

Objetos Canvas

Versión 3.0.8.3

Documentación para el usuario

www.MyOpenLab.de

Prof. José Manuel Ruiz Gutiérrez

1. REALIZACIÓN DE SIMULACIONES CON OBJETOS CANVAS

INTRODUCCIÓN

En este documento vamos a estudiar las posibilidades que ofrece MyOpenLab para realizar simulaciones en las que intervengan objetos gráficos animados.

El huso de las herramientas de animación resulta muy interesante en aquellas simulaciones en las que tengamos intención de mostrar un sinóptico animado o queramos ver la evolución de un objeto o imagen en la pantalla movido en función de los valores de las variables de un modelo matemático que pueda ser el núcleo de la simulación.

Las librerías para la realización de animaciones se encuentran en la carpeta “Canvas”

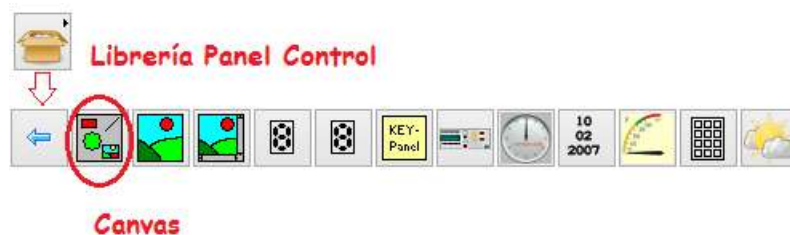


en cuyo interior aparecen los siguientes componentes:



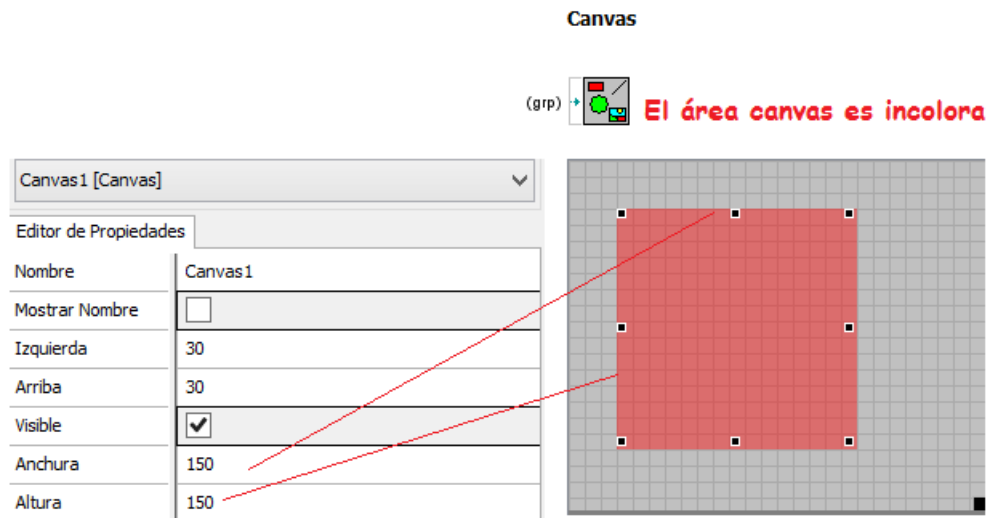
Estos componentes permiten una gran cantidad de animaciones y todos ellos se pueden integrar en un espacio gráfico de animación al que llamaremos “*espacio canvas para la animación*”.

El componente que define el espacio canvas se encuentra dentro de las librerías de componentes del panel de visualización. Tal como se muestra en la figura

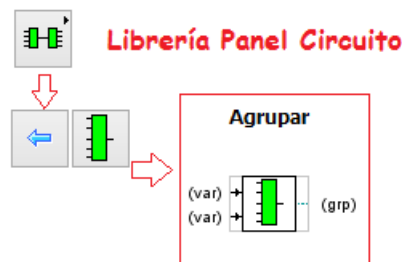


Este objeto, netamente gráfico crea un espacio de tamaño configurable en la pantalla dentro del cual podremos introducir otros objetos.

Este objeto tiene como entrada un grupo de imágenes que lógicamente deberá recibir a través del componente *de*



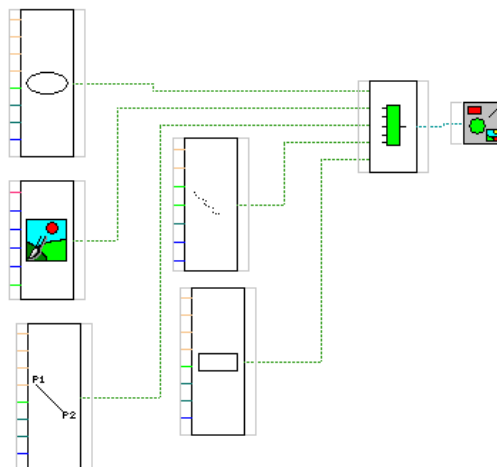
“Agrupación” llamado *Agrupar* que se muestra en la figura y que esta disponible en la librería del Panel Circuito.



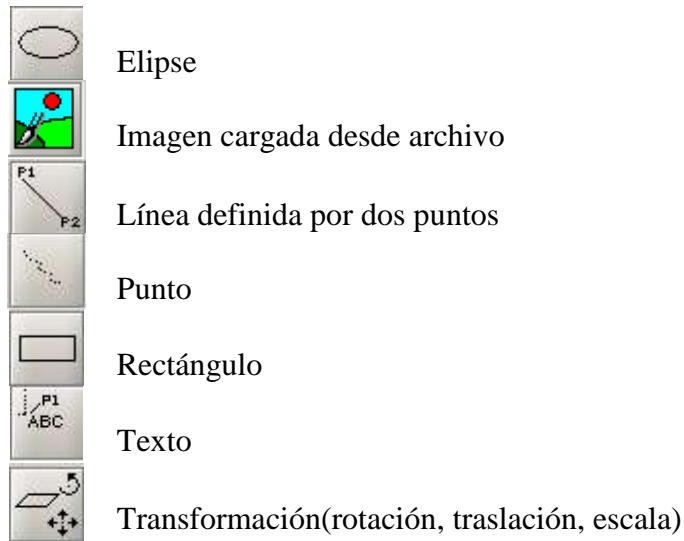
Componente Agrupar

El componente agrupar permite modificar el número de elementos de entrada mediante la opción “Pins Count” de sus propiedades. Por defecto el número de entradas es 2.

En la figura vemos como a través del elemento Agrupar se conectan varios objetos gráficos.

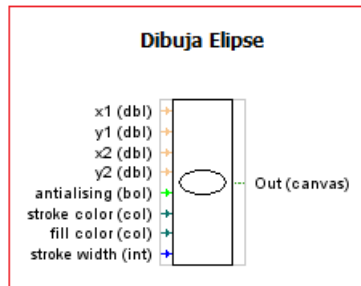


Los objetos que se pueden integrar dentro de un espacio Canvas son los que aparecen en la tabla siguiente.



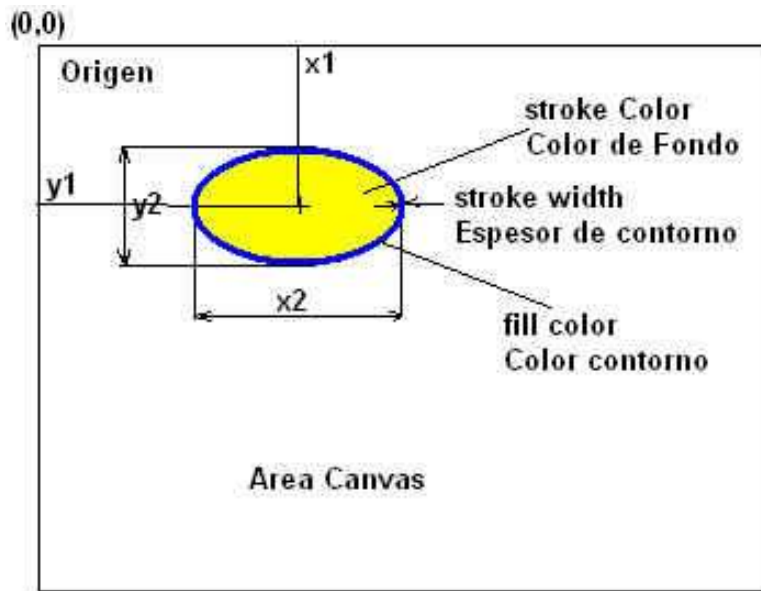
ELIPSE

Este objeto permite dibujar un círculo dentro de un área "canvas" del que podemos controlar las siguientes variables:



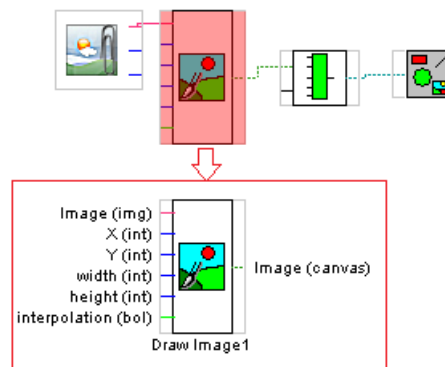
- **x1, y1:** Coordenadas del centro de la elipse
- **x2, y2:** Valores del eje mayor y menor de la elipse
- **antialiasing:** Corrige pixelado de contornos
- **stroke color:** Color de la línea de contorno
- **fill color:** Color de relleno
- **stroke width:** anchura de la línea de contorno

Es importante observar que el origen de las coordenadas gráficas del área Canvas está en la esquina superior izquierda tal como se indica en la figura.



IMAGEN

Este objeto permite colocar en el área Canvas una imagen que previamente se ha leído de un fichero. La imagen puede ser de formato **JPG, GIF o PNG**.



La imagen vemos que se carga mediante la función “*Static Image*” que forma parte de



la librería de imagen. este objeto tiene también de salida los valores (integer) del tamaño de la imagen, que en ocasiones pueden sernos útiles.

Las variables que se pueden manejar en este objeto son:

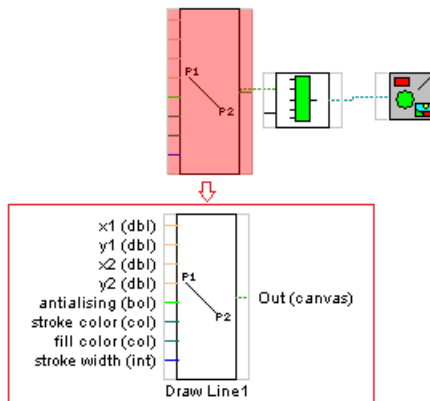
- **Image:** La entrada de la imagen
- **X,Y :** que indican la posición de la imagen dentro del Canvas en la que este integrada.
- **width:** Modifica la anchura de la imagen

- **height:** Modifica la altura de la imagen
- **Interpolation:** Alisa el pixelado de la imagen.



LINEA

Este objeto permite dibujar una línea entre dos puntos definidos como (x1,y1 ;x2,y2). Los parámetros que controlan este objeto son:



- **x1,y1:** Coordenadas del punto P1
- **x2,y2:** Coordenadas del punto P2
- **antialiasing:** Corrige el pixelado de contornos
- **stroke color:** Color de la línea de contorno
- **fill color:** Color de relleno
- **stroke width:** anchura de la línea de contorno

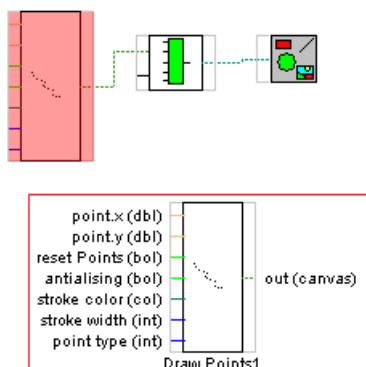
En la figura vemos una sencilla aplicación en donde se ha dibujado una línea entre los puntos P1(100,100) y P2(180,150). No olvidemos que las unidades vienen dadas en pixels.

En la figura vemos un sencillo ejemplo en el que se dibuja una línea que une los puntos P1(100,100) y P2(180,150)



PUNTO

Mediante esta función se consigue dibujar un punto en el espacio definido por el Canvas grafico. En esta ocasión las variables que se controlan son:



- **point x:** Coordenada x del punto
- **point y:** Coordenada y del punto
- **reset points:** borra las marcas de los puntos dibujados anteriormente.
- **antialiasing:** Corrige el pixelado de la imagen del punto
- **stroke color:** Color del punto
- **stroke width:** grosor del punto

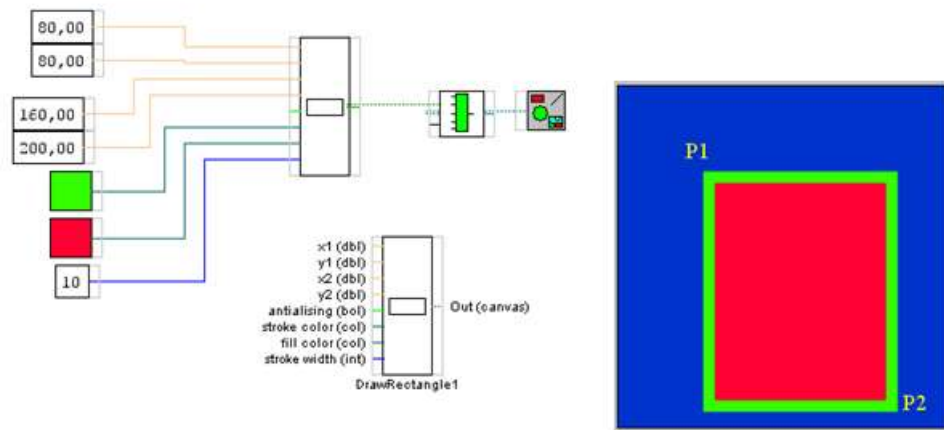
- **point type:** tipo de unión entre el punto actual y anterior(0 mediante puntos, 1 mediante segmentos)



RECTANGULO

Este bloque dibuja un rectángulo en el espacio gráfico Canvas de acuerdo con los siguientes parámetros:

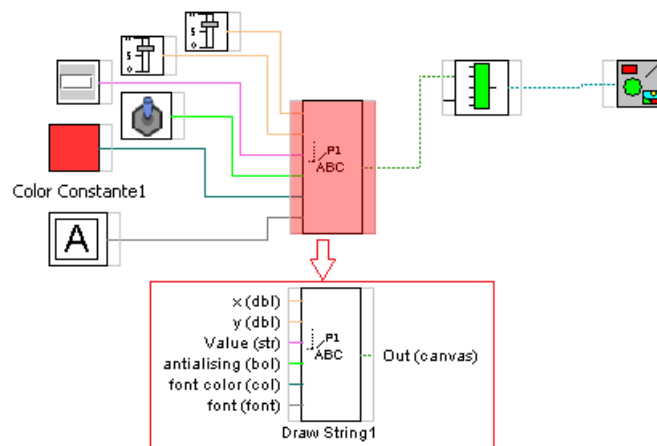
- **x1,y1:** coordenadas del punto P1
- **x2,y2:** coordenadas del punto P2
- **antialiasing:** Corrige pixelado de contornos
- **stroke color:** Color de la línea de contorno
- **fill color:** color del borde
- **stroke width:** anchura del punto



TEXTO

Esta función Canvas permite dibujar un texto en una determinada posición con un tipo de fuente y un color determinado. Los parámetros que pueden ser modificados son:

- **x:** Posición de la coordenada x
- **y:** Posición de la coordenada y
- **antialiasing:** Alisado de la imagen evitando el pixelado
- **font color:** Color de la letra
- **font:** Tipo de letra



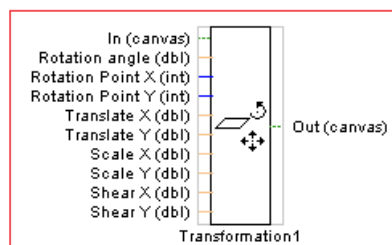
En la siguiente figura vemos el aspecto de como se presentaría el texto, que se introduce mediante el componente de entrada de texto que se conecta a la entrada “string” del componente “texto canvas”



TRANSFORMACIÓN

Esta es una poderosa función que permite manipular dentro de un Canvas uno cualquiera de los objetos que anteriormente hemos descrito. Las posibilidades de mover el objeto son: *Girarlo, trasladarlo de posición en los ejes x e y, modificar su escala en los ejes x e y, estirar en el eje x y en el eje y.*

Las entradas se corresponden a:



- **In:** Entrada del objeto canvas
- **Rotation angle:** Angulo de rotación.
- **Rotation point X:** Coordenada x del punto de rotación
- **Rotation point Y:** Coordenada y del punto de rotación
- **Scale X:** Valor de escalado en el eje x
- **Scale Y:** Valor de escalado en el eje y
- **Shear X:** Valor de estiramiento en el eje x
- **Shear Y:** Valor de estiramiento en el eje y

2. APLICACIONES DE LOS OBJETOS CANVAS

MyOpenLab es una poderosa herramienta que permite modelizar cualquier tipo de sistema o fenómeno físico. La utilización de una región Canvas en donde poder situar diversos objetos permite crear escenarios muy completos.

La idea básica consiste en hacer que las entradas X e Y de las posiciones de los objetos estén conectadas a las salidas de los modelos matemáticos que se puedan implementar de acuerdo con un fenómeno.

Veamos unos sencillos ejemplos.

MOVIMIENTO ACELERADO DE UN COCHE.

En este modelo vamos a simular el comportamiento de un coche que se mueve en línea recta sometido a una aceleración.

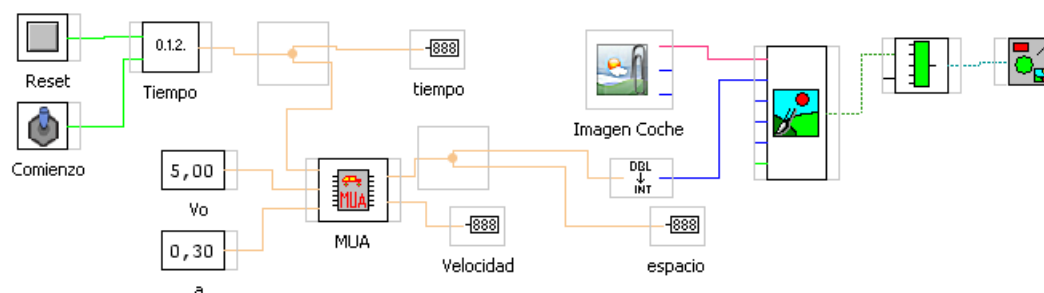
Variables dependientes del modelo:

Vo: Velocidad inicial
a: Aceleración
S: Espacio
Velocidad: Velocidad instantánea.

Variable independiente:

t: Tiempo

En la siguiente figura vemos el “Panel Circuito” de la simulación en donde aparecen los objetos funcionales que la integran.



La variable independiente vemos que procede de un bloque contador que viene a simular el paso del tiempo.

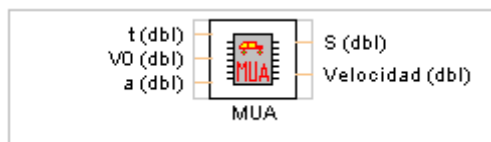
El objeto que se mueve es una imagen de un coche que es la que se entrega al correspondiente bloque “Canvas Imagen”. Vemos también que los valores de Vo y a se colocan fijos mediante un bloque de entrada de dato tipo Integer. Podríamos haberlos puesto como variables a modificar mediante una caja de entrada de texto de tal manera que se podrían cambiar los valores sin tener que salirnos del modo simulación.

El botón Reset sirve para poner a cero el valor del tiempo. El interruptor Comienzo sirve para iniciar la cuenta del tiempo.

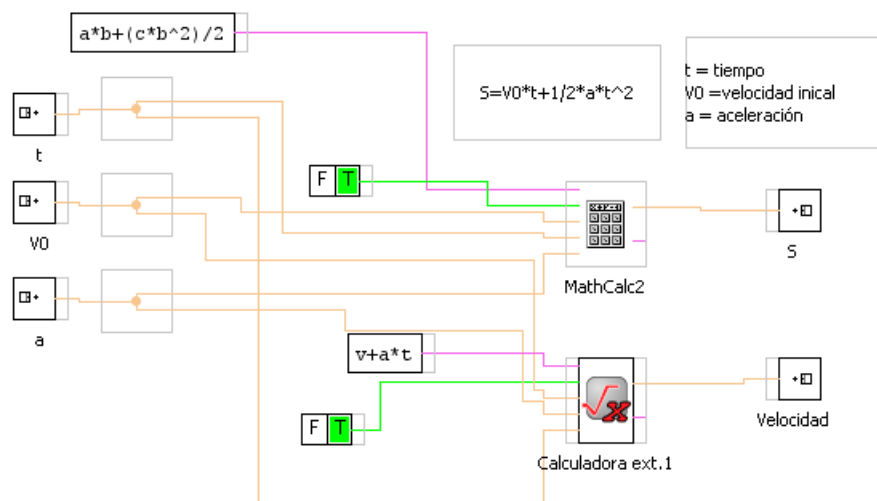
En la siguiente figura vemos el “Panel Visualización” de la simulación en el que se muestran los objetos gráficos con un fondo de pantalla (paisaje) que se ha colocado como “Imagen 2.0”



En la simulación del fenómeno se ha creado un bloque separado (encapsulado como fichero MUA.vlogic) que contiene el modelo matemático.



En el bloque vemos las variables de entrada (t , V_0 , a) y las de salida (S y Velocidad)



Este es el contenido del bloque MUA. Vemos que se han utilizado los bloques de cálculo **MathCalc2** y **Calculadora ext.1** que son los encargados de calcular el espacio recorrido y la velocidad instantánea respectivamente.

TIRO PARABOLICO: EL JUGADOR DE BASKET.

Con esta aplicación se trata de simular el lanzamiento de un objeto (balón) emulando el tiro parabólico.

Las ecuaciones de este movimiento son las correspondientes a un movimiento acelerado y un movimiento uniforme. En el sentido vertical el movimiento está afectado por la fuerza de la gravedad así como por la velocidad inicial del lanzamiento que dependerá a su vez del ángulo con el que se lanza el balón.

Variables de entrada:

t: El tiempo

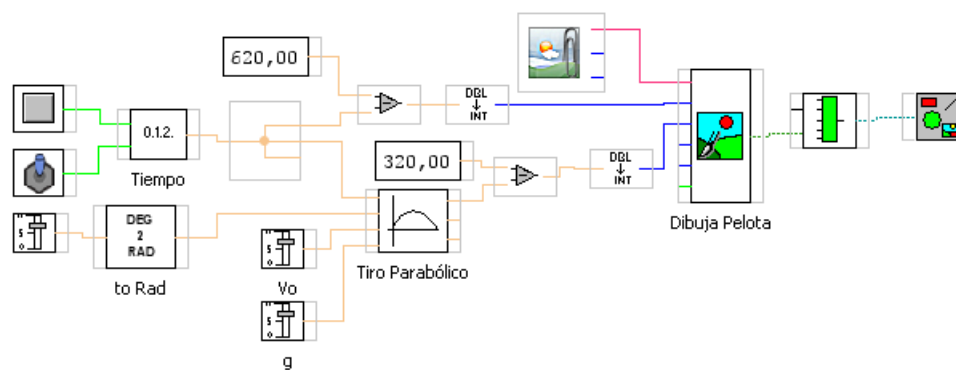
V_0 : Velocidad inicial

g: aceleración de la gravedad

Variables de salida:

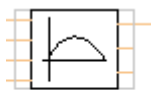
Posición y

Posición x (tiempo)



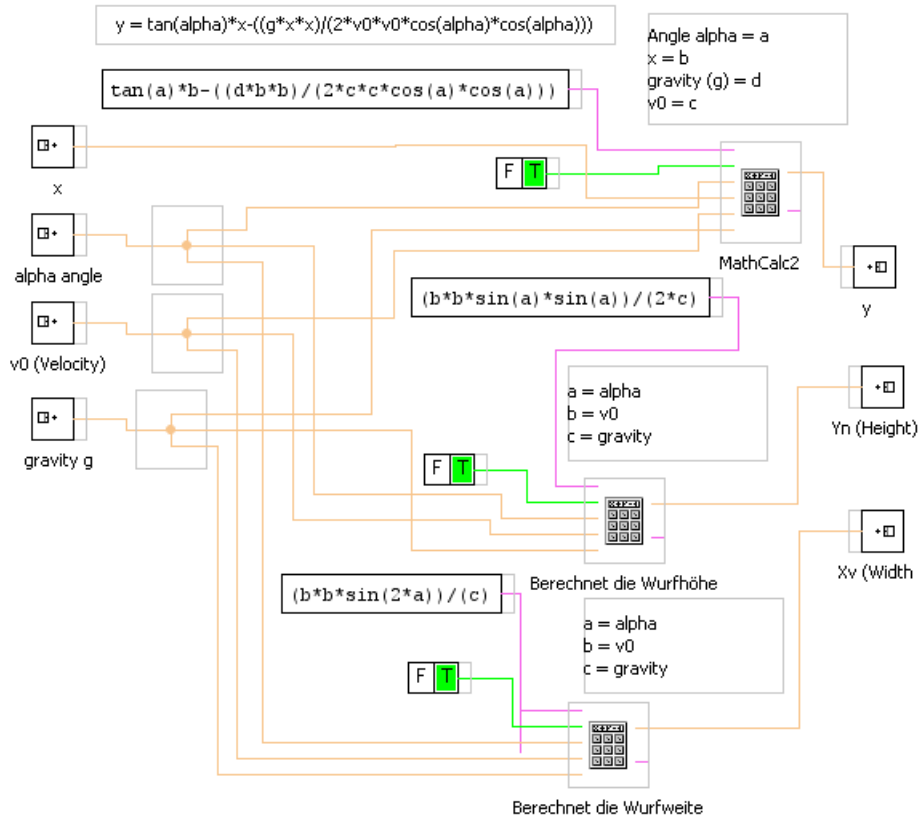
En la simulación se ha utilizado una Imagen Canvas que es la que representa el balón y que se mueve de acuerdo a las coordenadas x e y del modelo. Se han colocado así mismo dos imágenes (canasta y jugador) para completar el escenario de la simulación.

El modelo se centra en torno al bloque de librería ya disponible Tiro Parabólico



Tiro Parabólico

que implementa el modelo matemático de la simulación. En la figura vemos el contenido de este bloque.



Finalmente vemos la pantalla del panel en modo simulación.

A la hora de realizar la simulación se pueden variar de forma distinta los valores de las variables de entrada y estudiar el comportamiento de la salida (trayectoria de la pelota).

