



دانشکده مهندسی کامپیوتر

سیستم‌های نهفته و بی‌درنگ

تمرین تحقیقاتی

امیرحسین احمدی

97522292

## 1- اطلاعات قطعات سیستم و قیمت آنها

- میکروکنترلر: مرکز اصلی این سیستم میباشد که وظیفه پردازش بر عهده آن است که بتواند اطلاعات سرعت، جهت حرکت و ترمز را محاسبه و به سیستم حرکتی خودرو ارسال کند. با توجه به نیازهای پردازش تصویر برای تشخیص عابران پیاده و تخمین فواصل بهتر است از یک برد مجهز به جی پی یو استفاده کنیم. همچنین برای دریافت وضعیت ترافیک و وضعیت چراغ ها نیاز به اتصال بیسیم به شبکه اینترنت داریم. قطعه مورد استفاده: Raspberry Pi 4 بر اساس ARM به همراه جی پی یو و ماژول اینترنت داخلی

[https://esys.ir/Product\\_Detail.aspx?Type=2&Id=2461](https://esys.ir/Product_Detail.aspx?Type=2&Id=2461)

قیمت: 6 میلیون 620 هزار تومان

نوع: مجموع قطعه دیجیتال ولی بخشهایی از آن مانند ماژول اینترنت برای ارسال و دریافت اطلاعات روی آن آنالوگ هستند.

- جی پی اس: برای بدست آوردن موقعیت مکانی خودرو برای پیدا کردن وضعیت ترافیک در آن منطقه نیاز به یک ماژول جی پی اس داریم. قطعه مورد استفاده: NEO-6M

[https://esys.ir/Product\\_Detail.aspx?Type=140&Id=904](https://esys.ir/Product_Detail.aspx?Type=140&Id=904)

قیمت: 250 هزار تومان

نوع: آنالوگ

- دوربین: برای تخمین فواصل از عابرین پیاده و خودروهای جلویی نیاز به یک دوربین داریم که با ارسال تصاویر به پردازنده بتوانیم محاسبات مربوطه را انجام دهیم. قطعه مورد استفاده: IMX219-160 سازگار با ماژول های Raspberry Pi

[https://esys.ir/Product\\_Detail.aspx?Type=94&Id=1972](https://esys.ir/Product_Detail.aspx?Type=94&Id=1972)

قیمت: 874 هزار و 200 تومان

نوع: آنالوگ

مجموع هزینه ی قطعات استفاده شده برابر با 7 میلیون و 750 هزار تومان است که با توجه به نوع پیاده سازی ممکن است متفاوت باشد. همچنین در این سوال فرض شده که قابلیت اتصال به خودرو برای کنترل جهت و سرعت حرکت وجود دارد و نیاز به ماژول های اضافه برای این کار نیست.

## 2- بررسی نهفته، بی درنگ و هیبرید بودن

- نهفته بودن (Embedded): سیستم گفته شده یک سیستم نهفته است زیرا سیستم پردازش اطلاعات درون یک سیستم بزرگ تر (کنترل ماشین) قرار گرفته است، هدف استفاده از این سیستم پردازش اطلاعات نیست بلکه کنترل خودرو است، یک سیستم هوشمند است که با برق کار میکند و قطعات و نرم افزارهای داخلی از چشم کاربر پنهان است و فقط نتیجه را میبیند.

- بی درنگ بودن (Real-time):

سیستم کنترل خودرو نیاز است که به صورت real time بتواند در مورد سرعت، توقف و جهت حرکت تصمیم بگیرد و تاخیر و اشتباه در اینکار ممکن است فاجعه به بار بیاورد برای همین حتما یک سیستم بی درنگ است.

- هیبرید بودن (Hybrid):

اکثر سیستم های نهفته، هیبرید نیز هستند. این سیستم نیز از این قاعده مستثنی نیست، زیرا تشکیل شده از قطعات آنالوگ و دیجیتال که در کنار یکدیگر کار میکنند و با هم تعامل دارند.

## 3- ویژگی های طراحی یک سیستم نهفته

- Reliability: سیستم ما باید قابل اطمینان باشد و اشتباهی در هدایت خودرو نداشته باشد. هدایت ماشین وظیفه ی حساسی است و با یک اشتباه کوچک ممکن است جون افراد به خطر بیفتد.

- Maintainability: در صورتی که مشکلی برای قطعات و نحوه ی اتصالات آن ها پیش بیاید باید بتوان به سادگی آن ها را تعمیر یا تعویض کرد.

- **Availability:** از آنجایی که در رانندگی باید در لحظه تصمیم گرفت، در لحظه آماده بودن سیستم برای ما خیلی اهمیت دارد.
- **Safety:** از آنجایی که وظایف سنگینی مانند پردازش تصویر و همچنین اتصالات خطرناک مانند اتصال به برق داریم، باید به این نکات توجه کنیم تا خطرات اتصالی کردن، بالا رفتن دما و ... را کاهش دهیم.
- **Security:** اهمیت زیادی دارد، مثلا اگر کسی بتواند کنترل خودرو را بدست بگیرد میتواند حوادث بدی را شکل دهد.
- **Energy:** از آنجایی که پردازش تصویر انرژی زیادی میبرد باید حتما به منبع های انرژی توجه کنیم تا در این بخش به مشکل نخوریم.
- **Performance:** از آنجایی که سیستم به صورت **real time** باید تصمیمات مهمی بگیرد و محاسبات سنگینی انجام میدهیم، لازم است که پرفورمنس بالایی داشته باشیم.
- **Weight:** اولویت بالایی در سیستم ما ندارد زیرا حجم و وزن سیستم ما در مقابل خودرو بسیار ناچیز است.
- **Cost:** از آنجایی که این سیستم محاسبات پیچیده ای دارد طبیعی است که قطعات هزینه بری نیاز دارد و مجموعا سیستم گرانی است.
- **Code-size:** اهمیت چندانی ندارد و خروجی برای ما مهم تر است.

#### 4- مراحل طراحی و ساخت سامانه

- **Concept:** ایده کلی و کانسپت کار همان داک مربوط به تمرین است که توسط دستیاران آموزشی طراحی شده است.

- **Specification:** پس از ایده کلی نیاز است که نیازهای سیستم را به صورت دقیق در بیاوریم. به طور مثال برای گرفتن اطلاعات ترافیک نیاز داریم که بتوانیم به اینترنت متصل باشیم، برای ارزیابی خیابان نیاز است که بتوانیم تصاویر مربوط به آن را بررسی کنیم و برای اینکار نیاز به یک دوربین داریم و ...

- **HW/SW Partitioning:** در این مرحله باید سخت افزار و نرم افزار را از هم جدا کنیم. برای این کار به صورت تکرار شونده، ابتدا یک تخمین زده، سپس دنبال راه های بهتر میگردیم. در نهایت خروجی بدست آمده را از نظر مساحت، توان مصرفی مورد نیاز، کارایی و صحت سنجی بررسی میکنیم و دوباره آنقدر این مراحل را ادامه میدهیم تا به نتیجه ی مطلوب برسیم.

- **Design:** بعد از جدا کردن نرم افزار و سخت افزار نیاز است که طراحی سیستم را انجام دهیم. ابتدا طرح قسمت سخت افزاری را روی نرم افزارهای شبیه ساز مانند پروتئوس ایجاد میکنیم. همچنین اگر از سخت افزاری استفاده میکنیم که نیاز به سفارشی سازی دارد کد ورپلاگ یا وی اچ دی ال مربوط به آن را زده و سخت افزار بدست آمده را استفاده میکنیم. برای قسمت های نرم افزاری نیز کدهای مربوط به کنترلر را به زبان های مد نظر نوشته و بعد از کامپایل فایل خروجی آن (مثلا hex.) در پروتئوس آپلود میکنیم.

- **Merge:** در نهایت با تست قسمت سخت افزاری در کنار نرم افزار کامپایل شده سیستم را تست کرده و آن را اشکال یابی میکنیم و در نهایت آن ها را به صورت فیزیکی و با قطعات واقعی پیاده سازی و تست میکنیم.

- **Supply:** در نهایت محصول ما آماده است تا به عرضه نهایی رسیده و تولید انبوه گردد.