

# MyOpenLab

Versión 3.010

Guía de Usuario

<http://myopenlab.de>



**Prof. José Manuel Ruiz Gutiérrez**

# Índice

1. Instalación de MyOpenLab
2. Descripción básica del entorno
3. Panel Circuito
4. Librerías de Elementos: Panel de Circuito
5. Panel Visualización
6. Librerías de Elementos: Panel de Visualización
7. Ayuda
8. Organización de los ficheros de aplicaciones:
9. Creación de submodelos VM (Sub-VM) para incluir dentro de una aplicación genérica VM.
10. Creación de nuevos elementos para las librerías de MyOpenLab.
11. Protección y personalización de nuestros trabajos
12. Trazado y seguimiento de la ejecución de una simulación VM.
13. Otras Opciones e Informaciones

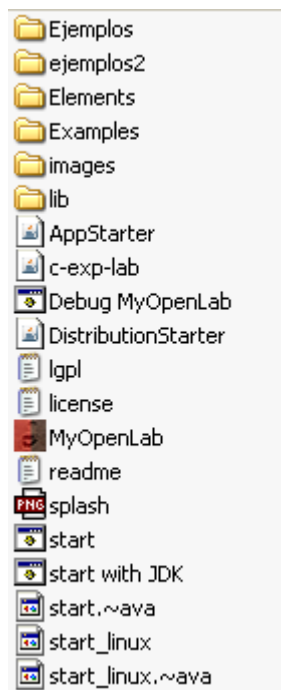
# 1. Instalación de MyOpenLab

Para disponer de la aplicación en su última versión se debe buscar en la dirección de Internet <http://myopenlab.de> (pagina original en Alemán) y <http://es.myopenlab.de> (página en Español) en donde se descargará el fichero de instalación correspondiente.

El fichero de la versión 3.010 no contiene un instalador por lo que bastará descompactarlo extrayendo los ficheros que contiene en una carpeta que podría llamarse

**../myopenlab.**

Una vez descompactado el fichero que hemos descargado de la pagina web correspondiente el aspecto de la estructura de ficheros que se crea es el que se muestra en la figura.



Se pueden distinguir las carpetas en donde están los proyectos (que contienen las aplicaciones VM de ejemplo) y la carpeta “Elements” que contiene las librerías tanto del Panel de Circuito como del Panel Frontal.

Los ficheros bat son los que lanzan la aplicación, teniendo en cuenta que “start with JDK.bat” es el que inicia la aplicación permitiendo la posibilidad de realizar compilaciones de código java.

Cuando arrancamos la aplicación por primera vez nos pedirá que seleccionemos el idioma y también el lugar (carpeta) en donde queremos que se guarden los nuevos elementos de librería que podamos crearnos.

No debemos olvidar que en cualquier caso la aplicación MyOpenLab requiere tener instalado al menos el runtime de java “**jre-6u1-windows-i586-p**” o la aplicación completa “**jdk-6u1-windows-i586-p**” si es que queremos realizar compilaciones de código.

Figura 1

La siguiente figura muestra la ventana de información de la aplicación en donde podemos encontrar importante información referida a las herramientas que se usan, los créditos y las condiciones de uso.

La aplicación también podrá trabajar en el sistema operativo LINUX, si bien en este manual se hablará de la aplicación funcionando bajo Windows XP.



Fig.2

## 2. Descripción básica del entorno

MyOpenLab es un entorno orientado a la simulación de circuitos y sistemas con un amplio campo de aplicaciones.

Sus características más importantes son:

- Facilidad de uso
- Amplia biblioteca de funciones tanto para manejo de señales analógicas como digitales.
- Tratamiento de los tipos de datos y operaciones con estos.
- Realización de las aplicaciones mediante el uso de bloques de función.
- Facilidad para crear pantallas de visualización que recojan el estado de las variables y eventos de las simulaciones.
- Posibilidad de ampliación de su librería de componentes.
- Posibilidad de creación de submodelos que se pueden encapsular a su vez en otros submodelos.

Dentro de las herramientas de Instrumentación Virtual (VI Virtual Instruments) esta herramienta es un ejemplo de sencillez y potencia a la vez.

Las aplicaciones que se realizan con MyOpenlab reciben el nombre de **VM** (Visual Modeling).

Toda aplicación constará de dos partes bien diferenciadas:

### **Circuito (Panel Circuito)**

### **Panel de Visualización (Panel Frontal)**

**El Circuito** será el conjunto de funciones y/o operadores que convenientemente enlazados responden a una funcionalidad relacionada con un circuito electrónico o un sistema del tipo que sea. Este conjunto de funciones se edita en el correspondiente panel de edición de circuito, que aparece en la pestaña etiquetada “**Panel Circuito**”.

**La parte de Panel Frontal de Visualización** será el que aparece en la pestaña etiquetada con “**Panel Frontal**” en la que se situaran los objetos de visualización grafica que asociados a las variables del circuito permitirán la visualización de la evolución de la simulación.

En la figura 3 vemos el aspecto de la pantalla una vez que se inicia la aplicación y se carga un ejemplo. En ella podemos distinguir fácilmente las siguientes partes:

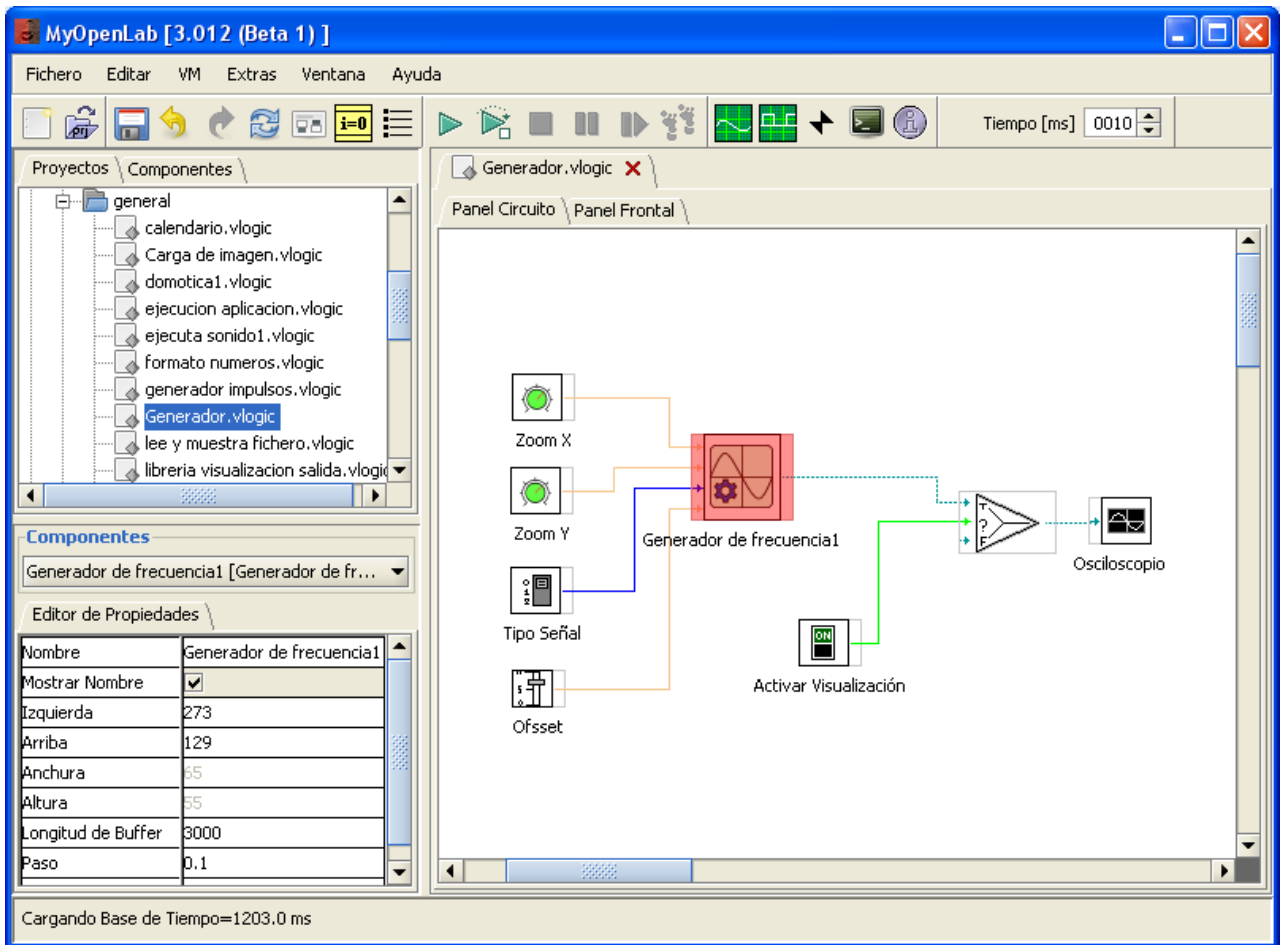


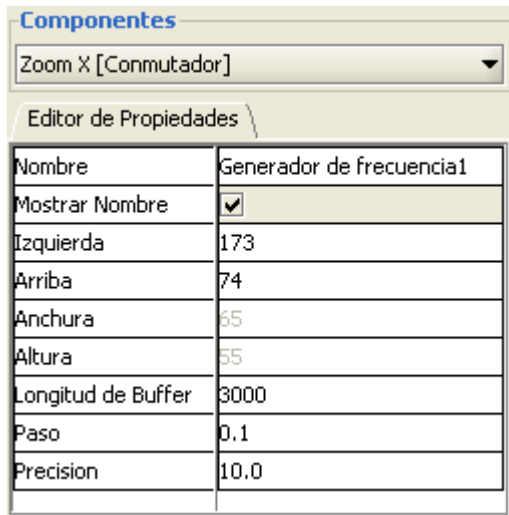
Figura 3

Las partes más importantes en las que se divide la pantalla principal de MyopenLab son:



Figura 4

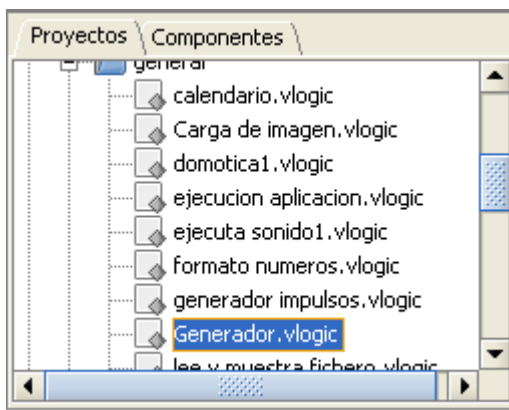
**Área de Componentes:** En donde aparecen las librerías del panel activo (Panel Circuito o Panel Frontal). Es posible navegar por las librerías pulsando sobre los iconos que representan las carpetas en donde están los distintos elementos. Fig.4



### Área de Propiedades de Componentes:

En esta aparece una ventana con las propiedades del objeto o función que señalemos. Fig.5

Figura 5



Área de Proyectos. En donde figura el árbol de carpetas y ficheros de aplicaciones VM. Fig. 6

Fig. 6

**Panel Circuito:** Es aquel en el que diseñamos el circuito y realizamos el cableado de todos los componentes uniendo entradas y salidas. Fig. 7

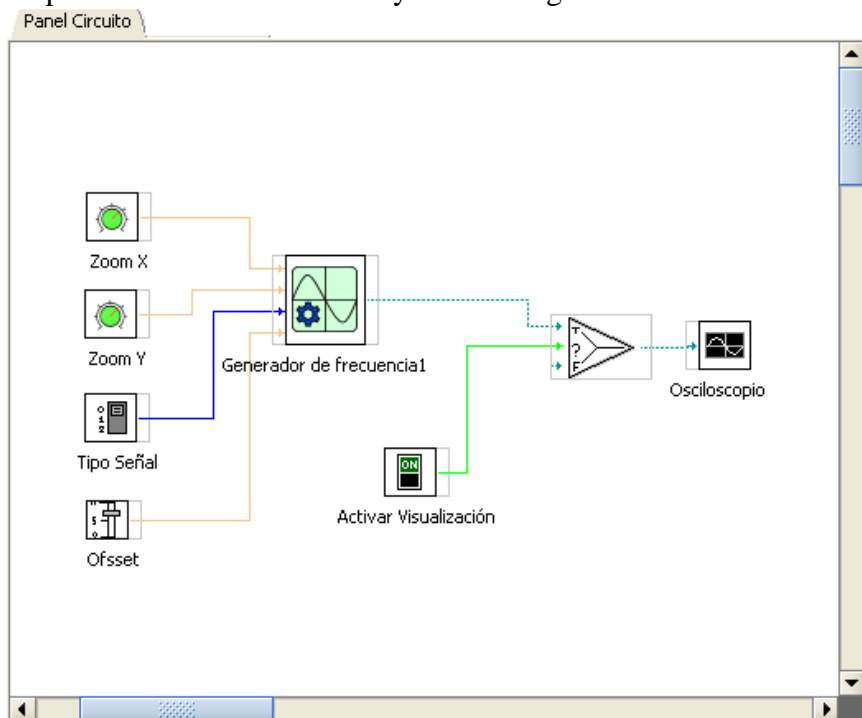


Figura 7

**Panel Frontal:** Es aquel sobre el que se depositan los objetos que van a mostrar los resultados o de los que se recogerán valores en la simulación. Fig.8

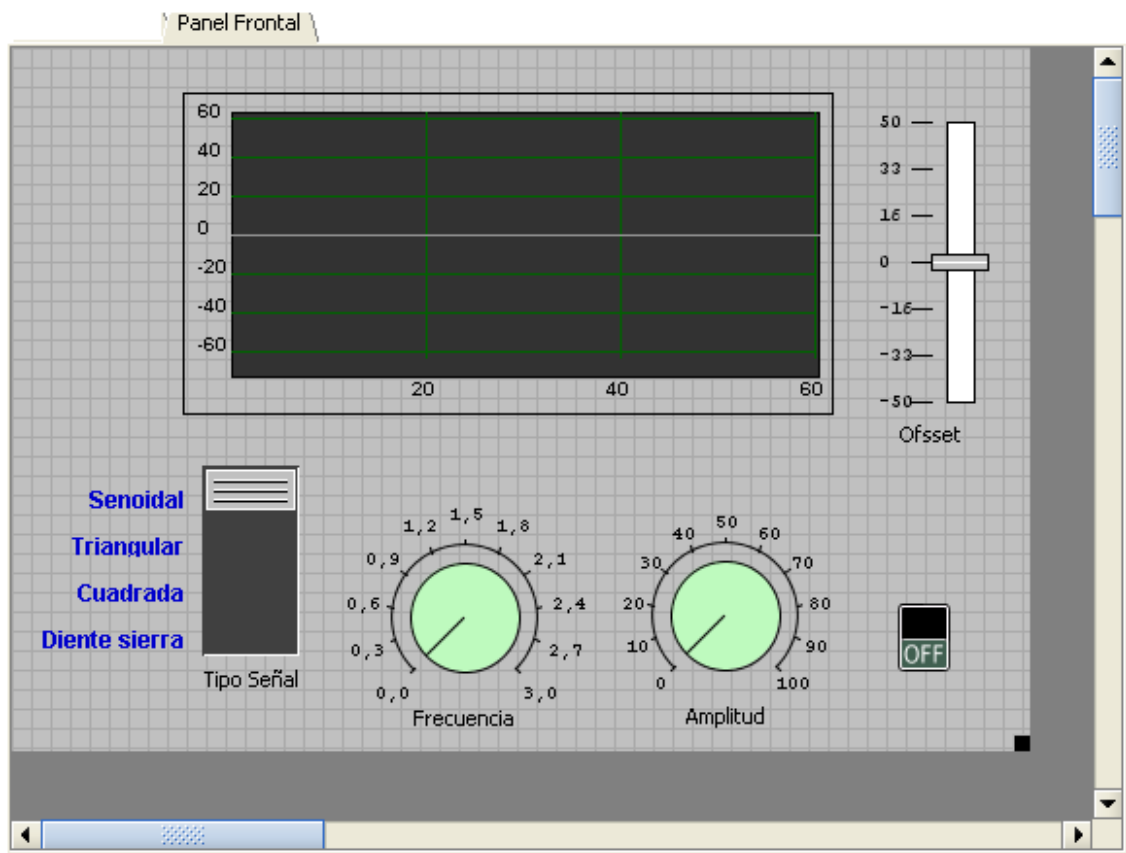


Figura 8

Vemos que algunos elementos se muestran en los dos paneles. Exactamente aquellos que hemos tomado de las librerías de objetos de visualización. Tal es el caso de los potenciómetros “Zoom X” y “Zoom Y” que aparecen en el panel de circuito y en el panel frontal como los controles de “Frecuencia” y “Amplitud”.

**Barra de Botones:** Esta barra contiene los botones asociados a las funciones más importantes de MyOpenLab. Figura 9



Figura 9

Vemos que de izquierda a derecha los botones sirven para:



Crear nuevo Proyecto: Nuevo Proyecto



Abrir un Proyecto ya creado: Abrir Proyecto



Guardar el fichero VM actualmente en edición: Grabar VM



Deshacer la última acción: Deshacer



Restablecer la acción deshecha: Restablecer



Actualiza los datos del VM activo: Actualización de VM



Permite configurar Opciones: Opciones



Permite definir y/o editar variables: Definir Variables



Mostrar ventana con información sobre los colores de las líneas de datos: Leyenda colores tipos de datos



Iniciar la Simulación: Arranca VM



Iniciar la simulación en modo depuración: Arranca VM (Modo debug)



Terminar la simulación: Terminar VM



Detener la simulación: Detener VM



Continuar la simulación: Continuar VM



Realizar la simulación paso a paso: Paso/paso VM



Muestra Ventana grafica numérica



Muestra Ventana Grafica Digital



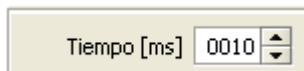
Muestra ventana Tabla de Datos



Muestra Consola de salida de datos



Muestra ventana con información del programa: Información



Ajusta tiempo de simulación

### 3. Panel Circuito

El Panel Circuito es el área de trabajo en donde se van colocando cada uno de los elementos de las distintas librerías que vamos a utilizar en nuestro proyecto. Figura10

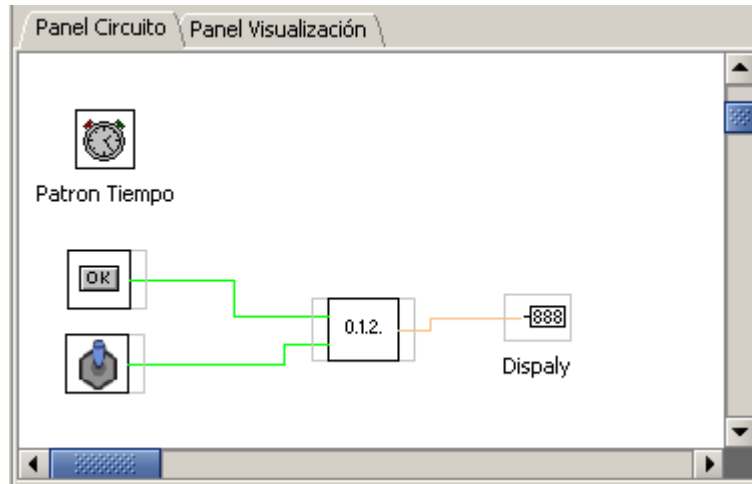


Figura 10

Los componentes una vez situados sobre este área están en disposición de ser unidos entre si utilizando el ratón. Basta acerca el puntero del ratón a un terminal de entrada o de salida de un componente para que aparezca una etiqueta que identifica el nombre del terminal y entre paréntesis el tipo de datos que admite (si es una entrada) o que entrega (si es una salida). Figura 11

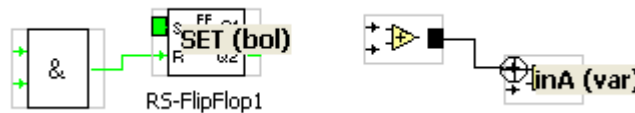


Fig. 11

Además podemos habilitar la llamada ventana de componente desde el menú desplegable “Ventana”

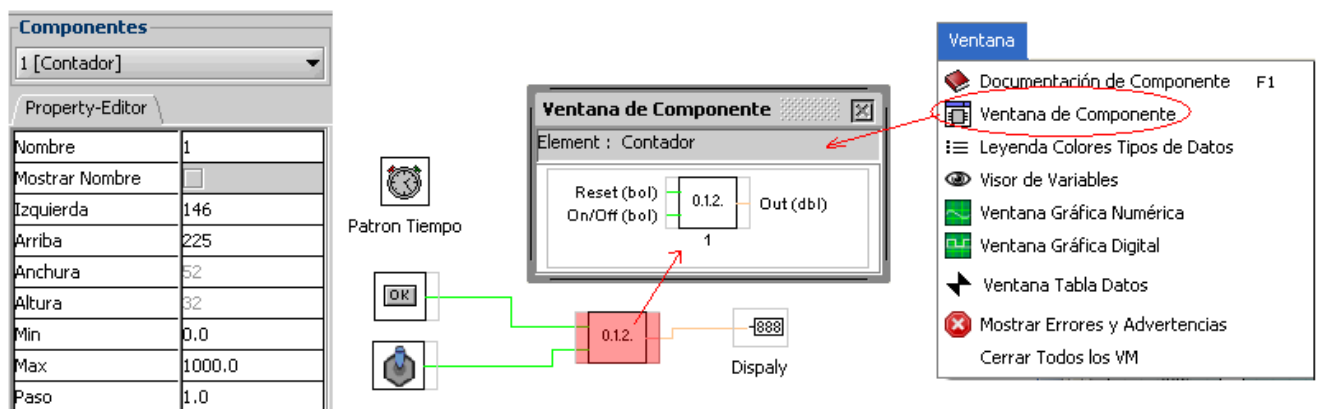


Figura 12

## Selección de un componente

Los elementos se seleccionan del área de librerías situada en la parte superior izquierda de la pantalla. A las librerías se accede pulsando sobre las carpetas y una vez localizado el elemento que necesitamos pulsamos con el botón izquierdo y lo seleccionamos, después vamos al área de trabajo del panel y pulsamos otra vez con el botón izquierdo quedando colocado el elemento. Podremos desplazarlo si lo deseamos simplemente manteniendo el botón izquierdo pulsado sobre el y desplazando el ratón.

En la figura 12 vemos el componente Contador junto a su ventana de componente y a su ventana de propiedades.

Obsérvese que este componente solo admite en sus entradas datos de tipo *booleano* y el valor que devuelve es *numérico dbl*.

Es muy importante que nos fijemos en los colores de las líneas de entrada y de salida de los componentes ya que en función de su color y tipo (continuo o punteado) se puede averiguar su naturaleza (ver figura 13).

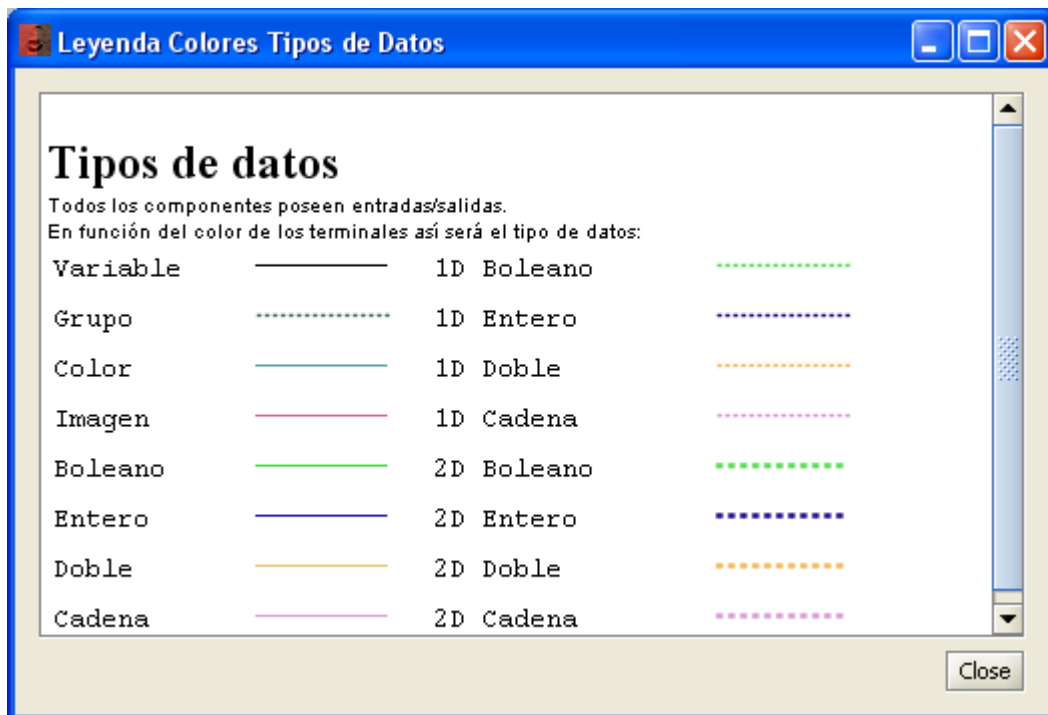


Fig. 13

Algunos componentes al ser colocados en el panel de Visualización aparecerán de manera automática en este panel. Por ejemplo el pulsador OK, el interruptor y el display. En el Panel de Visualización no se puede realizar el conexionado de componentes es por ello por lo que los componentes de Entra/Salida de este panel deben aparecer en el Panel de Circuito para en este ser conectados a la entra o salida correspondiente.

## Supresión de una conexión

Si queremos borrar una conexión marcamos con el ratón sobre ella, y después con la tecla *SUPR* o en el menú *Editar->Cortar* se elimina la conexión. Figura 14

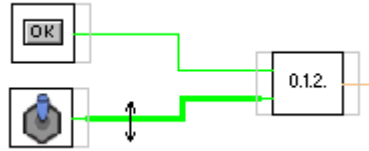


Figura 14

## Colocación de un nudo de derivación o empalme

Si queremos colocar un nudo para poder derivar una conexión a dos componentes bastará que se coloque el ratón sobre la línea de conexión y pulsado el botón derecho aparecerá un menú del que seleccionaremos agregar nudo. Figura 15

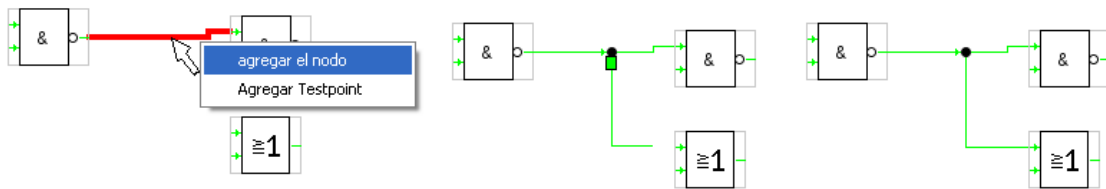


Fig. 15

## Conexión de componentes

Para realizar la conexión entre componentes bastará con marcar la salida con un clic del botón izquierdo y después marcar en la entrada correspondiente del componente seleccionado. Figura 16

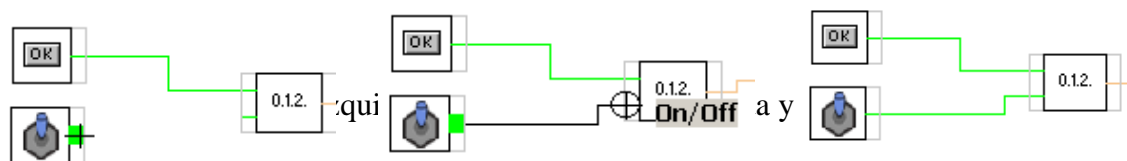


Figura 16

Siempre tenemos que conectar una salida a una entrada o la salida de un nudo a una entrada.

La etiqueta que aparece On/Off es el indicativo del Terminal de entrada.

## Para Borrar componentes

Para borrar uno o varios componentes o un área completa bastará con seleccionar el área o componentes con el ratón y después una vez que el área se muestra en video resaltado se pulsa la tecla *SUPR* del teclado o se selecciona la opción “Cortar” del menú *Editar*.

En la edición se podrán marcar digititos componentes y copiarlos y pegarlos, de esa manera podemos realizar la edición más rápidamente. Figura 17

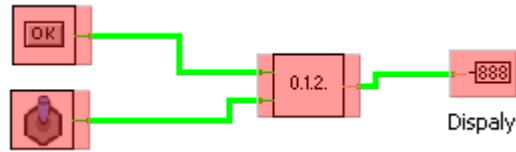


Figura 17

### Copiar y pegar Componentes o áreas completas.

Si queremos duplicar un área o componente basta con seleccionar el área o componente y seleccionar el menú Editar la opción "Copiar " y después la opción "Pegar". No se debe olvidar que al pegar se pega el arrea o componente copiado sobre el original.

La operación de copiar se puede hacer con las teclas CTRL+C y la operación de pegar con CTRL+V.

### Colocación de un Elemento SubVM en el area de trabajo del Panel de Circuito.

Cuando queramos colocar un componente SubVM en el área de trabajo bastara con que lo arrastremos desde el área de proyectos (seleccionando su nombre) y lo arrastremos a su lugar. Esto se podrá hacer siempre que el subelemento este en la misma carpeta de proyecto que el VM que estemos editando.

### Ventana de componentes.

Mediante la correspondiente opción del menú podemos hacer visible la estructura del componente que seleccionemos. *Menú Ventana -> Ventana de componentes*. En esta ventana se muestran las entradas y las salidas indicando en cada una su nombre y entre paréntesis el tipo de dato. Figura 18

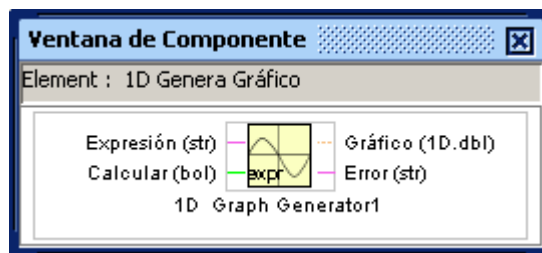


Figura 18

### Menú Contextual sobre el componente

Estando sobre un componente pulsando el botón de derecho del ratón aparece este menú. Figura 19

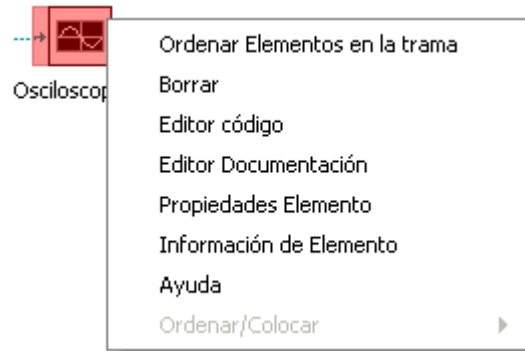
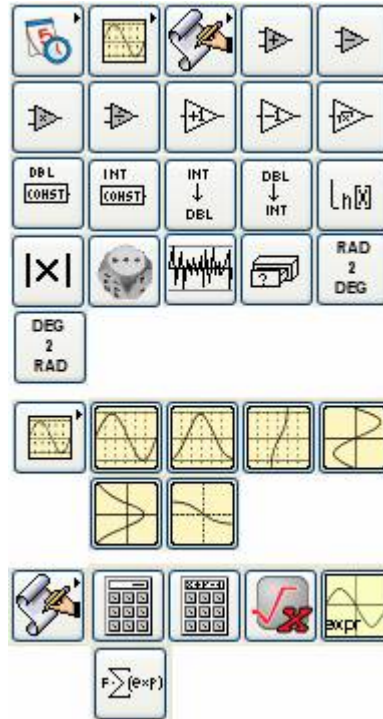


Fig. 19

Desde este menú podemos invocar el fichero de ayuda del componente y también editarlo. De la misma manera, desde la opción “*Editor Código*” podemos editar el código Java del componente y modificarlo si es nuestro deseo. Esta posibilidad hace de MyOpenLab una herramienta absolutamente flexible y abierta



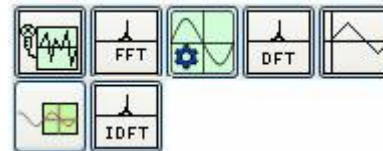
### Operadores Numéricos



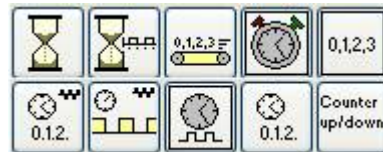
### Tratamiento de Caracteres



### Elementos Analógicos



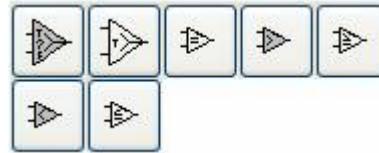
### Utilidades



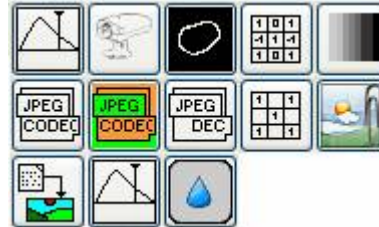
### Ficheros de Entrada/Salida



### Comparaciones



### Tratamiento de Imágenes



### Tratamiento de Sonidos



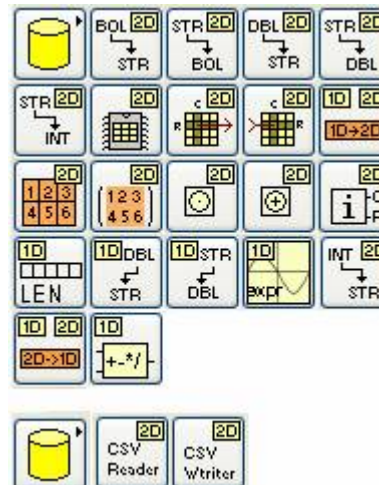
### Color



### Pines de E/S



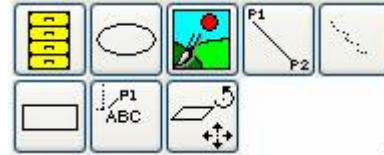
### Vectores y matrices



Agrupación de Elementos



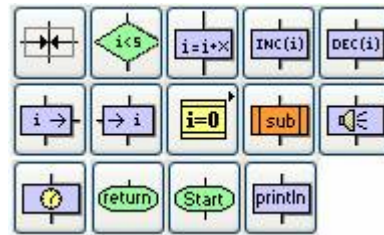
Objetos Gráficos "canvas"



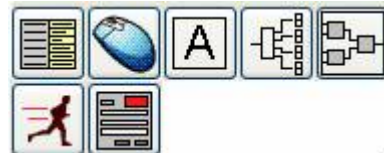
Librería de Física



Librería de Diagramas de Flujo



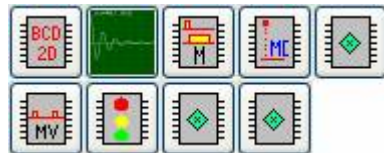
Librería de Extras



Librería de Conexiones entre aplicaciones



Librería definida por el Usuario



## Librería de Automatización



## Interfaces



## 5. Panel Frontal

Este panel es en el que se colocan los elementos que van a servir para introducir los valores y/o parámetros de entrada (estímulos) de la aplicación así como los elementos que permitan ver los datos de salida (elementos de salida). También nos permite colocar sobre él elementos decorativos que no tienen ninguna relación con los elementos que componen la aplicación. En la figura vemos el panel correspondiente al ejemplo que hemos utilizado en el Panel de Circuito. Figura 20

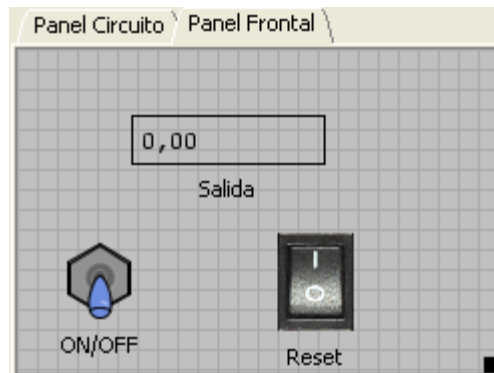


Figura 20

En la figura 21 se muestra el componente *salida numérica* con la ventana de componente y el menú de propiedades abiertos. No olvidemos que las propiedades que muestra un componente en este panel son distintas a las que muestra si estamos en el Panel de Circuito.

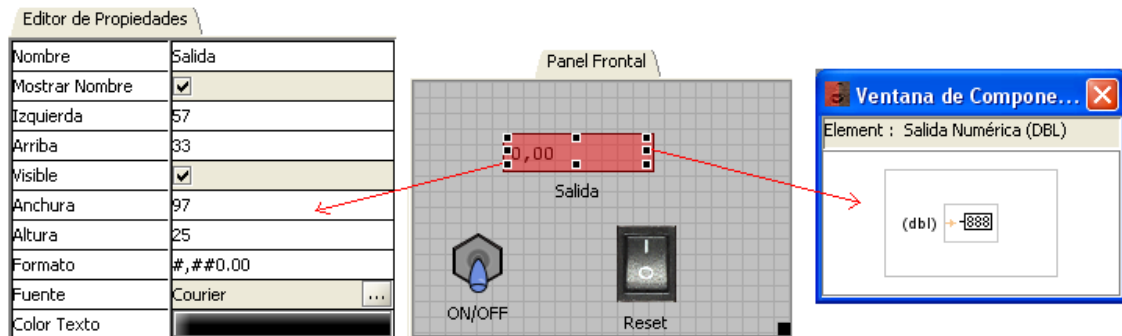


Figura 21

En este panel se van colocando los componentes buscando una forma armónica de visualización dado que esta será la imagen que nos dará MyOpenLab una vez que pasemos del modo edición al modo de simulación. En la figura 22 vemos el aspecto del




panel una vez que le hemos dado a simulación 



Figura 22

No olvidemos que para detener la simulación bastará con pulsar en  o en  de la barra de botones del menú

En el caso de que queramos que la ventana aparezca con un título bastará que lo escribamos en el lugar indicado en la figura 23

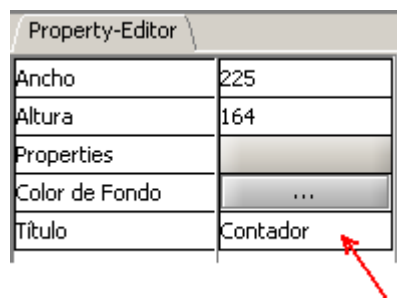


Figura 23

En cuyo caso el panel, al ejecutarse la aplicación muestra el nombre. Figura 24



Figura 24

Los objetos que se muestran en el Panel de Visualización pueden ser modificados de tamaño tal como se muestra en la figura 25.

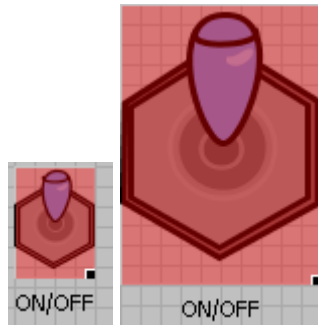


Figura 25

Para ello lo único que hacemos es estirar desde el punto negro marcado en las esquinas del componente cuando esta señalado

Los objetos de decoración son útiles para realizar el panel dado que permiten aproximarnos a la realidad grafica de un instrumento, sistema, maquina, etc...Figura 26

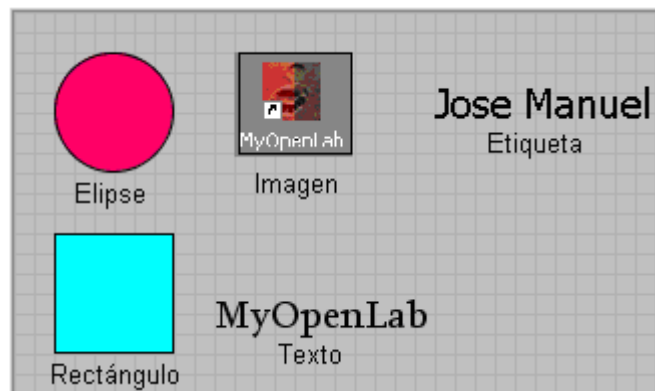


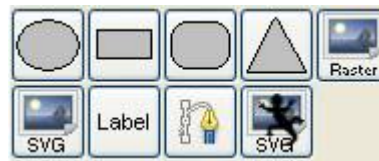
Figura 26

## 6. Librerías de Elementos: Panel de Visualización.

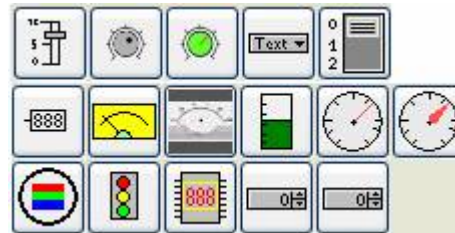


### Librería de elementos de Panel Frontal

#### Elementos de Decoración



#### Elementos de visualización numérica



#### Elementos de activación digital



#### Elementos de Entrada y salida de cadenas de caracteres



#### Elementos de entrada y salida tipo vectores y matrices de datos



Elementos de visualización grafica en ejes coordenados I



Elementos de visualización grafica en ejes coordenados II



Elementos extras



Elementos de Automatización



Elementos de librería de usuario




Robot 2D

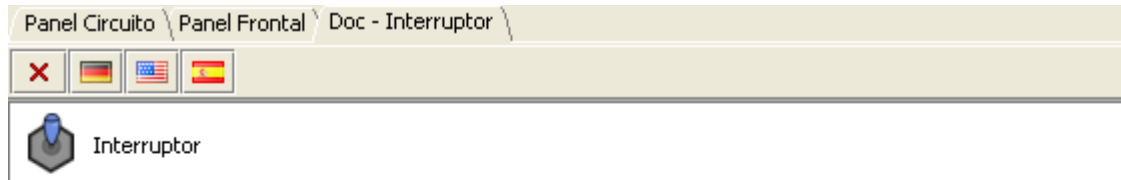


Figura 27

## 7. Ayuda.

Bastará marcar el componente y pulsando F1 se obtiene una ventana de ayuda. También se puede hacer con una opción del menú contextual que acabamos de comentar.

La ayuda se puede ver en cualquiera de los tres idiomas (fig. 29) en los que esta programado MyOpenLab. Para cerrar la venta bastará con pulsar sobre el icono: 



### **Objetos de Entrada (Botones/Interruptores): INTERRUPTOR**

Este elemento actua generando una variable Booleana en funcion de si esta en una posición u otra.

#### **Atributos modificables:**

Init-Valor Permite que al iniciar la simulacion la variable asociada al interruptor tenga el valor 1 o 0



Figura 29

El fichero de ayuda esta escrito en lenguaje HTML y se podrá editar en el momento que queramos bastará con situarse sobre le componente y pulsado la tecla derecha del rato seleccionamos “Editor Documentación”. Figura 30

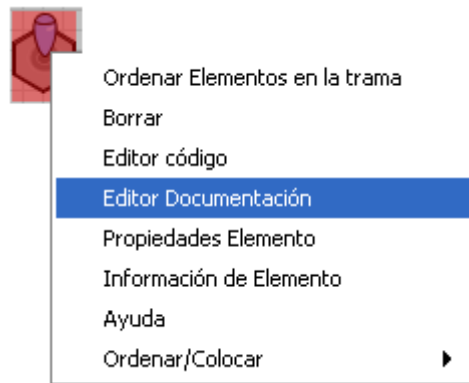


Figura 30

En la figura 31 vemos el aspecto de la ventana de edición del fichero de ayuda del componente interruptor.

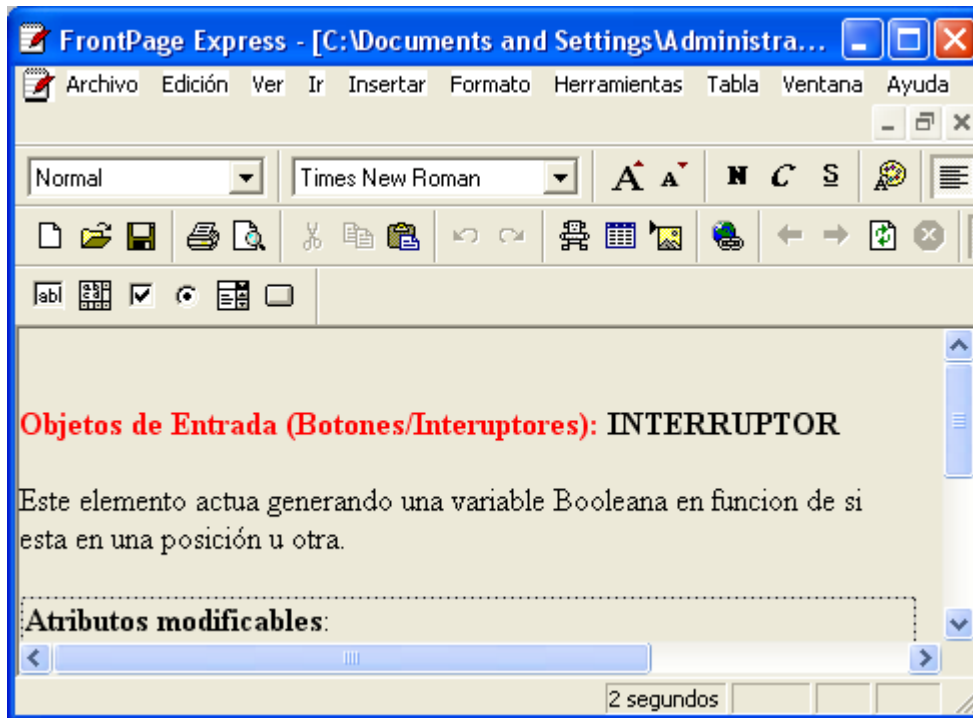


Fig.31

## 8. Organización de los ficheros de aplicaciones:

En MyOpenLab la organización de los ficheros de aplicaciones, también llamados ficheros VM (Visual Models), se realiza mediante estructuras de ficheros llamadas **“Proyectos”**.

Los proyectos físicamente constituyen carpetas dentro de las cuales están las aplicaciones VM.

Al crear un nuevo proyecto (pulsando el botón derecho estando sobre el icono “Proyectos”) aparece un menú mediante el cual se selecciona “Nuevo proyecto” apareciendo una ventana en la que se nos pide el nombre del nuevo proyecto, escribimos el nombre del proyecto y si lo deseamos cambiamos el nombre del VM principal del proyecto que se llama por defecto “Main.vlogic”.

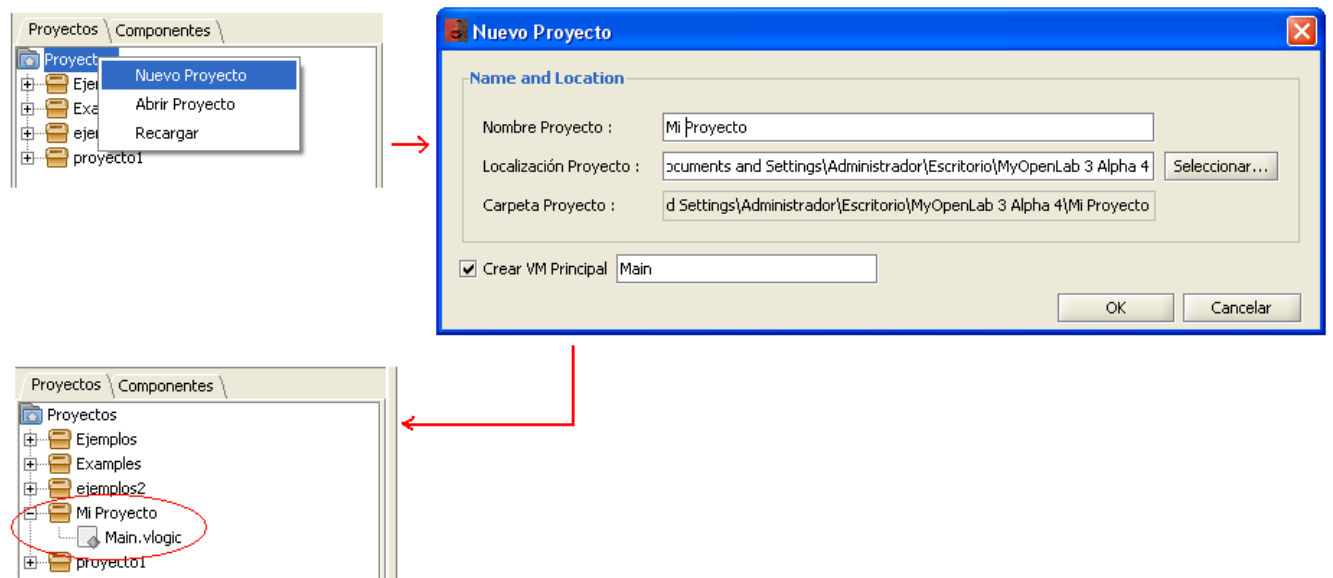


Figura 32

Este primer VM creado se presenta en blanco (sin ningún elemento de ni en el Panel Circuito ni el Panel Frontal) y en el podemos diseñar nuestra primera aplicación.

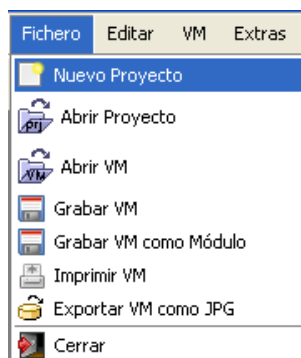


Figura 33

carpeta de proyecto nos permiten básicamente la manipulación de los archivos de aplicaciones VM que tengamos allí.

Para crear un nuevo proyecto también se puede hacer seleccionando la opción desde el menú desplegable de la aplicación (Nuevo Proyecto)

La edición de las carpetas de proyecto se puede realizar simplemente situando el ratón sobre la carpeta y haciendo uso del menú contextual (botón derecho de ratón)

Vemos que las operaciones que se pueden realizar con una

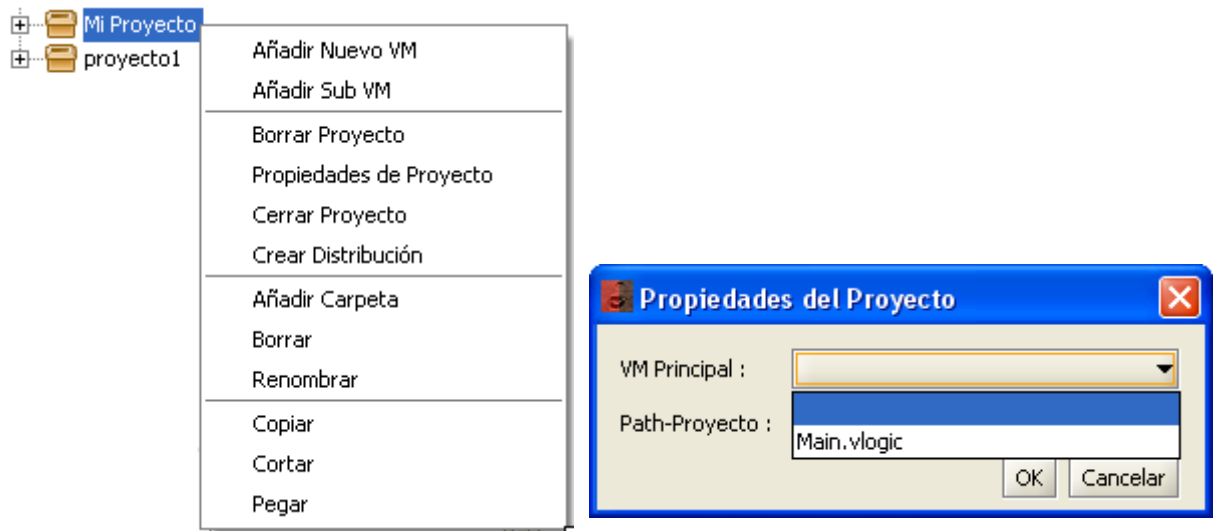


Figura 34

Es importante resaltar entre las operaciones que aparecen la de “Crear Distribución” y “Añadir Sub VM” de las que nos ocuparemos más adelante

Con la opción de “Propiedades de Proyecto” podemos seleccionar el VM principal de la aplicación, tal como se muestra en la figura.

Con las opciones de “Copiar”, “Cortar” y “Pegar” podemos cambiar de carpeta los distintos VM que tengamos creados.

### Ejemplo de creación de una “Estructura de Proyecto”.

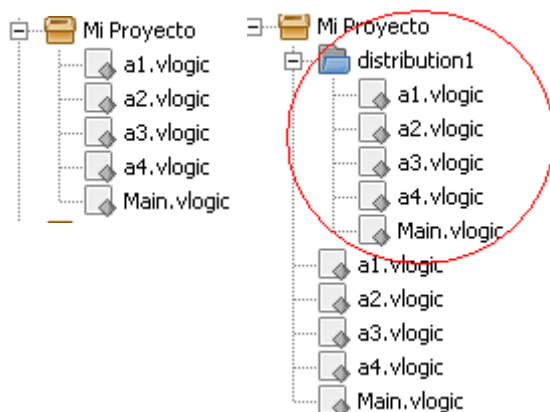


Figura 35

A continuación vemos un ejemplo de un proyecto que hemos creado en el que se contemplan 4 aplicaciones VM llamadas a1, a2, a3, a4.

En el VM Main.vlogic hemos colocado cuatro botones de acción que invocan mediante la función “Sub-VM Element” a cada una de ellas.

En la figura 37 vemos los paneles de circuito y frontal del VM Main apreciando cada una de las aplicaciones invocadas.

Una vez que se ha creado la configuración se puede crear una “Distribución” de la aplicación haciendo uso de la correspondiente opción del menú contextual de proyecto.

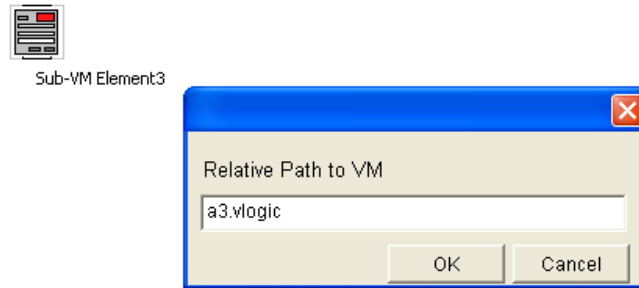


Figura 36

La figura 36 muestra la ventana de recogida del nombre del VM que queremos asociar al elemento “Sub-VM Element”

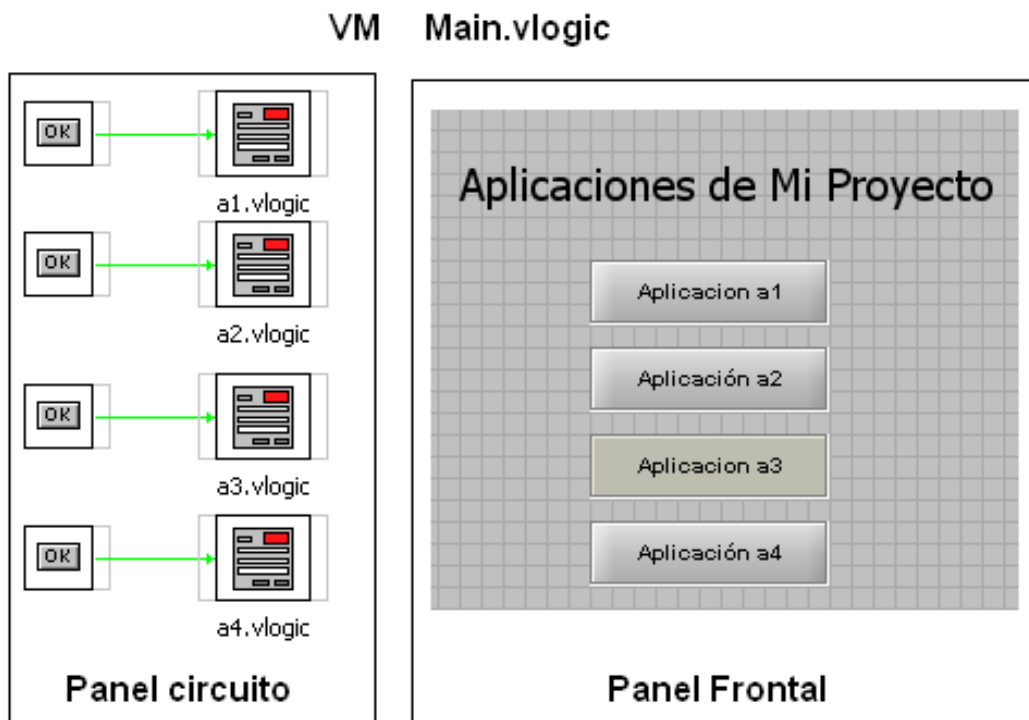


Figura 37

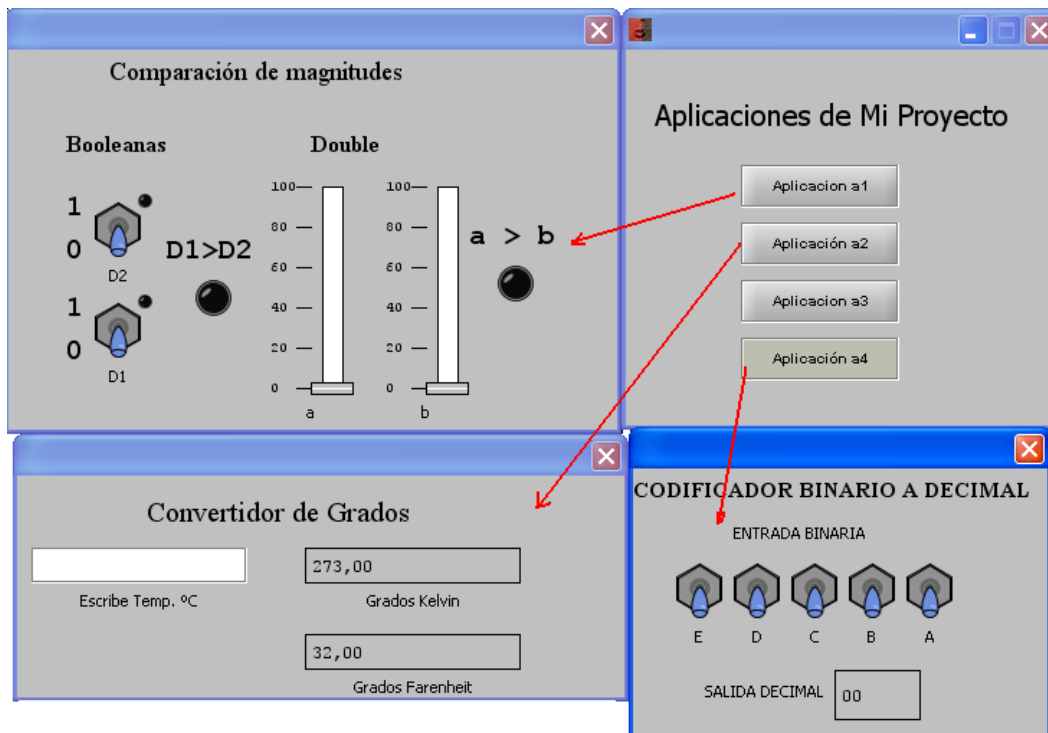


Figura 38

En la figura 37 vemos la definición del VM Main en la que se aprecian los cuatro botones que activan las aplicaciones.

En la figura 38 vemos en “modo ejecución” las pantallas desplegadas de las aplicaciones.

Este método de creación de proyectos permite de una manera bastante cómoda estructurar nuestras aplicaciones VM en temas y poder agruparlas todas en una misma pantalla desde la que se abran cada una de ellas.

## 9. Creación de submodelos VM (Sub-VM) para incluir dentro de una aplicación genérica VM.

MyOpenLab, en los casos en los que el modelo que se tenga que diseñar sea muy grande, permite crear submodelos que respondan a varias funciones del modelo y que se encapsulen en un único componente (Sub-VM).

Estos submodelos quedan incorporados en el árbol de carpetas de ejemplos y después podrán ser incorporados en otras nuevas simulaciones (VM).

Para explicar el procedimiento lo mejor será que realicemos un ejemplo.

Se trata de crear una simulación de un circuito digital que vamos a integrar en dos submodelos: SubDig1 y SubDig2

La creación de un submodelo se puede realizar de varias formas dependiendo del lugar al que recurramos dentro del entorno.

### **Método 1**

En nuestro ejemplo lo haremos creando directamente los submodelos desde el árbol de Proyectos. Pasos a seguir:

- 1.- Bastará con situarnos sobre el nombre de la carpeta de proyecto en la que vayamos a guardar la aplicación y pulsando el botón derecho seleccionemos “Crear Sub-VM” del menú contextual.

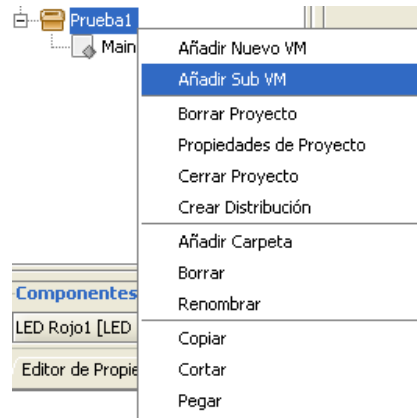


Figura 39

- 2.- A continuación seleccionamos el número de pines de entrada y salida que vaya a tener nuestro Sub-VM, a la izquierda (entradas) y la derecha (salidas).

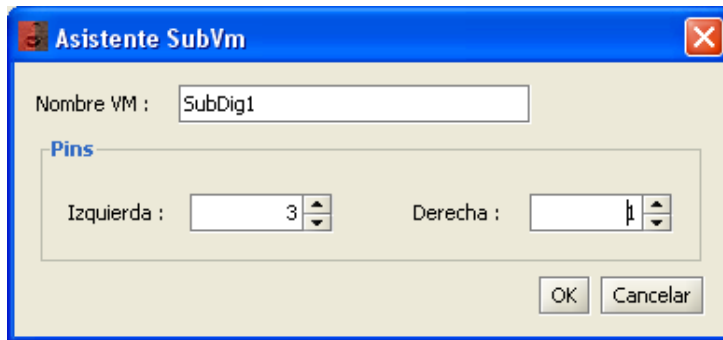


Figura 40

Al aceptar, una vez seleccionado el numero de Pins aparecerá en el “Panel de Circuito” la siguiente estructura.



Figura 41

- 3.- Lo que procede ahora es seleccionar los componentes que formaran parte del Sub-VM y designar la naturaleza de los Pins de entrada/salida. La hora de definir los pins también podremos ponerle a cada uno el nombre “etiqueta de terminal” que nos servirá para luego identificar cada entrada/salida.

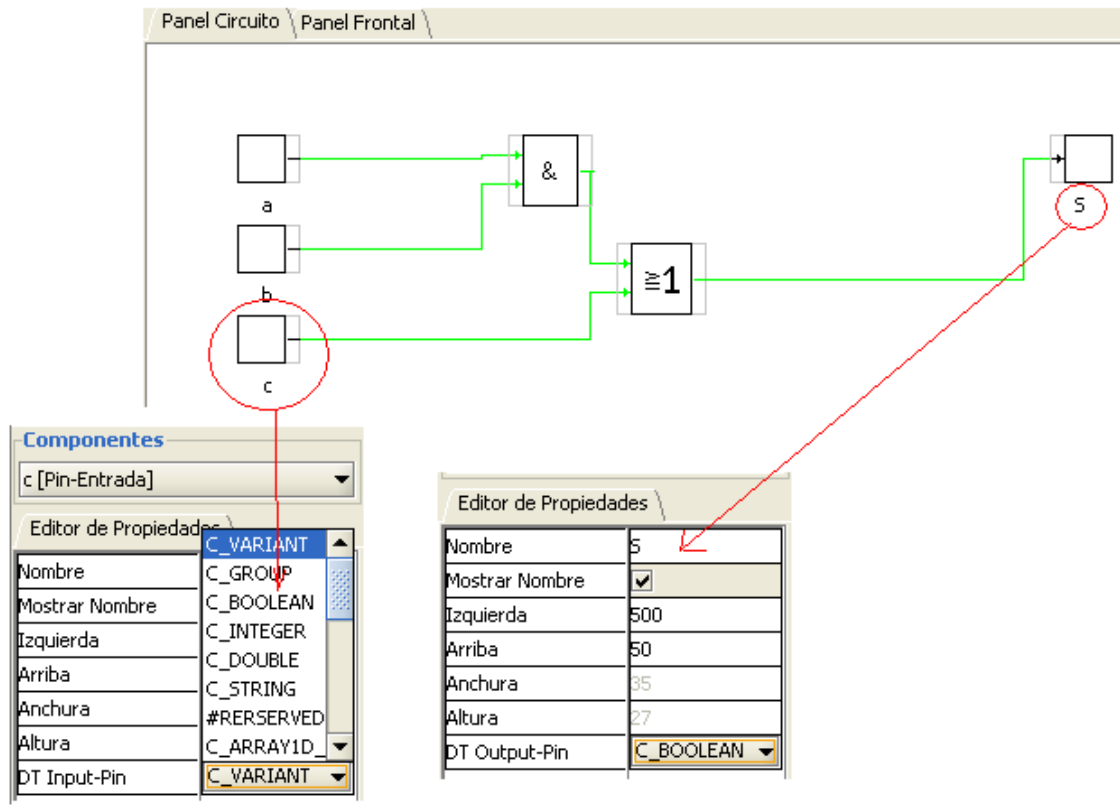


Figura 42

- 4.- Una vez creado el esquema completo del Sub-VM procedemos a guardarlo. Se pulsa sobre la cruz roja de la pestaña y pulsamos “Si” grabándose entonces el fichero tal como se muestra en la figura.

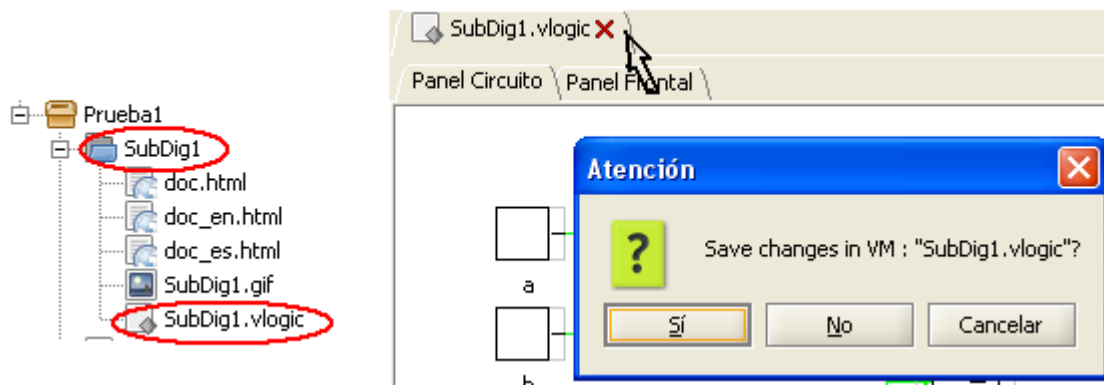


Figura 43

Vemos que el Sub-VM lo ha colocado en una carpeta que recibe el mismo nombre que el propio fichero (SubDig1) y además a creado tres ficheros .html para poder colocar en ellos la información de ayuda que queramos en los tres idiomas de MyOpenLab, así como un fichero (SubDig1.gif) que será el icono que podemos poner a este Sub-VM.

## Edición del fichero SubDig1.html

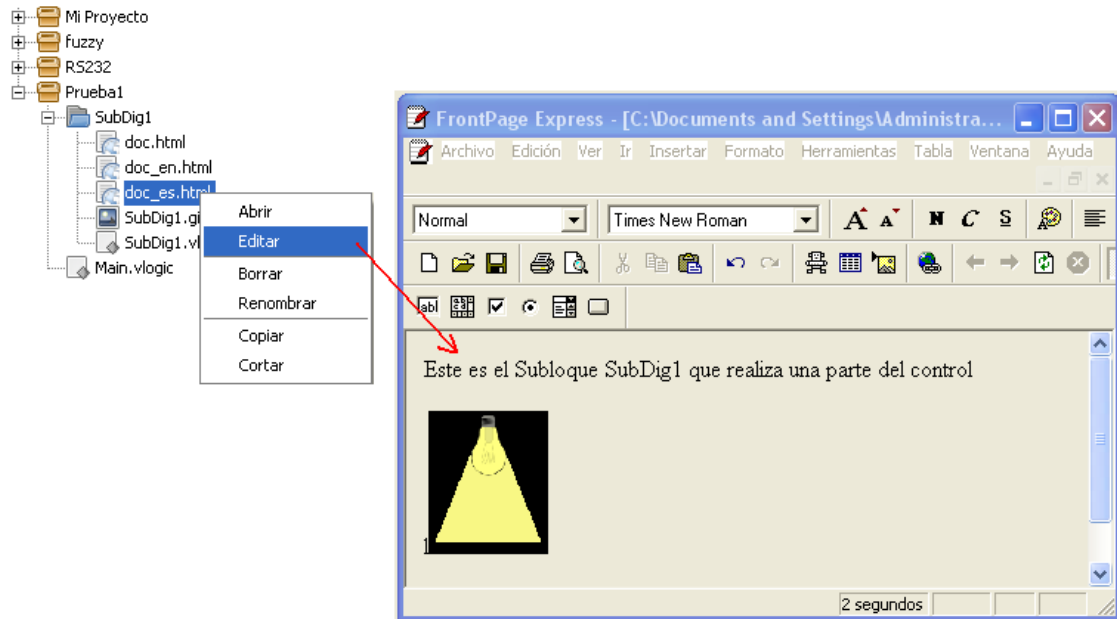


Figura 44

Igualmente se podrá editar el icono y cambiarlo por otro.

## Edición del icono asociado al Sub-VM SubDig1

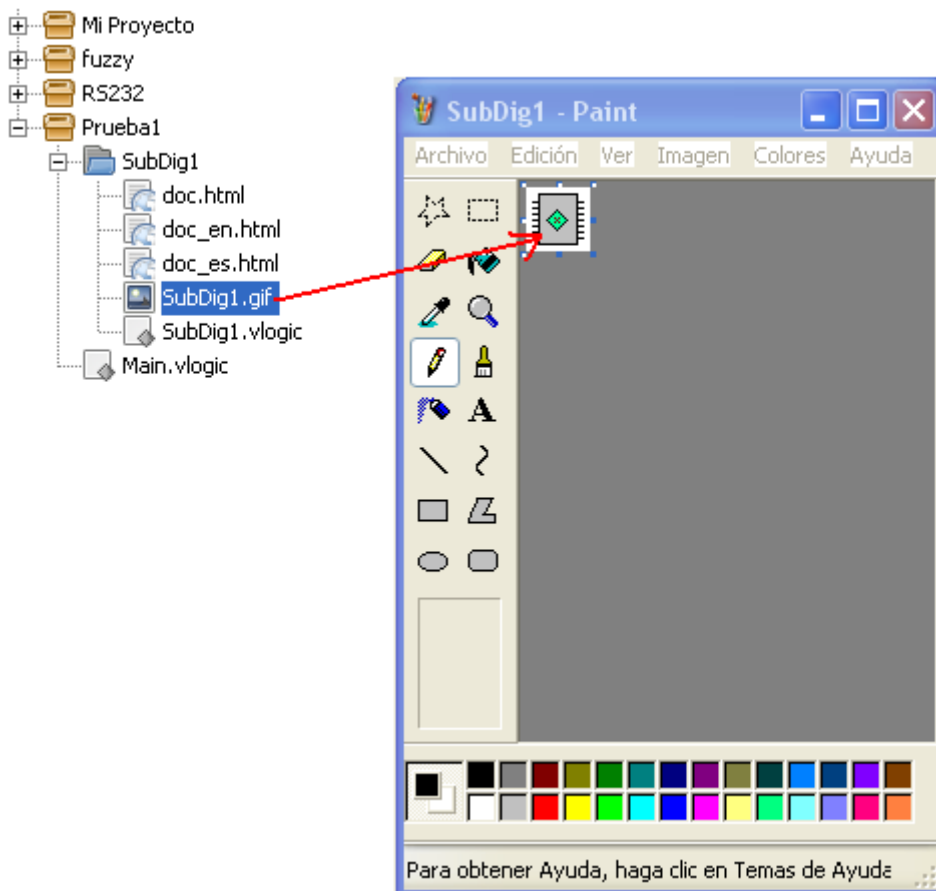


Figura 45

Finalmente vemos el aspecto de cómo quedaría el submodelo creado. Bastaría con crear una nuevo VM y arrastrando el fichero SubDig1.vlogic al área del “Panel de Circuito” quedaría allí simbolizado con su nuevo icono y como vemos sus pines quedarían perfectamente identificados en la “Ventana de Componente”, figura 46. Si deseamos ver el contenido del elemento SubDid1.vlogic bastara con situarnos sobre el botón derecho del ratón mostramos el menú contextual que aparece en la figura y seleccionamos “Show VM” abriéndose el esquema completo de este figura 47.

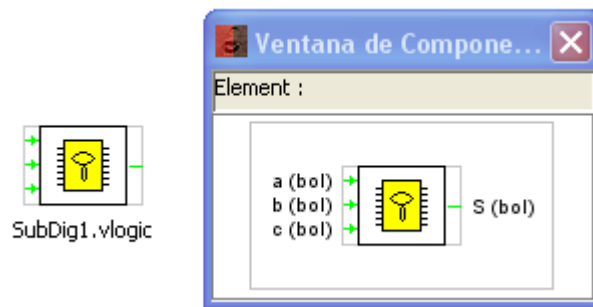


Figura 46

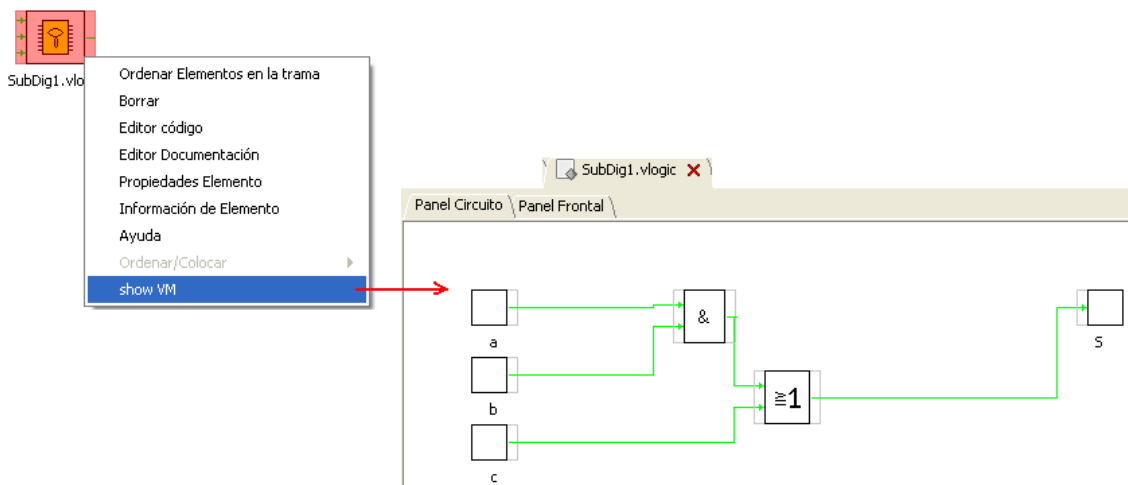


Figura 47

Al pulsar en “Show VM” vemos el contenido del SubDig1.vlogic

En el Sub-VM creado también pueden aparecer elementos de visualización propios del “Panel Frontal” como por ejemplo un Led.

- 5.- Una vez creados los Subcircuitos SubDig1 y SubDig2 para incorporarlos a un diseño bastará con crear un VM nuevo y arrastrar sobre el cada uno de los elementos.

En la figura 48 vemos como quedaría el VM Digital1.vlogic

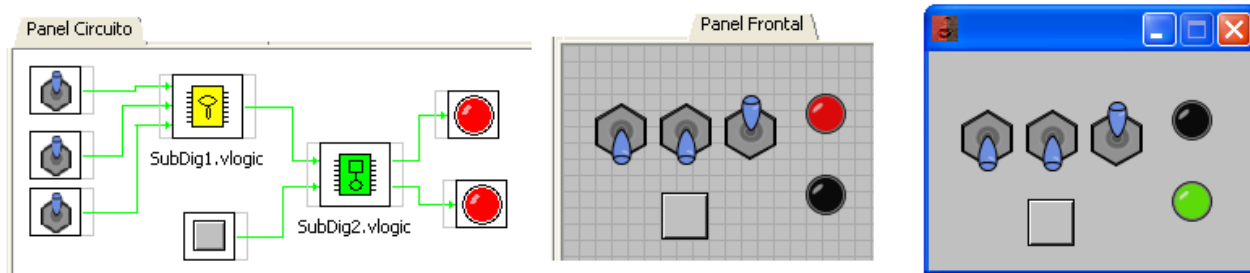


Figura 48

Aspecto de la aplicación VM Digital1.vlogic

En la siguiente figura vemos el aspecto de cómo quedaría el árbol de “Proyectos”.

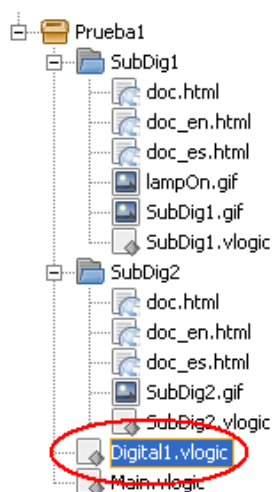


Figura 49

## Método 2

Otra forma de crear un estructura de submodelos Vm es recurriendo a la opción “Crear SubVM del menú Extras de MyOpenLab, Figura 50.

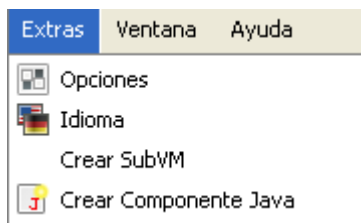


Figura 50

El procedimiento sería el siguiente.

- 1.- Creamos una nueva aplicación teniendo marcada la carpeta de proyecto en la que queramos depositarla, seleccionando

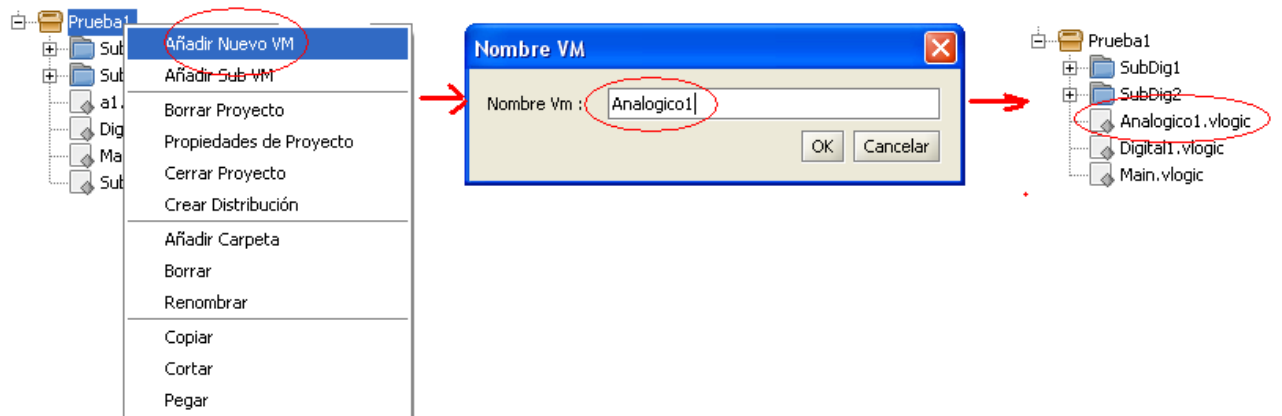


Figura 51

- 2.- Sobre el “Panel de circuito” colocamos todos los elementos que necesitemos y realizamos el esquema como lo que después será el SubA1.

La selección del área que hagamos será la que acoja a los componentes que vamos a integrar en el nuevo Sub-VM al que llamaremos SubA1.

- 3.- El siguiente paso será definir los pines de entrada y salida del nuevo elemento creado, teniendo cuidado de definir el nombre del pin y la naturaleza de la señal que va a gestionar tanto si es de entrada como si es de salida.
- 4.- Una vez que se ha realizado la definición de los pines lo que corresponde es guardar el nuevo Sub-VM que ya tiene puesto el nombre que le dimos anteriormente SubA1. Observamos como se incorpora en el árbol de proyecto e la carpeta en la que estemos trabajando.

Este nuevo Sub-VM estará disponible para cuando lo necesitemos utilizar.

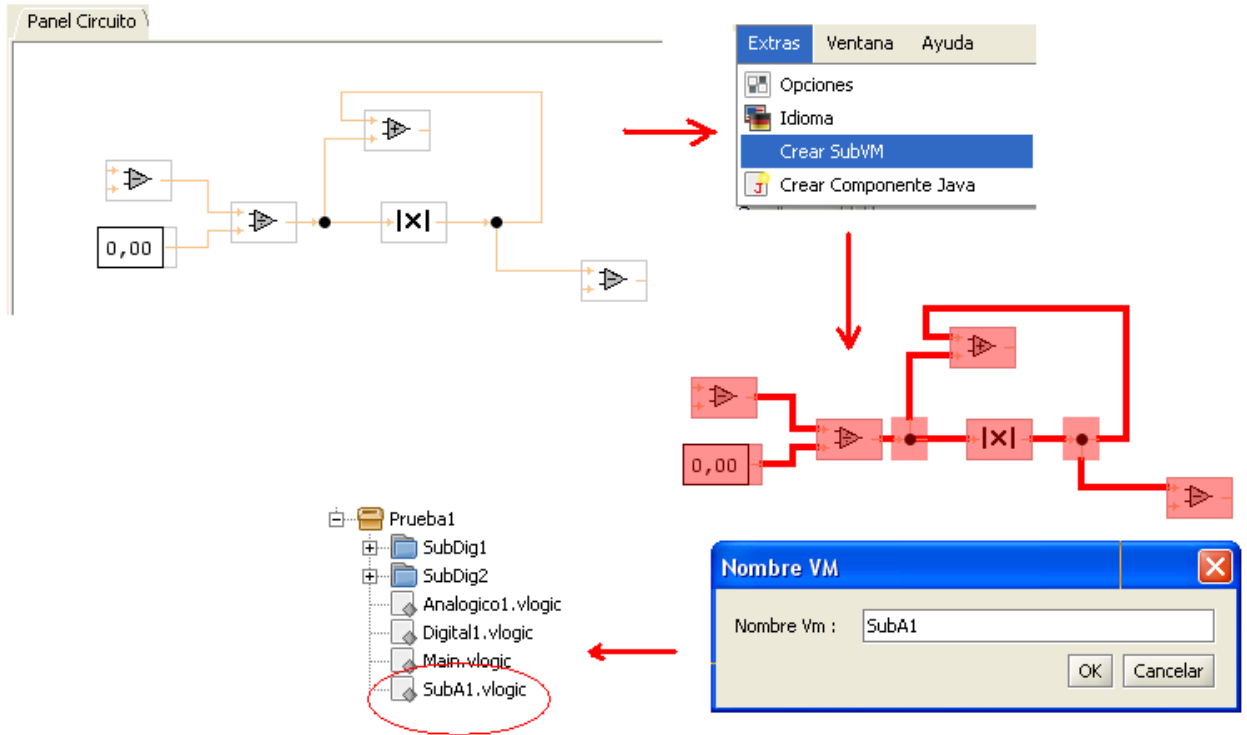


Figura 52

Proceso de creación del Sub-VM SubA1.vlogic

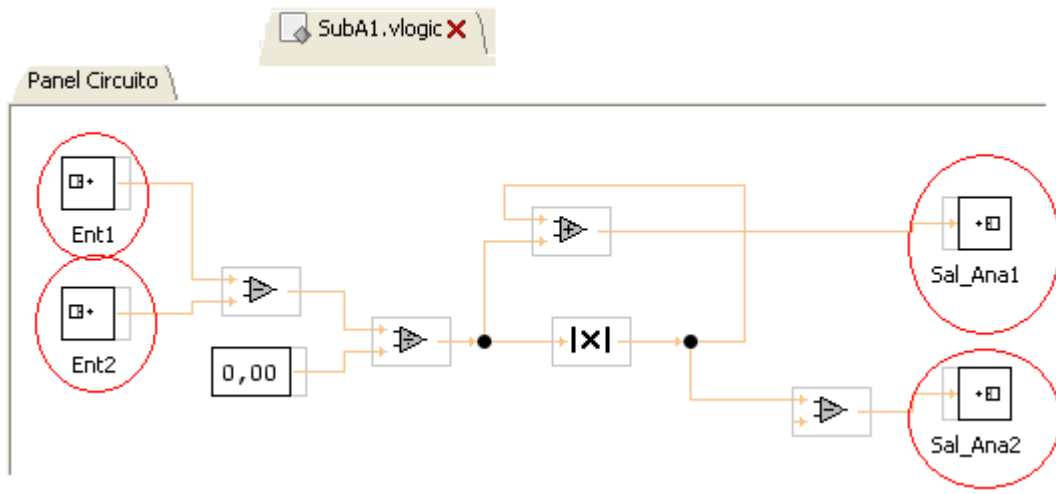


Figura 53

Designación de los pines de entrada y salida del Sub-VM SubA1.vlogic

Bastara con arrastrar el Sub-Vm al área de trabajo del “Panel de Circuito” y ya dispondremos de nuestro nuevo SubA1.vlogic.

Obsérvese que el componente aparece con sus entradas y salidas etiquetadas perfectamente.

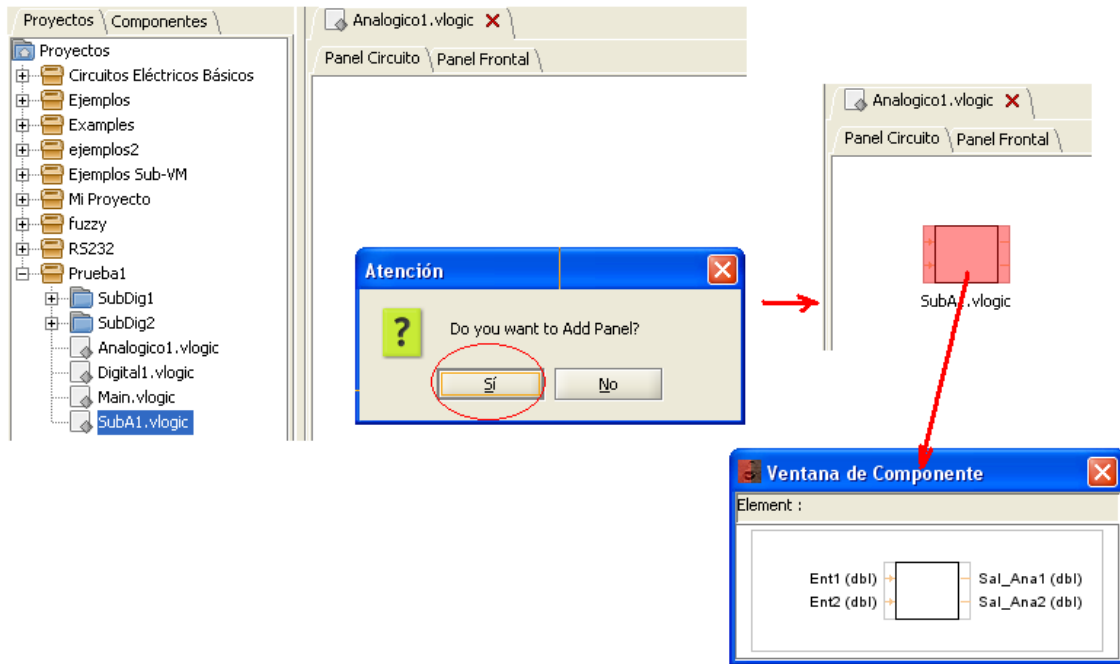


Figura 54

### Método 3

Este tercer método consiste en hacer uso del componente de librería marcado en la figura siguiente.

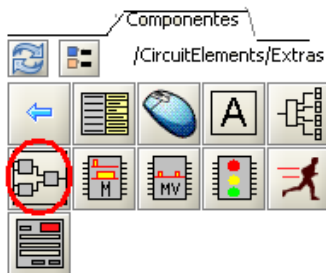


Figura 55

Este componente permite directamente escribir el nombre del fichero asociado al Sub-VM y lo incluye en el modelo que estemos realizando.

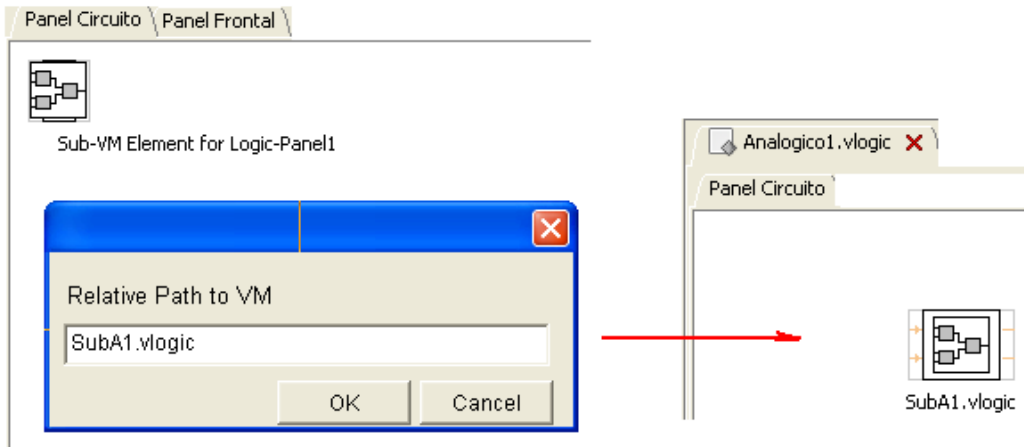


Figura 56

## 10. Creación de nuevos elementos para las librerías de MyOpenLab.

### Método 1: Creación utilizando los recursos de MyOpenLab

A continuación vamos a explicar la forma de crear nuevos elementos o bloque de función a las librerías.

Esta opción es muy útil por cuanto que permite incorporar nuevas librerías a las existentes con el objeto de facilitar su uso repetitivo ahorrándonos trabajo y simplificando la realización de nuevas simulaciones (VM's).

Myopenlab tiene sus librerías agrupadas en dos grandes bloques:

**Librerías para Panel Circuito**  
**Librerías para Panel Frontal**

Ambas se ubican en sendas carpetas separadas:

... \Distribution\Elements\CircuitElements  
... \Distribution\Elements\FrontElements

En nuestro caso los elementos nuevos que incorporaremos se ubicaran en la carpeta que hayamos designado para ello mediante la opción correspondiente dentro del menú Extras-> "Opciones"

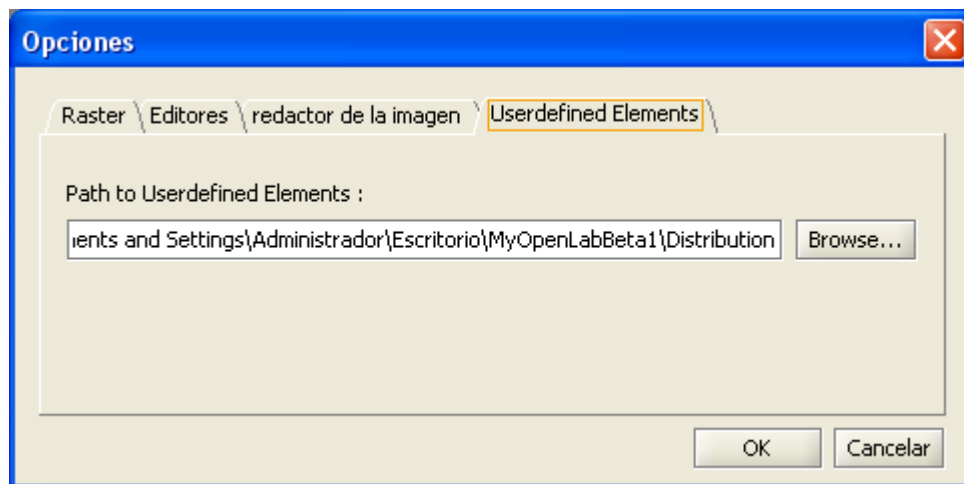


Figura 57

Nosotros vamos a grabar los componentes que creamos en la carpeta:

... \Distribution\Mi Librería

Dentro de la cual estarán las dos carpetas

... \Distribution\Mi Libreria\CircuitElements  
... \Distribution\Mi Libreria\FrontElements

En donde se guardaran los elementos que vayamos creando.

Tengamos muy en cuenta que a la hora de crear un componente de librería si este no contiene ningún elemento correspondiente a la librería Para Panel Frontal se guardará en la carpeta correspondiente a los elementos de Librería de Panel Circuito. Si acaso contiene alguno se grabará en la Librería de Panel Frontal.

## Creación de un Elemento que calcule la media de dos números.

Vamos a crear un componente de librería que se encargara de realizar la media aritmética de dos números que se le introduzcan en sus dos entradas. Los números serán en formato “dbl”. Se trata por lo tanto de una librería netamente de cálculo matemático y que por lo tanto no tendrá ningún componente de tipo “visual”. Por ello se grabara en el directorio correspondiente a los elementos “CircuitsElements” (elementos propios del Panel de Circuito).

### 1. Creación del VM correspondiente

Crearemos un modelo como el de la figura. Tengamos en cuenta que las entradas y salidas irán necesariamente conectadas a elementos tipo “Pines” en los que definiremos la naturaleza de los datos (en este caso son todos de tipo “dbl” (seleccionando C\_DOUBLE en el menú de propiedades de los pines).

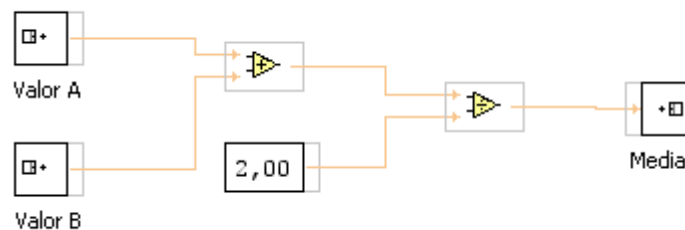


Figura 58

### 2. Guardar el fichero como elemento.

Una vez que esté terminado el esquema procederemos a guardar el fichero con la opción “Grabar VM como Módulo”.

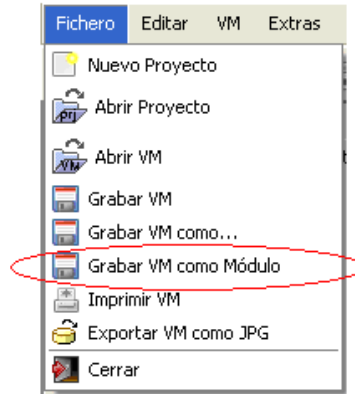


Figura 59

Aparecerá una ventana en la que definiremos varias cosas:

**Name:** El nombre del Elemento

**Caption:** Los textos en Alemán, Inglés y Español con los que se mostrar cuando estemos sobre ella en el bloque de librerías

**Icon:** El icono con el que aparecerá. Será editable.

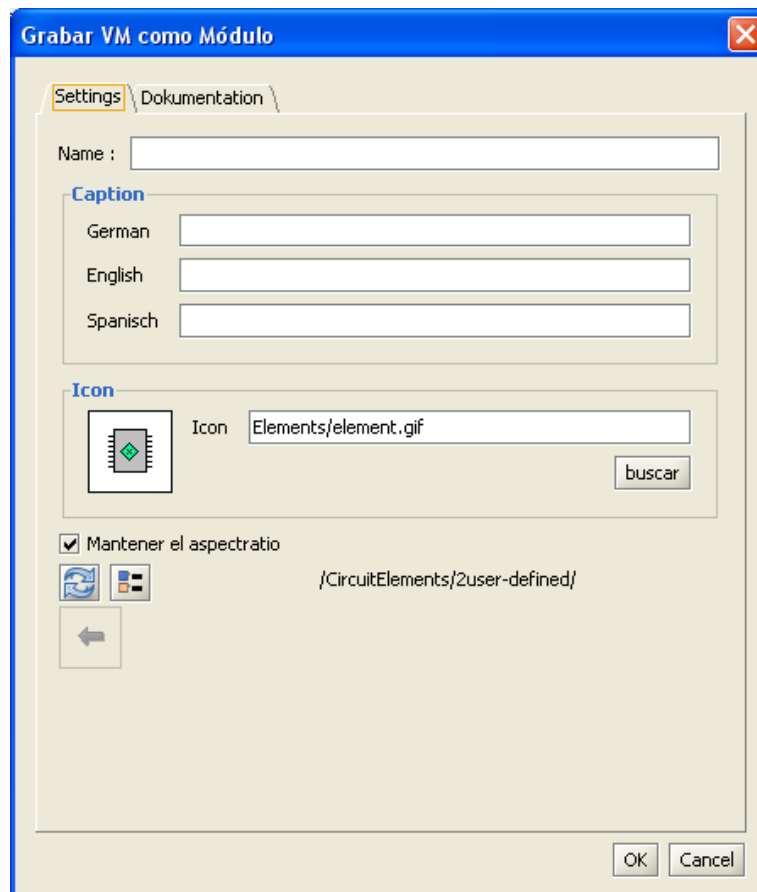


Figura 60

El componente una vez creado y si lo colocamos en el área del Panel de Circuito aparecerá tal como se muestra en la figura.

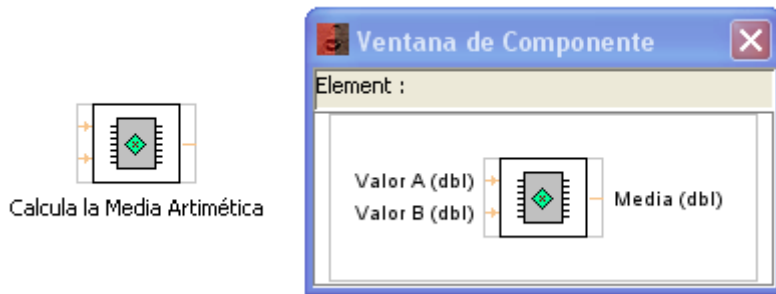


Figura 61

A partir de este momento el Elemento “Media” está listo para ser usado en cualquier VM que construyamos... En la figura aparece una sencilla aplicación de uso. Si quisiéramos cambiar el icono podríamos hacerlo sencillamente editándolo.

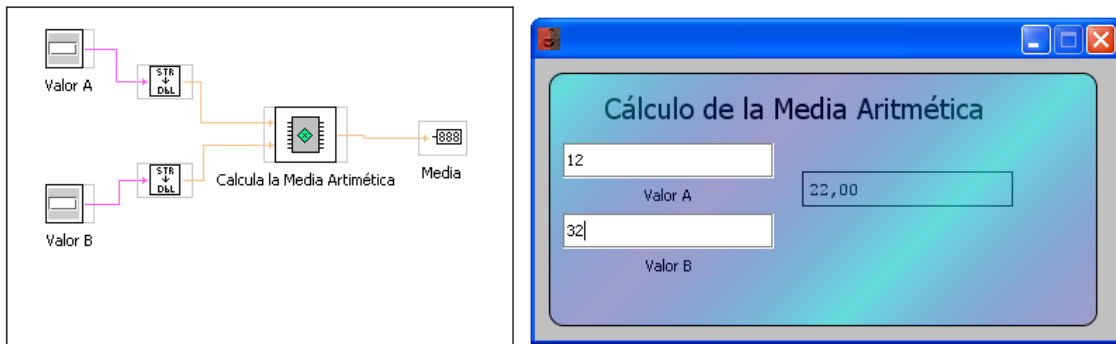


Figura 62

No olvidemos que en el lugar correspondiente ha quedado creada la carpeta que contiene este nuevo elemento.

... \Distribution\Mi Libreria\CircuitElements\Media

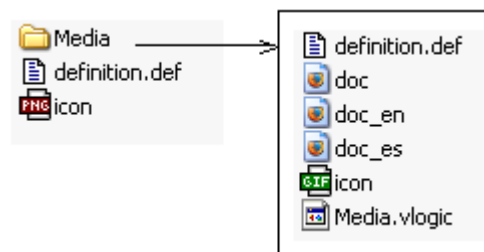


Figura 63

En las librerías parecerá tal como se muestra en la siguiente figura

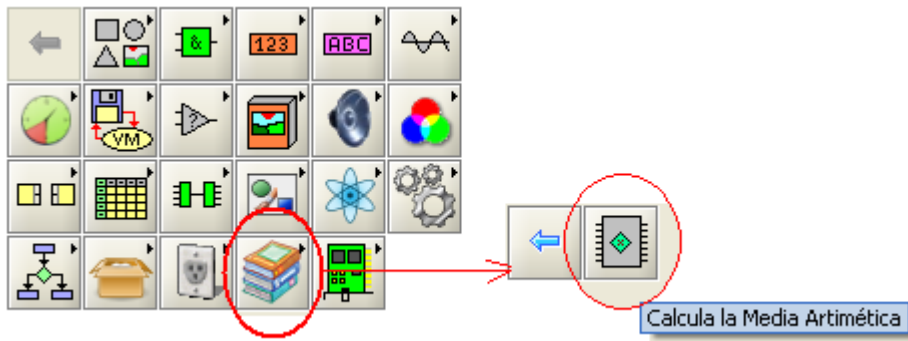


Figura 64

## Método 2: Creación escribiendo el código del componente en JAVA.

El método de creación de componentes que se explica a continuación está orientado a quienes conozcan el lenguaje Java y sean capaces de crear funciones que relacionen las variables de entrada y los parámetros que intervienen en un componente. Seguiremos los siguientes pasos:

### 1. Seleccionamos del menú la opción Crear Componente Java

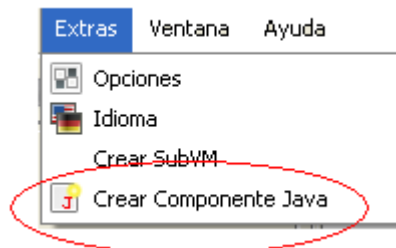


Figura 65

### 2. Configuración de los parámetros del componente

Inmediatamente aparece la ventana de parámetros y configuración de componente que se muestra a continuación.

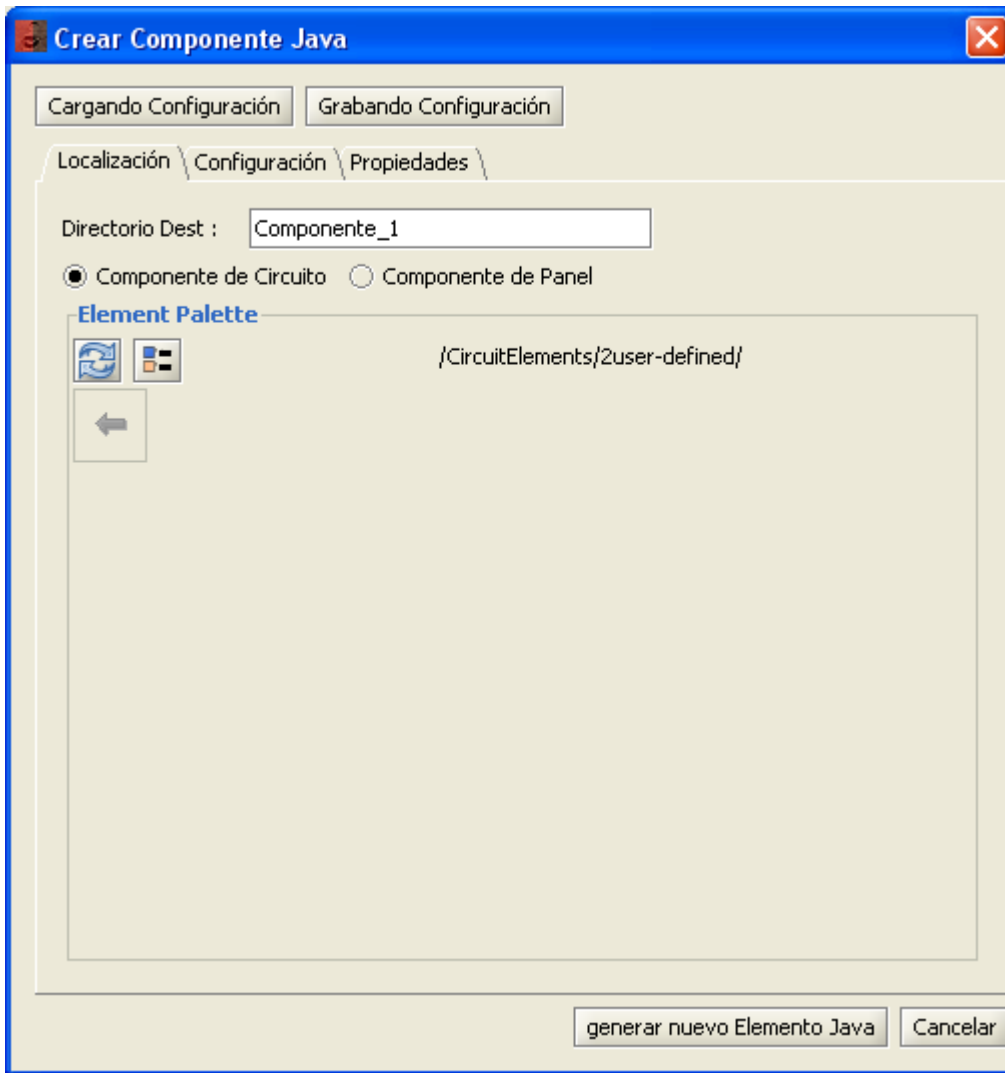


Figura 66

Desde esta ventana damos el nombre del Directorio Destino del componente estando en la pestaña de "Localización".

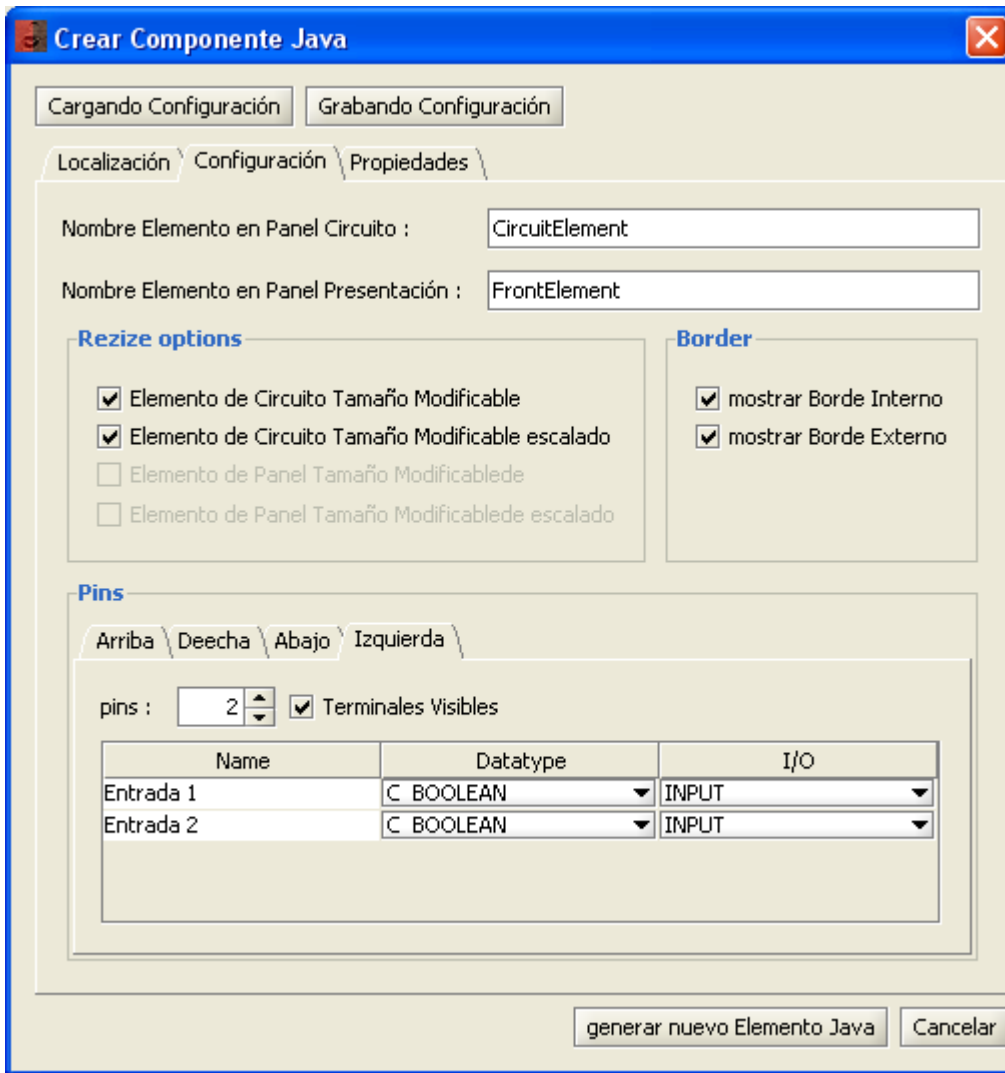


Figura 67

Mediante la pestaña de “Configuración” definimos los parámetros que hacen referencia a:

**Nombre:** “Nombre del componente” con el que aparecerá en el Panel Circuito y en el Panel Frontal.

**Resize Options:** Opciones de modificación de tamaño del componente cuando este se coloca en los paneles

**Border:** Aspecto del borde del componente una vez colocado en los paneles.

**Pins:** Desde aquí se configuran los pines (entradas/salidas) del componente pudiendo estar estas en la derecha, Izquierda, arriba y debajo de la caja con la que se representa el componente. Del mismo modo se definen los tipos de datos de cada pin y si son entradas o salidas (I/O).

En la pestaña de Propiedades podremos designar los Parámetros de nuestro componente mediante su “Nombre”, “Tipo de dato” y valores “Min” y “Max”



### 3. Edición y Compilación del componente creado.

El componente que acabamos de definir aun no esta disponible para poder utilizarse dado que lo único que hemos hecho es crear la estructura del programa Java de dicho componente. Ahora lo que hemos de hacer es escribir el código Java de acuerdo con la función que deseamos que realice el componente y después compilarlo.

El componente ha sido creado en el grupo de librerías de usuario tal como se ve en la figura siguiente

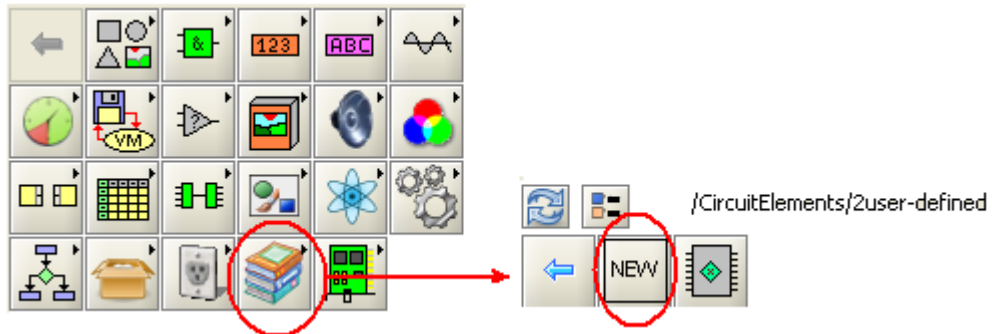


Figura 70

Bastará que nos situemos sobre el icono del componente y pulsemos el botón derecho del ratón apareciendo el menú que se muestra en la figura y desde este podremos seleccionar cada una de las opciones de edición que se muestran.

En la imagen siguiente se muestra la ventana de “Editor definición de Elementos” desde la que se pueden modificar varias cosas entre ellas el icono del Elemento creado. La opción Caption permite escribir el texto que aparecerá cuando pasemos el ratón sobre su icono.

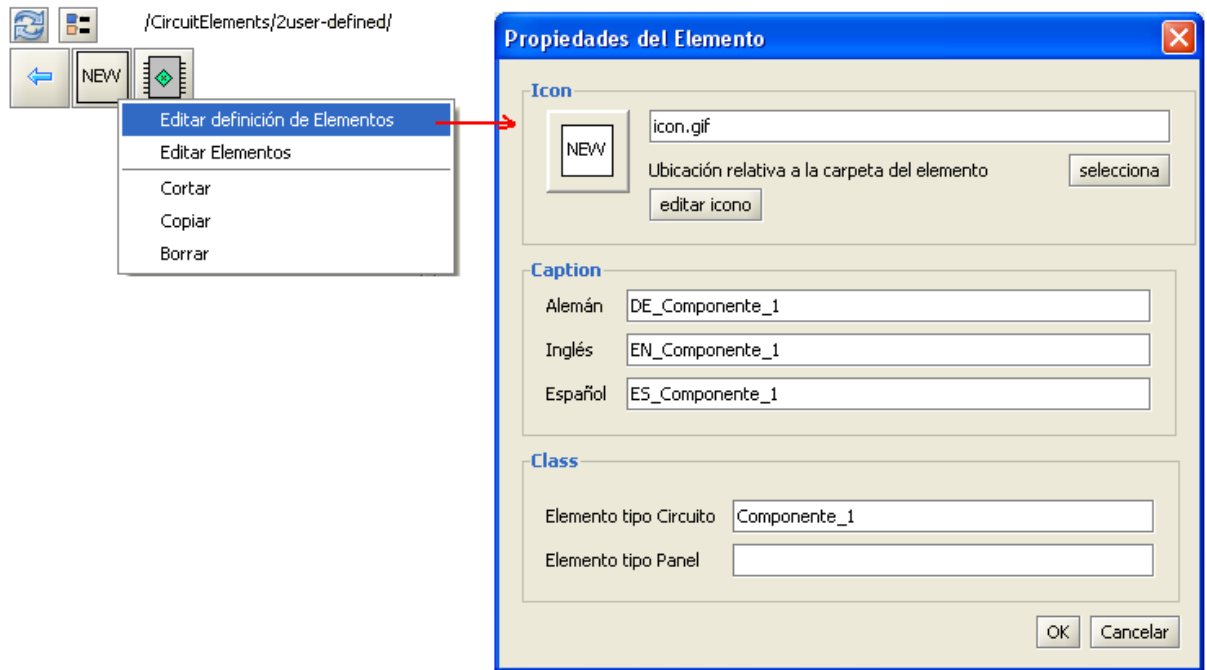


Figura 71

Con la opción Editar Elementos se despliega la ventana de edición de código java desde la que podemos escribir, añadiendo al código básico ya escrito, el código que permita la realización de nuestra función para este componente.

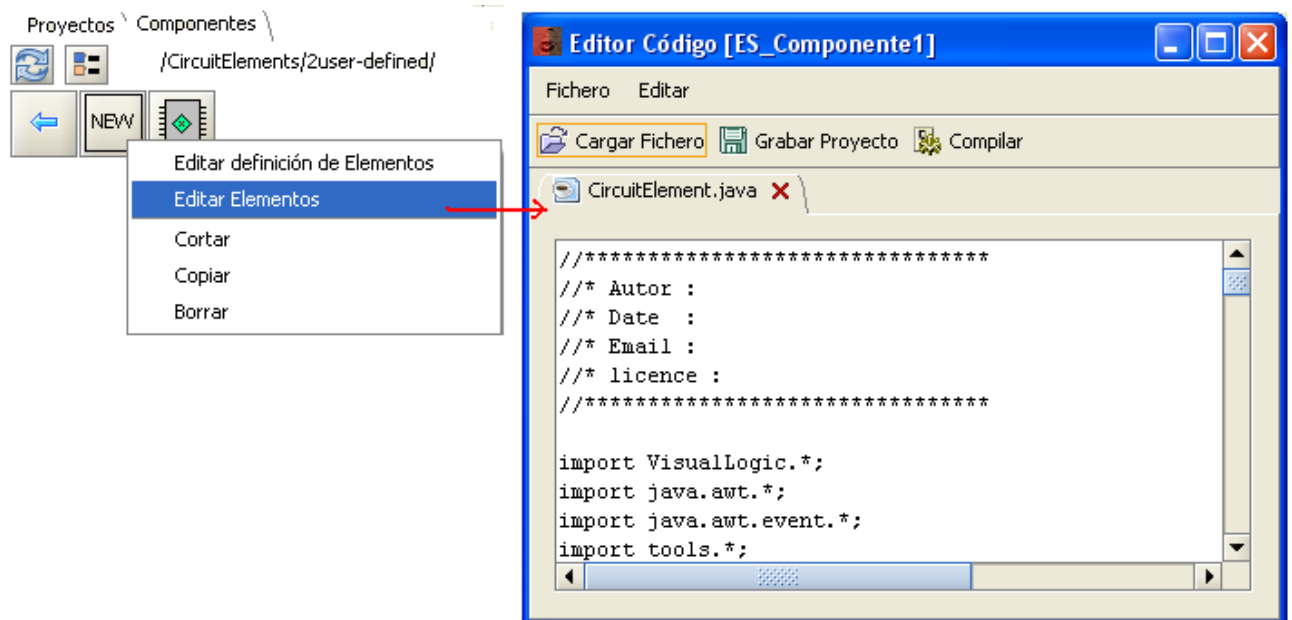


Figura 72

Una vez escrito el código debemos pulsar la opción “Compilar” que será la que genere definitivamente el componente utilizable.

El código del componente podremos editarlo cuantas veces queramos.

En la figura siguiente vemos el aspecto que tendrá el componente que acabamos de crear en donde aparecen la identificación de sus terminales (Ventana de Componente) y el Editor de Propiedades en donde aparece el parámetro que definimos anteriormente “Parámetro 1”

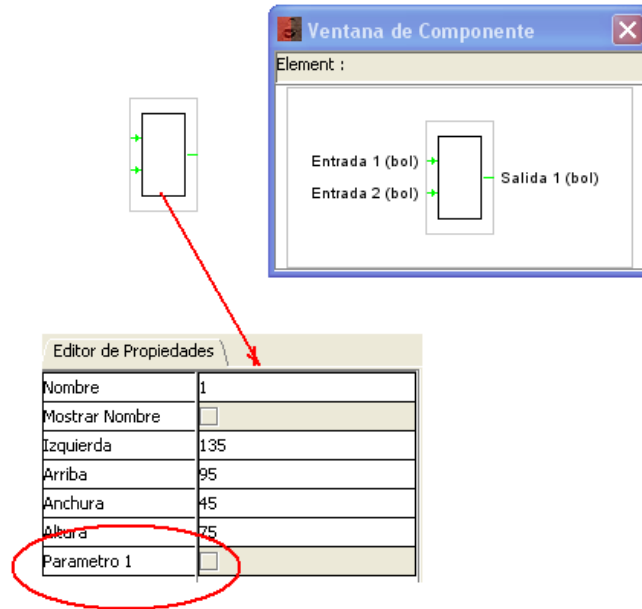


Figura 73

## 11. Protección y personalización de nuestros trabajos

En la nueva versión de Myopenlab aparece la posibilidad de proteger nuestros desarrollos de VM's mediante una clave (password) con el fin de que al abrirlos solo aparezca la pantalla correspondiente el Panel Frontal y no la del Panel de Circuito. De esta manera el que utilice abra la aplicación VM solo podrá realizar la simulación pero no modificarla.

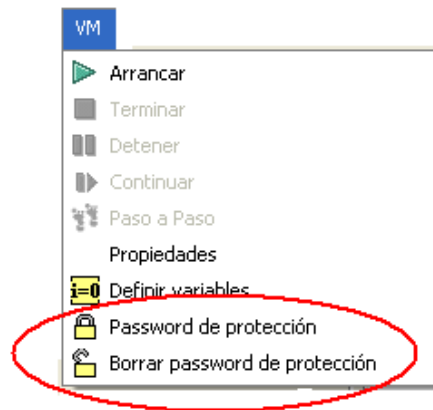


Figura 74

Para proteger una aplicación VM procederemos de la siguiente manera:

Abrimos el fichero VM y seleccionamos la opción “Password de Protección”. Aparecerá una ventana en la que se nos pide el password (mínimo 6 caracteres), lo escribimos y seleccionamos “OK”. Ya está protegido el fichero, pero no olvidemos guardarlo cuando lo cerremos o salgamos del programa.

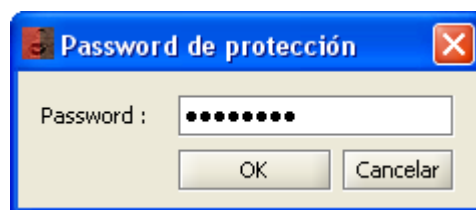


Figura 75

Cuando intentemos abrir un fichero protegido lo primero que se nos pedirá es su password en la correspondiente ventana. Si lo escribimos bien lo abrimos completo si le damos a “CANCELAR” abriremos solo el Panel Frontal..



Figura 76

Si deseamos quitar la protección de un fichero debemos seleccionar “Borrar Password de Protección” del menú Seleccionamos “SI” en la ventan y el fichero quedará totalmente abierto.

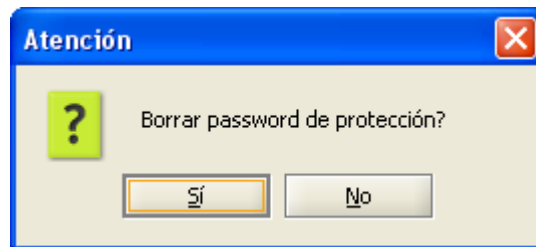


Figura 77

Apareciendo la ventana de mensaje siguiente.

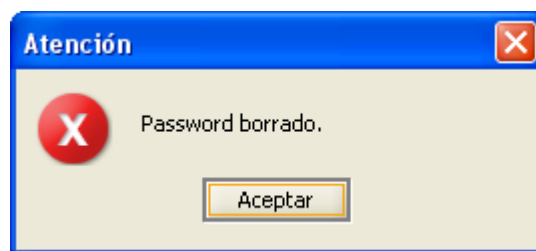


Figura 78

## Identificación del autor de una aplicación VM

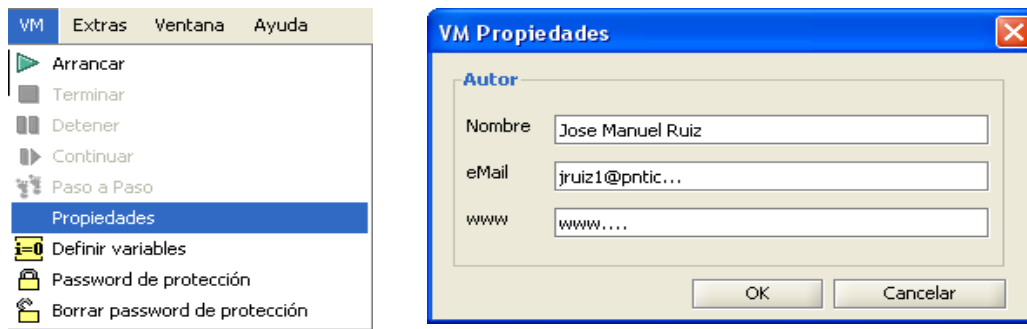


Figura 79

A la hora de realizar una aplicación con MyOpenlab es posible indicar en ella el nombre del autor, su correo electrónico y su dirección de página web. Esto se hace mediante la opción VM -> “Propiedades” del Menú. En la figura siguiente se ve dicha opción y la ventana de propiedades que despliega cuando se activa.

## 12. Trazado y seguimiento de la ejecución de una simulación VM.

Cuando se diseña un modelo y se realiza la simulación a veces es necesario realizar un seguimiento (trazado) de los valores que van adoptando determinadas señales del modelo. Para conseguir este objetivo MyOpenlab dispone de la posibilidad de añadir puntos de test “Añadir Testpoint” .

### ¿Cómo añadir un punto de test de señal?

Para añadir un punto de test de señal basta con situarse en la línea de la que queremos recoger el valor de la variable y pulsar el botón derecho del ratón. Aparecerá un menú contextual y seleccionamos “Añadir Testpoint” quedando una marca en la posición señalada que indicará que ahí tenemos un punto de test.

En la figura vemos los pasos a realizar:

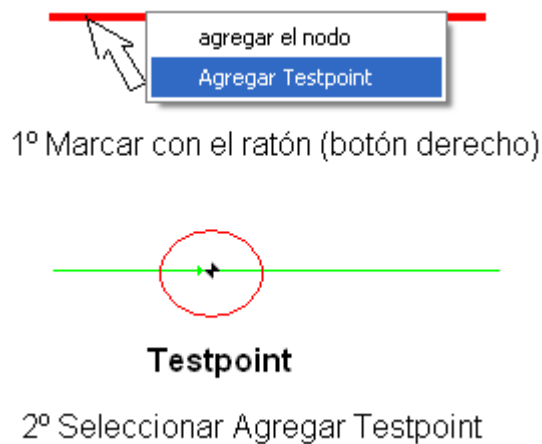


Figura 80

Habilitar la opción de mostrar gráficos de señales Testpoint en el modo simulación.

Para poder ver el comportamiento de las señales que hayamos marcado con la opción Testpoint en el modo simulación debemos habilitar la opción “Demostrar Toolbar” del menú de propiedades del Panel Frontal.

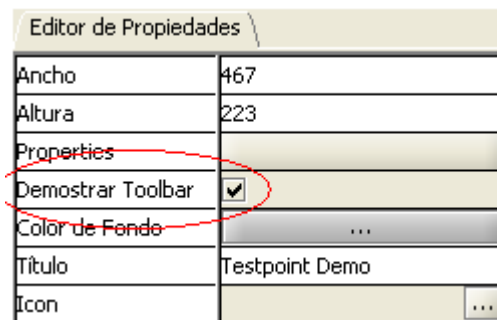


Figura 81

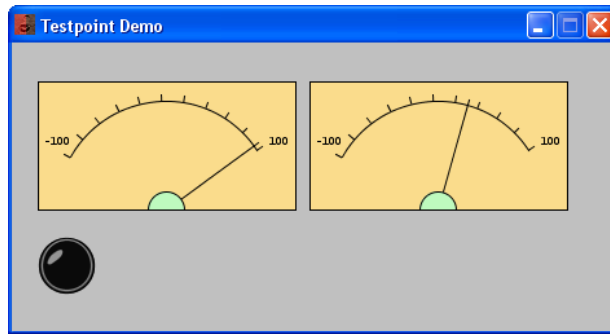


Figura 82

Con la opción “Demostrar Toolbar” deshabilitada

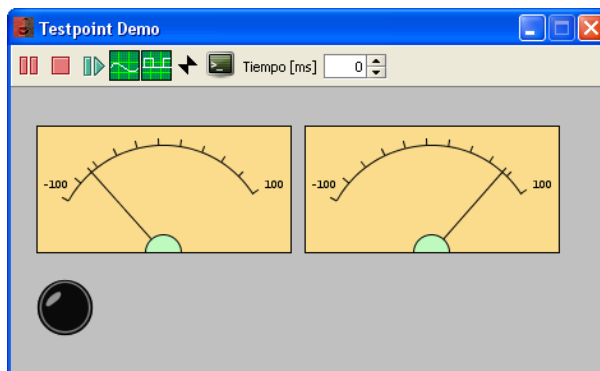


Figura 83

Con la opción “Demostrar Toolbar” deshabilitada

Estando habilitada esta opción se incorporara en la pantalla de simulación del Panel Frontal la barra superior de visualizadores en la que aparecen los siguientes tipos de visualizadores:



Ventana Gráfica Numérica






Ventana Grafica Digital



Ventana Tabla de Datos



Consola

	Detener Simulación VM
	Terminar Simulación VM
	Continuar Simulación VM
Tiempo [ms] <input type="text" value="0"/>	Ajustar el retardo de tiempo de simulación

En las siguientes figuras se ven las ventanas de información de los puntos Testpoint situados en una simulación VM.

También es posible abrir estas ventanas desde la barra de herramientas de la aplicación.

### Ventana Gráfica Numérica

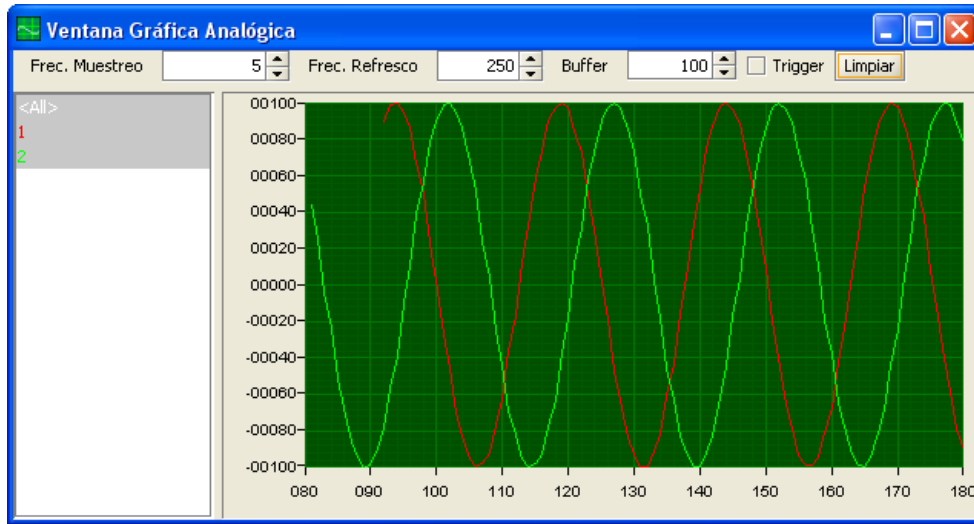


Figura 84

Mediante los controles Frec. Muestreo, Frec. Refresco, Buffer, Trigger y Limpiar se pueden configurar la visualización de los datos tanto analógicos como digitales.

### Ventana Gráfica Digital

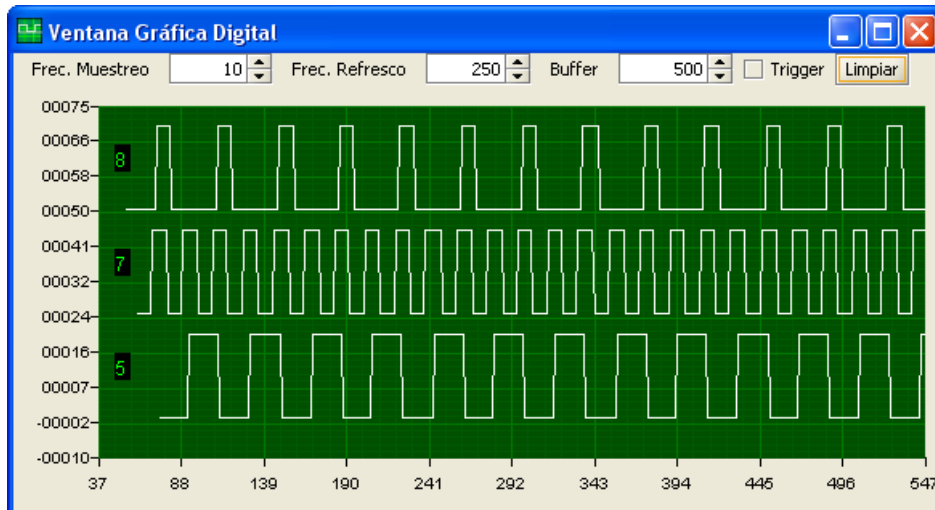
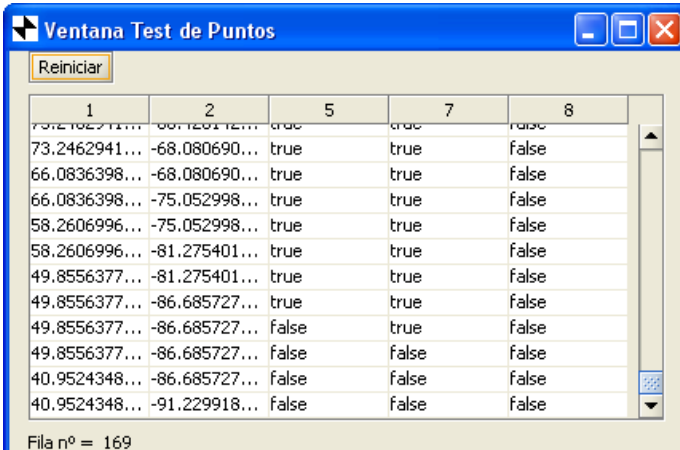


Figura 85

## Ventana Tabla de Datos



1	2	5	7	8
73.2462941...	-68.080690...	true	true	false
66.0836398...	-68.080690...	true	true	false
66.0836398...	-75.052998...	true	true	false
58.2606996...	-75.052998...	true	true	false
58.2606996...	-81.275401...	true	true	false
49.8556377...	-81.275401...	true	true	false
49.8556377...	-86.685727...	true	true	false
49.8556377...	-86.685727...	false	true	false
49.8556377...	-86.685727...	false	false	false
40.9524348...	-86.685727...	false	false	false
40.9524348...	-91.229918...	false	false	false

Fila nº = 169

Figura 86

## Visor de variables

En ocasiones será interesante ver el valor que toman determinadas variables a lo largo de la simulación de una aplicación VM. Ya hemos descrito las posibilidades del trazado de datos mediante las ventanas de graficas de datos analógicos, digitales y en forma de tabla.. En la forma explicada la simulación podrá correr en modo continuo.

En el caso que vamos a explicar haremos uso de la venta de visualización de datos llamada “Visor de Variables” y que aparece en el menú desplegable de “Ventana”, tal como se ve en la siguiente figura.

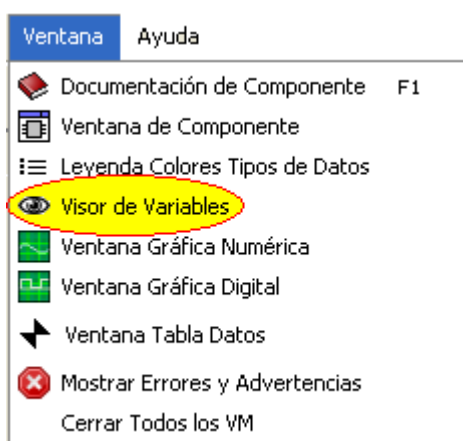


Figura 87

Digamos de antemano que en este caso las simulaciones se deberán de hace en “Modo debug” tal como se indica en la siguiente figura.

A la hora de definir variables es necesario hacerlo de manera expresa (las variables definidas por MyOpenlab no son accesibles en este modo de trabajo) mediante el editor

de variables que se activa en el botón  de la barra de botones de la ventana principal o bien a través del menú:

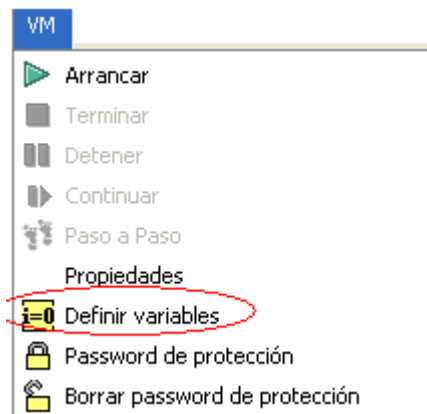


Figura 88

Con esta herramienta definimos las variables dándoles un nombre y un tipo de dato. Téngase en cuenta que en el tipo de dato solo podremos definir tres: Integer, String y Double

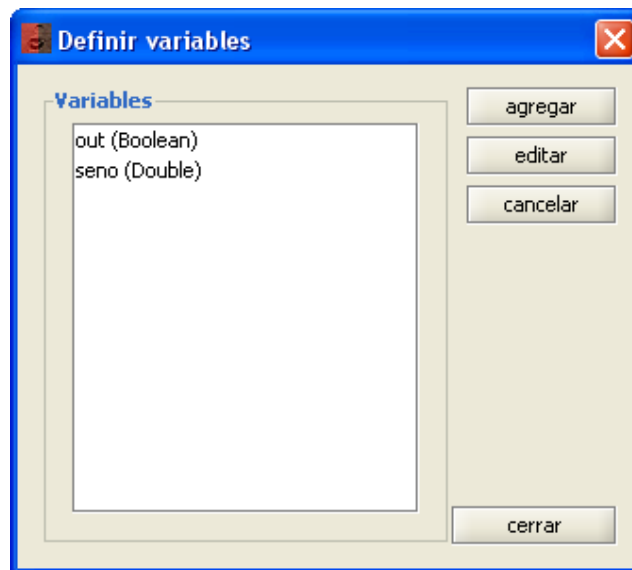


Figura 89

Cuando seleccionamos agregar aparece la ventana de la figura siguiente y en ella definimos la variable.

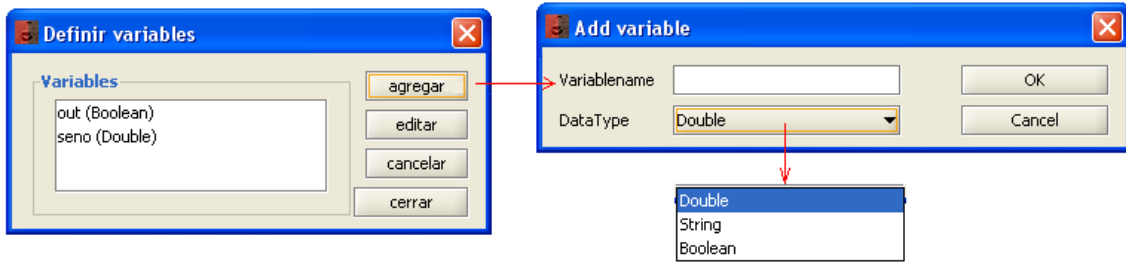
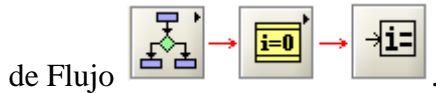


Figura 90

Una vez definidas las variables que necesitemos procederemos a extraerlas del “Panel de Circuito” mediante la función “Get(i)” que se encuentra en la librería de Diagramas



Para extraer el valor de la variable de la línea correspondiente del esquema VM basta con “agregar nodo” (función del menú contextual pulsando el botón derecho del ratón estando sobre la línea en donde queramos incluir el nodo).

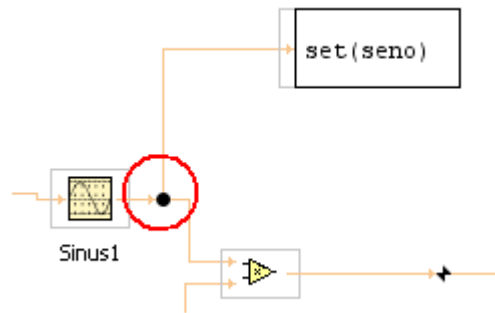


Figura 91

Cuando hemos definido las variables que queremos ver bastará que iniciemos la simulación del VM siempre en “Modo Debug”.

Abrimos la ventana “Visor de Variables” y podremos ver una lista de todas las variables definidas y sus valores evolucionando en función de los estímulos recibidos por el modelo.

En la siguiente figura vemos la evolución de un sencillo ejemplo de una aplicación con Diagrama de Flujo que implementa un contador de 0 a 10.

La variable en este ejemplo es “contador”

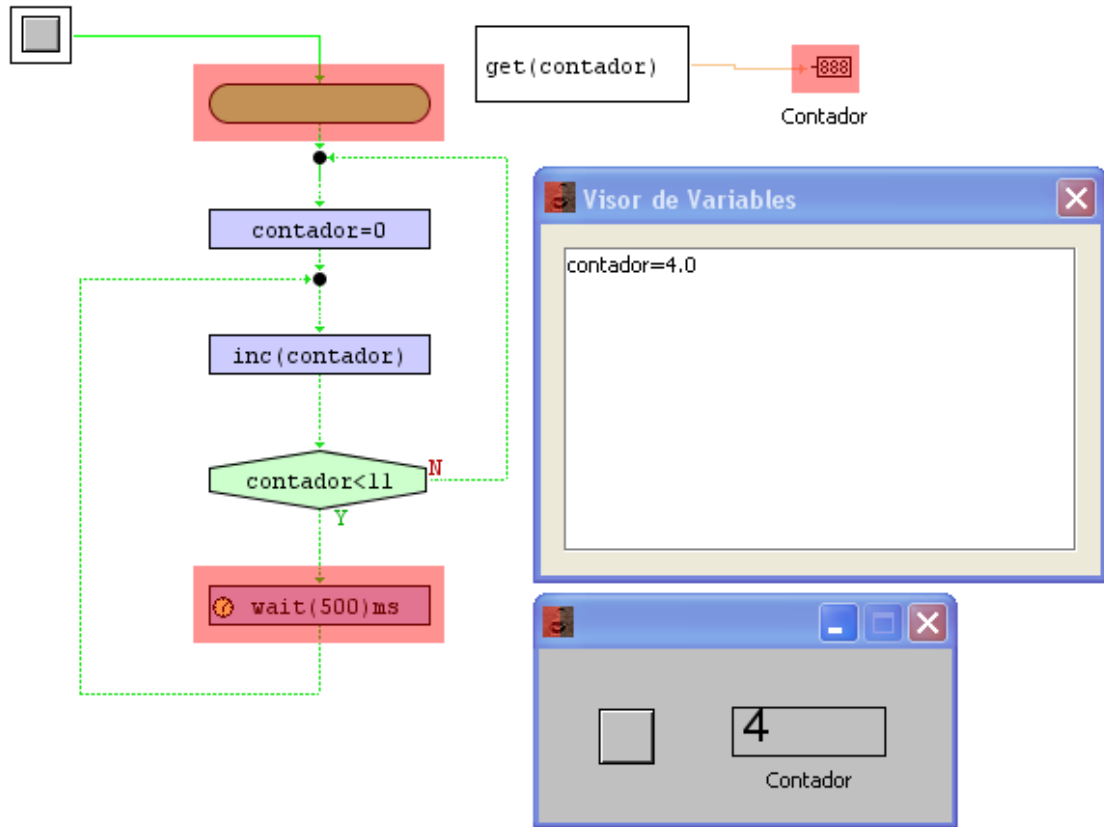


Figura 92

En el siguiente ejemplo vemos la simulación de una operación de suma en la que recogemos los valores de las variables de entrada “a” y “b” y la variable de salida “suma”.

Se indica en la figura el panel “Visor de Variables” junto a la pantalla de simulación

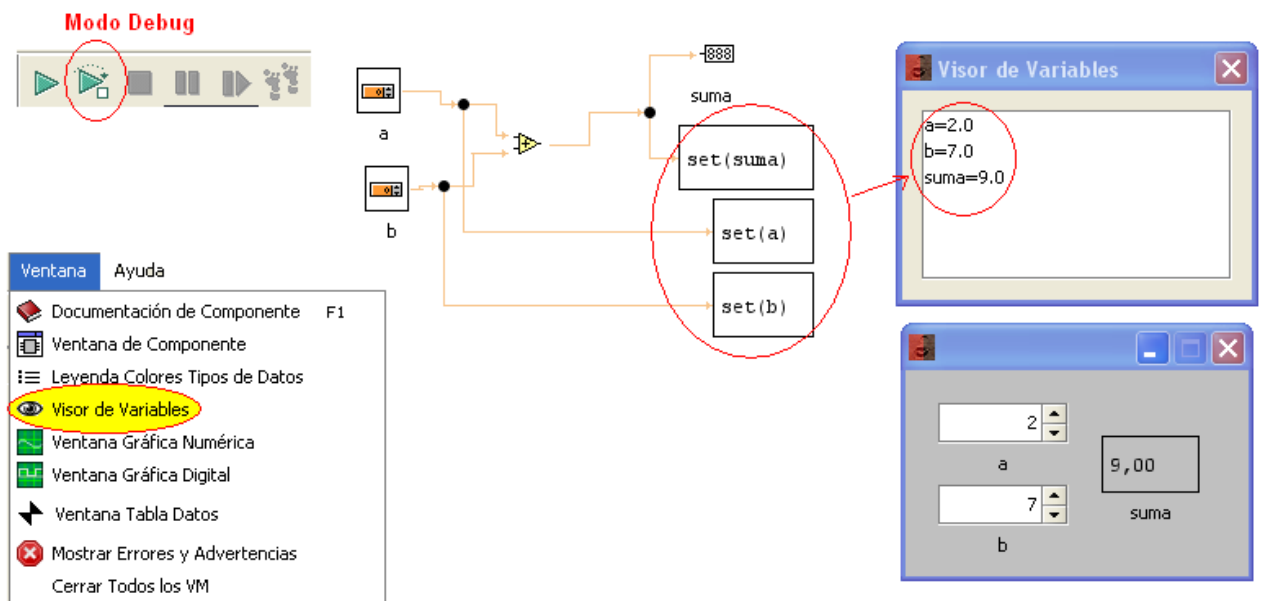


Figura 93

## Ejecución en modo debug

En algunas aplicaciones, a la hora de realizar su simulación nos interesará realizar la ejecución de la simulación en “modo debug”. Esto significa que podremos ver de manera grafica como se activan los elementos de entrada y de salida de valores y en definitiva permitirá explorar el flujo de datos.

Para entender esta forma de simulación lo mejor es utilizar un ejemplo en el que exista un diagrama de flujo pues se podrá ver con claridad como se activan en secuencia los distintos bloques que lo constituyen..

La opción se activa mediante el botón . El ajuste del tiempo de simulación se realiza con .

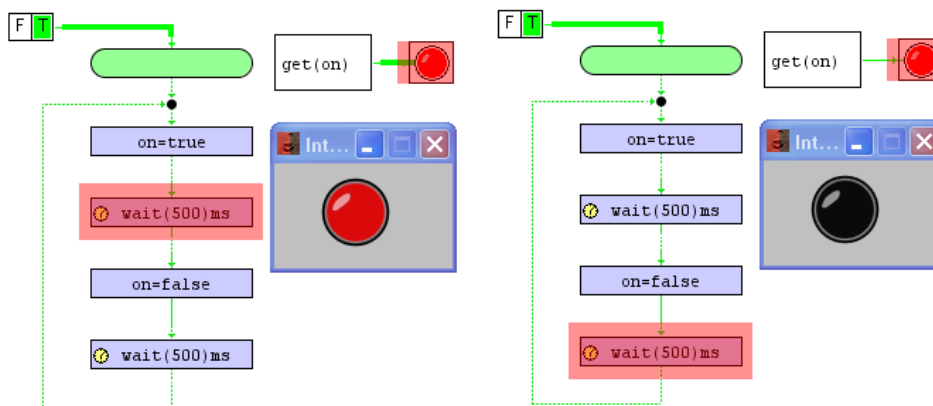


Figura 94

## Ventana de “Errores y Advertencias”.

Esta ventana nos muestra los posibles errores o advertencias a lo largo de la simulación o en el caso de que se efectúen operaciones no validas.



Figura 95

Se puede acceder a ella a través del menú "ventana", tal como se ve en la figura

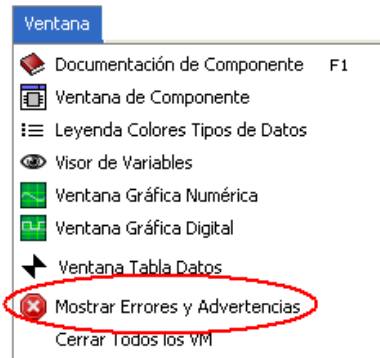



Figura 96

### ***Ventana "Salida Consola".***

Esta ventana permite visualizar los datos que se envía a la consola. 

Las órdenes que envían datos a la consola recordemos que son:

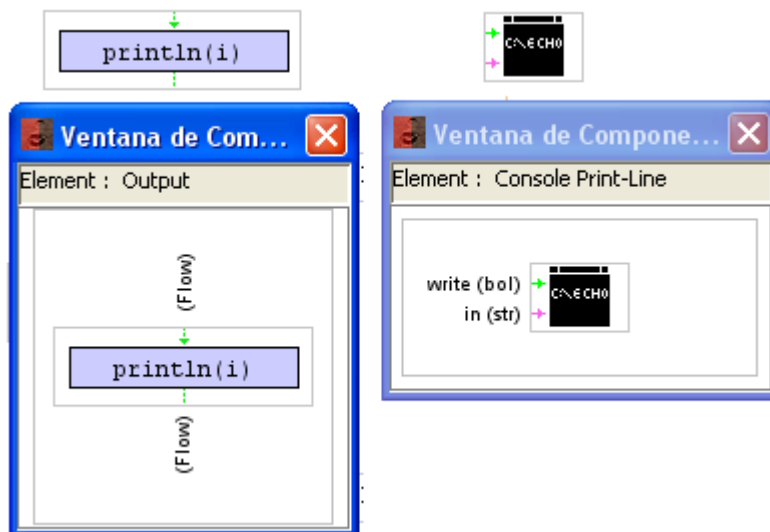


Figura 97

El aspecto de la ventana de consola es el de la figura siguiente, que corresponde a un sencillo ejemplo:



Figura 98

## 13. Otras Opciones e Informaciones

### *Opciones generales*

MyOpenlab tiene una opción para configurar algunos parámetros de funcionamiento. Para activar la ventana de opciones (Figura 99) se recurre a la opción del menú Extras->Opciones.

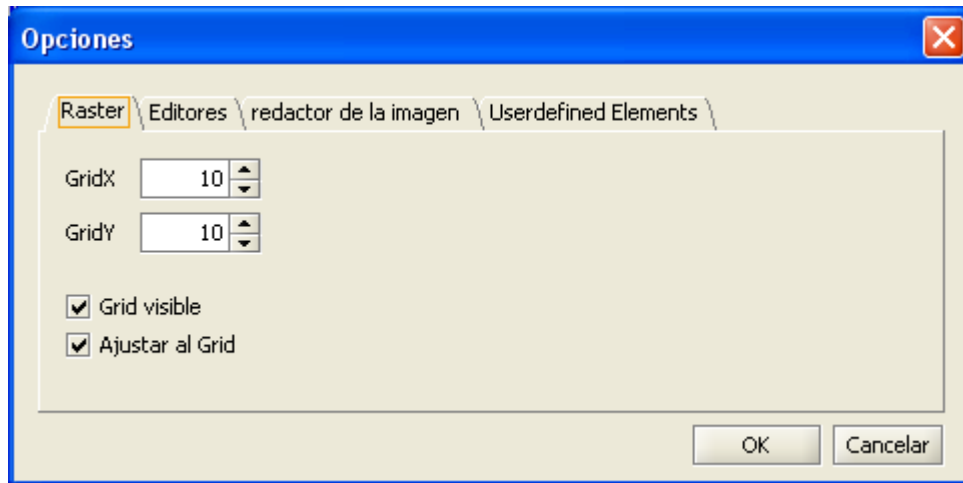


Figura 99

Activando las distintas pestañas que aparezcan podremos configurar:

**Raster:** Fija el tamaño del grid en la pantalla de edición "Panel Frontal" de la aplicación y la posibilidad de mostrarlo o no y ajustar a él el encaje de los componentes.

**Editores:** Permite seleccionar el editor de formato HTML para editar los ficheros de ayuda de los componentes.

**Redactor de la imagen:** Permite seleccionar el editor de las imágenes para el caso de modificar los iconos de las librerías.

**Userdefined Elements:** Permite definir el lugar (carpeta en el que se guardarán los Elementos de librería que podamos crear.

### ***Modificación del idioma***

Es posible que MyOpenmlab traiga tres idiomas: Inglés, Alemán y Español, para ello se recurre a la opción Extras-> Idioma del menú y se modifica. Será necesario arrancar de nuevo el programa para que tome el idioma seleccionado. Figura 100



Figura 100