

**به نام خدا**

**عنوان:**

گزارش آزمایش اول - آشنایی با مبانی آزمایشگاه ریزپردازنده

**اعضای گروه:**

فرید خدادادی

عرفان صبحایی روی

**استاد آزمایشگاه:**

استاد محمد لالی



دانشگاه خوارزمی

## چکیده

این آزمایش در قالب چهار سوال طراحی شده و اهداف آن شامل آشنایی با مفاهیم پایه آزمایشگاه ریزپردازنده، آشنایی با پیش نیازها و برنامه نویسی آردوینو و AVR می باشد. در این آزمایش با استفاده از نرم افزارهای مربوطه، یک برنامه ی چشمک زن برای آردوینو نوشته شده. همچنین در بخش های دیگر آزمایش در مورد مدار reset در آردوینو تحقیق به عمل آمده است و همچنین در مورد مقاومت pull-up توضیحاتی داده شده است. در بخش آخر آزمایش، اتصال دو حافظه ی EEPROM و انتقال اطلاعات بین آن دو برنامه نویسی و شبیه سازی شده است.

## مقدمه و معرفی

این آزمایش از چهار بخش مختلف تشکیل شده.

در بخش اول آزمایش هدف راه اندازی محیط برنامه نویسی آردوینو و نوشتن یک برنامه ساده چشمک زن است. بدین منظور از نرم افزار رسمی Arduino IDE استفاده شده است که خود شامل باقی ابزارهای لازم برای اتصال به آردوینو می باشد. کد برنامه ی مورد نظر در بخش مثال های نرم افزار مشاهده می شود.

در بخش دوم آزمایش هدف تحقیق و کسب اطلاعات در مورد مدار Reset است. با جستجو در سطح اینترنت، اطلاعات مربوط به این مدار جمع آوری شده و در بخش نتایج آزمایش آورده شده است.

در بخش سوم هدف شناخت و آشنایی با مقاومت Pull-up و پیدا کردن مقدار آن در مدار است. در این بخش نیز با جستجو در اینترنت اطلاعات مناسبی کسب شده است و در بخش نتایج آزمایش قابل مشاهده است.

در بخش چهارم و نهایی آزمایش هدف آشنایی با EEPROM و نحوه ی برنامه نویسی انتقال اطلاعات بین آنها می باشد. در این بخش با استفاده از AVR و کد C++ مربوط به انتقال اطلاعات در سیستم مذکور در صورت سوال، برنامه ی مورد نظر جهت انتقال یک رشته ی ۲۰۰ کاراکتری به EEPROM دریافت کننده و سپس نمایش آن در یک LCD برنامه نویسی شده است. همچنین با استفاده از نرم افزار Proteus شبیه سازی انجام شده و در قالب یک فایل تصویری در کنار فایل برنامه ها قرار گرفته است.

## تجهیزات مورد استفاده

برای آزمایشات از ابزارهای زیر استفاده شده است:

- نرم افزار Arduino IDE
- نرم افزار Proteus

- مرورگر Google Chrome
- یک IDE برای برنامه نویسی C++

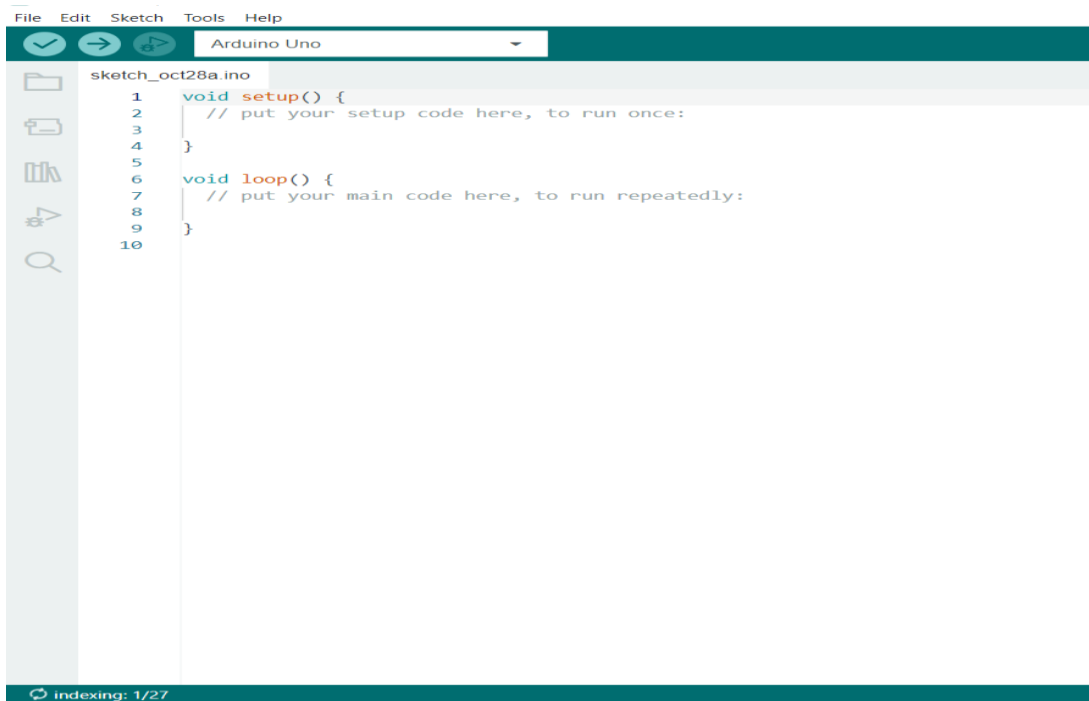
## روش آزمایش و نتایج

### سوال ۱

در این بخش نیاز بود تا ضمن آشنایی اولیه برنامه ای ساده بر برد آردوینو اجرا کنیم. در ابتدا نیاز به توضیح است که آردوینو پلتفرمی متن باز است که با استفاده از ارائه مدار های الکتریکی برنامه پذیر یا میکروکنترلر می توان کدهایی برای اجرا بر سخت افزار آن اجرا کرد. برای شروع کار در ابتدا نیاز بود تا dependency های مربوطه برای کدنویسی فراهم شوند. مرجع اصلی آردوینو به آدرس [arduino.cc](http://arduino.cc) ابزارهای مختلفی را برای نوشتن و اجرای برنامه بر روی بردها آماده کرده است. این سایت یک محیط یکپارچه برنامه نویسی (Integrated Development Environment) برای کدنویسی به نام Arduino IDE ارائه کرده است که پس از دانلود آن از وبسایت زیر می توان به کدنویسی پرداخت.

The screenshot shows the Arduino IDE 2.0.1 download page. The header is teal with navigation links: HARDWARE, SOFTWARE, CLOUD, DOCUMENTATION, COMMUNITY, BLOG, and ABOUT. Below the header are buttons for 'CODE ONLINE' and 'GETTING STARTED'. The main content area is titled 'Downloads' and includes a section for 'Arduino IDE 2.0.1' with descriptive text and a 'DOWNLOAD OPTIONS' sidebar listing download links for Windows (Win 10 and newer, 64 bits), Linux (AppImage 64 bits (X86-64)), and macOS (10.14: 'Mojave' or newer, 64 bits). A 'Nightly Builds' section at the bottom offers a preview of the incoming release.

سپس با اجرای این برنامه و اتصال برد آردوینو به سیستم با قطعه کد زیر مواجه می شویم:



همانطور که مشخص است ساختار کلی کد با دو تابع `setup` و `loop` مشخص شده اند، کدهایی که صرفاً در شروع به اجرا نیاز به عمل کردن دارند در بخش `setup` و کدهایی که نیاز دارند که به طور مکرر تکرار شوند نیز درون تابع `loop` قرار می گیرند.

در ادامه کد مربوط به چشمک زن (blink) قابل مشاهده است:



در تابع `setup` ابتدا پین مربوطه به `LED_BUILTIN` را به عنوان خروجی اعلام می‌کنیم. سپس در تابع `loop` مقدار ولتاژ آن را `HIGH` (یک منطقی) قرار می‌دهیم و سپس به مدت ۱۰۰ میلی ثانیه تاخیر ایجاد می‌کنیم. پس از این مراحل مقدار ولتاژ را `LOW` (صفر منطقی) قرار می‌دهیم تا چراغ خاموش شود. لازم است پس از خاموش شدن چراغ نیز تاخیر داشته باشیم تا بلافاصله در اجرای بعدی تابع `loop` چراغ روشن نشود.

## سوال ۲

برخی مواقع پس از آنکه برنامه ای بر روی یک برد آپلود شد، نیاز است تا آثار آن برنامه در مدار پاک شود و مدار به حالت پیش فرض خود بازگردد. به این عملیات `reset` می‌گوییم. هنگامی که این ریست همگام و هماهنگ با کلاک `CPU` باشد این ریست یک ریست `synchronous` است و هنگامی که به محض فعالسازی مدار ریست بدون کوچک ترین درنگی مدار ریست شود به آن `asynchronous` ریست می‌گوییم. ریست های `async` پتانسیل به وجود آوردن خطاهایی را در مدار دارند.

نحوه عملکرد این مدار به این شکل است که مقاومتی به ولتاژ جمع کننده مشترک وصل است که از طرفی به سوییچی به زمین متصل است و همچنین خازنی موازی با سوییچ نیز در مدار وجود دارد که با استفاده از ذخیره سازی بار الکتریکی عملیات ریست را انجام می‌دهد. این مدار، مداری `active-low` است، به معنای آنکه هرگاه مقدار ولتاژ `low` به آن اعمال شود، مدار ریست می‌شود، این اتفاق به این صورت می‌افتد که با فعال شدن آن، سوییچ متصل می‌شود که باعث خالی شدن شارژ خازن می‌شود که به ریست شدن مدار منجر می‌شود. طبق منابع مختلف در سطح اینترنت، مقدار مقاومت مذکور در مقیاس کیلو اهم و مقیاس خازن مذکور نیز در مقیاس نانو فاراد می‌باشد.

## سوال ۳

مقاومت‌های `pull-up`، مقاومت هایی هستند که برای اطمینان حاصل کردن از اعمال مقدار یک منطقی به یک سیم هستند که در غیاب سیگنال ورودی استفاده می‌شوند. در مدارات منطقی مقادیر حاضر در سیم ها یا یک منطقی و یا صفر منطقی هستند، این مقدار در برخی مدارات پنج ولت و صفر ولت هستند. اما در عمل ممکن است تفاوت میانولتاژ مربوط به یک منطقی و صفر منطقی به مقداری رسد که با اطمینان نتوان تشخیص داد که واقعا چه مقداری بوده است، به این حالت، حالت امیدانس بالا گفته می‌شود که مقدار ولتاژ در حالت شناور بین معادل یک منطقی و صفر منطقی قرار می‌گیرد. این مقدار شناور می‌تواند عملکرد مدار را دچار اختلال بکند و حتی باعث شود که مدار در حالت آنستیل قرار گیرد، لذا بایستی راه حلی برای جلوگیری از به وجود آمدن این مقادیر اندیشیده شود. مقاومت `pull-up` دقیقاً برای رفع چنین مشکلی ابداع شده است که به این صورت عمل

می کند که همواره سعی می کند در صورت وجود عدم قطعیت در اختلاف پتانسیل در مدار، آن را بالا نگه دارد تا میکروپرسور با مشکلات بالقوه مقادیر شناور و امپدانس بالا روبرو نشود.

## سوال ۴

ریپازیتوری زیر شامل پاسخ و کد این بخش است:

[لینک کد](#)

## منابع

- [1] "electronicwings," [Online]. Available: <https://www.electronicwings.com/avr-atmega/atmega16-eprom>. [Accessed 1 5 2022].
- [2] "electronicwings," [Online]. Available: <https://www.electronicwings.com/avr-atmega/lcd16x2-interfacing-with-atmega16-32>. [Accessed 1 5 2022].