VGA

Introduction

Using VGA we could display some colors on a monitor. The circuit contains a counter which provides dimensions of the display (resolution) and the top level module (which provides the colors and their distribution on the monitor).

Top level module

module vga3 (

input clk, //the clock

input reset, // reset signal

input [2:0] R,G,B, //colors

output [11:0] xPOS, //horizontal position

output [11:0] yPOS, //vertical position

output Display\_active, // flag for active display

output reg [3:0] OR,OG,OB, // colors shown on display

output Hsync, // horizontal synchronization

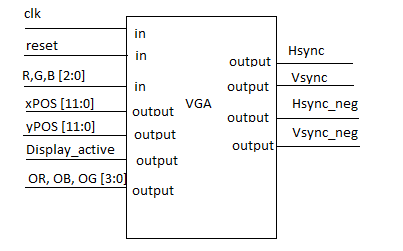
output Vsync, // vertical synchronization

output Hsync\_neg, //negative horizontal synchronization

output Vsync\_neg //negative vertical synchronization

);

Schematics



Conclusions

VGA implementation on FPGA supposed to define the active zone on a monitor in order to display different colors. The counter module established the active zone(where will be display colors) and non–active zone. This module was used twice in the top module in order to define vertical and horizontal dimensions of the screen.

PS/2 Keyboard Interface

Introduction

PS/2 is an interface for keyboards and mice to computer. A serial flux of data is transmited from keyboard to a shift register and after some verification, we will have an output bus of 8 bits.

Top level module

module ps3 ( input clk, //clock

input reset, //reset from a button

input ps2\_data, //data from keyboard

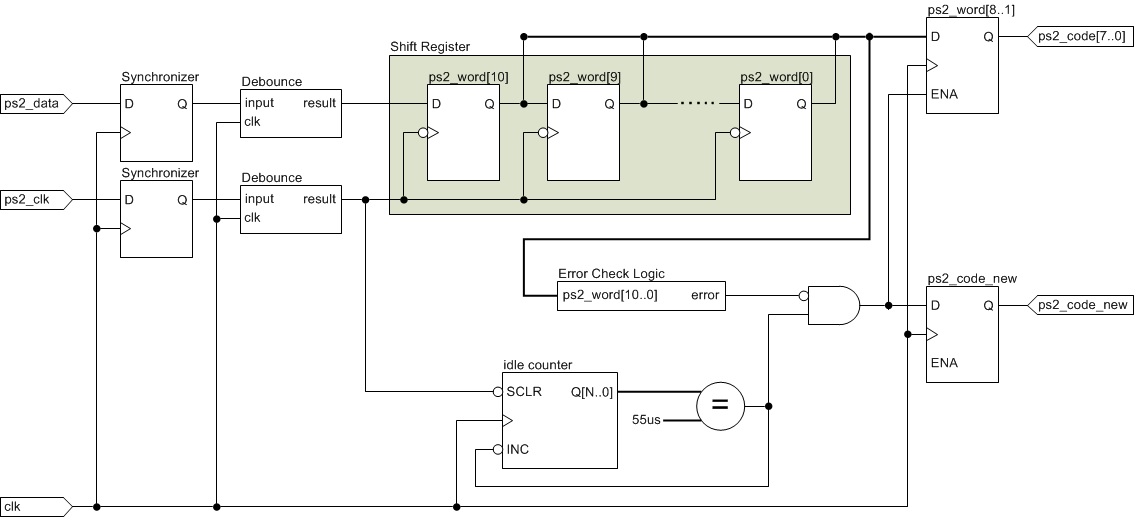
input ps2\_clk, //keyboard's clock

output [6:0] ps2\_code1, //output data from register

output [6:0] ps2\_code2 //output data from register

);

Schematics



Conclusions

This PS/2 keyboard interface is a programmable logic component that receives transactions from PS/2 keyboards.  It synchronizes the clocks domains, debounces the input signals, performs error checking, and notifies the user logic when new codes from the keyboard are available on its parallel output bus.

Controller SRAM

Introduction

This circuit is used for reading data from SRAM integrated on the FPGA or to write data on the SRAM integrated on the FPGA.

Top level module

module controll (input clk, //clock

input write\_enable,

input [3:0] data, //data from switch

input [3:0] adresa, //address from switch

output [17:0] date, //data out from ram

output reg WE,// write\_enable

output reg OE, //output\_enable

output reg CE, //chip enable

inout [15:0] IO, //bidirectional flux

output reg [17:0] A //address from sram

);

Schematics

