

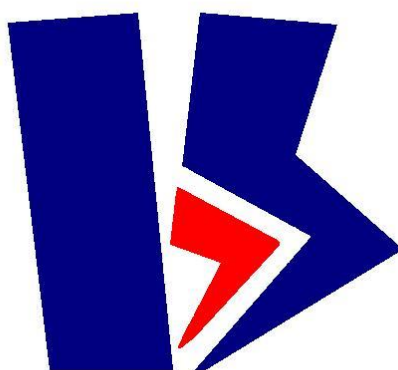
مرجع کار با برنامه مانیتورینگ کادفریم

محصولی از:

شرکت کنترل‌های صنعتی کاد

دی ۱۳۹۵

www.KadControls.com



۱	مقدمه	۷
۲	آشنایی با فایل های INI	۹
۲/۱	ساختار فایل INI	۹
۲/۲	فرمت استاندارد فایل INI به صورت عمومی در ویندوز	۱۰
۲/۳	فرمت استاندارد فایل INI به صورت اختصاصی در کادفریم	۱۱
۲/۴	فرمت نام Section برای کنترل های کادفریم در فایل INI	۱۲
۲/۵	فرمت نام پارامتر Key برای اتریب کنترل های کادفریم در فایل INI	۱۴
۲/۶	مقادیر (Values) مجاز برای پارامترهای یک اتریب کادفریم در فایل INI	۱۴
۳	آشنایی با فایل های DCT	۱۵
۳/۱	فرمت فایل DCT	۱۵
۳/۲	فرمت HwAccessPath برای هاردورهای مختلف	۱۷
۳/۲/۱	فرمت HwAccessPath در OPC	۱۷
۳/۲/۲	فرمت HwAccessPath در SRL (Modbus RTU)	۱۸
۳/۳	فرمت NetworkAccessPath تگ در DCT	۱۹
۳/۳/۱	قوانین NetworkAccessPath	۲۰
۳/۳/۲	تنظیمات برای ایجاد کادفریم سرور یا کلاینت در شبکه	۲۰
۴	نحوه استفاده از دو برنامه کادفریم سرور و کلاینت برای آشنایی با نحوه کار شبکه در کادفریم	۲۰
۵	آشنایی با عملکرد انجین دیتاسنتر کادفریم	۲۴
۵/۱	نحوه آپدیت تگ های سریال توسط انجین دیتاسنتر	۲۴
۵/۲	آشنایی با KadFrame Tag Scout	۲۵
۶	معرفی اشکال گرافیکی کادفریم	۲۶

۲۹.....	نحوه تغییر سایز و جابجایی کنترل های گرافیکی در کادفریم	۷
۲۹.....	ویرایش مشخصات گرافیکی کنترل در INI به صورت دستی	۷/۱
۲۹.....	انتخاب نوع اکچواتور و سیگنالهای کنترلی آن از داخل کادفریم	۸
۳۰.....	Device Operation mode	۸/۱
۳۱.....	Actuator Type	۸/۲
۳۲.....	Prcntg Of H/L	۸/۳
۳۲.....	Auto Calc LoadInWay	۸/۴
۳۳.....	No Taring	۸/۵
۳۳.....	ویرایش مشخصات گرافیکی کنترل در AttribScout	۸/۶
۳۳.....	نحوه جابجا کردن کنترل گرافیکی در کادفریم با ماوس	۸/۷
۳۴.....	نحوه تغییر سایز کنترل گرافیکی در کادفریم با ماوس	۸/۸
۳۵.....	خاصیت Snap در حین تغییر سایز یا جابجایی کنترل گرافیکی	۸/۹
۳۵.....	ساخت اتومات فایل DCT برای تگ های OPC	۹
۳۸.....	آشنایی با فایل های KAC.sys	۱۰
۴۰.....	رمزنگاری محتویات فایل KAC	۱۰/۱
۴۰.....	فرمت کلی رمزنگاری اطلاعات توسط کادفریم	۱۰/۲
۴۱.....	ایجاد اولین یوزر Kad-Admin	۱۰/۳
۴۲.....	مدیریت یوزرها توسط کادفریم (تکمیل شود)	۱۰/۴
۴۳.....	قسمتهای مختلف پنجره Login	۱۰/۴/۱
۴۴.....	مراحل ساخت اپلیکیشن	۱۱
۴۴.....	مرحله اول : نقشه P&I	۱۱/۱
۴۴.....	مرحله دوم : تهیه لیست تجهیزات (DeviceList)	۱۱/۲

۴۶.....	مرحله سوم: تهیه لیست ورودی خروجی	۱۱/۳
۴۷.....	مرحله چهارم : تهیه فایل ini	۱۱/۴
۵۰.....	مرحله پنجم : تنظیمات چیدمان گرافیکی برنامه	۱۱/۵
۵۰.....	مرحله ششم : تهیه فایل Sti	۱۱/۶
۵۲.....	مرحله هفتم : تهیه فایل DCT	۱۱/۷
۵۲.....	اساین کردن متغیرها به اتریبهای مناسب	۱۱/۸
۵۳.....	تست IO	۱۱/۹
۵۴.....	تعریف تگ های پروسس و شیفت و فرمولاسیون	۱۱/۱۰
۵۴.....	تست اجرای پروسه اتوماتیک	۱۱/۱۱
۵۴.....	ثبت در بانک اطلاعاتی	۱۲
۵۵.....	تنظیمات ثبت در بانک اطلاعاتی	۱۲/۱
۵۶.....	مراحل ایجاد یک جدول با مثال	۱۲/۲
۵۸.....	تگهای محاسباتی	۱۳
۵۸.....	نحوه معرفی و کار با تگ های محاسباتی	۱۳/۱
۶۰.....	تگ های نگهدارنده (با حافظه ماندگار)	۱۴
۶۰.....	نکات تنظیمات	۱۵
۶۰.....	Render DO2DI	۱۵/۱
۶۱.....	معرفی AutoConnect to Real	۱۵/۲
۶۲.....	معرفی Log در کادفریم	۱۵/۳
۶۲.....	فرمت فایل Log	۱۵/۳/۱
۶۴.....	تغییرات دستی و یا اتومات در کالیبراسیون	۱۵/۳/۲
۶۴.....	تغییر اطلاعات دارای سطح دسترسی	۱۵/۳/۳

۶۵.....	۱۵/۳/۴	مفهوم Clean Up در فایل Ini
۶۸.....	۱۵/۳/۵	نکات مهم در مورد فولدر لاگ
۶۹.....	۱۵/۴	تنظیمات کنترلی اشیا
۶۹.....	۱۵/۴/۱	کالیبراسیون آنالوگ
۷۴.....	۱۵/۴/۲	Graphic
۷۵.....	۱۵/۴/۳	Actuator Setting
۷۶.....	۱۵/۴/۴	General Setting
۷۶.....	۱۵/۵	صفحه اکشن
۷۷.....	۱۶	قابلیت About Box
۷۸.....	۱۷	قابلیت دیکشنری
۸۲.....	۱۸	تشریح اتریب های کنترل های گرافیکی کادفریم
۸۳.....	۱۸/۱	اتریب های عمومی (مشترک)
۸۶.....	۱۸/۲	Belt
۹۲.....	۱۸/۳	Pump
۹۲.....	۱۸/۴	Valve
۹۳.....	۱۸/۵	Vessel
۹۸.....	۱۸/۶	HeatZone
۹۹.....	۱۸/۷	Helix
۹۹.....	۱۸/۸	KLabel و KButton
۱۰۸.....	۱۸/۹	KCheckBox
۱۱۰.....	۱۸/۱۰	KComboBox
۱۱۲.....	۱۸/۱۱	kEditBox

۱۱۳.....	KPicture	۱۸/۱۲
۱۱۸.....	KSlider	۱۸/۱۳
۱۲۰.....	PrcAlarmSingle	۱۸/۱۴
۱۲۲.....	RotaryMixer	۱۸/۱۵
۱۲۳.....	RotaryWheel	۱۸/۱۶
۱۲۴.....	SprayDryer	۱۸/۱۷
۱۲۸.....	Design Mode	۱۹
۱۲۹.....	لوله های انتقال	۲۰
۱۳۱.....	Trends	۲۱
۱۳۲.....	نحوه ساخت و نمایش یک ترند در نرم افزار کادفریم.	۲۱/۱
۱۳۲.....	ترند آفلاین	۲۱/۱/۱
۱۳۴.....	ترند آنلاین	۲۱/۱/۲
۱۳۵.....	قابلیت ذخیره فرمول در کادفریم بیسیک	۲۲
۱۳۸.....	قابلیت آپلود فرمول از PLC	۲۳

از زمان تشکیل کادفریم که به زبان C تحت داس بود، ارتباط و آدرس دهی IO در سخت افزار ها به صورت هارد کد در بطن برنامه بود. یعنی ماجول مانیتورینگ و حتی ماجول کنترل به صورت مستقیم با هاردور برای تبادل اطلاعات درگیر بود. در راستای اصولی کردن کار برنامه و ماجولار کردن بخش های مختلف برنامه، ایجاد ماجول ارتباط با انواع هاردور به صورت مستقل از کار کنترل، در اوایل سال ۱۳۸۸ در برنامه قرار گرفت.

نقشه از این قرار شد که یک آرایه از متغیرها به نام تگ در حافظه ایجاد گردد. سپس یک ماجول به نام HardWareLinker به صورت مستقل از کار مانیتورینگ و کنترل در کادفریم، وظیفه تبادل اطلاعات بین تگ ها و IO هاردوری را به عهده بگیرد. ماجول HwLinker به این شکل عمل می کند که اگر تگی قرار است از HW خوانده شود، مقدار آن را به صورت اتومات از HW خوانده و در تگ کپی می کند. سپس تگ خبر (مسیج) تغییر در مقدارش را به صورت اتومات به پنجره (HWND) تمام استفاده کنندگانش (از جمله به کانتینر اتریب هایی که این تگ به آنها اساین شده است) می دهد. در ضمن ماجول HwLinker برای تگی که قرار است مقدارش در HW نوشته شود، به این شکل عمل می کند که در صورت تغییر در مقدار تگ (Message Base) یا در فواصل زمانی مشخص (Pooling Base)، مقدار تگ را در HW می نویسد. نسخه اولیه دیتاسنتر در 880213 متولد شد ☺ و از آن زمان تا کنون پیشرفت های زیادی کرده است.

خب تا اینجا آرایه ای از تگ به نام دیتاسنتر (DataCenter) داریم که به صورت اتومات در ارتباط با هاردور و یا هاردورها بوده و همیشه آپدیت نگه داشته می شود. این مرکز اطلاعات (دیتاسنتر) همیشه آماده استفاده بخش های مختلف برنامه است. لیست آرایه تگ، مستقل از کادفریم، از خارج از کادفریم از طریق فایل DCT به برنامه کادفریم معرفی می شود و همیشه قابل ادیت و آپدیت است.

دیتاسنتر دو نوع پروتکل نرم افزاری OPC و Modbus و یک نوع تگ بدون آدرس هاردوری را ساپورت می کند. تگ بدون آدرس هاردوری می تواند به عنوان یک مموری در دیتاسنتر استفاده شود که با هیچ هاردوری در ارتباط نیست. یکی از کاربردهای مموری تگ، اتصال مقدار دو اتریب با اساین یک تگ به دو اتریب مد نظر است. مثلاً برای برقراری لوپ بک می توان دو اتریب DO و DI را به یک تگ اساین کرد.

برنامه کادفریم از چندین کنترل تشکیل شده است. کنترل ها (مثل Vessel, Valve) برای نگهداری اطلاعات خود از آرایه ای از اتریب (Attrib) استفاده می کنند. اتریب این قابلیت را دارد با لینک شدن به تگ از طریق فایل INI، آپدیت دیتای خود را به صورت اتومات به تگ بسپارد. به این معنا که با ایجاد هر نوع تغییر در اتریب (مثلا تغییر اطلاعات کالیبراسیون)، خود اتریب مقدار دیتای جدید خود را به صورت اتومات به تگ لینک شده انتقال می دهد. تبادل اطلاعات تگ هم که توسط ماحول HwLinker انجام می شود.

در اینجا لازم به ذکر است که نقش یک تگ به تنهایی در کادفریم قابل تشخیص و درک نیست. مثلا تگی با نام Analogue1 را که در DCT تعریف شده است در نظر بگیرید (این اسم ممکن است بعدا در همان فایل DCT عوض شود). خب به نظر شما نقش Analogue1 چیست؟ نقش تگ وقتی مشخص می شود که آن را به یک اتریب اساین کنیم. مثلا به Vessel1.Level. پس از این اساین می فهمیم که Analogue1 همان لول مخزن یک را نشان می دهد. درک این مطلب از این لحاظ حایز اهمیت است که برنامه نویس از تعداد و نوع تگ هایی که قرار است بعدا توسط طراح DCT ایجاد شود خبر ندارد. ولی اتریب ها همیشه از پیش تعریف شده و مشخص هستند و پس از ریلیز برنامه قابل تغییر نیستند.

کادفریم دارای سیستم حق دسترسی است. اطلاعات مربوط به کاربران (Users) در فایلی به نام ProjectID_KAC.sys نگهداری می شود. فرمت فایل KAC همان فرمت فایل INi است. پسوند این فایل از آن جهت sys. قرار داده شده است که نتوان به آسانی آن را توسط ادیتور (مثل Notpad) باز کرد. پسوند sys. در اصل مبین فایل های اجرایی سیستم است.

و اما خلاصه؛ برنامه کادفریم اطلاعات نحوه ارتباط با هاردورها را از فایل DCT و اطلاعات کنترل ها را از فایل INI می گیرد. در این مقاله با فرمت و نحوه ایجاد و ویرایش انواع فایل های مورد استفاده توسط کادفریم آشنا خواهیم شد. البته عملکرد انجین دیتا سنتر نیز شرح داده می شود.

۲ آشنایی با فایل های INI.

این فایل ها در حقیقت فایل های متنی (TXT) ساده هستند، که بوسیله هر برنامه ویرایشگر متن مثل Notepad می توان آنها ایجاد و ویرایش کرد.

فایل های INI بوسیله ویندوز و برنامه های تحت ویندوز (مثل کادفریم) مورد استفاده قرار می گیرند تا تنظیمات برنامه در آن نوشته شود و یا از آن خوانده شود. تنظیمات ممکن است توسط خود برنامه و یا از طریق ویرایش دستی فایل INI در یک ادیتور، ایجاد شوند. تنظیمات معمولاً در ابتدای اجرای برنامه از فایل INI خوانده شده و مورد استفاده قرار می گیرد. کلمه INI مخفف initialization است. اگر در تنظیماتی که در ابتدای اجرای برنامه از فایل INI خوانده می شوند، در حین اجرای برنامه از طریق ادیتور خارجی تغییری ایجاد شود، آن تغییرات در برنامه منعکس نخواهد شد تا اینکه برنامه استفاده کننده INI یک بار بسته و مجدد استارت شود. مثال این نوع تنظیم اساین کردن یک تگ به اتریب در کادفریم می باشد. البته برنامه ممکن است به بعضی تنظیمات فقط وقتی به آنها نیاز دارد در INI مراجعه کند. تغییر در این تنظیمات در حین اجرای برنامه نیازی به ری استارت برنامه ندارد.

۲,۱ ساختار فایل INI

فایل INI شامل یک یا چند بخش است. هر بخش با یک نام بخش [section name] آغاز می شود که ممکن است، هیچ یا چندین پارامتر در هر سطر پس از آن وجود داشته باشد. نمای کلی محتویات فایل INI به شکل زیر است:

```
[Section1]
KeyName11=Value11
KeyName12=Value12
...
[Section2]
KeyName21=Value21
KeyName22=Value22
...
```

۲,۲ فرمت استاندارد فایل INI به صورت عمومی در ویندوز

- در هر سطر فقط نام یک Section می تواند قرار گیرد.
- همه کلمه یا کلمات ترکیبی که داخل دو کروشه قرار می گیرد، نام Section را تشکیل می دهد.
- در هر سطر فقط یک پارامتر می تواند افزوده شود.
- هر پارامتر از دو جزء KeyName و Value که با مساوی بین آنها، از هم جدا می شوند، تشکیل می شود.
- همه کلمه یا کلمات ترکیبی که قبل از مساوی قرار می گیرد، نام پارامتر (KeyName) را تشکیل می دهد.
- همه کلمه یا کلمات ترکیبی که بعد از مساوی قرار می گیرد، مقدار پارامتر (Value) را تشکیل می دهد.
- پارامتر می تواند Key بدون Value داشته باشد. یعنی سمت راست مساوی خالی باشد. در این صورت مقدار پیش فرضی که به صورت هاردکد در برنامه استفاده کننده INI تعریف شده است، به عنوان Value در نظر گرفته می شود.
- فایل INI می تواند حاوی یک یا چند بخش از تنظیمات برنامه باشد. در این صورت مقدار پیش فرضی که به صورت هاردکد در برنامه استفاده کننده INI تعریف شده است، به عنوان Value در نظر گرفته می شود. اگر وجود Value الزامی باشد، باید توسط برنامه اعلان شود.
- نام پارامترها (KeyNames) در زیر هر بخش باید یکتا (Unique) باشد. البته اگر دو پارامتر با نام یکسان در زیر یک Section آورده شود، معمولاً برنامه ها مقدار بالاترین سطر پارامتر را می خوانند و وجود پارامترهای تکراری را که ممکن باعث خطای شخص ادیتور شود را اعلان نمی کنند.
- پارامترهای هم نام می تواند در بخش های جداگانه قرار گیرد.
- نام بخش ها (Sections) در کل فایل INI باید یکتا (Unique) باشد. البته اگر دو بخش با نام یکسان در یک فایل آورده شود، معمولاً برنامه ها مقادیر بالاترین بخش را می خوانند و وجود بخش های تکراری را که ممکن باعث خطای شخص ادیتور شود را اعلان نمی کنند.
- قانون حاکم در کادفریم: برنامه کادفریم به کوچکی و بزرگی حروف در INI حساس است.

مرجع کار با برنامه مانیتورینگ کادفریم

✓ جهت اطلاع بدانید ویندوز از چندین فایل INI استفاده می کند تا اطلاعات پیکر بندی کامپیوتر را ذخیره کند. فایل هایی که این اطلاعات را نگهداری می کنند عبارتند از:

WIN.INI, SYSTEM.INI, PROGMAN.INI, PROTOCOL.INI, SHARED.INI, WINFILE.INI,
CONTROL.INI, SCHDPLUS.INI

۲.۳ فرمت استاندارد فایل INI به صورت اختصاصی در کادفریم

برنامه کادفریم از چندین کنترل تشکیل شده است.

قسمتی از این کنترل ها دارای گرافیک خاص یک کنترلر صنعتی می باشد. از جمله آنها می توان به Vessel, Belt, ... اشاره کرد.

قسمتی دیگری از کنترل ها برای تبادل اطلاعات پروسس استفاده می شود. از جمله آنها می توان به کنترل شیفت و فرمول و لوپ ها اشاره کرد.

وجه اشتراک همه این کنترل ها این است که همه آنها برای نگهداری اطلاعات خود از اتریب (Attrib) استفاده می کنند. اتریب در کادفریم می تواند یکی از چهار نوع دیتاتایپ اصلی یعنی INT, FLOAT, STRING, BOOL را نگهداری کند.

یک اتریب نیز به تنهایی دارای ۵ خاصیت (Property) است که عبارت است از:

Data: اینکه مقدار جاری این اتریب چیست. (مثلا ۱۲ برای نوع عددی و true برای نوع Bool)

Tag: اینکه اتریب دیتای خود را با کدام تگ در دیتاسنتر مبادله کند.

TagTransform: اینکه اتریب در هنگام مبادله دیتای خود با تگ اساین شده، چه تغییری در دیتای مبادله شونده

دهد. کادفریم فعلا فقط ترنسفرم Negative را آن هم روی دیتای Bool ساپورت می کند.

Visible: اینکه آیا ادیتور متناظر با این اتریب در هنگام ویرایش اتریب نشان داده شود یا نه.

ASD: که مخفف Attrib Security Descriptor است حق دسترسی به اتریب را تعریف می کند.

مرجع کار با برنامه مانیتورینگ کادفریم

- برنامه کادفریم اطلاعات اتریب را از فایل INI خوانده و در INI نیز ذخیره می کند. اطلاعات هر خاصیت (Property) از اتریب به صورت جداگانه در قالب یک پارامتر (در یک سط جداگانه) نگهداری می شود.
- ممکن از هیچ تا همه خاصیت های (ورژن جاری ۵ خاصیت) یک اتریب در INI نگهداری شود.

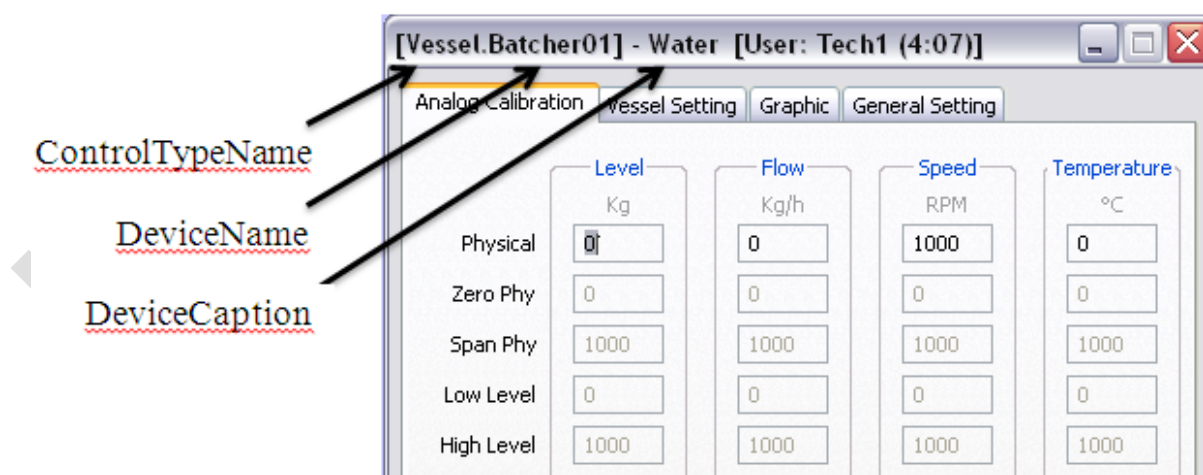
۲,۴ فرمت نام Section برای کنترل های کادفریم در فایل INI

نام بخش (Section) برای کنترل ها از دو کلمه تشکیل می شود، که شامل نام تایپ کنترل (ControlTypeName) و نام آبجکت (DeviceName) است و با یک نقطه از هم جدا می شوند.

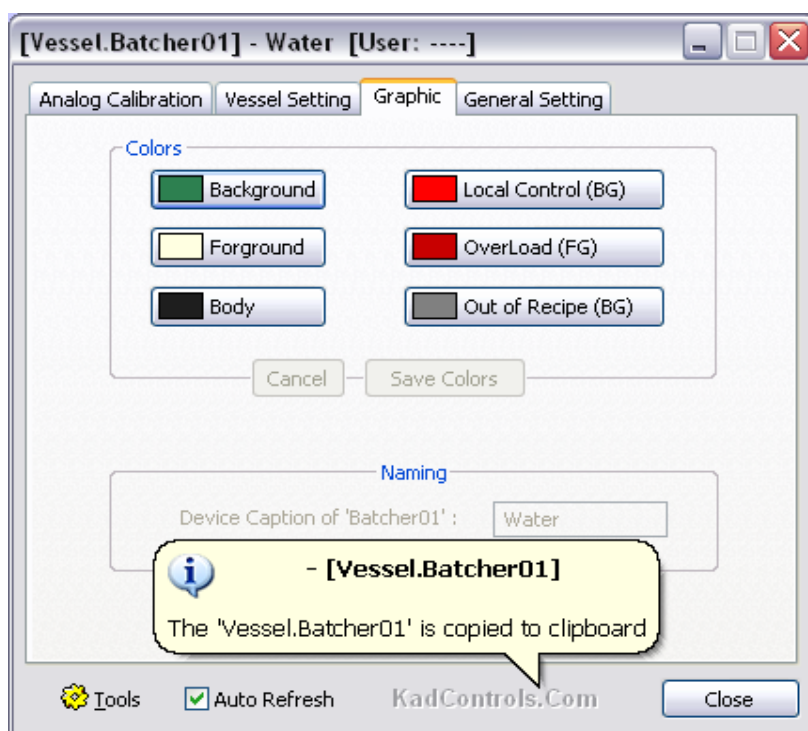
یعنی در فایل INI به شکل [ControlTypeName.DeviceName] نوشته می شود.

تمام کنترل تایپ هایی که در ورژن جاری کادفریم ساپورت می شود عبارتند از :

Vessel, Belt, DosePump, Valve, Helix, GearPump, CntrFglPump, RotaryMixer, HeatZone, RotaryWheel, SprayDryer, DBFValve, PrcGeneral, PrcShift, PrcRecipeGen, PrcRecipeStep, PrcLoop, PrcEngine, PrcEngStep, KButton, KCheckbox, KComboBox, KEditBox, KLabel, KSlider, Piping, PrcAlarmSingle, Trends,...



شکل ۱



شکل ۲

کلمه DeviceName نامی است که طراح کادفریم به صورت یونیک به هر آبجکت کنترل اختصاص می دهد. کلمه DeviceName در بالای Properties-Page کنترل های گرافیکی نشان داده می شود. در شکل ۱ عبارت نشان داده شده در بین کروشه دقیقاً برابر عبارت Section-Name کنترل در INI است. لازم به ذکر است که عبارت بعد از کروشه یعنی DeviceCaption در صورتی که در INI برای کنترل تعریف شده باشد، نشان داده می شود.

مثلاً: DeviceCaption.Data=Water

در ضمن در کادفریم یک کلید مخفی برای کپی کردن نام Section مرتبط با کنترل وجود دارد. اگر در قسمت خالی بالا یا پایین Properties-Page (مثلاً سمت چپ دکمه Close) دابل کلیک کنید، نام Section در Clipboard ویندوز کپی می شود و خبر این کار توسط یک بالون شبیه شکل ۲ نشان داده می شود.

۲,۵ فرمت نام پارامتر Key برای اتریب کنترل های کادفریم در فایل INI

نام پارامتر (KeyName) برای اتریب های کنترل از دو کلمه تشکیل می شود، که شامل نام خود اتریب و نام Property است و با یک نقطه از هم جدا می شوند.

یعنی در فایل INI به شکل `AttribName.AttribProperty=???` نوشته می شود. پس کل

`AttribName.AttribProperty` نام یک پارامتر (KeyName) در یک فایل INI را تشکیل می دهد.

کنترل ها دارای اتریبهایی با نام های گوناگون هستند. مثلا `Level, MinLevel, DiOnOff, ...`

اما `AttribProperty` همانطور که در بالا به آن اشاره شده فقط یکی از مقادیر زیر می تواند باشد:

`.Data, .Tag, .TagTransform, .Visible, .ASD`

۲,۶ مقادیر (Values) مجاز برای پارامترهای یک اتریب کادفریم در فایل INI

پارامترهایی که KeyName آنها به `Data` ختم می شود، می تواند هر نوع مقدار (Value) از جنس دیتاتایپ خود اتریب های بگیرد. مثلا دیتاتایپ اتریب `DeviceCaption` از نوع `String` است و دیتاتایپ اتریب `Level` از نوع `float` است.

پارامترهایی که KeyName آنها به `Tag` ختم می شود، فقط می تواند نام یکی از تگ هایی باشد که در فایل `DCT` ایجاد شده است. به این کار اساین تگ به اتریب می گوئیم. دیتاتایپ تگ اساین شده باید با دیتاتایپ اتریب گیرنده تگ برابر باشد. لازم به ذکر است که یک تگ می تواند همزمان به چندین اتریب اساین شود.

پارامترهایی که KeyName آنها به `TagTransform` ختم می شود، در ورژن جاری کادفریم فقط می تواند مقدار `Negative` به خود بگیرد. وجود این پارامتر در صورتی قابل قبول است که به اتریب از طریق پارامتر `Tag` تگی از نوع `Bool` اساین شده باشد. مقدار `Negative` به این معنا است که اتریب و تگ اساین شده به آن، هنگام مبادله دیتای `Bool` خود، مقدار آن را معکوس کنند.

`LocalCntrlSpeed.Tag=PaCo.DosePump.RemoteCtrl.DosePump0`

`LocalCntrlSpeed.TagTransform=Negative`

مرجع کار با برنامه مانیتورینگ کادفریم

پارامترهایی که KeyName آنها به Visible. ختم می شود، فقط می تواند یکی از مقدار Yes و یا No به خود بگیرد. نکته مهم اینجا این است که از مقداری (Value) برای این پارامتر مشخص نشود، کادفریم آن را Yes یعنی Visible بودن ادیتور اتریب فرض خواهد کرد.

برای پارامترهایی که KeyName آنها به ASD. ختم می شود، هنوز در کادفریم استفاده ای وجود ندارد!

۳ آشنایی با فایل های DCT.

فایل DCT (Data Center Tags) حاوی تگ های دیتاسنتر کادفریم است. نوع فایل DCT، متنی است و با هر ادیتوری متنی (مثلا Notepad, VisualC) قابل ایجاد و ویرایش است. دیتاسنتر کادفریم تا الان از تگ هایی که HW-Type آنها از نوع OPC, ADV(Advantech), SRL(Serial-Modbus) است، ساپورت می کند. اگر در پروژه فقط از تگ های OPC استفاده شده باشد، می توان توسط برنامه Sti2Dct-891110.exe (یا هر ورژن جدیدتر آن)، فایل DCT را از روی فایل STI ساخت. البته فایل DCT ساخته شده از روی STI، همیشه قابل ادیت و آپدیت می باشد. مثلا ما می توانیم، بعدا تگ های پورت سریال را به صورت دستی در DCT تایپ و اضافه کنیم.

۳,۱ فرمت فایل DCT

در هر خط از فایل DCT اطلاعات فقط یک تگ آورده می شود. اطلاعات خطهایی که با یک '-' و یا '/' شروع می شوند جزء کامنت حساب می شوند.

✓ یک خط نمی تواند هم تگ داشته باشد و هم کامنت.

فرمت هر سطر تگ از استاندارد CSV (Comma Separated Value) تبعیت می کند. ورژن جاری کادفریم از تگهایی که در فایل DCT دارای ۱۰ جزء (ستون) که توسط ۹ کاما ',' از هم جدا شده اند، پشتیبانی می کند.

افزودن شماره ستون به ابتدای هر مقدار اختیاری است. چون شماره ستون فقط جنبه راهنما برای ادیتور DCT است. مثلا دو مقدار "02-BOOL" و "BOOL" از دید کادفریم یکسان است.

۸ جزء یک تگ عبارت از:

01-Name, 02-DataType, 03-DefaultData, 04-HWType, 05-IOType, 06-HwAccessPath, 07-NetworkAccessPath, 08-DataStackSize

۱- Name: نام تگ که باید در کل فایل یونیک باشد. این همان نامی است که برای اساین کردن به اتریب

استفاده می شود.

۲- DataType: نوع دیتای تگ را مشخص می کند. تگ بر خلاف اتریب همه دیتاتایپ های استاندارد را از جمله

STRING, FLOAT, BOOL, INT, DATE, BYTE, DWORD, ... ساپورت می کند.

۳- DefaultData: مقدار اولیه ولو تگ را مشخص می کند. مقدار NULL معادل صفر یا False است.

۴- HW-Type: اینکه این تگ به کدام یک از سه نوع هاردور ساپورت شده در کادفریم یعنی OPC, SRL,

ADV اشاره می کند.

۵- IOType: نوع ورودی و یا خروجی بودن تگ را مشخص می کند. تگ یکی از ۳ حالت INPUT و

OUTPUT و یا DUPLEX را می پذیرد. مقدار DUPLEX یعنی اینکه تگ هم INPUT و هم OUTPUT

است. IOType از دیدگاه HwAccessPath است.

۶- HwAccessPath: آدرس مموری از هاردور است که تگ به آن اشاره می کند. فرمت HwAccessPath برای

هاردورهای مختلف متفاوت است و بعداً توضیح داده خواهد شد.

۷- NetworkAccessPath: منبع آپدیت دیتای تگ را در شبکه مشخص می کند. منبع آپدیت دیتای تگ در

شبکه، آدرس تگی از یک نسخه دیگر کادفریم (در شبکه) است. این قضیه مستقل از هاردور اکسس این تگ

است ولی باید توجه داشته باشید که منطقاً :

✓ معمولاً اگر تگی NetworkAccessPath داشته باشد، در صورت وجود HwAccessPath، آن باید از نوع

OUTPUT باشد. فرمت NetworkAccessPath بعداً توضیح داده خواهد شد.

۸- DataStackSize: انجین دیتاسنتر در کادفریم برای آپدیت یک تگ عددی از نوع Float, Int می تواند به دو

روش عمل کند. در روش اول اگر سائز استک صفر یا یک باشد، دیتای تگ به صورت لحظه ای از هاردور

خوانده شده و در تگ نوشته می شود. در روش دوم اگر سائز استک عددی بزرگتر از ۲ باشد، در ابتدای شروع کادفریم یک استک FIFO (First Input First Output) برای تگ ایجاد می شود. سپس همیشه آخرین قرائت لحظه ای به ابتدای لیست استک اضافه می شود و قدیمی ترین قرائت از آخر لیست استک حذف می شود. با افزوده شدن هر مقدار جدید به لیست استک تگ، متوسط استک در همان لحظه محاسبه شده و به عنوان مقدار تگ ست می شود.

۳,۲ فرمت HwAccessPath برای هاردورهای مختلف

عمده تفاوت در تعریف یک تگ در بین پروتکل های مختلف، تفاوت در نوع تعریف HwAccessPath می باشد. مشخصه HwAccessPath که آدرس نهایی محلی بر روی سخت افزار است که قرار است تگ با آن به صورت اتوماتیک تبادل اطلاعات کند.

۳,۲,۱ فرمت HwAccessPath در OPC

تگ های OPC تعریف شده در دیتاسنتر کادفریم، خود از تگ های دیتا سنتر OPC-Server استفاده می کند! لذا محل اصلی HwAccessPath در دل دیتاسنتر OPC-Server است. بله، OPC نیز دارای دیتاسنتر است. فایل دیتاسنتر OPC همان فایل STI(Siemens Tag Info) است. تگ دیتاسنتر OPC را نیز ما با نام سیمبول می شناسیم. اینکه یک سیمبول OPC-Server به چه آدرسی اشاره می کند، در فایل STI در ستون "Runtime Name" مشخص شده است.

در فایل DCT برای تگ های OPC در ستون HwAccessPath، استثنأ به جای آدرس هاردوری تگ، اطلاعات گروه های ایجاد شده برای ارتباط با OPC-Server آورده می شود. برای ارتباط با OPC-Server ابتدا لازم است گروهی از طرف کلاینت (مثلا کادفریم) به سرور معرفی شود. (لطفا نحوه کار با OPC-Scout را مطالعه کنید). گروه معرفی شده از طرف کلاینت دارای مشخصه های مختلفی از جمله UpdateRate و DeadBand است. سپس برای مانیتور کردن، سیمبول ها به گروه ایجاد شده، اضافه می شوند.

مرجع کار با برنامه مانیتورینگ کادفریم

کادفریم برای همه تگ های OPC که قسمت ماقبل آخرین نقطه در اسم آنها مساوی است، یک گروه ایجاد می کند. سپس آن تگ ها را به گروه ایجاد شده اضافه می کند. مشخصه گروه ایجاد شده در اولین تگ آن گروه در قسمت HwAccessPath با فرمتی شبیه این مثال یعنی UR100-DB2.5 آورده می شود. در مثال مذکور UpdateRate=100ms و DeadBand=%2.5 است.

۳,۲,۲ فرمت HwAccessPath در SRL (Modbus RTU)

فرمت HwAccessPath برای مادباس در قالب یک مثال به شکل زیر است:

MBRTU16-DA7-REG40001-FP0-UR100-TOB90

همانطور که از مثال مشخص است، HwAccessPath در تگ سریال دارای ۶ جزء است.

بازه هر کدام از شش جزء تگ سریال به صورت خلاصه به شرح ذیل می باشد:

MBRTU16: ModBus RTU

DA: Device Address (Range: 1 <= DA <= 247)

REG: Register Index (Range: 40000 <= REG <= 50000)

FP: Floating Point (Range: FP >= 0)

UR: Update Rate (Range: UR >= 20), Unit: milli seconds

TOB: Time Out Band (Range: TOB < UR), Unit: milli seconds

جزء اول، نوع پروتکل استفاده شده می باشد. کادفریم در سریال فقط مود مادباس RTU را ساپورت می کند.

جزء دوم، آدرس دیوایس طبق استاندارد پروتکل عددی مابین ۱ تا ۲۴۷ می تواند باشد.

جزء سوم، شماره رجیستری است که قرار است اطلاعات آن خوانده (Pool) شود. دو بازه 40000 و 50000 ساپورت می شود.

جزء چهارم، محل نمایش عدد اعشار می باشد. لازم به ذکر است اعداد از روی دیوایس مخصوصا PAXها بدون عدد

اعشار در قالب یک عدد ۳۲ بیتی بدون اعشار و علامت (UNSIGNED INT) خوانده می شود. اینکه در کجای عدد

ممیز (نقطه) اعشار قرار داده شود جزء آپشن تگ است.

مرجع کار با برنامه مانیتورینگ کادفریم

جزء پنجم، بازه زمانی یا همان پریود آپدیت تگ است. انجین دیتاسنتر به صورت Pooling تگ های سریال را از دیوایس های آدرس دهی شده خوانده و آپدیت می کند.

جزء ششم، حداکثر بازه زمانی است که انجین دیتاسنتر در هنگام Pooling تگ سریال منتظر دریافت پاسخ از دیوایس آدرس دهی شده می ماند.

لطفا قبل از تعریف مشخصات تگ سریال، بخش آشنایی با عملکرد انجین کادفریم مطالعه شود.

۳,۳ فرمت NetworkAccessPath تگ در DCT

مقدار NetworkAccessPath از اجزاء تعریف تگ در DCT می باشد. مقدار NetworkAccessPath منبع آپدیت دیتای تگ (ورود اطلاعات) را در شبکه مشخص می کند. منبع آپدیت دیتای تگ در شبکه، نام تگی در یک نسخه دیگر کادفریم (در شبکه) است.

تگی که دیتای آن از طریق شبکه آپدیت می شود، از دید کادفریم حاوی تگ اصلی RegistrarTag است.

تگی که دیتای آن از طریق شبکه آپدیت می شود، از دید کادفریم تامین کننده دیتا ListenerTag است.

فرمت فیلد NetworkAccessPath به شکل `App:{SrcAppID}-Tag:{SrcTagName}-TransRate:{nn}` است.

مقدار SrcAppID نام آی-دی اپلیکشن تامین کننده دیتای تگ را مشخص میکند.

✓ اگر تگ در برنامه کلاینت تعریف شده باشد، می توان از مقدار خاص `%.server%` به عنوان SrcAppID

استفاده کرد. چون تنها سورس کلاینت سرور است. آی پی برنامه سرور (و در پی آن آی دی برنامه سرور)

از طریق INi مشخص می شود.

مقدار SrcTagName نام تگ را در اپلیکشن سورس مشخص میکند که قرار است مقدار دیتای تگ از آن تامین شود.

در صورتی نام تگ مبدا و نام تگ مقصد یکسان باشد، می توان از مقدار خاص `%.same%` به عنوان SrcTagName

استفاده کرد. در این حالت از فیلد TagName در DCT استفاده خواهد شد.

مرجع کار با برنامه مانیتورینگ کادفریم

مقدار TransRate (اختصار یافته TransferRate) پریود زمان انتقال دیتای تگ را مشخص می کند. واحد آن $100\text{ms} \times$ است.

۳,۳,۱ قوانین NetworkAccessPath

- معمولاً اگر تگی NetworkAccessPath داشته باشد، در صورت وجود HwAccessPath، مقدار IOType باید از نوع OUTPUT باشد. چون منطقی نیست که منبع دیتای یک تگ به صورت همزمان از دو جای مختلف باشد. در اینجا منظور از دو جای مختلف یکی شبکه و دیگری هاردور واقعی است.

۳,۳,۲ تنظیمات برای ایجاد کادفریم سرور یا کلاینت در شبکه

تنظیمات شبکه در فایل INI در سکشن [Network] انجام می شود.

✓ فقط یکی از کادفریم ها می تواند طبق فلگ bTcpApplicationIsServer سرور باشد.

✓ برای اجرای چند نسخه از یک کادفریم با یک آی دی در یک کامپیوتر، بعد از دادن پیغام در حال اجرا بودن برنامه، کلیک راست ماوس را نگه داشته و کلید OK را بزنید.

مقدار strTcpApplicationID در NetworkAccessPath به عنوان App:{SrcAppID} استفاده می شود.

۴ نحوه استفاده از دو برنامه کادفریم سرور و کلاینت برای آشنایی با نحوه کار شبکه در کادفریم

- ۱- ابتدا دو پکیج سرور و کلاینت را در یک کامپیوتر (و یا دو کامپیوتر متفاوت) اکسترکت کنید.
- ۲- سپس ابتدا برنامه کادفریم سرور را اجرا کنید. اگر Windows Security Alart آمد، حتما گزینه Unblock را انتخاب کنید.
- ۳- اگر IP که برای سرور در فایل INI در سکشن [Network] تعریف شده است، با IP جاری کامپیوتر متفاوت باشد، کادفریم با دادن پیغامی (طبق شکل زیر) آن را به اولین IP جاری کامپیوتر آپدیت می کند. باید IP جدید سرور برای ست کردن در کلاینت یادداشت شود.

مرجع کار با برنامه مانیتورینگ کادفریم

۴- در صورت تغییر IP سرور، قبل از اجرای کادفریم کلاینت، فایل INI کلاینت را باز کرده و مقابل strTcpServerIPAddress مقدار جدید IP سرور را می نویسیم و فایل را سیو می کنیم.

۵- حال برنامه کادفریم کلاینت را اجرا می کنیم. اگر هر دو کادفریم سرور و کلاینت برای تست در یک کامپیوتر اجرا می شوند، در هنگام اجرای کادفریم دوم، کلید مخفی (کلید راست و یا هر دو کلید شیفت کنترل) را نگه می داریم تا کادفریم دوم اجازه اجرا پیدا کند.

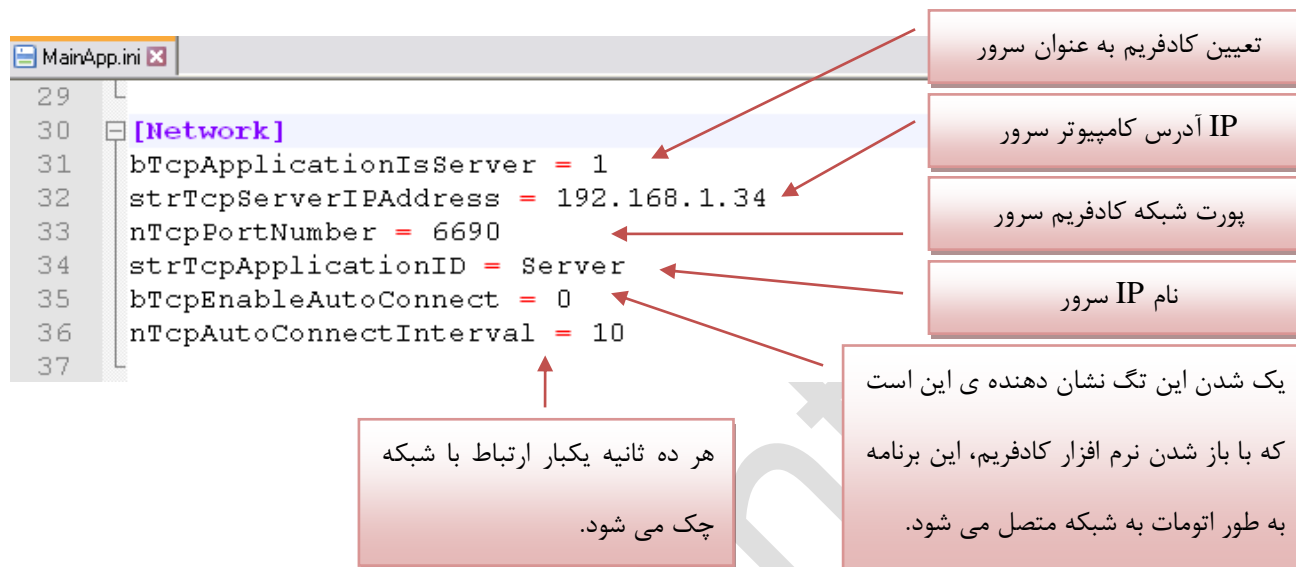


نکته: کادفریم کلاینت می تواند تا ۶۲ نسخه تکثیر یابد. فقط لازمه هر کادفریم این است که یک TcpApplicationID منحصر به فرد (نسبت به یک کادفریم سرور) داشته باشند. مقدار TCpAppID در فایل INI در مقابل strTcpApplicationID ست می شود.

در ادامه با ذکر یک مثال نحوه ی تنظیم دو کادفریم، یکی به عنوان سرور و دیگری به عنوان کلاینت توضیح داده خواهد شد:

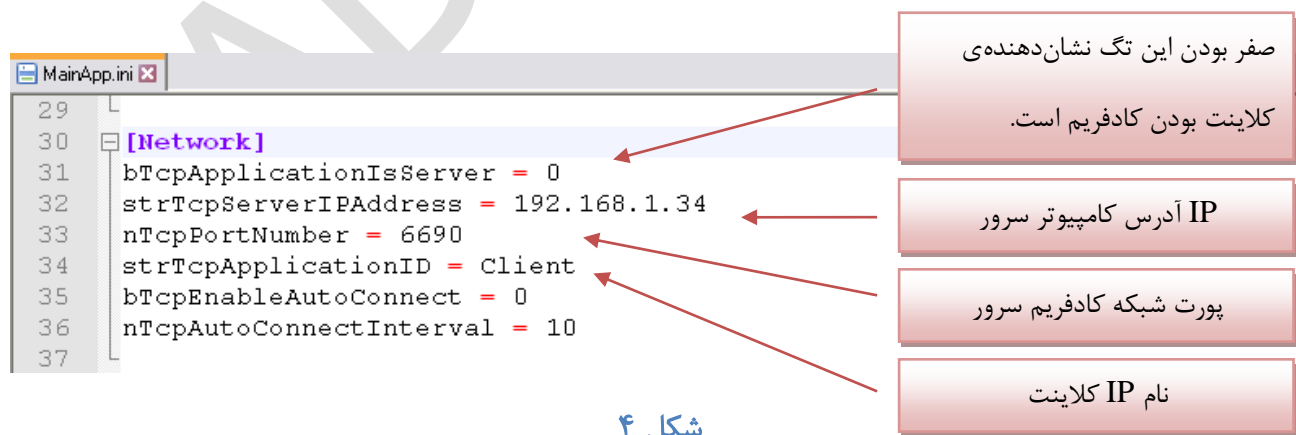
در ابتدا در کادفریمی که به عنوان سرور در نظر گرفته شده است، در فولدر Ini، فایل MainApp در قسمت Network مشابه توضیحات یادداشت شده در

شکل ۳ موارد زیر تغییر داده می شود.



شکل ۳

حال باید تنظیمات شکل ۴ در فایل MainApp کادفریم کلاینت انجام شود:



شکل ۴

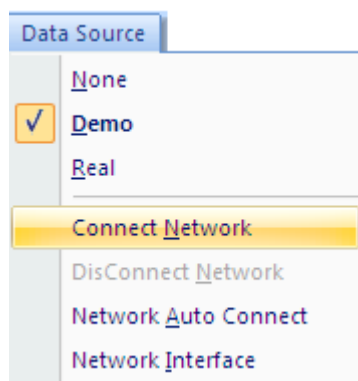
مرجع کار با برنامه مانیتورینگ کادفریم

پس از تنظیم IP آدرس ها، در مرحله بعد طبق توضیحات آمده در صفحه ی ۱۹ در فایل DCT کادفریم کلاینت باید مسیر NetworkAccessPath به شکل زیر تعریف شود.

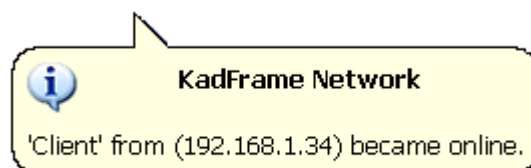
```
DCT-950419-Test.dct
1 //-----
2 //01-Name,02-DataType,03-DefaultData,04-HWType,05-IOType,06-HwAccessPath,07-NetworkAccessPath,08-DataStackSize
3 //-----
4 // 'UR' in OPC means: dwUpdateRate in milliseconds
5 // 'DB' in OPC means: fDeadband percent deadband
6 //-----
7
8 // Group:
9 fTest , FLOAT , NULL , NULL , NULL, NULL , 07-App:%Server%-Tag:%Same%-TransRate:5, 08-NULL
10
```

شکل ۵

در مرحله آخر با باز کردن نرم افزار کادفریم از منوی Data Source ارتباط شبکه را برقرار کرد:



زمانی که کادفریم کلاینت نیز به شبکه متصل می شود، بالن زیر در صفحه کادفریم سرور ظاهر می شود:



۵ آشنایی با عملکرد انجین دیتاسنتر کادفریم

۵.۱ نحوه آپدیت تگ های سریال توسط انجین دیتاسنتر

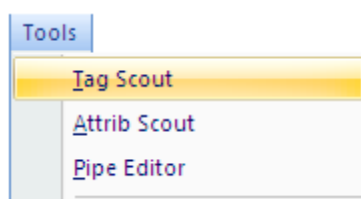
انجین دیتاسنتر، هر یک از تگ های سریال را به صورت جداگانه و یک به یک Pool می کند. پروسه آپدیت هر تگ هم شامل دو مرحله است. ابتدا کامندی از سمت دیتاسنتر به دیوایس آدرس دهی شده ارسال می شود. سپس دیتاسنتر منتظر پاسخ از سمت دیوایس می ماند. در صورتی که پاسخ با محتویات درست (Correct_CRC) از سمت دیوایس دریافت شود، مقدار و TimeStamp تگ آپدیت می شود. اگر پاسخی در مدت زمان (Time out Band) دریافت نشود، انجین دیتاسنتر سراغ تگ بعدی می رود. اگر فاصله زمانی بین آخرین آپدیت موفق یک تگ سریال تا نوبت آپدیت بعدی همین تگ سریال، کمتر از UR باشد، انجین دیتاسنتر آن تگ را آپدیت نمی کند و سراغ تگ بعدی می رود. اگر تگی نیاز به Write داشته باشد پس از اتمام Read تگ جاری آن تگ بلافاصله write میشود. اگر در آن واحد چندین تگ نیاز به Write داشته باشند، توالی یک Read و سپس یک write تکرار میشود. به عبارت دیگر همه تگهای نیازمند write به یک جا در دیوایسها نوشته نمیشود.

با توجه به نحوه آپدیت تگهای سریال در لیست DCT می توان استنباط کرد که هر چه تعداد تگهای سریال در لیست DCT زیاد باشد، فاصله بین دو آپدیت یک تگ بیشتر می شود. یعنی آپدیت یک تگ کندتر می شود. پس به عبارتی دیگر حداقل تاخیر بین دو آپدیت یک تگ و یا همان حداکثر سرعت آپدیت یک تگ، همان UR (Update Rate) می باشد. حداکثر تاخیر بین دو آپدیت یک تگ سریال در صورتی که اشکال هاردوری در ارتباط با دیوایس مورد نظر نباشد، مجموع TOB همه تگ های موجود در لیست DCT (در صورتی که از UR خود تگ بزرگتر باشد) می باشد.

مرجع کار با برنامه مانیتورینگ کادفریم

دقت در این نکته که TOB باید از UR کوچکتر باشد، حائز اهمیت است. فرض کنید یکی از تگها که دارای اهمیت کمی است قرار است هر 10000 میلی ثانیه (ده ثانیه) آپدیت شود. یک انتخاب معقولانه برای TOB این تگ می تواند 350ms باشد تا اگر اشکالی در ارتباط با دیوایس مرتبط بوجود آمد، آپدیت سایر تگ ها کند نشود. در چند تجربه عملی حداکثر زمان پاسخ یک PAX به یک کامند مادباس به طور متوسط حدود 120ms است.

۵.۲ آشنایی با KadFrame Tag Scout



لیست تمام تگ های تعریف شده در فایل DCT در هنگام اجرای کادفریم در پنجره Tag Scout نمایش داده می شود. این پنجره در کادفریم از طریق منوی Tools قابل دسترس است. پنجره Scout در ۸ ستون مشخصات خوانده شده تگ از فایل DCT را نشان می دهد. ستون "Tag Value" مقدار دیتای جاری تگ را نشان می دهد. ستون "Modifier" علاوه بر نمایش مقدار دیتای جاری تگ، امکان تغییر تگ را نیز می دهد. در هنگام اعمال تغییر در مقدار تگ باید کلید Ctrl کیبرد پایین نگه داشته شود. این کار برای جلوگیری از تغییر ناخواسته اتوماتیک توسط خود Scout در مقدار تگ هاست!

مرجع کار با برنامه مانیتورینگ کادفریم

Tag (DataCenter) Scout of KadFrame ('BasicTest' 133 Tags) [User: Tech1 (3:47)]

☐ Suspend Data Obtainer Line 0 ☐ Highlight (0)

ID	Tag Name	Tag Value	Modifier	IO Type	Data Type	HW Type	HW Access Path	Network Access Path
0	eDataSource	1	1	NULL	Int32	NonHW	NULL	NULL
1	bIsDataSourceDemo	True	True	NULL	Bool	NonHW	NULL	NULL
2	bIsDataSourceReal	False	False	NULL	Bool	NonHW	NULL	NULL
3	strDataSource	Demo	Demo	NULL	String	NonHW	NULL	NULL
4	strActivePageName	Any Name	Any Name	NULL	String	NonHW	NULL	NULL
5	nActivePageID	1	1	NULL	Int32	NonHW	NULL	NULL
6	strLoggedInUser	Tech1	Tech1	NULL	String	NonHW	NULL	NULL
7	strLoggedInUserInfo	[User: Tech1 (3:48)]	[User: Tech1 (3:48)]	NULL	String	NonHW	NULL	NULL
8	bIsActiveShift	True	True	NULL	Bool	NonHW	NULL	NULL
9	nActiveShiftID	1	1	NULL	Int32	NonHW	NULL	NULL
10	strShiftOperatorName	A ... A	...	NULL	String	NonHW	NULL	NULL
11	strShiftStartTime	1395/03/24 - 18:33:47	1395/03/24 - 18:33:47	NULL	String	NonHW	NULL	NULL
12	strShiftEndTime			NULL	String	NonHW	NULL	NULL
13	bIsActiveRecipe	False	False	NULL	Bool	NonHW	NULL	NULL
14	nActiveRecipeID	-1	-1	NULL	Int32	NonHW	NULL	NULL
15	strActiveRecipeName	No Recipe Name!	No Recipe Name!	NULL	String	NonHW	NULL	NULL
16	strAppStartTime	1395/10/02 - 12:04:46	1395/10/02 - 12:04:46	NULL	String	NonHW	NULL	NULL
17	strCurDateShamsi	1395/10/02 - 13:00:59	1395/10/02 - 13:00:59	NULL	String	NonHW	NULL	NULL
18	strCurDateMiladi	12/22/16 13:00:59	12/22/16 13:00:59	NULL	String	NonHW	NULL	NULL

شکل ۶

سایر آپشنهای پنجره Scout عبارتند از:

Line: با نوشتن هر عددی در این ادیت باکس، برنامه به سطر مورد نظر در لیست می رود.

ادیت باکسی که در سمت راست است برای جستجوی یک تگ بر اساس قسمتی از یا تمام نام تگ است. پنجره Scout تعداد تگ هایی را که با عبارت جستجو شده مطابقت دارد در سمت چپ باکس روبروی کلمه Highlight(nn) می نویسد. پنجره Scout به صورت اتومات به خط اولین تگ Match شده می رود.

۶ معرفی اشکال گرافیکی کادفریم

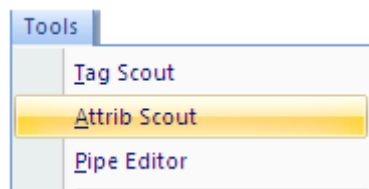
کادفریم به عنوان یک برنامه مانیتورینگ از ۲۱ ابجکت به عنوان اشیا گرافیکی پشتیبانی می کند. که این اشیا

عبارتند از :

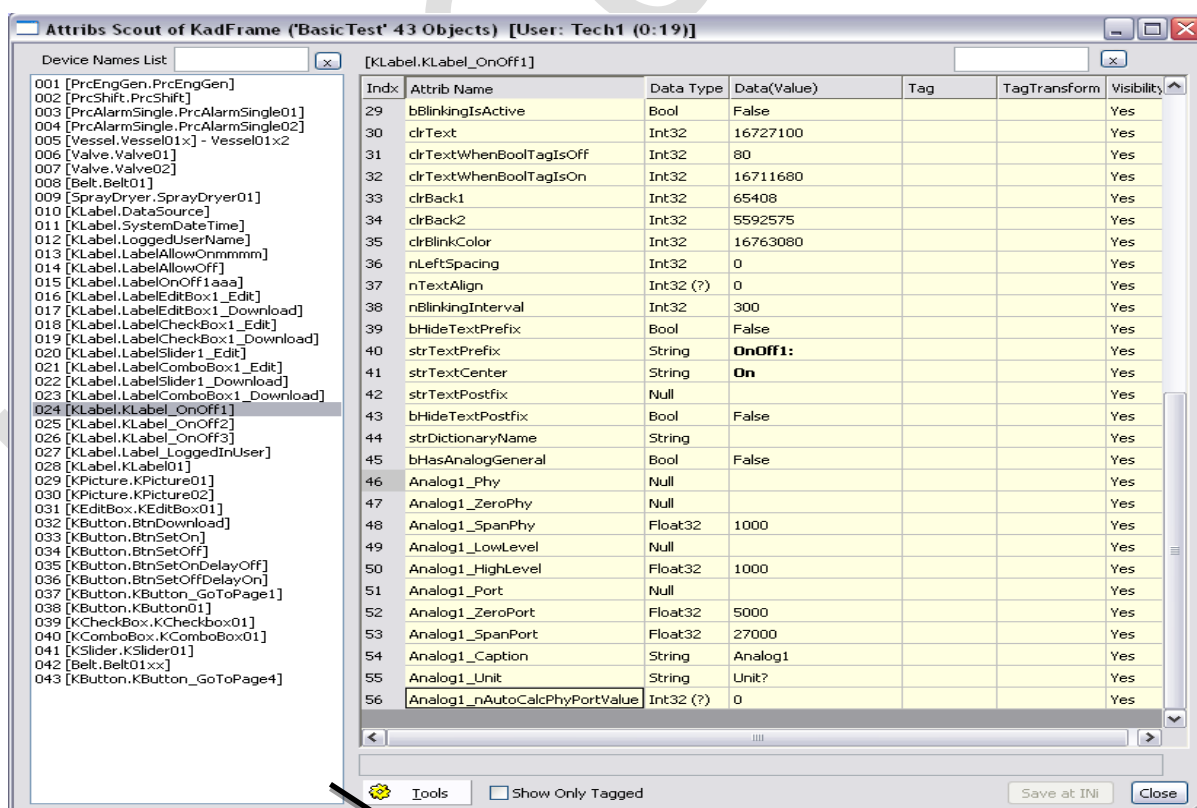
مرجع کار با برنامه مانیتورینگ کادفریم

Belt, CentrifugalPump, DBFValve, DosePump, GearPump, HeatZone, Helix, KButton, KCheckBox, KComboBox, KEditBox, KLabel, KPicture, KSlider, PrcAlarmSingle, RotaryMixer, RotaryWheel, SprayDryer, Valve, Vessel, VFeeder

برای اضافه کردن هر کدام از این اشیا به صفحه اصلی کادفریم باید از منوی Tools گزینه Attrib Scout را انتخاب کرد:



از پنجره باز شده مطابق شکل ۷ گزینه Tools را انتخاب کرده و سپس از لیست باز شده Add New Control را انتخاب تا پنجره ای مطابق شکل ۸ باز شود.



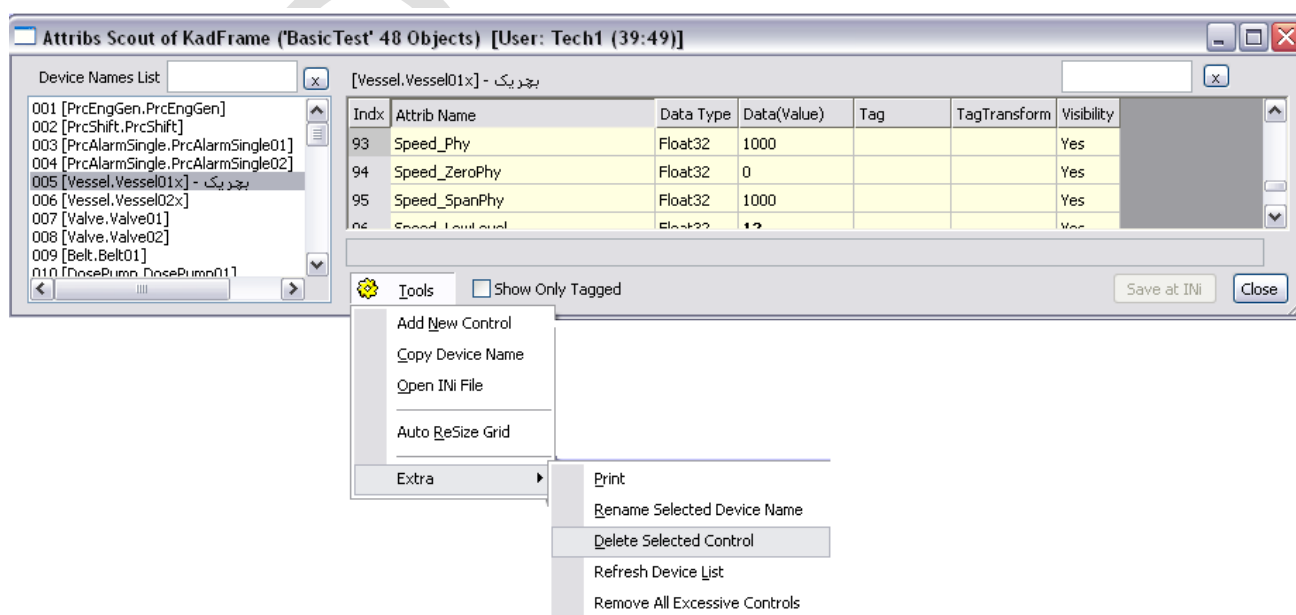
شکل ۷



شکل ۸

سپس از لیست باز شونده در زیر Control Type ID می توان نام ابجکت مورد نظر خود را انتخاب کرده و در قسمت New Device Name(Unique) نام مورد نظر طراح را برای آن شی را وارد کرد. در نهایت گزینه ی Add را زده تا کنترل گرافیکی مورد نظر به صفحه کادفریم اضافه شود و در نهایت این پنجره را می بندیم.

برای حذف هریک از کنترلهای گرافیکی باید بر روی آبجکت مورد نظر راست کلیک را نگهداشت و دبل کلیک چپ میکنیم و مطابق شکل ۹ از قسمت Tools، Extra کنترل انتخاب شده را حذف می کنیم.



شکل ۹

۷ نحوه تغییر سایز و جابجایی کنترل های گرافیکی در کادفریم

در ورژن جدید کادفریم امکان تغییر سایز و جابجایی کنترل هم از طریق ویرایش دستی INI و هم در Runtime از طریق گرافیک کادفریم وجود دارد (از ورژن 900220 به بعد کادفریم).

۷.۱ ویرایش مشخصات گرافیکی کنترل در INI به صورت دستی

پارامترهای گرافیکی قابل تغییر در INI برای هر کنترل گرافیکی در زیر سکشن هر کنترل لیست شده است که عبارتند از:

nDrawPosX.Data مکان کنترل در محور افقی با واحد پیکسل

nDrawPosY.Data مکان کنترل در محور عمودی با واحد پیکسل

nDrawSizeX.Data سایز افقی کنترل در واحد پیکسل

nDrawSizeY.Data سایز عمودی کنترل در واحد پیکسل

nDrawGraphicType.Data عددی است معمولاً یک رقمی که از صفر شروع می شود و تایپ کلی کنترل را مشخص می کند.

strDrawOrientationHex.Data عددی است در مبنای Hex که با 0x شروع می شود. جزئیات آن در ادامه شرح داده خواهد شد.

strDrawStyleHex.Data عددی است در مبنای Hex که با 0x شروع می شود. جزئیات آن در ادامه شرح داده خواهد شد.

۸ انتخاب نوع اکچواتور و سیگنالهای کنترلی آن از داخل کادفریم

در پنجره properties هر دیوایس یک تب به نام General Setting وجود دارد که شامل دو پارامتر اساسی در انتخاب نوع دیوایس از نظر سیگنالهای کنترلی و مود کاری آن میباشد :

۸,۱ Device Operation mode

مربوط به وضعیت هر اکچواتور میباشد که به صورت اتوماتیک توسط خود برنامه با توجه به وضعیت کاری که اپراتور تعیین میکند در این قسمت درج و نشان داده میشود. این حالتها مطابق شکل 10 شامل پنج حالت زیر میباشد:

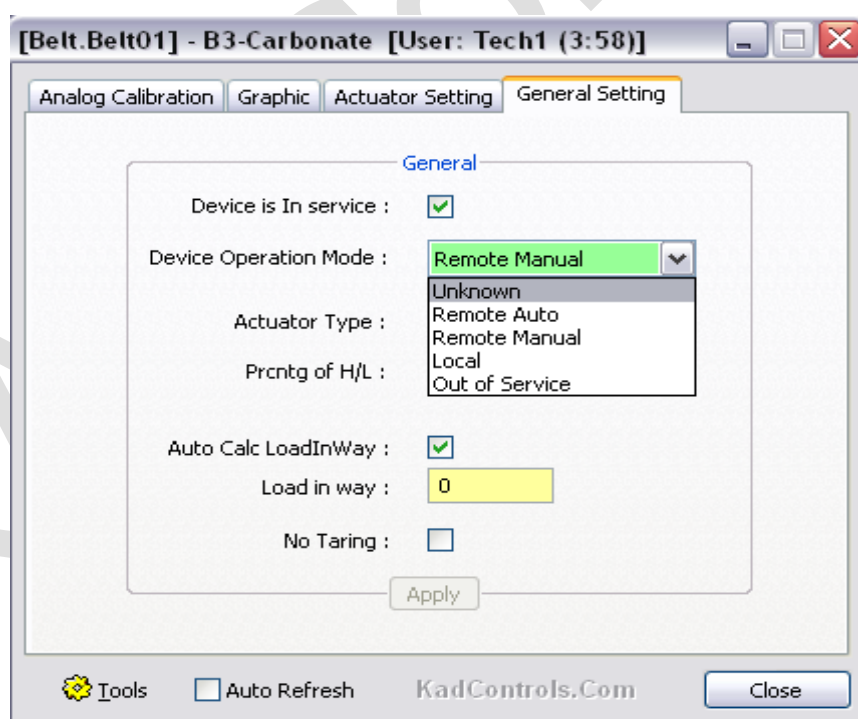
Unknown: یعنی وضعیت تجهیز کنترلی خارج از چهار حالت زیر میباشد.

Remote Auto: یعنی تجهیز کنترلی در مورد کنترل اتوماتیک است و توسط برنامه کنترل میشود.

Remote Manual: یعنی تجهیز کنترلی در مورد کنترل اتوماتیک است و توسط اپراتور به صورت دستی از کادفریم کنترل میشود.

Local: یعنی تجهیز کنترلی در مورد کنترل دستی از تابلوی محلی میباشد.

Out of service: برای سرویس کردن یا تعمیرات خارج از سرویس است.

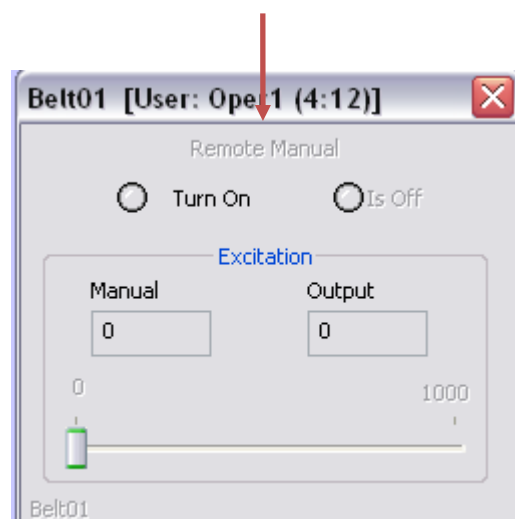


شکل 10

در ضمن در پنجره Action حالت جاری Device Operation Mode نمایش داده می شود:

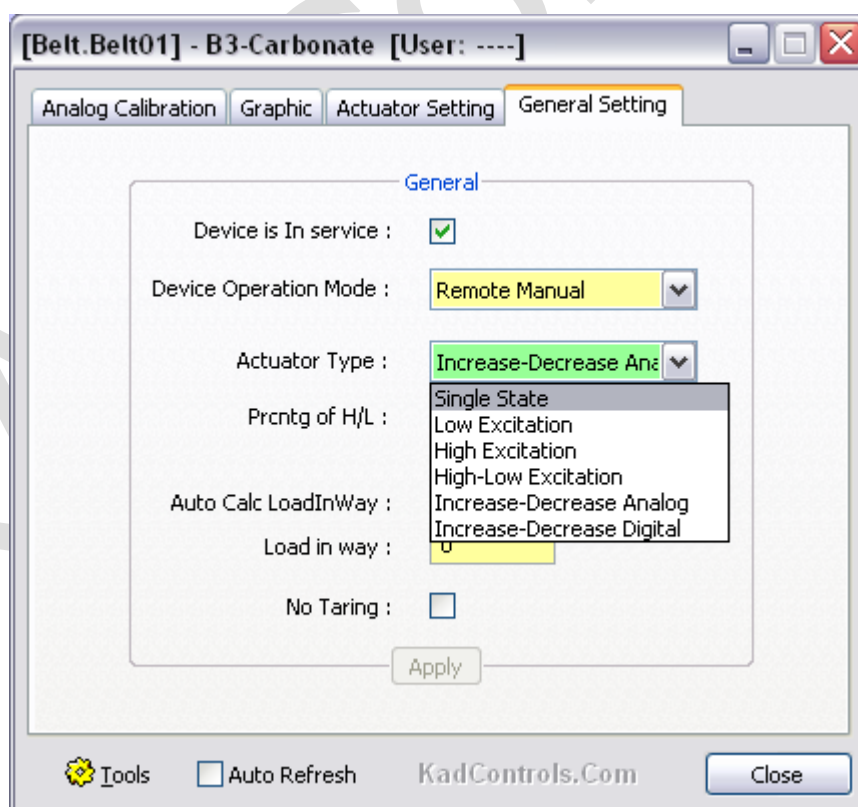
مرجع کار با برنامه مانیتورینگ کادفریم

Device Operation Mode



Actuator Type ۸,۲

مربوط به انتخاب نوع اکچواتور از نظر نحوه کنترل یک تجهیز کنترلی میشود. که شامل موارد زیر است (شکل ۱۱):



شکل ۱۱

Single State: مخصوص اکچواتورهای تک حالت است که شامل یک سیگنال on/off میباشد.

Low Excitation: مخصوص اکچواتورهای دو حالت است که تنها از سیگنال سرعت پایین آن استفاده میشود.

High Excitation: مخصوص اکچواتورهای دو حالت است که تنها از سیگنال سرعت بالای آن استفاده میشود.

High-Low Excitation: مخصوص اکچواتورهای دو حالت است که از هر دو سیگنال سرعت بالا و پایین آن استفاده میشود.

Increase-Decrease Analog: مخصوص اکچواتورهای سرعت متغیر است که کنترل سرعت آنها به صورت آنالوگ است.

Increase-Decrease Digital: مخصوص اکچواتورهای سرعت متغیر است که کنترل سرعت آنها به صورت دیجیتال (افزایش و کاهش) انجام میشود.

پنجره Action مربوط به هر تجهیز کنترلی با توجه به نوع اکچواتور تغییر می کند .

۸,۳ Prcntg Of H/L

زمانی که نوع اکچواتور High-Low Excitation (دوسرعت) انتخاب شود، این ادیت باکس فعال می شود. به طور مثال اگر این مقدار ۸۰ باشد، ۸۰٪ ست پوینت را با سرعت تند و ۲۰٪ باقی مانده را با سرعت پایین حرکت می کند.

۸,۴ Auto Calc LoadInWay

بار در راه یا Load in Way یکی از مشخصه های سیستم های توزین بچ می باشد، به این معنی که از زمان قطع کردن فعالیت اکچواتور تا زمان تثبیت وزنی که باید کنترل شود، یک تغییرات وزن وجود دارد که باید در محاسبات کنترلی منظور گردد و چون این مقدار تغییرات ثابت نیست و ممکن است در طول شبانه روز متغیر باشد، در واقع

یک تگ PLC تعریف میشود که اگر True باشد، LoadInWay را به صورت اتومات محاسبه می کند. در غیر این صورت اپراتور میتواند آن را تنظیم کند.

۸.۵ No Taring

منظور از Tare یا پاره سنگ کردن این است که وزن اولیه ظرف توزین خنثی گردد؛ اگر این چک باکس تیک خورده باشد، وزن شروع را در برنامه PLC صفر در نظر می گیرد. اگر این چک باکس تیک نخورده باشد، وزن شروع را در برنامه PLC به عنوان بخشی از ماده مورد توزین در نظر می گیرد.

۸.۶ ویرایش مشخصات گرافیکی کنترل در AttribScout

پارامترهای گرافیکی از سه روش قابل تغییر هستند، اول به صورت متنی در فایل INI برای هر کنترل گرافیکی که در بخش قبلی لیست شده است، دوم از صفحه Property Pages که با راست کلیک یا دبل کلیک روی هر کنترل گرافیکی قابل دسترسی است و سوم از طریق جدول AttribScout که دارای کلیه اتریب ها یا پارامتر های گرافیکی همه کنترل ها می باشد؛ این جدول به دو روش قابل دسترسی است، هم از منوی Tools کادفریم و هم اگر بر روی هر کنترل کلیک راست را گرفته و دو بار کلیک چپ زده شود، صفحه AttribScout ظاهر می شود. برای جزئیات بیشتر به بخش AttribScout مراجعه شود.

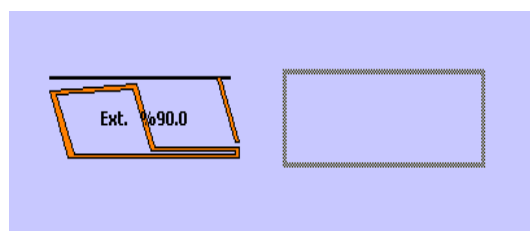
۸.۷ نحوه جابجا کردن کنترل گرافیکی در کادفریم با ماوس

برای جابجا کردن کنترل گرافیکی، کلید کنترل را نگه داشته و سپس کنترل مد نظر را با نگه داشتن کلیک راست جابجا کنید.

به محض اینکه کنترل شروع به جابجایی کرد، می توان همه کلیدها (کلید کنترل + کلیک راست) را رها کرده و به راحتی با حرکت ماوس کنترل را جابجا کرد.

مرجع کار با برنامه مانیتورینگ کادفریم

پس از رسیدن کنترل به مقصد، با کلیک چپ کنترل را در جای جدید تثبیت کنید. در صورت نیاز به انصراف، با کلیک راست جابجایی کنترل کنسل شده و به جای اول برمیگردد. برای دقت بالاتر در هنگام چسباندن کنترل، میتوان کلید Alt را نگه داشت.



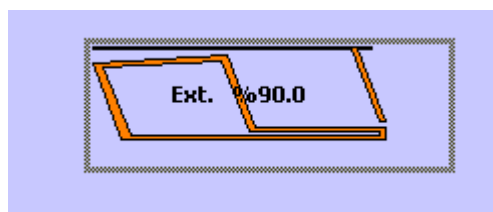
نکته: کلیک راست برای جابجایی یا تغییر سایز در داخل رجیون کنترل کار می کند. اگر در جایی از کنترل دابل کلیک کرده و پنجره Property باز نشود، آن ناحیه خارج از رجیون کنترل است. در انتها موقع بستن کادفریم، کادفریم با پرسیدن و تایید حق دسترسی، تغییرات جابجایی را ذخیره خواهد کرد. در صورت عدم تایید حق دسترسی همه تغییرات از بین خواهد رفت!

۸,۸ نحوه تغییر سایز کنترل گرافیکی در کادفریم با ماوس

برای تغییر سایز کنترل گرافیکی، کلید کنترل + کلید شیفت را نگه داشته و سپس کنترل مد نظر را با نگه داشتن کلیک راست تغییر سایز دهید.

به محض اینکه کنترل شروع به تغییر سایز کرد، می توان همه کلیدها (کلید کنترل + کلید شیفت + کلیک راست) را رها کرده و به راحتی با حرکت ماوس کنترل را ریسایز کرد.

پس از رسیدن کنترل به سایز جدید، با کلیک چپ کنترل را در سایز جدید تثبیت کنید. در صورت نیاز به انصراف، با کلیک راست تغییر سایز کنترل کنسل شده و به حالت اول برمیگردد.



مرجع کار با برنامه مانیتورینگ کادفریم

نکته: کلیک راست برای جابجایی یا تغییر سایز در داخل ناحیه کنترل کار می کند. اگر در جایی از کنترل دابل کلیک کرده و پنجره پروپرتی باز نشود، آن ناحیه خارج از ناحیه کنترل است.

در انتها موقع بستن کادفریم، کادفریم با پرسیدن و تایید حق دسترسی، تغییرات جابجایی را ذخیره خواهد کرد. در صورت عدم تایید حق دسترسی همه تغییرات از بین خواهد رفت!

۸,۹ خاصیت Snap در حین تغییر سایز یا جابجایی کنترل گرافیکی

در هنگام تغییر سایز یا جابجایی کنترل گرافیکی، عدد مکان جدید و عدد سایز جدید کنترل، به ضربی از عدد مشخص شده در INi رند می شود. این کار حالتی شبیه Snap شدن کنترل ها در یک جدول مجازی ترسیم شده در View اصلی کادفریم را تداعی می کند تا کار جابجایی و تغییر سایز کنترل با ماوس راحت تر باشد.

دو عدد جداگانه یکی به نام nSnapStepOfCtrlLocation برای مقدار Snap مکان کنترل و دیگری به نام nSnapStepOfCtrlSize برای مقدار Snap سایز کنترل در فایل INi اصلی کادفریم در نظر گرفته شده است. به طور مثال با مشخص شدن مقدار ۵ در Ini دیوایس کنترلی در صفحه کادفریم با مختصات پنج تا جا به جا می شود.

[Graphics]

nSnapStepOfCtrlLocation=5

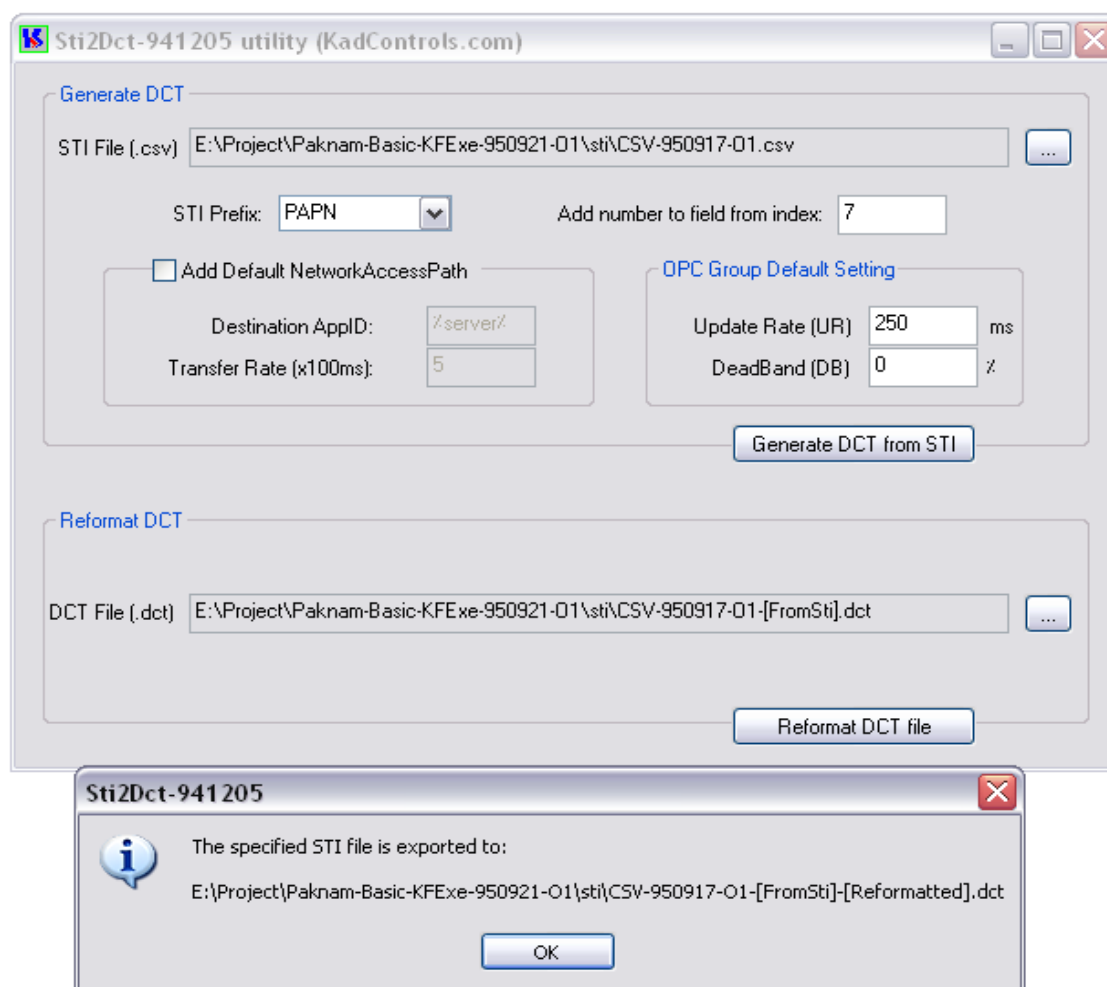
nSnapStepOfCtrlSize=5

روش غیرفعال کردن موقت خاصیت Snap در ویرایش کنترل: اگر در هنگام تثبیت سایز یا مکان جدید کنترل (که با کلیک چپ) انجام می شود، کلید Alt نگه داشته شده باشد، خاصیت Snap موقتا غیر فعال خواهد شد.

۹ ساخت اتومات فایل DCT برای تگ های OPC

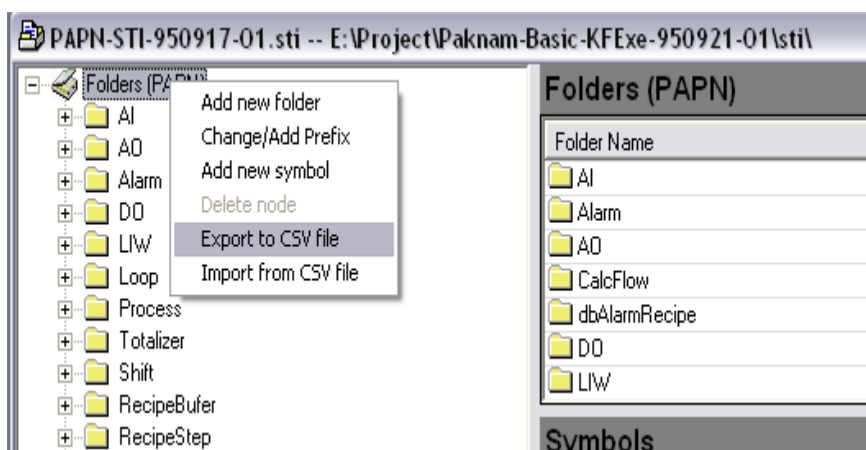
اگر در پروژه از تگ های OPC استفاده شده باشد، می توان توسط برنامه Sti2Dct-941205.exe (یا هر ورژن جدیدتر آن)، فایل DCT را از روی فایل STI ساخت. البته فایل DCT ساخته شده از روی STI، همیشه قابل ادیت و آپدیت می باشد. مثلا ما می توانیم، بعدا تگ های پورت سریال را به صورت دستی در DCT تایپ و اضافه کنیم.

برنامه Sti2Dct طبق شکل ۱۲، دارای دو بخش است.



شکل ۱۲

بخش بالایی کار اصلی برنامه یعنی تبدیل STI به DCT را انجام می دهد. البته این برنامه فقط فرمت اکسپورت شده STI به CSV را به عنوان ورودی قبول می کند. برای اکسپورت کردن فایل STI به CSV طبق شکل ۱۳ روی روت اصلی کلیک راست را می زنیم و از منوی پاپ شده کار Export to CSV را انتخاب می کنیم. برنامه در ادامه نام و محل ذخیره سازی فایل CSV را می پرسد.



شکل ۱۳

بعد از آماده شدن فایل CSV، به سراغ برنامه Sti2Dct (شکل ۱۲) می رویم. با زدن دکمه ... در سمت راست ادیت باکس (STI File (.CSV)، فایل ایجاد شده در مرحله قبل را انتخاب می کنیم. سپس “STI Prefix” را که باید دقیقاً مساوی Prefix داخل STI باشد، در ادیت باکس سفید تایپ می کنیم. Prefix در شکل ۱۳ در روت اصلی در داخل پرانتز قابل مشاهده و تغییر است. Prefix پروژه داخل عکس PAPN می باشد. پس از مشخص شدن دو پارامتر ورودی برای در بخش بالایی برنامه Sti2Dct، دکمه “Generate DCT from STI” را کلیک می کنیم. اگر نتیجه کار موفق باشد، توسط یک MessageBox که در پایین شکل ۱۲ مشخص است، محل فایل DCT ایجاد شده نشان داده می شود.

بخش پایینی برنامه Sti2Dct (شکل ۱۲)، برای Reformat کردن و یا به عبارتی دیگر مرتب و زیبا کردن محتویات فایل DCT به کار می رود. پس از Reformat شدن فایل DCT، خوانایی فایل DCT در ادیتور بهتر می شود. و اما فرمول Reformat کردن فایل DCT؛ محتویات فایل DCT به فرمت CSV(Comma Separated Value) می باشد. یعنی اینکه همه سطرهای DCT حاوی تعدادی ستون مشخص و یکسان است. برنامه Sti2Dct، سطرها را طوری Reformat می کند که ابتدای همه ستون ها یکسان باشد. بدیهی است که ابتدای هر ستون به اندازه بلندترین آیتم ستون قبلی، جلو خواهد رفت. نتیجه قسمتی از کار در شکل ۱۴ نشان داده شده است.

همانگونه که از شکل ۱۲ مشخص است در ورژن های جدید DCT ساخته شده Reformat نیز شده اند ولی اگر پس

از ساختن DCT تغییراتی در آن به وجود آمد می توان با استفاده از این گزینه آن را مرتب سازی کرد.

Before Reformat

```
// Group: PAPN.AI.WeightBelt0.
PAPN.AI.WeightBelt0.fHightLevelPhy , FLOAT , NULL , OPC , DUPLEX, UR250-DB0 , 07-NULL , 08-NULL
PAPN.AI.WeightBelt0.fLowLevelPhy , FLOAT , NULL , OPC , DUPLEX, NULL , 07-NULL , 08-NULL
PAPN.AI.WeightBelt0.fPhy , FLOAT , NULL , OPC , DUPLEX, NULL , 07-NULL , 08-NULL
PAPN.AI.WeightBelt0.fPhy_Avg_10Sec , FLOAT , NULL , OPC , DUPLEX, NULL , 07-NULL , 08-NULL
PAPN.AI.WeightBelt0.fPhy_Avg_1Sec , FLOAT , NULL , OPC , DUPLEX, NULL , 07-NULL , 08-NULL
PAPN.AI.WeightBelt0.fPortValue , FLOAT , NULL , OPC , DUPLEX, NULL , 07-NULL , 08-NULL
PAPN.AI.WeightBelt0.fSpanPhy , FLOAT , NULL , OPC , DUPLEX, NULL , 07-NULL , 08-NULL
PAPN.AI.WeightBelt0.fSpanPort , FLOAT , NULL , OPC , DUPLEX, NULL , 07-NULL , 08-NULL
PAPN.AI.WeightBelt0.fZeroPhy , FLOAT , NULL , OPC , DUPLEX, NULL , 07-NULL , 08-NULL
PAPN.AI.WeightBelt0.fZeroPort , FLOAT , NULL , OPC , DUPLEX, NULL , 07-NULL , 08-NULL
```

After Reformat

```
// Group: PAPN.AI.WeightBelt0.
PAPN.AI.WeightBelt0.fHightLevelPhy , FLOAT , NULL , OPC , DUPLEX, UR250-DB0 , 07-NULL , 08-NULL
PAPN.AI.WeightBelt0.fLowLevelPhy , FLOAT , NULL , OPC , DUPLEX, NULL , 07-NULL , 08-NULL
PAPN.AI.WeightBelt0.fPhy , FLOAT , NULL , OPC , DUPLEX, NULL , 07-NULL , 08-NULL
PAPN.AI.WeightBelt0.fPhy_Avg_10Sec , FLOAT , NULL , OPC , DUPLEX, NULL , 07-NULL , 08-NULL
PAPN.AI.WeightBelt0.fPhy_Avg_1Sec , FLOAT , NULL , OPC , DUPLEX, NULL , 07-NULL , 08-NULL
PAPN.AI.WeightBelt0.fPortValue , FLOAT , NULL , OPC , DUPLEX, NULL , 07-NULL , 08-NULL
PAPN.AI.WeightBelt0.fSpanPhy , FLOAT , NULL , OPC , DUPLEX, NULL , 07-NULL , 08-NULL
PAPN.AI.WeightBelt0.fSpanPort , FLOAT , NULL , OPC , DUPLEX, NULL , 07-NULL , 08-NULL
PAPN.AI.WeightBelt0.fZeroPhy , FLOAT , NULL , OPC , DUPLEX, NULL , 07-NULL , 08-NULL
PAPN.AI.WeightBelt0.fZeroPort , FLOAT , NULL , OPC , DUPLEX, NULL , 07-NULL , 08-NULL
```

شکل ۱۴

۱۰ آشنایی با فایل های KAC.sys

فایل KAC (Kad Access Control) حاوی اطلاعات کاربران سیستم حق دسترسی است.

کادفریم اطلاعات مربوط به کاربران (Users) را در فایلی به نام ProjectID_KAC.sys نگهداری می کند. فرمت فایل

KAC همان فرمت فایل INi است. پسوند این فایل از آن جهت sys. قرار داده شده است که نتوان به آسانی آن را

توسط ادیتور (مثل Notpad) باز کرد. پسوند sys. در اصل مبین فایل های اجرایی سیستم است.

هر یوزر در INi دارای یک Section منحصر به فرد است. فرمت section هر یوزر به شکل [User-ID] است.

قسمت ID از [User-ID] هر مقداری متشکل از اعداد و حروف می تواند باشد.

مثالها: [User-1], [User-2], [User-Khaas], [User-2x], ...

مرجع کار با برنامه مانیتورینگ کادفریم

هر یوزر دارای سه Key در سکشن خود است که عبارتست از: Name, PW, GRP

- کلید Name نام کاربر را مشخص می کند. این نام در صفحه لاگین نمایش می دهد.
- کلید PW حاوی پسورد کاربر است.
- کلید GRP نشان می دهد که جز چه گروه هایی است.

در کادفریم از قبل برای کاربران ۶ گروه مختلف تعریف شده است.

گروه (Group) یک عدد ۸ بیتی که هر بیت آن مبین یک گروه خاص است:

- بیت هشتم: **Kad-Admin** (نماینده شرکت کاد)
- بیت هفتم: **Plant-Admin** (مدیر سایت)
- بیت ششم: **Production** (مدیر تولید)
- بیت پنجم: **Technical** (مدیر فنی و ابزار دقیق)
- بیت چهارم: **Shift** (مدیر شیفت)
- بیت سوم: **Reserve**
- بیت دوم: **Reserve**
- بیت اول: **Operator** (اپراتور)

یوزر می تواند جز یک یا چند گروه باشند. فرمت مشخص کردن عضویت یک یوزر در گروه ها در فایل ini توسط یک عدد Hex به فرمت 0xNN است. در عدد مذکور بیت گروه هایی که یوزر در آن شرکت دارد باید یک شود. به عنوان مثال GRP=0x0C نشان دهنده این است که User1 جز دو گروه Production و Shift است.

مرجع کار با برنامه مانیتورینگ کادفریم

هر برنامه کادفریم نیاز دارد که حداقل یک یوزر که جزء Kad-Admin است، داشته باشد. در غیر اینصورت کادفریم اجرا نخواهد شد. از این قانون می توان به نوعی به عنوان License کادفریم اشاره کرد که اگر نباشد کادفریم اجرا نخواهد شد. نحوه ایجاد یوزر از طریق ادیت دستی فایل KAC.sys در انتها شرح داده خواهد شد.

۱۰,۱ رمزنگاری محتویات فایل KAC

مقادیر همه Values of Keys (یعنی مقادیر سمت راست علامت مساوی) در فایل KAC.sys رمزنگاری (Encrypt) شده است. کادفریم کار Encrypt/Decrypt را به صورت اتومات انجام می دهد.

[User-1]

Name=D63767615245363C3B37403F3A333C3838413B483F4A35343A3B35

PW=53303630373F303635313A39342D3632323B354239442F2E34352F

GRP=2C3136733240313736323B3A352E3733333C36433A45302F353630

۱۰,۲ فرمت کلی رمزنگاری اطلاعات توسط کادفریم

کادفریم فقط اطلاعات String را رمزنگاری می کند و به کوچکی و بزرگی حروف انگلیسی حساس است. در هنگام رمزنگاری، دو مقدار AppID و OptionalKey نیز با String اصلی میکس شده و سپس دیتا Encrypt می شود. لازمه Decrypt کردن دیتای رمز شده، داشتن دو مقدار AppID و OptionalKey است. در اینجا می توان دو مقدار AppID و OptionalKey را به نوعی به عنوان پسورد Encyption معرفی کرد. دو مقدار مذکور (پسوردها) می توانند خالی باشد.

فرمت AppID معمولاً (نه الزاماً) چهار حرفی است و نشان دهنده نوع پروژه و محل اجرای پروژه است. مثل PAPX, SDHP, SLMY, ...

مرجع کار با برنامه مانیتورینگ کادفریم

در فایل اطلاعات کاربران KAC.sys، کلمه OptionalKey همان User-ID است که در ابتدای این بخش به آن اشاره شد.

استفاده از کلید پسورد برای رمزنگاری این اطمینان را بوجود می آورد، که هر نوع دیتای با پسورد خاص خودش رمز شود. با این تدبیر نمی توان از اطلاعات رمز شده، به روش کپی از یک جا به جای دیگر استفاده کرد. مثلاً پسورد User-1 را نمی توان با کپی در پسورد User-2 قرار داد. چون کلید Decryption پسورد این دو یوزر متفاوت است.

۱۰,۳ ایجاد اولین یوزر Kad-Admin

هر برنامه کادفریم نیاز دارد که حداقل یک یوزر که جزء Kad-Admin است، داشته باشد. در غیر اینصورت کادفریم اجرا نخواهد شد. از این قانون می توان به نوعی به عنوان License کادفریم اشاره کرد که اگر نباشد کادفریم اجرا نخواهد شد.

در ابتدا Notepad را باز کرده و در آن یک Section به نام [User-1] می نویسیم. سپس فایل را به نام AppID_KAC.sys ذخیره می کنیم. سپس کلید به نامهای Name, PW, GRP در زیر سکشن جدید ایجاد می کنیم:

[User-1]

Name=

PW=

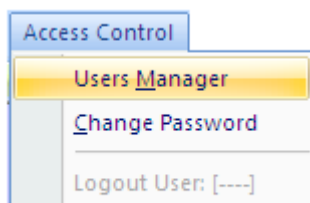
GRP=

توضیح اینکه یوزرنیم و یوزر آی دی نمیتواند یکی باشد. برای جزئیات تعریف KAD-Admin با شرکت کنترل های صنعتی کاد تماس بگیرید.

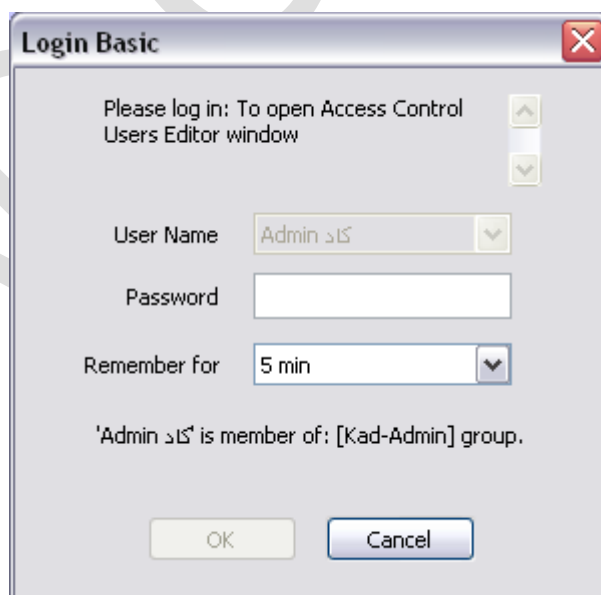
مرجع کار با برنامه مانیتورینگ کادفریم

۱۰,۴ مدیریت یوزرها توسط کادفریم (تکمیل شود)

برای جلوگیری از تغییر تنظیمات توسط افراد ناوارد و افزایش امنیت، کادفریم این قابلیت را دارد که برای قسمتهای مختلف خود سطح دسترسی تعریف کند، برای اختصاص سطح دسترسی به برنامه کلیدهای **Shift+Ctrl** رانگه داشته و از منوی **Access Control** روی گزینه **Users Manager** کلیک می کنیم.

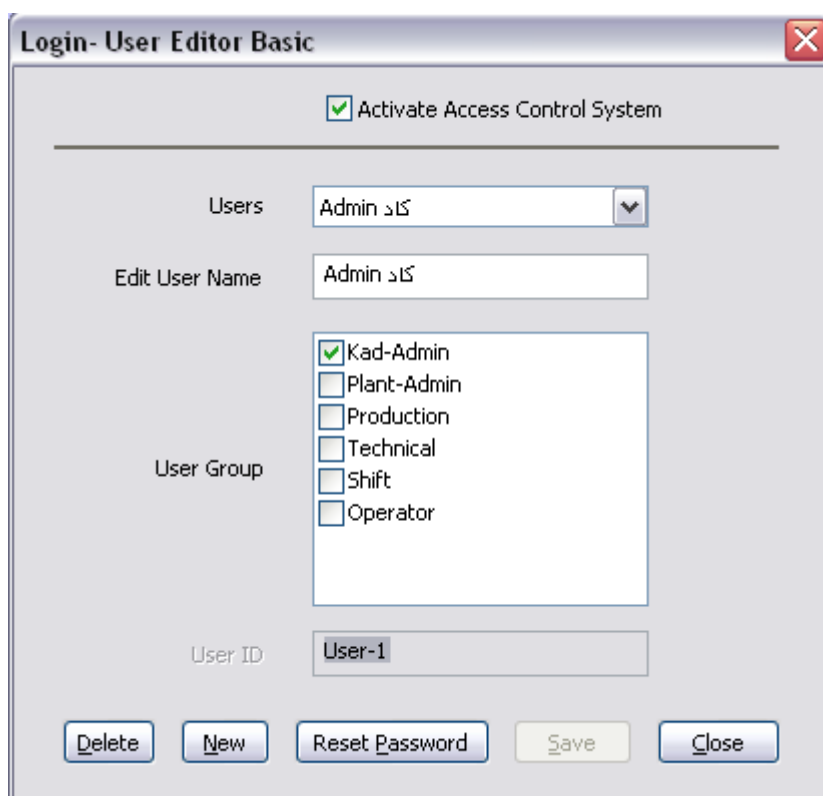


پنجره ای که باز می شود حاوی تنظیمات سطح دسترسی می باشد. در این پنجره در قسمت **User Name** اسامی یوزرهایی که دارای بالاترین سطح دسترسی هستند را نمایش میدهد (اگر برای اولین بار است که این پنجره را باز می کنید فقط یوزر **Kad-Admin** نمایش داده می شود) یوزر مورد نظر را انتخاب کرده و بازدن پسورد صحیح وارد صفحه ویرایش تنظیمات سطح دسترسی می شویم.



شکل ۱۵

در صفحه ای مطابق با شکل 16 زیر باز می شود:



شکل ۱۶

۱۰,۴,۱ قسمت‌های مختلف پنجره Login:

قسمت های مختلف این پنجره به شرح زیر میباشد:

- Active Access Control System: برای فعال و یا غیر فعال کردن کل تنظیمات امنیتی است.
- Users: در این قسمت میتوانیم یوزر مورد نظر خود را انتخاب کنیم.
- Edit User Name: در این بخش می توانیم نام مورد نظر خود را به یوزر ایجاد شده اختصاص دهیم، هم با فونت فارسی و هم با فونت انگلیسی
- User Group: در این پنجره سطح دسترسی مد نظر خود را به یوزر ایجاد شده میدهیم.
- Delete: برای پاک کردن یوزر ایجاد شده
- New: برای ایجاد یوزر جدید
- Reset Password: برای ریست کردن پسورد

مرجع کار با برنامه مانیتورینگ کادفریم

○ Save: برای ذخیره تنظیمات

○ Close: برای بستن پنجره

نکته) پس از ایجاد هر یوزر یک بار پسورد آن را ریست کنید کادفریم به طور پیش فرض پسورد ۱۱۱۱ را برای همه یوزرها در نظر میگیرد.

۱۱ مراحل ساخت اپلیکیشن

در شروع ساخت نرم افزار کاربردی مانیتورینگ کادفریم برای یک پروژه معلوم میبایست ابتدا این پروسس را به خوبی بشناسید. این شناخت شامل ادوات کنترلی و سیگنالهای کنترلی موجود در پروژه و همچنین پروسه کنترل مربوط به آن میشود. مراحل زیر شما را در این مسیر هدایت میکند:

۱۱,۱ مرحله اول : نقشه P&I

نقشه P&I یک پروژه یک نقشه فلودیگرام فنی است که در آن علاوه بر دیاگرام پروسس ، تجهیزات کنترلی و سیگنالهای کنترلی به کار رفته در این پروژه همچنین ابعاد و موقعیت قرار گرفتن دیوایس ها اعم از کنترلی و غیرکنترلی در طبقات مختلف کارخانه نمایش داده میشود لذا شما میبایست حتما پیش از هر کاری یک دید کلی کافی از شمای کارخانه و نحوه ی ترتیب اجرای پروسه کنترلی داشته باشید.

۱۱,۲ مرحله دوم : تهیه لیست تجهیزات (DeviceList)

در این مرحله لازم است که شما لیست تمام دیوایسها (تجهیزات) خود اعم از کنترلی و غیرکنترلی که تنها نقش نمایشی در صفحه اصلی کادفریم خواهند داشت میبایست لیست شود. استاندارد که برای نامگذاری نام دیوایس ها در پیش گرفته میشود در ادامه مسیر بسیار اهمیت پیدا میکند. زیرا این نامگذاری پایه نامگذاری سکشن های

مرجع کار با برنامه مانیتورینگ کادفریم

کنترلی و نام تگ ها و در نهایت سیگنالهای کنترلی میشود. در تعریف DeviceName میبایست موارد زیر در نظر گرفته شود:

نامگذاری بچرها (پروسه کنترل توزین) را با اسم کامل انجام دهیم و سپس بقیه ملحقات آن شامل مخازن ذخیره مواد اولیه، شیرهای ورودی و خروجی، همچنین پمپ ها و هلیس های ورودی و خروجی و... را با اختصار نویسی نام آن دیوایس نامگذاری کنید.

به طور کلی بایست به تنهایی گویای نام دیوایس و نوع اکچواتور (شارژ یا دیشارژ) باشد. منحصر به فرد باشد. نام تکراری پذیرفته نیست. اختصار کافی رعایت شود. زیرا Symbol Name ها در سیماتیک محدودیت مصرف ۲۴ کاراکتر دارند و در آنها یک سری کلمات مانند Is on/off و ... باید اضافه شود.

لذا بچرها و مخازن اصلی با نام کامل نامگذاری میشود و متعلقات مربوط به هر دیوایس اصلی با نام اختصار آن نامگذاری میشود.

۱۱,۳ مرحله سوم: تهیه لیست ورودی خروجی

شما از روی نقشه P&I میتوانید لیست دیوایس های کنترلی و همچنین سیگنال های کنترلی آنها را استخراج کنید و لیست نمایید. این لیست باید با مشاور فنی کارخانه چک شود و تایید گردد. ممکن است برخی سیگنال های کنترلی در حال حاضر قابل استفاده نباشد که با توافق میتوان آنها را در نظر گرفت و در مراحل تکامل پروژه پس از راه اندازی میتوان آن ها را در نظر گرفت. لیست ورودی و خروجی شما شامل تنها سیگنال های فیزیکیال میشود و متغیرهای داخلی برنامه مربوط به آلارم و پروسس و ... (تگ لیست کامل پروژه) را شامل نمیشود. معمولاً لیست ورودی خروجی یک فایل اکسل میباشد که شامل ستونهای زیر میباشد:

Device- KeyName	SignalDescriptio n-F	SignalDescription- En	SignalType e	Signal- Address	H/W Address
--------------------	-------------------------	--------------------------	-----------------	--------------------	----------------

Device-KeyName

در این ستون [DeviceType+DeviceName] مربوط به هر تجهیز نویسی میشود همه سیگنالهای مربوط به یک تجهیز یکسان دارای مقادیر یکسانی در این ستون هستند.

SignalDescription

در این ستون شرح سیگنال کنترلی مورد نظر نوشته میشود.

SignalType

در این ستون نوع سیگنال معلوم میشود که از نوع DI یا DO یا AI و یا در برخی موارد AO است.

Signal-Address

در این ستون آدرس سیگنالها نوشته میشود. این آدرس شامل نام DB و آدرس سیگنال مورد نظر است.

H/W Address

در این ستون آدرس های سخت افزاری مربوط به این سیگنال ها نوشته میشود. منظور همان شماره ترمینال ها است.

✓ پس از این مرحله تا رسیدن به مرحله ششم میبایست : نفر برنامه نویس PLC برای تک تک تجهیزات کنترلی بلاک های مناسب LOOP یا Analogue یا Extra تعریف کند و آدرس های مناسب در ستون **signal-Address** تکمیل نماید.

۱۱,۴ مرحله چهارم : تهیه فایل ini

از روی لیست تجهیزات (DeviceList) میبایست فایل ini تهیه شود. کادفریم در حال حاضر این قابلیت را دارد که میتوان هر تعداد کنترل که خواستید بدون کمک برنامه نویس در صفحه اصلی کادفریم اضافه کنید. دقت کنید که تجهیزات کنترلی برنامه هر کدام میبایست در گروه Ini مربوط به خود تعریف شود. اینکار با افزودن سکشن کنترل مورد نظر در ini امکان پذیر است. تمام مشخصات (اتریبهای) این سکشن در ابتدا مقادیر دیفالت خود که در کد تعریف شده است را دارد. در صورت تغییر در هر کدام از پنجره AttribScout داخل برنامه کادفریم این اتریب و مقدار جدید آن در زیر سکشن مربوطه به صورت اتومات نوشته میشود.

در زیر لیست ControlTypeName در 14 نوع که تاکنون وجود دارد لیست میشود:

جدول ۱

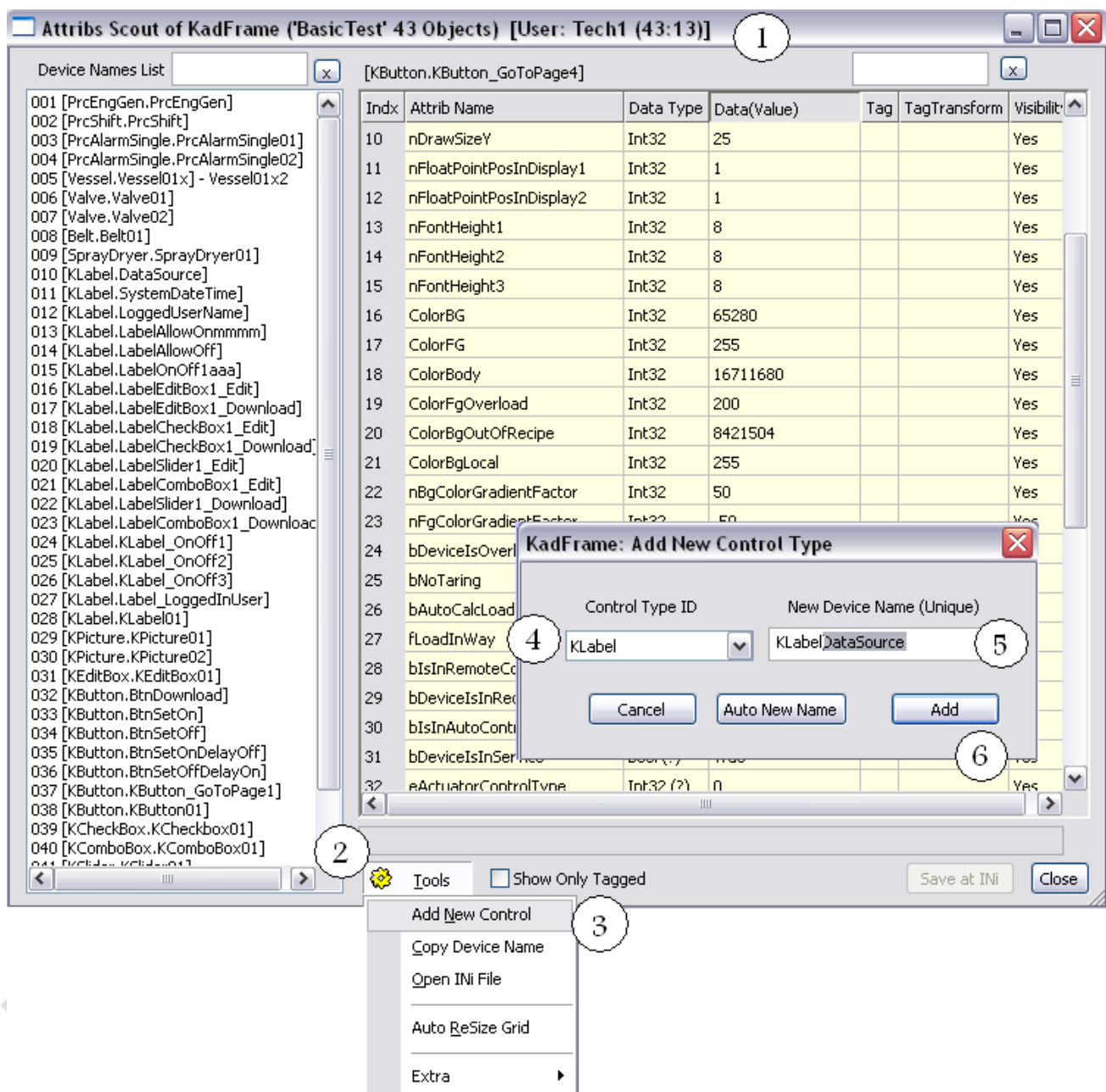
شماره	نام المان کنترل	توضیحات
۱	-Vessel.ini	Vessel.Batcher; Vessel.Mixer; Vessel.Crucher; Vessel.HoldingTank; Vessel.Tank; Vessel.Hopper
۲	-VFeeder.ini	فیدر
۳	-Belt.ini	نوار
۴	-DosePump.ini	دوز پمپ
۵	-Valve.ini	شیر

هلیس	-Helix.ini	۶
دنده ای	-GearPump.ini	۷
پمپ سانتریفوژ	-CentrifugalPump.ini	۸
روتاری میکسر	-RotaryMixer.ini	۹
کوره ی هوای گرم	-HeatZone.ini	۱۰
روتاری فیلتر	-RotaryWheel.ini	۱۱
اسپری درایر	-SprayDryer.ini	۱۲
شیر پروانه ای	-DBFValve.ini	۱۳
لیبل	-KLabel.ini	۱۴
کلید	-KButton.ini	15
چک باکس	-KCheckBox.ini	16
کمبوباکس	-KComboBox.ini	17
ادیت باکس	-KEditBox.ini	18
تصویر	-KPicture.ini	19
اسلایدر	-KSlider.ini	20
آلارم	-PrcAlarmSingle	21

البته علاوه بر روش دستی افزودن کنترل (یعنی ادیت Ini)، برنامه کادفریم نیز قابلیت راحت تر و مطمئن تر افزودن کنترل از طریق اتریب اسکات(۱) را طبق شکل ۱۷ ارائه می دهد. بعد از کلیک دکمه (2)Tools و سپس منوی Add New Control(3) پنجره Add New Control Type باز می شود. در پنجره مذکور ابتدا (4)Control Type ID را انتخاب می کنیم. برنامه به طور اتوماتیک یک نام یونیک به عنوان DeviceName در (5)New Device Name قرار می دهد. پیشنهاد می شود در مرحله ۵ برای DeviceName یک نام متناسب از DeviceList انتخاب شود. مثلا اگر

مرجع کار با برنامه مانیتورینگ کادفریم

کنترل KLabel ایجاد می‌کنید به جای نام هایی مثل KLabel01, KLabel02 از نام هایی مثل KLabelOperatorName استفاده شود.



شکل ۱۷

۱۱,۵ مرحله پنجم : تنظیمات چیدمان گرافیکی برنامه

پس از اینکه تجهیزات پروژه خود را در گروههای مناسب Ini در برنامه تعریف کردید. حال میبایست کادفریم را اجرا کنید. اگر در حین اجرای کادفریم دکمه Ctrl را نگه دارید. درستی فایل‌های Ini در حین اجرا چک میشود و پیغامهایی ظاهر میشود که میبایست به آنها دقت شود. همانطور که می دانیم زیر سکشن هر کنترل در Ini اطلاعات اتریب های آن کنترل نوشته می شود. لذا اگر خطی در Ini موجود باشد که متعلق به کنترل نباشد، کادفریم پیشنهاد پاک کردن آن را می دهد. یکی از دلایل ایجاد اشکال، می تواند تغییر نام اتریب ها در کادفریم باشد. رفرنس تعداد و نام آخرین اتریب های یک کنترل، اتریب اسکات می باشد.

پس از باز شدن برنامه تجهیزات با تنظیمات دیفالت روی صفحه نمایش کادفریم ظاهر میشود که میبایست در چیدمان درست قرار گیرد. علت اینکار این است که در تستهای برنامه محل تجهیزات به شما کمک میکند که از خطاهای احتمالی جلوگیری شود. شما در این مرحله نیازی به پایپینگ و تنظیمات دقیق گرافیکی ندارید.

۱۱,۶ مرحله ششم : تهیه فایل Sti

برای برقراری ارتباط کادفریم با PLC لازم است که یک دیتابیس بین آنها وجود داشته باشد که هر کدام بتوانند مقدار متغیرها را تغییر دهند و یا از آن بخوانند (Read, Write) این دیتابیس Sti نامیده میشود. فرض بر این است که لیست تمام این Address ها قبلا توسط PLC Programmer تعیین و تعریف شده است. در فایل Sti این متغیرها در چندین گروه قرار میگیرند که شامل هر کدام از گروه های تجهیزات و سیگنال های داخلی برنامه مانند Shift و Recipe و Process است. نمونه آن در شکل ۱۸ نشان داده شده است. روی هر مسیر که کلیک کنید لیست آدرس های متغیرهای مربوطه به صورت زیر لیست میشود:

مرجع کار با برنامه مانیتورینگ کادفریم

PAPN-STI-950917-01.sti -- E:\Project\Paknam-Basic-KFExe-950921-01\sti

Folders (PAPN)

- AI
- AQ
- Alarm
- DQ
- LJW
- Loop
- Process
- Totalizer
- Shift
- RecipeBufcr
- RecipeStep
- dbAlarmRecipe
- CalcFlow
- LookupTable
- SIDvFeed1
- SIDvBelt0

Index	Symbol Name	Item ID	Runtime Name	Data Type	Access Rights	EU Lo	EU Hi
1	bIsOnExtruderA	Process.bIsOnExtru	S7:[PAPN-PLC]DB9	VT_BOOL(0x000B)	Read/Write	0	0
2	bIsOnExtruderB	Process.bIsOnExtru	S7:[PAPN-PLC]DB9	VT_BOOL(0x000B)	Read/Write	0	0
3	bHwTrigStartProce	Process.bHwTrigSta	S7:[PAPN-PLC]DB9	VT_BOOL(0x000B)	Read/Write	0	0
4	bHwTrigStopProce	Process.bHwTrigStc	S7:[PAPN-PLC]DB9	VT_BOOL(0x000B)	Read/Write	0	0
5	bRestartProcess	Process.bRestartPr	S7:[PAPN-PLC]DB9	VT_BOOL(0x000B)	Read/Write	0	0
6	bStartProcess	Process.bStartProce	S7:[PAPN-PLC]DB9	VT_BOOL(0x000B)	Read/Write	0	0
7	bProcessRunning	Process.bProcessRi	S7:[PAPN-PLC]DB9	VT_BOOL(0x000B)	Read/Write	0	0
8	bIsInStartupPeriod	Process.bIsInStartu	S7:[PAPN-PLC]DB9	VT_BOOL(0x000B)	Read/Write	0	0
9	bSaveRecipe	Process.bSaveReci	S7:[PAPN-PLC]DB9	VT_BOOL(0x000B)	Read/Write	0	0
10	bStartNewShift	Process.bStartNewS	S7:[PAPN-PLC]DB9	VT_BOOL(0x000B)	Read/Write	0	0
11	bAudioAlarm	Process.bAudioAlar	S7:[PAPN-PLC]DB9	VT_BOOL(0x000B)	Read/Write	0	0
12	bVisualAlarm	Process.bVisualAla	S7:[PAPN-PLC]DB9	VT_BOOL(0x000B)	Read/Write	0	0
13	bInterlockOfStartPrc	Process.bInterlockO	S7:[PAPN-PLC]DB9	VT_BOOL(0x000B)	Read/Write	0	0
14	bTurnOnMixerBelt0	Process.bTurnOnMi	S7:[PAPN-PLC]DB9	VT_BOOL(0x000B)	Read/Write	0	0
15	bRecipesActive	Process.bRecipesA	S7:[PAPN-PLC]DB9	VT_BOOL(0x000B)	Read/Write	0	0
16	bWatchDogCounter	Process.bWatchDo	S7:[PAPN-PLC]DB9	VT_BOOL(0x000B)	Read/Write	0	0
17	eDetergentCtrlMode	Process.eDetergent	S7:[PAPN-PLC]DB9	VT_I2(0x0002)	Read/Write	0	0
18	nStartupAlarmServic	Process.nStartupAle	S7:[PAPN-PLC]DB9	VT_I2(0x0002)	Read/Write	0	0
19	ivatchDogCounterf	Process.ivatchDog	S7:[PAPN-PLC]DB9	VT_I2(0x0002)	Read/Write	0	0
20	iMaxGapTimeInOnlri	Process.iMaxGapTir	S7:[PAPN-PLC]DB9	VT_I2(0x0002)	Read/Write	0	0
21	nShiftTotalTimeSec	Process.nShiftTotal	S7:[PAPN-PLC]DB9	VT_I2(0x0002)	Read/Write	0	0
22	nShiftUsefulTimeSev	Process.nShiftUsefu	S7:[PAPN-PLC]DB9	VT_I2(0x0002)	Read/Write	0	0
23	nProcStartTimeSec	Process.nProcStartTi	S7:[PAPN-PLC]DB9	VT_I4(0x0003)	Read/Write	0	0
24	nProcStopTimeSec	Process.nProcStopTi	S7:[PAPN-PLC]DB9	VT_I4(0x0003)	Read/Write	0	0
25	rShiftTotalProductv	Process.rShiftTotal	S7:[PAPN-PI]DR9	VT_R4(0x0004)	Read/Write	0	0

Symbols

شكل ١٨

برای تهیه فایل Sti شما میتوانید از یک فایل Sti در یک پروژه مشابه کپی بگیرید زیرا این فایل شامل متغیرهای پروسس میباشد و کافیست که شما اکچوآتورهای مخصوص این پروژه را اضافه کنید و آدرسهای آنها را ویرایش کنید. برای برخی ویرایش ها که نیاز به Replace یک مقدار یکسان دارید میتوانید روی دایرکتوری اصلی Sti کلیک راست نموده و گزینه Export را انتخاب کنید. سپس یک مسیر و نام فایل به آن معرفی میکنید و میتوانید فایل Sti خود را با فرمت CSV ذخیره کنید. این فایل با ادیتورهای ویندوز و همچنین VisualC قابل ویرایش است و در نهایت میتواند مجددا در فایل Sti آن را import نمود. برای import کردن ابتدا یک Sti جدید ایجاد کنید. سپس فایل csv را در آن import کنید. سپس روی شاخه اصلی کلیک راست نموده و گزینه Change/Add Prefix را انتخاب کنید و نام پروژه را به اختصار با چهار حرف در آن بنویسید.

توجه کنید ارتباط بین PLC با فایل Sti از طریق ID که در ستون RunTimeName متغیر مشاهده میکنید برقرار میشود. مشابه همین ID در کانفیگوریشن OPC شما تعریف شده است:

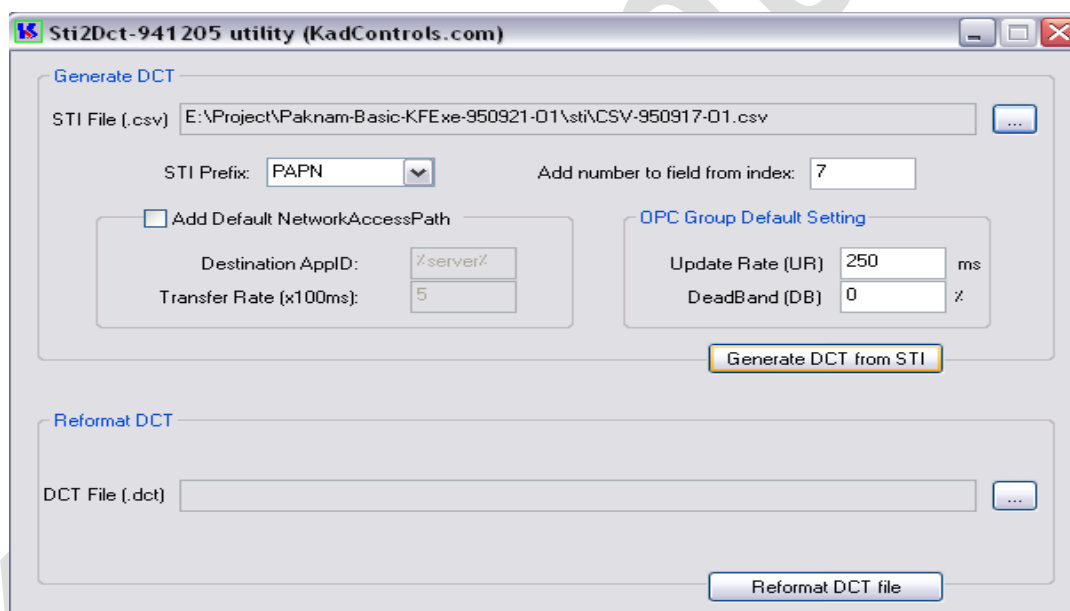
مرجع کار با برنامه مانیتورینگ کادفریم

SV:[PAPN-PLC]DB5,X2,0,1;B;RW;0,0

و ارتباط بین کادفریم با فایل Sti از طریق یک PrefixName است که در نام دایرکتوری فایل Sti تعریف شده و در نگارش DCT که همان OPC داخلی کادفریم است نیز از آن استفاده میشود.

۱۱,۷ مرحله هفتم : تهیه فایل DCT

برای تهیه فایل DCT کافیست فایل csv را داخل Sti2Dct-941205.exe که یک برنامه آگزه در همان دایرکتوری Sti هر پروژه میباشد، درگ و دراپ کنید. سپس دکمه Generate DCT from Sti را انتخاب کنید. فایل DCT در همان دایرکتوری ایجاد میشود که شما میبایست آن را به دایرکتوری به نام DCT انتقال دهید.



شکل 19

۱۱,۸ اساین کردن متغیرها به اتریبهای مناسب

در این مرحله کادفریم را باز کنید و سپس در پنجره AttribScout لازم است که در مقابل اتریبهای تجهیزات کنترلی نام تگ مناسب را اساین کنید. مثلا تگ مربوط به فرمان روشن/خاموش ، نمایش روشن/خاموش ، آلام ، سرعت تند و کند و همچنین مقادیر آنالوگ از فایل DCT میبایست به اتریب مناسب در لیست اتریب های هر تجهیز نسبت

مرجع کار با برنامه مانیتورینگ کادفریم

داده شود. اگر چنانچه آنچه در ستون Tag نوشته شود در DCT موجود نباشد، کادفریم با دادن پیغام خطا رنگ آن را قرمز می کند. توجه کنید هر پنجره Attrib را باید جداگانه ذخیره کنید و فقط ستون هایی که به رنگ قرمز کم رنگ مشخص میشود با کلیک روی Save in ini ذخیره میگردد.

Attribs Scout of KadFrame ('BasicBGPBZ_2' 200 Objects)

Device Names List [Vessel.HopperFeed1]

Indx	Attrib Name	Data Type	Data(Value)	Tag	TagTransform	Visibility
001	[PrcEngGen.PrcEngGen]					
002	[PrcShift.PrcShift]					
003	[PrcAlarmSingle.AlarmDevSP]					
004	[PrcAlarmSingle.AlarmMasterSP]					
005	[PrcAlarmSingle.PrcAlarmSingle02]					
006	[Vessel.HopperFeed1]					
007	[Vessel.HopperBelt0]					
008	[VFeeder.Feed1]					
009	[DBFValve.DBFValve1]					
010	[Belt.Belt01]					
011	[KLabel.DataSource]					
012	[KLabel.RecipeName]					
013	[KLabel.ProcessState]					
014	[KLabel.OperatorName]					
015	[KLabel.DetergentControlMode]					
016	[KLabel.FlowFeed1]					
017	[KLabel.LabelPath]					
018	[KLabel.ExcitationBelt0]					
019	[KLabel.LabelCalcSpBelt0]					
020	[KLabel.LabelCalcSpFeed1]					
021	[KLabel.LabelLoadBelt0]					
022	[KLabel.LabelSpeedBelt0]					
023	[KLabel.DevOpMode_Belt0]					
024	[KLabel.DevOpMode_Feed1]					
025	[KLabel.LabelTotalAutoTitle]					
026	[KLabel.LabelTotalManualTitle]					
027	[KLabel.TotalizerShiftBelt0]					
028	[KLabel.TotalizerShiftFeed1]					
029	[KLabel.TotalizerManualBelt0]					
030	[KLabel.TotalizerManualFeed1]					
031	[KLabel.FlowBelt0]					
032	[KLabel.LabelIsChargingFeed1]					
033	[KLabel.LabelDateShamsi]					
034	[KLabel.LabelVisualAlarm]					
035	[KLabel.LabelBelt0_Weight_1_m]					
036	[KLabel.LabelBelt0_Load_1m]					
037	[KLabel.LabelLengthOfBelt0 - m]					
038	[KLabel.SystemStatus]					
039	[KLabel.OperationMode]					
040	[KLabel.KLabel01]					
041	[KLabel.Device]					
042	[KLabel.Material]					
043	[KLabel.KLabel02]					
044	[KLabel.Dose]					
045	[KLabel.Belt0]					
046	[KLabel.Feed] - Feeder1					
047	[KLabel.KLabel47]					
048	[KLabel.KLabel03]					
049	[KLabel.KLabel04]					
050	[KLabel.KLabelControlMode]					
051	[KLabel.KLabel05]					
052	[KLabel.KLabel06]					
053	[KLabel.KLabel07]					
054	[KLabel.CalibrationMethod]					
055	[KLabel.Effectivelength]					
48	nShapeBottomPipe	Int32	1			Yes
49	nShapeBottomHead	Int32	5			Yes
50	fAutoChargeStartPercent	Float32	20	PAPN.LIW.Feed1.fAutoChargeStartPercent		Yes
51	fAutoChargeStopPercent	Float32	80	PAPN.LIW.Feed1.fAutoChargeStopPercent		Yes
52	fTareLevel	Float32	0.2			Yes
53	fOverloadWarningLevel	Float32 (?)	5			Yes
54	fOverloadErrorLevel	Float32 (?)	30			Yes
55	bNextEligibleFilling	Bool	False			Yes
56	bIsFilling	Bool	False			Yes
57	bShowOnOffMenu	Bool	True			Yes
58	bHasTwoOnOff	Bool	False			Yes
59	bTurnOn1	Bool	True			Yes
60	bIsOn1	Bool	True			Yes
61	bTurnOn2	Bool	True			Yes
62	bIsOn2	Bool	True			Yes
63	bRunHigh	Bool	False			Yes
64	bRunLow	Bool	False			Yes
65	bIncreaseCmd	Bool	False			Yes
66	bDecreaseCmd	Bool	False			Yes
67	bHasAlarm	Bool	False			Yes
68	bInternal_LLA	Bool (?)	False			Yes
69	bInternal_HLA	Bool (?)	False			Yes
70	fAlarmHysteresis_Percent	Float32 (?)	5			Yes
71	Level_Phy	Float32	9	PAPN.AI.LoadFeed1.fPhy_Avg_1Sec		Yes
72	Level_ZeroPhy	Float32	0	PAPN.AI.LoadFeed1.fZeroPhy		Yes
73	Level_SpanPhy	Float32	10	PAPN.AI.LoadFeed1.fSpanPhy		Yes
74	Level_LowLevel	Float32	1	PAPN.AI.LoadFeed1.fLowLevelPhy		Yes
75	Level_HighLevel	Float32	10	PAPN.AI.LoadFeed1.fHighLevelPhy		Yes
76	Level_Port	Float32	0	PAPN.AI.LoadFeed1.fPortValue		Yes
77	Level_ZeroPort	Float32	0	PAPN.AI.LoadFeed1.fZeroPort		Yes
78	Level_SpanPort	Float32	27000	PAPN.AI.LoadFeed1.fSpanPort		Yes
79	Level_Caption	String	Level			Yes
80	Level_Unit	String	Kg			Yes

Tools ☐ Show Only Tagged Save at INI Close

شکل ۲۰

۱۱,۹ تست IO

در این مرحله شما میتوانید پس از تعریف کردن فایل STI به Configuration console ابتدا با OPCScout و سپس با کادفریم متغیرهای PLC را بخوانید یا با سیمولاتورهای خود ورودی های دیجیتال و آنالوگ را تغییر دهید. تست و رفع ایرادات در تک به تک سیگنالهای هر تجهیز در این مرحله ضرورت دارد.

مرجع کار با برنامه مانیتورینگ کادفریم

✓ توجه کنید که پس از انجام هر تغییری در نام فایل STI میبایست آن را مجدد به Configuration Console معرفی کرد و اگر این تغییرات مربوطه به SymbolName یا AccessType باشد میبایست آن را Export نموده و فایل DCT را نیز آپدیت نمود و کادفریم را ری استارت کرد.

۱۱,۱۰ تعریف تگ های پروسس و شیفت و فرمولاسیون

برای هر نوع پروژه ای یک لیست از تگ های پروسس و شیفت و فرمول است که در لیست ورودی خروجی های هاردوری قرار ندارد. بلکه متغیرهای داخلی تبادل شده بین کادفریم و PLC است. این لیست ابتدا در Sti سپس در ini میتواند از سایر پروژه ها مطالعه و بررسی، سپس کپی و در نهایت برای پروژه خاص شما اصلاح شود و در نهایت میبایست در AttribScout به تگهای مناسب اساین گردد.

✓ در این مرحله لازم است که برنامه نویس PLC نیازهای برنامه را برای اجرای پروسه اتوماتیک ایجاد و به سیمولاتورهای پی ال سی ابتدا آن را در سیماتیک منیجر تست نماید و صحت آدرس متغیرهای پروسس را از طریق OPCSCOUT تست و تصحیح نماید.

۱۱,۱۱ تست اجرای پروسه اتوماتیک

در نهایت شما میتوانید با STI, DCT, INI نهایی کانکت به ریل شده و از طریق کادفریم یک پروسه تستی از اجرای انجین کنترل PLC را مشاهده کنید. بیشتر اشکالاتی که در این مرحله مشاهده میشود مربوط به کدنویسی پروسه تولید اتوماتیک است که باید اصلاح شود.

۱۲ ثبت در بانک اطلاعاتی

کادفریم قابلیت این را دارد که مقادیر تکست، تاریخ و آنالوگ (بصورت فلوت) را که تگ های آن در کادفریم تعریف شده اند، بصورت یک گروه از اطلاعات در بانک اطلاعاتی ذخیره کند.

مرجع کار با برنامه مانیتورینگ کادفریم

هر گروه از اطلاعات مورد نظر را در یک جدول (Table) ذخیره میکنیم.

هر جدول شامل ریکورد هاست و هر ریکورد در کادفریم بطور دیفالت دارای شش فیلد اولیه است شامل شماره ریکورد، تاریخ و زمان لاتین ثبت ریکورد، تاریخ و زمان شمسی ثبت ریکورد و

هر ریکورد علاوه بر فیلدهای دیفالت میتواند تعداد یک گروه از تگ های مورد نظر ما را که هر یک از تگ ها بصورت

یک فیلد در این گروه که ریکورد نامیده میشود جای میگیرند .

ثبت یک ریکورد چه زمانی انجام میشود؟ به طریق پریودیک در زمان های معین، با تغییر وضعیت از صفر به یک تگ دیجیتال، تغییر وضعیت از یک به صفر همان تگ دیجیتال و با تاخیر نسبت به تغییر وضعیت اخیر. که انتخاب هر یک از این حالت ها یا همه آنها با هم در اختیار مصرف کننده است.

۱۲،۱ تنظیمات ثبت در بانک اطلاعاتی

اطلاعات همه جداول بانک اطلاعاتی در فولدری به نام Table نوشته می شود. برای هر یک از Table ها باید یک فایل تکست ساده با پسوند Ini ایجاد و مشخصات مورد نیاز جدول در آن با فرمت نمونه داده شده نام گذاری شوند. هر فایل ini دارای سه قسمت خواهد بود:

➤ قسمت [Setting] شامل:

- نام جدول (strTableName=Test1)
 - زمان پریودیک ریکورد کردن بر حسب ثانیه (nRecordingIntervalSec=10)
 - معرفی تگی که با تغییر وضعیت آن یک ریکورد جدید ثبت شود:
- (strTagToTrigRecord=__NO_TAG__) اگر نیاز نیست ثابت __NO_TAG__ گذاشته میشود.
- همچنین تگی که با تغییر وضعیت آن عمل ریکورد کردن ادامه می یابد یا متوقف میشود:
- (strTagToDisableRecording=__NO_TAG__)
- و بلاخره تعیین زمان تاخیری که پس از گذشت آن از آخرین تغییر وضعیت تگ تریگر کننده اصلی یک ریکورد دیگر ثبت شود:

(nDelaySecToRecordAfterCloseActuator=-1) اگر منهای یک باشد عمل نخواهد کرد.

توضیح نیاز چنین ثبت تاخیری این است که در پروسه های کنترل زمانی پس از تغییر وضعیت یک عمل کننده یا اکچواتور (مثلا یک هیتر یا کولر یا پمپ) مقدار آنالوگ اندازه گیری شده مرتبط با این تغییر وضعیت، یک زمان ترانزیت را طی کرده و به ثبات نسبی می رسد و لازم است از این ثبات نیز نمونه برداری شود.

➤ قسمت [Help] شامل فرمت جمله معرفی فیلد ها (تگ هائی که باید ریکورد شوند):

Template=strFieldName, nAveragingBufferSize, strTagName

➤ قسمت [Fields] از سمت چپ شامل:

- نام فیلد در بانک اطلاعاتی : strFieldName
- معدل متغیر آنالوگ (تعداد نمونه را مشخص میکنیم) : nAveragingBufferSize
- و بلاخره تگ متغیری که باید مقدار لحظه ای آن در این فیلد ثبت شود: strTagName

۱۲,۲ مراحل ایجاد یک جدول با مثال

فرض کنید می خواهیم جدولی به نام Event ایجاد کنیم. یکی از نرم افزارهای ویرایش Text مانند Notepad++ را باز کرده و اطلاعات مورد نیاز را مشابه فایل Test موجود در فولدر Table در آن وارد می کنیم.


```

1  [Setting]
2  strTableName=Test1
3  nRecordingIntervalSec=10
4  strTagToTrigRecord=__NO_TAG__ // Comment
5  strTagToDisableRecording=__NO_TAG__ // Comment
6  nDelaySecToRecordAfterCloseActuator=-1 // if -1, will not be triggered
7
8  [Help]
9  Template=strFieldName, nAveragingBufferSize, strTagName
10
11 [Fields]
12 dtStartDate , 0 , dtStartDate_Shift
13 fAnalog1    , 20, fAnalog1
14 fDemoFloat  , 10, fDemoFloat
15

```

شکل 21

اطلاعات قسمت Setting عبارتند از:

- strTableName= Event → نام جدول ذخیره شده در بانک اطلاعاتی
- nRecordingIntervalSec=10 → ثبت دیتا پس از این بازه زمانی
- strTagToTrigRecord=PAPN.Process.bTurnOnValve → تگی که با تریگ شدن آن دیتاها در تیبیل ذخیره می شود.
- strTagToDisableRecording=__NO_TAG__
- nDelaySecToRecordAfterCloseActuator=-1 → ثبت با تاخیر زمانی پس از بسته شدن یک اکتواتور

مرجع کار با برنامه مانیتورینگ کادفریم

در قسمت فیلد به ترتیب نام دیتا در بانک اطلاعاتی، تعداد نمونه برای میانگین گیری و تگ مربوطه نوشته می شود. پس از ذخیره فایل، کادفریم را اجرا کرده، آلازم های مربوطه برای ایجاد فیلدها ظاهر می شود و با تایید کردن تمامی آلازم ها، کادفریم به صورت اتومات بسته می شود. در نهایت Table مورد نظر در بانک اطلاعاتی ذخیره می شود.

۱۳ تگهای محاسباتی

گاهی ممکن است بخواهیم بدون اعمال تغییر در برنامه پی ال سی عملیات محاسباتی را روی تگهای خوانده شده از پی ال سی داشته باشیم مثلاً فرض کنید در یک پروژه ما چندین مخزن حاوی مواد مختلف داریم که با میزان تناژ مشخصی مواد از آن ها خارج می شود و این مقادیر در تگ های مربوط به توتالایزر هر مخزن ذخیره می شود حال اگر ما بخواهیم جمع مصرفی کل مواد را محاسبه کنیم می توانیم به راحتی با استفاده از تگ های محاسباتی تک تک تگهای مربوط به توتالایزر هر مخزن را با هم جمع کرده و به عنوان خروجی مورد نظر خود نمایش دهیم بدون تغییر در اصل برنامه پی ال سی!!!

۱۳,۱ نحوه معرفی و کار با تگ های محاسباتی

برای کار با تگ های محاسباتی ابتدا به مسیر نصب برنامه کادفریم رفته سپس فولدر DCT را باز کرده و از داخل این فالدر فایل ini با نام Basic-CalcTags.ini را باز می کنیم. فایل باز شده از دو قسمت کلی تشکیل می شود

- معرفی تگ
- مشخص نمودن نوع عملیات محاسباتی

مرجع کار با برنامه مانیتورینگ کادفریم

```
1 [Help]
2 DefineTag=(no need '=') TagName, TagDataType, TagDefaultValue
3 SupportedDataType=DataType(Bytes): Float(R4), Int(I4), Bool, String, Byte(UI1), Short(I2), Word(UI2), DWord(UI4), Double(R8), Date
4
5
6 [Define-Tags]
7 bRecordDate1      , Bool , NULL
8 bRecordDate2      , Bool , NULL
9
10 dtStartDate_Shift , Date , NULL
11 dtEndDate_Shift  , Date , NULL
12
13 fRetentive1       , Float , NULL
14 nRetentive2       , Int32 , NULL
15
16
17 fEditBox1_Edit    , Float , NULL
18 fEditBox1_Download , Float , NULL
19
20 bCheckBox1_Edit    , Bool , NULL
21 bCheckBox1_Download , Bool , NULL
22
23 [Calc-Tags]
24
25 bDemoBoolToInt = (bDemoBool)
26
27 nAnalog1 = {fAnalog1}
28
29 dtStartDate_Shift = ((bRecordDate1) == 1) ? {fCurDateTimeMiladi} : {dtStartDate_Shift}
30 bRecordDate1 = 0
31 dtEndDate_Shift = ((bRecordDate2) == 1) ? {fCurDateTimeMiladi} : {dtEndDate_Shift}
32 bRecordDate2 = 0
33
34 //bIsOnTransBelt1 = {PAHP.Belt.bIsOn.ToLeftTransBelt1} || {PAHP.Belt.bIsOn.ToRightTransBelt1}
35 //Render D02DI in Demo for Trans Belts.
36 //PAHP.Belt.bIsOn.ToLeftTransBelt1 = ((eDataSource)==1) ? {PAHP.Belt.bTurnOn.LeftTransBt1} : {PAHP.Belt.bIsOn.ToLeftTransBelt1}
37 // Interlock, Do NOT let to take on both 'bTurnOn.Left' and 'bTurnOn.Right' of Trans Belts.
```

شکل 22

Define-Tags: در این قسمت تگ مورد نظر تعریف می شود که خود شامل سه بخش است:

TagName: نام تگ مورد نظر

TagDataType: نوع تگ مورد نظر

TagDefaultValue: مقدار اولیه تگ مد نظر

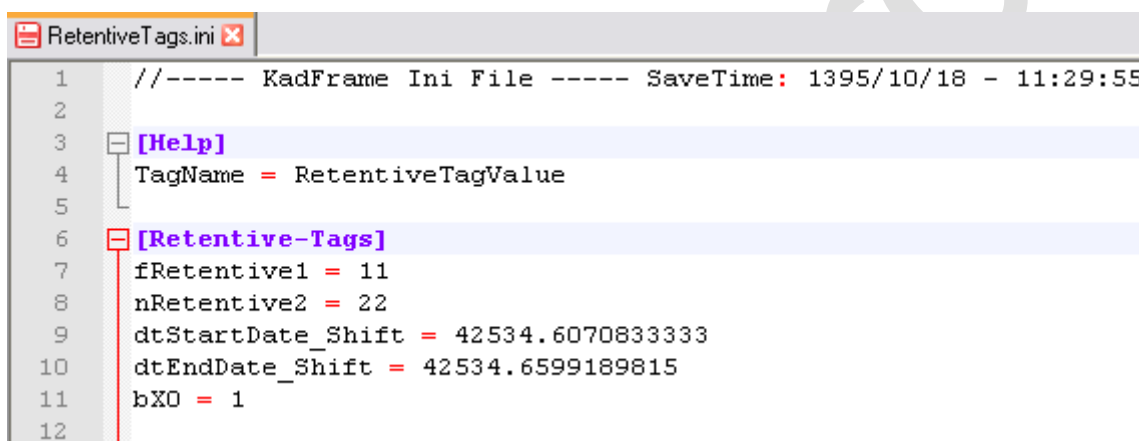
که این سه بخش هر کدام با یک کاما "," از یکدیگر جدا می شوند .

Calc-Tags: در این بخش عملیات محاسباتی مورد نظر را روی تگ تعریف شده در بالا می نویسیم.

فقط باید توجه داشت که برای فراخوانی یک تگ در این فایل باید از علامت {} استفاده کرد.

۱۴ تگ های نگهدارنده (با حافظه ماندگار)

Retentive Tags، تگ هایی هستند که می خواهیم در صورتی که کادفریم به هر دلیلی بسته شد، تگ حالت در حال کار خودش را حفظ کند. به طور مثال برای شروع و پایان شیفت مهم است که در صورت بسته شدن کادفریم، هنگامی که کادفریم مجدداً باز می شود، مقادیر مشخص شده برای شیفت همان مقادیر قبلی باشد. پس در فولدر DCT باید این تگ ها را در فایل RetentiveTags.ini تعریف کنیم (شکل ۲۳).



```
1 //----- KadFrame Ini File ----- SaveTime: 1395/10/18 - 11:29:55
2
3 [Help]
4 TagName = RetentiveTagValue
5
6 [Retentive-Tags]
7 fRetentive1 = 11
8 nRetentive2 = 22
9 dtStartDate_Shift = 42534.6070833333
10 dtEndDate_Shift = 42534.6599189815
11 bX0 = 1
12
```

شکل ۲۳

۱۵ نکات تنظیمات

۱۵,۱ Render DO2DI

توضیح : فرمانی برای عمل کردن یک اکچواتور صادر میشود (DO)، سیستم میخواهد از عمل کردن اکچواتور اطمینان حاصل کند و یا در گرافیک حالت عمل کردن را بازسازی کند و نیاز به یک سیگنال ورودی دارد. چنین سیگنالی ممکن است برای چهار حالت زیر بازسازی گردد. برای تشریح نحوه انتقال اطلاعات از پارامتر DO به DI در مجموعه اتریب های کادفریم باید عملکرد هر کدام از این اتریب ها را حالت های مختلفی که برنامه کادفریم می تواند داشته باشد (Demo,Real) و همچنین خود این اتریب ها از لحاظ داشتن و نداشتن تگ مورد بررسی قرار داد.

مرجع کار با برنامه مانیتورینگ کادفریم

حالت اول برنامه در مد Real و به DI تکی اساین نشده است:

در این حالت $DI=DO$ است یعنی هر مقداری که DO داشته باشد DI نیز همان مقدار را خواهد داشت.

حالت دوم برنامه در مد Real و به DI تگ اساین شده است:

در این حالت $DI=Tag\ DI$ یعنی مقدار DI صرف نظر از مقدار DO برابر تگی است که به آن اساین شده است.

حالت سوم: برنامه در مد Demo است و به DI تگی اساین نشده است

در این حالت $DI=DO$ است یعنی هر مقداری که DO داشته باشد DI نیز همان مقدار را خواهد داشت.

حالت چهارم برنامه در مد Demo است و به DI تگ اساین شده است:

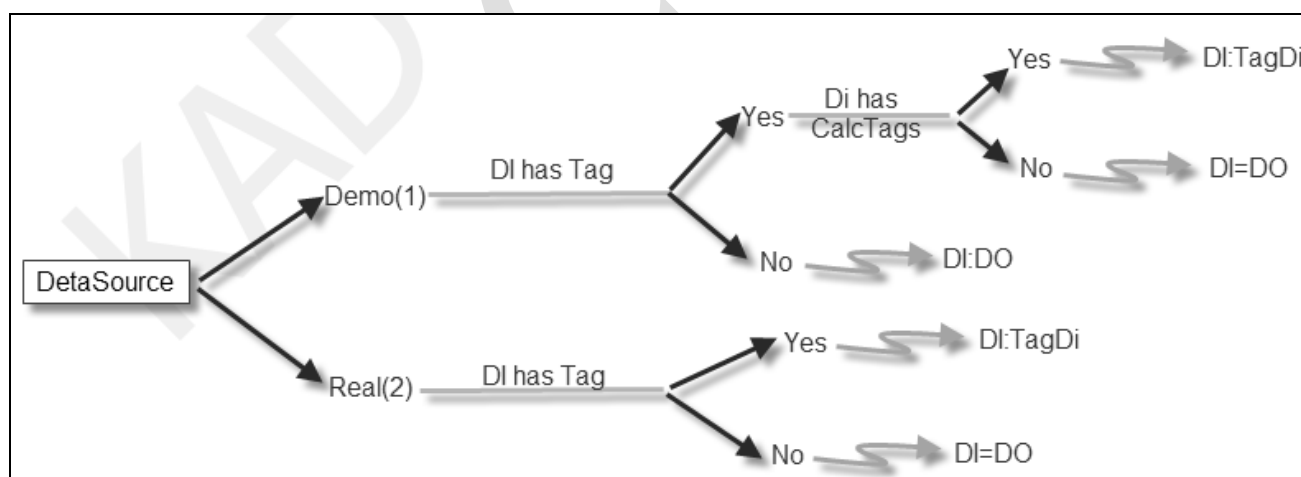
این حالت شامل دو زیر شاخه است:

الف) تگ اساین شده از جنس تگهای محاسباتی نیست:

در این حالت داریم که $DI=DO$

ب) تگ اساین شده از جنس تگهای محاسباتی است:

در این حالت $DI=Tag\ DI$ یعنی مقدار DI صرف نظر از مقدار DO برابر تگی است که به آن اساین شده است.



۱۵,۲ معرفی AutoConnect to Real

کادفریم در دو مد آموزشی (Demo) و مد کاری (Real) عمل می کند. پیش فرض سیستم در ابتدای کار، مد

دمو می باشد. در این مد، برنامه بصورت آموزشی کار خواهد کرد و کنترل های هیچ گونه ارتباطی با تجهیزات واقعی

مرجع کار با برنامه مانیتورینگ کادفریم

موجود در محیط کارخانه ندارند، لیکن برای شروع کار در مد واقعی باید از منوی [Data Source](#) مد برنامه را از حالت Demo به Real تغییر داد تا بتوان با سخت افزار واقعی مخصوص کامپیوتر ارتباط برقرار کرد. دربقیه مواردی که شرح داده خواهد شد فرقی بین مودهای آموزشی و واقعی وجود ندارد.

کادفریم پس از مدت زمان پیش فرض (قابل تنظیم می باشد) با دادن پیغامی برای وصل شدن به مد ریل از کاربر سوال می کند (البته به طور پیش فرض اگر این پیغام را پاسخ ندهیم سیستم به طور خود کار ۱۰ ثانیه پس از ظاهر شدن این پیغام به مد ریل وصل می شود) اما اگر به هر دلیلی نخواهیم که به مد ریل متصل بشویم می توانیم با کلید مجدد بروی مد Demo تایمر شمارش معکوس را غیر فعال کرده و در مد دمو کار کنیم.

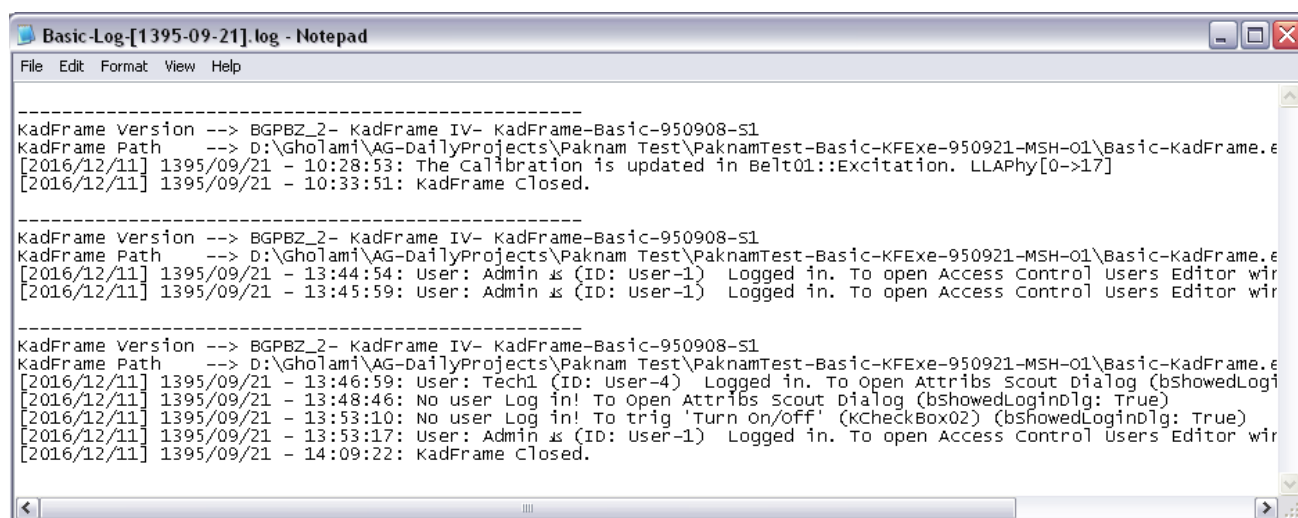
۱۵,۳ معرفی Log در کادفریم

کادفریم دارای سیستم لاگ به شکل فایل تکست است. لاگ های هر روز به صورت یک فایل جداگانه در فولدر Log ایجاد می شود. تاریخ ایجاد فایل لاگ جز قسمتی از نام فایل لاگ است. در ضمن فولدر Log در مسیری که Exe کادفریم قرار دارد ایجاد میشود. فایل لاگ فقط در صورتی ایجاد می شود که اتفاق مهم و نیاز به بررسی آتی در کادفریم رخ دهد.

۱۵,۳,۱ فرمت فایل Log

همانطور که گفته شد فایل Log از نوع تکست است. همیشه قبل از اولین Log، چند خط تیره به همراه ورژن کادفریم و مسیری که کادفریم از آن اجرا شده است یک بار در ابتدای هر اجرای کادفریم آورده میشود. (طبق شکل ۲۴)

مرجع کار با برنامه مانیتورینگ کادفریم



```
-----
KadFrame Version --> BGPBZ_2- KadFrame IV- KadFrame-Basic-950908-S1
KadFrame Path --> D:\Gholami\AG-DailyProjects\Paknam Test\PaknamTest-Basic-KFExe-950921-MSH-01\Basic-KadFrame.e
[2016/12/11] 1395/09/21 - 10:28:53: The Calibration is updated in Belt01::Excitation. LLAPhy[0->17]
[2016/12/11] 1395/09/21 - 10:33:51: KadFrame Closed.
-----
KadFrame Version --> BGPBZ_2- KadFrame IV- KadFrame-Basic-950908-S1
KadFrame Path --> D:\Gholami\AG-DailyProjects\Paknam Test\PaknamTest-Basic-KFExe-950921-MSH-01\Basic-KadFrame.e
[2016/12/11] 1395/09/21 - 13:44:54: User: Admin & (ID: User-1) Logged in. To open Access Control Users Editor wir
[2016/12/11] 1395/09/21 - 13:45:59: User: Admin & (ID: User-1) Logged in. To open Access Control Users Editor wir
-----
KadFrame Version --> BGPBZ_2- KadFrame IV- KadFrame-Basic-950908-S1
KadFrame Path --> D:\Gholami\AG-DailyProjects\Paknam Test\PaknamTest-Basic-KFExe-950921-MSH-01\Basic-KadFrame.e
[2016/12/11] 1395/09/21 - 13:46:59: User: Tech1 (ID: User-4) Logged in. To Open Attribs Scout Dialog (bshowedLogi
[2016/12/11] 1395/09/21 - 13:48:46: No user Log in! To Open Attribs Scout Dialog (bshowedLoginDlg: True)
[2016/12/11] 1395/09/21 - 13:53:10: No user Log in! To trig 'Turn on/off' (KCheckBox02) (bshowedLoginDlg: True)
[2016/12/11] 1395/09/21 - 13:53:17: User: Admin & (ID: User-1) Logged in. To open Access Control Users Editor wir
[2016/12/11] 1395/09/21 - 14:09:22: KadFrame Closed.
```

شکل ۲۴

اگر کادفریم با موفقیت بسته شود همیشه آخرین لاگ پیغام KadFrame Closed خواهد بود. پس می توان نتیجه گرفت که اگر پیغام KadFrame Closed ما بین دو بلوک لاگ (هر بلوک با ورژن کادفریم شروع می شود) وجود نداشت، نشان از بسته شدن ناموفق کادفریم دارد.

هر Log به صورت یک سطر جداگانه که در ابتدای آن زمان Log اضافه شده است، ثبت می شود. اگر لاگی هنگام شروع کادفریم ثبت شود، فایل لاگ ایجاد شده در Notepad و فولدر حاوی فایل لاگ به صورت اتومات توسط کادفریم برای جلب توجه بیشتر باز می شود.

اکشن هایی که کادفریم ان را لاگ میکند به چند گروه زیر تقسیم می شود:

- تغییرات در کالیبراسیون به صورت دستی و اتومات
- هر نوع لاگین در سیستم حق دسترسی کادفریم
- هر نوع Clean Up در فولدر Ini

۱۵,۳,۲ تغییرات دستی و یا اتومات در کالیبراسیون

الف) تغییر کالیبراسیون به صورت دستی

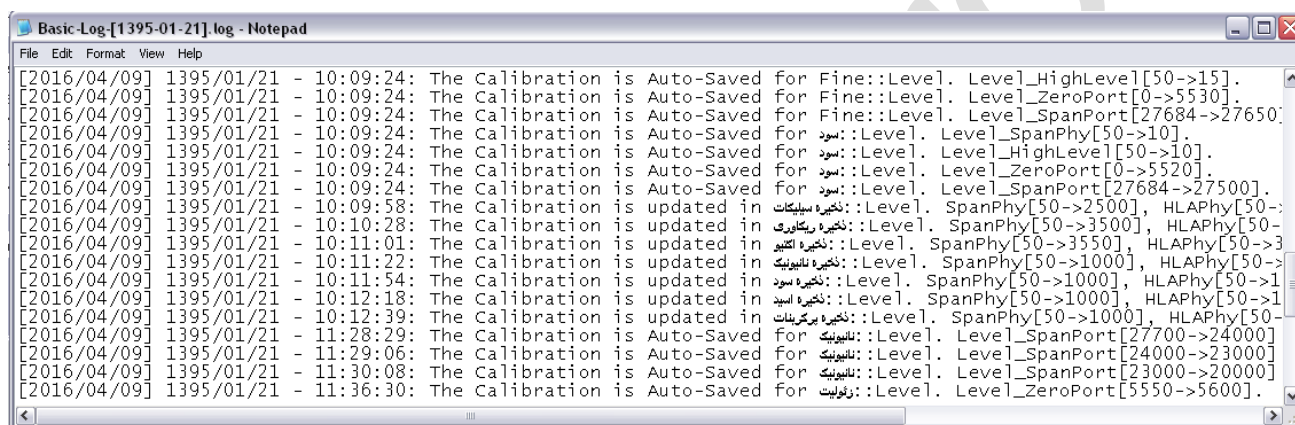
اگر کالیبراسیون وسیله تحت کنترل ما به صورت دستی از طریق برنامه کادفریم تغییر کند این تغییرات در فولدر

لاگ تحت عنوان تغییرات دستی ثبت می شود

ب) تغییر کالیبراسیون به صورت اتومات:

اگر کالیبراسیون وسیله تحت کنترل ما از منبعی به غیر از کادفریم (پی ال سی و یا نسخه دیگری از کادفریم) تغییر

کند این اطلاعات تحت عنوان تغییرات اتومات لاگ می شود.



شکل ۲۵

همانطور که در شکل ۲۵ می بینید بعد از تغییر در کالیبراسیون چند دیوایس فولدر لاگ اطلاعات مربوط به این تغییرات را در خود ذخیره کرده و به همراه تاریخ و ساعت اعمال تغییرات نمایش می دهد. همچنین تغییرات کالیبراسیون به صورت اتومات به صورت Auto-Saved ذخیره شده است.

۱۵,۳,۳ تغییر اطلاعات دارای سطح دسترسی

به طور کلی برای افزایش ضریب امنیت و جلوگیری از تغییرات ناخواسته برای اطلاعات مهم، حق دسترسی تعریف می کنیم و همچنین برای امنیت بیشتر ردپای تغییرات توسط افرادی که دارای سطح دسترسی هستند نیز در فولدر لاگ ثبت می شود تا در صورت بروز هرگونه مشکل تصمیم گیری های لازم به عمل آید.


```

Basic-Log-[1395-07-29].log - Notepad
File Edit Format View Help
-----
KadFrame Version --> MYSL- KadFrame Free Version- KadFrame-Basic-950510-01
KadFrame Path --> F:\kad\SL\Kadframe\MYSL-BKFEexe-950728-S2-MD\Basic-KadFrame.exe
[2016/10/20] 1395/07/29 - 16:41:50: User: آقای جلالیان (ID: User-4) Logged in. To Open Attribs Scout Dialog
[2016/10/20] 1395/07/29 - 16:56:41: User: آقای جلالیان (ID: User-4) Logged in. To Open Attribs Scout Dialog
[2016/10/20] 1395/07/29 - 16:57:00: User: آقای جلالیان (ID: User-4) Logged in. To Open Attribs Scout Dialog
[2016/10/20] 1395/07/29 - 17:16:38: User: آقای جلالیان (ID: User-4) Logged in. To Open Attribs Scout Dialog
[2016/10/20] 1395/07/29 - 17:16:52: User: آقای جلالیان (ID: User-4) Logged in. To save Attribs Scout setting
[2016/10/20] 1395/07/29 - 17:17:03: User: آقای جلالیان (ID: User-4) Logged in. To save Attribs Scout setting
[2016/10/20] 1395/07/29 - 17:18:10: User: آقای جلالیان (ID: User-4) Logged in. To save Attribs Scout setting
[2016/10/20] 1395/07/29 - 17:19:03: User: آقای جلالیان (ID: User-4) Logged in. To Open Attribs Scout Dialog
[2016/10/20] 1395/07/29 - 17:21:43: User: آقای جلالیان (ID: User-4) Logged in. To Open Attribs Scout Dialog
[2016/10/20] 1395/07/29 - 22:15:14: User: آقای دانش پژو (ID: User-3) Logged in. To stop active 'Shift' (bshc
[2016/10/20] 1395/07/29 - 22:15:18: User: آقای دانش پژو (ID: User-3) Logged in. To start new 'shift'
    
```

شکل ۲۶

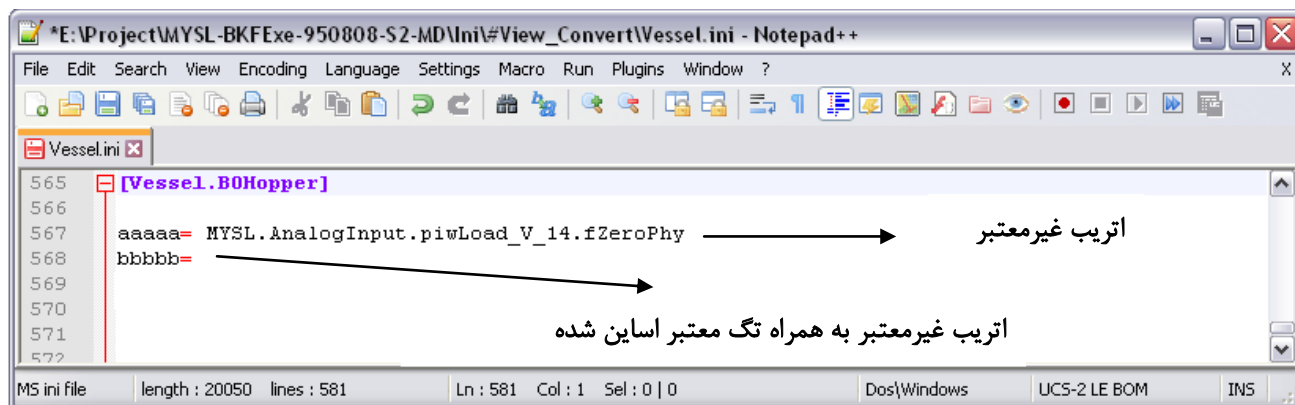
همانطور که در بالا می بینیم ابتدا یوزر شماره ۴ آقای به نام جلالیان تغییراتی را در اتریب اسکات اعمال کرده و سپس آقای دانش پژو تحت عنوان یوزر ۳ شیفت فعال را متوقف کرده است.

۱۵,۳,۴ مفهوم Clean Up در فایل Ini

کادفریم برای آنکه در Ini اطلاعات زاید وجود نداشته باشد هر ۸ ساعت یک بار در هر اجرای جدید کادفریم اقدام به چک کردن تمامی فایل های Ini می کند. تا موارد اطلاعات نامعتبر شناسایی شده و یا به صورت اتومات و در صورت مهم بودن موارد با گرفتن تایید از کاربر پاک شوند. به این عملیات Clean Up گفته می شود. مواردی که کادفریم در چک دوره ای خود به عنوان اطلاعات زاید شناسایی می کند عبارتند از:

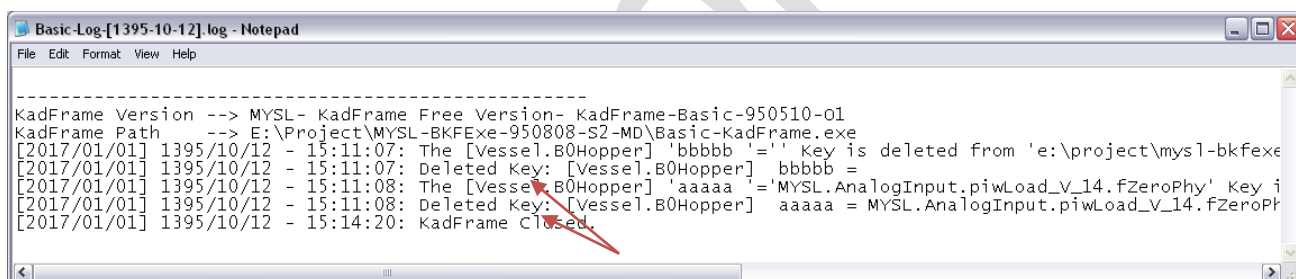
○ هر Key بدون Value:

کادفریم به طور کلی برای هر Key ای که از دید آن نامعتبر باشد، چه تگ معتبر به آن اساین شده باشد و چه جلوی آن خالی باشد، آلام می دهد. در صورت تایید آلام Key از داخل Ini حذف شده و این اطلاعات در فایل لاگ نوشته می شود.



شکل ۲۷

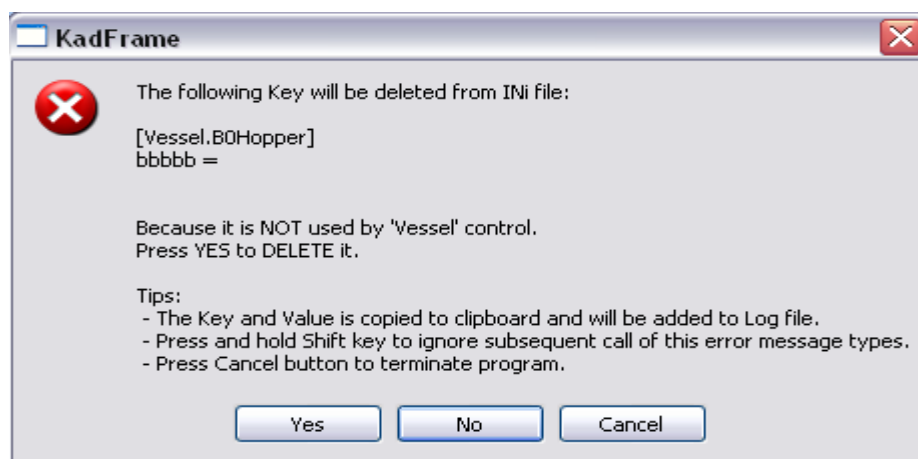
در شکل بالا در فولدر Ini مربوط به مخزن ها دو اتریب غیر معتبر تعریف شده که یکی از آنها بدون Value است و دیگری دارای تگی معتبر است، پس از Save و اجرای کادفریم و همچنین تایید آلارم ها برای حذف این دو اتریب فولدر لاگ ساخته شده مانند شکل ۲۸ است:



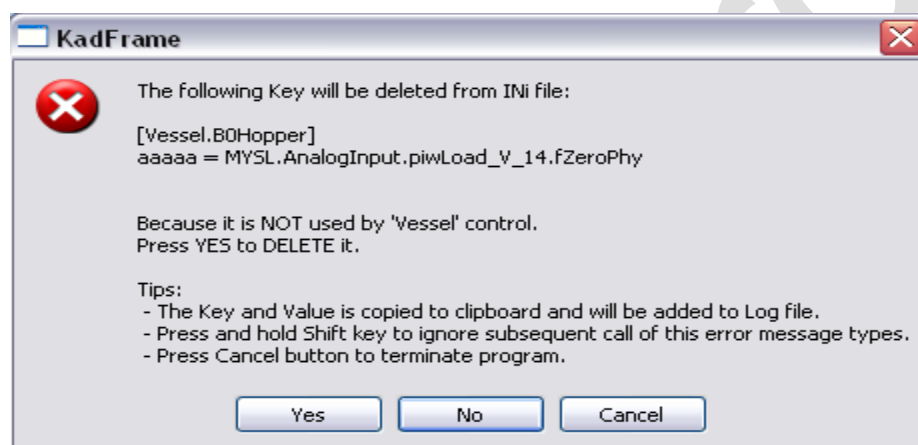
شکل ۲۸

همانطور که مشاهده می کنید تمام اتریب های ساخته شده توسط کادفریم پاک شده و این تغییرات در فولدر لاگ ذخیره می شود.

توجه) در مثال بالا هنگام Run کردن کادفریم پیغام های زیر ظاهر می شود:



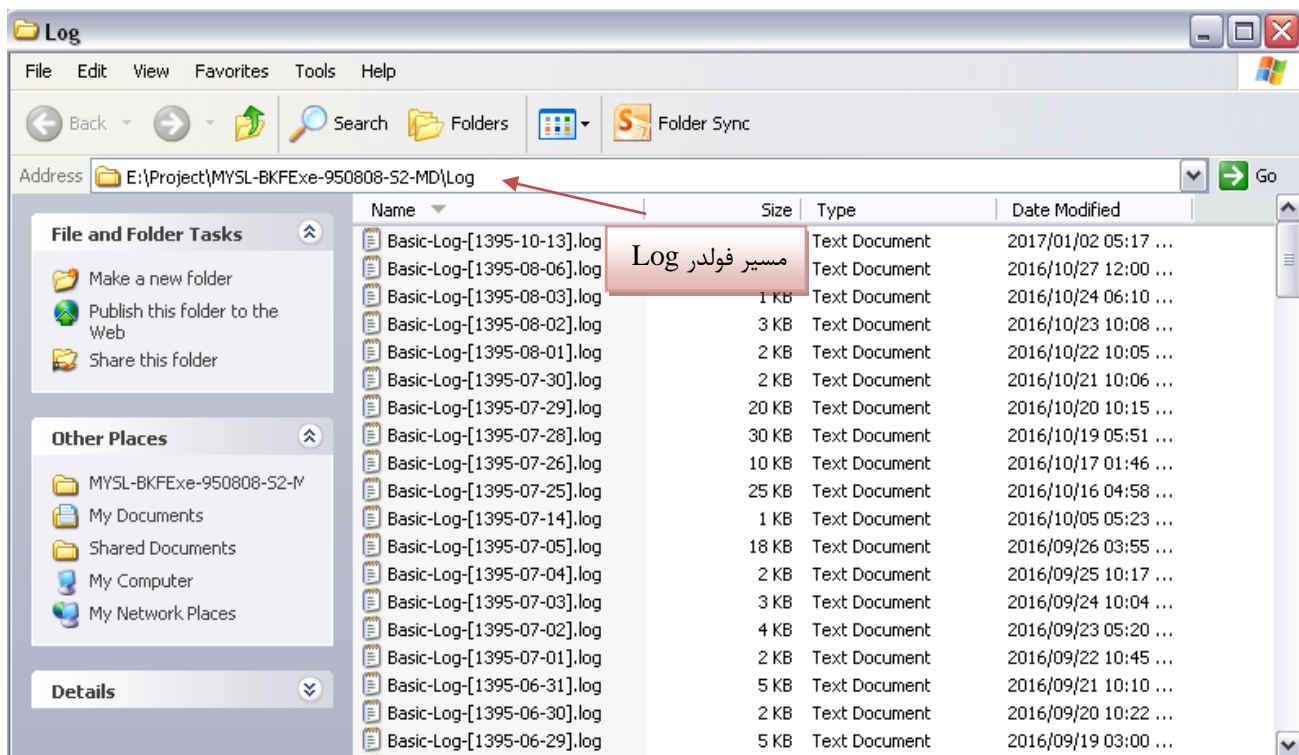
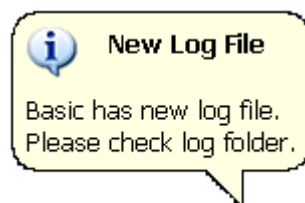
شکل ۲۹



شکل ۳۰

- همانطور که مشاهده می کنیم کادفریم برای پاک کردن اتریب نامعتبر bbbbbb و aaaaa از کاربر اجازه می گیرد. اگر کاربر گزینه تایید را بزند، پیغام زیر بر روی کادفریم دیده می شود و صفحه ی Log که نشان می دهد اتریب ها پاک شده اند (شکل ۲۸)، نمایش داده می شود. در واقع کادفریم به طور اتومات همواره پس از ایجاد یک لاگ جدید برای تاکید بر ضرورت بررسی فولدر لاگ با ظاهر کردن بالنی مبنی بر ایجاد فایل لاگ جدید فایل لاگ به همراه فولدر ذخیره آن را همراه با باز شدن کادفریم باز میکند (شکل ۳۱).

مرجع کار با برنامه مانیتورینگ کادفریم



شکل ۳۱

۱۵,۳,۵ نکات مهم در مورد فولدر لاگ

- پس از ایجاد فولدر لاگ بهتر است ابتدا آن را مشاهده کرده و اگر تغییرات ثبت شده در آن مورد تایید طراح کادفریم قرار داشت فولدر ایجاد شده پاک شود.
- کادفریم برای آنکه سرعت Load شدن آن در هنگام اجرا زیاد باشد همیشه هنگام اجرا، فایل های Ini را برای پاک سازی از اطلاعات زاید بررسی نمی کند و هر ۸ ساعت یک بار این عمل را انجام می دهد. برای آنکه در طول طراحی بتوانیم این اطلاعات زاید را پاک سازی نماییم باید هنگام اجرای برنامه کادفریم کلید Ctrl را نگه داشته و کادفریم را اجرا کنید در این شرایط کادفریم شروع به بررسی فایل های Ini برای پاک سازی اطلاعات زاید می کند. این عمل چک کردن تا زمانی ادامه می یابد که دیگر لاگ جدیدی از آخرین باری که

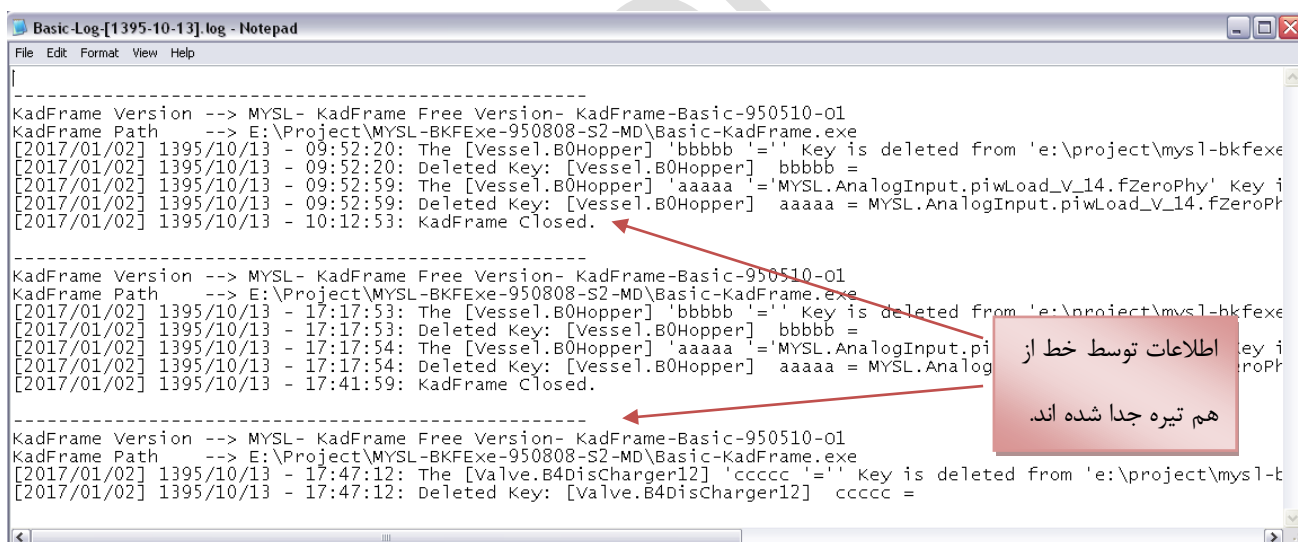
مرجع کار با برنامه مانیتورینگ کادفریم

کادفریم را با نگه داشتن کلید کنترل اجرا کردیم ایجاد نشود و بعد از آن مجدد هر ۸ ساعت یک بار این بررسی صورت می گیرد.



(هنگام نگه داشتن کلید کنترل و اجرای کادفریم پیغام بالا نمایش داده می شود)

- اگر شرایط ایجاد لاگ پیش بیاید در هر بار باز و بسته شدن کادفریم اطلاعات باز و بسته شدن کادفریم در فولدر لاگ ثبت و توسط خط تیره از هم جدا می شوند (شکل ۳۲).

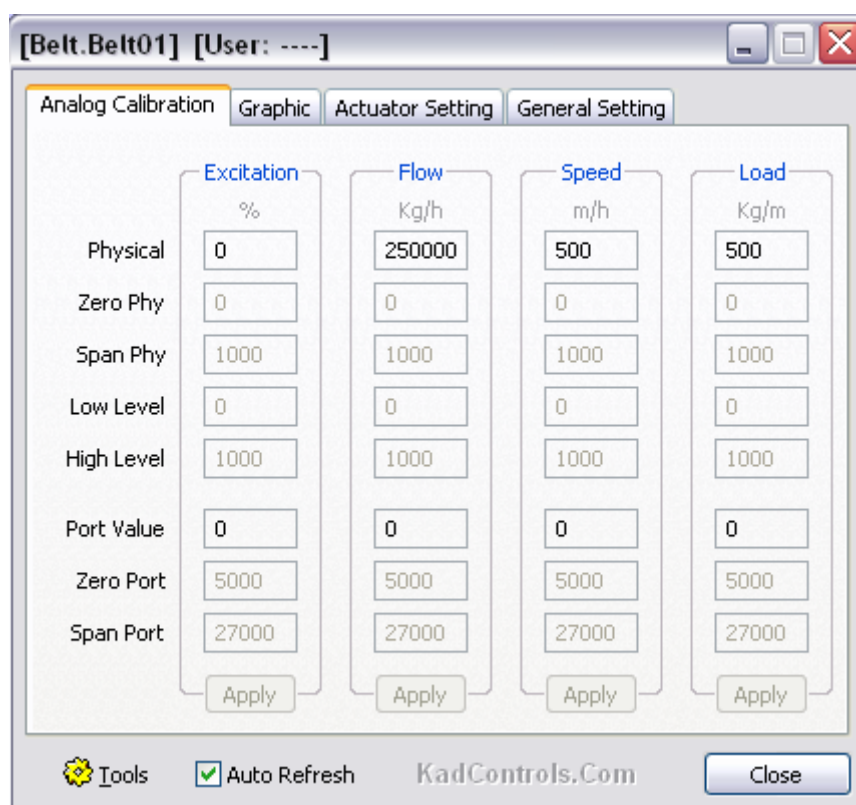


شکل ۳۲

۱۵,۴ تنظیمات کنترلی اشیا

۱۵,۴,۱ کالیبراسیون آنالوگ

اگر روی هر کدام از اشیا گرافیکی کادفریم دوبار کلید کنیم صفحه Properties متناسب با آن کنترل باز می شود، این صفحه که یک صفحه Non-Modal است، حاوی تنظیمات کالیبراسیون آن شی می باشد.



شکل ۳۳

بسته به نوع دیوایس Analog Calibration دارای ستون هایی مربوط به موارد کنترلی آن دیوایس است که هر یک شامل سه ردیف با نام های Physical , Min Phy و Max Phy است که نشان دهنده مقادیر با واحد فیزیکی جاری مواد و مقدار کمترین و بیشترین مجاز در مورد آن مشخصه و دو ردیف Low Level و High Level فیزیکی برای نشان دهنده آلارم و سه ردیف Port Value و Min Value و Max Value که نشان دهنده مقدار جاری و کمترین و بیشترین مقدار آنالوگ ورودی (عدد قرائت شده از پورت آنالوگ) مربوطه به دو مشخصه بار و تناژ نوار است و پورت جاری مقدار جاری پارامترهای بار و تناژ نوار می باشد. تنظیم این پارامترها قسمتی از عملیات کالیبراسیون است که توسط واحد ابزار دقیق انجام میگیرد.

همانطور که گفته شد در این پنجره مشخصات کالیبراسیون مربوط به هر شی گرافیکی نمایش داده میشود که برای عدم نمایش این مشخصات در صفحه کالیبراسیون کافی است در پنجره Attribs Scout مربوط به همان شی مشخصه Visibility اتریب مربوطه را No کرد.

به عنوان مثال به شکل های زیر توجه کنید:

پنجرہ ی Properties یک مخزن

مشخصه Visibility

Attribs Scout of KadFrame ("BasicBGPBZ_2" 200 Objects)

Device Names List

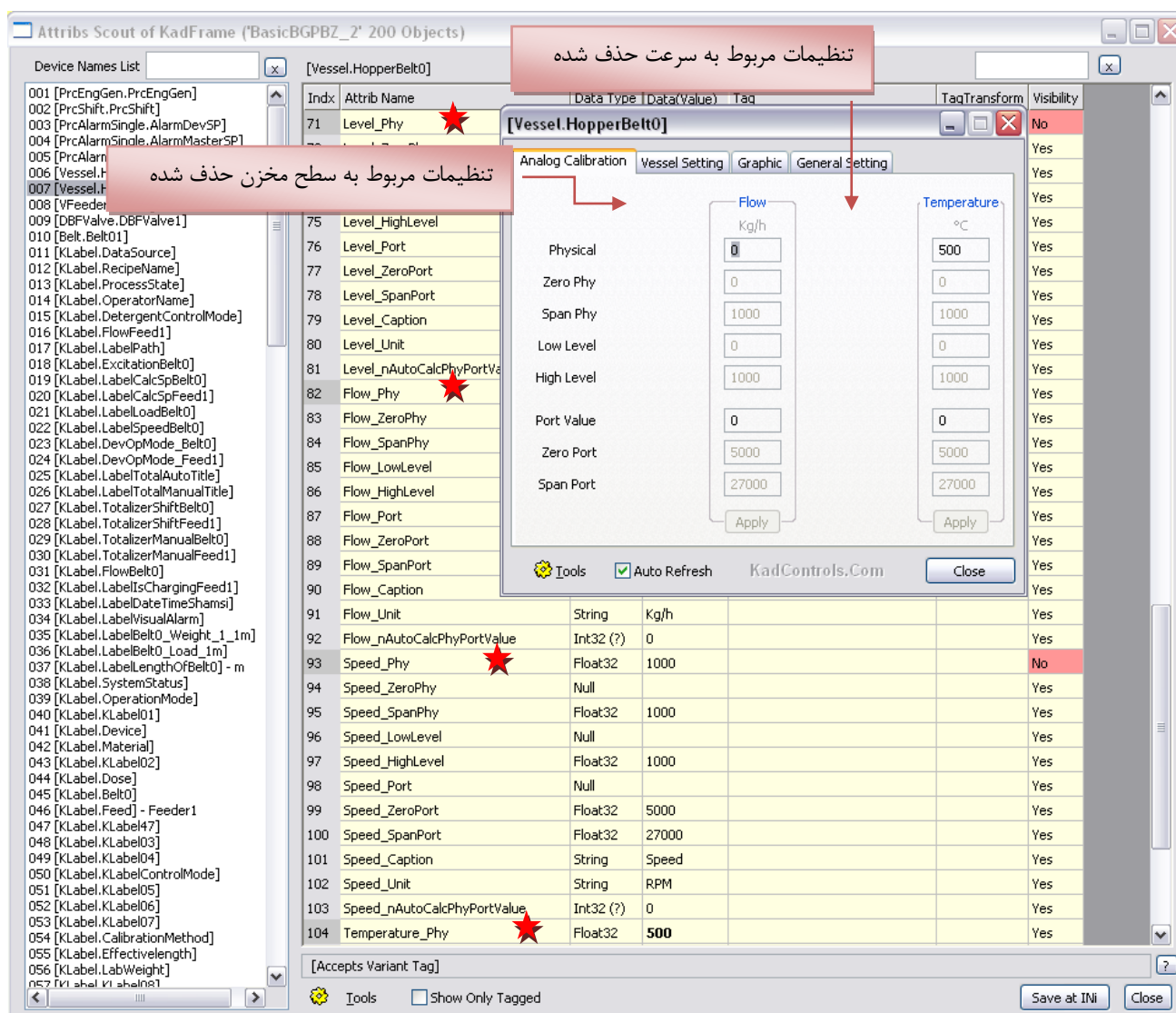
- 001 [PrcEngGen.PrcEngGen]
- 002 [PrcShift.PrcShift]
- 003 [PrcAlarmSingle.AlarmDevSP]
- 004 [PrcAlarmSingle.AlarmMasterSP]
- 005 [PrcAlarmSingle.PrcAlarmSingle02]
- 006 [Vessel.HopperFeed1]
- 007 [Vessel.HopperBelt0]
- 008 [VFeeder.Feed1]
- 009 [DBFValve.DBFValve1]
- 010 [Belt.Belt01]
- 011 [KLabel.DataSource]
- 012 [KLabel.RecipeName]
- 013 [KLabel.ProcessState]
- 014 [KLabel.OperatorName]
- 015 [KLabel.DetergentControlMode]
- 016 [KLabel.FlowFeed1]
- 017 [KLabel.LabelPath]
- 018 [KLabel.ExcitationBelt0]
- 019 [KLabel.LabelCalcSpBelt0]
- 020 [KLabel.LabelCalcSpFeed1]
- 021 [KLabel.LabelLoadBelt0]
- 022 [KLabel.LabelSpeedBelt0]
- 023 [KLabel.DevOpMode_Belt0]
- 024 [KLabel.DevOpMode_Feed1]
- 025 [KLabel.LabelTotalAutoTitle]
- 026 [KLabel.LabelTotalManualTitle]
- 027 [KLabel.TotalizerShiftBelt0]
- 028 [KLabel.TotalizerShiftFeed1]
- 029 [KLabel.TotalizerManualFeed1]
- 030 [KLabel.TotalizerManualFeed1]
- 031 [KLabel.FlowBelt0]
- 032 [KLabel.LabelIsChargingFeed1]
- 033 [KLabel.LabelDateTimeShamsi]
- 034 [KLabel.LabelVisualAlarm]
- 035 [KLabel.LabelBelt0_Weight_1_1m]
- 036 [KLabel.LabelBelt0_Load_1m]
- 037 [KLabel.LabelLengthOfBelt0] - m
- 038 [KLabel.SystemStatus]
- 039 [KLabel.OperationMode]
- 040 [KLabel.KLabel01]
- 041 [KLabel.Device]
- 042 [KLabel.Material]
- 043 [KLabel.KLabel02]
- 044 [KLabel.Dose]
- 045 [KLabel.Belt0]
- 046 [KLabel.Feed] - Feeder1
- 047 [KLabel.KLabel47]
- 048 [KLabel.KLabel03]
- 049 [KLabel.KLabel04]
- 050 [KLabel.KLabelControlMode]
- 051 [KLabel.KLabel05]
- 052 [KLabel.KLabel06]
- 053 [KLabel.KLabel07]
- 054 [KLabel.CalibrationMethod]
- 055 [KLabel.Effectivelength]
- 056 [KLabel.LabWeight]
- 057 [KLabel.KLabel08]

Indx	Attrib Name	Data Type	Data(Value)	Tag	TagTra...	Visibility
71	Level_Ph					Yes
72	Level_ZeroPh					Yes
73	Level_SpanPh					Yes
74	Level_LowLevel					Yes
75	Level_HighLevel					Yes
76	Level_Port					Yes
77	Level_ZeroPort					Yes
78	Level_SpanPort					Yes
79	Level_Caption					Yes
80	Level_Unit					Yes
81	Level_nAutoCalcPhyPortValue					Yes
82	Flow_Ph					Yes
83	Flow_ZeroPh					Yes
84	Flow_SpanPh					Yes
85	Flow_LowLevel					Yes
86	Flow_HighLevel					Yes
87	Flow_Port					Yes
88	Flow_ZeroPort					Yes
89	Flow_SpanPort					Yes
90	Flow_Caption					Yes
91	Flow_Unit	String	Kg/h			Yes
92	Flow_nAutoCalcPhyPortValue	Int32 (?)	0			Yes
93	Speed_Ph	Float32	1000			Yes
94	Speed_ZeroPh	Null				Yes
95	Speed_SpanPh	Float32	1000			Yes
96	Speed_LowLevel	Null				Yes
97	Speed_HighLevel	Float32	1000			Yes
98	Speed_Port	Null				Yes
99	Speed_ZeroPort	Float32	5000			Yes
100	Speed_SpanPort	Float32	27000			Yes
101	Speed_Caption	String	Speed			Yes
102	Speed_Unit	String	RPM			Yes
103	Speed_nAutoCalcPhyPortValue	Int32 (?)	0			Yes
104	Temperature_Ph	Float32	500			Yes

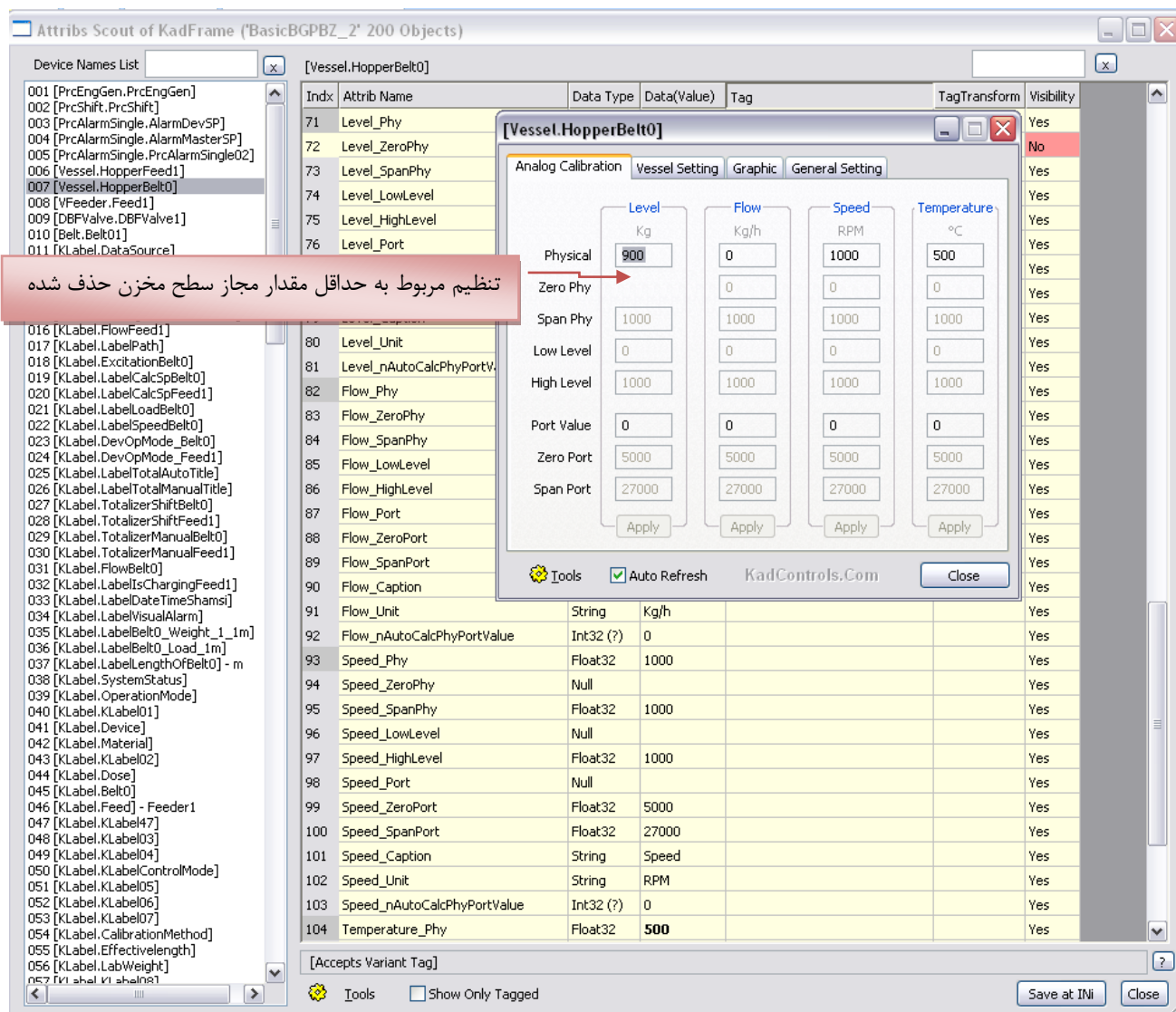
[Accepts Variant Tag]

☐ Show Only Tagged

شكل ٣٤



شکل ۳۵



شکل ۳۶

در این پنجره می توان تنظیمات رنگ مربوط به کنترل انتخابی را تغییر داد.



شکل ۳۷

قسمت گرافیک مربوط به تغییر رنگ مربوط به کنترل وسیله مورد نظر است، که شامل موارد زیر است:

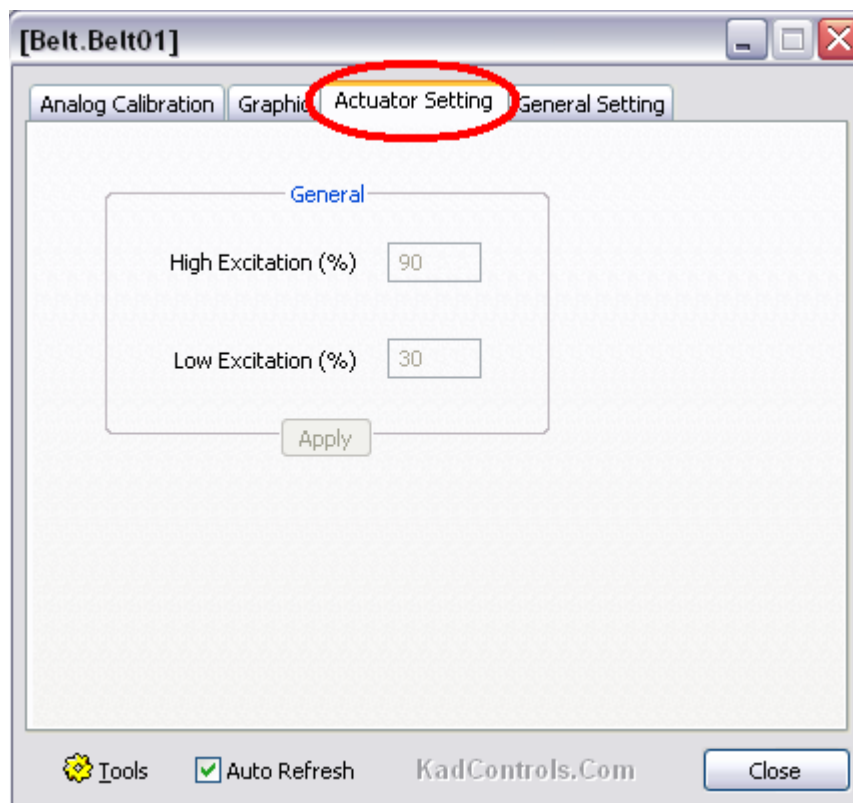
رنگ Back Ground، Foreground، Body، کنترل محلی (لوکال)، اورلود و غیر فعال بودن در فرمولاسیون (مد دستی) است.

با کلیک روی هر گزینه میتوانید رنگ مورد نظر خود را انتخاب کنید سپس دکمه Save Colors را کلیک کنید تا تغییر رنگ تعریف شده ذخیره شود.

○ **AutoReferesh:** برای ادیت کردن مقادیر پارامترها کافیست در این چک باکس تیک بزنید.

Actuator Setting ۱۵,۴,۳

در این پنجره می توان تنظیمات مربوط به Low/High کنترل انتخابی را تغییر داد.



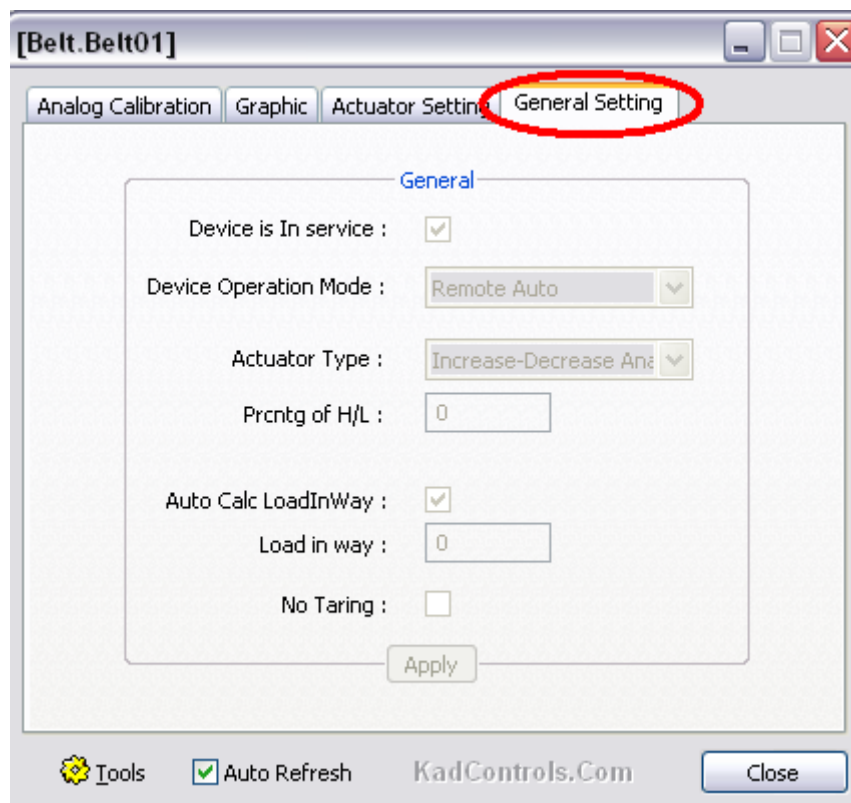
شکل ۳۸

در قسمت High Excitation درصد سرعت بالا را می توان تعیین کرد، که از فرمول زیر تبعیت می کند:

$$\text{Excitation} = (\text{High Excitation} / 100) * \text{Span Excitation}$$

در قسمت Low Excitation درصد سرعت پایین مشخص می شود، که طبق فرمول زیر داریم:

$$\text{Excitation} = (\text{Low Excitation} / 100) * \text{Span Excitation}$$



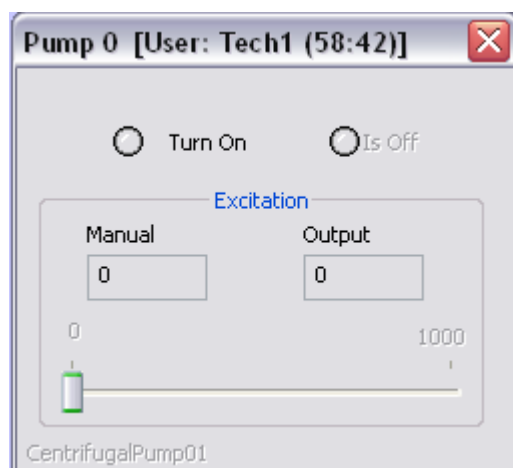
شکل ۳۹

توضیحات این بخش در صفحات ۳۱ تا ۳۳ توضیح داده شده است.

۱۵,۵ صفحه اکشن

باکلید راست کردن روی هر دیواس منویی باز می شود که می توان از منوی باز شده روی گزینه Action کلید کرد، تا پنجره Action باز شود.

مرجع کار با برنامه مانیتورینگ کادفریم



شکل ۴۰

در این پنجره Turn On و Is Off به ترتیب مربوط به روشن و خاموش کردن عملگر و فیدبک آن می باشد و

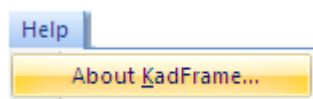
همچنین با تغییر مقدار تحریک، مقدار آن در قسمت Manual نشان داده می شود.

نکته) پس از باز شدن صفحه Action در صورتی که حق دسترسی برای آن مشخص شده باشد، تا زمانی که یوزر

تعیین کرده است پنجره باز می ماند و پس از آن به طور خودکار بسته خواهد شد.

نکته) پس از باز شدن پنجره Action تنها با کلید چپ کردن رو اشیاء دیگر می توان بین صفحات اکشن حرکت کرد.

۱۶ قابلیت About Box



از قسمت Help می توانید پنجره About Kadframe را انتخاب کنید.

مرجع کار با برنامه مانیتورینگ کادفریم

در این قسمت تعداد تمام تگ هایی که در برنامه و در درایورهای مختلف استفاده شده، نمایش داده میشود.

System Tag: تعداد تگ های داخلی

NonHW Tag: تعداد تگ های محاسباتی

OPC Tag: تعداد تگ های OPC

ModbusRTU Tag: تعداد تگ های مادباس آر تی یو

Hardware Linked Tag Count: تعداد تگ هایی که به سخت افزار متصل است.



شکل 41

۱۷ قابلیت دیکشنری

کاربرد دیکشنری در نمایش دادن مقدار های متفاوت در لیبل ها و کمبو باکس ها است. ابتدا در قسمت Main.ini

دیکشنری را تعریف کرده مثلا گروه ۱ با حالت های ۰ تا ۱۰:

[Dictionary.Group1]

0=-----

1=strMaterialName_B1

2=strMaterialName_2_1

مرجع کار با برنامه مانیتورینگ کادفریم

3=strMaterialName_2_2

4=strMaterialName_2_3

5=strMaterialName_3

6=strMaterialName_4

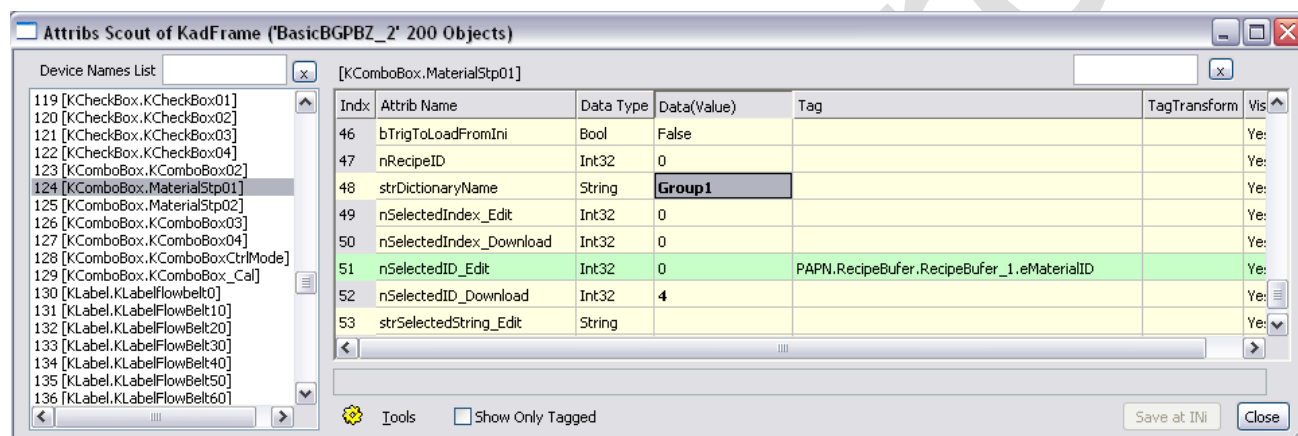
7=strMaterialName_5

8=strMaterialName_6

9=Confirmation

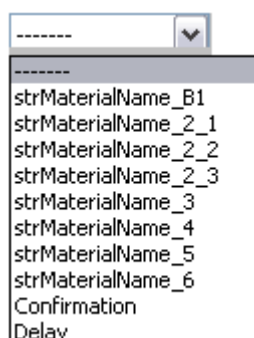
10=Delay

سپس در اتریب اسکات، اتریب strDictionaryName، قسمت Data (value) را که در این مثال Group1 است، اضافه میکنیم.



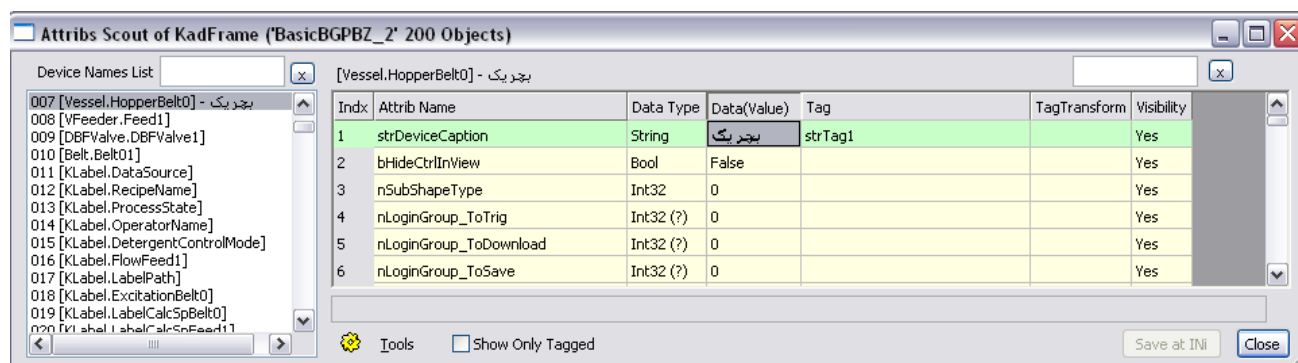
شکل ۴۲

همانطور که در شکل زیر مشاهده میکنید متغیر هایی که از ۰ تا ۱۰ تعریف کرده بودیم در کمبو باکس نمایش داده میشود:



مرجع کار با برنامه مانیتورینگ کادفریم

اگر خواستیم مقدار متغیر ما از یک Vessel گرفته شود باید یک تگ محاسباتی به قسمت strDeviceCaption مخزن اساین کنیم و در قسمت Data(value) یک نام مثل بچر یک تعریف کنیم:



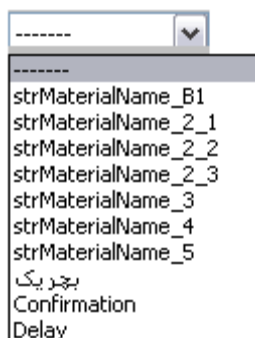
شکل 43

سپس همان تگ را در INI تعریف میکنیم مثلاً در این مثال مطابق شکل زیر:

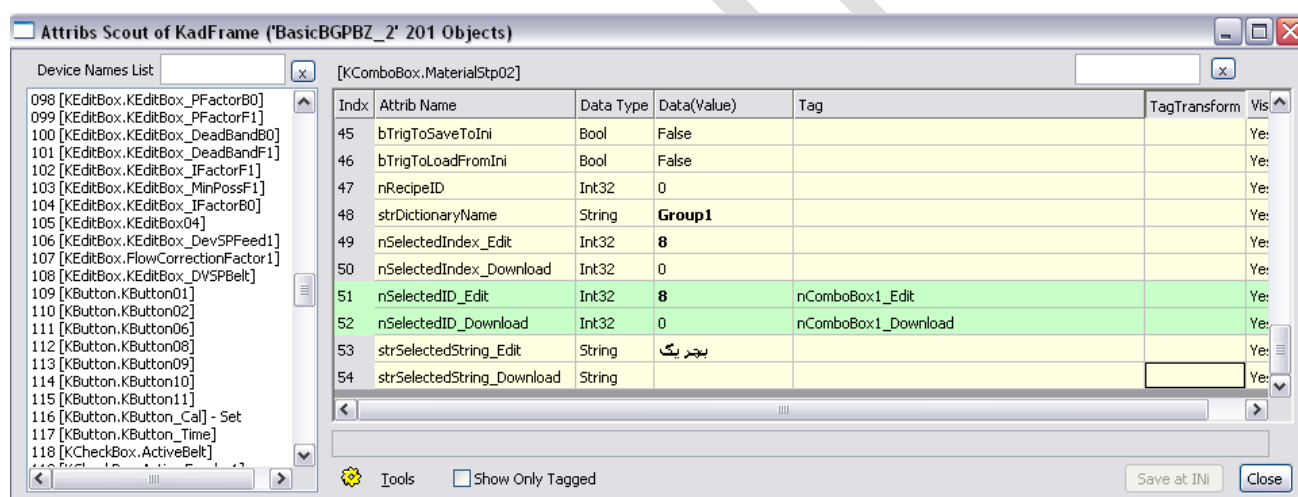
```
[Dictionary.Group1]
0=-----
1=strMaterialName_B1
2=strMaterialName_2_1
3=strMaterialName_2_2
4=strMaterialName_2_3
5=strMaterialName_3
6=strMaterialName_4
7=strMaterialName_5
8=strTag1
9=Confirmation
10=Delay
```

پس از یک بار بازو بسته کردن کادفریم، در کمبو باکس دقیقاً در همان شماره ی ID ای که تگ را در فایل INI مربوط به آن وارد کردید نوشته ی "بچر یک" را مشاهده میکنید، که در واقع الان مقدار بچر یک در کمبو باکس نمایش داده میشود.

مرجع کار با برنامه مانیتورینگ کادفریم

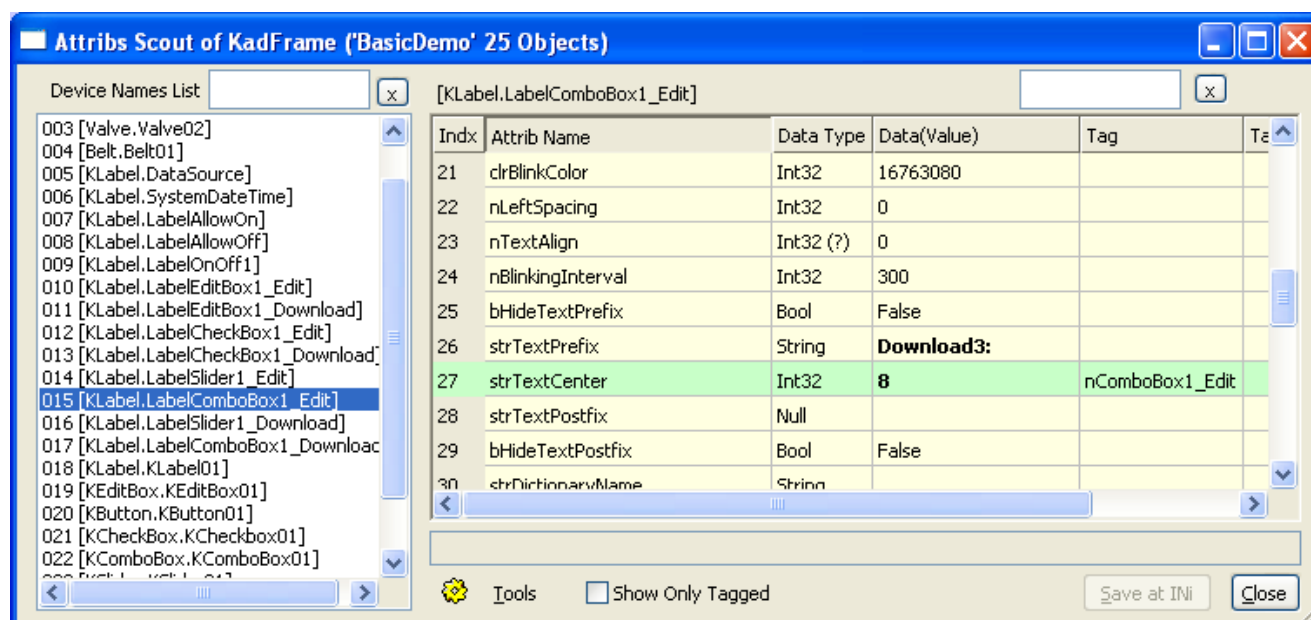


نکته: برای نمایش SelectedID (ID متغیر انتخاب شده از کمبوباکس) در یک لیبل، باید دو تگ محاسباتی مربوط به ادیت و دانلود برای کمبوباکس تعریف کرده و آن ها را به nSelectedID_Edit و nSelectedID_Download اساین کنید، این دو تگ باید از نوع Int تعریف شوند.



شکل ۴۴

همچنین باید تگ مربوط به nSelectedID_Edit را به strTextCenter در اتریب اسکات لیبل مربوطه اساین کرد (شکل ۴۵):



شکل ۴۵

این گونه می‌توانید شماره ID متغیر انتخاب شده در کمبو باکس را در لیبل مربوطه ببینید. همانطور که در شکل زیر مشاهده می‌کنید، در لیبل عدد ۸ که در واقع IDی "بچریک" است نمایش داده می‌شود.



۱۸ تشریح اتریب های کنترل های گرافیکی کادفریم

همانطور که گفتیم هر کنترل گرافیکی برای نگهداری اطلاعات خود از مجموعه ای از اتریب ها استفاده می کند. در واقع تنظیمات طراح از طریق همین اتریب ها به کنترل گرافیکی انتقال داده می شود. از این رو برای اعمال تنظیمات خود در کنترل گرافیکی مدنظر شناخت اتریب های آن کنترل ضروری می باشد.

به طور کلی برای معرفی اتریبهای موجود در کادفریم باید آنها را به دو دسته تقسیم کرد:

دسته اول اتریب هایی هستند که به صورت عمومی می باشند و در همه کنترل ها مشترک هستند. دسته دوم اتریب هایی هستند که برای هر کنترل منحصر به فرد می باشند که هر شی متناسب با نوع عملکردش دارای مجموعه ای از این اتریب ها است. در ادامه ابتدا اتریب های دسته اول را معرفی کرده و سپس همراه با معرفی هر کنترل گرافیکی اتریب های خاص مرتبط با آن ابجکت را معرفی می کنیم.

۱۸,۱ اتریب های عمومی (مشترک)

به طور کلی ۳۵ اتریب اول هر کنترل جز اتریب های عمومی می باشند که در همه کنترل های گرافیکی مشترک هستند. تعریف هر کدام از این اتریب ها در جدول ۲ آمده است.

جدول ۲

شماره	اتریب های مشترک	نوع دیتا	توضیحات
1	strDeviceCaption	String	نام المان کنترل
2	bHideCtrlInView	Bool	با فعال شدن این اتریب المان مربوطه مخفی می شود.
3	nSubShapeType	Int32	تغییر شکل ظاهری مخازن
4	nLoginGroup_ToTrig	Int32 (?)	هر عدد مشخص نشان دهنده ی سطح دسترسی برای کاربر مشخص است.
5	nLoginGroup_ToDownload	Int32 (?)	هر عدد مشخص نشان دهنده ی سطح دسترسی برای کاربر مشخص است.
6	nLoginGroup_ToSave	Int32 (?)	هر عدد مشخص نشان دهنده ی سطح دسترسی برای کاربر مشخص است.
7	nDrawPosX	Int32	موقعیت المان در محور X

مرجع کار با برنامه مانیتورینگ کادفریم

موقعیت المان در محور Y	Int32	nDrawPosY	8
اندازه ی عرض المان	Int32	nDrawSizeX	9
اندازه ی طول المان	Int32	nDrawSizeY	10
تعداد اعشار برای مقدار نمایش عدد اول بر روی مخزن	Int32	nFloatPointPosInDisplay1	11
تعداد اعشار برای مقدار نمایش عدد دوم بر روی مخزن	Int32	nFloatPointPosInDisplay2	12
تغییر فونت	Int32	nFontHeight1	13
تغییر فونت	Int32	nFontHeight2	14
تغییر فونت	Int32	nFontHeight3	15
رنگ پس زمینه	Int32	ColorBG	16
رنگ زمینه	Int32	ColorFG	17
رنگ بدنه	Int32	ColorBody	18
تغییر رنگ زمینه با OverLoad (اضافه بار) المان	Int32	ColorFgOverload	19
تغییر رنگ پس زمینه زمانی که المان در فرمول قرار ندارد.	Int32	ColorBgOutOfRecipe	20
تغییر رنگ المان در زمانی که آن المان در مود محلی قرار دارد.	Int32	ColorBgLocal	21
برای تیرگی و روشنی رنگ پس زمینه	Int32	nBgColorGradientFactor	22
برای تیرگی و روشنی رنگ زمینه	Int32	nFgColorGradientFactor	23
فعال شدن این اتریب نشان دهنده ی OverLoad (اضافه بار) المان است.	Bool	bDeviceIsOverloaded	24
مربوط به Tare کردن مخزن می باشد.	Bool (?)	bNoTaring	25
فعال شدن این اتریب نشان دهنده ی محاسبه ی بار در راه	Bool	bAutoCalcLoadInWay	26

			به صورت اتومات می باشد.
27	fLoadInWay	Float32	نشان دهنده مقدار بار در راه المان است.
28	bIsInRemoteControl	Bool	نشان دهنده ی مود کاری المان است.
29	bDeviceIsInRecipe	Bool	نشان دهنده ی فعال بودن المان در فرمول است.
30	bIsInAutoControl	Bool	فعال بودن این اتریب نشان دهنده ی کنترل اتوماتیک است.
31	bDeviceIsInService	Bool (?)	فعال بودن این اتریب نشان دهنده ی در حال تعمیر بودن المان است.
32	eActuatorControlType	Int32 (?)	مشخص کننده نوع اکچواتور می باشد.
33	eDeviceOperationMode	Int32 (?)	مود عملکرد المان را معین می کند.
34	strDeviceOperationMode	String (?)	متنی که نشان دهنده مود عملکرد المان است.
35	nPercentOfHighLowExcitation	Int32	درصدی که نشان دهنده ی کارکرد با دور تند اکچواتور است.

توضیحاتی درباره بعضی اتریب ها:

وقتی بر روی اتریب های زیر کلیک می کنیم مواردی در قسمت پایین صفحه اتریب اسکات می آید که شامل نکات زیر است:

برای اتریب های ۴ تا ۶ این موارد مربوط به حق دسترسی و انتخاب کاربر با توجه به عددهای اختصاص داده شده به آن هاست:

4) nLoginGroup_ToTrig: Used in click of 'KButton' and 'KCheckBox' and 'TrunOnOff' of all Actuators. 0:None, 1:Operator, 8:Shift, 16:Technical, 32:Production, 64:PlantAdmin, 128:KadAdmin, 121:AllGroups. Set one or summation of multiple values.

5) nLoginGroup_ToDownload: 0:None, 1:Operator, 8:Shift, 16:Technical, 32:Production, 64:PlantAdmin, 128:KadAdmin, 121:AllGroups. Set one or summation of multiple values.

6) nLoginGroup_ToSave: 0:None, 1:Operator, 8:Shift, 16:Technical, 32:Production, 64:PlantAdmin, 128:KadAdmin, 121:AllGroups. Set one or summation of multiple values.

انتخاب هریک از اعداد اشاره شده در اتریب ۳۲، نوع کنترل اکچواتور را مشخص می کند:

32) eActuatorControlType: 0:Single State, 1:Low Excitation, 2:High Excitation, 3:High-Low Excitation, 4:Increase-Decrease Analog, 5:Increase-Decrease Digital

موارد نوشته شده در جلو اتریب ۳۳ مود عملکرد المان کنترلی را انتخاب می کند:

33) eDeviceOperationMode: 0:Unknown, 1:Remote Auto, 2:Remote Manual, 3:Local, 4:Out of Service

Belt ۱۸,۲



مشخصات نوار همگی در اتریب های جدول ۳ تعریف می شوند.

جدول ۳

شماره	اتریم های Belt	نوع دیتا	توضیح
36	bHasOnOffCommand	Bool (?)	برای فعال و غیرفعال کردن قابلیت روشن و خاموش شدن به صورت دستی
37	bShowOnOffMenu	Bool (?)	برای فعال و غیرفعال کردن منوی روشن و خاموش شدن به صورت دستی
38	bTurnOn	Bool (?)	فرمان روشن شدن المان
39	bIsOn	Bool (?)	فیدبک به منظور روشن بودن المان
40	bInterlockTurnOn	Bool	اینترلاک برای فرمان روشن شدن
41	bInterlockTurnOff	Bool	اینترلاک برای فرمان خاموش شدن
42	strTextBoolTagTrue	String	نمایش متن در هنگام فعال شدن المان
43	strTextBoolTagFalse	String	نمایش متن در هنگام غیرفعال شدن المان
44	strNoInterlockTurnOn	String	نمایش متن برای زمانی که اینترلاک روشن شدن المان برقرار نیست.
45	strNoInterlockTurnOff	String	نمایش متن برای زمانی که اینترلاک خاموش شدن المان برقرار نیست.
46	fManualExcitation	Float32	تحریک دستی
47	Excitation_Phy	Null	تحریک جاری المان
48	Excitation_ZeroPhy	Null	مقدار Zero فیزیکی
49	Excitation_SpanPhy	Float32	مقدار Span فیزیکی
50	Excitation_LowLevel	Null	حداقل تحریک فیزیکی
51	Excitation_HighLevel	Float32	حداکثر تحریک فیزیکی
52	Excitation_Port	Null	عدد الکتریکی جاری (که از PLC خوانده می شود).

مرجع کار با برنامه مانیتورینگ کادفریم

53	Excitation_ZeroPort	Float32	مقدار Zero پورت
54	Excitation_SpanPort	Float32	مقدار Span پورت
55	Excitation_Caption	String	نام در نظر گرفته شده برای تحریک در جدول Properties
56	Excitation_Unit	String	واحد تحریک که به صورت % بیان می شود.
57	Excitation_nAutoCalcPhyPortValue	Int32 (?)	برای محاسبه ی عدد پورت به فیزیکال و یا فیزیکال به پورت به صورت اتومات
58	Flow_Phy	Float32	فلوی جاری المان
59	Flow_ZeroPhy	Null	مقدار Zero فیزیکال
60	Flow_SpanPhy	Float32	مقدار Span فیزیکال
61	Flow_LowLevel	Null	حداقل فلو فیزیکال
62	Flow_HighLevel	Float32	حداکثر فلو فیزیکال
63	Flow_Port	Null	عدد الکتریکی جاری (که از PLC خوانده می شود).
64	Flow_ZeroPort	Float32	مقدار Zero پورت
65	Flow_SpanPort	Float32	مقدار Span پورت
66	Flow_Caption	String	نام در نظر گرفته شده برای فلو در جدول Properties
67	Flow_Unit	String	واحد فلو که به صورت Kg/h بیان می شود.
68	Flow_nAutoCalcPhyPortValue	Int32 (?)	برای محاسبه ی عدد پورت به فیزیکال و یا فیزیکال به پورت به صورت اتومات
69	fIncDecStepPercent	Float32	ست پوینت به منظور افزایش یا کاهش تحریک با فعال شدن اتریب های ۷۷ و ۷۸
70	fExcitationHighPercent	Float32	درصد ماکزیمم تحریک
71	fExcitationLowPercent	Float32	درصد مینیمم تحریک

مرجع کار با برنامه مانیتورینگ کادفریم

72	nOnTimeAppRecordID	Int32 (?)	ID که به محض روشن شدن المان ایجاد می شود (جهت استفاده در بانک اطلاعاتی).
73	bRunHigh	Bool	فرمان روشن شدن با دور تند
74	bIsOnHigh	Bool	فرمان فیدبک با دور تند
75	bRunLow	Bool	فرمان روشن شدن با دور کند
76	bIsOnLow	Bool	فرمان فیدبک با دور کند
77	bIncreaseCmd	Bool	دستور افزایش تحریک به اندازه ست پوینت مشخص شده در اتریب ۶۹
78	bDecreaseCmd	Bool	دستور کاهش تحریک به اندازه ست پوینت مشخص شده در اتریب ۶۹
79	bHasAlarm	Bool	آلارم داشتن المان
80	fCalculatedAnalogFactorInDemo	Float32 (?)	محاسبه ی آنالوگ ورودی با یکی از فرمول های تعریف شده در مود دمو که بستگی به اتریب ۸۵ دارد.
81	fCalculatedAnalogFactorInReal	Float32 (?)	محاسبه ی آنالوگ ورودی با یکی از فرمول های تعریف شده در مود Real که بستگی به اتریب ۸۶ دارد.
82	bDirectionIsToRight	Bool	قابلیت چرخش در دو جهت ساعتگرد و پادساعتگرد، با True یا False کردن این اتریب امکانپذیر است.
83	nSourceOfText1	Int32 (?)	نمایش مقدار و واحد آنالوگ ورودی انتخاب شده
84	nBladesCount	Int32	تعداد تیغه های نمایش داده شده در شماتیک نوار
85	nCalculatedAnalogInDemo	Int32 (?)	انتخاب نوع کمیت برای محاسبه ی آنالوگ ورودی در مود دمو

انتخاب نوع کمیت برای محاسبه ی آنالوگ ورودی در مود Real	Int32 (?)	nCalculatedAnalogInReal	86
سرعت جاری المان	Float32	Speed_Phy	87
مقدار Zero فیزیکی	Null	Speed_ZeroPhy	88
مقدار Span فیزیکی	Float32	Speed_SpanPhy	89
حداقل سرعت فیزیکی	Null	Speed_LowLevel	90
حداکثر سرعت فیزیکی	Float32	Speed_HighLevel	91
عدد الکتریکی جاری (که از PLC خوانده می شود).	Null	Speed_Port	92
مقدار Zero پورت	Float32	Speed_ZeroPort	93
مقدار Span پورت	Float32	Speed_SpanPort	94
نام در نظر گرفته شده برای سرعت در جدول Properties	String	Speed_Caption	95
واحد سرعت که به صورت m/h بیان می شود.	String	Speed_Unit	96
برای محاسبه ی عدد پورت به فیزیکی و یا فیزیکی به پورت به صورت اتومات	Int32 (?)	Speed_nAutoCalcPhyPortValue	97
وزن جاری المان	Float32	Load_Phy	98
مقدار Zero فیزیکی	Null	Load_ZeroPhy	99
مقدار Span فیزیکی	Float32	Load_SpanPhy	100
حداقل وزن فیزیکی	Null	Load_LowLevel	101
حداکثر وزن فیزیکی	Float32	Load_HighLevel	102
عدد الکتریکی جاری (که از PLC خوانده می شود).	Null	Load_Port	103
مقدار Zero پورت	Float32	Load_ZeroPort	104
مقدار Span پورت	Float32	Load_SpanPort	105

106	Load_Caption	String	نام در نظر گرفته شده برای وزن در جدول Properties
107	Load_Unit	String	واحد وزن که به صورت Kg/m بیان می شود.
108	Load_nAutoCalcPhyPortValue	Int32 (?)	برای محاسبه ی عدد پورت به فیزیکال و یا فیزیکال به پورت به صورت اتومات

توضیحات داده شده برای بعضی اتریب ها:

در اتریب های ۵۷، ۶۸، ۹۷ و ۱۰۸ به ترتیب برای تحریک، فلو، سرعت و وزن، توضیح زیر آمده است. زمانی که می خواهیم به آنالوگ خروجی مقدار دهیم، ۱- و هنگامی که به آنالوگ ورودی مقدار داده می شود، ۱+ انتخاب می شود. این در حالی است که انتخاب صفر به معنی محاسبه در برنامه PLC است.

۵۷) Excitation_nAutoCalcPhyPortValue: Auto Calc => -۱:Phy to Port, ۰:No Calc, +۱:Port to Phy

در حالت دمو یا Real از طریق فرمول های زیر مقدار آنالوگ ورودی محاسبه می شود:

80) fCalculatedAnalogFactorInDemo: $fFlow = fFactor * (fSpeed * fLoad)$;

$fSpeed = fFactor * (fFlow / fLoad)$; $fLoad = fCFactor * (fFlow / fSpeed)$; Formula is depend to Attrib <85: nCalculatedAnalogInDemo>.

81) fCalculatedAnalogFactorInReal: $fFlow = fFactor * (fSpeed * fLoad)$;

$fSpeed = fFactor * (fFlow / fLoad)$; $fLoad = fCFactor * (fFlow / fSpeed)$; Formula is depend to Attrib <86: nCalculatedAnalogInReal>.

هریک از کمیت های زیر می تواند در هنگام روشن بودن نوار بر روی آن نمایش داده شود:

83) nSourceOfText1: -1:NoSource, 0:Excitation, 1:Flow, 2:Speed, 3:Load

هر یک از کمیت های زیر می توانند با قرار گرفتن در فرمول اتریب های ۸۰ و ۸۱ محاسبه شوند:

85) nCalculatedAnalogInDemo: 0:Flow, 1:Speed, 2:Load, 3:None. See also Attrib <80:

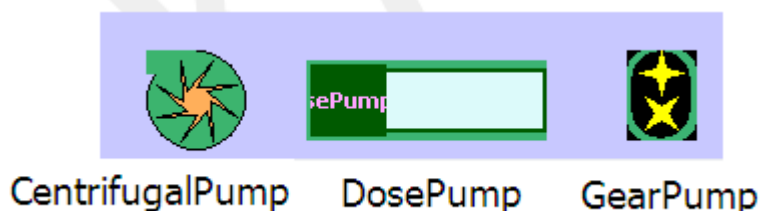
fCalculatedAnalogFactorInDemo>.

86) nCalculatedAnalogInReal: 0:Flow, 1:Speed, 2:Load, 3:None. See also Attrib <81:

fCalculatedAnalogFactorInReal>.

Pump ۱۸,۳

سه نوع پمپ در دیوایس های انتخابی کادفریم وجود دارد: DosePump, CentrifugalPump, GearPump



دو پمپ دنده ای و سانتریفیوژ دارای ۷۹ اتریب می باشند که این اتریب ها با ۷۹ اتریب اولی نوار مشابه می باشند (به جدول ۳ رجوع شود). Dose Pump نیز مشابه دو پمپ دیگر است و تنها ۲ اتریب جدول ۴ به آن اضافه شده است:

جدول ۴

شماره	اتریب های اضافه Dose Pump	نوع دیتا	توضیح
80	fInputPressure	Float32	
81	nMaxStrokePercent	Int32	

Valve ۱۸,۴

دو نوع شیر در کادفریم وجود دارد: DBF Valve و Valve



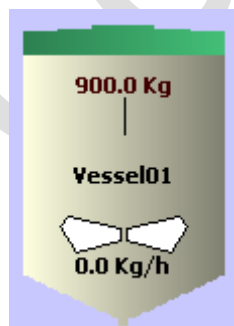
مرجع کار با برنامه مانیتورینگ کادفریم

اترید های در نظر گرفته شده برای این دو المان تا اترید ۷۹ با اترید های نوار یکسان می باشد (به جدول ۳ رجوع شود). Valve تنها یک اترید بیشتر به نام bIsNormalyOpen دارد که با فعال شدن این اترید شیر در حالت نرمال باز است و با دادن فرمان TurnOn شیر بسته می شود. شیر DBF Valve نیز ۴ اترید زیر را اضافه دارد:

جدول ۵

شماره	اترید های اضافه DBF Valve	نوع دیتا	توضیح
80	bMinFlgPosition	Bool	
81	bMaxFlgPosition	Bool	
82	bLlaFlgPosition	Bool	
83	bHlaFlgPosition	Bool	

۱۸,۵ Vessel



مشخصات مخزن همگی در اترید های زیر تعریف می شوند:

جدول ۶

شماره	اترید های Vessel	نوع دیتا	توضیح
36	nBlade1LocY	Int32	موقعیت همزن شماره ۱ نمایش داده شده روی مخزن در محور عمودی

مرجع کار با برنامه مانیتورینگ کادفریم

موقعیت همزن شماره ۲ نمایش داده شده روی مخزن در محور عمودی	Int32	nBlade2LocY	37
موقعیت متن شماره ۱ نمایش داده شده روی مخزن در محور عمودی	Int32	nText1LocY	38
موقعیت متن شماره ۲ نمایش داده شده روی مخزن در محور عمودی	Int32	nText2LocY	39
موقعیت متن شماره ۳ نمایش داده شده روی مخزن در محور عمودی	Int32	nText3LocY	40
انتخاب کمیت اول مورد نمایش روی مخزن	Int32 (?)	nSourceOfText1	41
انتخاب کمیت سوم مورد نمایش روی مخزن	Int32 (?)	nSourceOfText3	42
تغییر شکل سر بالایی مخزن	Int32	nShapeTopHead	43
تغییر شکل لوله بالایی مخزن	Int32	nShapeTopPipe	44
تغییر شکل شیب بالای مخزن	Int32	nShapeTopSlope	45
تغییر شکل بدنه مخزن	Int32	nVerticalBody	46
تغییر شکل شیب پایینی مخزن	Int32	nShapeBottomSlope	47
تغییر شکل لوله پایین مخزن	Int32	nShapeBottomPipe	48
تغییر شکل سر پایینی مخزن	Int32	nShapeBottomHead	49
مقدار درصدی که با رسیدن وزن مخزن به آن عمل شارژ مخزن صورت می گیرد.	Float32	fAutoChargeStartPercent	50
مقدار درصدی که با رسیدن وزن مخزن به آن عمل شارژ مخزن متوقف می شود.	Float32	fAutoChargeStopPercent	51
در صورت رسیدن وزن مخزن به این مقدار عمل تخلیه مخزن متوقف می شود.	Float32	fTareLevel	52
	Float32 (?)	fOverloadWarningLevel	53
	Float32 (?)	fOverloadErrorLevel	54
	Bool	bNextEligibleFilling	55

مرجع کار با برنامه مانیتورینگ کادفریم

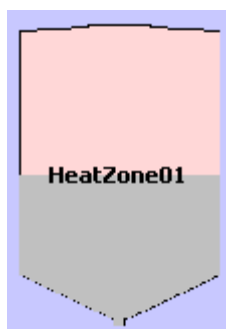
می توان به عنوان آلام مخزن استفاده کرد.	Bool	bIsFilling	56
نمایش یا عدم نمایش روشن شدن المان از منو Properties	Bool	bShowOnOffMenu	57
فعال یا غیرفعال شدن قابلیت روشن و خاموش شدن همزمان دو همزن	Bool	bHasTwoOnOff	58
فرمان روشن شدن همزن شماره ۱	Bool	bTurnOn1	59
فیدبک به منظور روشن بودن المان همزن شماره ۱	Bool	bIsOn1	60
فرمان روشن شدن همزن شماره ۲	Bool	bTurnOn2	61
فیدبک به منظور روشن بودن المان همزن شماره ۲	Bool	bIsOn2	62
فرمان روشن شدن با دور تند	Bool	bRunHigh	63
فرمان روشن شدن با دور کند	Bool	bRunLow	64
دستور افزایش تحریک به اندازه ست پوینت مشخص شده	Bool	bIncreaseCmd	65
دستور کاهش تحریک به اندازه ست پوینت مشخص شده	Bool	bDecreaseCmd	66
آلام داشتن المان	Bool	bHasAlarm	67
	Bool (?)	bInternal_LLA	68
	Bool (?)	bInternal_HLA	69
	Float32 (?)	fAlarmHysteresis_Percent	70
وزن جاری المان	Int32	Level_Phy	71
مقدار Zero فیزیکی	Null	Level_ZeroPhy	72
مقدار Span فیزیکی	Float32	Level_SpanPhy	73
حداقل وزن فیزیکی	Float32	Level_LowLevel	74
حداکثر وزن فیزیکی	Float32	Level_HighLevel	75
عدد الکتریکی جاری (که از PLC خوانده می شود).	Null	Level_Port	76
مقدار Zero پورت	Float32	Level_ZeroPort	77

مرجع کار با برنامه مانیتورینگ کادفریم

78	Level_SpanPort	Float32	مقدار Span پورت
79	Level_Caption	String	نام در نظر گرفته شده برای وزن در جدول Properties
80	Level_Unit	String	واحد وزن که به صورت Kg بیان می شود.
81	Level_nAutoCalcPhyPortValue	Int32 (?)	برای محاسبه ی عدد پورت به فیزیکیال و یا فیزیکیال به پورت به صورت اتومات
82	Flow_Phy	Null	فلوی جاری المان
83	Flow_ZeroPhy	Null	مقدار Zero فیزیکیال
84	Flow_SpanPhy	Float32	مقدار Span فیزیکیال
85	Flow_LowLevel	Float32	حداقل فلو فیزیکیال
86	Flow_HighLevel	Float32	حداکثر فلو فیزیکیال
87	Flow_Port	Null	عدد الکتریکیال جاری (که از PLC خوانده می شود).
88	Flow_ZeroPort	Float32	مقدار Zero پورت
89	Flow_SpanPort	Float32	مقدار Span پورت
90	Flow_Caption	String	نام در نظر گرفته شده برای فلو در جدول Properties
91	Flow_Unit	String	واحد فلو که به صورت Kg/h بیان می شود.
92	Flow_nAutoCalcPhyPortValue	Int32 (?)	برای محاسبه ی عدد پورت به فیزیکیال و یا فیزیکیال به پورت به صورت اتومات
93	Speed_Phy	Float32	سرعت جاری المان
94	Speed_ZeroPhy	Float32	مقدار Zero فیزیکیال
95	Speed_SpanPhy	Float32	مقدار Span فیزیکیال
96	Speed_LowLevel	Float32	حداقل سرعت فیزیکیال
97	Speed_HighLevel	Float32	حداکثر سرعت فیزیکیال
98	Speed_Port	Null	عدد الکتریکیال جاری (که از PLC خوانده می شود).

مرجع کار با برنامه مانیتورینگ کادفریم

99	Speed_ZeroPort	Float32	مقدار Zero پورت
100	Speed_SpanPort	Float32	مقدار Span پورت
101	Speed_Caption	String	نام در نظر گرفته شده برای سرعت در جدول Properties
102	Speed_Unit	String	واحد سرعت که به صورت RPM بیان می شود.
103	Speed_nAutoCalcPhyPortValue	Int32 (?)	برای محاسبه ی عدد پورت به فیزیکیال و یا فیزیکیال به پورت به صورت اتومات
104	Temperature_Phy	Float32	دمای جاری المان
105	Temperature_ZeroPhy	Null	مقدار Zero فیزیکیال
106	Temperature_SpanPhy	Float32	مقدار Span فیزیکیال
107	Temperature_LowLevel	Null	حداقل دمای فیزیکیال
108	Temperature_HighLevel	Float32	حداکثر دمای فیزیکیال
109	Temperature_Port	Null	عدد الکتریکیال جاری (که از PLC خوانده می شود).
110	Temperature_ZeroPort	Float32	مقدار Zero پورت
111	Temperature_SpanPort	Float32	مقدار Span پورت
112	Temperature_Caption	String	نام در نظر گرفته شده برای دما در جدول Properties
113	Temperature_Unit	String	واحد دما که به صورت °C بیان می شود.
114	Temperature_nAutoCalcPhyPortValue	Int32 (?)	برای محاسبه ی عدد پورت به فیزیکیال و یا فیزیکیال به پورت به صورت اتومات



مشخصات HeatZone همگی در اتریب های زیر تعریف می شوند:

جدول ۷

شماره	اتریب های HeatZone	نوع دیتا	توضیحات
36	bDO_INCREASE_TEMPERATURE	Bool	
37	bDO_DECREASE_TEMPERATURE	Bool	
38	bDO_ON_OFF_TEMPERATURE	Bool	
39	bDI_ON_OFF_TEMPERATURE	Bool	
40	bDI_ALARM_FLAG_TEMPERATURE	Bool	
41	nShapeTopHead	Int32	تغییر شکل سر بالایی المان
42	nShapeTopPipe	Int32	تغییر شکل لوله بالایی المان
43	nShapeTopSlope	Int32	تغییر شکل شیب بالای المان
44	nVerticalBody	Int32	تغییر شکل بدنه المان
45	nShapeBottomSlope	Int32	تغییر شکل شیب پایینی المان
46	nShapeBottomPipe	Int32	تغییر شکل لوله پایین المان
47	nShapeBottomHead	Int32	تغییر شکل سر پایینی المان
48	Temperature_Phy	Null	دمای جاری المان
49	Temperature_ZeroPhy	Null	مقدار Zero فیزیکی

50	Temperature_SpanPhy	Float32	مقدار Span فیزیکی
51	Temperature_LowLevel	Null	حداقل دمای فیزیکی
52	Temperature_HighLevel	Float32	حداکثر دمای فیزیکی
53	Temperature_Port	Null	عدد الکتریکال جاری (که از PLC خوانده می شود.)
54	Temperature_ZeroPort	Float32	مقدار Zero پورت
55	Temperature_SpanPort	Float32	مقدار Span پورت
56	Temperature_Caption	String	نام در نظر گرفته شده برای دما در جدول Properties
57	Temperature_Unit	String	واحد دما که به صورت °C بیان می شود.
58	Temperature_nAutoCalcPhyPortValue	Int32 (?)	برای محاسبه ی عدد پورت به فیزیکی و یا فیزیکی به پورت به صورت اتومات

۱۸,۷ Helix



اتریم های در نظر گرفته برای این المان کاملاً مشابه ۷۹ اتریم اول نوار است (به جدول ۳ رجوع شود).

۱۸,۸ KLabel و KButton

در ابتدا اتریم های مربوط به هریک از این دو المان تشریح و در ادامه با یک مثال، قابلیت این دو المان مفصل تر توضیح داده خواهد شد.

جدول ۸

شماره	اتریم های KButton	نوع دیتا	توضیحات
36	bInterlock_Edit	Bool	اساین تگ اینترلاک به این اتریم
37	strInterlock_Edit	String	راهنمای متنی نوشته شده برای اینترلاک
38	bEnable_Edit	Bool	فعال کردن المان
39	strInsufficientAccessMsg	String	
40	strLabel	String	متن نوشته شده روی باتن در حالت Demo
41	strLabel_WhenTrigIsTrue	String (?)	متن نوشته شده روی باتن در حالتی که تگ باتن True شده است.
42	strLabel_WhenTrigIsFalse	String (?)	متن نوشته شده روی باتن در حالتی که تگ باتن False شده است.
43	bButtonTrigger	Bool	اتریم مخصوص روشن یا خاموش شدن باتن
44	eButtonType	Int32 (?)	* همانگونه که در توضیح پایین جدول آمده است در این قسمت می توان نوع عملکرد باتن را مشخص کرد.
45	nActionDelaySetpoint	Int32	
46	nActionDelayCounter	Int32	
47	strGoToPage_ByName	String (?)	انتقال به صفحه ی دیگر با نوشتن نام صفحه ی مورد نظر در این قسمت
48	nGoToPage_ByID	Int32 (?)	انتقال به صفحه ی دیگر با نوشتن شماره آی دی صفحه ی مورد نظر در این قسمت
49	strTrigConfirmationMessage	String	پیغام تایید در هنگام فشردن شدن باتن
50	strTrigBalloonMessage	String	پیغام نشان داده شده در یک بالن در هنگام تریگ شدن تگ باتن

* 44) eButtonType: 0:SetToggle, 1:SetOn, 2:SetOff, 3:SetOnDelayOff, 4:SetOffDelayOn, 5:GoToPageX

لیبل تنها المانی است که در ۳۵ اتریب اول (اتریب های عمومی) با بقیه مشترک نیست و تنها ۱۵ اتریب اول آن مشابه بقیه است:

جدول ۹

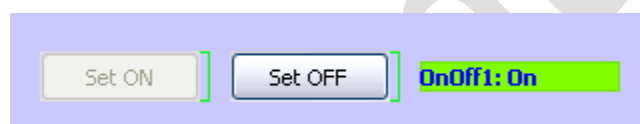
شماره	اتریب های KLabel	نوع دیتا	توضیحات
16	bHasOnOffCommand	Bool (?)	برای فعال و غیرفعال کردن قابلیت روشن و خاموش شدن به صورت دستی
17	bShowOnOffMenu	Bool (?)	برای فعال و غیرفعال کردن منوی روشن و خاموش شدن به صورت دستی
18	bTurnOn	Bool (?)	فرمان روشن شدن المان
19	bIsOn	Bool (?)	فیدبک به منظور روشن بودن المان
20	bInterlockTurnOn	Bool	اینترلاک برای فرمان روشن شدن
21	bInterlockTurnOff	Bool	اینترلاک برای فرمان خاموش شدن
22	strTextBoolTagTrue	String	نمایش متن در هنگام فعال شدن المان
23	strTextBoolTagFalse	String	نمایش متن در هنگام غیرفعال شدن المان
24	strNoInterlockTurnOn	String	نمایش متن برای زمانی که اینترلاک روشن شدن المان برقرار نیست.
25	strNoInterlockTurnOff	String	نمایش متن برای زمانی که اینترلاک خاموش شدن المان برقرار نیست.
26	bIsGradient	Bool	تلفیق دو رنگ انتخاب شده در ClrBack2 و ClrBack1
27	bIsTransparent	Bool	انتخاب پس زمینه شفاف برای متن
28	bVerticalGradient	Bool	با فعال شدن این اتریب تلفیق رنگ ها به صورت افقی انجام می شود.
29	bBlinkingIsActive	Bool	قابلیت چشمک زدن المان
30	clrText	Int32	رنگ متن
31	clrTextWhenBoolTagIsOff	Int32	رنگ متن نمایش داده شده، زمانی که تگ Off است.

مرجع کار با برنامه مانیتورینگ کادفریم

32	clrTextWhenBoolTagIsOn	Int32	رنگ متن نمایش داده شده، زمانی که تگ On است.
33	clrBack1	Int32	رنگ پس زمینه زمانی که تگ فعال است.
34	clrBack2	Int32	رنگ پس زمینه زمانی که تگ غیرفعال است.
35	clrBlinkColor	Int32	در زمان فعال بودن قابلیت چشمک زدن، رنگ متن در حالت چشمک زن است.
36	nLeftSpacing	Int32	فاصله از سمت چپ برای قرارگیری متن در مکان مشخص شده
37	nTextAlign	Int32 (?)	جهت قرار گرفتن متن (چپ، وسط یا راست) nTextAlign: 0:Left, 1:Center, 2:Right
38	nBlinkingInterval	Int32	فاصله زمانی هر چشمک
39	bHideTextPrefix	Bool	پنهان کردن متن پیشوند
40	strTextPrefix	Null	متن نوشته شده قبل از متن اصلی یا خروجی نمایش داده شده
41	strTextCenter	Date	متن اصلی یا خروجی نمایش داده شده (با اعمال تگ)
42	strTextPostfix	Null	متن نوشته شده بعد از متن اصلی یا خروجی نمایش داده شده
43	bHideTextPostfix	Bool	پنهان کردن متن پسوند
44	strDictionaryName	String	نام دیکشنری مورد نظر (این قسمت در بخش ۱۷ به طور کامل توضیح داده شده است).
45	bHasAnalogGeneral	Bool	
46	Analog1_Phy	Null	
47	Analog1_ZeroPhy	Null	
48	Analog1_SpanPhy	Float32	
49	Analog1_LowLevel	Null	
50	Analog1_HighLevel	Float32	
51	Analog1_Port	Null	

	Float32	Analog1_ZeroPort	52
	Float32	Analog1_SpanPort	53
	String	Analog1_Caption	54
	String	Analog1_Unit	55
توضیح این اتریب در ادامه آمده است.	Int32 (?)	Analog1_nAutoCalcPhyPortValue	56
56) Analog1_nAutoCalcPhyPortValue: Auto Calc => -1:Phy to Port, 0:No Calc, +1:Port to Phy ®			

با یک مثال کاربردی این دو المان توضیح داده می شوند. شکل زیر را در نظر داشته باشید:



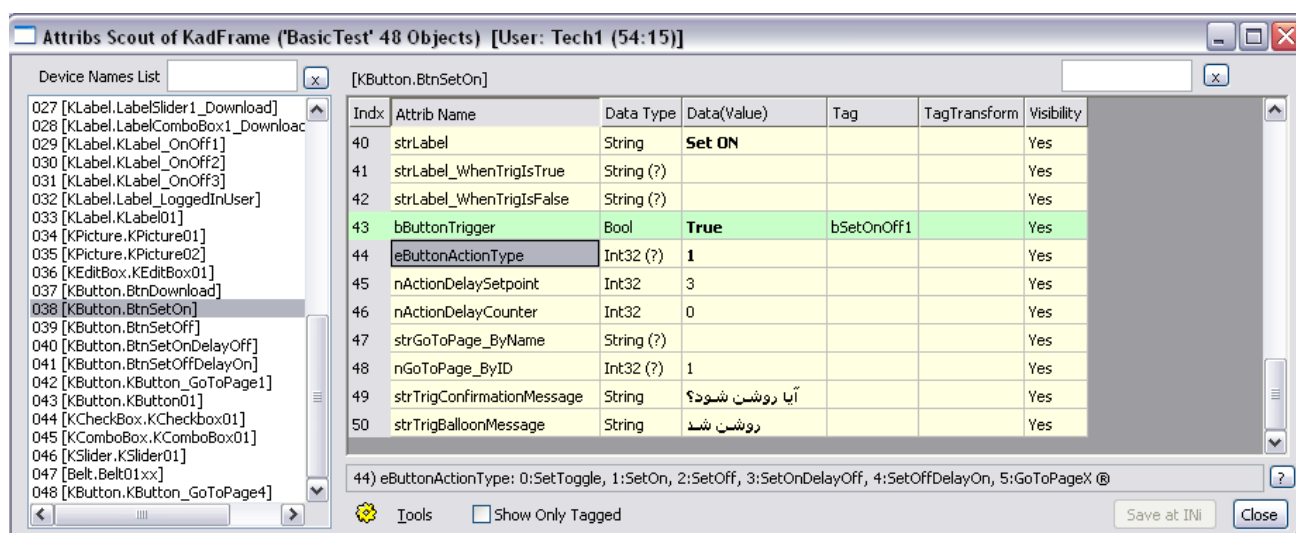
شکل ۴۶

در شکل بالا دو باتن و یک لیبل مشاهده میکنید. در اتریب اسکات مربوط به باتن اتریبی به نام eButtonType داریم. همان گونه که اشاره شد این اتریب نوع اکشن کنترل را تعیین میکند. در شکل زیر انواع اکشنهای این کنترل بر اساس شماره نمایش داده شده است:

44) eButtonType: 0:SetToggle, 1:SetOn, 2:SetOff, 3:SetOnDelayOff, 4:SetOffDelayOn, 5:GoToPageX ®

در اینجا اکشن شماره ۱ را انتخاب کردیم. وقتی این باتن را میزنیم مقدار True آن در تگ bSetOnOff1 نوشته میشود. این تگی دلخواه است که در فایل DCT تعریف کردیم. این تگ در اتریب bButtonTrigger مربوط به کنترل باتن اساین شده است (شکل ۴۷):

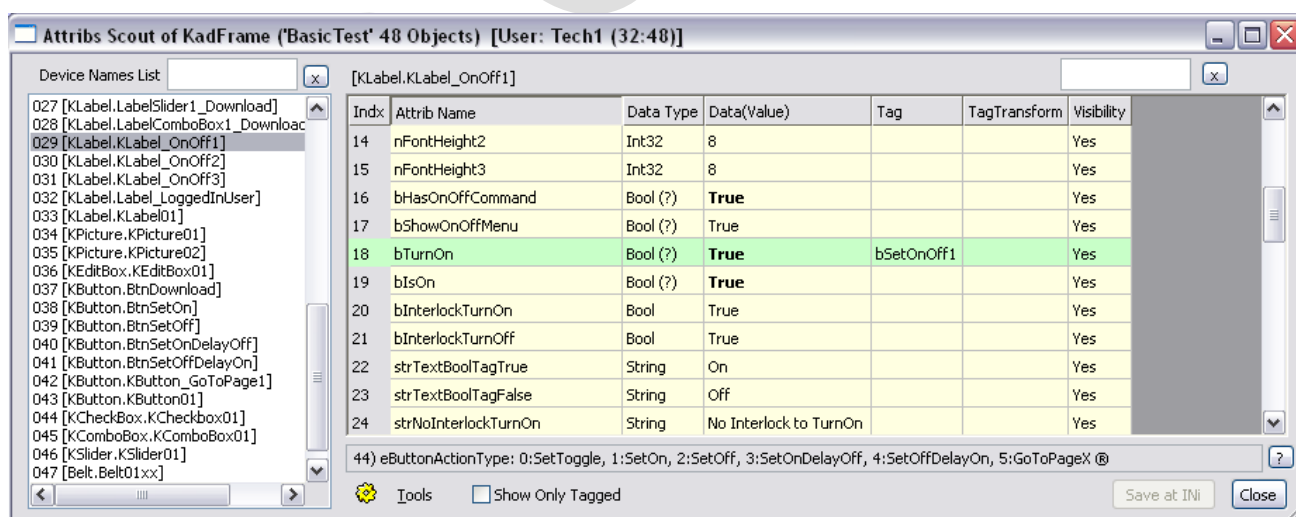
مرجع کار با برنامه مانیتورینگ کادفریم



شکل ۴۷

در شکل ۴۶ باتن دوم به نام Set Off است؛ که تنظیمات آن همانند باتن Set On می باشد. فرق آن اینست که اکشن تایپ ما ۲ یعنی set Off است. توجه کنید که در اتریب strLabel هر چه بنویسیم روی باتن نمایش داده میشود.

برای ادامه توضیحات و بخصوص در مورد کنترل لیبل ابتدا شکل زیر را در نظر داشته باشید:



شکل ۴۸

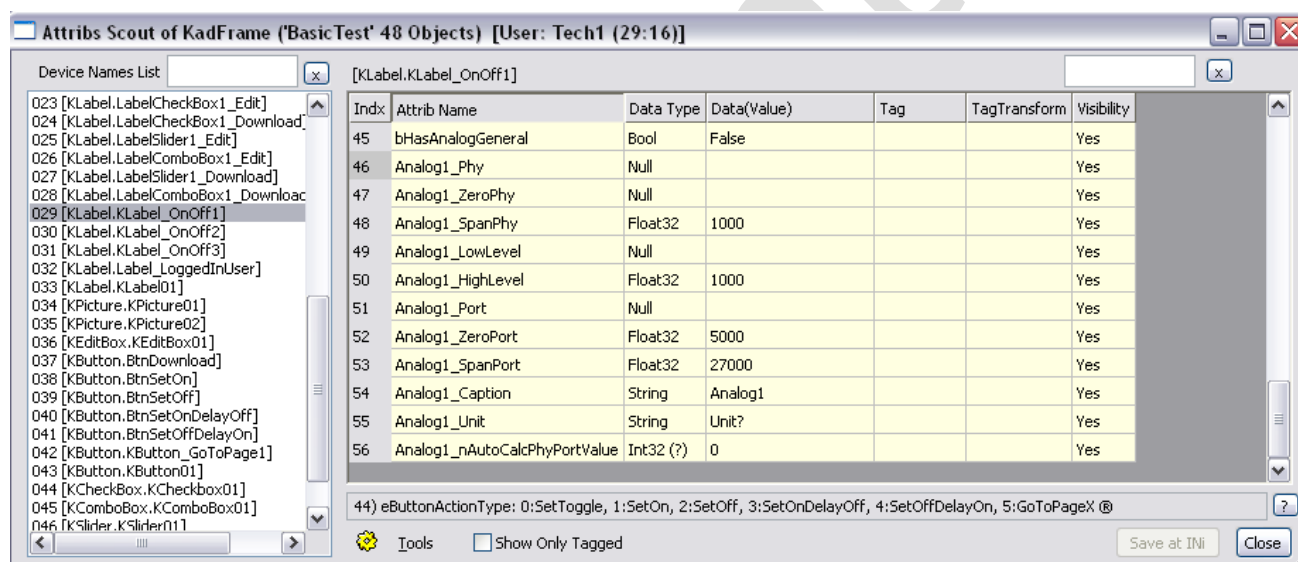
مرجع کار با برنامه مانیتورینگ کادفریم

شکل ۴۸ اتریبهای مربوط به KLabel را نمایش میدهد. در اتریب اسکات، در اتریب strTextBoolTagTrue مقداری که قرار است در حالت True نمایش داده شود، قرار دارد. در اتریب strTextBoolTagFalse مقداری که قرار است در حالت False نمایش داده شود قرار میدهیم، که در اینجا ما تعریف کردیم مقدار استرینگ Off را نمایش دهد.

توجه: منظور ما از حالت True یا False در این مثال کاربردی این است که آیا اتریب bTurnOn این کنترل False است یا True.

True یا False بودن این لیبل در اتریب bTurnOn تعیین میشود.

نکته مهم: کنترل لیبل دارای امکانات آنالوگ نیز میباشد که اتریبهای مربوط به آن در شکل زیر نمایش داده شده است:



شکل ۴۹

در اینجا اتریب مهم اتریب Analog1_nAutoCalcPhyPortValue است که نوع تبدیل انجام شده روی این سیگنال آنالوگ را تعیین میکند. اگر روی این اتریب کلیک کنید در بخش پائینی همین صفحه اتریب، توضیحاتی به صورت زیر نمایش داده میشود که بر اساس آن میتوان عمل انجام شده روی آن را تعیین کرد:

56) Analog1_nAutoCalcPhyPortValue: Auto Calc => -1:Phy to Port, 0:No Calc, +1:Port to Phy ®

مرجع کار با برنامه مانیتورینگ کادفریم

توجه کنید که این امکانات زمانی کاربرد مهم پیدا میکند که ما پی ال سی نداریم و میخواهیم یک سری عملیات را روی سیگنال آنالوگ انجام داد و روی کادفریم نمایش داد.

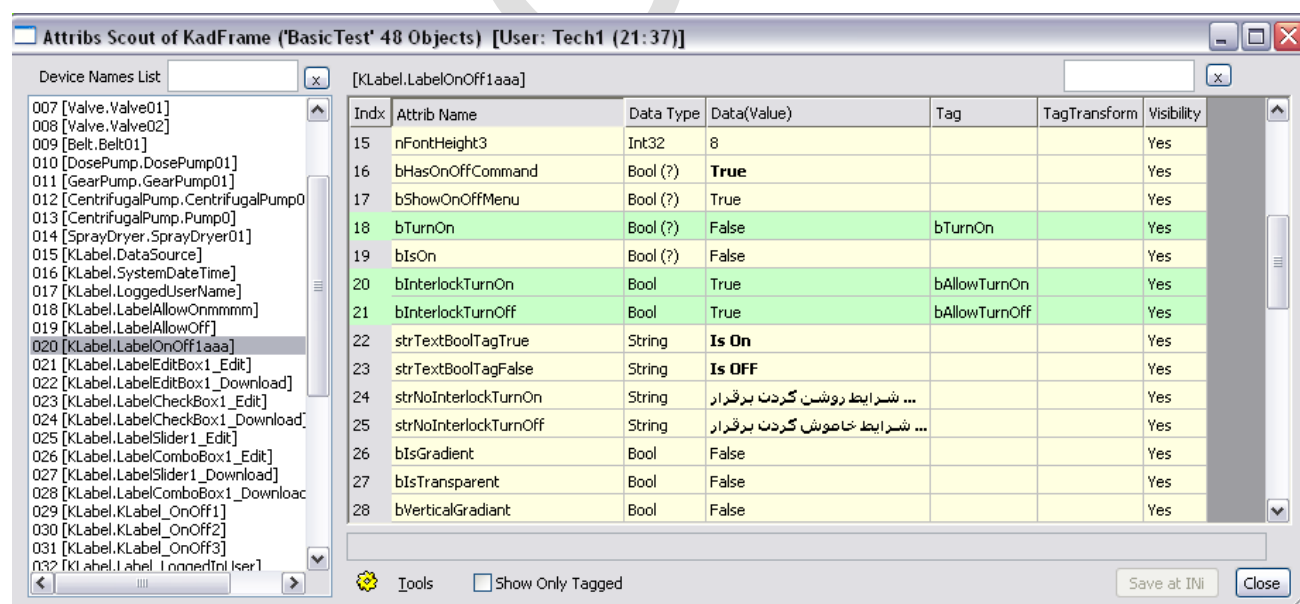
قابلیت دیگر کنترل KLabel اتریبهای strNoInterlockTurnOff و strNoInterlockTurnOn است. این دو اتریب در واقع یک اینترلاک را برای TurnOn یا TurOff تعریف میکند.

در مورد تعریف و کاربرد اینترلاک برای کنترل لیبل مثال زیر را بیان میکنیم. شکل ۵۰ را در نظر داشته باشید:



شکل ۵۰

در این مثال دو لیبل پایینی با لیبل بالایی اینترلاک شده است. شکل زیر اتریبهای لیبل بالایی را نمایش میدهد:



شکل 51

مرجع کار با برنامه مانیتورینگ کادفریم

به اتریب bInterlockTurnOn تگ bAllowTurnOn اساین شده است که در واقع وقتی این تگ True شود، اینترلاک True بودن این لیبل برقرار میشود (توجه کنید که این تگ مقدارش توسط لیبل پایینی خود تغییر میکند). به همین صورت به اتریب bInterlockTurnOff تگی به نام bAllowTurnOff را اساین شده است. توجه کنید که منظور از اینترلاک این است که تا زمانی که اینترلاک برقرار نشود در لیبل که اینترلاک دارد نمیتوان تغییراتی را اعمال کرد.

اتریب اسکات مربوط به لیبل های بالایی را در شکل زیر در نظر بگیرید:

Attribs Scout of KadFrame ('BasicTest' 48 Objects) [User: Tech1 (17:06)]

Device Names List

[KLabel.LabelAllowOff]

Indx	Attrib Name	Data Type	Data(Value)	Tag	TagTransform	Visibility
15	nFontHeight3	Int32	8			Yes
16	bHasOnOffCommand	Bool (?)	True			Yes
17	bShowOnOffMenu	Bool (?)	True			Yes
18	bTurnOn	Bool (?)	True	bAllowTurnOff		Yes
19	bIsOn	Bool (?)	True			Yes
20	bInterlockTurnOn	Bool	True			Yes
21	bInterlockTurnOff	Bool	True			Yes
22	strTextBoolTagTrue	String	Allow Turn Off			Yes
23	strTextBoolTagFalse	String	Not Allow Turn Off			Yes
24	strNoInterlockTurnOn	String	شرایط روشن کردن برقرار نیست			Yes
25	strNoInterlockTurnOff	String	شرایط خاموش کردن برقرار نیست			Yes
26	bIsGradient	Bool	False			Yes

56) Analog1_nAutoCalcPhyPortValue: Auto Calc => -1:Phy to Port, 0:No Calc, +1:Port to Phy @

Tools Show Only Tagged Save at INI Close

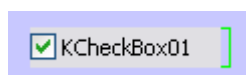
شکل ۵۲

در این بخش یعنی در اتریبهای strNoInterlockTurnOn و strNoInterlockTurnOff ما تعیین میکنیم در حالت برقرار نبودن هر یک از اینترلاکها چه پیغامی نمایش داده شود. شکل زیر را در نظر بگیرید:



در این شکل از آنجا که اینترلاک ها برقرار نیست (دو تا لیبیل پایینی قرمز هستند)، وقتی می‌خواهیم لیبیل بالایی را Turn On کنیم یک بالن ارور نمایش داده میشود که بیان میکند شرایط اینترلاک برقرار نیست.

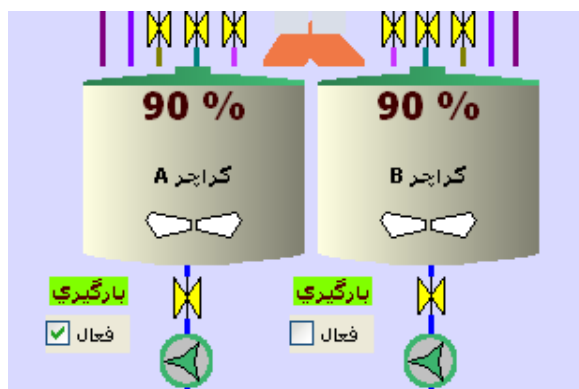
۱۸,۹ KCheckBox



این قابلیت معمولاً برای نشان دادن فعال بودن یا غیر فعال بودن چیزی استفاده میشود، کاربرد بیشتر آن در صفحه فرمول، برای نشان دادن فعال یا غیر فعال بودن یک سطر از آن است:

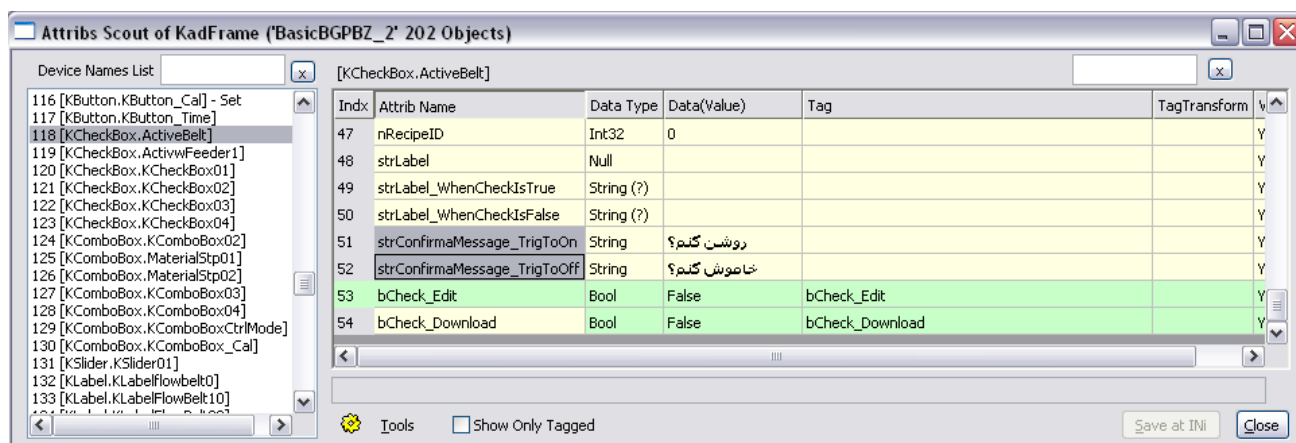
فعال	پیش شرط تخلیه	تخلیه دارد	پیش شرط شارژ واحد	ست پوینت	نام ماده	وضعیت گراچر B	وضعیت گراچر A	Step
<input checked="" type="checkbox"/>	0	<input type="checkbox"/>	0	700 Kg	پس آب	End of Step	End of Step	1
<input type="checkbox"/>	0	<input checked="" type="checkbox"/>	1	200 Kg	سود	Reset	Reset	2
<input checked="" type="checkbox"/>	8	<input checked="" type="checkbox"/>	2	480 Kg	اسپد	Reset	Reset	3

و یا فعال بود یک اکچواتور است:



مرجع کار با برنامه مانیتورینگ کادفریم

برای امنیت بیشتر، چک باکس دارای قابلیت ConfirmaMessage است که آن و آفش تصادفی تریگ نشود، مثلاً همان گونه که در شکل ۵۳ مشاهده می شود، میتوانیم کاری کنیم که برای تریگ کردنش سوالی از اپراتور بپرسد.



شکل ۵۳

مشخصات چک باکس همگی در اتریب های زیر تعریف می شوند:

جدول ۱۰

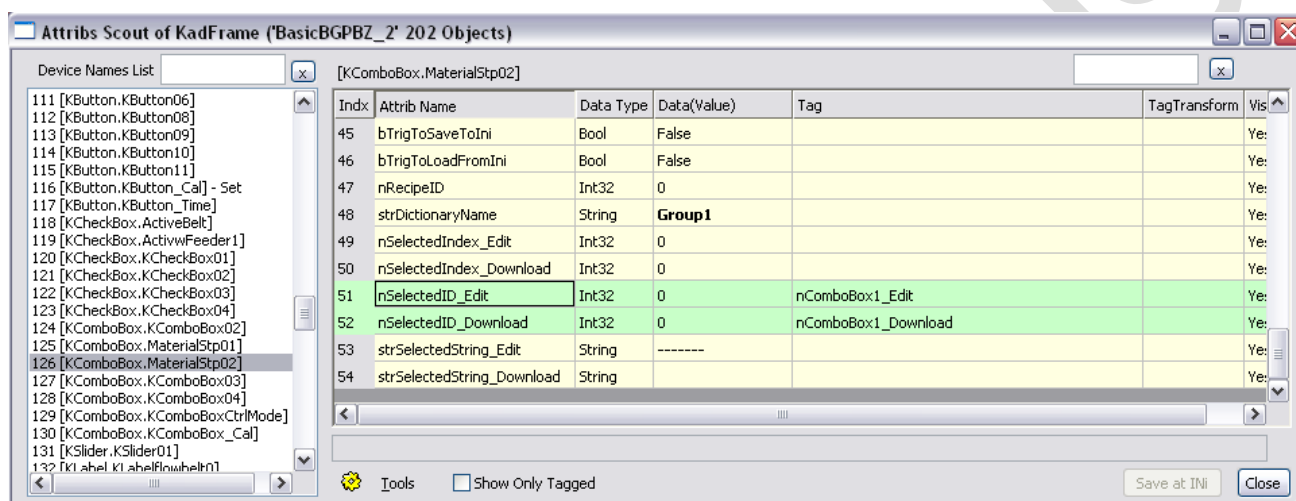
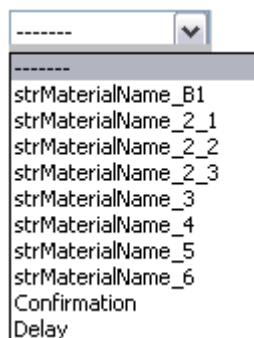
شماره	اتریب های KCheckBox	نوع دیتا	توضیحات
36	bInterlock_Edit	Bool	اساین تگ اینترلاک به این اتریب
37	strInterlock_Edit	String	راهنمای متنی نوشته شده برای اینترلاک
38	bEnable_Edit	Bool	فعال کردن المان
39	strInsufficientAccessMsg	String	
40	nGroupID	Int32	شماره گروه همه المانهای موجود در هر صفحه فرمول باید یکی باشد.
41	bNeedToDownload	Bool	
42	nNeedToDownload_Code	Int32 (?)	

	Bool	bTrigToDownloadToPLC	43
	Bool	bTrigToUploadFromPLC	44
	Bool	bTrigToSaveToIni	45
	Bool	bTrigToLoadFromIni	46
شماره فرمول انتخاب شده	Int32	nRecipeID	47
متن نوشته شده در کنار چک باکس	Null	strLabel	48
متن نوشته شده در کنار چک باکس زمانی که فعال است.	String (?)	strLabel_WhenCheckIsTrue	49
متن نوشته شده در کنار چک باکس زمانی که غیرفعال است.	String (?)	strLabel_WhenCheckIsFalse	50
پیغام تایید در هنگام فعال کردن تیک چک باکس	String	strConfirmMessage_TrigToOn	51
پیغام تایید در هنگام غیرفعال کردن تیک چک باکس	String	strConfirmMessage_TrigToOff	52
اساین تگی که با فعال یا غیرفعال شدن تیک چک باکس، فعال یا غیرفعال می شود.	Bool	bCheck_Edit	53
	Bool	bCheck_Download	54

۱۸,۱۰ KComboBox

قابلیتی است که به صورت یک کشو بازشونده میتوان یک تعداد پارامتر یا مقدار را انتخاب کرد. در این کنترل باید یک دیکشنری را در اتریب strDictionaryName تعریف کنیم. وقتی از نوار کشویی یکی را انتخاب میکنیم، آی دی آن در اتریب nSelectedID_Edit نوشته میشود و اگر تریگ این کنترل زده شود، مقدار ادیت در اتریب nSelectedID_Download وارد میشود.

مرجع کار با برنامه مانیتورینگ کادفریم



شکل ۵۴

مشخصات کمبوباکس همگی در اتریب های زیر تعریف می شوند:

جدول ۱۱

شماره	اتریب های KComboBox	نوع دیتا	توضیحات
36	bInterlock_Edit	Bool	اساین تگ اینترلاک به این اتریب
37	strInterlock_Edit	String	راهنمای متنی نوشته شده برای اینترلاک
38	bEnable_Edit	Bool	فعال کردن المان
39	strInsufficientAccessMsg	String	
40	nGroupID	Int32	شماره گروه همه المانهای موجود در هر صفحه فرمول باید یکی باشد.

مرجع کار با برنامه مانیتورینگ کادفریم

	Bool	bNeedToDownload	41
	Int32 (?)	nNeedToDownload_Code	42
	Bool	bTrigToDownloadToPLC	43
	Bool	bTrigToUploadFromPLC	44
	Bool	bTrigToSaveToIni	45
	Bool	bTrigToLoadFromIni	46
شماره فرمول انتخاب شده	Int32	nRecipeID	47
نام دیکشنری مورد نظر (این قسمت در بخش ۱۷ به طور کامل توضیح داده شده است).	String	strDictionaryName	48
	Int32	nSelectedIndex_Edit	49
	Int32	nSelectedIndex_Download	50
تگ مربوط به کمبویاکس، به این اتریب اساین می شود.	Int32	nSelectedID_Edit	51
	Int32	nSelectedID_Download	52
نمایش مورد انتخاب شده از کمبویاکس در این اتریب	String	strSelectedString_Edit	53
	String	strSelectedString_Download	54

۱۸,۱۱ kEditBox



مشخصات ادیت باکس همگی در اتریب های زیر تعریف می شوند:

جدول ۱۲

شماره	اترپ های KEditBox	نوع دیتا	توضیحات
36	bInterlock_Edit	Bool	اساین تگ اینترلاک به این اترپ
37	strInterlock_Edit	String	راهنمای متنی نوشته شده برای اینترلاک
38	bEnable_Edit	Bool	فعال کردن المان
39	strInsufficientAccessMsg	String	
40	nGroupID	Int32	شماره گروه همه المانهای موجود در هر صفحه فرمول باید یکی باشد.
41	bNeedToDownload	Bool	
42	nNeedToDownload_Code	Int32 (?)	
43	bTrigToDownloadToPLC	Bool	
44	bTrigToUploadFromPLC	Bool	
45	bTrigToSaveToIni	Bool	
46	bTrigToLoadFromIni	Bool	
47	nRecipeID	Int32	شماره فرمول انتخاب شده
48	bAllowOnlyIntegerNumber	Bool	
49	nMinRangeNumber	Int32	
50	nMaxRangeNumber	Int32	
51	nAutoEnterByDeleySp	Int32 (?)	
52	strText_Edit	Null	تگ ورودی ادیت باکس به این اترپ اساین می شود.
53	strText_Download	Null	

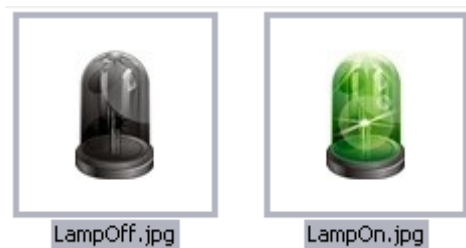
KPicture ۱۸,۱۲

کادفریم این قابلیت را دارد تا با انتخاب این گزینه اشیای دیگر را به محیط خود اضافه کند. به طور مثال برای نشان

دادن یک لامپ و خاموش و روشن بودن آن به روش زیر عمل می شود:

۱- عکسی از لامپ در دو حالت (خاموش و روشن) با فرمت jpg در فولدر Pic تعریف شود:

مرجع کار با برنامه مانیتورینگ کادفریم

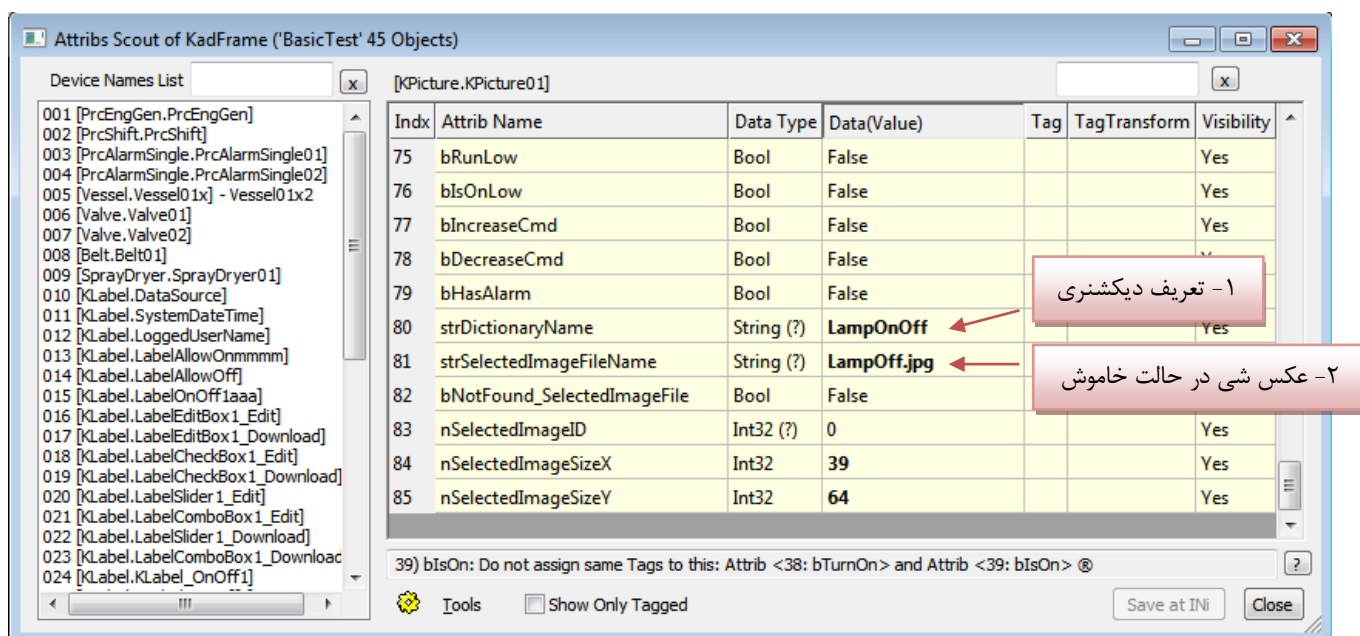


۲- در فولدر MainApp، Ini یک دیکشنری مطابق شکل 55 ایجاد شود:



شکل 55

۳- با انتخاب KPicture در صفحه کادفریم، تنظیمات زیر انجام می شود. به strSelectedImageFileName عکس لامپ در حالت خاموش داده شده و با یک شدن تگ اساین شده به nSelectedImageID، لامپ روشن می شود. مقادیر موجود در Data (Value) اتریب های شماره ی ۸۴ و ۸۵ نیز ساینز عکس را نشان می دهند.



شکل 56

تمامی مشخصات KPicture در اتریب های زیر تعریف می شوند:

جدول ۱۳

شماره	اتریب های KPicture	نوع دیتا	توضیحات
36	bHasOnOffCommand	Bool (?)	برای فعال و غیرفعال کردن قابلیت روشن و خاموش شدن به صورت دستی
37	bShowOnOffMenu	Bool (?)	برای فعال و غیرفعال کردن منوی روشن و خاموش شدن به صورت دستی
38	bTurnOn	Bool (?)	فرمان روشن شدن المان
39	bIsOn	Bool (?)	فیدبک به منظور روشن بودن المان
40	bInterlockTurnOn	Bool	اینترلاک برای فرمان روشن شدن
41	bInterlockTurnOff	Bool	اینترلاک برای فرمان خاموش شدن
42	strTextBoolTagTrue	String	نمایش متن در هنگام فعال شدن المان
43	strTextBoolTagFalse	String	نمایش متن در هنگام غیرفعال شدن المان

44	strNoInterlockTurnOn	String	نمایش متن برای زمانی که اینترلاک روشن شدن المان برقرار نیست.
45	strNoInterlockTurnOff	String	نمایش متن برای زمانی که اینترلاک خاموش شدن المان برقرار نیست.
46	fManualExcitation	Float32	تحریک دستی
47	Excitation_Phy	Null	تحریک جاری المان
48	Excitation_ZeroPhy	Null	مقدار Zero فیزیکی
49	Excitation_SpanPhy	Float32	مقدار Span فیزیکی
50	Excitation_LowLevel	Null	حداقل تحریک فیزیکی
51	Excitation_HighLevel	Float32	حداکثر تحریک فیزیکی
52	Excitation_Port	Null	عدد الکتریکی جاری (که از PLC خوانده می شود).
53	Excitation_ZeroPort	Float32	مقدار Zero پورت
54	Excitation_SpanPort	Float32	مقدار Span پورت
55	Excitation_Caption	String	نام در نظر گرفته شده برای تحریک در جدول Properties
56	Excitation_Unit	String	واحد تحریک که به صورت % بیان می شود.
57	Excitation_nAutoCalcPhyPortValue	Int32 (?)	برای محاسبه ی عدد پورت به فیزیکی و یا فیزیکی به پورت به صورت اتومات
58	Flow_Phy	Null	فلوی جاری المان
59	Flow_ZeroPhy	Null	مقدار Zero فیزیکی
60	Flow_SpanPhy	Float32	مقدار Span فیزیکی
61	Flow_LowLevel	Null	حداقل فلو فیزیکی
62	Flow_HighLevel	Float32	حداکثر فلو فیزیکی
63	Flow_Port	Null	عدد الکتریکی جاری (که از PLC خوانده می شود).

مرجع کار با برنامه مانیتورینگ کادفریم

			شود.)
64	Flow_ZeroPort	Float32	مقدار Zero پورت
65	Flow_SpanPort	Float32	مقدار Span پورت
66	Flow_Caption	String	نام در نظر گرفته شده برای فلو در جدول Properties
67	Flow_Unit	String	واحد فلو که به صورت Kg/h بیان می شود.
68	Flow_nAutoCalcPhyPortValue	Int32 (?)	برای محاسبه ی عدد پورت به فیزیکیال و یا فیزیکیال به پورت به صورت اتومات
69	fIncDecStepPercent	Float32	ست پوینت به منظور افزایش یا کاهش تحریک با فعال شدن اتریب های ۷۷ و ۷۸
70	fExcitationHighPercent	Float32	درصد ماکزیمم تحریک
71	fExcitationLowPercent	Float32	درصد مینیمم تحریک
72	nOnTimeAppRecordID	Int32 (?)	ID که به محض روشن شدن المان ایجاد می شود (جهت استفاده در بانک اطلاعاتی).
73	bRunHigh	Bool	فرمان روشن شدن با دور تند
74	bIsOnHigh	Bool	فرمان فیدبک با دور تند
75	bRunLow	Bool	فرمان روشن شدن با دور کند
76	bIsOnLow	Bool	فرمان فیدبک با دور کند
77	bIncreaseCmd	Bool	دستور افزایش تحریک به اندازه ست پوینت مشخص شده
78	bDecreaseCmd	Bool	دستور کاهش تحریک به اندازه ست پوینت مشخص شده
79	bHasAlarm	Bool	آلارم داشتن المان (اساین تگ مربوط به آلارم)
80	strDictionaryName	String	نام دیکشنری مورد نظر (این قسمت در بخش ۱۷ به طور کامل توضیح داده شده است).

اساین عکس مورد نظر در قالب Gif به این اتریب و یا در صورت فعال یا غیرفعال بودن المان مانند مثال ذکر شده	String (?)	strSelectedImageFileName	81
	Bool (?)	bNotFound_SelectedImageFile	82
زمانی که bHasOnOffCommand یک باشد، مقدار صفر یا یک bIsOn را می گیرد.	Int32 (?)	nSelectedImageID	83
نمایش سایز X عکس انتخاب شده	Int32	nSelectedImageSizeX	84
نمایش سایز Y عکس انتخاب شده	Int32	nSelectedImageSizeY	85

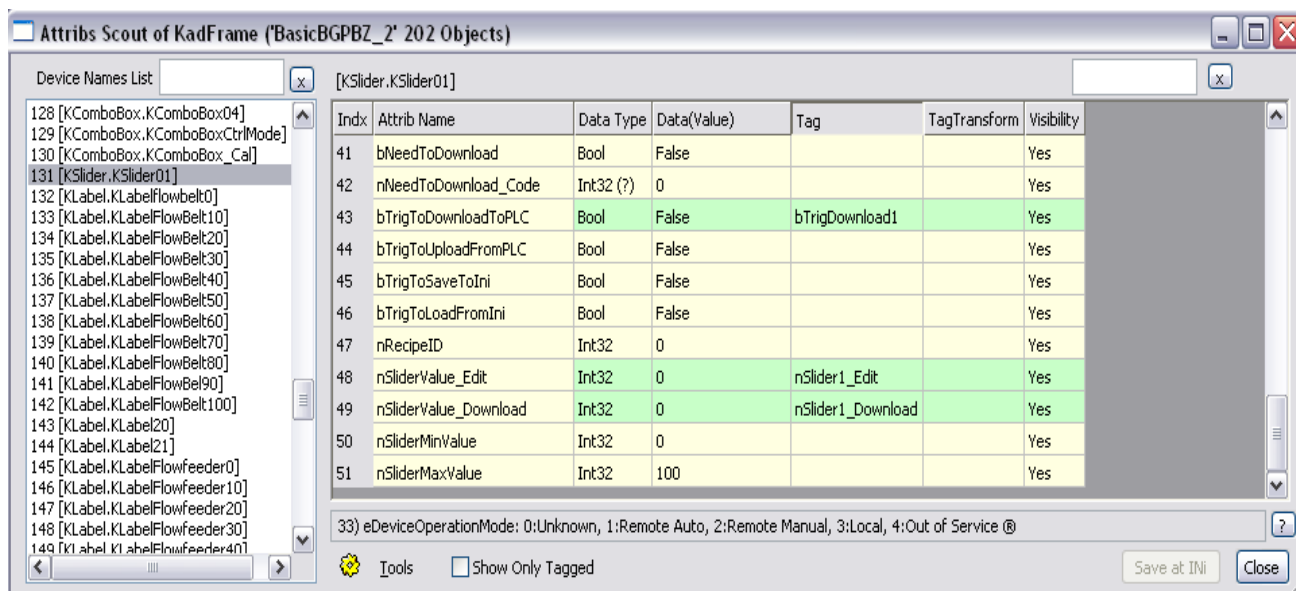
۱۸,۱۳ KSlider

قابلیتی است که میتوان یک مقدار را به صورت کشویی تعیین کرد. برای این اسلایدر میتوان بازه مینیمم و ماکزیمم تعیین کرد. اتریب مربوط به بازه مینیمم و ماکزیمم به ترتیب nSliderMinValue و nSliderMaxValue هستند.



اتریب دیگری که در این کنترل وجود دارد nSliderValue_Edit است، مقدار KSlider ابتدا در این اتریب نوشته میشود و بعد از تریگ شدن در اتریب nSliderValue_Download نوشته میشود. تریگ این کنترل از اتریب bTrigToDownloadToPLC انجام میشود. این تگ میتواند از هر جایی ست شود.

مرجع کار با برنامه مانیتورینگ کادفریم



شکل ۵۷

تمامی مشخصات KSlider در اتریب های زیر تعریف می شوند:

جدول ۱۴

شماره	اتریب های KSlider	نوع دیتا	توضیحات
36	bInterlock_Edit	Bool	اساین تگ اینترلاک به این اتریب
37	strInterlock_Edit	String	راهنمای متنی نوشته شده برای اینترلاک
38	bEnable_Edit	Bool	فعال کردن المان
39	strInsufficientAccessMsg	String	
40	nGroupID	Int32	شماره گروه همه المانهای موجود در هر صفحه فرمول باید یکی باشد.
41	bNeedToDownload	Bool	
42	nNeedToDownload_Code	Int32 (?)	
43	bTrigToDownloadToPLC	Bool	
44	bTrigToUploadFromPLC	Bool	
45	bTrigToSaveToIni	Bool	

مرجع کار با برنامه مانیتورینگ کادفریم

	Bool	bTrigToLoadFromIni	46
شماره فرمول انتخاب شده	Int32	nRecipeID	47
تگ مربوط به مقدار مشخص شده در اسلایدر	Int32	nSliderValue_Edit	48
	Int32	nSliderValue_Download	49
حداقل مقدار اسلایدر	Float32	nSliderMinValue	50
حداکثر مقدار اسلایدر	Int32	nSliderMaxValue	51

۱۸,۱۴ PrcAlarmSingle

با انتخاب این المان می توان آلام را به کادفریم اضافه کرد. اتریب های PrcAlarmSingle شامل موارد زیر می باشند:

جدول ۱۵

شماره	اتریب های PrcAlarmSingle	نوع دیتا	توضیحات
1	strDeviceCaption	String	
3	bServiceTheAlarmInPC_Demo	Bool	
4	bServiceTheAlarmInPC_Real	Bool	
5	bFault	Bool	
6	bWarningOrRunAction	Bool	
7	bAcknowledge	Bool	
8	bActionDone	Bool	
9	bConfigAction_StopProcess	Bool	
10	bConfigAction_0	Bool	
11	bConfigAction_1	Bool	
12	bConfigAction_2	Bool	

	Bool	bConfigAction_3	13
	Bool	bConfigAction_4	14
	Bool	bConfigAction_5	15
	Bool	bConfigAction_6	16
	Bool	bConfigAction_7	17
	Bool	bAutoResetAlarmAck	18
	Bool	bDoNotRecordInDataBase	19
	Int32	nConfigAction_Decimal	20
	Int32 (?)	nAlarmDisplayType	21
	Int32 (?)	nMessageBoxButtonsCode	22
	Int32 (?)	nMessageBoxResponseCode	23
	Int32	nAlarmDeleyTimer	24
	Int32	nAlarmDeleySetPoint	25
	Int32	nAudioDeleyTimer	26
	Int32	nAudioDeleySetpoint	27
	Int32 (?)	nBalloonPosX	28
	Int32 (?)	nBalloonPosY	29
	Int32	nActionExecuteOneFC	30
	Int32	nStepNumber	31
	String	strAlarmTitle	32
	String	strAlarmDescriptionLine1	33
	String	strAlarmDescriptionLine2	34
	String	strAlarmDescriptionLine3	35
	String	strAlarmDescriptionLine4	36
	String	strAlarmDescriptionLine5	37

	String (?)	strDeviceNameForBalloon	38
		Anchor	

RotaryMixer ۱۸,۱۵



اترید های این المان در ۷۹ اترید اول کاملاً مشابه اترید های نوار می باشد (به جدول ۳ رجوع شود) و چند اترید دیگر در جدول ۱۶ معرفی شده اند.

جدول ۱۶

شماره	اترید های RotaryMixer	نوع دیتا	توضیحات
80	Level_Phy	Null	وزن جاری المان
81	Level_ZeroPhy	Null	مقدار Zero فیزیکی
82	Level_SpanPhy	Float32	مقدار Span فیزیکی
83	Level_LowLevel	Null	حداقل وزن فیزیکی
84	Level_HighLevel	Float32	حداکثر وزن فیزیکی
85	Level_Port	Null	عدد الکتریکی جاری (که از PLC خوانده می شود.)
86	Level_ZeroPort	Float32	مقدار Zero پورت

مرجع کار با برنامه مانیتورینگ کادفریم

مقدار Span پورت	Float32	Level_SpanPort	87
نام در نظر گرفته شده برای وزن در جدول Properties	String	Level_Caption	88
واحد وزن که به صورت Kg بیان می شود.	String	Level_Unit	89
برای محاسبه ی عدد پورت به فیزیکال و یا فیزیکال به پورت به صورت اتومات	Int32 (?)	Level_nAutoCalcPhyPortValue	90

RotaryWheel ۱۸,۱۶



RotaryWheel دارای 81 اتریب می باشد که ۷۹ اتریب اول با نوار مشابه می باشد (به جدول ۳ رجوع شود) و ۲ اتریب زیر به آن اضافه شده است:

جدول ۱۷

شماره	اتریب های اضافه RotaryWheel	نوع دیتا	توضیح
80	nBladesCount	Int32	
81	nMovingDirection	Int32	



مشخصات اسپری درایر همگی در اتریب های زیر تعریف می شوند:

جدول ۱۸

شماره	اتریب های SprayDryer	نوع دیتا	توضیحات
36	fTowerSafetyCurMaxTower Temperatur	Float32	
37	fTowerSafetyTemperSlope1	Float32	
38	fTowerSafetyTemperSlope2	Float32	
39	fTowerSafetyTemperSlope3	Float32	
40	fTowerSafetyTemperSlope4	Float32	
41	fTowerSafetyTemperSlope5	Float32	
42	fTowerSafetyTemperSlope6	Float32	
43	fTowerSafetyTemperSlope7	Float32	
44	fTowerSafetyTemperSlope8	Float32	
45	bShowTowerSafetyTempera t1	Bool	نمایش مقدار دمای اول امنیت برج بر روی المان
46	bShowTowerSafetyTempera t2	Bool	نمایش مقدار دمای دوم امنیت برج بر روی المان
47	bShowTowerSafetyTempera t3	Bool	نمایش مقدار دمای سوم امنیت برج بر روی المان
48	bShowTowerSafetyTempera	Bool	نمایش مقدار دمای چهارم امنیت برج بر روی المان

		t4	
	Bool	bFireDetected	49
	Bool	bFireFighting	50
	Bool	bStopBurner	51
	Bool	bReduceBurner	52
	Bool	bLetHammerAutoService	53
	Bool	bHammerIsInAutoService	54
	Bool	bTurnOnHammer1	55
	Bool	bTurnOnHammer2	56
	Bool	bTurnOnHammer3	57
	Bool	bTurnOnHammer4	58
	Bool	bTurnOnHammer5	59
	Bool	bTurnOnHammer6	60
	Bool	bTurnOnHammer7	61
	Bool	bTurnOnHammer8	62
	Int32	iHammerOperationMode	63
	Int32	iHammerCyclePeriod	64
	Int32	iHammerCycleTimer	65
	Int32	iOnDelayHammer1	66
	Int32	iOnDelayHammer2	67
	Int32	iOnDelayHammer3	68
	Int32	iOnDelayHammer4	69
	Int32	iOnDelayHammer5	70
	Int32	iOnDelayHammer6	71
	Int32	iOnDelayHammer7	72
	Int32	iOnDelayHammer8	73
	Int32	iOnStateTimeHammer1	74
	Int32	iOnStateTimeHammer2	75
	Int32	iOnStateTimeHammer3	76
	Int32	iOnStateTimeHammer4	77
	Int32	iOnStateTimeHammer5	78
	Int32	iOnStateTimeHammer6	79

	Int32	iOnStateTimeHammer7	80
	Int32	iOnStateTimeHammer8	81
	Int32	iTowerSafetyCriticalDeltaT _PerMin	82
	Int32	iTowerSafetySlopeCalcPeri od	83
	Int32	iTowerSafetyCriticalTopTo werTemper	84
	Int32	iTowerSafetyCriticalFireTe mper	85
دمای ۱ جاری امنیت برج المان	Float32	TwrSfty Tempr1_Phy	86
مقدار Zero فیزیکی	Null	TwrSfty Tempr1_ZeroPhy	87
مقدار Span فیزیکی	Float32	TwrSfty Tempr1_SpanPhy	88
حداقل دمای ۱ امنیت برج فیزیکی	Null	TwrSfty Tempr1_LowLevel	89
حداکثر دمای ۱ امنیت برج فیزیکی	Float32	TwrSfty Tempr1_HighLevel	90
عدد الکتریکی جاری (که از PLC خوانده می شود).	Null	TwrSfty Tempr1_Port	91
مقدار Zero پورت	Float32	TwrSfty Tempr1_ZeroPort	92
مقدار Span پورت	Float32	TwrSfty Tempr1_SpanPort	93
نام در نظر گرفته شده برای دمای ۱ امنیت برج در جدول Properties	String	TwrSfty Tempr1_Caption	94
واحد دما که به صورت درجه سانتی گراد بیان می شود.	String	TwrSfty Tempr1_Unit	95
برای محاسبه ی عدد پورت به فیزیکی و یا فیزیکی به پورت به صورت اتومات	Int32 (?)	TwrSfty Tempr1_nAutoCalcPhyPort Value	96
دمای ۲ جاری امنیت برج المان	Float32	TwrSfty Tempr2_Phy	97
مقدار Zero فیزیکی	Null	TwrSfty Tempr2_ZeroPhy	98
مقدار Span فیزیکی	Float32	TwrSfty Tempr2_SpanPhy	99

مرجع کار با برنامه مانیتورینگ کادفریم

100	TwrSfty Tempr2_LowLevel	Null	حداقل دمای ۲ امنیت برج فیزیکی
101	TwrSfty Tempr2_HighLevel	Float32	حداکثر دمای ۲ امنیت برج فیزیکی
102	TwrSfty Tempr2_Port	Null	عدد الکتریکی جاری (که از PLC خوانده می شود).
103	TwrSfty Tempr2_ZeroPort	Float32	مقدار Zero پورت
104	TwrSfty Tempr2_SpanPort	Float32	مقدار Span پورت
105	TwrSfty Tempr2_Caption	String	نام در نظر گرفته شده برای دمای ۲ امنیت برج در جدول Properties
106	TwrSfty Tempr2_Unit	String	واحد دما که به صورت درجه سانتی گراد بیان می شود.
107	TwrSfty Tempr2_nAutoCalcPhyPort Value	Int32 (?)	برای محاسبه ی عدد پورت به فیزیکی و یا فیزیکی به پورت به صورت اتومات
108	TwrSfty Tempr3_Phy	Float32	دمای ۳ جاری امنیت برج المان
109	TwrSfty Tempr3_ZeroPhy	Null	مقدار Zero فیزیکی
110	TwrSfty Tempr3_SpanPhy	Float32	مقدار Span فیزیکی
111	TwrSfty Tempr3_LowLevel	Null	حداقل دمای ۳ امنیت برج فیزیکی
112	TwrSfty Tempr3_HighLevel	Float32	حداکثر دمای ۳ امنیت برج فیزیکی
113	TwrSfty Tempr3_Port	Null	عدد الکتریکی جاری (که از PLC خوانده می شود).
114	TwrSfty Tempr3_ZeroPort	Float32	مقدار Zero پورت
115	TwrSfty Tempr3_SpanPort	Float32	مقدار Span پورت
116	TwrSfty Tempr3_Caption	String	نام در نظر گرفته شده برای دمای ۳ امنیت برج در جدول Properties
117	TwrSfty Tempr3_Unit	String	واحد دما که به صورت درجه سانتی گراد بیان می شود.
118	TwrSfty Tempr3_nAutoCalcPhyPort	Int32 (?)	برای محاسبه ی عدد پورت به فیزیکی و یا فیزیکی به

مرجع کار با برنامه مانیتورینگ کادفریم

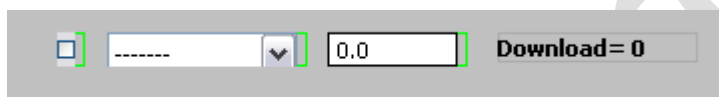
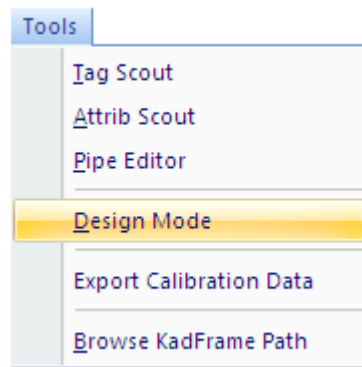
پورت به صورت اتومات	Value		
دمای ۴ جاری امنیت برج المان	Float32	TwrSfty Tempr4_Phy	119
مقدار Zero فیزیکی	Null	TwrSfty Tempr4_ZeroPhy	120
مقدار Span فیزیکی	Float32	TwrSfty Tempr4_SpanPhy	121
حداقل دمای ۴ امنیت برج فیزیکی	Null	TwrSfty Tempr4_LowLevel	122
حداکثر دمای ۴ امنیت برج فیزیکی	Float32	TwrSfty Tempr4_HighLevel	123
عدد الکتریکی جاری (که از PLC خوانده می شود).	Null	TwrSfty Tempr4_Port	124
مقدار Zero پورت	Float32	TwrSfty Tempr4_ZeroPort	125
مقدار Span پورت	Float32	TwrSfty Tempr4_SpanPort	126
نام در نظر گرفته شده برای دمای ۴ امنیت برج در جدول Properties	String	TwrSfty Tempr4_Caption	127
واحد دما که به صورت درجه سانتی گراد بیان می شود.	String	TwrSfty Tempr4_Unit	128
برای محاسبه ی عدد پورت به فیزیکی و یا فیزیکی به پورت به صورت اتومات	Int32 (?)	TwrSfty Tempr4_nAutoCalcPhyPort Value	129

۱۹ Design Mode

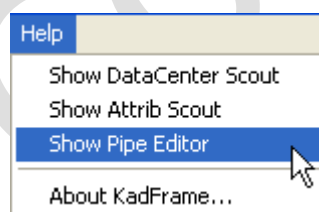
این قابلیت در منوی Tools می باشد.

همانطور که در شکل زیر میبیند در مد دیزاین دور هر لیبیل کادری مشاهده میشود که نشان دهنده ی سایز اشغال شده توسط هر لیبیل است. همچنین در کنار ادیت باکس، کمبوباکس، چک باکس و باتن ها هم حاشیه ی سبز رنگی مشاهده میشود که امکان جابجایی و دسترسی راحت تر به اتریب اسکات هر کنترل را به ما می دهد. در حالت دیزاین مود میتوان با کلیک راست و دابل کلیک به اتریب اسکات آن کنترل دسترسی داشت.

مرجع کار با برنامه مانیتورینگ کادفریم



۲۰ لوله های انتقال



پس از آنکه جای هر کدام از اشیا گرافیکی در صفحه کادفریم مشخص شد نوبت به کشیدن لوله های انتقال می شود
برای این کار منوی Help گزینه Show Pipe Editor را انتخاب کرده تا وارد صفحه Piping Editor شویم.



شکل ۵۸

همانطور که در این صفحه می بینیم ستون هایی برای تنظیم خط وجود دارد که در ادامه این ستون ها شرح داده میشود:

nStartX: مختصات نقطه شروع خط را در محور افقی نشان میدهد.

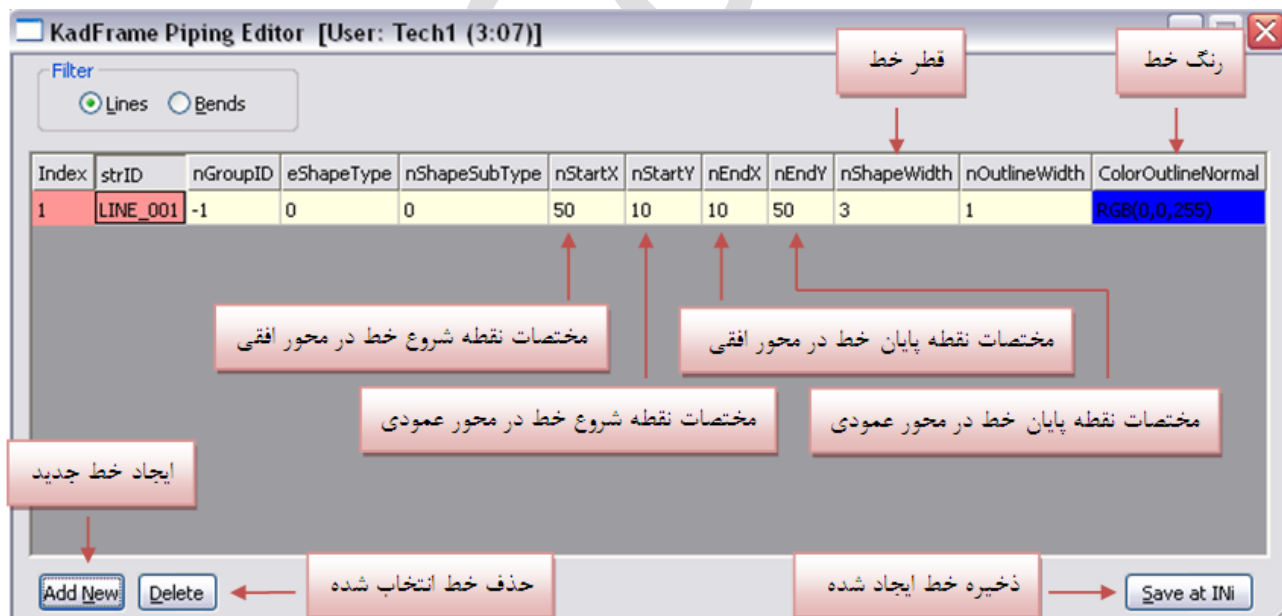
nStartY: مختصات نقطه شروع خط در محور عمودی را نشان میدهد.

nEndX: مختصات نقطه پایان خط در محور افقی را نشان می دهد.

nEndY: مختصات نقطه پایان خط در محور عمودی را نشان می دهد.

nShapeWidth: در این قسمت قطر (ضخامت) خط تعیین می شود.

ColorOutlineNormal: در این قسمت رنگ خط ایجاد شده تنظیم می شود.



شکل ۵۹

حال روی گزینه Add New کلید کرده تا یک خط جدید در صفحه کادفریم ایجاد گردد. همانطور که گفته شد خط ایجاد شده در محور X-Y دارای مختصات خاصی می باشد برای قرار دادن خط ایجاد شده در محل مورد نظر کافی است مختصات نقطه مورد نظر را در قسمت تنظیمات خط وارد کرده تا خط در محل مورد نظر ما قرار گیرد. نکته ای که باید در رسم خطوط در نظر داشت این است که همیشه خطوط از زیر سایر اشکال داخل صفحه رسم می شوند.

۲۱ Trends

در کادفریم دو نوع Trend داریم، ترند آف لاین و ترند آن لاین که این کنترل هم باید در INI تعریف شود، [Trend- DataBase] برای نمودارهای آف لاین و [Trend-Live] برای نمودارهای آنلاین می باشد. اجزاء ترند آفلاین عبارتند از:

Template_DataBase=(no need '=') strTable, strField, nLineColor, strTrendTitle

strTable: جدولی که مقادیر نمودار از آن خوانده می شود.

strField: نام فیلدی در جدول فوق که مقادیر این فیلد در نمودار نمایش داده می شود.

nLineColor: رنگ منحنی نمودار

strTrendTitle: عنوان نمودار

اجزاء ترند آنلاین عبارتند از:

Template_Live=(no need '=') nRecordCount, nRecordInterval, nLineColor, strTagName, strTrendTitle

مرجع کار با برنامه مانیتورینگ کادفریم

nRecordCount: حداکثر تعداد رکوردها ی قابل نمایش در نمودار

nRecordInterval: مدت زمان بین ثبت رکورد ها

nLineColor: رنگ منحنی نمودار

strTagName: نام تگی که مقادیر آن در نمودار نمایش داده می شود.

strTrendTitle: عنوان نمودار

۲۱,۱ نحوه ساخت و نمایش یک ترند در نرم افزار کادفریم

۲۱,۱,۱ ترند آفلاین

به عنوان مثال در حالت ترند آفلاین برای Belt0 مشخصه ی Load برای نمایش درخواست می شود:
از آن جا که در حالت آفلاین اطلاعات از بانک اطلاعاتی خوانده می شود، باید قبل از نوشتن اطلاعات در بخش ترند، در یک جدول (Table) مشخص (در این جا تیبل Event) تگ مشخصه ی Load را برای یک فیلد تعریف کنیم (توضیحات مربوط به ایجاد یک جدول در سرفصل ۱۲,۲ آمده است).

Event.ini

```
1
2 [Setting]
3 strTableName=Event
4 nRecordingIntervalSec=5
5 strTagToTrigRecord=PAPN.Process.bProcessRunning // Comment
6 strTagToDisableRecording=__NO_TAG__ // Comment
7 nDelaySecToRecordAfterCloseActuator=-1 // if -1, will not be trigged
8
9 [Help]
10 Template=strFieldName, nAveragingBufferSize, strTagName
11
12 [Fields]
13 //===== Belt0
14 fLoadBelt0 , 10, PAPN.Process.fLoadBelt0
15
```

نام فیلد تعریف شده در جدول

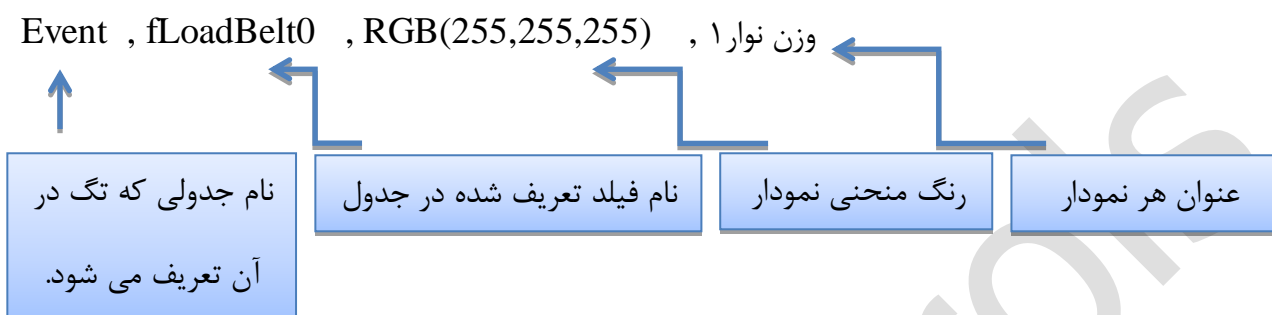
تعداد نمونه میانگین گیری

تگ های مربوطه

Event

شکل ۶۰

حال مطابق در فولدر Ini, در فایل Trend.ini در بخش Trend-DataBase نام فیلد و جدولی که اطلاعات باید از آن خوانده شود، نوشته می شود.



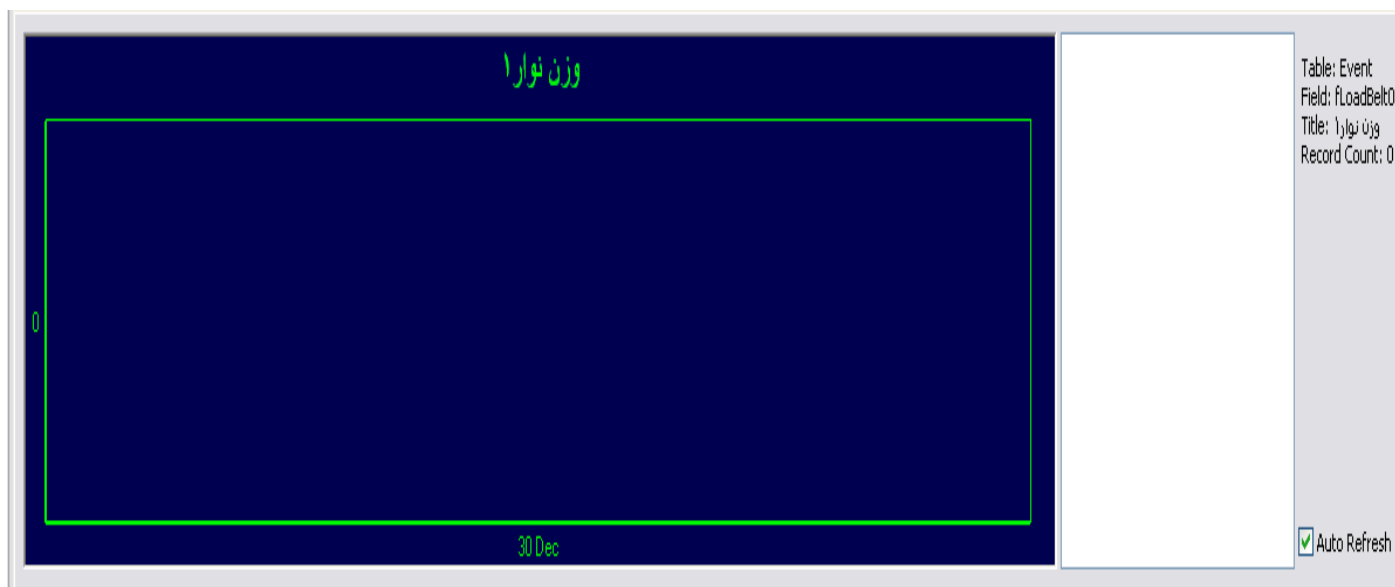
در شکل ۶۲ نمودار در صفحه Trend DataBase (Offline) کادفریم، ایجاد می شود.

```

Trends.ini
1 //----- KadFrame Ini File ----- SaveTime: 1394/01/09 - 14:57:32
2
3 [Help]
4 Template_Live = (no need '=') nRecordCount, nRecordInterval, nLineColor, strTagName, strTrendTitle
5 Template_DataBase = (no need '=') strTable, strField, nLineColor, strTrendTitle
6
7 [Trend-Live]
8 28800, 1 , RGB(255,0,0) , PPN.Process.fLoadBelt0 , وزن نوار ۱
9
10
11 [Trend-DataBase]
12 Event, fLoadBelt0 , RGB(255,255,255), وزن نوار ۱
13
14 [Setting]
15 nTrendLive_BaseInterval_ms = 1000
16

```

شکل ۶۱

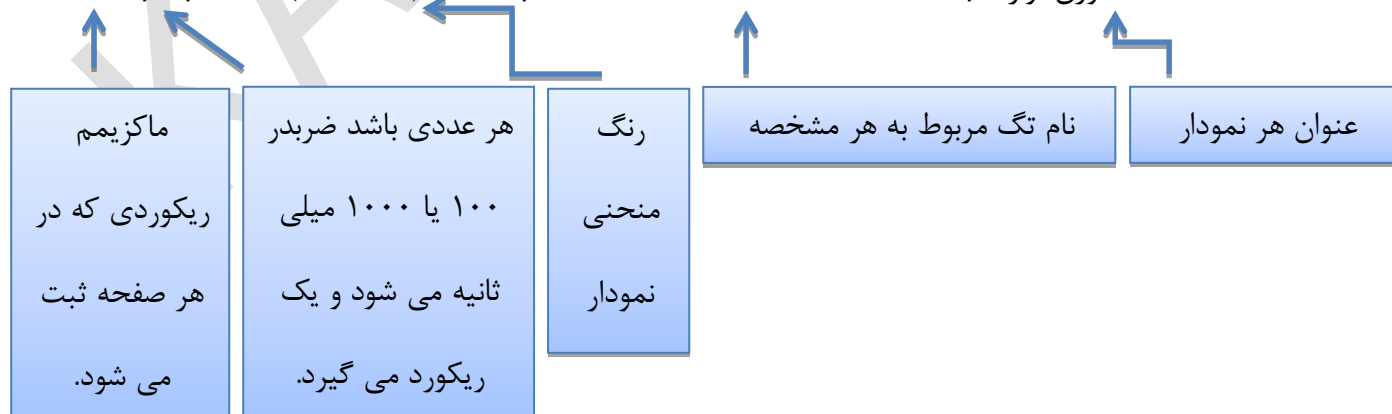


شکل ۶۲

۲۱،۱،۲ ترند آنالاین

برای حالت آنالاین مشخصه ی Load همانگونه که در شکل ۶۱ مشاهده می شود در فولدر Ini، در فایل Trend.ini، در بخش Trend-Live نوشته می شود.

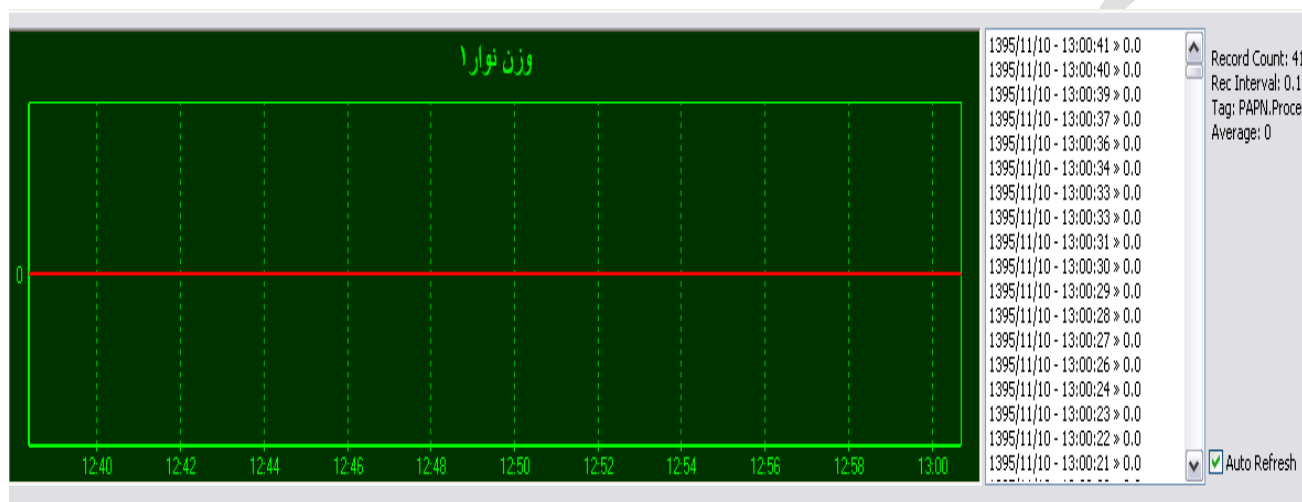
وزن نوار ۱ ، PAPN.Process.fLoadBelt0 ، 28800, 1, RGB(255,0,0)



در کادفریم برای حالت آنلاین می توان سرعت نمونه برداری را ۱۰۰ یا ۱۰۰۰ میلی ثانیه انتخاب کرد.

`nTrendLive_BaseInterval_ms = ۱۰۰۰`

در شکل ۶۳ نمودار در صفحه Trend Live (Offline) کادفریم، ایجاد می شود.



شکل ۶۳

۲۲ قابلیت ذخیره فرمول در کادفریم بیسیک

در کادفریم بیسیک میتوان یک فرمول دلخواه را در صفحه فرمول وارد کرده و با یک نام مشخص آن را ذخیره کرد. برای این منظور باید یک کومبو باکس نویگیتور را ایجاد کرد. در اتریب `nSelectedID_Edit` تگ `nRecipeID_Group1` را اساین میکنیم. یعنی مشخص میکنیم که فرمولی که قرار است ذخیره شود مربوط به کدام گروه مواد است. (منظور از گروه همان گروه المان های صفحه فرمول است که در حال حاضر میتوان چهار گروه را در نظر گرفت). در حقیقت به جای ذخیره، دانلود یا آپلود فرمول از طریق پی ال سی از تگ های داخلی کادفریم

استفاده می شود.

به اتریبهای زیر توجه کنید:

43	bTrigToDownloadToPLC	Bool	False		Yes
44	bTrigToUploadFromPLC	Bool	False		Yes
45	bTrigToSaveToIni	Bool	False		Yes
46	bTrigToLoadFromIni	Bool	False		Yes

اگر اتریب bTrigToSaveToIni تریگ شود در این صورت تمامی مقادیر فرمول در ini ذخیره میشود. پس میتوانیم تگ باتن را با عنوان "ذخیره فرمول" تعریف کنیم. این باتن تگی را تریگ میکند که این تگ اتریب مربوط به ذخیره فرمول در ini را تریگ میکند.

بر اساس تعداد فرمول هایی که میخواهیم ذخیره کنیم میتوان یک دیکشنری برای نام فرمول ها ایجاد کرد؛ که مقادیر این دیکشنری همان نام فرمول های مورد نظر خواهد بود.

فرض کنید که میخواهیم یک فرمول را وارد کرده و آن را با نام Recipe01 ذخیره کنیم. برای این منظور از کومبو باکس نویگیتور مربوط به انتخاب فرمول، فرمول شماره یک را انتخاب میکنیم و فرمول مورد نظر را در صفحه فرمول وارد کرده و فرمول را ذخیره میکنیم. با این کار فرمول وارد شده با نام Recipe01 ذخیره میشود و در ادامه کار هر زمان که از کومبو باکس نویگیتور، فرمول یک را انتخاب کنیم، فرمول مورد نظر در صفحه فرمول لود میشود.

توجه: گروه همه المانهای موجود در صفحه فرمول باید یکی باشد (nGroupID). مثلاً همه از گروه یک باشند.

در ذیل نحوه ذخیره فرمول در Ini شرح داده شده است:

ابتدا فرمت ini را مرور می کنیم:

[Section]

Key=Value

فرمت فوق در کادفریم به شکل اختصاصی زیر استفاده می شود:

[ControlTypeNam.DeviceName]

AttribName.AttribPropType=Value

انواع AttribPropType به شرح زیر است:

Data, Tag, TagTransform, Visible

فوکوس ما رو ذخیره Data یک اتریب است. به عنوان مثال:

مرجع کار با برنامه مانیتورینگ کادفریم

[KEditBox.EditDose01]

strText_Edit.Data=12

نشان می دهد که مقدار ذخیره شده برای ادیت باکس مذکور برابر ۱۲ است.

حال اگر بخواهیم چندین مقدار متفاوت برای یک ادیت باکس ذخیره کنیم از RecipeID استفاده می کنیم:

[ControlTypeNam.DeviceName]

AttribName.Data_NN=Value

که در اینجا NN در Data_NN همان آی دی فرمول است. یعنی به ازای هر آی دی فرمول، یک سطر جداگانه KeyValue برای یک اتریب نگهدارنده مقدار استفاده می شود.

لازم به ذکر است که ذخیره دیتا با فرمت Data_NN فقط برای کنترل های ادیتوری (مثل کمبوباکس، ادیت باکس و ...) و آن هم فقط برای اتریب نگهدارنده مقدار ادیت شده استفاده می شود و کار می کند.

در شکل ۶۴، مقدار متفاوت برای ۳ فرمول با شماره ۰ و ۴ و ۱۲ ذخیره شده است.

```
[KEditBox.EditDose01]
nDrawSizeX.Data=45
nDrawSizeY.Data=17
nDrawPosX.Data=1133
nDrawPosY.Data=50
bEnable_Edit.Tag=bStepIsActive_Edit_01
strText_Download.Tag=SLBD.Recipe.Step01.fLoopDose
strText_Edit.Data=110
strText_Edit.Data_04=1200
strText_Edit.Data_12=1350
```

شکل ۶۴

تگ داخلی برای کار با فرمول در کادفریم nRecipeID_GroupN است.

کمبوباکس نویگیتور فرمول باید جز گروه صفر باشد.

به محض تغییر مقدار در nRecipeID_Group1 (به عنوان نمونه و به هر شکلی)، تمامی کنترل های ادیتوری جز گروه یک، مقادیر متناظر با RecipeID خود را لود کرده و نمایش می دهند.

۲۳ قابلیت آپلود فرمول از PLC

فرض کنید که یک فرمول دلخواه در صفحه فرمول وجود دارد و اپراتور یکی از مقادیر را در صفحه فرمول تغییر میدهد ولی در ادامه میخواهد که همان مقدار قبلی را برگرداند و از تغییر فرمول منصرف میشود. اپراتور میتواند با زدن یک باتن، فرمول جاری در PLC را در صفحه فرمول آپلود کند.

برای اینکار باید به اتریب `bTrigToUploadFromPLC` مربوط به همه المان های مربوط به صفحه فرمول یک تگ مشترک را اساین کرد بطوریکه با زدن یک باتن این تگ تریگ شود. این باتن همان باتن آپلود فرمول از PLC خواهد بود.