Informatik-Fachberichte 160

Herausgegeben von W. Brauer
im Auftrag der Gesellschaft für Informatik (GI)

Hubert Mäncher

Fehlertolerante dezentrale Prozeßautomatisierung

Springer-Verlag
Berlin Heidelberg New York
London Paris Tokyo

Autor

Hubert Mäncher
MAGNUM GmbH
Kiesstraße 63, D-6100 Darmstadt

CR Subject Classifications (1987): C.2.4, C.3, D.4.5, D.4.7, J.7

ISBN-13:978-3-540-18754-7 e-ISBN-13:978-3-642-73324-6
DOI: 10.1007/978-3-642-73324-6

CIP-Titelaufnahme der Deutschen Bibliothek.
Mäncher, Hubert:
Fehlertolerante dezentrale Prozessautomatisierung / Hubert Mäncher. - Berlin; Heidelberg; New York; Tokyo: Springer, 1987
(Informatik-Fachberichte; 160)
Zugl.: Darmstadt, Techn. Hochsch., Diss. u. d. T.: Mäncher, Hubert: Ein fehlertolerantes Mikrorechnersystem für die dezentrale Prozessautomatisierung
ISBN-13:978-3-540-18754-7

NE: GT

2145/3140 - 543210

VORWORT

Das vorliegende Buch gibt die Darmstädter Dissertation (D17) "Ein fehlertolerantes Mikrorechnersystem für die dezentrale Prozeßautomatisierung" mit einer erweiterten Einführung und einem verkürzten Anhang wieder. Sie entstand während meiner Tätigkeit als wissenschaftlicher Mitarbeiter bei Herrn Prof. Dr.-Ing. R. Isermann am Institut für Regelungstechnik (Fachgebiet Regelsystemtechnik und Prozeßlenkung) an der Technischen Hochschule Darmstadt.

Mein besonderer Dank gilt daher Herrn Prof. Dr.-Ing. R. Isermann für die Anregung zu dieser Arbeit, für die Gelegenheit zu ihrer Durchführung und für die stete Förderung.

Herrn Prof. Dr.rer.nat. M. Glesner danke ich für das der Arbeit entgegengebrachte Interesse und die Übernahme des Korreferats.

Mein weiterer Dank gilt allen Kollegen für die anregenden Diskussionen und kritischen Anmerkungen zu dieser Arbeit, der Institutswerkstatt und zahlreichen Studenten für ihre Hilfe beim Aufbau des Experimentiersystems.

Schließlich danke ich auch meiner Familie, die mich oft entbehren mußte, für ihre Geduld und Rücksichtnahme.

Die finanziellen Mittel für die Durchführung dieser Arbeit stellte die Deutsche Forschungsgemeinschaft im Rahmen des Schwerpunktprogrammes "Technische Grundsatzfragen beim Einsatz von Mikroprozessoren" (Forschungsvorhaben Is 14/28 und Is 14/30) zur Verfügung. Auch für diese Unterstützung sei gedankt.

Darmstadt, im Oktober 1987 Hubert Mäncher

KURZFASSUNG

Es werden Verfahren zur Implementierung von Fehlertoleranz in dezentralen Automatisierungssystemen untersucht. Ausgehend von einer Analyse der angestrebten Ziele Zuverlässigkeit, Sicherheit und Wirtschaftlichkeit wird eine modifizierte Fehlertoleranz-Klassifizierung angegeben. Die exemplarische Realisierung eines fehlertoleranten Mikrorechnersystems basiert auf der globalen Grundstruktur dezentraler Automatisierungssysteme, in der durch die Zusammenfassung lokaler Subsysteme eine Rekonfiguration zur Tolerierung bestimmter Hardware-Fehler möglich wird. Dezentral realisierte Vergleichs- und Votiereinrichtungen erlauben auch die Tolerierung beliebiger Einfachfehler. Das zugehörige Echtzeitbetriebssystem ist mit der für Anwendungsprogramme transparenten Kommunikation in der Lage, alle Aufträge unabhängig vom Ort des Systemaufrufs und des Zielobjektes auszuführen. Zur Unterstützung des Betriebs von Echtzeitprogrammen werden Funktionszustände zur flexiblen Anpassung an äußere Abläufe behandelt. Anwenderprogramme in Form konfigurierbarer Software-Bausteine zeigen beispielartig die Realisierung fehlertoleranter Funktionen verschiedener Klassen, die gleichzeitig betrieben werden können. Ihre Erprobung an analog simulierten Testprozessen demonstriert die Anwendbarkeit der vorgestellten Konzepte.

SCHLAGWÖRTER

Fehlertoleranz; Fehlertoleranzklassen; Zuverlässigkeit; Sicherheit; Mikrorechner; Fehlertolerantes Mikrorechnersystem; Dezentrales Automatisierungssystem; Fehlertolerantes Automatisierungssystem; Bussystem; Kommunikation; Rekonfiguration; Dezentraler Voter; Echtzeit-Betriebssystem; Mehrrechner-Betriebssystem; Fehlertolerante digitale Regelung.

ABSTRACT

This work investigates methods to implement fault-tolerance in distributed automation systems. Starting with an analysis of the intended goals reliability, security, and economy a modified fault-tolerance classification is shown. The realization of a fault-tolerant microcomputer system as an example uses the basic global structure of distributed automation systems, in which collecting local subsystems enables a system reconfiguration to tolerate particular hardware-faults. Tolerance against unspecified single faults is also achieved by distributed voting and comparing mechanisms. Using its transparent communication functions the according real-time operating system is able to perform all system services independent of the location of the calling task and the destination object. Function-states of real-time programs, supporting their flexible adaption to the outside technical process, are treated. Some user programs in form of configurable software show the realization of fault-tolerant functions in different classes, which may be used simultaneously. Their closed-loop test with analog computers demonstrates the applicability of the presented conceptions.

KEYWORDS

Fault-Tolerance; Fault-Tolerance Classification; Reliability; Security; Microcomputer; Fault-Tolerant Microcomputer System; Distributed Automation System; Fault-Tolerant Automation System; Bussystem; Communication; Reconfiguration; Distributed Voter; Real-Time Operating System; Multicomputer Operating System; Fault-Tolerant Digital Control.

INHALTSVERZEICHNIS

VERZEICHNIS DER WICHTIGSTEN FORMELZEICHEN UND ABKÜRZUNGEN

Allgemeine Sonderzeichen

$\underline{A}$, $\underline{a}$	Matrix A, Vektor a
$\underline{A}^T$, $\underline{a}^T$	transponierte Matrix $\underline{A}$, transponierter Vektor $\underline{a}$
$\hat{a}$	Schätzwert für a
a^*	numerischer Wert von a
Σ	Summe
Π	Produkt

Zuverlässigkeits- und Sicherheitstheorie

F(t)	Ausfallwahrscheinlichkeit
G(t)	Gefährdungswahrscheinlichkeit
P(Ereignis)	Wahrscheinlichkeit von Ereignis
t	variable Zeit
T	feste Zeit
U_S	Sicherheitsunverfügbarkeit
U_Z	Unverfügbarkeit
V_S	Sicherheitsverfügbarkeit
V_Z	Verfügbarkeit

MTBF	mean time between failures	mittlere ausfallfreie Zeit
MTTD	mean time to danger	mittlerer zeitlicher Abstand zw. Gefährdungen
MTTR	mean time to repair	mittlere Reparaturzeit
MTTSR	mean time to safe repair	mittlere Zeit zur Sicherheitswiederherstellung

Weitere Symbole und Abkürzungen der Zuverlässigkeits- und Sicherheitstheorie und eine ausführlichere Erklärung finden sich in Kap. 2.2.1, Tabelle 2.1.

Regelungs- und Automatisierungstechnik

a_i	Nennerparameter der Prozeß-Übertragungsfunktion
A(z)	Nennerpolynom der Prozeß-Übertragungsfunktion
$\underline{A}$	Systemmatrix des Zustandsraummodells
b_i	Zählerparameter der Prozeß-Übertragungsfunktion
B(z)	Zählerpolynom der Prozeß-Übertragungsfunktion
$\underline{B}$, $\underline{b}$	Steuermatrix (-vektor) des Zustandsraummodells
$\underline{C}$, $\underline{c}$	Ausgangsmatrix (-vektor) des Zustandsraummodells
d	diskrete Totzeit
e	Regelabweichung oder Vorhersagefehler
$\underline{I}$	Einheitsmatrix
k	diskrete Zeit $k = t/T_o$
$\underline{K}$, $\underline{k}$	Rückführmatrix (-vektor) des Zustandsreglers
m	Prozeßmodellordnung
p_i, q_i	Parameter eines Ein/Ausgangsreglers
t	kontinuierliche Zeit
T_o	Abtastzeit
U	Stellgröße, absolut
u	Stellgröße, arbeitspunktbezogen
$\underline{U}$, $\underline{u}$	Stellgrößenvektoren
U_o, $\underline{U}_o$	Stellgrößen-Beharrungswerte
W	Sollwert, absolut
w	Sollwert, arbeitspunktbezogen
$\underline{W}$, $\underline{w}$	Sollwertvektoren
W_o, $\underline{W}_o$	Sollwert-Beharrungswerte
X	Zustandsgröße, absolut
x	Zustandsgröße, arbeitspunktbezogen
$\underline{X}$, $\underline{x}$	Zustandsgrößenvektoren
X_o, $\underline{X}_o$	Zustandsgrößen-Beharrungswerte
Y	Regelgröße, absolut
y	Regelgröße, arbeitspunktbezogen
$\underline{Y}$, $\underline{y}$	Regelgrößenvektoren
Y_o, $\underline{Y}_o$	Regelgrößen-Beharrungswerte
z	Variable der z-Transformation

DB	Regler mit Deadbeat-Verhalten
P, PI, PID	Regler mit proportionalem (P), integrierendem (I) und differenzierendem Verhalten
ZR	Zustandsregler

Datentechnik und Informatik

A/D	analog to digital	analog in digital
CPU	central processing unit	Zentraleinheit
D/A	digital to analog	digital in analog
EPROM	erasable programmable read-only memory	löschbarer programmierbarer Festwertspeicher
INT	interrupt	Interrupt, Unterbrechung
I/O	input/output	Ein/Ausgabe
NDP	numeric data processor	Numerikprozessor
NMI	non-maskable interrupt	nicht maskierbarer Int.
PIC	programmable interrupt controller	programmierbarer Int.-Steuerungsbaustein
RAM	random access memory	Schreib/Lese-Speicher
ROM	read-only memory	Festwertspeicher

Beschreibung des fehlertoleranten Mikrorechnersystems FIPS

FIPS	Fault-tolerance Implementation in distr. Processautomation Systems	Fehlertoleranz-Implementierung in dezentr. Prozeßautomatisierungs-Systemen
REX	regional executive	regionale Exekutive
AIO, A-I/O	analog input/output	analoge Ein/Ausgabe
BIO, B-I/O	binary input/output	binäre Ein/Ausgabe
CI	console interface	Konsolen-Interface
G...	global...	global...
GB	global bus	Globalbus
GBC	global bus coupler	Globalbuskopplung

GS	global system	Globalsystem
L...	local...	lokal...
LB	local bus	Lokalbus
LBC	local bus coupler	Lokalbuskopplung
LOP	local operator panel	lokale Bedienkonsole
LS	local system	Lokalsystem
M	memory	Speicher
MIC	microcomputer	Mikrorechner
PS	power supply	Stromversorgung
R...	regional...	regional...
RB	regional bus	Regionalbus
RBC	regional bus coupler	Regionalbuskopplung
RS	regional system	Regionalsystem
S	switch	Schalter
SE	special extension	spezielle Erweiterung