

SPARTAN 6 TRAINER KIT

(Model No : VPTB -20)

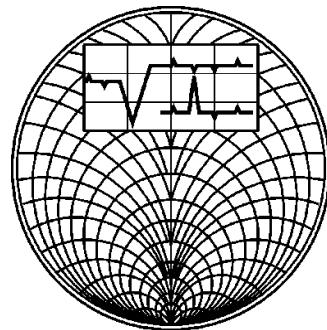
User Manual

Version 1.0

Technical Clarification /Suggestion :

✉ / ☎

**Technical Support Division,
Vi Microsystems Pvt. Ltd.,
Plot No :75, Electronics Estate,
Perungudi, Chennai - 600 096, INDIA.
Ph: 91- 44-2496 1842, 91-44-2496 1852
Mail : rnd@vimicrosystems.com
Web : www.vimicrosystem.com
01/06/08/18**



CONTENTS

| | | |
|------------------|---|-----------|
| Chapter-1 | INTRODUCTION | 1 |
| Chapter-2 | CLOCK SOURCE | 3 |
| Chapter-3 | SWITCHES & LEDS | 6 |
| Chapter-4 | RELAY & BUZZER | 10 |
| Chapter-5 | STEPPER MOTOR & DC MOTOR CONTROL | 11 |
| Chapter-6 | LCD DISPLAY | 12 |
| Chapter-7 | PROCEDURE TO WORK IN XILINX SOFTWARE | 23 |

CHAPTER-1

INTRODUCTION

Introduction

Xilinx Spartan-6 Trainer Kit is a demonstration platform intended for you to become familiar with the new features and availability of the Spartan-6 FPGA family. This Kit provides a low-cost, easy to use development and evaluation platform for Spartan-6 FPGA designs.

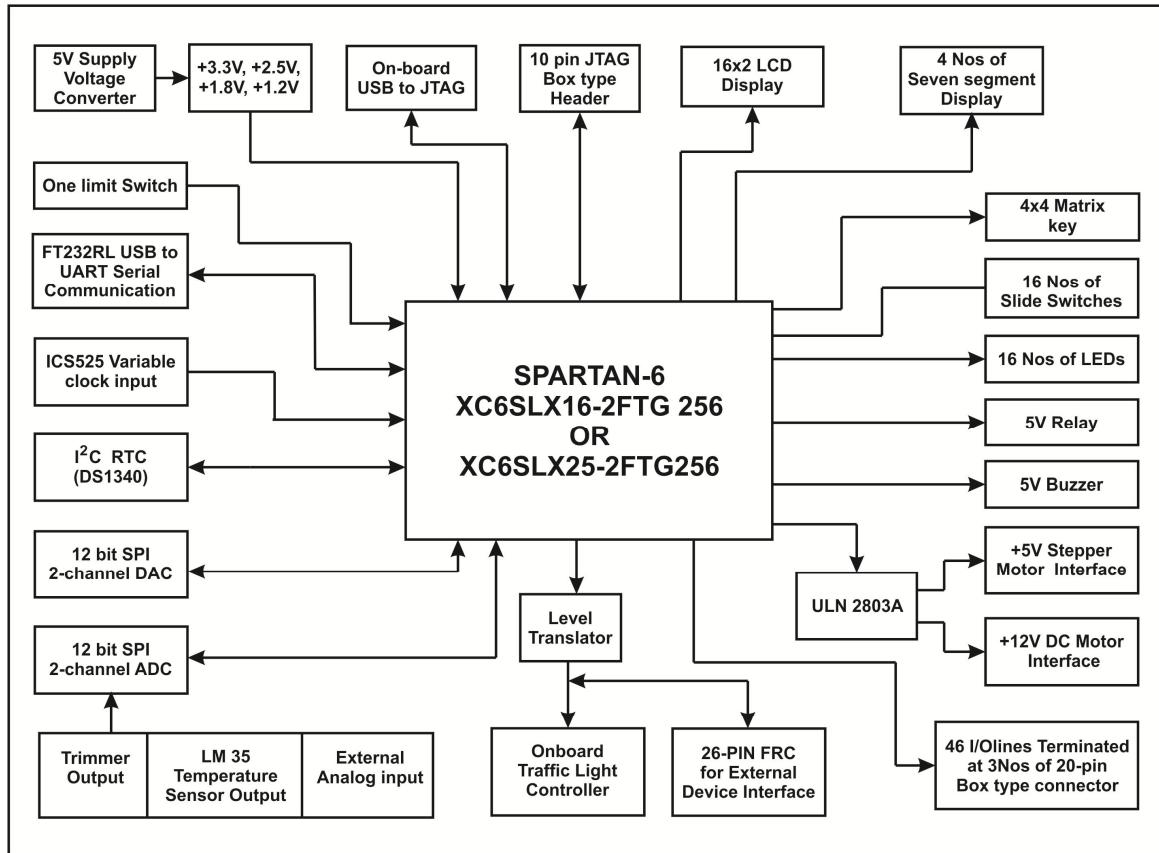
Features

- ➔ XC6SLX16 Based on Xilinx Spartan-6 FPGA features:
 - 14579 Logic cells
 - 2278 Kb of Distributed RAM
 - 136 Kb of Block RAM
 - 32 NOS of DSP Slices
 - 2 NOS of Clock Management Tiles (CMT)
 - 2 NOS of Memory Controller Block (MCB)
 - 186 User I/O lines
- ➔ 16 Nos. of digital input using slide switches with LED indication
- ➔ 16 Nos. of digital output using discrete LEDs
- ➔ 16× 2 LCD is provided for display the text message.
- ➔ One Reset Switch
- ➔ One Limit Switch for giving manual clock
- ➔ FPGA configuration through
 - On-board Xilinx USB to JTAG Programmer with isolation to configure FPGA
 - On board Flash Prom XCF04S (programmable through USB to JTAG programmer)
- ➔ Total 186 I/O Pins: 100 Pins used for integrating peripheral like LED, Switches etc., balance pins are available to user in 3 nos of 20-pin headers.
- ➔ 1 No of 26 pin header to interface VLIM cards like Traffic Light Controller / On-board TLC factory settable.
- ➔ On board programmable PLL oscillator from 1 MHz to 100MHz using jumpers.
- ➔ 4 Nos of 7 segment LED display
- ➔ One relay and Buzzer provided.
- ➔ Stepper & DC motor driver provided on board (Motor Optional)
- ➔ 4x4 matrix key provided
- ➔ SPI Based Analog to Digital Converter
 - Resolution → 12 bits
 - Number of Channel → 2 Channels
 - Input Range → 0 to 5V
 - Speed → 1 Msps

→ SPI Based Digital to Analog converter

- Resolution → 12 bits
- Number of Channel → 2 Channels
- Output Range → 0 to 5V
- Speed → 8.5 μS settling time

Block Diagram



CHAPTER 2

CLOCK SOURCE

Spartan-6 FPGA works in different clock frequencies. User can use any frequencies for their applications as given below,

PLL Oscillator Settings

Clock connection with FPGA

| SCHEMATIC NAME | FPGA PIN |
|----------------|----------|
| CLK | E10 |

Default Factory settings = 20 Mhz

For PLL, ICS525-01. Select the clock settings onboard

0 = Shorted
1 = Open

ICS525-01 Clock Table 1 to 100 Mhz

Table-1

| | 1Mhz | 3.6864 Mhz | 4Mhz | 20Mhz | 24Mhz | 25.175 Mhz | 48Mhz | 66Mhz | 80Mhz | 100Mhz |
|----|------|------------|------|-------|-------|------------|-------|-------|-------|--------|
| S2 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 |
| S1 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 |
| S0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 |
| R6 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| R5 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R4 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R3 | 1 | 0 | 1 | 0 | 0 | 1 | 0 | 1 | 0 | 0 |
| R2 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R1 | 1 | 0 | 1 | 0 | 1 | 1 | 1 | 0 | 0 | 0 |
| R0 | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 0 | 1 | 1 |
| V8 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| V7 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| V6 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| V5 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| V4 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 |
| V3 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
| V2 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 1 |
| V1 | 0 | 1 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 1 |
| V0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 1 |

For any other frequency use the online ICS252 calculator at

<http://www.idt.com//app=calculator> & device=525_01 or alternatively, the output of the ICS252 output frequency in between 1Mhz to 100Mhz use the following formula.

$$\text{Output Frequency} = \text{Crystal Frequency} \times 2^{\frac{(RDW + 8)}{(VDW + 2)(OD)}}$$

Where,

- Reference Divider Word (RDW) = 1 to 127 (0 is not permitted)
VCO Divider Word (VDW) = 4 to 511 (0,1,2,3 are not permitted)
Output Divider(OD) = Values below

EXAMPLE

To generate 12 MHz, assume Crystal frequency or Input frequency is 20MHz.

In general,

$$\text{Output Frequency} = \text{Crystal Frequency} \times 2^{\frac{(RDW + 8)}{(VDW + 2)(OD)}}$$

$$20\text{MHz} \times 2^{\frac{(RDW + 8)}{(VDW + 2)(OD)}} = 12\text{MHz}$$

ICS525-01 Output Divider and Maximum Output Frequency Table

Table:2

| S2 | S1 | S0 | CLK | Max. Output Frequency (MHz) | | | |
|------|------|------|----------------|-----------------------------|--------------|------------|--------------|
| Pin5 | Pin4 | Pin3 | Output Divider | VDD = 5V | | VDD = 3.3V | |
| | | | | 0 - 70°C | -40 to 85 °C | 0 - 70°C | -40 to 85 °C |
| 0 | 0 | 0 | 10 | 26 | 23 | 18 | 16 |
| 0 | 0 | 1 | 2 | 160 | 140 | 100 | 90 |
| 0 | 1 | 0 | 8 | 40 | 36 | 25 | 22 |
| 0 | 1 | 1 | 4 | 80 | 72 | 50 | 45 |
| 1 | 0 | 0 | 5 | 50 | 45 | 34 | 30 |
| 1 | 0 | 1 | 7 | 40 | 36 | 26 | 23 |
| 1 | 1 | 0 | 9 | 33.3 | 30 | 20 | 18 |
| 1 | 1 | 1 | 6 | 53 | 47 | 27 | 24 |

- * VCO Divider Word (VDW) = V8 to V0 = 000000100 = 4 (Decimal);
- * Reference Divider Word (RDW) = R6 to R0 = 0010010 = 18 (Decimal);
- * Output Divider (OP) = 2 (from the table given above)
- * VCO Divider Word (VDW) = V8 to V0 = 000000100 = 4 (Decimal);
- * Reference Divider Word (RDW) = R6 to R0 = 0010010 = 18 (Decimal);
- * Output Divider (OP) = 2 (from the table given above)

Fixed Clock / External Clock Input

On-board 20MHz oscillator (Y4) is used to give one more clock input to FPGA device. When using this clock input set the Jumper J15 in upward direction. When using external clock from P13 second pin / P10 second pin short the jumper J15 in downward direction.

Clock FPGA Pin → E7

CHAPTER - 3

SWITCHES & LEDS

3.1. Power Switch (Slide Switch (SW1)) & LED (L1)

The Spartan-6 Trainer Kit has a slide power switch. Moving the power switch Up for Power ON, the LED L1 will glow and down for power OFF, the LED L1 will off.

3.2. Configuration Switch & LED (SW2, L2)

The Spartan-6 Trainer Kit has a push button Switch (named as CONFIG) to Configure the FPGA from Xilinx Serial Flash PROM. During configuration process of the FPGA the LED L2 will be off and after successful configuration the LED L2 will glow.

3.3. Reset Switch (SW38)

The Spartan-6 Trainer kit has a push button switch (named as RESET) which can be configured as input by assigning the corresponding FPGA pin location.

| Name | FPGA |
|--------------|------|
| RESET (SW38) | E2 |

3.4 Limit Switch (SW21)

This switch is used to increment the speed or used to give manual clock. FPGA location for the switch is given below

| Name | SW21 |
|----------|------|
| FPGA Pin | R7 |

3.5 Input Switches

The Spartan-6 Trainer Kit has 16 nos of slide switches with led indication for giving inputs to the FPGA I/O lines. The slide switches are located in the bottom corner of the board and are labeled as SW22 through SW37. Switch SW22 is the left-most switch, and SW37 is the right-most switch. When in the UP or ON position, a switch connects the FPGA pin to 3.3V, a logic High. When DOWN or in the OFF position, the switch connects the FPGA pin to ground, a logic Low. The switches typically exhibit about 2 ms of mechanical bounce. There is no active de-bouncing circuitry, although such circuitry could easily be added to the FPGA design programmed on the board.

SPARTAN 6 TRAINER KIT

Slide Switch Connection with FPGA

| | | | | | | | | |
|----------|------|------|------|------|------|------|------|------|
| SWITCH | SW22 | SW23 | SW24 | SW25 | SW26 | SW27 | SW28 | SW29 |
| FPGA Pin | B8 | D8 | A7 | F7 | B6 | D6 | F6 | B5 |
| SWITCH | SW30 | SW31 | SW32 | SW33 | SW34 | SW35 | SW36 | SW37 |
| FPGA Pin | D5 | A4 | A3 | C3 | B2 | C2 | D3 | E3 |

3.6 Matrix Keypad

There are 16 momentary-contact push button switches arranged in a 4 x 4 matrix format, and multiplexed by 8 IO pins to the FPGA. There are 4 output lines named a3 to a0 (row) and 4 input lines available named as b3 to b0 (column).

Matrix Keypad Connections with FPGA

| | | | | | | | | |
|----------|------|------|------|------|------|------|------|------|
| Signal | a<0> | a<1> | a<2> | a<3> | b<0> | b<1> | b<2> | b<3> |
| FPGA Pin | J1 | H1 | H2 | H3 | H4 | H5 | G1 | G3 |

Output LEDs

The Spartan-6 Trainer Kit board has sixteen individual surface-mount LEDs located immediately above the slide switches. The LEDs are labeled L40 through L55. L40 is the left-most LED, L55 the right-most LED.

Each LED has one side connected to ground and the other side connected to a pin on the Spartan-6 device via a 270Ω current limiting resistor. To light an individual LED, drive the associated FPGA control signal High.

LED connections with FPGA

| | | | | | | | | |
|----------|-----|-----|-----|-----|-----|-----|-----|-----|
| LED | L40 | L41 | L42 | L43 | L44 | L45 | L46 | L47 |
| FPGA Pin | A8 | C8 | E8 | C7 | A6 | C6 | E6 | A5 |
| LED | L48 | L49 | L50 | L51 | L52 | L53 | L54 | L55 |
| FPGA Pin | C5 | F5 | E4 | B3 | A2 | B1 | C1 | D1 |

Seven Segment Display

The Spartan 6 Trainer board has a four-character, seven segment LED display controlled by FPGA user-I/O pins. Each individual character has a separate common pin, and this pin is connected to J10 3-pin jumper. When using common cathode control input seven segment display short this jumper to ground and when using common anode control input seven segment display short this jumper to +3.3V. The LED control signals are connected to the individual line of the FPGA I/O.

| Character | a | b | c | d | e | f | g |
|-----------|---|---|---|---|---|---|---|
| 1 | 0 | 1 | 1 | 0 | 0 | 0 | 0 |
| 2 | 1 | 1 | 0 | 1 | 1 | 0 | 1 |
| 3 | 1 | 1 | 1 | 1 | 0 | 0 | 1 |
| 4 | 0 | 1 | 1 | 0 | 0 | 1 | 1 |
| 5 | 1 | 0 | 1 | 1 | 0 | 1 | 1 |
| 6 | 1 | 0 | 1 | 1 | 1 | 1 | 1 |
| 7 | 1 | 1 | 1 | 0 | 0 | 0 | 0 |
| 8 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| 9 | 1 | 1 | 1 | 0 | 0 | 1 | 1 |
| A | 1 | 1 | 1 | 0 | 1 | 1 | 1 |
| B | 0 | 0 | 1 | 1 | 1 | 1 | 1 |
| C | 1 | 0 | 0 | 1 | 1 | 1 | 0 |
| D | 0 | 1 | 1 | 1 | 1 | 0 | 1 |
| E | 1 | 0 | 0 | 1 | 1 | 1 | 1 |
| F | 1 | 0 | 0 | 0 | 1 | 1 | 1 |

SEVEN SEGMENT Connections with FPGA**Disp1**

| DISPLAY1 | seg-a | seg-b | seg-c | seg-d | seg-e | seg-f | seg-g | seg-dp |
|----------|-------|-------|-------|-------|-------|-------|-------|--------|
| FPGA Pin | T7 | P7 | N5 | P1 | P2 | L8 | N8 | N4 |

SPARTAN 6 TRAINER KIT

Disp3

| | | | | | | | | |
|----------|-------|-------|-------|-------|-------|-------|-------|--------|
| DISPLAY2 | seg-a | seg-b | seg-c | seg-d | seg-e | seg-f | seg-g | seg-dp |
| FPGA Pin | P5 | P4 | L4 | L5 | L7 | R1 | R2 | L3 |

Disp2

| | | | | | | | | |
|----------|-------|-------|-------|-------|-------|-------|-------|--------|
| DISPLAY3 | seg-a | seg-b | seg-c | seg-d | seg-e | seg-f | seg-g | seg-dp |
| FPGA Pin | P6 | N6 | M6 | N1 | N3 | T6 | M7 | M5 |

Disp4

| | | | | | | | | |
|----------|-------|-------|-------|-------|-------|-------|-------|--------|
| DISPLAY4 | seg-a | seg-b | seg-c | seg-d | seg-e | seg-f | seg-g | seg-dp |
| FPGA Pin | T4 | T3 | M2 | M3 | M4 | T5 | R5 | M1 |

CHAPTER-4

RELAY & BUZZER

The Spartan 6 trainer contains Buzzer & Relay circuit, and both works in 5V DC Voltage. A buzzer or beeper is a signaling device, usually electronic, typically used in automobiles, household appliances such as a microwave oven, or game shows. Buzzer generates different tone Generator. Relay is used as On/Off switch depending upon user needs.

BUZZER & RELAY Connections with FPGA

| SIGNAL | Buzzer | Relay |
|----------|--------|-------|
| FPGA Pin | K2 | L1 |

CHAPTER - 5

STEPPER MOTOR & DC MOTOR CONTROL

STEPPER MOTOR & DC MOTOR

The Spartan 6 trainer Contains Stepper Motor Interface Connector to Control the Stepper Motor Speed and it also controls the Step Angle of the Motor.

Spartan 6 trainer board has the provision to interface the 5V stepper motor and +12V DC motor through the ULN2803A driver IC. P9 2-pin J801 connector has the option to give external supply voltage to the driver IC and J13 jumper will choose the driver voltage is be the on-board +5V or be the external voltage. Driver output lines for stepper motor are terminated in P11 5-pin RMC connector and for DC motor lines are terminated in P7 2-pin J801 connector. But we cannot execute the stepper motor and dc motor alone not in the same time.

Motor connections with FPGA

| SIGNAL | STM1/DC1 | STM2 | STM3 | STM4 |
|----------|----------|------|------|------|
| FPGA Pin | K1 | J6 | J4 | J3 |

CHAPTER – 6

LCD DISPLAY

Once mastered, the LCD is a practical way to display a variety of information using standard ASCII and custom characters. However, these displays are not fast. Scrolling the display at half-second intervals tests the practical limit for clarity.

LCD Connections with FPGA

| LCD | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 | RS | CS |
|----------|-----|----|----|----|----|----|----|----|-----|-----|
| FPGA Pin | T10 | M9 | N9 | P9 | R9 | T9 | T8 | P8 | T10 | M10 |

Voltage Compatibility

The character LCD is power by +5V. The FPGA I/O signals are powered by 3.3V. However, the FPGA's output levels are recognized as valid Low or High logic levels by the LCD. The LCD controller accepts 5V TTL signal levels and the 3.3V LVC MOS outputs provided by the FPGA meet the 5V TTL voltage level requirements.

The 390Ω series resistors on the data lines prevent over stressing on the FPGA and Strata Flash I/O pins when the character LCD drives a High logic value. The character LCD drives the data lines when LCD R/W is High. Most applications treat the LCD as a write only peripheral and never read from the display and hence in this trainer the R/W pin is grounded by default.

| | | |
|-----------------------------|---|--------|
| 1 st line enable | → | X “80” |
| 2 nd line enable | → | X “C0” |
| Function set | → | X “38” |
| To clear the display | → | X “01” |
| Character blinking | → | X “0F” |
| Entry mode set | → | X “06” |

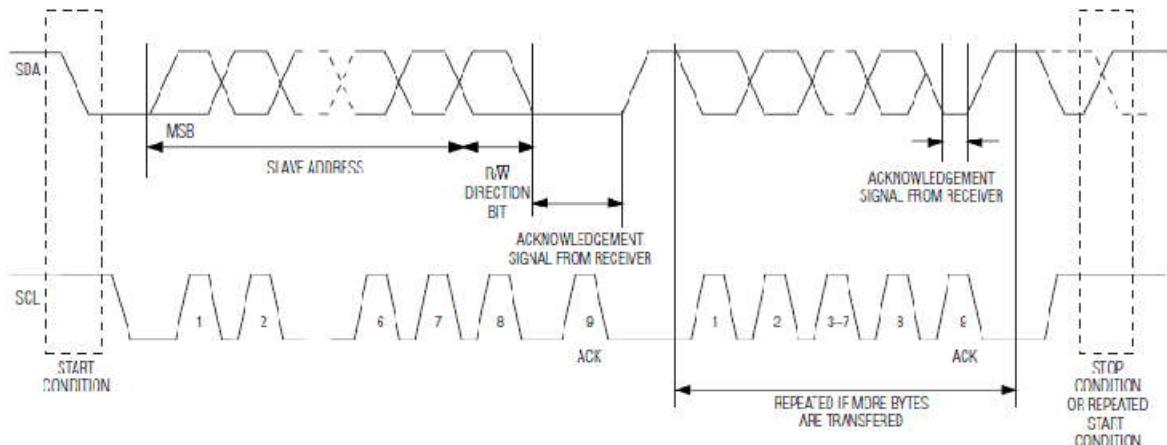
Real Time Clock (RTC)

Spartan 6 project card has one RTC (Real Time Clock) IC DS1340 in its onboard. It's a 2-wire 8pin RTC IC, uses a low-cost 32.768 kHz crystal, it tracks time using several internal registers. The clock/calendar automatically adjusts for months with fewer than 31 days, including corrections for leap years. Each pin function and FPGA connections is given below.

| PIN | NAME | FUNCTION | FPGA PIN |
|-----|---------|--|----------|
| 1 | X1 | X1 is the input to the oscillator | - |
| 2 | X2 | X1 is the output to the oscillator | - |
| 3 | VBACKUP | Connection for a Secondary Power Supply. For the 1.8V and 3V devices, VBACKUP must be held between 1.3V and 3.7V for proper operation. VBACKUP can be as high as 5.5V on the 3.3V device. This pin can be connected to a primary cell such as a lithium coin cell. | - |
| 4 | GND | Ground | - |
| 5 | SDA | Serial Data Input/output. SDA is the data input/output for the 2-wire serial interface. The SDA pin is open drain and requires an external pull-up resistor. | P12 |
| 6 | SCL | Serial Clock Input. SCL is the clock input for the 2-wire interface and is used to synchronize data movement on the serial interface. | R12 |
| 7 | FT/OUT | Frequency Test/Output. This pin is used to output either a 512Hz signal or the value of the OUT bit. When the FT bit is logic 1, the FT/OUT pin toggles at a 512Hz rate. When the FT bit is logic 0, the FT/OUT pin reflects the value of the OUT bit. | T12 |
| 8 | VCC | DC Power for Primary Power Supply (3.3V). | - |

0x07 is the control register to control the data transfer.

DATA TRANSFER SEQUENCE ON THE SERIAL BUS



The RTCC registers are contained in addresses 0x00h-0x06h.

0x00 register contains the time format seconds data, it have BCD ones (bit 0 to 3) and tens (bit 4 to 6).

0x01 register have time format minutes data in the order of BCD ones (bit 0 to 3) and tens (bit 4 to 6).

0x02 register have time format hour data, is situated as BCD ones (bit 0 to 3) and tens (bit 4 to 5).

0x03 register for day.

0x04 register for date, have BCD ones in bit 0 to 3 and tens in bit 4 to 5.

0x05 register for month, have BCD ones in bit 0 to 3 and tens in bit 4.

0x06 register for year, have BCD ones in bit 0 to 3 and tens in bit 4 to 7. Corrections include for leap year also.

ADC & DAC

Analog to Digital Conversion (ADC)

Spartan 6 Trainer Kit has one SPI-compatible; two channel Analog to Digital Converter (ADC) by using AD122S101 IC. It is a single channel, 12 bit, low power, high speed, and single power supply with 2.7V - 5.25V range, successive approximation, serial interface IC. And it is a 8 pin IC.

1st pin (CS bar) is a chip select pin. When it is low then only the ADC conversion process will start.

2nd pin (VA) is connected to +3.3V supply and bypassed with 0.1uF capacitor.

3rd pin (GND) is directly connected to ground.

4th (IN2) and 5th (IN1) pin (Vin) is a analog input, it ranges from 0V to VA.

6th pin is a digital data input. Control Register is loaded through this pin on rising edges of the SCLK pin.

7th pin (DOUT) is a serial digital output, at falling edge of the SCLK signal.

8th pin (SCLK) is a serial clock input. This controls the conversion and readout processes.

CS is chip select, which initiates conversions and frames the serial data transfers. SCLK (serial clock) controls both the conversion process and the timing of serial data. DOUT is the serial data output pin, where a conversion result is sent as a serial data stream,

MSB first. Data to be written to the ADC122S101's Control Register is placed at DIN, the serial data input pin. New data is written to DIN with each conversion.

A serial frame is initiated on the falling edge of CS and ends on the rising edge of CS. Each frame must contain an integer multiple of 16 rising SCLK edges. The ADC output data (DOUT) is in a high impedance state when CS is high and is active when CS is low. Thus, CS acts as an output enable.

ADC Control Register and Bit Explanation

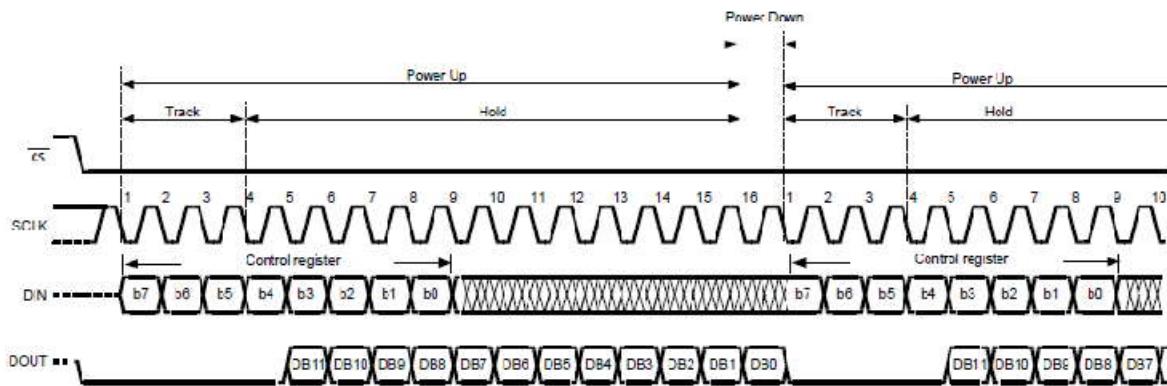
| Bit 7 (MSB) | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bt 2 | Bit 1 | Bit 0 |
|-------------|-------|-------|-------|-------|-------|-------|-------|
| DONTC | DONTC | ADD2 | ADD1 | ADD0 | DONTC | DONTC | DONTC |

| Bit #: | Symbol: | Description |
|--------------|---------|---|
| 7 - 6, 2 - 0 | DONTC | Don't care. The value of these bits do not affect the device. |
| 3 | ADD0 | These three bits determine which input channel will be sampled and converted in the next track/hold cycle. The mapping between codes and channels is shown in Table 4 . |
| 4 | ADD1 | |
| 5 | ADD2 | |

Input Channel Selection

| ADD2 | ADD1 | ADD0 | Input Channel |
|------|------|------|--|
| x | 0 | 0 | IN1 (Default) |
| x | 0 | 1 | IN2 |
| x | 1 | x | Not allowed. The output signal at the DOUT pin is indeterminate if ADD1 is high. |

Timing Diagram



ADC Analog inputs**External Input**

Pot meter or other analog input signal is used to give input to ADC channels and the ADC will support the voltage range from 0V to +3.3V. Connect the 1st pin of pot meter to the 5th pin of P12 connector, 1st pin to 6th pin of P6 and 2nd pin to 1st pin of P12 (ADC Channel 1). If we want to give analog input to second channel of ADC, connect 2nd point of pot meter to 2nd pin of P12 connector. Download .bit file and rotate the pot meter and the corresponding digital output is displayed in the 12 LEDs (L55 to L44).

On-Board Analog Input

On-board TP5 trimmer is used to give analog input to ADC CH1 or ADC CH2 by using J16 jumper. When we short J16 pin 1 and 2, the trim-pot output is connected to ADC CH1 and when J16 pin 2 and 3 shorted, trim-pot output is connected to ADC CH2. Set the jumpers and download the .bit file. The output will be displayed in 16 LEDs (L55 to L40).

Temperature Sensor Input

LM35 temperature is used in Spatan-6 Trainer Kit to measure the external environment temperature. This sensor converts the temperature into corresponding voltage. Temperature conversion is in °C (Centigrade). 1°C will gives 10mV voltage. By using this relation, calibration is made in program. Voltage output is given as an analog input to ADC CH1 by short the jumper J19. At that time we should not give external analog input and on-board trimmer analog input to ADC CH1. Download the .bit file, at normal condition room temperature is measured and that voltage is given to ADC. When we apply external heat to the sensor, that corresponding unknown temperature value is displayed in LCD in °C (Centigrade).

FPGA Pin Connections with ADC

| ADC | SCLK | SDATA | CS bar | DIN |
|----------|------|-------|--------|-----|
| FPGA Pin | F2 | F1 | G5 | G6 |

Digital to Analog Conversion (DAC)

Spartan 6 Trainer Kit has one SPI-compatible, single channel Digital to Analog Converter (DAC) by using DAC121S085 IC. It is a two channel, 12 bit, low power, high speed, and power supply with the range of 2.7V - 5.25V, serial interface IC. And it is a 10 pin IC.

In this IC 1st pin (VA) is connected to +3.3V supply and bypassed with 0.1uF capacitor.

(DACOUT) is a DAC output pin.

2nd pin (VOUTA) is a first channel DAC output pin.

3rd pin (VOUTB) is a second channel DAC output pin.

4th and 5th pin NC.

6th pin is (GND) is directly connected to ground.

7th pin (Vrefin) is connected to +3.3V supply and bypassed with 0.1uF capacitor.

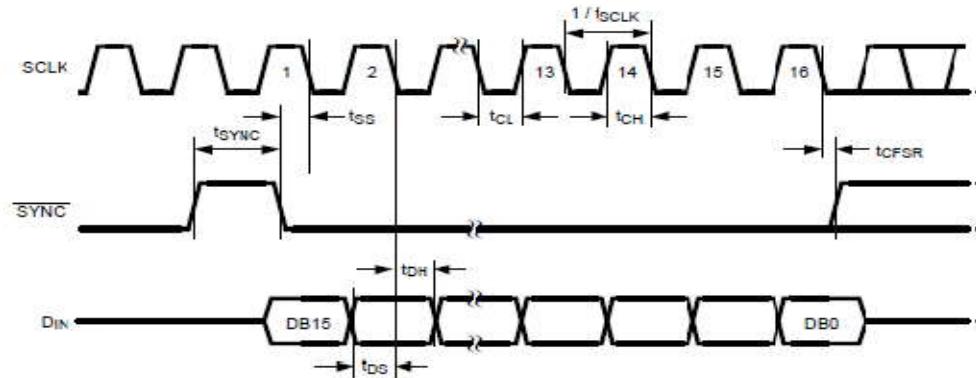
8th pin (DIN) is serial Data Input. Data is clocked into the 16-bit shift register on the falling edges of SCLK after the fall of SYNC.

9th pin (SYNC bar) is frame synchronization input for the data input. When this pin goes low, it enables the input shift register and data is transferred on the falling edges of SCLK.

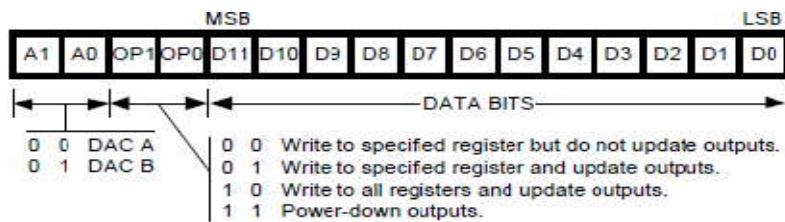
10th pin (SCLK) is serial Clock Input. Data is clocked into the input shift register on the falling edges of this pin.

The input shift register has sixteen bits. The first bit must be set to "0" and the second bit is an address bit. The address bit determines whether the register data is for DAC A or DAC B. This bit is followed by two bits that determine the mode of operation (writing to a DAC register without updating the outputs of both DACs, writing to a DAC register and updating the outputs of both DACs, writing to the register of both DACs and updating their outputs, or powering down both outputs). The final twelve bits of the shift register are the data bits. The data format is straight binary (MSB first, LSB last), with all 0's corresponding to an output of 0V and all 1's corresponding to a full-scale output of VREFIN - 1 LSB. The contents of the serial input register are transferred to the DAC register on the sixteenth falling edge of SCLK.

SYNC line is kept low for at least 16 falling edges of SCLK and the DAC is updated on the 16th SCLK falling edge. However, if SYNC is brought high before the 16th falling edge, the data transfer to the shift register is aborted and the write sequence is invalid. Under this condition, the DAC register is not updated and there is no change in the mode of operation or in the DAC output voltages.



Input Shift Register



FPGA Pin Connections with DAC

| DAC PIN | DAC CS Bar (SYNC) | DAC SCLK | DAC IN (SDA) |
|----------|----------------------|----------|--------------|
| FPGA PIN | F4 | E1 | F3 |

USB to UART

Spartan-6 Trainer Kit supports UART communication with PC using USB to UART Converter Bridge (USB 2.0). FT232RL device act as the USB to UART Bridge between Spartan-6 FPGA and PC. Transmit (TXD) and Receive (RXD) lines are only used in this board. When connecting the board USB to UART with PC, open Device Manager to know the COM port. We can choose the baud rate for the program using the following formula,

$$\text{Baud Rate} = \text{Frequency} / \text{Count Value}$$

Ex, Frequency → 20MHz, Count Value → 2083 and Baud Rate → 9600

FPGA Pin Connections

| UART PIN | TXD | RXD |
|----------|-----|-----|
| FPGA PIN | P11 | M11 |

On-Board USB to JTAG

Spartan-6 Trainer Kit has On-Board USB to JTAG programmer section to download the configuration file into the FPGA and PROM device and it has the USB provision to connect the Spartan-6 Trainer Kit to PC USB port using USB cable. Thus we may not require the external programmer to configure the device. JTAG lines from the device are converted to USB signals and these USB signals are terminated in the P3 USB connector.

20-pin Expansion Connectors

There are three 20-pin connectors presented in Spartan-6 Trainer Kit and general purpose I/O lines are terminated in these connectors to interface the external hardware peripherals with Spartan-6 FPGA. We can give 3.3V level external signals to FPGA through these connectors and can output 3.3V level signals from FPGA. Don't give more than 3.3V level signals to FPGA because it induces harmful effect to Spartan-6 device.

P10 20-pin Connector

J14 3-pin jumper is used to select the supply voltage for P10 connector. When we short J14 pin-1 and pin-2, 3.3V is given to pin-1 of the P10 connector and short J14 pin-2 and pin-3, +5V is connected. This gives the supply to the external hardware which is connected to P10 connector.

| Connector Pin | Signal Name | FPGA Pin |
|---------------|---------------------|---|
| 1 | Supply (3.3V or 5V) | - |
| 2 | EXCLK | E7 (J15 3-pin jump 3 rd pin) |
| 3 | IO35 | J12 |
| 4 | IO36 | J13 |
| 5 | IO37 | K15 |
| 6 | IO38 | J11 |
| 7 | IO39 | K14 |
| 8 | IO40 | K16 |
| 9 | IO41 | K11 |
| 10 | IO42 | K12 |
| 11 | IO43 | L16 |
| 12 | IO44 | G14 |
| 13 | IO45 | A12 |
| 14 | IO46 | A9 |
| 15 | M0 | T11 |
| 16 | M1 | N11 |
| 17 | DIN | P10 |
| 18 | INIT_B | R3 |
| 19 | CCLK | R11 |
| 20 | Ground | GND |

P6 20-pin Connector

J11 3-pin jumper is used to select the supply voltage for P6 connector. When we short J11 pin-1 and pin-2, 3.3V is given to pin-1 of the P6 connector and short J11 pin-2 and pin-3, +5V is connected. This gives the supply to the external hardware which is connected to P6 connector.

| Connector Pin | Signal Name | FPGA Pin |
|---------------|---------------------|----------|
| 1 | Supply (3.3V or 5V) | - |
| 2 | - | - |
| 3 | IO18 | M16 |
| 4 | IO19 | L12 |
| 5 | IO20 | M14 |
| 6 | IO21 | M15 |
| 7 | IO22 | M12 |
| 8 | IO23 | M13 |
| 9 | IO24 | N14 |
| 10 | IO25 | N16 |
| 11 | IO26 | P16 |
| 12 | IO27 | N12 |
| 13 | IO28 | R16 |
| 14 | IO29 | P15 |
| 15 | IO30 | T15 |
| 16 | IO31 | R15 |
| 17 | IO32 | T14 |
| 18 | IO33 | R14 |
| 19 | IO34 | T13 |
| 20 | Ground | GND |

P13 20-pin Connector

J17 3-pin jumper is used to select the supply voltage for P13 connector. When we short J17 pin-1 and pin-2, 3.3V is given to pin-1 of the P13 connector and short J17 pin-2 and pin-3, +5V is connected. This gives the supply to the external hardware which is connected to P13 connector. **In board J17 voltage marking is interchanged. +5V notification for +3.3V, +3.3V notification for +5V.**

| Connector Pin | Signal Name | FPGA Pin |
|---------------|---------------------|---|
| 1 | Supply (3.3V or 5V) | - |
| 2 | EXCLK | E7 (J15 3-pin jump 3 rd pin) |
| 3 | IO1 | D11 |
| 4 | IO2 | D12 |
| 5 | IO3 | D14 |
| 6 | IO4 | D16 |
| 7 | IO5 | E11 |
| 8 | IO6 | E12 |

SPARTAN 6 TRAINER KIT

| | | |
|----|--------|-----|
| 9 | IO7 | E13 |
| 10 | IO8 | E15 |
| 11 | IO9 | E16 |
| 12 | IO10 | F10 |
| 13 | IO11 | F12 |
| 14 | IO12 | F13 |
| 15 | IO13 | F14 |
| 16 | IO14 | F15 |
| 17 | IO15 | F16 |
| 18 | IO16 | G11 |
| 19 | IO17 | G12 |
| 20 | Ground | GND |

P14 26-PIN Connector

P14 connector allows the restrictions of 20-pin connectors because we can in and out the +5V logic level signals in P14 connector. For that voltage level translators (bidirectional) U15, U21 and U22 (SN74LVCC3245A) are used in Spartan-6 Trainer Kit. Totally 24 I/O lines are terminated in this connector through three SN74LVCC3245A ICs. Voltage conversion from A → B (FPGA +3.3V to Connector +5V) and B → A (Connector +5V to FPGA +3.3V) is controlled by the DIR pin in level translator IC.

| Connector Pin | Signal Name | IC | IC Signal | FPGA Pin |
|---------------|-------------|-----|-----------|----------|
| 1 | PA0 | U15 | SPA0 | J14 |
| 2 | PA1 | | SPA1 | J16 |
| 3 | PA2 | | SPA2 | H16 |
| 4 | PA3 | | SPA3 | H15 |
| 5 | PA4 | | SPA4 | H14 |
| 6 | PA5 | | SPA5 | H13 |
| 7 | PA6 | | SPA6 | H11 |
| 8 | PA7 | | SPA7 | G16 |
| | | | PADIR | L14 |
| 9 | PB0 | U21 | SPB0 | B12 |
| 10 | PB1 | | SPB1 | A11 |
| 11 | PB2 | | SPB2 | C10 |
| 12 | PB3 | | SPB3 | B10 |
| 13 | PB4 | | SPB4 | A10 |
| 14 | PB5 | | SPB5 | F9 |
| 15 | PB6 | | SPB6 | D9 |
| 16 | PB7 | | SPB7 | C9 |
| | | | PBDIR | L13 |
| 17 | PC0 | U22 | SPC0 | C15 |
| 18 | PC1 | | SPC1 | C13 |
| 19 | PC2 | | SPC2 | C11 |
| 20 | PC3 | | SPC3 | B16 |
| 21 | PC4 | | SPC4 | B15 |

SPARTAN 6 TRAINER KIT

| | | | | |
|----|--------|--|-------|-----|
| 22 | PC5 | | SPC5 | B14 |
| 23 | PC6 | | SPC6 | A14 |
| 24 | PC7 | | SPC7 | A13 |
| | | | PCDIR | C16 |
| 25 | Ground | | | Gnd |
| 26 | VCC | | | +5V |

Traffic Light Controller (TLC)

Spartan-6 Trainer Kit has On-Board Traffic Light Controller section which reflects the original TLC what we seen in our day to day life. This section includes three colors (Red, Green and Yellow) controller LEDs for signal controlling and 8 pedestrian LEDs are also placed. Each road has red, yellow and three green LEDs are placed to indicate right turning, left turning and straight.

| Road | Color | LED | Signal Name | FPGA Pin |
|-------|------------|-----|-------------|----------|
| Road1 | Red | L25 | PB6 | D9 |
| | Yellow | L24 | PB7 | C9 |
| | Green | L23 | PC0 | C15 |
| | Green | L17 | PC2 | C11 |
| | Green | L29 | PC1 | C13 |
| | Pedestrian | L19 | NOT PB0 | B12 |
| | Pedestrian | L31 | NOT PB0 | B12 |
| Road2 | Red | L12 | PA2 | H16 |
| | Yellow | L11 | PA1 | J16 |
| | Green | L9 | PA0 | J14 |
| | Green | L10 | PA4 | H14 |
| | Green | L8 | PA3 | H15 |
| | Pedestrian | L16 | NOT PB1 | A11 |
| | Pedestrian | L14 | NOT PB1 | A11 |
| Road3 | Red | L26 | PC3 | B16 |
| | Yellow | L27 | PC4 | B15 |
| | Green | L28 | PC5 | B14 |
| | Green | L22 | PC7 | A13 |
| | Green | L34 | PC6 | A13 |
| | Pedestrian | L20 | NOT PB2 | C10 |
| | Pedestrian | L33 | NOT PB2 | C10 |
| Road4 | Red | L39 | PA5 | H13 |
| | Yellow | L56 | PA6 | H11 |
| | Green | L58 | PA7 | G16 |
| | Green | L59 | PB5 | F9 |
| | Green | L57 | PB4 | H14 |
| | Pedestrian | L37 | NOT PB3 | B10 |
| | Pedestrian | L35 | NOT PB3 | B10 |

CHAPTER-7

PROCEDURE TO WORK IN XILINX SOFTWARE

1. After installing Xilinx software, go to **Start Menu → Programs → Xilinx ISE 12.1→ ISE Design tools→Project Navigator** (refer **Figure-1**).

A Window shown in **Figure-2** will appear.

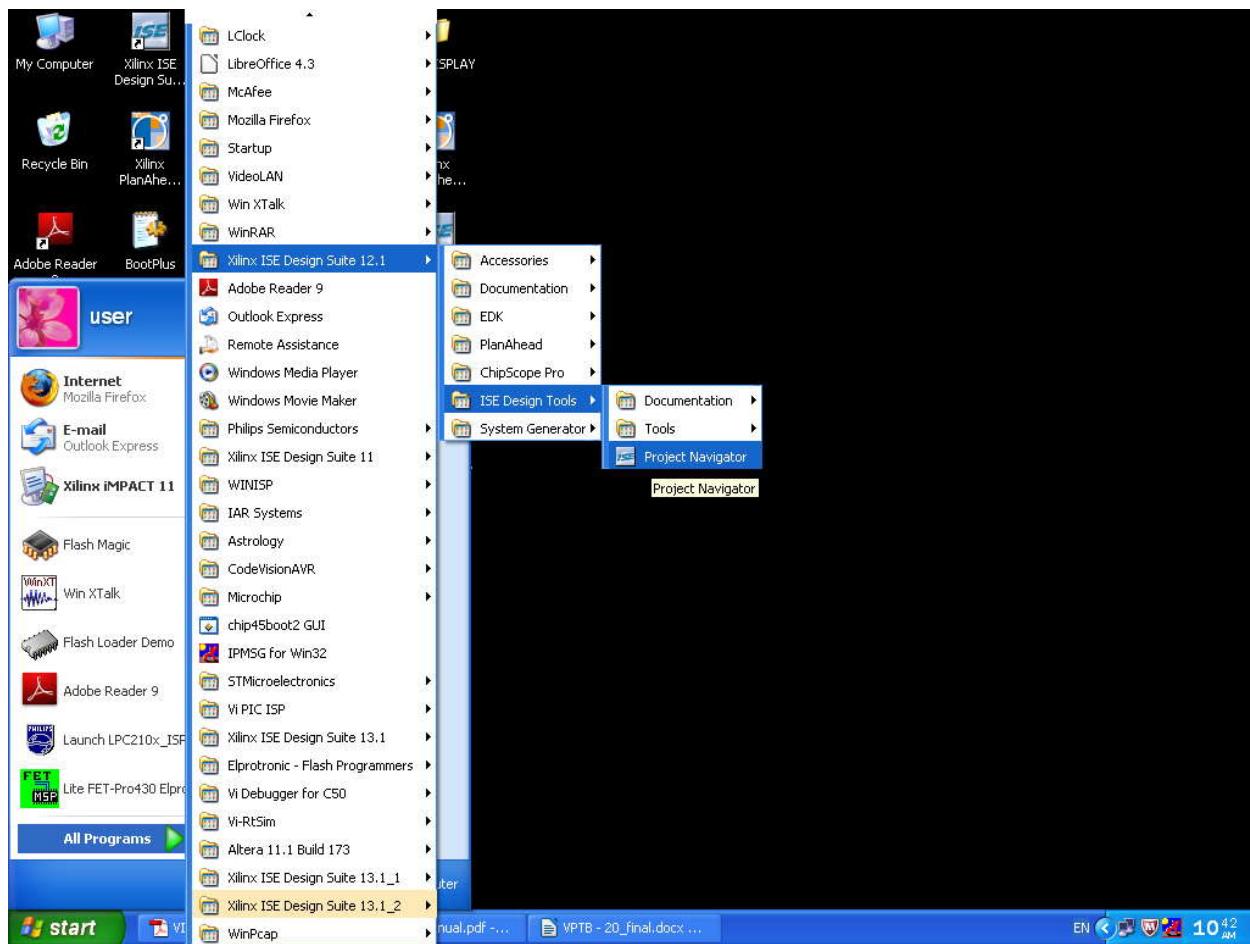


Figure-1

SPARTAN 6 TRAINER KIT

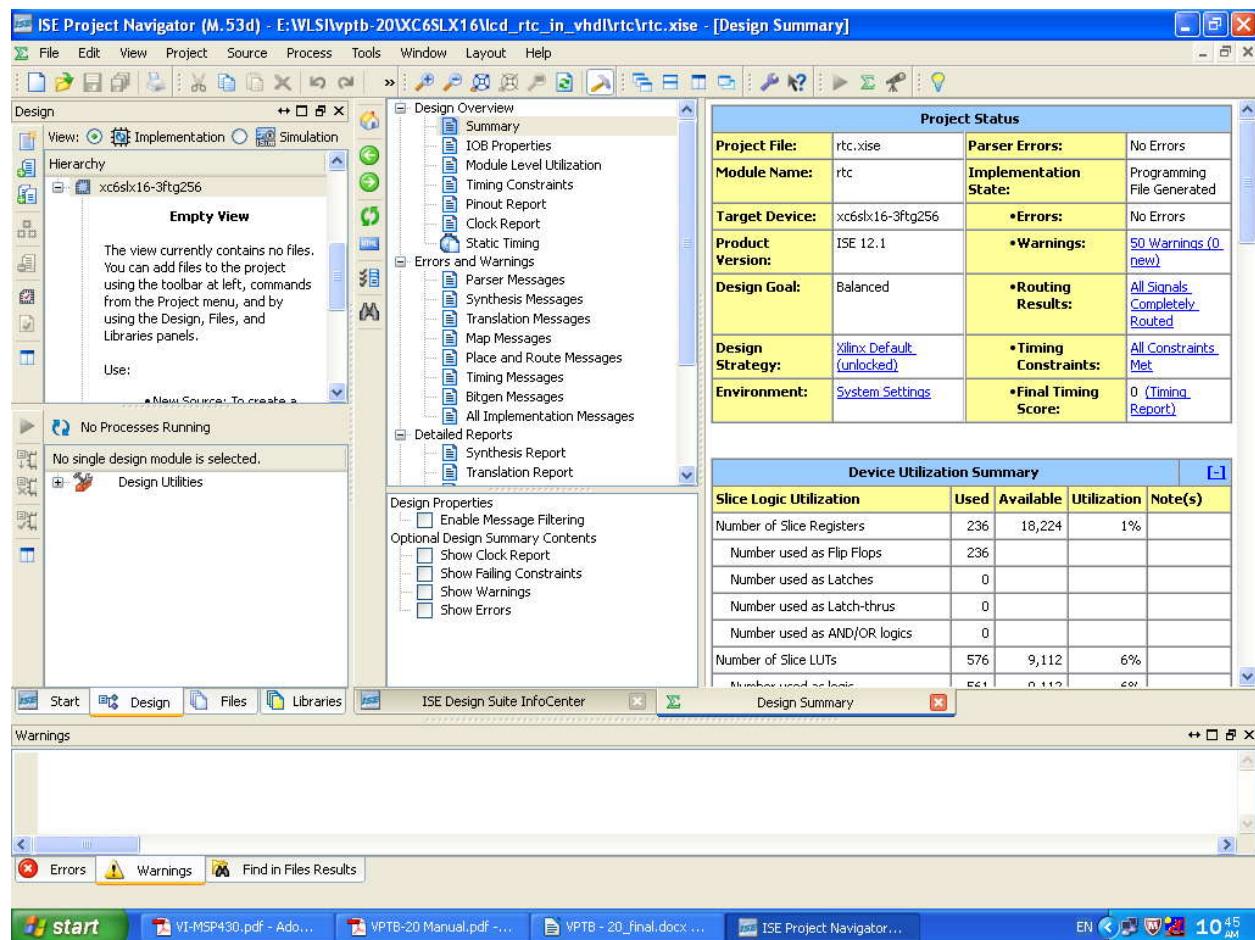


Figure-2

SPARTAN 6 TRAINER KIT

2. Select File → New Project

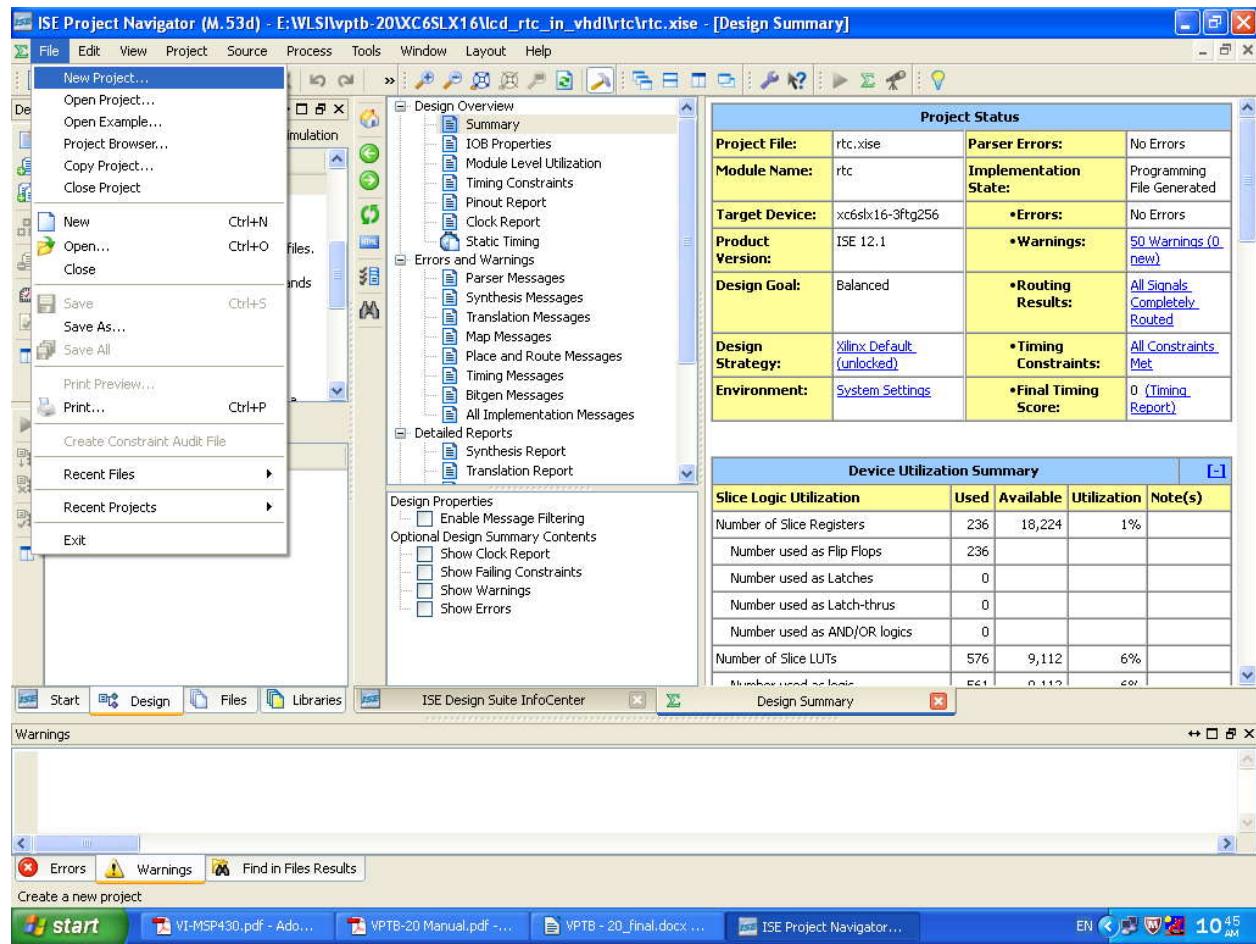


Figure-3

SPARTAN 6 TRAINER KIT

3. A window given in **Figure-4** will appear. In the project name field, give your project Name and select the location where you want to save the project (refer **Figure-5**). In the **Top-Level Module** selects **HDL** and click **Next**.

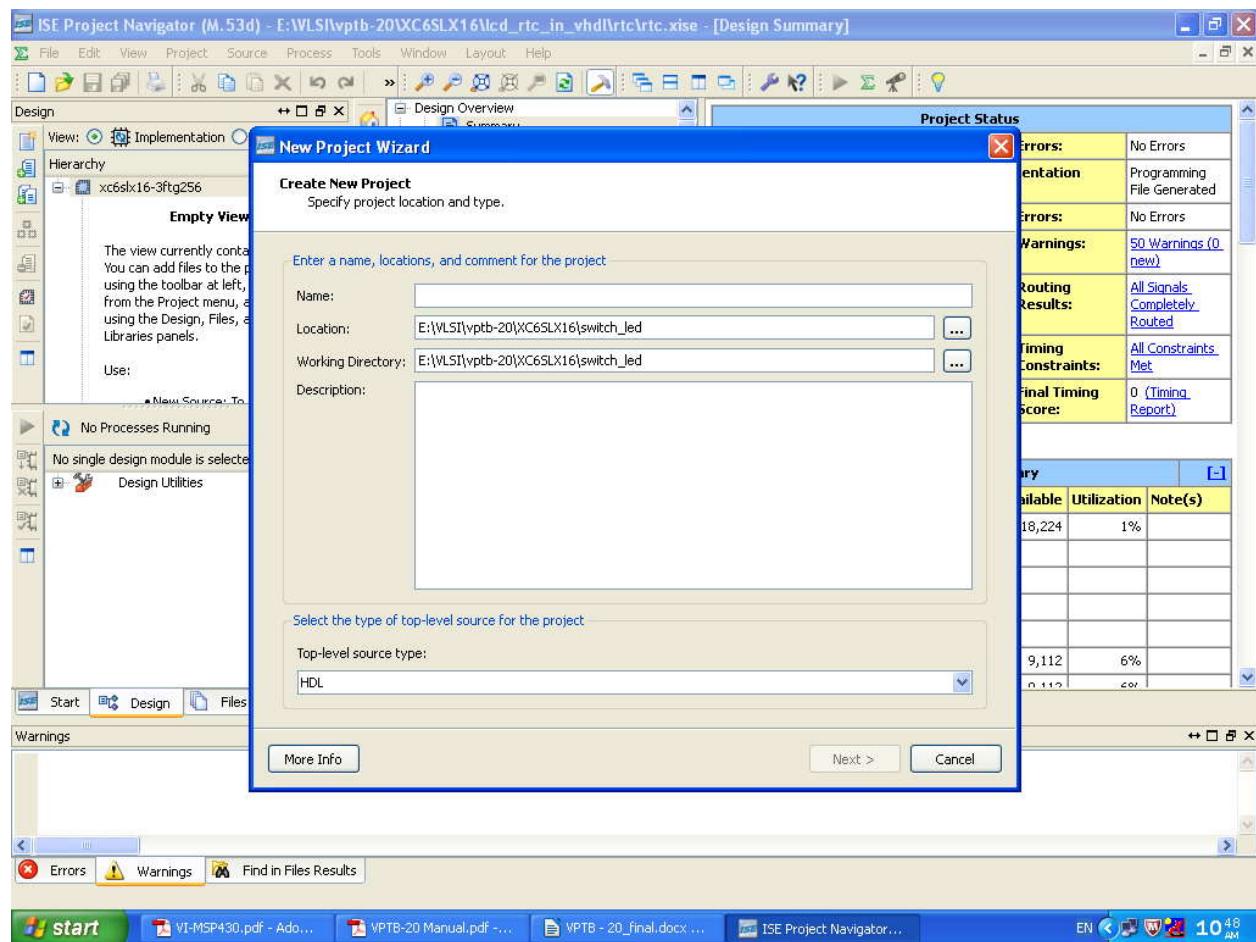


Figure-4

SPARTAN 6 TRAINER KIT

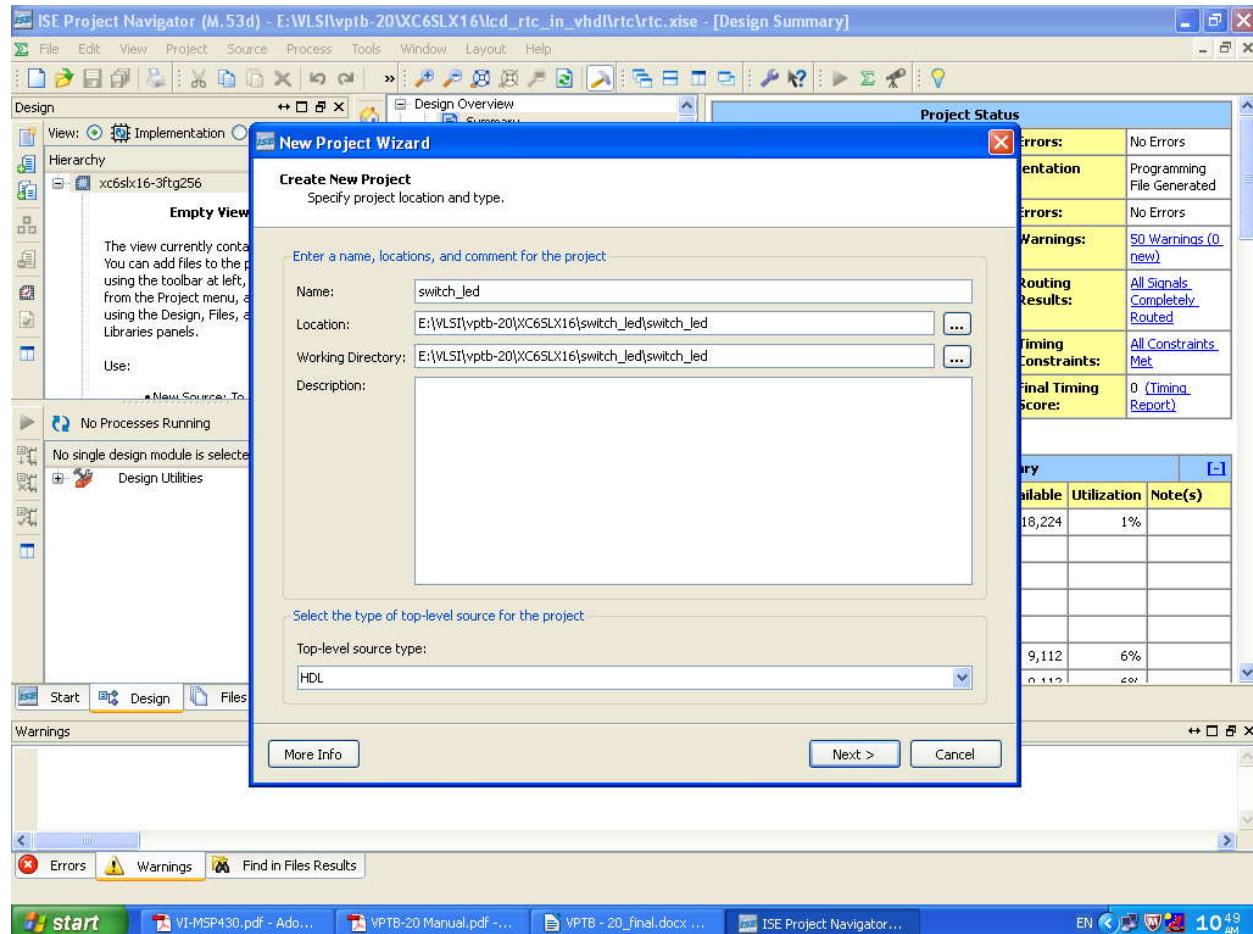


Figure-5

4. A window given in **Figure-6** will appear.

In the Device and Design flow for the project, select

- | | | |
|--------------------|---|--------------------------------|
| Product category | → | All |
| Family | → | Spartan6 |
| Device | → | XC6SLX16 |
| Package | → | FTG256 |
| Speed | → | -2 |
| Synthesis Tool | → | XST (VHDL / Verilog) |
| Simulator | → | ISE Simulator (VHDL / Verilog) |
| Preferred Language | → | VHDL |

Then click "Next" and "Finish" (refer **Figure-6, 7**).

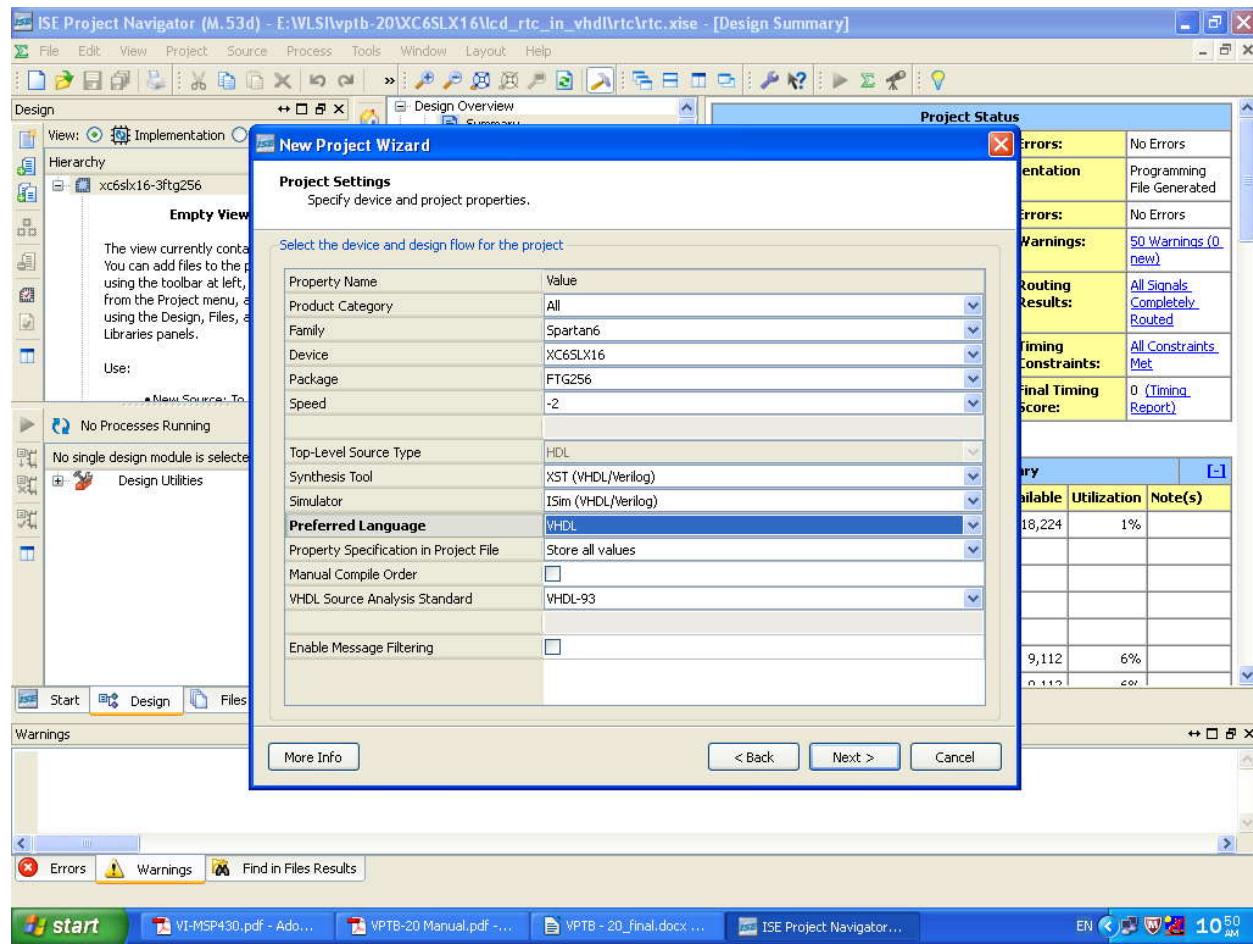


Figure-6

SPARTAN 6 TRAINER KIT

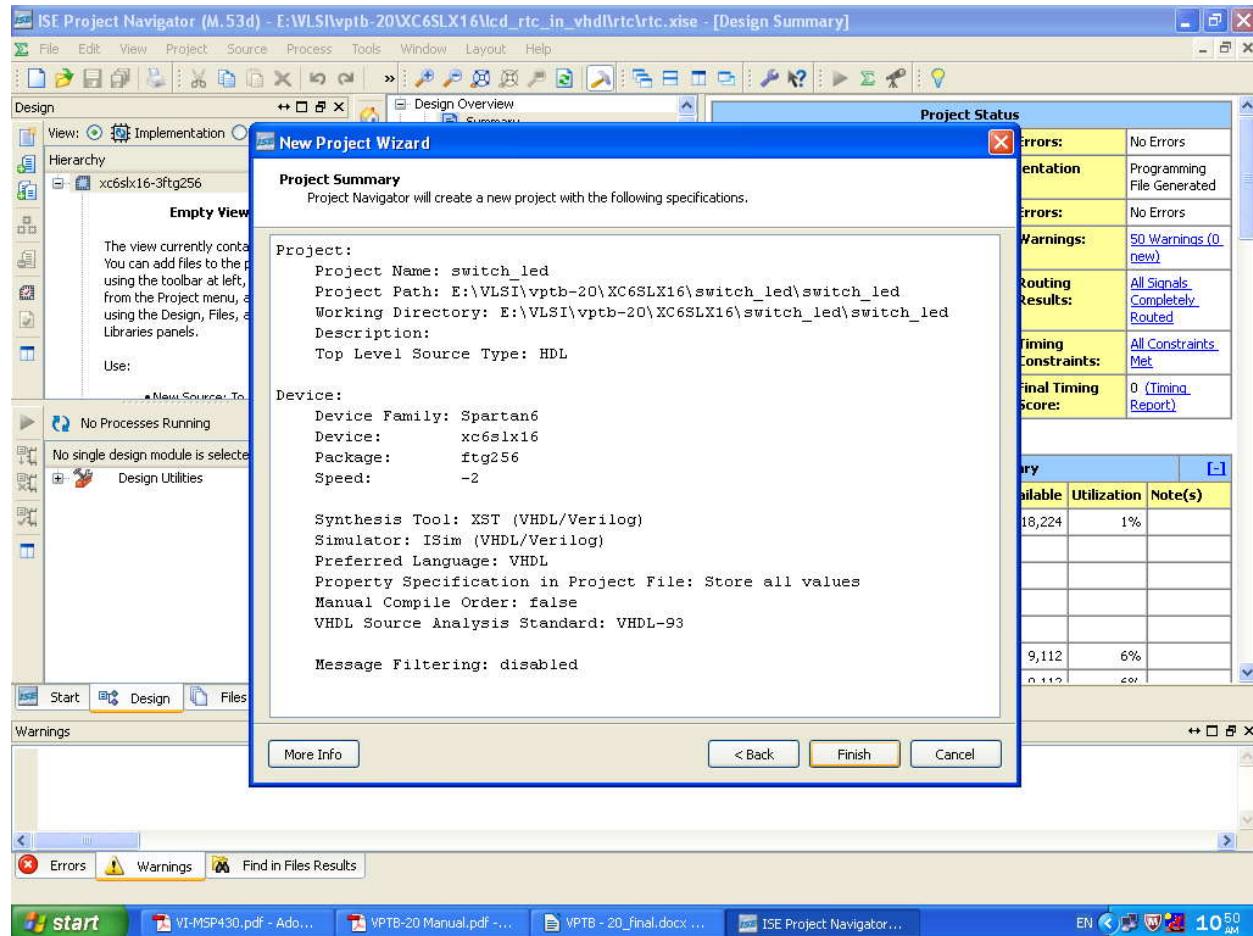


Figure-7

SPARTAN 6 TRAINER KIT

5. A Window given in **Figure-8** will appear. Select **Project → New source**, a window given in **Figure-9,10** will appear. Then select **VHDL module**, and specify the file name in appropriate field(refer **Figure-11**). Click **Next**, then if you want you can give inputs & outputs in the appropriate positions as then in the **Figure - 12**. You can also skip these information by simply clicking **Next** button and click **Finish** (refer **Figure - 13**).

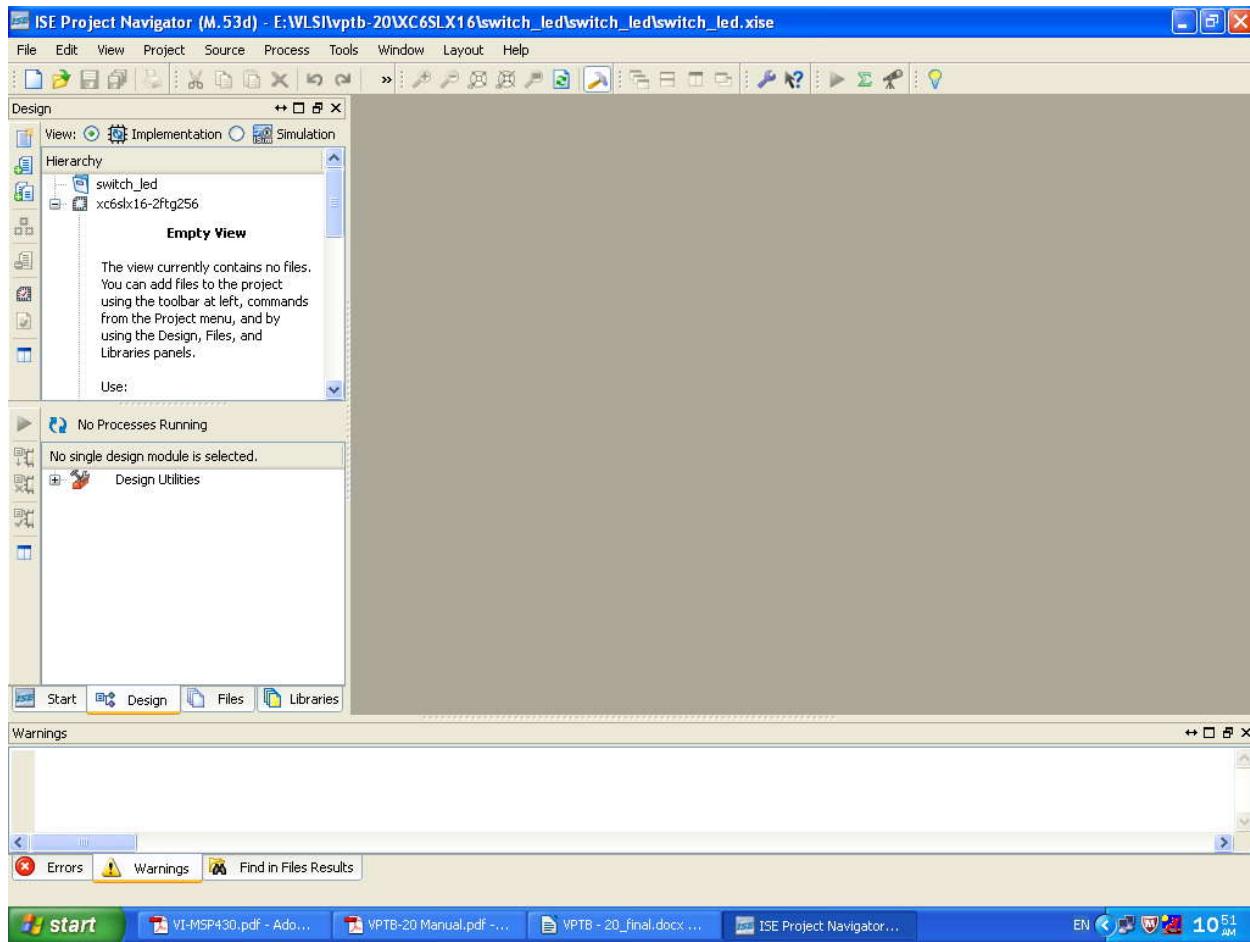


Figure-8

SPARTAN 6 TRAINER KIT

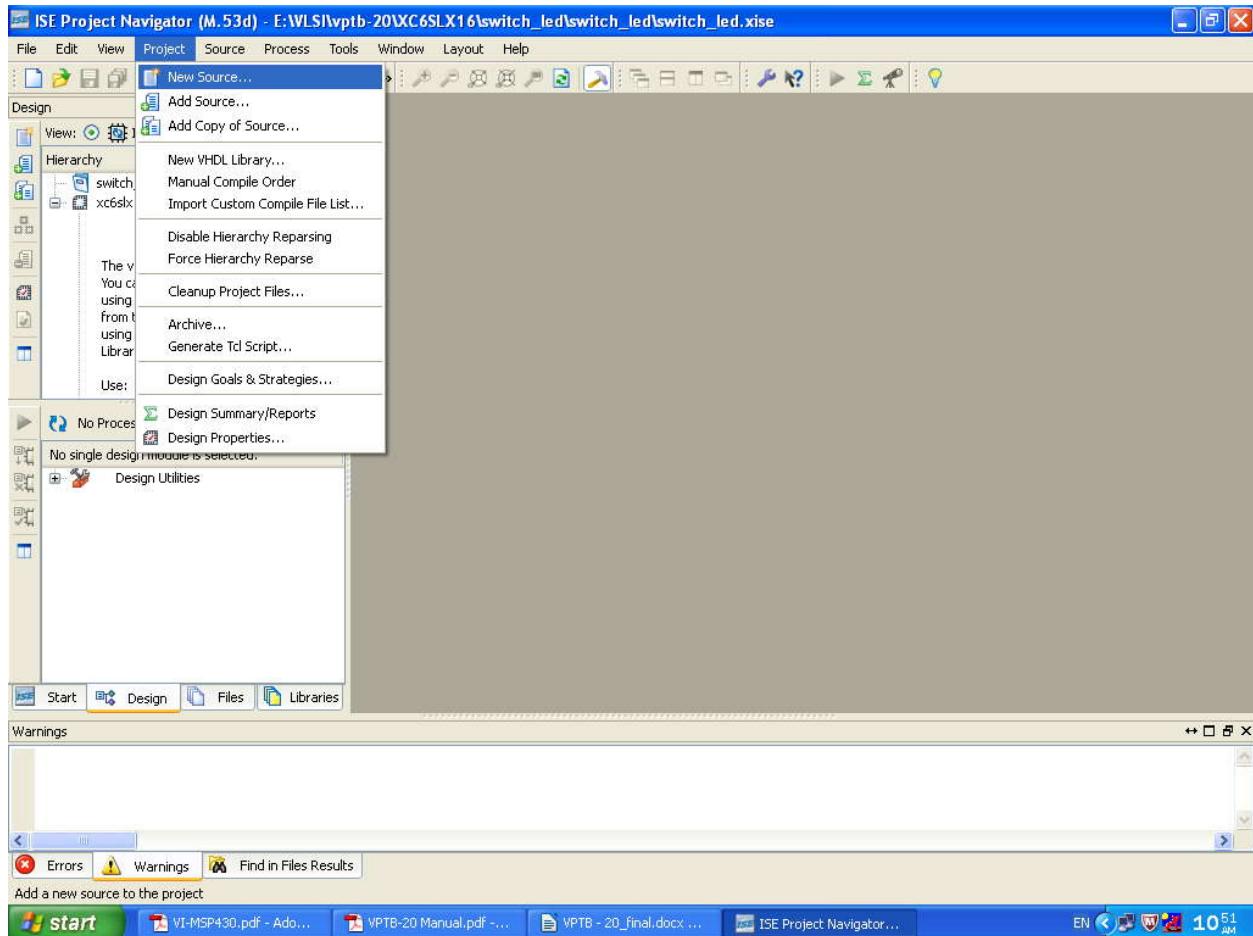


Figure-9

SPARTAN 6 TRAINER KIT

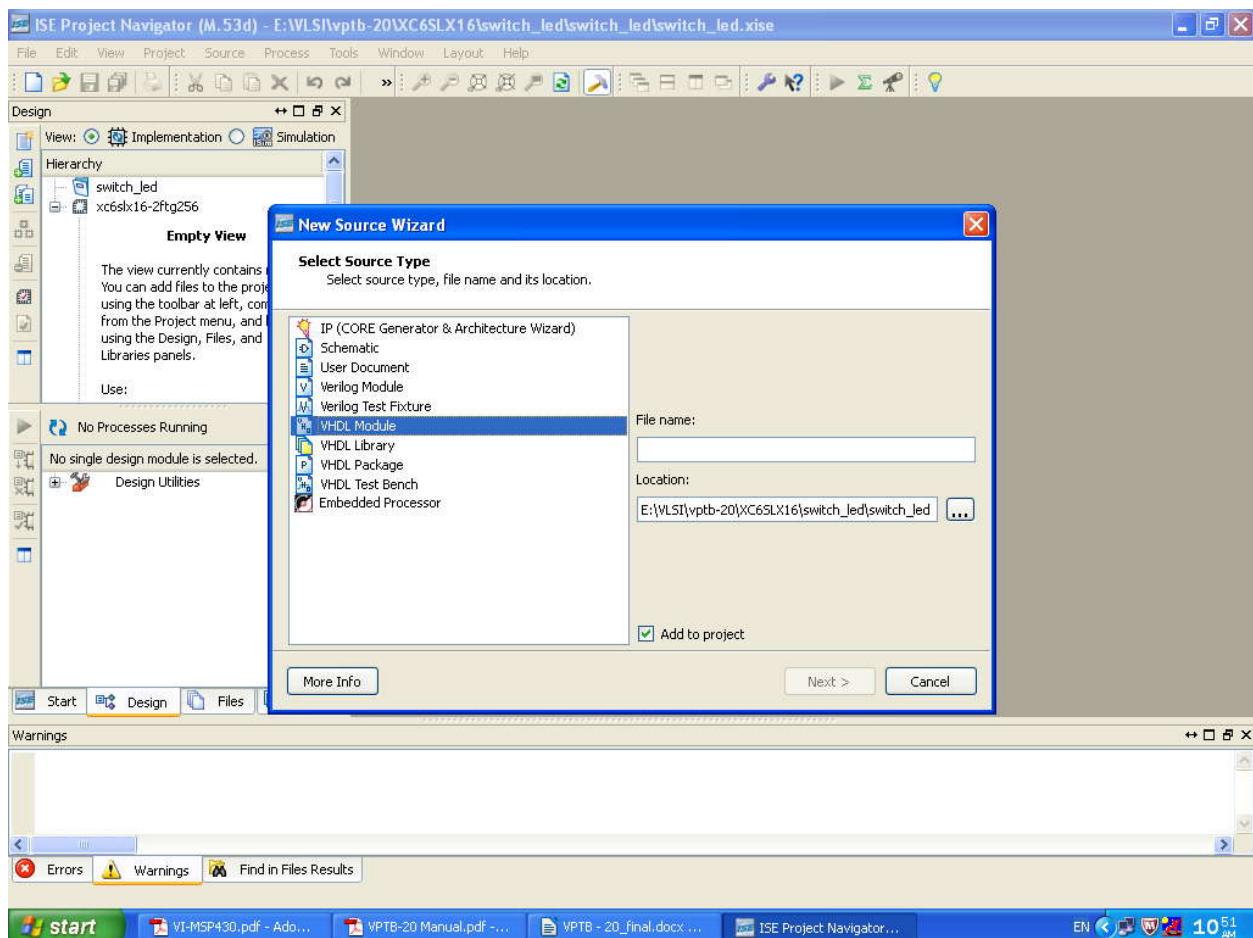


Figure-10

SPARTAN 6 TRAINER KIT

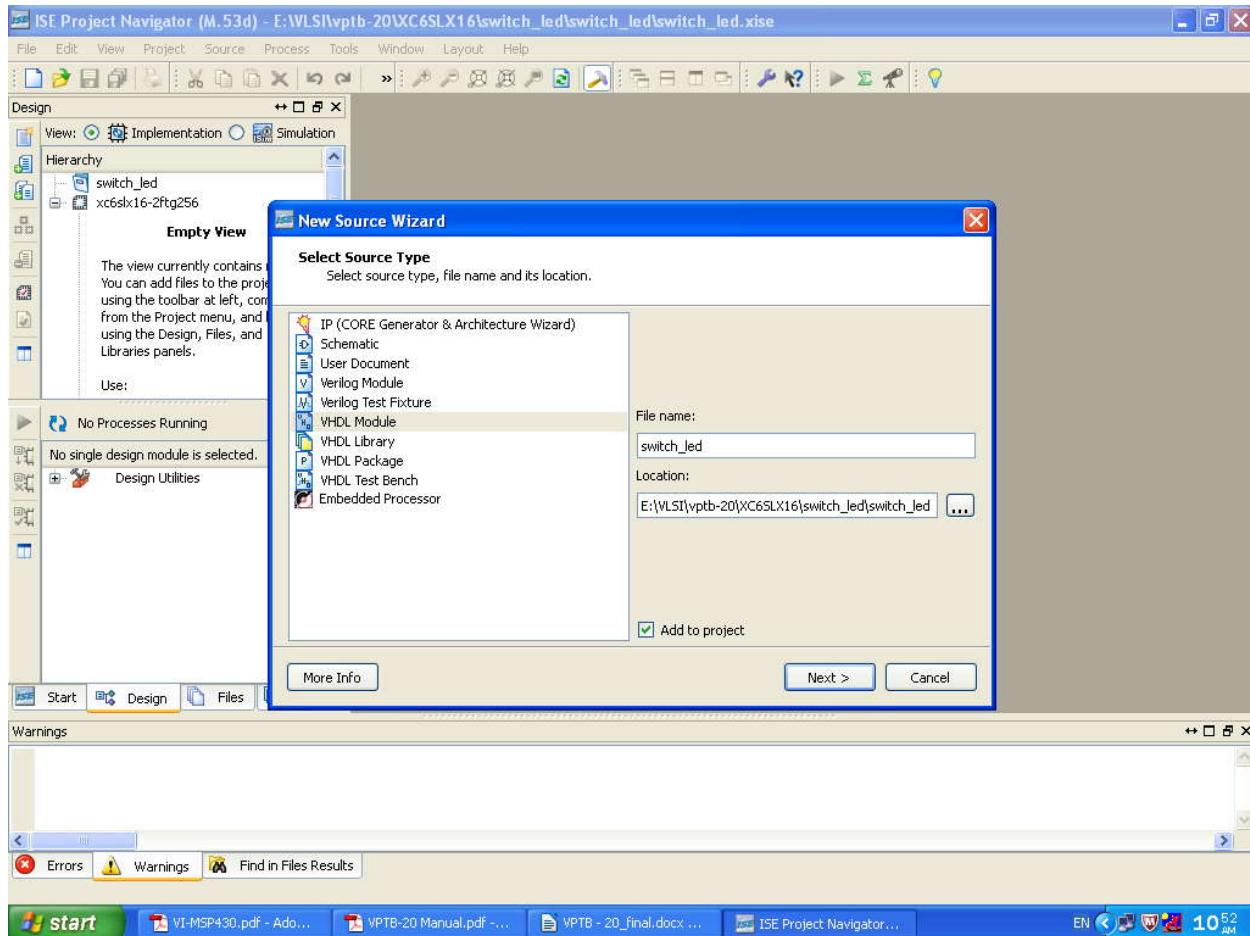


Figure-11

SPARTAN 6 TRAINER KIT

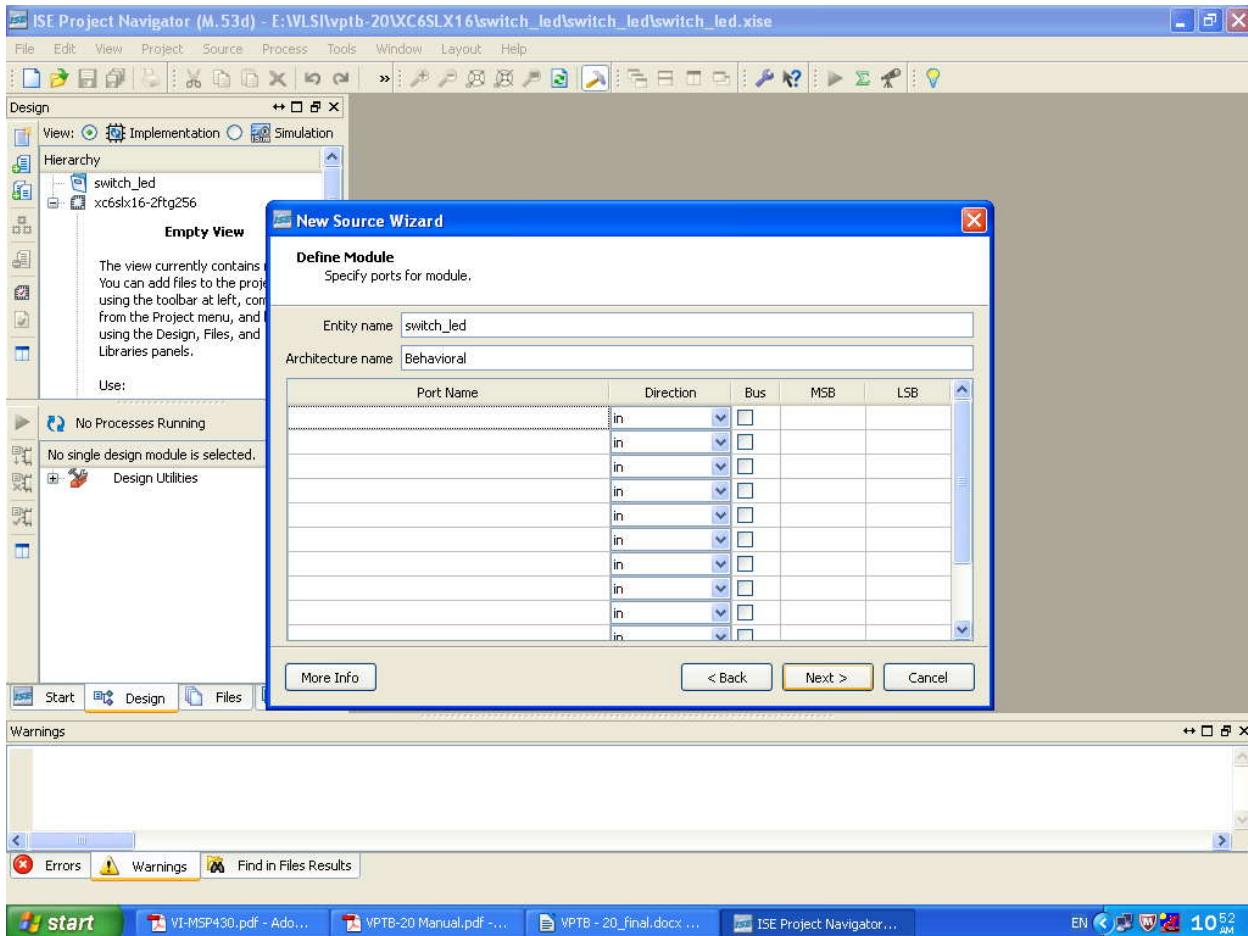


Figure-12

SPARTAN 6 TRAINER KIT

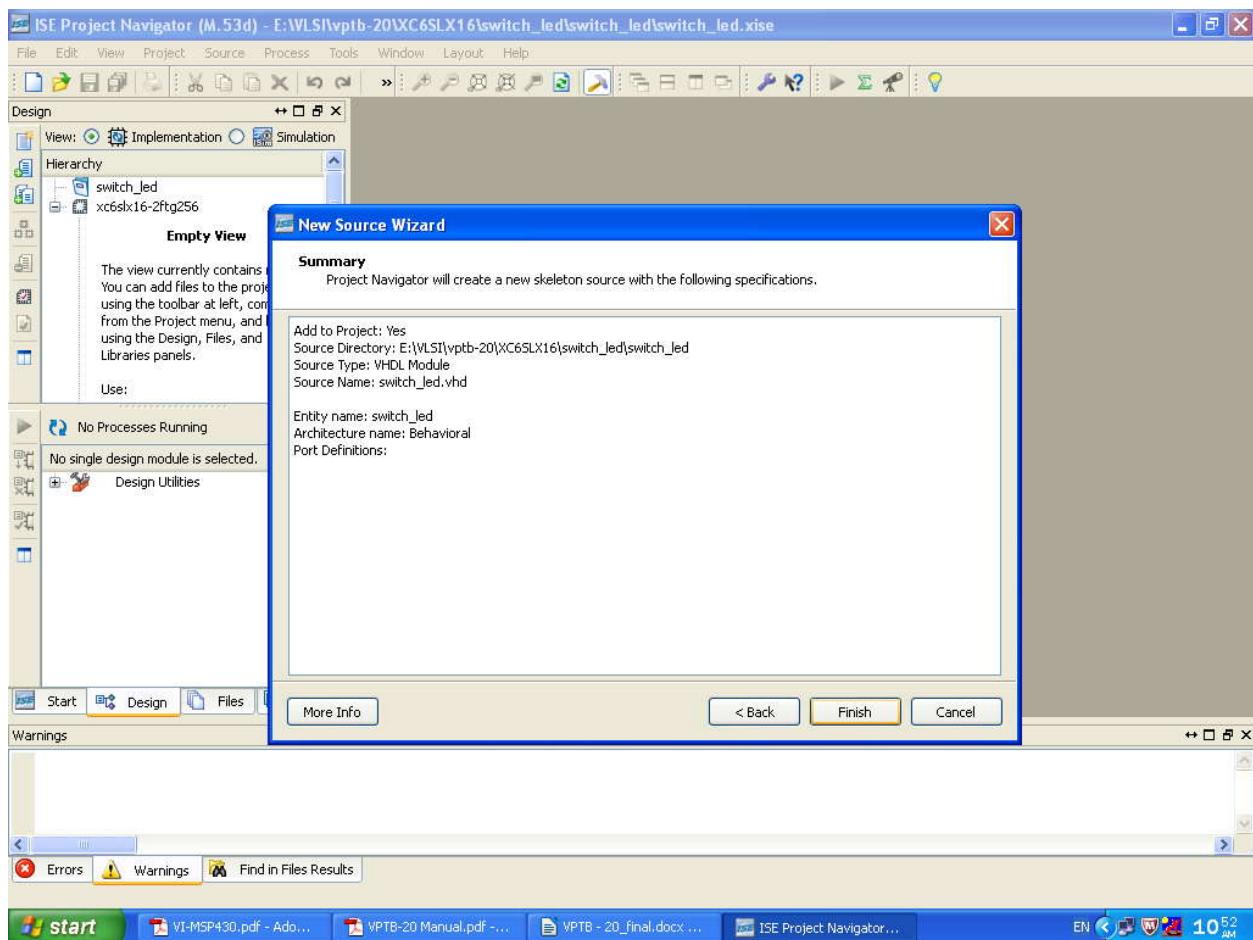


Figure-13

SPARTAN 6 TRAINER KIT

6. A window given in **Figure-14** will appear. You can type another library functions

use IEEE.STD_LOGIC_ARITH.ALL;

use IEEE.STD_LOGIC_UNSIGNED.ALL; under the library function

use IEEE.STD_LOGIC_1164.ALL;

refer ISE Window (refer **Figure-15**).

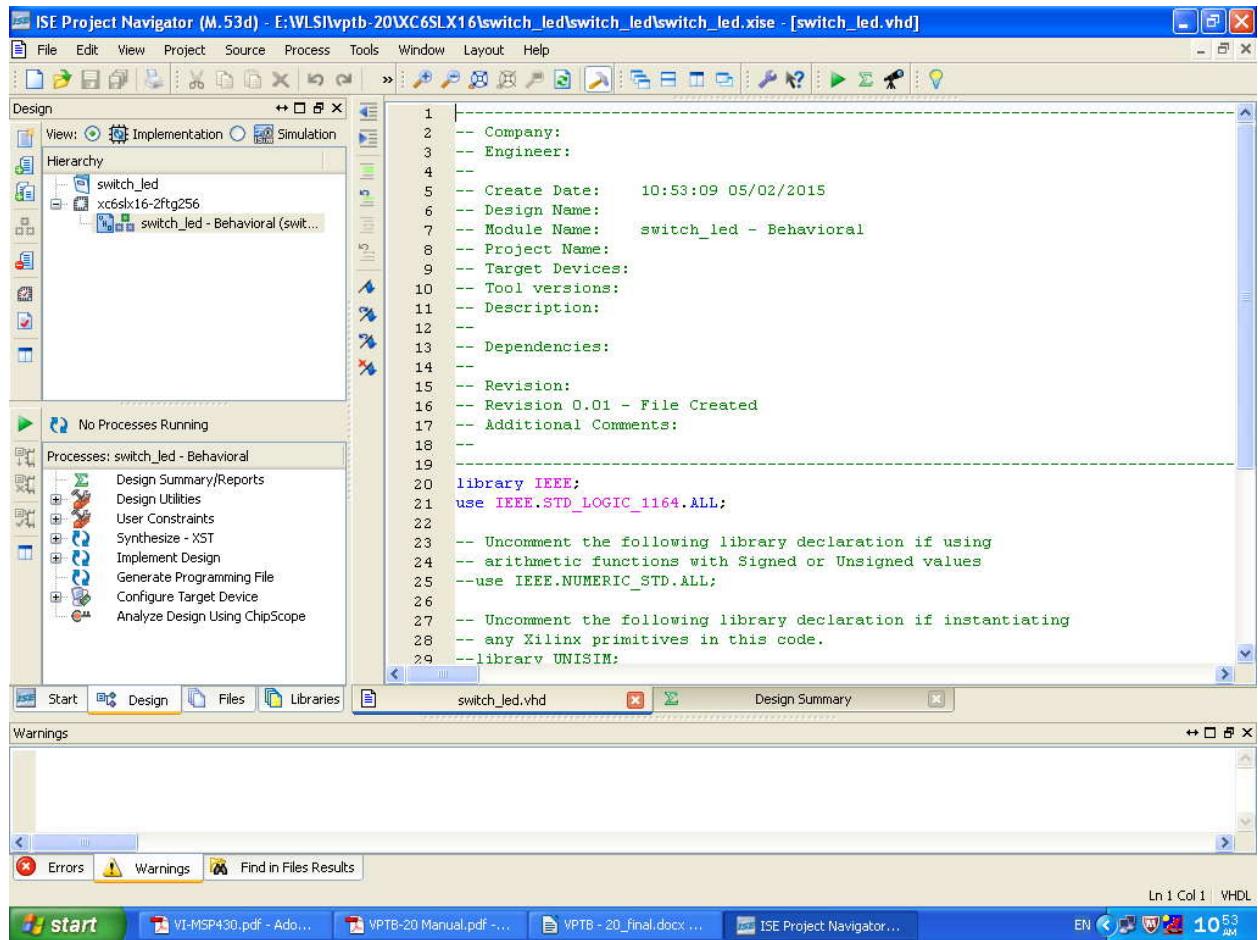


Figure-14

SPARTAN 6 TRAINER KIT

7. You can type your VHDL code in the right side of window and save the code by pressing the save button in the tool bar of the ISE window (refer **Figure-16**).

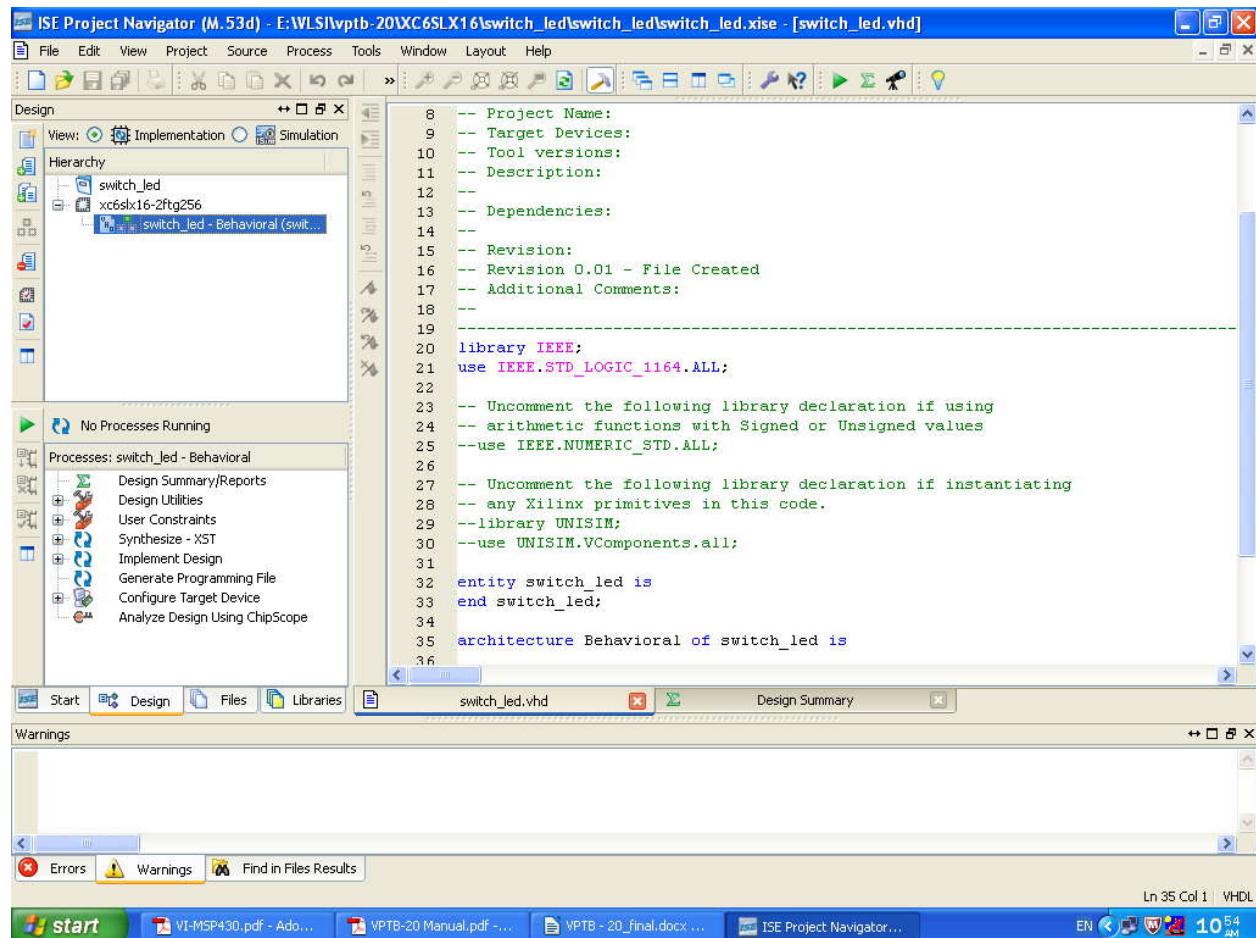


Figure-15

SPARTAN 6 TRAINER KIT

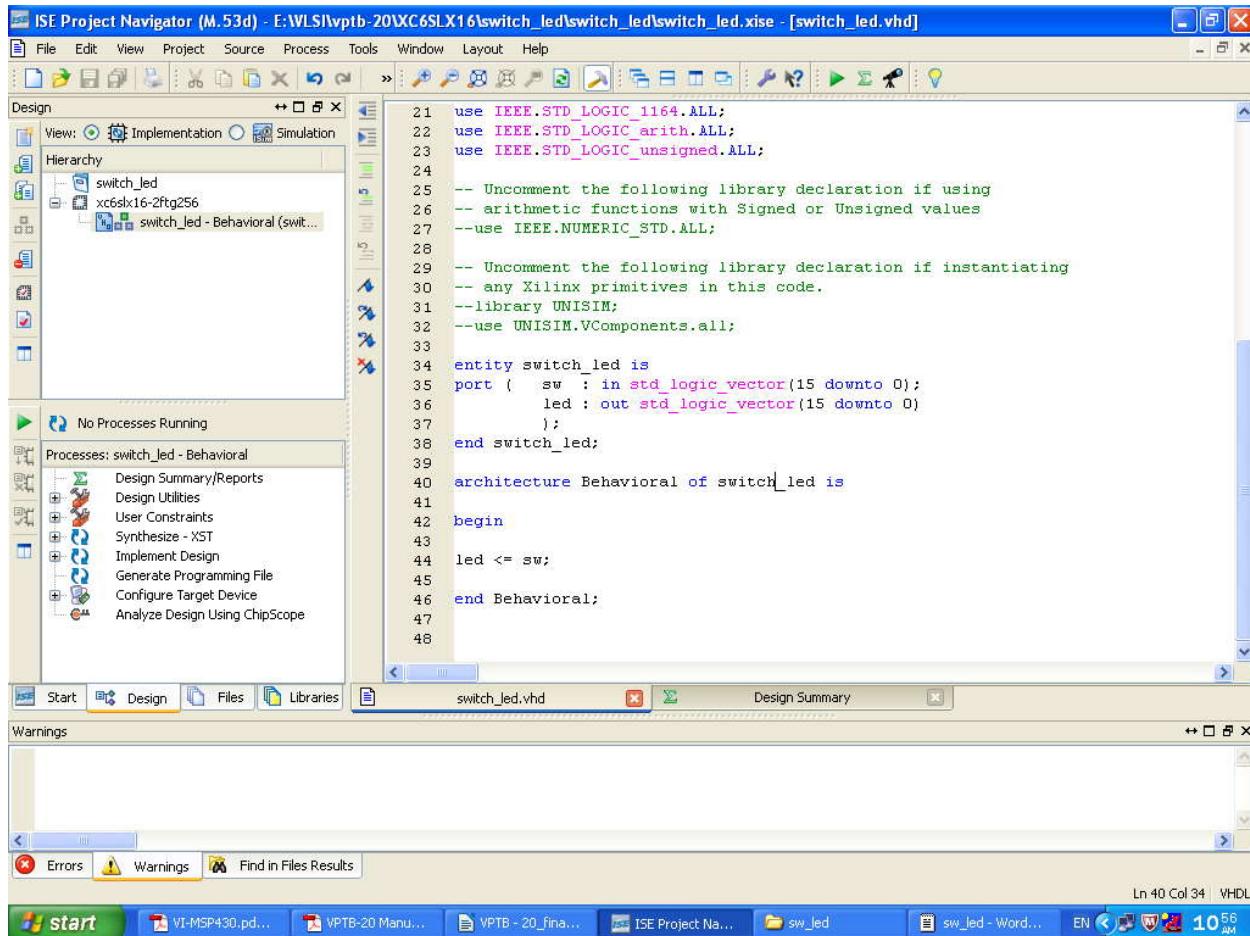


Figure-16

SPARTAN 6 TRAINER KIT

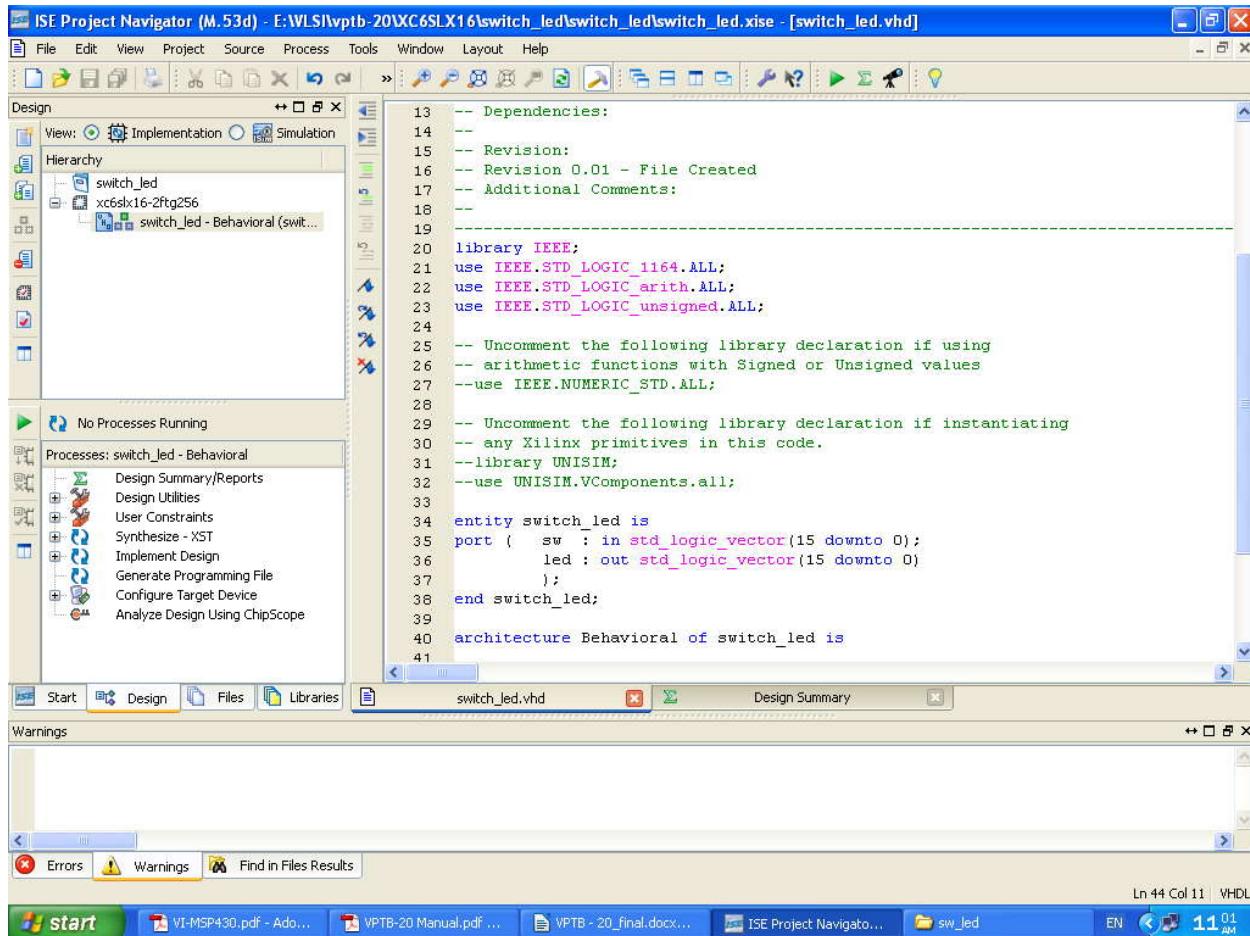


Figure-17

SPARTAN 6 TRAINER KIT

8. Then select **Project → New Source → Implementation Constraint File** (refer Figures- 18). Give filename in the appropriate field and then click **next** and **Finish**. (Refer Figures-19, 20, 21).

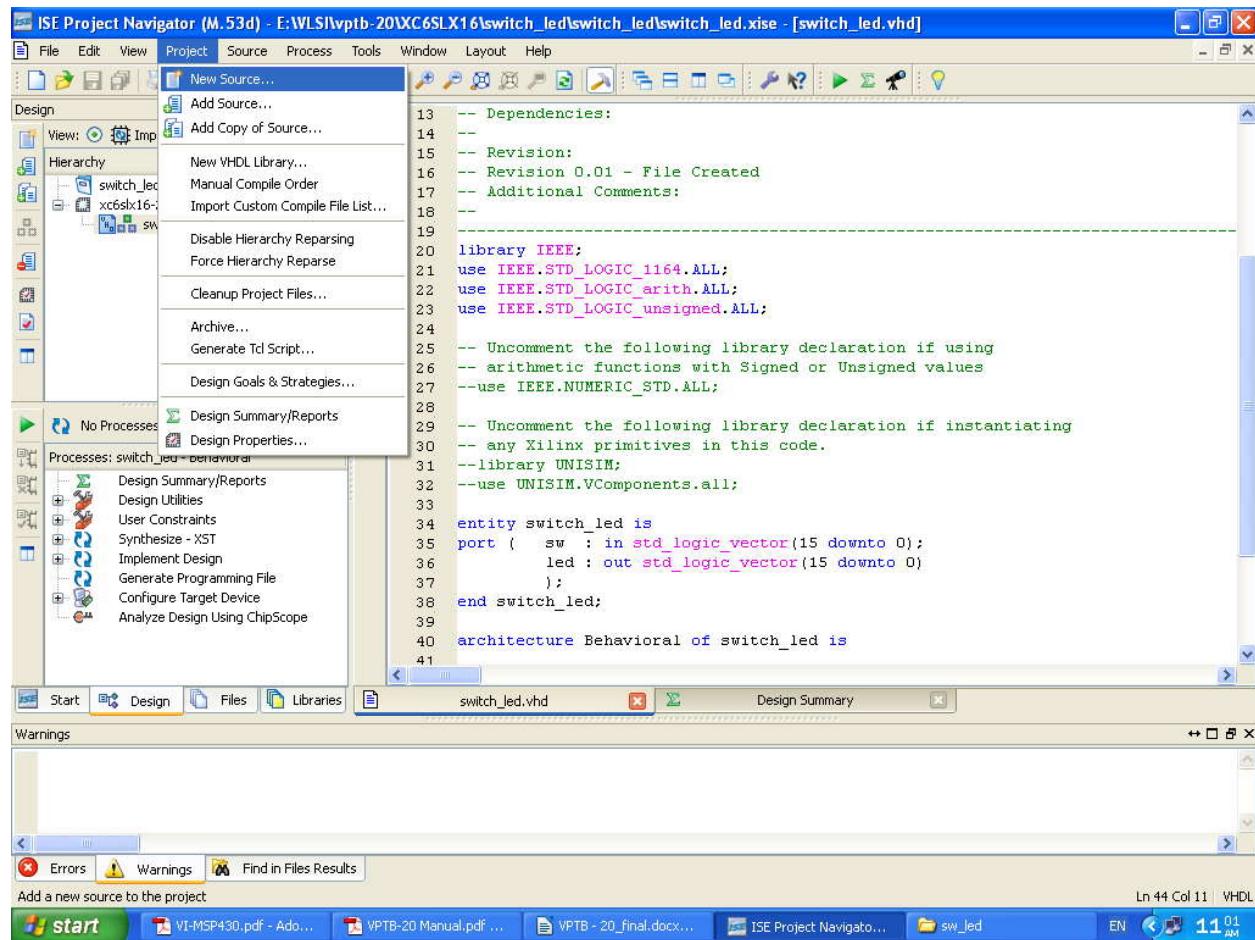


Figure-18

SPARTAN 6 TRAINER KIT

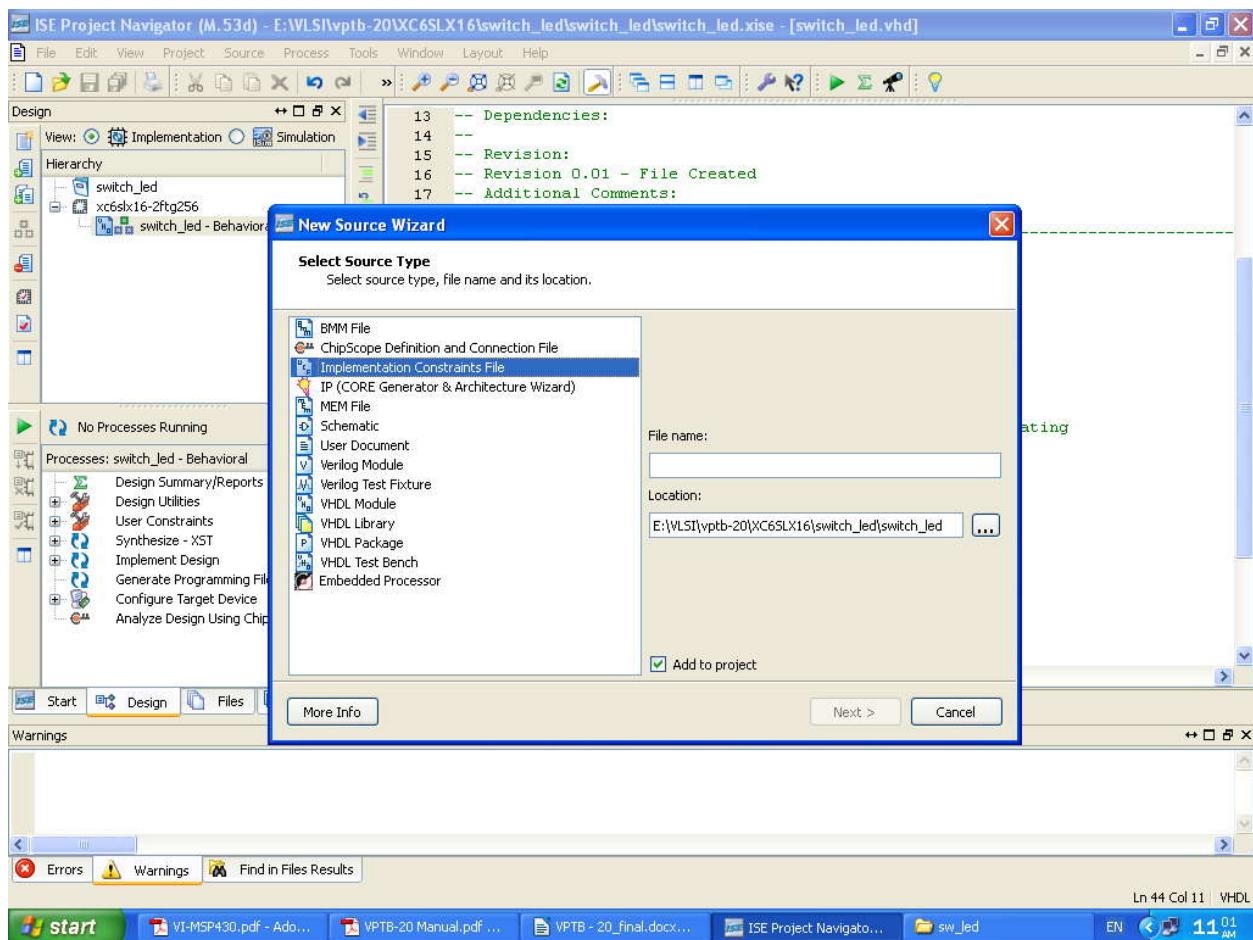


Figure-19

SPARTAN 6 TRAINER KIT

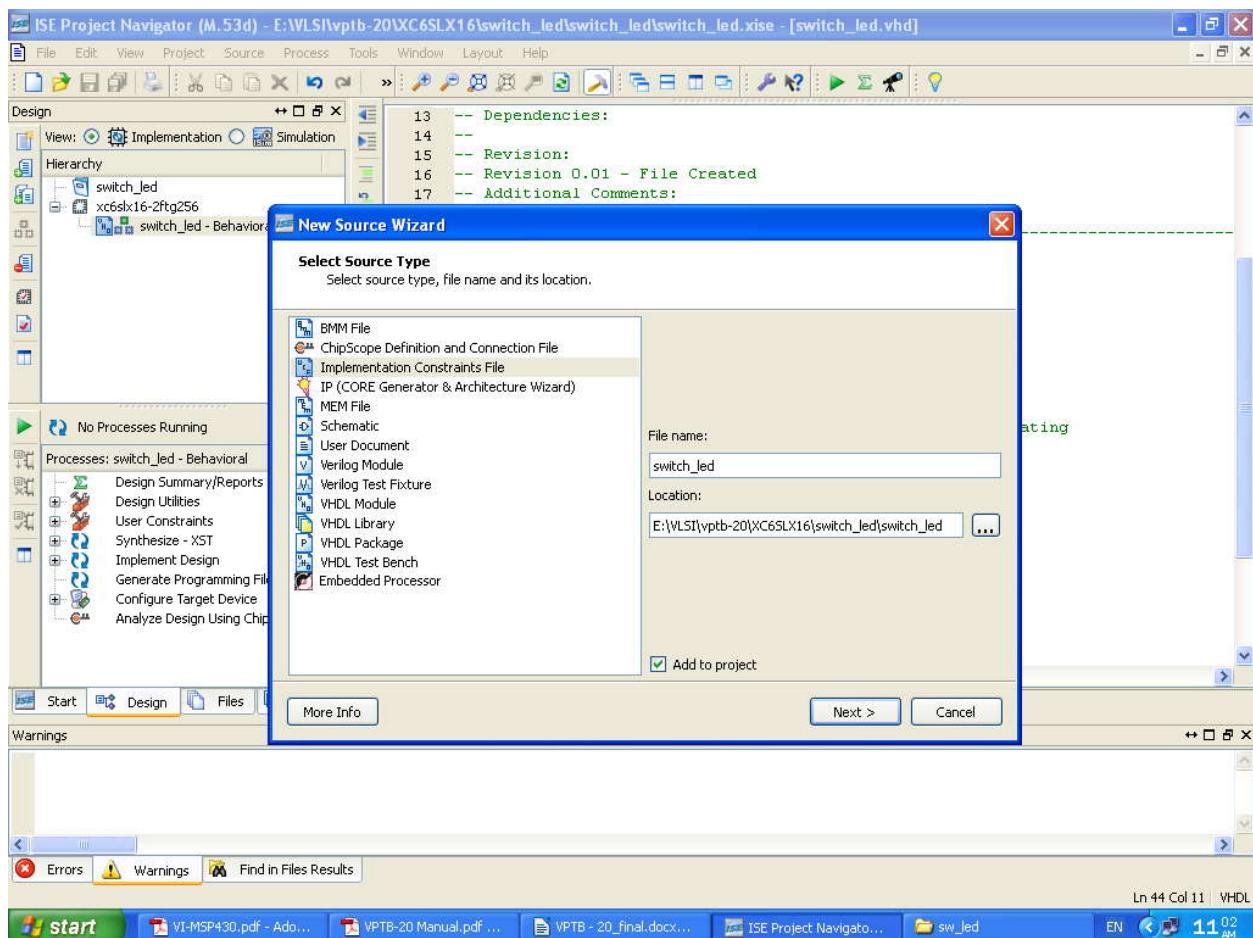


Figure-20

SPARTAN 6 TRAINER KIT

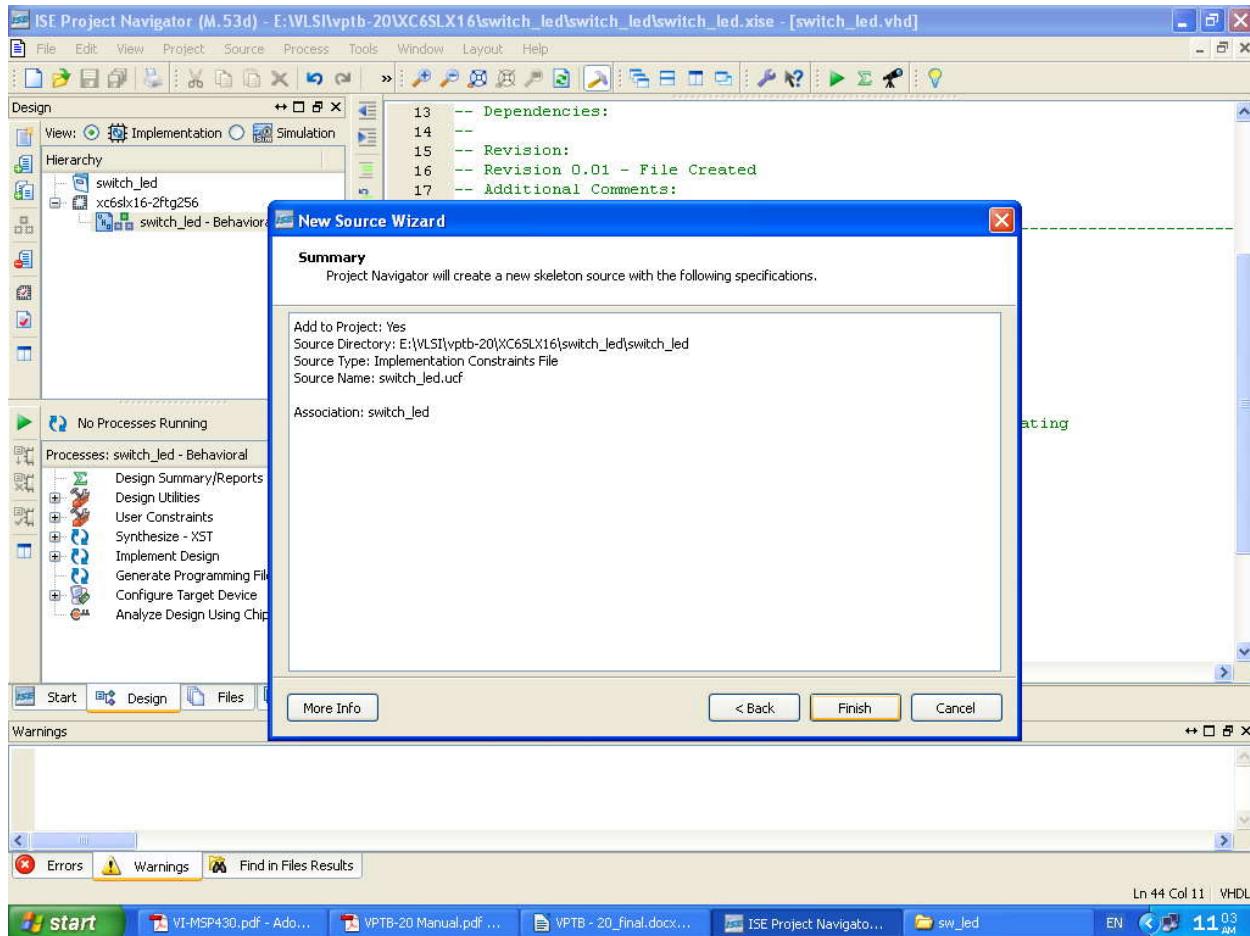


Figure-21

SPARTAN 6 TRAINER KIT

9. Click the UCF file in source window. In process window under **User constraint** double click the **Edit constraints (text)** double click (refer **Figure-23, 24**).

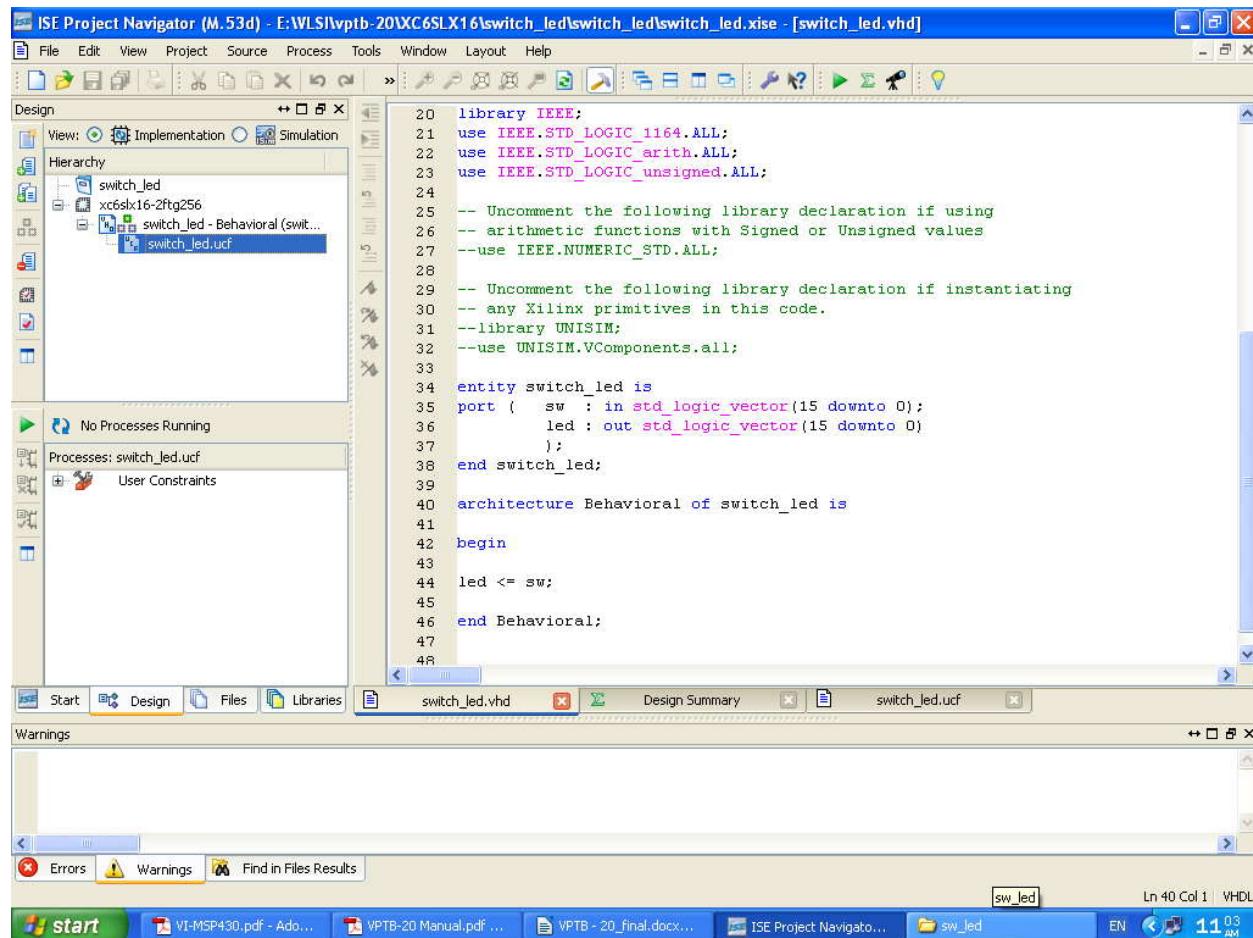


Figure-22

SPARTAN 6 TRAINER KIT

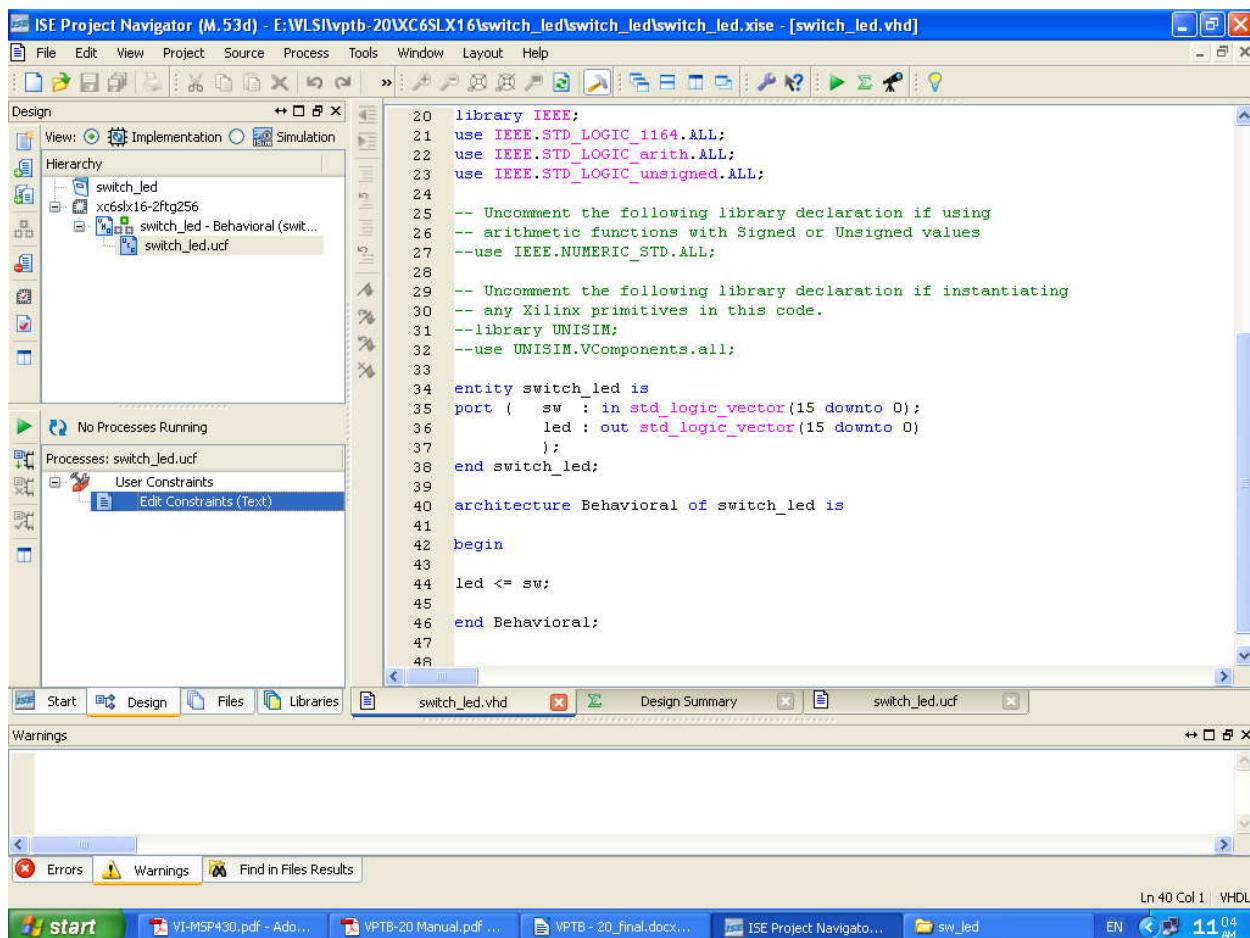


Figure-23

SPARTAN 6 TRAINER KIT

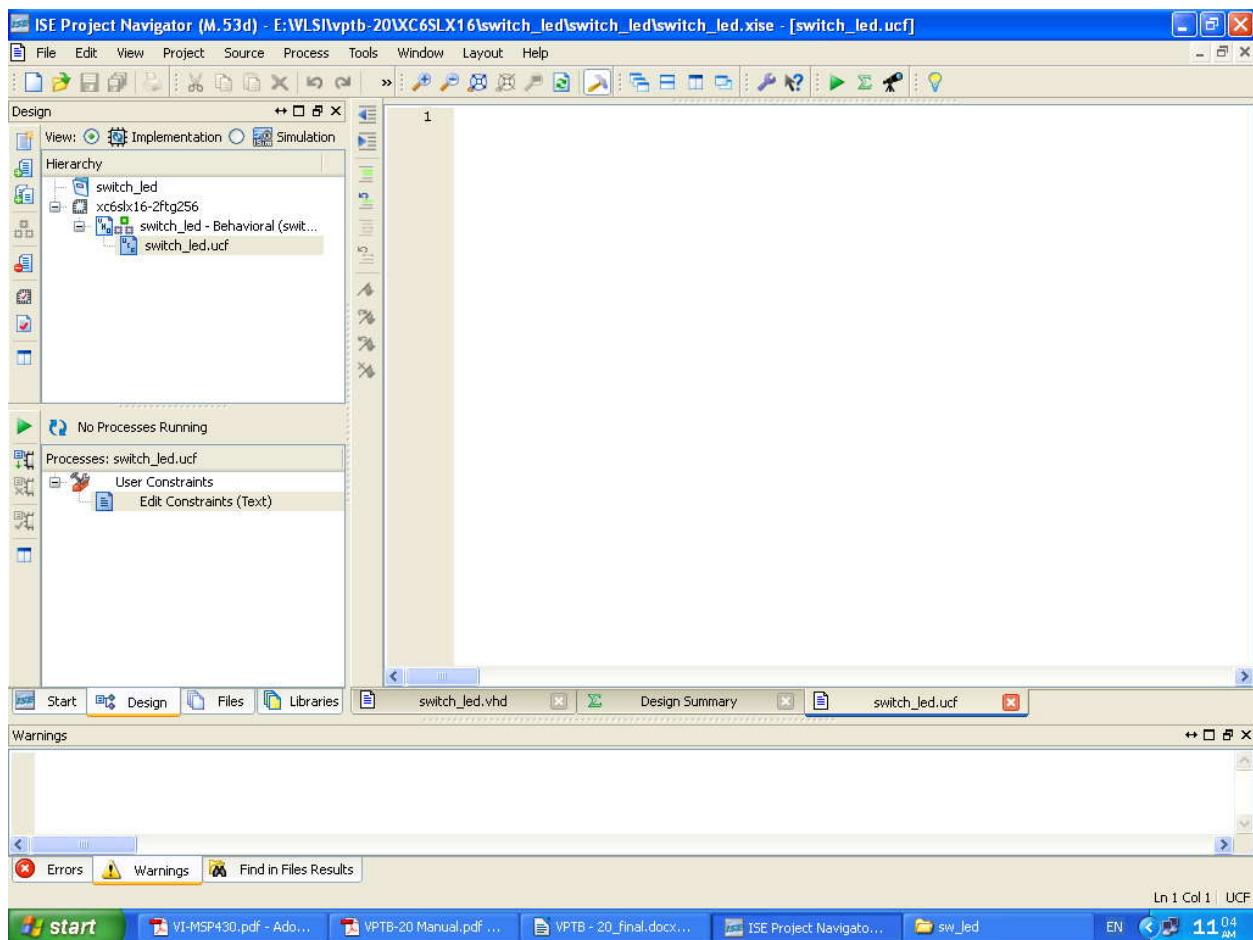


Figure-24

SPARTAN 6 TRAINER KIT

10. In this window type your pin assignments to the inputs and outputs corresponding to the FPGA pins. Refer Figure-25

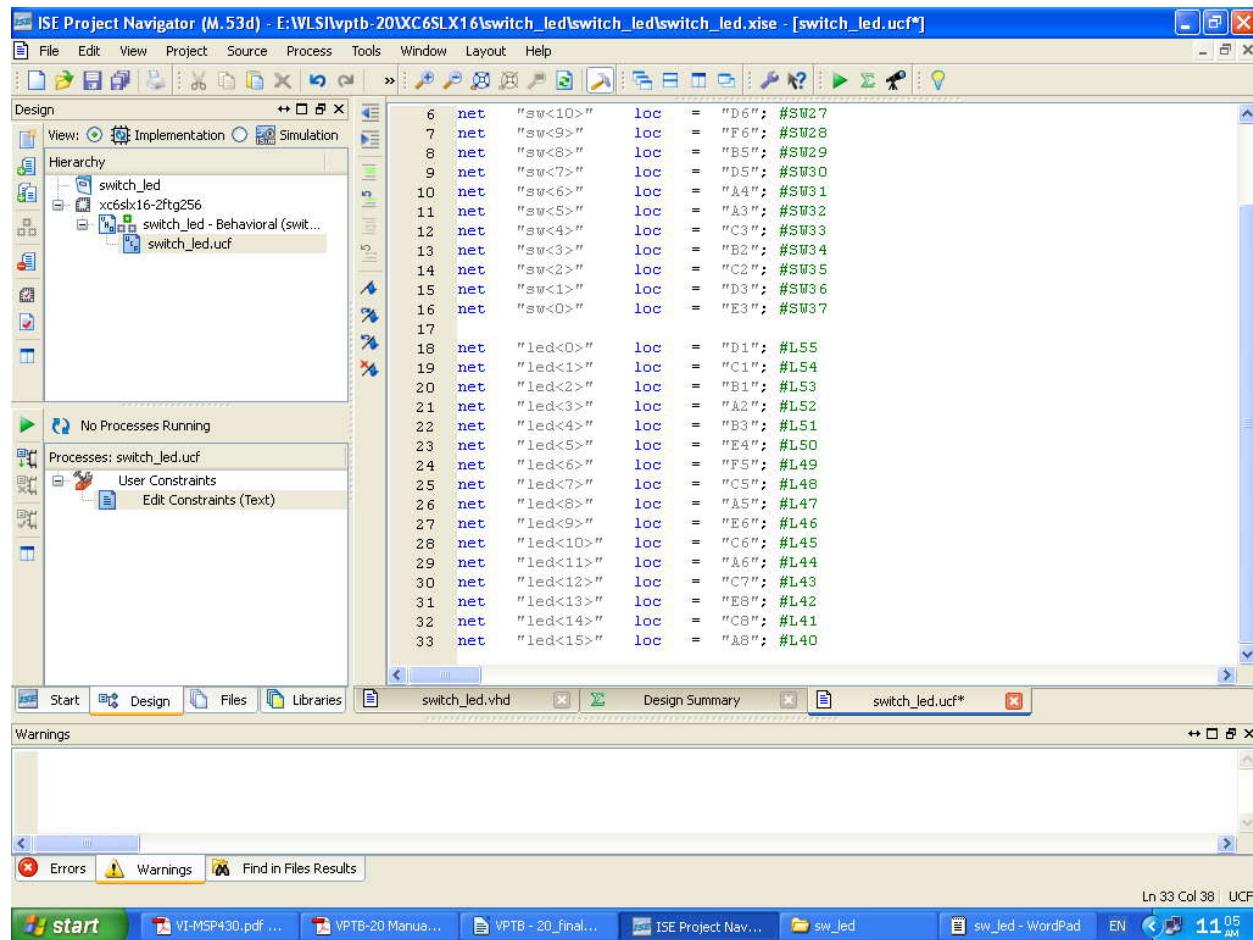


Figure-25

SPARTAN 6 TRAINER KIT

11. To download the program into **Boundary Scan** mode, do the following steps. Come to **Project Navigator** window and click **SYNTHESIZE-XST** and when synthesis is completed successfully, click **IMPLEMENT DESIGN** in process for current source. After implement design has become successful, click **GENERATE PROGRAMMING FILE** and then click **Configure Device (iMPACT)** (refer Figure-26, 27, 28, 29, 30).

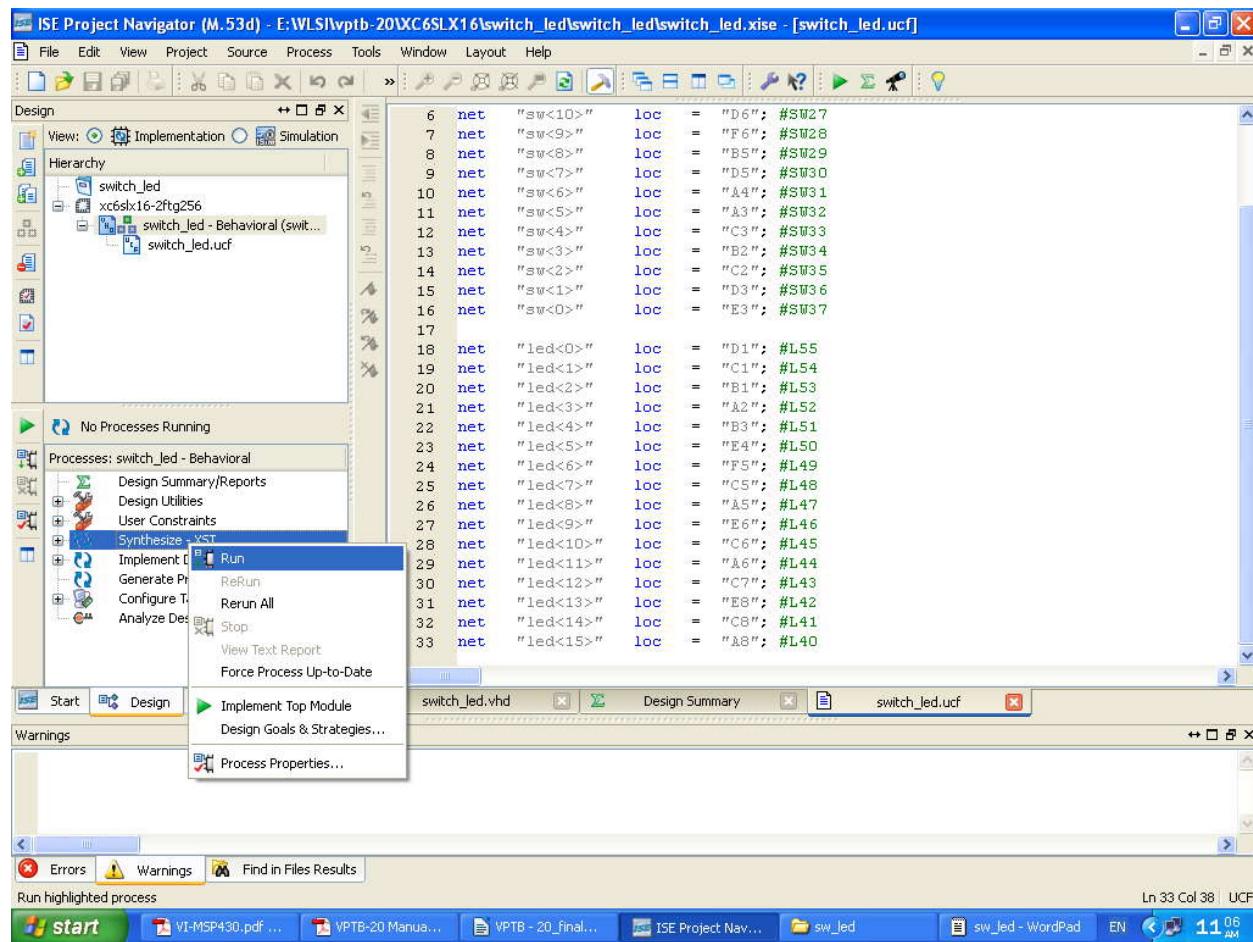


Figure-26

SPARTAN 6 TRAINER KIT

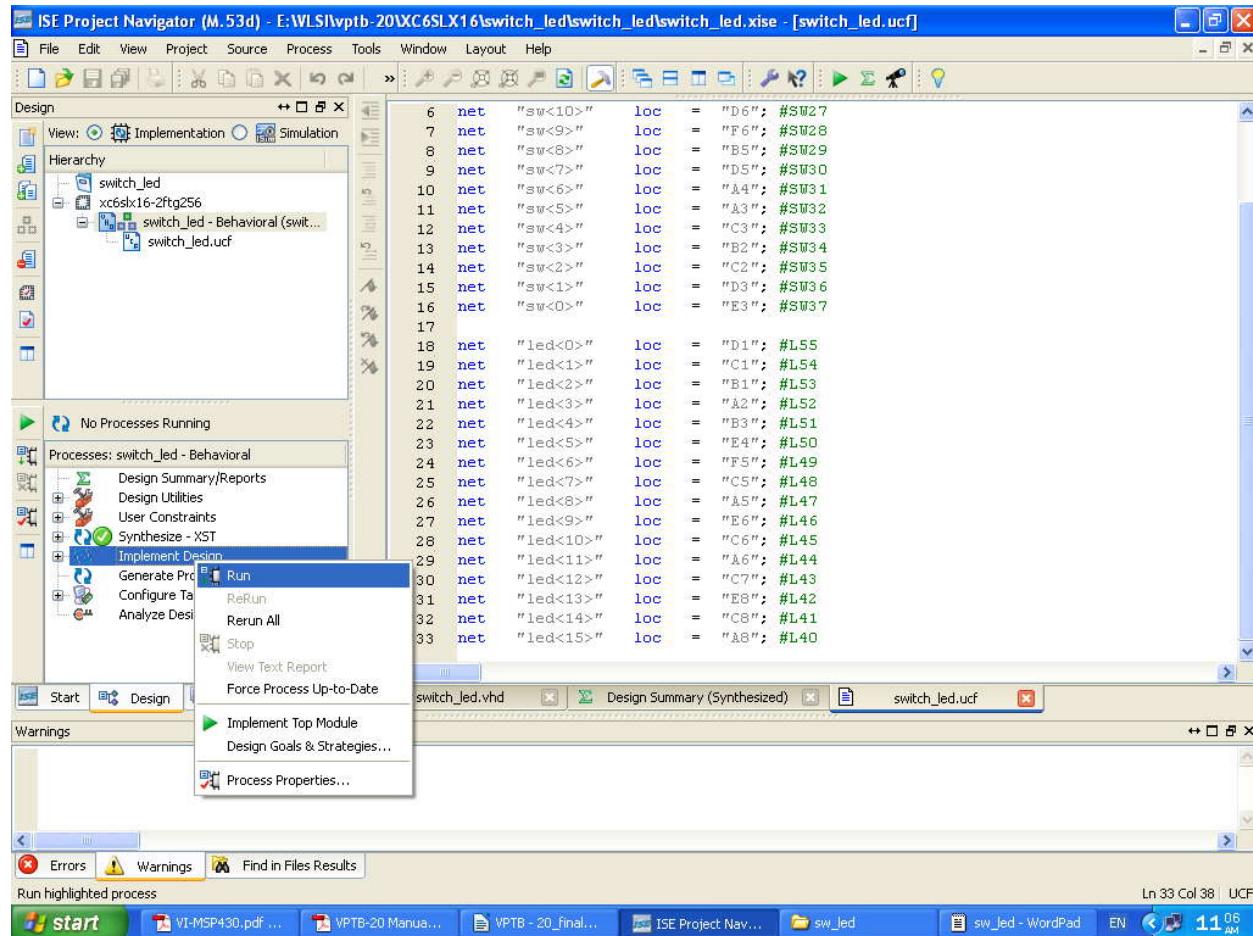


Figure-27

SPARTAN 6 TRAINER KIT

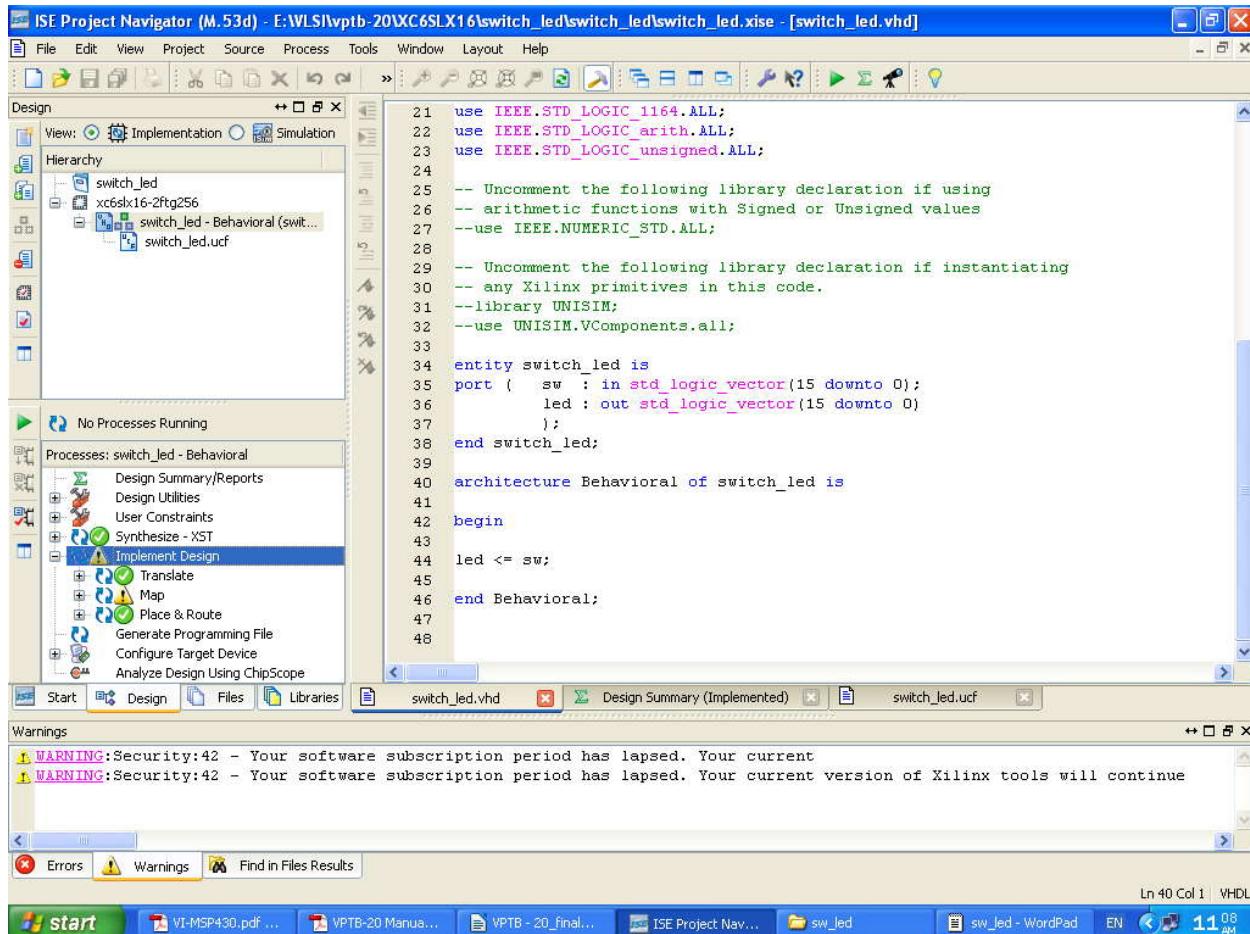


Figure-28

SPARTAN 6 TRAINER KIT

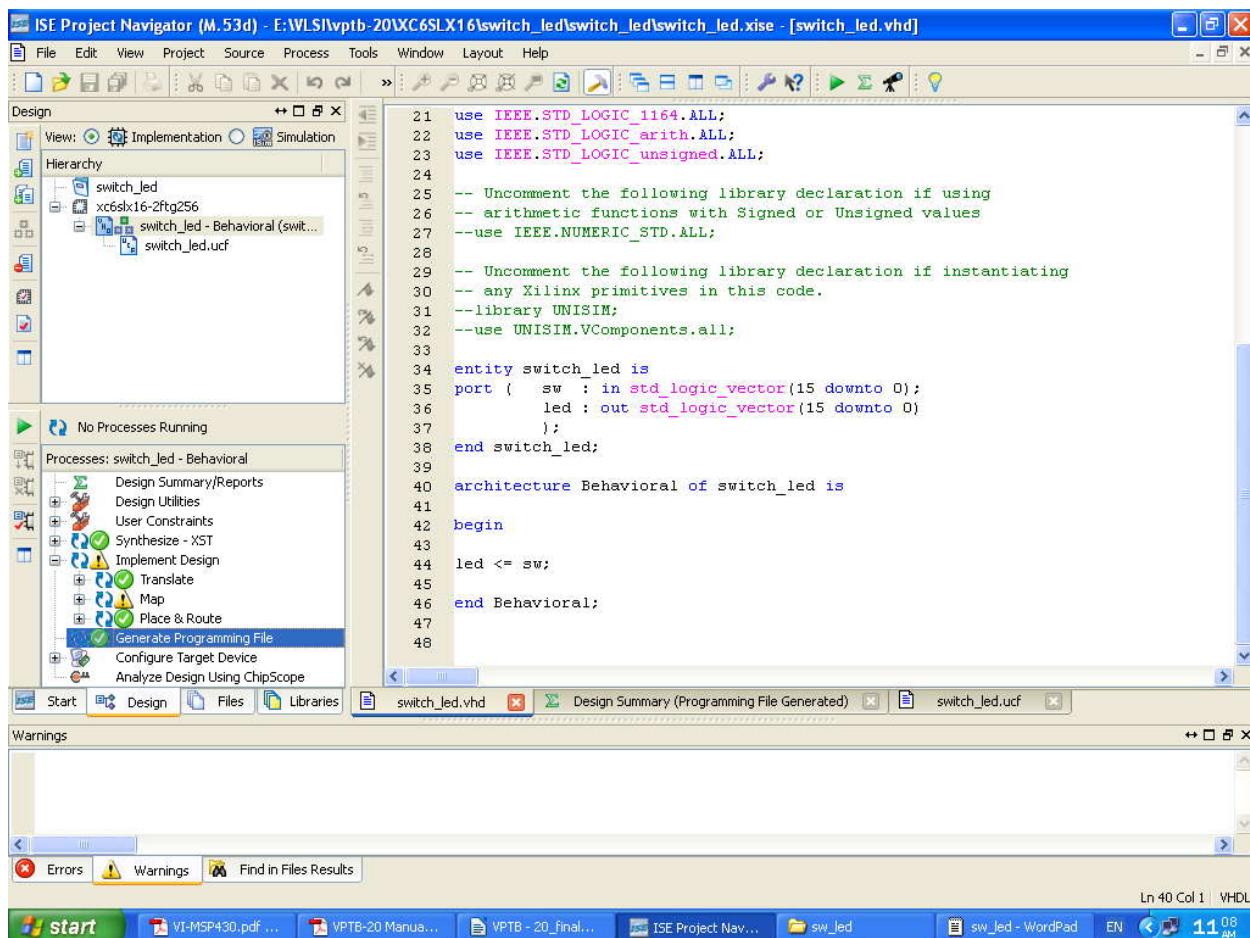


Figure-29

SPARTAN 6 TRAINER KIT

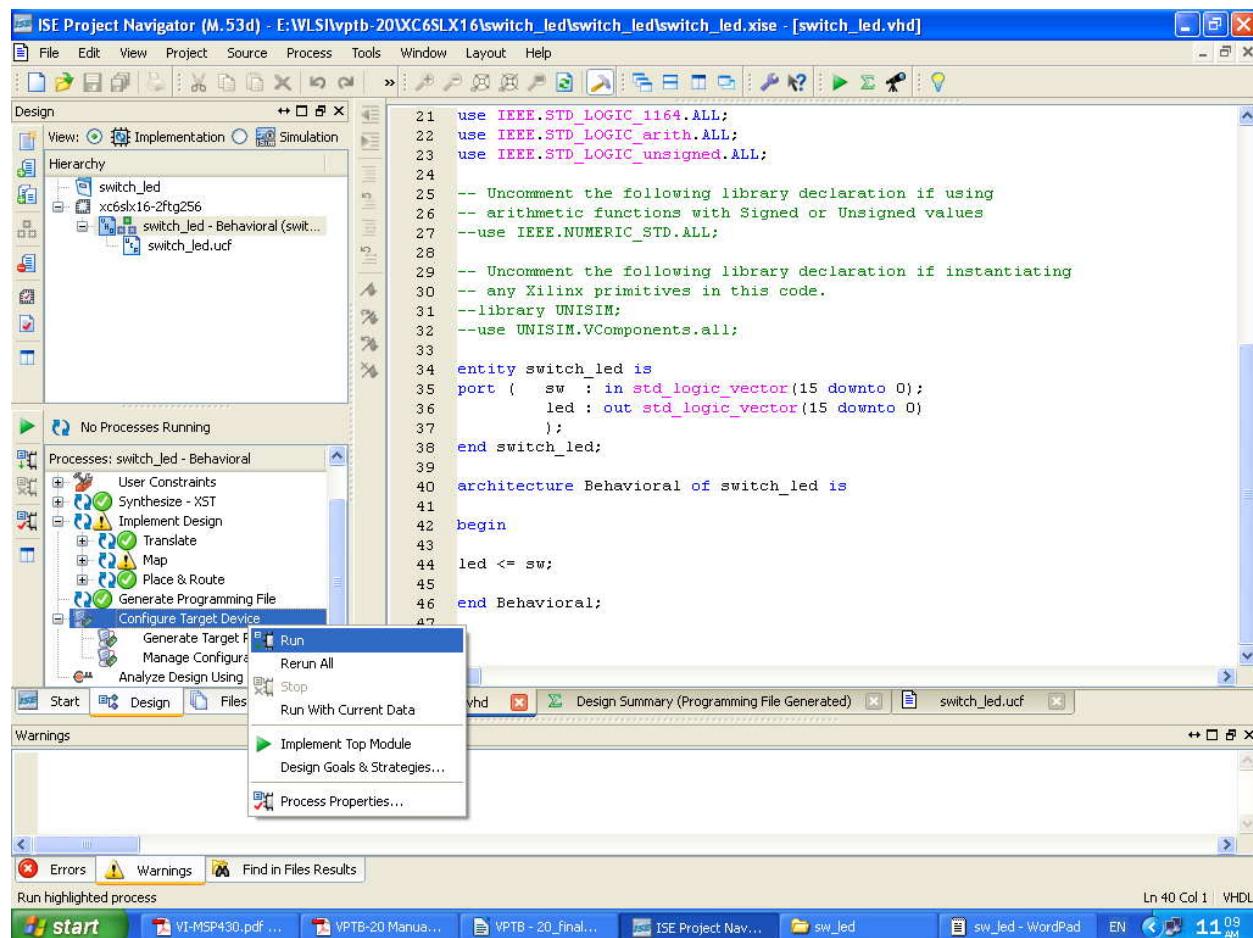


Figure-30

SPARTAN 6 TRAINER KIT

12. Click ok to continue the process. Then new impact window will open. Refer figure-31

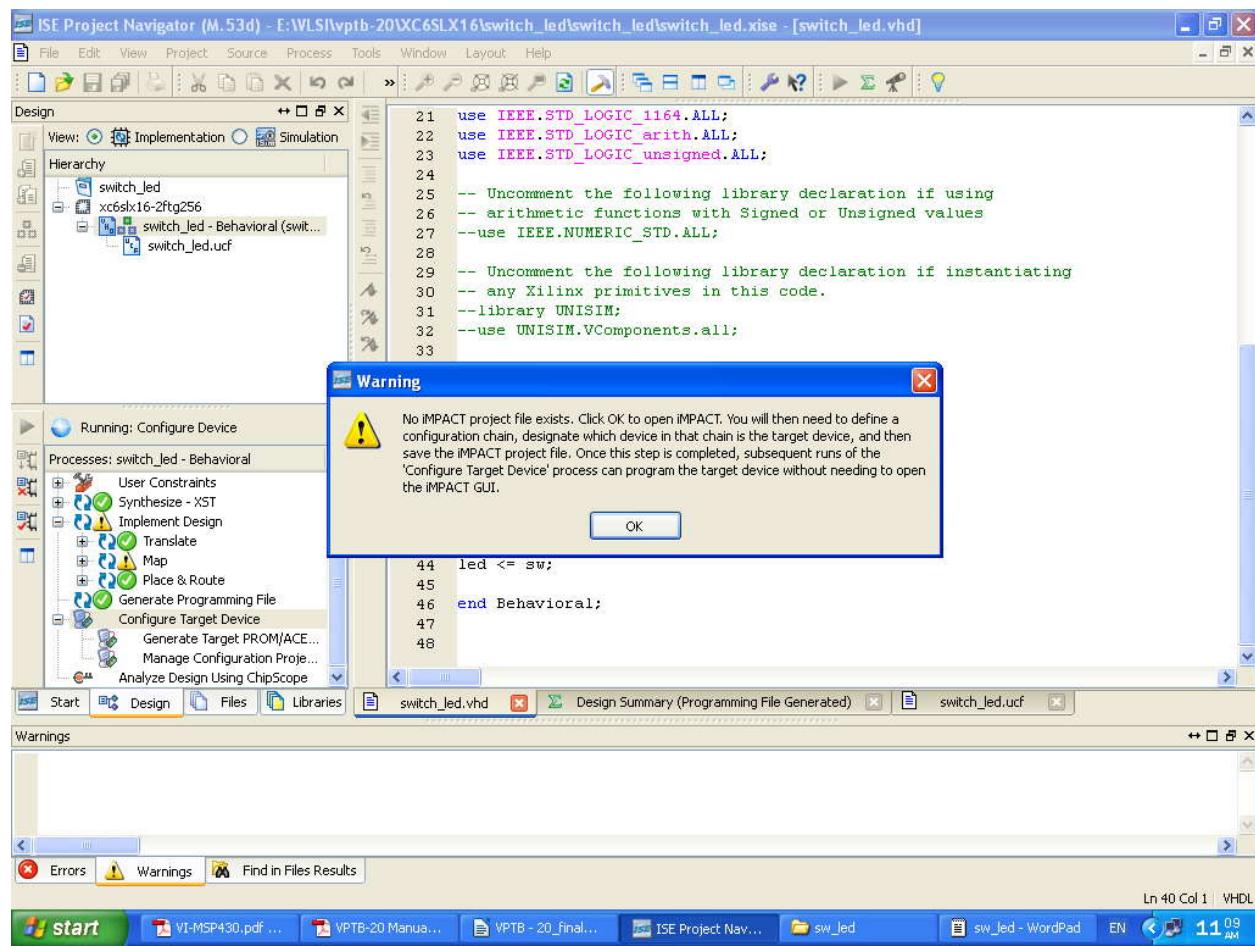


Figure-31

SPARTAN 6 TRAINER KIT

13. Now one new impact window will open. (Refer figure- 32)

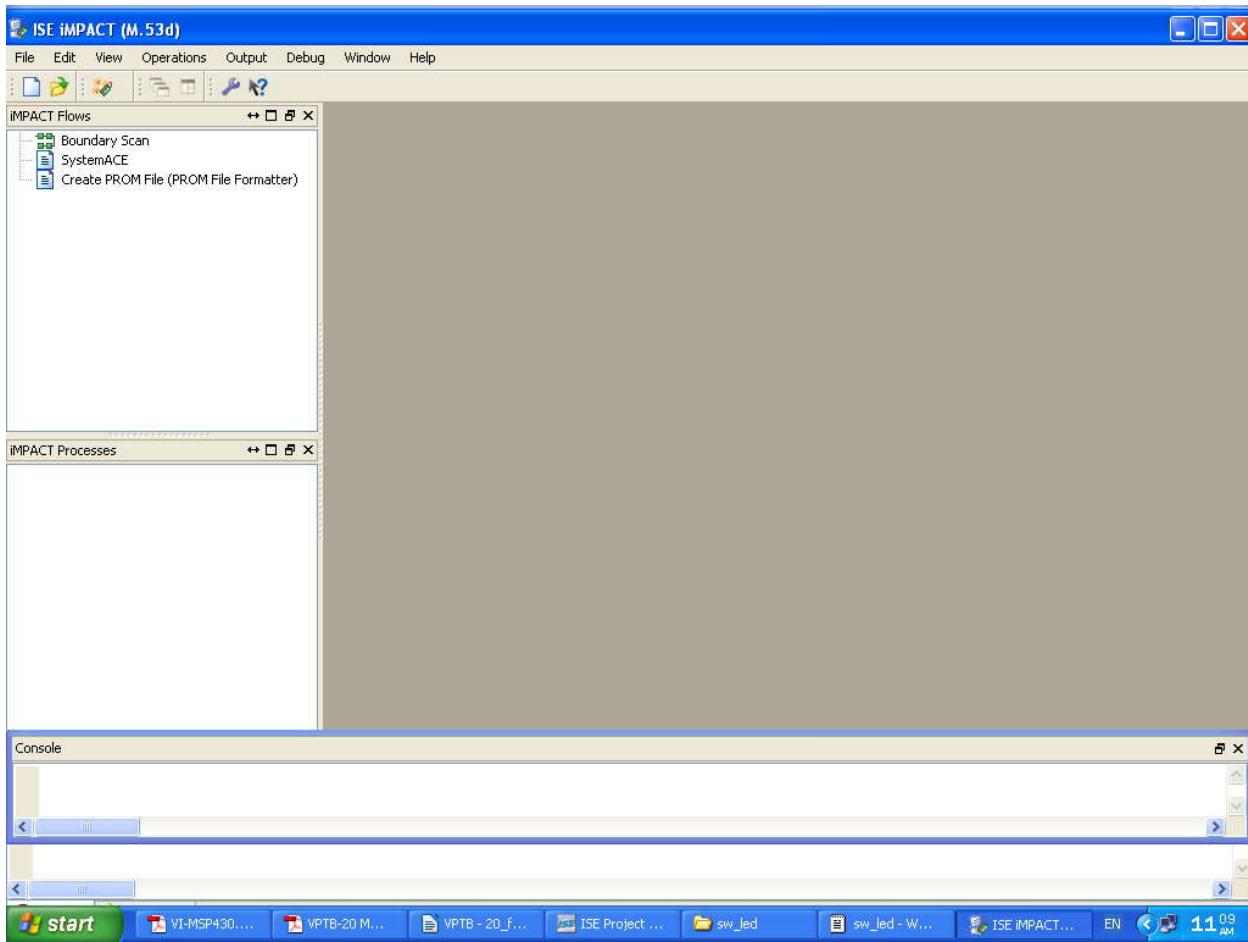


Figure-32

SPARTAN 6 TRAINER KIT

14. Double click the Boundary Scan mode, presented in the left side of the window. Refer figure- 33, 34.

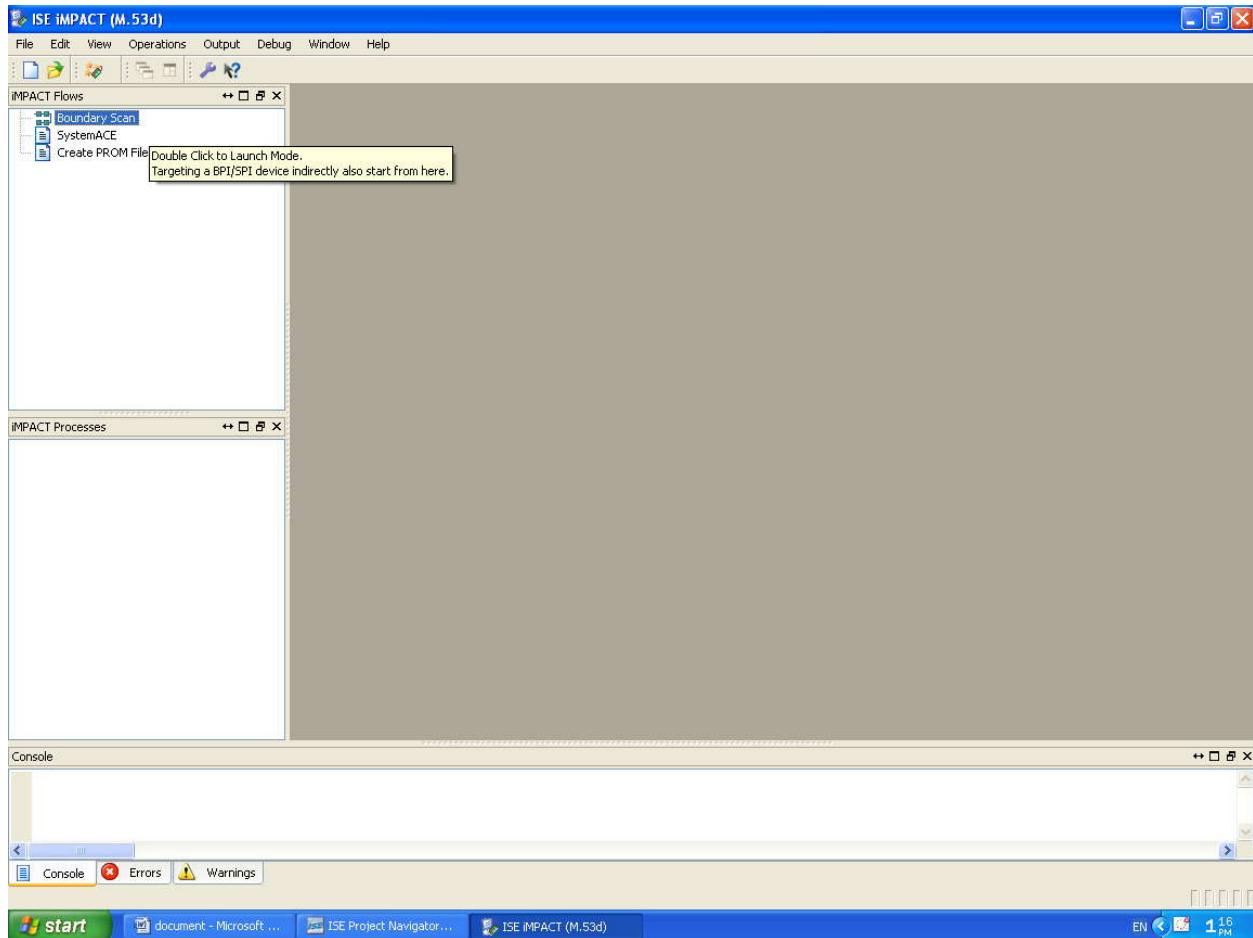


Figure-33

SPARTAN 6 TRAINER KIT

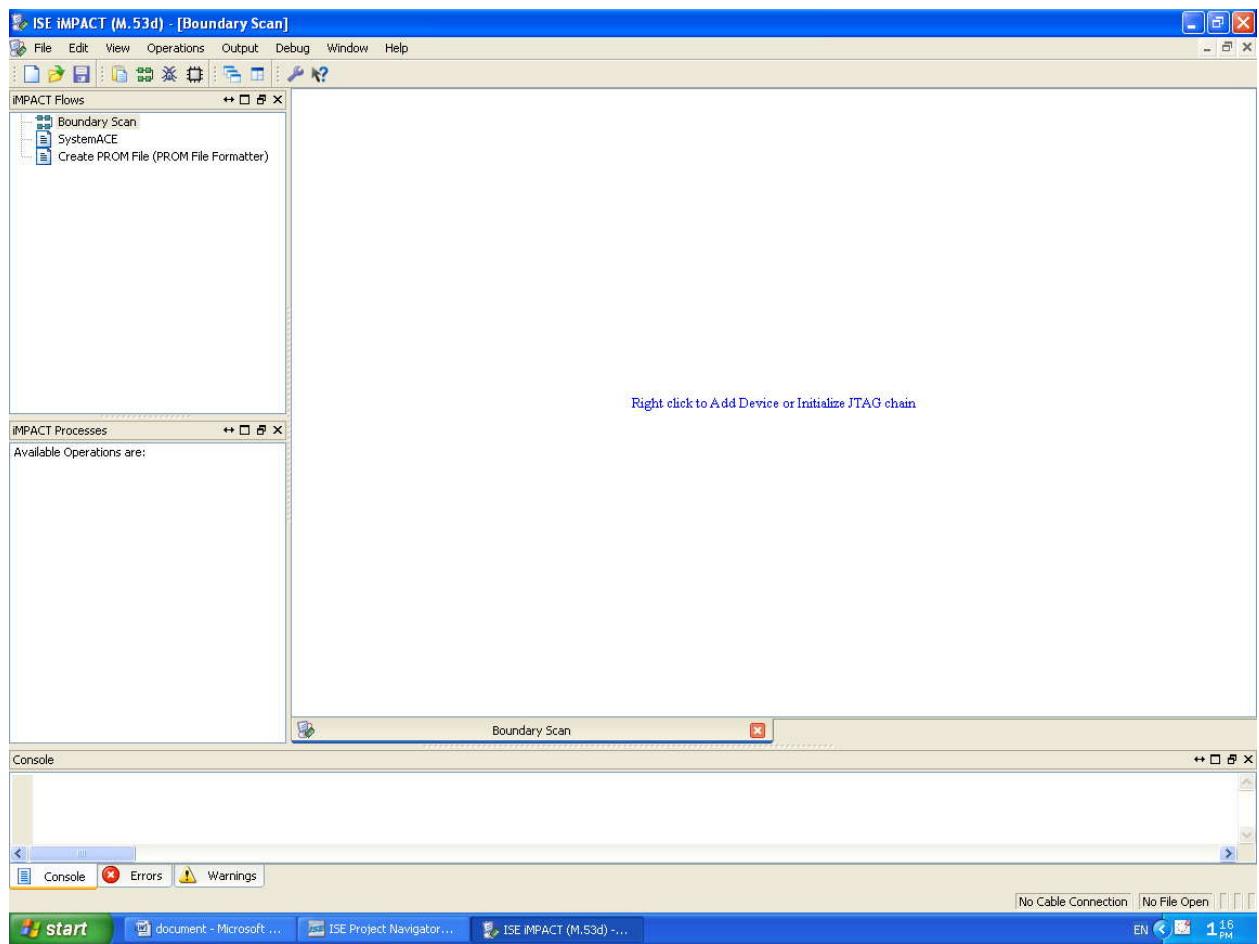


Figure-34

SPARTAN 6 TRAINER KIT

15. Right click → Initialize chain. You got a connection via the cable. Refer figure 35, 36.

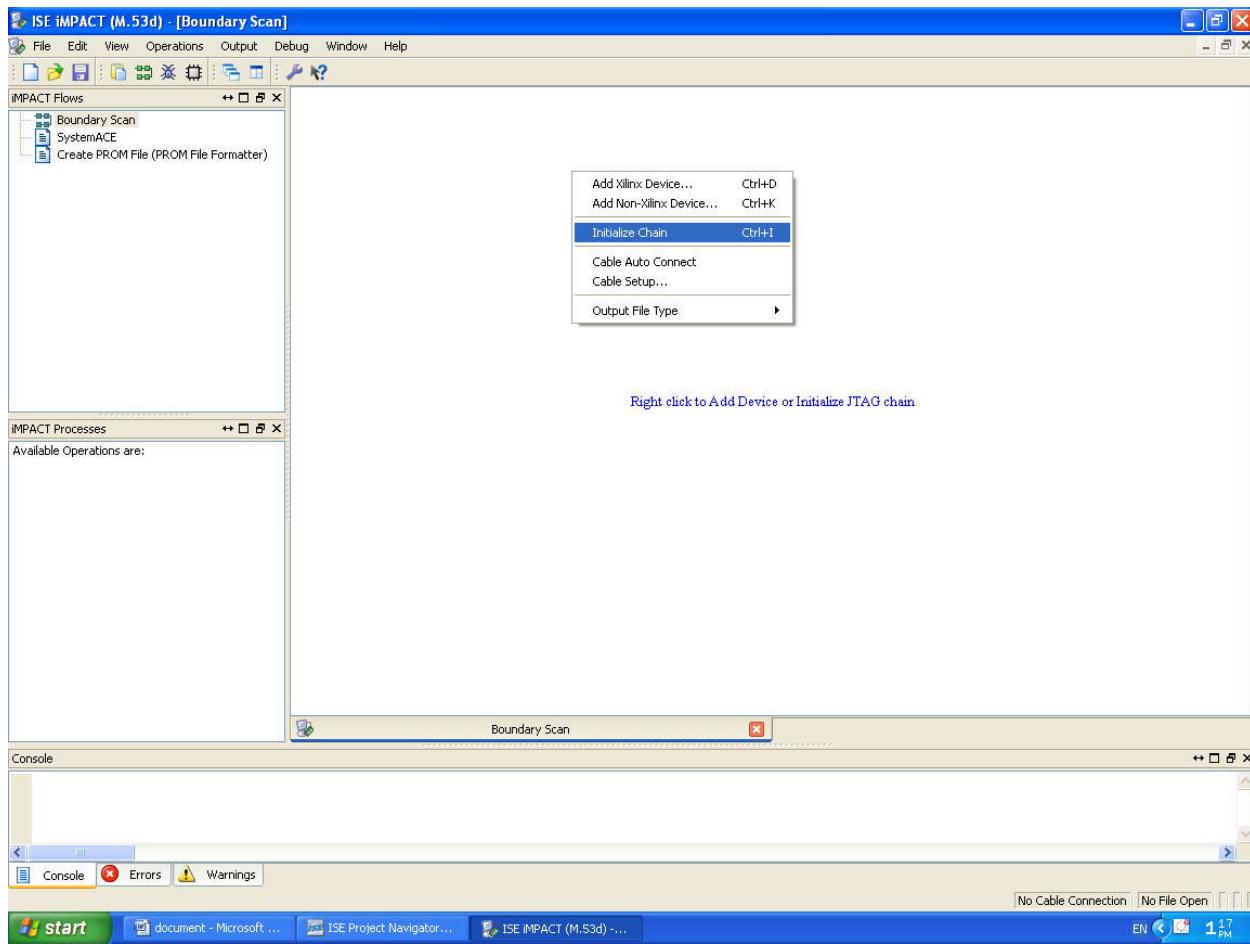


Figure-35

SPARTAN 6 TRAINER KIT

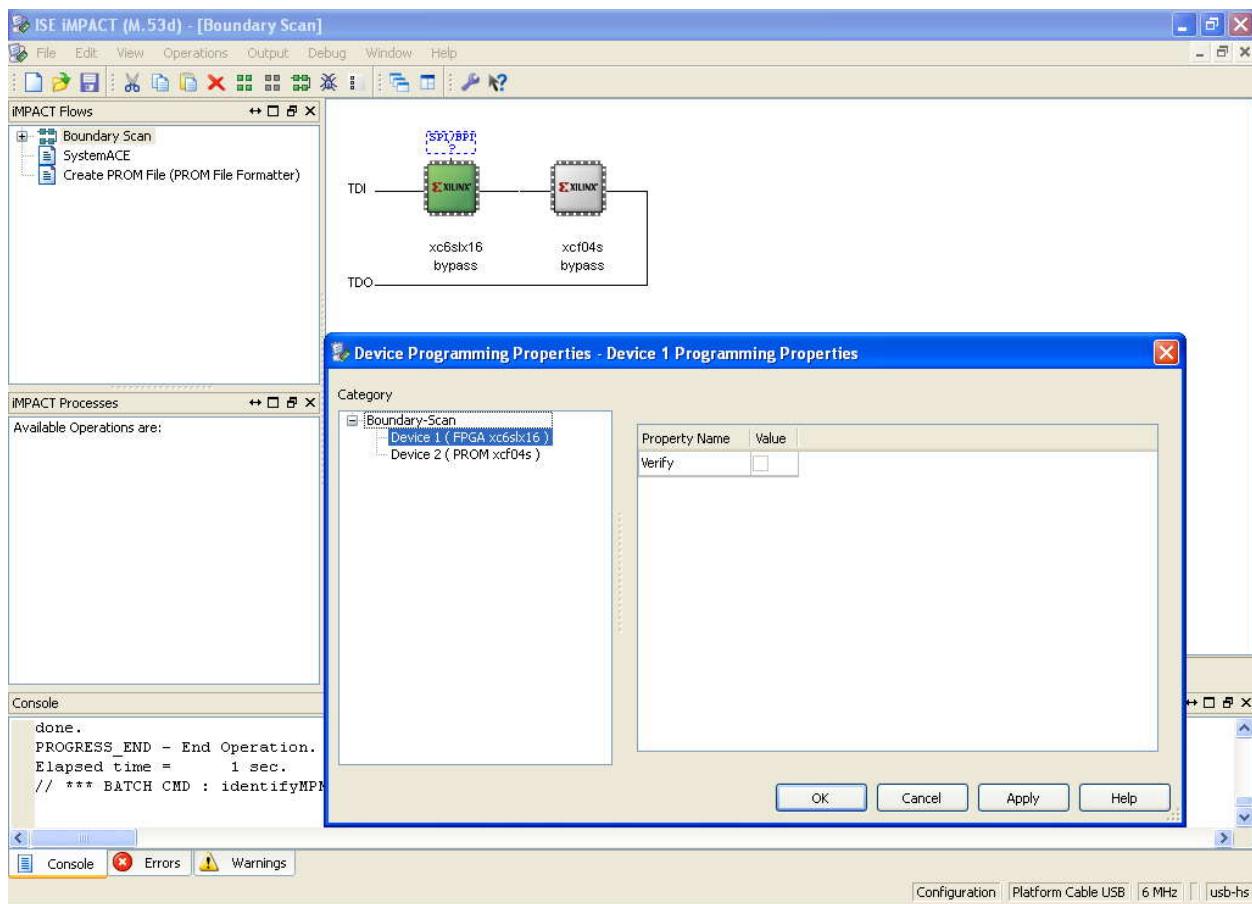


Figure – 36

SPARTAN 6 TRAINER KIT

16. Now the cable is identified **IDENTIFY SUCCEEDED**. Then select the bit file swled.bit and select open. Refer figure 37. The new window will open. Here select NO for continues the process. Refer figure-38. Then the new window will open click Cancel all refer figure 39.

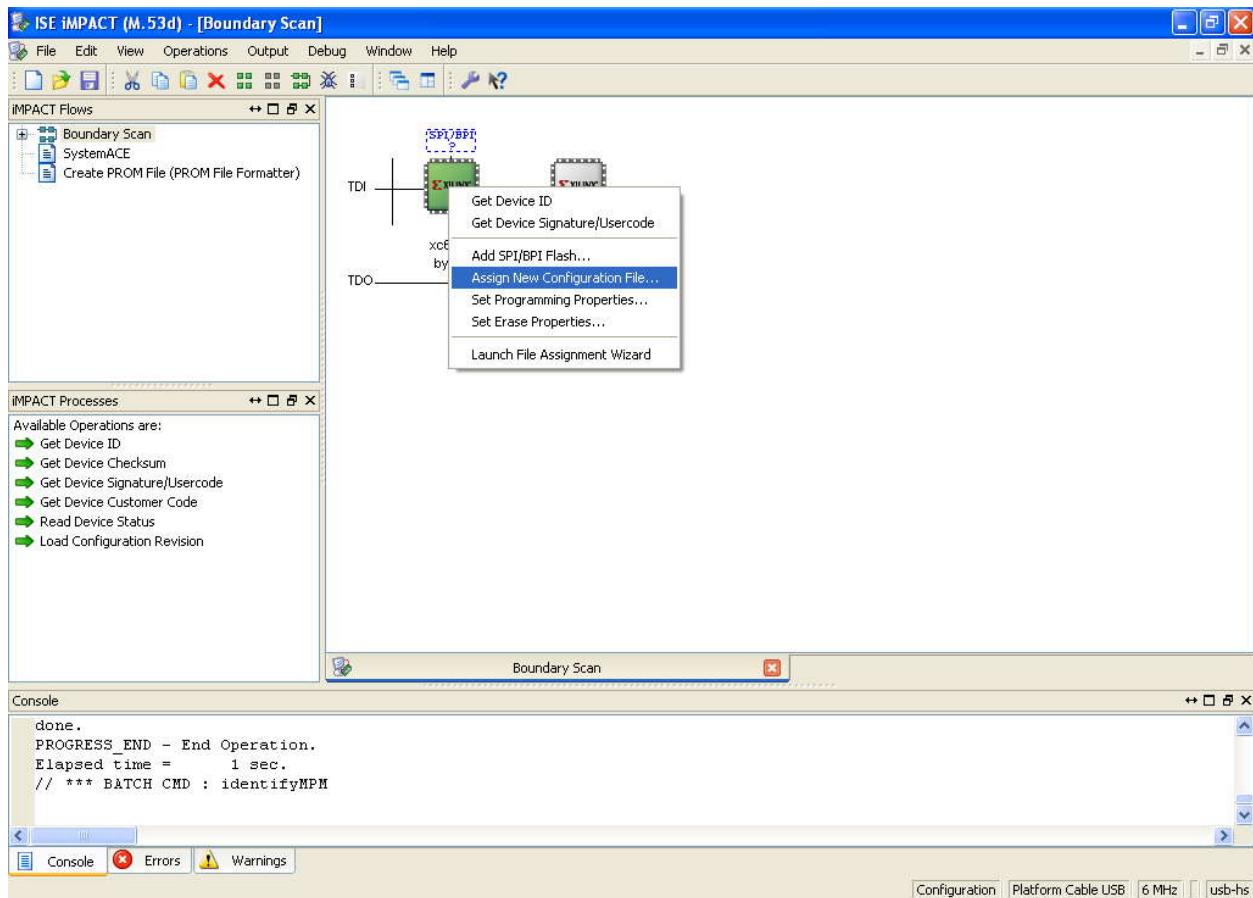


Figure-37

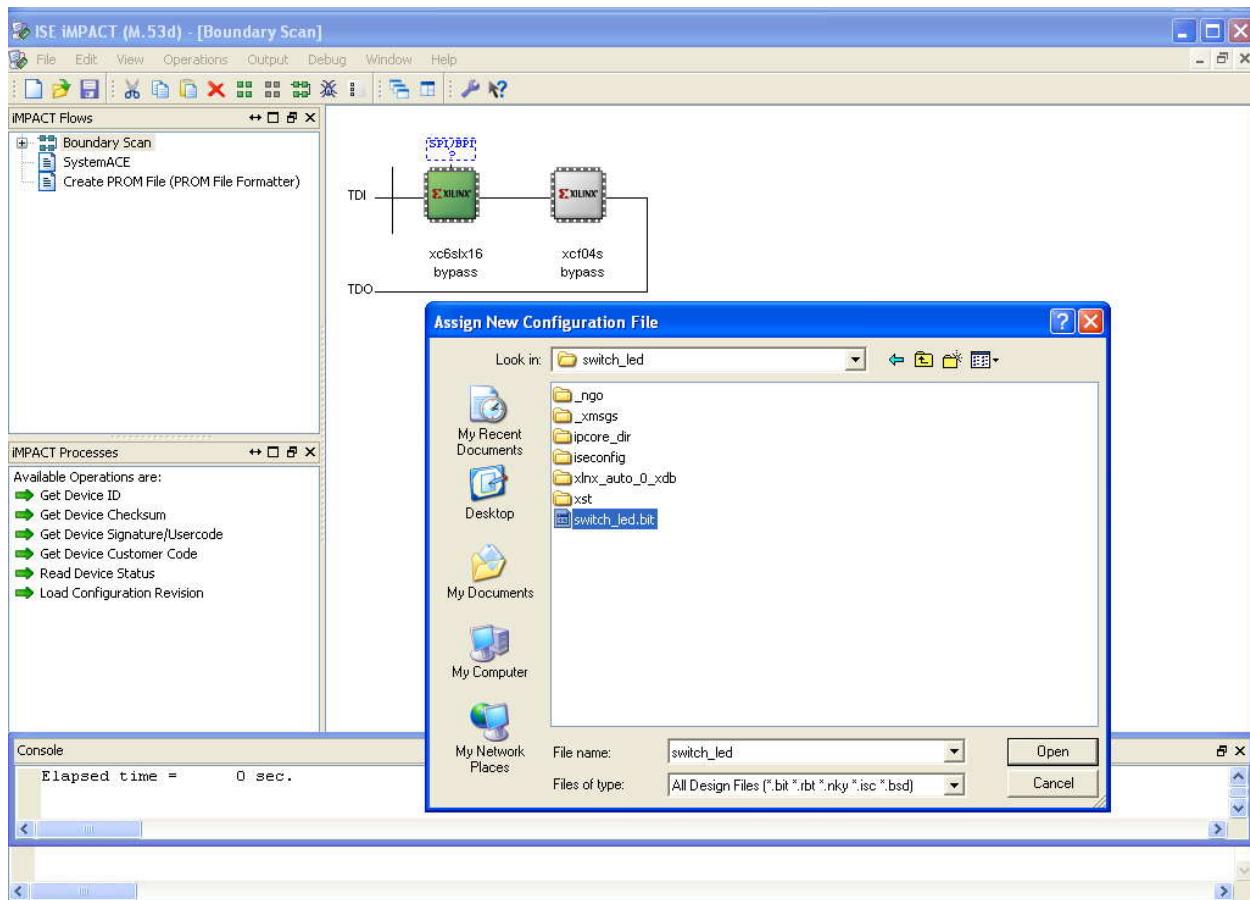


Figure-38

SPARTAN 6 TRAINER KIT

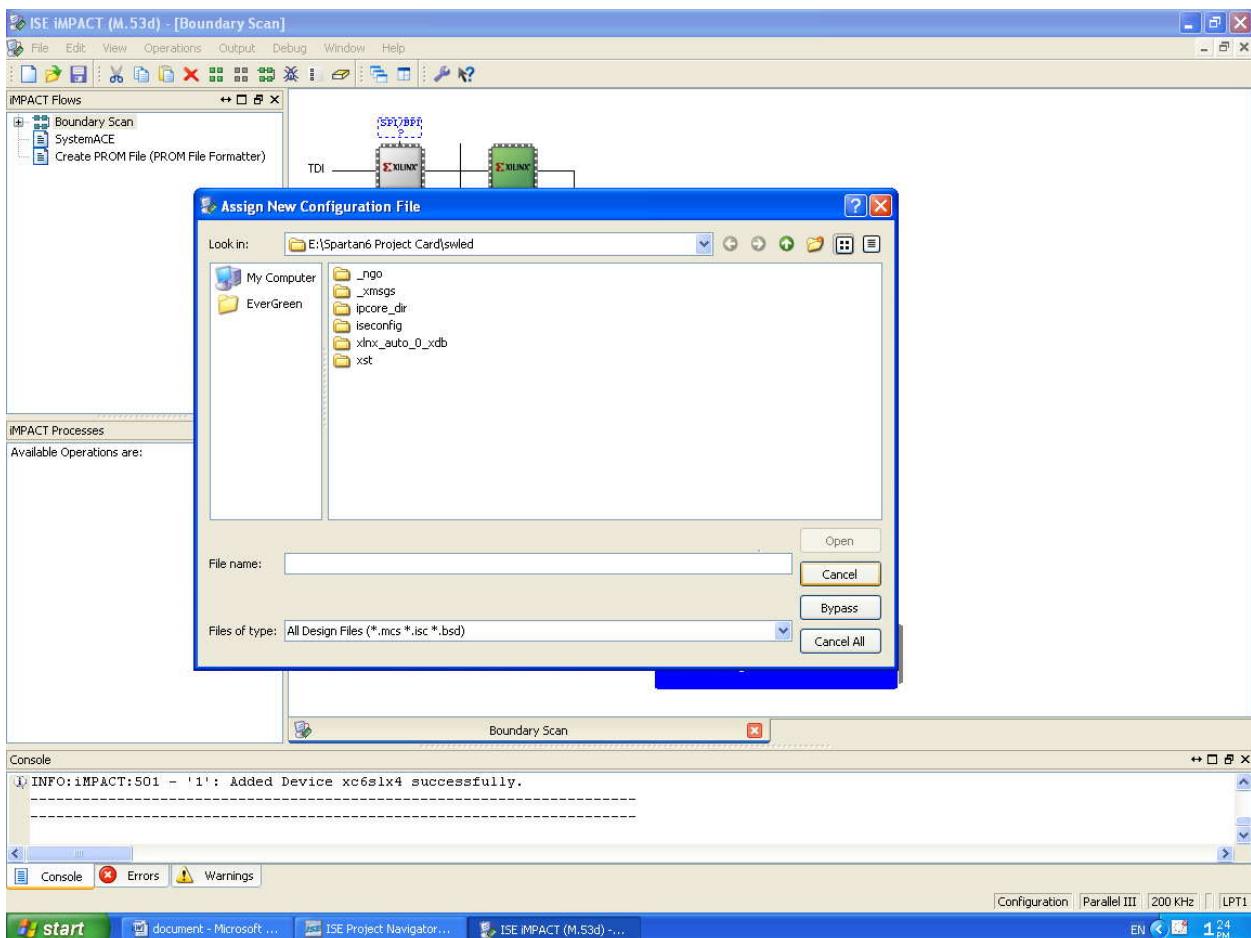


Figure-39

SPARTAN 6 TRAINER KIT

17. The new window will open it is given below. Click Apply then click Ok. Refer figure 40, 41.

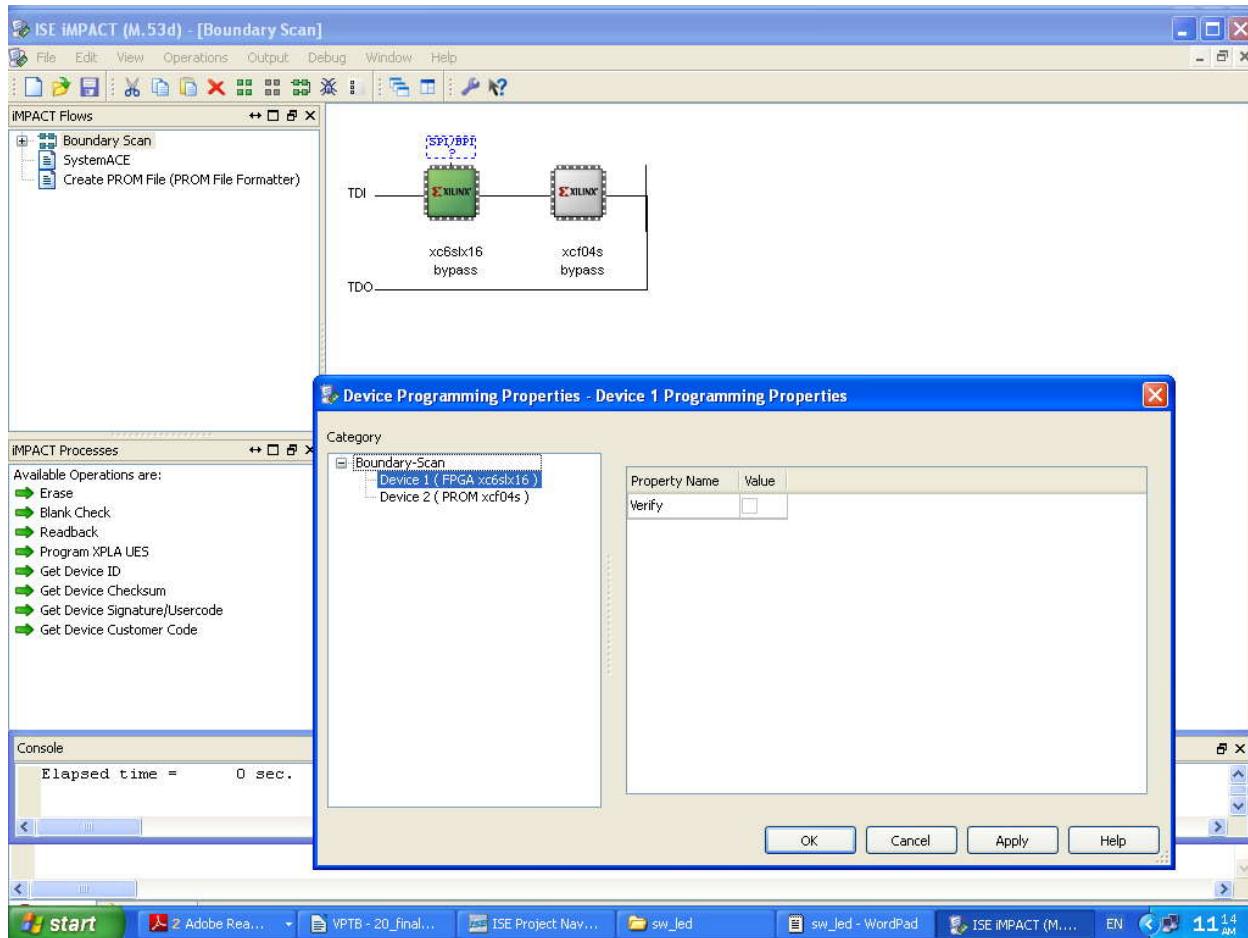


Figure-40

SPARTAN 6 TRAINER KIT

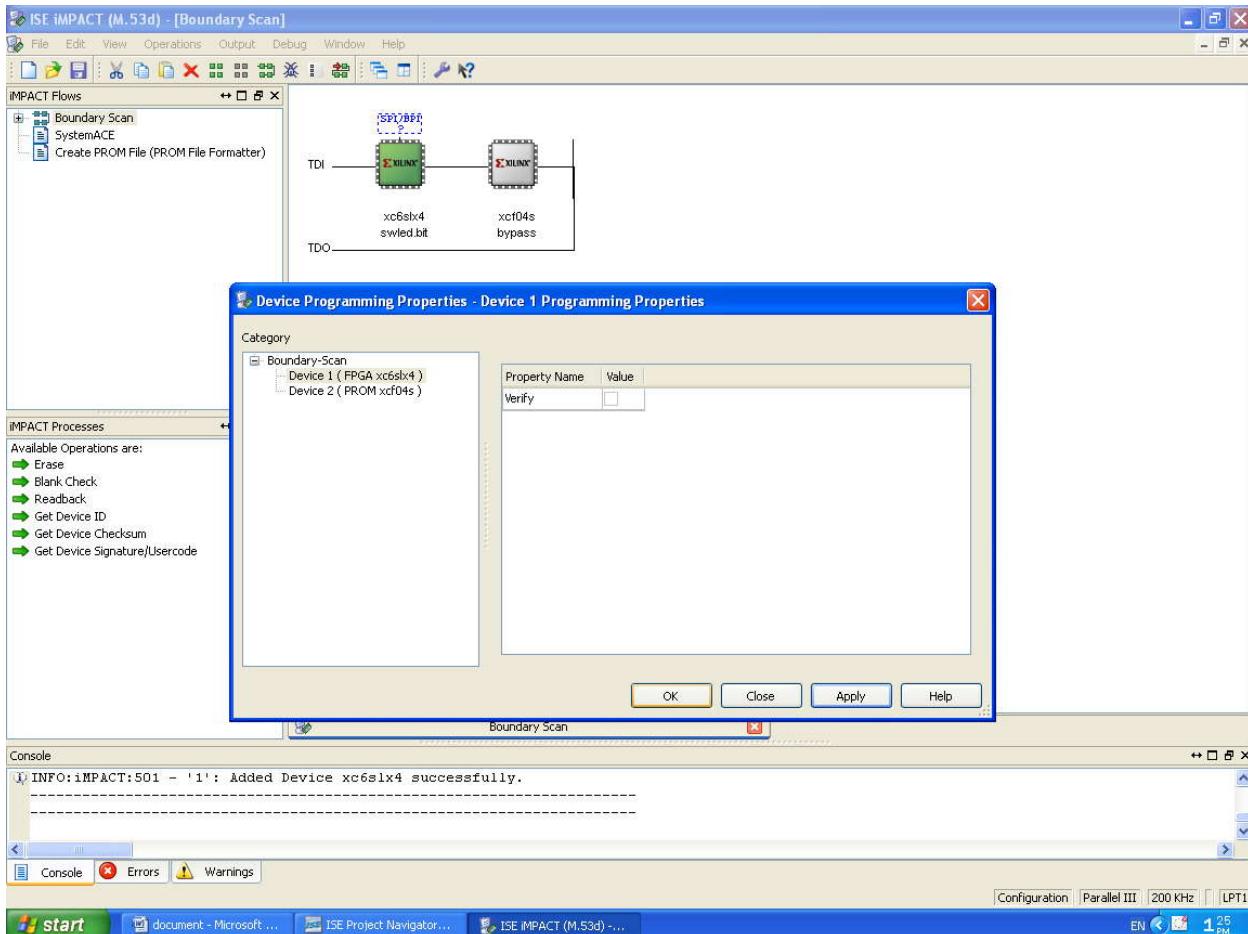


Figure-41

SPARTAN 6 TRAINER KIT

18. Then select the Xilinx device and right click then select program. Refer figure –42. Then the program is downloaded into the kit refer figure 43. Now the program was successfully downloaded into the kit. Refer figure 44.

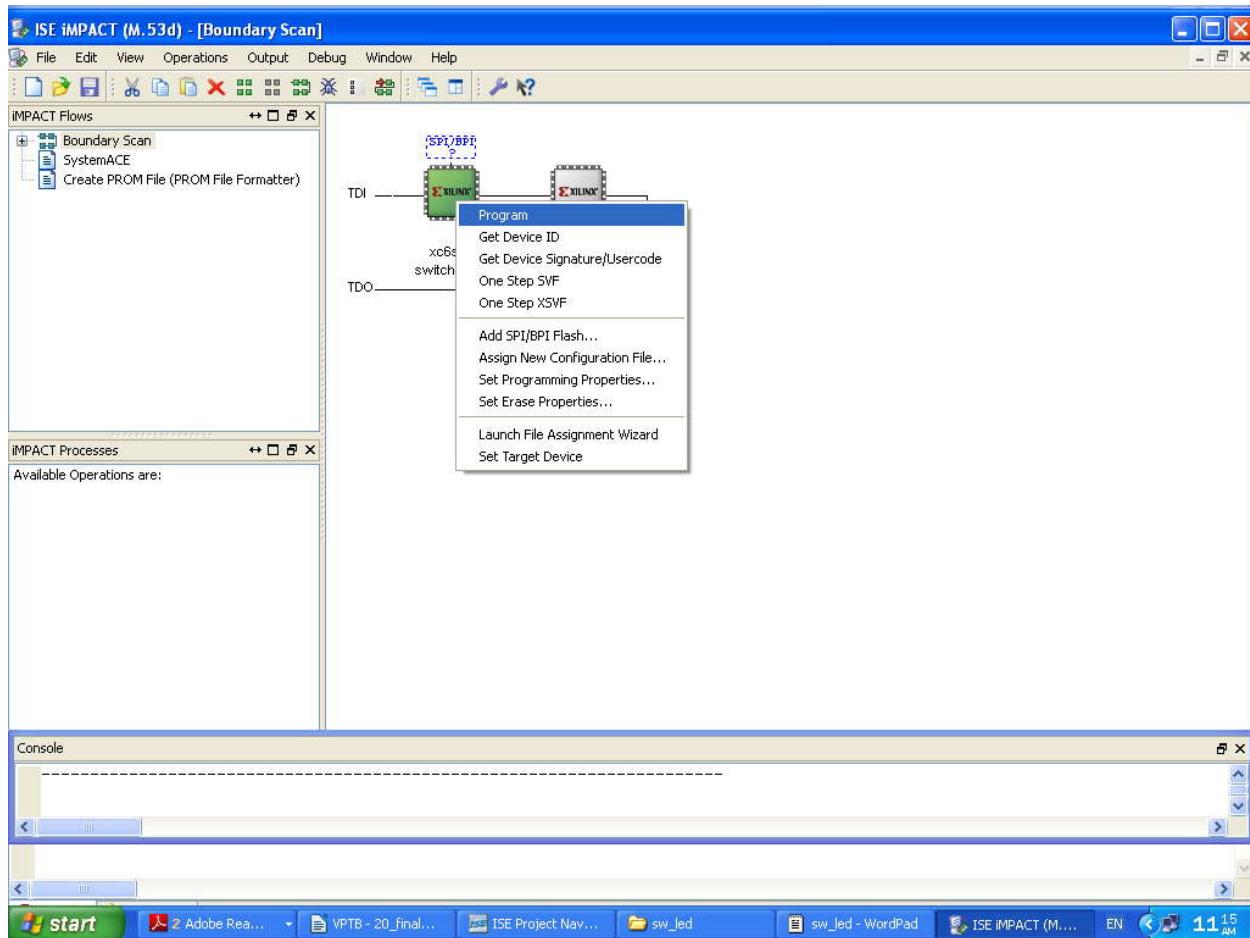


Figure-42

SPARTAN 6 TRAINER KIT

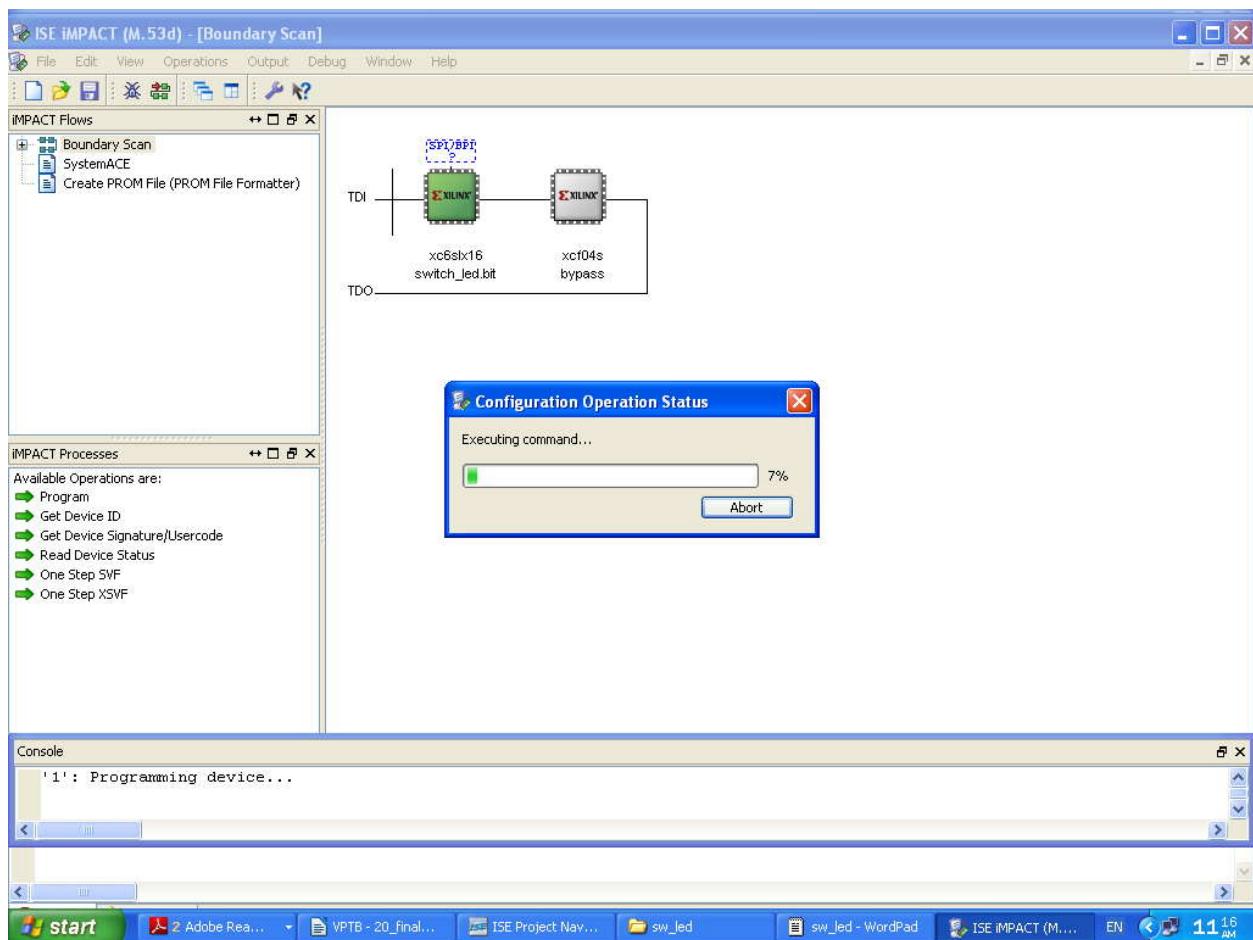


Figure-43

SPARTAN 6 TRAINER KIT

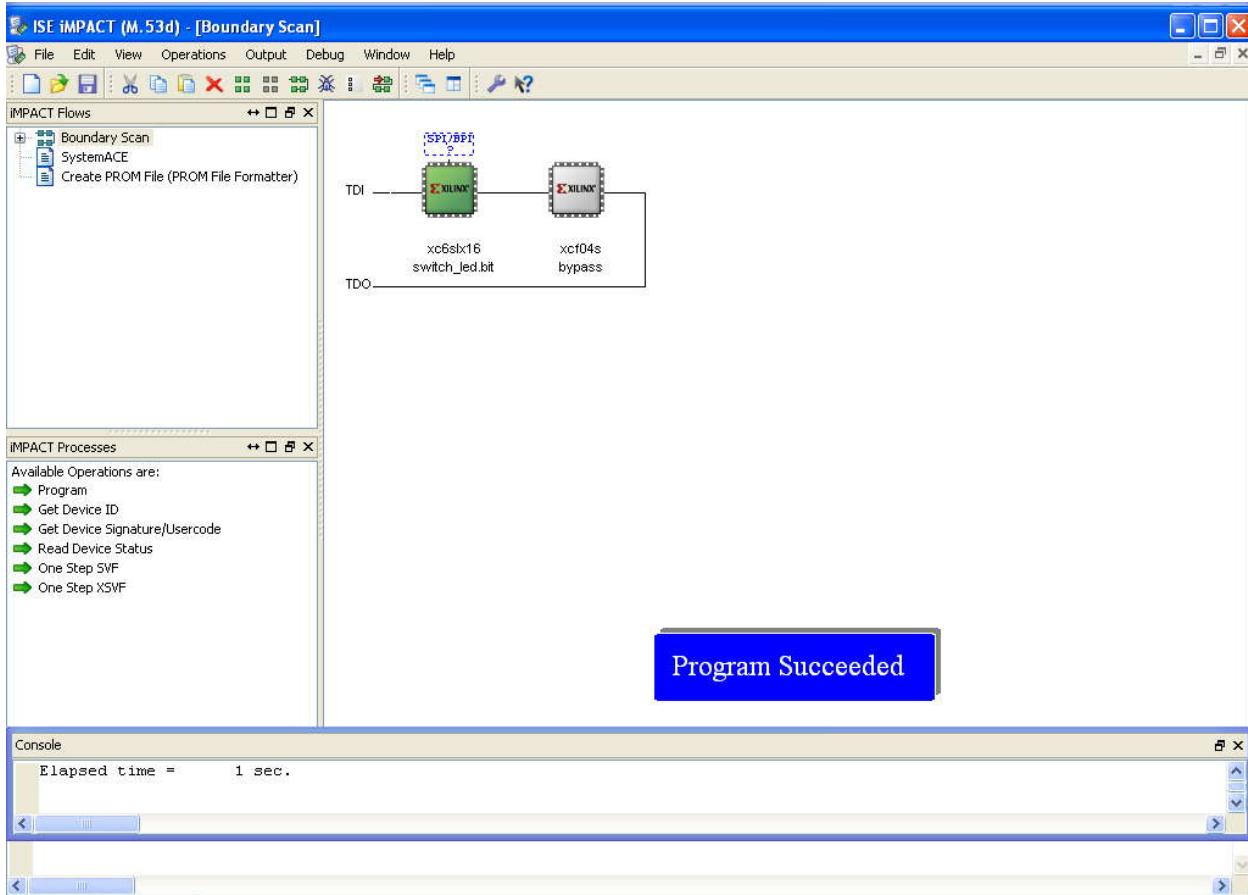


Figure-44

SPARTAN 6 TRAINER KIT

PROM FILE GENERATION

To generate a PROM file double click Create PROM File in the impact window. Refer figure 45. Then the new window will open. Refer figure 46. PROM reset (SW2) switch is used to reset the PROM IC.

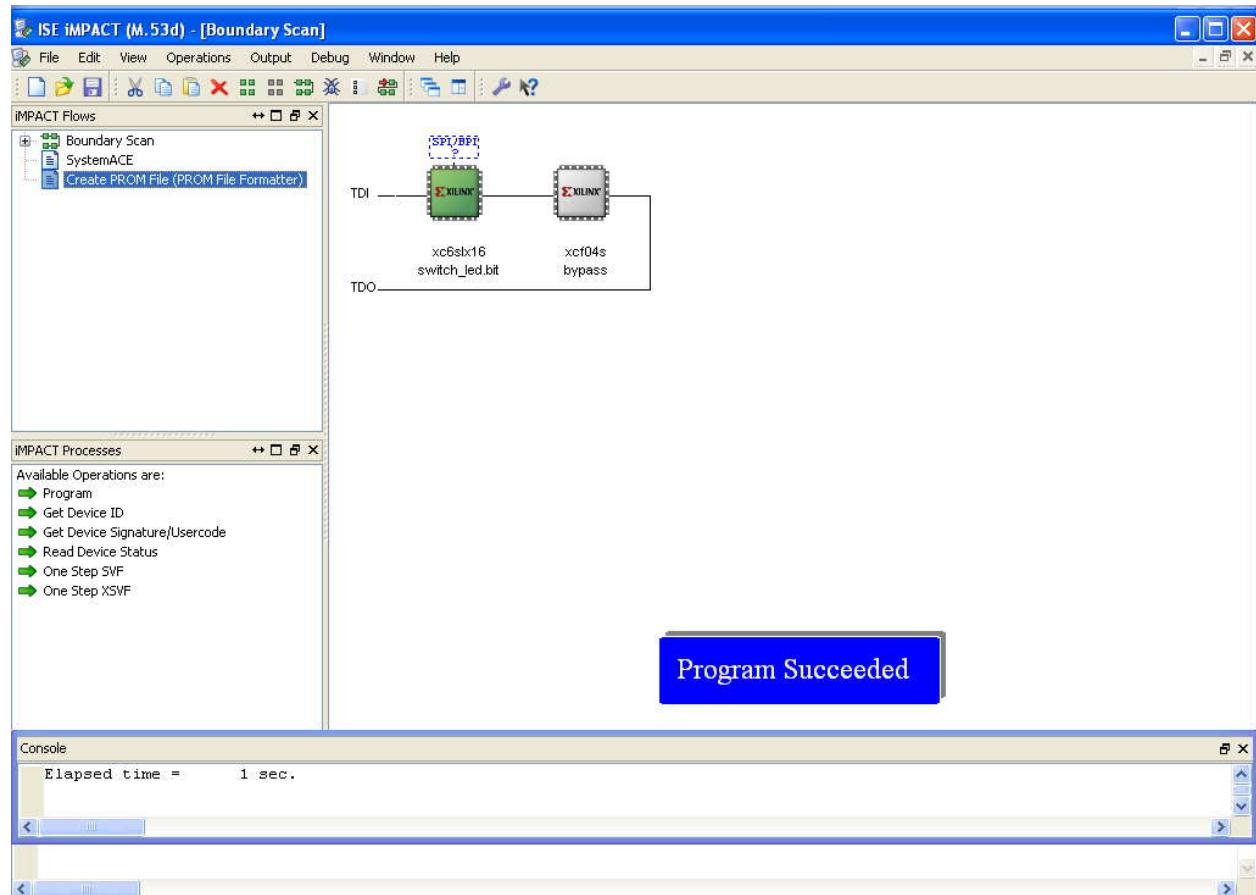


Figure - 45

SPARTAN 6 TRAINER KIT

Select first arrow marks in the window and assign xcf04s [4M]. Then click Add Storage device. Refer Figure 47.

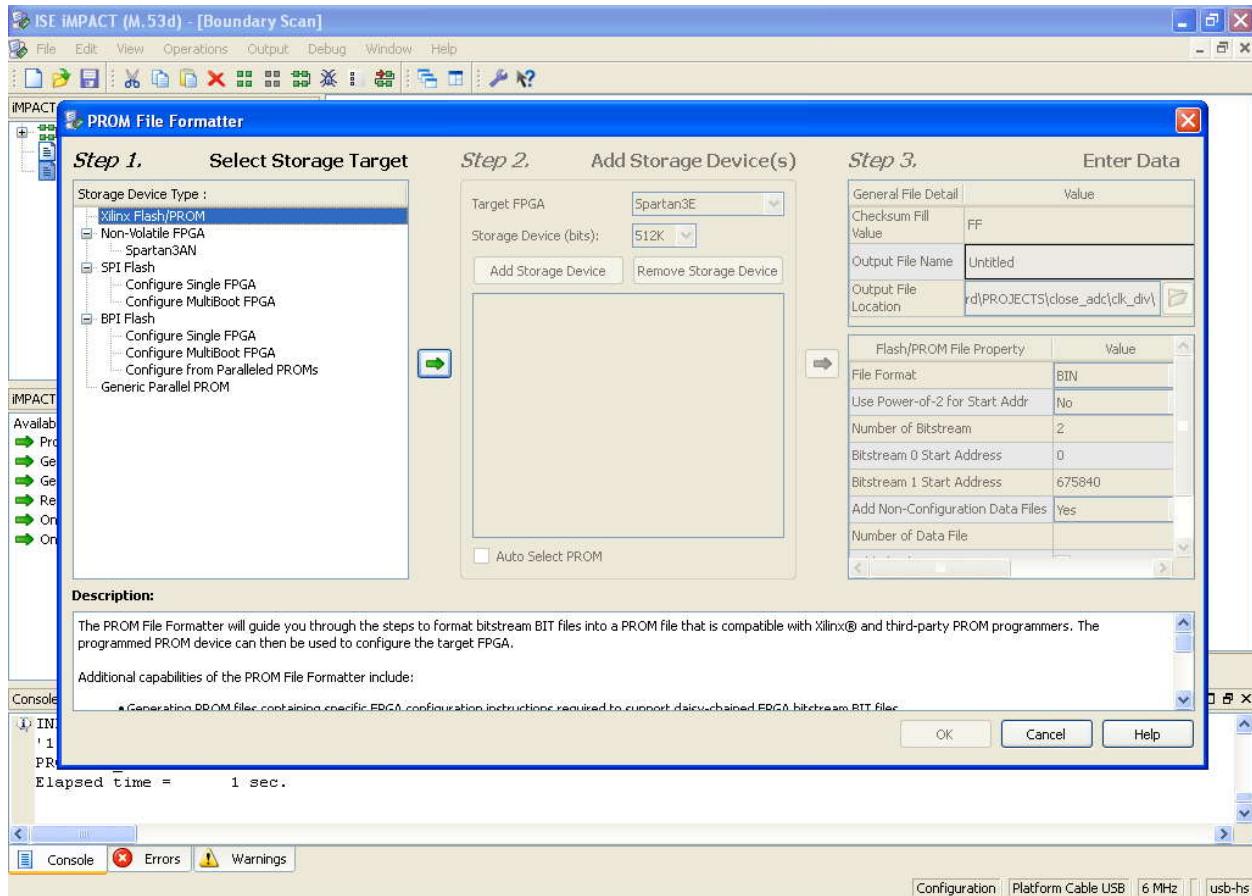


Figure - 46

SPARTAN 6 TRAINER KIT

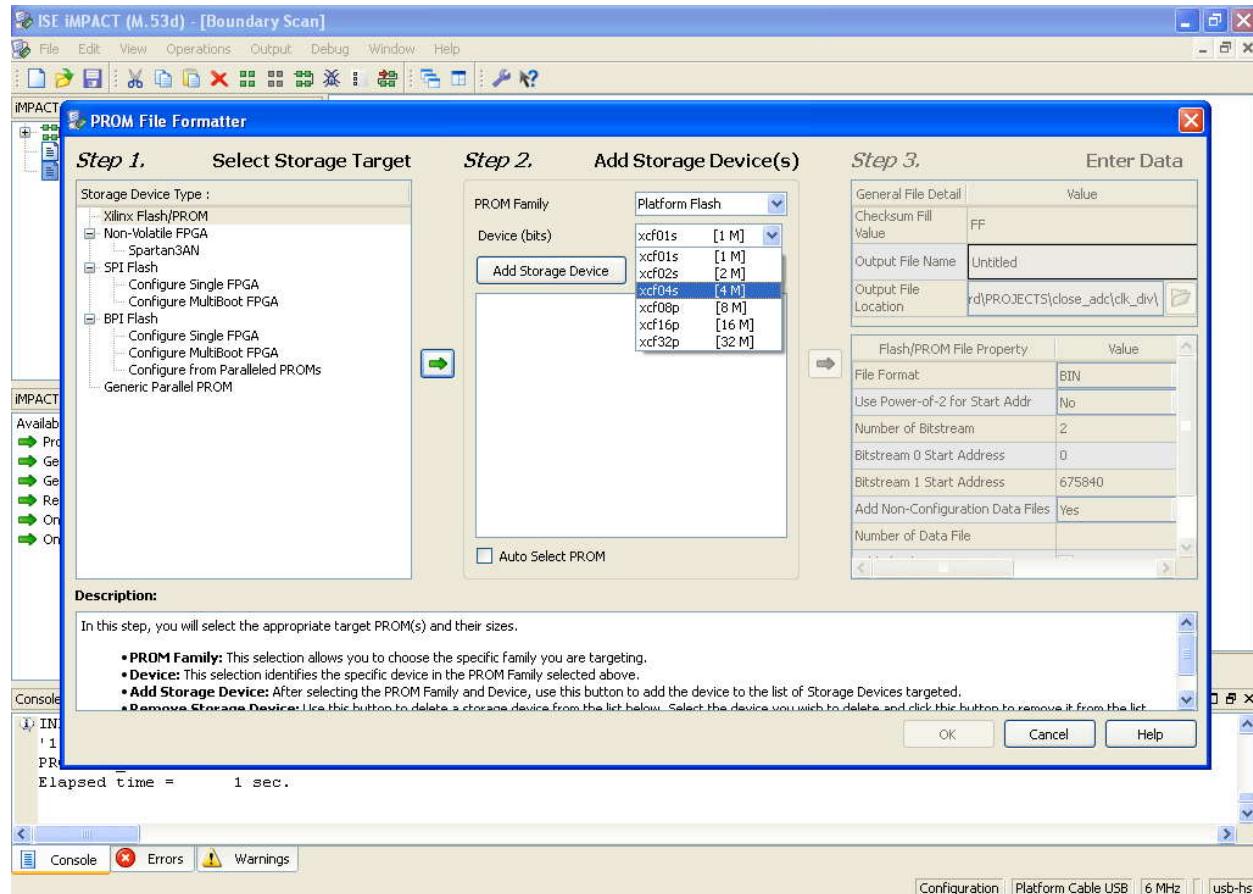


Figure - 47

SPARTAN 6 TRAINER KIT

Click second arrow mark and give output file name and output file location. Then select MCS in the file format. After doing this above process click OK. Refer figure 48.

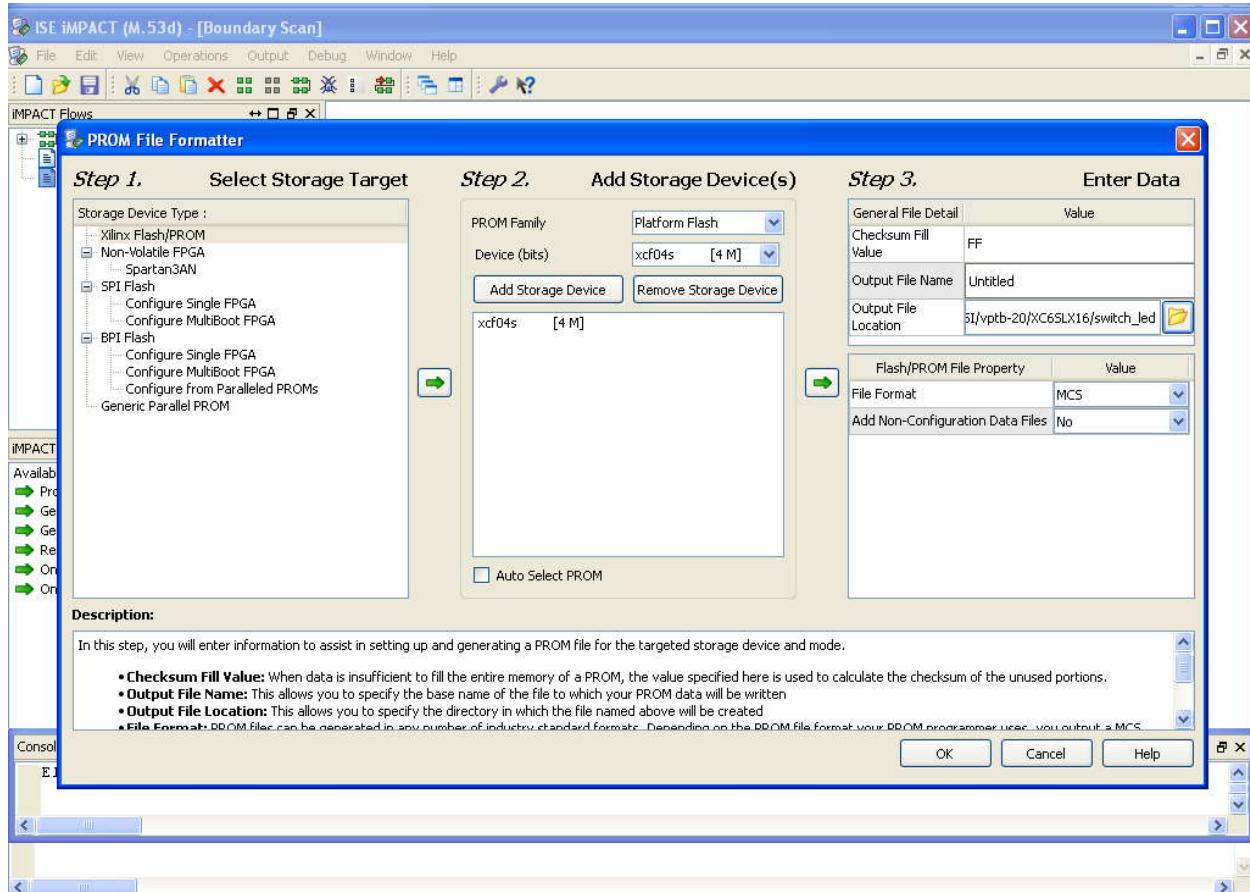


Figure - 48

SPARTAN 6 TRAINER KIT

Click OK to continue the process. One small window will appear, this is about to add a Xilinx device. Here click OK. Then the new window will open, here select the swled.bit Bit file and click Open. Refer figure 49, 50.

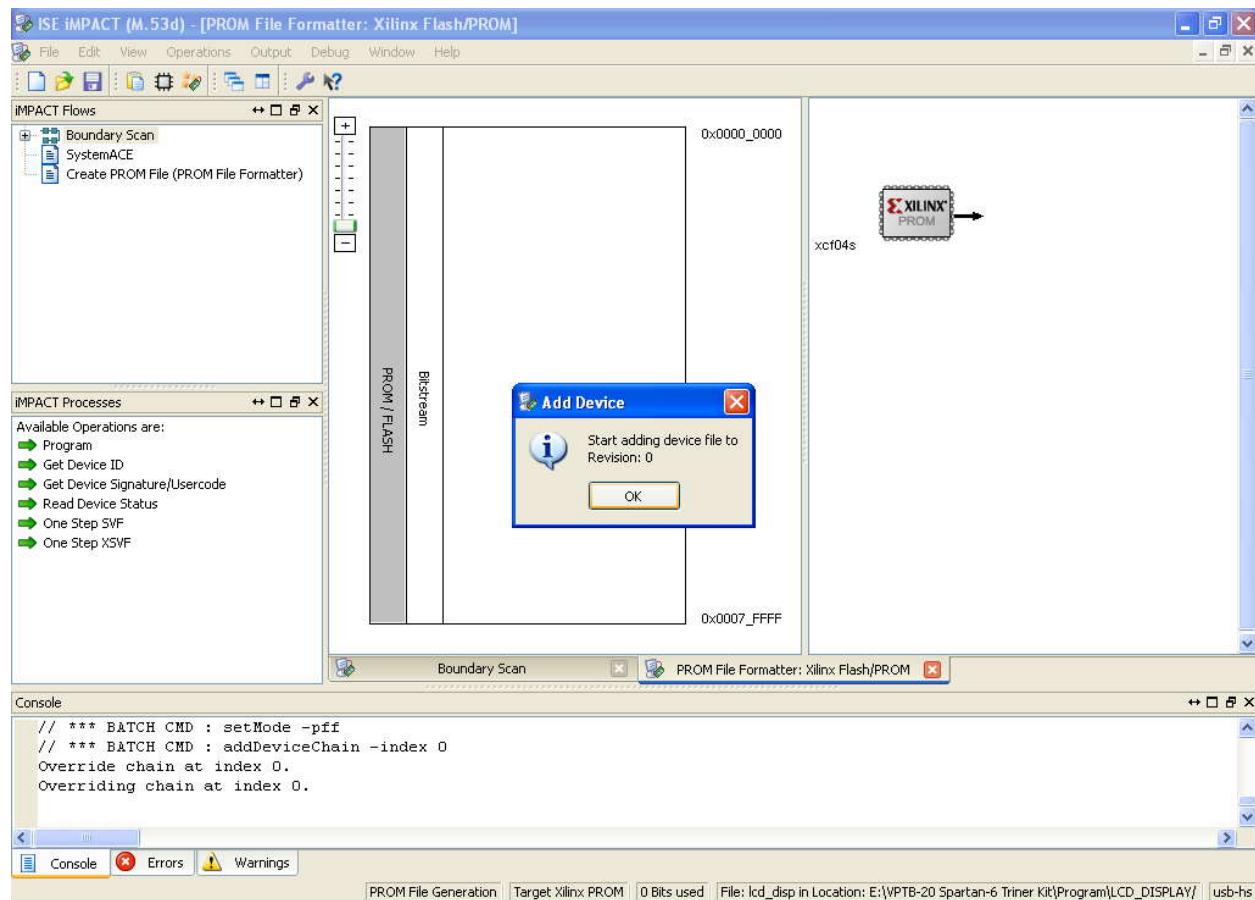


Figure - 49

SPARTAN 6 TRAINER KIT

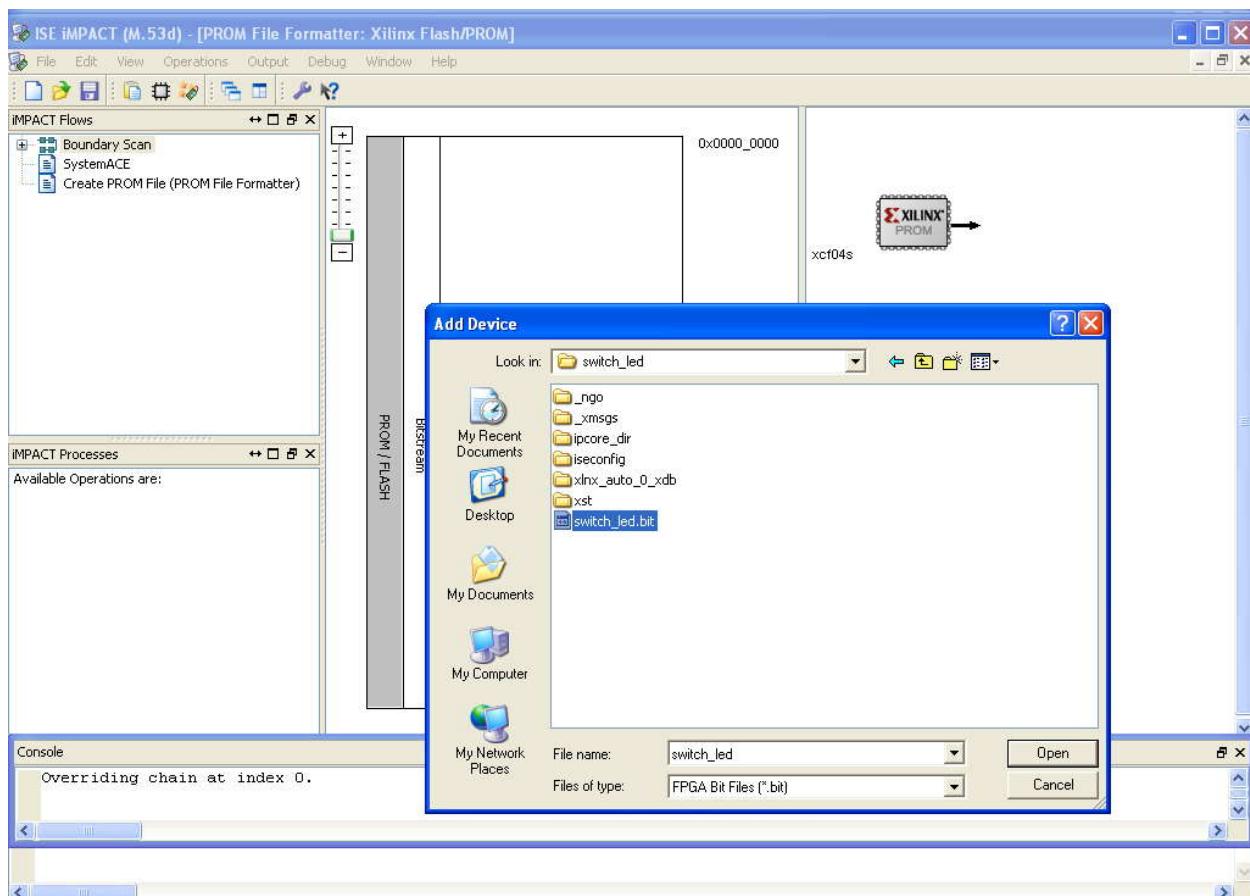


Figure - 50

SPARTAN 6 TRAINER KIT

After selecting the bit file a new small window will open. This is about to add another device. Click NO to continue the process. Then click OK. Refer figure 51, 52.

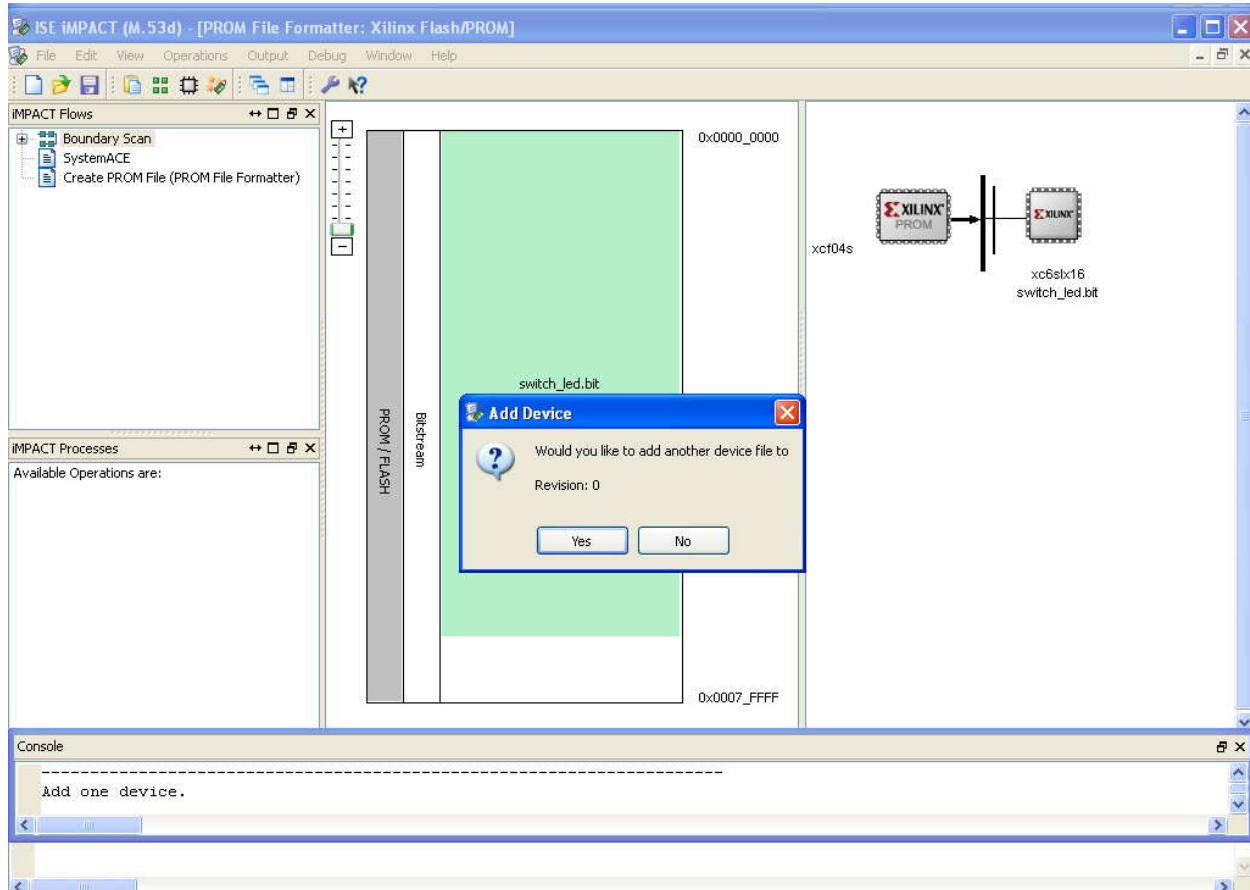


Figure - 51

SPARTAN 6 TRAINER KIT

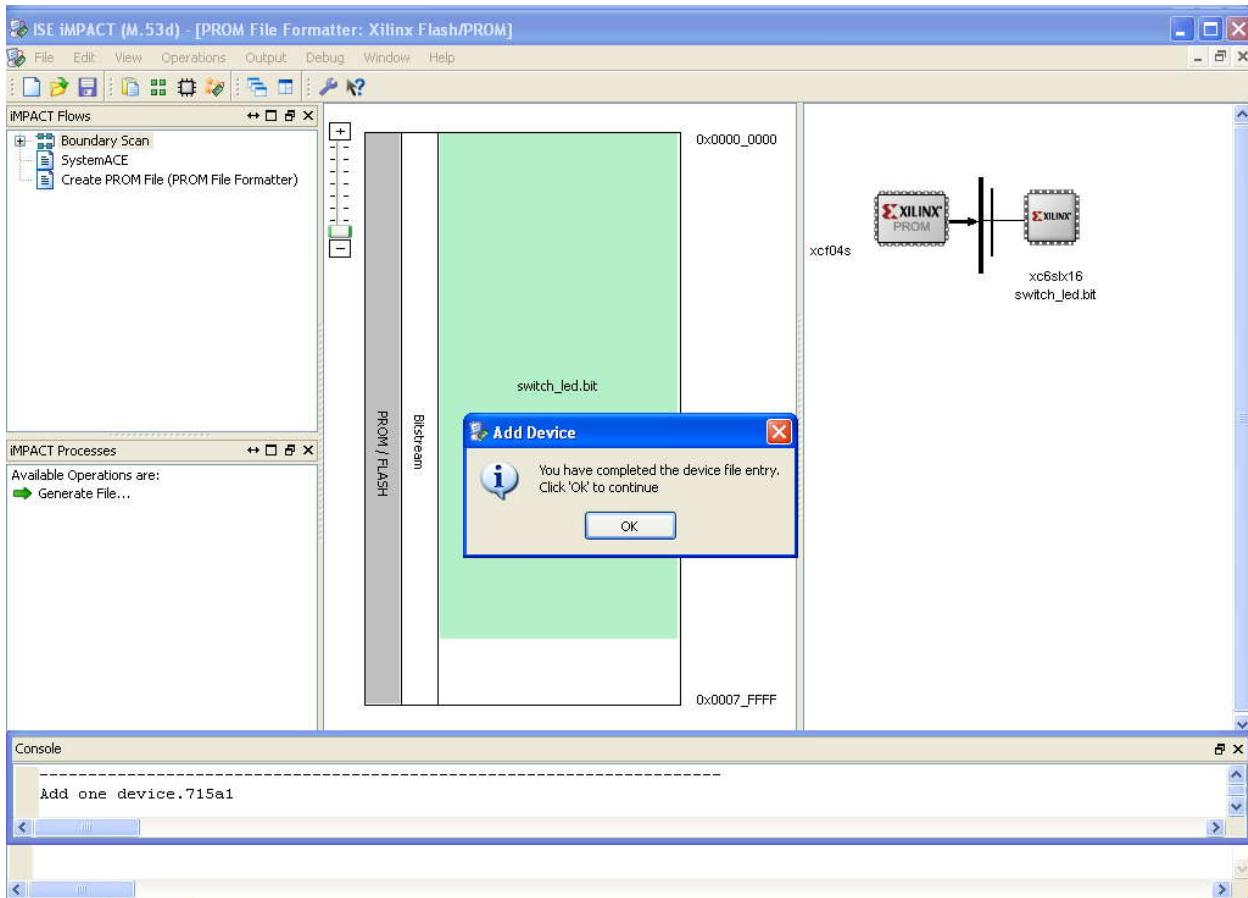


Figure - 52

SPARTAN 6 TRAINER KIT

Select impact process window. And Double click Generate file then the PROM file was generated is notified as GENERATION SUCCEEDED. Refer figure 53, 54.

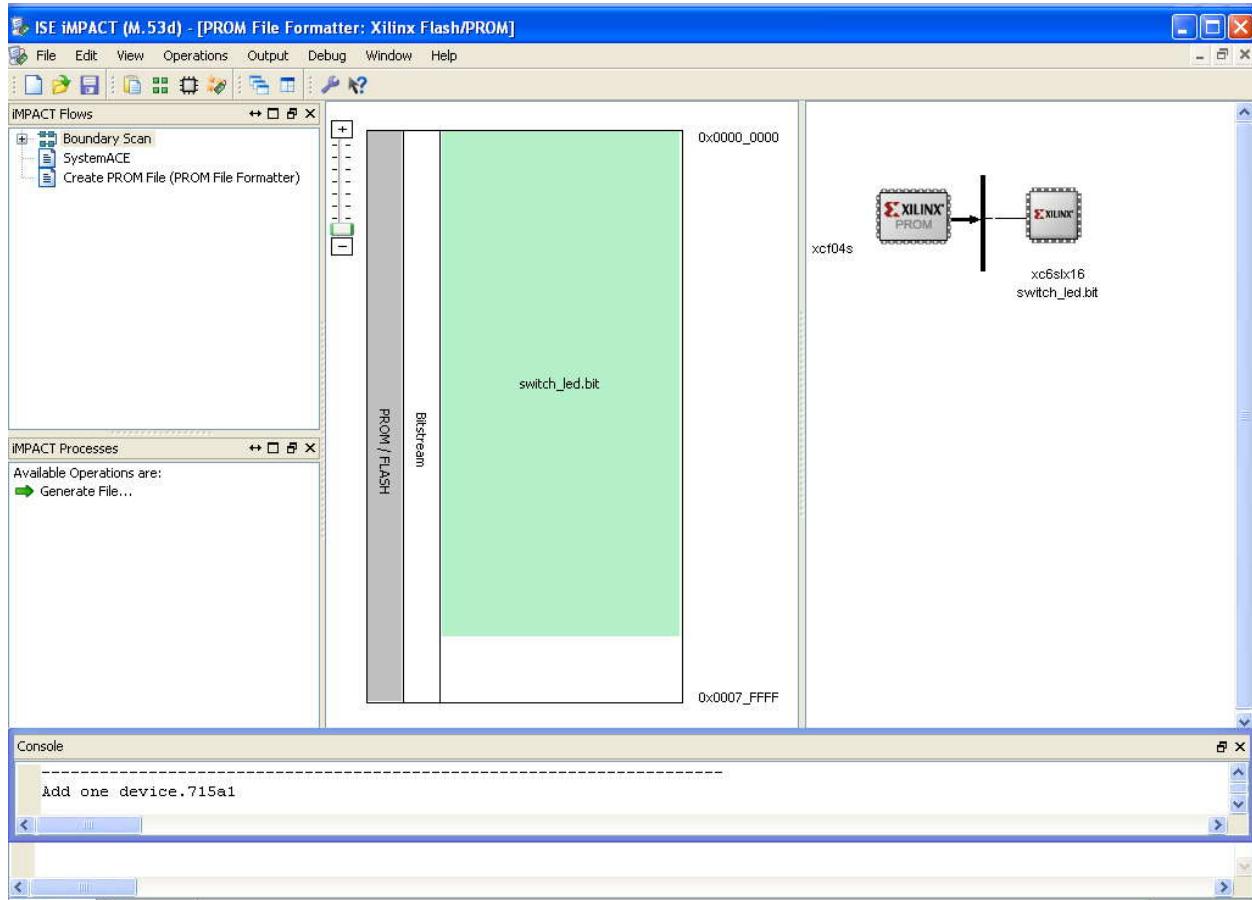


Figure - 53

SPARTAN 6 TRAINER KIT

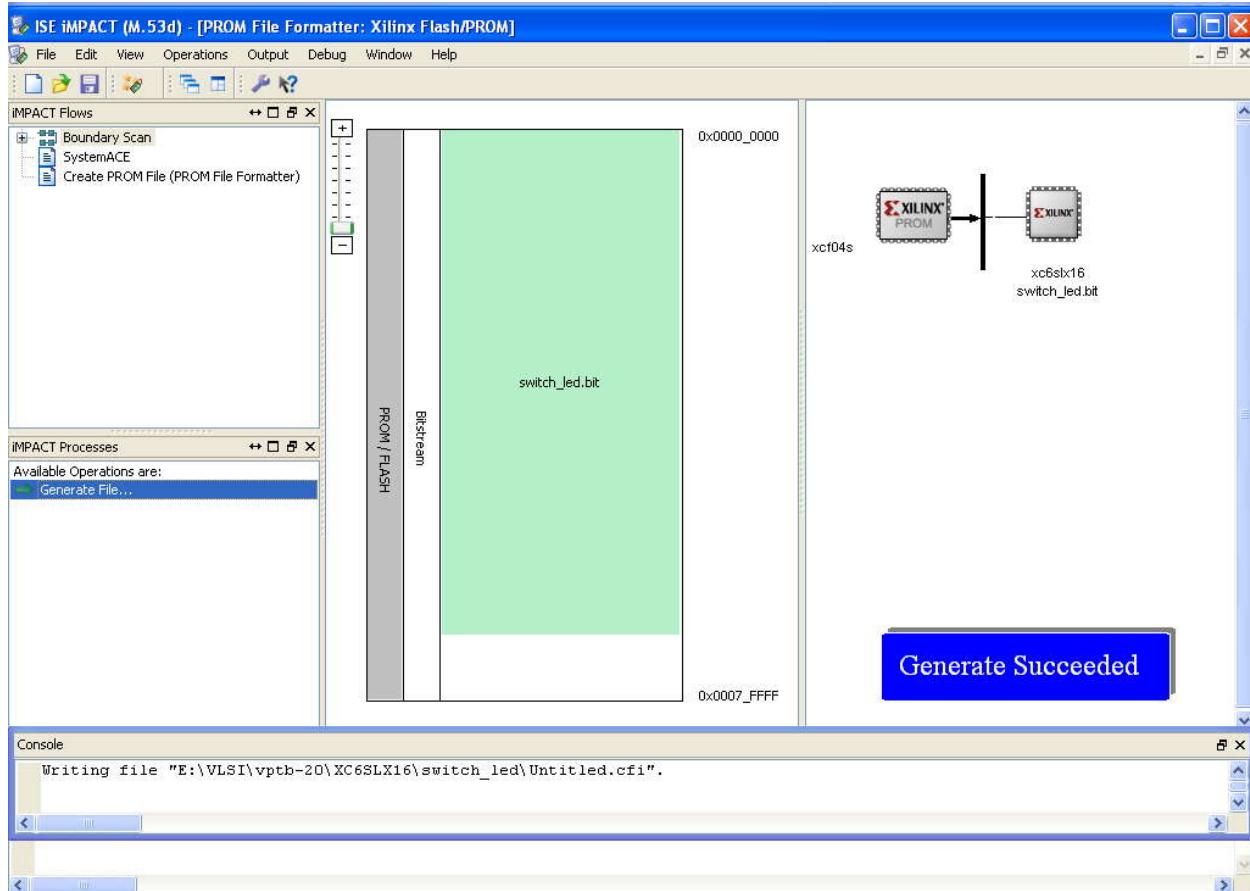


Figure - 54

SPARTAN 6 TRAINER KIT

Select middle device and right click → Assign new configuration file. New window will appear. Refer figure 55.

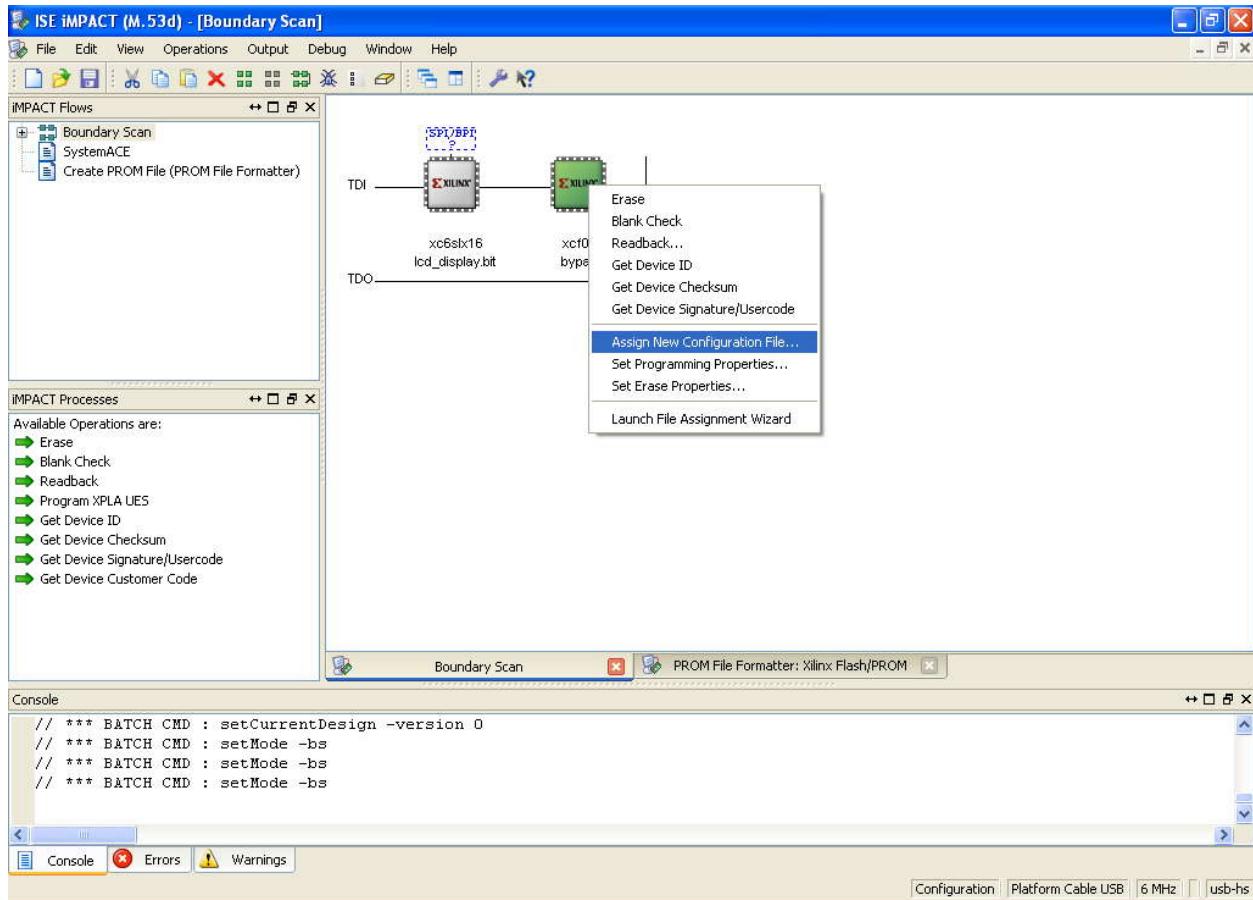


Figure - 55

SPARTAN 6 TRAINER KIT

Select the MCS file **swled.mcs**, where you stored and click open. Refer figure 56.

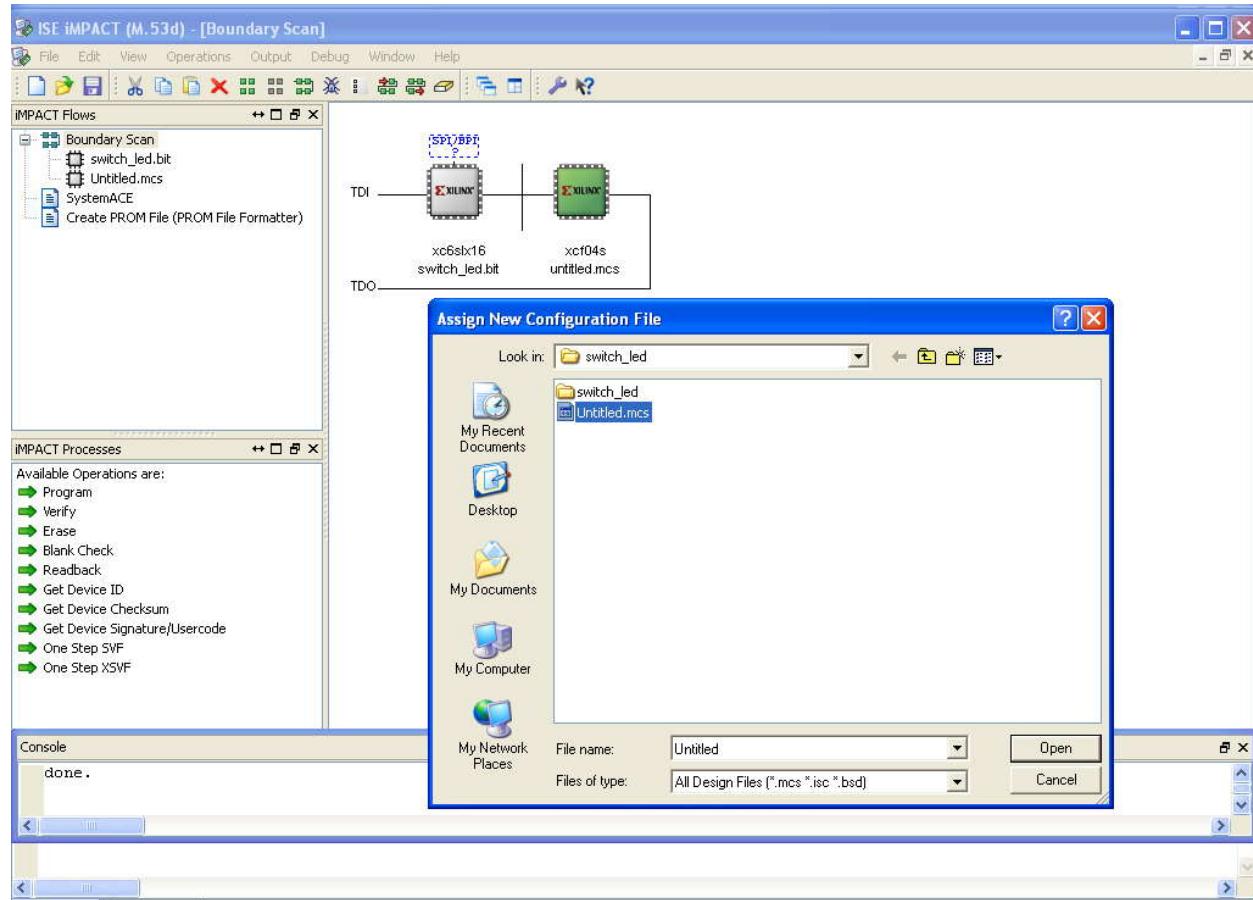


Figure - 56

SPARTAN 6 TRAINER KIT

Select the middle device, right click → Program. Then the new window will appear, select PROM Device (PROM xcf04s) → Apply → Ok. Refer figure 57, 58.

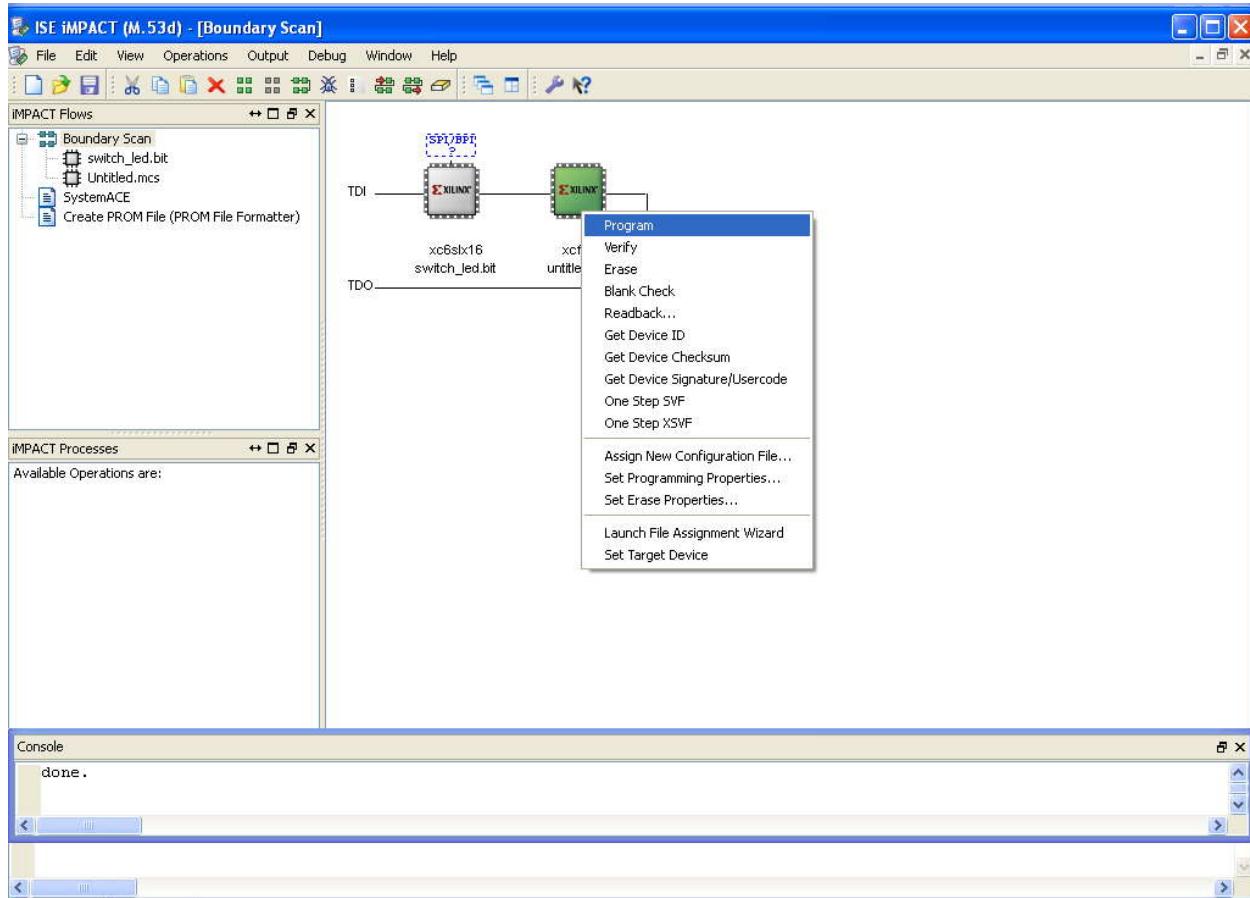


Figure - 57

SPARTAN 6 TRAINER KIT

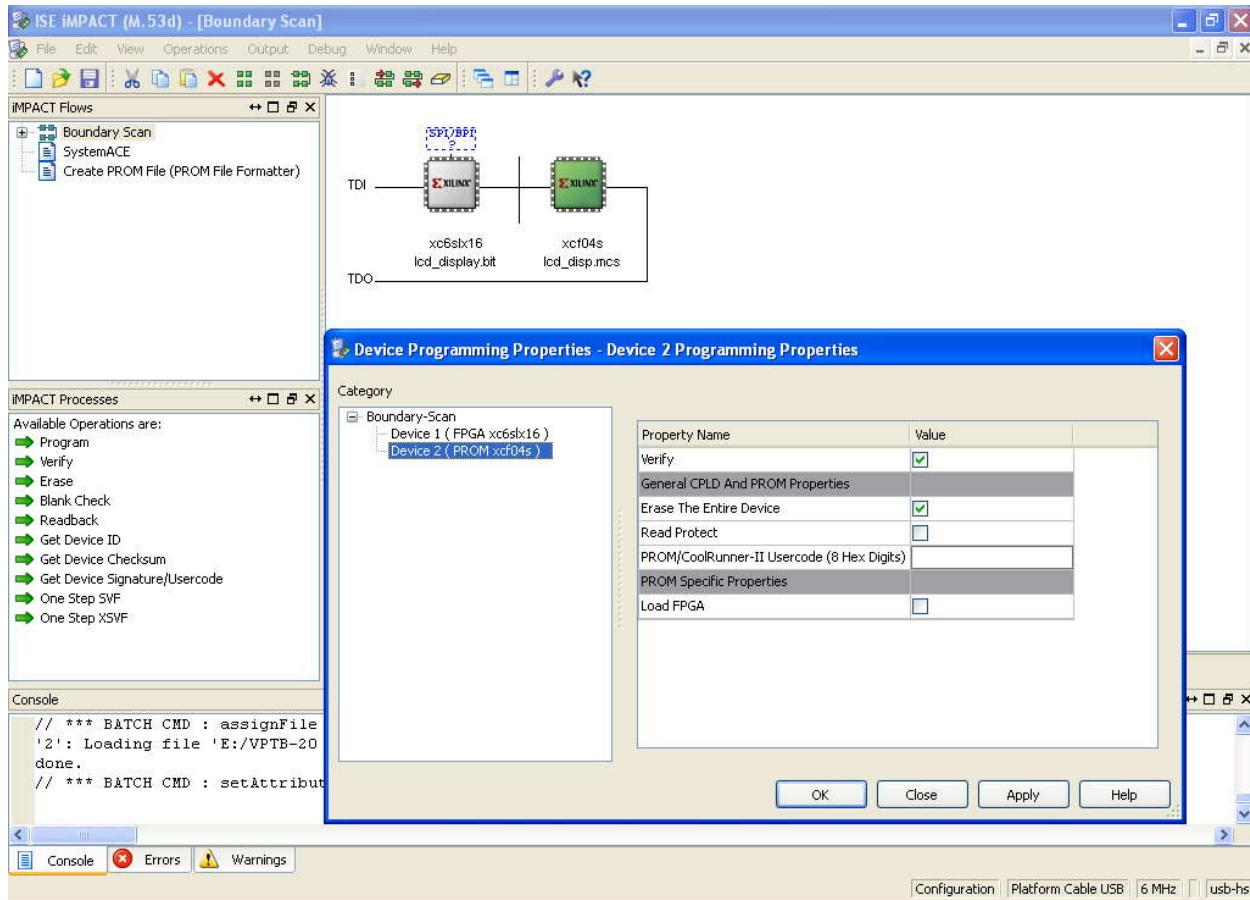


Figure - 58

SPARTAN 6 TRAINER KIT

Then the command was executed. You got a notification as **Program Succeeded**. Refer figure 59, 60.

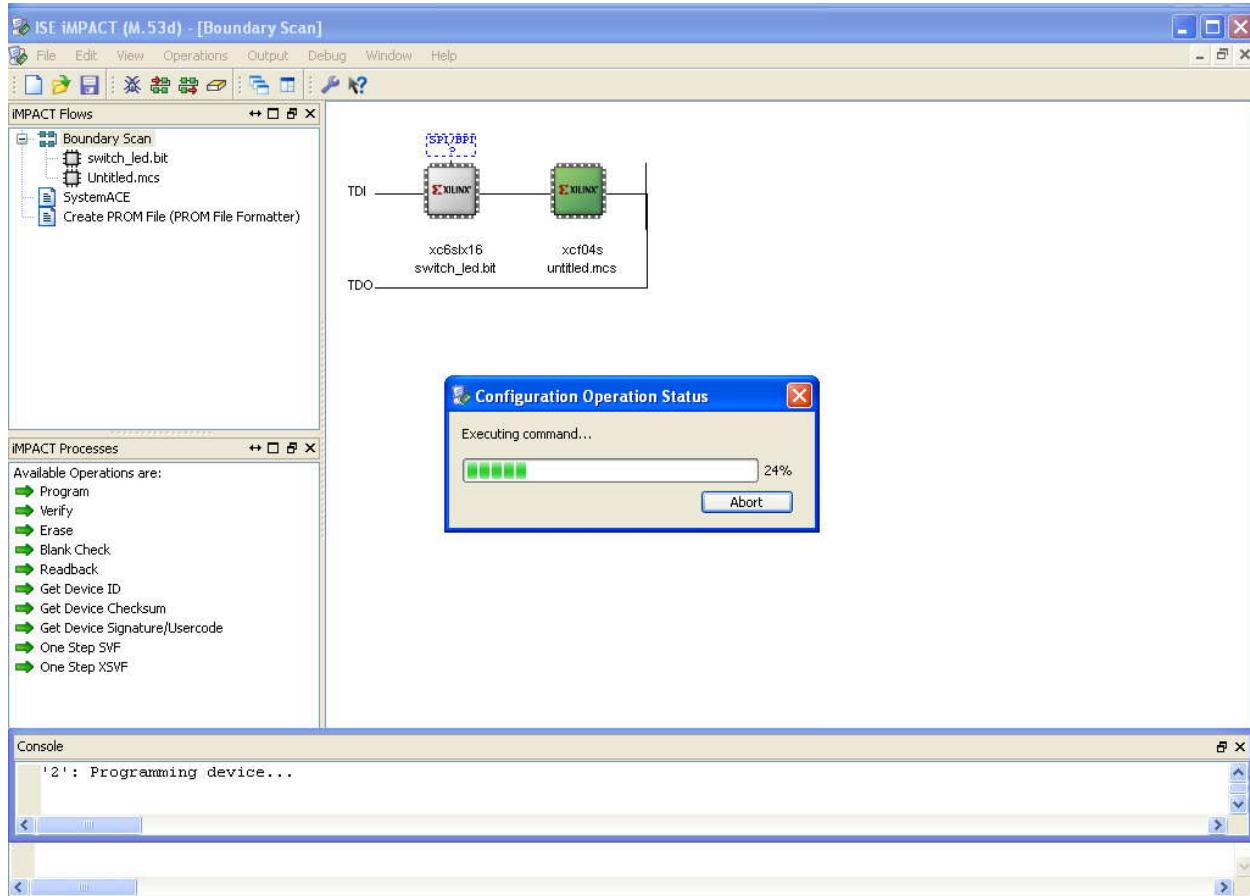


Figure - 59

SPARTAN 6 TRAINER KIT

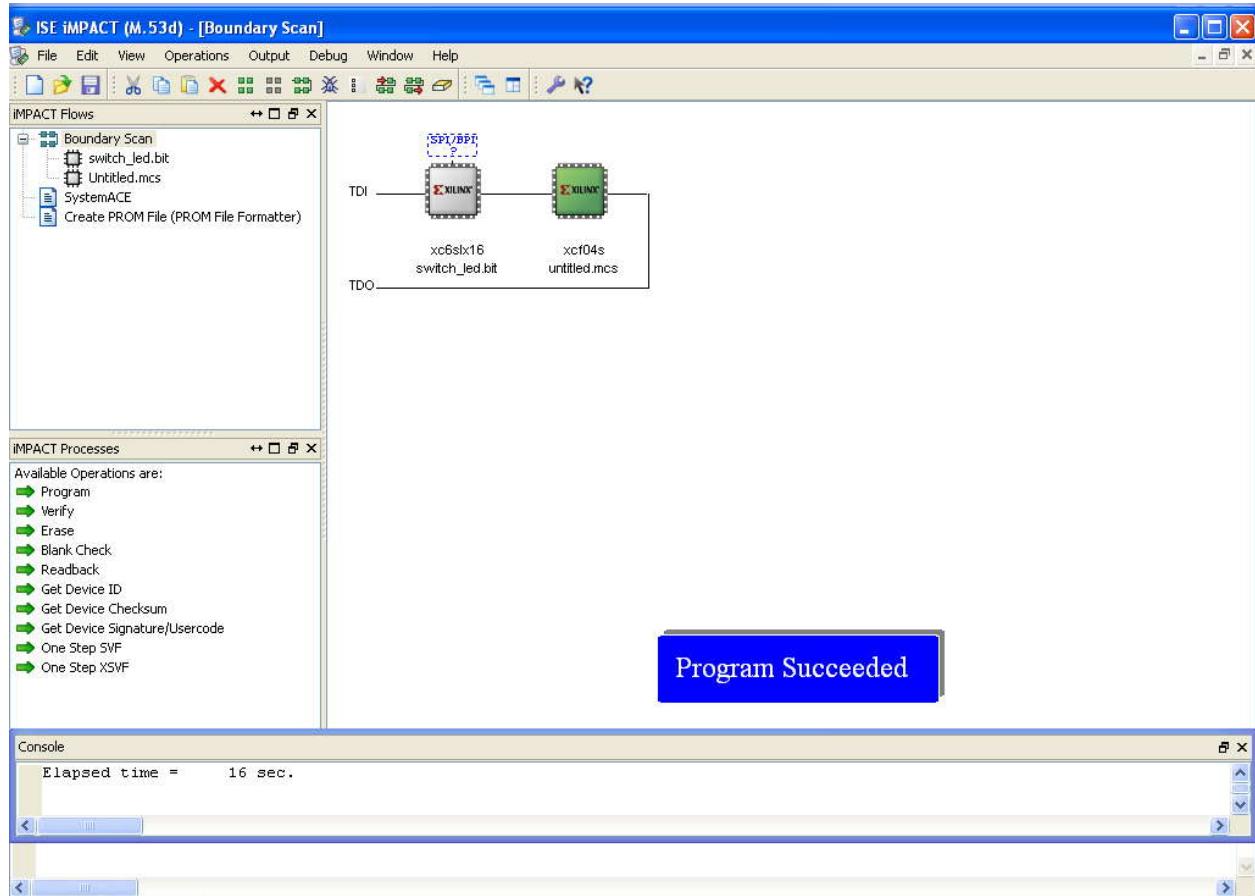


Figure – 60

Peripheral Testing Procedure

Switch & LED :

Download the .bit file from the swled program folder. If we set the switches high or low (SW22 to SW37) the corresponding LEDs (L40 to L55) are ON or OFF.

Matrix Key :

Switches SW5 to SW20 are arranged in 4x4 matrix keypad. Corresponding hex values are represented near to the switches. Download the matrix_key.bit file into FPGA from matrix_key program folder. When press the switch, corresponding hex value is displayed in LCD.

Limit Switch :

Download the .bit file from limit_switch program folder. Short the jumper J18 in upward direction, now the LED L55 will glow. When we press the limit switch SW21 the LED will off. Then short the jumper in downward direction, at this condition the LED L55 will glow when the switch SW21 is pressed.

Seven Segment Display :

Download the seven_segment.bit file from the program folder. If the segment is common cathode short the jumper J10 to ground (downward), if common anode short to +3.3V (upward direction). There are 16 switches (SW22 to SW37) are used in this program. 4-bit binary value is displayed in seven segment display.

| Switches | Display |
|-----------------|----------------|
| SW37 To SW34 | DISP1 |
| SW33 To SW30 | DISP2 |
| SW29 To SW26 | DISP3 |
| SW25 To SW22 | DISP4 |

LCD Display :

Download the .bit file from the LCD program folder. Now set the switch SW37 as low to allow the LCD to display the “**FLIGHT ELECTRONICS**” in first row in a continue manner in first row. When SW37 is set as high, the execution will be stopped.

Buzzer :

Download the .bit file from the buzzer program folder. Now the buzzer will produce the beep sound for a particular time period.

Relay :

Download the .bit file from the relay program folder. The switching sound of relay will indicate us to as the relay is working. We may connect the two LEDs in P5 2nd and 3rd pin. The LEDs will ON and OFF continuously for the particular time period.

RTC :

Download the .bit file from the rtc_lcd_vhdl program folder. The LCD will display the date and time details. These details are already set in the program.

ADC :

➤ External Input

Pot meter is used to give analog input to ADC channels and the ADC will support the voltage range up-to 3.3V only. Connect the 1st pin of pot meter to the 5th pin of P12 connector, 1st pin to 6th pin of P6 and 2nd pin to 1st pin of P12 (ADC Channel 1). If we want to give analog input to second channel of ADC, connect 2nd point of pot meter to 2nd pin of P12 connector. Download the adc.bit file from the ADC_ch1 program folder. Rotate the pot meter and the corresponding digital output is displayed in the 12 LEDs (L55 to L44).

➤ On-Board Analog Input

On-board TP5 trimmer is used to give analog input to ADC CH1 or ADC CH2 by using J16 jumper. When we short J16 pin 1 and 2, the trim-pot output is connected to ADC CH1 and when J16 pin 2 and 3 shorted, trim-pot output is connected to ADC CH2. Set the jumpers and download the .bit file from the corresponding ADC channel program folders. The output will be displayed in 16 LEDs (L55 to L40).

Temperature Sensor Input :

LM35 temperature sensor voltage output is given as a analog input to ADC CH1 by short the jumper J19. At that time we should not give external analog input and on-board trimmer analog input to ADC CH1. Download the .bit file from adc_temp program folder. Room temperature is displayed at initial condition. When we apply external heat to the sensor, that corresponding unknown temperature value is displayed in LCD in °C (Centigrade).

DAC :

Download the .bit file from dac_tri program folder. This program will execute the triangular output in both DAC channels (Channel 1 and 2). Connect the CRO probe positive to the 3rd pin (DAC1) or 4th pin (DAC2) of P12 connector and negative to 6th pin of P12 connector. Now the CRO will display the triangular waveform and that is generated by the DAC. Check also dac_sawtooth and dac_square program output.

Stepper Motor Interface :

Connect the stepper motor signal terminals to P11 5-pin RMC connector using 5-pin RMC wire. Download the .bit file from the stepper_motor program folder. Now the motor will rotate in clockwise direction. To run the motor in anti-clockwise direction download the .bit file from stepper_anti program folder.

DC Motor Interface :

Connect the DC motor negative to P7-D- point and positive to motor P7 D+ point. Download the .bit file from the dc_motor program folder. Now the motor will run in forward (clockwise) direction. If we want to run the motor in anti-clockwise direction, just change the motor connection together (positive to D- and negative to motor D+).

Traffic Light Controller (TLC) :

Download the .bit file from the tlc_board program folder. Set the switch SW37 as high. TLC signals are continuously run depending on the traffic light concept.

USB to UART Communication :

Download the .bit file from the UART program folder. Connect the USB cable to P4 connector. Open WinXTalk and set the baud rate as 9600 then open the communication window. Give the data input to FPGA via PC keyboard and FPGA will transmit this data to PC, which will be displayed in PC communication window.

20-pin Connectors :

Download the .bit file from header_20pin program folder. Connect the CRO probe negative to board ground and positive each I/O lines (one after another) in the 20-pin connectors. Verify that square waveform will displayed in CRO.