Network and Communication Technologies; Vol. 1, No. 2; 2012
ISSN 1927-064X E-ISSN 1927-0658
Published by Canadian Center of Science and Education

Diagnostic of Fault-tolerant System S7-400H

Anna Bystričanová[1]

[1] Department of Mechatronics and Electronics, University of Zilina, Slovakia

Correspondence: Anna Bystričanová, Department of Mechatronics and Electronics, University of Zilina, Univerzitna 1, 010 26 Zilina, Slovakia. E-mail: anna.bystricanova@siemens.com

Received: June 25, 2012 Accepted: August 20, 2012 Online Published: September 19, 2012

doi:10.5539/nct.v1n2p20 URL: http://dx.doi.org/10.5539/nct.v1n2p20

Abstract

In a standard technical or PLC programmer praxis we meet in our daily with an obligation to know to diagnose control processing units CPUs, industrial network or devices which are connected to the CPU. We can remove failures very quickly in dependence on the right fault recognition. There is a lot of industrial applications which are required continuous production. Stop such production can means large material claims. In a worse occurrence it means risk to person. Because of these, we choose control system SIMATIC S7-400H as a system, which we will monitor. These control systems are used for more difficult and demanding industrial solutions, where requires on a redundancy exist. This article provides you information about using of system and standard blocks in the operating system of CPU for SIMATIC S7-400H. We will show one way how to read out diagnostic datas from CPU through SZL function. We will use WinCC flexible for simply results visualization.

Keywords: diagnostic, CPU (central processing unit), PLC, SIMATIC S7-400H, SFC 51

1. Introduction

In the present automatic era resolutions big or smaller industrial applications are builted on programmable logic automats PLCs. We can see PLCs in a wide range of using. This could ensure easy and reliable runnig various technologies. At the beginning it is important to define type of a suitable CPU in accordance with customers requires and technology difficulties.

In this article we will pay attention to the CPU from a line of SIMATIC S7-400H, concrete to CPU 416 PN/DP. It is a redundant control system which is resistent to failures.

The reason of using such automation systems is to reduce production downtimes, regardless of whether the failures are cause by an error/fault or are due to maintenance work (Parrot & Venayagamoorthy, 2008).

The higher the costs of production stops, the greater the need to use a fault-tolerant system and at last the generally higher investment costs of fault-tolerant systems are soon recovered since production stops are avoided.

2. Redundant Automation System

Redundant automation systems are used in practice with the aim of achieving a higher degree of availability or fault tolerance.

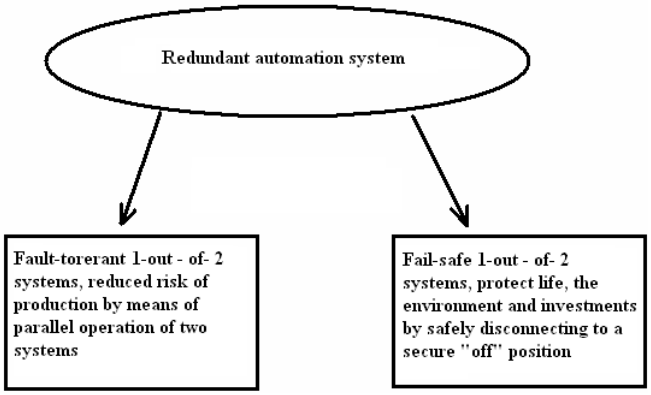

Figure 1. Possibilities of redundant automation systems (SIEMENS AG, 2012)

The S7–400H is a fault-tolerant automation system. We may only use it to control safety related processes if we have programmed and configured it in accordance with the rules for F systems (SIEMENS AG, 2012).

The S7-400H automation system satisfies the high demands on availability, intelligence, and decentralization placed on modern automation systems. It also provides all functions required for the acquisition and preparation of process data, including functions for the open-loopcontrol, closed-loop control, and monitoring of assemblies and plants (SIEMENS AG, 2012).

We also use redundant input/output modules to obtain a certain redundant automation system. Input/output modules are termed redundant when they exist twice and they are configured and operated as redundant pairs. The use of redundant I/O provides the highest degree of availability, because the system tolerates the failure of a CPU or of a signal module (Pfeffer, 2006).

The redundant structure of the S7-400H ensures requirements to reliability at all times. This means: all essential components are duplicated. This redundant structure includes the CPU, the power supply, and the hardware for linking the two CPUs.

3. SIMATIC S7-400H System

The basic SIMATIC S7-400H system consists of the hardware components required for a fault-tolerant controller. We can see hardware components for this system on the Figure 2. The basic system can be expanded with S7–400 standard modules (SIEMENS AG, 2012).

The two CPUs are the heart of the S7-400H. Use the switch on the rear of the CPU to set the rack numbers (SIEMENS AG, 2012). We will refer to the CPU in rack 0 as CPU 0, and to the CPU in rack 1 as CPU 1.

The UR2-H rack supports the installation of two separate subsystems with nine slots each, and is suitable for installation in 19" cabinets. We can also set up the S7-400H in two separate racks (SIEMENS AG, 2012).

The synchronization modules are used to link the two CPUs. They are installed in the CPUs and interconnected by means of fiber-optic cables (Franeková, Kállay, Peniak, & Vestenický, 2007). A fault-tolerant system requires 4 synchronization modules of the same type.

I/O modules of the SIMATIC S7 series can be used for the S7-400H. The I/O modules can be used in the following devices (SIEMENS AG, 2012):

- Central devices
- Expansion devices
- Distributed via PROFIBUS DP

The S7-400H supports the following communication methods and mechanisms (SIEMENS AG, 2012):

- Plant buses with Industrial Ethernet
- Point-to-point connection

Figure 3 shows how to connect remote input/output modules to the automation system.

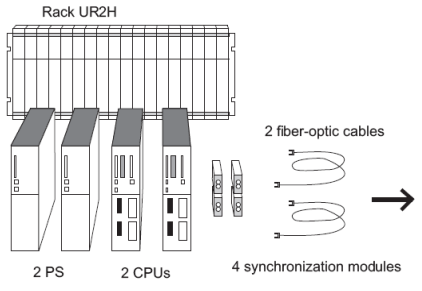

Figure 2. Hardware of the S7-400H system (SIEMENS AG, 2012)

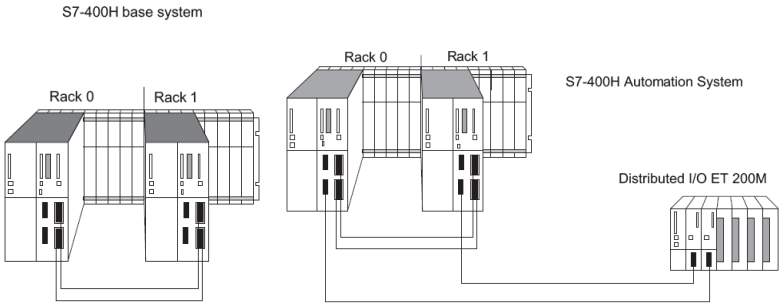

Figure 3. Hardware assembly (SIEMENS AG, 2012)

4. Operator Controls and Display Elements of the CPU 416-5 PN/DP

Each processor unit is equipped by LED signals for diagnostic automatic system status and failures.

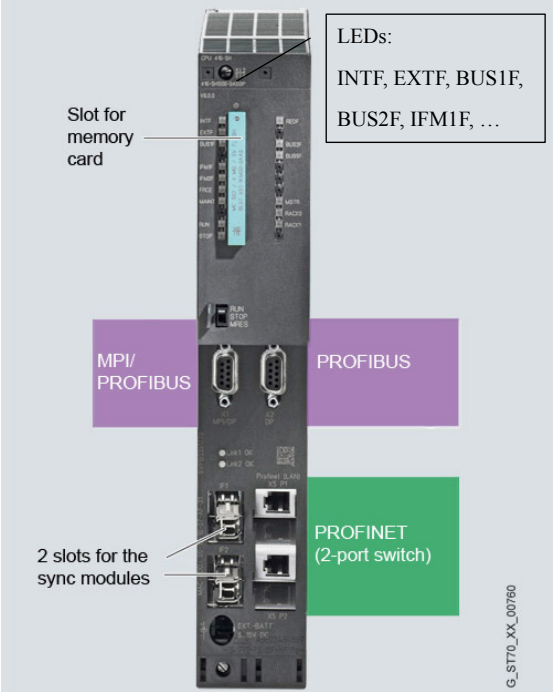

Figure 4. Operator control and display elements on the CPU

The following Table 1 shows an overview of the LED displays on the CPU 416-5H PN/DP (SIEMENS AG, 2012).

Table 1. LED displays on the CPU

LED	Meaning	LED	Meaning
SF	group error	RACK0	CPU in rack 0
INTF	Internal error	RACK1	CPU in rack 1
EXTF	External error	RACK2	CPU in rack 2
RUN	RUN mode	IFM1F	Error in synchronization module 1
STOP	STOP mode	IFM2F	Error in synchronization module 2
FRCE	Active force request	BUS3F	Bus fault on MPI/PROFIBUS DP interface 3
CRST	cold restart	MAINT	Signalization requirements of maintenance
BAF	battery fault/overload,	DC24V	DC24V
USR	user-defined	IF	init failure
USR1	user-defined	UF	user failure
BUS1F	Bus fault on MPI/PROFIBUS DP interface 1	MF	monitoring failure
BUS2F	Bus fault on PROFIBUS DP interface 2	CF	communication failure
REDF	Loss of redundancy/Redundancy fault	TF	task failure
MSTR	CPU controls the process		

The hardware of the CPU and operating system provide monitoring functions to ensure proper operation and defined reactions to errors. Various errors may also trigger a reaction in the user program. We can diagnostic some errors from the way of lightening LED signals.

We could show it in Table 2 below for LEDs RUN and STOP on the CPU (SIEMENS AG, 2012).

Table 2. RUN and STOP LEDs diagnostic (SIEMENS AG, 2012)

LED		Meaning
RUN	***STOP***	
Lit	Dark	The CPU is in RUN mode.
Dark	Lit	The CPU is in STOP mode. The user program is not being executed. Cold restart/restart is possible. If the STOP status was triggered by an error, the error indicator (INTF or EXTF) is also set.
Flashes 2 Hz	Flashes 2 Hz	CPU is DEFECTIVE. All other LEDs also flash at 2 Hz.
Flashes 0.5 Hz	Lit	HOLD status has been triggered by a test function.
Flashes 2 Hz	Lit	A cold restart/restart was initiated. The cold restart/warm start may take a minute or longer, depending on the length of the called OB. If the CPU still does not change to RUN, there might be an error in the system configuration, for example.
Dark	Flashes 2 Hz	Self-test when unbuffered POWER ON is running. The self-test may take up to 10 Minutes. Memory is being reset
Irrelevant	Flashes 0.5 Hz	The CPU requests a memory reset.
Flashes 0.5 Hz	Flashes 0.5 Hz	Troubleshooting mode.

5. Read Out Diagnostic Information by Step 7

For read out diagnostic datas by software Step 7 it is possible use som of the system blocks, which are integrated in this sotware.

With system function SFC 51 “RDSYSST (read system status), we read a system status list or a partial system status list (SIEMENS AG, 1995-2012).

We start the reading by assigning the value 1 to the input parameter REQ when SFC 51 is called. If the system status could be read immediately, the SFC returns the value 0 at the BUSY output parameter. If BUSY has the value 1, the read function is not yet completed (SIEMENS AG, 1995-2012).

5.1 Structure of SFC 51

By this function we can read out number of information in dependence on the actual type of CPU.

Basic system function structure we can see in the Table 3.

Table 3. Structure of SFC 51 function (SIEMENS AG, 1995-2012)

Param.	Declaration	Data Type	Description
REQ	INPUT	BOOL	REQ=1: Starts processing
SSL_ID	INPUT	WORD	SSL-ID of the system status list or partial list to be read.
INDEX	INPUT	WORD	Type or number of an object in a partial list.
RET_VAL	OUTPUT	INT	If an error occurs while executing the SFC, the RET_VAL parameter contains an error code.
BUSY	OUTPUT	BOOL	TRUE: Reading not yet completed.
SSL_HEADER	OUTPUT	STRUCT	Parameter with defined structure.
DR	OUTPUT	ANY	Destination area of the SSL list read or the SSL partial list read: • If you have only read out the header information of an SSL list, you must not evaluate DR but only SSL_HEADER. • Otherwise, the product of LENTHDR and N_DR indicates how many bytes were entered in DR.

SSL_HEADER

The SSL_HEADER parameter is a structure defined as follows:

SSL_HEADER: STRUCT

LENTHDR: WORD

N_DR: WORD

END_STRUCT

LENTHDR is the length of a data record of the SSL list or the SSL partial list.

• If you have only read out the header information of an SSL list, N_DR contains the number of data records belonging to it.

• Otherwise, N_DR contains the number of data records transferred to the destination area.

List, which we could read by the system function SFC 51, we can define by SSL_ID parameter. This also depends on the CPU type. It is possible to read about specify information in the Help in Step 7 or in manuals (SIEMENS AG, 2012).

6. Diagnostic Program

You can see how we can read out LED signals status on the practical example. We was working with SIMATIC station S7-400H with CPU 416-5H PN/DP. It is automatic redundant system.

Figure 5 shows our system hardware. You can see CPU and operator panel which are both connected on the Industrial Ethernet. We use SIMATIC HMI Station for results visualisating.

We have programed a function block in the Step 7, where we have inserted SFC 51 function with SSL_ID defined to 0174. We could obtain the status of the module LEDs.

If the H CPUs are in a non-redundant H mode, we obtain the LED status of the CPU addressed. If the H CPUs are in the RUN-REDUNDANT mode, the LED status of all redundant H CPUs is returned.

Figure 6 shows our programed function SFC 51 according our requirements. The index in this block contains required LEDs code. In our case it is LED signal for RACK 0.

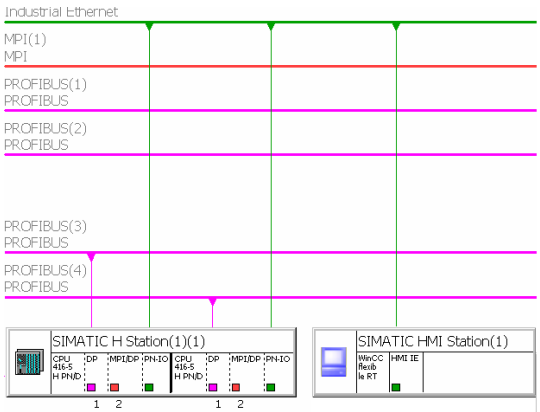

Figure 5. Hardware configuration control automation system

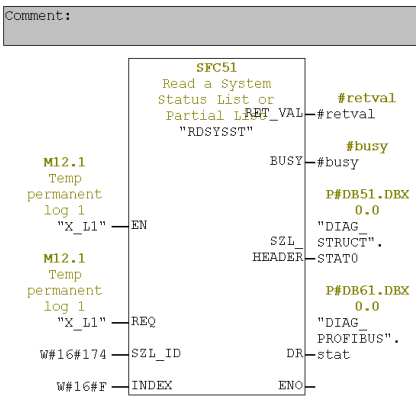

Figure 6. Programmed function SFC 51 in the STEP 7

We can see output structure SZL_HEADER which was desribed thereinbefore. Figure 7 shows data Block (DB) with just structure.

Address	Name	Type	Initial value
0.0		STRUCT	
+0.0	STAT0	STRUCT	
+0.0	STAT1	WORD	W#16#0
+2.0	STAT2	WORD	W#16#0
=4.0		END_STRUCT	
=4.0		END_STRUCT	

Figure 7. Data block with SZL_HEADER structure

A data record of partial list extract SSL-ID has the following structure:

Address	Name	Type	Initial value
0.0		STRUCT	
+0.0	stat	STRUCT	
+0.0	MASTERPLC	STRUCT	
+0.0	index	STRUCT	
+0.0	_0	BOOL	FALSE
+0.1	_1	BOOL	FALSE
+0.2	_2	BOOL	FALSE
+0.3	_3	BOOL	FALSE
+0.4	_4	BOOL	FALSE
+0.5	_5	BOOL	FALSE
+0.6	_6	BOOL	FALSE
+0.7	_7	BOOL	FALSE
+1.0	_01	BYTE	B#16#0
=2.0		END_STRUCT	
+2.0	led_on1	BYTE	B#16#0
+3.0	led_blink	BYTE	B#16#0
=4.0		END_STRUCT	
+4.0	STANDBYPLC	STRUCT	
+0.0	index	STRUCT	
+0.0	_0	BOOL	FALSE
+0.1	_1	BOOL	FALSE
+0.2	_2	BOOL	FALSE
+0.3	_3	BOOL	FALSE
+0.4	_4	BOOL	FALSE
+0.5	_5	BOOL	FALSE
+0.6	_6	BOOL	FALSE
+0.7	_7	BOOL	FALSE
+1.0	_01	BYTE	B#16#0
=2.0		END_STRUCT	
+2.0	led_on1	BYTE	B#16#0
+3.0	led_blink	BYTE	B#16#0
=4.0		END_STRUCT	
=8.0		END_STRUCT	
=8.0		END_STRUCT	

Figure 8. Data block to record information from SFC 51

Middle of this DB belongs CPU with MASTER function and the second is for CPU with STADBY function. You can see meaning single variables in the following Table 4 (SIEMENS AG, 2012).

Table 4. Meaning variables in the record for diagnostic

Name	Length	Meaning
cpu_led_ID	1 word	• Byte 0 Standard CPU: B#16#00 H-CPU: Bits 0 to 2: rack number Bit 3: 0=standby CPU, 1=master CPU Bits 4 to 7: 1111 • Byte 1: LED ID
led_on 1	1 byte	Status of the LED: 0: off 1: on
led_blink	1 byte	Flashing status of the LED: 0: not flashing 1: flashing normally (2 Hz) 2: flashing slowly (0.5 Hz))

7. LEDs Status

We have loaded our program for reading out LEDs status to the control system. This control system consists from 2 CPUs, one of them works as a MASTER, the other works as a STANDPY CPU.

At the Figure 9, we can see both CPU`s status and on the HMI vizualization we can see LED`s status. We can read out all this information by using SFC 51 functions which we call regularly in the organization block OB1 every scan.

We have switched the second CPU to MASTER's task. We can see this change from the actual state LEDs in the both CPU. See Figure 10.

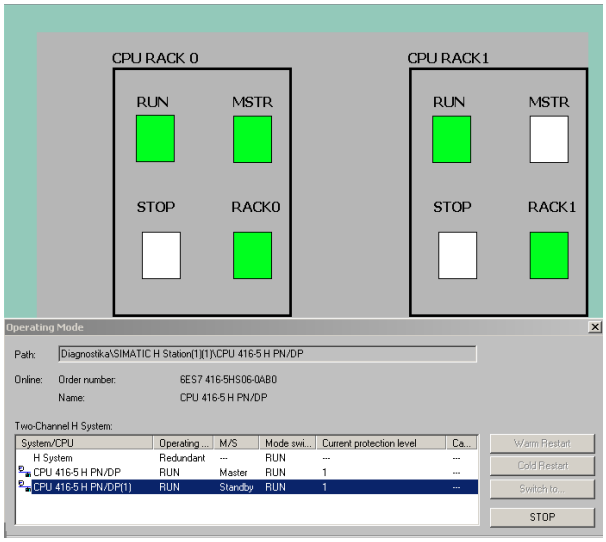

Figure 9. CPU SIMATIC S7-400H LEDs status

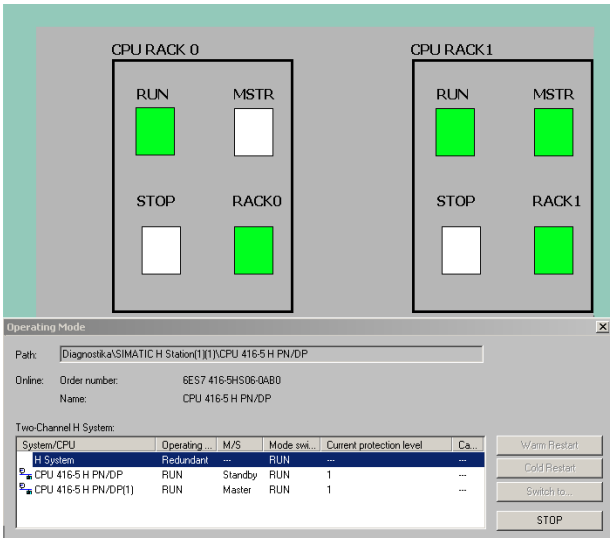

Figure 10. CPU SIMATIC S7-400H LEDs status after the change between CPU's status

8. Conclusion

In the present exist really high requirements to automation systems. This could be why are redundant automation systems used. These are used in practice with the aim of achieving a higher degree of availability or fault tolerance. Reduction production downtimes is a big advantage such control system.

We have chosen this automation system to describe it in this article for its popularity in the present automation world.

This system consists from two redundant control units, where one works as a MASTER and the other is in the STANDBY mode. In the case of crashing MASTER CPU, the second CPU is switched from STANDBY to MASTER function.

It is possible to do detailed diagnostic by various ways on the automation system. We have showed it by using system function SFC 51 with defined SSL_ID.

The results of our experiments are visible on the Figures 9 and 10. We can see there LEDs status both CPU. At the end we can assure about availability this function for diagnostic's purpose.

This article does not solve communication between HMI station an redundant automation system. This could be another topic.

Acknowledgements

This article was written with support of grant project VEGA 1/0943/11, Research of multi bin system for energy.

References

Franeková, M., Kállay, F., Peniak, P., & Vestenický, P. (2007). *Komunikačná bezpečnosť priemyselných sietí*. Žilina (SVK).

Obermaisser, R., & Schlager, M. (2007). *A Simulation Framework for VirtualIntegration of Integrated Systems.* In Proceedings of the IEEE International Conference on Computer as Tool, EUROCON. On page(s): 2208-2216. Warsaw, Poland, September 2007.

Papp, Z., Dorrepaal, M., & Verburg, D. J. (2003). *Distributed hardware-in-the-loop simu-lator for autonomous continuous dynamical systems with spatially constrained interactions*. In: Proceedings of the IEEE International Parallel and Distributed Processing Symposium, Nice, France.

Parrot, C., & Venayagamoorthy, G. K. (2008). *Real-Time Implementation of Intelligent Modeling and Control Techniques on a PLC Platform.* Industry Applications Society Annual Meeting. pp. 1-7.

Pfeffer, A. (2006). *Průmyslová síť Ethernet – srovnání uklidňuje*. In: Automatizace 2/2006, pp. 99-101 (CZE).

Rankin, D. J., & Jiang, J. (2011). *A Hardware-in-the-Loop Simulation Platform for the Verification and Validation of Safety Control Systems.* In: Nuclear Science, IEEE Transaction, pp. 468-478. USA.

SIEMENS AG. (1995-2012). STEP 7 Programme manual. *STEP 7, Version 5.5.* SIEMENS.

SIEMENS AG. (2012). *Manual for users.* Retrieved from: https://support.automation.siemens.com/WW/llisapi.dll?query=support.automation.siemens.com%2F.+S7_400_h_en_en-US&func=cslib.cssearch&content=adsearch%2Fadsearch.aspx&lang=en&siteid=csius&objaction=cssearch&searchinprim=&nodeid99=&x=0&y=0.