نام گزارشکار درس ریز پردازنده جلسه دوم نام و نام خانوادگی:مریم گروهی

استاد: اقای دکتر عباسی

نام آزمایش : رقص نور

وسایل آزمایش: نرم افزار ار دی اونو

هدف آزمایش : کنترل ال ای دی ها

توضیحات کد

متغیر key1:

key1 digitalRead(A0)

این خط به اشتباه نوشته شده است. باید به صورت int key1 = digitalRead(A0); باشد. این متغیر برای خواندن وضعیت یک سنسور یا دکمه متصل به پین A0 استفاده می‌شود.

تابع setup:setup:

void setup() {

pinMode(A0, INPUT);

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

pinMode(i, OUTPUT);

}

}

pinMode(A0, INPUT);: این خط پین A0 را به عنوان ورودی تنظیم می‌کند که می‌تواند برای خواندن وضعیت دکمه یا سنسور استفاده شود.

حلقه for که از 0 تا 3 تکرار می‌کند، پین‌های 0 تا 3 را به عنوان خروجی تنظیم می‌کند. این پین‌ها معمولاً به LED‌ها متصل هستند.

تابع loop:loop:

void loop() {

int i, a;

int dancinglight[4][4] = {

{0, 0, 0, 1},

{0, 0, 1, 0},

{0, 1, 0, 0},

{1, 0, 0, 0}

};

این تابع به صورت مکرر اجرا می‌شود. متغیرهای i و a برای استفاده در حلقه‌ها تعریف شده‌اند.

dancinglight یک آرایه 2 بعدی است که وضعیت LED‌ها را برای یک الگوی رقص نشان می‌دهد. هر ردیف از آرایه مشخص کننده این است که کدام LED روشن و کدام خاموش هستند.

خاموش کردن همه LED‌ها:

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

digitalWrite(i, 0);

}

این حلقه، تمام LED‌ها را خاموش می‌کند.

شرط برای تحریک الگو:

if (key1 == 0) {

for (a = 0; a <= 3; a++) {

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

digitalWrite(i, dancinglight[a][i]);

}

delay(200);

}

}

اگر key1 برابر با 0 باشد (یعنی دکمه یا سنسور فعال باشد)، دو حلقه تو در تو اجرا می‌شوند.

در هر دورة خارجی for (که a را تغییر می‌دهد)، با استفاده از digitalWrite وضعیت LED‌ها طبق آرایه dancinglight تنظیم می‌شود. یعنی LED‌ها به ترتیب روشن و خاموش می‌شوند.

delay(200); بعد از هر تغییر حالت LED‌ها، برنامه 200 میلی‌ثانیه وقفه دارد که باعث می‌شود الگوها با فواصل معین نمایش داده شوند.

نکات و اصلاحات

خواندن متغیر key1: بهتر است key1 را در داخل loop و با استفاده از digitalRead(A0) به‌روز کنید.

حلقه داخلی: در حلقه داخلی for، اشتباه است که شرط a<=3 را به جای i<=3 نوشته‌اید. باید:

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

نام گزارشکار درس ریز پردازنده جلسه دوم نام و نام خانوادگی:مریم گروهی

استاد: اقای دکتر عباسی

Pulldownنام ازمایش :

وسایل آزمایش: نرم افزار ار دی اونو

براساس وضعیت یک دکمه ledهدف ازمایش :کنترل

توضیحات کد

تعریف متغیرها:

int buttonpin = 8; // پین دکمه

int ledPin = 2; // پین LED

int buttonState; // وضعیت دکمه

buttonpin: شماره پینی که دکمه به آن متصل است. در اینجا از پین 8 استفاده شده است.

ledPin: شماره پینی که LED به آن متصل است. در اینجا از پین 2 استفاده شده است.

buttonState: متغیری که وضعیت دکمه (فشرده شده یا غیر فشرده) را ذخیره می‌کند.

تابع setup:

void setup() {

// put your setup code here, to run once:

pinMode(ledPin, OUTPUT); // تنظیم پین LED به عنوان خروجی

pinMode(buttonpin, INPUT); // تنظیم پین دکمه به عنوان ورودی

}

در این تابع که فقط یک بار هنگام روشن شدن میکروکنترلر اجرا می‌شود:

pinMode(ledPin, OUTPUT);: پین LED را به عنوان خروجی تنظیم می‌کند، به این معنی که می‌توانید سیگنال‌هایی (خاموش یا روشن) به این پین ارسال کنید.

pinMode(buttonpin, INPUT);: پین دکمه را به عنوان ورودی تنظیم می‌کند، که به Arduino اجازه می‌دهد تا وضعیت آن را بخواند.

تابع loop:loop:

void loop() {

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

buttonState = digitalRead(buttonpin); // خواندن وضعیت دکمه

if (buttonState == LOW) {

digitalWrite(ledPin, HIGH); // اگر دکمه فشرده شده باشد، LED را روشن کن

} else {

digitalWrite(ledPin, LOW); // در غیر این صورت، LED را خاموش کن

}

}

این تابع به طور مکرر اجرا می‌شود و وضعیت دکمه را بررسی می‌کند.

buttonState = digitalRead(buttonpin);: با خواندن وضعیت پین دکمه، مشخص می‌کند که آیا دکمه فشرده شده است یا خیر.

اگر دکمه فشرده شده باشد (که معمولاً در این حالت، وضعیت LOW خوانده می‌شود)، با digitalWrite(ledPin, HIGH); LED روشن می‌شود.

در غیر این صورت (یعنی دکمه فشرده نشده باشد)، با digitalWrite(ledPin, LOW); LED خاموش می‌شود.

نتیجه‌گیری:

این کد به شما این امکان را می‌دهد که با فشردن دکمه‌ای که به پین 8 متصل شده است، LED متصل به پین 2 را روشن یا خاموش کنید. به عبارت دیگر، LED روشن می‌شود وقتی که دکمه فشرده شده و خاموش می‌شود وقتی که دکمه رها می‌شود

نام گزارشکار درس ریز پردازنده جلسه دوم نام و نام خانوادگی:مریم گروهی

استاد: اقای دکتر عباسی

pullupنام ازمایش :

وسایل آزمایش: نرم افزار ار دی اونو

بر اساس وضعیت دکمه led هدف ازمایش : کنترل

1. تعریف متغیرها

int buttonpin = 8; // شماره پین برای دکمه

int ledPin = 2; // شماره پین برای LED

int buttonState; // متغیر برای ذخیره وضعیت دکمه

int buttonpin = 8;: این خط مشخص می‌کند که دکمه به پین شماره ۸ آردوینو متصل شده است.

int ledPin = 2;: این خط مشخص می‌کند که LED به پین شماره ۲ آردوینو متصل شده است.

int buttonState;: این متغیر برای نگهداری وضعیت دکمه (فشرده یا غیر فشرده) استفاده می‌شود.

2. تابع setup

void setup() {

// put your setup code here, to run once:

pinMode(ledPin, OUTPUT); // تنظیم پین LED به حالت خروجی

pinMode(buttonpin, INPUT); // تنظیم پین دکمه به حالت ورودی

}

void setup(): این تابع یک بار در ابتدای اجرای برنامه برای پیکربندی اولیه اجرا می‌شود.

pinMode(ledPin, OUTPUT);: این خط پین مربوط به LED را به عنوان خروجی تنظیم می‌کند، به این معنی که آردوینو می‌تواند به این پین سیگنال بدهد تا LED روشن یا خاموش شود.

pinMode(buttonpin, INPUT);: این خط پین مربوط به دکمه را به عنوان ورودی تنظیم می‌کند، به این معنی که آردوینو می‌تواند وضعیت این پین را بخواند.

3. تابع loop

void loop() {

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

buttonState = digitalRead(buttonpin); // خواندن وضعیت دکمه

if (buttonState == LOW) { // اگر دکمه فشرده شده باشد

digitalWrite(ledPin, HIGH); // روشن کردن LED

} else { // در غیر این صورت

digitalWrite(ledPin, LOW); // خاموش کردن LED

}

}

void loop(): این تابع به‌طور مداوم در حال اجرا است و کد اصلی برنامه در آن قرار دارد.

buttonState = digitalRead(buttonpin);: این خط وضعیت پین دکمه (HIGH یا LOW) را خوانده و مقدار آن را در متغیر buttonState ذخیره می‌کند. اگر دکمه فشرده شود، buttonState برابر با LOW خواهد بود.

if (buttonState == LOW): این شرط بررسی می‌کند آیا دکمه فشرده شده است یا نه.

digitalWrite(ledPin, HIGH);: اگر دکمه فشرده شده باشد، LED روشن می‌شود.

else: اگر دکمه فشرده نشده باشد:

digitalWrite(ledPin, LOW);: LED خاموش می‌شود.

نتیجه‌گیری

کد به طور کلی، یک دکمه و یک LED را کنترل می‌کند. وقتی دکمه فشرده می‌شود (وضعیت LOW)، LED روشن می‌شود. و وقتی دکمه رها می‌شود (وضعیت HIGH)، LED خاموش می‌شود. این کد می‌تواند به عنوان یک پروژه ابتدایی برای یادگیری درباره ورودی‌ها و خروجی‌ها در آردوینو استفاده شود.