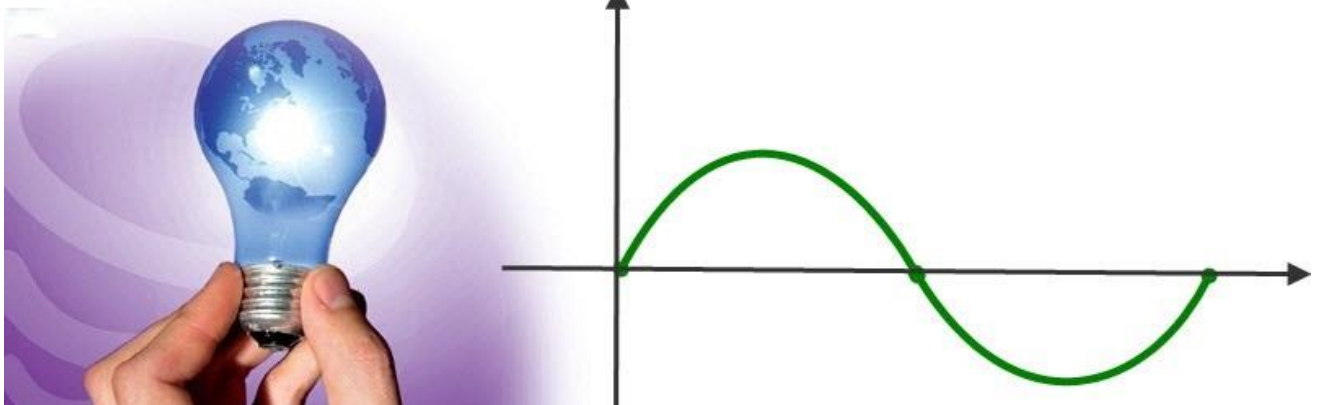


برای دریافت فایل Word پروژه به سایت **ویکی پاور** مراجعه کنید. فاقد آرم سایت و به همراه فونت های لازم



برای دریافت فایل Word پروژه به سایت **ویکی پاور** مراجعه کنید. فاقد آرم سایت و به همراه فونت های لازم

موضوع پروژه:

plc



برای خرید فایل word این پروژه [اینجا کلیک کنید](#).

(شماره پروژه = ۵۱۸)

پشتیبانی: ۰۹۳۵۵۴۰۵۹۸۶

برای دریافت فایل Word پروژه به سایت ویکی پاور مراجعه کنید. فاقد آرم سایت و به همراه فونت های لازم

خلاصه:

Plc مخفف عبارت programming logic control میباشد. این سیستم وسیله ایاست که متناسب با برنامه ای که دریافت میکند وظیفه ای خاص را اجرا میکند به عبارت دیگر plc نوعی کامپیوتر است که برنامه ای خاص را اجرا میکند .

با ظهور plc تجهیزات و قطعات استفاده شده در کنترل فرایند های صنعتی و خطوط تولید تغییر نموده و مدار های رله کنتاکتوری و سخت افزاری حالت جامد کم کم جای خود را به کنترل کننده های قابل برنامه ریزی یعنی plc دادند .

امروزه در طراحی کنترل کننده خطوط تولید و فرایند های صنعتی استفاده از مدار های رله کنتاکتوری منسوخ گردیده و در اکثر کارخانه ها و مراکز صنعتی از سیستم plc استفاده میشود.

بدون تردید plc مهمترین و پر کاربرد ترین وسیله اتوماسیون در صنایع مدرن امروزی است . در ماشین ها و خطوط تولید جدید کمتر موردی را میتوان یافت که از کنترل کننده های منطقی قابل برنامه ریزی استفاده نشده باشد .

در حقیقت این وسیله بسیار قابل انعطاف که خود یک کنترل کننده کامل است به عنوان قطعه ای برنامه ریز در صنایع گوناگون کاربرد وسیعی یافته است به گونه ای که با پیشرفت تکنولوژی و حضور اتوماسیون در عرصه صنعت در طراحی کنترل کننده ها و مدار های فرمان خطوط تولید و فرایند های صنعتی استفاده از مدار های فرمان قدیمی منسوخ گردیده و در اکثر مراکز صنعتی از کنترل کننده های منطقی قابل برنامه ریزی استفاده میگردد.

برای دریافت فایل Word پروژه به سایت **ویکی پاور** مراجعه کنید. فاقد آرم سایت و به همراه فونت های لازمه

پیشگفتار:

قرن بیستم قرنی است گسترده بین دو انقلاب. انقلابی در آغاز قرن و انقلابی در پایان آن. انقلاب آغازین ظهور تولید انبوه و پایان گرفتن عصر تولید دستی و انقلاب پایانی همانا ظهور تولید ناب و خاتمه یافتن تولید انبوه است. اکنون جهان در استانه عصر جدید به سر میبرد عصری که در آن دگرگونی شیوه های تولید مصنوعات و ساخته های بشر چهره زندگی را یکسره دگرگون خواهد کرد.

امروزه با رشد شگفت آور دانش فنی بشر و افزایش تعداد تولید کنندگان مناطق مختلف جهان سهم بیشتر بازار های جهان از آن کشور ها و شرکت های است که در خصوص کیفیت نوآوری و تنوع محصول ... حرف های تازه ای را برای گفتن دارند. اکنون تولید کنندگانی در جهان ظهور کرده اند که میتوانند با نیمی از نیروی کار و سرمایه و میزان مهندسی و مکان وزمان که برای تولید کنندگان انبوه قدیمی لازم است محصولاتی به جهان عرضه کنند که از نظر کیفیت و جنبه های نوآورانه بسی برتر باشد. اکنون دیگر ان انبوه سازان که زمانی الگو و قبله آمال دیگر تولید کنندگان بودند پس از دهها سال سروری به غیر از عقب نشینی و از دست ندادن سهم بازار خود و یا تغییر کلی شیوه های خود راه دیگری ندارند بنابراین جا دارد که پرسیم تولید کنندگان محصولات برتر چگونه توانسته اند در مقابل تولید کنندگان انبوه قدیمی با وجود یک قرن تجربه در ساخت تولید و تجارت این میان قد علم کنند و با نیمی از سرمایه و نیروی فکری و کاری آنها و بهروری و کیفیت خود را چنین ارتقا بخشند؟

امروزه صنعت کشور بیش از هر چیز نیازمند نو سازی و به کار گیری نگرش های نوین صنعتی میباشد روش های کهنه و مرسوم در صنعت کشور کاهش بهروری و افت کیفیت را به ارمغان آورده است و این در حالی است که مرز های صنعت به سرعت در حال گسترش است و اصرار بر روش های سنتی فاصله ایران را با دنیای صنعتی افزایش خواهد داد. از طرف دیگر ورود صنعت بدون دانش فنی چیزی از این فاصله نخواهد کاست. اکنون اگر چه صنعت ایران گام هایی به سوی توسعه استفاده از اتوماسیون و سیستم های مدیریت صنعتی متکی بر این دانش برداشته است اما متاسفانه انتقال دانش فنی در این عرصه با کندی صورت میگیرد.

برای دریافت فایل Word پروژه به سایت ویکی پاور مراجعه کنید. فاقد آرم سایت و به همراه فونت های لازم

مقدمه:

امروزه با پدیدار شدن ریز پردازنده ها و پیشرفت فن اوری حالت جامد در عرصه علم و تکنولوژی که بی شک ان را میتوان بزرگترین پدیده در علم الکترونیک دانست چهره محیط های صنعتی به کلی دگرگون شده است .

Plc نیز مولود این پدیده یعنی ظهور ریز پردازنده ها بوده است . بدن تردید plc مهمترین و پر کاربرد ترین وسیله اتوماسیون در صنایع مدرن امروزی است در ماشین ها و خطوط تولید جدید کمتر موردی را میتوان یافت که از کنترل کننده های منطقی قابل برنامه ریزی استفاده نشده باشد در حقیقت این وسیله بسیار قابل انعطاف که خود یک کنترل کننده کامل است به عنوان قطعه ای برنامه پذیر کاربرد وسیعی یافته است به گونه ای که با پیشرفت تکنولوژی و حضور اتوماسیون در عرصه صنعت در طراحی کنترل کننده ها و مدارات فرمان خطوط تولید و فرایندهای صنعتی استفاده از مدارات فرمان قدیمی منسوخ گردیده و در اکثر مراکز صنعتی از کنترل کننده های منطقی قابل برنامه ریزی استفاده میگردد.

اکنون برای توجه بیشتر به تفاوت ها و مزایای plc نسبت به مدارات کنتاکتوری موارد زیر را بر می شماریم :

- استفاده از plc موجب کاهش حجم تابلوی فرمان میگردد
- استفاده از plc مخصوصا در فرایندهای عظیم موجب صرفه جویی قابل توجهی در هزینه لوازم و قطعات میشود
- plc استهلاک مکانیکی ندارد بنابراین علاوه بر عمر بیشتر نیازی به تعمیرات و سرویس های دوره ای نخواهد داشت
- plc انرژی کمتری مصرف میکند
- plc ها بر خلاف مدارات رله کنتاکتوری نویز الکتریکی و صوتی ایجاد نمی کند

برای دریافت فایل Word پروژه به سایت ویکی پاور مراجعه کنید. فاقد آرم سایت و به همراه فونت های لازم

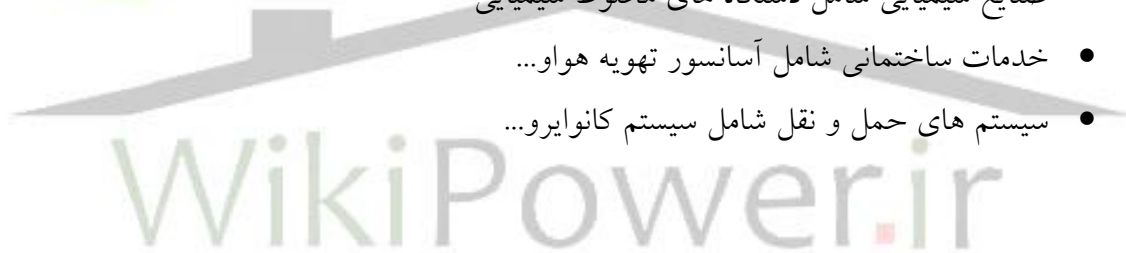
- استفاده از plc منحصر به یک پروسه و فرایند خاصی نیست و با تغییراتی که در برنامه میتوان به اسانی از ان برای کنترل پروسه های دیگر استفاده کرد
- طراحی و اجرای مدارات کنترل و فرمان با استفاده از plc بسیار سریع و اسان است
- برای عیب یابی مدارات کنتاکتوری الگوریتم و روش خاصی نداریم اما در عیب یابی مدارات plc براحتی با تغییرات در نرم افزار و simulation کردن ان میتوان عیب

یابی کرد

کاربرد های plc در صنایع مختلف :

امروزه کاربرد های فراوانی از plc در پروسه های مختلف صنعتی به چشم میخورد که خود نشانگر اهمیت فراوان plc در صنعت است . از جمله این استفاده ها میتوان به موارد زیر اشاره کرد :

- صنایع اتومبیل سازی شامل سوراخ کاری و پاشش رنگ و حمل موتور lift,drop
- صنایع پلاستیک سازی شامل ذوب قالب گیری و دمش هوا
- صنایع سنگین شامل کوره های صنعتی کنترل دمای اتوماتیک
- صنایع شیمیایی شامل دستگاه های مخلوط شیمیایی
- خدمات ساختمانی شامل آسانسور تهویه هوا...
- سیستم های حمل و نقل شامل سیستم کانوایرو...



برای دریافت فایل Word پروژه به سایت **ویکی پاور** مراجعه کنید. فاقد آرم سایت و به همراه فونت های لازم

شرح مختصری بر رساله:

Plc سیستمی است که متناسب با برنامه ای که دریافت میکند وظیفه ای خاص را انجام میدهد امروزه دز طراحی کنترل کننده های خطوط تولید و فرایند های صنعتی از ان استفاده میشود به عنوان مثال در سالن پرس ۳ ایران خودرو اتوماسیون خط شولر ساخت شرکت زیمنس و از نوع s7 و مدل cpu416-2dp که از پیشرفته ترین نوع plc هابشمار میروود مورد استفاده قرار گرفته است



برای دریافت فایل Word پروژه به سایت **ویکی پاور** مراجعه کنید. فاقد آرم سایت و به همراه فونت های لازم

PLC در یک نگاه:

PLC: programmable logic controller که با نام programmable controller نیز شناخته می شود کنترل کننده برنامه پذیری است که از خانواده کامپیوترها بشمار می آید. این کنترل کننده که عمدتاً در مقاصد صنعتی بکار می رود ورودی ها را می گیرد و بر اساس برنامه ای که در حافظه آن نوشته شده خروجی های لازم را برای ماشین یا فرایندی که تحت کنترل آن است صادر می نماید.

بنا بر این در نگاه اول PLC از سه قسمت اصلی یعنی مدول های ورودی، CPU و مدول های خروجی تشکیل شده است. مدول ورودی سیگنالهای متنوع دیجیتال یا آنالوگ را از FIELD قبول میکند و سپس آنها را به سیگنال های منطقی (۰ و ۱) که برای CPU قابل پردازش باشد تبدیل می نماید. CPU مطابق با برنامه ای که قبلاً کاربر در حافظه آن ذخیره کرده است دستورات کنترلی را اجرا کرده و خروجی لازم را بصورت سیگنال های منطقی به مدول های خارجی می فرستد. این مدول ها سیگنال های مذکور را به فرم دیجیتال یا با تبدیل به آنالوگ به تجهیزات FIELD مانند عملگرها (ACTUATOR) ارسال می نماید.

قبل از اینکه PLC در صنعت مورد استفاده قرار گیرد مدار های کنترلی کاملاً سخت افزاری بودند این مدارهای بر اساس رله ها طراحی و سپس سیم بندی می شدند. بزرگترین عیب این روش آن بود که کوچکترین تغییری در سیستم کنترل مستلزم تغییر سخت افزار و سیم کشی بود که علاوه بر هزینه زیاد زمان زیادی را نیز برای اجرا نیاز داشت بعلاوه در هنگام بروز خطا کار عیب یابی این مدارها چندان ساده نبود.

سیستم جدید یعنی PLC مسایل فوق را به همراه نداشت. به سادگی قابل برنامه ریزی بود و تغییر در سیستم کنترل با تغییر در نرم افزار بر نامه کنترل بسهولت امکان پذیر می شد.

برای دریافت فایل Word پروژه به سایت ویکی پاور مراجعه کنید. فاقد آرم سایت و به همراه فونت های لازم

مزایای فوق همراه با مزایای دیگری چون کوچکتر شدن ابعاد سیستم کنترل، عیب یابی سریعتر، خرابی کمتر توانایی اجرای فانکشنهای پیچیده، توانایی تبادل اطلاعات با سیستم های دیگر... موجب شد که مدارهای رله ای بسرعت میدان را برای حضور PLC خالی کنند.

اولین PLC ها در سال ۱۹۶۸ ساخته شدند در دهه ۷۰ قابلیت برقراری ارتباط به آنها اضافه شد در دهه ۸۰ پروتکل های ارتباطی استاندارد شد و بلاخره در دهه ۹۰ استاندارد زبانهای برنامه نویسی PLC یعنی استاندارد IEC1131 ارائه گردید

استاندارد IEC1131

در سال ۱۹۷۹ یک گروه متخصص در IEC کار بررسی جامع PLC ها را شامل سخت افزار، برنامه نویسی و ارتباطات بر عهده گرفت. هدف این گروه تدوین روش های استاندارد بود که موارد فوق را پوشش دهد و توسط سازندگان PLC بکار گرفته شود. این کار حدود ۱۲ سال بطول انجامید و نهایتاً پس از بحث های موافق و مخالفی که انجام شد استاندارد IEC1131 شکل گرفت و جنبه های مختلف این وسیله از طراحی سخت افزار گرفته تا نصب، تست، برنامه ریزی و ارتباطات آن را زیر پوشش قرار داد.

PLC های مختلف زیمنس

در طبقه بندی محصولات زیمنس PLC هادر زیر مجموعه محصولات SIMATIC قرار می گیرند. برخی از آنها بصورت COMPACT طراحی و ساخته شده اند به این معنا که منبع تغذیه و CPU و مدول های ورودی و خروجی بصورت یک پارچه در کنار هم بیکدیگر متصل هستند و یک واحد تلقی می شوند و برخی دیگر به صورت مدولار هستند که بر خلاف نوع compact کاربر میتواند مدول های دلخواه از آن خانواده را بسته به نیاز خود انتخاب و در کنار هم قرار دهد. PLC های زیمنس را میتوان به پنج خانواده زیر تقسیم کرد

Simatic s5

این PLC ها که نسبتاً قدیمی هستند انواع مختلف دارند برخی مانند s5-95u به صورت compact بوده

و

حوزه عملکرد محدود دارند. برخی دیگر مانند s5-100u و s5-115 مدولار بوده و برای سیستم های کنترلی با ابعاد متوسط بکار می روند برای حوزه های عملکرد وسیع PLC های دیگری با نام های s5-

برای دریافت فایل Word پروژه به سایت ویکی پاور مراجعه کنید. فاقد آرم سایت و به همراه فونت های لازم

135u و s5-155u از این خانواده عرضه شده اند. برنامه نویسی plc های فوق با نرم افزار step5 انجام میگیرد.

Simatic s7

این plc ها بعد از s5 عرضه شده اند و خود به سه خانواده مختلف تقسیم می شوند s7-200 بصورت compact بوده و برای سیستم های کنترلی کوچک بکار می رود. s7-300 مدولار است و عملکرد متوسط دارد s7-400 نیز مدولار است ولی می تواند حوزه عملکرد وسیع داشته باشد. این plc ها با نرم افزار step7 برنامه نویسی و پیکر بندی می شوند.

Logo!logic modules

کنترل کننده ساده و ارزان قیمتی است که برای کار های کنترلی کوچک (مانند ساختمان ها یا ماشین های کوچک) کاربرد دارد. این plc بصورت compact است و برنامه ریزی آن توسط کلید های روی آن انجام می شود. برای برنامه ریزی از طریق کامپیوتر باید نرم افزار logo!softcompact نصب گردد.

Simatic c7

C7 ترکیبی است از s7-300 و operator control علاوه بر اینکه کار کنترلی را انجام می دهد بر روی نمایشگر آن میتوان پیغام ها، رخدادها، مقادیر مرتبط با فرایند را دید و فانکشن هایی را نیز توسط صفحه کلید روی آن اعمال نمود. C7 کمپکت بوده و انواع مختلفی دارد که توانایی آنها با هم متفاوت است برای برنامه نویسی این plc ها باید علاوه بر step7 نرم افزار protocol نیز روی کامپیوتر نصب شود

Simatic505

سری 505 که خود انواع مختلفی دارد برای کاربرد در حوزه های کوچک و متوسط طراحی شده است همه اعضای این خانواده به صورت compact عرضه می شوند و برنامه نویسی آنها با نرم افزار texas instruments می باشد.

s7 خانواده

s7-20

برای دریافت فایل Word پروژه به سایت ویکی پاور مراجعه کنید. فاقد آرم سایت و به همراه فونت های لازم

یک micro plc ارزان قیمت است. می تواند برای مقاصد ساده تا نسبتا پیچیده کنترلی بکار رود. نصب برنامه نویسی، و کار با آن ساده است. بصورت compact عرضه می شود و i/o های آن on-bord است. انواع مختلفی دارد و در برخی انواع آن می توان مدول اضافی نیز در کنار cpu قرار داد. برنامه نویسی آن با نرم افزار step7-micro/win انجام می شود.

S7-300

یک mini plc است. حوزه عملکرد آن متوسط است مدولار است مدول های آن تنوع زیاد دارد سهولت قابل توسعه است بر نامه نویسی آن با step7 انجام می شود

s7-300f

برای سیستم های که نیاز به ایمنی زیاد دارند یا اصطلاحا fail-safe هستند طراحی شده است پایه آن s7-300 است در انتهای کد cpu حرف f معرف این نوع است مانند cpu315f

S7-300c

شبیه s7-300 است با این تفاوت که cpu همراه با مدول دیگری مانند ورودی خروجی بصورت compact عرضه شده است در انتهای کد cpu حرف c معرف این نوع است مانند cpu314c

S7-400

حوزه عملکرد وسیع دارد مدولار است حجم زیادی از سیگنال ها را می تواند پوشش دهد براحتی قابل توسعه است در مقایسه با s7-300 سرعت پردازش بالاتر، حافظه بیشتر و امکانات وسیعتری را داراست برنامه نویسی آن با s7 انجام میشود

S7-400h

پایه ان همان s7 است ولی در جایی که high availability مورد نیاز است بکار می رود مانند جایی که هزینه راه اندازی مجدد سیستم پس از رفع عیب بالا است پروسه ای که اگر متوقف شود منجر به خسارت زیاد می شود جایی که بهره برداری از پروسه بدون مانیتورینگ و با حداقل پرسنل تعمیراتی انجام می شود.

S7-400fh

پایه آن s7-400 است توانائی های s7-400h را دارا است توانائی های f-system را دارا است یعنی برای کاربرد هائی که درجه ایمنی بالائی دارند نیز متناسب است

برای دریافت فایل Word پروژه به سایت ویکی پاور مراجعه کنید. فاقد آرم سایت و به همراه فونت های لازم

S7 و نسخه های مختلف آن :

در نگاه اول نرم افزار S7 را باید به دو نوع زیر تقسیم نمود:

۱. S7-micro win که برای plc های S7-200 بکار می رود

۲. S7 که برای S7-300, S7-400 و همچنین C7 بکار می رود.

مورد دوم یعنی S7 نسخه های مختلفی دارد که آخرین آنها نسخه step7 v5.3 می باشد از مارس

۲۰۰۴ عرضه شده است و تفاوت های مختصری با نسخه قبلی آن یعنی نسخه 5.2 دارد

Step7(5.2) از دسامبر ۲۰۰۲ به بازار آمد و جایگزین نسخه قبلی یعنی S7 v5.1 گردید به طور کلی

این نرم افزار قادر به انجام امور زیر روی کنترل کننده ها و متعلقات آنها میباشد:

پیکر بندی سخت افزار و تنظیم پارامتر های آن

-پیکر بندی و تنظیم ارتباطات(شبکه)

-برنامه نویسی

-تست و راه اندازی و عیب یابی

ارشیو سازی

در v5.2 نسبت به نسخه قبلی امکانات جدید تری اضافه شده است که از مهمترین آنها می توان امکان

پیکر بندی سخت افزار در مد کاری run یا اصطلاحاً cir (configuration in run) را نام برده در

فرایندهای پیوسته که هیچ توفیقی نباید ایجاد شود توسط این قابلیت میتوان در مد run پیکر بندی سخت

افزار را تغییر داد مثلاً یک مدول جدید اضافه کرد در این حال وقفه ای که به پروسه داده می شود کمتر از

یک ثانیه خواهد بود و در طول این مدت ورودی ها و خروجی ها آخرین حالت خود را حفظ می کند

cir برای cpu های S7-400 از firm ware 3.1 به بعد امکان پذیر است.

Step7 mini ,step 7 lite

این دو نسخه هایی از S7 هستند که نسبت به step7 پایه (یعنی v5.1 یا v5.2) امکانات کمتری در

انها وجود دارد و برای کارهای نسبتاً ساده تر طراحی شده اند به عنوان مثال نسخه lite :

برای دریافت فایل Word پروژه به سایت ویکی پاور مراجعه کنید. فاقد آرم سایت و به همراه فونت های لازمه

فقط برای s7 300 قابل استفاده است .

برنامه نویسی فقط به سه زبان lad, fbd, stl امکان پذیر است

ارتباط با شبکه را ساپورت نمی کند .

: Step 7 professional

در این نسخه علاوه بر s7 v5.2 پکیج های دیگری که قبلا به صورت optional عرضه می شدند یکجا ارائه شده اند که عبارتند از :

S7-plcsim سیمولاتور نرم افزاری است

S7-pdiag برای تشخیص عیب بکار می رود

S7-graph v5.2 برای برنامه نویسی به صورت sfc بکار می رود

S7-scl v5.2 برای برنامه نویسی بصورت st بکار می رود

مزیت های s7 به s5 :

S7 نسبت به s5 نقاط قوت و مزیت های متعددی دارد اما از مهمترین ویژگی های آن می توان به دو مورد زیر اشاره کرد :

۱- تطابق با استاندارد iec ۱۱۳۱:

زیمنس مدعی است که این استاندارد بویژه بخش سوم آنرا که مربوط به برنامه نویسی است در s7 تا حد زیاد رعایت کرده است در حالیکه s5 فاقد این تطابق است

کارت یا مبدل ارتباطی بین کامپیوتر و plc که می تواند یکی از انواع زیر باشد :

Pc adaptor

این اداپتوراز یک طرف به پورت mpi کنترل کننده وصل می شود و از سمت دیگر به کامپیوتر . دو نوع

آداپتور قابل اتصال به پورت usb را نشان می دهد

کارت برای نصب در اسلات isa یا pci کامپیوتر

با نصب این کارت خروجی مستقیما توسط کابل وکانکتور به plc متصل می گردد و نیاز به آداپتور بیرونی

نمی باشد (مانند کارت cp5611)

کارت pcmcia :

برای دریافت فایل Word پروژه به سایت ویکی پاور مراجعه کنید. فاقد آرم سایت و به همراه فونت های لازم

این کارت در اسلات notebook نصب می گردد مانند کارت cp5511

تذکر : اگر به جای کامپیوتر از pg استفاده شود نیازی به استفاده از مبدل های فوق نیست pg های زمینس دارای پورت خروجی که مستقیما به plc وصل می گردند هستند. پساز اینکه کارت ارتباطی در اسلات کامپیوتر قرار گرفت و توسط کابل ارتباطی به پورت plc متصل گردید باید تنظیم های لازم انجام پذیرد. برای اداپتور نیز ابتدا انرا به پورت plc وصل کرده و سپس ارتباطش را با کامپیوتر توسط کابل ارتباطی برقرار می کنیم تنظیمات لازم توسط برنامه set pag/pc inter face که اکنون انرا بعد از نصب s7 میتوان در control panel مشاهده کرد امکان پذیر است .

Mpi در حالتی انتخاب می شود که آداپتور به پورت mpi مربوط به plc متصل باشد

Profibus در حالتی انتخاب می شود که آداپتور به پورت dp مربوط به plc متصل باشد auto هر دو حالت فوق را پوشش می دهد با کلیک روی properties می توان مشخص کرد که آداپتور به کدام پورت سریال متصل شده است سایر پارامتر ها را معمولا برای آداپتور لازم نیست تغییر دهیم سرعت پیش فرض ۱۹۲۰۰ میباشد اگر ۳۸۴۰۰ انتخاب شود بشرط اینکه کابل ارتباطی انرا ساپورت کند باید این تنظیم توسط dip سوئیچ روی آداپتور در حالتی که اکتیو نیست نیز انجام شود نکته دیگری که باید خاطر نشان شود این است که سیستم عامل های me,98,95,xp,windows2000 به طور اتوماتیک کارت یا آداپتور را میشناسد ولی در windows nt باید به صورت دستی اختصاص داده شود چون nt قابلیت plug and play را ندارد.

نرم افزار های جنبی و مرتبط با s7 :

برخی نرم افزار های دیگر که توسط زمینس در خانواده simatic عرضه می شوند و بعضا مکمل

step7 هستند با تقسیم بندی به سه دسته hmi, runtime, engineering در زیر آمده است

: Engineering tools

S7 scl

زبان برنامه نویسی سطح بالا میباشد که با زبان st ذکر شده در استاندارد iec1131-3 تطبیق دارد و برای plc های s7 300 cpu 314 و بالاتر و s7-400,c7 بکار می رود همانطور که قبلا اشاره شد

این نرم افزار در نسخه step 7 professional موجود است

S7 higraph

برای دریافت فایل Word پروژه به سایت ویکی پاور مراجعه کنید. فاقد آرم سایت و به همراه فونت های لازم

برای کنترل ترتیبی بصورت گرافیکی با ابزار های پیشرفته و در plc های s7-300,s7-400,c7 بکار می رود

S7graph

برنامه نویسی به صورت گرافیکی است که برای کنترل ترتیبی بکار می رود و با زبان sfc مندرج در استاندارد iec 1131-3 تطبیق دارد و برای plc های s7-300,s7-400 بکار میز رود این نرم افزار در نسخه s7 professional موجود است .

: S7plcsim

سیمولاتور نرم افزاری است که برای تست برنامه وقتی plc در دسترس نباشد بکار می رود این نرم افزار نیز در نسخه s7 professional موجود است

: Cfc

توسط این نرم افزار برنامه نویسی بصورت گرافیکی توسط یکسری بلوک های از پیش تعیین شده طراحی و انجام می شود. این نرم افزار را باید جداگانه تهیه کرد و برای s7-300,s7-400,f/h system کاربرد دارد

: S7-pdiag

ابزار عیب یابی است که برای plc های s7-300 با cpu314 و بالاتر و s7-400 بکار می رود در نسخه s7 professional موجود است

: Teleservice

برای ارتباط با plc از طریق خط تلفن به کار می رود وقتی plc توسط آداپتور خاص (ts) به مودم متصل باشد با استفاده از کامپیوتر به صورت remote می توان انرا از هر نقطه ای برنامه نویسی و رفع عیب کرد

: Docpro

برای دریافت فایل Word پروژه به سایت ویکی پاور مراجعه کنید. فاقد آرم سایت و به همراه فونت های لازم

برای مستند سازی به کار می رود با استفاده از آن می توان پس از اتمام پیکر بندی و برنامه نویسی نقشه های wiring و متن برنامه را با فرمت مناسب تهیه و چاپ کرد

: Standard pid control

ابزار کمکی برای طراحی کنترل کننده های pid است که برای plc های s7-300 با cpu31c و بالاتر و s7-400,c7 بکار می رود

: Fuzzy control

برای کنترل فازی است و در مواردی به کار می رود که توصیف ریاضی پروسه مشکل یا ناممکن باشد در برخی موارد ترکیب این روش با لوپ های pid نتیجه بهینه را برای سیستم کنترل به همراه دارد

: Modular pid control

ابزلی است که برای طراحی لوپ های کنترلی پیچیده بکار می رود و دارای فانکشن ها و بلوک های از قبل طراحی شده می باشد

: Neurosystem

شبکه های عصبی مورد استفاده در سیستم کنترل را می توان با این ابزار طراحی کرد و آموزش داد.

: Prosave mpi

برای پردازش ترافیک دیتا در شبکه mpi بین سیستم های s7,m7,c7 بکار می رود

: Simatic protocol

ابزار پیکر بنی است که برای سیستم های کنترل اپراتوری و بخش مانیتورینگ مربوط به c7 بکار می رود

: Simatic win cc

نرم افزاری است که برای طراحی سیستم مانیتورینگ بکار می رود

جایگاه نرم افزار s7 در سیستم کنترل :

در هنگام طراحی معمولاً نیازی به اینکه plc یا ماشین در کنار pc یا pg موجود باشد نیست فقط لازم است که قبل از شروع کار فرایند به خوبی مطالعه شده و ورودی و خروجی ها مشخص باشند و منطق

برای دریافت فایل Word پروژه به سایت ویکی پاور مراجعه کنید. فاقد آرم سایت و به همراه فونت های لازم

سیستم کنترل معلوم شده باشد بهتر است سخت افزار plc نیز انتخاب شده باشد با چنین معلوماتی می توان کار طراحی را با استفاده از s7 بصورت offline یعنی بدون اتصال به plc انجام داد. پس از تکمیل برنامه لازم است آنرا به plc دانلود کنیم پس در این حالت pc یا pg و نرم افزار plc ابزار کار هستند اگر سیمولاتور نرم افزاری در دست باشد بسیاری از نیاز های این مرحله را مرتفع می کند و نیاز چندانی به plc نیست .

در این مرحله ماشین یا تجهیز نیز به جمع قبلی می پیوندد و برنامه به صورت عملی و ابتدا در حالتی که ماشین بدون بار است یا از تجهیز هنوز بهره برداری نمی شود تست می گردد که به این مرحله تست سرد (cold test) نیز می گویند سیگنال ها به تدریج و نه یک دفعه وارد مدار می شوند و بخش های برنامه قدم به قدم تست می شود پس از آن تست گرم شروع می شود یعنی ماشین زیر بار می رود و از تجهیز به صورت آزمایشی بهره برداری می شود تا سایر ورودی خروجی هایی که در تست سرد فعال نبودند تست گردند . برای انجام تست های فوق وجود s7 روی pc یا pg و ارتباط online با plc ضروری است

operation یا بهره برداری :
پس از تکمیل مراحل تست و اعمال تغییرات لازم در برنامه plc کار عادی فرایند شروع می شود در اینجا نیازی به pg یا pc و نرم افزار s7 نیست اگر چه باید برای نیاز های احتمالی در دسترس باشند .
Troubleshooting یا عیب یابی :

در صورتیکه مشکلی در کار بهره برداری از فرایند پیش بیاید که ناشی از اجزای سیستم کنترلی باشد . مجدداً به pc یا pg و نرم افزار s7 نیاز پیدا می شود این برنامه با امکانات مختلفی که در آن تعبیه شده می تواند به شناخت عیب و رفع آن کمک زیادی بنماید .
تنظیم پارامتر های کارت های di

برای دریافت فایل Word پروژه به سایت ویکی پاور مراجعه کنید. فاقد آرم سایت و به همراه فونت های لازم

در پنجره کاتالوگ در زیر مجموعه sm-300 کارت های digital input متنوعی را مشاهده می کنیم که به کلیک روی آنها توضیحات مختصری راجع به کارت در پایین پنجره کاتالوگ ظاهر می شود به طور کلی این کارت ها را می توان به شکل زیر دسته بندی کرد

تقسیم بندی کارت های digital input

از نظر تعداد ورودی	از نظر ولتاژ	از نظر قابلیت های خاص
۴ ورودی	24vdc	بدون ویژگی خاص
۸ ورودی	48vdc	تشخیص قطع شدن تغذیه
۱۶ ورودی	120vdc	ایجاد وقفه بر اساس لبه ورودی
۳۲ ورودی	230vdc	تاخیر در گرفتن

برای کارت هایی که قابلیت خاص ندارند وقتی روی آنها کلیک می کنیم پنجره ای باز می شود که دو بخش دارد

: General

در این بخش توضیحاتی راجع به کارت ، ویژگیها و کد سفارش آن همراه با نام آن آمده است که کاربر در صورت تمایل میتواند نام را به دلخواه تغییر دهد .

: Address

در این بخش آدرس هایی که توسط سیستم به کارت اختصاص داده شده آمده است. start آدرس شروع و end آدرس نهایی را نشان می دهد. بعنوان مثال برای کارت DI16XDC24V با ۱۶ ورودی در شکل صفحه بعد مشاهده می کنیم که آدرس شروع ۰ و آدرس انتها ۱ است بنابراین لیست آدرس های ۱۶ کانال که

برای دریافت فایل Word پروژه به سایت **ویکی پاور** مراجعه کنید. فاقد آرم سایت و به همراه فونت های لازمه

هر کدام یک بیت (۱ و ۰) هستند مانند جدول زیر خواهد بود بعبارت دیگر این مدول دارای دو بایت آدرس

است و میدانیم که $2\text{BYTE}=16\text{BIT}$

کانال	آدرس
۰	۰,۰
۱	۰,۱
۲	۰,۲
۳	۰,۳
۴	۰,۴
۵	۰,۵
۶	۰,۶
۷	۰,۷
۸	۱,۰
۹	۱,۱
۱۰	۱,۲
۱۱	۱,۳
۱۲	۱,۴
۱۳	۱,۵
۱۴	۱,۶
۱۵	۱,۷

اگر چند مدول DI مشابه یا متفاوت داشته باشیم نیز مشاهده می کنیم که آدرس های تولید شده توسط سیستم با یکدیگر هیچ تداخلی ندارند. در S7-300 تغییر آدرس توسط کاربر بعضا امکان پذیر است برخی از CPU های ۳۰۰ این امکان را ساپورت می کنند از CPU315 به بالا .

برای دریافت فایل Word پروژه به سایت ویکی پاور مراجعه کنید. فاقد آرم سایت و به همراه فونت های لازم

در این حالت گزینه SYSTEM SELECTION قابل انتخاب است میتوان انرا غیر فعال نمود و ادرس جدید را وارد کرد شماره ادرس نمی تواند از ADDRESS AREA مربوط به CPU بزرگتر باشد بعلاوه اگر ادرس جدید تداخلی با ادرس دیگر داشته باشد سیستم پیغام میدهد و در عین حال ادرس دیگری را پیشنهاد می دهد در مجموع پیشنهاد میشود که حتی المقدور کاربر ادرس های پیش فرض سیستم را تغییر ندهد .

در بین کارت های DI موجود در کاتالوگ برخی از کارت ها قابلیت های خاص دارند . توانایی اعمال وقفه (INTERRUPT) مهمترین قابلیت آنهاست که این ویژگی در توضیحات زیر پنجره کاتالوگ دیده می شود بخش PROPERTIES این کارت ها نسبت به کارت های معمولی یک بخش اضافه بنام INPUT دارد که از بخش های زیر تشکیل شده است

: DIAGNOSTIC INTERRUPT

در حالت عادی غیر فعال است اگر فعال شود در صورت قطع تغذیه سنسور (مثلا به علت قطع فیوز) شماره کانال مربوطه در بافر تشخیص عیب CPU ثبت می شود . در جلوی NO SENSOR SUPPLY یک گزینه برای ورودی های ۰ تا ۷ و یک گزینه نیز برای ورودی های ۸ تا ۱۵ وجود دارد میتوان هر دو یا یکی را بدلیخواه فعال نمود بدیهی است در صورت قطع تغذیه آنچه در بافر ثبت می شود آدرس گروه کانال است نه ادرس خود کانال .

: HARDWARE INTRRUPT

در حالت عادی غیر فعال است اگر فعال شود جدول پایین که مربوط به تریگر کردن این وقفه است نیز فعال می شود در این جدول برای هر دو کانال ورودی یک گزینه وجود دارد . با انتخاب این گزینه میتوان تعیین کرد که وقتی ورودی این کانال تغییر میکند (لبه مثبت یا منفی) وقفه اعمال نماید .

: INPUT DELAY

در این قسمت میتوان تعیین کرد که ورودی را با چند میلی ثانیه تاخیر بگیرد توصیه میشود در دو حالت زیر این عدد را روی ماگزیمم بگذارید

برای دریافت فایل Word پروژه به سایت ویکی پاور مراجعه کنید. فاقد آرم سایت و به همراه فونت های لازم

۱- با سویچ های ساده و بدون حفاظت. تاخیر فوق باعث می شود که پرش لحظه ای ولتاژ مشکلی ایجاد نکند.

۲- اگر طول کابل تا سنسور زیاد و کابل بدون شیلد باشد تاخیر فوق باعث می شود که ورودی را در زمان مناسب بگیرد.

تنظیم پارامترهای کارت های DO :

کارت های DIGITAL INPUT موجود در کاتالوگ را نیز می توان به صورت زیر دسته بندی کرد :

از نظر تعداد خروجی	از نظر جریان خروجی	از نظر ولتاژ	از نظر قابلیت های خاص
۴ خروجی	بدون رله یا جریان های ۰,۵ و ۱,۵ و ۲ با رله و جریان های ۵ و ۸	24VDC	بدون ویژگی خاص
۸		48VDC	تشخیص قطعی
۱۶		120VDC	تشخیص اتصال کوتاه
۳۲		230VDC	واکنش در موقع توقف CPU

تذکر: کارت هایی که برای F- SYSTEM استفاده می شوند ممکن است ویژگیهای خاص دیگری نیز داشته باشند. در پنجره ای که برای نمایش ویژگی های کارت DO ظاهر میشود بخش های GENERAL, ADDRESS وجود دارند که توضیحات آن مانند کارت DI میباشد بخش OUT PUT فقط برای برخی از کارت ها که قابلیت خاص دارند ظاهر میشود.

: WIRE BREAK

برای تشخیص قطعی سیم قبل از ارسال خروجی , جریان ثابتدر زمان کوتاهی به آن تزریق میشود و براساس ولتاژ نتیجه گیری به عمل می آید.

برای دریافت فایل Word پروژه به سایت ویکی پاور مراجعه کنید. فاقد آرم سایت و به همراه فونت های لازم

: NO LOAD VOLTAGE

برای تشخیص عدم وجود ولتاژ .

: SHORT CIRCUIT TO M

برای تشخیص اتصال کوتاه به زمین

: SHORT CIRCUIT TO L+

برای تشخیص اتصال کوتاه در ۲۴ ولت

برای هر یک از موارد فوق میتوان کانال مورد نظر را علامت زد و فعال نمود

: REACTION TO CPU STOP

توسط این قسمت میتوان تعیین کرد که در صورت تحقق CPU خروجی کانال مربوطه از DO چگونه

باشد. ممکن است بخاطر مسائل ایمنی کاربر بخواهد این خروجی ها را ۱ نگهدارد. با انتخاب

SUBSTITUTE VALUE و علامت زدن کانال ها در زیر آن این امر میسر میشود بدیهی است خروجی

کانالی که علامت نخورده صفر خواهد بود اگر KEEP LAST VALUE انتخاب گردد در هنگام توقف

CPU آخرین مقدار قبلی سیگنال حفظ می گردد

تنظیم پارامتر های کارت های DO/DI :

این کارت ها همانطور که از نامشان پیداست ورودی و خروجی DIGITAL را با هم دارند. تنوع آنها کم

است و دسته بندی آنها مانند جدول صفحه بعدمی باشد

از نظر تعداد ورودی و خروجی	از نظر جریان خروجی	از نظر ولتاژ	از نظر قابلیت های خاص
۱۶ ورودی + ۱۶ خروجی ۸ ورودی + ۸ خروجی	0.5A	24VDC	بدون ویژگی خاص

پارامتر های این کارت ها صرفا شامل دو بخش GENERAL ,ADDRESS میباشد که مشابه DI,DO

میباشد. تنها تفاوتی که وجود دارد این است که بخش آدرس به دو قسمت یکی برای ورودی و دیگری

برای دریافت فایل Word پروژه به سایت ویکی پاور مراجعه کنید. فاقد آرم سایت و به همراه فونت های لازم

برای خروجی تفکیک شده است لازم به ذکر است که کارت DO/DI مختص S7-300 است و نوع S7-400 وجود ندارد سیگنال های آنالوگ:

سیگنال های input analog استاندارد انواع مختلفی دارند ولتاژ و جریان و مقاومت از جمله این سیگنال ها هستند کارت های ورودی آنالوگ ممکن است یکی یا ترکیبی از این سیگنال ها را قبول کنند. قبل از پرداختن به تنظیمات کارت های مذکور توضیحاتی در مورد سیگنال ها ارائه می دهیم سیگنال آنالوگ از جنس ولتاژ:

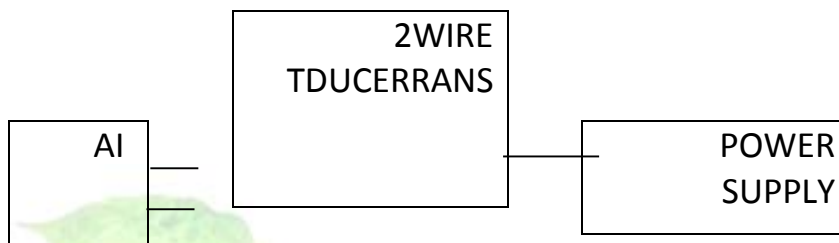
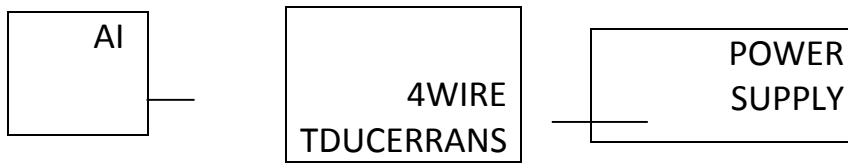
این سیگنال ها انواع مختلفی به شرح زیر دارند

- +/-25mv
- +/-50mv
- +/-80mv
- +/-250mv
- +/-500mv
- +/-1v
- +/-2.5v
- +/-5v
- +/-1..5v
- +/-10v

سیگنال آنالوگ از جنس جریان:

جریان میتواند از ترانسدیوسر (مبدل) ۲سیمه باشد یا از ترانسدیوسر ۴سیمه. با توجه به شکل زیر میتوان دریافت که در نوع ۴سیمه تغذیه سنسور از سیگنال آن جدا است (دو سیم برای تغذیه و دو سیم برای سیگنال) ولی در نوع دو سیمه تغذیه و سیگنال مشترک است .

برای دریافت فایل Word پروژه به سایت ویکی پاور مراجعه کنید. فاقد آرم سایت و به همراه فونت های لازم



سیگنال جریان از ترنسدیوسر های ۲ سیمه $4-20\text{ MA}$ و در ترنسدیوسر های ۳ سیمه بشرح زیر میباشد

WikiPower.ir

+/-5MV

+/-10MV

+/-20MV

0MA TO 20MA

4MA TO 20MA

سیگنال از نوع مقاومت :

مقاومت میتواند یک مقاومت معمولی (RESISTOR) با اهم مشخص مانند انواع زیر باشد

(48/150/300/600/6000 اهم)

همین طور میتواند یک ترمیتور (RTD) از نوع دو سیمه یا ۳ سیمه یا ۴ سیمه باشد. نوع ۲ سیمه کمترین

دقت و نوع ۴ سیمه بیشترین دقت را دارد و کاربرد نوع ۳ سیمه مرسوم تر است. در نوع ۲ سیمه مقاومت

برای دریافت فایل Word پروژه به سایت ویکی پاور مراجعه کنید. فاقد آرم سایت و به همراه فونت های لازم

کابل و دمای محیطی که کابل از آن میگذرد ایجاد خطا میکند ولی در نوع ۴سیمه به دو سر ترمیتور منبع جریان ثابتی اعمال میگردد و ولتاژ دو سر دیگر اندازه گیری میگردد پس اولاً جریان عبوری از ترمیتور ثابت است ثانیاً از مسیری که در دو سری آن ولتاژ اندازه گیری میشود جریان نمی گذرد. بنابراین آنچه اندازه گیری میشود دقیقاً نسبت ولتاژ دو سر ترمیتور به جریان عبوری از آن یعنی در واقع مقاومت ترمیتور است. در نوع ۳ سیمه چون مسیر عبور جریان در یک طرف با نقطه ای که ولتاژ آن اندازه گیری میشود مشترک است اندک خطایی وجود دارد. ترمیتور ها انواع مختلفی دارند که هر کدام برای رنج دماییو شرایط خاصی مناسب هستند. معروف ترین آنها pt100 است. لیست انواع ترمیتور ها در زیر آورده شده است .

Pt100

Pt200

Pt500

Pt1000

Ni100

Ni1000

ترموکوپل:

همانطور که میدانیم ترمیتور از دو فلز غیر همجنس که از یک سمت به یکدیگر متصل هستند تشکیل شده است. با قرار گرفتن محل اتصال در محیط گرم سیگنالی به صورت mv در سمت آزاد ظاهر میشود که میتوان از آن استفاده کرد .

اگر کابلی که به ترموکوپل وصل میشود هر رشته آن همجنس فلز مربوطه از ترموکوپل یا به آن نزدیک باشد بدون هیچ مشکلی میتوان از سیگنال استفاده کرد. ولی این کار به صرفه نیست زیرا کابل از جنس فلز ترموکوپل گرانقیمت خواهد بود. حال اگر از کابل سیمی استفاده کنیم مشکلی که وجود دارد این است که در نقطه اتصال کابل به ترموکوپل دو ترموکوپل دیگر (به دلیل همجنس نبودن دو فلز) تشکیل میشود که نهایتاً منجر به خطا در اندازه گیری میشود بنابراین ناچار هستیم از جبران کننده (compensator) استفاده کنیم یک روش الین است که دما را در نقطه اتصال کابل به ترموکوپل (reference junction) اندازه گیری کرده و از دمای ارسال شده به کارت کم کنیم. به این کار جبران سازی بیرونی (external compensation) گفته میشود این امکان در کارت های آنالوگی که ورودی ترموکوپل قبول میکنند پیش

برای دریافت فایل Word پروژه به سایت ویکی پاور مراجعه کنید. فاقد آرم سایت و به همراه فونت های لازمه

بینی شده و میتوان انرا به جبران کننده بیرونی یا یک ترمومتر مرجع متصل کرد روش اول که از کابل همجنس استفاده میشود جبران سازی داخلی (internal compensator) نامیده میشود .

تنظیم سخت افزاری لازم برای کارت های AI :

در کارت های AI که در نرم افزار HWCONFING اعمال میشود یک تنظیم سخت افزاری نیز لازم است. مدول کوچکی به نام MEASURING RANGE MODULE در کنار کارت وجود دارد که روی آن حروف A,B,C,D نوشته شده است بسته به نوع کارت و مطابق با دستورالعمل موجود در کاتالوگ آن این مدول باید تنظیم شود بنحویکه یکی از حروف مدول به سمت جلوی کارت قرار گیرد. در پارامتر های کارت که توسط HWCONFING تنظیم میشود وضعیت مدول به عنوان راهنمایی داده میشود .

به عنوان مثال برای یک کارت AI8*12BIT باکد سفارش 6ES7 331-7KF01-0AB0 که انواع ورودی را قبول میکند هر دو کانال یک MEASURING RANGE MODULE دارد که تنظیم آن باید مطابق جدول زیر باشد

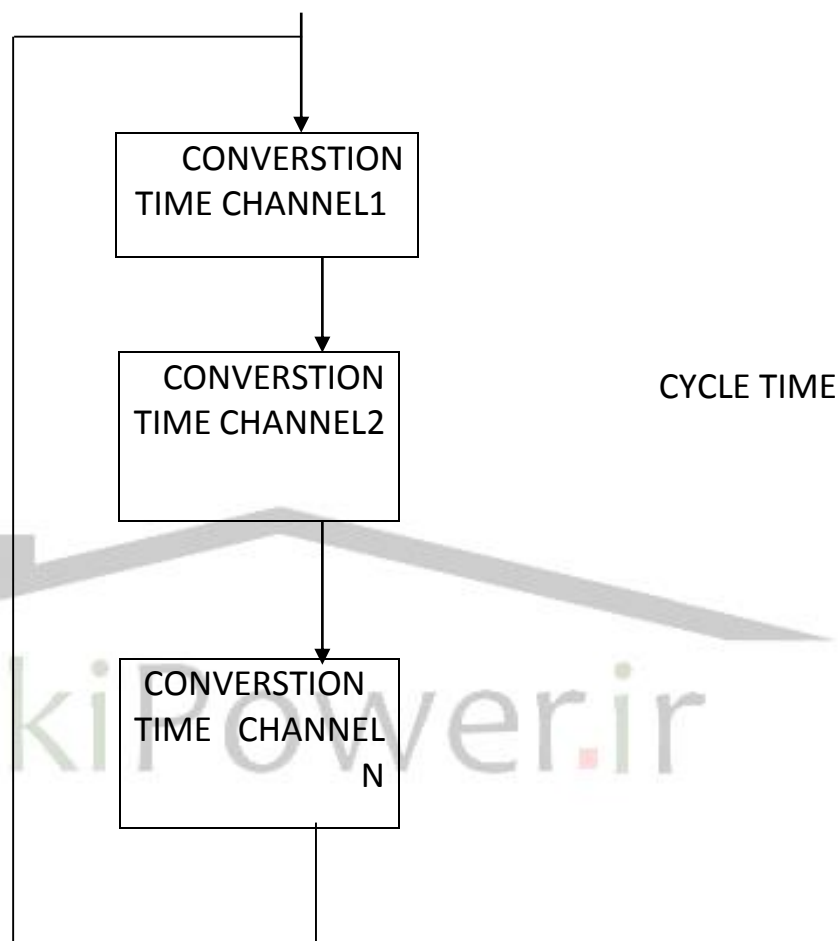
وضعیت مدول	نوع سیگنال
A	ولتاژ MV
B	ولتاژ V
A	ترموکوپل
D	جریان از سنسور دو سیمه
C	جریان از سنسور چهار سیمه
A	ترموتر

نحوه خواندن سیگنال های آنالوگ ورودی توسط PLC :

بطور کلی همه مقادیر آنالوگ اعم از ولتاژ و جریان و مقاومت ابتدا در کارت AI از آنالوگ به دیجیتال تبدیل میشوند سپس توسط CPU برای پردازش مورد استفاده قرار میگیرد بنابراین برای PLC مقدار ۱۰ یا به عبارت دیگری فرمت باینری یا HEX مفهوم دارد .

برای دریافت فایل Word پروژه به سایت ویکی پاور مراجعه کنید. فاقد آرم سایت و به همراه فونت های لازم

برای هر مدول آنالوگ یک CYCLE TIME وجود دارد ابتدا در کانال یک تبدیل صورت میگیرد سپس کانال ۲ و نهایتا کانال N. کل زمانی که این کار طول میکشد را CYCLE TIME میگویند. اگر کانال یا کانال هایی از کارت در عمل استفاده نشده باشد باید در پارامتر های کارت انرا غیر فعال کنیم. این کار منجر به کاهش زمان سیکل فوق خواهد



برای دریافت فایل Word پروژه به سایت ویکی پاور مراجعه کنید. فاقد آرم سایت و به همراه فونت های لازم

پس از تبدیل مقدار باینری بصورت ۱۶ بیتی است بیت آخر سمت چپ علامت عدد را نشان میدهد
 * برای مثبت و ابرای منفی است.

برای هر نوع سیگنال جدول مشخصی وجود دارد که معادل هگز و دسیمال ورودی مربوط به کارت را نشان میدهد در این جداول علاوه بر رنج نرمال مقادیر بالای رنج زیر رنج overflow, underflow نیز داده شده اند بعنوان مثال فرض کنید به کارت ورودی سیگنال 4-20ma متصل شده است plc مقادیر میلی آمپر را نمایش میدهد و در برنامه نویسی نمیتوان این مقادیر را مستقیماً به کار برد. ابتدا باید با توجه به کاتالوگ معادل هگز یا دسیمال آنرا پیدا کرد سپس در برنامه از آنها استفاده نمود. بعنوان مثال مطابق جدول روبرو میبینیم که معادل دسیمال ۴ میلی آمپر عدد صفر و معادل دسیمال ۲۰ میلی آمپر عدد ۲۷۶۴۸ میباشد این مقادیر از نظر plc مفهوم دارد

Measuring range	units	hexadecimal	range
4to20ma	decimal	hexadecimal	
22.810	32767	7fff	Over flow
22.810	32511	7effh	Over range
.20.0005	.	.	
	.	.	
	27649	6c01h	
20.000	27648	6c00h	Nominal range
16.000	20736	5100h	
.	.	.	
.4.000	.	.	
	0	oh	
3.9995	-1	Ffffh	Under range
.	.	.	
.11852	.	.	
	-4864	Ed00	
<1.1852	-32768	8000h	Under flow

برای دریافت فایل Word پروژه به سایت ویکی پاور مراجعه کنید. فاقد آرم سایت و به همراه فونت های لازم

مقایسه نحوه تبدیل سیگنال آنالوگ در S7, S5 :

شاید برای کاربرانی که با plc های سری S5 زمینس آشنایی دارند مقایسه بین نحوه تبدیل آنالوگ در S7, S5 خالی از فایده نباشد. همانطور که در عنوان گردید در S7 سر ریزی توسط مقدار سیگنال که از محاسبه ارزش ۱۶ بیت بدست میآید چک شود ولی در S5 یکی از بیت ها برای overflow رزرو شده که اگر یک شد نشان میدهد مقدار سیگنال سرریز شده است بطور کلی در S5 سه بیت برای مقاصد زیر رزرو شده است .

بیت ۰ یعنی ۰ برای over flow رزرو شده است .

بیت ۱ یعنی e برای error bit رزرو شده. مثلا برای کارت با قابلیت خاص اگر این بیت اشد نشان دهنده wire break است

بیت ۲ یعنی a برای activity bit رزرو شده است که نشان دهنده valid یا invalid بودن مقدار است

نتیجه اینکه رزرو شدن سه بیت در S5 منجر شده که resolution حداکثر ۱۲ بیتی با علامت باشد . در حالتیکه در S7 این بیت ها آزاد است و resolution حداکثر ۱۵ بیتی با علامت میباشد.

تنظیم پارامتر های کارت های AI:

کارت های INPUT ANALOG را میتوان به صورت زیر دسته بندی نمود

از نظر تعداد ورودی	از نظر نوع سیگنال	از نظر قابلیت های خاص
۲ ورودی	ولتاژ/جریان	بدون ویژگی خاص
۴ ورودی	مقاومت/TC	ایجاد وقفه
۸ ورودی	RTD / ترکیبی از موارد فوق	تشخیص قطعی

برای دریافت فایل Word پروژه به سایت ویکی پاور مراجعه کنید. فاقد آرم سایت و به همراه فونت های لازم

پس از وارد کردن هر کارت AI و کلیک روی آن پنجره ای که ویژگیهای آن را نشان میدهد باز میشود با توضیحاتی که در مورد کارت های دیجیتال داده شد بخش های ADDRESS و GENERAL روشن است و نیاز به شرح ندارد. صرفاً این نکته را باید افزود که برای هر ورودی آنالوگ ۲ بایت آدرس یا به عبارت دیگر یک WORD اختصاص داده میشود. بخش INPUT در تمام کارت های AI وجود دارد و برای انتخاب نوع سیگنال بکار میرود. بعنوان مثال حتی اگر کارت فقط ولتاژ را قبول کند باز لازم است رنج آن را تعیین نمود در عین حال اگر کارت ویژگی خاص برای اعمال وقفه داشته باشد این پارامترها در بخش INPUT قابل توجه است وضعیت تنظیم سخت افزاری مدول MEASURING RANGE میباشد که وضعیت A یا B یا C یا D آنرا مشخص میکند انتخاب نوع سیگنال برای هر کانال با کلیک کردن در جلوی سطر MEASURING TYPE انجام میشود

بسته به نوع کارت ممکن است انتخاب های متفاوتی داشته باشیم مثلاً در کارتی که مختص ترموکوپل است سایر سیگنال ها ظاهر نمیشود در بخش measurement type گزینه deactivated که در پارامتر تمامی کارت ها ظاهر میشود برای غیر فعال کردن کانال است طبق توضیحاتی که قبلاً داده شد اگر کانالی عملاً استفاده نشود برای اینکه زمان سیکل تبدیل آنالوگ به دیجیتال را کاهش دهیم این کانال را deactivate میکنیم.

پس از انتخاب measuring type در زیر آن اگر روی measuring range کلیک کنیم لیستی که انواع رنجهای قابل کاربرد برای آن سیگنال را در بر دارد نمایش داده میشود برخی از کارت های AI صرفاً برای ورودی خاص مانند ترموکوپل یا ترمیتور طراحی شده اند. در این کارت ها در بخش MEASURING RANGE تنوع بیشتری را مشاهده میکنیم شکل زیر انواع ترمیتور در کارت مخصوص RTD را نشان میدهد.

Pt100cl,	pt100climatic range
Pt200cl,	pt200climatic range
Pt500cl,	pt500climatic range
Pt1000cl,	pt1000 climatic range
Pt100std,	pt100 standard range
Pt200std,	pt200 standard range
Pt 500 std,	pt500standard range
Pt1000 std,	pt1000 standard range

برای دریافت فایل Word پروژه به سایت ویکی پاور مراجعه کنید. فاقد آرم سایت و به همراه فونت های لازمه

Ni 100 cl,	ni 100 climatic range
Ni 120cl,	ni120climatic range
Ni200cl,	ni200climatic range
Ni500cl,	ni500climatic range
Ni1000 cl,	ni1000climatic range
Ni100 std,	ni100standard range
Ni120 std,	ni120standard range
Ni200std,	ni200 standard range
Ni500std,	ni500 standard range
Ni1000std,	ni1000standard range
Cu10cl,	cu10climatic range
Cu10std,	cu10 standard range

و شکل زیر انواع ترموکوپل در کارت خاص TC را نشان میدهد بعلاوه در این کارت ها پارامترهای خاص دیگری مانند واحد اندازه گیری دما و ضرایب استاندارد برای RTD و امثال آنرا میتوان یافت

TC-EL

TYPE B [PTRH-PTRH]
 TYPE N [NICRSI-NISI]
 TYPE E [NICR-CUNI]
 TYPE R [PTRH-PT]
 TYPE S [PTRH-PT]
 TYPE J [FE-CUNI]
 TYPE L [FE-CUNI]
 TYPE T [CU-CUNI]
 TYPE K [NICR-NI]
 TYPE U [CU-CUNI]
 TYPE C [W5RE-W26RE]

قابلیت های خاص کارت های AI :

مهمترین قابلیت های خاصی که در برخی کارت های AI وجود دارد عبارتند از :

برای دریافت فایل Word پروژه به سایت ویکی پاور مراجعه کنید. فاقد آرم سایت و به همراه فونت های لازم

: DIAGNOSTICS INTERRUPT

در صورت فعال شدن برای حالت هایی مانند WIRE BREAK و قطع تغذیه شبیه کارت های DI ایجاد وقفه میکند .

: HARDWARE INTRRUPT

در صورت فعال شدن در صورتی که سیگنال از حدود تعیین شده تجاوز نماید ایجاد وقفه مینماید برای کارت AI 8*TC که خاص ترموکوپل است پارامتر دیگری با عنوان REACTION TO OPEN THERMOCOUPLE وجود دارد . برای پروسه های گرمایشی باید UNDER FLOW در جلوی این پارامتر انتخاب شود . تا در صورت بتز بودن ترموکوپل مقدار ۸۰۰۰ توسط کارت برگردانده شود و لوپ کنترلی به صورت اتوماتیک گرما را کاهش دهد (وضعیت ایمن تر) .

برای پروسه های سرمایشی باید UNDERFLOW در جلوی این پارامتر انتخاب شود تا در صورت باز بودن ترموکوپل مقدار ۸۰۰۰ توسط کارت برگردانده شود و لوپ کنترلی بصورت اتوماتیک سرما را کاهش دهد . (وضعیت ایمن تر).

گزینه دیگری که در بسیاری از کارت های AI وجود دارد INTEGRATION TIME است . این پارامتر در واقع RESOLUTION را مطابق با جدول زیر تعیین میکند

RESOLUTION	INTERFERENCE FREQUENCY	INTEGRATION TIME (MS)
9+SIGN BIT	400	2.5
12+SIGN BIT	60	16.5
12+SIGN BIT	50	20
14+SIGN BIT	10	100

تنظیم پارامتر های کارت های AO :

برای دریافت فایل Word پروژه به سایت ویکی پاور مراجعه کنید. فاقد آرم سایت و به همراه فونت های لازم

کارت های ANALOG OUTPUT را میتوان بصورت زیر دسته بندی کرد .

از نظر تعداد خروجی	از نظر نوع سیگنال	از نظر قابلیت های خاص
۲ ورودی	ولتاژ	بدون ویژگی خاص
۴ ورودی	جریان	تشخیص قطعی
۸ ورودی	ترکیب دو مورد فوق	تشخیص اتصال کوتاه واکنش به توقف CPU

ویژگی این کارت ها در بخش های GENERAL, ADDRESS نیاز به توضیح ندارد و آنچه برای کارت های AI گفته شد در اینجا نیز صادق است از اینرو صرفا در مورد بخش OUTPUT مطالبی بیان میشود همانطور که در جدول بالا آمده است خروجی این کارت ها یا از جنس جریان است یا از جنس ولتاژ. برای هر یک از این دو حالت رنج های مختلفی در جلوی output range قابل انتخاب است بجز I, E گزینه سومی که برای انواع خروجی وجود دارد deactivated است. همانطور که قبلا برای آنالوگ توضیح داده شد برای خروجی های آنالوگ نیز یک سیکل تبدیل وجود دارد در این سیکل مقادیر باینری تولید شده توسط cpu به مقادیر متناسب با جنس خروجی تبدیل میشوند. از ویژگی های خاص این کارت diagnostic interrupt است که اگر فعال شود و سپس group diagnostics برای کانال مورد نظر علامت بخورد در مواردی مانند اتصال زمین (برای خروجی های ولتاژ) و اتصال کوتاه (برای خروجی های جریان) و عدم وجود تغذیه وقفه اعمال میکند. واکنش به قطع شدن cpu پارامتر دیگری است که قابل تنظیم است ocv یعنی خروجی صفر شود و klv یعنی آخرین مقدار قبلی حفظ گردد و sv یعنی خروجی برابر با مقدار مشخصی که در زیر آن تعیین میشود. تنظیم این پارامتر با توجه به پروسه و موارد ایمنی باید انجام گردد.

تنظیم پارامتر های AI/AO :

برای دریافت فایل Word پروژه به سایت ویکی پاور مراجعه کنید. فاقد آرم سایت و به همراه فونت های لازم

این کارت ها ترکیبی از ورودی و خروجی را شامل میشوند . با توجه به توضیحاتی که تا کنون ارائه شده با آرمتر های آنها نیاز به شرح ندارد برای این کارت ها دو بخش **input, output** وجود دارد که برای هر کدام میتوان نوع ورودی و خروجی را مشخص نمود و سایر پارامتر ها را تنظیم کرد .
تنظیم پارامتر های کارت های **special 300** :

به جز کارت های **I/O** در زیر مجموعه مدل های **SM** از **s7300** کارت های دیگری را تحت عنوان **special300** میتوان یافت . که یکی از آنها **dummy** نام دارد این کارت دارای هیچ ورودی یا خروجی خاصی نیست و همانطور که از نامش پیدا است یک مدول مجازی یا قلبی است این مدول میتواند همانند سایر کارت های **I/O** در اسلات ۴ تا ۱۱ قرار بگیرد و کاربرد آن صرفا برای رزرو کردن محل و آدرس ان اسلات برای استفاده در آینده است اگر کاربر بخواهد در بین کارت های **i/o** در اسلاتی را برای استفاده در آینده رزرو کند از آنجا که نباید بین کارت ها در اسلات فاصله باشد این کارت خاص دیگری که در این گروه وجود دارد **sm338** است که برای ارتباط با انکودر در نظر گرفته میشود
تنظیم پارامتر های **cpu** :

Cpu های ۳۰۰ انواع مختلفی دارند که توانایی آنها متفاوت است در جدول زیر مقایسه برخی پارامتر های اصلی **cpu** های غیر کمپکت این خانواده ارائه شده است توصیه میشود در فاز طراحی قبل از انتخاب **cpu** از یک طرف نباید دست بالا یا باصطلاح **over design** عمل کرد و از طرف دیگر نباید آنرا دقیقا مطابق با نیاز موجود انتخاب نمود . بطوریکه امکان توسعه در آینده وجود نداشته باشد

برای دریافت فایل Word پروژه به سایت ویکی پاور مراجعه کنید. فاقد آرم سایت و به همراه فونت های لازم

Cpu318-2dp	Cpu315-2dp	Cpu314	Cpu312	
256kb	128kb	48kb	16kb	Ram
512	256	256	128	Counter
512	256	256	128	Timer
8192byte	2048byte	256 byte	128byte	Bit memory
65536byte input/output	28byte	128byte	128 byte	Digital channel
4096byte input/output	16384 byte	1024byte	256byte	Analog channel
Max.4	Max.4	Max.4	Max .1	rack
yes	yes	no	no	Profibus dp

نکات زیر در موقع وارد کردن cpu به اسلات دوم در برنامه hwconfing قابل توجه است :

1. cpu هایی که در انتهای کد آنها 2dp نوشته شده مستقیماً دارای پورت ارتباط با شبکه پروفی باس میباشند. در موقع وارد کردن آنها در اسلات پنجره ای ظاهر میشود که میتوان از آنجا با کلیک کردن روی new شبکه ارتباطی را مشخص کرد پس از وارد کردن cpu در اسلات می بینیم که یک سطر بنام dp در زیر آن ظاهر میشود .
 2. cpu های کمپکت که انتهای کد آنها حرف C نوشته شده وقتی وارد اسلات میشوند در زیر آنها مدول های متصل به cpu نیز مانند شکل زیر ظاهر میگردند. در واقع مجموعه cpu و مدول های متصل به آن همگی در اسلات دوم قرار میگیرند. تنظیم ورودی و خروجی های این بخش نیز شبیه ورودی خروجی های کارت I/O انجام میشود ولی در اینجا مدول سخت افزاری وجود ندارد پس از وارد کردن cpu در اسلات با کلیک کردن روی آن پنجره ای که پارامترهای آن را نمایش میدهد ظاهر میگردد. بسته به نوع cpu ممکن است برخی بخش ها موجود نباشند یا در برخی بخش ها گزینه هایی فعال نباشند
- مد های کاری plc :

: Stop

در این مد پردازش برنامه متوقف میشود دسترسی به I/O ها وجود ندارد CPU بصورت read, write قابل دسترسی است یعنی میتوان برنامه آنرا خواند و یا برنامه جدیدی را به آن انتقال داد .

: Run

برای دریافت فایل Word پروژه به سایت ویکی پاور مراجعه کنید. فاقد آرم سایت و به همراه فونت های لازم

در این مد برنامه اجرا میشود cpu به I/O ها دسترسی دارد برنامه cpu به صورت read only است یعنی نمی توان برنامه جدیدی را به آن download کرد
: Run-p

در این مد برنامه اجرا میشود. cpu به I/O دسترسی دارد در عین حال cpu بصورت read,write قابل دسترسی است .

تذکر:مد های فوق توسط سوئیچ روی cpu نیز قابل انتخاب است در برخی از cpu ها این سوئیچ حالت run-p ندارد .

: Mers

این وضعیت برای ریست کردن حافظه cpu بکار میرود یعنی هم مقادیر متغیر های حافظه و هم برنامه ای که توسط کاربر به حافظه ارسال شده پاک میگردند جزئیات بیشتر راجع به حافظه cpu در بخش بعد آمده است .

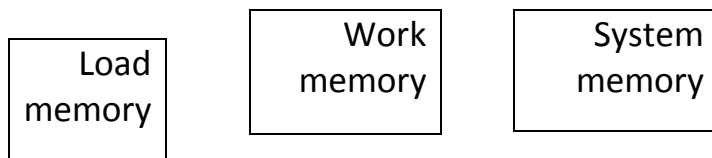
برای ریست کردن cpu باید سوئیچ بین وضعیت mors,stop جابجا شود

۱. سوئیچ در وضعیت stop است و led مربوط به stop روشن است .

۲. سوئیچ را از stop به mors میبریم و ۳ ثانیه نگه میداریم و مجدداً به stop بر میگردانیم

۳. با کمی مکث (حداکثر ۳ ثانیه) سوئیچ را از stop دوباره به mors میبریم. led فوق به حالت چشمک زن سریع در میآید. حالت چشمک زن فوق نشان دهنده این است که حافظه cpu ریست شده است اگر در این مرحله led به حالت چشمک زن در نیامد باید مراحل فوق از اول تکرار شود حافظه cpu های s7-300 :

به طور کلی حافظه cpu های s7 (اعم از ۳۰۰ و ۴۰۰) ساختاری شبیه شکل زیر دارند .



: Load memory

وقتی برنامه به cpu ارسال میشود در این قسمت وارد میگردد.

برای دریافت فایل Word پروژه به سایت ویکی پاور مراجعه کنید. فاقد آرم سایت و به همراه فونت های لازم

: Work memory

این حافظه بخشی از برنامه که اجرایی است را در بر میگیرد. به عنوان مثال یک فانکشن فقط در زمانی که صدا زده میشود

: System memory

این بخش عناصر حافظه مانند جداول PIQ, PII. و فلگ ها و تایمر ها و کانتر ها و ... را در بر میگیرد. در صورت ریست شدن CPU کل محتویات بخش های WORK MEMORY, SYSTEM MEMORY پاک میشود بخش LOAD MEMORY نیز بسته به نوع آن ممکن است پاک شود. طراحی بخش های فوق در CPU های مختلف متفاوت است در برخی CPU ها مانند ۳۱۸-2DP حافظه LOAD MEMORY از نوع RAM یا EPROM است اکثر CPU های جدید ۳۰۰ فاقد LOAD MEMORY داخلی هستند و یک کارت حافظه بیرونی بنام MMC که مانند شکل زیر در اسلات CPU قرار میگیرند بعنوان میگیرند بعنوان LOAD MEMORY آنها تلقی میگردد بدون این کارت CPU راه اندازی نمیشوند. قبلا طراحی به صورتی بود که حافظه CPU علاوه بر داشتن LOAD MEMORY داخلی میتواند توسط یک کارت حافظه MEMORY CARD از جنس RAM یا FLASH EPROM افزایش پیدا کند در طراحی جدید در برخی موارد MMC یا MASTER MEMORY CARD که از نوع FEPRM است جایگزینی کارت های قبلی و LOAD MEMORY داخلی شده است سائز MMC ها متفاوت بوده و کاربر میتواند بسته به نوع نیاز آنها را سفارش دهد. بعنوان مثال در CPU315-2DP سائز این کارت میتواند تا 8MB باشد.

انواع راه اندازی

سه نوع راه اندازی برای CPU های S7 وجود دارد :

۱. COLD RESTART
۲. WARM RESTART
۳. HOT RESTART

COLD RESTART

تمامی تایمر ها و کانتر ها و فلاگ ها ریست میشوند چه از نوع قابل ذخیره باشند چه نباشند برنامه از اولین دستور OB1 اجرا میگردد.

برای دریافت فایل Word پروژه به سایت **ویکی پاور** مراجعه کنید. فاقد آرم سایت و به همراه فونت های لازم

: Warm restart

آنچه به عنوان retentive تعریف شده پاک نمیشود. برنامه از اولین ob1 اجرا میگردد.

: Hot restart

هیچ دیتایی پاک نمیشود چه از نوع retentive باشد چه نباشد .

برنامه از جایی که قطع شده بود ادامه می یابد. خاص s7-400 است. Cpu در این حالت باید باتری

backup دارد.

مد hold: در این مد پردازش برنامه متوقف شده و کاربر میتواند برنامه را قدم به قدم توسط pg یا

pc تست کند این مد کاری برای تست و عیب یابی برنامه به ویژه در زمانی که کار می رود که برنامه

نویسی از نظر دستورات صحیح است ولی به علت وجود اشکال در برنامه منطبق برنامه جواب برنامه

مورد نظر بدست نمی آید .

اولویت های مدهای کاری cpu :

اگر چند مد بطور همزمان درخواست شود مدی که دارای اولویت بالاتری است انتخاب میشود بعنوان

مثال اگر سوئیچ cpu روی run قرار گیرد و همزمان از طریق pg مد stop می رود زیرا اولویت

بالاتری دارد. این اولویت ها در جدول زیر آمده است.

اولویت	مد کاری
بیشترین	stop
	holt
	Startup
کمترین	run

برای دریافت فایل Word پروژه به سایت ویکی پاور مراجعه کنید. فاقد آرم سایت و به همراه فونت های لازم

مدول های fm :

Function modules مدول هایی هستند که فانکشن خاصی را مستقل از cpu اجرا میکند و به اصطلاح باری را از دوش cpu بر میدارند. ورودیها مستقیماً به مدول ها داده میشوند و خروجیها نیز مستقیماً از آنها ارسال میشود در عین حال fm ها میتوانند با cpu تبادل دیتا داشته باشند. در برخی کاربردها مانند شمارش سریع که امکان آن توسط cpu وجود ندارد چارهای جز استفاده از مدول های fm نیست. برای تنظیمات مربوط به فانکشن داخلی fm که توسط آنها پارامترها به fm اختصاص داده میشوند عرضه میشوند. اگر پس از وارد کردن fm در hwconfig و ذخیره سازی به برنامه simatic manager باز گردیم میبینیم که آیکون fm در پنجره ظاهر میشود و دارای پوشه و بلاک های جداگانه ای است. بلاک های خاص fm در این پوشه قرار میگیرند با وارد کردن fm مورد نظر در اسلات ۴ تا ۱۱ از رک ۳۰۰ و سپس کلیک روی آن پنجره که باز میشود که دو بخش ان یعنی بخش general که توضیحات کلی راجع به مدول را در بردارد و بخش ادرس در همه مشترک هستند آدرس هایی که در بخش parameters basic موسوم است در این بخش از جمله میتوان نوع وقفه که توسط fm به cpu اعمال میشود را تنظیم کرد. وقفه از نوع diagnostic در صورت وجود اشکال مانند قطع اتصال کوتاه اعمال میکند. وقفه از نوع hardware interrupt وقتی مقدار actual از مقدار رفرنس که در پارامترهای مدل مینما ید. واکنش به توقف COU نیز در برخی قاب تنظیم است. پارامترهای دیگری نیز بطور خاص برای fm ها وجود دارد.

در 300_7s فا نکشن مدل های مختلف وجود دارد که برخی از آنها اشاره میکنیم:

Fm350_1(counter modul):

این مدل کاتر یک کا ناله ای است که برای شمارشهای ساده بکار میرود. انکودرهای افزایشی (incremental) را میتوان مستقیماً" به این مدول های متصل کرد این مدول قادر است پالس های را از انکودر با فرکانس ماکزیمم 500khz دریافت کند. fm350_1 دارای مدهای کاری مختلف است مانند مد شمارش پریودیک و مد شمارش یک نوبته .

Fm350_2(counter module):

برای دریافت فایل Word پروژه به سایت **ویکی پاور** مراجعه کنید. فاقد آرم سایت و به همراه فونت های لازم

این مدول کانترا ۱۸ کاناله ای است که برای شمارش های یونیورسال بکار می رود می توان به انکودرهای افزایشی incremental متصل گردد پالسهای با فرکانس ماکزیمم 10khz از آنها بگیرد. همچنین می توان به برخی سنسور های خاص با فرکانس سیگنال حد اکثر 20khz وصل کرد .

Fm351 (position module):

این مدل برای کنترل موقیت تراورس های سریع و درایو های لرزشی بکار میرود . دارای ۴ خروجی دیجیتال برای کنترل موتور میتواند توسط کنتاکتر یا درایو تغذیه میشود.

Fm353 (position module):

این مدول برای کنترل موقعیت موتورهای پله ای بکار میرود

Fm354 (position module):

این مدول برای کنترل موقعیت سرورموتورها بکار میرود.

Fm352 (electronic com controller):

این مدول برای زشی زخده فخم بکار میرود . سرعتش خیلی بالاست . موقعیت را از طریق سنسورها که به ورمدی آن متصل است دریافت میکند سپس فرمانهای لارم را برای کنترل ماشین میفرستند. از کاربردهای آن میتوان به نوار نقاله ای که قطعات روی آن حرکت کرده و در جولوی ماشین دریل یا سنگ قرار میگیرند

Fm355 (closed controller):

این مدول یک لوپ کنترلر است که میتوان برای کنترل فشار و دما بکار میرود . اگر بصورت continues control تنظیم شود از ۴ خروجی آنالوگ آن استفاده میشود و اگر بصورت step control تنظیم شود ۸ خروجی دیجیتالی آن بکار برده میشود.

fm350-1 پیکر بندی مدول

همانطور که اشاره شد این مدل یک شمارنده سریع است که میتواند به انکودر وصل شده و عمل شمارش رامستقل از cpu انجام دهد . علاوه بر وروری انکودر این مدول دارای ورودی دیجیتالی

برای دریافت فایل Word پروژه به سایت ویکی پاور مراجعه کنید. فاقد آرم سایت و به همراه فونت های لازم

دیگری است آن میتوان عمل شمارش را کنترل نمود. این کنترل از طریق فانکشن های خاص که در برنامه cpu صدا زده میشود امکان پذیر است. برای تنظیم پارامترهای 1-350fm نرم افزارهای استاندارد step7 کافی نیست و لازم است پکیج پیکر بندی مربوطه نیز روی آن نصب شود. این پکیج همواره با کارت fm توسط فروشنده عرضه میگردد در عین این حال به صورت آزاد از سایت قابل download است. پس از نصب پکیج step7 را اجرا کرده و در hwconfih مدول 1-350fm را در اسلات ۴ تا ۱۱ قرار میدیم سپس با دوبار کلیک روی آن مشاهده میکنیم که برنامه جدیدی اجرا شده. با کلیک کردن روی هر یک از باکسهای فوق پنجره جدیدی باز میشود.

با کلیک کردن روی باکس encodres پنجره ی باز میشود. بسته به نوع انکوردی که سمت چپ انتخاب میشود گزینه های سمت راست را فعال یا غیر فعال میکند.

با کلیک روی باکس operating mode باز میشود. در مد continuous اگر کانتر افزایشی به حد ماکزیمم خود برسد و باز پالس شمارش دریافت شود کانتر به حد مینیمم خود پرش کرده و آنجا شروع به افزایش میکند در کانتر کاهشی عکس عمل فوق اتفاق می افتد یعنی وقتی به حد مینیمم خود رسید با پالس جدید به حد ماکزیمم پرش و از آنجا شروع به کاهش میکند. در مد single continh اگر کانتر به حد ماکزیمم خود برسد و باز پالس شمارش دریافت شود کانتر افزایشی به نقطه مینیمم و کانتر کاهشی به نقطه ماکزیمم رفته و آنجا میماند. مد periodic conting شبیه continuous است با این تفاوت که نقطه شروع از مقدار load value است.

مدلهای cp:

این مدل ها همانطور که از نامشان (communication processor) پیداست برای ارتباط با شبکه به کار میروند در پنجره ی کاتالوک از نوع کارتها cp در چند دسته قرار گرفته اند.

منبع تغذیه ps

منبع تغذیه های ۳۰۰ همگی ۲۳۰/۱۲۰ ولتی هستند که بسته به جریان آنها به سه دسته 2a و 5a و 10a تقسیم میکنند. این ککاراتر ها هیچ تنظیم خاصی که لازم باشد توسط hwconfig نیاز ندارد

برای دریافت فایل Word پروژه به سایت ویکی پاور مراجعه کنید. فاقد آرم سایت و به همراه فونت های لازم

پایان پیکربندی و چک سازگاری اجزا:

آخرین مرحله چک کردن اجزا است. این کار با منوی `consistency check` انجام میشود اگر اشکالی وجود داشته باشد نمایش داده میشود. پس از رفع اشکال و انجام چک مجدد پیغام `noerror` ظاهر میشود. در این مرحله باید تغییرات را ذخیره کنیم منوی `file>save` فقط ذخیره سازی را انجام میدهد و منوی `faile>save nad compile` علاوه بر ذخیره کردن عمل کامپایل و چک سازگاری نیز انجام میدهد. کامپایل کردن منجر به ایجاد به آیکونی به نام `system data` میگردد که اطلاعات سخت افزار پیکر بندی را در بر دارد آخرین مرحله پس از ذخیره سازی دانلود کردن به `plc` است.

۳-۴ پیکر بندی s7-400

وارد کردن `simatic manger` در `statino`

- باز کردن `station` توسط برنامه `hwconfig`

- وارد کردن رک

- گذاشتن اجزا مورد نیاز در رک از کاتالوگ ۴۰۰

- تنظیم پارامترهای مدولها

- چک سازی و ذخیره سازی

تفاوتهای که پیکر بندی `s7-300` با `s7-400` دارد عبارتند از

۱. رک های ۴۰۰ چندین نوع دارد و فانکشن آنها با ۳۰۰ متفاوت است
۲. تعداد رک اضافی که میتوان به رک اصلی متصل شود بیش از این تعداد در ۳۰۰ است
۳. فضای خالی بین مدلها در رک ۴۰۰ اهمیت ندارد
۴. کارت `ai/ao` و `di/do` در `s7-400` وجود ندارد
۵. تنوع کارتهای ۴۰۰ نسبت به ۳۰۰ کمتر
۶. تمام `cpu`های ۴۰۰ تغییر ادرس مدلهای ورودی و خروجی را ساپورت میکند
۷. قابلیت `multicomputing` در `cpu` های ۴۰۰ وجود دارد و میتوان چند `cpu` از این نوع را در یک رک وارد کرد
۸. علاوه بر راه اندازی های `cold, warm` مد `hot startup` نیز برای `cpu` های ۴۰۰ وجود دارد

برای دریافت فایل Word پروژه به سایت ویکی پاور مراجعه کنید. فاقد آرم سایت و به همراه فونت های لازمه

۹. تواناییها و سرعت و حجم ادرسی که cpu های ۴۰۰ دارند بسیار بیشتر از ۳۰۰ است با توجه به توضیحات فوق موارد خاصی که نیاز به توضیح بیشتر دارند به شرح زیر ارائه میشود:

منبع تغذیه ps :

منبع تغذیه در S7-400 را میتوان طبق جدول زیر دسته بندی کرد .

ولتاژ	جریان	نوع
24V	10	Redundant
120/230V		
24V	4A	standard
120/230V	10A	
	20A	

میتوان ۲ منبع تغذیه از نوع REDUNDANT را در کنار هم در اسلات های اولیه قرار داد. اگر یکی از این منابع تغذیه قطع شود سیستم از دومی تغذیه میگردد لازم است برخی از منابع تغذیه عملا و در هنگام پیکر بندی ۲ اسلات را اشغال میکنند .

رکهای S7-400

رک های S7-400 وظایفی بشرح زیر دارند :

- نگهدارنده مدولهاست
- تغذیه کننده مدولها از طریق BACKPLANE BUS است
- دارای I/O BUS برای ارتباط سیگنالهاست
- دارای COMMUNICATION BUS برای ارتباط شبکه است

همانطور که در شکل روبرو نمایش داده شده است رک های S7-400 انواع مختلف دارند. انواع این رک ها در جدول زیر معرفی شده اند .

برای دریافت فایل Word پروژه به سایت ویکی پاور مراجعه کنید. فاقد آرم سایت و به همراه فونت های لازم

نام	نوع	کاربرد	تعداد اسلات	ملاحظات
UR1	UNIVERSAL	بعنوان رک اصلی بعنوان رک اضافی	۱۸	دارای دو نوع استکه در یک نوع آن میتوان ۲ منبع تغذیه از نوع REDUDANT قرار داد
UR2		"	۹	"
ER1	EXPANSION	بعنوان رک اضافی	۱۸	"
ER2	"	"	۹	"
CR2	SEGMENTED	بعنوان رک اصلی	۸+۱۰	دارای دو بخش است که هر بخش آن میتواند یک رک اصلی مستقل اشد
CR2-H	UNIVERSAL	برای PLC های REDUNDAN	2*9	شبيه دو رک اصلی UR2 است

رک اضافی در S7-400 :

قبل از پیکر بندی رک اضافی در S7-400 باید نکات زیر را مد نظر قرار داد

- در S7-400 ماکزیمم ۲۱ رک اضافی میتوانیم داشته باشیم

برای دریافت فایل Word پروژه به سایت ویکی پاور مراجعه کنید. فاقد آرم سایت و به همراه فونت های لازم

- در رک اصلی میتوان حداکثر ۶ عدد IM وارد کرد
 - IM ها در رک اصلی میتوانند از ۳ اسلات به بعد قرار گیرند
 - IM ها در رک اضافی فقط در آخرین اسلات قرار میگیرند
 - هر IM میتواند به شکل زنجیری حداکثر به چهار IM متصل شود
- IM ها انواع مختلفی دارند که در جدول روبر آمده است و بصورت جفتی بکار میروند میتوان ترکیبی از آنها را استفاده کرد .

اگر IM تغذیه را نیز منتقل کند دیگر نیازی به منبع تغذیه در رک اضافی نداریم
 بجز IM های فوق سایر موارد که در پنجره کاتالوگ ظاهر میشوند برای ارتباط با رک اضافی نیستند
 بعنوان مثال IM467 برای ارتباط با شبکه پروفی باس بکار میرود
 پیکر بندی رک اضافی :

روش وارد کردن رک اضافی شبیه آنچه برای S7-300 ذکر شد میباشد در S7-30 تعداد رک اضافی محدود بود و قرار دادن IM ارتباط توسط برنامه برقرار میشود ولی در S7-400 بدلیل زیاد بودن IM ها و اینکه مشخص نیست به چه صورت باید گروه بندی و اتصال داده شوند اینکار بطور دستی توسط کاربر باید انجام گیرد برای این منظور با موس روی هر کدام از IM های رک اصلی کلیک کرده پنجره های باز میشود شماره رک اضافی مورد نظر را انتخاب کرده و با کلید CONNECT آنرا به رک اصلی متصل میکنیم از آنجا که IM دارای ۲ پورت برای اتصال است که C1, C2 نامیده میشود میتوان به هر یک از آنها رک اضافی را به صورت زنجیری وصل نمود
 ترتیب مدولها در رک ۴۰۰ :

برایچیدن مدول ها در رک ۴۰۰ باید توجه داشت که:

- CPU فقط در رک اصلی قرار میگیرد .
- IM از نوع RECEIVE فقط در رک اضافی قرار میگیرد
- میتوان دو منبع تغذیه از نوع REDUNDANT را در رک هایی که این آرایش را ساپورت میکنند از اسلات اول و پشت سر هم قرار داد
- اگر CPU قابلیت MULTICOMPUTING رزا داشته باشد در این صورت میتوان چند CPU را در اسلات های مختلف رک قرار داد

برای دریافت فایل Word پروژه به سایت ویکی پاور مراجعه کنید. فاقد آرم سایت و به همراه فونت های لازم

- برخی منابع تغذیه و CPU ها در عمل و نیز در هنگام پیکر بندی دو اسلات را اشغال میکنند .
- با توجه به نکات فوق ترتیب مدول ها در رک S7-400 را نمیتوان شبیه رک S7-300 در قالب فرمولی ارائه داد همینقدر میتوان گفت که ابتدا VPS سپس CPU و پس از آنها سایر کارت ها قرار میگیرند
- پارامتر های CPU در S7-400 :
- اکثر پارامتر های CPU در S7-400 شبیه S7-300 میباشد و تنظیم آنها به همان نحو است نکات خاص و اضافی که باید مد نظر قرار داد عبارتند از :
- برخی CPU های ۴۰۰ دارای ۴ آکومولاتور هستند ولی در نوع ۳۰۰ فقط دو آکومولاتور وجود دارد
- بخش PROTECTION برای همه CPU های نوع ۴۰۰ وجود دارد
- در بخش های مربوط به وقفه تعداد OB ها بی که فعال هستند بیش از ۳۰۰ است
- در برخی CPU های ۴۰۰ قابلیت MOLTICOMPUTING وجود دارد که CPU های ۳۰۰ فاقد آن هستند

آشنایی با محیط S7:

پس از نصب S7 با کلیک روی start در desk top ویندوز، برنامه های نصب شده را می توان مشاهده کرد. بطور معمول کاربر فقط نیاز دارد که simatic manager را اجرا کند و سایر برنامه ها و زیر برنامه ها در صورت لزوم توسط آن فرا خوانده می شوند. با این وجود میتوان زیر برنامه ها را به صورت مجزا

برای دریافت فایل Word پروژه به سایت ویکی پاور مراجعه کنید. فاقد آرم سایت و به همراه فونت های لازم

نیز اجرا کرد. البته برخی زیر برنامه ها در لیست فوق ظاهر نمیشوند ولی در هنگام نصب step 7 professional برنامه های زیر اختیاری هستند که در صورت عدم انتخاب، نصب نشده و در لیست ظاهر نمی شوند

s7-scl

s7-pdiag

s7-plcsim

s7graph

بجز این موارد سایر برنامه ها به عنوان پکیج استاندارد s7 محسوب می شوند این پکیج دارای زیر برنامه هایی است که در زیرآمده است

Symbol editor: برای ایجاد و اصلاح سمبل هایی به کار می رود که در برنامه نویسی plc استفاده شده اند

Diagnostics: با این ابزار می توان وضعیت تک تک مدولهای plc را از نظر وجود یا عدم وجود خطا چک کرد

Programming language: برای برنامه نویسی با یکی از زبانهای stl, fbd, lad بکار می رود

Hardware configuration: برای پیکر بندی و اختصاص پارامتر به سخت افزار به کار می رود

Network configuration: ابزار ساختار بندی شبکه است با استفاده از آن می توان node های

ارتباطی و اتصالات را تنظیم نمود و به آنها پارامتر اختصاص داد

شروع کار با simatic manager :

Simatic manager برنامه اصلی است که کاربر نیاز به اجرای آن دارد. پس از نصب s7 آیکون این

برنامه روی صفحه desk top ظاهر می شود میتوان آنرا با استفاده از آیکون مذکور اجرا نمود با اجرای

simatic manager معمولاً پنجره wizard ظاهر می شود که توسط آن می توان بخشی از امور مورد

نیاز را انجام داد. انتخاب cpu و بلاک های برنامه نویسی می تواند با wizard دنبال شود. ولی بدلیل

جاء نبودن آن توصیه می شود که کاربران ضمن غیر فعال کردن wizard کار را به صورت دستی ادامه

دهند پس از cancel کردن wizard به محیط simatic manager وارد می شویم قدم های زیر باید

به ترتیب بر داشته شوند

قدم اول: ایجاد پروژه، قدم دوم: ایجاد station با ایجاد program

برای دریافت فایل Word پروژه به سایت ویکی پاور مراجعه کنید. فاقد آرم سایت و به همراه فونت های لازم

قدم سوم وارد کردن پروژه

منو های simatic manager :

file : برای ایجاد، باز کردن، ذخیره سازی، حذف، سازماندهی، آرشیو کردن پروژه استفاده می شود .

Edit : امکاناتی مانند copy,paste,cut,rename کردن و مشاهده properties المانها را دارد

Insert : برای وارد کردن المانهای سخت افزاری و شبکه و بلاک های برنامه نویسی به پروژه بکار می

رود

Plc : برای ارسال اطلاعات به plc و یا گرفتن اطلاعات از plc و مواردی از این قبیل استفاده می شود

View : نحوه نمایش object ها و فیلتر گذاری روی آنها از این منو امکان پذیر است

Options : کاربر توسط امکانات این منو می تواند تنظیمات پیش فرض برنامه را تغییر دهد

Window : در این قسمت نحوه نمایش پنجره simatic manager قابل انتخاب است

Help : راهنمای استفاده و بکار گیری simatic manager می باشد

پیکر بندی سخت افزار:

Hwconfig ابزار پیکر بندی سخت افزار :

با وارد کردن سخت افزار در پروژه ایجاد شده توسط simatic manager و کلیک کردن روی آن

آیکون **hardwar** در پنجره سمت راست ظاهر می شود فرض کنید **station** های مختلفی از جمله

station300 وارد پروژه کرده ایم در این صورت با کلیک کردن روی آیکون **hardwar** در پنجره سمت

راست برنامه **hwconfig** اجرا می شود کلیه کارهای پیکر بندی سخت افزار و تنظیم پارامتر های آن

توسط این برنامه انجام می شود بطور کلی برای پیکر بندی سخت افزار باید قدم های زیر را به ترتیب بر

داشته شود

ایجاد **station**

اجرای برنامه **hwconfig**

انتخاب رک (**rack**)

آرایش دادن مدولها در رک

تنظیم پارامترهای مدولها

چک سازی اجزا و ذخیره پیکر بندی

برای دریافت فایل Word پروژه به سایت ویکی پاور مراجعه کنید. فاقد آرم سایت و به همراه فونت های لازم

پیکر بندی plc از نوع s7-300 :

Station300 رادر simatic manager وارد پروژه کرده سپس با کلیک روی آیکون hardwar آنرا با برنامه hwconfing باز می کنیم از پنجره کاتالوگ این برنامه در زیر گروه simatic 300 اجزای سخت افزاری را وارد پروژه می نماییم اولین قدم وارد کردن رک است با دو بار کلیک روی آیکون رک اسلاتهای آن در پنجره سمت چپ ظاهر می شوند

ویژگی های رک ۳۰۰:

یارده اسلات دارد

بصورت ریل است

فقط یک نوع دارد که هم به عنوان رک اصلی و هم به عنوان رک اضافی استفاده می گردد فقط نقش نگهدارندگی برای مدول ها را دارد مدولها باید روی آن کنار یکدیگر و بدون فاصله قرار گیرند ترتیب مدول ها در رک ۳۰۰: روی رک نمی توان هر مدول را در اسلات دلخواهی قرار داد در هنگام پیکر

بندی با hwconfing

باید ترتیب ذکر شده در زیر رعایت شود

مدولهای مجاز	شماره اسلات
ps	1
Cpu	2
l m یا خالی	3
Sm,cp,f m	4-11

ابتدا روی اسلات خالی با موس کلیک کرده سپس مدول را از پنجره کاتالوگ انتخاب می کنیم با دو بار کلیک روی آن می بینیم که مدول در اسلات ظاهر می شود اگر کاربر مدولی را اشتباها در اسلات غیر مربوط قرار دهد با پیغام خطا توسط برنامه مواجه می شود

استفاده از رک اضافی:

برای دریافت فایل Word پروژه به سایت ویکی پاور مراجعه کنید. فاقد آرم سایت و به همراه فونت های لازم

رک اصلی را central و رک اضافی را expansion می گویند. اگر تعداد i/o ها زیاد باشد و رک اصلی پر شود از رک اضافی استفاده می شود کابلی که رک اضافی را به رک اصلی وصل می کند طول محدودی دارد بنا براین رک اضافی نمی تواند زیاد دور از رک اصلی قرار بگیرد برای استفاده از رک اضافی در s7-300 نکات زیر مد نظر قرار گیرد

ماگزیمم ۳ رک اضافی به رک اصلی می توان وصل کرد. ارتباط بین رک اصلی و رک اضافی توسط im انجام می شود. im ها چه در رک اصلی چه در رک اضافی همیشه در اسلات سوم قرار می گیرند. im ها به صورت جفتی بکار می روند یعنی یکی از دو حالت زیر :

1: I m 360s: در رک اصلی و I m 361r در رک های اضافی (ماگزیمم ۳ رک)

2: I m 365 در رک اصلی و I m 365 در رک اضافی (فقط یک رک اضافی)

ترتیب مدولها در رک اضافی مانند رک اصلی است با این تفاوت که رک اضافی cpu ندارد یعنی در پیکر بندی اسلات دوم خالی است

شروع برنامه نویسی :

سیستم های عددی مورد استفاده در plc: اعدادی که ما معمولاً با آنها در محاسبات سرو کار داریم اعداد مبنای ۱۰ هستند و decimal خوانده می شوند در کامپیوترها که plc نیز عضوی از خانواده آنهاست اطلاعات در مبنای ۲ یعنی ۰ و ۱ شناخته می شوند از نظر الکتریکی وصل شدن مدار معادل یک منطقی و قطع شدن مدار معادل صفر منطقی است تبدیل یک عدد از مبنای decimal به binary ساده است کفایت با تقسیم متوالی آن بر ۲ باقیمانده های بدست آمده و آخرین خارج قسمت را از آخر به اول در کنار هم قرار دهیم. علاوه بر سیستم باینری دو سیستم عددی دیگر نیز در plc مورد استفاده قرار می گیرد که عبارتند از bcd, hexadecimal

Hexadecimal یا مبنای ۱۶ ارتباط مستقیم با مبنای ۲ دارد. از آنجا که بیت های باینری در کنار هم به صورت طولانی ردیف می شوند فهم آنها مشکل است با تبدیل باینری به هگز این بیت ها در فضای کمتری جا می گیرند. سیستم bcd (binary coded decimal) اعداد دسیمال هستند این سیستم در بعضی وصایل ورودی خروجی plc مانند شمارنده ها مورد استفاده قرار می گیرد تفاوت bcd با decimal در این است که اعداد وزن دهگان یا صدگان و... ندارند

برای دریافت فایل Word پروژه به سایت **ویکی پاور** مراجعه کنید. فاقد آرم سایت و به همراه فونت های لازم

مثال:

Decimal	43	
binary	1011	0010
hexadecimal	b	2
Bcd	0100	0011

فرمت آدرس دهی در S7 :

آدرس دهی ورودی ها :

ورودی plc از جنس bit یا byte یا word یا dword باشد

برای آدرس دهی یک بیت باید ابتدا شماره بیت را بنویسیم سپس با گذاشتن نقطه آدرس بیت را در آن

بایت مشخص می کنیم مثال

آدرس بیت	توضیح
0.0	بیت صفر از بایت صفر
4.7	بیت 7 از بایت 4

بدیهی است عدد سمت راست که بیت را مشخص می کند نمی تواند از ۷ بزرگتر باشد چون در یک بایت

۸ بیت داریم از صفر تا ۷ از این رو آدرسی مانند 0.8 نادرست خواهد بود کلیه آدرس های ورودی در

S7 با علامت I شروع می شوند انواع آدرس دهی در زیر مشخص شده است

نوع ورودی	نحوه نمایش	مثال
Bit	I	I 0.1
Byte	I b	I b1
word	I w	I w2
Dword	I d	I d8

باید توجه داشت وقتی یک I w را در برنامه بکار می گیریم آدرس I w بعدی باید حداقل ۲ بایت با

آدرس قبلی فاصله داشته باشد این نکته برای double word نیز باید رعایت شود یعنی هر آدرس با

آدرس بعدی باید ۴ بایت فاصله داشته باشد

برای دریافت فایل Word پروژه به سایت ویکی پاور مراجعه کنید. فاقد آرم سایت و به همراه فونت های لازم

تذکر: آدرس دهی ورودی های جنبی (peripheral) که از طریق شبکه دریافت می شوند با علامت

pi میباشد

آدرس دهی خروجی ها نیز مانند ورودی ها ست بجز اینکه برای خروجی ها بجای علامت I از علامت

Q استفاده می شود

فرمت دیتا ها در S7 :

Bool,byte,word,dword,lword(long word)

متغیر های بولین برای عملیات منطقی مثل and,or,xor

Int(integer 16bit,+32767)

برای عملیات ریاضی مثل جمع و تفریق و ضرب و.. استفاده می شود

Dint(double int 32bit,+2147483648)

Sint(short integer,8bit,+127)

Real(32bit,+3.4*10⁺38)

اعداد حقیقی

Char(8bit)

کاراکتر حروف است

Time(32bit)

Time-of-day(32bit)

Date(16bit)

بلاک های برنامه نویسی:

بلاکهایی که در برنامه نویسی توسط S7 بکار می روند به ۳ دسته زیر تقسیم می شوند

1: بلاکهای منطقی (logic blocks)

این بلاک ها حاوی دستورات برنامه نویسی هستند که به آنها code blocks نیز گفته می شود و عبارتند

از:

Ob organization block

Fb function bock

Fc function

Sfb system function block

Sfc system function

2: دیتا بلاک ها

برای دریافت فایل Word پروژه به سایت **ویکی پاور** مراجعه کنید. فاقد آرم سایت و به همراه فونت های لازم

این بلاک ها همانطور که از نامشان پیداست حاوی دیتا ها هستند. دیتاهایی که توسط بلاک های منطقی خوانده یا نوشته می شوند انواع دیتا بلاک ها عبارتند از:

Db data block
Sdb system data block
user defined type یا Udt

بلاک هایی هستند که حاوی دیتا بوده و بسته به نیاز همراه با دیتا بلاک برای پرهیز از نوشتن دیتا های تکراری بکار می روند تعداد بلاک ها بسته به نوع cpu متفاوت است جدول زیر تعداد بلاک ها را در چند cpu مختلف مقایسه کرده است

Cpu416-2	Cpu412-1	Cpu314	Cpu312	
44	23	13	3	Ob
2048	256	128	32	Fb
2048	256	128	32	Fc
4095	511	127	63	Db

: Organization blocks

Ob ها در plc رابط بین سیستم عامل و برنامه کاربری هستند. این بلاک ها توسط سیستم عامل فرا خوانده شده و برای مقاصدی چون کنترل سیکلی | ،وقفه و راه اندازی بکار می روند در بین ob ها نام ob1 صدا زده می شوند ob ها انواع مختلفی دارند . لیست و نام آنها را می توان در جدول زیر مشاهده کرد :

برای دریافت فایل Word پروژه به سایت ویکی پاور مراجعه کنید. فاقد آرم سایت و به همراه فونت های لازم

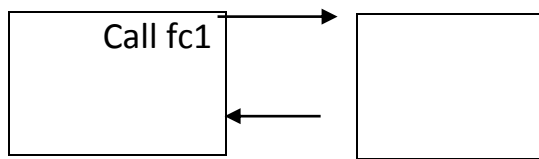
نام ob	نوع	درجه اولویت
Ob1	Main program scan	1
Ob10to ob17	Time of day interrupts	2
Ob 20 to ob23	Time-delay interrupts ob20	3to6
Ob 30 to ob38	Cyclic interrupts	7to15
Ob 40 to 47	Hardware interrupts	16to23
Ob 60	Multicomputing interrupts	25
Ob 70,72	Redundancy error	25 28
Ob80,ob82to87	Asynchronous error	26
Ob90	Background cycle	29
Ob100,ob102	Startup	27
Ob121,ob122	Synchronous error	

همانطور که مشاهده می شود هر گروه از ob ها یک درجه اولویت دارد. هر چه عدد اولویت بالاتر باشد اهمیت آن بلاک از نظر اجرا بیشتر است. به عبارت دیگر اجرای هر ob می تواند توسط ob دیگر که درجه اولویت بالاتری دارد قطع گردد. ob1 کترین اولویت را دارد و هر کدام از ob های دیگر می توانند آنرا قطع کنند. بسته به نوع cpu ممکن است برخی از این ob ها موجود نباشند درجات اولویت ob ها که در جدول فوق آورده شده بعنوان پیش فرض سیستم می باشند در s7-300 اولویت آنها ثابت بوده ولی در s7-400 میتوان آنها را تغییر داد. در پارامترهای cpu در بخش هایی که مربوط به وقفه است مانند time of day interrupt, cyclic interrupt, interrupt اولویت ها لیست شده و قابل تغییر می باشد.

fc: Fc (function) ها بلاک های منطقی و حاوی دستورات برنامه نویسی هستند

به طور معمول fc یک سری ورودی می گیرد و بر اساس برنامه ای که داخل آن نوشته شده خروجی هایی را ایجاد نماید ورودی ها در بلاک ما قبل که fc را صدا می زند به fc داده می شود و خروجی های fc نیز در همان بلاک مورد استفاده قرار می گیرد

برای دریافت فایل Word پروژه به سایت **ویکی پاور** مراجعه کنید. فاقد آرم سایت و به همراه فونت های لازم



Fc1

استفاده از fc بخصوص وقتی در برنامه اصلی نیاز به انجام function تکراری باشد که فقط هر بار ورودی های آن تغییر کند یا آدرس خروجی هایش عوض شود بسیار مفید است و منجر به کاهش حجم دستورات برنامه نویسی می شود

فانکشن بلاک (fb(function block) :

Fb از نظر عملکرد مانند fc است یعنی ورودی می گیرد خروجی ایجاد می کند ولی یک تفاوت مهم با fc دارد و آن اینکه دارای حافظه است. حافظه آن یک دیتا بلاک خاص است وقتی fb صدا زده می شود باید همراه با آن نام دیتا بلاک که حافظه اش تلقی می شود را نیز به کار برد مثال: call fb1,db1 تمام ورودی خروجی های fb و سایر پارامترهایی که در موقع ایجاد fb تعریف می شوند در این دیتا بلاک ذخیره می گردند

دیتا بلاک (db(data block) :

دیتا بلاک ها بر خلاف fb,fc,ob حاوی دستورات step7 نیستند بلکه برای ذخیره سازی دیتا بکار می روند می توان یک دیتا بلاک را write protect کرد تا plc در حین اجرای برنامه نتواند دیتا های آن را عوض کند این کار با کلیک راست روی db و انتخاب properties امکان پذیر است . دیتا بلاک ها دو نوع هستند :

1- shared db این دیتا بلاک ها همانطور که از نامشان پیداست بصورت اشتراکی بوده و برای تمام بلاک های برنامه قابل دسترسی هستند. دیتاهای آنها با بسته شدن دیتا بلاک از بین نمی روند. ساینز db متغییر است و بستگی به نوع cpu دارد.

برای دریافت فایل Word پروژه به سایت ویکی پاور مراجعه کنید. فاقد آرم سایت و به همراه فونت های لازم

2 – instance db این دیتا بلاک ها بعنوان حافظه ای برای fb ها منظور شده اند پارامترهایی که به fb ارسال شده و متغیر های استاتیک در این db ها ذخیره می شوند و وقتی اجرای fb کامل می شود از بین نمی روند

بلاکهای سیستم system blocks :

بلاکهایی هستند که از قبل برای مقاصد خاصی نوشته شده اند و به دو دسته زیر تقسیم میشوند

Sfc یا system function

Sfb یا system function fgh;

Sfc ها همانند fb ها دارای حافظه هستند و باید با نام دیتا بلاک مربوط به آنها صدا زده شوند ولی sfc ها مانند fc ها حافظه ندارند .

فراخوانی بلاک ها از داخل یکدیگر :

همانطور که می دانیم برنامه می تواند به یکی از دوروش زیر نوشته شود

1 – بطور کامل در بلاک اصلی ob1 که به این روش linear programming گویند.

2 – بخش های برنامه در بلاک های مختلف نوشته شده و از بلاک ما قبل صدا زده می شوند که به این روش structured programming گویند

از ob میتوان fc,fb را صدا زد همینطور از fbها و fc نیز می توان سایر fbها و fc ها را صدا زد. نکته ای که باید رعایت شود اینستکه توسط step7 ابتدا باید بلاک نهایی و سپس بلاک های ما قبل ایجاد شود یعنی مثلا ابتدا fb1,fc1 و سپس ob1 .

تعداد بلاکهایی که از داخل هم فراخوان می شوند بستگی به nesting depth دارد که از مشخصات cpu است مثلا برای cpu315 این پارامتر ۸برای cpu414-3 این پارامتر ۲۴ می باشد یعنی حداکثر 24 بلاک تودر تو می توانیم داشته باشیم .

نحوه ایجاد بلاک در simatic manager :

بلاکهای ob,fb,fc,db که توضیح داده شده توسط simatic manager ایجاد می شوند. باید توجه داشت که بلاک های سیستم یعنی sfb,sfc ها از قبل ایجاد شده اند و در نرم افزار موجود هستند از اینرو این بلاک ها را نمی توان ایجاد کرد. اگر قبلا سخت افزار پیکر بندی شده باشد با باز کردن station در پنجره سمت چپ s7 program را مشاهده می کنیم که در زیر cpu قرار گرفته است با باز کردن این

برای دریافت فایل Word پروژه به سایت ویکی پاور مراجعه کنید. فاقد آرم سایت و به همراه فونت های لازمه

پوشه دو پوشه دیگر مشاهده می شود یکی به نام **blocks** و دیگری بنام **sources** , پوشه **sources** برای برنامه نویسی به صورت **source** بکار می رود . بلاک های برنامه در پوشه **blocks** ایجاد می شوند معمولا پس از پیکر بندی سخت افزار بطور پیش فرض یک بلاک **ob1** در داخل پوشه **blocks** ایجاد می شود که البته خالی است و برنامه ای داخل آن وجود ندارد .بعلاوه اگر بعد از اتمام کار پیکر بندی سخت افزار توسط **hwconfing** عمل کامپایل را توسط **save and compile** انجام داده باشیم آیکون دیگری به نام **system data** در پوشه **blocks** ظاهر می شود که اطلاعات سخت افزار و شبکه را در خود دارد این اطلاعات به **plc** دانلود شده و بر اساس آن ها **plc** می تواند سخت افزار پیکر بندی شده را تشخیص دهد اگر سخت افزار قبلا پیکر بندی نشود و کاربر بخواهد ابتدا برنامه نویسی را انجام دهد در این صورت پوشه **s7 program** و در زیر مجموعه آن پوشه **block** را خواهیم داشت در هر دو حالت چه با وجود سخت افزار و چه بدون آن برای ایجاد بلاک می توان به یکی از دو روش زیر عمل کرد :

1- کلیک روی پوشه **blocks** و سپس با استفاده از منوی **insert.>s7 block** و انتخاب **bock** مورد نظر از لیست

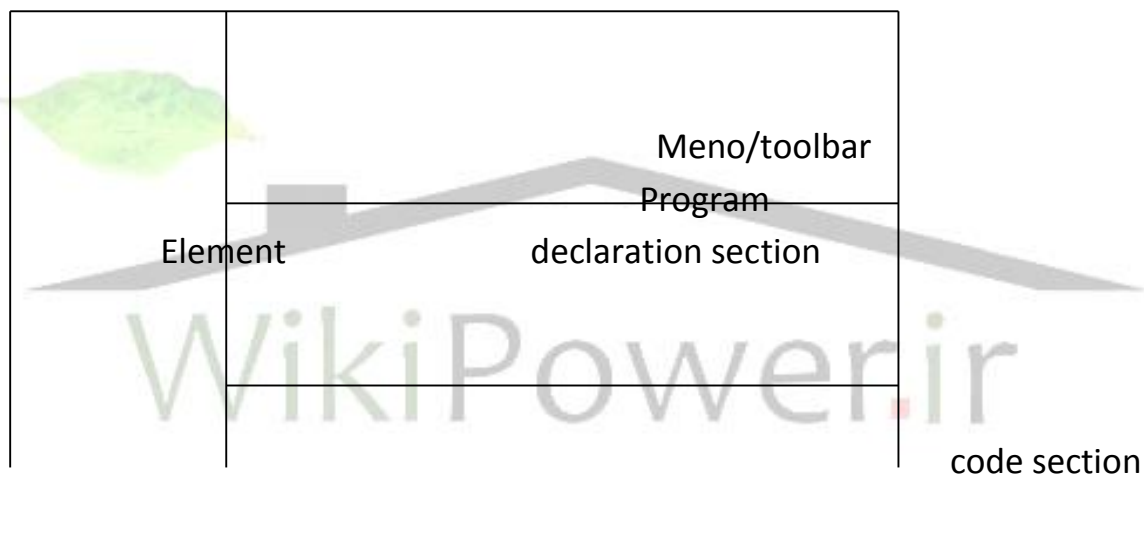
2- کلیک راست روی پوشه **blocks** یا در پنجره سمت راست و انتخاب **insert new object** پس از انتخاب بلاک مورد نظر پنجره جدیدی باز می شود . نکاتی که باید به آنها توجه کرد این است که :
 ۱- هر بلاک دارای یک شماره است و ود برنامه این شماره را بعد از اسم بلاک پیشنهاد می دهد که در صورت نیاز می توان آنرا تغییر داد ۲- زبان برنامه نویسی برای بلاک های منطقی یعنی **fc,fb,ob** زبان **stl** است که در صورت دلخواه قابل تغییر است

۳- وقتی دیتا بلاک ایجاد می شود نوع آنرا بطور پیش فرض **shared** می باشد .می توان آنرا از نوع **instans** انتخاب کرد که در این صورت شماره **fb** را باید وارد کنیم یعنی **fb** باید قبلا در پوشه **block** ایجاد شده باشد . ۴- اگر در **simatic manager** از منوی **option>customize** در بخش **general** گزینه **open new object** فعال شده باشد در اینصورت هر بلاک بمحض ایجاد شدن با برنامه مربوطه اش باز می شود

آشنایی با محیط ذیر برنامه **lad/stl/fbd** :

برای دریافت فایل Word پروژه به سایت ویکی پاور مراجعه کنید. فاقد آرم سایت و به همراه فونت های لازم

اگر روی هر کدام از بلاک های ob,fb,fc که در پوشه block برنامه simatic manager ایجاد شده اند کلیک کنیم برنامه جدیدی به نام lad/stl/fbd اجرا شده و بلاک مورد نظر در محیط آن باز می شود



Program element

این بخش المانهای برنامه نویسی را در بر دارد. بسته به اینکه چه نوع زبانی انتخاب شده باشد این المانها متفاوت خواهد بود کاربر می تواند با انتخاب هر یک از زبانهای lad, stl, fbd

برای دریافت فایل Word پروژه به سایت ویکی پاور مراجعه کنید. فاقد آرم سایت و به همراه فونت های لازم

منوی view تفاوت محتویات این قسمت را برای زبانهای مختلف ملاحظه کند در صورت فعال نبودن بخش program element کاربر می تواند آنرا از طریق insert>program element فعال نماید .

: Declaration selection

این بخش جهت تعریف متغیر های محلی و ورودی و خروجی بلاک به کار می رود. برای ob ها در این بخش صرفا متغیر temp را می توان دید ولی برای fb و fc میتوان علاوه بر temp ورودی و خروجی و موارد دیگری را مشاهده کرد . جدول زیر موضوع را کاملتر می کند

declaration	fc	fb	ob
in	x	x	-
out	x	x	-
In/out	x	x	-
temp	x	x	x
static	-	x	-

نکاتی که با توجه به جدول فوق باید به آن توجه داشت:

متغیر temp متغیر موقت محلی است که در تمام بلاک ها قابل تعریف است. مقدار ذخیره شده در این متغیر با بسته شدن بلاک از بین می رود .

از متغیر های temp که توسط کاربر تعریف می شوند می توان شبیه فلاگ ها برای ذخیره سازی نتایج میان برنامه استفاده کرد .

در ob ها متغیر های temp از قبل تعریف شده ای هستند که کاربرد مهمی در مدیریت خطاها دارند

Ob ورودی و خروجی ندارد بنابراین نمی توان آنرا شبیه فانکشن صدا زد

متغیر in برای ورودیهای بلاک و متغیر out برای خروجی های بلاک تعریف می شوند

برای دریافت فایل Word پروژه به سایت ویکی پاور مراجعه کنید. فاقد آرم سایت و به همراه فونت های لازم

متغیر in/out برای یک خروجی که به عنوان ورودی نیز به کار می رود (مانند خروجی که در حالت خود نگهدار بعنوان ورودی استفاده می شود)

متغیر static خاص fb است موارد مربوط به fb بجز نوع temp همگی در db مربوطه ذخیره می شوند. پس اگر نیاز به متغیری برای ذخیره سازی نتایج میان برنامه باشد و لازم باشد که این مقدار در db ذخیره شوند آنرا از نوع static تعریف می کنیم

قبل از تمام متغیر های فوق الذکر وقتی در برنامه نویسی استفاده می شوند با علامت # قرار دارد مثال:

l#test

: Code section

این بخش جهت برنامه نویسی استفاده می شود و خاص بلاک های منطقی است



Ob1:"main program sweep (cy cle)"

:

comment:

Network1:title:

Comment

همانطور که در شکل ملاحظه می شود در این قسمت موارد زیرو وجود دارد

برای دریافت فایل Word پروژه به سایت ویکی پاور مراجعه کنید. فاقد آر اسایت و به همراه فونت های لازم

Block title : با کلیک کردن ری این قسمت نام بلاک و عنوان آن را می توان نوشت حداکثر 64 کاراکتر

Block comment : در این باکس کاربر می تواند توضیحاتی در مورد بلاک و عملکرد آن بنویسد

Network comment : قبل از هر چیزی باید توجه داشت که **network** در اینجا معنای شبکه ندارد

بلکه نشاندهنده یک قسمت از برنامه یا به عبارت دیگر یک **segment** است. هر بلاک برنامه نویسی

میتواند شامل **network** های مختلف باشد که به دنبال هم قرار گرفته اند و در هر کدام از آنها یک بخش

از برنامه نوشته شده است. برای اضافه کردن **network** میتوان از منوی **insert network** استفاده

کرد هر **network** یک **title** دارد که میتوان با حداکثر 64 کاراکتر آنرا معرفی کرد در **network**

comment میتوان توضیحاتی را در مورد این بخش از برنامه نوشت. برنامه هر **network** در قسمت

statement نوشته می شود معمولا در برنامه یک بلاک، بخشهای مختلف برنامه کنترل را از یکدیگر جدا

کرده و هر بخش را در **network** جداگانه مینویسند.

نحوه استفاده از بلاک ها :

نحوه ایجاد و استفاده از فانکشن **fc**: همانطور که قبلا ذکر شد اگر از داخل یک بلاک مثلا **ob1** بخواهیم

فانکشنی را صدا بزیم لازم است که فبلا آنرا ایجاد کرده و ورودی و خروجی های آنرا تعیین کرده باشیم

اگر قبل از ایجاد فانکشن آنرا صدا بزیم می بینیم که دستور با رنگ قرمز که نشان دهنده وجود اشکال

است ظاهر می شود بنابر این لازم است ابتدا توسط **simatic manager** این فانکشن را ایجاد کرده

وسپس با کلیک کردن روی آن آنرا توسط **lad/stl/fbd** باز کنیم پس از آن ابتدا باید ورودی و خروجی

فانکشن را در قسمت **declaration** باید **a,b** بعنوان ورودی و **c** بعنوان خروجی و همه آنها از نوع

عدد صحیح یعنی **int** تعریف شوند. سپس برنامه جمع در **network** داخل **fc** نوشته شود اکنون اگر

fc را ذخیره کنیم و به **ob1** برگردیم در پنجره **program element** در زیر مجموعه **fc block** آنرا

مشاهده میکنیم ثانيا پس از صدا زدن فانکشن میبینیم که پارامتر های ورودی و خروجی آن در زیر دستور

call ظاهر می شود. با دادن مقادیر یا آدرس مناسب به ورودی ها و خروجی ها باید انر تکمیل نمود این

فانکشن می تواند بارها و از هر بلاکی صدا زده شود

نحوه ایجاد و استفاده از فانکشن بلاک **fb** :

برای دریافت فایل Word پروژه به سایت ویکی پاور مراجعه کنید. فاقد آرم سایت و به همراه فونت های لازم

روش ایجاد و استفاده از fb تا حدودی شبیه fc است ولی بدلیل نیاز به دیتا بلاک قدهایی که باید برداشته شود بصورت زیر خواهد بود :

۱- ایجاد fb در simatic manager

۲- باز کردن fb توسط برنامه lad/stl/fbd

۳- تعریف ورودی و خروجی و سایر متغیر ها در قسمت declaration

۴- نوشتن برنامه در network های مربوطه

۵- ذخیره سازی fb

۶- بازگشت به simatic manager و ایجاد دیتا بلاک

۷- انتخاب نوع instance برای دیتا بلاک و انتخاب fb مربوط به آن

پس از انجام مراحل فوق می توان fb را در پنجره program element در زیر مجموعه fb block مشاهده کرد و می توان هر بلاکی مثلا در ob1 آنرا صدا زد ولی در دستور call برنامه سوال می کند که آنرا ایجاد کند یا نه و در این صورت نیازی به مراحل 6,7 نمی باشد. در حین اجرای برنامه توسط plc تمام پارامتر های fb بلاک بجز نوع temp در دیتا بلاک مربوط به آن بطور اتوماتیک ذخیره می شوند و نیازی به استفاده از دستور برنامه نویسی خاصی برای این کار نیست. اگر کاربر بعد از مرحله 6 فوق الذکر روی db در simatic manager کلیک کند تا باز شود مشاهده خواهد کرد که محتویات db دقیقا شبیه آنچه در بالای fb تعریف کرده می باشد

نحوه ایجاد و استفاده از دیتا بلاک db :

دیتا بلاک instance که خاص fb, است در بحث fb توضیح داده شد نوع دیگر دیتا بلاک db اشتراکی یا نوع shared است که همه بلک های منطقی به آن دسترسی دارند میتوانند در ان بنویسند یا از آن بخوانند. قدمهایی که برای ایجاد و استفاده از این نوع دیتا بلاک باید برداشت اشاره میشود : ۱- ایجاد دیتا

بلاک در پوشه بلاک برنامه simatic manager

۲- باز کردن دیتا بلاک با دربار کلیک روی آن توسط برنامه lad/stl/fbd

۳- تعریف سطر های لازم با متغیر های دلخواه و لسانی سمبولیک دلخواه

برای دریافت فایل Word پروژه به سایت ویکی پاور مراجعه کنید. فاقد آرم سایت و به همراه فونت های لازم

نحوه ایجاد و استفاده از سمبل ها:

در برنامه s7 با آدرسهای مربوط به سیگنالهای ورودی خروجی، کانترها، تایمرها، دیتا بلاک ها... سروکار داریم که می توانند آدرس مطلق یا سمبولیک داشته باشند. q4.0 و i1.1 و c1 و t5 نمونه هایی از آدرس دهی مطلق هستند. اگر بجای آدرس های فوق از آدرسهای سمبولیک معنی دار استفاده شود، خواندن برنامه و عیب یابی آن سادهتر خواهد شد مثلاً بجای q4.0 از سمبل out استفاده کرده و بعد از آن این کلمه را در برنامه برای آدرس بکار ببریم.

سمبل ها می توانند به صورت محلی یا اشتراکی تعریف شوند. سمبل های اشتراکی در کل برنامه و کل بلاک ها قابل شناسایی و استفاده هستند ولی سمبل های محلی صرفاً در همان بلاکی که تعریف شدهاند میتوانند شناسایی شوند اسامی که برای ورودی خروجی و سایر متغیرها در بخش declaration یک بلاک بکار می رود سمبل های محلی هستند

بدیهی است در بلاک های مختلف می توان سمبل های محلی متفاوت ولی با نام یکسان بکار برد ولی نام سمبل های اشتراکی در کل برنامه باید منحصر به فرد باشد. برای ایجاد سمبل های اشتراکی لازم است از برنامه symbol editor استفاده کنیم که آیکون آن وقتی در simatic manager روی s7 program کلیک کنیم که در پنجره سمت راست ظاهر می شود با کلیک کردن روی آن برنامه symbol editor اجرا شده و جدول خالی سمبل ها نشان داده می شود. کاربر می تواند با وارد کردن سمبل ها این جدول را کامل کند. سمبل هایی که در این جدول ظاهر می شوند به صورت اشتراکی بوده و در کل برنامه قابل استفاده است نکات زیر قابل ذکر است:

۱- نام سمبل ها حداکثر ۲۴ کاراکتر و تعداد سمبل ها حداکثر ۱۶۳۸۰ میباشد حروف کوچک و بزرگ یکسان تلقی می گردند.

۲- در s7 میتوان جدول سمبل ها که توسط یک ادیتور بیرونی ایجاد شده (از جمله توسط s5) را به داخل برنامه import نمود. برای اینکار از منوی symbol>import در برنامه symbol editor استفاده می شود.

در موقع دانلود کردن برنامه به plc سمبل ها دانلود نمی شوند و plc فقط با آدرسهای مطلق کار می کند. بنا براین اگر برنامه ای را از plc به کامپیوتر منتقل کنیم در اینصورت برنامه بدون سمبل مشاهده خواهد شد مگر اینکه فایل سمبل ها از قبل روی کامپیوتر وجود داشته باشد

برای دریافت فایل Word پروژه به سایت ویکی پاور مراجعه کنید. فاقد آرم سایت و به همراه فونت های لازم

استفاده از راهنمای s7 :

روش اول: مکان نما را بر روی هر منوی فرمان قرار داده و کلید f1 را فشار دهید. راهنمای حساس به وارد نمودن متن برای منوی فرمان انتخاب شده پدیدار می گردد.

روش دوم: بر روی دکمه علامت سوال واقع در خط ابزار (tool bar) کلیک نمایید تا موس به یک مکان نمای راهنما تبدیل شود. حال در صورت کلیک کردن بر روی هر عنصر خاصی، راهنما فعال می گردد.

فرمهای نوشتن برنامه در s7 :

1 - فرم نردبانی (ladder)

2 - فرم نرم افزاری (statement list)

3 - فرم بلوکی (function block data)

فرم نردبانی (lad) :

هر گاه در برنامه المان ها به صورت پارالل رسم شوند یا به عبارت دیگر به صورت قطعات الکترونیکی موجود در کتابخانه قرار گیرند در این صورت مدار به صورت ladder بسته شده است.

حال باید در ابتدا با فرمانها و المانهای موجود در کتابخانه s7 آشنا شویم از چگونگی کاربرد آنها در مدارات منطقی آگاه شویم:

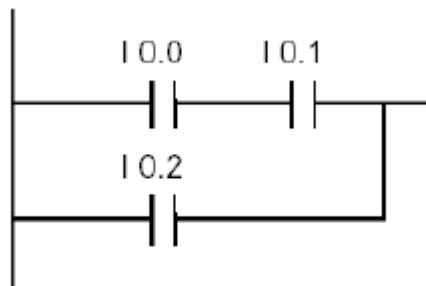
دستورات bit logic :

*a <bit> a : دستور a چک می کند که آیا وضعیت بیت آدرس داده شده 1 است یا نه و سپس rlo را

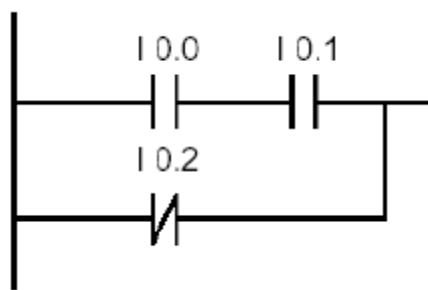
با وضعیت این بیت and میکند

مثال:

برای دریافت فایل Word پروژه به سایت ویکی پاور مراجعه کنید. فاقد آرم سایت و به همراه فونت های لازم

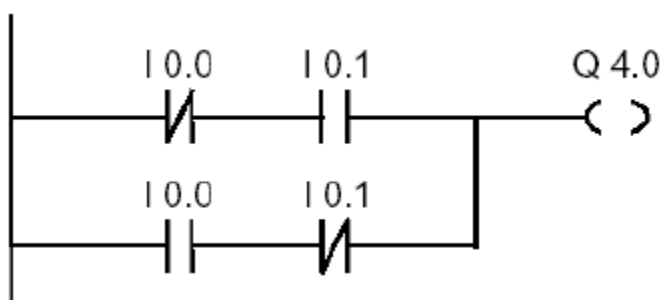


an :چک میکند که آیا وضعیت بیت آدرس داده شده 0 است یا نه و سپس rlo را با وضعیت این بیت and میکند



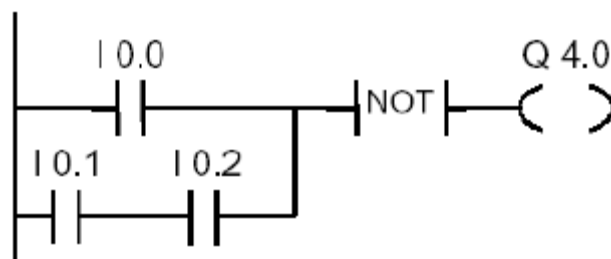
O :چک میکند آیا وضعیت بیت آدرس داده شده 1 است یا نه و سپس rlo را با وضعیت این بیت or میکند.

Xor :چک میکند که آیا وضعیت بیت آدرس داده شده 1 است یا نه و سپس rlo را با وضعیت این بیت xor میکند در واقع اگر یکی از این دو یک باشند نتیجه یک خواهد شد



not |
|
بیت rlo را منفی میکند

برای دریافت فایل Word پروژه به سایت ویکی پاور مراجعه کنید. فاقد آرم سایت و به همراه فونت های لازم



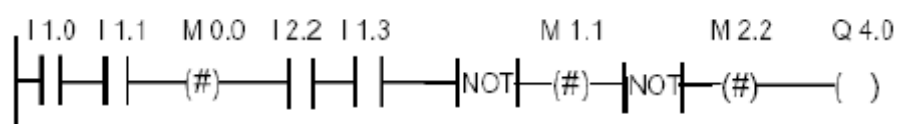
(OUTPUT COIL) --- ()

به این صورت کار میکند که اگر بیت rlo یک شود آدرس این coil نیز یک می شود و اگر هم rlo صفر باشد بیت آدرس نیز صفر می شود

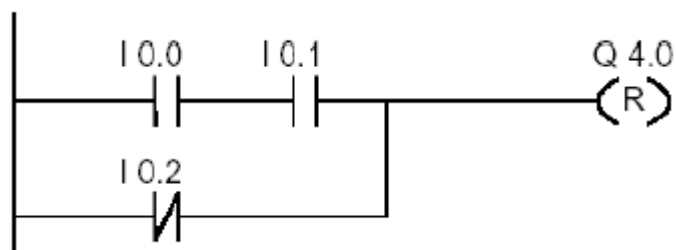
المانی که بیت rlo را در آرس مورد نظر save میکند

--- (#)---

مثال:

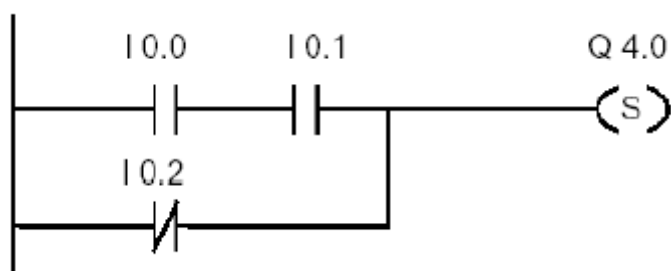


R <bit>: بیت rlo را ریست می کند یعنی اگر یک باشد صفر و اگر صفر باشد یک می کند



S(set): خروجی سیگنال هر چه باشد یک میکند.

برای دریافت فایل Word پروژه به سایت ویکی پاور مراجعه کنید. فاقد آرم سایت و به همراه فونت های لازم



Flip flap: برای ثابت نگهداشتن ورودی در حالت صفر و یا یک به فلیپ فلاپ احتیاج داریم اصطلاحاً

به عنوان یک تیغه خود نگهدار عمل می کند

Reset set flip flap(rs): عمل ست و ریست را وقتی که rlo باشد اجرا می کند. بنابراین اگر rlo

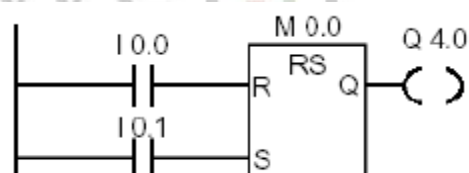
صفر باشد این دستور اجرا نمی شود عمل ریست کردن بیت آدرس داده شده وقتی اتفاق می افتد که

ورودی ۲ یک و ورودی S صفر باشد. عمل ست کردن بیت آدرس داده شده وقتی اتفاق می افتد که

ورودی ۲ صفر و ورودی S یک باشد اگر هر دو ورودی یک شوند نیز عمل ست انجام می شود



مثال:



اگر i0.0 یک باشد و i0.1 صفر باشد بیت حافظه set m 0.0 می شود q4.0 مقدار صفر را به خود می گیرد.

Set reset flip flap(sr): عمل ست و ریست را وقتی که rlo یک باشد اجرا می کند بنا براین اگر

rlo صفر باشد این دستور اجرا نمی گردد عمل ست کردن بیت آدرس داده شده وقتی اتفاق می افتد که

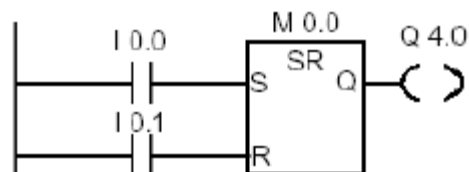
ورودی ۲ صفر و ورودی S یک باشد. عمل ریست کردن بیت آدرس داده شده وقتی اتفاق می افتد که

ورودی ۲ یک و ورودی S صفر باشد اگر هر دو ورودی یک باشند عمل ریست انجام می شود.

برای دریافت فایل Word پروژه به سایت ویکی پاور مراجعه کنید. فاقد آرم سایت و به همراه فونت های لازم



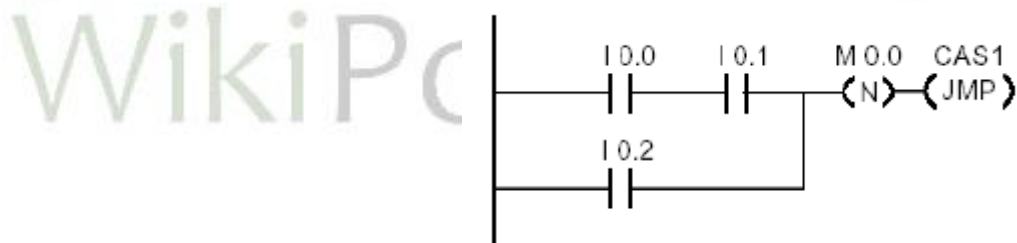
مثال:



اگر $i0.0$ یک باشد و $i0.1$ صفر باشد بیت $m0.0$ set می شود و $q4.0$ مقدار یک را به خود می گیرد

زمانیکه آدرس از صفر به یک تغییر می کند بیت rlo یک می شود --- (N) ---

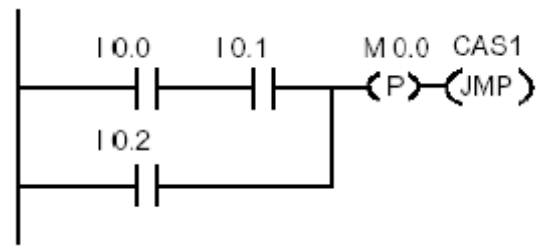
مثال:



بیت حافظه $m0.0$ در rlo قدیم $save$ شده است هنگامی که سیگنال تغییر کند و سیگنال از صفر به یک برود برنامه به آدرس $cas1$ پرش می کند

زمانیکه آدرس از صفر به یک تغییر کند بیت rlo یک می شود. --- (P) ---

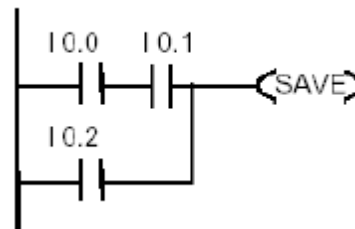
برای دریافت فایل Word پروژه به سایت ویکی پاور مراجعه کنید. فاقد آرم سایت و به همراه فونت های لازم



بیت m0.0 در rlo قدیم save شده است هنگامی که سیگنال از صفر به یک تغییر کند برنامه به ادرس cas1 پرش می کند .

---(SAVE)

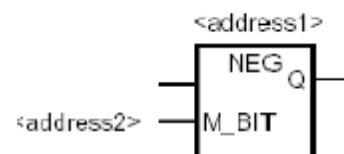
اطلاعت rlo را در br memory ذخیره می کند.



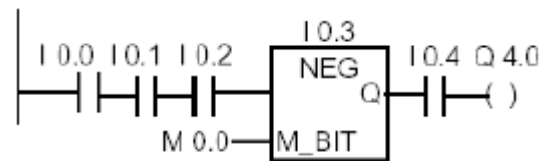
WikiPower.ir

: Neg(negative edge detection)

لبه پایین رونده rlo را مشخص می کند بعبارت دیگر وقتی rlo از یک به صفر می رود این دستور آن را تشخیص داده و نتیجه را با rlo یک آشکار می کند .

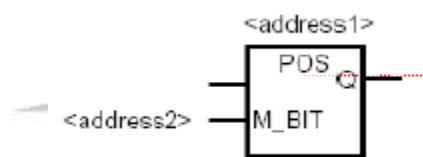


برای دریافت فایل Word پروژه به سایت ویکی پاور مراجعه کنید. فاقد آرم سایت و به همراه فونت های لازم

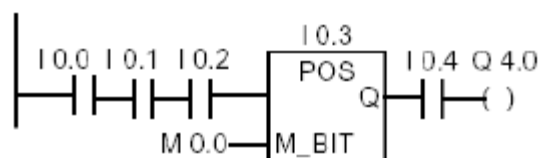


در I 0.3 لبه پایین رونده داریم I 0.0, I 0.1, I 0.2 سیگنال های یک میباشند با لبه پایین رونده در I 0.4 سیگنال یک گردیده و خروجی Q 4.0 یک می گردد.

Pos(positive edge detection) لبه بالا رونده rlo را مشخص می کند بعبارت دیگر وقتی rlo از صفر به یک کی رود این دستور آن را تشخیص داده و نتیجه را با rlo آشکار می کند .



مثال:



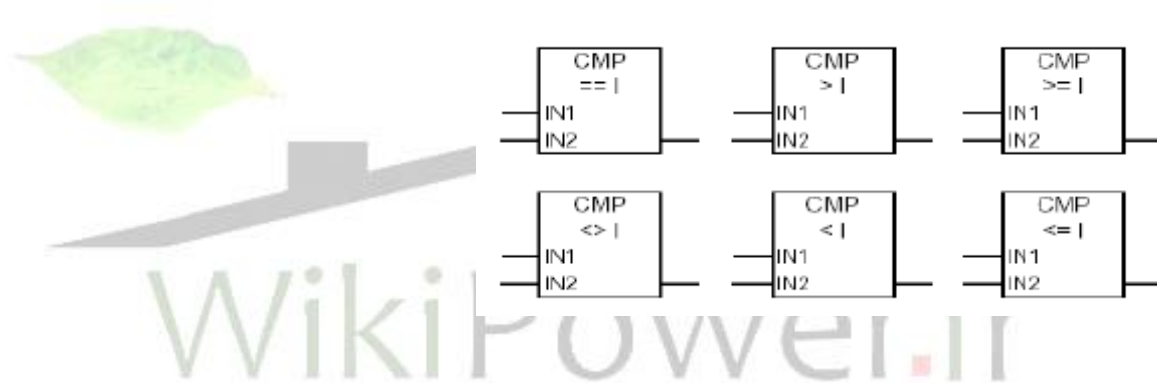
برای دریافت فایل Word پروژه به سایت ویکی پاور مراجعه کنید. فاقد آرم سایت و به همراه فونت های لازم

در i0.3 لبه بالا رونده داریم. در صورتیکه i0.0, i0.1, i0.2 به سیگنال یک تغییر حالت بدهند. در این حالت در i0.4 و خروجی q4.0 سیگنال یک می گردد.

دستورات مقایسه ای (comparison):

این دستورات عملیات مقایسه را انجام می دهند مقایسه در واقع بین محتویات دو آکومولاتور صورت میگیرد.:

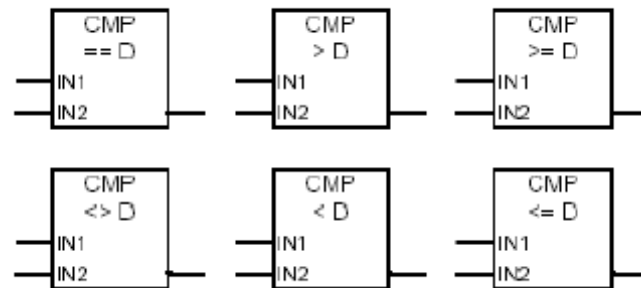
compare integer (cmpi): مقایسه کننده فوق برای مقایسه دو عدد (integer) اعداد صحیح بدون اعشار می گردد که انواع مختلف کوچکتر مساوی، بزرگتر مساوی و ساوی را می توان انتخاب نمود.



compare double integer (cmpd):

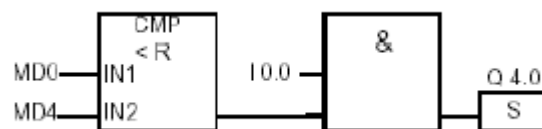
برای مقایسه اعداد integer که بزرگ بوده و بیش از د word برای ذخیره آنها در حافظه مورد نیاز می باشد

برای دریافت فایل Word پروژه به سایت ویکی پاور مراجعه کنید. فاقد آرم سایت و به همراه فونت های لازم



Compare floating point (Cmp r) : دستورات مقایسه ای 32 بیتی هستند

مثال:

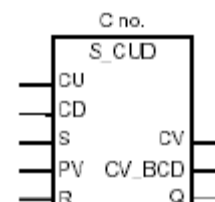


Md0 از md4 کوچکتر باشد در این صورت set q4.0 است

دستورات کانترها (counter instruction) :

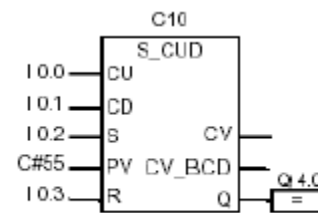
کانترها یا شمارنده ها المان هایی از زبان برنامه نویسی در s7 هستند که ناحیه خاصی از حافظه cpu را به خود اختصاص داده اند. در این ناحیه برای هر کانتر یک word یعنی 16 بیت رزرو شده است تعداد کانتر بستگی به نوع cpu دارد که در مشخصات فنی آن آورده می شود. مقدار کانتر به صورت bcd از 0 تا 999 میباشد

: Assign parameter and counter up/down



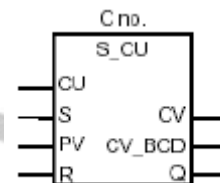
مثال:

برای دریافت فایل Word پروژه به سایت ویکی پاور مراجعه کنید. فاقد آرم سایت و به همراه فونت های لازم



در شمارنده فوق با تغییر سیگنال از صفر به یک در i0.2 set می شود c#55 یعنی مقدار شمارش تا 55 دفعه می باشد هنگامی که i0.0 از 0 به 1 تغییر کند. شمارنده به صورت افزایشی تا 999 به بالا می شمارد. و اگر i0.1 از 0 به 1 تغییر کند به صورت کاهش می شمارد. هنگامی که i0.3 از 0 به 1 تغییر کند مقدار counter در 0 ست میشود

: Assign parameter and count up(s-cu)



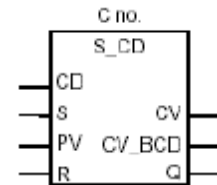
Parameter English	Parameter German	Data Type	Memory Area	Description
no.	Nr.	COUNTER	C	Counter identification number. The range depends on the CPU.
CU	ZV	BOOL	I, Q, M, D, L	ZV input: Up Counter
S	S	BOOL	I, Q, M, D, L, T, C	Input for presetting the counter
PV	ZW	WORD	I, Q, M, D, L or constant	Count value in the range between 0 and 999 or Count value entered as C#<value> in BCD format
R	R	BOOL	I, Q, M, D, L, T, C	Reset input
CV	DUAL	WORD	I, Q, M, D, L	Current count value (hexadecimal number)
CV_BCD	DEZ	WORD	I, Q, M, D, L	Current count value (BCD format)
Q	Q	BOOL	I, Q, M, D, L	Status of the counter

وقتی که rlo برای ورودی متصل به S از صفر به یک برود کانتر با مقدار pv ست می شود. با لبه بالا رونده ورودی CU مقدار کانتر یکی افزایش پیدا می کند. با فعال شدن ورودی R کانتر به مقدار صفر

برای دریافت فایل Word پروژه به سایت ویکی پاور مراجعه کنید. فاقد آرم سایت و به همراه فونت های لازم

ریست می شود وقتی مقدار شمارش بزرگتر از صفر است خروجی q یک و وقتی مقدار شمارش صفر شود این خروجی صفر می شود .

: assign parameter and counter down(s-cd)

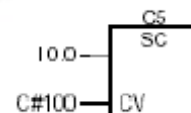


وقتی rlo برای ورودی متصل به S از صفر به یک برود کانتر با مقدار PV ست می شود. بالبه بالا رونده ورودی cd مقدار کانتر یکی کاهش پیدا می کند با فعال شدن ورودی r کانتر به مقدار صفر ریست می شود وقتی مقدار شمارش بزرگتر از صفر باشد خروجی q یک و وقتی مقدار شمارش صفر شود این خروجی صفر می گردد .

: (s)set counter preset value



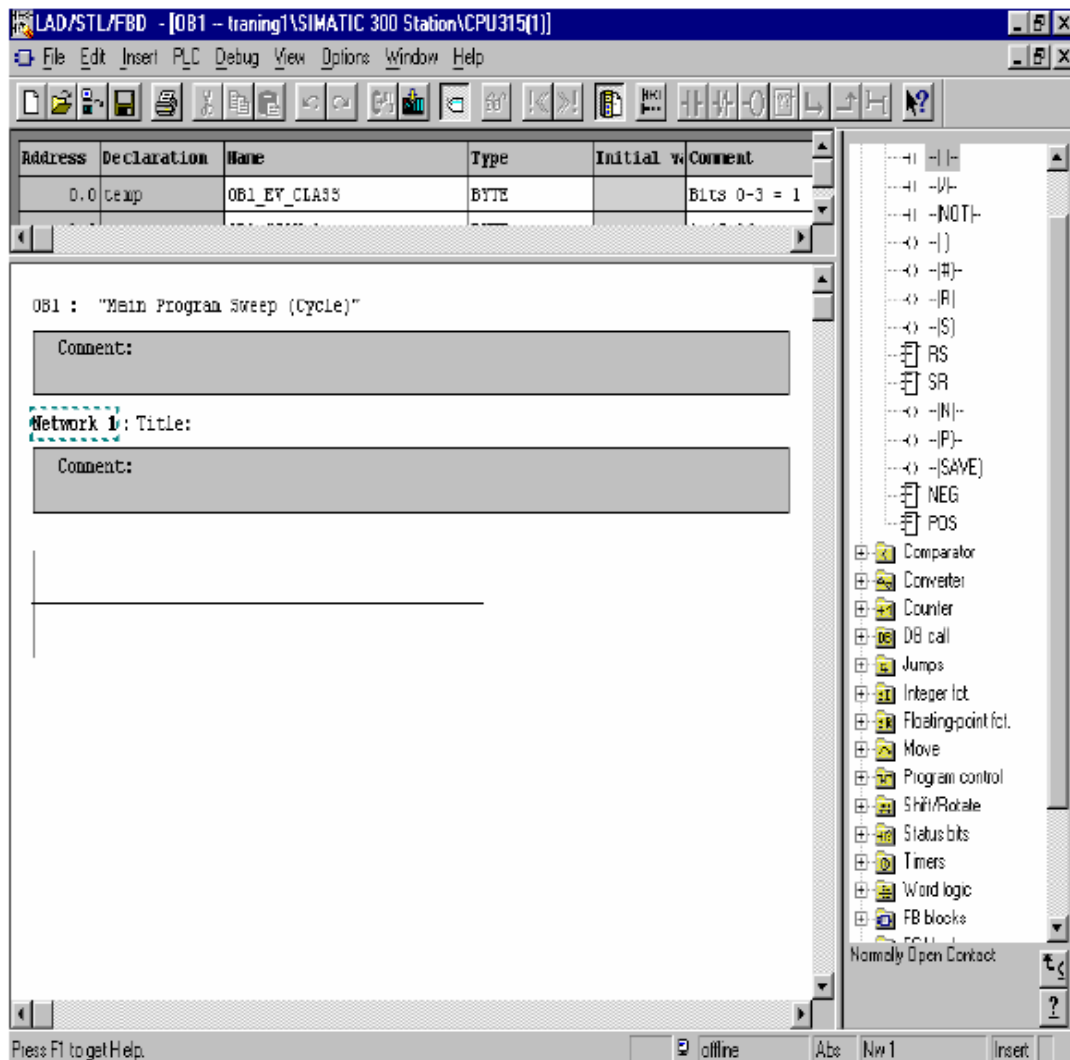
مثال:



در شمارنده فوق مقدار 100 به $S5$ داده شده با تحریک $i0.0$ مقدار کانتر به 100 میرود در غیر این صورت اتفاقی نمی افتد.

در شکل زیر در سمت راست کتابخانه $S7$ نشان داده شده است

برای دریافت فایل Word پروژه به سایت **ویکی پاور** مراجعه کنید. فاقد آرم سایت و به همراه فونت های لازمه



دستورات پرش (jump) :

از دستورات پرش برای کنترل لاجیک برنامه میتوان استفاده کرد بگونه‌ای که در طول سیکل اسکن برنامه از روی برخی دستورات پرش کرده و آنها را اجرا نکند .

بعلاوه میتوان دستور حلقه (loop) را برای بخشی از برنامه در یک سیکل اسکن چند بار اجرا نمود دستورات پرش به ادرس محل پرش که label نامیده می شوند نیاز دارند همین طور دستور loop .

نکاتی که در استفاده از دستورات jump و loop باید مد نظر قرار داد عبارتند از:

Label نباید از چهار کارکتر بیشتر باشد

Label باید با حرف و نه عدد شروع شود

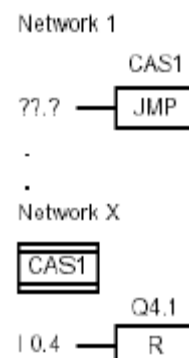
برای دریافت فایل Word پروژه به سایت ویکی پاور مراجعه کنید. فاقد آرم سایت و به همراه فونت های لازم

نقطه ای که به آن پرش label انجام می شود با label شروع شده و پس از آن علامت: قرار می گیرد
 پرش به جلو و عقب هر دو امکان پذیر است
 دستور پرش و label مربوطه هر دو باید در داخل یک بلاک باشند نه اینکه label در بلاک دیگری
 تعریف شده باشد همین طور برای loop .
 نام label باید در داخل بلاک منحصر به فرد باشد استفاده از چند label همانم در یک بلاک مجاز
 نیست
 ماکزیمم فاصله بین دستور پرش و label می تواند 32767 word از program code باشد.
 : Un conditional jump to block



Parameter	Data Type	Memory Area	Description
Name of a jump label	-	-	The address specifies the label to which the program will jump unconditionally.

مثال:

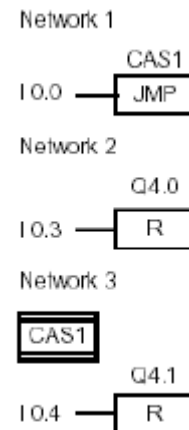


: conditional jump in a block

برای دریافت فایل Word پروژه به سایت ویکی پاور مراجعه کنید. فاقد آرم سایت و به همراه فونت های لازم



مثال:

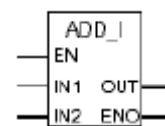


زمانیکه i0.0 فعال شود cas1 اجرا می شود زمانیکه i0.3 فعال شود فرمان reset می آید

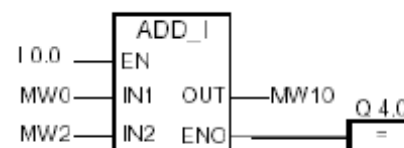
دستورات محاسبه اعداد صحیح (integer math instructions):

دستورات محاسباتی آکومولاتور های 1 و 2 را با هم ترکیب کرده و نتیجه را در آکومولاتور 1 ذخیره میسازد

دستور جمع اعداد صحیح:



مثال:

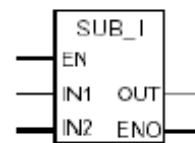


:

برای دریافت فایل Word پروژه به سایت ویکی پاور مراجعه کنید. فاقد آرم سایت و به همراه فونت های لازم

با فعال شدن این بلوک توسط i0.0 ورودی شروع به جمع می نماید و ورودی mw0 و mw2 را می گیرد و در خروجی mw10 می ریزد. اگر حاصل در رنج مورد نظر نباشد یا به هر دلیل قابل قبول نباشد i0.0 صفر میشود و q4.0 را صفر میکند

: Subtract integer (sub -I)



انواع تایمر (timer instructions) :

هر cpu تعداد تایمر مشخصی را ساپورت می کند این تعداد در مشخصات فنی cpu ذکر شده است برای هر تایمر در حافظه cpu یک word رزرو شده است که بیت های صفر تا نه برای مقدار زمان بکار می رود زمان بصورت باینری در این بیت ها ذخیره شده و پس از راه اندازی تایمر شروع به کم شدن می کند تا اینکه به صفر برسد مقدار زمان را باید قبل از راه اندازی تایمر مشخص و به حافظه بار کرد فرمت این زمان میتواند به یکی از دو روش زیر باشد :

1. txyz.#16#w

t. پله زمانی است مثلا ثانیه

Xyz مقدار زمان به صورت bcd

این روش معمولا استفاده نمی شود

2. s5t# ah .cs-dms – که در آن :

برای دریافت فایل Word پروژه به سایت ویکی پاور مراجعه کنید. فاقد آرم سایت و به همراه فونت های لازم

H ساعت و m دقیقه و s ثانیه و ms میلی ثانیه و a,b,c اعداد مربوط به موارد فوق می باشند ماکزیمم

زمانی که میتوان استفاده کرد 2h-46m-30s یا 9990 ثانیه میباشد.

انواع تایمر عبارتند از:

(Pulse timer)s-pu

(extended pulse timer) s-pext

(on-delay timer) s-odt

(retentive on-delay timer) s-odts

(of delay timer) s-offdt

sp

sd

se

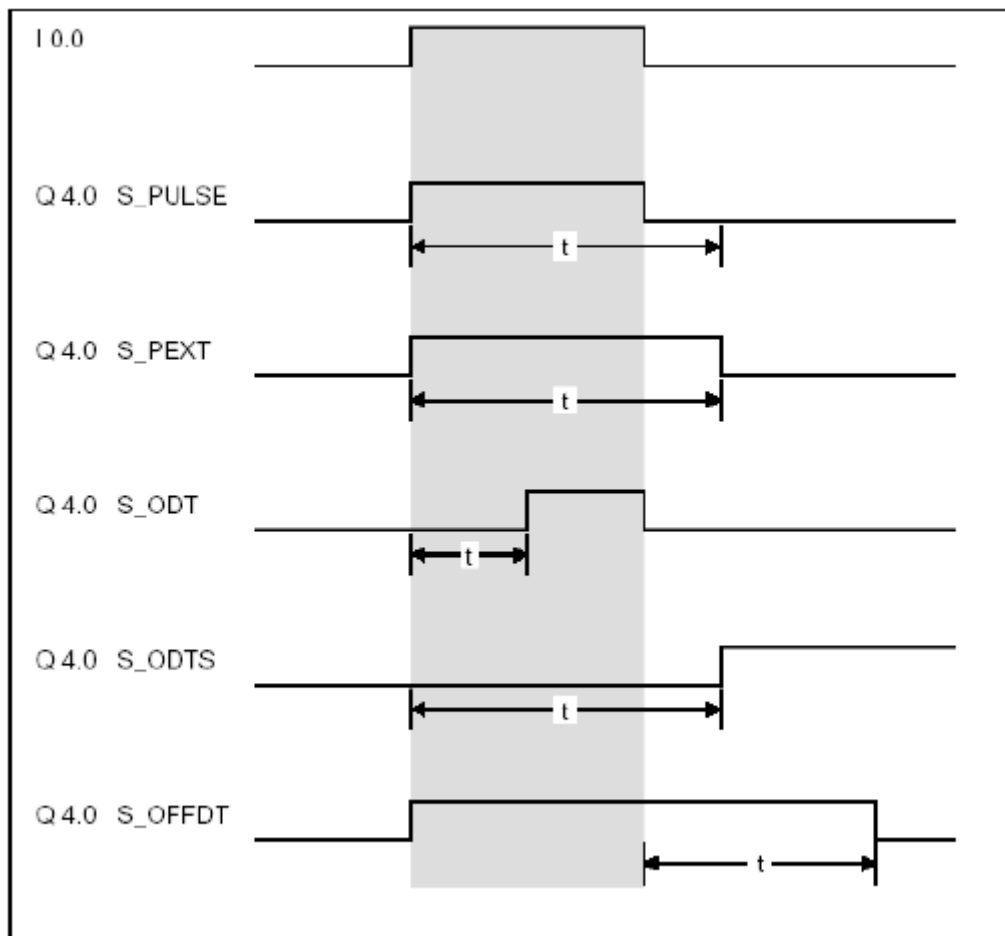
ss

sf

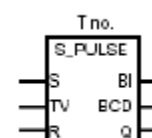


برای دریافت فایل Word پروژه به سایت **ویکی پاور** مراجعه کنید. فاقد آرم سایت و به همراه فونت های لازم

ابتدا مقایسه ای کلی در مورد تایمر ها و تفاوت های بین آنها از روی شکل در زیر ارائه می دهیم.

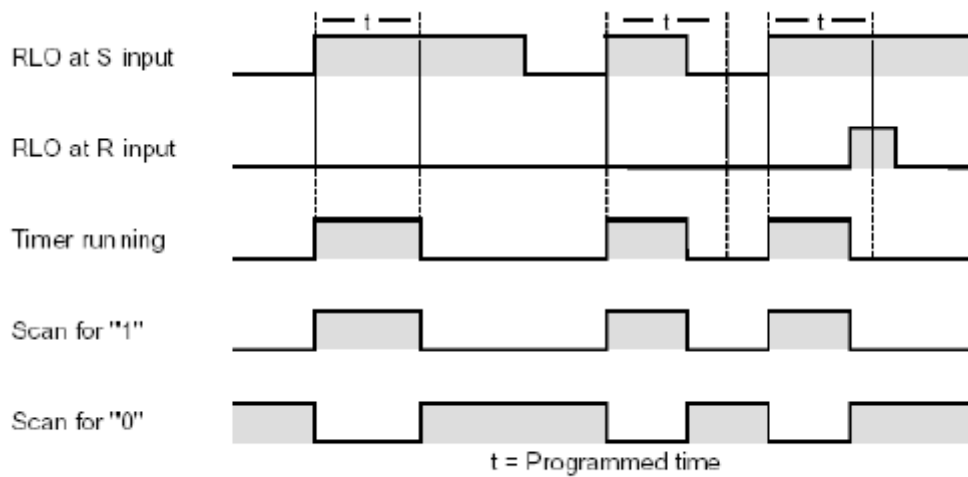


: s-puls



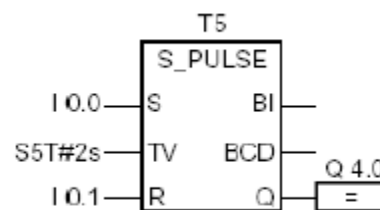
پاسخ این تایمر را به ورودی های مختلف set , resetv را ملاحظه می کنیم:

برای دریافت فایل Word پروژه به سایت **ویکی پاور** مراجعه کنید. فاقد آرم سایت و به همراه فونت های لازمه



که پالس rlo می آید و با توجه به reset پاسخ را ملاحظه می کنیم:

تایمر با توجه به پالس به مقدار زمان خود روشن است و پس از پایان یافتن زمانش به پایان می رسد



مثال:

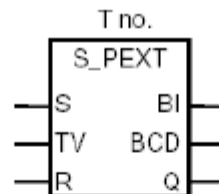
در ابتدا باید در مورد نام گذاری تایمر ها و چگونگی زمان دادن به انبه نکات زیر توجه کرد

Time Base	Binary Code for the Time Base
10 ms	00
100 ms	01
1 s	10
10 s	11

برای دریافت فایل Word پروژه به سایت ویکی پاور مراجعه کنید. فاقد آرم سایت و به همراه فونت های لازم

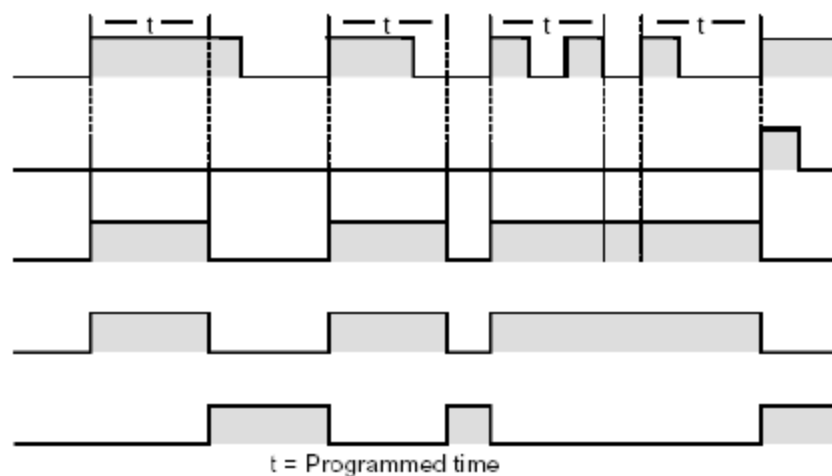
شکل باینری زمانها به شکل با لاست s5t شکل استاندارد نمایش تایمر هاست زمان مورد نظر به شکل #5 نشان داده می شود که در پایه tv نشان داده میشود در مثال بالا با تحریک i0.0 یعنی set تایمر فعال و دو ثانیه می شمارد و با فعال شدن reset دیگر نمی شمارد خروجی در q4 ریخته می شود و با دو فرم باینری و bcd نمایش میدهد

: s-pext



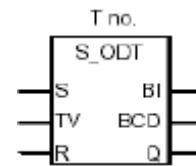
Parameter English	Parameter German	Data Type	Memory Area	Description
no.	Nr.	TIMER	T	Timer identification number. The range depends on the CPU.
S	S	BOOL	I, Q, M, D, L, T, C	Start input
TV	TW	S5TIME	I, Q, M, D, L or constant	Preset time value (range 0-9999)
R	R	BOOL	I, Q, M, D, L, T, C	Reset input
BI	DUAL	WORD	I, Q, M, D, L	Time remaining (value in Integer format)
BCD	DEZ	WORD	I, Q, M, D, L	Time remaining (value in BCD format)
Q	Q	BOOL	I, Q, M, D, L	Status of the timer

تا انتهای تایم میرود و منتظر لبه بالا رونده برای تحریک بعدی می ماند و اگر لبه نیاید ادامه می یابد

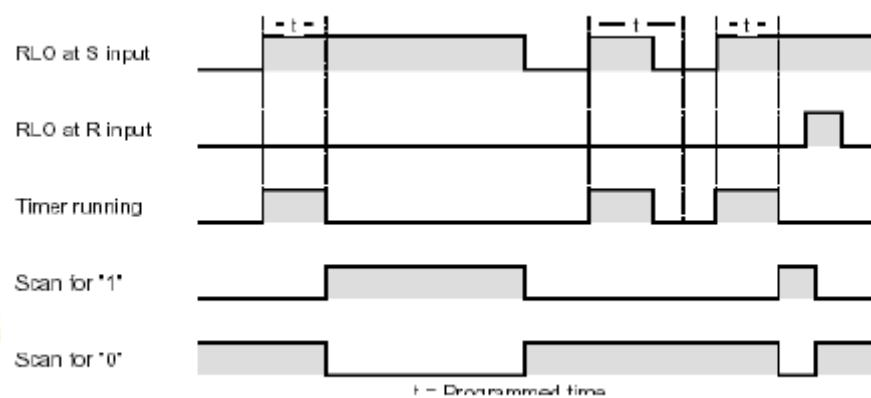


: s-odt

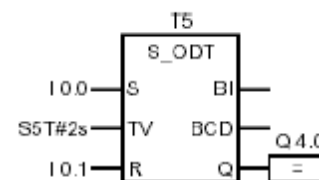
برای دریافت فایل Word پروژه به سایت ویکی پاور مراجعه کنید. فاقد آرم سایت و به همراه فونت های لازم



با تحریک تایمر پالس به اندازه زمان خود میآید و منتظر لبه بالا رونده بعدی می ماند و تایمر بدون توجه به اندازه پالس با توجه به لبه تحریک می شود

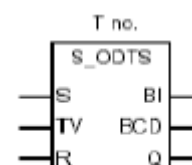


مثال:



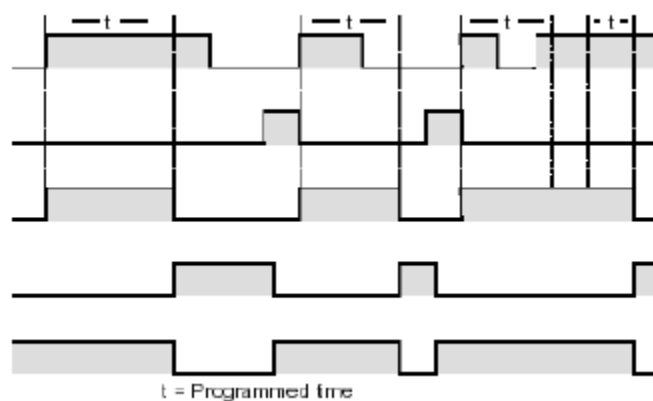
با تحریک i0.0 تایمر به کار می افتد و اگر دو ثانیه به پایان رسد و i0.0 همچنان فعال باشد q4.0 همچنان یک است و اگر i0.0 تغییر حالت بدهد q4.0 صفر می شود اگر i1.0 تغییر حالت بدهد از صفر به یک در زمانیکه تایمر کار می کند تایمر restart می شود.

s-odts



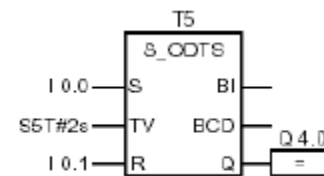
برای دریافت فایل Word پروژه به سایت **ویکی پاور** مراجعه کنید. فاقد آرم سایت و به همراه فونت های لازم

نکته در این تایمر بعد از آمدن فرمان تایمر مورد نظر را سپری می کند و تا آمدن لبه بالا رونده ادامه می یابد و اگر لبه نیاید پالس هم نداریم.

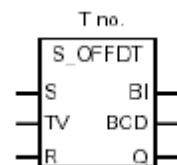


مثال:

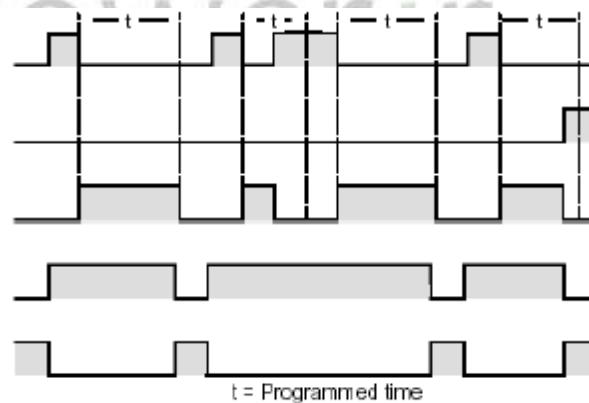
برای دریافت فایل Word پروژه به سایت ویکی پاور مراجعه کنید. فاقد آرم سایت و به همراه فونت های لازم



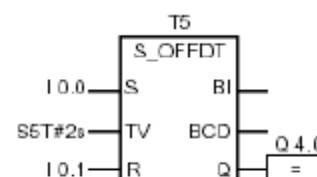
وقتی که i0.0 از صفر به یک تغییر حالت دهد تایمر بر خلاف تغییر سیگنال عمل می کند یعنی اگر i0.0 از صفر به یک تغییر حالت پیدا کند بعد از گذشت زمان t تایمر reset می شود و q4.0 در زمان تایمر تمام شده و i0.1 صفر است روشن است
: s-offdet



این تایمر با لبه پایین رونده فعال می شود و تا پایان time می رود اگر در این زمان پالس پایین رونده دیگری بیاید پالس به پایان می رود و اگر نه تا انتهای پالس می رود.



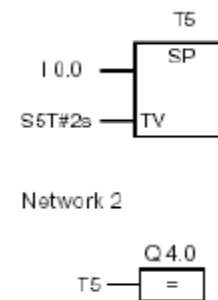
مثال:



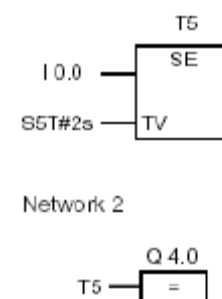
برای دریافت فایل Word پروژه به سایت ویکی پاور مراجعه کنید. فاقد آرم سایت و به همراه فونت های لازم

اگر i0.0 از یک به صفر تغییر حالت بدهد تایمر به کار می افتد و q4.0 زمانی یک است که i0.0 یک باشد یا تایمر در حال کار باشد و i0.1 اگر در زمان کار کرن تایمر فعال شود تایمر restart می شود
: Sp(start puls timer)

مثال:



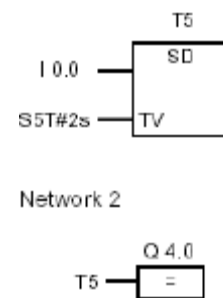
اگر i0.0 از یک به صفر تغییر کند تایمر به کار می افتد و تا زمانیکه signal state یک است تایمر می شمارد و تایمر مرتباً پالس مربعی با فواصل معین ایجاد می کند
: Se(start extended puls timer)



اگر i0.0 از یک به صفر تغییر کند تایمر به کار می افتد و بدون تاثیر از لبه پایین رونده ادامه می یابد و اگر i0.0 بعد از اتمام تایم فعال شود تایمر restart می شود.
: Sd(start on delay timer)

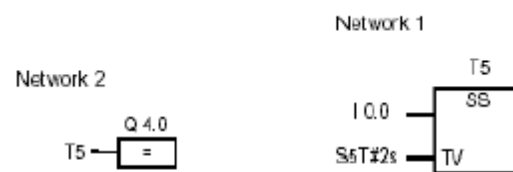
مثال:

برای دریافت فایل Word پروژه به سایت ویکی پاور مراجعه کنید. فاقد آرم سایت و به همراه فونت های لازم



اگر i0.0 از یک به صفر تغییر کند تایمر به کار می افتد و اگر i0.0 یک باشد و تایمر نیز از شمارش خارج q4.0 یک است اگر signal state از یک به صفر تغییر کند تایمر restart می شود
: Ss(start retentive on delay)

مثال:



: Sf(start off delay timer)

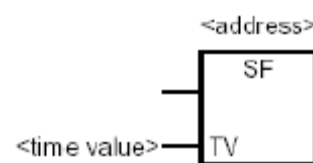
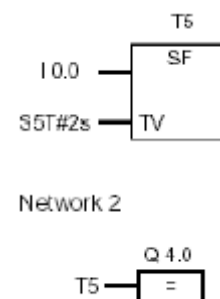


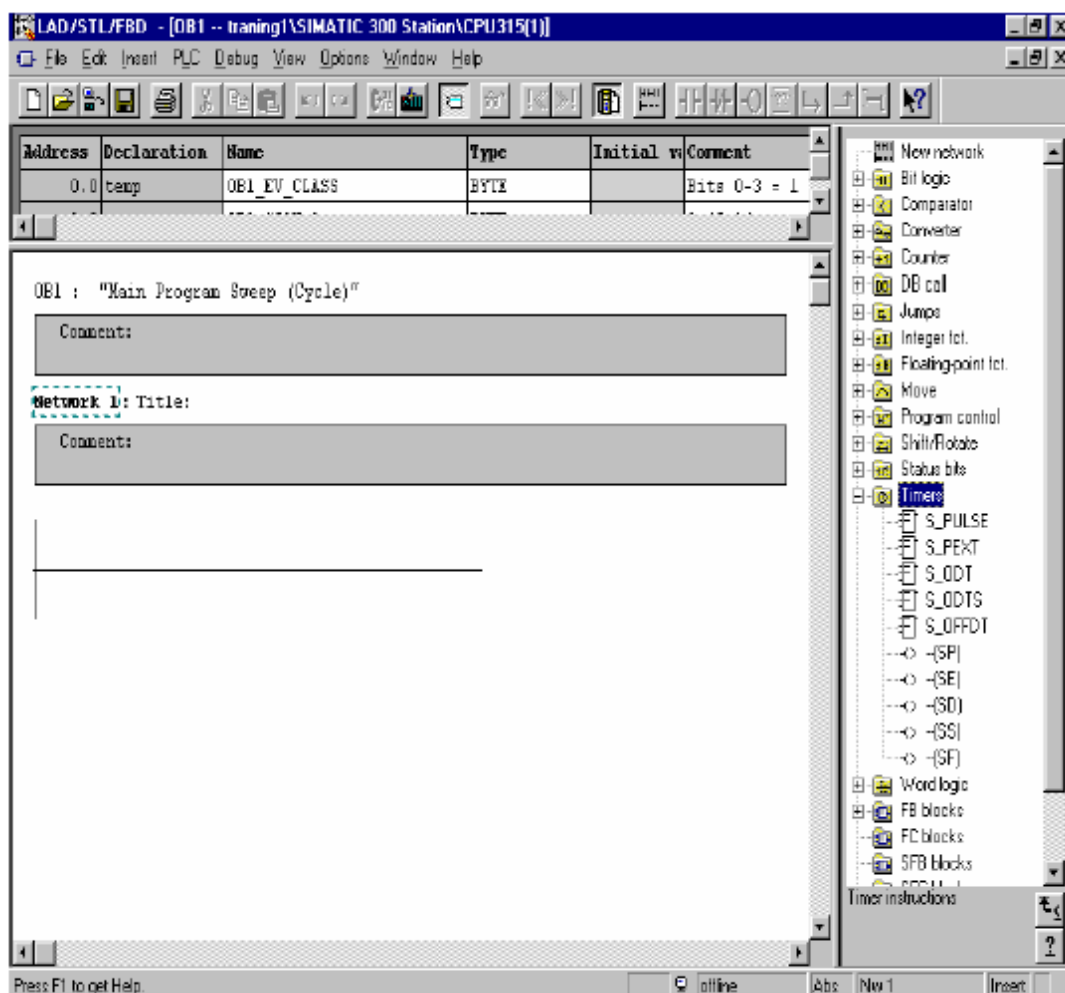
Figure 1

مثال:



برای دریافت فایل Word پروژه به سایت **ویکی پاور** مراجعه کنید. فاقد آرم سایت و به همراه فونت های لازم

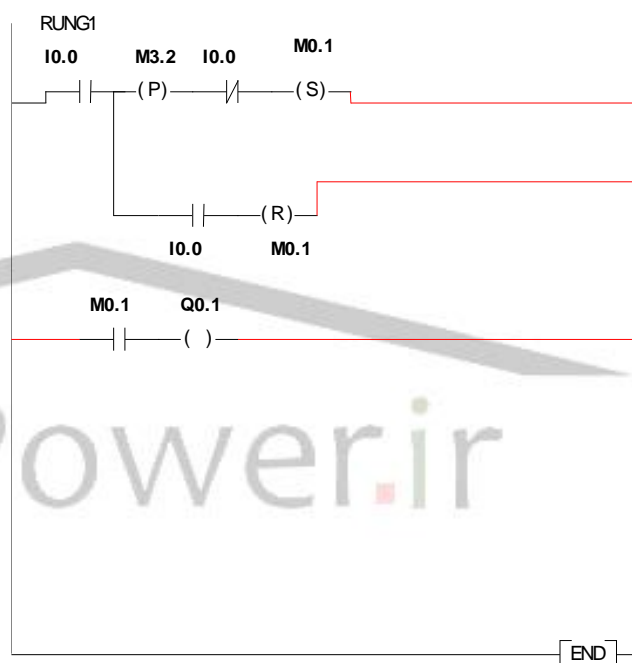
زمانیکه سیگنال state از یک به صفر تغییر کند تایمر به کار می افتد و اگر از صفر به یک برود تایمر restart می شود q4.0 زمانی یک استکه یا i0.0 یک باشد یا تایمر در حال کار باشد در شکل زیر نمای تایمر ها را در برنامه s7 در قسمت library ملاحظه می کنیم.



برای دریافت فایل Word پروژه به سایت ویکی پاور مراجعه کنید. فاقد آرم سایت و به همراه فونت های لازم

نمونه مثال های حل شده در S7 :

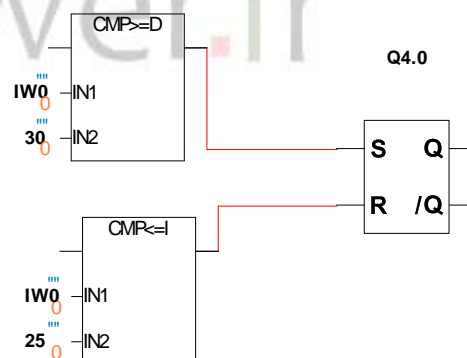
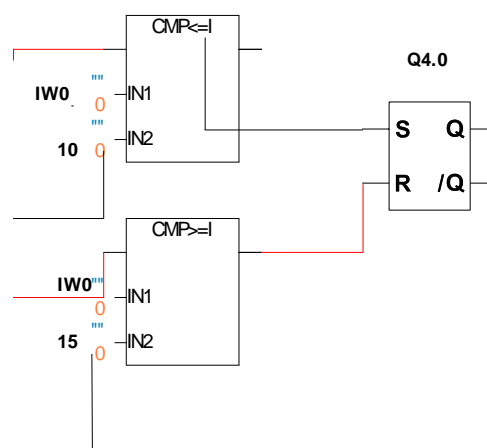
مثال 1: برنامه ای بنویسید که با فشار دادن شاسی در اتوبوس باز شود و با فشار دادن مجدد آن در بسته شود



مثال 2: برنامه ای بنویسید که کولر را در 30 درجه روشن و در 25 درجه خاموش کند و هیتر را در 10 درجه روشن و در 15 درجه خاموش کند:

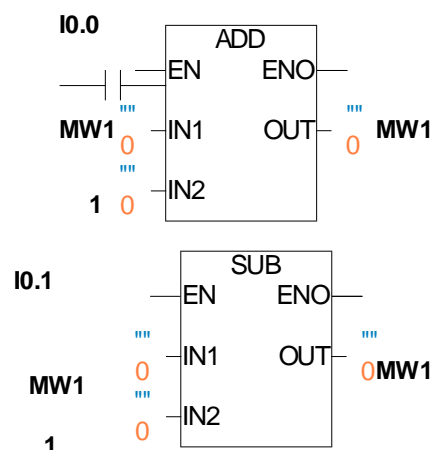
ورودی دما TW0 خروجی کولر و Q4.0 خروجی هیتر Q4.1

برای دریافت فایل Word پروژه به سایت ویکی پاور مراجعه کنید. فاقد آرم سایت و به همراه فونت های لازم



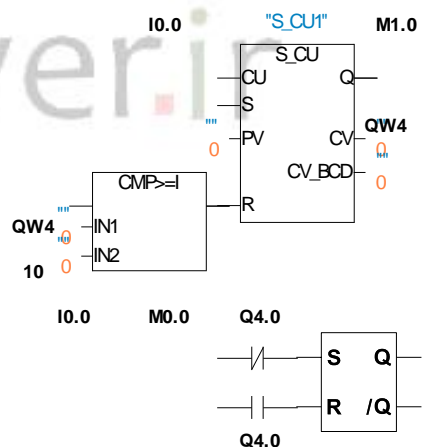
مثال 3: دو شاسی داریم اگر دستمان را روی شاسی یک قرار دهیم مقدار عددی که نشان داده می شود تا زمانی که دستمان روی شاسی است مقدارش زیاد می شود و اگر دستمان را روی شاسی دوم بگذاریم مقدارش کم می شود و وقتی که دستمان را بر می داریم مقدارش ثابت بماند:

برای دریافت فایل Word پروژه به سایت ویکی پاور مراجعه کنید. فاقد آرم سایت و به همراه فونت های لازم



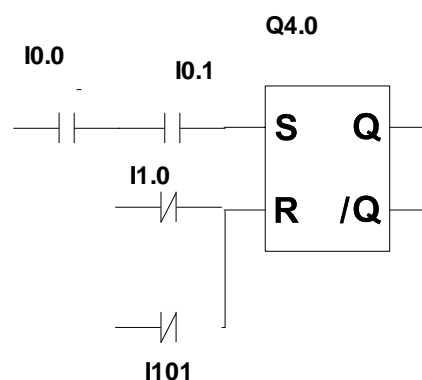
مثال 4: یک دستگاه بسته بندی داریم به ازای هر قطعه که رد می شود یکی می شمارد و در هر جعبه ده تا جا می گیرد:

IO.0 سنسور و QW4 نتیجه میباشد.

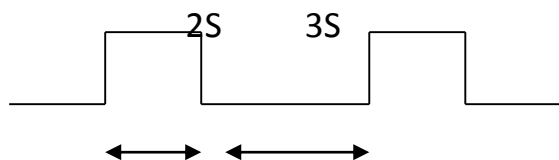


مثال 5: دستگاهی داریم دو تا استارت هم زمان دارد زمانیکه هر دو استارت را زدیم روشن می شود و با E-STOP, STOP خاموش می شود

برای دریافت فایل Word پروژه به سایت ویکی پاور مراجعه کنید. فاقد آرم سایت و به همراه فونت های لازم

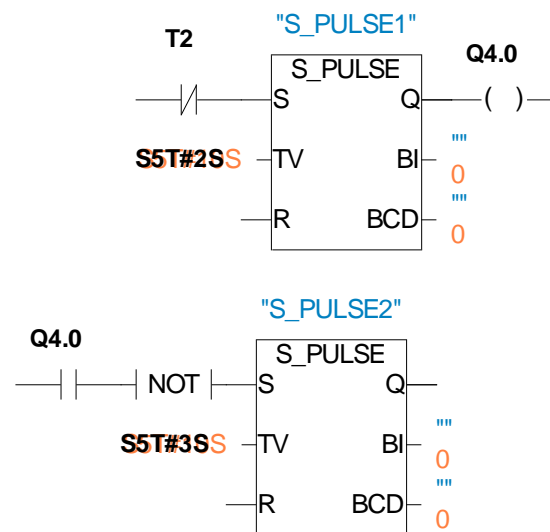


مثال 6: با استفاده از تایمر برنامه ای بنویسید که Q4.0 را به صورت زیر خاموش روشن کند:

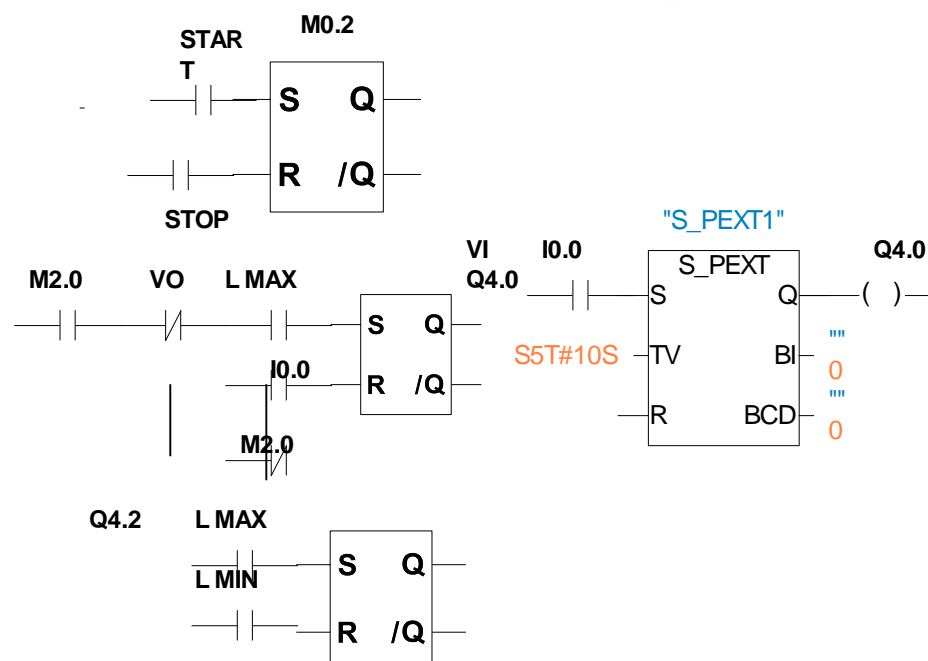


WikiPower.ir

برای دریافت فایل Word پروژه به سایت ویکی پاور مراجعه کنید. فاقد آر م سایت و به همراه فونت های لازم



مثال 7: یک مخزن موجود است که در بالا و پایین آن شیر های برقی وجود دارد داخل این مخزن یک همزن است وقتی VI باز می شود مواد از طریق این شیر داخل مخزن می شوند و تا زمانیکه سطح مواد به L MAX برسد شیر باز می باشد (L MAX یک لیمیت سویچ می باشد) وقتی به آن سطح رسید شیر بسته می شود و به مدت 5 ثانیه هم زن مواد را هم می زند و بعد شیر VO باز می شود و مواد تا زمانیکه به سطح L MIN برسد تخلیه می شود به محض اینکه به L MIN رسید شیر VO بسته می شود و VI باز می شود:



برای دریافت فایل Word پروژه به سایت ویکی پاور مراجعه کنید. فاقد آرم سایت و به همراه فونت های لازم

ضمیمه:

لیست دستورات STL در S7-300 و S7-400 :

English mnemonics	Program elements	description
+	Integer math instruction	Add integer constant(16,32-bit)
=	Bit logic instruction	assign
)	Bit logic instruction	Nesting closed
+ar1	accumulator	Ar1 add accu 1 to adress register 1
+ar2	accumulation	Ar 2 add accu 1 to adress register 2
+d	Integer math instruction	Add accu 1 and accu 2 as double integer(32-bit)
_d	Integer math instruction	Multiply accu 1 and accu 2 as double integer(32-bit)
*d	Integer math instruction	Divide accu 2 by accu 1 as double integer (32-bit)= =,<>,>,<,>=,<=
/d	Integer math instruction	Add accu 1 and accu 2 as integer (16-bit)
?d	Compare	Subtract accu 1 from accu 2 as integer(16-bit)
+i	Integer math instruction	Multiply accu 1 and accu 2 as integer(16-bit)
_i	Integer math instruction	Divide accu 2 by accu 1 as floating-point number (16-bit)
*i	Integer math instruction	Compare integer (16-bit)= =,<>,>,<,>=,<=

برای دریافت فایل Word پروژه به سایت ویکی پاور مراجعه کنید. فاقد آرم سایت و به همراه فونت های لازم

/i	Integer	math instruction	Add accu 1 and accu 2 as a floating-point number (32-bit ieee-fp)
?i		Compare	Subtract accu 1 from accu 2 as a floating-point number(32-bit ieee-fp)
+r	Floating	point instruction	Multiply accu 1 and acu 2 as floating-point
*r	Floating	point instruction	Divide accu 2 by accu 1 as floating-point number (32-bit ieee-fp)
/r	Floating	point instruction	Number 932-bit ieee-fp)
?r		Compare	Compare floating-point number(32-bit)= =,<>,>,<,>=,<=
a		Bit logic instruction	And
A(Bit logic instruction	And with nesting open
abs	Floating	point instruction	Absolute value of a floating-point number (32-bit ieee-fp)
acos	Floating	point instruction	Generate the are cosine of a floating -point number(32-bit)
ad		Word logic intrsuction	And double word (32-bit)
an		Bit logic instruction	And not
An(Bit logic instruction	And not with nesting open
asin	Floting	point instruction	Generate the aic sine of a floating-point number(32-bit)
atan	Floting	point instruction	Generate the arc tangent of a floating -point number(32-bit)
aw		Word logic instruction	And word 916-bit0
be		Program control	Block end

برای دریافت فایل Word پروژه به سایت ویکی پاور مراجعه کنید. فاقد آرم سایت و به همراه فونت های لازم

bec	Program control	Block end conditional
beu	Program control	Block unconditional
bld	Program control	Program display instructional(null)
btd	convert	Bcd to integer (32-bit)
btl	convert	Bcd to integer(16-bit)
call	Convert	Chenge byte sequence in accu 1(32-bit)
English mnemonics	program element catalog	description
CALL	Program control	Call multiple instanc
CALL	Program control	Block call
CALL	Program control	Call block from a library
CAR	Load/transfer	Exchange address register 1 with address registser2
CAW	Convert	Cheng bye sequence 1 in accu 1-l(16-bit)
CC	Program control	Conditional call
CD	counters	Contter down
CDB	convert	Exchange shared db and instance db
CLR	Bit logic instraction	Clear rlo(=0)
COS	Floatiog point instraction	Generate the cosine of angles as floating-point number
CU	counters	Counter up
DEC	Accumulator	Decrement accu1-L-L
DTB	Convert	Double integer (32)to bcd
DTR	Convert	Convert double integer (32) to floating - point(32)
ENT	Accumulator	Enter accu stack
EXP	loating point instraction	Generate the exponential value of a

برای دریافت فایل Word پروژه به سایت ویکی پاور مراجعه کنید. فاقد آرم سایت و به همراه فونت های لازم

		floating-point number(32)
FN	Bit logic instraction	Edge negative
FP	Bit logic hnstraction	Edge positive
FR	counters	Enable counter
FR	Timers	Enable timer
INC	accumulator	Increment accu1-L-L
INVD	convert	Ones complement double integer
INVI	Convert	Ones compelement integer (16)
ITB	Convert	Integer to bcd
ITD	Convert	Integer to double integer
JBI	Jumps	Jump if BR=1
JC	Jumps	Jump ifRLO=1
JCB	Jumps	Jcb spbb jumps jump ifRLO=1withBR
JCN	Jumps	Jump if RLO=0
JL	Jumps	jumps jump TO labels
JM	Jumps	Jump if minus
JMZ	Jumps	Jump if minus or zero
JN	Jumps	Jump if not zero
JNB	Jumps	Jump if RLO=0with BR
JNBI	Jumps	Jump if BR=0
JO	Jumps	JUMP IF ov=1
JOS	Jumps	JUMP IF os=1
JP	Jumps	Jump if plus
JPZ	Jumps	jump if plus or zero
JU	Lums	Jump unconditional
JUO	Jums	jump if unordered
JZ	Jums	jump if zero
L	Load/transfer	Load
L DBLG	Load/transfer	Load length of shared db in accu 1
L DBNO	Load/transfer	Load number of db in accu 1

برای دریافت فایل Word پروژه به سایت ویکی پاور مراجعه کنید. فاقد آرم سایت و به همراه فونت های لازم

L DILG	Load/transfer	Load length of instance db in accu 1
L DINO	Load/transfer	Load number of instance db in accu 1
L STW	Load/transfer	Load status word into accu 1
L	Timers	Load current timer value into accu 1 as integer (the current time value can be a number from 0 to 255, for exmapel,lt32)
L	Counters	Load current counter value into accu1 (the current conter value can be number from 0 to 225,for exampel,l c 15)
L AR1	Load/ transfer	Load address register 1 from accu1
LAR1<D>	Load/transfer	Load adrses register 1 with doble integer (32-bit pointer)
LAR1 AR2	Load/transfer	Load address resgistr 1 from address register2
LAR2	Load/transfer	Load address register 2 from accu1
LAR2<D>	Load/transfer	Load address regisiter 2 with double integer (32-bit pointer)
LC	counters	Load current counter value into accu 1 as bcd(thurrent timer value can be a number from 0 to 255.for exampel,lc c 15)
LC	timer	Load current timer value into accu 1 as bcd(the curent counter

برای دریافت فایل Word پروژه به سایت ویکی پاور مراجعه کنید. فاقد آرم سایت و به همراه فونت های لازم

		alue can be a number from 0 to 255,for exampel,lct32)
LEAVE	accumulator	Leave accu atack
LN	Floating point instruction	Generathe natural logarthm of a floathing-point number (32-bit)
LOOP	Jumps	Loop
MCR(Program control	Save rlo in mcr stack,begin mcr
)MCR	Program control	End mcr
MCRA	Program control	Activate mcr area
MCRD	Program convert	Deactivate mcr area
MOD	integer math instraction	Division remainder double integer(32-bit)
NEGD	convert	Twos complement double integer(32-bit)
NEGI	convert	Twos compelement integer (16-bit)
NEGR	Convert	Negate floating-point number(32-bit,ieee-fp)
NOPO	accumulator	Null instruction
NOP1	accumulator	Null instruction
NOT	Bit logic instruction	Negate rlo
O	Bit logic instraction	or
O(Bit logic instraction	Or with nesting open
OD	Bit logic instraction	Or double word(32-bit)
ON	Bit logic instruction	Or not
ON(Bit logic instraction	Or not with nesting open
OPN	Db call	Open a data block
OW	Word logic instraction	Or word(16-bit)
POP	accumulator	Cpu with two accus
POP	accumulator	Cpu with two accus
POP	accumulator	Cpu with four accus
PUSH	accumulator	Cpu with four accus
R	Bit instraction	reset

برای دریافت فایل Word پروژه به سایت ویکی پاور مراجعه کنید. فاقد آرم سایت و به همراه فونت های لازم

R	counters	Reset counter (the current counter can be a number from 0 to 255, for example, r c 15)
R	timers	Reset timer (the current timer can be a number from 0 to 255, for example, r t 32)
RLD	Shift/rotate	Rotate left double word (32-bit)
RLDA	Shift/rotate	Rotate accu 1 left via cc 1 (32-bit)
RND	convert	Round
RND-	convert	Round to lower double integer
RND+	convert	Round to upper double integer
RRD	Shift/rotate	Rotate right double word (32-bit)
RRDA	Shift/rotate	rotate accu 1 right via cc 1 (32-bit)
S	Bit logic insert	set
S	counters	Set counter preset value (the current counter can be a number from 0 to 255, for example, s c 15)
SAVE	Bit logic instruction	Save rlo in br register
sd	timers	On-delay timer
SIN	Floating point instruction	Generate the sine of angles as floating-point number (32-bit)
SLD	Shift/rotate	Shift left double word (32-bit)
SLW	Shift/rotate	Shift left word (16-bit)
SP	Timers	Pulse timer

برای دریافت فایل Word پروژه به سایت ویکی پاور مراجعه کنید. فاقد آرم سایت و به همراه فونت های لازم

SQR	Floating point	Generate the square of floating –point number (32-bit)
SQRT	Floating point instruction	Generate the square root of a floating –point number (32-bit)
SRD	Shift/rotate	Shift right double word (32-bit)
SRW	Shift/rotate	Shift right word (16-bit)
SS	Timers	Retentive on-delay timer
SSD	Shift//rotate	Shift sing double integer (32-bit)
SSI	Shift/rotate	Shift sing integer (16-bit)
T	Load/transfer	transfer
T	Load/transfer	Transfer accu 1 into status word
TAK	accumulator	Toggle accu 1 eihth accu 2
TAN	Floating point instruction	Generate the tangent of angles as floating –point number (32-bit)
TAR1	Load/transfer	Transfer address re
TAR1	Load/transfer	Transfer address register 1 to accu 1
TAR1	Load/transfer	Transfer address register 1 to address register2
TAR2	Load/transfer	Transfer address register 2 to accu 1
TAR2	Load/transfer	Transfer address register 2 to destination (32-bitpointer)
TRUNC	Convert	truncate
UC	Program control	Unconditional call
X	Bit logic instruction	Exclusive or

برای دریافت فایل Word پروژه به سایت ویکی پاور مراجعه کنید. فاقد آرم سایت و به همراه فونت های لازم

X(Bit logic instruction	Exclusive or with nesting open
XN	Bit logic instruction	Exclusive or not
XN(Bit logic instruction	Exclusive or not with nesting open
XOD	Word logic instruction	Exclusive
XOW	Word logic instruction	exclusive or word (16-bit)



برای دریافت فایل Word پروژه به سایت ویکی پاور مراجعه کنید. فاقد آرم سایت و به همراه فونت های لازم

فهرست منابع:

- [۱] ماهر, محمد رضا, "راهنمای جامع step 7", پیشگامان علم و صنعت آریا, ۱۳۸۵
- [2] قابوسی, فرید, "مرجع کامل PLC" انتشارات ارکان, ۱۳۸۰
- [3] یادگار توچالی, محمد, "نحوه استفاده و آشنایی با PLC خانواده زیمنس و نرم افزار مربوطه" اداره پشتیبانی و مهندسی تعمیرات ایران خودرو, ۱۳۸۴
- [4] برنده فرد, محسن, "آموزش سطح یک PLC S7" مرکز آموزش ایران خودرو, ۱۳۸۵
- [5] رحمانی, محمد علی, "PLC های زیمنس STEP7" سایت آموزش ایران خودرو, ۱۳۸۵
- [6] نحوی, محمد, "جزوه درسی آموزش PLC و زبان برنامه نویسی LD" WWW.ECR.IR, ۱۳۸۴