

(12) NACH DEM VERTRAG ÜBER DIE INTERNATIONALE ZUSAMMENARBEIT AUF DEM GEBIET DES
PATENTWESENS (PCT) VERÖFFENTLICHTE INTERNATIONALE ANMELDUNG

(19) Weltorganisation für geistiges Eigentum
Internationales Büro



(43) Internationales Veröffentlichungsdatum
4. Mai 2006 (04.05.2006)

PCT

(10) Internationale Veröffentlichungsnummer
WO 2006/045800 A1

(51) Internationale Patentklassifikation:
G06F 11/16 (2006.01)

(21) Internationales Aktenzeichen: PCT/EP2005/055537

(22) Internationales Anmeldedatum:
25. Oktober 2005 (25.10.2005)

(25) Einreichungssprache: Deutsch

(26) Veröffentlichungssprache: Deutsch

(30) Angaben zur Priorität:
10 2004 051 952.8
25. Oktober 2004 (25.10.2004) DE
10 2004 051 964.1
25. Oktober 2004 (25.10.2004) DE
10 2004 051 950.1
25. Oktober 2004 (25.10.2004) DE
10 2004 051 937.4
25. Oktober 2004 (25.10.2004) DE
10 2004 051 992.7
25. Oktober 2004 (25.10.2004) DE

(71) Anmelder (für alle Bestimmungsstaaten mit Ausnahme von
US): **ROBERT BOSCH GMBH** [DE/DE]; Postfach 30 02
20, 70442 Stuttgart (DE).

(72) Erfinder; und

(75) Erfinder/Anmelder (nur für US): **KOTTKE, Thomas**
[DE/DE]; Leimentalstrasse 13/1, 71139 Ehningen (DE).

(74) Gemeinsamer Vertreter: **ROBERT BOSCH GMBH**;
Postfach 30 02 20, 70442 Stuttgart (DE).

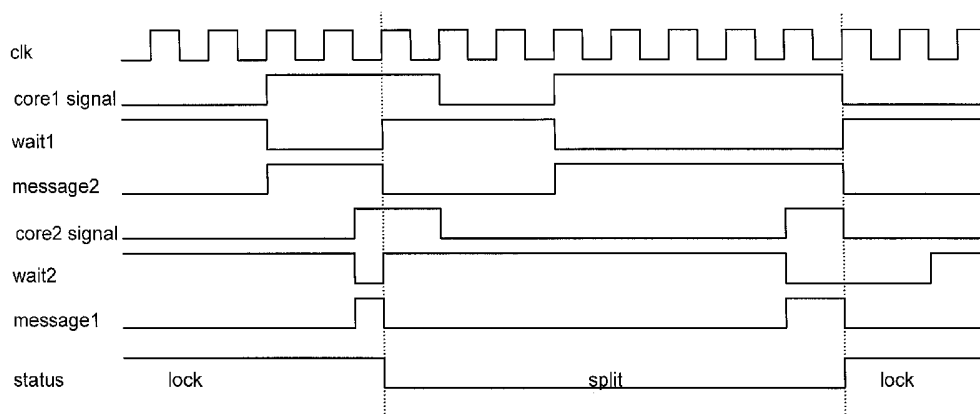
(81) Bestimmungsstaaten (soweit nicht anders angegeben, für
jede verfügbare nationale Schutzrechtsart): AE, AG, AL,
AM, AT, AU, AZ, BA, BB, BG, BR, BW, BY, BZ, CA, CH,
CN, CO, CR, CU, CZ, DK, DM, DZ, EC, EE, EG, ES, FI,
GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE,
KG, KM, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, LY,
MA, MD, MG, MK, MN, MW, MX, MZ, NA, NG, NI, NO,
NZ, OM, PG, PH, PL, PT, RO, RU, SC, SD, SE, SG, SK,
SL, SM, SY, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ,
VC, VN, YU, ZA, ZM, ZW.

(84) Bestimmungsstaaten (soweit nicht anders angegeben, für
jede verfügbare regionale Schutzrechtsart): ARIPO (BW,
GH, GM, KE, LS, MW, MZ, NA, SD, SL, SZ, TZ, UG,
ZM, ZW), eurasisches (AM, AZ, BY, KG, KZ, MD, RU,
TJ, TM), europäisches (AT, BE, BG, CH, CY, CZ, DE, DK,
EE, ES, FI, FR, GB, GR, HU, IE, IS, IT, LT, LU, LV, MC,
NL, PL, PT, RO, SE, SI, SK, TR), OAPI (BF, BJ, CF, CG,
CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

[Fortsetzung auf der nächsten Seite]

(54) Title: METHOD AND DEVICE FOR SYNCHRONISING IN A MULTI-PROCESSOR SYSTEM

(54) Bezeichnung: VERFAHREN UND VORRICHTUNG ZUR SYNCHRONISIERUNG IN EINEM MEHRPROZESSORSYS-
TEM



(57) Abstract: The invention relates to a method and to a device for synchronising in a multi-processor system comprising at least two processors and switching means which can be switched between at least two operational modes. The inventive device is embodied in such a manner that synchronisation is carried out by a stop signal which stops an advancing processor in order to synchronise the stop signal with the at least two processors.

(57) Zusammenfassung: Verfahren und Vorrichtung zur Synchronisierung in einem Mehrprozessorsystem mit wenigstens zwei Prozessoren, wobei Umschaltmittel enthalten sind durch welche zwischen wenigstens zwei Betriebsmodi umgeschaltet werden kann, wobei die Vorrichtung derart ausgestaltet sind, dass eine Synchronisierung durch ein Anhaltesignal durchgeführt wird, welches einen vorausseilenden Prozessor anhält um diesen mit dem wenigstens zweiten Prozessor zu synchronisieren.



WO 2006/045800 A1



Veröffentlicht:

— mit internationalem Recherchenbericht

Zur Erklärung der Zweibuchstaben-Codes und der anderen Abkürzungen wird auf die Erklärungen ("Guidance Notes on Codes and Abbreviations") am Anfang jeder regulären Ausgabe der PCT-Gazette verwiesen.

5

10 Verfahren und Vorrichtung zur Synchronisierung in einem Mehrprozessorsystem

Stand der Technik

15 In technischen Anwendungen, wie insbesondere im Kraftfahrzeug oder im
Industriegüterbereich also z.B. Maschinenbereich und in der Automatisierung werden
ständig mehr und mehr mikroprozessor- oder rechnerbasierte Steuerungs- und
Regelungssysteme für sicherheitskritische Anwendungen eingesetzt. Dabei sind
20 Zweirechnersysteme oder Zweiprozessorsysteme (Dual Cores) heutzutage gängige
Rechnersysteme für sicherheitskritische Anwendungen, insbesondere im Fahrzeug wie
beispielsweise für Antiblockiersysteme, das Elektronische Stabilitätsprogramm (ESP), X-
by-Wire-Systeme wie Drive-by-Wire oder Steer-by-Wire sowie Break-by-Wire, usw.
oder auch bei sonstigen vernetzten Systemen. Um diese hohen Sicherheitsansprüche in
zukünftigen Anwendungen zu befriedigen, sind mächtige Fehlermechanismen und
25 Fehlerbehandlungsmechanismen erforderlich, insbesondere um transienten Fehler, die
beispielsweise bei Verkleinerung der Halbleiterstrukturen der Rechnersysteme entstehen,
zu begegnen. Dabei ist es relativ schwierig den Core selbst, also den Prozessor zu
schützen. Eine Lösung hierfür ist wie erwähnt die Verwendung eines
Zweirechnersystems oder Dual Core-Systems zur Fehlerdetektion.

30 Solche Prozessoreinheiten mit wenigstens zwei integrierten Ausführungseinheiten sind
somit als Dual-Core- oder Multi-Core-Architekturen bekannt. Solche Dual-Core- oder
Multi-Core-Architekturen werden nach heutigem Stand der Technik hauptsächlich aus
zwei Gründen vorgeschlagen:

35

Zum Einen kann damit eine Leistungssteigerung, also eine Performance-Steigerung erreicht werden, indem die beiden Ausführungseinheiten oder Cores als zwei Recheneinheiten auf einem Halbleiterbaustein betrachtet und behandelt werden. In dieser Konfiguration bearbeiten die zwei Ausführungseinheiten oder Cores unterschiedliche Programme respektive Tasks. Dadurch lässt sich eine Leistungssteigerung erzielen, weshalb diese Konfiguration als Leistungsmodus oder Performance-Mode bezeichnet wird.

Der zweite Grund, eine Dual-Core- oder Multi-Core-Architektur zu realisieren, ist eine Sicherheitssteigerung, indem die beiden Ausführungseinheiten redundant das gleiche Programm abarbeiten. Die Ergebnisse der beiden Ausführungseinheiten oder CPUs, also Cores werden verglichen und ein Fehler kann bei dem Vergleich auf Übereinstimmung erkannt werden. Im Folgenden wird diese Konfiguration als Sicherheitsmodus oder Safety-Mode oder auch Fehlererkennungsmodus bezeichnet.

Heutzutage gibt es somit einerseits Zwei- oder Mehrprozessorsysteme die zur Erkennung von Hardware-Fehlern redundant arbeiten (siehe Dual-Core oder Master-Checker-Systeme) und andererseits Zwei- oder Mehrprozessorsysteme, die auf ihren Prozessoren unterschiedliche Daten abarbeiten. Kombiniert man nun diese beiden Betriebsarten in einem Zwei- oder Mehrprozessorsystem (der Einfachheit halber wird nun nur noch von einem Zweiprocessorsystem gesprochen, die nachfolgende Erfindung ist aber genauso auf Mehrprozessorsystemen anwendbar), so müssen die beiden Prozessoren im Performance-Modus unterschiedliche Daten erhalten und im Fehlererkennungsmodus die gleichen Daten.

Die Aufgabe der Erfindung ist nun eine Einheit und eine Verfahren vorzustellen, die den wenigstens beiden Prozessoren oder Cores abhängig vom Modus die Instruktionen/ Daten redundant oder unterschiedlich liefert und insbesondere im Performance-Modus die Speicherzugriffsrechte aufteilt, die Synchronisation und/oder Desynchronisation der beiden Prozessoren oder Cores bei einem Moduswechsel ermöglichen.

Solch eine Einheit ist bis jetzt noch nicht bekannt. Sie ermöglicht den effektiven Betrieb eines Zweiprocessorsystems, so dass in den beiden Modi Sicherheit und Performance im Betrieb umgeschaltet werden kann. Dabei wird im weiteren von Prozessoren gesprochen, was aber ebenso Cores bzw. Recheneinheiten begrifflich einschließt.

Weiterhin ist es Aufgabe der Erfindung eine Möglichkeit anzugeben mit der eine Synchronisierung des Mehrprozessorsystems erfolgen kann. Bis jetzt ist noch kein solches Verfahren, bzw. eine Implementierung bekannt. Es gibt Multiprozessorsysteme die nur einen von beiden Modi können, aber keines, dass taktsynchron arbeitet, umschaltbar ist und die Daten taktgenau vergleichen kann. Die Synchronisation kann taktgenau und taktsynchron durch das vorgestellte Verfahren erfolgen, es kann aber auch Anwendungen geben, bei denen solch eine genaue Synchronisation nicht erforderlich ist. Dann kann dieses Verfahren auch angewandt werden um eine „lose“ Synchronisation zu erreichen. Eine lose Synchronisation ist eine Synchronisation bei der die beiden Prozessoren zwar das gleiche abarbeiten, der zeitliche Abstand der Abarbeitung jedoch in einem durch den Vergleich vorgegebenen Rahmen schwanken kann.

Beschreibung der Ausführungsbeispiele und Vorteile der Erfindung

In einem Zweirechnersystem gibt es 2 Prozessoren die dieselben oder verschiedene Aufgaben abarbeiten können. Diese beiden Prozessoren des Zweirechnersystems können diese Aufgaben taktsynchron oder taktversetzt abarbeiten.

Um diese Umschaltung in einem taktsynchronen System, bei dem die Ausgangsdaten taktsynchron verglichen werden sollen, zwischen den beiden Modi zu ermöglichen, müssen diese beide Prozessoren synchronisierbar sein. D.h., dass wenn der Prozessor vom Performance Modus (=Modus, in dem sie verschiedene Aufgaben abarbeiten und die Ausgangsdaten nicht verglichen werden) in einen sicheren Modus (=Modus, in dem die beiden Prozessoren die gleiche Aufgabe abarbeiten und deren Ausgaben in jedem Takt verglichen werden) wechselt, müssen die beiden Prozessoren im Programmablauf synchronisiert werden können.

So zeigt die Erfindung auch ein Verfahren und eine Vorrichtung bei der der Umschaltwunsch durch ein Signal ausgelöst wird,. Dies kann z.B. durch das Beobachten des Befehlsbusses erzeugt werden (schauen ob der Umschaltwunsch ausgeführt wird) oder ein Steuersignal des Decoders sein (z.B. Auslösen eines Interrupts, Schreiben in ein Register, ... bei dem anderen Prozessor) und der Prozessor dadurch an die vorgegeben Programmadresse springt.

- 4 -

Daneben können zweckmäßiger Weise die beiden Prozessoren aufgrund einer Kennung, also ID, die für jeden Prozessor in dem Mehrprozessorsystem einzigartig ist, an verschiedene Programmstellen springen und somit desynchronisiert werden (wichtig: Prozessoridentifikation durch ein ProzessorID-Bit, bedingter Sprung, Auslesen des ProzessorId-Bit aus einem für
5 jeden Prozessor separaten Speicherbereich aber mit gleicher Adresse, ProzessorId-Bit abgelegt in internem Prozessorregister).

Die Erfindung offenbart ein Verfahren und eine Vorrichtung zur Synchronisierung in einem Mehrprozessorsystem mit wenigstens zwei Prozessoren, wobei Umschaltmittel
10 enthalten sind durch welche zwischen wenigstens zwei Betriebsmodi umgeschaltet werden kann, wobei die Vorrichtung derart ausgestaltet sind, dass eine Synchronisierung durch ein Anhaltesignal durchgeführt wird, welches einen vorausseilenden Prozessor anhält um diesen mit dem wenigstens zweiten Prozessor zu synchronisieren.

15 Zweckmäßig kann die Synchronisation durch Mitteilung eines Umschaltwunsches wenigstens eines Prozessors erfolgen (z.B. Auslösen eines Interrupts, Schreiben in ein Register, ... bei dem anderen Prozessor) und der Prozessor dadurch an eine vorgegebene Programmadresse springen.

20 Die Erfindung zeigt auch eine Einheit zur Datenverteilung aus wenigstens einer Datenquelle in einem System mit wenigstens zwei Recheneinheiten, wobei Umschaltmittel (ModeSwitch) enthalten sind durch welche zwischen wenigstens zwei Betriebsmodi des Systems umgeschaltet werden kann, wobei die Einheit derart ausgestaltet ist, dass die Datenverteilung und/oder die Datenquelle (insbesondere
25 Instr.Speicher, Datenspeicher, Cache) abhängig von dem Betriebsmodus ist. Gleichmaßen ist ein System mit einer solchen Einheit gezeigt.

Dabei entspricht der erste Betriebsmodus einem Sicherheitsmodus, bei dem die zwei Recheneinheiten gleiche Programme und/oder Daten abarbeiten und Vergleichsmittel
30 vorgesehen sind, welche die bei der Abarbeitung der gleichen Programme entstehenden Zustände auf Übereinstimmung vergleichen

Die erfindungsgemäße Einheit bzw. das erfindungsgemäße Verfahren ermöglicht die Implementierung der beiden Modi in einem Zweiprozessorsystem.

35

Arbeiten die beiden Prozessoren im Fehlererkennungsmodus (F-Modus), so erhalten die beiden Prozessoren die gleichen Daten/Instruktionen und arbeiten sie im Performancemodus (P-Modus), so kann jeder Prozessor auf den Speicher zugreifen. Dann verwaltet diese Einheit die Zugriffe auf den nur einfach vorhandenen Speicher oder Peripherie.

Im F-Modus übernimmt die Einheit die Daten/Adressen eines Prozessors (hier Master genannt) und leitet diese an die Komponenten wie Speicher, Bus, usw. weiter. Der zweite Prozessor (hier Slave) möchte den gleichen Zugriff machen. Die Datenverteilungseinheit nimmt dies an einem zweiten Port entgegen, aber leitet die Anfrage nicht an die weiteren Komponenten weiter. Die Datenverteilungseinheit übergibt dem Slave die gleichen Daten wie dem Master und vergleicht die Daten der beiden Prozessoren. Sind diese unterschiedlich, so zeigt dies die Datenverteilungseinheit (hier DVE) durch ein Fehlersignal an. Es arbeitet somit nur der Master auf den Bus/Speicher und der Slave bekommt die selben Daten (Funktionsweise wie bei einem Dual-Core System).

Im P-Modus arbeiten die beiden Prozessoren unterschiedliche Programmteile ab. Die Speicherzugriffe sind somit auch unterschiedlich. Die DVE nimmt somit die Anforderung der Prozessoren entgegen und gibt die Ergebnisse/angeforderte Daten an den Prozessor zurück, der sie angefordert hat. Möchten nun beide Prozessoren gleichzeitig auf eine Komponenten zugreifen, so wird ein Prozessor in einen Wartezustand versetzt, bis der andere bedient wurde.

Die Umschaltung zwischen den beiden Modi und somit der unterschiedlichen Arbeitsweise der Datenverteilungseinheit erfolgt durch ein Steuersignal. Dies kann entweder von einem der beiden Prozessoren generiert werden oder extern.

Wird das Zweiprozessorsystem im F-Modus mit einem Taktversatz betrieben und im P-Modus nicht, so verzögert die DVE—Einheit die Daten für den Slave entsprechend, bzw. speichert die Ausgangsdaten des Master solange, bis sie mit den Ausgangsdaten des Slave zur Fehlererkennung verglichen werden können.

Der Taktversatz wird anhand der Figur 1 näher erläutert:

Figur 1 zeigt ein Zweirechnersystem mit einem ersten Rechner 100, insbesondere einem Masterrechner und einem zweiten Rechner 101, insbesondere einem Slave-Rechner. Das

gesamte System wird dabei mit einem vorgebbaren Takt bzw. in vorgebbaren Taktzyklen (clock cycle) CLK betrieben. Über den Takteingang CLK1 des Rechners 100 sowie über den Takteingang CLK2 des Rechners 101 wird diesem der Takt zugeführt. Bei diesem Zweirechnersystem ist darüber hinaus beispielhaft ein spezielles Merkmal zur Fehlererkennung enthalten, in dem nämlich der erste Rechner 100 sowie der zweite Rechner 101 mit einem Zeitversatz, insbesondere einem vorgebbaren Zeitversatz bzw. einem vorgebbaren Taktversatz arbeiten. Dabei ist jede beliebige Zeit für einen Zeitversatz vorgebbbar und auch jeder beliebige Takt bezüglich eines Versatzes der Taktzyklen. Dies kann ein ganzzahliger Versatz des Taktzyklus (clock cycle) sein, aber eben auch wie in diesem Beispiel dargestellt, beispielsweise ein Versatz von 1,5 Taktzyklen, wobei hier der erste Rechner 100 eben 1,5 Taktzyklen vor dem zweiten Rechner 101 arbeitet respektive betrieben wird. Durch diesen Versatz kann vermieden werden, dass Gleichtaktfehler, sogenannte common mode failures, die Rechner oder Prozessoren, also die Cores des Dual Cores Systems, gleichartig stören und damit unerkannt bleiben. D.h. solche Gleichtaktfehler betreffen durch den Versatz die Rechner zu unterschiedlichen Zeitpunkten im Programmablauf und bewirken demnach unterschiedliche Effekte bezüglich der beiden Rechner wodurch Fehler erkennbar werden. Gleichartige Fehlerwirkungen ohne Taktversatz wären u.U. in einem Vergleich nicht erkennbar, dies wird dadurch vermieden. Um diesen Versatz bezüglich der Zeit oder des Taktes, hier insbesondere 1,5 Taktzyklen im Zweirechnersystem zum implementieren sind die Versatzbausteine 112 bis 115 implementiert.

Um die genannten Gleichtaktfehler zu erkennen ist dieses System eben beispielsweise dazu ausgelegt in einem vorgegebenen Zeitversatz oder Taktzyklenversatz zu arbeiten, insbesondere hier 1,5 Taktzyklen, d.h. während der eine Rechner, z. B. Rechner 100 direkt die Komponenten, insbesondere die externen Komponenten 103 und 104 anspricht, arbeitet der zweite Rechner 101 mit einer Verzögerung von genau 1,5 Taktzyklen dazu. Um in diesem Fall die gewünschte Eineinhalbzyklusverzögerung, also von 1,5 Taktzyklen zu erzeugen wird Rechner 101 mit der invertierten Clock, also dem invertierten Takt am Takteingang CLK2 gespeist. Dadurch müssen aber auch die vorgenannten Anschlüsse des Rechners also seine Daten bzw. Befehle über die Busse um die genannten Taktzyklen, also hier insbesondere 1,5 Taktzyklen verzögert werden, wozu eben wie gesagt die Versatz- oder Verzögerungsbausteine 112 bis 115 vorgesehen sind.

- 7 -

Neben den beiden Rechnern oder Prozessoren 100 und 101 sind Komponenten 103 und 104 vorgesehen, die über Busse 116, bestehend aus den Busleitungen 116A und 116B und 116C sowie 117, bestehend aus den Busleitungen 117A und 117B mit den beiden Rechnern 100 und 101 in Verbindung stehen. 117 ist dabei ein Befehlsbus, bei welchem mit 117A ein Befehlsadressbus und mit 117B der Teil-Befehls(daten)bus bezeichnet ist. Der Adressbus 117A ist über einen Befehlsadressanschluss IA1 (Instruction Adress 1) mit Rechner 100 und über einen Befehlsadressanschluss IA2 (Instruction Adress 2) mit Rechner 101 verbunden. Die Befehle selbst werden über den Teil-Befehlsbus 117B übertragen, der über einen Befehlsanschluss I1 (Instruction 1) mit Rechner 100 und über einen Befehlsanschluss I2 (Instruction 2) mit Rechner 101 verbunden ist. In diesem Befehlsbus 117 bestehend aus 117A und 117B ist eine Komponente 103 z. B. ein Befehlsspeicher, insbesondere ein sicherer Befehlsspeicher oder dergleichen zwischengeschaltet. Auch diese Komponente, insbesondere als Befehlsspeicher wird in diesem Beispiel mit dem Takt CLK betrieben. Daneben ist mit 116 ein Datenbus dargestellt, welcher einen Datenadressbus oder eine Datenadressleitung 116A und einen Datenbus oder eine Datenleitung 116B enthält. Dabei ist 116A, also die Datenadressleitung, über einen Datenadressanschluss DA1 (Data Adress 1) mit dem Rechner 100 und über einen Datenadressanschluss DA2 (Data Adress 2) mit Rechner 101 verbunden. Ebenso ist der Datenbus oder die Datenleitung 116B über einen Datenanschluss DO1 (Data Out 1) und einen Datenanschluss DO2 (Data Out 2) mit Rechner 100 bzw. Rechner 101 verbunden. Weiterhin zu Datenbus 116 gehört die Datenbusleitung 116C, welche über einen Datenanschluss DI1 (Data In 1) und einen Datenanschluss DI2 (Data In 2) jeweils mit Rechner 100 bzw. Rechner 101 verbunden ist. In diesem Datenbus 116 bestehend aus den Leitungen 116A, 116B und 116C ist eine Komponente 104 zwischengeschaltet, beispielsweise ein Datenspeicher, insbesondere ein sicherer Datenspeicher o. ä. Auch diese Komponente 104 wird in diesem Beispiel mit dem Takt CLK versorgt.

Dabei stehen die Komponenten 103 und 104 stellvertretend für beliebige Komponenten die über einen Datenbus und/oder Befehlsbus mit den Rechnern des Zweirechnersystems verbunden sind und entsprechend der Zugriffe über Daten und/oder Befehle des Zweirechnersystems bezüglich Schreiboperationen und/oder Leseoperationen fehlerhafte Daten und/oder Befehle erhalten oder abgeben können. Zur Fehlervermeidung sind zwar Fehlerkennungsgeneratoren 105, 106 und 107 vorgesehen welche eine Fehlerkennung

erzeugen wie beispielsweise ein Parity-Bit oder auch einen anderen Fehlercode wie beispielsweise einen Error- Correction-Code, also ECC, o. ä.. Dazu vorgesehen sind dann auch die entsprechenden Fehlerkennungsprüfeinrichtungen oder Check-Einrichtungen 108 und 109 zur Überprüfung der jeweiligen Fehlerkennung also beispielsweise des Parity-Bit oder eines anderen Fehlercodes wie ECC.

Der Vergleich der Daten und/oder Befehle bezüglich der redundanten Ausführung im Zweirechnersystem erfolgt in den Vergleichern oder Komparatoren 110 und 111 wie in Figur 1 dargestellt. Existiert nun aber ein Zeitversatz, insbesondere ein Takt- oder Taktzyklusversatz zwischen den Rechnern 100 und 101, entweder hervorgerufen durch ein nichtsynchrones Zweiprozessorsystem oder bei einem synchronen Zweiprozessorsystem durch Fehler in der Synchronisierung oder auch wie in diesem speziellen Beispiel durch einen zur Fehlererkennung gewünschten Zeit- bzw. Taktzyklusversatz, insbesondere hier von 1,5 Taktzyklen, so kann in diesem Zeit- oder Taktversatz ein Rechner hier insbesondere Rechner 100 fehlerhafte Daten und/oder Befehle in Komponenten, insbesondere externe Komponenten wie z. B. hier insbesondere die Speicher 103 oder 104, aber auch bezüglich anderen Teilnehmern oder Aktuatoren oder Sensoren schreiben oder lesen. So kann er auch in fehlerhafter Weise einen Schreibzugriff anstatt eines vorgesehenen Lesezugriffs durch diesen Taktversatz durchführen. Diese Szenarien führen selbstverständlich zu Fehlern im gesamten System, insbesondere ohne klare Anzeigemöglichkeit welche Daten und/oder Befehle gerade fehlerhaft geändert wurden, wodurch auch die Recovery-Problematik entsteht.

Um diese Problematik zu lösen wird nun eine Verzögerungseinheit 102 wie dargestellt in die Leitungen des Datenbusses und/oder in den Befehlsbus geschaltet. Aus Gründen der Übersichtlichkeit ist nur die Einschaltung in den Datenbus dargestellt. Bezüglich des Befehlsbusses ist dies natürlich genauso möglich und denkbar. Diese Verzögerungseinheit 102 oder die Delay Unit verzögert die Zugriffe, hier insbesondere die Speicherzugriffe so, dass ein möglicher Zeit- oder Taktversatz kompensiert wird, insbesondere bei einer Fehlererkennung beispielsweise über die Komparatoren 110 und 111 z.B. mindestens solange, bis das Fehlersignal im Zweirechnersystem erzeugt ist, also die Fehlererkennung im Zweirechnersystem durchgeführt ist. Dabei können verschiedene Varianten implementiert sein:

Verzögerung der Schreib- und Leseoperationen, Verzögerung nur der Schreiboperationen oder auch, wenn auch nicht bevorzugt, eine Verzögerung der Leseoperationen. Dabei kann durch ein Änderungssignal, insbesondere das Fehlersignal, eine verzögerte Schreiboperation in eine Leseoperation gewandelt werden um fehlerhaftes Schreiben zu unterbinden.

Nachfolgend anhand Figur 2 nun eine beispielhafte Implementierung bezüglich der Datenverteilungseinheit (DVE), die sich vorzugsweise aus einer Vorrichtung zur Detektierung des Umschaltwunsches (durch IllOpDetect), der Mode-Switch-Einheit sowie dem Iram- und Dram-Control-Baustein zusammensetzt:

IllOpDetect: Die Umschaltung zwischen den beiden Modi wird durch die Einheiten "Switch-Detect" erkannt. Diese Einheit liegt zwischen dem Cache und dem Prozessor auf dem Instruktionsbus und schaut ob der Befehl IllOp in den Prozessor geladen wird. Wird der Befehl detektiert, so wird dieses Ereignis der Modeswitch Einheit mitgeteilt. Die "Switch-Detect" Einheit ist für jeden Prozessor einzeln vorhanden. Die Einheit "Switch-Detect" muss nicht fehlertolerant ausgeführt sein, da sie doppelt und somit redundant vorhanden ist. Andererseits ist es denkbar diese Einheit fehlertolerant und damit singulär auszuführen, bevorzugt ist aber die redundante Ausführung.

ModeSwitch: Die Umschaltung zwischen den beiden Modi wird durch die "Switch-Detect" Einheit getriggert. Soll eine Umschaltung vom Lock in den Split Modus erfolgen, detektieren beide "Switch-Detect" Einheiten die Umschaltung, da beide Prozessoren den gleichen Programmcode im Lock Modus abarbeiten. Die "Switch-Detect" Einheit des Prozessors 1 erkennt dies 1,5 Takte vor der "Switch-Detect" Einheit des Prozessors 2. Die "Modeswitch" Einheit hält mit Hilfe des Wait Signals den Prozessor 1 um 2 Takte an. Der Prozessor 2 wird 1,5 Takte später ebenfalls angehalten, aber nur um einen halben Takt, damit er zum Systemtakt synchronisiert wird. Anschließend wird das Status-Signal auf Split geschaltet für die weiteren Komponenten und die beiden Prozessoren arbeiten weiter. Damit die beiden Prozessoren nun unterschiedliche Tasks ausführen, müssen sie im Programmcode auseinanderlaufen. Dies erfolgt, indem direkt nach Umschalten in den Split-Modus ein Lesezugriff auf die Prozessor-ID erfolgt. Diese ausgelesene Prozessor-ID ist für jeden der beiden Prozessoren unterschiedlich. Wird nun auf eine Soll-Prozessor-ID verglichen, kann anschließend mit einem Conditional Jump Befehl der entsprechende Prozessor an eine

andere Programmstelle gebracht werden. Bei einer Umschaltung vom Split-Modus in den Lock-Modus wird dies ein Prozessor bemerken, bzw. einer der beiden zuerst. Dieser Prozessor wird Programmcode ausführen, in dem der Umschaltbefehl enthalten ist. Dies wird nun durch die "Switch-Detect" Einheit registriert und teilt dies der Modeswitch Einheit mit. Diese hält den entsprechenden Prozessor an und teilt dem zweiten den Wunsch der Synchronisation durch einen Interrupt mit. Der zweite Prozessor erhält einen Interrupt und kann nun eine Softwareroutine zur Beendigung seines Tasks ausführen. Nun springt er ebenfalls an die Programmstelle, in der sich der Befehl zur Umschaltung befindet. Seine "Switch-Detect" Einheit signalisiert nun ebenfalls den Wunsch zum Moduswechsel an die Modeswitch Einheit. Zur nächsten steigenden Systemtaktflanke wird nun das Wait Signal für den Prozessor 1 deaktiviert und 1,5 Takte später für den Prozessor 2. Nun arbeiten beide wieder mit einem Taktversatz von 1,5 Takten synchron.

Befindet sich das System im Lock Modus, so müssen beide "Switch-Detect" Einheiten der Modeswitch Einheit mitteilen, dass sie in den Split Modus wollen. Erfolgt der Umschaltwunsch nur von einer Einheit, so wird der Fehler von den Vergleichseinheiten erkannt, da diese von einem der beiden Prozessoren weiterhin Daten geliefert bekommen und diese nicht mit dem angehaltenen Prozessoren übereinstimmen.

Sind die beiden Prozessoren im Split Modus und einer schaltet nicht zurück in den Lock-Modus, so kann dies durch einen externen Watchdog erkannt werden. Bei einem Triggersignal für jeden Prozessor bemerkt der Watchdog dass der wartende Prozessor sich nicht mehr meldet. Ist nur ein Watchdogsignal für das Prozessorsystem vorhanden, so darf die Triggerung des Watchdogs nur im Lock-Modus erfolgen. Somit würde der Watchdog erkennen, dass die Modusumschaltung nicht erfolgte. Das Modussignal liegt als Dual-Rail Signal vor. Dabei steht "10" für den Lock-Modus und "01" für den Split-Modus. Bei "00" und "11" sind Fehler aufgetreten.

IramControl: Der Zugriff auf den Befehlsspeicher der beiden Prozessoren wird über die IRAM Control gesteuert. Diese muss sicher ausgelegt sein, da sie ein Single Point of Failure ist. Sie besteht aus zwei Zustandsautomaten für jeden Prozessor, also je einen taktsynchronen iram1clkreset und einen asynchronen readiram1. Im sicherheitskritischen Modus überwachen sich die Zustandsautomaten der beiden Prozessoren gegenseitig und im Performancemodus arbeiten sie getrennt.

Das Nachladen der beiden Caches der Prozessoren werden durch 2 Zustandsautomaten gesteuert. Einem synchronen Zustandsautomaten iramclkreset und einem asynchronen readiram. Durch diese beiden Zustandsautomaten werden auch die Speicherzugriffe im Split-Modus verteilt. Hierbei hat Prozessor 1 die höhere Priorität. Nach einem Zugriff auf den Hauptspeicher durch Prozessor 1 bekommt nun -- wenn beide Prozessoren wieder auf den Hauptspeicher zugreifen wollen -- Prozessor 2 die Speicherzugriffserlaubnis zugeteilt. Diese beiden Zustandsautomaten sind für jeden Prozessor implementiert. Im Lock-Modus werden die Ausgangssignale der Automaten verglichen um auftretende Fehler erkennen zu können.

Die Daten zum Aktualisieren des Cache 2 im Lock-Modus werden in der IRAM-Control Einheit um 1,5 Takte verzögert.

In Bit 5 im Register 0 der SysControl wird codiert um welchen Core es sich handelt. Core 1 ist das Bit 0 und bei Core 2 ist es High. Dieses Register ist in den Speicherbereich mit der Adresse 65528 gespiegelt.

Bei einem Speicherzugriff von Core 2 wird erst überprüft in welchem Modus sich der Rechner befindet. Ist er im Lock-Modus so wird sein Speicherzugriff unterdrückt. Dieses Signal liegt als Common-Rail Signal vor, da es sicherheitskritisch ist.

Der Programmcounter des Prozessors 1 wird um 1,5 Takte verzögert um im Lock-Modus mit dem Programmcounter des Prozessors 2 verglichen werden zu können.

Im Split Modus können die Caches der beiden Prozessoren unterschiedlich nachgeladen werden. Wenn nun in den Lock-Modus umgeschaltet wird, sind die beiden Caches nicht kohärent zueinander. Dadurch können die beiden Prozessoren auseinanderlaufen und die Vergleiche signalisieren folglich einen Fehler. Um dies zu vermeiden, ist in der IRAM Control eine Flag Tabelle aufgebaut. In dieser wird vermerkt, ob eine Cachezeile im Lock- oder im Split-Modus geschrieben wurde. Im Lock-Modus wird der für die Cachezeile entsprechende Eintrag bei einer Cachezeilennachladung auf 0 gesetzt und im Split-Modus -- auch bei einer Cacheaktualisierung der Cachezeile von nur einem Cache -- auf 1. Führt der Prozessor nun im Lock-Modus einen Speicherzugriff aus, so wird

überprüft, ob diese Cachezeile im Lock-Modus aktualisiert wurde, d.h. in beiden Caches gleich ist. Im

Split-Modus kann der Prozessor immer auf die Cachezeile zugreifen, unabhängig wie der Flag_Vector ist. Diese Tabelle muss nur einmal vorhanden sein, da bei einem Fehler die
5 beiden Prozessoren auseinanderlaufen und somit an den Vergleichen dieser Fehler sicher erkannt wird. Da die Zugriffszeiten auf der zentralen Tabelle relativ hoch sind, kann diese Tabelle auch zu jedem Cache kopiert werden.

10 DramControl: In dieser Komponente werden für die Adress-, Daten- und Speichersteuersignale von jedem Prozessor das Parity gebildet.

Es gibt einen Prozess für beide Prozessor zum Sperren des Speichers. Dieser Prozess muss nicht sicher implementiert sein, da im Lock-Modus fehlerhafte Speicherzugriffe durch die Vergleiche erkannt werden und im Split-Modus keine sicherheitsrelevanten
15 Anwendungen ausgeführt werden. Hierin wird überprüft, ob der Prozessor den Speicher für den anderen Prozessor sperren möchte. Dieses Sperren des Datenspeichers erfolgt durch einen Zugriff auf die Speicheradresse \$FBFF\$=64511. Dieses Signal soll genau ein Takt lang anliegen, auch wenn am Prozessor zum Zeitpunkt des Aufrufens ein wait-command anliegt. Der Zustandsautomat zur Verwaltung der Datenspeicherzugriffe
20 besteht aus 2 Hauptzuständen:

- Prozessorstatus Lock: Die beiden Prozessoren arbeiten im Lock-Modus. D.h. die Funktionalität des Datenspeicherlocking ist nicht notwendig. Prozessor 1 koordiniert die Speicherzugriffe.
- Prozessorstatus Split: Nun ist eine Zugriffskonfliktauflösung auf den Datenspeicher
25 nötig und ein Speichersperren muss erfolgen können.

Der Zustand im Split-Modus ist wiederum in 7 Zustände untergliedert, die die Zugriffskonflikte auflösen und den

Datenspeicher für jeweils den anderen Prozessor sperren können. Bei gleichzeitigem
30 Wunsch der beiden Prozessoren bei einem Zugriff, stellt die aufgeführte Reihenfolge gleichzeitig die Priorisierung dar.

- Core1_Lock: Prozessor 1 hat den Datenspeicher gesperrt. Möchte in diesem Zustand Prozessor 2 auf den Speicher zugreifen, so wird er durch ein Wartesignal angehalten, bis
35 Prozessor 1 den Datenspeicher wieder freigibt. \

- 13 -

- Core2_Lock: Ist der gleiche Zustand wie der vorige nur dass nun Prozessor 2 den Datenspeicher gesperrt hat und Prozessor 1 bei Datenspeicheroperationen angehalten wird.
- lock1_wait: Der Datenspeicher war durch den Prozessor 2 gesperrt als Prozessor 1 ihn ebenfalls für sich reservieren wollte. Prozessor 1 ist somit für die nächste Speichersperrung vorgemerkt.
- nex: Das gleiche für Prozessor 2. Der Datenspeicher war während des Sperrversuchs durch Prozessor 1 gesperrt. Prozessor 2 bekommt den Speicher vorreserviert. Bei normalen Speicherzugriff ohne Sperren kann hier Prozessor 2 vor Prozessor 1 zugreifen wenn davor Prozessor 1 dran war.
- Speicherzugriff von Prozessor 1: Der Speicher ist in diesem Fall nicht gesperrt. Prozessor 1 darf auf den Datenspeicher zugreifen. Falls er ihn sperren möchte, kann er dies in diesem Zustand vornehmen.
- Speicherzugriff durch Prozessor 2. Im selben Takt wollte Prozessor 1 nicht auf den Speicher zugreifen somit ist der Speicher frei für den Prozessor 2.
- kein Prozessor möchte auf den Datenspeicher zugreifen

Die DVE setzt sich wie erwähnt zusammen aus dem Detektierung des Umschaltwunsches (IllOPDetect) der ModeSwitch-Einheit und der Iram- und DramControl.

Die Mode-Switch Funktionsweise wird nachfolgend anhand Figur 3 nochmals erläutert:

Die Umschaltung der beiden Prozessoren wird hier beispielhaft durch den Befehl IllOp im Programm getriggert. Eine Voraussetzung ist, dass jeder Prozessor eindeutig identifiziert werden kann. Hierzu wird jedem Prozessor eine Ziffer zugeordnet. Hier in diesem Bsp. ist ein Core die 1 und der andere die 0. Dies ist im Prozessorstatusregister codiert.

Die beiden Prozessoren werden hier zur Synchronisation durch das Wait-Kommando angehalten. Ebenso kann der Takt für den Prozessor, der angehalten werden soll, solange gestoppt werden (durch z.B. eine OR-Verknüpfung mit 0 zum anhalten und 1 zum weitertakten).

„Switch-Detect“ Einheit: Die Umschaltung zwischen den beiden Modi wird durch die Einheiten "Switch-Detect" erkannt. Diese Einheit liegt zwischen dem Cache und dem

Prozessor auf dem Instruktionsbus und schaut ob der Befehl IllOp in den Prozessor geladen wird. Wird der Befehl detektiert, so wird dieses Ereignis der Modeswitch Einheit mitgeteilt. Diese Erkennen wird durch das „Core 1-Signal“ oder „Core 2-Signal“ (siehe Bild2) der „Mode-Switch“ Einheit mitgeteilt. Die "Switch-Detect" Einheit ist für jeden
5 Prozessor einzeln vorhanden. Die Einheit "Switch-Detect" muss nicht fehlertolerant ausgeführt sein, da sie doppelt und somit redundant vorhanden ist.

„Mode-Switch“ Einheit: Die Umschaltung zwischen den beiden Modi wird durch die "Switch-Detect" Einheit getriggert. Soll eine Umschaltung vom Lock in den Split Modus
10 erfolgen, detektieren beide "Switch-Detect" Einheiten die Umschaltung, da beide Prozessoren den gleichen Programmcode im Lock Modus abarbeiten. Die "Switch-Detect" Einheit des Prozessor 1 erkennt dies 1,5 Takte vor der "Switch-Detect" Einheit des Prozessors 2. Die "Modeswitch" Einheit hält mit Hilfe des Wait Signals den Prozessor 1 um 2 Takte an. Der Prozessor 2 wird 1,5 Takte später ebenfalls angehalten,
15 aber nur um einen halben Takt, damit er zum Systemtakt synchronisiert wird. Anschließend wird das Status-Signal auf Split geschaltet für die weiteren Komponenten und die beiden Prozessoren arbeiten weiter. Damit die beiden Prozessoren nun unterschiedliche Tasks ausführen, müssen sie im Programmcode auseinanderlaufen. Dies erfolgt, indem direkt nach Umschalten in den Split-Modus ein Lesezugriff auf die
20 Prozessor-ID erfolgt. Diese ausgelesene Prozessor-ID ist für jeden der beiden Prozessoren unterschiedlich. Wird nun auf eine Soll-Prozessor-ID verglichen, kann anschließend mit einem Conditional Jump Befehl der entsprechende Prozessor an eine andere Programmstelle gebracht werden. Ein Beispiel für eine Trennung der beiden Prozessoren sieht folgendermaßen aus:

Bei dem Splitvorgang wird zuerst Adresse die des Statusregisters in dem die Prozessor-ID abgelegt ist in r1 geschrieben

LDL r1,248

LDH r1,255

anschließend Trennung der beiden Prozessoren durch IllOp-Befehl
error

Inhalt Statusregister von Adresse r1 in r2 laden

LDW r2,r1

Bit an 5.Stelle des Statusregisters testen. Wenn es 0 ist, dann ist es Prozessor 1, bei 1
35 handelt es sich um Prozessor 2

BTEST r2,5

durch einen bedingten Sprung wird nun Prozessor 2 an eine andere Programmstelle gebracht

JMPI_CT 10

5

Bei einer Umschaltung vom Split-Modus in den Lock-Modus wird dies ein Prozessor bemerken, bzw. einer der beiden zuerst. Dieser Prozessor wird Programmcode ausführen, in dem der Umschaltbefehl enthalten ist. Dies wird nun durch die "Switch-Detect" Einheit registriert und teilt dies der „Mode-Switch“ Einheit mit. Diese hält den entsprechenden Prozessor an und teilt dem zweiten den Wunsch der Synchronisation durch einen Interrupt („Message 1“ oder „Message 2“ in Bild 2) mit. Der zweite Prozessor erhält einen Interrupt und kann nun eine Softwareroutine zur Beendigung seines Tasks ausführen. Nun springt er ebenfalls an die Programmstelle, in der sich der Befehl zur Umschaltung befindet. Seine "Switch-Detect" Einheit signalisiert nun ebenfalls den Wunsch zum Moduswechsel an die Modeswitch Einheit. Zur nächsten steigenden Systemtaktflanke wird nun das Wait Signal für den Prozessor 1 deaktiviert und 1,5 Takte später für den Prozessor 2. Nun arbeiten beide wieder mit einem Taktversatz von 1,5 Takten synchron.

10

15

20

Befinden sich das System im Lock Modus, so müssen beide "Switch-Detect" Einheiten der Modeswitch Einheit mitteilen, dass sie in den Split Modus wollen. Erfolgt der Umschaltwunsch nur von einer Einheit, so wird der Fehler von den Vergleichseinheiten erkannt, da diese von einem der beiden Prozessoren weiterhin Daten geliefert bekommen und diese nicht mit dem angehaltenen Prozssoren übereinstimmen.

25

Sind die beiden Prozessoren im Split Modus und einer schaltet nicht zurück in den Lock-Modus, so kann dies durch einen externen Watchdog erkannt werden. Bei einem Triggersignal für jeden Prozessor bemerkt der Watchdog dass der wartende Prozessor sich nicht mehr meldet. Ist nur ein Watchdogsignal für das Prozessorsystem vorhanden, so darf die Triggerung des Watchdogs nur im Lock-Modus erfolgen. Somit würde der Watchdog erkennen, dass die Modusumschaltung nicht erfolgte. Das Modussignal liegt als Dual-Rail Signal vor (in Bild 2 wird es Status genannt). Dabei steht "10" für den Lock-Modus und "01" für den Split-Modus. Bei "00" und "11" sind Fehler aufgetreten.

30

Da der Befehl zur Umschaltung am Anfang der Pipeline des Prozessors detektiert wird, darf in den Pipeline Stufen nach der Detektierung keine Sprünge vorhanden sein. Die einfachste Methode dies zu verhindern ist vor dem Befehl IllOp zwei NOP einzuführen.

5 Der Kern der Erfindung ist wie oben ausgeführt die generelle Funktionsweise des Mode-Switch Verfahrens (je nach Modus unterschiedliche Datenzuteilung und somit auch Auswahl des Betriebsmodus) und dabei speziell die Synchronisierung der Prozessoren.

10 Daneben löst aber auch die dargestellte spezielle Implementierung die Eingangs genannten Aufgaben.

5

Ansprüche

10

1. Verfahren zur Synchronisierung in einem Mehrprozessorsystem mit wenigstens zwei Prozessoren, wobei Umschaltmittel enthalten sind durch welche zwischen wenigstens zwei Betriebsmodi umgeschalten werden kann, wobei eine Synchronisierung durch ein Anhaltesignal durchgeführt wird, welches einen vorausseilenden Prozessor anhält um diesen mit dem wenigstens zweiten Prozessor zu synchronisieren.

15

2. Verfahren zur Synchronisierung nach Anspruch 1, dadurch gekennzeichnet, dass die Synchronisation durch einen Synchronisationswunsch ausgelöst wird, wobei der Synchronisationswunsch von einem oder mehreren Prozessoren erfolgen kann.

20

3. Verfahren zur Synchronisierung nach Anspruch 1, dadurch gekennzeichnet, dass als Anhaltesignal das Wait-Signal eines Prozessors verwendet wird.

4. Verfahren zur Synchronisierung nach Anspruch 1, dadurch gekennzeichnet, dass als Anhaltesignal ein Interrupt-Signal ausgelöst wird.

25

5. Verfahren zur Synchronisierung nach Anspruch 1, dadurch gekennzeichnet, dass der Prozessor für die Synchronisation durch ein Auslassen von Taktzyklen angehalten wird.

30

6. Verfahren zur Synchronisierung nach Anspruch 1, dadurch gekennzeichnet, dass der Prozessor für die Synchronisation durch ein Abschalten eines Taktsignales angehalten wird.

7. Verfahren zur Synchronisierung nach Anspruch 1, dadurch gekennzeichnet, dass die Umschaltung durch einen Umschaltwunsch dargestellt wird und der Umschaltwunsch durch

ein Signal ausgelöst wird.

8. Verfahren zur Synchronisierung nach Anspruch 1, dadurch gekennzeichnet, dass die Umschaltung durch einen Umschaltwunsch dargestellt wird, wobei die Umschaltung erst erfolgt, wenn der Umschaltwunsch von zwei oder mehreren Prozessoren gefordert wird.

9. Verfahren zur Synchronisierung nach Anspruch 1, dadurch gekennzeichnet, dass die Umschaltung durch einen Umschaltwunsch ausgelöst wird, wobei nach einem Umschaltwunsch der Betriebsmodus des Mehrprozessorsystems gewechselt wird und der Umschaltwunsch durch ein Signal angezeigt wird.

10. Verfahren zur Synchronisierung nach Anspruch 1, dadurch gekennzeichnet, dass der vorliegende Betriebsmodus durch ein Modussignal angezeigt wird.

11. Verfahren zur Synchronisierung nach Anspruch 10, dadurch gekennzeichnet, dass, das Modussignal als codiertes Signal, insbesondere als Dual-Rail-Signal, vorliegt.

12. Verfahren zur Synchronisierung nach Anspruch 10, dadurch gekennzeichnet, dass das Modussignal redundant erzeugt wird, insbesondere durch zwei Zustandsautomaten oder durch eine Dual-Rail-Logik.

13. Verfahren zur Synchronisierung nach Anspruch 1, dadurch gekennzeichnet, dass ein Synchronisationswunsch an eine zentrale Einheit geleitet wird und diese den Synchronisationswunsch an den wenigstens einen weiteren Prozessor weiterleitet.

14. Verfahren zur Synchronisierung nach Anspruch 1, dadurch gekennzeichnet, dass die Synchronisation durch Mitteilung des Umschaltwunsches erfolgt und der Prozessor dadurch an eine vorgegebene Programmadresse springt.

15. Verfahren zur Synchronisierung nach Anspruch 1, dadurch gekennzeichnet, dass der eine Prozessor solange angehalten wird bis der andere eine Task abgearbeitet hat und dann ebenfalls an der gleichen Programmstelle angelangt ist.

16. Verfahren zur Synchronisierung nach Anspruch 1, dadurch gekennzeichnet, dass nach einer Synchronisierung zur Desynchronisierung die beiden Prozessoren aufgrund einer Kennung (ID), die für jeden Prozessor in dem Mehrprozessorsystem einzigartig ist, an verschiedene Programmstellen springen und somit desynchronisiert werden.

5

17. Vorrichtung zur Synchronisierung in einem Mehrprozessorsystem mit wenigstens zwei Prozessoren, wobei Umschaltmittel enthalten sind durch welche zwischen wenigstens zwei Betriebsmodi umgeschaltet werden kann, wobei die Vorrichtung derart ausgestaltet sind, dass eine Synchronisierung durch ein Anhaltesignal durchgeführt wird, welches einen vorausseilenden Prozessor anhält um diesen mit dem wenigstens zweiten Prozessor zu synchronisieren.

10

18. Vorrichtung zur Synchronisation nach Anspruch 17, dadurch gekennzeichnet, dass der erste Betriebsmodus einem Sicherheitsmodus entspricht, bei dem die zwei Prozessoren gleiche Programme abarbeiten und Vergleichsmittel vorgesehen sind, welche die bei der Abarbeitung der gleichen Programme entstehenden Zustände auf Übereinstimmung vergleichen.

15

19. Vorrichtung zur Synchronisierung nach Anspruch 1, dadurch gekennzeichnet, dass die Vorrichtung derart ausgestaltet ist, dass als Anhaltesignal das Wait-Signal eines Prozessors verwendet wird.

20

20. Vorrichtung zur Synchronisierung nach Anspruch 17, dadurch gekennzeichnet, dass die Vorrichtung derart ausgestaltet ist, dass als Anhaltesignal ein Interrupt-Signal ausgelöst wird.

25

21. Vorrichtung zur Synchronisierung nach Anspruch 17, dadurch gekennzeichnet, dass die Vorrichtung derart ausgestaltet ist, dass der Prozessor für die Synchronisation durch ein Auslassen von Taktzyklen angehalten wird.

30

22. Vorrichtung zur Synchronisierung nach Anspruch 17, dadurch gekennzeichnet, dass die Vorrichtung derart ausgestaltet ist, dass der Prozessor für die Synchronisation durch ein Abschalten eines Taktsignales angehalten wird.

35

23. Vorrichtung zur Synchronisierung nach Anspruch 17, dadurch gekennzeichnet, dass die Vorrichtung derart ausgestaltet ist, dass der vorliegende Betriebsmodus durch ein Modussignal angezeigt wird.

5 24. Vorrichtung zur Synchronisierung nach Anspruch 23, dadurch gekennzeichnet, dass die Vorrichtung derart ausgestaltet ist, dass das Modussignal als codiertes Signal, insbesondere als Dual-Rail-Signal, vorliegt.

10 25. Vorrichtung zur Synchronisierung nach Anspruch 23, dadurch gekennzeichnet, dass die Vorrichtung derart ausgestaltet ist, dass das Modussignal redundant erzeugt wird, insbesondere durch zwei Zustandsautomaten oder durch eine Dual-Rail-Logik.

15 26. Vorrichtung zur Synchronisierung nach Anspruch 17, dadurch gekennzeichnet, dass eine zentrale Einheit vorgesehen ist und die Vorrichtung derart ausgestaltet ist, dass ein Synchronisationswunsch an eine zentrale Einheit geleitet wird und diese den Synchronisationswunsch an den wenigstens einen weiteren Prozessor weiterleitet.

20 27. Vorrichtung zur Synchronisierung nach Anspruch 17, dadurch gekennzeichnet, dass die Vorrichtung derart ausgestaltet ist, dass der eine Prozessor solange angehalten wird bis der wenigstens zweite eine Task abgearbeitet hat und dann ebenfalls an der gleichen Programmstelle angelangt ist.

25 28. Vorrichtung zur Synchronisierung nach Anspruch 17, dadurch gekennzeichnet, dass die Vorrichtung derart ausgestaltet ist, dass nach einer Synchronisierung zur Desynchronisierung die beiden Prozessoren aufgrund einer Kennung (ID), die für jeden Prozessor in dem Mehrprozessorsystem einzigartig ist, an verschiedene Programmstellen springen und somit desynchronisiert werden.

30 29. Vorrichtung zur Synchronisierung nach Anspruch 28, dadurch gekennzeichnet, dass ein Prozessorregister vorgesehen ist und die Kennung in dem Prozessorregister abgelegt ist.

30. Vorrichtung nach Anspruch 28, dadurch gekennzeichnet, dass die Kennung extern zu den Prozessoren abgelegt ist, insbesondere in einer zentralen Einheit (DramCtrl).

31. Vorrichtung zur Synchronisierung nach Anspruch 17, dadurch gekennzeichnet, dass das Umschaltmittel zur Umschaltung des Betriebsmodus fehlertolerant ausgelegt ist, insbesondere durch Duplizieren von Zustandsautomaten oder Implementierung in Dual-Rail Logik.

5

32. Mehrprozessorsystem mit einer Vorrichtung nach einem der Ansprüche 17 bis 31.

10

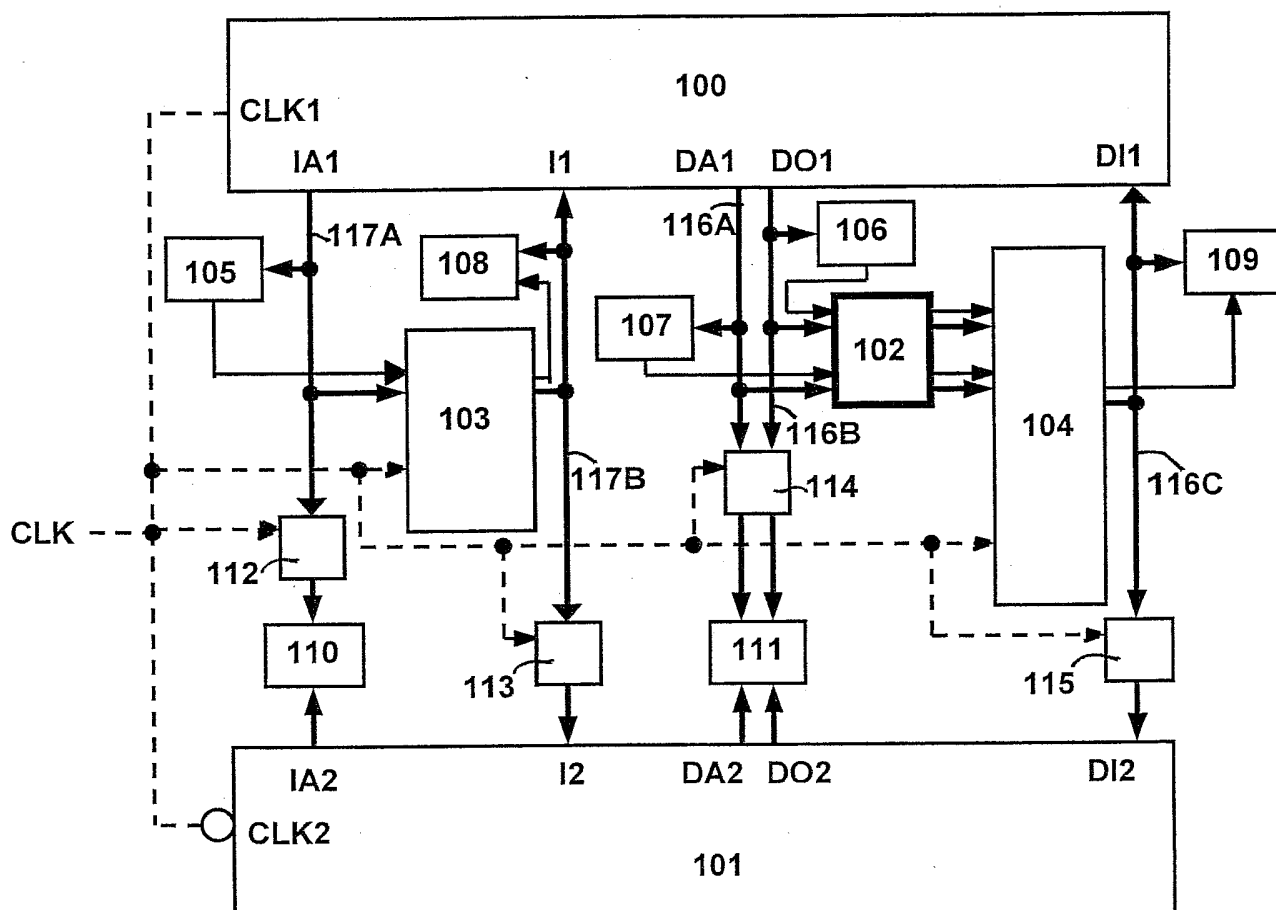


Fig. 1

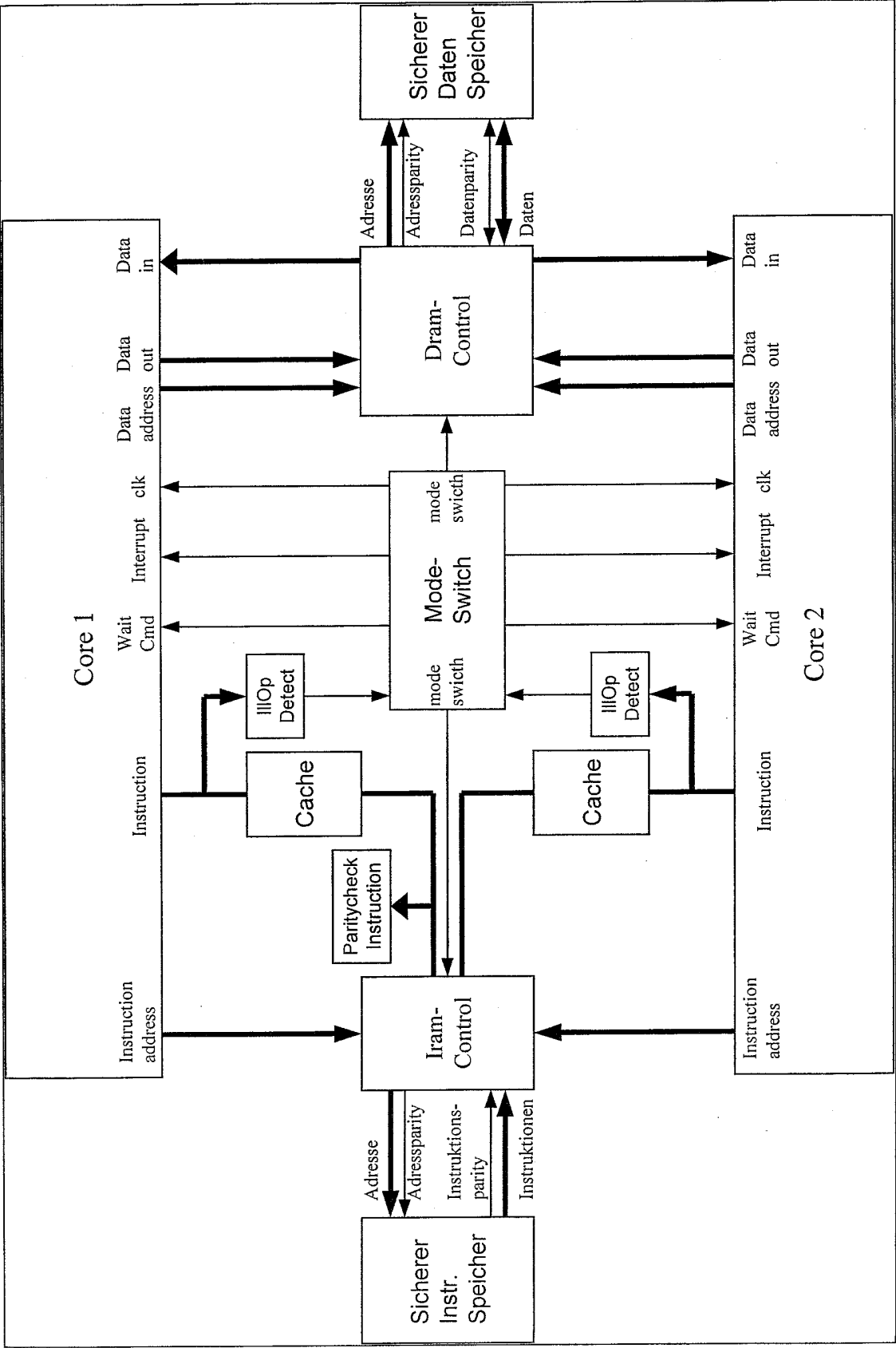


Fig. 2

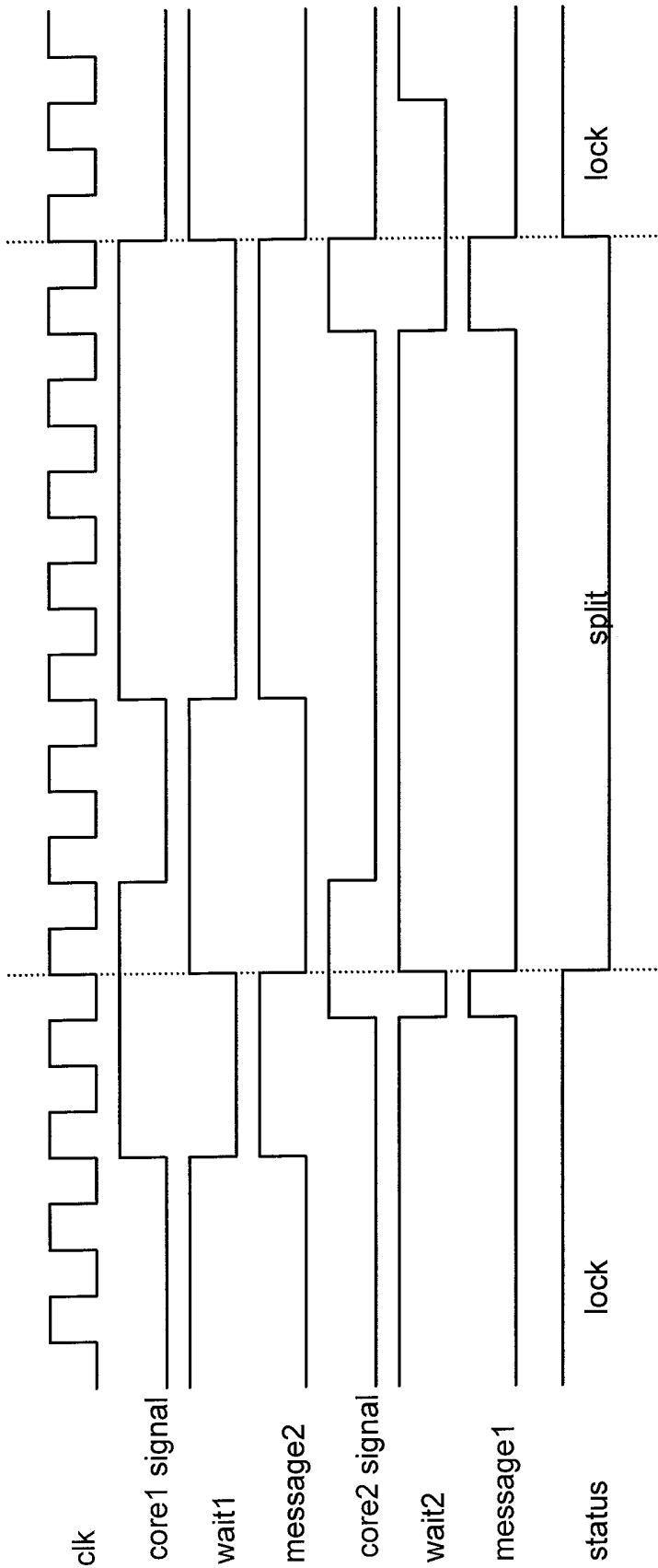


Fig. 3

INTERNATIONAL SEARCH REPORT

International application No

PCT/EP2005/055537

A. CLASSIFICATION OF SUBJECT MATTER

G06F11/16

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

G06F

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practical, search terms used)

EPO-Internal, WPI Data

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
Y A	US 6 615 366 B1 (GROCHOWSKI EDWARD T ET AL) 2 September 2003 (2003-09-02) column 2, lines 27-52 column 5, line 62 - column 6, line 11 column 7, line 46 - column 9, line 58 figures 1,3A	1-15, 17-27,32 16,28-30
Y	DE 12 69 827 B (SIEMENS AKTIENGESELLSCHAFT) 6 June 1968 (1968-06-06) column 1, line 1 - column 3, line 31	1-15, 17-27,32
Y	DE 41 04 114 A1 (SIEMENS AG, 8000 MUENCHEN, DE; SIEMENS AG) 13 August 1992 (1992-08-13) column 1, line 56 - column 2, line 46 ----- -/--	4,20

☒ Further documents are listed in the continuation of Box C.

☒ See patent family annex.

* Special categories of cited documents :

A document defining the general state of the art which is not considered to be of particular relevance

E earlier document but published on or after the international filing date

L document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

O document referring to an oral disclosure, use, exhibition or other means

P document published prior to the international filing date but later than the priority date claimed

T later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

X document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

Y document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art.

G document member of the same patent family

Date of the actual completion of the international search

8 February 2006

Date of mailing of the international search report

17/02/2006

Name and mailing address of the ISA/

European Patent Office, P.B. 5818 Patentlaan 2
NL - 2280 HV Rijswijk
Tel. (+31-70) 340-2040, Tx. 31 651 epo nl,
Fax: (+31-70) 340-3016

Authorized officer

Lanchès, P

INTERNATIONAL SEARCH REPORT

International application No

PCT/EP2005/055537

C(Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
P,X	WO 2005/003962 A (ROBERT BOSCH GMBH; WEIBERLE, REINHARD; MUELLER, BERND; ANGERBAUER, RAL) 13 January 2005 (2005-01-13) page 2, line 16 - page 3, line 17 page 5, line 22 - page 6, line 2 page 10, line 16 - page 12, line 26 -----	1,17
A	US 6 519 710 B1 (SAUNDERS JONATHAN M ET AL) 11 February 2003 (2003-02-11) column 2, line 29 - column 5, line 22 -----	1-32
A	KANEKAWA N ET AL: "Fault detection and recovery coverage improvement by clock synchronized duplicated systems with optimal time diversity" FAULT-TOLERANT COMPUTING, 1998. DIGEST OF PAPERS. TWENTY-EIGHTH ANNUAL INTERNATIONAL SYMPOSIUM ON MUNICH, GERMANY 23-25 JUNE 1998, LOS ALAMITOS, CA, USA, IEEE COMPUT. SOC, US, 23 June 1998 (1998-06-23), pages 196-200, XP010291315 ISBN: 0-8186-8470-4 page 197, left-hand column, line 1 - page 199, left-hand column, line 21 -----	1-32

INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No

PCT/EP2005/055537

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 6615366	B1	02-09-2003	NONE
DE 1269827	B	06-06-1968	NONE
DE 4104114	A1	13-08-1992	NONE
WO 2005003962	A	13-01-2005	NONE
US 6519710	B1	11-02-2003	CN 1248748 A 29-03-2000
		DE 69908717 D1	17-07-2003
		DE 69908717 T2	11-12-2003
		EP 0980040 A2	16-02-2000
		ES 2199527 T3	16-02-2004
		GB 2340627 A	23-02-2000
		HK 1024067 A1	01-04-2005
		RU 99118019 A	27-08-2001

INTERNATIONALER RECHERCHENBERICHT

Internationales Aktenzeichen

PCT/EP2005/055537

A. KLASSIFIZIERUNG DES ANMELDUNGSGEGENSTANDES
G06F11/16

Nach der Internationalen Patentklassifikation (IPC) oder nach der nationalen Klassifikation und der IPC

B. RECHERCHIERTE GEBIETE

Recherchierter Mindestprüfstoff (Klassifikationssystem und Klassifikationssymbole)
G06F

Recherchierte, aber nicht zum Mindestprüfstoff gehörende Veröffentlichungen, soweit diese unter die recherchierten Gebiete fallen

Während der internationalen Recherche konsultierte elektronische Datenbank (Name der Datenbank und evtl. verwendete Suchbegriffe)

EPO-Internal, WPI Data

C. ALS WESENTLICH ANGESEHENE UNTERLAGEN

Kategorie*	Bezeichnung der Veröffentlichung, soweit erforderlich unter Angabe der in Betracht kommenden Teile	Betr. Anspruch Nr.
Y A	US 6 615 366 B1 (GROCHOWSKI EDWARD T ET AL) 2. September 2003 (2003-09-02) Spalte 2, Zeilen 27-52 Spalte 5, Zeile 62 - Spalte 6, Zeile 11 Spalte 7, Zeile 46 - Spalte 9, Zeile 58 Abbildungen 1,3A	1-15, 17-27,32 16,28-30
Y	DE 12 69 827 B (SIEMENS AKTIENGESELLSCHAFT) 6. Juni 1968 (1968-06-06) Spalte 1, Zeile 1 - Spalte 3, Zeile 31	1-15, 17-27,32
Y	DE 41 04 114 A1 (SIEMENS AG, 8000 MUENCHEN, DE; SIEMENS AG) 13. August 1992 (1992-08-13) Spalte 1, Zeile 56 - Spalte 2, Zeile 46	4,20
	----- -/--	

☒ Weitere Veröffentlichungen sind der Fortsetzung von Feld C zu entnehmen ☒ Siehe Anhang Patentfamilie

- * Besondere Kategorien von angegebenen Veröffentlichungen :
 - *A* Veröffentlichung, die den allgemeinen Stand der Technik definiert, aber nicht als besonders bedeutsam anzusehen ist
 - *E* älteres Dokument, das jedoch erst am oder nach dem internationalen Anmeldedatum veröffentlicht worden ist
 - *L* Veröffentlichung, die geeignet ist, einen Prioritätsanspruch zweifelhaft erscheinen zu lassen, oder durch die das Veröffentlichungsdatum einer anderen im Recherchenbericht genannten Veröffentlichung belegt werden soll oder die aus einem anderen besonderen Grund angegeben ist (wie ausgeführt)
 - *O* Veröffentlichung, die sich auf eine mündliche Offenbarung, eine Benutzung, eine Ausstellung oder andere Maßnahmen bezieht
 - *P* Veröffentlichung, die vor dem internationalen Anmeldedatum, aber nach dem beanspruchten Prioritätsdatum veröffentlicht worden ist
- *T* Spätere Veröffentlichung, die nach dem internationalen Anmeldedatum oder dem Prioritätsdatum veröffentlicht worden ist und mit der Anmeldung nicht kollidiert, sondern nur zum Verständnis des der Erfindung zugrundeliegenden Prinzips oder der ihr zugrundeliegenden Theorie angegeben ist
- *X* Veröffentlichung von besonderer Bedeutung; die beanspruchte Erfindung kann allein aufgrund dieser Veröffentlichung nicht als neu oder auf erfinderischer Tätigkeit beruhend betrachtet werden
- *Y* Veröffentlichung von besonderer Bedeutung; die beanspruchte Erfindung kann nicht als auf erfinderischer Tätigkeit beruhend betrachtet werden, wenn die Veröffentlichung mit einer oder mehreren anderen Veröffentlichungen dieser Kategorie in Verbindung gebracht wird und diese Verbindung für einen Fachmann naheliegend ist
- *Z* Veröffentlichung, die Mitglied derselben Patentfamilie ist

Datum des Abschlusses der internationalen Recherche

8. Februar 2006

Absenddatum des internationalen Recherchenberichts

17/02/2006

Name und Postanschrift der Internationalen Recherchenbehörde
Europäisches Patentamt, P.B. 5818 Patentlaan 2
NL - 2280 HV Rijswijk
Tel. (+31-70) 340-2040, Tx. 31 651 epo nl,
Fax: (+31-70) 340-3016

Bevollmächtigter Bediensteter

Lanchès, P

C. (Fortsetzung) ALS WESENTLICH ANGESEHENE UNTERLAGEN

Kategorie*	Bezeichnung der Veröffentlichung, soweit erforderlich unter Angabe der in Betracht kommenden Teile	Betr. Anspruch Nr.
P,X	WO 2005/003962 A (ROBERT BOSCH GMBH; WEIBERLE, REINHARD; MUELLER, BERND; ANGERBAUER, RAL) 13. Januar 2005 (2005-01-13) Seite 2, Zeile 16 - Seite 3, Zeile 17 Seite 5, Zeile 22 - Seite 6, Zeile 2 Seite 10, Zeile 16 - Seite 12, Zeile 26 -----	1,17
A	US 6 519 710 B1 (SAUNDERS JONATHAN M ET AL) 11. Februar 2003 (2003-02-11) Spalte 2, Zeile 29 - Spalte 5, Zeile 22 -----	1-32
A	KANEKAWA N ET AL: "Fault detection and recovery coverage improvement by clock synchronized duplicated systems with optimal time diversity" FAULT-TOLERANT COMPUTING, 1998. DIGEST OF PAPERS. TWENTY-EIGHTH ANNUAL INTERNATIONAL SYMPOSIUM ON MUNICH, GERMANY 23-25 JUNE 1998, LOS ALAMITOS, CA, USA, IEEE COMPUT. SOC, US, 23. Juni 1998 (1998-06-23), Seiten 196-200, XP010291315 ISBN: 0-8186-8470-4 Seite 197, linke Spalte, Zeile 1 - Seite 199, linke Spalte, Zeile 21 -----	1-32

INTERNATIONALER RECHERCHENBERICHT

Angaben zu Veröffentlichungen, die zur selben Patentfamilie gehören

Internationales Aktenzeichen

PCT/EP2005/055537

Im Recherchenbericht angeführtes Patentdokument	Datum der Veröffentlichung	Mitglied(er) der Patentfamilie	Datum der Veröffentlichung
US 6615366	B1	02-09-2003	KEINE
DE 1269827	B	06-06-1968	KEINE
DE 4104114	A1	13-08-1992	KEINE
WO 2005003962	A	13-01-2005	KEINE
US 6519710	B1	11-02-2003	CN 1248748 A 29-03-2000
		DE 69908717 D1	17-07-2003
		DE 69908717 T2	11-12-2003
		EP 0980040 A2	16-02-2000
		ES 2199527 T3	16-02-2004
		GB 2340627 A	23-02-2000
		HK 1024067 A1	01-04-2005
		RU 99118019 A	27-08-2001