**DLD PROJECT PROPOSAL**

*[Shahjehan, Muhammad Shaheer, Syed Asghar Abbas Zaidi, Syed Muhammad Ather Hashmi]*

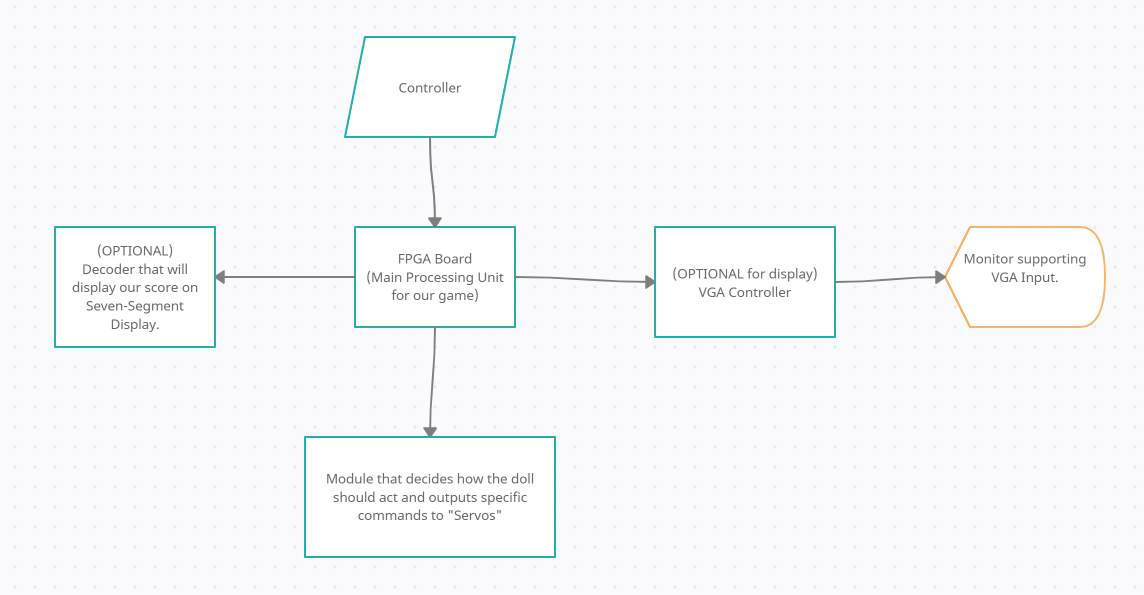
**Project Idea A: Main Idea**

Project X - The Cybernetic Playful Doll  
Or  
The Fanciful Ultra-reliable Machinating Obsequious Doll - F.U.M.O

**Project Description**

Our goal is making a cute character come to life and making your day better! We want to do it by allowing the user to interact with it to play games or possibly other activities if we have time.  
However, the main game that we are planning to possibly implement is a game of rock, paper and scissor with a random chance of having our “system” throw either a rock, paper or a scissor. Before it decides that, there will be a buffer time-period in which the user will give input with the use of an input peripheral (Gamepad Controller/PS4 Controller/Rudimentary Buttons).

The computer will compare both of these answers and give an output in the form of either displaying on a monitor using VGA or having a real-life doll act (optional).

****

**Prototype(s)**

This [video](https://www.youtube.com/watch?v=Wpd5-Yd4p3M) is our inspiration (Advice: turn down the audio abit).

Here you can see a doll dancing with a moveset on loop for a few seconds before transitioning to a different loop of dance animation. Thus, if we could manipulate the movement of the doll with coding differently on servo, we can have the doll showcase different outputs with its’ movements of its hands.

[*Animation Canvas Page.*](https://www.canva.com/design/DAFPHmqIevQ/xQKpDIdPI8iETQzhJQydMQ/edit?utm_content=DAFPHmqIevQ&utm_campaign=designshare&utm_medium=link2&utm_source=sharebutton)

We will be using Verilog Hardware Development Language (VHDL) alongside a Basys 3 FPGA board to implement this design.

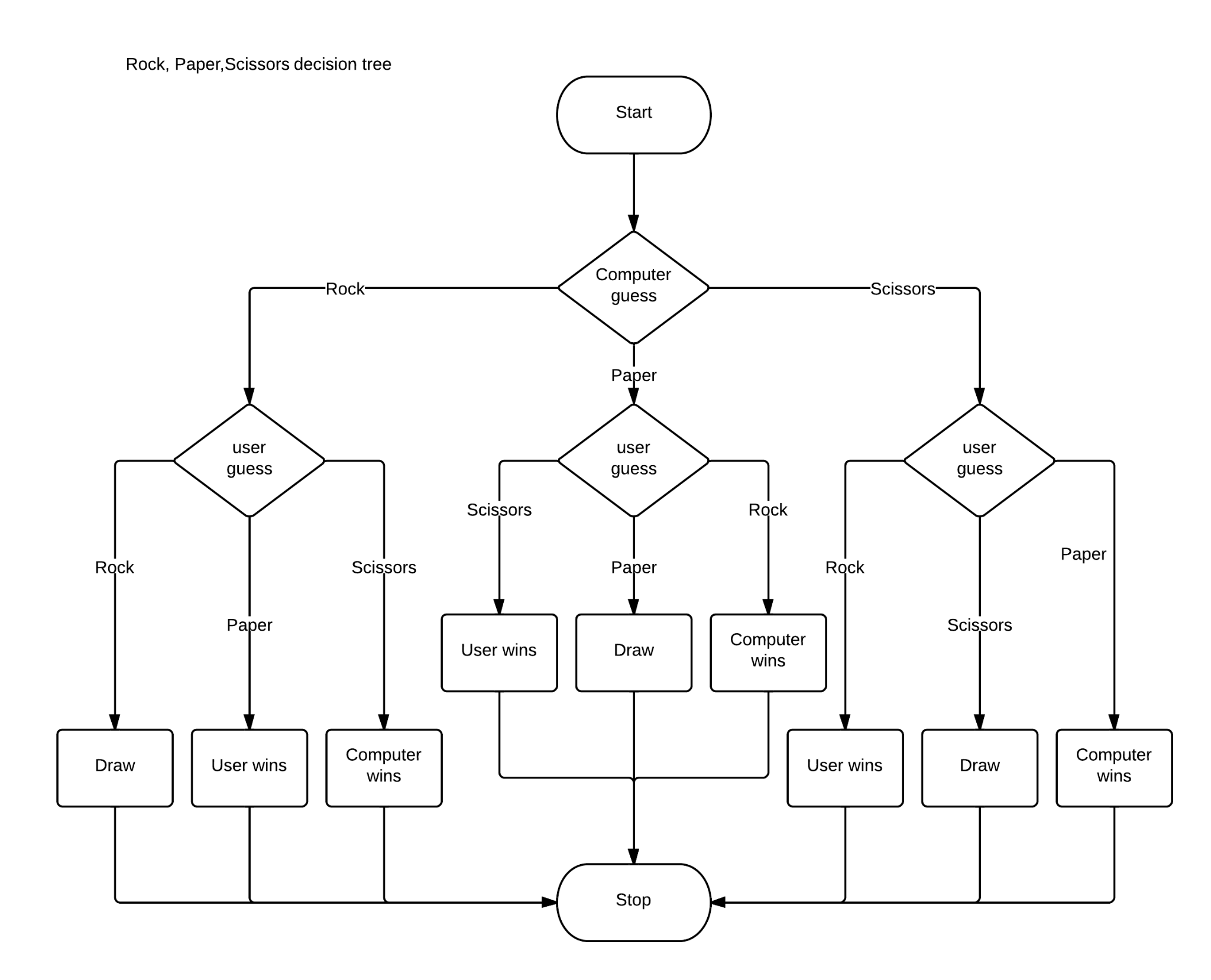
**Basic Sketch:  
\***(All sentences that start with an asterisk are meant to showcase that they are optional).

By default, the doll will stay still (Animation #4) and won’t do anything.

Stage 1: Pressing “Start” button on the Gamepad Controller, it will give input to the FPGA to jump start the entire system/game.  
 Doll will go from Animation #1 -> #2 -> #3-> #4 -> #5 -> #4 -> #5 (dancing) in the span of 3 seconds before going to either Animation #1 or 2 or 3 depending on the randomly-generated rock/paper/scissor by FPGA.

After Pressing “Start” and within those 3 seconds where Doll is dancing, user will press either 1 or 2 or 3 on Gamepad (they will represent rock, paper and scissor individually respectively for the FPGA).

Stage 2: 3.5 seconds after “Start has been pressed”, it will compare what it threw and what the player threw, then will credit the computer or player appropriately.



Accreditation:   
Scenario #1: Doll wins. Doll Dances (#1 -> #2 -> #3 -> #2 -> #1 -> #2 -> #3 -> #4) to celebrate its victory. \*(Animation is done faster to showcase excitement)

Scenario #2: Player wins: Doll Dances (#4 -> #5 -> #4 -> #5 -> #4 -> #5 -> #4). \*(Animation is done faster to showcase excitement)

Scenario #3: Tie (Or Player didn’t give any input): (#4 -> #2 -> #3 -> #2 -> #3) [feels like the doll is saying hi].

\*Animation can be canceled and doll will go to #4 upon pressing button#4, and the game will restart.

\*Scores can either be stored and displayed on Monitor or Seven-segment display.

Entire system can go to “idle state” when “start” is pressed again, \*Stored Scores will be resetted.

.

**EXTRAS:**

\*Sound System that will play appropriate sound effects.  
\*Storing Accreditation.  
\*Allowing more interactions between the user and the doll aside from just playing games with it.

**Tentative division of work:**

Input Peripheral - Muhammad Shaheer

Game’s logic design - Syed Asghar Abbas Zaidi

Implementing the randomness of the system - Syed Muhammad Ather Hashmi

Output Peripheral (Servos or VGA) - Shahjehan

There may be some instances where one of us gets stuck. In such a case, we would rely on peer reviews and peer programming to debug and fix the issue

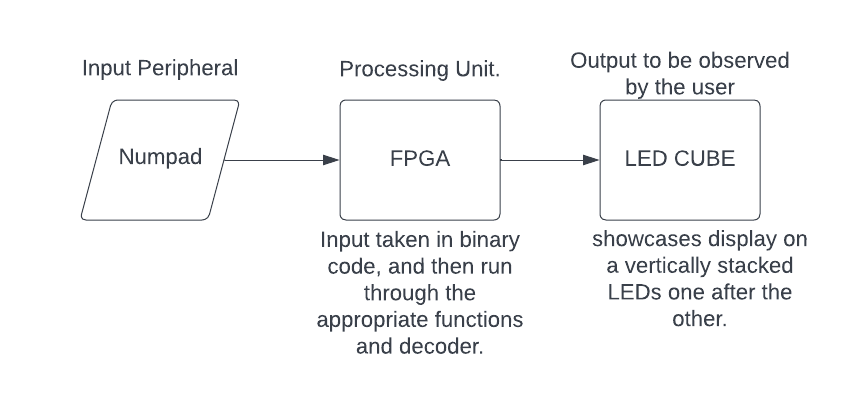
**References**

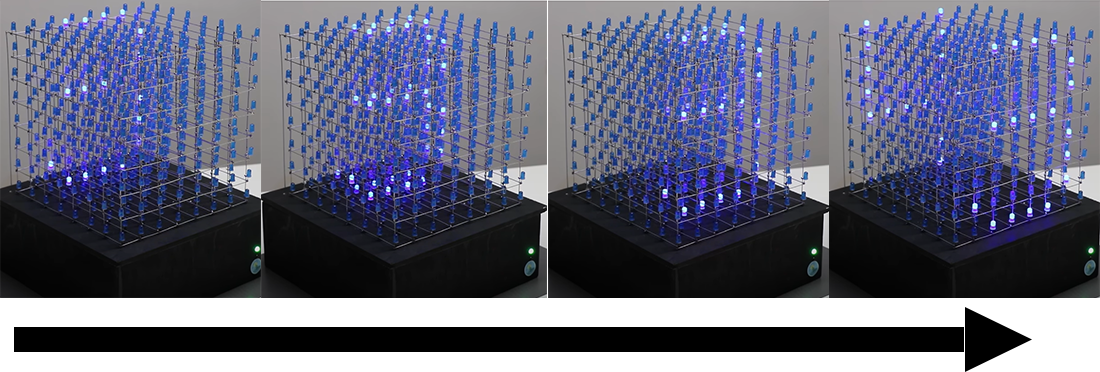
Main Inspiration: [*https://www.youtube.com/watch?v=Wpd5-Yd4p3M*](https://www.youtube.com/watch?v=Wpd5-Yd4p3M)

**Project Idea B: Side Idea**4x4x4 LED Cube

**Project Description**

Although our main goal remains to make F.U.M.O. exclusively and in the worst scenario, change its functionalities. But as a formality and as an absolute back-up plan, we plan to make a 4x4x4 LED Cube if nothing works out regarding F.U.M.O.  
In this [video](https://www.youtube.com/watch?v=T5Aq7cRc-mU), at 6:35, you can see an animation being played showcasing each number. We plan to do something similar where we will showcase the number being given as an input on LED CUBE as a cool animation. Our input peripheral will be “Numpad”.  
\*We also plan to showcase basic arithmetic and have the result of that display on an LED cube.



****

**Prototype(s)**Numpad will be taken as an input peripheral, FPGA will act as our main processing unit and display will be shown on 4x4x4 LED Cube.  
In a normal state, it will showcase the number pressed on the keypad onto the LED CUBE.  
The input will go in the form of binary into the FPGA, and utilizing what we learned in a seven-segment decoder, we will showcase our result in 2D stacked LEDs one by one. This will give the animation effect like we can see in the video attached above.

**EXTRAS:**

\*You will also be able to do basic calculations on it.  
\*To represent positive or negative, we can choose to light up specific 2D vertically stacked LEDs.  
\*To remove the constraint of our output only being between 0-9 for calculations or else “invalid” output will come, we can use base Base-19 number system that goes 0,1,2,3,4,5,6,7,8,9,A,B,C,D,E,F,G,H to showcase full range of arithmetic calculations.

**Tentative division of work:**

Input Peripheral - Muhammad Shaheer

Processing Unit’s functionalities - Syed Asghar Abbas Zaidi and Syed Muhammad Ather Hashmi

LED Cube logistics - Shahjehan

**References:  
Inspiration:**> <https://www.youtube.com/watch?v=hjrfa04KPYI>  
> <https://www.youtube.com/watch?v=T5Aq7cRc-mU> (main inspiration)