در قسمت معرفی محل برنامه نویسی ابزار های بسیار مهمی معرفی شدند. یکی از این ابزار ها سریال مانیتور (serial monitor) بود. به دلیل حجم مطالب و اهمیت بالا آنها این ابزرا در 3 بخش به صورت تفکیک شده مورد بررسی قرار خواهد گرفت. در این بخش می‌خواهیم به بررسی ابتدایی این ابزار قدرتمند بپردازیم و دستورات ابتدایی برای استفاده از این ابزار را فرا بگیریم.

# ارتباط سریال

ارتباط سریال یک روش برای فرستادن اطلاعات به صورت بایت های پشت سر هم می‌باشد. به بیان ساده تر اطلاعات به بسته های یک بایتی تقسیم می‌شوند و 8 بیت پشت سر هم فرستاده می‌شوند. جابه‌جایی اطلاعات در این روش توسط یک یا در شرایطی 2 سیم اتفاق می‌افتد (یک سیم برای فرستادن اطلاعات و یک سیم برای دریافت اطلاعات). به همین دلیل از این روش برای فرستادن اطلاعات در مفاصل بالا توسط سیم استفاده می‌کنند. در مقابل این روش، روش فرستادن اطلاعات به صورت موازی قرار دارد که به 8 سیم نیاز دارد تا بتواند هر بیت از یک بایت را از طریق یک سیم مجزا بفرستد. این روش سیم بیشتری نیاز دارد ولی از سرعت بیشتری نیز برخوردار می‌باشد.

در حالت کلی ارتباط سریال دارای 4 پرتکل اصلی برای انتقال اطلاعات می‌باشد.

**SPI**: این پروتکل دارای سرعت بالایی می‌باشد و در فواصل کم کاربرد دارد. به طور مثال برنامه ریزی یک میکروکنترلر توسط این پروتکل اتفاق می‌افتد. در بخش SPI درباره این پروتکل بحث شده است.

**UART و USART**: این 2 پروتکل مناسب برای سرعت های متوسط و فواصل بالا می‌باشند. پروتکلی که سریال مانیتور برای برقراری ارتباط بین آردینو و کامپیوتر استفاده می‌کند UART می‌باشد.

**I2C**: این پروتکل برای فواصل کم و سرعت کم ولی برای برقراری ارتباط بین چند دستگاه مناسب می‌باشد. در بخش I2C به صورت کامل درباره این پروتکل بحث شده است.

**USB**: این همان پروتکل جهانی انتقال اطلاعات است که برای فواصل کم و سرعت بالا مناسب است. کامپیوتر شما همواره از این پروتکل برای انتقال اطلاعات استفاده می‌کند. بعضی از میکروکنتلر ها نیز از این پروتکل پشتیبانی می‌کنند.

# پروتکل UART

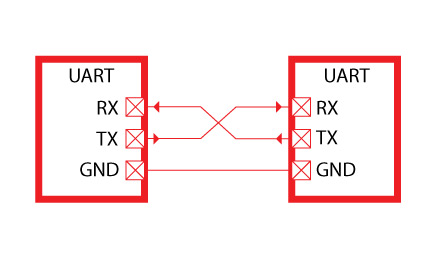
در بخش قبل گفته شد که در ارتباط سریالی داده ها به صورت بیت بیتی فرستاده می‌شوند. در واقع در بسته های 1 بایتی (8 بیتی) فرستاده می‌شوند. نحوه ارسال اطلاعات به این صورت می‌باشد. ابتدا فرستنده وضعیت سیگنال را از حالت 1 به حالت 0 تغییر می‌دهد. به بیان ساده تر از این طریق به گیرنده می‌فهماند که می‌خواهد داده بفرستد. به این بیت بیت شروع می‌گویند. سپس در 8 بیت بعدی دیتا را می‌فرستد و در نهایت دوباره سیگنال را به حالت 1 باز می‌گرداند تا گیرنده متوجه شود که بایت اول تمام شده است. به این بیت بیت پایان می‌گویند. قبل از فرستادن بیت پایان فرستنده می‌تواند یک بیت برابری نیز بفرستد. در داده های شما تعدادی 0 و 1 وجود دارد. اگر تعداد یک ها زوج باشد بیت برابری برابر 0 خواهد بود و اگر فرد باشد بیت برابری 1 خواهد شد. به وسیله این بیت شما می‌توانید وجود یا عدم وجود خطا در فرستادن کد را بررسی کنید.



حال مسئله‌ای که وجود دارد این است که گیرنده از کجا بداند که مدت زمانی که فرستادن یک بیت توسط فرستنده طول می‌کشد چقدر است؟ به بیان ساده تر اگر داده شما شامل 8 عدد 0 باشد شکل سیگنال شما مانند یک چاله خواهد بود. حال گیرنده از کجا می‌فهمد که در این سیگنال چند عدد بیت وجود دارد؟ یا به بیان ساده تر مدت زمان فرستادن بیت برای فرستنده چقدر بوده است؟ این موضوع در پروتکل UART به وسیله متغیری به نام baud rate یا همان bit rate بر طرف شده است. این متغیر که برای فرستنده و گیرنده مشخص می‌شود، سرعت تبادل داده‌ بین این دو را مشخص می‌کند. این متغیر مشخص می‌کند که بین این 2 دستگاه در ثانیه چند بیت تبادل می‌شود. به طور مثال اگر نرخ تبادل داده برابر 9600 باشد، یعنی در هر ثانیه 9600 بیت فرستاده می‌شود. در استاندارد مقادیر110، 300، 600، 1200، 2400، 4800، 9600، 14400، 19200، 38400، 57600، 115200، 128000 و 256000 تعریف شده‌اند. البته شما می‌توانید از مقادیر دلخواه نیز استفاده کنید که در این صورت باید برای هر دو دستگاه به یک اندازه تعریف شود.

# ارتباط سریالی در آردینو

آردینو نیز برای برقراری ارتباط بین برد و کامپیوتر از پروتکل UART استفاده می‌کند و برای نرخ انتقال داده از مقادیر استاندارد استفاده می‌کند. آردینو از پروتکل 2 سیمه استفاده می‌کند یعنی یک سیم برای فرستادن (Tx) اطلاعات و یک برای دریافت کردن (Rx) اطلاعات در نظر گرفته شده است. طبیعی است که سیم فرستادن (Tx) دستگاه اول و سیم دریافت (Rx) دستگاه دوم یکی هستند و سیم فرستادن دستگاه دوم و سیم دریافت دستگاه اول نیز یکی هستند.



# سریال مانیتور

در بخش معرفی فضای برنامه نویسی با ابزار سریال مانیتور آشنا شدید. شما به وسیله این ابزار می‌توانید تمامی اطلاعاتی که در ارتباط سریالی منتقل می‌شود را مشاهده کنید و در صورت نیاز اطلاعاتی نیز خودتان به برد بفرستید.

## نمایش اطلاعات در سریال مانیتور

در این بخش از سریال هدف ما نمایش داده هایی است که برد برای کامپیوتر از طریق ارتباط سرالی می‌فرستد. به طور مثال فرض بفرمایید دماسنجی تهیه کرده‌اید اما از سالم بودن آن مطمئن نیستید. یکی از راه های چک کردن از سالم بودن این قطعه این است که بعد نصب کردن قطعه به برد آردینو، به برد دستور دهید که دمای خوانده شده توسط این سنسور را برای کامپیوتر از طریق ارتباط سریالی بفرستد. حال شما می‌توانید به نگاه کردن به این اطلاعات از سالم بودن قطعه مطمئن شوید. در کل یکی از کاربرد های این ارتباط دیباگ کردن نرم افزار یا سخت افزار موجود در ستاپ می‌باشد. برای نوشتن این کد، ابتدا باید با توابع موارد زیر آشنا شویم.

## Serial.begin ( )

این تابع یک ارتباط سریالی با نرخ انتقال اطلاعات مشخص را با کامپیوتر شروع می‌کند. به بیان ساده تر شما با این دستور به برد خود می‌فهمانید که قرار است از طریق ارتباط سریالی‌ با نرخ انتقال مشخص به کامپیوتر وصل شود و اطلاعات مبادله کند.



توجه 1: حرف اول Serial بزرگ نوشته شده است.

توجه 2: اگر بعد از Serial عددی نوشته نشود به این معنی می‌باشد که ارتباط قرار است از طریق پورت صفرم برد صورت بپزیرد. این پورت از طریق سیم برد آردینو قابل دسترسی می‌باشد. در واقع این پورت برای برقرار کردن اطلاعات بین برد و کامپیوتر ایجاد شده است. هر برد به صورت پیش فرض دارای پورت های سریالی می‌باشد. به طور مثال برد Uno دارای یک پورت سریالی می‌باشد ولی برد mega دارای 4 پورت سریالی می‌باشد. همواره پورت صفرم برای برقراری ارتباط بین کامپیوتر و برد استفاده می‌شود و بقیه پورت ها برای برقراری ارتباط بین برد و ماژول های دیگر استفاده می‌شوند.



## Serial.print ( )

این تابع اطلاعات را از طریق سریال می‌فرستد و دستور خط جدید را نمی‌فرستد. به بیان ساده تر این تابع اطلاعات طوری می‌فرستد که در سریال مانیتور این اطلاعات در یک خط نوشته می‌شوند. مثلا اگر اعداد 1 تا 5 فرستاده شوند در سریال مانیتور به صورت 12345 نمایش داده خواهند شد.



توجه: این بار نیز اگر بعد از Serial عددی نوشته نشود منظور سریال صفرم یا همان ارتباط به کامپیوتر است و اگر نوشته شود منظور پورت مورد نظر می‌باشد.

## Serial.println ( )

این تابع اطلاعات را از طریق سریال به همراه دستور خط جدید می‌فرستد. به بیان ساده تر این تابع اطلاعات طوری می‌فرستد که در سریال مانیتور این اطلاعات در خطا های مجزا نوشته می‌شوند. مثلا اگر اعداد 1 تا 5 فرستاده شوند در سریال مانیتور هر عدد در یک خط نمایش داده خواهند شد.

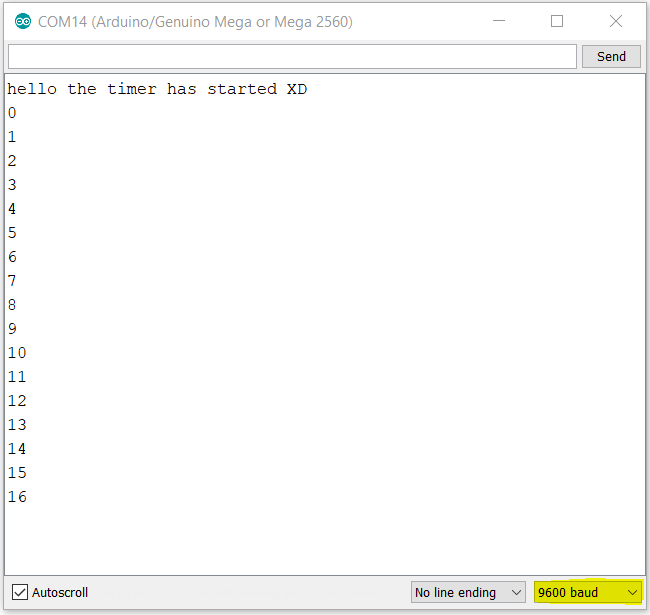


## کد نمایش اطلاعات در سریال مانیتور

در اینجا ما می‌خواهیم یک ثانیه شمار بسازیم. به بیان دیگر از لحظه شروع برنامه هر یک ثانیه مدت زمان در حال اجرا بودن برنامه را برای کامپیوتر بفرستد. ما این کار را به وسیله تابع delay و توابع مربوط به سریال انجام خواهیم داد. برای دیدن اطلاعات ارسالی از برد آردینو، باید به بخش سریال مانیتور که در بخش معرفی فضای برنامه نویسی معرفی شد بروید و نرخ تبادل داده را با آن مقداری که در کد هست یکی کنید تا بتوانید داده ها را ببینید.



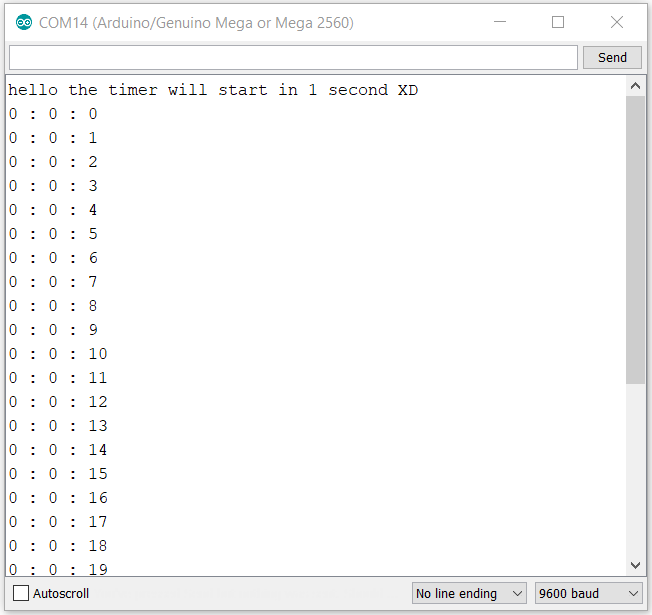
توجه: در پایین سمت چپ سریال مانیتور شما می‌توانید نرخ انتقال اطلاعات را مشخص کنید. دقت کنید که این نرخ باید با مقداری که در کد مشخص می‌شود یکی باشد.



## تایمر کامل

حال اگر بخواهیم دقیقه و ساعت را نیز نشان دهیم باید کمی کد را تغییر دهیم تا به شکل زیر در بیاید.





توجه: به تفاوت بین Serial.print و Serial.println در اینجا توجه کنید. به خوبی تفاوت این دو تابع در این مثال خاص نمایان می‌باشد.