به نام خدا

پروژه درس رباتیکز

استاد : محمد زارع

دانشجو : محمد حسن نیک صبا

**Arduino Controlled Pinball Machine یا**

**دستگاه کنترل شده پین بال آردوینو**



یک دستگاه پین ​​بال با اندازه استاندارد که با استفاده از قطعات خانگی و فروشگاهی و خانه های عرضه پین ​​بال ساخته شده است ، بازی توسط آردوینو کنترل می شود.

لیست مواردی که در این پروژه استفاده شده است

1 - Arduino Mega 2560

arduino mega 2560 یک برد میکرو کنترلر بر مبتنی بر ATmega2560 است. پایه های برد آردوینو Mega 2560 شامل 54 پین ورودی / خروجی دیجیتال (که 15 تا می تواند به عنوان خروجی PWM استفاده گردد)، 16 عدد پین ورودی آنالوگ، 4 پورت UART (پورت های سریال سخت افزاری)، یک نوسان ساز کریستال 16 مگا هرتز ، یک پورت USB، یک پاورجک، یک ICSP Header و یک عدد دکمه ریست می باشد.

Adafruit 2-pin LCD Display - 2

این محافظ Adafruit جدید استفاده از LCD کاراکتر 16x2 را آسان می کند. متأسفانه، این LCDها به تعداد زیادی پین دیجیتالی نیاز دارند، 6 عدد برای کنترل LCD و سپس 3 عدد دیگر برای کنترل نور پس زمینه RGB برای مجموع 9 پین. این نیمی از پین های موجود در یک آردوینو کلاسیک است.

24V Power Supply - 3

منبع تغذیه یا پاور ساپلای Power Supply سوئیچینگ گلدستار جهت تبدیل ولت و آمپر AC به DC سپس کاهش ولت و آمپر جهت رسیدن به دامنه مطلوب سیگنال. بطور مثال منبع تغذیه یا پاورساپلای سوئیچینگ یا SMPS ها برق AC را یکسو کرده و با استفاده از متد سوئیچ جریان بار (ON/OFF) به ولتاژ دلخواه دست می یابد.

Adafruit Force Senso 4 -

FSR ها حسگرهایی هستند که به شما امکان تشخیص فشار فیزیکی، فشردن و وزن را می دهند. استفاده از آنها ساده و کم هزینه است. این سنسور آلفا MF01A-N-221-A01 با ناحیه سنجش قطر 1/2 است.

Pop Bumper Assembly – 5

کیت مونتاژ ضربه گیر پاپ (سپر توربو) مورد استفاده در دستگاه پین ​​بال دیتا شرق و سایر موارد.

6 - Pop Bumper Switch

سوئیچ ضربه گیر پاپ بادوام برای ماشین های پین بال.

7 – Target Switch

برای ضربه زدن از آن استفاده میکنیم هدف ثابت، براکت نصب جلو، گرد سفید.

8 – Rollover Switch

در صورت مشکل وسیله به طور خودکار جریان پمپ را قطع می کند.

Flipper Assembly – 9

این مجموعه استاندارد باله یک جایگزین کامل برای مجموعه باله اصلی در P3 است.

LED Lamps - 10

خلاصه

ساخت یک ماشین پین بال با کنترل آردوینو به ترکیبی از مهارت های نجاری، مکانیکی، الکترونیکی و برنامه نویسی نیاز دارد.

این پروژه شرحی از نحوه ساخت این دستگاه ارائه می دهد. من سعی خواهم کرد به اندازه کافی مرجع ارائه کنم تا اگر تصمیم به ساختن خود دارید، زمان آن را راحت تر از من داشته باشید. در عین حال، نمی توانم به اندازه کافی بر نیاز به آزمایش همه چیز تاکید کنم. من از یک تخته چندلای یدکی برای نمونه اولیه هر جزء قبل از نصب در دستگاه واقعی استفاده کردم.

موارد اسافاده شده

کیس از تخته سه لا با روکش بلوط ساخته شده است که در فروشگاه خانگی خریداری شده است. تمام ابعاد در این سند بر حسب اینچ می باشد. پنل جلو و عقب 20H x 23W هستند. پانل های جانبی 20H x 47L هستند. قطعات با لبه های میتر بریده شده و با استفاده از وصال بیسکویت و چسب چوب به هم متصل شدند. پیچ های جیبی و چسب نیز می توانستند کار کنند. دستگاه مورد سوء استفاده زیادی قرار می گیرد، بنابراین باید قوی باشد. قبل از اینکه آن را به هم وصل کنم، یک دادو ¾ اینچی را نیز به اندازه 1 اینچ از پایین برش دادم. قفسه را چسب نزنید. برای انبساط و انقباض باید شناور شود. قفسه برای نگه داشتن منبع تغذیه اصلی و منبع تغذیه چراغ LED تزئینی نصب شده در زیر درب استفاده می شود. سطح بازی 22w x 42L است. توجه داشته باشید که به طور قابل توجهی کوتاهتر از کابینت است. این به من حدود 3 اینچ از جلو می دهد تا سیم ها را تا دکمه های بالن و به چراغ های LED منتقل کنم. همچنین فضایی برای مکانیزم پیستون توپ می دهد. سطح بازی به سادگی روی چند بلوک چوبی کوچک قرار دارد که به دو طرف حدود 4 اینچ زیر بالای کیس پیچ شده است. برای دستیابی به دستگاه، به سادگی درب آن را باز می کنم، زمین بازی را برمی دارم و آن را برمی گردانم. کیس و سطح بازی همگی در زوایای قائمه ساخته شده اند. زمین بازی همانطور که در این نمای جانبی نشان داده شده توسط پاها فراهم می شود. دستگاه های پین بال می توانند زاویه هایی از 1 تا 7 درجه داشته باشند. هر چه زاویه بالاتر باشد سرعت بازی بیشتر می شود. از آنجایی که مطمئن نبودم به چه زاویه ای نیاز دارم، پاها را طوری طراحی کردم که قابل تنظیم باشند. خود پاها چوبی هستند و در اکثر فروشگاه های خانگی موجود است. سپس با استفاده از یک مارپیچ سوراخی به عمق 12 اینچ در انتهای هر پایه ایجاد کردم. در انتهای پایین از یک اپوکسی دو جزئی برای قرار دادن مهره سه‌راهی 3/8 اینچی استفاده کردم. مراقب باشید که هیچ چسبی روی نخ ها یا جایی که سوراخی که دریل کرده اید را مسدود نکند. سپس یک میله نخ 12 اینچی 3/8 اینچی را داخل ساق پا فرو کردم و یک مهره را روی میله رزوه کردم. سپس یک پای تراز را به انتها وصل کردم. صفحات نصب پایه میز به سادگی به قفسه پایینی پیچ می شوند و پایه ها به داخل پیچ می شوند. برای تنظیم ارتفاع دستگاه، به سادگی مهره را شل کنید و سپس میله رزوه دار را به داخل یا خارج کنید. سپس مهره را دوباره محکم کنید تا مطمئن شوید که میز محکم می ماند. قاب درب از چوب بلوط ساخته شده است و همان ابعاد بیرونی کابینت است. عرض بالا و کناره ها 1.5 اینچ و عرض پایین 5 اینچ است. عرض اضافی در پایین برای پنهان کردن شکاف بین زمین بازی و کابینت استفاده می شود. یک دادو از لبه داخلی قطعات قاب پایین کشیده شد تا پوشش پلکسی گلاس را نگه دارد. من به جای شیشه از پلکسی گلاس استفاده کردم زیرا سبک تر، ایمن تر و برش راحت تر است. درب با استفاده از پیچ های جیبی کنار هم قرار داده شد. من این کار را انجام دادم تا در صورتی که پلکسی گلاس نیاز به تعویض داشت، بتوانم آن را جدا کنم. من همچنین یک نوار از چراغ های LED رنگارنگ را برای اهداف تزئینی زیر درب نصب کردم. در نهایت، درب را با استفاده از یک لولای پیانو که هم در درب و هم در کیس فرو رفتم، روی قاب نصب کردم.

CODE

long OldScore = 0;

long Target = 1;

long Pop = 1;

long Roll = 10;

int Targets[8];

int Rolls[3];

int Pops[4];

int Milli = 10;

int Sum = 0;

int Flash = 100;

int Ball = 0;

int i=0;

int Shot = 0;

int Lost = 0;

int Pressure = 1024;

#include <SoftwareSerial.h>;

SoftwareSerial mySerial = SoftwareSerial(255, TxPin);

void setup() {

/\* Words without an s are the value achieved by interacting with a device.

\* Works with an s keep track of which individual ones were interacted with.

\* The latter is needed to determine when all have been hit and the value needs upgrading

\* and the lights need turning off.

\*/

pinMode(TxPin, OUTPUT);

digitalWrite(TxPin, HIGH);

mySerial.begin(9600);

mySerial.write(12); // Clear

mySerial.write(17); // Turn backlight on

//target inputs

pinMode(2,INPUT\_PULLUP);

pinMode(3,INPUT\_PULLUP);

pinMode(4,INPUT\_PULLUP);

pinMode(5,INPUT\_PULLUP);

pinMode(6,INPUT\_PULLUP);

pinMode(7,INPUT\_PULLUP);

pinMode(8,INPUT\_PULLUP);

pinMode(9,INPUT\_PULLUP);

//rollover inputs

pinMode(10,INPUT\_PULLUP);

pinMode(11,INPUT\_PULLUP);

pinMode(12,INPUT\_PULLUP);

//lower ball shot switch

pinMode(15,INPUT\_PULLUP);

//upper ball shot switch

pinMode(16,INPUT\_PULLUP);

//lcd output

pinMode(17,OUTPUT);

//target lights, respective

pinMode(32,OUTPUT);

pinMode(33,OUTPUT);

pinMode(34,OUTPUT);

pinMode(35,OUTPUT);

pinMode(36,OUTPUT);

pinMode(37,OUTPUT);

pinMode(38,OUTPUT);

pinMode(39,OUTPUT);

//rollover lights, respective

pinMode(40,OUTPUT);

pinMode(41,OUTPUT);

pinMode(42,OUTPUT);

//pop bumper lights

pinMode(50,OUTPUT);

pinMode(51,OUTPUT);

pinMode(52,OUTPUT);

pinMode(53,OUTPUT);

}

void loop() {

// put your main code here, to run repeatedly:

//If a pull-down resistor is used, the input pin will be LOW when the switch is open and HIGH when the switch is closed.

//check if a target was hit

//\*\*\*\*\*\* Targets \*\*\*\*\*

for (int i=0; i<8; i++){

if (digitalRead(i+2) == LOW){

//Target activated

Targets[i]=1;

Score = Score + Target;

//turn on Target light

digitalWrite(i+32,HIGH);

//delay so as not get multiple points for one hit

delay(Milli);

break;

}

}

Sum = 0;

for (int i=0; i<8; i++){

Sum = Sum + Targets[i];

}

if (Sum == 8){

//all Targets lit, so flash and then turn off.

for (int j=0; j<3; j++){

for (int i=0; i<8; i++){

digitalWrite(i+32, LOW);

}

delay(Flash);

for (int i=0; i<8; i++){

digitalWrite(i+32, HIGH);

}

delay(Flash);

}

for (int i=0; i<8; i++){

digitalWrite(i+32, LOW);

Targets[i]=0;

}

delay(Flash);

//Multiply target value by 10

Target = Target \* 5;

//goto Skip;

}

// \*\*\*\*\*\*\*\*\*\*\* Rollovers \*\*\*\*\*\*\*\*\*

for (int i=0; i<3; i++){

if (digitalRead(i+10) == LOW){

//rollover activated

Rolls[i]=1;

Score = Score + Roll;

//turn on rollover light

digitalWrite(i+40,HIGH);

//delay so as not get multiple points for one hit

delay(Milli);

break;

}

}

Sum = 0;

for (int i=0; i<3; i++){

Sum = Sum + Rolls[i];

}

if (Sum == 3){

//all rollovers lit, so flash and then turn off.

for (int j=0; j<3; j++){

for (int i=0; i<3; i++){

digitalWrite(i+40, LOW);

}

delay(Flash);

for (int i=0; i<3; i++){

digitalWrite(i+40, HIGH);

}

delay(Flash);

}

for (int i=0; i<3; i++){

digitalWrite(i+40, LOW);

Rolls[i]=0;

}

delay(Flash);

//Multiply score by 2

Score = Score \* 2;

Roll = Roll \* 10;

//goto Skip;

}

//\*\*\*\*\*\*\*\*\*\* Pop Bumpers \*\*\*\*\*\*\*\*\*\*

for (int i=0; i<4; i++){

if (analogRead(i) > 500){

//pop activated

Pops[i]=1;

Score = Score + Pop;

//turn on pop bumper light

digitalWrite(i+50,HIGH);

//delay so as not get multiple points for one hit

//mySerial.print(analogRead(i));

//mySerial.print(" ");

delay(Milli);

break;

}

}

Sum = 0;

for (int i=0; i<4; i++){

Sum = Sum + Pops[i];

}

if (Sum == 4){

//all pop bumpers lit, so flash and then turn off.

for (int j=0; j<3; j++){

for (int i=0; i<4; i++){

digitalWrite(i+50, LOW);

}

delay(Flash);

for (int i=0; i<4; i++){

digitalWrite(i+50, HIGH);

}

delay(Flash);

}

for (int i=0; i<4; i++){

digitalWrite(i+50, LOW);

Pops[i]=0;

}

delay(Flash);

//Multiply target value by 10

Pop = Pop \* 2;

//goto Skip;

}

Skip:

//Determine ball number

if (digitalRead(15) == LOW){

//ball hit lower alley switch

//if not already done so, increase Ball

if (Shot == 0){

//Set Lost = 0 since not on pressure pad

Lost = 0;

Pressure = analogRead(7) + 20;

//set OldScore so as to reprint ball value on LCD

OldScore =-1;

Ball = Ball + 1;

if (Ball == 6){

Ball = 1;

Score = 0;

Target = 1;

Roll = 1;

Pop = 1;

}

Shot = 1;

}

}

if (digitalRead(16) == LOW){

//ball hit lower alley switch

//if not already done so, increase Ball

if (Shot == 0){

//Set Lost = 0 since not on pressure pad

Lost = 0;

Pressure = analogRead(7) + 15;

//set OldScore so as to reprint ball value on LCD

OldScore =-1;

Ball = Ball + 1;

if (Ball == 6){

Ball = 1;

Score = 0;

Target = 1;

Roll = 1;

Pop = 1;

}

Shot = 1;

}

}

if (analogRead(7) > Pressure){

//ball on pressure pad

Shot = 0;

if (Lost == 0){

//mySerial.print(analogRead(7));

//Score = Score + 100;

Lost = 1;

if (Ball == 5){

//Game Over

//flash rollovers and then turn off.

for (int j=0; j<3; j++){

for (int i=0; i<3; i++){

digitalWrite(i+40, LOW);

}

delay(Flash);

for (int i=0; i<3; i++){

digitalWrite(i+40, HIGH);

}

delay(Flash);

}

for (int i=0; i<3; i++){

digitalWrite(i+40, LOW);

Rolls[i]=0;

}

// flash pop bumpers and then turn off

for (int j=0; j<3; j++){

for (int i=0; i<4; i++){

digitalWrite(i+50, LOW);

}

delay(Flash);

for (int i=0; i<4; i++){

digitalWrite(i+50, HIGH);

}

delay(Flash);

}

for (int i=0; i<4; i++){

digitalWrite(i+50, LOW);

Pops[i]=0;

}

//Flash Targets and then turn off.

for (int j=0; j<3; j++){

for (int i=0; i<8; i++){

digitalWrite(i+32, LOW);

}

delay(Flash);

for (int i=0; i<8; i++){

digitalWrite(i+32, HIGH);

}

delay(Flash);

}

for (int i=0; i<8; i++){

digitalWrite(i+32, LOW);

Targets[i]=0;

}

mySerial.write(12); // Clear

delay(5);

// Required delay

mySerial.print(Score); // First line

mySerial.write(13); // Form feed

mySerial.print("Game Over!!!"); // Second line

}

}

}

//print to LCD

if (Score != OldScore){

mySerial.write(12); // Clear

delay(5); // Required delay

//mySerial.print(analogRead(7));

mySerial.print(Score); // First line

mySerial.write(13); // Form feed

mySerial.print("Ball = "); // Second line

mySerial.print(Ball);

OldScore = Score;

}

}