FPGA Design Guide

DG00003

Rev: A01

Jinzhi Cai

2019–07–18

Table of Contents

1 Introduction

In this guide, we will discuss FieldProgrammable Gate Array and System on Chip technologies and relative application. Due to the need, ORBiT Avionics department desided to create next generation bus system for the rocket. For help club member to learn and understand this technology, this guide will introdute the basic knowledge of FPGA and SoC.

1.1 FPGA and SoC

1.1.1 FPGA

test2

1.1.2 SoC FPGA

SoC FPGA is a new kind of FPGA chip which embedded a hardware processor inside to improve performent. In a SoC FPGA, there are two parts. Programmable Logic(PL) is the part that use FPGA technology which allow user to create costum logic, and Processor System(PS) is an embedded hardware processor to provide computing power. Cyclone V(Intel) and ZYNQ 7000(Xilinux) are two most known SoC FPGA platform. They both use ARM Cortex processor and its AXI bus system.

1.2 What is VHDL

1.2.1 Basic Structure

The example following is a example format entity in VHDL. When writing entity, the name for the entity should be unique and match with the name in the architecture. Architecture name and port name should also unique.

```vhdl
library ieee; -- import libraries
use ieee.std_logic_1164.all; -- use standard library

entity <entity_name> is port(

        <name>  : <in/out/inout> <type>;
        <name>  : <in/out/inout> <type>

);
end <entity_name>;

architecture <arch_name> of <entity_name> is
        -- this is a comment
        -- add signals and other variables
```

```vhdl
begin

        -- VHDL sentence

end <arch_name>;
```

This is a example design for a Single port RAM.

```vhdl
-- Quartus Prime VHDL Template
-- Single port RAM with single read/write address

library ieee;
use ieee.std_logic_1164.all;

entity single_port_ram is

        generic
        (
                DATA_WIDTH : natural := 8;
                ADDR_WIDTH : natural := 6
        );

        port
        (
                clk: in std_logic;
                addr: in natural range 0 to 2**ADDR_WIDTH - 1;
                data: in std_logic_vector((DATA_WIDTH-1) downto 0);
                we: in std_logic := '1';
                q: out std_logic_vector((DATA_WIDTH -1) downto 0)
        );

end entity;

architecture rtl of single_port_ram is

        -- Build a 2-D array type for the RAM
        subtype word_t is std_logic_vector((DATA_WIDTH-1) downto 0);
        type memory_t is array(2**ADDR_WIDTH-1 downto 0) of word_t;

        -- Declare the RAM signal.
        signal ram : memory_t;

        -- Register to hold the address
        signal addr_reg : natural range 0 to 2**ADDR_WIDTH-1;

begin

        process(clk) -- process block indecate the code will synchronize with the clock.
        begin
        if(rising_edge(clk)) then -- if block indecate only when clock have a rising edge,
                if(we = '1') then
                        ram(addr) <= data;
                end if;

                -- Register the address for reading
                addr_reg <= addr;
        end if;
        end process;

        q <= ram(addr_reg);

end rtl;
```

1.3 Introduction of Quartus Prime

test3

1.4 Introduction of Vivado Design Suite(TBD)

1.5 Introduction of Embedded Linux

1.5.1 Embedded Linux

1.5.2 Common Use Command

1.6 Introduction of Hardware Bus

1.6.1 I2C

1.6.2 SPI

1.6.3 UART

2 Improtant Circuit Design Requirement

2.1 SoC FPGA Power and CLock Design

2.2 SoC FPGA JTAG Programming Design

3 Quartus Prime SoC FPGA Development

3.1 Hello World for FPGA

test

3.2 Create a Complex Entity

test

3.3 Introduction of Platform Designer(QSYS)

test

3.4 Introduction of NIOS II and HPS

In Platform Designer, it majorly has two type of processor core; softcore and hardcore. Softcore processor only require LUT(Look Up Table) resource to perform. Hardcore processor require special design processor circuit to function. In resource, softcore processor have a adventage due to it only need about 3000 LUT to function, but the trade off is the performent is worse than hardcore processor. In Cyclone V series, it contain an ARM Cortex A9 hardcore processor and over 30000 LUT for extra IP core.

3.4.1 Recommend Configration for NIOS II(Softcore)

3.4.2 Recommend Configration for HPS(ARM Cortex A9)

3.4.3 How to create a costum IP core

3.5 Introduction of Avalon Memory Mapping Bus and AXI4 Bus

3.5.1 Timming Diagram for Avalon Memory Mapping Bus

3.5.2 Timming Diagram for AXI4 Bus

4 Vivado Design Suite SoC FPGA Development(TBD)