

سیستم‌های نهفته‌ی بی‌درنگ

گزارش تمرین شماره‌ی دو

آشنایی با پروتکل‌های ارتباطی و سنسورها

اعضای گروه

سعید زنگنه

محمدعلی توفیقی

حسین سلطانیلو

سؤالات:

۱. پروتکل I2C قابلیت اتصال Multi Masters - Multi Slaves را دارد. توضیح دهید مشکلات پیش روی این حالت چیست؟ در این پروتکل از چه مکانیزم داوری‌ای استفاده می‌شود و چگونه مشکل تداخل را حل کرده است؟

مشکل زمانی بروز پیدا می‌کند که دو یا چند Master بخواهند داده‌ای را از طریق SDA ارسال یا دریافت کنند. برای حل این مشکل، هر Master باید پیش از ارسال پیام، از صفر یا یک بودن خط SDA مطلع شود. اگر خط صفر است یعنی یک Master دیگر در حال تبادل پیام است و باس را در اختیار دارد و بنابراین Master فعلی باید دست نگه دارد. اگر هم خط SDA دارای ولتاژ بالا است، پس یعنی خط آزاد است و تبادل پیام می‌تواند به‌طور امن انجام شود. گاهی نیز پیش می‌آید که دو یا چند Master به‌طور تقریباً هم‌زمان اقدام به ارسال یا دریافت پیام کنند. در این شرایط از مکانیزم Bus Arbitration استفاده می‌شود که در آن یک Master هنگام ارسال یک بیت، دقت می‌کند که آن بیت ارسالی با سطح ولتاژ فعلی مورد انتظار SDA هم‌خوانی داشته باشد. هر Masterی که زودتر متوجه این موضوع شود یا این‌که آدرس Slave مرتبط با آن بزرگ‌تر باشد، ارسال یا دریافت را رها می‌کند و SCL را نیز قطع می‌کند و منتظر می‌ماند تا زمانی که بیت STOP را مشاهده کند و بتواند مجدداً شروع به ارسال یا دریافت پیام کند.

۲. مراحل ارتباط دو دستگاه به یک‌دیگر از طریق این پروتکل (I2C) را بیان کنید.

۱. ارسال بیت شروع ارتباط از طرف Master

۲. ارسال آدرس Slave که خواهان ارتباط با آن هستیم از طرف Master

۳. ارسال بیت R/W از طرف Master تا مشخص شود که ارتباط از نوع خواندن است یا نوشتن

۴. انتظار برای دریافت بیت Acknowledgment از طرف دریافت‌کننده جهت راستی‌آزمایی ارتباط

۵. ارسال یا دریافت پیام موردنظر در قالب ۸ بیت

۶. انتظار برای دریافت بیت Acknowledgment از طرف دریافت‌کننده جهت راستی‌آزمایی ارتباط

۷. ارسال یا دریافت پیام موردنظر در قالب ۸ بیت

۸. انتظار برای دریافت بیت Acknowledgment از طرف دریافت‌کننده جهت راستی‌آزمایی ارتباط

۹. ارسال بیت خاتمه‌ی ارتباط از طرف Master

۳. در اتصال UART (که معروف‌ترین پروتکل ارتباط سریال می‌باشد)، نقش پارامتر Baud Rate چیست؟ و به چه دلیل این پارامتر اهمیت پیدا می‌کند؟

یک گیرنده پس از دریافت بیت شروع ارتباط باید بداند که با چه نرخ اطلاعات ارسال می‌شود تا آن‌ها را بخواند. این نرخ Baud Rate نام‌گذاری شده است و واحد آن هم بیت‌برثانیه است. بدیهی‌ست که این نرخ باید پیش از ارتباط بین هر دو طرف توافق شده باشد تا مشکلاتی از قبیل این‌که فرستنده با نرخ بالاتر یا پایین‌تر از خواندن گیرنده اقدام به ارسال اطلاعات کند پیش نیاید و باعث نشود داده‌های غلط به دست گیرنده برسد. معمولاً سقف تفاوت بین Baud Rate فرستنده و گیرنده می‌تواند ۱۰ درصد باشد؛ وگرنه داده‌های مخابره‌شده اشتباه خواهد بود. ضمن این‌که در Rate‌های بالاتر، نیاز است که طول سیم کوتاه شود و کیفیت اتصال سیم به دیوایس‌ها اهمیت بیشتری پیدا می‌کند.

۴. هرکدام از روش‌های I2C و UART مزایا و معایبی دارند. این دو را باهم مقایسه کنید و برتری‌های هریک را برشمارید.

● پروتکل I2C:

○ مزایا:

- در آن فقط از دو سیم استفاده می‌شود.
- از چند Master و Slave در آن پشتیبانی می‌شود.
- با استفاده از بیت ACK می‌توان صحت ارسال اطلاعات را سنجید.
- استفاده از آن نسبت به UART، طراحی را ساده‌تر می‌کند.

○ معایب:

- اندازه‌ی بسته‌ی اطلاعات محدود به ۸ بیت است.

● پروتکل UART:

○ مزایا:

- در آن فقط از دو سیم استفاده می‌شود.

- در آن نیازی به استفاده از سیگنال کلاک نیست.
- با استفاده از بیت Parity می‌توان صحت ارسال اطلاعات را سنجید.
- انعطاف کوتاهی در اندازه‌ی بسته‌ی ارسالی وجود دارد.

○ معایب:

- اندازه‌ی بسته‌ی ارسالی محدود به ۹ بیت است.
- نرخ Baud فرستنده و گیرنده می‌تواند حداکثر ۱۰ درصد تفاوت داشته باشد.

۵. در مورد سریال نرم‌افزاری و نحوه‌ی کارکرد آن توضیح مختصری ارائه دهید. محدودیت‌های آن نسبت به سریال سخت‌افزاری چیست؟

سخت‌افزار آردوینو پشتیبانی داخلی برای ارتباط سریال را از طریق پین‌های ۰ و ۱ برقرار می‌کند. این پشتیبانی از طریق قطعه‌ای سخت‌افزاری به نام UART انجام می‌پذیرد. این سخت‌افزار به چیپ Atmega اجازه می‌دهد که داده‌های سریال را حتی در موقعی که در حال کار بر روی سایر تسک‌هاست، دریافت کند؛ البته تا زمانی که در بافر ۶۴ بیتی آن فضایی وجود داشته باشد.

کتابخانه‌ی SoftwareSerial به صورت نرم‌افزاری این اجازه را می‌دهد که بر روی سایر پین‌های آردوینو، ارتباط سریال برقرار گردد و این امکان را فراهم می‌سازد که به صورت همزمان چندین پورت سریال فعال باشند.

از محدودیت‌هایی که در استفاده از ارتباط سریال نرم‌افزاری وجود دارد، می‌توان به این اشاره کرد که بعضی از کتابخانه‌های سریال نرم‌افزاری، فقط Baud Rate های خاصی را پشتیبانی می‌کنند. (مثلاً NewSWSerial که فقط Baud Rate های ۹۶۰۰، ۱۹۲۰۰ و ۳۸۴۰۰ را پشتیبانی می‌کند).

همچنین از محدودیت‌های اصلی آن می‌توان به این اشاره کرد که سریال نرم‌افزاری بروز وقفه را در تمام زمانی که یک کاراکتر در حال ارسال یا دریافت است غیر فعال می‌کند و نمی‌تواند هم‌زمان عمل ارسال و دریافت را انجام دهد.

گفتنی است کتابخانه‌ی SoftwareSerial فقط تا نرخ ۹۶۰۰ برای Baud Rate قابل اعتماد است و در نرخ‌های بالاتر اتکاپذیری بالایی ندارد.

۶. نحوه‌ی کارکرد سنسور فاصله‌سنج گفته‌شده را توضیح دهید. دقت این سنسور در چه حدودی است؟ حداقل فاصله‌ی لازم برای دریافت داده‌ی صحیح از این سنسور و بیشترین فاصله‌ای که می‌تواند پوشش دهد تقریباً چقدر است؟



این سنسور دو فرستنده و گیرنده‌ی فراصوت دارد. ابتدا فرستنده، یک موج فراصوت را ارسال می‌کند و این موج پس از انتشار و برخورد به یک جسم، بازتابیده می‌شود و سنسور گیرنده آن را دریافت می‌کند. سپس با استفاده از تأخیر زمانی انتشار و دریافت مجدد آن موج و با توجه به سرعت صوت، می‌توان فاصله‌ی بین سنسور تا جسم را سنجید که از رابطه‌ی ساده‌ی زیر استفاده می‌شود:

$$\text{مسافت} = \text{سرعت} \times \text{زمان}$$

گفتنی‌ست این سنسور پس از دریافت بازتاب موج، به اندازه‌ی زمانی که این رفت‌و برگشت طول کشیده است یک پایه‌ی خود را دارای ولتاژ می‌کند تا از این طریق زمان به دست آید. سپس با استفاده از یک پردازنده و با استفاده از رابطه‌ی بالا می‌توان مسافت رفت‌و برگشت موج را محاسبه کرد و با نصف کردن آن، به فاصله دست یافت.

- دقت: ۳ میلی‌متر
- حداقل فاصله‌ی لازم برای دریافت داده‌ی صحیح: ۲ سانتی‌متر
- بیش‌ترین فاصله‌ی تحت پوشش: ۸۰ سانتی‌متر به‌طور عملی و ۴۵۰ سانتی‌متر در تئوری

۷. اگر در پروژه‌ای نیاز به استفاده از دو ماژول فاصله‌سنج داشته باشیم، چه مشکلی می‌تواند به وجود بیاید؟

مهم‌ترین مشکل تداخل میان امواج فراصوت این دو ماژول است. مثلاً حالتی را می‌توانیم در نظر بگیریم که دو ماژول هم‌زمان اقدام به ارسال موج می‌کنند ولی موجی که به گیرنده‌ی هرکدام از آن‌ها می‌رسد، مربوط به دیگری است؛ بنابراین فاصله‌ی شیء متفاوتی سنجیده می‌شود و اساس صحت سیستم زیر سؤال می‌رود.

۸. اگر نیازمند به اتصال چند سنسور مختلف که از پروتکل I2C استفاده می‌کنند بشویم، آیا در پیاده‌سازی با مشکلی روبه‌رو می‌شویم؟ حال اگر دو سنسور از یک نوع (مثلاً دوتا سنسور 6050MPU) داشته باشیم چطور؟ اگر سه‌تا از این سنسور داشته باشیم چطور؟ (جواب به‌ترتیب خیر، خیر، بله است! توضیح دهید چرا)

الف) از آن‌جا که در ارتباط I2C نیازمندیم که شماره‌ی دیوایسی که قصد ارتباط با آن را داریم مشخص کنیم، این که بخواهیم به چند سنسور مختلف متصل شویم مشکلی ایجاد نمی‌شود و فقط سنسوری که مشخص کردیم اطلاعات را دریافت خواهد کرد.

ب) بسیاری از سنسورها و دیوایس‌ها پینی برای تخصیص آدرس آلترناتیو دارند که در صورتی که آن را ست کنیم، از آدرس دیگری استفاده می‌کند که کانفلیکت پیش نیاید. مثلاً در همین سنسور 6050MPU می‌توان از پین ADO برای سوئیچ بین آدرس‌های Ox68 و Ox69 استفاده کرد.

ج) پینی که در قسمت فوق به آن اشاره شد تنها ۲ حالت دارد و به همین دلیل نمی‌توان برای آن حالت سوم در نظر گرفت و ۳ آدرس برای سنسور I2C ایجاد کرد. البته روش‌های پیچیده‌تری برای ایجاد امکان اتصال بیش از ۲ سنسور مشابه I2C وجود دارد که فرض این است که از این روش‌ها استفاده نمی‌کنیم.

۹. طراحی مفهومی این تمرین را مانند نمونه‌ای که در تمرین اول دیدید رسم کنید.

