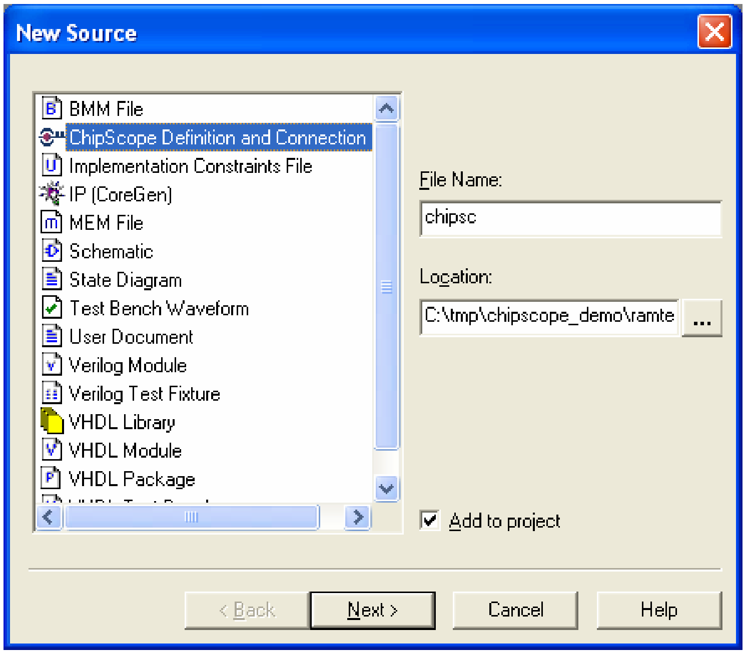
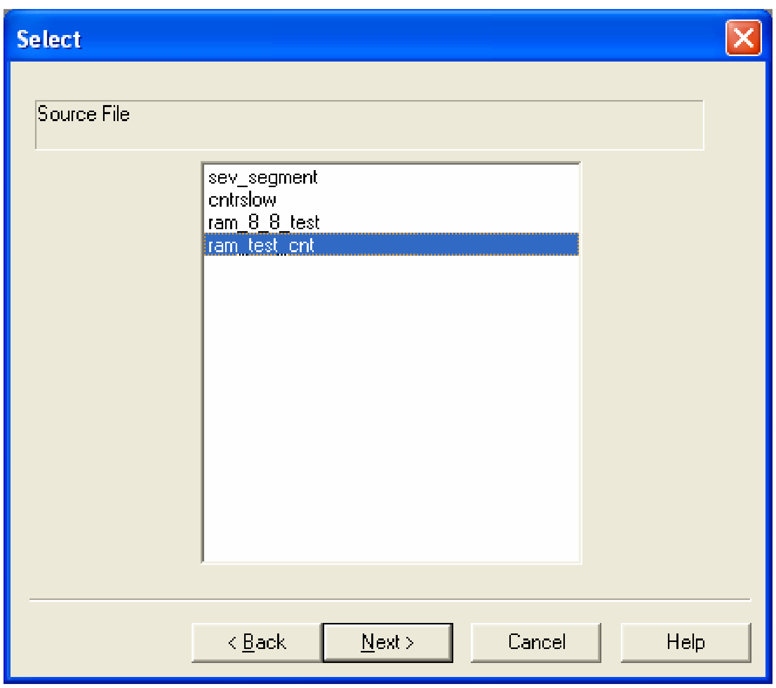
Chipscope Tutorial (Anselmo Lastra, 2/2004)

Chipscope is a Xilinx product that inserts logic-analyzer circuitry onto the chip.

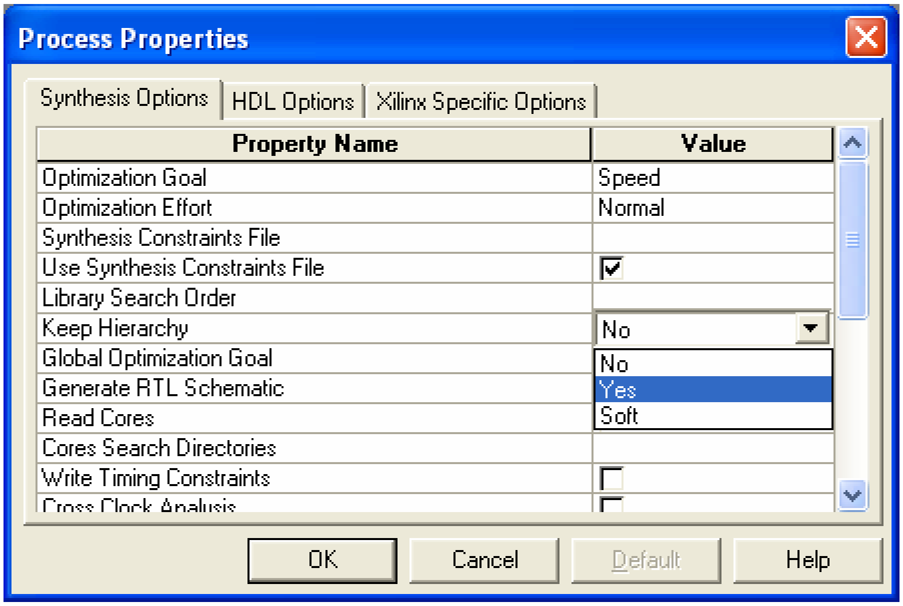
Add a new source of type Chipscope.



Associate it with the top-level source.

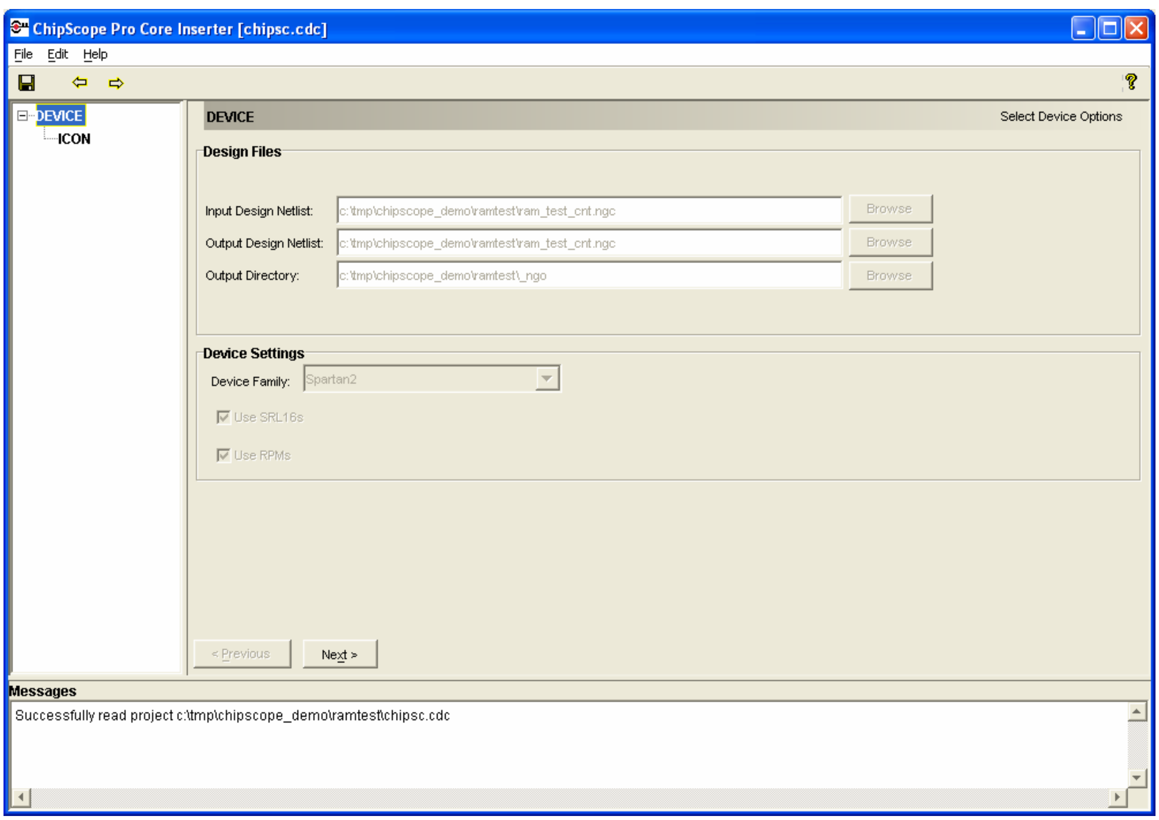


Choose menu item Process/Properties. In order to make it easier to decipher the signal names, choose "Yes" under *Keep Hierarchy*.

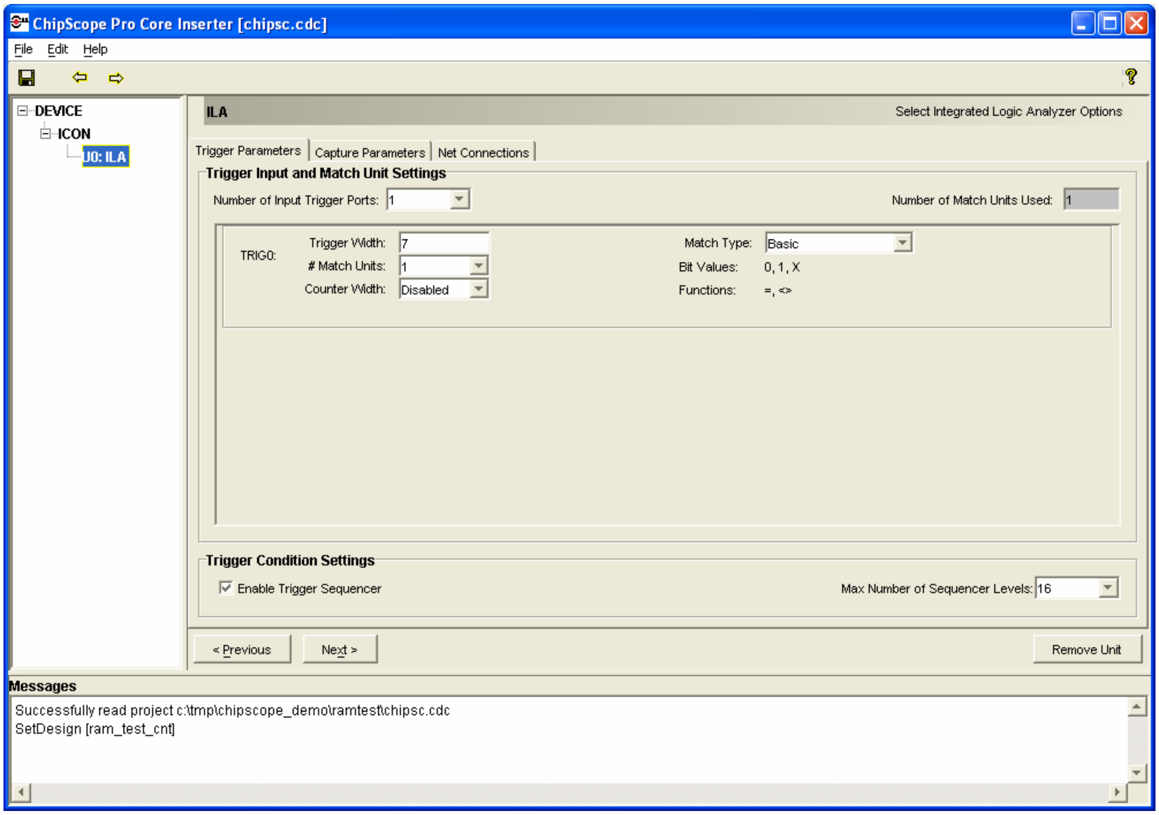


Choosing Signals to Monitor

Now open the source that you added (chipsc in this case). Note that the design must synthesize, so it needs to be at least at that level of completion at this point. You'll see the following window.



This shows that a Controller Core (ICON) has been added. You need to click *twice* on either the yellow arrows below the menu bar, or on *Next*.



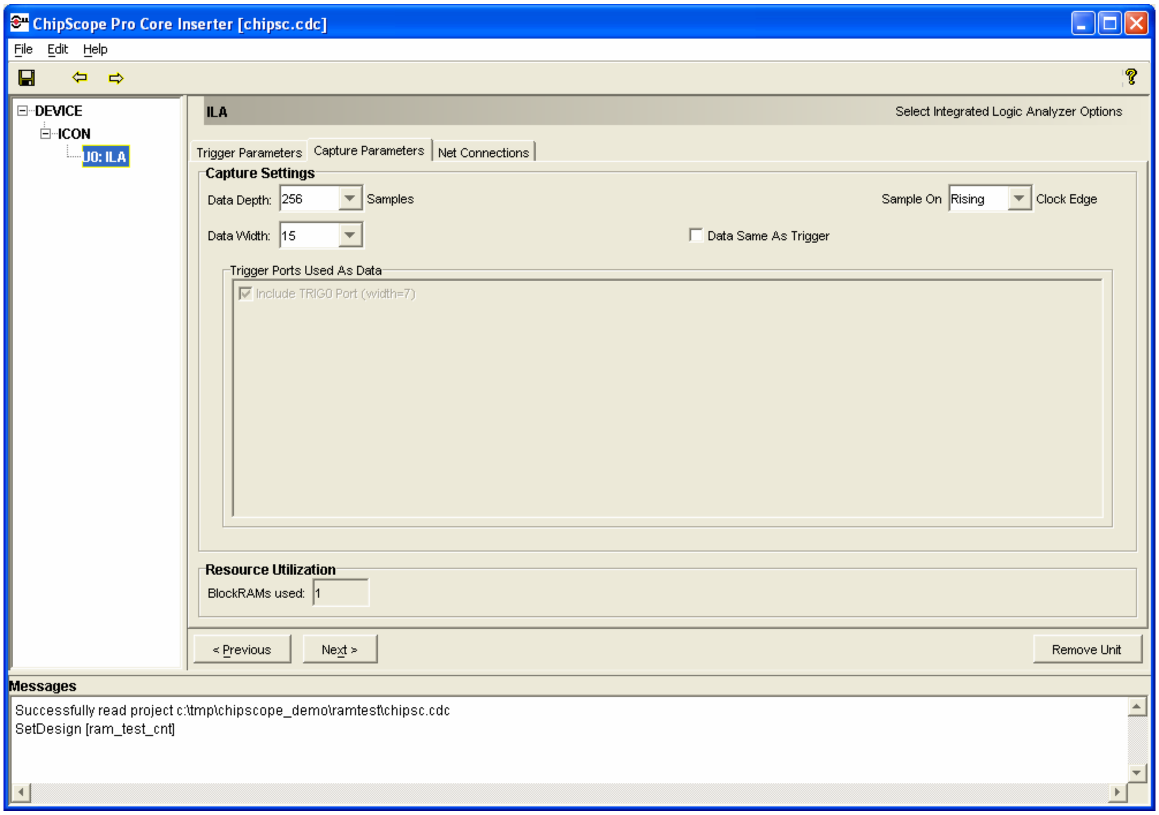
This screen shows that a Logic Analyzer Core (ILA) has now been added. This and the next few screens will be where you select the signals to trigger and to sample.

We'll select the trigger width to be seven, so we can use six low-order address bits and the slower clock. The *Match Type* box allows you to select a core that provides the complexity of trigger that you need. The basic match unit just tests for equals or not-equals. Fancier ones (which use more logic) can test whether a set of signals has a value within a specified range.

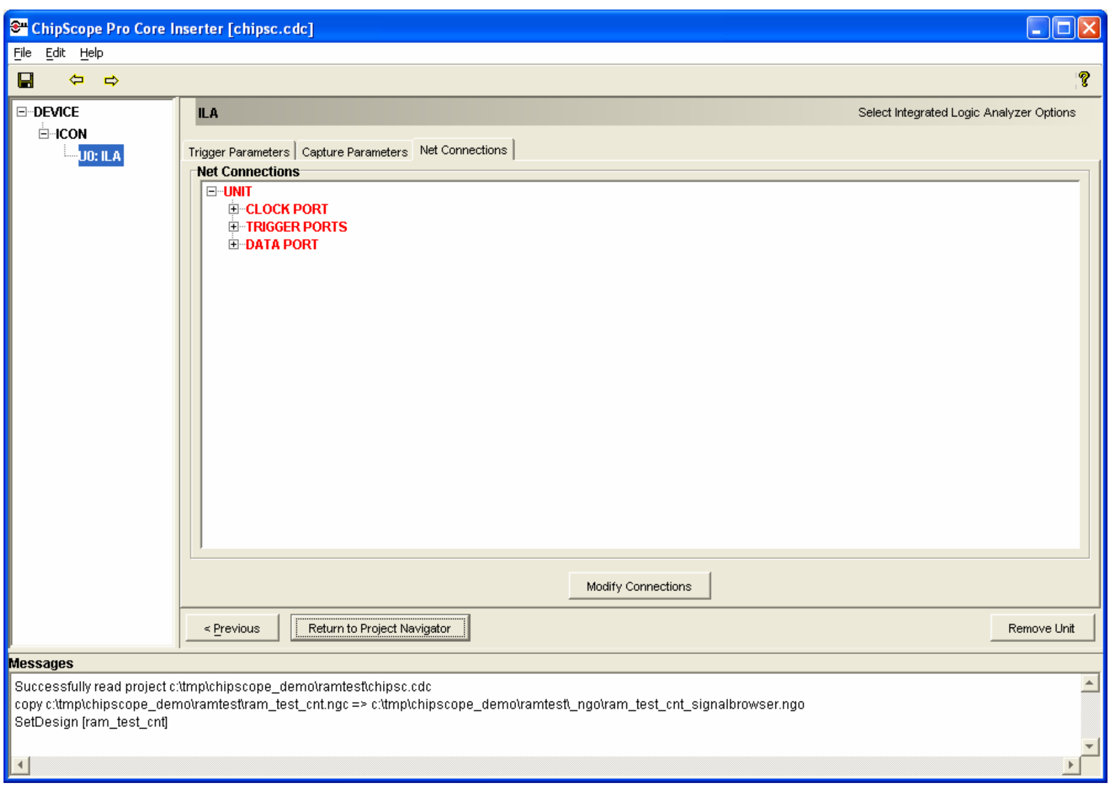
You choose the actual match values when you run. Right now you only need to choose the type of match you'll require. We'll just choose *Basic*. Note that you can have multiple match units in case you need very complex triggering.

The next screen is where you choose the signals to monitor. They can be only the trigger signals, or you can specify more signals.

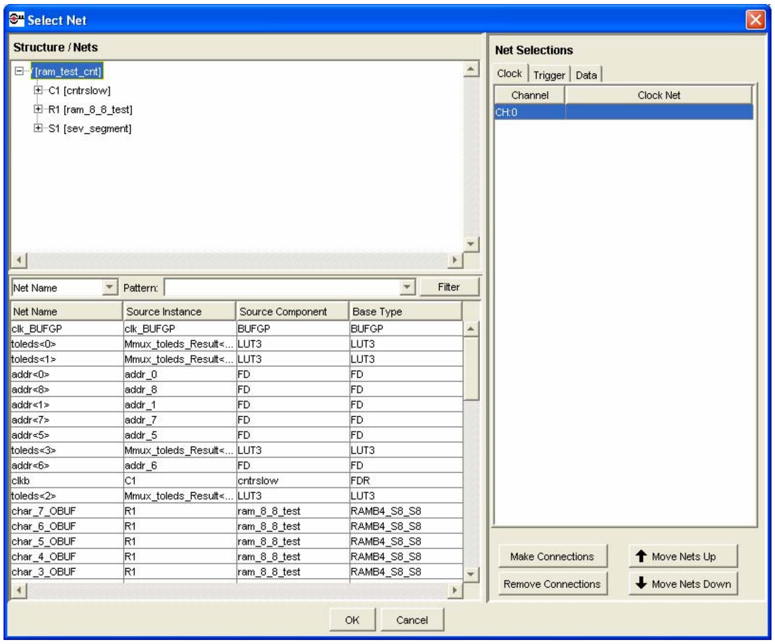
We'll include the memory data signals, so uncheck the *Data Same as Trigger* box. The *Data Width* is the number of signals you will monitor. Note that the number of Block RAMs used changes as you change this value. The Data Depth is the number of sets of samples that you'll be storing over time. This also affects the amount of RAM needed. We'll monitor 15 lines for 256 samples, using one Block RAM.



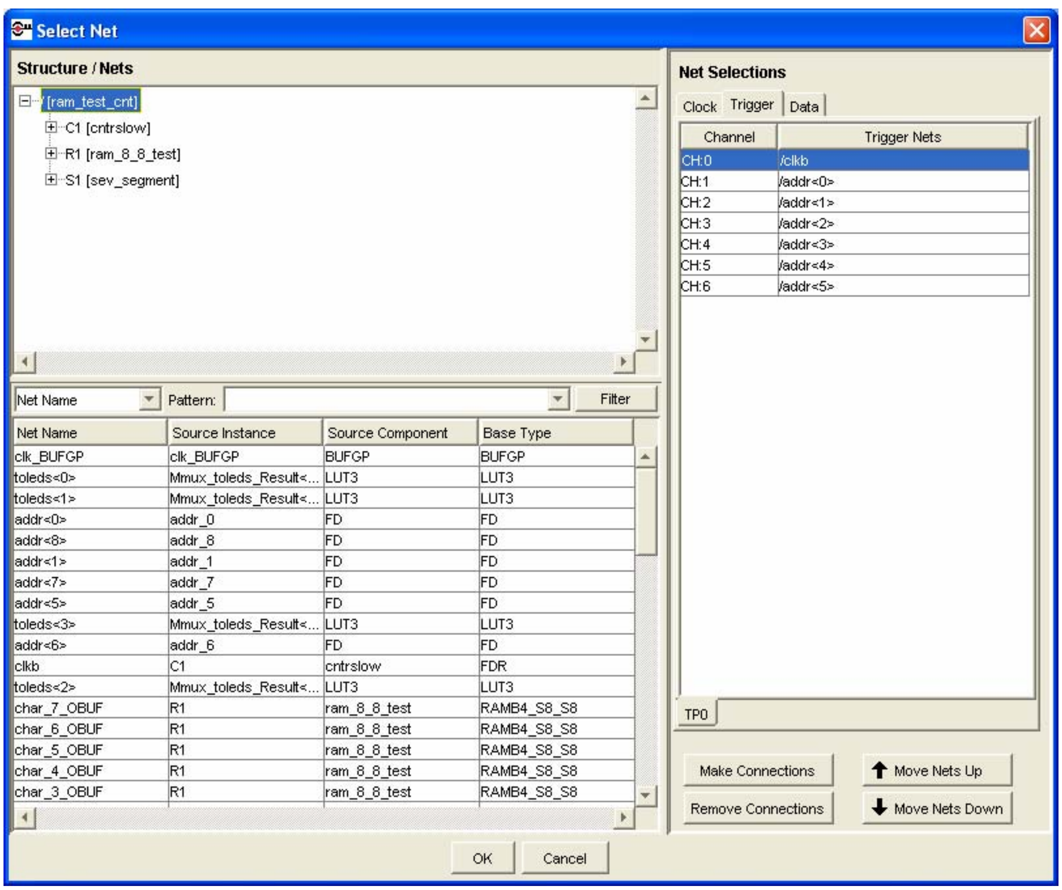
The next screen is where you choose the actual lines that will be monitored or will trigger. Select *Modify Connections*.



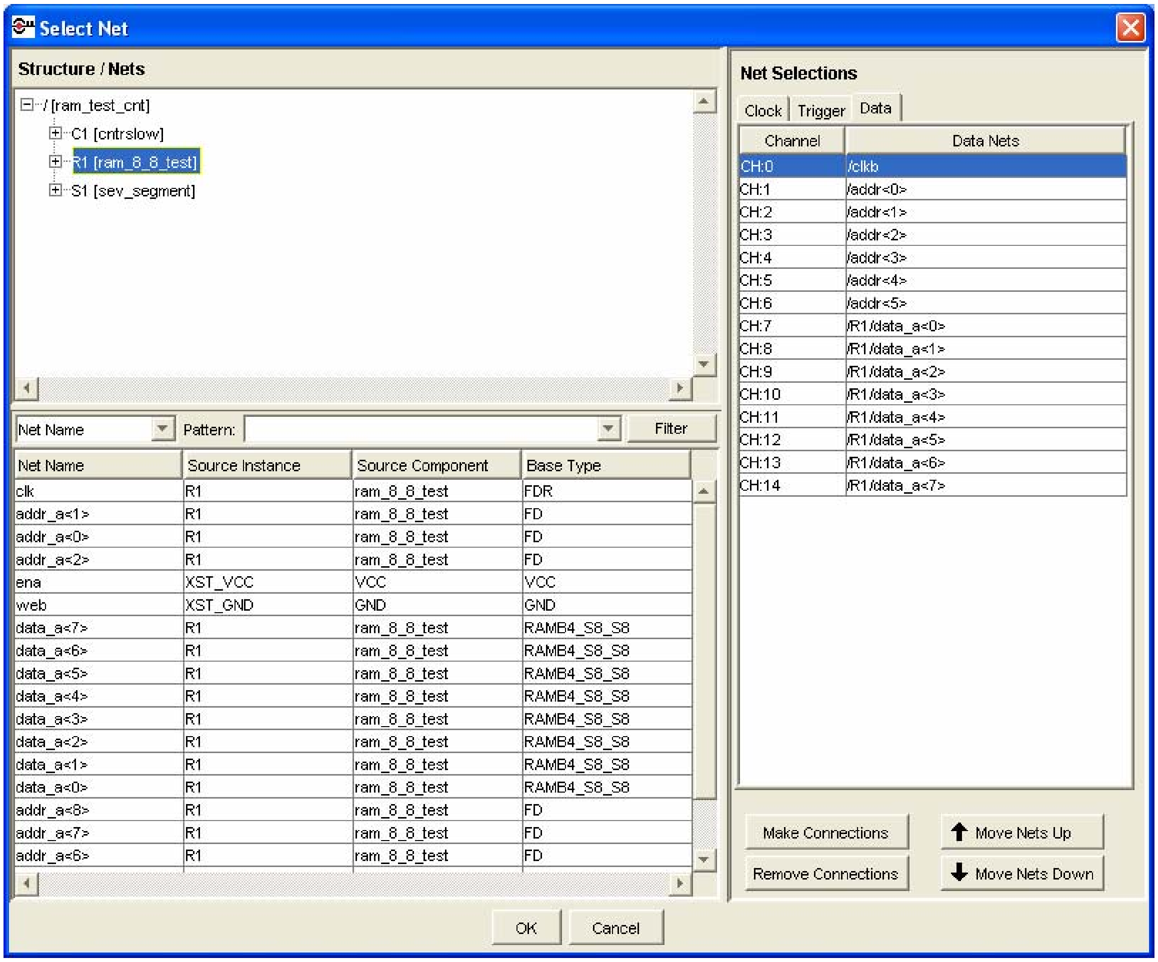
You'll see a window that allows you to associate design signals (left pane) with clock, trigger, and data of the logic analyzer.



You can select signals on the left and logic analyzer ports on the right, then click *Make Connections*. We'll use the basic clock, and trigger using the slow clock (clkb) and addr[5:0].



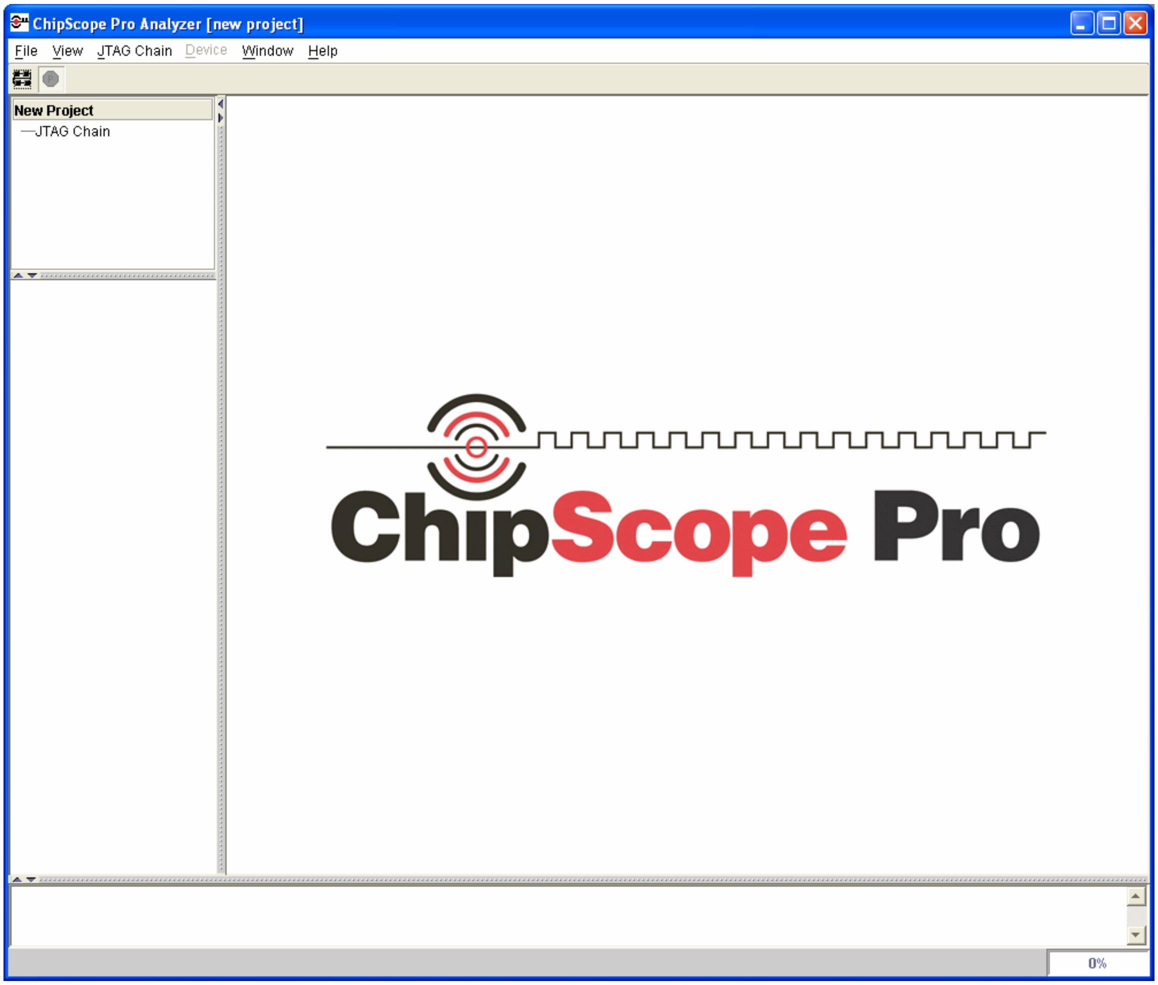
We'll monitor the slow clock, address lines, and also the eight data lines.



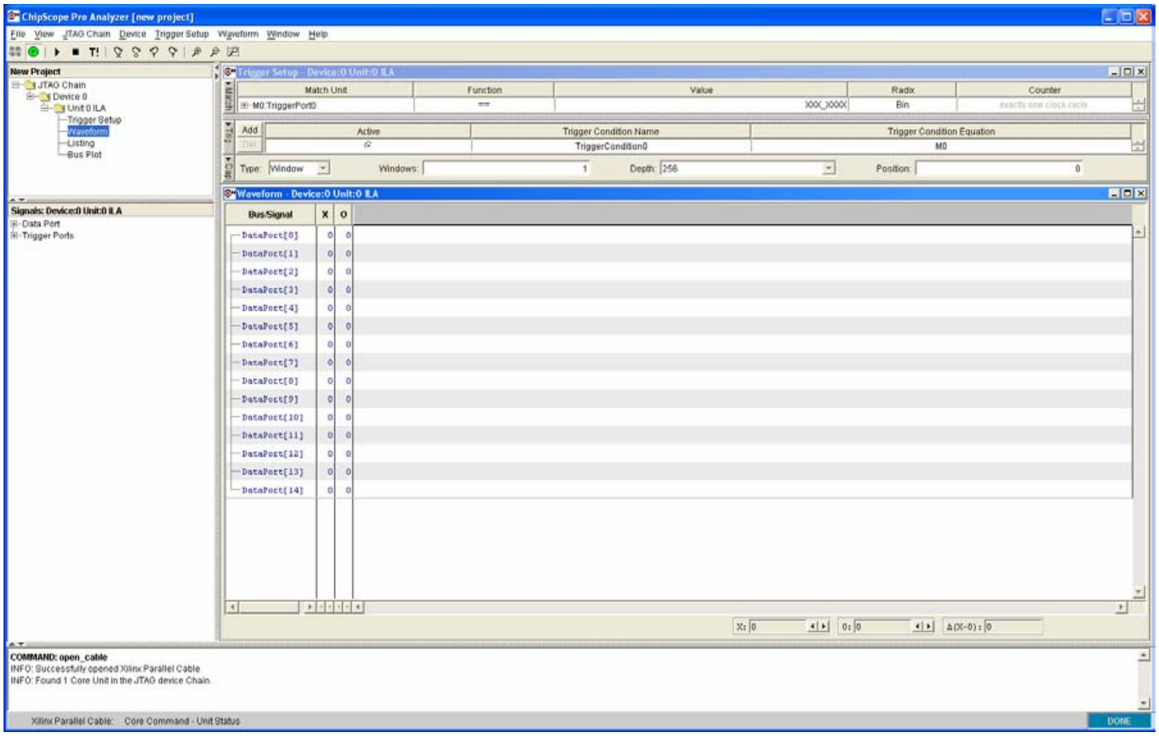
Click on the floppy-disk icon to save your settings.

Analyzer

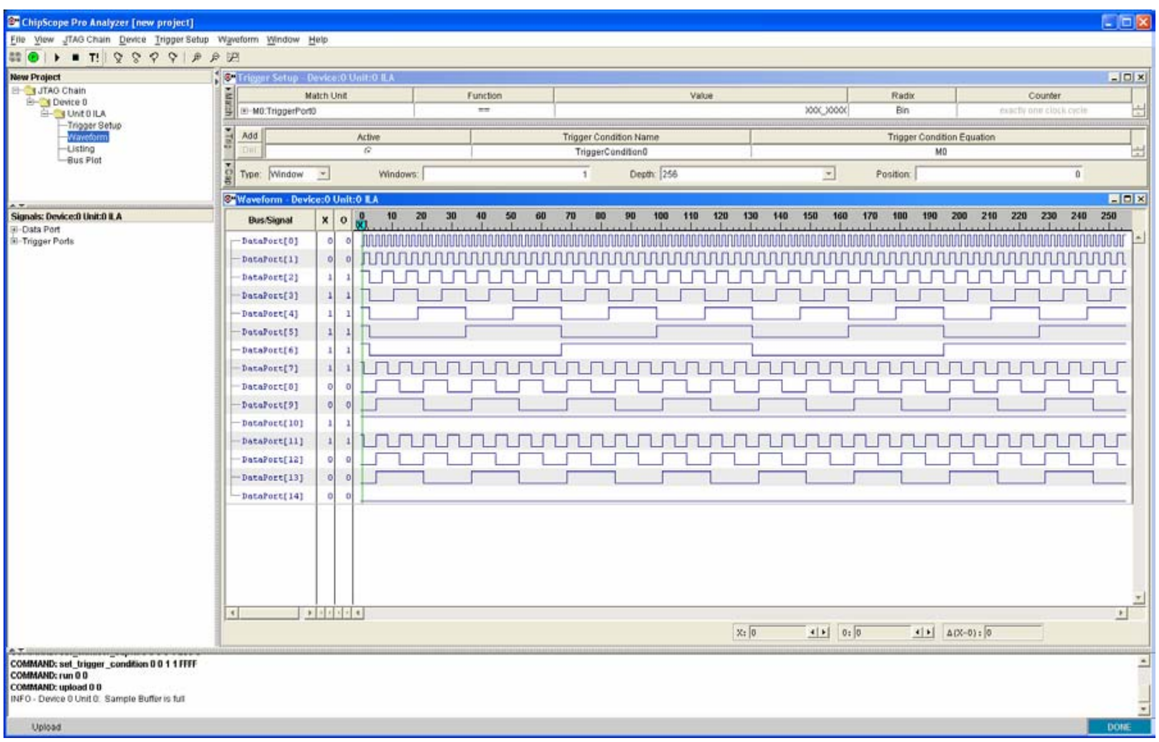
Download the design using iMPACT, and then double click on *Analyze Design Using Chipscope* in the *Process View* pane of the Project Navigator.



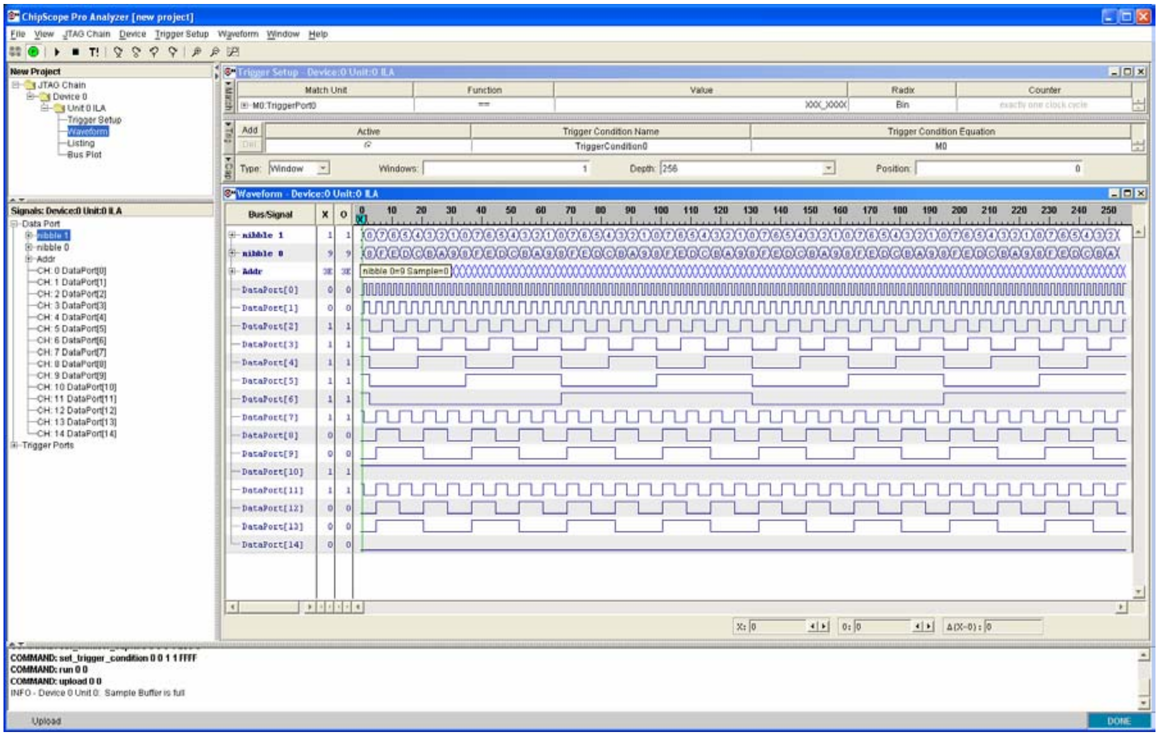
Click on the JTAG icon just below the File menu. This identifies the chip it sees in the JTAG chain. Just click OK.



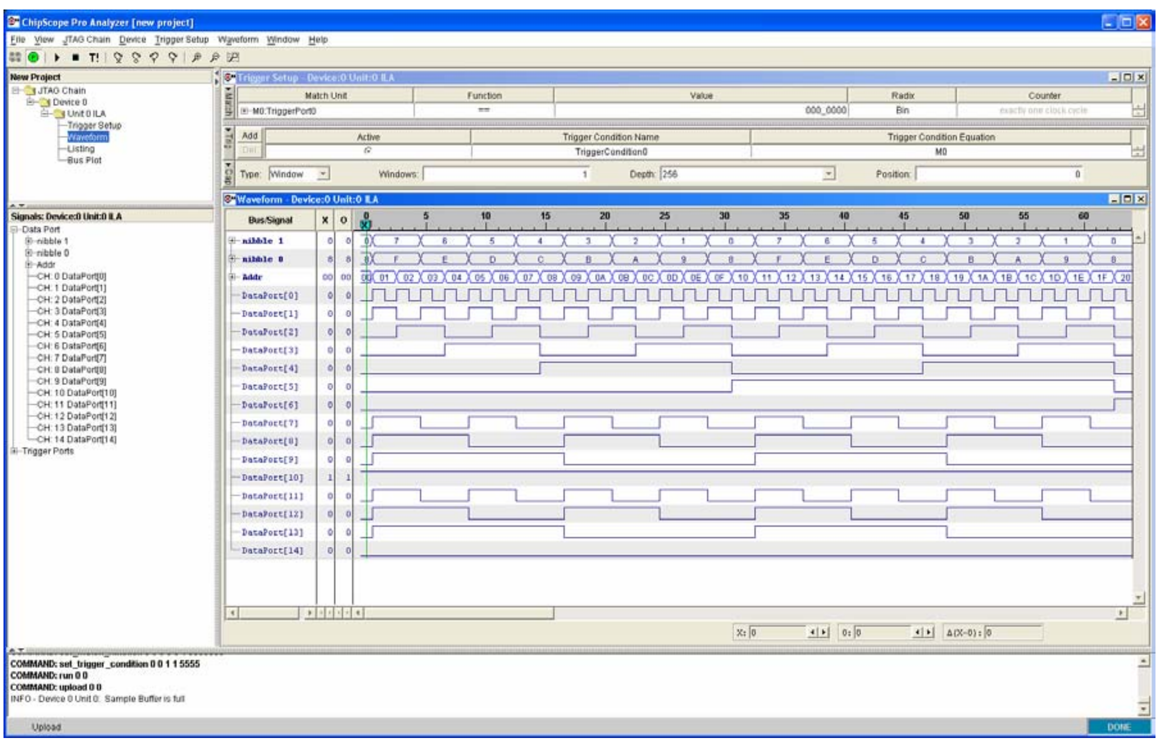
This window shows triggering information at the top, and waveform information below. You can gather data (without triggering) by clicking the **T!** icon.



You can rename the signals to something project specific, and also assign colors. However, the most important tool provides the ability to group them into buses. In our case, we can group the address into one bus, and the two nibbles of data into another two buses, then rename.



We can set the trigger to 0, and then press the triangle on the control bar to arm the trigger. This will collect another 256-sample buffer of data that begins with the address at zero. Also zoom.



We can move the X and O markers in a manner similar to the Acute logic analyzer, except here the units are clock ticks because the capture clock of this analyzer is whichever on-chip clock we've specified.