

Fault Tolerant Configuration of Distributed Discrete Controllers

Yasutaka Fujimoto, *Member, IEEE* and Takashi Sekiguchi, *Member, IEEE*

Abstract

This paper presents a fault tolerant configuration for a multiple discrete control system. The distributed control nodes, such as programmable logic controllers, communicate over networks. The design methodology of an additional redundant controller using Galois field and an error-detecting code is proposed. The proposed method is implemented and tested on distributed controllers connected on a network. It is shown that the system is functional even if one of the controllers fails. From a reliability analysis, it is also shown that the proposed design method drastically improves the mean time to failure of the discrete control system.

Keywords: Programmable logic controller, Discrete controller, Fault tolerance, Reliability, Galois field

I. INTRODUCTION

Discrete controls with programmable logic controllers (PLCs) are widely used for manufacturing automation and process control. Recently years, the efficiency of production systems has improved. It is required to reduce stocks on the manufacturing lines as much as possible. Faults of machinery and stops of lines in the production systems will bring heavy losses. In this paper, a novel method of fault tolerant configuration of discrete controllers is presented. The proposed system can continue operations even if one of any controllers fails. Unlike fully redundant control systems, the proposed system requires minimal additional cost and increases their reliability.

In the field of feedback control engineering, there are two types of strategies for fault tolerant control. One is based on a design of a controller that simultaneously stabilizes both the non-faulty and faulty plants[2][9]. Another is based on fault detection and controller reconfiguration[6][7]. These conventional control systems are robust against a pre-defined set of faults. On the other hand, the major concern in this paper is a fault of one of a set of distributed discrete controllers.

Section II gives a basic model of the controllers using Galois field. Next, a self-identifying failure recovery approach using parity code is derived in Section III. Section IV analyzes the reliability of the proposed system. Finally, in Section V, fault tolerance is demonstrated using experimental results for an illustrative example of a material handling system.

II. MODELING OF CONTROLLERS

Consider N non-homogeneous controllers shown in Fig. 1 which are independently designed to realize each specified function. Each controller has its own remote input/output interface. Assume that all of the controllers and the remote I/Os are connected over a communication network. Suppose that the i th controller is represented by a *Ladder Diagram*. The Ladder Diagram can be transformed into the form[8]

$$x_i(t+1) = f_{Bi}(x_i(t), u_i(t)) \quad (1)$$

$$y_i(t) = h_{Bi}(x_i(t), u_i(t)) \quad (2)$$

where x_i is a n_i -dimensional state vector whose elements are in $\{0, 1\}$, u_i is a m_i -dimensional input vector with elements in $\{0, 1\}$, and y_i is a p_i -dimensional output vector with elements in $\{0, 1\}$. The functions f_{Bi} , h_{Bi} consist of Boolean operation i. e., logical sums \vee , logical products \wedge , and negation $\bar{}$.

The function f_{Bi} in Boolean algebra is transformed into a function f_i in Galois field as follows. $GF(2)$ represents the Galois field with two elements in $\{0, 1\}$. Four arithmetic operations are defined in the Galois field[3][4]. For any two variables $\{a, b\}$ in the Boolean function $f_{Bi}(\cdot)$, consider the transformations

$$a \vee b \rightarrow a + b + ab, \quad a \wedge b \rightarrow ab, \quad \bar{a} \rightarrow 1 + a. \quad (3)$$

These are also applied to the function $h_{Bi}(\cdot)$.

These transformations yield an equivalent representation of the discrete systems on Galois field

$$x_i(t+1) = f_i(x_i(t), u_i(t)) \quad (4)$$

$$y_i(t) = h_i(x_i(t), u_i(t)) \quad (5)$$

where f_i and h_i are polynomial functions such that $f_i : GF(2)^{n_i} \times GF(2)^{m_i} \mapsto GF(2)^{n_i}$ and $h_i : GF(2)^{n_i} \times GF(2)^{m_i} \mapsto GF(2)^{p_i}$. Now we can write $x_i(t) \in GF(2)^{n_i}$, $u_i(t) \in GF(2)^{m_i}$, and $y_i(t) \in GF(2)^{p_i}$. The arithmetic in Galois field is wider than that in Boolean algebra. Table I shows the arithmetic in Galois field and in Boolean algebra.

Also the representations (4)(5) over Galois field can be inversely transformed into Boolean systems (1)(2) by

$$a + b \rightarrow (a \wedge \bar{b}) \vee (\bar{a} \wedge b), \quad ab \rightarrow a \wedge b. \quad (6)$$

Generally, the state space becomes a metric space if a Galois field is introduced. The metric space is needed to express a dynamic transition of a discrete state because a fluctuation of a transition is based on a concept of a distance.

Conventionally, a concept of a distance was not considered in an analysis of discrete systems because it uses a Boolean algebra. Boolean algebra is applicable not to a dynamic transition analysis but to a kinematic structure analysis.

III. SELF-IDENTIFYING FAILURE RECOVERY

A. Design of A Redundant Controller

In this section, a recovery method from a self-identifying failure based on a simple parity code is proposed. One redundant controller PLC_{N+1} is added as shown in Fig. 2 to improve reliability of the whole control system against a fault of controllers. The basic idea is based on a parity code. One redundant controller that dynamically generates parity output at any moment $\forall t \geq 0$ can be designed as follows:

$$x_{N+1}(t+1) = f_{N+1}(x_{N+1}(t), u_{N+1}(t)) \quad (7)$$

$$y_{N+1}(t) = h_{N+1}(x_{N+1}(t), u_{N+1}(t)) \quad (8)$$

where $x_{N+1}(t) = [x_1(t), x_2(t), \dots, x_n(t)]^T \in GF(2)^{n_{N+1}}$ is an extended state variable which has copies of states of all sub-modules, where $n_{N+1} = n_1 + n_2 + \dots + n_N$. Also $u_{N+1}(t) = [u_1(t), u_2(t), \dots, u_n(t)]^T \in GF(2)^{m_{N+1}}$ is an extended input variable which includes inputs of all sub modules, where $m_{N+1} = m_1 + m_2 + \dots + m_N$. The signals $u_i(t)$ are available if we use a communication network with broadcast type protocols such as *UDP/IP*.

The function f_{N+1} consists of copies of the dynamics of all sub-modules, i. e. $f_{N+1} = [f_1, f_2, \dots, f_n]^T$. The output is $y_{N+1}(t) \in GF(2)^{p_{N+1}}$ where $p_{N+1} = \max(p_1, p_2, \dots, p_N)$. The function h_{N+1} is given by

$$h_{N+1} = h_1 + h_2 + \dots + h_N. \quad (9)$$

Theorem 1: When there is no failure, the output y_{N+1} of the controller (7)(8) satisfies the parity condition for $\forall t \geq 0$

$$y_1(t) + y_2(t) + \dots + y_N(t) + y_{N+1}(t) = 0 \quad (10)$$

Proof: From equations (5) and (9), we compute

$$\begin{aligned}
& y_1(t) + y_2(t) + \cdots + y_N(t) + y_{N+1}(t) \\
&= h_1(x_1(t), u_1(t)) + h_2(x_2(t), u_2(t)) + \cdots + h_N(x_N(t), u_N(t)) + h_{N+1}(x_{N+1}(t), u_{N+1}(t)) \\
&= h_1(x_1(t), u_1(t)) + h_2(x_2(t), u_2(t)) + \cdots + h_N(x_N(t), u_N(t)) \\
&\quad + h_1(x_1(t), u_1(t)) + h_2(x_2(t), u_2(t)) + \cdots + h_N(x_N(t), u_N(t)) \\
&= 0
\end{aligned} \tag{11}$$

Note that the Galois field operation $+$ is equivalent to *Exclusive OR*. Hence the sum of two equal functions is zero. ■

B. Failure Recovery

If i th controller fails at time $t \geq t_1$, the possible outputs $y_i(t)$ is $y_i(t) = \xi(t)$ for $t \geq t_1$ where $\xi(t) \in GF(2)^{p_i}$ is noise. Assume that the failure is detectable for all other non-faulty controllers and the remote I/Os. Let the failure flag for the i th controller be $\theta_i(t)$. We define $\theta_i(t) = 1$ if the i th controller fails.

Theorem 2: The estimated output of the i th controller $\hat{y}_i(t)$ including both the failure case $\theta_i(t) = 1$ and the non-failure case $\theta_i(t) = 0$ is given by

$$\hat{y}_i(t) = (1 - \theta_i(t))y_i(t) + \theta_i(t) \sum_{j \neq i}^{N+1} y_j(t) \tag{12}$$

Proof: If there is no failure of the i th controller, the estimated output $\hat{y}_i(t) = y_i(t)$ because $\theta_i(t) = 0$. In a failure case $\theta_i(t) = 1$, from (8)

$$\begin{aligned}
\hat{y}_i(t) &= \sum_{j \neq i}^{N+1} y_j(t) \\
&= \sum_{j \neq i}^N h_j(x_j(t), u_j(t)) + h_{N+1}(x_{N+1}, u_{N+1}) \\
&= \sum_{j \neq i}^N h_j(x_j(t), u_j(t)) + \sum_{j=1}^N h_j(x_j(t), u_j(t)) \\
&= h_i(x_i(t), u_i(t)) \\
&= \bar{y}_i(t)
\end{aligned} \tag{13}$$

where $\bar{y}_i(t)$ is the desired output of the i th controller. ■

The $O(N)$ of calculation (12) must be implemented in intelligent remote I/O interfaces.

IV. RELIABILITY ANALYSIS

Assume that failure rate of each controller is constant and all controllers have same failure rate. Then the failure probability density function is $p(t) = \lambda e^{-\lambda t}$ where λ is failure rate. The probability distribution function $F(t) = \int_0^t p(\tau) d\tau = 1 - e^{-\lambda t}$ represents the proportion that fail until time t . The *Reliability* $R(t)$ is defined by $R(t) = 1 - F(t) = e^{-\lambda t}$ which represents the proportion that survive to time t .

The reliability analysis described in this section is based on a three-state Markov process model[1][5] as shown in Fig. 3. In this figure, S_0 represents a state that all controllers are functional. S_1 is a state that one controller is down and others are functional. Due to the redundancy, the whole system is still functional even if only one controller fails. S_2 is a state that more than two controllers are down, which means the whole system fails.

The diagram is derived from the fact that the transition probability from S_0 to S_1 is $(N + 1)\lambda\delta t$ for short period δt . The failure rate for one controller out of $N + 1$ controllers is $N + 1$ times larger than

that for one controller. The transition probability from S_1 to S_2 is also $N\lambda\delta t$. Then the transition probability from S_1 to S_0 is $\mu\delta t$ where μ is repair rate for one controller.

Let the probability of each state S_i be $P_i(t)$ for $0 \leq i \leq 2$. Then $\delta t \rightarrow 0$ yields a differential equation:

$$\frac{d}{dt} \begin{bmatrix} P_0(t) \\ P_1(t) \\ P_2(t) \end{bmatrix} = \begin{bmatrix} -(N+1)\lambda & \mu & 0 \\ (N+1)\lambda & -(N\lambda + \mu) & 0 \\ 0 & N\lambda & 0 \end{bmatrix} \begin{bmatrix} P_0(t) \\ P_1(t) \\ P_2(t) \end{bmatrix}. \quad (14)$$

From the fact that the solution of the linear differential equation $\frac{d}{dt}P(t) = AP(t)$ is $P(t) = e^{At}P(0)$, the solution of (14) with the initial condition $P(0) = [1, 0, 0]^T$ is given by

$$P_0(t) = \frac{(1 + e^{\theta t})\theta + (e^{\theta t} - 1)(\mu - \lambda)}{2\theta e^{((2N+1)\lambda + \mu + \theta)t/2}} \quad (15)$$

$$P_1(t) = \frac{(e^{\theta t} - 1)(N+1)\lambda}{\theta e^{((2N+1)\lambda + \mu + \theta)t/2}} \quad (16)$$

$$P_2(t) = \frac{(1 - e^{\theta t})((2N+1)\lambda + \mu) - (1 + e^{\theta t})\theta}{2\theta e^{((2N+1)\lambda + \mu + \theta)t/2}} + 1 \quad (17)$$

where $\theta = \sqrt{\lambda^2 + 2(2N+1)\lambda\mu + \mu^2}$.

Because both S_0 and S_1 are functional states and only the state S_2 indicates a failure, the reliability function $R_{all}(t)$ for whole system is obtained as

$$R_{all}(t) = P_0(t) + P_1(t) = 1 - P_2(t) = \frac{(e^{\theta t} - 1)((2N+1)\lambda + \mu) + (1 + e^{\theta t})\theta}{2\theta e^{((2N+1)\lambda + \mu + \theta)t/2}} \quad (18)$$

Also probability density function for system fault is obtained $p_{all}(t) = dP_2(t)/dt$.

The *Mean Time To Failure* ($MTTF_{all}$) for whole system is calculated by

$$\begin{aligned} MTTF_{all} &= \int_0^{\infty} tp_{all}(t)dt = \int_0^{\infty} R_{all}(t)dt \\ &= \frac{(2N+1)\lambda + \mu}{N(N+1)\lambda^2} \end{aligned} \quad (19)$$

$$= \frac{((2N+1)MTTR + MTBF)MTBF}{N(N+1)MTTR} \quad (20)$$

where $MTBF$ represents the *Mean Time Between Failure* of each controller which corresponds to $MTBF = 1/\lambda$. And $MTTR$ represents the *Mean Time To Repair* of each controller which corresponds to $MTTR = 1/\mu$. The reliability of the control system will be drastically improved because usually the failure time is extremely longer than the repair time, i. e., $MTBF \gg MTTR$.

Comparison to Conventional Configurations

In the case of a conventional configuration without redundancy, the $MTTF$ of the whole system becomes $MTTF_{no} = MTBF/N$ which is obviously shorter than that for the proposed configuration. On the other hand, the reliability function of the conventional full duplex system as shown in Fig. 4 is given by

$$R_{dup}(t) = \left(\frac{(e^{\theta t} - 1)(3\lambda + \mu) + (1 + e^{\theta t})\theta}{2\theta e^{(3\lambda + \mu + \theta)t/2}} \right)^N \quad (21)$$

The mean time to failure of the full duplex system is obtained by $MTTF_{dup} = \int_0^{\infty} R_{dup}(t)dt$.

Fig. 5 shows the numerical comparison of these configurations at the condition $MTBF = 10,000$ hours and $MTTR = 24$ hours. The numbers of total controllers are $N + 1$ for the proposed system, $2N$ for the full duplex system, and N for the non-redundant system. Although $MTTF$ of the proposed system is shorter than that of full duplex system, the proposed system requires only one redundant controller. With minimal additional cost, the proposed method still improves the reliability drastically in comparison with the conventional non-redundant system. The proposed system is most efficient from the viewpoint of gain in reliability per additional unit as shown in Fig. 6.

V. AN ILLUSTRATIVE EXAMPLE

A. Controller Design

Fig. 7 shows a simplified illustrative example of a material handling robot and a press machine. When independently designed controllers are given, we can design *parity controller* as follows to improve the reliability against a fault of any controllers.

Assume that two controllers for the material handling robot and the press machine are given by Fig. 8 and Fig. 9, respectively. By these controllers, the robot carries an object to the workspace of the press machine, then the machine presses it.

The control logic for the robot is designed as

$$x_1(t+1) = f_{B1}(x_1(t), u_1(t)) \quad (22)$$

$$y_1(t) = h_{B1}(x_1(t), u_1(t)) \quad (23)$$

where $x_1 = [x_{11}, x_{12}, x_{13}, x_{14}, x_{15}, x_{16}, x_{17}]^T$, $u_1 = [u_{10}, u_{11}, u_{12}, u_{13}, u_{14}, u_{15}]^T$, $y_1 = [y_{11}, y_{12}, y_{13}, y_{14}, y_{15}]^T$, and

$$f_{B1}(x_{11}, x_{12}, x_{13}, x_{14}, x_{15}, x_{16}, x_{17}, u_{10}, u_{11}, u_{12}, u_{13}, u_{14}, u_{15}) = \begin{bmatrix} (u_{13} \wedge u_{11} \wedge u_{15} \vee x_{11} \wedge \bar{u}_{12}) \wedge u_{13} \wedge \bar{u}_{15} \\ (u_{13} \wedge u_{12} \wedge \bar{u}_{15} \vee x_{12} \wedge \bar{u}_{11}) \wedge u_{13} \wedge \bar{u}_{15} \\ (u_{10} \wedge u_{11} \wedge u_{13} \wedge \bar{u}_{15} \vee x_{13} \wedge \bar{u}_{14}) \wedge u_{11} \wedge \bar{u}_{15} \\ (u_{14} \wedge u_{11} \wedge u_{15} \vee x_{14} \wedge \bar{u}_{13}) \wedge u_{11} \wedge \bar{u}_{15} \\ (u_{12} \wedge u_{13} \wedge u_{15} \vee x_{15} \wedge \bar{u}_{14}) \wedge u_{12} \wedge \bar{u}_{15} \\ (u_{14} \wedge u_{12} \wedge \bar{u}_{15} \vee x_{16} \wedge \bar{u}_{13}) \wedge u_{12} \wedge \bar{u}_{15} \\ (u_{11} \wedge u_{14} \vee x_{17} \wedge (\bar{u}_{12} \vee \bar{u}_{14})) \wedge \bar{u}_{15} \end{bmatrix} \quad (24)$$

$$h_{B1}(x_{11}, x_{12}, x_{13}, x_{14}, x_{15}, x_{16}, x_{17}, u_{10}, u_{11}, u_{12}, u_{13}, u_{14}, u_{15}) = \begin{bmatrix} x_{11} \wedge u_{13} \wedge \bar{u}_{15} \\ x_{12} \wedge u_{13} \wedge \bar{u}_{15} \\ (x_{13} \vee x_{15}) \wedge \bar{u}_{15} \\ (x_{14} \vee x_{16}) \wedge \bar{u}_{15} \\ x_{17} \wedge \bar{u}_{15} \end{bmatrix} \quad (25)$$

Then, the control logic for the press machine is

$$x_2(t+1) = f_{B2}(x_2(t), u_2(t)) \quad (26)$$

$$y_2(t) = h_{B2}(x_2(t), u_2(t)) \quad (27)$$

where $x_2 = [x_{21}, x_{22}]^T$, $u_2 = [u_{20}, u_{21}, u_{22}, u_{23}]^T$, $y_2 = [y_{21}, y_{22}]^T$, and

$$f_{B2}(x_{21}, x_{22}, u_{20}, u_{21}, u_{22}, u_{23}) = \begin{bmatrix} (u_{20} \wedge u_{21} \vee x_{21} \wedge \bar{u}_{22}) \wedge \bar{u}_{23} \\ (u_{22} \vee x_{22} \wedge \bar{u}_{21}) \wedge \bar{u}_{23} \end{bmatrix} \quad (28)$$

$$h_{B2}(x_{21}, x_{22}, u_{20}, u_{21}, u_{22}, u_{23}) = \begin{bmatrix} x_{21} \wedge \bar{u}_{23} \\ x_{22} \wedge \bar{u}_{23} \end{bmatrix} \quad (29)$$

(24)–(29) can be transformed into Galois field representation by (3).

$$\begin{aligned}
 & f_1(x_{11}, x_{12}, x_{13}, x_{14}, x_{15}, x_{16}, x_{17}, u_{10}, u_{11}, u_{12}, u_{13}, u_{14}, u_{15}) \\
 &= \begin{bmatrix} u_{13}(1+u_{15})((1+u_{12})x_{11} + u_{11}u_{15}(1+x_{11} + u_{12}x_{11})) \\ u_{13}(1+u_{15})((1+u_{11})x_{12} + u_{12}(1+u_{15})(1+x_{12} + u_{11}x_{12})) \\ u_{11}(1+u_{15})((1+u_{14})x_{13} + u_{10}u_{13}(1+u_{15})(1+x_{13} + u_{14}x_{13})) \\ u_{11}(1+u_{15})((1+u_{13})x_{14} + u_{14}u_{15}(1+x_{14} + u_{13}x_{14})) \\ u_{12}(1+u_{15})((1+u_{14})x_{15} + u_{13}u_{15}(1+x_{15} + u_{14}x_{15})) \\ u_{12}(1+u_{15})((1+u_{13})x_{16} + u_{14}(1+u_{15})(1+x_{16} + u_{13}x_{16})) \\ (1+u_{15})(x_{17} + u_{12}u_{14}x_{17} + u_{11}u_{14}(1+x_{17} + u_{12}x_{17})) \end{bmatrix} \quad (30)
 \end{aligned}$$

$$\begin{aligned}
 & h_1(x_{11}, x_{12}, x_{13}, x_{14}, x_{15}, x_{16}, x_{17}, u_{10}, u_{11}, u_{12}, u_{13}, u_{14}, u_{15}) \\
 &= \begin{bmatrix} u_{13}(1+u_{15})x_{11} \\ u_{13}(1+u_{15})x_{12} \\ (1+u_{15})(x_{13} + x_{15} + x_{13}x_{15}) \\ (1+u_{15})(x_{14} + x_{16} + x_{14}x_{16}) \\ (1+u_{15})x_{17} \end{bmatrix} \quad (31)
 \end{aligned}$$

$$\begin{aligned}
 & f_2(x_{21}, x_{22}, u_{20}, u_{21}, u_{22}, u_{23}) \\
 &= \begin{bmatrix} (1+u_{23})((1+u_{22})x_{21} + u_{20}u_{21}(1+x_{21} + u_{22}x_{21})) \\ (1+u_{23})(u_{22} + x_{22} + u_{21}x_{22} + u_{22}x_{22} + u_{21}u_{22}x_{22}) \end{bmatrix} \quad (32)
 \end{aligned}$$

$$\begin{aligned}
 & h_2(x_{21}, x_{22}, u_{20}, u_{21}, u_{22}, u_{23}) \\
 &= \begin{bmatrix} (1+u_{23})x_{21} \\ (1+u_{23})x_{22} \end{bmatrix} \quad (33)
 \end{aligned}$$

Using (7)–(9), the additional controller is designed as follows.

$$\begin{aligned}
 x_3(t+1) &= f_3(x_3(t), u_3(t)) \\
 y_3(t) &= h_3(x_3(t), u_3(t))
 \end{aligned}$$

where $x_3 = [x_{31}, x_{32}, x_{33}, x_{34}, x_{35}, x_{36}, x_{37}, x_{38}, x_{39}]^T$, $u_3 = [u_{10}, u_{11}, u_{12}, u_{13}, u_{14}, u_{15}, u_{20}, u_{21}, u_{22}, u_{23}]^T$, $y_3 = [y_{31}, y_{32}, y_{33}, y_{34}, y_{35}]^T$, and

$$\begin{aligned}
 & f_3(x_{31}, x_{32}, x_{33}, x_{34}, x_{35}, x_{36}, x_{37}, x_{38}, x_{39}, u_{10}, u_{11}, u_{12}, u_{13}, u_{14}, u_{15}, u_{20}, u_{21}, u_{22}, u_{23}) \\
 &= \begin{bmatrix} u_{13}(1+u_{15})((1+u_{12})x_{31} + u_{11}u_{15}(1+x_{31} + u_{12}x_{31})) \\ u_{13}(1+u_{15})((1+u_{11})x_{32} + u_{12}(1+u_{15})(1+x_{32} + u_{11}x_{32})) \\ u_{11}(1+u_{15})((1+u_{14})x_{33} + u_{10}u_{13}(1+u_{15})(1+x_{33} + u_{14}x_{33})) \\ u_{11}(1+u_{15})((1+u_{13})x_{34} + u_{14}u_{15}(1+x_{34} + u_{13}x_{34})) \\ u_{12}(1+u_{15})((1+u_{14})x_{35} + u_{13}u_{15}(1+x_{35} + u_{14}x_{35})) \\ u_{12}(1+u_{15})((1+u_{13})x_{36} + u_{14}(1+u_{15})(1+x_{36} + u_{13}x_{36})) \\ (1+u_{15})(x_{37} + u_{12}u_{14}x_{37} + u_{11}u_{14}(1+x_{37} + u_{12}x_{37})) \\ (1+u_{23})((1+u_{22})x_{38} + u_{20}u_{21}(1+x_{38} + u_{22}x_{38})) \\ (1+u_{23})(u_{22} + x_{39} + u_{21}x_{39} + u_{22}x_{39} + u_{21}u_{22}x_{39}) \end{bmatrix} \quad (34)
 \end{aligned}$$

$$\begin{aligned}
 & h_3(x_{31}, x_{32}, x_{33}, x_{34}, x_{35}, x_{36}, x_{37}, x_{38}, x_{39}, u_{10}, u_{11}, u_{12}, u_{13}, u_{14}, u_{15}, u_{20}, u_{21}, u_{22}, u_{23}) \\
 &= \begin{bmatrix} u_{13}(1+u_{15})x_{31} + (1+u_{23})x_{38} \\ u_{13}(1+u_{15})x_{32} + (1+u_{23})x_{39} \\ (1+u_{15})(x_{33} + x_{35} + x_{33}x_{35}) \\ (1+u_{15})(x_{34} + x_{36} + x_{34}x_{36}) \\ (1+u_{15})x_{37} \end{bmatrix} \quad (35)
 \end{aligned}$$

$$(36)$$

Finally we have control logic by inverse transformation (6)

$$x_3(t+1) = f_{B3}(x_3(t), u_3(t)) \quad (37)$$

$$y_3(t) = h_{B3}(x_3(t), u_3(t)) \quad (38)$$

where

$$f_{B3}(x_{31}, x_{32}, x_{33}, x_{34}, x_{35}, x_{36}, x_{37}, x_{38}, x_{39}, u_{10}, u_{11}, u_{12}, u_{13}, u_{14}, u_{15}, u_{20}, u_{21}, u_{22}, u_{23}) = \begin{bmatrix} (u_{13} \wedge u_{11} \wedge u_{15} \vee x_{31} \wedge \bar{u}_{12}) \wedge u_{13} \wedge \bar{u}_{15} \\ (u_{13} \wedge u_{12} \wedge \bar{u}_{15} \vee x_{32} \wedge \bar{u}_{11}) \wedge u_{13} \wedge \bar{u}_{15} \\ (u_{10} \wedge u_{11} \wedge u_{13} \wedge \bar{u}_{15} \vee x_{33} \wedge \bar{u}_{14}) \wedge u_{11} \wedge \bar{u}_{15} \\ (u_{14} \wedge u_{11} \wedge u_{15} \vee x_{34} \wedge \bar{u}_{13}) \wedge u_{11} \wedge \bar{u}_{15} \\ (u_{12} \wedge u_{13} \wedge u_{15} \vee x_{35} \wedge \bar{u}_{14}) \wedge u_{12} \wedge \bar{u}_{15} \\ (u_{14} \wedge u_{12} \wedge \bar{u}_{15} \vee x_{36} \wedge \bar{u}_{13}) \wedge u_{12} \wedge \bar{u}_{15} \\ (u_{11} \wedge u_{14} \vee x_{37} \wedge (\bar{u}_{12} \vee \bar{u}_{14})) \wedge \bar{u}_{15} \\ (u_{20} \wedge u_{21} \vee x_{38} \wedge \bar{u}_{22}) \wedge \bar{u}_{23} \\ (u_{22} \vee x_{39} \wedge \bar{u}_{21}) \wedge \bar{u}_{23} \end{bmatrix} \quad (39)$$

$$h_{B3}(x_{31}, x_{32}, x_{33}, x_{34}, x_{35}, x_{36}, x_{37}, x_{38}, x_{39}, u_{10}, u_{11}, u_{12}, u_{13}, u_{14}, u_{15}, u_{20}, u_{21}, u_{22}, u_{23}) = \begin{bmatrix} (u_{15} \vee \bar{u}_{13} \vee \bar{x}_{31}) \wedge x_{38} \wedge \bar{u}_{23} \vee (u_{23} \vee \bar{x}_{38}) \wedge u_{13} \wedge x_{31} \wedge \bar{u}_{15} \\ (u_{15} \vee \bar{u}_{13} \vee \bar{x}_{32}) \wedge x_{39} \wedge \bar{u}_{23} \vee (u_{23} \vee \bar{x}_{39}) \wedge u_{13} \wedge x_{32} \wedge \bar{u}_{15} \\ (x_{33} \vee x_{35}) \wedge \bar{u}_{15} \\ (x_{34} \vee x_{36}) \wedge \bar{u}_{15} \\ x_{37} \wedge \bar{u}_{15} \end{bmatrix} \quad (40)$$

B. Experiments

Five personal computers connected by *Fast Ethernet* are used to emulate three PLCs and two plants. A time critical token passing mechanism is introduced to identify a faulty controller and it is implemented over *UDP/IP*.

Fig. 10 shows the experimental results of the proposed system. Fig. 10 (a) and Fig. 10 (b) are outputs of controlled plants. Fig. 10 (c) and Fig. 10 (d) are outputs of controllers (22)–(25) and (26)–(29) which are independently designed for the material handling robot and for the press machine, respectively. Fig. 10 (e) shows an output of the additional controller (37)–(40). This controller behaves as a dynamic parity controller, i. e., the output of the controller always satisfies the parity condition (10) if there is no failure.

In this experiment, the controller #1 failed at $t = 200$ [cycle] then repaired at $t = 327$ [cycle]. As shown in Fig. 10 (a) and (b), the plants continued normal operation even if the controller #1 was down from $t = 200$ [cycle] to $t = 327$ [cycle]. The system was also functional when the controller #2 was down from $t = 400$ [cycle] to $t = 527$ [cycle] and the controller #3 was down from $t = 600$ [cycle] to $t = 727$ [cycle]. Table II shows the states of the controllers and plants.

In the repair process of the controller, the internal state values of other controllers were communicated and the state of the controller was reconstructed.

VI. CONCLUSIONS

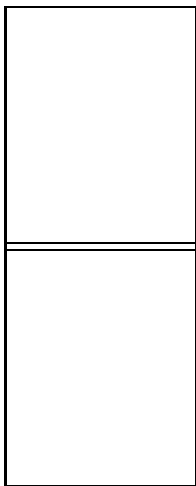
In this paper, an improvement of reliability of PLC based discrete systems is discussed. A fault tolerant configuration for distributed discrete controllers with one redundant controller is proposed. In combination with independently designed controllers, the additional controller is designed so as to satisfy an output parity condition. From the experiments, it is shown that the system is functional even if one of any controllers is down. According to the redundancy, the mean time to failure of the whole system is drastically improved with least additional cost.

VII. ACKNOWLEDGEMENT

The authors wish to thank Prof. M. Tomizuka, University of California at Berkeley, for his useful discussions. The authors would also like to thank Mr. Y. Tonozuka, Mr. Nagaiwa, Toshiba co., Dr. I. Miyazawa, Mr. T. Mizuya, Kanagawa Industrial Research Institute, and Dr. J. T. Ootsuki, Yokohama National University for their helpful suggestions.

REFERENCES

- [1] A. Birolini, *Quality and Reliability of Technical Systems*, Springer-Verlag, 1994.
- [2] V. Blondel, *Simultaneous Stabilization of Linear Systems*, Springer-Verlag, 1994.
- [3] I. E. Shparlinski, *Finite Field: Theory and Computation*, Kluwer Academic Publishers, 1999.
- [4] Y. Fujimoto and T. Sekiguchi, “An Algebraic Approach to Design of Discrete Systems,” in *Proc. IEEE IECON*, pp. 2608–2613, 2000.
- [5] G. A. Gibson, *Redundant Disk Arrays — Reliable, Parallel Secondary Storage*, The MIT Press, 1991.
- [6] R. Isermann, “Preface to the special edition of papers in supervision, fault detection and diagnosis in technical systems,” *Control Engineering Practice*, vol. 5, pp. 671–682, 1997.
- [7] R. Patton, P. Frank, and R. Clark, *Fault Diagnosis in Dynamic Systems*, Prentice Hall, 1989.
- [8] T. Sekiguchi, et al., *Sequential Control Engineering*, IEE of Japan, 1988. (in Japanese)
- [9] S. Suryanarayanan, M. Tomizuka, and T. Suzuki, “Fault Tolerant Lateral Control of Automated Vehicles Based on Simultaneous Stabilization,” in *Proc. IFAC Mechatronics Conference*, 2000.



Yasutaka Fujimoto received B.E., M.E., and Ph.D. degrees in electrical engineering from Yokohama National University, Japan, in 1993, 1995, and 1998, respectively. In 1998, he joined the Department of Electrical Engineering, Keio University. Since 1999, he has been with the Department of Electrical and Computer Engineering, Yokohama National University, where he is currently an Associate Professor. His research interests include discrete systems, motion control, and robotics.

Takashi Sekiguchi received B.E degree in electrical engineering from Yokohama National University, Japan, in 1959, and Ph.D. degree in electrical engineering from Osaka University, Japan, in 1975, respectively. From 1959 to 1961, He was with Yaskawa Electric Corporation, Japan. In 1961 he joined the Department of Electrical and Computer Engineering, Yokohama National University, Japan, where he is currently a professor emeritus. His research interests are in control systems, sequential control and programmable controllers, and Petri net and AI applications in FA and CIM environments. He is a member of IEE, IEIC, and SICE of Japan.

TABLE I
GALOIS FIELD AND BOOLEAN ALGEBRA.

(a) Arithmetic in Boolean algebra

$a \vee b$	
$a \backslash b$	0 1
0	0 1
1	1 1

$a \wedge b$	
$a \backslash b$	0 1
0	0 0
1	0 1

(b) Arithmetic in Galois field

$a + b$	
$a \backslash b$	0 1
0	0 1
1	1 0

$a \times b$	
$a \backslash b$	0 1
0	0 0
1	0 1

$a - b$	
$a \backslash b$	0 1
0	0 1
1	1 0

$a \div b$	
$a \backslash b$	0 1
0	- 0
1	- 1

TABLE II
STATES OF THE CONTROLLERS AND THE PLANTS.

	Time [cycle]						
	0–199	200–326	327–399	400–526	527–599	600–726	727–999
Controller #1	ON	OFF	ON				
Controller #2	ON			OFF	ON		
Controller #3	ON					OFF	ON
M. H. Robot	Functional						
Press Machine	Functional						

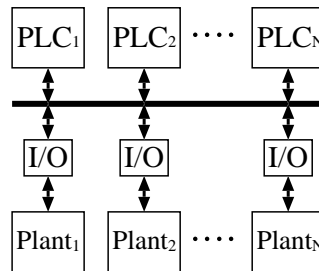


Fig. 1. Configuration of programmable logic controllers.

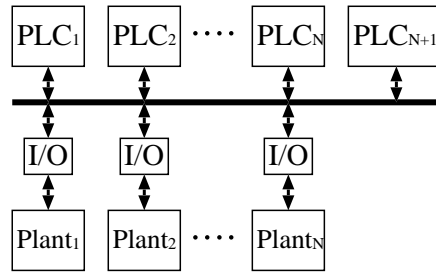


Fig. 2. Fault tolerant configuration of programmable logic controllers.

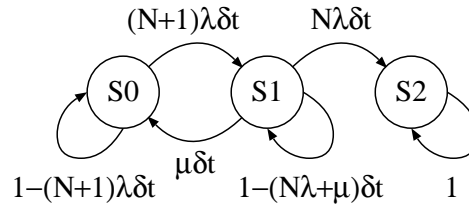


Fig. 3. Diagram of the transition probabilities in $(t, t + \delta t]$.

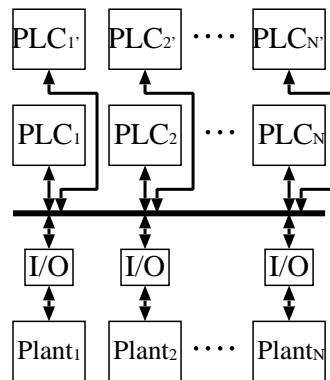


Fig. 4. Configuration of a full duplex system.

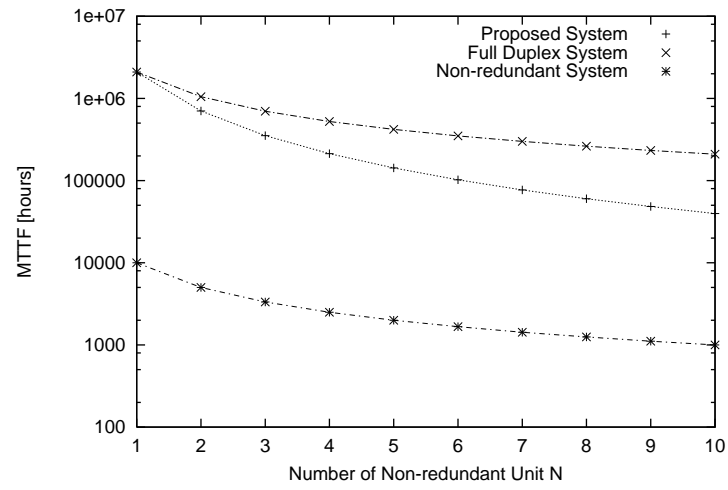


Fig. 5. Number of non-redundant unit versus mean time to failure of whole system.

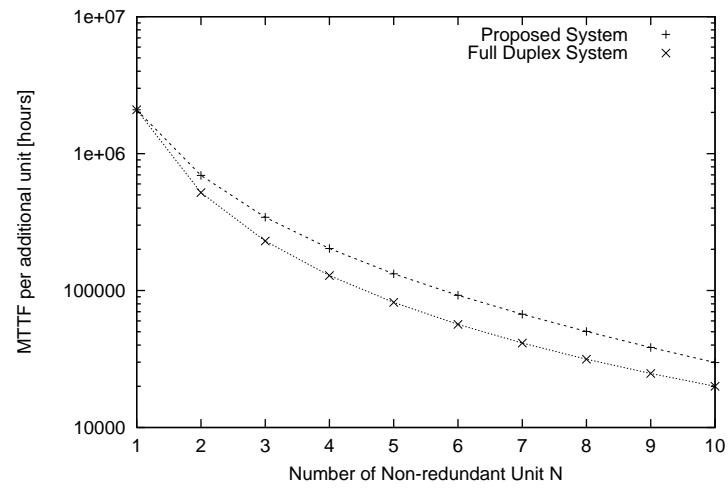


Fig. 6. Mean time to failure per additional unit.

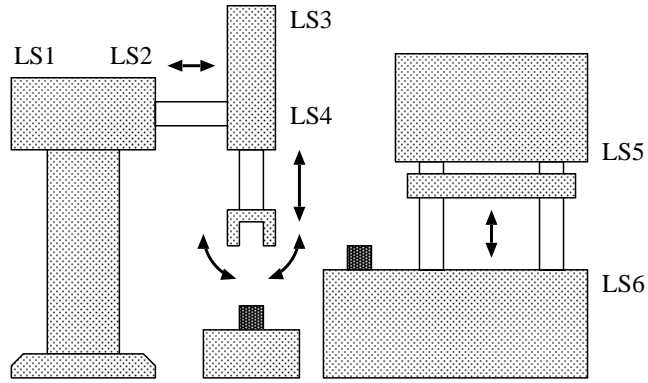


Fig. 7. An illustrative example of a carrying robot and a press machine.

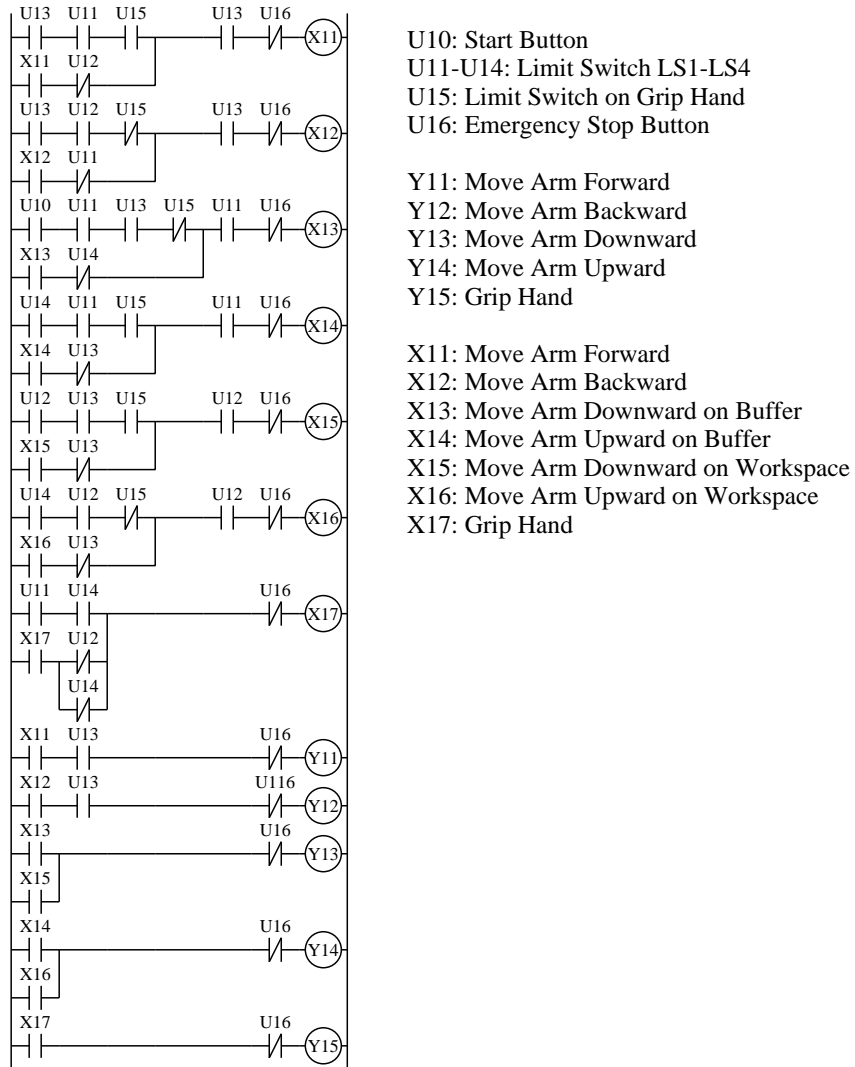


Fig. 8. A ladder diagram of a controller for a carrying robot.

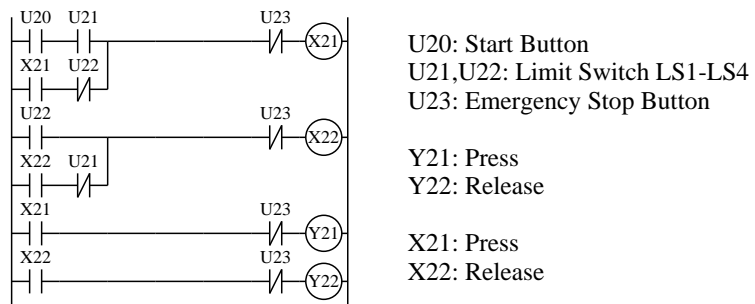
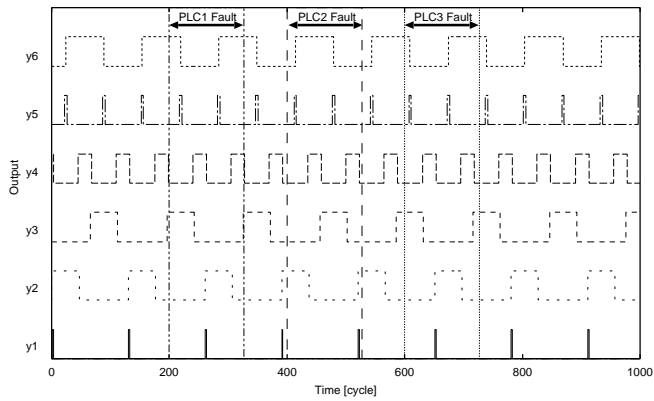
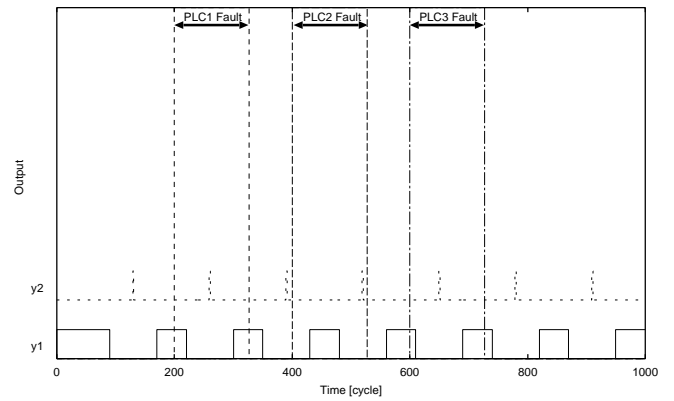


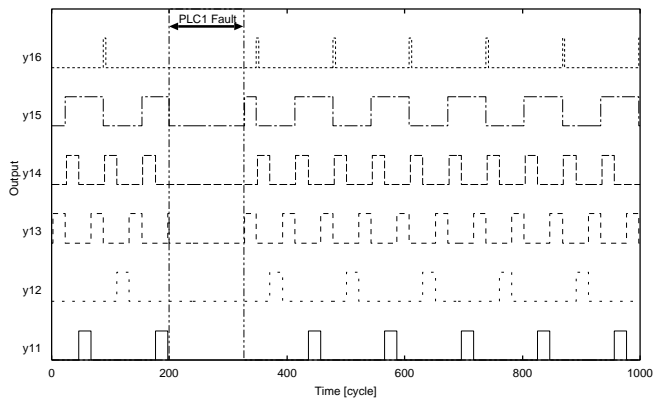
Fig. 9. A ladder diagram of a controller for a press machine.



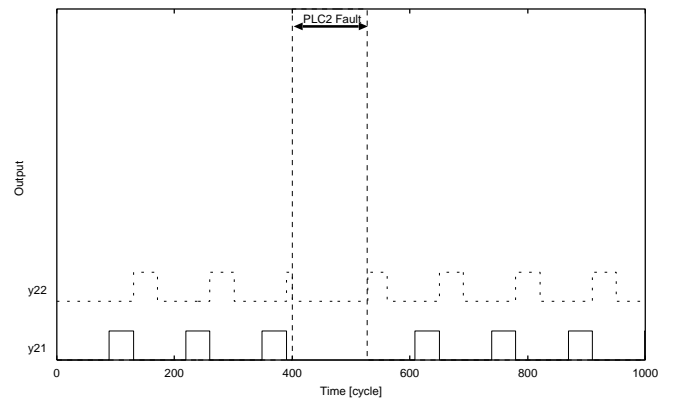
(a) Output of Plant #1 (Robot)



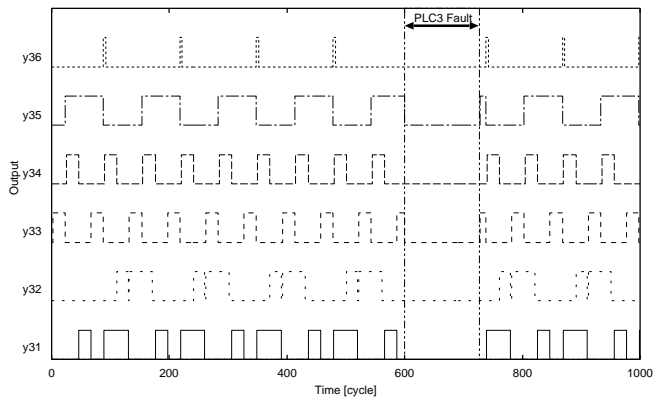
(b) Output of Plant #2 (Press)



(c) Output of Controller #1



(d) Output of Controller #2



(e) Output of Controller #3

Fig. 10. Experimental results.