Zynq SDR Intro

# Lab 1 PL Blinking LED Block Design

## Abstract

This lab illustrates the use of block design by blinking LED with counter using no HDL code

This lab should take approximately 45 minutes.

## Objectives

After completing this lab, you will be able to:

* Understand Vivado Design Flow
* Understand Vivado Block Design

## Introduction

This lab supports “Vivaldo Design Flow”, “Vivado Project Based” and “Vivado Block Design”.

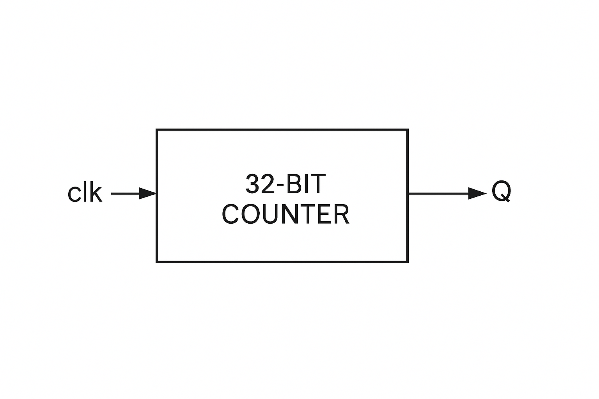


Figure 1‑1: Launching the Vivado Design Suite from the Start Menu

## General Flow

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Step 1:  Creating  a New  Project |  | Step 2:  Creating  a New  Source  [Block Design] |  | Step 3:  Synthesizing  the Design |  | Step 4:  Examining the Design |  | Step 5:  Simulating the Design  [Hardware] |

### Creating a New Project Step

In this step you will create a new project using the New Project Wizard in the Vivado® Design Suite.

The New Project Wizard in the Vivado IDE will create an XPR project file. The Vivado IDE project file (.xpr) organizes your design files and saves the design status whenever the processes are run from design entry through implementation to programming the targeted Xilinx device.

There are a number of ways to launch the Vivado Design Suite. The two most popular mechanisms are shown here.

1-1. Launch the Vivado Design Suite.

This can be done in two standard ways, use your preferred method.

1-1-1. Select Start > All Programs > Xilinx Design Tools > Vivado 2023.2

A screenshot of a computer

AI-generated content may be incorrect.

Figure 1‑2: Launching the Vivado Design Suite from the Start Menu

-- OR --

Double-click the Vivado Design Suite shortcut icon (A logo with a black background

AI-generated content may be incorrect.) on the desktop.

The Vivado Design Suite opens to the Welcome window. From the Welcome window you can create a new project, open an existing project, or enter Tcl commands directly into the Vivado Design Suite as well as access documentation and examples.

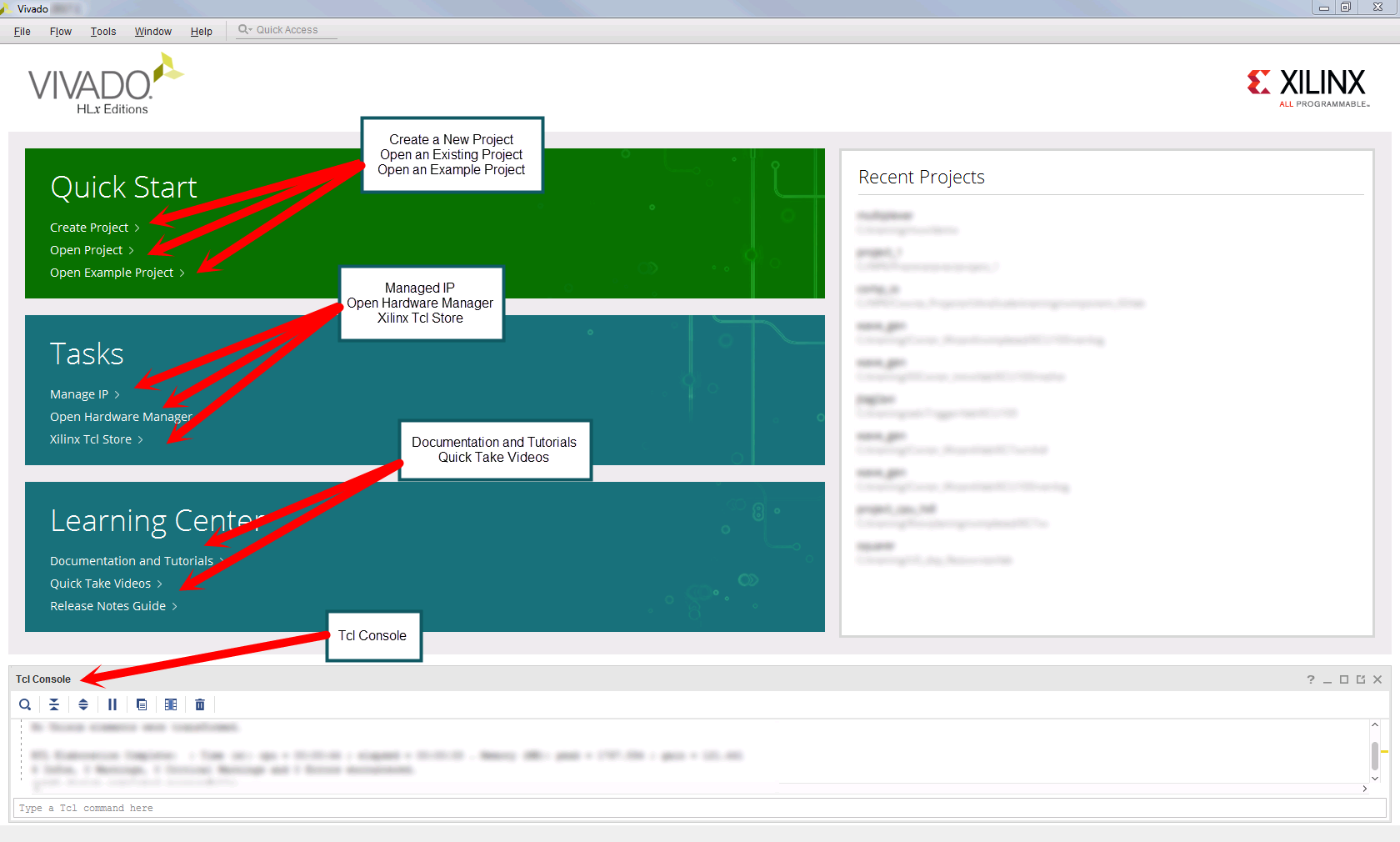


Figure 1‑3: Vivado Design Suite Welcome Screen

"Create Project" is the starting point for all designs. Projects contains sources, settings, graphics, IP, and other elements that are used to build a final bitstream and analyze a design. The Create New Project Wizard in the Vivado Design Suite allows you to specify HDL and other project resource files that will be included in the project.

1-2. Create a new blank Vivado Design Suite project.

1-2-1. Click Create Project to begin the process (1).

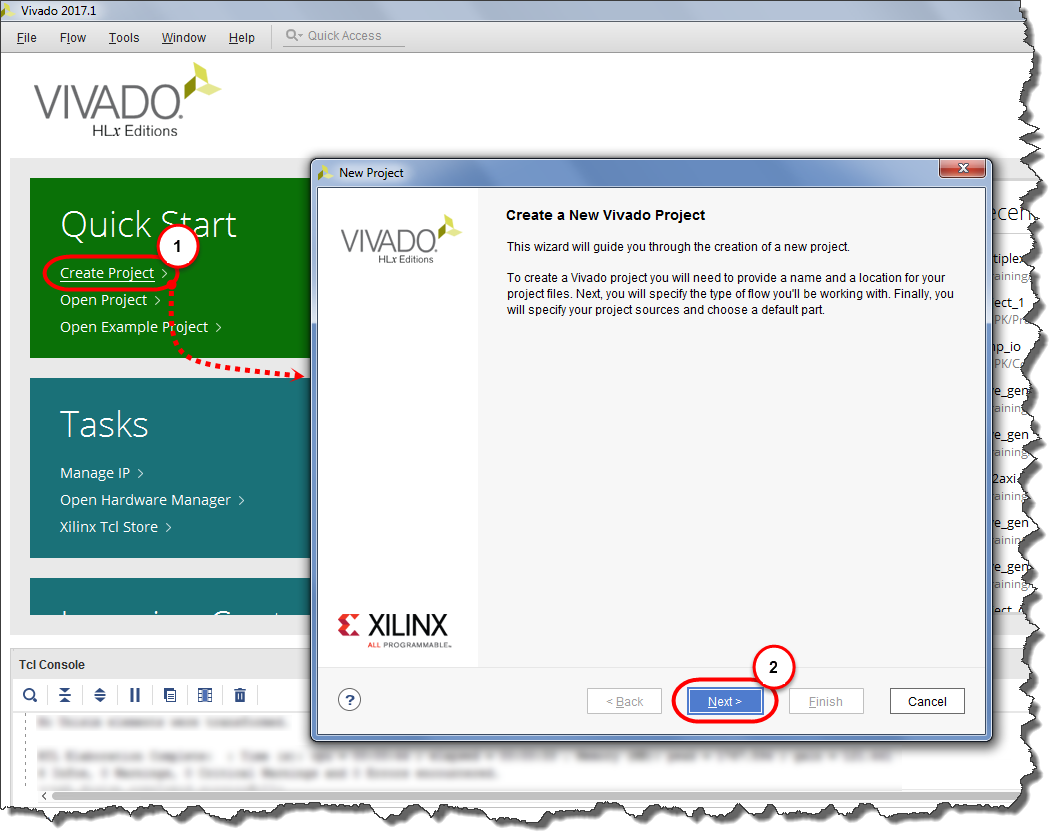


Figure 1‑4: Creating a New Vivado Design Suite Project

This will launch the New Project Wizard.

1-2-2. Click Next to exit the introductory dialog box and begin entering in project-specific information (2).

1-3. You will now encounter a series of dialog boxes asking you to enter different pieces of information describing the project.

1-3-1. Enter lab1\_blockdesign in the Project name field.

1-3-2. Enter C:\training\dsgnZynqSDR\labs\ in the Project location field.

Alternatively, you can use the browse feature to navigate to where you want the project to reside.

1-3-3. Deselect the Create Project Subdirectory option as leaving this checked will create an unnecessary level of hierarchy for this lab.

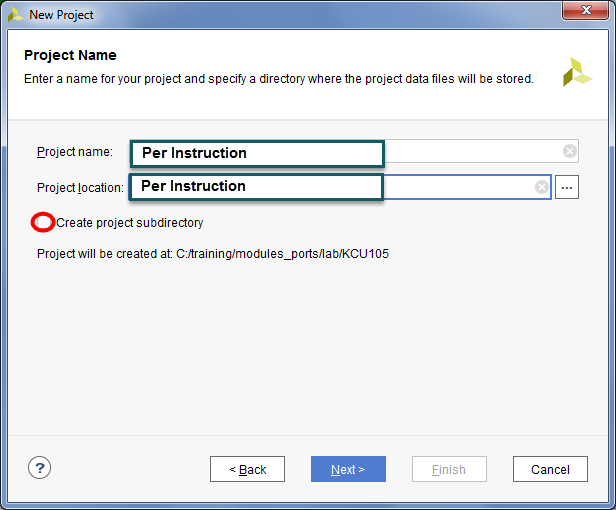


Figure 1‑5: Entering the Project Name and Location

1-3-4. Click Next to advance to the next dialog box.

Here you will choose between an RTL project or a post-synthesis project. Simply put, an RTL project enables you to add or create new HDL files and synthesize them, whereas the post-synthesis project requires pre-synthesized files. When an empty design is created, an RTL project is used.

1-3-5. Select RTL Project (1).

1-3-6. Select Do not specify sources at this time (2), which creates a blank project.

While existing sources could be entered at this time, you will enter them later so that you can move through this portion of the project creation process more quickly.

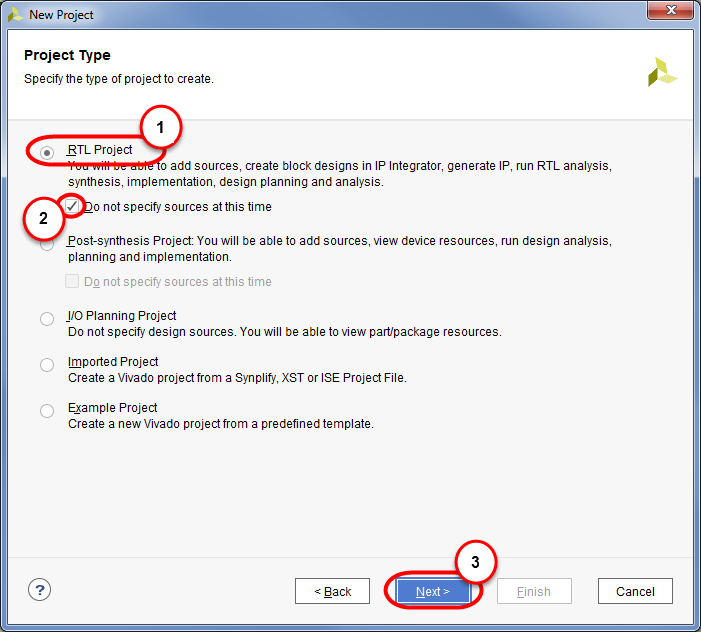


Figure 1‑6: Selecting Project Type

1-3-7. Click Next to advance to the target device/platform selection (3).

1-4. Select the target part by first filtering by board and then by family. If you are not using a supported board, you will need to filter by part.

1-4-1. Select Parts from the Select area to filter by board rather than by the specific part (1).

1-4-2. Select Search and type “xc7z020clg”.

This limits the number of boards seen to those manufactured by the specified vendor.

1-4-3. Select xc7z020clg484-1 from the list.

Alternatively you can select the board directly from the list at any time while in this dialog box.

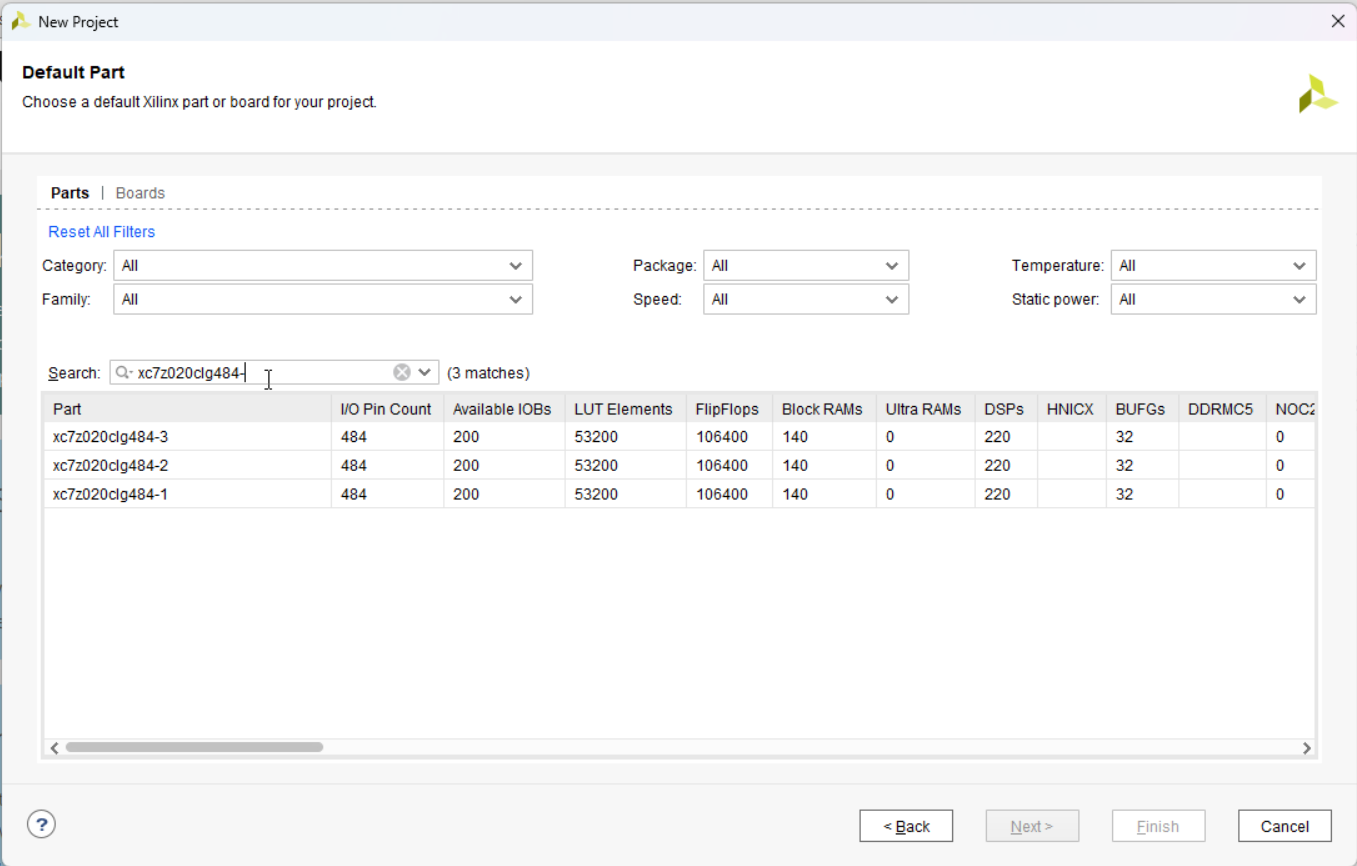


Figure 1‑7: Selecting the Board for the Project

1-4-4. Click Next to advance to the summary (3).

A summary of your project is displayed. If you want to change any of the information that you entered, you can do so now by clicking Back until you reach the correct dialog box and making the correction, or you can create the project now and edit the project properties, add or remove files, etc. later.

1-4-5. Click Finish to accept these settings and build the project.

Your project is constructed and leaves you in the operational portion of the Vivado Design Suite GUI.

1-4-6. Click Settings under Project Manager in the Flow Navigator and change the Target Language to VHDL.

### Creating a New Block Design Step

Now that the project has been created, it is time to begin creating the design. You will only block design without any code to learn concept of Vivado design flow.

2-1. Create a Vivado block design.

2-1-1. Select Create Block Design in the Flow Navigator under Project Manager.

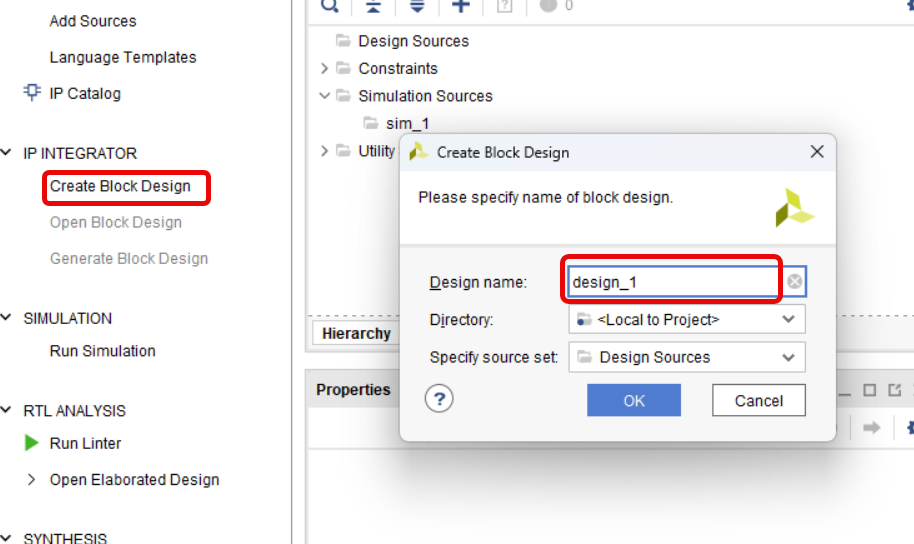


Figure 1‑8: Selecting Create Block Design

2-1-2. Select Add new IP in the block design.

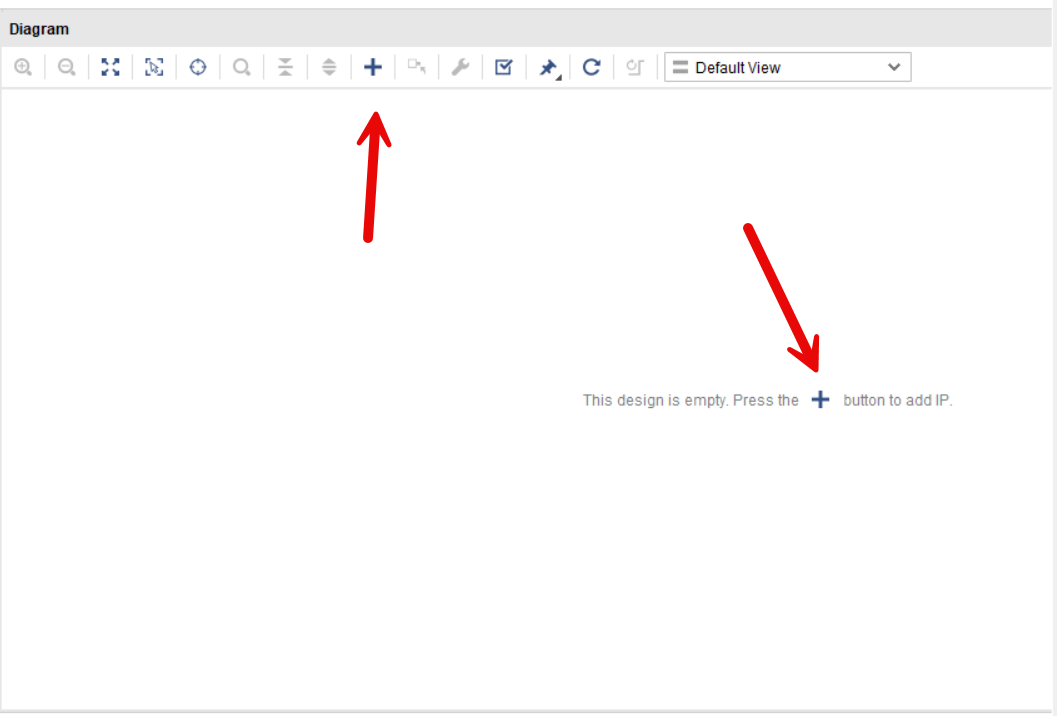


Figure 1‑9: Selecting Add new IP

2-1-3. The IP list will appear on screen.

2-1-4. Search for “Counter” and click Binary Counter.

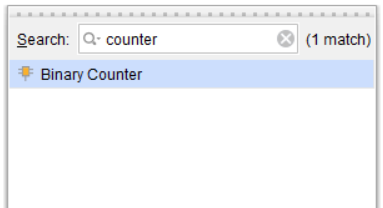


Figure 1‑10: Selecting IP

The Create Source File dialog box opens.

2-1-5. A counter will appear on block design. Now double click on counter to customize the IP.

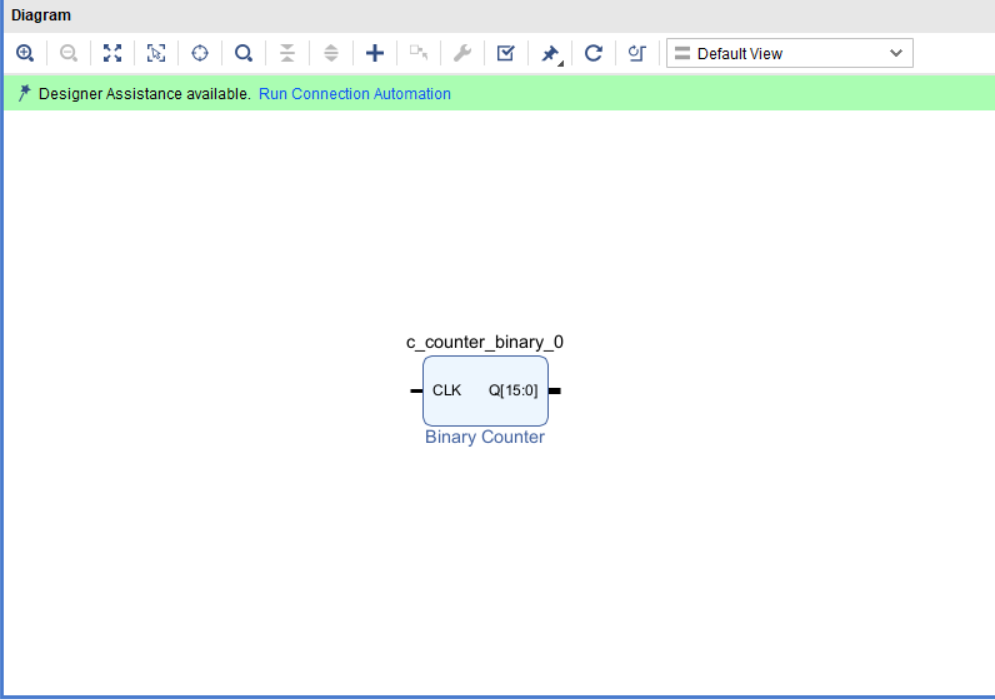


Figure 1‑: Block Design with Counter

2-1-6. Change Output Width to 32 to create 32 bits counter.



Figure 1‑: Block Design with Counter

2-1-7. Counter block has to connect with outside world. Input of the block will require IBUF and output will require OBUF as interfaces units. Add new IP to block design.

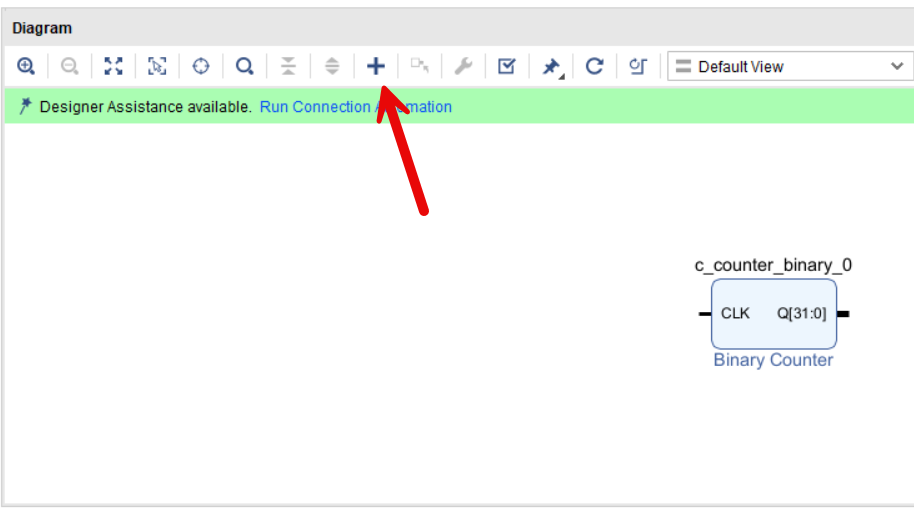


Figure 1‑: Block Design with Counter

2-1-6. Search for buffer in the search box, then select “Utility Buffer”

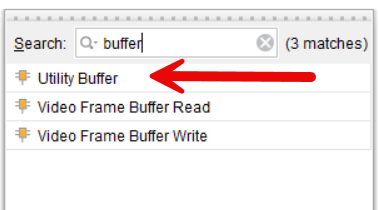


Figure 1‑14: Add Utility Buffer

2-1-9. Double click on the Buffer and edit “C size” to 1 and select BUFG

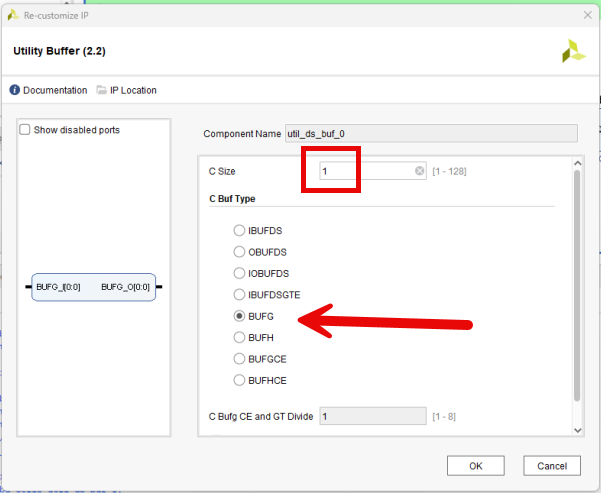


Figure 1‑15: Select BUFG and edit C Size

2-1-10. Connect Output of BUFG to input of counter

2-1-11. Add IP “Slice” to block design edit properties Din Width to 32, Din to 31 and Dto to 24. Connect Slice input to counter output.

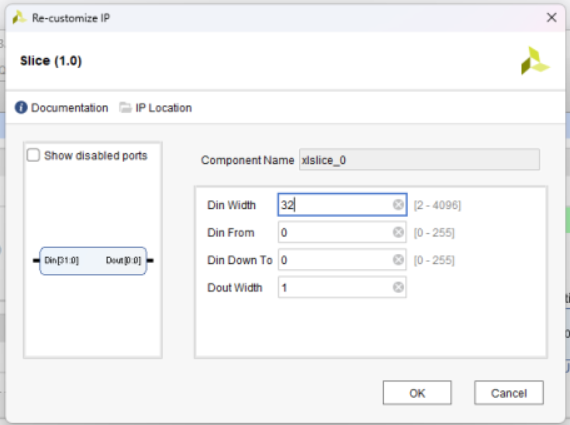


Figure 1‑16: Edit Din Width to 32, “Din From” to 31 and “Din To” to 24

2-1-12. Add more BUFG with C size=8 and connect BUFG input to slice output

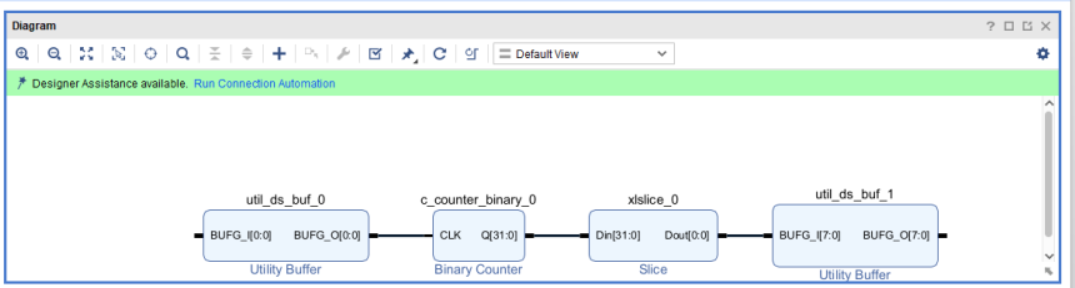


Figure 1‑17: Block Design with counter and BUFG

2-1-12. Select input of first BUFG and click on right mouse and select “Make External”

2-1-13. Select “Make External” on the output of last BUFG.

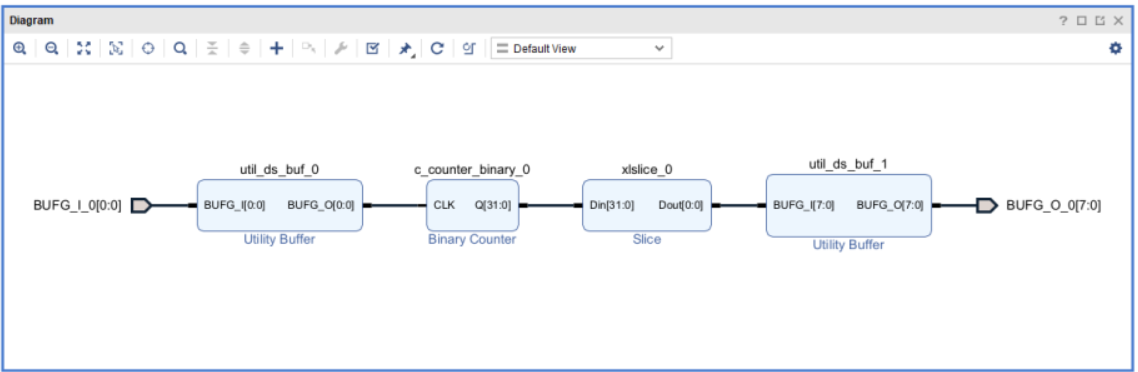


Figure 1‑18: Block Design with counter with BUFG

2-1-14. Click on save and “generate block design”, at the text pop up block, keep everything as default and select “Generate”

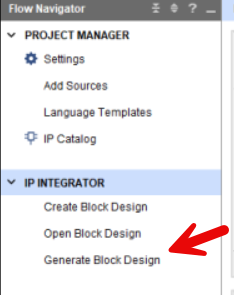


Figure 1‑19: Generate block design

2-1-15. Click on save and “generate block design”, at the text pop up block, keep everything as default and select “Generate”



Figure 1‑20: New VHDL file is generated from our block design

2-1-16. Select “design\_1 (design\_1.bd)” click on right mouse and select “Create HDL Wrapper”, at the text pop up box “select “Let Vivado manage…” and click OK”

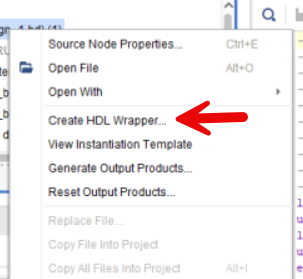


Figure 1‑21: Create HDL Wrapper from the menu in Flow manager

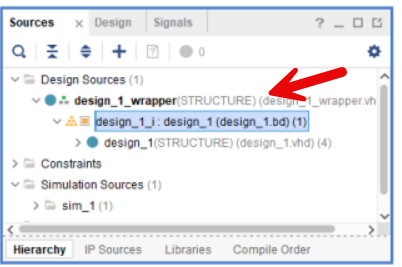


Figure 1‑22: New VHDL file is generated as the wrapper file for outside world

### Synthesizing theDesign Step 3

3-1. Synthesizing the Design

3-1-1. RTL Analysis Select Run Linter in RTL Analysis to verify HDL code

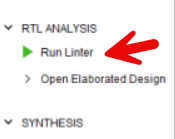


Figure 1‑23: Select Run Linter

3-1-3. Select Open Elaborate Design, the schematic diagram based on the design will be shown up

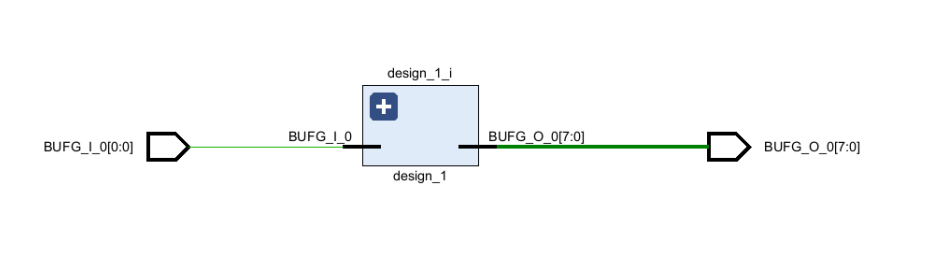


Figure 1‑24: Schematic based on RTL

3-1-4. Select Run Synthesis and click OK on the pop up text box. After finish select “Open Systhesis Design” and OK. Then, Vivado will show IC package view, and I/O ports.

### Examining the Design Step 4

4-1. Examining the Design

4-1-1. Your design may successfully be synthesized. Then you will have chance to view your design on actual FPGA. After screen layout after finish synthesis may look different. Please select drop down menu on top right of screen for a layout mode.

A computer screen shot of a black screen

AI-generated content may be incorrect.

Figure 1‑25: Vivado

4-1-2. IO Planning shows electrical connection of FPGA IC which is a BGA 484, so number of BGA balls connection is 484.

A screenshot of a game

AI-generated content may be incorrect.

Figure 1‑26: IO Planning Layout

Top left ball is A1 the last ball on bottom right is AB22.

4-1-3. Floor Planning shows resources used by our project.

A screenshot of a computer

AI-generated content may be incorrect.

Figure 1‑27: Floor Planning Layout

4-1-4. Open “zed\_sch\_rev\_f1-public.pdf” on page 9, look for GCLK. We will see the GCLK is generated from IC17 with 3.3V oscillator and connect to BGA #Y9. Look for LD0 to LD7 the pin connected to LEDs, which connect to BGA T22, T21, U22, U21, V22, W22, U19, U14. Then click IO ports on the

A screenshot of a computer

AI-generated content may be incorrect.

Figure 1‑28: IO Ports Window

4-1-5. Select I/O Ports and edit Package Pin corresponding to schematic information and I/O Standard to LVCMOS33 (3.3V). Then save project. Vivado will create the Constrain file, put in lab1blockdesign as a name.

A screenshot of a computer

AI-generated content may be incorrect.

Figure 1‑29: I/O Ports

4-1-6. Run Synthesis again after update constrain file.

A screenshot of a computer

AI-generated content may be incorrect.

Figure 1‑30: Run Synthesis

4-1-7. Run Implementation and Generate Bitstream. Then Open hardware manager.

A screenshot of a computer program

AI-generated content may be incorrect.

Figure 1‑31: Run Implementationm, Generate Bitstream, and Open Hardware Manager

4-1-8. Power on FPGA board. Click on Open Target, then Auto Connnect and Program Device.

A screenshot of a computer

AI-generated content may be incorrect.

Figure 1‑32: Open Target and Program Device

### Simulating the Design Step 5

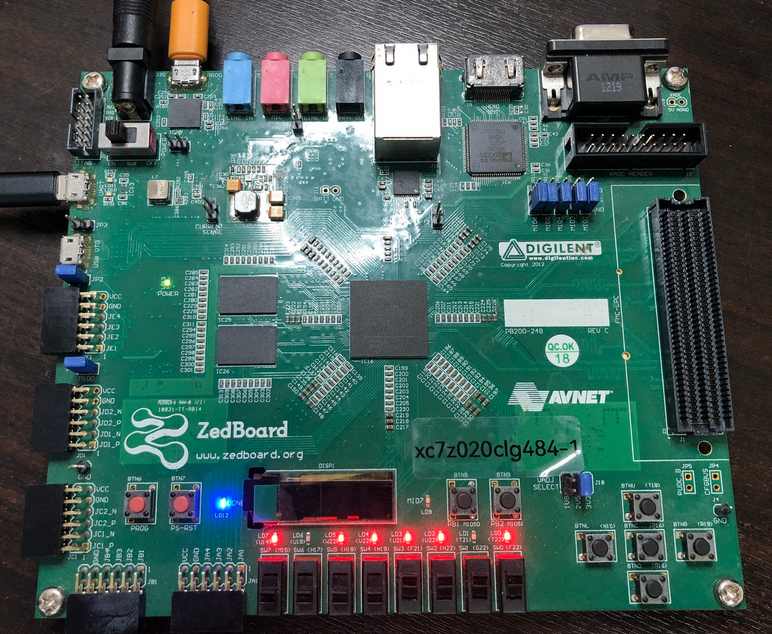


Figure 1‑33: Zedboard running lab1

## Summary

This lab shows how the flow of Vivado design, when to add outside world connection as constrain file. However there are still more thing, as HDL coding and timeing delay which are all very important for commercial and industrial solution.

# Lab 2 PS HelloWorld Blinking LED

## Abstract

This lab illustrates the use of PS PL interface by creating HelloWorld demo with blinking LED with software counter

This lab should take approximately 45 minutes.

## Objectives

After completing this lab, you will be able to:

* Understand Vivado Design and Vitis Interface

## Introduction

This lab supports “Vivado Design and Vitis Interface”

## General Flow

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Step 1:  Creating  a New  Project |  | Step 2:  Creating  a New  Source  [Block Design] |  | Step 3:  Synthesizing  the Design |  | Step 4:  Examining the Design |  | Step 5:  Simulating the Design  [Hardware] |

### Creating a New Project Step

In this step you will create a new project using the New Project Wizard in the Vivado® Design Suite.

The New Project Wizard in the Vivado IDE will create an XPR project file. The Vivado IDE project file (.xpr) organizes your design files and saves the design status whenever the processes are run from design entry through implementation to programming the targeted Xilinx device.

There are a number of ways to launch the Vivado Design Suite. The two most popular mechanisms are shown here.

1-1. Launch the Vivado Design Suite.

This can be done in two standard ways, use your preferred method.

1-1-1. Select Start > All Programs > Xilinx Design Tools > Vivado 2023.2 > Vivado 2023.2.

A screenshot of a computer

AI-generated content may be incorrect.

Figure 1‑: Launching the Vivado Design Suite from the Start Menu

-- OR --

Double-click the Vivado Design Suite shortcut icon (A logo with a black background

AI-generated content may be incorrect.) on the desktop.

The Vivado Design Suite opens to the Welcome window. From the Welcome window you can create a new project, open an existing project, or enter Tcl commands directly into the Vivado Design Suite as well as access documentation and examples.

A screenshot of a computer

AI-generated content may be incorrect.

Figure 1‑: Vivado Design Suite Welcome Screen

"Create Project" is the starting point for all designs. Projects contains sources, settings, graphics, IP, and other elements that are used to build a final bitstream and analyze a design. The Create New Project Wizard in the Vivado Design Suite allows you to specify HDL and other project resource files that will be included in the project.

1-2. Create a new blank Vivado Design Suite project.

1-2-1. Click Create Project to begin the process (1).

A screenshot of a computer

AI-generated content may be incorrect.

Figure 1‑: Creating a New Vivado Design Suite Project

This will launch the New Project Wizard.

1-2-2. Click Next to exit the introductory dialog box and begin entering in project-specific information (2).

1-3. You will now encounter a series of dialog boxes asking you to enter different pieces of information describing the project.

1-3-1. Enter lab2\_uartLED in the Project name field.

1-3-2. Enter C:\training\dsgnZynqSDR\labs\L in the Project location field.

Alternatively, you can use the browse feature to navigate to where you want the project to reside.

1-3-3. Deselect the Create Project Subdirectory option as leaving this checked will create an unnecessary level of hierarchy for this lab.

A screenshot of a computer

AI-generated content may be incorrect.

Figure 1‑: Entering the Project Name and Location

1-3-4. Click Next to advance to the next dialog box.

Here you will choose between an RTL project or a post-synthesis project. Simply put, an RTL project enables you to add or create new HDL files and synthesize them, whereas the post-synthesis project requires pre-synthesized files. When an empty design is created, an RTL project is used.

1-3-5. Select RTL Project (1).

1-3-6. Select Do not specify sources at this time (2), which creates a blank project.

While existing sources could be entered at this time, you will enter them later so that you can move through this portion of the project creation process more quickly.

A screenshot of a computer

AI-generated content may be incorrect.

Figure 1‑: Selecting Project Type

1-3-7. Click Next to advance to the target device/platform selection (3).

1-4. Select the target part by first filtering by board and then by family. If you are not using a supported board, you will need to filter by part.

1-4-1. Select Parts from the Select area to filter by board rather than by the specific part (1).

1-4-2. Select Search and type “xc7z020clg”.

This limits the number of boards seen to those manufactured by the specified vendor.

1-4-3. Select xc7z020clg484-1 from the list.

Alternatively you can select the board directly from the list at any time while in this dialog box.

A screenshot of a computer

AI-generated content may be incorrect.

Figure 1‑: Selecting the Board for the Project

1-4-4. Click Next to advance to the summary (3).

A summary of your project is displayed. If you want to change any of the information that you entered, you can do so now by clicking Back until you reach the correct dialog box and making the correction, or you can create the project now and edit the project properties, add or remove files, etc. later.

1-4-5. Click Finish to accept these settings and build the project.

Your project is constructed and leaves you in the operational portion of the Vivado Design Suite GUI.

1-4-6. Click Settings under Project Manager in the Flow Navigator and change the Target Language to VHDL.

### Creating a New Block Design Step

Now that the project has been created, it is time to begin creating the design. You will only block design without any code to learn concept of Vivado design flow.

2-1. Create a Vivado block design.

2-1-1. Select Create Block Design in the Flow Navigator under Project Manager.

A screenshot of a computer

AI-generated content may be incorrect.

Figure 1‑: Selecting Create Block Design

2-1-2. Select Add new IP in the block design.

A screenshot of a computer

AI-generated content may be incorrect.

Figure 1‑: Selecting Add new IP

2-1-3. The IP list will appear on screen.

2-1-4. Search for “Zynq” and click Zynq7 Processing System. And click “Run Block Automation”

A close-up of a computer chip

AI-generated content may be incorrect.

Figure 1‑10: Zynq Processor as an IP

2-1-5. Double click on Zynq to customize the IP

A computer screen shot of a computer

AI-generated content may be incorrect.

Figure 1‑11: Zynq Block Design with out any customization

2-1-6. Select MIO Configuration then I/O Peripherals, select Bank0 and Bank1 to be LVCMOS 3.3V and LVCMOS 1.8V consecutively, then enable UART1 set it to be MIO 48..49, double check with schamatic.

A screenshot of a computer

AI-generated content may be incorrect. .

A screenshot of a computer

AI-generated content may be incorrect.

Figure 1‑12: Enable UART1 at MIO Configuration

2-1-7. Scroll down to GPIO and select EMIO and enter 8 for width

A screenshot of a computer

AI-generated content may be incorrect.

Figure 1‑13: Enable EMIO for LED

2-1-8. Select DDR Configuration and enable Internal Vref and choose MT41J128M16HA-15E for memory IC.

A screenshot of a computer

AI-generated content may be incorrect.

Figure 1‑13: Enable EMIO for LED

2-1-7. Scroll down to Training/Board details and add DDR delay

A screenshot of a computer

AI-generated content may be incorrect.

Figure 1‑14: Add DDR clock delays

2-1-8. Select PS-PL and select speed for UART1 as 115200

A screenshot of a computer

AI-generated content may be incorrect.

Figure 1‑15: Uart1 speed

2-1-9. Scroll down to AXI Non Secure Enablement, uncheck on “M AXI GP0 Interface”

A screenshot of a computer

AI-generated content may be incorrect.

Figure 1‑16: Uncheck AXI input for PS

2-1-8. Click OK on “Zynq7 Processing System”. Ignore warning message

2-1-9. Double click on GPIO\_0+ to unfold the port. Then click on GPIO\_0 and “Make External”.

A computer system diagram with wires

AI-generated content may be incorrect.

Figure 1‑17: Block design with Zynq

2-1-10. Generate block design and create HDL wrapper

### Synthesizing theDesign Step 3

3-1. Synthesizing the Design

3-1-1. Similar to Lab1, Select “Run Linter” in RTL Analysis then “Run Synthesis” in Systhesis

A screenshot of a computer

AI-generated content may be incorrect.

3-1-2. Open Synthesis Design and set up I/O pins for LED (similar to Lab1)

A screenshot of a computer

AI-generated content may be incorrect.

Figure 1‑18: GPIO with BGA pins

3-1-3. Select voltage on I/O to be 3.3V and set the BGA pins for LED (as shown in Lab1)

3-1-4. Select “Run Synthesis” in Systhesis again, save constrain file as “led2\_helloworldLED”

3-1-5. Select “Run Implementation” in Implementation then “Generate Bitstream” in Program and Debug

3-1-6. Go to File in menu bar, select export then export hardware

A screenshot of a computer

AI-generated content may be incorrect.

Figure 1‑20: Export hardware

A screenshot of a software program

AI-generated content may be incorrect.

Figure 1‑21: Select export hardware with Bitstream

3-6-7. Set XSA filename to be lab2\_helloworldLED click next and finish.

### Creating a New Vitis Platform Project Step 4

4-1. Create Vitis Platform Project to Contain Hardware information

4-1-1. Open Vitis, you can open from Windows or select tools and Launch Vitis 2023.2.

4-1-2. If it is your first time using Vitis IDE, create workspace by select File menu and Open Workspace. The file menu will pop up, create new directory for workspace and select your new directory then click on Select Folder. For the latter time, you click on Open Recent Workspace and select your workspace to work.

A screenshot of a computer

AI-generated content may be incorrect.

4-1-3. Create New Component Platform Project, select File in the top menu, then New Component and “Platform Project”

A screenshot of a computer

AI-generated content may be incorrect.

Figure 1‑22: Select new platform project

4-1-4. Enter “mylab2\_zedboard” or any name you like as a Platform project name, click Next

A screenshot of a computer

AI-generated content may be incorrect.

4-1-5. Click on Browse to and select the \*.xsa file from your lab2 Vivado project. Then click Next.

A screenshot of a computer

AI-generated content may be incorrect.

Figure 1‑23: Choose our own hardware from Vivado project

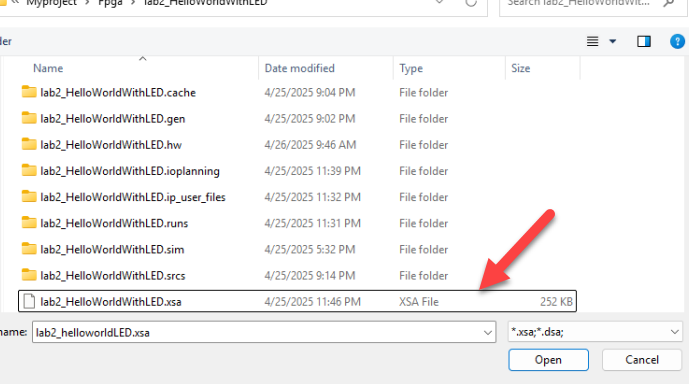


Figure 1‑24: Choose XSA file from Vivado project

4-1-6. For this project we do not use Operating system (Bare metal project) so we select opeating system as standalone and the processor is ps7\_cortexa9\_0. Makesure the option Generate Boot artifacts is checked. Then click Next and Finish.

A screenshot of a computer

AI-generated content may be incorrect.

Figure 1‑25: Choose previously created Platform Project

4-1-7. Click Build on Flow menu to create First Stage Boot Loader (FSBL) to be use later for Sdcard. The fsbl.elf (zynq\_fsbl.elf) will be created after building process.

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

### Add code to Vitis Step 5

5-1. Create Vitis Application Project to Contain our Software

5-1-1. Vitis creates application project by choosing a existing code from example.

5-1-2. The software can be written entirely from scratch or to simplify everything, we choose to write the software on “Helloworld demo”.

5-1-3. Select File in the top menu, then New Component and “From Example”

A screenshot of a computer

AI-generated content may be incorrect.

Figure 1‑26: Choose “Hello World” as template

5-1-4. The list of example will be showed on Examples menu, double click on “Hello World”.

A screen shot of a computer

AI-generated content may be incorrect.

5-1-5. The tab “Hello World” will be appeared on the main window box. Clikc on “Create Application Component from Template”

A screenshot of a computer

AI-generated content may be incorrect.

5-1-6 Enter the application name as “mylab2\_ps\_led” or any name you like. Then click Next.

A screenshot of a computer

AI-generated content may be incorrect.

5-1-7 Choose platform that we just create in last topic, (mylab2\_zedboard) and click Next.

\*\*\*If there is no platform shown, close Vitis and open again.\*\*\*\*

A screenshot of a computer

AI-generated content may be incorrect.

5-1-7 Choose CPU core to run application select “standalone\_ps7\_cortexa9\_0”, then click Next and Finish

A screenshot of a computer

AI-generated content may be incorrect.

5-1-8. Click Build on the flow window

. A screenshot of a computer

AI-generated content may be incorrect.

5-1-9. Open Terraterm select the serial port connecting to zedboard and speed of 115200. Then click on Run. And observer the Terraterm.

A screenshot of a computer program

AI-generated content may be incorrect.

A black and white screen with white text

AI-generated content may be incorrect.

5-2. Add a custom code to control LED Knight Rider Style

5-2-1. Click project name on Workspace window, click on Sources and src to see C code. Then double click helloworld.c to edit code.

A screen shot of a computer

AI-generated content may be incorrect.

5-2-2. Replace origitnal code in helloworld.c with this code.

**“**

#include "xparameters.h"

#include "xgpiops.h"

#include "xstatus.h"

#include "sleep.h"

#include "xil\_printf.h"

#define PIN\_OFFSET 54 // EMIO[0] = GPIO Pin 54

XGpioPs Gpio;

int main()

{

XGpioPs\_Config \*ConfigPtr;

int Status;

// Init PS GPIO

ConfigPtr = XGpioPs\_LookupConfig(0);

Status = XGpioPs\_CfgInitialize(&Gpio, ConfigPtr, ConfigPtr->BaseAddr);

if (Status != XST\_SUCCESS)

return XST\_FAILURE;

// Set EMIO[0:7] เป็น Output

for (int i = 0; i < 8; i++) {

XGpioPs\_SetDirectionPin(&Gpio, PIN\_OFFSET + i, 1);

XGpioPs\_SetOutputEnablePin(&Gpio, PIN\_OFFSET + i, 1);

}

xil\_printf("Knight Rider LED Effect via EMIO GPIO\r\n");

int pos = 0;

int dir = 1; // 1 = ไปขวา, -1 = ไปซ้าย

while (1) {

// เปิดแค่ 1 ดวงตามตำแหน่ง

for (int i = 0; i < 8; i++) {

XGpioPs\_WritePin(&Gpio, PIN\_OFFSET + i, (i == pos) ? 1 : 0);

}

usleep(100000); // 100 ms delay

pos += dir;

if (pos == 7 || pos == 0)

dir = -dir; // เปลี่ยนทิศทางเมื่อสุดขอบ

}

return 0;

}

**“**

5-2-3. Click Build on the flow window

. A screenshot of a computer

AI-generated content may be incorrect.

5-2-4. Open Terraterm select the serial port connecting to zedboard and speed of 115200. Then click on Run. And observer the Terraterm.

A screenshot of a computer program

AI-generated content may be incorrect.

## Summary

This lab shows how Vitis control GPIO on the FPGA