

---

## SISTEMA PORTÁTIL DE MEDICIÓN NO INVASIVA DE GLUCOSA SANGUÍNEA POR MEDIOS ESPECTROFOTOMÉTRICOS BASADO EN PDA Y PRINCIPIOS DE INTELIGENCIA ARTIFICIAL.

### PARTE I I: SISTEMAS DE CÓMPUTO MÓVILES

*Israel Rivera Zárate*

#### 1. Handhelds y PDAs.

Las handhelds y PDAs, se denominan dos clases completamente diferentes de computadoras de mano, son las más chicas de ellas y no poseen disco duro: almacenan los archivos y programas en una tarjeta el cual se inserta en la ranura de expansión de memoria, las handhelds son más grandes y pesadas, tienen teclado y módem incorporado y tienen un costo superior a las PDAs.

Las PDAs dependen del touch screen y usan un método de ingreso de datos la escritura a mano sobre la pantalla mediante un lápiz especial, llamado *stylus*, estos equipos fueron pensados para almacenar datos de manera temporal que irán a parar a la computadora de escritorio en la cual habrán de almacenarse por un mayor tiempo, sin embargo hoy en día las computadoras móviles cuentan con procesadores capaces de responder a cálculos más complejos con los cuales pueden generarse aplicaciones con un grado de complejidad elevado sobre todo para el campo de la medicina.

Para transmitir información las handhelds y PDAs vienen con cables de conexión, llamados "cunas" (cradle) de sincronización (donde se posiciona la computadora) el cual tiene la posibilidad de conectarse por el puerto serie o uso a la computadora de escritorio, estas también cuentan con el software necesario para realizar este trabajo, en la Palm esto es posible gracias la tecnología HotSync, mientras que en los equipos con Windows CE (PDAs) esta función se denomina Active Sync.

#### 2. Palm OS y Windows CE diferencias y similitudes.

En cuanto al numero de usuarios las Palms mantienen la preferencia (según datos de International Data Consulting IDC) el Palm OS acapara el 83 por ciento del mercado, mientras Windows CE araña el 9.7 %.

Dentro de las cualidades principales de las Palms se encuentra la austeridad y simplicidad ya que su interfaz gráfica es sencilla y muy ligera, es importante destacar el consumo de memoria de este sobre su rival Windows CE (ahora Windows mobile o Pocket PC) el cual requiere 16 mb para arrancar mientras que Palm OS opera desde los 2 MB de memoria, por lo mismo

existen en promedio un mayor numero de aplicaciones generadas para las palms que para Windows mobile.

Tradicionalmente las Palm eran más una "agenda supervitaminada" y las PocketPC más un "portátil recortado", de ahí que hasta hace poco las Palm no tenían nada que hacer en multimedia, pero a cambio eran muy, muy estables, sus aplicaciones consumían pocos recursos y sus baterías duraban mucho. No obstante en los últimos tiempos (PalmOS 5) las prestaciones multimedia de las Palm han mejorado, pero han perdido en estabilidad y autonomía.

Por otra parte, Windows Mobile se diferencia por ofrecer a sus usuarios un ambiente más familiar ya que permite trabajar con versiones reducidas de las aplicaciones más populares de las computadoras de escritorio como son Microsoft Office y algunos lenguajes de programación como Pocket C y Visual Basic Embedded, Windows también ofrece soporte a audio, por lo que se pueden grabar voces y escucharlas a través de micrófonos incorporados y entradas y salidas auxiliares.

### **3. Herramientas para diseñadores móviles**

Con la creciente popularidad de los dispositivos informáticos móviles existe una demanda cada vez mayor de desarrolladores que puedan diseñar de una vez aplicaciones que puedan ejecutarse en varios dispositivos. Es decir, no es lo mismo desarrollar una simple agenda telefónica, que un complejo programa de recuperación de información con informes impresos vía infrarrojos conectado todo ello a una fuente de datos externa tipo SQL Server.

A continuación presentamos los entornos de programación más sobresalientes del mercado.

#### **3.1 CASL software.**

CASL es un ambiente integrado de desarrollo en cual incluye variables, objetos y funciones para desarrollar poderosas y flexibles aplicaciones con capacidades como las que de describen a continuación:

- Colección de objetos de visualización con control dinámico de de localización, tamaño, fuente, colores
- Acceso vía puerto serie e infrarrojos.
- Acceso a socket de Red (telnet, http, tcp, udp o iNet web clipping).
- Canalización para sincroniza bases de datos PDA a través de PC companion db vía texto o a través de ODBC format2.

Objetos Grid con columnas, los cuales pueden ser llenados con etiquetas, cajas de texto, casillas de verificación destinadas a campos de bases de datos o arreglos para la fácil visualización de información desplazable sin escribir ningún código adicional.

(Precio: De 134.95 a \$334.95 dólares.) [1]

### 3.2 EMbedded Visual Tools 3.0

Entorno visual oficial de Microsoft para el desarrollo de aplicaciones, eMbedded Visual Tools es un completo entorno de desarrollo para crear aplicaciones y componentes de sistema para dispositivos Windows Powered, incluyendo sistemas PocketPC y Smartphone.

Los que estén familiarizados con Visual Basic o Visual C, entrarán en el mismo mundo pero con las limitaciones lógicas del dispositivo, razón por la cual es una de las herramientas más populares, esto en parte a la multitud de herramientas adicionales que contiene como el Spy, editor de registro, emulador (solo funciona en NT con SP5 ), etc.

En contraparte, la mayor dificultad que presenta este software es la falta de documentación y los errores constantes "bugs" y "crashes" que se hacen presentes en el momento de generar una aplicación.

La edición 2003 contiene Microsoft eMbedded Visual C++ 3.0, Microsoft eMbedded Visual Basic 3.0 y, como novedad, también presenta los kits de desarrollo (SDKs) para PocketPC 2002 y Smartphone 2002.

Requerimientos mínimos:

Windows 98/NT/2000 [2], [3]

### 3.3 OrbWorks PocketC

Características:

- Lenguaje simple de aprender y programar
- Compilador no nativo, permitiendo la compartición de applets
- Conversión automática de tipos de datos para facilitar la programación
- Amplia librería de funciones que incluye gráficos, sonido, database, archivos, y I/O serie
- Tipo de dato "String"
- Genera ejecutable independiente para cada CPU de un tamaño muy pequeño.
- Alocación dinámica de memoria
- Soporta objetos COM

Plataformas soportadas

- Windows2000, ME, NT, 98, 95
- Pocket PC Casio Cassiopeia E-115, EM-500, E-125
- HP Jornada 540 Series y Compaq iPAQ
- HP Jornada 720 Series, y dispositivos con WindowsCE OS HPC 2000
- HP Jornada 820, NEC Mobil Pro, Sharp TriPad PV6000, Sharp Mobilon PV5000, Vadem Clio, y dispositivos con WindowsCE OS HPCPro 2.11
- Palm-size PC como Compaq 2100, HP 428 etc. y dispositivos con WindowsCE OS PPC 2.11.
- Palm-size PC como Casio E10, E11, Nino 301,312, 320 etc. y dispositivos con WindowsCE OS PPC 2.01
- Philips Velo500, etc. y dispositivos con WindowsCE OS HPC 2.00.

#### Observaciones

Este lenguaje no es ni tan rápido ni tan potente como el C++, pero su facilidad de uso y su bajo precio compensa estos "problemas". Funciona generando código nativo que necesita de un Runtime, de libre distribución y 100K de tamaño.

Precio: EUR 25.70 / USD 29.99 [4]

## 4. Estudio de la Herramienta Seleccionada.

### 4.1 Fundamentos de Microsoft Visual eMbedded Visual

Microsoft eMbedded Visual Embedded cuenta con un ambiente integrado de desarrollo con el objeto de facilitar la creación de aplicaciones para dispositivos que usan Windows CE. Esta compuesto por un conjunto de ventanas, herramientas, menús, directorios y otros elementos con el fin de crear, probar y depurar una aplicación diseñada para Windows CE.

A continuación presentaremos de una manera resumida, una visión global del modelo de programación de Windows CE y los componentes que soportan este modelo. También de describirán las bases de trabajar con el ambiente de desarrollo IDE.

### 4.2 Modelo de programación de Windows CE

El sistema operativo Windows CE esta basado en la interfaz de programación para aplicaciones Win32 (API)<sup>[1]</sup>.

Como en los demás sistemas operativos de Windows, Windows CE posee un modelo de programación orientado a eventos.,un programa basado en Windows CE recibe mensajes, interpreta mensajes y actúa sobre los mensajes.

Estos programas tienen una o mas ventanas que reciben y procesan mensajes en un ciclo, las ventanas pueden ser visuales o no, en el caso en que la aplicación no requiera una interfaz de usuario, cada ventana posee un manejador de ventana (hwnd) asociado con un procesador de mensajes que maneja el mensaje para la ventana que corresponda.

Así mismo, los programas diseñados para Windows CE poseen dos funciones primarias, un procesador de mensajes (llamado WndProc) y WndMain, el cual provee un punto de entada hacia el programa. La función WndProc procesa mensajes para la ventana.

En general, una aplicación procesa exclusivamente aquellos mensajes que son relevantes a ella y transmite los otros mensajes de regreso al sistema operativo. Además de ser el proceso de mensaje de primario WinMain también maneja la inicialización y el cierre

Cuando se desarrollan programa para Windows CE, se debe determinar primero la plataforma de hardware y el procesado en cual va a correr tu aplicación. Por ejemplo Handheld PC, Palm-size PC, Pocket PC, Auto PC o la plataforma de juegos Dreamcast; los procesadores pueden ser ARM, MIPS, SH3, SH4, x86, o x86em.

## 5. Adquisición y Despliegue de la señal.

Primeramente se explicaran cada uno de los pasos asi como de las ventanas que aparecen al momento de abrir un nuevo proyecto en Visual Basic Embedded esto con el fin de introducirnos de una manera clara y sutil en nuestra aplicación:

Paso numero 1: Al momento de abrir Microsoft Embedded Visual Basic 3.0 aparece la siguiente pantalla permitiéndonos seleccionar entre abrir un proyecto nuevo, buscar un proyecto ya existente para trabajar en el o abrir un proyecto en el que se ha trabajado recientemente.

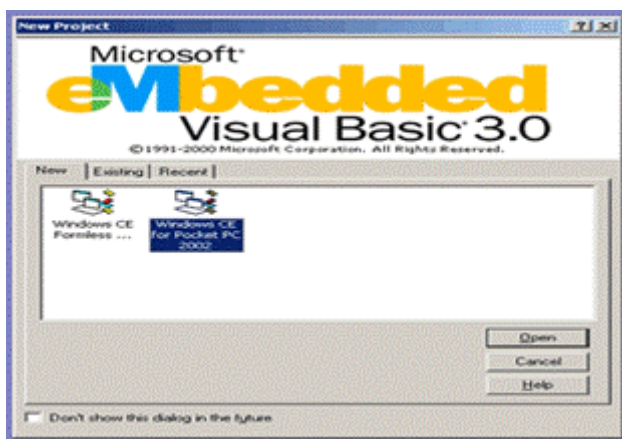


Figura 1. Explorador de proyecto

Paso numero 2: Abriendo un nuevo proyecto en Visual Basic Windows Ce for Pocket PC 2002 se desplegara la siguiente pantalla, en la cual se han resaltado cuatro regiones: 1) Explorador de Proyectos (2 Ventana de Propiedades (3 Barra de herramientas estándar (4 Área de trabajo donde se intercambian la ventana de código y la ventana del diseño de formulario.

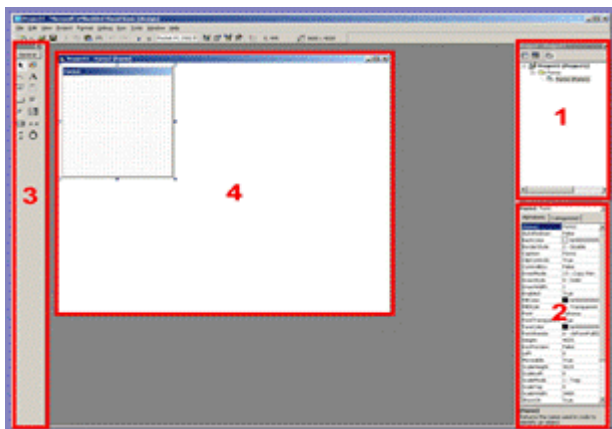


Figura 2. Área de trabajo

Paso numero 3: Configuramos nuestro proyecto accediendo al menú Project - Project Properties - General. Ahí asignaremos un nombre a nuestro proyecto (Project1) y en el apartado Remote Settings - Remote Path: cambiamos la ruta por omisión por la siguiente:

Windows/Inicio\Project1.vb

Paso numero 4: Ahora bajaremos la aplicación a nuestra Pocket PC (Compaq Ipaq en este caso) de manera que podamos probarla. Para esto hay que seleccionar en la parte superior del IDE a la opción de Pocket PC 2002 (Default Device) y presionamos el boton Play. Cualquier programa generado en visual Basic Embedded esta dividido en dos partes, la ventana de código y la ventana de objetos, para mantener un orden lógico de las cosas se explicara primeramente el código hecho para nuestra aplicación seguido de cada uno de los objetos que conforman la aplicación misma:

Para un mejor entendimiento del programa este se ha separado en partes:

### Parte numero 1:

Se declaran y se inicializan las variables globales que habrán de utilizarse durante la ejecución del programa, estas son:

- 1.- InputData es la variable de memoria donde se almacena el dato leído del puerto serie de la PDA.
- 2.- Cantidad es la variable de memoria donde se almacena el producto del dato leído con 0.0196 esto con el fin de tener valores en el rango de 0 a 5 y así poder graficarlos directamente sobre la IPAQ.
- 3.- iniciox, inicio2x son las variables iniciales de las coordenadas X que toma cada punto dentro del plano X, Y sobre la IPAQ.

4.- inicio es la variable inicial de la coordenada Y que toma cada punto dentro del plano X, Y sobre la IPAQ.

5.- bandera nos señala si es un primer dato adquirido o no, en caso de ser así se toman las coordenadas iniciales X, Y.

6. incremento es la variable que dicta el incremento de la variable x en el eje X del plano X, Y.

6.- contador es la variable que nos indica cuantos datos leídos llevamos.

7.- cuenta es la variable que indica al programa cuando tiene que borrar la pantalla de la aplicación para así seguir graficando los datos nuevos leídos del puerto.

### **Parte numero 2:**

La parte 2 del programa esta conformada por el subprocedimiento llamado Adquirir\_señal el cual se encarga de establecer los siguientes parámetros:

1.- Determina los valores iniciales para las variables cuenta e incremento que determinan la definición del grafico de la señal adquirida.

2.-Abre el puerto serie de la IPAQ ya sea para transmitir o recibir datos, esto a través de la instrucción:

```
PortOpen = True
```

3.-Establece los parámetros para el objeto de Visual Basic Embedded llamado COMM el cual es el encargado de realizar el protocolo de comunicación RS-232 entre la IPAQ y el PIC 16F73, a continuación se muestra este objeto así como cada uno de los parámetros utilizados en esta aplicación:

a.- RThreshold, esta propiedad establece y regresa el numero de caracteres a recibir antes de que se genere el comm event, colocando en 1 a esta propiedad se habilita la generación de comm event..

b.- SThreshold , esta propiedad establece el numero mínimo de caracteres a recibir en el buffer de transmisión antes de que se genere el comm event, colocando en 0 a esta propiedad se inhabilita la transmisión de datos hacia fuera del puerto de la PDA.

c.- InputLen, esta propiedad establece el número de caracteres a leer del buffer de datos recibidos, colocádola en 0 se lee entero el registro de datos recibidos.

d.- DTREnable, esta propiedad habilita o deshabilita la señal DTR en el Puerto serie de la IPAQ, colocándolo en 0 se deshabilita esta señal del puerto.

4.- Establece los parámetros para el objeto de Visual Basic Embedded llamado Picture Box

a.- ScaleHeight, esta propiedad establece y regresa el número de unidades de la medida vertical dentro del objeto.

b.- ScaleLeft , esta propiedad establece las coordenadas horizontales de la orilla izquierda del objeto.

c.- ScaleTop, esta propiedad coloca el tope vertical dentro del objeto picture box.

d.- ScaleWidth , esta propiedad coloca el tope horizontal dentro del objeto picture box.

### **Parte numero 3:**

La parte 2 del programa esta conformada por el subprocedimiento llamado Cerrar\_puerto el cual se encarga de establecer los siguientes parámetros:

1.- Determina los valores para los objetos Frec\_Altas, Frec\_Bajas después de haber adquirido ya una señal.

2.-Cierra el puerto serie de la IPAQ, esto a través de la instrucción:

```
PortOpen = False
```

3.-limpia la pantalla de la PDA con la siguiente instrucción después de haber adquirido un número de datos determinados

```
pantalla.Cls
```

### **Parte numero 4:**

La parte 4 del programa esta conformada por el subprocedimiento llamado Form Load el cual se encarga al momento de abrir la aplicación de cargar el fondo de la misma, esto se realizo así que cada vez que se necesitaba borrar la pantalla para graficar nuevos datos con la función Cls esta borraba también los ejes dejando así la aplicación con una pantalla blanca.

Para cargar un archivo con extensión bmp dentro de un Picture Box es necesario almacenar el archivo en el directorio donde se descargara la aplicación, en nuestro caso fue:

```
pantalla.Picture = "Windows/Inicio/juanis.bmp" ' donde pantalla.Picture es el nombre de nuestro objeto picture box, "Windows/Inicio/" es la ruta donde se descarga la aplicación y juanis.bmp es el nombre del archivo de nuestra imagen.
```

### **Parte numero 5:**

La sección 5 del programa es la encargada de recibir los datos y de graficarlos en pantalla, la recepción se realiza a través de la propiedad del objeto Comm llamada CommEvent esta propiedad toma diferentes códigos en función del evento generado durante el protocolo de comunicación RS-232 entre la IPAQ y el microcontroladorla.

## 6. Ventana de objetos.

La ventana de objetos de nuestra aplicación es la siguiente:



Podemos observar que esta pantalla esta conformada por 5 objetos de visual Basic Embedded cada uno con diferentes propiedades que a continuación se muestran, estas no se abordan de una manera muy profunda ya que ya han sido tratadas con anterioridad:



### Objeto 1.- Picture Box

El control Picture Box es un control ActiveX usado para desplegar bitmaps, graficas y texto, este tiene propiedades y métodos para definir estilos de dibujo, colores y tamaños de las graficas.

En adición el control Picture Box puede responder al manejo de un mouse o de un lápiz táctil que escriba sobre la pantalla.

## **Objeto 2.- Command Button**

El objeto command button nos sirve para introducir datos a través de la pantalla, este objeto puede usarse para la entrada de datos con el ratón, o para validar cualquier operación, el tamaño puede cambiarse a voluntad, pero la forma siempre es rectangular. A continuación se muestran las propiedades más importantes de este objeto:

### 1.- Name Nombre:

Es el nombre que define a este objeto durante la ejecución del programa, no se puede cambiar en tiempo de ejecución. El nombre por defecto es Command1, Command2, etc. Estos nombres por defecto deberían cambiarse por otros que nos indiquen algo respecto a este control, aunque en este caso no es tan imprescindible como para los formularios.

### 2.-Caption Título:

Es el texto que aparece en el botón, puede cambiarse en tiempo de ejecución.

### 3.-Enabled Habilitado:

Propiedad Booleana que habilita o deshabilita el botón, cuando está deshabilitado (Enabled = False), el botón no tiene efecto, y su apariencia varía, presentando un tono pálido tanto en su cuerpo como en su título. Puede variarse en tiempo de ejecución

## **Objeto 3.- Label (Etiqueta)**

Una etiqueta es un control que nos permite presentar un texto, la etiqueta debe usarse en aquellos casos en los que exista una información estática o dinámica que no deba ser cambiada por el operador. A continuación se muestran las propiedades más importantes de este objeto:

### 1.-Appearance Apariencia:

Es la dimensión en la que se muestra el objeto plano o tridimensional.

### 2.-BackColor Color de fondo:

Es el color de fondo de la etiqueta, puede cambiarse en tiempo de ejecución.

Objeto 4.- Option Button

ESTADO	INICIANDO	PRESIONADO
<ul style="list-style-type: none"><li>● Apagado</li><li>● Prendido</li><li>● Deshabilitado</li></ul>	<input type="radio"/> <input type="radio"/> <input checked="" type="radio"/>	<input type="radio"/> <input type="radio"/> <input type="radio"/>

Objeto 5.- Comm

El control Comm Box es un control invisible ActiveX que provee a una conexión serial al puerto, esta aplicación puede ser usada para conectar diferentes dispositivos, los cuales como computadoras, modems, sistemas GPS que son conectados a nuestra PDA. Cada control Comm corresponde a un puerto serie de la PDA, si necesitamos acceder a mas de un puerto serie es necesario agregarmas de un Comm Control.

## 6. Referencias

[1] <http://www.caslsoft.com>

[2] <http://www.microsoft.com/windows/embedded/ce/tools/factsheet.asp>

[3] <http://www.microsoft.com/mobile/downloads/emvt30.asp>

[4] <http://www.orbworks.com/wince>

---

<sup>[1]</sup>API de sus siglas en inglés por Interfaz de Programación de Aplicaciones.