بسمه تعالى

آموزشی PLC SIEMENS S7-300

WWW.AMANDEGARI.IR

جلسه اول

هدف:

- کلیاتی در معرفی PLC
- معرفی تجهیزات مطرح در یک سیستم خودکار
 - سخت افزار یک PLC
 - معرفی پروگرامر (PG) و نرمافزار
 - معرفی HMI
- معرفی تجهیزات سیستم خودکار جدا کننده مهرههای سیاه و سفید موجود در آزمایشگاه
 - معرفی سخت افزارهایPLC روی میزهای کار آزمایشگاه
 - نحوه آدرس دهی ماژولهای ورودی، خروجی و حافظه

۱–۱) مقدمه

کلیاتی در معرفی PLC ^۱ در یک سیستم خودکار ۱-۱-۱) جایگاه PLC ^۱ در یک سیستم خودکار PLC وسیلهای است که با آن میتوان عملیات مربوط به فرایند^۲های مختلف را بطور خودکار اجرا و کنترل کرد. کنترل فرایند با PLC ، میتواند بصورت حلقه باز و یا حلقه بسته انجام شود. برای مثال در خط تولید نوشابه از PLC برای خودکار کردن فرآیند تولید استفاده میشود. یا در نیروگاهها، برای تولید برق، عملیات تولید بکمک PLC بصورت خودکار انجام میشود. بطور کلی در خیلی از فرآیندهای شمیائی، حرارتی، مکانیکی و .. از PLC برای خودکارکردن عملیات فرآیند استفاده میشود.

PLC انتخاب (۲-۱-۱

PLC های سازنده های مختلف (−۱−۱) تفاوت کلی PLC

توانائیهای PLC های سازندگان مختلف، برای اجرای کارهای عمومی مشابه هستند. تفاوت آنها در پاسخ به حجم حافظه، سرعت پردازش، تعداد ورودی خروجی، قابلیت داشتن سخت افزار و نرم افزار برای اجرای امور خاص و پاسخ دهی در شرایط و محیطهای خاص است.

اگر ما با یکی از PLCهای یک سازنده آشنا شویم، به آسانی با کمی تلاش میتوانیم PLCهای دیگر سازندگان را هم بکار بگیریم چون اصول عملکرد همه آنها مشابه است. در این آزمایشگاه با PLC سری 300-S7 شرکت زیمنس آشنا خواهیم شد که نرم افزار برنامه ریزی آن Simatic Manager است.

¹ Programmable Logic Controller ²Process

۲-۱) تجهیزات مطرح در یک سیستم خودکار

طرح بلوکی شکل (۱-۱) تجهیزات مطرح در یک سیستم خودکار را که با PLC کار میکنند نشان میدهد. این تجهیزات شامل، PLC ،Plan و ۲MI هستند

: Plant (1-Y-1

منظور از Plant در یک سیستم خودکار، ماشین یا مجموعهای از ماشین های صنعتی هستند که برای اجرای هدف و یا هدفهای تعریف شدهای بطور خودکار عمل میکنند.

در هر Plant یک سیستم خودکار مجموعهای از ابزارها به نامهای حسگر^۳ و عملگر^۴ وجود دارند. که با ارتباط این ابزارها با یک کنترل کننده مرکزی شرایطی فراهم میشود، که در آن سیستم اجرای عملیات یک پروسه بصورت خودکار امکان پذیر میشود. حسگرها اطلاعات Plant را به کنترل کننده مرکزی منتقل و عملگرها فرامین را برای اجرای عمل مربوطه، از آن دریافت میکنند. حسگرها و عملگرها توسط طراح ماشین انتخاب و در محلهای مورد نظر نصب میشوند. در اینجا کنترل کننده مرکزی یک PLC است



شکل (۱–۱) طرح بلوکی امکانات مطرح در یک سیستم خودکار

PLC) سخت افزار

ساختار یک PLC ، بصورت یکپارچه^۵ و یا بصورت ماژولار^۶ است. معمولا PLC ها با قابلیتهای محدود بصورت یک-پارچه و PLC با توانائیهای بالا بصورت ماژولار ساخته میشوند. در اولی اجزای تشکیل دهنده آن بصورت سختافزاری یک پارچه و در دومی بصورت جدا جدا طراحی و ساخته میشوند. اجزای تشکیل دهنده PLCهای ماژولار با کانکتورهای مناسب بهم وصل و در کنار هم برروی ریل نصب میشوند.

¹ Programmer

- ² Human Machine Interface
- ³ Sensor
- [•]Actuator
- ⁵ Compact
- ⁶ Modular

در طراحی سخت افزار یک PLC لازم است اجزای تشکیل دهنده آن، به درستی انتخاب و مورد استفاده قرار گیرند.که این اجزاء شامل CPU، ماژولهای ورودی - خروجی، ماژولهای خاص مورنیازسیستم، منبع تغذیه و تجهیزات ارتباطی بین CPU و دیگر عنصر کنترلی Plant میباشند

انتخاب ⁽CPU

برای انتخاب یک CPU لازم است تمام سخت افزارها و نرم افزارهائی که باید توسط آن (برای کنترل یـک Plant) پشـتیبانی شود، مورد توجه قرار گیرد. بر این اساس باید CPU ای انتخاب شود، تا پارامترهای آن، نیاز سخت افزار و نـرمافـزار مـورد نظـر را پوشش دهد. پارامترهای مهم یک CPU شامل:

- سرعت - حجم حافظه برنامه و داده - پشتیبانی تعداد ورودی خروجی - پشتیبانی تعداد Timer و Counter - پشتیبانی از قابلیتهای خاص دیگر، مانند شبکه شدن، پاسخگوئی به وقفهها، دنبال کردن مکان و اندازهگری فرکانس بالا، شمارش سرعت بالا و قابلیتهای دیگر... است.

انتخاب ورودهای دیجیتال^۲

انتخاب ورودیهای دیجیتال، بر اساس تعداد و مقدار ولتاژ حسگرها (Sensor) دیجیتالی و تعدادکلیـدهای موجـود در Plant انتخاب میشوند. پارامترهای عمومی برای انتخاب یک ماژول ورودی دیجیتال شامل:

مقدار ولتاژ ورودی
 نوع جدا سازی^۳

است.

ورودیهای دیجیتال با ساختار ماژولار معملا در گروهای ۸ و یا ۱۶ و یا ۳۲ تائی ارائه میشوند.

انتخاب خروجي ديجيتال

انتخاب خروجیهای دیجیتال، بر اساس تعداد، مقادیر جریان و ولتاژ عملگرها دیجیتالی و لامپ سیگنالها بکار گرفتـه شـده در Plant، انتخاب میشوند. پارامترهای عمومی برای انتخاب یک مدول خروجی دیجیتال شامل:

مقدار ولتاژ خروجی
مقدار جریان خروجی
نوع جدا سازی
نوع حفاظت

است.

خروجیهای دیجیتال با ساختار ماژولار معمولا در گروهای ۸ و یا ۱۶ و یا ۳۲ تائی ارائه میشوند. سازندگان PLC ماژولهائی با ۸ ورودی و ۸ خروجی دیجیتال هم به بازار عرضه میکنند.

- ² Digital
- ^r Isolation

انتخاب ورودی آنالوگ^ا

انتخاب ورودیهای آنالوگ، بر اساس تعداد حسگرهای آنالوگ استفاده شده در Plant، نوع کارکرد آنها با ولتاژ و یا جریان، و همچنین تعداد تنظیم کنندههای ولتاژی و یا جریانی استفاده شده در Plant انتخاب میشوند. پارامترهای عمومی برای انتخاب ورودهای آنالوگ شامل: - سیگنال دیتای ورودی از نوع ولتاژ یا جریان 0 در نوع ولتاژ، مقادیر گستره ولتاژ ۰ تا ۱۰ ولت و یا ۱۰- تا ۱۰+ ولت است. - نوع جدا سازی - دوت جردان ر تعداد بیت مبدل آنالوگ به دیجیتال) است.

انتخاب خروجىهاى آنالوگ

انتخاب خروجیهای آنالوگ، بر اساس تعداد عملگرهای آنالوگ استفاده شده در Plant، نوع کارکرد آنها با ولتاژ و یا جریان، و همچنین تعداد نشاندهنده های ولتاژی و یا جریانی استفاده شده در Plant، انتخاب میشوند. پارامترهای عمومی برای انتخاب یک مدول خروجی آنالوگ شامل:

است.

سئوال: مزایای انتقال دیتا به بصورت جریان نسبت به انتقال آن به صورت ولتاژ چیست؟

انتخاب مدول خاص (Function Module)

سازندگان PLC به منظور ایجاد تسهیلات لازم برای کارهای خاص در صنعت اتوماسیون ماژولهای خاصی را عرضه میکنند که بطور مثال شامل:

و است. انتخاب این نوع ماژول ها PID Controller ،Stepper Motor Driver،Counter for Shaft Encoder و است. انتخاب این نوع ماژول ها بر اساس نیازهای موجود در طراحی یک پروژه انجام می شود.

- انتخاب Power Supply

در یک سیستم که از PLC استفاده می شود، معمولا از دو منبع تغذیه استفاده می شود. یکی از آنها برای تغذیه PL و تجهیزات (ماژولها) جانبی آن استفاده می شود که لازم است این منبع تغذیه دقیق با شد و معمولا سازندگان PLC آنرا عرضه می کنند. و منبع تغذیه دیگری برای تغذیه عملگرها و حسگرهای Plant استفاده می شود، با توجه به اینکه گستره پذیر ش ولت اژ می کنند. و منبع تغذیه دیگری برای تغذیه عملگرها و حسگرهای Plant استفاده می شود، با توجه به اینکه گستره پذیر ش ولت اژ می کنند. و منبع تغذیه دیگری برای تغذیه عملگرها و حسگرهای Plant استفاده می شود، با توجه به اینکه گستره پذیر ش ولت اژ می کنند. و منبع تغذیه دیگری برای تغذیه عملگرها و حسگرهای Plant استفاده می شود، با توجه به اینکه گستره پذیر ش ولت اژ اغلب عملگرها و حسگرها نسبتا بزرگ و معمولا توان مورنیاز برای تغذیه این ابزارها در یک Plant زیاد است، به منظور کاه شود می هرد یا توجه به ولتاژها و تعملگرها و تعان مورنیاز برای تغذیه این ابزارها در یک Plant زیاد است، به منظور کاه شود می هرد ای برای تغذیه می شود، با توجه به ولتاژها و توان مورنیاز برای تغذیه این ابزارها در یک Plant زیاد است، به منظور کاه می هزینه ها برای تغذیه این ابزارها در یک Plant زیاد است، به منظور کاه ش مرزیای می می می می می می می و باز و حملگرها از منابع تغذیه خیلی دقیق استفاده نمی شود، و با توجه به ولتاژها و توان ابزارهای موجود در Plant زیاد و با توجه به ولتاژها و توان ابزارهای موجود در Plant منبع تغذیه نه چندان گران قیمت تهیه می شود.

Programmer) PG (Programmer) و نرم افزار برنامهریزی

برای بکارگیری PLCها، نیاز به برنامهریزی آنها است. برای برنامه ریزی آنها نیاز به یک وسیلهای بنام PG بهمراه نرم-افزار برنامهریزی است. از نرمافزار برنامهریزی برای تولید برنامه، انتقال برنامه به PLC، راه اندازی و متوقف کردن اجرای برنامه در PLC، پایش اجرای برنامه و پاک کردن برنامه منتقل شده به PLC استفاده می شود. برای بکارگیری PLCهای هر شرکت سازنده، لازم است نرمافزار برنامهریزی خاص آن PLC تهیه و از آن استفاده شود. PLCهای آزمایشگاه دانشکده از سری 300 -S7 شرکت زیمنس و نرمافزار برنامهریزی آن Simatic Manager استفاده شود. Simatic Manager را می توان به دو روش بکار گرفت. یکی ازروش ها استفاده از PG خاص شرکت زیمنس و روش دیگر استفاده از OP و PC Adapter ماسخت شرکت زیمنس است. در آزمایشگاه از OP خاص شرکت زیمنس و روش دیگر استفاده از OP و PC مامود برای دو کار عمده استفاده می شود. این دو کار مامول برنامهریزی OP ماستفاده می شود. از نرم افزار برنامه ریزی عموماً برای دو کار عمده استفاده می شود. این دو کار

HMI¹ (۴-۲-۱)معرفی

وسائلHMI واسط بین انسان و ماشین هستند. بدین معنی که یک HMI وسیله یا تجهیزاتی است که انسان را به منظور تبادل اطلاعات (از طریق PLC) به ماشین ارتباط میدهد این وسائل یا تجهیزات نظیر:

- کلیدها و لامپ سیگنالها

- Operation Panel) OP) ها

- PC و نرم افزار HMI (مانند Win CC) مىباشند.

تبادل اطلاعات بین تجهیزات ردیف اول (آمده در بالا) و PLC بطور موازی و تبادل اطلاعات تجهیزات دو ردیف دیگر و PLC معمولا به طور سری انتقال انجام می شود.

۲-۱–۵) نحوه تبادل داده بین PLC و دیگر ابزارها و تجهیزات در یک سیستم خودکار

تبادل داده بین ابزارها و تجهیزات موجود در یک سیستم خود کار با PLC بصورت موازی یا سری انجام میشود، برای انتقال داده بصورت موازی، علاوه بر سیمهای منبعتغذیه، برای هر حسگر و یا عملگر حد اقل یک سیم ارتباطی استفاده میشود. ولی برای انتقال داده بصورت سری، برای همه ابزارها و تجهیزات موجود در یک سیستم خودکار فقط دو و یا سه سیم استفاده شده و تبادل داده بین آنها بشکل مالتیپلکس^۲، انجام میشود.

کمپانیهای مختلف، Busهائی با پروتکلهای متفاوت به بازار عرضه کردهاند که به وسیله هر یک از آنها، امکان تبادل داده بین CPU و دیگر تجهیزات یک سیستم خودکار، بصورت مالتی پلکس فراهم میشود. باسهای مطرح در صنیایع اتومسیون شامل:

- Mod Bus
- Fandation Field Bus
 - PROFIBUS -DP •
 - PROFIBUS –PA MPI

هستند،. در آزمایشگاه نحوه به کار گیری PROFIBUS –DP برای ارتباط PLC با تجهیـزات Plant و HMI بررسـی و از بـاس MPI^۳ برای ارتباط Programmer و CPU استفاده میشود.

^{*} Multiplex

¹ Human machine Interface

Moltpoint Interface

۱-۳) معرفی سیستم خودکار جداکننده مهرههای سفید و سیاه موجود در آمایشگاه

- شرح کار سیستم

در آزمایشگاه یک سیستم آموزشی وجود دارد که تصویر آن در شکل(۱-۲) نشان داده شده است. این سیستم برای جدا-سازی مهرههای سیاه و سفید بصورت خودکار است. مهرها توسط کاربر در داخل مخزن Plant ریخته میشوند و در خروجی Plant مهرهای سیاه و سفید بصورت خودکار از هم تفکیک میشوند. این سیستم از چند بخش عمده تشکیل شده که شامل PLC ،Plant مهراه میدند. علاوه بر این بخشها در تصویر یک PC بهمراه یک PC-Adapter دیده میشوند که مجموعا کار یک PG را انجام میدهند. در زیر تجهیزات هر یک از این بخشها معرفی شده اند.

1-۳-۱)معرفی تجهیزات Plant

Plant این سیستم دارای دو عدد حسگر و چهار عدد عملگر دیجیتالی است. یکی از حسگرها در زیر مخزن نصب شده و خارج شدن هر یک از مهرهها از مخزن را مشخص میکند. حسگر دوم در مسیر انتقال مهره در کنار تسمه نقاله نصب شده است که رنگ مهرهها را تشخیص میدهد. هر یک از این حسگرهای دیجیتالی تشخیص خود را بصورت صفر و یا یک از طریق ورودی دیجیتال

از چهار عملگرها موحود در سیستم یکی یک موتور DC است که از طریق خروجی دیجیتال از PLC فرمان گرفته و تسمه نقاله را (برای جابجا کردن مهرهها، از کنار مخزن تا خروجیهای مربوطه) به حرکت در میآورد. سه عملگر دیگر شامل سه شیر نیوماتیکی(جریان هوا) به همراه سه سیلندر و پیستون میباشند، که این شیرها از PLC فرمان دریافت کرده و مسیرهای جریان هوای سیلندر مربوطه را جابجا میکنند، با جابجائی مسیرهای جریان هوا، پیستون در داخل سیلندر به سمت خارج و یا داخل جابجا میشوند. یکی از شیرها برای خارج کردن مهرهها از مخزن بصورت دانه دانه، و دیگری برای جابجا کردن مهره خارج شده از مخزن، از زیر مخزن تا روی تسمه نقاله و آخرین آنها برای تفکیک مهره سیاه و سفید از یکدیگر استفاده میشوند.



شکل (۱-۲) تصویر سیستم جدا کننده مهرههای سیاه و سفید

PLC معرفی تجهیزات) PLC

PLC استفاده شده در این سیستم از سری 300-57 شرکت زیمنس و بصورت ماژولار است. این PLC از چهار ماژول تشکیل شده که بر روی یک ریل نصب شدهاند. ماژول منتهیالیه سمت چپ نصب شده بر روی ریل منبع تغذیه، ماژول کنار آن CPU، ماژول سمت راست CPU، ماژول ورودی- خروجی دیجیتال و آخرین آنها یک ماژول ورودی – خروجی آنالوگ است که از ماژول آنالوگ در این سیستم استفاده نشده است.

HMI معرفی تجهیزات (۳-۱

تجهیزات ارتباطی انسان با ماشین (HMI) این سیستم، شش عدد کلید On-Off و چند چراغ LED هستند. که کلیدها به ورودیهای دیجیتال PLC و LEDها به خروجی دیجیتال PLC متصل میباشند. کلیدها فرامین ماشین را از انسان دریافت و به PLC منتقل میکنند وLEDها وضعیت ماشین را از PLC دریافت و به انسان نشان میدهند.

یکی از این کلیدها برای راه اندازی و متوقف کردن ماشین، دیگری برای انتخاب راهانداز حالت دستی و یاحالت خودکار ماشین و چهار کلید دیگر برای راهاندازی چهار عملگر در حالت دستی استفاده میشوند. یک از LED ها وضعیت خالی بودن مخزن و بقیه LEDها وضعیت هر یک از عملگرها ماشین را به کاربر نشان میدهند.

۴-۱) معرفی سخت افزارهایPLC روی میزهای کار آزمایشگاه

روی هر میز کار آزمایشگاه یک PLC ساخت شرکت زیمنس و از سری 300 -ST قرار دارد که بصورت ماژولار است. ماژولهای این PLC بر روی یک ریل آلمینیومی کنار یکدیگر (توسط یک پیچ که در زیر هر یک است) نصب و هر دو ماژول مجاور از طریق یک کانکتور U شکل از پشت بهم متصل شدهاند. در جال حاضر تعداد ماژولهای نصب شده بر روی این ریل پنج عدد میباشد که تا یازده ماژول قابل افزایش است. تصویر یکی از PLCهای میز آموزشی در شکل (۱–۳) نشان داده شده است. ماژولها این PLC از سمت چپ شامل منبع تغذیه، CPU، ورودی دیجیتال، خروجی دیجیتال، و ورودی – خروجی آنالوگ هستند. در زیر قابلیتهای هر یک از این ماژولها به اختصار معرفی شدهاند.



شکل (۱–۳) تصویر ماژولهای یک PLC بر روی ریل

۱−۴−۱) ماژول منبع تغذیه (PS307-2A):

توان خروجی این منبع 24V/2A و ولتاژ ورودی آن 220V و یا 110V (قابل انتخاب) است. این منبع تغذیه مختص تغذیه CPU و ماژولهای جانبی آن است.

اجزای روی ماژول منبع تغذیه شامل:

- کلید On-Off : برای روشن و خاموش کردن منبع تغذیه است.
- نشانگر LED سبز رنگ: با روشن شدن منبع تغذیه روشن می شود.
- دریچه بالای کلید On-Off: محل انتخاب ولتاژ 110V و یا 220V ورودی منبع تغذیه است.
- U Connector: برای ارتباط دادن منبع تغذیه با ماژول CPU است و در زیر درب پائین ماژول قرار دارد.

CPU) ماژول CPU

در آزمایشگاه سه نوع ماژول CPU وجود دارد که شامل CPU315-2DP، CPU314IFM و CPU314IFM هستند. در اینجا CPU315-2DP که Order Number آنOrder Number است معرفی میشود.

اجزای روی صفحه ماژول CPU:

كليد انتخاب حالت CPU

این کلید چهار حالت دارد:

- RUN-P: در این حالت می توان برنامه را از PG به حافظه CPU منتقل کرد و CPU می تواند برنامه را اجرای کند.
 - RUN: در این حالت فقط CPU می تواند برنامه را اجرای کند.
 - STOP: در این حالت اجرای برنامه متوقف می شود و می توان برنامه را از PG به حافظه CPU منتقل کرد.
 - MRESET: در این حالت وضعیت کلید ناپایدار است و از این حالت می توان برای پاک کردن برنامه و یا برای Cold restart کردن برنامه که شرح آن در زیر آمده است استفاده می شود.

پاک کردن برنامه: برای پاک کردن برنامه، ابتدا دو ثانیه کلید در وضعیت MRESET نگه داشته میشود (در این شرایط LED زرد رنگ حالت توقف دو مرتبه با فرکانس نیم هرتز روشن و خاموش میشود). سپس کلید را رها و بلافاصله به این حالت برگردانده و نگهداشته میشود تا برنامه پاک شود (در این شرایط LED زرد رنگ حالت توقف با فرکانس دو هرتز روشن و خاموش میشود و با پاک شدن برنامه LED پیوشته روشن باقی میماند).

انواع شروع مجدد^۱برنامه و نحوه انجام آنهادر CPUهای مختلف با استفاده از کلید انتخاب حالت

Cold restart: در این نوع شروع مجدد، پردازش برنامه کاربر از نقطه ابتدایی اولین دستور (OB۱) شروع می شود و همه داده ها اعم از محتوای حافظههای نگهدار نده^۲ و غیرنگهدارنده^۳ پاک می شوند .برای انجام این نوع شروع مجدد لازم است مراحل زیر دنبال شوند:

۱) کلید به وضعیت STOP برده شده تا نشانگر STOP روشن شود. ۲) کلید را از وضعیت Stop به وضعیت MRES برده در این وضعیت نگه داشته، تا نشانگر STOP دو مرتبه، با فرکانس نیم هرتز خاموش و روشن شود و سپس روشن باقی بماند.

۳) در انتهای اجرای بند (۲) کلید به وضعیت RUN یا RUN-P برده تا شروع مجدد آغاز شود. از CPUهای سری 300 فقط CPU318 قابلیت Cold Restart را دارد.

Warm restart: در این نوع شروع مجدد نیز مانند Cold اجرای برنامه کاربر از ابتدا آغاز می شود با این تفاوت که این بار داده های حافظههای نگهدارنده و بلوکهای داده حفظ می شوند (در 300-57 تنها بلوکهای دادهای که خودنگهدار هستند حفظ می شوند). هر بار که کلید Stop به حالت Run و Run برده می شود عمل Warm restart انجام می شود.

Hot restart: در این نوع شروع مجدد پردازش برنامه کاربر از نقطه ای که متوقف شده بود از سر گرفته می شود و هیچ حافظهای پاک نمیشود.

برای اجرای Warm restart و یا Hot restart باید مراحل زیر دنبال شوند:

- کلید به وضعیت STOP برده شده تا نشانگر STOP روشن شود.
- ۲) کلید از وضعیت Stop به وضعیت RUN یا RUN-P برده شده تا شروع مجدد آغاز شود. در این حالت از اینکه کدام نوع شروع مجدد (Warm یا Hot) انجام شود بستگی به پارامتر تنظیم شده در CPU دارد. CPU های سری S7-300، توانائیHot restart شدن را ندارند و همه CPUهای آزمایشگاه از سری S00-S7 هستند.

نشانگرهای LED روی CPU

برای نشان دادن وضعیت کاری CPU هفت عدد LED روی CPU وجود دارد که عملکرد هر ک به شرح زیر است.

Restart

² Retentive

³ Non Retentive

- STOP: روشن شدن آن به رنگ زرد پیوسته حالت توقف CPU، روشن خاموش شدن آن به رنگ زرد با فرکانس نیم هرتز دوره شروع مجدد حالت Cold Restart اجرای برنامه، و روشن – خاموش شدن آن با فرکانس دو هرتز دوره پاک شدن برنامه را نشان میدهد.
 - RUN: روشن شدن آن به رنگ سبز حالت اجرای برنامه را نشان میدهد.
 - Force: روشن شدن آن به رنگ زرد وجود ورودی و خروجی اجباری در برنامه را نشان میدهد.
 - DC5: روشن شدن آن به رنگ سبز وجود منبعتغذیه پنج ولت را نشان میدهد.
 - Battery Fail . روشن شدن آن به رنگ قرمز خالی بودن باطری پشتیبان را نشان میدهد.
 - System Fail: روشن شدن آن به رنگ قرمز وجود اشکال در برنامه سختافزار و یا نرمافزار را نشان میدهد.
 - Bus Fail: روشن شدن آن به رنگ قرمز وجود اشکال در باس شبکه را نشان میدهد.

- Memory Slot: محل نصب كارت حافظه است.

– **باطری پشتیبان:** در زیر درب بخش پائینی ماژول یک باطری پشتیبان وجود دارد که برای نگهداری محتوای حافظههای برنامه و داده است. اگر لازم شود باطری CPU در شرایطی که CPU دارای برنامه است تعویض گردد، برای حفظ شدن برنامه باید در شرایطی که منبع ت**غ**ذیه CPU روشن است، این تعویض انجام شود.

- **MPI Connector:** یک کانکتور ۹ پین است و در زیر درب بخش پائینی ماژول قرار دارد. از این کانکتور برای ارتباط PG با PLC استفاده میشود. این کانکتور در تمام ماژولهای CPU وجود دارد.

Profibus Connector: یک کانکتور ۹ پین است و در زیر درب بخش پائینی ماژول قرار دارد. از این کانکتور برای ارتباط شبکه Profibus-DP استفاده می شود. این کانکتور در تمام ماژول های CPU وجود ندارد. در آزمایشگاه ماژول-های CPU315-2DP دارای این کانکنور است.

(Order Number:321-1BH02-0AA0) SM321) ماژول ورودی دیجیتال^۱

این ماژول دارای ۱۶ بیت ورودی ایزوله شده (در گروه ۱۶ تائی) ۲۴ ولت است. در برنامهها، برای دسترسی به اطلاعات ماژولهای ورودی دیجیتال، بصورت زیر آدرس دهی میشوند. که این آدرس دهیها بصورت Bit، یا Byte (۸ بیت)، و یا Word (۱۶ بیت) میباشند.

Im.n	آدرس دهی ورودی بصورت Bit (یک بیت)
IBm	آدرس دهی ورودی بصورت Byte (مجموعًا ۸ بیت)
IWm	آدرس دهی ورودی بصورت Word (مجموعًا ۱۶ بیت)

در این آدرس دهیها I ، IB و IW فرمت آدرس دهی، m شماره بایت و n شماره بیت (7-0) آدرس مورد نظرهستند. آدرس IBm در این آدرس دهیها Im.7 تا Im.7 تا Im.7 بیت (مجموعا ۱۶ بیت) میباشند. شماره بایت شامل Im.0 تا Im.7 (مجموعا 8 بیت) و آدرس IWm شامل IM.1 تا Im.7 بیت (مجموعا ۱۶ بیت) میباشند. شماره بایت آدرسها در برنامه پیکربندی سختافزار تعین میشود. که نحوه پیکربندی سخت افزار بعدا خواهد آمد.

(Order Number:322-1BH01-0AA0) SM322 (د ديجيتال ۲ -۴-۴) ماژول خروجی ديجيتال

این ماژول دارای ۱۶ بیت خروجی ۲۴ ولت ۵/۰ آمپر است. در برنامه برای دسترسی به اطلاعات این ماژول خروجی بصورت Bit (۸ بیت)، و یا Word (۱۶ بیت) قابل آدرس دهی است. این آدرس دهی به شکل زیر میباشد.

آدرس دهی خروجی بصورت Bit

آدرس دهی خروجی بصورت Byte

¹ Digital Input Modole

² Digital Input Modole

آدرس دهی خروجی بصورت Word

در این نوع آدرسدهیها Q، QB و QW فرمت آدرس دهی، حرف m شماره بایت و حرف n شماره بیت (7-0) آدرس مورد نظر هستند.آدرس QBm شامل Qm.0 تا Qm.7 بیت و آدرس QWm شامل Qm+1.0 تا Qm.7 بیت میباشند. شماره بایت آدرسها در برنامه پیکربندی سختافزار تعین میشود. که نحوه پیکربندی سخت افزار بعدا خواهد آمد.

توضیح: اگر ماژول ورودی یا خروجی دارای ۳۲ بیت باشد، علاوه بر آدرس دهی با فرمتهای آمده در بالا، میتوان بصورت Double Word (۳۲ بیتی) که در زیر آمده است، آدرسدهی کرد.

آدرس دهی ورودی بصورت Double Word

آدرس دهی خروجی بصورت Double Word

در این آدرس دهی ID فرمت آدرس دهی ورودی، QD فرمت آدرس دهی خروجی، حرف m شماره با ارزش ترین بایت دابلورد است. آدرس IDm شامل IDm تا Im.7 بیت و QDm شامل Qm+3.0 تا Qm.7 بیت است. شماره بایت آدرسها در برنامه پیکربندی سختافزار تعین میشود. که نحوه پیکربندی سخت افزار آینده توضیح داده خواهد شد. **توضیح:** در آزمایشگاه ماژول با ۸ بیت ورودی و ۸ بیت خروجی دیجیتال هم موجود است که شماره آن SM323 می باشد.

ا-۴-4) ماژول ورودی و خروجی آنالوگ SM324(Order Number:332-0CE01-0AA0)

این ماژول دارای چهار ورودی و دو خروجی آنالوگ با دقت تبدیل ۸ بیتی است. گستره سیگنال هریک از چهار ورودی این ماژول، 0 تا 20mA و یا 0 تا 10V است. همچنین گستره مقادیر هریک از دو خروجی این ماژول 0 تا 20mA و یا 0 تا 10V است. پارامترهای ورودیها و خروجیهای این ماژول قابل تنظیم نبوده وگستره سیگنالهای آنها بصورت ثابت و برابر مقادیر ذکر شده در بالا میباشند.

در برنامهها برای دسترسی به اطلاعات این ورودیها و خروجیها به شکل زیر قابل آدرسدهی مستقیم می باشند.

آدرس دهی ورودی بصورت Byte

آدرس دهی خروجی بصورت Byte

در این آدرس دهی، PIB و PQB به ترتیب فرمت آدرس دهی ورودی و خروجی آنالوگ بصورت بایت و حرف m شماره بایت آدرس مورد نظر است. شماره بایتها در برنامه پیکربندی سخت افزار که بعدا خواهد آمد تعین میشوند. **توضیح:** اگر دقت تبدیل در ماژول های ورودی ویا خروجی بیشتر از ۸ بیت باشند، متناسب با تعداد بیتها، آدرس دهی میشوند.

که می توانند بصورت Word یا Double Word آدرس دهی شوند.

۵-۱) آدرسدهی حافظه

حافظه را میتوان مانند ماژولهای ورودی و خروجی با فرمت زیر آدرس دهـی کـرد. در ایـن آدرس دهـیهـا M فرمـت آدرس دهی برای حافظه است.

آدرس دهی حافظه بصورت Mm.n Bit

آدرس دهی حافظه بصورت Byte

آدرس دهی حافظه بصورت MWm Word

آدرس دهی حافظه بصورت Double Word

تمرین: مدارک مربوط به ماژولهای میز کار خودتان را (با توجه شمارههای مربوطه) از منابع زیمنس مطالعه کرده و گزارش جامعی برای آنها تهیه کرده و به آزمایشگاه ارائه دهید.

جلسه دوم

هدف :

- کار با پرگرامر ارای برنامه ریزی PLC با نرم افزار Simatic Manager
 - ایجاد پروژه
 - نحوه پیکربندی سخت افزار^۲
- طراحی برنامه در بلوک OB1"، معرفی زبانهای برنامه نویسی، نحوه اجرا، پایش و اصلاح برنامه
 - استفاده از symbol بجای آدرس برای متغیرها
 - نحوه save as کردن یک پروژه
 - نحوه Stop و یا Restart کردن اجرای برنامه PLC از طریق نرم افزار
 - نحوه پاک کردن بر نامه موجود در PLC، از طریق نرم افزار
 - نحوه پاک کردن یک پروژه در نرم افزار Simatic Manager
 - نحوه Upload کردن برنامه از PLC
 - نحوه تنظیم اسم رمز برای پروژه جهت حفاظت از آن

مقدمه

کار با پرگرامر برای کارهای نرم افزاری

از Programmer (که از یک PC بهمراه نرمافزار Simatic Manager و یک PC-Adapter تشکیل شده است) برای تولید و انتقال برنامه به PLC، پایش اجرای برنامه، راهاندازی و متوقف کردن اجرای برنامه در PLC و پاک کردن برنامـه منتقـل شـده بـه PLC، استفاده میشود

با پرگرامر میتوان، به ترتیب روندنمای شکل (۲-۱)، تمام عملیات نرمافزاری مورد نیاز برای آماده سازی یک PLC، به منظور اجرای یک برنامه برای یک سیستم خودکار را انجام داد. در این روند میتوان، توالی پیکربندی سخت افزار و طراحی برنامه را جابجا اجرا کرد. اما روند نمایش داده شده مناسب تر است.



شکل (۲-۱) روند ایجاد برنامه برای PLC

¹ Programmer

Hard Ware Configuration

[°] Organization Block

۲-۱) ایجاد پروژه

نرم افزار Simatic Manager برای برنامه ریزی PLC های سری 300-S7 و یا S7400 شرکت زیمنس بکار می ود. این نرم افزار بر روی PCها آزمایشگاه نصب و از آنها برای برنامه ریزی PLCهای موجود در آزمایشگاه استفاده می شود. برای برنامه ریزی PLC در ابتدا لازم است یک پروژه ایجاد شود. در روند ایجاد یک پروژه کارهائی مانند انتخاب یک CPU برای پروژه، تعین آدرس برای ارتباط پروگرامر با PLC (MPI)، انتخاب بلوک یا بلوکهای سازمانی(^{TB}ها)، مورد نیاز در برنامه، انتخاب یک زبان برای برنامه نویسی (TSA، LAD^{*} ،STL) و تعین نام برای پروژه انجام می شود. برای انجام این کارها، بر طبق روند زیر عمل می شود.

الف) با باز کردن نرم افزار توسط آیکن آیکن فضحه Simatic Manager به همراه صفحه S7 Wizard 'New project بطوری که در شکل(۲-۲) نشان داده شده است باز می شود. در صورتی که پس از باز شدن نرم افزار Simatic Manager ، صفحه S7 Wizard 'New project باز نشود، می توان این صفحه را از منوی File با انتخاب گزینه New Project Wizard باز کرد.



شکل (۲-۲)

برای ایجاد پروژه جدید در صفحه S7 Wizard 'New project دکمه Next انتخاب می شود. با این انتخاب شکل (۲–۳) باز می شود.

STEP 7 Wizard: "New Pro	ject"		×
Which CPU are you us	ing in your projec	t?	2(4)
CPU: <u>C</u> PU name: MPI <u>a</u> ddress:	CPU Type CPU313C-2PIP CPU314 CPU314C-2DP CPU314C-2PIP CPU315 CPU3152DP CPU314(1) 2 2 24 KB w	Order No Order No 6ES7 313-6BE 00-0AB0 6ES7 314-6E 00-0AB0 6ES7 314-6E F00-0AB0 6ES7 314-6BF 00-0AB0 6ES7 314-6BF 00-0AB0 6ES7 315-1AF 03-0AB0 6ES7 315-2AG10-0AB0 6ES7 315-2AG10-0AB0 6ES7 315-2AG10-0AB0 6ES7 315-2AG10-0AB0	
< Back Next >	Finish	ation up to 32 modules, firmware	

شکل (۳-۳)

¹ Multi Point Interface

² Organization Block

³ Statement List

⁴ Ladder

⁵Function Block Diagram

ب) در صفحه شکل (۲-۳) نوع CPU با توجه به Order No ثبت شده روی ماژول (در این مثال برای ماژول -S7314-1AE04 می در این مثال عدد 2) انتخاب می شود،
 (OAB0) و تعین شماره آدرس باس MPI (برای ارتباط PG با CPU انتخاب شده بر روی باس در این مثال عدد 2) انتخاب می شود،
 پس از این انتخاب ها دکمه مفحه شکل (۲-۳) بسته و صفحه شکل (۲-۴) بسته و صفحه شکل (۲-۴) باز می شود.

توضیح۱: معمولا در ایجاد یک پروژه نیازی به تغییر آدرس MPI آمده بصورت پیشفرض نمیباشد، ولی اگر لازم باشـد CPU در یک شبکه MPI قرار بگیرد، در آن شرایط لازم است برای هر CPU یک آدرس مجزای اختصاص یابد. **توضح ۲:** در این صفحه، با انتخاب هر CPU خلاصهای از قابلیتهای آن CPU در محل اختصاص یافته برای ایـن منظـور آشـکار میشود. در شکل خلاصهای از قابلیتهای CPU314 به نمایش در آمده است.

STEP 7 Wizard: "New Pro	oject"		
🕀 Which blocks do you	want to add?		3(4)
Bloc <u>k</u> s:	Block Name OB1 OB1 OB10 OB11 OB12 OB13	Symbolic Name Cycle Execution Time of Day Interrup Time of Day Interrup Time of Day Interrup	t0 t1 t2 t3
	Select <u>A</u> II Language for Sele	cted Blocks	Help on <u>D</u> B
Create with source files			Previe <u>w</u> >>
< Back Next >	Fjnish	Cancel	Help
	(4-1)	شکا	

ج) در صفحه شکل (۲-۴) بلوکهای سازمانی (Organization Block های) مورد نیاز در برنامه (در این مثال OB1) ، زبان برنامه نویسی مورد علاقه (در این مثال LAD) انتخاب و سپس Next فعال میشود. با فعال شدن Next صفحه شکل (۲-۴) بسته و صفحه شکل(۲-۵) باز میشود.

توضیح: بلوکها بخشهائی از برنامه کاربر هستند. بلوکهای OB که بلوکهای خاص میباشند ارتباط بین Oprating system ((از S7CPU) و برنامه کاربر را شکل میدهد.

STEP 7 Wizard: "New Pro)ject"	×
🟐 What do you want to	call your project?	4(4)
Project name: Existing projects:	test_1 Only oneswitch Check your new project in the preview. Click "Make" to create the project with the displayed structure. Preview:	× IN
< <u>B</u> ack <u>N</u> ext>	Finish Cancel Help	

شکل (۲–۵)

د) در صفحه شکل (۲-۵) نام مورد علاقه برای پروژه (در این مثال 1_test) انتخاب سپس دکمه Finish فعال می شود. در اینجا کارهای ایجاد پروژه پایان می یابد. در صورت نیاز به تغییر هر یک از گزینه های انتخاب شده در چهار مرحله فوق، در بخش های دیگر نرم افزار Simatic Manager می توان این تغییر را انجام داد.

Simatic 300 Station در پوشه (Hard Ware Configuration) در پوشه (۲-۲

ییکربندی و Download کردن آن به PLC است.

پس از ایجاد پروژه، لازم است سخت افزارهای نصب شده بر روی ریل PLC به صورت نرمافزاری توسط PG پیکربندی شده و نتیجه پس از کامپایل شدن به PLC منتقل شود. مراحل پیکربند سختافزار، شامل باز کردن برنامه پیکربندی، انتخاب ماژولها و قرار دادن آنها در جای مربوطه، توجه کردن به آدرسهای پیش فرض ماژولها (ودر صورت لزوم تنظیم آدرسهای جدید برای ماژولها)، Save And Compile کردن برنامه

برای پیکربندی سخت افزار پروژه، ابتدا از مسیر منوی File با انتخاب گزینه Open پروژه ایجاد شده (test_1) باز و سپس کارهای آمده در مراحل زیر انجام میشود. با باز شدن پروژه ایجاد شده، صفحه شکل(۲-۶) ظاهر میشود.



الف) برای باز کردن نرمافزار پیکربندی سخت افزار، در پنجره مرور گر(پنجره سمت چپ شکل(۲-۶))، پوشه Simatic 300 الف) برای باز کردن نرمافزار به برنامه پیکر Hardware انتخاب ، سپس در پنجره سمت راست با فعال کردن آیکون Hardware ، صفحه شکل(۲-۷) که مربوط به برنامه پیکر بندی سخت افزار است باز می شود.

R HW Config - SIMATIC 300 Station					
Station Edit Insert PLC View Options Window Help					
D 🗃 🖫 🖷 🐴 🏛 🛍 🏛 🗊 🗖	11				
SIMATIC 300 Station (Configuration) Test_1					Profile Standard
CPU314(1) 3 4 5 6 7 7 - -					
(0) UR	Firmware	MPI address	laddress	Qaddr	General 300 SIMATIC 400 General Science 200 400
					E SIMATIC PC Based Control 300/400
2 CPU314(1) 6ES7 314-1AE04-0AB0		2			
					PROFIBUS-DP slaves for SIMATIC S7,
Press F1 to get Help.					

شکل(۲-۷)

ب) در صفحه باز شده، از پنجره سمت راست از پوشه 300 Simatic، ماژولهائی که باید پیکربندی شوند انتخاب و به پنجره (0) (بالا- سمت چپ) این صفحه، انتقال داده و در ردیف مخصوص به آن ماژول(بر طبق ماژولهای قرار گرفته بر روی ریل)
 Order قرار داده میشوند. برای این کار ماژولهای سختافزاری که از چپ به راست در روی ریل PLC نصب شده اند، بر طبق Order قرار داده میشوند. برای این کار ماژولهای سختافزاری که از چپ به راست در روی ریل ML نصب شده اند، بر طبق Number (۷-۷)
 Order هر یک از این ماژولها از پوشههای مربوطه انتخاب و از بالا به پائین در ردیفهای پنجره بالا – سمت چپ شکل(۲-۷)
 وزار داده میشوند. با قرار گرفتن هر ماژول در سطر خاص خود، آدرسهای پیش فرض آنها، در سطر متناظر، واقع در پنجره پائین- سمت چپ صفحه آشکار میشوند. برای طراحی برنامه اجرائی لازم است این آدرسها بخاطر سپرده شوند.
 توضیح: در این جاگذاری ردیف اول مخصوص ماژول های سیگنال (SM)، ارتباطات (CP) و خاص (PL) ، ردیف محصوص ماژولهای و محصوص ماژولهای سوم مخصوص ماژولهای سوم مخصوص ماژولهای و معود، آدرسهای پیش فرض آنها، در سطر متناظر، واقع در پنجره پائین- توضیح: در این جاگذاری ردیف اول مخصوص ماژول های سوم مخاص خود، آدرسهای پیش و محصوص ماژول منیع در محصوص ماژول منیع در محصوص ماژولهای و محصوص ماژول منیع در بند و ماژولهای سیگنال (SM)، ارتباطات (CP) و خاص (PL) ، محصوص ماژولهای واسط (IM) و ردیف چهارم به بعد برای ماژولهای سیگنال (SM)، ارتباطات (CP) و خاص (PL) هستند.

ج) پس از اتمام جاگذاری ماژولها، با انتخاب گزینه Save & Compile از منوی Station پیکربندی کمپایل و ذخیره میشود. سپس برنامه کومپایل شده به PLC منتقل (Download) میشود. برای انتقال برنامه پیکربندی از ابزار آن 🛍 در روی میله ابزار و یا از منوی PLC گزینه Download استفاده میشود. با این کار، صفحه Select Target Modules نشان داده شده در شکل(۲–۸) باز می شود. در صورت وجود چند CPU در پروژه، در این صفحه CPU مورد نظر انتخاب و سپس دکمه OK انتخاب می شود.

s Slot
s Slot
2
ancel Help

شکل(۲–۸)

با انتخاب دکمه OK صفحه Select node address نشان داده شده در شکل(۲-۹) باز می شود در این صفحه با انتخاب دکمه View(یا Update) تمام CPU های موجود در پروژه به همراه آدرسهای MPI مربوطه در پنجره Addressable Nodes آشکار می-شوند (در این مثال یک CPU در اسلات ۲ از ریل صفر موجود است). با انتخاب CPU مورد نظر و فعال کردن دکمه OK عمل انتقال شروع می شود.

Select node a	ddress			\mathbf{X}			
Over which station address is the programming device connected to the module CPU314(1)?							
Rack:							
Slot:	2 *						
Target Station:	C Local						
	🔿 Can b	e reached by me	eans of gatew	ay			
Enter connect	ion to target sta	tion:					
MPI address	Module type	Station name	CPU name	Plant designation			
2	CPU 314						
Accessible Nod	es						
2	CPU 314						
Update							
ОК	1			Cancel Help			

شکل(۲-۹)

با اجرای سه مرحله فوق کار پیکر بندی پایان مییابد.

توضیح: اگر لازم باشد MPI آدرس CPU تغییرداده شود، برای این کار در صفحه پیکر بندی سخت افزار پس از راست کلیک کردن بر روی ردیف CPU از منوئی که باز خواهد شد گزینه Object Properties انتخاب می شود. با این انتخاب مجموعه صفحه-های مربوط به CPU Object Properties باز می شود. از این مجموعه در صفحه با سر برگ General دکمه Properties انتخاب شده با این انتخاب صفحه نشان داده شده در شکل(۲–۱۰) باز می شود. در این صفحه، آدرس MPI جدید انتخاب شده سپس با انتخاب دکمههای OK دو صفحه باز شده بطور متوالی، کار تغییر آدرس MPI پایان می پذیرد. در پایان لازم است این تغییرات با آدرس MPI قبلی به CPU منتقل (Down Load) شود.

Properties - CPU 314 - (RO/S2	Properties - MPI interface CPU 314 (R0/S2)
Time-of-Day Interrupts 0 General Startup 0 Short Description: CPU 314 24 KB wc configura	General Parameters Address: Highest address: 31 Transmission rate: 187.5 5 Subnet:
Order No.: 6ES7 314 Name: CPU314(Interface	not networked MPI(1) 187.5 Kbps -
Address: 2 Networked: No	
	Can Comment
ок	Cancel Help
	شکا (۱۰–۲)

OB1) طراحی برنامه اجرائی در بلوک OB1

برای باز کردن محیط مربوط به طراحی برنامه اجرائی، در صفحه Simatic Manager از پنجره مرورگر پوشه Blocks انتخاب میشود. با این انتخاب بلوکهای موجود در پروژه در پنجره سمت راست مانند شکل(۲–۱۱) (در این مثالOB1) ظاهر میشوند.



شکل(۲–۱۱)

با با باز کردن بلوک OB1 از پنجره سمت راست صفحه شکل (۲–۱۲) که مربوط به محیط برنامه نویسی با زبان Ladder است باز میشود. توضیح اینکه این زبان برنامه نویسی در یک از مراحل ایجاد برنامه انتخاب شده است. درصورت نیاز در اینجا میتوان ایس زبان را از منوی View تغییر داد.



شکل(۲–۱۲)

۲-۳-۱)معرفی زبانهای برنامه نویسی

عموما برای نوشتن برنامه برای PLC، سه گونه زبان به نامهای STL ، LAD و FBD و FBD وجود دارد. برای برنامه نویسی به زبان LADD از سمبلهای مدارهای الکتریکی و مدارهای منطقی، به زبان STL از حروف و علائم نوشتاری، به زبان FBD از شکلهای بلوکی و سمبلهای مدارهای منطقی استفاده می شود.

در نرمافزارهائی که مجهز به این زبانها هستند، قابلیت تبدیل هر یک از اینزبانها به یکدیگر وجود دارد. و به راحتی میتوان برنامه نوشته شده به هریک از این زبانها را به دیگر زبان تبدیل کرد. در شکل(۲–۱۳) مسیر انتخاب گزینههای مربوط به زبان برنامه نویسی در نرم افزار Simatic Manager نشان داده شده است.

test_	88_1	\SIMATI	C 300 S	tation	\CPU	314(1)]		
Debug	View	Options	Window	Help					
	✓ Ov✓ Del	erviews tails	Ctri	l+C		!« »	!		₩-0
<u> </u>	PLC	Register		_		Con	itents Of	: 'Envii	ronme
		>	Ctr	+1	[Name		
	(• sti	-	Ctr	I+2)	- I	=	TEMP		
	FBI	>	Ctr	H3 /	- I				
	_	<u> </u>							

سکا (۲–۱۲)	â	شكا	۲), ا	۳-	()
------------	---	-----	-------	----	----

در زیر مثالی از AND کردن دو بیت ورودی دیجیتال PLC و انتقال نتیجه به یک بیت خروجی آن به سه زبان نمایش داده شده است.

نمایش به زبان LADD



0 1 0 نمایش به زبان FBD



نمایش به زبان STL

OB1 : "M Network 1	ain : AND	
А	I	0.
A	I	0.
-	Q	Ο.

PG انحوه پایش^۱ اجرای برنامه بوسیله)

از PG می توان برای پایش اجرای برنامه استفاده کرد، که این پایش برای رفع اشکال برنامه منتقل شده به PLC و یا عیبیابی تجهیزات سخت افزاری (بکار گرفته شده در سیستم خودکار تحت کنترل PLC) کابرد دارد. برای پایش کردن اجرای برنامه لازم است ابزار آن (است) که برروی میله ابزار قرار دارد فعال شود. با فعال شدن این ابزار به راحتی می توان نحوه اجرای برنامه را مورد مطالعه قرار داد.

۲-۳-۳) المانهای زبان برنامه نویسی

در آزمایشکاه همه برنامهها فقط به زبان Ladder نوشته و تست خواهد شد. اگر زبان برنامه نویسی LAD انتخاب شده باشد. ابزارهای نشان داده شده در شکل(۲–۱۴) در روی میله ابزار ظاهر می شوند. این ابزارها از ابزارهائی هستند که در این زبان زیاد کاربرد دارند.



این ابزارها از سمت چپ شامل:

- ابزار ایجاد شبکه '
 کنتاکت باز (این المان در صورت یک بودن محل آدرس بیتی نسبت داده شده به آن، فعال میشود)،
- ٥ كنتاكت بسته (اين المان در صورت صفر بودن محل آدرس بيتي نسبت داده شده به آن، فعال مي شود)،
 - کوئیل (برای انتقال نتیجه عملیات بیتی به یک متغیر)،
 - منوی دسترسی به همه توابع بلوکی موجود در نرمافزار،
 - ۰ باز کردن یک شاخه،
 - بستن شاخه باز شده
 - ماز کردن دو شاخه بصورت همزمان هستند.

تمام المانهای مربوط به زبان بر نامه نویسی انتخاب شده، علاوه بر روی میله ابزار در جعبه ابزارهای پنجره نشان داده شده در سمت چپ شکل(۲–۱۵) در دسترس می باشند، که این پنجره با انتخاب گزینه Program Element از منوی Insert قابل باز کردن است.



'Monitoring

مثال: مانند طرح سمت راست شکل(۲–۱۶) یک کلید فشاری NO (Start) و یک کلید فشاریNC (Stop) به یک ماژول ورودی PLC و یک کوئیل کنتاکتور موتور سه فاز، به ماژول خروجی همان PLC وصل هستند، با توجه به مدار معادل الکتریکی این برنامه که در وسط شکل(۲–1۶) ارائه شده است. برنامهای برای PLC آماده کنید تا با این کلیدها بتوان موتور سه فاز را راهاندازی و متوقف کرد.



شکل(۲–۱۶)

پاسخ: برنامه مدار فرمان شکل(۲-۱۶)



۴-۲)نحوه استفاده از symbol بجای آدرس ها

اگر بخواهیم در شبکه زیر بجای آدرسهای I0.0 ، I0.1 و Q4.0 به ترتیب سیمبولهای Start ، Stop و Motor جایگزین شوند به ترتیب زیر عمل می شود.



- در صفحه Simatic Manager از پنجره مرورگر، فایل S7 Program انتخاب می شود. با این انتخاب در پنجره سمت راست صفحه Simatic Manager آیکن Symbols مانند آمده در شکل (۲-۱۷) نمایان می شود.

🗃 test_88_1 E:\Siemens\Step7\s7proj\test_88_						
⊡ ≧) test_88_1	🛅 Sources	📷 Blocks	🔄 🔄 Symbols			
🗄 📆 SIMATIC 300 Station		_				
🖮 🔣 CPU314(1)						
S7 Program(1)						
Sources						
🔤 Blocks						
()Y	۔ شکل(۲–					

- با دو مرتبه کلیک (چپ) کردن روی آیکن جدول سیمبولها، صفحه شکل(۲–۱۸) که مربوط به این جدول است باز می شود. در این صفحه می توان سیمبولهای مورد نیاز در برنامه را بهمرا آدرسها و نوع دیتای مربوطه، در ردیفها (مانند مثال آورده شده در زیر) وارد کرد. پس از تنظیم جدول لازم است تغییرات ایجاد شده در جدول Save شود. برای این کار دکمه Save موجود در میله ابزار انتخاب می شود.

-	🚭 Symbol Editor - S7 Program(1) (Symbols)						
	Symbol Table Edit Insert View Options Window Help						
	😅 🔄 🎒 👗 🖻 💼 🕬 🕬 🗛 🛛 🗛 🖌						
ľ	S7 Program(1) (Symbols) test_88_1\SIMATIC 300 Station\						
I		Status	Symbol 🛆	Address	Data type	Comment	
I	1		Cycle Execution	OB 1	OB 1		
I	2		Stop	1 0.0	BOOL		
	3		Start	I 0.1	BOOL		
	4		Motor	Q 4.0	BOOL		
شکل(۲–۱۸)							

– پس از save کردن جدول سیمبولها، در برنامه اجرائی بجای آدرسها مانند شکل(۲–۱۹) سیمبولها ظاهر میشوند. بـا فعـال و غیر فعال کردن ابزار نشان داده شده در سمت چپ شکل(۲–۱۹) ، نمایش تمام آدرسها و یا سیمبولها در برنامه جابجا میشوند.



اگر بخواهیم آدرسها و سیمبلها را مانند شکل(۲-۲۰) با هم مشاهده کنیم از منوی نشان داده شده در شکل(۲-۲۱) که یکی از منوهای بلوک برنامه است گزینه Symbol Information انتخاب می شود



_	88_1\SIMATIC	300 Station\C	PU314(1)]				
	View Options V	Window Help					
	 ✓ Overviews Ctrl+C ✓ Details PLC Register 		<u></u>				
			Contents Of: 'Environment\Interface'				
	• LAD	Ctrl+1	Name				
	STL	Ctrl+2					
Į,	FBD	Ctrl+3					
	Display with		Symbolic Representation Ctrl+Q				
	Zoom In	Ctrl+Num+	Symbol Information Ctrl+Shift+Q				
	Zoom Out	Ctrl+Num-	Symbol Selection Ctrl+7				
	Zoom Factor		✓ Address Identification				
	🗸 Toolbar		Q4.0				
	Breakpoint Bar		J()(
	 Status Bar 		_1				
	Display Column	s F11	I <u>I</u>				
	شکل(۲–۲۱)						

توضیح: در صفحه جدول سیمبولهای از منوی view با گزینههای Sort و Filter می توان جدول سیمبول را سورت و فیلتر کرد. این گزینهها را بررسی کنید. این گزینهها در شکل (۲-۲۲) نشان داده شده است.

🚭 Symbol Editor - S7 Program(3) (Symbols)				
Symbol Table Edit Insert	View Options Window Help			
⊯ ⊟ ⊕ X №	Zoom In Ctrl+Num+ Zoom Out Ctrl+Num- Zoom Factor	Ctrl+Num+ Ctrl+Num-		
S7 Program(3)	Filter Sort			
Status Symt 1 Columps R O M C CC Ctrl⊥K (۲۲−۲)رکش				

۵-۲) نحوه save as کردن یک پروژه

در مواردی نیاز است که از همه برنامههای یک پروژه یک نسخه کپی شود برای این کار باید از گزینه Save as استفاده شود. برای Save as کردن یک پروژه به ترتیب زیر عمل میشود.

ابتدا در صفحه Simatic Manager (شکل ۲–۲۳) از منوی File گزینه Save As انتخاب می شود.

SIMATIC Manager - test_88_1
File Edit Insert PLC View Options Window Help
New Project' Wizard
Open nens\Step7\s7proj\test_88_
Open Version 1 Project Close ation
Multiproject
am(1) S7 Memory Card ces Memory Card File
Save As
(شکل ۲–۲۳)

با این انتخاب صفحه شکل(۲-۲۴) باز میشود

Sav	ve project as				
ſ	User projects		1		
	Name	Storage path	<u>^</u>		
	test_77 test_78	E:\Siemens\Step7\s7proj\test_7; E:\Siemens\Step7\s7proj\test_7;	7 3		
	test_88_1	E:\Siemens\Step7\s7proj\test_8	3_ 🕑		
	Add to multiproje	ect: E Replace ou	rrrent project Type:		
T	est_88_2		Project		
Storage location (path):					
1					
	OK	Cancel	Help		

شکل(۲–۲۴)

- در این صفحه برای پروژهائی که باید Save As شود نام جدید انتخاب (در این مثال 2_Test_88 و در محل مربوطه وارد می-شود و در زیر آن در صورت ضرورت مسیر ذخیره شدن پروژه هم وارد شده، سپس دکمه تائید Ok فعال می شود، بـا ایـن کارهـا پروژهای با نام انتخاب شده، با محتوای تمام برنامههای پروژه اولیه ایجاد می شود.

Stop از طريق نرم افزار Restart و يا Stop كردن PLC از طريق نرم افزار Simatic Manager

در صورتیکه کلید انتخاب حالت روی ماژول PLC بر روی حالت RUN و یا RUNP باشد، می توان از طریـق نـرم افـزاز Simatic Manager اجرای برنامه PLC را از حالت RUN به حالت STOP و بلعکس تغییر داد. برای متوقف کردن و یـا راه انـدازی کردن مجدد اجرای برنامه PLC از طریق نرمافزار، بر طبق روند زیر عمل می شود:

در شرایطی که برنامه یکی از بلوکهای پروژه باز است از منوی PLC مانند نشان داده شده در شکل(۲–۲۵) گزینه Operating Mode انتخاب می شود.



با این انتخاب صفحه شکل(۲-۲۶) در یکی از دو حالت نشان داده شده باز می شود. اگر PLC در حالت اجرای برنامـه باشد نمایش سمت چپ و اگر PLC در حالت Stop باشد نمایش سمت راست باز می شود.

Operati	ng Mode				Operatin	ig Mode				×
Path:	Test_0\SIMATIC 300 :	Station\CPU314(1)\S	7 Program(1)		Path:	Test_0\SIMATIC	300 Station\CPU314(1)\S7 Program(1)		-
Current (Operating Mode:	RUN	Warm	Restart	Current O	perating Mode:	STOP	[Warm Restart	
			Cold I	Restart					Cold Restart	
			Hot F	Restart					Hot Restart	
			\$1	rop					STOP	
Current k	Keyswitch Setting:	RUN-P			Current K	eyswitch Setting:	RUN-P			
Last Ope	erating Mode:	STARTUP			Last Ope	rating Mode:	RUN			
Upd	ate		Close	Help	Upda	ite		Close	Help	

شکل(۲–۲۶)

- با فعال کردن دکمه Stop ، برنامه PLC از حالت اجرا به حالت توقف، و با فعال کردن یکی از دکمههای Restart اجـرای برنامـه PLC راهاندازی می شود.

در راهاندازی Warm Restart برنامه PLC از اولین شبکه شروع به اجرا شده بطوریکه همه داده های موجود در حافظ ههای غیر نگهدارنده⁽ (داده های مربوط به نتایج اجرای برنامه در قبل از حالت Stop) پاک می شوند.

در راهاندازی Cold Restart برنامه PLC از اولین شبکه شروع به اجرا شده بطوریکه همه داده های موجود در حافظ ههای غیر نگهدارنده و حافظههای نگهدارنده^۲ (مربوط به نتایج اجرای برنامه قبل از حالت Stop) پاک می شوند.

توضیح: اگر در پروژه OB100 ایجاد شده باشد، با انتخاب Warm Restart ابتدا این OB و سپس OB1 اجرا می شوند. و همچنین اگر OB101 ایجاد شده باشد، با انتخاب Cold Restart ابتدا این OB و سپس OB1 اجرا می شوند. در آینده با نحوه بکار گیری این نوع OBها آشنا خواهیم شد. در PLC های سری 300 فقط CPU318 قابلیت اجرای Cold Restart را دارد.

۲-۷) نحوه پاک کردن بر نامه موجود در PLC، از طریق نرم افزار Simatic Manager

در شرایطی که یکی از بلوکهای لاجیک پروژه باز و PLC در حالت Stop و یا Run P است. میتوان برنامههای موجود در حافظه PLC را از طریق نرم افزار Simatic Manager پاک کرد. برای این کار:

- در شرایطی که یکی از بلوک های برنامه باز است از منوی PLC نشان داده شده در شکل(۲–۲۷) گزینه Clear\Reset انتخاب می شود.



- با این انتخاب، صفحه شکل(۲-۲۸) باز مِشود. در این صفحه با انتخاب دکمه Yes تمام برنامهها و داده های موجود در حافظه PLC یاک می شوند.

) Non Retentive

[°] Retentive



شکل(۲–۲۸)

لازم است توجه شود، با این کار، برنامههای همه بلوکهای موجود در PLC پاک می شوند. توضیح: درموقع اجرای روند پاک کردن برنامه در PLC، اگر PLC در حالت Stop نباشد در زمان درخواست، از کاربر برای بردن به حالت stop سئوال می شود، در صورت تائید، PLC به حالت Stop رفته سپس برنامه پاک می شود. اگر برنامه PLC با اسم رمز حفاظت شده باشد قابل پاک کردن نمی باشد.

۸−۲) نحوه پاک کردن یک پروژه در نرم افزار Simatic Manager

در مواردی نیاز است پروژهای که توسط کاربر در نرم افزار Simatic Manager ایجاد شده است پاک شود. برای این کار ابتدا باید در صفحه Simatic Manager از منوی File گزینه Delete انتخاب شود. با این انتخاب مجموعه صفحات مربوط به Delet باز می شود.

De	elete		×		
ſ	User projects Li	braries Sample projects Multiprojects	1		
	Name Storage path				
	🎒 S7_0B20	C:\Program Files\Siemens\Step7\s7proj\S7_0B20			
	By S7_Pro1	C:\Program Files\Siemens\Step7\s7proj\S7_Pro1	Ξ		
	By S7_Pro2	C:\Program Files\Siemens\Step7\s7proj\S7_Pro2	-		
	🞒 Test_1	C:\Program Files\Siemens\Step7\s7proj\Test_1			
	B Test_10	C:\Program Files\Siemens\Step7\s7proj\Test_10			
	Test_11	C:\Program Files\Siemens\Step7\s7proj\Test_11			
	Dest_12	C:\Program Files\Siemens\Step7\s7proj\Test_12	~		
	<				
-		Selected			
ι	Jser Projects:	1			
L	.ibraries:				
9	ample Projects:				
Multiprojects:		Browse			
		·			
[OK	Cancel Help			

شکل(۲–۲۹)

در صفحه با سر برگ User Projects که در شکل(۲۹–۲۹) نشان داده شده است، از پنجره لیست پروژهها، پـروژهای کـه بایـد پـاک شود انتخاب و سپس دکمه OK فعال میشود. با این عمل پروژه انتخاب شده پاک میشود.

PLC)نحوه Upload کردن برنامه از ∧-۲

در بعضی موارد نیاز است، پروژهای که در حافظه PLC وجود دارد به پروگرامر منتقل شود. برای این منظور(در صورت عـدم وجـود اسم رمز^۱ برای برنامه موجود در PLC) از گزینـه Upload در منـوی PLC (در صـفحه Simatic Manager) اسـتفاده شـده، و بـه ترتیب آمده در زیر عمل میشود.

الف) ایجاد یک پروژه تهی، برای Upload کردن برنامههای یک پروژه از PLC، ابتدا لازم است یک پروژه تهی(بدون هیچ برنامه) با نام مشخص ایجاد شود. برای این کار در صفحه Simatic Manager از منوی File گزینه New انتخاب می شود. با این انتخاب صفحه سمت چپ شکل(۲-۳۰) باز می شود.

lew Project				
User projects L	ibraries Multiprojects			
Name	Storage path			
11	E:\Siemens\Step7\s7proj\11			
exam_11	E:\Siemens\Step7\s7proj\exam_1			
exam_12 exam_13	E:\Siemens\Step7\s7proj\exam_1 F:\Siemens\Step7\s7proj\exam_1	3		Marca Harland
exam_14	E:\Siemens\Step7\s7proj\exam_1	4	SIMATIC Manager	- New_Optoad
Force_Test	E:\Siemens\Step7\s7proj\Force_	Te	File Edit Insert PLC	View Options Windo
New_Upload	E:\Siemens\Step7\s7proj\New_U	plo		
S7 Pro11	E:\Siemens\Step7\S7Proj\Uniy_d F:\Siemens\Step7\s7proj\S7_Pro	ine 11 😡		
Tor_non	E. colonicità colopit (al project _1 to		New Unload	:\Siemens\Sten7\s7
Add to current				
Mamai		Tuna	New_Upload	- MP1(1)
Name.		туре.		
New_Upload		Project 💌		
C	- 14-2			
storage location (satnj:			
E:\Siemens\Step	7\s7proj	Browse		
OK	Canad			
	Carice	я пер		
	/*	". Y \ 16 '	.	

شکل(۲–۳۰)

در محل نام این صفحه (User Project) اسم پروژه وارد و سپس دکمه OK انتخاب میشود. با ایـن عمـل یـک پـروژه تهـی (از برنامههای پیکربندی و بلوکها) در صفحه Simatic Manager (مانند شکل نشان داده شده در سمت راست شـکل(۸–۱۳)) ایجـاد میشود.

ب) انتقال برنامه از PLC به پروژه تهی از برنامه.

برای انتقال برنامههای موجود در PLC به پروژه تهـی در داخـل پروگرامـر، در صـفحه Simatic Manager از منـوی PLC گزینـه Upload Station انتخاب میشود. با این انتخاب صفحه شکل(۲–۳۱) باز میشود.

Select node a	ddress				×
Which module of	do you want to r	each?			
Rack:	0 .				
Slot	2 +				
Target Station: © Local					
Enter connect	ion to target sta	tion:	cans or yatew	ay 	
MPI address	Module type	Station name	CPU name	Plant designation	
2	CPU841-0				
, Accessible Nod	es				
2	CPU841-0				
Update					
ОК				Cancel	Help

شکل(۲–۳۱)

در این صفحه شماره محل اسلات و راک (ریل) محل قرار گیری CPU ئی که باید از آن، برنامه Upload شود و همچنین آدرس MPI آن تعین میشود. پس از تنظیمهای لازم در این صفحه با دکمه تائید OK، Upload اجرا میشود، و برنامه (بدون همراه داشتن هیچ گونه توضیحات و سیمبولهائی) از PLC به پروگرامر منتقل میشود. تر خصصا کی کی دیک بیستان سیامی میشود.

توضیح: با کمک دکمه View میتوان آدرسهای تعین شده از قبل در باس MPI را مشاهد کرد. که در اینجا فقط یک آدرس موجود است.

٩-٢) تنظيم حفاظت

برای تنظیم حفاظت از برنامه موجود در PLC ، در نرم افزار پیکر بندی سخت افزار، روی CPU راست کلیک کرده سپس گزینه Object Properties انتخاب می شود با این انتخاب مجموعه صفحات مربوط به CPU Object Properties باز می شود. از این مجموعه در صفحه با سر برگ Protection که در شکل(۲-۳۲) نشان داده شده است سه انتخاب برای تنظیم حفاظت وجود دارد

- ۱- به حالت کلید انتخاب حالت بستگی دارد که با انتخاب این کلید:
 در حالت PLP و RUN-P میتوان برنامه PLC را از طریق PG پاک کرد، خواند و یا نوشت
 در حالت RUN فقط می توان برنامه PLC را از طریق PG خواند.
 در حالت اگر RUN فقط می توان برنامه PLC را از طریق PG خواند.
 در این حالت اگر REMOVABLE انتخاب شده باشد، با اسم رمز از طریق PG می توان
 ۲۰ می توان برنامه بود.
 - ۲- به منظور تنظیم اسم رمز برای ممانعت از نوشتن برنامه PLC می توان حفاظت ایجاد کرد.
 - ۳- به منظور تنظیم اسم رمز برای ممانعت از خواندن و نوشتن برنامه PLC می توان حفاظت ایجاد کرد.

Properties - CPU 315-2 DP - (RO/S2	2)		×
General Startup Cycle/C Time-of-Day Interrupts Cyclic Interrupt Level of Protection	Clock Memory Re t Diagnostics/Clock Mode Process Mode Permitted Increase in Scan Cycle Time by T Test Mode	est Functions:	Interrupts Communication
ОК		Cancel	Help

شکل(۲-۳۲)

جلسه سوم

هدف:

- بررسی المان های بیت لاجیک، با ارائه مثال برای هر یک از المان ها شامل:
- o کنتاکتهای NO و NC، المانNot، کوئیلهای خروجی میانی و پایانی،
 - o کوئیل های Set ،Reset و کوئیل Save
 - o فیلپ فلاپ و S-R فیلپ فلاپ و تفاوت عملکرد آنها
 - آشکار کننده لبه بالا رونده و لبه پائین روندهء عملیات یک شبکه
- آشکار کننده لبه بالا رونده و یا لبه پائین رونده یک آدرس با شرط اینکه ورودی دیگر المان مربوطه یک باشد
 - بررسى انتقال داده با المانMove
 - معرفی سیمولاتور PLC:

1-۳) بررسی Bit Logic ها

مجموعه المانها برای اجرای عملیات بیتی در جدول زیر نشان داده شده است. همه این المانها دارای آدرس بیتی هستند. در ادامه پس از معرفی این المانها مثالی برای کار برد هر یک از آنها ارائه شده است.

مجموعه المانهای Bit logic در این فولدر قرار دارند	i⊑
ابزار ایجاد شبکه، با این ابزار میتوان بـه تعـداد مـورد نیـاز	
شبکه در برنامه ایجاد کرد	
کنتاکت NO، در صورتی <i>ک</i> ه محتوای آدرس مربوطه یـک	▋
منطقی باشد ارتباط بین دو پایه آن بر قرار میشود	
کنتاکت NC، در صورتی که محتوای آدرس مربوطه صفر	
منطقی باشد ارتباط بین دو پایه آن بر قرار میشود	
معکوس کننـدہ (Not)، نتیجـه عملیـات مـا قبـل خـود را	
معکوس میکند	
کوئیل خروجی، همواره محتوای آن برابر نتیجـه عملیـات	
شبکه متصل به آن است	
خروجی میانی ، نتیجه عملیات ما قبـل خـودش را در آدرس	
مربوطه ذخیره می کند.	
کوئیل Reset، در صورتیکه نتیجه عملیات شبکه متصـل بـه	(R)
آن یک شود و محتوای ایـن کوئیـل یـک باشـد، محتـوای آن	
صفر شده و با تغییر نتیجه شبکه از یک به صفر این کوئیل در	
وضعیت صفر باقی میماند	
کوئیل Set، در صورتی که نتیجه عملیات شبکه متصل بـه آن	
یک شود و محتوای این کوئیل صفر باشد، محتوای این کوئیل	
یک شده و با تغییر نتیجه شبکه از یک به صفر این کوئیل در	
وضعیت یک باقی میماند	

Rs فلیپ فلاپ، این المان دو ورودی بیتی به نامهای Reset	RS
و Set و یک خروجی بیتی دارد و بر طبق جـدول زیـر عمـل	
مىكند.	
Reset Set Output	
بدون تغيير 0 0	
$ \begin{array}{cccccccccccccccccccccccccccccccccccc$	
1 1 1	
SR فلیپ فلاپ ، این المان دو ورودی بیتی به نامهای Set و	II ⊨ E SR
Reset و یک خروجی بیتی دارد و بر طبق جـدول زیـر عمـل	
مىكند.	
Reset Set Output	
بدون تغيير 0 0	
$ \begin{array}{cccccccccccccccccccccccccccccccccccc$	
1 1 0	
اشکار کننده لبه پائین رونده، در صورتیکه نتیجه عملیات -	
شبکه متصل به ورودی آن از یک به صفر تغییر کند، خروجی	
آن به مدت یک Scan time، یک میشود. این المان به ازاء هر	
محل که در شبکه به کار گرفته شود، به یک بیت حافظه نیاز	
دارد.	
آشکار کننده لبه بالا رونده، در صورتیکه نتیجـه عملیـات	
شبکه متصل به وردی آن از صفر به یک تغییر کند، خروجی	
آن به مدت یک Scan time ، یک می شود. این المان به ازاء	
هر محل که در شبکه به کار گرفته شود، به یک بیت حافظـه	
نياز دارد.	
کوئیل Save، نتیجه عملیات لاجیکی شبکه متصل بـه آن در	
حافظه بیتی BR ذخیر شده و میتوان ایـن نتیجـه بـا شـبکه	
بعدی AND خواهد شد.	
آشکار کننده لبه پائين رونده متغيـر از يـک آدرس، در	NEG
صورتیکه ورودی این المان یک باشد و متغیر آدرس داده شـد	
به این المان از وضعیت یک به صفر تغییر کند خروجی آن به	
مدت یک Scan Time یک میشود. این المان به ازاء هـر جـا	
که در برنامه به کار گرفته شود، به یک بیت حافظه نیاز دارد.	
آشکار کننده لبه بالا رونده یک متغیر از یک آدرس، در	Pos
صورتیکه ورودی این المان یک باشد و متغیر آدرس داده شـد	
به این المان از وضعیت صفر به یک تغییر کند خروجی آن به	
مدت یک Scan Time یک میشود. این المان به ازاء هـر جـا	
که در برنامه به کار گرفته شود، به یک بیت حافظه نیاز دارد.	

مثال ۱:

با استفاده از المانهای Not، NC، NO، کوئیل خروجی و خروجی میانی برنامهای برای توابع زیر طرح کنید.

$F1 = (A \oplus$	B)C					
$F2 = (A \oplus$	$\overline{B}B)D \bullet \overline{E}$					
F1: Q4.0	F2= Q4.1	A: I0.0	B: I0.1	C: I0.2	D: I0.3	E:I0.4

برنامه:





مثال۲:

برای آشنا شدن با نحوه عملکرد کوئیلهای (S) و (R) برنامه زیر را آماده و عملکرد آنرا بررسی کنید. توجه شود، حتما لازم است کوئیلهای (S) و (R) در دو شبکه جداگانه به کار گرفته شوند. برای بررسی این دو کوئیل، بر طبق جدول زیر از بالا به پائین ورودیها را تغییر داده نتایج خروجی را در جدول ۱ وارد کنید. طرح دو شبکه زیر را جابجا کرده و بررسی را مجددا انجام داده و نتایج را در جدول ۲ وارد کنید.

نتایجی را که با این بررسی در ارتباط با عملکرد این کوئیلها بدست میآورید توضیح دهید.

work 1: Title:		I0.	I0.0	Q4.	I0.	I0.	Q4.
		1		0	1	0	0
10.0	04.0	0	0		0	0	
	-(s)	0	1		0	1	
		0	0		0	0	
twork 2: Title:		1	0		1	0	
10.1	04.0 (p)	0	0		0	0	
	-, R)	1	1		1	1	
			جدول ۱			جدول۲	

مثال ۳: برای بررسی المانهای RS و SR فلیپ فلاپ دو شبکه زیر را آماده کرده سپس بر طبق جدولهای زیر از بـالا بـه پـائین ورودیها را تغییر داده نتایج خروجی را در جدولهای مربوطه وارد کنید.

Netw	work 3: Title:	I0.	I0.	Q4.	I0.	I0.	Q4.
	MO.0	1	0	1	1	0	2
	I0.1 RS Q4.1	0	0		0	0	
		0	1		0	1	
		0	0		0	0	
Nets	work 4: Title:	1	0		1	0	
	M0.1	0	0		0	0	
		1	1		1	1	
	10.1-R		عدول ۱	?		<u>م</u> دول۲	•
					1		

سئوال: چرا نتایج عملکرد این دو فلیپ فلاپ در ردیف آخر با هم متفاوت هستند.

تمرین ۱ : در شکل زیر مدارهای الکتریکی قدرت و فرمان یک تابلوی الکتریکی مربوط به یک موتور سه فاز که بصورت چپگرد و راستگرد کار میکند نشان داده شده است. برنامههائی برای PLC به منظور جای گزین کردن مدار فرمان این تابلو در سه طراح مختلف آماده کرده آنرا تست کنید. آدرس ها بصورت سیمبول استفاده شود **الف**: فقط با استفاده از کنتاکتهای NC ،NO و کوئیل خروجی **ب**: فقط با استفاده از کنتاکتهای NC ،NO و کوئیل Sest و کوئیل Reset

ج: فقط با استفاده از کنتاکتهای NC ،NO و فلیپ فلاپ SR



برنامه الف:

OB1 : "Main Program Sweep (Cycle)" Network 1: Title:





سئوال: چرا بجای کنتاکت بسته S0 در برنامه از کنتاکت NO و بجای کنتاکتهای بسته K0 و K1 در برنامـه از کنتاکـتهـای NC استفاده شده است.

Q4.0

برنامه ب:

 OB1 : "Main Program Sweep (Cycle)"

 Network 1: Title:

 IO.0
 IO.1
 IO.2
 Q4.1

"50" "51" "52" "K1" "K0"

Network 2: Title:

10.0	Q4.0
"SO"	"KO"
1.4	(n)
	(R) – J

Network 3: Title:

	10.0	10.2	10.1	04.0	04.1
l	"SO"	"S2"	"S1"	"K0"	"K1"
ł	──┤		<u> </u> ∕	— <u> </u>	—(s)—

Network 4 : Title:

10.0	Q4.1
"S0"	"K1"
<u> </u> //────	(R)

برنامه ج:



مثال۴: برای بررسی آشکار کننده لبه بالا رونده و آشکار کننده لبه پائین رونده دو شبکه زیر را آماده کرده سپس بر طبق جدول-های زیر از بالا به پائین ورودیها را تغییر داده نتایج خروجی را در جدولهای مربوطه وارد کنید.

Network 5): Title:	I0.	I0.	Q4.	I0.	I0.	Q4.
M0.3	1	0	3	1	0	4
I0.0 SR Q4.3	0	0		0	0	
	0	1		0	1	
	0	0		0	0	
<pre>Wetwork 6: Title:</pre>						
M0.5	1	0		1	0	
I0.0 S Q ()	0	0		0	0	
IO.1 MO.4	1	1		1	1	
		مدول ۱	?		<u>م</u> دول۲	

سئوال: آشکار ساز لبه بالا روند و آشکار ساز لبه پائین رونده در دو شبکه بالا چه تاثیری دارند.

مثال۵: با تغییر مقادیر ورودی های آمده در جدول زیر، عملکرد دو شبکه مجاور جدول را بررسی کرده با این بررسی کار کوئیل Save را نتیجه گیری کنید

Network 7: Title:	I0.0	I0.1	I0.2	Q4.5
I0.0	0	0	0	
(SAVE)	1	0	0	
TO 1	0	1	0	
	1	1	0	
	0	0	1	
Network 8: Title:	1	0	1	
I0.2 Q4.5	0	1	1	
	1	1	1	

مثال ۶: عملکرد شبکه زیر را با تغییر مقادیر ورودی ها بر طبق جدول آمده در کنار آن بررسی کرده با این بررسی کار بلوک Pos را نتیجه گیری کنید.

Network 1: Title:		I0.0	I0.1	I0.2	Q4.0
10.1	04.0	0	0	0	
IO.O POS	SR	1	0	0	
	s	1	1	0	
THE REPORT		0	0	0	
M0.0-m_B11	10.2-R	0	0	1	
		0	0	0	
		0	1	0	
		1	1	0	
		0	0	0	
		0	0	1	

۲-۳) بررسی انتقال داده (Move):

با بلوک Move می توان داده ها با طولWord ، Byte ، و Double Word را از محلی به محل دیگر (بدون تغییر فرمت آن) انتقال داد. سمبول این بلوک در زیر ارائه شده است. هرگاه Enable ورودی فعال باشد این بلوک اجرا و Enable خروجی فعال می شود



اگر طول آدرس ورودی Byte و طول آدرس خروجی Double Word باشد، محتوای آدرس ورودی بـه کـم ارزش تـرین بایت مربوط به Double Word خروجی منتقل می شود.

اگر طول آدرس ورودی Byte و طول آدرس خروجی Word باشد، محتوای آدرس ورودی به کم ارزشترین بایت مربـوط به Wordخروجی منتقل می شود.

اگر طول آدرس ورودی Word و طول آدرس خروجی Double Word باشد، محتوای آدرس ورودی به Word کم ارزش-تر مربوط به Double Word خروجی منتقل می شود.

اگر طول آدرس ورودی Double Word و طول آدرس خروجی Byte باشد، محتوای کم ارزش ترین بایت Double Word و رودی به Byte خروجی منتقل می شود.

اگر طول آدرس ورودی Double Word و طول آدرس خروجـی Word باشـد، محتـوای Word کـم ارزشتـر ورودی بـه خروجی منتقل می شود.

Byte اگر طول آدرس ورودی Word و طول آدرس خروجی Byte باشد، محتوای بایت کم ارزش تر Word ورودی به خروجی خروجی منتقل می شود.

اگر طول آدرسهای ورودی وخروجی مثل هم باشند دادههای ورودی به آدرسها با ارزش نظیـر بـه نظیـر بـه خروجـی منتقل می شوند.

چند مثال برای المان MOVE:



در این مثال محتوای IB1 به QB4 منتقل می شود.



در این مثال محتوای IB0 به QB5 منتقل می شود.



در این مثال محتوای IB3 به IB2 ،QB7 به IB1 ،QB6 و IB0 به QB4 منتقل می شوند.

PLC)معرفي سيمولاتور PLC:

با آیکن نشان داده شده در شکل(۳-۱) که در میله ابزار صفحه Simatic Manager موجود است، نـرم افـزار سـیمولاتور PLC نشان داده شده در شکل(۳-۲) باز می شود.



توجه: توجه شود هرگاه سیمولاتور باز شده باشد و PG هم به PLC وصل باشد در این شرایط اگربرنامهای Download شود آن برنامه به PLC باید حتما سیمولاتور بسته شود.

🗑 S7-PLCSIM - SimView1 📃 🗆 🔀					
File Edit View Insert PLC	Execute Tools Window Hel	p			
🗋 D 🚅 🖬 🕹 🖬 🖬 👘	≧⊞-∞ №?				
· 使 使 何 行 它 答 何	1 12 12 12				
\$					
🖪 СРИ 📃 🗖 🔀	🖿 ІВ О 💶 🗖 🗙	🔜 QB 1 🔳 🗖 🗙	🖭 T 0 📃 🗆 🗙		
DP RUN-P	IB 0 Bits 💌	QB 1 Bits 💌	TO		
BUN STOP MRES	76543210	76543210	0 10ms 💌 T=0		
🖿 MB 10 🔳 🗖 🗙	🖭 QB 4 🔳 🗆 🗙	🖭 QB 5 🔳 🗖 🗙	🖼 C 0 💶 🖂		
MB 10 Bits 💌	QB 4 Bits 💌	QB 5 Bits 💌	C 0 Decimal 🖵		
	76543210	76543210	0		
Press F1 to get Help.					

شکل(۳–۲)

از ابزارهای این سیمولاتور می توان برای اجرای برنامه، مشـاهده نتـایج آن در آدرسـهای ورودی؛ خروجـی، حافظـه، زمـانسـنج، شمارنده و مشاهده کرد. در PLC سری 300 شرکت زیمنس فقط دو آکمولاتور وجود دارد.

جلسه چهارم

هدف:

- معرفی فرمت تنظیم زمان سنجش برای زمانسنجها و نحوه تنظیم زمان آن بصورت مقدار دهی ثابت و یا از طریق یک Word آدرس
 - بررسی زمانسنجها بصورت بلوکی با ارائه یک مثال برای هر یک از آنها شامل: Pulse Time،
 - Retentive On Delay Timer، On Delay Timer، Extended Pulse Timer
 - بررسی زمان سنج ها بصورت کوئیل با ارائه یک مثال برای یکی از زمانسنج ها
- بررسی فرمت تنظیم مقدار اولیه برای شمارندهها و بررسی شـمارندهای، صـعودی، نزولـی و صـعودی-نزولی بصورت بلوکی با ارائه یک مثال برای یکی از آنها
 - نحوه تنظیم مقدار اولیه برای شمارندهها از طریق یک Word آدرس
 - بررسی شمارنده ها بصورت کوئیل با ارائه یک مثال برای یکی از آنها

۱-۴ بررسی زمانسنجها (Timers)

مقدمه

یکی از مهمترین المانها در PLC زمانسنجها هستند که برای سنجش زمان در برنامه، از آنها استفاده می شود. انواع زمانسنجهای موجود در نرم افزار Simatic Manager در جدول زیر ارائه شده است. این زمانسنجها شامل pulse Timer در مختلف Retentive On Delay Timer ،On Delay Timer ،Extended Pulse Timer و Off Delay Timer ،هستند که در دو شکل مختلف بلوکی و کوئیلی در برنامه بکار گرفته می شوند. چهار زمان سنج بالائی نشان داده شده در جدول شکل بلوکی و چهار زمانسنج زیرین این جدول شکل کوئیلی این زمانسنجها هستند که در این بخش نحوه بکار گیری هر یک از آنها در برنامه بایک مشال

Pulse Timer Extended Pulse Timer ON Delay Timer Retentive On Delay Timer Off Delay Timer Pulse Timer Extended Pulse Timer On Delay Timer Retentive On Delay Timer Off Delay Timer	
--	--

تعداد زمانسنجهای یک PLC به توانائی CPU آن PLC بستگی دارد. حافظه اختصاص یافته برای هر زمانسنج یک Word است که می توان بصورت یکی از فرمتهای زیر آنها را مقدار دهی کرد.
- فرمت تنظیم مقدار زمان سنجش ٰ بصورت مقدار دهی ثابت

H فرمت تنظیم مقدار زمان سنجش بصورت مقدار دهی ثابت، به شکل S5T#aH_bM_cS_dmS است. در ایـن فرمـت H واحد دقیقه، S واحد ثانیه و ST واحد میلی ثانیه است و c ،b ،a و b ضرائبی هسـتند کـه توسط طـراح برنامـه مشخص میشوند. گستره تنظیم این زمان از 10MS تا 2H_46M_30S است.

مثال برای تنظیم زمان سنجش بصورت مقدار دهی ثابت: چهار ثانیه = S5T#4S یک ساعت و دوازده دقیقه و هیجده ثانیه = S5T#1H_12M_18S

فرمت تنظیم مقدار زمان سنجش از طریق یک Word آدرس

برای تنظیم مقدار زمان سنجش برای یک زمانسنج از طریق نتایج پردازش در برنامه، بصورت آدرس دهی یک Word انجام می شود. لازم است محتوای این آدرس بصورت فرمت wxyz باشد.

در این فرمت w دو بیت و مقدار آن تعین کننده پایه زمانسنجش است. و سه حرف z y x هر یک چهار بیت و بصورت کـد BCD تشکیل یک عدد سه رقمی را میدهند که با حاصلضرب این عدد سه رقمی در مقدار پایه، مقدار زمان سنجش برای یک زمانسنج تعین میشود.

باید توجه شود گستره تنظیم زمان برای زمانسنجها با مقدار پایه زمان سنجش تغییر میکند و لازم است توجه شود محتوای ۱۲ بیت z y x حتماً بصورت کد BCD (کد ۰ تا ۹) باشند در غیر این صورت برنامه اجرا نخواهد شد

فرمت خروجیهای زمان سنجها

همه زمان سنجها دارای دو خروجی بطول Word و یک خروجی بشکل بیت هستند. در دو خروجیای که طول حافظه آنها بصورت Word است مقادیر زمان باقیمانده از زمان سنجی، در دو فرمت هگزادسیمال و BCD در دسترس هستند. در خروجی با فرمت BCD آن، دو بیت متناظر w آمده در بالا تعین کنند واحد هر دو خروجی است. که این واحد دقت قرائت زمان باقی مانده از زمانسنج را مشخص میکند. نحوه عملکرد خروجی بیتی زمانسنجها بستگی به نوع آن زمان سنج دارد که در بحش مربوط بـه هر زمانسنج آمده است.

گستره تنظیم زمان سنجش	پایه زمان و یا دقت قرائت زمان باقی مانده	کد w
تا S_990 MS معادل 9 S_990 MS	10 MS میلی ثانیه 10	00
ا تا 99900 MS معادل 1M_39_900MS	100 MS میلی ثانیه 100	01
16M_398 معادل 999	1 ثانیه 1 S تا S	10
2H_46M_30S معادل 9990 S	10 ثانیه 10 S	11

در جدول زیر گستره تنظیم زمان سنجش برای زمانسنجها بر حسب مقدار زمان پایه نشان داده شده است.

مثال برای تنظیم زمان سنجش بصورت آدرس دهی:

آدرس مورد مثال	کد محتوای آدرس	مقدار زمان سنجش
MW0	0000 1001 0101 0010	9520 MS
IW4	0011 0011 1000 0000	3800 S

۲-۱-۱ زمانسنجهای پالسی

الف: Pulse Timer

این زمانسنج. با لبه بالا رونده ورودی S شروع به زمانسنجی کرده و با این شروع خروجی Q آن یک میشود. بـا بـاقی-ماندن ورودی S در سطح یک، روند زمانسنجی تا زمان تنظیم شده در ورودی TV^۲(زمان سنجش) ادامه یافتـه و پـس از پایـان این زمان، خروجی Q صفر میشود. در خروجی BI این زمانسنج، زمان باقی مانده از دوره زمانسنجی به صورت کـد بـاینری و در خروجی BCD آن، این زمان به صورت کد BCD در دسترس قرار می گیرند. توجـه شـود دوخروجـی BI و BCD از نظـر مقـادیر عددی با هم برابر هستند و دو بیت مربوط به مشخص کننده پایه زمان در خروجی BCD تعین کننده واحد این مقادیر هستند

اگر در دوره زمانسنجی، ورودی S صفر شود روند زمانسنجی متوقف، خروجی Q صفر شده و زمان باقی مانده از دوره زمانسنجی در خروجیها باقی میمانند. در صورتیکه ورودی S مجددا یک شود زمانسنج زمانسنجی مجددا از ابتدا شروع می-شود. با یک شدن ورودی R روند زمانسنجی متوقف شده، خروجی Q و زمان باقیمانده از زمانسنجی در دو خروجی صفر می شوند.

در شکل(۴-۱) دیاگرام زمانی این زمانسنج ارائه شده است. این دیاگرام زمانی را توسط طرح ارائـه شـده در مجـاور آن بررسی کرده و تغییرات خروجی QW4 را تحلیل کنید.



توضیح: استفاده از آدرسهای QW4 و Q4.7 در برنامه بخاطر محدودیت سخت افزار موجود در آزمایشگاه صورت گرفته است و با توجه به اینکه در خروجیهای BCD و BI این نوع زمان سنجها از دو بیت با ارزش آنها استفاده نمی شود، این شکل آدرس دهی مشکلی در دسترسی به پاسخ ایجاد نمی کند.

ب: Extended Pulse Timer

کار این زمانسنج مشابه Pulse Timer است با این تفاوت که با لبه بالا رونده ورودی S شروع به زمانسنجی کرده، و برای ادامه کار نیاز به باقی ماندن این ورودی در حالت یک ندارد.

اگر در دوره زمانسنجی ورودی S مجددا یک شود عمل زمانسجی مجددا از ابتدا شروع میشود. در شکل(۴–۲) دیـاگرام زمانی این زمانسنج ارائه شده است. این دیاگرام زمانی را توسط طرح ارائه شده در مجاور آن بررسی کرده، و تغییـرات خروجـی W44را تحلیل کنید.



شکل(۴–۲)

۲-۱-۴ زمانسنجهای با تاخیر روشن شونده

الف: On Delay Timer

این زمانسنج. با لبه بالا رونده ورودی S شروع به زمانسنجی کرده و با باقی ماندن این ورودی در سطح یک رونـد کار زمانسنجی تا زمان تنظیم شده در ورودی TV ادامه پیدا میکند. پـس از پایـان ایـن زمـان خروجـی Q یـک مـیشـود. در دوره زمانسنجی در خروجی BI زمانسنج، زمان باقی مانده از دوره زمانسنجی به صورت کد باینری و در خروجی BCD آن، این زمـان به صورت کد BCD در دسترس قرار میگیرند.

اگر در دوره زمانسنجی، ورودی S صفر شود روند زمانسنجی متوقف شـده، و زمـان بـاقی بانـده از دوره زمانسـنجی در خروجیها باقی میمانند. در صورتیکه ورودی S مجددا یک شود عمل زمانسنجی مجددا از ابتدا شروع میشود.

هر گاه ورودی R قبل از یک شدن خروجی Q یک شود کار زمانسنجی متوقف شده، و زمان باقیمانده از زمانسنجی هـم در دو خروجی صفر می شوند و اگر دوره زمانسجی پایان یافته باشد با یک شدن ورودی R خروجی Q صفر می شود. در شـکل(۴-۳) دیگرام زمانی این زمانسنج ارائه شده است. این زمانسنج را با طرح ارائه شده در مجاور آن بررسی کنید.



- معرفي نحوه استفاده از Help Simatic Manager

ب: Retentive On Delay Timer

کار این زمانسنج مشابه On Delay Timer است با این تفاوت که این زمانسنج، با لبه بالا رونده ورودی S شروع بـه زمانسنجی میکند و برای ادامه کار نیاز به باقی ماندن این ورودی در حالت یک ندارد. اگر در طول دوره زمانسنجی، در ورودی S مجددا لبه بالا رونده ایجاد شود عمـل زمانسـجی مجـددا از ابتـدا آغـاز مـیشـود. در شکل(۴-۴) دیاگرام زمانی این زمانسنج ارائه شده است. این دیاگرام زمانی را توسط طرح ارائه شده در مجاور آن بررسی کنید.



۴-۱-۴ زمانسنج با تاخیرخاموش شونده با

Off Delay Timer

در این زمانسنج. با یک شدن ورودی S خروجی Q یک میشود و با لبه پائین رونده ایـن ورودی زمـانسـنج شـروع بـه زمانسنجی کرده و روند زمانسنجی تا زمان تنظیم شده در ورودی TV ادامه پیدا میکند. خروجی Q تا پایان دوره زمانسـنجی در حالت یک باقی مانده و در پایان این دوره صفر می شود. در دوره زمانسنجی، در خروجی BI زمانسنج، زمـان بـاقی مانـده از دوره زمانسنجی به صورت کد باینری و در خروجی BCD آن، این زمان به صورت کد BCD در دسترس هستند.

اگر در طول دوره زمانسنجی ورودی S یک شود روند زمانسنجی متوقف شده و خروجی Q در حالت یک باقی میماند. و بـا صـفر شدن مجدد این ورودی زمانسجی مجددا از ابتدا آغاز شده و در پایان دوره زمانسنجی خروجی Q صفر میشود.

در این زمان سنج هرگاه ورودی R یک شود خروجی Q صفر می شود، و اگر این یک شدن(لبه بالا رونده) در طول دوره زمانسنجی رخ دهد، زمانسنجی متوقف شده و زمان باقی مانده از دوره زمانسنجی در هر دو خروجی صفر میشوند.

در شکل(۴–۵) دیاگرام زمانی این زمانسنج ارائه شده است. این دیاگرام زمانی را توسط طرح ارائه شده در مجاور آن بررسی کنید.



شکل(۴–۵)

توضیح ۱: اگر در زمانسنجها در شرایطی که ورودی R در حالت یک است، لبه فعال کننده زمانسنجی در وروی S، عمل کند، این سنجش انجام نمیشود، حتی اگر پس از عمل کردن لبه ورودی S ورودی R غیر فعال شود. چون زمانسنجها با لبه ورودی S شروع به کار می کنند.

- فرمت آدرسدهی خروجی زمانسنجها در جاهای مختلف یک برنامه

برای آدرس دهی جهت دسترسی به مقدار زمان باقیمانده در خروجی BI زمان سنجها و همچنین برای آدرس دهی وضعیت خروجی بیتی (Q) آنها در جاهای مختلف یک برنامه، میتوان بصورت آدرس آمده در برنامه شکل(۴-۶) عمل کرد. در این برنامه در شبکه ۲ خروجی BI زمان سنج T4 بصورت Word و در شبکه ۳ وضعیت Q این زمان سنج به صورت بیت انتقال می-یابند. بطوریکه مشاهده می شود در این مثال برای انتقال خروجی BI زمان سنج و انتقال وضعیت Q آن از فرمت آدرس دهی مشابه (T4) استفاده شده است ولی نرمافزار با توجه به محل آدرس دهی، نوع داده را تشخیص میهد.



شکل(۴-۶)

استفاده از آدرس برای تنظیم زمان سنجش(TV):

برای تنظیم مقدار زمانسنجش(TV) توسط آدرس به طول یک Word لازم است بصورت جدول نشان داده شده در شکل(۴-۲) مقدار دهیشود. بطوریکه ۱۲ بیت کم ارزش تر این Word یک عدد سه رقمی به صورت کد BCB و دو بیت بعدی آن مربوط به کد زمان پایه است. طرح آمده در شکل زیر را برای زمانهای مختلف بررسی کنید. بطور مثال در جدول آمده در شکل، مقدار زمان، ۱۵ ثانیه تنظیم شده است. در این مثال مقدار xx برابر 10 باینری است.



Network 1: on-delay timer with time value in memory word

۴-۱-۴ زمانسنجهای کوئیلی

اگر در طراحی یک برنامه لازم شود کارهای Reset ، Set و یا انتقال دادههای مربوط به ورودی و خروجی یک زمان-سنج، در شبکههای مختلف انجام شود، میتوان از زمانسنجهای کوئیلی استفاده کرد. برای آشنائی با این نوع کاربردها، در ادامه مثالی از یک زمانسنج پالسی ارائه شده که طرح آن در برنامه شکل (۴-۸) نشان داده شده است.

در این طرح، در شبکه 1 زمانسنج پالسی T0 با فعال شدن I0.0 برای اندازه گیری زمان دو ثانیه شروع به زمانسنجی میکند. در شبکه 2 زمان باقیمانده از زمانسنجی بصورت کد باینری به خروجی QW4 منتقل میشود و در شبکه 3 وضعیت خروجی Q ایـن زمانسنج به خروجی Q4.7 منتقل میشود. در شبکه 4 با فعال شدن I0.1 کار زمانسنجی این زمانسنج متوقف شده و خروجی Q و مقدار زمان باقیمانده آن صفر میشود. این مثال را بررسی شود

Network 1: Pulse Timer with coil
I0.0 T0 (SP)
Network 2: Title:
المان التقال المان ال كان المام يمبورت بايدزي الت المان
T0 Q4.7
Network 4: Title:
T0 T0 (R) (R)

شکل(۴–۸)

شکل(۴–۷)

بقیه زمانسنجهای کوئیلی شامل (SE) ، Extended Pulse Timer (SE) ، On Delay Timer (SD) ، Extended Pulse Timer (SE) را می توان مانند طرح قبل بررسی کرد. تمرین (SF) اطاقی بمنظور دوش هوا با مشخصا ت زیر در مسیر ورودی اطاق تمیز ، برای گرد و غبار روبی پرسنلهائی که وارد اطاق تمیز می شوند طراحی شده است. درب A : درب ورودی اطاق دوش هوا است که مجهز به یک میکروسوئیچ (MS_A) بوده، که با باز شدن این درب کنتاکت آن باز و با بسته شدن درب کنتاکت آن بسته می شود. همچنین این درب دارای یک قفل برقی (L_A) است که می توان با آن این درب را قفل کرد. درب B : درب خروجی اطاق دوش هوا (ورودی به اطاق تمیز)مجهز به یک میکروسوئیچ (MS_B) که با باز شدن این درب کنتاکت آن باز و با بسته شدن درب کنتاکت آن بسته می شود. همچنین این درب دارای یک قفل برقی (MS_B) که با باز شدن این درب کنتاکت آن باز و با بسته شدن درب کنتاکت آن بسته می شود. میز میزه میزه به یک میکروسوئیچ (MS_B) است که می توان با آن این درب

> با آن این درب را قفل کرد. M: موتور فن اطاق دوش هوا

Opto: حسگر تشخیص دهنده پرسنل در داخل اطاق دوش هوا که با حضور پرسنل خروجی آن یک می شود.

برنامهای طرح کنید تا با بسته شدن درب B قفل درب A باز شود و با باز شدن درب A درب B قفل شود. و با ورود پرسنل به داخل اطاق دوش هوا (از مسیر درب A) و با بسته شدن این درب، موتور فن به مدت سه دقیقه شروع به کار کند. در پایان این دوره از زمان فن متوقف، درب A قفل شده و قفل درب B باز شود تا پرسنل بتواند وارد اطاق تمیز شود.



تمرین۲: برنامهای طرح کنید تا خروجی Q4.0 به یک اسیلاتور موج مربع با دیوتی سایکل ۵۰٪ تبدیل شده، بطوریکه فرکانس آن از طریق ورودی IW0 قابل تغییر باشد.

تمرین ۳: مدارهای شکل زیر مربوط به مدار قدرت و فرمان راه اندازی موتورهای قدرت با تغییر مقاومت راهانداز بـرای روتـور آن است. برای مدار قرمان این شکل برنامهای با زبان LAD طرح کنید.



۲-۴ بررسی شمارندها Counters

مقدمه

یکی دیگر از المانهای مهم در PLC شمارندهها هستند که برای عمل شمارش در برنامه، از آنها استفاده می شود. انـواع شمارنده های موجود در نرم افزار Simatic Manager در جدول زیر نشان داده شده است. این شمارندها شامل: بالا و پائین شمار، فقط بالا شمار و فقط پائین شمار هستند. این شمارندهها در دو شکل مختلف در برنامه بکار گرفته می شوند. سه شمارنده نمایش داده شده در بالای جدول برای استفاده به شکل بلوکی.و سه کوئیل زیرین آنها برای بکار گیری این سه شمارنده به شکل کـوئیلی هستند که در این بخش نحوه بکار گیری این شمارندهها در برنامه، بایک.مثال برای هر یک بررسی می شوند.

Up & Down Counter Up Counter Down Counter Set Counter Count Up Count Down	Counter
	DB call

تعداد شمارندهای یک PLC بستگی به توانائی CPU آن PLC دارد. حافظه اختصاص یافته برای هر شمارنده به منظور نگهداری مقدار شمارش یک Word است که میتوان بصورت یکی از دو فرمت زیر برای آنها مقدار اولیه (PV) تعین کرد.

– فرمت تعين مقدار اوليه شمارنده بصورت مقدار ثابت

برای تنظیم مقدار اولیه یک شمارنده بصورت مقداردهی ثابت از فرمت C#0nnn یا W#16#0nnn استفاده می شود. در این فرمتها 0nnn عدد دسیمال سه رقمی هستند که توسط طراح به عنوان مقدار اولیه شمارنده مشخص می شود. گستره تنظیم این مقدار از صفر تا ۹۹۹ است.که مقدار مشخص شده آن در موقع برنامه نویسی وارد برنامه می شود. مثال: C#107 C#24

– فرمت تعین مقدار اولیه شمارنده با استفاده از آدرس دهی

برای تنظیم مقادیر اولیه مختلف برای یک شمارنده، با استفاده از نتایج پردازش برنامه، از روش آدرس دهی استفاده می می مود. طول این آدرس به اندازه یک Word است که لازم است محتوای آن در محدوده صفر تا ۹۹۹ و بصورت کد BCD باشد. مثال: QWm IWm MWm در این آدرسها m شماره بایت با ارزشتر حافظه مربوطه است.

۴–۲–۱ بررسی یک شمارنده بالاشمار و یک شمارنده پائینشمار در شکل (۴–۹) طرحهای یک شمارنده بالاشمار و یک شمارنده پائینشمار ارائه شده است که نحوه عملکرد آنها بصورت موارد آمده در زیر است. الف– در این شمارندهها با لبه بالا رونده ورودی(بیتی) S ، خروجی V^O ها به مقدار ^۲PV مقدار دهی شده و خروجی Q آنها یک میشود ب– با فعال شدن ورودی بیتی R هر یک از این شمارندهها، مقدار V^O و خروجی Q آنها صفر میشود. در ادامه : ج– در شمارنده بالاشمار، با هر لبه بالا رونده ^TCU یک واحد به خروجی VD این شمارنده اضافه شده و این روند تا عدد ۹۹۹ ادامه خواهد یافت.

Counter Value

[°] Pre value

[•] Count Up

د- در شمارنده پائینشمار با هر لبه بالا رونده CD^۱ یک واحد از خروجی CV این شمارنده کم شده و این روند تا عدد صفر ادامـه خواهد یافت. با صفر شدن مقدار CV خروجی Q هم صفر می شود



۲-۲-۴ بررسی شمارنده بالا و پائین شمار

در شکل(۴–۱۰) یک شمارنده بالا و پائین شمار ارائه شده است که نحوه عملکرد آنها بصورت موارد آمده در زیر است. الف- با لبه بالا رونده ورودی(بیتی) S، خروجی CV به مقدار PV (محتوای آدرس MW) مقدار دهی شده و خروجی Q یک می-شود.

ب- با هر لبه بالا رونده CU یک واحد به خروجی CV این شمارنده اضافه شده و این روند تا عدد ۹۹۹ ادامه خواهد یافت. ج- با هر لبه بالا رونده CD یک واحد از خروجی CV این شمارنده کم شده و این روند تا عدد صفر ادامـه خواهـد یافـت. بـا صـفر شدن مقدار شمارنده خروجی Q صفر میشود.

د- با فعال شدن ورودی R مقدار CV و خروجی Q صفر خواهد شد.



توضیح: هر یک از سه شمارنده آمده در بالا دارای دو خروجی بنامهای CV و CV_BCD، و هر یک بطول یک Word هستند. مقادیر این دو خروجی مانند هم ، ولی فرمت دادههای آنها متفاوت میباشند. بطوریکه فرمت داده در خروجی CV بصورت کد هگزا دسیمال و فرمت داده در خروجی CV_BCD بصورت کد BCD هستند.

۴-۲-۴ شمارنده بالا و پائین شمار بصورت کوئیل

اگر در طراحی برنامه لازم شود کارهای Reset ،Set ، عمل شمارش و انتقال دادههای مربوط به یک شمارنده در شبکه-های مختلف انجام شود، برای این منظور میتوان از شکل کوئیلی شمارندهها استفاده کرد. برای آشنائی با کاربرد شکل کوئیلی شمارندهها در شکل (۴–۱۲) مثالی از یک شمارنده ارائه شده است. **در این مثال،** در شبکه 1 با فعال شدن ورودی IO.0 شمارنده CO به مقدار ۹ واحد مقداردهی اولیه میشود. در شبکه 4 مقدار موجود در شمارنده CO بصورت کد هگزا دسیمال به خروجی QW4 منتقل و در شبکه 5 خروجی بیتی شمارنده CO به Q4.7 منتقل می شود. در شبکه 2 به ازاء هر لبه بالارونده I0.1 یک واحد از مقدار شمارنده کم میشود تا مقدار شمارنده صفر شود. در شبکه 3 به ازاء هر لبه بالارونده I0.2 یک واحد به مقدار شمارنده C0 اضافه شده تا مقدار شمارنده ۹۹۹ شود. در شبکه 6 با فعال شدن ورودی I0.3 مقدار شمارنده صفر و در نتیجه خروجی Qw4 در شبکه 4 و خروجیQ4.7 در شبکه 3 صفر میشوند. این مثال را بررسی کنید.



شکل(۴–۱۱)

توضیح : در این نوع طراحی برای مقدار دهی اولیه شمارندهها، میتوان بجای مقدار دهی اولیه با فرمت ثابت C#n از آدرس بطول یک Word برای مقدار دهی بصورت متغیر استفاده کرد. همچنین در طراحی برای شمارندههای فقط بالاشمار از کوئیـل CD و در شمارنده فقط پائینشمار از کوئیل CU استفاده میشود.

تمرین: برنامهای بنویسید، بطوری که اگر ورودی I0.0 یک است با هر بار از صفر به یک رفتن ورودی I0.1 یک واحد به خروجی QW4 اضافه شود(ماکزیمم تا 999). همچنین اگر ورودی I0.1 یک است با هر بار از صفر به یک رفتن ورودی I0.0 یک واحد از خروجی QW4 کم شود(مینیمم تا صفر).

جلسه پنجم

هدف:

- معرفی فرمت اعداد صحیح علامت دار ۱۶بیتی و ۳۲ بیتی،
 - معرفی فرمت اعداد حقیقی
 - معرفی فرمت داده ها بصورت باینری و یا هگزا دسیمال
- بررسی مقایسه کنندههای برای اعداد صحیح علامتدار و حقیقی
- بررسی توابع ریاضی برای اعداد صحیح علامت دار و اعداد حقیقی
 - بررسى توابع مثلثاتى
 - بررسی توابع منطقی

۱-۵) فرمت دادهها برای توابع ریاضی و منطقی

مقدمه

طول داده توابع ریاضی و منطقی در PLC که با اعداد صحیح علامتدار و یا بدون علامت کار میکنند ۱۶ بیتی و یا ۳۲ بیتی است. اعداد صحیح علامت دار ۱۶ بیتی بنام integer و ۳۲ بیتی آن بنام Double Integer نامیده میشوند. توابعی که با این نوع اعداد کار میکنند، مقادیر ثابت در ورودی خود را با فرمت اعداد صحیح علامت دار دسیمال میپذیرند. همچنین داده-های بدون علامت ۱۶ بیتی بنام Word و ۳۲ بیتی آن بنام Double Word نامیده میشوند. توابعی که با این نوع دادهها کار میکنند، مقادیر ثابت را در ورودی خود، با فرمت هگزا دسیمال و یا باینری میپذیرند.

در PLC، طول داده برای توابعی که با اعداد حقیقی^۱ کار میکنند ۳۲ بیتی است. فرمت این داده بشکل ممیز شناور^۲ و منطبق با استاندارد IEEE FP 32 است. در ادامه فرمت هر یک از دادهها ذکر شده در بالا معرفی میشوند.

۱−۱−۵) معرفی مجموعه اعداد صحیح علامت دار ۱۶ بیتی (Integer)

در شکل زیر فرمت اعداد صحیح علامت دار ۱۶ بیتی نشان داده شده که پر ارزشترین بیت آن بیت علامت و ۱۵ بیت دیگر مقدار ارزش اعداد را مشخص میکند. اگر بیت علامت یک باشد محتوای ۱۵ بیت دیگر بصورت کد مکمل^۳ باینری مشخص کننده ارزش عدد صحیح منفی و اگر بیت علامت صفر باشد محتوای ۱۵ بیت دیگر بصورت کد باینری مشخص کننده ارزش عدد صحیح مثبت هسنند.



نحوه مقدار دهی ثابت: توابعی که در ورودی خود اعداد Integer میپذیرند برای مقداردهی ثابت، بصورت فرمت و گستره آمـده در زیر عمل می شوند.

¹ Real

² Floating Point

^r Tow's Complement

فرمت برای مقدار دهی اعداد مثبت ثابت، بصورت N+ است که N میتواند صفر تا ۳۲۷۶۷ باشد همچنین برای مقدار دهی اعـداد منفی ثابت، فرمت بصورت N– است کهN میتواند یک تا ۳۲۷۶۸ باشد.

1-۵-۲)معرفی مجموعه اعداد صحیح علامت دار ۳۲ بیتی (Double Integer)

در زیر فرمت اعداد صحیح علامت دار ۳۲ بیتی نشان داده شده که با ارزشترین بیت آن بیت علامت و ۳۱ بیت دیگر مقدار ارزش اعداد را مشخص میکنند. اگر بیت علامت یک باشد محتوای ۳۱ بیت دیگر بصورت کد مکمل۲ باینری مشخص کننده ارزش عدد صحیح منفی و اگر بیت علامت صفر باشد محتوای ۳۱ بیت دیگر بصورت کد باینری مشخص کننده ارزش عدد صحیح مثبت هستند.



نحوه مقدار دهی ثابت: توابعی که در ورودی خود اعداد Double Integer می پذیرند برای مقداردهی ثابت، بصورت فرمت و گستره آمده در زیر مقدار دهی می شوند.

فرمت برای مقدار دهی اعداد مثبت ثابت، بصورت L#+N است که N می تواند صفر تا ۲۱۴۷۷۸۳۶۷ باشد همچنین برای مقدار دهی اعداد منفی ثابت، فرمت بصورت L#-N است که N می تواند یک تا ۲۱۴۷۷۸۳۶۸ باشد.

۵-۱-۵) معرفی مجموعه داده بدون علامت ۱۶ بیتی

فرمت دادههای بدون علامت ۱۶ بیتی در زیر نشان داده شده است. توابعی که در ورودی خـود ایـن نـوع دادههـا را مـی پذیرند، برای مقدار دهی ثابت در ورودی آنها میتوان بصورت باینری، هگزا دیسیمال و یا بایت ثابت، بصورت فرمتهای نشان داده شده در مثالهای زیر مقدار دهی کرد.

D	ata 1	Гуре	Ŀ	eng	th (b	its)		Forn	nat			Ec	orma	nt Ex	amp	les			
												Μ	in.		Ma	ax.			
N N	VOR	D	1	16				Bina	ry			23	#O		2#	:111	1111	1111	111111
								Hexa	adec	imal		W	#16	#0	W;	# 16≢	#FFF	F	
								Unsi	igne	d byl	tes	B	#(0,0))	B≉	ŧ(25	5,259	5)	
							Ŋ	Wor	dn										
,	_			Bγt	e n							Bγt	e n-	+1					
MSB	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0		3	
MSB	<u> </u>	-		•	-					•	-						LSE	3	

توضیح: توابعی که در ورودی خود دادههای بدون علامت، ۱۶ بیتی (Word) می پذیرند، علاوه بر فرمتهای آمده در بالا میتوانند در ورودی خود داده بطول بایت را هم بصورت فرمت B#16#nn بپذیرند که در این فرمت nn یک عدد هگزا دسیمال دورقمی است.

۵-۱-۵)معرفی مجموعه داده بدون علامت ۳۲ بیتی

فرمت دادههای بدون علامت ۳۲ بیتی در زیر نشان داده شده است. توابعی که در ورودی خود ایـن نـوع دادههـا را مـی پذیرند، برای مقدار دهی ثابت در ورودی آنها می توان بصورت باینری، هگزا دیسیمال و یا بایت بدون علامت، به شکل فرمتهای نشان داده شده در مثالهای زیر مقدار دهی کرد.

Data Ty)e	Le	ngth	(bit	s)	Fo	rma	<u>t</u>		ļ	Form	iat E	xam	ples					
DWORD)	32				Bi	nary,	min).		2#00	0000	0000	0000	000	0000	000	0000	0000000
						Bi	nary,	ma	Х.		2#11	111	1111	1111	111	1111	11	1111	1111111
											Min.			Ма	Х.				
						He	exad	ecim	nal		D₩#	±16#	0	D۷	V#1 6	i#FFI	FFF	FFF	
						Ur	nsigr	ned k	oytes	;	B#(0	,0,0,	0)	B#I	(255	,255,	,25	5,255	5)
,		Byt	en		_	Byt	e n+	-1		Ву	te n+ ~	+2		Byte	e n+	3			
мѕв	7			0	7			0	7			0	7			0	 L	.SB	

توضیح: توابعی که در ورودی خود دادههای بدون علامت، ۳۲ بیتی (DWord) می پذیرند، می توانند در ورودی خود داده ثابت بطول بایت را با فرمت B#16#nn و یا داده ثابت بطول Word را با فرمت W#16#nnnn بپذیرند که در این فرمتها هر یک از nها یک عدد هگزا دسیمال یک رقمی می باشند.

۵-۱-۵) معرفی مجموعه اعداد حقیقی ا

در PLCهای شرکت زیمنس اعداد حقیقی با فرمت ممیز شناور (با استاندارد IEEE FP 32) که ۳۲ بیتی است، استفاده می شوند. فرمت این اعداد در زیر نشان داده شده است.

S7 CPU memory format for the floating-point number (32 bits)

		Byte n	,		E	Bytes n+	1, n+2, i	n+3	
		2 ⁷		2 ⁰	2 ⁻¹	2 ⁻²		2 ⁻²²	2 ⁻²³
Si	an bit (s)	×8-bi	t exponer	nt (e)	·~	23-b	it mantis:	sa (f)	

بطوریکه از روی فرمت این اعداد ۳۲ بیتی مشاهده می شود، با ارزش ترین بیت آن مختص بیت علامت(s)، ۸ بیت مجاور بیت علامت بطور یک از این بیت ما بیت مجاور بیت علامت برای نما (e) و ۳۲ بت باقی مانده برای مانتیس (f) است که ارزش هر یک از این بیت ها در این فرمت مشخص شده است. بر طبق استاندارد 32 IEEE FP ، استفاده از مقادیر بیت علامت، نما و مانتیس در رابط و زیر، ارزش اعداد ممیز شناور مورد استفاده در این می شود. درادامه توضیحات دقیق تری از این متغیرها و رابط تعین کننده ارزش اعداد شناور ارائه شده است. معنور می بر طبق استاندارد 32 TP بت باقی مانده برای مانتیس (f) است که ارزش هر یک از این بیت ها در این فرمت مشخص شده است. بر طبق استاندارد 32 IEEE FP می می نده از می می به می بر طبق استاندارد 32 IEEE با معیز شناور ارائه شده است. استفاده از مقادیر بیت معلمت، نما و مانتیس در رابط تعین کننده ارزش اعداد مناور ارائه شده استفاده در است می شود. درادامه توضیحات دقیق تری از این متغیرها و رابطه تعین کننده ارزش اعداد شناور ارائه شده است.

ت*وجه* شود در این استاندارد فاصله دو عدد متوالی برای اعداد کوچک ، کوچکتر و برای اعددا بزرگ، بزرگتر میشود. این مطلب را با بررسی رابطه تعین کننده ارزش اعداد شناور که در زیر آمده است میتوان برداشت کرد.

S7 CPUs use the IEEE FP 32-bit binary memory format to process floating point numbers:

Any floating-point number = $(-1^{5})(1, f)(2^{e-127})$

Where in the normalized number equation shown above:

s = sign bit, (0 = positive, 1 = negative)

e = a binary integer exponent = the decimal integer exponent + 127 as an 8-bit unsigned binary integer exponent (0 < e < 255)

f = a 23-bit mantissa with the MSB equal to 2^{-1} and LSB equal to 2^{-23} .

در زیر گستره دامنه اعداد مثبت و منفی در این استاندارد نشان داده شده است.

<u>Data Type</u>	<u>Length (bits)</u>	Format	<u>Format Examples</u>	
			positive min.	positive max
REAL	32	Floating point	+1.175495e-38	+3.402823e+38
			negative min.	negative max.
			-1.175495e-38	-3.402823e+38

نحوه مقدار دهی ثابت: توابعی که در ورودی خود اعداد را بصورت اعداد حقیقی دریافت میکنند، برای مقدار دهی ثابت بدو صورت زیر عمل میشود.

صورت کسری: $pm M \pm N.M$ که در آن N مقدارعدد صحیح و M بخش کسری عدد است. اگر هر یک از اعداد M و N وجود نداشته باشند بجای آنها صفر منظور می شود. صورت نمائی: $pm M \pm Ne \pm P$

که در آن N مقدار عدد صحیح و P بخش نمای عدد 10 است. اگر در این عدد بخش کسری وجود نداشته باشد بجای P صفر منظور می شود.

(Comparator) بررسی توابع مقایسه (۲-۵)

مجموعه توابع عملیات مقایسه برای اعداد صحیح علامتدار (Integer و Double Integer) و اعداد حقیقی (Real) در سمت راست شکل(۵–۱) آمده است.

این توابع از بالا به پائین شامل مقایسه دو عدد صحیح علامتدار ۱۶ بیتی (Integer) برای حالات: مساوی، نا مساوی، بزرگتر، کوچکتر، بزرگتر و یا مساوی، کوچکتر و یا مساوی آمده است، در ادامه این مجموعه توابع دو مجموعه توابعی دیگر برای مقایسه دو عدد صحیح علامت دار ۳۲ بیتی (Double Integer) و برای مقایسه دو عدد حقیقی (نظیر توابع نامبرده شده در بالا) آمده است. این دو مجموعه توابع مشابه مجموعه توابع اعداد Integer هستند

در سمت چپ شکل(۵–۱) یک تمرین با دو مثال در دو شبکه ارائه شده است که در شبکه اول عمل مقایسه دو عدد صحیح علامت دار ۱۶ بیتی(Integer) برای تعیین بزرگتر و یا مساوی و در شبکه دوم عمل مقایسه دو عدد حقیقی (Real) برای تعین کوچکتر و یا مساوی انجام میشود. کار این دو مثال بررسی شود.

توجه شود، در این نوع توابع، ورودی IN1 با ورودی IN2 مقایسه شده و نتیجه بصورت یک بیت در خروجی ظاهر می شود.



شکل(۵–۱)

تمرین: برنامهای آماده کنید تا ورودی IW0 را با اعداد 20 و 10 مقایسه کرده بطوریکه اگر ورودی کوچکتر از 10 بود خروجی Q4.0 یک و اگر ورودی بزرگتر از 20 بود این خروجی صفر شود و اگر ورودی بین دو عدد بود خروجی تغییر نکند.

−۵) بررسی توابع عملیات ریاضی عدد صحیح (Integer Function)

مجموعه توابع عملیات ریاضی برای اعداد صحیح علامتدار در سمت راست شکل (۵-۲) آمده است. این توابع از بالا بـه پائین شامل جمع، تفریق، ضرب، تقسیم برای اعداد با طول ۱۶ بیتی(Integer) و در ادامه توابع جمع، تفریق، ضرب، تقسیم و تعین باقیمانده برای اعداد با طول ۳۲ بیتی (Double Integer) میباشند.

مثال: در سمت چپ شکل زیر یک مثال برای سه تابع ریاضی ارائه شده است که:

توضيح: در اين بلوکها اگر نتيجه خارج از گستره قابل قبول قرار گيرد خروجی ENO صفر می شود. ايـن موضـوع را در يکـی از بلوکهای تفريق و يا تقسيم بررسی کنيد. برای اين بررسی لازم است از ENO استفاده شود. مثلا نتيجـه ENO بـه يـک بيـت از حافظه منتقل شود.

OB1 : "Main Program Sweep (Cycle)"

Network 1: Title:



شکل (۲-۵)

تمرین، برنامهای آماده کنید تا مقادیر ورودی IB0 را بر IB1 تقسیم کرده، خارج قسمت این تقسیم را به خروجی QB4 و باقی-مانده آنرا به خروجی QB5 منتقل کند. مقادیر دو ورودی IB0 و IB1 دو عدد مثبت هستند.

وضعیت خروجی EN0 تقسیم کننده را به یک بیت حافظه منتقل کرده، و بررسی شود در شرایطی که عمل تقسیم بر صفر انجام می شود. وضعیت این بیت چگونه است؟

یادآوری: داده ها در ورودی های IN1 و IN2 و IN2 این توابع بصورت Integer و یا Double Integer است. اگر در ورود تابع nateger مقدار ثابت مقدار ثابت لازم باشد، این مقدار بصورت عدد دسیمال علامت دار و همچنین اگر در ورودی تابع Double Integer مقدار ثابت لازم باشد، این مقدار بصورت الله معدد دسیمال علامت دار است) تعین می شود.

4-4) توابع عمليات رياضي عدد حقيقي(floating Point Function)

مجموعه توابع عملیات ریاضی برای اعداد حقیقی در سمت راست شکل(۵-۳) آمده است. ایـن توابـع از بـالا بـه پـائین شامل جمع، تفریق، ضرب، تقسیم، قدرمطلق، جذر، مربع، LN، EXP و در ادامه توابع مثلثاتی و عکس توابع مثلثـاتی ایـن اعـداد میباشند.

توجه شود، واحد زوایا در توابع مثلثاتی رادیان است.

مثال: در سمت چپ شکل(۵–۳)یک مثال برای بررسی دو تابع ریاضی ارائه شده است. در این مثال دو عدد حقیقی ثابت از هـم کم و از نتیجه آن قدرمطلق گرفته شده است.



تمرین۱: برنامهای آماده کنید تا آن برنامه از عدد <u>یک</u> سینوس معکوس گرفته، نتیجه را در عدد ۲ ضرب و از حاصل آن کسینوس بگیرد.

تمرین۲:برنامهای طراحی کنید تا آن برنامه دو عدد Integer به نامهای IN1 و IN2 را از دو ورودی PLC دریافت کرده ، سپس ورودی IN1 را به توان IN2 رسانده نتیجه را بخروجی PLC با فرمت Double Integer منتقل کند

۵-۵) توابع عمليات جابجائي و چرخشي (Shift و Rotate)

مجموعه توابع جابجائی و چرخشی برای اعداد صحیح علامتدار و داده بدون علامت ۱۶ و ۳۲ بیتی در شکل(۵-۴) آمده است. این توابع از بالا به پائین شامل جابجائی به سمت راست برای اعداد Integer و Double Integer (اعداد صحیح علامت دار ۱۶ و ۳۲)، جابجائی به سمت راست و چپ برای داده بدون علامت ۱۶ بیتی (Word) و در ادامه جابجائی به سمت راست و چپ برای داده بدون علامت ۳۲ بیتی (Double Word) و در نهایت چرخش به سمت چپ و راست برای داده بدون علامت ۳۲ بیتی آمده است. در همه این توابع، ورودی IN مربوطه به عددی است که جابجائی یا چرخش باید روی آن انجام شود، و ورودی N مربوطه به عددی است که مقدار آن تعداد دفعات جابجائی یا چرخش برای عدد IN را تعین میکند.

باید توجه شود در همه این توابع فرمت ورودیN بصورت هگزا دسیمال ۱۶ بیتی (Word) بوده و اگر لازم باشد مقدار این ورودی ثابت نعین شود، فرمت آن بصورتW#16# است.

مثال: در سمت چپ شکل(۵-۴) یک مثال برای آشنائی با توابع جابجائی و چرخشی در سه شبکه ارائه شده است - در شبکه اول عدد صحیح علامت دار 100 دسیمال (که معادل باینری 0010 0100 0000 0000 است) به تعداد N=3 مرتبه جابجا و نتیجه آن که معادل عدد 12 دسیمال (که معادل باینری آن 1100 0000 0000 0000 است) به خروجی منتقل شده است.

توجه شود در عمل شیفت به راست اگر عدد صحیح علامتدار مثبت باشد با هر مرتبه جابجائی به راست، <u>صفر</u> و اگر منفی باشد <u>یک</u> از سمت چپ وارد داده ۱۶ (یا ۳۲) بیتی می شود. یا به عبارت دیگر با هر بار جابجای عدد علامت دار مثبت و یا منفی، تقسیم بر دو می شود. برای عدد مثبت کمترین مقدار نتیجه صفر و برای عدد منفی این نتیجه 1- خواهد بود. - در شبکه دوم داده بدون علامت ۱۶ بیتی 8fff هگزا دسیمال(که معادل باینری آن ۱۱۱۱ ۱۱۱۱ ۱۱۱۱ ۱۱۱۵ IN= است) به انداز 3=N مرتبه جابجا و نتیجه آن که معادل ۱۱۱۴ (۱۱۱۱ ۱۱۱۱ ۵۰۵۱) است به خروجی منتقل شده است.


Network 8: Title:

	ROR_I EN	DW ENO	
16#00000fff W#16#FFF-	IN	OUT	16#e00001ff -MD200
16#0003 IWO -	N		

شکل(۵–۴)

توضیح: چون شیفت به چپ برای Int و Dint مشابه شیفت به چپ Word و Dword هستند، این نوع توابع در نرم افزار نیامده است.

تمرین: شانزده عدد لامپ که بر روی یک دایره نصب شده و به خروجی شانزده بیت QW4 وصل شدهاند. برنامه ای آمده کنید تا با اجرای آن در هر ثانیه فقط یکی از لامپها روش و ترتیب روشن شدن لامپها در جهت عقربه ساعت باشد.

8-4) توابع عمليات Word Logic

مجموعـه توابـع Word Logic بـرای دادههـای Word (۱۶ بیتـی) و Double Word (۳۲ بیتـی) در سـمت راسـت شکل(۵-۵) آمده است. این توابع Logic از بالا به پائین شامل XOR، AND، NR، بـرای دادههـای ۱۶ بیتـی و در ادامـه بـرای دادههای ۳۲ بیتی هستند.

مثال: در سمت چپ شکل زیر یک مثال برای بررسی یک تابع OR، (۱۶ بیتی) ارائه شده است. در این مثال ورودی IN1 تابع (که دارای عدد ثابت 555 هگزا دسیمال با فرمتW#16#0555 است) با ورودی IN2 تابع (که دارای عدد 0003 هگزا دسیمال دریافت شده از ورودی دیجیتال با آدرس IW0 است) بیت به بیت OR و نتیجه آن (که معادل عدد 0557 هگزا دسیمال و یا 0111 0110 0101 0000 باینری است) از خروجی (Out) تابع به حافظه به طول یک Word (00W300) منتقل شده است.



تمرین۱: برنامهای آماده کنید تا دو داده را از ورودیهای IB0 و IB1 بگیرد و NOR آنها را در خروجی QB4 تحویل دهد.

مثال : برنامهای ایجاد کنید که با آن برنامه، دو عدد A و B با فرمت BYTE را از ورودهای IB0 و IB1 دریافت کرده و باهم مقایسه کند، اگر عدد A بزرگتر از عدد B بود حاصل عمل A-B را پس از دو واحد شیفت به راست به خروجی QW4 منتقل کند و اگر A کوچکتر از B بود حاصل عمل A+B را پس از دو واحد شیفت به چپ به خروجی QW4 منتقل کند در صورت مساوی بودن دو عدد A و B، خروجی قبلی تغیر نکند.



پاسخ تمرین بخش۵-۲



پاسخ:

جلسه ششم

هدف:

- بررسی مبدل های مختلف فرمت اعداد در PLC شامل: فرمت integer به فرمت کـد BCD و بلعکـس، فرمت Integer به فرمت Double Integer ، فرمت Double Integer و بلعکـس، مـتمم ۱ اعداد^۱ integer و Double Integer، متمم دو^۲ اعـداد integer و Double Integer، منفـی کـردن فرمـت Real
 - بررسی بیت های وضعیت در عملیات CPU شامل،بیت OS ، ^۳OS ، ^۳OV ،
 - بررسی بیتهای وضعیت مربوط به نتایج عملیات ریاضی و منطقی نسبت به صفر
 ۹-۱) مبدلها^۷

مقدمه، مجموعه توابع عملیات تبدیل فرمتهای مختلف اعداد در شکل (۶–۱) نشان داده شده است. این مبدلها از بالا به پائین شامل:

' Invert

' Tow's complement

"Over Flow

^sStore Over Flow

[°]Binary Result

[`]Unordered

⁷ Converter

۶−۱−۱) مبدل فرمت BCD به Integer

طول هر یک از داده ورودی و خروجی این مبدل یک Word، و محدوده اعدادی را که در ورودی خود میتواند بپذیرد از ۹۹۹- تا ۹۹۹ است.

مثال: در سمت چپ شکل (۶–۲) مثالی از مبدل BCD سه رقمی به Integer ارائه شده است در این مثال:

در شبکه اول، ورودی مبدل، عدد مثبت ۲۵۵ (Olio 0101 0100 0000) BCD را از آدرس IW0 دریافت کرده است.

در شبکه دوم عدد **مثبت** ۲۵۵ (W#16#0255) به صورت مقدار ثابت، با فرمت BCD، (W#16#0255) به ورودی مبکله دوم عدد مثبت مبدل داده شده است.

در شبکه سوم عدد **منفی** ۲۵۵- (W#16#8255) به صورت مقدار ثابت، با فرمت BCD، (W#16#8255) به ورودی مبدل داده شده است.

مثال فوق را میتوان با نرم افزار PLCSIM بررسی کرد. نتایج هر سه شبکه این مثال در سمت راست شکل(۶-۲) آورده شده است. **توجه شود،** چون ورودی بلوک شبکه سوم عدد BCD منفی (یعنی با ارزش ترین بیت آن یک) بوده، در نتیجه فرمت عدد Integer خروجی آن به شکل کد **مکمل دو^** باینری است.



توضیح: فرمت عدد ثابت در ورودی تابع BCD_I بصورت W#!6#snnn که اها عدد سه رقمی BCD و s علامت آن که صفر برای عدد مثبت و ۸ برای عدد منفی است.

۲−۱−۶) مبدل فرمت اعداد Integer به BCD:

طول هر یک از دادههای ورودی و خروجی این مبدل یک Word و محدوده اعدادی را که در ورودی خود میتواند بپذیرد از ۹۹۹- تا ۹۹۹ است.

مثال : در سمت چپ شکل (۶–۳) مثالی از مبدل Integer به BCD سه رقمی ارائه شده است در این مثال:

در شبکه اول، ورودی مبدل ، عدد مثبت ۲۵۵ (Integer (0000 0000 1111 1111) را از آدرس IW0 دریافت کرده است.

در شبکه دوم عدد **مثبت ۲۵۵ (۱۱۱۱ ۱۱۱۱ 0000 (0000) به شکل مقدار ثابت، با فرمت Integer (255) به ورودی مبدل داده** شده است.

در شبکه سوم عدد منفی ۲۵۵- (۵۵۵ 0000 ۱۱۱۱ ۱۱۱۱)) به شکل مقدار ثابت، با فرمت Integer (255-) به ورودی مبدل داده شده است. این مثال را می توان با نرم افزار PLCSIM بررسی کرد. نتایج هر سه شبکه این مثال در سمت راست شکل(۶-۳)

[^]Twos Compliment

آورده شده است. **توجه شود،** چون ورودی تابع شبکه سوم عدد منفی Integer(کد مکمل دو) بوده، در نتیجه با ارزشترین چهار بیت خروجی آن یک شده است.

OB1 : "Main Program Sweep (Cycle Network 1/: Title:	=) "	
LBCD EN ENO 255 IWO - IN 0UT -QWO -QWO	QB 0 Bits	QB 1 Bits Image: Constraint of the second secon
Network 2 : Title:		
LBCD EN ENC 255 255 IN OUT - QW2	QB 2 Bits Image: Constraint of the second secon	QB 3 Bits 7 6 5 4 3 2 1 0 7 7 7 7 7 7 7 7 7 7
Network 3: Title:		
I_BCD	QB 4 Bits 💌	QB 5 Bits 💌
EN ENO -255 -255 IN OUT -2W4	76543210 V V V V	7 6 5 4 3 2 1 0 7 7 7 7 7
	شکل (۶–۳	

در اینجا از بررسی نحوه عملکرد مبدلهای BCD به Double Integer وDouble Integer به BCD صرف نظر شده است، چون این مبدلها مانند دو مبدل قبل عمل میکنند و تفاوت آنها در گستره پذیرش اعداد در ورودی این مبدلها است که این گسترهها از ۹۹۹۹۹۹۹ – تا ۹۹۹۹۹۹۹ و طول حافظههای ورودی خروجی آنها Double Word میباشند.

Real به Double Integer به Double Integer

طول هر یک از دادههای ورودی و خروجی این مبدل یک Double Word است.

مثال : در شکل (۶-۴) مثالی از مبدل Double Integer به Real ارائه شده است در این مثال داده ورودی مبدل از آدرس ID0 داده دریافت می شود:

٢- اگر به ورودی آن عدد منفی ٣- (که شکل Integer آن بصورت Integer ۱۱۱۱ ۱۱۱۱۱ ۱۱۱۱۱ ۱۱۱۱۱ ۱۱۱۱ خواهد بود) داده شود، در حافظه خروجی آن عدد تبدیل شده به فرمت Floating Point، به شکل 0000 0000 0000 0000 1100 1100
 0000خواهد شد.

توضیج: این مثال را میتوان با نرم افزار PLCSIM بررسی کرد. ورودی و خروجیهای این مبدلها در هر یک از حالتهای فوق در شکل(۶-۴) نشان داده شده است. توجه شود، برای تبدیل اعداد منفی Double Integer به Real فرمت اعداد منفی Double Integer باید بصورت مکمل دو باشد.

برای وارد کردن عدد ثابت به ورودی مبدلهائی که اعداد Double Intege می پزیرند از فرمت n(- or +) استفاده می شود، که برای مثال فرمت عدد مثبت ۳ به شکل L#3 و فرمت عدد منفی ۳- به شکل L#-3 است.

	EN EN 00000000003 ID0 - IN OU			
	IB 0 Bits 💌	IB 1 Bits 💌	IB 2 Bits 💌	IB 3 Bits 💌
3	7 6 5 4 3 2 1 0	76543210	7654 3210	7 6 5 4 3 2 1 0
	QB O Bits 💌	QB 1 Bits 💌	QB 2 Bits 💌	QB 3 Bits 💌
	76543210 ГГГГГГГ	76543210 ГГГГГГГ	7654 3210 ГГГГ ГГГГ	76543210
	IB 0 Bits 💌	IB 1 Bits 💌	IB 2 Bits 💌	IB 3 Bits 💌
-3	7 6 5 4 3 2 1 0 v v v v v v v	7 6 5 4 3 2 1 0 v v v v v v v	7 6 5 4 3 2 1 0 V V V V V V V V	7 6 5 4 3 2 1 0 V V V V V V V
	QB 0 Bits 💌	QB 1 Bits 💌	QB 2 Bits 💌	QB 3 Bits 💌
	7654 3210	7654 3210 ГГГГ ГГГГ	7654 3210	7654 3210

شکل (۴-۴)

۶-۱-۶) مبدل معکوس کردن (Ones Complement) اعداد صحیح علامت دار ۱۶ بیتی (Integer)

طول هر یک از دادههای ورودی و خروجی این مبدل یک Word و نتیجه عمل تبدیل در خروجی آن، NOT بیت-های ورودی (فرمتOnes Complement) است.

برای مثال اگر به ورودی طرح شکل(۶-۵) عدد مثبت ۱۲ با فرمت Integer که شکل باینری آن بصورت 0000 0000 0000 برای مثال ا 1100 خواهد بود، داده شود، عدد تبدیل شده در حافظه خروجی بصورت 1011 1111 1111 خواهد شد.



شکل(۶–۵)

عملکرد معکوس کردن اعداد صحیح علامت دار ۳۲ بیتی (Double Integer) مشابه مبدل معکوس کردن اعداد صحیح علامتدار ۱۶ بیت است و تفاوت آنها فقط در طول داده آنها است که اعداد TT Double Integer بیتی و ۱۶ Integer بیتی میباشند. بر این اساس در اینجا از بررسی مثالی از عملکرد معکوس کردن اعداد Double Integer صرف نظر میشود.

(Twos Complement) مبدل مکمل دو

طول هریک از دادههای ورودی و خروجی این مبدل Word و نتیجه عمل تبدیل این مبدل در خروجی، حاصـلضـرب عدد منفی یک در عدد Integer ورودی است.

برای مثال اگر به ورودی طرح شکل(۶-۶) عدد ۱۲ Integer ۱۲ (که شکل باینری آن بصورت 1100 0000 0000 است) داده شود در حافظه خروجی عدد تبدیل شده بصورت عدد ۱۲ Integer - خواهد شد(که معادل باینری آن بصورت 1111 1111 0100 (مکمل دو) است).

و همچنین اگر به ورودی آن عدد ۱۲ Integer ۲۰- (که شکل باینری آن بصورت 0100 1111 1111 (مکمل دو) است) داده شود در حافظه خروجی آن عدد تبدیل شده بصورت عدد ۱۲ Integer ۲۰خواهد شد (که معادل باینری آن بصورت 0000 0000 0000 1100 است).



شکل(۶-۶)

عملکرد مکمل دو اعداد صحیح علامت دار ۳۲ بیتی (Double Integer) مشابه مکمل دو اعداد صحیح علامتدار ۱۶ بیت است و تفاوت آنها فقط در طول داده آنها است که اعداد Double Integer بیتی و ۱۶ Integer بیتی میباشند. بر این اساس در اینجا از بررسی عملکرد مبدل مکمل دو اعداد Double Integer صرف نظر میشود.

۹-۱-۶) مبدل عدد Real مثبت به عدد Real منفى (NEG_R)

طول هر یک از دادههای ورودی و خروجی این مبدل Double Word و نتیجه عمل تبدیل این مبدل در خروجی حاص طول هر یک از دادههای ورودی است. در این تبدیل فقط بیت علامت آن تغییر می کند.

توضیح: مثال شکل(۶-۷) را می توان با سیمولاتور(PLCSIM) بررسی و فرمت حافظههای ورودی و خروجی مبدل را مشاهده کرد.



شکل(۶–۷)

در این مثال اگر به ورودی تابع Move عدد 12.0- داده شود فرمت حافظه ورودی تـابع NEG_R 0000 0000 0100 0000 0000 در این مثال اگر به ورودی تابع Move قروحی آن بصورت 0000 0000 0000 0000 0000 0000 فواهد شد.

توضيح: در تابع Move داده با هر فورمت (مثلا INT ($\pm n \pm 1$ DINT ($\pm n \pm 1$ DINT ($\pm n \pm 1$ DINT ($\pm n \pm 1$) و يا Move ($\pm n \pm 1$) در ورودی داده شود در خروجی نتايج با همان فورمت انتقال میيابد. در مثال شکل(- (- + 12.0 + 10.0 +

۲-۶)مبدلهای Real به T-۶)

مقدمه

در این بخش چهار نوع تابع برای تبدیل فرمت Real به فرمت Double Integer بررسی می شود. تفاوت این توابع در روش حذف بخش مقادیر کسری اعداد Real است.. این چهار تابع شامل: CEIL ،TRUNC ،ROUND و FLOOR هستند که در ادامه هر یک از آنها با یک مثالی بررسی می شوند.

-8-۲-۶) تابع Round:

با این تابع عدد حقیقی (با فرمت Floating point) به نزدیکترین عدد صحیح گرد شده و حاصل به یک عدد صحیح علامت دار ۳۲ بیتی(Double Integer) تبدیل می شود. اگر عدد حقیقی دقیقا در وسط دو عدد صحیح باشد (مثل N.5) نتیجه به عدد زوج گرد می شود. این تابع را می توان با طرح شکل(۸-۶) بررسی کرد. بطوریکه از روی شکل مشاهده می شود در این مثال عدد ۲/۴۹۹ به عدد ۳ و عدد ۳/۵۱ به ۴ و عدد ۲/۵ به ۴ گرد شدهاند.



چون در PLC دامنه تغییرات اعداد حقیقی بزرگتر از دامنه تغییرات عدد Double Integer (عددصحیح علامت دار ۳۲ بیتی) است، اگر حاصل این تبدیل خارج از مجموعه اعداد Double Integer (۲۱۴۷۴۸۳۶۴۷ - ۲۱۴۷۴۸۳۶۴۸-) قرار گیرد، خروجی سرریز کرده، و در این وضعیت خروجی ENO تابع صفر می شود. این موضوع را می توان با طرح شکل(۶-۹) بررسی کرد.

در دو شبکه زیر، به ورودی تابع شبکه پنج عدد ۳ /۲۱۴۷۴۸۳۶۴۷ و به ورودی شبکه شـش عـدد ۳ /۲۱۴۷۴۸۳۶۴۸ داده شـده لست، بطوریکه از روی شکل مشاهده میشود مقدار خروجی در شبکه شش سرریز(Overflow)) کرده و باعث صفر شدن خروجی ENO شده است.



TRUNC) تابع TRUNC:

با این تابع، بخش کسری عدد حقیقی با فرمت Floating point به صفر گرد و حاصل به یک عدد Double Integer (عدد صحیح علامتدار ۳۲ بیتی) تبدیل میشود. برای مثال عدد ۴/۵- به ۴- و ۴/۵+ به ۴+ تبدیل میشود. P-۲-۶) تابع CEIL :

با این تابع عدد حقیقی با فرمت Floating point به بالا گرد شده و حاصل به یک عدد Double Integer (عدد صحیح علامـتدار ۳۲ بیتی) تبدیل میشود. برای مثال عدد ۴/۵- به ۴- و ۴/۵+ به ۵+ گرد میشود.

FLOOR) تابع

با این تابع عدد حقیقی با فرمت Floating point به پائین گرد و حاصل به یـک عـدد Double Integer (عـدد صـحیح علامـت دار ۳۲ بیتی) تبدیل می شود. برای مثال عدد ۴/۵- به ۵- و ۴/۵+ به ۴+ گرد می شود.

توضيح:همانند تابع Round اگر حاصل هر یک از توابع CELL ،TRUNK و FLOOR خارج از مجموعه اعداد Double Integer توضيح:همانند تابع مربوطه اعداد Trunk و ۲۱۴۷۴۸۳۶۴۷ + ۲۱۴۷۴۸۳۶۴۸ - قرار گیرد، خروجی سرریز کرده، و در این وضعیت ENO تابع مربوطه صفر می شود.

۶-۳ بررسی بیتهای وضعیت'

با بیتهای وضعیت آمده در جدول زیر می توان تعدادی از وضعیت عملیات ریاضی و منطقی را تعین کرد. که شرح وضعیت هر یـک از آنها در کنار این جداول آمده است.

Status bits اعداد تعریف شده برای آن تابع قرار گیرد) بیت Over Flow فعال (یک) می شود. این نتیجه تا اجرای
-۱۱- ۱۵۷ ۱۳۰۰ عرف علی از ای ابغ کرد کیری ای کابع کرد کیری بیک ۱۳۵۷ ۲۰۰۰ علی ریک) شکی شوی ایک کلیب کا اجرای
- OS السسسان تابع ریاضی دیگردر برنامه، ثابت باقیمیماند.
ست OVer Flow بیت OVer Flow بخ ردهای برگی اوک (Stored OV) (مدیریت OV در بریت Stored OV) (مخبر
// OW اا
UOS
// UO II
/ BR ال
Unordered فعال میشود. این نتیجه تا اجرای تابع ریاضی بعدی در برنامه فعال باقی می ماند.

بیت BR⁴: هرگاه یک بلوکی که دارای شرط اجرا برای ورودی EN است، اجرا شود بیت Binary Result فعال می شود. این نتیجه تا اجرای بلوک بعدی، فعال باقی می ماند.

ويستحام ومشرقت المتراجع والماسي والمتراجع والماقي المستحد والمتراجع والمتراجع والمراجع	🖃 🔐 Status bits
بيكهاي وطعيت لنايج عمليات رياطي والمنطعي تسبب به صفر.	⊣। ==0
با این بیتهای وضعیت، میتوان وضعیت انباره ^ع (رجیسترمحل قرارگیری نتایج عملیات ریاضی و	
را نسبت به صفر بررسی کرد. این بررسیها شامل مساوی، بزر کشر و یا مساوی، کوچکتر و یا	
بزرگتر، کوچکتر و نامساوی صفر است. ^۷	
	•

¹ Status Bite

منطقي)

مساوى،

- [°] Over Flow
- ^{*} Store Over Flow
- [£] Unordered
- [°] Binary Result
- Accumulator

توضیح: در برنامهها میتوان هر یک از این بیتهای وضعیت را در دو منطق مختلف (بصورت کنتاکت باز و یا بسته) استفاده کرد.که در ادامه نحوه عملکرد هریک از این بیتهای وضعیت با یک مثال بررسی می شوند.

۶–۳–۱) مثال برای بررسی بیت OV – هرگاه در طرح شکل(۶–۱۰) در اجرای تابع ریاضی شبکه یک، Over Flow رخ دهد (یعنی نتیجه حاصل ضرب خارج از محدوده اعداد EON – تا ۳۲۷۶۷ واقع شود)، بیت وضعیت OV یک (و خروجی EON تابع ضرب صفر) شده در نتیجه خروجی شبکه دو (Q2.0) یک می شود



تمرین: برنامهای آماده کنید تا با آن برنامه، دو عدد Integer را از ورودیهـــای ۱۳۷۱ و ۱۳۷2 دریافت و حاصل ضرب آنها را به خروجی QW4 بصورت Integer منتقل کند و اگر در اجرای ایس برنامه Over Flow رخ دهـد و

شکل(۶–۱۰)

نتیجه این Over flow مربوط به عدد منفی باشد به خروجی عدد 32000- و اگر مربوط به عدد مثبت باشـد بـه خروجـی عـدد 32000 منتقل کند.

هرگاه در شبکه یک عمل ضرب به درستی انجام شود و در شبکه سه با فعال شدن IO.0 در اجرا تابع تفریـق Over flow رخ دهـد خروجیهای Q2.1 و Q2.2 فعال خواهند شد. هر دو مورد را بررسی کنید.



۶–۳–۳)مثال برای بررسی بیت OU – اگر در طرح شکل(۶–۱۲) به دو ورودی دو عدد

 $\begin{array}{c} ID0 = 0111 \hspace{.1in} 1111 \hspace{.1in} *000 \hspace{.1in} 0000 \hspace{.1in} 0000 \hspace{.1in} 0000 \hspace{.1in} 0000 \hspace{.1in} 0001 \\ ID4 = 0*11 \hspace{.1in} 1111 \hspace{.1in} 1000 \hspace{.1in} 0000 \hspace{.1in} 0000 \hspace{.1in} 0000 \hspace{.1in} 0000 \hspace{.1in} 0000 \hspace{.1in} 0000 \end{array}$

داده شود، به طوریکه بجای دو علامت * (I1.7 و I1.6) مقادیر 0 و یا 1 گذاشته شود. با این تست میتوان عملکرد بیتهای UO و OV را در توابع Real بررسی کرد.





مثال دیگر برای بررسی بیت OU: اگر به ورودی تابع طرح شکل (۶–۱۳)عدد بزرگتر از ۱ داده شود UO فعال می شود. این موضوع



۹-۳-۴) مثال برای بررسی وضعیت Accumulator نسبت به صفر – در CPU یک PLC نتایج عملیات ریاضی و منطقی در درون Accumulator قرار میگیرند، برای بررسی وضعیت Accumulator نسبت به صفر، طرح شکل(۶-۱۴) را بررسی کرده، تعین کنید در چه وضعیتی از Accumulator خروجیهای Q2.0 تا Q2.3 فعال میشوند.



شکل(۶–۱۴)

تمرین : برنامهای طرح کنید که تا در آن برنامه، ورودی IW0 با عدد W#16#00ff بصورت منطقیAND شده، اگر نتیجه غیر صفربود، نتیجه بصورت کد BCD به خروجی QW4 منتقل شود.

معرفی ماژول آنالوگ و سامانه ٔ حرارتی

پاسخ تمرین بخش ۶-۳-۱



¹ Plant

جلسه هفتم

هدف:

- معرفی برنامه نویسی ساختار یافته و معرفی بلوکهای FC ^۲،FB ^۱،FC و DB ^۵ و DB
 - بررسی نحوه ایجاد بلوک FC و نحوه فراخوانی آن در بلوکها دیگر
- بررسی نحوه ایجاد بلوک FB و بلوکDB اختصاصی⁵ برای آن و نحوه فراخوانی بلوک FB در بلوکها دیگر
 - بررسی نحوه ایجاد DB اشتراکی^۲ و نحوه استفاده از دادههای آن در بلوکها

مقدمه: برنامه نویسی ساختار یافته

در برنامه نویسی برای PLC، می توان برنامه یک پروژه را بر جسب محتوای آن تقسیم بندی کرده و هر قسمت از برنامه را در یک بلوک جداگانه نوشت. سپس بلوکها را بر اساس نیاز در برنامه بلوکهای دیگر فراخوانی کرد. بلوکهای SC و FB بلوکهای دیگر هستندکه توسط برنامه نویس ایجاد، برنامه ریزی و در بلوکهای دیگر فراخوانی می شوند. فراخوانی BBها در بلوکهای دیگر نیاز به DB اختصاصی است که دادههای مورد نیاز برنامه فراخوانی شده در آن قرار می گیرند. علاوه بر این دادههای این BDها در بخش-های دیگر از برنامه قابل استفاده هستند. بلوکهای SFC، بلوک BD اختصاصی ندارند دادههای حافظه های محلی این بلوکها فقط در موقع فراخوانی این بلوک معتبر بوده و در خارج از آن اعتباری ندارند

در نرم افزار Simatic Manager بلوکهائی به نامهای SFC و SFB وجود دارند که عمل کرد آنها شبیه کار FC و FB هستند این نوع بلوکها را طراح نرم افزار از قبل برای منظورهای خاص آماده کرده و در اختیار کاربران قرار داده است. برنامههای موجود در این بلوکها شامل برنامههای عمومی مورد نیاز در صنعت اتوماسیون و برنامههای مربوط به بکار گیری ماژولهای خاص سازنده PLC هستند.

برای برنامه ریزی PLC، بلوکهای دیگری به نام OB^۸ ها وجود دارند که از پیش توسط طراح نرم افزار بـرای منظورهـای خاصـی آماده شدهاند. محتوای این بلوکها توسط کاربر برنامه ریزی و توسط Operating System در زمانهای خاص فراخوانی میشـوند. در شکل(۲–۱) جایگاه فراخوانی همه بلوکها در برنامه نشان داده شده است.



شکل(۷–۱)

[']Function

- ['] Function Block
- [°] System Function
- ⁴ System Function Block
- ° Data Block
- ¹ Instance
- ^v Share
- [^] Organization Block

برای تقسیمبندی برنامهی یک پروژه، میتوان برنامههای عملیاتی که اجرای آنها تکـرار مـیشـوند، و همچنـین برنامـههـائی کـه غیرتکراری بوده ولی بر حسب نوع عمل کرد دستهبندی شده اند را در بلوکهای FC و FB مجزا نوشته و آنها را در برنامه بلـوک-های دیگر فراخوانی کرد.

برای عیبیابی برنامه طراحی شده و یا عیبیابی تجهیزات ابزار دقیق موجود در Plant ، می توان هر یک از بلوکهای FC و FB مربوطه را جداگانه پایش^{(۲} کرد.

در برنامه نویسی برای PLC میتوان دادههای یک برنامه را بر اساس مضامین مختلف گروه بندی کرده، و هر گروه از این داده را در یک DB اشتراکی^۲ بطورمجزا واردکرد. با ایجاد این نوع بلوکها، میتوان هر یک از آنها را در قسمتهای مختلف برنامه باز و از دادههای آنها در آن قسمت از برنامه استفاده کرد. همچنین در صورت نیاز میتوان، در موقع اجرای برنامه، دادههای این بلوکها را پایش کرد.

۲-۷) نحوه ایجاد یک بلوک FC ، و نحوه فراخوانی آن در بلوک دیگر

مقدمه

در اینجا میخواهیم در یک بلوک مناسب برنامهای نوشته شود، که آ ن برنامه یک عدد مثبت دو رقمی با کد BCD را از ورودی بلوک گرفته و مجذور آنرا بصورت کد BCD سه رقمی به خروجی بلوک منتقل کند، بطوریکه اگر نتیجه مجذور بیشتر از ۹۹۹ شد عدد ۹۹۹ را بجای نتیجه به خروجی منتقل، و وضعیت پیش آمده را توسط یک بیت داده در خروجی بلوک نشان دهد. بلوک مناسب برای این برنامه بلوک FC است که در زیر نحوه ایجاد آن، نحوه نوشتن برنامه در آن و نحوه فراخوانی آن در بلوک دیگر آمده است.

FC ايجاد بلوک). FC

- برای ایجاد یک FC از صفحه Simatic manager ، در پنجره پوشههای اشیائ پروژه ، پوشه بلوک انتخاب شده، سپس از مسیر نشان داده شده در شکل(۲-۲) گزینه Function انتخاب می شود.



شکل(۲-۲)

- با انتخاب گزینه Function، صفحه Properties- Function به شکل (۲–۳) باز می شود. در این صفحه می توان نام تابع را بصورت سیمبول(که در این مثال BCD_Square است) در محل مربوطه وارد و زبان برنامه نویسی مورد نظر را (که در اینجاب کرد. در پایان با انتخاب دکمه OK این صفحه بسته می شود.

Properties - Function		×		
General - Part 1 General -	Part 2 Calls Attributes			
Name:	FC1			
Symbolic Name:	BCD_Square			
Symbol Comment:				
Created in Language:	LAD			
Project path:				
Storage location of project:	E:\Siemens\Step7\s7proj\test_FC2			
.	Code	Interface		
Last modified:	05/02/2010 12:58:54	05/02/2010 12:58:54		
Comment:		<u>^</u>		
ОК		Cancel Help		

شکل(۷–۳)

– پس از بسته شدن صفحه Properties- Function، آیکن Function ایجاد شده(FC)، در پنجـره بلـوکـهـا بطوریکـه کـه در شکل(۲-۴) نشان داده شده است، ظاهر میشود. با دو بار کلیک راست کردن بر روی ایـن آیکـن (FC) ایجـاد شـده محـیط برنامه نویسی آن باز میشود.

🖹 test_FC E:\Siemens\Step7\s7proj\test_FC					
test_FC SIMATIC 300 Station SIMATIC 300 Station Gr S7 Program(1) Sources Blocks	🚵 System data	⊕ OB1			
		(m			

شکل(۷-۴)

۷-۱-۷) نوشتن برنامه در بلوکFC، برای نوشتن برنامه در بلوک FC لازم است متغیرهائی که در برنامه داخل بلوک به کار گرفته شوند در جدول شامل ورودی (IN)، کار گرفته شوند در جدول شامل ورودی (IN)، کار گرفته شوند در جدول شامل ورودی (IN)، خروجی (OUT)، ورودی - خروجی (IN)، حافظه موقت (TEMP) و متغیر برگشتی(RETURN) هستند. که در این متال از همه این متغیرها استفاده نمی شود.

متغیرهای استفاده شده در برنامه FC مورد مثال، شامل یک متغیر ورودی (بنام Input_Value) و دو متغیر خروجی (بنـام-هـایStatus_Overload ، Output_Value) و چهـار حافظـه مـوقتی (بنامهـایTemp_square ،Temp_int ،Temp_BCD) و چهـار حافظـه مـوقتی (بنامهـایTemp_overload ، Output_Value) و جهـار حافظـه مـوقتی (بنامهـایTemp_Overload ، Output_Value) و جهـار حافظـه مـوقتی (بنامهـایTemp_overload ، Output_Value) و جهـار حافظـه مـوقتی (بنامهـایTemp_overload ، Output_Value) و دو متغیر خروجـی (بنام

	Con	Contents Of: 'Environment\Interface\TEMP'				
🕀 Interface		Name	Data Type	Address	Comment	
Ē. ₽ IN	1	Temp_BCD	Word	0.0		
🖾 🖾 Input_value	12	Temp_Int	Int	2.0		
⊡ -⊡ OUT	12	temp_Squar	Int	4.0		
Dutput_value	12	Temp_Overl	Bool	6.0		
	12					
Temp_BCD_1						
🔤 Temp_Int						
🔤 temp_Squar						
🖾 Temp_Overload						
E 🕂 - 💶 P RETURN						

شکل(۷–۵)

پس از آماده شدن جدول متغیرهای مورد نیاز، میتوان برنامه آنرا در آن نوشت. این برنامه که توضیح آن در مقدمه این بخش آمده در شکل(۷-۶) ارائه شده است. پس از نوشتن برنامه FC، لازم است این برنامه save و سپس به PLC منتقـل -شود.

توضیح: متغیرهای TEMP حافظههای موقتی هستند که فقط در داخل بلوک فراخوانی شده می توان از داده های آن ها استفاده کرد و در بیرون از بلوک داده های آن ها اعتباری ندارند.



شکل(۷-۶)

توضیح: در برنامه بلوک FC می توان بجای سیمبول از آدرسهای حافظههای TEMP نیز استفاده کرد. برای این کار از حرف L بصورت پیشوند در جلوی آدرسهای مربوطه (که در جدول متغیرهای بلوک آمدهاند) استفاده می شود. مثلا می توان بجای Temp_BCD از آذرس متناظر آن LW0 و بجای Temp_Over Load از آدرس متناظر آن L6.0 استفاده کرد.

۷–۱–۳)فراخوانی بلوک FC، برای اینکه برنامه FC ایجاد شده در برنامه بلوک دیگرکه در این مثال بلوک OBI است فراخوانی شود، لازم است ابتدا از پنجره بلوکها، در صفحه Simatic Manager ، بلوک OBI باز و سپس FC آماده شده در آن فراخوانی شود. در این مثال میخواهیم FCI ایجاد شده یک مرتبه در OBI فراخوانی شود. بازی مثال میخواهیم FCI ایجاد شده یک مرتبه در OBI فراخوانی شود. بازی مثال میخواهیم FCI ایجاد شده یک مرتبه در OBI فراخوانی شود. بازی مثال میخواهیم FCI ایجاد شده یک مرتبه در OBI فراخوانی شود. بازی مثال میخواهیم FCI ایجاد شده یک مرتبه در OBI فراخوانی شود. بازی مثال میخواهیم FCI ایجاد شده یک مرتبه در OBI فراخوانی شود. بازی فراخوانی این مثال میخواهیم FCI ایجاد شده یک مرتبه در OBI فراخوانی است. برای فراخوانی این بلوک در بلوک ICI ایجاد شده، در پنجره Network ما منع من مرتبه در آن با دسترسی است. برای فراخوانی این بلوک در بلوک ICI این مثال بلوک ICI ایند المانهای دیگر در Network ها نصب کرد. در ICI می میتوان آن را مانند المانهای دیگر در Network ها نصب کرد. برای مثال بلوک ICI ایجاد شده، بصورت نشان داده شده در شکل(۷–۷) در ICI به کار گرفته می شود. پر ایزامه بلوک ICI این را مانند المانهای دیگر در Network ها نصب کرد. بر اعان بلوک ICI ایجاد شده، بصورت نشان داده شده در شکل(۷–۷) در ICI به کار گرفته می شود. به می شود. بی برنامه بلوک ICI این برنامه PL منتقل شود. در موقع انتقال این بلوک VI ما ست بلوک ICI ما منتقل شود. در موقع انتقال این بلوک لازم است بلوک ICI مربوط به خطا (در روی ما ژول ICI بسورت قرمز رنگ) روشن می شود. اگر OLI می می شود. در خان ایتقال بلوکها خطائی رخ نمی دهد.



پس از انجام تمام کارهای فوق برنامه را آزمایش و نتایج آنرا بررسی کنید.

د) در برنامه OB1 یک بار دیگر بلوک FC1 را با آدرسهای IB1 ، QW2 و Q2.7 فراخوانی کرده عملکرد آنرا پس از انتقال به PLC بررسی کنید

۲-۷) نحوه ایجاد یک **FB و نحوه فراخوانی آن در بلوک دیگر**

مقدمه

در اینجا میخواهیم در یک بلوک مناسب برنامهای نوشته شود که تا آن برنامه دو داده بیتی را از ورودی بلوک دریافت و یک داده بطول Word را به خروجی منتقل کند، بطوریکه به ازاء هر لبه بالا رونده یکی از دو ورودهای بیتی، ۵ واحد به خروجی Word (که فرمت داده آن بصورت کد BCD خواهد بود) اضافه و با یک شدن ورودی بیتی دوم این بلوک مقدار خروجی Word آن صفر شود. بلوک مناسب برای این برنامه FB است که در زیر نحوه ایجاد آن، نحوه نوشتن برنامه در آن و نحوه فراخوانی آن در بلوک دیگر آمده است.

Function Block (FB) ایجاد (۱-۲-۷

- برای ایجاد یک FB در صفحه Simatic manager ، در پنجره پوشههای اشیائ پروژه، ابتدا پوشه بلوک انتخاب شده،. سپس از مسیرنشان داده شده در شکل(۲-۹) گزینه Function Block انتخاب می شود.



با انتخاب گزینه Function Block، صفحه Properties- Function Block که در شکل (۲-۱۰) نشان داده شده است، باز می شود. در این صفحه می توان نام بلوک را بصورت سیمبول (که در اینجا Adder_step5 است) در محل مربوطه وارد و زبان برنامه نویسی مورد نظر را(که در اینجا Ladder است) انتخاب کرد. در پایان با انتخاب دکمه OK این صفحه بسته خواهد شد

Properties - Function	Block	
General - Part 1 General	- Part 2 Calls Attributes	,
Name:	FB1	Multiple Instance Capability
Symbolic Name:	Adder_step5	
Symbol Comment:		
Created in Language:	LAD	
Project Path:	test_FB2\SIMATIC 300(1)\CP	U 314\S7 Program(1)\Blocks\FB1
Storage location of project:	E:\Siemens\Step7\s7proj\test	_FB2
Date created:	Code 17/01/2010 8:19:39 AM	Interface
Last modified:	03/02/2010 6:03:46 PM	23/01/2010 1:03:53 PM
Comment:		
ОК		Cancel Help

شکل (۷–۱۰)

با بسته شدن صفحه فوق، آیکن Function Block ایجاد شده (FB) در صفحه Simatic manager در پنجره بلوکها ظاهر می شود. FBI ایجاد شده در شکل(۲–۱۱) در پنجره بلوکها نشان داده شده است. . با دو بار کلیک راست کردن بر روی آیکن (FB)، محیط برنام نویسیمربوط به آن باز می شود.



۲–۲–۷) نوشتن برنامه در FB: برای نوشتن برنامه در FB، ابتدا لازم است متغیرهای مورد نیاز در برنامه ی این را بلوک در جدول بالای محیط برنامه نویسی این بلوک وارد و مقادیر اولیه انها تنظیم شوند. متغییرهای موجود در جدول شامل ورودی (IN)، خروجی (OUT)، خروجی (ITM)، حافظه نگهدارنده داده (Stat) و حافظه موقت (TEMP) هستند. که در این مثال از همه این متغیرها استفاده نمی شود.

متغیرهای مورد نیاز در برنامه توضیح داده شده در مقدمه این بخش که باید وارد بلوک شوند در شکل(۷–۱۲) نشان داه شده اند. این متغیرها شامل، دو ورودی بیتی (بنامهای add و Reset) یک خروجی از نوع Word (بنام sumation_out) و دو حافظه Stat که یکی برای داده Integer (بنام sumation)و دیگری برای داده بیتی(بنام P_edge) هستند.

	Contents Of: 'Environment\Interface\STAT'					
🕀 Interface		Name	Data Type	Address	Initial Value	Excl
🚊 💶 IN	1	sumation	Int	4.0	0	
add	13	P_edge	Bool	6.0	FALSE	
🖂 T 🖻 Reset	12					
i⊒ ≪ ⊒ • OUT			1			
🔲 🖾 Sumation_out						
E STAT						
····`` sumation						
P_edge						
E TEMP						

شکل(۷–۱۲)

پس از آماده کردن جدول متغیرها برای بلوک FB، برنامه مربطه در آن نوشته می شود. برنامـه مـورد نظـر در ایـن مثـال در شکل(۷–۱۳) ارائه شده است،



پس از آماده نمودن برنامه FB، لازم است برنامه Save و به PLC منتقل شود.

توضیح: در برنامه FB می توان بجای سیمبول متغیرهای Stat و Temp از آدرسهای متناظر آنها که در جدول مربوطه آمده-اند استفاده کرد. برای این کار، از حرف L بصورت پیشوند در جلوی آدرسهای مربوط ه ک ه در جدول متغییره آمدهاند، استفاده می شوند. بطور مثال می توان بجای Sumation از آدرس LW4 و بجای P-edge از آدرس L6.0 استفاده کرد.

۲-۲-۳) ایجاد DB اختصاصی، برای فراخوانی یک FB در بلوکهای دیگر نیاز به دیتا بلوک است. در مثال ارائه شده در این بخش میخواهیم FB ایجاد شده دو مرتبه به منظور اجرای دو عملیات با دو دیتا مختلف فراخوانی شود. برای برآورده شدن این منظور لازم است دو دیتا بلوک مجزا با روند زیر ایجاد شود.

الف – برای ایجاد هریک از DB ها، ابتدا در صفحه Simatic manager ، در پنجره پوشههای اشیائ پروژه، پوشه بلوک انتخاب شده سپس از مسیر نشان داده شده در شکل(۲–۱۴) گزینه Data Block انتخاب می شود.



شکل(۷–۱۴)

ب- با انتخاب گزینه فوق صفحه Properties Data Block به صورت نشان داده شده در شکل(۷-۱۵) باز می شود. در این
 صفحه می توان نام بلوک را بصورت سیمبول(که در اینجا Adder1 است) در محل مربوطه و نوع دیتا بلوک را Instance که
 در این مثال مختص FB1 است، انتخاب کرد. در پایان با انتخاب دکمه OK این صفحه بسته خواهد شد.

Properties - Da	ta Block 🛛 🔀					
General - Part 1 General	General - Part 1 General - Part 2 Calls Attributes					
Name and type:	DB1 Instance DB 💌 FB1 💌					
Symbolic Name:	Adder1					
Symbol Comment:						
Created in Language:	DB					
Project path:						
Storage location of project:	C:\Program Files\Siemens\Step7\s7proj\S7_Pro1					
	Code Interface					
Date created: Last modified:	03/20/2005 05:49:50 PM 03/20/2005 05:49:50 PM 03/20/2005 05:49:50 PM					
Comment:						
ОК	Cancel Help					

شکل(۷–۱۵)

با بسته شدن صفحه فوق، آیکنData Block ایجاد شده بصورتDB1 در پنجره بلوکها در صفحه Simatic manager ظاهر می شود که عدد یک شماره دیتا بلوک ایجاد شده است.

با روند فوق دیتا بلوک دیگری با نا م سیمبولیک Adder2 برای پروژه آماده شود. در پایان لازم است هر دو دیتا بلوک ایجاد شده، از پوشه بلوکها از صفحه Simatic Manager انتخاب و به PLC منتقل شود. . **توجه شود** انتقال ایـن نـوع بلـوکهـا، باید حتماً همزمان یا قبل از FB مربوطه، به PLC منتقل شوند. در غیر این صورت PLC دچار خطا شده و متوقف می شود.

توضیح : اگر لازم شود مقادیر دادههای مربوط به دیتا بلوکها تغییر یابند، برای اینکار می توان هر یک از این دیتا بلوکها را از پوشه بلوکها (در صفحه Simatic Manager)باز و تغییرات لازم را در ستون Actual value جدول آنها اعمال کرد. برای فعال کردن ستون Actual value ، باید از منوی View گزینه Data View انتخاب شود.

در این مثال میخواهیم برای متغیر sumation در DB1 مقدار 3 و در DB2 مقدار 1 اعمال شود. در شکل (۷–۱۶) بلوک باز شده DB1 نشان داده شده است. توجه شود پس از هر تغییرات در یک دیتا بلوک، لازم است دیتا بلوک تغییر داده شده Save و مجددا به PLC منتقل شود

🗷 DB Param - DB1								
Data blo	Data block Edit PLC Debug View Window Help							
) 😅 🔓	🖙 🖫 🛃 🗠 🖙 🕹 🖻 💼 !« »! 🏜 🏜 & 🕅 📢							
🗷 DB1	📧 DB1 test_FB\SIMATIC 300 Station\CPU 315-2 DP							
	Address	Declaration	Name	Туре	Initial value	Actual value		
1	0.0	in	add	BOOL	FALSE	FALSE		
2	0.1	in	Reset	BOOL	FALSE	FALSE		
3	2.0	out	sumat	WORD	W#16#0	W#16#0		
4	4.0	stat	sumation	INT	0	þ		
5	6.0	stat	P_edge	BOOL	FALSE	FALSE		
<						>		

Simatic ه) فراخوانی بلوک FB، برای اینکه برنامه FB ایجاد شده در برنامه بلوک دیگر فراخوانی شود، ابتدا در صفحه Simatic از پنجره بلوکها، بلوک مورد نظر باز و سپس FB ایجاد شده در آن فراخوانی می شود. در این مثال می-خواهیم FB1 ایجاد شده دو مرتبه در OB1 فراخوانی شود. برای این کار لمان FB1 ایجاد شده که در پنجره خواهیم FB1 در زیر گروه FB قابل دسترس است، در دو Network جداگانه در بلوک OB1 نصب می شوند. در شکل(۲–۱۷) برنامه آماده شده در بلوک IB1 که در آن بلوک دو مرتبه FB1 فراخوانی شده ارائه شده است. این فراخوانی در شبکه یک بهمراه DB1 و در شبکه دو بهمراه DB2 انجام شده است.



در نهایت بلوک OB1 آماده شده Save و سپس به PLC منتقل می شود.

توضیح، حتما لازم است همزمان و یا قبل از انتقال OB1 به PLC، بلوکهای FB و DB بکار گرفته در برنامه OB1 به PLC منتقل شوند. برای انتقال همه بلوکها همه بلوکها می توان در صفحه Simatic Manager از پوشه بلوکها همه بلوکها را همزمان انتخاب و به PLC منتقل کرد. اگر PLC در حال Stop باشد با تقدم و یاخر در انتقال بلوکهای خطائی رخ نمی دهد. پس از انجام تمام مراحل فوق پروژه را آزمایش و در صورت وجود اشکال احتمالی آنرا رفع و نتایج پروژه را بررسی کنید.

۲-۲-۷) بررسی استفاده از دادههای دیتا بلوکها در خارج از FBها

برای بررسی استفاده از دادههای دیتا بلوک در خارج FBها دو Network جدید بر طبق شکل(۷–۱۷) به برنامه اضافه کرده سپس اجرای این برنامه را بررسی و به سئوال زیر پاسخ دهید.


سئوال: خروجیهایQw6 و QW0 معادل کدام متغیرهای DB بلوکهای adder1 و adder2 هستند و چرا؟

توضیح۱: متغیرهای یک DB را میتوان بصورت سیمبول و یا غیر سیمبول در خارج از بلوک آدرس دهی کرد. **توجه شـود** در صورتی میتوان از آدرس دهیبصورت سیمبول استفاده کرد که DB مربوط هم نـام سـیمبول داشـته باشـد. بـرای مثـال بطوریکه در برنامه OB1 در شکل(۷-۱۷) مشـاهده مـیشـود در شـبکه ۲ بجـای آدرس DB1.DBW4 از شـکل آدرس.دهـی سیمبولیک آن، یعنی بجای DB1 از سیمبول adder1 و بجای DBW4 از سیمبول sumation استفاده شده است.

توضیح ۲: آدرسهای شبکههای دو و چهار شکل (۲–۱۷) را مانند شکل (۲–۱۸) تغییر داده و کار برنامه را مجددا بررسی کنید. بطوریکه مشاهده می شود، ورودیهای بلوکهای MOVE در شبکههای ۲ و ۴ این شکل مانند هم هستند، ولی مقادیر خروجیهای این دو شبکه باهم متفاوت هستند. دلیل آن این است که هر موقع FB فراخوانی شود DB مربوطه به آن باز می-شود. با باز شدن DB مربوطه میتوان بدون ذکر آدرس BD بلوک باز شده، از دیتای آن (در بلوک فراخوانی شده BG) استفاده کرد، تا زمانی که DB جدیدی باز نشود (در اینجا یعنی تا FB دیگری اجرا نشود) میتوان از دادههای DB باز شده استفاده کرد.



۳-۷) ایجاد (Data Block (DB اشتراکی و نحوه بکارگیری آن در بلوکها:

الف) ایجاد DB اشتراکی، برای ایجاد DB از نوع Shared، همانند روند ایجاد DB از نوع Instance که در بخش ۷-۲-۳ آمده است عمل میشود، با این تفاوت که در صفحه Shared معافت properties Data block عنوان Shared انتخاب میشود. در اینجا، به منظور تمرین استفاده از Data Block در برنامه، دو دیتا بلوک ایجاد میشوند. پس از ایجاد دو دیتا بلوک، برای وارد کردن دادهها در آنها، از صفحه Simatic Manager از پنجره بلوکها هر یک از آنها (بصورت Declaration View) باز و متغیرهای مورد نیاز برای اجرای برنامه در جدول آنها وارد میشوند. یکی از ستونهای جدول دیتا بلوک مربوط به نوع داده (Data Type) است. در شکل (۷–۱۹) مسیر انتخاب نوع دادهها برای این ستون نشان داده شده است.

🔣 LAD/STL/	FBD - [DB1 test	_34_DB	\SIMATIC 300 Station\CPU314(1)]
🕞 File Edit	Insert PLC Debug	View O	Options Window Help
	Object Block Template) }	
Address b	Data Type	•	Elementary Types BOOL
0.0	Declaration Line	•	Complex Types BYTE
+0.0	Network	Ctrl+R	Parameter Types WORD placeho
+2.0	Symbol	Ctrl+J	FALSE DWORD
+4.0	Program Elements	Ctrl+G	0#16#25 DINT
+6.0	Timer_Value	SSTIME	SST#10MS BEAL
+8.0	remain_Counter	WORD	W#16#0 S5TIME
+10.0	remain_timer	WORD	0#16#0 TIME
+12.0	IN_OUT	BYTE	B#16#5 DATE
=14.0		END_STR	RUCT TIME_OF_DAY

شکل(۷–۱۹)

درشکلهای (۲–۲۰) و (۲–۲۱) دیتاهای وارد شده در جدول دو دیتا بلوک نمایش داه شدهاند. در این شکلها، DBl بصورت Declaration View و DB2 بصورت Data View نمایش داده شدهاند. در نمایش Declaration View میتوان متغیرها را وارد جدول کرده و در نمایش Data View میتوان مقادیر Actual Value آنها را تغییر داد. این دو View از منوی View قابل انتخاب هستند.

پس از آماده کردن هر یک از دیتا بلوکها آنها را save و به PLC منتقل کنید.

<pre>Kad/STL/FBD - [DB1 test_34_DB\SIMATIC 300 Station\CPU314(1)]</pre>									
🕞 File Edit Insert PLC Debug View Options Window Help									
Address Name Type Initial value Comment									
0.0		STRUCT							
+0.0	DB_VAR	INT	0	Temporary placeho					
+2.0	Set_Counter	BOOL	FALSE						
+4.0	offset_counter	WORD	W#16#25						
+6.0	Timer_Value	S5TIME	S5T#10MS						
+8.0	remain_Counter	WORD	W#16#O						
+10.0	remain_timer	WORD	W#16#0						
+12.0	IN_OUT	BYTE	B#16#5						
=14.0		END_STRUCT							

شکل(۲۰-۷ Meclaration view)

	🔣 LAD/STL	./FBD - [DB2 tes	t_34_DB\SIMATIC 3	00 Station\CPU	314(1)]	
	🕞 File Edit	: Insert PLC Debu	g View Options Wind	dow Help		
ļ					(<u>>!</u> 🗖 🖪	?
	Address	Name	Туре	Initial value	Actual value	Солта
	0.0	DB_VAR	INT	0	0	Tempo
	2.0	Set_Counter	BOOL	FALSE	FALSE	
	4.0	Offset_counter	WORD	W#16#50	W#16#40	
	6.0	Time_value	SSTIME	S5T#30MS	S5T#2S	
	8.0	remain_Count	WORD	W#16#0	W#16#0	
	10.0	remain_time	WORD	W#16#0	W#16#0	

شکل(TI-۷)Mata View

ب) استفاده از DB اشتراکی در برنامه :

در برنامه ارائه شده در شکل(۲-۲۲) نحوه استفاده از دو دیتا بلوک(DB1 و DB2) آمده است، این برنامه را آماده کرده، سپس آنرا save و به PLC منتقل کنید. پس از انتقال برنامه عملکرد آنرا بررسی و به سئوالهای آمده در زیر پاسخ دهید؟





سئوال ۱: دیتای DB1 در کدام شبکههای این برنامه بکار گرفته شده است؟ سئوال ۲: دیتای DB2 در کدام شبکههای این برنامه بکار گرفته شده است؟ سئوال ۳: مقادیر Time Value زمانسنجهای To و T1 در این برنامه چقدر هستند؟ سئوال ۴: مقادیر Pre Value شمارندههای C0 و C1 در این برنامه چقدر هستند؟ سئوال ۵: از صفحه Pre Value شمارندههای D1 بلوک را باز کرده، سپس با یک کردن ورودی I0.3 و با تغییر مقادیر IB1 بررسی کنید کدام یک از دادهها مقدار آن تغییر میکند.

جلسه هشتم

هدف:

- استفاده از المانهای کنترل برنامه شامل Return ، Call و Master Control Relay
- استفاده از المانهای مربوط به پرش در برنامه شامل Not Jump ،Jump و Label
- نحوه ایجاد جدول متغیر^۲ و استفاده از آن برای اصلاح^۳ و پایش متغیرهای یک برنامه
 - نحوه Force متغیرهای ورودی خروجی PLC
 - نحوه استفاده از کلاک ژنراتور داخلی CPU

۸-۱) بررسی المانهای کنترل برنامه

مقدمه

در جدول زیر مجموعهای از المانهای مربوط به کنترل برنامه نشان داده شده، که توضیح و کاربرد هر یک از این المانها در ادامه این بخش آمده است.

	🖃 📅 Program control
برای فراخوانی FC و SFC های بدون ورودی خروجی تعریف شده	
Master Control Relay On	
Master Control Relay Off Master Control Relay Activate	
Master Control Relay Deactivate	
برای برگشت اجرای برنامه از بلوک فراخوانی شده	
	(RET)
	🗄 🧱 Shift/Rotate

۸–۱–۱)) بررسی المان (RET) Return (RET: در صورت فعال شدن این المان در یکی از شبکههای یک بلوک، ادامه اجرای برنامه در آن بلوک پایان یافته و اجرای برنامه به محل بلوکی که بلوک جاری از آنجا فراخوانی شده است برمی گردد . اگر این المان در بلوک OB1 فعال شود ادامه اجرای برنامه، به اولین شبکه این بلوک برمی گردد.

مثال برای کاربرد RET: برنامه شکل(۸-۱) دارای یک FC است. هرگاه در این FC کوئیل RET شبکه ۲ فعال شود، ادامه برنامه این بلوک(شبکه ۳) اجرا نشده و اجرای برنامه از آن بلوک خارج و به بلوک OB1 برمیگردد.

[°] Program Control [°] Variable Table ³ Modify



۸-۱-۸ بررسی المانCALL: با این المان میتوان بلوک های FC و SFC که دارای ورودی و خروجی تعریف شده نیستند را فراخوانی میتواند بدون شرط یا با شرط انجام شود.

مثال برای کاربرد CALL: در طرح مثال قبل در بلوک OB1 میتوان بلوک FC1 را که دارای ورودی و خروجی تعریف شده نیست بصورت شکل(۸-۲) فراخوانی کرد.

OB1	2	"Main	Program	Sweep	(Cycle)"	
Netw	orl	1: Ti	tle:			
						FC1
						(CADD)



Master Control Relay بررسى المانهاى (٣-١-٨)

با جفت المان MCRA و MCRD میتوان، انتقال نتایج اجرای بخشی از برنامه را با کمک جفت المانهای >MCR و MCRک میتوان، انتقال نتایج اجرای بخشی از برنامه را با کمک جفت المانهای >MCR ممکن و یا ناممکن ساخت. یعنی در صورت فعال بودن آن انتقال غیر ممکن شده، و بجای نتایج اجرای برنامه به آدرسهای مقصد صفر منتقل میشود. همچنین در این وضعیت تغییرات ورودیهای R و S روی فلیب فلاپهای مربوطه بی تاثیر خواهند شد.

توجه شود در برنامه المان >MCR بصورت شرطی اجرا می شود و المان های <MCR ، MCR و MCRD برای اجرا، شرط نمی پذیرند.

مثال برای کاربرد المانهای MCR: در برنامه شکل(۸–۳) مثالی برای بکار گیری المانهای MCR آمده است. در این برنامه اگر ورودی I1.0 صفر شود در ناحیه یک انتقال داده ها صفر و تغیرات ورویهای فلیب فلاب بر روی آن بی تاثیر خواهند بود و همچنین اگر ورودی I1.0 یک و ورودی I1.1 صفر شود در ناحیه دو بجای انتقال دادهها صفرمنتقل می شوند. اگر هردو ورودی I1.0 و I1.1 یک شوند نتایچ برنامه بصورت معمول اجرا خواهد شد.



شکل(۸–۳)

۸-۲) پرش در برنامه (JUMP):

مقدمه

المانهای نشان داده شد در سمت راست شکل (۸–۴) برای ایجاد پرش در برنامه است که شرح عملکرد هریک از آنها در سمت چپ شکل آمده است..

مثال برای کار برد المانهای Jumps: در برنامه شکل(۸–۵) اگر ورودی I1.0 یک شود شبکه ۳ و ۴ اجرا نخواهند شد و اگر ورودی I1.1 صفر باشد شبکه شش اجرا نخواهد. شد. توجه شود ، در این نرمافزار عمل پرش فقط از بالا به پائین قابل اجرا است.



شکل(۸–۵)

۸-۳) نحوه ایجاد بلوک جدول متغیر(Variable Table) و استفاده از آنها.

مقدمه

در برنامههای بزرگ، *در* بعضی از موارد، نیاز است، از مجموعه متغیرهای یک برنامه، تعدادی از آنها همزمان پایش شده و در صورت نیاز بعضی از آنها اصلاح شوند.

برای بررسی این موضوع ابتدا برنامه شکل(۸–۶) به همراه جدول سیمبول های آن آماده شده، سپس عملکرد آن بررسی



پس از آماده و بررسی کردن کار برنامه فوق یک بلوک جدول متغیرها برطبق مراحل آمده در زیر ایجاد و متغیرها مورد نیاز برای پایش و دستکاری را در آن وارد کنید آنگاه برای پایش و دستکاری متغیرها، جدول آماده کرده را بکار بگیرید.

نحوه ایجاد بلوک جدول متغیرها: برای ایجاد بلوک جدول متغیرها به ترتیب زیر عمل می شود:

الف) در صفحه Simatic Manager، پس از انتخاب پوشه بلوک در پنجـره جسـتجوگر، از مسیرنشـان داده شـده در شـکل(۸-۷) گزینه Variable Table انتخاب میشود.



ب) با این انتخاب، صفحه Properties Variable Table نشان داده شده در سمت چپ شکل(λ-۸) باز می شود،

Properties - Variable	Table		\mathbf{X}	
General - Part 1 General	- Part 2 Attributes			
Name:	VAT2			
Symbolic Name:	VAT_1			🖹 test_83 E:\siemens\s7proj\test_83
Symbol Comment:				
Project path:			1	Erer S7 Program(1)
Storage location of project:	E:\Siemens\Step7\s7proj\For	rce_Te	1	
Date created:	Code 07/12/2008 9:45:43 AM	Interface		
Last modified:	07/12/2008 9:45:43 AM	07/12/2008 9:45:43 AM		
Comment:				
ОК		Cancel Help		
		شکل(۸–۸)		

با تائید کردن محتوای این صفحه توسط دکمه OK، یک بلوک جدول متغیر ایجاد می شود که آیکون آن در پنجره بلوکها در صفحه Simatic Manager آشکار میشود. این آیکون در سمت راست شکل(۸-۸) در پنجره بلوکها مشاهده می شود.

نحوه استفاده از بلوک جدول متغیرها: برای کار کردن با بلوک جدول متغیرها، ابتدا با دو بار راست کلیک کردن آیکون جـدول متغیرها در پنجره بلوک ها آنرا باز و متغیرهای مورد نظر مانند مثال شکل(۸-۹) در آن وارد میشوند.

醫	Va	r - VAT_	1						
Table Edit Insert PLC Variable View Options Window Help									
82	V	AT_1	@test_84\SIMATI	C 300 Statio	n\CPU 315-2 [P(1)\S7 Prog.			
	1	Address	Symbol	Display format	Status value	Modify value			
1		1 0.3	"Set_Count"	BOOL	false	true			
2 M 0.0 "EN_Move" BOOL true true									
3 MW 10 "Count_Val" HEX W#16#0000									
4		QW 4	"output"	HEX	W#16#0000	W#16#0099			
شکا (۹–۸)									

در روی میله ابزار جدول متغیرها ابزارهای خاصی وجود دارند که در جدول زیر شرح عملکرد هر یک از این ابزارهای آمده است.



¹ Monitoring

² Modifying

با انتخاب ابزار مربوط به **تنظیم پارامترهای پایش و دستکاری متغیرها** صفحه شکل(۸–۸) باز می شود. در این صفحه می توان عمل پایش و اصلاح در چه لحظهای از اجرای برنامه صورت پذیرد تنظیم می شود. همچنین در این صفحه، از اینکه پایش و یا اصلاح در هر سیکل و یا (با هر بار انتخاب ابزار مربوطه) یک بار انجام شود تنظیم می شود.

Trigger	X
Trigger Point for Monitoring Beginning of scan cycle C End of scan cycle C Transition to STOP	Trigger Condition for Monitoring C Once C Every cycle
Trigger Point for Modifying Beginning of scan cycle C End of scan cycle C Transition to STOP	Trigger Condition for Modifying
ОК	Cancel Help

شکل(۸–۱۰)

Force (۴-۸ کردن متغییرها ورودی و خروجی

مقدمه

در بعضی از موارد برای اجرای یک برنامه PLC (بطور مثال در زمان تعمیرات) لازم میشود بعضی از متغییرها: مانند **ورودی** و **خروجی** برنامه بصورت اجباری به یک مقدار مشخص تثبیت شوند. برای این کار بصورت زیر عمل میشود.

نحوه باز کردن جدولForce : برای باز کردن جدول Force، در صفحه بلوک جدول متغیرها از منوی Variable (یا در صفحه Simatic Manager از منوی PLC) بطوری که در شکل(۱۱–۸) نشان داده شده است گزینه Display Force Value (یا در انتخاب می شود.

توجه کنید برای این انتخاب نباید جدول متغییرها به PLC ارتباط تبادل اطلاعات داشته باشد. (برای قطع این ارتباط، از منوی PLC گزینه Disconnect انتخاب، و برای اتصال مجدد آن از همین منو گزینه Connect to انتخاب می شود.)



شکل(۸–۱۱)

- با این انتخاب، صفحه Display Force Value که مانند جدول نشان داده شده در شکل(۸-۱۲) است باز می شود. در این جدول متغیرهائی که می بایستی اجبارا به یک مقدار مشخص تثبیت شوند، وارد می شوند این متغییرها برای PLC سری 300 فقط شامل وروی، خروجی (بصورت بیت، بایت، Word و Double Word) هستند. **لازم است توجه شود**، در این نوع PLC ها مقادیر حافظه را نمی توان Force کرد.

نحوه Force کردن: پس از وارد کردن متغیرها در جدول و تعیین مقادیر مورد لـزوم بـرای آنهـا، از منـوی Variable ایـن جدول گزینه Force انتخاب میشود. با این انتخاب متغییرهای وارد شده در جدول بـه مقـادیر تعـین شـده، در ورودی خروجـی PLC تثبیت میشوند. در این شرایط چراغ Force روی ماژول PLC به رنگ زرد روشن می شود.

Stop نحوه حذف Force : برای حذف متغیرهای تثبیت شده در PLC، در صفحه جدول متغییرها از منوی Variable گزینه Forcing انتخاب می شود.

👪 Var - Force Values : test_83	SIMATIC 30	0 Station\CPU	315-2 DP(1)\S	7 Program(1)				
Table Edit Insert PLC Variable V	iew Options	Window Help						
▰▯▰◨◓▯◧▫▫××▫▯▯№ ഈഈഈ™™™								
VAT_1 @test_83\SIMATIC 3()0 Station\Cl	PU 315-2 DP(1)\S7 Progra					
Address Symbol	Display format	Status value	Modify value					
I 0.1 "res"	BOOL	false	//true					
🖁 Force Values : test_83\SIMATIC 300 Station\CPU 315-2 DP(1)\S7 🌖 🔳 🔲 🔀								
🖌 🖌 Address Symbol Display forma	t Force value							
F I 0.0 "set" BOOL	true							

شکل(۸–۱۲)

۵-۸) نحوه استفاده از پالس ژنراتورهای داخلی CPU:

مقدمه

در ماژول CPU هشت بیت پالس ژنراتور وجود دارد که زمان پریود و فرکانس هر بیت آن در جـدول زیـر نشـان داده شـده است. برای استفاده از این پالس ژنراتورها لازم است برای هشت بیت آن یک بایت آدرس اختصاص داده شود. با اختصـاص یـافتن ایـن یک بایت آدرس میتوان در برنامه از هر بیت آن بعنوان یک پالس ژنراتور با فرکانس امده در جدول زیر استفاده کرد.

7	6	5	4	3	2	1	0	بيت
2	1.6	1	0.8	0.5	0.4	0.2	0.1	زمان پريود
0.5	0.625	1	1.25	2	2.5	5	10	فركانس

نحوه تنظِم آدرس اختصاصی برای پالس ژنراتورها

برای تنظیم یک بایت آدرس اختصاصی برای هشت پالس ژنراتور داخلی CPU ، ابتدا در برنامه پیکربندی سخت افزار، با راست کلیک کردن بر روی ردیف CPU ، از منوئی که باز خواهد شد گزینه Object properties انتخاب میشود (یا اینکه ابتدا در برنامه پیکربندی سخت افزار ردیف مربوط به CPU انتخاب شده و سپس از منوی Edit گزینه Object properties انتخاب میشود)، با این انتخاب مجموعه صفحههای شکل(۸–۱۳) باز میشوند. در قسمت پائین صفحه با سر برگ Cycle/Clock Memory ، گزینه Clock Memory انتخاب میشود)، با این انتخاب مجموعه صفحههای شکل(۸–۱۳) باز میشوند. در قسمت پائین صفحه با سر برگ Cycle/Clock Memory ، گزینه Clock Memory ، انتخاب و سپس در جلوی عنوانMemory Byte شماره آدرس یک بایت اختصاصی برای پالس ژنراتورها (مثلا ۱۰۰) وارد میشود. با تائید این تنظیمها توسط دکمه Ok و Down load کردن برنامه پیکربندی سخت افزار در CPU ، می توان از این آدرسها در برنامهی PLC برای تولید Sole

💼 (0) UR	Properties - CPU 314 - (R0/S2)
1 PS 307 2A 2 CPU 314	Time-of-Day Interrupts Cyclic Interrupt Diagnostics/Clock Protection General Startup Cycle/Clock Memory Retentive Memory Interrupts
4 DI16xDC24V 5 D016xDC24V/0.5A 6 7	Cycle Update 0B1 process image cyclically Scan Cycle Monitoring Time [ms]:
	Minimum Scan Cycle Time [ms]: Scan Cycle Load from Communication [%]: 20
	Size of the Process Image
	0885 - Call Up at 1/0 Access Error: No 0885 call up
	Clock Memory
	Clock memory Memory Byte: 100
	Cancel Help

شکل(۸–۱۳)

مثال برای استفاده از پالسژنراتور داخلی CPU: برای بررسی سه بیت از یک بایت آدرس تنظیم شده در برنامه پیکر بندی سخت افزار برای پالس ژنراتور داخلی PLC. برنامه زیرا در PLC اجرا و نتایج آن مشاهده شود.



توضیح: در این مثال می توان از المان Move برای مشاهده همه هشت بیت پالس ژنراتور بطور همزمان در خروجی QB5 استفاده کرد.

جلسه نهم

هدف :

- بررسی نحوه تنظیم تعداد حافظههای نگدارنده^۱ داده پس از توقف و یاخاموش شدن PLC
 - معرفی بلوکهای سازمانی^۲، نحوه و ترتیب الویت فراخوانی آنها
 - بررسی بلوک سازمانی راه انداز^۳ (OB100)
- بررسی بلوک سازمانی وقفه زمان از روز^¹ (OB10) و بکار گیری بلوکهای سیستمی مربوط به آن(SFC28، SFC28)
 SFC30 (SFC31 و SFC30)
- بررسی بلوک سازمانی وقفه تاخیر زمانی⁶ (OB20) و بکار گیری بلوکهای سیستمی مربوط به آن (SFC32 SFC32 SFC32) و بکار گیری بلوکهای سیستمی مربوط به آن (SFC33 SFC32 SFC32)
- بررسی بلوک سازمانی وقفه دورهای ⁶ (OB35) و بکار گیری بلوکهای سیستمی مربوط به آن(SFC39، SFC40 SFC39) و SFC41
 SFC41 و SFC42)

۱-۹) تنظیم حافظههای نگهدارنده^۷ داده:

با نرم افزار Simatic Manager می توان تعدادی از شمارنده ها، تایمرها، بایت های حافظه و تعدادی بایت های دیتا بلوک-های PLC را برای نگهداری داده های مربوط به نتایج عملیات قبل از توقف و یا خاموشی PLC، بمنظور استفاده پس از راه اندازی مجدد[^] PLC تنظیم کرد. برای این منظور، در نرم افزار پیکربندی سخت افزار، پس از کلیک روی ردیف ماژول CPU، از منوی Edit گزینه Object Properties این منظوب می این انتخاب مجموعه صفحات مربوط به Object Properties این ماژول باز می شوند. در صفحه با سر برگ Memory این اهم این مجموعه از صفحات که در شکل(۹-۵) نشان داد شده است، تنظیمات لازم انجام می شود.

Station Edit Insert PLC View	Options Window Help			
	roperties - CPU 314 - /B	0/52)		5
SIMATIC 300(1) (Config OUB SiMATIC 300(1) (Config OUB Simple and a state of the state of	Time-of-Day Interrupts General Startup Retentivity Number of Memory Bytes S Number of S7 Timers Starti Number of S7 Counters Sta	Cyclic Interrupt Cycle/Clock Memory tarting with MB0:	Diagnostics/Clock Retentive Memory	Protection Interrupts
7 	Areas Area 1: [Retentive Area 1:] Retentive Area 2:] Retentive Area 3:]	DB No. Byte Ad 1 0 1 0 1 0	dress Number of Byte	25

- ¹ Retentive Memory
- Organization Blocks
- ^{*} Startup Organization Block
- ' Time-of-day interrupt
- [°] Time delay interrupt
- [°]Cyclic interupt
- ^v Retentive Memory
- 8 Restart

برای مثال در شکل(۹–۱) برای نگهداری دادهها برای استفاده در Restart مجدد، ۱۶ بایت حافظه، ۴عدد تایمر و ۸ عدد شمارنده انتخاب شده است. اما برای دیتا بلوکها انتخابی صورت نگرفته است. **لازم است توجه شود، پس از این تنظیم** حتما باید برنامه پیکربندی سخت افزار به PLC منتقل(Download) شود. ۵۰۰

مثال:

برای درک حافظههای نگهدارنده و غیر نگهدارنده^۱ دو برنامه مثال شکل(۹–۲) را جدا گانه اجرا کرده، و در هـر یـک از ایـن برنامهها در شرایطی که مقادیر شمارنده و فیلیپ فلاپ غیر صفر هستند، یک بار PLC را خاموش و مجداد روشن کنید، و بار دیگر PLC را به حالت Stop و مجددا به حالت Run تغییر حالت دهید.

اگر این برنامهها درست اجرا شوند، در برنامه ۱ پس از راه اندازی مجدد PLC، مقادیر شمارنده و وضعیت فیلیپ فلاپ مانند قبل از توقف ثابت باقی میمانند، ولی در برنامه 2 این مقادیر صفر میشوند.



معرفی بلوک های سازمانی

مقدمه

شکل (۷-۱ از جلسه هفتم) ساختار کلی ارتباط بلوکها با یکدیگر و با Operating System را نشان می دهد. در این operating System و FG و یا FG دیگر و بلوکهای OBها توسط OBa توسط Operating فراخوانی می شخار بلوکهای FC و FC ارائه شدهاند. هر یک از بلوک های سازمانی به می شوند. در شکل (۹–۳) بلوکهای سازمانی PLCهای سری 300-S7 و 750 ارائه شدهاند. هر یک از بلوک های سازمانی به می شوند. در شکل (۹–۳) بلوکهای سازمانی Operating می شوند. برای PLC ارائه شدهاند. هر یک از بلوک های سازمانی به طور مجزا توسط می از بلوک های سازمانی می در از مان و یا شرایط خاص فراخوانی می شوند. برای PLC رویدادهای متعددی وجود دارند که باعث می شوند بلوکهای سازمانی مربوطه، بر اساس تقدم شان توسط سیستم عامل فراخوانی شوند. در جلسههای قبل نحوه بکارگیری OB4 می سری OB4 و در جلسه آینده نحوه بکارگیری OB4 می سود. Operating System این می و در این جلسه نحوه بکار گیری Operating System می شوند. برای OPE و در جلسه آینده نحوه بکارگیری Operating System در JOperation می شوند. در جلسه آینده نحوه بکارگیری Operating System در JOperating System در این جلسه نحوه بخار گیری Operating کام و در جلسه آینده نحوه بکارگیری OB40 در JOperating System Syst



شکل(۹–۳)

ترتيب الويت فراخواني OBها

اجرای برنامه یک OB می تواند توسط اتفاق با الویت بالاتر در مرزهای مشخص شدهای متوقف و برنامه OB با تقدم بالاتر اجرا شود. سپس دنباله برنامه باقیمانده از OB متوقف شده، پس از پایان اجرای برنامه OB با تقدم بالاتر، اجرا خواهد شد. تقدم GB ها ز ۱ تا ۲۶ رتبه بندی شده اند، به طوری که OB با تقدم ۱ کمترین ارزش و OB با تقدم ۲۶ دارای بالاترین ارزش در اجرای برنامه میرنامه I تا ۲۶ رتبه بندی شده اند، به طوری که OB با تقدم ۱ کمترین ارزش و OB با تقدم ۲۶ دارای بالاترین ارزش در اجرای برنامه AD با تقدم ۲۶ دارای بالاترین ارزش در اجرای برنامه می باز ۲۶ رتبه بندی شده اند، به طوری که DP با تقدم ۱ کمترین ارزش و OB با تقدم ۲۶ دارای بالاترین ارزش در اجرای برنامه مربوطه را دارند. در جدول(۹–۱) وقفههای قابل اجرا در PLC های سری 300 و ترتیب الویتهای آنها نشان داده شده است. برای مثال، هرگاه در حالیکه برنامه یکی از OB در حال اجرا است، توسط System System در خواست اجرای برنامه OB با الویت بالاتر تشخیص داده شود . در این شرایط اجرای برنامهی در حال اجرا و در حال ابرا متوقف و برنامه AD با الویت بالاتر اجرای برنامه Operating System یکی از OB در این شرایط اجرای برنامهی در حال اجرا است، توسط Operating System در خواست اجرای برنامه OB با الویت بالاتر تشخیص داده شود . در این شرایط اجرای برنامهی در حال اجرا متوقف و برنامه AD با الویت بالا تر اجرا و در یا یا الویت بالا تر اجرای برنامه یکی از Operating System در حال اجرا متوقف و برنامه AD با الویت بالا تر اجرا خواهد شد.

OB No.	ОВ Туре	Priority
08.1	Cyclic program	1
08.10	Time-of day interrupt	2
08.20	Time-delay interrupt	3
08.35	Cyclic interrupt	12
08 40	Hardware interrupt	16
08.82	Error handling	26/28

جدول(۹-۱)

در PLC های سری 400 زیمنس OBهای با الویت یکسان وجود دارند که این OBها نمی توانند یکدیگر را متوقف کنند, و یکی پس از دیگری به همان ترتیبی که توسط Operating System تشخیص داده شدهاند اجرا میشوند. شکل (۹-۴) مثالی از روند توقف OBها با الویت پائینتر و اجرای برنامه OB با الویت بالاتر را نشان میدهد.



شکل(۹–۴)

اجرای چرخشی برنامه (OB1)

برنامهای که باید به طور پیوسته اجرا شود، در بلوک سازمانی OB1 ذخیره می شود. این بلوک بـه صـورت چرخشـی بـه ترتیب نشان داده شده در شکل(۹–۵) توسط CPU اجرا می شود. یعنی ابتدا همه دادههای ورودی های PLC خوانده شده و وارد بخش حافظه PII^۱ می شوند، سپس برنامه OB1 اجرا و در نهایت نتایج اجرای برنامه از بخش حافظه PIQ^۲ به خروجی های PLC منتقل مي شوند.



شکل(۹–۵)

این روند تا زمانی که وقفه ای با درجه بالا تر اتفاق نیفتد ادامه می یابد. زمان پاسخ هر چرخش، برابر با مجموع زمان های اجرای کارهای سیستم عامل و زمان اجرای یک دوره از کل برنامه است. در بین حد فاصل زمانی انتقال نتـایج اجـرای برنامـه از PIQ بـه خروجی PLC تا انتقال دادههای ورودی PLC به PII کارهای سیستم عامل(OS^۳) اجرا می شود. نحوه اجرای این OB در جلسـه-های قبل نحوه OB1 تجربه شده است.

بلوکهای راه اندازی^۴ (OB100، OB101 و OB102)

با روشن شدن PLC، يا با تغيير از حالت Stop به Run (توسط كليد انتخاب حالت روى ماژول CPU، يا از طريق نرم افزار Simatic Manager) برنامه یکی از بلوکهای راه اندازی، قبل از اجرای برنامه بلوک چرخشی (OB1)، یک مرتبه اجرا می شود. بلوکهای راه اندازی شامل OB100 تا OB102 هستند که:

Warm restart برای OB100

. Hot restart براي OB101

OB102 برای Cold restart مورد استفاده قرار می گیرند. با اجرای برنامه این بلوکها می توان کارهای اولیه یک سیستم را قبل از شروع اجرای مجدد برنامه OB1، اجرا کرد.

- ¹ Process Image Input
- ² Process Image Output
- ³ Operating system
- ⁴ Restart

در PLCهای سری S7-300 شرکت زیمنس، به استثنای PLC مجهز به 2-PU318 فقط راه اندازی Warm بصورت دستی و یا اتوماتیک قابل اجرا است، اما در PLCهای مجهز به 2-PU318 این سری، علاوه بر داشتن قابلیت راهاندازی Warm، دارای قابلیت راهاندازی Cold بصورت دستی هم است. اما در PLCهای سری 400-57 این شرکت علاوه بر دو نوع راهاندازی ذکر شده، راهاندازی Hot هم قابل اجرا است. شکل (۹-۶) محل اجرای هر یک از بلوکهای راه اندازی OB100، OB100، و OB102 را نشان میدهد.



شکل (۶-۹)

توضیح: برای اگاهی از مطالب ۱ تا ۵ روی شکل به مرجع اصلی مراجعه شود.

مقدمه

در PLCهای سری S7-300 راهاندازی Warm Restart به صورت دستی و یا اتوماتیک قابل اجرا است. راه اندازی بصورت دستی با کلید انتخاب حالت روی ماژول CPU و یا از طریق نرم افزار Simatic Manager و همچنین راهاندازی بصورت اتوماتیک با وصل شدن برق PLC انجام می شود.

بلوک OB100 که قابل فراخوانی در زمان راه اندازی Warm Restart است را میتوان در برنامههای PLC های سری 300 بکار گرفت. برنامه این بلوک در هر نوبت راهاندازی Warm Restart فقط یکبار و قبل از هرOB دیگری توسط Operating System فراخوانی می شود. این فراخوانی در هنگام اجرای یکی از راه اندازی های زیر انجام می شود.

- ۱) هرگاه در حالتی که کلید انتخاب کننده حالت CPU در حالت RUN یا RUN-P است، به دنبال قطع برق CPU مجدد برق آن وصل شود، راه اندازی بصورت اتوماتیک اجرا می شود.
- ۲) هرگاه کلید انتخاب کننده حالت CPU، از وضعیت STOP به وضعیت RUN و یا RUN_P منتقل شود، راهاندازی بصورت دستی اجرا می شود.
- ۳) هرگاه در حالتی که کلید انتخاب کننده حالت CPU در حالت RUN و یا RUN-P است، از طریق نـرم افـزار Simatic Manager راهاندازی Warm انجام شود، راهاندازی بصورت دستی اجرا می شود.شکل (۹-۹) روند اجرای Worm Restart را در حالتهای دستی و اتوماتیک نشان میدهد.



Startup OBs

مثال برای استفاده از OB100:

برای بررسی عملکرد OB100 مراحل آمده در زیر اجرا شود.

الف) در این مثال ابتدا فقط برنامه OB1 آمده در شکل(۹-۹) به PLC منتقل و اجرا شود. سپس در شرایطی که M0.0 در وضعیت ۱ و خروجی QW4 غیر صفر است، PLC به روش دستی و اتوماتیک مجددا راهاندازی شده، در هر مرحله راهاندازی وضعیت M0.0 و مقادیر شمارنده به ذهن سپرده شود.

ب) برنامه OB100 آمده در شکل(۹–۱۰) به PLC منتقل و برنامههای OB1 بهمراه OB100 مانند راهاندازیهای بنـد الـف را ه اندازیها تکرارشود و در پایان تفاوت نتایج به دست آمده در بندهای الف و ب تحلیل شود.

توضیح: برای اضافه کردن یک OB به پروژه می توان از مسیر نشان داده در شکل(۹-۸) استفاده کرد.

🞒 File Edit	Insert	PLC	View C	Options	: Windov	v Help		
🗅 🛩 🖁	Stat	ion het		: 11	9	<u> </u>	- 100	E
E 🛃 test_3	Prog	iram		▶ Sy	stem data	-	∎ OB1	
Ē~ [57 5	oftware	э					
Ė	57 B	lock			1 Organiz	ation Bloc	:k	
	- 1917 3	ortwar	e	-	2 Function 3 Euroction	1 DIOCK		
	Sym	bol Tab	le		4 Data Blo	, 		
	Text	library		•	5 Data Tv	ne		
	Exte	rnal So	urce		6 Variable	Table		

برنامهOB1:

برنامه OB1 که در شکل(۹-۹) ارائه شده، که از دو شبکه تشکیل شده است. در شبکه اول این برنامه یک شمارنده پائین شمار آمده، که با لبه بالا رونده ورودی I0.1 مقدار اولیه شمارنده وارد آن و با هر لبه بالا رونده ورودیI0.0 یک واحد از آن کم می-شود. همچنین با سطح بالای ورودی I0.2 محتوای شمارنده پاک میشود. در شبکه دوم برنامه یک SR فیلیپفلاپ آمده است که با یک شدن ورودی I1.1 خروجی آن (Q4.7) Set (Q4.7 و با یک شدن ورودی I0.2 و یا با صفر شدن شمارنده C0 این خروجی Reset میشود.



شکل(۹-۹)

برنامه OB100:

برنامه OB100 که در شکل(۹-۱۰) ارائه شده است از دو شبکه تشکیل شده که درشبکه اول مقدار اولیه شمارند به جالت ۲۰ و



OB10.....OB17) (Time _of_ Day)) (Time _of_ Day)) (Time _of_ Day)

برنامهای که در بلوکهای مربوط به وقفه <u>زمان از روز</u> نوشته می شود، می تواند فقط یک بار و در یک زمان مشخص، و یا به صورت متناوب (یک بار در: دقیقه، یا ساعت، یا روز، یا هفته، یا ماه و یا سال) که زمان شروع دوره تناوب آن از قبل مشخص می شود، اجراء شود. برای مثال از وقفههای of _ day _ of _ of _ a توان برای ذخیره کردن اطلاعات یک فرآیند، در یک زمان مشخص از روز (مثلا هر روز ساعت ۱۷:۰۰ .) استفاده کرد. بلوکهای سازمانی OB10 تا OB17 برای وقفههای گروه <u>زمان از روز</u> هستند که در DLPهای سری S7-300 فقط بلوک OB10 قابل اجرا است. تنظیمات مورد نیاز برای بکار گیری وقفه (<u>زمان از روز</u> را به دو طریق زیر می توان انجام داد.

الف)طریقه استفاده از HW configuration) از طریق نرمافزار "HW configuration"می توان تنظیمات مورنیاز برای وقفه های مربوط به <u>زمان از روز</u> را انجام داد. روند این کار در شکل های (۹–۱۱ و ۹–۱۲) نشان داده شده است. برای اجرای این روند، در نرم افزار پیکربندی سخت افزار، پس از انتخاب ردیف CPU، از منوی Edit گزینه Object Properties مانند حالت نشان داده شده در شکل (۹–۱۱) انتخاب می شود.

🗃 (0) U	R		
1			~
2	CPU314(1)	Cody	Ctrl+C
3		Pacte	Chrl+V
4	DI16xDC24		Contry
5	DO16xDC2	Insert Object	
6	🚦 AI4/AO2x8	Add Master System	
7		Disconnect Master System	
8		Clock Synchronization	
		Delete	Del
		Go To	•
		Filter Assigned Modules	
		Object Properties	Alt+Return
		Product Support Information	Ctrl+F2
		شکل(۹–۱۱)	

با این انتخاب مجموعه صفحههای Object Properties باز می شوند که در صفحه با سربرگ Time- Of - Day که در شکل (۹–۱۲) نشان داده شده تنظیمات مور نیاز برای این نوع وقفه انجام می شود.

Propertie	s - CPU	314 - (RC)/S2)					
Genera Time-o	General Startup Time-of-Day Interrupts		Cycle/Clock Memory		Ret Diagno	entive Memory stics/Clock	Interrupts Protection	
	Priority	Active	Execution		Start date	Time of day	Process image partition	
OB10:	2		None	-	01.01.99	11:30	OB1-PA 💌	
OB11:	2		None Once	<u>^</u>	01.01.94	00:00	OB1-PA 💌	
OB12:	2		Every minute		01.01.94	00:00	OB1-PA 💌	
OB13:	2		None	-	01.01.94	00:00	OB1-PA 💌	
OB14:	2		None	-	01.01.94	00:00	OB1-PA 💌	
OB15:	2		None	-	01.01.94	00:00	OB1-PA 💌	
OB16:	2		None	-	01.01.94	00:00	OB1-PA 💌	
OB17:	2		None	-	01.01.94	00:00	OB1-PA 💌	
ОК						Cancel	Help	

شکل(۹–۱۲)

تنظیم تاریخ و زمان PLC: برای درست کارکردن وقفه Time_of_Day لازم است تاریخ و زمان PLC حتما تنظیم باشد.. برای بررسی و یا تنظیم آن در شرایطی که یکی از بلوکههای پروژه باز است **از منوی PLC گزینه Set Time Of Day** انتخاب می شود با این انتخاب صفحهی شکل(۹–۱۳) باز می شود.

Set Time of	Day		
Path:	test_FB2\SIMATIC 3	800 Station\CPU 31	15-2 DP\S7 Program
	Date:		Time of Day:
PG/PC time	05 / 13	3/10	05:34:52 pm
Module time	: 05.71	3 /10	05 :34 :52 pm
			Extended >>
Apply		Close	Help

شکل(۹–۱۳)

در قسمت بالای این صفحه زمان PC Adapter + PC) PG و در قسمت پائین صفجه زمان PLC را نشان میدهد. اگر زمان PLC درست باشد گزینه Close و اگر زمان PG درست و زمان PLC درست نباشد گزینه Apply انتخاب می شود.

در این صفحه در ستون دوم فعال و غیرفعال شدن وقفه، در ستون سوم دوره اجرا آن، در ستون چهارم تـاریخ شـروع اجـرای و در ستون پنجم زمان شروع اجرا این وقفه در تاریخ معین تنظیم میشوند. با تائید این تنظیمات توسط دکمه OK این صفحه و download کردن آن در PLC کار تنظیمات در این طریق پایان می یابد.

ب) طریقه استفاده از تابع سیستمی: از طریق به کاری گیری تابع سیستمی SFC28 در برنامه اجرائی میتوان، پارامترهای وقفه های <u>زمان از روز</u> را تنظیم کرد. در شکل(۱–۹) شکل بلوکی این تابع که در برنامه اجرائی(با زبان Ladder) قابل استفاده است نشان داده شده است.



ورودی OB_NR این بلوک برای تعین شماره OB مربوط به وقفه زمان از روز با فرمت INT، ورود SDT آن برای تعین زمان آغاز اجرای وقفه با فرمت مربوطه، ورودی PERIOD برای شماره کد تعین کننده نوع دوره اجرا با فرمت هگزادسیمال است و با کـد خروجی RET_VAL خطای احتمالی در اجرای این بلوک بررسی می شود. در جدول(۹-۲) توضیحات همه ورودی ها و خروجی مربوط به SFC28 ارائه شده است.

Parameter	Declaration	Data Type	Memory Area	Description
OB_NR	INPUT	INT	I, Q, M, D, L, constant	Number of the OB started at the time SDT + multiple of PERIOD (OB10 to OB17).
SDT	INPUT	DT	D, L, constant	Start date and time: The seconds and milliseconds of the specified start time are ignored and set to 0.
PERIOD	INPUT	WORD	I, Q, M, D, L, constant	Periods from start point SDT onwards: W#16#0000 = once W#16#0201 = every minute W#16#0401 = hourly W#16#1001 = daily W#16#1202 = weekly W#16#1202 = weekly W#16#1201 = monthly W#16#1801 = yearly W#16#2001 = at month's end
RET_VAL	OUTPUT	INT	I, Q, M, D, L	If an error occurs while the function is active, the actual parameter of RET_VAL contains an error code.

جدول(۹-۲)

اگر لازم شود تاریخ و زمان شروع دوره تناوب این وقفه(STD)، بصورت یک فرمت معین به برنامه اعمال شود این فرمت به شکل زیر است. Data Type Length (hits) Format

Data Type	Length (bits)	Format
DATE_AND_TIME	64	Yr-Mo-Day-hr:min:s.ms
or		
DT		Format Example
		DT#2002-02-02-08:30:01.999
ِ مسیر نشان داده شده	این عمل با کمک FC3 که د _ر	اگر لازم شود تاریخ و زمان بصورت دو فرمت جدا گانه به برنامه اعمال شود،

در شکل(۹–۱۵) در نرم افزار وجود دارد قابل انجام است.





در شکل(۹–۱۶) زیر مثالی ارائه شده که در آن از FC3 برای تجمیع تلریخ و زمان استفاده شده است. در این مثال حافظههای ورودهای IN1 و IN2 بصورت Word هستند که در این مثال بصورت فرمت ثابت مقدار دهی شدهاند. که ورودی IN1 مختص تاریخ و ورودی IN2 مختص زمان هستند.



در این مثال، متغیر D_A_T# یک حافظه موقت محلی بطول هشت بایت است که در OB1 بصورت DAT-AND-TIME تعریف شده است.

برای بکار گیری OB10 (وقفه زمان از روز) علاوه بر SFC28 توابع سیستمی دیگری در نرم افزار موجود است که برای فعال، غیر فعال و بررسی کردن وقفه های **زمان از روز** استفاده میشوند. شماره و کاربرد هر یک از این توابع در زیر آمده است.

- o "CAN_TINT" o وقفه ساعت زماني را لغو يا باطل مي كند.
 - SFC30 "ACT_TINT" cd وقفه ساعت زماني را فعال مي كند.
 - o "SFC31"QRY_TINT وقفه ساعت زمانی را بررسی می کند.

نحوه عملكرد SFC31 QRY_TINT را كمك HELP نرم افزار بررسى كرده و نتايج را ياداشت كنيد.

مجموعه توابع سیستمی نامبرده شده در بالا، در پنجره المانهای برنامهریزی، در مسیرنشان داه شده در شکل(۹–۱۷) در دسترس هستند





مثال برای اجرای وقفه زمان از روز: در شکل (۹–۱۸) برنامهای که در آن از وقفه زمان از روز استفاده شده، ارائه شده است. در این برنامه برای تنظیم و کنترل OB10 از SFC های مربوطه استفاده شده است. با اجرای این مثال عملکرد برنامه را بررسی کنید.

برنامهOB1:



شکل (الف۹–۱۸)

برنامه OB10:



شکل (ب۹–۱۸)

شرح برنامه :

در برنامه OB1 برای مقدار دهی به ورودی STD بلوک SFC28 از متغییر OB1_DATE_TIME استفاده شده است . این متغیر یکی از متغییرهای TEMP بلوک OB1 است که محتوای آن زمان شروع اجرای OB1 در CPU را مشخص میکند. عبارت W#16#0201 نشان دهنده اجرای برنامه وقفه به صورت دقیقهای یک بار است.

دراین برنامه با فعال شدن ورودی I0.0 خروجی بیتی Q4.0 فعال میشود. با فعال شدن این بیت بلوکهای SFC28 و SFC20 و SFC20 اجرا میشوند. چون در برنامه با کمک SFC28 برای OB10 زمان شروع وقفه (SDT) دو پریود آن WB1_DATE_TIME (SDT) و پریود آن WB16#201 تعریف شده است، به محض شروع به کار OB1، وقفه زمان_روز (OB10)، دقیقهای یک بار فراخوانی می شود. با ایس فراخوانی می شود. با ایس ایس میشود. همچناین با فعال شدن ورودی I0.1 ایس ایس می فراخوانی می شود و یک دقیقه خاموش میشود. با فعال مدن این بیت بلوکهای SFC28 و SFC28 ای OB1_DATE_TIME (SDT) امی شود. با ایس می فراخوانی می شود. با ایس فراخوانی خروجی Q5.0 در برنامه OB10 یک دقیقه روشن و یک دقیقه خاموش می شود. همچناین با فعال شدن ورودی I0.1 اورای وقفه زمان_روز وقفه زمان_روز می وقفه شد.

توضیح : در این برنامه کافی است هریک از بلوکهای SFC28، SFC29 و SFC30 فقط یک Scan Time اجرا شوند و نیاز به باقی ماندن ورودی EN هر یک در سطح یک ندارند.

4-4) وقفه تاخير_ زمانی (Time _ Delay) (OB20)

برنامه بلوک **وقفه تاخیر زمانی** پس از تاخیر معینی، بدنبال رخ داد یک عمل مشخص در برنامه اجرائی، اجرا می شود. بلوکهای -OB20 تا OB20 همگی برای وقفه تاخیر_ زمانی هستند که در PLCهای سری 300-57 فقط OB20 و در PLCهای سری S7-400 همه آنها قابل اجرا هستند. شکل (۹–۱۹) صفحه interrupt از مجموعه صفحات Object Properties مربوط به CPU را نشان میدهد. در این صفحه وضعیت تعدادی از OB هائی که در PLC های سری 300 -S7 قابل اجرا است را نشان می-دهد. در این صفحه بلوک OB20 که قابل اجرا در PLCهای سری 300 است را بهمراه شماره الویت آن که 3 است نشان داده شده است.

Time-of-Day Interrupt Cyclic Interrupt Diagnostics/Clock Protection General Statup Cyclic/Lock Memory Retentive Memory Interrupts Hardware Interrupts Process image patition: Process image patition: Process image patition: Process image patition: Async. Errol Interrupts 0840: [6] 081-PA 081-PA 081-PA 0813 26 0841: 10 081-PA 081-PA 0812 26 0883 26 0844: 10 081-PA 0812 081-PA 0826 0812 26 0844: 10 081-PA 0827 081-PA 0826 26 0885 26 0845: 12 081-PA 0855 24 0867 25 0872 28 0847: 12 081-PA 0855 24 0873 0	operties - CPU 314 - (0/\$2)		4
General Statup Cycle/Lock Memory Heteritive Memory Interrupts Hardware Interrupts Process mage patition: Process mage patition: Process mage patition: Async. Errol Interrupts 0B40 IB 011-PA 0 0B1-PA 0 0B41 10 081-PA 0 0B1-PA 0 0B42 18 081-PA 0 0B1-PA 0 0B43 19 0B1-PA 0 0 0B22 5 0B1-PA 0 0B44 20 0B1-PA 0	Time-of-Day Interrupts	Cyclic Interrupt Diagnostic:	s/Clock	Protection
	General Statup Hardware Interrupts Process partition; 0840: 105 0841: 107 0842: 18 0844: 20 0844: 20 0845: 21 0845: 21 0847: 081:PA 0846: 22 081:PA 084: 20 081:PA 081:PA	Cycle/Clock Memory Retention Time-Delay Interrupts Process Prostav partition: 0820:3 081-PA 0822:5 081-PA 0823:6 081-PA 0826:7 081-PA 0826:7 24	ve Memory Async. Err 0B81: 0B82: 0B83: 0B86: 0B86: 0B87: 0B70: 0B72: 0B73:	Interrupts Priority: 26 26 26 26 26 26 26 26 26 26 26 26 26 26 26 26 26 26 26 26 26 26 26 27 28 28 28 29 28 20 28 29 28 20 28 28 28 29 28 29 28 29 28 29 28 29 28 29 28 29 28 29 28 29 28 29 28 29 29 29 29 29 29 29 29

شکل(۹–۱۹)

وقفه های تاخیر_ زمانی میتوانند با توابع سیستمی زیر، تنظیم و کنترل شوند. بطوریکه:

- o "SFC32 "SRT_DINT" o برای فعال کردن وقفه تاخیر_ زمانی و برای تنظیم مقدار زمان تاخیر آن به کار گرفته می شود.
 - o "SFC33"CAN_DINT برای لغو یا باطل کردن وقفه تاخیر_ زمانی به کار گرفته میشود.

o "SFC34"QRY_DINT" وبرای بررسی وقفه تاخیر_ زمانی بکار گرفته می شود. مثلا اینکه آیا فراخوانی انجام گرفته یا
 نه.

توضیح اینکه، فقط با فراخوانی تابع سیستمی SFC32 از طریق برنامه اجرائی میتوان زمان تاخیر فراخوانی OB20 را تنظیم و اجرای آنرا فعال کرد. پارامترهای مربوط به این تابع سیستمی در جدول(۹-۳) آمده است. بطوریکه: OB_NR: شماره OBهایی که باید پس از یک زمان تاخیر مشخص اجرا شوند. DTIME : زمان تاخیر که از ۱ تا Soft به برنامه OB وقفه تاخیر_زمان در موقع شروع فراخوانی ایسن OB استفاده میشود. SIGN : برای انتقال داده با فرمت Word به برنامه OB وقفه تاخیر_زمان در موقع شروع فراخوانی OB2

این داده در برنامه OB های وقفه تاخیرزمان توسط حافظه موقت OBN_SIGN در دسترس قرار می گیرد. کـه در آن N شـماره OB فراخوانی شده است.

RET_VAL: اگر یک خطایی در حین اجرای وقفه تاخیر زمان اتفاق بیافتد، کد مربوطه به این خروجی منتقل میشود. (با استفاده از On line help میتوان کد خطاها را مشاهده کرد.)

Parameter	Declaration	Data Type	Memory Area	Description
OB_NR	INPUT	INT	I, Q, M, D, L, constant	Number of the OB, to be started after a time delay (OB20 to OB23).
DTIME	INPUT	TIME	I, Q, M, D, L,	Time delay value (1 to 60000 ms)
			constant	You can realize longer times, for example, by using a counter in a time-delay interrupt OB.
SIGN	INPUT	WORD	I, Q, M, D, L, constant	Identifier which appears in the start event information of the OB when the time-delay interrupt OB is called.
RET_VAL	OUTPUT	INT	I, Q, M, D, L	If an error occurs while the system function is active, the actual parameter of RET_VAL contains an error code.

جدول(۹-۳)

مثال برای اجرای وقفه تاخیر زمانی: برنامههای شکل(۱۵–۲۰) را که در آن از وقفه تاخیر_زمانی استفاده شده اجرا کنید . در این طرح ورودیهای I0.0 او I0.1 به سه عددکلید ON-Off و خروجیهای Q4.0 ، Q4.1 و QB5 به نه عدد لامپ سیگنال وصل شده اند.

برنامه OB1:



برنامه OB20:



شرح برنامه :

دراین برنامه با فعال شدن ورودی IO.0، خروجی Q4.0 فعال می شود. و با یک کردن ورودی IO.1 ، SFC32 فراخوانی می شود. با این عمل ورودی IW0 به حافظه موقت OB20_SIGN و پس از IOS خروجی Q4.1 فعال شده و بخش Low Byte حافظه موقت Q4.1 به خروجی QB5 منتقل میشوند. حال اگر کلید IO.0 غیر فعال شود خروجی Q4.0 غیر فعال میشود، ولی Q4.1 کماکان فعال باقی می ماند. حال اگر مجددا با یک شدن ورودی IO.1 تابع SFC32 فراخوانی شود، چون در این وضعیت خروجی Q4.0 صفر است خروجی Q4.1 پس از IOS خاموش می شود.

عملیات Canceling توسط SFC33 انجام می شود، یعنی زمانی که SFC32 فراخوانی شده ولی هنوز زمان Delay Time (در اینجا زمان 108) سپری نشده است اگر کلید I0.2 فعال شود، عملیات وقفه باطل می شود و OB20 فراخوانی نخواهد شد. ولی اگر بعد از سپری شدن زمان Delay Time این کلید فعال شود عملا این کلید تاثیری در اجرای برنامه ندارد.

(OB35) وقفه دورهای (OB35)

با استفاده از بلوکهای OB30 تا OB37، میتوان اجرای برنامه بلوکهای با الویت پائینتر(مثلا برنامه چرخشی OB1) را در دوره زمانی ثابت، متوقف، و برنامه بلوک عامل وقفه را اجرا کرد.

بطور مثال اگر بلوک OB35 در یک پروژهای که دارای OB1 است بکار گرفته شود و فاصله زمانی فراخوانی آن 500mS تنظیم گردد، در هر 500mS اجرای برنامه OB1 متوقف شده و برنامه این بلوک اجرا می شود. و در پایان، اجرا برنامه به ادامه برنامه OB1 برمی گردد. در PLCهای شرکت زیمنس ۹ بلوک وقفه دورهای (OB30 تا OB38) موجود است. که فقط بلوک OB35 در PLCهای سری 300-S7 قابل اجرا می باشد. در شکل (۹–۲۱) نحوه اجرای بلوک OB35 در بلوک OB1 نشان داده شده است



شکل(۹–۲۱)

فاصله زمانی و زمان شروع وقفه دورهای

در هنگام استفاده از این نوع وقفهها باید اطمینان حاصل شود، تا مدت زمانی که برای این وقفه دورهای تنظیم می شود از زمان لازم برای اجرای برنامه بلوکی که توسط این وقفه فراخوانی می شود بیشتر باشد. چون سیستم عامل، بلوک را در فاصله زمانی تنظیم شده برای آن فرا خوانی می کند و اگر اجرای برنامه بلوک فراخوانی شده در این فاصله زمانی هنوز پایان نیافته باشد سیستم عامل OB80 (بلوک خطای وقفه دورهای) را فرا می خواند. هر گاه حالت CPU از حالت STOP به RUN تغییر کند، زمان شروع دوره این وقفهدورهای آغاز می شود.

زمان دوره وقفه با استفاده از نرمافزار پیکر بندی سختافزار، تنظیم می شود. در شکل(۹-۲۲) محل تنظیم این زمان در صفحه Cyclic interrupt (از مجموعه صفحات Object Properties CPU) نشان داده شد است, مدت زمان فراخوانی پیش فرضms 100 است و می توان آن را به مقداری در محدوده مجاز (از 1ms تا 608) تنظیم کرد. برای نیاز به زمان بیشتر می توان با به کار گیری یک شمارنده در OB35 به آن دست یافت.

Properties - C	:PU 314 - (R	0/\$2)			
General	Startup	Cycle/Clo	ck Memory	Retentive Memory	Interrupts
Time-of-Day	/ Interrupts	Cyclic Int	errupt	Diagnostics/Clock	Protection
Prio	rity Exe	cution (ms)	Phase of	roc fset (ms)	cess image partition
OB30; 7	500	0	0	OE	31-PA 💌
OB31: 8	200	0	0	OE	31-PA 💌
OB32: 9	100	0	0	OE	31-PA 💌
OB33; 10	500		0	OE	31-PA 💌
OB34: 11	200		0	OE	31-PA 💌
OB35: 12	100	$ \rightarrow $	0	OE	31-PA 💌
0836; 13			0	OE	31-PA 💌
OB37: 14	20		0	OE	81-PA 💌
OB38: 15	10		0	OE	31-PA 💌
OK				Cancel	Help

شکل(۹–۲۲)

توجه شود، بعد از تنظیم زمان وقفه در برنامه پیکر بندی سختافزار، حتما باید این برنامه Down load شود. این نوع وقفهها را

- می توان توسط بلوکهای سیستمی SFC39 تا SFC42 در برنامه اجرائی کنترل کرد. که:
 - SFC39 برای Disable کردن وقفه
 - SFC40 برای Enable کردن وقفه
 - SFC41 براى به تاخير انداختن اجراى وقفه با تقدم بالاتر
 - SFC42 براي خذف به تاخير انداختن اجراي وقفه با تقدم بالاتر هستند

مثال برای اجرای وقفه دورهای: برنامههای شکل(۹-۲۳) را اجرا و نتایج آنرا بررسی کنید. برای تنظیم زمان ثابت برای وقفه دورهای، در نرم افزار پیکر بندی سختافزار پس از باز کردن مجموعه صفحات Object properties مربوط به CPU، در سر برگ Cyclic Interrupt ، مقدار Execution time را 500ms وارد کنید. توجه کنید پس از این تنظیم لازم است حتماً پیکربندی Download شود برنامه OB1:



شرح برنامه:

بر نامه OB35:

به محض شروع به اجرای این برنامه در PLC ، در هر 500ms یک بار OB35 فراخوانی و با این فراخوانی خروجی Q5.0 با فرکانس یک هرتز روشن و خاموش می شود. با فعال شدن ورودی I0.1، بلوکSFC39 فراخوانی شده، و با اجرای آن عمل فراخوانی دوره ای بلوک OB35 لغو شده و تا زمانی که SFC40 توسط ورودی I0.2 فراخوانی نشود در این حالت باقی خواهد ماند. کد ورودی Mode این دو بلوک سیستمی تعین کننده این است که کدام گروه از وقفه های PLC فعال و یا غیر فعال شوند.

در این برنامه هر یک از بلوکهای SFC39 و SFC40 کافی است فقط در یک Scan Time اجرا شوند، .و نیازی بـه بـاقی ماننـد ورودی EN در وضعیت یک ندارند.

جلسه دهم

هدف:

بررسی امکانات سخت افزار همراه ماژول CPUIFM برای:

- ایجاد وقفه سخت افزاری⁽ (با OB40)
- ایجاد یک شمارنده سخت افزاری بالا و پائین شمار تا فرکانس 10KHZ
- ایجاد دو شمارنده موازی سخت افزاری بالا و پائین شمار A و B تا فرکانس 10K
 - ایجاد یک فرکانس متر سخت افزاری تا 10kHZ
 - ایجاد یک شمارنده تعیین موقعیت برای استفاده از شفت انکودر

۱-۱۰) وقفه سخت افزاری

مقدمه

برنامه وقفه سخت افزاری توسط سیگنال های مشخصی از بعضی ماژولها فراخوانی میشود. همه ماژولها دارای قابلیت ایجاد وقفه سخت افزاری را ندارند. برای ماژولهائی که این قابلیت را دارند، لازم است با استفاده از نرمافزار پیکر بندی سخت افزار، تنظیمهای مورد نیاز برای بکار گیری این قابلیت انجام شود. این تنظیمها در بخش Object Properties این ماژولها انجام میشود.



شکل(۱۰–۱)

برای مثال، در شکل(۱۰–۱) نمونهای از این نوع تنظیم در بخش Object Properties یک ماژول ورودی آنالوگ نشان داده شده است. در این تنظیم محدوده مقادیر ورودی آنالوگ طوری تنظیم شده که اگر دامنه سیگنال ورودی این ماژول از محدودهی تنظیم شده خارج شود, بلوک برنامه وقفه سخت افزاری انتخاب شده برای آن فراخوانی می شود. توضیج اینکه در PLCهای سری 300-S7 فقط بلوک0B40 قابل اجرا بوده و همچنین هر ماژولی از این سری PLC دارای قابلیت ایجاد وقفه سخت افزاری نیست. ماژول PLC یکپارچه OB40-IFM به همراه CPU آن دارای امکانات سخت افزاری است که یکی از قابلیت این امکانات ایجاد وقفه سخت افزاری از طریق ورودی دیجیتال است. برای این منظور چهار بیت از ورودی های دیجیتال این ماژول (با آدرسهای 1126.0 (با آدرسهای 1126.0) دارای توانائی ایجاد وقفه، طبق رود ریما می شود. از این مازول (با آدرسهای 1126.0) افزاری از دارای از طریق ورودی دیجیتال است. برای این منظور چهار بیت از ورودی های دیجیتال این ماژول (با آدرسهای 1126.0)

- ۱- یک پروژه با استفاده از ماژول CPU314 IFM بهمراه بلوکهای OB1 و OB40 ایجاد می شود.
- ۲- در برنامه پیکربندی سخت افزار، بر روی ماژول CPU314 IFM راست کلیک شده و از منوئی که مانند شکل(۲۰–۲) باز خواهد شد گزینه Object Properties انتخاب می شود. با این انتخاب، مجموعه صفحههای Object Properties این ماژول باز می شوند.
 - 0) UR CPU314 IFM(1) Copy Ease CrHV Ease CrHV Insert DP Master System Delete Delete Cono Filter Assigned Modules Edit Symbolic Names... Object Properties... Alt+Return (Y − 1+) L
- <u>Integrated</u> از مجموعه صفحههای Object Properties که در شکل(۱۰–۳) نشان داده شده است صفحه با سربرگ <u>Parameter Assignment</u> انتخاب و دکمه Interrupt Inputs فعال می-شود.

Properties - C	PU 314IFM	- (R0/S2)					2
Interrupts	Time-of-D	ay Interrupts	Cyclic I	nterrupt	Diagnostics/0	Clock	Protection
General	Startup	Integrated Fur	nction	Cycle/C	lock Memory	Reter	tive Memory
Select Inte	egrated Funct	on					
🔿 <u>N</u> o Fur	nction Active						
Interru	pt Inputs						
C <u>C</u> ounte	er						
C Paralle	l Counter						
C <u>F</u> reque	ency Meter						
○ Positio	ning		[1	7)	
				Faramet	er Assignment		
ПК					Car		Help
-1							
شکل(۱۰–۳)							

۴- با فعال شدن این دکمه مجموعه صفحههای Integrated Function که در شکل(۱۰-۴) نشان داه شده است باز می شود. در این شکل صفحهای که در رو قرار گرفته دارای سربرگ Interrupts است.

Integrated functions					X				
Interrupts Counter Counter A Counter B Frequency Meter Positioning									
Enabling the Interrupt Inputs									
Interrupt Input:	1	2	3	4					
Deactivated:	c	ø	¢	۲					
Rising (Positive) Edge:	• 🗆	•	•	•					
Falling (Negative) Edge:	C	c	C	¢					
ОК				Cancel	Help				

شکل(۴-۱۰)

۵- ستونهای ۱ تا ۴ این صفحه مربوط به ایجاد وقفه سخت افزاری از طریق ورودیهای دیجیتال با آدرسهای I126.0 تا
 ۵- مستونهای ۱ تا ۴ این صفحه مربوط به ایجاد وقفه سخت افزاری از طریق ورودیهای دیجیتال با آدرسهای I126.0 تا
 ۶) فقط ستون یک (که مربوط به ورودی دیجیتال را میتوان برای ایجاد وقفه با لبه مورد نظر تنظیم کرد. در شکل (۰۱- ۴) فقط ستون یک (که مربوط به ورودی دیجیتال I126.0 است) برای ایجاد وقفه با لبه مورد نظر تنظیم شده است بقیه
 ۳) فقط ستون یک (که مربوط به ورودی دیجیتال را میتوان برای ایجاد وقفه با لبه بالارونده تنظیم شده است بقیه مستون ها ستون یک (که مربوط به ورودی دیجیتال I126.0 است) برای ایجاد وقفه با لبه بالارونده تنظیم شده است بقیه ستونها غیر فعال هستند. در پایان با تائید تنظیمها بوسیله دکمههای OK ، کار پیکربندی سخت افزار برای ایجاد وقفه سخت افزاری با ماژول CPU314IFM پایان مییابد.

توجه شود در اینجا لازم است پیکربندی آماده شده به PLC منتقل (Down Load) شود.

توضیح: وقفه های سخت افزاری میتوانند با توابع سیستمی (SFC ها) مربوطه کنترل شوند بطوریکه:

با اجراى تابع DIS_IRT) SFC39) بذيرش وقفه غيرفعال مىشود

با اجرای تابع SFC40 (EN_IRT) پذیرش وقفه فعال می شود.

با اجرای SFC41 (DIS_AIRT) در زمان اجرای وقفه، اجرای وقفههای جدید با اولویت بالاتر و یا وقفههای خطاها غیرهمزمانی تا پایان اجرای وقفه جاری به تاخیر میافتد.

با اجرای SFC42 (EN_AIRT) به تاخیر افتادن اجرای وقفههای با اولویت بالاتر و یا وقفههای خطاها لغو می شود.

مثال برای کاربرد وقفه سخت افزاری:

پس از تنظیم مازول CPU314IFM برای ایجاد وقفه سخت افزاری برنامه زیر را اجرا و عملکرد آن را بررسی کنید

برنامه OB1:



برنامهOB40:

OB40 : "Hardware Interrupt"

Network 1: Title:

	ADD. EN	l ENO	
QW124 —	IN1	OUT	-QW124
2 —	IN2		

10KHZ شمارنده سخت افزاری بالا و پائین شمار تا فرکانس 10KHZ

مقدمه

از امکانات سخت افزاری همراه CPU در ماژول CPU314IFM میتوان برای بکار گیری یک شمارنده بالا و پائین شمار، تا فرکانس 10KHZ استفاده کرد. تمام امکانات این شمارنده در شکل بلوکی (۱۰–۵) نشان داده شده است. این امکانات با استفاده از سخت افزار موجود در ماژول CPU314IFM که در برنامه پیکربندی سخت افزار پارامترهای آن تنظیم میشوند و بلوک نرم افزاری SFB29 که بعدا معرفی خواهد شد، حاصل میشوند.

شکل بلوکی(۱۰–۵) نحوه ارتباط بخش سخت افزار شمارنده با بلوک سیستمی SFB29 و بلوک سازمانی OB40 (مربوط به وقفه سختافزاری) و ورودی و خروجیهای سخت افزاری ماژول CPU314IFM را نشان میدهد.



شکل(۱۰–۵)

شکل بلوکی (۱۰-۶) مجموعه امکانات سختافزاری _ نرمافزاری این شمارنده را نشان میدهد. این بلوک شامل یک شمارنده و دو مقایسه کننده A و B است. این سه المان همگی ۳۲ بیتی و سخت افزاری هستند. در این طرح مقادیر دو مقایسه کننده، با مقادیر شمارنده مقایسه شده و نتایج آنها بصورت دو سیگنال سختافزاری در خروجی ماژول و دو بیت داده نرمافزاری از طریق برنامه اجرائی کاربر در اختیار قرار میگیرند. مقادیر مقایسه کنندها و مقدار اولیه شمارنده از طریق برنامه اجرائی کاربر در آنها بار می-شوند. مقادیر بار شده در مقایسه کنندهها و مقدار واقعی شمارنده از طریق نرامه اجرائی کاربر در آنها بار می-شوند. مقادیر بار شده در مقایسه کنندهها و مقدار واقعی شمارنده از طریق نزم افزار در اختیار برنامه کاربر قرار میگیرند. سیگنالهای شمارش برای شمارش بالا و پائین، سیگنال تعیین جهت شمارش و سیگنال میشوند. افزار و بیت Start/Stop نرمافزاری از طریق برنامه اجرائی کاربر به شمارنده اعمال میشوند.



شکل(۱۰-۶)

توضیح اینکه در شکل بلوکی (۱۰–۶) همه ورودی و خروجیهائی که با دو نوع نوشتار نام گزاری شدهاند نرم افزاری و بقیه ورودی خروجیها سخت افزاری میباشند.

شکل(۱۰-۷) مثالی از تغییر مقادیر شمارنده را در ارتباط با سیگنالهای شمارش بالا و پائین نشان میهد. بطوریکه از روی شکل مشاهده میشود در این مثال شمارش بالا با لبه بالا رونده سیگنال و شمارش پائین با لبه پائین رونده سیگنال انجام میشود. فعال و غیر فعال شدن هر یک از سیگنالهای شمارش و انتخاب نوع لبه برای آنها، در برنامه پیکربندی سختافزار تنظیم میشوند.







شکل(۱۰–۸)

در شکل(۱۰–۹) فعال و غیر فعال شدن سیگنال سخت افزاری Digital output A و بیت نرم افزاری Status bit A در ارتباط با وضعیت مقایسه کننده A را نشان می دهد. بطوریکه از روی شکل مشاهده می شود، فعال و غیر فعال شدن سیگنال خروجی دیجیتال علاوه بر نتایخ حاصل از مقایسه کننده وابسته به نوع تنظیم این خروجی که در نرم افزار پیکربندی سخت افزار انجام خوهد شد، است. ولی بیت نرمافزاری Status bit فقط وابسته به وضعیت نتایخ مقایسه کننده است.



شکل(۱۰–۹)

معرفی بلوک سیستمی SFB29

بلوک سیستمی SFB29 که شمای آن در شکل(۱۰–۹) آمده، رابط بین شمارنده سخت افزاری و برنامه کاربراست. بجز ورودی و خروجی EN و ENO دیگر ورودی خروجیهای این بلوک منطبق با ورودی خروجیهای نرم افزاری شکل بلوکی(۱۰–۵) امده در مقدمه میباشند. وروری و خروجیهای این بلوک سیستمی به شرح زیر هستند.



شکل(۱۰-۹)

معرفی ورودیهای بلوک:

EN: بصورت BOOL، با یک شدن آن به بلوک اختیار عملکرد داده می شود PRES_COUNT: برای مقدار دهی اولیه برای شمارنده بوده و فرمت داده آن Dint است. PRES_COMP_A: برای مقدار دهی اولیه برای مقایسه کننده A بوده و فرمت داده آن Dint است. PRES_COMP_B: برای مقدار دهی اولیه برای مقایسه کننده B بوده و فرمت داده آن Dint است. PRES_COMP_B: برای مقدار دهی اولیه برای مقایسه کننده B بوده و فرمت داده آن Dint است. EN_COUNT: بصورت BOOL و با یک شدن آن به شمارنده اختیار شمارش سیگنال های شمارش بالا و پائین داده می شود. EN_DO: بصورت BOOL و یک شدن آن به دو خروجی دیجیتال اختیار عمل داده می شود SET_CONT: بصورت BOOL، با یک شدن آن مقدار اولیه شمارنده به داخل حافظه آن وارد می شود. SET_COMP_A: با یک شدن آن مقدار اولیه مقایسه کننده B به داخل حافظه آن وارد می شود. SET_COMP_B: بصورت BOOL، با یک شدن آن مقدار اولیه مقایسه کننده B به داخل حافظه آن وارد می شود. معرفی خروجی های بلوک: ENO: با فرمت BOOL، هر گاه در عملکرد بلوک خطائی رخ دهد و یا اینکه ورودی EN یک نشود صفر می شود. COUNT: با فرمت Dint نشان دهنده مقدار واقعی شمارنده COMP_A: با فرمت Dint نشان دهنده مقدار بارگذاری شده برای مقایسه کننده A STATUS_B: با فرمت BOOL نشان دهنده وضعیت مقایسه کننده A STATUS_B: با فرمت BOOL نشان دهنده وضعیت مقایسه کننده B

- معرفی ورودی خروجی سیگنالهای سخت افزاری شمارنده

در جدول(۱۰–۱) آدرسهای ورودی خروجی سیگنالهای سخت افزاری شمارنده نشان داده شده است. این ورودی خروجیها منطبق با ورودی خروجیهای سخت افزاری آمده در شکل بلوکی (۱۰–۵) مقدمه میباشند.

CPU 314 IFM	Function
I 126.0	Digital input up
I 126.1	Digital input down
I 126.2	Digital input direction
I 126.3	Digital input hardware start/stop
Q 124.0	Digital output A
Q 124.1	Digital output B

جدول(۱۰–۱)

پیکربندی سخت افزاری برای شمارنده

برای انتخاب نوع کاربری از سخت افزار موجود در ماژول CPU314IFM در صفحه شکل(۱۰–۱۱) یک از گزینهها انتخاب می شوند. این صفحه یکی از مجموعه صفحههای مربوط به Object Properties ماژول CPU314IFM با سر برگ Integrated Function است، که در اینجا نوع استفاده از سخت افزار، شمارنده انتخاب شده است.

Properties - CPU 314 IFM - (R0/S2)			
Time-of-Day Interrupts Cyclic Interrupt General Startup Integrated Function	Diagnostics/Clock Cycle/Clock Memory	Protection Cor Retentive Memory	nmunication Interrupts
Select Integrated Function			
C No Function Active			
C Interrupt Inputs			
 Counter 			
C Parallel Counter			
C Frequency Meter			
C Positioning	Parameter As	signment	
ОК		Cancel	Help

شکل(۱۰–۱۱)

در این صفحه پس از انتخاب Counter با فعال شدن دکمه Parameter Assignment ، صفحه شکل(۱۰–۱۲) باز می شود. در ایـن صفحه پارامترهای شمارنده انتخاب شده تنظیم می شوند. این پارامترها شامل:

- تعین شماره DB اختصاصی برای بلوک سیستمی SFB29
- تعیین نوع لبه برای سیگنالهای ورودی شمارش بالا و پائین و یا غیر فعال کرد هر یک از این ورودیها

- on o: برای set کردن خروجی دیجتال است
- o Off رای reset کردن خروجی دجیتال لست
- o Unaffected: برای بدون تغییر حالت خروج است.
 - o تعیین نوع تاثیرگذاری مقایسهکنندههای A و B روی سخت افزار شامل:
 - ٥ فعال كردن وقفه سخت افزارى.
 - o Reset کردن شمارنده.
 - بارگذاری مقایسهکننده.

Integrated functions				×		
Interrupts Counter Counter A Counter B Frequency Meter Positioning						
Instance DB	Counting Inputs					
DB Number: 63	Up:	Rising (positive)	edge 💌			
Automatic update at the scan cycle check point	Down:	Rising (positive) Deactivated Rising (positive) Falling (pegative)	edge			
Output and Comparator		T annig (riogauri	,	-		
Counting Event: Comp. Value A Reached Going Up	Comp. Value A Abandoned Going Down	Comp. Value B Reached Going Up	Comp. Value B Abandoned Going Down			
Digital Output: On 💌	Off 🗨	Off	On 💌			
Hardware interrup		Γ				
Reset counter: Off		Γ				
Set comparator:	Γ	Γ	Γ			
ОК		C	ancel Help			

شکل(۱۰–۱۲)

- برنامه برای مثالی از کاربرد شمارنده سخت افزاری در ماژولCPU314-IFM :

این برنامه را آماده کرده و نحوه عملکرد شمارنده را بر اساس آنچه در این بخش آمده بررسی کنید.


۰۱-۲)دو شمارنده سخت افزاری بالا و پائین شمار موازی A/B (Counter A/B Integrated Function) مقدمه

از امکانات سخت افزاری همراه CPU در ماژول CPU314IFM میتوان برای بکار گیری دو شمارنده بالا و پائین شمار مشابه، و بطور موازی برای شمارش فرکانس تا 10KHZ استفاده کرد. این دو شمارنده A و B نام گذاری شدهاند شکل بلوکی زیر قابلیتهای هر یک از شمارندهها را نشان میدهد. این قابلیت با استفاده از سخت افزار موجود در ماژول CPU314IFM که در برنامه پیکربندی سخت افزار پارامترهای آن تنظیم میشوند و بلوک نرم افزاری SFB38 که در ادامه معرفی میشود، حاصل میشود.

شکل بلوکی (۱۰–۱۳) نحوه ارتباط بخش سخت افزار شمارنده با بلوک سیستمی SFB38 و بلوک سازمانی OB40 (مربوط به وقفه سختافزاری و ورودیو خروجیهای سخت افزاری ماژول CPU314IFM را نشان میدهد.



شکل (۱۰–۱۳)

شکل بلوکی (۱۰–۱۴) که مجموعه امکانات سخت افزاری – نرم افزاری هر یک از شمارندهها را نشان میدهد. شامل یک شمارنده و یک مقایسه کننده ۳۲ بیتی سخت افزاری است. در این طرح مقدار موجود در مقایسه کننده، با مقدار شمارنده مقایسه شده و نتایج این مقایسه بصورت یک سیگنال سخت افزاری به خروجی سخت دیجیتال افزاری منتقل میشود. مقدار اولیه مقایسه کننده از طریق نرم افزار در آن بار میشود، این مقدار بار گذاری شده و همچنین مقادیر شمارش شده توسط شمارنده از طریق نرم افزار در اختیار برنامه اجرائی کاربرقرار می گیرند. سیگنالهای شمارش بالا و پائین و سیگنال فعال و غیر فعال کنند آنها از طریق سخت افزار به شمارنده اعمال می شوند. با بیت نرم افزاری مقدار اولیه شمارنده که در برنامه پیکربندی سخت افزاری مقدار آن تعیین خواهد شد، در شمارنده بار گذاری می شود.



شکل بلوکی(۱۰–۱۴) برای شمارندههای موازی A و B

در شکل بلوکی (۱۰–۱۴) همه ورودی و خروجیهائی که با دو نوع نوشتار نام گزاری شدهاند نرم افزاری و بقیه ورودی خروجیها سخت افزاری میباشند.

شکل (۱۰–۱۵) مثالی از تغییر مقادیر یکی از دو شمارنده را در ارتباط با سیگنالهای شمارش بالا و پائین نشان میهد. بطوریکه از روی شکل مشاهده میشود شمارش بالا و شمارش پائین در این مثال هر دو با لبه بالارونده سیگنال مربوطه انجام میشود.



شکل (۱۰–۱۵)

توضیح: بطوریکه در شکل مثال(۱۰–۱۵) مشاهده هریک از شمارندههای A و یا B میتوانند با سیگنال یکی از ورودیها بسمت بالا و با سیگنال ورودی دیگر بسمت پائین شمارش کنند. اما در برنامه پیکربندی سختافزار میتوان با تغییر انتخاب گزینهای، نحوه عمل شمارش بصورت شکل (۱۰–۱۶) تغییر داده شوند. بطوریکه با لبه سیگنال یکی از ورودیها شمارنده بالا و یا پائین شمارش کرده و با سطح سیگنال ورودی دیگر جهت این شمارش تعین میشود.



شکل (۱۰–۱۶)

شکل (۱۰-۱۷) مثالی از نحوه عملکرد مقایسه کننده را در ارتباط با شمارنده نشان می هد.



شکل (۱۰–۱۷)

در شکل (۱۰–۱۸) وضعیت سیگنال سخت افزاری Digital output را در ارتباط با وضعیت مقایسه کننده A نشان میدهد. بطوریکه از روی شکل مشاهده میشود، فعال و غیر فعال شدن سیگنال خروجی دیجیتال علاوه بر نتایخ حاصل از مقایسه کننده وابسته به نوع تنظیم این خروجی در نرم افزار پیکربندی سخت افزار است. این تنظیمها شامل موار زیر است:

On: برای set کردن خروجی دیجتال است Off برای reset کردن خروجی دجیتال لست Toggle) (Change): برای تغییر حال قبلی خروجی است از Set به Reset و یا از Reset به Set Set به Set به Inaffected (Unaffected: برای بدون تغییر حالت خروج است.



شکل (۱۰–۱۸)

– معرفی ورودی خروجی سیگنالهای سخت افزاری شمارندهها در جدول(۱۰-۲) آدرسهای ورودی خروجی سیگنالهای سخت افزاری شمارندهها نشان داده شده است. این ورودی خروجیها منطبق با ورودی خروجیهای سخت افزاری آمده در شکل بلوکی مقدمه میباشند.

Identifier	Description		
I 126.0	Counter A: Up (Up/Down)		
I 126.1	Counter A: Down (Direction)		
I 126.2	Counter B: Up (Up/Down)	Identifier	Description
T 126 3	Counter B: Down	L+	Supply voltage
1120.5	(Direction)	Q 124.0	Digital output counter A
L+	Supply voltage	Q 124.1	Digital output counter B
М	Ground	М	Ground

جدول(۱۰-۲)

توضیح: برای اینکه هر یک از شمارندههای A و یا B با پالس یکی از ورودیها(مثل I126.0) بالا و با پالس دیگر ورودی(مثل I126.1)) بالا و با پالس دیگر ورودی(مثل I126.1)) پائین شمارش کند و یا اینکه با پالس یک ورودی (مثل I126.0)) هم بالا و هم پائین شمارش کرده و دیگر ورودی (مثل I126.1)) جهت شمارش را تعین شود. تنظیم آن در Object Properties مربوط به CPU314IFM انجام می شود.

معرفي بلوک سيستمي SFB38

بلوک سیستمی SFB38 که شمای آن در شکل (۱۰–۱۹) آمده، رابط بین شمارنده سخت افزاری و برنامه کاربراست. بجز ورودی و خروجی EN و ENO دیگر ورودی خروجیهای این بلوک منطبق با ورودی خروجیهای نرم افزاری شکل بلوکی (۱۰– ۱۳) امده در مقدمه میباشند وروری و خروجیهای این بلوک سیستمی به شرح زیر هستند.



شکل (۱۹–۱۹)

ورودیهای بلوک:

EN: با فرمت Bool با یک شدن آن به بلوک اختیار عملکرد داده می شود

PRES_COMP: با فرمت Dint، برای مقدار دهی اولیه مقایسه کننده استفاده می شود.

EN_COUNT: با فرمت Bool، با یک شدن این ورودی سیگنالهای ورودی برای شمارش پذیرفته میشود.

Reset: با فرمت Bool با یک شدن این ورودی مقدار Reset که در Object Properties تنظیم می شود بعنوان مقدار اولیه وارد شمارند می شود.

SET_COMP: با فرمت Bool، با لبه بالارونده این ورودی مقدار اولیه مقایسه کننده بار گذاری می شود.

خروجیهای بلوک:

ENO: با فرمت BOOL، هرگاه در عملکرد بلوک خطائی رخ دهد و یا اینکه ورودی EN یک نشود صفر میشود. COUNT: با فرمت Dint، مقدار خروجی شمارنده را نشان میدهد.

COMP: با فرمت Dint، مقداری که در مقایسه کنند بار شده نشان میدهد

-پیکربندی سخت افزاری برای شمارنده A و B

نوع کاربری از سخت افزار موجود در ماژول CPU314IFM در صفحه شکل (۲۰–۲۰) انتخاب می شوند. این صفحه یکی از مجموعه صفحه های مربوط به Object Properties ماژول CPU314IFM با سر برگ Integrated Function است، که در اینجا نوع استفاده از سخت افزار، شمارنده های موازی A و B انتخاب شده است..

Properties - CPU 314 IFM - (R0/S2)	
Time-of-Day Interrupts Cyclic Interrupt Diag General Startup Integrated Function Cycle/	nostics/Clock Protection Communication / /Clock Memory Retentive Memory Interrupts
Select Integrated Function	
C No Function Active	
C Interrupt Inputs	
C Counter	
 Parallel Counter 	
I™ B C Frequency Meter	
C Positioning	Promoter Assimument
	Farameter Assignment
ОК	Cancel Help

شکل (۱۰–۲۰)

در صفحه بالا با انتخاب دکمه Parameter Assignment ، یکی از دو صفحه شکل (۱۰–۲۱) باز می شود. در این وضعیت با انتخاب هریک از سربرگهای A و B هر دو صفحه زیر در دسترس خواهد بود. در این صفحهها پارامترهای شمارندههای A و B تنظیم می شوند. این پارامترها شامل:

• - - تعین شماره DB اختصاصی برای بلوک سیستمی SFB38 برای هر یک از شمارندها

- تعیین نوع تاثیر گذاری مقایسه کننده های روی سیگنال خروجی دیجیتال برای هر یک از شمارنده ها شامل:
 - o On o: برای set کردن خروجی دیجتال است
 - off o برای reset کردن خروجی دجیتال لست
- o Toggle و یا از Toggle و یا از Set به Reset و یا از Set به Reset و یا از Reset به Set به Set به Set
 - o Unaffected: برای بدون تغییر حالت خروج است.
 - تعیین نوع تاثیر گذاری مقایسه کننده هر یک از شمارنده ها روی سخت افزار مربوطه شامل:

فعال کردن وقفه سخت افزاری. Reset کردن شمارنده. بارگذاری مقایسهکننده.

Integrated functions				X
Interrupts Counter Cou	nter A Counter B F	Frequency Meter Positionin	g	
Instance DB		Counter Characteristic Val	lues	
DB Number: 60		Counting Signals: 💿 Up	and Down	
		C Pul	se and Direction	
Automatic Update	heck Point	Reset Value	100	
- Output and Comparato	r			
Counting Event:	Comparison Value Reached Going Up	Comparison Value Reached Going Down		
Digital Output:	On 💌	Off		
Hardware Interrupt:		Unaffected On		
Reset Counter:		Off Toggle		
Set Comparator:				
			Cancel	Help
ارنده A	ظیم دای شم	۔ ۲) نشاندھندہ تنو	کا (۱-۱۰	ث
ارنده A	ظیم برای شم	۲) نشاندهنده تن	لکل(۱۰–۱	ے۔ ۲
ارنده A Integrated functions	ظیم برای شم	۲) نشاندهنده تن	لکل(۱۰–۱	ش 🗙
A بارنده Integrated functions Interrupts Counter Co	ظیم برای شم unter A Counter B	۲) نشاندهنده تنع Frequency Meter Positioni	، کل(۱۰)	ش 🗙
A ارنده Integrated functions Interrupts Counter Co Instance DB	ظیم برای شم unter A Counter B	تناندهنده تند Frequency Meter Positioni Counter Characteristic V	۔ مکل (۱۰ – ۱ ing alues	ش ×
A بارنده Integrated functions Interrupts Counter Co Instance DB DB Number: 6	ظیم برای شم unter A Counter B آ	۲۰ نشاندهنده تنظ Frequency Meter Positioni Counter Characteristic Va Counting Signals: C Up	بکل (۱۰–۱۰ ing alues p and Down	ش ×
A الرنده Integrated functions Interrupts Counter Co Instance DB DB Number: 6	ظیم برای شم unter A Counter B	تناندهنده تنا Frequency Meter Position Counter Characteristic Vi Counting Signals: C Ui C Pu	المحل (۱–۱۰ ing alues p and Down ulse and Direction	ش الا
A الرنده Integrated functions Interrupts Counter Co Instance DB DB Number: 6 DB Number: 6	ظیم برای شم unter A Counter B آ Check Point	Frequency Meter Position Counter Characteristic V، Counting Signals: ۲ Up ۲ Pu Reset Value	ing alues p and Down ulse and Direction 200	ش ×
A Julies A Integrated functions Interrupts Counter Co Instance DB DB Number: 6 V Automatic Update at the Scan Cycle (Output and Comparate	ظیم برای شم unter A Counter B آ Check Point	Frequency Meter Positioni Counter Characteristic Va Counting Signals: C Up C Pu Reset Value	ing alues p and Down ulse and Direction 200	ش الا
A Julies A J	dلیم برای شم unter A Counter B آ Check Point or Comparison Value Reached Going Up	Frequency Meter Positioni Counter Characteristic Vi Counting Signals: C Uf Pu Reset Value Comparison Value Reached Going Down	ing alues p and Down ulse and Direction 200	ش الا
A Julies A Interrupts Counter Co Interrupts Counter Co Instance DB DB Number: 6 Automatic Update at the Scan Cycle (Output and Comparate Counting Event: Digital Output:	dلیم برای شم unter A Counter B Check Point Comparison Value Reached Going Up	Frequency Meter Positioni Counter Characteristic Vi Counting Signals: ۲ ال Reset Value Reset Value Comparison Value Reached Going Down	ing alues p and Down ulse and Direction 200	ش الا
A Julies A Interrupts Counter Co Interrupts Counter Co Instance DB DB Number: 6 Automatic Update at the Scan Cycle D Output and Comparate Counting Event: Digital Output: Hardware Interrupt:	طیم برای شم unter A Counter B Unter A Counter B Counter B Co	Frequency Meter Positioni Counter Characteristic V Counting Signals: ۲ لار Reset Value Reset Value Comparison Value Reached Going Down	ing alues p and Down ulse and Direction 200	ش • • • • • • • • • • • • • • • • • • •
A Juites A Interrupts Counter Counting Event: DB Number: 6 Automatic Update at the Scan Cycle O Output and Comparate Counting Event: Digital Output: Hardware Interrupt: Reset Counter:	طیم برای شم unter A Counter B Unter A Counter B Check Point Check Point Comparison Value Reaching Up On Unaffected On Toggle	Frequency Meter Positioni Counter Characteristic V: Counting Signals: C Uj Reset Value Reset Value Comparison Value Reached Going Down Off ح	ing alues p and Down ulse and Direction 200	ش
A Juites A Conternations Interrupts Counter Conternations Interrupts Counter Conternation Instance DB DB Number: 6 V Automatic Update at the Scan Cycle O Output and Comparate Counting Event: Digital Output: Hardware Interrupt: Reset Counter: Set Comparator:	طیم برای شم unter A Counter B Check Point Comparison Value Reached Going Up On ▼ Unaffected Of Of Toggle	Frequency Meter Positioni Counter Characteristic V. Counting Signals: © Up © P. Reset Value Comparison Value Reached Going Down Off	ing alues p and Down ulse and Direction 200	ش بر بر بر بر بر بر بر بر بر بر بر بر بر

شکل(۱۰–۲۱) نشاندهنده تنظیم برای شمارنده B

توضیح: شمارههای دو دیتا بلوک اختصاصی، که برای SFB38 در Object Properties مربوط به CPU314IFM برای شمارندههای A و B تعیین میشوند، فراخوانی SFB38 را در برنامه کاربر برای این دو شمارنده از هم تفکیک میکنند.

> **برنامه برای مثالی از کاربرد دو شمارندههای A و B در ماژولCPU314-IFM :** این برنامه را آماده کرده و نحوه عملکرد شمارندهها را بر اساس آنچه در این بخش آمده بررسی کنید.



-۱- (Frequency Meter Integrated Function) فركانس متر سخت افزارى (۳-۱۰

از امکانات سخت افزاری همراه CPU در ماژول CPU314IFM میتوان برای اندازه گیری فرکانس (یا سرعت موتور) تا 10KHZ استفاده کرد. قابلیت این فرکانس متر در شکل بلوکی زیر نشان داده شده است. این قابلیت با استفاده از سخت افزار موجود در ماژول CPU314IFM که در برنامه پیکربندی سخت افزار پارامترهای آن تنظیم میشوند و بلوک نرم افزاری SFB30 که در ادامه معرفی می شود، نتیجه می دهد.

شکل بلوکی (۱۰–۲۲) نحوه ارتباط بخش سخت افزارفرکانسمتر با بلوک سیستمی SFB30 و بلوک سازمانی OB40 (مربوط به وقفه سختافزاری و ورودی سخت افزاری ماژول CPU314IFM را نشان میدهد.



شکل(۱۰–۲۲)

شکل بلوکی (۱۰–۲۳) که مجموعه امکانات سخت افزاری – نرم افزاری فرکانسمتر را نشان میدهد، شامل یک فرکانسمتر و دو مقایسه کننده (برای مقایسه حد بالا و پائین مقادیر فرکانس) بوده، و همگی ۳۲ بیتی و سخت افزاری هستند. در این طرح مقادیر موجود در مقایسه کنندهها، با مقادیر فرکانسمتر مقایسه شده و نتایج این مقایسه بصورت دو بیت اطاعات نرم افزاری به برنامه کاربر منتقل میشوند. مقادیر مقایسه کنندهها از طریق نرم افزار در آنها بار شده، و این مقادیر بار گذاری شده و همچنین مقادیر اندازه گیری شده توسط فرکانسمتر از طریق نرم افزاری در اختیار برنامه اجرائی کاربر قرار میگیرند. در این طرح سیگنالی که باید فرکانس آن سنجیده شود از طریق ورودی سخت افزاری به فرکانسمتر اعمال میشود.



بطوریکه در توضیح بالا آمده است در این طرح فقط ورودی فرکانسمتر سخت افزاری و بقیه ورودی و خروجیها همگی نرم افزاری هستند.

شکل(۱۰-۲۴) مثالی از عکس العمل عملکرد مقایسه کننده ما را در ارتباط با شمارنده نشان می هد.





در برنامه پیکربندی سختافزاری، زمان نمونهبرداری برای سنجش فرکانس قابل تنظیم و اساس اندازه گیری فرکانس در این طرح بر پایه رابطه زیر است.

$$Frequency = \frac{Number of positive edges}{Sample time}$$

درجدول(۱۰-۳) چند مثال از اندازه گیری فرکانس در ارتباط با زمان نمونهبرداری و تعداد لبههای بالارونده در یک دوره زمان نمونه برداری نشان ارائه شده است.

Sample Time	Resolution	Example of Positive Edges during 1 Sample Period	Frequency
0.1 s	The frequency can be calcu-	900	9000 Hz
	lated in 10 Hz steps	901	9010 Hz
1 s	The frequency can be calcu-	900	900 Hz
	lated in 1 Hz steps	901	901 Hz
10 s The frequency can be calcu-		900	90 Hz
	lated in 0.1 Hz steps	901	90.1 Hz

جدول(۱۰–۳)

این فرکانسمتر فقط یک ورودی سختافزاری دارد که در جدول(۱۰-۴) آدرس این ورودی نشان داده شده است.

CPU 314 IFM	Function
I 126.0	Measurement digital input

جدول(۱۰–۴)

معرفي بلوك سيستمي SFB30

بلوک سیستمی SFB30 که شمای آن در شکل(۱۰–۲۵) آمده، رابط بین فرکانسمتر سخت افزاری و برنامه کاربر است. بجز ورودی EN و خروجی ENO بقیه ورودی و خروجیهای این بلوک منطبق با ورودی خروجیهای نرم افزاری شکل بلوکی (۱۰–۲۳) آمده در مقدمه میباشند. وروری و خروجیهای این بلوک سیستمی به شرح زیر هستند.



شکل(۱۰–۲۵)

ورودیهای بلوک:

EN: بصورت BOOL، با یک شدن آن به بلوک اختیار عملکرد داده می شود PRES_U_LIMIT : برای مقدار دهی مقایسه کننده حد بالای فرکانس متر و فرمت داده آن Dint است. PRES_L_LIMIT : برای مقدار دهی مقایسه کننده حد پائین فرکانس متر و فرمت داده آن Dint است. SET_U_LIMIT : با لبه بالارونده این ورودی، مقدار اولیه مقایسه کننده حد بالای فرکانس متر بار گذاری می شود. SET_L_LIMIT : با لبه بالارونده این ورودی، مقدار اولیه مقایسه کننده حد پائین فرکانس متر بار گذاری می شود. خروجی های بلوک: SET_L_LIMI : با فرمت داده متان در عملکرد بلوک خطائی رخ دهد و یا اینکه ورودی EN یک نشود صفر می شود. FREQ : با فرمت داده Dint نشان دهنده مقدار فرکانس اندازه گیری شده FREQ : با فرمت داده Dint نشان دهنده مقدار بار گذاری شده در مقایسه کننده حد فرکانس بالا STATUS : با فرمت داده Dint نشان دهنده مقدار بار گذاری شده در مقایسه کننده حد فرکانس بالا STATUS_L: با فرمت داده BOOL نشان دهنده وضعيت مقايسه كننده حد فركانس پائين

- پیکربندی سخت افزاری برای فرکانس متر

نوع کاربری از سخت افزار موجود در ماژول CPU314IFM در صفحه شکل (۱۰–۲۶) انتخاب می شوند. این صفحه یکی از مجموعه صفحه های مربوط به Object Properties ماژول CPU314IFM با سر برگ Integrated Function است، که در اینجا نوع استفاده از سخت افزار، فرکانس متر انتخاب شده است.

Properties - CPU 314 IFM - (R0/S2)		X
Time-of-Day Interrupts Cyclic Interrupt Diagnostics/Clock I General Startup Integrated Function Cycle/Clock Memory F	Protection Retentive Men	Communication hory Interrupts
Select Integrated Function		
C Counter		
□ A □ B		
Frequency Meter		
C Positioning		
Parameter Assign	nment	
ОК	Cancel	Help

شکل(۱۰–۲۶)

در صفحه بالا با انتخاب دکمه Parameter Assignment ، صفحه (۱۰–۲۷) باز می شود. در این صفحه پارامترهای فرمانس متر تنظیم می شوند. این پارامترها شامل:

ntegrated functions	han é l. Coumbas D. Fre	guencu Meter De Jäine	x
Instance DB DB Number:	62 heck Point	Measurement Inp	2 ms 2 ms 4 ms 2 ms 1 ms
Comparator Frequency:	Below Lower Limit	Above Upper Limit	
Hardware Interrupt:	of Comparison Value	of Comparison Value	
ок			Cancel Help

شکل(۱۰–۲۷)

توضیح: در شکل(۱۰-۲۷) زمان اندازگیری فرگانس 2ms انتخاب شده و برای مثال زیر این زمان ¹s انتخاب می شود.

برنامه مثالی از کاربرد فرکانسمتر در ماژولCPU314-IFM برای اندازهگیری سرعت موتور:

سخت افزار – نرم افزار این طرح را آماده کرده و نحوه عملکرد فرکانس متر (سرعت موتور) را بر اساس آنچه در این بخش آمده بررسی کنید.



۵=۱۰) تعیین موقیت Functional Principle of the Positioning Integrated Function مقدمه

شکل بلوکی(۱۰–۲۸) روند عملکرد موقعیت یابی، برای ماشینکاری یک قطعه در موقعیتهای مشخص را نشان میدهد. اطلاعات انتقالی بین بلوکها در مسیرهای ۱ تا ۵ آمده به شرح زیر است.





- ۱) عملیات موقعیت یابی از برنامه کار بر شروع شده و فرامین لازم برای تعین موقیت به بخش سخت افزار موقیت یاب داده می شود.
- ۲) بخش سخت افزار موقعیت یاب موتور (جابجا کننده موقعیت) را راه اندازی و سرعت آن را بر پایه فرمان دریافتی از برنامه کاربر، کنترل کرده، تا جائیکه موقیت به نقطهای برسد که کلید موتور باید در آنجا خاموش شود برسد.
- ۳) سخت افزار موقعیت یاب پالسهای تعیین کنندهء موقعیت را (برای کنترل راه انداز موتور) تا رسیذن به موقعیت مطلوب از شفت انکودر (افزایشی) دریافت می *ک*ند.
 - ۴) سیگنالهای اتمام عملیات موقیت یابی از بخش سخت افزار به برنامه کاربر منتقل می شود.
 - ۵) برنامه کاربر تمام پاسخهای مربوط به ماشین کاری یک قطعه در موقعیت بدست آمده را به عملگرهای ماشین میدهد.

برای موقعیت یابی نظیر روند آمده در بالا، میتوان از امکانات سخت افزاری همراه CPU در ماژول CPU314IFM استفاده کرد. ساختار موقعیت یابی که با این امکانات میتوان بدست آورد در شکل بلوکی زیر نشان داده شده است. این ساختار با استفاده از سخت افزار موجود در ماژول CPU314IFM (که در برنامه پیکربندی سخت افزار پارامترهای آن تنظیم میشوند) و بلوک نرم افزاری SFB39 (که در بخش بعد معرفی شده است) حاصل میشود. ورودی خروجیهای آمده در شکل این بلوکی به شرح زیر است.



شکل(۱۰–۲۹)

در شکل بلوکی(۱۰–۲۹) همه ورودی و خروجیهائی که با دو نوع نوشتار نام گزاری شدهاند نرم افزاری و بقیه ورودی خروجیها سخت افزاری میباشند.

شکل (۱۰–۳۰) مثالی از تغییر مقادیر شمارنده موقعیت را در ارتباط با قطار پالسهای A و B نشان میهد. بطوریکه از روی شکل مشاهده میشود هرگاه پالس B صفر باشد با لبه بالا رونده پالس A عمل شمارش بسمت بالا انجام میشود. همچنین بر این اصول هرگاه پالس A صفر باشد با لبه بالا رونده پالس B عمل شمارش بسمت پائین انجام میشود



معرفي بلوک سيستمي SFB39

بلوک سیستمی SFB39 که شمای آن در شکل(۱۰–۳۱) آمده، رابط بین بخش سخت افزار تعین موقعیت CPU و برنامه کاربر است. بجز ورودی و خروجی ENO و ENO دیگر ورودی خروجیهای این بلوک منطبق با ورودی خروجیهای نرم افزاری شکل بلوکی (۱۰–۲۸) آمده در مقدمه میباشند. وروری و خروجیهای این بالوک سیستمی به شرح زیر هستند.



شکل(۱۰–۳۱)

معرفی ورودیهای بلوک:

EN: ورودی بصورت BOOL، با یک شدن آن به بلوک اختیار عمل داده میشود DEST_VAL: ورودی با فرمت Dint، مقدار آن موقعیت مقصد را مشخص میکند. REF_VAL: ورودی با فرمت Dint، مقدار آن محل نقطه مرجع جدید را مشخص میکند SWITH_OFF_DIFF: ورودی با فرمت Word، مقدار آن فاصله نقطه قطع کلید موتور تا موقعیت مقصد را مشخص میکند BREAK: ورودی با فرمت Byte، مقدار آن ماکزیمم سرعت جابجائی را در عملیات موقیت یابی (و یا جابجائی موقعیت بصورت دستی را) مشخص میکند. هرچه مقدار آن کم تر باشد سرعت جابجای بیشتر خواهد بود. POS_MODE2: ورودی بصورت BOOL و با یک شدن آن (در صورتیکه ورودی POS_STRT صفر باشد) عمل جابجائی موقعیت به سمت جلو اجرا میشود. از این ورودی برای تنظیم موقعیت بصورت دستی استفاده میشود.

POS_MODE1: ورودی بصورت BOOL و با یک شدن آن (در صورتیکه ورودی POS_STRT صفر باشد) عمل جابجائی موقعیت به سمت عقب اجرا میشود. از این ورودی برای تنظیم موقعیت بصورت دستی استفاده میشود.

توضیح: معمولا از ورودیهای POS_MODE2 و POS_MODE1 برای موقعیت یابی یک نقطه بصورت دستی و با جابجائی کم کم استفاده میشود. و صفر بودن این هر دو ورودی عملیات موقعیت یابی بصورت خودکار قابل اجرا است.

REF_ENABLE: ورودی بصورتBOOL و با یک شدن آن وضعیت کلید مشخص کننده نقطه مرجع (ورودیI126.2) مورد ارزیابی قرار می گیرد.

POS_STRT: ورودی بصورت BOOL، و با یک شدن آن عملیات موقعیت یابی مقصد انجام میشود. از این ورودی برای موقعیت یابی بصورت خودکار استفاده میشود.

SET_POS: بصورت BOOL، و با لبه بالا رونده آن مقدار مرجع جدید (مقدار REF_VAL) بعنوان مقدار موقعیت فعلی پذیرفته می شود و اگر خروجی POS_VALID قبلا یک نشده باشد یک خواهد شد.

معرفی خروجیهای بلوک:

ENO: خروجی با فرمت BOOL، هرگاه EN یک و عملیات بلوک بدون خطا اجرا شود این خروجی یک می شود.

ACTUAL_POS: خروجي با فرمت Dint نشاندهنده مقدار واقعى موقيت فعلى است.

POS_READY: خروجی بصورت BOOL و یک شدن آن نشاندهنده کامل شدن عملیات موقعیت یابی(یعنی رسیدن به مقدار تعین شده در ورودی DEST_VAL یس از فعال شدن Pos_Start) است.

REF_VALID: خروجی بصورت BOOL و یک شدن آن نشاندهنده رخ دادن هم آهنگی در حین اجرای عملیات موقعیتیابی اخیر است. یعنی اینکه هرگاه سیستم در جهت تعین شده (در پیکربندی سخت افزار) برای یافتن نقطه Refrens در حال حرکت باشد ، با رسیدن به نقطه مرجع (که ورودی I126.2 یک خواهد شد) مقدار خروجی ACTUAL_POS با REF_VAL با REF_VAL در صور در صورتیکه DEST_VAL با DEST_VAL برابر نباشد موقیتیابی ادامه مییابد.

توضیح: اگر ورودی REF_ENABLE یک باشد و جهت تغییر موقعیت در جهتی که برای ارزیابی نقطه مرجع در پیکر بندی سخت افزار (که بعدا خواهد آمد) انتخاب شده، باشد، در این شرایط ورودی II26.2 مورد ارزیابی قرار گرفته و با لبه بالا رونده این ورودی خروجی REF_VALID یک شده و مقدار REF_VAL شمارنده می شود. خروجی یک شده تا زمانی که مجددا ورودی REF_ENABLE از حالت صفر به یک انتقال نیابد در حالت یک باقی می ماند. با یک شدن مجدد ورودی REF_ENABLE خروجی REF VALID صفر شده و مجددا ورودی II216.2 مورد ارزیابی قرار می گیرد.

همچنین در این شرایط با لبه بالارونده ورودی I126.2 مقدار ورودی REF_VAL بعنوان مقدار مرجع جدید (موقعیت فعلی) قرار میگیردذ و اگر خروجی POS_VALID قبلا توسط ورودی SET_POS یک نشده باشد این خروجی هم یک می شود.

POS_VALID: خروجی بصورتBOOL و یک شدن آن نشان دهنده همآهنگ شدن موقعیت با نقطه مرجع (برای اولین مرتبه پس از هر Restart) است.

توضیح: پس از هر Restart، خروجی POS_VALID صفر خواهد بود، این خروجی با لبه بالا رونده ورودیI126.2 (درصورتیکه ورودی REF_ENABLE یک باشد) و یا با لبه بالا رونده ورودی SET_POS یک شده و تا Restart مجدد CPU در حالت یک باقی خواهد ماند.

معرفي ورودي خروجي سيگنالهاي سخت افزاري موقعيت سنج

در جدول(۱۰–۵) آدرسهای ورودی خروجی سیگنالهای سخت افزاری تعیین موقعیت داده شده است. این ورودی خروجیها منطبق با ورودی خروجیهای سخت افزاری آمده در شکل بلوکی (۱۰–۲۹) مقدمه میباشند.

شرح			Function when Controlling				
ییشتر ورودی خروجی ای سخت افزاری	Input/Output on the CPU		Rapid Traverse/Creep Speed Drive	Frequency Converter			
وروس براس پالس _م ان A و B	Digital input, track A	I 126.0	Connect incremental encoders for position encoding				
از شقت الكومر	Digital input, track B	I 126.1					
وریں برای کلید تعین کنندہ ثقطہ مرجع	Digital input, reference point switch	I 126.2	Connect reference point switch (e.g. BERO) for synchro- nization				
خروجى يراسراه اثداز لعسته	Digital output, creep speed	Q 124.0	Output velocities for drive	-			
و سريع موتور توسط داربور در مد ديجيتال	Digital output, rapid tra- verse	Q 124.1					
خروجی برای انتخاب جهت راه اندازی بوتور	Digital output, backward	Q 124.2	Output direction of rota- tion for drive	If frequency converter can only process positive analog signals,			
د در مد دیجیتال _و (ثالوگ	Digital output, forward	Q 124.3		specify direction of rotation for drive			
خروجى يرام راء ائداز موتور	Analog output, velocity	PQW 128	-	If frequency converter can pro-			
توسط درايور درمد أثالوك	Verocity			ify direction of rotation for drive			
				Specify velocity for drive			

جدول(۱۰-۵)

پیکربندی سخت افزاری برای فرکانس متر

نوع کاربری از سخت افزار موجود در ماژول CPU314IFM در صفحه شکل(۱۰–۳۲) انتخاب می شوند. این صفحه یکی از مجموعه صفحههای مربوط به Object Properties ماژول CPU314IFM با سر برگ Integrated Function است، که در اینجا نوع استفاده از سخت افزار، موقعیت یاب انتخاب شده است

Properties - CPU 314 IFM - (RO/S2)		
Properties - CPU 314 IFM - (R0/S2) Time-of-Day Interrupts Cyclic Interrupt General Startup Integrated Function Select Integrated Function One Function Active Interrupt Inputs Interrupt Inputs Counter Parallel Counter B Frequency Meter	Diagnostics/Clock Cycle/Clock Memory	Protection Communication Retentive Memory Interrupts
Positioning	Parameter As:	signment
ОК		Cancel Help

شکل(۱۰–۳۲)

در صفحه بالا با انتخاب دکمه Parameter Assignment ، صفحه شکل (۱۰-۳۳) باز می شود. در این صفحه پارامترهایموقعیت یابی تنظیم می شوند. این پارامترها شامل:

تعين شماره DB اختصاصي براي بلوک سيستمي SFB39

تعیین نوع سیگنال برای کنترل درایور موتور تغییر دهند موقعیت. تعیین جهت برای یافتن نقطه مرجع. تعیین فاصله برای شتاب مثبت و منفی برای درایور موتور. درموقع شروع و توقف حرکت تعین موقعیت.

Integrated functions							
Interrupts Counter Counter A Counter B Freque	ency Meter Positioning						
Instance DB	Drive Control by Means of						
DB Number:	C 4 Digital Outputs (DO)						
Automatic Update at the Scan Cycle Check Point	 ○ 2 D0 + 1 A0 ○ 1 Analog Output (A0) 						
Synchronization Evaluation of the Reference Point Switch for Direction: I Forward C Backward							
Acceleration/Deceleration Distance Acceleration Distance up to Highest Speed: [= Deceleration Distance up to Standstill]							
ОК	CancelHelp						

شکل(۱۰–۳۳)

توجه شود، ممکن است پس از انتقال پیکر بندی سخت افزار آماده شده به LED ،PLC مربوط به System failed روشن شود. دلیل این حالت آن خواهد بود که چون در پیکر بندی سخت افزار DB معرفی شده و از قبل یک چنین DB به PLC منتقل نشده است. پس از انتقال برنامه User بهمرا DB مربوطه سیستم از حالت Failed خارج می شود.

پیوست: ارتباط بلوکی سخت افزار نرم افزار در عملکرد CPU314IFM



برنامه مثالی از کاربرد سخت افزار موقعیت یابی در ماژول CPU314-IFM :

ارتباط الکریکی بین بلوکها زیر را برای بررسی تعین موقعیت با استفاده از سخت افزار CPU314IFM آماده کنید.



برنامه زیر را به منظور استفاده از سخت افزار CPU314IFM برای تعین موقعیت آماده و آنرا بررسی کنید.



تمرین ۱)با این برنامه موتور همواره شش مرحله با فاصله ۱۰۰۰۰ پالس شفت انکودر به جلو رفته سپس در شش مرحله با همان فاصله قبلی به عقب بر میگردد. بطوریکه در پایان هر مرحله ۲ ثانیه متوقف می شود.



تمرین ۲)با این برنامه موتور ۵ مرحله با فاصله ۱۰۰۰۰ پالس شفت انکودر به جلو حرکت کرده بطوریکه در پایان هر مرحله بمدت ۵ ثانیه متوقف می شود. در پایان اجرای ۵ مرحله حرکت بسمت جلو موتور کل ۵۰۰۰۰ پالس فاصله را در یک مرحله به عقب برمی گردد.



پیوست: شکل بلوکی ارتباط درایور موتور و شفت انکودر با ورودی خروجی CPU314IFM



پیوست: شکل بلوکی درایور موتور AC سه فاز



جلسه يازدهم

هدف:

- بررسی گسترش PLC با استفاده از ماژولهای IM
- بررسی گسترش PLC از طریق Profibus با استفاده از ماژول ET200M
 - بررسی بکار گیری Driver موتور AC در شبکه Profibus

۱−۱۱) گسترش PLC

مقدمه

بر روی یک ریل PLC میتوان ماکزیمم ۸ ماژول برای تبادل اطلاعات نصب کرد این ماژولها از اسلات ۴ تا ۱۱ هر ریل نصب میشوند برای استفاده از ماژولهای بیشتر، لازم است، PLC گسترش یابد. برای این کار میتوان با کمک ماژولهائی که برای همین منظور وجود دارد، ریلهای جدیدی به PLC اضافه کرد.

در PLCهای Step7 با کمک ماژولهای S-R 365 IM میتوان فقط یک ریل (به نام (U(1)) به ریل مرکزی (U(0) اضافه کرد. برای این کار، لازم است بر روی اسلات شماره ۳ هر یک از ریلهای (U(0) و (U(1)، یک ماژول 365 IM نصب و سپس با کابل مربوطه بهم وصل شوند. در سمت چپ شکل (۱۱–۱) نحوه ارتباط این دو ماژول نشان داده شده است. ماکزیمم طول کابل ارتباطی بین این نوع ماژولهای IM **یک** متر است. برای اضافه کردن ریل جدید به پیکربند سخت افزار، از ریل موجود در پوشه نشان داه شده در سمت راست شکل(۱–۱) که متعلق به پنجره کاتولگها از برنامه پیکربندی سخت افزار است، استفاده میشود.



در PLCهای Step7، برای اضافه کردن ریلهای بیشتر ماکزیمم تا سه ریل، از ماژولهای IM 360 و IM 361. استفاده میشود. نحوه این گسترش در سمت چپ شکل(۱۱–۲) نشان داده شده است. همه ماژولهای مورد نیاز برای گسترش یک PLC در پوشه نشان داه شده در سمت راست شکل(۱۱–۲) در دسترس هستند. این پوشه یکی از پوشههای پنجره کاتولوگها از برنامه پیکربندی سخت افزاز است. ماکزیمم طول کابل ارتباطی بین این نوع ماژولهای IM ده متر است.



شکل(۲-۱۱)

۱–۱–۱)روند پیکربندی سخت افزار PLC با دو ریل : برای پیکر بندی سخت افزار یک PLC با دو ریل، با استفاده از دو ماژول IM365 به ترتیب زیر عمل میشود.

ا- یک پروژه ایجاد میشود.

۲- برنامه پیکربندی سخت افزار پروژه ایجاد شده باز ، و سپس در کنار ریل صفر ((U(0) یک ریل جدید(U(1)) اط مسیر نشان داده شده در شکل(۱۱–۵) به پیکر بندی سخت افزار اضافه میشود.

۳- در اسلاتهای شماره سه هر یک از ریلهای صفر و یک، یک ماژول IM365 اضافه شود. با این کار دو ریل با یک خط بهم وصل می شوند. این خط بیانگر لازمه ارتباط دو ریل توسط کابل مربوطه است.

۴- ماژولهای سیگنال مورد نیاز در ریلهای U(0) و U(1) پیکر بندی میشوند.

در اینجا روند پیکربندی سخت افزار PLC با دو ریل پایان مییابد، حال میتوان پیکربندی آماده شده را به PLC منتقل (Down Load) کرد.

توضیح: چون در آزمایشگاه ماژولهای سخت افزاری IM365 موجود نیست از انتقال این پیکر بندی به PLC صرفنظر می شود. **سئوال:** در پایان این بخش از پیکر بندی بررسی شود که آیا می توان ریل جدید دیگری به این پیکر بندی اضافه کرد یا خیر؟

۱–۱–۲) روندپیکربندی سخت افزار PLC با چهار ریل: برای پیکر بندی سخت افزار یک PLC با چهار ریل، با استفاده از ماژولهای IM360 و IM361 به ترتیب زیر عمل می شود.

۱- یک پروژه ایجاد میشود.

۲- برنامه پیکربندی سخت افزار پروژه ایجاد شده باز ، و در کنار ریل صفر سه ریل جدید (ریلهای یک تا سه) از مسیر نشان داده شده در شکل(۱۱–۱) به پیکربندی سختافزار اضافه میشود.

۳- در اسلات شماره سه ریل صفر یک ماژولIM360 و در هر یک از اسلاتهای شماره سه ریلهای دیگر یک ماژول
 IM361 اضافه می شود. با این کار ریل ها با یک خط بهم وصل می شوند. این خط بیانگر لازمه ار تباط ریل ها توسط کابل های مربوطه است. ماژول های IM360 و IM361 در پوشه نشان داده شده در سمت راست شکل (۱۱-۶) وجود دارند.

۴- ماژولهای مورد نیاز در پروژه در هر یک از این ریلها پیکربندی میشوند.

در اینجا روند پیکربندی سخت افزار PLC با چهار ریل پایان یافته است، حال میتوان آنرا به PLC منتقل (Down) (Load

توضیح ۱: چون در آزمایشگاه تجهیزات سخت افزاری فقط برای اضافه کردن یک ریل موجود است از Down Load کردن این پیکربندی صرفنظر کرده، و در پایان بررسی شود که آیا میتوان ریل دیگری به این پیکربندی اضافه کرد یا خیر؟ برای ادامه کار ریل های دو و سه را از پیکربندی حذف و ماژولهای مورد نیاز در برنامه را به دو ریل باقی مانده اضافه کنید، سختافزار دو ریل باقیمانده در آزمایشگاه موجود بوده که در اسلاتهای شماره ۳ ریلهای صفر و یک آنها، به ترتیب ماژول IM360 و IM361 نصب و این دو ماژول توسط کابل مربوطه به هم وصل و همه ماژولهای آمده در شکل(۱۰–۳) بر روی آنها نصب . پیکربندی آماده شده را به PLC منتقل (Down Load) کنید



شکل(۱۱–۳)

توضیح ۲: با انتخاب میله نام هر یک از ریلها، پنجره خاص آن ریل مانند جدولهای آمده در شکلهای(۱۱-۴ و ۱۱-۵) در پائین صفحه پیکربندی سخت افزار باز میشوند. که آدرسهای پیش فرض هر یک از ماژولهای پیکربندی شده در آن ریل در ستون مربوطه نمایش داده میشوند.

آدرسهای ماژولهای ریل صفر (UR(0:

(⇒	(0) UR					
Slot		Module	 Order number	Fi	MPI ad	I ad	Q
1		PS 307 2A	6ES7 307 1BA00 0AA0				
2	8	CPU314(1)	6ES7 314-1AE04-0AB0		2		
3		IM 360	6ES7 360-3AA01-0AA0			2000	
4		DI16xDC24V	6ES7 321-1BH82-0AA0			01	
5		D016xDC24V/0.5A	6ES7 322-1BH10-0AA0				45

شکل(۱۱–۴)

آدرسهای ماژولهای ریل یک(UR(1):

	(🔶 (1) UR					
	Slot	🚺 Module 🛛	Order number	Fi	М	I add	Q add
	1	🚦 PS 307 2A	6ES7 307-1BA00-0AA0				
	2						
	3	IM 361	6ES7 361-3CA01-0AA0			2004	
l	4	🚦 D18/D08xDC24V/0,5A	6ES7 323-1BH01-0AA0			32	32
-							

شکل(۱۱–۵)

برنامه برای بررسی سخت افزار پیکر بندی شده در بالا،

برای بررسی سخت افزار آماده شده، برنامه ارائه شده در زیر را اجرا کنید.

Network 1:1	[itle:		
	EN MC	ENO	
IBO) - IN	OUT -QB32	
Network 2:3	Title:		
	M	DVE	
	En	ENO	
IB32	2 - IN	OUT - QB4	

مثال: برای سخت افزار پیکربندی شده، برنامهای آماده کنید تا دو عدد مثبت باینری را از ورودهای IB0 و IB32 دریافت کرده، ازیکدیگر کم و حاصل را بصورت BCD در خروجی QW4 نشان دهد. اگر نتیجه صفر شد، خروجی Q32.0 و اگر نتیجه منفی شد خروجی Q32.1 و اگر نتیجه بزرگتر از صفر شد، خروجی Q32.2 یک شود.



۲-۱۱) گسترش PLC از طریق PROFIBUS (با ماژول ET 200M)

مقدمه

انتقال دادهها، وضعیتها و پارامترها بین کنترل کننده مرکزی و تجهیزات Plant در سیستمهای خودکار میتواند بصورت سریال، و از طریق شبکههای طراحی شده برای همین منظور، انجام شود. یکی از این شبکهها PROFIBUS- DP است که تجهیزات شرکت زیمنس با آن کار میکنند. این شبکه در پیوست ۱ این بخش معرفی شده است.

در این بخش از بررسی، میخواهیم (امکانی برای استفاده بیشتر از ۸ عدد ماژول سیگنال برای PLC) ریلی به ریل مرکزی که CPU بر روی آن قرار دارد اضافه شود تا از طریق شبکه PROFIBUS ارتباط بین CPU و ماژولهای ریل اضافه شده برقرار شود. برای این کار لازم است CPU ای که روی ریل صفر نصب میشود دارای قابلیت ارتباط با شبکه PROFIBUS باشد.

برای پیکر بندی PLC، به منظور گسترش ظرفیت تعداد ماژولهای مورد نیاز در یک پروژه، با استفاده از شبکه به روش زیر عمل میشود.

پیکر بندی سخت افزار

۱- پس از ایجاد یک پروژه، در اسلات دو ریل صفر آن یک CPU که دارای قابلیت ارتباط با PROFIBUS است پیکر بندی می-شود. ماژولهای منبع تغذیه و دیگر مازولهای ورودی و خروجی مورد نیاز مانند شکل(۱۱-۶) به پیکربندی این ریل اضافه می-شوند. CPUئی که در آزمایشگاه برای ایجاد شبکه PROFIBUS استفاده می شود CPU315-2DP است که بصورت یک پارچه سخت افزار ایجاد شبکه را بهمراه خود دارد.

توضیح: ماژولهای CPU هائی که سخت افزار مربوط به ایجاد شبکه شدن را بصورت یک پارچه با خود ندارند، میتوان با تهیه و نصب ماژولی که این قابلیت را برای شبکه شدن آنها فراهم میکند استفاده کرد.

🙀 HW Config - SIMATIC 300 Station
Station Edit Insert PLC View Options Window Help
🕪 SIMATIC 300 Station (Configuration) test_91
Image: Constraint of the second sec
(0) UR
Slot Module Order number Firmware MPI address I add
1 S 307 2A 6ES7 307-1BA00-0AA0
2 CPU315-2DP(1) 6ES7 315-2AG10-0AB0 V2.0 2
X2 DP 2047
3
4 DI16xDC24V 6ES7 321-18H02-04A0 01 /
5 D016xDC24V/0.5A 6ES7 322-18H01-0AA0
Press F1 to get Help.

شکل(۱۱–۶)

۲- روی ردیف X2 از ماژول CPU315-2DP کلیک راست کرده و در منوی باز شده که تصویر آن در سمت چپ شکل(۲۰۱۰) آمده است گزینه Object Properties انتخاب می شود.
 توضیح: همچنین برای باز کردن این صفحه می توان پس از انتخاب ردیف X2 ، از منوی Edit گزینه Object Properties را انتخاب کرد

	Properties - DP - (R0/S2.1)	
🖳 HW Config - SIMATIC 300 Station	General Addresses Operating Mode Configuration	
Station Edit Insert PLC View Options Window Help	Short Description: DP	
D 🔊 🖓 🗣 🤮 🖻 🖻 🏙 🏙 🗊 🗖 器		
Blg SIMATIC 300 Station (Configuration) test_91		
🚍 (0) UR	Order No.:	
	Name: DP	—
χ^2 DP	laberta an	
3 Copy Ctrl+C		
4 DI16xD Pasce Cm+v	Address: 2	
6 Insert Object	Notices: 2	
Disconnect Master System		
	Comment:	
Edit Symbolic Names		
Slot Module Product Support Information Ctrl+F2		
	OK Cancel H	ielp
	· · · · · · · · · · · · · · · · · · ·	

شکل(۱۱–۷)

۳- در جلو کلمه Networked در بخش Interface صفحه باز شده (درصفحه با سربرگ General) کلمه No دیده می شود، کلمه NO در اینجایعنی در این برنامه شبکه ای ایجاد نشده است. برای ایجاد شبکه جدید دکمه Properties فعال می شود. با فعال شدن این دکمه، مجموعه صفحه های Properties PROFIBUS Interface باز می شوند. در این مجموعه که دارای دو سربرگ است با انتخاب سربرگ Parameters صفحه نشان داده شده در شکل (۱۱-۸)نمایان می شود.

Properties - PROFIBUS interface DP (R0/S2.1)	
General Parameters	
Address:	
Subnet:	
not networked	New
	Properties
	Delete
ОК	ncel Help

شکل(۱۱–۸)

۴- . آدرس نشان داده شده در بالای این صفحه (شماره 2) مربوط به آدرس پیش فرض CPU برای شبکه PROFIBUS است، که در اینجا این آدرس پیشفرض حفظ میشود. برای ایجاد شبکه جدید دکمه NEW روی این صفحه انتخاب میشود. با این انتخاب، دو صفحه آمده در شکل(۱۱–۹)، در دو سربرگ جداگانه قابل نمایش هستند. در این دو صفحه تمام تنظیمات مورد نیاز برای شبکه قابل انجام است که در اینجا تنظیمات پیشفرض حفظ خواهد شد.

Properties - New s	ubnet PROFIBUS	Pr	operties - New subne	t OFIBUS		
General Network Se	ettings		General Network Settings			
Name: S7 subnet ID:	FROFICUSO) [009C - [0014		Highest PROFIBUS Address:	126 💌	🥅 Change	Options
Project path: Storage location of the project: Author:	test_91 E:\Siemens\Step7\s7proj\test_91		Transmission Rate:	45.45 (31.25) Kbps 93.75 Kbps 187.5 Kbps 500 Kbps		
Date created: Last modified: Comment:	20.12.2008 17:16:10 20.12.2008 17:16:10		Profile:	1.5 Mbps 3 Mbns DP Standard Universal (DP/FMS) User-Defined		
ОК	Cancel Help	Ē	OK			Bus Parameters Cancel Help
	(9-1)	ل(شک			

۵- در صورت تائید تنظیمات این صفحات توسط دکمههای OK ضفحات باز شده، یک شاخه ارتباطی شبکه مانند شکل(۱۱ ۱۰) در صفحه پیکربندی سختافزار ظاهر می شود.

HW Config - SIMATIC Ration Edit Insert PLC	300 Station Vew Options Window Help Balls and The C	1 😢 👷					
2 2010 2 2010 2 2 000 3 000 4 00000 5 000000 5 000000 5 000000 5 0000000 5 000000 5 0000000000	PROFILIES	1): DP master syst	ees [1])		W Distribution W ET 2005 W ET 2005 W ET 2005 W M 103 <
<						-	IM 153/2 F0 IM 153/2 F0 IM 153/2 F0
(0) UR							🐵 🚆 IM 153-2 FO
Slot Module	Order number	Ferrovare	MPI address	I address	Q add	C.	 M 153-3 M 153-3
1 2 CPU315-20P(32 4 DP 3 4 DP 3	6ES7 315-24610-048	0 V2.0	2	2840~		< 68 Bi S	57 153 1AA03 0/80 at interface module for 57 300 Mt, FM 350 to FM 352, FM 355,
						- 0	P 340 to LP 342/2, module

شکل(۱۱–۱۰)

۶- حال از زیر مجموعه ET 200M، از پوشه شبکه PROFIBUS DP، در پنجره کاتولوگها المان IM153-1AA03-0XB (که سخت افزار آن در آزمایشگاه مو.جود است) انتخاب و به شاخه ارتباطی وصل می شود. با این عمل مجموعه صفحات شکل(۱۱–۱۱) که دارای دو سر برگ است باز می شوند.

Properties - PROFIBUS interface IM 153-1	\mathbf{X}
General Parameters	
Address:	
Transmission rate: 1.5 Mbps	
Subnet:	
not networked PROFIBUS(1) 1.5 Mbps	New
	Properties
	Delete
Са	incel Help

شکل(۱۱–۱۱)

- ۲- آدرسی که در بالای صفحه با سربرگ Parameter مشاهده می شود مربوط به آدرس ما ژول نET 200M نصب شده در شبکه است. این آدرس را می توان از 1 تا تا 125 (بجز 2 که قبلا برای CPU بر گزیده شده) انتخاب کرد. توجه کنید هر آدرسی که در اینجا انتخاب می شود، لازم است همان آدرس توسط Dip Switch ی که روی ما ژول IM153-1 وجود دارد بصورت سخت افزاری برای این ما ژول تنظیم شود.
- ۸ اگر در صفحه با سربرگ Parameter دکمه Properties انتخاب شود صفحههای شکل(۱۱–۱۲) قابل مشاهده هستند. در
 این دو صفجه تنظیمات مور نیاز برای ماژول IM153-1 انجام می شود. توجه شود، تنظیمهای ایـن صفحهها بایـد با
 تنظیمهای نظیر آنها که در شکل(۱۱–۹) برای ماژول CPU آمده است هماهنگ باشند.

Properties - PROFI	BUS	Pr	operties - PROFIBU	S	
General Network S	ettings	Ē	General Network Settin	ngs	
Name: S7 subnet ID:	PROFIBUS(1) 009C - 0014		Highest PROFIBUS Address:	126 💌 🗖 Change	Options
Project path: Storage location of the project: Author:	test_91 E:\Siemens\Step7\s7proj\test_91		Transmission Rate:	45.45 (31.25) Kbps 93.75 Kbps 187.5 Kbps 500 Kbps	
Date created: Last modified: Comment:	20.12.2008 17:16:10 20.12.2008 17:16:10		Profile:	15 Mbps 3 Mhos DP Standard Universal (DP/FMS) User-Defined	Bus Parameters
ОК	Cancel Help	Ē	OK		Cancel Help

شکل(۱۱– ۱۲)

 ۹- با تائید تنظیمهای صفحهها باز شده توسط دکمههای OK، یک ریل جدید با شمار(۱) در پنجره پیکربندی سخت افزار ایجاد می شود که این ریل به شاخه ارتباطی شبکه وصل خواهد بود. شکل این نوع ریل ها در قسمت بالای شکل(۱۱–۱۳) آمده است. با انتخاب میله نام ریل ایجاد شده، پنحره محل نصب ماژولها این ریل مانند قسمت پائین شکل(۱۱– ۱۳) باز می شود. ماژولهای مورد نیاز برای ریل جدید از زیر مجموعه IM153-1AA03-0XB0 انتخاب و در اسلات مربوطه نصب می شود.

توضیح۱: تمام ماژولهای نصب شده بر روی ریل (۱) لازم است در پیکر بندی آورده شود، ولی این موضوع برای ریل (۰) ضرورتی ندارد و می توان فقط ماژولهای مورد نیاز، در پیکربندی آورده شوند. همچنین لازم است **توجه شود**. در پایان پیکربندی سخت افزار تنظیمات انجام شده به PLC منتقل(Down Load) شود.

در اینجا کار پیکربندی گسترش PLC با PROFIBUS برای یک ماژول ET 200M پایان یافته است، در صورت نیاز می توان به همین روش شبکه را گسترش داد.

توضیح ۲: با انتخاب میله نام هر یک از ریلها، پنجره خاص آن ریل در پائین صفحه باز میشوند. که آدرسهای پیش فرض هر یک از ماژولهای پیکربندی شده در ستون مربوطه خواهد آمد.



شکل(۱۱– ۱۳)

بررسی تغییر آدرس پیش فرض ماژول ها سیگنال: ماژول ها سیگنالی که در کنار بعضی از CPU ها پیکر بندی می شوند قابلیت آنرا پیدا میکنند که آدرس های پیش فرض آنها تغییر یابند. در اینجا برای این بررسی، بر روی یکی از ماژول های ورودی و یا خروجی کلیک راست کرده و از منوی باز شده گزینه Object Properties را انتخاب کنید. سپس در مجموعه صفحاتی که باز می-شود بررسی کنید که آیا آدرس های پیش فرض تعیین شده برای ماژول های قابل تغییر هستند و یا خیر؟

طول کابل شبکه PROFBUS: طول کابل بین ریلها به فرکانس انتقال بستگی دارد و در کمترین فرکانس ۱۲۰۰ متر و در بالا-ترین فرکانس ۱۰۰ متر است. در زیر فرکانس و طول کابل متناسب با آن ارائه شده است.

PROFIBUS - DP(SIMATIC NET) 11.6 k Bit/s = max. 1200 m 111.2 k Bit/s = max. 1200 m 113.75 k Bit/s = max. 1200 m 187.5 k Bit/s = max. 1000 m 500 k Bit/s = max. 400 m 1.5 M Bit/s = max. 200 m 12.0 M Bit/s = max. 100 m FIP1.0

برنامهای برای بررسی سخت افزار پیکر بندی شده در این بخش:



تمرین: برای سخت افزار پیکربندی شده در بالا، برنامهای آماده کنید، تا دو عـدد یـک بـایتی را از ورودیهـای IB0 و IB2 بصورت باینری خوانده، در هم ضرب و حاصل را منفی کرده و نتیجه را به صورت BCD به خروجی QW4 منتقل کند. اگر حاصلضرب دارای Overflow بود خروجی Q0.0 و اگر عمل تبدیل BCD درست انجام نشد، خروجی Q0.1 فعال شود. برنامه:



PROFIBUS موتور Micro master 420) AC موتور Driver)در شبکه Micro master 420) موتور معدمه

داریور موتور AC شرکت زیمنس (Micro Master 420) را میتوان به سه طریق برای کنترل موتور AC بکار گرفت. این سه طریق به شرح زیر میباشند.

(۱) با استفاده از کلیدهای تابلوهای BOP^۱ و یا AOP⁷ که بر روی داریور نصب می شوند می توان پارامترهای داریور را برای موتور تنظیم و سیگنالهای کنتزلی آنرا تغییر داده و وضعیت عملکرد درایور را از طریق نشاندهنده این تابلوها مشاهده کرد. تصویرهای این تابلوها در شکل(۱۱–۱۴) نشان داده شده است. برای اطلاع از نحوه بکار گیری هر یک از این تابلوها می توان به Manual داریو مربوطه و یا به خلاصهای از آن که در پیوست ۲ آمده است رجوع کرد.





- ۲) با استفاده از سیگنالهای سخت افزاری ورودی و خروجی (دیجیتال و آنلوگ) داریور می توان متصل به موتور را کنترل و با کمک تابلوهای BOP و یا AOP می توان پارامترهای آنرا تنظیم و وضعیت عملکرد آنرا مشاهده کرد. در این روش برای تغییر سیگنالهای کنترلی داریور می توان از طریق برنامه یک PLC که سیگنالهای خروجی آن PLC به سیگنالهای ورودی درایور وصل می شوند عمل کرد. برای اطلاع از سیگنالهای ورودی و خروجی (دیجیتال و آنالوگ) درایور و نحوه بکار گری آنها می توان به Manual این داریور مراجعه کرد.
- ۳) با استفاده از شبکه PROFIBUS میتون از طریق برنامه PLC پارا مترهای دارایور را برای موتور تنظیم کرده و سیگنالهای کنترلی آنرا تغییر و وضعیت عملکرد موتور را به برنامه PLC انتقال داد. در شکل(۱۱–۱۵) تصویر ماژول واسط PROFIBUS Operational Board که از آن برای ارتباط درایور موتور AC (Micro Master 420) AC به شبکه PROFIBUS استفاده میشود، نشان داده شده است.



شکل(۱۱– ۱۵)

Basic Operator Panel

Advance Operator Panel

در جلوی این ماژول ۷ عدد Dip Switch (شماره 2 شکل ۱۱– ۱۵) وجود دارد که این Dip Switchها برای تنظیم آدرس این ماژول در شبکه است. این آدرس بصورت سخت افزاری بر اساس مقادیر نشان داده شده در جدول زیر تنظیم می شود. این آدرس باید منطبق بر آدرسی باشد که در برنامه پیکربندی سخت افزاری PLC برای این گره^۱ تنظیم می شود.

The PROFIBUS address can be set to between 1 and 125, as shown in the following table, on DIP switches 1 to 7.

Switch number:	1	2	3	4	5	6	7
Add to address:	1	2	4	8	16	32	64
Example 1: Address = 3 = 1 + 2	on	on	off	off	off	off	off
Example 2: Address = 88 = 8 + 16 + 64	off	off	off	on	on	off	on

چراغ LED روی ماژول که میتواند در سه رنگ بصورت پیوسته و یا چشمکزن روشن شود وضعیت ماژول را در ارتباط بـا شـبکه مشخص مِیکند. جدول زیر وضعیت LED را در رابطه با عملکرد ماژول در شبکه PROFIBUS شرح میدهد

LED display on PROFIBUS-DP communication board

LED	Diagnostic information
off	No power supply
red, flashing fast	Invalid PROFIBUS address on DIL switch (126/127 is invalid) or hardware fault or software error
red on	Startup and no communication (yet) with the inverter or new communication board configuration, after modification of a board parameter If this status is steady, then the inverter or PROFIBUS optional board is defective.
orange flashing	Communication link to inverter has been established No connection to PROFIBUS, e.g. PROFIBUS connector is not inserted or PROFIBUS master disconnected.
orange on	Communication link to inverter and connection to PROFIBUS have been established, but no cyclical data exchange is taking place.
green flashing	Cyclical process data exchange in progress, but setpoints invalid (control word = 0), e.g. because SIMATIC master is in "Stop" state
green on	Cyclical process data exchange in progress and o.k.

II-۳-۱۱) پیکر بندی سخت افزار درایور در PLC:

هر وسیلهای که بتواند با Step7 کار کند لازم است کاتولوگ آن در پنجر کاتولوگهای نرم افزار پیکر بندی سخت افزار Simatic Manager موجود باشد، تا بتوان از این کاتولوگ برای پیکر بندی سخت افزار آن وسیله استفاده کرد. اگر چنین کاتولوگی در پنجره کاتولوگها موجود نباشد، براحتی میتوان فایل آن را (که معمولا در CD مدارک همراه وسیله موجود است) در نرم افزار نصب کرد. این نصب از طریق انتخاب گزینه Install GSD File از منوی Option که مسیر آن در شکل(۱۱–۱۶) نشان داده شده است انجام میشود.

🙌 Station	Edit Insert PLC View	Options Window Help		
🗅 😅 🖁	- 6 9 1 🚑 🕒	Customize	Ctrl+Alt+E	
		Specify Module		
😑 (0) UR		Configure Network		
1		= Symbol Table	⊂trl+Alt+T	
		Report System Error		
	LPU 315-2 DP			
	DP	Edit Catalog Profile		PI
3		Update Catalog		Y
4	DI16xDC24V	Tostall HW/ Lindates		
5	D016xDC24V/0.5A	Tostall GSD File		💼 (1) IN
		Anaton GDD Them		
6				

شکل(۱۱ – ۱۶)

پس از نصب فایل GSD داریور Micromaster420 در مسیر پیش فـرض نـرم افـزار، کـاتولوگ آن در مسـیر نشـان داده شـده در شکل(۱۱–۱۷) آشکار میشود. **نوجه شود** این فایل قبلا برای نرم افزارهای آزمایشگاه نصب شده است. MICROMASTER4→SIMOVER → Additional → PROFIBUS- DP

1 HW Config - [SIMATIC 300 Station (Configuration) test_HMI_Drive]						
14 Station Edit Insert PLC View Options Window Help						
D 📁 📽 📲 🧏 🎒 🖪 🖻 💼 🖬 🏛 🛍 📳 🗁 🚼 😒						
Image: Constraint of the second sec		Erofile: St Profile: St PROF Au C	tandard IBUS DP Othes THE Devices THE De			
شکل(۱۱– ۱۷)						

برای پیکربندی این ماژول با موس کاتولوگ MICROMASTER4 بر روی شبکه PROFIBUS منتقـل مـیشـود. بـرای تنظـیم آدرس و پارامترهای این ماژول به روشی که برای ماژول ET200M تنظیم شد برای این ماژول هم عمل میشود، با این تفاوت کـه در اینجا آدرس این گره عدد 3 انتخاب میشود. این آدرس باید منطبق بر آدرسی باشد که بصورت سخت افزاری بـرای ایـن گـره^۱ توسط Dip Switch های روی ماژول واسط تنظیم میشود.

پس از انتقال ماژول از پنجره گاتولوگها به صفحه پیکربندی سخت افزار و تنظیم آدرس و پارامترهای آن، ماژول بصورت شکل(۱۱–۱۸) در برنامه پیکربندی سختافزار ظاهر میشود.



درایور موتور AC (Micro master420) در شبکه PROFIBUS مانند یک مجری^۱ کار میکند. و دستورها خود را از مدیر^۲ شبکه که در اینجا PLC است دریافت میکند. در شبکه PROFIBUS برای تبادل دادهها و پارامترها بین درایور و PLC از فریمی به شکل زیر استفاده میشود.



در این روش، تبادل دادههای مربوط به کنترل فرآیند بصورت دورهای و تبادل دادههای مربوط به پارامترهای سیستم بصورت دورهای و یا غیر دورهای قابل اجرا هستند در ساختار این فریم PPO^۳ شامل دادهها فرآیند(PZD) مانند فرمانهای کنترلی، مقادیر مطلوب، وضعیت سیستم و مقادیر واقعی فرآیند و پارامترهای سیستم (PKW) مانند پارامتر حد بالا و پائین سرعت وغیره در درایور است.

در داریورها پنج نوع PPO برای دسترسی به دادههای فرآیند و پارامترها به صورت شکل زیر تعریف شدهاند کـه ایـن PPO بـه دو شکل کلی تقسیم میشوند.



الف) فقط برای تبادل دادههای فرآیند بطول ۲ و ۶ کلمهای مثل PPO3 و PPO4 در شکل بالا است ب) برای تبادل دادههای فرآیند بطول ۲، ۶ و ۱۰ کلمهای بـه همـراه تبـادل پارامترهـای سیسـتم مثـل PPO2، PPO1 و PPO5 درشکل بالا است.

انتخاب PPO بر اساس استفاده از هر یک از قابلیتهای موجود در ماژول PROFIBUS Operational Board انجام می شود.

[°] Master

^rParameter Process data Object

ماژول استفاده شده برای درایور آزمایشگاه تنها از PPO1 و PPO3 پشتیبانی می کند. در اینجا برای بررسی کار درایور از طریق شبکه PROFIBUS از PPO3 که فقط برای تبادل دادههای کنترلی فرآیند مربوط به درایور و بطول دوکلمه است انتخاب میشود. پارامترها مورد نیاز در این درایور قبلا از طریق تابلو BOP تنظیم شدهاند.

برای انتخاب PPO3 و انتقال آن به برنامه پیکربندی، در ابتدا میله عنوان مازول Micro Master 420 در پنجرهی پیکربندی انتخاب می *شود* با این عمل *پنجره* ماژولهای مربوط به این داریور در پائین صفحه آشکار می شود. سپس از زیر مجموعه کاتولوگهای مازول Micro Master 420 در پنجره کاتولوگها، با موس PPO3 انتخاب و به ردیف اول پنجره ماژولهای درایور انتقال داده می شود. در اینجا پیکربندی *Micro Master420 د*ر شبکه PROFIBUS پایان می یابد. نتیجه ایس عملیات در شکل(۱۱–۱۹)نشان داده شده است. توضیح اینکه آدرسهای ورودی خروجی برای انتقال دادههای دادههای Micro Master 420 در ستون مربوط به آدرس در ردیف PPO3 انتفال داده شده امده می شود.

توجه شود شماره آدرسهای PIW و PQW در برنامه این مثال بر طبق شمارهها آدرسهای موجود در جـدول درایـور بـرای PPO3 تنظیم میشوند. این شماره های بستگی به ترتیب و تعداد ماژولهای پیکربندی شده در پروژه دارد.



شکل(۱۱–۱۹)

توضیح: خلاصه مجموعه پارامترهائی که در Micro Master 420 قابل تنظیم هستند در پیوست ۲ این بخـش ارائـه شـده است. برای تبادل دادههای فرآیند و پارامترهای سیستم از طریق PROFIBUS ، لازم است پارامتر 6=P0700 و پارامتر 6=P1000 درایـو (از طریق BOP) تنظیم شده باشند برای دستیابی به اطلاعات بیشتر در ارتباط با نحوه عملکرد داریور Micro Master 420 و نحوه تنظیم پرامترهای آن به Manual مربوطه و یا به خلاصهای از آن که در پیوست ۲ آمده است رجوع شود. PROFIBUS-DP و PLC و Micro Master 420 و Micro Master از طريق PLC او PLC او PROFIBUS-DP و

دادههای کنترلی فرایند برای درایور موتورر AC در چهار جدول پیوست ۳ ارائه شده است. برنامه زیر بر اساس این داده ها طراحی شده است. در این برنامه، در شبکههای یک و دو بر اساس جدولهای پ۳-۱ و پ۳-۲ پیوست ۳ دادههای کنترلی از PLC به درایور و در شبکههای سه و چهار دادههای وضعیت درایور از درایور به PLC منتقل میشوند. این برنامه را آماده کرده و به PLC منتقل کنید.

توضیح: چون نعداد ماژولها روی ریلهای ET200M در دو سیستم موجود در آزمایشگاه متفاوت هستند ، این تفاوت باعث تفاوت در آدرسهای ورودی خروجی مربوط به درایور میشود. در موقع نوشتن این برنامه توجه شود تا از آدرسهای صحیح استفاده شود.



در این برنامه در شبه اول کلمه کنترلی درایو از ورودی IWO به آن منتقل می شود. این کلمه کنترلی برای راهاندازی و توقف درایو استفاده می شود. کلمه کنترلی دوم دریور که در شبکه دوم و یا سوم از IW2 به آن منتقل می شود. مربوط به تنظیم سرعت موتور است، بطوری که اگر بیت I2.7 صفر باشد در شبکه دوم مقدار (Iw2(int) به درایور منتقل شده و با راه اندازی داریور موتور در جهت CW با سرعتی متناسب با مقدار عدد تنظیم شده در IW2 دوران می کند و اگر بیت I2.7 یک باشد در شبکه سوم مقدار I2.7 منفی شده و سپس به درایور منتقل می شود و با این انتقال موتور در جهت CCW و با سرعتی متناسب با مقدار عددی تنظیم شده در IW2 چرخش خواهد کرد.

پس از آماده کردن برنامه و انتقال آن به PLC، وروری IW0 را به **مقدار 047E** و ورودی IW2 را به **مقدار 4000** تنظیم کنید. پس از این تنظیم **ورودی I1.0 را از وضعیت 0 به 1 تغییر داده** با این تغییر اگر همه کارها درست انجام شده باشد باید موتور شروع به حرکت نماید. در صورت درست عمل کردن موتور مقدار IW2 را به **مقدار C000 تغ**ییر داده و نحوه حرکت موتور را با این تغییر بررسی کنید. همچنین تغییر وضعیت ورودی I0.3 را روی عملکرد موتور بررسی کنید.

توضیح: در صورتیکه برای درایور خطائی رخ دهد، برای راهاندازی مجدد حتماً لازم است به خروجی PQ257.7 یک لبه بالارونده (در اینجا از طریق ورودی 1.17) ارسال شود.

۲۱ –۴) استفاده از SFC14 و SFC15 برای انتقال دادهها در شبکه PROFIBUS_DP

برای انتقال همزمان دادههای بیشتر از چهار بایت مربوط به یک گره^۱ در شبکه PROFIBUS_DP، از توابع (SFC15 و SFC15(DPRW_DAT) استفاده می شود. بطوریکه از SFC15 برای انتقال داده از مدیر به مجری و از SFC14 برای انتقال داده از مجری به مدیر استفاده می شود.

2 General Definition of PROFIBUS-DP

Definition

PROFIBUS is an international, open field bus standard which is used widely in the fields of production and process automation. The bus is guaranteed brand-neutral and open by virtue of its compliance with international standard IEC 61158.

PROFIBUS-DP is a PROFIBUS communication profile optimized for high-speed, time-critical data transmission at field level using low-cost connections. PROFIBUS-DP is a suitable substitute for conventional, parallel 24 V signal transmission systems in manufacturing, as well as for analog 4..20 mA signal transmission systems in process automation.

PROFIBUS is a multi-master system, in other words, it is a bus on which several automation, engineering or visualization systems can operate together with the associated distributed field devices. In the context of PROFIBUS, master and slave devices are defined as follows:

- Master devices control data traffic on the bus; they are also referred to as "active" nodes. A master may transmit messages without prior receipt of an external request provided that it has bus access authorization (token). There are two classes of master:
 - Class 1 master:

These are central automation stations (e.g. SIMATIC S5, S7 and SIMADYN D) which exchange information with slaves in predefined message cycles.

RS-485 transmission system

Criteria such as high transmission speed and simple, low-cost installation are of critical importance in the selection of the transmission system. The RS-485 requires a screened copper cable with twisted-pair wires.

The transmission speed can be selected within the 9.6 kbaud to 12 MBaud range. It is set globally for all devices on the bus during system start-up.

General information about RS-485 transmission installation

All devices are connected in a bus structure (line). Up to 32 nodes (master or slaves) can be interconnected within one segment. The bus is terminated by an active bus terminator at the beginning and end of each segment. To ensure fault-free operation, both bus terminators must have a voltage supply at all times. The bus terminators can normally be activated either in the devices themselves or on the bus termination connectors.

To accommodate more than 32 nodes or increase the scope of the network, repeaters (cable amplifiers) can be installed to link the individual bus segments.

Fiber-optic transmission system

PROFIBUS can be implemented by means of fiber optics for installation in environments susceptible to high levels of radiated emission, where electrical isolation is essential or to increase the bus range with high transmission speeds. Various fiber types are available to suit different range, price and application requirements.

The following table shows a list of suitable fiber types:

ییوست ۲: تنظیم یارامترهای Micro Master 420

Operator panel/key	Function	Effects
^{اتا} ر 0000	Indicates Status	The LCD displays the settings currently used by the converter.
0	Start converter	Pressing the button starts the converter. This button is disabled by default. Activate the button: BOP: P0700 = 1 or P0719 = 10 16 AOP: P0700 = 4 or P0719 = 40 46 on BOP link P0700 = 5 or P0719 = 50 56 on COM link
0	Stop converter	 OFF1 Pressing the button causes the motor to come to a standstill at the selected ramp down rate. Activate the button: see button "Start converter" OFF2 Pressing the button twice (or once long) causes the motor to coast to a standstill. BOP: This function is always enabled (independent of P0700 or P0719).
۲	Change direction	Press this button to change the direction of rotation of the motor. Reverse is indicated by a minus (-) sign or a flashing decimal point. Disabled by default. Activate the button: see button "Start converter".
(09	Jog motor	In the "Ready to power-on" state, when this key is pressed, the motor starts and rotates with the pre-set jog frequency. The motor stops when the button is released. Pressing this button when the motor is running has no effect.
ø	Functions	 This button can be used to view additional information. It works by pressing and holding the button. It shows the following, starting from any parameter during operation: DC link voltage (indicated by d – units V). output current. (A) output requency (Hz) output voltage (indicated by o – units V). The value selected in P0005 (If P0005 is set to show any of the above (1 - 4) then this will not be shown again). Additional presses will toggle around the above displays. Jump Function From any parameter (rxxxx or Pxxxx) a short press of the Fn button will immediately jump to r0000, you can then change another parameter, if required. Upon returning to r0000, pressing the Fn button will return you to your starting point. Acknowledgement If alarm and fault messages are present, then these can be acknowledged by pressing key Fn.
Ø	Access parameters	Pressing this button allows access to the parameters.
0	Increase value	Pressing this button increases the displayed value.
0	Decrease value	Pressing this button decreases the displayed value.
ED+ P	AOP menu	Calls the AOP menu prompting (this is only available for AOP).

|--|

Operator panel keys

Parameterizing the drive with BOP or AOP

The frequency inverter is adapted to the motor using the quick commissioning function and important technological parameters are set. The quick commissioning shouldn't be carried-out if the rated motor data saved in the frequency inverter (4-pole 1LA Siemens motor, star circuit configuration \cong frequency inverter (FU)-specific) match the rating plate data.

Parameters, designated with a * offer more setting possibilities than are actually listed here. Refer to the parameter list for additional setting possibilities.






پیوست ۳ :کلمات کنترلی و کلمات وضعیت برای کنترل و بررسی وضعیت درایو Micro Master 420 از طریق PROFIBUS

Bit	Value	Meaning	Remarks
0	1	ON	Sets the inverter to the "Ready to run" state, direction of rotation must be defined via bit 11
	0	OFF1	Shutdown, deceleration along RFG ramp, pulse disable when f <f_{nit}< td=""></f_{nit}<>
1	1	Operating condition	
	0	OFF2	Instantaneous pulse disable, drive coasts to a standstill
2	1	Operating condition	
	0	OFF3	Rapid stop: Shutdown at fastest possible acceleration rate
3	1	Enable operation	Closed-loop control and inverter pulses are enabled
	o	Disable operation	Closed-loop control and inverter pulses are disabled
4	1	Operating condition	
	o	Disable ramp-function generator	Output of RFG is set to 0 (fastest possible braking operation), inverter remains in the ON state
5	1	Enable RFG	
	0	Stop RFG	Setpoint currently supplied by the RFG is "frozen"
6	1	Enable setpoint	Value selected at the RFG input is activated.
	0	Disable setpoint	Value selected at the RFG input is set to 0.
7	1	Acknowledge fault	Fault is acknowledged with a positive edge, inverter then switches to "starting lockout" state
	o	No meaning	
8	1	CW inching	
	0		
9	1	CCW inching	
	0		
10	1	Setpoints valid	Master transfers valid setpoints
	0	Setpoints invalid	
11	1	Setpoint inverted	Motor rotates CCW in response to positive setpoint
	0	Setpoint is not inverted	Motor rotates CW in response to positive setpoint
12		=	Not used
13	1	Motor potentiometer UP	
	Ũ		
14	1	Motor potentiometer DOWN	
	٥		
15	-	-Local control (BOP/AOP)	Local control active
		Remote control	Remote control active

Control word 1 bit assignments

جدول پ۳-۱

Assignment control word 2

Bit	Value	description
~	1	Fixed frequency Bit 0
U	0	
	1	Fixed frequency Bit 1
1	0	
	1	Fixed frequency Bit 2
2	0	
2	1	MICROMASTER 4, specific to model
3	0	
	1	MICROMASTER 4, specific to model
4	0	
e.	1	MICROMASTER 4, specific to model
0	0	
0	1	MICROMASTER 4, specific to model
0	0	
7	1	MICROMASTER 4, specific to model
1	0	
	1	MICROMASTER 4, specific to model
8	0	
9	1	MICROMASTER 4, specific to model
	0	
10	1	MICROMASTER 4, specific to model
10	0	
44	1	MICROMASTER 4, specific to model
	0	
12	1	MICROMASTER 4, specific to model
12	0	
13	1	MICROMASTER 4, specific to model
10	0	
14	1	MICROMASTER 4, specific to model
	0	
15	1	MICROMASTER 4, specific to model
15	0	

As standard, the 2nd control word is received as the 4^m word of the cyclic (ser data (PZD4) Bit assignments, status word 1

Bit	Value	Meaning	Remarks
0	1	Ready for ON	Power supply switched on, electronics initialized, pulses disabled
	0	Not ready for ON	
1	1	Ready to run	(see control word bit 0) Converter is switched on (ON command is applied), no fault is active, inverter can start when "Enable operation" command is issued.
	0	Not ready to run	Causes: No ON command, fault, OFF2 or OFF3 command, starting lockout
2	1	Operation enabled	See control word, bit 3
	0	Operation disabled	
3	1	Fault is active	Fault, see fault parameter r0947 etc. Drive is faulty and thus inoperative, switches to starting lockout state after successful correction and acknowledgement of fault.
	0		
4	1		
	0	OFF2 command applied	See control word, bit 1
5	1		
	0	OFF3 command applied	See control word, bit 2
6	1	Starting lockout	Drive can be restarted only by OFF1 followed by ON
	0	No starting lockout	
7	1	Alarm is active	Alarm, see alarm parameter r2110. Drive still in operation.
	0		
8	1	No setpoint/act.val. deviation	Setpoint/actual value deviation within tolerance range
	0	Setpoint/act.val. deviation	
9	1	Master control requested	The master is being requested to accept status as master control.
	0	Local operation	The master is not currently the master control.
10	1	freached	Converter output frequency is higher or equal to the maximum frequency
	0	f not reached	
11	1		
	0	Alarm: Motor at current limit	
12	1		Signal can be used to control a holding brake.
	0	Motor holding brake	
13	1		Motor data indicate overload condition
	0	Motor overload	
14	1	CW rotation	
	0	CCW rotation	
15	1 0	Converter overload	e.g. current or temperature

Assignment, status word 2

Bit	Value	Binary signal	Description
^	1		DC current brake active
5	0		
	1		
1	0		Drive inverter frequency < Shutdown limit
n	1		
4	0		
	1		Current ≥ Limit
3	0		
	1		Actual frequency > Reference frequency
4	0		
e	1		Actual frequency < Reference frequency
D	0		
0	1		Actual frequency ≥ Setpoint
0	0		
7	1		Voltage < Threshold value
,	0		
•	1		Voltage > Threshold value
0	0		
0	1		Opposite direction
5	0		
10	1		PI frequency < Threshold value
10	0		
11	1		PI saturation
	0		
12	1		MICROMASTER 4, model-specific
	0		
13	1		MICROMASTER 4, model-specific
	٥		
14	1		MICROMASTER 4, model-specific
	٥		
15	1		MICROMASTER 4, model-specific
	0		

The 2nd status word is sent from the drive inverter, as standard, as 4th word of the cyclic user data (PZD4).

جدول پ۳-۳

جلسه دوازدهم

هدف:

- معرفی HMIها
- معرفی سخت افزار TP170A
- معرفی صفحه نمایشهای سیستمی TP170A
- نحوه برنامه ریزی TP170A با نرم افزار Winncc flexible شامل:
 - نحوه ایجاد یک پروژه
 - **معرفی منظرگاههای صفحه ویرایش یک پروژه**
 - معرفی ساختار پروژه
 - معرفی صفحههای نمایش
- نحوه تنظیم جدول متغیرهای مربوط به تبادل اطلاعات بین HMI و PLC (tag)
 - o نحوه تنظیم پارامترهای ارتباطی PLC با PMI
 - o نحوه استفاده از اشیاء بخش Simple Objects بر روی صفحه نمایش
 - o معرفی چند ابزار خاص روی میله ابزار نرم افزار
 - نحوه انتقال برنامه یک پروژه از PG به HMI
 - نحوه پیکربندی HMI در نرم افزار Simatic manager
 - نحوه انتخاب و تنظیم زبانهای برنامه HMI
 - نحوه پاک کردن پروژه HMI
 - طراحی یک پروژه برای HMI و بررسی نحوه عملکرد آن در ارتباط با PLC

مقدمه:

HMI)معرفي HMIها

در صنعت اتوماسیون، سیستمهایخودکاری که با PLC کنترل میشوند عموما مجهز به وسایل واسطی هستند، که این واسط-ها امکان ارتباط کاربر با سیستم خودکار را فراهم میسازند، این نوع وسایل به اختصار HMI^۲ نامیده میشوند. ساده ترین ابزار ارتباطی انسان با یک سیستم خودکار تعدادی کلید ON-OFF و چند لامپ سیگنال (یا LED) میباشند که بصورت موازی به ورودی خروجیهای دیجیتال PLC وصل میشوند. اگرچه این نوع ابزارها در خیلی از کابردها کارساز میباشند، اما برای ارسال و دریافت همه مقادیر موجود در گستره تغییرات یک پارامتر از یک سیستم کارساز نمیباشند. بطور مثال اگر بخواهیم در یک سیستم حرارتی مقادیر دمای مطلوب از ۸۰۰ تا ۱۰۰۰درجه سانتگراد توسط کاربر تغییرداده شوند و یا اینکه دمای واقعی سیستم در این گستره توسط کاربر قابل روئیت باشند. دسترسی به این خواستهها با ابزارهای ذکر شده در بالا امکان پذیر نمیباشند.

سازندگان PLC برای ارتباط انسان با ماشین عموما تجهیزات سخت افزاری – نرمافزاری را ارائه میدهند که این تجهیزات تسهیلات لازم برای تبادل هرگونه اطلاعات بین کاربر و سیستم را فراهم می سازند. در این راستای بعضی از سازندگان PLC نرم-افزاری را برای HMI ارائه دادهاند که این نرمافزار بر روی یک PC نصب، و آن PC بر اساس نیازهای ارتباطاتی کاربر با سیستم، برنامه ریزی می شود. CC برنامه ریزی شده از طریق یک BUS مناسب به PLC وصل و از آن بعنوان HMI سیستم استفاده می-شود. این نوع HMIها تسهیلات لازم را برای ارتباط کاربر با سیستم با توانائی بالائی فراهم می سازند، بطوریکه کاربر می تواند فرامین مورد نیاز خود را از طریق صفحه کلید PL برای PLC سیستم ارسال و از طریق صفحه نمایش PL اطلاعات مورنیاز خود را بصورت گرافیگی و یا نوشتاری دریافت و ذخیره کند .

نوع دیگری از وسائل واسط ارتباطی که توسط سازندگان PLC ارائه میشوند OPها هستند. این نوع وسائل که با قلابلیتهای مختلف عرضه میشوند از طریق یک باس مناسب به PLC وصل و توسط پروگرامر مربوطه برنامه ریزی میشوند. شمای بلوکی یک

¹ Connection

²Human Machine Interface

OP در ارتباط با PLC در شکل (۱۲–۱) نشان داده شده است. ساده ترین نوع OPها دارای یک نشان دهنده تک خطی از نوع LCD (و یا LED) تک رنگ، بهمراه چند کلید الکترومکانیکی جهتدار ^۱ هستند، مجهزترین نوع OP دارای صفحه نمایش^۲ رنگی در ابعاد مختلف، بهمراه صفحه کلید الکترومکانیکی مجهز به کلیدهای اعداد، حروف و کلیدهای خاص^۳ میباشند. این نوع وسایل فرامین کاربر را از طریق صفحه کلید دریافت، و از مسیر پورت ارتباطی که عموما سریال هستند برای PLC ارسال میکنند و همچنین اطلاعات مور نیاز کاربر را از همین مسیر از PLC دریافت و در روی صفحه نمایش نشان میدهد.



استفاده از OPها در سیستمهای مختلف، نیاز به برنامه ریزی دارند. این برنامهریزیها بر اساس ضرورتهای ارتباطی انسان با سیستم مورد نظر انجام میشوند. برای این برنامه ریزی نیاز به نرمافزار خاص آن OP است که این نرمافزار خاص باید از سازنده OP تهیه و برروی یک PC نصب و از آن PC بعنوان پروگرامر OP استفاده شود.

بطور کلی اطلاعاتی که توسط وسائل HMIها ارسال و دریافت میشوند میتوانند شامل مقادیر کنترلی، پارامترهای سیستم و اطلاعات مربوط به وضعیت سیستم باشند. اطلاعات دریافت شده از یک سیستم میتواند بصورت نوشتاری و یا گرافیکی روی صفحه نمایش HMI نمایش داده و یا در حافظه HMI ذخیره شوند

نوع دیگری از وسائل واسط ارتباطیTPها هستند که این وسائل همانند OPها میباشند با این تفاوت که بجای صفحه کلید الکترومکانیکی دارای صفحه کلید لمسی میباشند. یعنی صفحه نمایش آنها علاوه بر عمل نمایش کار صفحه کلید را نیز انجام میدهد.

در این بخش از کار آزمایشگاهی برای بررسی کار یک نمونهHMI ، از یک TP (مدل TP170A) که ساخت شرکت زیمنس است استفاده می شود.

TP170A)معرفی سخت افزار

در شکل(۲-۱۲) نمای TP 170A از روبرو، پهلو و زیر نشان داه شده است. صفحه نمایش این HMI تک رنگ و بصورت صفحه کلید لمسی عمل میکند. در قسمت زیرین این وسیله سه کانکتور وجود دارند. یکی از آنها که دو پین است برای ارتباط منبع تغذیه 24V با این واسط است. دو کانکتور دیگر این واسط ۹ پین بوده و یکی برای ارتباط TP با PLC و دیگری برای ارتباط ۲ پروگرامر است. این وسیله واسط به کمک پروگرامری که مجهز به نرم افزار Flexible Wincc باشد. برنامه ریزی می شود.

¹ Arrow key

² Monitor

³ Function Key



در شکل (۲–۲۰) شمای ارتباطی یک TP170A با یک PLC سری 300 S7-30 و با یک PC (یا PG) نشان داده شده است. این PC که مجهز به نرم افزارهای Wince flexible و Simatic Manager است از آن برای برنامه ریزی PLC و HMI استفاده می شود. که وسیله اول با نرم افزار Simatic Manager و وسیله دوم با نرم افزار Wince Flexible برنامه ریزی می شوند ارتباط HMI با PLC برای تبادل اطلاعات کاربر با سیستم از طریق دستگاه HMI است



دستگاه TP170A علاوه بر قابلیتهائی که از طریق برنامهریزی برای آن ایجاد میشود. چهار صفحه نمایش سیستمی دارد که کاربرد هر یک از این چهار صفحه در زیر آمده است..

۲−۱۲) معرفی صفحه نمایشهای سیستمی TP170A:

پس از وصل شدن منبع تغذیه 24V به TP170A صفحه نمایش سیستمی شکل(۲۱-۴) روی صفحه نمایش TP آشکار می-شود. این صفحه نمایش دارای چهار دکمه است اگر هیچ کدام از آنها انتخاب نشود پس از چند ثانیه بعد از وصل شدن منبع تغذیه بصورت خودکار به حالت نمایش(حالت Start) رفته و (در صورت وجود برنامه نمایش در این HMI) اولین صفحه برنامه را نمایش میدهد. کاربرد هر یک از این چهار دکمهها به شرح زیر است.

Losder B 7.0.0 xx	
Tumber	
Turnahar	Contg
Start	Control

شکل(۴-۱۲)

دکمه Configure: هرگاه قبل از رفتن TP به حالت Start دکمه Configure انتخاب شود صفحه نمایش سیستمی شکل(۱۲-۵) باز می شود. در این صفحه تنظیمات لازم برای ارتباط TP با PG و TP با PLC انجام می شود. این تنظیمات شامل فعال کردن گزینه انتقال برنامه از PG به PF از راه دور و همچنین انتخاب آدرس و تنظیم سرعت تبادل اطلاعات برای TP (در شبکه PROFIBUS-DP و یا شبکه MPI) برای ارتباط با PLC است.



شکل(۱۲–۵)

دکمه Transfer: اگر در بخش Setting Transfer از صفحه نمایش Configure گزینه Remote انتخاب نشده باشد ، با انتخاب دکمه Transfer می توان HMI را برای دریافت برنامه از PG آماده کرد.

دکمه Control : هرگاه قبل از رفتن TP به حالت Start دکمه Control انتخاب شود صفحه نمایش سیستمی شکل(۲۱-۶) باز می شود. در این صفحه تنظیمات لازم برای TP که شامل تنظیم Contrast، تنظیم Calibration، تنظیم زمان Screensaver و انتخاب نوع زبان هستند انجام می شود.



Loader on the TP 170A

- Buttons for contrast control
- 2 Buttons for calibration
- 3 Buttons for the screensaver
- 4 Language selection checkboxes



شکل(۱۲-۶)

در صفحه نمایش سیستمی Control دکمهای بنام Calibrate وجود دارد که با انتخاب آن، صفحه نمایش سیستمی شکل(۱۲-۷) باز میشود. در این صفحه با روندی که در کنار این صفحه توضیح داده شده صفحه نمایش کالیبره میشود.





دکمهStart : اگر در TP که از قبل بر نامه ریزی شده است، دکمه Start (در صفحه نمایش سیستمی) آن انتخاب شود، اولین صفحه نمایش برنامه(یعنی Start Screen) به نمایش در میآید.

۲−۱۲) نحوه برنامه ریزی TP170A با نرم افزار Winncc flexible

برای برنامهریزی TP170A از نرمافزارWINC Flexible استفاده می شود. پس از نصب این نرمافزار در یک PC آیکن آن که مانند شکل زیر است بر روی صفحه نمایش PC ظاهر می شود. با دابل کلیک کردن بر روی این آیکون نرم افزار آن باز می شود. با باز



شدن این نرمافزار صفحهی شکل(۱۲–۸) روی صفحه نمایش PC نمایان میشود. در این صفحه پنج گزینه برای انتخاب موجود است که عملکرد هر یک از این گزینهها بشرح زیر میباشند.



- . Open the most Recently edited projects: با انتخاب این گزینه می توان آخرین پروژه ذخیره شده را برای ادامه تصحیحات، باز کرد.
 - ۲. Create a new project with the Project Wizard: با انتخاب این گزینه می توان با استفاده از برنامه Wizard در چند مرحله ساده، یک پروژه جدید را قدم به قدم ایجاد کرد.
 - ۳. Open an existing Project با انتخاب این گزینه می توان یک پروژه قدیمی را باز کرد.
 - ۴. Open an empty Project با انتخاب این گزینه می توان یک پروژه ی خالی را باز می کرد.
- ۵. Open a Protool project با انتخاب این گزینه می توان پروژه های ایجاد شده با نرم افزار Protool را باز کرد.

توضیح: ProTool نرم افزاری همانند WinCC Flexible است که در گذشته برای برنامه نویسی HMIها به کار گرفته می شد، اما امروزه نرمافزار WinCC Flexible جایگزین آن شده است.

Wizard الجاد یک پروژه با استفاده از برنامه) Wizard

با انتخاب گزینه Create a new project with the Project Wizard از صفحه شکل (۸-۱۲)، صفحه شکل(۱۲-۹) باز می شود. برای ایجاد یک پروژه هفت مرحله تنظیم وجود دارد که در ادامه تنظیم هر یک از مراحل توضیح داده شده است.

	The Project Wizard offers pre-defined scenarios for a Select the scenario that best matches your plant of Then select a STEP 7 project in which you wish to if you do not want to integrate it, leave the field bla Click on "Next" to continue configuration.	variety of plant configurations. configuration. integrate your HMI project. nk.
Select project type HMI device and controller Screen template Screen navigation	Small machine Large machine Distributed operation Control center and local operation	
System screens Libraries Project information	Sm@rtClient Integrate an S7 project	A controller is connected directly to an HMI device.
	 Back Cance (9–11), [3] 	el Finish Next

مرحله اول: Select Project Type: در این مرحله می توان نوع پروژه خود را انتخاب کرد که در اینجا بطوریکه در شکل(۱۲-۹) نشان داده شده است نوع پروژه Small machine انتخاب شده است. در هر مرحله، با انتخاب گزینه Next موجود در پائین صفحه پنجره مرحله بعدی باز می شود.

مرحله دوم HMI Device and Controller: نمای این مرحله در شکل(۱۰–۱۰) نشان داده شده است. در این مرحله، نوع HMI و نوع PLC و نوع Connection مشخص میشود. در اینجا TP170A برای HMI Device و SIMATIC S7 300/400 برای Contoroller

	Small machine		
	In this type of project, a controlle conforming to your plant configu • Click on the HMI device to so • If the selected HMI device su • Select your controller type fro	r is connected directly to an HMI device. S ration. Ject another type of HMI device. upports more than one resolution, select ; m the list.	Select here the HMI device, connection and controller your setting from the list.
Select project type	HMI device	Connection	Controller
Select project type 🔄			
HMI device and controller			
Screen template 🗌			
Screen navigation			
System screens 🗌	<u> </u>		
Libraries 🗌			
Project information			
	TP 170A	JIF1 B	SIMATIC \$7 300/400
	■ Back	Cancel	Finish Next

شکل(۱۲–۱۰)

برای انتخاب HMI Device با کلیک کردن روی شکل HMI Device صفحه شکل(۱۲–۱۱) باز می شود. در این صفحه می توان نوع HMI را انتخاب کرده و سپس با انتخاب دکمه OK از این صفحه خرج شد.

توضيح: نرمافزار WinCC Flexible امكانات خود را با توجه به HMI Device كه انتخاب مى شود محدود مى كند.

Device selection		
Device selection	Device type Micro Panels Mobile Panel Panels	
	OP 1778 mono DP OP 1778 mono DP OP 1778 color PN/DP ⊕: 220	Version of device 7.2.0.

شکل(۱۲–۱۱)

در شکل(۱۲–۱۰) در قسمت Connection نوع ارتباط HMI با PLC مشخص می شود. دو گزینه موجود است. MPI/DP و ETHERNET.

در اینجا اولی انتخاب میشود. در پایان با انتخاب دکمه Next این مرحله بسته شده و صفحه مرحله سوم باز میشود.

مرحله سوم: Screen Template: نمای این مرحله در شکل(۱۲–۱۲) نشان داده شده است. در این مرحله می توان یک صفحه نمایش الگو درست کرد. از آن برای ایجاد قابلیتهای مشترک همه صفحههای پروژه استفاده می شود. اگر تمام صفحات پروژه فرمی واحد داشته باشند، این صفحه نمایش می تواند طراحی را آسان کند. بطور مثال می توان محل آرم محل عنوان محل تاریخ و زمان را بطور یکسان برای همه صفحات نمایش در این صفحه طراحی کرد.

	Small machine
	Create a custom template for your screens. You can use this template in the project for each new screen created for the HMI devic Specify if you wish to create a header, navigation control and alarm line or alarm window. Select the elements to be included in the header. You can specify a graphic file for the company logo. Select the position and style of the navigation bar and alarm line / alarm window. If several different HMI devices are selected, their combined capabilities are displayed.
Select project type 🗹	محل تاريخ و زمان محل عنوان محل أرم V Screen title
Screen template	Company C
Screen navigation	Navigation control
System screens	Position:
Libraries	Cleft Cbelow Cright
Project information	Buttons:
	one row C two rows with text C C with graphics
	🖬 Alarm line / alarm window
	C Alarm line above Alarm line below C Alarm window
	◄ Back Cancel Finish Next

شکل(۱۲–۱۲)

در بالای صفحه الگو که در شکل مشاهده می شود. محلیهائی برای نمایش آرم، عنوان، تاریخ و زمان آماده شده و ناحیه نمایش اخطارها بصورت نواری در پانین صفحه برگزیده شده است.

در پایان این مرحله با انتخاب دکمه Next صفحهی این مرحله بسته و صفحه مرحله چهارم باز می شود.

مرحله چهارم: Screen Navigation: شکل (۱۲–۱۲) نمای مرحله چهرم را نشان میدهد. در این مرحله میتوان ترتیب باز شدن صفحههای نمایش را در زمان اجرا مشخص کرد. مثلا بطوریکه در شکل مشاهده می شود، صفحه نمایش آغازین^۱ پروژه به سه صفحهی دیگر مرتبط شده که یکی از این صفحه ها خود با دو صفحهی دیگر در ارتباط است. با بسته شدن این مرحله توسط دکمه Next صفحه مرحله پنجم باز میشود.



مرحله پنجم:System screens : نمای مرحله چهارم در شکل(۱۲–۱۴) نشان داده شده است. در این مرحله می توان تا شش صفحه نمایش سیستمی به پروژه اضافه کرد. این صفحه نمایشها که توسط طراح نرم افزار طراحی شدهاند برای نمایش اطلاعات سیستمی و یا برای در دسترس قرار دادن تجهیزات تنظیم مربوط به HMI است. در شکل نشان داده شده، یک صفحه نمایش سیستمی، بنام System Screen به پروژه اضافه شده است. در زیر کاربرد هر یک از شش صفحه نمایش سیستمی که در این مرحله می توان بصورت نرمافزاری به پروژه اضافه کرد معرفی شدهاند.

Start page				
	Small machine			
	Configure the system screens tha • Specify if you wish to create a • section screen level. • Click on "All system screens" t • Or select individual system sc	t should appear in the screen nar root screen for the system screen o select all system screens. reens and their objects.	Agation next to the plant screens. s or if all system screens should be	e displayed directly on the
Select project type 🖂		📕 Root screen for system	n screens	📕 All system scree
HMI device and controller	System screen for			
	Language switching			
Screen template	Runtime stop Opling/offling			
Screen navigation M	Screens for system diagnos	tics		
System screens	User administration			
	Project information			
Project information	System information			
	📕 System settings			
	◀ Back	Cancel	Finish	Next

شکل(۱۲–۱۴)

معرفى شش صفحه نمايش سيستمى موجود

¹ Start Screen

در نرمافزار امکانات شش صفحه سیستمی موجود در نرم افزار که در صورت نیاز میتوان هر یک از آنها را به پروژه اضافه کرد به شرح زیر هستند.

System Screen -1

این صفحه نمایش برای بکار گیری تعدادی تجهیزات سیستمی است که میتواند شامل:

- دكمه انتخاب Off line On line، براى قطع و
 وصل كردن تبادل اطلاعات بين HMI و PLC
- دکمه انتخاب حالت Transfer، برای آماده کردن
 HMI جهت دریافت برنامه از HMI
- دکمه انتخاب Exit Run time برای خارج شدن از حالت اجرای برنامه نمایش
- و دکمه انتخاب Switch Language برای تغییر زبان برنامه می باشد.

SIMATIC Wir	ICC flexible R	luntime	
SIEME	NS		
			SIMATIC PANEL
	SIEMENS	System screen	5:21:54 PM 1/10/2011
	Online	Offline	
	Transfer		
	Exit Runtime		
	Switch Language		
	Start Screen		Diagnostics screen

Diagnostic Screen - Y

این صفحه نمایش برای نمایش اعلان خطایهای سیستم است که رخ خواهد داد.

		SIMATIC PANEL
SIEMENS	Diagnostics screen	5:26:18 PM 1/10/2011
Start Scre	een	User administration

User Administration – ۳

این صفحه نمایش برای نمایش جعبههای محاوره برای کلمات عبور برای سطوح مختلف است.

SI/	MATIC W	inCC flexibl	e Runtime		
	SIEME	INS			
			S	SIMATIC PA	NEL
		SIEMENS	User administration	5:43:22 PM 1/10/2011	
		Start Screen	P	roject information	

Project Information -				EMENS
ن صفحه نمایش برای نمایش مشخصات پ	شامل	MATIC PANEL	2	
پروژه، تاریخ ایجاد پروژه و است.		5:45:58 PM 1/10/2011	Project information	SIEMENS
			test3	Project Name:
			1/10/2011	Created:
			plc1	Author:
				Description:
		em information		Start Screen

	System Information $-\Delta$
سيستم	این صفحه نمایش برای نمایش مشخصات
	شامل نوع HMI، نوع PLC و است.



System Setting -9

این صفحه نمایش برای نمایش و در دسترس قرار گرفتن تجهیزات مربوط به تنظیم HMI شامل تنظیم Contrast، پاک کردن صفحه نمایش و است.

WATIC WHICC HEXT	ble Kuntine	
SIEMENS		
	· 1,	SIMATIC PANEL
	:	
SIEMENS	System settings	5:47:49 PM
		1/10/2011
	Adjust Contrast	
Ca	librate touch screen	
	Clean screen	
Start Scre	en	

مرحله ششم: Libraries : نمای این مرحله در شکل(۱۲–۱۵) نشان داده شده است در این مرحله میتوان کتابخانههای مختلف را به پروژه اضافه کرد. دو کتابخانهی مهم که معمولا به پروژه اضافه میشود Graphics و Buttons and Switches هستند.

	Small machine		
	Select the libraries you wish to i Select the libraries you requ Select up to six files you wis	ntegrate into your project. re from the list of standard libraries. h to integrate as "custom libraries".	
Select project type 🗹	Available libraries		Selected libraries
HMI device and controller 🗹	Faceplates		Button and switches
Screen template 🗹	, accplates		Graphics
Screen navigation 🗹			
System screens 🗹			
Libraries 🗌			
Project information 🗌			
	Back	Cancel	Finish Next
	(شکل(۱۲–۱۵	

مرحله هفتم:Project Information : نمای این مرحله در شکل(۱۲-۱۶) نشان داده شده است سرانجام در این مرحله، اطلاعات کلی پروژه وارد میشود. این اطلاعات شامل نام پروژه، نام مالک آن و توضیحات اضافی مربوط به پروژه است.

part Edit Vew Issert Format F	aceptates Options Window Help				
Start page					
	Small machine				
	Enter information about the project Enter comments here. Press "Finish" to generate the	tha help you to later identify i project with your settings	Lin WinCC Bexible.		
Select project type 🖂	Project name		Comments		
HMI device and controller			-		
Screen template 🖂	Descure activity				
Screen navigation 🖂	Manadaraho				
System screens 🖂	admin				
Libraries 🖂	Creaton date				
Project information	6/76/2009				
					10
	L.				
	d Back	Cancel	Finish	100	
	And and a second s				-

شکل(۱۲–۱۶)

در پایان این مرحله با انتخاب دکمه Finish عملیات ایجاد پروژه پایان مییابد و صفحه مخصوص ایجاد پروژه بسته و صفحهی ویرایش نرم افزار باز میشود. این صفحه در شکل(۱۲–۱۷) نشان داده شده است.

۲-۲-۱۲) معرفی منظرگاههای صفحه ویرایش یک پروژه

در شکل (۱۲–۱۰) صفحه ویرایش یک پروژه نمایش داده شده است. این صفحه با باز شدن یک پروژه از قبل ایجاد شده، به نمایش در میآیند. این صفحه دارای یک محیط کار و پنج منظرگاه[\] مختلف به شرح زیر میباشند.

	 A strain per A str
	Electric d' Electric Ref Electric Ref Electric d' Electric d' Electric d' Electric de la constant Electric de la c
THEF. THEF.	
	2 - delineare
	To the second
	1 (mm - 1 m
Property view	
N State State State State	C. W.
and the second s	
Output view	An Anna Anna Anna Anna Anna Anna Anna A

شکل(۱۲–۱۷)

- محیط کار (Work Area): تمام اشیاء یک پروژه در این محیط پیکربندی، تنظیم و ویرایش و در نهایت نمایش داده می شوند. همه امکانات مورد نیاز در نرمافزار WINCC Flexible که برای این محیط کار استفاده می شوند، در اطراف آن در چند نظرگاه چیده شده اند. در این صفحه به استثنای محیط کار همه منظرگاههای اطراف محیط کار را می توان بر حسب خواسته کاربر جابجا و یا ناپدید کرد
- **منظرگاه پروژه (Project View**): همه پوشههای متعلق به یک پروژه و ویرایشگرها مربوطه به آن در این منظرگاه بصورت ساختار درختی نمایش داده میشوند. در این منظرگاه میتوان هر یک از پوشههای پروژه را باز و به ویرایشگرهای موجود در آن پوشه دسترسی پیداکرد.
- منظرگاه خواص اشیاء (Property View): در این منظرگاه می توان خواص اشیاء یک پروژه را ویرایش کرد. برای مثال می توان با کلیک راست کردن بر روی یک صفحه نمایش منظرگاه خواص آن صفحه نمایش را باز و از آن طریق رنگ زمینه صفحه نمایش را تغییر داد.
- **منظرگاه خروجی** (Output View): در موقع کمپایل و یا اجرا یک پروژه پیامهای آن بهمراه هشدارها و خطاها در این پنجره نمایش داده میشود.
- منظرگاه اشیاء (Object View): در این منظرگاه همه اشیاء یک پوشه موجود در پنجره پروژه، نمایش داده می شوند. برای نمایش اشیاء یک پوشه از پنجره پروژه، ابتدا لازم است با کلیک چپ کردن آن پوشه، پوشه مورد نظر انتخاب می شود.
- **پنجره ابزار** (Tool Window): در این پنجره مجموعهی از اشیاءای را که میتوان به صفحه نمایش یک پروژه اضافه کرد وجود دارد، بطور مثال این اشیاء شامل تصویر ابزار و یا عناصر عمل کننده در امور کنترل یک Plant میباشند.
- کتابخانه: کتابخانه در حقیقت محلی برای نگهداری پوشههای مختلف حاوی مجموعه ابزارهای دستهبندی شده است. همچنین درکتابخانه میتوان پوشه جدیدی ایجاد کرده و مجموعه ابزارهای مورد استفاده خود را در آن قرار داده تا بعدا مورد استفاده قرار گیرند.
- **توضیح**: اگر نظم نمایش شکل(۱۲-۱۷) تغییر یابد، با انتخاب گزینه Reset layout از منوی View نظم آن مجددا مانند حالت پیش فرض(مانند حالت نشان داده شده در شکل) مرتب می شود.

۲-۲-۳)معرفی ساختار منظرگاه پروژه:

هر پروژه مربوط به یک HMI از چند پوشه اصلی تشکیل شده است، بطوریکه در شکل(۱۲–۱۸) مشاهده میشود این پوشهها در منظرگاه پروژه در دسترس هستند. این پوشهها که داری ویرایشگرهای اجزای پروژه می-باشند. شامل:

- صفحه های نمایش^۱
 - ارتباطات^۲
- مدیریت هشدارها^۳
- Runtime User Administratinn
 - تنظیمات دستگاه[†]
 - تنظیمات زبان⁶ هستند.

در ادامه کابرد دو تا از پوشههای مهم که شامل صفحههای نمایش و Communicatiun هستند معرفی میشوند

۱۲-۲-۴)معرفی پوشه صفحههای نمایش:

هر پروژه از چند صفحهی نمایش مرتبط به هم تشکیل شده است. این صفحهها، در حقیقت مکانی هستند که اشیای مختلف، مانند متن و دکمه ها که قرار است در HMI نمایش داده شوند، روی آنها قرار داده می شوند. آرایش ترتیب نمایش این صفحه نمایشها در مرحله چهارم ایجاد پروژه انجام میشود. در قسمت بالای شکل(۱۲–۱۹) پوشه باز شده صفحههای نمایش در منظرگاه پروژه مشاهده میشود. در این پوشه سه برگه ویرایش مربوط به ویرایش صفحه الگوی همه صفحات نمایش است. تنظیمات این برگه در همه صفحات پروژه کپی میشود. دو برگه دیگر مربوط به ویرایش در مفحه نمایش موجود در این پروژه است که در این برگهها ویرایش اختصاصی هر یک از دو صفحه نمایش انجام میشود.

برای افزودن صفحهی نمایش جدید در پوشه صفحههای نمایش، از گزینهی Add Screen استفاده میشود. در قسمت پائین شکل(۱۲–۱۹) نمونه ای از یک ویرایشگر صفحه نمایش را که تعدادی اشیای بر روی آن نصب شده است مشاهده میشود.

با کلیک کردن بر روی صفحه نمایش و یا هر یک از اشیای نصب شده بر روی آن، در قسمت پایین محیط کار، پمنظرگاه تنظیمات خواص مربوط به شیئ یا صفحه نمایش کلیک شده آشکار می شود. در شکل(۱۲–۲۰) پنجره تنظیمات پارامترهای یک صفحه نمایش نشان داده شده است.

 General Properties 	×++ II	Function List
Events Loaded Cleared	1 <no function=""></no>	_
		1 1

¹ Screens

- ² Communication
- ³ Alarm Management
- ⁴Device Settings





Device_1(TP 170A)

Project



 $\mathbf{P}(\mathbf{x})$

شکل(۱۲–۱۹)

در این پنجره با انتخاب گزینه General مشخصات عمومی صفحه نمایش همچون نام، شماره و رنگ زمینه آن، قابل تعییر است. با انتخاب گزینه Events، میتوان گزینهای را از مجموعه آن انتخاب کرد که تا با توجه به اتفاقی که برای صفحه میافتد، تابعی اجرا

⁵ Language Setting

شود. مثلا میتوان تابعی را در قسمت Loaded اضافه کرد. تا هر بار که صفحه Load می شود، این تابع اجراشود.

Comunication معرفي يوشه (۵-۲-۱۲

در پوشه Comunication سه برگه مجزا وجود دار که اولی مربوط به جدول tagها و دومی مربوط به تنظیم Connection و سومی برای تنظیم Cycle Time ها میباشند. در ادامه هر یک از این برگهها معرفی می شوند.

۱- برگه جدول Tagها و نحوه تنظیم پارامترهای مربوط به آنها

		Name	Connection	Data type	Address	Array count	Acquisition cycle	Comment
Screens Communication		Speed_Max	Connection_1	Int	MW 3	1	500 ms	
		set	Connection_1	Bool	M 10.0	1	100 ms	
		Right_Left	Connection_1	Bool	M 12.2	1	100 ms	
		on_off	Connection_1	Bool	M 12.1	1	100 ms	
		Motor_speed	Connection_1	Int	MW 7	1	100 ms	
		dateTime	Connection_1	Date and t 🔻	MB 30 🔻	1	1 s 🔻	
	-	dateTime (Tag)						×
⊕ Song Language Settings ⊕ Structures ⊕ Structures ⊕ Song Version Management		General Properties Events	C Acquis Acquis A	Name dateT connection Conne Data type Date ition mode Cyclic ition cycle 1 s rray count 1	ime ection_1	• • • • • • • • • • • • • • • • • • •	Length 8	
	-							>

شکل(۱۲–۲۱)

۲- برگه Connection و نحوه تنظیم پارامترها برای ارتباط PLC با HMI :

در شکل(۲۲–۲۲) مسیر برگه Connection در منظرگاه پروژه بهمراه صفحه باز شده این برگه نشان داده شده است. در این برگه پارامترهای ارتباطی بین PLC و HMI تنظیم می شوند. در بالای صفحه در ستون Communication driver نوع PLC و در ستون name نامی برای این ارتباط تعین و در ستون Online گزینه وصل و قطع این ارتباط انتخاب می شوند. در قسمت پایین صفحه سه بخش مجزا بنامهای Network ،HMI Device و PLC Device وجود دارند که شرح تنظیم هر یک از این بخش ها در زیر آمده است.

¹ Connection

	Name	Communication driver	Online	Comment	
	Connection_1	SIMATIC 57 300/400	r On	•	
Communication	Parameters Area po	Interface IFIB	ice	Network	Station
	Type B TTY R5232 R5422 R5422 Simatic Simatic	aud rate 1500000 Address 4 Only master on the bus		Profile DP Highest station address (HSA) 15 Number of masters	Address 2 Expansion slot 2 Rack 0 Image: Cyclic operation

شکل(۱۲–۲۲)

تنظیم بخش HMI Device: در بخش HMI Device آدرس و سرعت ارتباط HMI با PLC از طریق شبکه مشخص می شود. این دو مقدار باید با توجه به مقادیری که در تنظیمات شبکه برای PLC انجام می شود، تنظیم شوند. در غیر این صورت ارتباط HMI با PLC دچار اختلال خواهد شد.

گزینه Only Master on the bus مربوط به زمانی است که تنها عنصر Master بر روی خط از نوع Simatic باشد.

تنظیم بخش Network: در بخش Network ، نوع شبکه (Profibus DP یا MPI) انتخاب می شود. دو مقدار زیرین این بخش با توجه به تنظیمات شبکه بکارگرفته شده تنظیم می شوند.

تنظیم بخشPLC Device: در بخش PLC Device اطلاعات مربوط به آدرس ماژول CPU بکار رفته در PLC تنظیم میشود. تنظیمات مورد استفاده در آزمایشگاه در شکل(۱۲–۲۲) نشان داده شده است.

۳- برگه تنظیم زمان تبادل اطلاعات PLC با HMI (Cyclic Time)

در این برگه که تصویر آن در شکل(۲۲–۲۲/۱) نشان داده شده است برای ایجاد زمانهای مختلف برای تبادل اطلاعات متغیرهای مختلف برای تبادل اطلاعات PLC با HMI است. برای اینکه متغیرهای مختلف PLC که در ارتباط با HMI هستند در زمانهای مختلف تازه شوند در این برگه این زمانها تنظیم و در برگه ایجاد تگها از آنها استفاده میشود.

Start Screen	Cycles	
Cycle time	Cycle unit	Name
1	Hour	1 h
1	Minute	1 min
1	Second	1 s
10	Second	10 s
100	Millisecond	100 ms
2	Second	2 s
5	Second	5 s
500	Millisecond	500 ms
1	Second	Cycle_1

شکل(۱۲–۲۱/۲۱)

در این برگه در ستون Cyclic Time زمان چرخه، در ستون Cyclic Unit واحد زمان چرخه و در ستون Name نامی برای این چرخه وارد می شود. بطور مثال در ردیف چهارم زمان چرخه ده ثانیه و نام آن I 10 تنظیم شده است.

	۲-۲-۱۲) نحوه استفاده از اشیاء بخش Simple Objects بر روی
Tools 📍 🍳	صفحه نمایش:
🕨 🔔 🛠	معمولا در سمت راست محیط کار ، منظرگاه ابزار دیده می شود. در
Simple Objects	این منظرگاه میتوان اشیای مورد نظر خود را برای نصب روی صفحه
A Text Field *	نمایش پیدا و با کمک موشواره به یکی از ویرایشگرهای صفحههای
ab) IO Field	نمایش پروژه منتقل کرد.
🕒 Date-Time Field *	نرمافزارWincc flexible از اشیای متعددی حمایت می کند، اما بر
👆 Graphic IO Field	اساس محدودیت HMI موجود در آزمایشگاه که TP170 است تعداد
💌 Symbolic IO Field	کمی از اشیاء در اینجا در دسترس میباشند.
🔽 Graphics View	در ادامه ۹ شیء موجود در جعبه ابزار Simple Objects که در
OK Button *	شکل(۱۲-۲۳) آمده. معرفی و نحوه تنیظیمات خواص دینامیکی و
o Switch	استاتیک مهم هر یک از آنها شرح داده می شود.
Bar	
V	
Ennanceo Objects	
Graphics	
Library	

شکل(۱۲–۲۳)

1- شيء متن (Text Field)

میدان این شی برای افزودن یک متن ثابت به صفحه نمایش استفاده می شود. این شیء را میتوان برای نمایش یک متن ثابت بر روی صفحه نمایش، بر روی ویرایشگر آن صفحه نصب کرد. با کلیک کردن بر روی شیء نصب شده پنجره تنظیم خواص آن در زیر پنجره محیط کار باز میشود.

TextField_0 (Text	Field)	(? (x	نمای باز شده این پنجره در شکل(۱۲–
General Properties Appearance Layout Text Misc	Fill Text color Background color Fill style Transparent	Appears	۲۴) نشان داده شده است. در این پنجره میتوان نوع متن، فونت، رنگ فونت و رنگ زمینه این شیء را تغییر داد
		>	

شکل(۱۲–۲۴)

۲- شیء ورودی / خروجی(IO Field)

میدان این شیئ برای وارد کردن و نمایش دادن دادهها یک تگ استفاده میشود. با این شی میتوان هم مقدار یک متغیر از برنامه PLC را تعیین، و هم مقدار آن را نمایش داد، و یا اینکه هر دو کار را با هم انجام داد. . با کلیک کردن بر روی این شیء پنجره خواص آن در زیر پنجره محیط کار باز میشود. در این پنجره سه فایل وجود دارد.

در فایل خواص General این شیء که در شکل(۱۲–۲۵) نشان داده شده است، سه بخش وجود دارد. که نحوه تنظیم هر یک از این بخشها بشرح زیراست.

General Droperties		General
Animations	Type Mode Output Process	Format The Decimal Format type Decimal Format pattern
	Tag Temp Cycle 1 s	9999 Shift decimal point 0 String field length

شکل(۱۲–۲۵)

ich sima	FIC WinCO	flexible F	Runtime	80	باشد، مشخص می شود. اگر این شیء بصورت ورودی انتخاب شود. با استفاده از صفحه کلیدی مشابه شکل (۱۲-۲۶) مقداردهی می شود. این صفحه کلید
A	1	2	3	ESC	با انتحاب شیء ورودی مورد نظر برروی صفحه نمایش طاهر میشود. این
в	4	5	6	BSP	انتخاب در حالت Runtime با کلیک و در حالت کار با HMI با لمس کردن
					شیء روی صفحه نمایش انجام میشود.
С	7	8	9	+1-	بخش Process: در این بخش متغیری از برنامه PLC که قرار است توسط
D	Е	F	0	•	این شیئ تنظیم شود، بصورت یک Tag مشخص می شود.
\leftarrow	\rightarrow		<	<u> </u>	بخش Format: در اینجا فرمت و الگوی داده مربوط به متغیر مشخص می-
شکل(۱۲–۲۶)			شک		شود. توضیح: در قسمت Properties مشخصات عمومی شیئ قابل تنظیم است.

۳- شیء ساعت/تاریخ:

میدان این شیء برای نمایش تاریخ و ساعت به کار می رود. همچنین میتوان به کمک آن تاریخ و ساعت سیستم را تغییر داد.

برای این کار باید در صفحه تنظیم خواص این شیء که در شکل(۱۲-۲۷) نشان داده شده است. پارامترهای مربوط به این شیء را همانند شیء قبلی تنظیم کرد.

nne rielu		
ral	Туре	Process
rties Hiops	Mode Input/output	O Display system time
5	Format	💿 Use tag
	✓ Display date	Tag dateTime
	✓ Display time	Cycle 1s

شکل(۱۲–۲۷)

۴- شیء ورودی خروجی گرافیکی:

میدان این شیء برای نوعی ورودی خروجی گرافیکی است که به کمک آن میتوان تغییرات یک متغیر را به صورت گرافیکی نمایش داد. مثلا وضعیت باز و بسته شدن یک شیر را با دو شکل متفاوت نمایش داد. صفحه تنظیم خواص Genrral این شیء در شکل(۱۲–۲۸) نشان داده شده است.

در بخش Display این صفحه فایلهای اشکالی را که میباید در دو وضعیت مختلف از یک Tag بیتی نمایش داده شود، و در بخش Process آن نام این Tag بیتی تعین میشود.

(Graphic IO Field)	(?)×
	General
Settings	Process
Mode Two states	Tag SetP
Display	Cycle 1 s
Value ON 1	
Graphic ON Pointing finger	
Graphic OFF Pointing finger	
	(Graphic IO Field) Settings Mode Two states Display Value ON 1 Graphic ON Pointing finger Graphic OFF Pointing finger

شکل(۱۲–۲۸)

۵- شیء ورودی خروجی نمادین(Symbolic)

کاربرد میدان این شیء همانند میدان شیء قبلی است با این تفاوت که به جای نمایش شکل از نمایش متن استفاده میشود.

۶- شىء Graphic View:

به کمک این شیی میتوان یک تصویر ثابت را به صفحه نمایش اضافه کرد. برای این کار کافی است در صفحه General از صفحات تنظیم خواص این شیء که در شکل(۱۲–۲۹) نشان داده شده است فایل تصویر مورد نظر انتخاب و سپس دکمه Set فعال شود.

Graphics View (Graphics View)	(?)×
General Properties		General
		Set Clear
	Boiler 5 Up_Arrow Down_Arrow Home	
	Left_Arrow Pointing finger Right_Arrow	

شکل(۱۲–۲۹)

۷− شيء Button:

یکی از پرکاربرد ترین اشیاء دکمه(Button) است. این شیء را میتوان به صفحه نمایش اضافه و برای آن یک تابعی تعریف کرد. به این ترتیب که هرگاه در زمان اجرا برنامه یک رخداد^۱ی به این دکمه وارد شود تابع تعریف شده برای آن دکمه اجرا می-شود. تنظیمات مختلفی برای خصوصیات این شیء وجود دارد. شکلهای(۱۲–۳۴ تا ۱۲–۳۴) صفحات باز شده از فایل تنظیم خصوصیات این دکمه را نشان میدهند.

تنظیم صفحه General: شکل(۱۲–۳۰) صفحه تنظیم General دکمه را نشان میدهد. که در این صفحه میتوان برای وضعیت روشن و یا خاموش دکمه، نوشتار و یا شکل تنظیم کرد. که در شکل نشان داده شده نوشتارهای on و ff تنظیم شده است.

Button_1 (Butto	n)		×
General Properties		Gener	al
Animations	Button mode	Text	
Events	Graphic Invisible	Text OFF OFF Text list Text OFF OFF Text ON On	

شکل(۱۲–۳۰)

تنظیم صفحه Properties: شکل(۱۲–۳۱) صفحه تنظیم Properties دکمه را نشان میدهد. بطوریکه از روی شکل مشاهد می-شود در این صفحه میتوان تعدادی از خصوصیات این شیء را تنظیم کرد که یکی از آنها تنظیم مجوز استفاده از این شیء برای کاربران در سطوح مختلف است. یعنی کاربر با چه سطحی از مجوز بتواند با این دکمه کار کند.

Button_1 (Button)		×
General Properties Appearance Layout Text Misc Security Animations Events	Fill & Focus Foreground color Background color Focus color Focus width	Appearance Border I 3D
	شکل(۱۲–۳۱)	

تنظیم صفحه Animation: شکل(۲۲-۱۲) صفحه تنظیم Animation دکمه را نشان میدهد. بطوریکه از روی شکل مشاهد می-شود در این صفحه می توان قابلیت تحرک این شیء را توسط یک Tag (متغیری از PLC) فعال و یا غیر فعال کرد. تنظیمات روی شکل برای تگ Dis_EN است که با یک شدن این Tag "قابلیت تحرک دکمه" غیرفعال می شود. نوع داده برای این تگ بیتی است.

Button_2 (Button)		×
General Properties	Enabled		Enable Obje
Animations	Гад	Object state	
Enable Object Events	Dis_EN	 Disabled Enabled 	
	Гуре		
	C Integer	Range from 0	🕂 to 🛛
	Bit Bit position	0 🕂	
	•		>

شکل(۱۲–۳۲)

اگر نوع داده برای این تگ Integer انتخاب شود، در محدودهای از تغییرات این تگ (که در قسمت Rang قابل تنظیم است) قابلیت تحرک دکمه غیر فعال و در بقیه گستره فعال خواهد شد.

تنظیم صفحه Event: شکل(۱۲–۳۳) صفحه تنظیم Event دکمه را نشان میدهد. با این تنظیم هرگاه رخداد^۱ی بر این دکمه اعمال شود تابع تنظیم شده برای آن دکمه اجرا خواهد شد. رخداد (Event)های متعددی برای اجرای یک تابع میتوان انتخاب کرد. بطوریکه در شکل(۱۲–۳۳) برای فایل Event نشان داده شده انواع رخدادهای مختلفی برای یک دکمه قابل انتخاب است. در این شکل مسیر دسترسی به لیست توابع قابل اجرا با این رخدادها را نشان میدهد. مهمترین این رخدادها تابع معرفی با است که با فشردن دکمه تابع تنظیم شده برای آن اجرا میشود. در ادامه این بخش چند تابع مهم و پرکاربرد از پوشه توابع معرفی شدهاند.

Button_1 (Button)		×
General Properties Animations Events Click Press Release Athente	× ◆ ◆ 上国 No function> G System functions All system functions All system functions Alarms Calculation	Function List
Deactivate Change	DecreaseValue IncreaseValue SetValue Edit bits Keyboard operation for screen objects	■

شکل(۱۲–۳۳)

• پوشه توابع Calculation : در این پوشه سه تابع بشرح زیر وجود دارند.

Decrease Value: اجرای این تابع یک واحد از متغیری کم میکند. Increase Value: اجرای این تابع یک واحد به متغیری اضافه میکند. Set Value: اجرای این تابع مقداری را برای متغیری تنظیم میکند.

• پوشه توابع Edit bit: در این پوشه سه تابع بشرح زیر وجود دارند.

Invert Bite: اجرای این تابع بیتی را وارونه میکند (صفر به یک و یک به صفر تبدیل می شود) Set Bite: اجرای این تابع بیتی را یک میکند. Reset Bite: اجرای این تابع بیتی را صفر می کند.

• در پوشه Screens توابع Screen Activate وجود دارند که هر یک صفحه مشخصی را باز میکنند.



• در پوشه سیستم برخی توابع مربوط به تنظیمات صفحه نمایش وجود دارند.

⊜- S'	ystem
	ActivateCleanScreen
	- AdjustContrast
	CalibrateTouchScreen

شکل(۱۲-۳۴) تنظیم انجام شده برای یک دکمه را نشان میدهد. تنظیمات این دکمه بگونهای انجام شده که هرگاه برای دکمه عمل کلیک رخ دهد تابعی اجرا می شود که با اجرای آن تابع یک واحد به تگ Speed_Max اضافه می شود.

Button_1 (Butto	on)			?
 General Properties 	X	★ ↓ Ⅲ≣		Function List
Animations	1	🖃 Increase¥alue		
Events		Tag (InOut)	Max_speed	
Press		Value	1	
Release	2	<no function=""></no>		
ActivateDeactivateChange				Value to be added.

شکل(۱۲–۳۴)

:(کلید) Switch-۸

با این شی میتوان وضعیت Tagهای دو وضعیتی را تغییر داد، شکل(۱۲–۳۵) صفحه باز شده خصوصیات General این شیء را نشان میدهد. در این صفحه امکان تنظیم دو نوع برچسب برای این شیء وجود دارد. برچسب با متن و برچسب با شکل. در این شکل تنظیم برچسب بصورت متن انجام شده است. بخش Process این صفحه نام تگ و زمان بروز رسانی آن تنظیم میشود. امکانات تنظیم خصوصیات در صفحات دیگر این شیء مشابه دکمه میباشد که در بخش قبل توضیح داده شده است.

Switch_2 (Swite	cn)	(
 General Properties 		General
Animations	Settings	Process
Events	Type Switch with text 🗸	Tag set 🗸
	Label Switch with text	Cycle 100 ms
	Text	Value ON 1
	Text ON on	
	Text OFF Off	

شکل(۱۲–۳۵)

A- Bar (نمودار میلهای):

مىشود.

این شی یک نمودار ستونی گرافیکی است. که میتواند به صورت پویا مقدار یک متغیر را نشان دهد. شکل(۱۲–۳۶) مربوط به صفحه تنظیمات General که برای تنظیمات پویا و ایستای این شیء استفاده میشود، نشان میدهد.

General Properties					Gener
Topordos	Scale				
			Static	Tag	Cycle
		Maximum value 📕	X 100		
		Process	x	Motor_speed	100 ms
		Minimum value	0		

شکل(۱۲–۳۶)

در بخش Scale صفحه General مقادیر حداقل و حداکثر متغیر و همچنین نام تگ و زمان بروز رسانی این شیء تنظیم می شود. در صفحه Properties→Scale که در شکل(۱۲–۳۷) نشان داده شده است تعداد تقسیم ها و دقت اعداد نمایشگر میله ای تنظیم

 General Properties 	·	Scale
 Appearance Layout Text Limits Misc Scales 	Settings Elements Large interval 20 Mark increment label 1 Number of subdivisions 5 Double-lined scale label	ıt Hs
	Scale labels Total length 3 + -12,00 0 + Decimal	places

شکل(۱۲–۳۷)

۱۲-۲-۷) معرفی چند ابزار خاص در روی میله ابزارنرم افزار:

روی نوار ابزار Wincc Flexible ۲ ابزار خاص این نرم افزار وجود دارند که در شکل(۱۲–۳۸) هر یک از آنها معرفی و در ادامه توضیح کاربرد آنها بترتیب شماره ثبت شده روی شکل آمده است



شکل(۱۲–۳۸) محل ابزارها روی نوار ابزار

- ۱- ابزار Generate: با این ابزار، سازگاری ساختار برنامه طراحی شده بررسی وگزارش این بررسی در منظرگاه خروجی نمایش
 داده می شود. در این گزارش اخطارها با رنگ آبی و خطاها با رنگ قرمز مشخص می شوند.
- ۲- ابزار Start Runtime System: با این ابزار، برنامه طراحی شده بطور عملیاتی (بدون داشتن دستگاه HMI) به اجرا در آمده و نحوه عملکرد اجزای صفحههای نمایش آن بررسی می شوند. از این ابزار بشرطی می توان استفاده کرد که نرم افزار Runtime در PC نصب شده باشد.
- ۳- ابزار Start runtime system with script debugger: با این ابزار، برنامه Runtime بهمرا اشکال یاب متن به اجرا درمی آید.

File	e Edit Vi	ew ?									
	🗋 🗅 🚔	🖬 🖁 🔏 🗎	8 8 ?								
	Tag	Data Type	Current val.	Write cyle (s)	Simulation	Set value	Min∀alue	MaxValue	Cycle	Start	
0	Max_Sp	INT	80	1.0	<display></display>		-32768	32767		V	
	Motor_s	INT	10	1.0	Increment		0	100	10.000	v	
2										Г	

شکل(۱۲–۳۹)

توضیح: نرم افزار Runtime در ارتباط با نرما افزا شبیهساز Simatic manager اجرا می شود و از طریق این شبیه ساز می توان مقادیر تگها را برای Runtime تغییر داده و عملکرد آنها را بررسی کرد.

توضيح:

- ۵- ابزار Transfer setting: با این ابزار، صفحه تنظیم پارامترهای انتقال برنامه به HMI، باز می شود.
- ۶- ابزار Fine Object in project با این ابزار، صفحه محاوره برای پیدا کردن یک شیء بکار گرفته در پروژه باز می شود.
- ۲- ابزار Fine text string in Current view: با این ابزار، صفحه محاوره برای پیدا کردن رشته ای از متن درمنظرگاه فعلی باز می می شود.

۲−۱۲–۸)نحوه انتقال برنامه یک پروژه از PG به HMI:

برای انتقال برنامه از Programmer به دستگاه HMI، ابتدا باید از وصل بودن کابل RS232 اطمینان حاصل شود. سپس برنامه از مسیر Project→Compiler→Generate کامپایل شده تا سازی اجزای عناصر برنامه بررسی شود. در صورت عدم وجود هر گونه اشکال از مسیر Project→Transfer Setting بطوریکه در شکل(۱۲-۴۰) نشان داده شده است گزینه Transfer Setting انتخاب می شود. **توضیح:** برای کمپایل برنامه و انتقال آن به HMI می توان علاوه بر مسیرهای آمده در بالا از ابزارهای موجود روی میله ابزار نرم افزار استفاده کرد. آیکنهای این ابزارها در بخش معرفی ابزارها روی میله ابزار معرفی شدهاند.

Wir	nCC flexible Advanc	ed - Proj	iect_HMI.hmi
ject	<u>E</u> dit ⊻iew Insert Eormat	F <u>a</u> ceplates	Options <u>W</u> indow <u>H</u> elp
D	<u>N</u> ew		/ 🐻 🕶 🐂 . 🛛 🕸 . 🛛 🚻
	Ne <u>w</u> Project with Project Wiz	ard	✓ B Z U X . N N
5	<u>O</u> pen	Ctrl+O	
	Close		Section_0
	<u>S</u> ave	Ctrl+S	ENS
	Save <u>A</u> s		
	Save As <u>V</u> ersion		
	Save and Optimi <u>z</u> e		
	Archive		Section_0
	<u>R</u> etrieve		
	Change Device Type		
	Import Tags		
	Import/Export		
	Print Project Documentation	. Ctrl+P	
	Print Selection	Ctrl+W	1/1/1
	Co <u>m</u> piler		Message lext Message Me
	<u>T</u> ransfer		Communication Settings
	Recent Projects		▶ 💐 Transfer <u>S</u> ettings
		(4+-1)	شکل(۲

با انتخاب گزینه و یا ابزار مربوط به انتقال برنامه، صفحه شکل(۲۱–۴۱) باز می شود. در این صفحه در بخش Mode نوع انتقال (MPI/DP یا Serial)، در بخش Port، پورت ارتباطی Programmer با HMI و در بخش Baud rate سرعت انتقال اطلاعات انتخاب می شود. در نهایت با فعال کردن دکمه Transfer، در صورتیکه در تنظیمات دستگاه HMI در بخش Transfer Setting گزینه ی Remote انتخاب شده باشد، Programmer اجرای برنامه یا HMI را متوقف کرده و عمل انتقال برنامه جدید به HMI را اجرا می-کند. در غیر این صورت، ابتدا لازم است توسط کاربر اجرای برنامه یا HMI متوقف و HMI به حالت Transfer برده شود، پس از این انتخاب حالت برای HMI دکمه Transfer فعال شود.

Select devices for transfe	r Territori	1 (70 1704)			Þ
Device_1 (TP 170A)		ce_1(1P170A)		Transfer to	◯ RAM
	Mode	Serial	v	Delta transfer	O Off
	Port	COM1	~	Enable back tra	insfer
	Baud rate	115200	~	🗌 Overwrite passv	word list
				Overwrite recipe	e data records
			Т	ransfer App	ly Cancel
		×۱۲), (۴۱–۱۲)	ش		

نکته مهم: TP170A سیستم عاملی قدیمی دارد که با نسخه های جدید WinCC سازگاری ندارد. برای رفع این OS Update ارتقاء یابد. برای این کار باید از مسیر Transfer گزینهی Project ک انتخاب شود. با این انتخاب صفحهی شکل(۲۱–۴۲) باز میشود. در این صفحه با انتخاب گزینهی Update OS کار ارتقاء سیستم عامل شروع و پس از اندک زمانی سیستم عامل HMI ارتقاء مییابد. توضیح اینکه برای HMI موجود در آزمایشگاه این ارتقاء قبلا انجام شده است.

توضیح: در اینجا هر سرعتی انتخاب شود کار انتقال برنامه به HMI انجام خواهد شد. ولی بهتر است هر چه فاصله HMI از PG زیادتر باشد سرعت انتقال کمتر انتخاب شود

Image path: C:\Program Files\S	iemens\SIMATIC WinCC flexible\WinCC flexible Ii Device status
Reset to factory	settings
Opening: Windows CE: Image version: OK	C:VProgram Files\Siemens\SIMATIC \WinCC flexible\WinCC flexible Images\TP1 30 B 7.2.0.0
<	
	Cancel Update OS

شکل(۲۴–۴۲)

۱۲-۱۲) نحوه پیکربندی HMI در نرم افزار Simatic manager :

دستگاه HMI در ارتباطات با PLC همواره به صورت Master شناخته می شود، و نیازی به پیکربندی سخت افزاری خاص ندارد. تنها کافی است یک شبکه پروفیباس برای PLC پیکربندی شده و پیکربندی آماده شده به PLC منتقل شود. پس از این اتنقال و اتصال HMI به شبکه، تنظیمات سخت افزاری HMI برای شبکه شناخته خواهد شد.

شکل(۲۲–۴۳) خط Profibus طراحی شده در برنامه پیکربندی سخت افزار PLC را نشان میدهد. برای آشنایی با نحوه پیکربندی Profibus و دیگر تنظیمات سخت افزاری به مطالب جلسه یازده مراجعه شود. توضیح اینکه آدرس CPU در این شبکه باید منطبق بر آدرس تنظیم شده در شکل(۲۱–۴۹) یعنی عدد ۲ باشد.

	0	
	n	
1	PS 307 2A	
2	CPU315-2DP(1)	
X2	DP	PROFIBILS(1): DR montes system (1)
3		PROFIBUS(1). DP IIIastel system(1)
4	DI16xDC24V	
5	D016xDC24V/0.5A	
6		
7		
8		
9		
10		
11		

شکل(۲۴–۴۳)

۲۱–۴) نحوه انتخاب و نحوه تنظیم زبانهای برنامه پروژه:

شکل(۱۲-۴۵) صفحه نمایش سیستمی System Screen را نشان میدهد. این صفحه نمایش سیستمی مجهز به دکمهای است که با آن میتوان زبان نمایش نوشتار برنامه را انتخاب کرد. محل این دکمه روی شکل نشان داده شده است. با کلیک کردن این دکمه بطور متوالی زبان برنامه به ترتیب تنظیم شده عوض میشود.

SIMATIC WinCC flexible Runtime	
SIEMENS	
SIMATIC PAN	EL
SIEMENS System screen 11:42:26 AM 1/10/2011	
Online Offine	
Exit Runtime	
Switch Language دکمه تغیر زبان بوشتارها در حالت سایش برنامه	
Start Screen	

شکل(۱۲–۴۵)

شکل(۱۲-۴۶) محل تنظیم زبانهای حالت ویرایش و نمایش برنامه را نشان میدهد.

Project 🔶 🔍	Start Screen OProject Languages	
w test2 ■ Device_1(TP 170A) ■ Device_s ■	General Editing Jacquage	
 Runtime User Administration Device Settings Project Languages Graphics Dictionaries Structures Version Management 	English (United States) English (United States) English (Belize) English (Carabda) English (Carabda) English (Carabda) English (Ireland) English (Ireland) English (New Zealand) English (New Zealand) English (South Africa)	English (United States) English (United States) English (United Kingdom) English (United Kingdom) English (United States) English (Zinbabwe) Estorian (Estoria) Faeroese (Faeroe Islands) Faero (Iran) Finnish (Finland)

شکل(۱۲–۴۶)

شکل(۱۲-۴۷) محل تنظیم توالی انتخاب زبان در حالت نمایش و تنظیم فونت زبانهای مربوطه را نشان میدهد.



شکل(۲۴–۴۷)

الف- نحوه پاک کردن پروژهHMI :

🖀 Open exi	sting proje	ct		×
Look in:	📋 My Document	8	رٍ € + T	K 📸 🎟 -
History History Stand-alone My Documents	My eBooks My Music My Pictures My Videos OneNote Notet Project_2_ites Project_1.hmi Project_3.hmi Project_4.hmi	oooks Documentation) ni	ابزار پاک کردن فایل	
	File name:	Project_2.hmi		~
	File of type:	HMI project (*.hmi)		*
			Open	Cancel

برای پاک کردن یک پروژه HMI میتوان در پنجره نشان داده شده در شکل(۱۲-۴۸) از ابزار نشان داده شده استفاده کرد.

شکل(۱۲–۴۸)

PLC) طراحی یک پروژه و بررسی نحوه عملکرد آن در ارتباط با

میخواهیم برای HMI آزمایشگاه (TP170A) پروژهای طراحی شود، تا کاربر با این HMI بتواند داریور موتور AC موجود در آزمایشگاه را راهاندازی، کنترل و پایش کند. این HMI برای بکارگیری برنامه PLC که برای همین منظور از قبل آماده شده است استفاده خواهد شد. در برنامه PLC آماده شده راهانداز موتور توسط کاربر روشن و خاموش شده و سرعت موتور در دو جهت بطور متوالی همواره از صفر تا ماکزیمم سرعتی که توسط کاربر تعین میشود، تغییر میکند. در این پروژه میخواهیم HMI فرامین روشن و خاموش شدن موتور و ماکزیمم سرعت آن را از کاربر دریافت کرده، و مقدار سرعت لحظهای موتور، جهت حرکت موتور و همچنین مقادیر تاریخ و زمان PLC را بر روی صفحههای خود نمایش دهد.

برای این طراحی، ابتدا یک پروژه با دو صفحه نمایش ایجاد، و سپس تنظیمات ارتباطی و پیکربندی اشیاء مورد نیاز پروژه برطبق روند زیر انجام میشوند. در پایان این طراحی، برنامه طراحی شده برای HMI، در سیمولاتور HMI (برنامه Runtime) با بکار گیری سیمولاتور PLC (Simatic Manager) بررسی میشود. پس از این بررسی و رفع اشکال احتمالی آن، در نهایت پروژه بصورت واقعی توسط تجهیزات آزمایشگاه شامل HMI (TP170)، PLC (سری 300)، Priver (200) (Micro Master 420) و با بکارگیری شبکه Profibus بررسی میشود. برنامههای مورد نیاز برای سیمولاتور PLC و PLC واقعی در پایان این بخش ارائه شده است.

روند طراحی پروژه:

الف) ایجاد پروژه: با روندی که در بخش (۱۲–۲–۱) آمده است یک پروژه HMI با دو صفحه نمایش با نامهایStart_Screen و Sectiom_0 برای A TP170 ایجاد میشود.

ب) تنظیم پارامترهای Connection: با کمک اطلاعات بخش (۱۲–۵–۵) پارامترهای ارتباطی بین PLC و HMI مانند آنچه آمده در شکل(۱۲–۶۹) تنظیم می شوند.



شکل(۴۹-۱۲)

ج: تنظیم جدول Tagها: تگهای مورنیاز برای پروژه (که در ارتباط با متغیرهای مربوط به نمایش اطلاعات و دریافت فرامین سیستم هستند) به روشی که در بخش (۱۲–۲–۵) آمده است وارد جدول Tagها می شوند. tagهای مورد نیاز در این پروژه در جدول شکل(۱۲–۵۰)آمده است.

🖃 🖛 ocieens	Name	Connection	Data type 📥	Address	Array cou	Acquisition cycle
Template	Right_Left	Connecti 🔻	Bool 🔹	M 12.2 🔻	1	100 ms 🔹 🔻
Diagnostics screen	set	Connection_1	Bool	M 12.1	1	100 ms
Start Screen	on_off	Connection_1	Bool	M 10.0	1	100 ms
🖃 🧏 Communication	dateTime	Connection_1	Date and time	MB 30	1	1 s
S Connections	Speed_Max	Connection_1	Int	MW 3	1	500 ms
Cycles	Motor_speed	Connection_1	Int	MW 7	1	100 ms

شکل(۱۲–۵۰)

د)پیکربندی اشیاء مورنیاز روی صفحههای نمایش و تنظیم آنها

در زیر در دو بخش جدا گانه، اشیای هر یک از دو صفحه نمایش Start Screen و Sectiom_0 معرفی و نحوه تنظیم آنها ارائه شده است.

اشیای صفحه نمایش Start Screen : در روی صفحه نمایش Start Screen که در شکل(۵۱–۵۱) نشان داده شده است، نه شیء مانند شکل نشان داده شده در زیر نصب می شوند. این اشاء شامل یک Switch برای On و Off کردن سیستم، یک IO Field بصورت ورودی برای تعین ماکزیمم سرعت موتور، یک Graphic IO برای نمایش جهت حرکت موتور بصورت تصویری، یک IO PLC بصورت خروجی برای نمایش سرعت موتور بصورت نوشتاری، یک Date- Time Field برای نمایش تاریخ و زمان از PLC و سه Field بصورت خروجی برای نمایش سرعت موتور بصورت نوشتاری، یک Motor Speed برای نمایش تاریخ و زمان از PLC و سه Text Field برای نمایش نوشتارهای Max Speed% و Motor Speed % می باشند، که در ادامه نحوه تنظیمات هر یک از این اشیاء توضیح داده شده است. هر یک از این اشیاء با موشواره از بخش Sintle Object به این صفحه انتقال داده شده و سپس در پنجره خواص مربوطه تنظیمات لازم انجام می شوند. شیء Rotton که برای پرش کردن از صفحه موازار بر روی این صفحه نصب و تنظیم می شوند.



شکل(۱۲–۵۱)

نصب تنظیم Text Filed ها: روی صفحه نمایش Start Screen سه عدد شیء Text Filed برای نمایش متنهای Start Screen، Max Speed% و Motor Speed% وجود دارد که نحوه تنظیم یکی از متنها در شکل(۱۲–۵۲)نشان داده شده است. که دو متن دیگر به همین صورت تنظیم میشوند.

Text Field_1 (Text	Field)	P X
 General Properties 	Text %Max Speed	
		>

شکل(۱۲–۵۲)

نصب شیء Date- Time: روی صفحه نمایش Start Screen یک عدد شیء Date- Time برای نمایش تاریخ و زمان با آدرس MB30 وجود دارد که نحوه تنظیم آن برای حالت ورودی / خروجی در شکل(۱۲-۵۳)نشان داده شده است.

Date-Time Field_1	(Date-Time Field)	
🔶 General	Туре	Process
Properties Animations	Mode Input/output	O Display system time
Events	Format	⊙ Use tag
	✓ Display date	Tag dateTime 👻
	✓ Display time	Cycle 1 s

شکل(۱۲–۵۳)

نصب دو شیء IO Field روی صفحه نمایش Start Screen دو عدد شیء IO Field وجود دارند. که یکی از این آنها برای نمایش در صد سرعت موتور که تگ آن بصورت خروجی و آدرس آن MW7 است و دیگری برای تنظیم ماکزیمم درصد سرعت موتور که تگ آن بصورت ورودی و آدرس آن MW3 است. نحوه تنظیم هر یک از آنها در دو شکل(۲۱–۵۴ و ۲۲- ۵۵)نشان داده شده است.

IO Field_2 (IO F	ield)	(? 🗙
General Properties		General
Animacions	Mode Output	Format Format type Decimal
	Process Tag Motor speed	Format pattern 99
	Cycle 100 ms	Shift decimal point
		String field length 2

شکل(۱۲–۵۴)

IO Field_1 (IO Fie	eld)	×
General Properties		General
Animations	Туре	Format
Events	Mode Input/output	Format type Decimal
	Process	Format pattern
	Tag Max_Speed 🗸	99
	Cycle 500 ms	Shift decimal point 0 📫
		String field length 2

شکل(۱۲–۵۵)

نصب **شیء Graphic IO Field:** روی صفحه نمایش Start Screen یک عدد شیء Graphic IO Field برای نمایش شکلهای Right_Arrow و Left_Arrow برای نشان دادن جهت حرکت موتور وجود دارد که آدرس تگ آن M12.2 است. نحوه تنظیم این شیء در شکل(۱۲-۵۶) نشان داده شده است.

Graphic IO Field_2	(Graphic IO Field)	(?)
General Properties		General
Animations	Settings	Process
	Mode Two states 🗾	Tag Right_Left
	Display	Cycle 100 ms
	Value ON 1	
	Graphic ON Right_Arrow	
	Graphic OFF Left_Arrow	

شکل(۱۲–۵۶)

نصب شیء Switch: روی صفحه نمایش Start Screen یک عدد شیء Switch برای روشن و خاموش کردن درایور موتور وجود دارد که نام تگ این شیء on_off و آدرس آن M10.0 است. نحوه تنظیم این شیء در شکل(۱۲–۵۷)نشان داده شده است.

Switch_2 (Switc	h)		?
General Properties Animations Events	Settings Type Switch with text	Process Tag on_Off Cycle 100 ms Value ON 1	General
	Text OFF Off		

شکل(۱۲–۵۷)

نصب شیء Button: روی صفحه نمایش Start Screen یک عدد شیء Button برای پرش از صفحه نمایش Start screen به صفحه نمایش Section_0 وجود دارد که نحوه تنظیم این کار برای رخداد کلیک در شکل(۱۲–۵۸)نشان داده شده است.

General Properties				Function Lis
Animations	1	ActivateScreen		
Events		Screen name	Section_0	
Press		Object number	0	
 Release 	2	<no function=""></no>		
 Activate 				



اشیای صفحه نمایش Section_0 ید روی صفحه نمایش Section_0 Screen که در شکل(۵۲-۵۹) نشان داده شده است، شش شیء نصب می شوند که یکی از آن ها Button و دیگری Switch و دو تای دیگر از آنها Text Filed هستند که هر سه نوع این اشیاء در صفحه نمایش قبلی موجود بوده و نحوه تنظیم آن ها در آن بخش توضیح داده شدهاند. در این صفحه علاوه بر این سه شی یک نمایشگر میلهای (Bar) برای نمایش سرعت موتور، و یک خروجی نوشتاری (Symbolic IO Field) برای مشاهده جهت حرکت موتور نصب می شوند. که نحوه تنظیمات این دو شیء در زیرآمده است. هر یک از این اشیاء با موشواره از بخش Button که برای به این صفحه انتقال داده شده و سپس در پنجره خواص مربوطه به هر یک تنظیمات لازم انجام می شوند. شیء Button که برای پرش کردن از صفحه [ومی این موجود این می موجود این موجود شوند. در موقع ایجاد صفحه نمایش ها توسط نرمافزار روی این صفحه نصب و تنظیم می شوند.



نصب شیء Bar: روی صفحه نمایش Section_0 یک عدد شیء Bar با تگ Motor_Speed برای نمایش سرعت موتور بصورت نمودار میلهای وجود دارد که آدرس این تگMw7 است. نحوه تنظیم این شیء در شکل(۱۲-۶۰) نشان داده شده است.

 General Properties 	Scale				Gener
		Maximum value Process Minimum value	Static	Tag Motor_speed	Cycle
نصب شِئ Symbolic IO Field: روی صفحه نمایش Section_0 یک عدد شیء Symbolic IO Field برای نمایش متنهای Right_ Left و آدرس آن M12.2 است. نحوه تنظیم این شیء در شکل شکل (۱۲–۶۱) نشان داده شده است.

General Properties		General
Animations	Settings	Process
	Mode Two states	Tag Right_Left
	Display	Cycle 100 ms
	Value at ON state 1 Text ON Right	
	Text OFF	

شکل(۱۲-۶۱)

پس از آماده شدن برنامه HMI، ابتدا آنرا با بکارگیری برنامه Runtime بررسی کرده و پس از رفع اشکال احتمالی به دستگاه HMI منتقل کنید. پس از انتقال نحوه عملکرد برنامه را در ارتباط با برنامه PLC بررسی کنید.

برنامه برای سیمولاتور PLC: برنامه برای سیمولاتور Simatic manager جهت بررسی پروژه HMI با برنامه Runtime



برنامه برای PLC واقعی: برای بکارگیری HMI در حالت واقعی برنامه زیر را به PLC منتقل کرده و عملکرد برنامه طراحی شده برای HMI را در ارتباط با این برنامه بررسی کنید.



توضيح برنامه:

.

در شبکه چهاردهم مقادیر مربوطه به تاریخ و زمان PLC از متغیرهای محلی آن (OB1_DATE_TIME#) به آدرسهای Tagهای مربوطه MD30 و MD34 منتقل می شوند.