1. 概述

在 S7-400H 系统中，当冗余控制器中任何一个出现故障后，整个冗余系统将进入单控制器运行的状态。此时如果不能及时发现并修复故障 CPU，整个系统将存在极大的安全隐患。此外，中控室和控制器通常不放置在一起，而且控制器被安装在独立封闭的控制柜内，巡检也仅仅是定时进行的。因此如何在上位机显示冗余 CPU 的状态信息，在故障发生后第一时间通知维护人员对设备进行维护一直是操作人员非常关心的问题。本文针对这一问题进行总结，归纳了几种在 OS 上监视冗余 CPU 的方法，其中部分也适合于单 CPU。

2. 通过 Lifebeat Monitoring 监视 H CPU 的状态

Lifebeat Monitoring(简称 LBM)用来监视所有能访问的服务器、客户机和 AS 系统。下面以 414-4H CPU 为例，介绍 PCS 7 软件下，使用 LBM 监视 H CPU 的组态过程。

- 第一步，打开 SIMATIC Manager，在 NetPro 中新建立两个 S7 connection，连接名分别为 OS_CPU0 和 OS_CPU1，用于 WinCC Application 和两个 CPU 的通讯，连接建完后，编译并且分别下装到 PC STATION 和 AS 站中，如下图所示：

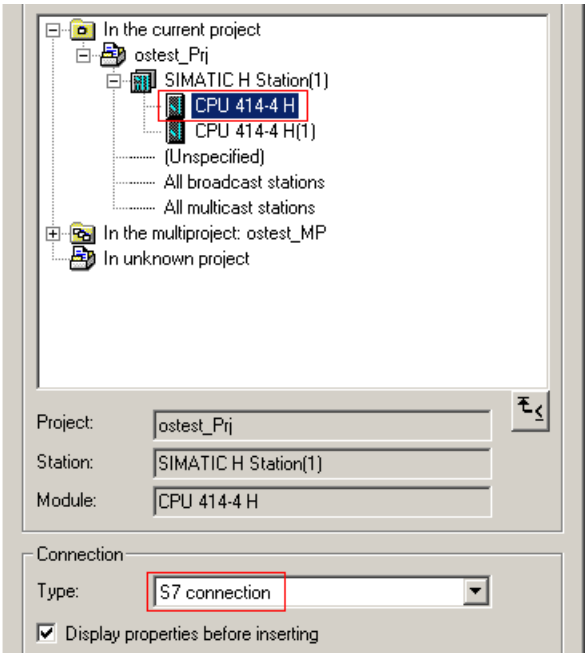


图 1： 建立 WinCC 和 rack0 CPU 的 S7 connection

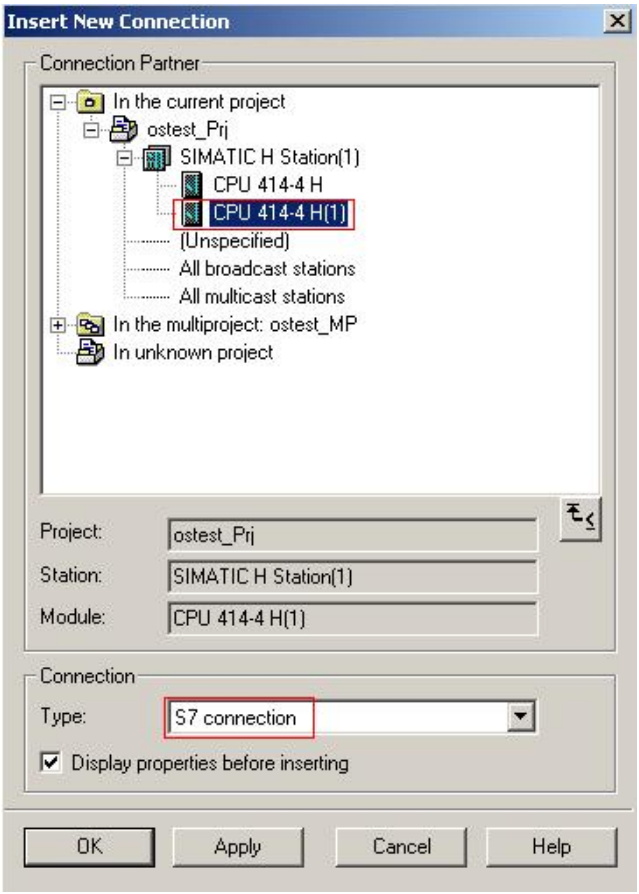


图 2： 建立 WinCC 和 rack1 CPU 的 S7 connection

Local ID	Partner ID	Partner
OS_CPU0	2	SIMATIC H Station(1) / CPU 414-4 H
OS_CPU1	3	SIMATIC H Station(1) / CPU 414-4 H(1)
S7 connection_1	1	SIMATIC H Station(1) / CPU 414-4 H / CPU ...

图 3： 新建立的两个 S7 connection

注意：此处一定要分别和两个 CPU 单独建立 S7 Connection，而不是一个 S7 Connection Fault-Tolerant；这两个链接仅仅用于状态监视，而数据通讯仍然要通过容错连接实现。

- 第二步，打开 OS 项目，在 Tag Management 中，找到 Named Connections，分别建立 OS 和俩冗余 CPU 的驱动连接 OS_CPU0 和 OS_CPU1，按下图所示设置属性；

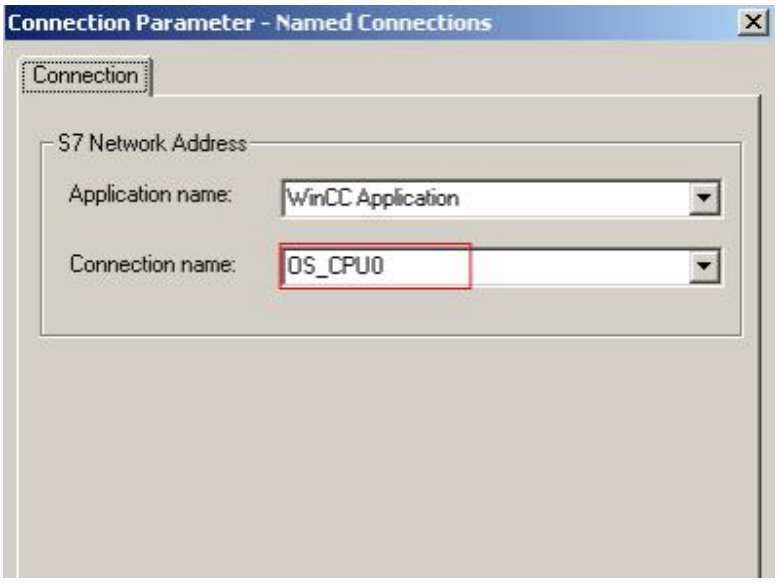


图 4： 建立 OS 和 rack0 CPU 的驱动连接

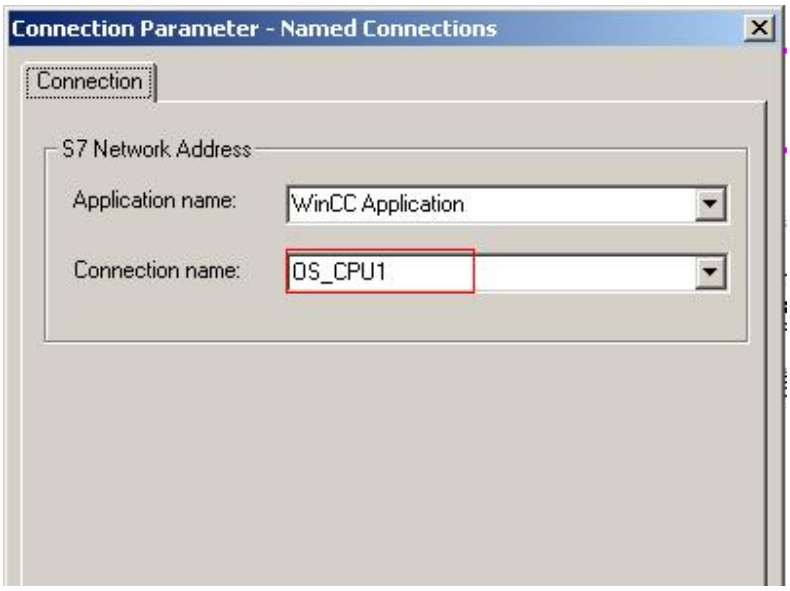


图 5：建立 OS 和 rack1 CPU 的驱动连接

- 第三步，在 WinCC Explorer 中，打开 Lifebeat Monitoring，如下图所示，在 Device List 中，插入两行 device，对应 H CPU 中的两个 CPU，Connection 中分别选择上面建立的两个连接，点击 Update 按钮更新画面。

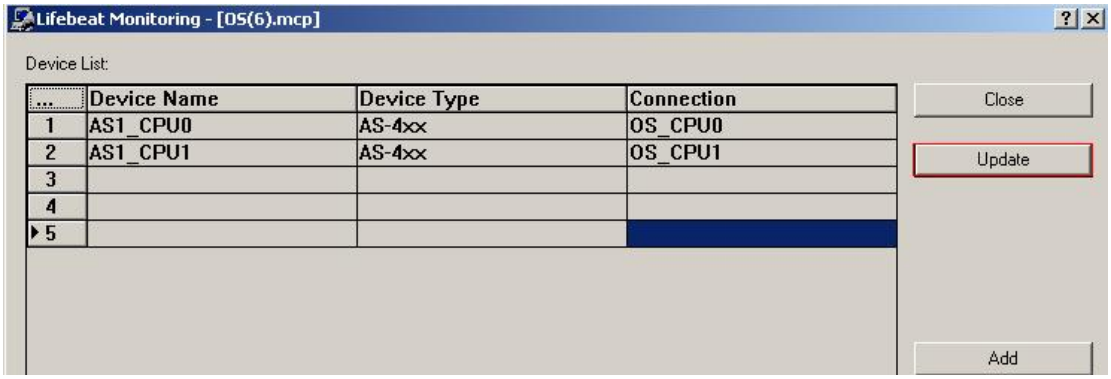


图 6：组态 Lifebeat Monitoring

注意：系统将自动生成@config.pdl 画面，如果需要对画面结构进行调整，可以用画面编辑器打开该画面调整位置。

- 第四步，编译 OS，下装到目标 PC，运行 OS。 OS 运行后，点击按钮区如下图所示的按钮就可以调出 Lifebeat Monitoring 的监视画面。

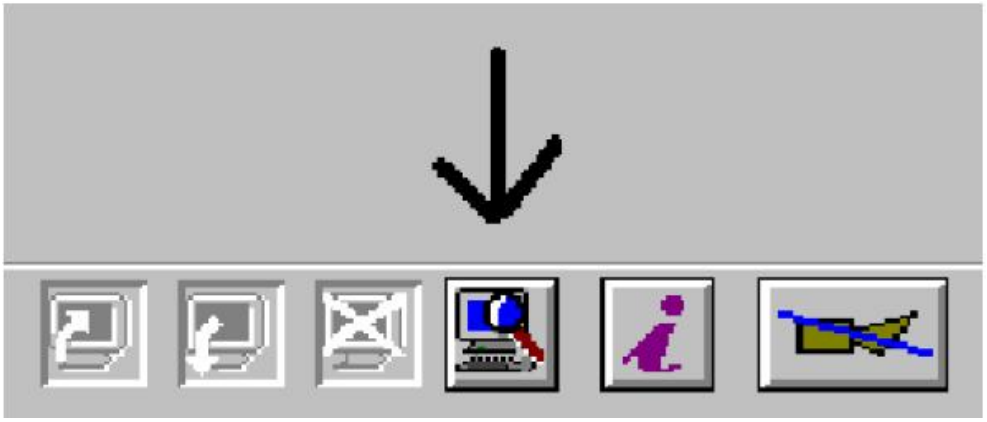


图 7： 点击该按钮调出监视画面

画面中两个图标对应 H CPU 的两个 CPU，CPU 正常运行时状态如下图所示：

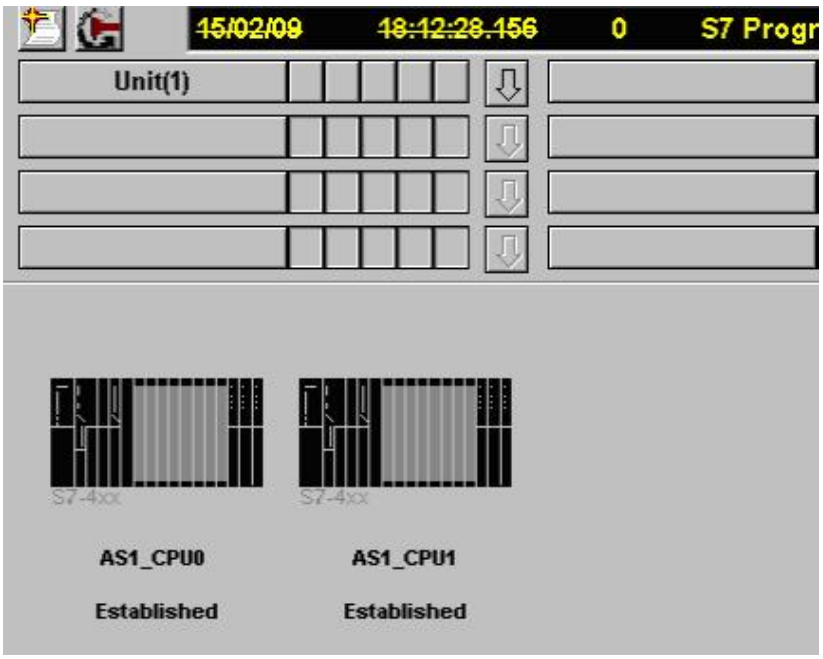


图 8：CPU 的监视画面

OS SERVER 与任何一个 CPU 的通讯中断后，都会在对应的图标上显示红色的“ X ”,如下图所示。

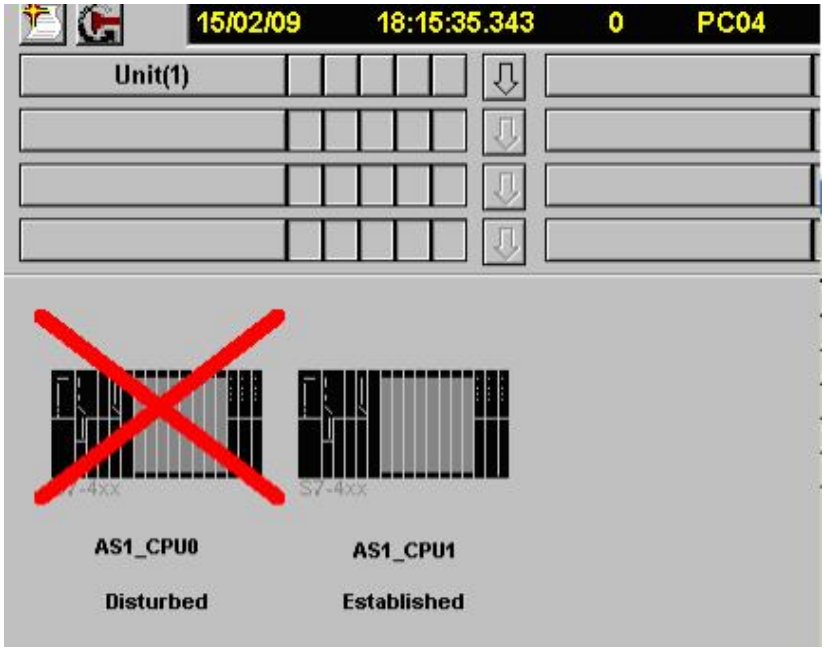


图 9：CPU 的监视画面

强烈建议用户将@config.pdl 中的图标拷贝到自己的监视画面中，使操作人员能够及时发现 CPU 和 OS 的连接状况。

总结： 通过 Lifebeat Monitoring 监视 CPU 特点是组态方便，简单易懂；但是实际上监视的不是 CPU 的运行状态，而是 OS 和 CPU 之间的连接状况，也就是说，如果上位机画面显示红色的＂X＂，只能表示 CPU 和 OS 之间的连接出现问题，而不能够判断是 CPU 出现了故障还是网络出现了故障。

3. 通过编程的方式读取 H CPU 的运行状态

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。在冗余控制器中，这些内部信息也包含了冗余控制器的状态灯信息。 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。SFC51 的参数解释如下，

REQ ：为 1 是读取 SZL_ID 指定的系统信息。

SZL_ID ：指定需要读取的系统信息。

INDEX ：根据 SZL_ID 为不同值时有不同的含义。

RET_VAL ：调用 SFC51 的状态字。

BUSY ： 为 1 时表示读进程没有完成。

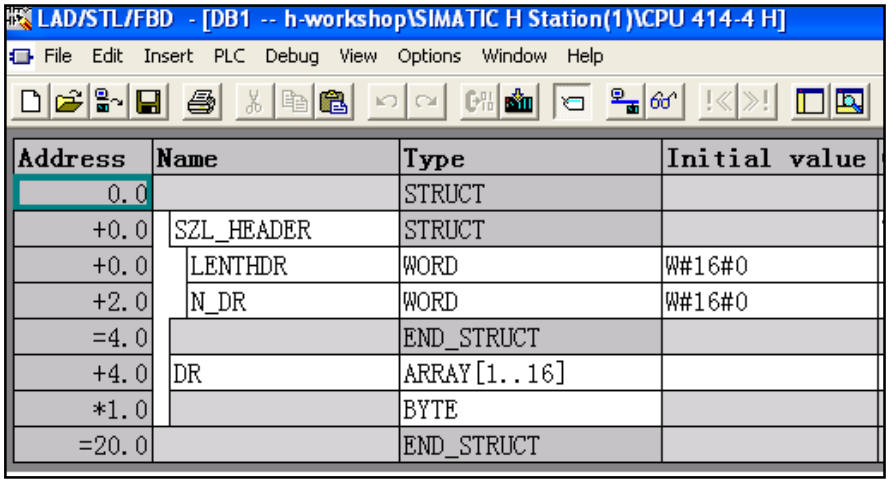
SZL_HEADER：输出系统信息存储的数据记录区号及长度，结构数据类型。

DR ：指定输出系统信息存储在 CPU 的地址区。

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，当 SSL-ID 等于 W#16#0071 时，表示需要读出 S7-400H 系统当前的状态，16#0074 则可以用于读取控制器的状态灯（包括单 CPU 和冗余 CPU）。下面分别介绍 SSL-ID= W#16#0071 和 SSL-ID= W#16#0074 时的诊断功能。

3.1 利用 SFC51(SSL-ID=W#16#xy71)读取 H 系统信息

当 SSL-ID 等于 W#16#0071 时,表示需要读出 S7-400H 系统当前的状态，将读出的诊断信息存放在 DB 块中，DB 块的数据结构如图 2－1 所示：



Address	Name	Type	Initial value
0.0		STRUCT	
+0.0	SZL_HEADER	STRUCT	
+0.0	LENTHDR	WORD	W#16#0
+2.0	N_DR	WORD	W#16#0
=4.0		END_STRUCT	
+4.0	DR	ARRAY[1..16]	
*1.0		BYTE	
=20.0		END_STRUCT	

图 10： 参数 SZL_HEADER 与 DR 的地址区

参数 SZL_HEADER 为一个结构数据，包括两个字：

- 第一个字输出系统信息长度，例如 W#16#10 表示输出 16 个字节；
- 第二个字输出存储系统信息的数据记录区，例如 W#16#1 表示数据记录区为 1；

参数 DR 为存储系统信息的地址区，数据类型为指针，长度必须大于参数 SZL_HEADER 第一个字输出的信息长度。读取的系统信息存储在 DB1.DBB4~DB1.DBB20 16 个字节中。用户可以对 16 个字节长度的系统信息进行分析和处理，系统信息内容如下：

内容　　长度　　含义

Redinf　　2 bytes　　冗余信息

W#16#0011：单机 H CPU运行

W#16#0012：H系统2备1运行

Mwstat1　　1 byte　　状态字节1

Bit 0：保留

Bit 1：保留

Bit 2：保留

Bit 3：保留

Bit 4：机架0中CPU的状态

=0：从 CPU

=1：主 CPU

Bit 5：机架1中CPU的状态

=0：从 CPU

=1：主 CPU

Bit 6：保留

Bit 7：保留

Mwstat2　1 byte　状态字节2

Bit 0：同步连接状态 01：

CPU 0 和CPU 1同步

=0：不可能

=1：可能

Bit 1： 0

Bit 2:：0

Bit 3：保留

Bit 4：=0：CPU没有在机架0

=1：CPU在机架0上

(冗余模式： bit 4 = 0)

Bit 5：=0：CPU没有在机架1

=1：CPU在机架1上

(冗余模式： bit 5 = 0)

Bit 6：保留

Bit 7：主从切换是否从新使能

=0：否

=1：是

Hsfcinfo 2 bytes SFC 90 "H_CTRL"状态字

Bit 0：=0：从新使能没有激活

=1：从新使能激活

Bit 1：=0：从站Updating使能

=1：从站Updating没有使能

Bit 2：=0： Link-up 模式没有使能

=1： Link-up 模式使能

Bit 3：保留

Bit 4：保留

Bit 5：保留

Bit 6：保留

Bit 7：保留

Bit 8：保留

Samfehl 2 bytes 保留

Bz_cpu_0 2 bytes CPU在机架0的模式

W#16#0001：停止 (update)

W#16#0002：停止 (reset memory)

W#16#0003：停止(self-initialization)

W#16#0004：停止(internal)

W#16#0005：启动(cold restart)

W#16#0006：启动(warm restart)

W#16#0007：启动(hot restart)

W#16#0008：运行(solo mode)

W#16#0009：运行(redundant mode)

W#16#000A：HOLD模式

W#16#000B：LINK-UP模式

W#16#000C：UPDATE模式

W#16#000D：故障

W#16#000E：自检测

W#16#000F：没有开机

Bz_cpu_1　2 bytes　CPU在机架1的模式

(与 bz_cpu_0相同)

Bz_cpu_2　2 bytes　保留

Cpu_valid　1 byte　信息变量 bz_cpu_0 和 bz_cpu_1有效性

B#16#01： bz_cpu_0 有效

B#16#02：bz_cpu_1 有效

B#16#03: bz_cpu_0 和 bz_cpu_1 有效

hsync_f　1 byte　连接质量的状态 (只有mwstat2 bit 0为1时有效)

Bit 0：上部插孔的同步模块光纤连接质量被限制

Bit 1：下部插孔的同步模块光纤连接质量被限制

Bit 2 到 7： 0

关于SSL_ID=W#16#0071的更详细的信息，请参考西门子网站下载中心文档“西门子冗余系统指南”的第五章S7-400H系统信息及诊断：

http://www.ad.siemens.com.cn/download/searchResult.aspx?searchText=F0153

3.2 利用 SFC51(SSL-ID W#16#xy74)读取控制器的状态灯

每个 CPU 上都有很多状态灯，根据状态灯能够判断出 CPU 的当前状态；使用 SFC51，SSL_ID =W#16#0074 能够读出 CPU 的状态灯信息。 使用 SSL-ID=16#0074 读取冗余 CPU 的状态灯时，SSL_HEADER 数据结构中 LENTHDR= 16#0004，即每条数据记录占用 4 个字节，具体结构如下图所示：

名称	长度	含义
cpu_led_ID	1个字	• 字节0 - 标准CPU：B#16#00 - H-CPU： 位0至2：机架号 位3：0 = 待机CPU，1 = 主站CPU 位4到7：1111 字节1：LED标识符
led_on	1字节	LED的状态： 0：关闭 1：打开
led_blink	1字节	LED的闪烁状态： 0：不闪烁 1：正常闪烁(2赫兹) 2：缓慢闪烁(0.5赫兹)

图 11：DR 数据记录结构

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，LED 标识符如下所示：

W#16#0001：SF(组出错)

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#0006：FRCE(强制)

W#16#0007：CRST(冷重启)

W#16#0008：BAF(总线上的电池故障/超载、电池电压短路)

W#16#0009：USR(用户自定义)

W#16#000A：USR1(用户自定义)

W#16#000B：BUS1F(总线出错接口1)

W#16#000C：BUS2F(总线出错接口2)

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)

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。

关于 SSL_ID=W#16#0074 的具体组态方法，请参考西门子网站下载中心文档：

http://www.ad.siemens.com.cn/download/searchResult.aspx?searchText=F0248

总结：通过系统功能块读取 H CPU 的运行状态能够帮助运行人员和维护人员全面的了解 CPU 和其他组件的状态，而且，读出的诊断数据可以送到任何一种上位机中，因此即使使用第三方的上位机软件，也能实现 H CPU 的状态监视。

4. 通过 PCS 7 系统功能块 OB_BEGIN 的 Faceplate 在上位机显示 H CPU 故障方法

OB_BEGIN 是 CFC 的一个系统功能块，在生成模板驱动时产生；通过 OB_BEGIN 的 Faceplate 同样能够显示 CPU 的故障状态。下面介绍这种方法的组态过程。

- 第一步：在项目中编译 CFC 时，选择激活模板驱动选项。

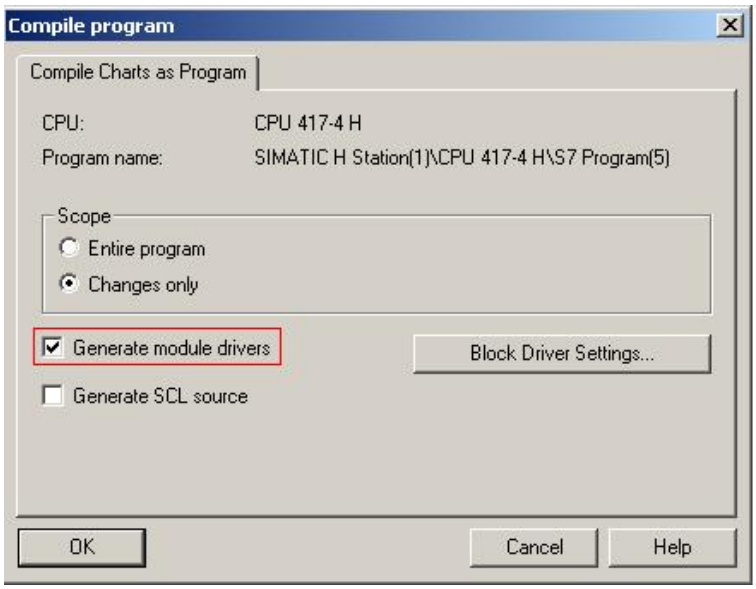


图 12：CFC 编译

注意：必须在 CFC 中至少插入一个 CH_XX 功能块，并且编译时选上 Generate module drivers 才能在编译时生成所需的驱动和诊断等信息。

- 第二步：编译 OS，然后打开 OS 项目。
- 第三步：打开模板@template.pdl，找到 Diagnostic 部分的图标 OB_BEGIN，

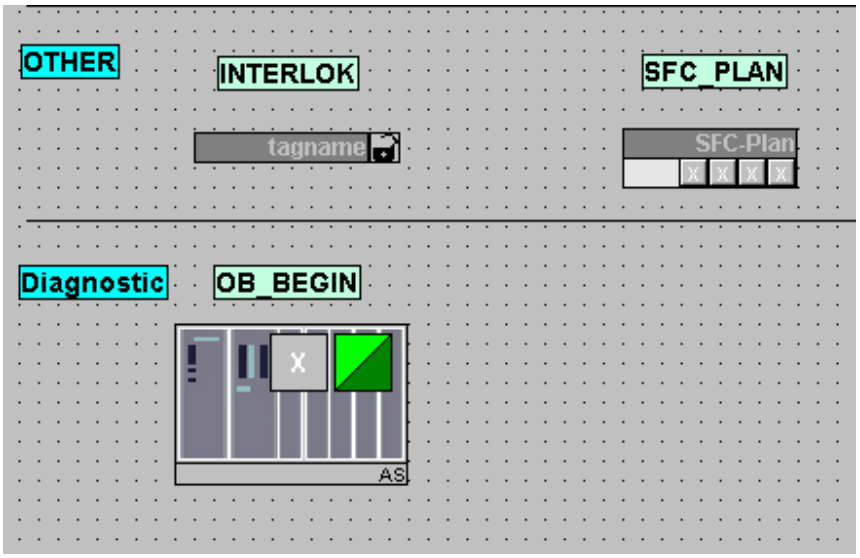


图 13：OB_BEGIN 的图标

然后把这个图标拷贝到自己的诊断画面中，使用动态向导 “Connect picture block to tag structure”，为 OB_BEGIN 的图标连接结构变量。

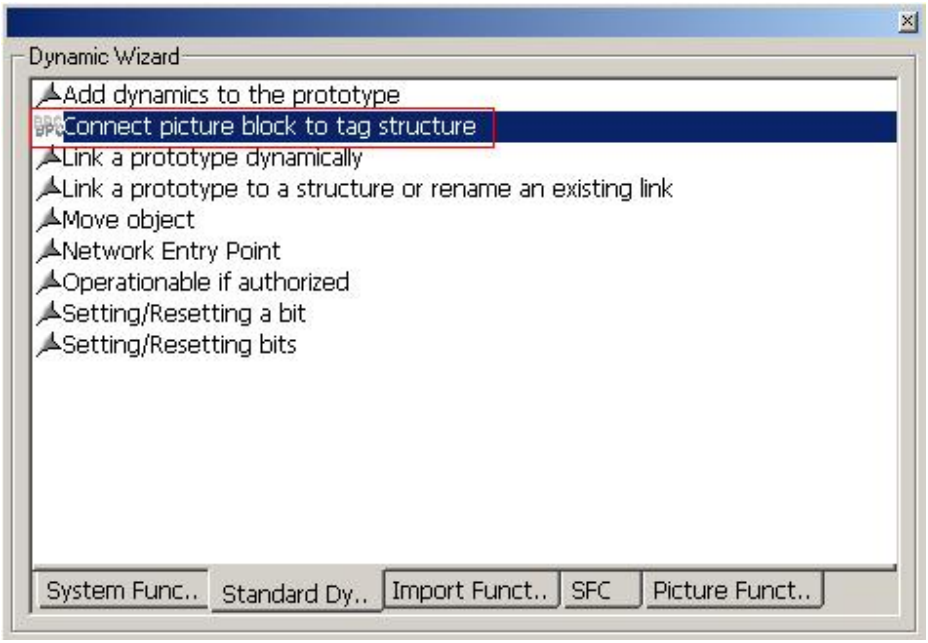


图 14：使用动态向导为 OB_BEGIN 的图标连接变量

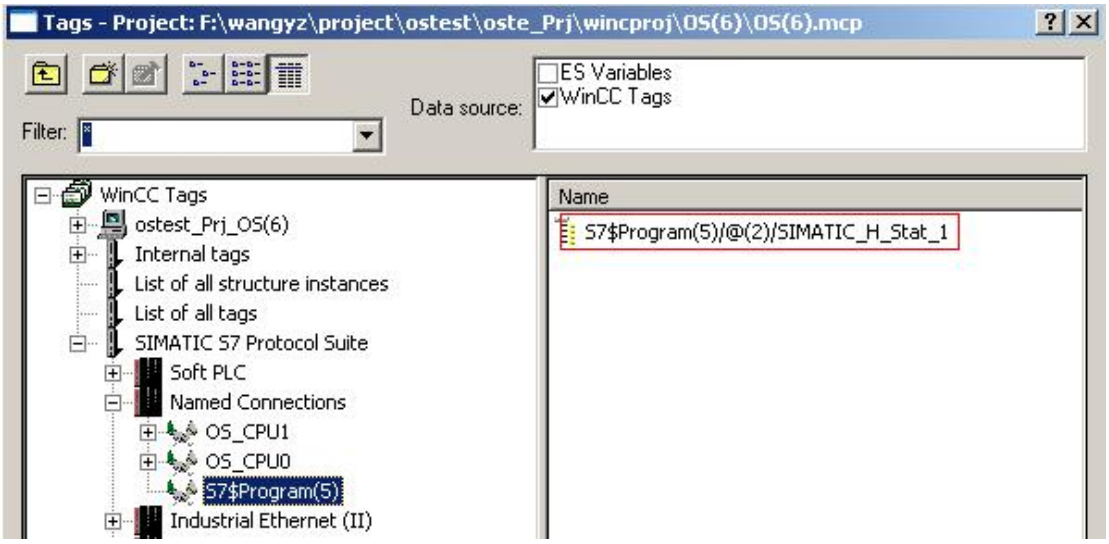


图 15：连接到 OB_BEGIN 的结构变量

- 第四步：保存画面，运行 WinCC。

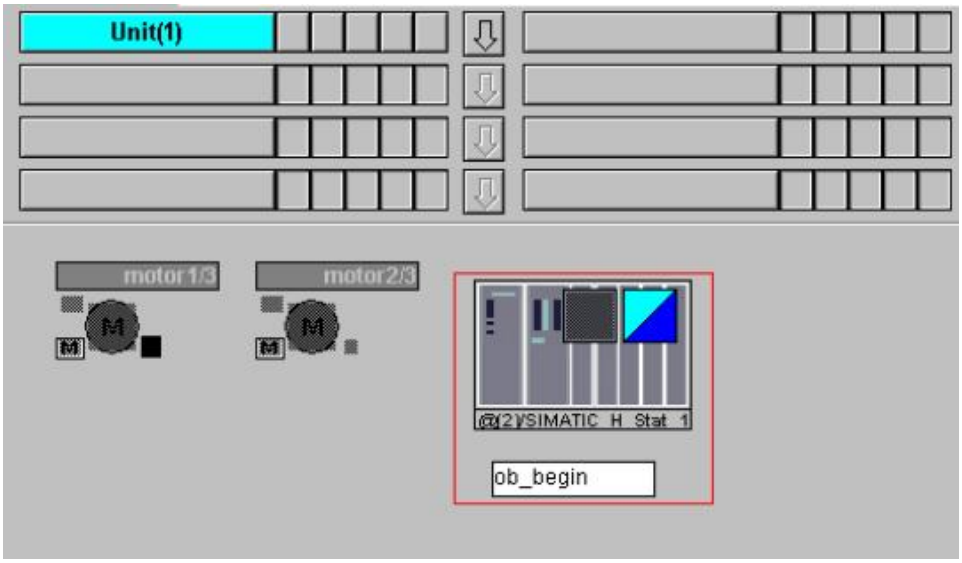


图 16：WinCC 运行后 OB_BEGIN 的图标

点击 OB_BEGIN 的图标，弹出 Faceplate。

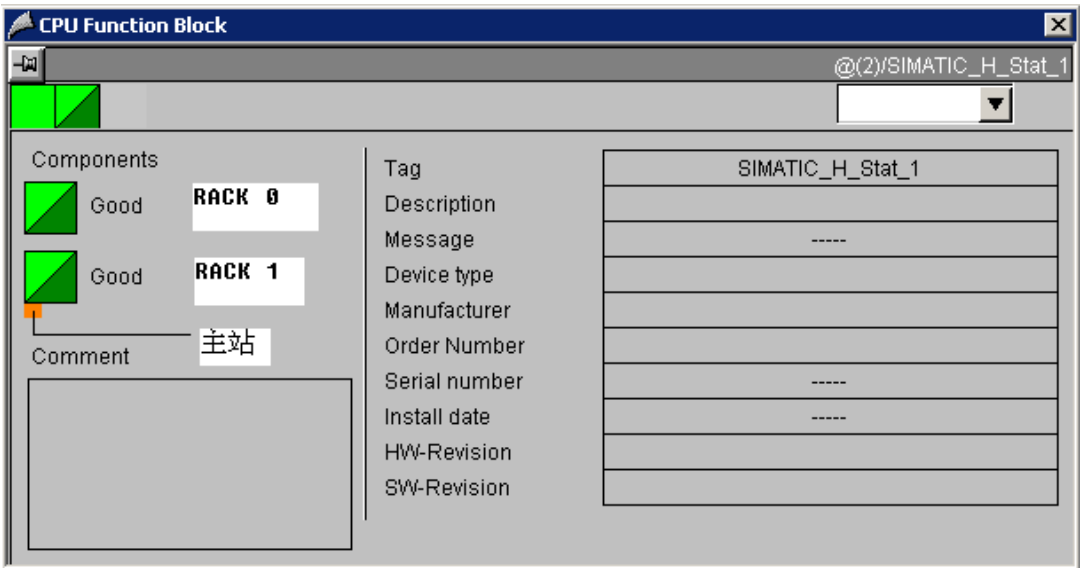


图 17：OB_BEGIN 的 Faceplate

在该画面中，上边的绿色方框代表 RACK 0 的 CPU，下边的代表 RACK 1 的 CPU，黄色方框代表主站。如果此时，RACK 1 的 CPU 停止运行或者断电，faceplate 的状态如下图所示。

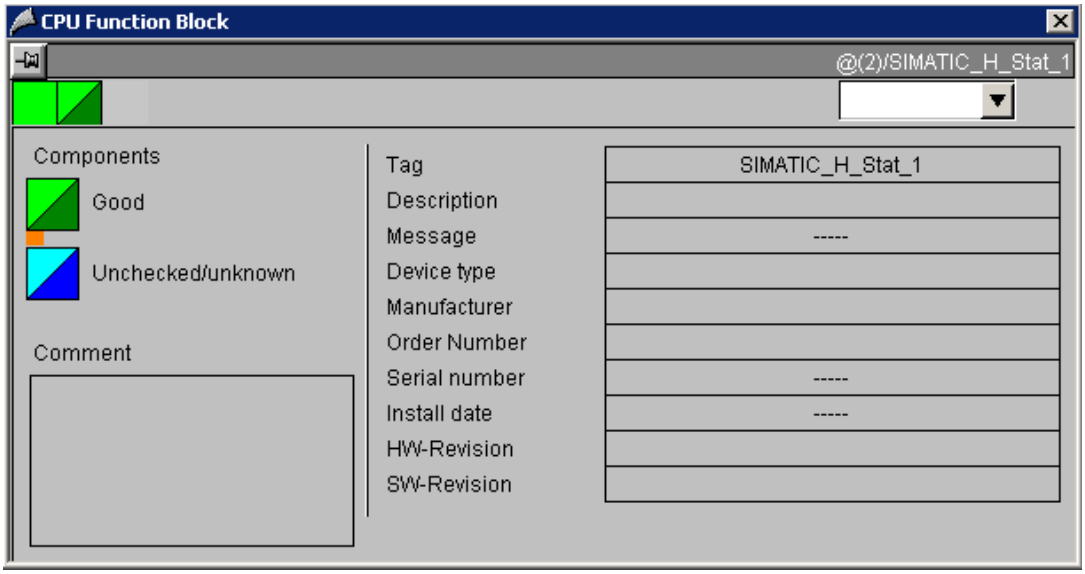


图 18：OB_BEGIN 的 Faceplate

表示 RACK 1 的 CPU 故障，RACK 0 的 CPU 为主站且在运行状态。

总结：OB_BEGIN 的 Faceplate 的方法组态简单，能够非常直观的反映出两个 CPU 的运行状态，而且能够指示出哪个 CPU 是主站，哪个是备用站。这种方法和后边的资产管理的方法原理上是一样的，虽然功能没有资产管理强大，但是不需要单独的授权，与资产管理相比，不失为一种简单实用而又经济的方法。

5. 通过 PCS 7 的资产管理 AM 进行 H CPU 的状态监视

PCS 7 的资产管理软件 Asset Management 是一个工厂维护软件，能够将控制系统的整个硬件结构分层映射到维护站中，从而实现对整个控制系统的状态监视、管理和维护。下面介绍通过资产管理在上位机显示 CPU 故障方法的组态过程：

- 第一步：项目中编译 CFC 时，选择激活模板驱动选项，如下图所示；

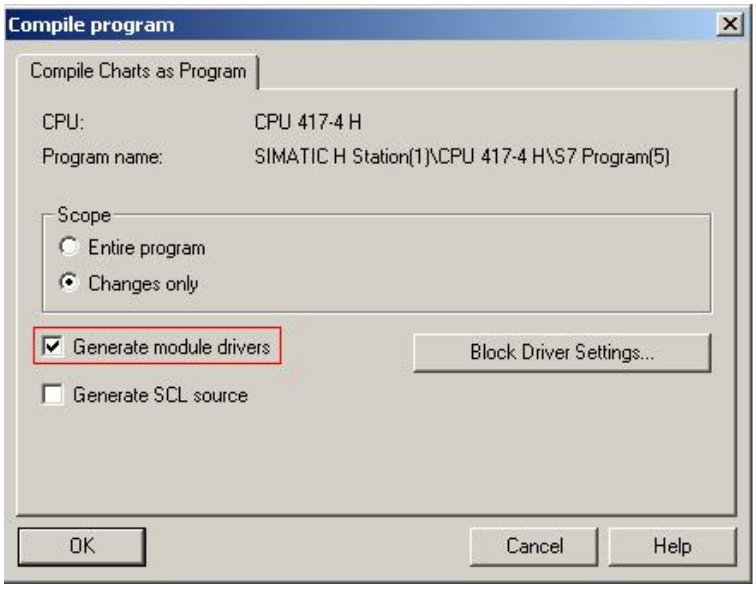


图 19：CFC 编译

注意：必须在 CFC 中至少插入一个 CH_XX 功能块，并且编译时选上 Generate module drivers 才能在编译时生成资产管理所需的驱动和诊断等信息。

- 第二步：在 plant view 中，右键单击层级文件夹，plant hierarchy → settings，打开设置对话框，如下图所示：

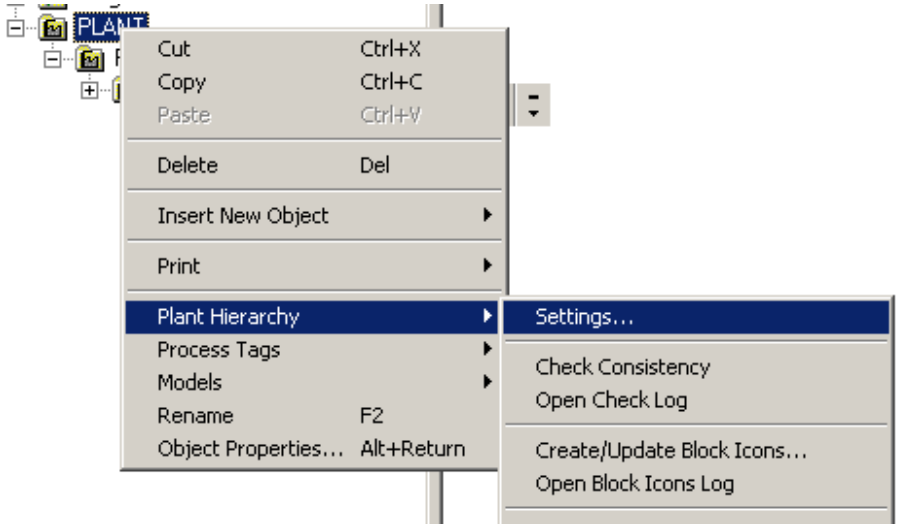


图 20：打开层级文件夹的设置对话框

选上 Derive diagnostic screens from the plant hierarchy，然后点击 OK。

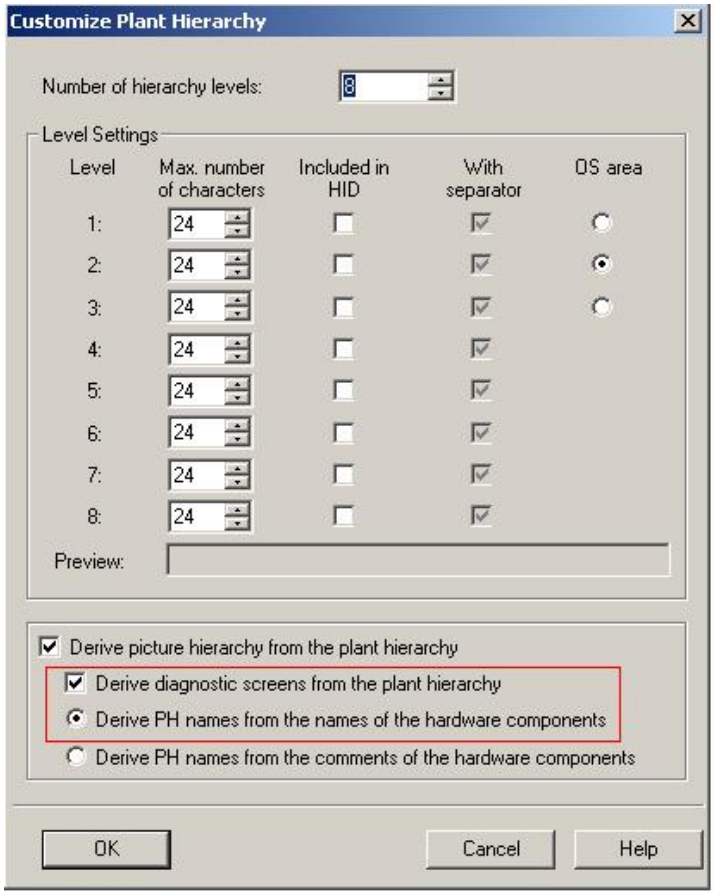


图 21：层级文件夹的设置对话框

系统会自动的产生一个顶级的层级文件夹 Diagnostics，并且根据硬件组态情况产生相应的子层级文件夹和诊断画面等。

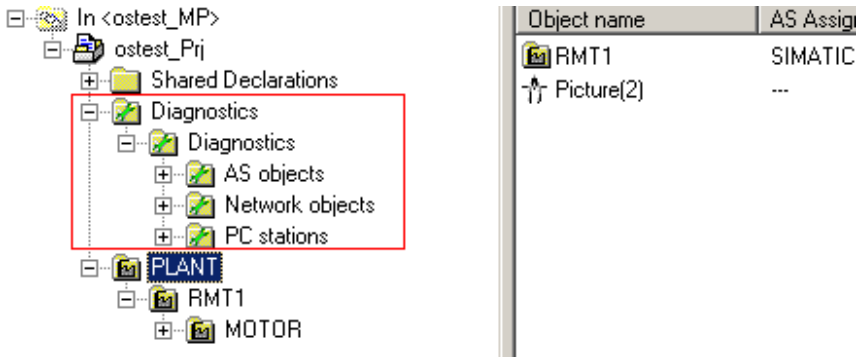


图 22：系统自动生成 Diagnostics 文件夹

- 第三步：右键单击层级文件夹，Plant hierarchy → Create/Update Diagnostic Screen 更新图标。

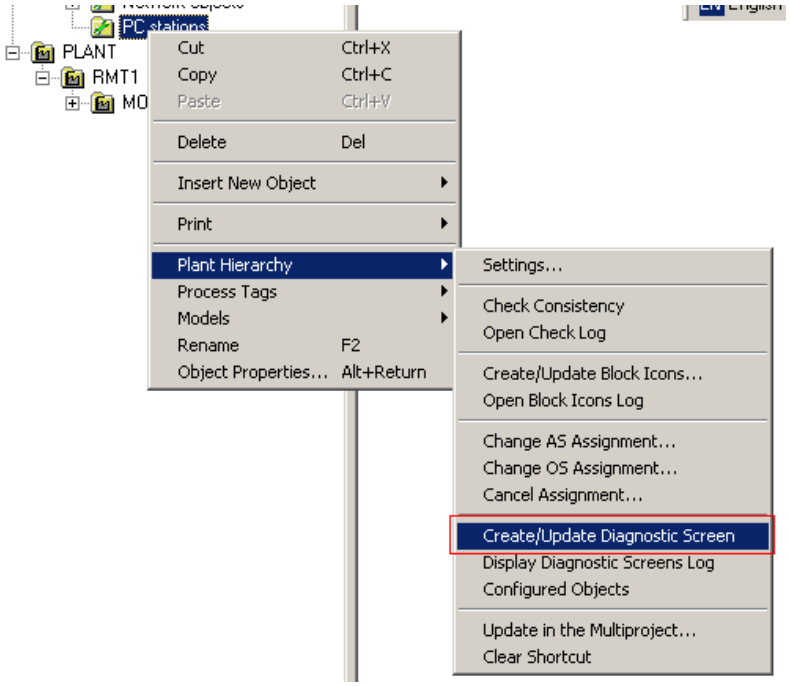


图 23：更新诊断画面

- 第四步：编译 OS 上传诊断画面，打开该 OS 项目，运行 WinCC。

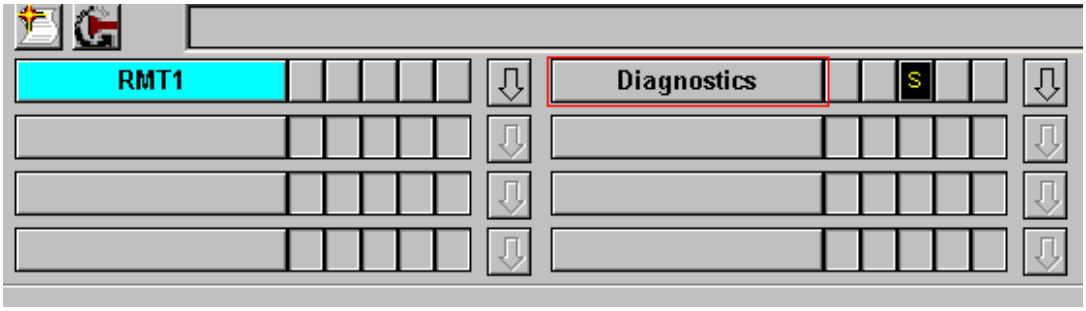


图 24：WinCC 运行后

可以看到，在导航栏有一个 Diagnostics 的按钮，点击该按钮，进入资产管理的画面。

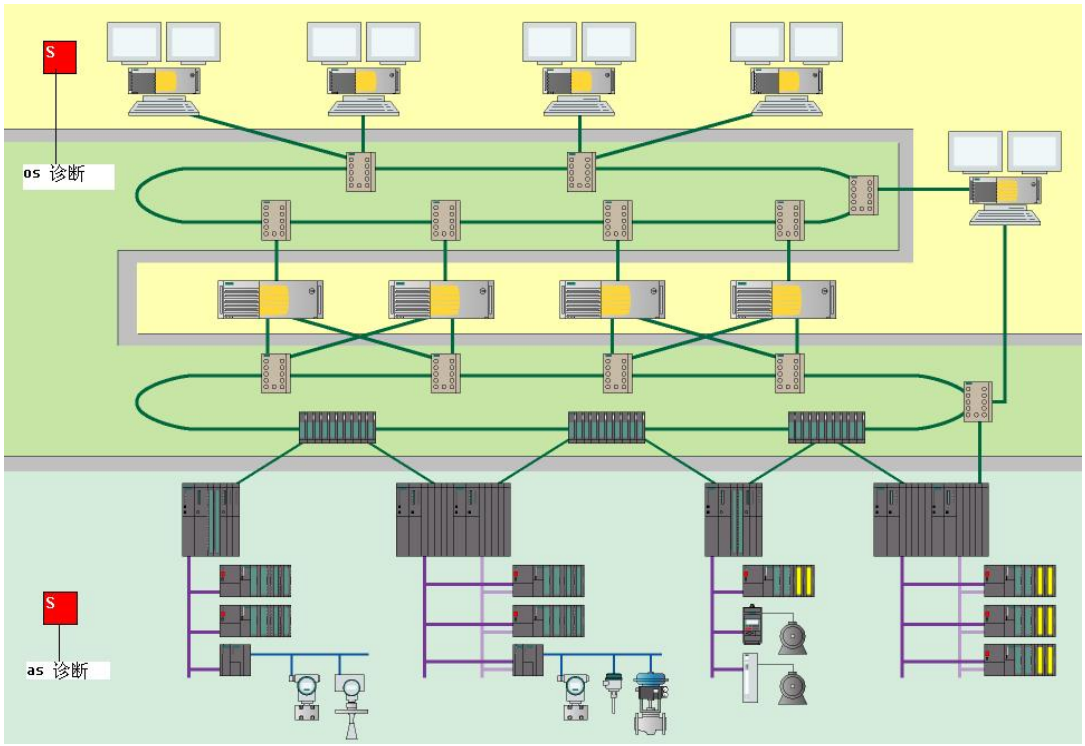


图 25：资产管理画面

可以看到，画面上有 OS 诊断和 AS 诊断。如果其中有故障，则是红色报警。点击 AS 的诊断按钮，进入 AS 诊断画面。

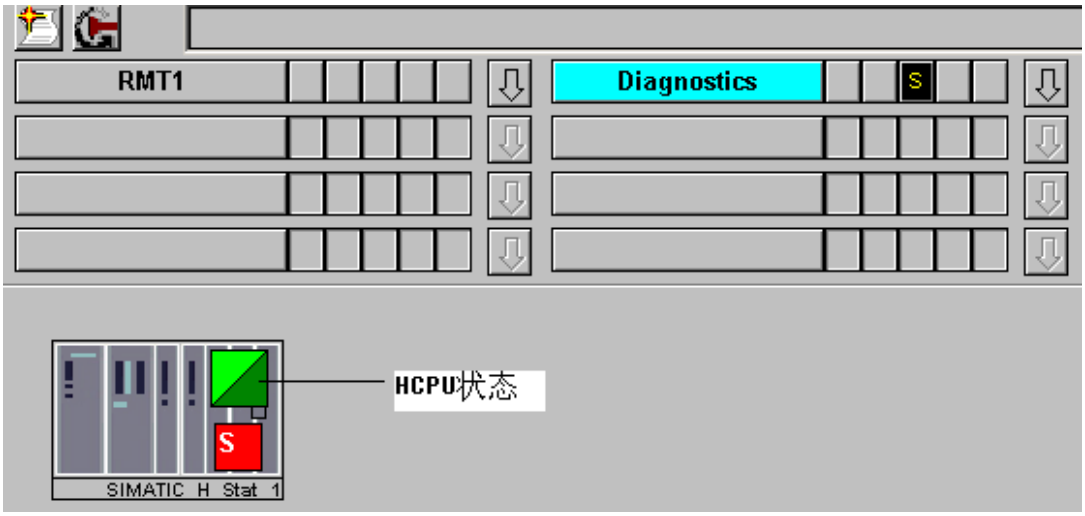


图 26： CPU 诊断画面

图上绿框代表 CPU 的状态良好，点击这个绿框，进入 CPU 详细状态。

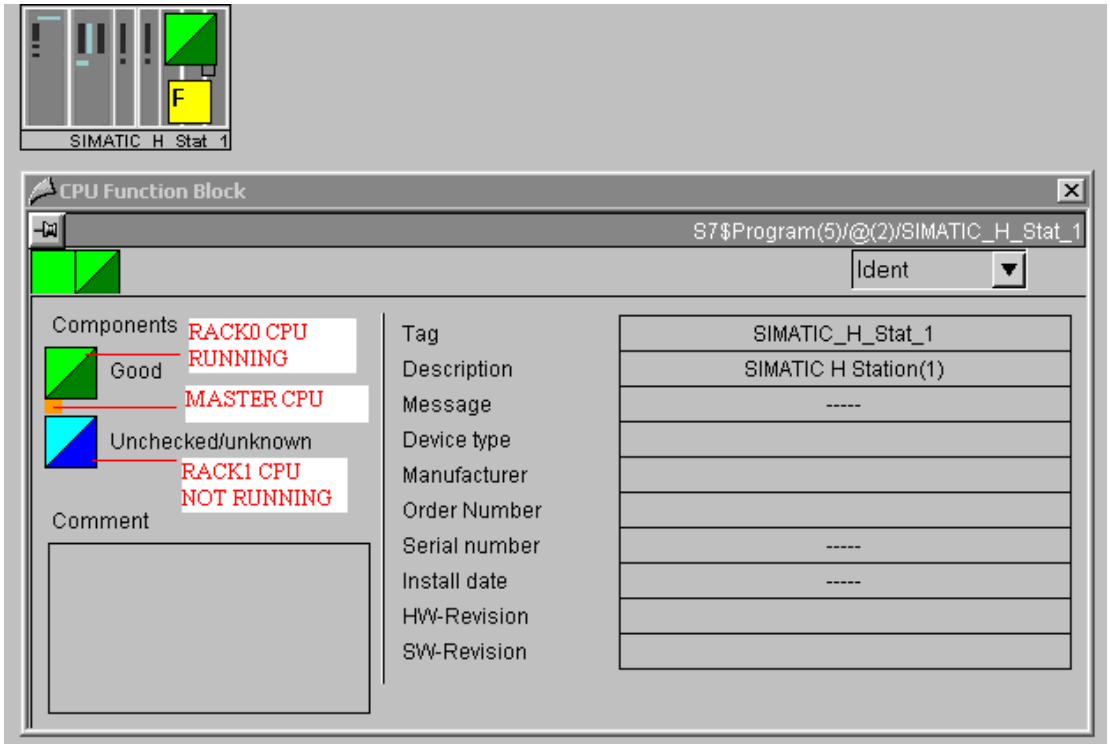


图 33：CPU 详细诊断画面

上图绿色方框表示，RACK 0 的 CPU 在运行状态，且为主 CPU；RACK 1 的 CPU 为备用站，且没有运行。

总结：资产管理诊断功能强大，而且组态简单方便，不仅能够诊断 CPU 的状态，也能够诊断 PC STATION，DP 网络，DP 从站，甚至 I/O 模板的状态都能够通过资产管理来监视；但是需要单独购买资产管理的授权。关于资产管理的详细使用信息请参考 PCS 7 手册或如下相关文档：（PCS 7 V7.0 版本后）

\SIEMENS\Step7\S7MANUAL\PCS 7GS\PCS 7_V70_in-practice_ASSET_en.pdf。