

CAB202 Assignment:

Student Name: Eunyoung(Jasmine) Hur

Student Number: n10622012

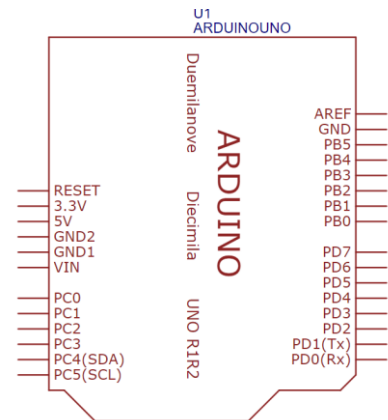
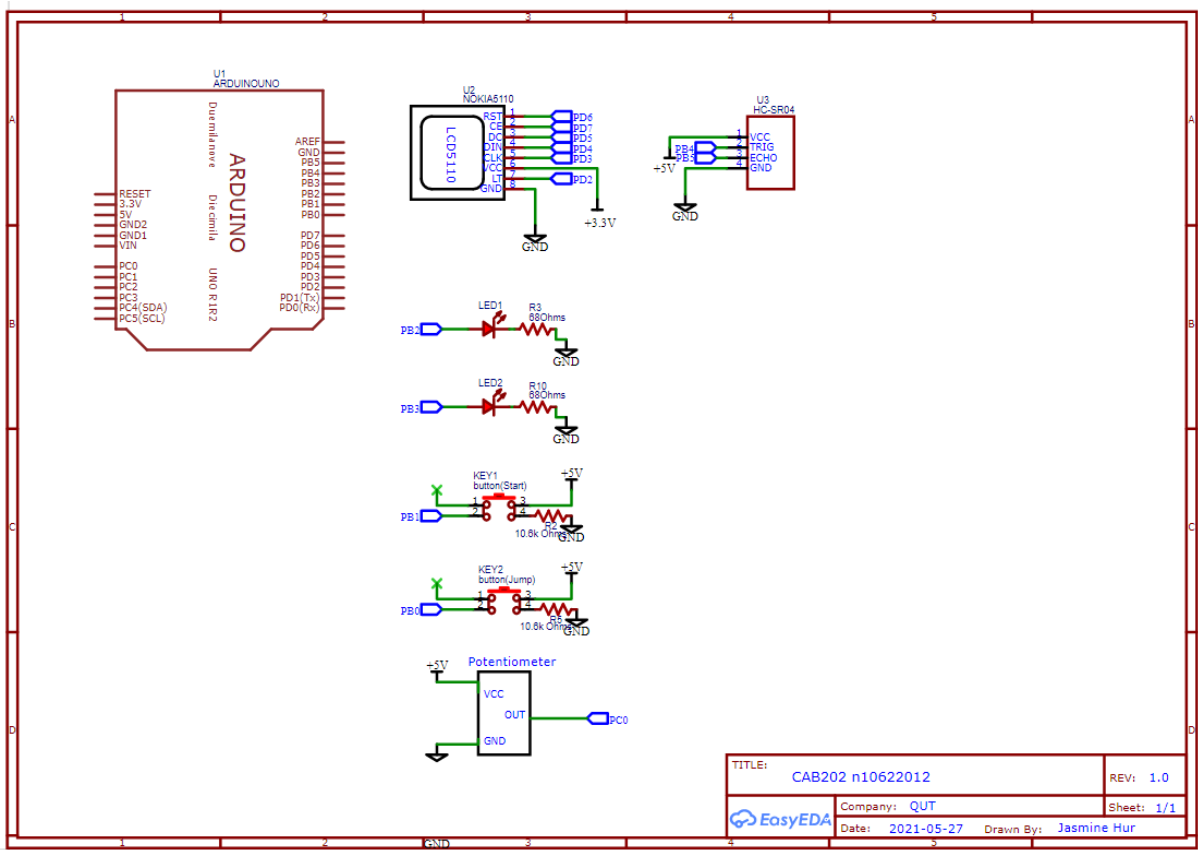
TinkerCAD Link (online students only): -

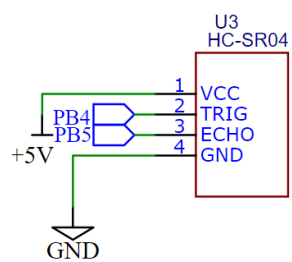
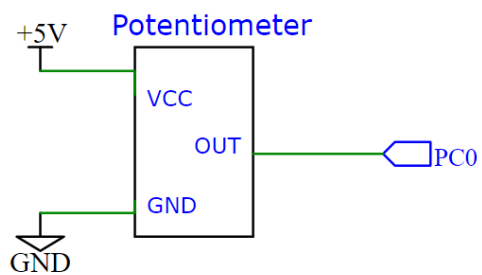
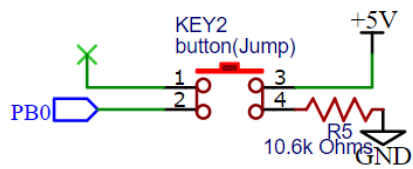
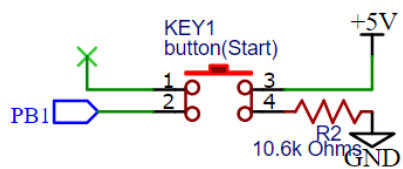
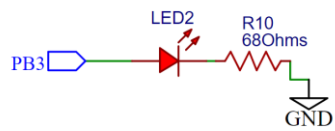
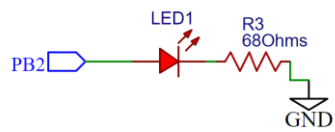
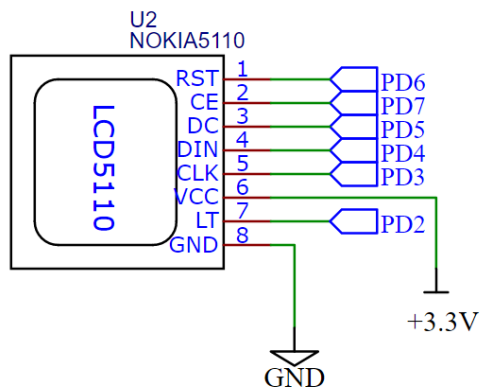
Introduction

This application is a simple implementation of the dinosaur game on Google. An user can use buttons to start the game and control the character. When a dinosaur hits an obstacle, the game ends and checks the total time spent on the game with a timer. Also, a distance sensor is used that can start the game only when I get close like a game machine.

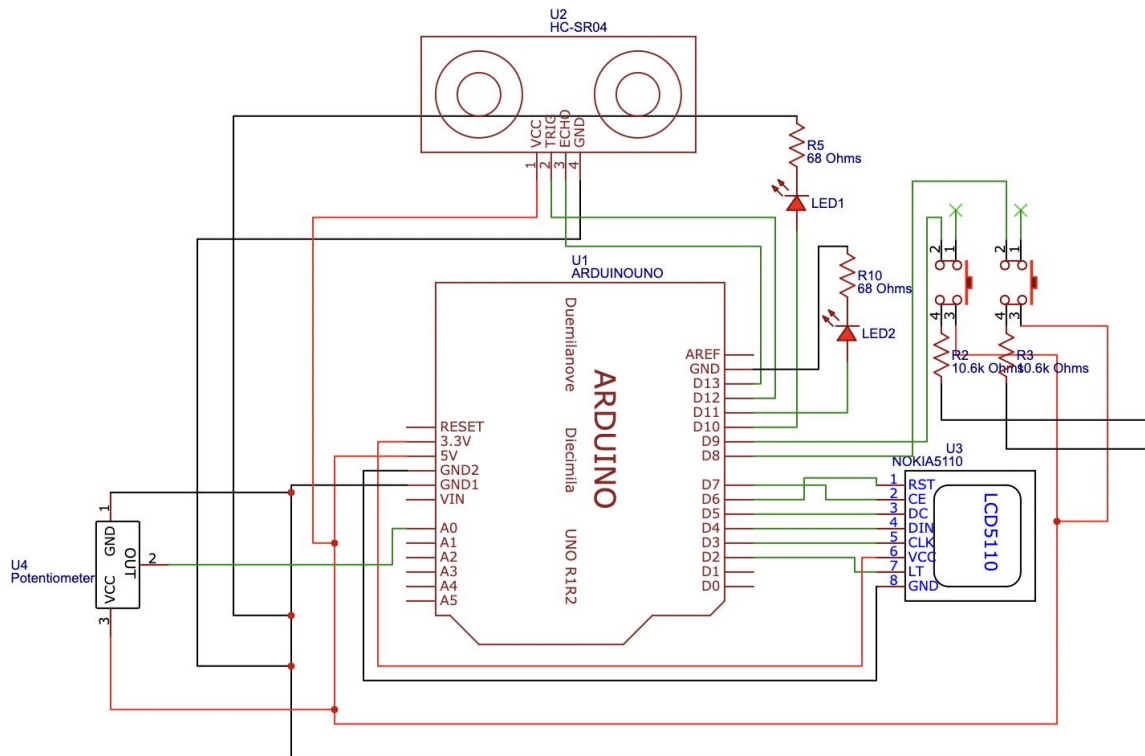
Digital I/O - Switch	In this project, two switches are used. One is for starting a game. The other switch is for jumping dinosaurs. Two switches are controlled with if condition and can be controlled LCD and LED.
Digital I/O – Interrupt-based Debouncing	Debouncing is controlled with programming which is a while loop. When the button is pressed, the light does not keep.
Digital I/O – LED	The LED is designed to light up when jumping. You can check if the user has pressed the jump button through the LED. This is controlled by a button. The other LED is designed to secure the user's sight. If the game was played in the dark, the button would not be visible, so it has been made such as a desk lamp.
Analog Input – ADC	A potentiometer used for adjusting the distance sensor. When you want to control the distance sensitivity through analog input, you can adjust it with the potentiometer without having to modify code every time.
Analog Output – PWM	A light and potentiometer are used for analog output. The light brightness can be controlled via potentiometer.
Serial I/O – UART	Serial communication is possible through Putty. When the user presses the jump button, the user can check when the character jumps.
LCD	The main components of this project. The LCD library has been used. The user plays and controls the game through LCD. The bit and execution are controlled through buttons and distance sensors.
Timers (other than debouncing or PWM)	It was used to check the total playtime of the game. You can check the time through LCD.

Schematic





Wiring Instructions



<Breadboard>

This is the initialization setup for the beginning connection circuit. When connecting every component, you should use a breadboard. On the breadboard, the LCD, 2 LEDs, 2 buttons, distance sensor(HC-SR04), and potentiometer will be placed. You should connect the - line which is on the breadboard to the ground on the Arduino and connect the + line which is on the breadboard to the 5V, then You can connect easier.

<Buttons>

Connect Button1: Only 3 of the 4 pins of each button are used. First, place button1 on the breadboard. Connect one pin on the - line on the breadboard with the resistance of 10.6k ohms. The other pin on the same side as the pin on the ground connects to the + on the breadboard. The pin will be connected to the Arduino (PB0) opposite the ground pin. The button connects with Arduino's PB0.

Connect Button2: This button connection is the same way as button1. but connect 1 which is opposite the side of the ground to the PB1.

<LED>

Connect LED1: place LED1 on the breadboard beside LED2. connect the short side pin of the LED to the ground (- line on the breadboard) and the rest to PB2 with a 68-ohm resistor together.

Connect LED2: place LED2 on the breadboard. One of the two LED's pins which is a short leg, connects to the ground with a 68-ohm resistor (- line on the breadboard) and the other to the PB3.

<LCD display - Nokia5110>

LCD uses a total of 8 pins. Place the LCD on the breadboard and connect the GND pin to the ground and (- line on the breadboard) the VCC pin to 3.3v on the Arduino directly.

And other pins such as LT, CLK, DIN, DC, CD, and RST need to follow this list for connection.

LT - PD2

CLK - PD3

DIN - PD4

DC - PD5

CD - PD7

RST - PD6

<Potentiometer>

Place a potentiometer on the breadboard. There are three pins in total. Connect to ground (- line on the breadboard), PC0, and 5V (+ line on the breadboard) in order.

<Distance sensor - HC-SR04>

Place a distance sensor on the breadboard. VCC and GND connect to 5V (+ line on the breadboard) and ground (- line on the breadboard) respectively of the 4 pins. TRIG connects to PB4 and ECHO connects to PB5.