OpenEP4CE6-C User Manual

From Waveshare Wiki
Jump to: navigation, search

Overview

Basic operation and demo user guide of Waveshare FPGA Altera serial board[1] are in the present document for helping you quick start your FPGA development.

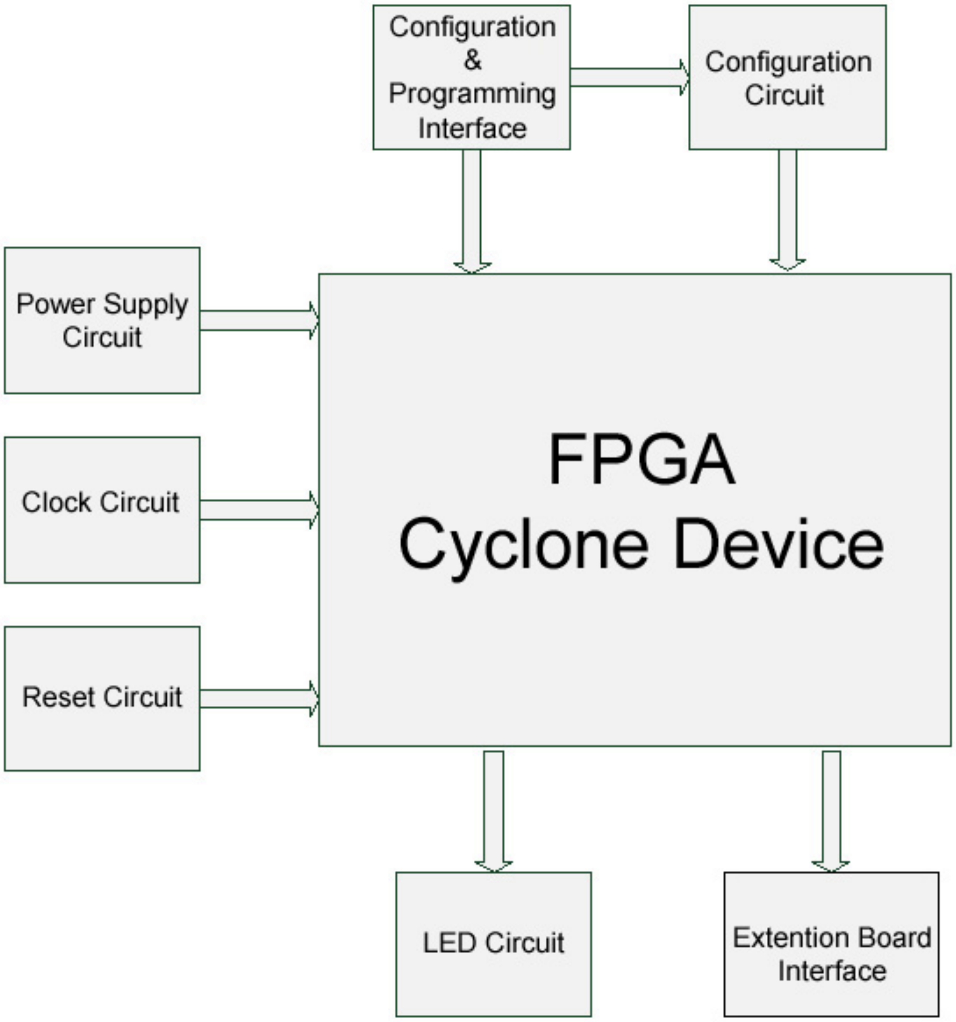
Hardware Design

This chapter mainly about the basic idea of core-board hardware design. Go with you to witness how a chip evolves to a board.

There are voltage regulator AMS1117, serial FLASH memory EPCS16, crystal oscillator, JTAG interface, LEDs, buttons, etc., beside main FPGA. Then, how and why do the devices connect together? What are their functions?

Framework of the Circuit

Core-board-circuit framework is shown in the following figure:

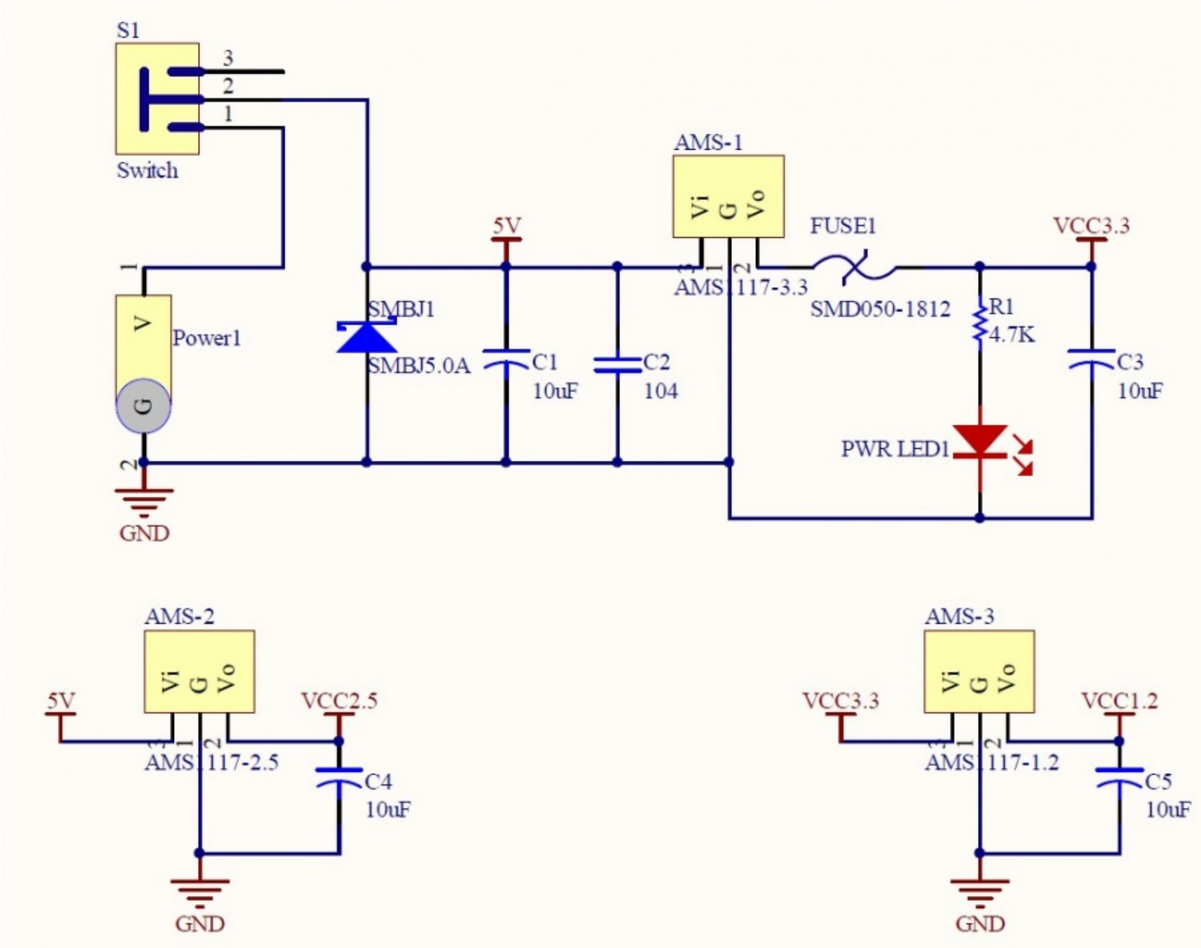


(/wiki/

File:FPGA_Cyclone_Device_Design.jpg)

Power Supply Circuit

Power Supply Circuit is the basic circuit for normal operation. You can find voltage supplies details from the File:Cyclone-IV-Device-Handbook.pdf (/wiki/File:Cyclone-IV-Device-Handbook.pdf). Noting that EP4CE6 requires 1.0V/1.2V for Internal core supply voltage (VCCINT) and PLL digital power supply (VCCD_PLL), requires 2.5V for PLL analog power supply (VCCA). Then I/O banks power supply VCCO can be connected to 1.2V, 1.5V, 1.8V, 2.5V, 3.0V, 3.3V to supply each area with different voltage standards. So for normal operation, the power supply of the board is designed for converting input voltage 5V to multiple voltages 3.3V, 2.5V, 1.2V. Meanwhile, a PWR LED is connected to 3.3V output, to meet the needs of checking power operation status. The schematic of the circuit:

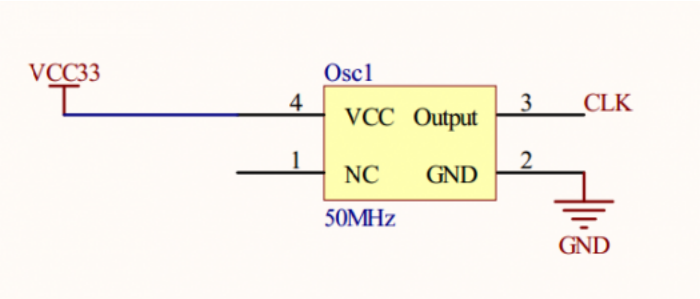


(/wiki/File:CoreEP4CE6-Power-Supply-Circuit.jpg)

Pin Name	Description
VCC5	5V supply voltage, External Input
VCC33	3.3V voltage, converted from AMS1117-3.3, is generally used to supply the voltage of clock, configure circuit, special features pin high, etc.
VCC25	2.5V voltage, converted from AMS1117-2.5, is generally used to supply the voltage of VCCA.
VCC12	1.2V voltage, converted from AMS1117-1.2, is generally used to supply the voltage of VCCINT, VCC_PLL, etc.

Clock Circuit

The best solution of FPGA clock circuit is: A main clock, which is driven by dedicated global clock input(GCLK), controls each timing device of the design. Try to use global clock at any design. The FPGA has dedicated global clock pin, which is connected to each register of the device. The shortest time span of GCLK can supply is used for delay. We use a global clock interface CLK in our design, because it is single clock interface, we consider the use of active crystal clock as an external clock source. A 50MHz crystal oscillator on board is used for supply accurate clock. The schematic of the circuit:

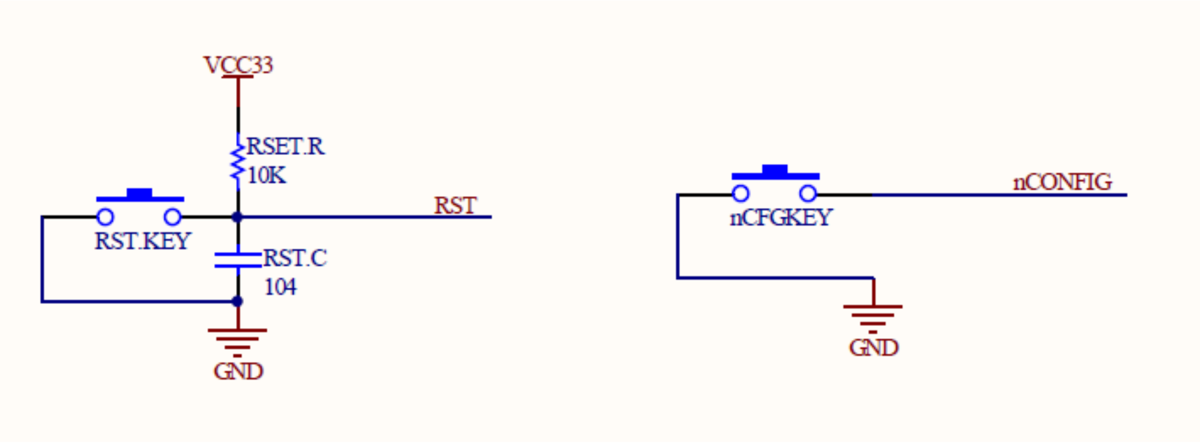


(/wiki/File:FPGA-Clock-Circuit.png)

Pin Name	Description
CLK	Clock input

Reset Circuit

The Reset Circuit contains RST Reset Circuit and nCONFIG Reconfigure Circuit. RST reset Circuit is a RC reset Circuit with RESET button switch, which is pressed to generate a Reset-signal, active-low. While nCONFIG Reset Circuit is triggered by nCONFIG key. FPGA will be reconfigured without reboot when the nCONFIG key is pressed. The schematic of the circuit:



(/wiki/ File:FPGA-RST-CONFIG-Circuit.png)

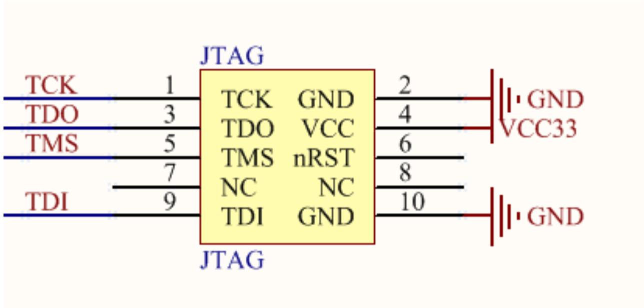
Pin Name	Description
RESET	Active-low Reset
nCONFIG	Active-low Reset, FPGA will be reconfigured, as soon as PROG_B pin restores to high level

Configuration/Programming Interface

Configuration is also known as loading and download. It is a process of FPGA programming. FPGA reconfigured at each reboot is a feature of SRAM-based FPGA. Within the FPGA, many programmable multiplexers, logic, interconnect nodes and RAM initialization, etc. are controlled by configuration data, which is stored in FPGA RAM.

The configured data of FPGA can be downloaded to target device with 3 methods, FPGA Active, FPGA Passive and JTAG, according to the role played in Configure Circuit. JTAG is an industrystandard interface. Usually All Altera FPGA can be configured via JTAG commands. Meanwhile, JTAG has more priority than other configuration method. Of course this board provides JTAG interface. The schematic of the circuit:

User can use dedicated Altera programmer USB Blaster (/wiki/USB_Blaster) to debug and program. File to be programmed to EPCS should be converted to .jic file by Quartus. That is, Set it as "JTAG Indirect Configuration File" and then uses the JTAG interface to program the EPCS device. See File:Cyclone-IV-Device-Handbook.pdf (/wiki/File:Cyclone-IV-Device-Handbook.pdf).

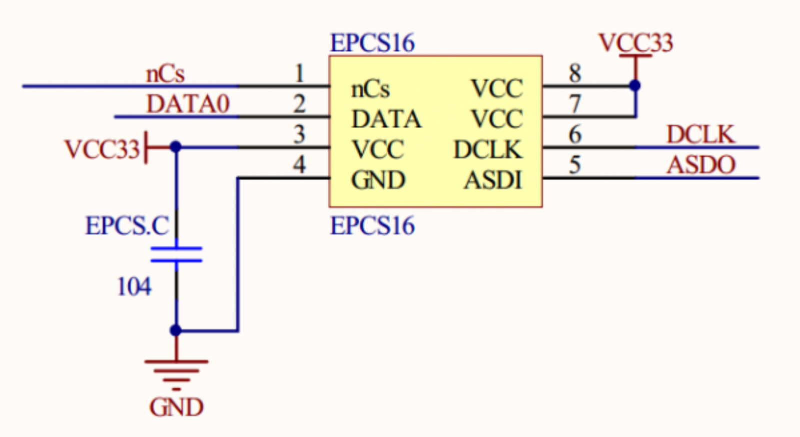


(/wiki/File:FPGA-JTAG-Circuit.png)

Pin Name	Description
TDI	Test data input. Serial input pin for instructions as well as test and programming data.
TDO	Test data output. Serial data output pin for instructions as well as test and programming data.
TMS	Test mode select. Input pin that provides the control signal to determine the transitions of the TAP controller state machine.
TCK	Test clock input. The clock input to the BST circuitry.

Configuration Circuit

An EPCS16, Altera EPCS serial flash devices, is connected to the FPGA for keeping the data without power supplied. EPCS16 is one of advanced Configuration Device, 16Mbit density. It supports high capacity single configuration of FPGA. It also support in-system programming by JTAG interface. The schematic of the circuit:

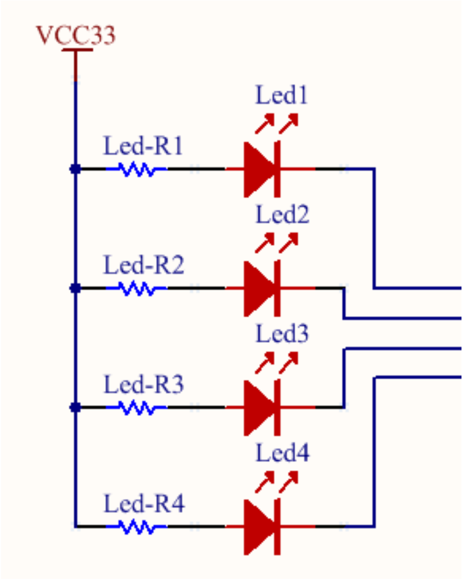


(/wiki/File:EPCS16-Config-

Circuit.png)

LED Circuit

4 LEDs onboard. Each LED is driven via one of the FPGA pins. When a low level inputs to LED pin, the corresponding LED turns on. The schematic of the circuit:



(/wiki/File:FPGA-LED-Circuit.png)

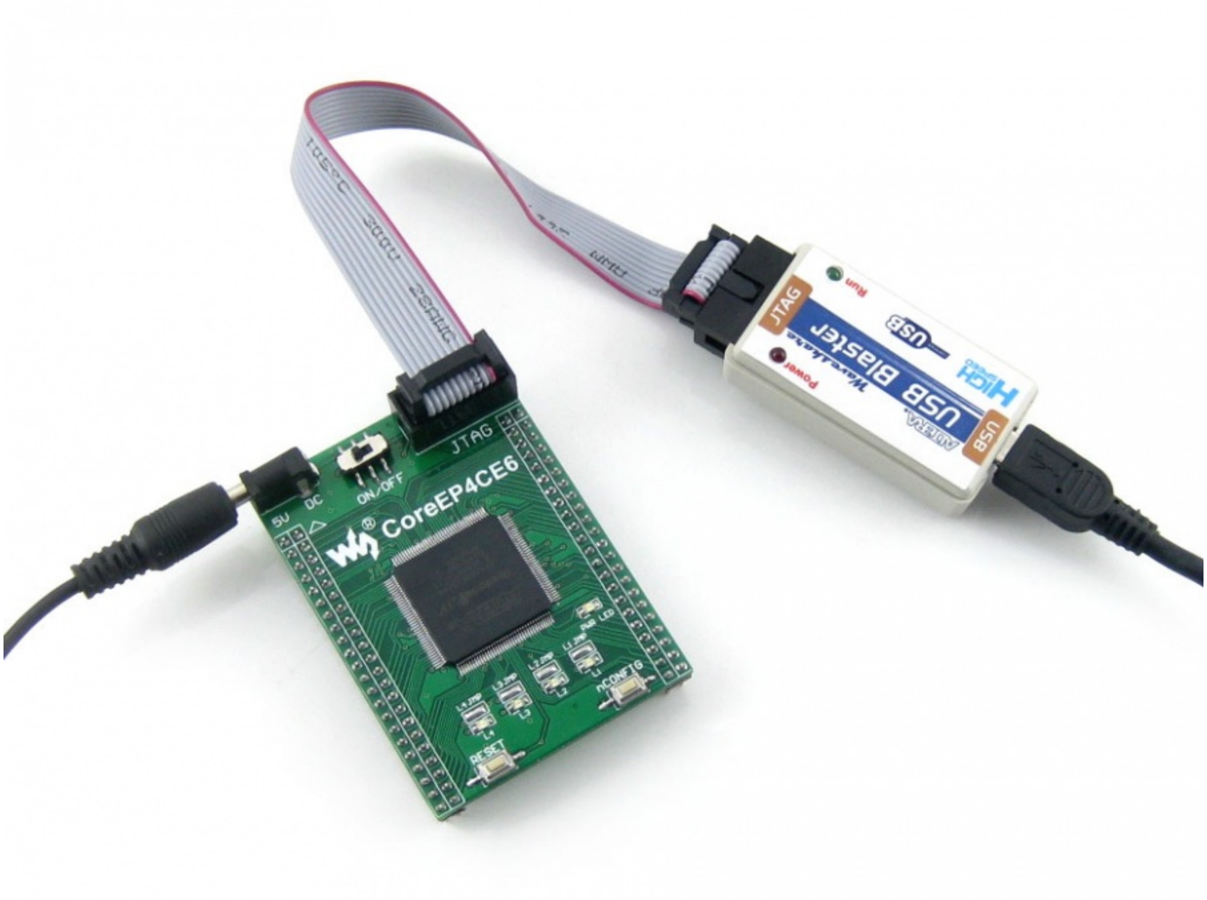
Extention Board interface

Series of Open boards designed by Waveshare are based on Core-Extension-Separated idea. On the one hand, users can easily design extension circuit according to their needs, On the other hand, the interfaces of core board and extension board are fully considered about the compatible to other FPGA boards, make update easier.

Basic Operation

Power Up And Download

Power up CoreEP4CE6 with 5V supply. Then the PWR_LED will light up in usual. The onboard JTAG interface is used for programming with dedicated programmer USB Blaster (/wiki/USB_Blaster), as shown in the following figure:



(/wiki/

File:CoreEP4CE6-Power-up-download.jpg)

If you use CoreEP4CE6 and OpenEP4CE6-C together, just connect the core board to the mother board, and plug a 5V adapter directly without any jumper wire. Turn the switch on to power up.

Run the software Quartus II to download the Verilog and VHDL demo:

General Download Process

1. Copy the Nios II processors (Quartus II projects) to your computer, which are located in ".\nios\Quartus II Project".

2. Launch Nios II IDE

1) Create a new Nios II project.

2) Config the Nios processor by specifying the PTF file directory on the "Select Target Hardware" section. The PTF file directory depends on where you placed the Nios II processors in step 1.

3) Copy the corresponding c code in ".\nios\Nios II C Code" into the new project.

4) Build the new Nios II project.

3. Connect the development board to the PC through a download cable.

4. Back to the Nios II IDE

1) Download the Nios II processor to the FPGA, select"Tools->Quaters II Programmer" to download the sof file.

2) Run the Nios II project (right click the makefile, select "Run ->Run As->2 Nios II Hardware", it may takes several minutes). When download completed, the demo code should starts to run automatically.

Quick Start

All the following demo require power supply. Verilog, VHDL and NIOS are used in the following demos, please download the corresponding one.

An OpenEP4CE10-C development board is used for demonstration, other Altera boards are similar to it. If there are differences on any example, they will be special explained.

Light Up LEDs

Language	Verilog	VHDL	Nios II C
Sample Program Name	LED		LED_hello_world
Steps	1. Connect the "SDRAM Board" to the SDRAM interface (required only for Nios II) 2. Download the program		
Phenomena	▪ Onboard 4 LEDs flow from the left to the right.		▪ LED keeps blinking; ▪ The NOIS II project Console tab shows "hello_world"

JOYSTICK Demo

Language	Verilog	VHDL	Nios II C
Sample Program Name	JOYSTICK		

Steps	1. Set the jumper JOYSTICK JMP to on. 2. Connect the “SDRAM Board” to the SDRAM interface (required only for Nios II) 3. Download the program
Phenomena	- LED status keeps changing according to the joystick pressed.

8 Push Buttons Demo

Language	Verilog	VHDL	Nios II C
Sample Program Name	8 Push Buttons		——
Steps	1. Connect the "8 Push Buttons" to 8I/Os_1 2. Download the program		
Phenomena	- LED turns on/off according to the pressed button		

8 SEG LED Board Demo

Language	Verilog	VHDL	Nios II C
Sample Program Name	8 SEG LED Board		——
Steps	1. Connect the "8 Push Buttons" to 16I/Os_2. 2. Download the program		
Phenomena	- Display 0~E on the "8 SEG LED board"		

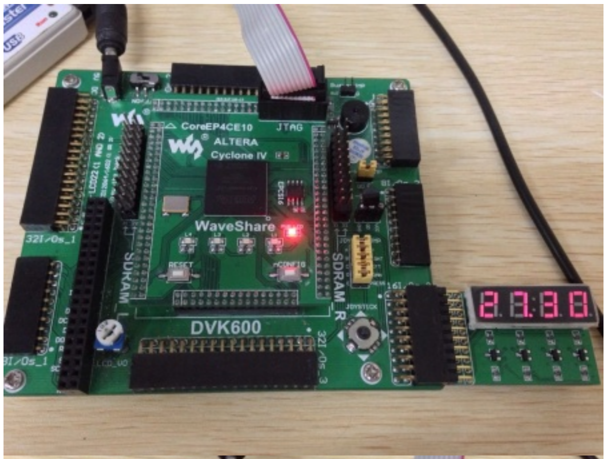
4x4 Keypad Demo

Language	Verilog	VHDL	Nios II C
Sample Program Name	4x4 Keypad		
Steps	1. Connect the "4x4 Keypad" to 8I/Os_1 2. Download the program	1. Connect the "4x4 Keypad" to 8I/Os_1 2. Connect the "8 SEG LED board" to 16I/Os_2 3. Download the program	1. Connect the "4x4 Keypad" to 8I/Os_1 2. Connect the “SDRAM Board” to the SDRAM interface 3. Download the program
Phenomena	▪ LED turns on/off according to the pressed button	▪ Display something on the "8 SEG LED board"	▪ LED turns on/off according to the pressed button

DS18B20 Temperature Sensor Demo

Language	Verilog	VHDL	Nios II C
Sample Program Name	DS18B20		

Steps	1. Connect the DS18B20+ to the ONE-WIRE socket 2. Set the jumper 1-WIREJMP to on 3. Connect the “8 SEG LED Board” to the 16I/Os_2 4. Download the program	1. Connect the DS18B20+ to the ONE-WIRE socket 2. Set the jumper 1-WIREJMP to on 3. Connect the LCD1602 to the LCD1602 interface, The LCD1602 pin 1 should correspond to the PCB printed mark "1" beside the onboard LCD connector 4. Download the program	1. Connect the DS18B20+ to the ONE-WIRE socket 2. Set the jumper 1-WIREJMP to on 3. Connect the “SDRAM Board” to the SDRAM interface 4. Reboot the board
Phenomena	▪ LCD1602 displays temperature	▪ "8 SEG LED board" displays temperature	▪ The Nios II Console tab displays temperature



(/wiki/File:Altera-DS18B20-Verilog.jpg)

(/wiki/File:Altera-DS18B20-VHDL.jpg)

Buzzer Demo

Language	Verilog	VHDL	Nios II C
Sample Program Name	Buzzer/PWM		——
Steps	1. Set the jumper Buzzer JMP to on 2. Download the program		
Phenomena	▪ Buzzer will buzz.		

PS/ 2 Keyboard Demo

Language	Verilog	VHDL	Nios II C
Sample Program Name	PS2		
Steps	1. Connect the "VGA PS2 Board" to 16I/Os_2, and connect a keyboard 2. Connect the LCD1602 to the LCD1602 interface, The LCD1602 pin 1 should correspond to the PCB printed mark "1" beside the onboard LCD connector 3. Download the program		1. Connect the "VGA PS2 Board" to 16I/Os_2, and connect a keyboard 2. Connect the “SDRAM Board” to the SDRAM interface 3. Download the program 4. Reboot the board
Phenomena	▪ Characters inputted from keyboard are displayed on LCD1602		▪ The corresponding ASCII inputted from keyboard are displayed on Console tab of Nios II software

VGA monitor Demo

Language	Verilog	VHDL	Nios II C

Sample Program Name	VGA_color	VGA	——
Steps	1. Connect the "VGA PS2 Board" to 16I/Os_2, and connect a VGA monitor 2. Download the program		
Phenomena	▪ Display something on the VGA monitor		

LCD1602 Demo

Language	Verilog	VHDL	Nios II C
Sample Program Name	LCD1602		——
Steps	1. Connect the LCD1602 to the LCD1602 interface, The LCD1602 pin 1 should correspond to the PCB printed mark "1" beside the onboard LCD connector 2. Download the program		
Phenomena	▪ Display something on the "LCD1602"		

(/wiki/File:Altera-LCD1602-Verilog.jpg)

(/wiki/File:Altera-LCD1602-VHDL-2.jpg)

(/wiki/File:Altera-LCD1602-VHDL-1.jpg)

(/wiki/File:Altera-LCD1602-VHDL-3.jpg)

LCD12864 Demo

Language	Verilog	VHDL	Nios II C
Sample Program Name	LCD12864	——	——
Steps	1. Connect the LCD12864 to the LCD12864 interface 2. Download the program		
Phenomena	▪ Display something on the "LCD12864"		

(/wiki/File:Altera-LCD12864-Verilog.jpg)

LCD32 touch screen Demo

Language	Verilog	VHDL	Nios II C
Sample Program Name	——	——	LCD32
Steps			1. Connect the "3.2inch 320x240 Touch LCD" to 32I/Os_2 interface via 3.2 inch LCD Adapter(B) 2. Connect the “SDRAM Board” to the SDRAM interface 3. Download the program 4. After program downloaded, reboot the board.

Phenomena			- Display something on the ""3.2inch 320x240 Touch LCD"", interact with it by touching (power should be reset before downloading the demo code each time)

USB Communication Demo

Language	Verilog	VHDL	Nios II C
Sample Program Name	——	USB	——
Steps		1. Connect the "CY7C68013A USB Board" to 32I/Os_1, then connect it to PC 2. Download the program	
Phenomena		- Control the onboard LED via USB_LED.exe	
Remark		- Before using CY7C68013A USB Board, please install driver EZ-USB.exe	

SD-Card Demo

Language	Verilog	VHDL	Nios II C
Sample Program Name	——	——	SD-Card
Steps			1. Connect the "Micro SD Storage Board" to 8I/Os_1 2. Connect the “SDRAM Board” to the SDRAM interface 3. Download the program 4. Reboot the board

Phenomena			- Place the MESSAGE.TXT file into SD card first, run the demo code, a new file hello.txt will be created. - The contents in MESSAGE.TXT will be displayed on the Console tab.

Ethernet Control Demo

Language	Verilog	VHDL	Nios II C
Sample Program Name	——	——	ENC28J60
Steps			1. Connect the ""ENC28J60 Ethernet Board"" to 16I/Os_2, then connect it to the PC through an ethernet cable 2. Download the program
Phenomena			- The Internet Explorer shows information, follow the wizard on Internet Explorer - Numbers will be displayed on the Nios II Console tab

UART Demo

Language	Verilog	VHDL	Nios II C
Sample Program Name	UART		

Steps	1. Connect the "RS232 Board" to 8I/Os_1, then connect it to PC 2. Download the program	1. Connect the "RS232 Board" to 8I/Os_1, then connect it to PC 2. Connect the "8 SEG LED Board" to 16I/Os_2 3. Download the program	1. Connect the "RS232 Board" to 8I/Os_1, then connect it to PC 2. Connect the "SDRAM Board" to the SDRAM interface 3. Download the program
Phenomena	- Launch Serial Port Monitor, select a proper COM port, config the speed as 9600 - Send any character/number via "Serial Port Monitor", and it should be sent back and displayed on "Serial Port Monitor" again.	- Launch Serial Port Monitor, select a proper COM port, config the speed as 9600 - Send any number via "Serial Port Monitor", and it should be displayed on the "8 SEG LED Board"	- Switch to the Nios II Console tab, send any character, then send "t", the "t" will be detected. - Send "v" to close serial port debugging.

I2C EEPROM Demo

Language	Verilog	VHDL	Nios II C
Sample Program Name	AT24CXX		

Steps	1. Connect the “AT24CXX EEPROM Board” to the 8I/Os_1 interface 2. Connect the “8 SEG LED Board” to the 16I/Os_2 interface 3. Download the program	1. Connect the “AT24CXX EEPROM Board” to the 8I/Os_1 interface 2. Connect the “8 SEG LED Board” to the 16I/Os_2 interface 3. Connect the “SDRAM Board” to the SDRAM interface 4. Download the program
Phenomena	▪ Use the "8 SEG LED board" to display the data sent from SDRAM_L pins 3, 5, 7, 9 to "AT24CXX", and the data "AT24CXX" received, using one 8-segment digit for each data. ▪ Short the SDRAM_R pin 5 and pin 6, then open them again, lastly short the SDRAM_R pin 3 and pin 4,data will be written into AT24CXX and then read from it to display on the "8 SEG LED board".	▪ The LEDs as a binary number will be increased by 1 each time.
Remark	▪ In the demo code, initially the SDRAM_L 3, 5, 7, 9 is set as pull-up input high-level, the SDRAM_L 4, 6, 8, 10 is set as output low-level. It is possible to config the written data by shorting the pins 3-4, 5-6, 7-8,9-10 respectively using jumper caps.	

AT45DB Demo

Language	Verilog	VHDL	Nios II C

Sample Program Name	——	——	AT45DBXX
Steps			1. Connect the "AT45DBXX DataFlash Board"to the 8I/Os_1 interface 2. Connect the “SDRAM Board” to the SDRAM interface
Phenomena			▪ The LEDs as a binary number will be increased by 1 each time. ▪ Explanation: The program writes data(0~255) to AT45DB, then read from it. The data is displayed as binary form.

PCF8563 Demo

Language	Verilog	VHDL	Nios II C
Sample Program Name	——	——	PCF8563
Steps			1. Connect the "PCF8563 RTC Board" to 8I/Os_1 2. Connect the “SDRAM Board” to the SDRAM interface
Phenomena			▪ The Nios II Console tab displays time

FT245 Demo

Language	Verilog	VHDL	Nios II C
Sample Program Name	——	——	FT245

Steps			1. Connect the "FT245 USB FIFO Board" to 16I/Os_2, then connect it to PC 2. Connect the “SDRAM Board” to the SDRAM interface 3. Download the program
Phenomena			▪ Send any character/number via "Serial Port Monitor", and it should be sent back and displayed on "Serial Port Monitor" again.

References

1. FPGA Altera Serial (http://www.waveshare.com/product/fpga-tools/altera.htm)

Retrieved from "https://www.waveshare.com/w/index.php?title=OpenEP4CE6-C_User_Manual&oldid=4501 (https://www.waveshare.com/w/index.php?title=OpenEP4CE6-C_User_Manual&oldid=4501)"