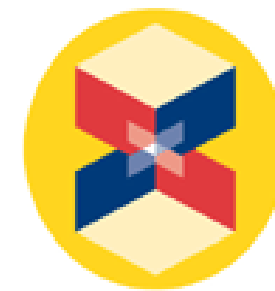


FPGA Design with Matlab & Simulink



Course Prepared by
Digitronix Nepal
www.digitronixnepal.com
Email: digitronixnepali@gmail.com



XILINX
SYSTEM
GENERATOR™
For DSP

Sections and Lecturers of the Course:

Section 1. Installing Matlab/Simulink and Xilinx VIVADO/ISE Design Suit

Lecture 1 : Downloading and Intalling Matlab/Simulink and Xilinx ISE

Lecture 2 : Version Compatibility for Matlab/Simulink and ISE/VIVADO

Section 2. Introduction to Matlab/Simulink and HDL Coder/System Generator

Lecture 1 : Matlab and Simulink Introduction

Lecture 2 : HDL Coder and System Generator Introduction

Lecture 3 : Program Xilinx Zynq SoC Devices with Embedded Coder and HDL Coder

Lecture 4 : Program Xilinx FPGAs Using HDL Coder with Xilinx System Generator

Section 3. Basic Project with System Generator and FPGA

Lecture 1 : Setting up FPGA and JTAG configuration (Adding your Board) to Sys Gen.

Lecture 2 : Dump in to Spartan 3E FPGA

Lecture 3 : Dump in to Zynq FPGA (Zedboard/Zybo)

Section 3. Basic Project with System Generator and FPGA

- Lab 31: Basic System Generator Design for FFT
- Lab 32: Creating JTAG Configuration for FPGA Board in System Generator

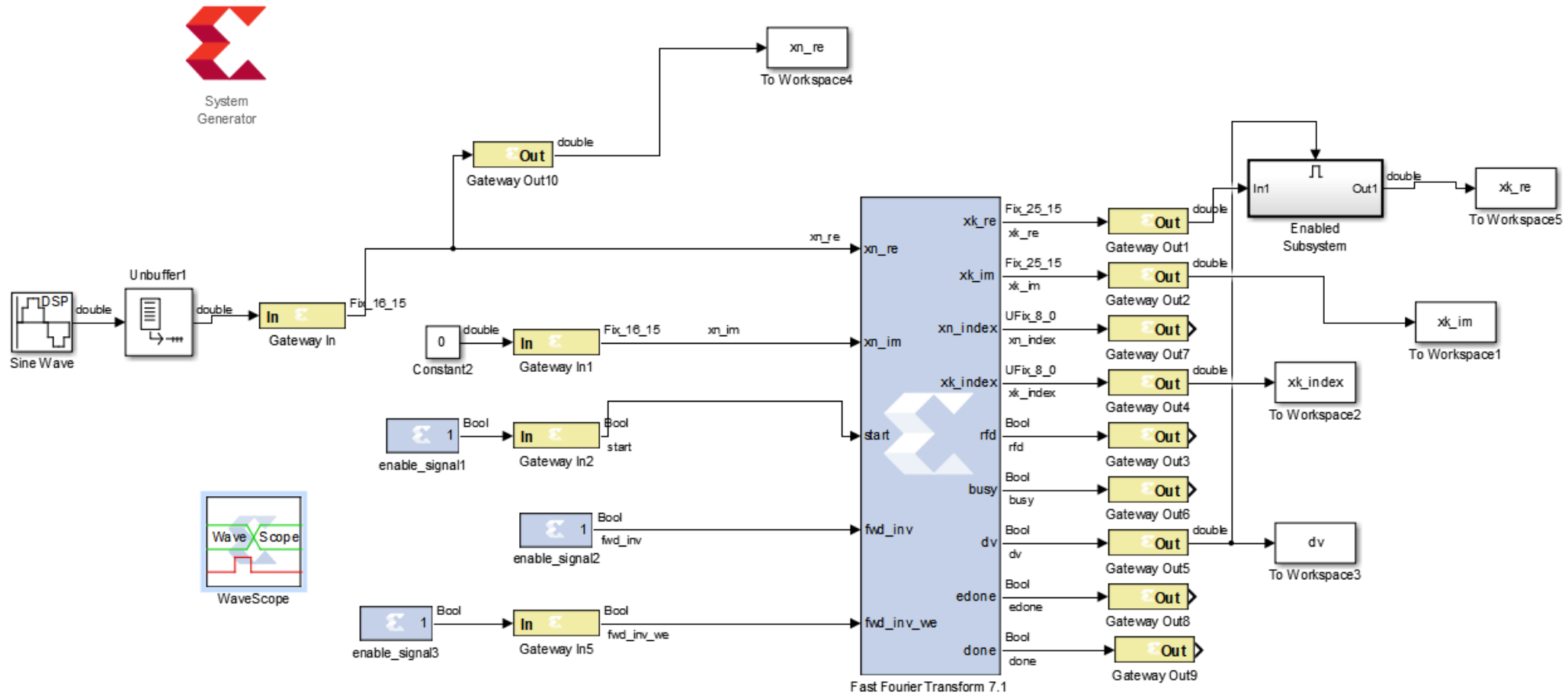
The background of the slide is a light green color with a faint, stylized image of a circuit board or PCB. The circuit traces are visible, particularly on the left side. The text is centered in a large, black, sans-serif font.

Lab 31: Basic System Generator Design for FFT

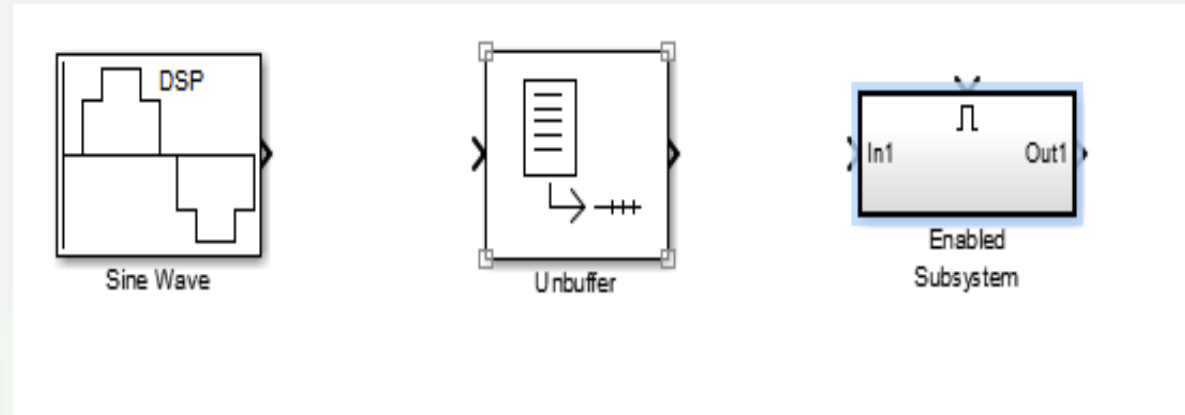
Innovation from Engineers...

Setting up Blocks

- First setup the blocks as shown in the picture.



Simulink Blocks



DSP sine wave: DSP sine wave block is used to generate a digital sinusoidal signal.

Unbuffer: Unbuffer block converts the frame output from DSP sinewave into a single dimensional scalar output.

Enabled Subsystem: Enabled subsystem will only output values when enable port is high. This is used after the system generator FFT block to get the valid outputs only.

Innovation from Engineers...

System Generator Blocks

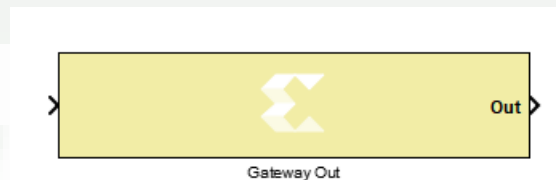
- Gateway In

- Gateway in block is used to provide input from Simulink blocks to the FPGA blocks



- Gateway Out

- Gateway out is used to output data or signal from system generator blocks back to Simulink environment.



FFT v7.1 core

- FFT v7.1 is a DSP block in System Generator and is core part of our design.
- It can process both FFT and IFFT algorithm.
- FFT v7.1 works in burst or pipelined mode and supports Radix-2 and Radix-4 algorithm.
- It can use either block ram or distributed ram resource in FPGA according to our need.
- FFT is used to process signal in frequency domain and it is also used in digital communication for different modulation scheme.

FFT v7.1 Settings

Fast Fourier Transform 7 1 (Xilinx Fast Fourier Transform 7.1)

Basic Advanced Implementation

Transform Length
256

Implementation Options
pipelined_streaming_io

Target clock frequency(MHz) 50

Target data throughput(MSPS) 50

Transform Length Options
 Run Time Configurable Transform Length

OK Cancel Help Apply

Fast Fourier Transform 7 1 (Xilinx Fast Fourier Transform 7.1)

Basic Advanced Implementation

Precision Options
Phase Factor Width 16

Scaling Options
 Unscaled Scaled Block Floating Point

Rounding Modes
 Truncation Convergent Rounding

Output Ordering
 Bit/Digit Reversed Order Natural Order
 Cyclic Prefix Insertion

Optional Pins
 en
 rst
 ovflo

Input Data Timing
 No offset 3 clock cycle offset (pre-v7.0 behavior)

OK Cancel Help Apply

FFT v7.1 Settings

Fast Fourier Transform 7.1 (Xilinx Fast Fourier Transform 7.1)

Basic | Advanced | **Implementation**

Memory Options

Data
 Block RAM Distributed RAM

Phase Factors
 Block RAM Distributed RAM

Number Of Stages Using Block RAM
1

Reorder Buffer
 Block RAM Distributed RAM

Hybrid Memories
 Optimize Block RAM count using hybrid memories

Optimize Options

Complex Multipliers
 Use CLB logic Use 3-multiplier structure (resource optimization) Use 4-multiplier structure (performance optimization)

Butterfly arithmetic
 Use CLB logic Use XtremeDSP Slices

FPGA Area Estimation

Define FPGA area for resource estimation

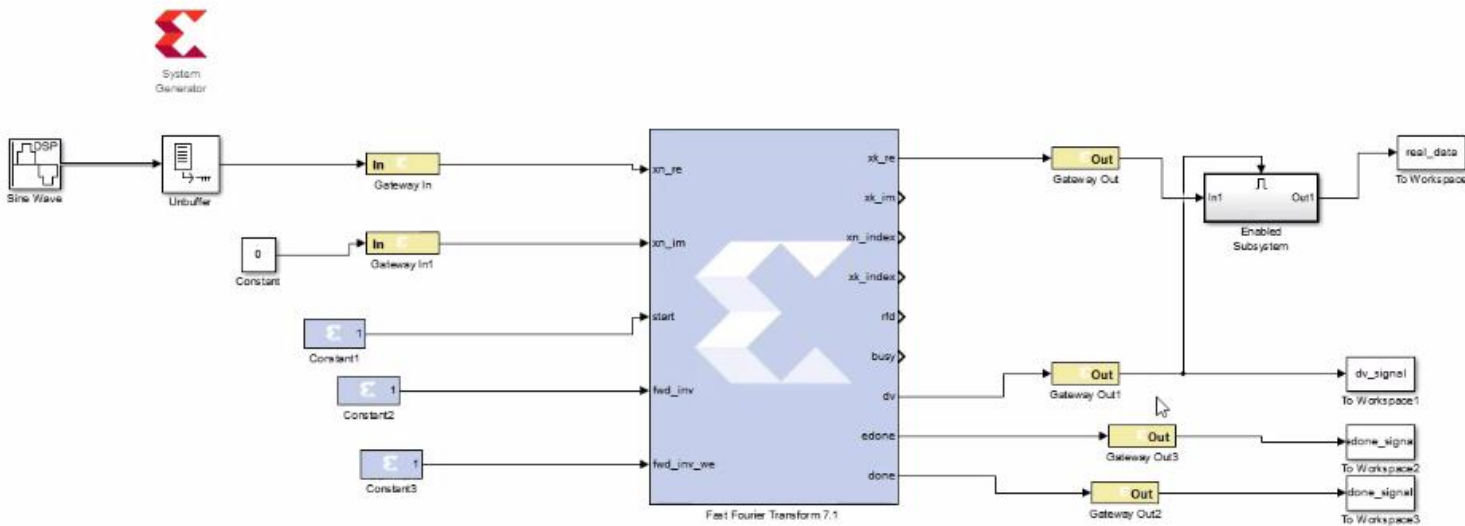
FPGA area [slices, FFs, BRAMs, LUTs, IOBs, emb. mults, TBUFs]
[0,0,0,0,0,0,0]

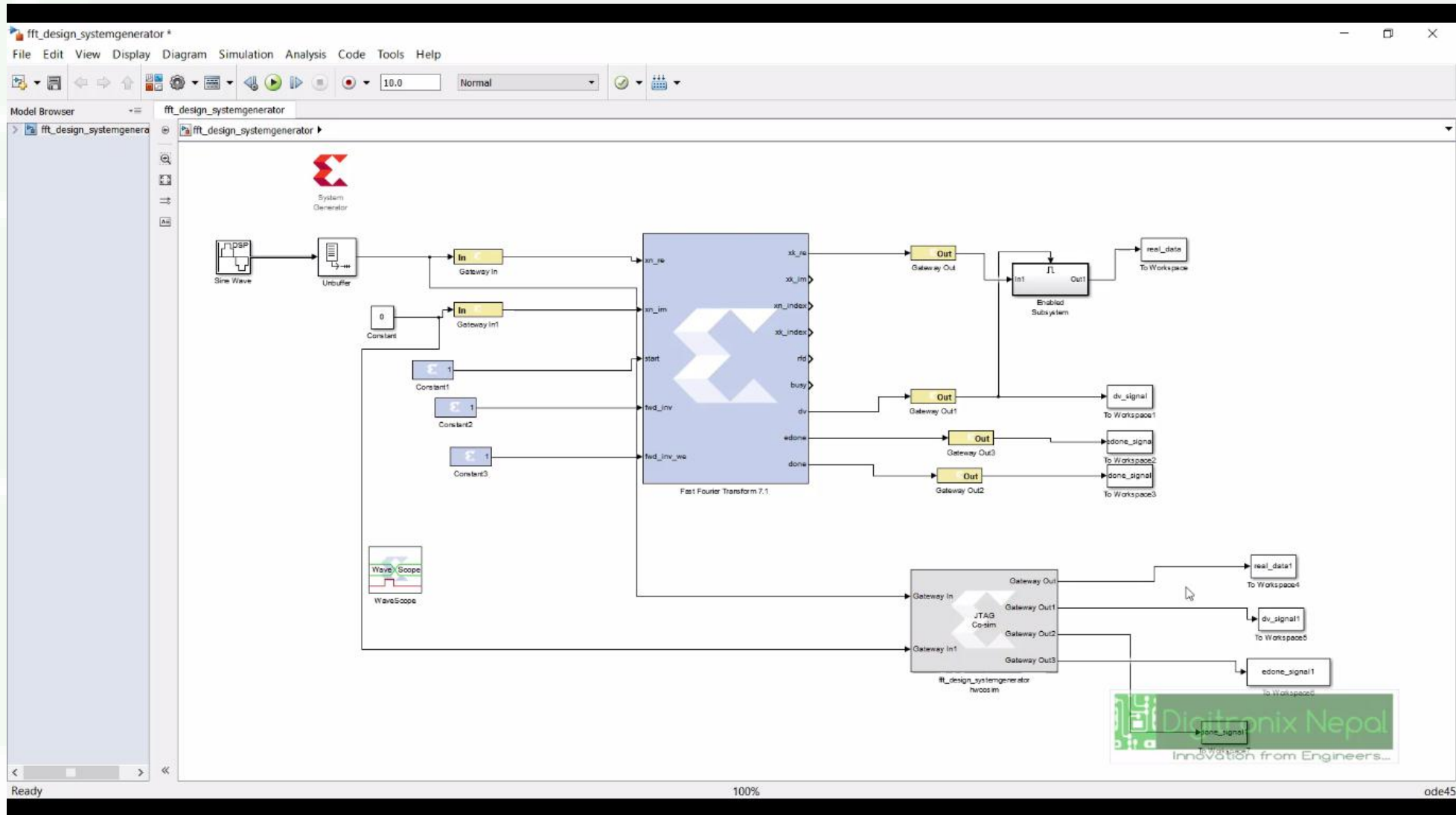
OK Cancel Help Apply

Lab 32: Creating JTAG Configuration for FPGA Board in System Generator



Innovation from Engineers...



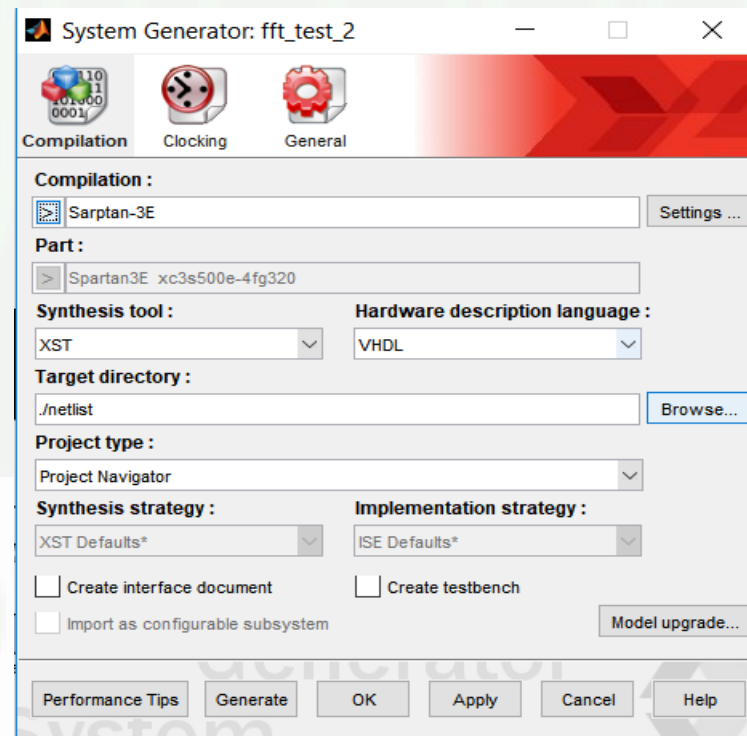


Hardware Co-simulation(HW-cosim)

- For hardware cosimulation we create a JTAG block that contains all the system generator portion.
- The JTAG block processes the our design in physical FPGA device.
- JTAG block can be referred as the physical device itself.
- JTAG configurations are device specific.
- To generate JTAG block we goto system generator token.
- Click Hardware Cosimulation in
- Choose the specific board for hardware cosimulation
- And then click on generate which we create a JTAG block of our design.

JTAG Configuration

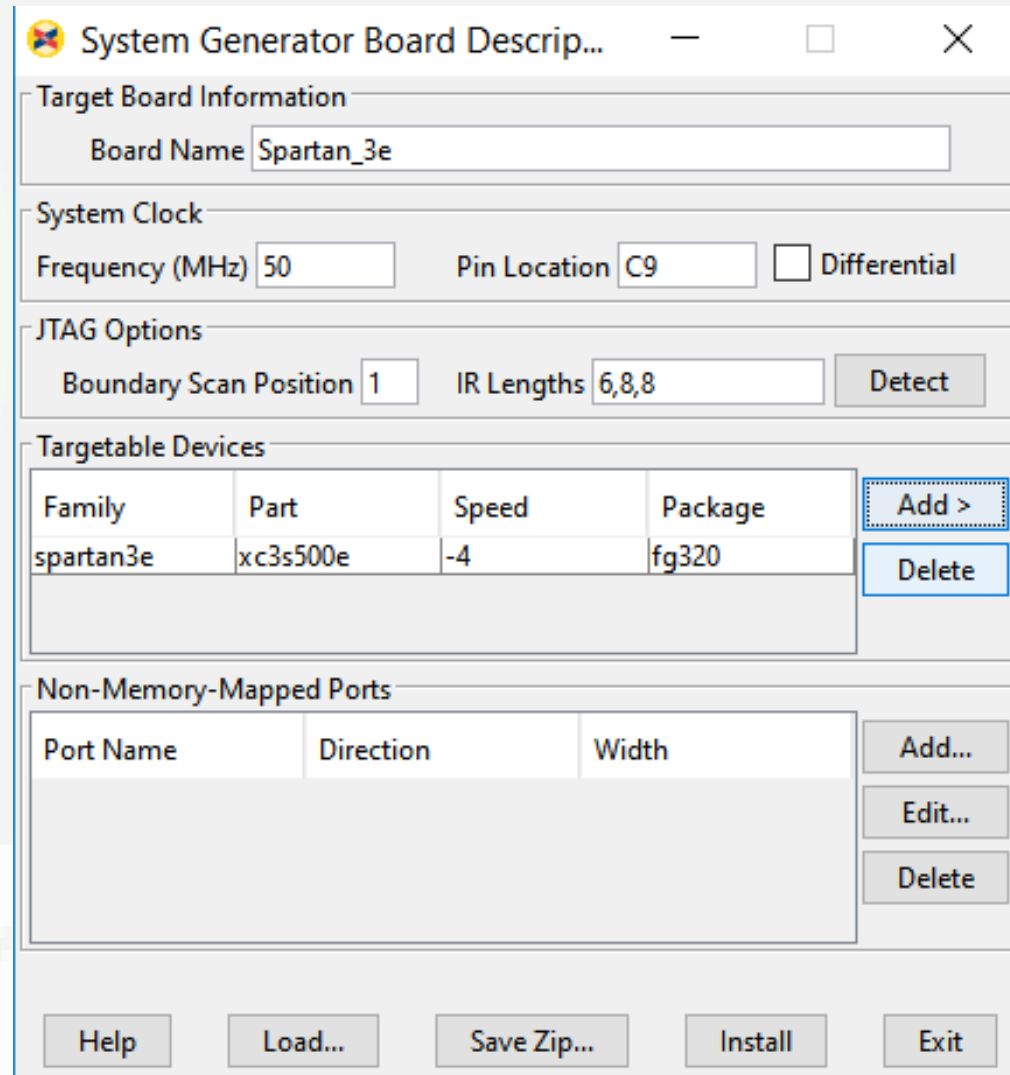
- For hardware co-simulation we have different pre-defined FPGA board configuration in System Generator Token.
- We can also add custom board configuration for different Xilinx FPGA boards



Custom JTAG Configuration

- Goto System generator token
- Select from compilation list and choose Hardware Co-simulation
- In hardware cosimulation choose New compilation target
- A new dialog box will appear where we have to set parameters of our target board.
- We have to input frequency of target FPGA and pin location of its system clock
- We also have to choose FPGA model and packaging configuration.

Custom JTAG Configuration for Spartan 3E Starter Kit



The screenshot shows the 'System Generator Board Description' dialog box with the following configuration:

- Target Board Information:** Board Name: Spartan_3e
- System Clock:** Frequency (MHz): 50, Pin Location: C9, Differential:
- JTAG Options:** Boundary Scan Position: 1, IR Lengths: 6,8,8, Detect:
- Targetable Devices:**

Family	Part	Speed	Package	
spartan3e	xc3s500e	-4	fg320	<input type="button" value="Add >"/>
				<input type="button" value="Delete"/>
- Non-Memory-Mapped Ports:**

Port Name	Direction	Width	
			<input type="button" value="Add..."/>
			<input type="button" value="Edit..."/>
			<input type="button" value="Delete"/>

Buttons at the bottom: Help, Load..., Save Zip..., Install, Exit

Engineers...



Lets go to
System Generator

Innovation from Engineers...

The background of the slide is a light green color with a faint, repeating pattern of a circuit board. The pattern is composed of white lines representing traces and rectangular shapes representing components, arranged in a grid-like fashion. The text "Thank You!" is centered on the slide. The word "Thank" is written in a red, italicized serif font, and the word "You!" is written in a black, bold serif font. The exclamation mark is also in black and bold.

Thank You!

Innovation from Engineers...