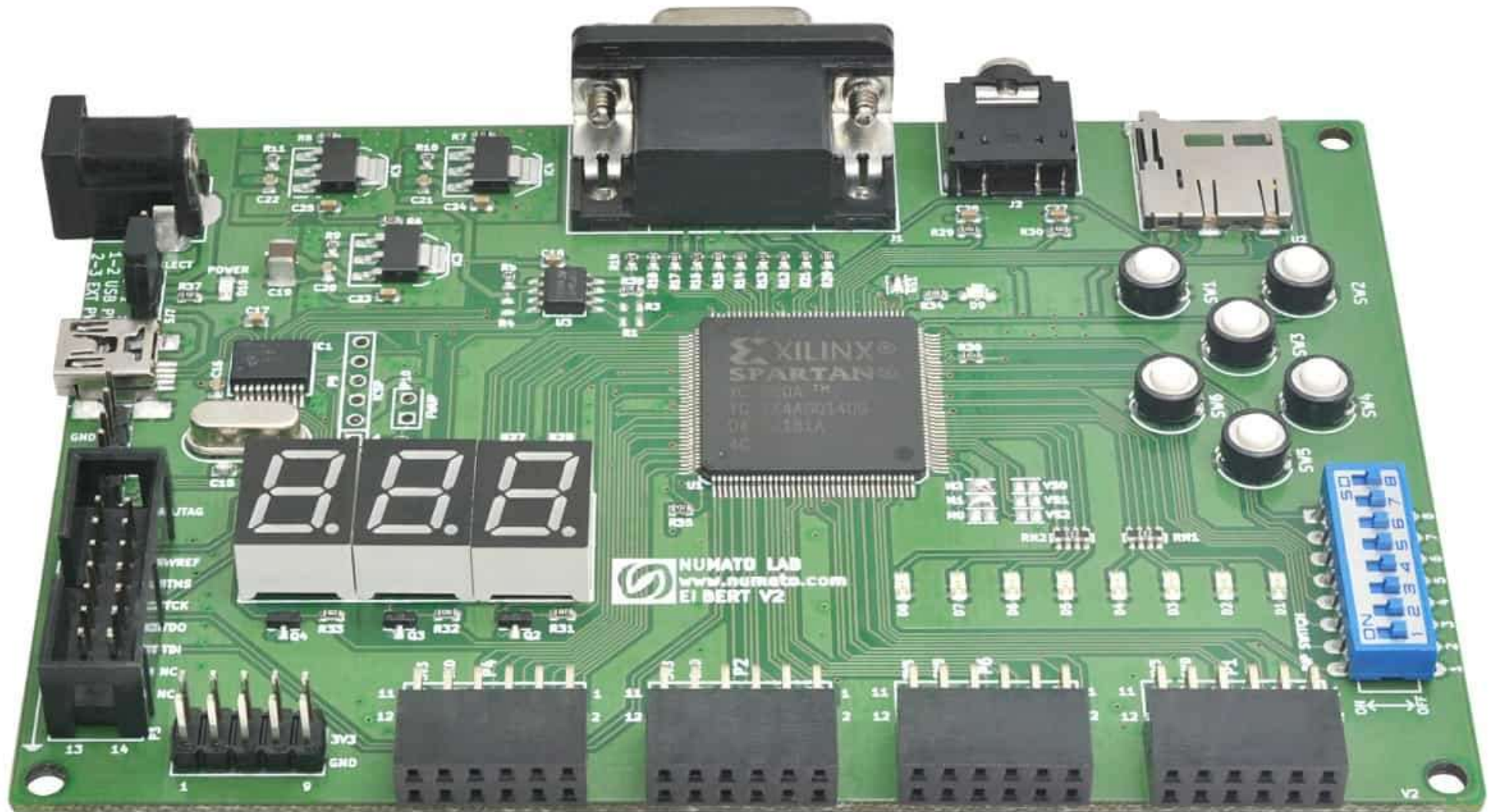


Introduction



(https://numato.com/help/wp-content/uploads/2016/03/elbertv2_4__09811.1437703972.1280.1280-e1459217096649.jpg)

Elbert V2 is an easy to use FPGA Development board featuring Xilinx Spartan-3A FPGA. Elbert V2 is specially designed for experimenting and learning system design with FPGAs. This development board features Xilinx XC3S50A TQG144 FPGA. The USB 2.0 interface provides fast and easy configuration download to the on-board SPI flash. You don't need a programmer or special downloader cable to download the bitstream to the board.

Applications

- Product Prototype Development
- Home Networking
- Signal Processing
- Wired and Wireless Communications
- Educational tool for schools and universities

Board features

- FPGA: Spartan XC3S50A in TQG144 package
- Flash memory: 16 Mb SPI flash memory (M25P16)
- USB 2.0 interface for On-board flash programming
- FPGA configuration via JTAG and USB
- 8 LEDs, Six Push Buttons and 8 way DIP switch for user-defined purposes
- One VGA Connector
- One Stereo Jack
- One Micro SD Card Adapter
- Three Seven Segment Displays
- 39 IOs for user-defined purposes
- On-board voltage regulators for single power rail operation

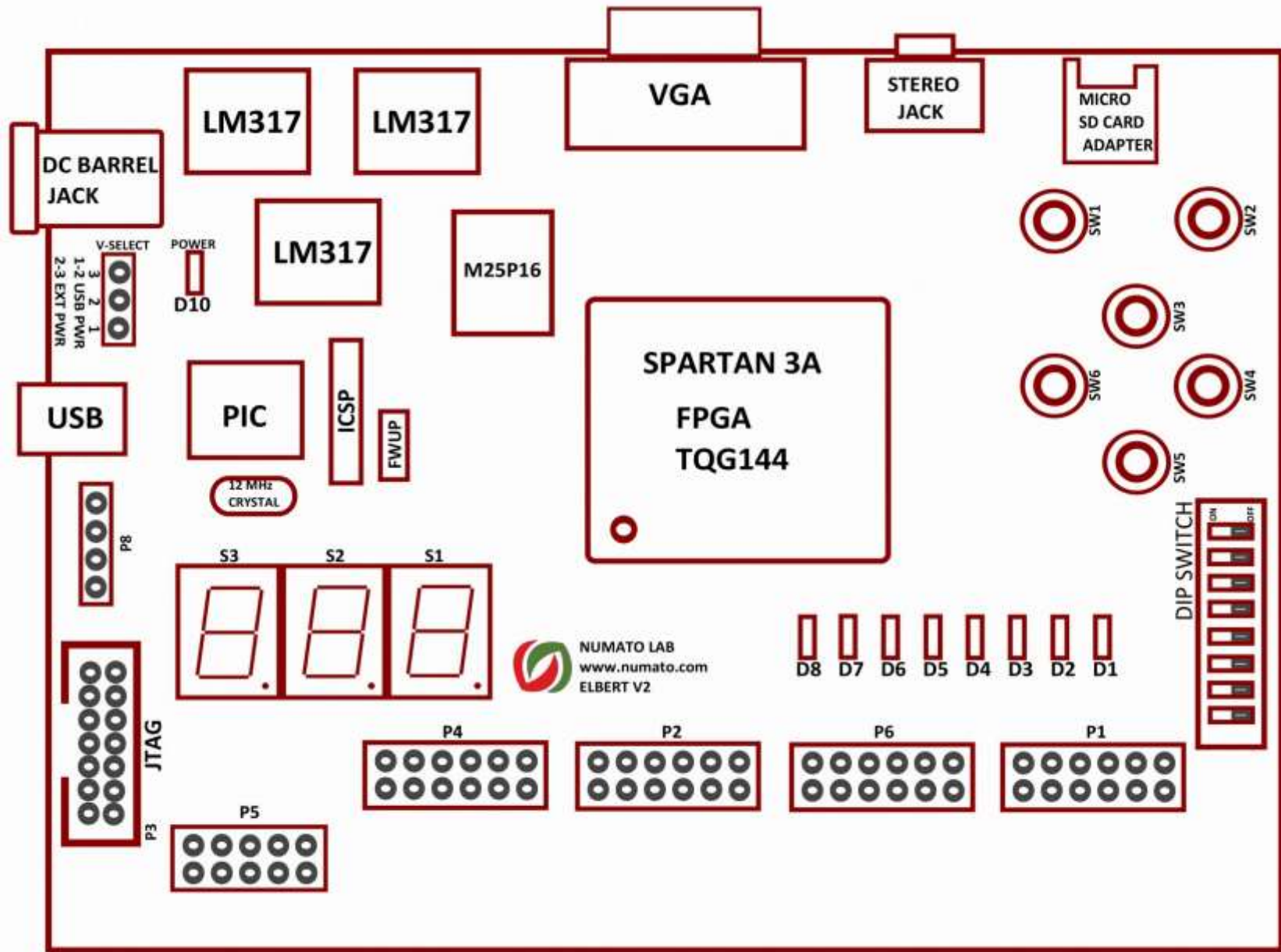
How to Use Elbert V2 Spartan 3A FPGA Development Board

Components/Tools Required

Along with the module, you may need the items in the list below for easy and fast installation.

1. USB A to Mini B cable.
2. DC Power supply (Optional).

Connection Diagram



(<https://numato.com/help/wp-content/uploads/2016/03/elbertv2-condiagram.png>)

This diagram should be used as a reference only. For detailed information, see Elbert V2 schematics at the end of this documentation. The details of individual connectors are as below.

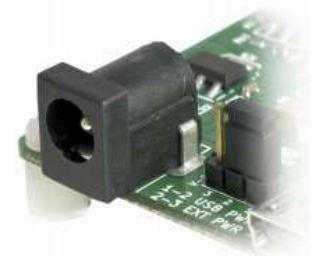
USB Interface

(<https://numato.com/help/wp-content/uploads/2016/03/elbertv2-usbinterface.png>) The onboard full-speed USB controller helps a Windows/Linux/Mac computer to communicate with this module. Use a USB A to Mini B cable to connect with a PC. By default, the module is powered from USB so make sure not to overcrowd unpowered USB hubs (the picture on the left shows USB Mini connector).



DC Power Supply

(<https://numato.com/help/wp-content/uploads/2016/03/elbertv2-dcpowersupply.png>) This module uses a +5V power supply to function properly. **By default, the board is configured to use the +5V supply from USB. So an external +5V power is not required unless the USB port is unable to supply enough current. In most cases, USB ports are capable of providing enough current for the module. The current requirement for this board largely depends on your application. Please consult the FPGA datasheet for more details on power requirements.** If for any reason, an external 5V power supply needs to be used for the module, the Power select jumper should be configured properly before connecting the power supply. Please refer to the marking on the board for more details.

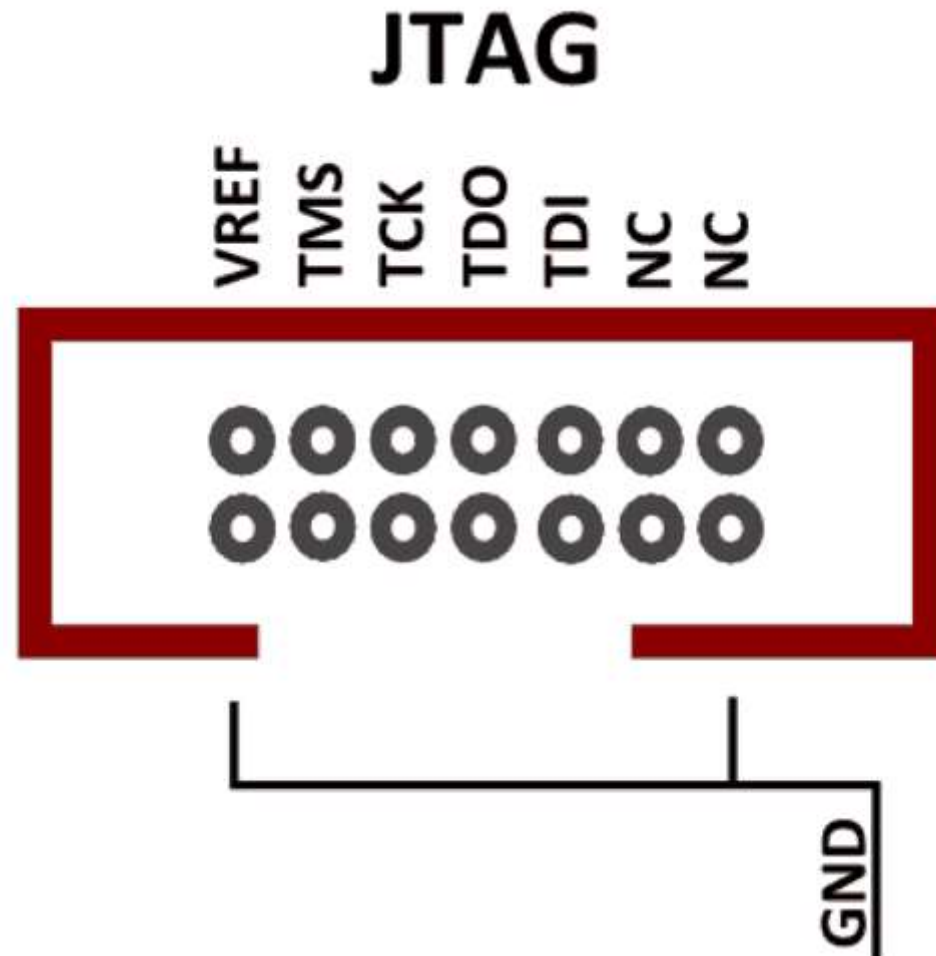


Power Select

The Solder jumper is used to configure the power source for the board. The Solder jumper in pin 1 and 2 is shorted to switch the power source to the onboard USB port and pin 2 and 3 to use the external DC power.

JTAG Connector

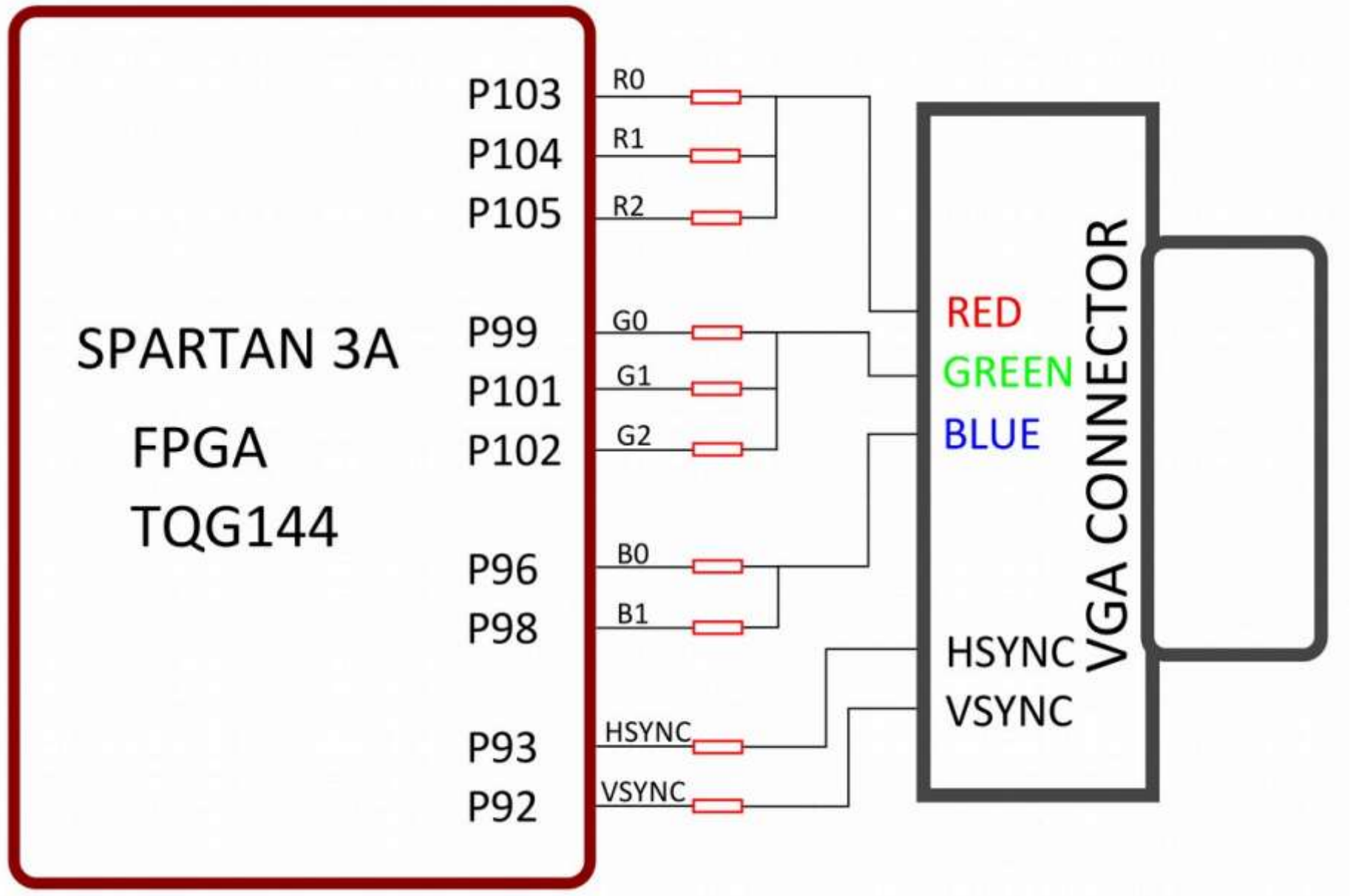
A JTAG connector provides access to FPGA's JTAG pins. A XILINX platform cable can be used for JTAG programming.



(<https://numato.com/help/wp-content/uploads/2016/03/elbertv2-jtag.png>)

VGA

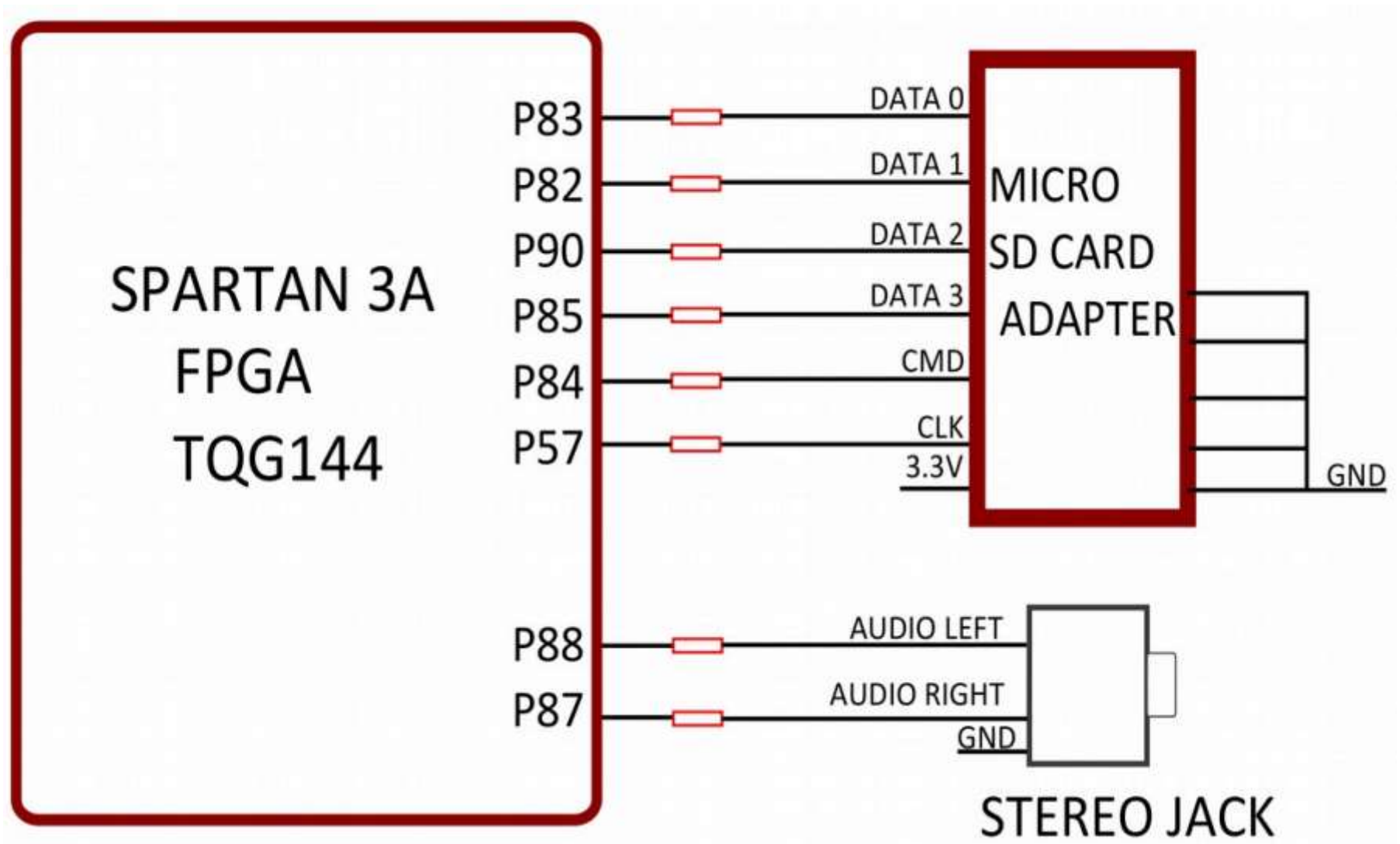
The VGA interface provides this board the ability to generate VGA signals from FPGA and display information any Display/monitor that supports standard VGA connector. This VGA interface uses resistor network-based DAC for easy use and code implementation. This 8 bit VGA interface can display up to 256 colors.



Micro SD and Audio

Elbert V2 features a Micro SD adapter on-board. By installing a Micro SD card, you can add data logging, media storage and other file storage to your design.

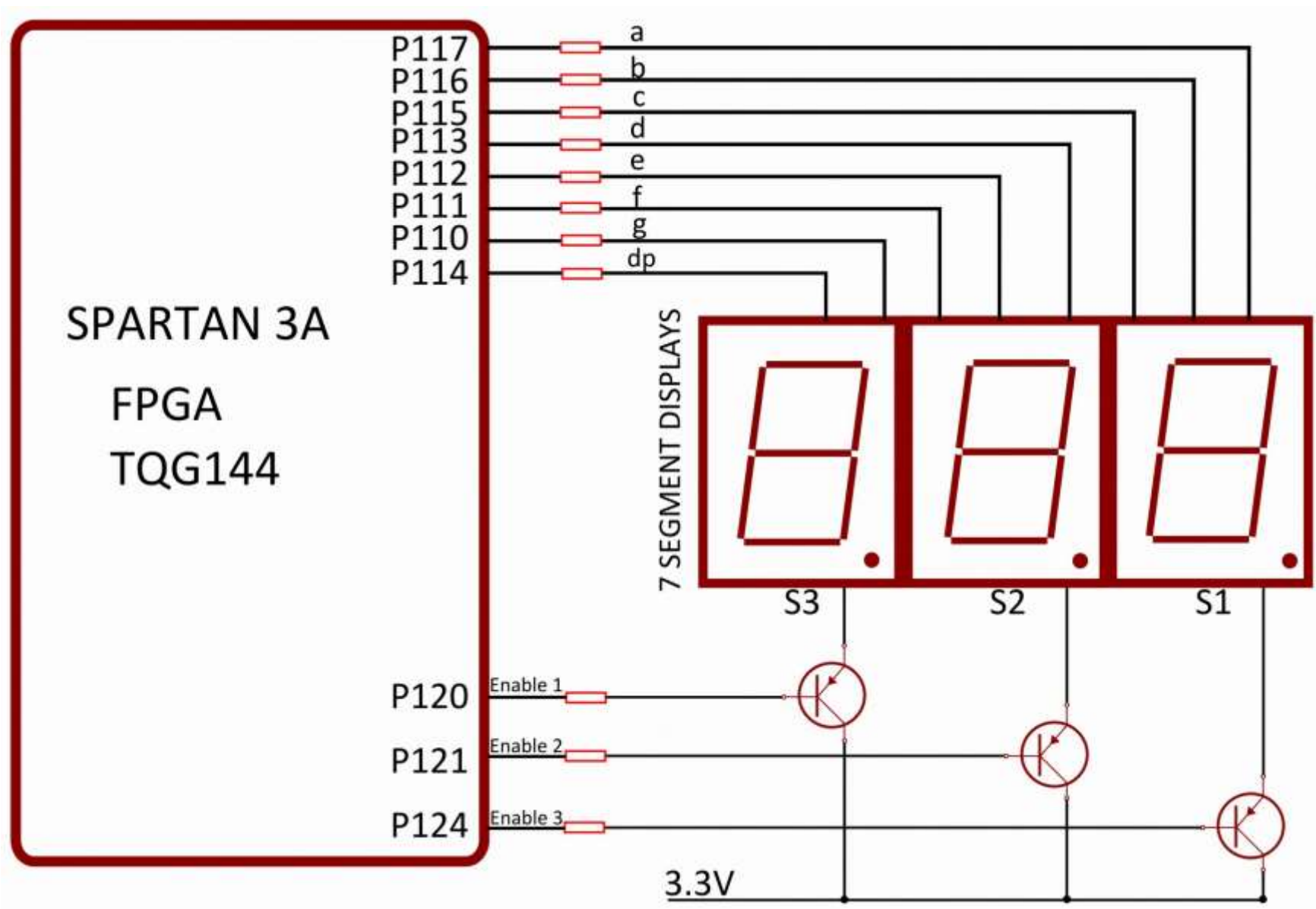
Two IOs on the FPGA is dedicated to generating two channels of audio. Different audio tones can be generated by using PWM and Frequency synthesis.



(https://numato.com/help/wp-content/uploads/2016/03/elbertv2-microsd_audio.png)

7Segment LED Display

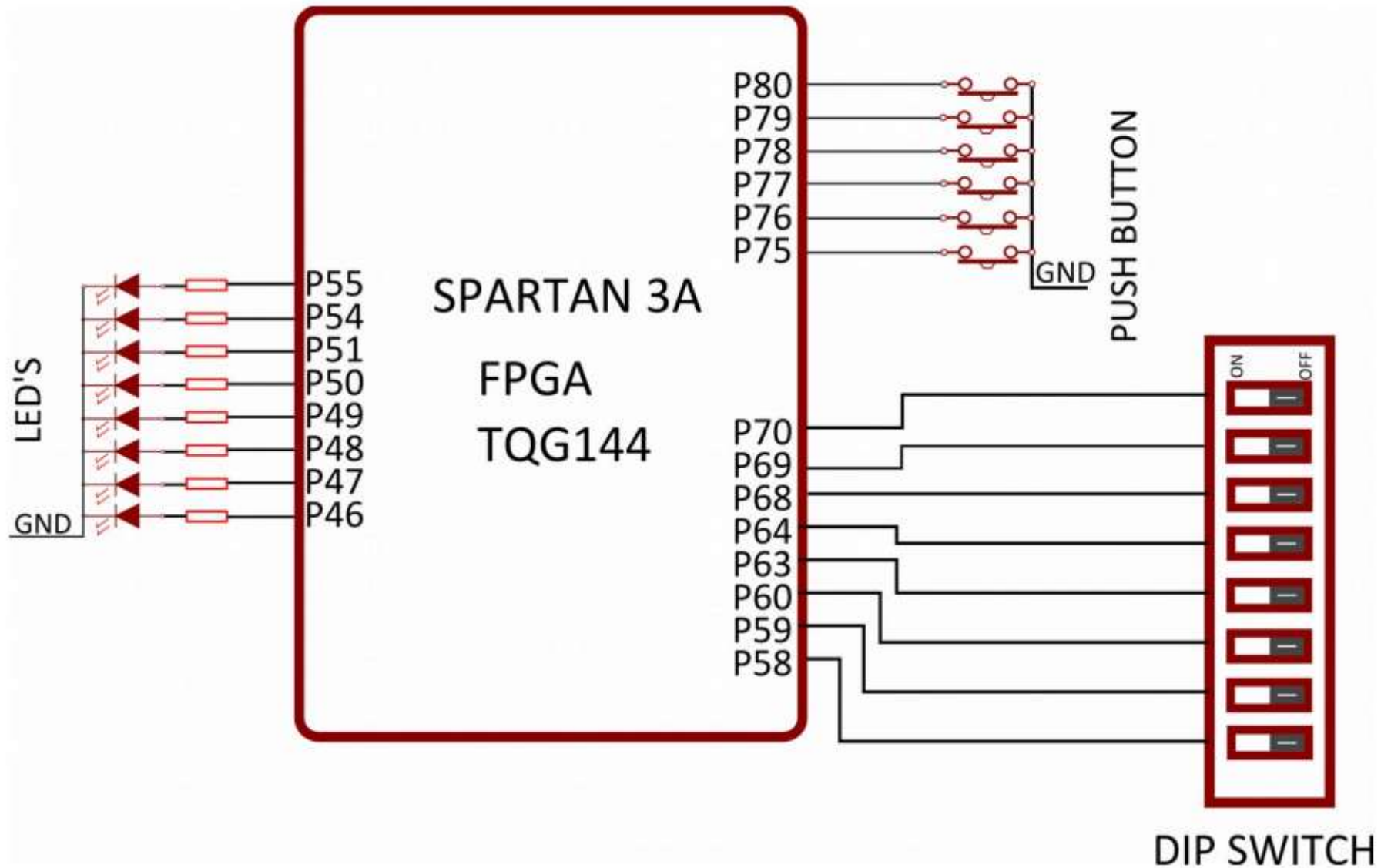
This board features three 7-segment LED display multiplexed for low pin count operation. Each module can be separately turned on and off with the three switching transistors.



(<https://numato.com/blog/wp-content/uploads/2016/03/elbertv2-7thsegmentleddisplay.png>)

LED, Push Button and Dip Switch

Elbert V2 has six push-button switches, an eight-position DIP switch and eight LEDs for human interaction. All switches are directly connected to Spartan 3 FPGA and can be used in your design with minimal effort. The switches will require weak pull up on the GPIOs to be enabled. Please see a sample UCF (User Constraints File) see how to enable pull-ups on IOs.



GPIOs

This board is equipped with 39 user IO pins that can be used for various custom applications. Pin assignments on the connectors are available in the tables below.

HEADER P1

Header Pin No.	Pin description	Spartan-3A (XC3S50A-TQG144)Pin No.
1	IO_L12P_3	31
2	IO_L12N_3	32
3	IO_L11P_3	28
4	IO_L11N_3	30
5	IO_L10P_3	27
6	IO_L10N_3	29
7	IO_L09P_3	24
8	IO_L09N_3	25
9	GND	NA
10	GND	NA
11	VCCAUX	NA
12	VCCAUX	NA

HEADER P6

Header Pin No.	Pin description	Spartan-3A (XC3S50A-TQG144)Pin No.
1	IO_L08P_3/TRDY2/LHCLK6	19
2	IO_L08N_3/LHCLK7	21

Header Pin No.	Pin description	Spartan-3A (XC3S50A-TQG144)Pin No.
3	IO_L07P_3/LHCLK4	18
4	IO_L07N_3/LHCLK5	20
5	IO_L06P_3/LHCLK2	15
6	IO_L06N_3/IRDY2/LHCLK3	16
7	IO_L05P_3/LHCLK0	12
8	IO_L05N_3/LHCLK1	13
9	GND	NA
10	GND	NA
11	VCCAUX	NA
12	VCCAUX	NA

HEADER P2

Header Pin No.	Pin description	Spartan-3A (XC3S50A-TQG144)Pin No.
1	IO_L04P_3	10
2	IO_L04N_3/VREF_3	11
3	IO_L03P_3	7
4	IO_L03N_3	8
5	IO_L02P_3	3
6	IO_L02N_3	5
7	IO_L01P_3	4

Header Pin No.	Pin description	Spartan-3A (XC3S50A-TQG144)Pin No.
8	IO_L01N_3	6
9	GND	NA
10	GND	NA
11	VCCAUX	NA
12	VCCAUX	NA

HEADER P4

Header Pin No.	Pin description	Spartan-3A (XC3S50A-TQG144)Pin No.
1	IO_L12P_0/VREF_0	141
2	IO_L12N_0/PUDC_B	143
3	IO_L11P_0	138
4	IO_L11N_0	139
5	IO_L10P_0	134
6	IO_L10N_0	135
7	IO_L09P_0/GCLK10	130
8	IO_L09N_0/GCLK11	132
9	GND	NA
10	GND	NA
11	VCCAUX	NA
12	VCCAUX	NA

HEADER P5

Header Pin No.	Pin description	Spartan-3A (XC3S50A-TQG144)Pin No.
1	IO_L07P_0/GCLK6	125
2	IP_0/VREF_0	123
3	IO_L07N_0/GCLK7	127
4	IO_L06N_0/GCLK5	126
5	IO_L08N_0/GCLK9	131
6	IO_L07P_1/IRDY1/RHCLK6	91
7	IO_0	142
8	IP_0	140
9	GND	NA
10	VCCAUX	NA

Driver Installation

Installing on Windows

(<https://numato.com/help/wp-content/uploads/2016/03/elbertv2-inswindows.png>) This product requires a driver to be installed for proper functioning when used with Windows. The driver package can be downloaded from the product page. To install the driver, unzip the contents of the downloaded driver package to a folder. Attach the USB cable to the PC and when asked by Windows

device installation wizard, point to the folder where driver files are present. When the driver installation is complete, the module should appear in Windows Device Manager as a serial port (see the picture on the right). Note down the name of the serial port (COM1, COM2, etc..). This information is required while programming the module with the configuration tool.



Installing on Linux

To use this product with Linux, the USB CDC driver needs to be compiled in with the kernel.

Fortunately, **most Linux distributions (Ubuntu, Redhat, Debian, etc.) have this driver pre-installed.** The chances of you requiring to rebuild the kernel to include the USB CDC driver is very slim. When connected to a Linux machine, this product should appear as a serial port in the /dev directory. Usually, the name of the device will be “ttyACMx” or similar. The name may be different depending on the Linux distribution you have.

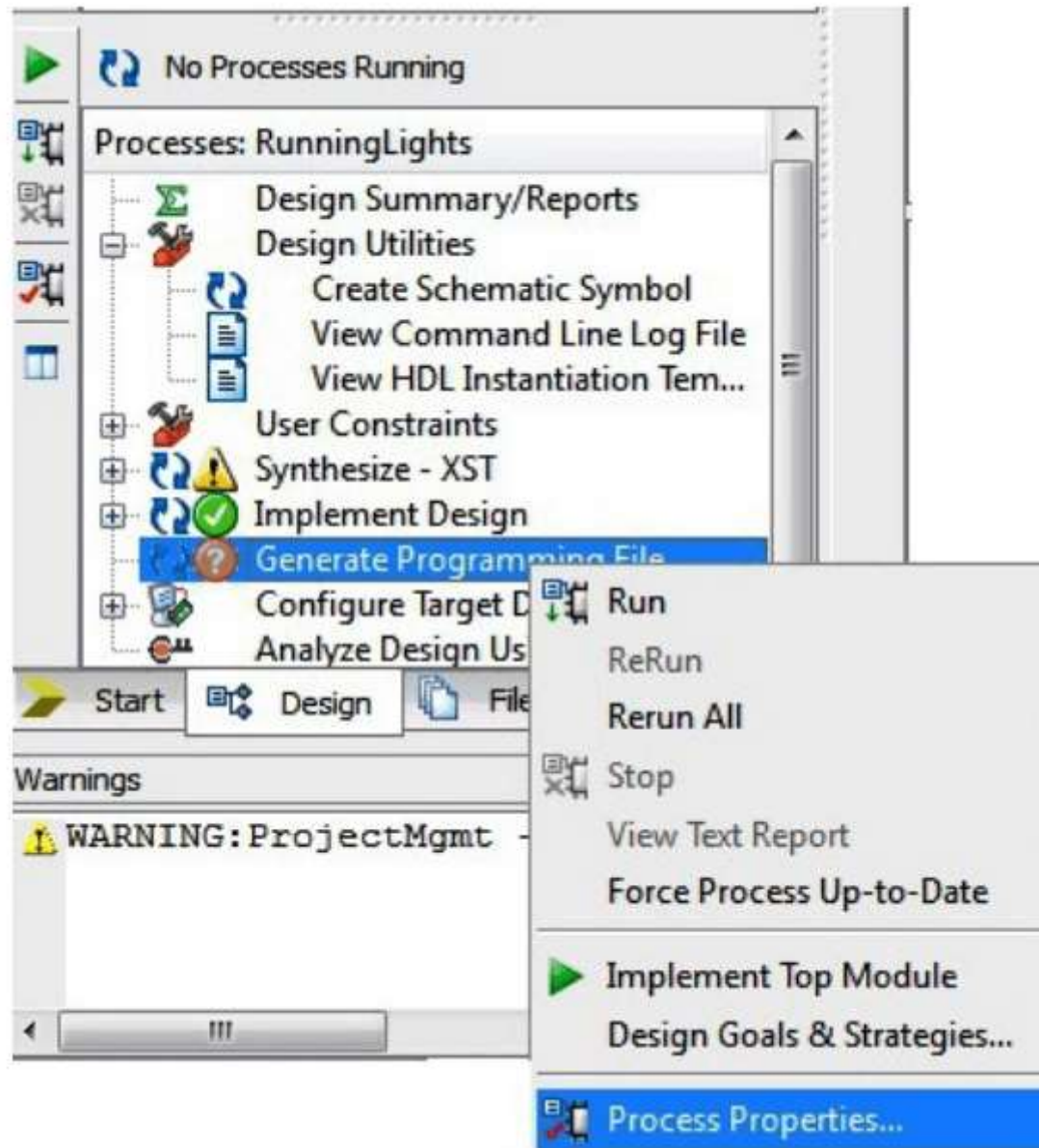
Installing on Mac

Similar to Linux, Mac operating system comes with the required drivers pre-installed. When connected to a Mac computer, the device should appear as a serial port.

Generating Bitstream for Elbert V2

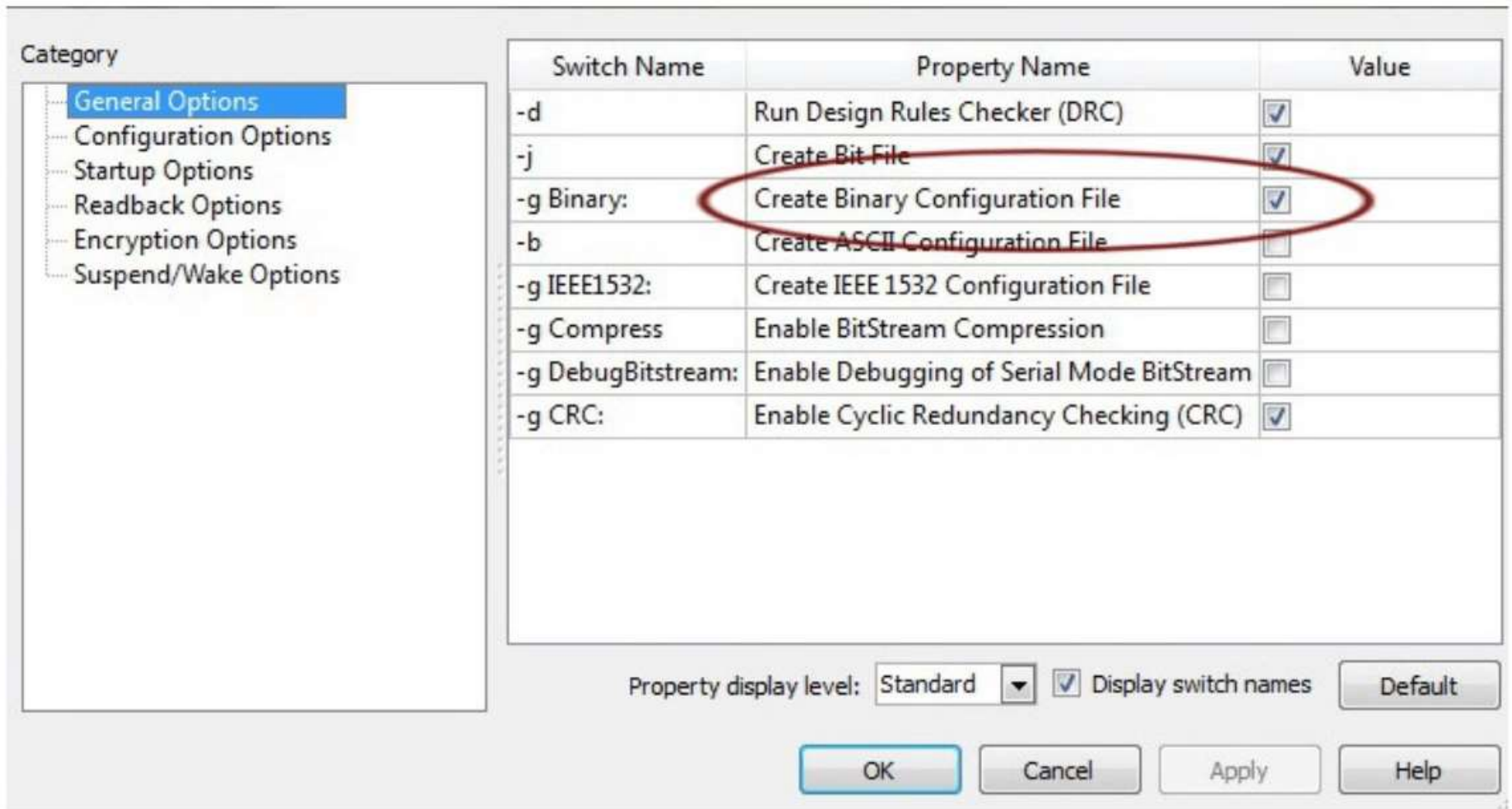
HDL design needs to be converted to bitstream before it can be programmed to FPGA. Elbert V2 at this time accepts only binary (.bin) bitstream created by Xilinx ISE (<http://www.xilinx.com/tools/webpack.htm>). Once the HDL is synthesized, it is easy to create a binary bitstream out of it. Please follow the steps below to generate binary bitstream from your design using ISE Web Pack.

Step 1: Right-click on the “Generate Programming File” option in the “Processes” window.



(<https://numato.com/help/wp-content/uploads/2016/03/elbertv2-bitstreamstep1.png>)

Step 2: Select “Process Properties” from the pop-up menu. In the dialog box, check the “Create Binary Configuration File” Checkbox and click “Apply”.



(<https://numato.com/help/wp-content/uploads/2016/03/elbertv2-bitstreamstep2.png>)

Step 3: Click “OK” to close the dialog box. Right-click on the “Generate Programming File” option again and select “Run”. Now you will be able to find a .bin file in the project directory and that file can be used for Elbert V2 configuration.

Powering Up Elbert V2

Elbert V2 can be powered directly from the USB port so make sure that you are using a USB port that can power the board properly. It is recommended to connect the board directly to the PC instead of using a hub. It is practically very difficult to estimate the power consumption of the board, as it depends heavily on your design and the clock used. Xilinx provides tools to estimate power consumption. In any case, if power from USB is not enough for your application, an external supply can be applied to the board. Elbert V2 requires two different voltages, a 3.3V, and a 1.2V supply. On-board regulators derive these voltages from the USB/Ext power supply.

Configuring Elbert V2

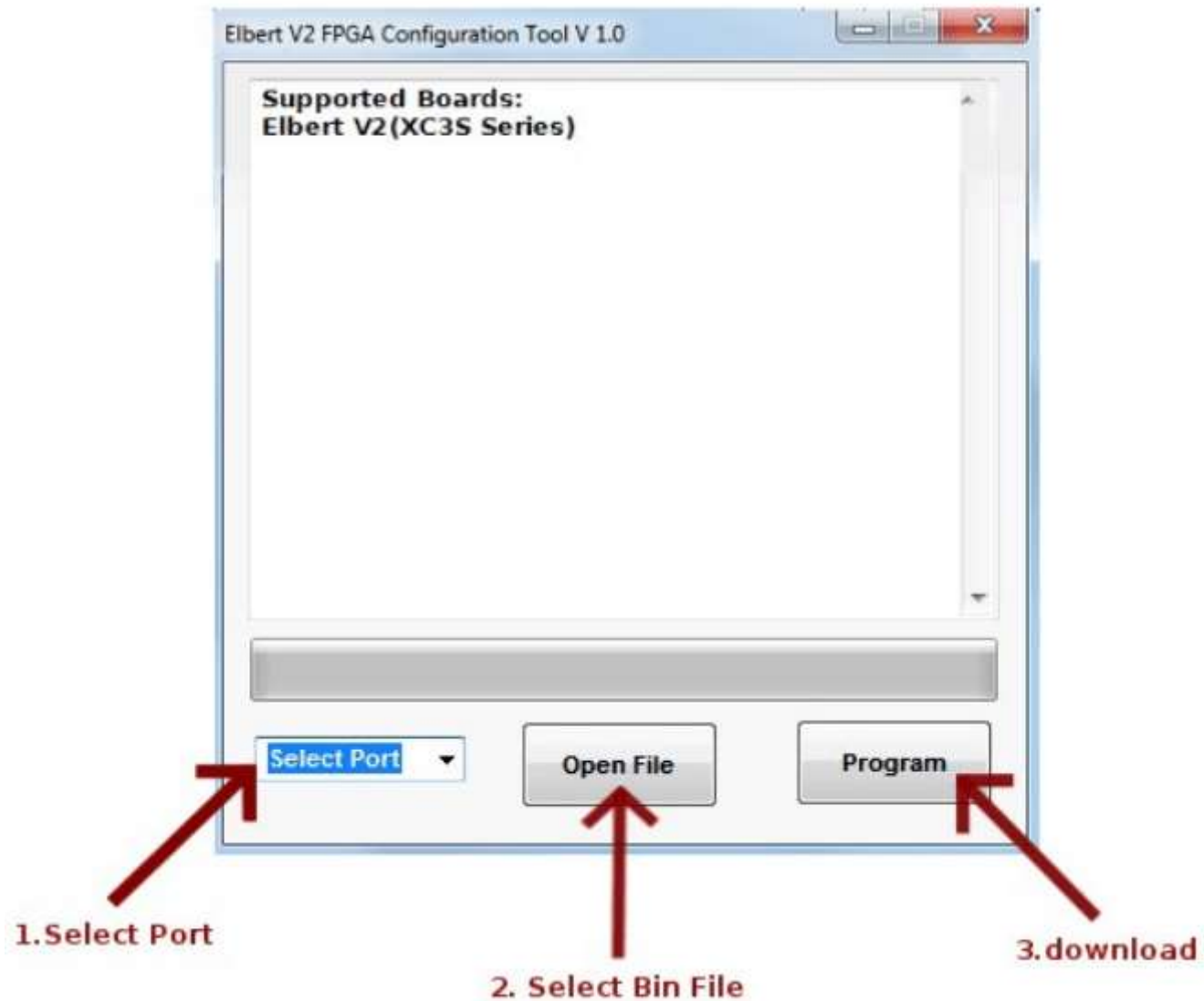
The Elbert V2 Spartan6 module can be configured by two methods,

- a) Using the Elbert V2 configuration tool through USB.
- b) Using the Xilinx programming cable.

Configuring Elbert V2 Using Configuration Tool

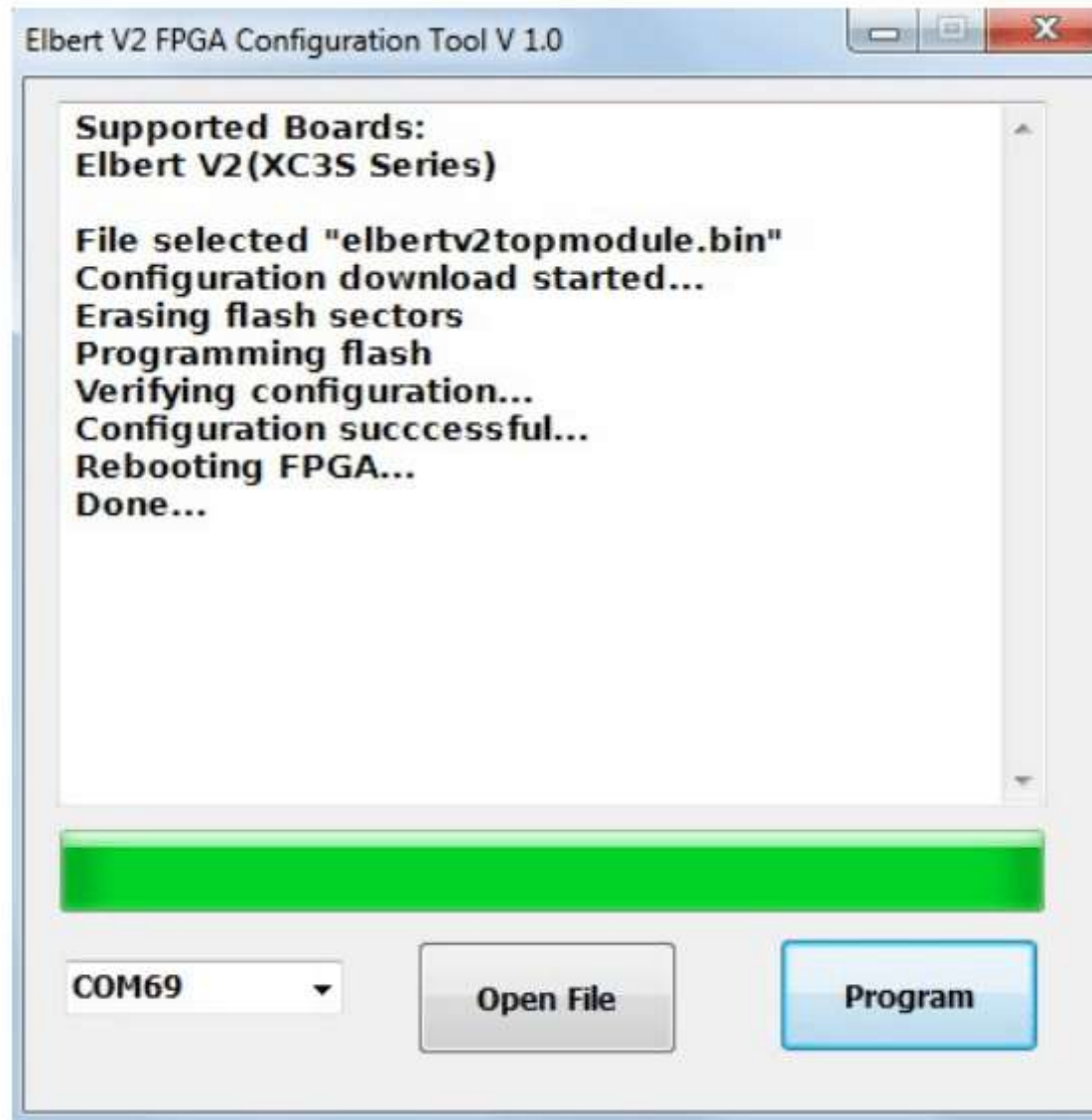
Elbert V2 has an on-board micro-controller which facilitates easy reprogramming of onboard SPI flash through the USB interface. The micro-controller receives bitstream from the host application and program it into the SPI Flash and lets the FPGA boot from the flash. The Elbert V2 configuration application can be downloaded from www.numato.com (<http://www.numato.com>) for free. When Elbert V2 is connected to PC, it shows up as a COM port in Device Manager. Run configuration application, select proper COM Port before downloading bitstream. Click on “Open File” to select the bitstream file (.bin) and press the “Program” button to download the bitstream. Wait till the download process is finished. Once the download process is over, the configuration controller will try to boot the FPGA from the SPI Flash automatically. Follow the below steps.

Step 1: Open Elbert V2 Configuration Tool. Select the port no. (Refer “Driver installation” for more information on finding port no.) Click Open file and select the .bin file.



(<https://numato.com/help/wp-content/uploads/2016/03/elbertv2-configtoolstep1.png>)

Step 2: Click on the “Program” button. Wait till “Done” appears on the screen.



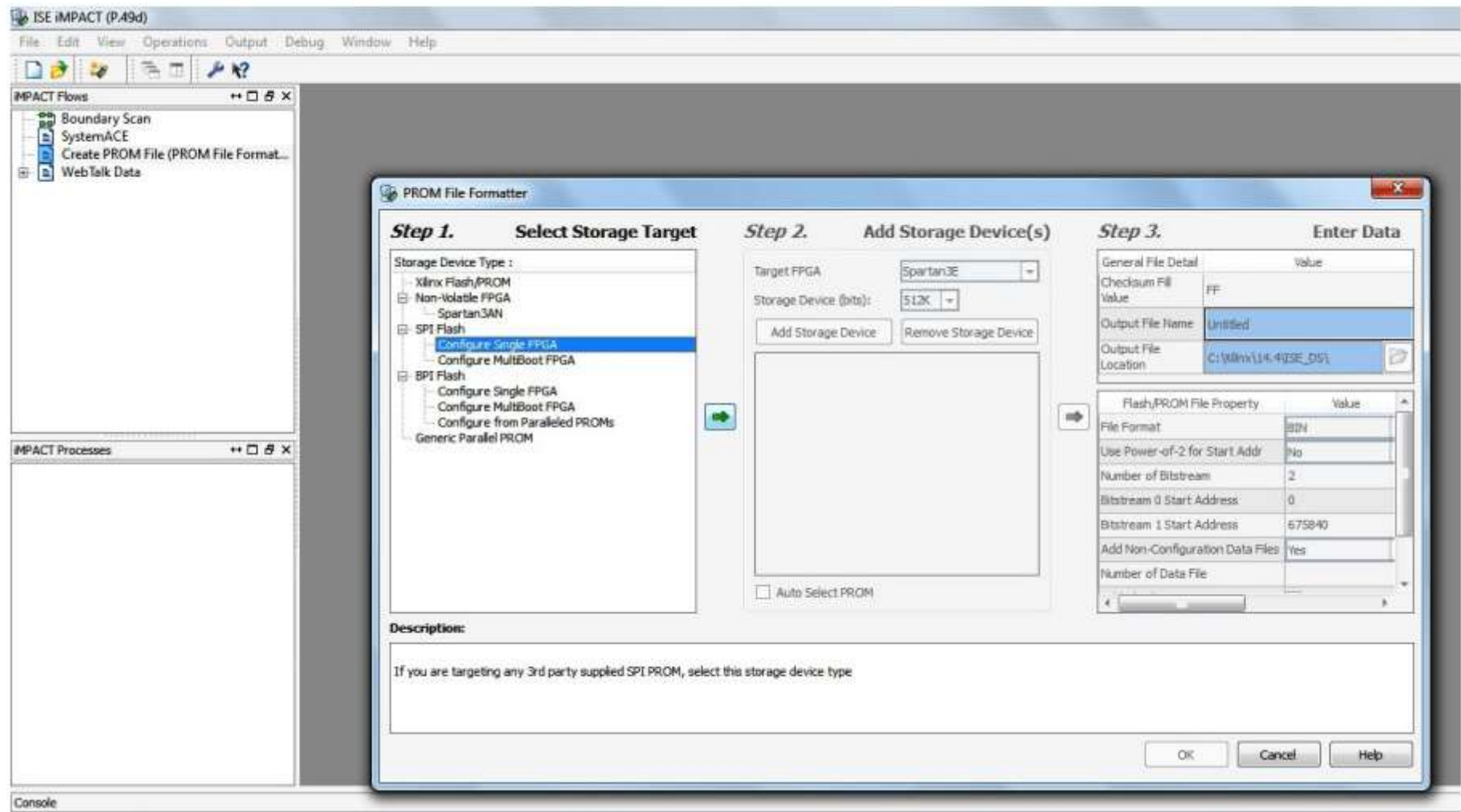
(<https://numato.com/help/wp-content/uploads/2016/03/elbertv2-configtoolstep2.png>)

Configuring Elbert V2 Using JTAG

Elbert V2 Spartan6 module features an onboard JTAG connector that facilitates easy reprogramming of SRAM and onboard SPI flash through JTAG programmer like “Xilinx Platform-cable usb”. Programming Elbert V2 using JTAG requires “Xilinx ISE iMPACT” software which is bundled with Xilinx ISE Design Suite. To program the SPI flash we need a “.mcs” file that needs to be generated from the “.bit” file. Steps for generating the “.mcs” file are discussed below. Programming FPGA SRAM does not require a “.mcs” file to be generated.

Generating “.mcs” file for Elbert V2

Step 1: Open ISE iMPACT. Click on “Create PROM file(PROM file formatter)”.In the dialog box, select “Configure Single FPGA” in the storage device type. Then click on the green arrow on the right side.



(<https://numato.com/help/wp-content/uploads/2016/03/elbertv2-jtagstep1.png>)

Step 2: Select the 16M in Storage Device (bits) list. Now click on “Add Storage Device”, then the green arrow at the right side.

Step 2. Add Storage Device(s)

Storage Device (bits)

Add Storage Device

Storage Device


16M
128K
256K
512K
1M
2M
4M
8M
16M
32M
64M
128M
256M


☐ Auto Select PROM



(<https://numato.com/help/wp-content/uploads/2016/03/elbertv2-jtagstep2.png>)

Step 3: Set an output file name and an output file location (the “.mcs” file will be generated at this location which will be required later for programming the FPGA), then click OK twice, then select the “.bit” file we already generated then click Open and click NO when it prompts to add another device file.

Step 3.
Enter Data

General File Detail		Value
Checksum Fill Value	FF	
Output File Name	ElbertV2	
Output File Location	C:\Xilinx\14.4\ISE_DS\	



Flash/PROM File Property		Value
File Format	MCS	
Add Non-Configuration Data Files	No	

d mode.

ed to calculate the checksum of the unused portions.

le format your PROM programmer uses, you output a MCS

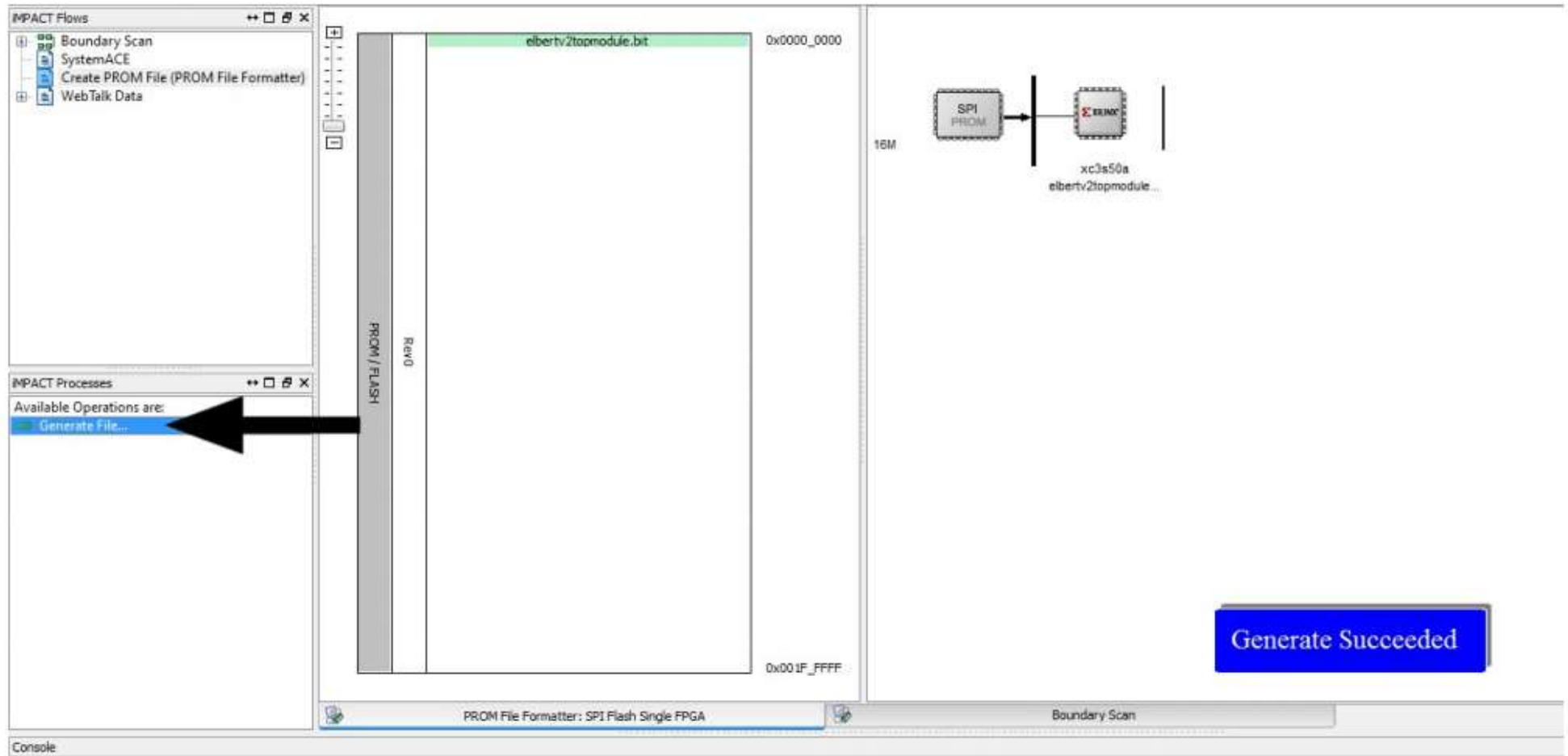
OK

Cancel

Help

(<https://numato.com/help/wp-content/uploads/2016/03/elbertv2-jtagstep3.png>)

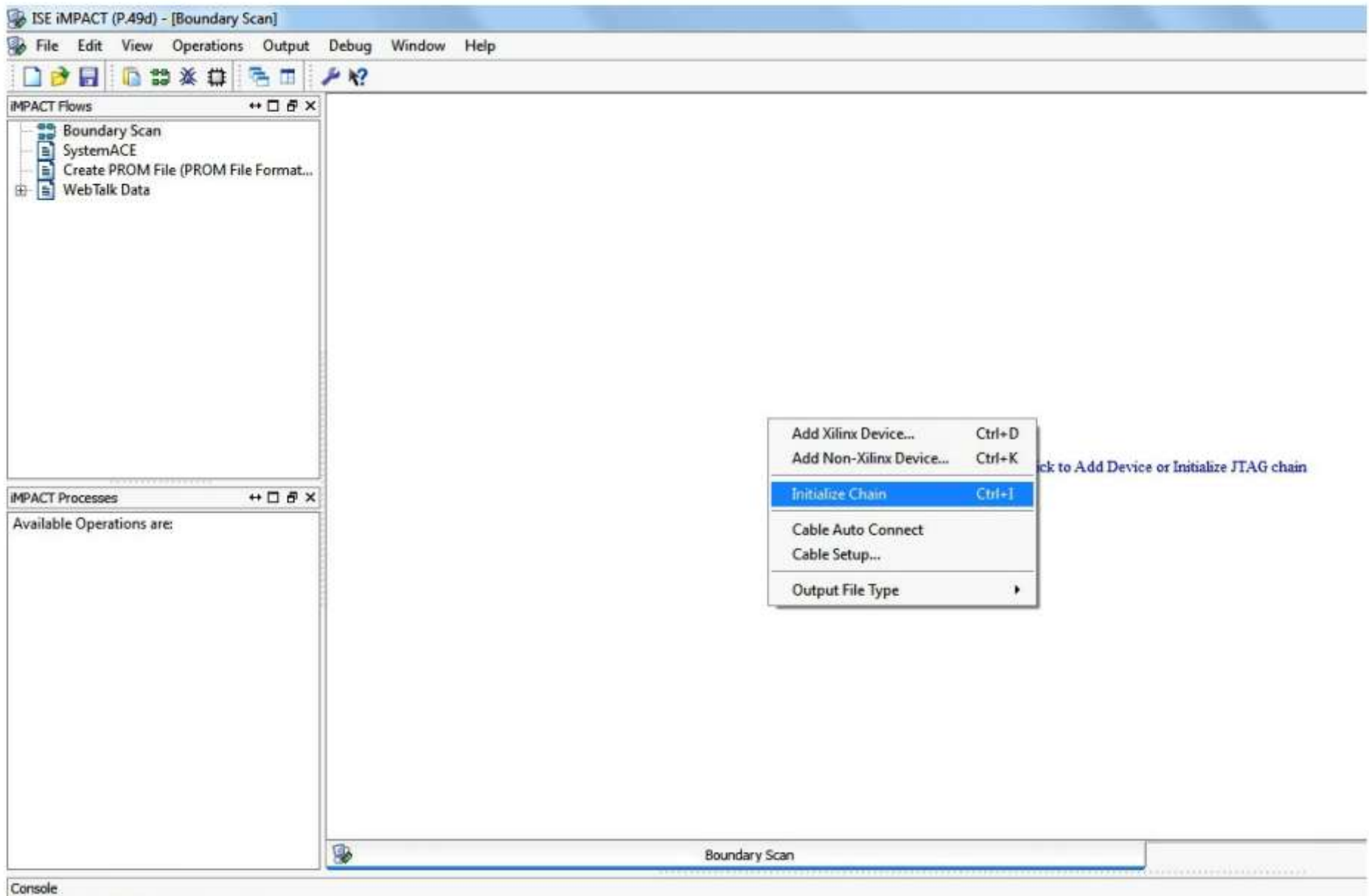
Step 4: Double click on “Generate File”. “Generate Succeeded” will be displayed as shown in fig below if the .mcs file is generated successfully.



(<https://numato.com/help/wp-content/uploads/2016/03/elbertv2-jtagstep4.png>)

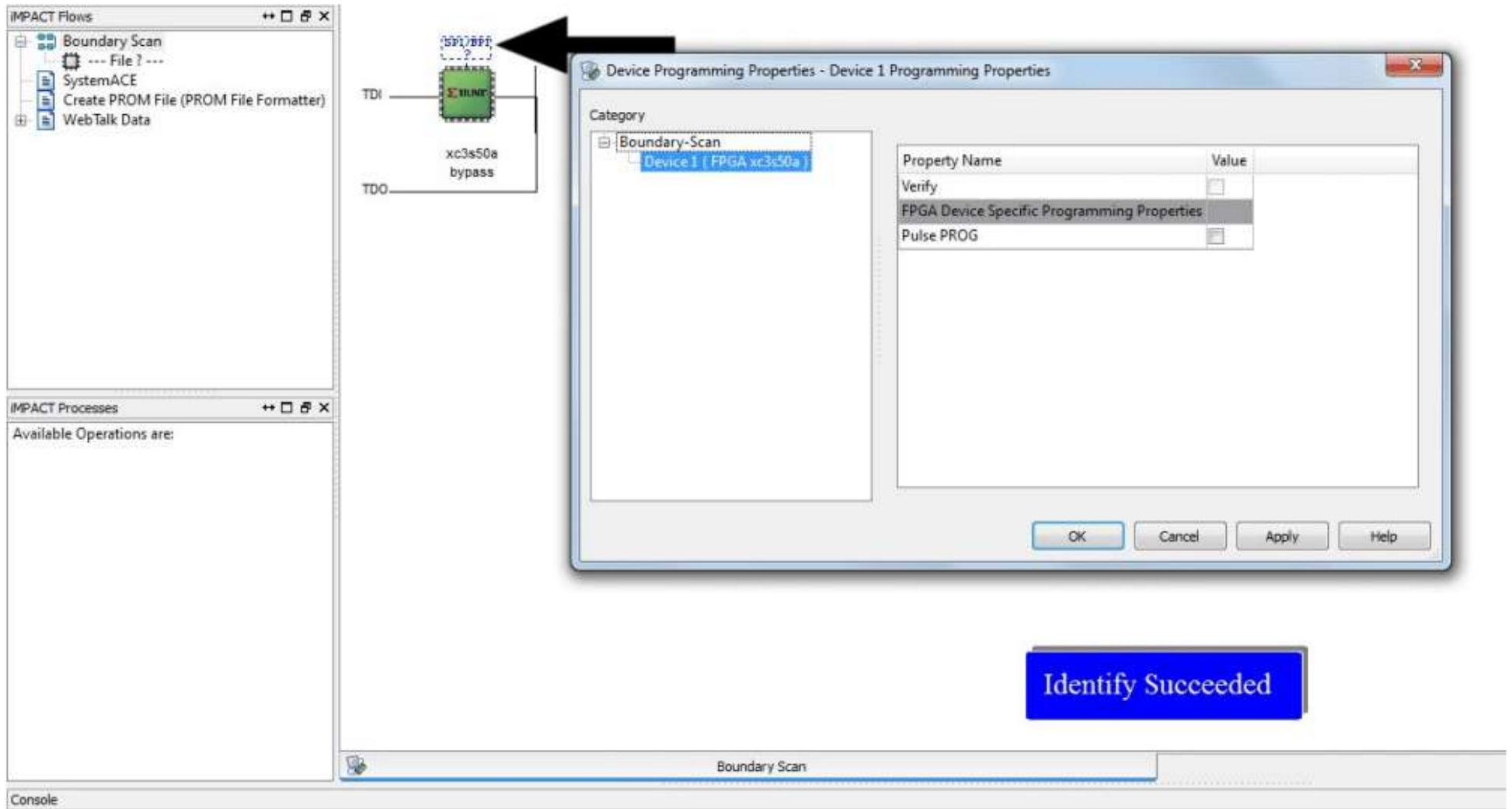
Programming FPGA Using ISE iMPACT

Step 1: Open ISE iMPACT. Click on “Boundary Scan” in the iMPACT flows window in the left top corner. Then right-click on the window panel on the right side. Select “Initialize Chain”.



(<https://numato.com/help/wp-content/uploads/2016/03/elbertv2-impactstep1.png>)

Step 2: If the device is detected properly you will get a pop-up window as shown below, Click OK. Then right-click on the SPI/BPI (next to the black arrow in the below fig.), select Add SPI/BPI Flash.



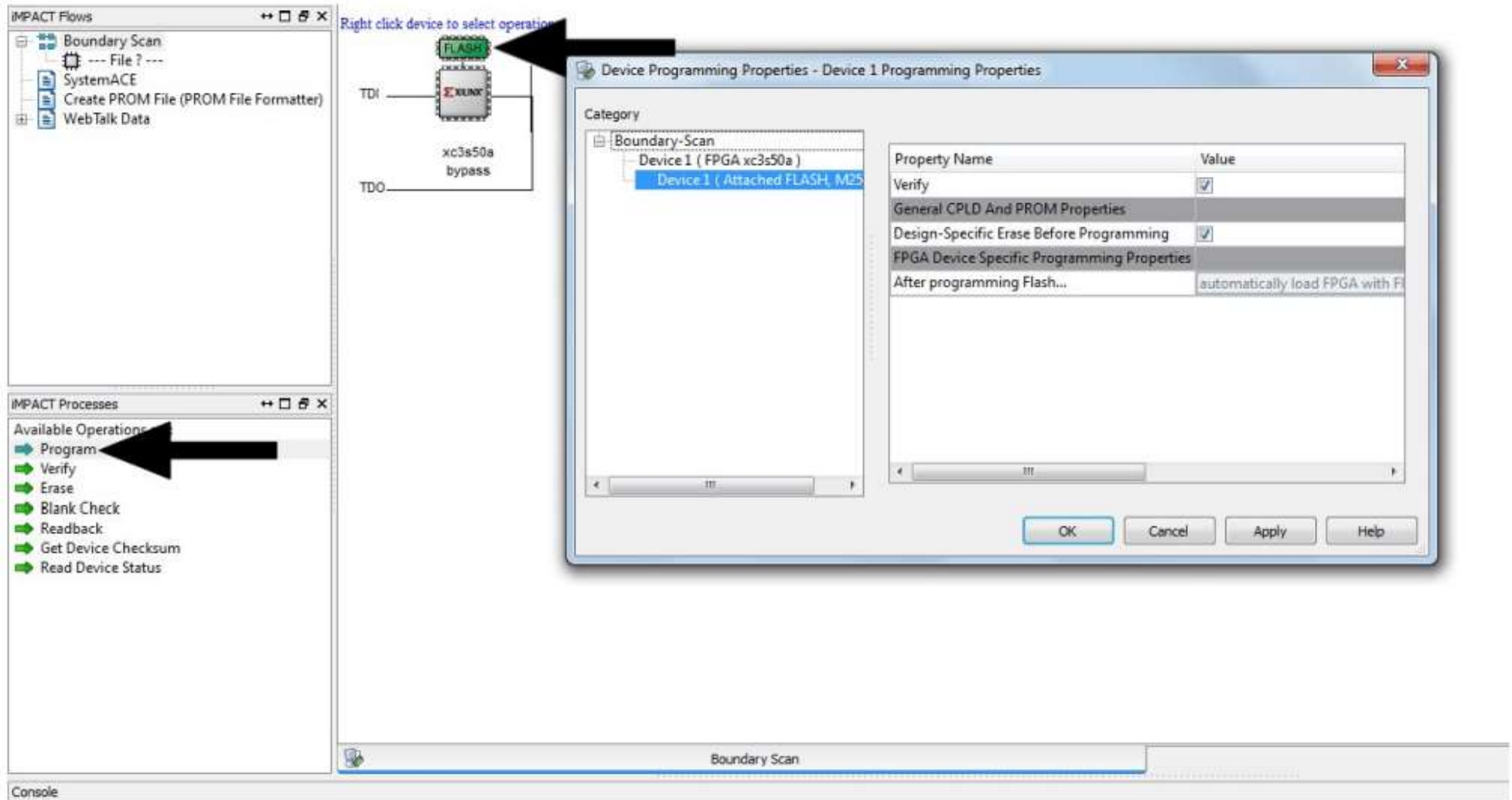
(<https://numato.com/help/wp-content/uploads/2016/03/elbertv2-impactstep2.png>)

Step 3: Select the “.mcs” file we already created and click OK. Now choose “M25P16” in the dialogue box appeared, then click OK.



(<https://numato.com/help/wp-content/uploads/2016/03/elbertv2-impactstep3.png>)

Step 4: Click on “Flash”, Double Click on Program, select OK. If the programming is successful, a confirmation message will be displayed.



(<https://numato.com/help/wp-content/uploads/2016/03/elbertv2-impactstep4.png>)

Technical Specifications

Parameter *

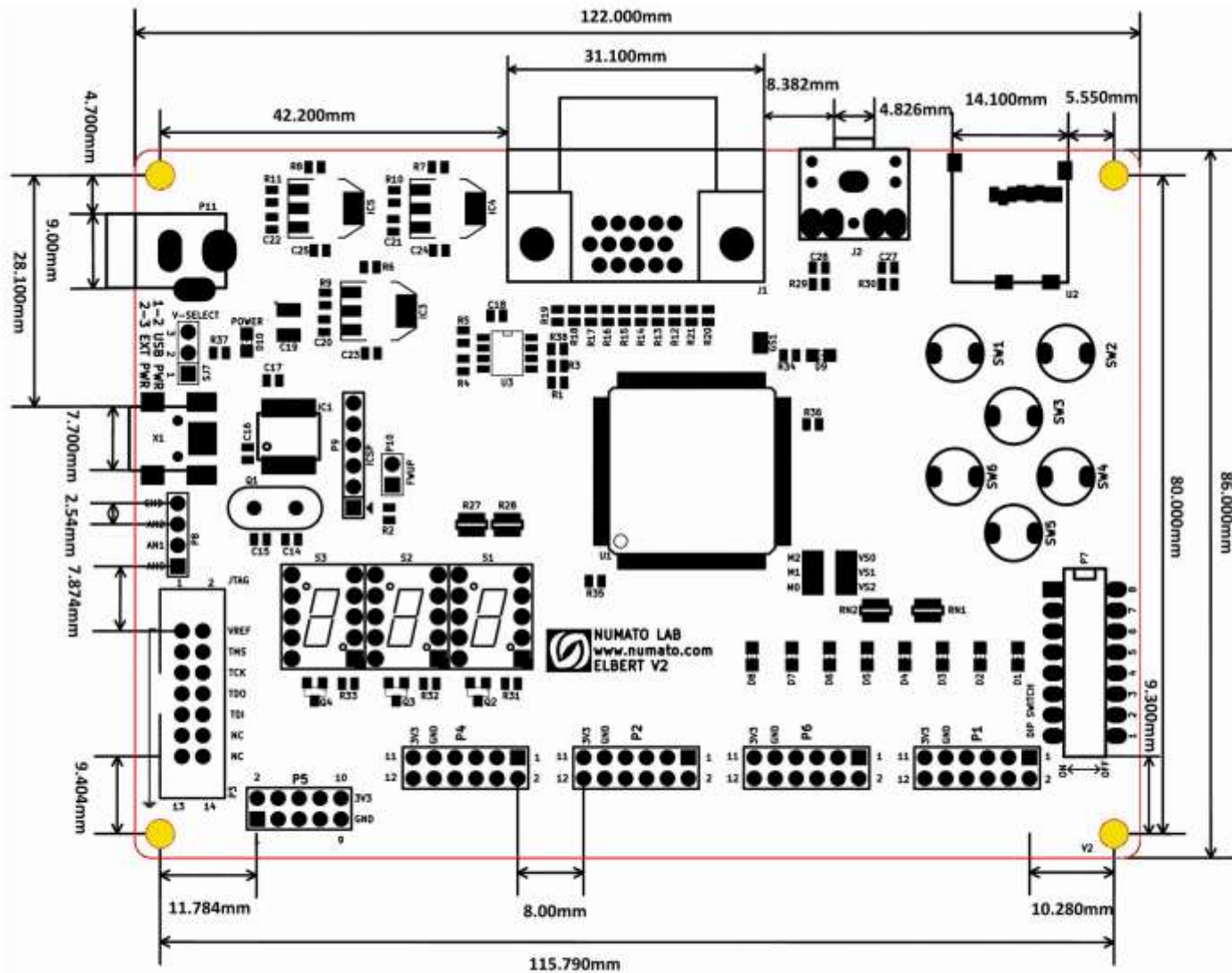
Value

Unit

Basic Specifications

Parameter *	Value	Unit
Number of GPIOs	39	
Number of LEDs	8	
Number of Push Buttons	6	
SPI Flash Memory (M25P16)	16	Mb
Power supply voltage (USB or external)	5 - 7	V
FPGA Specifications		
Internal supply voltage relative to GND	−0.5 to 1.25	V
Auxiliary supply voltage relative to GND	−0.5 to 3.75	V
Output drivers supply voltage relative to GND	−0.5 to 3.75	V

Mechanical Dimensions



L x W x H : 122.00 mm x 86.00 mm x 17 mm
Mechanical Hole Diameter : 3.2 mm

(<https://numato.com/help/wp-content/uploads/2016/03/elbertv2-dimensions.png>)

Schematics

Elbert V2 Schematics (<https://numato.com/blog/wp-content/uploads/2016/08/ElbertV2Sch.pdf>)