

# Step by step

## A quick guide to create your first IAR Visual State project

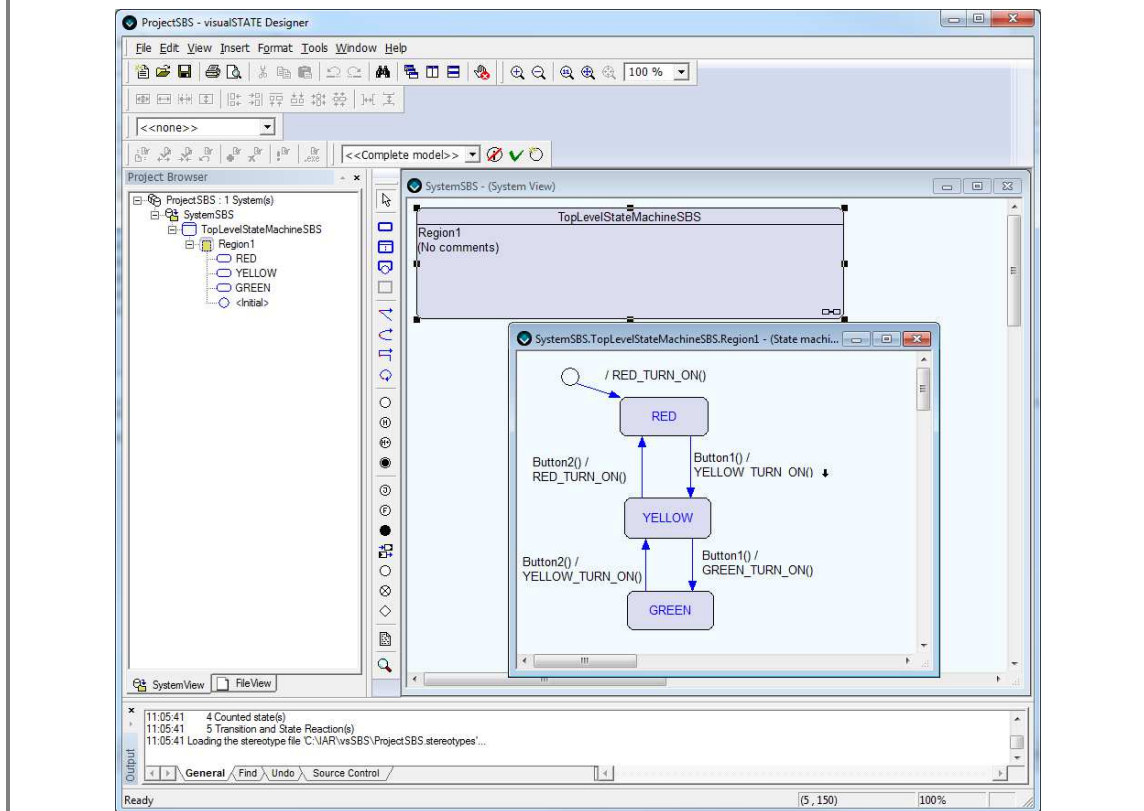
This guide briefly describes how to create your very first IAR Visual State project.

For more detailed information, see the *IAR Visual State User Guide* found under USER GUIDES in IAR Information Center.

### Step by step

This guide will take you through the following steps:

- Creating your project files in Navigator
- Using Designer to draw a few states and transitions
- Generating code
- Testing your design (validate and verify)
  - Validating the design
  - Verifying the design
- Documenting your code
- Adding variants to your design

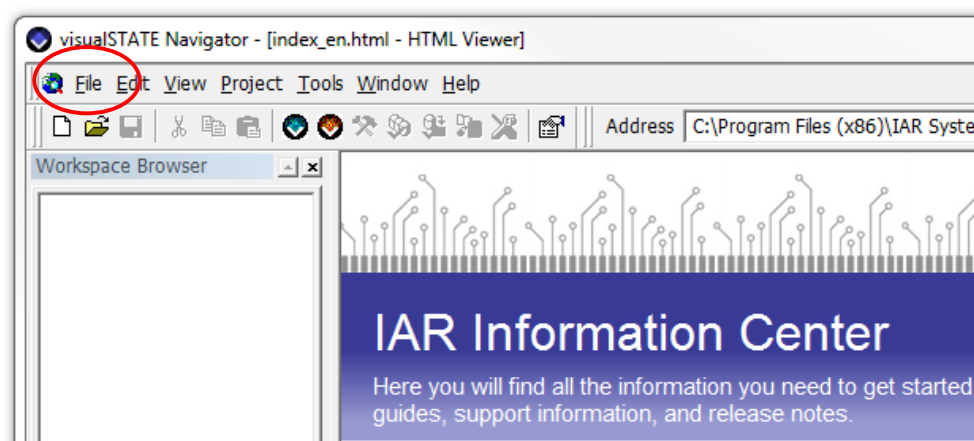


## Introduction

This step-by-step guide will give you an example on how to create your very first Visual State state machine. For this exercise we will create a traffic light.

## Creating your project files in Navigator

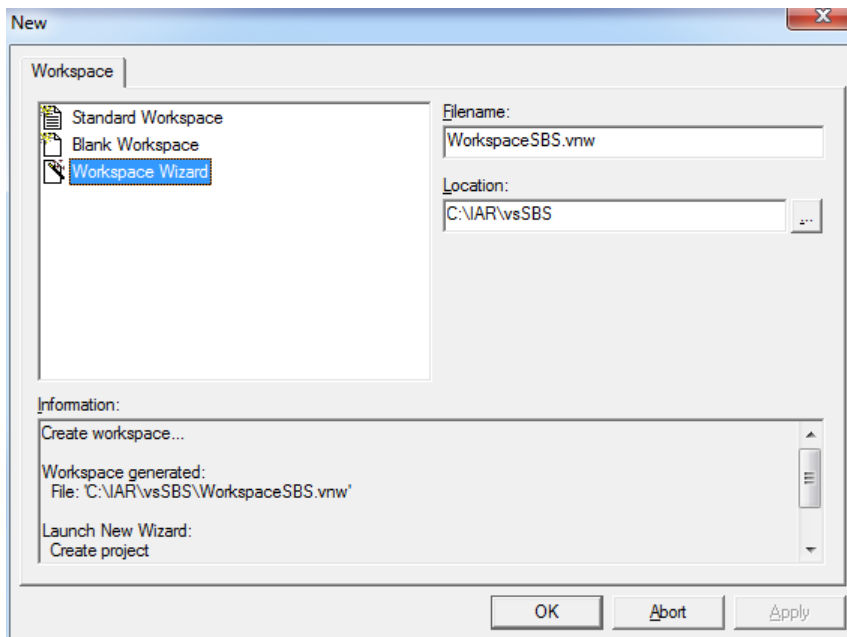
- 1 Open **IAR Visual State Navigator** found in the start menu in **All Programs> IAR Systems>IAR Visual State x.x**. Navigator is a graphical project management tool from where you deal with the overall handling of the projects and can access and activate the other components in Visual State.
- 2 Three windows will be seen: The Workspace browser window where you will find your projects later, the IAR Information Center (HTML) window from where you can reach example applications and guides and the Output window where information about your loaded workspace will be displayed
- 3 On the top menu open **File>New**.



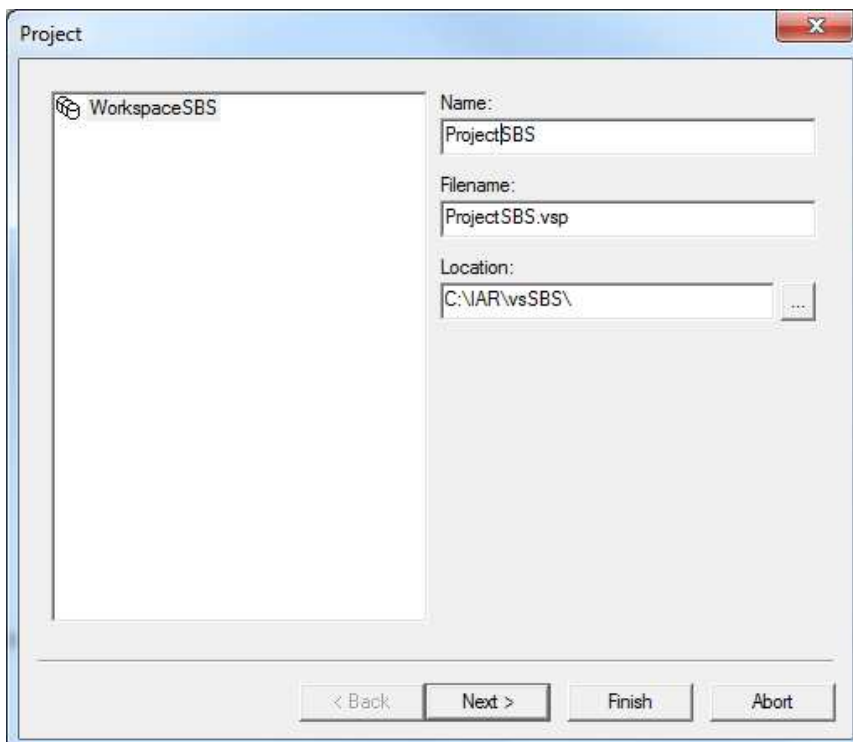
- 4 A window called New will let you name and create the workspace and files associated. You will get three choices:
  - **Standard Workspace** which creates a workspace with one project and one system with one top level state machine. You will be able to name your project but the associated files will be named with default names.
  - **Blank Workspace** which creates a blank workspace with no associated files. This is mostly used when you have an existing project and want to add it to a workspace.
  - The **Workspace Wizard** which will guide you through a complete setup where you can decide the number of top-level state machines of your system and also what to name the files.

In this exercise we will use the Workspace Wizard as we want to name the files separately.

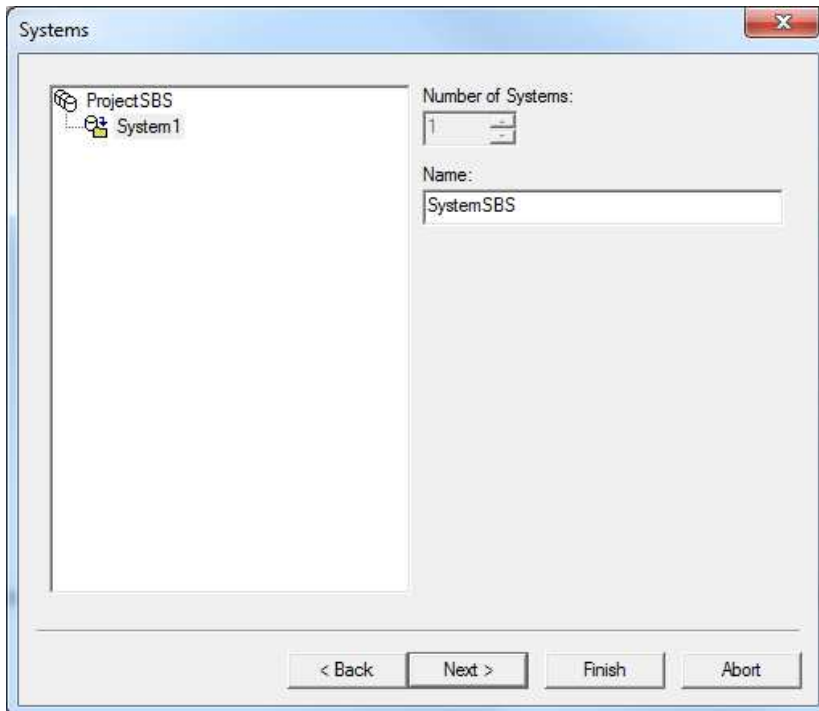
- 5 Choose Workspace Wizard and change the File name to WorkspaceSBS.vnw. Change the location to a new folder somewhere on your computer called vsSBS . Click **OK**.



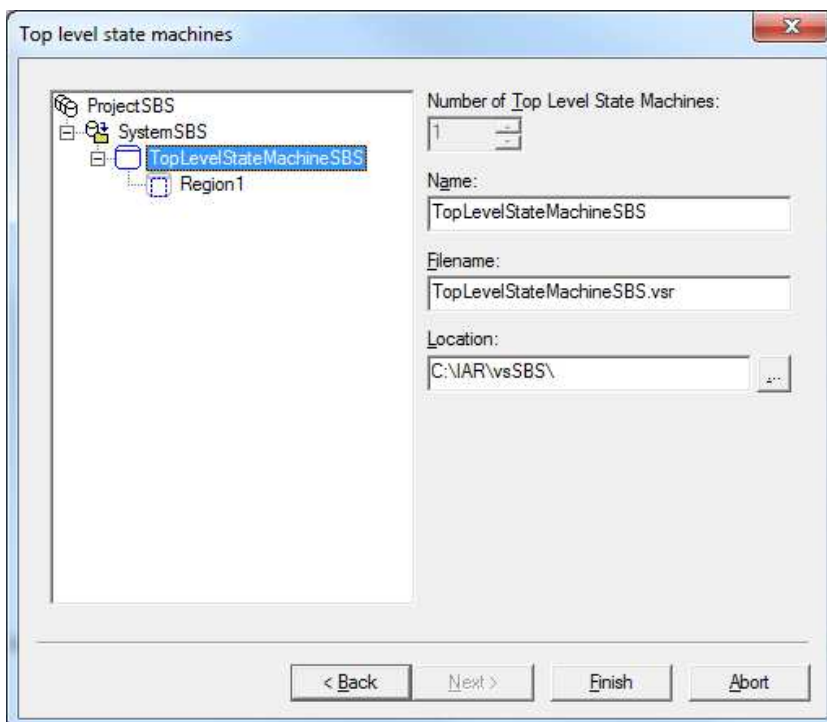
- 6 A window called Project will let you name the project name. Change the name to ProjectSBS. Leave the location folder as is and click **Next**.



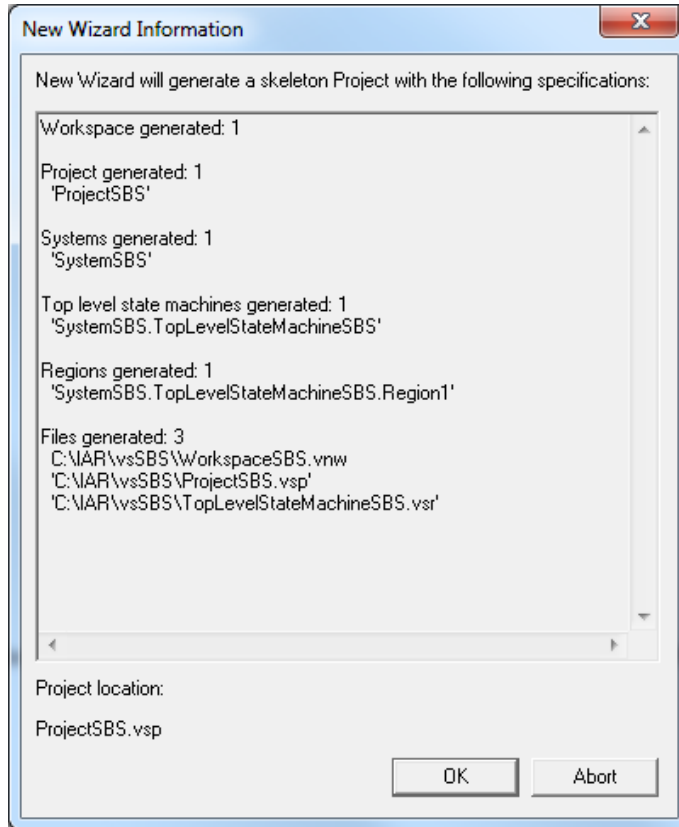
- 7 A window called Systems is appearing. This will let you decide if you want to have more than one system in your project. In this case we will use one. Mark System1 and rename it to SystemSBS. Click **Next**.



- 8 A window called Top level state machines will appear. Here you can change name, location and number of top level state machines. This is the topmost state machine in a state hierarchy. The names should be as below. Click **Finish**.



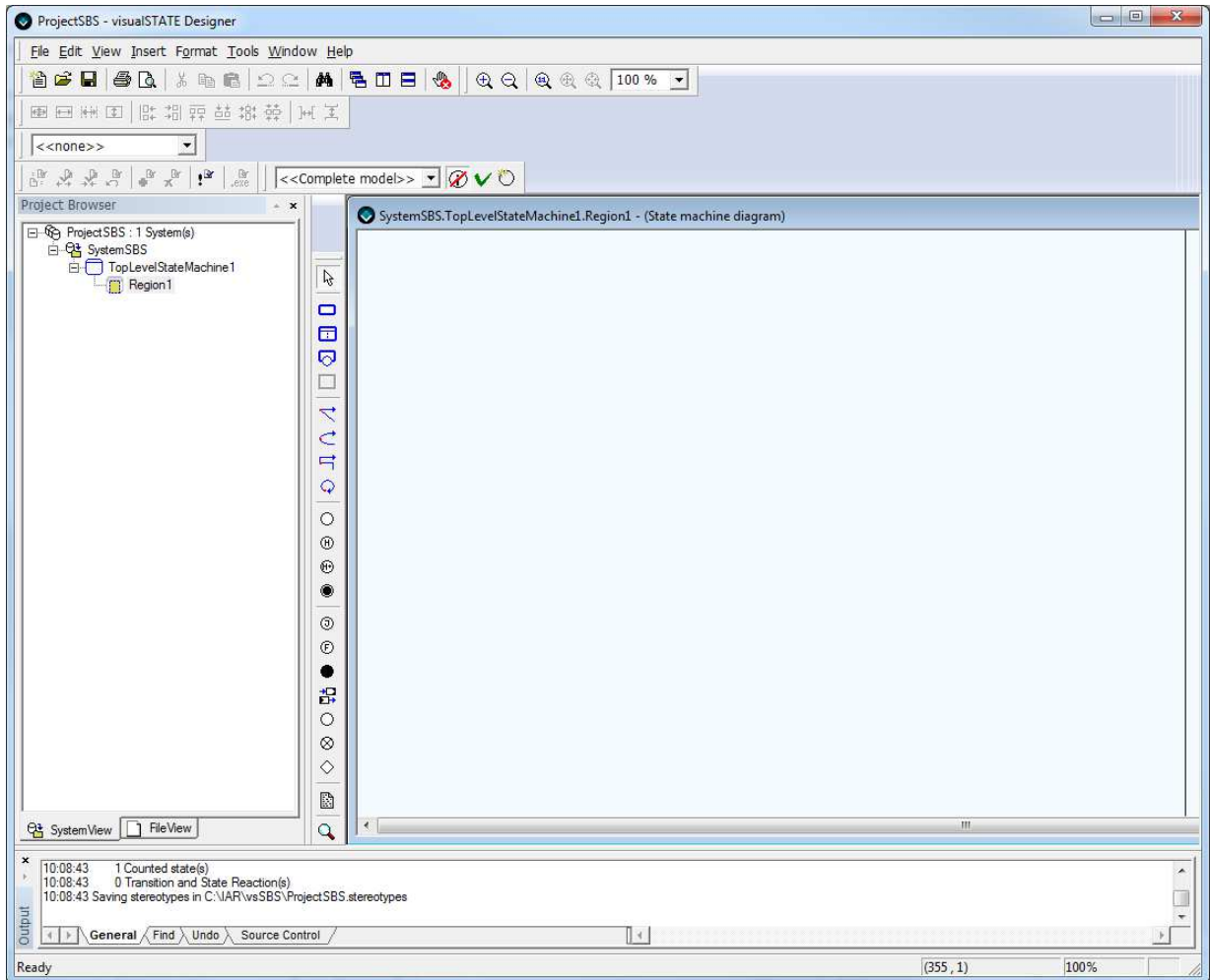
- 9 The New Wizard information window will open up and display all the information about the skeleton project created. Click **OK**.



- 10 The workspace wizard will now save the files for you, and automatically open the Designer component.

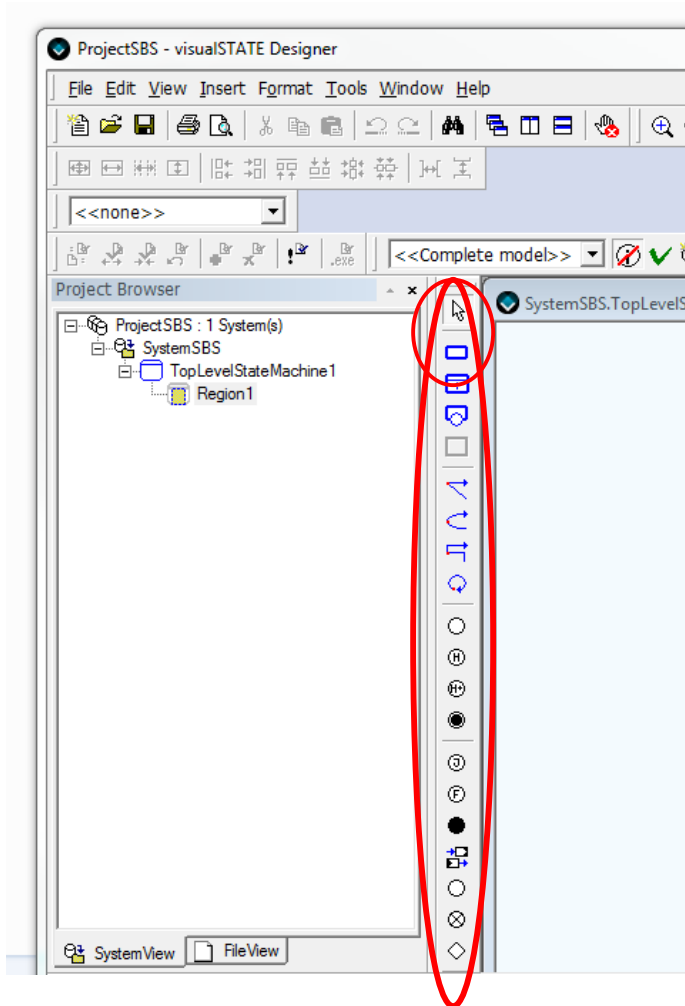
## Using Designer to create a few states and transitions

You should now have the Designer component open and a view like this:



It is time to draw your design.

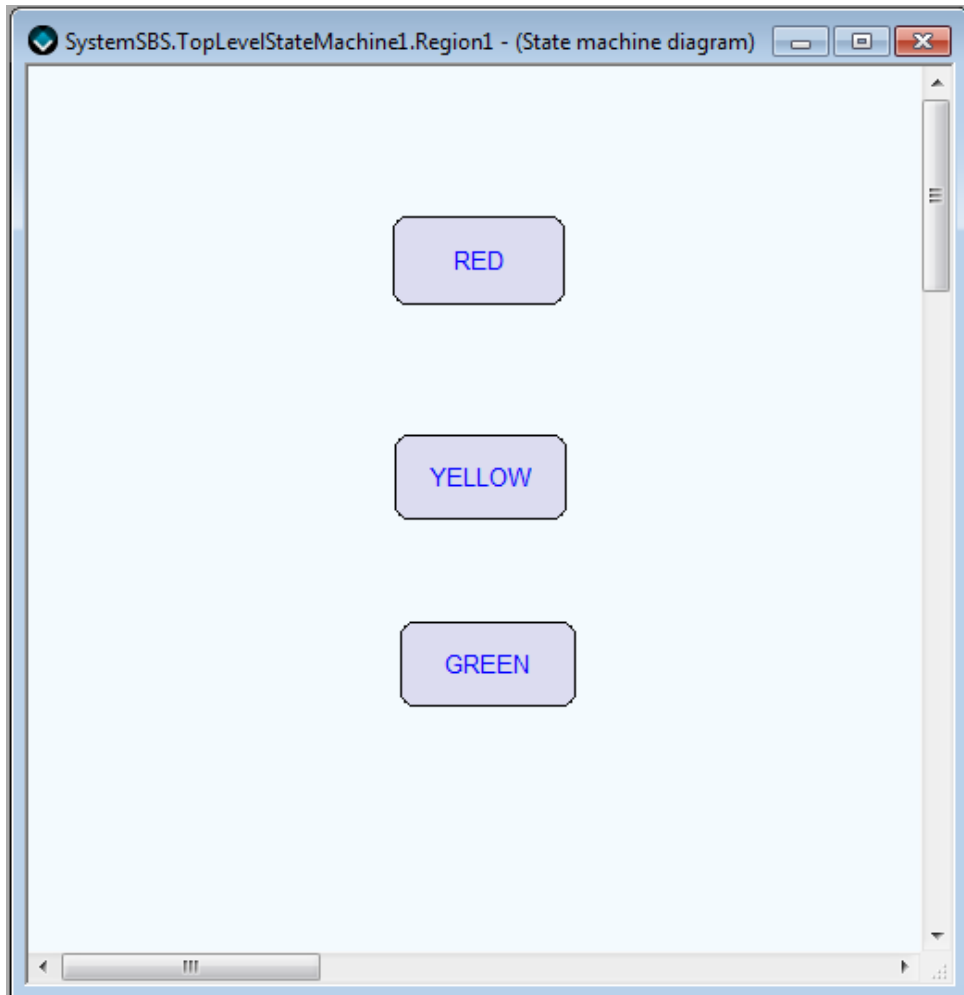
- 1 Double click on Region1 area in the System View window and the Region1 will open in a new State machine diagram window. This is where you will start to draw. Marked below is the toolbar which is also found under **Insert**.



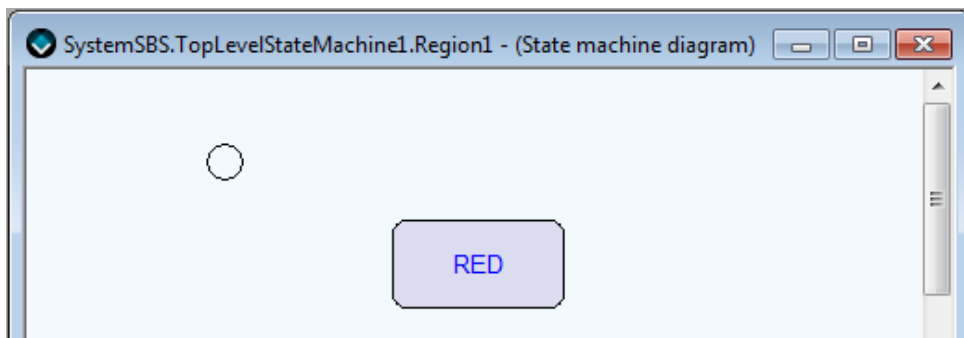
Insert	Format	Tools	Window	Help
	System			Ctrl+1
	Simple State			Ctrl+2
	Composite State			Ctrl+Skift+2
	Submachine State			
<hr/>				
	Transition			Ctrl+3
	Curved Transition			Ctrl+Alt+3
	Orthogonal Transition			Ctrl+Skift+3
	Self Transition			Ctrl+4
<hr/>				
	Initial State			Ctrl+5
	History State			Ctrl+Alt+5
	Deep History State			Ctrl+Skift+5
	Final State			Ctrl+6
<hr/>				
	Join			Ctrl+7
	Fork			Ctrl+Alt+7
	Junction			Ctrl+Skift+7
	Connector			Ctrl+8
	EntryPoint			
	ExitPoint			
	Choice			
<hr/>				
	Note			Ctrl+9

- 2 The first thing we want to draw is three states symbolising the traffic lights. States are the resting place between events. In this example the system will stay in red, green or yellow light until something happens - like a pedestrian pushing a button or a timer is going off.

Choose the Simple State in the toolbar and click on the drawing area once for each state (red, yellow, green). Rename the states to RED, YELLOW and GREEN by choosing the Select (arrow) in toolbar and click on the name in the state twice.

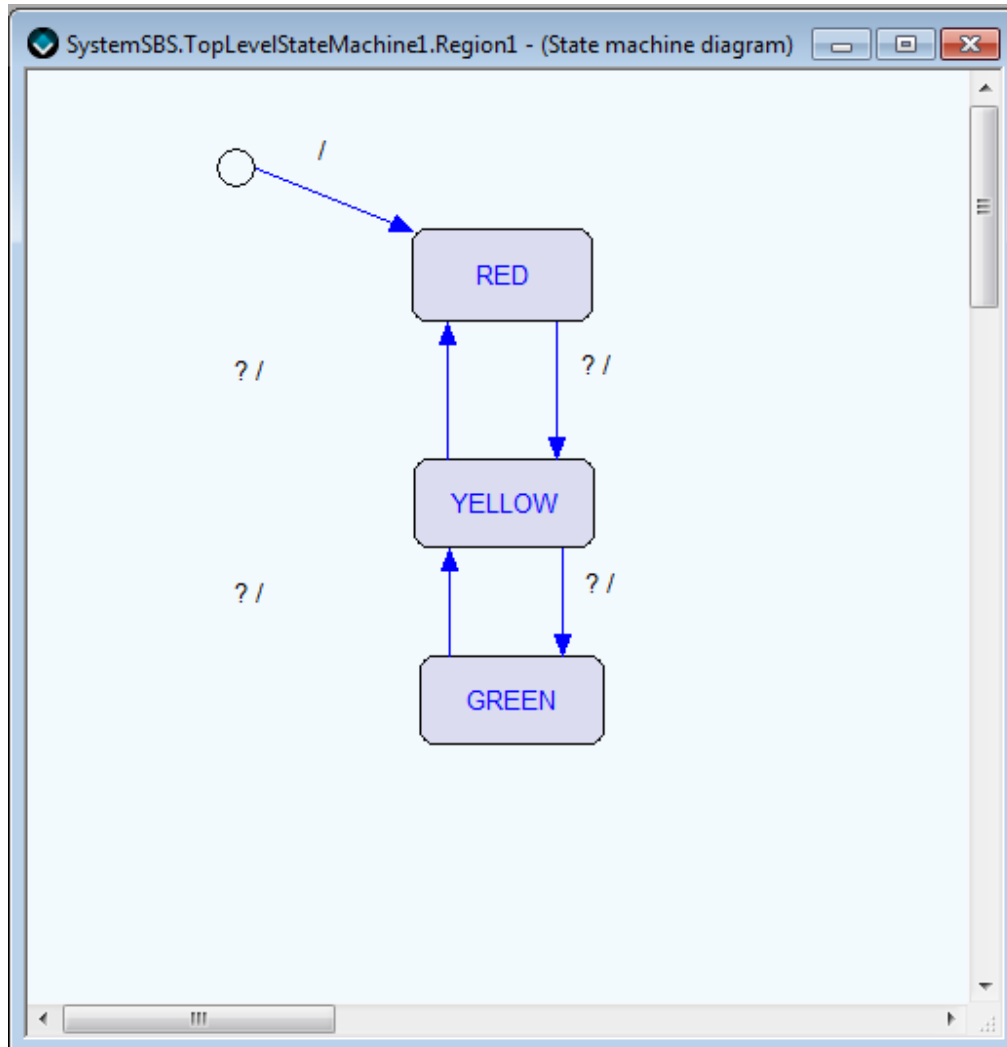


- 3 All systems need a starting point and for this we have to add an Initial State. Add that next to the RED state.



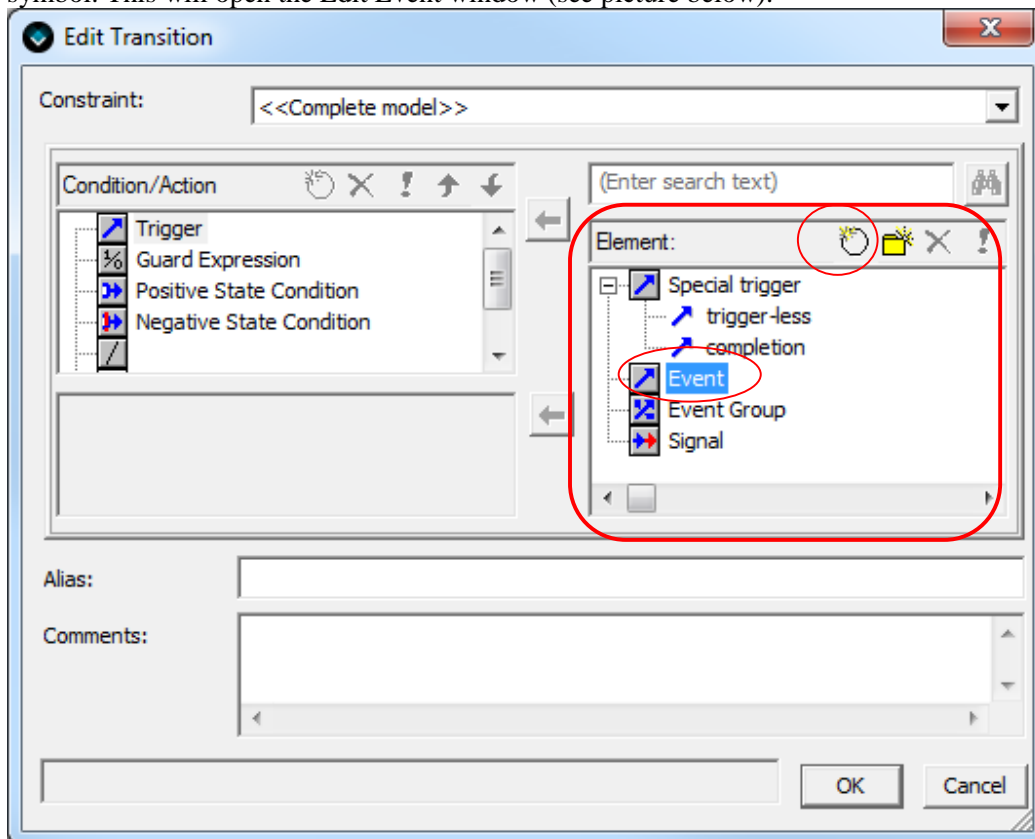


- 4 It is now time to add the transitions. Select one of the transitions from the toolbar according to your taste. Draw a transition by clicking on the source state to the target state. Create transitions from RED to YELLOW, from YELLOW to GREEN and vice versa. Also draw a transition from the initial state to the RED state. Your design should now look something like this:

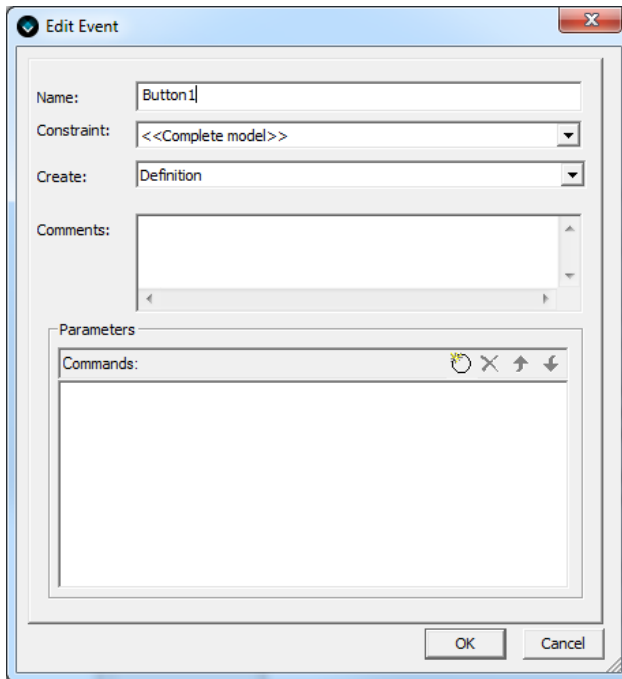


You can think of the “?/” as an “if/then” where if all conditions on the left side are true, the actions to the right of the “/” will be taken. The areas of “?/” can, and should be moved to easily see to which transition they belong. Remember to change to the arrow on the toolbar if you still have the transition tool marked. If you mark a transition arrow the corresponding “?/” area will be marked.

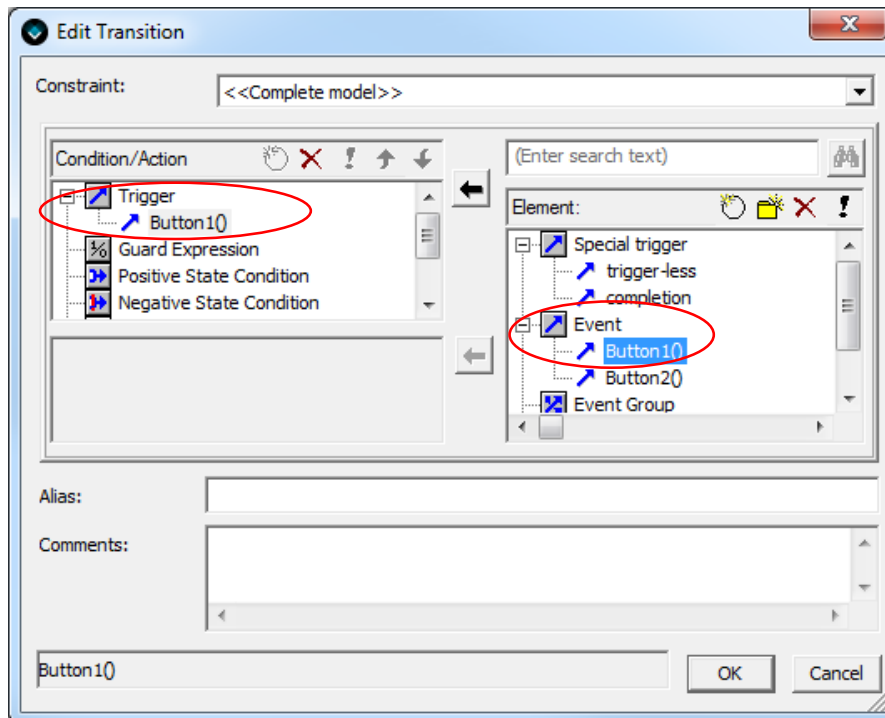
- 5 Double click on the transition area “?” from RED to YELLOW state to open up the Edit Transition window. Start by changing the **Element** to **Event** and then click on the Create New symbol. This will open the Edit Event window (see picture below).



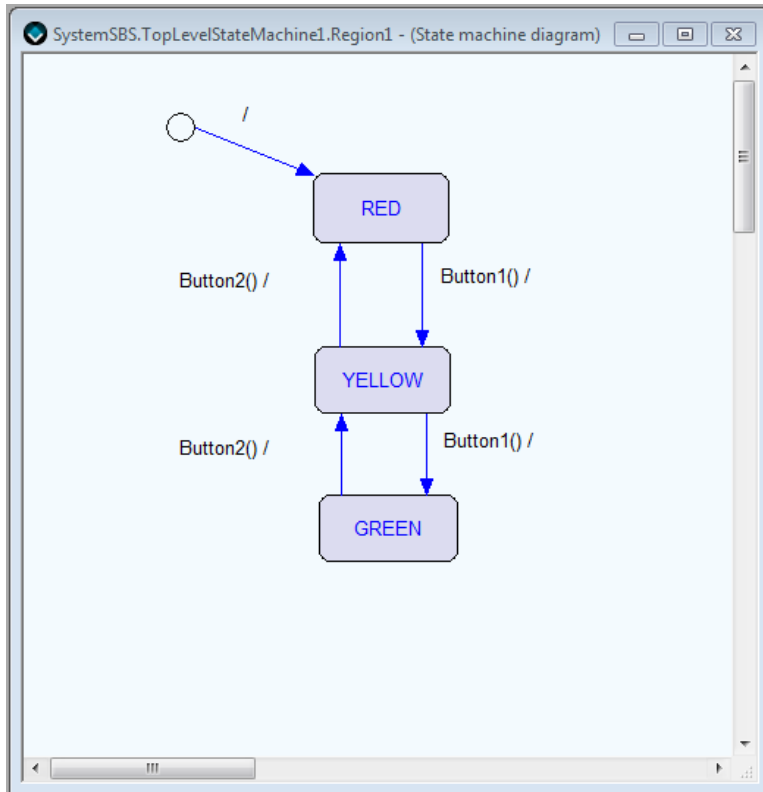
- 6 Create an event called Button1 and click OK. Repeat and create the event Button2.



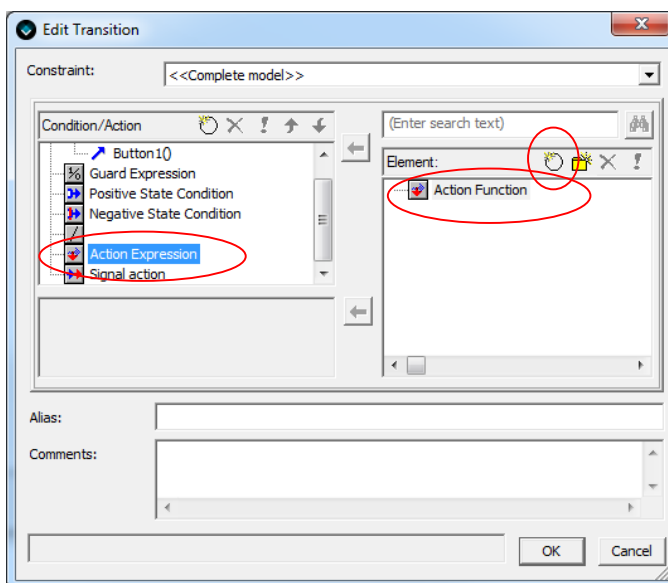
- 7 Now it is time to assign an event to the transition. Make sure that **Trigger** is marked in the Condition/Action list, and then double click on the **Button1** in the Element list. This will add Button1 as a trigger. Click **OK**.

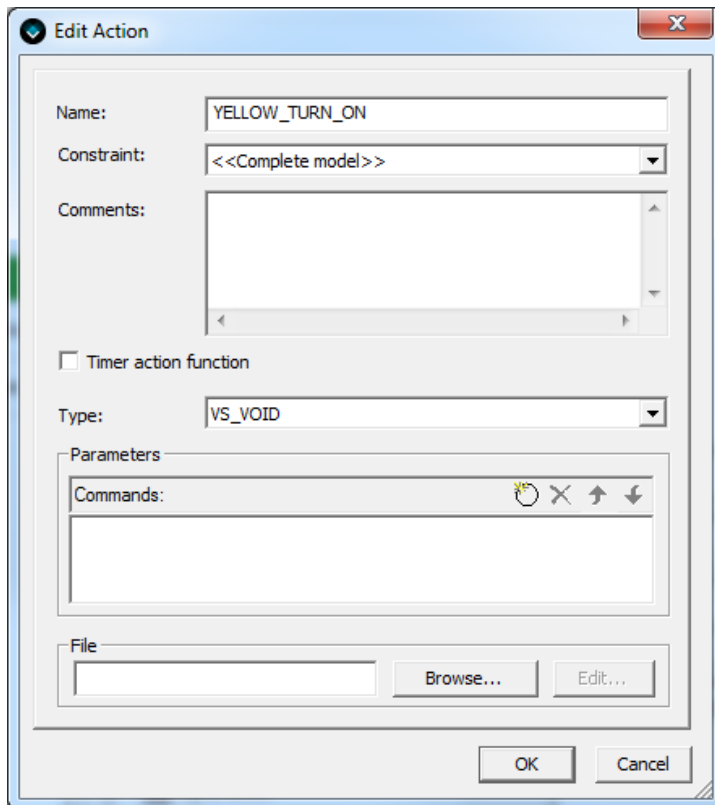


- 8 Go back to the State machine diagram window where you have been drawing your design. Double click on the transition from state YELLOW to GREEN and add the Button1 event to that transition as well. Repeat for the transitions from GREEN to YELLOW and YELLOW to RED, but add the Button2 event. Your design should now look something like this:

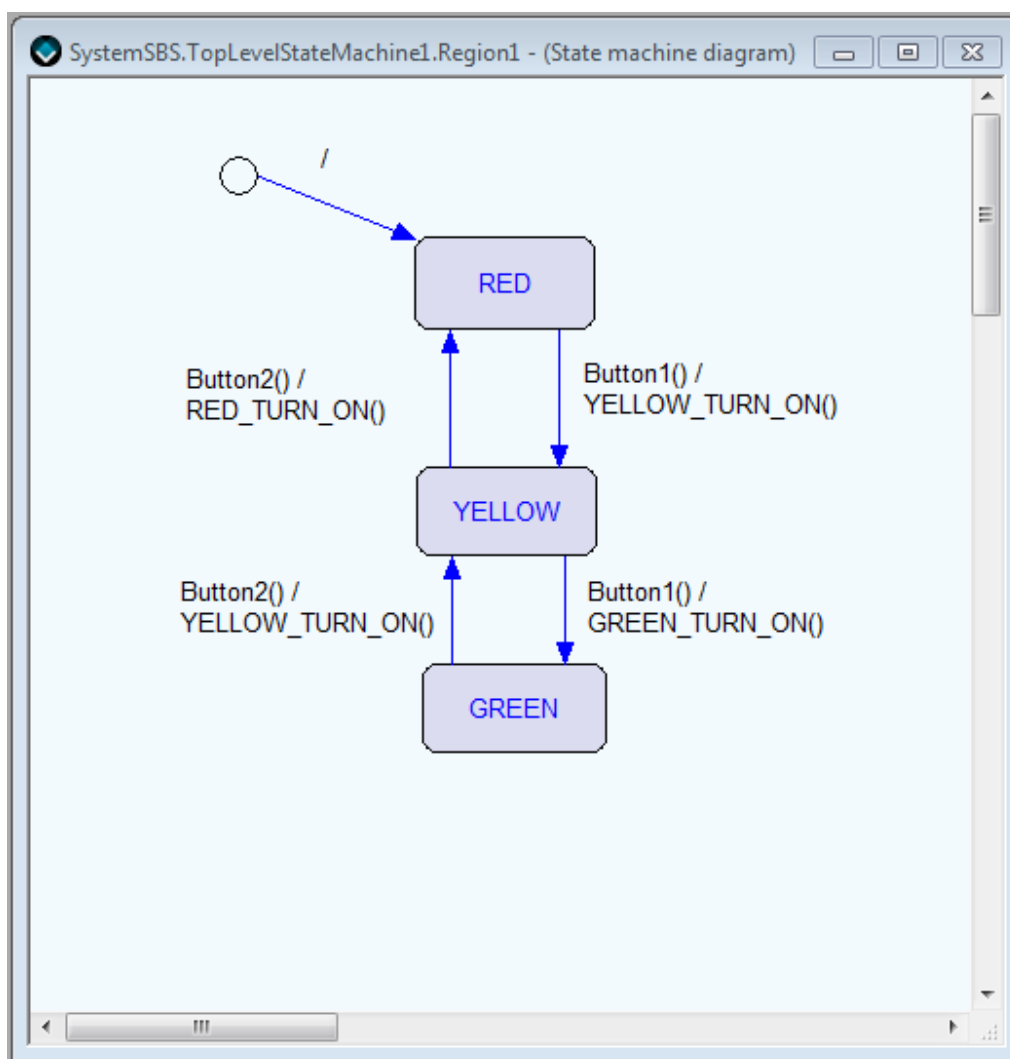


- 9 It is time to define actions. Click on the transition from RED to YELLOW again to open up the Edit Transition. This time mark the **Action Expression** in the Condition/Action list and the **Action Function** in the Element list and click on the New symbol to open up the Edit Action window. Create the action YELLOW\_TURN\_ON and click OK.

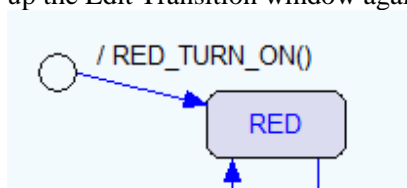




- 10 Repeat and create the actions RED\_TURN\_ON and GREEN\_TURN\_ON as well.
- 11 It is now time to add the actions to the transitions. Click on the transition arrow to open up the Compose Transition, in the Condition/Action list, mark the Action expression, and then in the Element list double click on the action you want to add to that specific transition. In this case we want the YELLOW\_TURN\_ON to be in the transitions leading to the YELLOW state. The RED\_TURN\_ON on the transition to the RED state, and the GREEN\_TURN\_ON on the transition to the GREEN state.

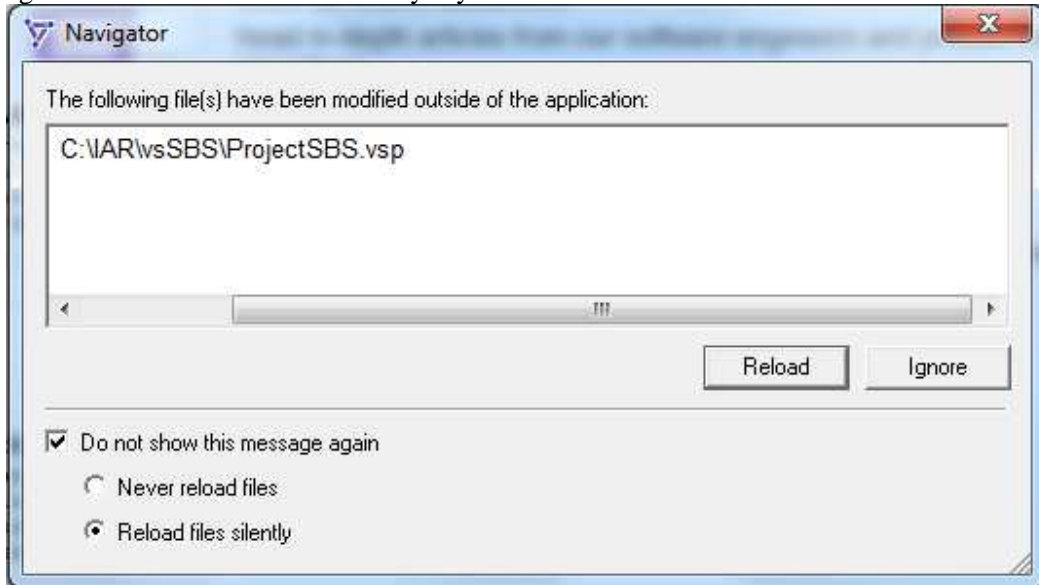


- 12** The final step here is to add the initial state transition. Double click on the transition line to open up the Edit Transition window again. Add the action RED\_ON to the transition.



**Note:** Double clicking the transition is one way to start to edit your transitions. If you want to add several events or actions at once, you can open the **View->Element Browser** and add the new events and actions from there as well.


- 13 It is now time to save this project (File->Save Project) and exit the Designer.
- 14 Click **Yes** when prompted to reload files in the Navigator. Check the Do not show this message again and select Reload files silently if you wish to hide the reload details in the future.



- 15 Congratulations, you have now designed your first state machine with IAR Visual State.

## Generating code

It is now time to generate the code. At this point you should see your WorkspaceSBS in the Workspace Browser window to the left in Navigator.


- 1 From the Navigator, press **F9**, or **Project>Code Generate** from the menu, or  on the toolbar.
- 2 A window with all the information on your project and its settings will appear. At the bottom of this window you should have a report of no warnings or errors when generating the code for the model.
- 3 The code generated will need to be integrated into your code. Information on how to do this is available in the IAR Visual State user guide.

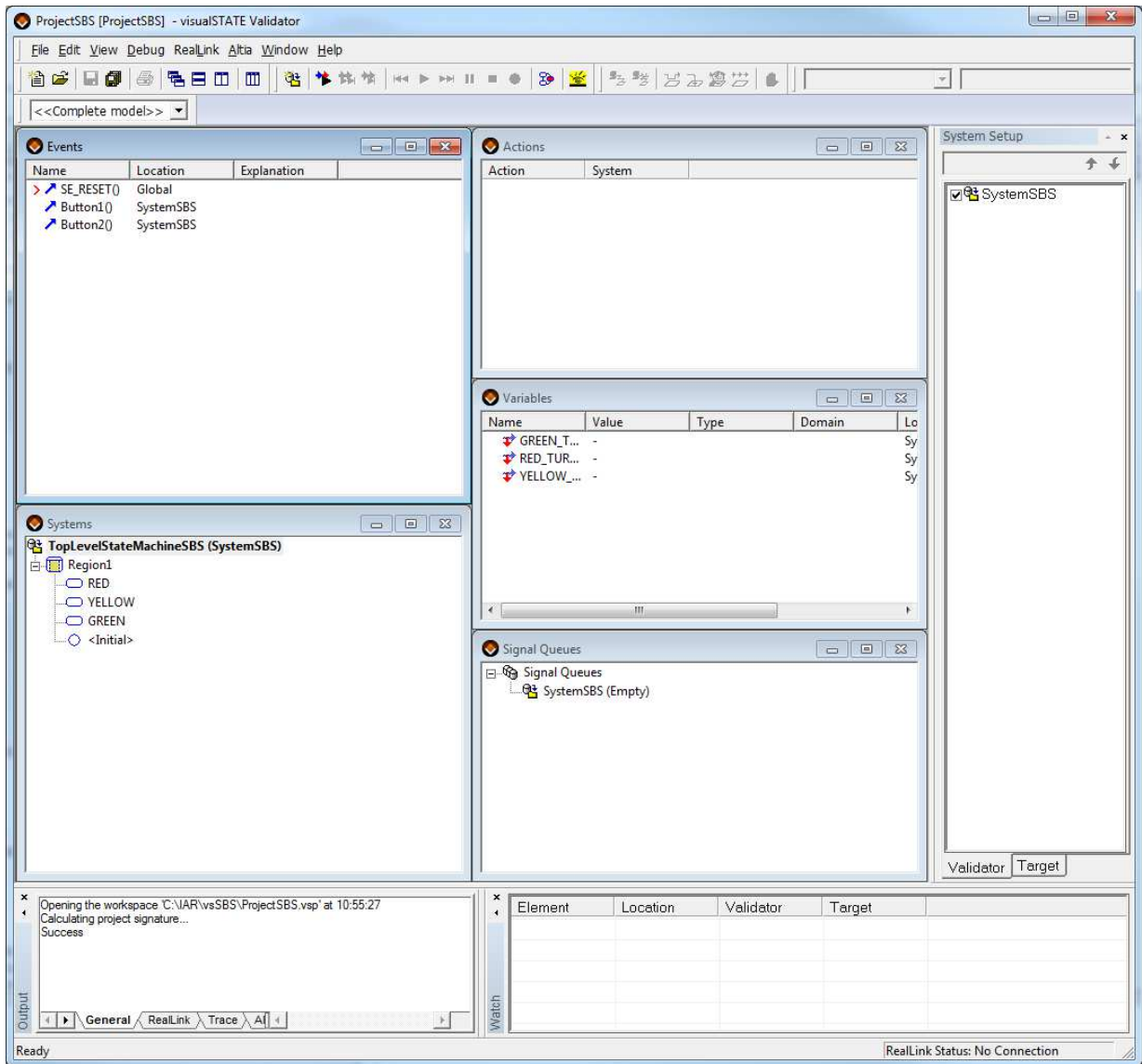
**Note:** If you would like to change settings before code generating, like change from table based to readable code, you can do that in **Project>Options>Code generation**.

## Testing your design (validating and verifying)

Now it is time to test your design. The Validator component is used to simulate and analyze your design and the Verificator component is used to check the logic consistency so you don't have dead ends, unreachable transitions and conflicting behavior of transitions.

## Validating

- 1 Open the Validator component by pressing F8 or click on the Validator symbol  on the toolbar. You should now have a window like this:

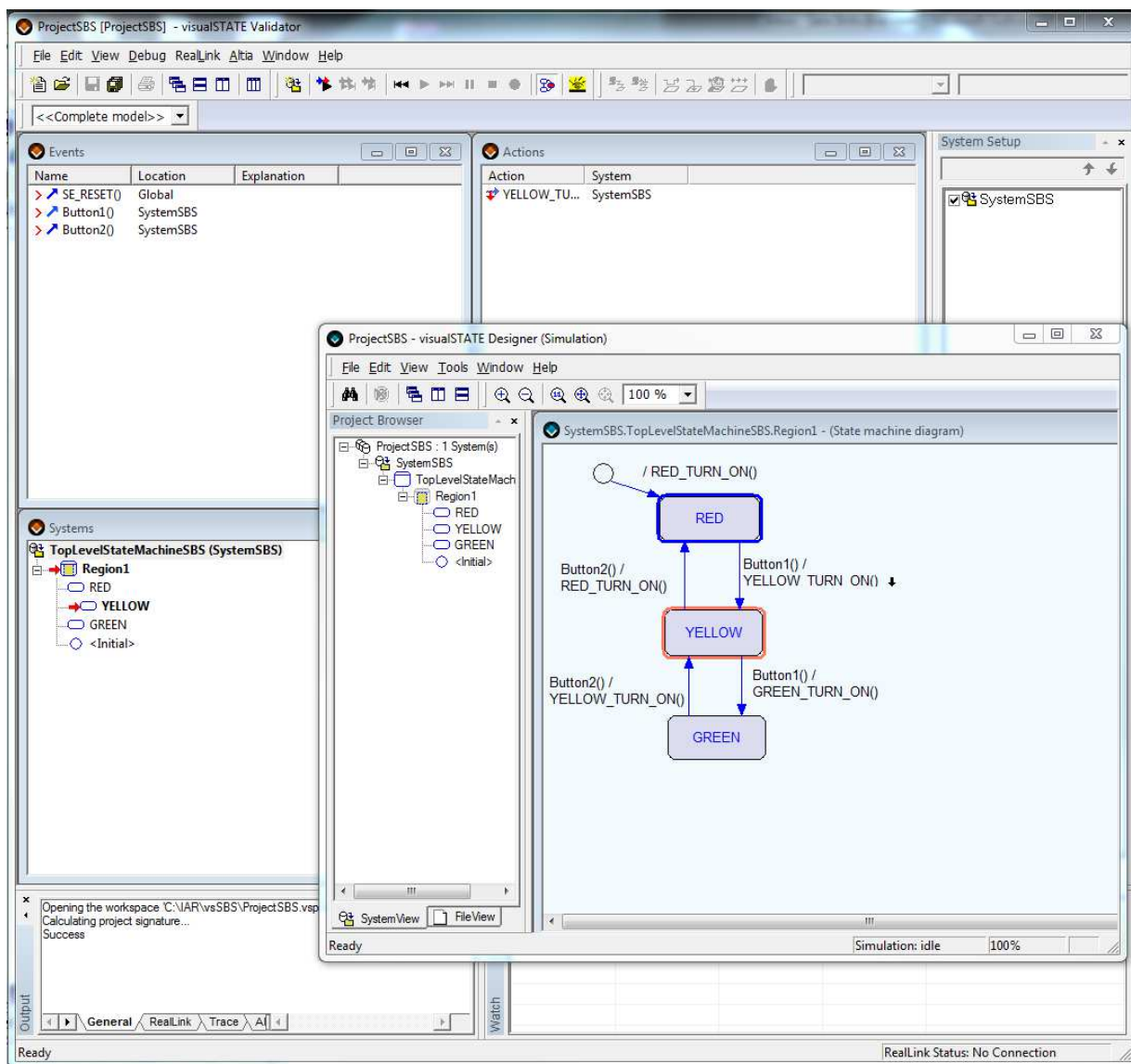


If not, open **Window>Simulation** for a quick arrangement of your windows.

- 2 To start your simulation, double click on SE\_RESET in the Event window. This will place your system in its initial state.
- 3 Step through your system by double-clicking on events in the Validator Events window and see if it behaves as you expect. In the Systems window you can see the resulting state combination. In the Actions window you can see which actions were generated by the event.



- There is also a graphical view. On the Validator menu, select **Debug>Graphical Animation** to step through your system graphically. Resize or reposition the Validator window in order to see the Designer Simulation Mode window. Double click on the Region1 in the System View to open the contents of the region in a separate window.

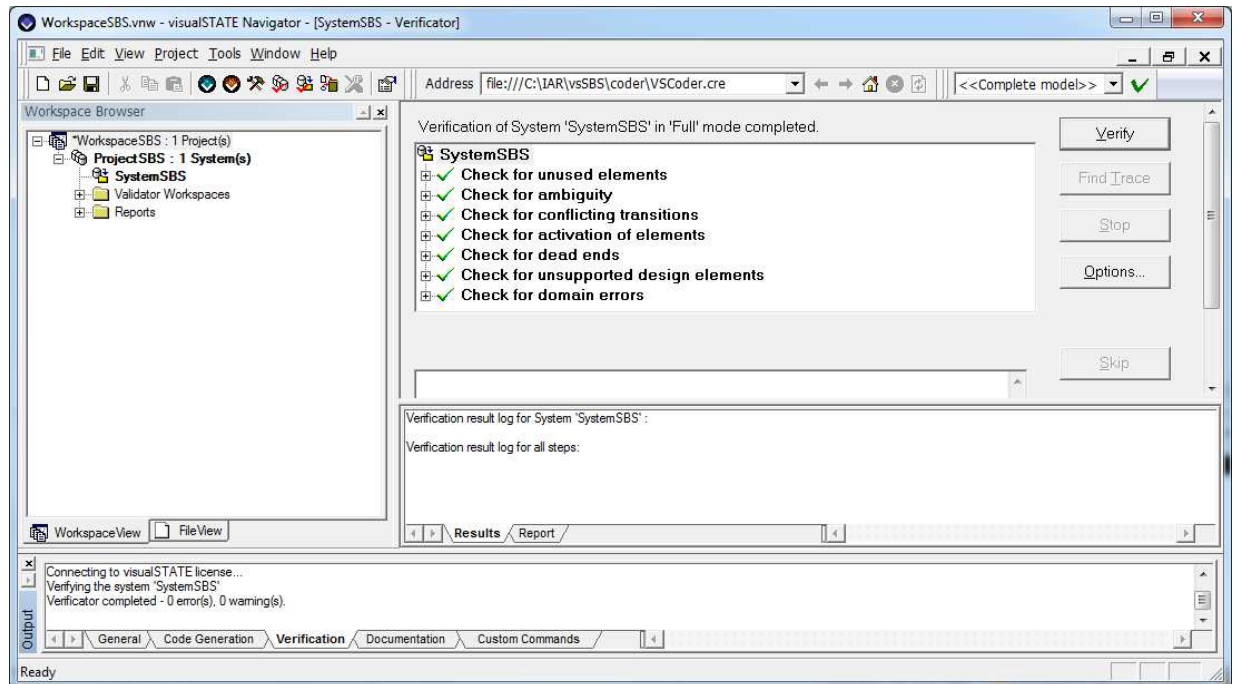


- When you double click events in the Events window, the resulting state combination will be shown with red borders in your state machine. Blue borders are used for states that were last active.
- Exit the Validator once you are done.

## Verifying

Now let us verify that the system does not have dead ends or other unexpected behavior.

- 1 After you have closed the Validator and are back in the Navigator, select **Project>Verify System** on the menu. As seen on the result, we have no conflicts or dead ends or other problems that the Verifier can detect.




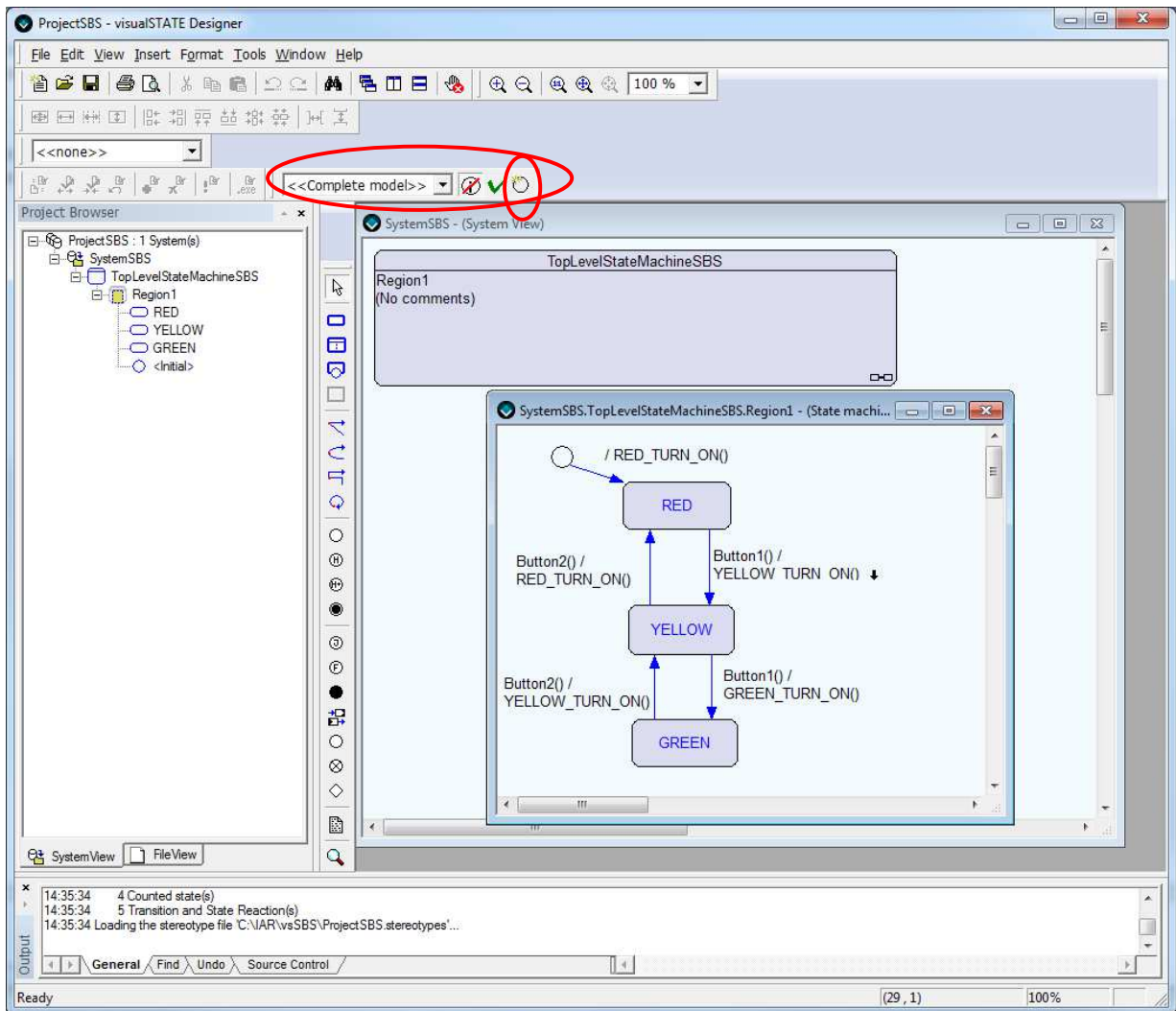
## Documenting

To create a documentation of your project open **Project>Document**. This will create a document with all possible information about your system.

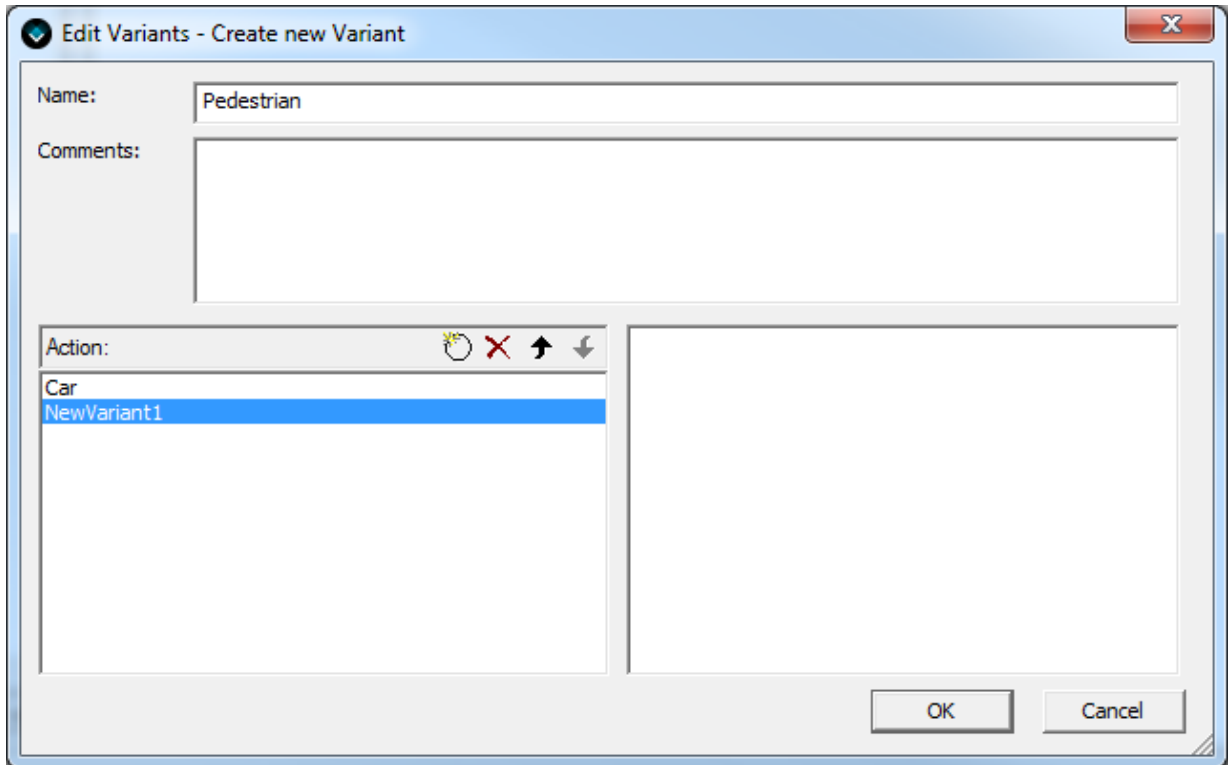
## Adding variants to your design

A new feature is the variants. Variants are used when you want to create different variants from the same model. In this case, what if we want to use this model to work on both a pedestrian light and a street light?

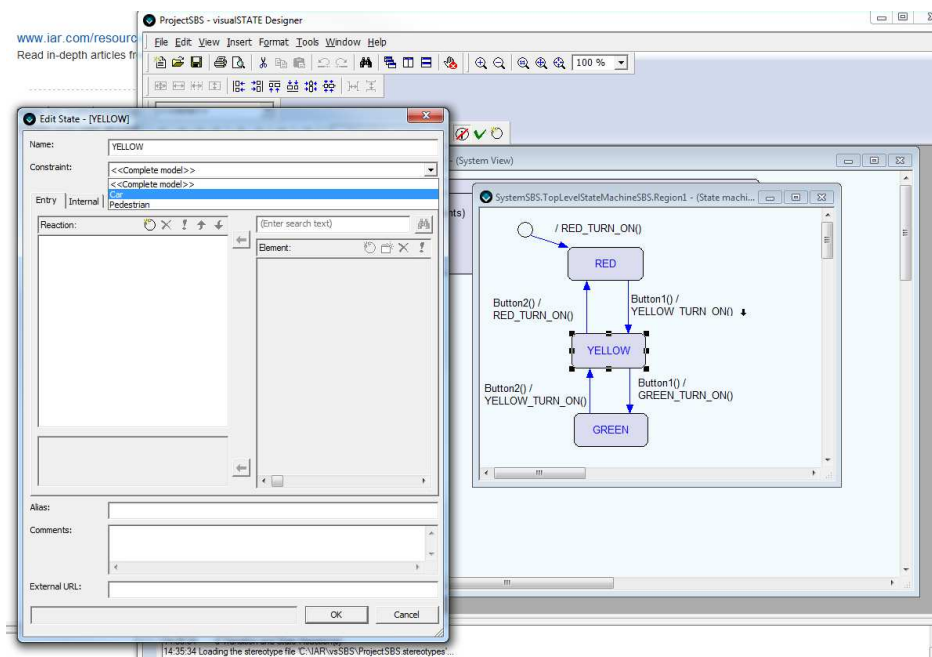
- I We start out with the previous design in Designer and on the menu bar you find a small toolbar for variants. If you click on the  you will be able to add the name of a new variant.



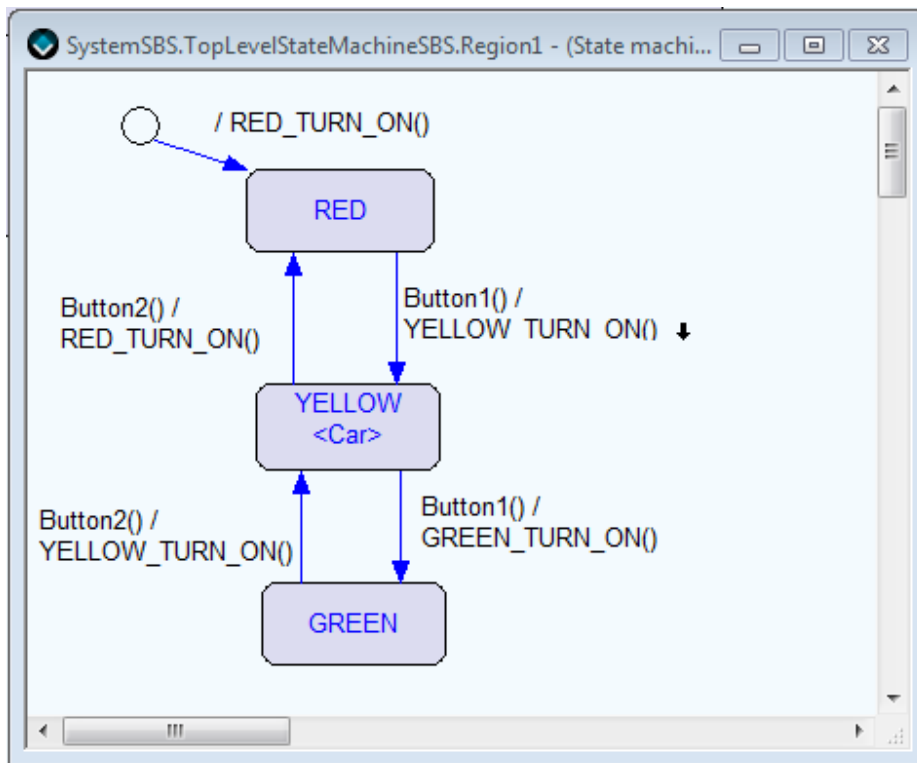
2 Create two new variants – Pedestrian and Car.



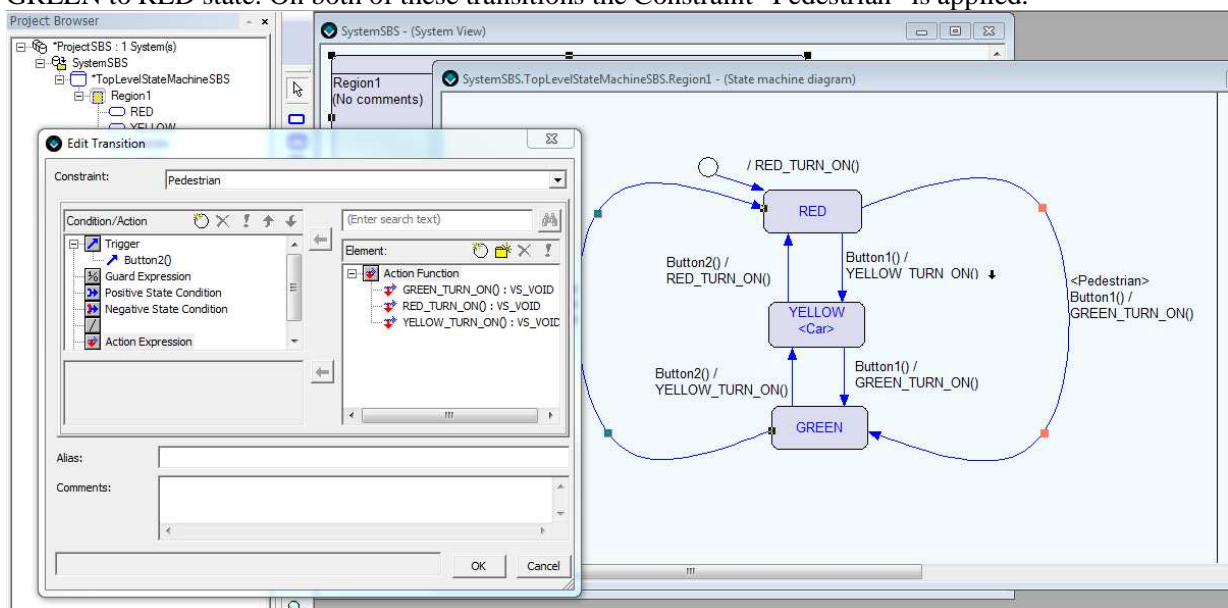
3 We want the pedestrian light to only show red and green, while the car/street light to go between red, yellow and green. There are several ways of solving this and shown is one of them. By double clicking on the state called YELLOW the Edit state opens up. In here I now choose the Constraint to be “Car” as I only want that state entered for the street light. Click **OK**.



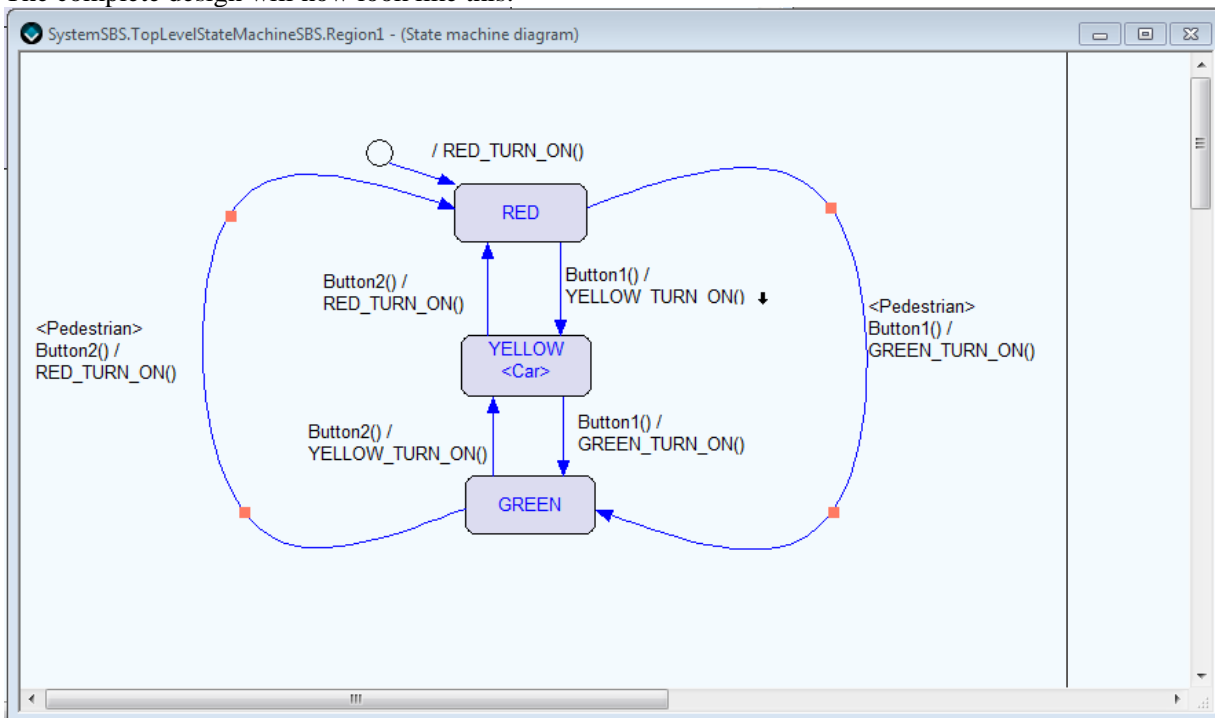
- The YELLOW state will now display the restriction <Car>. Transitions to and from this state will be under the same restriction.



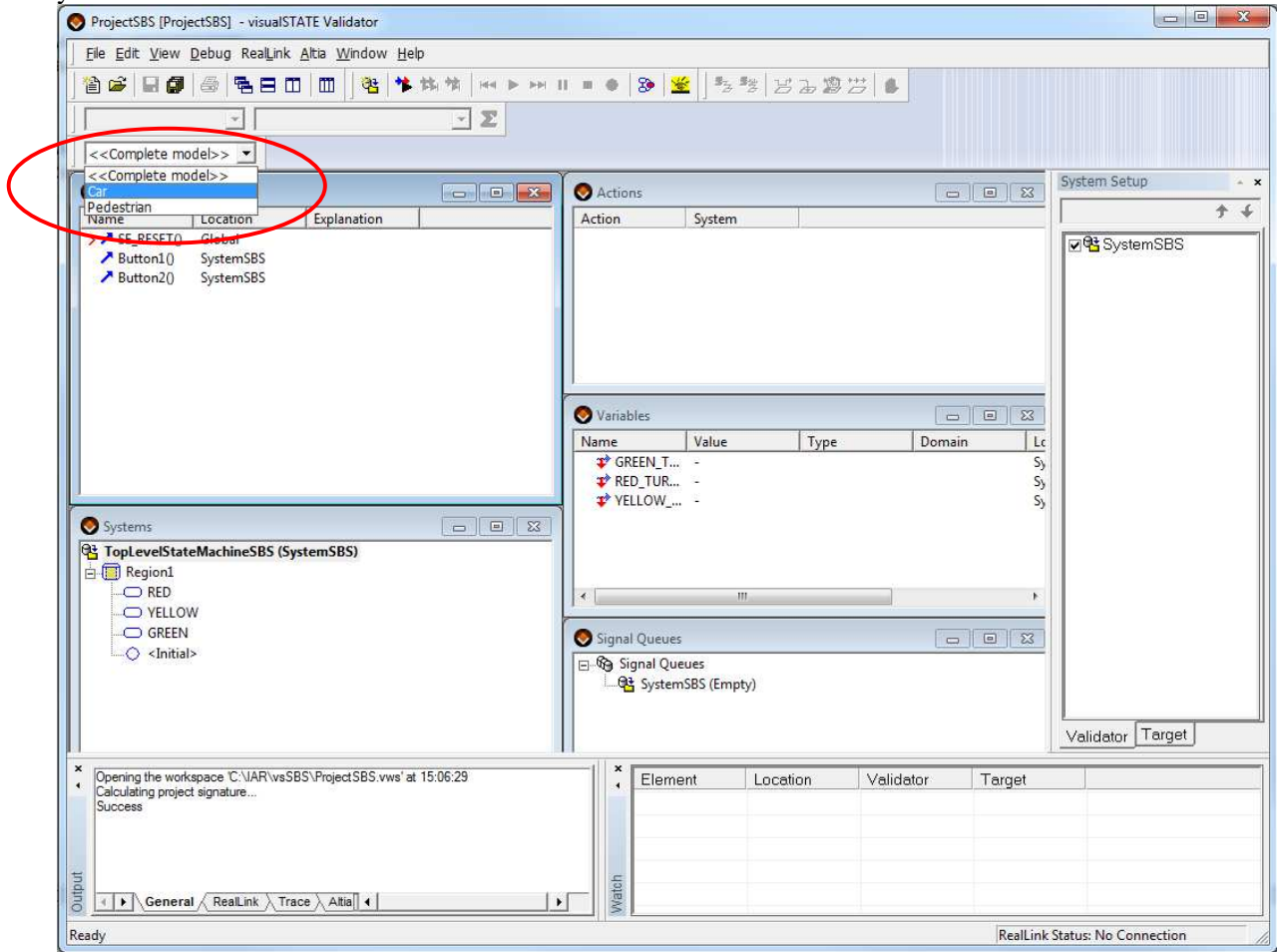
- Time to take care of the Pedestrian crossing. We have to add the transitions between the RED and GREEN state. We add trigger “Button1” and Action Expression “GREEN\_TURN\_ON” from RED to GREEN state, and trigger “Button2” and Action Expression “RED\_TURN\_ON” from GREEN to RED state. On both of these transitions the Constraint “Pedestrian” is applied.



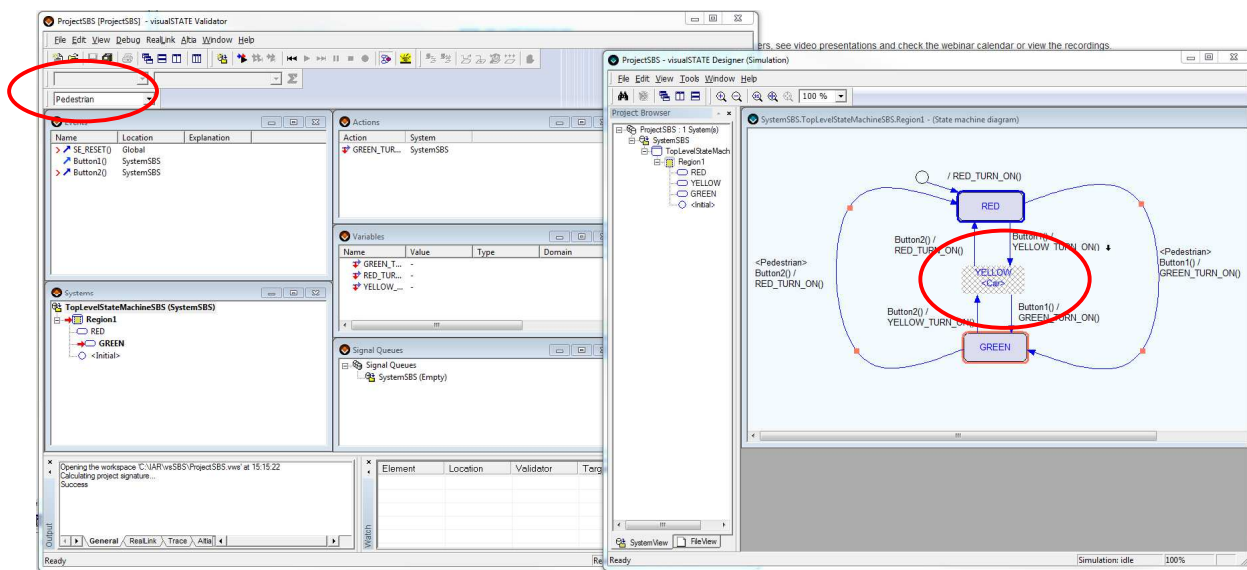
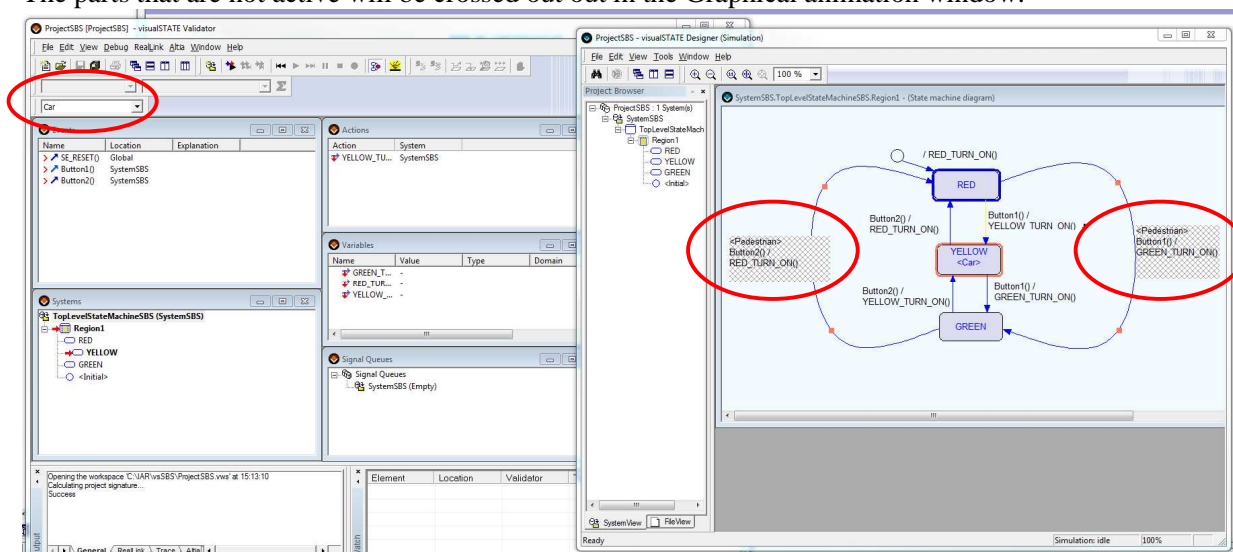
6 The complete design will now look like this:



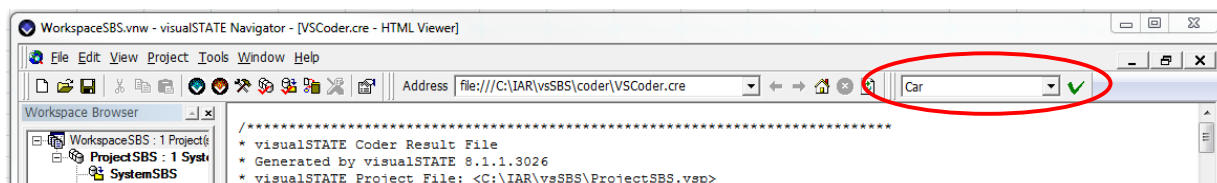
- 7 Save the project and open up Validator. You can now choose which one of the two variants that you want to validate.



8 The parts that are not active will be crossed out in the Graphical animation window.



When generating code for the different variants you specify in the Navigator toolbar which variant you want the code generated for.





### **Disclaimer**

Information given in this document is intended as a help for the user. IAR Systems shall not be held liable for any losses or problems arising from using the information given in this document.

### **Trademarks etc.**

IAR Systems, IAR Embedded Workbench, C-SPY, visualSTATE, IAR KickStart Kit, IAR and the IAR Systems logotype are trademarks or registered trademarks owned by IAR Systems AB. J-Link is a trademark licensed to IAR Systems AB.

All other trademarks or registered trademarks mentioned in this document are the properties of their respective owners.

© Copyright 2017 IAR Systems AB.

Part number: vS-StepByStep-3  
Third edition: February 2017