

ELBERT S7 HANDBOOK

An Educational Guide to FPGA Design and Development

CONFIDENTIAL and PROPRIETARY

This document contains confidential and proprietary information belonging exclusively to Numato	N
Systems Pyt Ltd. No confidential or proprietary	
information contained in this publication may be	RE
reproduced, stored in a retrieval system, or transmitted	Doc
in any form or by any means electronic, mechanical,	Doc
photocopied, recorded, or otherwise without prior	NS
written permission of Numato Systems Pvt Ltd.	
Copyright© 2025, Numato Lab.	

Numato Systems Pvt Ltd					
REV: V1.0					
Doc Name: ELBERT S7 HANDE	BOOK				
Doc No:	Page				
NSQMS25FP06B07-HB	1 of 1				

Table of Contents

SECTION I	4
1. INTRODUCTION	4
2. BOARD FEATURES	6
2.1. FPGA Device	6
2.2. Configuration and Memory	6
2.3. Communication Interface	7
2.4. User I/O	7
2.5. Peripheral Interfaces	7
3. APPLICATION	
4. WIRING DIAGRAM	9
5. USB Interfacing and Programming Options	10
6. JTAG Connector	11
7. PROG_B and Reset Buttons	11
8. Power Supply	
9. USER I/O	14
9.1. LEDs	
9.2. DIP Switches	14
9.3. PMOD Headers	15
10. PERIPHERAL INTERFACES	17
10.1. Seven Segment Display	17
10.2. HDMI TRANSMITTER	18
10.3. Micro SD Card	19
10.4. Gigabit Ethernet	20
10.5. Audio Jack	21
11. Generating Bitstream Using Vivado	
12. Programming Elbert S7 Using JTAG	23
13. Programming Elbert S7 Using USB-JTAG	25
14. Programming QSPI Flash using Vivado.	

SECTION II	28
TRAIN ON THE BOARD	28
Getting Started with Vivado: Creating a New FPGA Project	
1. UART Communication – Printing "Hello World"	31
2. Controlling Onboard Peripherals (LEDs, Seven Segment Display, and PM) with Slide Switches	OD) 44
3. HDMI OUTPUT EXAMPLE DESIGN	
4. DDR3 Mem test on ELBERT S7	58
5. Gigabit Ethernet Example Design	72
6. SD card test	
REFERENCES	105

Numato Lab[®]

SECTION I 1. INTRODUCTION



Unleash Your Creativity with Compact Power

The **Elbert S**₇ is a compact and versatile FPGA development board designed to support a wide range of educational and practical digital design applications. Built around the **Spartan-**7 **FPGA (XC7S50-1CSG324C)** from AMD, this board offers a reliable platform for both **beginners** and **experienced developers** looking to explore the world of FPGAs.

Whether you're a student learning digital logic or a developer building custom hardware solutions, the Elbert S7 makes FPGA development more accessible and engaging. Its thoughtful design supports hands-on experimentation, helping users understand fundamental concepts while also enabling the development of more advanced projects.

The board is an ideal choice for anyone interested in **digital design**, **embedded systems**, **signal processing**, **or rapid prototyping**. With strong support for industry-standard tools like **Vivado** and **Vitis**, and compatibility with both **Verilog** and **VHDL**, the Elbert S7 provides a smooth and scalable learning path for anyone stepping into the field of FPGA design.

Numato Lab[®]

2. BOARD FEATURES

The Elbert S7 FPGA development board offers a rich set of features, making it a powerful tool for learning and developing digital systems. Its well-balanced combination of core components and peripherals allows users to implement a wide variety of applications—from basic logic design to advanced embedded systems.

2.1. FPGA Device

- Model: AMD Spartan-7 (XC7S50)
- Package: CSGA324
- Speed Grade: -1

This low-power, high-performance FPGA offers enough logic resources and I/O options for both educational and practical digital design tasks.

2.2. Configuration and Memory

- DDR3 SDRAM: MT41J128M16JT-125: KTR, A high-speed 2 Gb (128M x 16)
- DDR3 memory chip used for applications requiring larger memory capacity, such as video processing, buffering, or running soft processors.
- **Flash Memory:** 128 Mb Quad-SPI (MT25QU128ABA1ESE-oSIT TR) Used to store FPGA configuration bitstreams and other application data.
- **Clock Source:** 100 MHz CMOS oscillator Serves as the main clock input to drive system timing.
- Configuration Methods:
 - JTAG (standard FPGA programming method)
 - USB (via onboard FTDI chip; supported on Both Windows and Linux systems)

2.3. Communication Interface

• **FTDI FT2232H**: Dual-channel USB-to-serial/FIFO interface Enables high-speed communication between the FPGA and a host PC. Also supports USB-based FPGA programming and debugging.

2.4. User I/O

• 8 DIP Switches & 8 LEDs:

Useful for creating simple input/output projects and learning digital logic control.

• Four PMOD Headers:

Standard 2x6 connectors for attaching external modules and user-defined peripherals.

2.5. Peripheral Interfaces

• SD Card Slot:

Provides external storage, useful for embedded applications requiring file access.

HDMI Transmitter:

Enables the board to send video signals to external displays, suitable for image processing or display generation projects.

• Gigabit Ethernet Port:

Supports high-speed networking, ideal for IoT, remote monitoring, or network-based applications.

• Audio Jack:

Allows audio input/output for sound-related projects.

• Seven-Segment Display:

Displays numeric or limited alphanumeric information, ideal for counter or output visualization tasks.

3. APPLICATION

The Elbert S7 is a versatile development board suitable for a wide range of educational and practical applications:

- Educational Use: Ideal for teaching digital design and embedded systems in schools and universities.
- **Prototype Development**: Enables rapid hardware prototyping and testing of new product ideas.
- Accelerated Computing: Supports hardware-based acceleration of compute-intensive tasks.
- **Custom Processor Design**: Allows development and testing of soft-core or custom embedded processors.
- **Signal Processing**: Suitable for implementing and testing real-time digital signal processing applications.
- **Communication Systems**: Enables the design and evaluation of communication protocols and devices.
 - Video Processing: Supports projects involving HDMI output and image/video processing tasks.

4. WIRING DIAGRAM



5. USB Interfacing and Programming Options

The Elbert S7 board features a high-speed USB interface powered by the FTDI FT2232H chip, enabling seamless communication with Windows, Linux, or macOS computers. A standard USB Type-A to Type-C cable is used to connect the board to a host system. The USB connection also supplies power to the board by



default, so it's important to avoid connecting it to overloaded or unpowered USB hubs.

To provide flexibility in programming, the board includes a programming mode selection mechanism. A multiplexer (MUX) is used internally to switch between two configuration sources:

- JTAG (via external programmer)
- USB-JTAG (via onboard FT2232H chip)

The FT2232H's Channel A is configured specifically for USB-based FPGA programming. A PGM SEL switch is provided on the board, allowing users to select the desired programming method:

- Set the switch to USB to use onboard USB-JTAG.
- Set the switch to JTAG to program via an external JTAG programmer.

This flexible configuration makes the Elbert S7 suitable for both beginners using USB for simplicity and advanced users who prefer direct JTAG access.



6. JTAG Connector



The Elbert S7 board includes a standard JTAG header that provides access to the FPGA's internal JTAG registers. This interface supports programming and debugging using tools such as the Xilinx Platform Cable USB. Users can connect a compatible JTAG cable to this header to perform direct configuration, in-system debugging, or low-level hardware testing. This option is especially useful for advanced users requiring greater control and visibility into the FPGA during development.



Elbert S7 features a Push-button **PROG B** normally meant to be used as a "PROG B" signal for configuration reset. Push-button PROG_B is connected to FPGA enabling manual configuration reset, push-button **PROG B** is pin **R8.** For connected to GND. The user can reconfigure the FPGA manually, by PROG B pressing this push-button **PROG_B**.



"PROG B" controls the configuration logic. When the PROG B pin is de-asserted, resets the FPGA and initializes the new configuration.

Elbert S7 features a Push-button **RESET** normally meant to be used as "Reset" signal for designs running on FPGA. Push-button RESET is connected to FPGA

pin **T14.** Push-button **RESET** is **active-high**. This push button can also be used for any other input and is not just limited to be used as a Reset signal.



8. Power Supply



The **Elbert S**₇ board features a dual power input architecture, allowing users to power the board either through a USB Type-C port or a **5V** DC jack. This flexible design is ideal for both development and deployment environments, where users may prefer different power sources based on convenience or availability.

Each input is routed through a Schottky diode (**D2** for USB Type-C and **D4** for the DC jack), which plays a crucial role in protecting the board. These diodes prevent reverse current flow, ensuring that power does not back feed into the sources, and also helps in automatic source selection by allowing the input with the higher voltage to take priority.

The outputs of these diodes feed into a power selector stage that determines which source will power the board. This intelligent selection ensures that only one source supplies power at a time, eliminating any risk of conflict or damage.

Once selected, the power signal goes through a filter and protection block. This section typically consists of inductors and capacitors to filter out voltage ripples and electromagnetic noise, ensuring clean and stable voltage. Additionally, it protects the board against sudden spikes or other electrical disturbances.

A user-accessible **PWR SW** (Sw11) is provided to manually control the power delivery to the board. This switch adds convenience by allowing users to turn the board ON or OFF without needing to disconnect the power cable physically.

Finally, an onboard Power LED (**PWR**) connected after the switch provides a visual indication when the board is powered.

The stable output voltage is labelled as VIN, which is then distributed to various subsystems and regulators on the board.

Note: Only a regulated 5V DC power supply should be used to power the board through either the USB Type-C port or the DC jack. Supplying a voltage higher than 5V may damage the internal circuitry.





9. USER I/O

9.1. LEDs

The board features 8 user-controllable LEDs. These LEDs serve as visual indicators and are commonly used for debugging or representing binary output data. Each LED is connected to a dedicated FPGA pin and can be driven directly using logic outputs.

PIN NAME	FPGA PIN	IO STANDARD	
 LED o	V14	LVCOMS33	
 LED 1	V15	LVCOMS33	
 LED 2	U12	LVCOMS33	
 LED 3	V13	LVCOMS33	
LED 4	T12	LVCOMS33	
LED 5	T13	LVCOMS33	
LED 6	R11	LVCOMS33	
 LED 7	T11	LVCOMS33	

Table 1: LED Pin Mapping

9.2. DIP Switches

There are 8 onboard DIP switches that can be used to input binary data or control signals into the FPGA. These inputs are ideal for controlling the flow of a design, setting modes, or triggering actions within a design.

PIN NAME	FPGA PIN	IO STANDARD
SW o	C4	LVCOMS18
SW 1	B4	LVCOMS18
SW 2	C3	LVCOMS18
SW 3	B3	LVCOMS18
SW 4	A5	LVCOMS18
SW 5	A4	LVCOMS18
SW 6	A3	LVCOMS18
SW 7	A2	LVCOMS18

Table 2: DIP Switch Pin Mapping

9.3. PMOD Headers

PMOD (Peripheral Module) connectors are a widely adopted standard for connecting peripheral devices to FPGA and microcontroller development boards. The Elbert S7 development board provides **four 2×6 PMOD headers** (P3, P4, P5, P6), allowing users to interface a variety of external modules including sensors, displays, communication interfaces, and custom digital circuits.

Each PMOD header offers **eight general-purpose I/O (GPIO) signals** plus **two dedicated power pins (3.3V and GND)**, organized in a dual-row format. The I/O lines from each PMOD are connected directly to the FPGA, enabling flexible software-defined control and configuration.



DINNAME		IO STANDARD			
	CONNo(P1)	CONN1(P2)	CONN2(P3)	CONN3(P4)	IO STANDARD
Do	M18	H18	A10	C17	LVCMOS33
D1	P14	G16	C10	D18	LVCMOS33
D2	P17	H16	C12	D16	LVCMOS33
D3	R18	K14	B11	E16	LVCMOS33
D4	N18	G18	A9	B18	LVCMOS33
D5	P15	G17	C9	C18	LVCMOS33
D6	P18	H17	C11	D17	LVCMOS33
D 7	T18	J15	A11	E17	LVCMOS ₃₃

Table 2: PMOD Pin Mapping

Numato Lab[®]

G

C

Ε

10. PERIPHERAL INTERFACES

10.1. Seven Segment Display

A seven-segment display is a simple output device commonly used for displaying decimal numbers or characters. The **Elbert S7** includes a 4-digit seven segment display, which is useful for displaying counters, timers, or debugging information.

Each digit is made up of 7 individual segments (labelled A–G) and an optional decimal point (dp). The display works by quickly switching between digits using control signals-a technique known as multiplexing.

You can control which digit is shown by activating the digit's enable line and then lighting up the correct segments to form a number or letter.

Note: All signals (*a*, *b*, *c*, *d*, *e*, *f*, *g*, *dot*, *enable 1*, *enable 2*, *enable 3*, *enable 4*) used for controlling 7-Segment display are **active-low** signals. So, for example, for displaying "8" in display-2, users need to drive *Enable 2* to *o* as well as drive signals *a*, *b*, *c*, *d*, *e*, *f* to *o*. All other signals need to be driven to 1.



10.2. HDMI TRANSMITTER

The HDMI (High-Definition Multimedia Interface) Transmitter on the Elbert S7 board allows the FPGA to send digital video and audio signals to an external display such as a monitor or TV.

HDMI uses a high-speed signalling technology known as **TMDS (Transition-Minimized Differential Signalling)**. This ensures reliable transmission of data over HDMI cables by reducing electromagnetic interference and ensuring signal integrity.

Inside the FPGA, video signals are generated in digital format—usually in RGB (Red, Green, Blue) colour space—along with synchronization signals such as horizontal sync, vertical sync, and data enable. These signals are then encoded and sent through the HDMI transmitter chip to the display device.

The HDMI interface typically includes the following signals:

- TMDS Data Channels (3 pairs): Carry video/audio/control data.
- TMDS Clock: Synchronizes the data.
- **DDC (I2C lines)**: Used to read the display's capabilities (EDID)
- **HPD**: Lets the FPGA know a display is connected.
- **CEC**: Optional control communication between HDMI devices

	1						
path	R15	HDMI_TX2_P	4		35	J4_D2_P	
-	T15	HDMI_TX2_N	6		33	J4_D2_N	
		HDMI_TX1_P	_			J4_D1_P	
	010		ľ		32	14 DL N	
,	V17		9		30		
SPARTAN 7	U17	HDMI_TX0_P	10		29	J4_D0_P	
X07550-1056A5240	U18	HDMI_TX0_N	12	RUFFER	27	J4_D0_N	НДМІ
				TODISESODETO			CONNECTOR
	R16	HDMI_TX_CLK_P	13	FUISSZODDIK	26	J4_CLK_P	
	R17	HDMI_TX_CLK_N	15		24	J4_CLK_N	
	U15	HDMI_TX_CEC	16		23	J4_CEC	
,	V16	HDMI_TX_SCL	17		22	J4_SCL	
	P13	HDMI_TX_SDA	18		21	J4_SDA	
	R13	HDMI_TX_HOT	19		20	J4_HPD	

10.3. Micro SD Card

The Elbert S7 board features a **microSD card slot**, which allows users to interface with removable flash memory for data storage and retrieval. This is especially useful for embedded applications where storing files, logs, configuration settings, or multimedia content is required.

The SD card slot on the board is wired to the FPGA through an **SPI (Serial Peripheral Interface)**. SPI is a simple and widely used protocol for communicating with memory devices like SD cards. Although SD cards also support a more complex SD bus mode, SPI is preferred in FPGA designs due to its simplicity and ease of implementation.

The typical SPI signals used are:

- **MOSI (Master Out Slave In)** Data sent from the FPGA to the SD card.
- **MISO (Master In Slave Out)** Data received from the SD card to the FPGA.
- SCLK (Serial Clock) Clock signal generated by the FPGA to control communication timing.
- **CS (Chip Select)** Used to select and activate the SD card.



10.4. Gigabit Ethernet

Elbert S7 Development Board features KSZ9031RNX, a highly integrated Ethernet transceiver from Microchip that comply with 10BASE-T, 100BASE-TX, and 1000Base-T IEEE 802.3 standards. It supports communication with the Ethernet MAC layer via standard RGMII interface. KSZ9031RNX implements auto-negotiation to automatically determine the best possible speed and mode of operation. It contains a high-performance 10/100/1000T transceiver and the RGMII interface supports 1000Mbps (1Gbps) operation.

						1	
	D7	ETH_TXD0	19				RJ45
	C7	ETH_TXD1	20			GPHY_TXRX_D1_N	
	B7	ETH_TXD2	21		11	GPHY_TXRX_D1_P	1
	A6	ETH_TXD3	22		10		1
CD4074417	D6	ETH_TX_CTL	25		8	GPHY_TXRX_C1_N	
5PARIAN / XC7550-1C5GA324C	50	ETH_TX_CLK	24	ETHERNET PHY	7	GPHY_TXRX_C1_P	
X07350 1050X5140	A8	ETH_RXD0	2.4	KSZ9031RNXIC-TR	6	GPHY_TXRX_B1_N	
	A7	ETH_RXD1	31		5	GPHY_TXRX_B1_P	
	C5	ETH_RXD2	28				
	85	ETH_RXD3	27		3	CONV TYPY AL D	
	EA	ETH_RX_CTL	1.1		2	GFIT_TARA_AL_P	
	~	ETH_RX_CLK	35		17	LED_ACT	
	D5	ETH_MDC	34		15	LED_LINK	
	55	ETH_MDIO	37				
	R1	ETH_RST	1.				31
			42			l	

10.5. Audio Jack

The Elbert S7 FPGA board is equipped with a 3.5mm stereo audio jack, enabling audio output to external speakers or headphones. This output is managed by a dedicated digital-to-analog converter (DAC) — the CS4345-CZZ, a high-performance stereo DAC from Cirrus Logic. The inclusion of this audio interface makes the board well-suited for multimedia, audio signal processing, and embedded audio playback applications.

Key Features

- **Stereo Output**: Supports two audio channels (Left and Right) for full stereo playback.
- **High-Quality DAC**: The CS4345 provides low distortion and high dynamic range, resulting in clear and high-fidelity sound.
- Digital Audio Interface: The DAC accepts audio data in I²S (Inter-IC Sound) format, a standard serial protocol for transmitting PCM audio between digital audio devices.
- **16/24-bit Audio Support**: Capable of processing CD-quality and high-resolution audio streams.
- **Headphone Compatible**: The output is designed to drive line-level audio, which can be connected to headphones or powered speakers.



11. Generating Bitstream Using Vivado

The bitstream can be generated for Elbert S7 in Vivado by following the steps below:

Step 1: It is recommended to generate .bin bitstream file along with .bit bitstream file. Click "Bitstream Settings".



Step 2: Select "-bin_file*" option in the dialog window and Click OK.

Q.	Bitstream	5	
Project Settings	Specify various settings related to writing Bitstream		
General Simulation	(i) Note: Additional bitstream settings will be availa	ble once you open an implemented design.	
Elaboration	~Write Bitstream (write_bitstream)		
Synthesis	tcl.pre		
Implementation	tcl.post		
Bitstream	-raw_bitfile		
> IP	-mask_file		
Tool Settings	-no_binary_bitfile		
Project	-bin_file		
IP Defaults	-readback_file		
> Vivado Store	-logic_location_file		
Source File	-verbose		
Display	More Options		
Help			
3rd Party Simulators > Colors Selection Rules Shortcuts Strategies	-bin_file Write a binary bit file without header (bin).		
> Window Behavior			

Step 3: Finally click "Generate Bitstream".

~	PROGRAM AND DEBUG
	👫 Generate Bitstream
	✓ Open Hardware Manager
	Open Target
	Program Device
	Add Configuration Memory Device

12. Programming Elbert S7 Using JTAG

Set Switch **PGM_SEL** to **JTAG** for JTAG programming.

Elbert S7 FPGA features an onboard JTAG connector which facilitates easy reprogramming of SRAM and onboard SPI flash through JTAG programmer like "AMD Platform cable USB". Following steps illustrate how to program FPGA on Elbert using JTAG.

Step 1: By using JTAG cable, connect AMD platform cable USB to Elbert S7 and power it up.

Step 2: Open Vivado project and open the target by clicking on the "Open Target" in "Open Hardware Manager" in the "Program and Debug" section of the Flow Navigator window. Select "Auto Connect".



Step 3: If the device is detected successfully, then select "Program Device" after right clicking on the target device "XC7S50_0" as shown below.



Step 4: In the dialog window which opens, Vivado automatically chooses correct bitstream file if the design was synthesized, implemented and bitstream generated

successfully. If needed, browse to the bitstream which needs to be programmed to FPGA. Finally, click "Program".

Program Device		×
Select a bitstream prog select a debug probes programming file.	gramming file and download it to your hardware device. You can optiona file that corresponds to the debug cores contained in the bitstream	
Bitstre <u>a</u> m file: Debu <u>q</u> probes file: I Enable end of s	C:/projects/Elbert_S7/Demo.bit	
?	<u>P</u> rogram C	ancel

As soon as "Program" is clicked, a green coloured DONE LED (DONE) on Elbert S7 should light up, indicating that programming process is going on. This LED will turn off when the configuration is complete.

13. Programming Elbert S7 Using USB-JTAG

Ensure that the <u>D2XX drivers</u> are installed prior to programming. The channel A of FTDI FT2232H chip on Elbert S7 board is connected to the JTAG interface of the FPGA. Through this connection, USB interface can be used as a JTAG programmer, eliminating the need for a dedicated JTAG cable or connector. Following steps illustrate how to program FPGA on Elbert S7 using USB.

1. Ensure that Switch **PGM_SEL** is set to **USB** and Connect the USB Type-C cable to the FPGA board.

2. Click on "Auto connect" under hardware manager and it will automatically establish the connection.



14. Programming QSPI Flash using Vivado.

A .bin or .mcs file is required for programming Elbert S7 onboard QSPI flash.

Step 1: Open Vivado project and open the target by clicking on the "Open Target" in "Open Hardware Manager" in the "Program and Debug" section of the Flow Navigator window. Select "Auto Connect".



Step 2: If the device is detected successfully, then select "Add Configuration Memory Device" after right clicking on the target device "xc7s50_0" as shown below.



Step 3: Select the memory device "mt25ql128-spi-x1_x2_x4", then click OK.

Add Configuration	on Memory Device					
Choose a cor	nfiguration memory	part.				
)evice: 🛑 xc7a5(0t_0					
r						
<u>M</u> anufacturer	Micron	~		Туре	spi	~
Density (<u>M</u> b)	128	~		Width	All	~
			<u>R</u> eset All Filters			
ect Configuration	Memory Part					
ect Configuration	Memory Part					
ect Configuration Search: Q- Name	Memory Part	Part	Manufacturer	Alias		
ect Configuration Search: Q- Name 9 mt25ql128-sj	Memory Part	Part mt25ql128	Manufacturer Micron	Alias n25q128-3.	3v-spi-x1_x2_x4	
ect Configuration Search: Q- Name IVA IVA IVA IVA IVA IVA IVA IVA IVA IVA	Memory Part pi-x1_x2_x4 pi-x1_x2_x4	Part mt25ql128 mt25qu128	Manufacturer Micron Micron	Alias n25q128-3. n25q128-1.	3v-spi-x1_x2_x4 8v-spi-x1_x2_x4	
ect Configuration Search: Q- Name P mt25ql128-sp P mt25qu128-s <	Memory Part pi-x1_x2_x4 pi-x1_x2_x4	Part mt25ql128 mt25qu128	Manufacturer Micron Micron	Alias n25q128-3. n25q128-1.	3v-spi-x1_x2_x4 8v-spi-x1_x2_x4	

Step 4: After completion of Step 3 the following dialog box will open. Click OK.



Step 5: Browse to the working .bin file or the .mcs file (whichever applicable) and click OK to program as shown below. If programming is successful, a confirmation message will be displayed.

	Program Configura	ation Me	emory Device							×	
	Select a configuration	n file ar	id set programn	ning options	s.					٦	
	Memory Device:	mt2 0://mt2	5ql128-spi-x1_x	2_x4					•		
	Configuration file:	C:/proj	ects/Elbert_S//s	ampie.bin							
	PR <u>M</u> file:				1					* * *	R
	State of non-config	i mem l	/O pins: Pull-	none 🗸							
• •	Program Operatio	ns									
	Address Range	Ð:	Configuration	File Only		~					
	✓ Erase										
	🗌 <u>B</u> lank Chec	k									
	✓ P <u>r</u> ogram										
	✓ Verify										
	Verify <u>C</u> hec	ksum									
	SVF Options										
	Create <u>S</u> VF	Only (n	o program oper	ations)							
	SVF File:									•	
	?				c	Ж	Cance	I	<u>A</u> pply	/	

SECTION II train on the board

Getting Started with Vivado: Creating a New FPGA Project

Before diving into the peripheral interface projects, it is essential to understand the basic procedure for setting up a new project in Vivado tailored for the Elbert S7 FPGA board. This section will guide you through the initial steps, including downloading and configuring the Board Support Package (BSP).

To begin, download the **Elbert S7 BSP** from our official <u>GitHub</u> repository and place it in the appropriate board files directory (follow the readme file in GitHub repo) on your computer. This allows Vivado to recognize the Elbert S7 board during project creation, simplifying IP integration and pin assignments.

The procedures described in this part will remain consistent across all peripheral interface projects throughout **Section II**. By following this workflow, you'll ensure that your development environment is correctly set up, enabling a smooth and efficient design experience.

Prerequisites:

Hardware:

- Elbert S7 FPGA Development Board.
- Xilinx Platform Cable USB II JTAG debugger. (optional)
- USB A to USB Type C cable.
- 5V DC power suppy.

Software:

- Vivado Design Suite with Vitis installed (2024.1)
- Serial terminal application (PuTTY, Tera Term, etc.)

Basic procedures to create new project in Vivado.

E

Step 1:

Download and install the Vivado Board Support Package files for Elbert S7 from <u>here</u>. Follow the README.md file on how to install Vivado board support files for Numato Lab boards.

Step 2:

Open the AMD Vivado Design suite, go to "File -> Project -> New" to create a new project. The "New project" window will pop up. Click "Next".

<u>F</u> ile	Flow Tools <u>W</u>	/indow	Help Q- Quick Act
	Project		<u>N</u> ew
	Checkpoint	+	Open
	<u>C</u> onstraints	•	Open <u>R</u> ecent ►
	Simulation Waveform	•	Open Ex <u>a</u> mple
	Īb	•	
	I <u>m</u> port	∍ar	t
	Exit		

Step 3:

In the "Project Name" window, enter a name for the project and save it at a suitable location. Select the option "Create project subdirectory" to keep all the project files in a single folder.



Now you will see the "Project Type" page as shown below. Select the "RTL Project" and select the option "Do not specify sources at this time". Click "Next".

Nev	w Project	×
Proje Speci	ect Type If the type of project to create.	٦
	RTL Project You will be able to add sources, create block designs in IP Integrator, generate IP, run RTL analysis, synthesis, implementation, design planning and analysis.	
?	<back ca<="" einish="" td=""><td>incel</td></back>	incel

Step 4: In the "Default Part" window, select the "Boards" tab. Choose the Vendor as "numato.com", filter the Name "Elbert_S7" and select the board as shown below. Click "Next" to continue. If Elbert S7 is not displayed in the boards list, make sure that the board support files are installed correctly.

efault Part hoose a default AMD	part or board for your p	project.									Ē
Parts Boards											
Vendor: numato.c	est available boards fri om	om git repo	ository, click or	Name: Elbe	n. Dismiss rt_S7			¥	Board Rev:	Latest	~
Q Ξ ≑ Search: Q-	K, ▼,		~								
Display Name	Preview	Status V	endor	File Version	Part	I/O Pin Count	Board Rev	Available IOBs	LUT Elements	FlipFlops	Block RAMs
Elbert_S7	Cost (California) Cost (California) Cost (California) Cost (California) Cost (California)	nstalled n	umato.com	1.0	xc7s50csga324-1	324	1.0	210	32600	65200	75
Refresh Catal	og was last updated or	n 02/28/202	25 12:36:09 P	М							3

In the next window, click "Finish" to complete creating the new project. When the new project wizard exits, a new project will be created by Vivado with the specified settings.

1. UART Communication – Printing "Hello World"

INTRODUCTION

In this first hands-on project, we will verify the UART (Universal Asynchronous Receiver/Transmitter) interface on the Elbert S7 FPGA board by printing a simple "Hello World" message to the serial terminal. UART is a widely used serial communication protocol that allows the FPGA to communicate with a host PC or other devices using simple text-based messaging.

This project serves as a basic sanity check to ensure the UART peripheral is functioning correctly and that the board can transmit data over a serial connection. It also helps familiarize users with integrating IP cores and observing output through a terminal emulator like Tera Term or PuTTY.

By completing this project, users will gain confidence in creating Vivado projects, generating bitstreams, using the BSP, and verifying output via UART communication.

Creating Microblaze based Hardware Platform for Elbert S7 The following steps will walk you through the process of creating a new project with Vivado and building a hardware platform with MicroBlaze soft processor using an IP integrator.

STEP 1:

To create a new Vivado project specifically for the **Elbert S7** FPGA board, follow the procedure outlined in the section "<u>Getting Started with Vivado: Creating a New</u> <u>FPGA Project.</u>"

STEP 2:

After creating a new project successfully, In the "Flow Navigator" panel, select "Create Block Design" under the IP integrator section. Give an appropriate name (Eg: "Hello_world ") to the design and click "OK ".

Create Block Design	I	×
Please specify name	of block design.	λ
<u>D</u> esign name:	Hello_world	
Directory:	Section 4	~
Specify source set:	🗅 Design Sources	~
?	ОК	Cancel

Step 3:

Go to Diagram window, right click and select "Add IP" from the popup menu. Search for "MicroBlaze" and add it to the design by double-clicking it.

Search: Q- mid (6 matches)	
👎 MicroBlaze	
👎 MicroBlaze Debug Module (MDM)	
👎 MicroBlaze Debug Module (MDM) V	
MicroBlaze MCS	
MicroBlaze MCS V	
👎 MicroBlaze V	
ENTER to select, ESC to cancel, Ctrl+Q for IP details	

Click "Run Block Automation" present in the "Designer Assistance available" bar on the top left corner of the window to complete the design. Select the settings as shown in the following image. Click "OK" for Vivado to automatically configure the blocks for you.

Q ↓ ↓ ✓ ✓ All Automation (1 out of 1 selected) ✓ ♥ ♥ ♥	Description MicroBlaze connection au MicroBlaze Debug Module are added and connected	tomation generates local memory of selected size, and caches can be configured. , Peripheral AXI Interconnect, Interrupt Controller, a clock source, Processor System Reset I as needed. A preset MicroBlaze configuration can also be selected.
	Information about the opti	ons can be found in the tooltips.
	Preset	None ~
	Local Memory ECC	None v
	Cache Configuration	None v
	Debug Module	Debug Only V
	Interrupt Controller	Enabled
	Clock Connection	New Clocking Wizard

Step 4:

Double click "Clocking Wizard" IP and customize "Board" settings as shown in the following image.

Documentation 🕞 IP Location			
IP Symbol Resource	Component Name clk_wiz_1		
Show disabled ports	Board Clocking Options Output Clocks MMCM	Settings Summary	
	Associate IP interface with board interface		
	IP Interface	Board Interface	
	CLK_IN1	sys clock	*
	CLK_IN2	Custom	*
	EXT_RESET_IN	reset	*
elk_in1 locke	d -		

Step 5:

Run "Connection Automation" and select all the pins.

Numato Lab[®] REV: V1.0

Run Connection Automation		×
Automatically make connections in your desig the right.	n by checking the boxes of the interfaces to connect. Select an interface on the left to display its configuration options on	λ
Q X \$ ✓ All Automation (3 out of 3 selected) ✓ * Image: Constraint of the selected of the selec	Select an interface pin on the left panel to view its options	
•	ОК Са	ncel

Step 6:

Go to the Board section, Drag and drop the USB UART from the Board section to the design.



Click on "Run Connection Automation" select all the pins and click ok.

Connect Slave interface (/axi_uartlite		
Options	_0/S_AXI) to a selected Master add	lress space.
Master interface	/microblaze_0 (Periph) 🗸 🗸	
Bridge IP	/microblaze_0_axi_periph 🗸 🗸	
Clock source for driving Bridge IP	/clk_wiz_1/clk_out1 (100 MHz)	¥
Clock source for Slave interface	Auto	*
Clock source for Master interface	/clk_wiz_1/clk_out1 (100 MHz)	*
	Options Master interface Bridge IP Clock source for driving Bridge IP Clock source for Slave interface Clock source for Master interface	Options Master interface Master interface Microblaze_0 (Periph) Pridge IP Clock source for driving Bridge IP Clock source for Slave interface Clock source for Master interface /clk_wiz_1/clk_out1 (100 MHz)

Step 7:

Connect interrupt output lines from "AXI Uartlite" to the "Concat" block as shown in the below figure. Select the "Validate Design" option from the "Tools" menu to make sure that connections are correct.



Step 8:

Select the "Validate Design" option from the "Tools" menu to make sure that connections are correct.



Step 9:

Right-click "Hello_world" in the "Sources" window and select "Create HDL Wrapper" from the popup menu. Click "OK" on the window that appears to finish generating a wrapper.



Step 10:

Click "Generate Bitstream" under the "Program and Debug" section to synthesize, implement, and generate a bitstream.


Step 11:

Once the implementation and generation of the bitstream are completed, we need to export the hardware along with the bitstream. Go to the "File" menu and select "Export->Export Hardware ". Select the "Include bitstream" checkbox and click "OK" in the "Export Hardware" wizard.

		Add Sources	Alt+A BLC	CK DESIGN - Hello_world		
	1	Save Block Design	Ctrl+S	Sources × Design		
		Save Block Design As	benn	Q 素 ♦ +		
		Close Block Design		V 🖨 Design Sources (1)		
		Chec <u>k</u> point	> 1115	> • Hello_world_\		
		<u>C</u> onstraints) b	Constraints Simulation Sources		
		Simulation Waveform	F	> a sim_1 (1)		
	1	<u>I</u> P	F	> 🚍 Utility Sources		
		Text E <u>d</u> itor	+			
		I <u>m</u> port	+			
		Export		Export <u>H</u> ardware		
	1	Print	Ctrl+P	Export Block Design		
		Exit		Export Simulation		
			1 1			
Output Set the platform proper Pre-synthesis This platform in	ies to inform dow	vnstream tools of the int re specification for down	ended use of t stream softwa	he target platform's hard ire tools.	ware design.	Σ
Output Set the platform proper Pre-synthesis This platform in Include bitstrea This platform in	ies to inform dow cludes a hardwar n cludes the compl	vnstream tools of the int re specification for down lete hardware implemer	ended use of t stream softwa itation and bits	he target platform's hard are tools. stream, in addition to the	ware design. hardware specific	ation for
Output Set the platform proper Pre-synthesis This platform in Software tools.	ies to inform dow cludes a hardwar n cludes the compl	vnstream tools of the int re specification for down lete hardware implemer	ended use of t stream softwa	he target platform's hard are tools. stream, in addition to the	ware design. hardware specific	ation for
Output Set the platform proper Pre-synthesis This platform in Include bitstrea This platform in software tools.	ies to inform dow cludes a hardwar n cludes the compl	vnstream tools of the int re specification for down lete hardware implemer	ended use of t stream softwa	he target platform's hard are tools. stream, in addition to the	ware design. hardware specific	ation for
Output Set the platform proper Pre-synthesis This platform in Software tools.	ies to inform dow cludes a hardwar n cludes the compl	vnstream tools of the int re specification for down lete hardware implemen	ended use of t stream softwa	he target platform's hard are tools. stream, in addition to the	ware design. hardware specific	ation for
Output Set the platform proper Pre-synthesis This platform in Include bitstrea This platform in software tools.	ies to inform dow cludes a hardwar n cludes the compl	vnstream tools of the int re specification for down lete hardware implemer	ended use of t stream softwa	he target platform's hard are tools. stream, in addition to the	ware design. hardware specific	ation for
Output Set the platform proper Pre-synthesis This platform in Include bitstrea This platform in software tools.	ies to inform dow cludes a hardwar n cludes the compl	vnstream tools of the int re specification for down lete hardware implemer	ended use of t	he target platform's hard are tools. stream, in addition to the	ware design. hardware specific	ation for
Output Set the platform proper Pre-synthesis This platform in Software tools.	ies to inform dow cludes a hardwar n cludes the compl	vnstream tools of the int re specification for down lete hardware implemer	ended use of t Istream softwa	he target platform's hard are tools. stream, in addition to the	ware design. hardware specific	ation for
Output Set the platform proper	ies to inform dow cludes a hardwar n cludes the compl	vnstream tools of the int re specification for down lete hardware implemer	ended use of t	he target platform's hard are tools. stream, in addition to the	ware design. hardware specific	ation for
Output Set the platform proper	ies to inform dow cludes a hardwar n cludes the compl	vnstream tools of the int re specification for down lete hardware implemen	ended use of t	he target platform's hard are tools. stream, in addition to the	ware design. hardware specific	ation for
Output Set the platform proper	ies to inform dow cludes a hardwar n cludes the compl	vnstream tools of the int re specification for down lete hardware implemer	ended use of t	he target platform's hard are tools. stream, in addition to the	ware design. hardware specific	ation for
Output Set the platform proper	ies to inform dow cludes a hardwar n cludes the compl	vnstream tools of the int re specification for down lete hardware implemer	ended use of t	he target platform's hard are tools. stream, in addition to the	ware design. hardware specific	ation for
Output Set the platform proper	ies to inform dow	vnstream tools of the int re specification for down lete hardware implemen	ended use of t	he target platform's hard are tools. stream, in addition to the	ware design. hardware specific	ation for
Output Set the platform proper	ies to inform dow cludes a hardwar n cludes the compl	vnstream tools of the int re specification for down lete hardware implemer	ended use of t	he target platform's hard are tools. stream, in addition to the	ware design. hardware specific	ation for
Output Set the platform proper	ies to inform dow cludes a hardwar n cludes the compl	vnstream tools of the int re specification for down lete hardware implemer	ended use of t	he target platform's hard are tools. stream, in addition to the	ware design. hardware specific	ation for
Output Set the platform proper	ies to inform dow	vnstream tools of the int re specification for down lete hardware implemen	ended use of t	he target platform's hard are tools. stream, in addition to the	ware design. hardware specific	ation for

Step 12:

Select Launch Vitis IDE from the Tools menu.



Step 13:

After Vitis Unified IDE window opens, click on "Open Workspace" and select necessary folder to keep the Vitis files.



Step 14:

Create a new platform for the project, by selecting "Create Platform Component", click "Next", in the Flow tab select the XSA file saved using the step 11 and finally click "Next" and "Finish" respectively.





After successful creation of the platform, build the platform.



Step 15:

Next create the Hello world Application component by selecting the "Hello world" template from the "examples",



In "Create Application Component" tab specify project name and location, click "Next".

Create Application Componen	nt - Hello World	×
Name and Location > H	lardware > Domain > Sysroot > Summary	
Name and Location		
Choose a name for your cor	mponent and specify a directory where component data files will be stored	
Component name	Hello	
Component location	C:\projects\Elbert_S7\Hello_world\VITIS v Br	rowse
Component will be created a	t C:\projects\Elbert_S7\Hello_world\VITIS\Hello	
Cancel		Next

Select newly created Platform and click "Next".

Create Application Component - Hello World	Create Application Component - Hello World					
Name and Location > Hardware > Domain > Summary						
Select Platform						
Platforms supporting the selected example from y	your repositories	s. To create a new	platform, us	e "File -> New Component -> Platform"		
≪ ¥ + − υ ρ						
NAME	BOARD	FLOW	VENDOR	ратн		
C:\projects\Elbert_S7\Hello_world\VITIS\HELLO_WO	RLD∖			t_S7\Hello_world\VITIS\HELLO_WORLD\export\HELLO_WORLD		
(1)						
A HELLO_WORLD	elbert	Embedded	xilinx.com	VITIS\HELLO_WORLD\export\HELLO_WORLD\HELLO_WORLD.x Info		
Cancel				Back	t	

Select the domain as "Standalone_microblaze_0" and click "Next" and click on "Finish".



When the Helloworld project is added successfully, build the project manually.



Step 16: Once the build is completed successfully, power up Elbert S7 FPGA Development Board using USB type C cable.

Step 17: Program the FPGA on Elbert S7 with a simple boot loop program by selecting the Program Device option from the Vitis menu.



Once the "Program Device" window opens click on "Program".

Program Device		×
Specify the bitstream a	and the ELF files that reside in BRAM memory.	
Project	Hello ~	
Connection	Local v	New
Bitstream/PDI	:\projects\Elbert_S7\Hello_world\VITIS\Hello_ide\bitstream\Hello_world_wrapper.bit	Browse Search
	Partial Bitstream	
BMM/MMI File	projects\Elbert_S7\Hello_world\VITIS\Hello_ide\bitstream\Hello_world_wrapper.mmi	<u>Browse</u> <u>Search</u>
Software Configuration	n	
PROCESSOR	ELF/MEM FILE TO INITIALIZE IN BLOCK RAM	
microblaze_0	bootloop	~
Skip Revision Check		
Cancel		Generate Program

Step 18:

Meanwhile, open any serial terminal program (such as PuTTY, Teraterm etc) and open the port corresponding to Elbert S7 with a 9600 baud rate (the default baud rate given in UART IP). Program the board by selecting the "Run."

✓ FLOW		≽	*
Component	🗰 Hello	~	錢
河 Build 🥪			
▷ Run			
🕸 Debug			

Step 19:

If everything went well, the application running on the board should print "Hello World" over the UART and should be displayed on the Serial Terminal application.



2. Controlling Onboard Peripherals (LEDs, Seven Segment Display, and PMOD) with Slide Switches

INTRODUCTION

This project demonstrates how to interface and control key onboard peripherals-**LEDs**, **Seven Segment Display**, and **PMOD connectors**-using the slide switches on the Elbert S7 FPGA board. It serves as a foundational experiment to understand how input signals (from switches) can control output peripherals in an FPGA-based system.

The design includes predefined behaviours:

- When **no switch is active**, the **seven-segment displays** continuously count from 0 to 9 (same digit on all displays), while the **LEDs** perform a running light pattern.
- When **specific switches are turned ON**, the system displays corresponding values:
 - **Switch 1 (SWO)**: The seven-segment shows **3**, and its binary representation (**00000011**) is shown on the LEDs.
 - Switch 2 (SW1): The seven-segment shows 5, and the LEDs display 00000101.
 - Switch 3 (SW2): The seven-segment shows 7, and the LEDs display
 00000111.
- When **Switch 4 (SW3)** is turned ON, **all PMOD pins go high**, enabling external module activation or signalling.

This project is designed as a basic interface demo. You are encouraged to **modify the functionality**-such as changing the numbers shown on the seven-segment display, customizing LED patterns, or controlling PMOD outputs differently-to suit your specific application needs or to experiment with your own designs.

By completing this project, you'll gain hands-on experience with I/O interfacing, logic control based on inputs.

STEP 1:

To create a new Vivado project specifically for the **Elbert S7** FPGA board, follow the procedure outlined in the section "<u>Getting Started with Vivado: Creating a New</u> **FPGA Project.**"

STEP 2:

After creating a new project successfully, In the Sources tab, Right-click '**Design Sources**' and click '**Add Sources**'. It will open a new '**Add Sources**' window.



Once the "Add Sources" window opens select "Add or create design sources" and click on "Next "

Add Sources	×
AMD Vivado ML Edition	Add Sources This guides you through the process of adding and creating sources for your project Add or greate constraints Add or create design sources Add or create gimulation sources
?	< <u>B</u> ack <u>Einish</u> Cancel

Step 3:

Download and extract the RTL source files from <u>here</u> and add them to the project by selecting 'Add Files' in the "Add or Create Design Sources" tab and click on "Finish".

.				
T ₂ = ⊪ ♥				
	Lico Add Film A	dd Diractoriae ar Croste F	ile buttons below	
	Use Add Flies, A	add Directories of Create F	lie buttons below	
				
	<u>A</u> dd Files	A <u>d</u> d Directories	<u>C</u> reate File	
Scan and add RTL include f	iles into project			
Copy <u>s</u> ources into project				

In Sources tab of Vivado, Right-Click on 'Constraints' and click 'Add Sources'.

 т. 		Properties	Ctrl+E
> 🖬 Con		Hierarchy Update	▶
> = 5im	C	Refresh Hierarchy	
> 🗀 Utili		IP Hierarchy	▶
		Copy Constraints Set	
		Edit Constraints Sets	
		Edit Simulation Sets	
	+	Add Sources	Alt+A
		Report IP Status	
Hierarch	y I	P Sources Libraries	Compile Order

Step 5:

Once the **"Add Sources**" tab opens select **"Add or create constraints**" and click on **"Next**".

Add Sources		×
AMD Vivado ML Edition	Add Sources This guides you through the process of adding and creating sources for your project Add or greate constraints Add or create design sources Add or create gimulation sources	
?	< Back Einish Ca	ncel

Step 6:

Download and extract the xdc file from <u>here</u> and add them to the project by selecting 'Add Files' in the "Add or Create Constraints" tab and click on "Finish".

Add Sources	×
Add or Create Constraints Specify or create constraint files for physical and timing constraint	to add to your project.
Specify constraint set: Constrs_1 (active)	
Use Add File	s or Create File buttons below
Add	Files Create File
(?)	<back next=""> Einish Cancel</back>

Step 7:

In Project Manager tab, click on 'Generate Bitstream'.

✓ SYNTHESIS
Run Synthesis
> Open Synthesized Design
✓ IMPLEMENTATION
Run Implementation
> Open Implemented Design
✓ PROGRAM AND DEBUG
Generate Bitstream
> Open Hardware Manager

Step 8:

Once the bitstream is successfully generated, close any "**Bitstream Generation Completed**" dialog which comes up asking for what to do next.

Bitstream Generation Completed	×	
i Bitstream Generation successfully completed.		
Next	_	
Open Implemented Design		
○ <u>V</u> iew Reports		
Open <u>H</u> ardware Manager		
O Generate Memory Configuration File		
Don't show this dialog again		
OK Cancel		

Step 9:

Click on 'Open target' and 'Auto Connect'.

HARDWARE MANAGER - unconnected					
1 No hardware target is open. Open target					
Handman	ø	Auto Connect			
пагамаге		Recent Targets			
0, ≚ ≑ Ø		Available Targets on Server	*		
		Open New Target			
			2		
		No content			

Step 10:

Right Click on the device (**xc7s50_0**) and select "**Program Device**" option.

Step 11:

Click "**Program**" and observe the output.

Program Device		×	
Select a bitstream prog optionally select a debu bitstream programming	ramming file and download it to your hardware device. You can Ig probes file that corresponds to the debug cores contained in the g file.		
Bitstre <u>a</u> m file: Debu <u>q</u> probes file: ✔ <u>E</u> nable end of st	ieven_segment/Seven_segment.runs/impl_1/Seven_segment.bit artup check	•••	
?	<u>P</u> rogram Ca	ncel	

3. HDMI OUTPUT EXAMPLE DESIGN

INTRODUCTION

HDMI (High-Definition Multimedia Interface) represents a significant advancement over the older VGA standard by offering a digital solution that integrates both highresolution video and audio into a single interface. Unlike VGA, which relies on analog signals, HDMI uses digital transmission to deliver superior picture and sound quality. HDMI achieves this by transmitting pixel data serially at ten times the pixel clock frequency through TMDS (Transition Minimized Differential Signalling). This method reduces the number of signal transitions, which helps minimize potential data errors and maintains signal integrity.

The tutorial focuses on demonstrating DVI-D output through the Elbert S7 FPGA Module. DVI-D, or Digital Visual Interface – Digital, is a variant of HDMI and shares the same electrical and physical layer specifications. As a result, HDMI cables can also carry DVI-D signals, meaning that HDMI monitors are fully compatible with DVI-D outputs.

STEP 1:

To create a new Vivado project specifically for the **Elbert S7** FPGA board, follow the procedure outlined in the section "<u>Getting Started with Vivado: Creating a New</u> <u>FPGA Project.</u>"

Step 2:

After creating a new project successfully, then in the Sources tab, Right-click '**Design Sources'** and click '**Add Sources**'. It will open a new '**Add Sources'** window.

ROJECT MANAGER	- HDI	MI		
Sources				? _ O Ľ X
Q ¥ ♦ H	-	2 0		٥
🖻 Design Sources	5			
> 🖻 Constraints		Properties	Ctrl+E	
∨		Hierarchy Update	•	
🖙 sim_1	С	Refresh Hierarchy		
> 🖻 Utility Sources		IP Hierarchy	•	
		Copy Constraints Set		
		Edit Constraints Sets		
		Edit Simulation Sets		
Hierarchy Libra	+	Add Sources	Alt+A	

Once the "Add Sources" window opens select "Add or create design sources" and click on "Next "

Add Sources		×
AMD Vivado ML Edition	Add Sources This guides you through the process of adding and creating sources for your project Add or greate constraints Add or create design sources Add or create simulation sources	
?	< <u>R</u> ack <u>Einish</u> Ca	incel

Step 3:

Download and extract the RTL source files from <u>here</u> and add them to the project by selecting 'Add Files' in the "Add or Create Design Sources" tab and click on "Finish".

Add or Create Design Sources Specify HDL, netlist, Block Design, and IP files, or directories containing those file types to add to your project. Create a new source file on disk
and add it to your project.
+ + +
Use Add Files, Add Directories or Create File buttons below
Add Hies Add Directones Create Hie
Scan and add KIL include tiles into project
Lopy sources into project

The HDMI interface has 3 pairs of differential data signals and 1 pair of differential clock signals:

- data_p[2:0] & data_n[2:0] : These are HDMI/DVI differential signals carrying the video data to be displayed on screen.
- clk_p & clk_n : HDMI pixel clock differential pair of signals.

First, VGA signals are generated inside vga module. Then the VGA signals are encoded to 10-bits per channel and the data is then serialised to 10x of pixel clock rate. Finally, the three channels along with pixel clock are driven out using TMDS differential drivers.

In the top module (dvid_test), the two submodules dvid and vga are instantiated. Clocking IP (clocking wizard) is used to generate clocks for VGA and DVI-D.

dvid_test: In this module, a "Clocking Wizard" IP core is instantiated to generate required clocks for VGA and DVI-D. A 100MHz clock from the onboard oscillator is provided as input, and following clocks are derived from it:

- clk_vga: 25MHz clock. This is the pixel clock frequency for 640×480@60Hz VGA resolution.
- clk_dvi & clk_dvin : 125 MHz clocks. clk_dvin is 180 degrees out of phase to clk_dvi. These clocks are used for serialization using ODDR2.

vga: VGA signals are generated in this module. This design generates VGA at 640×480@60 Hz resolution.

dvid: VGA signals and clocks are given as input to this module and the DVI TMDS signals are generated as the output. It uses TMDS_encoder module to generate TMDS signals. TMDS uses 8b/10b encoding in which the 8-bit color data (red, green & blue) generated in VGA module is converted to 10 bits. Then this data is serialised using ODDR2 (Double Data Rate primitive). The 10-bit TMDS data is generated at 25 MHz. ODDR2 uses 5 times the frequency of pixel clock (i.e. 125MHz) to serialize the 10-bit encoded data. Note that ODDR2 serialises 2-bits in 1 clock cycle of 125MHz clock. This serialised data is converted into differential signals in top module (dvid_test) using OBUFDS drivers.

Step 4:

Add Clocking Wizard by clicking on IP catalog in Project Manager, type '**clocking**' in search box and double-click '**Clocking Wizard**' IP. It will open customisation window for '**Clocking Wizard**'.

Project Summa	ry × IP Catalog ×					? 🗆	16
Cores Interf	aces						
Q ₹ ≑	🛛 🕫 🕹 🖌 🖉	Ø					۰
Search: Qr cloc	king	(2 matches)					
Name		^1 AXI4	Status	License	VLNV		
🗸 🗁 Vivado Re	epository						
🗸 🗁 FPGA F	Features and Design						
V 🖻 Clos	cking						
	Clocking Wizard	AXI4	Production	Included	xilinx.com:ip:clk_wiz:6.0		
Details							
Name: C	locking Wizard						â
Version: 6	i.0 (Rev. 14)						
Interfaces: A	XI4						~

Step 5:

In '**Clocking Options**' tab, give Component Name as '**clocking**' and primary clock port name as '**clk_in**'.

Documentation P Location C	Switch to Defaults	
IP Symbol Resource	Component Name clocking	\otimes
Show disabled ports	Board Clocking Options Output Clocks Port Renaming MMCM Settings Summary	
	Image: Spread Spectrum Minimize Output Jitter	~
	Dynamic Reconfig Dynamic Phase Shift Maximize Input Jitter filtering	
+ s_axi_lite	Safe Clock Startup	
	Safe Clock Startup Dynamic Reconfig Interface Optione Mrite DRP registers AXI4Lite DRP	
+ r_pol/tit + CL(PM_D) + CL(PM_D) + CL(PM_D) - restel	Safe Clock Startup Dynamic Reconfig Interface Optione AXI4Lite DRP Phase Duty Cycle Config Write DRP registers Input Clock Information	
+ c, cut, fut, D + cut, fut, D - tabulat - tabulat	Safe Clock Startup Dynamic Reconfig Interface Options AXI4Lite DRP Phase Duty Cycle Config Write DRP registers AXI4Lite DRP Input Clock Information Input Clock Port Name Input Frequency(MHz) Jitter Options	Input .
	□ Safe Clock Startup Dynamic Reconfig Interface Options ● AXI4Lite DRP ■ AXI4Lite DRP Input Clock Information Input Clock Port Name Input Frequency(MHz) Primary clk_in Question 100.000 © Secondary clk in	Input. • 0.010

In '**Output Clocks**' tab, enable 3 output clocks and provide their name, frequency as well as phase as shown in the image below. Click '**OK**'.

Documentation 📮 IP Location C	Switch to Defaults						
IP Symbol Resource	Component Name	clocking					
Show disabled ports	Board Clocki	ng Options	Output Clocks	Port	Renaming MM	ACM Settings Sum	ımary
	The phase is calc	ulated relative to	the active input	clock.			
	Output Clock	Port Name	Output Freq (I	MHz)		Phase (degrees)	
		CIV DVI	Requested		Actual	Requested	Actual
CLK_DVI 🗕	Clk_out1	CLK_DVI 💿	125.000	8	125.00000	0.000	0.000
– reset CLK_DVIn –	Clk_out2	CLK_DVIn 🛞	125.000	8	125.00000	0.000	0.000
– clk_in CLK_VGA –	Clk_out3	CLK_VGA 🛛 🛞	25.000	8	25.00000	0.000	0.000
locked 🗕	clk_out4	clk_out4	100.000		N/A	0.000	N/A
	clk_out5	clk_out5	100.000		N/A	0.000	N/A
	clk_out6	clk_out6	100.000		N/A	0.000	N/A
							1

Step 6:

In Sources tab of Vivado, Right-Click on 'Constraints' and click 'Add Sources'.



Step 7:

Once the "Add Sources" tab opens select "Add or create constraints" and click on "Next ".

Add Sources		×
AMD Vivado ML Edition	Add Sources This guides you through the process of adding and creating sources for your project Add or greate constraints Add or create design sources Add or create simulation sources	
?	< <u>B</u> ack <u>Next</u> <u>Finish</u>	Cancel

Step 8:

Download and extract the XDC file from <u>here</u> and add them to the project by selecting 'Add Files' in the "Add or Create Constraints" tab and click on "Finish".

Step 9:		
In Project Manager tab,	click on 'Generate Bitstream'.	
	✓ SYNTHESIS	
	Run Synthesis	
	> Open Synthesized Design	
	✓ IMPLEMENTATION	
	Run Implementation	
	> Open Implemented Design	
	Y PROGRAM AND DEBUG	
	Generate Bitstream	
	> Open Hardware Manager	

Step 10:

Once the bitstream is successfully generated, close any "**Bitstream Generation Completed**" dialog which comes up asking for what to do next.



Step 11:

Now click 'Open Hardware Manager' to program the FPGA.



Step 12:

Click on 'Open target' and 'Auto Connect'.

HARDWARE MANAGER - unconnected				
1 No hardware target is open. Open target				
Handurana	ø	Auto Connect		
naroware		Recent Targets		
0, ≚ ≑ Ø		Available Targets on Server	*	
		Open New Target		
			, 	
		No content		

Step 13:

Right Click on the device (**xc7s50_0**) and select "**Program Device**" option.

Step 14:

Click "**Program**" and observe the output.

Program Device	×
Select a bitstream prog select a debug probes programming file.	gramming file and download it to your hardware device. You can optionally file that corresponds to the debug cores contained in the bitstream
Bitstre <u>a</u> m file: Debug probes file: ✔ <u>E</u> nable end of s	C:/projects/Elbert_S7/HDMI/HDMI.runs/impl_1/dvid_test.bit
?	Program Cancel

Step 15:

Once Elbert S7 is successfully programmed, it should begin generating HDMI signals and the monitor should display a colourful pattern at 640×480 @ 60Hz resolution.



That was it! You can play with the vga module to output different patterns and try to generate higher resolutions as well.

4. DDR3 Mem test on ELBERT S7

INTRODUCTION

This article aims to guide readers on how to effectively utilize the DDR3 memory available on the Spartan-7 FPGA using the AMD Memory Interface Generator (MIG) 7 IP Core. The MIG 7 IP core is a powerful tool provided by AMD that simplifies the process of interfacing with external DDR3 memory by handling the complex timing and calibration requirements.

When working with the MIG 7 IP core, users are presented with two interface options:

User Interface: This is a straightforward wrapper built on top of the Native Interface. It simplifies communication with the DDR3 memory by providing clear signal naming and intuitive data flow control.

AXI4 Interface: This interface follows the AXI4 protocol, which is widely used in FPGA designs for memory-mapped transactions. It allows seamless integration with AXI-based designs, providing better scalability and compatibility with IP cores that utilize the AXI protocol.

In this tutorial, we'll focus on how to test the DDR3 memory using the Memory Tests Template available in Vitis. This test template is a convenient way to verify memory functionality, ensuring that data can be reliably written to and read from the DDR3 memory. By following this guide, you'll gain insights into configuring the MIG 7 IP core, integrating it into your Vivado design, and performing practical memory tests on the Spartan-7 FPGA.

Whether you're a beginner or an experienced FPGA developer, this article will provide clear steps and explanations to help you successfully interface with DDR3 memory on the Spartan-7 platform.

STEP 1:

To create a new Vivado project specifically for the **Elbert S7** FPGA board, follow the procedure outlined in the section "<u>Getting Started with Vivado: Creating a New</u> <u>FPGA Project.</u>"

Step 2:

After creating a new project, Under the "Flow Navigator" panel, click "Create Block Design" under the IP Integrator section. Enter a name for the block design and click "OK". An empty block design will be created.

	Create Block Design	n > of block design.	<
	Design name: Directory: Specify source set: ?	DDR ©	
Step 3:	Jm	ato	Lab®

In the Diagram window, right-click and select "Add IP" from the popup menu. Search for "MicroBlaze" & "AXI Timer" and add them to the design by double-clicking them.

Search: Q- MICR (6 matches)
👎 MicroBlaze
👎 MicroBlaze Debug Module (MDM)
👎 MicroBlaze Debug Module (MDM) V
👎 MicroBlaze MCS
MicroBlaze MCS V
👎 MicroBlaze V
ENTER to select, ESC to cancel, Ctrl+Q for IP details

Search: Q- AXI TI (6 matches)
👎 AXI4-Stream Verification IP
👎 AXI Multi Channel Direct Memory Access
👎 AXI Sideband Utility
👎 AXI Timebase Watchdog Timer
👎 AXI Timer
👎 AXI Verification IP
ENTER to select, ESC to cancel, Ctrl+Q for IP details

Step 4:

Click "Run Block Automation" present in the "Designer Assistance available" bar on the top left corner of the window to complete the design. Select the settings as shown in the following image.

Q, ¥ ♦	Description
✓ III Automation (1 out of 1 selected) ✓ ♥ microblaze_0	MicroBlaze connection automation generates local memory of selected size, and caches can be configured. MicroBlaze Debug Module, Peripheral AXI Interconnect, Interrupt Controller, a clock source, Processor System Reset are added and connected as needed. A preset MicroBlaze configuration can also be selected. Information about the options can be found in the tooltips.
	Preset None Local Memory 64KB Local Memory ECC None Cache Configuration None Debug Module Debug Only Peripheral AXI Port Enabled Import Controller New Clocking Wizard
(?)	OK Cancel

Click the "Board" tab. The default peripherals available for the Elbert board will be listed as shown below.



Add DDR3 SDRAM and USB UART to the design by double-clicking the corresponding peripherals.

Step 6:

Double-click on the "Clocking Wizard" IP block and change the settings as shown below. In the "Output Clocks" section, set clk_out1 frequency to 100 MHz and clk_out2 to 200MHz.

P Symbol Resource Show disabled ports Component Name (dk_wtz_1 Board Clocking Options Output Clocks MMCM Settings Summary Associate P interface with board interface Board interface Clocking Options View 1000 (Clocking Options) Image: Distribution of the provide of the provi	Clocking Wizard (6.0)			
P Symbol Resource Component Name dk_wt_1 Board Clocking Options Output Clocks MMCM Settings Summary Associate P Interface With board Interface Board Interface Board Interface CLK_N1 CLK_N1 sys dock < CLK_IN1 Clustom Clk_in1 locked Interface Board Interface ULK_in1 locked Interface Interface	Documentation 🚡 IP Location			
Board Clocking Options Output Clocks MMCM Settings Summary Associate IP interface with board interface Board Educe Clocking URL_N1 sys dock CLK_N1 sys dock CLK_N1 Custom CLK_in1 locked Iteration Clk_in1 locked Iteration Iteration Iteration Iteration Iteration <td< th=""><th>IP Symbol Resource</th><th>Component Name clk_wiz_1</th><th></th><th></th></td<>	IP Symbol Resource	Component Name clk_wiz_1		
Associate P interface With board interface CLK_IN1 sys dock CLK_IN2 Custom EXT_RESET_IN reset Clear Board Parameters Clear Board Parameters	Show disabled ports	Board Clocking Options Output Clocks MMCM Settings Sum	nmary	
reset clk_in1 locked		Associate IP interface with board interface	Poard Interface	
reset clk_out1 clk_in1 locked		CLK IN1	sys clock	
reset clk_out1 clk_in1 locked		CLK IN2	Custom	•
reset clk_out1 clk_in1 locked		EXT RESET IN	reset	•
	rocot olk out1			

nbol Resource	Component Name	clk_wiz_1							
ow disabled ports									
	Board Clockin	g Options O	utput Clocks MMC	M Settings Su	Immary				
	The phase is calo	culated relative f	to the active input clock	k.	Dhana (damaaa)		Duty Custa (
	Output Clock	Port Name	Requested	Actual	Requested	Actual	Requested	Actual	Drives
	Clk_out1	clk_out1	100.000 🛞	100.00000	0.000 📀	0.000	50.000	50.0	BUFG
	Clk_out2	clk_out2	2þ0.000 📀	100.00000	0.000	0.000	50.000	50.0	BUFG
	Clk_out3	clk_out3	100.000	N/A	0.000	N/A	50.000	N/A	BUFG
	clk_out4	clk_out4	100.000	N/A	0.000	N/A	50.000	N/A	BUFG
	clk_out5	clk_out5	100.000	N/A	0.000	N/A	50.000	N/A	BUFG
	clk_out6	clk_out6	100.000	N/A	0.000	N/A	50.000	N/A	BUFG
clk out1	clk_out7	clk_out7	100.000	N/A	0.000	N/A	50.000	N/A	BUFG
reset				4				1	
clk_out2 💻	USE CLOCH	SEQUENCING	G (Clocking Feedbac	k				
clk_in1				6		Cinnali			
iocked —	Output Cloc	k Sequenc	ce Number	source		Signali	1 <u>0</u>		
	clk_out1	1		Auto	matic Control On-Chip) Single-ended		
	clk_out2	1		Auto	matic Control Off-Chip		Differential		
	clk_out3	1		O Use	r-Controlled On-Chip				
	clk_out4	1		🔾 Use	r-Controlled Off-Chip				
		1							
		1							

Step 7: Remove the existing connection to sys_clk_i of the "MIG 7 Series" block and connect it to clk_out2.

Step 8: Run "Connection Automation" so Vivado can connect the blocks to make a complete system.

Run Connection Automation		×
Automatically make connections in your design the right.	by checking the boxes of the interfaces to connect. Select an interface on the left to display its configuration options on	λ
Q X ♦ V ✓ All Automation (7 out of 7 selected) V ✓ ♥ axi_time_0 V ♥ axi_uartitle_0 V ♥ axi_uartitle_0 V ♥ Taxi_uartitle_0 V ♥ axi_uartitle_0 V ♥ Taxi_uartitle_0 V ♥ S_AXI V ♥ Taxi_uartitle_0 V ♥ Taxi_uartitle_0 V ♥ S_AXI V	Select an interface pin on the left panel to view its options	
?	ОК Са	ncel

Click on "Run Block Automation" and select keep Classic Microblaze option as shown in the picture below.

. ጟ ≑	Description
All Automation (1 out of 1 selected) All Automation (1 out of 1 selected)	MicroBlaze conversion automation converts classic MicroBlaze processors to the RISC-V MicroBlaze V processor. The corresponding MicroBlaze Debug Module (MDM) is also converted.
	information about the options can be found in the toolitps. Options
	Keep Classic MicroBlaze Enable Compressed Instructions

Step 9: Connect interrupt output lines from "**AXI Timer**" and "**UARTLite**" to the "**Concat**" block as shown below figure. Select the "**Validate Design**" option from the **Tools** menu to make sure that connections are correct.



Step 10: Right-click "**ddr3**" in the "**Sources**" window and select "**Create HDL Wrapper**" from the popup menu. Click "**OK**" on the window that appears to finish generating a wrapper



Step 11:

Click **"Generate Bitstream**" under the **"Program And Debug**" section to synthesize, implement and generate a bitstream.

Run Implementation
> Open Implemented Design
 PROGRAM AND DEBUG
👫 Generate Bitstream
> Open Hardware Manager

Step 12: After generating the bitstream successfully, select **Export -> Export Hardware** from the **File menu**. Click **Next**.

Project Add Soyrces A Save Block Design C Save Block Design AS Qlose Block Design AS Qlose Block Design Checypoint Constraints Simulgion Waveform IP Text Editor Import	<pre>ppoils ppoils p</pre>	Window Lagout yew Sources ↓ Image: Second Se		
Export	•	Export <u>H</u> ardware		
Print C	Xrl+P	Export Block Design Export Bitstream File		

Select the "include bitstream" checkbox and click Next.

Expo	ort Hardware Platform	×
Outp Set the	ut e platform properties to inform downstream tools of the intended use of the target platform's hardware design.	Д
0	Pre-synthesis This platform includes a hardware specification for downstream software tools.	
۲	Include bitstream This platform includes the complete hardware implementation and bitstream, in addition to the hardware specification for software tools.	
	<back next=""> Einish Canc</back>	el

Export Hardwa	re Platform	×
Files		ភា
Enter the name of	f your hardware platform file, and the directory where the XSA file will be stored.	
XSA file name:	DDR_wrapper	8
Export to:	C:/projects/Elbert_S7/DDR3	⊗
	The XSA will be written to: C:\projects\Elbert_S7\DDR3\DDR_wrapper.xsa	
	< Back Next > Finish	Cancel

Step 13: Select Launch Vitis IDE from the Tools menu.



Step 14: After Vitis Unified IDE window opens, click on "**Open Workspace**" and select necessary folder to keep the Vitis files.



Step 15: Create a new platform for the project, by selecting "**Create Platform Component**", click "**Next**", in the Flow tab select the XSA file saved using the step 12 and finally click "**Next**" and "**Finish**" respectively.



	Create Platform Component	×
	Name and Location > Flow > OS and Processor > Summary	
	Select Platform Creation Flow	
	Create a platform component by selecting the hardware design and add software domains.	
•	Hardware Design Existing Platform	R
	Hardware Design (XSA) C:\projects\Elbert_S7\Hello_world\Hello_world_wrapper.xsa > Browse For Implementation	
	> Advanced Options	
	Cancel	Back Next

After successful creation of the platform, build the platform.



Step 16: Next create the DDR3_test Application component by selecting the "Memory tests" template from the "examples",



In "Create Application Component" tab specify project name and location, click "Next"

Create Application Component - Memory Tests							
Name and Location > Hardware > Domain > Sysroot > Summary							
Name and Location							
Choose a name for your co	Choose a name for your component and specify a directory where component data files will be stored						
Component name	memory_tests						
Component location	C:\projects\Elbert_S7\DDR3\vitis_unified \v Browse						
Component will be created a	at C:\projects\Elbert_S7\DDR3\vitis_unified\memory_tests						
Cancel		lext					

Select newly created Platform and click "Next".



Select the domain as "Standalone_microblaze_o" and click "Next" and click on "Finish"

reate Application Component - Memory Te	sts		×
Name and Location > Hardware > i	Domain > Summary		
Select Domain			
Choose a domain from the available doma	ins in the platform		
Name	Details		
standalone_microblaze_0 + create new	Name Display Name OS Processor	standalone_microblaze_0 standalone_microblaze_0 standalone microblaze_0	
Cancel			Back

When the Memory tests project is added successfully, build the project manually.

~	FLOV	N		≽	*
	Com	ponent	memory_tests	~	\$
	<i>?</i> ∕≈ e	Build			4
	⊳ F	Run			
	资口	Debug			

Step 17: Once the build is completed successfully, power up Elbert S7 using an USB type C cable

Step 18: Program the FPGA on Elbert S7 with a simple boot loop program by selecting the **Program Device** option from the **Vitis menu**.



Once the "Program Device" window opens click on "Program".

Program Device				×		
Specify the bitstream and the ELF files that reside in BRAM memory.						
Project	memory_tests ~					
Connection	Local ~	<u>New</u>				
Bitstream/PDI	rojects\Elbert_57\DDR3\vitis_unified\memory_tests_ide\bitstream\DDR_wrapper.bit	<u>Browse</u>	<u>Search</u>			
	Partial Bitstream					
BMM/MMI File	jects\Elbert_S7\DDR3\vitis_unified\memory_tests_ide\bitstream\DDR_wrapper.mmi	<u>Browse</u>	<u>Search</u>			
Software Configuration						
PROCESSOR	ELF/MEM FILE TO INITIALIZE IN BLOCK RAM					
microblaze_0	bootloop			~		
Skip Revision Check						
Cancel		Gener	ate P	rogram		

Step 19:

Meanwhile, open any serial terminal program (such as PuTTY, Teraterm etc) and open the port corresponding to Elbert S7 with a 9600 baud rate (the default baud rate given in UART IP). Program the board by selecting the "Run".



Step 20:

If everything went well, the application running on the board should print the memory testing Process over the UART and should be displayed on the Serial Terminal application.

B COM45 - PuTTY	-		×
Starting Memory Test Application			
NOTE: This application runs with D-Cache disabled.As a result,	cachelin	e req	uest
s will not be generated			
Testing memory region: mig 0			
Memory Controller: mig 0			
Base Address: 0x80000000			
Size: 0x10000000 bytes			
Memory Test Application Complete			
Successfully ran Memory Test Application			
			Ψ.
	,		

5. Gigabit Ethernet Example Design

INTRODUCTION

Ethernet is a Link Layer Protocol in the TCP/IP protocol stack between the physical and data link layer. It is the most widely used protocol for Local Area Networks (LANs). Every device on Ethernet is assigned a unique MAC address for communication. <u>Gigabit Ethernet</u> refers to various technologies developed for transmitting Ethernet frames at the rate of gigabits per second. The <u>Reduced Gigabit</u> <u>Media-Independent Interface (RGMII)</u> is used to interface the Ethernet IP core on FPGA with the Gigabit Ethernet PHY chip on Elbert S7. The Media Access Layer converts the packets into a stream of data to be sent while the Physical Layer converts the stream of data into electrical signals. RGMII provides a media-independent interface so that MAC and PHY can be compatible, irrespective of the hardware used. In this tutorial, the Numato Lab Elbert S7 FPGA Development Board is used to demonstrate a TCP/IP echo server application. The echo server application runs on lightweight IP (lwIP) TCP/IP stack.

STEP 1:

To create a new Vivado project specifically for the **Elbert S7** FPGA board, follow the procedure outlined in the section "<u>Getting Started with Vivado: Creating a</u> <u>New FPGA Project.</u>"

Step 2: After creating a new project successfully, In the Flow Navigator panel, select Create Block Design under IP INTEGRATOR. Enter a name for the block design and click OK. An empty block design will be created


Step 3: Go to Diagram window, right click and select "Add IP" from the popup menu. Search for "MicroBlaze" and add it to the design by double-clicking it.

Search: Q- mid (6 matches)
👎 MicroBlaze
👎 MicroBlaze Debug Module (MDM)
👎 MicroBlaze Debug Module (MDM) V
👎 MicroBlaze MCS
MicroBlaze MCS V
👎 MicroBlaze V
ENTER to select, ESC to cancel, Ctrl+Q for IP details

Click "Run Block Automation" present in the "Designer Assistance available" bar on the top left corner of the window to complete the design. Select the settings as shown in the following image. Click "OK" for Vivado to automatically configure the blocks for you.

Q ≥ ✓ All Automation (1 out of 1 selected)	Description MicroBlaze connection automation generates local memory of selected size, and caches can be configured	d.
✓ * microblaze_0	MicroBlaze Debug Module, Peripheral AXI Interconnect, Interrupt Controller, a clock source, Processor Syst are added and connected as needed. A preset MicroBlaze configuration can also be selected. Information about the options can be found in the tooltips. Options	em Reset
	Preset None V Local Memory 64KB V Local Memory ECC None V	
	Cache Configuration None Debug Module Debug Only Peripheral AXI Port Enabled	
	Interrupt Controller Clock Connection New Clocking Wizard	

Step 4: Double click "Clocking Wizard" IP and customize "Board" settings as shown in the following image.

ID Symbol Desource											
Show disabled ports		Component Nam	e CIK_WIZ_1								
		Board Clocki	ng Options	Output Clocks	имсм	Settings Su	mmary				
		Associate IP inte	erface with boar	d interface			Board Inte	arface			
		CLK_IN1					sys clock	enace			-
		CLK_IN2					Custom				•
		EXI_RESET_IP	•				reset				•
		Clear Board	Parameters								
- reset	clk_out1 💻										
- clk in1	locked -										
-	J										
Re-customize IP										ок	Cancel
Re-customize (P liocking Wizard (6.0) Documentation 💿 (P Location										OK	Cancel
Re-customize IP Siocking Wizard (6.0) Documentation 💿 IP Location										<u>ок</u>	Cancel
Re-customize IP Clocking Wizard (6.0) Documentation C IP Location		Component Nam	e cik_wiz_1							OK	Cancel
Re-customize IP locking Wizard (6.0) Documentation IP IP Location IP Symbol Resource Show disabled ports		Component Nam Board Clocki	e cik_wiz_1	Dutout Clocks		I Settings Su	mmary			ок	Cancel
Re-customize IP locking Wizard (6.0) Documentation IP Location IP Symbol Resource		Component Nam Board Clockii The chase is ca	e cik_wiz_1 Ig Options C	Dutput Clocks 1	MMCM	t Settings Su	mmary			<u>OK</u>	Cancel
Re-customize IP clocking Wizard (6.0) Documentation IP Location IP Symbol Resource Show disabled ports		Component Nam Board Clocki The phase is Co Dutrut Clock	e cik_wiz_1 ig Options c culated relative Port Name	Dutput Clocks / / to the active input - Output Freq (M	MMCM clock. [Hz]	t Settings Su	mmary Phase (degr	005)	Duty Cycle	(%)	Cancel
Re-customize IP clocking Wizard (6.0) Documentation IP Location IP Symbol Resource Show disabled ports		Component Nam Board Clockii The phase is ca Output Clock	e dk_wkz_1 tg Options C cutated relative Port Name dk out	Dutput Clocks 1 to the active input (Requested 100 pnn	MMCM clock. iHz)	I Settings Su Actual 100.00000	Phase (degr Requested 0.000	ees) Actual	Duty Cycle Requested 5 n non	(%) Actual 50 0	Cancel
Re-customize IP Icocking Wizard (6.0) Documentation IP PLocation IP Symbol Resource Show disabled ports		Component Nam Board Clockin The phase is ca Output Clock Ø dk_out1	e dk_wkz_1 tg Options C cutated relative Port Name dk_out1 dk_out1	Dutput Clocks 1 to the active input Output Frequested 100.000 200.001	MMCM clock. Hz)	t Settings Su Actual 100.00000 200.00000	Phase (dogr Requested 0.000	ees) Actual © 0.000 © 0.000	Duty Cycle Requested 50.000 50.000	(%) (%) Actual 50.0 50.0	Cancel
Re-customize IP Stocking Wizard (6.0) Documentation IP Location IP Symbol Resource Show disabled ports		Component Nam Board Clockin The phase is ca Output Clock Ø dk_out1 Ø dk_out2	e clk_wiz_1 19 Options C cutated relative clk_out1 clk_out2 clk_out2	Dutput Clocks 7 to the adve input. Output Freq (M Requested 100.000 200.000	MMCM clock. Hzj	t Settings Su Actual 100.00000 200.00000	Phase (degr Requested 0.000 0.000	ees) Actual S 0.000	Duty Cycle Requested 50.000 50.000	(%) (%) (%) (%) (%) (%) (%) (%)	Cancel
Re-customize IP Elocking Wizard (6.0) Documentation IP Location IP Symbol Resource Show disabled ports		Component Nam Board Clockii The phase is ca Output Clock Ø cik_out2 Ø cik_out2 @ cik_out4	e clK_wlz_1 ag Options C cutated relative clk_out1 clk_out2 clk_out3	Dutput Clocks 7 to the active input to Output Freq (M Requested 100.000 200.000 125.000	MMCM clock. HH2) S	t Settings Su Actual 100.00000 200.00000 100.00000	Phase (degr Requested 0.000 0.000 0.000	ees) Actual © 0.000 © 0.000 © 0.000 0.000 NoA	Dety Cycle Requested 50.000 50.000 50.000 50.000	(%) Actual 50.0 50.0 50.0 50.0 NNA	Cancel
Re-customize (P locking Wizard (6.0) Documentation C IP Location IP Symbol Resource		Component Nam Board Clocki The phase is ca Output Clock I ch could Ch could Ch could Ch could Ch could	e dK_wiz_1 Ig Options C cutated relative cit_out1 cit_out1 cit_out2 cit_out3 cit_out4	Dutput Clocks I to the active input. 0 Output Freq (M) 100.000 200.000 125.000 100.000 100.000	MMCM clock. Ht2) S S S S S S	t Settings Su Actual 100.0000 200.0000 NMA NMA	Phase (degr Requested 0.000 0.000 0.000 0.000 0.000	Actual © 0.000 © 0.000 © 0.000 NDA NDA	Duty Cycle Requested 50.000 50.000 50.000 50.000 50.000 50.000	(%) Actual 50.0 50.0 50.0 50.0 100A	Cancel
Re-customize IP Clocking Wizard (6.0) Documentation IP Location IP Symbol Resource Show disabled ports		Component Nam Board Clockil The phase is Ca Utput Clock Ø dk_out1 Ø dk_out2 Ø dk_out3 Ø dk_out4 Ø dk_out4	e clk_wtz_1 tg Options C cutated relative Port Name clk_out1 clk_out2 clk_out3 clk_out4 clk_out5	Dutput Clocks I to the advie input of the advie inp	MMCM clock. HZ2 S S S S S S S S S S S S S S S S S S S	1 Settings Su Actual 100.00000 200.00000 100.00000 NUA NUA	Phase (degr Requested 0.000 0.000 0.000 0.000 0.000 0.000	есеб Астиаl Солона	Daty Cycle Requested 50.000 50.000 50.000 50.000 50.000 50.000 50.000 50.000 50.000 50.000 50.000 50.000 50.000	(%) Actual 50.0 50.0 50.0 10/A N/A N/A	Cancel
Re-customize IP clocking Wizard (6.0) Documentation IP Location IP Symbol Resource Show disabled ports	clk_out1	Component Nam Board Clockii The phase is ca Output Clock I dk_out2 I dk_out3 I dk_out4 I dk_out5 I dk_out5 I dk_out7	e dk_wtz_1 g Options C Culated relative Port Name dk_out2 dk_out3 dk_out4 dk_out5 dk_out5	Dutput Clocks I to the active input 0 0.004pt FF qdl 100.000 100.000 100.000 100.000 100.000	MMCM clock. Hiz)	I Settings Su Actual 100.00000 200.00000 100.00000 N/A N/A N/A N/A	Phase (dogr Requested 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000	Actual © 0.000 © 0.000 © 0.000 © 0.000 © 0.000 № 1.00 № 1.00 № 1.00	Duty Cycle Requested 50.000 50.000 50.000 50.000 50.000 50.000 50.000 50.000 50.000 50.000 50.000 50.000 50.000 50.000	(%) Actual 50.0 50.0 50.0 100A 100A 100A	Cancel
Re-customize IP clocking Wizard (6.0) Documentation IP Location IP Symbol Resource Show disabled ports	cik_out1	Component Nam Beard Clockil The phase is ca Output Clock V dk_out1 V dk_out3 dk_out3 dk_out6 dk_out6 dk_out7	e cik_wiz_1 g Options c cutated relative cik_out1 cik_out3 cik_out4 cik_out4 cik_out5 cik_out6 cik_out7	Durbut Clocks I 0 the adve input 0 output Frequested 100.000 200.000 100.000 100.000 100.000 100.000 100.000 100.000	MMCM clock. HZZ S S S S S S S S S S S S S S S S S S	I Settings Su Actual 100.0000 200.0000 200.0000 100.0000 N/A N/A N/A N/A N/A	Phase (dogr Requested 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000	Actual Ø 0.000 Ø 0.000 Ø 0.000 Ø 0.000 Ø 1.00 Ø 1.00 Ø 1.00 Ø 1.00 Ø 1.00	Darty Cycle Requested 50.000 50.000 50.000 50.000 50.000 50.000 50.000 50.000 50.000 50.000 50.000 50.000 50.000	(%) (%) (%) (%) (%) (%) (%) (%)	Cancel
Re-customize IP clocking Wizard (6.0) Documentation IP Location IP Symbol Resource Show disabled ports	clk_out1	Component Nam Board Clockie The phase is ca Output Clock Ø dk_out3 Ø dk_out3 Ø dk_out3 Ø dk_out5 Ø dk_out5 Ø dk_out7	clk_ws_1 goptions C wgoptions C clk_out1 clk_out3 clk_out3 clk_out4 clk_out5 clk_out6 clk_out7 clk_out7	Datpat Clocks Image: Clock stars of the sta	MMCM clock. Hiz)	t Settlings Su Actual 100.00000 200.00000 104 N/A N/A N/A N/A N/A Socking Feedback	Phase (dogr Requested 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000	Actual δ 0.000 δ 0.000 δ 0.000 100 100 100 100	Duty Cycle Requested 50.000 50.000 50.000 50.000 50.000 50.000 50.000 50.000 50.000 50.000 50.000 50.000 50.000 50.000	(%) (%) (%) (%) (%) (%) (%) (%)	Cancel Ca
Re-customize IP licoking Wizard (6.0) Documentation IP Location IP Symbol Resource Show disabled ports	clk_out1 clk_out2 clk_out3	Component Nam Board Clockli The phase is ca Output Clock Ø dk_outs Ø dk_outs Ø dk_outs Ø dk_outs Ø dk_outs Ø dk_outs	e clk_wiz_1 ig Options C cutated relative dk_out1 clk_out2 clk_out3 clk_out4 clk_out5 cl	Dutput Clocks Image: Clock state input clockstate input clocks Image: Clock st	MMCCM clock.	t Settings Su Actual 100.00000 200.00000 100.00000 104 NA NA NA NA NA Source	Phase (dogr Requested 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000	Actual Θ 0.000 Θ 0.000 Θ 0.000 ΝΑ 1.04 ΝΑ ΝΑ ΝΑ ΝΑ ΝΑ ΝΑ ΝΑ ΝΑ ΝΑ ΝΑ	Duty Cycle Requested 50.000 50.000 50.000 50.000 50.000 50.000 50.000 50.000 50.000 50.000 50.000 50.000 50.000 50.000 50.000 50.000 50.000	(%) Actual 50.0 50.0 100A N0A N0A	Cancel Ca
Re-customize IP locking Wizard (6.0) Documentation IP PLocation IP Symbol Resource	clk_out1 clk_out2 clk_out3 locked	Component Nam Board Clockin The phase is ca Output Clock Ø dk_out1 Ø dk_out2 Ø dk_out3 Ø dk_out3 Ø dk_out5 Ø dk_out7 USE CLOC	e cik_wiz_1 1g Options C cutated relative Port Name cik_out1 cik_out2 cik_out3 cik_out4	Dutput Clocks I to the adve input. Output Freq (M Quiput Freq (M 100.000 100.000 125.000 1100.000 100.000 100.000 100.000 100.000 100.000 100.000 100.000 100.000 100.000 100.000 100.000	MMCCM clock. HHz) S S S S S S S S S S S S S S S S S S S	1 Settings Su Actual 100.00000 200.00000 100.00000 100.00000 100.00000 100.00000 100.00000 100.00000 100.00000 100.00000 100.00000 100.00000 100.00000 100.00000 100.00000 100.00000 100.00000000	Phase (dogr Requested 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.000000	Actual Image: Second	Duty Cycle Requested 50.000 50.000 50.000 50.000 50.000 50.000 50.000 50.000 50.000 50.000 50.000 50.000 50.000 50.000 50.000 50.000 50.000	OK (%) Actual 50.0 50.0 NUA NUA NUA	Cancel
Re-customize IP locking Wizard (6.0) Documentation IP Location IP Symbol Resource Show disabled ports reset clk_in1	clk_out1 clk_out2 clk_out3 locked	Component Nam Board Clocki The phase is ca Utiput Clock Ø dk_out3 dk_out3 dk_out4 dk_out6 dk_out7 Use cloce Dutput Clock dk_out7	e dk_wtz_1 tg Options C cutated relative dk_out1 dk_out2 dk_out3 dk_out4 dk_o	Dutput Clocks I to the active input to the active	MMCM clock. HHZ) × S = S = S = S = S = S = S = S = S = S =	t Settings Su Actual 100.00000 200.00000 100.00000 NMA NMA NMA NMA NMA Source © Autor	Phase (degr Requested 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.000000	ees) Actual © 0.000 © 0.0000 0.0000 0.000 0.00000 0.000000 0.00000 0.00000 0.0000 0.0000 0.0000 0.00000	Dety Cycle Requested 50.000 50.000 50.000 50.000 50.000 50.000 50.000 50.000 50.000 50.000 50.000 50.000 50.000 50.000 50.000 50.000 50.000 50.000	(%) Actual 50.0 50.0 50.0 10/A NAA NAA NAA	Cancel
Re-customize (P tocking Wizard (6.0) Documentation [] IP Location IP Symbol Resource Show disabled ports resot clk_in1	clk_out1 clk_out2 clk_out3 locked	Component Nam Board Clockil The phase is is ca Uutput Clock Ø dk_out2 Ø dk_out3 dk_out3 dk_out4 dk_out5 dk_out7 UUSE CLOC Disput Clock UUSE CLOC	e clk_wtz_1 tg Options C clculated relative clk_out3 clk_out3 clk_out4 clk_out4 clk_out4 clk_out4 clk_out4 clk_out4 clk_out4 clk_out4 clk_out4 clk_out4 clk_out4 clk_out7 clk_out7 ck Sequencin 1 1 1 1 1 1 1 1 1 1	Dutput Clocks I to the active input. Coldput Frequence 200.000 Coldput Frequence 100.000 100.000 100.000 Coldput Frequence G Coldput Frequence	MMCM clock. HZ Ccc	t Settings Su Actual 100.00000 200.00000 NMA NMA NMA NMA NMA NMA NMA NMA NMA NMA	Phase (degr Requested 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.00000 0.0000 0.000000	есэ) Астиаl © 0.000 © 0.000 №А №А №А №А №А №А №А №А	Dety Cycle Requested 50.000	(%) Actual 50.0 50.0 50.0 10.0	Cancel
Re-customize (P locking Wizard (6.0) Documentation in P Location IP Symbol Resource Show disabled ports	clk_out1 clk_out2 clk_out3 locked	Component Nam Board Clockii The phase is ca Output Clock Ø dk_out3 Ø dk_out3 Ø dk_out3 Ø dk_out4 Ø dk_out5 Ø dk_out5 Ø dk_out7 Ø USE CLOC Ø dk_out6 Ø dk_out7	e dk_wtz_1 g Options C culated relative Port Name ck_eut2 dk_out3 ck_out3 ck_out4 ck_out5 ck_o	Dutput Clocks I to the active input Output Fred 100.000 200.000 125.000 100.000 100.000 100.000 100.000 100.000 100.000 100.000 100.000 100.000	MMCM clock. Hiz)	I Settings Su Actual 100.00000 200.00000 100.00000 N/A N/A N/A N/A N/A N/A N/A N/A N/A Source € Autor Quere Quere	Phase (degr Requested 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.000000	еез) Астиа Фоло Фоло Астиа Фоло Фоло Астиа Фоло Фо	Darty Cycle Requested 50.000	(%) Actual 50.0 50.0 50.0 100A NUA NUA	Cancel
Re-customize IP tiocking Wizard (6.0) Documentation IP Location IP Symbol Resource Show disabled ports reset cik_in1	cik_out1 cik_out2 cik_out3 locked	Component Nam Beard Clocki The phase is ca Output Clock Clocki Cl	e cik_wiz_1 ig Options C ig Options C idk_out1 cik_out3 cik_out3 cik_out4 cik_out6 cik_out7 cik_seut7 cik_out6 cik_out6 cik_out7	Durput Clocks Image: Clock state	MMCM clock. Hiz)	It Settlings Su Actual 100.00000 200.00000 100.00000 NAA NAA NNA NNA NVA Oocking Feedback Source @ Autor © Autor User	Phase (dogr Requested 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.000000	Actual Ø 0.000 Ø 0.000 <td>Duty Cycle Requested 50.000</td> <td>(%) (%) (%) (%) (%) (%) (%) (%)</td> <td>Cancel</td>	Duty Cycle Requested 50.000	(%) (%) (%) (%) (%) (%) (%) (%)	Cancel
Re-customize IP ticcking Wizard (6.0) Documentation IP Location IP Symbol Resource III Show disabled ports reset clk_in1	clk_out1 clk_out2 clk_out3 locked	Component Nam Board Clocki The phase is ca Output Clock Ø dk_out3 Ø dk_out3 Ø dk_out3 Ø dk_out5 Ø dk_out5 Ø dk_out7 USE CLOC Ø dk_out7 USE CLOC	clk_ws_1 go potions C with an end of the second sec	Datput Clocks Image: Clock start	MMCM clock. Hiz)	t Settings Su Actual 100.00000 200.00000 100.00000 100.00000 100.00000 100.00000 100.00000 100.00000 100.00000 100.00000 100.00000000	Primary Phase (dogr Requested 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.000000	Actual O Φ 0.000 0.000 Φ 0.000 0.000 ΝΑ ΝΑ 1.0Α ΝΑ	Duty Cycle Requested 50.000	(%) Actual 50.0 50.0 N/A N/A N/A N/A	Cancel
Re-customize (P locking Wizard (6.0) Documentation IP PLocation P Symbol Resource	clk_out1 clk_out2 clk_out3 locked	Component Nam Board Clocki The phase is ca Output Clock Ø dk_outs Ø dk_outs	e clk_wiz_1 g Options C cutated relative dk_out3 clk_out3 clk_out3 clk_out4 clk_out5 clk_out5 clk_out5 clk_out5 clk_out5 clk_out5 clk_out6 clk_out1 clk_out5 cl	Dutput Clocks I to the adve input. Requested 100.000 I 200.000 I 100.000 I 100.000 I 100.000 I 100.000 I 100.000 I 100.000 I cce Number I	MMCM clock. HZ Clock	t Settings Su Actual 100.00000 200.00000 100.00000 100.00000 100.00000 100.00000 100.00000 100.00000 100.00000 100.00000 100.00000000	Phase (degr Requested 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.000000	Actual δ 0.000 0.000 0.000	Duty Cycle Requested 50.000	OK 6% 1 Actual 50.0 50.0 50.0 50.0 100.0 100.0 100.0 100.0 100.0 100.0 100.0 100.0	Cancel
Re-customize IP locking Wizard (6.0) Documentation IP Location IP Symbol Resource	clk_out1 clk_out2 clk_out3 locked	Component Nam Beard Cockie The phase is ca Utypus Cock Ø dk_out3 Ø dk_out3 Ø dk_out3 Ø dk_out4 Ø dk_out4 Ø dk_out4 Ø dk_out4 Ø dk_out7 USE CLOC	e clk_wlz_1 tg Options C cculated relative clk_out1 clk_out3 clk_out3 clk_out4 clk_out4 clk_out4 clk_out4 clk_out4 clk_out4 clk_out4 clk_out4 clk_out5 clk_out7 ck Sequent 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	Dutput Clocks I to the adve input. Requested 100.000 125.000 1100.000 125.000 100.000 100.000 100.000 100.000 100.000 100.000 100.000 100.000 100.000 100.000 100.000 100.000	MMCM clock.	1 Settings Su Actual 100.00000 200.00000 100.00000 100.00000 100.00000 100.00000 100.00000 100.00000 100.00000 100.00000 100.00000000	Phase (dogr Requested 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.000000	ees) / Actual South of the second of the	Duty Cycle Requested 50.000	(%) Actual 50.0 50.0 100.	Cancel

Click **OK** to customize the IP.

Step 5: Click the **Board** tab. The default peripherals available for the Elbert S7 board will be displayed.



Drag and drop **DDR3 SDRAM**, **USB UART**, and **Gigabit Ethernet PHY** into IP Canvas.

Ensure that **sys_clk_i** of Memory Interface Generator is connected to **clk_out2**.

Step 6: Click Run Connection Automation and select all.



Step 7:

Run Block Automation

• for AXI Ethernet and select "FIFO" for the AXI Streaming interface.

Run Block Automation		×
Automatically make connections in your design t	by checking the boxes of the blocks to connect. Select a block on the left to display its configuration options on the right.	2
Q X € ✓ Ø Al Automation (2 out of 2 selected) Ø # au_ethernet_0 Ø # microblaze_0	Description AXI Ethernet connection automation generates DMA or FIFO for TX and RX streaming interfaces of instance "axi_ethernet_0". GTX clock will be connected to 125MHz source /clk_wiz_1/clk_out3. REF clock will be connected to 200.00MHz source /clk_wiz_1/clk_out2. These clock connections are for Physical interface type "RGMII". Options Physical Interface Selection REF IF Connect AXI Streaming Interfaces to FIFO DMA FIFO	
?	ОК Сал	cel

• for Microblaze_o select "Keep Classic MicroBlaze" option and click "OK"

 All Automation (2 out of 2 selected) Image: All Automation (2 out of 2 selected) 	MicroBlaze conversion automation converts classic MicroBlaze processors to the RISC-V MicroBlaze V processor. The
✓ [‡] microblaze_0	corresponding MicroBlaze Debug Module (MDM) is also converted.
	Options
	Keep Classic MicroBlaze Finable Compressed Instructions

Step 8: Click **Run Connection Automation**. Select the **All Automation** option and click **OK**.

Run Connection Automation		×
Automatically make connections in your design the right.	by checking the boxes of the interfaces to connect. Select an interface on the left to display its configuration options on	2
Q X All Automation (1 out of 1 selected) ∨ ♥ ♣ axi_ethemet_0_fifo ♥ ⊕ \$_AXI	Select an interface pin on the left panel to view its options	
(?)	ок	Cancel

Step 9: Add AXI Timer into IP Canvas and click Run Connection Automation.

Q X All Automation (1 out of 1 selected) ✓ ♥ ♥ axi_timer_0 ♥ ♥ \$ AXI	Description Connect Slave interface (/axi_timer_0	/S_AXI) to a selected Master address space.
	Master interface Bridge IP Clock source for driving Bridge IP Clock source for Slave interface Clock source for Master interface	/microblaze_0 (Periph) v /microblaze_0_axi_periph v /clk_wiz_1/clk_out1 (100 MHz) v /clk_wiz_1/clk_out1 (100 MHz) v

Step 10: Customize the Concat IP block as shown below.

t(2.1) mentation IP Location w disabled ports Cor In0[0:0] In1[0:0] In1[0:0] dout[4:0] In3[0:0] dout[4:0]			
mentation IP Location w disabled ports Corr In0(0:0] In1(0:0] In12(0:0) dout[4:0]			E
w disabled ports			
In0[0:0] In1[0:0] In2[0:0] dout[4:0]	mponent Name microblaze_0_xlconcat		
In0[0:0] In1[0:0] In2[0:0] dout[4:0]	imber of Porte	[1] 1201	
In0[0:0] In1[0:0] In2[0:0] dout[4:0]	AUTO In0 Width 1	[1 - 4096]	
In0[0:0] In1[0:0] In2[0:0] dout[4:0]	AUTO In1 Width 1	[1 - 4096]	
In0[0:0] In1[0:0] In2[0:0] dout[4:0]	AUTO In2 Width 1	[1 - 4096]	
In0[0:0] In1[0:0] In2[0:0] dout[4:0]	AUTO In3 Width 1	[1 - 4096]	
In0[0:0] In1[0:0] In2[0:0] dout[4:0]	AUTO In4 Width 1	[1 - 4096]	
In1[0:0] In2[0:0] dout[4:0]			
In2[0:0] dout[4:0]			
In3[0:0]			
In4[0:0]			

Route the following connections to the inputs of the Concat block:

- **interrupt** on AXI Uartlite block
- **interrupt** on AXI Timer block
- interrupt on AXI-Stream FIFO
- interrupt and mac_irq on AXI 1G/2.5G Ethernet Subsystem



Make sure that the final design looks as shown above.

Step 11: Select the Validate Design option from the Tools menu to ensure that connections are correct.

Tool	s Rep <u>o</u> rts	<u>W</u> indow	Layout	View	Hel
Y	<u>V</u> alidate Desi	gn		F6	6
ž	Create and Pa Create Interfa <u>R</u> un Tcl Script Property Edito Associate ELJ Generate Mer Compile Simu	ackage New ce Definition t r E Files mory Config <u>u</u> ulation Lib <u>r</u> a	IP ration File ries	CI	trl+J
Q	Vivado Store Custom Com Launch Vitis I Language <u>T</u> e	mands DE mplates			Þ
•	Settings				

Step 12: In the **Sources window**, right-click on the design and select **Create HDL Wrapper**. Click **OK** in the dialog box that appears.

В	LOCK DE SIGN - Ethernet *		
erties	Sources × Design 9	Signals Board ? _ 🗆 🖸	
e File Prot	✓	net.bd)	
Source	> Constraints > Simulation Source > sim_1 (1)	Source Node Properties Ctrl+E Open File Alt+O	
	> 🗅 Utility Sources	Create HDL Wrapper	
		Generate Output Products Reset Output Products	

Step 13: Click **Generate Bitstream** under the **PROGRAM AND DEBUG** section of Vivado to synthesize, implement and generate the bitstream.

~	IMPLEMENTATION	
	Run Implementation	
	> Open Implemented Design	
~	PROGRAM AND DEBUG	
	Senerate Bitstream	

Step 14: After generating the bitstream successfully, select **Export -> Export Hardware** from the **File menu**. Click **Next**.



Select the "include bitstream" checkbox and click Next.

Export Hardware Platform	×
Output Set the platform properties to inform downstream tools of the intended use of the target platform's hardware design.	
 Pre-synthesis This platform includes a hardware specification for downstream software tools. Include bitstream This platform includes the complete hardware implementation and bitstream, in addition to the hardware specification for 	
software tools.	
< <u>B</u> ack <u>N</u> ext > <u>Finish</u> Cance	el

Provide the **XSA file name** and save it at a suitable **location**. Click **Next** and click **Finish** in the next dialog box.

Export Hardwa	re Platform	×
Files Enter the name o	f your hardware platform file, and the directory where the XSA file will be stored.	Σ
⊻SA file name: Export to:	Ethernet_wrapper C:/projects/Elbert_S7/Ethernet The XSA will be written to: C:\projects\Elbert_S7\Ethernet\Ethernet\Ethernet_wrapper.xSa	8
	< <u>B</u> ack <u>Next></u> <u>Finish</u>	Cancel

Step 15: Launch Vitis classic.

Note: In Vivado 2024.1, accessing Vitis via the tools menu inadvertently launches Vitis Unified instead of Vitis Classic, which is our preferred tool for project creation. To utilize Vitis Classic, it is necessary to launch it separately.

Step 16: In Vitis, IDE window select **Create Application Project** and click **Next** in the dialog box that appears.

A New Application Project			×
Create a New Application Project			•••
This wizard will guide you through the 4 steps of creating new application projects. 1. Choose a platform or create a platform project from Vivado exported XSA 2. Put application project in a system project , associate it with a processor 3. Prepare the application runtime – domain 4. Choose a template for application to quick start development			
Processor Domain App XSA			
A platform provides hardware information and software environment settings. Skip welcome page next time. (Can be reached with Back button))		
(7) < Back Next > Finish		Cance	el .

In the **Platform**, window select **Create a new platform from the hardware** Tab and import the **XSA file** which is already created (Provide XSA file location). Click **Next**.

Sele	t a platform from repository	tform from hardware (XSA)			
Har	vare Specification C:\projects\Elbert_S7\Ethernet\Ethernet wrappe	r.xsa			
	vck190				
	vmk180 zc702				
XSA	le: zc706 zcu102			Browse	
	zcu106 zed				
	C:\projects\Elbert_S7\Ethernet\Ethernet_wrapper	r.xsa			
Platf	m name: Ethernet_wrapper				

In the **Application Project Details** window, give an appropriate **name** for the Vitis Project and click **Next**. Click **Next** in the **Domain** window.

Select the **lwIP Echo Server** template from the list of available templates and click **Finish**.

Templates		
Select a template to create your project.		
Available Templates:		
Find:	⊞ ⊞	lwIP Echo Server
 Embedded software development templat Dhrystone Empty Application (C++) Empty Application(C) Hello World IMP Echo Server IMP TCP Perf Client IMP TCP Perf Server IMP UDP Perf Client IMP UDP Perf Server Memory Tests OpenAMP echo-test OpenAMP RPC Demo Peripheral Tests RSA Authentication App Zingo ECBI 	tes	The lwIP Echo Server application provides a simple demonstration of how to use the light-weight IP stack (wIP). This application sets up the board to use IP address 192.168.1.10 or IPv6 FE80:0:0:0:20A:35FF:FE00:102, with MAC address 00:0a:35:00:01:02. The server listens for input at port 7 and simply echoes back whatever data is sent to that port.

Step 17: Select Navigate to BSP Settings from Application Project Settings.

View current BSP settings, or configure settings like STDIO peripheral selection, compiler flags, SW intrusive profiling add/remove libraries, assign drivers to peripherals, change versions of OS/libraries/drivers etc. Navigate to BSP Settings	General		Options
	General Project name Platform: Runtime: Domain: CPU: OS: Hardware Spec	Ethernet Ethernet wrapper cpp standalone_microblaze_0 microblaze_0 standalone	Options View current BSP settings, or configure settings like STDIO peripheral selection, compiler flags, SW intrusive profiling, add/remove libraries, assign drivers to peripherals, change versions of OS/libraries/drivers etc. Navigate to BSP Settings
		citication: View processors, memory ranges and periphera	ls.

Select Board Support Package and click on Modify BSP Settings option.

type filter text 📄 🕀 🌵 🕷	Board Support Package								
 Ethernet_wrapper microblaze_0 standalone_microblaze_0 Board Support Package 	View current BSP settings, or configure settings like STDIO peripheral selection, compiler flags, SW intrusive profiling, add/remove libraries, assign drivers to peripherals, change versions of OS/libraries/drivers etc. Modify BSP Settings Reset BSP Sources								
	A BSP settings file is generated with the existing modifications done. All the sub	user options selected in t squent changes are applie	he settings dialog. To use exising setti ed on top of the loaded settings.	ngs, click the below link. This	operation clears any				
	Operating System								
	Version: 9.1 Description: Standalone is a simp	le, low-level software laye	r. It provides access to basic processo	r features such as caches, inte	rrupts and exceptions				
	Documentation: -		innent, such as standard input and oc	itput, proming, abort and exit.					
	Documentation: - Drivers Libraries Name	Driver	Documentation	Examples	·				
	Documentation: - Drivers Libraries Name axi ethernet 0	Driver aviethernet	Documentation	Examples					
	Documentation: - Drivers Libraries Name axi_ethernet_0 axi_ethernet 0 fifo	Driver axiethernet	Documentation Link Documentation Link	Examples Import Examples					
	Documentation: - Drivers Libraries Name axi_ethemet_0 axi_ethemet_0_fifo axi_ethemet_0_fifo	Driver axiethernet Ilfifo tmrctr	Documentation Documentation Link Documentation Link	Examples Import Examples Import Examples					
	Documentation: - Drivers Libraries Name au, ethernet_0 axi, ethernet_0, fifo axi, timer_0 axi, timer_0 axi, timer_0	Driver axiethernet Ilfifo tmrctr uartilite	Documentation Docume	Examples Import Examples					
	Documentation: - Drivers Libraries Name axi_ethemet_0 axi_ethemet_0_fifo axi_uartifie_0 microblaze 0 axi intc	Driver axiethernet Ilfifo tmrctr uartlite intc	Documentation Link Documentation Link Documentation Link Documentation Link Documentation Link	Examples Import Examples Import Examples Import Examples Import Examples					
	Documentation: - Drivers Libraries Name axi_ethemet_0 axi_ethemet_0_fifo axi_ethemet_0_fifo axi_ethemet_0_fifo axi_ethemet_0_microblaze_0_axi_intc microblaze_0_axi_intc	Driver axiethemet Ilfrio tmrctr uartite intc bram	Documentation Documentation Link Documentation Link Documentation Link Documentation Link Documentation Link	Examples Import Examples Import Examples Import Examples Import Examples Import Examples					
	Documentation: - Drivers Libraries Name axi, ethernet_0 axi_ethernet_0, fifo axi_itimer_0 axi_utife_0 microblaze_0_local_memory_dlm microblaze_0 local memory_dlm	Driver axiethernet Ilfifo tmctr uartitie intc bram bram	Documentation Documentation Link Documentation Link Documentation Link Documentation Link Documentation Link Documentation Link	Examples Import Examples Import Examples Import Examples Import Examples Import Examples Import Examples					
	Documentation: - Drivers Libraries Name axi, ethernet_0 axi, ethernet_0, fifo axi, timer_0 axi, utile_0 microblaze_0_local_memory_dlm microblaze_0_local_memory_dlm mig_Tseries_0	Driver axiethernet Ilfifo tmctr uardite intc bram bram mig_Tseries	Documentation Documentation Link Documentation Link Documentation Link Documentation Link Documentation Link Documentation Link	Examples Import Examples Import Examples Import Examples Import Examples Import Examples Import Examples Import Examples Import Examples					
	Documentation: - Drivers Libraries Name axi_ethernet_0 axi_ethernet_0_fifo axi_ethernet_	Driver axiethernet Iliffo tumctr uartite intc bram bram mig_7series	Documentation Documentation Link Documentation Link Documentation Link Documentation Link Documentation Link Documentation Link Documentation Link	Examples Import Example Import Examples Import Example Import Examples Import					

In the Board Support Package Settings window, select lwip (Iwip220) library, change the **dhcp_options** to "**false**" and ensure that "debug options" are "false".

Select **phy_link_speed** in **temac_adapter_options** as

CONFIG_LINKSPEED1000.

C L L L L L	C D 10 10 1				
Control various settings o	f your Board Support Package.				
 Overview 					
✓ standalone	Configuration for library: lwip220				
lwip220	Name	Value	Default	Type	Description
✓ drivers	api mode	RAW API (RAW API)	RAW API	enum	Mode of operation for IwIP (
microblaze_0	lwip tcp keepaliye	false	false	boolean	Enable keepalive processing
	no sys no timers	true	true	boolean	Drops support for sys timeou
	samii fixed link	false	false	boolean	Enable fixed link for GEM SGI
	socket mode thread prio	2	2	integer	Priority of threads in socket r
	use_axieth_on_zyng	1	1	integer	Option if set to 1 ensures axie
	use_emaclite_on_zyng	1	1	integer	Option if set to 1 ensures em
	> arp_options	true	true	boolean	ARP Options
	> debug_options	true	true	boolean	Turn on IwIP Debug?
	 dhcp_options 	true	true	boolean	Is DHCP required?
	lwip_dhcp	false	false	boolean	Is DHCP required?
	lwip_dhcp_does_acd_check	false	false	boolean	ACD check on offered addres
	> icmp_options	true	true	boolean	ICMP Options
	igmp_options	false	false	boolean	IGMP Options
	> lwip_ip_options	true	true	boolean	IP Options
	> ipv6_enable	false	false	boolean	IPv6 enable value
	> lwip_memory_options				Options controlling lwIP mer
	> mbox_options	true	true	boolean	Mbox Options
	> pbuf_options	true	true	boolean	Pbuf Options
	> stats_options	true	true	boolean	Turn on IwIP statistics?
	> tcp_options	true	true	boolean	Is TCP required ?
	> temac_adapter_options	true	true	boolean	Settings for xps-II-temac/Axi-
	> udp options	true	true	boolean	Is UDP required ?
	- 1-1				
?					
Board Support Package	: Settings				OK Cancel
Board Support Package oard Support Package Control various settings of	2 Settings age Settings f your Board Support Package.				OK Cancel
Board Support Package oard Support Package control various settings or overview	e Settings age Settings f your Board Support Package.				OK Cancel
 Board Support Package Control various settings or Overview standalone 	e Settings age Settings f your Board Support Package. Configuration for library: lwip220				OK Cancel
 Board Support Package Board Support Package Control various settings of Overview standalone Iwip220 	e Settings age Settings f your Board Support Package. Configuration for library: lwip220 Name	Value	Default	Туре	OK Cancel
 Board Support Package Board Support Package Control various settings of Overview standalone Iwip220 drivers 	: Settings age Settings f your Board Support Package. Configuration for library: lwip220 Name Socket mode thread prio	Value 2	Default 2	Type	OK Cancel Description Priority of threads in socket
Poard Support Package oard Support Package control various settings of v Overview v standalone iwip220 v drivers microblaze_0	e Settings age Settings f your Board Support Package. Configuration for library: Iwip220 Name Socket_mode_thread_prio use ayieth on zono	Value 2	Default 2	Type integer	OK Cancel Description Priority of threads in socket Option if set to 1 ensures avi
 Board Support Package Daard Support Package Control various settings or Overview standalone lwip220 drivers microblaze_0 	: Settings age Settings f your Board Support Package. Configuration for library: lwip220 Name socket_mode_thread_prio use_axieth_on_zynq use_axieth_on_zynq	Value 2 1	Default 2 1	Type integer integer	OK Cancel OEscription Priority of threads in socket Option if set to 1 ensures an Option if set to 1 ensures an
 Board Support Package Board Support Package Control various settings of Overview v standalone [wip220] v drivers microblaze_0 	s Settings age Settings f your Board Support Package. Configuration for library: lwip220 Name socket_mode_thread_prio use_axieth_on_zynq use_emaclite_on_zynq use_ention	Value 2 1 1	Default 2 1 1 tue	Type integer integer integer	OK Cancel Description Priority of threads in socket Option if set to 1 ensures axi Option if set to 1 ensures en APB Outloar:
Board Support Package Dard Support Package control various settings of Overview v standalone [wip220 v drivers microblaze_0	e Settings age Settings f your Board Support Package. Configuration for library: lwip220 Name Socket_mode_thread_prio use_axieth_on_zynq use_emaclite_on_zynq > arp_options	Value 2 1 1 true	Default 2 1 1 true true	Type integer integer integer boolean boolean	OK Cancel OK Cancel Description Priority of threads in socket Option if set to 1 ensures axi Option if set to 1 ensures en ARP Options TARP Options
 Board Support Package Daard Support Package Deard Support Package Control various settings of Overview v standalone Invip220 v drivers microblaze_0 	e Settings age Settings f your Board Support Package. Configuration for library: Iwip220 Name socket_mode_thread_prio use_axieth_on_zynq use_emaclite_on_zynq > arp_options > debug_options	Value 2 1 1 true true	Default 2 1 1 true true true	Type integer integer boolean boolean	OK Cancel OK Cancel Description Priority of threads in socket Option if set to 1 ensures axi Option if set to 1 ensures em ARP Options Turn on lwIP Debug? Unit of the time of time of time of the time of tim
 Board Support Package Deard Support Package Deard Support Package Control various settings of Overview standalone Iwip220 drivers microblaze_0 	s Settings age Settings f your Board Support Package. Configuration for library: lwip220 Name socket_mode_thread_prio use_axieth_on_zynq use_emaclite_on_zynq y arp_options > debug_options > debug_options	Value 2 1 1 true true true true true	Default 2 1 1 true true true true	Type integer integer boolean boolean	OK Cancel OK Cancel Description Priority of threads in socket Option if set to 1 ensures ax Option if set to 1 ensures en ARP Options Turn on IwIP Debug? Is DHCP required?
Poard Support Package Dard Support Package Control various settings of Overview v standalone [wip220 v drivers microblaze_0	e Settings age Settings fyour Board Support Package. Configuration for library: lwip220 Name socket_mode_thread_prio use_axieth_on_zynq use_emaclite_on_zynq arp_options debug_options dbug_options dbug_options inter_options	Value 2 1 1 true true true true true	Default 2 1 1 true true true true true	Type integer integer integer boolean boolean boolean	OK Cancel OK Cancel Description Priority of threads in socket Option if set to 1 ensures ax Option if set to 1 ensures en ARP Options Turn on INIP Debug? Is DHCP required? ICMP Options
 Board Support Package Doard Support Package Doard Support Package Control various settings of Overview standalone kwip220 drivers microblaze_0 	e Settings age Settings f your Board Support Package. Configuration for library: lwip220 Name socket_mode_thread_prio use_axieth_on_zynq use_emaclite_on_zynq > arp_options > debug_options > debug_options > idpc_options > igmp_options	Value 2 1 1 true true true true true false	Default 2 1 1 true true true true true false	Type integer integer integer boolean boolean boolean boolean	OK Cancel OK Cancel Description Priority of threads in socket Option if set to 1 ensures ax Option if set to 1 ensures en ARP Options Turn on lwIP Debug? Is DHCP required? ICMP Options IGMP Options
 Board Support Package Daard Support Package Control various settings of Overview v standalone lwip220 v drivers microblaze_0 	settings age Settings f your Board Support Package. Configuration for library: lwip220 Name socket_mode_thread_prio use_axieth_on_zynq use_axieth_on_zynq use_on_zynq > arp_options > debug_options > debug_options > debug_options > icmp_options > icmp_options > lip_options	Value 2 1 1 true true true true true true true true	Default 2 1 1 true true true true true true true true	Type integer integer boolean boolean boolean boolean boolean	OK Cancel OK Cancel OBSCIPION Description Priority of threads in socket Option if set to 1 ensures axi Option if set to 1 ensures axi Options Turn on IwIP Debug? Is DHCP required? Is CMP Options IGMP Options IP Options IP Options
 Board Support Package Board Support Package Control various settings of Overview v standalone [wip220] v drivers microblaze_0 	E Settings Age Settings fyour Board Support Package. Configuration for library: lwip220 Name socket_mode_thread_prio use_axieth_on_zynq use_emaclite_on_zynq use_emaclite_on_zynq of the bug_options ducp_options ducp_options imp_options imp_o	Value 2 1 1 true true true true true false	Default 2 1 1 true true true true true false true false	Type integer integer boolean boolean boolean boolean boolean boolean	OK Cancel OK Cancel Description Priority of threads in socket Option if set to 1 ensures ax Option if set to 1 ensures en ARP Options Turn on IMP Debug? Is DHCP required? ICMP Options IGMP Options IP Option IP Optio
 Board Support Package Dard Support Package Dard Support Package Dard Support Package Control various settings of Overview v standalone kvip220 v drivers microblaze_0 	Settings age Settings fyour Board Support Package. Configuration for library: lwip220 Name socket_mode_thread_prio use_axieth_on_zynq use_emaclite_on_zynq > arp_options > debug_options > debug_options > dep_options > idmp_options > imp_options > impo_options	Value 2 1 1 true true true false true false	Default 2 1 1 true true true true false true false	Type integer integer boolean boolean boolean boolean boolean boolean boolean	OK Cancel OK Cancel Description Priority of threads in socket Option if set to 1 ensures ax Option if set to 1 ensures en ARP Options Turn on IwIP Debug? Is DHCP required? ICMP Options IGMP Options IGMP Options IP Option IP Option IP Options IP Option IP Option IP Option IP Option IP O
 Board Support Package Daard Support Package Control various settings of Overview standalone Iwip220 drivers microblaze_0 	2 Settings 2 Settings 3 Settings 4 your Board Support Package. Configuration for library: Iwip220 Name Socket_mode_thread_prio use_axieth_on_zynq use_emaclite_on_zynq arp_options debug_options debug_options debug_options debug_options debug_options bipo_enable bipo_enable bipo_options cons c	Value 2 1 1 true true true true false true false true false	Default 2 1 1 true true true true false true false true false	Type integer integer boolean boolean boolean boolean boolean boolean boolean boolean boolean boolean boolean	OK Cancel OK Cancel OK Cancel OBSCIP
 Board Support Package Dard Support Package Control various settings of Overview standalone lwip220 drivers microblaze_0 	: Settings age Settings fyour Board Support Package. Configuration for library: lwip220 Name socket_mode_thread_prio use_axieth_on_zynq use_axieth_on_zynq use_axieth_on_zynq sap_options debug_options debug_options debug_options icmp_options icmp_options jicmp_options jivo_enable Mvip_inemory_options puto options mbox_options puto options puto options Supto_options su	Value 2 1 1 true true true true true true true false true false true	Default 2 1 1 true true true true true true true true	Type integer integer integer boolean boolean boolean boolean boolean boolean boolean boolean boolean	OK Cancel OK Cancel OK Cancel OBSCIPION OF CANCENT CAN
 Board Support Package Daard Support Package Control various settings or Overview standalone [wip220] drivers microblaze_0 	e Settings age Settings fyour Board Support Package. Configuration for library: lwip220 Name socket_mode_thread_prio use_axieth_on_zynq use_emaclite_on_zynq use_emaclite_on_zynq s arp_options dbeug_options b dbeug_options b icmp_options contemponents contempone	Value 2 1 1 true true true true false true false true false true true	Default 2 1 1 true true true false true false true false true false true true true true true	Type integer integer integer boolean boolean boolean boolean boolean boolean boolean boolean boolean boolean	OK Cancel OK Cancel OBC Cancel Cancel OBC Cancel Cancel OBC Cancel
 Board Support Package Dard Support Package Dard Support Package Control various settings of Overview standalone kwip220 drivers microblaze_0 	Settings age Settings fyour Board Support Package. Configuration for library: lwip220 Name socket_mode_thread_prio use_axieth_on_zynq use_emaclite_on_zynq > arp_options > debug_options > debug_options > debug_options > idmp_options > limp_options > lwip_ip_options > imp_options > imp_options > imp_options > imp_options > imp_options > bwjo_enable > lwio_toptions > pbuf_options > stats_options > tor, options > to	Value 2 1 1 true true true false true false true false true true true true	Default 2 1 1 1 true true true true false true false true	Type integer integer integer boolean boolean boolean boolean boolean boolean boolean boolean boolean boolean boolean boolean	OK Cancel OK Cancel OBSC Cancel Description Priority of threads in socket Option if set to 1 ensures ax Option if set to 1 ensures en ARP Options Turn on lwIP Debug? Is DHCP required? ICMP Options IGMP Options IGMP Options IP Options Turn on lwIP statistics? Is TCP required?
 Board Support Package Doard Support Package Control various settings of Overview standalone Iwip220 drivers microblaze_0 	e Settings age Settings f your Board Support Package. Configuration for library: lwip220 Name socket_mode_thread_prio use_axieth_on_zynq use_emaclite_on_zynq > arp_options > debug_options > debug_options > icmp_options > lwip_ip_options > ipv6_enable > bwip_options > pbuf_options > stats_options > top_options > top_options	Value 2 1 1 true true true false true false true false true true true true	Default 2 1 1 true true true false true false true false true false true false true	Type integer integer boolean boolean boolean boolean boolean boolean boolean boolean boolean boolean boolean boolean boolean boolean	OK Cancel OK Cancel OK Cancel OBSCIPION Priority of threads in socket Option if set to 1 ensures ax Option if set to 1 ensures ax Option if set to 1 ensures ax Options Turn on twiP Debug? Is DHCP required? ICMP Options IGMP Options IGMP Options IP Options IF Options ITurn on twiP statistics? Is TCP required ? Settings for vscill_temac/dx
 Board Support Package Daard Support Package Control various settings of Overview standalone [wip220] drivers microblaze_0 	Settings age Settings fyour Board Support Package. Configuration for library: lwip220 Name socket_mode_thread_prio use_axieth_on_zynq use_emaclite_on_zynq use_emaclite_on_zynq arp_options debug_options debug_options debug_options icmp_options binjo_ptions binjo_ptions binjo_ptions binjo_ptions binjo_ptions pbi_options pbi_options pbi_options pbi_options stat_options temac_adapter_options temac_adapter_options	Value 2 1 1 1 true true true true false true false true true true true true true true tru	Default 2 1 1 1 true true true false true false true true true true true true true tru	Type integer integer integer boolean boolean boolean boolean boolean boolean boolean boolean boolean boolean boolean boolean	OK Cancel OK Cancel OK Cancel OBC Cancel OCC
Poard Support Package pard Support Package control various settings or Overview v standalone hwip220 v drivers microblaze_0	Settings age Settings fyour Board Support Package. Configuration for library: lwip220 Name socket_mode_thread_prio use_emaclite_on_zynq arp_options debug_options debug_options debug_options debug_options b dicp_options b dicp_options b wip_ip_options b wip_ip_options b wip_ip_options b wip_ip_options b buf_options b buf_options b stats_options b tuf_options b tuf_options b tuf_options b tuf_options b tuf_options b tuf_options c = c_options	Value 2 1 1 true true true false true false true true true true true true true tru	Default 2 1 1 true true true false true false true	Type integer integer integer boolean boolean boolean boolean boolean boolean boolean boolean boolean boolean boolean boolean boolean boolean boolean boolean boolean boolean	OK Cancel OK Cancel OK Cancel OBSC Cancel OBSC Cancel OPTION OF CANCELER CANCELER OPTION OF CANCELER CANCELER OPTION OF CANCELER
 Board Support Package Dard Support Package Deard Support Package Dear	Settings age Settings fyour Board Support Package. Configuration for library: lwip220 Name socket_mode_thread_prio use_axieth_on_zynq use_emaclite_on_zynq arp_options > debug_options > debug_options > debug_options > debug_options > idmp_options > imp_options > imp_options > imp_options > imp_options > bwjp.jp_options > imp_options > bwjp.denable > lwip.arbitons > pbuf_options > stats_options > temac_adapter_options emac_number _n_m_coalesce _n_m_coalesce _n_m_coalesce	Value 2 1 1 true true true false true false true	Default 2 1 1 true true true false true	Type integer integer boolean b	OK Cancel OK Cancel OK Cancel OBSC Cancel
 Board Support Package control various settings of Overview standalone lwip220 drivers microblaze_0 	e Settings age Settings f your Board Support Package. Configuration for library: lwip220 Name socket_mode_thread_prio use_axieth_on_zynq use_emaclite_on_zynq use_emaclite_on_zynq use_emaclite_on_zynq ap_options debug_options debug_options debug_options b drop_options b inyo_enable b lwip_memory_options b mbox_options b state_options b temac_adapter_options emac_number n_m_coalesce n_n_x_descriptors	Value 2 1 true t	Default 2 1 1 true true true false true false true false true false true false true false true false true false true	Type integer integer boolean b	OK Cancel OK Cancel OK Cancel OK Cancel Description Priority of threads in socket Option if set to 1 ensures an Option if set to 1 ensures an ARP Options Turn on IwIP Debug? Is DHCP required? ICMP Options IGMP Options IGMP Options IP Options IP Options IP Options IP Options IP Options IP Options IF Options IF Options IF Options IF Options IF Options ITurn on IwIP statistics? Is TCP required? Settings for xps-II-temac/Ax Zyng Ethernet Interface nun Setting for RX Buffer Descrip
Poard Support Package pard Support Package control various settings or Overview v standalone wip220 v drivers microblaze_0	e Settings age Settings fyour Board Support Package. Configuration for library: lwip220 Name socket_mode_thread_prio use_axieth_on_zynq use_emaclite_on_zynq use_emaclite_on_zynq debug_options debug_options debug_options igmp_options igmp_options b lwip_inemory_options b lwip_inemory_options b mbox_options b top_options b top_options computed b top_options compu	Value 2 1 1 true true true false true	Default 2 1 1 true true true false true true true true false true true false 1 true true true true true true true true 1 64 1	Type integer integer integer boolean boolea	OK Cancel OK Cancel OK Cancel OK Cancel OCCANNERCENT
Poard Support Package oard Support Package control various settings of vorview v standalone iwip220 v drivers microblaze_0	Settings age Settings fyour Board Support Package. Configuration for library: lwip220 Name socket_mode_thread_prio use_axieth_on_zynq use_emaclite_on_zynq use_emaclite_on_zynq debug_options dhcp_options dhcp_options dhcp_options byip_ip_options byip_ip_options byip_options byip_options byip_options bybuf_options but_options but_options tota_coalesce n_tx_descriptors	Value 2 1 1 true true false true 1 64 1 64	Default 2 1 1 1 true true true false true false true true false false true false fal	Type integer integer integer boolean b	OK Cancel OK Cancel OK Cancel OBSC Cancel OBSC Cancel OPTION OF CANCERS CANCERS OPTION OF CANCERS CANCERS OPTION OF CANCERS CANCERS OPTION OF CANCERS CANCERS TURN ON INFORMATION IS OPTIONS IS OPTION
 Board Support Package Daard Support Package Daard Support Package Dotrol various settings of Overview standalone Iwip220 drivers microblaze_0 	2 Settings age Settings fyour Board Support Package. Configuration for library: lwip220 Name socket_mode_thread_prio use_axieth_on_zynq use_emaclite_on_zynq > arp_options > debug_options > debug_options > debug_options > debug_options > debug_options > bipo_entions > icmp_options > imp_options > imp_options > bivip_in_options > isop_options > buip_options > stats_options > tota_cadapter_options > temac_adapter_options mbox_options > temac_adapter_options mox_coalesce n_tx_coalesce n_tx_coalesce n_tx_coalesce n_tx_coalesce n_tx_coalesce n_tx_coalesce n_tx_descriptors phy_link_speed	Value 2 1 true true true true false true true true true false true true true true true true true true 1 64 1000 Mbps (CONFIG_LII	Default 2 1 1 1 true true true false true true false true true true true true true true tru	Type integer integer integer boolean b	OK Cancel OK Cancel OK Cancel OK Cancel October Contemporation Octob
 Board Support Package control various settings of Overview standalone lwip220 drivers microblaze_0 	s Settings age Settings fyour Board Support Package. Configuration for library: lwip220 Name socket_mode_thread_prio use_axieth_on_zynq use_emaclite_on_zynq use_emaclite_on_zynq of the set of	Value 2 1 true true true true true true true true true 1 1 64 1 64 1 64 1 64 1 64 1 64 1 64 1 64 1 64 1 64 1	Default 2 1 1 1 true true true false true false true false true false true false true true true true true true true tru	Type integer integer integer boolean b	OK Cancel OK Cancel OK Cancel OK Cancel OCTION Priority of threads in socket Option if set to 1 ensures axi Option if set to 1 ensures axi Option if set to 1 ensures axi Options Turn on IwIP Debug? Is DHCP required? ICMP Options IP Options IF Options ITurn on IwIP statistics? Is TCP required ? Settings for xps-II-temac/Ax Zynq Ethernet Interface num Setting for XX Interrupt coal Number of TX Buffer Descrip Setting for TX Interrupt coal Number of TX Buffer Descrip Setting for TX Interrupt coal Number of TX Buffer Descrip Setting for TX Interrupt coal Number of TX Buffer Descrip Iink speed as negotiated by 1 Offload TCP and IP Receive
Poard Support Package oard Support Package control various settings or voverview v standalone iwip220 v drivers microblaze_0	e Settings age Settings fyour Board Support Package. Configuration for library: lwip220 Name socket_mode_thread_prio use_axieth_on_zynq use_emaclite_on_zynq use_emaclite_on_zynq use_emaclite_on_zynq b arp_options b dhcp_options b dhcp_options b icmp_options b icmp_options b icmp_options b iptof_enable b lwip_memory_options b top_options b top_options computer any coalesce n_nx_coalesce n_tx_coalesce comp_to_tacksum_offload comp_to_tacksum_offload	Value 2 1 1 true true true false true	Default 2 1 1 1 true true true false true false true true false true false true false true true true true true true true tru	Type integer integer integer boolean	OK Cancel OK Cancel OK Cancel OK Cancel OCCAN OC
 Board Support Package Dard Support Package Dard Support Package Control various settings of Vorview standalone kwip220 drivers microblaze_0 	settings age Settings fyour Board Support Package. Configuration for library: lwip220 Name socket_mode_thread_prio use_axieth_on_zynq use_emaclite_on_zynq > arp_options > debug_options > debug_options > debug_options > debug_options > debug_options > debug_options > bipo_enable > lwip_ip_options > imp_options > buf_options > stats_options > top_options > top_op	Value 2 1 1 true true true false true false true talse ta	Default 2 1 1 1 true true true false true false true true false true false true true true true false true false false true true true true true true true tru	Type integer integer integer integer boolean	OK Cancel OK Cancel OK Cancel Intervention OK
 Board Support Package Doard Support Package Control various settings of Overview standalone Iwip220 drivers microblaze_0 	s Settings age Settings f your Board Support Package. Configuration for library: lwip220 Name socket_mode_thread_prio use_axieth_on_zynq use_emaclite_on_zynq use_emaclite_on_zynq use_emaclite_on_zynq ap_options debug_options debug_options debug_options b drop_options b inty_options components component	Value 2 1 1 1 true true true true true false true true true true true true false true true true true true true false true true true false true true true true false	Default 2 1 1 1 true true true false true false true false true false true false	Type integer integer boolean b	OK Cancel OK Cancel OK Cancel OK Cancel OCA CANCELERATION CANCELERATION OCA CANCELERATION CANCELERATION OCA CANCELERATION CANCELERATION OCA CANCELERATION CANCELERATION OCA CANCELERATION OCA CANCELERATION OCA CANCELERATION OCA CANCELERATION CONTRACT CANCELERATION CONTRACT CANCELERATION CONTRACT OCA CANCELERATION OCA CANCELERATION CONTRACT CONTR
 Board Support Package Daard Support Package Control various settings of Overview standalone [wip220] drivers microblaze_0 	Settings Age Settings fyour Board Support Package. Configuration for library: lwip220 Name socket_mode_thread_prio use_axieth_on_zynq use_emaclite_on_zynq vse_emaclite_on_zynq vse_emaclite_on_zyne vse_emaclite_on_zynq vse_emaclite_on_zynq vse_emaclite_on_zynq vse_emaclite_on_zynq vse_emaclite_on_zyne vse_emaclite_on_zyne vse_emaclite_on_zyne vse_emaclite_on_zyne vse_emaclite_on_zyne vse_emaclite_on_zyne vse_emaclite_on_zyne vse_emaclitte_on_zyne vse_emaclite_on_zyne vse_emaclite_on_zyne	Value 2 1 1 true	Default 2 1 1 1 true true true false true false true false true false	Type integer integer integer boolean boolea	OK Cancel OK Cancel OK Cancel OK Cancel OCT
 Board Support Package Deard Support Package Deard Support Package Control various settings of Overview standalone Iwip220 drivers microblaze_0 	settings age Settings fyour Board Support Package. Configuration for library: lwip220 Name socket_mode_thread_prio use_axieth_on_zynq use_emaclite_on_zynq arp_options debug_options debug_options debug_options debug_options debug_options debug_options b dhcp_options b dhcp_options b unp_options calcelectors n_rx_coalesce n_rx_descriptors n_tx_descriptors n_tx_descriptors tcp_jp_tc_checksum_offload tcp_int_speed tcp_int_speed tcp_int_speed tcp_int_speed tcp_int_speed tcp_int_checksum_offload tcp_autons b unp_ontence b unp_ontence tcp_ontence tcp_int_checksum_offload tcp_int_speed tcp_int_checksum_offload tcp_autons b unp_ontence	Value 2 1 1 1 true true true false true false true	Default 2 1 1 1 true true true false true true true false true true true true false	Type integer integer integer boolean boolea	OK Cancel OK Cancel OK Cancel OBC Cancel OBC Cancel OBC Cancel OBC Cancel OCC

After changing the library settings, click OK. vitis will update the BSP automatically. If that didn't happen for any reason, run a build manually.

NOTE: Vitis does not include built-in support for KSZ Ethernet PHY drivers. To enable compatibility, the xaxiemacif_physpeed file must be manually updated with the KSZ driver modifications. Replace the existing xaxiemacif_physpeed file in your project with the provided <u>file</u>, which includes the necessary changes to support KSZ PHY drivers. This ensures proper Ethernet functionality in your application.

After modifying the xaxiemacif_physpeed file Build the project.

Step 18: Once the build is completed successfully, power up Elbert S7 using an USB type C cable.

Step 19: Program the FPGA on Elbert S7 by selecting the **Program Device** option from the **vitis menu**.

Vitis Project Window Help Platform Repositories Platform Repositories Software Repositories Scan Repositories Examples Libraries	
🝌 Vivado Integration >	
XSCT Console Vitis Shell	
Senerate Linker Script	
Start/Stop Emulator	
🏥 Program Device	
Create Boot Image >	
📝 🛛 Program Flash	
Dump/Restore Memory	
Export Workspace To Unified IDE	

Project:	Ethernet system	~		
Connection	Local		New	
connection:	Local	~	INEW	
Device:	Auto Detect		Select	
Bitstream/PDI:	\${project_loc:Ether	rnet}/_ide/bitstream/Ethernet_wrapper.l	Search	Browse
Partial Bitstre	am			
BMM/MMI File:	\${project_loc:Ether	rnet}/_ide/bitstream/Ethernet_wrapper.i	Search	Browse
Software Config	juration			
Processor		ELF/MEM File to Initialize in Block RAM	N	
microblaze_0		bootloop		
O Shin Devision	2h h			
_) Skip Kevision (LNECK			
		Generate	Program	Cancel

Open the COM port corresponding to Elbert S7 in any serial terminal (PuTTY, Tera Term, etc.) with a 9600 baud rate. Now, right-click on the .elf file in Project Explorer and select "Launch on Hardware" as shown below.

1 - E C Segien - Mis - 13	\$< ▼ () ▼ // ▼ ⊡ Ø ■ ₹	þ (→ → → →	
È Explorer ⊠	E 😫 i		🖀 Ethernet_system 🛛 💥 Ethernet 💭 Ethernet_wrapper 🕅
 Ethernet_system [Ethernet_ Ethernet [standalone_m ※ Binaries Bincludes Bohng 	wrapper] icroblaze 0.1 New Move To System Project	>	type filter text Board Su Support Support Comparison of the data support Comparison of the data support Support Support
 > boost > construction 	 Paste Paste Delete Refresh Import Sources Export as Archive Build Project Clean Project Generate Linker Script C/C++ Build Settings 	Ctrl+V Delete F5	Standadore_Inicipitate_u Modify BSP Modify BSP existing mod Load BSP setting existing mod Load BSP sett Operating Sy Na Vers Descript Documentat
	Team Run As Debug As Properties	> Alt+Enter	
✔ Assistant 🛛	E 🗄 🖗 🔦 🛈 :	₩ 8 - -	Run Configurations

Observe the details displayed on the serial terminal.



Step 21: Connect the Ethernet cable to the board and the other end to the PC Ethernet port. Go to **Control Panel**. Go to **Network and Internet -> Network and Sharing Centre -> Change adapter settings**. Select "Change adapter settings". Right-click on Ethernet, click properties, and select "**IPv4**". Change the IPv4 address to **192.168.1.15** (any IP address can be used) and the default gateway to **192.168.1.1**.

🔄 Network Cor	nnections	- D X
$\leftarrow \rightarrow \lor$	↑ 💘 « Network and Internet → Network Con	nections \checkmark G Search Network Connections $\red P$
Organise 👻	Disable this network device Diagnose this conne	ction Rename this connection » 🗄 🗧 🔹 🗍 🥝
-	Ethernet NMware	Network Adapter VMware Network Adapter
· ·	Ethernet Properties	Internet Protocol Version 4 (TCP/IPv4) Properties
	Networking Sharing	General
×	Connect using:	You can get IP settings assigned automatically if your network supports this capability. Otherwise, you need to ask your network administrator for the appropriate IP settings.
	Config This connection uses the following items:	Obtain an IP address automatically
	Pleint for Microsoft Networks Provide Protocol Prile and Printer Sharing for Microsoft Networks OS Packet Scheduler Printe Bridge Driver	Dec dre following in address: 192.168.1.15 IP address: 255.255.255.0 Default gateway: 192.168.1.1
	Internet Protocol Version 4 (TCP/IPv4) Microsoft Network Adapter Multiplexor Protocol	Obtain DNS server address automatically Use the following DNS server addresses:
	Install Uninstall Proper	Preferred DNS server: 8 . 8 . 8 . 8
	Description	Alternative DNS server: 8 . 8 . 4 . 4
	Transmission Control Protocol/Internet Protocol. The def wide area network protocol that provides communication across diverse interconnected networks.	Validate settings upon exit Advanced
4 items 1	ОК	OK Cancel

Step 22: Open a telnet session with IP Address **192.168.1.10** (IP address as per main.c) at **port 7**, give input through the keyboard and observe the output. If you enter a character from the keyboard, you can observe the transmitted and echoed characters on telnet as shown.



6. SD card test

Introduction

In this tutorial, we'll explore the process of creating an SD card test project using Vivado and Vitis Unified IDE for the Elbert S7 FPGA Development Board. Our design will feature a MicroBlaze soft processor, which will be integrated with an SD card interface via the AXI bus. This project will allow us to perform basic read and write operations on the SD card, helping to verify the functionality of the SD card interface on the Elbert S7. Although MicroBlaze designs can utilize either PLB or AXI bus systems, we'll focus on the AXI bus for this tutorial. For detailed information on MicroBlaze and additional resources, including the datasheet, please visit <u>AMD's dedicated MicroBlaze page</u>.

STEP 1:

To create a new Vivado project specifically for the **Elbert S**₇ FPGA board, follow the procedure outlined in the section "<u>Getting Started with Vivado: Creating a New</u> <u>FPGA Project.</u>"

Step 2: After creating a project, In the "Flow Navigator" panel, select "Create Block Design" under the IP integrator section. Give an appropriate name (Eg: "SD_card ") to the design and click "OK ".



Step 3:

Go to Diagram window, right click and select "Add IP" from the popup menu. Search for "**MicroBlaze**" and add it to the design by double-clicking it.



Click "**Run Block Automation**" present in the "**Designer Assistance available**" bar on the top left corner of the window to complete the design. Select the settings as shown in the following image. Click "**OK**" for Vivado to automatically configure the blocks for you.

Q ¥ ♦		Description									
 ✓ All Automation ✓ ♥ microbla 	(1 out of 1 selected) aze_0	MicroBlaze connection au MicroBlaze Debug Modul are added and connected Information about the opt	tomation generat e, Peripheral AXI II d as needed. A pre ions can be found	es local nterconn eset Micr 1 in the to	memory o ect, Interru oBlaze cor oltips.	f selected s ipt Controll nfiguration	size, and c ler, a clock can also b	aches can be source, Proce e selected.	configured. ssor System F	Reset	
		Options	None	~							
		Local Memory	64KB V]							
		Local Memory ECC	None 🗸								
		Cache Configuration	None 🗸								
		Debug Module	Debug Only	~							
		Peripheral AXI Port	Enabled 🗸								
		Interrupt Controller									
		Clock Connection	New Clocking	Wizard	~						

Step 4:

Double click "**Clocking Wizard**" IP and customize "**Board**" and "**Output Clocks**" settings as shown in the following image.

ELBERT S7 HANDBOOK

🐂 Numato Lab' REV: V1.0

ing Wizard (6.0)									
mentation 🔛 IP Location									
mbol Resource	Component Name	e clk_wiz_1							
ow disabled ports	Poard Clockin	a Optione 0	utput Clocke	M Sottinge	ummany				
	Associate IP inte	rface with board		in octungo o	Juninary				
	IP Interface	nace with board	Intenace		Board I	nterface			
	CLK_IN1				sys clo	ick			
	EXT_RESET_IN	1			Custor	n n			
	Clear Board	Parameters							
reset clk_out1 - clk_in1 locked -									
								ОК	Ca
								ок	Ca
								ок	Ca
ustomize IP							_	ок	Ca
ustomize (P							(ок	Ca
ustomize IP Ing Wizard (6.0)							(ок	Ca
ustomize IP ing Wizard (6.0) imentation Di IP Location							(ок	Ca
ustomize IP ing Wizard (6.0) imentation 🚔 IP Location							(ок	Ca
ustomize IP Ing Wizard (6.0) mentation © IP Location mbol Resource	Component Nam	e cik_wiz_1					-	OK) Ca
ustomize IP Ing Wizard (6.0) Imentation C IP Location Imbol Resource Iow disabled ports	Component Nam	e clk_wiz_1						OK	Ca
ustomize IP Ing Wizard (6.0) Imentation P Location Imbol Resource Iow disabled ports	Component Nam Board Clockin	e dk_wiz_1 ag options 0	Putput Clocks MMC	M Settings St	ummary			ок	Ca
ustomize IP Ing Wizard (6.0) Imentation P Location Imbol Resource Intervention Inte	Component Nam Board Clockii The phase is ca	e ck_wiz_1 ng Options 0 cuciated relative t	Autput Clocks MMC to the active input clock	M Settings St	ummary Phase (dec	Irees)	Duty Cycle (%)	ок	
ustomize IP Ing Wizard (6.0) Imentation IP Location Imbol Resource Intervention Int	Component Nam Board Clockii The phase is ca Output Clock	e ck_wiz_1 ng Options 0 Incutated relative 1 Port Name	Autput Clocks MMC to the active input clock Output Freq (ME2) Requested	M Settings Si c Actual	ummary Phase (deg Requested	rees) Actual	Duty Cycle (%) Requested	OK	Drives
Istomize IP Ing Wizard (6.0) mentation IP Location mbol Resource ow disabled ports	Component Nam Board Clockii The phase is ca Output Clock I dk_out1	e ck_wiz_1 ng Options O Port Name ck_out1	Autput Clocks MMC to the active input cloc Output Freq (Miz) Requested 100.000	M Settings Si c Actual 100.00000	ummary Phase (deg Requested 0.000	rees) Actual	Duty Cycle (%) Requested 50.000	ок Асtual 50.0	Drives BUFG
stomize IP ng Wizard (6.0) mentation IP Location mbol Resource ow disabled ports	Component Nam Board Clockin The phase is ca Output Clock If it out i If it out i	e clk_wiz_1 ng Options O Port Name clk_out1 clk_SD	Autput Clocks MMC to the active input cloc Output Freq (M/z) Requested 100.000 © 32.000 ©	M Settings St C 2200000 32.00000	Ummary Phase (deg Requested 0.000 0.000	rrees) Actual ⊗ 0.000 ⊗ 0.000	Duty Cycle (%) Requested 50.000 50.000	ок Асtual 50.0 50.0	Drives BUFG BUFG
stomize P ng Wizard (6.0) mentation IP Location mbol Resource ovv disabled ports	Component Nam Board Clockin The phase is ca Output Clock @ cik_out1 @ cik_out2 cik_out2	e clk_wiz_1 by Options O cutated relative t Port Name clk_out1 clk_SD clk_out3	Autput Clocks MMC to the active input cloc Output Freq (Miz) Requested 100.000	M Settings 50 c Actual 100.00000 32.00000	Ummary Phase (deg Requested 0.000 0.000 0.000	rrees) Actual © 0.000 © 0.000 N/A	Duty Cycle (%) Requested 50.000 50.000 50.000	ок Асtual 50.0 50.0 М/А	Drives BUFG BUFG BUFG
stomize IP ng Wizard (6.0) mentation IP Location mbol Resource ov disabled ports	Component Nam Board Clockin The phase is ca Output Clock @ cik_out1 @ cik_out2 @ cik_out3 @ cik_out4	e clk_wiz_1 ng Options O Coulated relative t Port Name clk_out1 clk_SD clk_out3 clk_out4	Autput Clocks MMC to the active input cloc Output Freq (MHZ) Requested 100.000 © 32.000 © 100.000	M Settings 50 C Actual 100.00000 32.00000 NA NA	Phase (deg Requested 0.000 0.000 0.000 0.000	rres) Actual S 0.000 S 0.000 NA NA	Duty Cycle (%) Requested 50.000 50.000 50.000 50.000	ОК Астиа 50.0 50.0 М/А М/А	Drives BUFG BUFG BUFG BUFG
stomize IP ng Wizard (6.0) mentation IP Location mbol Resource ov disabled ports	Component Nam Board Clockin The phase is ca Dutput Clock @ cik_out1 @ cik_out2 @ cik_out3 @ cik_out4 @ cik_out5	e dk_wiz_1 ng Options O Port Name dk_out1 dk_SD dk_out3 dk_out4 dk_out5	Autput Clocks MMC to the active input cloc Output Freq MH2 Requested 100.000 © 32.000 © 100.000 100.000	M Settings 50 c Actual 100.00000 32.0000 NA NA NA NA	Phase (deg Requested 0.000 0.000 0.000 0.000 0.000	Actual © 0.000 © 0.000 NA NA NA NA	Duty Cycle (%) Requested 50.000 50.000 50.000 50.000 50.000 50.000	ОК Асtual 50.0 50.0 ИЛА ИЛА ИЛА	Drives BUFG BUFG BUFG BUFG BUFG
stomize IP ng Wizard (6.0) mentation IP Location mbol Resource ow disabled ports	Component Nam Board Clockin The phase is ca Dutput Clock @ dik_out1 @ dik_out2 @ dik_out3 @ dik_out4 @ dik_out5 @ dik_out6	e dk_wiz_1 g Options 0 cutated relative dk_out1 dk_SD dk_out3 dk_out3 dk_out5 dk_out5	Output Clocks MMC to the active input cloc Output Freq (MHz) Requested 100.000 32.000 3 100.000 100.000 100.000 100.000	M Settings Si Actual 100.00000 32.0000 N/A N/A N/A N/A	Phase (deg Requested 0.0000 0.00000 0.0000 0.0000 0.0000 0.00000 0.0000	Actual	Duty Cycle (%) Requested 50.000 50.000 50.000 50.000 50.000 50.000 50.000	ОК Асtual 50.0 50.0 NJA NJA NJA NJA	Drives BUFG BUFG BUFG BUFG BUFG BUFG
stomize IP ng Wizard (6.0) mentation IP Location mbol Resource ow disabled ports	Component Nam Board Clockin The phase is ca Output Clock @ dik_out1 @ dik_out2 @ dik_out3 @ dik_out4 @ dik_out5 @ dik_out7	e dk_wiz_1 g Options 0 cutated relative dk_out1 dk_s0 dk_out3 dk_out4 dk_out5 dk_out6 dk_out7	Autput Clocks MMC to the active input cloc Output Freq (MH2) Requested 100.000 32.000 3 100.000 3 100.000 3 100.000 3 100.000 3 100.000 3 100.000 3	M Settings Si C Actual 100.00000 32.0000 NA NA NA NA NA NA NA	Phase (deg Requested 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000	Actual © 0.000 © 0.000 NA NA NA NA NA NA NA NA NA NA NA NA	Duty Cycle (%) Requested 50.000 50.000 50.000 50.000 50.000 50.000 50.000 50.000	ОК Асtual 50.0 50.0 ИА ИА ИА ИА ИА ИА	Drives BUFG BUFG BUFG BUFG BUFG BUFG BUFG
stomize IP ng Wizard (6.0) mentation IP Location mbol Resource ow disabled ports	Component Nam Board Clockin The phase is ca Dutput Clock @ dik_out1 @ dik_out2 @ dik_out3 @ dik_out4 @ dik_out5 @ dik_out6 @ dik_out7	e dk_wiz_1 g Options 0 cutated relative dk_out1 dk_SD dk_out3 dk_out3 dk_out6 dk_out6 dk_out7	Output Clocks MMC to the active input cloc Output Freq (MHz) Requested 100.000 32.000 3 100.000 100.000 100.000 100.000	M Settings Si Actual 100.00000 32.0000 NIA NIA NIA NIA NIA NIA	Phase (deg Requested 0.0000 0.00000 0.0000 0.0000 0.00000 0.00000 0.000	Actual © 0.000 N/A N/A N/A N/A N/A N/A N/A N/A	Daty Cycle (%) Requested 50.000 50.000 50.000 50.000 50.000 50.000 50.000 50.000 50.000 50.000 50.000 50.000 50.000	ОК Асtual 50.0 50.0 ИЛА ИЛА ИЛА ИЛА ИЛА	Drives BUFG BUFG BUFG BUFG BUFG BUFG BUFG
ustomize IP Ing Wizard (6.0) Imentation IP Location Imbol Resource Index IP Continue Index IP Location	Component Nam Board Ctockin The phase is ca Output Clock @ dik_out1 @ dik_out2 @ dik_out3 @ dik_out4 @ dik_out5 @ dik_out6 @ dik_out7 USE CLOC	e dk_wiz_1 g Options 0 cutated relative Port Name dk_out1 dk_out3 dk_out3 dk_out4 ek_out5 dk_out6 dk_out7	Nutput Clocks MMC to the active input cloc Output Freq (MHz) Requested 100.000 32.000 32 100.000 100.000 100.000 100.000 100.000 100.000 100.000 100.000	М Settings Si 4000000 32.00000 NIA NIA NIA NIA NIA NIA NIA NIA	Phase (deg 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000	Actual © 0.000 N/A N/A N/A N/A N/A N/A N/A N/A	Duty Cycle (%) Requested 50.000 50.000 50.000 50.000 50.000 50.000 50.000 50.000 50.000 50.000 50.000 50.000 50.000	ОК Асtual 50.0 50.0 МА МА МА МА МА МА МА	Drives BUFG BUFG BUFG BUFG BUFG BUFG
ustomize IP Ing Wizard (6.0) Imentation IP Location Imbol Resource Index IP Content Index IP Location	Component Nam Board Ctockin The phase is ca Output Clock @ dk_out1 @ dk_out2 @ dk_out3 @ dk_out3 @ dk_out4 @ dk_out5 @ dk_out6 @ dk_out7	e dk_wiz_1 g Options O cutated relative I Port Name dk_out1 dk_out3 dk_out3 dk_out4 dk_out5 d	Utput Clocks MMC to the active input cloc Output Freq (MHz) Requested 100.000 22.000 23.000 100.000 100.000 100.000 100.000 100.000	M Settings Si C Actual 100.00000 32.0000 NIA NIA NIA NIA NIA NIA NIA NIA	Phase (deg 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000	rees) Actual 0.000 NA NA NA NA NA NA NA NA NA	Duty Cycle (%) Requested 50.000 50.000 50.000 50.000 50.000 50.000 50.000 50.000 50.000	ОК Асtual 50.0 50.0 МА МА МА МА	Drives BUFG BUFG BUFG BUFG BUFG
ustomize IP Ing Wizard (6.0) Imentation Imbol Resource Index	Component Nam Board Clockin The phase is ca Dutput Clock @ dik_out2 @ dik_out3 @ dik_out4 @ dik_out4	e dk_wiz_1 g Options O cutated relative I Port Name dk_out1 dk_out3 dk_out3 dk_out4 dk_out5 d	Nutput Clocks MMC to the active input cloc Output Freq (MHz) Requested 100.000 22.000 23.000 100.000 100.000 100.000 100.000 100.000 100.000	M Settings Si Actual 100.00000 32.0000 NIA NIA NIA NIA NIA NIA NIA NIA	Phase (deg 0.0000 0.0000 0.0000 0.000000	rrees) Actual O.000 NA NA NA NA NA NA Signalir	Duty Cycle (%) Requested 50.000 50.000 50.000 50.000 50.000 50.000 50.000 50.000 50.000	ОК Асtual 50.0 50.0 М/А М/А М/А М/А	Drives BUFG BUFG BUFG BUFG BUFG
ustomize IP ing Wizard (6.0) imentation IP Location imbol Resource iow disabled ports reset clk_out1 clk_out2 clk_out2 locked	Component Nam Board Clockin The phase is ca Output Clock @ dk_out1 @ dk_out3 @ dk_out3 @ dk_out7 @ USE CLOC Output Clo	e clk_wiz_1 g Options O ccutated relative I clk_out1 clk_out1 clk_out3 clk_out4 clk_out5 clk_out5 clk_out5 clk_out5 clk_out6 clk_out7 clk_out6 clk_out7	Nutput Clocks MMC to the active input cloc Output Freq (MHz) Requested 100.000 32.000 3 100.000 100.000 100.000 100.000 100.000 100.000 100.000 100.000 100.000 100.000 100.000 100.000	M Settings Si Actual 100.00000 32.0000 NA NA NA NA NA NA NA NA NA NA	Phase (deg 0.0000 0.0000 0.0000 0.00000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.000000	rrees) Actual	Duty Cycle (%) Requested 50.000 50.000 50.000 50.000 50.000 50.000 50.000 50.000 50.000 50.000 50.000 50.000 50.000	ОК Асtual 50.0 50.0 10/А 10/А 10/А 10/А 10/А	Drives BUFG BUFG BUFG BUFG BUFG
ustomize IP ing Wizard (6.0) Imentation IP Location Imbol Resource Intow disabled ports Intowed isabled ports Interest Cik_out1 Cik_out1 Cik_out2	Component Nam Board Clockin The phase is ca Output Clock @ dk_out1 @ dk_out3 @ dk_out4 @ dk_out7 @ USE CLOC Output Clo dk_out7	e (k_wiz_1 g Options 0 cutated relative 1 cit_out1 cit_sD cit_out3 cit_out3 cit_out4 cit_out4 cit_out5 cit_out5 cit_out5 cit_out6 cit_out7 cit_out6 cit_out7 cit_out6 cit_out7 cit	Nutput Clocks MMC to the active input cloc Output Freq (MHz) Requested 100.000 32.000 3 100.000 100.000 100.000 100.000 100.000 100.000 100.000 100.000 100.000 100.000 100.000 100.000	M Settings Si Actual 100.00000 32.0000 NA NA NA NA NA NA NA NA NA NA	Phase (deg Phase (deg 0.000	rrees) Actual O 0.000 O 0.000 N/A	Duty Cycle (%) Requested 50.000 50.000 50.000 50.000 50.000 50.000 50.000 50.000 50.000 50.000 50.000 50.000 50.000 50.000	ОК Асtual 50.0 50.0 МА МА МА МА	Car Drives BUFG BUFG BUFG BUFG BUFG
ustomize IP ing Wizard (6.0) Imentation IP Location Imbol Resource Intow disabled ports Intowed isabled ports Intervention Into Intervention Interventin Intervention Intervention Intervention Intervention In	Component Nam Board Clockin The phase is ca Output Clock @ dk_out2 @ dk_out3 @ dk_out4 @ dk_out5 @ dk_out6 @ dk_out7 @ USE CLOC Utput Clo dk_out6 @ dk_out7	e clk_wiz_1 ag Options 0	Autput Clocks MMC to the active input clock 0000 20000 20 100.000 20 100.000 20 100.000 20 100.000 20 100.000 20 100.000 20 100.000 20 100.000 20 100.000 20	M Settings Si Actual 100.00000 32.0000 NA	ummary Phase (deg Requested 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000	Prees)	Duty Cycle (%) Requested 50.0000 50.0000 50.0000 50.0000 50.0000 50.0000 50.0000 50.0000 50.0000 50.0000 50.0000 50.0000 50.0000 50.0000 50.00000 50.00000 50.00000 50.00000000	ОК Асtual 50.0 50.0 МА МА МА МА	Drives BUFG BUFG BUFG BUFG BUFG BUFG
ustomize IP ing Wizard (6.0) Imentation IP Location Imbol Resource how disabled ports reset clk_out1 clk_out2 clk_out2 locked	Component Nam Board Clocki W chc.out2 Clocki	e dk_wiz_1 g Options 0 Port Name dk_out1 ck_out3 ck_out5 ck_out6 ck_out7 ck Sequencind 1 1 1 1 1 1 1 1 1 1 1 1 1	Autput Clocks MMC to the active input clock 0000 20000 20 100.000 20 100.000 100.000 100.000 100.000 100.000 100.000 100.000 100.000	M Settings Si Actual 100.00000 32.00000 NAA NAA NAA NAA NAA NAA NAA NAA NA	Phase (deg Requested 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.000000	rrees) Actual Control	Duty Cycle (%) Requested 50.0000 50.0000 50.0000 50.0000 50.0000 50.0000 50.0000 50.0000 50.0000 50.0000 50.0000 50.0000 50.0000 50.0000 50.0000 50.0000 50.0000 50.00000 50.0000 50.00000000	ОК Асtual 50.0 50.0 МА МА МА МА	Drives BUFG BUFG BUFG BUFG BUFG BUFG
ustomize IP ing Wizard (6.0) Imentation IP Location Imbol Resource how disabled ports reset clk_out1 clk_out2 locked	Component Nam Board Clocki The phase is ca Output Clock @ dk_out2 @ dk_out2 @ dk_out3 @ dk_out3 @ dk_out4 @ dk_out4 @ dk_out4 @ dk_out5 @ dk_out4 @ dk_out5 @ dk_out4 @ dk_out5 @ dk_out4 @ dk_out5 @ dk_out4 @ dk_out5	e cik_wiz_1 cik_outs	Autput Clocks MMC to the active input clock 0.000 32.000 3 100.000 100.000 100.000 100.000 100.000 100.000 100.000 100.000	M Settings St ACtual 100 00000 32 00000 NA NA NA NA NA NA NA NA NA NA	Phase (deg Requested 0.000	Image: state	Duty Cycle (%) Requested 50.0000 50.00000 50.00000 50.00000000	ОК Асtual 50.0 50.0 МА МА МА МА	Drives BUFG BUFG BUFG BUFG BUFG
ustomize IP ing Wizard (6.0) umentation IP Location mbol Resource how disabled ports reset cik_out1 cik_out2 cik_out2 cik_out2 cik_out2	Component Nam Board Clockii The phase is co Output Clock @ dk_out1 @ dk_out2 @ dk_out2 @ dk_out3 @ dk_out3 @ dk_out3 @ dk_out3 @ dk_out3 @ dk_out3 @ dk_out3 @ dk_out3 @ dk_out4 @ dk_out5 @ dk_out6 @ dk_out5 @ dk_out6 @ dk_out6	e ck_wiz_1 g Options 0 end	Nutrout Clocks MMC to the active input clock 0 0 0 0 32.000 0 1 100.000 1 0 100.000 1 0 100.000 1 0 100.000 1 0 000 1 0 000 1 0 00.000 1 0	M Settings Si A Settings Si C Actual 100.00000 32.00000 NIA NIA NIA NIA NIA NIA NIA NIA	ummary Phase (deg Requested 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000	rees) Actual Constraints Actual	Duty Cycle (%) Requested 50.0000 50.00000 50.000000 50.00000000	ОК Асtual 50.0 МА МА МА МА МА	Drives BUFG BUFG BUFG BUFG BUFG BUFG
ustomize IP ing Wizard (6.0) umentation IP Location mbol Resource how disabled ports reset cik_out1 cik_out2 cik_out2 cik_out2 cik_out2	Component Nam Board Clockii The phase is ca Output Clock @ dk_out1 @ dk_out2 @ dk_out2 @ dk_out3 @ dk_out4 @ dk_out4 @ dk_out4 @ dk_out3 @ dk_out3 @ dk_out3 @ dk_out3 @ dk_out3 @ dk_out3 @ dk_out4 @ dk_out4 @ dk_out4 @ dk_out4 @ dk_out5 @ dk_out5 @ dk_out5 @ dk_out5 @ dk_out6 @ dk_out7	e ck_wiz_1 g Options 0 options 0 options 0 options ck_out	Nutput Clocks MMC to the active input clock 32.000 100.000 100.000 100.000 100.000 100.000 100.000 100.000 100.000 100.000 100.000 100.000 100.000	M Settings Si Actual 100.00000 32.0000 NIA NIA NIA NIA NIA NIA NIA Cocking Feedbac Source @ Auto Q Auto Q Auto Q Settings Si Source @ J Settings Si Si Si Si Si Si Si Si Si Si		rees) Actual O O O O O O O O O O O O O O O O O O O	Duty Cycle (%) Requested 50.0000 50.00000 50.000000 50.00000000	ОК Асtual 50.0 10/4 10/4 10/4	Drives BUFG BUFG BUFG BUFG BUFG BUFG
ustomize IP sing Wizard (6.0) umentation IP Location mbol Resource now disabled ports	Component Nam Board Clockin The phase is ca Output Clock @ dk_out1 @ dk_out2 @ dk_out3 @ dk_out4 @ dk_out5 @ dk_out4 @ dk_out5 @ dk_out4 @ dk_out5 @ dk_out7 @ USE CLOC	e ck_wiz_1 g Options 0 options 0 options 0 options ck_source options 0 optio	Nutput Clocks MMC to the active input clock MMC to the active input clock 32.000 32.000 32.000 100.000 100.000 <td>M Settings Si C Actual 100.0000 32.0000 NA NA</td> <td></td> <td>rees) Actual Control C</td> <td>Duty Cycle (%) Requested 50.0000 50.0000 50.000 50.000 50.000 50.000 50.000 50.000 50.0000 50.0000 50.0000 50.0000 50.0000 50.0000 50.0000 50.0000 50.0000 50.0000 50.0000 50.0000 50.0000 50.0000 50.0000 50.0000 50.0000 50.0000 50.00000 50.00000 50.00000 50.00000000</td> <td>ОК Асtual 50.0 50.0 МА МА МА МА МА</td> <td>Car Drives BUFG BUFG BUFG BUFG BUFG</td>	M Settings Si C Actual 100.0000 32.0000 NA		rees) Actual Control C	Duty Cycle (%) Requested 50.0000 50.0000 50.000 50.000 50.000 50.000 50.000 50.000 50.0000 50.0000 50.0000 50.0000 50.0000 50.0000 50.0000 50.0000 50.0000 50.0000 50.0000 50.0000 50.0000 50.0000 50.0000 50.0000 50.0000 50.0000 50.00000 50.00000 50.00000 50.00000000	ОК Асtual 50.0 50.0 МА МА МА МА МА	Car Drives BUFG BUFG BUFG BUFG BUFG
ustomize IP ing Wizard (6.0) amentation IP Location mbol Resource now disabled ports reset clk_out1 clk_out2 clk_in1 locked	Component Nam Board Clockii The phase is ca Output Clock @ dk_out3 @ dk_out3 @ dk_out3 @ dk_out3 @ dk_out3 @ dk_out3 @ dk_out3 @ dk_out3 @ dk_out4 @ dk_out5 @ dk_out4 @ dk_out5 @ dk_out7 @ USE CLOC Dutput Clo dk_out4 @ dk_out5 @ dk_out7 @ USE CLOC Dutput Clo dk_out4 @ dk_out5 @ dk_out7 @ USE CLOC Dutput Clo dk_out5 @ dk_out7 @ USE CLOC Dutput Clo dk_out6 @ dk_out7 @ dk_	e ck_wiz_1 g Options 0 Port Name ck_out1 ck_out3 ck_out3 ck_out3 ck_out3 ck_out3 ck_out3 ck_out3 ck_out4 ck_out3 ck_out4 ck_out3 ck_out4 ck_out4 ck_out4 ck_out4 ck_out5 ck sequend 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	Nutput Clocks MMC to the active input clock MMC to the active input clock 32.000 32.000 3 100.000	M Settings Si c ALLUAI 100.00000 32.0000 NIA NIA NIA NIA NIA NIA NIA NIA NIA NIA		rees) Actual O O O O O O O O O O O O O O O O O O O	Duty Cycle (%) Requested 50.000 50.000 50.000 50.000 50.000 50.000 50.000 50.000 90 90 90 90	OK Actual 50.0 N/A N/A N/A N/A N/A N/A N/A	Car Drives BUFG BUFG BUFG BUFG BUFG BUFG
ustomize IP Ing Wizard (6.0) Imentation IP Location Imbol Resource Index R	Component Nam Board Clockii The phase is ca Output Clock V dk_out1 V dk_out2 dk_out3 dk_out3 dk_out3 dk_out4 dk_out5 dk_out7 USE CLOC USE CLOC USE CLOC USE CLOC dk_out4 dk_out5 dk_out5 dk_out7 USE CLOC V dk_out7 USE CLOC V dk_out7 dk_out8 dk dk_out8 dk d	e ck_wiz_1 g Options 0 cutated relative f Port Name ck_out1 ck_out3 ck_out3 ck_out3 ck_out3 ck_out4 ck_out3 ck_out3 ck_out4 ck_out3 ck_out4 ck_out3 ck_out4 ck_out3 ck_out4 ck_out3 ck_out4 ck_out5 ck Sequenc 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	Nutput Clocks MMC to the active input cloc! 0 0 ubgut Frequested 100.000 100.000 2 100.000 1<	М Settings Su C Autual 100.00000 32.00000 NIA NIA NIA NIA NIA NIA NIA NIA		rees) Actual Control NA N	Duty Cycle (%) Requested 50.000 50.000 50.000 50.000 50.000 50.000 50.000 200 200 200 200 200 200 200 200 20	OK Actual 50.0 N/A N/A N/A N/A N/A	Car Drives BUFG BUFG BUFG BUFG BUFG

Step 5:

Run "Connection Automation" and select all the pins.



Go to the Board section, Drag and drop the USB UART from the Board section to the design.



Click on "Run Connection Automation" select all the pins and click ok.

All Automation (1 out of 1 selected)	Description Connect Slave interface (/axi_uartlite	_0/S_AXI) to a selected Master addr	ess space.
S_AXI	Options		
	Master interface	/microblaze_0 (Periph) 🗸	
	Bridge IP	/microblaze_0_axi_periph 🗸	
	Clock source for driving Bridge IP	/clk_wiz_1/clk_out1 (100 MHz)	~
	Clock source for Slave interface	Auto	¥
	Clock source for Master interface	/clk_wiz_1/clk_out1 (100 MHz)	~

Connect interrupt output lines from "**AXI Uartlite**" to the "**Concat**" block as shown in the below figure.



Step 7:

Add the SD card IP repository to Vivado IP catalog from <u>here</u>. Open IP Catalog under PROJECT MANAGER, right click on Vivado Repository -> Add Repository.

			Flow Navigato	or 😤	≑ ? _			
			✓ PROJECT	MANAGER				
			🂠 Setting	ļs				
			Add So	ources				
			Langua	age Templates				
			👎 IP Cata	alog				
	Diagram ×	Address Editor	× Address Map	× IP Catalog	×			
	Cores Inte	erfaces						
	Q	\$ ₩ •€ /	r 2					
	Search: Q-							
	Name		^ t	AXI4	Status	License	VLNV	
	> 📄 Vivado F	Repository	Properties	Ctrl+F				
			Add Repository					
			Refresh All Reposito	ries				
			Export to Spreadshee	et				
•								
Provid	le the Dir	ectory path	n of the IP an	d click Sele	ct.			
And A	dd the SI	Ocard IP to	the Block De	esign				
		Name			^1 AXI4			



Step 8:

After adding the **SD card** Ip to the Block design, click on **Run Block Automation.**

Now, connect the clock_32 pin of the SD card IP to the clk_SD pin of the Clocking wizard as shown in the below figure.



Step 9: Right click on Sdcard ip and click on Make External.



Step 10: In Sources tab of Vivado, Right-Click on '**Constraints**' and click '**Add Sources**'.



Step 11: Once the "**Add Sources**" tab opens select "**Add or create constraints**" and click on "**Next** ".

	Add Sources
Vivado	This guides you through the process of adding and creating sources for your project
ML Edition	Add or create constraints
	∆dd or create design sources
	Add or create simulation sources
•	< Back Finish Cancel
on ' Create F	ile ' and give ' SD_test ' as File name. Click ' OK ' and ' Finis l
on ' Create F Add Sources	ile ' and give ' SD_test ' as File name. Click ' OK ' and ' Finis l
on ' Create F Add Sources dd or Create Constra	ile' and give ' SD_test' as File name. Click ' OK ' and ' Finis l
on ' Create F Add Sources add or Create Constra pecify or create constraint	ile' and give ' SD_test ' as File name. Click ' OK ' and ' Finis l aints liles for physical and timing constraint to add to your project.
on ' Create F Add Sources dd or Create Constra pecify or create constraint f	ile' and give 'SD_test' as File name. Click 'OK' and 'Finisl nints Nes for physical and timing constraint to add to your project.
On ' Create F Add Sources add or Create Constra pecify or create constraint f	ile' and give 'SD_test' as File name. Click 'OK' and 'Finisl aints illes for physical and timing constraint to add to your project.
Add Sources Add Sources add or Create Constraint Specify constraint set: \downarrow	ile' and give 'SD_test' as File name. Click 'OK' and 'Finisl aints Tiles for physical and timing constraint to add to your project. Constrs_1 (active)
On ' Create F Add Sources add or Create Constra becify or create constraint f Specify constraint set:	ile' and give 'SD_test' as File name. Click 'OK' and 'Finisl aints liles for physical and timing constraint to add to your project. Create a new constraints file your project
Add Sources Add or Create Constraint Specify constraint set: + = + +	ile' and give 'SD_test' as File name. Click 'OK' and 'Finisl aints files for physical and timing constraint to add to your project. Create Constraints File Create Constraints File and add it to your project
On ' Create F Add Sources add or Create Constra pecify or create constraint of <u>Specify constraint set</u>	ile' and give 'SD_test' as File name. Click 'OK' and 'Finisk aints liles for physical and timing constraint to add to your project. Create a new constraints file Create a new constraints file and add it to your project Eile type: XDC ~
On 'Create F Add Sources add or Create Constraint abecify or create constraint f Specify constraint set:	<pre>ile' and give 'SD_test' as File name. Click 'OK' and 'Finisl nints lites for physical and timing constraint to add to your project. constrs_1 (active) Create a new constraints file Create a new constraints file and add it to your project Eile type: NDC File name: SD_test</pre>
on ' Create F Add Sources dd or Create Constra becify or create constraint f Specify constraint set:	aints Hiles for physical and timing constraint to add to your project. Create a new constraints file and add it to your project File name: SD_test File location: CLOCAL to Project>
Add Sources Add Sources add or Create Constraint Specify constraint set: + = + +	aints Hes for physical and timing constraint to add to your project. Create a new constraints file and add it to your project File name: SD_test Cancel
Add Sources	ile' and give 'SD_test' as File name. Click 'OK' and 'Finisl aints ites for physical and timing constraint to add to your project. Create a new constraints file and add it to your project File type: Distance File type: Create I o Project> File location: Concel
Add Sources	ile' and give 'SD_test' as File name. Click 'OK' and 'Finish anns iles for physical and timing constraint to add to your project. Constrs_1 (active) Create a new constraints File Create a new constraints file and add it to your project File name: SD_test File location: < Local to Project> Cancel Add Files Create File
Add Sources Add or Create Constraint Specify constraint set: + - + - • •	<pre>ile' and give 'SD_test' as File name. Click 'OK' and 'Finish ints iles for physical and timing constraint to add to your project. Create a new constraints file and add it to</pre>

Step 12:

Copy the following constraints in your constrains file and save it.

set_property -dict {PACKAGE_PIN F14 IOSTANDARD LVCMOS33} [get_ports sdcard_if_1_0_sd_clk]

set_property -dict {PACKAGE_PIN F15 IOSTANDARD LVCMOS33} [get_ports sdcard_if_1_0_sd_cs]

set_property -dict {PACKAGE_PIN B17 IOSTANDARD LVCMOS33} [get_ports sdcard_if_1_0_sd_miso]

set_property -dict {PACKAGE_PIN A17 IOSTANDARD LVCMOS33} [get_ports sdcard_if_1_0_sd_mosi]

Step 13:

Right-click "**SD_card**" in the "**Sources**" window, and select "**Create HDL Wrapper**" from the popup menu. Click "**OK**" on the window that appears to finish generating a wrapper.

	4	
Sources × Design Signals		? _ 🗆 🖸
Q 素 ♣ ╋ ? ● 0		٥
✓		
✓ ● ♣ SD_card_wrapper (SD_ca	rd_wra	apper.v) (1)
✓ ▲ ■ SD_card_i : SD_card (\$	SD ca	rd.bd) (1)
> OSD_card (SD_card.v)		Source Node Properties
✓ ☐ Constraints (1)		Open File
✓		Open With
D SD.xdc		One sta LIDI. Wassansa
> 🗁 Simulation Sources (1)		Create HDL Wrapper
> 🚍 Utility Sources		View Instantiation Template
		Generate Output Products
		Reset Output Products

Step 14:

Click "Generate Bitstream" under the "Program and Debug" section to synthesize, implement and generate a bitstream.

✓ SYNTHESIS
Run Synthesis
> Open Synthesized Design
Run Implementation
> Open Implemented Design
✓ PROGRAM AND DEBUG
Generate Bitstream
> Open Hardware Manager

Step 15: After generating the bitstream successfully, select **Export -> Export Hardware** from the **File menu**. Click **Next**.

Edit Flow Tools Repgts Project + + -	Window Layout View Image: Sources Sources Cock DESIGN - SD_card * Sources × Design Q Q X + 1 V Design Sources (1) > > = * SD_card_wrag Constraints (1) Sources :
Export	Export Hardware
Print Ctrl+P Egit	Export Block Design Export Bitstream File Export Simulation

Select the "include bitstream" checkbox and click Next.

ort Hardware Platform	×
out e platform properties to inform downstream tools of the intended use of the target platform's hardware design.	٦
Pre-synthesis This platform includes a hardware specification for downstream software tools.	
Include bitstream This platform includes the complete hardware implementation and bitstream, in addition to the hardware specification for software tools.	
< <u>Back</u> <u>Next></u> Einish Ca	ancel
	ort Hardware Platform Include a hardware specification for downstream software tools. Pre-synthesis This platform includes the complete hardware implementation and bitstream, in addition to the hardware specification for software tools. (< Back Metric Finish Ca

Provide the **XSA file name** and save it at a suitable **location**. Click **Next** and click **Finish** in the next dialog box.

Step 16:

Select Launch Vitis IDE from the Tools menu.

	<u>T</u> ools	Rep <u>o</u> rts <u>W</u> indow La <u>v</u> out	<u>V</u> iew <u>H</u> e		
		Validate Design	F6		
		Create and Package New IP			
		Create Interface Definition			
		Run Tcl Script			
		Property Editor	Ctrl+J		
		Associate EL <u>F</u> Files			
		Generate Memory Configuration File			
		Compile Simulation Libraries			
		Vivado Store			
		C <u>u</u> stom Commands	×		
		Launch Vitis IDE			
	Q	Language <u>T</u> emplates			
	۰	Settings			
Step 17:					

After Vitis Unified IDE window opens, click on "**Open Workspace**" and select necessary folder to keep the Vitis files.

Welcome to the Vitis Unified IDE
Get Started
₽ <u>Open Workspace</u>
â <u>Examples</u>
<i>Aigrate Classic IDE Workspace</i>

Step 18:

Create a new platform for the project, by selecting "**Create Platform Component**", click "**Next**", in the Flow tab select the XSA file saved using the step 20 and finally click "**Next**" and "**Finish**" respectively.



After successful creation of the platform, build the platform.

✓ FLOW		♦	*
Component	🔹 SD_test	~	땷
🖓 Build			

Step 19:

Next create the Helloworld Application component by selecting the "Helloworld" template from the "examples",



In "Create Application Component" tab specify project name and location, click "Next"

Create Application Compone Name and Location > I	nt - Hello World Hardware > Domain > Sysroot > Summary		×
Name and Location			
Choose a name for your co	sp_card_test		
Component location	C:\projects\Elbert_57\SD_card\vitis	Browse	
Component will be created	at C\projects\Elbert_S7\SD_card\vitis\SD_card_test		Next

Select newly created Platform and click "Next".

C	Create Application Component - Hello World									
Select Platform										
Platforms supporting the selected example from your repositories. To create a new platform, use "File -> New Component -> Platform"										
\approx \Rightarrow + - υ ρ										
	NAME	BOARD	FLOW	VENDOR	ратн					
	 C:\projects\Elbert_S7\SD_card\vitis\SD_test\export\SD_te 				ojects\Elbert_S7\SD_card\vitis\SD_test\export\SD_test					
	(1)									
	SD_test	elbert	Embedded	xilinx.com	nS7\SD_card\vitis\SD_test\export\SD_test\SD_test.xpfm					
	Cancel				Back	xt				

Select the domain as "**Standalone_microblaze_o**" and click "**Next**" and click on "**Finish**"

Step 20: Download the SD_test.c file from <u>here</u>, Copy the content of SD_test.c file to the helloworld.c file to test SD card .

After adding the source file build the Project.

Step 21:

Once the build is completed successfully, power up Elbert S7 FPGA Development Board using an USB type C cable. and insert the SD card into the micro SD card slot of Elbert S7 FPGA Development Board.

Step 22:

Program the FPGA on Elbert S7 with a simple boot loop program by selecting the **Program Device** option from the **Vitis menu**.



Once the "Program Device" window opens click on "Program ".

Program Device				×					
Specify the bitstream and the ELF files that reside in BRAM memory.									
Project _	5D_card_test ~								
Connection	.ocal v	<u>New</u>							
Bitstream/PDI	<pre>closed_closed closed_clos</pre>	<u>Browse</u>	<u>Search</u>						
C	Partial Bitstream								
BMM/MMI File	projects\Elbert_S7\SD_card\vitis\SD_card_test_ide\bitstream\SD_card_wrapper.mmi	<u>Browse</u>	<u>Search</u>						
Software Configuration									
PROCESSOR	ELF/MEM FILE TO INITIALIZE IN BLOCK RAM								
microblaze_0	bootloop			~					
Skip Revision Check									
Cancel		Gener	ate	Program					

Step 23:

Meanwhile, open any serial terminal program (such as PuTTY, Teraterm etc) and open the port corresponding to Elbert S7 with a 9600 baud rate (the default baud rate given in UART IP). Program the board by selecting the "Run".



Step 24:

If everything went well, Serial terminal would show the execution is Successful.



REFERENCES

> **PRODUCT LINKs**:

- o <u>Product Page</u>.
- o <u>User manual</u>.
- o <u>Schematics</u>.
- o <u>Xdc Constraints file</u>.

> Tools Link:

- o <u>Vivado Design suite</u>.
- <u>PUTTY</u>.

