

## **BLUETOOTH**

### **¿Qué significa Bluetooth?**

Hay diversas maneras de conectar dispositivos electrónicos entre sí, mediante cables, señales de radio y rayos de luz infrarrojos, y una variedad incluso mayor de conectores, enchufes y protocolos, por lo que el arte de conectar cosas es cada día más complejo, de ahí la necesidad de la tecnología inalámbrica (wireless). La tecnología Bluetooth es automática e inalámbrica, y tiene un número de características interesantes que pueden simplificar nuestra vida diaria.

### **¿Qué es Bluetooth?**

El Bluetooth Special Interest Group (SIG), una asociación comercial formada por líderes en telecomunicación, informática e industrias de red, está conduciendo el desarrollo de la tecnología inalámbrica Bluetooth y llevándola al mercado.

La tecnología inalámbrica Bluetooth es una tecnología de ondas de radio de corto alcance (2.4 gigahertzios de frecuencia) cuyo objetivo es el simplificar las comunicaciones entre dispositivos informáticos, como ordenadores móviles, teléfonos móviles, otros dispositivos de mano y entre estos dispositivos e Internet. También pretende simplificar la sincronización de datos entre los dispositivos y otros ordenadores.

Permite comunicaciones, incluso a través de obstáculos, a distancias de hasta unos 10 metros. Esto significa que, por ejemplo, puedes oír tus mp3 desde tu comedor, cocina, cuarto de baño, etc. También sirve para crear una conexión a Internet inalámbrica desde tu portátil usando tu teléfono móvil. Un caso aún más práctico es el poder sincronizar libretas de direcciones, calendarios etc en tu PDA, teléfono móvil, ordenador de sobremesa y portátil automáticamente y al mismo tiempo.

Los promotores de Bluetooth incluyen Agere, Ericsson, IBM, Intel, Microsoft, Motorola, Nokia y Toshiba, y centenares de compañías asociadas.

### **¿De dónde viene el nombre Bluetooth?**

El nombre viene de Harald Bluetooth, un Vikingo y rey de Dinamarca a de los años 940 a 981, fue reconocido por su capacidad de ayudar a la gente a comunicarse. Durante su reinado unió Dinamarca y Noruega.

### **Objetivos principales de la tecnología Bluetooth.**

1. Permitir la comunicación sencilla entre dispositivos fijos y móviles.
2. Evitar la dependencia de cables que permitan la comunicación.
3. Permitir la creación de pequeñas redes de forma inalámbrica.

### **¿Qué se puede hacer con los productos con tecnología Bluetooth?**

Las posibilidades son casi ilimitadas, pero a continuación enumeramos algunas de las posibilidades actuales:

- Eliminación de la necesidad de conexiones por cable entre los productos y accesorios electrónicos.
- Intercambio de archivos, tarjetas de visita, citas del calendario, etc. entre usuarios de Bluetooth.
- Sincronización y transferencia de archivos entre dispositivos.
- Conexión a determinados contenidos en áreas públicas.
- Como mandos a distancia funcionan como llave, entradas y monederos electrónicos.

### **¿En qué clases de productos se puede esperar encontrar la tecnología Bluetooth?**

La tecnología inalámbrica Bluetooth es única en su amplitud de usos. Los acoplamiento se pueden establecer entre grupos de productos simultáneamente o entre productos individuales con Internet.

Esta flexibilidad, además de que los productos con tecnología Bluetooth tienen que ser calificados y pasar pruebas de interoperabilidad por el Bluetooth Special Interest Group antes de su lanzamiento, ha hecho que una amplia gama de segmentos de mercado soporte esta tecnología, incluyendo técnicos de software, vendedores de silicio, fabricantes de periféricos y cámaras fotográficas, fabricantes de PCs móviles y técnicos de dispositivos de mano, fabricantes de coches, y fabricantes de equipos de pruebas y medidas.

### **Funcionamiento.**

Trabaja en dos capas del modelo OSI que son la de enlace y aplicación, incluye un transeiver que transmite y recibe a una frecuencia de 2.4 GHz Las conexiones que se realizan son de uno a uno con un rango máximo de 10 metros, si se deseara implementar la distancia se tendría que utilizar repetidores los cuales nos ayudarían a abarcar una distancia de 100 metros.

Bluetooth por cuestiones de seguridad cuenta con mecanismos de encriptación de 64 bits y autenticación para controlar la conexión y evitar que dispositivos puedan acceder a los datos o realizar su modificación.

El transmisor está integrado en un pequeño microchip de 9 x 9 milímetros y opera en una frecuencia de banda global. Los dispositivos que incorporan esta tecnología se reconocen entre si y utilizan el mismo lenguaje de la misma forma que lo realizan otros dispositivos como lo son la computadora y la impresora.

Durante la transferencia de datos el canal de comunicaciones permanece abierto y no requiere la intervención directa del usuario cada vez que se desea transferir voz o datos de un dispositivo a otro. La velocidad máxima que se alcanza durante la transferencia es de 700 Kb/s y consume un 97% menos que un teléfono móvil.

### **Arquitectura de hardware.**

Está compuesto por dos partes la primera de ellas es un dispositivos de radio que es el encargado de modular y transmitir la señal, un controlador digital que a su vez está compuesto por un procesador de señales digitales llamado link controller, una CPU que es el encargado de atender las instrucciones del

Bluetooth del dispositivo anfitrión, esto se logra gracias link manager que es un software el cual tiene como función permitir la comunicación con otros dispositivos por medio del protocolo LMP.

Entre las tareas realizadas por el link controller y link manager destacan el envío y recepción de datos, empaginamiento y peticiones, determinación de conexiones, autenticación, negociación y determinación de tipos de enlace, determinación del tipo de cuerpo de cada paquete y ubicación del dispositivo en modo sniff o hold.

### **Arquitectura de software.**

Se utilizan protocolos de alto nivel como SDP que es un protocolo que permite detectar otros dispositivos en el rango de comunicación permitido, otro protocolo utilizado es RFCOMM que permite emular la conexión de un puerto serial y TCS que es un protocolo de control de telefonía, todos estos protocolos interactúan entre sí para tener comunicación con el controlador de banda base través del protocolo L2CAP que es el encargado de la segmentación y re ensamble de los paquetes y a su vez envía los paquetes de mayor tamaño a través de la conexión Bluetooth.

### **Transmisión.**

Bluetooth está diseñado para usar acuses de recibos y saltos de frecuencias lo que permite tener conexiones robustas, lo cual es una ventaja muy grande porque permite ayudar a los problemas de interferencia y a su vez añade seguridad.

Esta transmisión puede ser realizada de manera síncrona o asíncrona. El método síncrona es orientado a conexión de voz que es conocido como SCO, y la conexión asíncrona que es utilizada para la transmisión de datos y es conocida como ACL. La división de tiempo dúplex es usado para este tipo de conexiones los cuales soportan 16 tipos de paquetes, cuatro de ellos son paquetes de control y son los mismos en cada tipo de conexión. Debido a la necesidad de tranquilidad en la transmisión de datos, los paquetes son enviados en grupos sin interrumpir otras transmisiones que se estén realizando en ese momento.

## **Protocolos de conexión.**

Las conexiones Bluetooth son establecidas a través de la siguiente técnica.

1. Standby. Cuando los dispositivos están en modo de reposo ellos escuchan mensajes cada 1.8 segundos sobre 32 saltos de frecuencia.
2. Page/inquirí. Permite el envío de un paquete denominado page que permite realizar la conexión con otro dispositivo, y si el receptor de este page contesta se comienza con la transferencia de datos.
3. Active. Permite la transmisión de datos.
4. Hold. Permite realizar la conexión sin necesidad de transferir datos la finalidad de esto es conservar el poder entre el master y el slave, siempre y cuando así se desee.
5. Sniff. Esta técnica solo es aplicada a unidades slave y permite conservar el poder, durante este modo el slave no toma un rol activo pero escucha a un nivel reducido.
6. Park. Este es un modo más reducido, que el modo hola, durante este modo el slave es sincronizado a la piconet, lo cual permite no requerir un reactivación completa, y no es parte del tráfico.

## **¿Cuáles son las diferencias entre Wi-Fi y la tecnología de radio Bluetooth?**

Las tecnologías inalámbricas Bluetooth y Wi-Fi son tecnologías complementarias.

La tecnología Bluetooth se diseña para sustituir los cables entre los teléfonos móviles, ordenadores portátiles, y otros dispositivos informáticos y de comunicación dentro de un radio de 10 metros.

Un router típico con Wi-Fi puede tener un radio de alcance de 45 m en interiores y 90 m al aire libre.

### **Chat a través del uso de Bluetooth - Código**

```
Imports System.Runtime.InteropServices
Public Class Form1
Inherits System.Windows.Forms.Form
#Region " Windows Form Designer generated code "

Public Sub New()
MyBase.New()
'This call is required by the Windows Form Designer.
InitializeComponent()
'Add any initialization after the InitializeComponent() call
End Sub
Protected Overrides Sub Dispose(ByVal disposing As Boolean)
MyBase.Dispose(disposing)
End Sub

'NOTE: The following procedure is required by the Windows Form Designer
'It can be modified using the Windows Form Designer.
'Do not modify it using the code editor.

'Friend WithEvents InputPanel1 As Microsoft.WindowsCE.Forms.InputPanel
'Friend WithEvents MainMenu1 As System.Windows.Forms.MainMenu
'Friend WithEvents Button1 As System.Windows.Forms.Button
'Friend WithEvents txtMessage As System.Windows.Forms.TextBox
'Friend WithEvents txtMessageLog As System.Windows.Forms.TextBox
'Friend WithEvents MenuItem1 As System.Windows.Forms.MenuItem
'Friend WithEvents MenuItem2 As System.Windows.Forms.MenuItem
'Friend WithEvents MenuItem3 As System.Windows.Forms.MenuItem

<System.Diagnostics.DebuggerStepThrough> Private Sub InitializeComponent()
Me.MainMenu1 = New System.Windows.Forms.MainMenu
Me.MenuItem1 = New System.Windows.Forms.MenuItem
```

```
Me.txtMessage = New System.Windows.Forms.TextBox
Me.Button1 = New System.Windows.Forms.Button
Me.txtMessageLog = New System.Windows.Forms.TextBox
Me.MenuItem2 = New System.Windows.Forms.MenuItem
Me.MenuItem3 = New System.Windows.Forms.MenuItem

'MainMenu1
'
Me.MainMenu1.Items.Add(Me.MenuItem1)
'
'MenuItem1
'
Me.MenuItem1.Items.Add(Me.MenuItem2)
Me.MenuItem1.Items.Add(Me.MenuItem3)
Me.MenuItem1.Text = "Settings"
'
'txtMessage
'
Me.txtMessage.Location = New System.Drawing.Point(8, 8)
Me.txtMessage.Size = New System.Drawing.Size(224, 22)
Me.txtMessage.Text = ""
'
'Button1
'
Me.Button1.Location = New System.Drawing.Point(160, 40)
Me.Button1.Text = "Send"
'
'txtMessageLog
'
Me.txtMessageLog.Location = New System.Drawing.Point(8, 64)
Me.txtMessageLog.Multiline = True
Me.txtMessageLog.ScrollBars = System.Windows.Forms.ScrollBars.Vertical
Me.txtMessageLog.Size = New System.Drawing.Size(224, 192)
Me.txtMessageLog.Text = ""
'
'MenuItem2
'
Me.MenuItem2.Text = "Connect"
```

```

'
'MenuItem3
'
Me.MenuItem3.Text = "Disconnect"
'
'Form1
'
Me.Controls.Add(Me.txtMessageLog)
Me.Controls.Add(Me.Button1)
Me.Controls.Add(Me.txtMessage)
Me.Menu = Me.MainMenu1
Me.Text = "Chat"
End Sub
#End Region

Dim infileHandler As Long
Dim outfileHandler As Long
Dim numReadWrite As Integer
Dim t1 As System.Threading.Thread
Dim stopThread As Boolean = False
Public Sub connect()
'---port number for Bluetooth connection
Dim inPort As Short = 7
Dim outPort As Short = 8
'---Opens the port for Bluetooth
infileHandler = CreateFile("COM" & inPort & ":", _
&HC0000000, 0, 0, 3, 0, 0)
Application.DoEvents()
outfileHandler = CreateFile("COM" & outPort & ":", _
&HC0000000, 0, 0, 3, 0, 0)
Application.DoEvents()
'---invoke the thread to receive incoming messages
stopThread = False
t1 = New Threading.Thread(AddressOf receiveLoop)
t1.Start()
End Sub

```

```
Public Sub disconnect()  
stopThread = True  
CloseHandle(infileHandler)  
CloseHandle(outfileHandler)  
End Sub
```

```
Private Sub Form1_Load(ByVal sender As System.Object, _  
ByVal e As System.EventArgs) _  
Handles MyBase.Load  
End Sub  
Public Function stringToByteArray(ByVal str As String) As Byte()  
'---e.g. "abcdefg" to {a,b,c,d,e,f,g}  
Dim s As Char()  
s = str.ToCharArray  
Dim b(s.Length - 1) As Byte  
Dim i As Integer  
For i = 0 To s.Length - 1  
b(i) = Convert.ToByte(s(i))  
Next  
Return b  
End Function
```

```
Function byteArrayToString(ByVal b() As Byte) As String  
'---e.g. {a,b,c,d,e,f,g} to "abcdefg"  
Dim str As String  
Dim enc As System.Text.ASCIIEncoding  
enc = New System.Text.ASCIIEncoding  
str = enc.GetString(b, 0, b.Length())  
Return str  
End Function  
<DllImport("coredll.dll")> _
```

```
Private Shared Function CreateFile(ByVal lpFileName As String, _  
ByVal dwDesiredAccess As Integer, _  
ByVal dwShareMode As Integer, _  
ByVal lpSecurityAttributes As Integer, _  
ByVal dwCreationDisposition As Integer, _  
ByVal dwFlagsAndAttributes As Integer, _
```

```
ByVal hTemplateFile As Integer) As Integer
End Function
<DllImport("coredll.dll")> _
```

```
Private Shared Function ReadFile(ByVal hFile As Integer, _
ByVal Buffer() As Byte, _
ByVal nNumberOfBytesToRead As Integer, _
ByRef lpNumberOfBytesRead As Integer, _
ByRef lpOverlapped As Integer) As Integer
End Function
<DllImport("coredll.dll")> _
```

```
Private Shared Function WriteFile(ByVal hFile As Integer, _
ByVal Buffer() As Byte, _
ByVal nNumberOfBytesToWrite As Integer, _
ByRef lpNumberOfBytesWritten As Integer, _
ByVal lpOverlapped As Integer) As Boolean
End Function
<DllImport("coredll.dll")> _
```

```
Private Shared Function CloseHandle(ByVal hObject As Integer) As Integer
End Function
Public Function send(ByVal message As String) As Integer
'---send the message through the serial port
Dim value As String = message & vbCrLf
Dim retCode As Integer = WriteFile(outfileHandler, _
stringToArray(value), _
value.Length(), _
numReadWrite, _
0)
txtMessageLog.Text += value
Return retCode
End Function
Public Sub receiveLoop()
'---receive the message through the serial port
Dim inbuff(300) As Byte
Dim retCode As Integer = ReadFile(infileHandler, _
inbuff, _
```

```

inbuff.Length, _
numReadWrite, _
0)
Application.DoEvents()
While True
If retCode = 0 Or stopThread Then
'MsgBox("Error reading message.")
Exit While
Else
Dim updateDelegate As New _
myDelegate(AddressOf updateMessageLog)
updateDelegate.Invoke(byteArrayToString(inbuff))
ReDim inbuff(300)
retCode = ReadFile(infileHandler, _
inbuff, _
inbuff.Length, _
numReadWrite, _
0)
Application.DoEvents()
End If
End While
End Sub

Private Sub Button1_Click(ByVal sender As System.Object, _
ByVal e As System.EventArgs) _
Handles Button1.Click
If send(txtMessage.Text) = 0 Then
MsgBox("Error sending message.")
End If
End Sub

Public Delegate Sub myDelegate(ByVal str As String)
Public Sub updateMessageLog(ByVal str As String)
If str.Length > 0 Then
txtMessageLog.Text += "-->" & str
End If
End Sub

Private Sub MenuItem1_Click(ByVal sender As System.Object, _
ByVal e As System.EventArgs) _

```

```
Handles MenuItem1.Click
```

```
End Sub
```

```
Private Sub MenuItem2_Click(ByVal sender As System.Object, ByVal e As  
System.EventArgs) Handles MenuItem2.Click
```

```
connect()
```

```
MenuItem2.Enabled = False
```

```
MenuItem3.Enabled = True
```

```
End Sub
```

```
Private Sub MenuItem3_Click(ByVal sender As System.Object, ByVal e As  
System.EventArgs) Handles MenuItem3.Click
```

```
disconnect()
```

```
MenuItem2.Enabled = True
```

```
MenuItem3.Enabled = False
```

```
End Sub
```

```
End Class
```

