

13. Brian Huge, (2005),
<http://www.ee.washington.edu/conselec/Sp96/projects/mitchrob/final/final.htm> ,
26 January 2006
14. William Aaron, (2006),
<http://www.singer-instruments.com/products/lvds.html> , 2 February 2007
15. Robert Mitchell, (2004),
<http://www.singer-instruments.com/products/lvds.html> , 18 Februari 2007
16. John Christiana, (2006),
<http://www.ratedsolutions.com/cgi-bin/bp/search.pl?keywords=home+security> ,
4 March 2007
17. Pamela Hersperger, (2005), <http://www.logix4u.net/paralleport1.htm> ,
28 March 2007
18. David A. George, (2001)
http://www.epanorama.net/circuits/parallel_output.html, 3 March 2007

Appendix A

Parts List for Detector Circuit

No	Parts	Quantity
1	Resistor 470 Ω	2
2	Resistor 10K Ω	1
3	Resistor 4.7K Ω	1
4	Resistor 2K Ω	1
5	Resistor 1.2K Ω	1
6	Diode 1N4148	1
7	IR LED and Detector	1 set
8	Transistor 2N3904	1
9	LED	1

*note that all component due to 1 set sensor

Appendix B

Source Code for Module

Option Explicit

```
Public Declare Sub PortOut Lib "io.dll" (ByVal Port As Integer, ByVal Value As Byte)
Public Declare Function PortIn Lib "io.dll" (ByVal Port As Integer) As Byte
```

```
Declare Function sndPlaySound Lib "winmm.dll" Alias "sndPlaySoundA" (ByVal
lpszSoundName As String, ByVal uFlags As Long) As Long
Public Declare Function waveOutGetNumDevs Lib "winmm" () As Long
```

```
Global Const SND_SYNC = &H0 'just after the sound is ended exit function
Global Const SND_ASYNC = &H1 'just after the beginning of the sound exit function
Global Const SND_NODEFAULT = &H2 'if the sound cannot be found no error message
Global Const SND_LOOP = &H8 'repeat the sound until the function is called again
Global Const SND_NOSTOP = &H10 'if currently a sound is played the function will
return without playing the selected sound
```

```
Global Const Flags& = SND_ASYNC Or SND_NODEFAULT
```

Appendix C

Source Code for Form 1 (Opening)

```
Private Sub Form_Load()  
Left = (Screen.Width - Width) \ 2  
Top = (Screen.Height - Height) \ 2  
  
Shape1.BackColor = vbWhite  
End Sub  
  
Private Sub Image4_Click()  
  
End Sub  
  
Private Sub Timer1_Timer()  
If Shape1.BackColor = vbWhite Then  
Shape2.Visible = True  
Shape3.Visible = False  
Shape4.Visible = False  
Shape5.Visible = False  
Shape6.Visible = False  
Shape7.Visible = False  
Shape8.Visible = False  
Shape9.Visible = False  
Shape10.Visible = False  
End If  
End Sub  
  
Private Sub Timer10_Timer()  
Timer9.Enabled = False  
If Shape10.Visible = True Then  
Form1.Hide  
  
Form1.Enabled = False  
Form2.Show  
Timer10.Enabled = False  
End If  
End Sub  
  
Private Sub Timer11_Timer()  
'Static Count As Integer  
If (p1.Left + p1.Width) <= 0 Then  
p1.Left = Form1.Width
```

```

End If

    p1.Left = p1.Left - 100
End Sub

Private Sub Timer12_Timer()
'Static Count As Integer
    If (p2.Right + p2.Width) <= 0 Then
        p2.Right = Form1.Width

    End If

    p2.Right = p2.Right - 100
End Sub

Private Sub Timer2_Timer()
Timer1.Enabled = False
If Shape2.Visible = True Then
Shape2.Visible = True
Shape3.Visible = True
Shape4.Visible = False
Shape5.Visible = False
Shape6.Visible = False
Shape7.Visible = False
Shape8.Visible = False
Shape9.Visible = False
Shape10.Visible = False
End If
End Sub

Private Sub Timer3_Timer()
Timer2.Enabled = False
If Shape3.Visible = True Then
Shape2.Visible = True
Shape3.Visible = True
Shape4.Visible = True
Shape5.Visible = False
Shape6.Visible = False
Shape7.Visible = False
Shape8.Visible = False
Shape9.Visible = False
Shape10.Visible = False
End If

```

End Sub

```
Private Sub Timer4_Timer()  
Timer3.Enabled = False  
If Shape4.Visible = True Then  
Shape2.Visible = True  
Shape3.Visible = True  
Shape4.Visible = True  
Shape5.Visible = True  
Shape6.Visible = False  
Shape7.Visible = False  
Shape8.Visible = False  
Shape9.Visible = False  
Shape10.Visible = False  
End If  
End Sub
```

```
Private Sub Timer5_Timer()  
Timer4.Enabled = False  
If Shape5.Visible = True Then  
Shape2.Visible = True  
Shape3.Visible = True  
Shape4.Visible = True  
Shape5.Visible = True  
Shape6.Visible = True  
Shape7.Visible = False  
Shape8.Visible = False  
Shape9.Visible = False  
Shape10.Visible = False  
End If  
End Sub
```

```
Private Sub Timer6_Timer()  
Timer5.Enabled = False  
If Shape6.Visible = True Then  
Shape2.Visible = True  
Shape3.Visible = True  
Shape4.Visible = True  
Shape5.Visible = True  
Shape6.Visible = True  
Shape7.Visible = True  
Shape8.Visible = False  
Shape9.Visible = False  
Shape10.Visible = False  
End If  
End Sub
```

```
Private Sub Timer7_Timer()  
Timer6.Enabled = False  
If Shape7.Visible = True Then  
Shape2.Visible = True  
Shape3.Visible = True  
Shape4.Visible = True  
Shape5.Visible = True  
Shape6.Visible = True  
Shape7.Visible = True  
Shape8.Visible = True  
Shape9.Visible = False  
Shape10.Visible = False  
End If  
End Sub
```

```
Private Sub Timer8_Timer()  
Timer7.Enabled = False  
If Shape8.Visible = True Then  
Shape2.Visible = True  
Shape3.Visible = True  
Shape4.Visible = True  
Shape5.Visible = True  
Shape6.Visible = True  
Shape7.Visible = True  
Shape8.Visible = True  
Shape9.Visible = True  
Shape10.Visible = False  
End If  
End Sub
```

```
Private Sub Timer9_Timer()  
Timer8.Enabled = False  
If Shape9.Visible = True Then  
Shape2.Visible = True  
Shape3.Visible = True  
Shape4.Visible = True  
Shape5.Visible = True  
Shape6.Visible = True  
Shape7.Visible = True  
Shape8.Visible = True  
Shape9.Visible = True  
Shape10.Visible = True  
End If  
End Sub
```

Appendix D

Source Code for Form 9 (Main Page)

```
Private Sub Command1_Click()  
Form9.Hide  
Form4.Show  
End Sub
```

```
Private Sub Command2_Click()  
Form9.Hide  
Form11.Show  
End Sub
```

```
Private Sub Command3_Click()  
Form9.Hide  
Form6.Show  
End Sub
```

```
Private Sub Command4_Click()  
Form9.Hide  
Form8.Show  
End Sub
```

```
Private Sub Command5_Click()  
End  
End Sub
```

```
Private Sub Command6_Click()  
Form9.Hide  
Form13.Show  
End Sub
```

```
Private Sub Form_Load()  
Left = (Screen.Width - Width) \ 2  
Top = (Screen.Height - Height) \ 2  
End Sub
```

```
Private Sub Image1_Click()  
  
End Sub
```

```
Private Sub Timer1_Timer()
```



```
'Static Count As Integer  
  If (p1.Left + p1.Width) <= 0 Then  
    p1.Left = Form1.Width
```

```
  End If
```

```
  p1.Left = p1.Left - 100  
End Sub
```

Appendix E

Product Details



[Larger photo](#)

Infrared LED

IR503C-B20Y1 Infrared LED with Reverse and Forward Current

[INQUIRE NOW](#)

[ADD TO BASKET](#)

[CALL NOW - FREE](#)

[View basket](#)

The time there is
07:44

Chat with supplier using: 

Country of Origin:

- China (mainland)

Main Export Markets:

- Eastern Europe
- North America
- Mid East/Africa
- Central/South America
- Asia
- Western Europe
- Australasia

Key Specifications/Special Features:

- View angle: 20 degree
- Size
- Diameter: 4.98mm
- Height: 8.7mm
- Lens color: water clear
- Forward voltage: 1.25 to 1.45V (test condition $I_F=20\text{mA}$)
- Spectral bandwidth: 45nm (test condition $I_F=20\text{mA}$)
- Reverse current: 10uA
- Forward current: 20mA
- Radiant intensity: 50 mW

Appendix F

Product Details



[Larger photo](#)

Country of Origin:

- China (mainland)

Main Export Markets:

- Eastern Europe
- North America
- Mid East/Africa
- Central/South America
- Asia
- Western Europe
- Australasia

Infrared Emitting Diode
IR523B-W50Q1 Infrared Emitting Diode with Blue
Transparent Plastic Packing

[INQUIRE NOW](#)

[ADD TO BASKET](#)

[CALL NOW - FREE](#)

[View basket](#)

The time there is
07:45

Chat with supplier using: 

Key Specifications/Special Features:

- Size: 5.0 x 8.7 mm
- Lens color: water clear (blue)
- Forward voltage: 1.18 to 1.20V
- Radiant intensity: 28 to 34mW/sr
- Peak wavelength: 940nm
- Viewing angle: 30 to 50 degrees