

# PROJECT PROPOSAL

## FPGA PAINT

---



**LUND**  
UNIVERSITY

Design of Embedded Systems, Advanced Course

Faculty of Engineering (LTH)

Lund University

Submitted to: Flavius Gruian

Project Members:

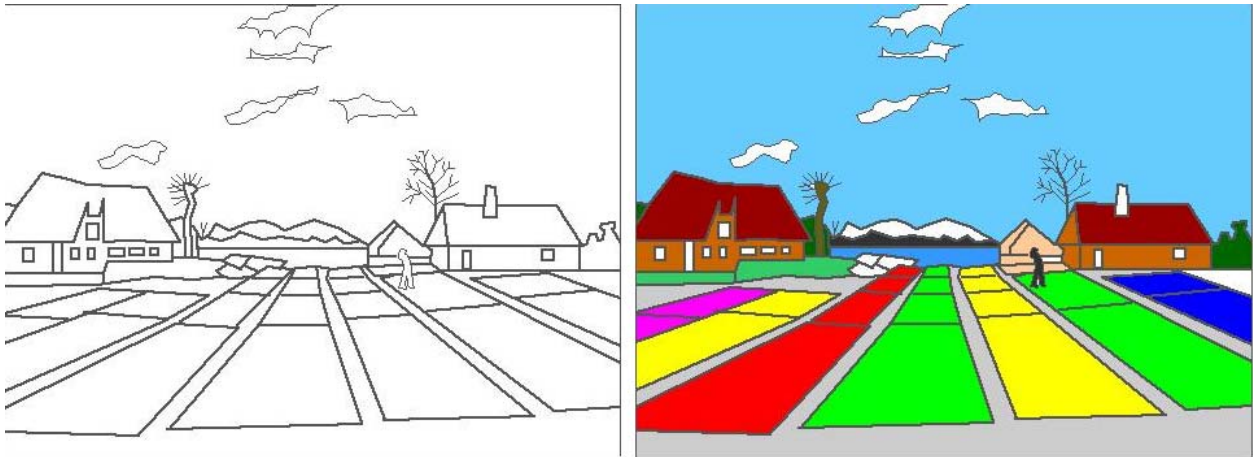
Viktor Do (dt06vd2)

Kazi Asifuzzaman (sx09ak8)

Zhang Dalong (eit06dzh)

## Introduction

We have decided to develop a paint program on Digilent Nexys2 FPGA board. We expect it to be a platform of painting on a plain space or painting an existing picture outline with color. There would be options for changing colors of the paint and once you are done you can reset the board to reload new picture. There will be options for 'fill' painting which would fill the blocks of the line drawing and also random scratches. Our initial plan is to use the BRAM to store a line picture to be painted but if time allows we plan to use a SD card to store the pictures.



## Components & I/O Processes

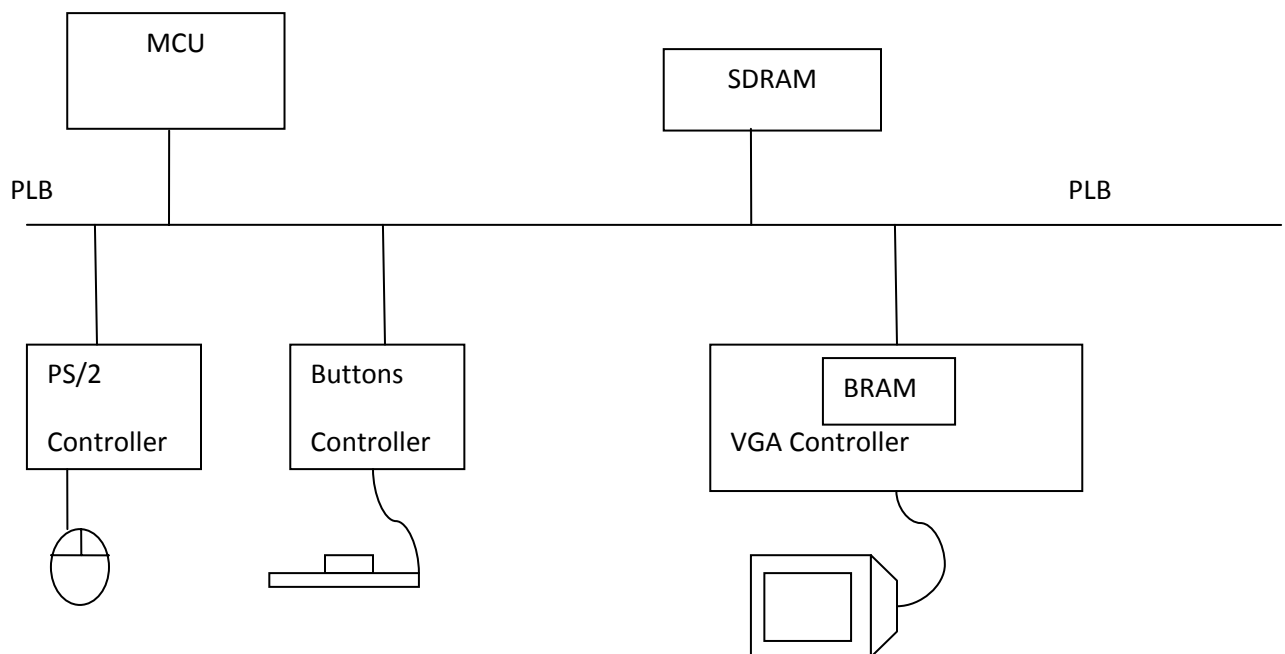


Figure 1: System Diagram

As illustrated in the diagram above, the following components have been identified for our project.

- FPGA board(MicroBlaze Spartan 3e)
- VGA display
- PS2 Mouse

If there is enough time, we will also implement the PModSD (SD card interface) component. The data exchange for the following I/O components is the following

### **VGA**

The software will handle the painting which will be saved on the BRAM where the VGA controller reads from and output it on the screen.

Resolution – Number of pixels that will be displayed on the screen will be 640\*480 with 8 bit depth.

Frequency – This displays how often the pixels will be drawn on the screen and it will be set to 60Hz.

### **PS/2 Mouse**

The PS/2 controller is interrupt based handling.

Coordinates – The coordinates for the mouse so that the screen can display the cursor

Button Clicks – Write pixel on screen with left click and change color with right click.

On-Board Push Buttons

Button clicks – Implement fill edges function on a button, for improvements we can add more functions like draw circle, rectangles, octagons, spray paint etc.

### **Debug**

RS232 UART – This will be used to make debugging our software easier.

### **SD-Card reader (improvement)**

If we got time over we will try to implement this module which allows us to load more and detailed pictures into screen. This requires the handling of the file system which will be a FAT file system implemented in C and a protocol for data exchange between the SD interface and the FPGA board.

## **Improvements**

Next to the basic system functionalities described above, some extensional works will be carried on if time allows.

### **Paint Software**

A more sophisticated paint logic and user friendly interface shall be programmed in the software, e.g. more painting modes, styles and tool boxes.

### **SD Card**

Initially we hard coded the picture that will be painted on and load it at the system start up. For further improvement we are going to store pictures in SD card. Thus there will be the SD card module and the driver, as well as a file system to be implemented.

## **Timeline**

The table describes the work distribution and timeline for tasks of the project:

<b>Task</b>	<b>Member</b>	<b>Week</b>	<b>Dependency</b>
1. VGA Controller	Dalong	1-3	None
2. Program Logic in C	Viktor	1-3	None
3. PS2 Mouse Controller	Kazi	1-3	None
4. Integrating T1, T2, T3 to have a basic structure	All	4-5	T1, T2, T3
5. Necessary changes to introduce the intended logic	All	4-5	T4
6. Testing and Debugging	Kazi	5-6	T5
7. Finalizing Project and report generation	All	5-6	T6
8. SD card Interface (Improvement)	All	5-7	T5