Transparent fault-tolerance for process control systems

Author(s):
Siegrist, Thomas Theodor

Publication Date:
1995

Permanent Link:

https://doi.org/10.3929/ethz-a-001441115 →

Diss. ETH No. 11043

Transparent Fault-Tolerance for Process Control Systems

A dissertation for the degree of
Doctor of Technical Sciences

submitted to the

SWISS FEDERAL INSTITUTE OF TECHNOLOGY
ZÜRICH

presented by
THOMAS THEODOR SIEGRIST
Dipl. El.-Ing. ETH
born July 17, 1961
citizen of Meisterschwanden AG

accepted on the recommendation of
Prof. Dr. A. Kündig, examiner
Prof. Dr. E. Maehle, co-examiner

Zürich 1995

Abstract

When computer systems are used to control processes, the reliability of such *process control systems* is a critical factor. Reliability can be increased by adding redundancy and thus making the process control system able to tolerate faults. This thesis presents a fault-tolerance concept suitable for industrial processes which require fast continuous and/or fast discrete control. The concept either guarantees integrity (for safety-related processes) or persistency (for processes demanding high availability) provided that only one hardware fault occurs at a time.

Both cases are covered by a single duplex architecture consisting of two redundant multiprocessors. The two multiprocessors run in parallel and perform the same operations; they remain synchronized by sporadically exchanging information via a special *Update and Synchronization Unit*. Thus, the process outputs of the two multiprocessors stay very close in both the time and the value domain. They can easily be compared, and a switch-over from one output to the other can happen smoothly. After a temporary shut-down of one multiprocessor, it is reintegrated while the other multiprocessor maintains full control of the process.

The operating systems of the two multiprocessors hide most of the fault-tolerance mechanisms from the application software. Only small modifications are necessary when the application software is transferred from a non-redundant system to the fault-tolerant system. The algorithms required to achieve this are described precisely, using the formal language 'CSP'. This description also allows to prove the correctness of the algorithms.

The presented concept has been implemented on a commercial process control system to demonstrate its feasibility. A detailed description of the implementation, a list of the prerequisites derived from the formal model and the implementation, and some hints on how to carry out a development make it possible to transfer the concept to other process control systems.

Zusammenfassung

Werden Rechner zur Steuerung von Prozessen eingesetzt, so ist die Zuverlässigkeit solcher *Prozesssteuerungen* ein wichtiger Faktor. Die Zuverlässigkeit lässt sich durch Hinzufügen von Redundanz verbessern. Mit Hilfe der Redundanz kann die Prozesssteuerung eigene Fehler tolerieren, d.h. deren Auswirkungen in definierten Grenzen halten oder gar verhindern. Die vorliegende Dissertation entwickelt ein Fehlertoleranzkonzept, das für industrielle Prozesse geeignet ist, die eine schnelle kontinuierliche und/oder eine schnelle diskrete Steuerung erfordern. Das Konzept garantiert entweder Integrität (für sicherheitsrelevante Prozesse) oder ein ununterbrochenes Funktionieren der Steuerung (für Prozesse, die eine hohe Verfügbarkeit verlangen) vorausgesetzt, dass nur ein Hardware-Fehler auf einmal auftritt.

Beide Fälle können durch eine einzige Doppelrechnerarchitektur, die aus zwei redundanten Multiprozessoren besteht, abgedeckt werden. Die beiden Multiprozessoren führen parallel die gleichen Operationen aus. Sie laufen synchron zueinander, weil sie sporadisch über eine spezielle Verbindungseinheit, *"Update and Sychronization Unit"* genannt, Informationen austauschen. Die Ausgangssignale an den Prozess bleiben so zeitlich und im Bezug auf ihre Werte sehr nahe beieinander. Sie können einfach verglichen werden, und eine Umschaltung von einem Ausgang zum anderen kann reibungslos erfolgen. Nach der vorübergehenden Ausserbetriebsetzung eines Multiprozessors wird dieser wieder in den Betrieb eingegliedert, während der andere Multiprozessor gleichzeitig den Prozess ohne Einschränkung weiter steuert.

Die Betriebssysteme in den beiden Multiprozessoren erfüllen fast alle Fehlertoleranz-Aufgaben, unsichtbar für die Anwender-Software. Nur geringfügige Anpassungen sind nötig, um eine Anwender-Software von einer nicht redundanten Steuerung auf die fehlertolerante Steuerung zu portieren. Die dafür erforderlichen Algorithmen werden exakt beschrieben, unter Verwendung der formalen Sprache "CSP". Diese Beschreibung erlaubt es auch, die Korrektheit der Algorithmen nachzuweisen.

Das vorgestellte Konzept wurde auf einem kommerziellen Prozesssteuerungssystem implementiert, um seine Brauchbarkeit zu zeigen. Eine detaillierte Beschreibung dieser Implementation, eine Liste der aus dem formalen Modell und der Implementation abgeleiteten Voraussetzungen und einige Hinweise, wie eine Entwicklung ablaufen müsste, machen es möglich, das Konzept auf ein anderes Prozesssteuerungssystem zu übertragen.