TAP MANIA

Be the fastest to tap all the right colours! Challenge with friends in your reaction time and hand-eye coordination skills.
Who will be the fastest tapper?

INFORMATION SYSTEMS TECHNOLOGY AND DESIGN

50.002 Computation Structures

Team 3-6 Yeh Swee Khim, Lin Cheng, Jia Yunze, Jisa Mariam Zachariah, Xie Han Keong

INSPIRATION

We were inspired by an old childhood game called Audition. In Audition, the objective is to press the arrow keys in a given order in order to make the player's virtual character do the best dance moves and earn the highest score. We felt that this idea was suitable within the limitations of this project.

On top of that, we thought that this game is skill-based and time-based, which can engage players more, encourage repeatability, and promote a spirit of competition between friends.

USER INTERFACE

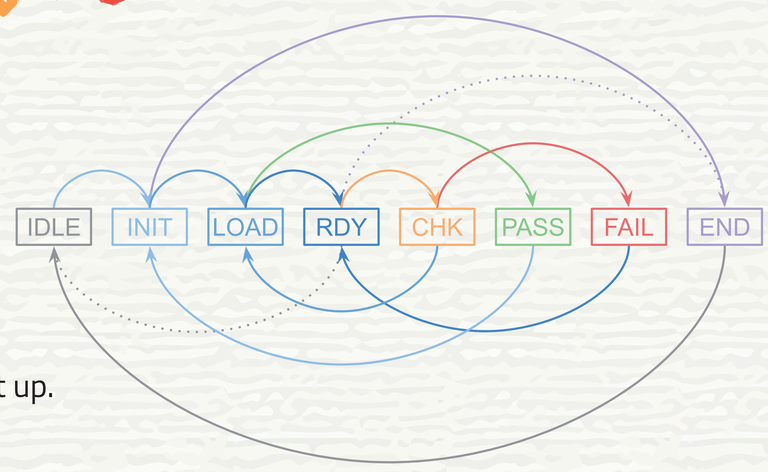
The top section contains three 7-segment displays to display the player's current time, current stage as well as the best time.

The middle section contains an array of eight LEDs (R/G/B/Y) for the game interface, surrounded by a region of LEDs (R/G) which will act as a backlight. All these LEDs will be covered by frosted acrylic (not shown in the picture) to diffuse the light so that it is less intense and spread out over a broader surface.

The lower section contains 6 buttons: the 2 buttons at the side are for starting the game and resetting the game, while the 4 buttons in the middle are for player input.

STATE TRANSITION DIAGRAM

Our game actually comprises a set of instructions that runs on a computer that we programmed onto the Mojo v3 FPGA. The instructions for the game can be generally classified into eight main internal states. The following diagram and table illustrates the sequence of events that will happen throughout the course of the game.



STATE	NEXT STATE	CONDITION
IDLE	INIT	Start button pressed
INIT	LOAD	Next stage is initialised
INIT	END	No more stages left
LOAD	RDY	Next color is loaded
LOAD	PASS	No more colors left
RDY	CHK	Color was pressed
RDY	IDLE	Reset button pressed
RDY	END	Timeout reached
CHK	LOAD	Color pressed was correct
CHK	FAIL	Color pressed was incorrect
PASS	INIT	Backlight flashed green
FAIL	RDY	Backlight flashed red
END	IDLE	Player time displayed

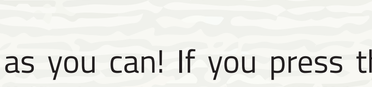
HOW TO PLAY

START

Press START to begin

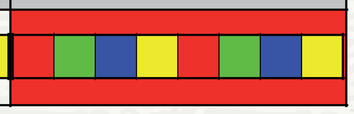
STEP 1

YOUR TIME will start counting. STAGE 1 has begun!
A sequence of lights on the game interface will light up.

Sequence	Game Interface
R, G, B, Y	
R, G, B, Y, G, B	
R, G, B, Y, R, G, B, Y	

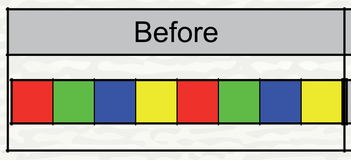
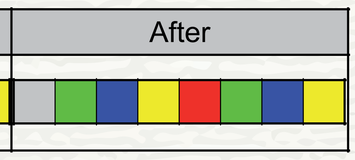
STEP 2

Tap the colors as fast as you can! If you press the wrong button, the backlight will flash red and there will be a delay.

Before	After
	

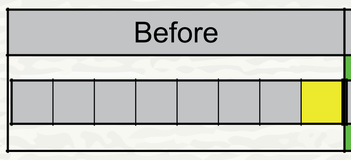
STEP 3

The display will tell you when you tap the right color!

Before	After
	

STEP 4

When you finish a sequence, the backlight will flash green.
You will move on to the next STAGE.

Before	After
	

FINISH

The last stage is 20. When completed, YOUR TIME will stop and blink. That is your final time.

If your time is better than BEST TIME, it will be the new BEST TIME and BEST TIME will blink as well. Congratulations, you have achieved the best time!

DATAPATH AND SCHEMATICS

To implement our game, we designed a von Neumann computer based on the Beta architecture we learnt in the 50.002 Computation Structures module. We decided to name it ***Gamma*** because ***G*** for ***Game***. The main differences are that data memory is not writable, PC is incremented by 1 only, and we introduced an I/O unit which will be connected to the inputs (i.e. buttons) and outputs (i.e. 7-segment displays, LED array).