MaxPlus II ROM Creation Instructions

Problem: You have an ASM or CPU that you would like to control/test from an EEPROM. How can you simulate the EEPROM under MaxPlus?

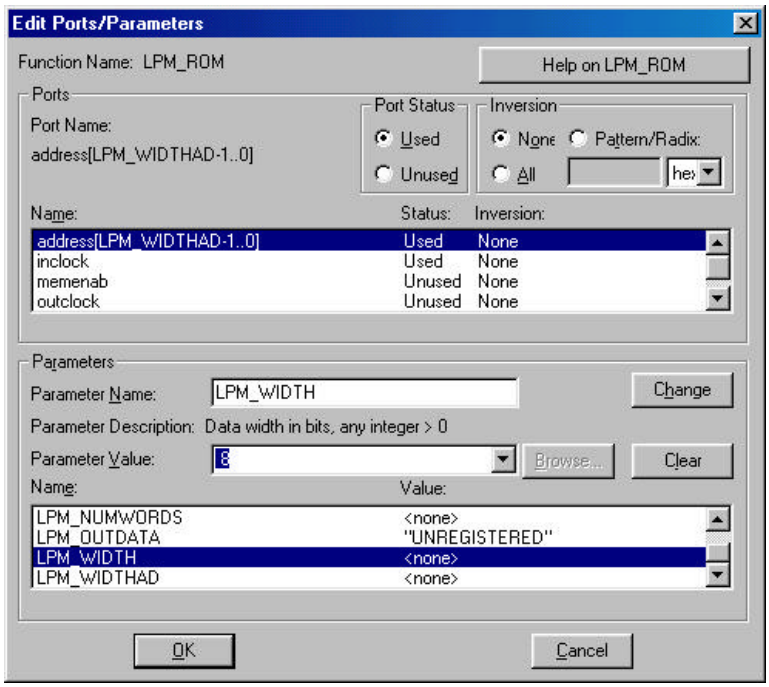
Solutions

Use the ROM model found in the LPM library.

Design Procedure:

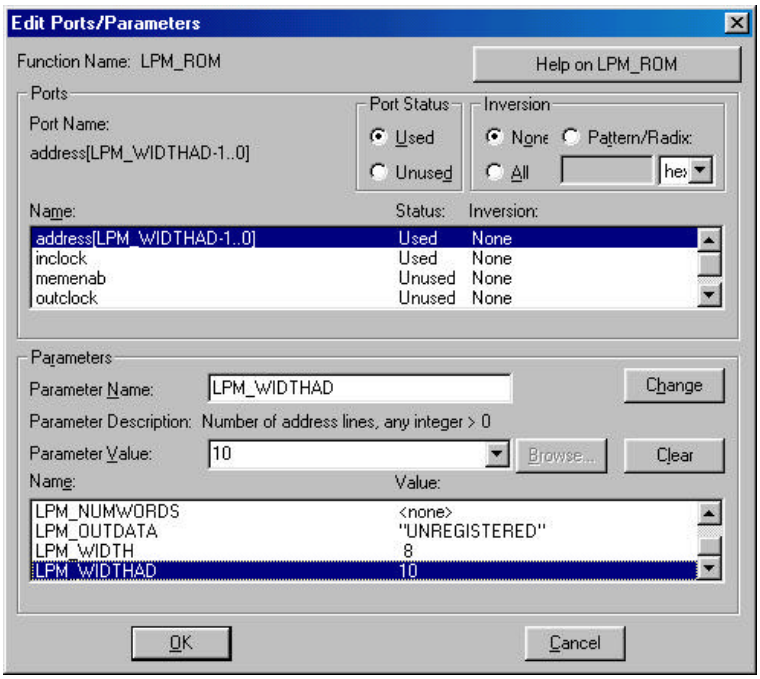
1. Create a new GDF file (File->New…). In your schematic (GDF file), add an "lpm_rom" component found in the "mega_lpm" library.

2. You should see the following window open up when you try to place the component onto your schematic:



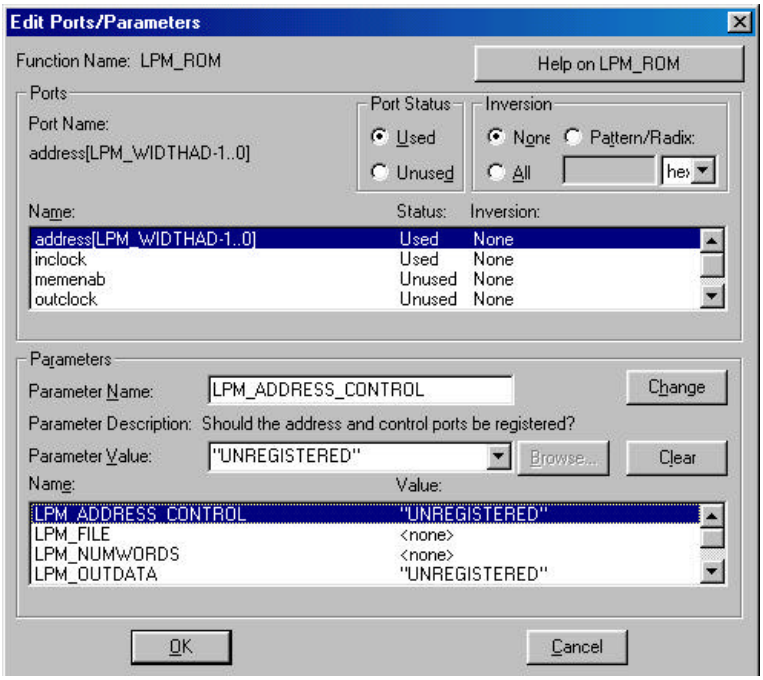
3. Select the "**LPM_WIDTH**" parameter in the bottom scroll window with your mouse and enter the size of the ROM's **data bus** in the "Parameter Value:" pull down menu. In this example we will set an 8 bit data bus. Then hit the "Change" button.

4. Select the "**LPM_WIDTHAD**" parameter in the bottom scroll window and set the parameter value in the "Parameter Value:" pull down menu. In this design we will create a 1KB ROM that has 10 address lines. Again hit "Change."



5. Make sure that the other parameters are now set as shown below by again using the pull down "Parameter Value:" menu.

   LPM_ADDRESS_CONTROL UNREGISTERED
   LPM_FILE <none>.
   LPM_NUMWORDS <none>.
   LPM_OUTDATA UNREGISTERED.



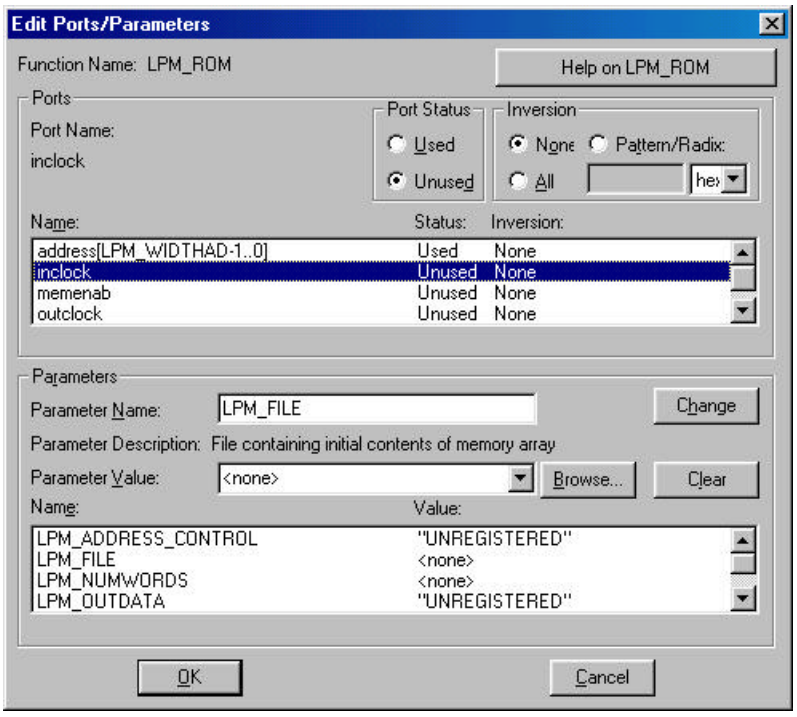
6. In the Ports area, make sure that only the following have Port Status Used.

   address[LPM_WIDTHAD-1..0] (address bus)
   q[LPM_WIDTH-1..0] (data bus)

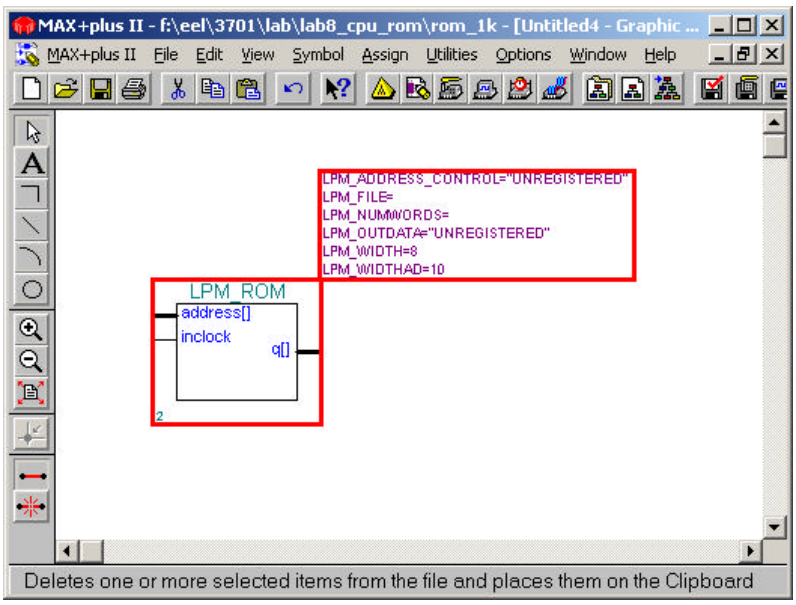
   The following have *Port Status* **Unused**

   inclock
   memenab
   outclock

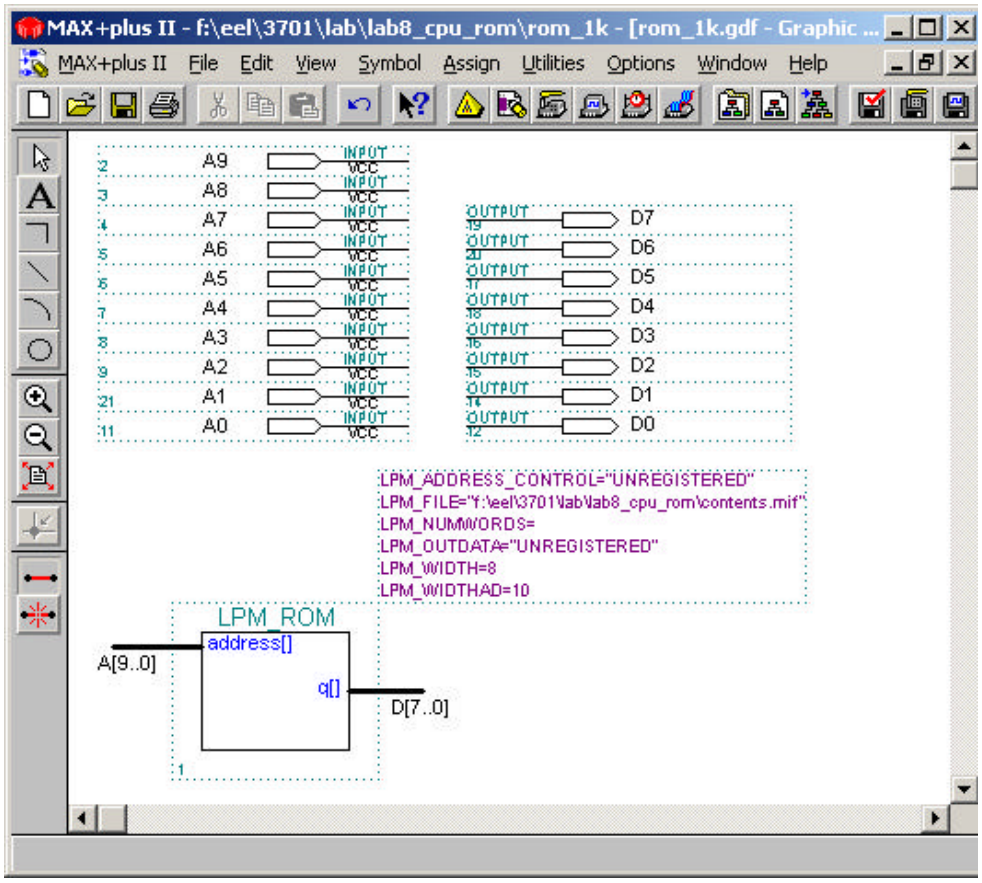
   These signals are in the *Name* section under Ports.

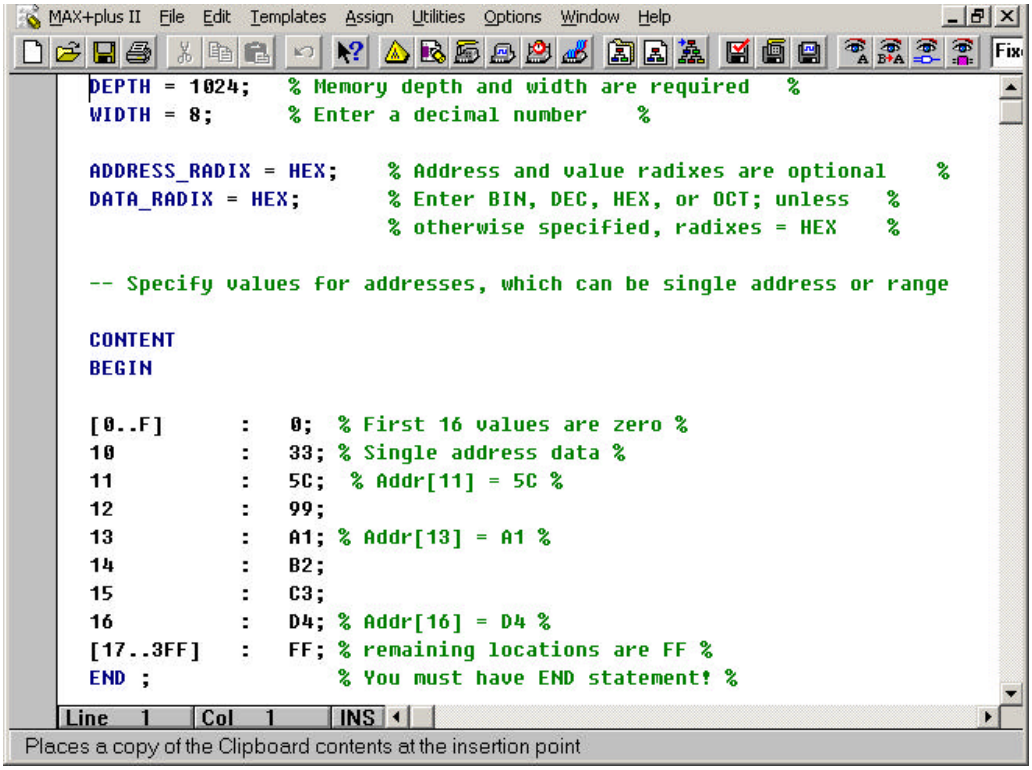


7. Hit "OK" and you now should see a component on your schematic that looks like the picture below.



8. Add a bus to the address inputs and a bus to the data (q[]) outputs. Label the address and data bus by clicking on them. Here we have used A[9..0] and D[7..0] as the example names and they must be in the form of name[msb..0] where msb is the most significant bit starting from zero. You can now use these signals anywhere else in your circuit or as inputs & outputs (as shown in this example).





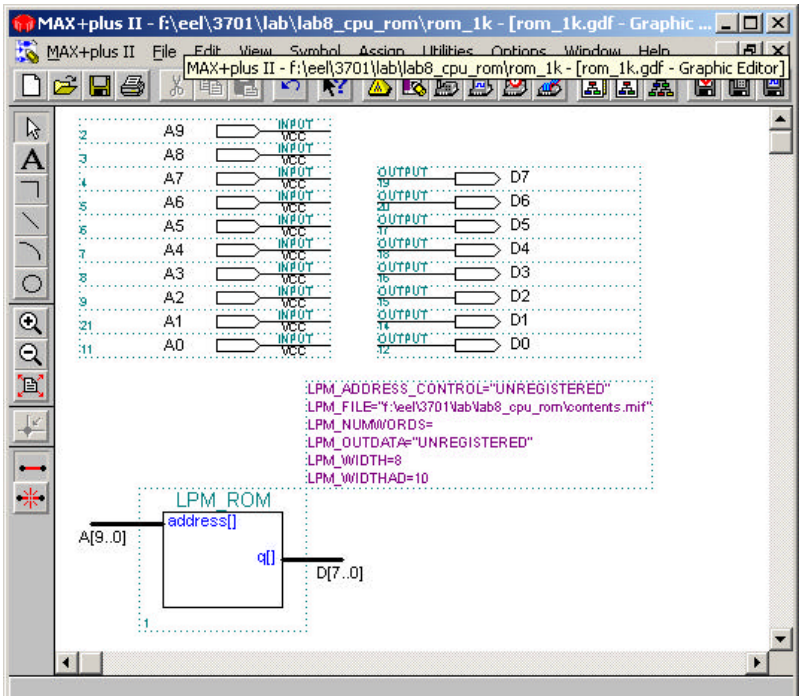
9. Create a MIF file that will contain the memory contents you want programmed in the ROM. This example is called "rom_1k.mif" and can be created with a text editor in MaxPlusII or other text editiors (i.e., Notepad, DOS editor, etc.). The "DEPTH" is the number of memory locations in the ROM and the "WIDTH" is the size of the data bus. We then specify the number format (BIN, DEC, HEX or OCT) for entering the address and data values. This example has 16 locations with zeros, 7 different data values and then FF for the remaining locations.

10. Now double click on the text on top of ROM component (or you can right mouse-click on the ROM component and select "Edit Ports/Parameters…"). Select the "LPM_FILE" and use the "Browse" button to set this parameter to your MIF file. Hit "Change" and it should show up in the LPM_FILE field now in your ROM component on the schematic. Hit "OK".

11. Assign a device that allows for ROM creation. The Flex10K devices are an example of a device that supports this feature.

    If you want to make a **larger ROM** than the Flex10K can fit (for **simulation only**, not for implementation), you will need to do a functional simulation instead of a timing simulation. After pulling up the compiler select Processing | Functional SNF Extractor. This will allow you to simulate extremely large ROMs.

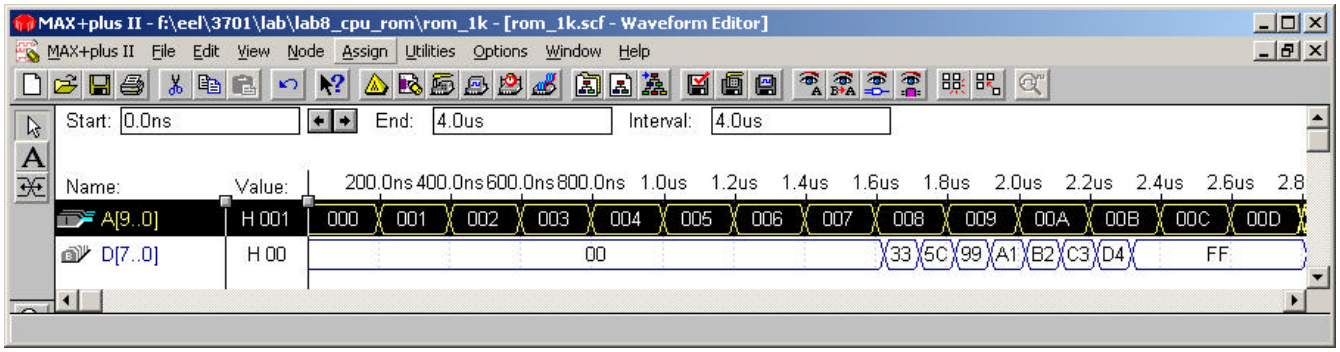
12. Make sure your project is set to this design and compile.



MaxPlus II ROM Creation Instructions

Simulation:

The simulation is very similar to any other simulation you have performed. Simply set the address input lines (or drive them from a counter or register) and observe the data that comes out of the ROM. An example is given below.

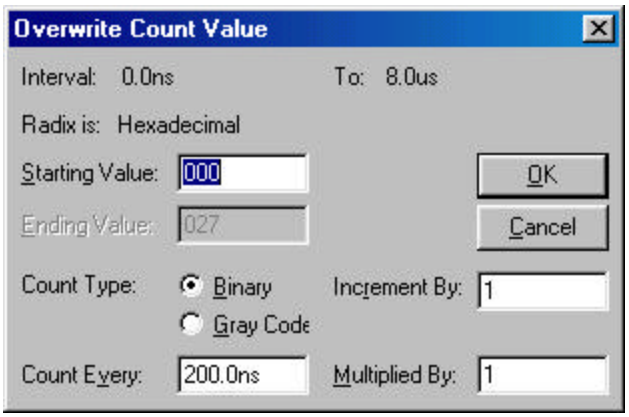


The nodes used in simulation were A9, A8, A7, … A0 and D7, D6, D5, … D0. These were then highlighted (selected using the mouse) and upon right clicking the mouse were set as buses using the "Enter Group" pull down option. The address lines were then set to increment by on for the simulation.

Very Important Note: The ROMs are very slow. Therefore when you simulate you should change the smallest increment of time from 25ns to 200 ns. You can do this by selecting "Options" on the top toolbar when you are in the waveform editor and select "Grid Size" to set the value to 200ns.

Another way to change the smallest increment of time in the waveform editor is to again go to the "Options" pull down and make sure the "Snap to Grid" feature is turned off. Now you can select the Address bus (A[9..0]) and press the count sequence button on the leftmost toolbar to open the Overwrite Count Value window. Here you can change the time in the "Count Every:" window and increment the address bus value by 1.

The end time was also increased for this simulation (via the "File" pulld own button on the top toolbar) and the results are as shown here.



The End.