UNIVERSITY OF LIVERPOOL

Digital Systems Design ELEC373/473

Logic Analysers for Digital Circuit Debugging

Prof J.S. Smith
Room A515;
E-mail: j.s.smith@liv.ac.uk

What is a Logic Analyser?

- A tool to examine how a digital system operates
  - Creates waveforms to visualize output
- Can find system errors and instabilities
- Better at analysing digital systems than an oscilloscope
  - An oscilloscope will show continuous voltage
    - Most new oscilloscopes are now digital and show a sampled voltage from an ADC.
  - A logic analyser shows discrete logic levels

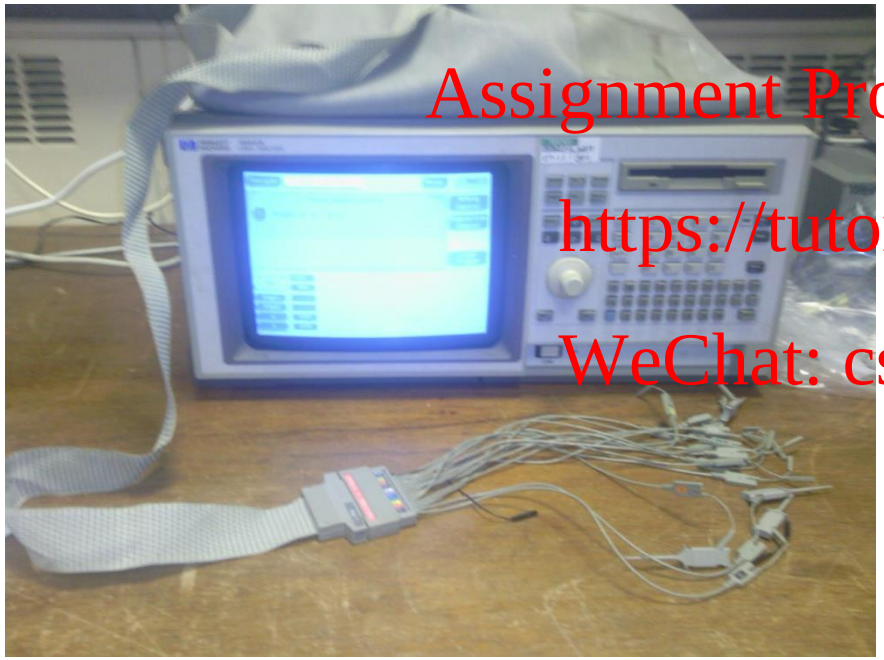
Logic Analyser vs. Oscilloscope

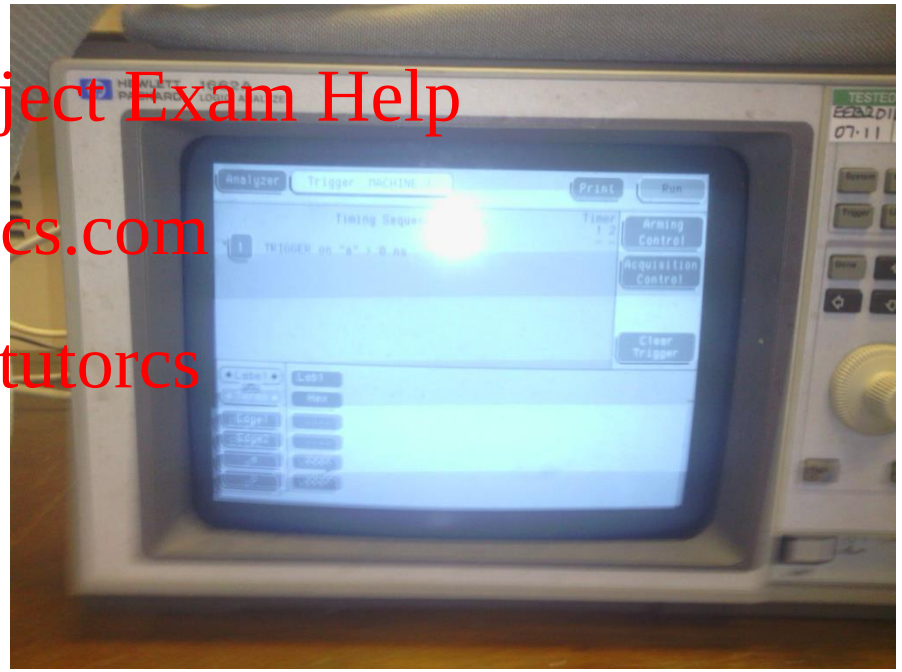
- A digital oscilloscope samples incoming signal at regular time intervals.
  - Stores sample amplitude as 8/10/12-bit digital value.
  - Preserves amplitude information
  - Signal is analogue in nature
- Logic Analyser monitors multiple channels simultaneously
  - Signal is digital in nature
  - Provides timing relationship information
- Both are powerful analysis and troubleshooting tools

Oscilloscope or logic analyzer?

- ***When to use a scope:***
  - When it is required to observe small voltage excursions on the signal.
  - When high time-interval accuracy is required.
- ***When to use a logic analyzer:***
  - When one wishes to observe many signals at the same time.
  - When it's necessary to look at signals in the system the same way hardware does.
  - When it's required to trigger on a pattern of highs and lows on several lines and see the results.

HP (Agilent) Logic Analysers

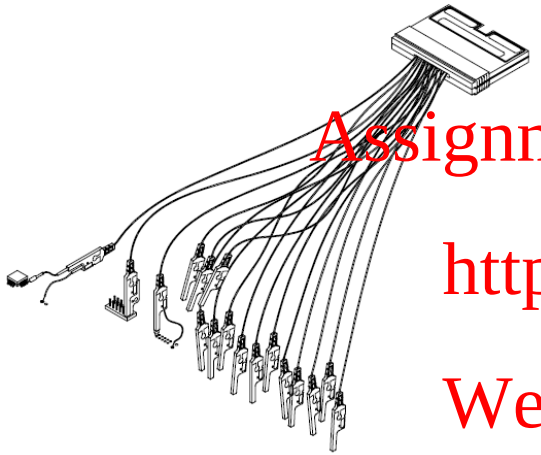




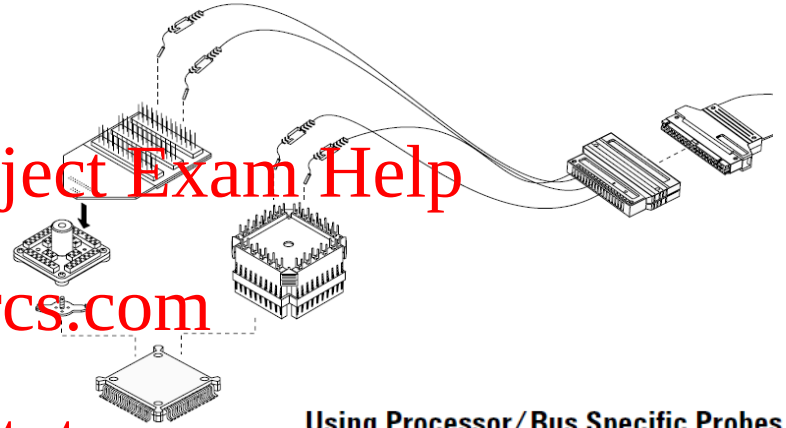
HP 1662A Logic Analyser 68 Channels 250MHz Sampling

Probes for Logic Analysers

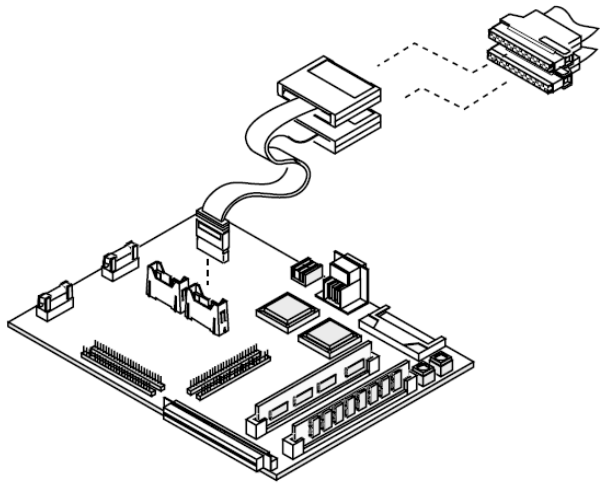
Connecting to Individual IC Pins, Test Points, Browsing or Solder Attach to Components, Traces or VIAs



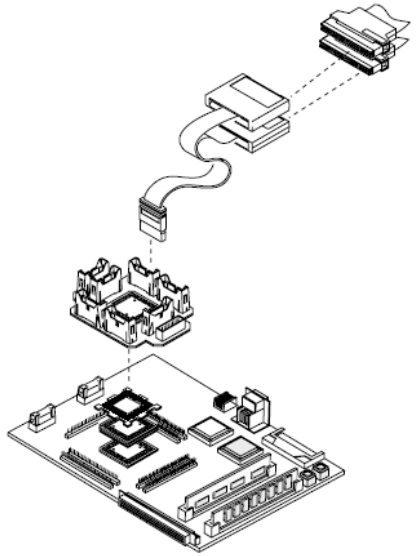
Connecting to all the Pins of a Specific Package



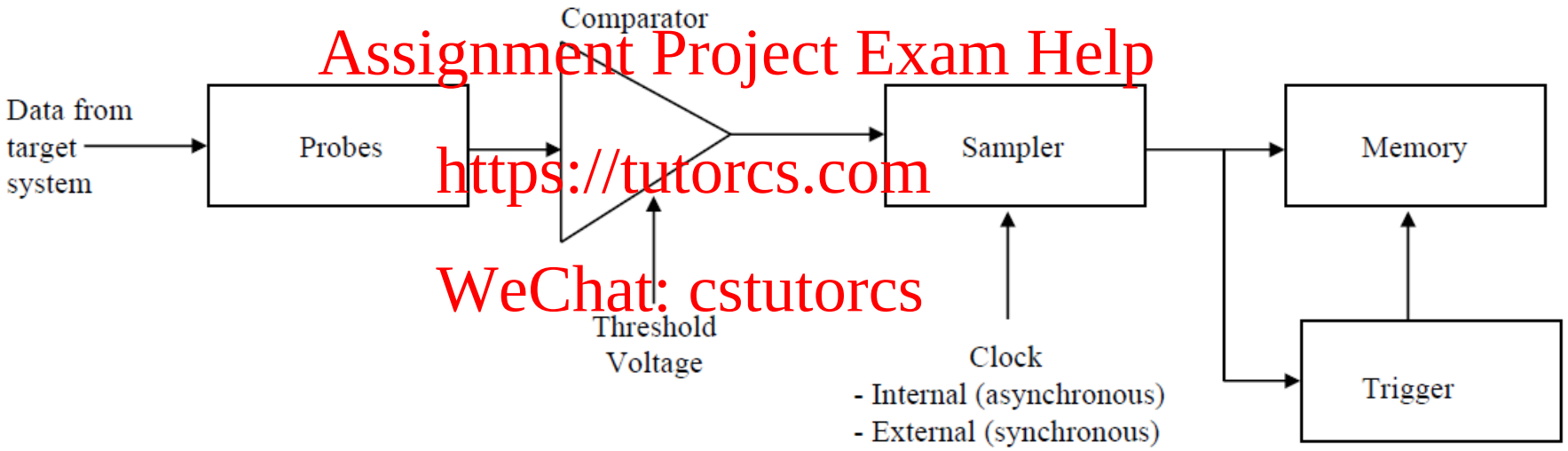
Designing Connectors Directly into the Target System



Using Processor/Bus Specific Probes



Logic Analyser Operation



The threshold voltage is normally programmable

Logic Analyser Operation (2)

- Clocks control when data is sampled
- Types of clocking
  - External (synchronous) - listing data
    - clock source external to Logic Analyser
    - useful for obtaining state aspects of data
  - Internal (asynchronous) waveform data
    - data may be lost between clocks
    - asynchronous provides all data
    - useful for obtaining timing aspects of data

Logic Analyser Operation (3)

- Acquiring data
  - Logic Analyser samples data from probes
  - Sample taken when clock occurs
- Triggering
  - Trigger program looks at sampled data for specific events and then takes specified action
  - Also provides storage qualification conditions.
    - If met, allows data to be stored in memory

Waveform Data Concepts

- Use Logic Analyser to observe timing relationship between signals
- All waveforms are time-aligned horizontally and displayed in same time per division
- Can display waveforms or magnitude mode.
  - Provides hexadecimal value of multi-channel bus

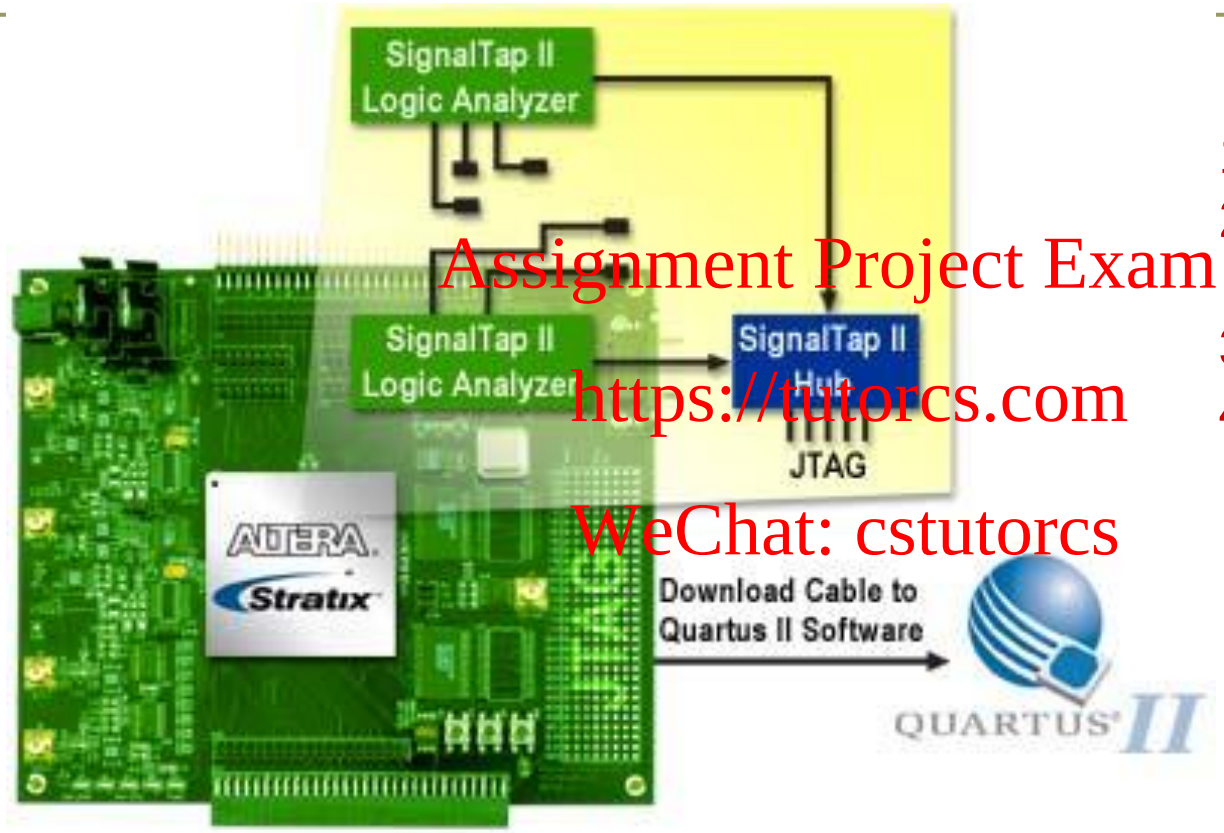
Waveform Data Concepts (2)

- Sampling resolution
  - Waveform accuracy depends on sample clock rate used to record incoming signals (remember Nyquist Criteria !?)
  - If sampled too slow, aliasing will occur
- Logic Analyser has fixed memory
  - Trade-off between resolution of recorded signal and its duration
  - A faster sample clock will record a smaller portion of the signal

Altera's SignalTap II ELA (Embedded Logic Analyser)

- Captures the logic state of FPGA internal signals using a defined clock signal
- Gives designers the ability to monitor buried signals
- Connects to Quartus II through FPGA JTAG pins
- Captures real-time data
  - Up to 200 MHz

How Does It Work?



1. Configure ELA
2. Download ELA into FPGA along with Design
3. ELA Samples Internal Signals
4. Quartus II Communicates with ELA through JTAG

Cyclone Resource Usage

Number of Channels	Logic Elements		
	Trigger Level 1	Trigger Level 2	Trigger Level 3
8	316	371	426
32	566	773	981
256	2900	4528	6156

33,216 Logic Elements on the EP2C35 on your DE2 board

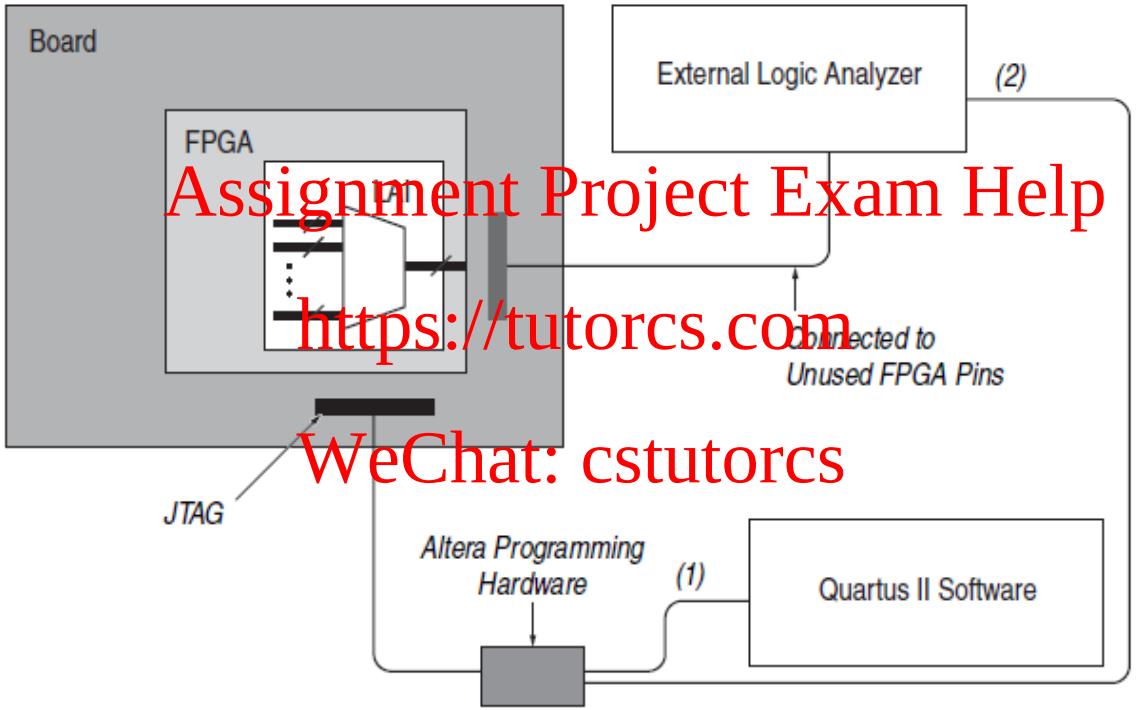
Number of Channels	M4Ks Based on Sample Depth				
	256	512	2K	8K	32K
8	< 1	1	4	16	64
32	2	4	16	64	256
256	16	32	128	512	

105 M4K RAM Blocks on the EP2C35 on your DE2 board

Modes of Operation

- Three different configurations
  - Internal RAM ELA configuration
  - Debug port ELA configuration
  - Hybrid approach
- Provides flexibility based on available device resources
  - Memory resources are limited
    - Use Debug port configuration
  - Pin resources are limited
    - Use internal RAM configuration

Logic Analyser Interface (LAI)



Uses external Logic Analyser with signal multiplexed to reduce pin count.

SignalTap II Key Features

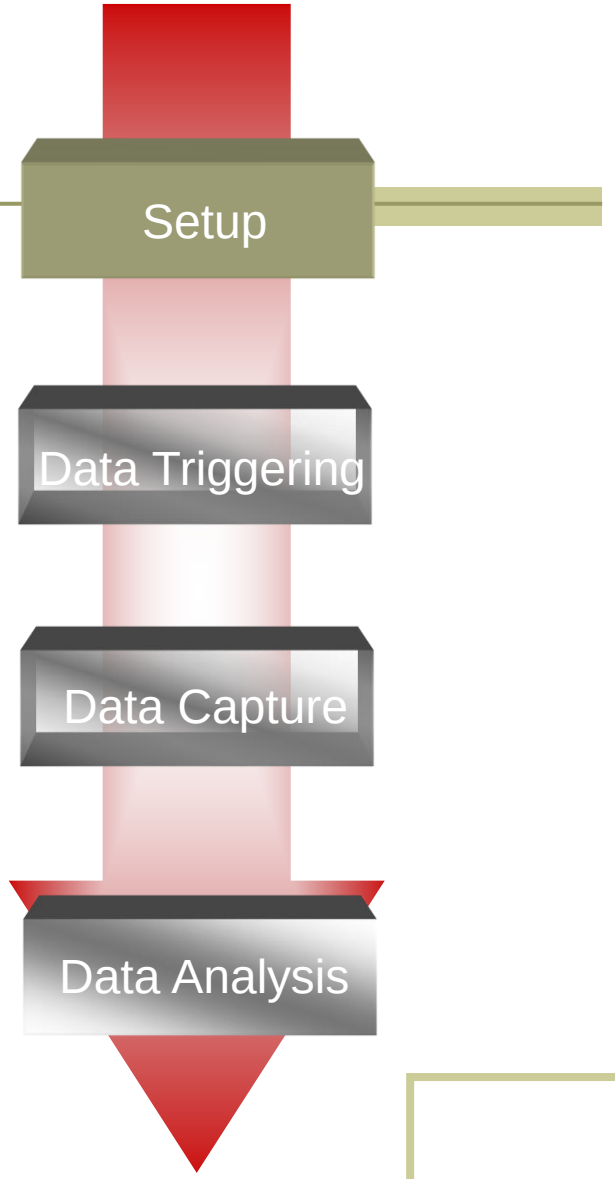
- Setup
- Data Triggering
- Data Capture
- Data Analysis

Setup Features

- Up to 1024 Data Channels
- Multiple analysers in one device
  - Supports analysis of multiple clock domains
  - Each analyser can run simultaneously

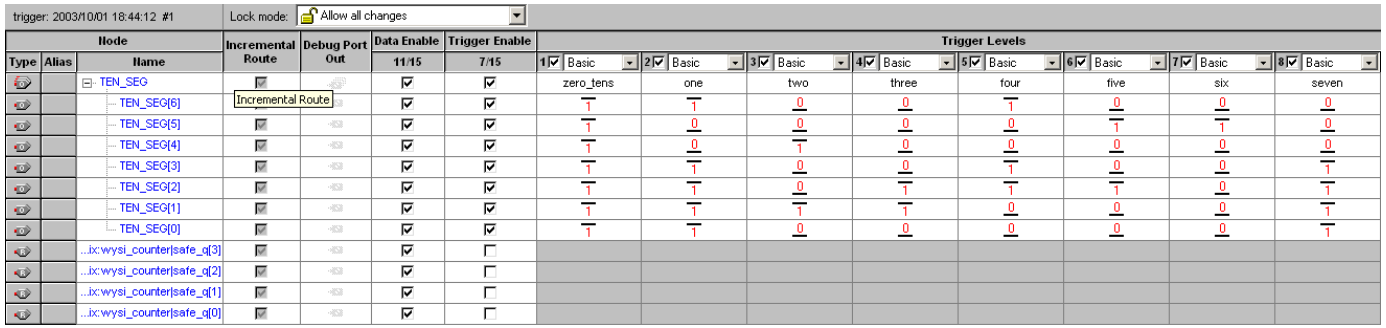
Instance	Status	LEs: 1183	Memory: 36096	Small: 0/0	Medium: 9/105	Large: 0/0
ELA_Coun24	Not running	546 cells	32768 bits	0 blocks	8 blocks	0 blocks
ELA_Coun24_1	Not running	637 cells	3328 bits	0 blocks	1 blocks	0 blocks

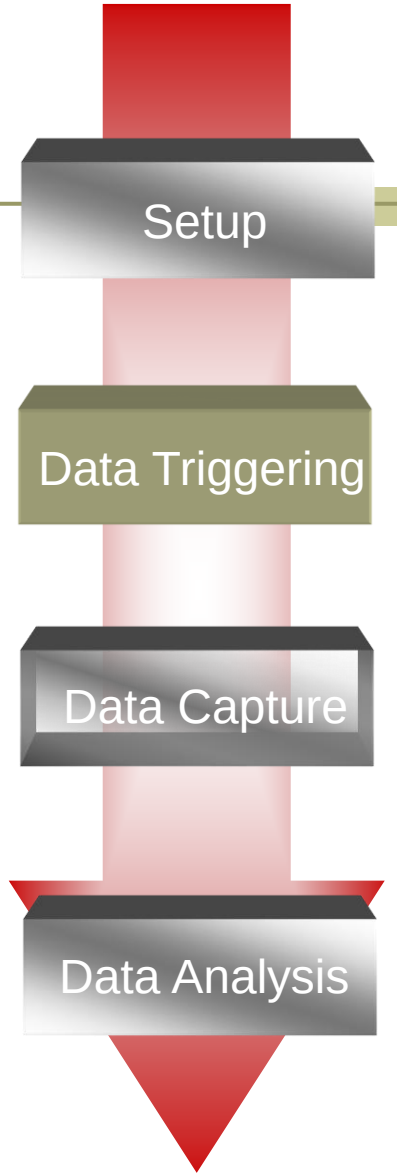
- Resource usage estimation



Data Triggering Features

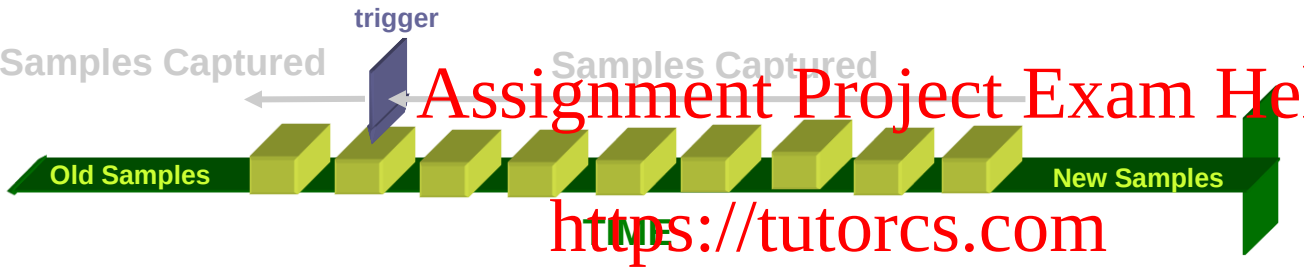
- Up to 10 trigger levels per channel
  - Allows application of simple (Basic) & complex (Advanced) triggering schemes
    - Defines a sequential pattern of logic conditions
  - Each trigger level is logically **ANDED**
    - If (L1 then L2 ... then L10) == TRUE → Data Capture



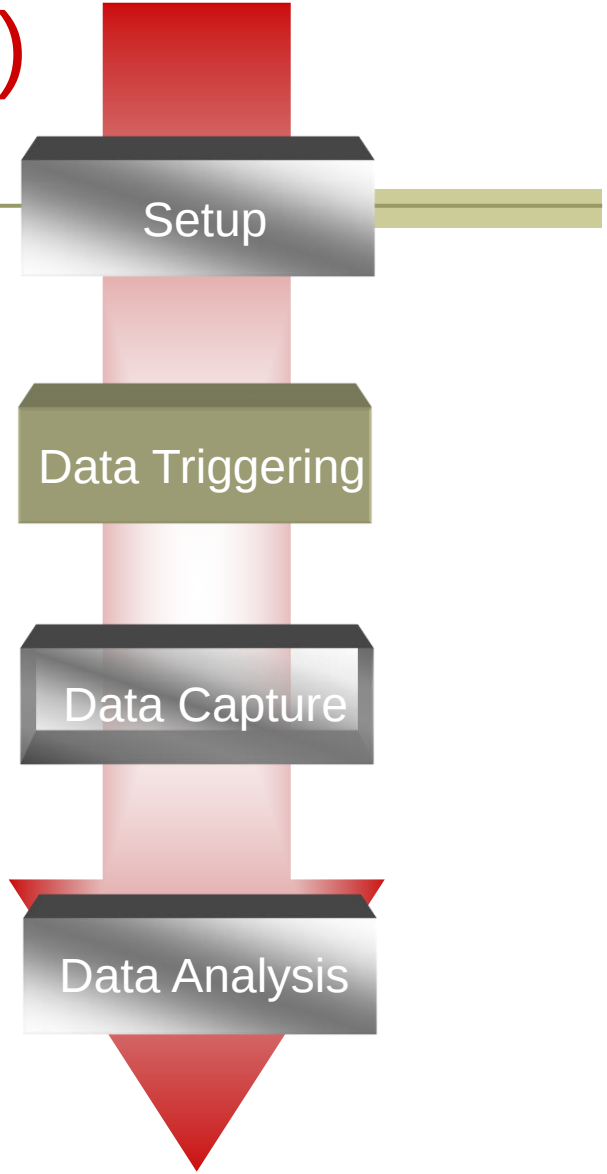


Data Triggering Features (Cont.)

- Three main trigger positions

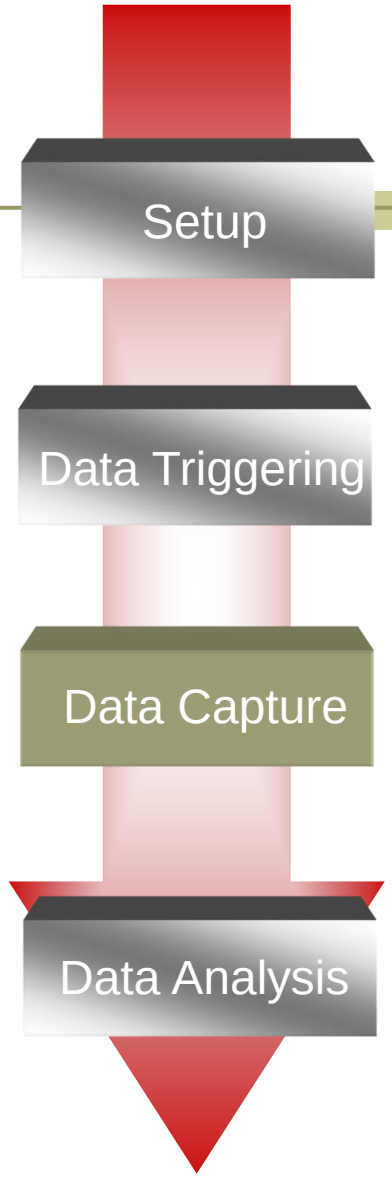


- Trigger input
  - Setup External Trigger to Trigger the Analyser
- Trigger output
  - Signifies Trigger Event Occurred with SignalTap II
- Use one ELA's trigger output as trigger input for another ELA



Data Capture Features

- Up to 128K samples per channel
  - Increases chance of catching target event
- Two methods of data acquisition
  1. Circular
  2. Segmented
- Mnemonic Tables
  - Create user-defined labels for bit sequences (Ex. State Machine)

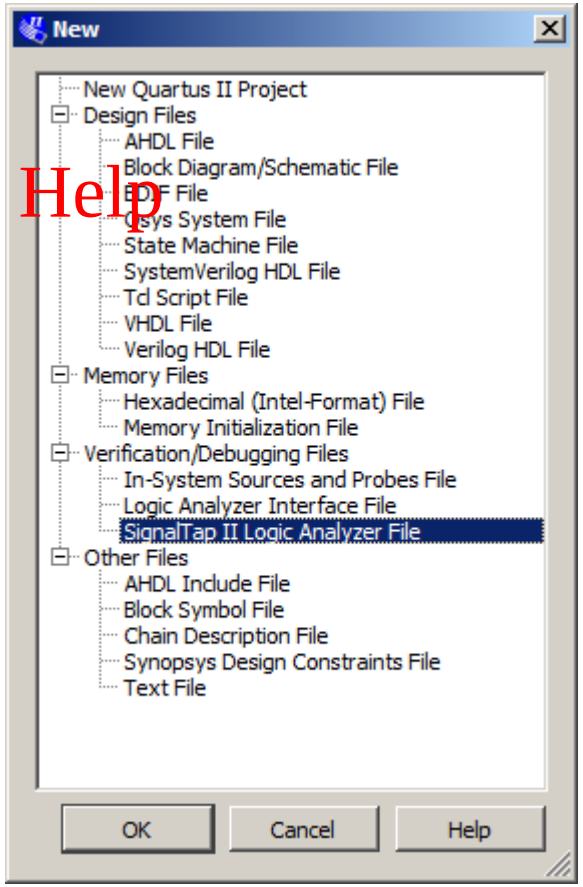


Using STP File

1. Create .STP File
   - Assign sample clock
   - Specify sample depth
   - Assign signals to STP file
   - Specify triggering
   - Setup JTAG
2. Save .STP File & compile with design
3. Program device
4. Acquire data

1) Creating a New .STP File

- To Create a .STP File
  - **Select New** (File Menu)
  - **Verification/Debugging Files**
  - **SignalTap II** Logic Analyser **File**
- Default file name will be STP1.stp

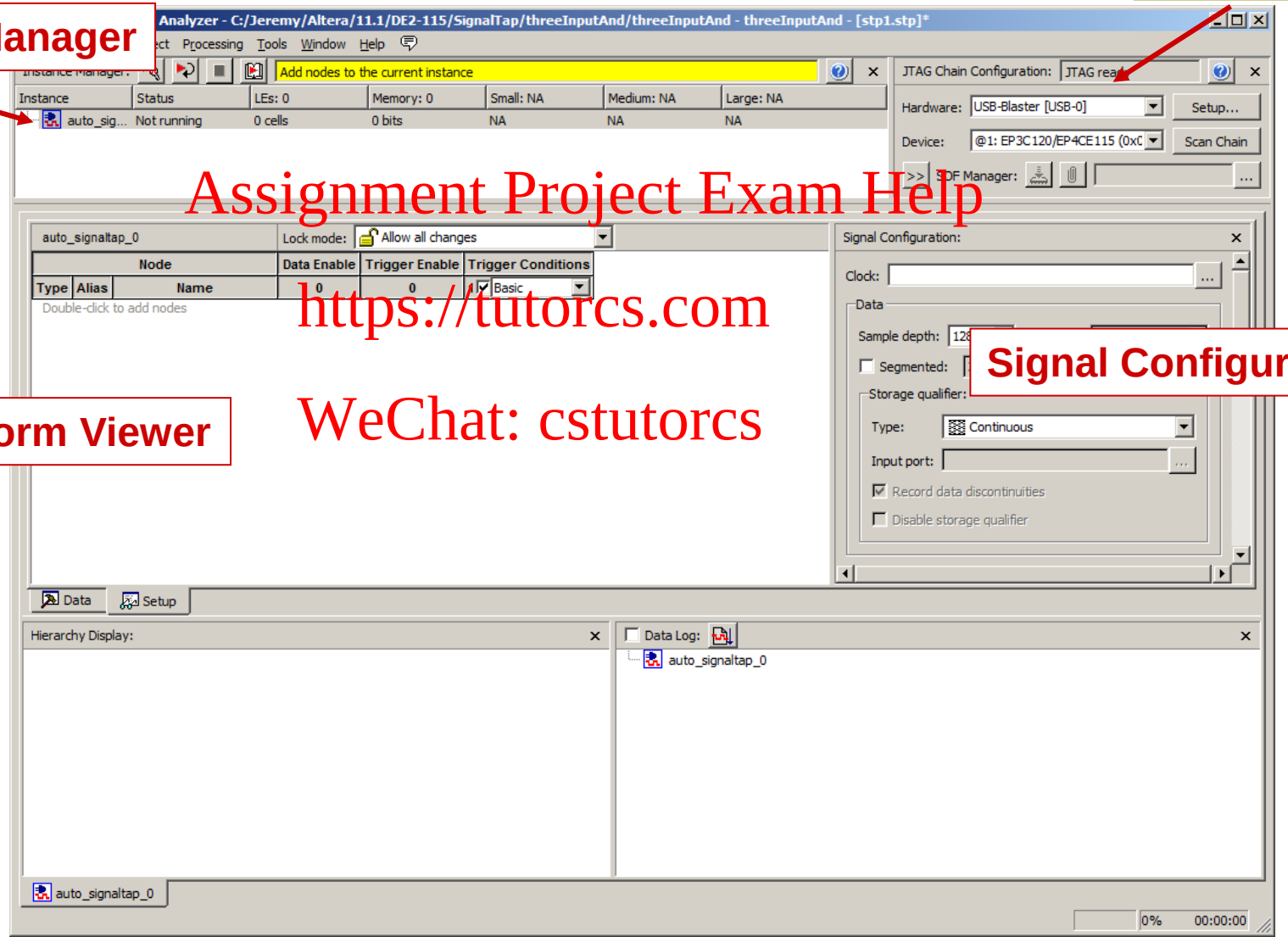


Main .STP File Components

.STP File

JTAG Chain Configuration

Instance Manager



Signal Configuration

Waveform Viewer

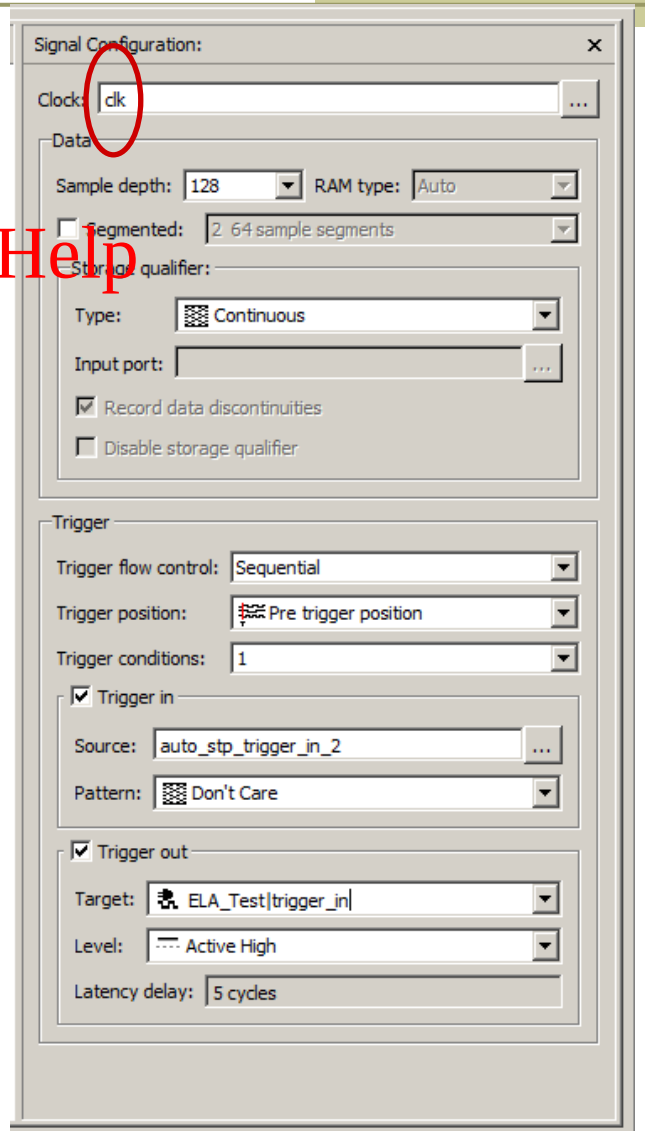
Instance Manager

- Instance Manager
  - Selects current ELA to Setup/View
  - Displays the current status of each instance
  - Displays size (resource usage) of ELA

Instance	Status	LEs: 1346	Memory: 37376	Small: 0/0	Medium: 10/105	Large: 0/0
ELA_Coun24	Not running	533 cells	32768 bits	0 blocks	8 blocks	0 blocks
ELA_Test	Not running	441 cells	4096 bits	0 blocks	1 blocks	0 blocks
ELA_high_bits	Not running	372 cells	512 bits	0 blocks	1 blocks	0 blocks

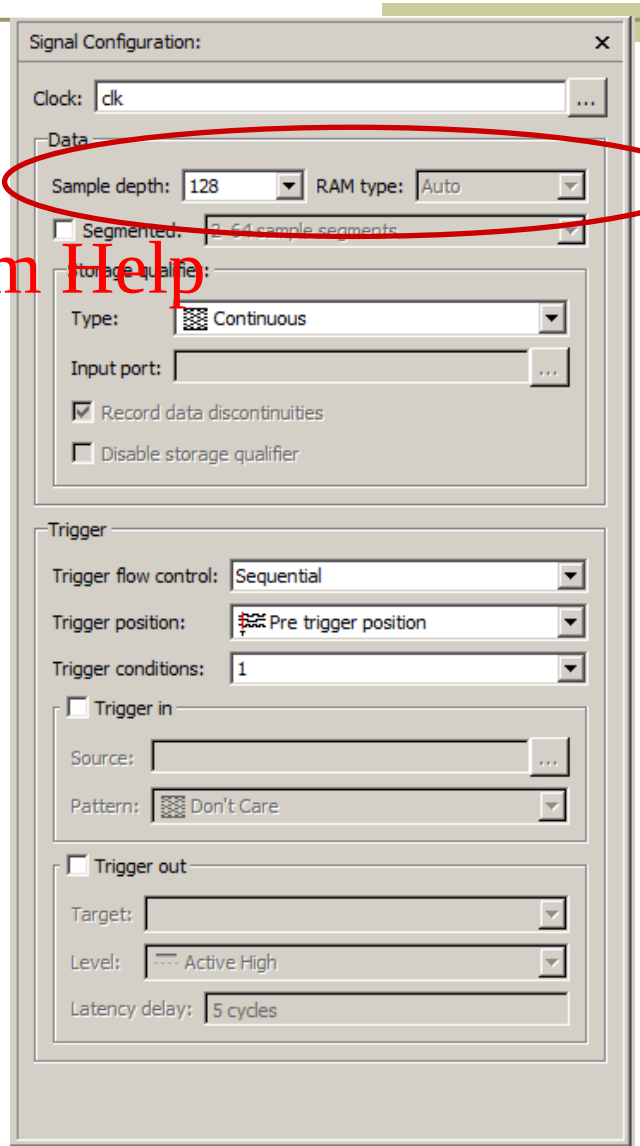
Assign Sample Clock

- Use global clock for best results
- Data written to memory on every sample clock rising edge
- Clock signal cannot be monitored as data
- External Clock pin created automatically if clock unassigned
  - auto_stp_external_clock
  - ELA expects external signal to be connected to clock pin



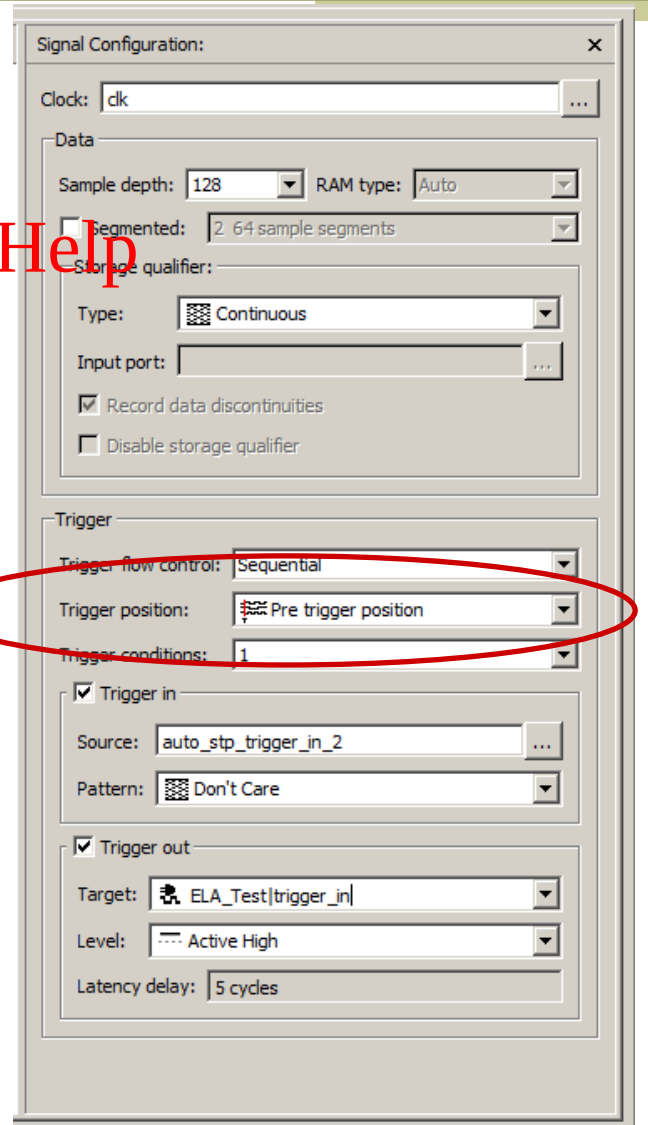
Specify Sample Depth

- Sample Depth
  - Set number of samples stored for each data signal
  - 0 to 128K sample depth
    - 0 selected when external analyser is used
- Select RAM type for Stratix & Stratix II Devices
  - Useful when preserving a specific memory type is necessary



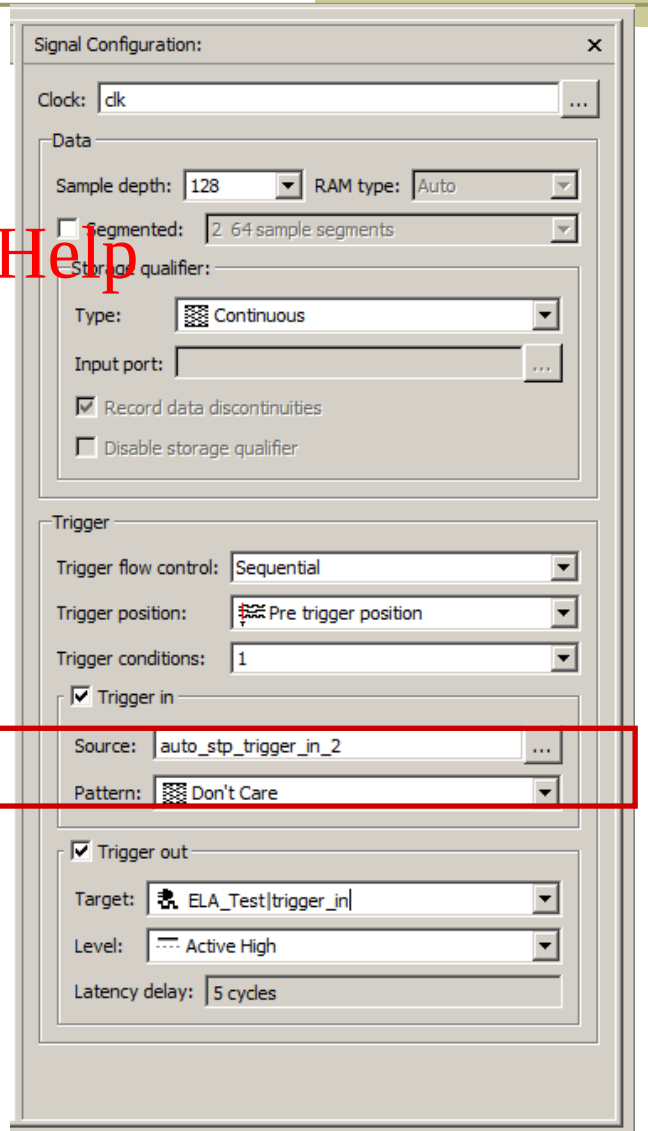
Data Capture

- Specify Trigger Position
  - Pre
  - Center
  - Post
- Segmented
  - Specify Segment Depth



Triggering

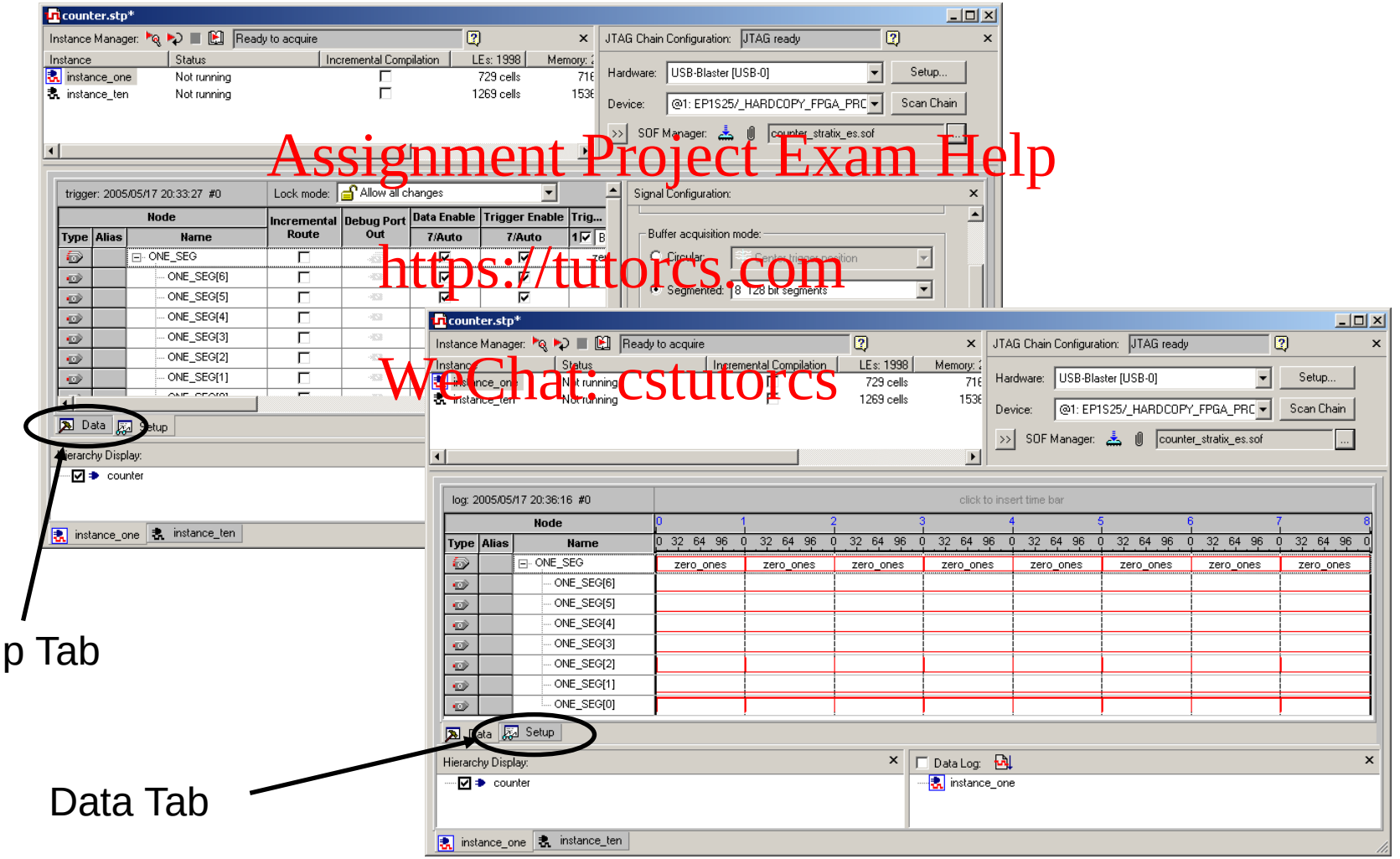
- Trigger levels
  - Indicate up to 10 trigger conditions
- Trigger-In
  - Any I/O pin can trigger the SignalTap II Analyser
  - Generates auto_stp_trigger_in_n Pin
- Trigger-Out
  - Indicates when a trigger pattern occurs
  - Delayed 5 clock cycles after actual trigger event



Waveform Viewer

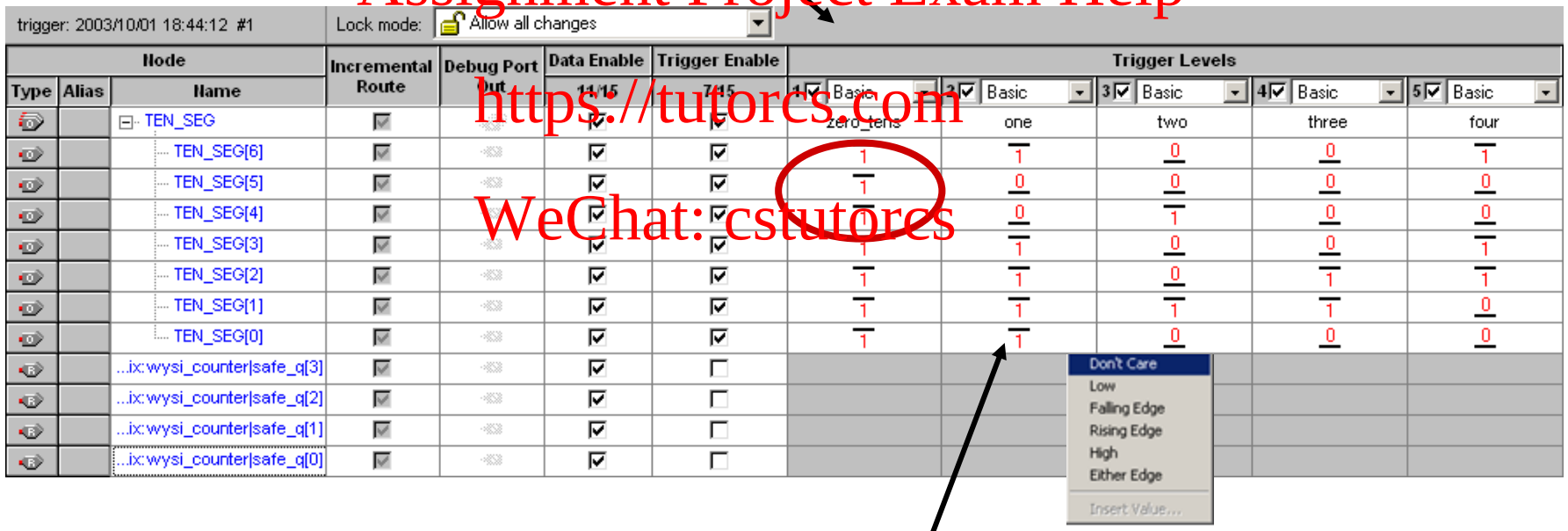
- Setup Tab Describes the Signal Settings
  - Data Signals vs. Trigger Signals
  - Sets up Each Triggering Level (L1 – L10)
- Data Tab Displays Captured Data

STP File Waveform Viewer



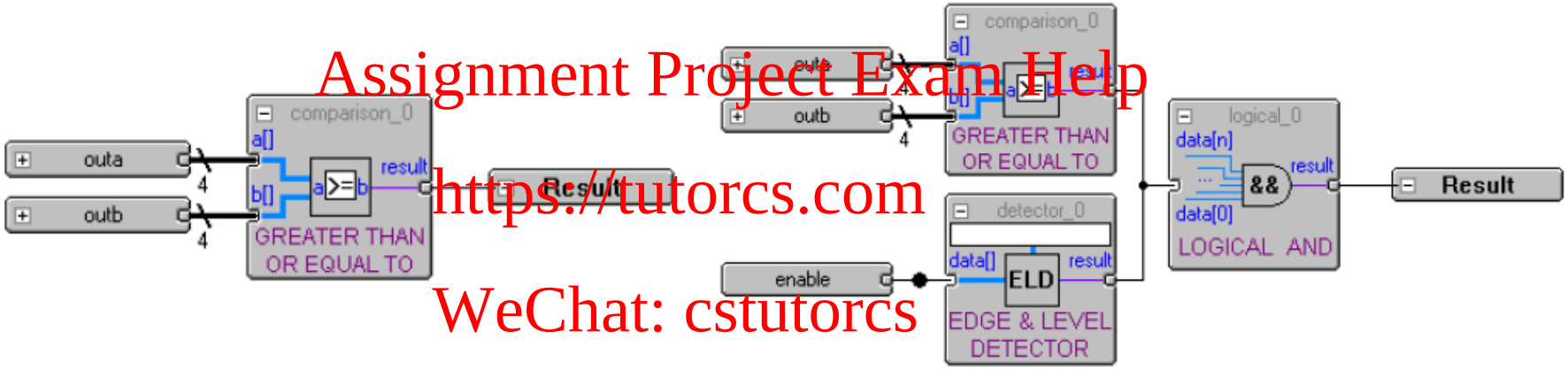
Basic Triggering

All signals must be true for level to cause data capture



Right-Click to Set Value

Advanced Triggering

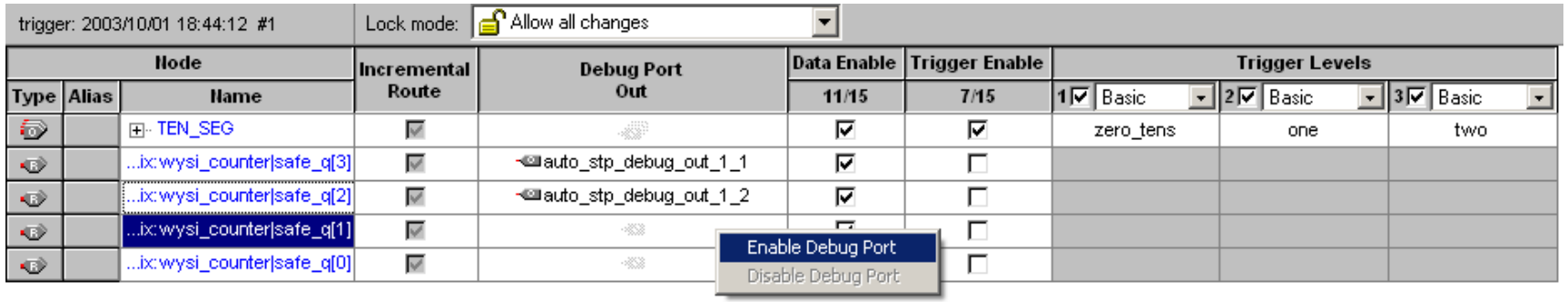


Bus outa is Greater than or Equal to Bus outb

Bus outa is Greater than or Equal to Bus outb and enable has rising edge

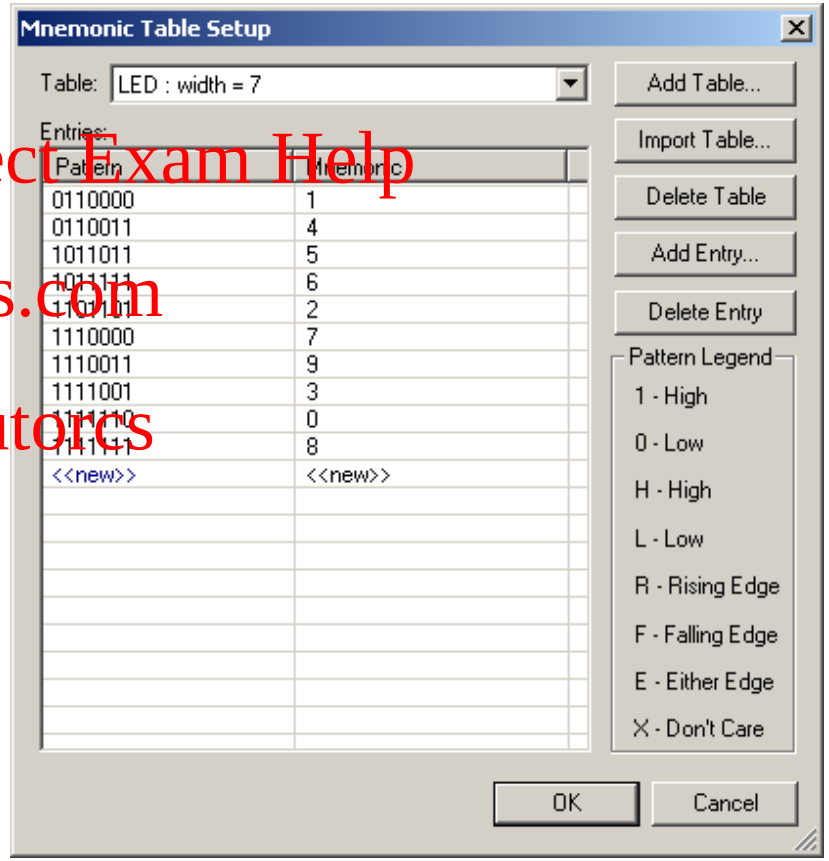
Debug Port

- Routes data signals to spare I/O pins for capture by external Logic Analyser
- Quartus II Automatically Generates auto_stp_debug_out_m_n Pin
  - m Represents the Instance Number of the Analyser
  - n Represents the Order the Debug Port Pin Occurs in the Signal List



Mnemonic Table

- Allows a set of bit patterns to be assigned user-defined names
  - Right-Click in the setup view of an STP file & select Mnemonic Setup
  - Select Add Table
  - Select Add Entry
- Ex. State Machines or Decoders/Encoders

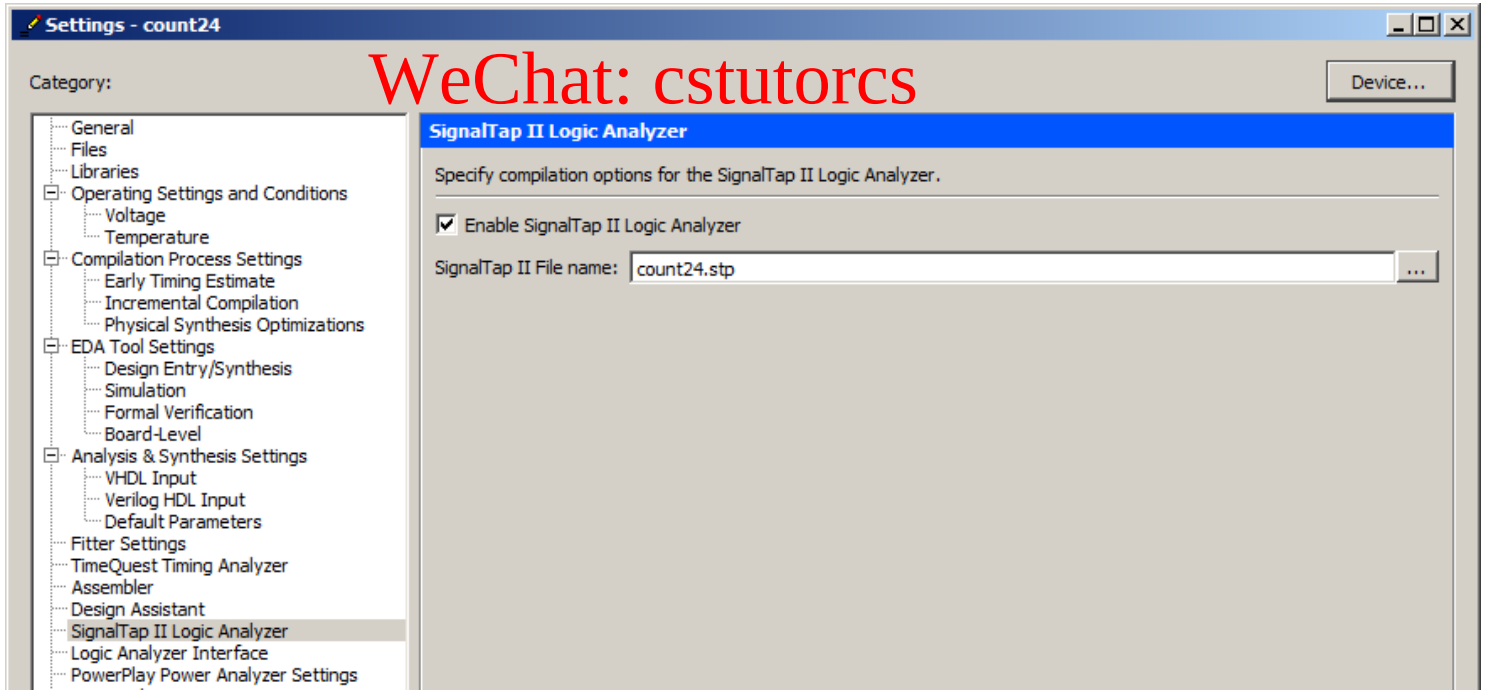


JTAG Chain Configuration

- Select programming hardware
- Scan Chain button automatically determines devices physically connected to the chain
  - Detects Non-Altera devices & displays them as unknown

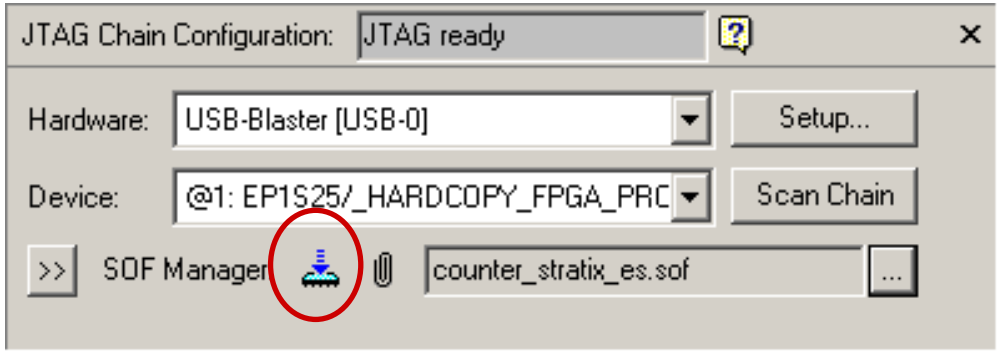
2) Save .STP File & Compile

- SignalTap II Logic Analyser control in Compiler Settings
  - Assignments ⇒ Settings
  - Specify the STP File to Compile with Project



3) Program Device(s)

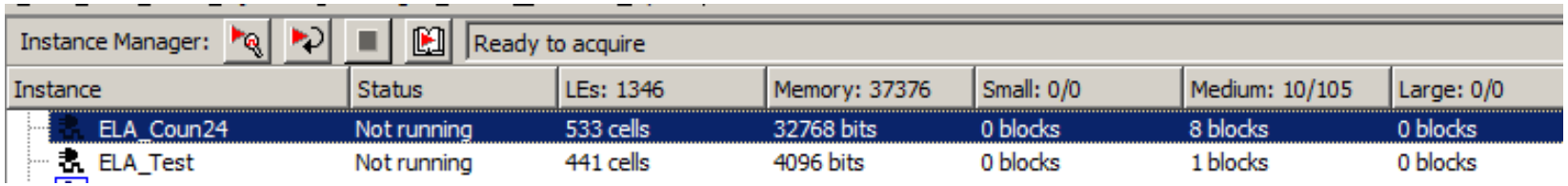
- Use Quartus II Programmer or STP File
  - Program Button in the SignalTap II interface only configures the selected device in the chain
  - Use Quartus II Programmer to program multiple devices
    - Can create a STP file for each device in the JTAG chain



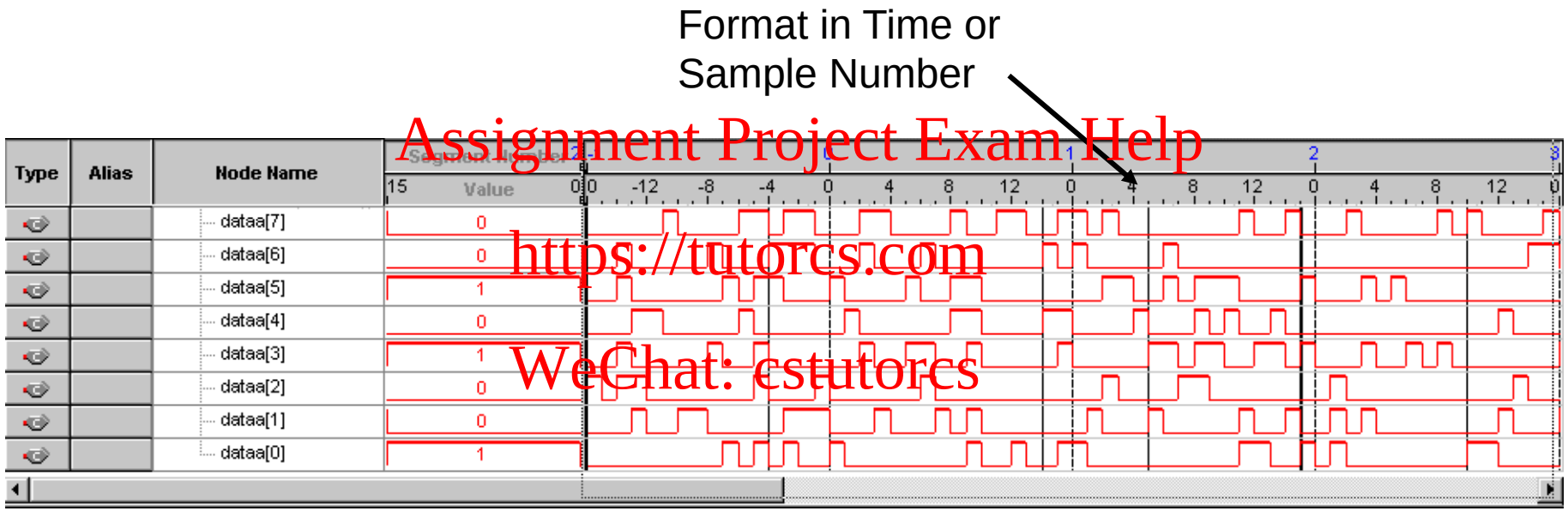
4) Acquire Data

- SignalTap II Toolbar & STP file controls

  - Run
  - Autorun
  - Stop
  - Read Data (Reads in Data from Last Analysis)

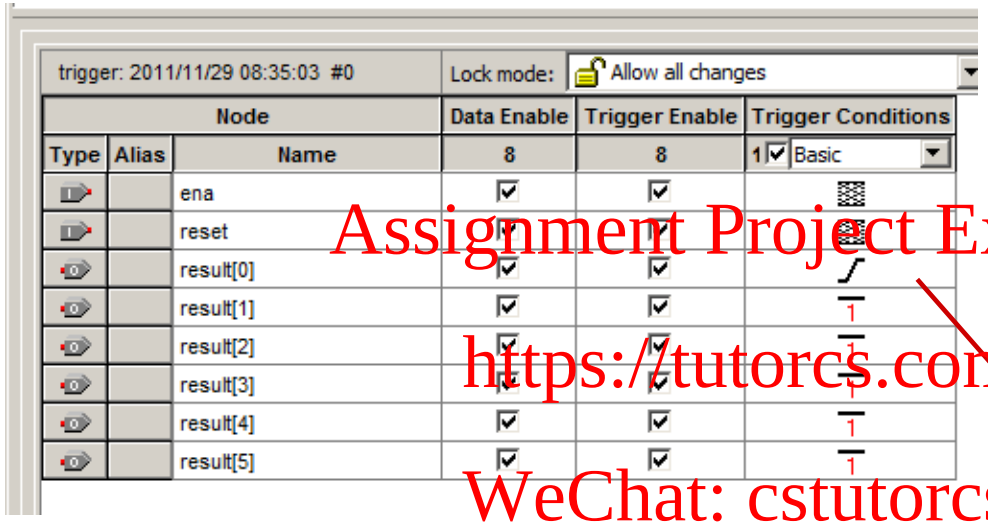


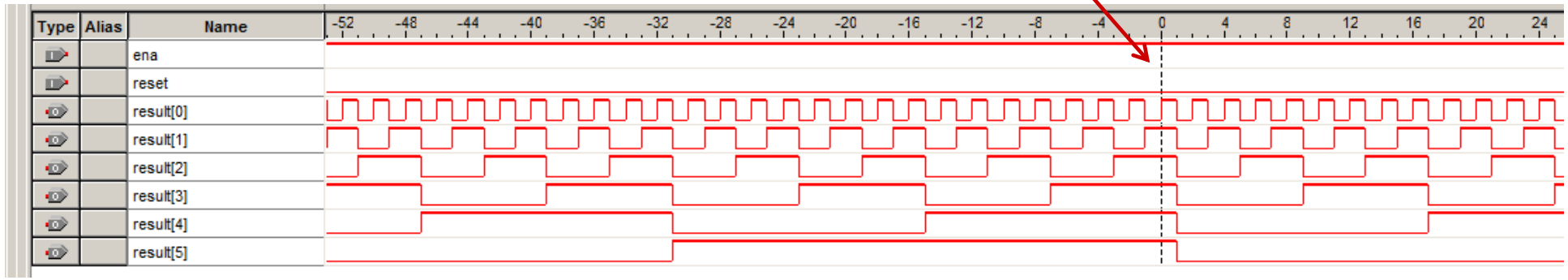
Displaying Acquired Data



- Display signal as bar or line chart
- Export to other tools for viewing or analysis (File Menu)
  - Creates .VWF, .TBL, .CSV, .VCD, .JPG or .BMP File

Example – 24 bit counter with enable

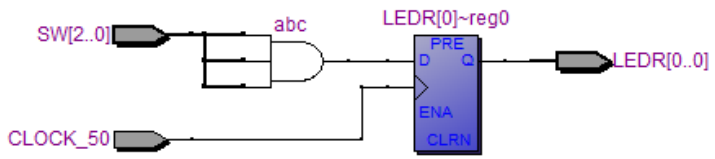


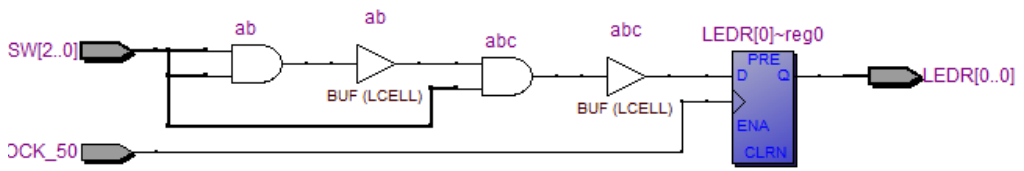


Preserving buried wires

```
1  module threeInputAnd(SW, LEDR, CLOCK_50);
2     input CLOCK_50;
3     input [2:0] SW;
4     output reg [0:0] LEDR;
5
6     wire ab, abc;
7
8     assign ab=SW[0]&SW[1];
9     assign abc=ab&SW[2];
10
11    always @ (posedge CLOCK_50)
12       begin
13          LEDR[0]<=abc;
14       end
15 endmodule
```

```
1  module threeInputAnd(SW, LEDR, CLOCK_50);
2     input CLOCK_50;
3     input [2:0] SW;
4     output reg [0:0] LEDR;
5
6     wire ab, abc /*synthesis keep*/;
7
8     assign ab=SW[0]&SW[1];
9     assign abc=ab&SW[2];
10
11    always @ (posedge CLOCK_50)
12       begin
13          LEDR[0]<=abc;
14       end
15 endmodule
```



Using STP File Review

1. Create .STP File
   - Assign Sample Clock
   - Specify Sample Depth
   - Assign Signals to STP File
   - Specify Triggering
   - Setup JTAG
2. Save .STP File & Compile with Design
3. Program Device
4. Acquire Data

Summary

- Design should be fully simulated before programming
- When the actual hardware does not follow the simulation, Logic Analysers can provide valuable information on debugging the design
- Altera's SignalTap II embedded logic analyser provides a low cost method of debugging designs in Altera FPGAs.