به نام خدا

درس مدارهای واسط

استاد:

دکتر امین فصحتی

پروژه سه

نویسندگان:

یوسف سدیدی | ۴۰۱۱۷۰۵۹۷

مهرشاد دهقانی | ۴۰۱۱۰۵۹۱۲

امیرعلی شیخی | ۴۰۱۱۰۶۱۵۸

کیهان مسعودی | ۴۰۱۱۰۶۵۰۹

زمستان ۱۴۰۳

فهرست مطالب

مقایسه عملکرد چند واسط ارتباطی در سیستم‌های نهفته

مقدمه:

در این پروژه، قصد داریم عملکرد چندین واسط ارتباطی را برای انتقال داده‌ها بین دو برد آردوینو مقایسه کنیم. این واسط‌ها شامل I2C، SPI، UART، Wi-Fi، و CAN، One-Wire هستند. در ادامه، هر پروتکل به‌صورت جداگانه توضیح داده شده و نحوه‌ی اتصال دو برد آردوینو به یکدیگر شرح داده می‌شود. کدهای مربوط به تمام موارد به پیوست این فایل موجود است.

بررسی پروتکل‌ها:

۱. پروتکل I2C

معرفی:

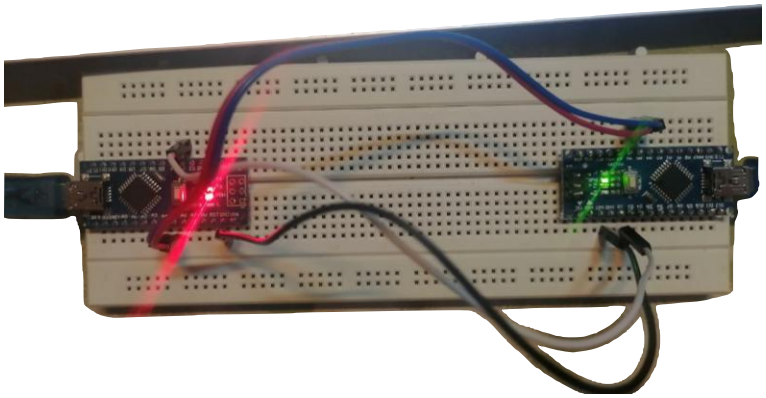
پروتکل I2C (Inter-Integrated Circuit) یک روش ارتباطی سریال دو سیمه است که برای انتقال داده بین میکروکنترلرها و قطعات جانبی مانند سنسورها، EEPROM و نمایشگرها مورد استفاده قرار می‌گیرد. این پروتکل دارای یک خط کلاک (SCL) و یک خط داده (SDA) است که برای همگام‌سازی انتقال داده به کار می‌روند. هر دستگاه متصل به باس I2C دارای یک آدرس منحصر به فرد است که امکان ارتباط چندین دستگاه Master-Slave را فراهم می‌کند. I2C دارای حالت‌های Master است که در آن، دستگاه کنترل ارتباط را در اختیار دارد. این پروتکل می‌تواند در نرخ‌های انتقال ۱۰۰ کیلوبیت، ۴۰۰ کیلوبیت، ۱ مگابیت و ۳.۴ مگابیت بر ثانیه عمل کند. سادگی سیم‌کشی، هزینه کم و مصرف توان پایین از مزایای این پروتکل هستند.

ویژگی‌ها:

- ارتباط چندگانه (Multi-Master, Multi-Slave)
- استفاده از فقط دو سیم
- قابلیت آدرس‌دهی ۱۲۷ دستگاه روی یک باس
- پشتیبانی از سرعت‌های ۱۰۰ کیلوبیت، ۴۰۰ کیلوبیت و ۳.۴ مگابیت بر ثانیه

نحوه‌ی اتصال دو برد آردوینو:

- SDA (A۴) به SDA
- SCL (A۵) به SCL
- GND به GND



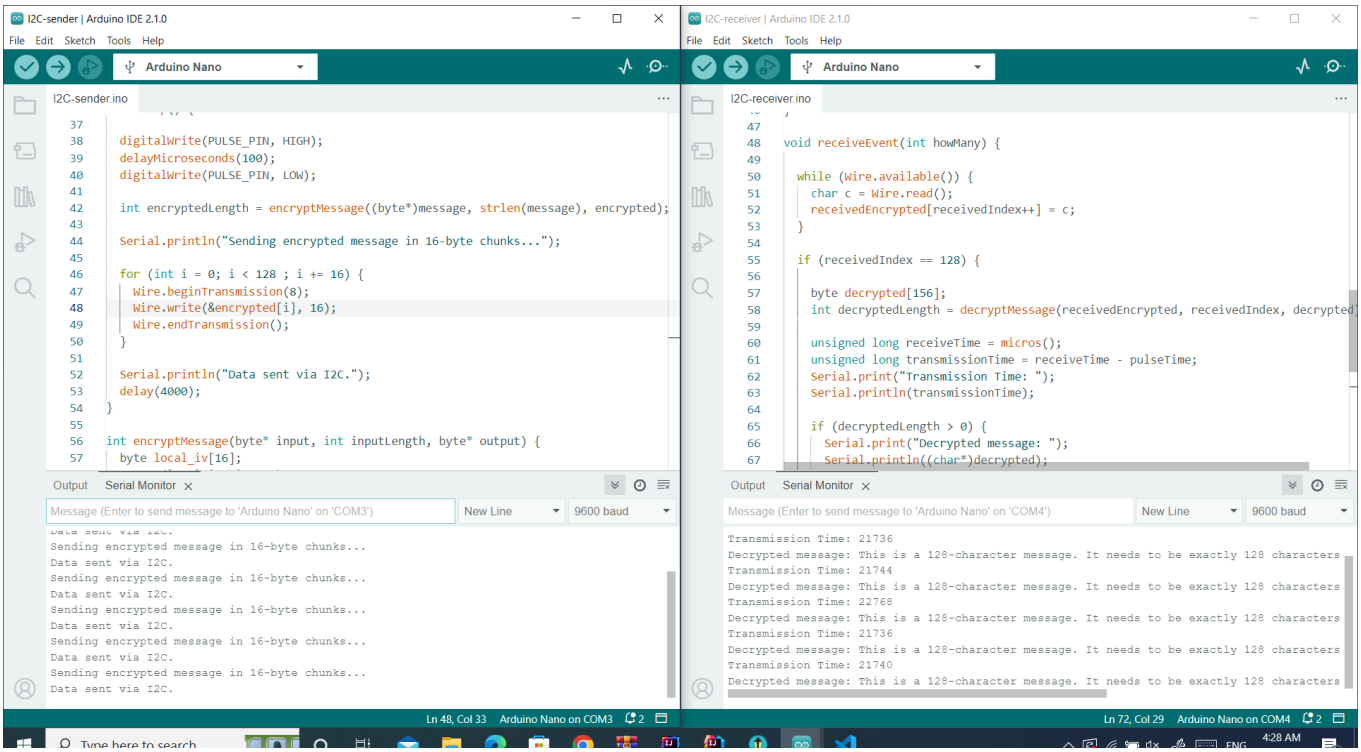
شکل ۱ برد نهایی I2C

مراحل پیاده‌سازی:

۱. تنظیم برد Masterبرای ارسال داده

۲. تنظیم برد Slaveبرای دریافت داده

۳. ارسال و دریافت داده با استفاده از توابع Wire.h



شکل ۲ تصاویر خروجی I2C

۲. پروتکل SPI

معرفی:

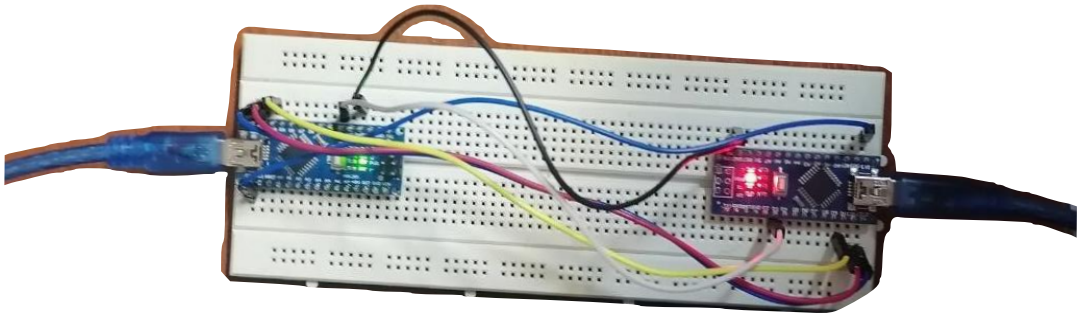
پروتکل SPI (Serial Peripheral Interface)یک روش ارتباطی سریال پرسرعت است که معمولاً برای ارتباط بین میکروکنترلرها و دستگاه‌های جانبی مانند کارت‌های حافظه، نمایشگرهای LCDو مبدل‌های ADC/DACبه کار می‌رود. این پروتکل از چهار خط MOSI (Master Out Slave In)، MISO (Master In Slave Out)، SCK (Serial Clock)و SS (Slave Select) برای انتقال داده استفاده می‌کند .نسبت به I2Cسرعت بالاتری دارد اما تعداد سیم‌های بیشتری نیاز دارد. این پروتکل معمولاً در مواردی که نیاز به انتقال سریع داده است، مانند پردازش تصویر و ذخیره‌سازی اطلاعات، استفاده می‌شود. ارتباط در SPIبه‌صورت Full Duplex است، به این معنی که داده می‌تواند به‌طور هم‌زمان ارسال و دریافت شود.

ویژگی‌ها:

- استفاده از ۴ سیم (MOSI، MISO، SCK، SS)
- پشتیبانی از سرعت‌های بسیار بالا (تا چندین مگابیت بر ثانیه)
- ساختار Master-Slave ساده

نحوه‌ی اتصال دو برد آردوینو:

- MOSI به MOSI (D۱۱)
- MISO به MISO (D۱۲)
- SCK به SCK (D۱۳)
- SS (CS) به SS (D۱۰)
- GND به GND



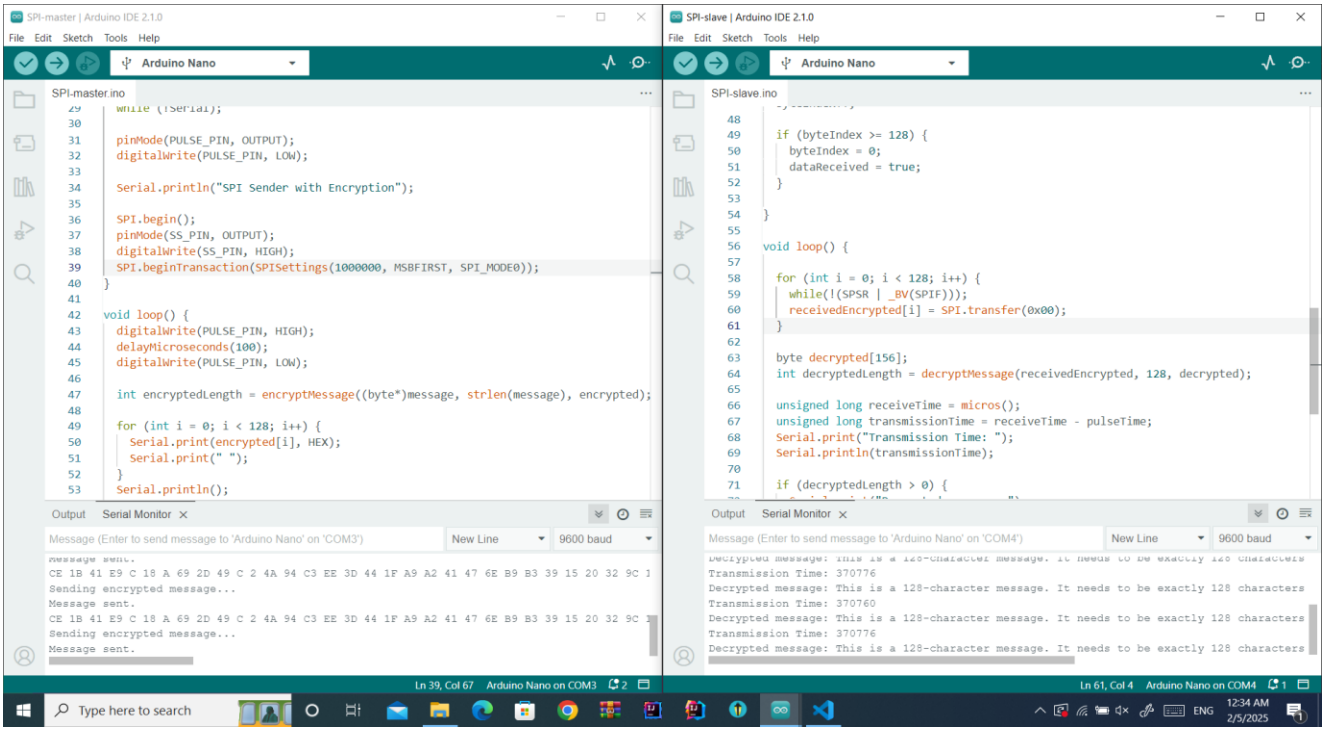
شکل ۳ برد نهایی SPI

مراحل پیاده‌سازی:

1. تنظیم Master برای ارسال داده

2. تنظیم Slave برای دریافت داده

3. تبادل داده با استفاده از کتابخانه SPI.h



شکل ۴ تصاویر خروجی کد SPI

۳. پروتکل UART

معرفی:

پروتکل UART (Universal Asynchronous Receiver-Transmitter) یکی از قدیمی‌ترین روش‌های ارتباطی سریال است که در آن داده‌ها بدون نیاز به کلاک مشترک بین دو دستگاه ارسال و دریافت می‌شوند. در این روش، دستگاه‌های فرستنده و گیرنده باید روی یک نرخ انتقال داده مشخص تنظیم شوند . UARTدر ارتباطات سریال پایه، ماژول‌های GPS، بلوتوث و ماژول‌های مخابراتی به کار می‌رود. این پروتکل

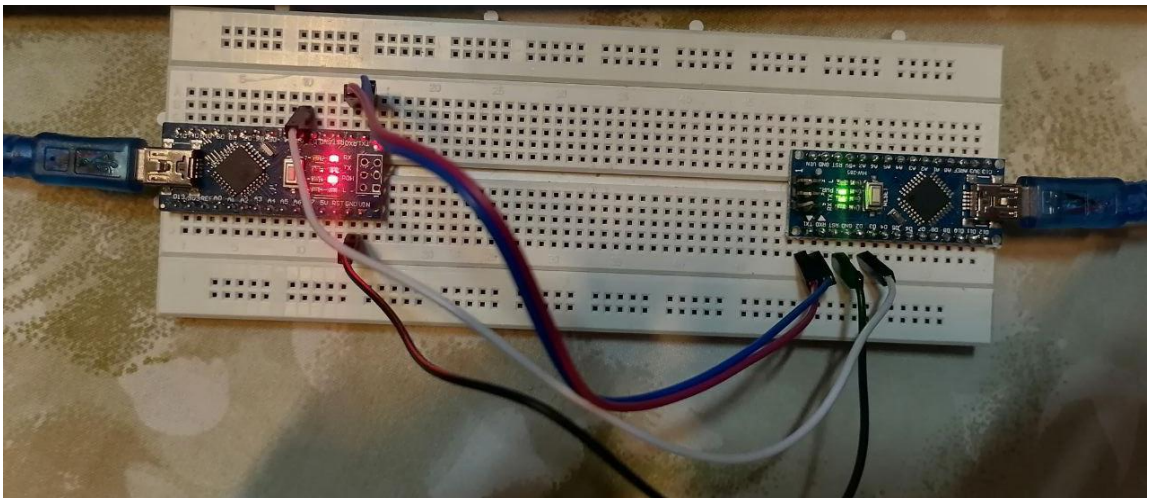
دارای دو خط اصلی TX (Transmit)و RX (Receive) است که داده را ارسال و دریافت می‌کنند. در مقایسه با I2Cو SPI، پروتکل UARTساده‌تر است اما نیاز به تنظیم دقیق نرخ انتقال داده دارد.

ویژگی‌ها:

- استفاده از دو سیم TX و RX
- عدم نیاز به سیگنال کلاک
- نرخ تبادل داده قابل تنظیم (۹۶۰۰، ۵۷۶۰۰، ۱۱۵۲۰۰ بیت بر ثانیه)

نحوه‌ی اتصال دو برد آردوینو:

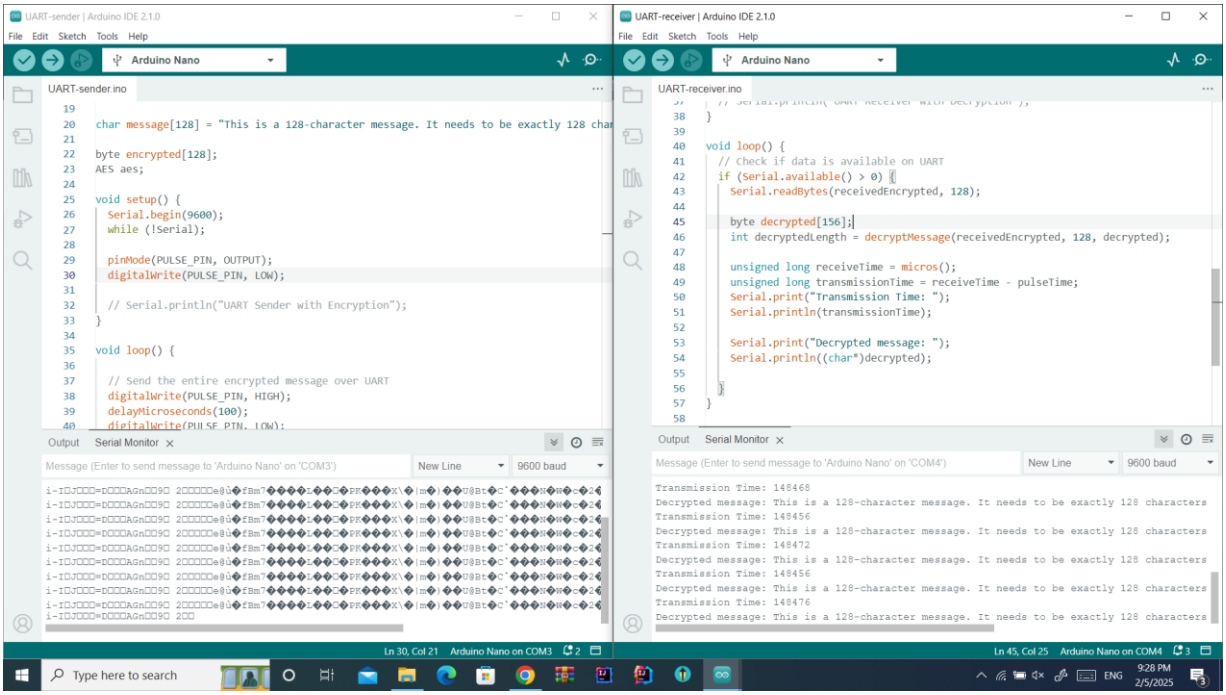
- از TX برد اول به RX برد دوم
- از RX برد اول به TXبرد دوم
- GND به GND



شکل ۵ برد نهایی UART

مراحل پیاده‌سازی:

1. تنظیم سرعت و مقدار داده در دو برد
2. ارسال و دریافت داده با استفاده از Serial.begin()
3. نمایش داده‌ها در Serial Monitor



شکل ۶ تصاویر خروجی کد UART

۴. پروتکل Wi-Fi

معرفی:

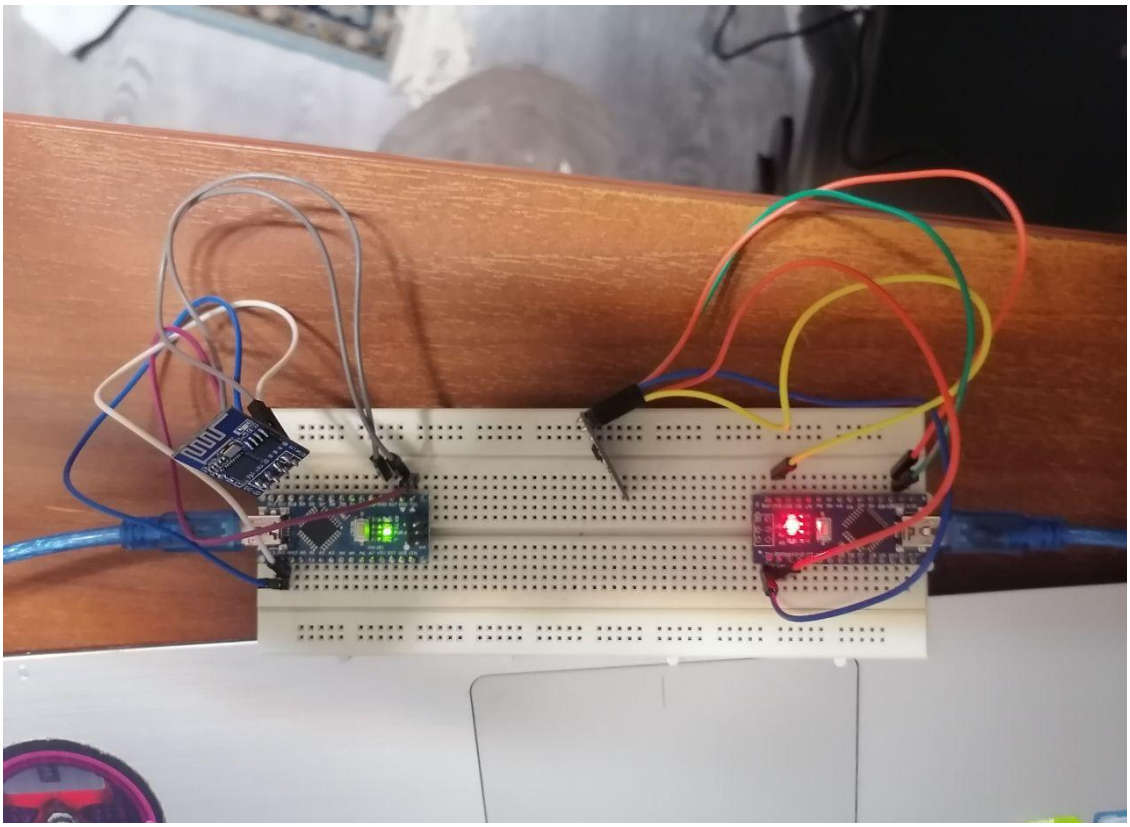
Wi-Fi یک روش ارتباط بی‌سیم است که از طریق امواج رادیویی داده‌ها را ارسال و دریافت می‌کند. در این پروژه، از ماژول ESP8266 برای اتصال بردهای آردوینو به شبکه استفاده می‌شود. این پروتکل در اینترنت اشیا (IoT) و سیستم‌های کنترل از راه دور کاربرد دارد Wi-Fi .امکان ارسال داده به بردهای دیگر و همچنین اتصال به سرورهای اینترنتی را فراهم می‌کند. این پروتکل از استانداردهای ۲.۴ گیگاهرتز و ۵ گیگاهرتز پشتیبانی می‌کند و بسته به شرایط محیطی برد قابل توجهی دارد.

ویژگی‌ها:

- انتقال داده بی‌سیم با برد زیاد
- نرخ انتقال داده بالا (تا چند صد مگابیت بر ثانیه)
- امکان ارتباط بردها با شبکه‌های محلی و اینترنت

نحوه‌ی اتصال دو برد آردوینو:

- استفاده از ماژول ESP8266
- برقراری ارتباط یک برد به عنوان سرور و دیگری به عنوان کلاینت



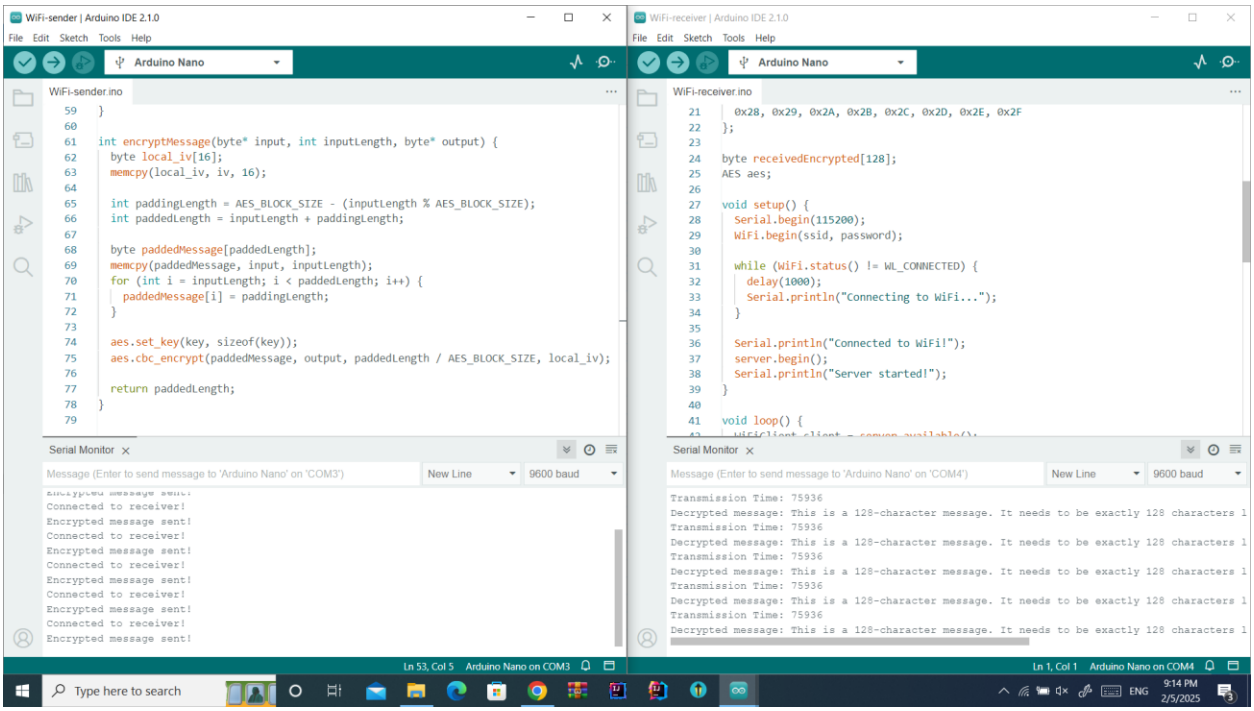
شکل ۷ برد نهایی WIFI

مراحل پیاده‌سازی:

1. تنظیم SSIDو Passwordشبکه

2. راه‌اندازی سرور و کلاینت با استفاده از WiFi.h

3. ارسال و دریافت داده از طریق پروتکل TCP یا UDP



شکل ۸ تصویر خروجی WIFI

۵. پروتکل CAN

معرفی:

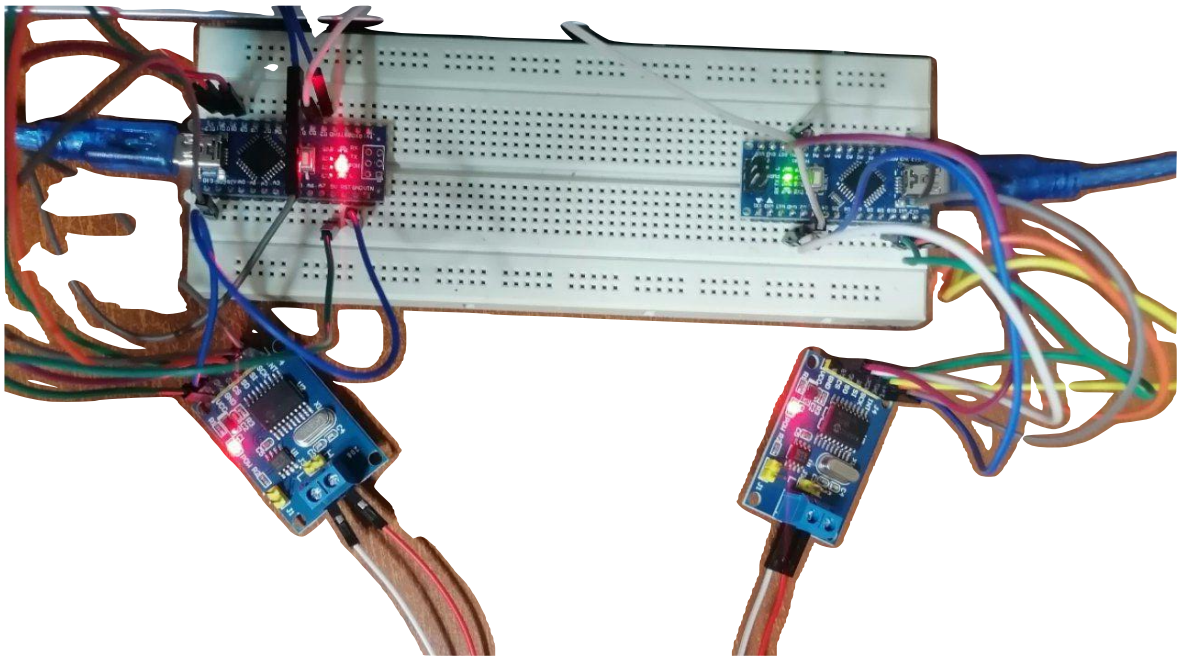
پروتکل CAN (Controller Area Network) یک استاندارد ارتباطی سریال است که در سیستم‌های صنعتی و خودرویی برای تبادل داده بین میکروکنترلرها استفاده می‌شود. این پروتکل به دلیل پایداری بالا، مقاومت در برابر نویز و قابلیت کار در محیط‌های صنعتی پر نویز، کاربرد گسترده‌ای دارد.

ویژگی‌ها:

- ارتباط چند دستگاه بدون نیاز به Master
- مقاوم در برابر نویز
- استفاده از دو سیم CAN_H و CAN_L

نحوه‌ی اتصال دو برد آردوینو:

- استفاده از ماژول MCP2515 CAN
- از CAN_H برد اول به CAN_H برد دوم
- از CAN_L برد اول به CAN_L برد دوم
- اتصال پایه های ماژول به پین‌های SPI در آردوینو



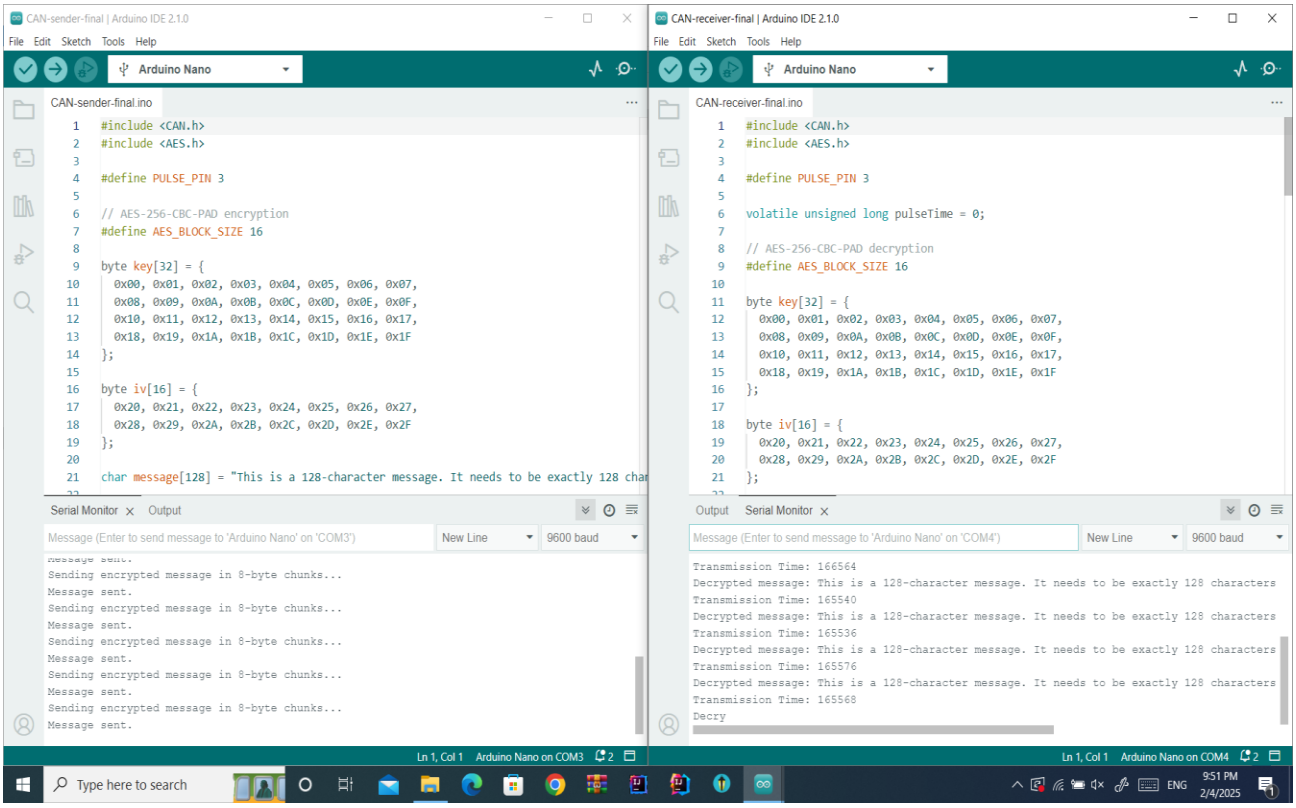
شکل ۹ برد نهایی CAN

مراحل پیاده‌سازی:

1. مقداردهی اولیه ماژول CAN.

2. ارسال پیام از یک برد و دریافت توسط دیگری

3 .پردازش داده‌ها



شکل ۱۰ تصاویر خروجی کد CAN

۶. پروتکل One-Wire

معرفی:

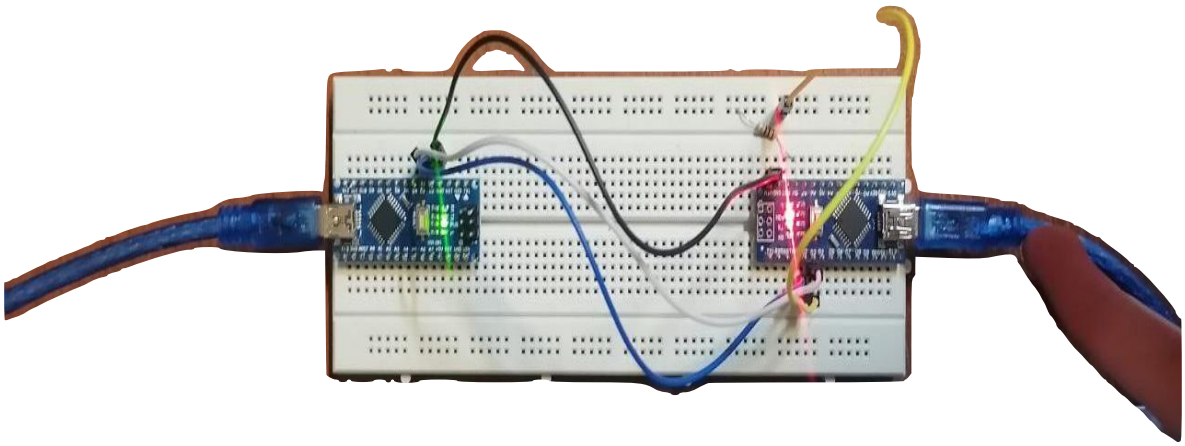
پروتکل One-Wire یک روش ارتباطی سریال ساده و کم‌هزینه است که تنها از یک سیم دیتا به همراه GNDاستفاده می‌کند. این پروتکل معمولاً در سنسورهای دما مانند DS18B20 و سیستم‌های خواندن RFIDبه کار می‌رود One-Wire .نیازی به خطوط اضافی مانند کلاک ندارد و به دلیل طراحی ساده، در بسیاری از سنسورها و سیستم‌های نهفته کاربرد دارد.

ویژگی‌ها:

- استفاده از یک سیم برای انتقال داده
- قابلیت اتصال چند دستگاه به یک خط
- سرعت پایین در مقایسه با SPIو I2C

نحوه‌ی اتصال دو برد آردوینو:

- یک سیم دیتا بین دو برد متصل شود
- استفاده از مقاومت پول‌آپ ۴.۷ کیلو اهم

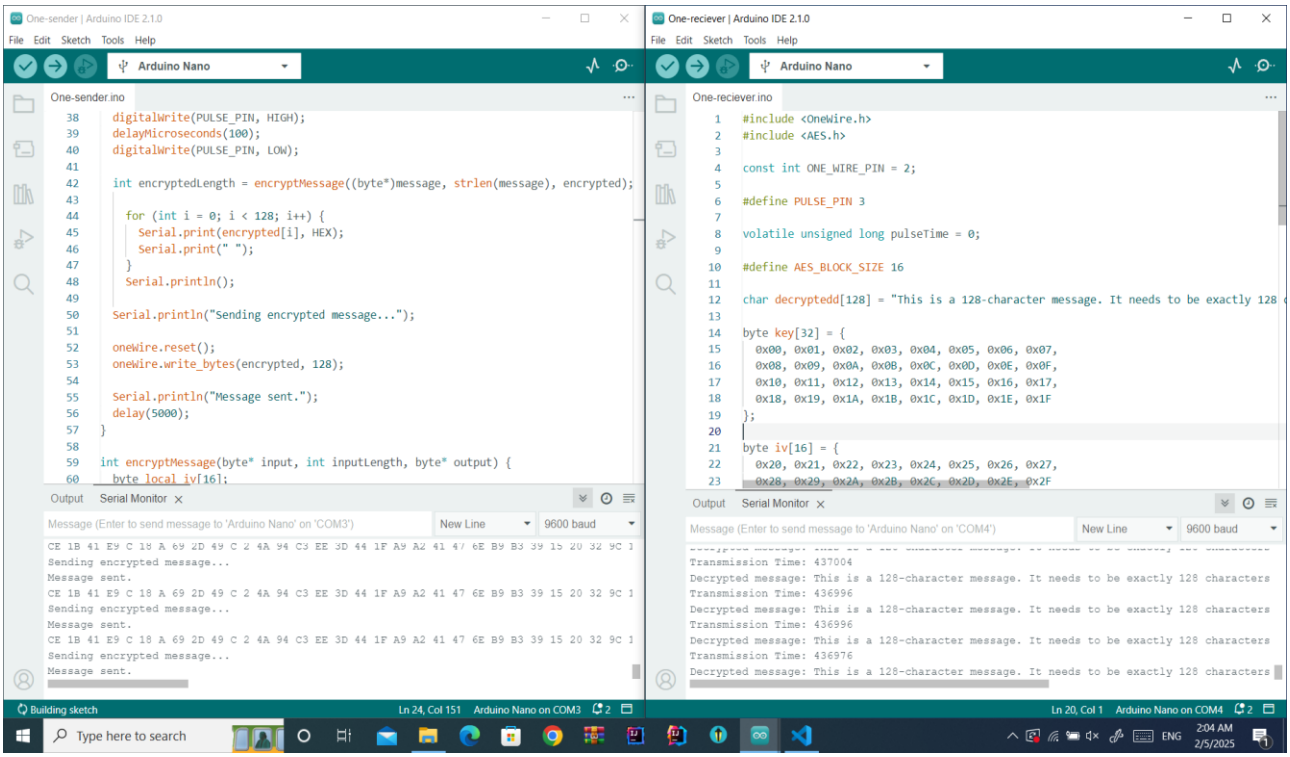


شکل ۱۱ برد نهایی One-Wire

مراحل پیاده‌سازی:

1. مقداردهی اولیه پروتکل با کتابخانه OneWire.h
2. ارسال و دریافت داده از طریق یک خط

3 .پردازش اطلاعات دریافت‌شده



شکل ۱۲ تصاویر خروجی کد One-Wire

مقایسه پروتکل‌ها:

Encryption & Decryption:

در تمام از کدها از کتابخانه AESLib و توابع آن برای Encryption و Decryption به صورت مشابهی استفاده شده است. کلیدها نیز برای همه یکسان در نظر گرفته شده است.

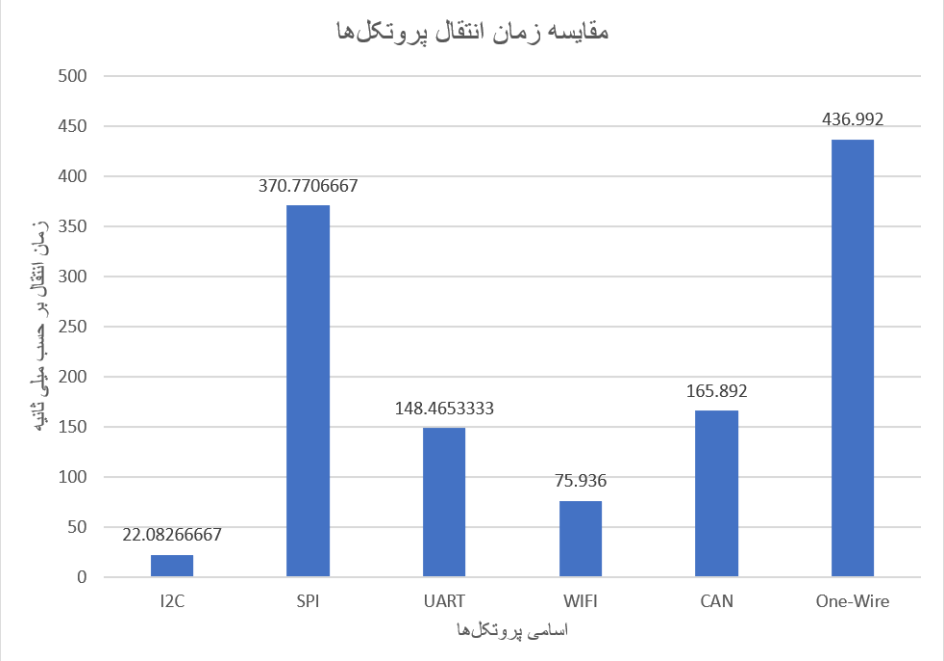
اندازه گیری زمان:

در تمام موارد برای اندازه گیری زمان ارسال از Digital Pin 3 استفاده شده است. به این صورت که فرستنده پیش از شروع کار یک پالس را به واسطه این پین به گیرنده ارسال می‌کند و در آن لحظه گیرنده زمان را ثبت

می‌کند. سپس انتقال پیام انجام می‌شود و در نهایت گیرنده مجددا زمان را ثبت می‌کند. با محاسبه اختلاف این دو زمان، transitionTime محاسبه می‌شود. (دریافت پالس در گیرنده به صورت event driven پیاده سازی شده است.)

مقایسه زمان‌ها:

با توجه به اعداد مشخص شده در تصاویر می‌توان نتیجه گرفت که پروتکل I2C در بین این پروتکل‌ها از سرعت بیشتری برخوردار است. پس از آن One-Wire ,SPI ,CAN ,UART ,WIFI قرار دارند.
توجه کنید که با توجه به کتابخانه‌های مختلف و توابع و پارامترهای مختلف این زمان‌ها می‌تواند متفاوت باشد. برای مثال در UART با تغییر baudRate برای پورت Serial زمان انتقال پیام تغییر چشمگیری خواهد داشت.



نمودار ۱

نتیجه‌گیری:

در این سند، پروتکل‌های مختلف ارتباطی مورد بررسی قرار گرفتند و نحوه‌ی اتصال دو برد آردوینو برای هر پروتکل شرح داده شد. در ادامه، مقایسه‌ی عملکرد آن‌ها انجام خواهد شد تا مشخص شود کدام روش برای شرایط مختلف بهتر است.