Indian Institute of Technology,
Bombay

B.Tech Technical Project

Department of Electrical Engineering

Low cost FPGA development Board

Author:

Kamal Galrani

Manmohan Mandhana

Guide:

Prof. Joseph John

Prof. Dipankar

Prof. Shankar Balachandran

December 9, 2015

Abstract

This project aims at designing a low cost FPGA development board which would used to run complex algorithm using partial reconfiguration on a collection of low density FPGAs (Field Programmable Gate Array) which are controlled by a CPLD (Complex Programmable Logic Device). The completed project will be able to take data over ethernet, process it and transmit it back over ethernet. High complexity architectures which take up enormous amount of resources will be broke down into smaller sections which can be uploaded on the low density FPGAs. The real time programming of the FPGAs will be handled by the CPLD. By breaking the complex algorithm into chunks we can perform the computation on our board which otherwise will require the costly high density FPGAs.

1 Introduction

A Field Programmable Gate Array or FPGA is an integrated circuit that could contain millions of logic gates that can be electrically configured to perform a certain task. The very basic nature of FPGAs allows it to be more flexible than most micro-controllers. FPGAs are capable of performing parallel processing. Partial re-configuration facilitates the user to re-program only a part of FPGA. This feature has many functionalities as it allows the FPGA to be configured partially and dynamically. But partially Reconfigurable FPGAs are extremely costly and the partial re-configuration functionality is not available in the low end FPGAs. Although, the low end FPGAs do have complete re-configuration functionality using parallel data transfer which makes it very fast. Exploiting the low cost and fast reconfiguration of low end FPGAs we are trying to make a development board which will allow the user to run larger programs on smaller FPGA. The larger program will be broken into smaller pieces which can run on the smaller FPGA. And using a controller CPLD we will re-configure the FPGAs in real time to perform the next functionality. This way we are able to mimic a larger FPGA with partial reconfiguration functionality in relatively low cost.

2 Technical Background

FPGA can be configured in a number of ways. For the reconfigurability, the configuration should be fast. Altera FPGAs can be configured via JTAG, Passive Serial, Active Serial, Fast Passive Parallel and Active Parallel. Passive schemes require an external host controller whereas active schemes read a memory by themselves. Among these schemes FPP is the fastest as it allows 8-bit parallel programming at 133MHz for Cyclone IV device. This means 3ms of configuration time if we have a fast enough memory. We chose a 16bit parallel flash for our purpose. It allows for 6ms configuration time. Faster memories are expensive so we made a compromise with the timing. Even at 6ms it allows for 5 stages in a 30fps pipeline. Each stage will have a maximum time limit on execution as 6ms and maximum resource limit as 15k logic elements.

2.1 Block Diagram

The development board is broken down into two main components :

- Main mother-board
  - This Board is where all the controlling and interfacing components are placed.
  - It contains all the peripherals interfacing chips such as Ethernet, Audio, Video, JTAG along with the Master CPLD and the power circuitry
- Daughter board
  - This Board is where the core computation occurs
  - This board contains both the FPGAs and the SRAM memory chip

The block diagram of the board including the daughter board is shown in the image below. The connector between the daughter board and the Main mother-board is a SODIMM DDR3 Connector which is generally used for connecting DDR3 RAM. As the connector supports high speed data transfer it is suitable for our use case.

We have also incorporated an Audio ADC ic and a Video decoder ic which are connected directly to the FPGAs and thus can be used to provide direct audio and video input data to the FPGA

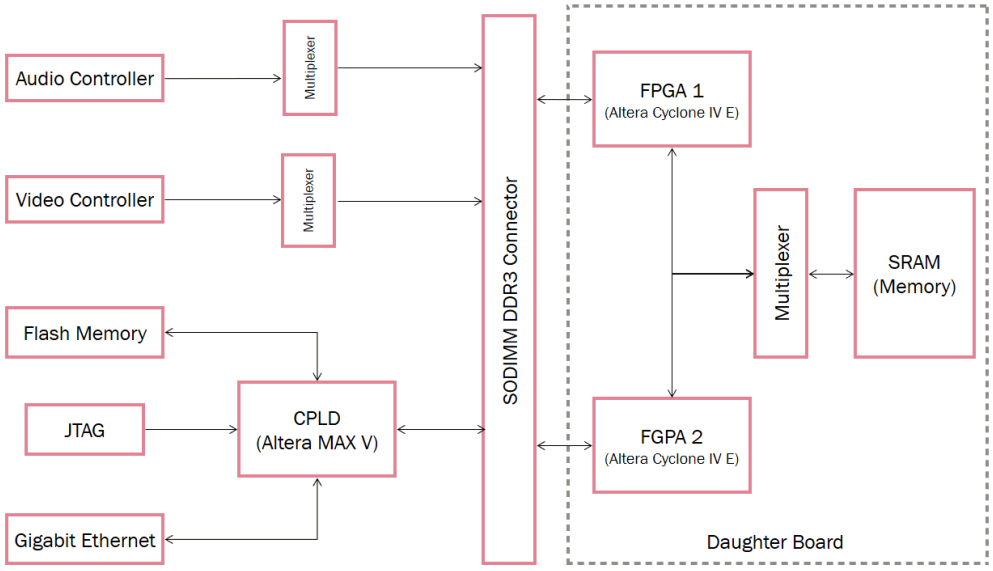

Figure 1: Block Diagram of Development Board

3 Project Implementation

3.1 Main mother-board

The main mother-board consists of the various interfacing peripherals and the Master CPLD. In this section we will explain the use of all the components on this board.

3.1.1 Master CPLD (Altera Max V)

The Master CPLD is used to configure the FPGAs on the daughter board in real time. The configuration files to be uploaded on the FPGAs are received by the Max V via Ethernet and are stored in the flash memory. The Max V also acts as a middle man between the FPGAs and the ethernet interface. The programming of the Max V is done using the JTAG interface via the on board FT2232(multi-purpose UART/FIFO controllers).

3.1.2 Gigabit Ethernet

The Gigabit ethernet is used as the main medium of data transfer between the development board and the user. We are using the ethernet controller IC called CP2200 by Silicon Labs which has a 8-bit parallel hosting interface providing upto 30 Mbps data transfer rate. The ethernet controller IC is connector the ethernet port on one end and the Max V on the other end. Ethernet is used in this development board to transfer the configuration files to Max V and to transfer the data to be processed by the FPGAs. All the data to the FPGAs is sent via the Max V CPLD.

3.1.3 JTAG Interface

To configure the Max V it requires a JTAG connection. For this purpose we have placed a multi-purpose USB controller (FT2232) which can be used to configure Max V using JTAG via a USB port. The same mechanism is used on the Krypton Board developed by WEL.

3.1.4 Flash Memory

The configuration files for the Cyclone IV E FPGAs are 3000 Mbits in size. Therefore a flash memory is connected to the Max V so that it can access the configuration files whenever it needs to reprogram the FPGAs. Flash memory usually have a slow write period but a fast read period. In our application we will write the configuration files in the flash at the start of the program and then during configuration we only need to read the data from the flash which helps in maintaining the configuration speed.

The Flash memory we are using is the SST39VF1602C by Microchip. This is a 16Mbits parallel flash memory with a 16bit data width. There is another flash by Microchip which has the same pinmap but it has 64Mbits size and thus anyone of these two can be used on the mother-board.

3.1.5 Power Circuitry

For the input power circuitry of the board we have used the TPS65400 by Texas Instruments which is a I2C controller Power Management unit with flexible input voltage from 4.5V - 18V. We are taking three outputs from this IC of 3.3V, 1.8V and 1.2V. The TPS65400 has 95% efficiency for each switching regulator.

3.1.6 Audio Controller

The mother-board contains a ADAU1977 by Analog Devices, which is a ADC usually used for audio applications. This ic is connected to both the FPGAs via a multiplexer and thus can be used to take input audio data to either of the FPGAs. We have provided a RCA input port for the audio in signal.

3.1.7 Video Controller

The mother-board also contains a ADV7180 by Analog Devices, which is a Video decoder chip. This ic is connected to both the FPGAs via a multiplexer and thus can be used to provide input analog video data to be processed data to either of the FPGAs. We have also provided a RCA input port for the analog video in signal.

3.1.8 SODIMM DDR3 Connector

As there many high speed connections between the mother-board and the daughter board, we have used a SODIMM DDR3 Connector by TE for connecting the daughter board.

3.2 Daughter Board

3.2.1 FPGAs (Cyclone IV E)

The FPGAs are the core computing units in this project. We have used the EP4CE15F17C8N by Altera which has 15,408 Logic Elements. The 15,000 logic elements helps to make the program chunks sufficiently large enough to contain a reasonable section of the program. We have used the FBGA-256 package which allows us to use the Fast Passive Parallel (FPP) mode of configuration for the FPGAs. The two FPGAs share all the resources via a multiplexer which is controlled by the Max V. Thus any of the FPGAs can access the audio, video and the SRAM without overloading the other FPGA's pins.

3.2.2 SRAM

The FPGAs cannot store large data as they have limited number of logic elements and no inbuilt memory unit. For this purpose we have connected an 4Mbit (128K x 36) SRAM CY7C1347G by Cypress via multiplexers. This

SRAM can provide output at 250MHz and thus is useful for the FPGAs while performing fast computations. The 36bit wide data line helps in increasing the data bandwidth between the FPGA and the SRAM.

4 PCB Designs

Below are the PCB design that we made for the Main mother-board and the Daughter Board

4.1 Main mother-board PCB

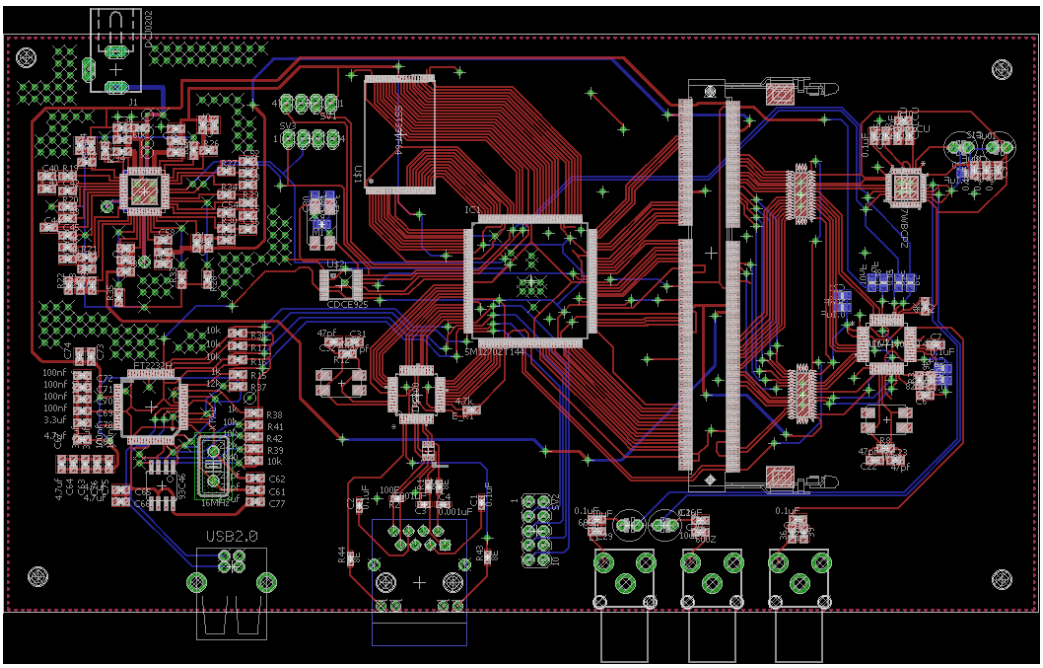

Figure 2: PCB layout of the motherboard

4.2 Daughter board PCB

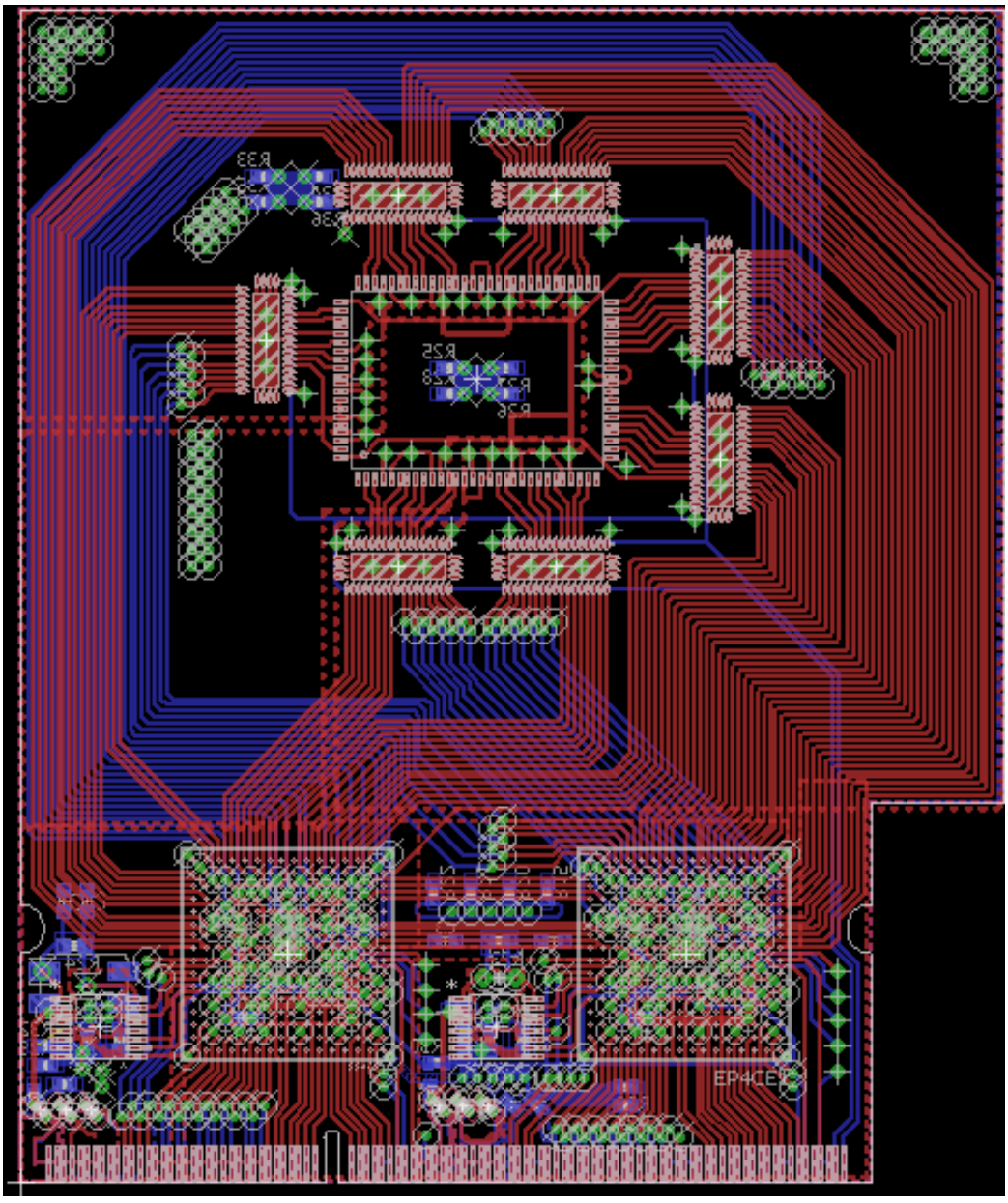

Figure 3: PCB layout of the Daughter Board

5 Work Done

We have completed the following work till now :

- Completed the PCB designs of both the daughter and the mother board along with acquiring all the components required for both the boards
- Sent over both the PCB files to PCB power for printing of which we have only received one board till now (main mother board)
- Written VHDL test programs for
  - Flash Controller
  - Reconfiguration with Max V
  - SRAM controller
  - UART driver
  - I2C driver
- Soldered the components on the Main Mother Board and the Max V was working on the board along with the PLL circuit. But due to some issue currently we are not able to program the Max V.

6 Work to be done

We have to complete the following work :

- Acquire the daughter board from PCB Power
- Debug the current mother board for the problem
- Solder and test all the components on the Daughter Board
- Use the test programs written for reconfiguration with both the boards connected together

7 References

1. Altera's Documentation for Cyclone-IV
2. Altera's Documentation for MAX-V