE-Bot Manual, written by Andy Linds
' Much of the original code was left untouched so you can see how
' altered.
```

```
' -----[ I/O Definitions ]-----
```

```
Piezo          PIN      4          ' Piezo Speaker
RightServo     PIN      12         ' Right Servo
LeftServo      PIN      13         ' Left Servo
PingServo      PIN      14         ' PING))) Servo
Ping           PIN      15         ' PING))) Sensor
```

```
' -----[ Variables ]-----
```

```
irDetectLeft   VAR      BIT        ' Variable For Left IR Inp
irDetectRight  VAR      BIT        ' Variable For Right IR In
pulseCount     VAR      BYTE       ' Used For Measuring Turns
distance       VAR      WORD       ' Current Distance Of Obje
oldDistance    VAR      WORD       ' Old Distance Value
counter        VAR      WORD       ' PING))) Cycle Counter
task           VAR      NIB        ' Current Task
```

```
' -----[ Initialization ]-----
```

```
FREQOUT Piezo, 2000, 3000          ' Signal Program Start/Res
```

```
' -----[ Program Code ]-----
```

```
Main:
```

```
DO
```

```
FREQOUT 8, 1, 38500              ' Emit 38.5 kHz IR To Left
irDetectLeft = IN9                ' Store IR Detection Value
```

```
FREQOUT 2, 1, 38500              ' Emit 38.5 kHz IR To Righ
irDetectRight = INO                ' Store IR Detection Value
```

```

IF (irDetectLeft = 0) AND (irDetectRight = 0) THEN
    GOSUB Ping_Around           ' Object Detected via IR F
ELSEIF (irDetectLeft = 0) THEN
    GOSUB Ping_Around           ' Object Detected via IR L
ELSEIF (irDetectRight = 0) THEN
    GOSUB Ping_Around           ' Object Detected via IR R
ENDIF

counter = counter + 1           ' Increment Passive Counte

IF counter > 10 THEN         ' Wait For 10 Servo Pulses
    GOSUB Ping_Out             ' Activate PING)))
ENDIF

IF (distance > 30) THEN     ' Is Object Farther Than 3
    GOSUB Forward_Pulse       ' If Yes Go Forward
ELSE
    GOSUB Ping_Around         ' Otherwise Scan For Clear
ENDIF

LOOP

' -----[ Subroutines ]-----
' *****
' * USE THE APPROPRIATE PULSOUT VALUES TO MAKE YOUR BOE-BOT MOVE F
' * WHILE THE PING))) IS FACING FORWARD.
' *****

Forward_Pulse:             ' Send A Single Forward Pu
    PULSOUT PingServo, 750     ' Ping Servo Forward Pulse
    PULSOUT LeftServo, 850    ' Left Servo Forward Pulse
    PULSOUT RightServo, 650   ' Right Servo Forward Puls
    PAUSE 20                  ' Refresh Delay
    RETURN

' *****
' * USE THE APPROPRIATE PULSOUT VALUES TO MAKE YOUR BOE-BOT TURN L
' * DEGREES. USE THE SAME VALUE AS ABOVE FOR THE PING))) BRACKET
' *****

Turn_Left:                 ' Left Turn, About 45 Degr

```

```

FOR pulseCount = 0 TO 7           ' Number Of Pulses To Turn
  PULSOUT PingServo, 750          ' Ping Servo Forward Pulse
  PULSOUT LeftServo, 650         ' Left Servo Left Pulse Va
  PULSOUT RightServo, 650       ' Right Servo Left Pulse V
  PAUSE 20                       ' Refresh Delay
NEXT
RETURN

' *****
' * USE THE APPROPRIATE PULSOUT VALUES TO MAKE YOUR BOE-BOT TURN R
' * DEGREES. USE THE SAME VALUE AS ABOVE FOR THE PING))) BRACKET
' *****

Turn_Right:                       ' Right Turn, About 45 Deg
FOR pulseCount = 0 TO 7          ' Number Of Pulses To Turn
  PULSOUT PingServo, 750        ' Ping Servo Forward Pulse
  PULSOUT LeftServo, 850        ' Left Servo Right Pulse V
  PULSOUT RightServo, 850      ' Right Servo Right Pulse
  PAUSE 20                      ' Refresh Delay
NEXT
RETURN

' *****
' * USE THE APPROPRIATE PULSOUT VALUES TO MAKE YOUR BOE-BOT MOVE B
' * WHILE THE PING))) IS FACING FORWARD.
' *****

Back_Up:                          ' Back Up
FOR pulseCount = 0 TO 40        ' Number Of Pulses To Back
  PULSOUT PingServo, 750        ' Ping Servo Forward Pulse
  PULSOUT LeftServo, 650        ' Left Servo Backup Pulse
  PULSOUT RightServo, 850      ' Right Servo Backup Pulse
  PAUSE 20                      ' Refresh Delay
NEXT
RETURN

Ping_Out:                         ' PING)))
counter = 0                     ' Reset Passive Delay Coun
LOW Ping                         ' Force PING))) Line Low
PULSOUT Ping, 5                 ' Activate PING))) Pulse
PULSIN Ping, 1, distance        ' Receive Return Pulse
distance = distance ** 2257     ' Calculate Distance
RETURN

```

```

Ping_Around:                                ' Start 180 Degree Pan-Scan
counter = 0                                    ' Reset Passive Delay Counter
oldDistance = 30                              ' Current Old Distance Value
task = 0                                       ' Current Task Priority

' *****
' * USE THE APPROPRIATE PULSOUT VALUE TO MAKE YOUR PING)))
' * TURN 90 DEGREES LEFT.
' *****

FOR pulseCount = 0 TO 20                    ' Number Of Pulses To Spin
  LOW Ping                                    ' Force PING))) Line Low
  PULSOUT PingServo, 1085                    ' Ping Servo 90 Left Pulse
  PULSOUT Ping, 5                            ' Activate PING)))
  PULSIN Ping, 1, distance                  ' Receive Distance Value
  PAUSE 20                                   ' Refresh Delay
NEXT

distance = distance ** 2257                   ' Calculate Distance In cm
IF distance > oldDistance THEN             ' Is distance > Last Clear
  oldDistance = distance                     ' Update oldDistance Value
  task = 1
ENDIF

' *****
' * USE THE APPROPRIATE PULSOUT VALUE TO MAKE YOUR PING)))
' * TURN 45 DEGREES LEFT.
' *****

FOR pulseCount = 0 TO 20                    ' Number Of Pulses To Spin
  LOW Ping                                    ' Force PING))) Line Low
  PULSOUT PingServo, 850                    ' Ping Servo 45 Left Pulse
  PULSOUT Ping, 5                            ' Activate PING)))
  PULSIN Ping, 1, distance                  ' Receive Distance Value
  PAUSE 20                                   ' Refresh Delay
NEXT

distance = distance ** 2257                   ' Calculate Distance In cm
IF distance > oldDistance THEN             ' Is distance > Last Clear
  oldDistance = distance                     ' Update oldDistance Value
  task = 2
ENDIF

```

```

' *****
' * USE THE APPROPRIATE PULSOUT VALUE TO MAKE YOUR PING)))
' * TURN 45 DEGREES RIGHT.
' *****

FOR pulseCount = 0 TO 20           ' Number Of Pulses To Spin
  LOW Ping                          ' Force PING))) Line Low
  PULSOUT PingServo, 400            ' Ping Servo 45 Right Puls
  PULSOUT Ping, 5                   ' Activate PING)))
  PULSIN Ping, 1, distance          ' Receive Distance Value
  PAUSE 20                           ' Refresh Delay
NEXT

distance = distance ** 2257         ' Calculate Distance In cm
IF distance > oldDistance THEN     ' Is distance > Last Clear
  oldDistance = distance           ' Update oldDistance Value
  task = 3
ENDIF

' *****
' * USE THE APPROPRIATE PULSOUT VALUE TO MAKE YOUR PING)))
' * TURN 90 DEGREES RIGHT.
' *****

FOR pulseCount = 0 TO 20           ' Number Of Pulses To Spin
  LOW Ping                          ' Force PING))) Line Low
  PULSOUT PingServo, 225            ' Ping Servo 90 Right Puls
  PULSOUT Ping, 5                   ' Activate PING)))
  PULSIN Ping, 1, distance          ' Receive Distance Value
  PAUSE 20                           ' Refresh Delay
NEXT

distance = distance ** 2257         ' Calculate Distance In cm
IF distance > oldDistance THEN     ' Is distance > Last Clear
  oldDistance = distance           ' Update oldDistance Value
  task = 4
ENDIF

ON task GOSUB Task0, Task1, Task2, Task3, Task4

distance = 50                       ' Prevent Scan From Loopin

```

RETURN

Task0: ' Forward Was Clearest Pat
' This Could Mean Narrow P
GOSUB Turn_Right ' So We'll Turn Around
GOSUB Turn_Right ' You Can Change The Behav
GOSUB Turn_Right ' Of Any Of The Tasks
GOSUB Turn_Right
RETURN

Task1: ' 90 Degrees Left Was Clea
GOSUB Turn_Left
GOSUB Turn_Left
RETURN

Task2: ' 45 Degrees Left Was Clea
GOSUB Turn_Left
RETURN

Task3: ' 45 Degrees Right Was Cle
GOSUB Turn_Right
RETURN

Task4: ' 90 Degrees Right Was Cle
GOSUB Turn_Right
GOSUB Turn_Right
RETURN