Dodge!

Custom Project Report

Winter 2020

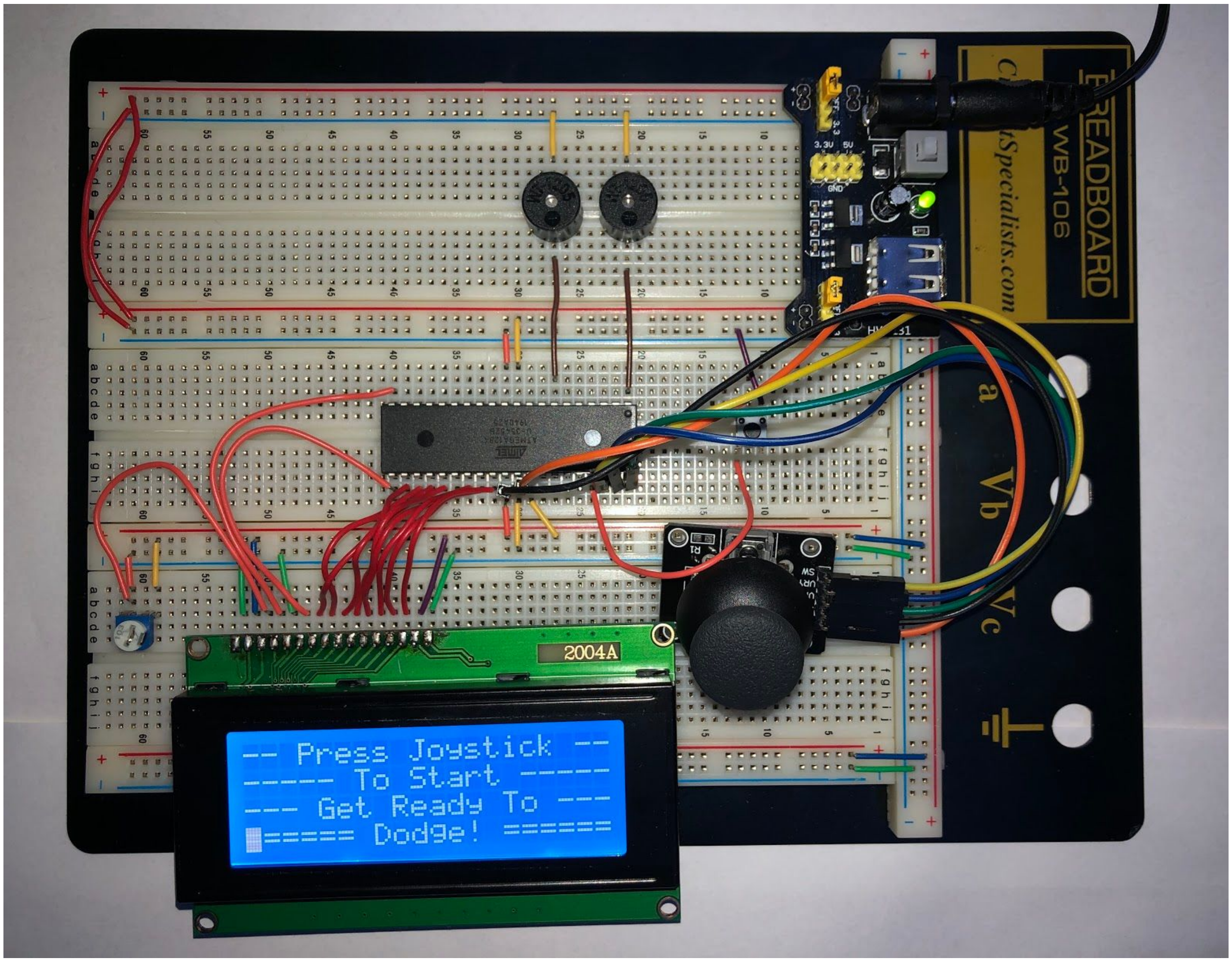
Rizelle Vinluan

Introduction

Dodge! is a simple platform game in which the player controls a character on the LCD screen with a joystick. The character must dodge obstacles in order to survive. If the character collides with an obstacle, then they die and are able to see their score. It is important to note that the character and obstacles are depicted as unique, custom characters.

The player starts the game by pushing the joystick. At any point, the player can press the button to reset the game, which also resets their current score.

There are also two speakers used for this game. One plays the background music and the other makes a sound whenever the player's character moves. Neither speaker cancels the other out.



User Guide

Rules

There is only one rule in this game: dodge the obstacles to survive as long as you can.

Controls

To move up, move the joystick upwards. To move down, move the joystick down. It is the same concept for moving left or right.

To start the game, the player must press the joystick. To reset the game, the player must press the button. The player can reset the game at any moment.

Components

Hardware

- ATmega 1284 Microcontroller
- Atmel ICE
- Breadboard WB-106
- HW-131 Power Supply
- ATmega ISP Adapter
- Potentiometer

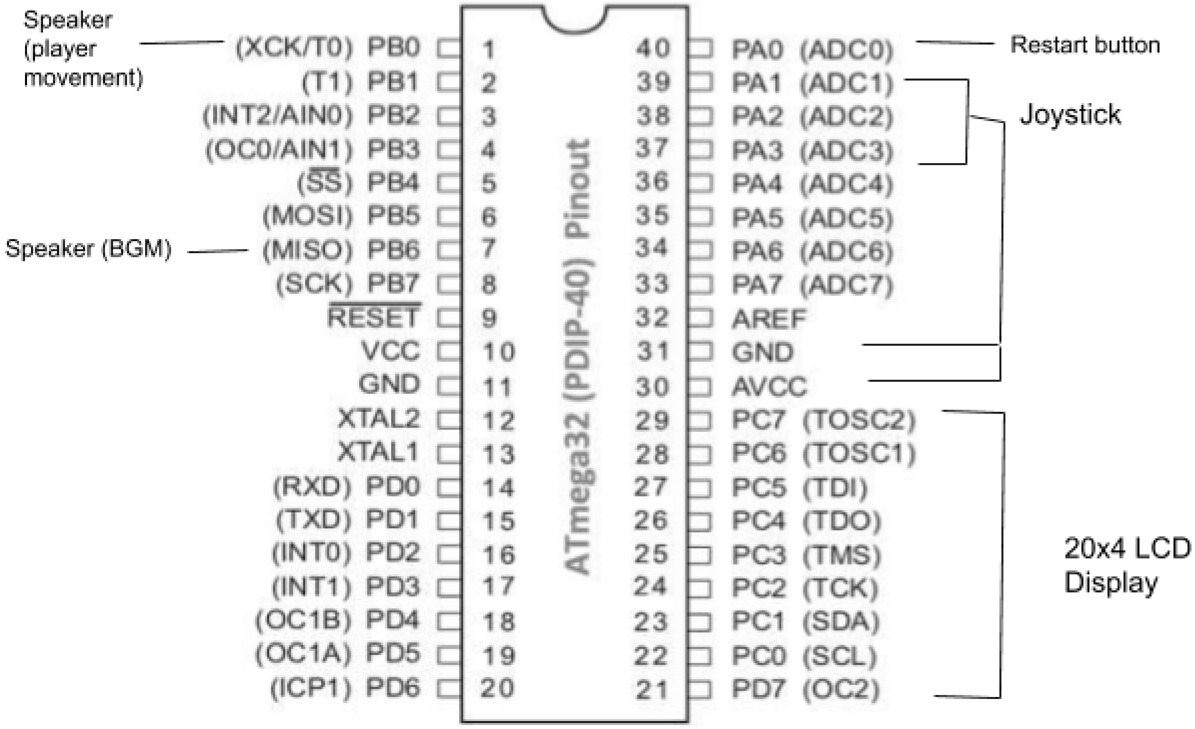
Inputs

- KY-023 Axis Joystick
- Push Button

Outputs

- 2 Speakers/Buzzer
    - One to play BGM
    - One to play when character moves
- 20x4 LCD IIC/I2C Display

Pin-out



Complexities/Build-upons

- Integrating and programming with a 20x4 IIC/I2C LCD Display
- Utilizing the joystick for player movement
- Having two speakers that don't cancel each other out when one plays
- Creating a custom character for the obstacles as well as the player

Demo Video Link

https://youtu.be/N2RRgQTSBNM

Source File Links

- main.c file needed for the game to work (https://github.com/rvinl001/CS120B-Custom-Project-Winter-2020/blob/master/source/main.c)
- io.h file, which is the header file for io.c (https://github.com/rvinl001/CS120B-Custom-Project-Winter-2020/blob/master/header/io.h)
- io.c file, which contains the functions needed to configure the LCD (https://github.com/rvinl001/CS120B-Custom-Project-Winter-2020/blob/master/header/io.c)
- joystick.h file, which is used to configure the joystick (https://github.com/rvinl001/CS120B-Custom-Project-Winter-2020/blob/master/header/joystick.h)
- pwn.h file, which is used to output noise from the speaker that makes the BGM (https://github.com/rvinl001/CS120B-Custom-Project-Winter-2020/blob/master/header/pwm.h)
- scheduler.h file, which is used to create tasks and implemented in the main function (https://github.com/rvinl001/CS120B-Custom-Project-Winter-2020/blob/master/header/scheduler.h)
- timer.h file, which is used to turn the timer on and off for the speaker as well as set the time for the states (https://github.com/rvinl001/CS120B-Custom-Project-Winter-2020/blob/master/header/timer.h)

Resume Description

Dodge! is a simple platform game that I developed using the material I learned in my embedded systems class. This game uses an analog-to-digital converter to interface with the LCD's light sensor and joystick. It also utilizes a pulse width modulator to create music for one of the speakers, and I had to mimic the PWM for the other one because the PWM only connects to a specific pin.