Research and Design of Vital Computer Applied to Autonomic Machine

Xiaohe Sun* and Shenghua Dai

¹School of Electronic and Information Engineering, Beijing Jiaotong University, Beijing, China

*Corresponding author

Abstract—This paper introduces a simplified TMR VC system for self-discipline machine in railway stations by analyzing the reliability and safety between hot-standby and TMR. Moreover, a TMR comparator based on FPGA is designed to improve the speed of data transmission and comparison.

Keywords—TMR; fault-tolerant; redundancy technique; fail-safe

I. INTRODUCTION

The technical conditions for FZ-CTC (interim) is formulated by CHINA RAILWAY to solve the problem that a passenger train is running with the goods train on the same line and the large amount of shunting operation [1]. At the same time, the autonomous machine is the core equipment in the FZ-CTC system. The autonomic machine has the ability to generate a path action command and can turn instructions into commands. After that the plan of adjusting train operation can be automatically parsed into the sequence of train route operation instructions by it. Then, the autonomic machine issues orders to the interlock system. So the autonomic machine plays an important role in the CTC system, and it is the hub between center and station.

Generally, the FZ-CTC system works at self-regulation mode which can reduce effectively errors made by human, so the automatic machine should possess higher reliability and safety. The fault-avoidance and fault-tolerance are the commonest techniques to improve reliability and safety [2]. The technique of fault-avoidance need a chip with high reliability to reduce the failure probability, but the reliability of all the chips can't be 100%. So we can adopt the fault-tolerant technique to improve the reliability. This technique has been used in aircraft control fields and industrial control fields [3-4]. There are a lot of methods about Fault-tolerance, such as hardware redundancy, software redundancy, time redundancy and so on. The hardware redundancy and software redundancy are the most commonly used methods [5]. At present, the hot-standby, 2 out of 3 and double 2 out of 2 architecture which belong to hardware redundancy have been researched more than others.

The autonomic machine basically adopts the hot-standby architecture. Markov analysis is applicable to redundant system model which is mentioned at EN50128. So this method is used to analyze the hot-standby and 2 out of 3 about reliability and safety in this paper. At the same time, a new comparator is designed in this paper.

II. TWO KINDS OF HARDWARE REDUNDANT ARCHITECTURES

A. Hot-Standby Architecture

FIGURE I. HOT-STANDBY ARCHITECTURE

Hot-standby is also called standby reserve system which is a kind of dynamic redundancy [6]. The primary CPU and standby CPU work simultaneously. If the primary CPU breaks down, the standby CPU will work instead of the primary CPU immediately.

Because of this mechanism, the reliability and safety of hot-standby architecture is very high [7]. Meanwhile this architecture possesses a great advantage in railway because of its features of fast switching and continuous working. The Markov analysis is referenced in Literature 8.

B. 2 out of 3 Architecture

FIGURE II. 2 OUT OF 3 ARCHITECTURE

2 out of 3 architecture is a kind of static redundancy. The three CPUs run independently of each other and the outputs of each will be sent to the comparator. If two or three outputs are

consistent, the output is effective. If two CPUs have the same failure and export the same data, the 2 of 3 system will fail. However, the probability of this status is very low, so this architecture is commonly used in train control on board equipment and computer interlocking system. The reliability and safety of this architecture will be analyzed in the next section.

Firstly, there is a conservative view that a danger will be caused if two CPUs malfunction unmeasurably, but actually this will never happen. So the safety which is calculated will be lower than the actual. Even so, the result still have some reference value.

Figure 3 shows the Markov model of 2 out of 3 system with considering failure maintenance rates.

FIGURE III. STATE-TRANSITION DIAGRAM

The definition of the state of the system shows as follows:

State0: All three units work properly.

State1: One of the units has a measurable fault, others work properly. Work mode turns to 2 out of 2 mode.

State2: Under the 2 out of 2 mode, one of the units has a measurable fault. The system works at a fail-safe state.

State3: Under the 2 out of 2 mode, one of the units has an unmeasurable fault. The system does not work.

State4: One of the units has an unmeasurable fault, others work properly.

State5: Two of the units have an unmeasurable fault, the other one works properly. The system produces a dangerous result.

State6: Under the status of State 5, the normal unit has a measurable fault. The system produces a dangerous result.

State7: Under the status of State 6, one of the units which has an unmeasurable fault has a measurable fault. The system does not work.

State8: All three units have an unmeasurable fault. The system produces a dangerous result.

The corresponding differential equation can be obtained by the state transition diagram:

$$\begin{cases} \dot{p}_0(t) = -3\lambda p_0(t) + 3\mu p_1(t) \\ \dot{p}_1(t) = 3\lambda c p_0(t) - (2\lambda + 3\mu) p_1(t) + 2\mu p_2(t) + \lambda c p_4(t) \\ \dot{p}_2(t) = 2\lambda c p_1(t) - 2\mu p_2(t) \\ \dot{p}_3(t) = 2\lambda(1-c) p_1(t) + 2\lambda c p_4(t) + 2\lambda c p_5(t) + 2\mu p_7(t) \\ \dot{p}_4(t) = 3\lambda(1-c) p_0(t) - \lambda(2+c) p_4(t) \\ \dot{p}_5(t) = 2\lambda(1-c) p_4(t) - \lambda(1+2c) p_5(t) + \mu p_6(t) \\ \dot{p}_6(t) = \lambda c p_5(t) - (2\lambda c + \mu) p_6(t) \\ \dot{p}_7(t) = 2\lambda c p_6(t) - 2\mu p_7(t) \\ \dot{p}_8(t) = \lambda(1-c) p_5(t) \end{cases}$$

The initial condition of the differential equation is $P_0 = [1 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0]$, the reliability and safety can be obtained by solving the differential equation. As is shown in Formula 1 and Formula 2.

$$R(t) = p_0(t) + p_1(t) + p_2(t) \quad (1)$$

$$S(t) = 1 - p_5(t) - p_6(t) - p_8(t) \quad (2)$$

III. ANALYSIS OF RELIABILITY AND SAFETY

The advantages and disadvantages of hot-standby and 2 out of 3 can't be shown well by differential equation, so this paper use the ode45 function of MATLAB to solve the differential equation.

In the simulation, it assumes that the system runs a total 10000 hours, the failure rate is $0.0001/h$, maintenance rate $\mu = 0.1$, fault-coverage rate $c = 0.9$, the results show as Figure 4 and Figure 5.

FIGURE IV. RELIABILITY CURVE OF HOT-STANDBY AND 2 OUT OF 3

FIGURE V. SAFETY CURVE OF HOT-STANDBY AND 2 OUT OF 3

As Figure 4 shows, the 2 out of 3 is better than hot-standby in the reliability before 5000 hours, but the hot-standby will be better after 5000 hours, because the architecture of 2 out of 3 is more complex than hot-standby.

As Figure 5 show, both the 2 out of 3 and hot-standby are at high level in safety, but 2 out of 3 possesses a distinct advantage over hot-standby.

Fault coverage rate is an important parameter for the reliability and safety in the railway equipment. This paper analyzes the effect of different fault coverage rate on 2 out of 3 system about the reliability and safety.

FIGURE VI. THE EFFECT OF μ FOR RELIABILITY

As Figure 6 shows, the reliability of the system will be improved a lot when the fault coverage rate is considered. Although the reliability increases with the increase of the fault coverage rate, when the fault coverage rate comes to a certain value, the effect on reliability is not obvious.

So it follows that introduction of fault coverage rate is an important way to improve reliability, but it is not the decisive parameter for the system with fault coverage rate.

FIGURE VII. THE EFFECT OF μ FOR SAFETY

As Figure 7 shows, the safety of the system decreases as fault coverage rate increases, but it still stays at a relatively high level.

In general, although the safety will decrease when fault coverage rate is considered, the reliability will increase a lot. So fault coverage rate plays a key role in the whole system.

In fact, the VC usually adopts more than one architecture, so it always combines the advantage of all kinds of architecture. Because the autonomous machine should follow the fail-safe principle, this paper will adopt 2 out of 3 architecture to finish the main control unit of VC.

IV. VITAL COMPUTE ARCHITECTURE

FIGURE VIII. THE ARCHITECTURE OF VITAL COMPUTER

Figure 8 shows the main architecture of the whole system, which contains CPU module, communication equipment, communication interface and so on. The CPU communicates with each other through fast Ethernet as well as RS485. Meanwhile they can exchange data through the comparator designed below. Considering synchronous technology and comparison technology are the most important technical of 2 out of 3, this paper will give a brief description about both of these techniques in the next section.

A. Design of Synchronization

The three CPUs should keep synchronous at the time of data reception, data processing, data comparison and data transmission. So the design of synchronization is very important in the development of VC. The most common methods of synchronization include clock synchronization, fixed cycle synchronization and task synchronization [9].

Clock synchronization is easier to implement, but this approach may cause common mode failures easily. So this paper adopts task synchronization to finish the design. The specific work processes is shown as Figure 9.

FIGURE IX. THE FLOW CHART ABOUT TASK SYNCHRONIZATION

The task synchronization used in this design is loose synchronization. The CPU1 will for wait a period of time to receive messages from CPU3 when it has received messages from CPU2. The CPU3 will be regarded as failure if CPU1 does not receive messages from it. So the system can keep synchronous in this way. It is designed specifically in the comparator.

B. Design of Vote

The commonly used data validation techniques can be divided into software comparison and hardware comparison in the VC system [10]. Software comparison means that one CPU receives data from other CPUs, then it will process the data by itself. But the hardware comparison needs another device which can check the information independently. The advantage of hardware is that it runs faster than software, because it does not take up CPU resources. This design will combine software comparison and hardware comparison.

1) Software comparison:

The three CPUs communicate with each other through Ethernet and RS485 bus which is a spare method. One of them will be the primary processing unit at random. If the results compared are the same, the system works properly, otherwise the system will determine which one is a failure. If the CPU fails many times, it will be severed, and the system will work at 2 out of 2 mode. The advantage of software is that it is easy to achieve, but it runs a little slow.

2) Hardware comparison:

The design of the comparator needs to meet the real time of communication and the rapidity of comparison. The FPGA uses the hardware description language to program, it can not only solve the shortage of custom circuit, but also overcome the limitations of the number of existing programmable devices. At the same time, it is more flexible than traditional MCU. So this paper designs the comparator based on FPGA.

FIGURE X. THE STRUCTURE OF COMPARATOR

Figure 10 shows the architecture of comparator designed on FPFA. The working process can be briefly described as follows: First, the comparator receives the data and control commands from CPU. Then it will check the data, and according to the checking result, it will control the output unit. After that the result will be sent to 2 out of 3 voter. Finally the state of safety switch circuit is controlled.

The bus comparator contains FIFO module, state control module, CRC module, data comparison module, output control module, self-checking module and 2 out of 3 voting module. If the working state of the CPU and the comparator are not synchronized, it is likely to lead to blocking or even loss if they transmit data directly, so this paper adopts FIFO to solve this problem [11]. Each module of the comparator works out of series and all of them are connected with the state control module which is the core of the comparator. The CRC module can determine whether the data which is received by the comparator is correct. The output control module will send the result to 2 out 3 voter. The voting unit can determine which CPU is wrong. At the same time, it also has the function to record the error messages. If the error reaches limit, it will mark a feedback to the state control module, and the communication of the CPU will be severed.

The bus comparator is disconnected with the CPUs after the system has begun to work. Only when the initialization of CPUs is finished, they will send a command to bus comparator, then the comparator can work properly. It is designed to prevent the unstable state at the beginning of the current.

V. CONCLUSION

This paper proposes that the autonomic machine is the core of the FZ-CTC, then it comes to the conclusions as follows by analyzing hot-standby and 2 out of 3 through Markov analysis method.

(1)The reliability of 2 out of 3 architecture is higher than hot-standby architecture when the running time is not long. Meanwhile the safety of 2 out of 3 is much higher than hot-standby architecture. Because the autonomic machine must follow the fail-safe principle, the main processing unit should adopt the 2 out of 3 architecture, and other units which need higher reliability should adopt the hot-standby architecture.

(2)The maintenance rate has a huge impact on the reliability of system. However, as the maintenance rate increases, the reliability of system doesn't increase in proportion. So it draws a conclusion that when the maintenance rate reaches a certain value, other parameters affect the reliability of the system more, such as the fault coverage rate.

(3)This paper combines the software comparison and hardware comparison to design the bus comparator which can meet the real-time, reliability and safety of the system.

ACKNOWLEDGMENT

This paper is part of the achievements of the CTCS-3 virtual simulation experiment platform project. Thanks Professor Dai for his guidance.

REFERENCES

- [1] Guo Hui,Wang xiujuan. FZj-CTC model of Centralized Traffic Control System[J]. Raliway Signalling & Communication, 2005, (06):11-14.
- [2] Xue Hua. The design of the microcomputer interlock control system of lanzhou petrochemical company railway [D]. Lanzhou University of Technology,2006.
- [3] International Electrotechnical Commission. IEC 61508-1997 Functional Safety of Electrical/Electronic Programmable Electronic Safety-related Systems[S].1997.
- [4] Levi S T, Agrawala A K. Fault-tolerant Computer System Design[M]. [S.l.]:McGraw-Hill,1994.
- [5] Shen Meilan. protective measures of fail-safe about computer interlocking system [J]. information technology,2010,(01):831-832.
- [6] Wan Jing. Fault tolerant technology of computer interlocking system[J]. Policy & Management,2010,(06):76-77.
- [7] Zhao Bingxian. The design of vital computer platform for CBTC [D].Beijing Jiaotong University,2016.
- [8] Sun Lei,Xu Hongze. The analysis of safety and availability about hot-standby computer interlocking system [J].
- [9] Xu Xiaoyan,Chen Wensai. The design and cpplication of a kind of vital computer about rail transsit[J]. Informatization Research, 2009,35(8):32-36.
- [10] Gang Jianlei,Han Anping,Hu Yanlai. The design based on FPGA of 2 out of 2 data comparator [J]. Electronic Engineering & Product World for Engineering Managers & Desingers,2012,06:56-58.
- [11] Zhang Qian. The design and formal validation about hardware platforms based on programmable logic[D].Beijing Jiaotong University,200