

DOCUMENT MADE AVAILABLE UNDER THE PATENT COOPERATION TREATY (PCT)

International application number:	PCT/EP2017/072617
International filing date:	08 September 2017 (08.09.2017)
Document type:	Certified copy of priority document
Document details:	Country/Office: DE
	Number: 10 2016 117 056.9
	Filing date: 12 September 2016 (12.09.2016)
Date of receipt at the International Bureau:	28 September 2017 (28.09.2017)

Remark: Priority document submitted or transmitted to the International Bureau in compliance with Rule 17.1(a),(b) or (b-bis)

BUNDESREPUBLIK DEUTSCHLAND



Prioritätsbescheinigung DE 10 2016 117 056.9 über die Einreichung einer Patentanmeldung

Aktenzeichen: 10 2016 117 056.9
Anmeldetag: 12. September 2016
Anmelder/Inhaber: Hella KGaA Hueck & Co., 59557 Lippstadt, DE
Bezeichnung: Verfahren zur sicheren Bereitstellung von
gespeicherten Informationen bei einer
Elektronikkomponente
IPC: G06F 21/64

Die angehefteten Stücke sind eine richtige und genaue Wiedergabe der Teile der am 12. September 2016 eingereichten elektronischen Dokumente dieser Patentanmeldung unabhängig von gegebenenfalls durch das Druckverfahren bedingten Farbabweichungen.

München, den 4. September 2017
Deutsches Patent- und Markenamt
Die Präsidentin
Im Auftrag



Kahle

Verfahren zur sicheren Bereitstellung von gespeicherten Informationen bei einer Elektronikkomponente

Beschreibung

Die vorliegende Erfindung betrifft ein Verfahren zur sicheren Bereitstellung von gespeicherten Informationen bei einer Elektronikkomponente, insbesondere zur Gewährleistung der Datenintegrität. Ferner bezieht sich die Erfindung auf ein Computerprogrammprodukt zur Durchführung des Verfahrens.

Es ist aus dem Stand der Technik bekannt, dass verschiedene Methoden zur Gewährleistung der Datenintegrität bei Schreib- und Lesevorgängen eines Datenspeichers eingesetzt werden. Bei vielen Elektronikkomponenten, wie den Steuergeräten eines Fahrzeuges, insbesondere eines Gleichspannungswandlers, kommen zur persistenten Speicherung von Daten nicht-flüchtige Datenspeicher, wie Flash-Speicher oder dergleichen, zum Einsatz. Aufgrund der komplexen Lese- und/oder Schreibvorgänge bei solchen Datenspeichern wird häufig der Datenzugriff und/oder das Schreiben von Daten durch eine Treibereinheit, insbesondere durch einen sogenannten Flash- bzw. EEPROM (Electrically Erasable Programmable Read-Only Memory)-Emulator, ermöglicht. Hierbei können von einer Applikation der Treibereinheit die Daten zum Speichern übergeben werden, und/oder die Treibereinheit übergibt ausgelesene Daten an die Applikation. Entsprechend erfolgt der Datenzugriff oft nicht direkt am Flash-Speicher, sondern indirekt über die Treibereinheit.

Grundsätzlich können dabei auch durch die Treibereinheit Sicherheitsmechanismen zur Gewährleistung der Datenintegrität bereitgestellt werden. Diese sind allerdings oft nicht ausreichend, sodass weiterhin ein Risiko dafür besteht, dass fehlerhafte oder manipulierte Daten ausgelesen werden. Insbesondere für sicherheitskritische Anwendungen, z. B. bei einer Ansteuerung von Fahrzeugkomponenten in Abhängigkeit von den ausgelesenen Daten, muss jedoch eine erhöhte Datenintegrität gewährleistet werden.

Es ist somit oft ein Nachteil, dass die Datenablage bei sicherheitsrelevanten Elektronikkomponenten nur unzureichend gesichert ist, und insbesondere die Datenintegrität nicht ausreichend gewährleistet ist. Insbesondere sind die Daten nur unzureichend vor Manipulationen und unberechtigten Fremdzugriff geschützt.

Es ist daher eine Aufgabe der vorliegenden Erfindung, die voranstehend beschriebenen Nachteile zumindest teilweise zu beheben. Insbesondere ist es eine Aufgabe der vorliegenden Erfindung, eine verbesserte Sicherheit bei sicherheitsrelevanten Elektronikkomponenten zu ermöglichen, und insbesondere die Sicherheit bei der Datenablage zu verbessern.

Die voranstehende Aufgabe wird gelöst durch ein Verfahren mit den Merkmalen des Anspruchs 1 und durch ein Computerprogrammprodukt mit den Merkmalen des Anspruchs 10. Weitere Merkmale und Details der Erfindung ergeben sich aus den jeweiligen Unteransprüchen, der Beschreibung und den Zeichnungen. Dabei gelten Merkmale und Details, die im Zusammenhang mit dem erfindungsgemäßen Verfahren beschrieben sind, selbstverständlich auch im Zusammenhang mit dem erfindungsgemäßen Computerprogrammprodukt, und jeweils umgekehrt, so dass bzgl. der Offenbarung zu den einzelnen Erfindungsaspekten stets wechselseitig Bezug genommen wird bzw. werden kann.

Die Aufgabe wird insbesondere gelöst durch ein Verfahren zur sicheren Bereitstellung von gespeicherten Informationen bei einer Elektronikkomponente mit einem (insbesondere nicht-flüchtigen) Datenspeicher, und insbesondere zur Gewährleistung einer Datenintegrität bei der Elektronikkomponente.

Vorzugsweise ist die Elektronikkomponente als ein Controller und/oder als ein Steuergerät und/oder als ein Gleichspannungswandler, insbesondere eines Fahrzeuges und/oder Kraftfahrzeuges und/oder Elektrofahrzeuges, ausgeführt. Insbesondere dient der Gleichspannungswandler zur Umwandlung einer Gleichspannung und/oder zur Verbindung mit einem weiteren Gleichspannungsnetz

mit einem unterschiedlichen Potenzial. Bevorzugt dient die Elektronikkomponente zur Durchführung von sicherheitskritischen Aufgaben, insbesondere zur Ansteuerung sicherheitskritischer Funktionen bei einem Fahrzeug, bspw. einer Fahrzeugbeleuchtung und/oder eines Bremssystems des Fahrzeuges. Insbesondere erfolgt die Ansteuerung und/oder die Durchführung in Abhängigkeit von sicherheitsrelevanten Daten, insbesondere Nutzdateninformationen.

Hierbei ist insbesondere vorgesehen, dass zumindest einer der nachfolgenden Schritte durchgeführt wird, wobei vorzugsweise die Schritte nacheinander oder in beliebiger Reihenfolge durchgeführt werden, wobei bevorzugt einzelne Schritte auch wiederholt durchgeführt werden können:

- Auslesen von wenigstens zwei (insbesondere redundanten) Datenstrukturen, insbesondere einer ersten und zweiten (insbesondere redundanten) Datenstruktur, aus dem (insbesondere nicht-flüchtigen) Datenspeicher, insbesondere einem Flash-Speicher,
- Durchführen von jeweils wenigstens einer Fehlerprüfung bei jeder der Datenstrukturen, insbesondere einer ersten Fehlerprüfung bei der ersten Datenstruktur und einer zweiten Fehlerprüfung bei der zweiten Datenstruktur, sodass jeweils eine Datenkorrektheit der jeweiligen Datenstruktur (positiv oder negativ) bestätigt wird,
- Ermitteln jeweils wenigstens eines Versionsmerkmals aus jeder der Datenstrukturen, insbesondere eines ersten und zweiten Versionsmerkmals,
- Durchführen wenigstens einer Versionsprüfung bei den Datenstrukturen, insbesondere bei der ersten und zweiten Datenstruktur, anhand der (insbesondere beiden) ermittelten Versionsmerkmale, sodass eine definierte Versionsabhängigkeit zwischen den Datenstrukturen (positiv oder negativ) bestätigt wird,
- Ermitteln von jeweils wenigstens einer Nutzdateninformation aus jeder der Datenstrukturen, insbesondere einer ersten Nutzdateninformation aus der ersten Datenstruktur und einer zweiten Nutzdateninformation aus der zweiten Datenstruktur,

- Durchführen wenigstens einer Redundanzprüfung bei den Datenstrukturen, sodass eine Übereinstimmung der ermittelten Nutzdateninformationen der Datenstrukturen (positiv oder negativ) bestätigt wird, insbesondere eine Übereinstimmung der ersten Nutzdateninformation mit der zweiten Nutzdateninformation,
- Bereitstellen wenigstens einer der übereinstimmenden Nutzdateninformationen in Abhängigkeit von der Fehlerprüfung und der Versionsprüfung und der Redundanzprüfung.

Dies hat den Vorteil, dass neben der Fehlerprüfung, insbesondere zur Erkennung von Übertragungsfehlern, auch eine Manipulation der Daten anhand der Versionsprüfung erkannt werden kann. Bevorzugt erfolgt dabei eine redundante Speicherung der Nutzdateninformation derart, dass sich die Datenstrukturen trotz der übereinstimmenden Nutzdateninformationen voneinander in definierter Abhängigkeit unterscheiden, insbesondere durch Nutzung der Versionsmerkmale. Dies ermöglicht eine besonders zuverlässige Prüfung der Datenintegrität.

Die Nutzdateninformation ist insbesondere eine sicherheitsrelevante Information, vorzugsweise zur Ansteuerung sicherheitskritischer Komponenten, bevorzugt eines Fahrzeuges. Bevorzugt sind die Datenstrukturen persistent in dem Datenspeicher gespeichert, wobei der Datenspeicher vorzugsweise in der Elektronikkomponente integriert ist. Insbesondere erfolgt die Speicherung der Datenstrukturen dabei in einem geschützten Speicherbereich.

Bevorzugt erfolgt das Auslesen der Datenstrukturen initial beim Starten der Elektronikkomponente, bevorzugt durch die Initialisierung durch eine Applikationseinheit.

Des Weiteren ist es im Rahmen der Erfindung optional möglich, dass eine Treibereinheit, insbesondere ein Flash-Emulator (EEPROM-Emulator), zur Steuerung zumindest eines Datenzugriffs bei dem nicht-flüchtigen Datenspeicher vorgesehen ist, wobei vorzugsweise das Auslesen der wenigstens zwei Datenstrukturen über die

Treibereinheit erfolgt, und bevorzugt das Bereitstellen der Nutzdateninformation und/oder die Durchführung der Fehlerprüfung und/oder Versionsprüfung und/oder Redundanzprüfung separat von der Treibereinheit durch eine Schnittstelleneinheit erfolgt. Die Treibereinheit und/oder die Schnittstelleneinheit sind bevorzugt als Softwarekomponenten und/oder zumindest teilweise als Hardwarekomponenten (und/oder Elektronik) ausgeführt. Bevorzugt kann die Schnittstelleneinheit und die Treibereinheit jeweils eine Schnittstelle aufweisen, welche insbesondere gleich sind bzw. gleichartig definiert sind. Bspw. umfasst die Treibereinheit eine Treibereinheitschnittstelle gemäß einer Treibereinheitschnittstellendefinition, über welche üblicherweise eine Applikationseinheit Daten abrufen und/oder speichern kann. Insbesondere weist auch die Schnittstelleneinheit eine Schnittstelle mit dieser Treibereinheitschnittstellendefinition auf, sodass die Applikationseinheit auch über die (Schnittstelle der) Schnittstelleneinheit in gleicher Weise Daten abrufen und/oder speichern kann. Bevorzugt interagiert die Applikationseinheit (nur bzw. ausschließlich) direkt mit der Schnittstelleneinheit, wobei die Schnittstelleneinheit wiederum direkt mit der Treibereinheit interagiert, um Daten zu speichern und/oder abzurufen. Der herkömmliche Vorgang, dass die Applikationseinheit direkt über die Treibereinheit Daten abrufen und/oder speichert, ist damit verhindert. Dies gewährleistet, dass sämtliche durch die Applikationseinheit angeforderte Daten sicher abgerufen werden.

Außerdem kann es im Rahmen der Erfindung von Vorteil sein, dass bei jeder der Fehlerprüfungen (welche vorzugsweise jeweils als eine Übertragungsfehlerprüfung ausgeführt sind), bevorzugt bei der ersten und zweiten Fehlerprüfung, eine Datenauswertung (von Daten) der jeweiligen Datenstruktur erfolgt, wobei die Datenauswertung jeweils wenigstens einen der nachfolgenden Schritte umfasst:

- Ermitteln eines Prüfergebnisses anhand wenigstens einer Prüfinformation, insbesondere einer Prüfsumme (Checksumme), der jeweiligen Datenstruktur, wobei das Prüfergebnis in Abhängigkeit von den (vollständigen) Daten der jeweiligen Datenstruktur berechnet wird, und wobei vorzugsweise die vollständigen Daten zumindest die wenigstens eine Nutzdateninformation und/oder das wenigstens eine Versionsmerkmal, und insbesondere die wenigstens eine Prüfinformation, umfassen,

- Vergleichen des Prüfergebnisses mit einem Sollergebnis, welches für eine Fehlerfreiheit, insbesondere in Bezug auf Übertragungsfehler, der Daten spezifisch ist,
- Positives Bestätigen der Datenkorrektheit, wenn das Prüfergebnis mit dem Sollergebnis übereinstimmt.

Bevorzugt ist dabei das Sollergebnis, in Abhängigkeit von einer Fehlerprüfmethode, bspw. ein vordefinierter Wert, bspw. „0“. Insbesondere ist die jeweilige Prüfinformation derart in Abhängigkeit von den Daten der jeweiligen Datenstruktur gewählt, dass bei der Berechnung (bspw. gemäß der Fehlerprüfmethode) das Prüfergebnis dem Sollergebnis entspricht. Dies hat den Vorteil, dass bei einer Veränderung der Daten die Berechnung nicht mehr als Prüfergebnis das Sollergebnis ergibt, sodass die Veränderung zuverlässig festgestellt werden kann. Bevorzugt ist dabei die Fehlerprüfung lediglich für die jeweiligen Datenstrukturen (individuell bzw. einzeln) spezifisch, und somit (im Gegensatz zur Versionsprüfung) nicht für beide oder sämtliche ausgelesenen (redundanten) Datenstrukturen. Mit anderen Worten erfolgt die Fehlerprüfung der ersten Datenstruktur unabhängig von der Fehlerprüfung der zweiten Datenstruktur.

Ferner kann im Rahmen der Erfindung vorgesehen sein, dass bei der Versionsprüfung wenigstens ein erstes Versionsmerkmal einer ersten Datenstruktur mit wenigstens einem zweiten Versionsmerkmal einer zweiten Datenstruktur verglichen wird, und vorzugsweise die Versionsabhängigkeit nur dann positiv bestätigt wird, wenn ein Ergebnis des Vergleichs mit einer Abhängigkeitsvorgabe übereinstimmt. Insbesondere erfolgt dabei die Versionsprüfung derart, dass die Bestätigung der Versionsabhängigkeit von sämtlichen ausgelesenen (redundanten) Datenstrukturen, d. h. insbesondere sowohl von der ersten, als auch von der zweiten, Datenstruktur, abhängig ist. Dies hat den Vorteil, dass zusätzlich zur Feststellung von Übertragungsfehlern durch die Fehlerprüfung der einzelnen Datenstrukturen auch eine bewusste Manipulation festgestellt werden kann. Denn diese bewusste Manipulation kann insbesondere derart erfolgen, dass die Prüfinformation gefälscht wird, und das Prüfergebnis korrekt ist. In diesem Fall ermöglicht die Versionsprüfung

es insbesondere diese Manipulation zu erkennen, da eine (geheime) definierte Abhängigkeit zwischen den (unterschiedlichen und/oder redundanten) Datenstrukturen vorliegen muss. Insbesondere wenn diese Abhängigkeit nicht mehr vorhanden ist, liegt somit auch keine Datenintegrität vor.

Ferner ist es optional vorgesehen, dass bei der Redundanzprüfung eine erste Nutzdateninformation einer ersten Datenstruktur mit einer zweiten Nutzdateninformation einer zweiten Datenstruktur verglichen wird, und vorzugsweise die Übereinstimmung nur dann positiv bestätigt wird, wenn die erste Nutzdateninformation mit der zweiten Nutzdateninformation vollständig übereinstimmt, insbesondere bitweise übereinstimmt. Hierdurch ist auch eine Kontrolle möglich, dass es sich bei den ausgelesenen Datenstrukturen tatsächlich um redundante Datenstrukturen handelt.

Es kann von Vorteil sein, wenn im Rahmen der Erfindung die Fehlerprüfungen jeweils eine zyklische Redundanzprüfung anhand der Daten der jeweiligen Datenstruktur umfassen. Bspw. wird die zyklische Redundanzprüfung (englisch: cyclic redundancy check, CRC) zur Erkennung von Übertragungsfehlern, insbesondere durch den Einsatz von Prüfsummen eingesetzt.

Weiter ist im Rahmen der Erfindung denkbar, dass zum Bereitstellen (der wenigstens einen Nutzdateninformation) die Nutzdateninformation an eine Applikationseinheit übergeben wird (insbesondere über die Schnittstelle), wenn die Bestätigungen der Übereinstimmung und der Versionsabhängigkeit und der jeweiligen Datenkorrektheit der Datenstrukturen jeweils positiv sind, wobei vorzugsweise andernfalls die Elektronikkomponente in einen sicheren Zustand überführt wird. Insbesondere ist es denkbar, dass bereits bei der ersten negativen Bestätigung die Elektronikkomponente in den sicheren Zustand überführt wird (ohne dass die weiteren Prüfungen durchgeführt werden müssen). Im sicheren Zustand erfolgt bspw. ein Laden von und/oder ein Ansteuern gemäß vordefinierten Daten und/oder vordefinierten Nutzdateninformationen und/oder vordefinierten Parameter durch die

Elektronikkomponente. Somit wird gewährleistet, dass trotz fehlerhafter Daten ein sicherer Betrieb der Elektronikkomponente gewährleistet ist.

In einer weiteren Möglichkeit kann vorgesehen sein, dass für eine Speicherung der Nutzdateninformation (und/oder weiterer Nutzdateninformationen für die Elektronikkomponente) in dem nicht-flüchtigen Datenspeicher zumindest einer der nachfolgenden Schritte durchgeführt wird, wobei die Schritte vorzugsweise nacheinander oder in beliebiger Reihenfolge durchgeführt werden, und insbesondere einzelne Schritte auch wiederholt durchgeführt werden können:

- Hinterlegen der zu speichernden Nutzdateninformation als eine erste Nutzdateninformation in eine erste Datenstruktur,
- Hinterlegen der zu speichernden Nutzdateninformation als eine zweite Nutzdateninformation in eine zweite Datenstruktur, sodass die erste Nutzdateninformation der ersten Datenstruktur und die zweite Nutzdateninformation der zweiten Datenstruktur übereinstimmen, sodass insbesondere die Datenstrukturen redundant zueinander sind,
- Generieren eines ersten Versionsmerkmals für die erste Datenstruktur und eines zweiten Versionsmerkmals für die zweite Datenstruktur, insbesondere durch einen Abhängigkeitsalgorithmus (insbesondere als Abhängigkeitsvorgabe), sodass eine definierte Abhängigkeit erzeugt wird, wobei sich das erste Versionsmerkmal von dem zweiten Versionsmerkmal unterscheidet, bspw. durch einen vorgegebenen Wert als Abhängigkeitsvorgabe,
- Hinterlegen des ersten Versionsmerkmals in der ersten Datenstruktur und des zweiten Versionsmerkmals in der zweiten Datenstruktur, sodass die erste Datenstruktur gegenüber der zweiten Datenstruktur die definierte Abhängigkeit aufweist, wobei insbesondere die Versionsmerkmale jeweils als Daten der jeweiligen Datenstruktur gespeichert werden,
- Generieren einer ersten Prüfinformation für die erste Datenstruktur, sodass die erste Prüfinformation für die Daten, insbesondere für die erste Nutzdateninformation mit dem ersten Versionsmerkmal, der ersten Datenstruktur spezifisch ist,

- Generieren einer zweiten Prüfinformation für die zweite Datenstruktur, sodass die zweite Prüfinformation für die Daten, insbesondere für die zweite Nutzdateninformation mit dem zweiten Versionsmerkmal, der zweiten Datenstruktur spezifisch ist,
- Übermitteln der ersten und zweiten Datenstruktur an den nicht-flüchtigen Datenspeicher, insbesondere über eine Treibereinheit, insbesondere über eine Treibereinheitschnittstelle,

wobei vorzugsweise die Versionsmerkmale derart generiert werden, dass sie für eine vorbestimmte Anzahl von Schreibvorgängen eindeutig sind, insbesondere in Abhängigkeit von einem Schreibcounter, und wobei vorzugsweise die Prüfinformationen und/oder die Daten der Datenstrukturen sich voneinander unterscheiden. Bspw. erfolgt bei jedem Schreibvorgang der Nutzdateninformation ein Inkrementieren des Schreibcounters (Schreibzählers), insbesondere bis zu einem Maximalwert, welcher durch die vorbestimmte Anzahl bestimmt wird. Hierdurch wird die Sicherheit beim Betrieb der Elektronikkomponente deutlich erhöht.

Vorzugsweise kann vorgesehen sein, dass eine Diagnoseeinheit vorgesehen ist, wobei vorzugsweise ausschließlich durch die Diagnoseeinheit sicherheitsrelevante Schreibvorgänge und/oder eine Speicherung der Nutzdateninformation in dem nicht-flüchtigen Datenspeicher durchgeführt werden. Insbesondere erfolgt hierzu durch die Diagnoseeinheit eine Interaktion mit der Schnittstelleneinheit und/oder mit der Treibereinheit. Bevorzugt kann die Nutzung der Diagnoseeinheit (z. B. kryptografisch) abgesichert sein.

Ebenfalls Gegenstand der Erfindung ist ein Computerprogrammprodukt. Hierbei ist insbesondere vorgesehen, dass das erfindungsgemäße Computerprogrammprodukt dazu ausgeführt ist, ein erfindungsgemäßes Verfahren durchzuführen. Insbesondere ist das erfindungsgemäße Computerprogrammprodukt geeignet, einen Computer und/oder einen Prozessor und/oder eine Elektronikkomponente derart anzusteuern oder zu beeinflussen, dass das erfindungsgemäße Verfahren ausgeführt wird. Damit bringt das erfindungsgemäße Computerprogrammprodukt die gleichen Vorteile mit sich, wie sie ausführlich mit Bezug auf ein erfindungsgemäßes Verfahren beschrieben

worden sind. Insbesondere ist das Computerprogrammprodukt als computerlesbares Medium, insbesondere als ein Flash-Speicher, oder als Firmware oder als Computerprogramm ausgeführt.

Anhand der beigefügten Zeichnungen wird die Erfindung nachfolgend näher erläutert. Es zeigen:

Fig. 1 eine schematische Darstellung zur Visualisierung eines erfindungsgemäßen Verfahrens,

Fig. 2 eine schematische Darstellung zur Visualisierung der Fehlerprüfung, der Versionsprüfung und der Redundanzprüfung und

Fig. 3 eine schematische Darstellung zur Visualisierung der Datenstrukturen.

In Figur 1 ist schematisch ein erfindungsgemäßes Verfahren zur sicheren Bereitstellung von gespeicherten Informationen bei einer Elektronikkomponente 10 mit einem nicht-flüchtigen Datenspeicher 11 visualisiert. Dabei ist erkennbar, dass zum Datenabruf und/oder zur Datenspeicherung eine Applikationseinheit 400 mit einer Schnittstelleneinheit 300 interagiert, und die Schnittstelleneinheit 300 mit einer Treibereinheit 350 interagiert. Hierzu weist sowohl die Schnittstelleneinheit 300 als auch die Treibereinheit 350 jeweils eine entsprechende Schnittstelle auf, welche vorzugsweise auf einer gleichen Definition basieren. Somit ist es möglich, dass die Schnittstelleneinheit 300 in einfacher Weise durch die Applikationseinheit 400 angesteuert werden kann, auch wenn die Applikationseinheit 400 lediglich auf die Treibereinheit 350 angepasst wurde. Die Treibereinheit 350 führt dabei in direkter Weise das Schreiben und/oder Auslesen an dem Datenspeicher 11 durch. Insbesondere kann die Treibereinheit 350 hierzu in der Elektronikkomponente 10 und/oder in dem Datenspeicher 11 integriert sein, insbesondere als Softwaremodul. Somit kann zuverlässig ein Auslesen von Datenstrukturen 200 ermöglicht werden. Des Weiteren ist erkennbar, dass optional auch eine Diagnoseeinheit 500 vorgesehen sein kann.

Insbesondere ist es vorgesehen, dass für jede Nutzdateninformation 210 redundant (mindestens) zwei verschiedene Datenstrukturen 200 abgerufen werden. In Figur 2 ist gezeigt, dass zur Gewährleistung der Datenintegrität eine Fehlerprüfung 100 vorgesehen ist, welche für jede der abgerufenen (redundanten) Datenstrukturen 200 durchgeführt wird. Mit anderen Worten wird für eine erste Datenstruktur 201 eine erste Fehlerprüfung 101 und für eine zweite Datenstruktur 202 eine zweite Fehlerprüfung 102 durchgeführt. Anschließend werden für beide oder sämtliche (redundante) Datenstrukturen 200 eine Versionsprüfung 110 und/oder eine Redundanzprüfung 120 durchgeführt. Für die Versionsprüfung 110 kommen (unterschiedliche) Versionsmerkmale 220 zum Einsatz, welche insbesondere in jeder der Datenstrukturen 200 integriert sind. Die Fehlerprüfung 100 und/oder die Versionsprüfung 110 und/oder die Redundanzprüfung 120 werden dabei insbesondere durch die Schnittstelleneinheit 300 und/oder durch die Diagnoseeinheit 500 durchgeführt.

In Figur 3 ist der Aufbau der Datenstrukturen 200 verdeutlicht. Es ist erkennbar, dass eine erste Datenstruktur 201 eine erste Nutzdateninformation 211 und ein erstes Versionsmerkmal 221 und insbesondere auch eine erste Prüfinformation (nicht dargestellt) aufweist. Weiter ist dargestellt, dass eine zweite Datenstruktur 202 eine zweite Nutzdateninformation 212 und ein zweites Versionsmerkmal 222 und vorzugsweise auch eine zweite Prüfinformation (nicht dargestellt) aufweist. Bevorzugt ist die erste und zweite Nutzdateninformation 211, 212 für redundante Datenstrukturen 200 gleich, d. h. inhaltlich übereinstimmend. Diese Struktur gewährleistet die Durchführung der Datenintegritätsprüfungen, d. h. der Versionsprüfung 110 und/oder der Redundanzprüfung 120 und/oder der Fehlerprüfung 100.

Die voranstehende Erläuterung der Ausführungsformen beschreibt die vorliegende Erfindung ausschließlich im Rahmen von Beispielen. Selbstverständlich können einzelne Merkmale der Ausführungsformen, sofern technisch sinnvoll, frei miteinander kombiniert werden, ohne den Rahmen der vorliegenden Erfindung zu verlassen.

Bezugszeichenliste

10	Elektronikkomponente
11	Datenspeicher
100	Fehlerprüfung
101	erste Fehlerprüfung
102	zweite Fehlerprüfung
110	Versionsprüfung
120	Redundanzprüfung
200	Datenstruktur
201	erste Datenstruktur
202	zweite Datenstruktur
210	Nutzdateninformation
211	erste Nutzdateninformation
212	zweite Nutzdateninformation
220	Versionsmerkmal
221	erstes Versionsmerkmal
222	zweiten Versionsmerkmal
300	Schnittstelleneinheit
350	Treibereinheit
400	Applikationseinheit
500	Diagnoseeinheit

Verfahren zur sicheren Bereitstellung von gespeicherten Informationen bei einer Elektronikkomponente

Patentansprüche

1. Verfahren zur sicheren Bereitstellung von gespeicherten Informationen bei einer Elektronikkomponente (10) mit einem nicht-flüchtigen Datenspeicher (11),
gekennzeichnet durch die nachfolgenden Schritte:
 - Auslesen von wenigstens zwei Datenstrukturen (200) aus dem nicht-flüchtigen Datenspeicher (11),
 - Durchführen von jeweils einer Fehlerprüfung (100) bei jeder der Datenstrukturen (200), sodass jeweils eine Datenkorrektheit der jeweiligen Datenstruktur (200) bestätigt wird,
 - Ermitteln von jeweils wenigstens einem Versionsmerkmal (220) aus jeder der Datenstrukturen (200),
 - Durchführen einer Versionsprüfung (110) bei den Datenstrukturen (200) anhand der ermittelten Versionsmerkmale (220), sodass eine definierte Versionsabhängigkeit zwischen den Datenstrukturen (200) bestätigt wird,
 - Ermitteln von jeweils einer Nutzdateninformation (210) aus jeder der Datenstrukturen (200),
 - Durchführen einer Redundanzprüfung (120) bei den Datenstrukturen (200), sodass eine Übereinstimmung der ermittelten Nutzdateninformationen (210) der Datenstrukturen (200) bestätigt wird,
 - Bereitstellen wenigstens einer der übereinstimmenden Nutzdateninformationen (210) in Abhängigkeit von der Fehlerprüfung (100) und der Versionsprüfung (110) und der Redundanzprüfung (120).

2. Verfahren nach Anspruch 1,
dadurch gekennzeichnet,
dass eine Treibereinheit (350), insbesondere ein Flash-Emulator, zur Steuerung zumindest eines Datenzugriffs bei dem nicht-flüchtigen Datenspeicher (11) vorgesehen ist, wobei das Auslesen der wenigstens zwei Datenstrukturen (200) über die Treibereinheit (350) erfolgt, und das Bereitstellen der Nutzdateninformation (210) und/oder die Durchführung der Fehlerprüfung (100) und/oder Versionsprüfung (110) und/oder Redundanzprüfung (120) separat von der Treibereinheit (350) durch eine Schnittstelleneinheit (300) erfolgt.

3. Verfahren nach Anspruch 1 oder 2,
dadurch gekennzeichnet,
dass bei jeder der Fehlerprüfungen (100), vorzugsweise als Übertragungsfehlerprüfungen (100), eine Datenauswertung der jeweiligen Datenstruktur (200) erfolgt, wobei die Datenauswertung jeweils wenigstens einen der nachfolgenden Schritte umfasst:
 - Ermitteln eines Prüfergebnisses anhand einer Prüfinformation, insbesondere einer Prüfsumme, der jeweiligen Datenstruktur (200), wobei das Prüfergebnis in Abhängigkeit von den vollständigen Daten der jeweiligen Datenstruktur (200) berechnet wird, und wobei die vollständigen Daten zumindest die Nutzdateninformation (210) und das Versionsmerkmal (220) und insbesondere die Prüfinformation umfassen,
 - Vergleichen des Prüfergebnisses mit einem Sollergebnis, welches für eine Fehlerfreiheit, insbesondere in Bezug auf Übertragungsfehler, der Daten spezifisch ist,
 - Positives Bestätigen der Datenkorrektheit, wenn das Prüfergebnis mit dem Sollergebnis übereinstimmt.

4. Verfahren nach einem der vorhergehenden Ansprüche,
dadurch gekennzeichnet,
dass bei der Versionsprüfung (110) wenigstens ein erstes Versionsmerkmal (221) einer ersten Datenstruktur (201) mit wenigstens einem zweiten Versionsmerkmal (222) einer zweiten Datenstruktur (202) verglichen wird, und vorzugsweise die Versionsabhängigkeit nur dann positiv bestätigt wird, wenn ein Ergebnis des Vergleichs mit einer Abhängigkeitsvorgabe übereinstimmt.
5. Verfahren nach einem der vorhergehenden Ansprüche,
dadurch gekennzeichnet,
dass bei der Redundanzprüfung (120) eine erste Nutzdateninformation (211) einer ersten Datenstruktur (201) mit einer zweiten Nutzdateninformation (212) einer zweiten Datenstruktur (202) verglichen wird, und vorzugsweise die Übereinstimmung nur dann positiv bestätigt wird, wenn die erste Nutzdateninformation (211) mit der zweiten Nutzdateninformation (212) vollständig übereinstimmt.
6. Verfahren nach einem der vorhergehenden Ansprüche,
dadurch gekennzeichnet,
dass die Fehlerprüfungen (100) jeweils eine zyklische Redundanzprüfung (120) anhand der Daten der jeweiligen Datenstruktur (200) umfassen.
7. Verfahren nach einem der vorhergehenden Ansprüche,
dadurch gekennzeichnet,
dass zum Bereitstellen die Nutzdateninformation (210) an eine Applikationseinheit (400) übergeben wird, wenn die Bestätigungen der Übereinstimmung und der Versionsabhängigkeit und der jeweiligen Datenkorrektheit der Datenstrukturen (200) jeweils positiv sind, wobei vorzugsweise andernfalls die Elektronikkomponente (10) in einen sicheren Zustand überführt wird.

8. Verfahren nach einem der vorhergehenden Ansprüche, **dadurch gekennzeichnet**, dass für eine Speicherung der Nutzdateninformation (210) in dem nicht-flüchtigen Datenspeicher (11) zumindest einer der nachfolgenden Schritte durchgeführt wird:
- Hinterlegen der zu speichernden Nutzdateninformation (210) als eine erste Nutzdateninformation (211) in eine erste Datenstruktur (201),
 - Hinterlegen der zu speichernden Nutzdateninformation (210) als eine zweite Nutzdateninformation (212) in eine zweite Datenstruktur (202), sodass die erste Nutzdateninformation (211) der ersten Datenstruktur (201) und die zweite Nutzdateninformation (212) der zweiten Datenstruktur (202) übereinstimmen,
 - Generieren eines ersten Versionsmerkmals (221) für die erste Datenstruktur (201) und eines zweiten Versionsmerkmals (222) für die zweite Datenstruktur (202), insbesondere durch einen Abhängigkeitsalgorithmus, sodass eine definierte Abhängigkeit erzeugt wird, wobei sich das erste Versionsmerkmal (221) von dem zweiten Versionsmerkmal (222) unterscheidet,
 - Hinterlegen des ersten Versionsmerkmals (221) in der ersten Datenstruktur (201) und des zweiten Versionsmerkmals (222) in der zweiten Datenstruktur (202), sodass die erste Datenstruktur (201) gegenüber der zweiten Datenstruktur (202) die definierte Abhängigkeit aufweist,
 - Generieren einer ersten Prüfinformation für die erste Datenstruktur (201), sodass die erste Prüfinformation für die Daten, insbesondere für die erste Nutzdateninformation (211) mit dem ersten Versionsmerkmal (221), der ersten Datenstruktur (201) spezifisch ist,
 - Generieren einer zweiten Prüfinformation für die zweite Datenstruktur (202), sodass die zweite Prüfinformation für die Daten, insbesondere für die zweite Nutzdateninformation (212) mit dem zweiten Versionsmerkmal (222), der zweiten Datenstruktur (202) spezifisch ist,

- Übermitteln der ersten und zweiten Datenstruktur (201, 202) an den nicht-flüchtigen Datenspeicher (11), insbesondere über eine Treibereinheit (350),

wobei vorzugsweise die Versionsmerkmale (220) derart generiert werden, dass sie für eine vorbestimmte Anzahl von Schreibvorgängen eindeutig sind, insbesondere in Abhängigkeit von einem Schreibcounter, und wobei vorzugsweise die Prüfinformationen und/oder die Daten der Datenstrukturen (200) sich voneinander unterscheiden.

9. Verfahren nach einem der vorhergehenden Ansprüche, **dadurch gekennzeichnet**, dass eine Diagnoseeinheit (500) vorgesehen ist, wobei ausschließlich durch die Diagnoseeinheit (500) sicherheitsrelevante Schreibvorgänge und/oder eine Speicherung der Nutzdateninformation (210) in dem nicht-flüchtigen Datenspeicher (11) durchgeführt werden.
10. Computerprogrammprodukt, welches dazu ausgeführt ist, ein Verfahren nach einem der vorhergehenden Ansprüche durchzuführen.

Zusammenfassung

Verfahren zur sicheren Bereitstellung von gespeicherten Informationen bei einer Elektronikkomponente (10) mit einem nicht-flüchtigen Datenspeicher (11), mit den nachfolgenden Schritten:

- Auslesen von wenigstens zwei Datenstrukturen (200) aus dem nicht-flüchtigen Datenspeicher (11),
- Durchführen von jeweils einer Fehlerprüfung (100) bei jeder der Datenstrukturen (200), sodass jeweils eine Datenkorrektheit der jeweiligen Datenstruktur (200) bestätigt wird,
- Ermitteln von jeweils wenigstens einem Versionsmerkmal (220) aus jeder der Datenstrukturen (200),
- Durchführen einer Versionsprüfung (110) bei den Datenstrukturen (200) anhand der ermittelten Versionsmerkmale (220), sodass eine definierte Versionsabhängigkeit zwischen den Datenstrukturen (200) bestätigt wird,
- Ermitteln von jeweils einer Nutzdateninformation (210) aus jeder der Datenstrukturen (200),
- Durchführen einer Redundanzprüfung (120) bei den Datenstrukturen (200), sodass eine Übereinstimmung der ermittelten Nutzdateninformationen (210) der Datenstrukturen (200) bestätigt wird,
- Bereitstellen wenigstens einer der übereinstimmenden Nutzdateninformationen (210) in Abhängigkeit von der Fehlerprüfung (100) und der Versionsprüfung (110) und der Redundanzprüfung (120).

(Hierzu Fig. 1)



1/3

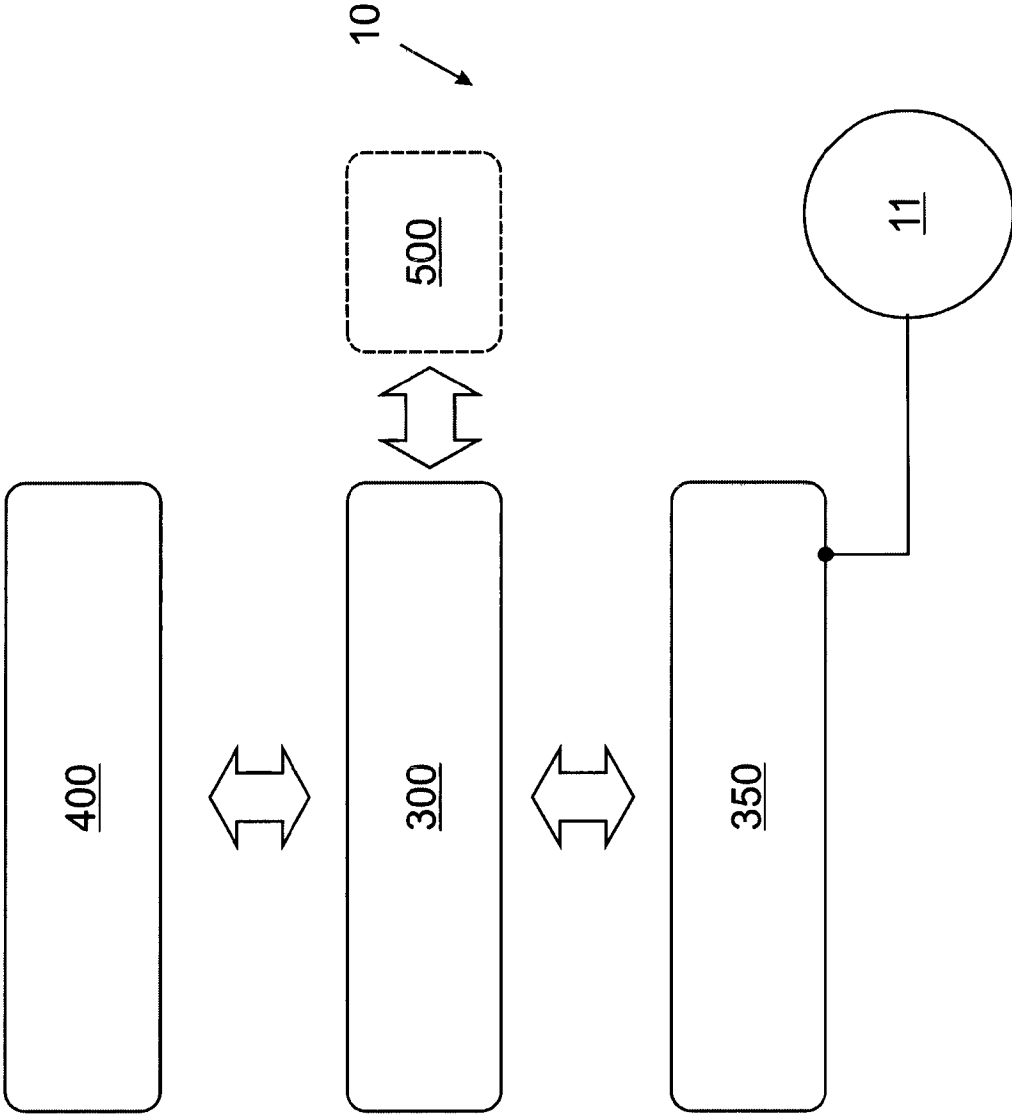


Fig. 1

2/3

300

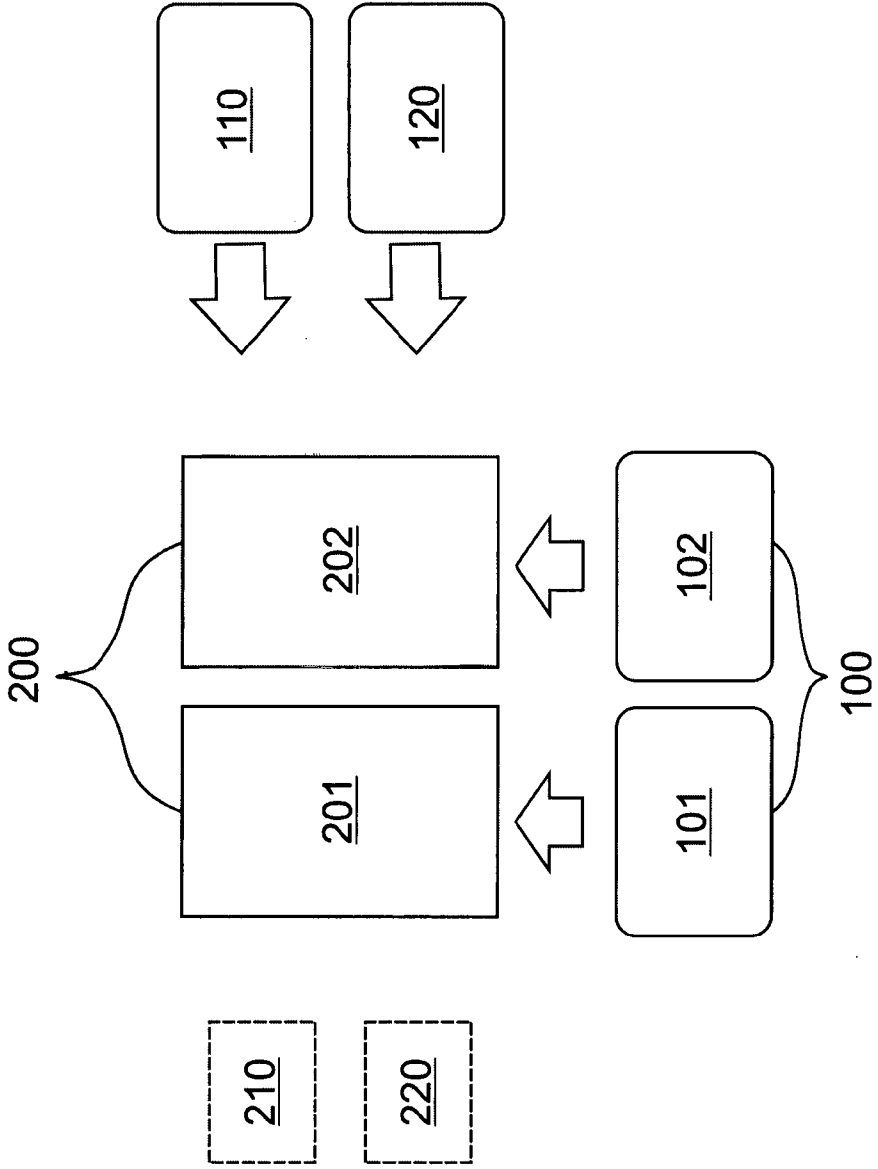


Fig. 2

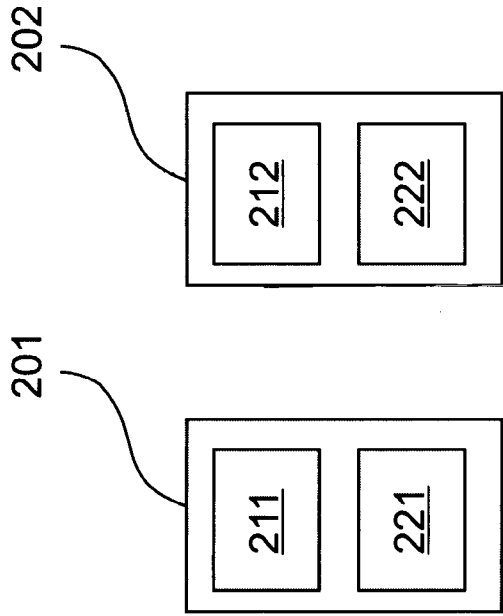


Fig. 3