CEC Continuing Education Center

DesignNews

Field-Programmable Gate Array (FPGA) Primer

Day 1:
Vivado ML Installation

Sponsored by

Digi-Key ELECTRONICS

informa markets

Webinar Logistics

- Turn on your system sound to hear the streaming presentation.
- If you have technical problems, click “Help” or submit a question asking for assistance.
- Participate in ‘Attendee Chat’ by maximizing the chat widget in your dock.

Fred Eady

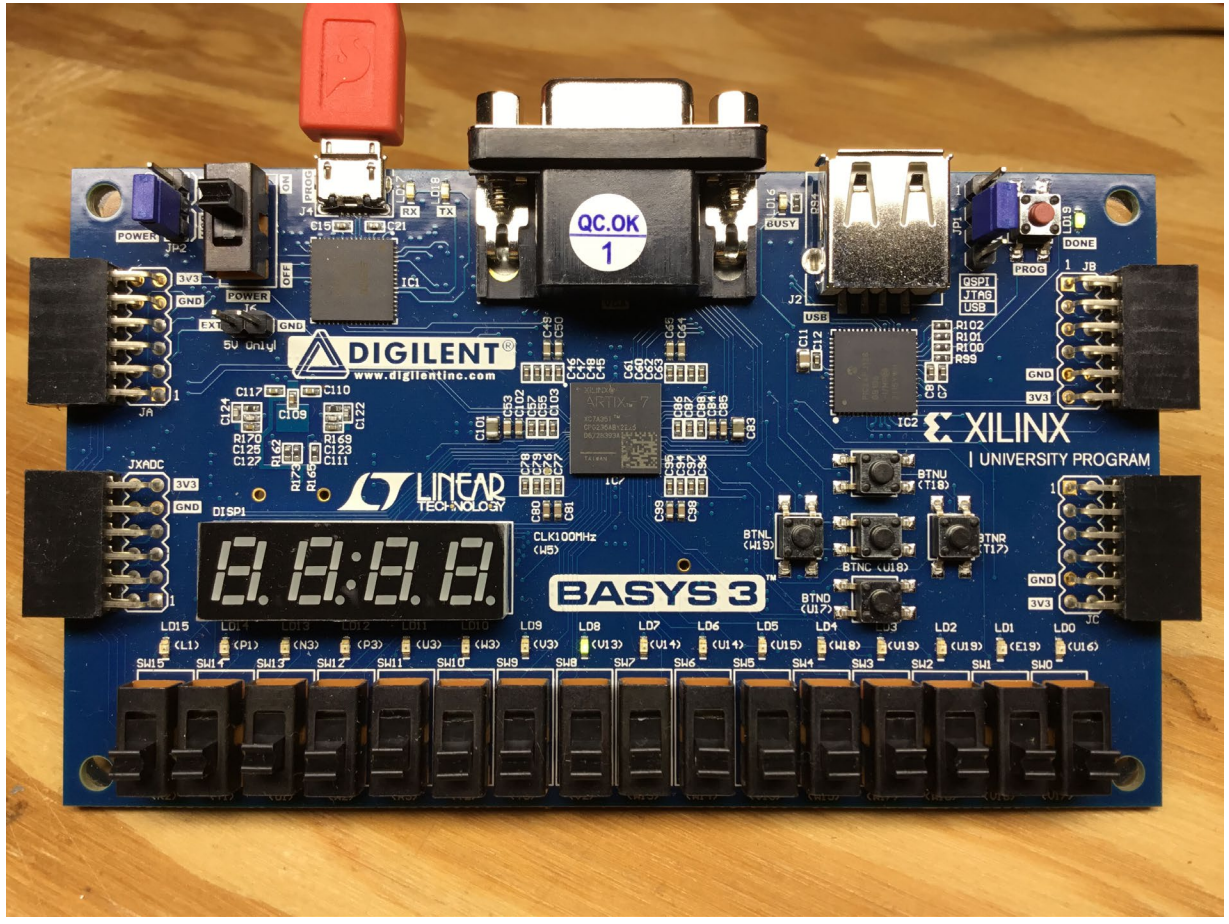
Visit ‘Lecturer Profile’ in your console for more details.

CEC Continuing Education Center

Sponsored By Digi-Key Corporation

AGENDA

- Ubuntu Installation Prerequisites
- Vivado Design Suite Installation
- A Very Expensive But Fancy LED Blinker



Install libncurses5 and libtinfo5

fred@shopUbuntu885: ~

File Edit View Search Terminal Help

```
fred@shopUbuntu885:~$ sudo apt install libncurses5
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  libtinfo5
The following NEW packages will be installed:
  libncurses5 libtinfo5
0 upgraded, 2 newly installed, 0 to remove and 2 not upgra
Need to get 207 kB of archives.
After this operation, 883 kB of additional disk space will
Do you want to continue? [Y/n] Y
Get:1 http://us.archive.ubuntu.com/ubuntu jammy-updates/ur
 amd64 6.3-2ubuntu0.1 [100 kB]
Get:2 http://us.archive.ubuntu.com/ubuntu jammy-updates/ur
s5 amd64 6.3-2ubuntu0.1 [107 kB]
Fetched 207 kB in 0s (417 kB/s)
Selecting previously unselected package libtinfo5:amd64.
(Reading database ... 201071 files and directories current
Preparing to unpack .../libtinfo5_6.3-2ubuntu0.1_amd64.deb
Unpacking libtinfo5:amd64 (6.3-2ubuntu0.1) ...
Selecting previously unselected package libncurses5:amd64.
Preparing to unpack .../libncurses5_6.3-2ubuntu0.1_amd64.d
```

Allows Simulation to run

Allows Vivado to start

fred@shopUbuntu885: ~

```
fred@shopUbuntu885:~$ sudo apt install libtinfo5
[sudo] password for fred:
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
libtinfo5 is already the newest version (6.3-2ubuntu0.1).
libtinfo5 set to manually installed.
0 upgraded, 0 newly installed, 0 to remove and 2 not upgraded.
fred@shopUbuntu885:~$
```

Download the Unified Installer

AMD Unified Installer for FPGAs & Adaptive SoCs 2023.1: Windows Self Extracting Web Installer (EXE - 199.47 MB)

MD5 SUM Value : 4c6a1e5d5cf7c44c3f201c9056b6cf45

Download Verification

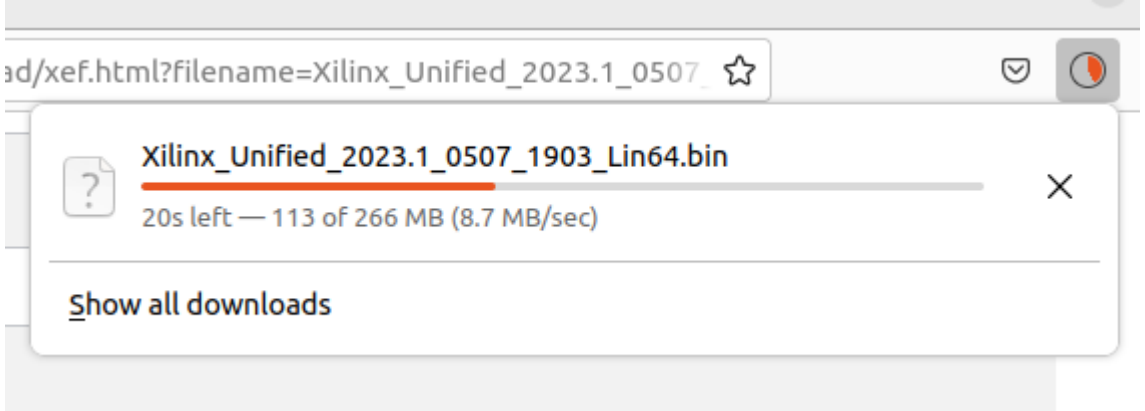
Digests | Signature | Public Key

AMD Unified Installer for FPGAs & Adaptive SoCs 2023.1: Linux Self Extracting Web Installer (BIN - 265.94 MB)

MD5 SUM Value : e47ad71388b27a6e2339ee82c3c8765f

Download Verification

Digests | Signature | Public Key



Run the Unified Installer

fred@shopUbuntu885: ~/Downloads

File Edit View Search Terminal Help

```
Xilinx_Unified_2023.1_0507_1903_Lin64.bin
fred@shopUbuntu885:~/Downloads$ chmod +x Xilinx_Unified_2023.1_0507_1903_Lin64.bin
fred@shopUbuntu885:~/Downloads$ sudo ./Xilinx_Unified_2023.1_0507_1903_Lin64.bin
[sudo] password for fred:
Verifying archive integrity... All good.
Uncompressing AMD Installer for FPGAs and Adaptive SoCs.....................................
.........................................................................................
.........................................................................................
.........................................................................................
.........................................................................................
.........................................................................................
.........................................................................................
.........................................................................................
.........................................................................................
.........
This is a fresh install.
INFO Could not detect the display scale (hDPI).
       If you are using a high resolution monitor, you can set the insaller scale factor like th
is:
       export XINSTALLER_SCALE=2
       setenv XINSTALLER_SCALE 2
INFO  - Started in: 2 Sec
INFO  - Internet connection validated, can connect to internet.
```

Install Vitis.. Not Vivado?

AMD Unified Installer for FPGAs & Adaptive SoCs 2023.1 - Select Install Type

Select Install Type

Please select install type and provide your AMD.com E-mail Address and password for authentication.

User Authentication

Please provide your AMD user account credentials to download the required files.
If you don't have an account, please create one. If you forgot your password, you can reset it here.

E-mail Address: therealfredeady@gmail.com

Password:

(•) Download and Install Now

Select your desired device and tool installation options and the installer will download and install just what is requi

( ) Download Image (Install Separately)

The installer will download an image containing all devices and tool options for later installation. Use this option if a full image on a network drive or allow different users maximum flexibility when installing.

< Back | Next > | Cancel

AMD Unified Installer for FPGAs & Adaptive SoCs 2023.1 - Select Product to Install

Select Product to Install

Select a product to continue installation. You will be able to customize the content in the next page.

(•) **Vitis**

Installs Vitis Core Development Kit for embedded software and application acceleration development on AMD platforms. Vitis installation includes Vivado Design Suite. Users can also install Vitis Model Composer to design for AI Engines and Programmable Logic in MATLAB and Simulink.

( ) Vivado

Includes the full complement of Vivado Design Suite tools for design, including C-based design with Vitis High-Level Synthesis, implementation, verification and device programming. Complete device support, cable driver, and Document Navigator included. Users can also install Vitis Model Composer to design for AI Engines and Programmable Logic in MATLAB and Simulink.

( ) On-Premises Install for Cloud Deployments

Install on-premises version of tools for cloud deployments.

( ) BootGen

Installs Bootgen for creating bootable images targeting AMD SoCs and FPGAs.

( ) Lab Edition

Installs only the Vivado Lab Edition. This standalone product includes Vivado Design Programmer, Vivado Logic Analyzer and UpdateMEM tools.

( ) Hardware Server

Installs hardware server and JTAG cable drivers for remote debugging.

( ) Power Design Manager (PDM)

Installs only the Power Design Manager (PDM). Power Design Manager is a standalone design tool used to estimate power

< Back | Next > | Cancel

Customize Installation for BASYS 3

AMD Unified Installer for FPGAs & Adaptive SoCs 2023.1 - Vitis Unified Software Platform

Vitis Unified Software Platform

Customize your installation by (de)selecting items in the tree below. Moving cursor over selections below provide additional information.

The Vitis unified software platform enables the development of embedded software and accelerated applications on heterogeneous AMD platforms including FPGAs, SoCs, and Versal ACAPs. It provides a unified programming model for accelerating Edge, Cloud, and Hybrid computing applications. This installation is a superset that includes the Vivado Design Suite as well. Users can add Vitis Model Composer which is a AMD toolbox for MATLAB and Simulink to design for AI Engines and Programmable Logic. If you have been using AMD System Generator for DSP, you can continue development using Vitis Model Composer.

- Design Tools
  - [-] Vitis Unified Software Platform
    - [x] Vitis
    - [ ] Vitis IP Cache (Enable faster on-boarding for new users)
    - [x] Vivado
    - [x] Vitis HLS
  - [x] Vitis Model Composer(Toolbox for MATLAB and Simulink. Includes the functionality of System Generator for DSP)
  - [x] DocNav
- Devices
  - [ ] Install devices for Alveo and edge acceleration platforms
  - [ ] Install Devices for Kria SOMs and Starter Kits
  - [-] Devices for Custom Platforms
    - [ ] SoCs
    - [-] 7 Series
      - [x] Artix-7
      - [ ] Kintex-7
      - [ ] Spartan-7
      - [ ] Virtex-7
    - [ ] UltraScale
    - [ ] UltraScale+
    - [ ] Versal ACAP
  - [ ] Engineering Sample Devices for Custom Platforms
- Installation Options
  - [ ] NOTE: Cable Drivers are not installed on Linux. Please follow the instructions in UG973 to install Linux cable drivers

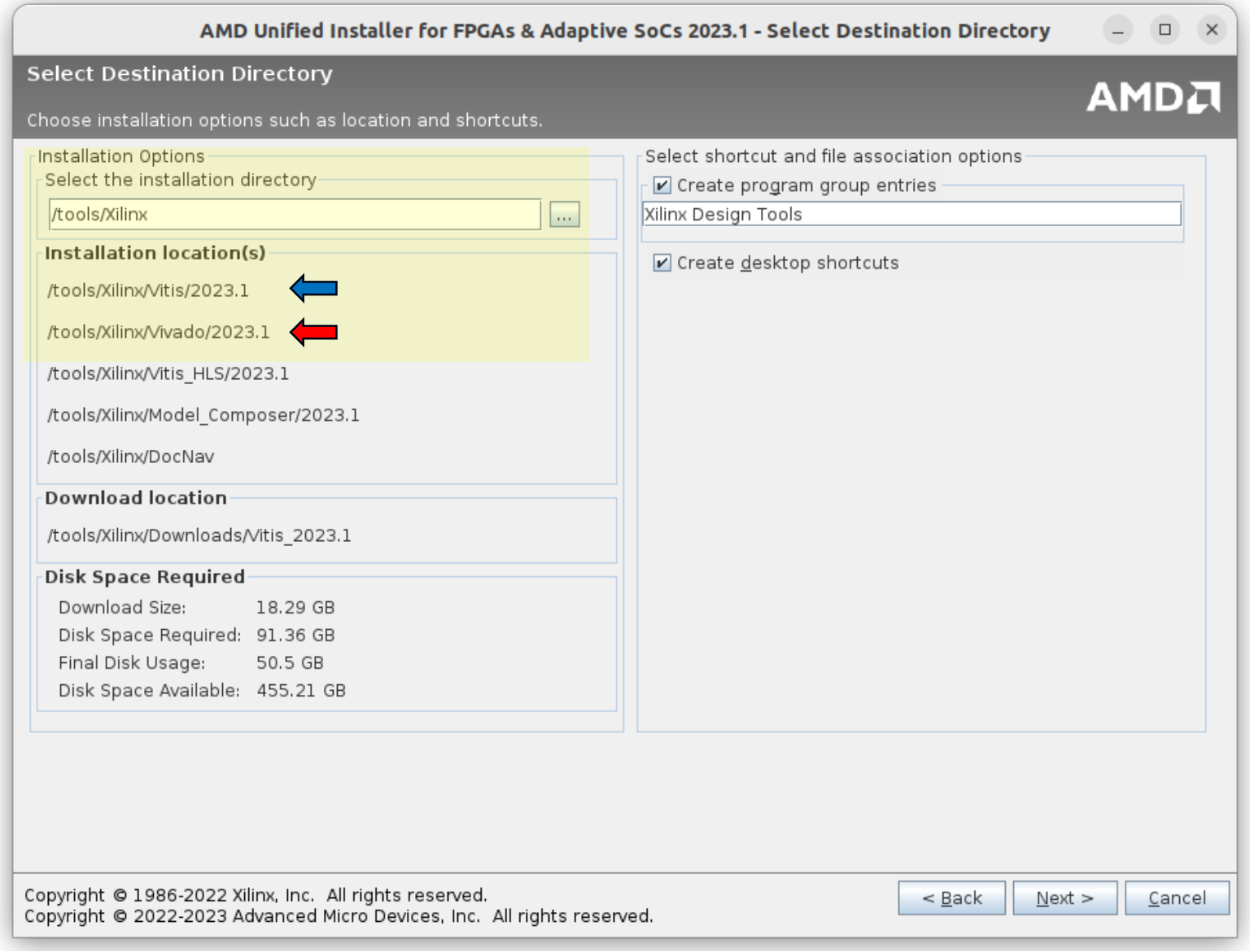
BASYS 3

Download Size: 18.29 GB
Disk Space Required: 91.36 GB

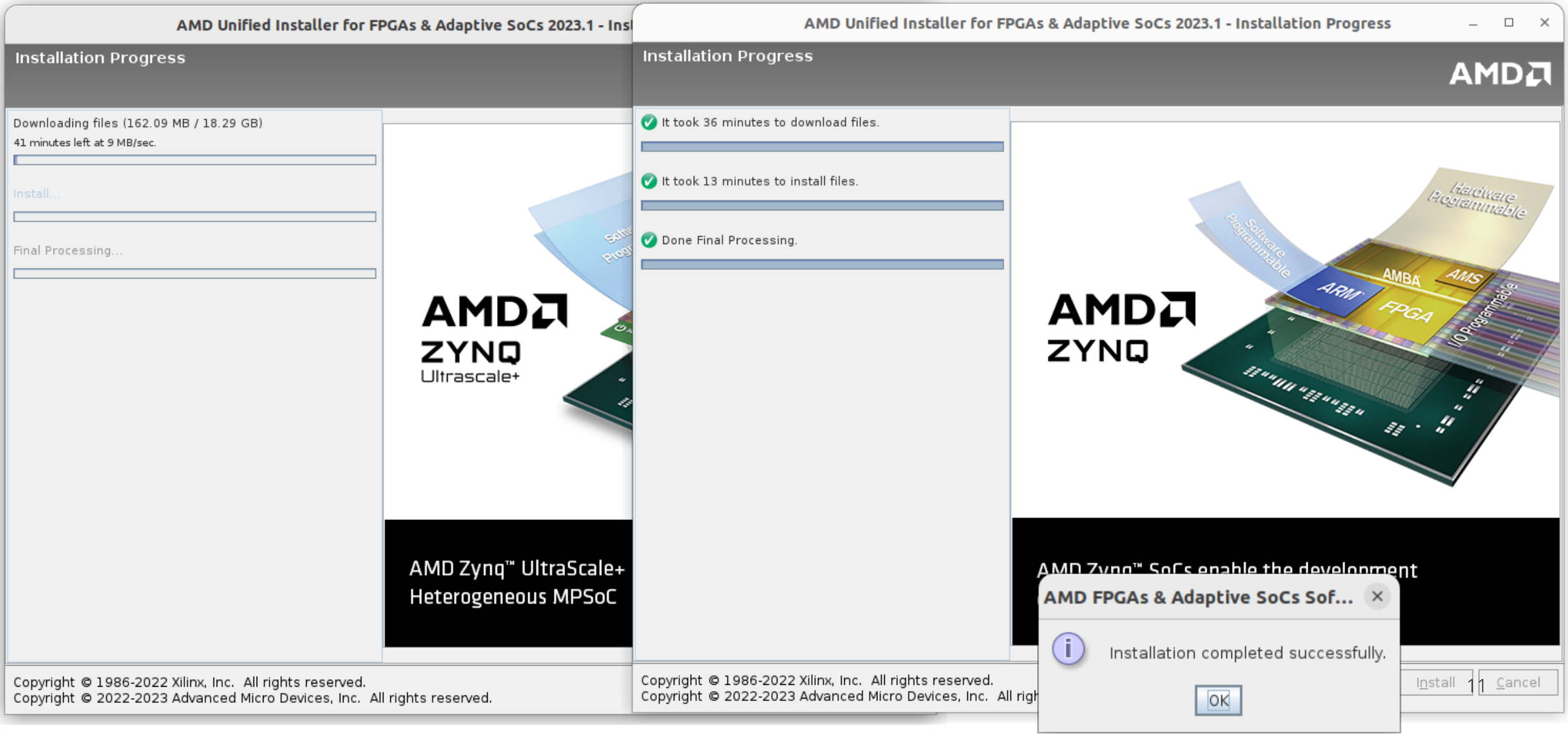
Reset to Defaults

< Back | Next > | Cancel

Recommendation: Use the Default Installation Directory



Hurry Up and Wait



Install the Cable Drivers

fred@shopUbuntu885: /tools/Xilinx/Vivado/2023.1/data/xicom/cable_drivers/lin64/install_script/install_drivers

```
fred@shopUbuntu885:~$ cd /tools/Xilinx/Vivado/2023.1/data/xicom/cable_drivers/lin64/install_script/install_drivers
fred@shopUbuntu885:/tools/Xilinx/Vivado/2023.1/data/xicom/cable_drivers/lin64/install_script/install_drivers$ sudo ./install_drivers
[sudo] password for fred:
INFO: Installing cable drivers.
INFO: Script name = ./install_drivers
INFO: HostName = shopUbuntu885
INFO: RDI_BINROOT= .
INFO: Current working dir = /tools/Xilinx/Vivado/2023.1/data/xicom/cable_drivers/lin64/install_script/install_drivers
INFO: Kernel version = 6.2.0-31-generic.
INFO: Arch = x86_64.
USB udev file exists and will not be updated.
--File /etc/udev/rules.d/52-xilinx-ftdi-usb.rules exists.
--File /etc/udev/rules.d/52-xilinx-ftdi-usb.rules version = 0001
--File 52-xilinx-ftdi-usb.rules exists.
--File 52-xilinx-ftdi-usb.rules version = 0001
--File 52-xilinx-ftdi-usb.rules is already updated.
--File /etc/udev/rules.d/52-xilinx-pcusb.rules exists.
--File /etc/udev/rules.d/52-xilinx-pcusb.rules version = 0002
--File 52-xilinx-pcusb.rules exists.
--File 52-xilinx-pcusb.rules version = 0002
--File 52-xilinx-pcusb.rules is already updated.

INFO: Digilent Return code = 0
INFO: Xilinx Return code = 0
INFO: Xilinx FTDI Return code = 0
INFO: Return code = 0
INFO: Driver installation successful.
CRITICAL WARNING: Cable(s) on the system must be unplugged then plugged back in order for the driver scripts to update the cables.
fred@shopUbuntu885:/tools/Xilinx/Vivado/2023.1/data/xicom/cable_drivers/lin64/install_script/install_drivers$
```

Setup to Launch Vivado

fred@shopUbuntu885: ~

File Edit View Search Terminal Help

```
fred@shopUbuntu885:/tools/Xilinx/Vivado/2023.1$ ./settings64.sh
fred@shopUbuntu885:/tools/Xilinx/Vivado/2023.1$ cd
fred@shopUbuntu885:~$ vivado

****** Vivado v2023.1 (64-bit)
  **** SW Build 3865809 on Sun May  7 15:04:56 MDT 2023
  **** IP Build 3864474 on Sun May  7 20:36:21 MDT
  **** SharedData Build 3865790 on Sun May 07 13:33
    ** Copyright 1986-2022 Xilinx, Inc. All Rights
    ** Copyright 2022-2023 Advanced Micro Devices,

start_gui
INFO: [Common 17-206] Exiting Vivado at Fri Sep  1
fred@shopUbuntu885:~$ sudo adduser $USER dialout
[sudo] password for fred:
The user `fred' is already a member of `dialout'.
fred@shopUbuntu885:~$
```

Enable USB drivers to be used with a serial terminal

UG893 Page 9

XILINX

Chapter 1: Introduction

Launching the Vivado IDE from the Command Line on Windows or Linux

Enter the following command at the command prompt:

```
<install_path>/Vivado/<version>/bin/vivado
```

Note: When you enter this command, it automatically runs `vivado -mode gui` to launch the Vivado IDE. If you need help, type `vivado -help`.

TIP: *To add the Vivado tools path to your current shell/command prompt, run `settings64.bat` or `settings64.sh` from the `<install_path>/Vivado/<version>` directory.*

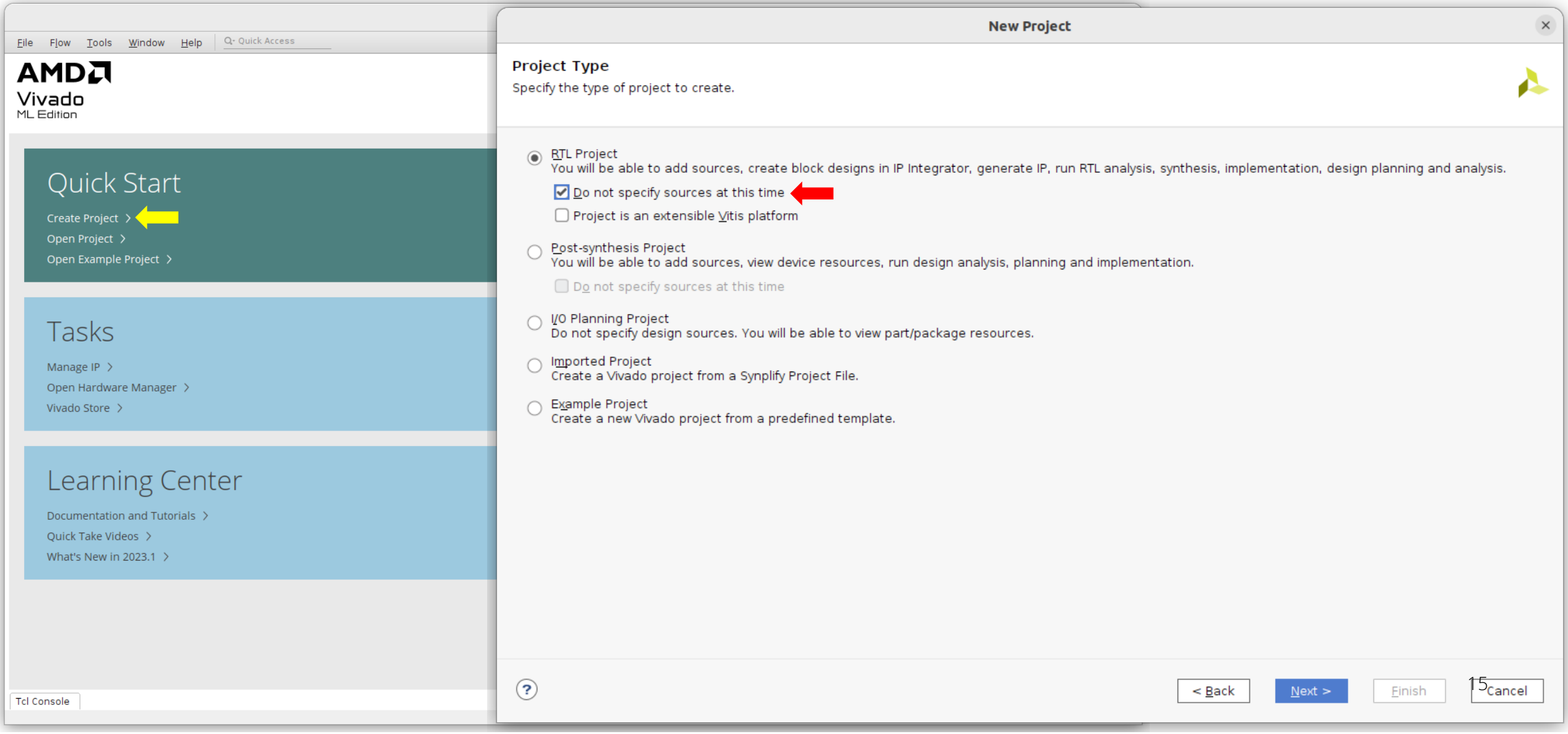
Preferred Setup to Launch Vivado

Open .bashrc ~/ Save

```
    alias egrep='egrep --color=auto'
fi

# colored GCC warnings and errors
#export GCC_COLORS='error=01;31:warning=01;35:note=01;36:caret=01;32:locus=01:quote=01'

# some more ls aliases
alias ll='ls -alF'
alias la='ls -A'
alias l='ls -CF'

# Add an "alert" alias for long running commands.  Use like so:
#   sleep 10; alert
alias alert='notify-send --urgency=low -i "$([ $? = 0 ] && echo terminal || echo error)" "$(history|tail -n1|sed -e '\''s/^\s*[0-9]\+\s*//;s/[;&|]\s*alert$//'\'')"'

# Alias definitions.
# You may want to put all your additions into a separate file like
# ~/.bash_aliases, instead of adding them here directly.
# See /usr/share/doc/bash-doc/examples in the bash-doc package.

if [ -f ~/.bash_aliases ]; then
    . ~/.bash_aliases
fi

# enable programmable completion features (you don't need to enable
# this, if it's already enabled in /etc/bash.bashrc and /etc/profile
# sources /etc/bash.bashrc).
if ! shopt -oq posix; then
  if [ -f /usr/share/bash-completion/bash_completion ]; then
    . /usr/share/bash-completion/bash_completion
  elif [ -f /etc/bash_completion ]; then
    . /etc/bash_completion
  fi
fi

source /tools/Xilinx/Vivado/2023.1/settings64.sh
```

sh Tab Width: 8 Ln 85, Col 3 INS

Installation Verification – Create a New Project



Installation Verification – Create a New Project

expensive_led_blinker - [/home/fred/expensive_led_blinker/expensive_led_blinker.xpr] - Vivado 2023.1

File Edit Flow Tools Reports Window Layout View Help

write_bitstream Complete

Default Layout

Flow Navigator

- PROJECT MANAGER
  - Settings
  - Add Sources
  - Language Templates
  - IP Catalog
- IP INTEGRATOR
  - Create Block Design
  - Open Block Design
  - Generate Block Design
- SIMULATION
  - Run Simulation
- RTL ANALYSIS
  - Run Linter
  - Open Elaborated Design
- SYNTHESIS
  - Run Synthesis
  - Open Synthesized Design
- IMPLEMENTATION
  - Run Implementation
  - Open Implemented Design
- PROGRAM AND DEBUG
  - Generate Bitstream
  - Open Hardware Manager

PROJECT MANAGER - expensive_led_blinker

Sources

- Design Sources (1)
  - fancy_blinker_top (fancy_blinker_top.v) (2)
    - led_clk_gen : led_clk_generator (led_clk_generator.v)
    - led_drvr : led_driver (led_driver.v)
- Constraints (1)
  - constrs_1 (1)
    - Basys-3-Master.xdc
- Simulation Sources (1)
- Utility Sources (1)

Hierarchy | Libraries | Compile Order

Synthesis Run Properties — synth_1

Name:	synth_1
Part:	xc7a35tcpg236-1 (active)
Description:	Vivado Synthesis Defaults
Status:	synth_design Complete!
Constraints:	constrs_1 (active)
Run directory:	/home/fred/expensive_led_blinker/expensive_led_blinker.runs

General | Properties | Options | Log | Reports | Messages

Project Summary | **Basys-3-Master.xdc**

/home/fred/expensive_led_blinker/Basys-3-Master.xdc

```
## Clock signal
set_property -dict { PACKAGE_PIN W5   IOSTANDARD LVCMOS33 } [get_ports clk]
create_clock -add -name sys_clk_pin -period 10.00 -waveform {0 5} [get_ports clk]
```

Sources

- Design Sources (1)
  - fancy_blinker_top (fancy_blinker_top.v) (2)
    - led_clk_gen : led_clk_generator (led_clk_generator.v)
    - led_drvr : led_driver (led_driver.v)
- Constraints (1)
  - constrs_1 (1)
    - Basys-3-Master.xdc
- Simulation Sources (1)
- Utility Sources (1)

Hierarchy | Libraries | Compile Order

Tcl Console | Messages | Log | Reports | Design Runs

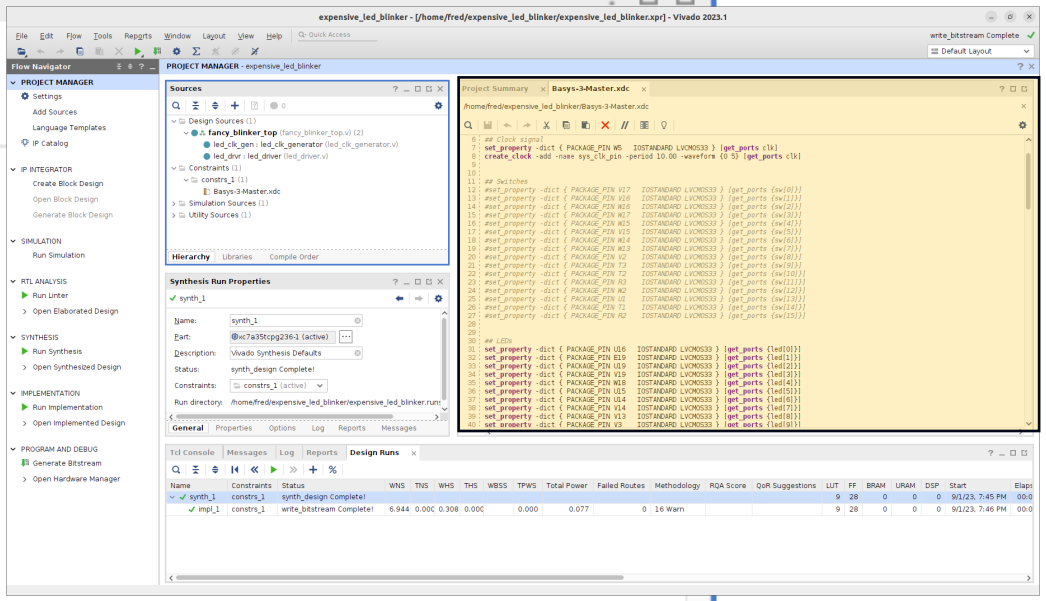
Name	Constraints	Status	WNS	TNS	WHS	THS	WBSS																
synth_1	constrs_1	synth_design Complete!																					
impl_1	constrs_1	write_bitstream Complete!	6.944	0.000	0.308	0.000		0.000	0.077	0	16 Warn			9	28	0	0	0	9/1/23, 7:46 PM	00:0

Installation Verification – Basys-3-Master.xdc

Project Summary | Basys-3-Master.xdc

/home/fred/expensive_led_blinker/Basys-3-Master.xdc

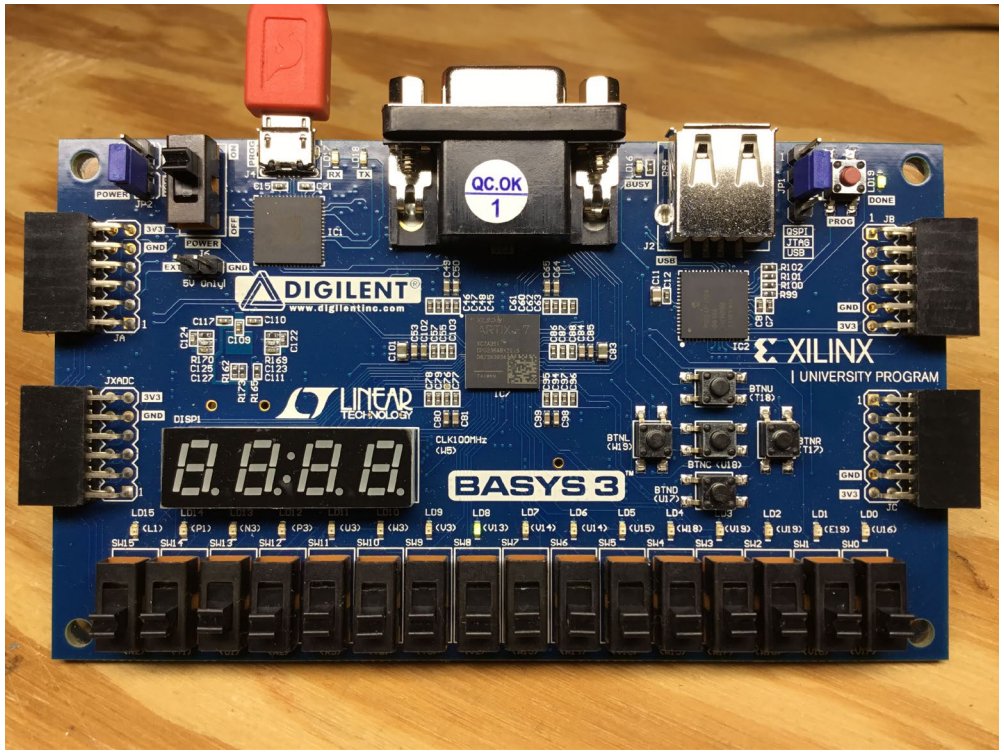
```
#set_property -dict { PACKAGE_PIN V2    IOSTANDARD LVCMOS33 } [get_ports {sw[8]}]
#set_property -dict { PACKAGE_PIN T3    IOSTANDARD LVCMOS33 } [get_ports {sw[9]}]
#set_property -dict { PACKAGE_PIN T2    IOSTANDARD LVCMOS33 } [get_ports {sw[10]}]
#set_property -dict { PACKAGE_PIN R3    IOSTANDARD LVCMOS33 } [get_ports {sw[11]}]
#set_property -dict { PACKAGE_PIN W2    IOSTANDARD LVCMOS33 } [get_ports {sw[12]}]
#set_property -dict { PACKAGE_PIN U1    IOSTANDARD LVCMOS33 } [get_ports {sw[13]}]
#set_property -dict { PACKAGE_PIN T1    IOSTANDARD LVCMOS33 } [get_ports {sw[14]}]
#set_property -dict { PACKAGE_PIN R2    IOSTANDARD LVCMOS33 } [get_ports {sw[15]}]


## LEDs
set_property -dict { PACKAGE_PIN U16   IOSTANDARD LVCMOS33 } [get_ports {led[0]}]
set_property -dict { PACKAGE_PIN E19   IOSTANDARD LVCMOS33 } [get_ports {led[1]}]
set_property -dict { PACKAGE_PIN U19   IOSTANDARD LVCMOS33 } [get_ports {led[2]}]
set_property -dict { PACKAGE_PIN V19   IOSTANDARD LVCMOS33 } [get_ports {led[3]}]
set_property -dict { PACKAGE_PIN W18   IOSTANDARD LVCMOS33 } [get_ports {led[4]}]
set_property -dict { PACKAGE_PIN U15   IOSTANDARD LVCMOS33 } [get_ports {led[5]}]
set_property -dict { PACKAGE_PIN U14   IOSTANDARD LVCMOS33 } [get_ports {led[6]}]
set_property -dict { PACKAGE_PIN V14   IOSTANDARD LVCMOS33 } [get_ports {led[7]}]
set_property -dict { PACKAGE_PIN V13   IOSTANDARD LVCMOS33 } [get_ports {led[8]}]
set_property -dict { PACKAGE_PIN V3    IOSTANDARD LVCMOS33 } [get_ports {led[9]}]
set_property -dict { PACKAGE_PIN W3    IOSTANDARD LVCMOS33 } [get_ports {led[10]}]
set_property -dict { PACKAGE_PIN U3    IOSTANDARD LVCMOS33 } [get_ports {led[11]}]
set_property -dict { PACKAGE_PIN P3    IOSTANDARD LVCMOS33 } [get_ports {led[12]}]
set_property -dict { PACKAGE_PIN N3    IOSTANDARD LVCMOS33 } [get_ports {led[13]}]
set_property -dict { PACKAGE_PIN P1    IOSTANDARD LVCMOS33 } [get_ports {led[14]}]
set_property -dict { PACKAGE_PIN L1    IOSTANDARD LVCMOS33 } [get_ports {led[15]}]


##7 Segment Display
#set_property -dict { PACKAGE_PIN W7    IOSTANDARD LVCMOS33 } [get_ports {seg[0]}]
#set_property -dict { PACKAGE_PIN W6    IOSTANDARD LVCMOS33 } [get_ports {seg[1]}]
#set_property -dict { PACKAGE_PIN U8    IOSTANDARD LVCMOS33 } [get_ports {seg[2]}]
#set_property -dict { PACKAGE_PIN V8    IOSTANDARD LVCMOS33 } [get_ports {seg[3]}]
#set_property -dict { PACKAGE_PIN U5    IOSTANDARD LVCMOS33 } [get_ports {seg[4]}]
```



Installation Verification – LED Blinker Modules

```
module fancy_blinker_top(
    input clk,
    output [15:0] led,
    output [0:3] an
    );

    assign an = 4'b1111;

    wire [0:3] w_led_clk_out;

    led_clk_generator led_clk_gen(
    .clk(clk),
    .led_clk_out(w_led_clk_out)
    );

    led_driver led_drvr(
    .led_clk_in(w_led_clk_out),
    .led_select(led)
    );

 endmodule
```

```
module led_clk_generator(
    input clk,
    output [3:0] led_clk_out
    );

    //led_counter overflows at 16,777,216
    //blink rate = 16,777,216/100,000,000 = 0.1677216s = 5.96Hz
    reg [27:0] led_counter = 0;
    always @ (posedge clk)
    begin
       led_counter <= led_counter + 1;
    end
    assign led_clk_out[3:0] = led_counter[27:24];
endmodule
```




```
module led_driver(
    input [3:0] led_clk_in,
    output reg [15:0] led_select
    );

    always @*
    begin
       case(led_clk_in)
        4'b0000:
            begin
                led_select = 16'h0001;
            end
        4'b0001:
            begin
                led_select = 16'h0002;
            end
        4'b0010:...
        4'b0011:...
        4'b0100:...
        4'b0101:...
        4'b0110:...
        4'b0111:...
        4'b1000:...
        4'b1001:...
        4'b1010:...
        4'b1011:...
        4'b1100:...
        4'b1101:...
        4'b1110:...
        4'b1111:
            begin
                led_select = 16'h8000;
            end
        endcase
    end
endmodule
```

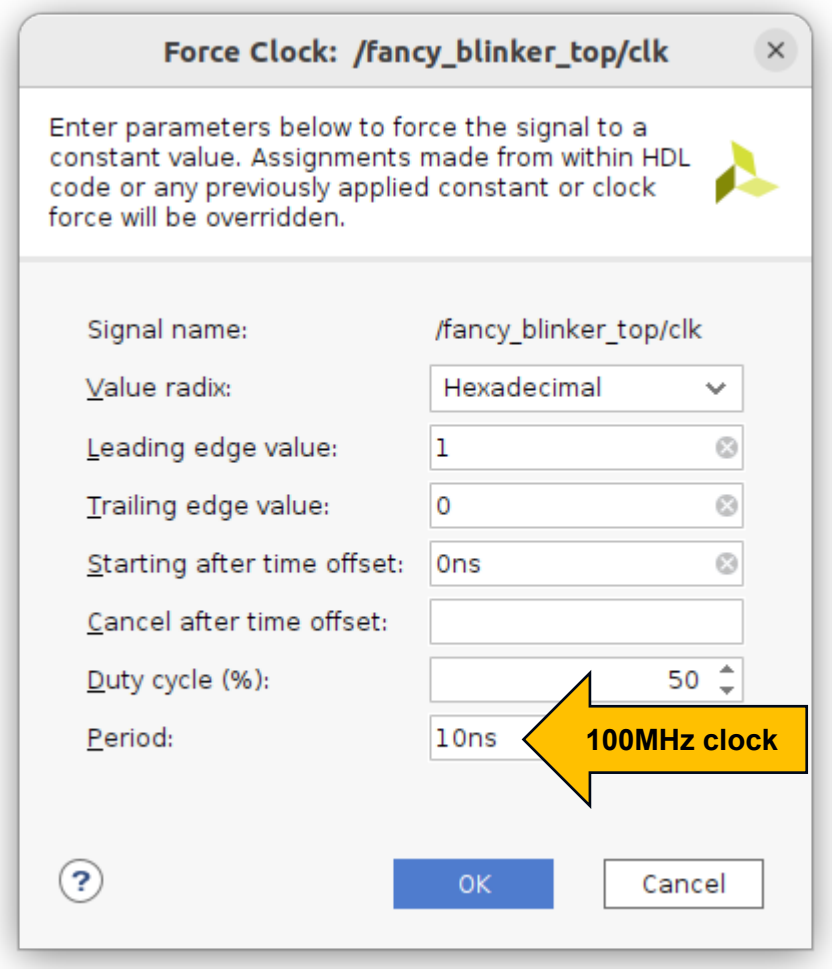
Installation Verification – LED Blinker Modules

This week we will always run using a timescale of 1ns/1ps. The following code will appear at the top of each module:

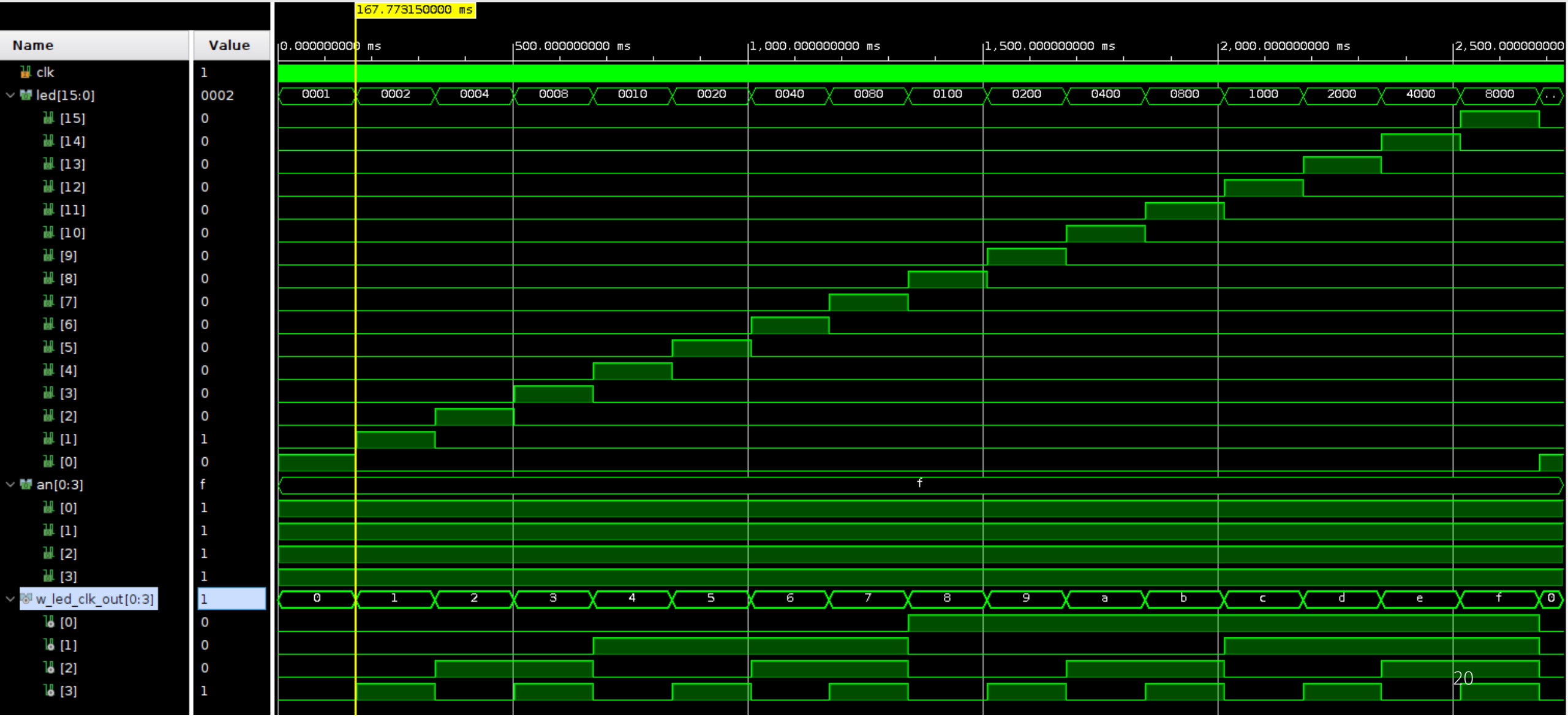
‘timescale 1ns / 1ps

This translates to a minimum clock delay of 1ns with a resolution of 0.001ns.

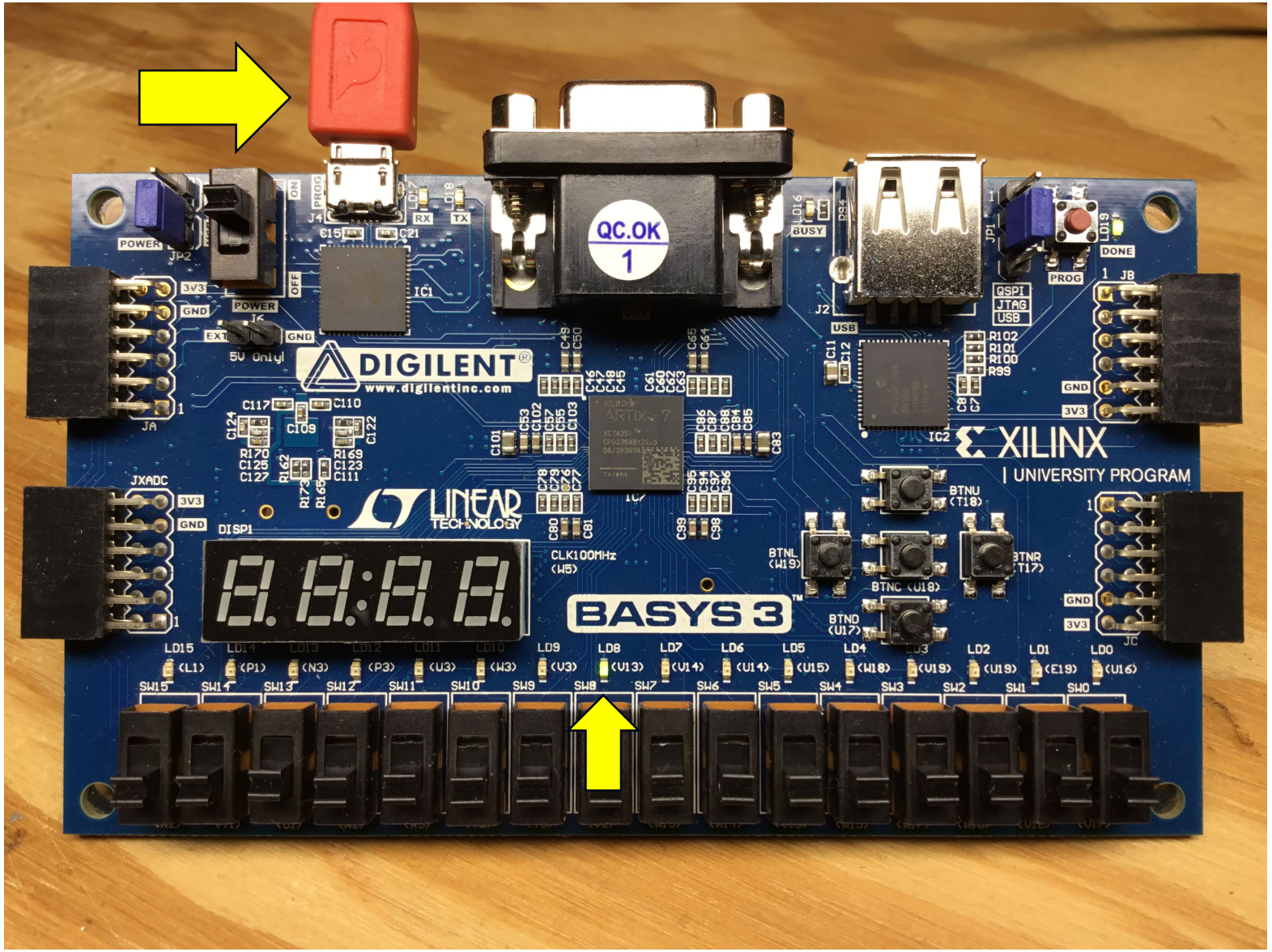
```
`timescale 1ns / 1ps
module fancy_blinker_top(
    input clk,
    output [15:0] led,
    output [0:3] an
    );

    assign an = 4'b1111;

    wire [0:3] w_led_clk_out;

    led_clk_generator led_clk_gen(
    .clk(clk),
    .led_clk_out(w_led_clk_out)
    );

    led_driver led_drvr(
    .led_clk_in(w_led_clk_out),
    .led_select(led)
    );

endmodule
```

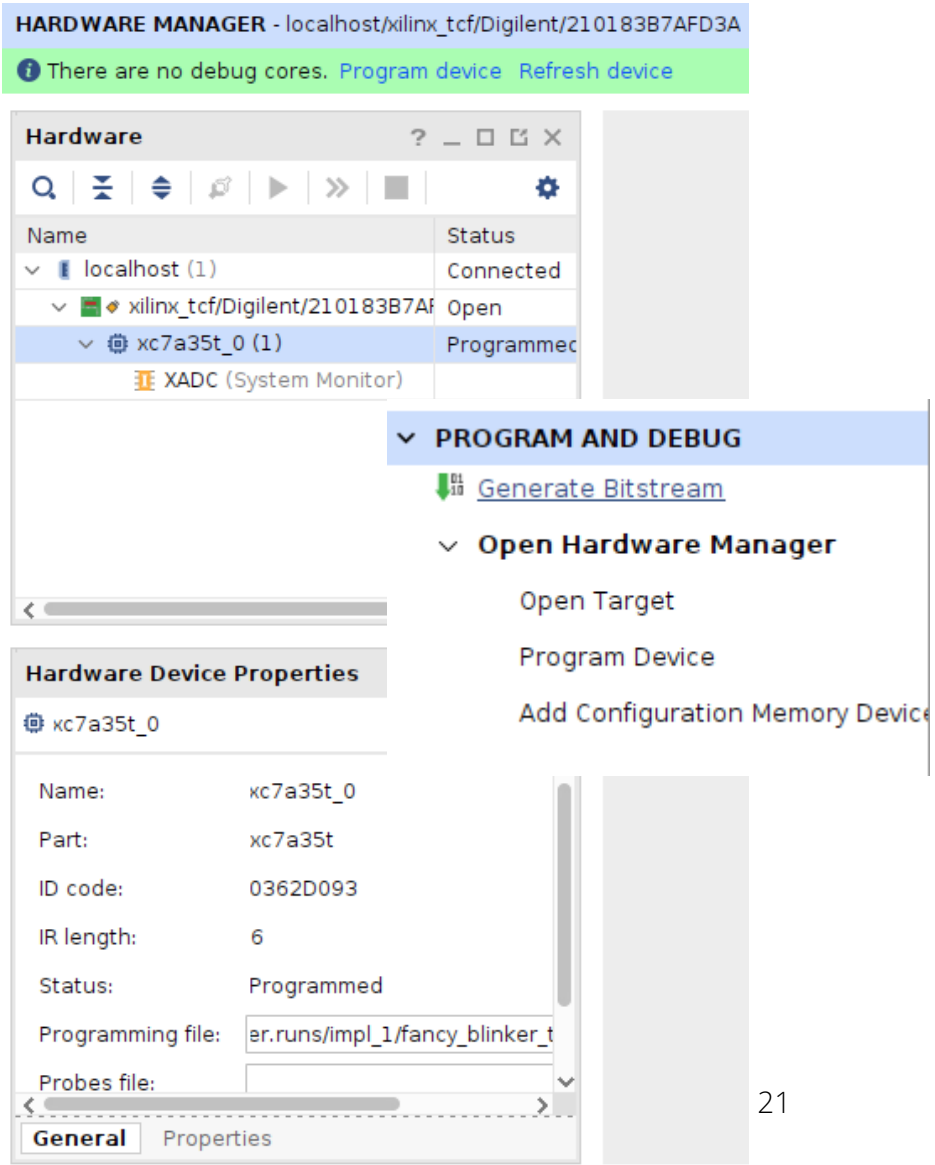


Installation Verification – LED Blinker Modules



Installation Verification – Program the Device



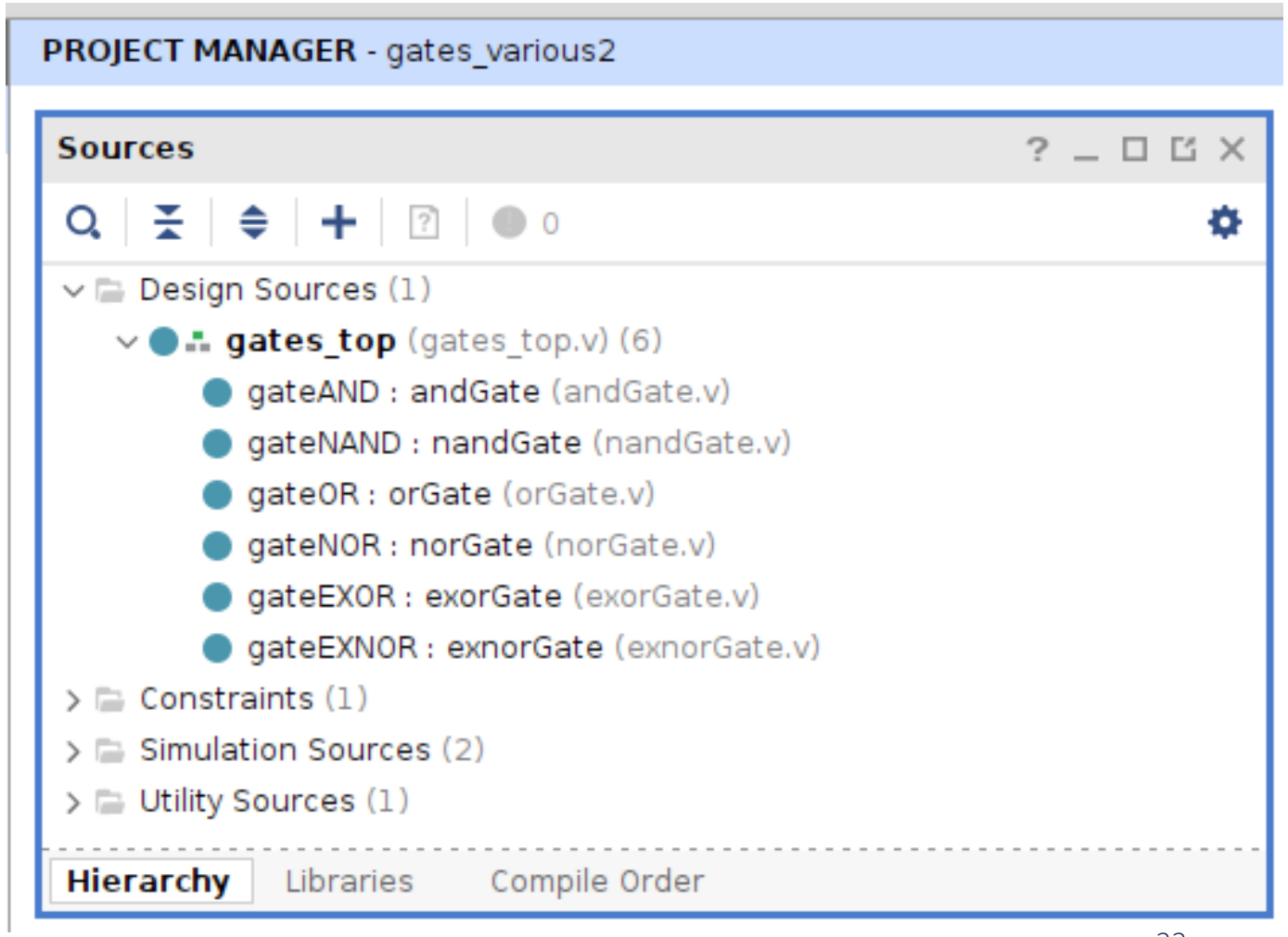


MORE TO COME..

Thank you for attending!!!

Please consider the resources below:

- **xilinx.com**



CEC Continuing Education Center

DesignNews

Thank You

Sponsored by

Digi-Key ELECTRONICS

informa markets