

LITERATURVERZEICHNIS

- Ackermann, J. (1972).
Abtastregelung. Springer Verlag, Berlin, Heidelberg, New York.
- Avizienis, A. (1984).
On current problems in fault-tolerant computing. GI-Fachtagung 'Fehlertolerierende Rechnersysteme', Informatik-Fachberichte Nr. 84, Springer Verlag, Berlin, Heidelberg, New York.
- Bayer, M; Demmelmeier, F.; Endl, H.; Ries, W.; Wiemann, B. (1983).
Implementierung der Betriebssystemfunktionen für das fehlertolerante Multimikrorechnersystem FUTURE. Lehrstuhl für Prozeßrechner, TU München, Interner Bericht 1/83.
- Bergmann, S. (1983).
Digitale parameteradaptive Regelung mit Mikrorechner. Dissertation am Fachbereich 19, TH Darmstadt. Fortschritt-Berichte der VDI-Zeitschriften, Reihe 8, Nr. 55, VDI-Verlag Düsseldorf.
- Birck, H. (1977).
Kompaktregler mit Mikrorechner-Bausteinen. Dissertation an der Fakultät für Elektrotechnik, TU München.
- Bonn, G.; Lorenz, L. (1980).
Steuerung, Synchronisation und Kommunikation bei parallelen Prozessen. FhG-Berichte 2-80.
- Brause, R.; Ammann, E.; Dal Cin, M.; Dilger, E.; Lutz, J.; Risse, R. (1983).
Konzepte des fehlertoleranten Arbeitsplatzrechners ATTEMPTO. GI-Workshop 'Fehlertolerante Mehrprozessor- und Mehrrechnersysteme', Universität Erlangen.
- Brinch Hansen, P. (1977).
Betriebssysteme. Carl Hanser Verlag, München, Wien.
- Dal Cin, M. (1979).
Fehlertolerante Systeme. Teubner-Verlag, Stuttgart.
- Demmelmeier, F.; Ries, W. (1982).
Implementierung von anwendungsspezifischer Fehlertoleranz für Prozeßautomatisierungssysteme. GI-Fachtagung 'Fehlertolerierende Rechnersysteme', Informatik-Fachberichte Nr. 54, Springer Verlag, Berlin, Heidelberg, New York.
- Demmelmeier, F. (1984a).
Betriebssystemfunktionen im fehlertoleranten Multimikrorechnersystem FUTURE. Regelungstechnische Praxis, Heft 9, S. 408-415.
- Echtle, K. (1984).
Fehlermaskierende verteilte Systeme zur Erfüllung hoher Zuverlässigkeits-Anforderungen in Prozeßrechner-Netzen. GI/NTG-

Fachtagung 'Architektur und Betrieb von Rechnersystemen',
Informatik-Fachberichte 78, Springer Verlag, Berlin, Heidelberg,
New York.

Electronics (1983).

Fault-Tolerant Computers, special issue. Electronics, January
1983.

Endl, H. (1982).

*Prozeß-Ein/Ausgabe für ein fehlertolerantes Multimikrorech-
nersystem.* GI-Fachtagung 'Fehlertolerierende Rechnersysteme',
Informatik-Fachberichte Nr. 54, Springer Verlag, Berlin,
Heidelberg, New York.

Färber, G. (1979).

Prozeßrechentechnik. Springer Verlag, Berlin, Heidelberg, New
York.

Färber, G. (1980).

Ein dezentralisierter fairer Bus-Arbeiter. Elektronik, Heft
8, 1980.

Färber, G. (1981).

*Task Specific Implementation of Fault Tolerance in Process
Automation Systems.* In Dal Cin, M. und Dilger, E. (1981).
Self-Diagnosis and Fault-Tolerance. Attempto Verlag, Tübingen.

Färber, G. (1982).

Fehlertolerante Rechnersysteme für die Prozeßautomatisierung.
Regelungstechnische Praxis, Heft 5, 1982, S. 160-168.

Gaede, K. (1977).

Zuverlässigkeit, Mathematische Modelle. Carl Hanser Verlag,
München, Wien.

Goldberg, J. (1984).

The Problem of Confidence in Fault-Tolerant Computer Design.
GI/NTG-Fachtagung Architektur und Betrieb von Rechnersystemen,
Informatik-Fachberichte 78. Springer Verlag, Berlin, Heidelberg,
New York.

Görke, W. (1983).

*Zur Begriffsbildung im Fachgebiet fehlertoleranter Rechen-
systeme.* Mitteilungen der GI/GMR/NTG-Fachgruppe Fehlertole-
rierende Rechensysteme, Nr. 2, November 1983.

Großpietsch, K.-E.; Kaiser, J.; Kröger, R.; Nett, E.; Pedar, A.;
Speicher, M. (1983).

*Recovery-Strategien in objektorientierten verteilten Systeme-
men.* GI-Workshop 'Fehlertolerante Mehrprozessor- und Mehr-
rechnersysteme', Universität Erlangen.

Gunningberg, Per (1983).

*Voting and Redundancy Management implemented by Protocols in
Distributed Systems.* Proc. of 13th Int. Symposium on Fault
Tolerant Computing, IEEE Computer Society, S. 182-185.

- Heger, D.; Steusloff, H.; Syrbe, M. (1979).
Echtzeitrechnersystem mit verteilten Mikroprozessoren.
Forschungsbericht DV 79-01 des BMFT, Karlsruhe.
- Heinzl, J. (1984).
Entwicklungsmethodik für Geräte mit Steuerung durch Mikroelektronik. VDI-Berichte Nr. 55.
- Höfle-Isphording, U. (1978).
Zuverlässigkeitstheorie. Springer Verlag, Berlin, Heidelberg, New York.
- IEEE-spectrum (1981).
Reliability - a special issue. IEEE-spectrum, Vol. 18, No. 10.
- Isermann, R. (1974).
Prozeßidentifikation. Springer Verlag Berlin, Heidelberg, New York.
- Isermann, R. (1977).
Digitale Regelsysteme. Springer Verlag, Berlin, Heidelberg, New York.
- Isermann, R. (1980).
Parameter-adaptive control algorithms - a tutorial.
6th IFAC/IFIP-Conference on Digital Computer Applications, Düsseldorf.
- Klein, A.; Will, B. (1983).
Model of a Basic Fault-Tolerant System BFS. Siemens Forsch.-u. Entwickl.-Ber. Bd. 12, Nr. 1.
- Konakovsky, R. (1977).
Definition und Berechnung der Sicherheit von Automatisierungssystemen. Dissertation an der Universität Stuttgart, Vieweg-Verlag, Braunschweig.
- Kopetz, H.; Lohnert, F.; Merker, W.; Pauthner, G. (1982).
Fehlertoleranz in MARS. GI-Fachtagung 'Fehlertolerierende Rechnersysteme', Informatik-Fachberichte, Nr. 54. Springer Verlag, Berlin, Heidelberg, New York.
- Kronawitter, G.; Neudorfer, E.; Sandler, W.; Freyberger, F. (1981).
Entwicklung und Aufbau einer fehlertolerierenden Reglerstation FTR. PDV-Bericht 212, Gesellschaft für Kernforschung, Karlsruhe.
- Lange, P. (1976).
Die Möglichkeiten einer verbesserten Sicherheitsbeurteilung für elektrotechnische Anlagen. Elektrie 30, Heft 2.
- Laprie, J.-C. (1984).
Dependable Computing and Fault-Tolerance: Concepts and Terminology. Proposal to the IFIP WG 10.4 'Reliable Computing und Fault-Tolerance', Summer 1984 Meeting, Kissimmee, Florida, USA. Mitteilungen der GI/GMR/NTG-Fachgruppe Fehlertolerierende Rechensysteme, Nr. 4, Januar 1985.

- Lauber, R. (1976).
Prozeßautomatisierung I. Springer Verlag, Berlin, Heidelberg, New York.
- Levi, P. (1981).
Betriebssysteme für Realzeitanwendungen. Datakontext-Verlag, Köln.
- Lüers, H. (1983).
Mehrmikrorechner-System für redundante Flugführungsaufgaben. GI-Workshop 'Fehlertolerante Mehrprozessor- und Mehrrechner-systeme', Universität Erlangen.
- Maehle, E. (1981).
Self-Test Programs and their Application to Fault-Tolerant Multiprocessor Systems. In Dal Cin, M. und Dilger, E. (1981), *Self-Diagnosis and Fault-Tolerance*, Attempto Verlag, Tübingen.
- Maehle, E.; Joseph, H. (1982).
Selbstdiagnose in fehlertoleranten DIRMU-Multi-Mikroprozessorkonfigurationen. GI-Fachtagung 'Fehlertolerierende Rechensysteme', Informatik-Fachberichte, Nr. 54. Springer Verlag, Berlin, Heidelberg, New York.
- Mäncher, H. (1984a).
Synchronization tools and a restart method in the fault-tolerant distributed automation system FIPS. 2. GI/NTG/GMR-Fachtagung 'Fehlertolerierende Rechensysteme', Informatik-Fachberichte Nr. 84, Springer Verlag, Berlin, Heidelberg, New York.
- Mäncher, H.; Waldschmidt, U. (1985).
Anwenderhandbuch: Fehlertolerantes Mikrorechnersystem FIPS mit Echtzeitbetriebssystem REX. Institut für Regelungstechnik, TH Darmstadt.
- Meyna, Arno (1982).
Einführung in die Sicherheitstheorie: sicherheitstechnische Analyseverfahren. Carl Hanser Verlag, München, Wien.
- Nilson, S.A. (1978a).
Selbsttestverfahren für Mikrorechner. VDI-Berichte Nr. 328.
- Nix, H.-G. (1985).
Einsatz von Mikrorechnern für Sicherheitsaufgaben. VDI/VDE-GMR-Fachtagung Mikroelektronik in der Automatisierungstechnik, Baden-Baden, VDI-Berichte 550, VDI-Verlag, Düsseldorf.
- Oppelt, W. (1972).
Kleines Handbuch technischer Regelvorgänge, 5. Auflage. Verlag Chemie, Weinheim.
- Randell, B. (1978).
Reliability issues in computing system design. ACM Computing surveys, Vol. 10.

- Rau, J. (1970).
Optimization and probability in systems engineering. Reinhold van Nostrand, New York.
- Reinschke, K. (1973).
Zuverlässigkeit von Systemen, Band 1. VEB Verlag Technik, Berlin.
- Ripps, D. L. (1980).
On Operating Systems. IPI Industrial Programming Inc., Jericho, New York.
- Risse, T.; Brause, R.; Dal Cin, M; Dilger, E.; Lutz, J. (1984).
Entwurf und Struktur einer Betriebssystemschicht zur Implementierung von Fehlertoleranz. GI-Fachtagung 'Fehlertolerierende Rechnersysteme', Informatik-Fachberichte Nr. 84, Springer Verlag, Berlin, Heidelberg, New York.
- Schmidt, G.; Sandler, W. (1980a).
Redundanzkonzepte in modernen Prozeßautomatisierungssystemen. Regelungstechnische Praxis, 23, Heft 9.
- Schneeweiss, W. (1980a).
Zuverlässigkeits-Systemtheorie. Datakontext-Verlag, Köln.
- Schneeweiss, W. (1980b).
Minimale Pfade und minimale Schnitte bei Zuverlässigkeitsuntersuchungen. Regelungstechnik, Heft 9.
- Schneider, W. (1974).
Berechnung und Sicherheit von parallelredundanten Schaltwerken. Dissertation an der TU Braunschweig.
- Schumann, R. (1982).
Digitale parameteradaptive Mehrgrößenregelung. -Ein Beitrag zu Entwurf und Analyse -. Dissertation am Fachbereich 19, TH Darmstadt.
- Seifert, M. (1984).
Commercial Available Fault-Tolerant Systems. Mitteilungen der GI/GMR/NTG-Fachgruppe Fehlertolerierende Rechensysteme, Nr. 3, Juni 1984.
- Sandler, W. (1982).
Eine fehlertolerierende Reglerstation mit busorientierter Multi-Mikrorechner-Struktur und einem ökonomischen sowie effizienten Redundanzkonzept. Dissertation an der Fakultät für Elektrotechnik, TU München.
- Störmer, H. (1970).
Mathematische Theorie der Zuverlässigkeit. Oldenbourg Verlag, München.
- Strejc, V. (1979).
Least squares parameter estimation. Tutorial on systems identification. IFAC-Symposium on identification and system parameter estimation, Darmstadt.

Stübler, H. J. (1978).

Zuverlässigkeitserfahrungen mit Prozeßrechnern. Regelungstechnische Praxis, Heft 6.

Syrbe, M. (1981).

The description of fault-tolerant systems - a necessity of the practice. In Dal Cin, M. and Dilger, E. Self-Diagnosis and Fault-Tolerance. Attempto Verlag, Tübingen.

Trauboth, H. (1984).

Zuverlässigkeit von DV-Systemen - eine systemtechnische Aufgabe. GI/NTG-Fachtagung 'Architektur und Betrieb von Rechen-systemen', Informatik-Fachberichte 78. Springer Verlag, Berlin, Heidelberg, New York.

Weiss, R. (1983).

Fehlertolerante Rechnersysteme - Funktionsprinzipien und Realisierungsformen. Regelungstechnische Praxis, Heft 10, S. 408-416.

Wettstein, H. (1978).

Aufbau und Struktur von Betriebssystemen. Carl Hanser Verlag, München, Wien.

Wettstein, H. (1984).

Architektur von Betriebssystemen. Carl Hanser Verlag, München, Wien.

Wensley, John H. (1983).

An Operating System for a TMR Fault-Tolerant System. Proc. of 13th Int. Symposium on Fault Tolerant Computing, IEEE Computer Society, S. 452-455.

Literatur von Firmen, Behörden und Organisationen

AMD (1981).

The Am9513, System Timing Controller. Advanced Micro Devices, Inc., Sunnyvale, Kalifornien.

BBC (PROCONTROL I).

Technische Beschreibungen und Firmenunterlagen. BROWN, BOVERI & CIE. AG, Mannheim.

Beckman (1980).

Microprocessor Compatible CMOS 12-Bit D-to-A Converter. Beckman Instruments, Inc., Fullerton, Kalifornien.

Elba (1981).

ESS 345112/H, sekundär getaktete Gleichspannungsnetzteile, Firmenunterlagen. ELBA-electronic GmbH, Altlußheim.

Harris (1979).

HI-5900, Analog Data Acquisition Signal Processor, Harris Corporation, Melbourne, Florida.

- Harris (1980).
HI-5712, High Performance 12 Bit Analog to Digital Converter.
Harris Corporation, Melbourne, Florida.
- Hartmann & Braun (Contronic P).
Technische Beschreibungen und Firmenunterlagen. Hartmann &
Braun AG, Meß- und Regeltechnik, Frankfurt.
- Hitachi (1981).
Alpha-numerische LCD-Anzeigen-Module, Firmenunterlagen.
Vertrieb Data-Modul, München.
- Honeywell (TDC 2000).
Technische Beschreibungen und Firmenunterlagen. Honeywell
GmbH, Offenbach.
- Intel (1977).
MCS-48 Microcomputer User's Manual. Intel Corporation, Santa
Clara, Kalifornien.
- Intel (1979a).
Intel Multibus Specification. Intel Corporation, Santa Clara,
Kalifornien.
- Intel (1979b).
The 8086 Family User's Manual. Intel Corporation, Santa
Clara, Kalifornien.
- Intel (1980a).
*The 8086 Family User's Manual: Supplement for the 8087
Numeric Data Processor.* Intel Corporation, Santa Clara,
Kalifornien.
- Intel (1980b).
Component Data Catalog. Intel Corporation, Santa Clara,
Kalifornien.
- Intel (1980c).
PL/M-86 User's Guide. Intel Corporation, Santa Clara,
Kalifornien.
- Intel (1980d).
iAPX 86, 88 Family Utilities User's Guide. Intel Corporation,
Santa Clara, Kalifornien.
- Intel (1980e).
iRMX 86 Operating System Description. Intel Corporation,
Santa Clara, Kalifornien.
- Intel (1981).
8087 Support Library Reference Manual. Intel Corporation,
Santa Clara, Kalifornien.
- National Semiconductor (1980).
MM57499 96 or 144-Keybaord Interface (SKI). National
Semiconductor Corp., Santa Clara, Kalifornien.

Siemens (1981a).

AMS-Bus-System, Technische Beschreibung. Siemens AG, München.

Siemens (1981b).

AMS-D7-A5, Technische Beschreibung. Siemens AG, München.

Siemens (1981c).

AMS-D219-A1, Technische Beschreibung. Siemens AG, München.

Siemens (1983).

Realzeit-Multitasking-Betriebssystem SMP-RMOS1, Benutzerhandbuch. Siemens AG, München.

Siemens (Teleperm M).

Technische Beschreibungen und Firmenunterlagen. Siemens AG, München.

TÜV-Rheinland (1982).

Zuverlässige und sichere Rechensysteme, ein Aufgabengebiet des TÜV-Rheinland. TÜV-Informationen 7/82, TÜV Rheinland e.V., Köln.

U.S. Department of Defense (1980).

Military Standardization Handbook, Reliability Prediction of Electronic Equipment, (MIL-HDBK-217C).

VDI/VDE-Richtlinie 2180 (1984).

Sicherung von Anlagen der Verfahrenstechnik mit Mitteln der Meß-, Steuerungs- und Regelungstechnik (Gründruck). VDI-Verlag, Düsseldorf.

VDI/VDE-Richtlinie 3541 (1984).

Steuerungseinrichtungen mit vereinbarter gesicherter Funktion. VDI-Verlag, Düsseldorf.

VDI/VDE-Richtlinie 3542 (1986, Entwurf).

Sicherheitstechnische Begriffe für Automatisierungssysteme. VDI-Verlag, Düsseldorf.

VDI/VDE-Richtlinie 3691 (1984).

Erfassung von Zuverlässigkeitswerten bei Prozeßrechnereinsätzen. VDI-Verlag, Düsseldorf.

Zilog (1980).

Z8030 Serial Communications Controller. Zilog Corp., Cupertino, Kalifornien.

Studien- und Diplomarbeiten

Bieniek, D. (1984).

Selbsttestprogramme für ein fehlertolerantes dezentrales Automatisierungssystem. Studienarbeit I/1061 am Institut für Regelungstechnik, TH Darmstadt.

Eisel, K. (1983).

Programme zur Erfüllung hoher Sicherheitsanforderungen durch 2-von-2-Vergleich in einem dezentralen Automatisierungssystem. Diplomarbeit I/995 am Institut für Regelungstechnik, TH Darmstadt.

Grawunder, N. (1984).

Erstellung eines Laders und Rekonfigurators für ein fehler-tolerantes dezentrales Prozeßautomatisierungssystem. Diplomarbeit I/1056 am Institut für Regelungstechnik, TH Darmstadt.

Güre, B. (1982).

Entwicklung einer Monitorkarte mit lokaler Bedienkonsole. Studienarbeit I/986 am Institut für Regelungstechnik, TH Darmstadt.

Heß, W. (1982).

Buskopplung für hohe Zuverlässigkeitsanforderungen. Studienarbeit I/999 am Institut für Regelungstechnik, TH Darmstadt.

Heß, W. (1983).

Realisierung eines Regelprogramms mit der Möglichkeit zu 2-von-3-Entscheidungen in einem dezentralen Automatisierungssystem. Diplomarbeit I/1011 am Institut für Regelungstechnik, TH Darmstadt.

Hiller, B. (1984).

Implementierung regelungstechnischer Anwenderprogramme auf einem fehlertoleranten dezentralen Mikrorechnersystem. Diplomarbeit I/1055 am Institut für Regelungstechnik, TH Darmstadt.

Johnson, A., Dzombic, R. (1984).

Erstellung und Integration eines Bedienprogramms für die dezentrale Prozeßlenkung; Programmierung einer intelligenten Bedienerchnittstelle für die dezentrale Prozeßlenkung. Studienarbeit I/1062 am Institut für Regelungstechnik, TH Darmstadt.

Johnson, A. (1984).

Zustandsregler verschiedener Zuverlässigkeitsklassen für ein fehlertolerantes Mikrorechnersystem. Diplomarbeit I/1082 am Institut für Regelungstechnik, TH Darmstadt.

Leidel, E. (1982).

Programmbibliothek für einen 16-Bit-Mikrorechner. Studienarbeit I/959 am Institut für Regelungstechnik, TH Darmstadt.

Leidel, E. (1983).

Entwicklung, Programmierung und Implementierung eines parallelen adaptiven Mehrgrößenreglers für ein 16-Bit-Mehrprozessorsystem. Diplomarbeit I/996 am Institut für Regelungstechnik, TH Darmstadt.

Mäncher, H. (1980)

Vergleich verschiedener Rekursionsalgorithmen für die Methode der kleinsten Quadrate. Diplomarbeit I/820 am Institut für Regelungstechnik, TH Darmstadt.

Peter, K. (1984).

Erstellung des Exekutivteils eines Echtzeitbetriebssystems mit Multitasking und Multiprocessing. Diplomarbeit I/1033 am Institut für Regelungstechnik, TH Darmstadt.

Speth, H. (1982).

Entwicklung einer Analog-Ein/Ausgabe-Baugruppe für ein Mehrrechner-System. Studienarbeit I/956 am Institut für Regelungstechnik, TH Darmstadt.

Waldschmidt, U. (1985a).

Realisierung und Vergleich von verschiedenen Kombinationen aus digitalen Reglern und Fehlertoleranzverfahren. Studienarbeit I/1079 am Institut für Regelungstechnik, TH Darmstadt.

Weiß, W. (1982a).

Zuverlässigkeitsmodelle für verschiedene Multi-Mikrorechner-Strukturen. Studienarbeit I/958 am Institut für Regelungstechnik, TH Darmstadt.

Weiß, W. (1982b).

Implementierung und Erweiterung eines Selbsttestprogramms für Mikrorechner. Diplomarbeit I/990 am Institut für Regelungstechnik, TH Darmstadt.

A N H A N G

- A.1 Verfügbarkeitsberechnung für die Fehler-
toleranzklasse W

A.1 Verfügbarkeitsberechnung für die Fehlertoleranzklasse W

Für die Ermittlung der Verfügbarkeit einer vermaschten Systemstruktur wird von Schneeweiss (1980a) ein Arbeitsschema angegeben:

1. Aufstellen des Zuverlässigkeitsnetzes.

Das zur Reglertask der Fehlertoleranzklasse W aus Kap. 6.1.1.1 gehörende Zuverlässigkeitsnetz ist in Bild A.1.1 wiedergegeben.

2. Aufstellen der booleschen Gleichung für die Funktionsfähigkeit der Reglertask anhand des Zuverlässigkeitsnetzes.

X_i ist eine boolesche Variable mit der Aussage:

- $X_i = 1$ - betrachtete Einheit ist funktionsfähig,
- $X_i = 0$ - betrachtete Einheit ist ausgefallen.

Die seriell angeordneten Moduln werden UND-verknüpft ('·'), die parallelen ODER-verknüpft ('+').

$$X = (X_{CI1} + X_{RBL50}) \cdot (X_{MIC1} \cdot X_{M1} + X_{RBL50}) \cdot X_{AIO1} \cdot X_{PS1} \cdot X_{LB1}$$

Die darin enthaltene Abkürzung

$$X_{RBL50} = X_{CIO} \cdot X_{MICO} \cdot X_{M0} \cdot X_{PS0} \cdot X_{LB0} \cdot X_{RBC0} \cdot X_{RB} \cdot X_{RBC1}$$

beschreibt dabei den "Umweg" über das zweite Lokalsystem. Der ist hier für die Umgehung von CI1 und die Umgehung von MIC1-M1 gleich.

3. Reduzierung der booleschen Gleichung bis keine Potenzen einer Variablen X_i mehr enthalten sind.

Durch verschiedene Umformungen erhält man schließlich die Form

$$X = (X_{CI1} \cdot X_{MIC1} \cdot X_{M1} + X_{RBLSO}) \cdot X_{AIO1} \cdot X_{PS1} \cdot X_{LB1} \cdot$$

4. Ersetzen der booleschen Variablen X_i durch die Verfügbarkeiten V_i .

Dieser Austausch ist aufgrund der Beziehung

$$V_i = P (X_i = 1)$$

möglich, die besagt, daß die Wahrscheinlichkeit, eine Einheit im intakten Zustand anzutreffen, gleich der Verfügbarkeit ist. Für den Austausch verknüpfter Variablen gelten verschiedene Umformregeln, die z. B. aus einer Tabelle in Dal Cin (1979) zu ersehen sind.

Es folgt die Gleichung zur numerischen Bestimmung der Verfügbarkeit

$$V_Z = (V_{CI1} V_{MIC1} V_{M1} + V_{RBLSO} - V_{CI1} V_{MIC1} V_{M1} V_{RBLSO}) V_{AIO1} V_{PS1} V_{LB1}$$

mit der Abkürzung

$$V_{RBLSO} = V_{CIO} V_{MICO} V_{MO} V_{PSO} V_{LBO} V_{RBCO} V_{RB} V_{RBC1}$$

und den Verfügbarkeiten der einzelnen Moduln als Variablen. Werden gleiche Verfügbarkeiten für gleiche Moduln in beiden Lokalsystemen angenommen, kann bei der weiteren Rechnung die Indizierung entfallen.

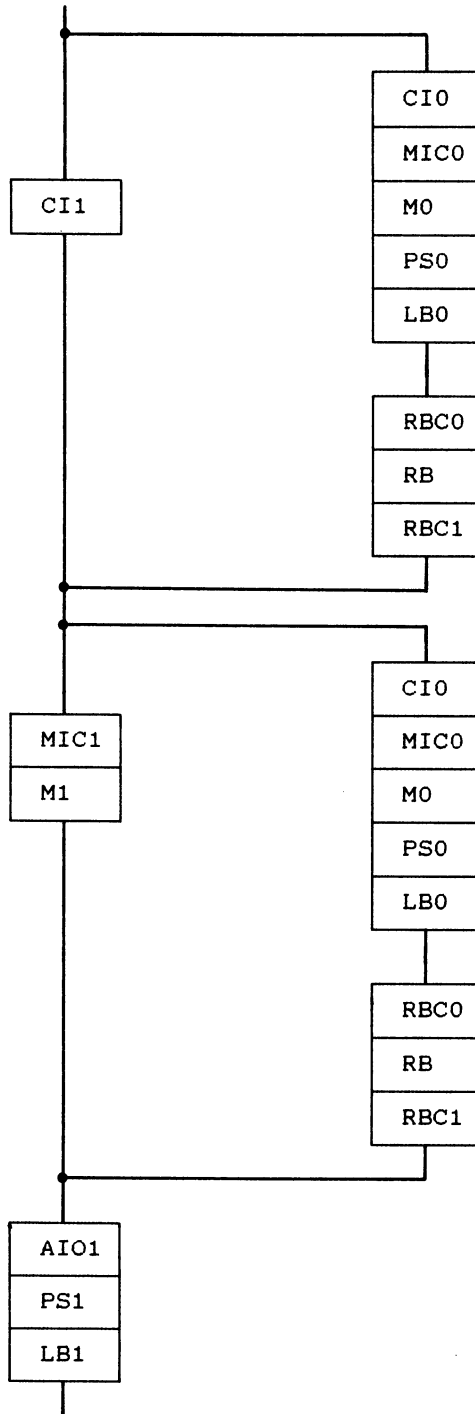


Bild A.1.1: Zuverlässigkeitsnetz der Regelprogramme in der Fehlertoleranzklasse W.