Project Title: Маратона на Мишо

Overview

This Arduino project is a simple game that displays a running and jumping character on a 20x4 LCD screen. The player controls the character to avoid obstacles, and the game keeps track of the score. The game starts when the joystick is pushed, and the character can jump to avoid obstacles by pressing the button again.

Components:

* Arduino Board
* 20x4 I2C LCD Display
* Joystick
* Jumper Wires
* Breadboard

LCD Display Connection:

* VCC to 5V
* GND to GND
* SDA to A4
* SCL to A5

Push Button Connection:

* One terminal to digital pin 2
* The other terminal to GND
* Use a pull-up resistor configuration by connecting digital pin 2 to 5V through a 10kΩ resistor

Potentiometer Connection:

* Middle pin to LCD contrast pin (V0)
* One outer pin to GND
* The other outer pin to 5V
* Auto-start Pin Connection:
* Pin 1 to digital pin 10
* Pin 2 to GND

Code Explanation

* Libraries
* The LiquidCrystal\_I2C library is used to interface with the LCD display using the I2C protocol.
* Constants and Macros
* Defines various constants for the sprites and positions used in the game, such as terrain types, character positions, and LCD configuration.
* Global Variables
* Arrays to hold the terrain for the upper and lower parts of the screen and a boolean variable to track the button state.

Function Definitions

1. initializeGraphics()

Initializes the custom characters for the LCD display and sets the initial terrain to empty.

1. advanceTerrain()

Shifts the terrain to the left and introduces new terrain on the right.

1. drawBoy()

Handles the drawing of the boy character on the screen based on its current position. It updates the terrain arrays and checks for collisions. If a collision is detected, it returns true; otherwise, it returns false.

1. buttonPush()

Interrupt service routine to handle button presses. It sets the buttonPushed variable to true.

1. setup()

Sets up the initial configuration for the Arduino, including pin modes, initializing the LCD, and attaching the interrupt for the button.

1. loop()

The main game loop. It handles the game state, including starting the game, updating the terrain, handling jumps, and checking for collisions. It updates the display and manages the game timing.

How to Use

* Build the Circuit: Connect the components as described in the circuit diagram.
* Upload the Code: Upload the provided code to your Arduino board.
* Start the Game: Press the button to start the game. The character will start running.
* Control the Character: Press the button again to make the character jump and avoid obstacles.
* Game Over: The game ends when the character collides with an obstacle. The score will be displayed.

Notes

* The terrain is continuously generated and moves from right to left.
* The character can jump to avoid obstacles, and the jumping sequence includes several states for smooth animation.
* The score increases as the character continues to run without collisions.