# Recursos pedagógicos en las asignaturas de electrónica: animaciones y objetos interactivos

Julio Brégains Dpto. Ing. de Computadores, CITIC, GTEC Universidade da Coruña A Coruña, España julio.bregains@udc.es José M. Andión Dpto. Ing. de Computadores, CITIC, GAC Universidade da Coruña A Coruña, España jose.manuel.andion@udc.es

Paula M. Castro Dpto. Ing. de Computadores, CITIC, GTEC Universidade da Coruña A Coruña, España paula.castro@udc.es

*Abstract*— En este trabajo se presentan técnicas para crear representaciones de dispositivos y sistemas, consistentes en gráficas animadas y objetos interactivos, que pueden incluirse en presentaciones orientadas a la enseñanza de la electrónica.

Keywords— animaciones, objetos interactivos, PowerPoint, VBA, recursos pedagógicos.

# I. INTRODUCCIÓN

electricidad y la electrónica se basan, La fundamentalmente, en el estudio y el aprovechamiento de la interacción entre dispositivos dinámicos. Dicha dinámica puede referirse, [1], a las relaciones entre parámetros medibles -como la relación entre la corriente eléctrica a través de un diodo y el voltaje entre sus terminales-, o a la variación temporal de dichos parámetros -como el aumento o la disminución de la carga de un condensador en un circuito RC-. Mientras que el lenguaje matemático permite cuantificar y representar simbólicamente dichas relaciones, y el lenguaje verbal es capaz de describirlas en detalle, la comprensión de las ideas involucradas alcanza su punto culminante al complementar ambos lenguajes con gráficas interactivas y/o dotadas de animación.

En el ámbito académico, Microsoft PowerPoint [2] está establecido como un software estándar para la creación y edición de presentaciones [3], y es muy utilizado para la transmisión de conocimientos en prácticamente cualquier campo científico-técnico. En los casos más elementales, su uso se limita a la creación de diapositivas con párrafos de extensión reducida, quizás mezclados con listas con viñetas. Esta práctica, naturalmente, no aprovecha las numerosas potencialidades de dicha aplicación. Si se desean resultados más efectivos, el programa cuenta con una amplia variedad de herramientas que contribuyen a captar con mayor intensidad la atención del estudiantado. Este abanico de opciones va desde la configuración de objetos propios del programa -como figuras planas, cuadros de texto o imágenes- para que ejecuten acciones ante pulsaciones del ratón [4], hasta la posibilidad de usar herramientas más avanzadas basadas en macros<sup>1</sup> o en la codificación con un lenguaje de programación denominado Visual Basic para Aplicaciones<sup>2</sup> (VBA, [5]). La combinación de objetos interactivos y macros editadas con VBA conforma la herramienta más avanzada de PowerPoint con la que es posible crear sistemas virtuales complejos para representar, por ejemplo, circuitos con símbolos, esquemas realistas, funciones matemáticas simbólicas, figuras geométricas (que incluyen curvas con ejes cartesianos) o resultados numéricos. Esto representa una gran ayuda para el equipo docente ya que, con ellas, el estudiantado puede observar, por una parte, cómo funcionan sistemas con más de un dispositivo –es decir, cuál es la relación entre ellos–, y, por otra, cuál es la utilidad de las fórmulas matemáticas de las leyes físicas que gobiernan dichos sistemas.

Otra opción que ofrece PowerPoint es la de dotar de movimiento a los objetos en las diapositivas. De este modo, las variaciones temporales de los parámetros (en forma de curvas), o entes físicos (posiciones, tamaños, etc.), pueden ser representadas en detalle. Esto constituye una opción invaluable a la hora de transmitir conceptos dinámicos de un modo explícito.

En esta comunicación se presentan algunas técnicas que permiten aprovechar estas herramientas para crear, por un lado, gráficas interactivas de circuitos –incluyendo gráficas de curvas y relaciones matemáticas entre unidades representativas– y, por otro, animaciones que muestran cómo se comportan algunas entidades –así como la variación de sus parámetros– en función del tiempo.

## II. DESCRIPCIÓN DEL TRABAJO

A continuación, describimos, en líneas generales, la metodología utilizada basada en el uso de las herramientas de Microsoft (MS) PowerPoint 365 [6].

Las gráficas interactivas se diseñan utilizando formas planas editables (rectángulos, líneas y círculos), que pueden obtenerse desde la pestaña "Insertar" del propio programa, como se puede ver en la Fig.1, combinadas con gráficas vectoriales (por ejemplo, de extensión svg (Scalable Vector Graphics)<sup>3</sup> obtenidas externamente (mediante un editor, como Microsoft Visio [9]). Esto permite configurar los esquemas de los circuitos: no solo los simbólicos, sino también los que simulan la apariencia física de los dispositivos reales. La interactividad de estos objetos se establece con la codificación en VBA, incluyendo, en este caso, botones, casillas de verificación, cuadros de texto y botones de opción pertenecientes a la familia de controles ActiveX [5], como se puede observar en la Fig. 1. Estos controles son característicos de aplicaciones bajo el sistema operativo Windows 10, aunque son similares en otros sistemas operativos basados en ventanas (sistemas operativos basados en eventos). Las gráficas cartesianas complementarias (para representar, por ejemplo, curvas o rectas corriente-voltaje) se diseñan utilizando objetos

<sup>&</sup>lt;sup>1</sup> Una macro es una secuencia de operaciones que se almacenan para ser ejecutadas posteriormente con, por ejemplo, una pulsación de un botón o una combinación de teclas.

<sup>&</sup>lt;sup>2</sup> De hecho, es posible crear o editar una macro utilizando directamente VBA. Ambas herramientas son complementarias.

<sup>&</sup>lt;sup>3</sup> Las svg [7] son gráficos vectoriales que no pierden calidad al aplicarles zoom, en contraposición a lo que les sucede a las imágenes pixeladas, tipo "bitmap" [8].



Fig. 1. Arriba: conjunto de algunos objetos disponibles en la pestaña "Insertar" de MS PowerPoint 365. Centro: botones de herramientas de programación con macros y VBA. Abajo: conjunto de controles ActiveX. Estos dos últimos conjuntos de botones se encuentran en la pestaña "Programador" de PowerPoint [5].



Fig. 2. Breve muestra de las herramientas disponibles en la pestaña "Animaciones" [2].

tipo chart del mismo conjunto de controles. La programación con VBA permite, por lo tanto, obtener una simulación numérica cuyos parámetros son controlables mediante botones y cuadros de texto que el usuario (en este caso, quien utiliza las presentaciones con fines pedagógicos) puede manipular.

Por otra parte, las gráficas con animaciones se obtienen utilizando herramientas propias del editor de presentaciones [2], [4], disponibles en la pestaña "Animaciones", y aprovechando el "Panel de animación" con el que es posible establecer diversos parámetros como: instante de inicio, duración, tipo de animación, etcétera, de los objetos involucrados. En la Fig. 2 se observa un pequeño conjunto de las herramientas disponibles para tal fin.

En la siguiente sección se presentan algunos de los resultados obtenidos con el uso de las herramientas descritas.

# III. RESULTADOS OBTENIDOS.

Como ejemplos de uso, se presentan primeramente dos circuitos interactivos. La Fig. 3 muestra dos diapositivas en las que se han configurado circuitos básicos. En la Fig. 3 (A) se observa una configuración para enseñar la ley de Ohm desde cuatro puntos de vista diferentes. En la parte superior izquierda se presenta el esquema similar a cómo se vería en la realidad, consistente en una fuente de alimentación conectada a una resistencia, con la posibilidad de incluir en el diagrama un voltímetro y un amperímetro. En la parte inferior izquierda se incluye la fórmula de la ley de Ohm, con un pequeño diagrama cartesiano para recordar cuál es la representación de una recta que pasa por el origen. En la parte inferior central se encuentra un esquemático, en el que se muestran las flechas de voltajes y corrientes, junto con sus símbolos y los valores obtenidos por cálculo. En la parte superior derecha se puede ver el diagrama corriente-voltaje con dos líneas rectas: la azul corresponde a la obtenida con el valor nominal de la resistencia, mientras que la roja corresponde a la línea con el valor real de la resistencia (en la que se considera un error dentro de los límites de tolerancia de dicho dispositivo). Los círculos rojos sobre las líneas indican los puntos de trabajo (nominal y real) obtenidos con los valores específicos del voltaje de la fuente y las corrientes obtenidas con dicho voltaje. Finalmente, en la parte de abajo, a la derecha, se encuentran diversos controles para elegir el valor nominal de la resistencia (valores normalizados) o su tolerancia, e indicar si se quieren incluir en el esquema real los dibujos del voltímetro (para medir la diferencia de potencial en los extremos del resistor) y el amperímetro (para medir la correspondiente corriente). También se incluyen dos botones para alternar entre mostrar la ecuación (como se observa) o la tabla que se utiliza para obtener el valor nominal del resistor de acuerdo a sus bandas de colores. Cabe señalar que, al seleccionar el valor nominal de la resistencia desde la lista desplegable, en el dibujo del resistor se actualizan correspondientemente los colores de sus bandas. El valor del voltaje puede elegirse escribiéndolo con el teclado directamente en la pantalla (display) del diagrama de la fuente de voltaje variable, o desplazando el cursor horizontal (slide) que se encuentra debajo de dicha pantalla.

En la Fig. 3 (B) se observa un circuito básico conteniendo (según se observa en la esquina superior izquierda) una fuente de corriente continua variable, conectada a un diodo Zéner ideal en serie con un resistor. En este caso, el valor del resistor permanece fijo, y los parámetros modificables son el voltaje de la fuente de alimentación (a través de su display o su cursor horizontal) o los voltajes umbral (tensión de codo) e inversa (tensión Zéner) del diodo (estos valores se establecen en los cuadros de texto ubicados en la parte inferior derecha de la diapositiva). A la derecha se observa la curva teórica del diodo (modelo lineal por segmentos). El modelo equivalente del Zéner se modifica convenientemente de acuerdo a su punto de funcionamiento (reemplazando el diodo, según corresponda, por su tensión Zéner, un circuito abierto o su voltaje umbral). Nótese que este cambio se realiza no solo en el circuito (esquemático), sino también en el símbolo incluido en uno de los cuadrantes de los ejes cartesianos corriente-voltaje.





Fig.3. (A): esquema interactivo de un circuito básico enseñando la ley de Ohm. (B): esquema interactivo de un circuito con diodo Zéner ideal.



Fig. 4. Animación para describir el comportamiento de carga del condensador en un circuito RC básico.









Fig. 5. Tres instantes en la secuencia de animación para enseñar, paso a paso, cómo se completa la tabla de verdad de una puerta lógica. (A) Instante en el que se ha completado la primera fila de la tabla. (B) Instante en el que se ha completado la tabla. (C) Instante final, en el que se observa la tabla de verdad simplificada, y el símbolo correspondiente al circuito. Para más detalles, ver texto.

Los respectivos voltímetros indican las correspondientes diferencias de potencial en el diodo y en la resistencia, y un amperímetro indica la corriente que circula por la única malla del circuito. Como muestra del proceso de codificación de esta herramienta interactiva, al final del artículo se incluye un Apéndice con fragmentos de código correspondientes a tres subrutinas que permiten controlar el comportamiento de algunos de los objetos ActiveX [5] de la correspondiente transparencia.

En la Fig. 4 se tiene la animación de un circuito RC conectado a una fuente de alimentación de voltaje constante, con el condensador en proceso de carga. Con esta animación se enseña no solo el proceso físico idealizado (para el movimiento de cargas se utiliza la corriente convencional en lugar de la corriente electrónica), sino la variación de los diversos parámetros conforme transcurre el tiempo. Específicamente, se observan las curvas de carga eléctrica y voltaje en el condensador, así como de la corriente y del voltaje en la resistencia, todos ellos como funciones exponenciales temporales. Esta animación puede utilizarse en combinación con el planteamiento de la correspondiente ecuación diferencial que gobierna el comportamiento de los parámetros del circuito.

Finalmente, se ha incluido la Fig. 5 como muestra de la utilización de animación para enseñar las secuencias establecidas en los comportamientos de los transistores MOSFET complementarios (CMOS) configurando una puerta lógica. A la izquierda de cada una de ellas se observan tres cuadros (debajo del subtítulo "Example I") como resumen de la secuencia a seguir para obtener la tabla de verdad del circuito: considerar los transistores como interruptores en pares complementarios -cuando uno se enciende, su complementario se apaga, y viceversa-, dar valores (0 voltios o V<sub>SS</sub> voltios) a las entradas, y, finalmente, de acuerdo con los estados obtenidos, obtener el circuito lógico equivalente. La gráfica superior (A) muestra el instante en el que se ha completado la primera fila de la tabla, y en la segunda se han hallado los estados de los transistores complementarios  $Q_1$  y  $Q_3$ . En la gráfica central (B) se observa el contenido de la diapositiva en el instante en el que se ha completado la tabla, y se indica que los transistores  $Q_1$  y  $Q_2$  configuran una red "Pull-down" (nótese que los transistores han sido reemplazados por interruptores). Finalmente, en la gráfica (C) se enseña la tabla de verdad simplificada, y el símbolo correspondiente a la puerta lógica que constituye el circuito CMOS en estudio (en este caso, una puerta NAND). Aunque esta figura no corresponde precisamente a la utilización de parámetros que varían en función del tiempo, sirve como ejemplo adicional para indicar que las herramientas de animación en asignaturas técnicas tienen aplicaciones prácticamente ilimitadas.

#### IV. DISCUSIÓN Y CONCLUSIONES.

A pesar de las evidentes ventajas de emplear este tipo de herramientas en las presentaciones utilizadas por el equipo docente en la difusión de contenidos de electricidad y electrónica, es necesario puntualizar sobre algunos de sus inconvenientes. Uno de ellos es la relativamente pronunciada curva de aprendizaje para llegar a manejar con cierta fluidez estas opciones avanzadas (i.e., objetos editables, animaciones, controles especiales y el lenguaje VBA) de PowerPoint. El otro, no menos importante, es el tiempo de edición necesario para obtener estos resultados: para un usuario experimentado, una sola de estas transparencias puede suponer entre 4 y 12 horas de edición (fundamentalmente para diseñar las figuras, editar sus tamaños y posiciones, ajustar los tiempos y trayectorias de animación, escribir el código -en el caso de uso de VBA- y depurarlo, además de hacer frente a las constantes incompatibilidades que surgen al completar esta tarea). Finalmente, el uso de código VBA hace que estos archivos no puedan compartirse siempre de un modo sencillo. Por un lado, la versión online de PowerPoint (que se ejecuta con un navegador) no soporta macros. Y, por otro, como la ejecución de macros puede suponer un riesgo a la seguridad informática (ya que muchos códigos maliciosos pueden ejecutarse mediante el uso de estas extensiones<sup>4</sup>), algunas veces estos ficheros pueden verse bloqueados por la propia aplicación (algo que se soluciona configurando correctamente las opciones de seguridad de PowerPoint<sup>5</sup>).

#### AGRADECIMIENTOS

Este trabajo ha sido financiado por la Xunta de Galicia (proyectos ED431C 2020/15, ED431C 2021/30 y ED431G 2019/01 para el apoyo del Centro de Investigación de Galicia "CITIC"), la Agencia Estatal de Investigación de España (proyectos RED2018-102668-T y PID2019-104958RB-C42) y fondos ERDF de la UE (FEDER Galicia 2014-2020 y Programas AEI/FEDER).

### REFERENCIAS

- [1] R. Boylestad, *Electronic Devices and Circuit Theory*, Pearson, 2014.
- [2] Microsoft, "Ayuda y aprendizaje de PowerPoint," <u>https://support.microsoft.com/es-es/powerpoint</u> (acceso: 15/05/22).
- [3] J. E. Susskind, "PowerPoint's Power in the Classroom: Enhancing Students' Self-efficacy and Attitudes", Computers and Education, 45(2), 2005. <u>https://doi.org/10.1016/j.compedu.2004.07.005</u>.
- [4] C. Jordan, Advanced ICDL presentations: a step-by-step guide to advanced presentations using Microsoft PowerPoint, Independently Published, 2020.
- [5] Microsoft, "Referencia de VBA para PowerPoint," <u>https://docs.microsoft.com/es-es/office/vba/api/overview/powerpoint</u> (acceso: 15/05/22).
- [6] M. Vic, "Microsoft PowerPoint 365 for Beginners 2021", Amazon, 2021.
- [7] Wikipedia, "Scalable Vector Graphics", https://en.wikipedia.org/wiki/Scalable\_Vector\_Graphics (acceso: 15/05/22).
- [8] Wikipedia, "Bitmap", <u>https://en.wikipedia.org/wiki/Bitmap</u> (acceso: 15/05/22).
- [9] S. Helmers, Microsoft Visio 2016 Step By Step, Microsoft Press, 2016.

<sup>&</sup>lt;sup>4</sup> En esta comunicación se ha planteado cómo utilizar macros programadas en VBA para modificar la propia presentación que las contiene. Sin embargo, VBA también permite realizar otras acciones fuera del propio archivo, como, por ejemplo, el borrado de ficheros, la descarga y posterior ejecución de programas maliciosos, o el envío de claves (contraseñas). Por ello, solo deben ejecutarse aquellas macros provenientes de fuentes confiables.

<sup>&</sup>lt;sup>5</sup> Esto se realiza en el propio PowerPoint accediendo a la ventana de configuración, mediante la secuencia de menús: Archivo → Opciones → Centro de confianza → Configuración del Centro de confianza... → Configuración de macros. Cuando se instala el programa, la configuración que deshabilita automáticamente las macros está establecida por defecto.

## APÉNDICE

Como ejemplos del código utilizado para la simulación del circuito con diodo Zéner (ver Sección III), a continuación presentamos tres porciones de código (subrutinas).

 a) Líneas de código para inicializar los parámetros de la diapositiva correspondiente al simulador del circuito con diodo Zéner (las líneas en verde, que se inician con el carácter de apóstrofe, son comentarios que el intérprete de VBA no ejecuta [5]):

```
Sub InitializeD()
'Subroutine for initializing
Zener_Diode_function slide parameters
NameD = "Zener_Diode" 'Choose that slide
NameDS = "Zener_Diode_data" 'Data table name
Vprec = 0.009 'Voltage precision
nVDmin = 2 'Min value of VD slide
nVDmax = 201 'Max value of VD slide
Ntan = 25 'Tan parameter for VD
End Sub
```

#### *b)* Subrutina que permite mostrar u ocultar el amperímetro en la diapositiva:

```
Private Sub AmmeterICheck Change()
Call InitializeD 'Set the initial values
'Select the corresponding slide from the
active presentation
Set mydocument =
ActivePresentation.Slides (NameD)
'If AmmeterICheck checkbox is true
If Me.AmmeterICheck.Value = True Then
`...hide the wire that was there,
mydocument.Shapes("AmmeterConnector").Visible
= False
'...show the ammeter cage and its display,
mydocument.Shapes("Ammeter").Visible = True
Me.AmmeterDisplay.Visible = True
'...show ammeter arrow on equiv. schematic,
mydocument.Shapes("IEquiv arrow").Visible =
True
'...show current value on equiv. schematic,
Me.IEquiv.Visible = True
'If CircVis parameter is true
If CircVis = True Then
'...show current arrow on left schematic
mydocument.Shapes("I arrow").Visible = True
End If
Else 'If AmmeterICheck checkbox is false
'...hide both the Ammeter and its display
mydocument.Shapes("Ammeter").Visible = False
Me.AmmeterDisplay.Visible = False
'...show the wire instead
mydocument.Shapes("AmmeterConnector").Visible
= True
'...and hide Ammeter objects on schematics
mydocument.Shapes("IEquiv arrow").Visible =
False
mydocument.Shapes("I arrow").Visible = False
Me.IEquiv.Visible = False
End If
End Sub
```

# c) Subrutina ejecutada cuando se inserta el valor de $V_z$ (voltaje Zéner) en la correspondiente caja de texto:

Private Sub Vz value KeyDown (ByVal KeyCode As MSForms.ReturnInteger, ByVal Shift As Integer) Call InitializeD 'Set the initial values 'Select the corresponding slide from the active presentation Set mydocument = ActivePresentation.Slides(NameD) 'String to be shown in case a wrong value is entered Str = "Please, enter a number" & vbCrLf & "between 1.0 and 3.5 (no units)" 'If "return" key is pushed... If KeyCode = vbKeyReturn Then '... if the entered value is not a number If Not IsNumeric(Me.Vz value.Value) Then '... inform the user through a message box MsgBox Str, vbInformation, "Range for Vz" '... set Vz to 1 Me.Vz\_value.Value = Format(1, "1.00") '... and set Vz to -1 in the table of computed values used for the chart mydocument.Shapes("Figure").Chart.ChartData.W orkbook.Sheets(NameDS).Range("E2").Value=-1# Else '... if the entered value is a number '...store its absolute value in Vz variable Vz = Abs(Me.Vz value.Value) '...and if such a value is out of range If Not (Vz >= 1# And Vz <= 3.5) Then '... inform the user through a message box MsgBox Str, vbInformation, "Range for Vz" '... and if the value is less than 1 If Vz < 1# Then '... set it to 1 in the textbox Me.Vz value.Value = Format(1, "0.00") '... and set it to -1 in the table of computed values used for the chart mydocument.Shapes("Figure").Chart.ChartData.W orkbook.Sheets(NameDS).Range("E2").Value=-1# Else '... or if the value is greater than 3.5'... set it to 3.5 in the textbox Me.Vz\_value.Value = Format(3.5, "0.00") '... and set it to 3.5 in the table of computed values used for the chart mydocument.Shapes("Figure").Chart.ChartData.W orkbook.Sheets(NameDS).Range("E2").Value=-3.5 End If 'For setting Vz when out of range Else 'If entered value is a number in range '... show it in text box with two decimals Me.Vz\_value.Value = Format(Vz, "0.00") '... and set it in the table of computed values used for the chart mydocument.Shapes("Figure").Chart.ChartData.W orkbook.Sheets(NameDS).Range("E2").Value= -Vz End If 'For Vz not in range End If 'For Vz "Not IsNumeric" End If 'For "return" key pushed 'Finally, execute voltage slide change event for upddating all the other values in the corresponding slide of the presentation Call V scrollbar Change 'End the subroutine End Sub