

SIEMENS

常问问题 • 02/2015

通过 SIMOTION 控制电源模块启动

SIMOTION, ALM, SLM, BLM

<https://support.industry.siemens.com/cs/cn/zh/view/109477022>

目录

| | | |
|----------|----------------|----------|
| 1 | 概述..... | 3 |
| 1.1 | 模块介绍..... | 3 |
| 1.2 | 程序介绍..... | 3 |
| 2 | 配置..... | 4 |
| 2.1 | 操作步骤..... | 4 |

1 概述

1.1 模块介绍

带有 DRIVE-CLiQ 接口的电源模块，如 ALM (Active Line Modules)、BLM (Basic Line Modules)、SLM (Smart Line Modules)，可以通过 SIMOTION 的程序控制其启动和停止。电源模块可以连接在 SIMOTION 内置的 SINAMICS_Integrated 上,也可以连接到 Profibus 或者 Profinet 扩展的 S120 上。

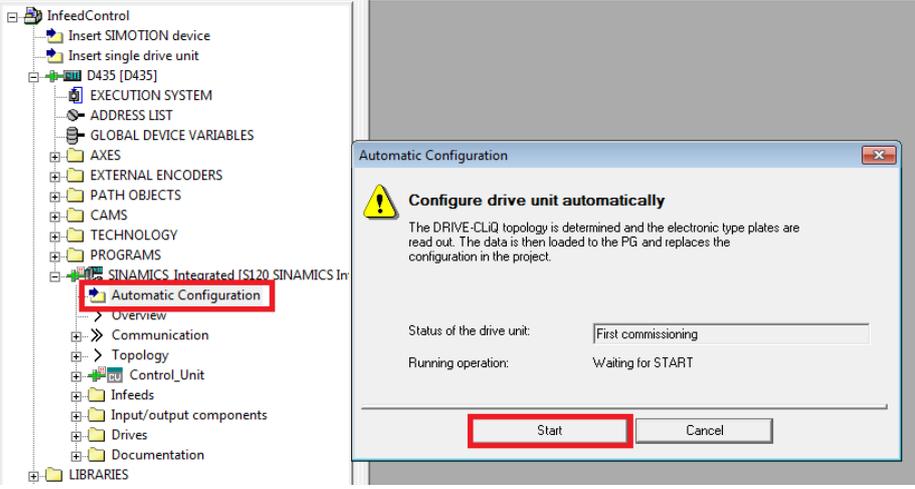
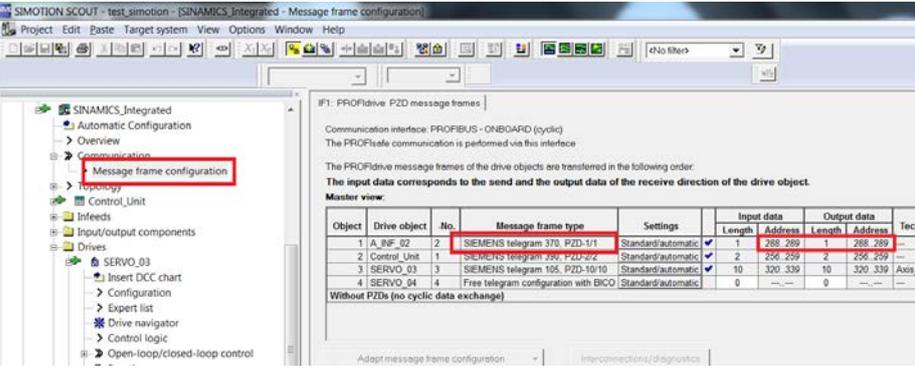
1.2 程序介绍

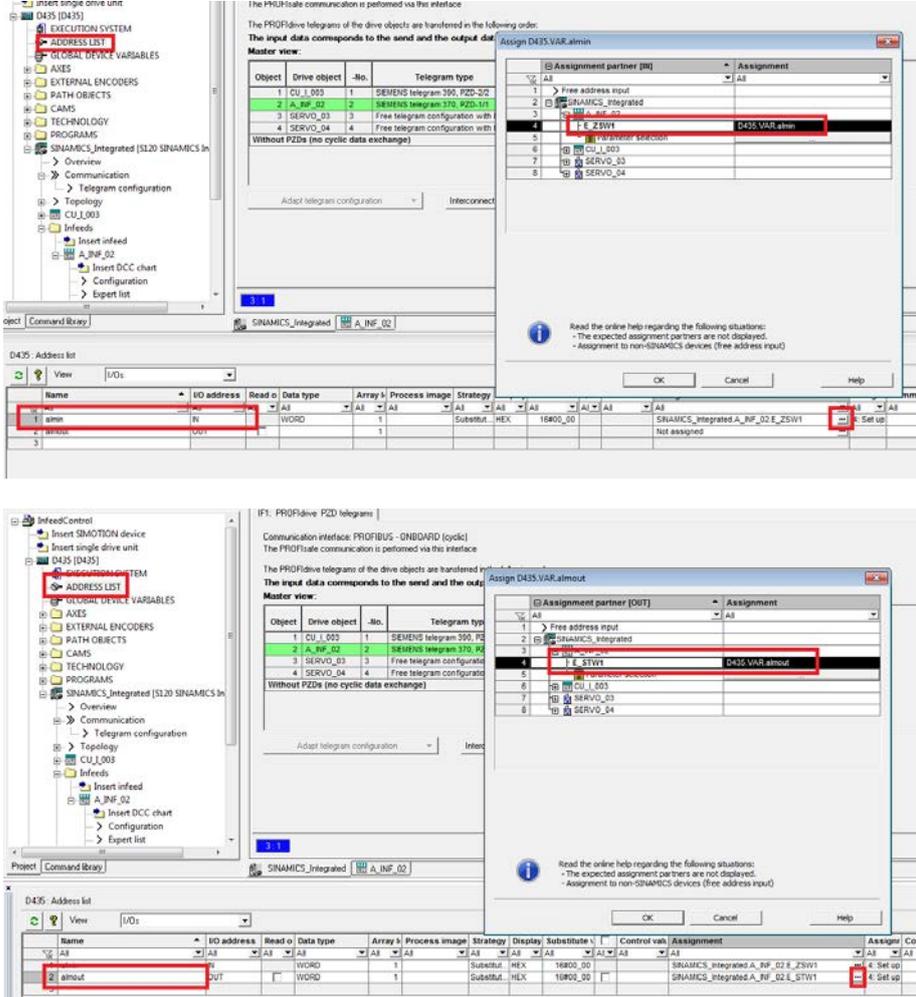
SIMOTION 通过使用 LineModule_control 程序块可以控制带有 DRIVE-CLiQ 接口的电源模块，同时也可以进行模块的类型读取和显示。对于被控制的电源模块必须为其组态 SIEMENS message frame 370 (370 报文)。

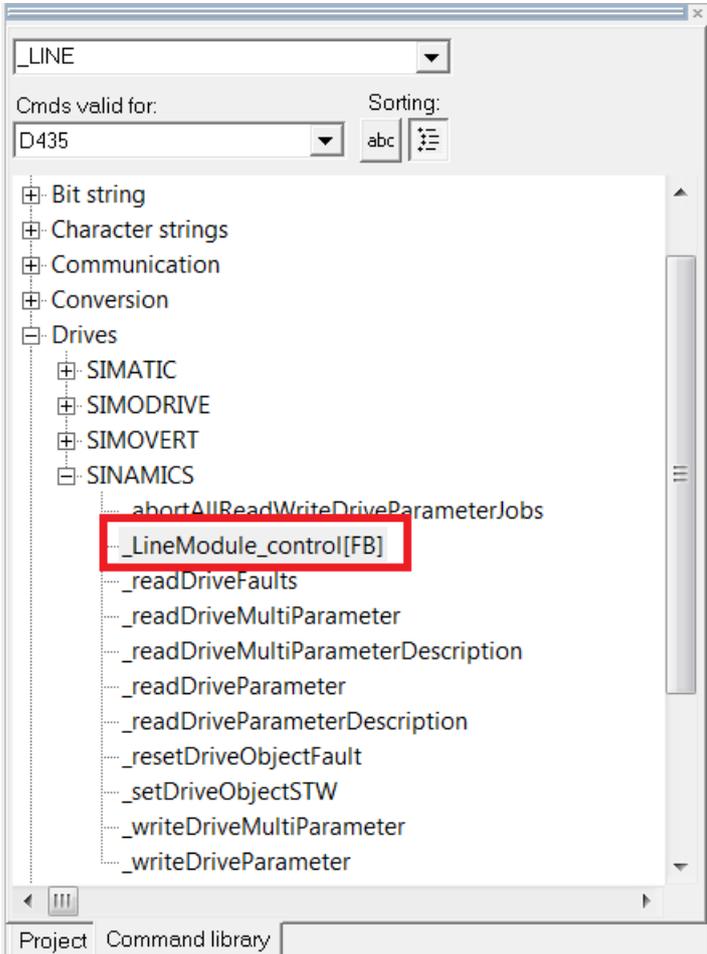
2 配置

2.1 操作步骤

以下步骤针对的是 SIMOTION 控制连接在内置的 SINAMICS_Integrated 上的电源模块，其他连接方式与此类似。

| 序号 | 图示与说明 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|--------|---|--------|---------------------------------------|--------------------|--------------------|--------------------|--------------------|---------------------|--------------------|---------------------|--------|---|----------|---|-------------------------------|--------------------|---|----------|---|----------|-----|---|--------------|---|-------------------------------|--------------------|---|----------|---|----------|-----|---|----------|---|---------------------------------|--------------------|----|----------|----|----------|--------|---|----------|---|---------------------------------------|--------------------|---|-----|---|-----|-----|
| 1. | <p>打开 SCOUT 软件，对 SIMOTION 进行基本的配置，并且对集成的 SINAMICS_Integrated 进行自动配置：</p>  | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 2. | <p>保存编译离线项目后，记录 370 报文的输入、输出的起始地址：</p>  <table border="1" data-bbox="778 1451 1380 1541"> <thead> <tr> <th>Object</th> <th>Drive object</th> <th>No.</th> <th>Message frame type</th> <th>Settings</th> <th>Input data Length</th> <th>Input data Address</th> <th>Output data Length</th> <th>Output data Address</th> <th>Techno</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>A_INF_02</td> <td>2</td> <td>SIEMENS telegram 370, PZD-1/1</td> <td>Standard/automatic</td> <td>1</td> <td>288, 289</td> <td>1</td> <td>288, 289</td> <td>---</td> </tr> <tr> <td>2</td> <td>Control_Unit</td> <td>1</td> <td>SIEMENS telegram 350, PZD-2/2</td> <td>Standard/automatic</td> <td>2</td> <td>250, 259</td> <td>2</td> <td>250, 259</td> <td>---</td> </tr> <tr> <td>3</td> <td>SERVO_03</td> <td>3</td> <td>SIEMENS telegram 105, PZD-10/10</td> <td>Standard/automatic</td> <td>10</td> <td>320, 339</td> <td>10</td> <td>320, 339</td> <td>Axis 1</td> </tr> <tr> <td>4</td> <td>SERVO_04</td> <td>4</td> <td>Free telegram configuration with DICO</td> <td>Standard/automatic</td> <td>0</td> <td>---</td> <td>0</td> <td>---</td> <td>---</td> </tr> </tbody> </table> | Object | Drive object | No. | Message frame type | Settings | Input data Length | Input data Address | Output data Length | Output data Address | Techno | 1 | A_INF_02 | 2 | SIEMENS telegram 370, PZD-1/1 | Standard/automatic | 1 | 288, 289 | 1 | 288, 289 | --- | 2 | Control_Unit | 1 | SIEMENS telegram 350, PZD-2/2 | Standard/automatic | 2 | 250, 259 | 2 | 250, 259 | --- | 3 | SERVO_03 | 3 | SIEMENS telegram 105, PZD-10/10 | Standard/automatic | 10 | 320, 339 | 10 | 320, 339 | Axis 1 | 4 | SERVO_04 | 4 | Free telegram configuration with DICO | Standard/automatic | 0 | --- | 0 | --- | --- |
| Object | Drive object | No. | Message frame type | Settings | Input data Length | Input data Address | Output data Length | Output data Address | Techno | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 1 | A_INF_02 | 2 | SIEMENS telegram 370, PZD-1/1 | Standard/automatic | 1 | 288, 289 | 1 | 288, 289 | --- | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 2 | Control_Unit | 1 | SIEMENS telegram 350, PZD-2/2 | Standard/automatic | 2 | 250, 259 | 2 | 250, 259 | --- | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 3 | SERVO_03 | 3 | SIEMENS telegram 105, PZD-10/10 | Standard/automatic | 10 | 320, 339 | 10 | 320, 339 | Axis 1 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 4 | SERVO_04 | 4 | Free telegram configuration with DICO | Standard/automatic | 0 | --- | 0 | --- | --- | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

| 序号 | 图示与说明 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|---------|---|--------|------------|---------|---------------|----------|---------------|--------------|-------------|-----------------------------------|-------------|------------|---------|-----|-------|----|----|------|---|-----------|-----|----------|--|--|-----------------------------------|-----------|--|------|------------|--------|-----------|---------|---------------|----------|---------|--------------|-------------|------------|---------|-----|---------|----|----|------|---|-----------|-----|----------|--|--|-----------------------------------|-----------|--|
| 3. | <p>随后在 SIMOTION 的地址表中建立分配下图地址变量：almin 和 almount，在 assignment 中选择控制字和状态字：</p>  <p>The screenshot displays the SIMATIC Manager interface for configuring the D435 drive unit. The 'ADDRESS LIST' is visible, showing the assignment of variables 'almin' and 'almount'. The 'Assignment' dialog boxes are shown for both variables, with the 'Assignment partner' set to 'D435:VAR.almin' and 'D435:VAR.almount' respectively. The 'almin' variable is assigned to the 'ZSWI' telegram, and the 'almount' variable is assigned to the 'STW1' telegram.</p> <table border="1" data-bbox="448 795 1364 884"> <caption>D435: Address list (almin)</caption> <thead> <tr> <th>Name</th> <th>IO address</th> <th>Read o</th> <th>Data type</th> <th>Array s</th> <th>Process image</th> <th>Strategy</th> <th>Display</th> <th>Substitute v</th> <th>Control var</th> <th>Assignment</th> <th>Assignm</th> <th>Com</th> </tr> </thead> <tbody> <tr> <td>almin</td> <td>16</td> <td>AI</td> <td>WORD</td> <td>1</td> <td>Substitut</td> <td>HEX</td> <td>16R00_00</td> <td></td> <td></td> <td>SINAMICS_Integrated_A_INF_02_ZSWI</td> <td>4. Set up</td> <td></td> </tr> </tbody> </table> <table border="1" data-bbox="448 1355 1364 1422"> <caption>D435: Address list (almount)</caption> <thead> <tr> <th>Name</th> <th>IO address</th> <th>Read o</th> <th>Data type</th> <th>Array s</th> <th>Process image</th> <th>Strategy</th> <th>Display</th> <th>Substitute v</th> <th>Control var</th> <th>Assignment</th> <th>Assignm</th> <th>Com</th> </tr> </thead> <tbody> <tr> <td>almount</td> <td>17</td> <td>AI</td> <td>WORD</td> <td>1</td> <td>Substitut</td> <td>HEX</td> <td>16R00_00</td> <td></td> <td></td> <td>SINAMICS_Integrated_A_INF_02_STW1</td> <td>4. Set up</td> <td></td> </tr> </tbody> </table> | Name | IO address | Read o | Data type | Array s | Process image | Strategy | Display | Substitute v | Control var | Assignment | Assignm | Com | almin | 16 | AI | WORD | 1 | Substitut | HEX | 16R00_00 | | | SINAMICS_Integrated_A_INF_02_ZSWI | 4. Set up | | Name | IO address | Read o | Data type | Array s | Process image | Strategy | Display | Substitute v | Control var | Assignment | Assignm | Com | almount | 17 | AI | WORD | 1 | Substitut | HEX | 16R00_00 | | | SINAMICS_Integrated_A_INF_02_STW1 | 4. Set up | |
| Name | IO address | Read o | Data type | Array s | Process image | Strategy | Display | Substitute v | Control var | Assignment | Assignm | Com | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| almin | 16 | AI | WORD | 1 | Substitut | HEX | 16R00_00 | | | SINAMICS_Integrated_A_INF_02_ZSWI | 4. Set up | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Name | IO address | Read o | Data type | Array s | Process image | Strategy | Display | Substitute v | Control var | Assignment | Assignm | Com | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| almount | 17 | AI | WORD | 1 | Substitut | HEX | 16R00_00 | | | SINAMICS_Integrated_A_INF_02_STW1 | 4. Set up | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

| 序号 | 图示与说明 |
|----|--|
| 4. | <p>在指令库中调用对应的指令 <code>_LineModule_control ()</code> :</p>  <p>The screenshot shows a software interface for a command library. At the top, there is a dropdown menu with the text "_LINE". Below it, there are two fields: "Cmds valid for:" with a dropdown showing "D435" and "Sorting:" with a dropdown showing "abc" and a list icon. The main area is a tree view of command categories. Under "Drives", there are sub-categories: "SIMATIC", "SIMODRIVE", "SIMOVERT", and "SINAMICS". Under "SINAMICS", several functions are listed, including "_LineModule_control[FB]", which is highlighted with a red rectangular box. Other functions include "_readDriveFaults", "_readDriveMultiParameter", "_readDriveMultiParameterDescription", "_readDriveParameter", "_readDriveParameterDescription", "_resetDriveObjectFault", "_setDriveObjectSTW", "_writeDriveMultiParameter", and "_writeDriveParameter". At the bottom of the window, there are tabs for "Project" and "Command library".</p> |

| 序号 | 图示与说明 |
|----|--|
| 5. | <p>在 BACKGROUND 执行任务中编写调用下图程序：</p> <pre> 1 INTERFACE 2 USEPACKAGE cam ; 3 PROGRAM a ; 4 END_INTERFACE 5 IMPLEMENTATION 6 VAR_GLOBAL 7 enableALM : BOOL; 8 resetALM : BOOL; 9 doneALM : BOOL; 10 errorALM : BOOL; 11 errorALMID : DWORD; 12 END_VAR 13 PROGRAM a 14 VAR 15 myRetDINT : DINT; 16 myALM : _LineModule_control ; 17 END_VAR 18 myALM(19 enable := enableALM 20 ,reset := resetALM 21 ,periIn := almin 22 ,typeLM := ACTIVE_LINE_MODULE 23 ,moduleAddress :=288 24 ,done => doneALM 25 ,error => errorALM 26 ,errorID => errorALMID 27 // ,stateRdPar => 28 // ,activated => 29 ,periOut => almout 30 // ,selectedLM => 31); 32 END_PROGRAM 33 END_IMPLEMENTATION 34 35 </pre> <p>注意：程序的输入 typeLM 参数需要根据实际使用的电源模块类型进行填写，如果使用的是 BLM, 则此参数填写 BASIC_LINE_MODULE, 如果使用的是 SLM, 则此参数填写 SMART_LINE_MODULE。</p> |

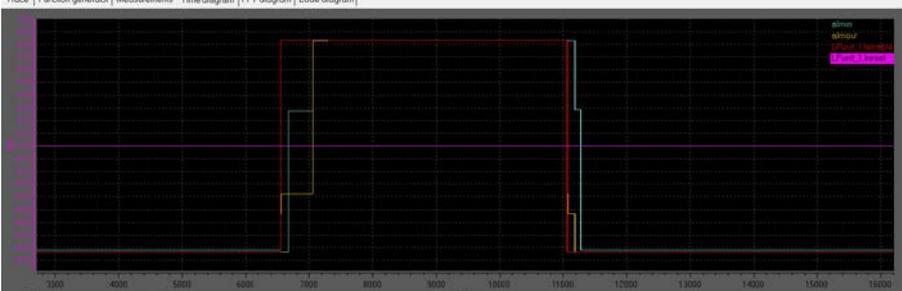
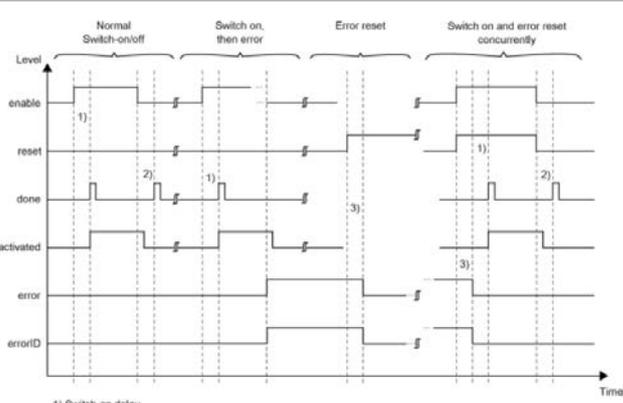
| 序号 | 图示与说明 | |
|----|---|--|
| 6. | enable | 从 false 到 true 的上升沿启动电源模块，false 停止电源模块 |
| | reset | 上升沿应答错误 |
| | TypeLM | 使用的整流模块的类型 ACTIVE_LINE_MODULE = 1 SMART_LINE_MODULE = 2 BASIC_LINE_MODULE = 3 AUTO_DETECT=4 自动型号检测 |
| | periIn | 电源模块状态字起始地址 |
| | moduleAddress | 电源模块报文状态字起始地址（如果使用自动检测需输入此地址） |
| | Done | 请求的任务执行完成 |
| | error | 模块故障 |
| | errorID | 故障代码 |
| | stateRdPar | 自动检测时出错信息 |
| | activated | true=模块启动，false=模块停止 |
| | periout | 电源模块报文控制字起始地址 |
| | selectedLM | 自动检测的结果，ACTIVE_LINE_MODULE = 1 SMART_LINE_MODULE = 2 BASIC_LINE_MODULE = 3 |
| 7. | <p>通过控制 enableALM 变量即可实现控制电源模块的功能，当 enableALM 上升沿启动电源模块，下降沿关闭电源模块。控制如下图：</p>  <p>时序图：</p>  <p>1) Switch-on delay 2) Switch-off delay 3) Time in which the error will be reset</p> | |

表 2-1 操作步骤说明