

مراحل اساسی طراحی یک پروژه صنعتی

تقسیم پروسه به نواحی مختلف

تشریح نواحی اصلی

برقراری الزامات ایمنی

ایجاد وسایل کنترلی
و نمایشگرهای لازم برای اپراتور

تشکیل دیاگرام پیکربندی

برای مشاهده کامل صفحات از کلید Space استفاده کنید.



تقسیم پروسه به نواحی مختلف



برای مشاهده کامل صفحات از کلید Space استفاده کنید.



تقسیم پروسه به نواحی مختلف

هر پروسه اتوماسیون شامل چندگروه کار مشخص است که با تشخیص وظیفه هر گروه کاری و تفکیک این گروه ها به اجزاء کوچکتر، حتی پروسه های پیچیده تر هم به خوبی قابل اجرا هستند.

مثال "پروسه مخلوط صنعتی" مراحل ساماندهی و تقسیم پروسه به نواحی مختلف و تعریف وظایف محوله هر ناحیه را روشن می کند.

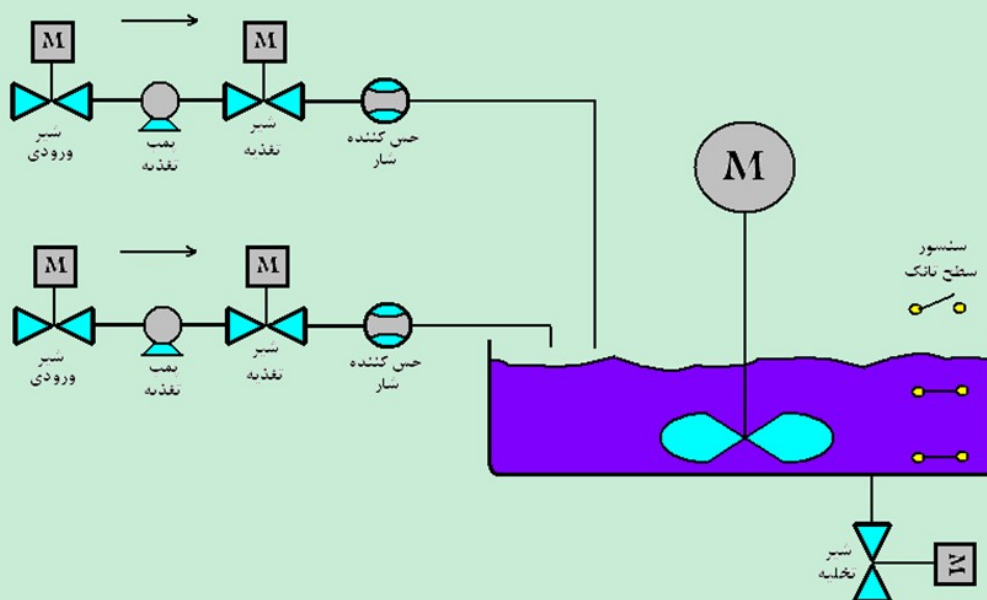
دو سیال (ingredient) توسط یک همزن، در داخل یک مخزن با هم مخلوط می شوند و محصول نهایی توسط شیر تخلیه خارج می شود.

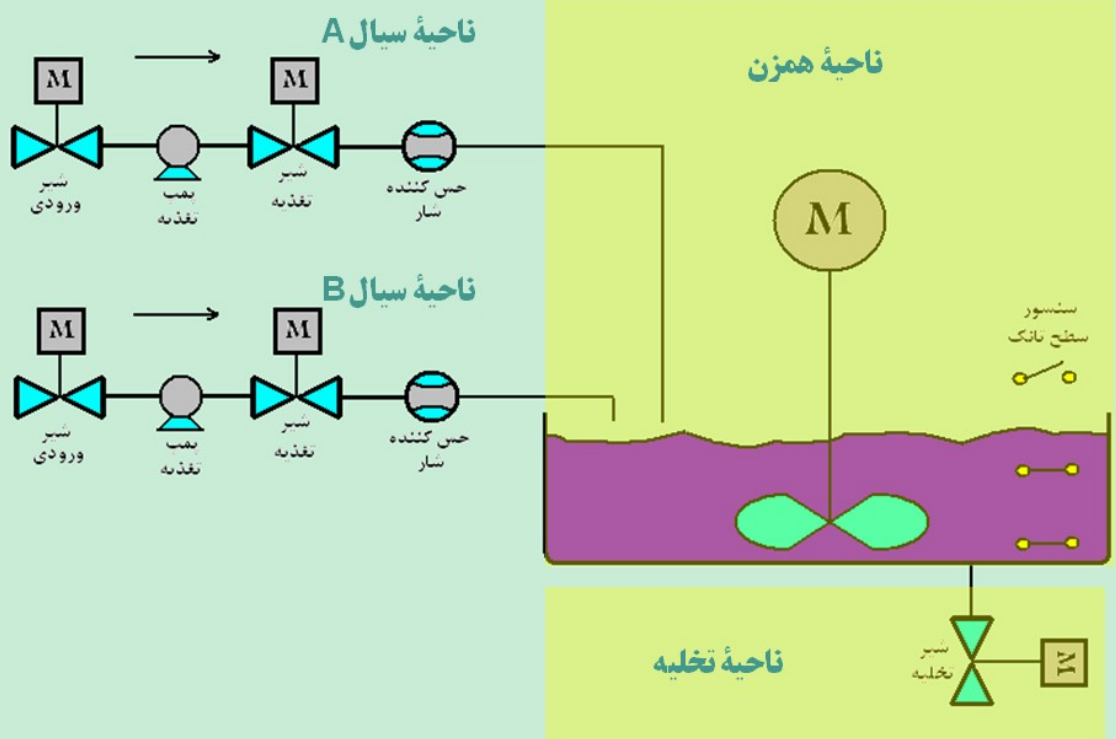
برای مشاهده کامل صفحات از کلید Space استفاده کنید.



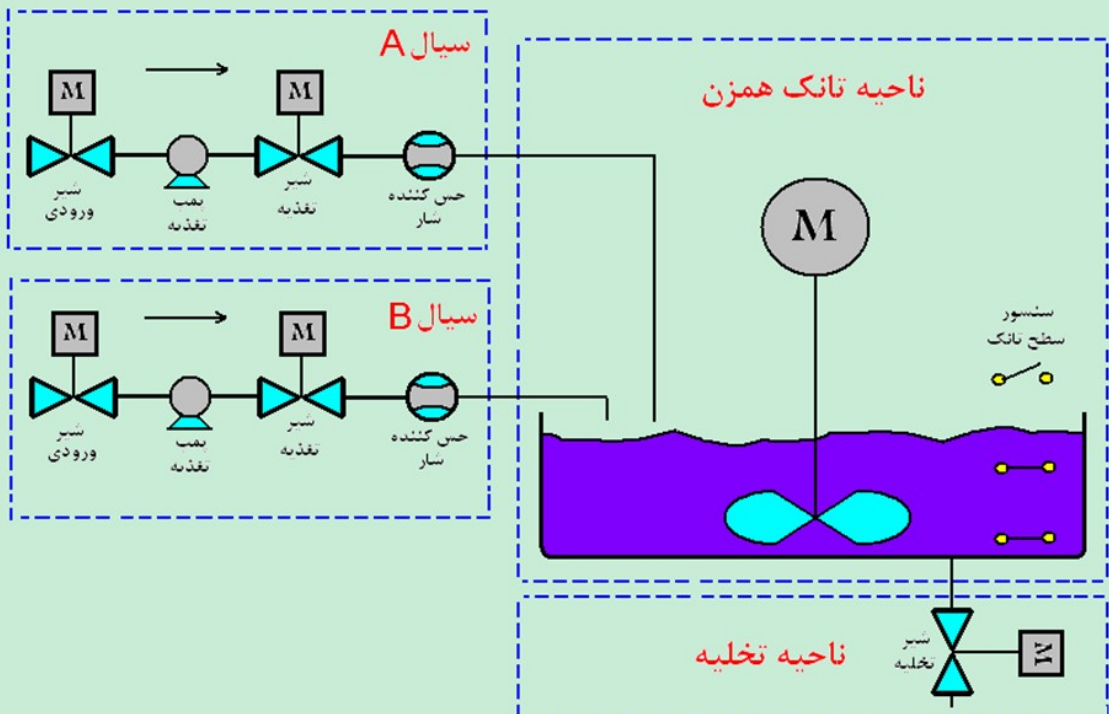
این پروسه به چهار ناحیه زیر تقسیم می شود:

ingredient A	ناحیه "عنصر ترکیبی (سیال) اول"
ingredient B	ناحیه "عنصر ترکیبی (سیال) دوم"
Mixing Tank	ناحیه "مخزن همزن"
Drain	ناحیه تخلیه

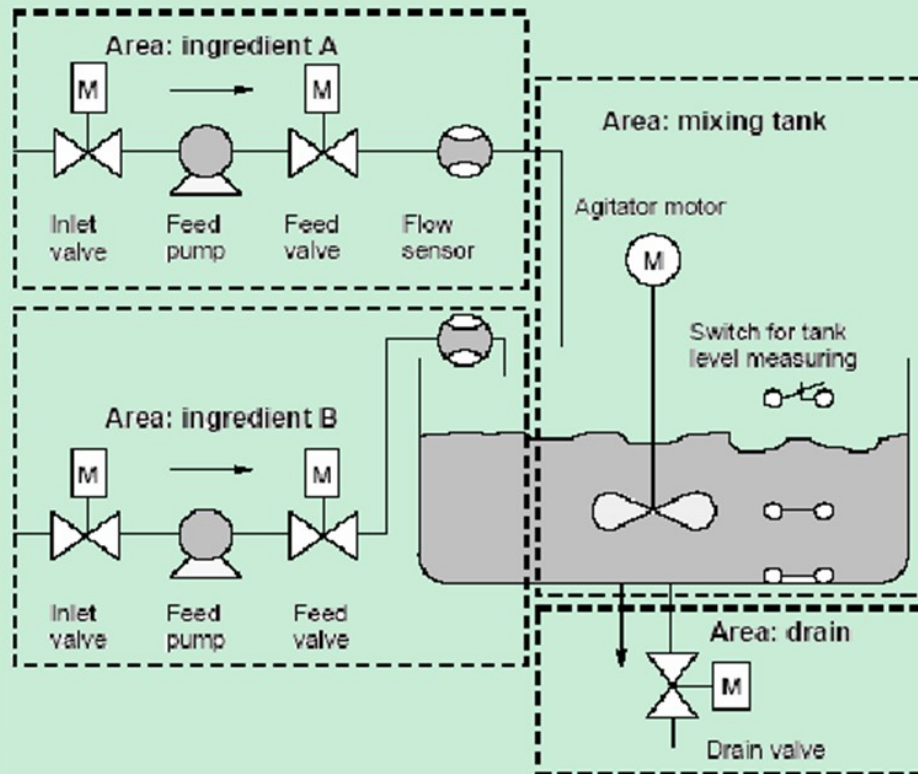




این چهار ناحیه بدین صورت تفکیک می شوند



از این پس از اصطلاحات اصلی (لاتین) به جای اصطلاحات فارسی استفاده می کنیم.



نواحی اصلی و تجهیزات موجود در هر ناحیه

ناحیه	تجهیزات استفاده شده در هر ناحیه
سیال A	- پمپ تغذیه - شیر ورودی - شیر تغذیه - سنسور شار
سیال B	- پمپ تغذیه - شیر ورودی - شیر تغذیه - سنسور شار
همزن	- موتور همزن - کلیدهای اندازه گیری سطح
تخلیه	شیر تخلیه

تشریح نواحی مختلف



تشریح نواحی اصلی

در این بخش باید عملکرد هر ناحیه و همچنین اجزایی که آن ناحیه را کنترل می کنند تشریح شوند.
وسایل الکتریکی، مکانیکی و ورودی-خروجی های لاجیکی شناسایی می شوند.
حالت Interlock یا عدم وابستگی نواحی مختلف تعیین می شوند

پروسه "مخلوط کن صنعتی" از پمپ ها، موتورها و شیرها استفاده می کند.
جداول زیر تجهیزات و حالت های Interlock موجود در این پروسه را تشریح می نماید.
توسط این جداول باید اقدام به سفارش و خرید لوازم و تجهیزات نمود.



ناحیه سیال ها : موتورهای پمپ تغذیه

موتور های پمپ های تغذیه سیال ها را به داخل مخزن هدایت می کنند.

میزان شارش: ۴۰۰ لیتر در دقیقه

قدرت موتورها: ۱۰۰ کیلووات در دور نامی ۱۲۰۰ rpm

پمپ ها توسط کلیدهای start/stop از برد کنترل نزدیک مخزن روشن و خاموش می شوند

تعداد دفعات start به خاطر انجام تعمیر و نگهداری شمرده می شوند. مقدار شمارنده و

نمایشگر آن با یک دکمه روی برد اپراتور صفر می شوند.

شرایط زیر باید برای فعال شدن پمپ ها مهیا شوند:

✓ مخزن پر نباشد.

✓ شیر تخلیه بسته باشد.

✓ کلید اضطراری عمل نکرده باشد.

هنگام وقوع شرایط زیر پمپ ها خاموش می شوند:

✓ ۷ ثانیه پس از راه افتادن پمپ ها سیگنالی از سنسور های شار دریافت نگردد.

✓ سنسور های شار، سیگنالی مبنی بر متوقف شدن شار ارسال کنند.



ناحیه سیال ها : شیرهای ورودی و تغذیه

شیرهای ورودی و تغذیه اجازه ورود یا توقف سیال ها به مخزن را می دهند.

این شیرها دارای سننویید با فنر برگشتی می باشند

وقتی سننویید عمل کند، شیر باز می شود

وقتی سننویید غیر فعال شود، شیر بسته می شود

شیرها ی ورودی و تغذیه توسط برنامه کنترل می شوند.

برای فعال شدن شیرها شرط زیر باید مهیا شود:

✓ موتور پمپ تغذیه حداقل یک ثانیه قبل روشن شده باشد.

هنگام وقوع وضعیت زیر، شیرهای ورودی و تغذیه خاموش می شوند:

✓ سنسور شار سیگنالی ارسال نکند تا از وقوع نشت سیال از پمپ جلوگیری شود.



ناحیه موتور همزن

موتور همزن، سیال A, B را داخل مخزن مخلوط می کند
قدرت موتور: ۱۰۰ کیلووات در دور نامی ۱۲۰۰ rpm

پمپها توسط کلیدهای start/stop از برد کنترل نزدیک مخزن روشن و خاموش می شوند
تعداد دفعات start به خاطر انجام تعمیر و نگهداری شمرده می شوند. مقدار شمارنده و
نمایشگر آن با یک دگمه صفر می شوند.

شرایط زیر باید برای فعال شدن موتور همزن مهیا شوند:

- ✓ سنسور تشخیص سطح سیگنال "مخزن زیر مینیمم مقدار است" صادر نکند
- ✓ شیر تخلیه بسته باشد
- ✓ کلید اضطراری عمل نکرده باشد

هنگام وقوع شرط زیر موتور همزن خاموش می شود:

- ✓ تاکومتر رسیدن موتور به سرعت نامی را در ۱۰ ثانیه (از زمان روشن شدن موتور) اعلام نکند



شیر تخلیه

شیر تخلیه اجازه خروج محصول نهایی را از مخزن صادر می کند. این شیر دارای
سنلوییدی همراه با فنر برگشتی می باشند.
وقتی سنلویید عمل کند، شیر باز می شود
وقتی سنلویید غیر فعال شود، شیر بسته می شود

این شیر توسط کلیدهای start/stop از برد کنترل نزدیک مخزن روشن و خاموش می شوند.

شرایط زیر باید برای فعال شدن شیر تخلیه مهیا شوند:

- ✓ موتور همزن خاموش باشد
- ✓ سنسور تشخیص سطح سیگنال "مخزن خالی است" صادر نکند
- ✓ کلید اضطراری عمل نکرده باشد

هنگام وقوع شرط زیر شیر تخلیه خاموش می شود:

- ✓ سنسور تشخیص سطح سیگنال "مخزن خالی است" صادر کند



کلید های تشخیص سطح مخزن

کلید های داخل مخزن سطح مواد داخل مخزن را نشان می دهد و برای Interlock کردن پمپهای تغذیه و موتور همزن به کار می رود

کلیدهای مخزن حالت های زیر را تشخیص می دهند:

حالت پر بودن تانک : که توسط یک کنتاکت NC تشخیص داده می شود.
هنگامی که سطح مخزن به مقدار ماکزیمم می رسد، کنتاکت باز می شود.

سطح مخزن بالای مینیمم مقدار است: که توسط یک کنتاکت NO تشخیص داده می شود.
وقتی سطح مخزن به مقدار مینیمم خود می رسد، کنتاکت بسته می شود.

تانک خالی نیست: که توسط یک کنتاکت NO تشخیص داده می شود.
وقتی تانک خالی نباشد، کنتاکت بسته می شود.



لیست کردن Input ها، Output ها و In/Out ها

پس از نوشتن توصیف های فیزیکی هر وسیله کنترل شونده نوبت به رسم دیاگرام های ورودی و خروجی هر وسیله می گردد

دیاگرام ورودی-خروجی



ایجاد دیاگرام موتورها

دو پمپ تغذیه و همچنین موتور همزن در این پروسه به علت داشتن شرایط یکسان از یک بلوک با نام "motor block" استفاده می کنند. این بلوک احتیاج به شش ورودی دارد:

- ✓ دو ورودی برای روشن خاموش کردن موتور Start, Stop
- ✓ یک ورودی برای ریست کردن نمایشگر تعمیر و نگهداری
- ✓ یک ورودی برای دریافت سیگنال از تاکومتر جهت تعیین وضعیت در حال کار بودن یا نبودن موتور
- ✓ یک ورودی برای زمانی است که در طی آن باید سیگنال پاسخ دریافت شود
- ✓ یک ورودی برای تعداد تایمرهایی که برای اندازه گیری زمان به کار میروند



این بلوک همچنین احتیاج به چهار خروجی به شرح زیر دارد:

- ✓ دو خروجی جهت نشان دادن روشن یا خاموش بودن موتور
- ✓ یک خروجی برای نمایش خطا
- ✓ یک خروجی برای نمایش حالتی که موتور در حال تعمیر است

این بلوک همچنین احتیاج به یک ورودی- خروجی (In/Out) برای فعال کردن موتور دارد که برای کنترل نمودن موتور به کار می رود ولی در آن واحد در برنامه تغییر و اصلاح می یابد.

دیاگرام I/O برای "Motor Block"



ایجاد دیاگرام I/O برای شیرها

شیرها نیز همانند موتورها از یک بلوک واحد استفاده می کنند
بلوک لاجیکی شیرها دارای دو ورودی یکی برای باز کردن و دیگری برای بستن شیرها می باشد
همچنین دو خروجی برای نمایش وضعیت باز/بسته بودن شیرها تعبیه می شود
این بلوک دارای یک In/Out می باشد که در آن واحد هم برای کنترل شیر
و هم برای تغییر و اصلاح برنامه به کار می ورد



برقراری الزامات ایمنی



برقراری الزامات ایمنی

پیدا کنید کدام يك از وسایل احتیاج به مدارات سخت افزاري اضافي براي ایجاد ایمنی دارند.

این مدارات ایمنی مستقل از PLC عمل می کنند

(اگر چه مدار ایمنی عموماً يك I/O واسطه براي تطابق با برنامه دارد).

پروسه مورد بحث از لاجیک زیر براي مدار ایمنی خود استفاده می کند.

✓ يك کلید اضطراري بسته، مستقل از PLC وسایل زیر را خاموش می کند:

✓ پمپ تغذیه براي سیال A

✓ پمپ تغذیه براي سیال B

✓ موتور همزن

✓ شیرها

این کلید در جلوي اپراتور قرار دارد

يك ورودی در PLC وضعیت این کلید را تشخیص می دهد.



ایجاد وسایل کنترلي
ونمایشگرهاي لازم براي اپراتور

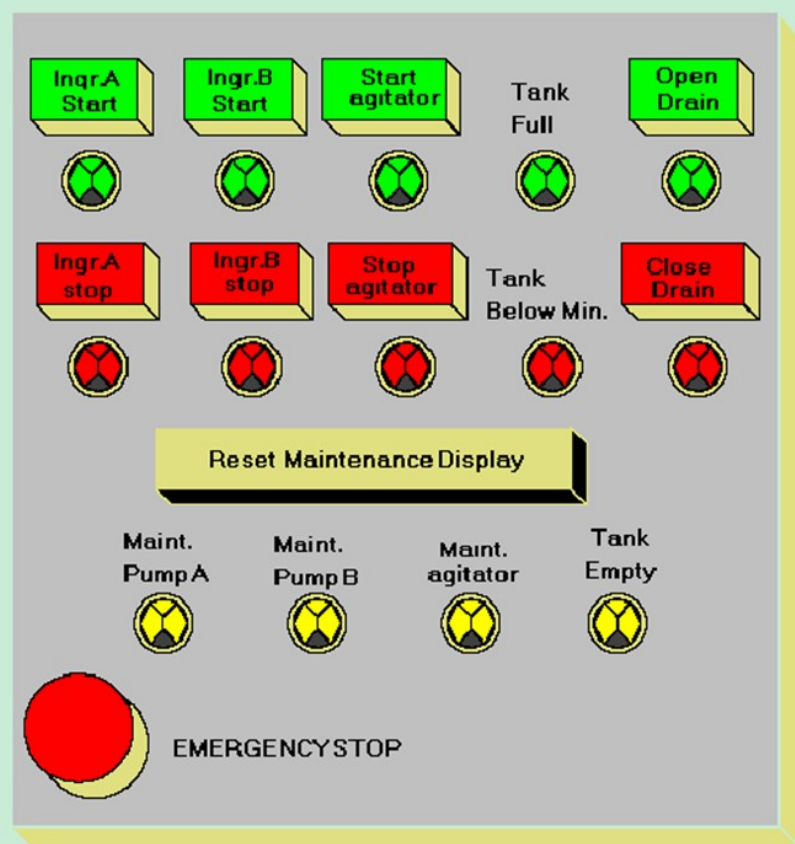


ایجاد وسایل کنترلی و نمایشگرهای لازم برای اپراتور

طراحی برد اپراتور بخش مهمی از انجام پروژه اتوماسیون است که اجازه دخالت انسان در پروسه را می دهد

در پروسه "مخلوط کن صنعتی" هر وسیله می تواند توسط کلیدهای فشاری روی برد، روشن یا خاموش شود. ضمناً وضعیت روشن یا خاموش بودن هر وسیله توسط سیگنال ها قابل رؤیت می باشد. سیگنالهایی نیز برای رؤیت وضعیت "در حال تعمیر و سرویس" هر وسیله و همچنین کلید قطع اضطراری در روی برد تعبیه شده است.

برد اپراتور در پروسه "مخلوط کن صنعتی" با توجه به مطالب بحث شده بدین صورت است.



تشکیل دیاگرام پیکربندی

دیاگرام پیکربندی

سطح بندی بلوکها

ایجاد جدول سمبل ها

ایجاد فانکشن بلاک موتورها

ایجاد دیتا بلاک ها

ایجاد فانکشن برای شیرها

ایجاد OB1

روی قسمت های مختلف کلیک کنید



تشکیل دیاگرام پیکربندی

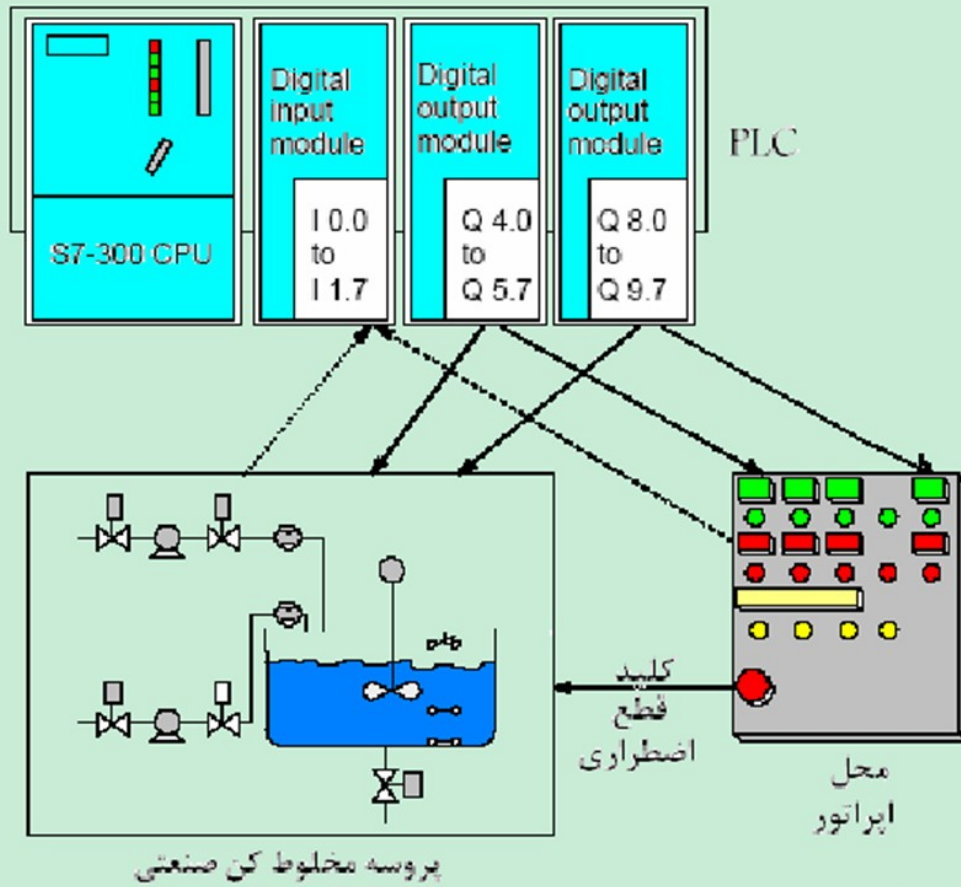
پس از تشخیص و ثبت نیازهای طراحی، باید روی نوع تجهیزات کنترلی تصمیم گیری کرد.
باید نوع ماجول ها و ساختار PLC تعیین گردد.
دیاگرام پیکربندی باید شامل موارد زیر باشد:

نوع CPU

تعداد و نوع ماجول های ورودی و خروجی (I/O)
پیکربندی ورودی و خروجی های فیزیکی

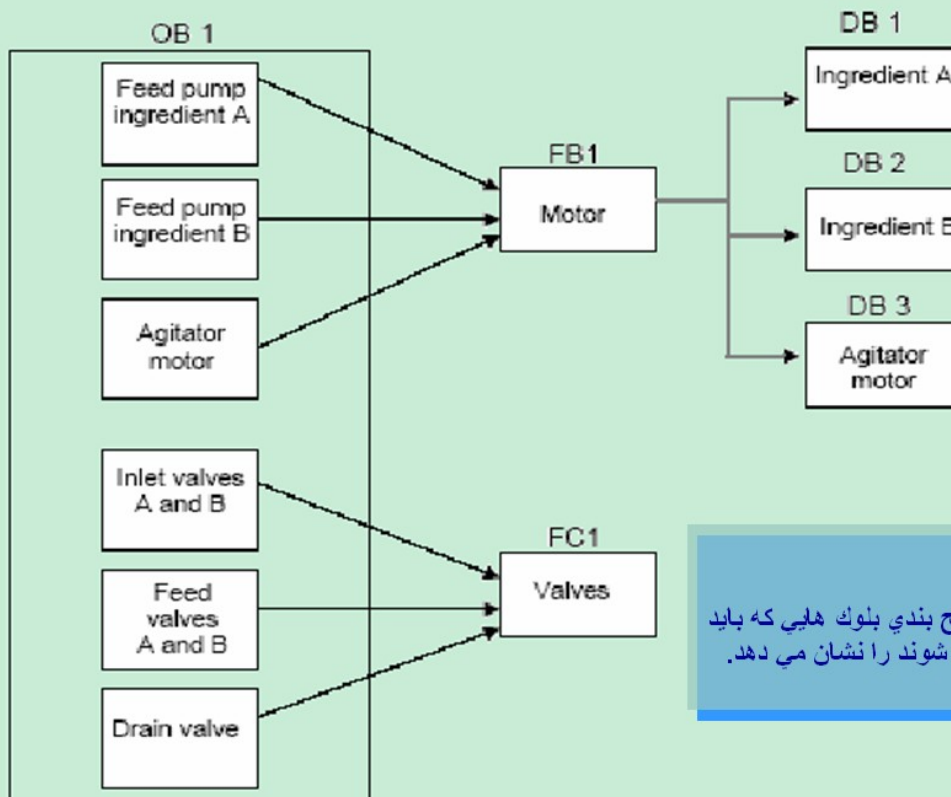
دیاگرام پیکربندی مربوط به پروسه "مخلوط کن صنعتی" که
مثال استفاده از پیکربندی S7 است بدین صورت است.





تشکیل دیاگرام پیکربندی

سطح بندی (Hierarchy) بلوکها



این شکل، سطح بندی بلوک هایی که باید صدا زده شوند را نشان می دهد.

تشکیل دیاگرام پیکربندی

OB1

واسطه بین سیستم عمل CPU بوده و حاوی برنامه اصلی می باشد. در OB1 بلاک های FB1 و FC1 صدا زده شده و پارامتر های مشخصی که برای کنترل پروسه لازم است فرستاده می شود.

FB1

پمپ های تغذیه سیال ها و موتور همزن توسط یک فانکشن بلاک کنترل می شوند، چون نحوه روشن و خاموش کردن، شمارش دفعات روشن شدن و ... یکسان است.

DB 1-3 از نوع Instance:

پمپ های تغذیه سیال ها و موتور همزن توسط یک فانکشن بلاک کنترل می شوند، چون نحوه روشن و خاموش کردن، شمارش دفعات روشن شدن و ... یکسان است.

FC1

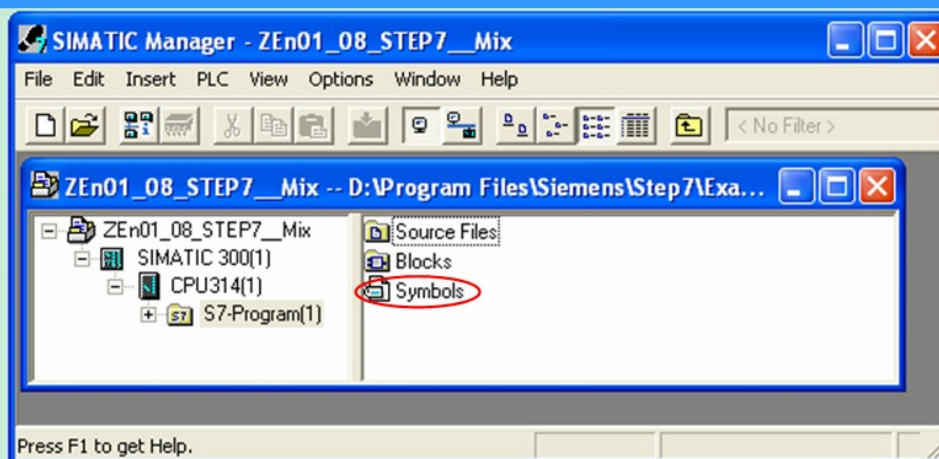
شیرهای ورودی و تغذیه و شیر تخلیه از یک بلاک منطقی مشترک استفاده می کنند. از آنجا که تنها فانکشن "Open And Close" باید برنامه ریزی شوند، برنامه نویسی یک FC کافیست.

تشکیل دیاگرام پیکربندی



ایجاد جدول سمبل ها

در پنجره SIMATIC Manager روی پوشه Symbols دابل کلیک می کنیم تا جدول تعریف سمبل ها باز شود.



مطابق شکل های بعد سمبل هر آدرس را با بیان توضیحات (Comment) تایپ نمایید.

تشکیل دیاگرام پیکربندی



Symbolic Name	Address	Data Type	Description
Feed_pump_A_start	I0.0	BOOL	Starts the feed pump for ingredient A
Feed_pump_A_stop	I0.1	BOOL	Stops the feed pump for ingredient A
Flow_A	I0.2	BOOL	Ingredient A flowing
Inlet_valve_A	Q4.0	BOOL	Activates the inlet valve for ingredient A
Feed_valve_A	Q4.1	BOOL	Activates the feed valve for ingredient A
Feed_pump_A_on	Q4.2	BOOL	Lamp for "feed pump ingredient A running"
Feed_pump_A_off	Q4.3	BOOL	Lamp for "feed pump ingredient A not running"
Feed_pump_A	Q4.4	BOOL	Activates the feed pump for ingredient A
Feed_pump_A_fault	Q4.5	BOOL	Lamp for "feed pump A fault"
Feed_pump_A_maint	Q4.6	BOOL	Lamp for "feed pump A maintenance"
Feed_pump_B_start	I0.3	BOOL	Starts the feed pump for ingredient B
Feed_pump_B_stop	I0.4	BOOL	Stops the feed pump for ingredient B
Flow_B	I0.5	BOOL	Ingredient B flowing
Inlet_valve_B	Q5.0	BOOL	Activates the inlet valve for ingredient A
Feed_valve_B	Q5.1	BOOL	Activates the feed valve for ingredient B
Feed_pump_B_on	Q5.2	BOOL	Lamp for "feed pump ingredient B running"
Feed_pump_B_off	Q5.3	BOOL	Lamp for "feed pump ingredient B not running"
Feed_pump_B	Q5.4	BOOL	Activates the feed pump for ingredient B
Feed_pump_B_fault	Q5.5	BOOL	Lamp for "feed pump B fault"
Feed_pump_B_maint	Q5.6	BOOL	Lamp for "feed pump B maintenance"
Agitator_running	I1.0	BOOL	Response signal of the agitator motor
Agitator_start	I1.1	BOOL	Agitator start button
Agitator_stop	I1.2	BOOL	Agitator stop button
Agitator	Q8.0	BOOL	Activates the agitator
Agitator_on	Q8.1	BOOL	Lamp for "agitator running"
Agitator_off	Q8.2	BOOL	Lamp for "agitator not running"
Agitator_fault	Q8.3	BOOL	Lamp for "agitator motor fault"
Agitator_maint	Q8.4	BOOL	Lamp for "agitator motor maintenance"

تشکیل دیاگرام پیکربندی

آدرس های
سمبلیک پمپ های
تغذیه، موتور همزن و
شیرهای ورودی



Symbolic Name	Address	Data Type	Description
Tank_below_max	I1.3	BOOL	Sensor "mixing tank not full"
Tank_above_min	I1.4	BOOL	Sensor "mixing tank above minimum level"
Tank_not_empty	I1.5	BOOL	Sensor "mixing tank not empty"
Tank_max_disp	Q9.0	BOOL	Lamp for "mixing tank full"
Tank_min_disp	Q9.1	BOOL	Lamp for "mixing tank below minimum level"
Tank_empty_disp	Q9.2	BOOL	Lamp for "mixing tank empty"

Symbolic Name	Address	Data Type	Description
Drain_open	I0.6	BOOL	Button for opening the drain valve
Drain_closed	I0.7	BOOL	Button for closing the drain valve
Drain	Q9.5	BOOL	Activates the drain valve
Drain_open_disp	Q9.6	BOOL	Lamp for "drain valve open"
Drain_closed_disp	Q9.7	BOOL	Lamp for "drain valve closed"

Symbolic Name	Address	Data Type	Description
EMER_STOP_off	I1.6	BOOL	EMERGENCY STOP switch
Reset_maint	I1.7	BOOL	Reset switch for the maintenance lamps on all motors
Motor_block	FB1	FB1	FB for controlling pumps and motor
Valve_block	FC1	FC1	FC for controlling the valves
DB_feed_pump_A	DB1	FB1	Instance DB for controlling feed pump A
DB_feed_pump_B	DB2	FB1	Instance DB for controlling feed pump B
DB_agitator	DB3	FB1	Instance DB for controlling the agitator motor

تشکیل دیاگرام پیکربندی

آدرس های سمبلیک
سنسورها و
نشانگرهای سطح

آدرس های سمبلیک
شیر تخلیه

آدرس های سمبلیک
سایر المان های
برنامه



ایجاد فانکشن بلاک موتورها

چه چیزهایی برای FB نیاز است؟

FB مربوط به موتور دارای توابع زیر است:

یک ورودی Start و یک ورودی Stop ✓

یک سری حالت های Interlock به تجهیزات (پمپ ها و موتور همزن) اجازه کار می دهند. ✓

حالت های Interlock در اطلاعات محلی موقتی (L Stack) مربوط به OB1 ذخیره می شوند:
("Valve_enable", "Motor_enable")

و هنگامی که FB موتور پردازش می شود، به صورت منطقی با ورودی های Start, Stop ترکیب می شوند.

✓ Feedback های تجهیزات در محدوده زمانی مشخصی باید ظاهر شوند. در غیر این صورت فرض می شود که اشکال یا خطایی رخ داده است. سپس Function موتور را متوقف می کند.

✓ اگر دکمه Start زده شود وسیله روشن شده و تا زدن دکمه Stop روشن باقی می ماند.

✓ وقتی وسیله روشن شود، یک تایمر شروع به کار می کند. اگر سیگنال پاسخ قبل از سپری شدن زمان تایمر از وسیله دریافت نشود، وسیله متوقف می شود.

تشکیل دیاگرام پیکربندی



مشخص کردن ورودی-خروجی های FB1

همان طور که قبلاً ذکر شود ورودی و خروجی های فانکشن بلاک موتورها به صورت زیر است:



تشکیل دیاگرام پیکربندی



معرفی پارامترهای FB1

اگر يك FB از نوع Multiple Instance براي موتور استفاده مي كنيد، بايد نام هاي پارامترهاي عمومي را براي وروديتها و خروجي ها تعريف كنيد.
FB مربوط به موتور در پروسه ساده احتياج به موارد زير دارد:

✓ از برد ايراتور سيگنالهايي براي خاموش روشن کردن موتور و پمپ ها داشته باشد.

✓ احتياج به سيگنال پاسخ پمپ ها و موتور براي اعلام در حال كار بودن داشته باشد.

✓ بايد زمان بين ارسال سيگنال براي روشن کردن موتور و دريافت سيگنال پاسخ را محاسبه كند.
اگر در طي اين زمان سيگنال پاسخي دريافت نشد بايد موتور را خاموش كند.

✓ بايد لامپ هاي برد ايراتور را روشن و خاموش كند.

✓ بايد يك سيگنال براي فعال نمودن موتور فراهم نمايد.

تشكيل دياگرام پيكربندي



اين ملزومات را مي توان به عنوان ورودي و خروجي هاي FB تعيين مي شوند.
جدول زير پارامترهاي موتور را نشان مي دهد.

Parameter Name	Input	Output	In/Out
Start	n		
Stop	n		
Response	n		
Reset_maint	n		
Timer_No	n		
Response_Time	n		
Fault		n	
Start_Dsp		n	
Stop_Dsp		n	
Maint		n	
Motor			n

تشكيل دياگرام پيكربندي



Contents Of: 'Environment\Interface'	
Interface	Name
IN	IN
OUT	OUT
IN_OUT	IN_OUT
STAT	STAT
TEMP	TEMP

باید پارامترهای Input و Output و In/Out را به FB مربوط به موتور بشناسانید.
بنابراین پس از ایجاد FB1، آن را باز کرده و در پنجره Declaration (شکل مقابل) پارامترها را مطابق جدول زیر وارد می کنیم.

Address	Declaration	Name	Type	Initial Value
0.0	IN	Start	BOOL	FALSE
0.1	IN	Stop	BOOL	FALSE
0.2	IN	Response	BOOL	FALSE
0.3	IN	Reset_Maint	BOOL	FALSE
2.0	IN	Timer_No	TIMER	
4.0	IN	Response_Time	S5TIME	S5T#0MS
6.0	OUT	Fault	BOOL	FALSE
6.1	OUT	Start_Dsp	BOOL	FALSE
6.2	OUT	Stop_Dsp	BOOL	FALSE
6.3	OUT	Maint	BOOL	FALSE
8.0	IN_OUT	Motor	BOOL	FALSE
10.0	STAT	Time_bin	WORD	W#16#0
12.0	STAT	Time_BCD	WORD	W#16#0
14.0	STAT	Starts	INT	0
16.0	STAT	Start_Edge	BOOL	FALSE

تشکیل دیاگرام پیکربندی

برنامه نویسی FB1

چون FB1 توسط OB1 صدا زده می شود، بنابراین باید اول FB1 را برنامه نویسی نماییم.

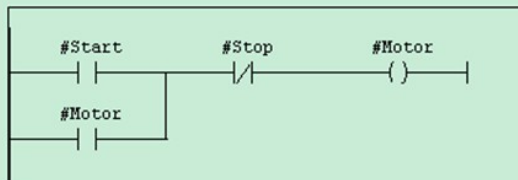
روشن خاموش کردن موتور

Network 1

```

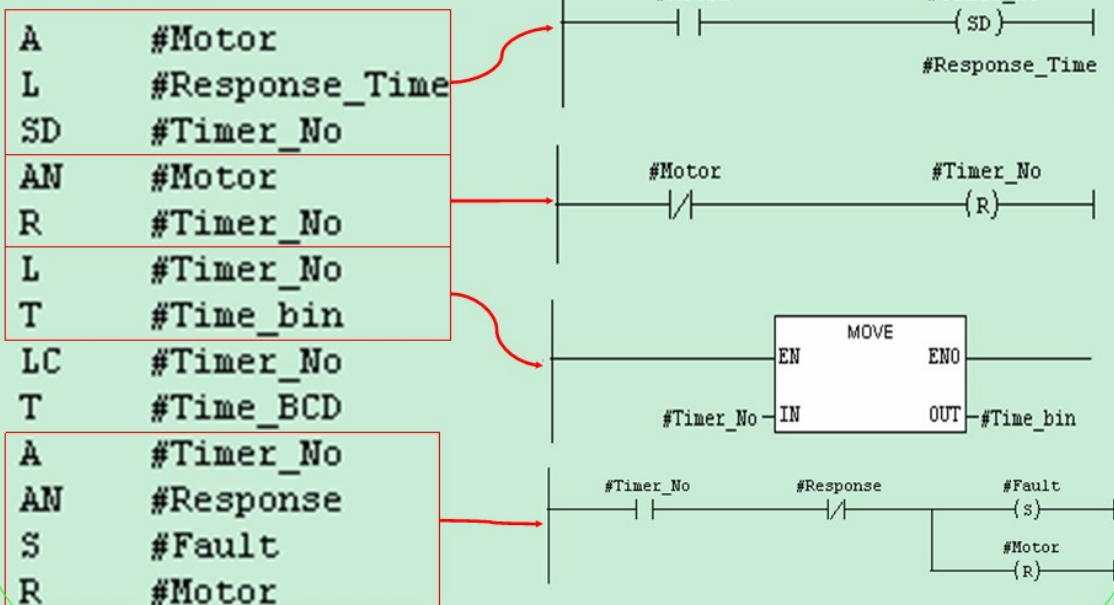
A(
0   #Start
0   #Motor
)
AN  #Stop
=   #Motor

```



تشکیل دیاگرام پیکربندی

Network 2 کنترل وضعیت شروع به کار



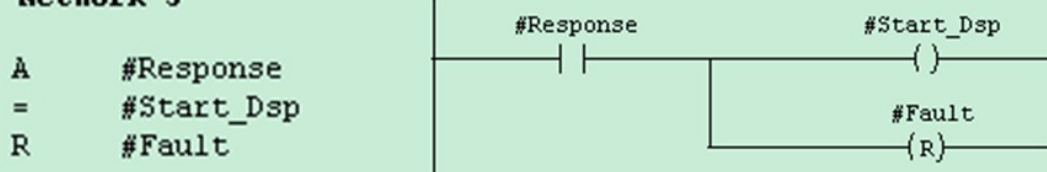
تایمر SD با روشن شدن موتور شروع به کار میکند و در پس از سپری شدن زمان تنظیمی (Response_Time) و در صورتی که هنوز موتور روشن باشد، خروجی آن (Timer_No) یک می شود.

تشکیل دیاگرام پیکربندی



روشن شدن لامپ و ریست شدن خطا

Network 3



Network 4

خاموش کردن لامپ



تشکیل دیاگرام پیکربندی



Network 5 شمارش تعداد دفعات Start

```

A      #Motor
FP     #Start_Edge
JCN    lab1
L      #Starts
+      1
T      #Starts
lab1: NOP 0

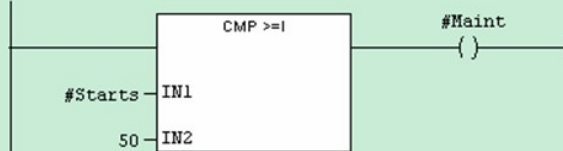
```

Network 6 تصویر و نگهداری

```

L      #Starts
L      50
>=I
=      #Maint

```



Network 7 ریست کردن تصویر و نگهداری

```

A      #Reset_Maint
A      #Maint
JCN    END
L      0
T      #Starts
END: NOP 0

```

تشکیل دیاگرام پیکربندی



ایجاد دیتا بلاک ها (DBها)

سه دیتا بلاک با نام های DB1, DB2, DB3 ایجاد کرده و آنها را یکی پس از دیگری باز کنید. مطابق شکل گزینه "Instance DB" را انتخاب کنید. از لیست، FB1 را انتخاب کنید.

تشکیل دیاگرام پیکربندی



ایجاد فانکشن بلاک برای شیرها

فانکشن شیرهای ورودی و تغذیه و همچنین شیر تخلیه دارای توابع زیر هستند:
 یک ورودی برای باز کردن و یک ورودی برای بستن شیرها وجود دارد.
 حالت **Interlock** یا قفل داخلی اجازه باز شدن شیرها را می دهد.
 حالت های **Interlock** در اطلاعات محلی موقتی (**L Stack**) مربوط به **OB1** ذخیره می شوند:
 ("Valve_enable")
 و هنگامی که **FB** موتور پردازش می شود، به صورت منطقی با ورودی های **Start**, **Stop** ترکیب می شوند.

جدول زیر پارامترهایی را که باید به **FC** انتقال یابد را نشان می دهد.

Parameters for the Valves	Input	Output	In/Out
Open	✓		
Close	✓		
Dsp_Open		✓	
Dsp_Closed		✓	
Valve			✓

تشکیل دیاگرام پیکربندی



مشخص کردن ورودی-خروجی های FC

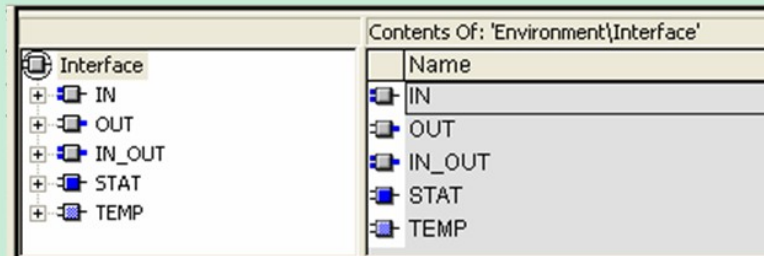
همان طور که قبلاً ذکر شود ورودی و خروجی های فانکشن شیرها به صورت زیر است:



تشکیل دیاگرام پیکربندی



باید پارامترهای Input و Output و In/Out را به FC مربوط به شیرها شناسانید.
بنابراین پس از ایجاد FC، آن را باز کرده و در پنجره Declaration پارامترها را مطابق جدول زیر وارد می‌کنیم.



Address	Declaration	Name	Type	Initial Value
0.0	IN	Open	BOOL	FALSE
0.1	IN	Close	BOOL	FALSE
2.0	OUT	Dsp_Open	BOOL	FALSE
2.1	OUT	Dsp_Closed	BOOL	FALSE
4.0	IN_OUT	Valve	BOOL	FALSE

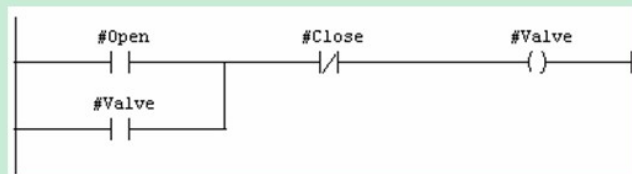
تشکیل دیاگرام پیکربندی

برنامه نویسی FC

چون FC نیز توسط OB1 صدا زده می‌شود، بنابراین باید اول FC را برنامه نویسی نماییم.

Network 1 Open/close and latching

```
A(
O  #Open
O  #Valve
)
AN #Close
=  #Valve
```



Network 2 Display "valve open"

```
A  #Valve
=  #Dsp_Open
```



Network 3 Display "valve closed"

```
AN #Valve
=  #Dsp_Closed
```



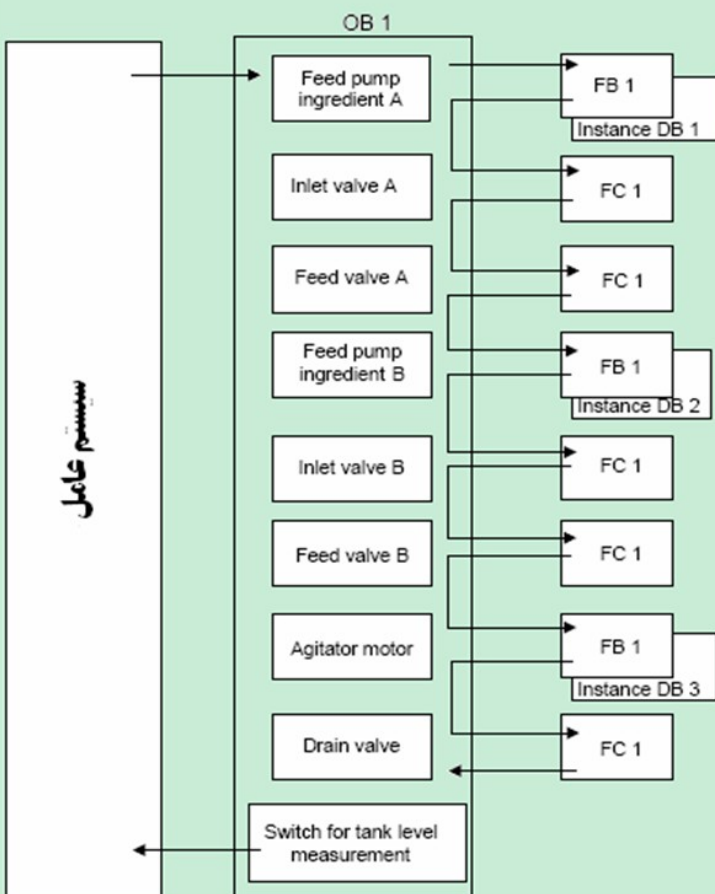
تشکیل دیاگرام پیکربندی

ایجاد OB1

Address	Declaration	Name	Type
0.0	TEMP	OB1_EV_CLASS	BYTE
1.0	TEMP	OB1_SCAN1	BYTE
2.0	TEMP	OB1_PRIORITY	BYTE
3.0	TEMP	OB1_OB_NUMBR	BYTE
4.0	TEMP	OB1_RESERVED_1	BYTE
5.0	TEMP	OB1_RESERVED_2	BYTE
6.0	TEMP	OB1_PREV_CYCLE	INT
8.0	TEMP	OB1_MIN_CYCLE	INT
10.0	TEMP	OB1_MAX_CYCLE	INT
12.0	TEMP	OB1_DATE_TIME	DATE_AND_TIME
20.0	TEMP	Enable_motor	BOOL
20.1	TEMP	Enable_valve	BOOL
20.2	TEMP	Start_fulfilled	BOOL
20.3	TEMP	Stop_fulfilled	BOOL
20.4	TEMP	Inlet_valve_A_open	BOOL
20.5	TEMP	Inlet_valve_A_closed	BOOL
20.6	TEMP	Feed_valve_A_open	BOOL
20.7	TEMP	Feed_valve_A_closed	BOOL
21.0	TEMP	Inlet_valve_B_open	BOOL
21.1	TEMP	Inlet_valve_B_closed	BOOL
21.2	TEMP	Feed_valve_B_open	BOOL
21.3	TEMP	Feed_valve_B_closed	BOOL
21.4	TEMP	Open_drain	BOOL
21.5	TEMP	Close_drain	BOOL
21.6	TEMP	Valve_closed_fulfilled	BOOL

جدول معرفی متغیرهای
OB1 به صورت زیر
است. 20 بایت اول
حاوی اطلاعات
شروع به کار می
باشد و نباید تغییر
یابد.

تشکیل دیاگرام پیکربندی



توجه داشته باشید چون
OB1 توسط FC1 و FB1
صدا زده می شوند، باید
ابتدا این دو بلاک برنامه ریزی شوند.

بلاک های
FB1 و FC1 بیش از یک بار توسط OB1
صدا زده می شوند.
FB1 با دیتابلاک های Instance
مختلف صدا زده می شود.

تشکیل دیاگرام پیکربندی



Network 1 Interlocks for feed pump A

```

A      "EMER_STOP_off"
A      "Tank_below_max"
AN     "Drain"
=       #Enable_Motor

```

Network 2 Calling FB Motor for ingredient A

```

A      "Feed_pump_A_start"
A      #Enable_Motor
=       #Start_Fulfilled
A(
O      "Feed_pump_A_stop"
ON     #Enable_Motor
)
=       #Stop_Fulfilled
CALL   "Motor_block", "DB_feed_pump_A"
Start  := #Start_Fulfilled
Stop   := #Stop_Fulfilled
Response := "Flow_A"
Reset_Maint := "Reset_maint"
Timer_No := T12
Reponse_Time:=S5T#7S
Fault  := "Feed_pump_A_fault"
Start_Dsp := "Feed_pump_A_on"
Stop_Dsp := "Feed_pump_A_off"
Maint  := "Feed_pump_A_maint"
Motor  := "Feed_pump_A"

```

تشکیل دیاگرام پیکربندی



Network 1 Interlocks for feed pump A

```

A      "EMER_STOP_off"
A      "Tank_below_max"
AN     "Drain"
=       #Enable_Motor

```

Network 2 Calling FB Motor for ingredient A

```

A      "Feed_pump_A_start"
A      #Enable_Motor
=       #Start_Fulfilled
A(
O      "Feed_pump_A_stop"
ON     #Enable_Motor
)
=       #Stop_Fulfilled
CALL   "Motor_block", "DB_feed_pump_A"
Start  := #Start_Fulfilled
Stop   := #Stop_Fulfilled
Response := "Flow_A"
Reset_Maint := "Reset_maint"
Timer_No := T12
Reponse_Time:=S5T#7S
Fault  := "Feed_pump_A_fault"
Start_Dsp := "Feed_pump_A_on"
Stop_Dsp := "Feed_pump_A_off"
Maint  := "Feed_pump_A_maint"
Motor  := "Feed_pump_A"

```

تشکیل دیاگرام پیکربندی



Network 6 Interlocks for feed pump B

```

A      "EMER_STOP_off"
A      "Tank_below_max"
AN     "Drain"
=      "Enable_Motor"

```

Network 7 Calling FB Motor for ingredient B

```

A      "Feed_pump_B_start"
A      #Enable_Motor
=      #Start_Fulfilled
A(
O      "Feed_pump_B_stop"
ON     #Enable_Motor
)
=      #Stop_Fulfilled
CALL   "Motor_block", "DB_feed_pump_B"
Start  := #Start_Fulfilled
Stop   := #Stop_Fulfilled
Response := "Flow_B"
Reset_Maint := "Reset_maint"
Timer_No := T14
Reponse_Time := S5T#7S
Fault   := "Feed_pump_B_fault"
Start_Dsp := "Feed_pump_B_on"
Stop_Dsp := "Feed_pump_B_off"
Maint   := "Feed_pump_B_maint"
Motor   := "Feed_pump_B"

```

تشکیل دیاگرام پیکربندی

**Network 8 Delaying the valve enable ingredient B**

```

A      "Feed_pump_B"
L      S5T#1S
SD     T      15
AN     "Feed_pump_B"
R      T      15
A      T      15
=      #Enable_Valve

```

Network 9 Inlet valve control for ingredient B

```

AN     "Flow_B"
AN     "Feed_pump_B"
=      #Close_Valve_Fulfilled
CALL   "Valve_block"
Open   := #Enable_Valve
Close  := #Close_Valve_Fulfilled
Dsp_Open := #Inlet_Valve_B_Open
Dsp_Closed := #Inlet_Valve_B_Closed
Valve  := "Inlet_Valve_B"

```

Network 10 Feed valve control for ingredient B

```

AN     "Flow_B"
AN     "Feed_pump_B"
=      #Close_Valve_Fulfilled
CALL   "Valve_block"
Open   := #Enable_Valve
Close  := #Close_Valve_Fulfilled
Dsp_Open := #Feed_Valve_B_Open
Dsp_Closed := #Feed_Valve_B_Closed
Valve  := "Feed_Valve_B"

```

تشکیل دیاگرام پیکربندی



Network 11 Interlocks for agitator

```

A      "EMER_STOP_off"
A      "Tank_above_min"
AN     "Drain"
=      #Enable_Motor

```

Network 12 Calling FB Motor for agitator

```

A      "Agitator_start"
A      #Enable_Motor
=      #Start_Fulfilled
A(
O      "Agitator_stop"
ON     #Enable_Motor
)
=      #Stop_Fulfilled
CALL  "Motor_block", "DB_Agitator"
      Start   := #Start_Fulfilled
      Stop    := #Stop_Fulfilled
      Response := "Agitator_running"
      Reset_Maint := "Reset_maint"
      Timer_No := T16
      Reponse_Time := S5T#10S
      Fault    := "Agitator_fault"
      Start_Dsp := "Agitator_on"
      Stop_Dsp  := "Agitator_off"
      Maint     := "Agitator_maint"
      Motor     := "Agitator"

```

تشکیل دیاگرام پیکربندی

**Network 13 Interlocks for drain valve**

```

A      "EMER_STOP_off"
A      "Tank_not_empty"
AN     "Agitator"
=      "Enable_Valve"

```

Network 14 Drain valve control

```

A      "Drain_open"
A      #Enable_Valve
=      #Open_Drain
A(
O      "Drain_closed"
ON     #Enable_Valve
)
=      #Close_Drain
CALL  "Valve_block"
      Open   := #Open_Drain
      Close  := #Close_Drain
      Dsp_Open := "Drain_open_disp"
      Dsp_Closed := "Drain_closed_disp"
      Valve   := "Drain"

```

تشکیل دیاگرام پیکربندی



Network 15 Tank level display

```

AN    "Tank_below_max"
=      "Tank_max_disp"
AN    "Tank_above_min"
=      "Tank_min_disp"
AN    "Tank_not_empty"
=      "Tank_empty_disp"
    
```

