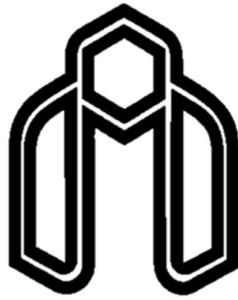


یا لطیف



دانشگاه صنعتی شهرود

دانشکده مهندسی برق

گزارش کارآموزی

تهییه و تنظیم: رضا آدینه پور

رضا آدینه پور - ۹۸۱۴۳۰۳

استاد مربوطه:

جناب اقای دکتر هادی گرابلو

تاریخ تهییه و ارائه:

شهریور ماه ۱۴۰۱

جناب آقای دکتر گرایلو

با سلام و احترام؛

متن فوق گزارش بnde از گذراندن دوره کارآموزی خود در شرکت دانش بنیان رادان الکترونیک فردوسی می باشد. اینجانب به مدت چندین ماه در این شرکت مشغول کارآموزی بوده‌ام و مهارت‌های مختلفی هم در زمینه فنی مرتبط با الکترونیک و هم در زمینه رفتار حرفه‌ای در محیط کار کسب کرده‌ام که در ذیل خلاصه‌ای از آن جهت اطلاع شما استاد بزرگوار ارائه می‌گردد. در برخی قسمت‌ها، تصاویری نیز تهییه کردم تا ارتباط با توضیحات نوشته شده را برای شما ملموس‌تر نماید.

از شما جهت زمانی که برای مطالعه و بررسی این گزارش صرف می‌نمایید سپاسگزارم.

ساختار اداری

شرکت دانش بنیان رادان الکترونیک فردوسی در سال ۱۳۹۵ با هدف طراحی و ساخت انواع بوردهای الکترونیکی و صنعتی و همچنین مشاوره و اجرای انواع سیستمهای اتوماسیون صنعتی و مانیتورینگ تشکیل شده است. این شرکت موفق شده تا با بهره‌گیری از دانش روز و بهترین فارغ التحصیلان دانشگاه‌های برتر کشور انواع پژوهش‌های محول شده از سمت کارفرمایان را در کمترین زمان و با کمترین هزینه به انجام برساند و حداکثر رضایت مشتری را جذب نماید.

از جمله چشم اندازهای شرکت رادان الکترونیک می‌توان به موارد ذیل اشاره نمود:

- ۱) تأمین نیاز تمام صنایع کشور در زمینه الکترونیک
- ۲) مکانیزه کردن و بهینه‌سازی فرایند تولید صنایع مختلف
- ۳) پیاده سازی استانداردهای روز دنیا در صنایع مختلف

عمده فعالیت این شرکت در خصوص طراحی، ساخت و تولید بوردهای الکترونیکی اعم از انواع مبدل‌های داده، مدار‌های کنترلی و ابزار دقیق، درایور در توان‌های مختلف، اجرای سیستم‌های SCADA است.

این شرکت دارای کارکنان مهندسی الکترونیک، بازرگانی، فروش و بازاریابی می‌باشد و تلاش می‌کند جدیدترین محصولات و تکنولوژی‌ها را در اختیار کلیه سازمانها و مراکز دولتی و خصوصی قرار دهد.

به طور کلی این سازمان از پنج واحد بازرگانی، تحقیق و توسعه، فروش، مالی، منابع انسانی تشکیل شده است که همگی به طور مستقیم تحت نظارت و رهبری مدیرعامل شرکت (عالی‌ترین مقام سازمانی) قرار دارند. شخص مدیرعامل نیز توسط هیئت مدیره انتخاب می‌شود. وظایف آن‌ها به طور خلاصه به صورت زیر است:

واحد مدیریت: مهم‌ترین رکن سازمان، رهبری آن می‌باشد. مدیرعامل به عنوان رهبر سازمان تصمیمات عالی را برای تعیین مسیر حرکت شرکت گرفته و وظایف هریک از مدیران را به آن‌ها ابلاغ می‌کند. در واقع مدیرعامل مسئول تمامی رفتارهای سازمان و نیز پاسخگوی تصمیمات خود در برابر هیئت مدیره می‌باشد. همچنین در این واحد مشاور مدیرعامل و مسئول دفتر، ایشان را در پیگیری أمور مربوط و تنظیم جلسات با مدیران یاری می‌کند.

واحد بازرگانی: وظیفه اصلی این واحد انجام همکاری‌ها و ساماندهی امور مرتبط با واردات کالاها می‌باشد. تمامی قراردادهای خرید محصولات، ترخیص کالاها در گمرک و همچنین خرید ارز جهت انجام مبادلات با شرکت‌های خارجی توسط کارمندان این واحد صورت می‌پذیرد و مدیر بازرگانی به حسن فعالیت کارمندان این بخش ناظرت دارد.

واحد تحقیق و توسعه: در این واحد، کارهای تحقیقاتی در زمینه توسعه محصولات و ارایه محصولات جدیدتر و به روزتر و در عین حال مناسب با نیاز روز جامعه انجام می‌گیرد و اینجانب در این بخش مشغول به کارآموزی بوده‌ام.

واحد فروش و مارکتینگ: پس از واردات کالاها و ترخیص آن‌ها و نهایتاً تایید کیفیت فنی محصولات، نوبت به حیاتی‌ترین وظیفه کارمندان شرکت می‌رسد؛ فروش محصولات. واحد فروش دو هدف اصلی مارکتینگ (بازاریابی) و فروش جزئی و کلی محصولات را دنبال می‌کند. در بخش بازاریابی و تبلیغات مدیر مارکتینگ وظیفه فراهم نمودن تبلیغات مناسب در همه سطوح جهت افزایش فروش محصولات و همچنین هماهنگی با شرکت‌های پخش و ارائه طرح‌های تشویقی برای فروشندگان آن‌ها را دارد. در بخش فروش نیز، سرپرست و کارمندان فروش وظیفه ارائه پیشنهاد قیمت فروش برای محصولات به مدیرعامل، سنجش پتانسیل بازار و سایر رقبا و نیز هماهنگی امور جزئی‌تر نظیر پیگیری سفارشات فروشگاه‌های زنجیره‌ای و شرکت‌های پخش در زمینه فروش محصولات هستند.

واحد مالی: تمامی امور مرتبط با خرید و فروش محصولات، نهایتاً با تایید مدیر مالی صورت می‌گیرد. از جمله این موارد می‌توان به قراردادها با سازمان‌های مختلف در پروژه‌ها، پیگیری دریافت‌ها و پرداخت‌های مالی به حساب شرکت و ارائه گزارش‌های مالی برای شرکت‌های بیمه و ... اشاره نمود. همچنین طرح کلی بودجه‌بندی و نیز ناظارت بر تامین سود موردنظر سالیانه شرکت از دیگر وظایف واحد مالی می‌باشد.

واحد منابع انسانی: اصلی‌ترین مسئولیت این واحد پیگیری تمامی امور مرتبط با استخدام نیروهای جدید و تمدید یا لغو قرارداد کارمندان است. همچنین فراهم آوردن برخی امکانات رفاهی-خدماتی برای کارمندان و نیز ناظارت بر ساعات کاری آن‌ها و نهایتاً محاسبه کارکرد و حقوق ماهیانه آن‌ها نیز در زمینه وظایف این واحد قرار می‌گیرد. انجام ارزیابی عملکرد نیز از مهمترین مسئولیت‌های این واحد است.

لازم به ذکر است که محل شرکت در مرکز رشد شماره ۴ دانشگاه فردوسی مشهد واحد ۴۱۴ است.

اعضای هیئت مدیره به صورت زیر است:

مهندس علی رحمانزاده کرمانی

رئیس هیئت مدیره
کارشناسی ارشد مهندسی برق - مخابرات
ali.rahmanzade@yahoo.com



کارشناسی ارشد مهندسی برق - مخابرات از دانشگاه صنعتی امیر کبیر

دکتر علی کریمپور

نایب رئیس هیئت مدیره
دکتری مهندسی برق - کنترل
karimpor@um.ac.ir



دکترای مهندسی برق کنترل و عضو هیئت علمی دانشگاه فردوسی مشهد

مهندس امیر مسعودیان

مدیر عامل
کارشناسی ارشد مهندسی برق - الکترونیک
amirmasoudian20@gmail.com



کارشناسی ارشد مهندسی برق - الکترونیک از دانشگاه فردوسی مشهد

اخلاق حرفه‌ای

به طور کلی اخلاق حرفه‌ای در این شرکت توسط همکاران رعایت می‌شود. به این معنا که افراد در هر سطحی که باشند، نسبت به سایرین وظایف و حقوقی دارند. به طور مثال کارشناسان نسبت به مدیران خود با احترام برخورد می‌کنند و وظایف محله را به درستی انجام می‌دهند و در عین حال که فضای صمیمی در بین تمامی کارکنان شرکت برقرار است، به عنوان مثال چنانچه یکی از کارمندان وظیفه خود را به درستی و کامل انجام ندهد مدیر مستقیم آن شخص به او یادآوری کرده و کمک می‌کند تا مسئولیت خود را بهتر انجام دهد. به نظر من مهمترین نکته در زمینه اخلاق حرفه‌ای همین مععدل نگهداشت رابطه بین همکاران در انجام امور است که باید توسط رفتار مدیران به سایرین آموزش داده شود.

از طرفی یکی از مهم ترین موارد در هر شرکت و سازمانی، ارزیابی عملکرد کارکنان توسط واحد مربوطه می‌باشد. به این معنا که با توجه به پست و شرح وظایف هر یک از کارمندان، نحوه عملکرد آن‌ها سنجیده می‌شود. این اتفاق فواید بسیار زیادی در جهت کارآیی سازمان دارد. همچنین با این روش می‌توان هم اخلاق حرفه‌ای و هم نحوه عملکرد در تخصص مربوطه را برای هریک از کارمندان سنجید و مناسب با عملکرد آن‌ها، تذکرات یا تشویق در دستور کار مدیران قرار گیرد.

از دیگر مصادیق اخلاق حرفه‌ای در این شرکت، جدا بودن امور کاری از سایر امور است. به این معنا که کارمندان در امور کاری شرکت، با یکدیگر کمال همکاری را دارند تا اهداف شرکت برآورده شود. چنانچه فردی با فرد دیگری مشکلی داشته باشد، این مشکلات را در کار دخیل نکرده و به صورت حرفه‌ای عمل می‌کند.

همچنین روابط دوستانه بین همکاران در محیط و فضای غیر اداری وجود دارد. به این صورت که هر زگاهی، همکاران در فضایی خارج از شرکت با یکدیگر جمعی را تشکیل داده و درباره سایر موضوعات غیرکاری صحبت می‌کنند.

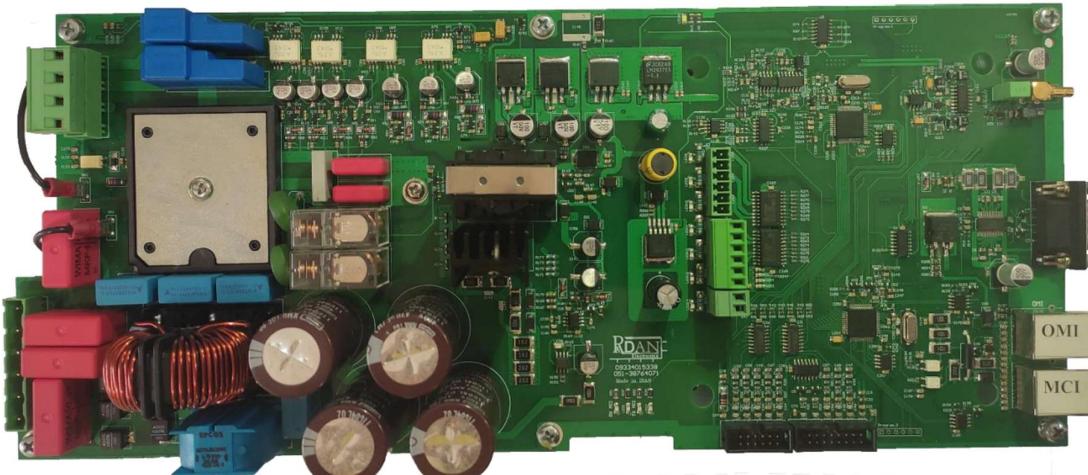
یکی دیگر از مصادیق اخلاق حرفه‌ای، کمک همکاران به یکدیگر در جهت پیشرفت کاری و ارتقای سازمانی می‌باشد. بر مبنای اصل شایسته سالاری، تمامی افراد در این سازمان امکان پیشرفت و ارتقا را دارند. در این زمینه، تمامی همکاران و مدیران به یکدیگر کمک می‌کنند تا با دستیابی به کیفیت بالاتر در امور کاری، افراد پیشرفت کرده و حتی از مدیران خود پیشی بگیرند. به نظر بند، رعایت این مصدق اخلاقی در هر شرکت و سازمانی ضروری و عیناً روحیه بخش است.

و نهایتاً آخرین مورد مربوط به ضایع نکردن زمان کاری و حضور به موقع است. هر کارمند در اینجا موظف است که ۶ ساعت و نیم از وقت روزانه خود را در ساعات مشخص مشغول به کار باشد. در طی این مدت تمامی همکاران تمرکز خود را روی امور خود معطوف کرده و از تلف کردن وقت دوری می‌کنند. هرچند که به صورت ناخواسته ممکن است دقایقی از این دوره زمانی به موجب انجام اموری شخصی مثل صرف ناهار یا ادای نماز شود اما همگی در رعایت حضور به موقع و صرف وقت اداری، برای امور اداری می‌کوشند.

خدمات و محصولات

در ادامه گزیده اندکی از محصولات تولیدی شرکت به عنوان نمونه همراه با توضیحات مختصر آورده شده است.

۱) برد درایو و کنترل دستگاه سرسیم زن KOMAX



این برد با دریافت اطلاعات و تنظیمات دستگاه سرسیم زن از طریق کامپیوتر، کلیه فرایندهای دستگاه سرسیم زن را کنترل مینماید. موتور استفاده شده در این دستگاه از نوع مغناطیس دائم (PM) بوده که از روش کنترلی FOC بهره میبرد. همچنین با اندازه گیری فشار وارد شده به سرسیم در هنگام پرس، نمودار فشار وارد شده را با نمودار فشار مرجع تعیین شده برای دستگاه مقایسه مینماید و از وارد شدن فشار استاندارد هنگام پرس سرسیم اطمینان حاصل میشود. بدین ترتیب کنترل کیفیت فرایند تولید همزمان بررسی میشود.

۲) مودم مبدل S7-Communication به Modbus



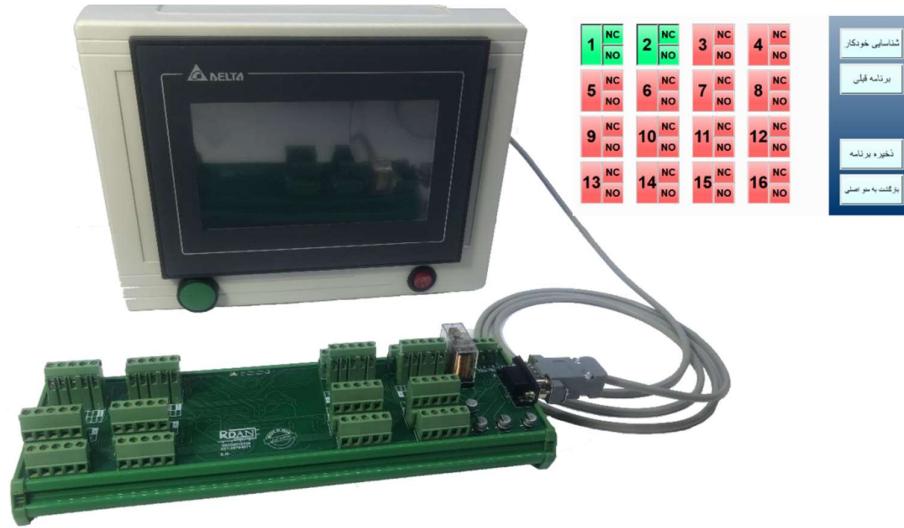
این مودم با جمع آوری داده‌های سنسورها مختلف نظیر دما، رطوبت و فشار از طریق ارتباط Modbus این اطلاعات را روی شبکه LAN و با پروتکل ارتباطی S7-Communication برای سرور سیستم مانیتورینگ ارسال مینماید. این اطلاعات بر روی اکثر بسترهای مانیتورینگ نظیر WINCC و Citect و همچنین HMI های تولیدی شرکتهای مختلف قابل مشاهده است و جایگزین ارزان قیمت و قابل اطمینانی PLC ها میباشد. همچنین در صورت خرابی سنسورها، خطاهای مناسب را برای سیستم SCADA ارسال مینماید.

(۳) تستر برد های ژاکارد



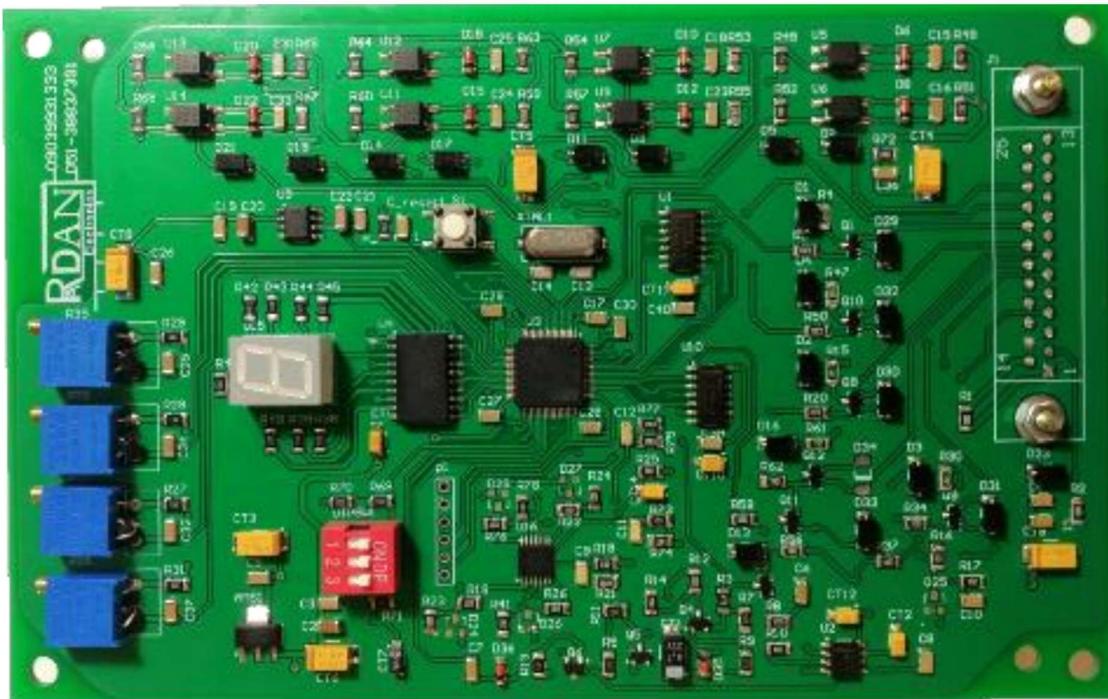
این دستگاه با اتصال به برد های ژاکارد، که برد های اصلی ماشینهای فرش بافی شرکت وندویل است، عملکردها و فرامین رمزگذاری شده این دستگاهها را برای آنها شبیه سازی میکند و آنها را عیب یابی نموده و قسمت معیوب در برد را بر روی HMI دستگاه نمایش میدهد. بدین وسیله فرایند تعمیر و نگهداری دستگاههای فرش بافی به مقدار زیادی کاهش یافته و از بروز ضررها ناشی از خرابی دستگاه به شکل قابل توجهی کاسته میشود.

۴) دستگاه تستر رله خودرو



این دستگاه قابلیت تست همزمان کنتاکت NO و کویل ۱۶ عدد رله تعییه شده درون دسته سیم خودرو با هر آرایش اتصالی را دارا میباشد. کار بر میتواند از طریق HMI دستگاه برنامه های مختلف تست ساده و ترکیبی را تعریف کرده و درون حافظه دستگاه ذخیره نماید. این دستگاه تست اتصال کوتاه و اتصال باز را بررسی کرده و نتیجه آن را برای کاربر روی HMI نمایش میدهد. همچنین دستگاه قابلیت تنظیم بازه جریان مجاز و مدت زمان عبور جریان از رله را دارد. همچنین سطوح دسترسی سی برای افراد مختلف قابل تعریف میباشد تا اوپراتور خط به تمام تنظیمات دستگاه دسترسی نداشته باشد.

(۵) برد کنترل لیفتراک



کنترل و تنظیم گشتاور لیفتراک، کنترل و تنظیم سرعت لیفتراک، حفاظت لیفتراک در برابر خطاهای احتمالی من جمله اضافه جریان و اضافه دما

(۶) برد کنترل پاریزون



کنترل ضخامت پاریزون، قابلیت اتصال خط کش دیجیتال و آنالوگ، دارای قابلیت اتوکالیبراسیون، فرمان پذیری از طریق Modbus

(7) برد فلومتر



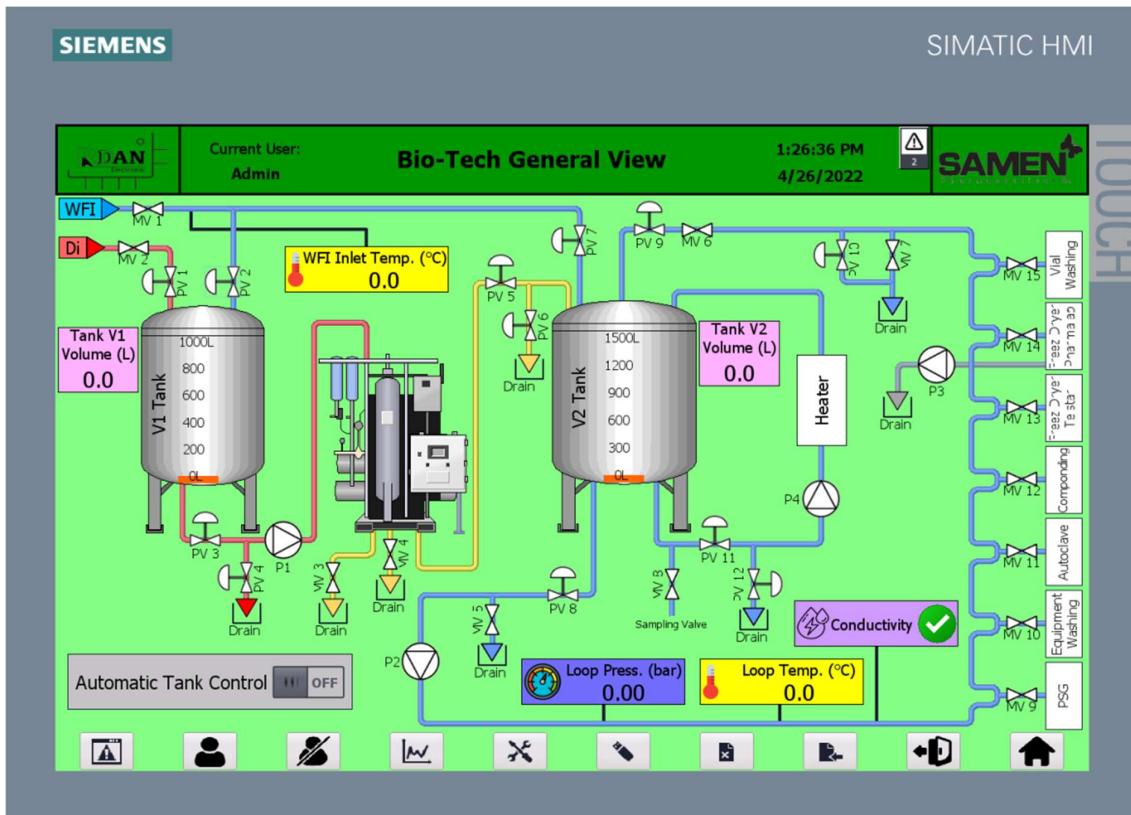
اندازه گیری جریان آب در لوله ها با مکانیزم حرارتی، دارای خروجی رله برای تشخیص وجود جریان آب

(8) کنترلر دمای تیغه جوش پلاستیک



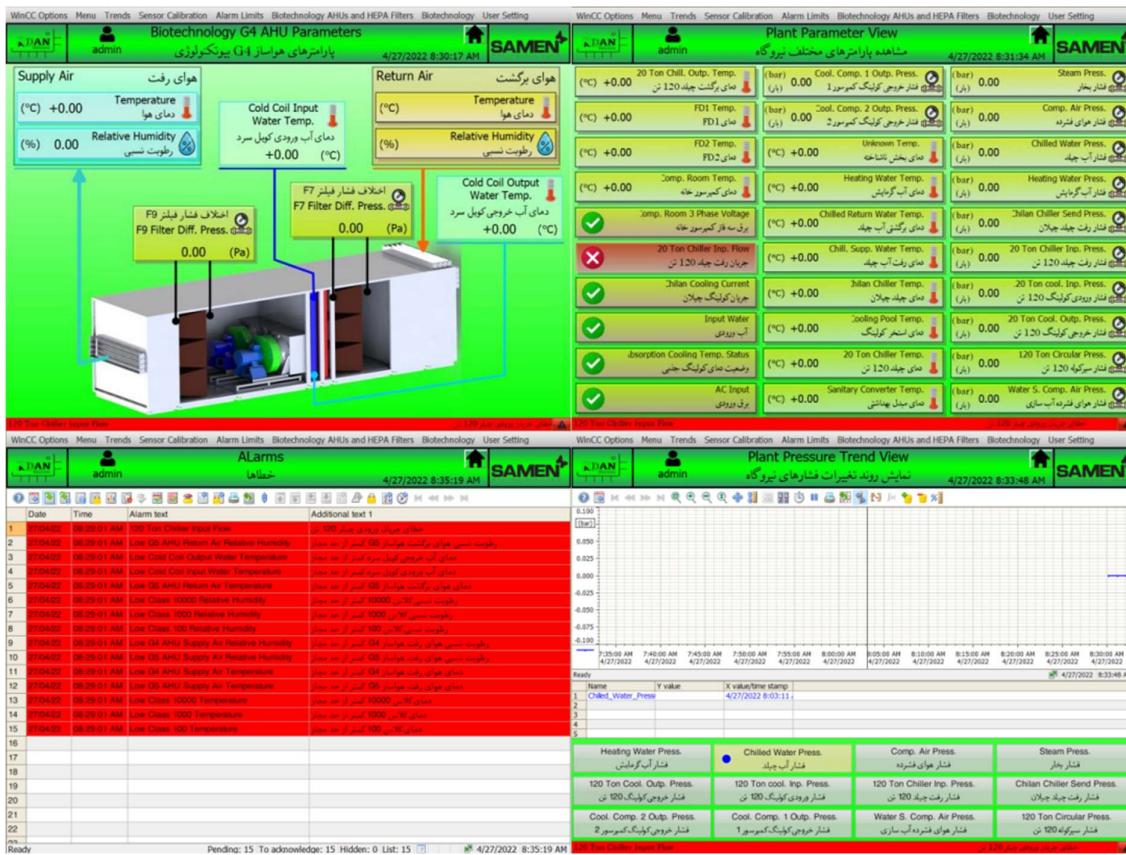
کنترل دمای تیغه جوش، دریافت فرمانی از طریق Modbus یا ورودی آنالوگ، دارای قابلیت اتوکالیبراسیون

۹) سیستم کنترل و مانیتورینگ تاسیسات بیوتکنولوژی



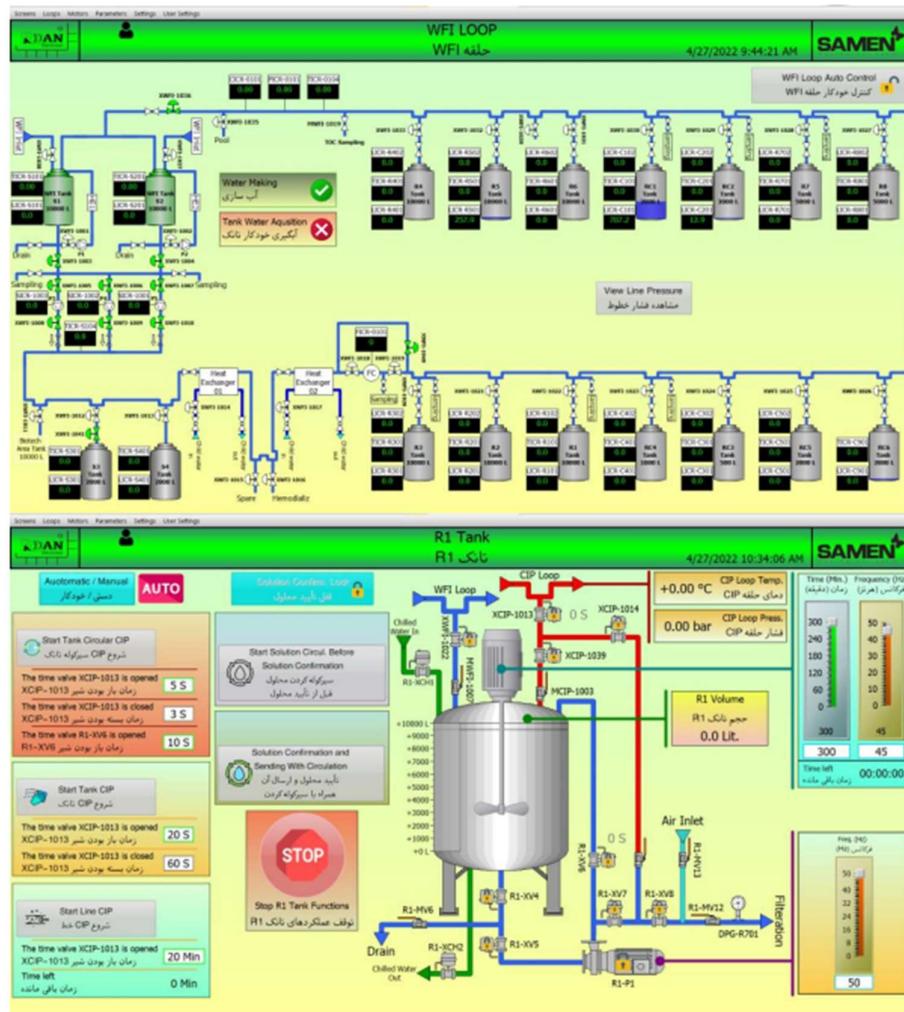
این پروژه با هدف اتوماسیون فرایندهای آبسازی و اندازه گیری پارامتر های تولید بخش بیوتکنولوژی شرکت داروسازی ثامن تعریف شده است. در این پروژه با ساخت تابلو برق کنترلی شامل HMI, PLC، شیرهای برقی و درایوهای الکتریکی، فرایندهای مربوط به تاسیسات این بخش شامل فرمان شیرهای پنوماتیک و پمپ ها کنترل می شود. سنسورهای موجود در مسیر خط، بازخوردهای مناسب را برای سیستم کنترلی ارسال کرده و این اطلاعات بر روی HMI نمایش داده میشود. همچنین برای بررسی های آینده اطلاعات ذخیره میشود.

۱۰) سیستم SCADA نیروگاه و هواسازها



این سیستم مانیتورینگ، با جمع اوری اطلاعات ۴ تابلو PLC و دو عدد مودم S7-Communication ساخت شرکت، کلیه پارامتر های هواسازها، چیلر و کمپرسورها و... را از سراسر کارخانه روی سرور ذخیره کرده و نمایش می دهد. این پارامترها شامل فشارها، دماها، رطوبت نسبی و وضعیت سطح ولتاژ دستگاهها می باشد. افراد مختلف با سطح دسترسی های متفاوت با اتصال به این سیستم از طریق WebViewer امکان مشاهده همه ویا تعدادی از پارامترها را خواهند داشت و می توانند تعدادی از تنظیمات سیستم را تغییر دهند. همچنین با تعریف محدوده مجاز برای هر پارامتر، سیستم مانیتورینگ خطاهای سمعی و بصری مناسب را در صورت تجاوز از محدوده مجاز، به اپراتور اعلام می کند تا در اسرع وقت مشکل پیش آمده برطرف شود.

(11) سیستم کنترل و مانیتورینگ سالن ساخت محلول



این سیستم SCADA شامل پایش و کنترل کامل فرایند ساخت محلول در شرکت داروسازی ثامن مشهد میباشد. سیستم مذکور با ارسال فرامین مناسب به ۱۲۸ شیر برقی و ۲۶ درایو الکتریکی، بهره برداری از تانک ها، مبدل های حرارتی، پمپ ها، میکسر ها و ... را به صورت اتوماتیک کنترل می نماید. همچنین اطلاعات مربوط به حجم تانک ها، دماها، فشار خطوط، هدايت الکتریکی آب و فلومتر کولیوریس را جمع آوری و پایش می نماید. به مژول های شیر برقی Festo تحت شبکه Profibus Vacon فرمان کنترلی ارسال می شود. همچنین فعالیت ها و فرامین ارسال شده توسط اپراتور ها و نیز کلیه پارامتر های اندازه گیری شده جهت بررسی های اینده و کاهش خطای انسانی در بهره برداری از سیستم ذخیره می شود.

خدمات و تجهیزات

واحد تحقیق و توسعه رادان الکترونیک تجهیزات مناسب و کاربردی در زمینه تست و طراحی سیستم هالی الکترونیکی و کنترلی دارد. برخی از این دستگاه‌ها عبارتند از:

دستگاه هویه و هیتر: از این دستگاه به منظور هویه کاری قطعات الکتریکی روی برد استفاده می‌شود. همچنین به کمک هیتر این دستگاه می‌توان قطعات ریز SMD را به کمک حرارت از برد جدا نمود. استفاده از این دستگاه در تمامی شرکت‌هایی که با قطعات الکتریکی سر و کار دارند ضروری به نظر می‌رسد.

اسیلوسکوپ دیجیتال: از این دستگاه به منظور مشاهده سیگنال‌های مورد نظر استفاده می‌شود.

دستگاه اسپات ولدر: از این دستگاه به منظور جوش نقطه‌ای استفاده می‌شود.

دستگاه وات متر: این دستگاه مقادیر پارامترهای توان، جریان، ولتاژ، ضریب قدرت و فرکانس امپدانس تحت آزمایش را اندازه گیری کرده و نمایش می‌دهد.

واریابل: به کمک این دستگاه برق شهر به ولتاژهای تا ۲۵۰ ولت متناوب تبدیل می‌شود.

مبدل بوستر: این دستگاه مبدل قابلیت تبدیل ولتاژ ورودی به ولتاژهای بالاتر را دارا بوده و محدوده بزرگی از ولتاژ و جریان را در خروجی تحويل می‌دهد

مهارت‌های کسب شده

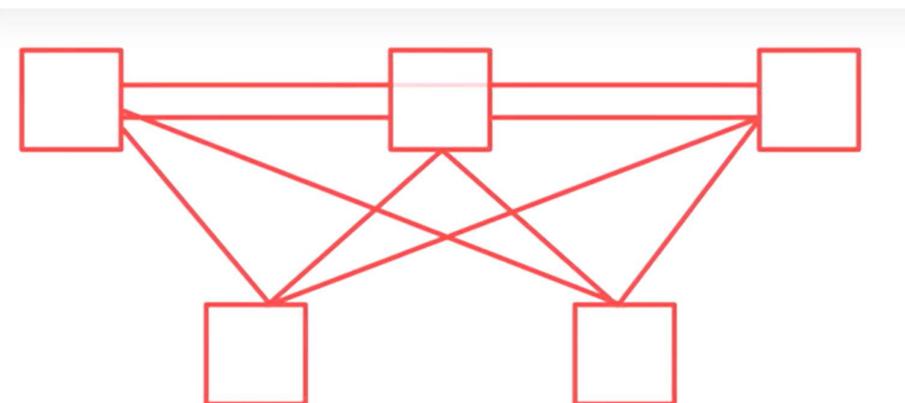
بنده در طی این مدت در بخش تحقیق و توسعه (R&D) تجربیات زیادی کسب کردم و در پروژه‌های متعدد فعال شرکت همکاری کردم که بخشی از آنها به صورت زیر است:

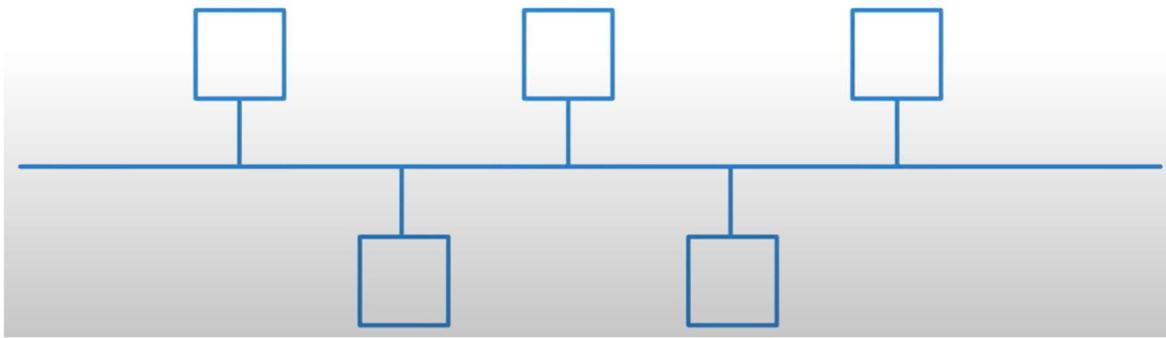
۱- تست کیفیت بورد: بورد های مونتاژ شده با کمک مولتی متر و یا مدار تست برداشت میشوند تا از صحت کارکرد انها اطمینان حاصل شود

۲- یکی از پروژه‌های فعال شرکت که بنده در بد و ورود دوره کارآموزی خود را با آن شروع کردم پروژه ارتباط دیوایس‌های خارجی با پروتوكل CAN با خودرو بود که به درخواست شرکت ایرانخودرو شرق کشور ثبت سفارش شده بود. در ابتدا بنده اطلاعات کمی درمورد این پروتوكل داشتم به همین دلیل، مدتی را مشغول رسیرج درمورد این پروتوكل و نحوه ارتباط و ساخت افزار آن بودم که بخشی از اطلاعات بدست امده من به صورت زیر است:

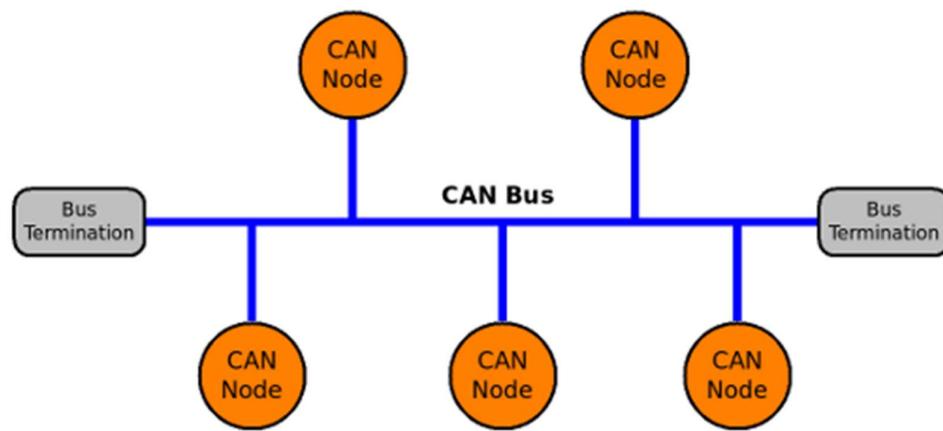
پروتکل کن (Controller Area Network)

- اولین بار در صنعت خودرو سازی به منظور کاهش سیم کشی و ارتباط راحت تر بین سنسور‌ها ایجاد شد.
- به جای ارتباط نقطه به نقطه بین ECU‌های خودرو، یک بس مشترک درنظر گرفته شد که تمام ECU‌ها میتوانند در آن بس دیتا ارسال کنند و دیتا مورد نظر خود را از همان طریق دریافت کنند.

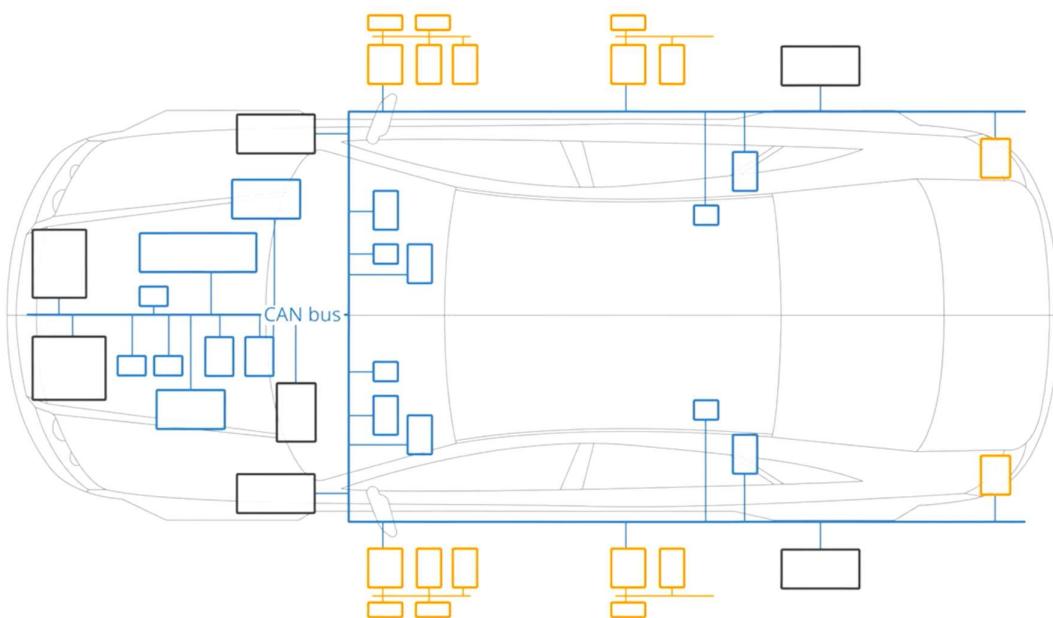
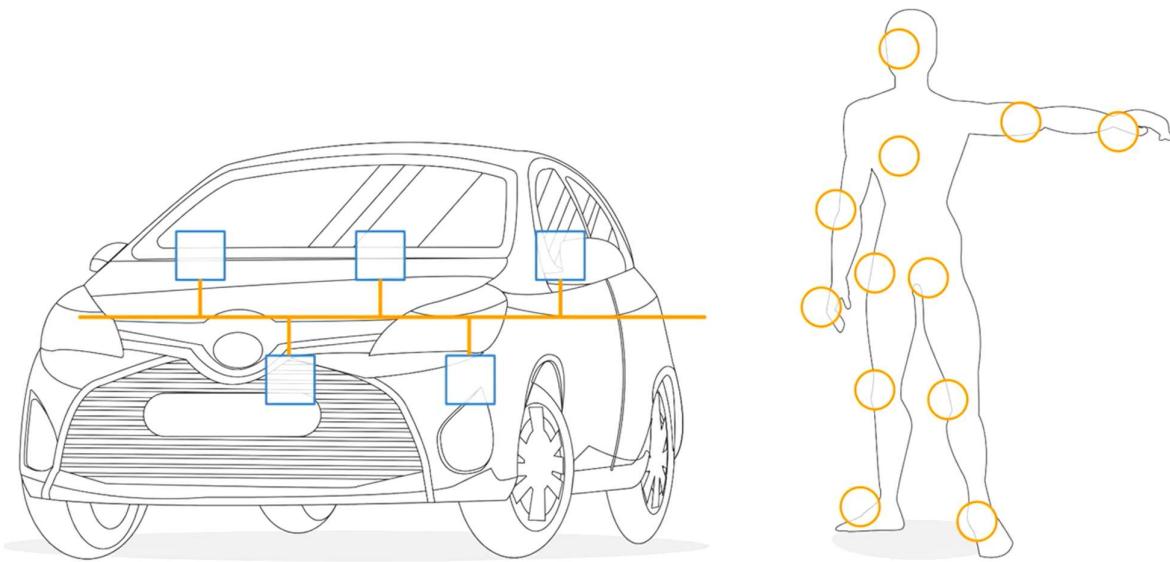




- با فاصله گرفتن از روش ارتباط نقطه به نقطه حجم انبوهی از سیم کشی ها کاهش یافت.
- پروتوكل کن از یک BUS و NODE های متصل به باس تشکیل شده است.



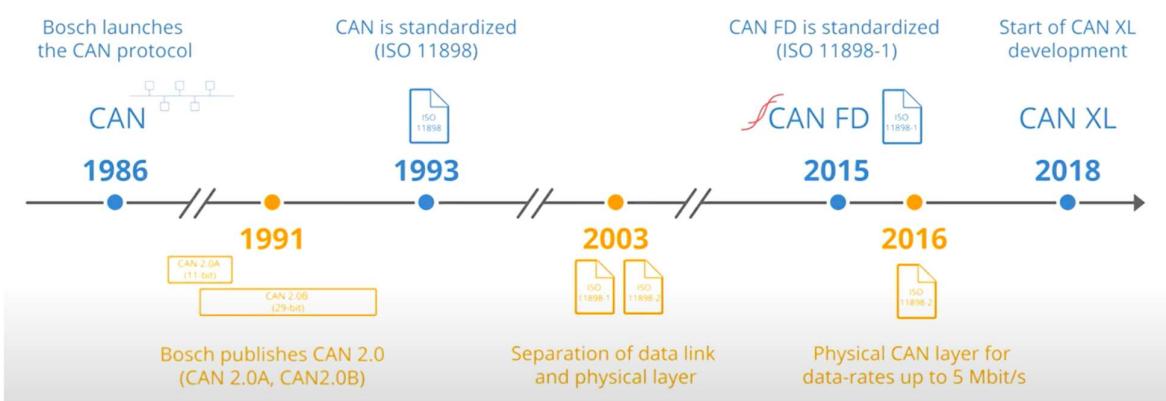
- برای درک بهتر می توان ماشین را مانند بدن انسان درنظر گرفت و سیستم عصبی بدن انسان را مشابه به can bus فرض کرد که واسط ارتباطی انتقال پیام بین اعضای مختلف بدن(ecu ها در ماشین) است.



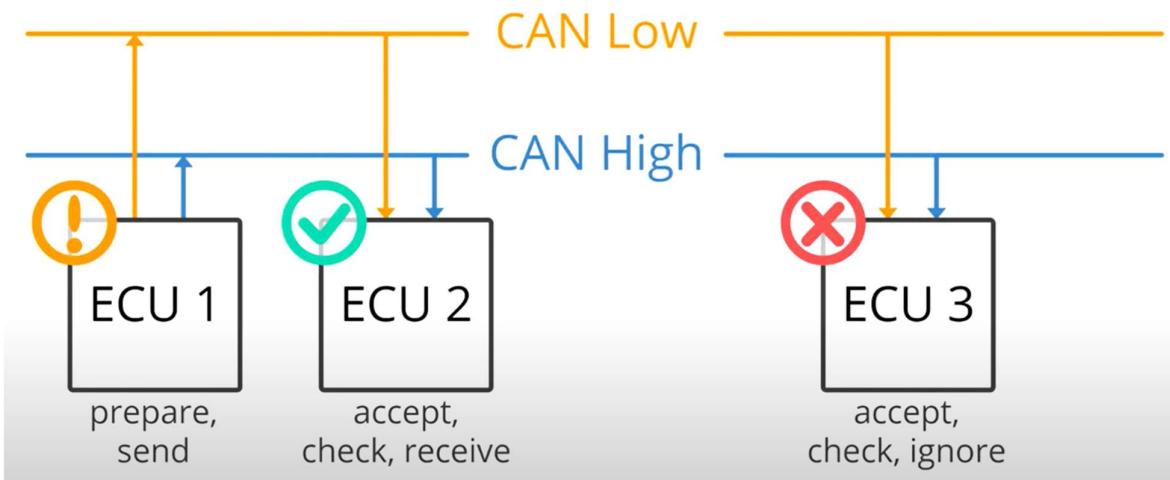
Ecus متصل شده به can bus خودرو

- در خودرو های امروزی به دلیل حجم زیاد دیتای قابل انتقال با اولویت های کم و زیاد، الزاماً یک باس وجود ندارد، و حداقل ۲ باس وجود دارد. یکی برای node های با اولویت پایین تر و دیگری برای node هایی با اولویت بالاتر.

- سیر تکاملی پروتوكل کن به صورت زیر است:



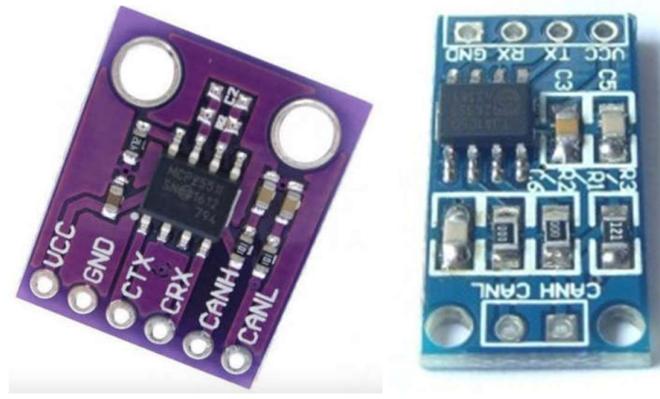
- حال فرض کنید این ارتباط قرار بود به صورت نقطه به نقطه انجام شود!
- برخلاف سایر ارتباط های موجود (I2C، SPI)، به صورت master&slave کار نمیکند و تمام node های متصل به can bus میتوانند به صورت همزمان دیتا ارسال و دریافت کنند.
- ویژگی ای که ارتباط کن را از سایرین متمایز میکند نرخ بالای انتقال دیتا در فواصل دور و قابلیت اولویت بندی دیتا است، یعنی میتوان دیتا های ارسالی از node های مختلف را فیلتر کرد (بر حسب ادرس دیتا) به طوری که هر node، دیتا مربوط به خود را دریافت کند و سایر دیتا های موجود در باس، در عملکرد ارتباط مشکلی ایجاد نکند.



- اگر یکی از node های متصل به بس، به هر دلیلی دچار مشکل شود و عملکردن مختل شود، این امر باعث مختل شدن کامل بس نمی شود و این node توسط لایه فیزیکی از bus جدا می شود.
- کن، ارتباطی آسنکرون و دیفرانسیلی است. (CAN-High و CAN-Low)

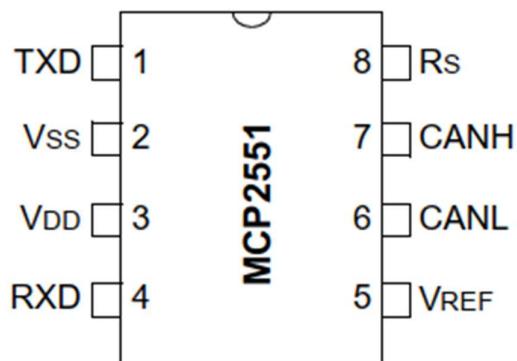


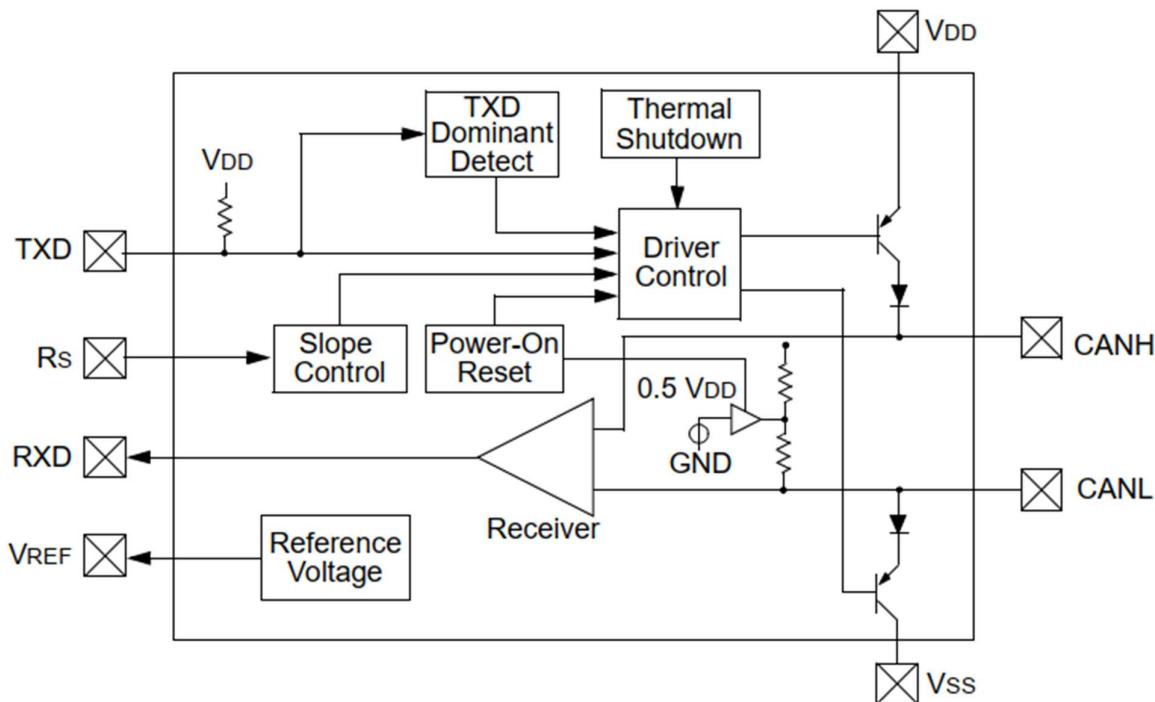
- در میکرو های ST برای ایجاد bus مشترک و خروجی های CAN-H و CAN-L باید از ایسی های TJA1050, MCP2551 استفاده کرد.



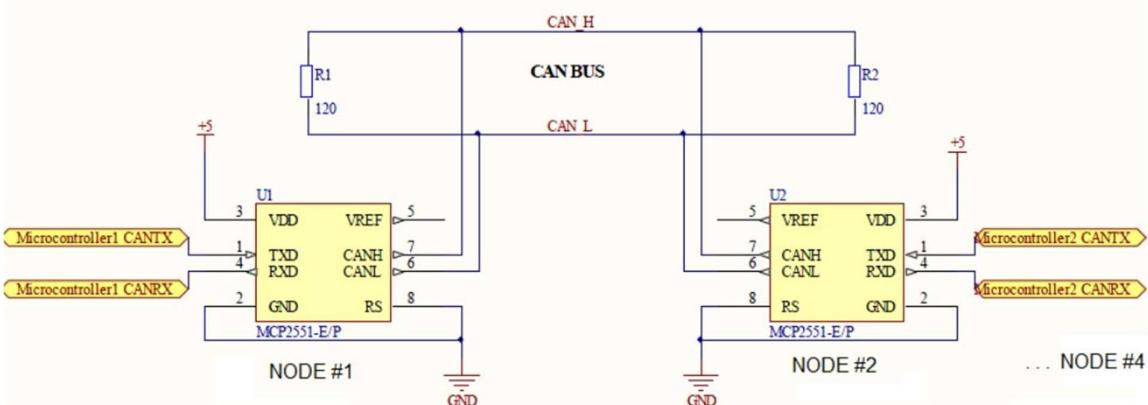
- پکیج و بلوک دیاگرام MCP2551 به صورت زیر است:

PDIP/SOIC

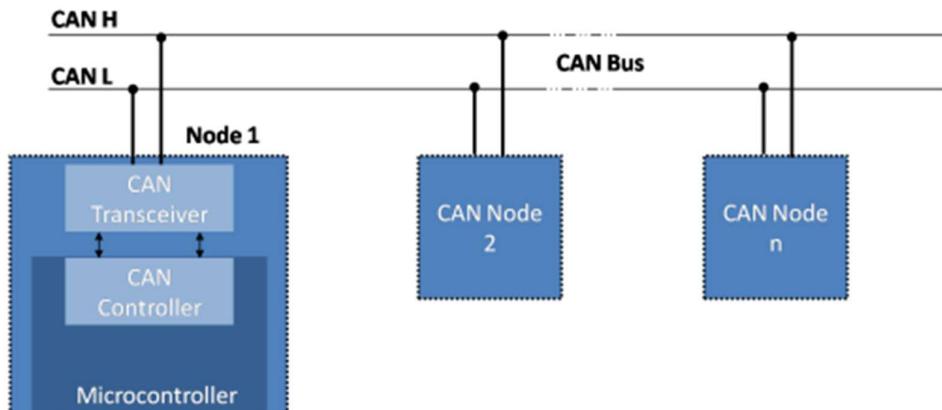




- مدار پیشنهادی در صورت استفاده از ایسی تک:

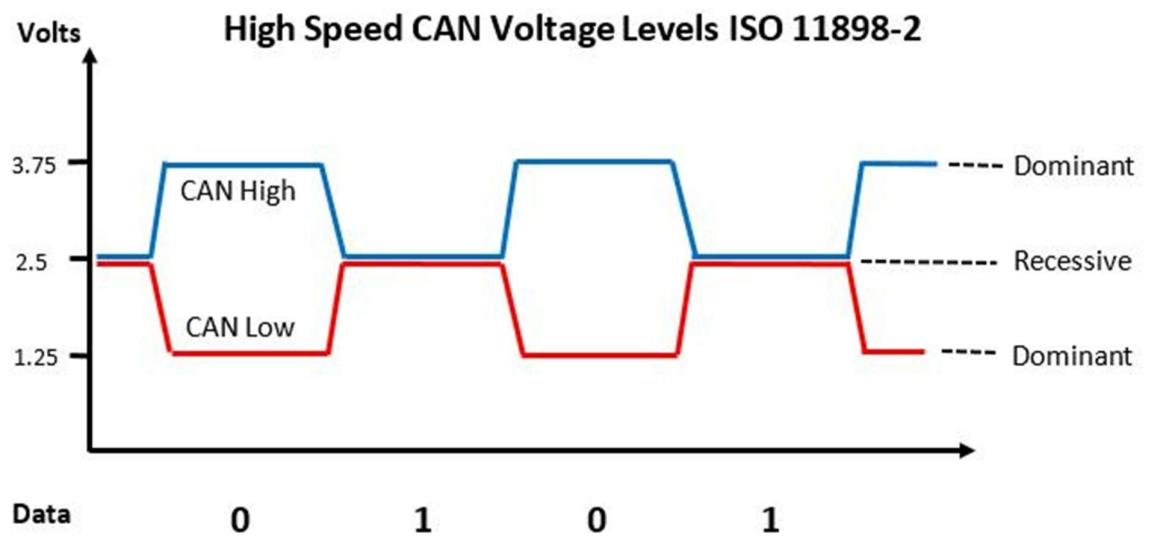


- در خودرو، can bus موجود است، بنابر این می توان با استفاده از هر ایسی به عنوان یک node به bus متصل شد.



Architecture of CAN bus

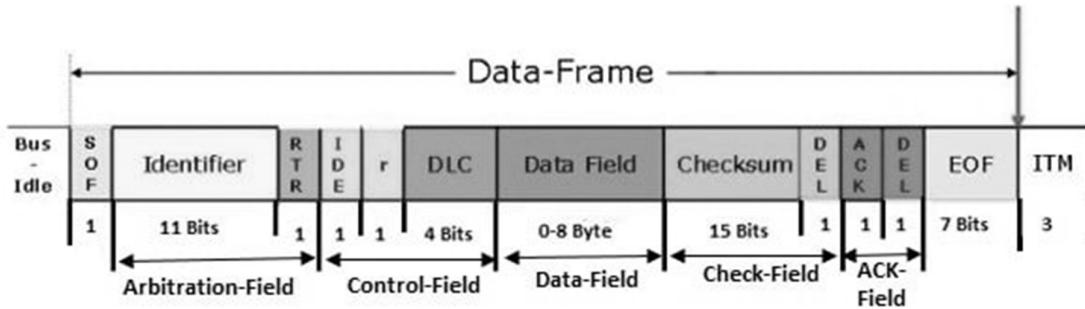
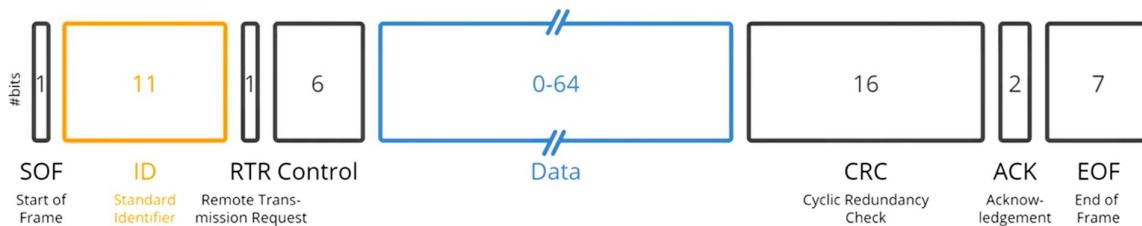
- سطوح منطقی کن به صورت زیر است:



اگر CAN-H و CAN-L هردو ۲.۵ ولت باشند، منطق خروجی HIGH است.

اگر CAN-H، CAN-L ۳.۷۵ و ۱.۲۵ باشد، منطق خروجی LOW است.

- قالب استاندارد(فریم) هر پیام در پروتوكل کن به صورت زیر است:



قسمت ۱: **SOF** (Start of Frame) شروع فریم است که وظیفه ان اطلاع دادن به node ها برای انتقال دیتا است. غالبا صفر میباشد.

قسمت ۲: **ID** (Standard ID): آدرس هر فریم در این قسمت با یک عدد ۱۱ بیتی مشخص می شود. فریم هایی با ارزش پایین تر دارای اولویت ارسال بالاتری هستند.

قسمت ۳: **RTR** (Remote Transmission Request): وظیفه ان این است که مشخص میکند یک node ارسال کننده دیتابست یا دریافت کننده.

قسمت ۴: **Control**: این قسمت به ما اطلاع می دهد که CAN ID از قالب ۱۱ بیتی استفاده میکنده یا از ۲۹ بیتی و با یک عدد ۲ بیتی مشخص می شود. ۴ بیت باقی مانده مربوط میشود به DLC که وظیفه ان مشخص کردن طول دیتا به بایت است(حداکثر طول دیتا در این پروتکل ۸ بایت) است.

قسمت ۵: **Data**: پیام ارسالی در این قسمت قرار میگیرد

قسمت ۶: **CRC** (Cyclic Redundancy Check): برای چک کردن صحت از دیتابس ارسالی مورد استفاده قرار میگیرد

قسمت ۷: **ACK** (Acknowledgement) نشان میدهد که ایا node دیتا را به درستی دریافت کرده است یا خیر.

قسمت ۸: **EOF** (End of Frame) انتهای یک فریم را مشخص میکند.

- نحوه تنظیم کردن دیتا فریم برای میکرو های خانواده ST:

```
CAN_TxHeaderTypeDef TxHeader;
uint8_t TxData[8];
uint32_t TxMailbox;

TxHeader.IDE = CAN_ID_STD;
TxHeader.StdId = 0x446;
TxHeader.RTR = CAN_RTR_DATA;
TxHeader.DLC = 2;
TxData[0] = 50;
TxData[1] = 0xAA;
```

سایر قسمت ها توسط توابع HAL، هندل میشود

بنده برنامه زیر را به منظور تست ارتباط بین دو میکروکنترل با پروتکل CAN نوشتم که خروجی مطلوبی داشت

```
/* USER CODE END Header */
/* Includes ----- */
#include "main.h"

/* Private includes ----- */
/* USER CODE BEGIN Includes */

/* USER CODE END Includes */

/* Private typedef ----- */
/* USER CODE BEGIN PTD */

/* USER CODE END PTD */

/* Private define ----- */
/* USER CODE BEGIN PD */
/* USER CODE END PD */

/* Private macro ----- */
/* USER CODE BEGIN PM */

/* USER CODE END PM */
```

```

/* Private variables -----
CAN_HandleTypeDef hcan1;

UART_HandleTypeDef huart2;

/* USER CODE BEGIN PV */

/* USER CODE END PV */

/* Private function prototypes -----
void SystemClock_Config(void);
static void MX_GPIO_Init(void);
static void MX_USART2_UART_Init(void);
static void MX_CAN1_Init(void);
/* USER CODE BEGIN PFP */

/* USER CODE END PFP */

/* Private user code -----
/* USER CODE BEGIN 0 */
CAN_TxHeaderTypeDef TxHeader;
CAN_RxHeaderTypeDef RxHeader;
uint8_t TxData[8];
uint8_t RxData[8];
uint32_t TxMailbox;

int dataCheck = 0;

void HAL_GPIO_EXTI_Callback(uint16_t GPIO_Pin)
{
    if(GPIO_Pin == GPIO_PIN_13)
    {
        TxData[0] = 100; //delay in ms
        TxData[1] = 10; //number of repeat

        HAL_CAN_AddTxMessage(&hcan1, &TxHeader, TxData, &TxMailbox);
    }
}

void HAL_CAN_RxFifo0MsgPendingCallback(CAN_HandleTypeDef *hcan)
{
    HAL_CAN_GetRxMessage(&hcan1, CAN_RX_FIFO0, &RxHeader, RxData);
    if(RxHeader.DLC == 2)
        dataCheck = 1;
}
/* USER CODE END 0 */

/**
 * @brief The application entry point.
 * @retval int
 */
int main(void)
{
    /* USER CODE BEGIN 1 */

    /* USER CODE END 1 */

    /* MCU Configuration-----*/
}

```

```

/* Reset of all peripherals, Initializes the Flash interface and the Systick. */
HAL_Init();

/* USER CODE BEGIN Init */

/* USER CODE END Init */

/* Configure the system clock */
SystemClock_Config();

/* USER CODE BEGIN SysInit */

/* USER CODE END SysInit */

/* Initialize all configured peripherals */
MX_GPIO_Init();
MX_USART2_UART_Init();
MX_CAN1_Init();
/* USER CODE BEGIN 2 */

HAL_CAN_Start(&hcan1); //start can
HAL_CAN_ActivateNotification(&hcan1, CAN_IT_RX_FIF00_MSG_PENDING); //activate
notification

TxHeader.DLC = 2; //data length
TxHeader.IDE = CAN_ID_STD;
TxHeader.RTR = CAN_RTR_DATA;
TxHeader.StdId = 0x446; //standard id

/* USER CODE END 2 */

/* Infinite loop */
/* USER CODE BEGIN WHILE */
while (1)
{
    if(dataCheck == 1)
    {
        for(int i = 0; i < RxData[1]; i++)
        {
            HAL_GPIO_TogglePin(GPIOA, GPIO_PIN_5);
            HAL_Delay(RxData[0]);
        }
        dataCheck = 0;
    }

    //HAL_GPIO_TogglePin(GPIOA, GPIO_PIN_5);
    //HAL_Delay(500);
    /* USER CODE END WHILE */

    /* USER CODE BEGIN 3 */
}
/* USER CODE END 3 */
}

/**
 * @brief System Clock Configuration
 * @retval None

```

```

/*
void SystemClock_Config(void)
{
    RCC_OscInitTypeDef RCC_OscInitStruct = {0};
    RCC_ClkInitTypeDef RCC_ClkInitStruct = {0};

    /** Configure the main internal regulator output voltage
    */
    __HAL_RCC_PWR_CLK_ENABLE();
    __HAL_PWR_VOLTAGESCALING_CONFIG(PWR_REGULATOR_VOLTAGE_SCALE1);
    /** Initializes the RCC Oscillators according to the specified parameters
     * in the RCC_OscInitTypeDef structure.
    */
    RCC_OscInitStruct.OscillatorType = RCC_OSCILLATORTYPE_HSE;
    RCC_OscInitStruct.HSEState = RCC_HSE_ON;
    RCC_OscInitStruct.PLL.PLLState = RCC_PLL_ON;
    RCC_OscInitStruct.PLL.PLLSource = RCC_PLLSOURCE_HSE;
    RCC_OscInitStruct.PLL.PLLM = 4;
    RCC_OscInitStruct.PLL.PLLN = 180;
    RCC_OscInitStruct.PLL.PLLP = RCC_PLLP_DIV2;
    RCC_OscInitStruct.PLL.PLLQ = 2;
    RCC_OscInitStruct.PLL.PLLR = 2;
    if (HAL_RCC_OscConfig(&RCC_OscInitStruct) != HAL_OK)
    {
        Error_Handler();
    }
    /** Activate the Over-Drive mode
    */
    if (HAL_PWREx_EnableOverDrive() != HAL_OK)
    {
        Error_Handler();
    }
    /** Initializes the CPU, AHB and APB buses clocks
    */
    RCC_ClkInitStruct.ClockType = RCC_CLOCKTYPE_HCLK|RCC_CLOCKTYPE_SYSCLK
                                |RCC_CLOCKTYPE_PCLK1|RCC_CLOCKTYPE_PCLK2;
    RCC_ClkInitStruct.SYSCLKSource = RCC_SYSCLKSOURCE_PLLCLK;
    RCC_ClkInitStruct.AHBClockDivider = RCC_SYSCLK_DIV1;
    RCC_ClkInitStruct.APB1ClockDivider = RCC_HCLK_DIV4;
    RCC_ClkInitStruct.APB2ClockDivider = RCC_HCLK_DIV2;

    if (HAL_RCC_ClockConfig(&RCC_ClkInitStruct, FLASH_LATENCY_5) != HAL_OK)
    {
        Error_Handler();
    }

    /**
     * @brief CAN1 Initialization Function
     * @param None
     * @retval None
     */
    static void MX_CAN1_Init(void)
    {

        /* USER CODE BEGIN CAN1_Init_0 */

        /* USER CODE END CAN1_Init_0 */
}

```

```

/* USER CODE BEGIN CAN1_Init_1 */

/* USER CODE END CAN1_Init_1 */
hcan1.Instance = CAN1;
hcan1.Init.Prescaler = 18;
hcan1.Init.Mode = CAN_MODE_NORMAL;
hcan1.Init.SyncJumpWidth = CAN SJW_1TQ;
hcan1.Init.TimeSeg1 = CAN_BS1_2TQ;
hcan1.Init.TimeSeg2 = CAN_BS2_2TQ;
hcan1.Init.TimeTriggeredMode = DISABLE;
hcan1.Init.AutoBusOff = DISABLE;
hcan1.Init.AutoWakeUp = DISABLE;
hcan1.Init.AutoRetransmission = DISABLE;
hcan1.Init.ReceiveFifoLocked = DISABLE;
hcan1.Init.TransmitFifoPriority = DISABLE;
if (HAL_CAN_Init(&hcan1) != HAL_OK)
{
    Error_Handler();
}
/* USER CODE BEGIN CAN1_Init_2 */

CAN_FilterTypeDef canfilterconfig;

canfilterconfig.FilterActivation = CAN_FILTER_ENABLE;
canfilterconfig.SlaveStartFilterBank = 18; // which filter bank to use from
the assigned ones
canfilterconfig.FilterFIFOAssignment = CAN_FILTER_FIFO0;
canfilterconfig.FilterIdHigh = 0x103<<5;
canfilterconfig.FilterIdLow = 0;
canfilterconfig.FilterMaskIdHigh = 0x100<<5;
canfilterconfig.FilterMaskIdLow = 0x0000;
canfilterconfig.FilterMode = CAN_FILTERMODE_IDMASK;
canfilterconfig.FilterScale = CAN_FILTERSCALE_32BIT;
canfilterconfig.SlaveStartFilterBank = 20; // how many filters to assign to the
CAN1 (master can)

HAL_CAN_ConfigFilter(&hcan1, &canfilterconfig);

/* USER CODE END CAN1_Init_2 */

}

/**
 * @brief USART2 Initialization Function
 * @param None
 * @retval None
 */
static void MX_USART2_UART_Init(void)
{

/* USER CODE BEGIN USART2_Init_0 */

/* USER CODE END USART2_Init_0 */
/* USER CODE BEGIN USART2_Init_1 */

/* USER CODE END USART2_Init_1 */
huart2.Instance = USART2;
huart2.Init.BaudRate = 115200;

```

```

        huart2.Init.WordLength = UART_WORDLENGTH_8B;
        huart2.Init.StopBits = UART_STOPBITS_1;
        huart2.Init.Parity = UART_PARITY_NONE;
        huart2.Init.Mode = UART_MODE_TX_RX;
        huart2.Init.HwFlowCtl = UART_HWCONTROL_NONE;
        huart2.Init.OverSampling = UART_OVERSAMPLING_16;
        if (HAL_UART_Init(&huart2) != HAL_OK)
        {
            Error_Handler();
        }
        /* USER CODE BEGIN USART2_Init_2 */

        /* USER CODE END USART2_Init_2 */

    }

    /**
     * @brief GPIO Initialization Function
     * @param None
     * @retval None
     */
    static void MX_GPIO_Init(void)
{
    GPIO_InitTypeDef GPIO_InitStruct = {0};

    /* GPIO Ports Clock Enable */
    __HAL_RCC_GPIOC_CLK_ENABLE();
    __HAL_RCC_GPIOH_CLK_ENABLE();
    __HAL_RCC_GPIOA_CLK_ENABLE();

    /*Configure GPIO pin Output Level */
    HAL_GPIO_WritePin(LD2_GPIO_Port, LD2_Pin, GPIO_PIN_RESET);

    /*Configure GPIO pin : PC13 */
    GPIO_InitStruct.Pin = GPIO_PIN_13;
    GPIO_InitStruct.Mode = GPIO_MODE_IT_RISING;
    GPIO_InitStruct.Pull = GPIO_NOPULL;
    HAL_GPIO_Init(GPIOC, &GPIO_InitStruct);

    /*Configure GPIO pin : LD2_Pin */
    GPIO_InitStruct.Pin = LD2_Pin;
    GPIO_InitStruct.Mode = GPIO_MODE_OUTPUT_PP;
    GPIO_InitStruct.Pull = GPIO_NOPULL;
    GPIO_InitStruct.Speed = GPIO_SPEED_FREQ_LOW;
    HAL_GPIO_Init(LD2_GPIO_Port, &GPIO_InitStruct);

    /* EXTI interrupt init*/
    HAL_NVIC_SetPriority(EXTI15_10_IRQn, 0, 0);
    HAL_NVIC_EnableIRQ(EXTI15_10_IRQn);

}

/* USER CODE BEGIN 4 */

/* USER CODE END 4 */

    /**
     * @brief This function is executed in case of error occurrence.
     * @retval None
     */

```

```

/*
void Error_Handler(void)
{
    /* USER CODE BEGIN Error_Handler_Debug */
    /* User can add his own implementation to report the HAL error return state */
    _disable_irq();
    while (1)
    {
    }
    /* USER CODE END Error_Handler_Debug */
}

#ifndef USE_FULL_ASSERT
/**
 * @brief Reports the name of the source file and the source line number
 * where the assert_param error has occurred.
 * @param file: pointer to the source file name
 * @param line: assert_param error line source number
 * @retval None
 */
void assert_failed(uint8_t *file, uint32_t line)
{
    /* USER CODE BEGIN 6 */
    /* User can add his own implementation to report the file name and line number,
     * ex: printf("Wrong parameters value: file %s on line %d\r\n", file, line) */
    /* USER CODE END 6 */
}
#endif /* USE_FULL_ASSERT */

```

۳- پروژه دومی که در آن همکاری داشتم، پروژه تستر دسته سیم های خودرو بود که این پروژه هم سفارش شرکت ایرانخودرو شرق کشور بود. توضیحات کلی پروژه به این صورت است که ما به دستگاه و بستری نیاز داشتیم که چندین پورت ورودی ۱۰۲۴ پینی داشته باشد و بتواند به صورت همزمان عملیات تست صحت عملکرد دسته سیم ها، مقاومت و خاصیت خازنی سیم ها را اندازه گیری کند و در یک سیستم مانیتورینگ نشان دهد. وظیفه بنده در این پروژه تهیه مشخصات فنی و امکانات و تجهیزات جانبی نمونه های خارجی این دستگاه بود که بخشی از آن به صورت زیر خدمت شما ارائه می شود: (این پروژه به اتمام نرسیده و همچنان در حال طراحی و ساخت است)

Cable Tester

کمپانی های تولید کننده: 1) WEETECH و 2) CIRRIS

دسته بندی محصولات شرکت WEETECH :

- 1) WK 260 MU
- 2) WK 260 PC

- 3) WK 260 DC
- 4) W 434
- 5) W 434 PRO
- 6) W444
- 7) W484/W 484 PLUS
- 8) W454
- 9) W454 HV/W 454 SHV

<https://www.weetech-usa.com/products/wk-260-mu> :WK 260 MU (۱



نحوه عملکرد دستگاه:

<https://fhi.nl/profiel/weetech-bv>

<https://www.youtube.com/watch?v=bVx9CxR4DRs>

<https://www.youtube.com/watch?v=n4KPNUeYMDQ>

<https://www.youtube.com/watch?v=BFOWIdsCyS0>

https://www.youtube.com/watch?v=uFTZ-tu_Gxo

ویژگی ها:

- ولتاژ کاری دستگاه: ۱۳۵ الی ۳۷۰ ولت dc یا ۹۰ الی ۲۶۴ ولت ac، ولتاژ خروجی: ۲۴ ولت dc با جریان ۶۲۵ mA
- ال سی دی گرافیکی تک رنگ (۱۹۲*۶۴)
- رابط ethernet برای اتصال به کامپیوتر یا شبکه
- ۲ رابط سریال و ۲ رابط پارالل
- رابط usb
- خروجی و ورودی های دردسترس کاربر: ۱۰ عدد ورودی ۰ تا ۲۵ ولت dc
- عدد خروجی ۸ open collector ولت dc ۲۵، ولت dc ۱۰۰، ولت dc ۲۵، ولت dc، ۱ آمپر
- ماکزیمم جریان تست (قابل تغییر) ۱۵۰ میلی آمپر
- اندازه گیری ولتاژ های خارجی تا ۲۴ ولت dc
- اندازه گیری جریان های خارجی تا ۷۵ میلی آمپر

[/https://www.weetech-usa.com/products/wk-260-pc](https://www.weetech-usa.com/products/wk-260-pc) :MK 260 PC (۲



ویژگی ها:

- ولتاژ کاری دستگاه: ۱۳۵ الی ۳۷۰ ولت dc یا ۹۰ الی ۲۶۴ ولت ac، ولتاژ خروجی: ۲۴ ولت dc با جریان ۶۲۵ mA
- ال سی دی گرافیکی تک رنگ (۱۹۲*۶۴)
- رابط ethernet برای اتصال به کامپیوتر یا شبکه
- ۲ رابط سریال و ۲ رابط پارالل
- رابط usb
- خروجی و ورودی های دردسترس کاربر: ۱۰ عدد ورودی ۰ تا ۲۵ ولت dc
- عدد خروجی ۸ ولت dc، open collector
- ۱ عدد خروجی رله، ۲۵ ولت dc، ۱ آمپر
- ماکزیمم جریان تست (قابل تغییر) ۱۵۰ میلی آمپر
- اندازه گیری ولتاژ های خارجی تا ۲۴ ولت dc
- اندازه گیری جریان های خارجی تا ۷۵ میلی آمپر

[/https://www.weetech-usa.com/products/wk-260-dc](https://www.weetech-usa.com/products/wk-260-dc) :WK 260 DC (۳



دسته بندی محصولات شرکت CIRRIS

- 1) 4200
- 2) 4250
- 3) Easy Touch Pro

<https://www.cirris.com/products/testers/item/199-4200> (۱)



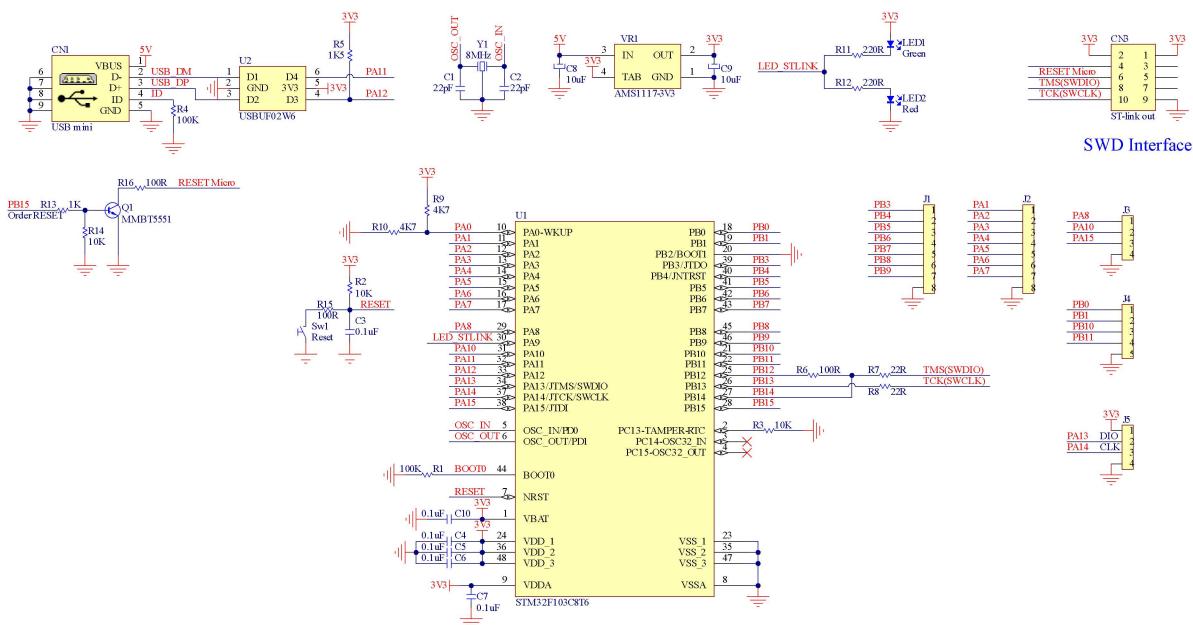
<https://www.cirris.com/products/testers/item/198-4250> (۲)

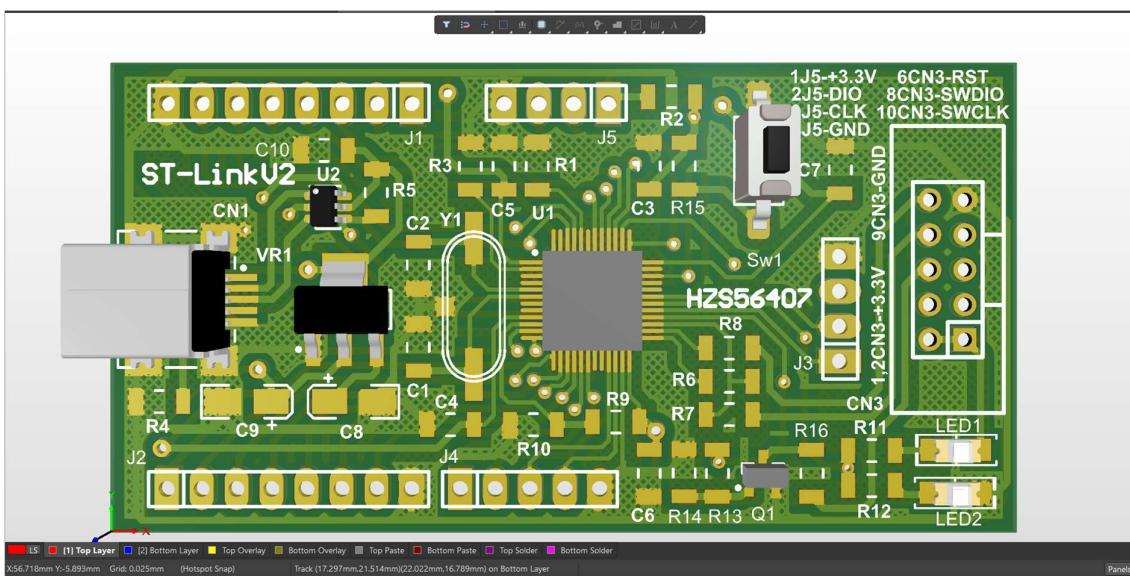
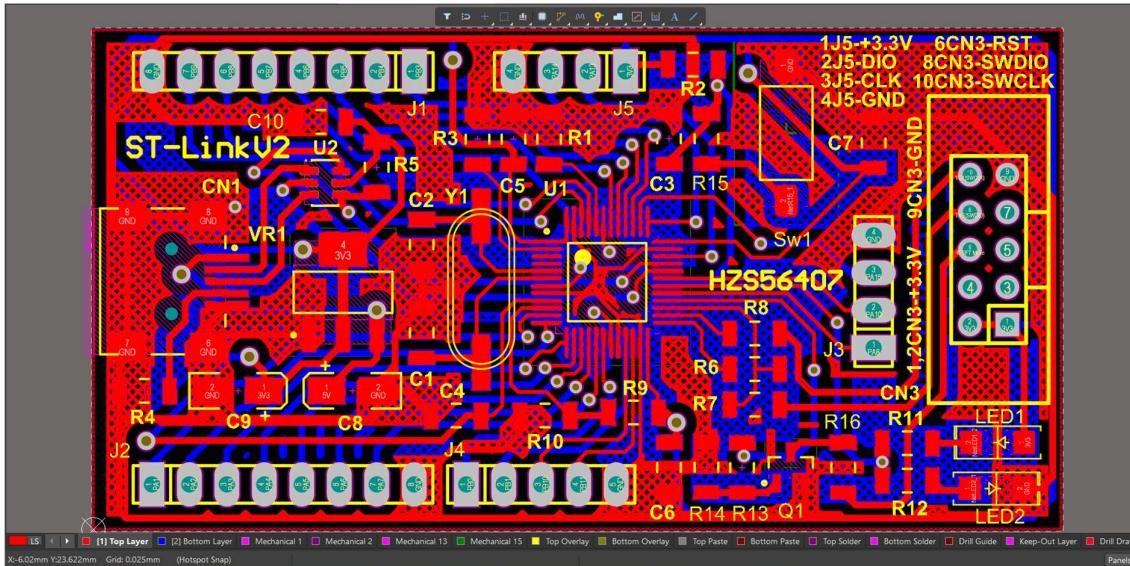


<https://www.cirris.com/products/testers/item/1- :Easy-Touch Pro> (۳)
[easy-touch](#)



۴- پروژه سوم عیب یابی پروگرامر های شرکت ST بود. متأسفانه به دلیل تحریم ها این پروگرامر ها به صورت اورجینال به کشور نمیرسد و مدل های فیک ان معتملا در بازار ایران موجود است. این امر باعث شده بود هنگام کار و دیباگ با این پروگرامر ها در ولتاژ و فرکانس های بالا از کار بیوفتد و از مد دیباگ خارج شود. پس از عیب یابی پروگرامر زیر توسط بنده به شرکت ارائه شد و کارهای ساخت آن در حال انجام است.





۵- پروژه بعدی که بنده در ان همکاری داشتم، یک وسیله اسباب بازی الکترونیکی (شمშیر) سفارش یک شرکت کانادایی بود. مشکلی که شرکت ما در ابتدای کار با ان مواجه شد آن بود که به دلیل لاینسنس گران قیمت نرم افزار التیوم در خارج از کشور و رواج بیشتر نرم افزار KiCAD در کشور های امریکای شمالی، شماتیک و PCB طرح باید در نرم افزار اوپن سورس KiCAD انجام شود و کسی به این نرم افزار تسلط نداشت. بندе مسئول این شدم که به صورت فوری در چند روز این نرم افزار را نصب و به آن تسلط پیدا کنم و به سایر اعضای دخیل در این پروژه اموزش دهم. در طی ۴ روز به این نرم افزار تسلط پیدا کردم و یک داکیومنت مناسب جهت توضیح و ارائه به سایرین تهیه کردم که بخش هایی از ان به صورت زیر است:

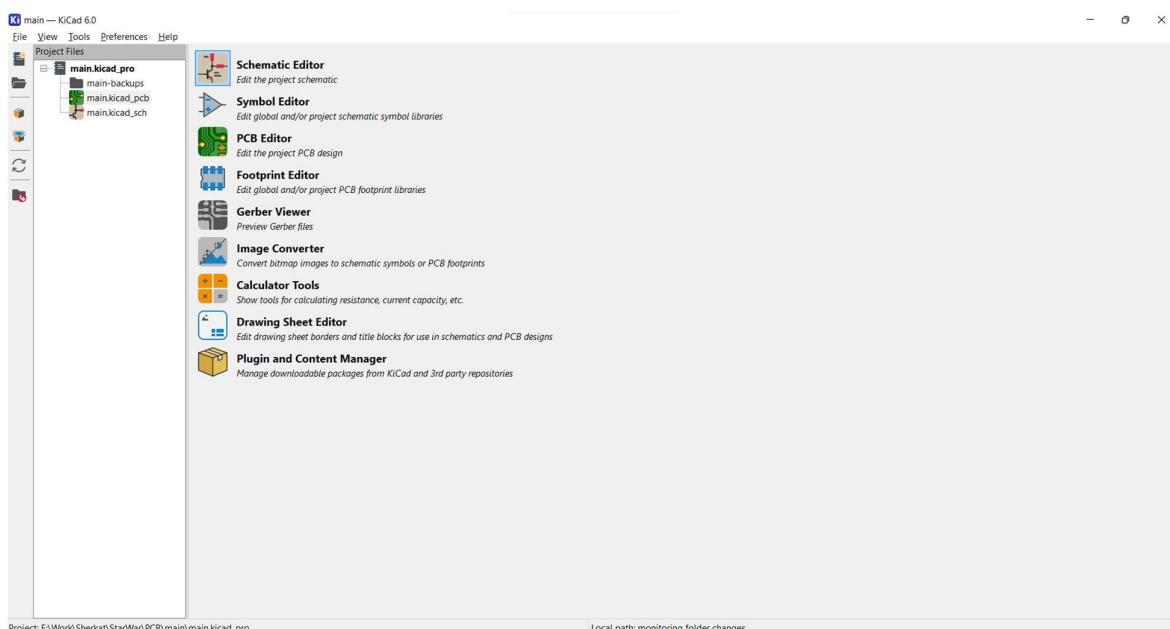
آموزش سریع : KiCAD

با توجه به سیستم عامل خود، نرم افزار را از این لینک دانلود کنید:

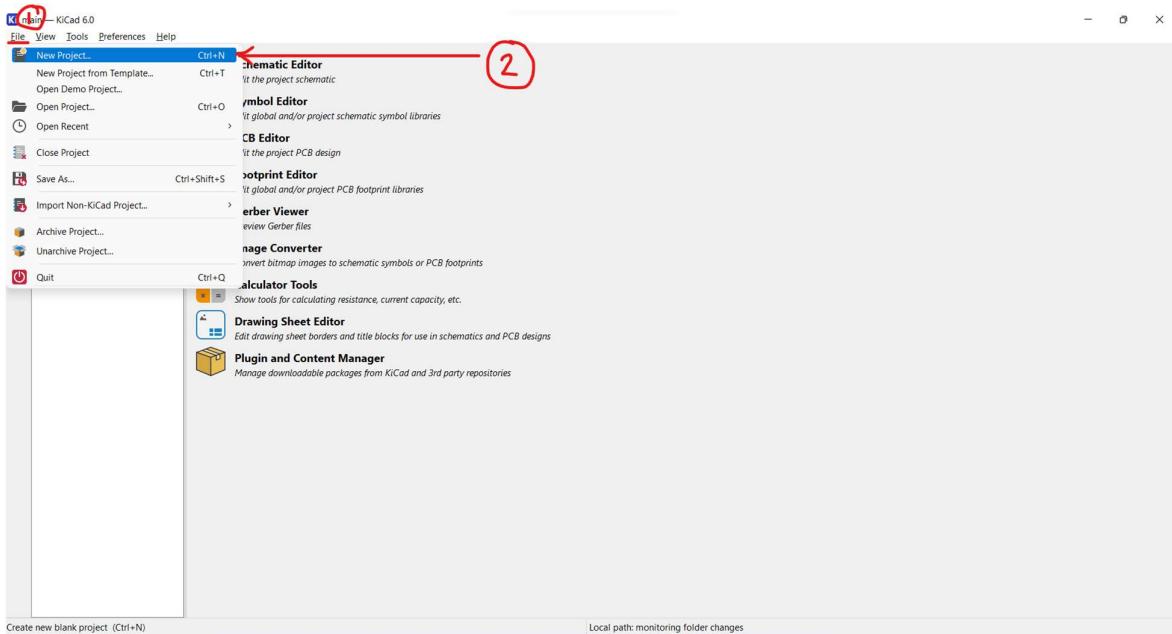
[/https://www.kicad.org/download](https://www.kicad.org/download)

نرم افزار را نصب میکنیم 😊

محیط نرم افزار به صورت زیر است:

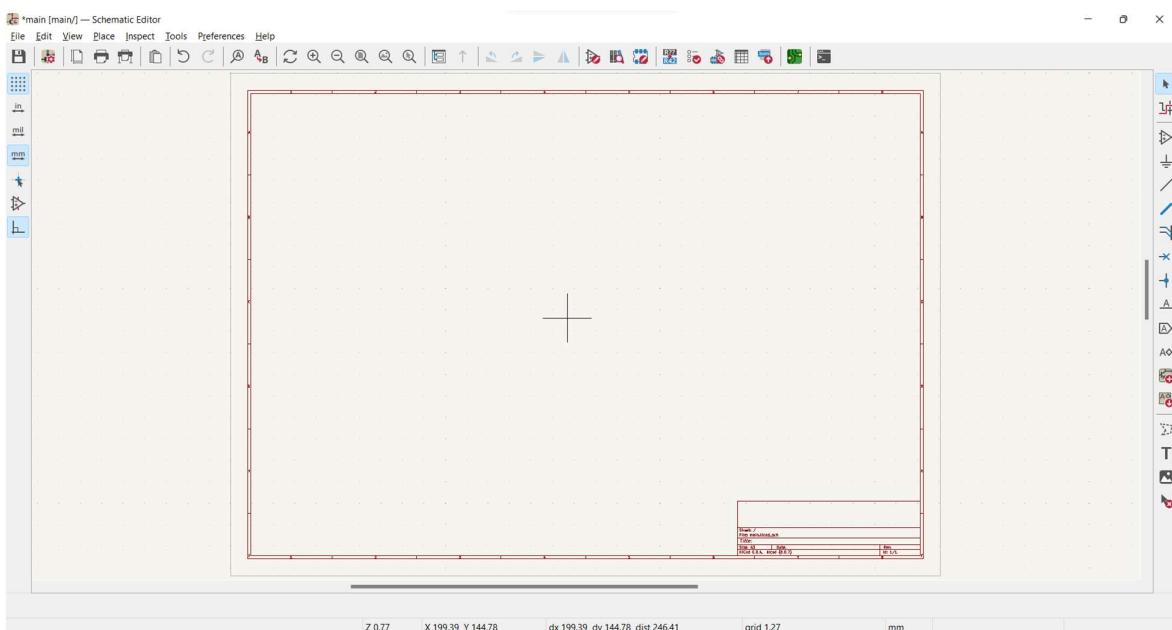


از مسیر زیر می توان پروژه جدید ایجاد کرد:



پس از مشخص کردن مسیر ذخیره پروژه دو فایل شماتیک و PCB در پروژه ایجاد میشود (برخلاف التیوم
که باید قایل های شماتیک و PCB را جدا اضاف کنیم)

با کلیک بر روی فایل شماتیک وارد محیط شماتیک می شویم:



اولین مرحله اضاف کردن کتابخانه ها به نرم افزار است.

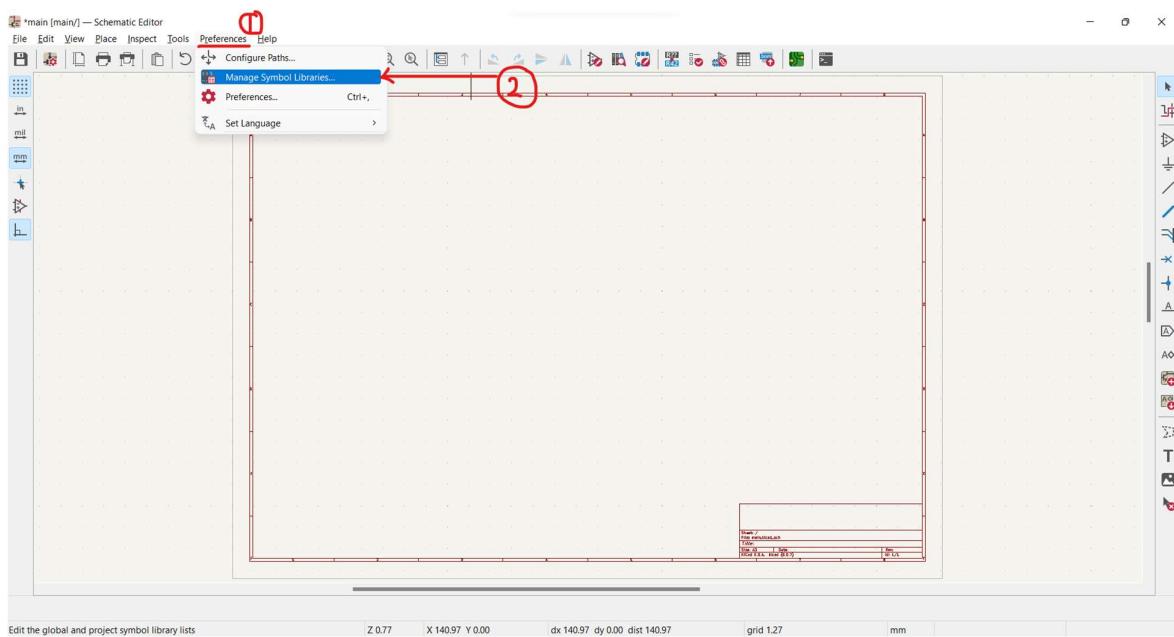
خوبی بختانه چون نرم افزار open source است، کتابخانه های متعددی برای آن وجود دارد.

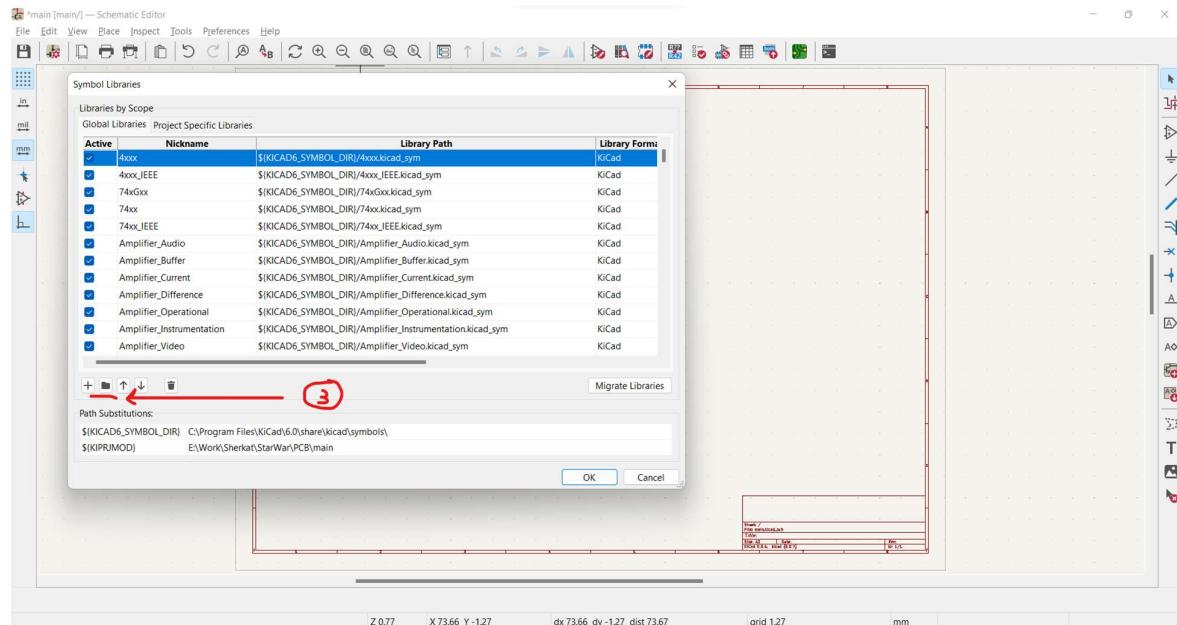
به هنگام نصب نرم افزار پکیج کامل کتابخانه ها در مسیر پیش فرض زیر نصب می شود:

C:\Program Files\KiCad\6.0\share\kicad

از مسیر زیر می توان کتابخانه ها را به نرم افزار اضاف کرد.

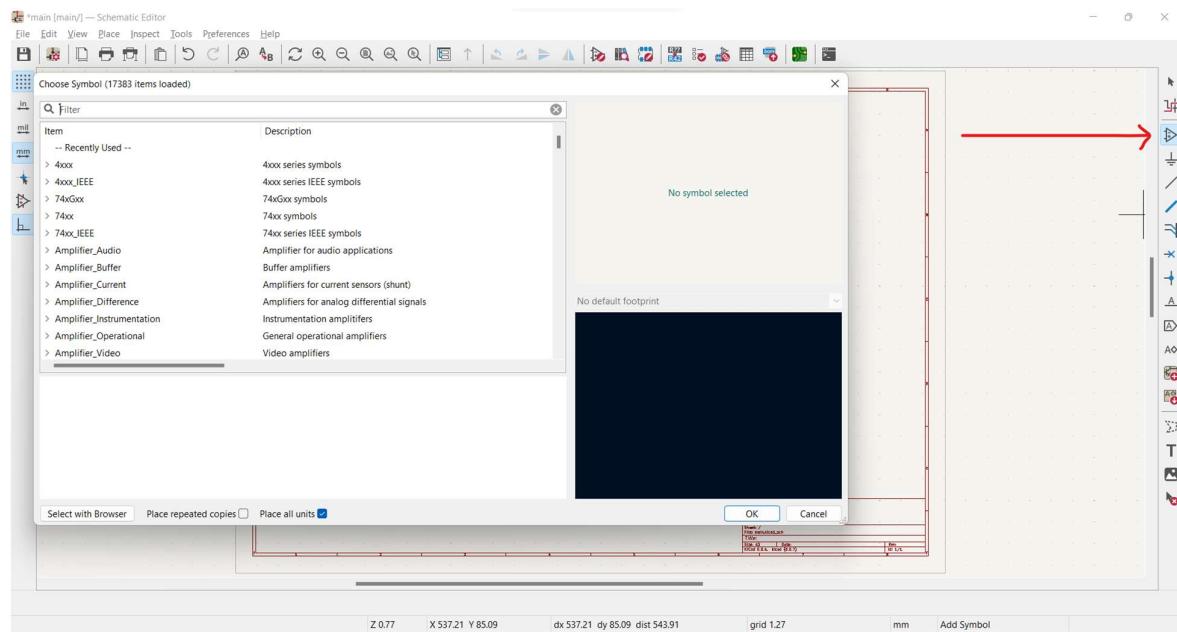
Preferences/Manage Symbol Libraries



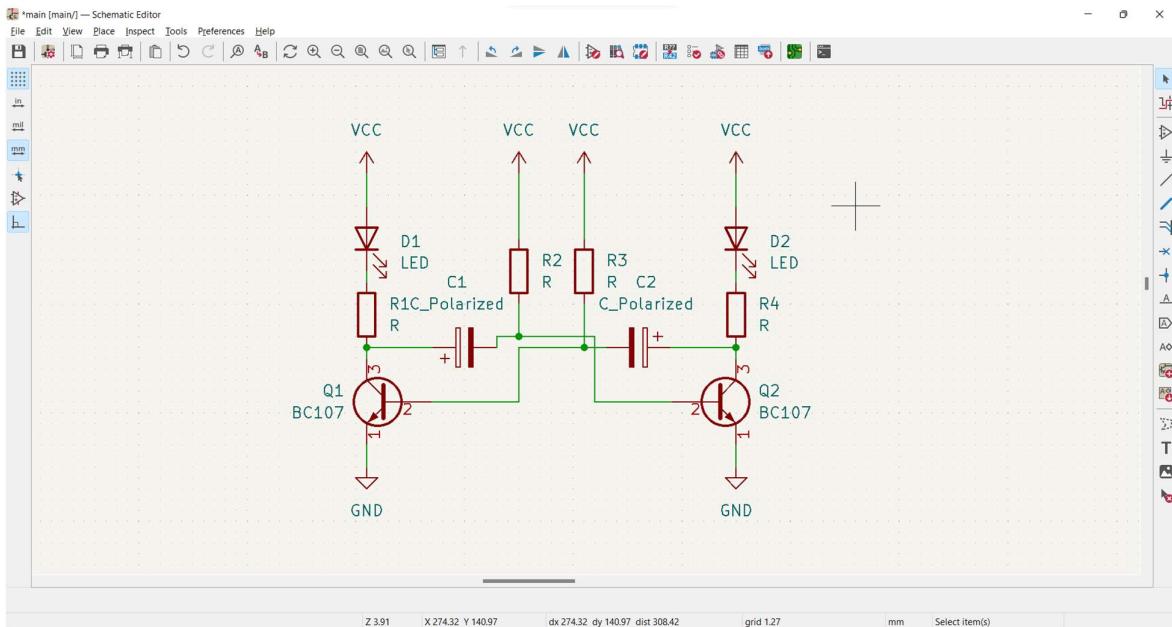


مرحله دوم در یک طراحی کشیدن شماتیک مدار است.

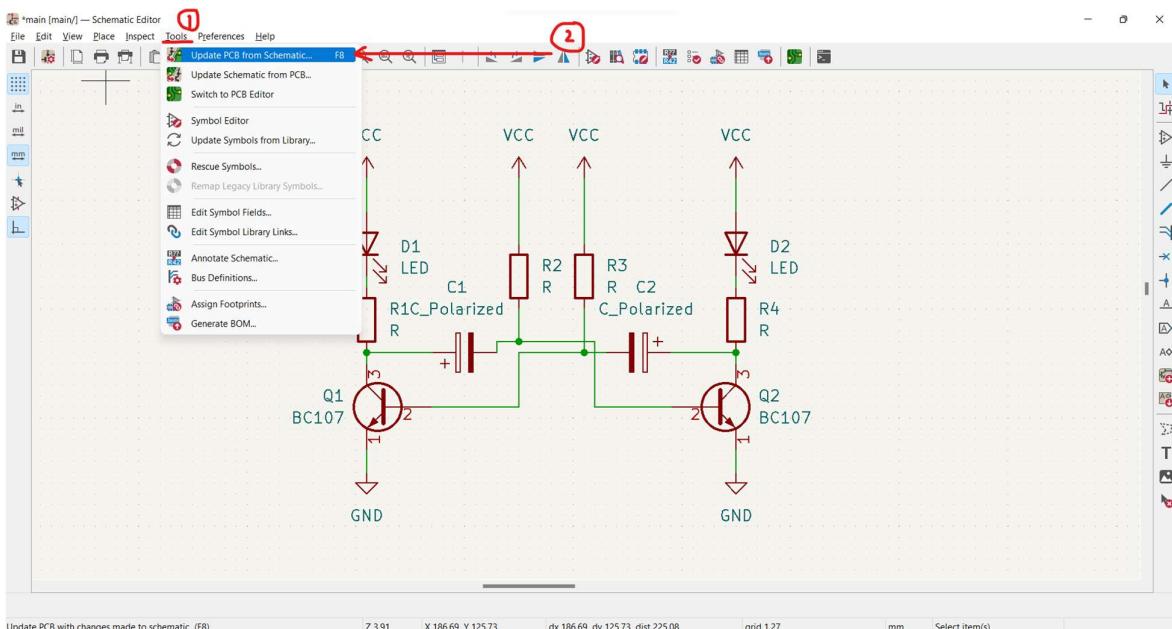
از قسمت Add symbol در سمت راست صفحه (ایکون آپ امپ) میتوان قطعات را انتخاب کرد.



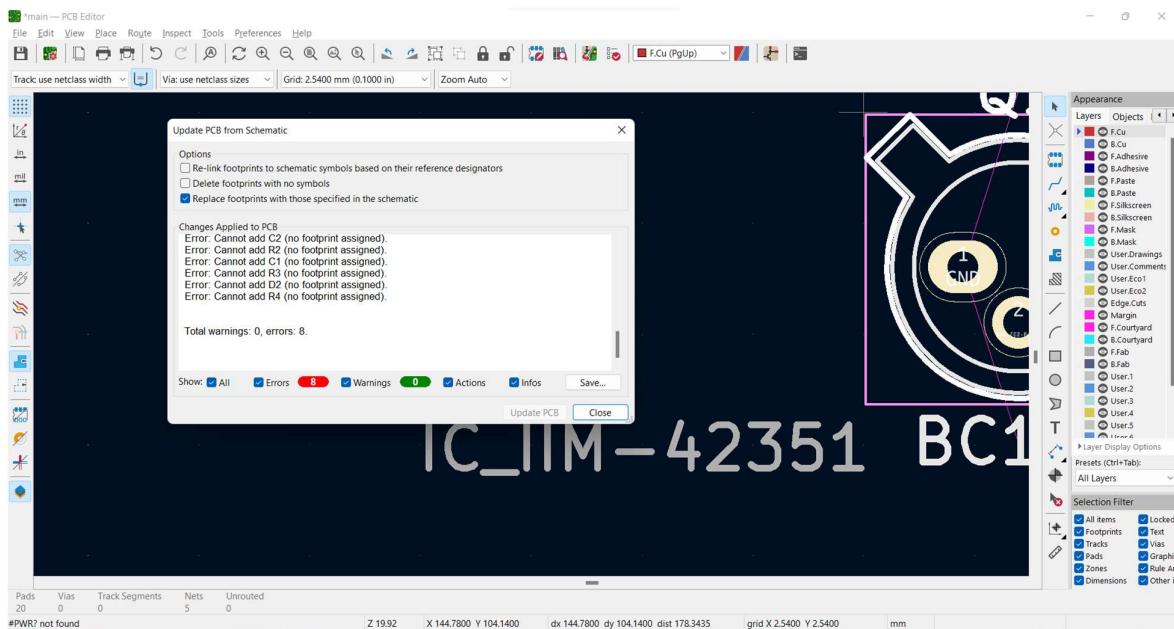
برای نمونه یک مدار ساده اشمييت تريگر را رسم ميکنيم:



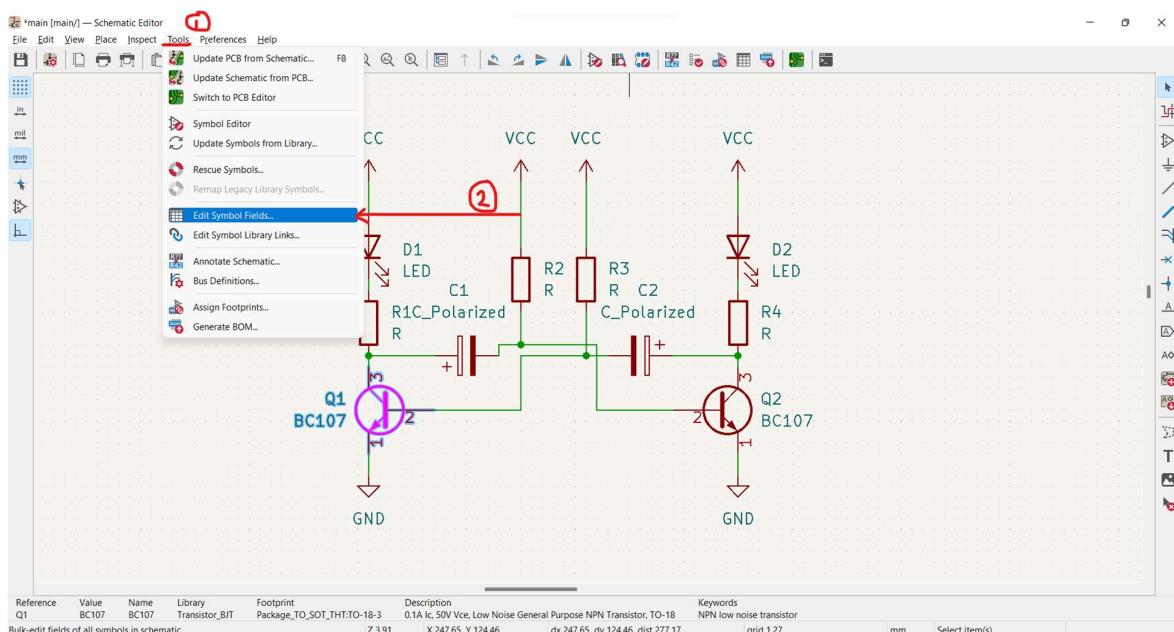
سپس از مسیر زیر میتوان شماتیک را به محیط PCB منتقل کرد:

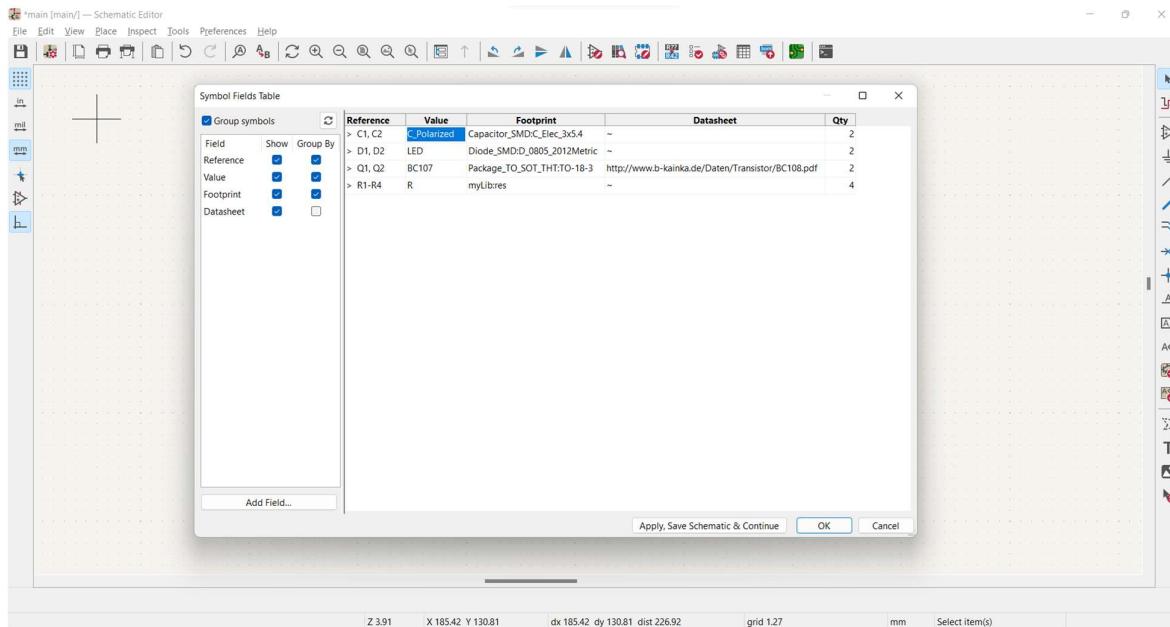


به دلیل اینکه فوت پرینت برخی از قطعات به شماتیک اضاف نشده است به هنگام انتقال به PCB با خطای زیر مواجه می شویم:



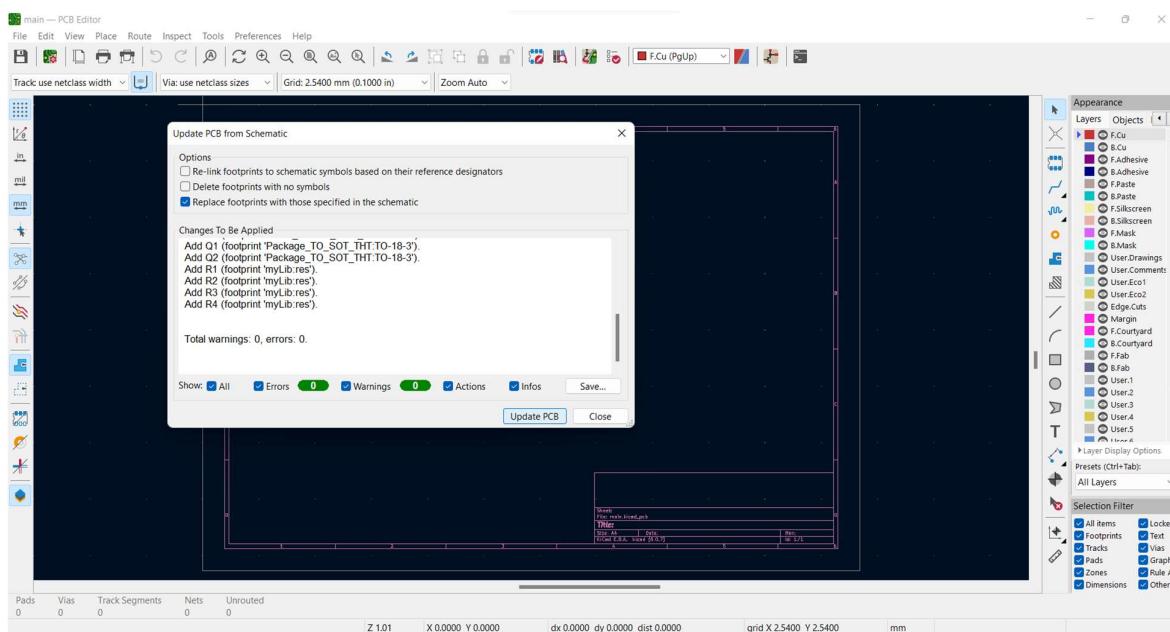
بنابر این به محیط شماتیک برگشته و از مسیر زیر فوت پرینت قطعات را چک میکنیم:

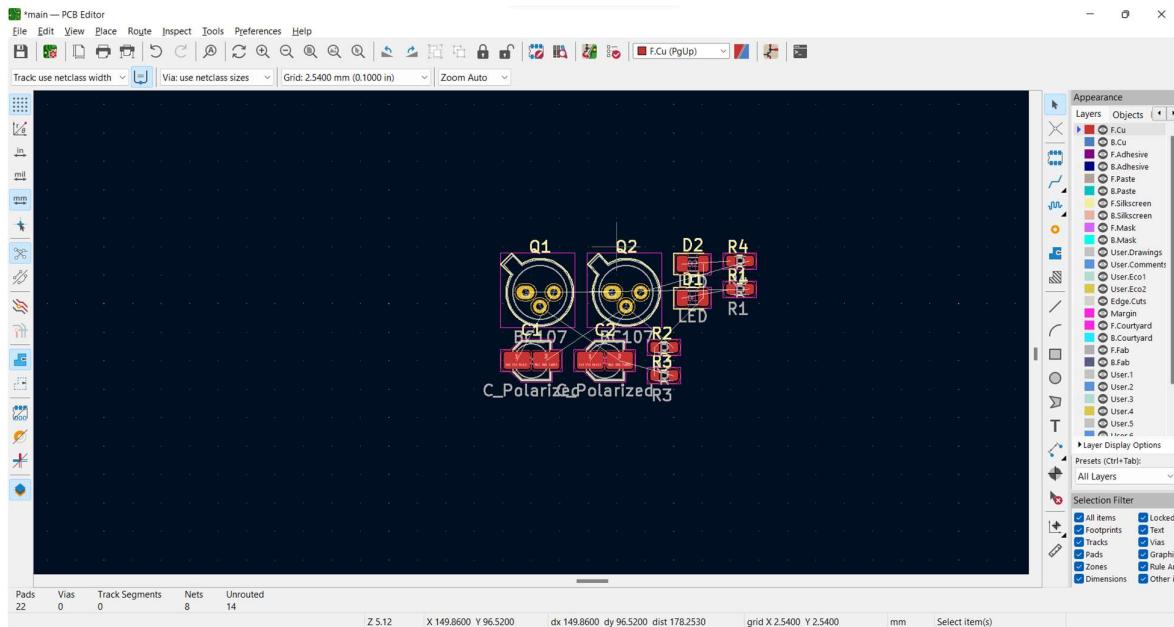




پس از انتخاب فوت پرینت ها مجدد مرحله قبل را تکرار میکنیم

مشاهده میشود که خطای رفع شده و با زدن کلید Update PCB قطعات به محیط PCB منتقل میشوند:





اولین مرحله در محیط PCB شناخت لایه ها و تنظیم آنهاست.

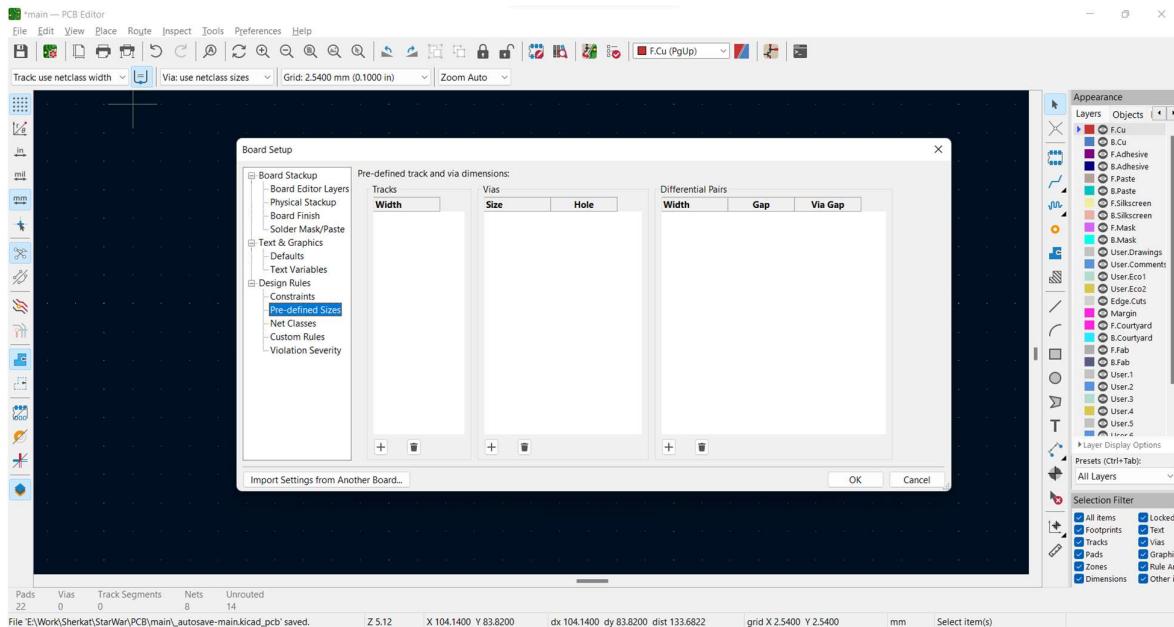
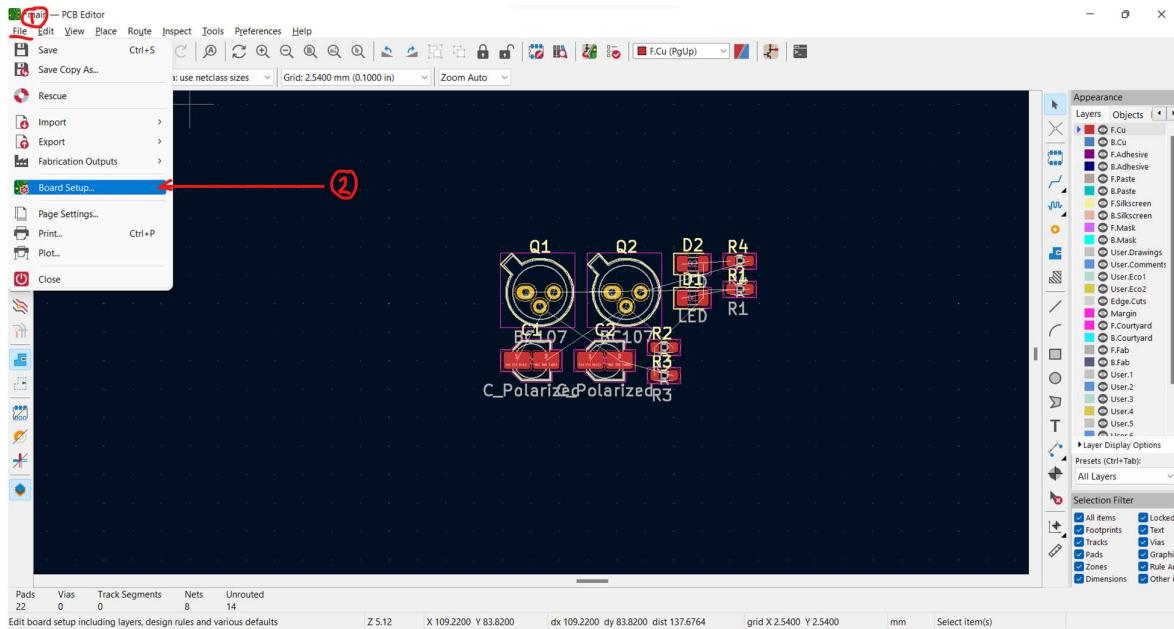
در سمت راست محیط، لایه های فعال را مشاهده میکنیم.

لایه F.Cu: لایه رو برای کشیدن ترک است (معادل Top Layer در آلتیوم)

لایه B.Cu: لایه زیر برای کشیدن ترک است (معادل Bottom Layer در آلتیوم)

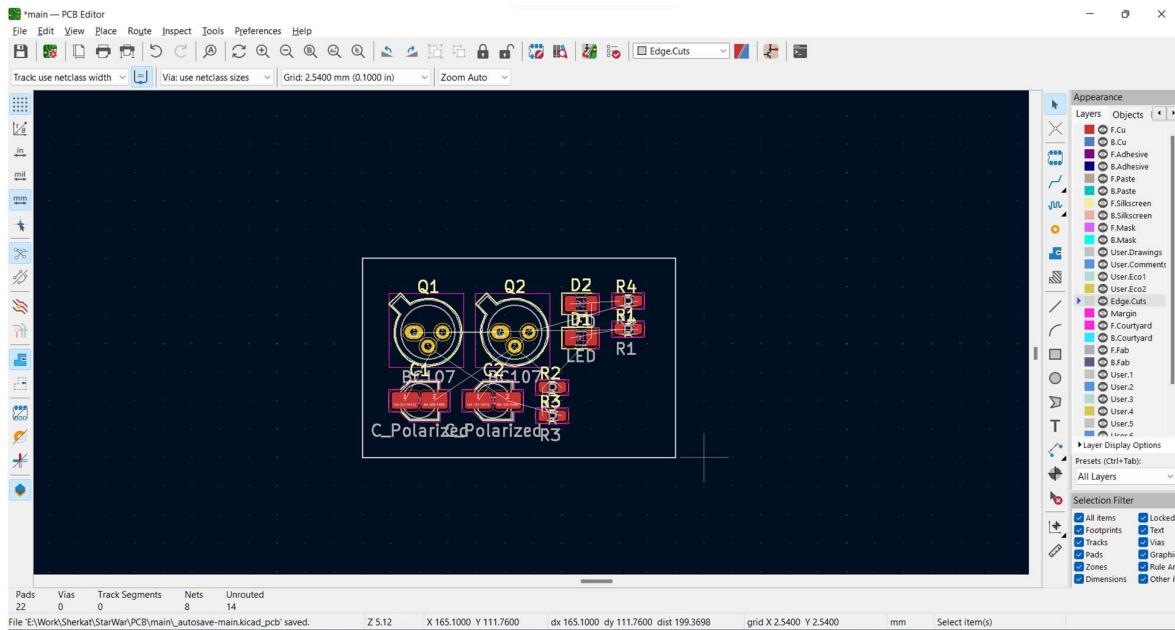
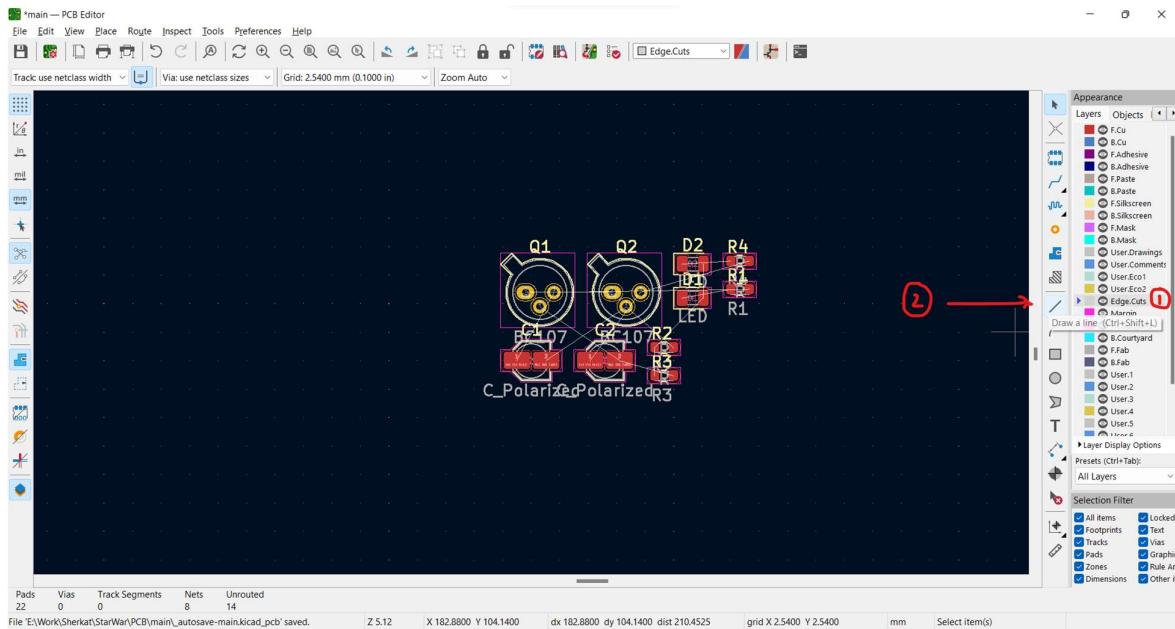
از مسیر زیر میتوان تعداد لایه های بورد و تنظیمات Rule و Clearance را مشخص کرد.

File/Board Setup

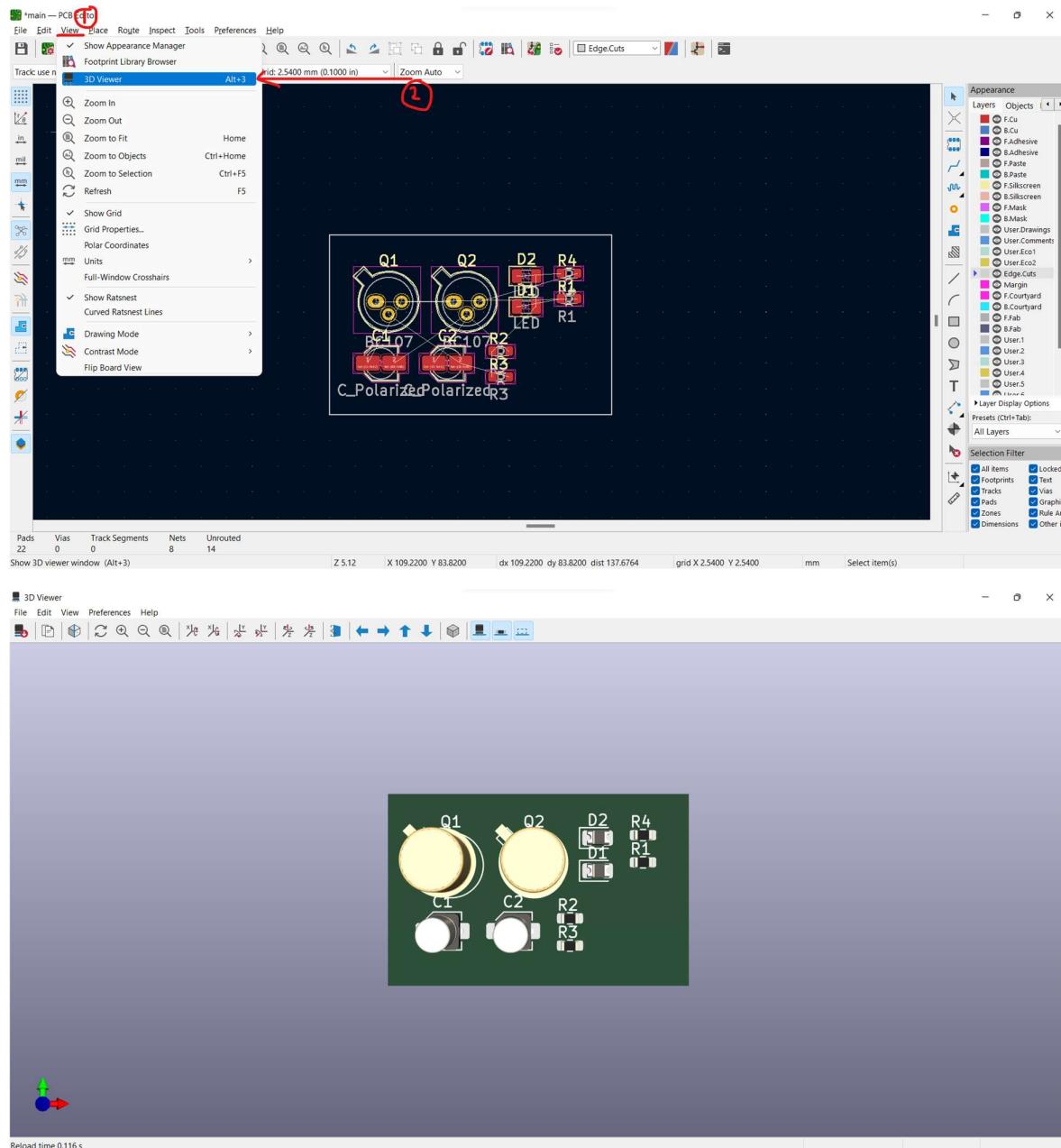


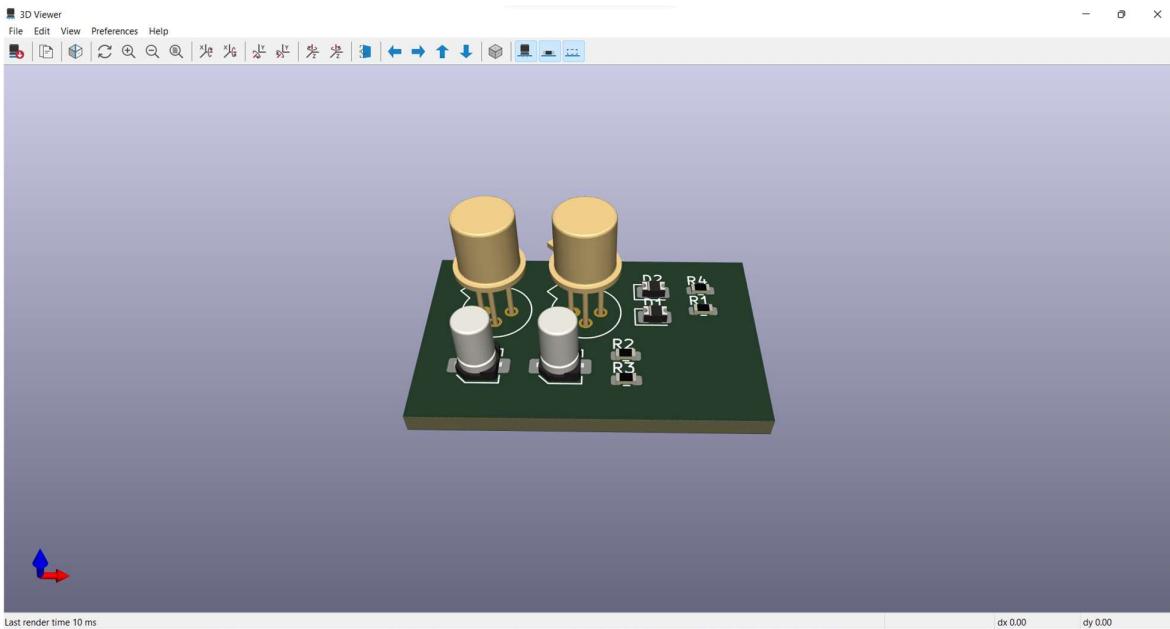
مرحله دوم مشخص کردن ابعاد و ناحیه بورد است.

لایه برش بود، Edge.Cut است که می توان از سمت راست صفحه ان را انتخاب کرد و با انتخاب کردن Draw Line ناحیه بورد را مشخص کرد.



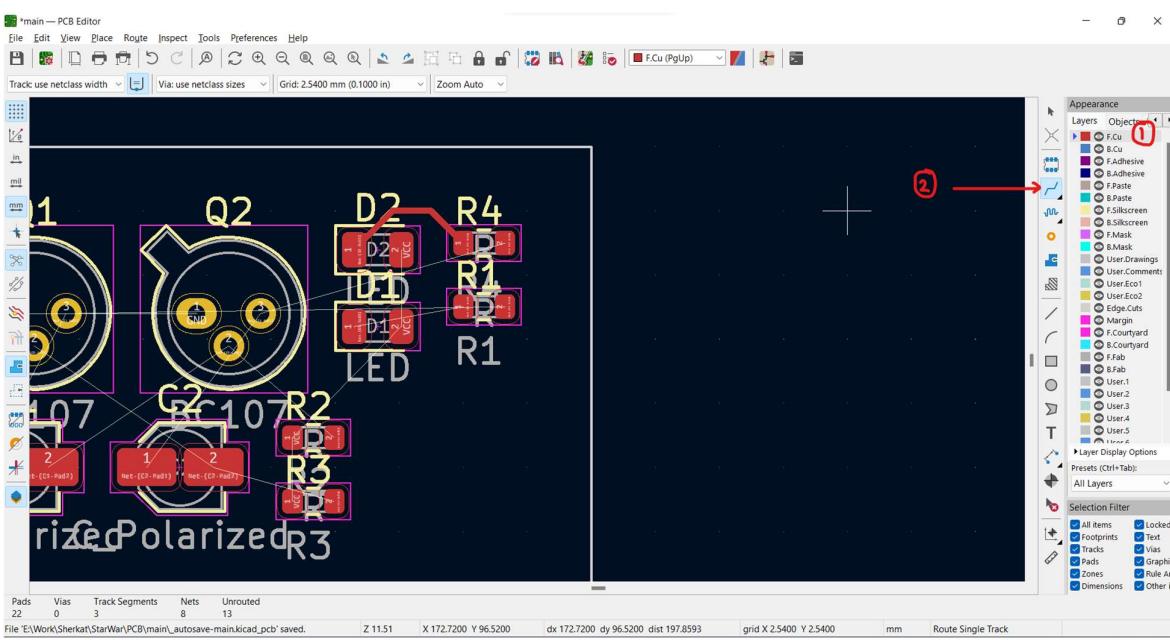
از مسیر زیر میتوان سه بعدی بورد را مشاهده کرد:



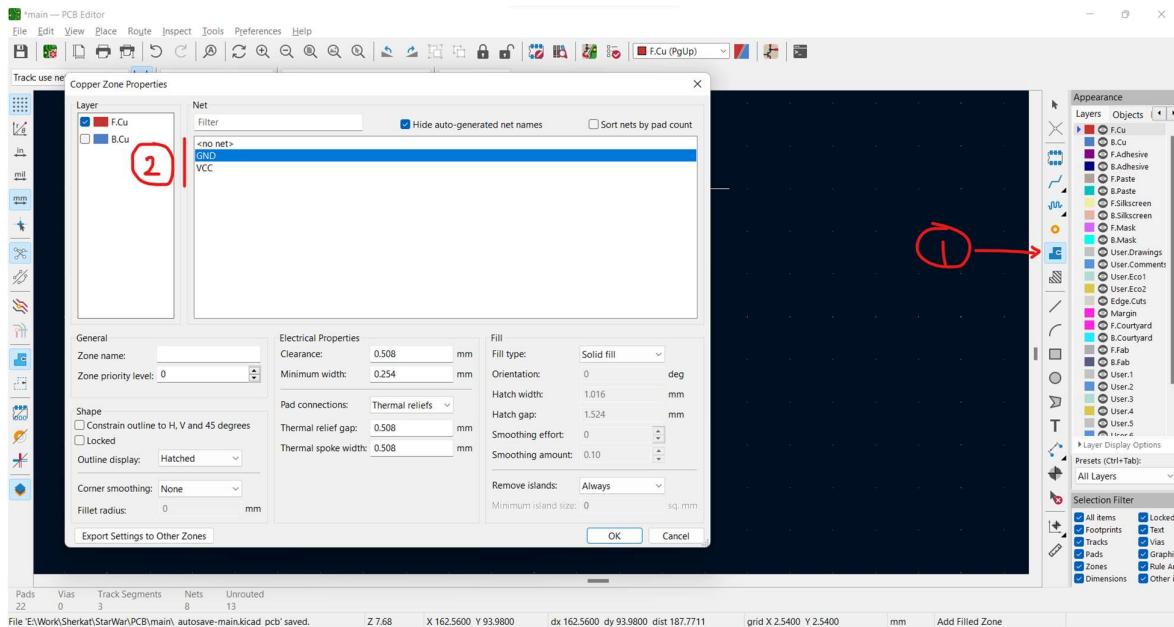


با انجام این تنظیمات اماده Route کردن بورد هستیم.

به صورت زیر لایه خود را نتخاب میکنیم و مسیر ترک ها را می کشیم:



پس از تکمیل شدت ترک کشی ها میتوان به صورت زیر به مدار پولیگان اضاف کرد:



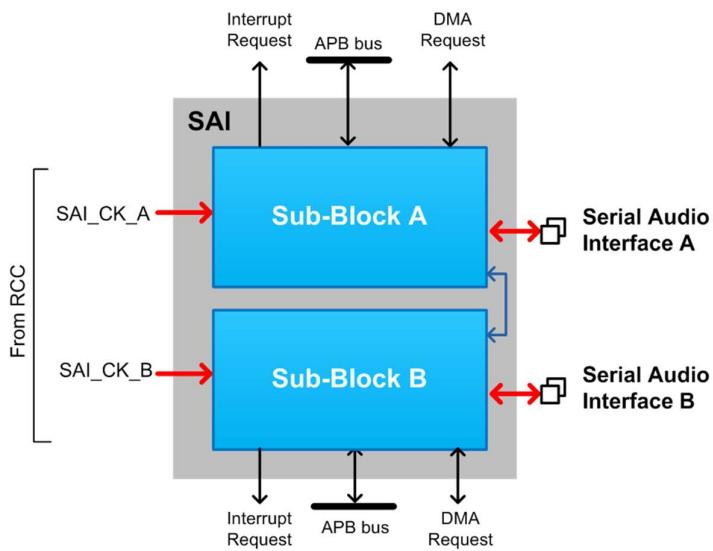
در لینک های زیر:

<https://source.mnt.re/reform/reform>

<https://mntre.com/reform2-handbook/system.html>

وظیقه دوم من در این پروژه مطالعه دیتا شیت های مربوط به پروتوكل انتقال دیتا صوتی SAI و نحوه کانفیگ آن با میکروکنترل STM-32L4 بود که جمع بندی مطالعات در این بخش بنده به صورت زیر است:

SPI (Serial Peripheral Interface) in STM32-L4



بررسی کلی:

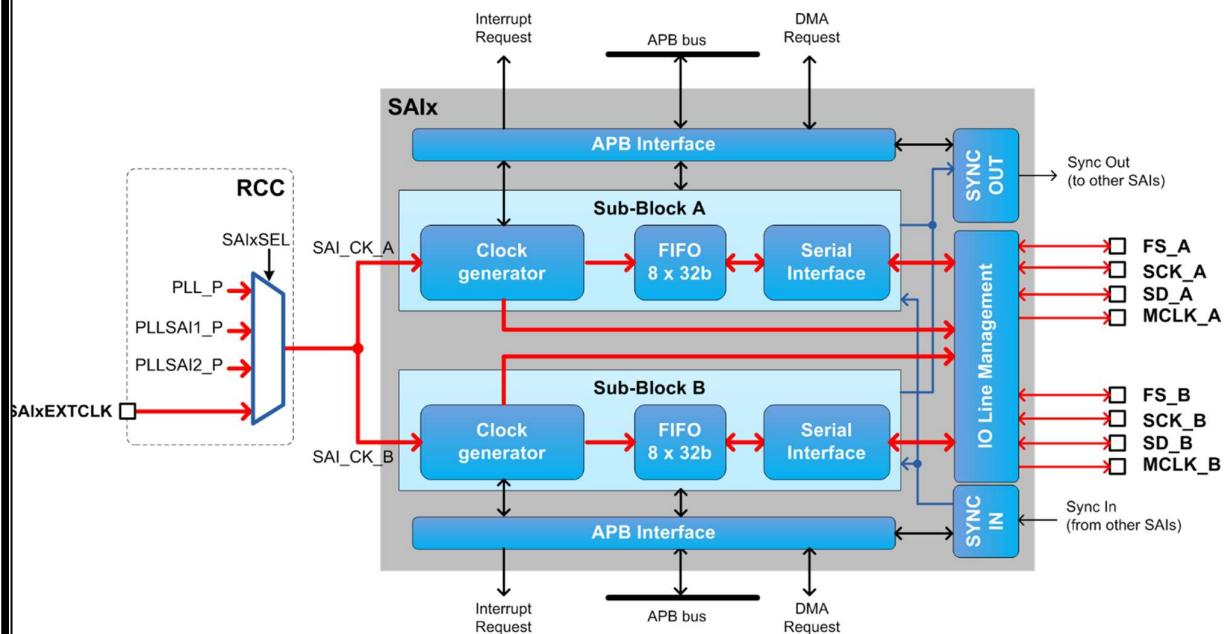
- این واحد در میکرو، رابط ارتباطی با دستگاه های صوتی خارجی را بفرار میکند.
- قابل تنظیم و برنامه ریزی است.
- استاندارد های I2C, PCM, TDM, SPDIF, AC97 را پشتیبانی میکند.
- این واحد از دو بخش کاملا مستقل از هم تشکیل میشود. (Sub-Block A, Sub-Block B)

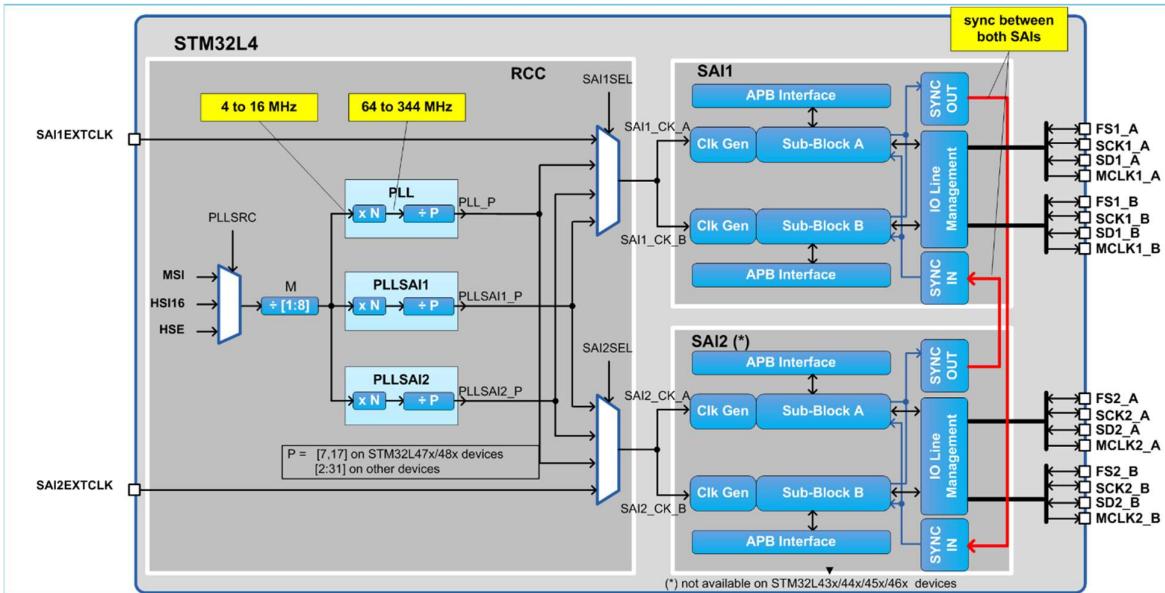
ویژگی های مهم:

- از چندین حالت مختلف برای برنامه ریزی پشتیبانی میکند:
I2C, PCM, TDM, ... :Free Protocol Mode
- SPDIF Output (Sony/Philips Digital Interface) استاندارد IEC609558 ارسال کنیم، این مد را تنظیم میکنیم.
- AC97 (Audio Codes 97 from Intel)
- تمام Sampling Rate های استاندارد را با توجه به فرکانس کریستال پشتیبانی میکند.
- از حالت Master و Slave برای هر کدام از Sub-Block ها میتوان استفاده کرد.

- میتوان تا ۳۲ بیت بر سریل به صورت Full Duplex و یا Half Duplex استفاده کرد.
- میتوان Sub-Block ها و یا چند SAI را به صورت داخلی و یا خارجی با هم سنکرون کرد.
- 8Word FIFO Size
- ۲ عدد Interrupt و DMA دارد.

بلوک دیاگرام :SAI





:لينك ها:

[https://st-onlinetraining.s3.amazonaws.com/STM32L4_Peripheral_Serial_Audio_Interface_\(SAI\)/index.html](https://st-onlinetraining.s3.amazonaws.com/STM32L4_Peripheral_Serial_Audio_Interface_(SAI)/index.html)

<https://www.youtube.com/watch?v=goffIp9ZrxM>

<https://eu.mouser.com/datasheet/2/389/stm32l433cc-1696557.pdf>

۶- استفاده از میکرو کنترلر های STM-32 سری ARM به صورت صنعتی و نحوه نوشتمن کد یک پروژه به صورت گروهی با سایر اعضای شرکت.

۷- طراحی PCB برد های الکترونیکی با استفاده از نرم افزار های Altium Designer و KiCAD

۸- حضور در نمایشگاه بینالمللی برق و الکترونیک: در مرداد ماه سال جاری، نمایشگاه بینالمللی صنعت برق و الکترونیک در مشهد برگزار شد و شرکت ما در بخش شرکت های دانش بنیان در این نمایشگاه حضور داشت و بنده به عنوان یکی از مسئولین فنی غرفه در این نمایشگاه تجربه خوبی کسب کردم. وظیفه اصلی بنده ارائه توضیحات فنی مرتبط با محصولات شرکت برای بازدیدکنندگان بود.