## آموزش PLC دلتا

قسمت دوم

در قسمت قبل با نرم افزار WPLSOFT آشنا شدیم و نحوه کار با نرم افزار، برخی تیغه ها و دستورات SET و RST آشنا شدیم.

در این بخش میخواهیم با دستوراتی از قبیل تایمر ها و کانتر ها آشنا شویم.

در صفحه اصلی برنامه و در نوار ابزار گفته شده در قسمت قبل ، آیکنی به شکل بیضی با کلید میانبر ۲٦ وجود دارد.

۱ – تایمر ها :

برای برنامه نوسی PLCهای دلتا ، دستوراتی بر پایه زمان سنجی موجود است که در این قسمت به معرفی برخی از از آنها میپردازیم.

✓ دستور TMR:

اولین دستور و پر کاربرد ترین دستور زمانسنجی ، دستور تایمر ( TMR ) میباشد. با استفاده از این دستور میتوان تایمرهای تاخیر در وصل یا تاخیر در قطع را ساخت .

\*\*میخواهیم برنامه ای بنویسیم که بوسیله ی آن با فعال کردن ورودی ( مثلا ۲۰ ) بعد از ۱۰ ثانیه خروجی ۲۰ فعال شود.

Relay Type

با زدن این آیکن وارد صفحه ی APPLICATION INSTRUCTIONs میشوید. این پنجره شامل تمامی دستورات موجود در برنامه WPLSOFt جهت برنامه نویسی میباشد. ( شکل زیر )

<del>\$</del> #	Dv	p1[كارخانه كفش] - Delta WPLSoft - [Ladder Diagram Mode]	- 🗆 🛛
🔚 Eile Edit Compiler Comments Search View	Communication Options Wizard W	indow Help	_ B ×
🗋 🖻 🗃 🗃 🕥 💿 🕺 🐘 🛍 🍠 😘	🔍 ९ ९ 🔞 🛛 🐺 🗟 🔿	1	
Relay Type 🔥 詐 診 鹊 鹊 醫 😭 😥	雨 詩 前 縱 柿 柿 徳 講	新 🔤 🏦 💁 与 🔶 📾 🖂 👫 👔 🎥 📾	
H I I I I I I I I I I I I I I I I I I	* 🖻 🗑 🔿 🗑 📓 🕷	端 옷 욺 암 팥 둑 國 수 수 몸 속	
	SVY Function Loop Control Transmission Comparison Ford Arithmetic Operations Ford Arithmetic Operations Ford Arithmetic Operations Ford Arithmetic Operations Ford Arithmetic Operation Serial 10 Bask Instructions Fording Point Operation Additional Instructions Additional Instructions Additional Instructions Additional Instructions Additional Instructions Contact Type Logic Operation Contact Type Cogneration Gray Code Floating Point Contact Type C Specific Bit Control Additional Point Operation Contact Type Cogneration Gray Code Floating Point Contact Type C Specific Bit Control Adsolute Comparison	API List Image: Constraint of the second s	×
			1

همانطور که ملاحظه میکنید در سمت چپ پنجره باز شده لیستی به صورت طبقه بندی شده از دستورات موجود است.

۱ – در صورتی که شماره ی دستور مورد نظر را میدانید در این قسمت آنرا وارد کنید اما حفظ کردن شماره ی دستورات کمی مشکل بوده و امکان اشتباه در آن زیاد است.

۲ – در این قسمت میتوانید نام دستور مورد نظر خود را تایپ کنید تا نماید داده شود.

۳ – در نظر داشته باشید که تایمر ها دارای تیغه ی داخلی هستند و بعد از محاسبه ی زمانی که کاربر قرار میدهد، این تیغه تغییر وضعیت میدهد.

٤ – برای تایمر مورد استفاده باید مقداری را برای شمارش اختصاص داد. این مقدار میتواند عددی ثابت ( K ) یا عددی متغیر (D) باشد. تمامی اعداد ثابت در دلتا با پیشوند K نشان داده میشوند.

٥ - شماره این تیغه ها طبق جدول موجود در فایل منوال مختص به هر PLC موجود است.

۲ – برای مقدار دهی باید در نظر داشته باشید که تایمرها طبق شماره ی تیغه مورد استفاده ( به فایل منوال PLC مراجعه شود ) دارای ضرایب متفاوت هستند. برای مثال در عکس زیر قسمتی از جدول خصوصیات تایمرهای PLC نوع "۱٤SS۲" را ملاحظه میکنید.

	-				
		Timer	100ms (M1028=ON, T64~T126: 10ms)	T0~T126, 127 points, (*1) T128~T183, 56 points, (*1) T184~T199 for Subroutines, 16 points, (*1) T250~T255(accumulative), 6 points	Total
	т		10ms (M1038=ON, T200~T245: 1ms)	(*1) T200~T239, 40 points, (*1) T240~T245(accumulative), 6 points, (*1)	256 points
Bit Contacts			1ms	T246~T249(accumulative), 4 points, (*1)	

۷ - در این قسمت میتوانید توضیحات مربوط به دستور را ملاحظه کنید.

۸ – در پایان با زدن گزینه OK ، دستور نوشته خواهد شد.

نکته: مهندسین گرامی در نظر داشته باشید که با قطع جریان ورودی دستور تایمر ( TMR )، مقدار تایمر صفر میشود.

مانند سایر برند ها در PLC های دلتا نیز حافظه های داخلی وجود داشته که در برنامه نویسی بسیار کاربردی هستند.

این حافظه ها که با حرف M نمایش داده میشوند نیز مانند تایمرها ، در جدول مشخصات هر PLC ، جهت مراجعه موجود هستند. ( مانند شکل زیر PLC ۱٤S۲ )

	_				
	A	M Auxiliary relay Lat	General	M0~M511, 512 points, (*1) M768~M999, 232 points, (*1) M2000~M2047, 48 points, (*1)	Total
	м		Latched	M512~M767, 256 points, (*2) M2048~M4095, 2048 points, (*2)	4096 points
		Special	M1000~M1999, 1000 points, some are latched		

در ادامه بیشتر به بحث در رابطه با حافظه ها خواهیم پرداخت.

با فعال سّدن این ورودی حافظه ی داخلیM سک میشود			SET	M٠
با استفاده از نیغه ی این حافظه جریان ورودی تایمر قطع نخواهد سّد	TN	1R	T.	K)++
نَيْغه نَالِمِر بِس از نَبْتَ زِمان ۱۰ نَالَيْه بِسَنَّه سَده و خَرُوجي ۷۰ را سَتَ مَيِكَد			SET	Y•
	با فعال سّدن این ورودی حافظه ی داخلیM ست میسّود با استفاده از تیغه ی این حافظه جریان ورودی تایمر قطع نخواهد سّد تیغه تایمر پس از تیت زمان ۱۰ تانیه بسته سّده و خروجی ۲۰ را ست میکند	با فعال سّدن این ورودی حافظه ی داخلی M سک میسّود با استفاده از تینه ی این حافظه جریان ورودی تایمر قطع نخواهد سّد TM تینه تایمر پس از تبت زمان ۱۰ تانیه بسته سّده و خروجی ۲۰ را ست میکند	با فعال سّدن این ورودی حافظه ی داخلی M سک میسّود با استفاده از تیغه ی این حافظه جریان ورودی تایمر قطع نخواهد سّد TMR تیغه تایمر پس از تبت زمان ۱۰ تانیه بسته سّده و خروجی ۲۰ را ست میکند	با فعال شدن این ورودی حافظه ی داخلی M سک میشود SET با استفاده از تینه ی این حافظه جریان ورودی کایمر قطع نخواهد شد TMR T۰ SET SET

از این پس برنامه نویسی را به صورت تایپ کردن انجام خواهیم داد تا سرعت برنامه نویسی را افزایش دهیم. دستورات نوشتاری در قسمت قبلی آموزش در جدول نمایش داده شد.

در اینجا از T۰ استفاده شده است که ضریب ۱۰۰ms دارد. نذا در برنامه از عدد k۱۰۰ استفاده میکنیم.

وقتی عدد ثابت ۱۰۰ در ضریب ۱۰۰ میلی ثانیه ضرب شود، حاصل ۱۰ ثاینه خواهد شد.

LD X  $\boldsymbol{\cdot} \to \,$  SET M  $\boldsymbol{\cdot} \to \,$ 

 $\text{LD M} {\boldsymbol{\cdot}} \to \text{TMR T} {\boldsymbol{\cdot}} \ {\boldsymbol{K}} {\boldsymbol{\cdot}} {\boldsymbol{\cdot}} \to$ 

 $\mathsf{LD}\:\mathsf{T}{}^{\scriptscriptstyle\mathsf{I}}\to\:\mathsf{SET}\:\mathsf{Y}{}^{\scriptscriptstyle\mathsf{I}}\to$ 

## ✓ دستور ATMR:

دومین دستور پر کاربرد در میان دستورات زمانسنجی، دستور ATMR میباشد که تا حدودی شبیه به دستور TMR است، با این فرق که از این دستور میتوان مستقیما خروجی گرفت.

میخواهیم مثال قبلی را با این دستور برنامه نویسی کنیم:



فرم نوشتاری برنامه بالا:

 $\begin{array}{l} \mathsf{LD} \ \mathsf{X} \overset{}{\cdot} \to \ \mathsf{SET} \ \mathsf{M} \overset{}{\cdot} \to \\ \mathsf{LD} \ \mathsf{M} \overset{}{\cdot} \to \ \mathsf{ATMR} \ \mathsf{T} \overset{}{\cdot} \ \mathsf{K} \overset{}{\cdot} \overset{}{\cdot} \overset{}{\cdot} \to \ \mathsf{SET} \ \mathsf{Y} \overset{}{\cdot} \to \end{array}$ 

√ دستور TRD:

این دستور برای خواند زمان داخلی PLC استفاده میشود.

باید در نظر داشت که این دستور با استفاده از این دستور تعداد ۷ رجیستر اشغال میشود.

نکته: در PLCهای دلتا حافظه های ۱۲ بیتی ای وجود دارد که میتوانند همانند ظرفی که مقداری درونشان ریخته میشود، عمل کند. به این حافظه ها رجیستر گفته میشود و با حرف D نشان داده میشوند.

این رجیستر ها نیز مانند حافظه های داخلی، دارای ترتیبی میباشند که در فایل منوال هر PLC موجود است.

برای مثال در PLC ۱٤SS۲ داریم:

Word Register		Data	General	D0~D407, 408 words, (*1) D600~D999, 400 words, (*1) D3920~D4999, 1080 words, (*1)	
	D		Latched	D408~D599, 192 words, (*2) D2000~D3919, 1920 words, (*2)	Total 5000 points
			Special	D1000~D1999, 1000 words, some are latched	
			Index	E0~E7, F0~F7, 16 words, (*1)	]

برای نمونه در مثال های تایمر گفته شده در بالا ، میتوان به جای عدد ثابت ۲۰۰ K۱ که در دستور تایمر قرار دادیم از رجیستر D۰ استفاده کنیم. در این حالت با قرار دادن آدرس D۰ در HMI به صورت NUMERIC ENTERY ، میتوان هر مقداری را در تایمر قرار داد. در ادامه به بررسی بیشتر رجیسترها خواهیم پرداخت. میخواهیم تاریخ داخلی PLC را بخوانیم. برای این کار کافیست دستور TRD را به شکل زیر بنویسیم.

	D.
[[	END
ل با فعال شدن X۰ تاریخ PLC خوانده شده و در رجیسترهای D۰ تا D۱ ریخته میشود. محتوای این رجیستر ها :	در این مثال عبارتند از
	D۰ : D۰
های هفته	D۱ : روز
	D۲ : D۲
	D۳ : هفته
ث	D٤ : ساع
4	D۵ : دقيق

D٦ : ثانيه

پس از نوشتن این برنامه، میتوان رجیستر ها را در HMI فراخوانی کرد و تاریخ PLC را مشاهده نمود.

## ✓ دستور TWR:

این دستور عکس دستور TRD بوده و برای تنظیم زمان صحیح PLC استفاده میشود. برای استفاده از این دستور به شکل زیر عمل میکنیم:

X+				
			TWR	D۲۰
X•				
├──┤↑├──┬	سال ۲۰۰۳ -	 моч	ĸ٣	DT•
	سە شنبە -	 моч	KT	DTI
	أڭوست	 МОМ	KA	DYY
		hou		
	زور ۱۹م	MUV	K19	DTT
	Colu	MOV	K)A	DYF
			IX IW	
	دفقه	моу	HTV	DT۵
	ئانيە	моч	К <b>т∙</b>	D۲۶
		L		
				END
1				

با فعال شدن X۰ تاریخ سه شنبه ۱۹ آگوست سال ۲۰۰۳ ساعت ۱۹:۲۷:۳۰ در حافظه ی PLC ذخیره میشود.

## ✓ دستور HOUR :

این دستور به ساعت سنج معروف است. با استفاده ازاین دستور میتوان میزان روشن یا خاموش بودن یک خروجی را بر حسب ساعت دید. به مثال زیر توجه کنید:

ו			
		SET	YI
Y1			
	HOUR K)+	D•	M•
M•			
		RST	YI
			END

در این مثال با فعال شدن ۲۰ ، خروجی ۲۰ (مثلا واتر پمپ ) روشن میشود. دستور HOUR بعد از گذشت ۱۰ ساعت، M۰ را فعال میکند و باعث خاموش شدن پمپ جهت سرویس میشود.

در این دستور مقدار ساعت در رجیستر D۰ و مقدار ثانیه در رجیستر D۱ ذخیره میشود.

✓ دستور TCMP :

از این برای مقایسه زمان داخلی PLC با زمان تنظیمی استفاده میشود.

به این مثال توجه کنید:

X•						
				TRD	D٠	
M)•••						
-+	TCMP KI+	K) <b>Q</b>	K۳۰	D۴	M٠	
	۲۰۰ کمتر از زمان تنظیمی ۱۰:۱۵:۳۰ بود این خروجی فعال میشود   M۱	ان داخلی یی ال سی	- اگرز،		—_( Y•	
	برابر با زمان تنظیمی ۱۰:۱۵:۳۰ بود این خروجی فعال میشود ۲۲	ان داخلی یی ال سی	- اگرز،		— ( Y)	
	يېسَتَر از زمان تَنظيمي ٣٠:١٥:١٠ بود اين خروجي فعال ميسّود	ان داخلی یی ال سی	- اگرزم		—( Y <b>t</b>	
					END	

توضیح : در خط اول این برنامه تاریخ داخلی PLC خوانده شده و درون رجیستر های D۰ تا D۰ ریخته میشود.

در نظر داشته باشید که

D٤ ساعت

oD دقيقه

D٦ ثانیه را نشان میدهند و ما میخواهیم ساعت ، دقیقه و ثانیه را مقایسه کنیم. لذا در خط دوم از D٤ استفاده میکنیم که شروع ساعت، دقیقه و ثانیه است.

برای بررسی صحت کارکرد برنامه میتوان آنرا بصورت شبیه ساز (SIMULATOR ) به ترتیب زیر اجرا کرد.

برای حالت شبیه ساز به ترتیب زیر عمل میکنیم.

░ 櫽 酋 堂 Ճ і 3\_8) 및 ♥ 4>300 ₽₽₽ (2) % % ≚ ₴ ₸ थ (1>€

با فعال کردن این گزینه میتوان برنامه را به صورت شبیه سازی شده و بدون نیاز به سخت افزار، تست کرد.

برای آپلود و دانلود کردن برنامه بر روی PLC باید از کابل سریال (RS۲۳۲) استفاده کرد. در سری جدید لپ تاپ ها این درگاه وجود ندارد و فقط بر روی لپ تاپ های صنعتی تعبیه شده است لذا برای برقراری ارتباط میان PC و PLC میتوان از تبدیل RS۲۳۲ به USB استفاده کرد. این مبدلها ساخت کمپانی دلتا بوده و براحتی با کامپیوتر و لپتاپ ارتباط برقرار میکند.

پایان قسمت دوم مقصودی ۹۱۲۷۲۱۸۰۱۱ ،