LogiBLOX Lab

LogiBLOX Lab

Introduction

In this lab, the user will create the TENTHS macro using the LogiBLOX GUI. This macro can then be added to the top-level schematic in Foundation. The TENTHS macro is a 10-bit one-hot encoded counter with reset logic to clear the counter after counting to the number nine. If the counter were built in the schematic editor, the appropriate reset logic would have to be designed into the schematic. By using the LogiBLOX GUI, the high-level specifications can be entered without having to wire-up many primitives into the counter.

Objectives

In this laboratory, it will be shown:

- How to open the LogiBLOX GUI.
- How to create the TENTHS macro using the LogiBLOX GUI.
- How to add the TENTHS macro to the top-level schematic of the WATCH Project.
- How to display a symbol name in a schematic.
- How to make pin assignments on the OUTS, OUTS1 and OUTS2 macros.

Procedure

Opening the LogiBLOX GUI from the Foundation Project Manager

1) From the "**Foundation Project Manager**", open the LogiBLOX GUI with the menu command: ***Tools>LogiBLOX***. After LogiBLOX has been loaded, the "**LogiBLOX Module Selector**" dialog box opens.

Creating the TENTHS macro

1) To create a 10-bit one-hot encoded counter that resets automatically at nine, select a Module Type of "**Counters**", enter a bus width of "**10**", and enter a Module Name of "**TENTHS**".

2) The counter's operation should be "**up**", the style could be any of the available selections, and the Encoding should be "**One-Hot**". Since performance is not a necessity for this project, the Style is irrelevant.

3) Finally, since the stop watch should be externally resetable, there needs to be an asynchronous clear on the counter by setting the Asynchronous Value to "**0000000001**". Since the counter is one-hot encoded, one register must be asserted to denote a zero.

4) Since there is no need to load this counter, the D_IN box should be deasserted. Likewise, since this counter will generate a carry out signal when it reaches the terminal count, the Terminal Count box should be asserted.

5) After entering this information, the "**LogiBLOX Module Selector**" dialog box should look like Figure 1.

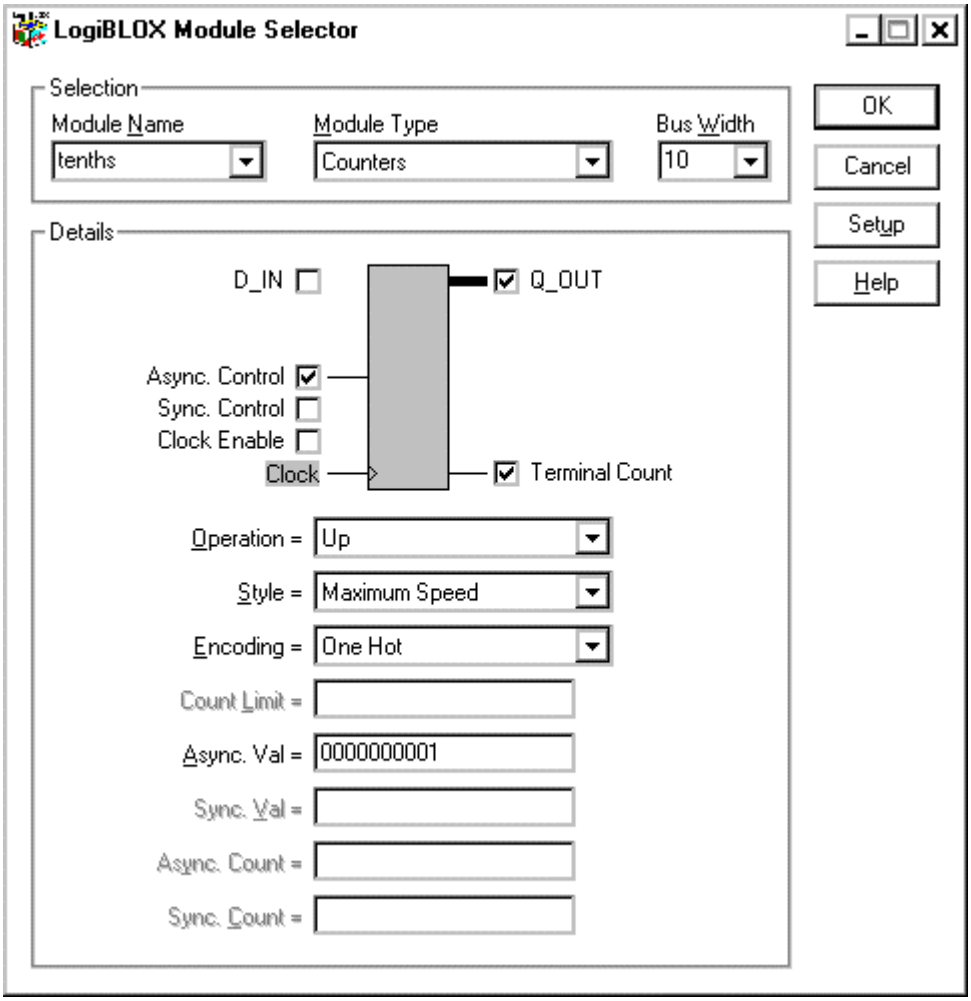

Figure 1. The LogiBLOX Module Selector for the TENTHS macro.

6) Once the appropriate selections and details have been entered into the "**LogiBLOX Module Selector**" dialog box, click on the "**OK**" button. This will generate an EDIF netlist, create a symbol for the TENTHS macro, and attach these to the WATCH project library.

7) The LogiBLOX GUI will then issue a message letting the user know the macro was created and attached to the WATCH Project successfully. Click on "**OK**" in the message window. Once this is completed, the TENTHS macro can be added to the WATCH schematic.

Adding the TENTHS macro to the top-level schematic

1) Open the "**Schematic Editor**" by clicking on the menu command ***Tools>Schematic Editor*** or by clicking on the icon.

2) Once the program has started, click on the "**Symbols Toolbox**" icon on the vertical toolbar. This will open the "**SC Symbols**" box, which now contains the

TENTHS macro.

3) To obtain the TENTHS symbol, scroll through the list of symbols, or enter the name at the bottom of the "**Symbols Toolbox**". After finding the TENTHS symbol, move the pointer into the work area, and place it as shown in Figure 2, by clicking the left mouse button.

4) After placing the TENTHS symbol, go ahead and place the CNT60, OUTS, OUTS1, and OUTS2 symbols as in Figure 2. Do not bother to wire them up since, this will be performed in the last lab.

5) Enter "**Select and Drag**" mode by clicking on its icon.

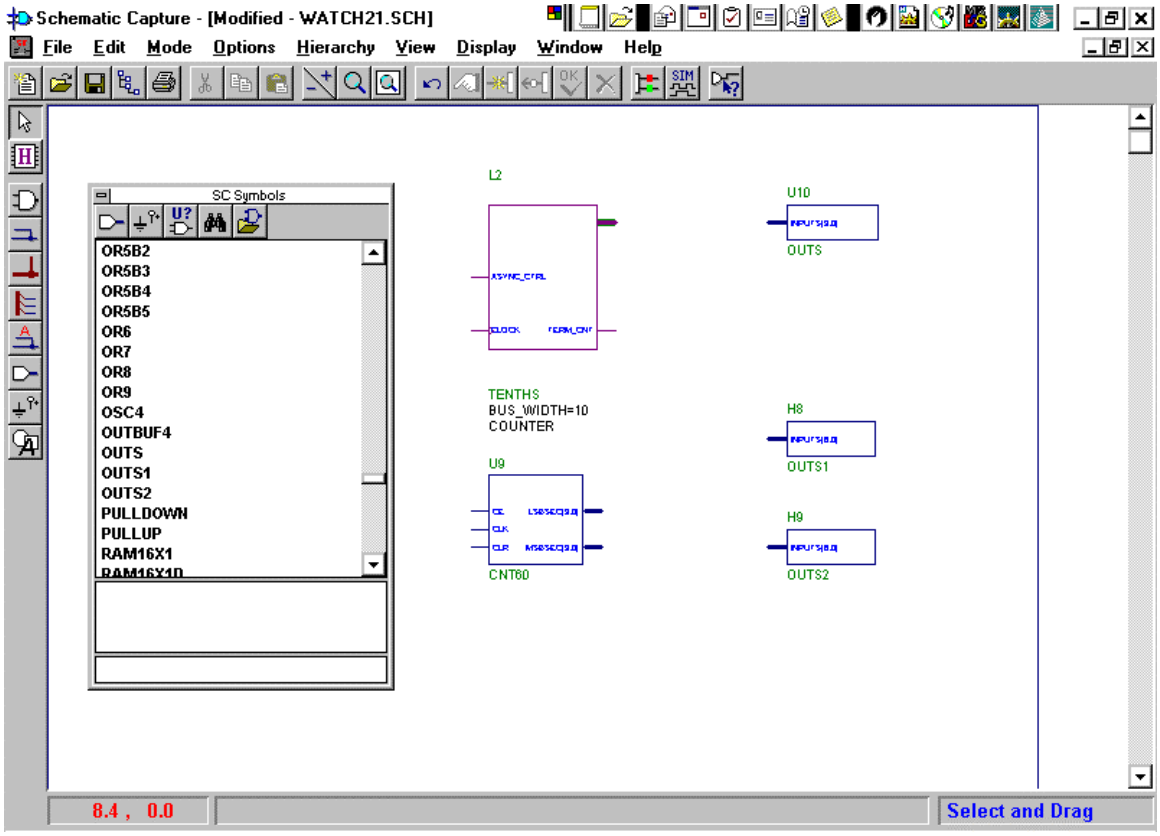

Figure 2. The WATCH schematic.

Displaying the name attribute of the TENTHS macro on the schematic

1) Open the "**Symbol Properties**" dialog box by double clicking on the TENTHS macro in the schematic editor (see Figure 3).

2) Click on the "**Attributes**" button. This will open the "**Symbol Attributes**" dialog box (see Figure 4).

3) Assert the "**Symbol Name**" box in the Visible Text section of the "**Symbol Attributes**" dialog box. Click on the "**OK**" button. Finally, click on the "**OK**" button in the "**Symbol Properties**" dialog box.

4) Note that the name "**TENTHS**" appears below the symbol in green.

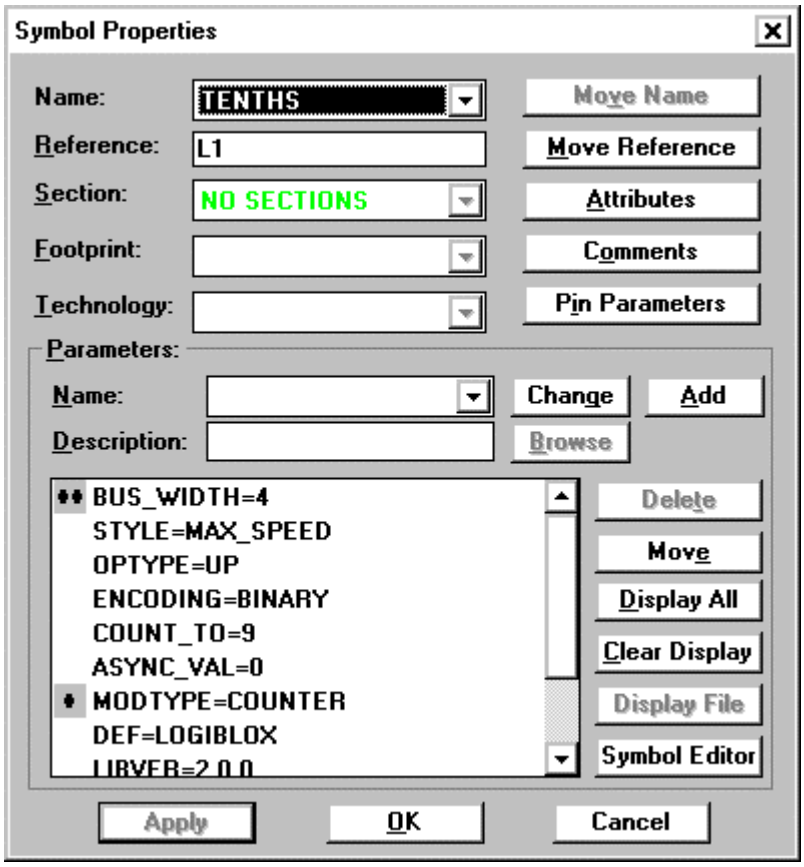

Figure 3. The Symbol Properties dialog box for the TENTHS macro.

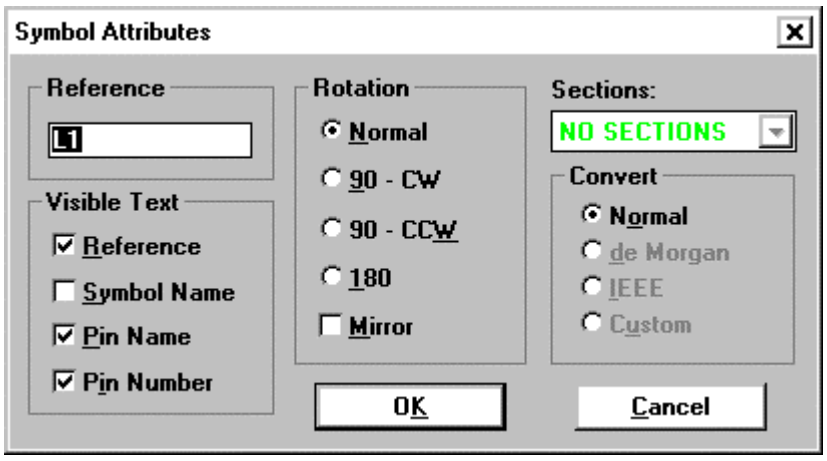

Figure 4. The Symbol Attributes dialog box for the TENTHS macro.

Placing Pin Assignments on the OUTS, OUTS1, and OUTS2 macros

1) After placing the OUTS, OUTS1, and OUTS2 macros in the schematic, push into the OUTS macro by clicking on the "**Hierarchy Push/Pop**" button in the Schematic Editor, and then double-clicking on the OUTS symbol in the schematic. This will open the OUTS sheet attached to the project and enable the user to see its contents. Enter "**Select and Drag**" mode by clicking on its icon.

TENTH0 = P37
TENTH1 = P41
TENTH2 = P61
TENTH3 = P62
TENTH4 = P65
TENTH5 = P66
TENTH6 = P57
TENTH7 = P58
TENTH8 = P59
TENTH9 = P60

Figure 5. The pin assignments for the OUTS macro.

ONES0 = P49
ONES1 = P48
ONES2 = P47
ONES3 = P46
ONES4 = P45
ONES5 = P50
ONES6 = P51

Figure 6. The pin assignments for the OUTS1 macro.

TENS0 = P39
TENS1 = P38
TENS2 = P36
TENS3 = P35
TENS4 = P29
TENS5 = P40
TENS6 = P44

Figure 7. The pin assignments for the OUTS2 macro.

2) Double-click on the OPAD symbol connected to the TENTH0 node. This will open the "**Symbol Properties**" dialog box.

3) In the Parameters section of the "**Symbol Properties**" dialog box, click on the arrow button next to the Name field. This will open the parameter list.

4) Once the parameter list is opened, scroll down in the list and click on "**LOC**". Alternatively, "**LOC**" can be entered into the Name field. Once this has been selected, it should appear in the Name field, and the user can enter "**P37**" in the Description field.

5) After entering this information, click on the "**Add**" button to add this parameter to the symbol parameters list. It should now appear highlighted in the window in the Symbol Properties dialog box. Double-click on this parameter until it appears with two dots next to its name. The double-dots make the pin assignment label appear in the schematic.

6) After completing these steps, click on "**OK**" in the "**Symbol Properties**" dialog box to go back to the schematic.

7) Once in the schematic, if the placement of the pin assignment label is poor, re-enter the symbol properties dialog box and select the pin assignment again by clicking on the "**LOC**" parameter. Next, click on the "**Move**" button in the Parameters section. This will open the schematic editor, and allow the user to place the pin assignment label in a better location.

8) After all the pin assignments have been made, double-click on empty space while in "**Hierarchy Push/Pop**" mode to go to the top-level schematic. Save the changes made to the schematic if prompted to save.

9) Repeat these steps until all the pin assignments have been made to the OUTS, OUTS1, and OUTS2 macros (see Figures 5, 6, and 7). It is not necessary to have the pin assignment labels appear in the schematic for the assignments to be made, however they must appear in the window of the "**Symbol Properties**" dialog box.

10) Save the WATCH Project within the Schematic Editor with the menu command: ***File>Save***.

11) Exit the "**Schematic Editor**" by clicking on ***File>Exit***.

Placing Pin Assignments in a UCF file

1) Alternatively, pin assignment can be entered into a User Constraints File provided by the Project Manager. To open the this file, double-click on its icon in the Project Manager.
2) Once this file has opened, numerous possible constraints examples are given. Appropriate syntax for constraints and some of their capabilties is provided in the M1 Programmable Logic Training Course.
3) Enter the following text into the WATCH.UCF file:

```
NET TENTH0 LOC = P37;
NET TENTH1 LOC = P41;
NET TENTH2 LOC = P61;
NET TENTH3 LOC = P62;
NET TENTH4 LOC = P65;
NET TENTH5 LOC = P66;
NET TENTH6 LOC = P57;
NET TENTH7 LOC = P58;
NET TENTH8 LOC = P59;
NET TENTH9 LOC = P60;

NET ONES0 LOC = P49;
NET ONES1 LOC = P48;
NET ONES2 LOC = P47;
NET ONES3 LOC = P46;
NET ONES4 LOC = P45;
NET ONES5 LOC = P50;
NET ONES6 LOC = P51;

NET TENS0 LOC = P39;
NET TENS1 LOC = P38;
NET TENS2 LOC = P36;
NET TENS3 LOC = P35;
NET TENS4 LOC = P29;
NET TENS5 LOC = P40;
NET TENS6 LOC = P44;
```

4) After the file is saved, it will automatically be read by the Alliance M1 software.

5) Note. Do not make duplicate constraints in a UCF and a schematic since the UCF always overrides schematic constraints. Xilinx recommends making location constraints in a schematic since changes are less likely to be made by mistake.

Conclusion

In this laboratory, it was shown:

- How to create a 10-bit one-hot encoded counter using the LogiBLOX GUI
- That LogiBLOX modules can be entered into a schematic using the Symbols Toolbox in the Schematic Editor
- The "**Symbol Properties**" and "**Attributes**" dialog boxes allow the user to control how a symbol is displayed in the Schematic Editor
- Pin assignments can be entered into a schematic by entering the "**LOC**" attribute and the pin number into the "**Symbol Properties**" dialog box

Questions

1) How is the operation for a LogiBLOX counter chosen?

2) How is a symbol rotated in the Schematic Editor?

3) How can a pin assignment be made visible on a schematic?

4) How can pin assignments be made without entering them into a schematic?