



Namatek

True Education

Arduino Projects

www.namatek.com

پروژه آردوینو

فهرست مطالب

۱. آردوینو چیست؟
۲. اجزای قطعات جانبی برد نانو
۳. شناخت قطعات روی برد نانو
۴. شناخت قطعات پشت برد نانو
۵. پین های برد آردوینو نانو
۶. ویژگی های برد آردوینو نانو
۷. اجرای چندین پروژه آردوینو
- ۱-۷. پروژه پخش موسیقی
- ۲-۷. پروژه راه اندازی ماژول RFID
- ۳-۷. پروژه ساخت چراغ راهنمایی رانندگی

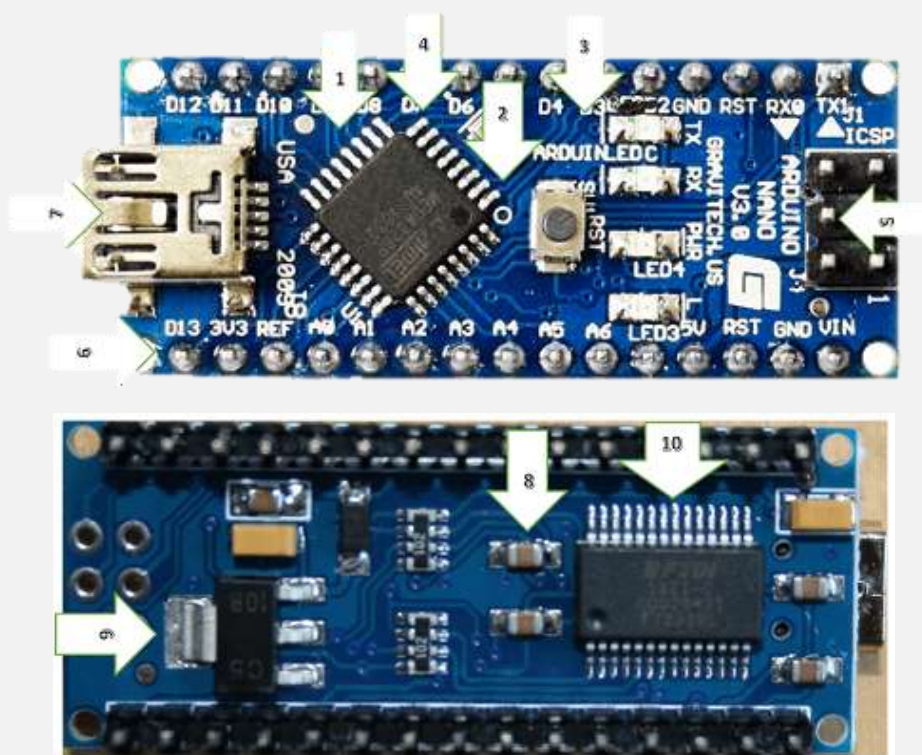
نکته اولیه ولی مهم در آشنایی با آردوینو این است که، آردوینو یک میکروکنترلر نیست و در واقع یک ماژول است که از هسته مرکزی میکروکنترلر AVR استفاده می‌کند و با قرار دادن سایر المان های مورد نیاز راه اندازی میکروکنترلر، کار با آن را بسیار راحت کرده است.

آردوینو چیست؟



آردوینوها در خانواده های مختلفی با امکانات متفاوت ارائه می‌شوند که ساده ترین مدل و کوچکترین آنها NANO است. آشنایی با آردوینو نانو برای گام اول در آموزش بهترین و سبک ترین گزینه است.

اجزای قطعات جانبی برد نانو



برای آشنایی با آردوینو به طور کامل لازم است تا قطعات تشکیل دهنده آن را بشناسیم.

شناخت قطعات روی برد نانو

۱. هسته مرکزی میکرو AVR ATmega328
۲. یک دکمه برای ریست کردن ماژول
۳. چهار عدد LED از نوع SMD برای کاربردهای مختلف (ارتباط سریال و پایه خروجی)
۴. یک کریستال اسیلاتور (کریستال) که برای تولید قلب تپنده میکروکنترلر یعنی "سیگنال پالس مربعی" استفاده می‌شود.

۵. تعدادی پین با پایه های بلند و دیپ از برد بیرون آمده اند که هر کدام یک کاربرد خاص دارند. به این قطعه که پایه های مدار SMD را به پایه های DIP تبدیل می کند، pin header می گویند.
۶. تعدادی پین با اسامی مشخص از طرف دیگر برد بیرون آمده اند که در واقع اکثر پایه های میکرو هستند که با track (سیم کشی) از کنار میکرو با پین های نزدیک به هم و کوچک، به طرفین برد رفته تا به راحتی قابل استفاده باشند.
۷. یک پورت USB mini برای اتصال برد به کامپیوتر

شناخت قطعات پشت برد نانو

۸. در سمت دیگر برد چندین خازن از نوع سرامیکی قرار دارد (با هدف بای پس کردن و حذف نویز)
۹. یک رگولاتور تبدیل ۵ ولت به ۳/۳ ولت قابل استفاده میکروهای AVR
۱۰. یک IC هم در پشت برد به منظور تبدیل USB به ارتباط سریال است قرار دارد.

پین های برد آردوینو نانو

- یکی از مباحث مهم در شناخت برد نانو، کاربرد هر کدام از پین های خارج شده از این برد است.

- ۱- تعدادی پین دیجیتال تحت عناوین D2-D13 که برای کاربردهای صفر و یکی استفاده می‌شوند. مثلاً برای روشن خاموش کردن LED از این پین می‌توان استفاده کرد.
- ۲- تعدادی پین آنالوگ با عناوین A0-A7 که در اصل برای استفاده از سیگنال‌های آنالوگ هستند و همان کاربرد تبدیل آنالوگ به دیجیتال را دارند. مثلاً برای کار با سنسور دمای LM35 نسبت به دما یک ولتاژ آنالوگ می‌سازد که با اتصال به این پین‌ها به دیجیتال تبدیل می‌شود.
- ۳- پین‌های 5v و 3v مانند یک منبع تغذیه DC هستند و ولتاژ برای کاربردهای مختلف تولید می‌کنند.
- ۴- پین GND که زمین مشترک مدار است.
- ۵- یک پایه VIN دارد که برای اتصال تغذیه خارجی استفاده می‌شود ولی از مزایای آردوینو این است که با کامپیوتر و با استفاده از پورت USB روشن می‌شود و نیازی به دستگاه منبع خارجی نیست.

ویژگی‌های برد آردوینو نانو

۱. کریستال موجود در این مدل 16 MHz است.
۲. پایه reset (RST)، همه میکروکنترلرها یک پین برای ریست شدن دارند که بدون نیاز به قطع و وصل شدن منبع تغذیه مدار خاموش و روشن شود. این پایه در اکثر میکروها با علامت (not) - مشخص شده‌اند که نشان دهنده active low بودن پایه است. یعنی این پایه برای اینکه فعال باشد برخلاف پیش فرض همه پین‌ها باید به صفر ولت وصل

شود. برای این پین معمولا یک مدار RC هم قرار می‌دهند که میکروی از قبل روشن شده، توسط خازن پین RST به زمین وصل و میکرو کاملا ریست شود. سپس با شارژ شدن خازن، ولتاژ ۵ ولت روی این پین افتاده و سیستم شروع به کار خواهد کرد.

۳. یکی از LED های موجود روی برد برای نمایش این موضوع است که آیا تغذیه به میکرو می‌رسد یا نه.

۴. دو LED دیگر متصل به پین های RX و TX ارتباط سریال است که برای ارتباط با کامپیوتر استفاده می‌گردند.

اجرای چندین پروژه آردوینو

پروژه پخش موسیقی

در این پروژه با یکی از قابلیت های جذاب آردوینو آشنا می‌شویم که برای هر پروژه مدنظر لاقلا یک کد در اینترنت وجود دارد. مرجع کد پخش موسیقی با آردوینو، برای هر موسیقی دلخواه شما موجود است.



تحلیل کد پروژه آردوینو برای پخش موسیقی

برای پخش موسیقی با آردوینو به کد ساده ای نیازمندیم که در ادامه به جزئیات آن می‌پردازیم.

- در خط اول کد متغیر buzzer را از نوع const تعریف شده و مقدار ۱۰ داخل آن ریخته می‌شود. این متغیر پایه ای از برد است که اسپیکر به آن متصل شده است.
- در ادامه متغیر ثابت songSpeed با مقدار ۱/۵ تعریف شده که برای ساختن دوره زمان پخش استفاده خواهد شد هر چه این ضریب بیشتر باشد موسیقی کندتر است و بر عکس.
- نت هایی که برای این موزیک استفاده شده است را define کرده و می‌بینیم که این موسیقی جالب با تعداد خیلی کمی از نت ها ساخته شده است.
- در بخش بعدی آرایه notes را تعریف کرده و داخل آن به ترتیب نت های استفاده شده را می‌نویسیم.
- در ادامه آرایه duration برای مدت زمان play شدن هر نت تعریف شده است که اعدادی مثل ۲۵۰ یا ۱۲۵ میلی ثانیه درون آن به همان ترتیب نت ها چیده شده است. (این دو بخش برای هر پخش موسیقی با آردوینو و نواختن هر موسیقی خاص یک کد مشخص دارند)
- در بخش setup بدنه اصلی کد را می‌نویسیم که یک بار فقط اجرا شود.
- یک حلقه for با شاخص i با ۲۰۳ حالت و گام ۱ نوشته شده است و

در ادامه tone را صدا زده روی پایه buzzer با فرکانس [i]notes و مدت زمان wait نت مورد نظر را پخش می‌کند. wait یک متغیر است که مقدار آن از ضرب [i]duration در عدد songSpeed حاصل شده است.

- در نهایت هم یک مقدار wait تاخیر داریم که برای این است که صدای نت ها داخل هم نروند.

تغییر پارامترهای پروژه آردوینو

اگر songSpeed را ۱ بگذاریم، سرعت پخش موزیک بالاتر می‌رود و موزیک روی دور تند پخش خواهد شد. اگر این مقدار را ۵ قرار دهیم موزیک خیلی کند پخش می‌شود. برای پخش آهنگ با آردوینو می‌توانید کد نت های مورد علاقه خودتان را از اینترنت پیدا کرده و آن را با همین کد پیاده کنید.

پروژه راه اندازی ماژول RFID

برای راه اندازی ماژول RFID در این پروژه لازم است، کارت RFID صحیح را بخوانیم و آن را به عنوان بازکننده درب ذخیره کنیم. در ابتدای پروژه بجای زبونه برقی در، از LED استفاده می‌کنیم که اگر کارت صحیح بود LED سبز روشن شده و اگر کارت اشتباه استفاده شده بود LED قرمز روشن شود.

کدنویسی پروژه RFID

2 پین از آردوینو مثل ۱۲ و ۱۱ را به عنوان خروجی با اسم greenLED و redLED به صورت const و یک بافر با ظرفیت ۱۳ برای خواندن کد کارت RFID تعریف می‌کنیم.

در ادامه باید این ID باید با مقدار مورد نظر ما مقایسه شود. برای این کار یک string با اسم karteMajid تعریف می‌کنیم و آن را مساوی "رمز" باز کننده درب قرار می‌دهیم.

بخش setup کد

داخل setup باید پین های خروجی را تعریف کرده و ارتباط سریال را راه اندازی کنیم. خانه آخر (۱۲ امی) از بافر را مساوی مقدار "n/" قرار می‌دهیم که انتهای بازه بافر را برای میکرو مشخص کرده باشیم.

بخش loop کد

در بخش loop این پروژه آردوینو، یک while با شرط Serial.available نوشته و با استفاده از دستور Serial.readString() متغیر Buffer را با اندازه ۱۲ بایت می‌خوانیم. با استفاده از این دستور تا زمانی که ۱۲ کاراکتر از ارتباط سریال وارد نشود کد در این خط متوقف شده و به بخش های بعدی نمی‌رود.

در ادامه یک if می‌نویسیم با این شرط که بافر خوانده شده از ارتباط سریال و متغیر karteMajid مساوی باشند و توجه داریم که باید متغیر بافر تعریف شده را با روش cast کردن از نوع char به string تغییر دهیم.

در داخل شرط if باید led سبز را روشن، قرمز را خاموش و برای else به صورت برعکس روشن و خاموش کنیم.

در ادامه برای طبیعی تر شدن، کد را به حالت چشمک زن led تغییر می‌دهیم. با آپلود کردن کد روی آردوینو و تست کردن ۲ کارت مختلف می‌بینیم که کد به درستی در حال اجراست.

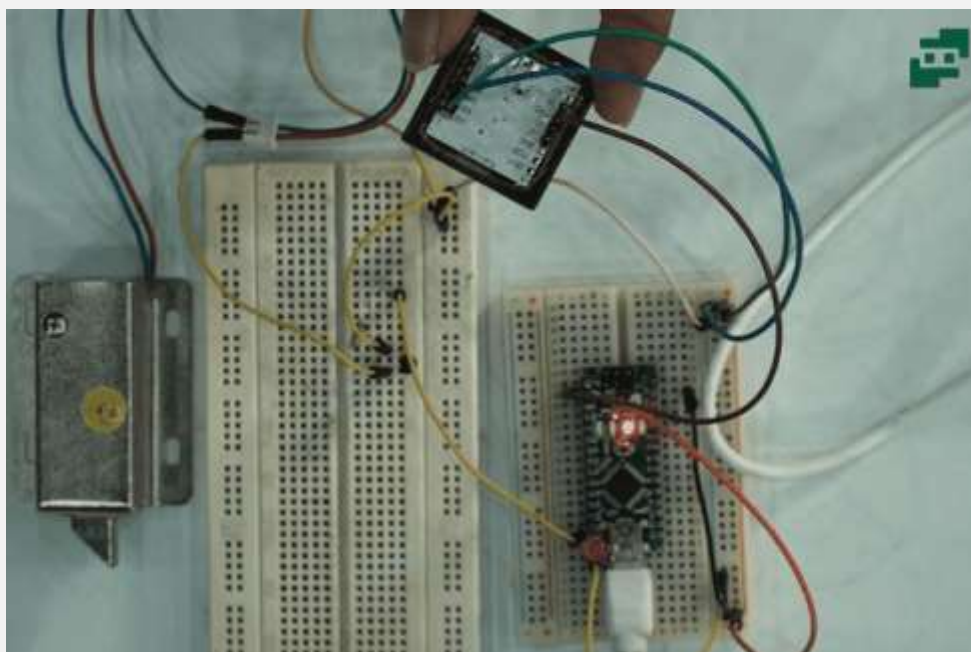
نکته: اگر همزمان ۲ کارت را با هم به ماژول RFID نزدیک کنیم مشکل تداخل (COLLISION) اتفاق می‌افتد، کارت‌های معمولی مثل کارت مترو با این مشکل کار نمی‌کنند.

تگ‌هایی وجود دارند که ANTI COLLISION هستند و می‌توانند چندین تگ را با هم بخوانند.

ناحیه کاری ماژول و فاصله تشخیص کارت توسط آن به توان ماژول بستگی دارد.

در فرکانس کاری ما (۱۲۵ کیوهرتز) که رنج low frequency به حساب می‌آید، RFID نهایتاً تا ۱۰ سانتی متر را پوشش می‌دهد.

مدار پروژه آردوینو



برای بستن مدار راه اندازی RFID و زبونه برقی، باید به پین ۱۲ که به led متصل است، یک ترانزیستور وصل کنیم و بیس را با مقاومت ۱ کیلو به همان پایه led متصل کنیم و سپس از کلکتور به پایه تحریک زبونه برقی و از پایه دیگر آن به منبع تغذیه ۱۰ ولت وصل می‌کنیم.

در نهایت با تست کردن ۲ کارت RFID می‌بینیم که زبونه درب با یک کارت باز شده و با دیگری باز نمی‌شود.

به راحتی با تعمیم و توسعه این پروژه آردوینو ساده، میتوان کارت های مختلفی را برای سیستم تعریف کرد.

پروژه ساخت چراغ راهنمایی رانندگی

وسایل مورد نیاز پروژه

۱. 3 عدد led به رنگ های سبز، زرد و قرمز

۲. یک سون سگمنت

۳. آردوینو نانو

منطق پروژه چراغ راهنمایی رانندگی با آردوینو

منطق کلی پروژه این است که ابتدا در محض روشن شدن led قرمز روشن است و شمارنده ۹ ثانیه می شمارد. سپس به محض ۰ شدن مقدار شمارنده، led سبز روشن شده و قرمز خاموش می شود. ۹ ثانیه شمارش می شود و وقتی به ۰ صفر رسید تایمر قطع شده و led زرد به تنهایی به مدت ۲ ثانیه روشن مانده و دوباره به قرمز روشن می رسیم.

کد نوشته شده پروژه آردوینو

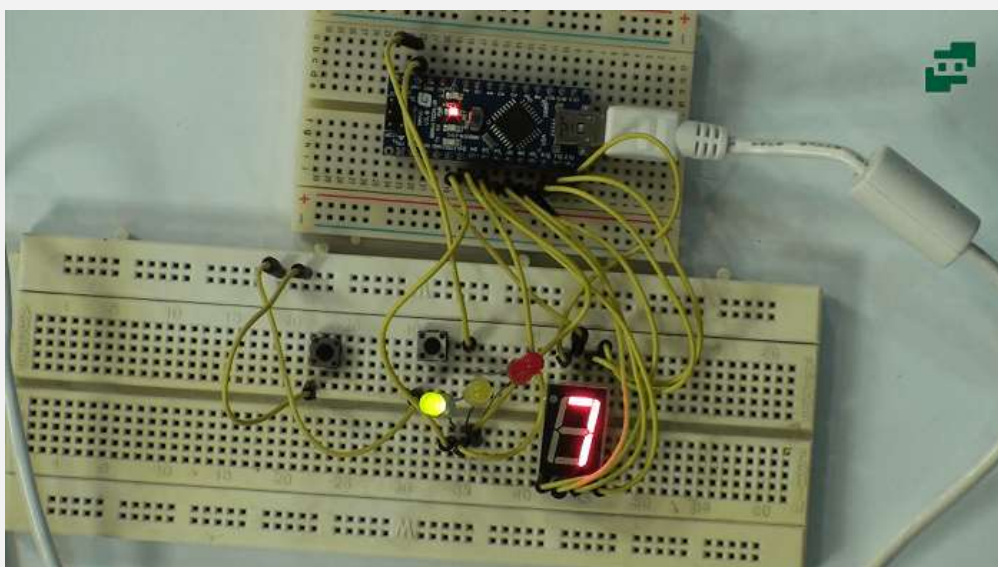
در گام نخست از این پروژه با آردوینو، باید به کد نوشته شده برای این پروژه مسلط شویم.

چون کد به صورت حلقه ای اجرا می شود آن را داخل loop می نویسیم. بعد از تعریف متغیر i ابتدا led قرمز را با دستور digitalWrite روشن می کنیم و شمارنده را با حلقه for معکوس و تاخیر ۱ ثانیه ای صدا می زنیم. بعد از اتمام دستورات for با دستور digitalWrite باید led قرمز را خاموش کرده و سبز را روشن کنیم.

بعد از آن دوباره دستور for معکوس را تکرار می‌کنیم تا شمارش چراغ سبز را داشته باشیم و در نهایت بعد از صفر شدن این مقدار تابع reset7segment را صدا زده و آن را خاموش می‌کنیم. بلافاصله led سبز رنگ خاموش و زرد روشن می‌شود و به مدت ۲ ثانیه تاخیر می‌دهیم.

می‌توان زمان این تاخیر را با عنوان yellowDelay در بخش اول کد define کرد. در خط آخر حلقه، led زرد خاموش می‌شود و با تکرار مجدد loop، led قرمز روشن خواهد شد.

بستن مدار پروژه



پایه های سون سگمنت را به پین های ۶ تا ۱۲ را به ترتیب متصل کرده و led ها به پین های ۳-۴-۵ میکرو متصل شده اند.

اجرای پروژه

بخش جالب و نهایی از این پروژه آردوینو اجرا کردن آن و تماشای حاصل دست رنج خودتان است. در مدار بسته شده کد را آپلود می‌کنیم و با اجرای پروژه می‌بینیم که به راحتی کار می‌کند.

توسعه پروژه ساخت چراغ راهنمایی رانندگی

می‌توان از این روش برای چراغ راهنمایی واقعی هم استفاده کرد، فقط کافی است از یک رقم دوم برای شمارنده استفاده کنیم تا اعداد متنوع و بیشتری برای شمارش داشته باشیم.

از طرفی مقدار این که قرار است چند ثانیه شمارش داشته باشیم می‌تواند define شود و از یا از طریق پورت سریال مانند پروژه های قبلی گرفته شود و از سمت یک اپراتور با تغییر وضعیت ترافیک، عوض شود. یا ممکن است به صورت برنامه ریزی شده از قبل، در ساعاتی از روز مقدار معینی زمان برای هر چراغ داشته باشد.

برای یادگیری و راه اندازی پروژه آردوینو بیشتر آموزش جامع الکترونیک نماتک را تهیه کنید.