



COMPUTADORES III

VIRTUAL CYBERTECH: USER MANUAL



UNIVERSIDAD POLITÉCNICA DE MADRID
**ESCUELA TÉCNICA SUPERIOR DE
INGENIEROS INDUSTRIALES**

EQUIPO YELLOW:

Marta Dorado (Project Manager), Álvaro López, Antonio Díez, Carlos Sampedro, Francisco Suárez, Marie Destarac y Ricardo Espinoza.

Índice

0. Seguimiento del documento	3
1. Introducción	4
2. Instalación del software	5
2.1. Requisitos mínimos	5
2.2. Instalación	5
3. Descripción de la interfaz de usuario	8
3.1. Ventana de Controles.....	8
3.2. Ventana de Simulación.....	11
3.3. Ventana de Código	12
3.4. Ventana de Estado del Robot.....	12
3.5. Ventana de Información.....	13
3.6. Barra de estado	15
4. Anexos	16

0. Seguimiento del documento

Histórico de modificaciones

Versión Número	Fecha <<mm/dd/aa>>	Cambios	Autor	Revisor	Secciones afectadas
1	06/11/12	Creación del documento	Equipo Yellow	Guadalupe Sánchez	Todas
2	06/22/12	Introducción del apartado instalación	Equipo Yellow	Guadalupe Sánchez	Instalación del sistema

1. Introducción

Este manual de usuario describe los detalles que debe conocer el usuario para poder hacer uso del software, y de todas sus utilidades y aplicaciones. El objetivo principal es dar soporte a los participantes de la competición *Cybertech* que se realiza anualmente entre grupos de distintas universidades y es organizada por la División de Ingeniería de Sistemas y Automática, y la asociación de estudiantes Reset de la Escuela Superior de Ingenieros Industriales de la Universidad Politécnica de Madrid, específicamente en el ámbito de programación y control, de manera que el usuario pueda hacer las pruebas necesarias con código de control de su robot sin necesidad del hardware respectivo y en un ambiente de fácil acceso.

En el siguiente apartado se presenta una guía básica para la instalación del software. Seguidamente, en el apartado 3 se realiza una descripción detallada de la apariencia del simulador, así como de sus múltiples funcionalidades. Para terminar, en el apartado 4, se incluye un pequeño tutorial que muestra el manejo del software.

2. Instalación del software

El objetivo de esta sección es proporcionar al usuario una guía básica que le permita realizar la instalación del software VCT, para su posterior uso, así como darle a conocer los requisitos mínimos que debe tener el sistema donde va a ser instalado. El binario del simulador así como el código fuente se pueden descargar en:

<http://www.romin.upm.es/fsuarez/YellowSimulator/index.html>

2.1. Requisitos mínimos

Para la instalación del software VCT, el sistema debe cumplir los siguientes requisitos de hardware y software:

- Hardware mínimo requerido:
CPU: >300 (Valoración CPU)
RAM: >=512 MB
Capacidad de almacenamiento: >50 MB
Tarjeta de vídeo >200 (Valoración G3D)
Resolución del monitor: >=1024x768
- Software requerido:
Sistema operativo *Microsoft Windows XP/Vista/7*

2.2. Instalación

La instalación del software VCT se realiza mediante un asistente, que va mostrando ventanas para guiar al usuario durante todo el proceso.

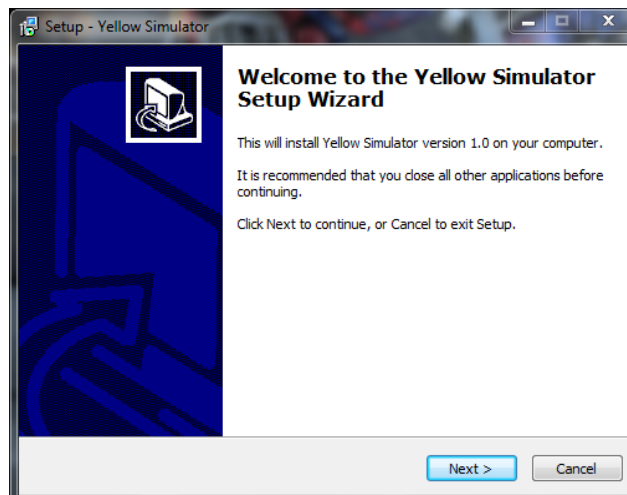


Figura 1. Ventana de instalación 1

Como primer paso, bastaría con ejecutar el instalador *Yellow Simulator.exe* con permisos de administrador. Lo primero que aparecerá, como podemos ver en la Figura 1, es una ventana dando la bienvenida al proceso de instalación.

Para comenzar con la instalación se deberá presionar “Next”.

Acto seguido, el instalable pedirá al usuario que indique el directorio de instalación (Figura 2). Se deberá pulsar “Next” para continuar con el proceso.

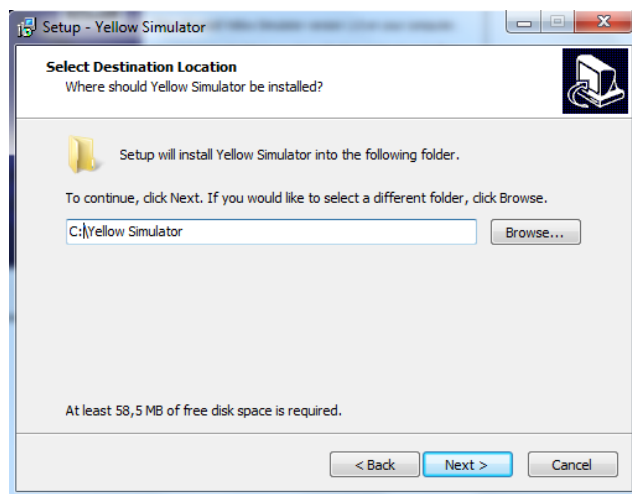


Figura 2. Ventana de instalación 2

Seguidamente, aparecerá una ventana indicando que el asistente está preparado para comenzar la instalación (Figura 3). La instalación comenzará tras pulsarse el botón “Install”.

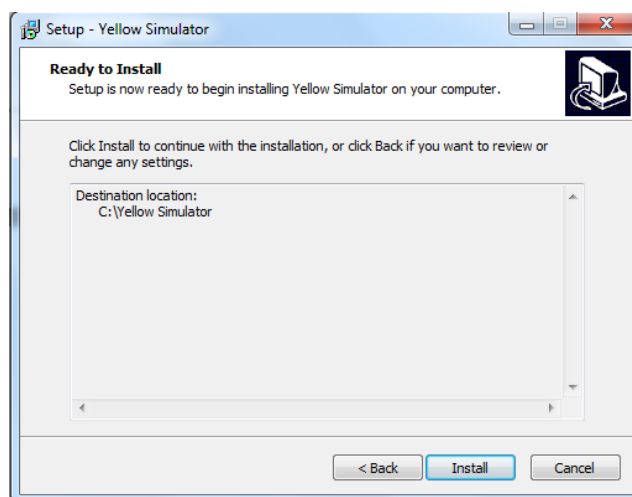


Figura 3. Ventana de instalación 3

Durante el proceso de instalación, el asistente irá mostrando las acciones que se están realizando, así como una barra que indica el progreso de la misma (Figura 4).

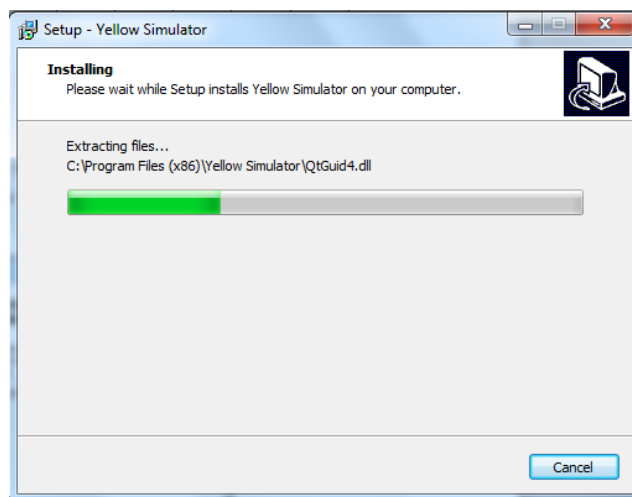


Figura 4. Ventana de instalación 4

Al finalizar, tal y como se puede ver en la figura 5, el instalable muestra una pantalla indicando que el proceso de instalación ha finalizado correctamente. Se deberá pulsar “Finish” para salir del asistente.

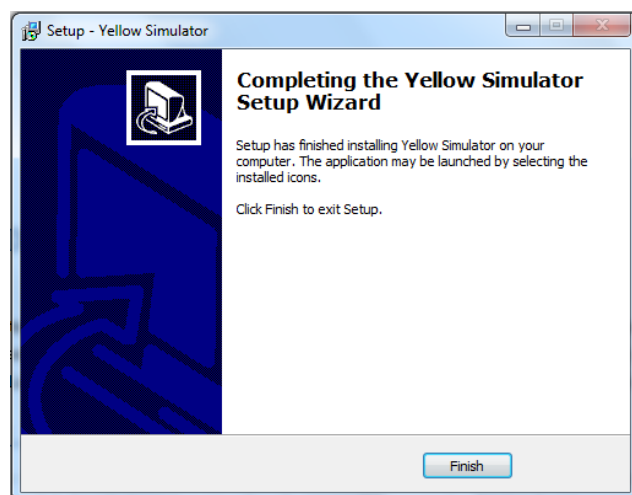


Figura 5. Ventana de instalación 5

Una vez finalizado correctamente todo el proceso, se habrán instalado en el equipo los componentes necesarios para el uso del software Yellow Simulator, creándose además un acceso directo en el escritorio para acceder de manera rápida y sencilla.

3. Descripción de la interfaz de usuario

La interfaz de usuario desarrollada con el software QT de Nokia contiene las funciones necesarias para que el usuario pueda realizar fácilmente la simulación de su código para el KIII. Ésta consta de seis ventanas que se describen a continuación:

3.1. Ventana de Controles

Es la ventana principal del software desde la cual se accede a la mayoría de las funcionalidades de la aplicación. En ella es posible controlar la simulación, generar los laberintos y guardar archivos, entre otras funciones. En la figura 6 se presenta la ventana de controles.

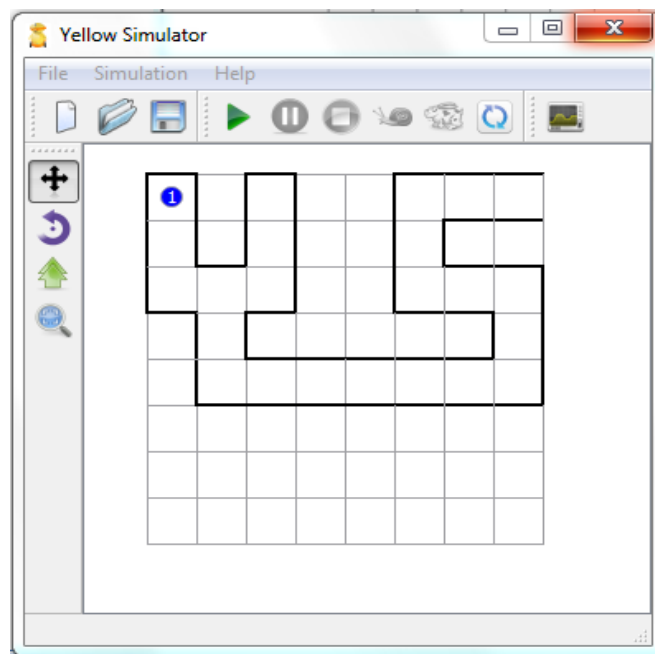
















Figura 6. Ventana de Controles

A continuación se describe en detalle la función de cada botón, desplazándonos por la barra de herramientas de izquierda a derecha y de arriba abajo, así como los distintos menús desplegables:

-  New: genera un nuevo laberinto. La generación de un nuevo laberinto implica su creación y para esto es necesario ir colocando las paredes según se prefiera en la retícula central de la ventana de controles. Cuando una nueva pared se haya colocado, dicha porción de la retícula se oscurecerá para su fácil detección.
-  Open: abre un laberinto creado previamente.

-  Save: guarda el laberinto, los datos o la simulación sobre el cual se está trabajando. Los datos pueden grabarse en tres formatos distintos: .txt, .csv (para su uso en Excel) o en .mat (para su uso en Matlab). El nombre y ubicación del archivo es seleccionable por parte del usuario a través de una pantalla de navegación. A continuación se describe cómo se almacenan los datos en cada formato:
 - **TXT:** Se crean tantas líneas como variables se vayan a guardar. El nombre de la variable se coloca al principio de la línea y los distintos valores se van guardando uno detrás del otro. El salto de línea indica el final de dicho vector.
 - **CSV:** similar al archivo .txt, pero con un formato que permite al programa EXCEL introducir cada dato en una celda y las distintas variable en filas.
 - **MAT:** Se crea un fichero con tantos variables como se quieran guardar. Estas variables serán vectores cuyo tamaño concuerda con el número de datos que se desea guardar. De esta forma se tendrán todos los datos en un fichero AAA.mat, en el que estarán guardadas n variables de longitud m.
-  Start: inicia la simulación.
-  Pause: detiene la simulación momentaneamente.
-  Stop: detiene por completo la simulación.
-  Slow Down Simulation: reduce la velocidad de la simulación.
-  Speed Up Simulation: aumenta la velocidad de la simulación.
-  Update Simulation: actualiza el estado del laberinto y la posición del robot en la simulación. Cada vez que se modifique una pared o la posición del robot en el laberinto será necesario presionar este botón para que se haga efectivo dicho cambio.
-  Graphs: da acceso a la ventana con las gráficas del estado del robot, como posición y velocidad. En la figura 13 se muestra dicha ventana.
-  Camera Pan: desplaza la vista de la ventana de simulación manteniendo presionado el botón izquierdo del ratón. El desplazamiento puede ser vertical, horizontal o en diagonal.
-  Camera Rotate: rota la vista de la ventana de simulación manteniendo presionado el botón izquierdo del ratón.

-  Camera Shift: acerca o aleja la vista (Zoom) de la ventana de simulación manteniendo presionado el botón izquierdo del ratón.
-  Fit to View: ajusta la vista de la simulación centrando el laberinto en la ventana. Este ajuste depende de la vista seleccionada.
- Menu File: permite acceder a New, Open, Save, Exit y User Login. Esta última función se implementa para detectar el tipo de usuario que esta trabajando sobre el simulador, que puede ser el administrador o usuario normal. En la figura 7 se muestra este desplegable y sus opciones.

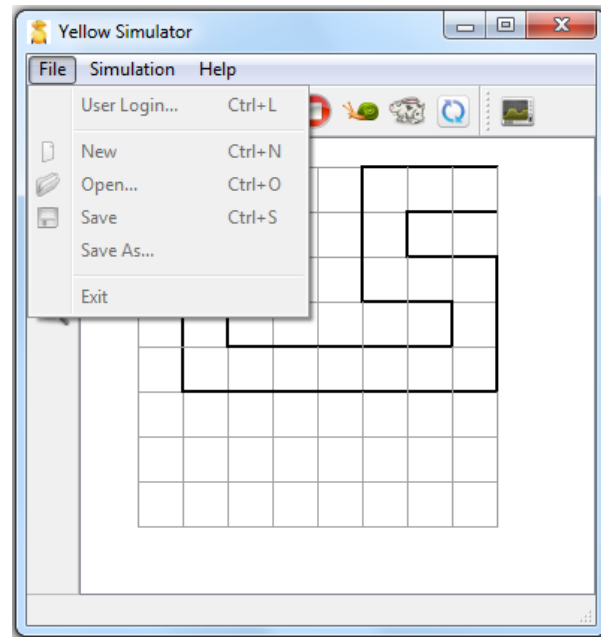


Figura 7. Detalle menú desplegable "File".

- Menu Simulation: permite acceder a Start, Pause, Stop, Slow Down Simulation, Speed Up Simulation y Update Simulation. Esta pestaña contiene los controles de la simulación. En la figura 8 se muestra este desplegable.

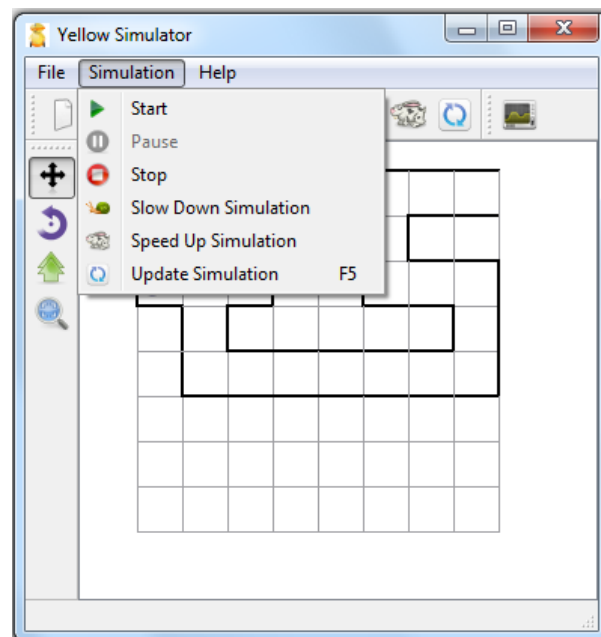


Figura 8. Detalle menú desplegable "Simulation".

- Menu Help: muestra información acerca de la aplicación y de Qt

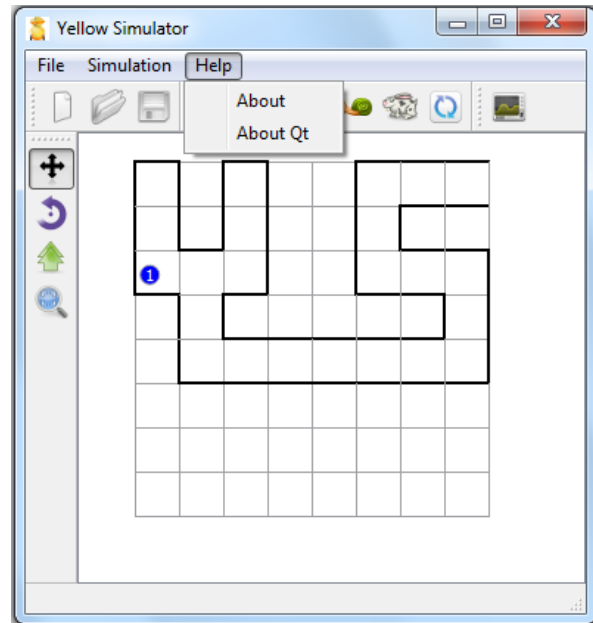


Figura 9. Detalle menú desplegable “Help”.

3.2. Ventana de Simulación

Permite visualizar el laberinto y la interacción del robot con éste durante la prueba del código introducido. En la figura 10 se muestra la apariencia de la ventana de aplicación, dentro de la cual aparece la ventana de controles y la de estado.

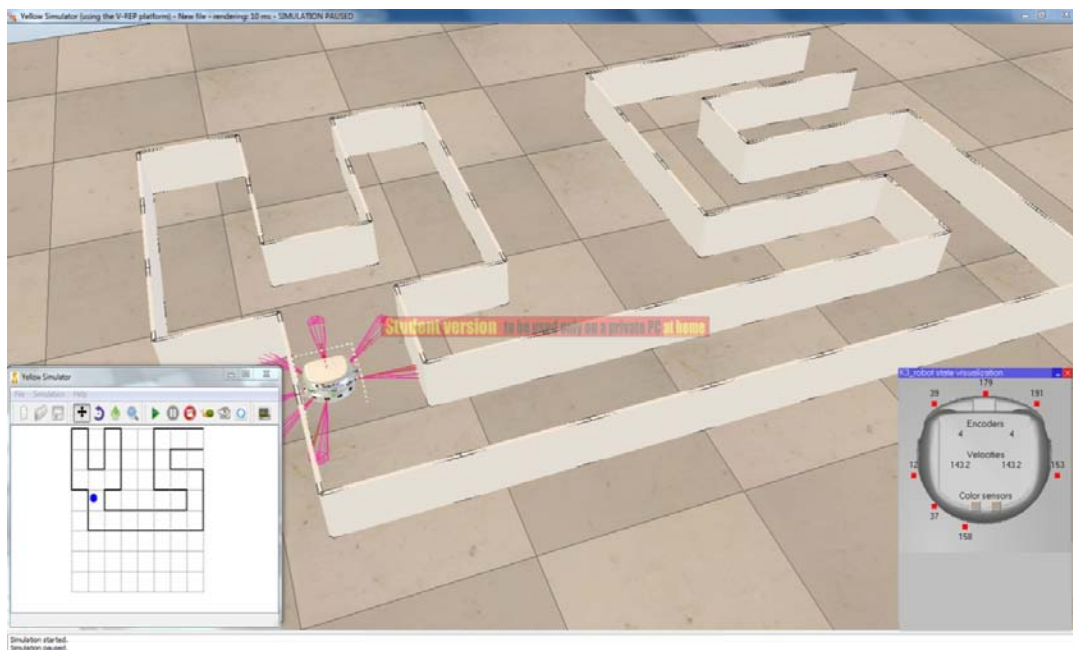
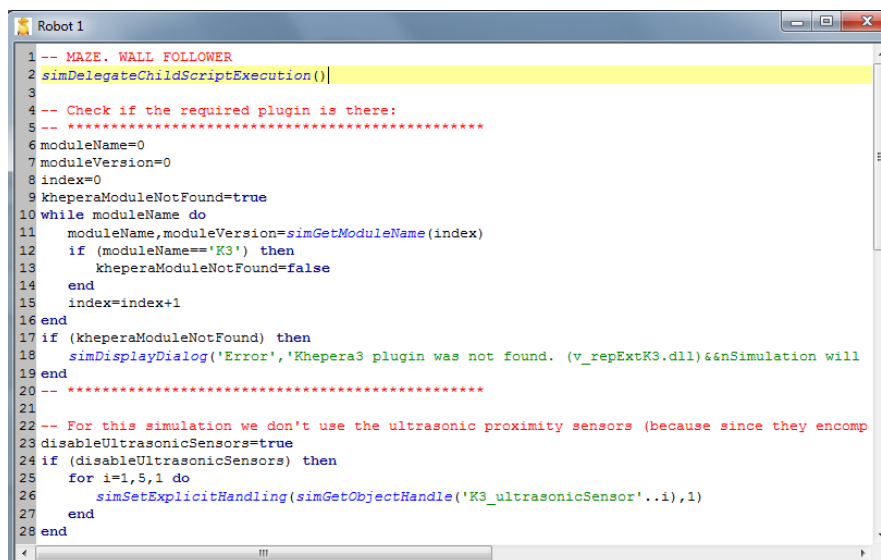


Figura 10. Ventana de simulación.

3.3. Ventana de Código

En esta ventana se introduce y edita el código de control a ejecutar en el robot durante la simulación. En la figura 11 se muestra el editor disponible para tal propósito.



```
1 -- MAZE. WALL FOLLOWER
2 simDelegateChildScriptExecution()
3
4 -- Check if the required plugin is there:
5 -- *****
6 moduleName=0
7 moduleVersion=0
8 index=0
9 kheperaModuleNotFound=true
10 while moduleName do
11   moduleName,moduleVersion=simGetModuleName(index)
12   if (moduleName=='K3') then
13     kheperaModuleNotFound=false
14   end
15   index=index+1
16 end
17 if (kheperaModuleNotFound) then
18   simDisplayDialog('Error','Khepera3 plugin was not found. (v_repExtK3.dll)& Simulation will
19 end
20 -- *****
21
22 -- For this simulation we don't use the ultrasonic proximity sensors (because since they encomp
23 disableUltrasonicSensors=true
24 if (disableUltrasonicSensors) then
25   for i=1,5,1 do
26     simSetExplicitHandling(simGetObjectHandle('K3_ultrasonicSensor'..i),1)
27   end
28 end
```

Figura 11. Ventana de código.

3.4. Ventana de Estado del Robot

Esta ventana muestra el estado de los sensores del KIII durante la realización de la simulación actualizándose en tiempo real. En la figura 12 puede observarse su apariencia.

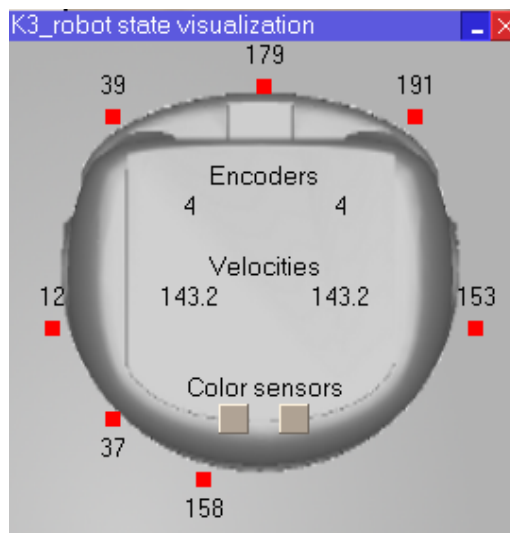


Figura 12. Ventana de Estado del Robot

3.5. Ventana de Información

En ella se puede observar gráficamente los valores de velocidad y posición del robot en tiempo real. Esta ventana puede verse en la figura 13.

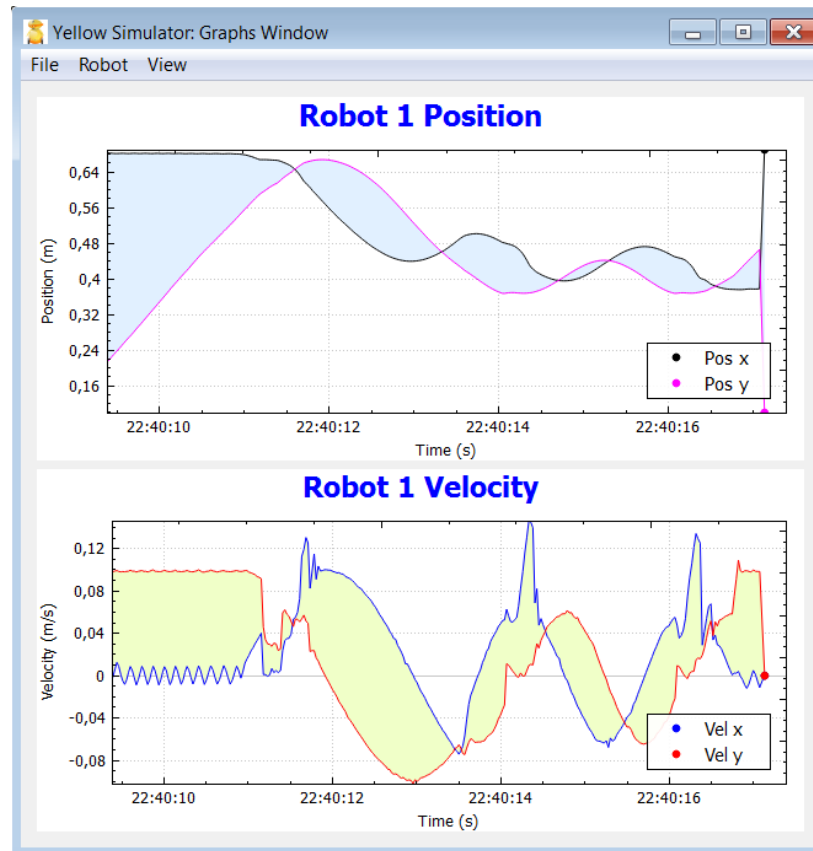


Figura 13. Ventana de información.

A continuación se describen las funcionalidades incluidas en la barra de menús:

- Menú File: desde este desplegable se pueden grabar los datos en diferentes formatos, como son .mat, .csv, .txt, .png y .pdf. Además de esto, se permite ocultar la ventana (Figura 14).
- Menú Robot: permite seleccionar el robot del que se quieren visualizar las gráficas (Figura 15).
- Menú View: desde este desplegable se permiten diferentes visualizaciones de las gráficas:
 - Show position: muestra únicamente la gráfica de posición respecto al tiempo.
 - Show velocity: muestra únicamente la gráfica de velocidad respecto al tiempo.
 - Show position and velocity: muestra las dos gráficas anteriores en conjunto.

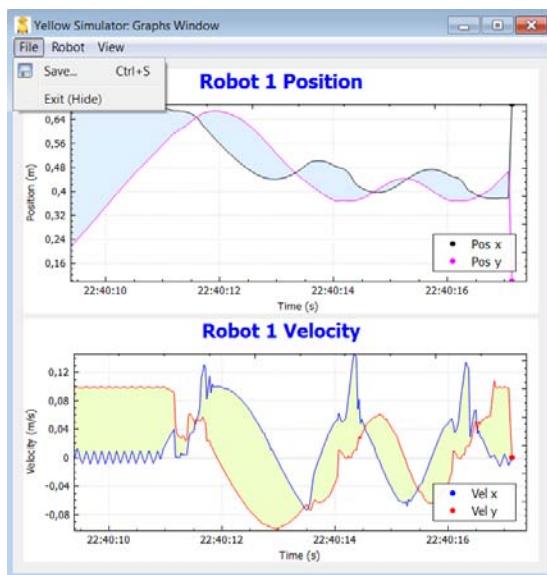


Figura 14. Detalle menú desplegable “File”.

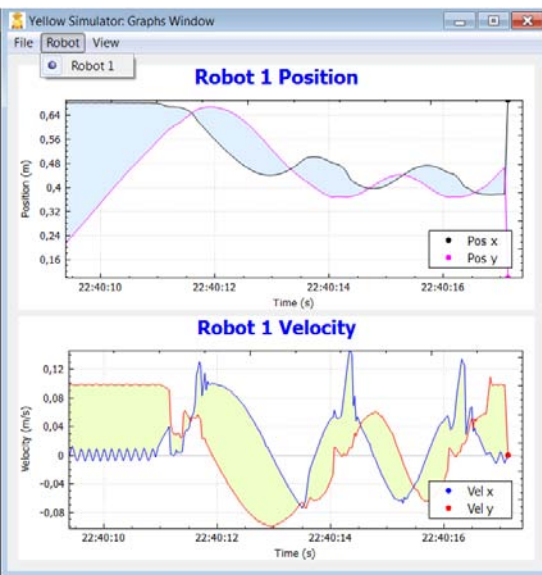


Figura 15. Detalle menú desplegable “Robot”.

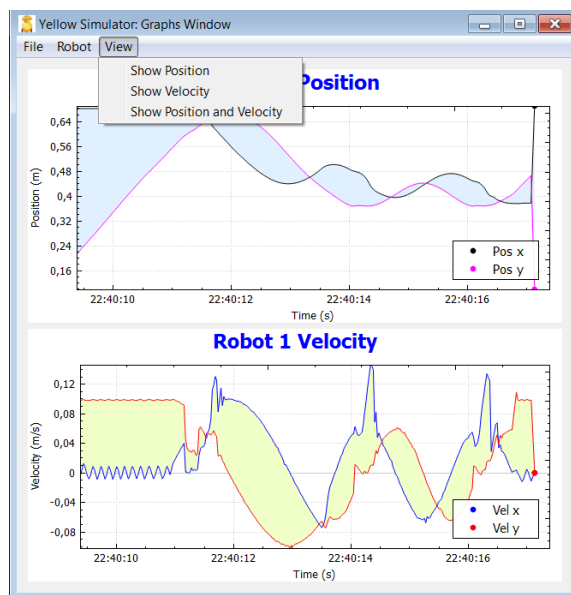


Figura 16. Detalle menú desplegable “View”.

3.6. Barra de estado

En ella se presentan los errores que se produzcan durante la compilación del código, así como el estado de la simulación. En la figura 17 se puede observar su ubicación dentro de la ventana de simulación y el detalle de ésta en la figura 18.

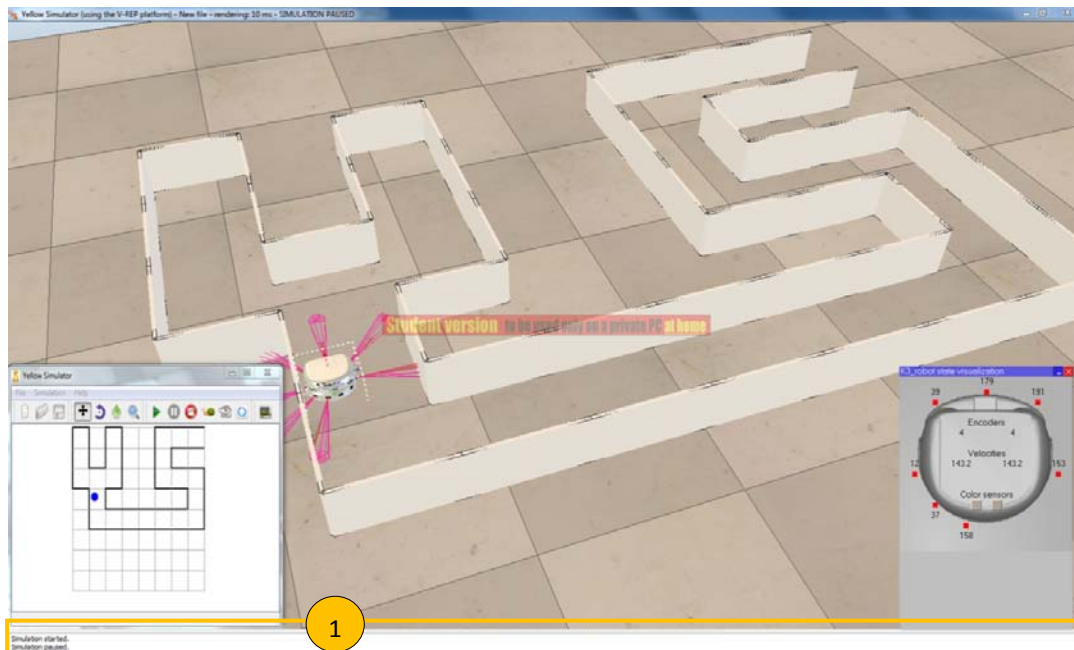


Figura 17. Ventana de simulación.



Figura 18. Detalle ventana de errores.

4. Anexos

En este apartado se adjunta un video tutorial como primera toma de contacto con el software, en el que se realiza una breve explicación a partir de una simulación de prueba.

Tutorial - YouTube