

(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
27. März 2003 (27.03.2003)

PCT

(10) Internationale Veröffentlichungsnummer
WO 03/025748 A2

(51) Internationale Patentklassifikation⁷: G06F 11/00

(72) Erfinder; und

(21) Internationales Aktenzeichen: PCT/DE02/03051

(75) Erfinder/Anmelder (nur für US): STEINERT, Alexander [DE/DE]; Schellingstrasse 36, 72072 Tuebingen (DE).

(22) Internationales Anmeldedatum:
21. August 2002 (21.08.2002)

(81) Bestimmungsstaaten (national): JP, KR, US.

(25) Einreichungssprache: Deutsch

(84) Bestimmungsstaaten (regional): europäisches Patent (AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE, SK, TR).

(26) Veröffentlichungssprache: Deutsch

Veröffentlicht:

(30) Angaben zur Priorität:
101 43 142.2 4. September 2001 (04.09.2001) DE

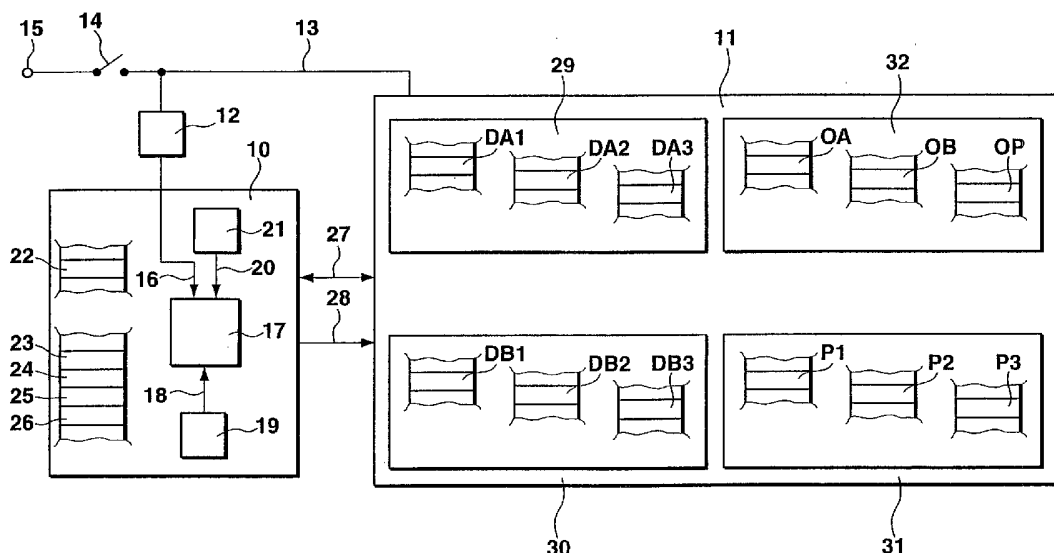
— ohne internationalen Recherchenbericht und erneut zu veröffentlichen nach Erhalt des Berichts

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

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.

(54) Title: METHOD FOR OPERATING A CIRCUIT ARRANGEMENT CONTAINING A MICROCONTROLLER AND AN EEPROM

(54) Bezeichnung: VERFAHREN ZUM BETREIBEN EINER SCHALTUNGSANORDNUNG, DIE EINEN MIKROCONTROLLER UND EIN EEPROM ENTHÄLT



(57) Abstract: The invention relates to a method for operating a circuit arrangement containing a microcontroller (10) and an EEPROM (11). A first memory area (39) having at least one data set memory (DA1, DA2, DA3) and at least one second memory area (30) also having at least one data set memory (DB1, DB2, DB3) are provided in the EEPROM (11). A pointer (P1, P2, P3) located in a third memory area (31) refers to the respectively valid memory area (29, 30). In one step (53), the microcontroller (10) stores a data set in the invalid memory area (29, 30) and, in a subsequent step (54), changes the pointer (P1, P2, P3) so that the invalid memory area (29, 30) becomes the valid memory area (30, 29). Both steps (53, 54) are cyclically repeated. In the event of a fault during a storage process, a correct data set is available at all times in the memory area (29, 30) indicated as being valid.

[Fortsetzung auf der nächsten Seite]

WO 03/025748 A2



(57) Zusammenfassung: Es wird ein Verfahren zum Betreiben einer Schaltungsanordnung vorgeschlagen, die einen Mikrocontroller (10) und ein EEPROM (11) enthält. Im EEPROM (11) wird ein erster Speicherbereich (29) mit zumindest einem Datensatzspeicher (DA1, DA2, DA3) und zumindest ein zweiter Speicherbereich (30) mit zumindest einem Datensatzspeicher (DB1, DB2, DB3) vorgesehen. Ein in einem dritten Speicherbereich (31) angeordneter Pointer (P1, P2, P3) verweist auf den jeweils gültigen Speicherbereich (29, 30). Der Mikrocontroller (10) speichert in einem Schritt (53) einen Datensatz in den ungültigen Speicherbereich (29, 30) und ändert in einem nachfolgenden Schritt (54) den Pointer (P1, P2, P3) derart, dass der ungültige zum gültigen Speicherbereich (29, 30) wird. Die beiden Schritte (53, 54) werden zyklisch wiederholt. Bei einem auftretenden Fehler während eines Speichervorgangs steht in dem als gültig gekennzeichneten Speicherbereich (29, 30) stets ein korrekter Datensatz zur Verfügung.

Verfahren zum Betreiben einer Schaltungsanordnung, die einen Mikrocontroller und ein EEPROM enthält

Stand der Technik

Die Erfindung geht aus von einem Verfahren zum Betreiben einer Schaltungsanordnung, die einen Mikrocontroller und ein EEPROM enthält, nach der Gattung des unabhängigen Anspruchs. Aus der DE-A 197 16 520 ist ein gattungsgemäßes Verfahren bekannt geworden, bei dem Betriebsgrößen von Elektromotoren erfasst werden. Zur Speicherung vorgesehen sind insbesondere die Betriebsstunden. Als Langzeitspeicher kann ein PROM, ein EPROM oder ein EEPROM vorgesehen sein.

Bei einem PROM werden die abzulegenden Datensätze durch irreversibles Programmieren der PROM-Speicherzellen gespeichert. Ein EPROM kann durch das Löschen des Speicherinhalts am Ende eines Nutzungszyklus weiter verwendet werden. Da beide Speichertypen die Speicherung eines Datensatzes in derselben Speicherzelle während eines Nutzungszyklus nur einmal zulassen, sind für einen Betriebsstundenzähler zahlreiche Speicherzellen zu reservieren.

Die flexibelste Lösung sieht ein EEPROM vor, dessen Speicherzellen elektrisch beschreib- und löschbar sind. Den Vorteilen des Einsatzes eines EEPROMs stehen aber auch Nachteile gegenüber. Das EEPROM kann vorübergehend in einen undefinierten Betriebszustand gelangen, wenn während eines Spei-

chervorgangs die Betriebsspannung zusammenbricht. Die Folge ist ein fehlerhaft hinterlegter Datensatz.

Der Erfindung liegt die Aufgabe zugrunde, ein Verfahren zum Betreiben einer Schaltungsanordnung anzugeben, die einen Mikrocontroller und ein EEPROM enthält, bei dem das EEPROM zuverlässige Datensätze enthält.

Die Aufgabe wird durch die im unabhängigen Anspruch angegebenen Merkmale gelöst.

Vorteile der Erfindung

Das erfindungsgemäße Verfahren sieht zunächst vor, dass im EEPROM ein erster und zumindest ein zweiter Speicherbereich zur Speicherung zumindest eines Datensatzes vorgesehen werden. Weiterhin wird ein dritter Speicherbereich angelegt, der einen Pointer enthält, welcher auf den gültigen Speicherbereich zeigt. Der Mikrocontroller speichert in einem ersten Schritt einen Datensatz in den ungültigen Speicherbereich und ändert in einem zweiten Schritt den Pointer, so dass der ungültige zum gültigen Speicherbereich wird. Die beiden Schritte werden zyklisch wiederholt.

Ein wesentlicher Vorteil des erfindungsgemäßen Verfahrens liegt darin, dass in jedem Betriebszustand ein gültiger Datensatz zumindest in einem der wenigstens zwei Speicherbereichen abgelegt ist. Eine Störung im Betriebsablauf, insbesondere ein Abschalten der Spannungsversorgung während eines Schreibvorgangs, die zur Folge haben kann, dass ein Speichervorgang eines Datensatzes in einen Speicherbereich fehlschlägt, hat keine Auswirkungen auf den korrekt abgelegten Datensatz im anderen Speicherbereich. Es kann daher jederzeit auf den zuletzt abgespeicherten Datensatz im anderen Speicherbereich zurückgegriffen werden.

Eine besonders vorteilhafte Weiterbildung sieht die Einbindung einer Resetanordnung in das erfindungsgemäße Verfahren vor. Eine erkannte Störung im Programmablauf oder eine Inbetriebnahme der Schaltungsanordnung löst einen Reset aus. Erfindungsgemäß ist vorgesehen, dass der Mikrocontroller nach einem Reset in einem ersten Startschritt den Pointer ausliest und in einem zweiten Startschritt den Datensatz aus dem gültigen Speicherbereich ausliest, auf den der Pointer zeigt.

Das erfindungsgemäße Verfahren eignet sich insbesondere für Schaltungsanordnungen, deren Betriebsspannung von einem Benutzer jederzeit ein- bzw. abgeschaltet werden kann. Da der Pointer nach einem Reset stets auf einen gespeicherten Datensatz zeigt, der in einem normalen Betriebszustand abgespeichert wurde, kann den weiteren Verfahrensschritten immer ein korrekter Datensatz zugrunde gelegt werden.

Eine vorteilhafte Ausgestaltung sieht vor, dass der Pointer als Bitpointer realisiert ist. Der Bitpointer zeigt anhand des niederwertigsten Bits eines ggf. längeren Datensatzes, beispielsweise eines Bytes, auf den jeweils gültigen Speicherbereich. Die Maßnahme weist den Vorteil auf, dass ein während des Änderns des Speicherinhalts des Pointers auftretender Fehler, beispielsweise durch einen Spannungseinbruch, lediglich dazu führt, dass der Pointer anstelle auf den einen korrekten Datensatz in einem Speicherbereich auf den ebenfalls korrekten Datensatz im anderen Speicherbereich zeigt.

Aufgrund der sequentiellen Arbeitsweise des Mikrocontrollers kann ausgeschlossen werden, dass mehr als ein Speichervorgang fehlerhaft durchgeführt wird. Ein im zumindest vorangegangenen Zyklus hinterlegter korrekter Datensatz steht daher in jedem Fall nach einem Reset zur Verfügung.

Eine andere vorteilhafte Maßnahme sieht einen Timer vor, der die zyklische Wiederholung des ersten und zweiten Schritts steuert. Der Timer kann im Mikrocontroller enthalten sein. In Abhängigkeit von Taktzyklen, die vorzugsweise von einem Quarzgenerator vorgegebenen werden, veranlasst der Timer den ersten Schritt, in welchem ein Datensatz in den derzeit noch ungültigen Speicherbereich gespeichert wird. Im zweiten Schritt wird der Pointer geändert, so dass der Pointer vom ungültigen auf den nunmehr gültigen Speicherbereich zeigt. Die Maßnahme weist den Vorteil auf, dass unabhängig vom Programmablauf im Mikrocontroller nach vorgegebenen Zeiten das erfindungsgemäße Verfahren mit den beiden Schritten stattfindet.

Eine andere vorteilhafte Ausgestaltung sieht vor, dass die zu speichernden Datensätze von Zyklus zu Zyklus inkrementiert oder dekrementiert werden. Mit dieser Maßnahme ist auf einfachste Weise ein Betriebszeitzähler realisierbar. Das Maß für die Betriebszeit kann mit der Taktfrequenz in Verbindung mit dem Timer vorgegeben werden. Das erfindungsgemäße Verfahren stellt sicher, dass bei einem fehlerhaften Speichervorgang nicht zum Verlust der Betriebszeit führt, da bei einem Reset auf den zuvor im anderen Speicherbereich gespeicherten korrekten Datensatz zurückgegriffen wird.

Das erfindungsgemäße Verfahren eignet sich insbesondere zur Verwendung für Vorrichtungen in einem Kraftfahrzeug. Aufgrund der nur begrenzt zur Verfügung stehenden elektrischen Energie im abgeschalteten Zustand des Kraftfahrzeugs ist es vorteilhaft, wenn die Schaltungsanordnung vollständig abgeschaltet werden kann. Ein ggf. auftretender Fehler beim Speichern eines Datensatzes während des Abschaltvorgangs hat dann keine weiteren Auswirkungen, da bei der Wiederinbetriebnahme auf den zuletzt korrekt gespeicherten Datensatz zurückgegriffen wird. Eine Einsatzmöglichkeit in einem Kraftfahrzeug ist bei einem Luftgütesensor gegeben, der die

Qualität der Aussenluft erfasst und der Steuersignale an eine Klimaanlage abgibt. Anhand des Betriebszeitzählers können sowohl Kurzzeit- als auch Langzeitkorrekturen des Sensorsignals vorgenommen werden.

Weitere vorteilhafte Ausgestaltungen des erfindungsgemäßen Verfahrens ergeben sich aus weiteren abhängigen Ansprüchen und aus der folgenden Beschreibung.

Figur 1 zeigt ein Blockschaltbild einer Schaltungsanordnung, in der ein erfindungsgemäßes Verfahren gemäß Figur 2 abläuft.

Die Schaltungsanordnung enthält einen Mikrocontroller 10, ein EEPROM 11 und eine Resetanordnung 12. Sowohl die Resetanordnung 12 als auch das EEPROM 11 sind mit einer Stromversorgungsleitung 13 verbunden, die über einen Schalter 14 mit einer Energiequelle 15 verbindbar ist.

Die Resetanordnung 12 gibt ein Resetsignal 16 an einen Prozessorkern 17 ab. Dem Prozessorkern 17 werden weiterhin zugeführt ein Taktsignal 18, das von einem Taktgenerator 19 bereitgestellt ist, sowie ein Timersignal 20, das ein Timer 21 bereitstellt.

Der Mikrocontroller 10 enthält einen Resetspeicher 22, einen Datensatzspeicher 23, einen Pointerspeicher 24, einen Datensatzoffsetspeicher 25 sowie einen Pointeroffsetspeicher 26.

Der Mikrocontroller 10 kommuniziert mit dem EEPROM 11 über einen bidirektionalen Datenbus 27 sowie über einen Adressbus 28.

Das EEPROM 11 enthält einen ersten, zweiten, dritten und vierten Speicherbereich 29, 30, 31, 32. Der erste Speicherbereich 29 enthält einen ersten, zweiten und dritten Daten-

satzspeicher DA1, DA2, DA3. Der zweite Speicherbereich 30 enthält ebenfalls einen ersten, zweiten und dritten Datensatzspeicher DB1, DB2, DB3. Der dritte Speicherbereich 31 enthält einen ersten, zweiten und dritten Pointer P1, P2, P3. Der vierte Speicherbereich 32 enthält einen ersten und zweiten Datenoffsetspeicher OA, OB sowie einen Pointeroffsetspeicher OP.

Figur 2 zeigt ein erfindungsgemäßes Verfahren, das nach einem Start S in einem ersten Startschritt 50 einen Resetvorgang vorsieht. In einem zweiten Startschritt 51 ist ein Auslesen des Pointers P1, P2, P3 vorgesehen. In einem dritten Startschritt 52 wird ein Datensatz entweder aus dem ersten oder zweiten Speicherbereich 29, 30 ausgelesen, je nachdem auf welchen Speicherbereich 29, 30 der Pointer P1, P2, P3 als dem gültigen Speicherbereich 29, 30 zeigt.

In einem ersten Schritt 53 wird ein Datensatz in einen Datensatzspeicher DA1, DA2, DA3 bzw. DB1, DB2, DB3 gespeichert, der in dem als ungültig gekennzeichneten ersten bzw. zweiten Speicherbereich 29, 30 liegt. In einem zweiten Schritt 54 wird der Pointer P1, P2, P3 geändert, so dass er auf den nunmehr gültigen ersten oder zweiten Speicherbereich 29, 30 zeigt.

Das erfindungsgemäße Verfahren zum Betreiben der Schaltungsanordnung arbeitet folgendermaßen:

Zunächst wird die Stromversorgungsleitung 13 über den Schalter 14 mit der Energiequelle 15, beispielsweise eine Batterie, verbunden. Die Resetanordnung 12, die im Mikrocontroller 10 enthalten sein kann, stellt das Resetsignal 16 bereit. Das Resetsignal 16 veranlasst den Mikrocontroller 10 zu einem Neustart eines Ablaufprogramms. Die für einen Reset benötigte Information, im allgemeinen eine Startadresse, ist im Resetspeicher 23 gespeichert. Der Resetspeicher 23 ist

vorzugsweise in einem ROM enthalten, dessen Inhalt herstellerseitig festgelegt ist. Das ROM kann im Mikrocontroller 10 enthalten sein.

Nach dem Start S, der durch Betätigen des Schalters 14 eingeleitet wird, wird der erste Startschritt 50 durchgeführt, der dem Resetvorgang entspricht.

Der Mikrocontroller 10 steht über den Datenbus 27 und den Adressbus 28 mit dem EEPROM 11 in Verbindung. Der Begriff "EEPROM" (Electrical Erasable and Programmable Memory) wird hier verwendet für einen Speichertyp, der nach dem Abschalten der Betriebsspannung seinen Inhalt nicht verliert und dessen Speicherzellen mehrfach beschreibbar sind. Das EEPROM kann im Mikrocontroller 10 enthalten sein, der in diesem Fall bereits als Mikroprozessor bezeichnet werden kann.

Im dritten Speicherbereich 31 des EEPROMs 11 ist der Pointer P1, P2, P3 untergebracht. Der Pointer P1, P2, P3 ist ein Datensatz, aus dem hervorgeht, ob der erste oder zweite Speicherbereich 29, 30 des EEPROMs 11 gültig ist bzw. ungültig ist. Vorzugsweise wird der Pointer P1, P2, P3 als Bitpointer realisiert, so dass nur ein Bit benötigt wird, das den Zustand 0 oder 1 aufweist. Bei der Realisierung des Bitpointers mit beispielsweise einem Byte, wird die Information vorzugsweise im niederwertigsten Bit kodiert.

Anstelle eines Pointers sind im gezeigten Ausführungsbeispiel der erste, zweite und dritte Pointer P1, P2, P3 gezeigt. Welcher der drei Pointer P1, P2, P3 die aktuelle Information enthält, ergibt sich aus dem im Pointeroffsetspeicher OP gespeicherten Datensatz, der im vierten Speicherbereich 32 des EEPROMs 11 untergebracht ist. Der Mikrocontroller 10 erreicht den geltenden Pointer P1, P2, P3 durch lesen der im Pointeroffsetspeicher OP hinterlegten Offsetadresse, die vorzugsweise zu einer dem Mikrocontroller 10 bekannten

Basisadresse zu addieren ist. Anstelle der drei gezeigten Pointer P1, P2, P3 können noch weitere Pointer enthalten sein, die sich lediglich durch die Adresse im EEPROM 11 unterscheiden.

Der wesentliche Grund für das Vorsehen der unterschiedlichen Pointer P1, P2, P3 liegt in der aus technischen Gründen begrenzten Anzahl, Daten in das EEPROM 11 in dieselbe Speicherstelle schreiben zu können. Deshalb wird der Offset im Pointeroffsetspeicher OP nach einer vorgegebenen Anzahl von Speichervorgängen geändert auf einen anderen Wert, der auf den nächsten Pointer P1, P2, P3 führt. Der Pointer P1, P2, P3 "wandert" gewissermaßen durch das EEPROM 11. Die Information darüber, wieviel Speichervorgänge seit Inbetriebnahme des EEPROMs 11 aufgetreten sind, kann in einer nicht näher gezeigten Speicherzelle des EEPROMs 11 hinterlegt sein.

Vorteilhafterweise kann die Information über die Anzahl der bereits erfolgten Speichervorgängen aus dem im Datensatzspeicher DA1, DA2, DA3 bzw. DB1, DB2, DB3 gespeicherten Datensatz ermittelt werden. Diese Ausgestaltung ist beispielsweise im Rahmen eines Betriebszeitzählers möglich, der folgendermaßen realisierbar ist:

Der Zeittakt wird vom Mikrocontroller 10 mittels des Taktgenerators 19, der vorzugsweise ein Quarzgenerator ist, und mittels des Timers 21 festgelegt. Ein Zeittakt von beispielsweise 10 Sekunden ermöglicht mit einem Zähler, der drei Byte binäre Speicherzellen aufweist, eine maximale Zeitvorgabe von etwa 46603 Stunden. Der Betriebszeitzähler wird vorzugsweise im Datensatzspeicher DA1, DA2, DA3 bzw. DB1, DB2, DB3 des EEPROMs 11 untergebracht. Ein Datenverlust, der durch einen fehlerhaften Speichervorgang auftreten kann, ist unter allen Umständen zu vermeiden. Erfindungsgemäß ist deshalb die Aufteilung auf den ersten und den wenigstens einen weiteren Speicherbereich 29, 30 vorgesehen.

Der erste Speicherbereich enthält den ersten, zweiten und dritten Datensatzspeicher DA1, DA2, DA3 und der zweite Speicherbereich enthält den ersten, zweiten und dritten Datensatzspeicher DB1, DB2, DB3.

Der Grund dafür, dass anstelle jeweils eines Datensatzspeichers mehrere Datensatzspeicher DA1, DA2, DA3 bzw. DB1, DB2, DB3 vorgesehen sind, liegt wieder darin, dass die maximale Anzahl von Schreibvorgängen in dieselbe Speicherzelle des EEPROMS 11 begrenzt ist. Nach einer vorgegebenen Anzahl von Speichervorgängen kann deshalb mit dieser Maßnahme auf einen anderen Datensatzspeicher gewechselt werden werden. Ebenso wie beim Pointer P1, P2, P3 ist auch hier, vorzugsweise ausgehend von einer Basisadresse, ein Offset zu addieren, der vorzugsweise in den Datenoffsetspeichern OA, OB hinterlegt ist. Der erste Datenoffsetspeicher OA enthält den Offset für die Datensatzspeicher DA1, DA2, DA3 im ersten Speicherbereich 29 und der zweite Datenoffsetspeicher OB enthält den Offset für die Datensatzspeicher DB1, DB2, DB3 im zweiten Speicherbereich 30. Die Datensatzspeicher DA1, DA2, DA3 bzw. DB1, DB2, DB3 "wandern" gewissermaßen durch das EEPROM 11.

Zur weiteren Erläuterung wird lediglich Bezug genommen auf den ersten Datensatzspeicher DA1 im ersten Speicherbereich 29 und den ersten Datensatzspeicher DB1 im zweiten Speicherbereich 30. Nach Ablauf der vom Timer 21 vorgegebenen Zeit addiert der Prozessorkern 17 zu einem die Zeit widerspiegelnden Datensatz eine Einheit, beispielsweise eine Eins und speichert den derart festgelegten neuen Datensatz in entweder den ersten Datensatzspeicher DA1 oder den ersten Datensatzspeicher DB1 in Abhängigkeit davon, welcher Speicherbereich 29, 30 vom Pointer P1 als ungültig gekennzeichnet ist. Hinsichtlich der Pointer P1, P2, P3 wird im Folgenden ebenfalls ausschließlich Bezug genommen auf den Pointer P1. Den die Betriebszeit widerspiegelnde Datensatz kann der Prozessorkern 17 beispielsweise aus dem Datensatzspeicher DA1 bzw.

DB1 lesen, der sich in dem vom Pointer P1 als gültig gekennzeichneten Speicherbereich 29, 30 befindet. Vorzugsweise ist im Mikrocontroller 10 für diesen Datensatz ein RAM - Speicher 23 angelegt. Ebenso sind vorzugsweise der Pointerspeicher 24, der Datensatzoffsetspeicher 25 sowie der Pointeroffsetspeicher 26 als RAM - Speicher vorhanden.

Dem erfindungsgemäß vorgesehenen ersten Schritt 53, in dem ein Datensatz in den ersten Datensatzspeicher DA1, DB1 des als ungültig gekennzeichneten Speicherbereichs 29, 30 gespeichert wird, folgt ein erfindungsgemäß vorgesehener zweiter Schritt 54, in dem der Pointer P1 geändert wird auf einen Wert, der den anderen Speicherbereich 29, 30, in den zuletzt ein Datensatz gespeichert wurde, zum gültigen Speicherbereich 29, 30 erklärt.

Anstelle der im Ausführungsbeispiel gezeigten zwei Speicherbereichen 29, 30 können weitere Speicherbereiche vorhanden sein. Das Ausführungsbeispiel mit lediglich zwei Speicherbereichen 29, 30 weist den Vorteil auf, dass der Pointer P1, P2, P3 als Bitpointer realisierbar ist, der im einfachsten Fall lediglich ein Speicherbit aufweisen muss, das den Wert 0 oder 1 annimmt.

Im weiteren Verlauf werden der erste und zweite Schritt 53, 54 zyklisch abgearbeitet. Sollte nun zu irgendeinem Zeitpunkt, beispielsweise durch Öffnen des Schalters 14, ein fehlerhafter Wert im Datensatzspeicher DA1, DB1 hinterlegt worden sein, so steht in jedem Fall der im zurückliegenden Zyklus abgelegte Datensatz im entsprechend anderen Datensatzspeicher DA1, DB1 noch zur Verfügung, weil der Bitpointer P1 noch auf den alten gültigen Datensatzspeicher DA1, DB1 zeigt.

Falls es zu einem fehlerhaften Speichervorgang im Pointer P1 kommen sollte, so hat dies keine Auswirkungen auf die Daten

in den Datensatzspeichern DA1, DB1. Im Falle eines Betriebszeitzählers tritt in 50% der Fehlerfälle ein Zeitfehler von einem Zyklus auf. Im beschriebenen Ausführungsbeispiel würde der Fehler nur 10 Sekunden betragen.

Nach jedem Abschalten der Betriebsspannung auf der Stromversorgungsleitung 13 entweder durch den Schalter 14 oder durch ein anderes Ereignis wird die Resetschaltung 12 aktiv und erzeugt das Resetsignal 16. Bei einem Initialisieren des Mikrocontrollers 10 wird im bereits beschriebenen ersten Startschritt der Pointer P1 ausgelesen und im zweiten Startschritt 51 der Datensatz aus dem ersten Datensatzspeicher DA1, DB1 des gültigen Speicherbereichs 29, 30 ausgelesen und dem weiteren Programmlauf zugrunde gelegt. Ggf. ist zusätzlich der Pointeroffsetspeicher OP auszulesen, sofern mehrere Pointer P1, P2, P3 aufgrund der begrenzten Schreibzyklenzahl des EEPROMs 11 vorgesehen sind. Ebenso ist ggf. zusätzlich der Offset für den Datensatzspeicher aus dem ersten oder zweiten Datenoffsetspeicher OA, OB auszulesen, sofern anstelle lediglich eines Datensatzspeichers DA1, DB1 im ersten bzw. zweiten Speicherbereich 29, 30 aufgrund der begrenzten Speicherzyklenzahl des EEPROMs 11 weitere Datensatzspeicher DA2, DA3 im ersten Speicherbereich 29 und entsprechend weitere Datensatzspeicher DB2, DB3 im zweiten Speicherbereich 30 vorhanden sind. Den gültigen Datenoffsetspeicher OA, OB ermittelt der Mikrocontroller 10 anhand des Inhalts des Pointers P1, P2, P3.

Das erfindungsgemäße Verfahren ist insbesondere geeignet zur Verwendung für Vorrichtungen, die in einem Kraftfahrzeug eingebaut sind. In dieser Verwendung besteht zumindest im abgestellten Zustand des Kraftfahrzeugs nur eine begrenzte Energiemenge zum Betreiben der Schaltungsanordnung zur Verfügung, so dass eine Möglichkeit zur vollständigen Abschaltung der Schaltungsanordnung mittels des Schalters 14 zweckmäßig ist. Das erfindungsgemäße Verfahren ermöglicht trotz

der vollständigen Abschaltungsmöglichkeit und den dadurch möglicherweise auftretenden Datenfehler bei einem Speichervorgang im EEPROM 11 während des Abschaltvorgangs dennoch ein sicheres Betreiben der Schaltungsanordnung. Eine bevorzugte Verwendung in einem Kraftfahrzeug ist bei einem Luftgütesensor gegeben, der einen Betriebszeitähler zur Korrektur der Signale hinsichtlich Kurzzeit und hinsichtlich Langzeitänderungen verwenden kann.

Ansprüche

1. Verfahren zum Betreiben einer Schaltungsanordnung, die einen Mikrocontroller (10) und ein EEPROM (11) enthält, dadurch gekennzeichnet, dass im EEPROM (11) ein erster Speicherbereich (29) mit einem Datensatzspeicher (DA1, DA2, DA3) und zumindest ein zweiter Speicherbereich (30) mit einem Datensatzspeicher (DB1, DB2, DB3) vorgesehen werden, dass ein dritter Speicherbereich (31) vorgesehen wird, der zumindest einen Pointer (P1, P2, P3) enthält, welcher auf den gültigen Speicherbereich (29, 30) zeigt, dass der Mikrocontroller (10) in einem Schritt (53) einen Datensatz in den Datensatzspeicher (DA1, DA2, DA3 bzw. DB1, DB2, DB3) des ungültigen Speicherbereichs (29, 30) speichert und in einem nachfolgenden Schritt (54) den Pointer (P1, P2, P3) ändert, so dass der ungültige zum gültigen Speicherbereich (29, 30) wird und dass die beiden Schritte (53, 54) zyklisch wiederholt werden.
2. Verfahren nach Anspruch 1, dadurch gekennzeichnet, dass der Mikrocontroller (10) vor den beiden Schritten (53, 54) in einem ersten Startschritt (50) nach einem von der Resetanordnung (12) ausgelösten Reset (16) den Pointer (P1, P2, P3) ausliest und in einem zweiten Startschritt (51) den Datensatz aus dem gültigen Speicherbereich (29, 30) ausliest.

3. Verfahren nach Anspruch 1 oder 2, dadurch gekennzeichnet, dass der Pointer (P1, P2, P3) als Bitpointer realisiert ist.
4. Verfahren nach Anspruch 1 und 3, dadurch gekennzeichnet, dass im zweiten Schritt (54) eine Inkrementierung des Bitpointers (P1, P2, P3) vorgesehen ist.
5. Verfahren nach einem der vorhergehenden Ansprüche, dadurch gekennzeichnet, dass ein Timer (21) vorgesehen ist, der die beiden Schritte (53, 54) zeitlich festlegt.
6. Verfahren nach einem der vorhergehenden Ansprüche, dadurch gekennzeichnet, dass der Datensatz zwischen den beiden Schritten (53, 54) inkrementiert wird.
7. Verfahren nach einem der vorhergehenden Ansprüche, dadurch gekennzeichnet, dass der Datensatzspeicher (DA1, DA2, DA3) im ersten Speicherbereich (29) und der Datensatzspeicher (DB1, DB2, DB3) im zweiten Speicherbereich (30) jeweils einen Speicherplatz von drei Bytes aufweist.
8. Verfahren nach einem der vorhergehenden Ansprüche, dadurch gekennzeichnet, dass der Datensatz einem Betriebszeitzähler entspricht.
9. Verwendung des Verfahrens nach einem der vorhergehenden Ansprüche für Vorrichtungen in einem Kraftfahrzeug.
10. Verwendung nach Anspruch 9 in einem Luftgütesensor.

1 / 2

Fig. 1

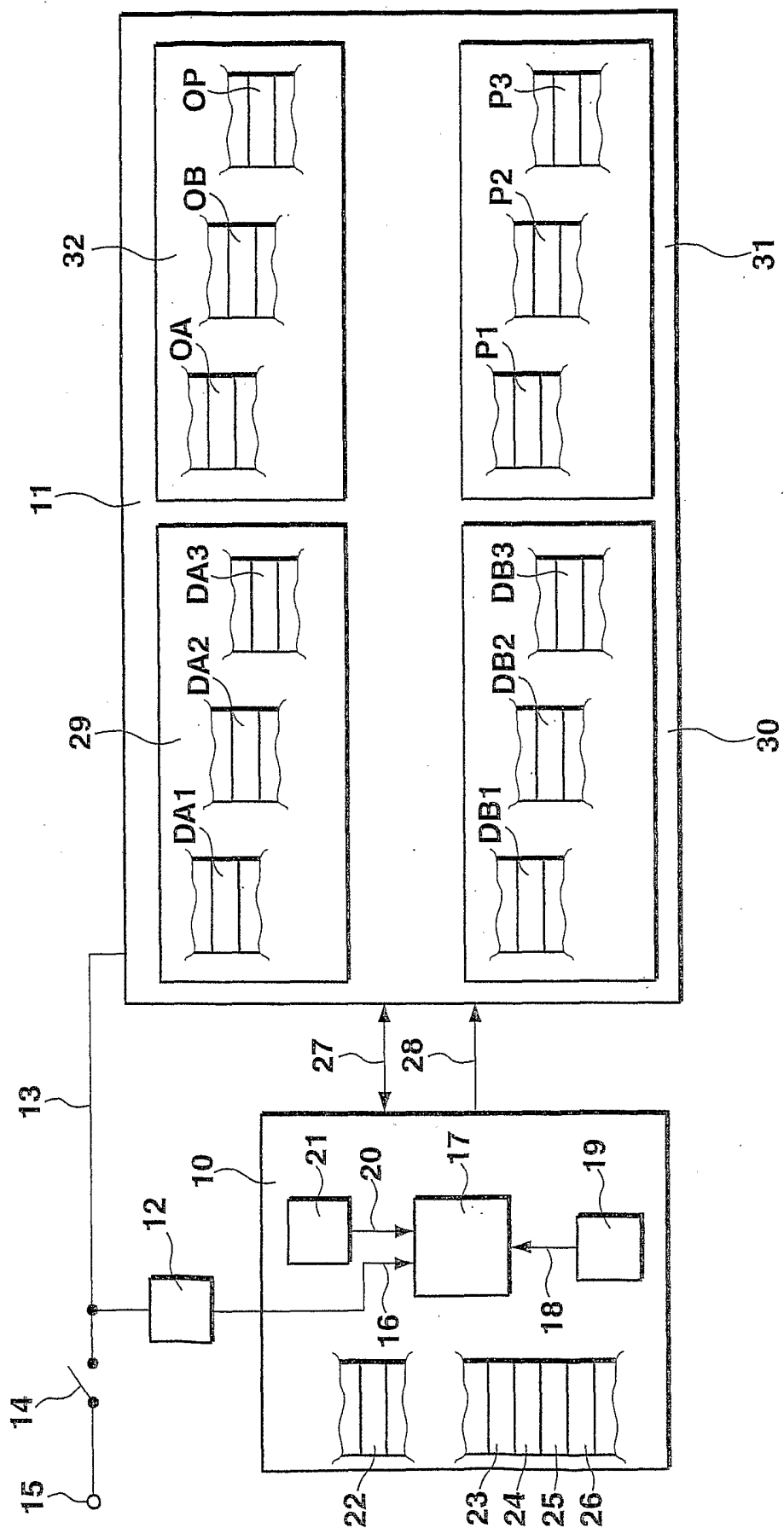


Fig. 2

