

常问问题 • 1/ 2015

如何读取 410H 控制器状态灯

How to read the LEDs of 410H controller

http://support.automation.siemens.com/CN/view/zh/109095268

Copyright © Siemens AG Copyright year All rights reserved 目 录

1	读取 410-5H 故障灯的原理介绍	3
2	例子程序的使用介绍	5
	2.1 导入 SCL 数据源,生成功能块	5
	2.2 组态相关的程序	7
	2.3 修改图标和面板	9
	2.4 单 410-5H 应用的设置	9

可靠、耐用的全能系统 AS 410 是 SIMATIC S7-400 系列一款独特的新产品, 特别针对 SIMATIC PCS 7 过程控制系统设计。该系统具有优秀的通用性,适用 于所有领域。

CPU 410-5H 过程控制器涵盖了传统 AS 412 至 AS 417 自动化系统的全部应用领域。其自动化性能可根据 SIMATIC PCS 7 过程对象 (PO) 数量灵活调整。

在系统维护时,需要知道 CPU 是否有故障,包括冗余的状态是否完好、IO卡件是否有故障,如果维护工程师定期打开控制柜巡检,这样比较直观,但是比较费时间且效率低,如果在 OS 画面上显示出 CPU 的指示灯,就可以实时了解CPU 的工作状态。

1 读取 410-5H故障灯的原理介绍

系统控制器运行过程中,控制器内部的各种不同信息都被保存在 CPU 的内部存储器中,并根据运行情况由控制器内部的操作系统实时进行更新。在 410H 中,这些内部信息也包含了 410H 的状态灯信息。

在系统提供的系统功能 SFC 中,功能 SFC51(RDSYSST)专门用于读取系统的状态信息。该功能块提供的 SSL-ID 功能码输入管脚用于设置需要读取的信息 类型,其中功能码 16#0074 可以用于读取控制器的状态灯(包括单 CPU 和冗余 CPU)。

关于 SFC51 的详细帮助请参考 Step7 在向帮助或有个系统提供的系统功能介绍。

参数	声明	数据类型	描述
REQ	INPUT	BOOL	REQ = 1: 启动处理
SSL-ID	INPUT	WORD	需要读取的系统状态功能码
INDEX	INPUT	WORD	部分功能码中对象的类型或编 号
RET_VAL	OUTPUT	INT	如果执行 SFC 时出错,则 RET_VAL 将包含出错代码
BUSY	OUTPUT	BOOL	TRUE: 尚未完成读取
SSL_HEADER	OUTPUT	STRUCT	数据记录信息: LENTHDR:单条数据记录长 度 N_DR:DR中记录的数据记录 的条数
DR	OUTPUT	ANY	读取的数据记录存储区域: •如果仅读取了SSL列表的单 条信息,则不能评估DR的 值,而只能评估 SSL_HEADER的值。 •否则,LENTHDR和N_DR 的乘积为已在DR中存储的字 节数

图 1-1 SFC51 各引脚的说明

Copyright © Siemens AG Copyright year All rights reserved 也就是说,通过 SFC51(功能码 16#0074)读取的数据记录存储在 DR 中,每条数据记录都拥有上述的结构,每条数据记录通过 CPU_LED_ID 字节 1(LED 标识符)来标识该记录对应的具体 LED,通过评估记录的后两字节,即可判断该 LED 的当前状态。下面是在试验中得到的数据记录:

R_ LAD/STL/FBD - [@DB51 "DB_LED_Status" 20150113_TEST_RuiLin_81\410H\CPU 410-5H\\DB51 ONLINE]						
🖬 File E	dit Insert PLC Debug View Options	Window H	lelp			
🗅 🚅 🏪 🛃 🎒 👗 🖻 💼 🗠 🛥 🕅 🎽 🔁 🎥 🔐 🛛 🔛 🛄 🔛						
Address	Name	Туре	Initial value	Actual value	Comment	
0.0	SZL_Header.LENGTHDR	WORD	W#16#0	W#16#0004		
2.0	SZL_Header.N_DR	WORD	W#16#0	W#16#001C		
4.0	CPU_LEDS[1].Data_record.index	WORD	W#16#0	W#16#F802		
6.0	CPU_LEDS[1].Data_record.led_on	BYTE	B#16#0	B#16#00	タ粉切り	
7.0	CPU_LEDS[1].Data_record.led_blink	BYTE	B#16#0	B#16#00	一宗奴据证	CX
8.0	CPU_LEDS[2].Data_record.index	WORD	W#16#0	W#16#F803		
10.0	CPU_LEDS[2].Data_record.led_on	BYTE	B#16#0	B#16#00		
11.0	CPU_LEDS[2].Data_record.led_blink	BYTE	B#16#0	B#16#00		
12.0	CPU_LEDS[3].Data_record.index	WORD	W#16#0	W#16#F804		
14.0	CPU_LEDS[3].Data_record.led_on	BYTE	B#16#0	B#16#01		
15.0	CPU_LEDS[3].Data_record.led_blink	BYTE	B#16#0	B#16#00		
16.0	CPU_LEDS[4].Data_record.index	WORD	W#16#0	W#16#F805		
18.0	CPU_LEDS[4].Data_record.led_on	BYTE	B#16#0	B#16#00		
19.0	CPU_LEDS[4].Data_record.led_blink	BYTE	B#16#0	B#16#00		
4						

中间是第4~26条数据记录

I	107.0	CPU_LEDS[26].Data_record.led_blink	BYTE	B#16#0	B#16#00	
I	108.0	CPU_LEDS[27].Data_record.index	WORD	W#16#0	W#16#F117	
I	110.0	CPU_LEDS[27].Data_record.led_on	BYTE	B#16#0	B#16#00	
I	111.0	CPU_LEDS[27].Data_record.led_blink	BYTE	B#16#0	B#16#00	
I	112.0	CPU_LEDS[28].Data_record.index	WORD	W#16#0	W#16#F118	
I	114.0	CPU_LEDS[28].Data_record.led_on	BYTE	B#16#0	B#16#00	
I	115.0	CPU_LEDS[28].Data_record.led_blink	BYTE	B#16#0	B#16#00	
	116.0	CPU_LEDS[29].Data_rec(💻 📂 🛠	約据につき	1 #16#0	W#16#0000	
	118.0	CPU_LEDS[29].Data_rec(印/口 示号	치기여민구정	\$ \$16#0	B#16#00	
_						

图 1-2 SFC51 各引脚的说明

在上图里可以看到,第一个红色方框里的数据表示一个灯的状态,其中"INLEX"代表指示灯的类型,INDEX="W#16#F802"中的"F8"表示 RACK0 CPU的指示灯,"02"表示是"INTF内部故障"指示灯,"LED_ON=B#16#0"表示该灯没有亮,"LED_BLINK=B#16#0"表示该指示灯没有闪烁。与410-5H 相关 LED 标识符如下所示:

- W#16#0002: INTF(内部出错)
- W#16#0003: EXTF(外部出错)
- W#16#0004: RUN
- W#16#0005: STOP
- W#16#000B: BUS1F(总线接口1故障)
- W#16#000D: REDF(冗余出错)
- W#16#000E: MSTR(主站)
- W#16#000F: RACK0(机架号 0)
- W#16#0010: RACK1(机架号1)
- W#16#0011: RACK2(机架号 2)
- W#16#0012: IFM1F(接口出错接口模块 1)

W#16#0013: IFM2F(接口出错接口模块 2)

W#16#0015: MAINT(维护请求)

W#16#0017: BUS5F(总线接口5故障)

W#16#0018: BUS8F(总线接口 8 故障)

使用 SSL-ID=16#0074 读取冗余 CPU 的状态灯时, SSL_HEADER 数据结构中 N_DR= 16#001C, 即每次读取并存储在 DR 中的数据记录条数最大为 28 条。因此,设置接受 DR 的数据区间长度应该至少大于等于:

LENTHDR×N_DR=4×28=112 bytes

根据上述的 DR 数据记录结构来分析每条数据记录,并将其显示在 OS 上即可。

2 例子程序的使用介绍

由于 CFC 编程语言的特点,要用 SFC51 读取并分析数据记录比较困难,推荐 使用结构化编程语言 SCL 来实现,为了方便用户,本文将相关的 SCL 源程序例 子和 OS 的面板附在文档后面,对于 SCL 源程序的编程不做介绍。附件里内容 介绍如下:

- ◆ @SFC51_LED_410H: 采用 SCL 编写的读取 LED 的源代码,支持冗余 CPU 和单 CPU 两种类型的控制器,已经在 410-5H V8.1 CPU 测试通过;
- ◆ @*.PDL 文件:专门为 PCS7 开发的上位显示图标和面板,需结合上述源代码一起使用,需要将它们拷贝到项目下所有 OS 站的画面目录下:



图 2-1 将画面文件拷贝到 OS 项目里

例子程序可以支持 410-5H 控制器做为冗余控制器或者单 CPU 使用,在 OS 编译时自动生成图标和面板,用户下面对例子程序的用法介绍如下:

2.1 导入SCL数据源,生成功能块

在项目的设备视图下,选中 CPU 下的"Sources"文件夹,在右侧的空白处点 击鼠标右键,在快捷菜单里依次选择"Insert New Project"→"External Source",在打开的窗口里找到解压缩的 SCL 源文件,选择"Open"后,即可 成功导入 SCL 源程序。

SIMATIC Manager - [20150113_LED_V80_OK (Component view) N:\V80_SP2\20150113]					
🎒 File Edit Insert PLC Vie	ew Options Window Help				
🗅 🗃 🔡 🛲 🕹 🖻 💼		🚺 🔁 🛛 < No Filter >	💽 🏹 🞇 🎰 🔣 🖷		
□ 20150113_LED_V80_OK □ 410H □ Image: CPU 410-5H □ Image: Sources Image: CPU 410-5H Image: Sources Image: Sources	Open Object Cut Copy Paste Delete Insert New Object PLC Access Protection Compile Export Source	Ctrl+Alt+O Ctrl+X Ctrl+C Ctrl+V Del , , , Ctrl+B	STL source SCL source SCL compile control file External Source		
🋃 Insert external	source				
Look in: 🚺 V81	•	. ← 🗈 💣 📰 -			
Name	410H.SCL	Date modified 1/24/2015 5:42 PM	T) SC		
Files of type: Sou	urces (*.awl;*.gr7;*.scl;*.inp;*.zg;*.sd	g;*.sd ▼ Cancel			

图 2-2 插入 SCL 程序源文件

打开 SCL 源程序后,找到功能块的符号名,在符号表里插入符号表和对应的 FB 块号,本文以 FB501 为例,块号推荐使用 500 以后的数字,不要与项目里已经 存在的块号冲突。如下所示:

😪 Symbol Editor - AS1 (Symbols)						Set SCL - [@SFC51 LED 410H 20150113 LED V80	
	Symbol 1	Fable E	dit Insert View O	ptions Window —	者要	፤——致	File Edit Insert PLC Debug View Opt
	🚰 🗐 🎒 👗 🖻 💼 🗠 🖂 🛛 All Symbols						
l	-			· ·	_	-	
l	🔄 AS1	(Symbo	ls) 20150113_LED_V80	0_OK\410H\CPU.410-5H			
I		Status	Symbol /	Address	Dat	ta type	
I	1		ALARM_8P	SEB 35	SFE	3 35	A % %
I	2		ChkREAL	FC 260	FC	260	//Comunication (C) Sigmong AC 1000
I	3		CONEC	FB 88	FB	88	//copyright (C) Stemens AG 1999.
I	4		CPU_RT	FB 128	FB	128	// Plack banda
I	5		LED_410H	FB 501	FB	501	// BIOCK HEADER
I	6		MOD_1		ED	 1	//
I	7		MotL	FB 185(状号随息 ,	推	存 50	FUNCTION DI CONTENDA 1 OU
I	8		OB_BEGIN		2	10	TITLE _ LIED 410H
I	9		OB_END	FC 280 FD 300 K/F	1	30	IIITE - TED AION ((T
I	10		PA_CPU	FB 16	FB	16	1 // 1 //57 tookligtur 10000 001001. //
I	11		Pcs7Diln	FB 1871	FB	1871	//5/_Caskiist:= '0680,06100'; //
I	12		PIDConL	FB 1874	FB	1874	S7_m_c:= ·crue; //r
	13		PO_UPDAT	FC 279	FC	279	
1	14		PS	FB 89	FB	89	
J	15		RACK	FB 107	FB	107	

图 2-3 在符号表里定义 LED_410H

Copyright © Siemens AG Copyright year All rights reserved 接下来,打开导入的 SCL 源文件,鼠标左键点击编译按钮,系统会生成 FB501 功能块。

SCL - [@SFC51_LED_410H 20150113_TEST_RuiLin_81\410H\CPU 410-5H]
🔀 File Edit Insert PLC Debug View Options Window Help
□≥₽ @ ▷♀४฿€ @8₩₩₩ ₩ !<>! ! ₻■Ⅲ
//Copyright (C) Siemens AG 1999. All Rights Reserved. Confiden
//*************************************
// Block header
//*************************************
FUNCTION_BLOCK LED_410H
TITLE = 'LED_410H'
{ // List of system attributes
<pre>//S7_tasklist:= 'OB80,OB100'; // Block is called if there is</pre>
S7_m_c:= 'true'; // Block can be controlled and mo
S7_alarm_ui:= '1' // Setting for PCS 7 message dial
}
AUTHOR: LOGGY_GONG
NAME: LED_410H
VERSION: '0.01'
FAMILY: CS2PA
KNOW_HOW_PROTECT

图 2-4 编译 SCL 源生成 FB 块

2.2 组态相关的程序

在工厂视图里找到合适的层级,插入一个新的 CFC,在 Blocks 里找到编译生成的功能块(本例是 FB501),将它拖入到 CFC 里调用,将块的名字改成一个有意义的名字,这样在画面上生成图标时便于理解。

注意调用这个块的 CFC 要在某一层级下,并且层级下是有画面,否则无法自动 生成图标。



Copyright © Siemens AG Copyright year All rights reserved

图 2-5 CFC 里调用功能块

注意:由于调用 SFC51 将占用 CPU 系统资源,建议采用较慢循环周期的 OB 来 调用该 SFC51,例如 OB32,1 秒钟调用一次或更慢。

接下来,对报警进行相关设定,报警文本可以根据需要自己设定,相关的步骤和 解释如下图:



图 2-6 组态块的报警

接下来,对 CFC 进行编译、下载,对 OS 项目进行编译,这时在相关的画面上和下载,运行 OS 项目,最终上位显示效果如下图所示:



图 2-7 画面运行后的效果

2.3 修改图标和面板

Copyright © Siemens AG Copyright year All rights reserved

在项目执行过程中,用户可能对图标和面板中对象的颜色有自己的要求,这时需要做一些改动,例如:用户觉得机架的指示灯是黄颜色不合适,可以打开" PG_LED_410H_STANDARD"画面进行修改,如果 PCS7 是中文版,也可以将 注释文本改为中文,如下图所示:

Graphics Designer @PG_LED_410H_STANDARD.pdl								
File Edit View Arran	File Edit View Arrange Tools Window Help							
E 🗈 🖻 🚰 🛃 🕨 🐰 🗈 🖭 🔊 🗠 🚑 1111 👯 🔲 🔂 🚼 🗞								
i RACK11 · 🖫 🖫 🍼 🏏 🗛 🚽 i 臣 릐 市 业 후 아 Hel								
@PG_LED_410H_STAND	ARD.pdl ×							
Rack 0 INTF REDF EXTF MSTR BUS1 RACK0 BUS5 RACK1 BUS8 FM1F RUN FM1F STOP MAINT	Rack 1 INTF EXTF MSTR BUS1 RACK0 BUS5 RACK1 BUS8 IFM1F RUN IFM2F STOP MAINT							
Object Properties								
Properties Events	显示文字							
Customized object Geometry Miscellaneous UserDefined2	Attribute 可以自定义 Static Dynamic Update Indir ont Color ont Size 12 ext RACKL ONCOLOR FLASHONCOLOR 3rt_NO 9.000000 指示灯的颜色可以改变							

图 2-8 自定义指示灯的文本和颜色

如果还觉得图标里对象的颜色也需要修改,就需要打开@Pcs7typical 可以参考下面的 FAQ 进行自定义:

http://support.automation.siemens.com/CN/view/zh/16514590

如何创建自己的块图标自动的集成到过程画面中?

http://support.automation.siemens.com/CN/view/zh/26697820

如何使用模板画面 "@PCS7Typicals*.pdl" 来创建块图标 (block icon)?

2.4 单 410-5H应用的设置

如果将附件的程序在单 410-5H 的应用时,需要将图标的类型设置为 2,如下 图所示:

Properties - Block TES	T_410H\1	R1 RUN — R0 MAST — R1 MAST — CPU FAIL — CPU FAUL —
General I/Os		
Type: Name:	LED_410H	Block group:
Comment:	LED_410H	
Inputs:	16	
Internal identifier:	FB501	OCM
Instance DB:	DB67	
Name (header):	LED_410H	Create block icon: 2
Family:	CS2PA	
Author:	LOGGY_GO	单410 - 5H应用,输入"2"

LED_410H LED_410H

LED RO

REO

SSL ID 16#0 - INDEX

No_req-

16#74-

图 2-9 设置图标的类型

另外,还要把 SIG1 的报警取消,因为功能在检测到读回的数据长度只有 14条 时会触发此报警,方法是在 Message class 里把它设置为"no message"。

P	PCS7 Message Configuration - 20150113_LED_V80_OK\410H\CPU 410-5H\AS1\Charts\PID\AS01					
	Last changed 01/24/2015 08:33:43 PM Type: FB501 Display language: English (United States)					
		Message identifier	Message class	Priority	Event	
		EV_ID				
	F	SIG1	< no message >	1	AS01 CPU FAILURE	
	F	SIG2	Alarm - high 💦	1	AS01 CPU FAULT(BUSF/ INTF/ EXTF / REDF / IFMF)	
	F	SIG3	Alarm - high	1		
	F	SIG4	Alarm - high 如果	是单41아	H CPU应用,	
	F	SIG5	Alarm - high			
	F	SIG6	Alarm - high 需要	取消SIG1	.报警	
	F	SIG7	Alarm - high	1		
		SIG8	Alarm - high	1		
	•		111		•	
					More>>	
	Save Cased Help					
		9040				
	_					

